BrightStor® ARCserve® Backup for Windows

Agent for Microsoft SQL Server

r11.5

D01173-2E

Contents

Chapter 1: Introducing the Agent

BrightStor® ARCserve® Backup is a comprehensive, distributed storage solution for applications, databases, distributed servers, and file systems. It provides backup and restore capabilities for databases, business-critical applications, and network clients.

Among the agents BrightStor ARCserve Backup offers is the BrightStor® ARCserve® Backup Agent for Microsoft SQL Server. This agent enables you to perform the following actions:

- Back up your Microsoft SQL Server databases using BrightStor ARCserve Backup without taking your database off line or preventing users from adding new data
- Manage backups of Microsoft SQL Server databases remotely
- Schedule backups
- Back up to a wide array of media storage devices
- Restore Microsoft SQL Server databases using BrightStor ARCserve Backup

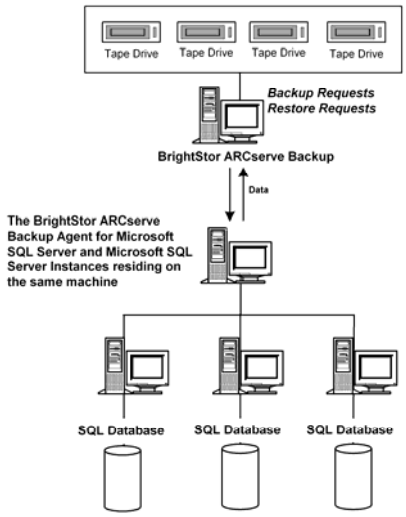
The agent handles all communications between BrightStor ARCserve Backup and Microsoft SQL Server during backup and restore jobs, including preparing, retrieving, and processing the data packets that are sent back and forth between Microsoft SQL Server and BrightStor ARCserve Backup.

Architectural Overview

BrightStor ARCserve Backup can be installed on the same host as the Agent for Microsoft SQL Server for local operation or they can be installed on separate systems. A single installation of BrightStor ARCserve Backup can work with agents on many systems, allowing multiple machines to be backed up by a single backup server. BrightStor ARCserve Backup and the agent work together to back up and restore SQL database objects.

The agent provides services that allow BrightStor ARCserve Backup to back up and restore a SQL database associated with the agent. The agent may reside on the same server as Microsoft SQL Server or on each local drive of all nodes in a SQL cluster environment. BrightStor ARCserve Backup manages both the agent and Microsoft SQL Server.

Architecturally, the agent is positioned between BrightStor ARCserve Backup and Microsoft SQL Server, as the following diagram illustrates:



How the Agent Works

BrightStor ARCserve Backup and the agent work together to back up and restore SQL Server databases. When BrightStor ARCserve Backup backs up a database, it sends a request to the agent. The agent retrieves the database from Microsoft SQL Server and sends it to BrightStor ARCserve Backup, where the complete database is backed up to media. During a restore, the agent functions in a similar fashion and helps transfer the backed up database from BrightStor ARCserve Backup to Microsoft SQL Server.

The agent takes advantage of the Microsoft SQL Server dump database and transaction log backup method (commonly called the *dump*). A dump backs up the database or transaction log in a single step. This ensures that a consistent image of the database is backed up.

For each database submitted for backup, the agent initiates a dump in Microsoft SQL Server. Microsoft SQL Server sends the database to the agent in multiple data chunks. The agent receives the data, one chunk at a time, and passes it directly to BrightStor ARCserve Backup, where it is recorded to backup media.

How a Backup Request Works

Backup jobs use the following process:

1. You issue a backup command from BrightStor ARCserve Backup.
2. BrightStor ARCserve Backup sends the request to the agent for a database.
3. The agent retrieves a particular database or log from Microsoft SQL Server, which sends multiple data chunks to the agent.
4. The agent retrieves the data chunks and transfers them to BrightStor ARCserve Backup, which backs up the data to the specified storage media.

How a Restore Request Works

Restore jobs use the following process:

1. You issue a restore command from BrightStor ARCserve Backup.
2. BrightStor ARCserve Backup informs the agent of the restore job.
3. The agent instructs Microsoft SQL Server to prepare to receive the data.
4. BrightStor ARCserve Backup accesses storage media and begins restoring data.
5. BrightStor ARCserve Backup transfers data to the agent.
6. The agent transfers data to Microsoft SQL Server.
7. Microsoft SQL Server recovers the database.

How Data Flows During Backup

The following steps describe the data flow when BrightStor ARCserve Backup uses the Agent for Microsoft SQL Server to back up a Microsoft SQL Server instance:

1. BrightStor ARCserve Backup sends a request to the agent for a database.
2. The agent instructs Microsoft SQL Server to perform a backup of a particular database or log.
3. Microsoft SQL Server returns the data from the database in multiple chunks to the agent, one chunk at a time.

4. The agent receives the data chunks from Microsoft SQL Server and transfers them to BrightStor ARCserve Backup.
5. BrightStor ARCserve Backup writes the data chunks to media.

These steps are repeated until there is no more data to be backed up. The agent and the Microsoft SQL Server backup function guarantee the consistency and accuracy of the data being backed up.

Agent Services

The Agent for Microsoft SQL Server consists of the following services:

- **CA BrightStor Backup Agent RPC (Remote Procedure Call) Server Service:** The CA BrightStor Backup Agent RPC Server service operates as a Windows service. This service starts automatically after the installation of the Agent for Microsoft SQL Server is completed. This service allows the Agent for Microsoft SQL Server to browse, back up, and restore using named pipes. This service can be shared with other agents.
- **CA BrightStor Backup Agent Remote Service:** The CA BrightStor Backup Agent Remote Service starts automatically after the installation of the Agent for Microsoft SQL Server is completed. This service backs up Microsoft SQL Server databases remotely using the TCP/IP protocol.

Access Requirements

When you submit a job that includes remote Windows database servers, BrightStor ARCserve Backup prompts you for a default user name and password for the system on which the database resides. BrightStor ARCserve Backup accesses the remote servers using this user name and password.

A remote Microsoft SQL Server user ID and password are also required to access remote database servers. When prompted by the system, enter the Microsoft SQL Server user ID and the password of the system administrator (sa), or enter a user ID and password with equivalent privileges. This user may be a Windows user, depending on security settings. For more information about agent security configuration, see the topic Configuring Microsoft SQL Server Security Settings in this guide.

Agent Activity Log

The Agent for Microsoft SQL Server generates a log with information about backup or restore jobs and their status. This Activity log is called dbasql.log and is located in the directory in which the agent has been installed. If errors appear in the BrightStor ARCserve Backup job logs, check the Activity log for more information about why the errors occurred.

Additional Functionality

When you use the Agent for Microsoft SQL Server with Microsoft SQL Server 2000 and Microsoft SQL Server 2005, you can use multiple instance support to perform backups and restores of databases on named SQL Server instances. For more information, see the topics Multiple Instance Support and Backup and Restore Options in this guide.

When you use the Agent for Microsoft SQL Server with the BrightStor ARCserve Backup Enterprise Module and Microsoft SQL Server 7.0, Microsoft SQL Server 2000, or Microsoft SQL Server 2005, you can also use the Multistriping feature. For information about this feature, see the topic Multistriping Support in this chapter.

Multistriping Support

When the agent is used with the multistreaming feature of the BrightStor ARCserve Backup Enterprise Module, you can use multiple processes and multiple backup devices to accelerate your backup beyond the speed of a single tape drive. With very large databases, this can make the difference between a backup that takes a few hours and a backup that takes an entire day.

Backup and Restore Options

Backup options enable you to do the following:

- Back up differential files or FileGroups of a database
- Back up transaction logs
- Leave the database in a restoring state
- Check the physical consistency of databases

For more information about backup options, see the chapter, ”Backing Microsoft SQL Server Databases”.

Restore options enable you to do the following:

- Restore data and stop before or at a specific mark
- Restore data using replication settings
- Restore with restricted user access
- Partially restore data

- Move log files
- Check the physical consistency of databases
- Automatically determine a sequence of backups to restore to produce a live, consistent database from a single restore job

For more information about restore options, see the chapter, ''Restoring Microsoft SQL Server Databases."

Online Backup of SAP R/3 Databases

When you use Microsoft SQL Server 7.0 or Microsoft SQL Server 2000 as the database server for SAP R/3, you can perform an online backup of SAP R/3 databases using the Agent for Microsoft SQL Server. A separate agent for SAP R/3 is not required. The online backup procedure is the same as it is with any other database in the Microsoft SQL Server.

Note: You cannot perform offline backups of SAP R/3 databases in Microsoft SQL Server 7.0 or Microsoft SQL Server 2000 using the Agent for Microsoft SQL Server.

Chapter 2: Installing the Agent

The Agent for Microsoft SQL Server is a client program you can install in two configurations:

- On the server machine with Microsoft SQL Server
- On the local drives of all nodes in a Microsoft SQL cluster environment

This chapter explains how to install the Agent for Microsoft SQL Server in both configurations.

Installation Prerequisites

This section provides prerequisite information you must satisfy before installing the Agent for Microsoft SQL Server in a standard Microsoft SQL environment, Microsoft SQL Server 7.0 cluster environment, or a Microsoft SQL Server 2000 cluster environment.

Basic Prerequisites

Before you install the Agent for Microsoft SQL Server in a standard Microsoft SQL environment, verify the following:

- You have administrator privileges or the proper authority to install software on the computers where you will be installing the product or its components. Contact your systems administrator to obtain the proper rights if you do not have them.
- You have used SQL or Windows authentication for each Microsoft SQL Server instance. For each Microsoft SQL Server instance with Microsoft SQL authentication, provide a user name and password of a Microsoft SQL Server user with system administrator privileges.
- You have the necessary licensing and registration information for the Agent for Microsoft SQL Server.
- Your system meets the minimum requirements needed to install the Agent for Microsoft SQL Server. For a list of requirements, see the readme file.
- You have determined whether you are installing the Agent for Microsoft SQL Server in a standard Microsoft SQL Server environment, in a Microsoft SQL Server 7.0 cluster environment, or in a Microsoft SQL Server 2000 cluster environment.
- You have installed BrightStor ARCserve Backup. For information about installing BrightStor ARCserve Backup, see the *Getting Started* guide.

- You have determined whether you plan to use multistriping. If so, install the BrightStor ARCserve Backup Enterprise Module. For information about installing the Enterprise Module, see the *Getting Started* guide.
- You have selected one of the following types of installation:
  - Local installation
  - Remote installation
  - Create a response file (silent installation)
- You have noted the installation path for easy reference if you are changing the default installation path.
- You have noted the computer name and the valid user name and password for the computers on which you are installing the Agent for Microsoft SQL Server.

Microsoft SQL Server 7.0 Cluster Environment Prerequisites

Before you install the Agent for Microsoft SQL Server in a Microsoft SQL Server 7.0 cluster environment, in addition to the basic prerequisites, perform the following tasks:

- Ensure that your Microsoft SQL Server 7.0 virtual server has Mixed Mode authentication selected. For specific instructions on checking and changing this setting, see Check or Change the Microsoft SQL Server Authentication Method in the appendix ''Configuring Microsoft SQL Server Security Settings".

  If you change the setting, stop and restart Microsoft SQL Server services from the Microsoft Cluster Administrator to allow this change to take effect.
- Use SQL authentication, rather than Windows authentication.
- Install Microsoft SQL Server Client Connectivity to all secondary nodes of the cluster. This enables a backup to occur if Microsoft SQL Server Quorum and the Windows Quorum are on separate nodes of the cluster.
- Make a note of the Microsoft SQL Server virtual server name and the user name and password of a Microsoft Clustering Server (MSCS) domain user with administrator privileges.
- Make a note of the user name and password of a Microsoft SQL Server user with system administrator privileges.
- Install the agent on the local drives of all nodes in the MSCS cluster as part of the initial agent installation.
- Select local as your installation type if you are installing on a node of a Microsoft SQL cluster environment.

Microsoft SQL Server 2000 Cluster Environment Prerequisites

Before you install the Agent for Microsoft SQL Server in a Microsoft SQL Server 2000 cluster environment, in addition to the basic prerequisites, perform the following tasks:

- Make a note of the user name and password of an MSCS domain user with system administrator privileges.
- Select Windows authentication for the Microsoft SQL Server 2000 instance.
- Make a note of the Microsoft SQL 2000 virtual server name, cluster server user name, and cluster server password.
- Install the Agent for Microsoft SQL Server on the local drives of all nodes in the MSCS cluster as part of the initial agent installation.
- Select local as your installation type if you are installing on a node of a Microsoft SQL cluster environment.

Install the Agent

Ensure that you have confirmed the installation prerequisites and performed the required pre-installation tasks. When you have completed these tasks and gathered the required information, you are ready to begin the installation process.

Note: If you have multiple versions of Microsoft SQL Server installed on the same machine, the version of SQLVDI.dll registered with the machine must be from the latest version of Microsoft SQL Server. If it is not, backup operations can fail.

Install the Agent in a Standard Microsoft SQL Server Environment

To install the Agent for Microsoft SQL Server in a standard Microsoft SQL Server environment, follow the standard installation procedure for the system components, agents, and options of BrightStor ARCserve Backup. For the detailed steps in this procedure, see the *Getting Started* guide.

During the installation procedure, after you select the Agent for Microsoft SQL Server for installation, the Account Configuration dialog appears.

Enter the appropriate information for each instance of your standard Microsoft Server:

- Select either SQL or Windows Authentication.
- Enter the user name and password of a Microsoft SQL Server user with system administrator privileges for each Microsoft SQL Server instance for which you have specified Microsoft SQL authentication.

Install the Agent in a Microsoft Server 7.0 Cluster Environment

To install the Agent for Microsoft SQL Server in a Microsoft SQL Server 7.0 cluster environment, follow the standard installation procedure for the system components, agents, and options of BrightStor ARCserve Backup. For the detailed steps in this procedure, see the *Getting Started* guide.

During the installation procedure, after you select the Agent for Microsoft SQL Server for installation, the Account Configuration dialog appears.

Enter the appropriate cluster information for the Microsoft SQL Server 7.0 virtual server you must specify for each node on the cluster:

- Enter Microsoft SQL Authentication for a Microsoft SQL Server 7.0 virtual server.
- Enter the user name and password for the system administrator (sa) or an equivalent account on the Microsoft SQL virtual server.
- Enter the Microsoft SQL virtual server name, and the user name and password of an MSCS domain user with system administrator privileges.

Install the Agent in a Microsoft Server 2000 Cluster Environment

To install the Agent for Microsoft SQL Server in a Microsoft SQL Server 2000 cluster environment, follow the standard installation procedure for the system components, agents, and options of BrightStor ARCserve Backup. For the detailed steps in this procedure, see the *Getting Started* guide.

During the installation procedure, after you select the Agent for Microsoft SQL Server for installation, the Account Configuration dialog appears.

Enter the appropriate cluster information for each instance of your Microsoft SQL Server 2000 virtual server:

- Click the cell containing the instruction in the Instance column to add Microsoft SQL virtual server instances to the configuration window.
- Specify either Windows NT or Microsoft SQL authentication in the Authentication column. If you specify Microsoft SQL authentication, enter the user name and password of a Microsoft SQL Server user with system administrator (sa) rights for that instance.
- Enter the name of the Microsoft SQL 2000 virtual server associated with this instance.
- Enter the login ID of an MSCS domain user with system administrator privileges and the password for that user. Confirm the password.

Post-Installation Procedures

After you have installed the agent, you may need to perform one or more of the following post-installation tasks:

- Configure the agent backup and restore parameters for Microsoft SQL Server 7.0 and Microsoft SQL Server 2000 cluster environments
- Configure the TCP port address
- Configure parameters to use multistriping

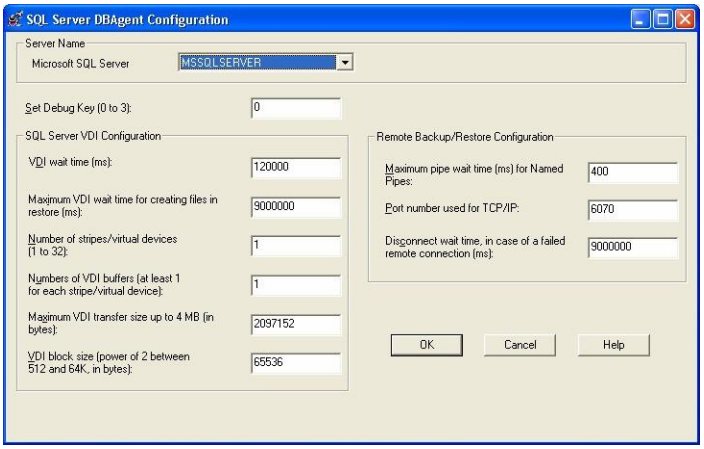
Configure Backup and Restore Parameters

The Microsoft SQL Agent Configuration utility enables you to configure the Agent for Microsoft SQL Server backup and restore parameters for Microsoft SQL Server 7.0 and Microsoft SQL Server 2000. The parameters include settings for Microsoft Virtual Device Interface (VDI) objects and remote communication.

1. From the BrightStor ARCserve Backup Program group in the Start menu, start the utility.

   Note: From the Start menu, choose Programs, Computer Associates, BrightStor, ARCserve Backup Agents. From ARCserve Backup Agents, open Microsoft SQL Agent Configuration.

   The Microsoft SQL Agent Configuration utility opens the SQL Server DBAgent Configuration dialog:



2. In the Server Name field, select the default instance (MSSQLSERVER) or an instance name (for Microsoft SQL Server 2000 and SQL Server 2005) for which the Agent for Microsoft SQL Server configuration is to be changed.

3. The Set Debug Key (0 to 3) field is used to create a detailed debug log. Change this parameter only when asked to do so by Customer Support.

4. In the SQL Server VDI Configuration fields, configure the Agent for Microsoft SQL Server for backup or restore operations using the VDI. Use the SQL Server VDI Configuration section to set the following parameters:

   - **VDI wait time (ms)**: The time, in milliseconds, a VDI object waits. The default setting is 120000 ms.
   - **Maximum VDI wait time for creating files in restore (ms)**: The time, in milliseconds, a VDI object waits before timing out when creating data files during a restore. Increase this time if the database to be restored contains very large data files. The default setting is 9000000 ms.

- **Number of stripes/virtual devices (1 to 32)**: The number of threads used to retrieve data from Microsoft SQL Server. This parameter determines the number of CPUs used to perform backups. Set this value to match the number of CPUs in the database server for the fastest backup performance. The default setting is 1.
- **Number of VDI buffers (at least 1 for each stripe/virtual device)**: The total number of buffers (of maximum transfer size) used to back up and restore. The default setting is 1.
- **Maximum VDI transfer size up to 4 MB (in bytes):** The maximum input or output request issued by Microsoft SQL Server to the device. This is the data portion of the buffer. This parameter value must be a multiple of 64 KB. The range is from 64 KB to 4 MB. The default setting is 2097152 or 2 MB.
- **VDI block size (power of 2 between 512 and 64 KB bytes):** All data transfers are in integral multiples of this value. Values must be a power of 2 between 512 bytes and 64 KB inclusive. The default is 65536 or 64 KB.

5. In the fields in the Remote Backup/Restore Configuration section, configure the Agent for Microsoft SQL Server for remote backup and restore jobs. Use the Remote Backup/Restore Configuration fields to set the following parameters:
   - **Maximum pipe wait time (ms) for Named Pipes:** The time, in milliseconds, the Agent for Microsoft SQL Server waits to close a named pipe if a remote connection fails. The default setting is 400 ms.
   - **Port Number used for TCP/IP:** The port number for remote backup and restore using TCP/IP. For more information about port number, see TCP Port Address Configuration in this chapter. The default setting is 6070.
   - **Disconnect wait time, in case of a failed remote connection (ms):** The time, in milliseconds, the Agent for Microsoft SQL Server waits to close a TCP/IP session if a remote connection fails. The default setting is 9000000.

Agent Services

The Agent for Microsoft SQL Server consists of the following services:

- **CA BrightStor Backup Agent RPC (Remote Procedure Call) Server Service:** The CA BrightStor Backup Agent RPC Server service operates as a Windows service. This service starts automatically after the installation of the Agent for Microsoft SQL Server is completed. This service allows the Agent for Microsoft SQL Server to browse, back up, and restore using named pipes. This service can be shared with other agents.
- **CA BrightStor Backup Agent Remote Service:** The CA BrightStor Backup Agent Remote Service starts automatically after the installation of the Agent for Microsoft SQL Server is completed. This service backs up Microsoft SQL Server databases remotely using the TCP/IP protocol.

Configure the TCP Port Address

To change the port number for the Agent Remote service, follow these steps:

1. On the server where the Agent for Microsoft SQL Server is installed, from the Start menu, select the Microsoft SQL Agent Configuration utility.

   Note: From the Start menu, choose Programs, Computer Associates, BrightStor, ARCserve Backup Agents. From ARCserve Backup Agents, open Microsoft SQL Agent Configuration.

   The SQL Server DBAgent Configuration dialog appears.

2. Specify or change the value in the Port Number Used for TCP/IP field in the Remote Backup/Restore Configuration section.

   Note: If you have more than one instance of Microsoft SQL Server installed, you must have the same port number for every instance.

3. Stop and restart the CA BrightStor Backup Agent Remote Service and CA BrightStor Backup Agent RPC Server service.

4. Update settings on the backup server with the revised port number for this agent. For information on configuring the settings, see the *Administrator Guide*.

Configure Multistriping Parameters

You can fine-tune the performance and operations of your backup jobs to run on multiple devices. Tune these settings immediately after installing the Agent for Microsoft SQL Server or at a later time, depending on your specific database needs. To fine-tune these settings, perform the following procedure:

1. From the Start menu, open the Agent for Microsoft SQL Configuration utility.

   Note: From the Start menu, choose Programs, Computer Associates, BrightStor, ARCserve Backup Agents. From ARCserve Backup Agents, open Microsoft SQL Agent Configuration.

   The SQL Server DBAgent Configuration dialog appears.

2. We recommend that you make the following changes in the Microsoft SQL Server DBAgent Configuration dialog:

   - Increase the VDI wait time from the default setting of 2 minutes (120000 ms) to 20 minutes (1200000) if you have very large databases or slow tape changers. If a backup must span from one tape to another, the default time may be insufficient for some changers to replace a full tape with a blank tape.
   - Increase the number of VDI buffers to double the number of tape devices you have attached to the system running BrightStor ARCserve Backup and the Agent for Microsoft SQL Server. The number of VDI buffers used for a multistriping backup or restore is the number of VDI buffers set here unless the number of devices used for the backup or restore is higher.
   - Increase the maximum VDI wait time if your databases have very large data files. The default value of 9000000 ms (2 1/2 hours) is typically adequate for most databases. However, if data files exceed 200 GB, additional time may be required, depending on CPU and disk speed, or restore operations may fail due to timeouts while waiting for the Microsoft SQL Server to create data files.

Chapter 3: Backing Up Microsoft SQL Server Databases

This chapter contains information about backing up databases and Transaction logs using BrightStor ARCserve Backup, the Agent for Microsoft SQL Server, and Microsoft SQL Server 7.0, Microsoft SQL Server 2000, or Microsoft SQL Server 2005.

Backup Overview

To *back up* is to create a copy of a database, Transaction log, database differential, or collection of files or FileGroups on another device (typically a media drive). Use BrightStor ARCserve Backup and the Agent for Microsoft SQL Server to perform backups using the SQL Server Backup statement.

Backing up a database creates a copy of its tables, data, and user-defined objects. In the event of media failure, if you made regular backups of your databases and their Transaction logs, you can recover your databases.

Important! *Transaction logs are not backed up or truncated during full or differential database backups. To back up and truncate Transaction logs, perform a separate Transaction log backup. When you perform the Transaction log backup, select the Remove inactive entries from Transaction log option to truncate the log files. For more information about Transaction log backups see Transaction Log Backups in this chapter.*

When a Microsoft SQL Server database backup is started in BrightStor ARCserve Backup, the Agent for Microsoft SQL Server initiates an online backup of the database. This backup can take place while the database is active. The backup captures the state of the data at the moment the statement is executed. No partial transactions are captured. Any data changes made after the backup begins are not captured in the backed up copy of the database.

Types of Backups

The Agent for Microsoft SQL Server supports the following types of backups:

- **Database Complete**: Backs up the entire database.
- **Database Differential**: Backs up data that has changed since the last complete backup. For example, if you ran a complete backup of your database on Sunday night, you can run a differential backup on Monday night to back up only the data that changed on Monday.

- **Transaction Log**: Backs up the Transaction log. Transaction log backups provide the following options:
  - **Remove inactive entries from transaction log**: Truncates the log files. This is the default option.
  - **Do not remove inactive entries from transaction log**: Retains inactive log entries after backup. These entries are included in the next Transaction log backup.
  - **Back up only the log tail and leave the database in unrecovered mode**: Backs up the log and leaves the database in a restoring state. This option is available for Microsoft SQL Server 2000 only. Use this option to capture activity since the last backup and take the database offline to restore it.
- **Files and FileGroups**: Backs up selected files in a database. Use this option to back up a file or FileGroup when the database size and performance requirements make it impractical to perform a full database backup.
- **Files and FileGroups; Differential**: Backs up data changed in selected files since the last File and FileGroup backup. Differential file backups reduce recovery time by reducing the number of transactions from the Transaction log that must be restored. This option is available for Microsoft SQL Server 2000 only.

Backup Strategy Recommendations

To establish a good backup strategy, follow these recommendations:

- If your database activity is low to medium, we recommend the following frequency:
  - Full backups: once per week
  - Differential backups: once per day
  - Transaction Log backups: every two to four hours
- If your database activity is high and your database is of small to medium size, we recommend the following frequency:
  - Full backups: twice per week
  - Differential backups: twice per day
  - Transaction Log backups: every 60 minutes

- If your database activity is high for a large size database using the Full or Bulk-Logged Recovery model, we recommend the following frequency:
  - Full backup: once per week
  - Differential backup: once per day
  - Transaction Log backup: every 20 minutes
- If your database activity is high for a large size database using the Simple Recovery model, we recommend the following frequency:
  - Full backup: once per week
  - Differential backup: twice per day

Required Full Backup

After you perform certain database management actions, your next backup **must** be a full database backup. If you perform one of these actions and then perform a differential backup, Transaction log backup, or files and FileGroups backup, you may be unable to use that backup with your last full database backup to restore the database successfully.

To prevent this problem, always perform a full database backup immediately after you have performed any of the following actions:

- Created a new database
- Changed the recovery model of the database
- Changed the number of files or FileGroups in the database
- Changed the arrangement of files among the FileGroups
- Canceled a full backup job while it is running
- Modified the structure of the database, including adding or removing a table or a column in a table
- Added or removed an index
- Restored the database from backups

Differential Backups

A differential backup records only the data that has changed since the last full database backup. On average, these backups are smaller and typically take less time to complete than full database backups, although they are typically larger and take more time to complete than Transaction log backups. To restore a database, a differential backup requires only the last full backup, and does not need any of the other differential or Transaction log backups performed since the last backup. A differential backup is also faster to restore than a Transaction log backup because transactions do not need to be reprocessed.

Note: If a database is very active or if a long time has passed since the last full backup, a differential backup could take as long as a full backup.

Differential Backup Timing

Perform differential backups as a supplement to full backups. Because they are usually faster and smaller, you can perform them more often than full database backups. They are also typically more efficient because they require less space on media and have a briefer impact on database performance than frequent, full database backups. Additionally, you can use them to minimize the number of Transaction logs you need to recover during a restore, because you need to restore only those Transaction log backups performed since the differential backup.

Differential backups are most beneficial under the following circumstances:

- A relatively small portion of the data in the database has changed since the last database backup. Differential database backups are most efficient if the same data is modified frequently.
- You are using the Simple Recovery model, which does not permit Transaction log backups, and you want to perform backups more frequently than is practical for full database backups.
- You are using the Full or Bulk-Logged Recovery model and want to minimize the time it takes to roll forward Transaction log backups when restoring a database.

Note: After you have modified the structure or configuration of the database (for example, by adding more data or log files or changing the recovery model), you must perform a full database backup before performing a differential or Transaction log backup.

Transaction Log Backups

Transaction logs contain the record of Microsoft SQL Server database activity; back them up frequently. To back them up, run self-contained Transaction log backups separately from database backups. Transaction log backups offer the following advantages over other types of backups:

- Generally faster than differential backups
- Typically faster and smaller than full database backups (unless they have not been truncated recently)
- Typically, less impact on database performance while running
- Normally able to be restored to a specific point in time, rather than only to the time the backup was made

Important! *Transaction logs are not backed up during full or differential database backups. You must back them up by running separate Transaction log backups.*

After you have modified the structure or configuration of the database (for example, by adding more data or log files, or changing the recovery model), you must perform a full database backup before performing a differential or Transaction log backup.

Restore Requirements for Transaction Log Backups

To restore a Transaction log backup, you must first restore the following:

- The last full database backup performed
- The last differential backup performed since that full database backup, if any
- Any other Transaction log backups performed since the last full database or differential backup

It takes longer to recover a database when you restore the database and several Transaction logs than it does when you restore only the database. Finding the correct strategy depends on your environment. You must consider the time required to perform backups in relation to the time required to restore.

Important! *Do not perform a Transaction log backup until you have performed at least one full database backup.*

Truncate Transaction Logs

You can truncate Transaction logs when you back them up. To truncate a Transaction log, select the Remove inactive entries from transaction log option when you configure the backup. If the Transaction log is not truncated, it may eventually grow large.

Multiple Instance Support

This feature provides backup and restore support on multiple instances of Microsoft SQL Server running concurrently on the same computer, with each instance having its own set of system and user databases that are not shared between instances. An application can connect to each Microsoft SQL Server instance on a local computer in the same way that it connects to Microsoft SQL Server running on a remote computer.

The Agent for Microsoft SQL Server offers backup and restore support for multiple Microsoft SQL Server instances. The Backup Manager displays instances for the local computer and for the remote computer. The default instance is called Microsoft SQL Server, while named instances append their instance names.

File and FileGroup Backups

You can choose to back up one or more FileGroups or individual files when the database size and performance requirements make it impractical to perform a full database backup.

If you choose to back up an individual file instead of the full database, put procedures in place to ensure that all files in the database are backed up regularly, and perform separate Transaction log backups for the databases whose files or FileGroups you back up individually. After restoring a file backup, you must apply the Transaction log to roll the contents of the file forward to make it consistent with the rest of the database. For more information, see the Microsoft SQL Server documentation.

The Impact of the Create Index Statement on File and FileGroup Backups

The Backup statement requires that you back up entire FileGroups affected by a Create Index statement. This requirement exists in the following situations:

- If you create an index on a FileGroup, you must back up that entire FileGroup in a single backup operation. Microsoft SQL Server does not allow backups of individual files that are part of the affected FileGroup.
- If you create an index on a FileGroup separate from the FileGroup in which the table resides, then you must backup both FileGroups (the FileGroup containing the table and the FileGroup containing the newly created index) together.
- If you create more than one index on a FileGroup separate from the FileGroup in which the table resides, you must back up all the FileGroups immediately to accommodate these different FileGroups.

The Backup statement detects all of these FileGroup situations and communicates the minimum number of FileGroups that you must back up. Microsoft SQL Server reports this information to the user when the backup job is run.

Database Consistency Checks

When your database activity is low, you should run a database consistency check (DBCC), particularly with a large database. Although it takes some time, it is important to determine that your Microsoft SQL Server database is functioning well.

A DBCC tests the physical and logical consistency of a database. When you enable the Database Consistency Check option for a backup, the DBCC performs the following tests:

- **DBCC CHECKDB**: Checks the allocation and structural integrity of all objects in the specified database. By default, the CHECKDB performs a check for indexes that can increase the overall execution time.

  Note: The system table indexes are checked regardless of whether you select this option.
- **DBCC CHECKCATALOG**: Checks for consistency in and between system tables in the specified database.

Database Consistency Check (DBCC) Options

A DBCC tests the physical and logical consistency of a database. DBCC provides the following options:

- **Before Backup**: Checks consistency before the backup of the database.
- **After Backup**: Checks consistency after the backup of the database.
- **Continue with backup, if DBCC fails**: Performs a database backup even if the consistency check fails.
- **After restore**: Performs DBCC after the restore of the database.
- **Do not check indexes**: Checks the database for consistency without checking indexes for user-defined tables.
- **Check only the physical consistency of the database**: Detects torn pages and common hardware failures. In addition, it checks the integrity of the physical structure of the page and record headers, and the consistency between the page's object ID and index ID. This feature is available for Microsoft SQL Server 2000 only.

All error messages that are generated during the DBCC are displayed in the Agent for Microsoft SQL Server log file called dbasql.log. The log is located in the Backup Agent directory.

Back Up Databases

The following procedure provides the basic steps required to back up a database:

1. Ensure that Microsoft SQL Server is running on your server. The Microsoft SQL Server service must be started.
2. Start the CA BrightStor Backup Agent RPC Server service and CA BrightStor Backup Agent Remote Service, if necessary.

   Note: These services are started automatically when the agent is installed and are set to start automatically if the machine is restarted.
3. Open the Backup Manager and locate the Microsoft SQL Server instance, listed under the name of the computer on which it is installed. Expand the Microsoft SQL Server instance to display a list of databases.

4. If you are backing up from a remote Microsoft SQL Server database, right-click the Microsoft SQL Server instance and select Remote Protocol from the pop-up menu. The Remote Protocol dialog opens. Select a protocol and click OK.

   If you are **not** backing up from a remote Microsoft SQL Server database, go on to the next step.

   The default protocol is TCP/IP.

   Note: Select Named Pipes only if you have backup operator rights, but do not have system administrator rights, or if you cannot use TCP/IP. The system automatically tries to use named pipes if it cannot connect to the Agent for Microsoft SQL Server using TCP/IP. Named Pipes is not available for Microsoft SQL Server 2005.

5. Choose a database under the Microsoft SQL Server instance. Information about the selected database appears in the right pane of the Backup Manager.

   Note: For information about selecting a database to apply options properly for backup, see Dynamic and Explicit Job Packaging in this chapter.

6. Right-click the database object and select Backup Agent Options from the pop-up menu. The Backup Agent Options dialog for Microsoft SQL Server 7.0 Microsoft SQL Server 2000 opens, depending on the version of Microsoft SQL Server you are using.

7. Select the type of backup you want to perform. For more information about backup types, see Backup Types in this chapter.

8. If you chose the Files/FileGroups or Files/FileGroups - Differential backup type, click the Browse or Browse Files/FileGroups button. The Specify FileGroups and Files dialog opens.

   Select the specific files and FileGroups you want to back up and click OK.

9. (Optional) Enable a Database Consistency Check and select the Database Consistency Check options. For more information about Database Consistency Checks, see Database Consistency Checks in this chapter and your Microsoft SQL documentation. Click OK.

10. Repeat the preceding steps for each database you are backing up in this job.

11. On the Destination tab in the Backup Manager, select a backup destination.

    Note: You can use the * symbol in the Group or Media fields to create partial wildcards when you select a backup destination. For example, if you have two sets of device groups, one with all members beginning GroupA and the other with all members beginning GroupB, you can select all the GroupA members by entering GroupA* in the Group field. For more information about selecting devices and media, see the *Administrator Guide*.

12. Click the Schedule tab and select the scheduling options for this backup. For information about scheduling backups, see the *Administrator Guide*.

13. Click Start. The Security and Agent Information dialog opens.

    Note: In this dialog, the column and button labeled Agent refer to the Client Agent for Windows, not the Agent for Microsoft SQL Server. You can edit client agent information at this time. For more information about client agents, see the *Administrator Guide*.

14. Verify the user name and password for the target machine and for Microsoft SQL Server. To change the security information for Microsoft SQL Server, click Security and change the information in the dialog.

15. After verifying or changing the security information, click OK. The Submit Job dialog opens.

16. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.

17. Click OK. The Job Status window opens. Use this window to monitor the current status of your job. For more information about the Job Status window, see the *Administrator Guide*.

Backup Considerations

You should back up a database immediately after you create it and continue to back it up on a regular schedule to ensure smooth recovery from a database or media failure. Maintain regular backups of all databases, including:

- The master, msdb, and model databases
- All user databases
- The distribution database (if the server is configured as a replication distributor)

Note: Microsoft SQL Server 2005 Mirror databases and report snapshots cannot be backed up and do not appear in the database list. For more information about database mirroring, see the Microsoft SQL Server 2005 documentation.

Rotation Schemes and Global Options

BrightStor ARCserve Backup can use incremental and differential global backup methods when backing up Microsoft SQL Server, allowing you to use a rotation scheme to perform differential and Transaction log backups of Microsoft SQL Server databases, dynamically adjusting for the limitations of each individual database.

Note: For more information about rotation schemes and automatic backups, see the *Administrator Guide.*

You can use the following backup methods:

- **Full Backup Method**: If you specified backup options for the target database, these options are applied. Otherwise, a full database backup is performed by default.
- **Differential Backup Method:**
  - If you are backing up a system database (for example, master, model, or msdb), a full database backup is always performed.
  - If the Microsoft SQL Server has no record of a prior full database backup of the target database, a full database backup is performed.
  - If the backup options selected for the target database include a selection of specific database files and FileGroups, a File-and-FileGroup differential backup is performed for Microsoft SQL 2000 databases, and a File-and-FileGroup full backup is performed for Microsoft SQL 7.0 databases.
  - In all other circumstances, a differential database backup is performed.
- **Incremental Backup Method:**
  - If you are backing up a system database (for example, master, model, or msdb), a full database backup is always performed.
  - If the Microsoft SQL Server has no record of a prior full database backup of the target database, a full database backup is performed.
  - If the database uses the Simple Recovery Model, a differential database backup is performed.
  - In all other circumstances, a Transaction log backup with truncation is performed.

Important! *Performing a backup can slow the system down. Run backups when the database is not being heavily updated.*

Dynamic and Explicit Job Packaging

BrightStor ARCserve Backup provides two ways to package or set up your backup jobs:

- Dynamic job packaging
- Explicit job packaging

Dynamic Job Packaging

If you mark a database instance for dynamic job packaging when defining a backup job, BrightStor ARCserve Backup automatically selects, at the time the backup job runs, all the components of the object (for example, child volumes and files) for backup.

For example, if you choose to back up an entire server and mark the server for dynamic job packaging and you change the volumes on the server, when you run the next backup job, the volumes on the server at the time of the backup are the volumes that are backed up. Any change that you make on the server marked for dynamic job packaging is included in the next backup.

Important! *When you mark a parent object for dynamic job packaging, all of its associated (or child) objects are also marked for dynamic job packaging and are selected for backup. Child objects of an object marked for dynamic packaging lose any separate options you have assigned them when the job is submitted.*

Mark Objects for Dynamic Job Packaging

To mark an object for dynamic job packaging, follow these steps:

1. On the Source tab in the Backup Manager, expand the directory tree until the object you want to mark for dynamic job packaging is displayed.
2. Click the square next to the object. The square next to the object, and the squares next to all the children of the object, become completely green. In the following example, Microsoft SQL Server has been marked for dynamic job packaging. All of its children are also marked for dynamic job packaging.

Explicit Job Packaging

If you mark a database object for explicit job packaging when defining your backup job, you mark some or all of its child objects for dynamic job packaging, but do not mark the parent.

For example, if you select only the C and E drives to back up on a server, the server, which is the parent, is packaged explicitly. If you add another drive to the server between the time you scheduled the job and the time it runs, the new drive is not included in the job. However, because you dynamically packaged the C and E drives, any change in the contents of those two drives between the time you scheduled the job and the time you run it, is included in the backup job.

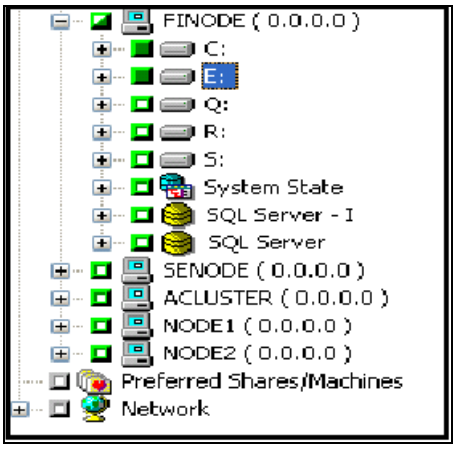
Explicit job packaging gives you the ability to customize local backup options. For example, if you run a backup job for which you have dynamically packaged the C and E drives on your server (which is explicitly packaged), you can select one set of options for drive C and another set of options for drive E.

Note: To customize volume or database options, you must package the volume or database parent items explicitly.

Mark Objects for Explicit Job Packaging

To mark an object for explicit job packaging, follow these steps:

1. On the Source tab in the Backup Manager, expand the directory tree until the object you want to mark for explicit job packaging is displayed.
2. Click the squares next to the children of the object. The squares next to the child objects become completely green and the square next to the parent object becomes half green and half white. In the following example, the C and E drives have been marked for dynamic job packaging. The computer on which they exist, FINODE, has been marked for explicit job packaging.



Chapter 4: Restoring Microsoft SQL Server Databases

This chapter contains information about restoring databases and Transaction logs using BrightStor ARCserve Backup, the Agent for Microsoft SQL Server, and Microsoft SQL Server 7.0, Microsoft SQL Server 2000 or Microsoft SQL Server 2005.

Restore Overview

To *restore* is to load a database from a backup of that database and (if applicable) one or more backups of its Transaction log. If a database is lost or damaged, you can restore the database by reloading the most recent database backup and the successive log backups. A restore overwrites any information in the database with the backed up information. Use BrightStor ARCserve Backup and the Agent for Microsoft SQL Server to perform restore operations using the Microsoft SQL Server Restore statement.

When you restore a database, Microsoft SQL Server rolls back any uncommitted transactions that were active at the moment the restore job began. When the restore operation is complete, the database is in the same state it was in when the Backup statement for the backup used in the restore job was initiated, excluding any transactions that were active at that point.

As the data from the backup is been reloaded, Microsoft SQL Server reinitializes any remaining unused pages. For example, if a 100 MB database contains only five MB of data, Microsoft SQL Server rewrites all 100 MB of space. Consequently, it takes at least as long to restore a database as it does to create a database.

Microsoft SQL Server locks a database while restoring it, so that the database cannot be modified during the restore operation. However, users can access and modify other Microsoft SQL Server databases during this time.

Note: If users access a database when a restore job is running, Microsoft SQL Server stops the restore operation.

If a failure occurs while a database is being restored, Microsoft SQL Server notifies the system administrator but does not recover the partially restored database. You must restart the database restore to complete the restore job.

Note: If you cancel a restore job, the database is left in a loading state and is unusable until the restore sequence is completed. If the session that was being restored when the job was cancelled is not the first session in the restore sequence, you may have to start the restore sequence over from the beginning.

The destination database must have at least as much storage space allocated to it as the backed up database. The actual amount of data in the backed up database is irrelevant. To get information about allocated storage space, use the Microsoft SQL Enterprise Manager or the DBCC CHECKALLOC statement.

If you have a media failure, restart Microsoft SQL Server. If, after a media failure, Microsoft SQL Server cannot access a database, it marks the database as suspect, locks it, and displays a warning message. You may have to drop (detach from Microsoft SQL Server) a damaged database, a process that you can perform using the Microsoft SQL Server Enterprise Manager.

During a restore operation, the selected database must not be in use. Any data in the selected database is replaced by the restored data.

Restore Types

The Agent for Microsoft SQL Server supports the following types of restores:

- **Complete database restore**: Restores the entire database.
- **Differential backup restore**: Restores the database to the point in time when you created the differential backup. Typically, you would supplement differential backups by creating multiple Transaction log backups after each database backup. To use this restore type, you must restore the last complete database backup before restoring the differential backup. Using a combination of full-database, differential, and Transaction log backups, you can minimize the time of database recovery and the amount of data loss due to failure.
- **Transaction log restore**: Restores the Transaction log. Restoring a Transaction log is also referred to as "applying" a Transaction log. When you restore a Transaction log, Microsoft SQL Server re-executes the changes contained in the log and rolls back any transactions that were uncommitted when you backed up the Transaction log.

  After you have restored a database, you can load the differential backup (if any) and the Transaction log backups you created after backing up that database. Loading Transaction logs lets you to recover as much of a database as possible.

You must load backups of the Transaction log in the sequence in which they were created. Microsoft SQL Server checks the timestamps on each backed up database and each backed up Transaction log to verify that the sequence is correct.

After Microsoft SQL Server has loaded the entire sequence of Transaction log backups, the database is restored to its state at the time of the last Transaction log backup, excluding any uncommitted transactions. The only uncommitted transaction that Microsoft SQL Server does not roll back is the Microsoft SQL Backup Log transaction, which is completed as part of the restore process instead.

- **File and FileGroup restore:** Restores files or FileGroups. You can restore files and FileGroups from either a file or FileGroup backup or from a full database backup. When restoring files or FileGroups, you must apply all of the differential file or FileGroup backup sessions, followed by all of the Transaction log backup sessions performed after the file or FileGroup backup. When you restore a file or FileGroup, you must apply the Transaction log to the database files immediately after the last file or FileGroup operation.
- **Partial restore**: A partial restore always restores the primary FileGroup and other files you specify and their corresponding FileGroups. The result is a subset of the database. FileGroups that are not restored are marked as off line and are not accessible.

Important! *When using Automatic Selection, you may be unable to restore a database to a different location on a disk (for example, to a different drive letter or directory path, or with a different file name) using backups from previous versions of BrightStor ARCserve Backup or BrightStor® Enterprise Backup. For more information about restoring to a different location, see Restore to Alternative Disk Locations Using Automatic Selection in this chapter.*

Differential Backup Restores

A differential backup contains only the data that has changed since the last full backup. If you have performed multiple differential backups after a full backup, you need only the last differential backup and the last full backup to restore the database to its most recent state.

When you have selected a differential backup session to restore, the Automatic Selection option automatically selects the appropriate full-database backup session and the proper options. Automatic Selection ensures that the correct sessions are restored in your job. Although you can package the appropriate sessions manually, Automatic Selection can save time and prevent errors.

When you restore a differential backup, the database must not be in use. Any data in the specified database is replaced by the restored data. If you are not using Automatic Selection, the database must be in a Loading state from a Full Database restore.

Unlike Transaction log restores, differential restores restore your data only to the point in time when you created the differential backup, not to the exact point of failure.

Transaction Log Restores

When restoring from a Transaction log backup, you must apply the Transaction log backup to the appropriate full database, differential backup, or FileGroup backup. When restoring, you must restore your data in the following order:

- Restore the full database backup
- Restore the latest differential backup, if any
- Restore the Transaction log backups made since the full or differential backup

When you select a Transaction log backup to restore, the Automatic Selection option automatically selects the appropriate Transaction log backup, differential backup, and database backups, and the proper options. Automatic Selection ensures that the correct sessions are restored in your job. Although you can package the appropriate sessions manually, Automatic Selection can save time and prevent errors.

File and FileGroup Restores

You can restore individual files or FileGroups from either a file or FileGroup backup or from a full database backup. When using this option, you must apply the Transaction log to the database immediately after the last file or FileGroup restore operation. This allows the contents of the file to roll forward, making it consistent with the rest of the database.

When you have selected a file or FileGroup backup to restore and selected Automatic Selection, the Automatic Selection option automatically selects all of the Transaction log backups needed to run the restore successfully. Automatic Selection ensures that your job restores the correct sessions. Although you can package the appropriate sessions manually, Automatic Selection can save time and prevent errors.

When you have selected a file or FileGroup differential backup and Automatic Selection, the Automatic Selection option selects the file or FileGroup backup session on which the differential was based and all of the Transaction log backups needed to run the restore successfully.

Restore Master Databases in Windows NT

When you restore a master database in a Windows NT Version 4.0, you must start the Microsoft SQL Server service in single-user mode. To start Microsoft SQL Server service in single-user mode, follow these steps:

1. From the Control Panel, open Services. The Service dialog opens.
2. Select one of the following Microsoft SQL Server services:
   - **MSSQLServer**: For Microsoft SQL Server 7.0.
   - **MSSQLSERVER**: For a default instance of Microsoft SQL Server 2000.
   - **MSSQL$**: Named instances of Microsoft SQL Server 2000. The name of the instance follows $.
3. Stop the Microsoft SQL Server service if it is running. To stop the service, click Stop on the General Tab in the Service Properties dialog.
4. In the Startup parameters field at the bottom of the dialog, enter the following:

   ```
   -c -m
   ```

5. Click Start to restart Microsoft SQL Server.

Important! *Ensure that none of the supporting Microsoft SQL Server services are running during the master database restore job or the restore job will fail. If you are restoring the msdb database from a backup, you must stop the SQLServerAgent service before the restore begins. Restart the service after the restore is complete. If you do not stop SQLServerAgent, it will not function properly after you have restored the master database.*

Return Microsoft SQL Server Service to Normal (Multi-User) Mode

When you finish restoring the master database, you must return the Microsoft SQL Server service to normal (multi-user) mode.

1. From the Control Panel, open Services. The Services dialog opens.
2. Select one of the following Microsoft SQL Server services:
   - **MSSQLServer**: For Microsoft SQL Server 7.0.
   - **MSSQLSERVER**: For a default instance of Microsoft SQL Server 2000.
   - **MSSQL$**: Named instances of Microsoft SQL Server 2000. The name of the instance follows $.
3. If the Microsoft SQL Server service is running, stop the service. To stop the service, click the service and click Stop. Note that the Startup Parameters field is cleared.
4. Click Start to restart Microsoft SQL Server.

Restore Master Databases in Windows 2000, Windows XP, or Windows 2003

Before you restore the master database, you may need to rebuild it from Microsoft SQL Server 7.0 or Microsoft SQL Server 2000 using the Microsoft SQL Server Rebuild Master utility. Restoring the master database requires that the restore operation have exclusive access to the SQL Server instance. To ensure this, you must start Microsoft SQL Server in single-user mode and restore the master database from the most recent backup. For complete instructions on rebuilding your Microsoft SQL Server 7.0 or Microsoft SQL Server 2000 master database, see the Microsoft SQL Server documentation.

Important! *Microsoft SQL Server Desktop Engine (MSDE) does not contain the Rebuild Master Database utility. To recreate the master and model databases using MSDE, you must reinstall your MSDE-based application, then restore the appropriate online backups.*

A permissions conflict may occur when you run Microsoft SQL Server from a command line as system administrator while the Agent for Microsoft SQL Server is running as a service. Use the following procedure to run Microsoft SQL Server in single-user mode as a service to avoid the permissions conflict.

Note: The following procedure for restoring a master database does not apply to Windows NT environments. For information about restoring a master database in a Windows NT environment, see Restore Master Databases in Windows NT in this chapter.

To start Microsoft SQL Server in single-user mode, follow these steps:

1. From the Control Panel open Administrative Tools and open Services. The Services window opens.
2. Select one of the following Microsoft SQL Server services:
   - **MSSQLServer:** For Microsoft SQL Server 7.0.
   - **MSSQLSERVER:** For a default instance of Microsoft SQL Server 2000.
   - **MSSQL$:** Named instances of Microsoft SQL Server 2000. The name of the instance follows $.
3. Right-click the service. From the pop-up menu, choose Properties. The Service Properties dialog opens.
4. Stop the Microsoft SQL Server service if it is running. To stop the service, click Stop on the General Tab in the Service Properties dialog.

5. In the Start parameters field, enter the following for Microsoft SQL Server 7.0 or Microsoft SQL Server 2000:

   ```
   -c -m
   ```

   For Microsoft SQL Server 2005, enter the following:

   ```
   -m
   ```

6. Click Start to restart the service and click OK.

Return Microsoft SQL Server Service to Normal (Multi-User) Mode

When you finish restoring the master database, you must return Microsoft SQL Server service to normal (multi-user) mode. To return Microsoft SQL Server service to multi-user mode, follow these steps:

1. From the Start menu, choose Settings, Control Panel. From the Control Panel open Administrative Tools.
2. Open Services. The Services window opens.
3. Select one of the following Microsoft SQL Server services:
   - **MSSQLServer:** For Microsoft SQL Server 7.0.
   - **MSSQLSERVER:** For a default instance of Microsoft SQL Server 2000.
   - **MSSQL$:** Named instances of Microsoft SQL Server 2000. The name of the instance follows $.
4. Right-click the service. From the pop-up menu, choose Properties. The Service Properties dialog opens.
5. Remove the entry from the Start parameters field.
6. Click Start to restart the service and click OK.

For more information about this type of restore operation, as well as other special cases, see the Microsoft SQL Server documentation.

Restore Options

The Agent for Microsoft SQL Server provides the following restore options:

- Automatic Selection
- Restore Type (Full or File and FileGroup)
- Force Restore Over Existing Files
- Log Point in Time Restore
- Restore Database Files As

- Recovery Completion State
- Database Consistency Check
- Restricted User Access After Restore (Microsoft SQL Server 2000 and 2005 only)
- Keep Replication Settings (Microsoft SQL Server 2000 and 2005 only)

Automatic Selection Option

The Automatic Selection option automatically selects:

- Other sessions that must be restored with the session you are restoring for the restore job to be successful
- Appropriate options for the restore job

The Automatic Selection option is enabled by default for every restore job. Using Automatic Selection saves you time and prevents errors in packaging restore jobs.

Important! *Using Automatic Selection, you may be unable to restore a database to a different location on a disk (for example, to a different drive letter or directory path, or with a different file name) For more information about restoring to a different location, see Restore to Alternative Disk Locations Using Automatic Selection in this chapter.*

Restore Type Option

The Restore Type option lets you select the type of restore from the following choices:

- **Database option:** Use the Database option to restore complete databases, differential backups, and Transaction log backups.
- **Files or FileGroups option:** Use the File or FileGroups option to restore file or FileGroup backups or files that belong to database backups, but not files that belong to log backups or differential backups. After you have selected the File or FileGroups option, you must select the files you want to restore.
- **Partial Restore option:** In Microsoft SQL Server 2000 or Microsoft SQL Server 2005, use the Partial Restore option to:
  - Restore part of a database to its original location
  - Restore part of a database to another location so you can copy damaged or missing data back to the original database

  You can use this option with full database backup sessions only.

Requirements to Restore Files or FileGroups

After restoring a file or FileGroup, you must apply a Transaction log session to roll the contents of the file forward to make it consistent with the rest of the database. Therefore, each time you back up a file or FileGroup, you should perform a Transaction log backup immediately afterwards.

Microsoft SQL Server requires that you restore all FileGroups, for which you created indexes since their last backup, in a single operation. You must fulfill this requirement whether you are restoring from a FileGroup backup or from a full database backup. Microsoft SQL Server detects the FileGroup index and compiles a list of the FileGroups you must restore. If you do not fulfill this requirement, Microsoft SQL Server reports this information to you when the restore is run. See the Agent for Microsoft SQL Server Activity log for the complete results.

For more information about the requirements for restoring files and FileGroups, see the Microsoft SQL Server documentation.

Force Restore Option

The Force Restore Over Existing Files option lets Microsoft SQL Server overwrite files it does not recognize as part of the database it is restoring. Use this option only if you receive a message from Microsoft SQL Server prompting you to use the With Replace option.

Microsoft SQL Server supports this option for database restore and file or FileGroup restore operations.

Log Point in Time Restore Option

The Log Point in Time Restore option restores a database to the state it was in at a date and time you have specified. You should use Automatic Selection with this option.

Important! *You cannot use the Log Point in Time Restore option if the database you are recovering uses the Bulk-Logged recovery model.*

To find the correct log when you have selected the Log Point in Time Restore option, Microsoft SQL Server restores the record in each Transaction log backup containing the start and finish time of the backup. Microsoft SQL Server then searches this record for the time you have specified.

- If Microsoft SQL Server finds the specified time, it restores the log to the point in the record that contains the time you submitted. The agent then signals BrightStor ARCserve Backup to stop restoring and the database is fully recovered. If there are other logs with the same time, those logs are ignored and the subsequent sessions are skipped.

- If the specified time occurs after those contained in the log, Microsoft SQL Server restores the tog and leaves the database in a restoring state, waiting for the next log restore operation.
- If the specified time occurs before those contained in the log, Microsoft SQL Server cannot restore the log.

The Log Point in Time Restore option has limitations. For example, if you do not select Automatic Selection and choose the Force Restore Over Existing Files option and you restore one or more logs belonging to the same database, but you do not select the appropriate database, differential backup, and FileGroup sessions to be restored first, the job will be incomplete and the subsequent sessions for that database will be ignored.

The options available for Log Point in Time Restore are:

- **Stop at time**: This option includes date and time fields in which you can enter a specific date and time. The option recovers the database to the specified date and time. This is the default option.
- **Stop before log mark:** This option includes date and time fields in which you can set a specific date and time mark. The option recovers the database to the specified mark but does not include the transaction that contains the mark. If you do not select the After datetime option, recovery stops at the first mark with the specified name. If you select the After datetime option, recovery stops at the first mark with the specified name exactly at or after the date and time.

  Note: This option is available in Microsoft SQL Server 2000 and Microsoft SQL Server 2005.
- **Stop at log mark:** This option includes date and time fields in which you can set a specific date and time mark. The option recovers the database to the specified mark, including the transaction that contains the mark. If you do not select the After datetime option, recovery stops at the first mark with the specified name. If you select the After datetime option, recovery stops at the first mark with the specified name exactly at or after the date and time.

  Note: This option is available in Microsoft SQL Server 2000 and Microsoft SQL Server 2005.
- **After Time**: This option includes date and time fields in which you can set a specific date and time mark. The recovery stops at the specified mark only when the timestamp for the log mark is later than the specified time. Use this option with the Stop at Log Mark or Stop Before Log Mark options.

Restore Database Files As Option

Using the Restore Database Files As option, you can:

- View the list of files for the database and related information.
- Select the files to be restored in a file or FileGroup backup session or in a file or FileGroup or Partial Restore of a database backup session.
- Change the location or file name of files during a restore. Use the location change function only with complete database restores or when using Automatic Selection.

Important! *If you are restoring from a log or differential backup made with a previous version of BrightStor ARCserve Backup or BrightStor Enterprise Backup, you may not be able to restore a database to a different location on a disk (for example, with a different file name or to a different drive letter, or directory path) using Automatic Selection. For more information about restoring to a different location, see Restore to Alternative Disk Locations Using Automatic Selection in this chapter.*

Recovery Completion State Option

The Recovery Completion State option lets you specify the final state of a session restore using the following options:

- **Leave database operational. No additional transaction logs can be restored:** This option instructs the restore operation to roll back any uncommitted transactions**.** After the recovery process, the database is ready for use.

  Note: If you use Automatic Selection, you do not have to choose any of the Recovery Completion State selections manually because BrightStor ARCserve Backup performs the selection of sessions and the necessary options automatically. If you do not choose Automatic Selection, you must follow Microsoft SQL Server rules regarding the restore flow. For more information, see Microsoft SQL Server documentation.

- **Leave database nonoperational, but able to restore additional transaction logs:** This option instructs the restore operation **not** to roll back any uncommitted transactions. You must choose either this option or the Leave database read-only option to apply another differential backup or Transaction log. Microsoft SQL Server requires that you use this option for all but the final session. You must also use this option for database restores with multiple Transaction log restores or when a restore requires multiple sessions (for example, a full database backup followed by a differential backup).

- **Leave database read-only and able to restore additional transaction logs:** This option prepares a standby (warm backup) server. A standby server is a second server that you can bring on line if the primary production server fails. It contains a copy of the databases on the primary server. For more information about standby servers, see Microsoft SQL Server documentation.

Database Consistency Check (DBCC) Options

When you enable the database consistency check option for a restore, it performs the following test:

- **DBCC CHECKDB:** Checks the allocation and structural integrity of all the objects in a specified database. By default, the CHECKDB performs a check for indexes that can increase the overall execution time.

  Note: The system table indexes are checked regardless of whether you select this option.

- **DBCC CHECKCATALOG**: Checks for consistency in and between system tables in the specified database.

Restricted User Access After Restore Option

Available for Microsoft SQL Server 2000 and Microsoft SQL Server 2005, the Restricted User Access after Restore option restricts access to a newly restored database to members of the db_owner, dbcreator, or sysadmin roles. In Microsoft SQL Server 2000 and Microsoft SQL Server 2005, Restricted_User replaces the DBO_Only option from Microsoft SQL Server 7.0. This option requires the Leave database operational, no additional transaction logs can be restored option.

Keep Replication Settings Option

Available for Microsoft SQL Server 2000 and Microsoft SQL Server 2005, the Keep Replication Settings option instructs the restore operation to preserve replication settings when restoring a published database to a server other than the one on which it was created. As a result, it prevents Microsoft SQL Server from resetting the replication settings when it restores a database or log backup on a warm standby server and recovers the database. Use the Keep Replication Settings option when setting up replication to work with log shipping.

You cannot select this option when restoring a backup with the Leave database non-operational, but able to restore additional transaction logs option. Use this option only with the Leave database operational, no additional transaction logs can be restored option.

SQL Agent Filter Options

You can use the SQL Agent Filter option in the Restore Manager to display backup sessions of a particular database belonging to a particular server name and, for Microsoft SQL Server 2000 and Microsoft SQL Server 2005, a particular instance. This option is available when you are using the Restore by Session method.

To display backup sessions of a database belonging to a server name, follow these steps:

1. Open the Restore Manager and select Filter from the Restore menu.
2. In the Filter dialog, select the SQL Agent Filter tab.
3. If you are using Microsoft SQL Server 7.0, enter a machine name, or a machine name and database name.

   If you are using Microsoft SQL Server 2000 or Microsoft SQL Server 2005, enter a machine name and a database name, or a machine name, instance name, and database name.

   Note: If you have several databases with the same matching character set in the name, they are all displayed.
4. Click OK.

Note: Once you apply the filter settings, expand the media item to view the results. If the media item was already expanded, collapse it and expand it again to see the results.

Database Restore Overview

If a database is damaged or lost, you can restore the database by reloading the most recent database backup, the latest differential backup, and the succeeding Transaction log backups. If you have corrupted data in the database, you can reload a database backup over the former database.

When you restore a database from a backup, that database must not be in use because any data in the specified database is replaced by the restored data.

Restore Types and Methods

You can restore from the following types of backup sessions:

- Complete database backups
- Differential backups
- Transaction log backups
- File and FileGroup backups

For any type of restore operation, you can use the following restore methods:

- **Restore by Tree:** This option displays a tree of the networks and computers backed up by BrightStor ARCserve Backup. To run a restore operation, expand the networks and computers to select the databases or files you want to restore. The databases are from the most recent backup sessions. This option is selected by default.
- **Restore by Session:** This option displays a list of media you have used when backing up with BrightStor ARCserve Backup. To run a restore operation, select the media containing the backup you want, select the session containing the specific database or log session you want to restore.

Note: BrightStor ARCserve Backup supports only Restore by Tree and Restore by Session for Microsoft SQL Server.

Restore Databases Using Restore by Tree Method

To restore using the Restore by Tree method, follow these steps:

1. On the Restore Manager Source tab, select Restore by Tree from the drop-down list.
2. In the navigation tree, expand the computer from which the database was backed up to view the database instances. Click the yellow database icon to expand the database instance containing the database you want to restore and click the database name to select it.
3. To use the most recent backup, go to the following step.

   To use a backup other than the most recent one, click Version History. The Version History dialog opens.

   Note: Version History provides information on the media name attached to each backup session, its backup session number, backup method, and the date and time the backup was made. You can select the session to restore from the backup media using Version History.
4. Right-click the database name you selected and choose Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog appears. The dialog varies depending on the restore sessions selected.

   Important! *The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the restore options appropriate for your backup.*
5. Perform one of the following to select restore options:
   - Click OK to allow the Automatic Selection option select the appropriate restore options for the restore job. The Automatic Selection option is the default option for every restore job.
   - Manually select the options you want to use for this restore session and click OK. For more information about options, see the topic Restore Options in the chapter Restoring Microsoft SQL Server Databases.

6. On the Source tab, ensure that the session you want to restore is selected.
7. In the Restore Manager, click the Destination tab and select a destination using one of the following procedures:
   - To restore to the **original** server using the **original** database name, select the Restore files to their original locations option if not selected already.
   - To restore to a **different** server but to an instance of Microsoft SQL Server with the **same version and instance name**, clear the Original Location check box and select the destination machine. The destination machine must have an instance with the same name as the original and must have the same version of Microsoft SQL Server as the source.
   - To restore using a **different** database name, clear the Restore files to their original locations option, select the destination server, and select the Microsoft SQL Server instance on the destination server. Enter a backslash and the new name of the database at the end of the displayed path, as in the following example:

     ```
     \\ABC12-3\dbasql70\Lightning
     ```

     Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 instance, you must explicitly select the target instance even if it is a default instance.
   - To restore to a **different** server or a **different instance** on the original server using the **original** database name, clear the Restore files to their original locations option and select the Microsoft SQL Server instance for the destination server.
8. Click Start. The Session User Name and Password dialog opens.
9. Verify or change the user name or password for the Windows computer on which Microsoft SQL Server is loaded. To verify or change the user name or password, follow these steps:
   a. Select a session on the Machine tab and click Edit. The Enter User Name and Password dialog opens.
   b. Enter or modify the user name and password.
   c. If a session password was assigned to this session, enter the session password.
   d. To apply the user name, password, and session password you entered to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.
   e. Click OK.

10. Verify or change the user name or password for the database servers to which you are restoring. To verify or change the user name or password for the database servers, follow these steps:
    a. Select the DBAgent tab.
    b. Select a session and click Edit. The Enter User Name and Password dialog opens.
    c. Enter or modify the user name and password.
    d. If you want the user name, password, and session password you entered applied to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.
    e. Click OK.
11. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
12. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.
13. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Restore Databases Using Restore by Session Method

To perform a restore operation using the Restore by Session method, follow these steps:

1. On the Restore Manager Source tab, select Restore by Session from the drop-down list. A list of the media you have used when backing up with BrightStor ARCserve Backup opens.
2. To create a filter to view only the sessions from a specific server or the sessions from a specific database on a specific server, perform the following procedure:
    a. Select the Filter tab. The Filter dialog opens.
    b. Click the SQL Agent Filter tab. The SQL Agent Filter dialog opens.
    c. Enter a machine name to restore the sessions from a particular server, or a machine name and database name to restore the sessions of a particular database.

       If you are using Microsoft SQL Server 2000 or Microsoft SQL Server 2005, you can also enter a machine name, instance name, and database name to restore a database from a specific instance of Microsoft SQL Server.
    d. Click OK.

    Note: Once you apply the filter settings, expand the media item to view the results. If the media item was already expanded, collapse it and expand it again to see the results.
3. Choose the media containing the backup you want to restore and select the session containing the specific database or log you want to restore.

    Note: Microsoft SQL Server backups have one database, file, FileGroup, or log backup for each session on the media.
4. Right-click the session containing the database or log you want to restore and choose Backup Agent Options from the pop-up menu.

    The Backup Agent Restore Options dialog appears. This dialog varies depending on the restore sessions selected.
5. Perform one of the following to select restore options:
    - Click OK to allow the Automatic Selection option select the appropriate restore options for the restore job. The Automatic Selection option is the default option for every restore job.
    - Manually select the options you want to use for this restore session and click OK. For more information about options, see the topic Restore Options in the chapter Restoring Microsoft SQL Server Databases.
6. In the Source tab, ensure that the session you want to restore is selected.

7. In the Restore Manager, click the Destination tab and select a destination using one of the following procedures:
   - To restore to the **original** server using the **original** database name, select the Restore files to their original locations option if not selected already.
   - To restore to a **different** server but to an instance of Microsoft SQL Server with the **same version and instance name**, clear the Original Location check box and select the destination machine. The destination machine must have an instance with the same name as the original and must have the same version of Microsoft SQL Server as the source.
   - To restore using a **different** database name, clear the Restore files to their original locations option, select the destination server, and select the Microsoft SQL Server instance on the destination server. Enter a backslash and the new name of the database at the end of the displayed path, as in the following example:

     ```
     \\ABC12-3\dbasql70\Lightning
     ```

     Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 instance, you must explicitly select the target instance even if it is a default instance.
   - To restore to a **different** server or a **different instance** on the original server using the **original** database name, clear the Restore files to their original locations option and select the Microsoft SQL Server instance for the destination server.
8. Click Start. The Session User Name and Password dialog opens.
9. Verify or change the user name or password for the Windows computer on which Microsoft SQL Server is loaded. To verify or change the user name or password, follow these steps:
   a. Select a session on the Machine tab and click Edit. The Enter User Name and Password dialog opens.
   b. Enter or modify the user name and password.
   c. If a session password was assigned to this session, enter the session password.
   d. To apply the user name, password, and session password you entered to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.
   e. Click OK.

10. Verify or change the user name or password for the database servers to which you are restoring. To verify or change the user name or password for the database servers, follow these steps:
    a. Select the DBAgent tab.
    b. Select a session and click Edit. The Enter User Name and Password dialog opens.
    c. Enter or modify the user name and password.
    d. If you want the user name, password, and session password you entered applied to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.
    e. Click OK.
11. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
12. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.
13. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Restore to Alternative Disk Locations Using Automatic Selection

You can restore a database to a different location on a disk (for example, to a different drive letter or directory path, or with a different file name) while using Automatic Selection only if the Backup Agent Restore Options dialog displays file path entries for the data files.

Note: The Backup Agent Restore Options dialog does not have the complete file list for Transaction log and differential backups made using BrightStor ARCserve Backup Version 9.0 or earlier, or BrightStor Enterprise Backup Release 10.0.

To determine whether you can use Automatic Selection to restore a database or session to a different location, follow these steps:

1. Right-click the database, if you are using Restore by Tree, or the last backup session for this database, if you are using Restore by Session. A pop-up menu opens.
2. Choose Backup Agent Options. The Backup Agent Restore Options dialog opens.
3. In the section called Restore database files as, if the FileGroups and data files appear, you can use Automatic Selection. Follow the appropriate procedures in this chapter to restore your data.

Restore to Alternate Disk Locations by Individual Session

If the FileGroups and data files are not displayed in the section called Restore database files as, you must restore the sessions individually to restore them to an alternative disk location. To restore sessions to an alternative disk location individually, use one of the following methods:

- Restore by Session using a single restore job
- Restore by Session using a separate job for each session
- Restore by Tree

For information on these restore methods, see the sections Restore by Session Using a Single Restore Job, Restore by Session Using a Separate Job for Each Session, and Restore Using the Restore by Tree in this chapter.

Restore by Session Using a Single Restore Job

To restore databases by session using a single restore job, follow these steps:

1. On the Restore Manager Source tab, select Restore by Session from the drop-down menu. A list of the media you have used when backing up with BrightStor ARCserve Backup opens.
2. Choose the media containing the backup you want to restore, expand the session containing that backup, and select the current backup session.
3. Right-click the backup session and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens.
4. Clear the Automatic Selection check box and select the Leave database nonoperational, but able to restore additional transaction logs option under Recovery Completion State.

   Note: No additional transaction logs can be restored if this option is not selected.
5. Click OK.
6. For each additional required backup of the database, select the next most recent session, open the Backup Agent Restore Options dialog, clear Automatic Selection, and select the Leave database nonoperational, but able to restore additional transaction logs option under Recovery Completion State. Click OK.
7. For the earliest of these backup sessions (the full backup on which the others depend), make the appropriate changes to the file paths and names.

   Important! *Do not edit the file names or paths for any of the sessions except the full backup session.*
8. Complete the restore job packaging, and submit the restore job. For instructions on restoring by session, see the appropriate section in this guide.

Restore by Session Using a Separate Job for Each Session

If you restore databases using a separate job for each session, you can submit each job on hold and then make each job ready individually as the previous one finishes. To package the database restore job as separate jobs, follow these steps:

1. On the Restore Manager Source tab, select Restore by Session from the drop-down menu. A list of the media you have used when backing up with BrightStor ARCserve Backup opens.
2. Choose the media containing the backup you want to restore, expand the session containing that backup, and select the most recent full database backup of the database you want to restore. This is the full backup on which the more recent backup sessions depend.
3. Right-click the backup session and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens.
4. Clear the Automatic Selection option and edit the file names or paths as appropriate.
5. Select the Leave database nonoperational, but able to restore additional transaction logs option under Recovery Completion State.
6. Click OK to close the Backup Agent Restore Options dialog and submit the restore job.
7. Select the next backup session for the database you want to restore.
8. Right-click the backup session and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens.
9. Clear the Automatic Selection option.
10. If this is **not** the last session to be restored, select the Leave database nonoperational, but able to restore additional transaction logs option under Recovery Completion State.

    If this **is** the last session to be restored, confirm that the Leave database operational. No additional transaction logs can be restored option is selected under Recovery Completion State.
11. Click OK to close the Backup Agent Restore Options dialog and submit the restore job. For instructions about restoring by session, see the appropriate section in this chapter.
12. Repeat the preceding steps from the point at which you close the Backup Agent Restore Options dialog and submit the restore job until all backup sessions have been submitted for the restore.

    Note: You must clear your previous selections before selecting options for the next job.

Restore by Tree

If you are using the Restore by Tree method, you must submit each session as a separate restore job. You may want to submit each job on hold and make each job ready individually as the previous one finishes. To restore sessions as separate jobs using the Restore by Tree method, follow these steps:

1. On the Restore Manager Source tab, select Restore by Tree from the drop-down list.
2. In the navigation tree, expand the computer from which the database you want to restore was backed up. Click the yellow database icon to expand the database instance containing the database you want to restore and select the database.
3. Click Version History. The Version History dialog appears. Scroll to the right to find the columns labeled Method and Backup Time.

   Note: Entries appear in reverse chronological order; more recent backups appear higher on the list.
4. Select the most recent backup with the method Database and click Select.
5. Right-click the selected database session and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens.
6. Edit the file names or paths as appropriate, and select the Leave database nonoperational, but able to restore additional transaction logs option under Recovery Completion State.
7. Click OK to close the Backup Agent Restore Options dialog and submit this restore job. For instructions about restoring by tree, see the section Restore Databases Using Restore by Tree Method.
8. Click Version History again and select the next backup session.
9. Open the Backup Agent Restore Options dialog. Clear the Automatic Selection option.
10. If this is **not** the last session to be restored, select the Leave database nonoperational, but able to restore additional transaction logs option under Recovery Completion State.

    If this **is** the last session to be restored, confirm that the Leave database operational. No additional transaction logs can be restored option is selected under Recovery Completion State.
11. Click OK to close the Backup Agent Restore Options dialog.
12. Submit the restore job. For instructions about restoring by tree, see the section Restore Databases Using Restore by Tree Method.
13. Repeat these steps from the point at which you close the Backup Agent Restore Options dialog and submit the restore job until all backup sessions have been submitted for a restore.

Chapter 5: Backing Up and Restoring in Cluster Environments

This chapter contains information about backing up and restoring database and transaction log files using BrightStor ARCserve Backup, the Agent for Microsoft SQL Server, and Microsoft SQL Server 7.0, Microsoft SQL Server 2000, or Microsoft SQL Server 2005 in a Microsoft SQL Server cluster environment.

Backup and Restore Considerations in Microsoft SQL Server Cluster Environments

BrightStor ARCserve Backup supports the Microsoft SQL virtual server in the Microsoft Clustering Server (MSCS) environment using Microsoft SQL Server 7.0 with Windows NT 4.0 and Microsoft SQL Server 2000 with Windows 2000.

Use BrightStor ARCserve Backup and the Agent for Microsoft SQL Server to back up and restore Microsoft SQL virtual servers just like any nonclustered Microsoft SQL Server, with the following important differences:

- You must install the Agent for Microsoft SQL Server on the local drives of all nodes in the Microsoft SQL virtual server.

  Note: For information about installing the Agent for Microsoft SQL Server, see Install the Agent in a Standard Microsoft SQL Server Environment in the chapter "Installing the Agent."

- If the node on which the Microsoft SQL virtual server is currently running fails over during a backup job, the backup job fails too, and you must restart the job if no makeup job is generated.

Microsoft SQL Virtual Server 7.0 Environment Backups

The following sections provide the procedures to back up in a Microsoft SQL Virtual Server 7.0 environment.

Select Server, Protocol, Security, and Backup Type

To select the server, protocol, security, and backup type when backing up a Microsoft SQL Virtual Server 7.0 environment, perform the following steps:

1. Ensure that the Microsoft SQL virtual server is running in your Microsoft cluster environment on the same node as the Microsoft Cluster Server Quorum Resources.
2. Start BrightStor ARCserve Backup and open the Backup Manager.
3. On the Source tab, either browse the MSCS computer through the Windows Systems or browse through the preferred shares (by adding the MSCS computer to the preferred shares). The Microsoft SQL virtual server object is located under the MSCS cluster on which the server is installed.

   Important! *In the Microsoft SQL Server 7.0 cluster environment, do not go directly to the Microsoft SQL virtual server object in the Microsoft Network.*
4. Right-click the Microsoft SQL virtual server object and select Remote Protocol from the pop-up menu. The Remote Protocol dialog opens.
5. Select Named Pipes and click OK.
6. Right-click the Microsoft SQL virtual server object and select Security. The Security dialog opens.
7. Verify the security information displayed in the Security dialog and click OK.
8. Expand the Microsoft SQL virtual server object to display a list of databases for this server and select a database.

   Note: For information about how to select a database to apply options properly for backup, see Dynamic and Explicit Job Packaging in the chapter "Backing Up Microsoft SQL Databases."
9. Right-click the database and select Backup Agent Options from the pop-up menu. The Agent Backup Options dialog opens.
10. Select the type of backup you want to perform. For more information about backup types, see Types of Backup in the chapter "Backing Up Microsoft SQL Server Databases."
11. If you chose Files/FileGroups, click Browse. The Specify FileGroups and Files dialog opens. Select the files or FileGroups you want to back up and click OK.

12. (Optional) In the Agent Backup Options dialog, enable a Database Consistency Check and select the Database Consistency Check options. For more information about Database Consistency Checks, see the chapter "Backing Up Microsoft SQL Server Databases" and the Microsoft SQL Server documentation.
13. Click OK.
14. Repeat these steps from the point at which you expand the Microsoft SQL virtual server object for each database or database object that you are backing up on this job.

Select Backup Destination, Schedule, and Submit the Job

To select the backup destination, schedule, and to submit the job, perform the following steps:

1. On the Backup Manager Destination tab, select a backup destination.

   Note: You can use the * symbol in the Group or Media fields to create partial wildcards when you select a backup destination. For example, if you have two sets of device groups, one with all members beginning GroupA and the other with all members beginning GroupB, you can select all the GroupA members by entering GroupA* in the Group field. For more information about selecting devices and media, see the *Administrator Guide*.
2. Click the Schedule tab and select the scheduling options for this backup job. For information about scheduling backups, see the *Administrator Guide*.
3. Click Start. The Security and Agent Information dialog opens.

   Note: In this dialog, the column and button labeled Agent refer to the Client Agent for Windows, not the Agent for Microsoft SQL Server. You can edit client agent information at this time. For more information about client agents, see the *Administrator Guide*.
4. In the Security and Agent Information dialog, verify the user name and password for the computer where Microsoft SQL Server is running and for the Microsoft SQL virtual server instance. To enter or change security information for the computer or the Microsoft SQL virtual server instance, select the computer or the Microsoft SQL virtual server instance object, click Security, enter the user name and password, and click OK.

   Note: We recommend that you use the user name and password of a domain administrator for machine authentication. Domain administrators are not dependent on the computer on which the Microsoft SQL Server instance is still running. Specify a domain user with the format DomainName\UserName.
5. Click OK. The Submit Job dialog opens.

6. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.
7. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Restore by Tree in Microsoft SQL Virtual Server 7.0 Environments

To restore using the Restore by Tree method in a Microsoft SQL Virtual Server 7.0 environment, follow these steps:

1. On the Restore Manager Source tab, select Restore by Tree from the drop-down list.
2. In the navigation tree, expand the virtual Windows system from which the database you want to restore was backed up. Click the yellow database icon to expand the database instance and click the database you want to restore to select it.
3. To use the most recent backup, go to the next step.

   To use a backup other than the most recent backup, click Version History. The Version History dialog opens. Select a backup session to use for the restore and click Select.

   Note: Version History provides information on the media name attached to each backup session, its backup session number, backup method, and the date and time the backup was made. You can select the session to restore from the backup media using Version History.
4. Right-click the database you selected and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens. The dialog varies depending on the restore sessions selected.

   Important! *The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the restore options appropriate for your backup.*
5. Perform one of the following to select restore options:
   - Click OK to allow the Automatic Selection option select the appropriate restore options for the restore job. The Automatic Selection option is the default option for every restore job.
   - Manually select the options you want to use for this restore session and click OK. For more information about options, see the topic Restore Options in the chapter Restoring Microsoft SQL Server Databases.
6. On the Source tab, ensure that the session you want to restore is selected.

7. In the Restore Manager, click the Destination tab and select a destination using one of the following procedures:

   - To restore to the **original** server using the **original** database name, select the Restore files to their original locations option if not selected already.
   - To restore to a **different** server but to an instance of Microsoft SQL Server with the **same version and instance name**, clear the Original Location check box and select the destination machine. The destination machine must have an instance with the same name as the original and must have the same version of Microsoft SQL Server as the source.
   - To restore using a **different** database name, clear the Restore files to their original locations option, select the destination server, and select the Microsoft SQL Server instance on the destination server. Enter a backslash and the new name of the database at the end of the displayed path, as in the following example:

     ```
     \\ABC12-3\dbasql70\Lightning
     ```

     Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 instance, you must explicitly select the target instance even if it is a default instance.
   - To restore to a **different** server or a **different instance** on the original server using the **original** database name, clear the Restore files to their original locations option and select the Microsoft SQL Server instance for the destination server.

     Note: If the destination is a Microsoft SQL Server 7.0 Virtual Server, you must select it from the Virtual Windows Server machine.

8. Click Start. The Session User Name and Password dialog opens.

9. Verify or change the user name or password for the Windows computer on which Microsoft SQL Server is loaded. To verify or change the user name or password, follow these steps:

   a. Select a session on the Machine tab and click Edit. The Enter User Name and Password dialog opens.

   b. Enter or modify the user name and password.

   c. If a session password was assigned to this session, enter the session password.

   d. To apply the user name, password, and session password you entered to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.

   e. Click OK.

   Note: We recommend that you use the user name and password of a domain administrator for machine authentication. Domain administrators are not dependent on the computer on which the Microsoft SQL Server instance is still running. Specify a domain user with the format DomainName\UserName.

10. Verify or change the user name or password for the database servers to which you are restoring. To verify or change the user name or password for the database servers, follow these steps:

    a. Select the DBAgent tab.

    b. Select a session and click Edit. The Enter User Name and Password dialog opens.

    c. Enter or modify the user name and password.

    d. If you want the user name, password, and session password you entered applied to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option**.**

    e. Click OK.

    Note: You must use the Microsoft SQL-native user with system administrator privileges. You cannot use a Windows user for Microsoft SQL Server authentication in a Microsoft SQL Server 7.0 cluster.

11. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.

12. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.

13. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Restore by Session in Microsoft SQL Virtual Server 7.0 Environments

To perform a restore job in a Microsoft SQL Virtual Server 7.0 environment using the Restore by Session method, follow these steps:

1. On the Restore Manager Source tab, select Restore by Session from the drop-down menu. A list of the media you have used when backing up with BrightStor ARCserve Backup opens.
2. To create a filter to restore only the sessions from a specific server or the sessions from a specific database on a specific server, perform the following procedure:
   a. Click the Filter tab in the Restore Manager. The Filter dialog opens.
   b. Click the SQL Agent Filter tab. The SQL Agent Filter dialog opens.
   c. Enter a machine name to restore the sessions from a particular server, or a machine name and database name to restore the sessions of a particular database and click OK.

   Note: Once you apply the filter settings, expand the media item to view the results. If the media item was already expanded, collapse it and expand it again to see the results.

   If you do not want to create a filter, go to the next step.
3. Choose the media containing the backup you want to restore, expand the session containing that backup, and select the specific database or log you want to restore.
4. Right-click the name of the database or log you want to restore and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens. The dialog varies depending on the Restore sessions selected.

   Important! *The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the restore options appropriate for your backup.*
5. Perform one of the following to select restore options:
   - Click OK to allow the Automatic Selection option select the appropriate restore options for the restore job. The Automatic Selection option is the default option for every restore job.
   - Manually select the options you want to use for this restore session and click OK. For more information about options, see the topic Restore Options in the chapter Restoring Microsoft SQL Server Databases.
6. In the Source tab, ensure that the session you want to restore is selected.

7. In the Restore Manager, click the Destination tab and select a destination using one of the following procedures:

   - To restore to the **original** server using the **original** database name, select the Restore files to their original locations option if not selected already.
   - To restore to a **different** server but to an instance of Microsoft SQL Server with the **same version and instance name**, clear the Original Location check box and select the destination machine. The destination machine must have an instance with the same name as the original and must have the same version of Microsoft SQL Server as the source.
   - To restore using a **different** database name, clear the Restore files to their original locations option, select the destination server, and select the Microsoft SQL Server instance on the destination server. Enter a backslash and the new name of the database at the end of the displayed path, as in the following example:

     ```
     \\ABC12-3\dbasql70\Lightning
     ```

     Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 instance, you must explicitly select the target instance even if it is a default instance.
   - To restore to a **different** server or a **different instance** on the original server using the **original** database name, clear the Restore files to their original locations option and select the Microsoft SQL Server instance for the destination server.

8. Click Start. The Session User Name and Password dialog opens.
9. Verify or change the user name or password for the Windows computer on which Microsoft SQL Server is loaded. To verify or change the user name or password, follow these steps:

   a. Select a session on the Machine tab and click Edit. The Enter User Name and Password dialog opens.

   b. Enter or modify the user name and password.

   c. If a session password was assigned to this session, enter the session password.

   d. To apply the user name, password, and session password you entered to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.

   e. Click OK.

   Note: We recommend that you use the user name and password of a domain administrator for machine authentication. Domain administrators are not dependent on the computer on which the Microsoft SQL Server instance is still running. Specify a domain user with the format DomainName\UserName.

10. Verify or change the user name or password for the database servers to which you are restoring. To verify or change the user name or password for the database servers, follow these steps:

    a. Select the DBAgent tab.

    b. Select a session and click Edit. The Enter User Name and Password dialog opens.

    c. Enter or modify the user name and password.

    d. If you want the user name, password, and session password you entered applied to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option**.**

    e. Click OK.

    Note: You must use the Microsoft SQL-native user with system administrator privileges. You cannot use a Windows user for Microsoft SQL server authentication in a Windows NT 4.0 Microsoft SQL Server 7.0 cluster.

11. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.

12. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.

13. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Microsoft SQL Virtual Server 2000 Environment Backups

The following sections provide the procedures to back up in a Microsoft SQL Virtual Server 2000 environment.

Select Server, Protocol, Security, and Backup Type

To select a server, protocol, security, and backup type when backing up in a Microsoft SQL Virtual Server 2000 environment, perform the following steps:

1. Verify that Microsoft SQL virtual server is running in your Microsoft cluster environment.

2. Start BrightStor ARCserve Backup and open the Backup Manager.

3. On the Source tab, expand the Microsoft SQL virtual server name, **not** the physical nodes or the Windows virtual server. The Microsoft SQL virtual server instances are located under the Microsoft SQL virtual server object with which they are associated.

   Note: Browse the Microsoft SQL virtual server instances through their associated Microsoft SQL virtual server only. Browsing from any other entry point may cause backup failures if the Windows virtual server or the Microsoft SQL Virtual Server 2000 moves to a different node of the cluster. Do not browse through the MSCS computer for the Microsoft SQL Virtual Server 2000 configuration.

4. Right-click the Microsoft SQL virtual server object and select Remote Protocol from the pop-up menu. The Remote Protocol dialog opens.

5. Select TCP/IP and click OK.

6. Right-click the Microsoft SQL virtual server object and select Security. The Security dialog opens.

7. Verify the security information in the Security dialog and click OK.

8. Expand the Microsoft SQL virtual server object to display a list of databases for this server and select a database or database object.

9. Right-click the database and select Backup Agent Options from the pop-up menu. The Agent Backup Options dialog opens.

10. Select the type of backup you want to perform. For more information about backup types, see the topic Types of Backup in the chapter “Backing Up Microsoft SQL Server Databases.”

11. If you chose Files/FileGroups or Files/FileGroups - Differential, click the Browse Files/FileGroups button. The Specify FileGroups and Files dialog opens. Select the files or FileGroups you want to back up and click OK.

12. (Optional) In the Agent Backup Options dialog, enable a Database Consistency Check, select Database Consistency Check options, and click OK.

    Note: For more information about Database Consistency Checks, see the chapter “Backing Up Microsoft SQL Server Databases” and the Microsoft SQL Server documentation.

13. Repeat these steps for each database or database object you are backing up in this job.

Select Backup Destination, Schedule, and Submit the Job

To select the backup destination, schedule, and to submit the job, perform the following steps:

1. On the Backup Manager Destination tab, select a backup destination.

   Note: You can use the * symbol in the Group or Media fields to create partial wildcards when you select a backup destination. For example, if you have two sets of device groups, one with all members beginning GroupA and the other with all members beginning GroupB, you can select all the GroupA members by entering GroupA* in the Group field. For more information about selecting devices and media, see the *Administrator Guide*.

2. Click the Schedule tab and select the scheduling options for this backup job. For information about scheduling backups, see the *Administrator Guide*.

3. Click Start. The Security and Agent Information dialog opens.

   Note: In this dialog, the column and button labeled Agent refer to the Client Agent for Windows, not the Agent for Microsoft SQL Server. You can edit client agent information at this time. For more information about client agents, see the *Administrator Guide*.

4. In the Security and Agent Information dialog, verify the user name and password for the computer where Microsoft SQL Server is running and for the Microsoft SQL virtual server instance. To enter or change security information for the computer or the Microsoft SQL virtual server instance, select the computer or the Microsoft SQL virtual server instance object, click Security, enter the user name and password, and click OK.

   Note: We recommend that you use the user name and password of a domain administrator for machine authentication. Domain administrators are not dependent on the computer on which the Microsoft SQL Server instance is still running. Specify a domain user with the format DomainName\UserName.

5. Click OK. The Submit Job dialog opens.

6. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.

7. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Restore by Tree in Microsoft SQL Virtual Server 2000 or 2005 Environments

To restore using the Restore by Tree method in a Microsoft SQL Virtual Server 2000 or 2005 environment, follow these steps:

1. On the Restore Manager Source tab, select Restore by Tree from the drop-down list.
2. In the navigation tree, expand the Microsoft SQL virtual server computer name from which the database you want to restore was backed up. Click the yellow database icon to expand the database instance and click the database you want to restore to select it.
3. To use the most recent backup, go to the next step.

   To use a backup other than the most recent backup, click Version History. The Version History dialog opens. Select a backup session to use for the restore and click Select.

   Note: Version History provides information on the media name attached to each backup session, its backup session number, backup method, and the date and time the backup was made. You can select the session to restore from the backup media using Version History.
4. Right-click the database name you selected and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens. The dialog varies depending upon the restore sessions selected.

   Important! *The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the restore options appropriate for your backup.*
5. Perform one of the following to select restore options:
   - Click OK to allow the Automatic Selection option select the appropriate restore options for the restore job. The Automatic Selection option is the default option for every restore job.
   - Manually select the options you want to use for this restore session and click OK. For more information about options, see the topic Restore Options in the chapter Restoring Microsoft SQL Server Databases.
6. On the Source tab, ensure that the session you want to restore is selected.

7. In the Restore Manager, click the Destination tab and select a destination using one of the following procedures:

   - To restore to the **original** server using the **original** database name, select the Restore files to their original locations option if not selected already.
   - To restore to a **different** server but to an instance of Microsoft SQL Server with the **same version and instance name**, clear the Original Location check box and select the destination machine. The destination machine must have an instance with the same name as the original and must have the same version of Microsoft SQL Server as the source.
   - To restore using a **different** database name, clear the Restore files to their original locations option, select the destination server, and select the Microsoft SQL Server instance on the destination server. Enter a backslash and the new name of the database at the end of the displayed path, as in the following example:

     ```
     \\ABC12-3\dbasql70\Lightning
     ```

     Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 instance, you must explicitly select the target instance even if it is a default instance.
   - To restore to a **different** server or a **different instance** on the original server using the **original** database name, clear the Restore files to their original locations option and select the Microsoft SQL Server instance for the destination server.

8. Click Start. The Session User Name and Password dialog opens.
9. Verify or change the user name or password for the Windows computer on which Microsoft SQL Server is loaded. To verify or change the user name or password, follow these steps:

   a. Select a session on the Machine tab and click Edit. The Enter User Name and Password dialog opens.

   b. Enter or modify the user name and password.

   c. If a session password was assigned to this session, enter the session password.

   d. To apply the user name, password, and session password you entered to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.

   e. Click OK.

   Note: We recommend that you use the user name and password of a domain administrator for machine authentication. Domain administrators are not dependent on the computer on which the Microsoft SQL Server instance is still running. Specify a domain user with the format DomainName\UserName.

10. Verify or change the user name or password for the database servers to which you are restoring. To verify or change the user name or password for the database servers, follow these steps:
    a. Select the DBAgent tab.
    b. Select a session and click Edit. The Enter User Name and Password dialog opens.
    c. Enter or modify the user name and password.
    d. If you want the user name, password, and session password you entered applied to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.
    e. Click OK.
11. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
12. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.
13. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Restore by Session in Microsoft SQL Virtual Server 2000 Environments

To perform a restore using the Restore by Session method, follow these steps:

1. On the Restore Manager Source tab, select Restore by Session from the drop-down menu. A list of the media that you have used when backing up with BrightStor ARCserve Backup opens.
2. To create a filter to restore only the sessions from a specific server or the sessions from a specific database on a specific server, perform the following steps:
    a. Click the Filter tab in the Restore Manager. The Filter dialog opens.
    b. Click the SQL Agent Filter tab. The SQL Agent Filter dialog opens.

c. Enter a machine name to restore the sessions from a particular server, a machine name and database name to restore the sessions of a particular database, or a machine name, instance name, and database name to restore a particular instance of a database and click OK.

Note: Once you apply the filter settings, expand the media item to view the results. If the media item was already expanded, collapse it and expand it again to see the results.

If you do not want to create a filter, go to the next step.

3. Choose the media containing the backup you want to restore, expand the session containing that backup, and select the specific database or log you want to restore.
4. Right-click the name of the database or log you want to restore and select Backup Agent Options from the pop-up menu. The Backup Agent Restore Options dialog opens.

   Important! *The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the restore options appropriate for your backup.*

5. Perform one of the following to select restore options:
   - Click OK to allow the Automatic Selection option select the appropriate restore options for the restore job. The Automatic Selection option is the default option for every restore job.
   - Manually select the options you want to use for this restore session and click OK. For more information about options, see the topic Restore Options in the chapter Restoring Microsoft SQL Server Databases.
6. In the Source tab, ensure that the session you want to restore is selected.

7. In the Restore Manager, click the Destination tab and select a destination using one of the following procedures:

   - To restore to the **original** server using the **original** database name, select the Restore files to their original locations option if not selected already.
   - To restore to a **different** server but to an instance of Microsoft SQL Server with the **same version and instance name**, clear the Original Location check box and select the destination machine. The destination machine must have an instance with the same name as the original and must have the same version of Microsoft SQL Server as the source.
   - To restore using a **different** database name, clear the Restore files to their original locations option, select the destination server, and select the Microsoft SQL Server instance on the destination server. Enter a backslash and the new name of the database at the end of the displayed path, as in the following example:

     ```
     \\ABC12-3\dbasql70\Lightning
     ```

     Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 instance, you must explicitly select the target instance even if it is a default instance.
   - To restore to a **different** server or a **different instance** on the original server using the **original** database name, clear the Restore files to their original locations option and select the Microsoft SQL Server instance for the destination server.

8. Click Start. The Session User Name and Password dialog opens.
9. Verify or change the user name or password for the Windows computer on which Microsoft SQL Server is loaded. To verify or change the user name or password, follow these steps:

   a. Select a session on the Machine tab and click Edit. The Enter User Name and Password dialog opens.

   b. Enter or modify the user name and password.

   c. If a session password was assigned to this session, enter the session password.

   d. To apply the user name, password, and session password you entered to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option.

   e. Click OK.

   Note: We recommend that you use the user name and password of a domain administrator for machine authentication. Domain administrators are not dependent on the computer on which the Microsoft SQL Server instance is still running. Specify a domain user with the format DomainName\UserName.

10. Verify or change the user name or password for the database servers to which you are restoring. To verify or change the user name or password for the database servers, follow these steps:
    a. Select the DBAgent tab.
    b. Select a session and click Edit. The Enter User Name and Password dialog opens.
    c. Enter or modify the user name and password.
    d. If you want the user name, password, and session password you entered applied to all the sessions you are restoring, select the Apply [User Name and Password] to all rows option**.**
    e. Click OK.
11. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
12. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.
13. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.

Perform Disaster Recovery in a Microsoft SQL Server 7.0 Cluster Environment

To perform a disaster recovery in a Microsoft SQL Server 7.0 cluster environment, follow these steps:

1. Reinstall Microsoft SQL Server but not the Microsoft SQL virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Set the BrightStor ARCserve Backup SQL Agent services and the SQL Agent services to log in as the domain administrator.
4. Start Microsoft SQL Server in single user mode.
5. Restore the Microsoft SQL Server master database.
6. Set up the Microsoft SQL virtual server and reinstall the Agent for Microsoft SQL Server with the new Microsoft SQL virtual server settings.
7. Restart Microsoft SQL Server in normal, multi-user mode and restore the rest of the databases, starting with msdb.

Note: For more information about recovering from a disaster, see the *Disaster Recovery Option Guide*.

Perform Disaster Recovery in Microsoft SQL Server 2000 Cluster Environments

To perform a disaster recovery in a cluster environment, follow these steps:

1. Reinstall Microsoft SQL Server and reinstall the Microsoft SQL virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Take the Microsoft SQL virtual server off line and start Microsoft SQL Server in single-user mode.
4. Restore the Microsoft SQL Server master database.
5. Restart Microsoft SQL Server in normal, multi-user mode and restore the rest of the databases, beginning with msdb and ending with the replication database, if any.

Note: For more information about recovering from a disaster, see the *Disaster Recovery Guide*.

Chapter 6: Backing Up and Restoring Using Multistriping

This chapter contains information about backing up and restoring a Microsoft SQL database using the multistriping feature. These processes use the BrightStor ARCserve Backup, the BrightStor® ARCserve® Backup Enterprise Module, the Agent for Microsoft SQL Server, and Microsoft SQL Server 7.0, Microsoft SQL Server 2000, or Microsoft SQL Server 2005.

Multistriping with the Agent

The Agent for Microsoft SQL Server, in combination with the multistreaming feature of the Enterprise Module, can use multiple processes and multiple backup devices to accelerate your backup beyond the speed of a single tape drive. With extremely large databases (for example, over 100 GB), this can make the difference between a backup job taking a few hours or an entire day. For more information about multistreaming, see the *Administrator Guide*.

In a single-stripe backup job, a single backup device is used; each database backup is written to one session on one backup media. In a multistriped backup, the multistreaming option is enabled, two or more groups are selected, and the number of backup devices (from 2 to 32) is specified. When multistriping is enabled, several child or subjobs are sent simultaneously to the number of devices specified. If the requested number of devices specified are unavailable, the master job waits until they become available or creates as many child jobs as there are available devices (at least two). You can have as many jobs running simultaneously as you have devices or groups on your system.

How Multistriping Works

To enable multistriping, select the multistreaming option and two or more devices or device groups. Multistriping jobs use the following process:

1. The master job checks available devices and creates multiple child jobs based on available tape drives.
2. The child jobs select devices and media.
3. The Agent for Microsoft SQL Server, Microsoft SQL Server, and ODBC interface communicate. The master job contacts Microsoft SQL Server to begin the backup operation and releases the child jobs to continue.
4. The master job and child jobs communicate. The master job monitors the progress of child jobs and updates the status

5. The child jobs signal the master job when they are complete and the master job reports when the child jobs are complete
6. The master job creates a master session for the backup, containing an index of the tapes and sessions written by the child jobs.

Multistriping Advantages

Using the Agent for Microsoft SQL Server with multistriping can increase the speed of your operations. With multistriping, the combined data transfer speed of all of these devices can serve the target database. This enables the database to be archived up to 32 times faster than using a single drive. Multistriping offers dramatic improvements in throughput for local backup and recovery of very large databases by splitting the data across the allocated devices. With a sufficient number of devices, this can allow extremely large databases, which may have taken more than a day to back up to a single device, to be archived in hours. Multistriping offers robustness and makes highly efficient use of system resources.

System Requirements

To use the Agent for Microsoft SQL Server with multistriping properly, you must ensure that the following requirements are met:

- Install and run the following programs on the same computer with the databases on which you are using this feature:
  - Microsoft Windows 2000 or 2003 with the required service packs. For a list of requirements, see the readme file.
  - Microsoft SQL Server 7.0, 2000, or 2005 with the required service packs. For a list of requirements, see the readme file.
  - BrightStor ARCserve Backup
  - Enterprise Module
  - Agent for Microsoft SQL Server

  Note: We recommend that you install the BrightStor® ARCserve® Backup Tape Library Option.
- Attach multiple backup devices to this machine, or one or more tape changers with multiple drives. You can use up to 32 drives in parallel for a single database.
- We recommend that the machine being used has at least two CPUs.

For a list of machine requirements and required applications for this agent, see the readme file. Check the Computer Associates website for any updates to the requirements.

General Considerations and Operating Requirements

The following list provides general considerations and operating requirements for the multistriping feature:

- This feature supports local backup and restore only; remote backup and restore operations are not supported.
- Tape Copy by Session is not supported for all the sessions associated with multistriping backup. Whole media cloning is the only tape copy method supported.
- You cannot use multistriping on the master, msdb, and model SQL Server databases. A master database requires that only a single connection be made to Microsoft SQL Server while multistriping requires multiple connections. Other system databases, such as model or msdb, are too small to gain significant benefits from multistriping.
- You must select a multistriping Microsoft SQL master session to restore its child sessions.
- All related backup media must be present and listed in the BrightStor ARCserve Backup database at the time of the restore operation.
- The same number of backup devices that were used to backup the database must be available at the time of the restore operation.

Back Up Local Microsoft SQL Databases Using Multistriping

This procedure provides the basic steps required to back up a local database using Multistriping.

Important! *You must have properly configured your device groups and a media pool for a multistreamed backup operation before performing a multistriped local backup. For information on how to configure device groups and a media pool, see the online help. For information on tuning your parameters, see Configure Multistriping Parameters in the chapter "Installing the Agent."*

1. On the Backup Manager Source tab, expand the local server.
2. Right-click the Microsoft SQL Server on which you want to perform a multistriped backup, and select Security.
3. On the Security dialog, enter the user name and password for a user who has the system administrator role for the Microsoft SQL Server, and click OK. Expand the local Microsoft SQL Server.
4. Select the first database on which you want to perform a multistriped backup by clicking the green box to the left of the database name. The box turns full green.

5. Right-click the database name and select Agent Options from the pop-up menu. The Agent Backup Options dialog appears. This dialog varies depending upon the version of Microsoft SQL Server you are using.

6. In the Number of Backup Devices to use field, enter the number of backup media devices (between 2 and 32) you want to use to back up this database.

   Note: If you enter a 1 in this field, multistriping is deactivated for this database and the database is backed up to a single device.

7. Select other backup options, as applicable. Click OK.

   Note: For information on database consistency checks, see Database Consistency Checks in the chapter "Backing Up Microsoft SQL Server Databases."

8. Repeat the preceding steps for each database on which you want to use multistriping.

9. On the Destination tab, select the Multistream and Use Any Group options or enter a partial group name using a wildcard that matches the appropriate number of device groups. You can select a single group if the group is a multidrive changer configured with the Tape Library Option.

   Note: To use multistriping, you must select the Multistream option, and select two or more groups or a single group containing two or more backup devices. You must also have selected more than one backup device on the Microsoft SQL Server 2000 Agent Backup Options dialog. If you select only one device, this feature is deactivated and your backup may be unsuccessful. For information on multistreaming and multiple backup devices, see the *Administrator Guide*.

10. Choose a media pool from the drop-down list or enter a media pool in the Media Pool field.

11. On the Schedule tab, set the appropriate scheduling options, including rotation settings, if applicable. For information on rotation schemes, see the *Administrator Guide.*

    Note: Selecting the differential or incremental backup methods override options selected for individual databases. If you select the incremental backup method, the agent performs a Transaction log backup for databases using the full or bulk-logged recovery models, and a differential backup for databases using the Simple Recovery model. For more information about recovery models, see the *Microsoft SQL Server Books Online*.

12. Click Start. The Security and Agent Information dialog opens.

13. In the Security and Agent Information dialog, verify that the authentication information shown is correct. To change a user name or password, highlight the entry, click Security, and make the changes and click OK. The Submit Job dialog opens.

    Note: The job runs successfully if the machine authentication is omitted for the local server. The BrightStor System Account is used as machine authentication for local backup operations.

14. On the Submit Job dialog, enter a description, if necessary, and specify when to run the job:

    - To run the job immediately, select Run Now and click OK.

      Note: The Run Now option is unavailable when more than one device group is selected.

    - To run a job on a specific date and time, select Run On, enter a date from the Date drop-down list, and enter hour, minutes, and seconds in the Time field. Click OK.

    - To run the job at a later time without specifying a time or date, select Run On and then select Submit on Hold. Click OK.

15. To monitor the job, open the Job Status Manager from the Quick Start bar or the Quick Start menu.

16. The job you submitted from the Backup Manager serves as a master job for this backup. When the job first begins, it starts several child or sub-jobs, one for each backup media device of a Multistriped database backup, and one for each multistreaming item included in the job.

    Note: If the number of backup devices available is too few to support simultaneous Multistriped backups from the same master job, the master job performs these backups sequentially.

17. Each child job begins and obtains a backup device and media. After all child jobs have started for a particular database, the agent instructs Microsoft SQL Server to begin transferring data.

18. When each child job completes, it reports its completion status to the master job.

19. When all child jobs are complete for the backup job, the master job creates a master session on backup media for each multistriped database backup and reports its own completion status.

Restore Local Microsoft SQL Databases Using Multistriping

This procedure provides the basic steps required to restore a local database using Multistriping.

Important! *You must have properly configured your device groups for a Multistriped operation before performing a multistriped local restore operation. For information on how to configure device groups, see the online help. For information on tuning your parameters, see the topic Configure Multistriping Parameters in the chapter "Installing the Agent." The media containing the master session and all child sessions must be available and listed in the BrightStor ARCserve database. If not, you must merge these media. For more information on merging tapes, see the Administrator Guide.*

1. From the Restore Manager Source tab, select one of the following options from the drop-down list:
   - **Restore by Tree:** Builds a directory tree of all machines, directories, and files that were backed up.
   - **Restore by Session:** Displays all media used for backups and the source for each session on the media.
2. Expand the media containing the master session for the multistriped backup you want to restore. Select the master session for that backup. Master sessions, like standard database backup sessions, are preceded by a green box; child sessions are preceded by a gray box. You cannot select child sessions directly.

   Note: If you select a differential or Transaction log backup, the Automatic Session Selection feature automatically identifies any standard database backups and the master sessions for multistriped backups on which the selected session depends. Child sessions are not selected through the Automatic Session selection feature. These sessions are identified during the restore process.

   If your backup history for a database includes both Multistriped backups and regular database backups, they can be restored together safely using Automatic Selection.
3. (Optional) You can select one or both of the following options:
   - To restore options other than the defaults, right-click the selected session and select Backup Agent Options from the pop-up menu.
   - To restore to an alternate SQL Server instance on the local server, or to give the database a different name as part of the restore process, use the Destination tab.

   Note: For information on restoring databases, see the chapter "Restoring Microsoft SQL Server Databases."
4. Click Start.

5. The Session User Name and Password dialog opens:
6. For each session, set the appropriate authentication information. To do this, double-click the first session. The User Name and Password dialog opens.
7. Enter the user name and password for Windows authentication for the server. If you used a Session Password as part of your backup options, enter this password in the appropriate field. To apply the same Windows user name and password to the entire session, select the Apply [User Name and Password] to All Rows option. Click OK.
8. Click the DBAgent folder tab. Repeat the preceding steps, but enter the user name and password for authentication to the Microsoft SQL server instance.

   Note: You do not need to reenter session password information here.
9. Click OK. The Submit Job dialog opens.
10. (Optional) Use the Submit Job dialog to select the job execution time, submit the job on hold, add an optional description for the backup job, or select source priority.
11. Click OK to submit this job. If you selected Run Now, the Job Status window opens. Use this window to monitor your job. For more information about the Job Status window, see the *Administrator Guide*.
12. As with a backup job, the job you submit from the Restore Manager serves as a master job; however, restore jobs do not use the multistriping feature in the same way as backup jobs. Operations are performed sequentially, with the master job starting child jobs for one restore operation at a time.
13. For each backup being restored, the master job determines if the backup to be restored is a multistriped backup or a regular database backup.
    - For a regular database backup, the backup session is restored normally.
    - For a multistriped backup, the master job again starts several child jobs, one for each child job started for the backup.
14. Each child job begins and obtains one of the media to which the backup was recorded.
15. When all child jobs have begun, the agent instructs Microsoft SQL Server to prepare to restore and starts transferring data.
16. When each child job completes, it reports its completion status to the master job.
17. After all child jobs for a restore operation are completed, the master job proceeds to the next restore, if any, until all selected sessions have been restored, creating additional child jobs as appropriate.
18. When all restore jobs are completed, the master job reports its own completion status to the Job Status Manager.

Appendix A: Troubleshooting and Disaster Recovery

This appendix explains the most common messages for BrightStor ARCserve Backup and the Agent for Microsoft SQL Server and provides general troubleshooting information and important information about disaster recovery.

Agent and BrightStor ARCserve Backup Error Messages

This section explains the most common error messages for BrightStor ARCserve Backup and the Agent for Microsoft SQL Server.

General Considerations for BrightStor ARCserve Backup and the Agent

The following general considerations apply to BrightStor ARCserve Backup and the Agent for Microsoft SQL Server:

- BrightStor ARCserve Backup does not support special characters (for example, /, *, <>, or ?) in file names, FileGroup names, and database names.
- When restoring a file or FileGroup, if there are no log sessions to follow the file or FileGroup session, BrightStor ARCserve Backup cannot verify if the file was modified. As a result, it cannot identify the final recovery completion state. By default, it chooses the option Leave database non-operational, but able to restore additional transaction logs. Each time you back up a file or FileGroup, ensure that you back up a log immediately afterwards.

Backup or Restore Operation Failed

Backup or restore failed.

Reason:

There are a number of reasons for backup or restore failure.

Action:

To resolve this condition, perform the following actions:

- Determine if the backup or restore failed because an incompatible database option was set. For more information, see the section Valid Operations with SQL Server Database Options.
- Check the Agent for Microsoft SQL Server log file dbasql.log for specific errors. This log is located in the Backup Agent directory.
- See the Microsoft SQL Server manual for information on backup and restore operations.

E8737

Database <dbname> will be skipped. Multistriping requires the use of the Multistream option.

Reason:

You have chosen more than one device in the Agent Backup Options dialog but have not selected the Multistream option on the Destination tab.

Action:

To enable multistriping, ensure that the Multistream option on the Destination tab is selected.

W2247

Multistream backup requires two or more backup devices. Only one is selected. Backup will occur without Multistreaming.

Reason:

You have selected the Multistream option on the Destination tab, but have selected only one backup device in the Agent Backup Options dialog or the Use Any Group option on the Destination tab. This feature is not enabled unless you select two or more devices.

Action:

To use multistreaming, select two or more backup devices on the Source tab and enable the Multistream option on the Destination tab.

From the Source tab, right-click the database you want to back up or restore and select Agent Options. In the Agent Backup Options dialog, ensure that you have selected at least two, but not more than 32, backup devices. On the Destination tab, ensure that you have selected the Use Any Groups option or have selected at least two device groups.

No Icon in Browser

No Microsoft SQL Server icon in BrightStor ARCserve Backup browser.

Reason:

This message can be generated if the Agent for Microsoft SQL Server is not installed or the CA BrightStor Backup Agent RPC Server service is not running or not functioning, or if no agent entry exists in either of the following two places in the registry:

```
SOFTWARE\ComputerAssociates\BrightStor ARCserve Backup\DSAgent\
CurrentVersion\agent\dbasql70
```

or

```
SOFTWARE\ComputerAssociates\BrightStor ARCserve Backup\DSAgent\
CurrentVersion\agent\dbasql@INSTANCENAME
```

Action:

To address this error, verify that the Agent for Microsoft SQL Server is installed. Restart the CA BrightStor Backup Agent RPC Server service. Check the registry for the agent entry and reinstall the Agent for Microsoft SQL Server, if necessary.

Backup Agent Error-(82)

The Backup Agent appears to hang when double clicking on it.

Reason:

Microsoft SQL Server is not running.

Action:

Start Microsoft SQL Server.

Backup Agent Error-(83)

Remote pipe open failed: time-out.

Reason:

The CA BrightStor Backup Agent RPC Server service on the remote machine is not responding.

Action:

Restart the CA BrightStor Backup Agent RPC Server service on the remote computer. If it continues to fail, reboot the remote computer.

Backup Agent Error (85)

Invalid User ID or password was submitted.

Reason:

An incorrect password has been submitted.

Action:

Enter the correct password for the database server.

E8542

Failed to receive data from the SQL server database agent.

Reason:

The network connection was lost or a response was not received from the agent within the specified timeout period.

Action

Check the network connections and verify that the Agent for Microsoft SQL Server services are running. Increase the TCPTimeout value in the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\BrightStor ARCserve Backup\Base\Task\Common\DBAgentsSettings.
```

This value is measured in seconds. The default value is 1200 (20 minutes).

E8562

Failed to receive data from the database agent.

Reason:

The network connection was lost or a response was not received from the agent within the specified timeout period.

Action:

Check the network connections and verify that the Agent for Microsoft SQL Server services are running. Increase the TCPTimeout value in the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\BrightStor ARCserve Backup\Base\Task\Common\DBAgentsSettings.
```

This value is measured in seconds. The default value is 1200 (20 minutes).

E8601

Failed to connect agent. Backup Agent Error - (73)

Reason:

BrightStor ARCserve Backup may have failed to connect to the Agent for Microsoft SQL Server for any of the following reasons:

- The CA BrightStor Backup Agent RPC Server service or the CA BrightStor Backup Agent Remote Service are not running on the target server.
- The target database instance may be stopped or inaccessible or the target server may be unreachable over the network.
- Machine authentication failure may have occurred.
- The Agent for Microsoft SQL Server DLL failed to load into memory on the target server.

Action:

To address this error, perform the following actions:

1. Restart the CA BrightStor Backup Agent RPC Server service or the CA BrightStor Backup Agent Remote Service on the target server.
2. Verify that the target database server instance is running on the target server. If it is not, restart it. Check the network connections.
3. Check the user name and password for the target server logon.
4. Specify domain or machine authentication for the target server logon.
5. Verify that the target server has sufficient free memory available.
6. Verify that the Agent for Microsoft SQL Server DLL exists on the target server.
7. Verify that the DLL path is correct in the target server registry.

E8602 (73)

Failed to read from database. Backup Agent Error - (73)

Reason:

This error can occur for the following reasons:

- trunc.log on chkpt is enabled and is attempting to back up a log
- select into/bulkcopy is set and is attempting to back up a log

Action:

Disable these options in the database.

E8604 (73)

Failed to start backup - Backup Agent Error - (73)

Reason:

A database has been marked as read-only and an attempt to back up the transaction log was made.

Action:

Re-enable write access to the database.

E8604 (406)

Failed to start backup - Backup Agent Error - (406) VDI message: The API was waiting and the timeout interval had elapsed.

Reason:

A backup failed because Microsoft SQL Server did not respond to the Agent for Microsoft SQL Server within the time-out period.

Action:

In the Microsoft SQL Server DBAgent Configuration dialog, increase the VDI wait time (in milliseconds) value. The default value is 120000 milliseconds (2 minutes).

E8603

Restore of the master database failed.

Reason:

The restore of the master database failed; Microsoft SQL Server was not started in single-user mode.

Action:

Start Microsoft SQL server in single-user mode. For more information, see Restore Master Databases in Windows 2000, XP, or 2003 or Restore Master Databases in Windows NT in the chapter “Restoring Microsoft SQL Server Databases.”

E8602 (405-409 or 512-520)

Failed to read from database. Backup Agent Error - (405-409 or 512-520)

Reason:

An internal error has occurred on a Virtual Device during a remote backup using TCP.

Action:

Perform the following actions:

1. Restart the CA BrightStor Backup Agent RPC Server service.
2. Restart the CA BrightStor Backup Agent Remote Service.
3. Restart Microsoft SQL Server.

If this error persists, contact Customer Support.

Agent Messages

The following section provides information regarding messages received from the Agent for Microsoft SQL Server.

143

The current log session contains the time required by the user. The database was fully recovered, so this session is the last log session restored for the current job.

144

The current log session is skipped because of a STOPAT restore option found in a previous session.

146

The required time was not found in this log session or the database is in an unrecovered state.

Microsoft SQL Server Error Messages

Microsoft SQL Server errors can occur for a variety of reasons and can appear in several different BrightStor ARCserve Backup error messages. If a Microsoft SQL Server error occurs, find the value shown for NativeError.

Limitations of Microsoft SQL Server Database

Microsoft SQL Server databases have the following limitations:

- If you have named a Microsoft SQL database with a special character (for example, /, *, <>, or ?), by default, Microsoft SQL Server names the file, FileGroup, or database with the same characters. To perform a backup or restore job, rename the file, FileGroup, or database so it does not include one of these special characters.
- If you use the Log point in time restore option for Microsoft SQL Server 7.0, or the Stop at time, Stop at log mark, or Stop before log mark options for Microsoft SQL Server 2000, and the specified time stamp or mark is not found by the Microsoft SQL Server, the database remains in a loading state and the job result is incomplete.

3023

Backup and file manipulation operations on a database must be serialized.

Reason:

An attempt was made to back up or restore a database while another backup or restore operation was in progress on that database.

Action:

Close all programs that might be accessing the database, including the SQL Server Enterprise Manager. Wait for other operations on the database to finish and retry the operation.

3101

Exclusive access could not be obtained because the database is in use.

Reason:

An attempt was made to restore a database while another program was accessing that database.

Action:

Close all programs that might be accessing the database, including the SQL Server Enterprise Manager, and retry the operation.

3108

"RESTORE DATABASE" must be used in single user mode when trying to restore the master database.

Reason:

An attempt was made to restore the master database without starting the database server in single-user mode.

Action:

For instructions on starting the database server in single-user mode, see Restore Master Databases in Windows 2000, XP, or 2003 or Restore Master Databases in Windows NT in the chapter "Restoring Microsoft SQL Server Databases."

4305 or 4326

The log in this backup set terminates at ..., which is too early to apply to the database.

Reason:

An attempt was made to restore transaction logs out of order, with an older log being restored after a newer one or an attempt was made to restore a transaction log after a more recent backup.

Action:

Restore the last full or differential database backup again and reapply the transaction logs in the order in which they were backed up.

For additional information, see the *Microsoft SQL Server Books Online*.

Valid Operations with SQL Server Database Options

The following list identifies each database option and indicates whether a backup or restore operation can be performed if that database option is set:

Backup Option: Read only

- Database Backup: Yes
- Log Backup: Yes (Microsoft SQL Server 7.0 only)
- Database Restore: Yes
- Log Restore: Yes

Backup Option: Dbo use only

- Database Backup: Yes
- Log Backup: Yes
- Database Restore: Yes
- Log Restore: Yes

Backup Option: No chkpt on recovery

- Database Backup: Yes
- Log Backup: Yes
- Database Restore: Yes
- Log Restore: Yes

Backup Option: Do not recover

- Database Backup: No
- Log Backup: No
- Database Restore: No
- Log Restore: No

Backup Option: Not recovered

- Database Backup: No
- Log Backup: No
- Database Restore: No
- Log Restore: No

Backup Option: trunc.log on chkpt.

- Database Backup: Yes
- Log Backup: No
- Database Restore: N/A
- Log Restore: N/A

Backup Option: single user

- Database Backup: Yes, if the single connection to the database is not in use
- Log Backup: Yes, if the single connection to the database is not in use
- Database Restore: Yes
- Log Restore: Yes

Backup Option: Select into/bulkcopy

- Database Backup: Yes
- Log Backup: No
- Database Restore: N/A
- Log Restore: N/A

Replication of Microsoft SQL Server

According to Microsoft, the replication capability of Microsoft SQL Server is not specifically designed to accomplish hot backups. See the *Microsoft SQL Server Database Administrator Guide* for more information about how to back up and restore in a replication scenario.

MS SQL Server Disaster Recovery

The Agent for Microsoft SQL Server uses the Microsoft SQL Server Backup and Restore database functions, but these functions do not actually back up the physical files that make up the database. Consequently, for a restore operation to succeed, the database must exist; the Load command restores the data to it.

The Master Database

For Microsoft SQL Server to run, the master database must be set up as follows:

- A master database and a model database **must** exist.
- To have a master and a model database, you must either reinstall Microsoft SQL Server, rebuild the master database using Microsoft SQL Server setup, or restore an offline copy of the master database from media.
- After the master database exists, Microsoft SQL Server must be running to execute the Restore command.

To restore the master database, Microsoft SQL server must be running in single-user mode. For more information about restoring the master database, see Restore Master Databases in Windows 2000, XP, or 2003 or Restore Master Databases in Windows NT in the chapter, “Restoring Microsoft SQL Server Databases.”

Potential Restore Problems

The master database tracks all of the resources allocated to Microsoft SQL Server. If you do not perform an offline backup after you have made a major change in the Microsoft SQL Server configuration, restore problems can occur.

For example, for a Microsoft SQL Server configuration with five databases in addition to the master database, you back up the master database, drop one database (detach it from Microsoft SQL Server) and delete the files that make it up. If you do not perform an offline backup and you restore the master database backup at this point, it contains information for the dropped database. As a result, Microsoft SQL Server marks the database as suspect (inaccessible by users). You must drop the database again.

To avoid such problems, perform at least one offline backup. In addition, each time you make a major change in the Microsoft SQL Server configuration (create or drop a database or add a device), you should perform an offline backup.

Suggested Database Restore Sequence

We recommend that you restore the databases in the following order to avoid conflicts:

1. Restore the master database in single-user mode.
2. Restart the Microsoft SQL Server in multi-user mode and restore msdb immediately after you restore the master database.
3. Restore all other databases in normal, multi-user mode.

Note: These suggestions are not requirements, but following them speeds and simplifies the disaster recovery procedure. If you restore other databases in addition to the master database before restoring msdb, Microsoft SQL server loses part of the backup and restore history for the other databases when msdb is restored.

For more information, see the Microsoft SQL Server documentation.

Disaster Recovery Scenario in a Standard MS SQL Server Environment

A typical disaster recovery scenario consists of the following steps:

1. Reinstall Windows, if necessary.
2. Reinstall BrightStor ARCserve Backup, if necessary.

3. Perform one of the following steps as appropriate:
   - If an offline backup exists, restore it.
   - If an offline backup does not exist and you have the Microsoft SQL rebuildm.exe utility, use the utility to recreate the master and model database. For more information, see the Microsoft documentation.
   - If an offline backup does not exist and you do not have the Microsoft SQL rebuildm.exe utility, reinstall the Microsoft SQL Server or MSDE-based application.
4. Reinstall the Agent for Microsoft SQL Server, if necessary.
5. Start Microsoft SQL Server in single-user mode and restore the master database. After the restoration is complete, Microsoft SQL Server automatically stops.
6. Restart Microsoft SQL Server in normal, multi-user mode.
7. Restore the MSDB database.
8. Restore all other databases and transaction logs, except the replication database.
9. If replication is being used, restore the replication database.

Perform Disaster Recovery in a Microsoft SQL Server 7.0 Cluster Environment

To perform a disaster recovery in a Microsoft SQL Server 7.0 cluster environment, follow these steps:

1. Reinstall Microsoft SQL Server but not the Microsoft SQL virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Set the BrightStor ARCserve Backup SQL Agent services and the SQL Agent services to log in as the domain administrator.
4. Start Microsoft SQL Server in single user mode.
5. Restore the Microsoft SQL Server master database.
6. Set up the Microsoft SQL virtual server and reinstall the Agent for Microsoft SQL Server with the new Microsoft SQL virtual server settings.
7. Restart Microsoft SQL Server in normal, multi-user mode and restore the rest of the databases, starting with msdb.

Note: For more information about recovering from a disaster, see the *Disaster Recovery Option Guide*.

Perform Disaster Recovery in Microsoft SQL Server 2000 Cluster Environments

To perform a disaster recovery in a cluster environment, follow these steps:

1. Reinstall Microsoft SQL Server and reinstall the Microsoft SQL virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Take the Microsoft SQL virtual server off line and start Microsoft SQL Server in single-user mode.
4. Restore the Microsoft SQL Server master database.
5. Restart Microsoft SQL Server in normal, multi-user mode and restore the rest of the databases, beginning with msdb and ending with the replication database, if any.

Note: For more information about recovering from a disaster, see the *Disaster Recovery Guide*.

Appendix B: Configuring Microsoft SQL Server Security Settings

This appendix explains how to configure Microsoft SQL Server security settings for BrightStor ARCserve Backup.

Types of Microsoft SQL Authentication

Microsoft SQL Server provides two types of user authentication:

- Apply the Windows login identification
- Use separate user credentials specific to Microsoft SQL Server

Although Microsoft recommends using only Windows authentication wherever possible, in certain instances it is appropriate, or even necessary, to enable Microsoft SQL Server-based authentication. For example, you must use Microsoft SQL Server authentication for Microsoft SQL Server 7.0 or Microsoft SQL Server 2000 when the database is running in a cluster.

Authentication Requirements

For Microsoft SQL Server authentication, you must specify a user account with system administrator access privileges. By default, Microsoft SQL Server creates an account with this level of access, called **sa**, but the Agent for Microsoft SQL Server can use any account with equivalent privileges.

For Windows authentication, any account with administrator equivalency for the machine on which the database is running typically has system administrator access privileges for the database.

Note: A Windows or domain administrator does not automatically have system administrator access privileges for the database if the BUILTIN\Administrators login entry in Microsoft SQL Server has been removed or does not have this role, or if a separate login entry for this user in Microsoft SQL Server does not have this role.

How User Authentication Is Changed

Regardless of the authentication option you choose, you must configure both Windows and BrightStor ARCserve Backup. In addition, if you change the option, you must update settings so that both Windows and BrightStor ARCserve Backup reflect the change. If you are using Microsoft SQL Server 2000, you must make the change separately for each instance of the server.

To change user authentication and update the settings for both Windows and BrightStor ARCserve Backup so that this change is reflected, perform the following process:

1. Check and change the Microsoft SQL Server authentication method.
2. Update the Agent for Microsoft SQL Server Account Configuration.
3. Check and change the ODBC (open database connectivity) settings.
4. Update the Backup Manager in BrightStor ARCserve Backup.

More information regarding the steps in this process are contained in the following sections.

Check or Change the Microsoft SQL Server Authentication Method

To check or change the Microsoft SQL Server authentication method, follow these steps:

1. On the system where Microsoft SQL Server is running, open the SQL Server Enterprise Manager.
2. In the Tree pane, expand the entries below Console Root until you reach the instance of the database server.
3. Right-click the instance and select Properties from the drop-down menu. The Properties dialog opens.
4. In the Properties dialog, choose the Security tab.
5. Under Authentication, select either Microsoft SQL Server and Windows to enable Microsoft SQL server-based authentication or Windows Only to enable only Windows user names and passwords.
6. Click OK.

Update the Agent Account Configuration

To update the Agent for Microsoft SQL Server account configuration, follow these steps:

1. From the Start menu, start the Microsoft SQL Account Configuration utility. The Account Configuration dialog opens.
2. Locate the Microsoft SQL Server instance you modified in the first part of the process. For more information, see "Check or Change the Microsoft SQL Server Authentication Method."
3. Use the following criteria to select either Microsoft SQL authentication or Windows NT authentication:
   - If you have set Microsoft SQL Server for Windows Only, select Windows NT authentication.
   - If you have selected Mixed Mode and want to be able to back up Microsoft SQL using only Microsoft SQL native users, select Microsoft SQL authentication.
   - If you have selected Mixed Mode and want to be able to back up Microsoft SQL using either Windows or Microsoft SQL users, select Windows NT authentication.
4. If you have selected Microsoft SQL authentication, enter the user name and password for a Microsoft SQL native user with system administrator privileges.
5. If Microsoft SQL Server is running in a cluster environment, verify that the cluster-related information is correct.
6. Click Finish to apply the changes.

Check and Change the ODBC Settings

To check and change the ODBC settings, follow these steps:

1. On the system where Microsoft SQL Server is running, from the Start Menu, select Control Panel (if necessary) and select Administrative Tools.
2. Select Data Sources (ODBC). The ODBC Data Source Administrator dialog opens.
3. On the System DSN folder tab, select the Microsoft SQL Server instance or instances. For Microsoft SQL Server 7.0, the instance is called DBASQL7. For Microsoft SQL Server 2000 and 2005, the default instance is called dbasql_MSSQLSERVER. Other instances use the format dbasql_ followed by the instance name.

4. Click Configure. The Microsoft SQL Server DSN Configuration dialog opens. The name field displays the same name as the entry you selected. The Description field displays SQL Backup Agent. The Server field displays the computer on which Microsoft SQL Server is running.

5. Click Next **without changing any settings**. You are prompted to indicate whether Microsoft SQL Server should use Windows authentication or Microsoft SQL Server authentication to verify the authenticity of the login ID:

   - If you are using Windows Only for Microsoft SQL Server, ensure that Windows authentication is selected.
   - If you are using Mixed Mode for Microsoft SQL Server, select either Windows authentication to enable both Windows and Microsoft SQL Server-native user logins or Microsoft SQL Server authentication to enable only Microsoft SQL Server-native user logins.
   - If you are using Microsoft SQL Server authentication, ensure that the specified login ID has system administrator access. If you are uncertain, use **sa**, and consult the Microsoft SQL Server documentation for instructions on setting the password for this account in Microsoft SQL Server. Reenter the password to ensure it is correct.

6. Click Next. If you have specified Microsoft SQL Server authentication, and either the login ID or password is incorrect, an error message appears. Click OK to close the error message, reenter the login ID and password, and click Next.

7. Click Next and click Finish. The ODBC Microsoft SQL Server Setup dialog opens, summarizing the settings. Click Test Data Source.

   The SQL Server ODBC Data Source Test dialog opens. After a few seconds, the dialog displays the results of a quick connection test:

   - If the Microsoft SQL Server ODBC Data Source Test dialog reports that the tests finished successfully, click OK and click OK on the ODBC Microsoft SQL Server Setup dialog. You are returned to the ODBC Data Source Administrator dialog. Click OK.
   - If the SQL Server ODBC Data Source Test dialog reports any errors or failures, this is due to an error in ODBC or the Microsoft SQL Server. See the Microsoft SQL Server documentation for troubleshooting instructions.

Update the Backup Manager

To update the Backup Manager and backup jobs, follow these steps.

1. On the system where you have installed BrightStor ARCserve Backup, start BrightStor ARCserve Backup and open the Backup Manager.
2. Select the Source tab.
3. In the left pane navigation tree, expand the server on which Microsoft SQL Server is running to show the database. If the server does not appear in the tree, see the *Getting Started* guide for instructions on how to add it.
4. Right-click the database and choose Security from the pop-up menu. The Security dialog opens.
5. If you are using Windows security, specify the user name for the account you are using to access the server and either enter the password or leave the field blank.

   If you are using Microsoft SQL Server security, enter the user name and password for the Microsoft SQL Server account and click OK.
6. If you have any recurring backup jobs already scheduled for this Microsoft SQL Server, select the Job Queue folder tab in the Job Status Manager.
7. Select an applicable backup job and click Modify on the toolbar. The Backup Manager opens.
8. Expand the server, right-click this job, repeat the Security dialog procedures, and click Start to resubmit the backup job.
9. Repeat these steps for each applicable backup job.

Note: If you have any restore jobs scheduled for this Microsoft SQL Server, you must delete and recreate them.

Index

D

E

F

H

I

J

K

L

M

O

P

R

S

T

U

V