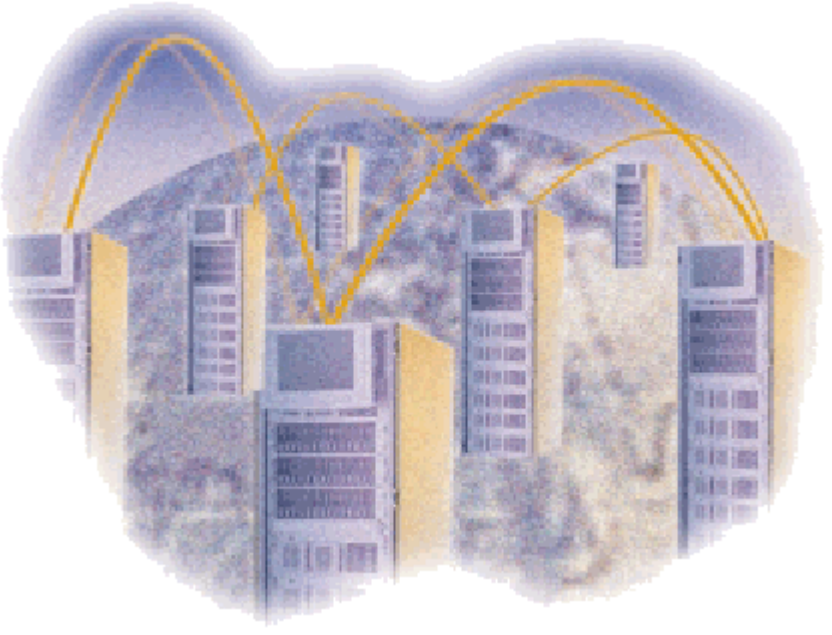
High Availability for Microsoft SQL Server Using Double-Take 4.x



High Availability for Microsoft SQL Server Using Double-Take 4.x published January 2003

Double-Take Support for Application Failover

Double-Take's file system replication process is application independent and replicates any file system changes (including permissions and attributes) written to NTFS, FAT or FAT32 file systems by any application or process, subject to specific exceptions called out in the *User's Guide* or readme.txt file. Maintaining point-in-time consistent file system replicas and providing server monitoring and automatic or manual failover of the server name and IP address are the primary functions of the Double-Take software and we offer support to qualified customers should these functions fail to operate in accordance with our published documentation, regardless of what application or process is manipulating the data.

NSI Software may provide application notes and other documents that provide implementation guidelines on how to use Double-Take functions and replicas to manually or automatically failover or recover many popular third party applications and a general process to accomplish failover or recovery of many other third party applications. While these steps are believed to be accurate for the specific configuration, Double-Take version, and application versions originally tested, due to the number of possible configurations and variables, NSI Software can only test selected combinations and may provide only limited support for the operation and configuration of third party applications or the behavior of those applications before, during, or after failover, in its discretion. In cases where NSI Software has no direct access to or experience with a particular application or configuration, NSI Software support may also be limited to only the actual replication of the file system data and failover (name and IP address) of the server.

For assistance in validating, implementing or troubleshooting these or other possible configurations with third party applications, NSI Software and its partners may offer professional services on a fee basis to apply best practices for assisting with third party applications to recover automatically or manually using replicated data.

This, and any other, application note is provided solely for the convenience of our customers and is not intended to bind NSI Software to any obligation.

Table of Contents

Introduction

Microsoft SQL Server is a scalable, reliable, flexible, and high-performance relational database management system for Microsoft Windows 2000/NT server-based systems. NSI Software's Double-Take provides real-time enterprise data protection and replication. Double-Take can be used to provide high availability for your SQL server.

This document describes the steps necessary to configure Double-Take to provide high availability for Windows 2000/NT servers running Microsoft SQL Server versions 6.5 or 7.0. These procedures allow a secondary server to assume the identity and role of a failed SQL server while maintaining the availability of SQL services with minimal disruption or data loss.

To complete these instructions, you will install Microsoft SQL Server and Double-Take, and configure Double-Take for replication and failover. Due to the complexities of these applications, this document is intended for network administrators with experience installing, configuring, and maintaining network applications including Double-Take and Microsoft SQL Server.

NOTE: Double-Take allows you to configure one target to monitor and failover for one or more source machines. In a one-to-one configuration, you will want to replicate your SQL data to the same location on the target so that failover is automatic. In a many-to-one configuration, each SQL data store will need to be replicated to a unique location and then renamed to the corresponding SQL directory on the source before failover occurs.

This application note focuses on a single SQL server being replicated to a single target.

Requirements

- Two servers that meet one of the following operating system requirements:
  - Microsoft Windows NT 4.0 with Service Pack 4 or higher
  - Microsoft Windows 2000

  NOTE: The two servers should both be running the same operating system. Although cross-platform mirroring and replication are available, NSI Software recommends that the two servers be the same platform for effective failover and failback.
- It is recommended that both source and target servers be standalone servers. (You may experience problems with promotion and demotion during failover if either of the machines are Primary or Backup Domain Controllers.)
- Both servers must be connected to the same physical network
- Two licensed copies of Microsoft SQL Server 6.5 or 7.0
- Two licensed copies of Double-Take 4.x

Protecting Your SQL Data

Install Software on the Source

1. Install Microsoft SQL Server version 6.5 or 7.0 on the source, if it is not already installed.

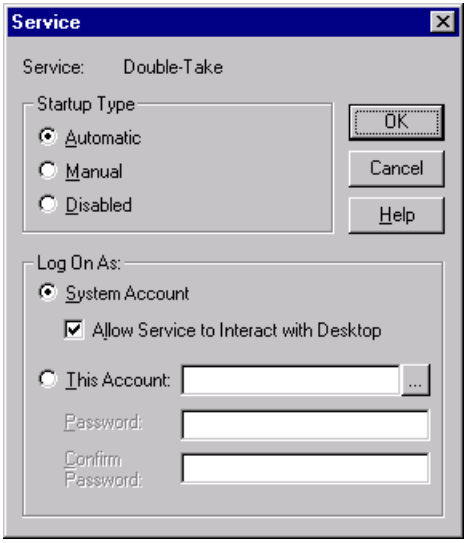
2. Record the drive and directory where Microsoft SQL is installed. For example, the default directories for SQL 6.5 and 7.0 are `<drive>:\MSSQL` and `<drive>:\MSSQL7`, respectively.

   SQL Installation Drive and Directory: _____

3. Install Double-Take 4.x on the source machine using the installation defaults. See the Double-Take *Getting Started* guide for details.

Install and Configure Software on the Target

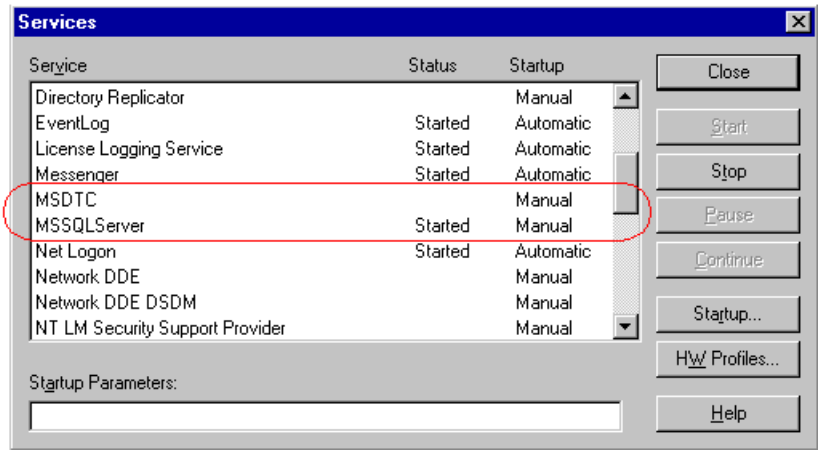
1. Install Double-Take 4.x on the target using the installation defaults. See the Double-Take *Getting Started* guide for details.
2. In Control Panel, Services, double-click the Double-Take service.
3. Mark the check box **Allow Service to Interact with Desktop** and click **OK**.



4. Install Microsoft SQL version 6.5 or 7.0 on the target using the same drive and directory specifications recorded in step 2 of the previous section.

5. Using the table below, identify the services that you will need to set to manual startup so that all SQL files are closed on the target and the Double-Take source can replicate the changes.

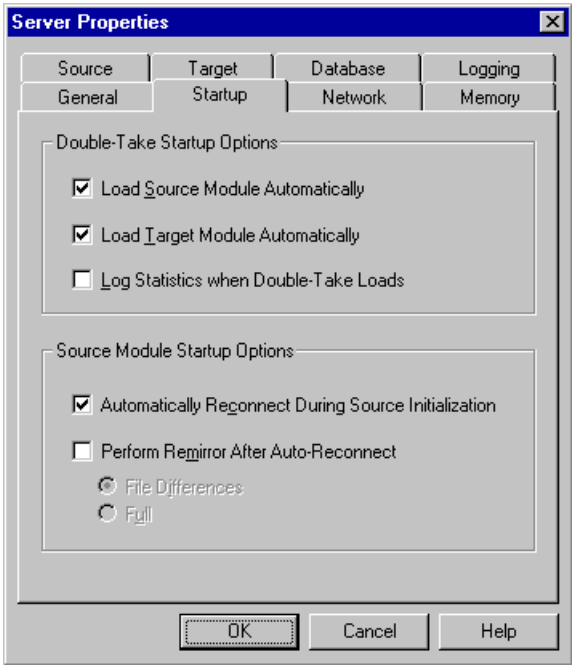
	SQL 6.5	SQL 7.0
Windows 2000	◆ Distributed Transaction Coordinator ◆ Message Queuing ◆ MSSQLServer ◆ SQLExecutive	◆ Distributed Transaction Coordinator ◆ Message Queuing ◆ MSSQLServer ◆ SQLServerAgent
Windows NT 4.0	◆ MSDTC ◆ Message Queuing ◆ MSSQLServer ◆ SQLExecutive	◆ MSDTC ◆ Message Queuing ◆ MSSQLServer ◆ SQLServerAgent



NOTE: If a failure should occur, the failover and failback scripts that you will be creating will control the stopping and starting of the SQL services.

Configure and Begin Mirroring and Replication

1. Select **Start**, **Programs**, **Double-Take**, **Management Console**.
2. Double-click your source machine to log on.
3. If you are using Double-Take version 4.1 or earlier, you will need to disable auto-remirror on auto-reconnect so that the source does not remirror files after failback. In version 4.2 and later, the source automatically recognizes that a restore is required and will not remirror. If you are using 4.1 or earlier, complete steps a-c below. If you are using 4.2, you can continue with the next numbered step.
   a. Right-click the source machine and select **Properties**.
   b. Select the Startup tab.



   c. By default, **Perform Remirror After Auto-Reconnect** will be selected. Disable this option so that the source does not remirror files after failback. Click **OK** to continue.

NOTE:	If you disable this option and a auto-disconnect occurs, you will need to remirror manually after the connection is reestablished.

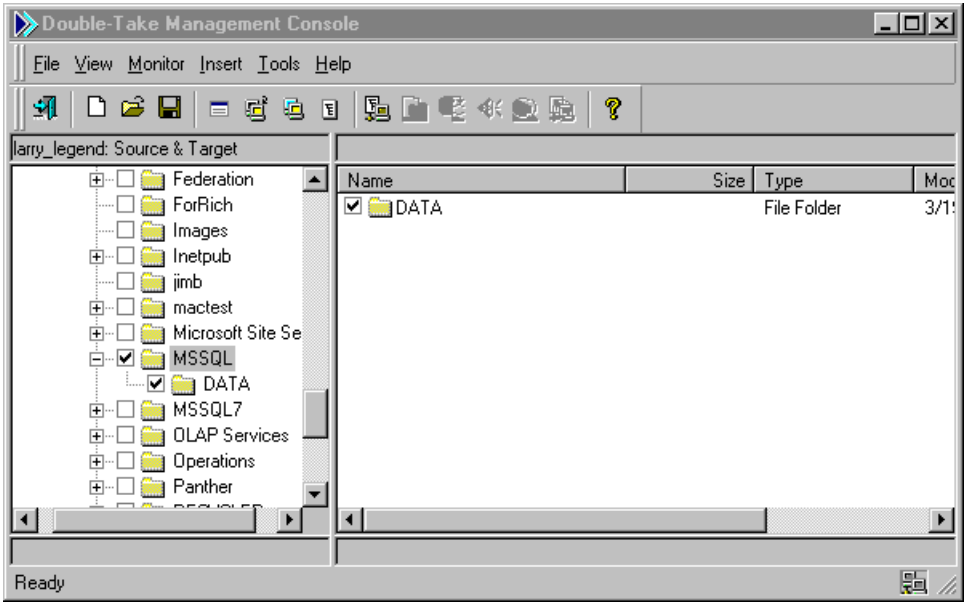
4. Right-click your source machine and select **New**, **Replication Set** and enter the desired name for the replication set.

5. Select the SQL data you wish to protect. Most likely, this will only include the SQL data and log files. It is not necessary to replicate the application files since they are already exist on the target machine.

   - Select the following directories:

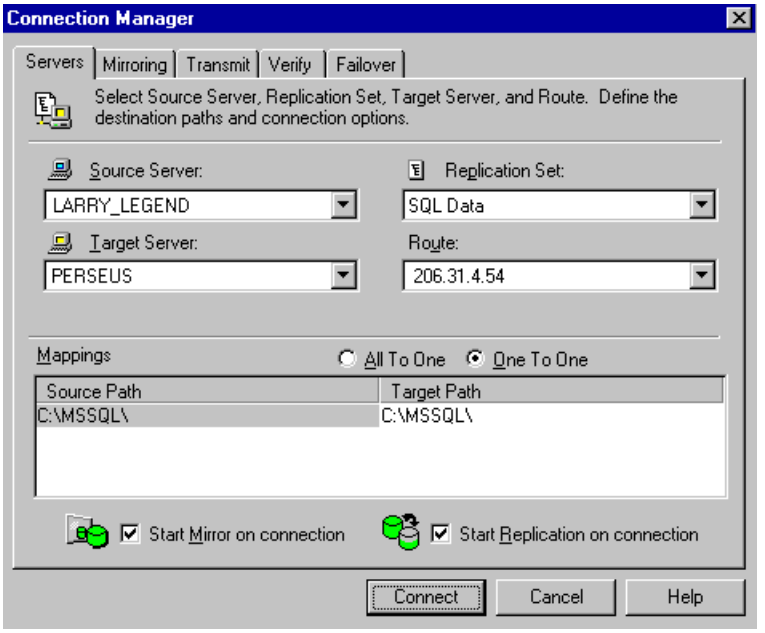
     ```
     <drive>:\<SQL Install Directory>\LOG
     <drive>:\<SQL Install Directory>\Data
     ```

   - Select any other directories (even if on different drives) that you may have created to store SQL data files.



6. Right-click the replication set name and select **Save** to save the replication set.

7. Drag and drop the replication set onto the target. The Connection Manager dialog box opens.



8. The **Source Server**, **Target Server**, **Replication Set**, and **Route** fields will automatically be populated. If you have multiple IP addresses on your target, verify the **Route** field is set to the correct network path. (For detailed information on connecting a source and target, reference Double-Take's *User's Guide*.)

9. Select the **One To One** mapping so that the replication set data is transmitted to the same directory structure on the target.

10. Click **Connect** to start the mirror and replication processes.

Configure Failover and Begin Failure Monitoring

1. If a failure occurs, you will want to have the SQL services start on the target machine automatically. To do this, create a batch file called `postover.bat` using the sample batch file below. Save the batch file to the same directory where your Double-Take files are installed.

POSTOVER.BAT

```
rem  This file is configured by default to run with SQL 7.0 on Windows 2000
rem  You may need to comment the default lines out and uncomment the set of
rem  lines for the SQL and Windows version you are using

rem If you are using SQL 6.5 and Windows 2000 use the following four commands.
rem net start "Distributed Transaction Coordinator"
rem net start "Message Queuing"
rem net start "MSSQLServer"
rem net start "SQLExecutive"

rem If you are using SQL 6.5 and Windows NT 4.0 use the following four commands.
rem net start "MSDTC"
rem net start "Message Queuing"
rem net start "MSSQLServer"
rem net start "SQLExecutive"

rem If you are using SQL 7.0 and Windows 2000 use the following four commands.
net start "Distributed Transaction Coordinator"
net start "Message Queuing"
net start "MSSQLServer"
net start "SQLServerAgent"

rem If you are using SQL 7.0 and Windows NT 4.0 use the following four commands.
rem net start "MSDTC"
rem net start "Message Queuing"
rem net start "MSSQLServer"
rem net start "SQLServerAgent"
```

2. After a failure is resolved, you will be ready to bring your source back online. At this time, you will want to stop the SQL services on the target automatically. To do this, create a batch file called `preback.bat` using the sample batch file below. Save the batch file to the same directory where your Double-Take files are installed.

PREBACK.BAT

```
rem  This file is configured by default to run with SQL 7.0 on Windows 2000
rem  You may need to comment the default lines out and uncomment the set of
rem  lines for the SQL and Windows version you are using

rem If you are using SQL 6.5 and Windows 2000 use the following four commands.
rem net stop "Distributed Transaction Coordinator"
rem net stop "Message Queuing"
rem net stop "MSSQLServer"
rem net stop "SQLExecutive"

rem If you are using SQL 6.5 and Windows NT 4.0 use the following four commands.
rem net stop "MSDTC"
rem net stop "Message Queuing"
rem net stop "MSSQLServer"
rem net stop "SQLExecutive"

rem If you are using SQL 7.0 and Windows 2000 use the following four commands.
net stop "Distributed Transaction Coordinator"
net stop "Message Queuing"
net stop "SQLServerAgent"
net stop "MSSQLServer" /y

rem If you are using SQL 7.0 and Windows NT 4.0 use the following four commands.
rem net stop "MSDTC"
rem net stop "Message Queuing"
rem net stop "SQLServerAgent"
rem net stop "MSSQLServer" /y
```

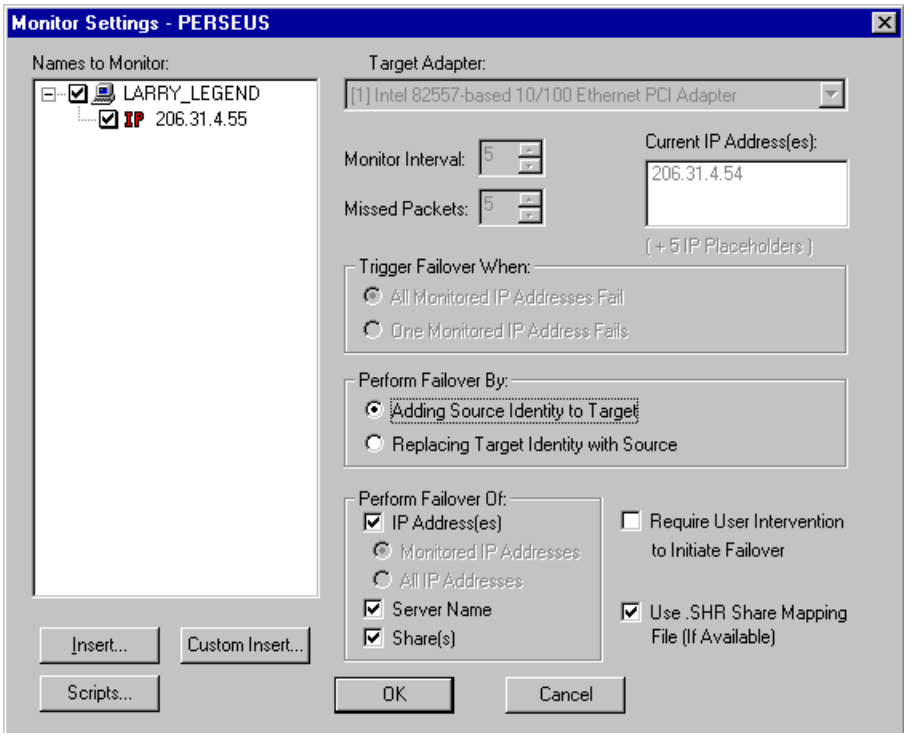
NOTE: These sample batch files are available on the NSI Software web site at `www.nsisoftware.com/download/sqlscrp.exe`.

3. Select **Start**, **Programs**, **Double-Take**, **Failover Control Center**.

4. Select the target machine from the list of available machines. If the target you need is not displayed, click **Add Target**, enter the machine name, and click **OK**.

5. To add a monitor for the selected target, click **Add Monitor**. Type the name of the source machine and click **OK**. The Monitor Settings window will open.

6. In the Monitor Settings window, mark the IP address that is going to failover and verify that **Adding Source Identity to Target** is selected.



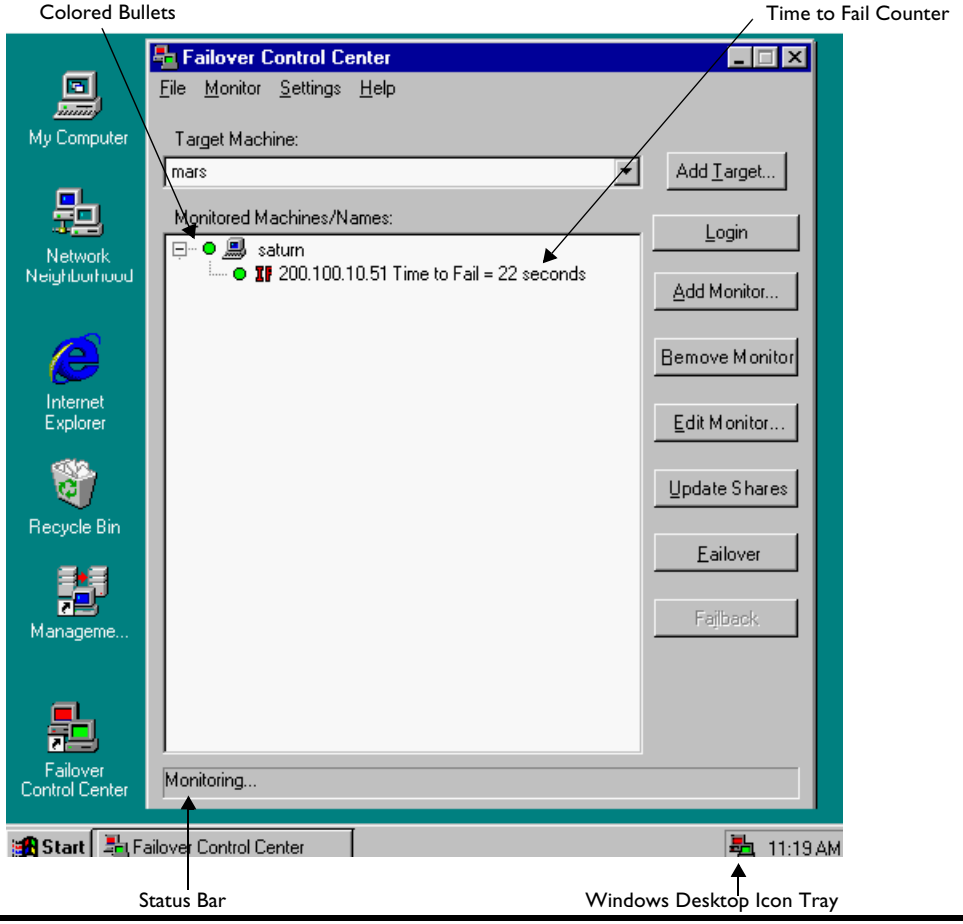
7. Click **Scripts** and specify the scripts that were created in steps 1 and 2 on page 8 and page 9.
8. Click **OK** to go back to the Monitor Settings dialog box.
9. Click **OK** to begin monitoring the source machine.

In the event of a source machine failure, your target machine is now ready to stand in for your source.

Monitoring Failover

Now that replication and failover monitoring are configured and started, you will need to know if and when there is a problem. Since it can be essential to quickly know the status of your machines, Double-Take offers various methods for monitoring the status of failover. When the Failover Control Center is running, you will see four visual indicators:

- The Failover Control Center Time to Fail counter
- The Failover Control Center status bar located at the bottom of the window
- The Failover Control Center colored bullets to the left of each IP address and source machine
- The Windows desktop icon tray containing a failover icon



NOTE: You can minimize the Failover Control Center and, although it will not appear in your Windows taskbar, it will still be active and the failover icon will still appear in the desktop icon tray.

The Failover Control Center does not have to be running for failover to occur.

The following table identifies how the visual indicators change as the status of failover changes.

	Time to Fail Countdown	Status Bar	Colored Bullets	Desktop Icon Tray
Source is Online	The Time to Fail counter is counting down and resetting each time a heartbeat is received from the source machine.	The status bar indicates that the target machine is monitoring the source machine.	The bullets are green. [a]	The Windows desktop icon tray contains a failover icon with red and green computers.
Source Fails and Failover is Initiated	The Time to Fail countdown value is 0.	The status bar displays the source machine and IP address currently being assumed by the target.	The bullets are red.	The Windows desktop icon tray contains a failover icon with red and green computers.
Failover is Complete	The Time to Fail counter is replaced with the **"Failed Over"** message.	The status bar indicates that monitoring has continued.	The bullets are red.	The Windows desktop icon tray contains a failover icon with a red computer.

a. When the **Time to Fail** value has decreased by 25% of the entire timeout period, the bullet changes from green to yellow, indicating that the target has not received a response from the source. The yellow bullet is a caution signal. If a response from the source is received, the countdown resets and the bullets change back to green. If the countown reaches zero without the target receiving a response from the source, failover begins.

Once failover is complete, any clients logging into the SQL server will be automatically directed to the target. Depending on the type of client software being used, SQL clients with an exsiting connection may have to be restarted in order to establish a connection with the target.

NOTE: For additional detailed information on failover and other monitoring tools, see the Double-Take *User's Guide*.

Restoring Your SQL Data

If your source experiences a failure, such as a power, network, or disk failure, your target machine will stand in for the source while you resolve the source machine issues. During the source machine downtime, data is updated on the target machine. When your source machine is ready to come back online, the data is no longer current and must be updated with the new data on the target machine.

1. Verify that your source machine is not connected to the network. If it is, disconnect it.
2. Resolve the source machine problem that caused the failure.

NOTE:	If you must rebuild your hard drive, continue with step 3. If you do not need to rebuild your hard drive, continue with step 8.

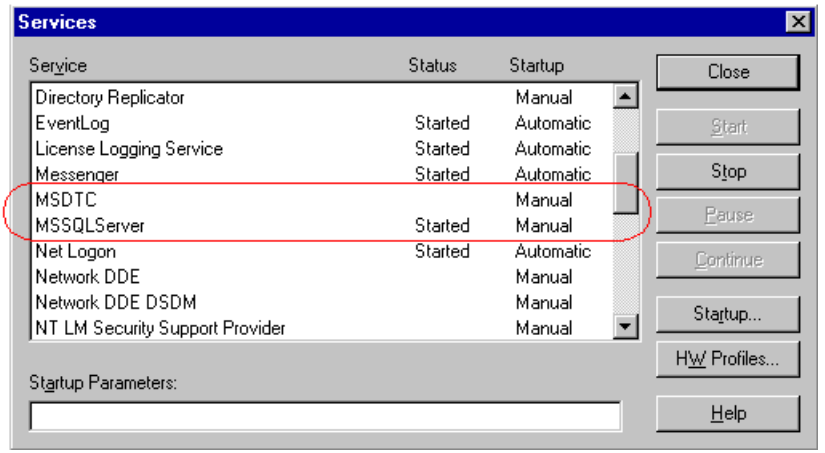
3. Install Windows 2000/NT. Since your source machine is not connected to the network, go ahead and use the source's original name and IP address.
4. Install Double-Take 4.x using the installation defaults.

NOTE:	Verify that the **transacational applications** option is selected on the Double-Take Optimizations screen since Microsoft SQL Server is a transactional database application. See the Double-Take guide *Getting Started* for further details.

5. Install SQL using the same drive and directory settings recorded in step 2 of the first section.

6. Using the table below, identify the services that you will need to set to manual startup so that all SQL files are closed on the target and the Double-Take source can replicate the changes.

	SQL 6.5	SQL 7.0
Windows 2000	◆ Distributed Transaction Coordinator ◆ Message Queuing ◆ MSSQLServer ◆ SQLExecutive	◆ Distributed Transaction Coordinator ◆ Message Queuing ◆ MSSQLServer ◆ SQLServerAgent
Windows NT 4.0	◆ MSDTC ◆ Message Queuing ◆ MSSQLServer ◆ SQLExecutive	◆ MSDTC ◆ Message Queuing ◆ MSSQLServer ◆ SQLServerAgent



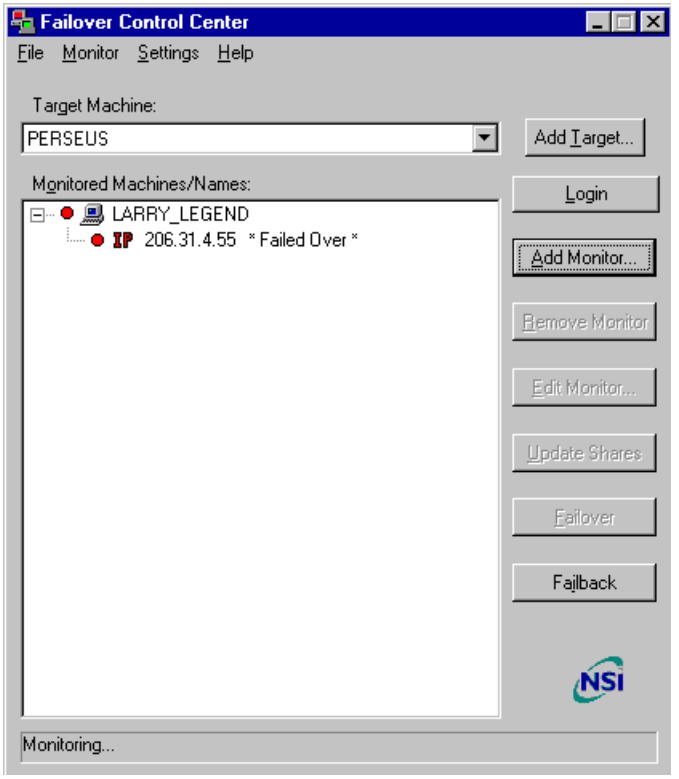
7. Rename any log files located in SQL log directory. By default, this is MSSQL\Log for SQL 6.5 and MSSQL7\Log for SQL 7.0.

8. **Verify that SQL is not running on the source.** The SQL services must not be running at this time. Depending on the type of failure, your services may be set to manual startup but could still be running. **Stop your SQL services and set them to manual.**

9. Select **Start**, **Programs**, **Double-Take**, **Failover Control Center**.

10. Select the target machine that is currently standing in for the failed source.

11. Select the failed source and click **Failback**.



The pre-failback script entered during the failover configuration stops the SQL services on the target so that no additional changes can be made.

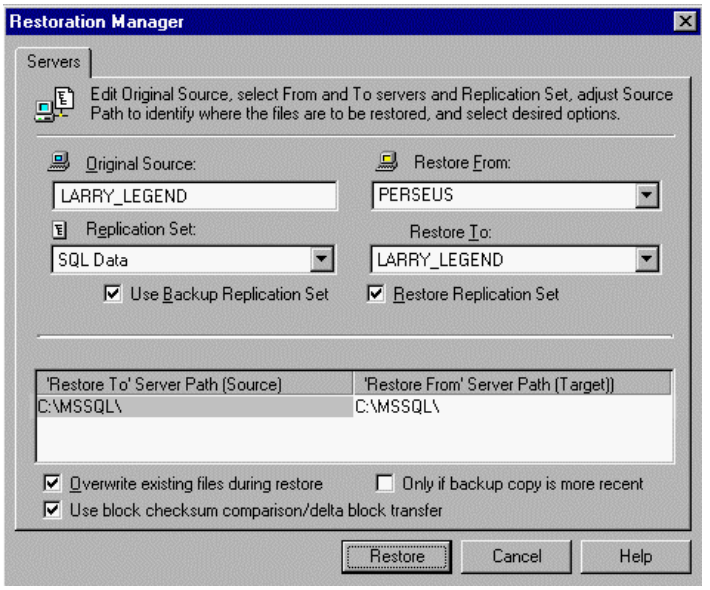
12. You will be prompted to determine if you want to continue monitoring the source server. Do not choose **Continue** or **Stop** at this time.

13. Connect the source machine to the network.

14. After the source is back online, select whether or not you want to continue monitoring this source machine (**Continue** or **Stop**).

15. To begin the restoration process, open the Double-Take Management Console and select **Tools**, **Restoration Manager**.

NOTE: You can also run the Double-Take DTCL automated restoration script, which can be found in the Double-Take *User's Guide*, to complete the remaining steps in this section.



16. Complete the appropriate fields as described below.
    - **Original Source**—The name of the source machine where the data original resided.
    - **Restore From**—The name of the target machine that contains the replicated data.
    - **Replication Set**—The name of the replication set to be restored.
    - **Restore To**—The name of the machine where you the data will be restored. This may or may not be the same as the original source machine.
17. Identify the correct drive mappings for the data and any other restoration options necessary. For detailed information on the restoration options, see Double-Take's *User's Guide*.
18. Verify that the selections you have made are correct and click **Restore**. The restoration procedure time will vary depending on the amount of data that you have to restore.
19. After the restoration is complete, start the SQL services on the source machine.
20. Reestablish the Double-Take SQL replication set connection.

At this time, your data is restored back to your source machine, the source machine is again the primary SQL server, and, if you selected to continue failover monitoring, the target is available to stand in for the source in the event of a failure.