d·i·g·i·t·a·l ™

DIGITAL Clusters for Windows NT

Administrator's Guide Addendum

Part Number: AV−R171B−TE

May 1997

This addendum provides updated information on installing and configuring Microsoft®
SQL Server™ with DIGITAL Clusters for Windows NT. Also included is new
information on configuring IP address failover for file shares, supported in DIGITAL
Clusters for Windows NT Version 1.1 with Service Pack 1.

Table of Contents

3 Configuring Database Software for Failover

12 IP Address Failover for File Shares

About This Addendum

This addendum supplies updated information on installing and configuring Microsoft SQL Server™ with DIGITAL Clusters for Windows NT Version 1.1 with Service Pack 1, and DIGITAL Clusters for Windows NT Version 1.0 with Service Pack 3.

Also included is new information on configuring IP address failover for file shares, supported in DIGITAL Clusters for Windows NT Version 1.1 with Service Pack 1.

Audience

This addendum is for system administrators who will manage the DIGITAL Clusters software. The addendum assumes that you are familiar with the tools and methodologies needed to maintain your hardware, operating system, and network.

Organization

This addendum consists of two chapters, as follows:

Chapter 3   Presents the database software installation and configuration steps you must complete before using Cluster Administrator to configure for failover of Microsoft SQL Server and Oracle7 Server.

                       The information supercedes Chapter 3, Configuring Database Software for Failover, of the *DIGITAL Clusters for Windows NT Administrator's Guide*, Second Edition.

Chapter 12   Discusses IP address failover for file shares and outlines the configuration steps you must perform to take advantage of this feature.

                       This information is a supplemental chapter to the *DIGITAL Clusters for Windows NT Administrator's Guide*, Second Edition.

Conventions

The following conventions are used in this addendum:

Convention	Meaning
DIGITAL Clusters 1.*n*	Refers to DIGITAL Clusters for Windows NT Version 1.*n*, where *n* has a value of 0 or 1.
Chapter *n*	Chapter references throughout this addendum refer to chapters in the *DIGITAL Clusters for Windows NT Administrator's Guide*.
Bold	Bold type indicates the actual commands, words, or characters that you type in a dialog box or at the command prompt.
Italic	Italic type indicates a placeholder for information on parameters that you must provide. For example, if the procedure asks you to type *filename*, you must type the actual name of a file. Italic type also indicates new terms and the titles of other manuals in the DIGITAL Clusters for Windows NT package. Italic type is used for emphasis within procedures as well.
ALL UPPERCASE	All uppercase letters indicates an acronym.
`Monospace`	Monospaced type represents examples of screen text or entries that you might type at the command line or in initialization files.
▶	A right triangle indicates a procedure with sequential steps.

Related Information

Several other key sources of information included in the DIGITAL Clusters for Windows NT package will help you plan for and use the cluster software:

- Online release notes

- *DIGITAL Clusters for Windows NT Administrator's Guide*

- *DIGITAL Clusters for Windows NT Configuration and Installation Guide*

- Online help

3

Configuring Database Software for Failover

This chapter presents Microsoft SQL Server and Oracle7 Server installation and configuration requirements in the DIGITAL Clusters for Windows NT environment; recommendations; restrictions; and verification procedures.

Microsoft SQL Server Installation and Configuration Requirements

By combining the failover features of DIGITAL Clusters for Windows NT with those of Microsoft SQL Server for Windows NT, you can have a high availability database solution. This section discusses the necessary steps to ensure high availability of SQL Server databases in the DIGITAL Clusters for Windows NT environment.

New Features and Enhancements in DIGITAL Clusters 1.1

The following new features and enhancements to SQL Server since DIGITAL Clusters 1.0 are included in DIGITAL Clusters 1.1 with Service Pack 1 (SP1) and Version 1.0 with Service Pack 3 (SP3):

- Support for multiple, independent SQL Server databases. You can now use multiple SQL Server databases independent of one another. You can enroll and unenroll databases in failover groups as desired. SQL Server clients now have the option of establishing connections to databases using either TCP/IP sockets or named pipes. Clients can connect to a particular group of SQL Server databases.

- Ability to run SQL Server simultaneously on both cluster servers. You can now run SQL Server on both cluster servers, each supporting a different group of SQL Server databases. If one server fails, the other server can assume support for all databases.

- Automated configuration of SQL Server databases. Cluster Administrator now provides a more automated setup sequence for SQL Server databases, eliminating manual invocations of stored procedures.

- IP address failover support. DIGITAL Clusters 1.1 provides support for failing over SQL Server clients that use the TCP/IP network protocol. A cluster IP address can be associated with the SQL Server software. If SQL Server fails over from one cluster server to the other, the associated cluster IP address will migrate with the application. SQL Server clients continue to use the same cluster IP address. The address is transparently routed to the alternate cluster server.

Prerequisite Information

This section gives important information that you should review before using Cluster Administrator to configure for SQL Server database failover.

Software Requirements

Software requirements are as follows:

- Acquire one or more Microsoft SQL Server licenses for each cluster server in accordance with the Microsoft SQL Server licensing requirements.

- Install Microsoft SQL Server Version 6.5 with Service Pack 1 or 2 on a local disk on each cluster server.

Designating Primary and Failover Servers

In Microsoft SQL Server terms, the SQL Server product assumes a static definition of both a *primary server* and a *fallback server*. In contrast, the DIGITAL Clusters software defines a *primary server* and a *failover server* when you use Cluster Administrator to add a failover group. The cluster software uses these definitions only for the purpose of failback, in which failover group control is returned to the primary server when the primary server returns to operational status. See the section Failback in Chapter 2 for details.

To simplify the discussion, we will use the term *failover server* in references to both the Microsoft SQL Server product and the DIGITAL Clusters product.

SQL Server Database Failover

Microsoft SQL Server database failover refers to a database failing over, not the entire application. Upon primary server failure, the failover server will service the databases on the shared disks.

To ensure failover, the servers and databases must be properly configured by invoking stored procedures supplied with the Microsoft SQL Server product. In DIGITAL Clusters 1.1, the cluster software automatically invokes these procedures when you use the Manage SQL Server Databases dialog box of Cluster Administrator and choose the Enroll button. See the section Configuring SQL Server for Failover for details.

Access to Shared Disks

Access to shared disks is not required when installing Microsoft SQL Server. However, it is required when adding new SQL Server databases that you would like to designate as highly available. You can create the new databases either before or after installing the cluster server software.

Configuration Requirements

Verify the following to ensure that your cluster servers and clients are configured properly to run Microsoft SQL Server with DIGITAL Clusters for Windows NT:

- Both cluster servers must reside in the same Windows NT domain.

- Neither cluster server can be a member of any other cluster.

- The cluster servers must be configured with the same network transports to enable interserver communication. For example, if you use the TCP/IP network protocol, the servers must be in the same IP subnet and NetBIOS must be installed on each.

- Cluster clients connecting to SQL Server using named pipes must be configured with the same network transports as the server systems. For example, if the servers communicate with clients using named pipes over TCP/IP, the clients and servers must be in the same IP subnet and NetBIOS must be installed on each. In comparison, clients using the NetBEUI protocol can communicate with cluster servers on the same LAN.

  Cluster clients connecting to SQL Server using TCP/IP sockets do not have the same subnet restriction as named pipe connections over TCP/IP.

- The SQL Server administrator (sa) account password *must* be the same on each cluster server.

- Passwords must match between the SQL Server administrator (sa) account and the SQL Server failover object. Mismatched passwords can cause the DIGITAL Clusters Failover Manager to freeze.

- Both cluster servers *must* be configured to have the same drive-letter assignments for the shared disk or storage set.

Microsoft SQL Server Installation and Configuration Requirements

Configuration and Run-Time Recommendations

Before you configure Microsoft SQL Server to run with DIGITAL Clusters, review the following recommendations:

- Do not install the SQL Server software on a Windows NT Primary Domain Controller (PDC) or Backup Domain Controller (BDC). This recommendation is stated in the Microsoft SQL Server product documentation as well.

- DIGITAL strongly recommends that you install the SQL Server software on a local disk on each server system *before* installing the DIGITAL Clusters software.

- Before initiating a manual failover of an SQL Server database, you *must* close all active client connections to the database.

Restrictions

The following Microsoft SQL Server restrictions have been identified in DIGITAL Clusters 1.1:

- You cannot install SQL Server while the Cluster Failover Manager and Cluster Failover Management Database Server (CFMD Server) services are running.

  If you have chosen to install the SQL Server software after installing the cluster software (DIGITAL strongly recommends that you install the SQL Server software *before* the cluster software), verify that the Cluster Failover Manager and CFMD Server services are not running by using the Services applet on the Windows NT Control Panel.

- When creating SQL Server databases on shared cluster disks, you must give each database in the cluster a unique name. There cannot be two databases with the same name on shared disks served by different cluster servers.

- If you create an SQL Server database for load, you must restore the database before enrolling it for failover.

- Before manually failing over a group in Cluster Administrator, all SQL Server databases on the group's disks *must* be enrolled for failover support. If some databases are enrolled and others are not, Cluster Administrator will disallow manual failover.

- Before deleting an SQL Server database, you first must use Cluster Administrator to unenroll it. (See the section Managing SQL Server Databases in Chapter 8 for instructions.) If you fail to do this, you will not be able to enroll additional databases in the cluster.

Configuring SQL Server for Failover

This section presents the steps you must perform to configure Microsoft SQL Server for failover. Two step sequences are presented depending on when you choose to create the SQL Server databases on the shared disks. DIGITAL strongly recommends using the first step sequence because you are less likely to encounter potential problems.

In the first sequence, the steps are as follows:

1. Install the SQL Server software on a local disk on each cluster server.

2. Install the DIGITAL Clusters software on each server system.

3. Create a failover group for one or more SQL Server databases.

4. Create the SQL Server databases on shared disks.

5. Enroll the SQL Server databases for high availability.

In the second sequence, the steps are as follows:

1. Install the SQL Server software on a local disk on each cluster server.

2. Create the SQL Server databases on shared disks.

3. Install the DIGITAL Clusters software on each server system

4. Create a failover group for one or more SQL Server databases.

5. Enroll the SQL Server databases for high availability.

▶ **To configure the SQL Server software for failover:**

1. Install Microsoft SQL Server on a local disk on each cluster server.

2. If you choose to create your shared SQL Server databases before installing the cluster software, you must follow these steps:

   a. Shut down and turn off the power to the failover server. This will ensure that only the primary server will be allowed access to the shared disks.

   _____ **Caution** _____

   There is danger of disk corruption if both server systems are turned on with the shared bus connected when the cluster software is not installed.
   _____

   b.   On the primary server, use a database administrator tool to create the SQL Server databases (and optionally, the transaction logs) on the shared disks.

_____ **Caution** _____

A database may not be split between disks that will be placed in different failover groups.
_____

3.   Install the cluster software on each server system. See the *Configuration and Installation Guide* for instructions.

4.   On the primary server, use Cluster Administrator to create a failover group for one or more SQL Server databases using the following guidelines:

   a.   Each failover group must contain *all* the shared disks on which a given SQL Server database resides.

   b.   You may have more than one database in a given failover group.

   c.   A shared disk may be specified only in one failover group.

_____ **Caution** _____

A database may not be split between disks that are in different failover groups.
_____

After creating a failover group, the shared disks will be placed on line on the primary server for the group. See the section Creating a Failover Group in Chapter 9 for instructions.

5.   Repeat step 4 for additional SQL Server databases that resides on shared disks.

6.   If you have not already done so, use a database administrator tool to create the SQL Server databases (and optionally, the transaction logs) on the shared disks.

_____ **Caution** _____

A database may not be split between disks that are in different failover groups.
_____

7. Configure an SQL Server database for high availability:

_____ **Note** _____

Both cluster servers must be running, and the MSSQLServer service must be
running on each server.

_____

a. On each cluster server, use the Services applet of the Windows NT Control Panel
    to check if the MSSQLServer service is running. If not, start the service.

b. Using Cluster Administrator, choose SQL Server Databases from the Manage
    menu.

    The Manage SQL Server Databases dialog box is displayed. It lists all the
    databases residing on shared cluster disks that are eligible for failover support.
    Databases that already have been configured for high availability are followed by
    "Yes", whereas databases that have not been configured for high availability yet
    are followed by "No".

c. Select the database that you want to configure for high availability.

d. Choose the Enroll button. This operation configures the SQL Server software on
    both cluster servers and automatically creates the SQL Server failover object
    needed to enable database failover.

See the next section, Moving SQL Server Objects Within a Failover Group, and the
section Managing SQL Server Databases in Chapter 8 for further information.

8. Repeat step 7 for each additional SQL Server database that you want to configure for
    high availability.

Moving SQL Server Objects Within a Failover Group

In DIGITAL Clusters 1.1 and 1.1 SP1, Cluster Administrator automatically adds and
removes SQL Server failover objects from the appropriate groups when you perform SQL
Server management functions such as enrolling or unenrolling databases for failover.
Generally, you do not need to add or remove SQL Server failover objects from a group.

An exception is when you want to change the order of objects in a group. For example,
you may want to place a script failover object after an SQL Server failover object to
ensure that the script runs correctly. In this case, you can use the Modify Failover Group
dialog box to remove the SQL Server failover object from the group and then reinsert the
object in the desired sequence.

Configuring Clients to Access SQL Server Databases

To access an SQL Server database, a client must use either named pipes or TCP/IP sockets. Your choice of client configuration method will depend on which versions of Windows NT and the DIGITAL Clusters software you are running.

DIGITAL Clusters 1.1 SP1 adds support for client connections to a NetBIOS name via the Windows Internet Naming Service (WINS) Server. A client can connect to a database using the NetBIOS name associated with the IP failover object in the same failover group. Using named pipes with NetBT simplifies client configuration because you do not need to use the SQL Server Client Configuration Utility to create a database alias.

Using Named Pipes to Configure an SQL Server Client Connection

Configuring an SQL Server client connection using named pipes requires the following versions of the Windows NT and DIGITAL Clusters software:

- Windows NT 3.51 with NT SP5 *and* DIGITAL Clusters 1.0 or later

- Windows NT 4.0 with NT SP2 or later *and* DIGITAL Clusters 1.1 or later

▶ **To configure an SQL Server client connection using named pipes:**

1. Using the SQL Server 6.5 installation CD–ROM, install the WIN32® client utilities (Utilities Only option).

2. Start the SQL Server Client Configuration Utility as described in the section Starting the Client Configuration Utility of the *Microsoft SQL Server Administrator's Companion*.

3. Choose the Advanced option as described in the section Setting Up Server Connections of the *Microsoft SQL Server Administrator's Companion:*

    - *For Windows NT and Windows 95 Clients:*

      Choose the Advanced tab. The SQL Server Client Configuration Utility dialog box is displayed.

    - *For Windows for Workgroup Clients:*

      Choose the Advanced button. The Advanced Client Options dialog box is displayed.

4. Fill in the boxes as follows:

_____**Note** _____

A client cannot simultaneously establish connections to an SQL Server using named pipes and IP.
_____

a. In the Server box, create an SQL Server database alias.

b. In the DLL Name box, type the dynamic link library (DLL) name for the named pipe Net-Library. Or, type or choose from the drop-down list Named Pipes.

c. In the Connection String box, use this syntax:

`\\`*ClusterName*`\pipe\`*ObjectName*`\query`

where the variables have the following meanings:

ClusterName	Specifies the name of your cluster.
ObjectName	Specifies the name of the SQL Server failover object that the cluster software created automatically for you. This name is the *same* as the name of the failover group you created (in step 4 of the procedure "To configure the SQL Server software for failover") that contains the shared disks on which the SQL Server database resides.

For example, suppose your cluster name is `sqlcluster`; you have created a failover group for your SQL Server database named `sqldbl`; and SQL Server is using the default named pipe name `\\.\pipe\sql\query`. Your connection string would be:

`\\sqlcluster\pipe\sqldbl\query`

5. Add the connection information to the appropriate configuration file on the client.

- *For Windows NT and Windows 95 Clients:*
  Choose the Add/Modify button.

- *For Windows for Workgroups Clients:*
  Choose the Add/Change button, and then the OK button.

6. Choose the Done button.

Using TCP/IP Sockets to Configure an SQL Server Client Connection

Configuring an SQL Server client connection using TCP/IP sockets requires the following versions of the Windows NT and DIGITAL Clusters software:

- Windows NT 4.0 with NT SP2 or later *and* DIGITAL Clusters 1.1 or later

▶ **To configure a SQL Server client connection using TCP/IP sockets:**

1. Using the SQL Server 6.5 installation CD–ROM, install the WIN32® client utilities (Utilities Only option).

2. Start the SQL Server Client Configuration Utility as described in the section Starting the Client Configuration Utility of the *Microsoft SQL Server Administrator's Companion*.

3. Choose the Advanced option as described in the section Setting Up Server Connections of the *Microsoft SQL Server Administrator's Companion:*

   - *For Windows NT and Windows 95 Clients:*

     Choose the Advanced tab. The SQL Server Client Configuration Utility dialog box is displayed.

   - *For Windows for Workgroup Clients:*

     Choose the Advanced button. The Advanced Client Options dialog box is displayed.

4. Fill in the boxes as follows:

   _____**Note** _____

   A client cannot simultaneously establish connections to an SQL Server using named pipes and IP.

   _____

   a. In the Server box, create an alias for the server that is currently controlling the SQL Server database to which the client wants to connect. The Server name should be unique amongst all servers to which the client can connect.

   b. In the DLL Name box, type the following:

      - *For Windows NT and Windows 95 clients:*

        Type the DLL name for the TCP/IP Net-Library. Or, type or choose from the drop-down list TCP/IP Sockets.

      - *For Windows for Workgroups clients:*

        Type the DLL name dbmssoc3.

    c.   In the Connection String box, enter either the IP address that you used when creating the IP failover object for the SQL Server database, or the name that you associated with the IP address in the `etc/hosts` file. See the section Creating an IP Failover Object in Chapter 9 for details.

_____**Note** _____

The IP address contained in the SQL Server failover object must be in the same subnet as the server IP address.

_____

5. Add the connection information to the appropriate configuration file on the client.

- *For Windows NT and Windows 95 Clients:*
  Choose the Add/Modify button.

- *For Windows for Workgroups Clients:*
  Choose the Add/Change button, and then the OK button.

6. Choose the Done button.

Using Named Pipes with NetBIOS and TCP/IP (NetBT) to Configure an SQL Server Client Connection

Configuring an SQL Server client connection using named pipes with NetBIOS and TCP/IP (NetBT) requires the following versions of the Windows NT and DIGITAL Clusters software:

- Windows NT 4.0 with NT SP3 *and* DIGITAL Clusters 1.1 with Clusters SP1

▶ **To configure a SQL Server client connection using named pipes with NetBIOS and TCP/IP (NetBT):**

1. Using the SQL Server 6.5 installation CD–ROM, install the WIN32® client utilities (Utilities Only option).

2. Start the ISQL/w utility.

   The Connect Server dialog box is displayed.

3. In the Server field, type the NetBIOS name associated with the IP failover object in the same group.

4. In the Login Id and Password fields, type the appropriate user account information.

5. Choose the Connect button.

Verifying SQL Server Failover

Use the next procedure to verify that the SQL Server software is failing over properly in the DIGITAL Clusters for Windows NT environment.

▶ **To verify that the SQL Server software is failing over properly:**

On the Primary Server:

1. Use the Windows NT Registry editor, `regedt32.exe`, to open and examine the Registry. Be sure to place the editor in read-only mode by enabling Read Only Mode on the Options menu. For details on using the Registry editor, either select Help from within the program or refer to the documentation packaged with your Windows NT operating system:

    a. Locate the following keys:

    For DIGITAL Clusters 1.0 with Clusters SP1:

    ```
    HKEY_LOCAL_MACHINE/System/CurrentControlSet/Services/
    Cfmd/Database/Pipes/SQL
    ```

    For DIGITAL Clusters 1.0 with Clusters SP3 and Version 1.1 with Clusters SP1:

    ```
    HKEY LOCAL MACHINE/System/CurrentControlSet/Services/
    Cfmd/Database/Pipes/SQL-failover-object
    ```

    Note that in Version 1.0 with Clusters SP3 and Version 1.1 with Clusters SP1, there are one or more keys (in addition to HKEY LOCAL MACHINE/SYSTEM/CurrentControlSet/Services/Cfmd/Database/Pipes/SQL, retained for backward compatibility with Clusters SP1) with names corresponding to all the SQL Server failover objects automatically created by enrolling databases for failover support.

    b. Expand the keys and locate the `ConnectionPoint` parameter.

    c. Verify that the value of the `ConnectionPoint` parameter is the name of the primary server.

2. Shut down the primary server.

On the Failover Server:

3. Repeat step 1, verifying that the value of the `ConnectionPoint` parameter is now the name of the failover server.

_____ **Note** _____

It may take up to 2 minutes for the Registry on the failover server to be updated. Be sure to enable Auto Refresh on the Options menu.
_____

On a Cluster Client:

4. For client connections using named pipes, use the SQL Client Configuration Utility to configure an SQL Server alias. See step 3 of the section Configuring Clients to Access SQL Server Databases for details.

5. Using either the ISQL/w application or the Enterprise Manager Query Tool, use the SQL Server alias created in step 4, not the server name, to query an SQL Server database residing on a shared disk:

   - If the client cannot access the SQL Server database, verify that you have followed the instructions outlined earlier in this chapter.

   - If problems persist, contact your local DIGITAL Multivendor Customer Services sales specialist.

On the Primary Server:

6. Bring the primary server on line again.

7. Repeat step 1, verifying that the value of the `ConnectionPoint` parameter has returned to the name of the primary server.

_____ **Note** _____

It may take up to 2 minutes for the Registry on the primary server to be updated. Be sure to enable Auto Refresh on the Options menu.
_____

On a Cluster Client:

8. Repeat step 4.

Adding a Shared Disk to a Failover Group for SQL Server

If you want to modify a failover group for an SQL Server failover object to include another shared disk, you must follow the instructions in the next procedure.

▶ **To modify a failover group for an SQL Server failover object to include another shared disk:**

1. Using Cluster Administrator, add the new shared disk to the failover group. Be sure to arrange the objects in the group so that all the disks are placed *before* the SQL Server failover object. See the section Modifying a Failover Group in Chapter 9 for instructions.

2. Use a database administrator tool to create additional SQL Server databases (and optionally, the transaction log) on the shared disk.

3. Configure the new databases for high availability:

   a. On each cluster server, use the Services applet of the Windows NT Control Panel to check if the MSSQLServer service is running. If not, start the service.

   b. Using Cluster Administrator, choose SQL Server Databases from the Manage menu.

      The Manage SQL Server Databases dialog box is displayed. It lists all the databases residing on shared cluster disks that are eligible for failover support. Databases that already have been configured for high availability are followed by "Yes", whereas databases that have not been configured for high availability yet are followed by "No".

   c. Select the database that you want to configure for high availability.

   d. Choose the Enroll button. This operation configures the SQL Server software on both cluster servers and creates the SQL Server failover object needed to enable database failover.

   See the Section Managing SQL Server Databases in Chapter 8 for details.

Unenrolling an SQL Server Database from Failover Support

There may be times when you will want to unenroll an SQL Server database from cluster failover support. The next procedure outlines the steps you need to perform to do so.

▶ **To unenroll an SQL Server database from cluster failover support:**

1. On each cluster server, use the Services applet of the Windows NT Control Panel to check if the MSSQLServer service is running. If not, start the service.

2. Using Cluster Administrator, choose SQL Server Databases from the Manage menu.

   The Manage SQL Server Databases dialog box is displayed. It lists all the databases residing on shared cluster disks that are eligible for failover support. Databases that already have been configured for high availability are followed by "Yes", whereas databases that have not been configured for high availability yet are followed by "No".

3. Select a database that you want to unenroll.

4. Choose the Unenroll button.

Upon completing this procedure, the SQL Server database will have returned to the same state it was in just prior to enrolling it for cluster failover support. Note that unenrolling an SQL Server database does not affect primary server access.

See the section Managing SQL Server Databases in Chapter 8 for further details.

Expanding or Shrinking an SQL Server Database

To change the size of a database that has been enrolled for cluster failover support, follow the instructions outlined in the next procedure.

▶ **To expand or shrink an SQL Server database:**

1. Execute the procedure in the previous section Unenrolling an SQL Server Database from Failover Support.

2. If you will be expanding the database onto an additional disk, check if the failover group containing the SQL Server failover object also contains this disk. If not, use Cluster Administrator to add the new disk to the failover group, placing it *before* the SQL Server failover object. See the section Modifying a Failover Group in Chapter 9 for instructions.

3. Use a database administrator tool to expand or shrink the SQL Server database.

4. Reenroll the database for failover support:

   a. On each cluster server, use the Services applet of the Windows NT Control Panel to verify that the MSSQLServer service is running.

   b. Using Cluster Administrator, choose SQL Server Databases from the Manage menu.

      The Manage SQL Server Databases dialog box is displayed. It lists all the databases residing on shared cluster disks that are eligible for failover support. Databases that already have been configured for high availability are followed by "Yes", whereas databases that have not been configured for high availability yet are followed by "No".

    c.   Select the database that you want to configure for high availability.

    d.   Choose the Enroll button. This operation configures the SQL Server software on both cluster servers and creates the SQL Server failover object needed to enable database failover.

See the section Managing SQL Server Databases in Chapter 8 for further details.

Moving an SQL Server Database and Disk to a Different Failover Group

If you want to move an SQL Server database and the disk on which it resides to a different failover group, both groups must be on line on the *same* server at the time you move the database.

▶ **To move an SQL Server database and its disk to a different failover group:**

1.   Verify on which disk the database resides.

2.   Deactivate the database. See the section Managing SQL Server Databases in Chapter 8 for instructions.

3.   Move the disk on which the database resides out of the current failover group. See the section Modifying a Failover Group in Chapter 9 for instructions.

4.   Move the disk into the new failover group, placing it *before* the SQL Server failover object.

5.   Enroll the database. See the section Managing SQL Server Databases in Chapter 8 for instructions.

Moving SQL Server Databases to a New Cluster

If you have existing SQL Server databases that you want to move to a new cluster, for example, if you are upgrading from DIGITAL Clusters 1.0 with Clusters SP1, SP2, or SP3 to DIGITAL Clusters 1.1 with Clusters SP1, use the following procedure.

_____ **Note** _____

The databases must be placed on a shared disk in the new cluster.

_____

▶ **To move SQL Server databases to a new cluster:**

1.   Ensure that there are no client connections to the SQL Server databases.

2.   For *each* database, invoke the following stored procedure on the primary server to place it off line:

```
sp_dboption DatabaseName, offline, true
```

3. Install the DIGITAL Clusters software. See the *Configuration and Installation Guide* for instructions.

4. On the primary server, use Cluster Administrator to create a failover group for one or more SQL Server databases using the following guidelines:

   a. Each failover group must contain *all* the shared disks on which a given SQL Server database resides.

   b. You may have more than one database in a given failover group.

   c. A shared disk may be specified only in one failover group.

   _____ **Caution** _____

   A database may not be split between disks that are in different failover groups.
   _____

   After creating a failover group, the shared disks will be placed on line on the primary server for the group. See the section Creating a Failover Group in Chapter 9 for instructions.

5. Configure an SQL Server database for high availability:

   _____ **Note** _____

   Both cluster servers must be running, and the MSSQLServer service must be running on each server.
   _____

   a. On each cluster server, use the Services applet of the Windows NT Control Panel to check if the MSSQLServer service is running. If not, start the service.

   b. Using Cluster Administrator, choose SQL Server Databases from the Manage menu.

   The Manage SQL Server Databases dialog box is displayed. It lists all the databases residing on shared cluster disks that are eligible for failover support. Databases that already have been configured for high availability are followed by "Yes", whereas databases that have not been configured for high availability yet are followed by "No".

   c. Select the database that you want to configure for high availability.

      d.   Choose the Enroll button. This operation configures the SQL Server software on both cluster servers and creates the SQL Server failover object needed to enable database failover.

      See the section Managing SQL Server Databases in Chapter 8 for further details.

6.   Repeat step 5 for each additional SQL Server database that you want to configure for high availability.

7.   For *each* database, invoke the following stored procedure on the primary server to place it on line:

```
sp_dboption DatabaseName, offline, false
```

Resetting the Suspect Status of an Unenrolled SQL Server Database

If an unenrolled SQL Server database on a cluster share becomes suspect after a failover, use the next procedure to reset the database status.

▶ **To reset the suspect status of an unenrolled SQL Server database on a cluster share after a failover:**

1.   Enroll the database in a failover group. See the section Managing SQL Server Databases in Chapter 8 for instructions.

2.   Using the SQL Enterprise Manager Query Tool, run the following stored procedures. These procedures are listed in the master database:

   a.   `sp_fallback_dec_clean`

   b.   `sp_fallback_activate_svr_db PrimaryServerName, DatabaseName`

      where the parameters have the following meanings:

PrimaryServerName	Specifies the name of the server where the database's disks are on line.
DatabaseName	Specifies the name of the database.

3.   Unenroll the database from the failover group. See the section Managing SQL Server Databases in Chapter 8 for instructions.

Oracle7 Server Installation and Configuration Requirements

Oracle failover support in the DIGITAL Clusters for Windows NT product offers a high availability database solution for Oracle7 Workgroup Servers and Oracle7 Enterprise Servers. With proper configuration, when a primary server running Oracle7 Server fails, the server instance will fail over to the failover server. This section outlines the necessary steps for configuring a highly available Oracle7 Server in the DIGITAL environment.

New Features in DIGITAL Clusters 1.1

DIGITAL Clusters 1.1 and Version 1.0 with SP2 include the following new features and enhancements to Oracle7 Server since DIGITAL Clusters 1.0:

- Failover support for Oracle7 Server Version 7.3.

- IP address failover support. DIGITAL Clusters 1.1 provides support for failing over Oracle7 Server clients that use the TCP/IP network protocol. A cluster IP address can be associated with an Oracle7 Server instance. If the instance fails over from one cluster server to the other, the associated cluster IP address will migrate with the application. Oracle7 Server clients continue to use the same cluster IP address. The address is transparently routed to the alternate cluster server.

Prerequisite Information

This section gives important information that you should review before using Cluster Administrator to configure for Oracle7 Server database failover.

Software Requirements

Software requirements are as follows:

- Acquire one or more Oracle7 Workgroup Server or Oracle7 Enterprise Server licenses for each cluster server in accordance with the Oracle7 Server licensing requirements.

- Install Oracle7 Server Version 7.1, 7.2, or 7.3 on a local disk on each cluster server.

_____**Note** _____

Do not mix Oracle7 Server versions between cluster servers.
_____

- If you want to use the DIGITAL Clusters IP failover feature with Oracle7 Server, you must use Oracle7 Server Version 7.3 with DIGITAL Clusters 1.1. IP failover requires Microsoft Windows NT 4.0, which is not supported by earlier versions of Oracle7 Server and DIGITAL Clusters.

Oracle Server Database Failover

Throughout the DIGITAL Clusters for Windows NT documentation, *Oracle Server database failover* refers to an Oracle7 Server instance failing over. Oracle7 Server allows multiple instances to run on a single cluster server. Operations of instances that have failed over will not affect operations of other instances.

The database associated with a failover instance can be accessed only by one server system at a time. Should the primary server become unavailable, the failover server is ready to service the instance database on the shared disks, thus providing high availability.

Configuration Requirements

Verify the following to ensure failover of your Oracle instance:

- Both cluster servers must reside in the same Windows NT domain.

- Neither cluster server can be a member of any other cluster.

- The cluster servers must be configured with the same network transports to enable interserver communication. For example, if you use the TCP/IP network protocol, the servers must be in the same IP subnet and NetBIOS must be installed on each.

- The shared disks used for Oracle instance failover must be assigned identical fixed drive letters on each cluster server.

- All files associated with the Oracle instance, including the data files, control files, log files, and parameter file, must reside on one or more shared disks.

- When you create or modify a failover group for Oracle Server, you *must* place the objects in the following sequence:

    1. Disk object or objects

    2. IP failover object

    3. Oracle failover object

    If you fail to do this, the Oracle network listener will not start.

Creating and Configuring an Oracle Instance for Failover

This section presents the steps you must perform to create and configure an Oracle instance for failover. Following is an overview of the step sequence. For detailed instructions, see the next procedures "Before creating an Oracle instance for failover" and "To create an Oracle instance for failover."

1. Install the DIGITAL Clusters software on each server system.

2. Create a failover group containing all shared disks to be used for Oracle instance failover.

3. On each cluster server, assign identical fixed drive letters for the shared disks.

4. *For IP socket connections:* Create an IP failover object. Then modify the failover group to include the IP failover object.

5. *On the Primary Server:* Create an Oracle instance for failover; create an associated database on one or more shared disks; create and configure the Oracle network listener description; start the network listener; then manually fail over the group to the failover server.

6. *On the Failover Server:* Create an identical Oracle instance to the one you created on the primary server.

7. *On the Primary Server:* Create an Oracle failover object. Then modify the failover group to include the Oracle failover object.

8. Repeat the appropriate steps for *each* instance that you configure for failover.

▶ **Before creating an Oracle instance for failover:**

1. If you have not already done so, install the DIGITAL Clusters software on each server system.

2. Using Cluster Administrator, create a failover group containing all shared disks to be used for Oracle instance failover. Give the group a meaningful name. See the section Creating a Failover Group in Chapter 9 for instructions.

    The shared disks will be placed on line on the primary server.

3. On each cluster server, assign identical fixed drive letters for the shared disks used for Oracle instance failover.

    On the Primary Server:

    a. On the primary server for the shared disks used for Oracle instance failover, click Start and choose Programs→Administrative Tools→Disk Administrator.

    b. Select a shared disk for which you want to modify the drive-letter assignment.

    c. From the Tools menu, choose Drive Letter.

        The Assign Drive Letter dialog box is displayed.

    d. Assign a fixed drive letter. To avoid conflict with network drive letters, DIGITAL strongly recommends that you select a drive letter at the end of the alphabet, such as X, Y, or Z.

    e. Repeat steps b to d for each shared disk used for Oracle instance failover.

   f. Reboot the server.

   g. Manually fail over the group to the failover server. See the section Managing Manual Failover in Chapter 8 for instructions.

  On the Failover Server:

   h. Once failover has completed, repeat steps a to f on the failover server. You must assign the same fixed drive letters on both server systems.

   i. Manually fail back the group to the primary server. See the section Managing Manual Failover in Chapter 8 for instructions.

  On the Primary Server:

   j. Using Disk Administrator, verify that the shared disks have failed over.

4. *For IP socket connections:*

  a. Create an IP failover object, making note of the IP address that you specify for later use in creating the `listener.ora` file. See the section Creating an IP Failover Object in Chapter 9 for instructions.

  b. Modify the failover group that you created in step 2 to include the IP failover object.

▶ **To create an Oracle instance for failover:**

Repeat this procedure for *each* instance that you configure for failover.

On the Primary Server:

1. Use an Oracle database administrator tool (for example, ORADIM*nn*, where *nn* corresponds to the version number [7.1, 7.2, or 7.3] of Oracle7 Server running on your cluster servers) to create an Oracle instance for failover. Note the Oracle System Identifier (SID) that you assign to the instance for later use in creating the `listener.ora` file.

2. Create an associated database on one or more shared disks. If you are working with an existing database, move the database data files to one or more shared disks. Also use a shared disk to store all files associated with the database, including the control file, log file, and parameter file.

  Note that creating a database involves several steps, such as creating the parameter file, editing the associated parameters, creating the database, allocating rollback segments, and so forth. Consult the Oracle7 Server documentation for details.

3. Create and configure the network listener description.

   For Versions 7.2 and 7.3: If you are running Oracle7 Server Version 7.2 or 7.3, create a separate network listener description for each instance using a supported Oracle7 Server tool such as Oracle Network Manager for Windows. See the Oracle Network Manager for Windows product documentation for details. Use the following guidelines when creating the listener (note that Oracle7 Server terminology is designated in quotes):

   For named pipe connections:

   a. Create an address for the network listener by specifying Named Pipe as the "network protocol".

   b. Specify a unique name for the named "pipe" associated with this address of the listener.

   c. Specify the cluster alias as the "server" name for this address.

   For IP socket connections:

   a. Create an address for the network listener by specifying TCP/IP as the "network protocol."

   b. Specify a unique port number for the "port" associated with this address.

   c. Specify either the IP address or the associated IP name (as translated by the name service you are using) of the IP failover object for the "host" associated with this address of the listener.

   For simultaneous use of named pipe and IP socket connections:

   a. Create both a named pipe address and a TCP/IP address for the same network listener following the previous instructions in this step.

   The end result of step 3 is a description of the network listener in the file `listener.ora`. This file contains descriptions of all the listeners used on the server system. There is only one `listener.ora` file per server system. See the next section, Example listener.ora File, for a sample file extract.

4. Using either the Oracle utility LSNRCTL or the Services icon on the Windows NT Control Panel, check if there is a network listener running. If so, stop and then restart it. Otherwise, start the network listener.

   _____ **Note** _____

   You must use the LSNRCTL utility to start a new network listener for the first time. The service name on the Control Panel will appear as the listener name prefixed by "OracleTNSListener".

   _____

5. With the associated database mounted and open, verify that you can access it using an Oracle database administrator tool such as SQLDBA*nn* or SVRMGR*nn* (where *nn* designates the Oracle7 Server version number). Use the Oracle SID to access the database. If you cannot access the database:

   a. In the parameter file, verify that you have modified all path specifications so that files associated with the instance are created on a shared disk.

   b. Verify that all files associated with the instance, including the data file, control file, log file, and parameter file, are on a shared disk.

6. Shut down the instance using an Oracle database administrator tool. Be sure to leave services for the instance running.

7. Using either the Oracle utility LSNRCTL or the Services icon on the Windows NT Control Panel, stop the network listener.

8. Verify that services for the instance are set to restart automatically using either the Oracle Instance Manager or the Services icon on the Windows NT Control Panel. If the services are not set to "Startup Automatic", change this.

9. Manually fail over the group to the failover server. See the section Managing Manual Failover in Chapter 8 for instructions.

On the Failover Server:

10. Once the failover server can access the shared disks, repeat steps 1 and 3 to 9 to create an *identical* Oracle instance to the one you created on the primary server using the following guidelines:

   • When repeating step 1, use the same Oracle SID that you used on the primary server.

   • When repeating step 3, create an identical listener description to the one stored in the primary server's `listener.ora` file. If you have not configured local databases on either cluster server, you can use the same `listener.ora` file by copying it from the primary server.

   • When repeating step 9, manually fail over the group to the primary server.

On the Primary Server:

11. Create an Oracle failover object, supplying information about the instance that you created in steps 1 to 3. See the section Creating an Oracle Failover Object in Chapter 9 for instructions.

12. Modify the failover group that you created in step 2 of the procedure "Before creating an Oracle instance for failover" to include the Oracle failover object that you created in step 11. The Oracle failover object must be placed *last* in the failover group. See the section Modifying a Failover Group in Chapter 9 for instructions.

    Modifying the failover group will start the Oracle Server instance and will make the database available to the Oracle clients, once configured.

13. Verify that the Oracle Server instance is now running on the primary server. If the instance is not running, verify that the information you supplied in step 11 to create the Oracle failover object is correct.

Example listener.ora File

Before Oracle Server can receive connections from clients, a network listener must be active on the server system. The configuration file for the network listener is `listener.ora`.

Following is an extract from a properly configured `listener.ora` file. The extract contains an example description of a network listener named TestList configured for both named pipe and IP socket connections.

```
################
# Filename......: listener.ora
# Name..........: TestCluster.world
# Date..........: 27-JAN-97 10:47:58
###############
SQLNET.AUTHENTICATION_SERVICES = (NONE)

USE_PLUG_AND_PLAY_TestList = OFF
USE_CKPFILE_TestList = OFF
TestList =                                              1
  (ADDRESS_LIST =
        (ADDRESS=
          (PROTOCOL=IPC)
          (KEY= TestDB.world)                           2
        )
        (ADDRESS=
          (PROTOCOL=IPC)
          (KEY= TEST)                                   3
        )
        (ADDRESS =
          (COMMUNITY = NMP.world)
          (PROTOCOL = NMP)                              4
          (Server = TestCluster)
```

```
          (Pipe = testpipe1)                    _____   5
        )
        (ADDRESS =
          (COMMUNITY = TCP.world)
          (PROTOCOL = TCP)
          (Host = 16.151.8.202)                 _____   6
          (Port = 1565)
        )                                                          7
   )
STARTUP_WAIT_TIME_TestList = 0
CONNECT_TIMEOUT_TestList = 10
TRACE_LEVEL_TestList = OFF
SID_LIST_TestList =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = TestDB.world)            _____   8
      (SID_NAME = TEST)
      (PRESPAWN_MAX = 10)                                   9
    )
  )
PASSWORDS_TestList = (oracle)
```

In the previous file extract:

1 Specifies the Oracle network listener name.

2 Specifies the Oracle global database name. This address is used when the Oracle client and server reside on the same system.

3 Specifies the Oracle SID. This name is used when the client and server reside on the same system.

4 Specifies the cluster alias.

5 Specifies a unique pipe name.

6 Specifies the address of the cluster IP failover object associated with this Oracle database. You can use an IP name instead provided the name can be translated to the associated IP address.

7 Specifies a unique port number.

8 Specifies the global database name.

9 Specifies the Oracle SID.

Configuring Clients to Access Oracle Server Databases

You must establish your Oracle client/server connections over one of the following supported network transports: named pipes or IP. A prerequisite for clients using named pipe connections to Oracle Servers is that they must have installed the DIGITAL Clusters client software. Clients using IP connections to Oracle Servers do not need to install the DIGITAL Clusters client software.

A key goal in configuring clients is to provide the addresses of the Oracle Servers. This task must be done regardless of whether the Oracle Server software is running in a DIGITAL Clusters environment. In the previous procedure "To create an Oracle instance for failover", you configured the servers to respond to the network addresses specified in the listener.ora file. The network addresses also must be provided to the clients so that they can communicate with the servers. Typically, clients identify servers using an Oracle service name that gets translated into a network address.

There are several methods that a client can use to translate Oracle service names into network addresses, including:

1. Use a local client configuration file, tnsnames.ora.

2. Use an Oracle Names server if one is used on the network.

3. Use a name service such as Distributed Computing Environment Cell Directory Service (DCE CDS), Sun® Network Information Service (NIS), Novell® NetWare Directory Services (NDS), or Banyan® Street Talk™.

4. Use a combination of the methods just listed.

If you will be using a local client configuration file, Oracle7 Server supplies several utilities to assist in creating this, including the SQL*Net Easy Configuration utility and the Oracle Network Manager for Windows. Note that the SQL*Net Easy Configuration utility can be used only for simple configurations. See your Oracle7 Server documentation for details on configuring clients in the Oracle Server environment.

Configuring the tnsnames.ora File

This section discusses proper configuration of the tnsnames.ora file. If you use other name services, be sure that these map the Oracle service name to the appropriate network address.

You can use Oracle Network Manager to create the tnsnames.ora file for specific servers or network communities. Following is an example tnsnames.ora file that has been created to use with the listener.ora file you created in the procedure "To create an Oracle instance for failover." Note that if you are using only named pipes or only IP, your tnsnames.ora file will contain only the transport you are using:

```
###############
# Filename......: tnsnames.ora
# Name..........: LOCAL_REGION.world
# Date..........: 27-JAN-97 10:47:58
###############
TestDB.world =                                                    1
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = NMP.world)
          (PROTOCOL = NMP)
          (Server = TestCluster)                                 2
          (Pipe = testpipe1)
        )                                                        3
        (ADDRESS =
          (COMMUNITY = TCP.world)
          (PROTOCOL = TCP)
          (Host = 16.151.8.202)                                  4
          (Port = 1565)
        )                                                        5
    )
    (CONNECT_DATA =                                              6
      (SID = TEST)
          (GLOBAL_NAME = TestDB.world)
    )                                                            7
  )
```

In the previous file extract:

1 Specifies the Oracle service name or Oracle database alias.

2 Specifies the cluster alias as configured in the listener.ora file.

3 Specifies a unique pipe name matching the one configured in the listener.ora file.

4 Specifies the IP address configured in the listener.ora file. You can use an IP name instead provided the name can be translated to its IP address.

5 Specifies a unique port matching the one configured in the listener.ora file.

6 Specifies the Oracle SID.

7 Specifies the Oracle global database name.

Client Connections During a Failover

During a failover, client connections to Oracle Server via the Oracle database alias are broken. Clients are responsible for handling this. Once failover has completed, clients should reconnect to the same Oracle database alias, which will connect them to the failover server.

Initiating Manual Failover of an Oracle Instance

If you need to manually fail over an Oracle instance, DIGITAL strongly recommends that you first shut down the instance using the database administrator tools provided with the Oracle7 Server software. This guarantees orderly shutdown of the instance. Otherwise, current transactions will be rolled back instead of committed, and client applications will be responsible for handling this.

▶ **To initiate manual failover of an Oracle instance:**

1. Shut down the instance using an Oracle database administrator tool.

2. Initiate a manual failover using Cluster Administrator. See the section Managing Manual Failover in Chapter 8 for instructions.

12

IP Address Failover for File Shares

This chapter provides information on configuring IP failover for file shares. This support is provided in the DIGITAL Clusters for Windows NT, Version 1.1, Service Pack 1 environment only.

_____ **Note** _____

This feature requires Windows NT 4.0, Service Pack 3.
_____

DIGITAL Clusters for Windows NT can be used to fail over a NetBIOS over TCP/IP (NetBT) name in association with an IP address. This feature enables you to fail over file shares that are accessible over a wide area network using TCP/IP.

The NetBIOS name and IP address are defined as follows:

- The NetBIOS name associated with an IP failover object acts as a virtual cluster server name. Each NetBIOS name and related IP address is associated with a single failover group. File shares are associated with disks in a given group, and addressed using this virtual cluster server name (NetBIOS name) and the name of the file share.

_____ **Note** _____

Your cluster can have multiple NetBIOS names and IP addresses. A NetBIOS name and associated IP address moves between cluster servers to follow their resources, such as shares and disks.

_____

- In the context of a cluster failover object, a NetBIOS name is optional.

  If used, the NetBIOS name is specified in the Create IP Failover Object dialog box (15 characters maximum). The NetBIOS name you select must be unique across your entire network and associated with an IP address.

- There can be multiple NetBIOS names and IP addresses for the cluster.

A name can be associated with an IP address as follows:

- IP addresses can use a string translation to represent the host name, for example:

  BLUE1.MOON2.XYZ.COM

- Server names can be translated into IP addresses using a HOSTS file or the DNS server, for example:

  Host name BLUE1.MOON2.XYZ.COM might translate to 11.22.33.44.

  _____ **Note** _____

  The cluster software does not perform any DNS host name to address translation. You must add host names to the HOSTS file manually or register them with the DNS servers. The Cluster administration tools cannot translate a host name into an IP address. All IP addresses must be entered in the format: aa.bb.cc.dd

  _____

- Cluster servers and clients must have TCP/IP and Winsock enabled to access a socket-based application.

An IP address or host name can establish access to socket-based applications, such as web servers, ftp, and gopher. Servers and clients communicate with each other using the socket API.

Configuring IP Failover for File Shares

The Cluster Administrator allows you to specify NetBIOS names in association with IP failover objects. A NetBIOS name identifies a host that provides LAN manager services such as file shares or named pipes. The NetBIOS name is associated with an IP address, because the LAN manager protocol is actually contained inside TCP/IP packets. LAN manager services are accessible on a wide area network when layered on TCP/IP.

Configuration Requirements

Before you can configure IP failover for file shares, be sure these requirements are met:

- Cluster servers must specify a Windows Internet Naming Service (WINS) server.

- Clients on subnets other than where the server is located must use the WINS name server to resolve Cluster NetBIOS names.

  As an alternative, you can also use a configured LMHOSTS file with NetBIOS names and their related IP addresses.

- Clients must have TCP/IP, NetBT, and NetBIOS enabled to access file shares using a NetBIOS name.

_____ **Note** _____

This feature requires Windows NT 4.0, Service Pack 3.

_____

Configuring IP failover for file shares includes the following steps:

- Create an IP failover object.

- Create a failover group.

- Verify the IP address and NetBIOS name are enabled.

- Create shared files on the assigned disks.

Create an IP Failover Object

The file share failover using IP failover support adds the NetBIOS name field to the Create IP Failover Object dialog box. The IP address and NetBIOS name specified in this dialog box are enabled when the IP failover object goes on line.

▶ **To create an IP failover object:**

1. From the toolbar, select Create, IP Failover Object. The Create IP Failover Object dialog box is displayed.

IP Address Failover for File Shares

2. In the IP Address field, enter the IP address to use as a cluster IP address. For example: 11.22.33.44

3. In the Subnet Mask field, supply the subnet mask associated with the IP address. For example: 255.255.255.0

_____**Note**_____

See your Network Administrator for the correct subnet mask value.

_____

4. In the NetBIOS Name field, enter the name (15 characters or less). For example:
   GROUP1

5. Specify which network adapters on each cluster server can enable the IP address. The list box for each server displays the list of available network adapters.

   To fail over a NetBIOS name, the selected adapter must have NetBT enabled. If a NetBIOS name is not specified for the IP failover object, only TCP/IP needs to be enabled.

   If the default value is Any, the IP address can fail over to any adapter when a failover occurs. Select a specific adapter only if needed for your cluster configuration. For example, if one network adapter belongs to a private LAN, you should select a different network adapter for failover.

6. Click OK to save the modifications to the settings, or click Cancel to cancel the settings.

Create a Failover Group

Use the Cluster Administrator to create a failover group. After creating the group, add the specific failover objects you want to fail over together. See the section Working with a Failover Group in Chapter 9 for specific information.

Verify that the IP Address and NetBIOS Name Are Enabled

On the primary server for the group or the system on which the group is currently located or executing, from an MS–DOS window, type:

`ipconfig/all` (this command displays the IP address)

`nbtstat -n` (this command verifies whether the NetBIOS name is enabled or not)

Create File Shares on the Assigned Disks

Use Windows NT Explorer or My Computer to create file shares on the disks that you previously assigned to the failover group.

Accessing Resources by NetBIOS Name

You can use the NetBIOS name to:

- View a network resource.

- Connect to a network file share.

Use NetBIOS Name to View a Network Resource

Clients can use the `net view` command to display the resources associated with a particular cluster NetBIOS name.

For example:

`M:\>net view \\GROUP1`

`Shared resources at \\GROUP1`

Share name	Type	Used as	Comment
locsharezx	Disk		
SH1	Disk	D:	
SH2	Disk	F:	

`The command completed successfully.`

In this case, both cluster and local shares may be listed. Any local shares listed would be those served from the cluster server on which the group associated with the cluster NetBIOS name is currently executing. The local shares listed may vary depending whether the group is currently on the Primary or the Secondary cluster server for the group.

Resources can also be viewed using the IP address, which is associated with the NetBIOS name.

For example:

```
M:\>net view \\11.22.33.44
```

_____ **Note** _____

The server must have a NetBIOS name associated with the IP address before the host can accept LAN manager requests directed to that NetBIOS name.
_____

Use NetBIOS Name to Connect to a Network File Share

The `net use` command or Map Network Drive in Explorer can be used to connect to a file share, using a NetBIOS name.

For example:

```
M:\>net use * \\GROUP1\SH1

The command completed successfully.

Drive F: is now connected to \\GROUP1\SH1
```