Step-by-Step Guide to Distributed File System (Dfs)

Because shared files are widely distributed across networks, administrators face growing problems as they try to keep users connected to the data they need. The Distributed File System (Dfs) in the Microsoft Windows® 2000 operating system provides a mechanism for administrators to create logical views of directories and files, regardless of where those files physically reside in the network. Fault tolerance of network storage resources is also possible using Dfs.

On This Page

Introduction

Because shared files are widely distributed across networks, administrators face growing problems as they try to keep users connected to the data they need. The distributed file system (Dfs) provides a mechanism for administrators to create logical views of directories and files, regardless of where those files physically reside in the network. Fault tolerance of network storage resources is also possible using Dfs.

This guide describes how to use the Dfs Share Creation wizard.

Prerequisites

The examples provided in this document assume you have already configured the Microsoft Active Directory® service, and have administrator permissions for both the domain and the server where you will be configuring Dfs.

You can create the base configuration by following the Step-by-Step Guide to a Common Infrastructure for Windows 2000 Server Deployment http://www.microsoft.com/windows2000/techinfo/planning/server/serversteps.asp before beginning this document.

If you are not using the common infrastructure, you need to make the appropriate changes to this instruction set.

⇑Top of page

Using the Dfs Administrator Tool

This step-by-step guide describes how to use the Dfs Administrator snap-in. Installation of the Dfs service takes place automatically during Windows 2000 Server Setup. However, you must configure Dfs in order for a Dfs share to be accessible to clients.

Perform these steps on the domain controller while logged on as a user with administrative privileges.

In the Windows 2000 operating system, Dfs can integrate with Active Directory to create fault tolerant Dfs roots on Windows 2000 domain controllers (DCs) and member servers. If you have multiple servers in your Windows 2000 domain, any or all participating servers can host and provide fault tolerance for a given Dfs root. Active Directory is used to ensure domain controllers in the domain share a common Dfs topology, thus providing redundancy and fault tolerance.

Alternatively, you can create a stand-alone Dfs server, which does not take advantage of Active Directory and does not provide root level fault tolerance. A DC can host a single Dfs root, and you can have an unlimited number of Dfs roots in each domain. Up to 32 DCs can host the same root. Multiple Dfs root volumes can be hosted in the domain. Additional computers hosting the root or child nodes (links) improves load balancing, fault tolerance, and site preference to directory service-aware network clients. Dfs links below the root can reside on any UNC path accessible to the Dfs server and clients.

In this walkthrough, it is assumed you are creating a fault-tolerant Dfs Root.

Starting Dfs

1. Click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
2. Right-click **Distributed File System** in the left pane, and click **New Dfs Root**. The **Create New Dfs Root** wizard appears, then click **Next**.
3. Make sure that **Create a domain Dfs root** is selected, and then click **Next**.
4. Select the host domain for the Dfs root; in our example, this is **reskit.com**, and then click **Next**.
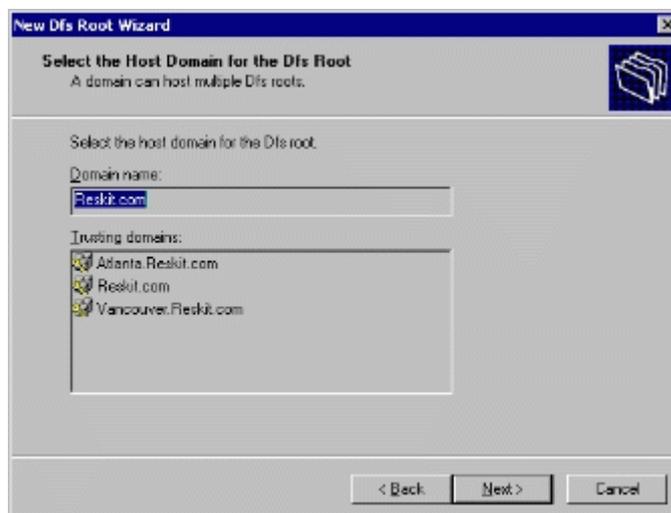

Figure 1: Select Host Domain for Dfs Root

5. Accept the name of the host server for the Dfs root. In our example, this is displayed as **HQ-RES-DC-01.Reskit.com**. Click **Next**
6. Choose the local share point to be used on the target to host the Dfs root. In our example, click **Create a new share** and type the path to share as **c:\dfsbooks** and the share name as **books**. The snap-in lets you create both a new share and new directory if they do not already exist.
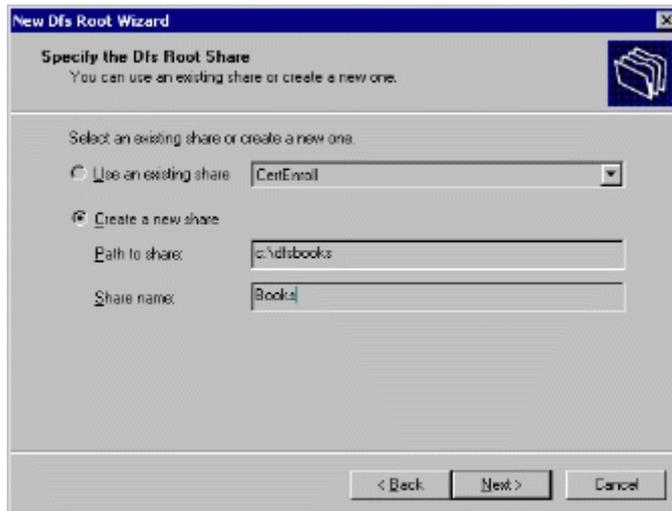
Figure2 Select the share for the Dfs root volume

7. Click **Next**. If the specified folder does not exist, you are asked if you want to create it. Click **Yes** to continue. Add a comment if you wish to further describe this root. Click **Next**.
8. Click **Finish** to create the Dfs root. After the **Create New Dfs Root** wizard has completed, you are ready to administer your Dfs root.

If you have multiple domain controllers hosting a fault tolerant Dfs root, keep in mind that fault tolerant Dfs uses Active Directory to store topology knowledge. Thus, it is necessary for the topology knowledge to converge between the domain controllers. Updates to the Dfs configuration initially take place on the host server in the Windows 2000 domain. Domain controllers may have a different view of the Dfs configuration until multimaster replication from the Dfs host makes changes fully replicated between all domain controllers in a domain. Dfsthe root and all its linksis stored as a single entity known as a *blob*. When a change is made to the blob, the whole blob replicates until consistent throughout the domain.

This takes about five minutes between any given two replicating domain controllers in the same site, and at least 15 minutes if the domain controllers are in different sites. Until convergence occurs, Dfs administrator tools located on different Dfs clients can be presented with a different Dfs configuration. You can click **Refresh** to update Dfs with the current configuration from the Dfs host.

At this point, you have an empty Dfs root in Active Directory. For this share to be interesting to users, you need to publish non-local shares in the Dfs namespace.

To publish non-local shares

1. Right-click your **Dfs Root** name and then click **New Dfs Link**.
2. Right-click **\\Reskit.com\Books**.
3. Click **New Dfs Link**.
4. Specify a directory for the link name. In this example, call the link name **ART**. Locate a valid Windows 2000 share anywhere on your network, and type the full universal naming convention (UNC) name in the **Send the user to this network**

path box. Alternatively, you can browse for it. In our example, this is the **Architecture** share on the **BR3-VAN-SRV-01** server in the **Vancouver** domain. (Note: these shared folders were pre-created for this exercise.)
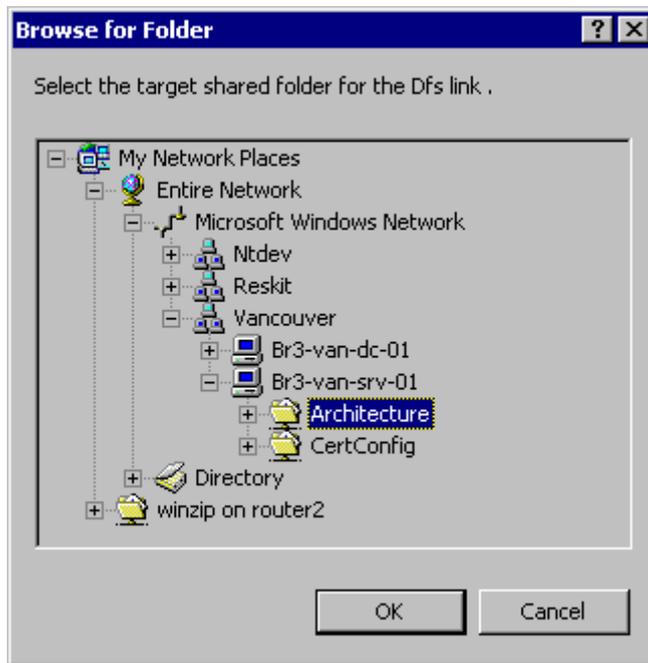

Figure 3: Browse for folder

5. Click **OK**. You can optionally specify a comment and a time-out value. The time-out value is the number of non-use seconds that individual clients have to cache the referral, after which they must retrieve a fresh referral from one of the hosting Dfs servers.
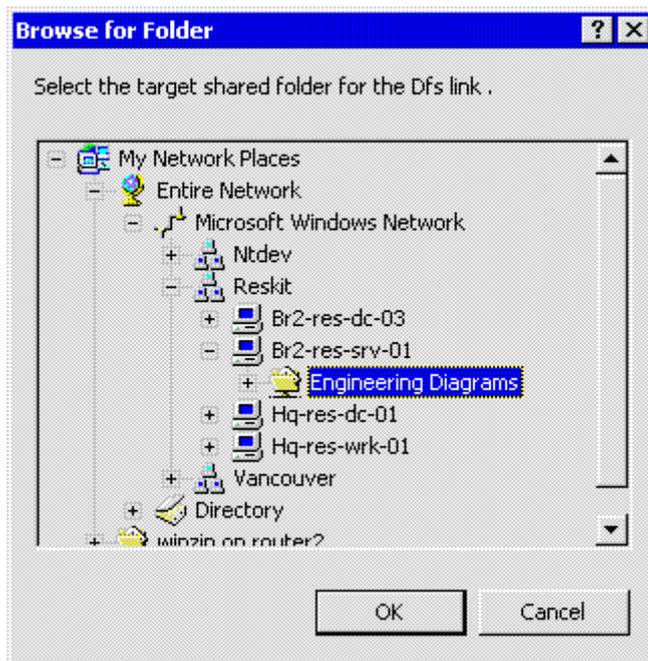

Figure 4: Add second Dfs link

6. If there are multiple servers to configure (for example, two servers host identical information, one in Hartford, the other in Seattle), you can add to this replica set. To do this, highlight the junction, right-click it, and click **New Replica**.
7. Browse to the **Reskit\BR2-RES-SRV-01\Engineering Diagrams** folder and click **OK**.
8. Click **OK** again.
9. Right-click the juncture and click **Replication Policy**. Select each shared folder and click the **Enable** button; then click **OK**.

Note: For replication to be enabled, the shares for the Dfs root or link must reside on an NTFS 5.0 formatted partition on a Windows 2000 domain controller or member server. The **Primary** flag marks the specified servers' files and folders as authoritative the first time replication takes place, after which normal multimaster replication takes place.
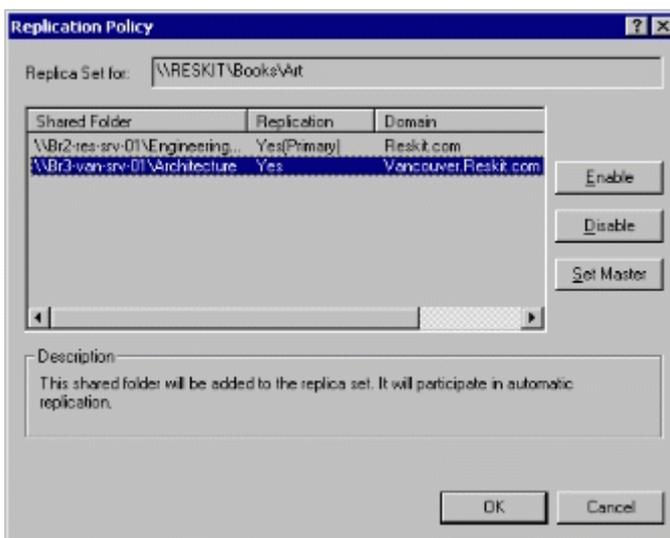

Figure 5: Replication policy

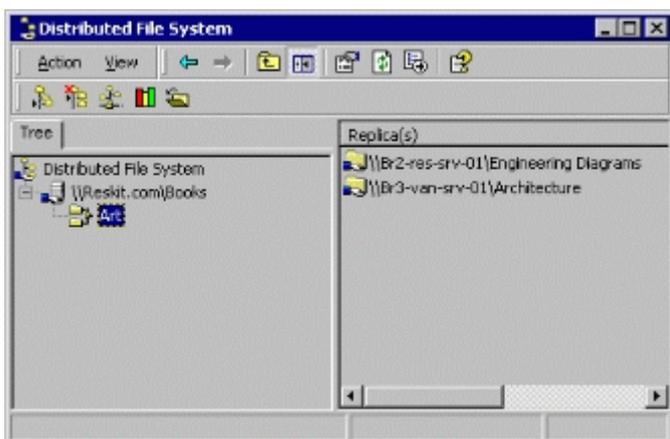The **DFS** root snap-in now looks like the one illustrated below.

Figure 6: Dfs root

Test Drive the Dfs

Any user of Windows 2000 logged on to your domain can now access the fault tolerant Dfs. Assuming they have proper access privileges, they can negotiate the individual junctions by using the following commands.

1. Click **Start**, click **Run**, type **cmd** into the Open box, and click **OK**. Then type:
2.   `NET USE driveletter: \\your domain name\your Dfs share name`

   In the example used in the document, the command would be:

   ```
   NET USE J: \\RESKIT.COM\BOOKS
    J:
    DIR
   ```

In a production environment, this alternate drive could reside on another server or on a user's workstation. Any user accessing the fault tolerant share would be able to continue to work uninterrupted. Scheduled file server maintenance, software upgrades, and other tasks that normally require taking a server off-line can now be accomplished without user disruption.

To access the Dfs root using Windows Explorer

1. Click **Start**, click **Run**, and type **\\reskit.com\books** in the **Open** box. Click **OK**.

   Click the **DFS** tab in Windows Explorer to view:

   - The list of servers backing the Dfs root or link.
   - The specific server the Dfs client is connected to.
   - The Clear History function, which flushes the Partition Knowledge Table (PKT) to obtain a new one the next time that part of the Dfs namespace is accessed.

You can also turn off one of the two servers and access the same Dfs path. This will show the failover that occurs when a server in the Dfs namespace becomes unavailable. Note that this takes place for fault tolerant roots and child nodes that are backed by more than one server.

Note: Regarding Microsoft Cluster service, at present, Dfs supports Microsoft Cluster service using machine-based Dfs only. You cannot create fault tolerant Dfs topologies on systems running Microsoft Cluster service.

Convergence

If you are using fault tolerant Dfs where multiple domain controllers exist, it is important to consider that the Dfs configuration requires time to converge between domain controllers in the domain. For immediate replication, install and use the **REPLMon** tool that is found in the **support\tools** directory of the Windows 2000 Server product CD-ROM.

Dfs-aware clients using earlier versions of the operating system (such as Microsoft Windows NT® 4.0) are not able to connect with fault tolerant Dfs roots. They can,

however, connect directly to individual Dfs roots that participate in a fault tolerant Dfs. To do this, substitute the machine name for the domain name in the above Net Use command.

Windows NT–based workstations browsing Dfs can also verify what physical storage they are referencing by viewing the **Dfs** tab available in **System Properties** found in **Windows Explorer**.

Note: Most administrative functions can be performed from the command line or scripted using DFSCMD.EXE. Type **DFSCMD /?** for online Help.
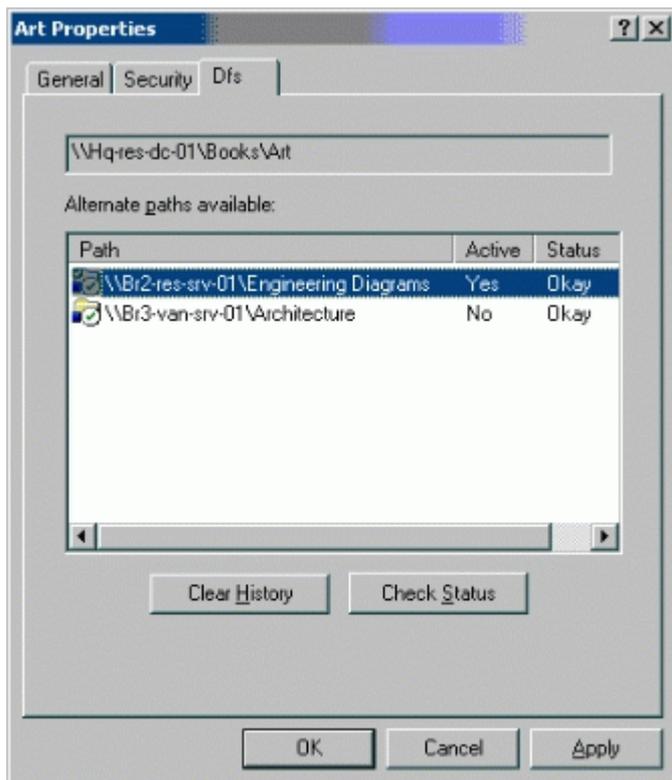

Figure 7: View Dfs properties

You can later modify the properties of this object.

You can also publish your fault tolerant Dfs root as a shared folder in the directory service, and then access it using any directory service browsing tools. From the **Active Directory Management** snap-in, select your domain, right-click **New, Volume**. Fill in the appropriate information.