

CA ARCserve® Backup for Windows

Agent for Microsoft SQL Server Guide

r12



This documentation and any related computer software help programs (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of the Documentation for their own internal use, and may make one copy of the related software as reasonably required for back-up and disaster recovery purposes, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the provisions of the license for the Product are permitted to have access to such copies.

The right to print copies of the Documentation and to make a copy of the related software is limited to the period during which the applicable license for the Product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

EXCEPT AS OTHERWISE STATED IN THE APPLICABLE LICENSE AGREEMENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in the Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Copyright © 2008 CA. All rights reserved.

CA Product References

This documentation set references the following CA products:

- Advantage™ Ingres®
- BrightStor® ARCserve® Backup for Laptops and Desktops
- BrightStor® CA-1® Tape Management
- BrightStor® CA-Dynam®/B Backup for VM
- BrightStor® CA-Dynam®/TLMS Tape Management
- BrightStor® CA-Vtape™ Virtual Tape System
- BrightStor® Enterprise Backup
- BrightStor® High Availability
- BrightStor® Storage Resource Manager
- BrightStor® VM:Tape®
- CA ARCserve® Backup Agent for Novell Open Enterprise Server for Linux
- CA ARCserve® Backup Agent for Open Files on NetWare
- CA ARCserve® Backup Agent for Open Files on Windows
- CA ARCserve® Backup Client Agent for FreeBSD
- CA ARCserve® Backup Client Agent for Linux
- CA ARCserve® Backup Client Agent for Mainframe Linux
- CA ARCserve® Backup Client Agent for NetWare
- CA ARCserve® Backup Client Agent for UNIX
- CA ARCserve® Backup Client Agent for Windows
- CA ARCserve® Backup Enterprise Option for AS/400
- CA ARCserve® Backup Enterprise Option for Open VMS
- CA ARCserve® Backup for Windows
- CA ARCserve® Backup for Windows Agent for IBM Informix
- CA ARCserve® Backup for Windows Agent for Lotus Domino
- CA ARCserve® Backup for Windows Agent for Microsoft Data Protection Manager
- CA ARCserve® Backup for Windows Agent for Microsoft Exchange
- CA ARCserve® Backup for Windows Agent for Microsoft SharePoint

- CA ARCserve® Backup for Windows Agent for Microsoft SQL Server
- CA ARCserve® Backup for Windows Agent for Oracle
- CA ARCserve® Backup for Windows Agent for Sybase
- CA ARCserve® Backup for Windows Agent for VMware
- CA ARCserve® Backup for Windows Disaster Recovery Option
- CA ARCserve® Backup for Windows Disk to Disk to Tape Option
- CA ARCserve® Backup for Windows Enterprise Module
- CA ARCserve® Backup for Windows Enterprise Option for IBM 3494
- CA ARCserve® Backup for Windows Enterprise Option for SAP R/3 for Oracle
- CA ARCserve® Backup for Windows Enterprise Option for StorageTek ACSLS
- CA ARCserve® Backup for Windows Image Option
- CA ARCserve® Backup for Windows Microsoft Volume Shadow Copy Service
- CA ARCserve® Backup for Windows NDMP NAS Option
- CA ARCserve® Backup for Windows Serverless Backup Option
- CA ARCserve® Backup for Windows Storage Area Network (SAN) Option
- CA ARCserve® Backup for Windows Tape Library Option
- CA XOsoft™ Assured Recovery™
- CA XOsoft™
- Common Services™
- eTrust® Antivirus
- eTrust® Firewall
- Unicenter® Network and Systems Management
- Unicenter® Software Delivery
- Unicenter® VM: Operator®

Contact Technical Support

For online technical assistance and a complete list of locations, primary service hours, and telephone numbers, contact Technical Support at <http://ca.com/support>.

Contents

Chapter 1: Introducing the Agent	9
Architectural Overview	10
How the Agent Works	11
How a Backup Request Works	12
How a Restore Request Works	12
How Data Flows During Backup	13
Agent Services	13
Access Requirements	13
Agent Activity Log	14
Additional Functionality	14
Multiple Instance Support	14
Backup and Restore Options	15
Online Backup of SAP R/3 Databases	16
Agent for ARCserve Database	16
Chapter 2: Installing the Agent	19
Installation Prerequisites	19
Basic Prerequisites	19
Microsoft SQL Server 7.0 Cluster Environment Prerequisites	20
Microsoft SQL Server 2000 and 2005 Cluster Environment Prerequisites	21
Install the Agent	21
Install the Agent in a Standard Microsoft SQL Server Environment	21
Install the Agent in a Microsoft Server 7.0 Cluster Environment	22
Install the Agent in a Microsoft Server 2000 or 2005 Cluster Environment	22
Post-Installation Procedures	23
Configure Backup and Restore Parameters	23
Chapter 3: Backing Up Microsoft SQL Server Databases	27
Backup Overview	27
Database Backup Options	28
Backup Strategy Recommendations	31
Required Full Backup	32
Differential Backups	33
Differential Backup Timing	33
Transaction Log Backups	34
Files and FileGroup Backups	35

Partial Backups	37
Database Consistency Checks	37
Back Up Databases	39
Backup Considerations	41
Rotation Schemes and Global Options	41
Dynamic and Explicit Job Packaging	43
Dynamic Job Packaging	43
Mark Objects for Dynamic Job Packaging	44
Explicit Job Packaging	44
Mark Objects for Explicit Job Packaging	45
Chapter 4: Restoring Microsoft SQL Server Databases	47
Restore Overview	47
Restore Types	49
Differential Backup Restores	51
Transaction Log Restores	51
File and FileGroup Restores	52
Microsoft SQL Server Disaster Recovery Elements	52
Restore Master Databases	53
Restore Options	55
Restore Type Option	56
Log Point-in-Time Restore Options	59
Recovery Completion State Options	61
Database Consistency Check (DBCC) Options	62
Force Restore to use Named Pipes	62
Continue Restore After Checksum Failure	63
Miscellaneous	63
Perform an Offline Torn Page Restore Using Microsoft SQL Server 2005	65
Perform an Online Torn Page Restore Using Microsoft SQL Server 2005 Enterprise, Data Center, or Developer Editions	67
Database File Options	68
Database Restore Overview	71
Restore Types and Methods	71
SQL Agent Filter Options	78
Restore to Alternate Disk Locations Using Automatic Selection	78
Restore to Alternate Disk Locations by Individual Session	79
Chapter 5: Backing Up and Restoring in Cluster Environments	83
Backup and Restore Considerations in Microsoft SQL Server Cluster Environments	84
Microsoft SQL Server 7.0 Cluster Environment Backups	84
Select Server, Protocol, Security, and Backup Type	84

Select Backup Destination, Schedule, and Submit the Job	86
Restore by Tree in Microsoft SQL Server 7.0 Cluster Environments	87
Restore by Session in Microsoft SQL Server 7.0 Cluster Environments	90
Microsoft SQL Server 2000 or 2005 Cluster Environment Backups.....	93
Select Server, Protocol, Security, and Backup Type	93
Select Backup Destination, Schedule, and Submit the Job	94
Restore by Tree in Microsoft SQL Server 2000 or 2005 Environments	95
Restore by Session in Microsoft SQL Server 2000 or 2005 Environments	98
Perform Disaster Recovery in a Microsoft SQL Server 7.0 Cluster Environment	101
Perform Disaster Recovery in Microsoft SQL Server 2000 or 2005 Cluster Environments	101
Appendix A: Troubleshooting and Disaster Recovery	103
Agent and CA ARCserve Backup Error Messages	103
General Considerations for CA ARCserve Backup and the Agent	103
Microsoft SQL Server Error Messages	106
Limitations of Microsoft SQL Server Database	107
Replication of Microsoft SQL Server	108
MS SQL Server Disaster Recovery	108
The Master Database.....	109
Disaster Recovery Scenario.....	110
Perform Disaster Recovery in a Microsoft SQL Server 7.0 Cluster Environment	111
Perform Disaster Recovery in Microsoft SQL Server 2000 or 2005 Cluster Environments	111
Appendix B: Configuring Microsoft SQL Server Security Settings	113
Types of Microsoft SQL Server Authentication	113
Authentication Requirements	113
How User Authentication Is Changed.....	114
Check or Change the Microsoft SQL Server Authentication Method	114
Update the Agent Account Configuration.....	115
Check and Change the ODBC Settings.....	116
Update the Backup Manager	117
Index	119

Chapter 1: Introducing the Agent

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

- Back up your Microsoft SQL Server databases using CA 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 CA ARCserve Backup

The agent handles all communications between CA 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 CA ARCserve Backup.

This section contains the following topics:

[Architectural Overview](#) (see page 10)
[How the Agent Works](#) (see page 11)
[Additional Functionality](#) (see page 14)
[Online Backup of SAP R/3 Databases](#) (see page 16)
[Agent for ARCserve Database](#) (see page 16)

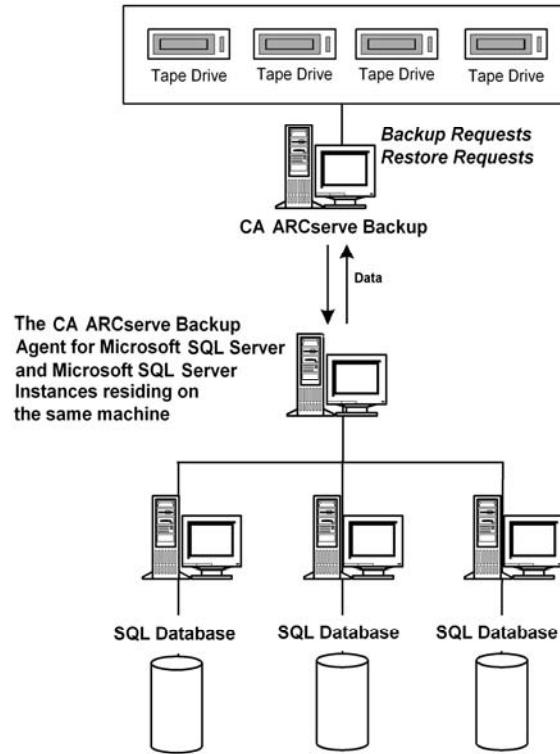
Architectural Overview

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

The agent provides services that allow CA ARCserve Backup to back up and restore Microsoft SQL Server databases. The agent must reside on the same server as Microsoft SQL Server, or on a local drive of each node in a Microsoft Cluster Services environment which contains Microsoft SQL Server. In the Cluster, the agent dynamically handles the association between Microsoft SQL Server instances and Virtual Server names, and also recognizes which instances are running on which nodes.

Note: If you are using a clustered instance of Microsoft SQL Server to house your ARCserve Database, you will need to use the stand-alone installer to install the Agent for ARCserve Database on the nodes of the cluster where the instance is not active when CA ARCserve Backup is installed. However, If you are installing a clustered CA ARCserve Database Primary Server using the Microsoft SQL Server 2005 Express Edition option for your ARCserve Database, this will not be necessary, since the agent will be automatically installed on each node along with CA ARCserve Backup before the ARCserve Primary Server is configured for Clustered operation.

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



How the Agent Works

CA ARCserve Backup and the agent work together to back up and restore SQL Server databases. When CA ARCserve Backup backs up a database, it sends a request to the agent. The agent retrieves a point-in-time-consistent image of the database or its transaction log from Microsoft SQL Server in the form of a logical data stream, and sends it to CA ARCserve Backup, where the complete database image is backed up to media. During a restore, the agent functions in a similar fashion, and transfers the backed up database from CA ARCserve Backup to Microsoft SQL Server.

The agent takes advantage of the Microsoft SQL Server dump database and transaction log backup method. 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 or transaction log submitted for backup, the agent initiates a dump in Microsoft SQL Server. Microsoft SQL Server sends the database to the agent in a series of data chunks. The agent receives the data, one chunk at a time, and passes it directly to CA ARCserve Backup, where it is recorded to backup media.

In a restore operation, for each database or transaction log backup being restored, the agent initiates a load operation in Microsoft SQL Server, and then returns the backed-up data to Microsoft SQL Server in the same manner that the data was provided during backup. If a single database requires more than one backup to be used in restoring it, CA ARCserve Backup assists in assembling the correct sequence of restore operations to be able to fully restore the database.

How a Backup Request Works

Backup jobs use the following process:

1. You issue a backup command from CA ARCserve Backup.
2. CA ARCserve Backup sends the request to the agent for a database.
3. The agent retrieves a particular database or transaction log from Microsoft SQL Server, which sends multiple data chunks to the agent.
4. The agent retrieves the data chunks and transfers them to CA 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 CA ARCserve Backup.
2. CA ARCserve Backup informs the agent of the restore job.
3. The agent instructs Microsoft SQL Server to prepare to receive the data.
4. CA ARCserve Backup accesses storage media and begins restoring data.
5. CA 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 CA ARCserve Backup uses the Agent for Microsoft SQL Server to back up a Microsoft SQL Server instance:

1. CA 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 CA ARCserve Backup.
5. CA 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 runs within the context of the CA ARCserve Universal Agent Service. This service is shared with other agents, providing a single point of access for backup and restore operations. The service starts automatically after the installation is completed, and dynamically recognizes the addition of new agents as they are installed.

Access Requirements

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

A Microsoft SQL Server native user name and password are also required to access some database instances. 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. For more information about agent security configuration, see Configuring Microsoft SQL Server Security Settings.

Note: There are two different data transfer mechanisms available to the agent, and that they have different permission requirements. A backup using Named Pipes only requires the Backup Operator permission for the specific database being backed up, and the Database Creator role to perform the database. A backup using Virtual Devices, requires the System Administrator role. However, the Named Pipes mechanism is available only for Microsoft SQL Server 7.0 and 2000.

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 `sqlpag.log` and is located in the directory in which the agent has been installed. If errors appear in the CA 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 [Multiple Instance Support and Backup and Restore Options](#).

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.

Backup and Restore Options

Backup options enable you to do the following:

- Perform a Full or Differential backup of a database
- Back up the entire database, a selected set of files and FileGroups within a database, or an automatically-selected set of FileGroups which contain changeable data
- Back up the transaction log of a database, with or without truncating the log
- Take the database offline automatically after a transaction log backup, leaving it in a restoring state
- Check the consistency of databases before and after backup
- Perform both a Data and Transaction Log backup of a database in a single backup job
- Include SQL Server's native error checking information as part of the backed-up data

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

Restore options enable you to do the following:

- Restore data and transaction logs
- Automatically determine a sequence of backups to restore, to produce a live, consistent database with a single restore job
- Use transaction log restores to restore the data to a specific point in time, or to the start or end of a named transaction
- Restore the entire database or a selected subset of the database files and FileGroups
- Restore a selected subset of the database as a new database
- Perform a Torn Page Repair of a database, even while it is online
- Leave the database in a restricted access mode
- Preserve or clear the replication settings of the restored database
- Change the physical location on disk of data and transaction log files
- Check only the physical consistency of a database after restore is complete
- Override any inconsistencies found with SQL Server's native error checking information

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 later 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 using the Agent for Microsoft SQL Server.

Agent for ARCserve Database

The Agent for ARCserve Database is a form of the CA ARCserve Backup Agent for Microsoft SQL Server. It is either installed automatically when you install CA ARCserve Backup, or manually using a special utility after the location of the CA ARCserve Backup database is changed. By itself, the Agent for ARCserve Database allows you to back up and restore the ARCserve database itself, and the system databases and Disaster Recovery Elements from the Microsoft SQL Server instance which contains the ARCserve database. When installed with the Agent for Microsoft SQL Server, it allows the Agent for Microsoft SQL Server to recognize the presence of an ARCserve database, and to work with CA ARCserve Backup to provide the special recovery mechanisms that are available for the ARCserve database.

Because the Agent for ARCserve Database is a form of the Agent for Microsoft SQL Server, it will appear as the CA ARCserve Backup Agent for Microsoft SQL Server in the system's installed programs list. If both are present, only a single entry will appear. If you need to uninstall one or the other, the installation sequence will prompt you to select which variant to remove.

You can use the stand-alone utility which installs the Agent for ARCserve Database in any of the following situations:

- When the ARCserve database is moved
- To re-install the agent if it is accidentally uninstalled
- To install the agent to additional nodes of a cluster
- To install the agent on a remote computer, if the CA ARCserve Backup installer is unable to do it directly

This utility is placed in the "Packages" sub-folder of the CA ARCserve Backup home directory, in a folder called "ASDBSQLAgent", when you install CA ARCserve Backup. If you need to install the agent on a computer which is not a CA ARCserve Backup server, you will need to copy the "ASDBSQLAgent" folder to the system where you are installing the agent, and run the utility on that machine.

Chapter 2: Installing the Agent

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

- On the same machine as Microsoft SQL Server
- On a local drive of each node in a Microsoft Cluster Services cluster which contains Microsoft SQL Server

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

This section contains the following topics:

[Installation Prerequisites](#) (see page 19)
[Install the Agent](#) (see page 21)
[Post-Installation Procedures](#) (see page 23)

Installation Prerequisites

This section provides prerequisite information you must satisfy before installing the Agent for Microsoft SQL Server in a standard Microsoft SQL environment or a Microsoft SQL Server 2000 or 2005 cluster environment.

Basic Prerequisites

Before you install the Agent for Microsoft SQL Server in a standard Microsoft SQL Server 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 the System Administrator role in Microsoft SQL Server, either as a Windows user (Windows Authentication) or through a SQL Server internal user name and password (SQL Server Authentication), for each Microsoft SQL Server instance. For each Microsoft SQL Server instance where you have this role as a SQL Server internal user, you will need to provide the 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, or in a Microsoft Cluster Services cluster environment.
- You have installed CA ARCserve Backup. For information about installing CA ARCserve Backup, see the *Implementation Guide*.
- You have selected one of the following types of installation:
 - Local installation
 - Remote installation
 - Create a response file (for 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](#).

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 the local disk of all the nodes of the cluster. This enables a backup to occur if Microsoft SQL Server 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 and 2005 Cluster Environment Prerequisites

Before you install the Agent for Microsoft SQL Server in a Microsoft SQL Server 2000 or 2005 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 or 2005 instance.
- Make a note of the Microsoft SQL Server 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 will fail for instances of the later versions.

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 CA ARCserve Backup. For the detailed steps in this procedure, see the *Implementation 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 Server 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 SQL Server 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 CA ARCserve Backup. For the detailed steps in this procedure, see the *Implementation 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:

- Select the SQL Server authentication method for the clustered Microsoft SQL Server 7.0 virtual server.
- Enter the user name and password for the system administrator (sa) or an equivalent account on the clustered Microsoft SQL Server instance. Confirm the password.
- Enter the Microsoft SQL Server 7.0 virtual server name.
- Enter the login ID of an MSCS domain user with system administrator privileges and the password for that user. Confirm the password.

Install the Agent in a Microsoft Server 2000 or 2005 Cluster Environment

To install the Agent for Microsoft SQL Server in a Microsoft SQL Server 2000 or 2005 cluster environment, follow the standard installation procedure for the system components, agents, and options of CA ARCserve Backup. For the detailed steps in this procedure, see the *Implementation 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 or 2005 virtual server:

- Click the cell containing the *instruction* in the Instance column to add virtual Microsoft SQL Server instances to the configuration window.
- Specify either Windows or SQL Server authentication in the Authentication column. If you specify SQL Server authentication, enter the user name and password of a Microsoft SQL Server user with system administrator (sa) rights for that instance. Confirm the password.
- Enter the name of the Microsoft SQL Server 2000 or 2005 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 customize the Data Transfer settings, including Virtual Device Striping parameters.

Configure Backup and Restore Parameters

Use the Microsoft SQL Agent Configuration utility to configure the Agent for Microsoft SQL Server backup and restore parameters for Microsoft SQL Server 7.0, Microsoft SQL Server 2000, and Microsoft SQL Server 2005. The parameters include settings for Microsoft Virtual Device Interface (VDI) objects and remote communication.

To configure backup and restore parameters

1. Open Windows Explorer and browse to the following directory:
`C:\Program Files\CA\SharedComponents\ARCserve Backup\UniAgent`
2. Double-click the file labeled `admin.exe`
The ARCserve Backup Agent Admin window opens.
3. From the pull-down list, select CA ARCserve Backup Agent for Microsoft SQL Server and click the Configuration button on the toolbar.
The SQL Server DBAgent Configuration dialog opens.

4. Click the Common Settings tab and specify the level of detail and synchronized recording under Agent Log Settings as follows:

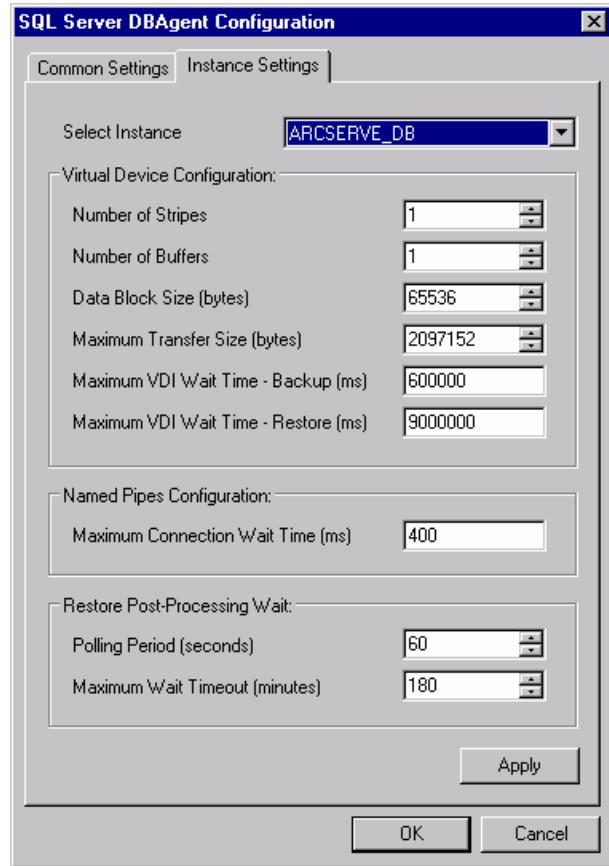
Level of Detail

Controls the settings for level of detail of the agent's Activity Log and Debugging Log, and configure the Cluster Visibility Rules. For the Activity Log settings, a Level of Detail setting of Normal (0) includes basic information about agent activity. A setting of Detail (1) includes more detailed information about agent activity. A setting of Debug (2) enables the Debugging Log at a moderate level of detail. A setting of Trace (3) enables the Debugging Log at a very high level of detail. The Activity Log is localized for your reference. The Debugging Log is meant for CA Support use, and will not be localized.

Synchronized Recording

Forces the log messages to be written to the Activity Log as they are posted. You can disable this option to improve the performance on high-load systems by caching several messages and writing them as a group.

5. Click the Instance Settings tab. The following screen appears:



6. Select the default instance (MSSQLSERVER) or the name of the instance (for Microsoft SQL Server 2000 and SQL Server 2005) for which the Agent for Microsoft SQL Server configuration is to be changed.
7. Set the parameters under Virtual Device Configuration as follows:

Number of Stripes

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 and the maximum value is 32.

Number of Buffers

The total number of VDI buffers (of maximum transfer size) used to back up and restore. The default setting is 1. This number cannot be less than the number of stripes.

Data Block Size (in bytes)

All data transfer sizes are 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.

Maximum transfer size

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.

Maximum VDI Wait Time - Backup (ms)

The time, in milliseconds, a Virtual Device object waits for a response from Microsoft SQL Server during a backup operation. This setting is also used by the agent when waiting for parallel operations to synchronize or background operations to complete, including during some parts of restore operations. The default setting is 60000 ms (ten minutes).

Maximum VDI Wait Time - Restore (ms)

The time, in milliseconds, a Virtual Device object waits for a response from Microsoft SQL Server during a restore. Increase this time if the database to be restored contains very large data files. The default setting is 9000000 ms (2.5 hours).

8. Under Named Pipes Configuration specify the Maximum Connection Wait Time (ms) time, in milliseconds, the Agent for Microsoft SQL Server should wait to close a named pipe if a remote connection fails. The default setting is 400 ms.

9. Set the parameters under Restore Post-Processing Wait as follows:

Polling Period (seconds)

The amount of time to wait between checks of the database status.
The default setting is 60 seconds (one minute).

Maximum Wait Timeout (minutes)

The total amount of time to wait before abandoning the waiting process. If this timeout elapses and the job contains additional Transaction Log sessions to be restored, then those additional sessions may fail to restore because SQL Server is not yet ready. The default setting is 180 minutes (three hours).

10. Click Apply to apply the changes to that Instance.

If you want to change the settings for another instance, select the next instance from the drop-down list, and go to step 4.

Chapter 3: Backing Up Microsoft SQL Server Databases

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

This section contains the following topics:

[Backup Overview](#) (see page 27)

[Dynamic and Explicit Job Packaging](#) (see page 43)

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 CA 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 CA ARCserve Backup, the Agent for Microsoft SQL Server initiates an online backup of the database. This backup takes 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.

Database Backup Options

The Agent for Microsoft SQL Server supports the following database backup options:

- [Backup Methods](#) (see page 28) (Full, Differential, Transaction Log Only)
- [Database Subset](#) (see page 29) (Entire, Files-and-FileGroups, Partial, and Transaction Log After Data)
- [Log Truncation Options](#) (see page 30) (Truncate, Do Not Truncate)
- [Database Consistency Checks](#) (see page 30)

Backup Methods

The following backup methods are provided:

Use Global or Rotation Options

Backs up the selected database using the job's Global or Rotation Phase Backup Method. The Global or Rotation Options provides the following options:

- The Full job method will result in a Full backup of the database.
- The Differential job method will result in a Differential backup of the database, unless this database has not yet had a Full backup.
- The Incremental job method will result in a Transaction Log backup With Truncation for databases using the Full and Bulk-Logged Recovery Models, and a Differential backup of databases using the Simple Recovery Model, unless this database has not yet had a Full backup.
- The three main System databases are exempt from the Global or Rotation job method; selecting this option for databases [master], [model], or [msdb] will always result in a Full backup.

Full

A Full backup is performed. The files included in the Database Subset will be backed up in their entirety.

Differential

Backs up data that has changed since the last Full 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. This option is not available for the [master] database.

Transaction Log

Backs up only the Transaction log. This option is only available for databases using the Full and Bulk-Logged Recovery Models.

Database Subset

Selects which files in a database will be backed up. Use this option to choose between the entire database, or a selection of files and FileGroups when the database size and performance requirements do not allow you to perform a full database backup.

Note: The Database Subset options are disabled if the selected Backup Method is Transaction Log Only.

The Database Subset options are as follows:

Entire Database

Backs up the entire database.

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. This option is only available for databases using the Full and Bulk-Logged Recovery Models.

Partial Database

Backs up the Primary FileGroup, and any other Read-Write FileGroups. For a Read-Only database, only the Primary FileGroup will be backed up. This option requires SQL Server 2005 or later.

Backup Transaction Log After Database

Backs up the transaction log after the database is backed up. This allows you to perform both a Full or Differential backup and a Transaction Log backup in the same job. This option is only available for databases using the Full and Bulk-Logged Recovery Models.

Log Truncation Options

The Log Truncation Options include the following:

Remove inactive entries from transaction log, after backup

Truncates the log files. This is the default option.

Do not remove inactive entries from transaction log, after backup

Retains inactive log entries after backup. These entries are included in the next Transaction log backup.

Backup 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 or later. Use this option to capture activity since the last backup and take the database offline to restore it.

Important! For SQL Server databases, do not use the "Backup only the log tail and leave the database in unrecovered mode" log truncation option to back up the ARCserve Database. Performing a backup with this option causes the database to be placed in an offline status, and you can lose the ability to find the backups of the ARCserve Database in order to perform a restore and bring the database online. If you perform a backup of the ARCserve Database using this option, you can use the ca_recoverdb command line utility to restore the ARCserve Database and bring it back online.

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 a consistency check before backup reports errors.

After restore

Performs DBCC after the restore of the database.

Before restore

Performs DBCC before an Online Torn Page Repair restore of the database. (SQL Server 2005 or later, Enterprise Edition only)

Do not check indexes

Checks the database for consistency without checking indexes for user-defined tables.

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

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 option is available for Microsoft SQL Server 2000 or later.

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

Other Options

Include Checksum Generated by SQL Server

Includes error checking information from Microsoft SQL Server, which can be used to validate the integrity of the backed-up data during restore. This option requires SQL Server 2005 or later.

Backup Strategy Recommendations

To establish a good backup strategy, follow these recommendations:

- If your database activity is low to medium, we recommend the following pattern:
 - 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 pattern:
 - 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 pattern:
 - 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 pattern:
 - 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 Database Differential backup, Transaction Log backup, Partial Database backup (full or differential), or Files and FileGroups backup (full or differential), 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
- Changed the arrangement of tables among the FileGroups
- Changed the partition scheme of a Partitioned Table or Index
- Canceled a full backup job while it is running
- Modified the database definition, including adding or removing a table, or changing the definition of 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. Usually, these backups are smaller and faster than full database backups, but larger and longer 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 the 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 replay 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 record Microsoft SQL Server database activity. When using the Full or Bulk-Logged Recovery Models, they should be backed 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

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.

It is sometimes possible to perform a Transaction Log backup of a damaged database. If the database is in the Suspect or Damaged state, and its transaction log files are intact, you will be able to perform a Transaction Log backup without Truncation. This will allow you to recover the database to the point in time immediately before the failure.

Important! Transaction logs are not backed up during full or differential database backups. You must back them up by running separate Transaction log backups or using the Backup Transaction Log After Database option. Transaction logs are only Truncated as part of a Transaction Log backup. If they are not backed up and truncated, they may grow until the disk is full. If this happens, you must perform a Transaction Log backup With Truncation, and then shrink the Transaction Log files to reclaim disk space.

Note: Databases which use the Simple Recovery Model do not permit or require Transaction Log backups. Microsoft SQL Server manages the maintenance of the Transaction Logs for these databases automatically.

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 database backup performed since that full database backup and before the selected Transaction Log backup, if any
- Any other Transaction log backups performed since the last full database or differential backup and before the selected Transaction Log backup

You can alternatively also restore the following:

- A prior full database backup
- Optionally, a differential database backup that is newer than the full backup selected, and older than the next full database backup, if any
- Every Transaction Log backup performed since the selected full database or differential database 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 against 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.

Files and FileGroup Backups

You can 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 back up an individual file instead of the full database, create procedures 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.

Microsoft SQL Server 2000 and later allow both Full and Differential File and FileGroup backups. If a File and FileGroup Differential backup is performed, restore of this session is dependent on the last Full backup of each file being restored. If you perform Full and Differential backups with different file sets but which contain some files in common, ARCserve will not be able to construct the correct dependency sequence to restore these backups, and you will need to do so manually.

Important! In Microsoft SQL Server 2000, a Database Differential backup will not be dependent on any Files-and-FileGroups Full backups that might be between it and the previous Database Full backup. However, in SQL Server 2005, a Database Differential backup is dependent on any intervening Full backup, including Files-and-FileGroups Full and Partial Full backups. The Automatic Selection feature cannot detect this condition. If you mix Files-and-FileGroups backups with Database Differential and Partial Differential backups in SQL Server 2005, you may need to assemble the restore sequences manually.

Note: You cannot perform Database or Partial restores from a Files-and-FileGroups backup. Only a Files-and-FileGroups restore can be performed from a Files-and-FileGroups backup.

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 when the backup job is run in one or more error messages, which will be written to the Activity Log by the Agent for Microsoft SQL Server.

Partial Backups

Microsoft SQL Server 2005 introduces a special type of Files-and-FileGroups backup called a Partial Backup. These backups automatically select the Primary FileGroup, along with all other FileGroups which are not Read-Only. If the database itself is Read-Only, then only the Primary FileGroup will be included.

Unlike Files-and-FileGroups backups, Partial Backups can be performed on databases which use the Simple Recovery Model, because all of the data which can be changed is included. If you have a large database which contains a large amount of static data in Read-Only FileGroups, you can use a Partial Backup to reduce the size and time required to perform regular backups. As long as you retain the latest Full Database backup and the database structure does not change, you can use Partial Full and Partial Differential backups without needing to perform another Full Database backup.

Also, you can perform a Partial Restore from a Partial Backup, provided all of the desired FileGroups are contained within the Partial Backup session. You can also perform a Database Restore from a Partial Backup, with the last Database Full backup as a pre-requisite.

Note: When using the Simple Recovery Model, you can not perform a Database Differential backup after a Partial Full backup, until you first perform another Database Full backup. If you perform a Partial Full backup after a Database Differential backup, the Partial Full backup will be dependent on the Database Differential backup as a pre-requisite.

Important! In SQL Server 2005, a Database Differential backup is dependent on any intervening Full backup, including Files-and-FileGroups Full and Partial Full backups. If a Files-and-FileGroups Full or Partial Differential backup is performed, restore of this session is dependent on the last Full backup of each file being restored. If you perform Full and Differential backups with different file sets but which contain some files in common, ARCserve will not be able to construct the correct dependency sequence to restore these backups, and you will need to do so manually.

Database Consistency Checks

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

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.

DBCC CHECKCATALOG

Checks for consistency in and between system tables in the specified database.

Database Consistency Checks can be performed before a backup after a backup (except a Log Tail backup), or both. 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 a consistency check before backup reports errors.

After restore

Performs DBCC after the restore of the database.

Before restore

Performs DBCC before an Online Torn Page Repair restore of the database. (SQL Server 2005 or later, Enterprise Edition only)

Do not check indexes

Checks the database for consistency without checking indexes for user-defined tables.

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

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 option is available for Microsoft SQL Server 2000 or later.

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

Back Up Databases

Use the Backup Manager in CA ARCserve Backup to backup databases.

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 Backup Universal Agent, if necessary.
Note: This service is started automatically when the agent is installed and is 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 running. Expand the Microsoft SQL Server instance to display a list of databases.
4. If you are backing up from a Microsoft SQL Server 7.0 or 2000 instance, right-click the Microsoft SQL Server instance and select Transfer Method from the pop-up window. The Remote Protocol dialog opens. Select a data transfer mechanism and click OK.

If you are backing up from Microsoft SQL Server 2005 instance, go to the next step.

The default mechanism is Virtual Devices.

Note: Select Named Pipes only if you have backup operator rights, but do not have system administrator rights, or if you cannot use Virtual Devices. Named Pipes is not available for Microsoft SQL Server 2005.

In prior releases of this agent, Virtual Devices were always used for local backups and TCP/IP backups.

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 how you can select a database to apply options properly for backup, see Dynamic and Explicit Job Packaging.
6. Right-click the database object and select Backup Agent Options from the pop-up window. The Backup Agent Options dialog for Microsoft SQL Server opens. The options available will vary based on the version of the Microsoft SQL Server instance and the Recovery Model of the database.
7. Select the backup method you want to perform, and the subset type you want to perform it on. For more information about backup methods and subsets, see Backup Methods.
8. If you chose the Files-and-FileGroups subset, click the Browse 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 and the Microsoft SQL Server documentation.
10. If you selected the Transaction Log backup type, or the Backup Log After Database option, select a Transaction Log Truncation Option.
11. Click OK.
12. Repeat the preceding steps for each database you are backing up in this job.
13. 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 *Administration Guide*.

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

16. 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.
17. After verifying or changing the security information, click OK. The Submit Job dialog opens.
18. (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.
19. Click OK.

The job is submitted.

If you selected the Run Now option, 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 *Administration 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 and reporting snapshots of databases, see the Microsoft SQL Server 2005 documentation.

Important! Performing a backup can slow the system down. For better backup performance and to minimize impact on applications using the database, run backups when the database is not being heavily updated.

Rotation Schemes and Global Options

CA 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 *Administration Guide*.

You can use the following backup methods:

- **Full Backup Method:** A full database backup is performed. If the Backup Log After Database option is selected, the Transaction Log will be backed up as well.
- **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 or 2005 databases, and a File-and-FileGroup full backup is performed for Microsoft SQL 7.0 databases.
- If the backup options selected for the target database include selection of the Partial subset (available only for SQL Server 2005 or later), a Partial Differential backup is performed.
- In all other circumstances, a differential database backup is performed.
- If the Backup Log After Database option is selected, the Transaction Log will be backed up as well.

■ **Incremental Backup Method:**

- If you are backing up a system database (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, and the Database subset was selected, a Database Differential backup is performed.
- If the database uses the Simple Recovery Model, and the Partial subset was selected, a Partial Differential backup is performed.
- In all other circumstances, a Transaction Log backup With Truncation is performed.
- If the Backup Log After Database option is selected, the Transaction Log will be backed up only once.

For more information about rotation schemes and automatic backups, see the *Administration Guide*.

Note: In previous releases of the Agent, the Global Backup Method or Rotation Scheme would override local database options. In this release, the Global Backup Method or Rotation Phase will only be applied if the Use Global or Rotation option is selected. If the Agent has been upgraded from a previous release and the options in the job have not been manually modified since then, the Database Full option will be treated the same as the Use Global or Rotation option with the Database subset, and the Files-and-FileGroups option will be treated the same as the Use Global or Rotation option with the Files-and-FileGroups subset.

When you backup a Microsoft SQL Server instance which includes Full Database backups of the [master], [model] and [msdb] system databases, the Agent for Microsoft SQL Server will generate an additional backup session called Microsoft SQL Server Disaster Recovery Elements. This session contains an image of the [master] and [model] databases which can be restored as files, with the Microsoft SQL Server instance offline. Restoring this session will take the place of performing a Rebuild [master] Database operation. This allows you to bring Microsoft SQL Server online, allowing you to restore from backups which have been taken online.

Dynamic and Explicit Job Packaging

CA ARCserve Backup provides the following 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, CA 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.

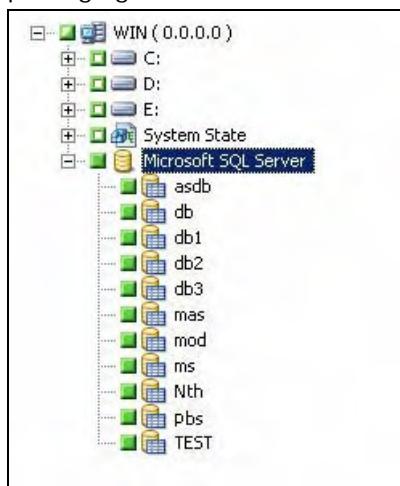
Note: If you select the full SQL instance backup and then mark the individual agent backup option, the individual agent backup option will be ignored. Only when you select the individual database and mark the agent backup option, the agent backup option will be packaged and retained when you want to modify the job.

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

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

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 CA ARCserve Backup, the Agent for Microsoft SQL Server, and Microsoft SQL Server 7.0, Microsoft SQL Server 2000 or Microsoft SQL Server 2005.

This section contains the following topics:

- [Restore Overview](#) (see page 47)
- [Restore Options](#) (see page 55)
- [Perform an Offline Torn Page Restore Using Microsoft SQL Server 2005](#) (see page 65)
- [Perform an Online Torn Page Restore Using Microsoft SQL Server 2005](#)
- [Enterprise, Data Center, or Developer Editions](#) (see page 67)
- [Database File Options](#) (see page 68)
- [Database Restore Overview](#) (see page 71)

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 CA 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 being restored, Microsoft SQL Server reinitializes any remaining unused pages. For example, if a 100 MB database contains only 5 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 a restore is attempted when you are accessing a database, Microsoft SQL Server will not allow the restore operation.

Important! By default, Microsoft SQL Server 2005 will not allow a restore to be attempted if a database is online and is using the Full or Bulk-Logged Recovery Models. To restore these databases, you must either take the database offline by performing a Transaction Log backup with the Log Tail option so that no transactions are lost, or restore with the Overwrite Existing Database and Files ("WITH REPLACE") option. The only exception to this is an Online Torn Page Repair restore.

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 may be left in a loading state and will be 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 available for it as the backed up database. The actual amount of data in the backed up database is not relevant for this requirement. To get information about allocated storage space, use the Microsoft SQL Enterprise Manager or the DBCC CHECKALLOC statement. If the database is offline, then check the file sizes listed in the file tree of the Agent Restore Options dialog for that session, or in the ARCserve Database Manager view.

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 or Management Studio.

Restore Types

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

Database restore

Restores the entire database. If the selected session is a Database Differential backup or a Partial Full backup, the last prior Database Full backup is required as a pre-requisite. If the selected session is a Partial Differential backup, the last Database Full or Partial Full backup is required as a pre-requisite.

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 restoring a database from a Full Backup, 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.

Note: A Transaction Log restore must be applied to the original database or to a restored copy of the original database. Otherwise, the logs will not be applicable.

Files and FileGroups restore

Restores selected files and FileGroups. You can restore files and FileGroups from either a Files-and-FileGroups backup, Partial backup or Database backup. When restoring files or FileGroups, you must first restore from a Full backup session, then optionally from a differential session, followed by all of the Transaction log backup sessions performed after the Full or Differential 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.

Note: A Files and FileGroups restore must be applied to the original database or to a restored copy of the original database. Otherwise, the logs will not be applicable.

Partial restore

A partial restore always restores the primary FileGroup and any other FileGroups you specify as a new database. The result is a subset of the database. FileGroups that are not restored are marked as offline and are not accessible.

Note: A database that is created using a Partial Restore may not be eligible for a Database backup, due to the unrestored FileGroups being offline. To resolve this condition, either restore the remaining FileGroups to the partially-restored database, or remove them from the database structure using the Microsoft SQL Server Enterprise Manager or Management Studio.

Torn Page Repair

Restores only the pages of data on disk which have been marked as damaged. SQL Server 2005 has the ability to isolate the damaged parts of a database while leaving the rest intact. When this happens with a Full Recovery Model database, the database may be eligible for a Torn Page Repair, which is much faster than a Database restore. The dependencies for this restore are similar to a Files and FileGroups restore.

There are both Online and Offline versions of this operation. In an Online Torn Page Repair, the database remains online the entire time, and undamaged tables are still accessible. In an Offline Torn Page Repair, the database should be taken offline using a Log Tail backup before the restore is performed. Online Repair requires the Enterprise Edition of SQL Server.

Note: Torn Page Repair restore must be applied to the original database, or to a restored copy of the original database. Otherwise, the data and logs will not be applicable.

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 CA ARCserve Backup or BrightStor® Enterprise Backup. For more information about restoring to a different location, see [Restore to Alternative Disk Locations Using Automatic Selection](#).

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 chooses the appropriate full-database backup session and the proper options for you. 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 when you created the differential backup. They cannot be used to restore the database to the exact point of failure or to a specific time.

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, partial, or files-and-FileGroups) 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 chooses the appropriate Transaction log, differential, and database backups, and automatically sets the proper options for each. 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 Files-and-FileGroups backup, Partial backup, or from a Database backup. When using this option, you must apply the Transaction Log to the database immediately after the last Files-and-FileGroups 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 Files-and-FileGroups restore and selected the Automatic Selection option, Automatic Selection 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 Files-and-FileGroups restore from a Differential session and selected the Automatic Selection option, Automatic Selection attempts to select the Full backup session on which the differential was based, and all of the Transaction log backups needed to run the restore successfully.

Important! In Microsoft SQL Server 2000, a Database Differential backup will not be dependent on any Files-and-FileGroups Full backups that may be between it and the previous Database Full backup. However, in SQL Server 2005, a Database Differential backup *is* dependent on any existing Full backup, including Files-and-FileGroups Full and Partial Full backups. In Microsoft SQL Server 2005, a Differential backup is dependent on the latest Full backup, including Files-and-FileGroups Full and Partial Full backups, which contains the selected file. If a Files-and-FileGroups Full or Partial Differential backup is performed, restoring this session is dependent on the last Full backup of each file being restored. If you perform Full and Differential backups with different file sets which contain some common files, ARCserve will not be able to construct the correct dependency sequence to restore these backups. You should restore these backups manually.

Microsoft SQL Server Disaster Recovery Elements

When you backup a Microsoft SQL Server instance which includes Full Database backups of the [master], [model] and [msdb] system databases, the Agent for Microsoft SQL Server will generate an additional backup session called Microsoft SQL Server Disaster Recovery Elements. This session contains an image of the [master] and [model] databases which can be restored as files, with the Microsoft SQL Server instance offline. Restoring this session will take the place of performing a Rebuild [master] Database operation. This allows you to bring Microsoft SQL Server online, allowing you to restore from backups which have been taken online.

The Disaster Recovery Elements session will appear in the Restore by Tree view with the name "Microsoft SQL Server Disaster Recovery Elements", followed by the Instance Name for a named instance of Microsoft SQL Server. In the Restore by Session view, it will have a volume name of "sqldr@" followed by the instance name. For a default instance of Microsoft SQL Server, the instance name will appear as "MSSQLSERVER", regardless of the SQL Server version.

Restoring a Microsoft SQL Server Disaster Recovery Elements session to its original location will restore the files to the location where the [master] and [model] database files existed. Restoring them to an Alternate Location and selecting a disk or directory will place the files in the directory that you select. Restoring them to an Alternate Location and selecting only the computer will restore them to the original file path on the selected computer.

Note: If you attempt to restore SQL Server Disaster Recovery Elements to their original location while the database is online, the restore will fail because Microsoft SQL Server is using the existing files.

After restoring Disaster Recovery Elements, you should immediately restore the [master] database, the [msdb] database if it is offline, and the [model] database, from the regular online backups.

Important! Disaster Recovery Elements are specific to the instance that they were backed up from. If you use them for a different instance, that instance may not start, or may behave incorrectly after starting.

Note: The CA ARCserve Backup Disaster Recovery Option will include the Microsoft SQL Server Disaster Recovery Elements sessions when performing a Disaster Recovery restore. If you use the Disaster Recovery Option to restore the computer which houses the ARCserve Database, you should restore the three system databases first after the Recover Database operation is finished. For more information, see the *Administration Guide*.

Restore Master Databases

Before you restore the [master] database, if the database files are missing, you may need to either restore Microsoft SQL Server Disaster Recovery Elements for that instance, or rebuild it 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, the Microsoft SQL Server instance must be running in single-user mode. For complete instructions on rebuilding your Microsoft SQL Server master database, see the Microsoft SQL Server documentation.

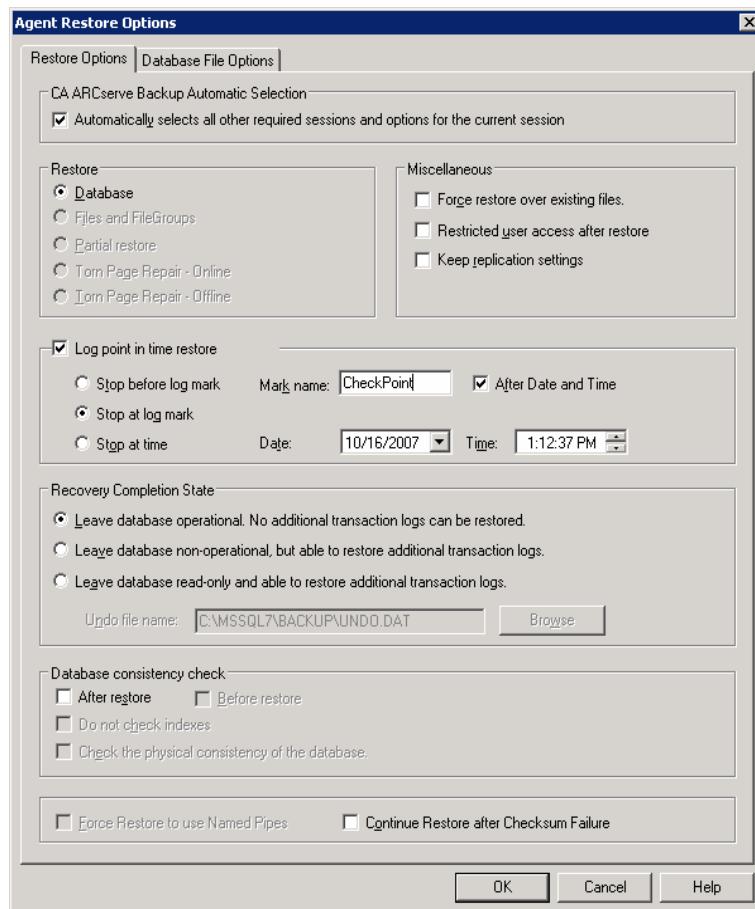
Before restoring the [master] database, you must stop all applications which may be using that instance of Microsoft SQL Server. This includes some secondary services which are part of Microsoft SQL Server, such as the SQL Server Automation Agent (“SQL Agent” service). The only exception to this is CA ARCserve Backup itself; the Agent for ARCserve Database will ensure CA ARCserve Backup is not using any ARCserve Database that the instance might contain.

When you restore the [master] database, the Agent for Microsoft SQL Server automatically detects this. The agent will restart Microsoft SQL Server in Single-User Mode before executing the restore operation, and will return Microsoft SQL Server to Multi-User Mode at the conclusion of the [master] database restore.

Important! You must stop *all* applications and services other than CA ARCserve Backup which may be using databases in the Microsoft SQL Server instance for which you are restoring the [master] database. Otherwise, one of these other applications may re-connect to the SQL Server instance after it is restarted, and prevent the agent from performing the restore.

Restore Options

The Agent for Microsoft SQL Server provides restore options as shown in the following dialog:



- [Automatic Selection](#) (see page 56)
- [Restore Type](#) (see page 56): Database, Files or FileGroups, Partial (Microsoft SQL Server 2000 and later), or Torn Page Repair (Microsoft SQL Server 2005)
- [Log Point in Time Restore](#) (see page 59)
- [Recovery Completion State](#) (see page 61)
- [Database Consistency Check](#) (see page 62)
- [Force Restore to use Named Pipes](#) (see page 62)
- [Continue Restore After Checksum Failure](#) (see page 63)
- [Miscellaneous Options](#) (see page 63)

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
- Applies selected options appropriately to the automatically selected sessions

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) using backups from previous versions of CA ARCserve Backup or BrightStor® Enterprise Backup. For more information about restoring to a different location, see [Restore to Alternative Disk Locations Using Automatic Selection](#).

Restore Type Option

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

Database

Restores complete databases, differential backups, and Transaction log backups.

Files and FileGroups

Restores selected files and FileGroups. After you have selected the File or FileGroups option, you must select the files you want to restore.

Partial Restore

Restores part of a database to its original location or another location so you can copy damaged or missing data back to the original database. You can use this option with Database and Partial backup sessions only.

Torn Page Repair - Offline

Repairs databases in place without the need to perform a restore of the entire database. This operation is recommended when only a few pages are damaged and an immediate recovery is critical.

The database should first be taken offline by performing a Transaction Log backup with the Log Tail option. A Database Consistency Check Before Backup with the Continue if DBCC Fails option is recommended to identify damaged pages which are not recognized, and forestall the possible need of repeating the process. This option is available for all editions of Microsoft SQL Server 2005.

The Torn Page Repair restore can then be performed from the latest Full or Differential backup session of that database. If the Automatic Selection option is selected, all of the successive Transaction Log sessions are located, as they would for a Files-and-FileGroups restore. If a Differential session is selected, then the corresponding Full backup session will also be automatically selected. The database remains offline until the restore is complete.

Note: Microsoft recommends this only as an emergency measure. A Torn Page Repair can be used to return a damaged database to service when time is critical, but it is strongly recommended that you migrate the database to a new disk at the earliest opportunity to forestall the risk of further errors.

Torn Page Repair - Online

This option requires the Enterprise Edition of Microsoft SQL Server 2005. A Transaction Log backup with the Do Not Truncate option is used to obtain the latest transaction information which might need to be applied to the damaged pages. A Database Consistency Check Before Backup with the Continue If DBCC Fails option is recommended to identify any damaged pages which have not yet been encountered, and forestall the possible need to repeat the process.

You can then perform the Torn Page Repair restore from the latest Full or Differential backup session of that database. If the Automatic Selection option is selected, Automatic Selection will locate all of the successive Transaction Log sessions, as they would for a Files-and-FileGroups restore. If a Differential session is selected, then the corresponding Full backup session will also be automatically selected.

The database remains online during the entire process and any tables which are not affected by the damaged pages will remain accessible.

Note: In some cases, you may need to perform an additional Transaction Log Backup with the Do Not Truncate option, and restore that backup without the Automatic Selection option, to fully reactivate the repaired tables. This usually occurs if such a backup was not taken at the start of the process.

For Torn Page Repair restores, the Recovery Completion State option is restricted to the Leave Database Online option.

The Database Consistency Check Before Restore option is only enabled when using the Torn Page Repair – Online option, as this is the only time the database will be online during a restore. If a Database Consistency Check was not performed before the last Transaction Log backup, this option can be used to ensure that Microsoft SQL Server identifies any additional torn pages.

Note: Microsoft recommends this only as an emergency measure. A Torn Page Repair can be used to return a damaged database to service when time is critical, but it is strongly recommended that you migrate the database to a new disk at the earliest opportunity to forestall the risk of further errors.

Requirements to Restore Files or FileGroups

After restoring a selected set of files and FileGroups, 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.

Log Point-in-Time Restore Options

The Log Point-in-Time Restore option restores a database to the state it was in at a date and time, or as of a named transaction, that you have specified. You should use Automatic Selection with this option. This option is only available when the selected session is a Transaction Log backup.

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 CA 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 log 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

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 at log mark

The option recovers the database activity through the transaction which is marked with the specified name, including the transaction that contains the mark. If you do not select the After Date and Time option, recovery stops at the first marked transaction with the specified name. If you select the After Date and Time option, recovery stops at the first marked transaction with the specified name exactly at or after the specified date and time.

Note: This option is available in Microsoft SQL Server 2000 and Microsoft SQL Server 2005.

Stop before log mark

The option recovers the database activity prior to the transaction which is marked with the specified name. The transaction which contains the mark is not re-executed. If you do not select the After Date and Time option, recovery stops at the first marked transaction with the specified name. If you select the After Date and Time option, recovery stops at the first marked transaction with the specified name exactly at or after the specified date and time.

Note: This option is available in Microsoft SQL Server 2000 and Microsoft SQL Server 2005.

After Date and Time

Allows you to specify a point in time after which Microsoft SQL Server will look for the specified Log Mark. The recovery stops at the specified mark only when the timestamp for the log mark is later than the specified time. The same Date and Time entry fields are used as for the Stop at Time option. Use this option with the Stop at Log Mark or Stop Before Log Mark options.

Recovery Completion State Options

The Recovery Completion State options let you specify the final state of a database after restore. The following options are available:

Leave database operational. No additional transaction logs can be restored

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 for each session. CA ARCserve Backup performs the selection of sessions and applies the appropriate options automatically to each session. 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

Instructs the restore operation *not* to roll back any uncommitted transactions and to leave the database in a state where it can accept additional Files-and-FileGroups, Differential, or Transaction Log restores. You must choose either this option or the Leave database read-only option to apply another differential backup or Transaction log after this restore job is complete. This option is generally used when restoring a database from multiple sessions without using Automatic Selection.

Leave database read-only and able to restore additional transaction logs

Prepares a standby (warm backup) database. A standby database is a second database on a different server which you can bring online if the primary production server fails. It contains a copy of the database on the primary server. The database is brought online in a special form of Read-Only state which can be reverted to a Restoring state if another restore occurs. An Undo File is created during the restore which contains the information SQL Server needs to make this transition. The location and filename of the Undo File must be specified in the restore options.

Note: Standby databases are not eligible for backup. If a standby database is explicitly selected for backup, the backup will fail. If a SQL Server instance which contains a standby database is selected for backup, the standby database will be excluded by the agent.

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.

The Database Consistency Check option lets you restore using the following options:

After restore

Performs consistency checks for the specified database, after the database is restored.

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.

Before restore

Performs consistency checks for the specified database before the database is restored. This option is only available for an Online Torn Page Repair, since this is the only restore method in which the database is online before the restore begins and is not completely overwritten.

Force Restore to use Named Pipes

This option forces the agent to use Named Pipes to return the data to the Microsoft SQL Server. This provides an alternate mechanism for restoring a session if the Virtual Device Interface is not functioning properly. This option is available for Microsoft SQL Server 7.0 and Microsoft SQL Server 2000.

Continue Restore After Checksum Failure

This option allows Microsoft SQL Server 2005 to continue processing a restore if it detects an inconsistency between the data and checksums included in the backup.

Miscellaneous

The following are the miscellaneous options you can select from:

Force Restore Over Existing Files and Databases

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.

Important! Microsoft SQL Server 2005 will refuse by default to overwrite an online database using the Full or Bulk-Logged Recovery Model. Instead, it produces an error message saying to either take the database offline by performing a Log Tail backup, or restore with the “WITH REPLACE” option. Selecting this option applies the “WITH REPLACE” option to the restore, and forces SQL Server to overwrite the existing database.

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

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. This 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.

Use Current ARCserve Database as Original Location

Overwrites the current ARCserve Database instead of the database which was backed up to this session when a restore to Original Location is selected. You can use this option to migrate the session and log information from one ARCserve Domain to another.

Preserve Current ARCserve Domain Memberships

Retrieves the current information about ARCserve Domains, such as the ARCserve Domain name, Primary Server identity, and Member Server identities from the destination database before the restore begins. This information is written back after the restore is completed so that the information is preserved even after the restore. This option is enabled when the Automatic Selection, Leave Database Operational and Use Current ARCserve Database as Original Location options are selected.

Perform an Offline Torn Page Restore Using Microsoft SQL Server 2005

Microsoft SQL Server 2005 has the ability to detect when the data in the database has been damaged and isolate the damage at the data page level. You can find the current list of known damaged pages in the [suspect_pages] table of system database [msdb], at any point of time. In addition to torn page detection and isolation, SQL 2005 also introduces the ability to perform a restore in which only those data pages which are damaged are overwritten. This will allow you to bring a database which is slightly damaged back into operation quickly.

Note: Do *NOT* take the database offline before you start the restore procedure.

To perform an offline torn page restore using Microsoft SQL Server 2005

1. Change to Full Recovery Model, if the database is using the Simple Recovery Model.
2. (Optional) Perform a Database Consistency Check (DBCC CheckDB) on the database to locate any additional damaged pages beyond the one already reported. This can be done as part of step #4.
3. Disconnect all clients using the database. (Otherwise, the following step will fail).
4. Perform a Transaction Log backup with the Backup only the log tail and leave the database in an unrecovered mode (Log Tail) option. If you have not performed step 2 separately, you should also select the Database Consistency Check Before Backup option and Continue Backup if DBCC Fails option.
5. Perform an Offline Torn Page Repair Restore of the database as follows:
 - a. Open the Restore Manager.
 - b. On the Source tab, use the Restore By Tree view to find and select the database.
 - c. Open the Restore Options.
 - d. Confirm that Automatic Selection is selected.
 - e. Under Subset, select Torn Page Repair – Offline.
 - f. Under Recovery Completion State, select Leave database operational.
 - g. (Optional) You may select a Database Consistency Check before the restore, to identify any other damaged or corrupted pages. (Note that this is the only restore type for which a DBCC is permitted before a Restore, because DBCC requires the database to be online.)

- h. (Optional) You may select a Database Consistency Check after the restore.
- i. Click OK.
- j. On the Destination tab, select Restore to Original Location, if it is not selected.
- k. Start the Restore.
6. Change to Simple Recovery Model, if you changed the Recovery Model in step 1.
7. Resume use of the database.

Perform an Online Torn Page Restore Using Microsoft SQL Server 2005 Enterprise, Data Center, or Developer Editions

Microsoft SQL Server 2005 has the ability to detect when the data in the database has been damaged, and isolate the damage at the data page level. At any point in time, the current list of known damaged pages can be found in the [suspect_pages] table of system database [msdb]. In addition to torn page detection and isolation, SQL 2005 also introduces the ability to perform a Restore in which only those data pages which are damaged are overwritten. This will allow you to bring a database which is slightly damaged back into service quickly.

Note: Do *NOT* take the database offline before you start the restore procedure.

To perform an online torn page restore using Microsoft SQL Server 2005 Enterprise, Data Center, or Developer Editions

1. Change to Full Recovery Model, if the database is using the Simple Recovery Model.
2. (Optional) Perform a Database Consistency Check (DBCC CheckDB) on the database to locate any additional damaged pages beyond the one already reported. This can be done as part of step 4.
3. Perform an Online Torn Page Repair Restore of the database as follows:
 - a. Open the Restore Manager.
 - b. On the Source tab, use the Restore By Tree view to find and select the database.
 - c. Open the Restore Options.
 - d. Confirm that Automatic Selection is selected.
 - e. Under Subset, select Torn Page Repair – Online.
 - f. Under Recovery Completion State, select Leave database operational.
 - g. (Optional) You may select a Database Consistency Check before the restore, to identify any other damaged or corrupted pages. (Note that this is the only restore type for which a DBCC is permitted before a Restore, because DBCC requires the database to be online.)
 - h. (Optional) You may select a Database Consistency Check after the restore.
 - i. Click OK.
 - j. On the Destination tab, select Restore to Original Location, if it is not selected.
 - k. Start the Restore.
4. Attempt to query the table that contained the damaged page.

5. Perform a Transaction Log backup with the default options.
6. Perform a restore of this final Transaction Log backup without Automatic Selection, and with the Recovery Completion State set to Leave Database operational.
7. Change to Simple Recovery Model, if you changed the Recovery Model in step 1.
8. Resume use of the database.

Database File Options

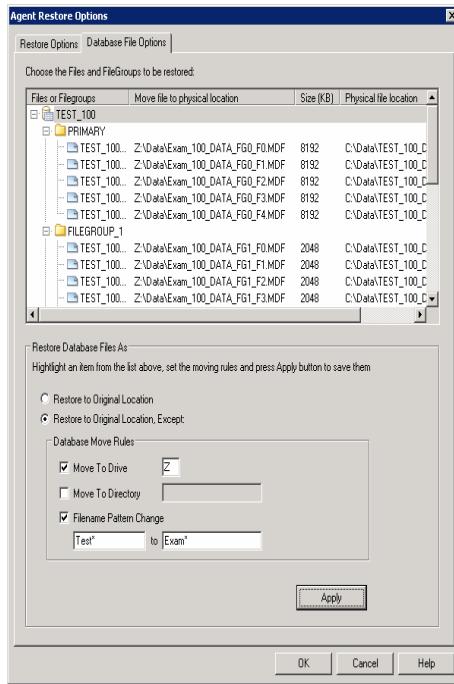
Using the Database File options, you can:

- View the list of files for the database and related information.
- Select the files to be restored in a Files and FileGroup restore, or the FileGroups to be restored in a Partial Restore.
- Change the location or file names of files during a restore. Use the location change function only when restoring from Full backups, or when using Automatic Selection.

You can move the files to a different location on disk either individually or using Move Rules. Move Rules can be applied to an entire database, individual FileGroup or Transaction Log, or to an individual file. You can specify rules to move files to a particular drive, directory path, or rename the file, all independently of each other. If you want to rename files at the Database or FileGroup level, you can specify a filename change using a wildcard pattern.

To restore the database files

1. On the Agent Restore Options dialog, select the Database File Options tab.



2. In the navigation tree, select one of the following options to apply the change:
 - Select the database, if you want to apply the rule to all of the files in the database.
 - Select the FileGroup or the Transaction Log, if you want to apply the rule to the files in a particular FileGroup or the Transaction Log.
 - Select the file, if you want to change just a single file.
3. Under Restore Database Files As, select one of the following options, as appropriate:

Restore to Original Location

Available at the Database level. Clears any changes to the drive letters, paths, and file names. You must still click the Apply button after selecting this option for the change to take effect.

Restore to Original Location, Except

Available at the Database, FileGroup and Transaction Log, and File levels. Applies the requested changes to the drive letter, paths and file names based on the location of the file when the backup was performed.

Inherit Move Rules, Except

Available at the FileGroup and Transaction Log, and File levels. Applies the requested changes to the drive letter, paths and file names based on the changes which have already been made.

4. In the box labeled Database Move Rules, FileGroup Move Rules, or File Move Rules, select one or more of the following options:
 - Select the Move To Drive check box and enter a different drive letter in the field beside.
 - Select the Move To Directory check box and enter a different directory path in the field beside.
 - Select the Filename Pattern Change check box, to change the filenames for the entire database, FileGroup, or Transaction Log. Enter a wildcard pattern that matches the names of the files you want to rename in the field below and enter the wildcard pattern that you want it to be renamed to in the to field.
For example, if you want to rename all the files that begin with Group as Members, enter Group* in the field and Member* in the to field.
Note: If you are using a wildcard pattern to rename files, and the pattern for the original filenames does not match with one or more of the files to which it would be applied, a yellow indicator will appear at the bottom of the dialog, and in the tree next to both the affected files and the object where the rule was applied.
 - Select the Rename File check box and enter a different file name, to rename a single file.

5. Click the Apply button.

The changes are applied to the tree.

Note: If you make a different selection or close the Agent Restore Options dialog without applying the changes, the selections you made will be lost.

If the rules you specify result in two files being given the same physical location on disk, a red indicator appears at the bottom of the dialog, and in the tree next to both the affected files and the object where the rule was applied.

6. Repeat the steps from the point at which you make a selection from the navigation tree for each change that you need to make.

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:

- Database Full and Differential backups
- Partial Full and Differential backups
- Transaction log backups
- File and FileGroup Full and Differential 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 CA 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 CA 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: CA ARCserve Backup supports only Restore by Tree and Restore by Session for Microsoft SQL Server databases.

Restore Databases Using Restore by Tree Method

To restore using the Restore by Tree method

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 window. The Backup Agent Restore Options dialog appears. The dialog varies depending on the backup method of the selected session and the version of SQL Server from which the database was backed up.
5. Perform one of the following to select restore options:
 - Click OK to accept the default options and allow the Automatic Selection option select the appropriate restore sequence and 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 sequence and click OK. For more information about options, see [Restore Options](#).

Important! The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the sessions which must be restored and applies the selected options appropriately to each session.
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 the same version of Microsoft SQL Server or higher.
 - 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 clustered instance, you must select it from the Virtual Windows Server machine.

- 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 examples:

\\SERVER1\\MSSQLSERVER\\Lightning

\\SERVER2\\WEATHER\\Thunder

Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 or 2005 instance, you must explicitly select the target instance even if it is a default instance.

8. Click Start.

The Restore Media dialog appears.

9. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.

10. 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.
11. 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.
12. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
13. (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.
14. 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 *Administration Guide*.

Restore Databases Using Restore by Session Method

To perform a restore operation using Restore by Session method

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 CA 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. Expand 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, Partial, Files-and-FileGroups, or Transaction log backup for each session on the media.
4. Right-click the session containing the backup session you want to restore and choose Backup Agent Options from the pop-up window.

The Backup Agent Restore Options dialog appears. This dialog varies depending on the backup method of the session selected, and the version of SQL Server from which the database was backed up.

5. Perform one of the following to select restore options:
 - Click OK to accept the default options and allow the Automatic Selection option select the appropriate restore sequence and 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 sequence and click OK. For more information about options, see [Restore Options](#).
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 the same version of Microsoft SQL Server or higher.
 - 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 clustered instance, you must select it from the Virtual Windows Server machine.

- 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 examples:

\\SERVER1\\MSSQLSERVER\\Lightning

\\SERVER2\\WEATHER\\Thunder

- **Note:** If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 or 2005 instance, you must explicitly select the target instance even if it is a default instance.

8. Click Start.

The Restore Media dialog appears.
9. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.
10. 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.
11. 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.
12. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
13. (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.
14. 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 *Administration Guide*.

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

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.

Restore to Alternate 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

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 window opens.

2. Choose Backup Agent Options.
The Backup Agent Restore Options dialog opens.
3. In the second folder tab, 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](#) (see page 79).
- [Restore by Session Using a Separate Job for Each Session](#) (see page 80).
- [Restore by Tree Using a Separate Job for Each Session](#) (see page 81).

Restore by Session Using a Single Restore Job

Use the Restore Manager in CA ARCserve Backup to restore databases by session using a single restore job.

To restore databases by session using a single restore job

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 CA 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 window. 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: You cannot restore any additional transaction logs 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

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 CA 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 window. 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 window. 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 Using a Separate Job for Each Session

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

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 *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 [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 logs using CA 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.

This section contains the following topics:

- [Backup and Restore Considerations in Microsoft SQL Server Cluster Environments](#) (see page 84)
- [Microsoft SQL Server 7.0 Cluster Environment Backups](#) (see page 84)
- [Restore by Tree in Microsoft SQL Server 7.0 Cluster Environments](#) (see page 87)
- [Restore by Session in Microsoft SQL Server 7.0 Cluster Environments](#) (see page 90)
- [Microsoft SQL Server 2000 or 2005 Cluster Environment Backups](#) (see page 93)
- [Restore by Tree in Microsoft SQL Server 2000 or 2005 Environments](#) (see page 95)
- [Restore by Session in Microsoft SQL Server 2000 or 2005 Environments](#) (see page 98)
- [Perform Disaster Recovery in a Microsoft SQL Server 7.0 Cluster Environment](#) (see page 101)
- [Perform Disaster Recovery in Microsoft SQL Server 2000 or 2005 Cluster Environments](#) (see page 101)

Backup and Restore Considerations in Microsoft SQL Server Cluster Environments

CA ARCserve Backup supports clustered Microsoft SQL Server in the Microsoft Clustering Server (MSCS) environment.

Use CA ARCserve Backup and the Agent for Microsoft SQL Server to back up and restore clustered Microsoft SQL Servers just like any nonclustered instance of Microsoft SQL Server, with the following important differences:

- You must install the Agent for Microsoft SQL Server on the local drives of every node in the cluster.
Note: For information about installing the Agent for Microsoft SQL Server, see [Install the Agent in a Standard Microsoft SQL Server Environment](#).
- If the node on which the clustered Microsoft SQL Server instance 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 Server 7.0 Cluster Environment Backups

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

Select Server, Protocol, Security, and Backup Type

To select the server, protocol, security, and backup type when backing up in a Microsoft SQL Server 7.0 cluster environment

1. Ensure that the Microsoft SQL Server instance is running in your Microsoft cluster environment. It does not need to be on the same node as the Microsoft Cluster Server Quorum Resources, though that is recommended.
2. Start CA ARCserve Backup and open the Backup Manager.
3. On the Source tab, browse the MSCS cluster name through the Windows Systems category. The Microsoft SQL Server instance is located under the name of 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 Server virtual server entry in Windows Systems.
4. Right-click the Microsoft SQL Server instance and select Transfer Mechanism from the pop-up window. The Transfer Mechanism dialog opens.

5. Select Named Pipes and click OK. This will allow the backup to work if the Windows Quorum and Microsoft SQL Server are running on different nodes of the cluster.
6. Right-click the Microsoft SQL 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 Server instance to display a list of databases, 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.
9. Right-click the database and select Backup Agent Options from the pop-up window. The Agent Backup Options dialog opens.
10. Select the type of backup you want to perform, and whether to backup the entire database or a subset. For more information about backup types, see Backup Methods.
11. If you chose Files-and-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 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

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 *Administration Guide*.

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

The Restore Media dialog appears.

4. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.

Note: In this dialog, the column and button labeled Agent refer to the Universal Agent, which hosts the Client Agent for Windows and the Agent for Microsoft SQL Server. You can edit agent connection information at this time. For more information about client agents, see the *Administration Guide*.

5. In the Security and Agent Information dialog, verify the user name and password for the Windows cluster where Microsoft SQL Server is running and for the Microsoft SQL Server instance. To enter or change security information for the computer or the Microsoft SQL Server instance, select the computer or the Microsoft SQL Server instance, 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.

6. Click OK. The Submit Job dialog opens.
7. (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.
8. 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 *Administration Guide*.

Restore by Tree in Microsoft SQL Server 7.0 Cluster Environments

To restore using the Restore by Tree method in a Microsoft SQL Server 7.0 cluster environment

1. On the Restore Manager Source tab, select Restore by Tree from the drop-down list.
2. In the navigation tree, expand the Windows Quorum virtual server 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 window. The Backup Agent Restore Options dialog opens. The dialog varies depending on the session selected.
5. Perform one of the following to select restore options:
 - Click OK to accept the default options and allow the Automatic Selection option select the appropriate restore sequence and 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 sequence and click OK. For more information about options, see Restore Options.

Important! The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the sessions which must be restored and applies the selected options appropriately to each session.

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 the same version of Microsoft SQL Server or higher.
 - 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 clustered instance, you must select it from the Virtual Windows Server machine.

- 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 examples:

\\SERVER1\\MSSQLSERVER\\Lightning

\\SERVER2\\WEATHER\\Thunder

Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 or 2005 instance, you must explicitly select the target instance even if it is a default instance.

8. Click Start.

The Restore Media dialog appears.

9. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.

10. 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.

11. 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.

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

13. (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.

14. 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 *Administration Guide*.

Restore by Session in Microsoft SQL Server 7.0 Cluster Environments

To perform a restore job in a Microsoft SQL Server 7.0 cluster environment using the Restore by Session method

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 CA 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 window. The Backup Agent Restore Options dialog opens. The dialog varies depending on the Restore sessions selected.
5. Perform one of the following to select restore options:
 - Click OK to accept the default options and allow the Automatic Selection option select the appropriate restore sequence and 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 sequence and click OK. For more information about options, see Restore Options.
6. **Important!** The Automatic Selection option on the Backup Agent Restore Options dialog automatically selects the sessions which must be restored and applies the selected options appropriately to each session.
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 the same version of Microsoft SQL Server or higher.
 - 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 clustered instance, you must select it from the Virtual Windows Server machine.

- 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 examples:

\\SERVER1\MSSQLSERVER\Lightning

\\SERVER2\WEATHER\Thunder

Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 or 2005 instance, you must explicitly select the target instance even if it is a default instance.

8. Click Start.

The Restore Media dialog appears.

9. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.

10. 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.

11. 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 a 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.

12. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
13. (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.
14. 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 *Administration Guide*.

Microsoft SQL Server 2000 or 2005 Cluster Environment Backups

The following sections provide the procedures to back up in a Microsoft SQL Server 2000 or 2005 cluster environments.

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

1. Verify that Microsoft SQL virtual server is running in your Microsoft cluster environment.
2. Start CA 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: It is recommended that you browse the clustered Microsoft SQL Server instances through their associated Microsoft SQL Server virtual server names only. Browsing from any other entry point may cause backup failures if the Windows virtual server or the Microsoft SQL Server virtual server moves to a different node of the cluster. It is recommended that you do not browse through the MSCS computer name for clustered Microsoft SQL Server 2000 and 2005 instances.

An instance of Microsoft SQL Server 2005 Express Edition which is used as the ARCserve Database of a clustered CA ARCserve Backup installation should be browsed through the virtual server name associated with the clustered CA ARCserve Backup primary server.

4. Right-click the Microsoft SQL Server instance and select Transfer Mechanism from the pop-up window. The Transfer Mechanism dialog opens.
5. Select Virtual Devices and click OK.
6. Right-click the Microsoft SQL Server instance and select Security. The Security dialog opens.
7. Verify the security information in the Security dialog and click OK.
8. Expand the Microsoft SQL Server instance to display a list of databases, and select a database.
9. Right-click the database and select Backup Agent Options from the pop-up window. The Agent Backup Options dialog opens.

10. Select the type of backup you want to perform, and any subset options you wish to use. For more information about backup types, see [Backup Methods](#).
11. If you chose the Files-and-FileGroups subset, 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

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 [Administration Guide](#).
2. Click the [Schedule](#) tab and select the scheduling options for this backup job. For information about scheduling backups, see the [Administration Guide](#).
3. Click [Start](#).
The [Restore Media](#) dialog appears.
4. Select the backup server where the restore job is going to run and click [OK](#).
The [Session User Name and Password](#) dialog opens.
Note: In this dialog, the column and button labeled Agent refer to the Universal Agent, which hosts the Client Agent for Windows and the Agent for Microsoft SQL Server. You can edit agent connection information at this time. For more information about client agents, see the [Administration Guide](#).

5. In the Security and Agent Information dialog, verify the user name and password for the Windows cluster where Microsoft SQL Server is running and for the Microsoft SQL Server instance. To enter or change security information for the computer or the Microsoft SQL Server instance, select the computer or the Microsoft SQL Server instance, 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.

6. Click OK. The Submit Job dialog opens.
7. (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.
8. 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 *Administration Guide*.

Restore by Tree in Microsoft SQL Server 2000 or 2005 Environments

To restore using the Restore by Tree method in a Microsoft SQL Server 2000 or 2005 cluster environment

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 Server virtual server 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 window. The Backup Agent Restore Options dialog opens. The dialog varies depending upon the restore sessions selected.

5. Perform one of the following to select restore options:
 - Click OK to accept the default options and allow the Automatic Selection option select the appropriate restore sequence and 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 sequence and click OK. For more information about options, see [Restore Options](#).
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 the same version of Microsoft SQL Server or higher.
 - 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 clustered instance, you must select it from the Virtual Windows Server machine.

- 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 examples:

\\SERVER1\\MSSQLSERVER\\Lightning

\\SERVER2\\WEATHER\\Thunder

Note: If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 or 2005 instance, you must explicitly select the target instance even if it is a default instance.

8. Click Start.

The Restore Media dialog appears.

9. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.

10. 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.

11. 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.

12. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
13. (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.
14. 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 *Administration Guide*.

Restore by Session in Microsoft SQL Server 2000 or 2005 Environments

To perform a restore using the Restore by Session method

1. On the Restore Manager Source tab, select Restore by Session from the drop-down list. A list of the media that you have used when backing up with CA 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 window. The Backup Agent Restore Options dialog opens.

5. Perform one of the following to select restore options:
 - Click OK to accept the default options and allow the Automatic Selection option select the appropriate restore sequence and 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 sequence and click OK. For more information about options, see [Restore Options](#).
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 the same version of Microsoft SQL Server or higher.
 - 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 clustered instance, you must select it from the Virtual Windows Server machine.

- 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 examples:

\\SERVER1\\MSSQLSERVER\\Lightning

\\SERVER2\\WEATHER\\Thunder

- **Note:** If you are restoring a database backed up from a Microsoft SQL Server 7.0 instance to a Microsoft SQL Server 2000 or 2005 instance, you must explicitly select the target instance even if it is a default instance.

8. Click Start.

The Restore Media dialog appears.

9. Select the backup server where the restore job is going to run and click OK.

The Session User Name and Password dialog opens.

10. 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.

11. 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.

12. Click OK on the Session User Name and Password dialog. The Submit Job dialog opens.
13. (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.
14. 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 *Administration 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

1. Reinstall Microsoft SQL Server but do not re-create the Microsoft SQL Server virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Set the CA ARCserve Backup SQL Agent services and the SQL Agent services to log in as the domain administrator.
4. Restore the Microsoft SQL Server [master] database.
5. Set up the Microsoft SQL Server virtual server and reinstall the Agent for Microsoft SQL Server with the new Microsoft SQL virtual server settings.
6. 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 or 2005 Cluster Environments

To perform a disaster recovery in a Microsoft SQL Server 2000 or 2005 cluster environment

1. Reinstall Microsoft SQL Server and re-create the Microsoft SQL Server virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.

3. Restore the Microsoft SQL Server [master] database.
4. 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 Option Guide*.

Appendix A: Troubleshooting and Disaster Recovery

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

This section contains the following topics:

- [Agent and CA ARCserve Backup Error Messages](#) (see page 103)
- [General Considerations for CA ARCserve Backup and the Agent](#) (see page 103)
- [Microsoft SQL Server Error Messages](#) (see page 106)
- [Replication of Microsoft SQL Server](#) (see page 108)
- [MS SQL Server Disaster Recovery](#) (see page 108)

Agent and CA ARCserve Backup Error Messages

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

General Considerations for CA ARCserve Backup and the Agent

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

- CA 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, CA 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 Valid Operations with SQL Server Database Options.
- Check the Agent for Microsoft SQL Server log file sqlpag.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.

No Icon in Browse Tree

No Microsoft SQL Server icon in the the backup source or restore destination tree.

Reason:

This can occur if the Agent for Microsoft SQL Server is not installed or the CA 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\CA ARCserve Backup\SQLPAAdp\Instances\dbasql70

or

SOFTWARE\ComputerAssociates\CA ARCserve Backup\SQLPAAdp\Instances
\dbasql@INSTANCENAME

Action:

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

E8535

Failed to receive data from the Client 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 ReceiveTimeout value in the following registry key:

`HKEY_LOCAL_MACHINE\SOFTWARE\CA\ARCserve Backup\Base\Task\Remote`

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

Note: This setting is shared for all agents which operate under the Universal Agent service.

AE50009 or AE50010

Unable to sign in to Windows as specified user.

Unable to sign in to Microsoft SQL Server as specified user.

Reason:

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

- Machine authentication failure may have occurred.
- The Windows user specified may not have the required Backup Operator privilege.
- Database authentication failure may have occurred.
- The target database instance may be stopped or inaccessible.

The specific error code and message from either Windows or SQL Server will be appended to the message.

Action:

To address this error

1. Verify that the CA Backup Agent Universal Agent service is running on the target server.
2. Verify that the target database server instance is running on the target server. If it is not, restart it.
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.

Microsoft SQL Server Error Messages

Microsoft SQL Server errors can occur for a variety of reasons and can appear in several different CA 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](#).

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*.

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 a Microsoft SQL Server Disaster Recovery Elements session or 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](#).

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. Restore the [msdb] database in multi-user mode 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

A typical disaster recovery scenario consists of the following steps:

1. Reinstall Windows, if necessary.
2. Reinstall CA ARCserve Backup, if necessary.
3. Reinstall the Agent for Microsoft SQL Server and the Client Agent for Windows, if necessary. (The Client Agent is needed to restore Microsoft SQL Server Disaster Recovery Elements.)
4. Perform one of the following steps as appropriate:
 - If you have a Microsoft SQL Server Disaster Recovery Elements session, restore it.
 - If an offline backup exists, restore it.
 - If you do not have an offline backup or a Disaster Recovery Elements session, 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 or Disaster Recovery Elements backup do not exist and you do not have the Microsoft SQL rebuildm.exe utility, reinstall the Microsoft SQL Server or MSDE-based application.
5. Restore the [master] database.
6. Restart Microsoft SQL Server in normal, multi-user mode.
7. Restore the [msdb] database.

8. Restore the MSDB database.
9. Restore all other databases and transaction logs, except the replication database.
10. 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

1. Reinstall Microsoft SQL Server but do not re-create the Microsoft SQL Server virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Set the CA ARCserve Backup SQL Agent services and the SQL Agent services to log in as the domain administrator.
4. Restore the Microsoft SQL Server [master] database.
5. Set up the Microsoft SQL Server virtual server and reinstall the Agent for Microsoft SQL Server with the new Microsoft SQL virtual server settings.
6. 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 or 2005 Cluster Environments

To perform a disaster recovery in a Microsoft SQL Server 2000 or 2005 cluster environment

1. Reinstall Microsoft SQL Server and re-create the Microsoft SQL Server virtual server.
2. Reinstall the Agent for Microsoft SQL Server, if necessary.
3. Restore the Microsoft SQL Server [master] database.
4. 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 Option Guide*.

Appendix B: Configuring Microsoft SQL Server Security Settings

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

This section contains the following topics:

- [Types of Microsoft SQL Server Authentication](#) (see page 113)
- [Authentication Requirements](#) (see page 113)
- [How User Authentication Is Changed](#) (see page 114)

Types of Microsoft SQL Server 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 exists for this user in Microsoft SQL Server and does not have this role.

How User Authentication Is Changed

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

To change user authentication and update the settings for both Windows and CA 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 existing backup jobs in CA 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 for SQL Server 7.0 or 2000

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 list. 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.

To check or change the Microsoft SQL Server authentication method for SQL Server 2005

1. On the system where Microsoft SQL Server is running, open the SQL Server Management Studio.
2. Connect to the instance of Microsoft SQL Server that you want to change.
3. In the Object Explorer pane, right-click the instance and select Properties from the drop-down list. The Properties dialog opens.
4. In the Properties dialog, choose the Security page.
5. Under Server Authentication, select either SQL Server and Windows Authentication Mode to enable Microsoft SQL server-based authentication, or Windows Authentication Mode to enable only Windows users.
6. Click OK.

Update the Agent Account Configuration

To update the Agent for Microsoft SQL Server Account Configuration

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 authentication:
 - If you have set Microsoft SQL Server for Windows Only, select Windows authentication.
 - If you have selected Mixed Mode and want to be able to back up Microsoft SQL Server using only Microsoft SQL Server 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 Server users, select Windows authentication.
4. If you have selected Microsoft SQL authentication, enter the user name and password for a Microsoft SQL Server 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

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

1. On the system where you have installed CA ARCserve Backup, start CA ARCserve Backup and open the Job Status Manager.
2. Select the first job you need to update, right-click and select Modify from the pop-up window. The Backup Manager will open and load the job.
3. Select the Source tab.
4. 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 *Implementation Guide* for instructions on how to add it.
5. Right-click the database and choose Security from the pop-up window. The Security dialog opens.
6. 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.
7. 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.
8. Select an applicable backup job and click Modify on the toolbar. The Backup Manager opens.
9. Expand the server, right-click this job, repeat the Security dialog procedures, and click Start to resubmit the backup job.
10. 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

A

access requirements • 13
Account Configuration dialog • 115
agent activity log • 14
alternate disk location, restoring to • 78
applying a transaction log • 49
architectural diagram • 10
architectural overview • 10
authentication
 changing the method • 114
 SQL • 113
 types • 113
 updating setting for • 114
automatic selection • 56
 in a differential backup restore • 51
 in transaction log restore • 51
avoiding restore problems • 109

B

backup
 actions requiring a full backup after • 32
 Backup Agent Options dialog • 39
 Backup Manager • 39
 configuring parameters • 23
 data flow steps with agent • 13
 Destination tab • 39
 differential • 33
 dynamic job packaging • 43
 explicit job packaging • 43
 in a Microsoft SQL Server Cluster
 Environment, restore • 83
 in a Microsoft SQL Virtual Server 2000
 environment • 92
 in a Microsoft SQL Virtual Server 7.0
 environment • 84
 named pipes • 39
 options • 30
 overview of • 27
 partial wildcards for destinations • 39, 85
 procedure • 39
 process • 11
 Remote Protocol option • 39
 SAP R/3 • 16
 scheduling • 41

strategy recommendations • 31
TCP/IP • 39
transaction log • 34
transaction logs, back up separately • 27
types • 30
Backup Agent Backup Options dialog • 39
Backup Agent Restore Options dialog, backup
 agent restore • 71
backup and restore parameters, configuring •
 23
Backup and restore options • 15
Backup Manager
 updating • 117
 used in backup • 39

C

CA ARCserve Backup Agent for Microsoft SQL
 Server
 error • 103
 troubleshooting • 103
cluster environment
 backup, Microsoft SQL Server 2000 • 92
 backup, Microsoft SQL Server 7.0 • 84
 differences • 83
 disaster recovery • 101
 installing agent on all nodes • 83
 restore by session, Microsoft SQL Server 2 •
 98
 restore by session, Microsoft SQL Server 7 •
 89
 restore by tree, Microsoft SQL Server 2000
 • 94
 restore by tree, Microsoft SQL Server 7.0 •
 86
complete backup
 defined • 30
 restoring • 49
configuring the TCP port address
 post • 23
CREATE INDEX statement, impact on backups
 • 36

D

Database
 locking • 47

restoring lost or damaged • 47
Database Consistency Check option explained • 30
Database option • 56
dbasql.log • 14, 30
dbasql60.log • 14
destination database • 47
differential backup • 33
 advantages • 33
 defined • 33
 file and file group • 30
 restoring • 49, 51
 supplemental to full backups • 33
 type • 30
 when to perform • 33
Disaster • 101
disaster recovery
 in a Microsoft SQL Server 7.0 Cluster Envir • 101
 in a standard MS SQL Environment • 110
 in a standard MS SQL Server Environment • 110
 in cluster environment • 101
 scenario • 110
disaster recovery Microsoft SQL Server • 108
display backup sessions of selected database • 78
dump • 11
dynamic job packaging • 43

E

error messages • 103
explicit job packaging • 43

F

failure during restore • 47
features, additional in Microsoft SQL Server 2000 • 14
file and file group
 applying log sessions to restore • 58
 backup type • 30
 restore requirements • 58
File or Filegroups option • 56

H

history, version • 71, 86, 94

I

installation prerequisites • 19

installing the agent
 in a Microsoft SQL Server 7.0 cluster env • 20
 Microsoft SQL Server 2000 cluster environ • 21
installing the agent in a Microsoft Server 2000 cluster environ • 22
installing the agent in a Microsoft SQL Server 7.0 cluster envi • 22
installing the agent in a standard Microsoft SQL Server environ • 21

J

job packaging • 43

K

Keep Replication Settings option • 63

L

load • 47
loading transaction logs sequentially • 49
Log Point in Time option
 explained • 59
 limitations • 59
 options • 59

M

master database
 requirements to restore • 109
 restoring in single-user mode • 53
 restoring in Windows 2000 • 53
 restoring in Windows 2003 • 53
 restoring in Windows XP • 53
media failure • 47
Microsoft SQL Account Configuration utility • 115
Microsoft SQL Agent Configuration utility • 23
Microsoft SQL authentication • 113
Microsoft SQL Server
 disaster recovery • 108
 starting in single-user mode • 53
Microsoft SQL Server 2000 additional features available • 14
Microsoft SQL Server 2000 cluster environment
 Account Configur • 22
Microsoft SQL Server 7.0 cluster environment
 Account Configura • 22
 installing the ag • 20
 mixed mode authen • 20

Microsoft SQL Virtual Server 2000 environment
 backup • 92
 restore by session • 98
 restore by tree • 94
Microsoft SQL Virtual Server 7.0 environment
 backup • 84
 restore by session • 89
 restore by tree • 86
multiple instance support • 14
multistriping support
 with rotation schemes • 41

O

ODBC settings, checking and changing • 116

P

Partial Restore option • 56
password changes • 71, 75, 86, 89, 94, 98
post-installation procedures
 configuring backup and restore paths • 23
 installing • 23
preserving replication settings • 78

R

reinitializing unused pages, in reloading • 47
remote Microsoft SQL server backup • 39
Remote Protocol option • 39
remote Windows database servers, accessing • 13
replication
 preserving settings • 78
restore
 automatic selection • 56
 automatic selection • 56
 automatic selection option • 56
 automatic selection, automatic selection • 56
 avoiding problems • 109
 by session using separate restore job for each session • 80
 by session using single restore job for each session • 79
 by tree using a separate restore job for each session • 81
 change password • 71
 configuring parameters • 23
 destination database, requirements • 47
 differential backup • 51

display backup sessions of selected database • 78
failure during • 47
file groups with indexes • 58
Keep Replication Settings option • 63
Log Point in Time option • 59
master database • 53
options • 55
overview • 47
partial • 56
reinitializing unused pages • 47
restore by session • 71
restore by session in a Microsoft SQL Virtual Server 7 • 89
restore by session in Microsoft SQL Virtual Server 2000 • 98
restore by tree • 71
restore by tree in a Microsoft SQL Virtual Server 2000 • 94
Restore Database Files As option • 68
Restore files to their original locations option • 71, 75, 86, 89, 94, 98
restore type option • 56
Restricted User Access After Restore option • 63
security • 71, 75, 86, 89, 94, 98
SQL Agent Filter option • 78
to alternate disk location • 78
transaction log • 51
types • 49
types of backup sessions • 71
uncommitted transactions • 47
using restore by session • 75
using restore by tree • 71, 81, 86, 94
Version History dialog • 71
restore by session
 explanation • 71
 procedure • 75
restore by tree • 71
 explanation • 71
 procedure • 71
Restore Database Files As option • 68
Restore Files to Their Original Locations option • 71, 75, 86, 89, 94, 98
restore types • 49
Restricted User Access after Restore option • 63
rotation schemes, global options • 41

S

SAP R/3, backing up with SQL Server • 16
scheduling backups • 39, 41
security
 information • 71, 75, 86, 89, 94, 98
 specifying during restore • 71
Security and Agent Information dialog • 39
SQL Agent Filter option • 78
SQL Server DBAgent Configuration dialog • 23
standard Microsoft SQL Server environment,
 Account Configuration • 21
system administrator privileges • 113

T

transaction log
 applying • 49
 database activity records • 34
 loading sequentially • 49
 minimizing number of, with differential
 backup • 33
 restoring • 49, 51
transaction log backup
 advantages • 34
 defined • 34
 files to restore before • 35
 full database backup required before • 35
 options • 30
 run separately • 34
 truncate • 35
truncating transaction log backups • 35
types of backups • 30

U

user account specifications • 113

V

version history • 71, 86, 94