

Definitely DFS

Kathy Ivens

(Reprinted from WindowsItPro Magazine)

Microsoft Dfs is slightly misnamed. Distributed File System isn't a true file system, although it sort of acts like one. Dfs lets you display to users a logical file-system structure that looks like a typical drive tree. However, the file system's components actually exist on a variety of computers in your network. Dfs is installed automatically with Windows 2000 Server, and the Dfs service starts automatically at boot-up.

Dfs use on your network can be as complex and as widespread as you want to make it. I suggest that you first set up Dfs in a lab environment (the best test setup is two Win2K Server machines and one Win2K Professional machine) and use the following information to explore the tool's possibilities. The more you experiment with Dfs, the better you'll understand its capabilities. You'll find that you can come up with creative and innovative ways to take advantage of this powerful system.

What Are the Benefits?

Dfs provides an environment in which users can easily and transparently access information from any location. With Dfs, users don't need to understand or wade through the complexities of navigating the network, and they don't need to enter complicated Uniform Naming Convention (UNC) paths to access resources on remote computers. Instead, users see a unified display of all the available network resources. This view is especially handy when your resources are in different sites or domains (UNC paths can become quite long in such situations).

For you as an administrator, using Dfs is easier than mapping drives. Not only do you avoid the risk of running out of letters, but you also can attain a higher level of consistency. For example, suppose two users, Sam and Gloria, map a particular resource to different drives on their workstations. After you set up Dfs, Sam (who mapped the resource to his G drive) and Gloria (who mapped the same resource to her F drive) both use the same name to access the resource.

As an administrator, you can change the point of origin for a resource that you've named. If a server goes down, you can substitute another server. If a server provides multiple shared resources, you can substitute another server for one of those resources—an easy-to-administer load-balancing technique. Changes are transparent to users, who never know that the source has changed and never need to adjust their program shortcuts or folder displays.

As an added advantage, Dfs might also boost your security level. Some administrators believe that Dfs enhances system security because users can't tell which servers provide which resources.

Start at the Root

Before users can access Dfs shares, you must create a Dfs root object. This object is a container that holds links to the shares and files in your Dfs configuration.

To create a root object, open the Microsoft Management Console (MMC) Distributed File System snap-in. The console displays the Distributed File System object in the scope (i.e., left-hand) panel. Right-click the object, and choose New Dfs Root to launch the New Dfs Root Wizard. Click Next to move past the introductory window.

The wizard first asks whether you want to create a domain or standalone Dfs root. You can replicate and manage a domain root through Active Directory (AD), whereas a standalone root is exactly what its name implies: a root that exists independently. You can't replicate a standalone root, and this root type doesn't have access to AD's features. Therefore, the best choice is a domain root (which is the wizard's default selection).

Definitely DFS

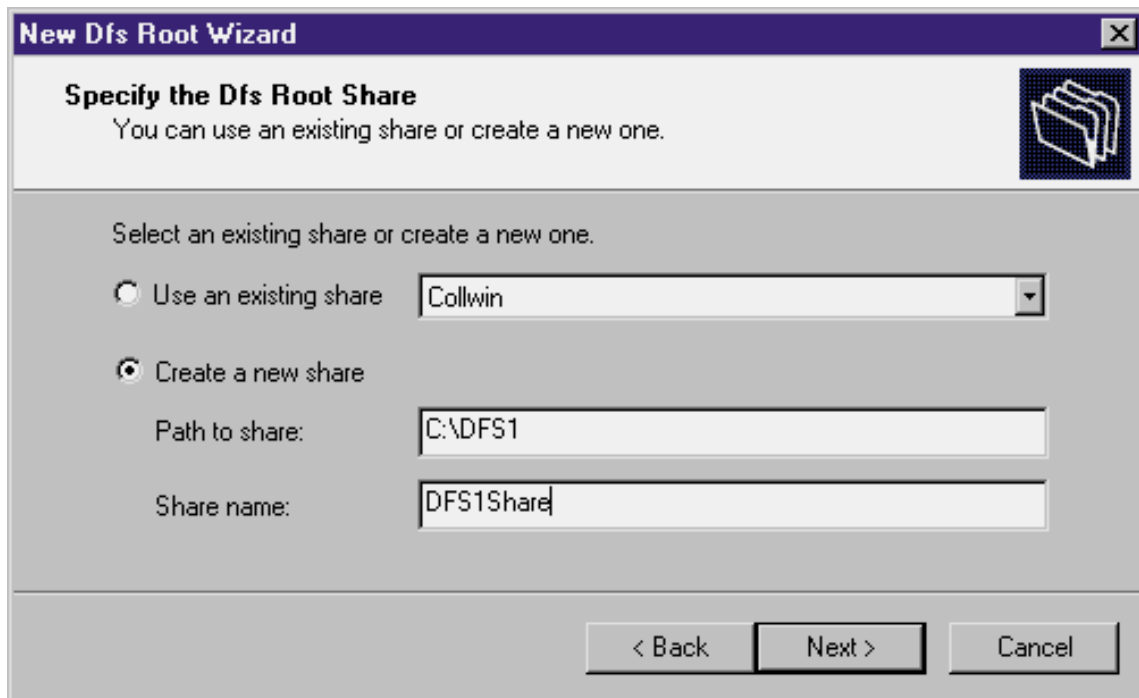
Kathy Ivens

(Reprinted from WindowsItPro Magazine)

Next, the wizard displays a list of available domains (i.e., the current domain and any trusted domains in your enterprise) and asks you to select the host domain for your Dfs root. I always use the current domain.

The next window prompts you to select a server to host the root. The wizard displays the name of the server you're working on, but you can click Browse to select another server. The best practice is to choose a server that's running NTFS. You can set up a Dfs root on a FAT system, but you lose the ability to replicate automatically, and of course you lose the security that NTFS provides.

In the next window, select or create a share to host the root. If shares exist on the server, the wizard suggests the first existing share (in alphabetical order). You can accept that share, or you can click the down arrow to the right of the Use an existing share text box and choose another share. You can also create a new share—you can even create and share a new folder. The best practice for a Dfs root is to create a new folder and share, unless you already have an empty shared folder. Figure 1 shows the process of creating a new folder and root share. When you create a new folder, the wizard asks for confirmation when you click Next. Click Yes to proceed.



The wizard next asks you to name the root and suggests a share name. I generally find the suggested name appropriate. You can optionally enter a description in the Comment field.

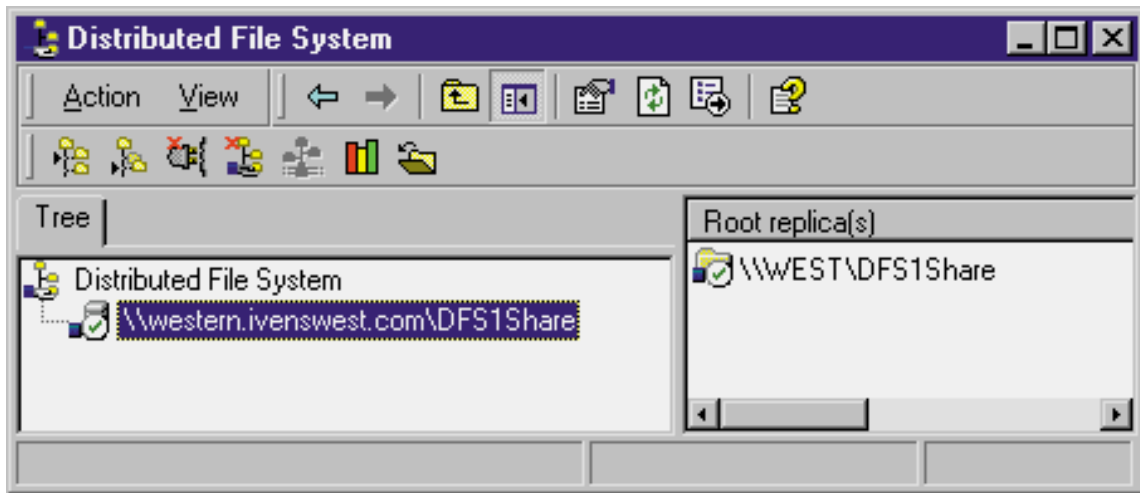
The wizard then presents a summary of the settings for your Dfs root. Click Finish if everything is correct, or click Back to return to a previous window and make any necessary corrections.

When the wizard closes, your new Dfs root appears as a node below the Distributed File System entry in the Distributed File System snap-in's scope pane. To verify that the root is functioning as you expected, right-click the Dfs root node in the Distributed File System snap-in and choose Check Status from the resulting short-cut menu. If everything is working properly, a check mark inside a circle appears on the root's icon, as Figure 2 shows.

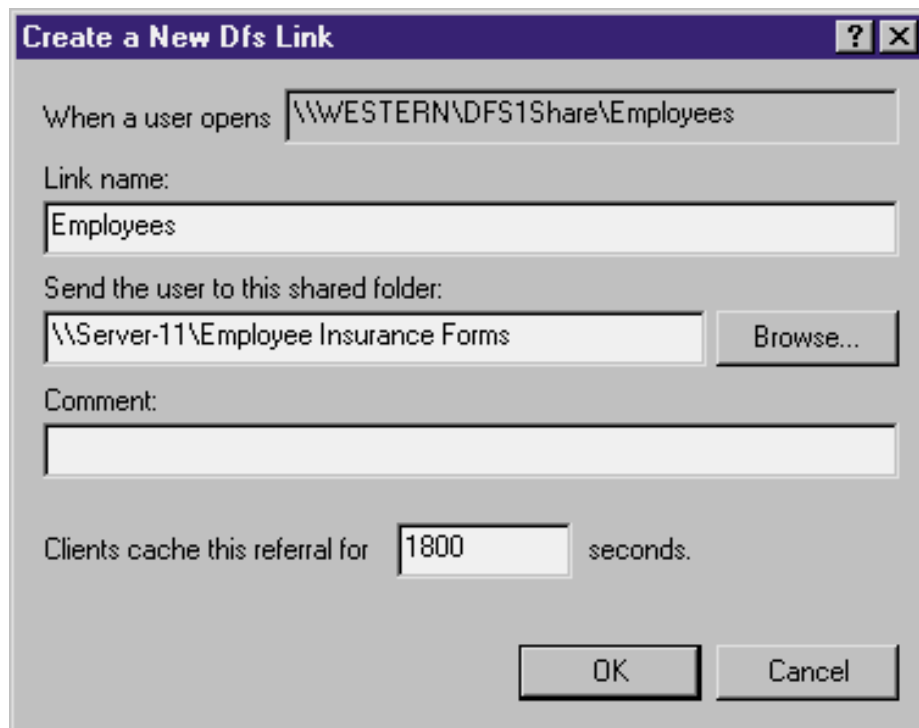
Definitely DFS

Kathy Ivens

(Reprinted from WindowsItPro Magazine)



To establish Dfs links, right-click the root object and choose New Dfs Link. In the resulting Create a New Dfs Link dialog box, which Figure 3 shows, enter the name of the link, the folder it links to, and an optional comment. Also enter a client cache duration to specify how long client systems will locally store link information. (After the cache expires, client users must manually select the link.) Repeat this process for other links that you want to locate in the Dfs root.



Just Published

You must publish the share to AD so that client computers can find it. You can publish the link to the entire domain or to an organizational unit (OU).

Definitely DFS

Kathy Ivens

(Reprinted from WindowsItPro Magazine)

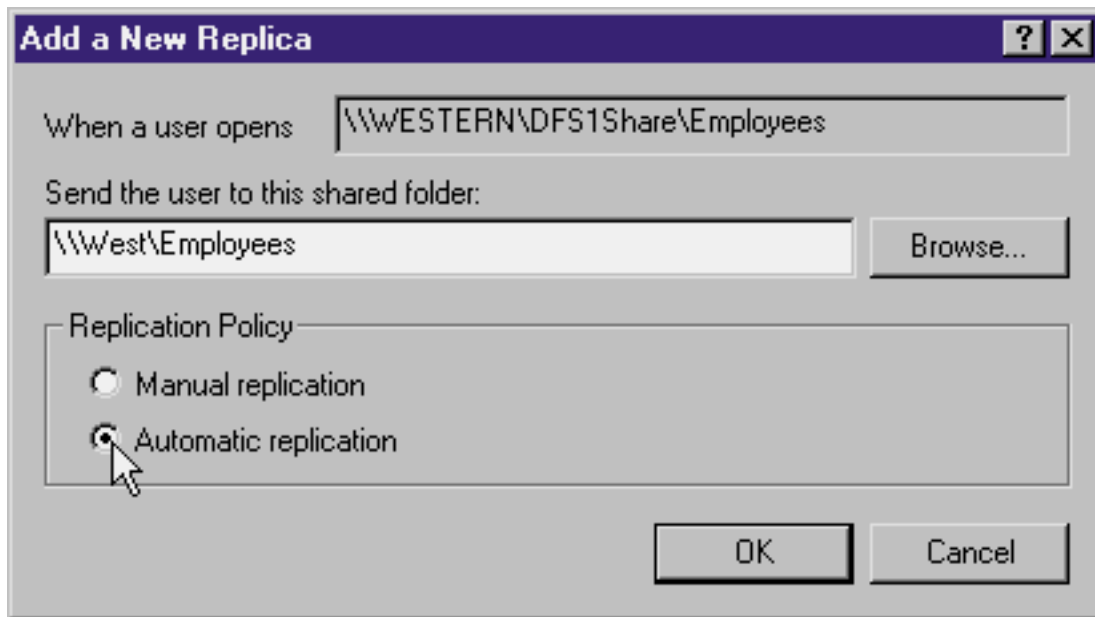
Open the MMC Active Directory Users and Computers snap-in, and right-click the domain or OU of your choice. Choose New, Shared Folder, and enter a name for the published share as well as the path (i.e., \\server\share) to the Dfs share. You can repeat these steps to publish the Dfs share in multiple OUs.

You can target Dfs links to specific user groups. For example, publish Dfs shares that hold budget documents to OUs that contain the computers in your executive and accounting departments.

Replicate, Replicate

When you create a Dfs root or link on multiple servers, you can set up replicas to provide redundancy for failover or load balancing. Automatic replication keeps folders on Win2K NTFS servers identical.

To configure replication, open the Distributed File System snap-in, right-click the link that you want to replicate, and choose New Replica. In the Add a New Replica dialog box, which Figure 4 shows, specify the location of the second shared folder (i.e., the folder that will hold the replicated link) and select a Replication Policy (i.e., manual or automatic).

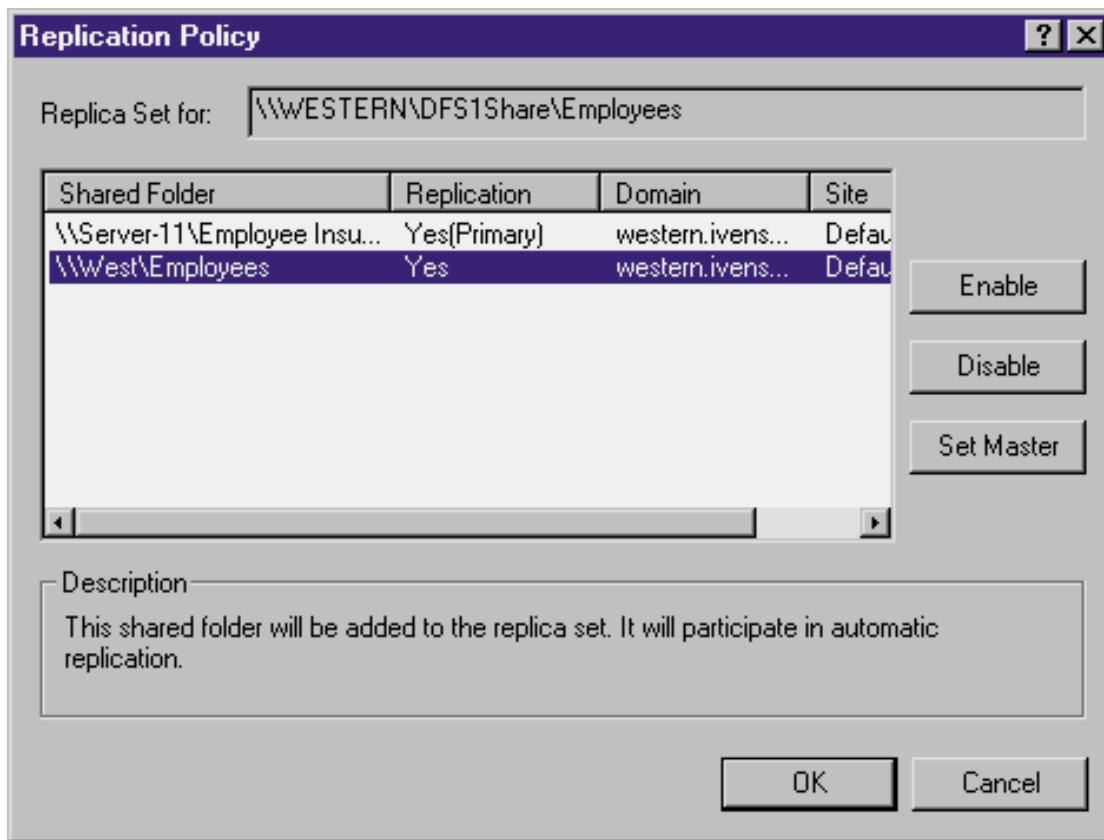


When you click OK, the Replication Policy dialog box, which Figure 5 shows, opens. Select the original folder (i.e., the one that currently holds files), and click Set Master. This action tells the File Replication Service (FRS) where to start the replication. Select the replica folder, and click Enable. You can also use this process to replicate a root object rather than a link.

Definitely DFS

Kathy Ivens

(Reprinted from WindowsItPro Magazine)



Good Management

As you add links, you build the hierarchical tree that users see. The tree looks like a single network component but is in fact made up of folders from different servers—perhaps even different Win2K sites.

You can use the Distributed File System snap-in to manage the objects in your Dfs tree. You can use the console toolbar to delete a link, take a link offline, remove a link from the namespace, add replicas, check replication status, or set replication policy.

To maintain the efficiency of your Dfs environment, keep an eye on shared folders' content changes and tweak your links as necessary. For example, if a link is frequently busy and the linked folder's contents change frequently, shorten the client cache duration to help ensure that users get the latest version of any file they access. If a folder's contents don't change often (e.g., because the folder holds only boilerplate documents), lengthen the client cache duration to reduce network traffic.

Meanwhile, Back at the Client

To access published shares, users need to open My Network Places, Entire Network, then use the Directory icon rather than the Microsoft Windows Network icon. You can, however, map a drive to the Dfs share in your users' logon scripts. The mapped drive letter won't change, even if you change the source folder's UNC path in the script.

If problems arise, you can walk a user through troubleshooting tasks. Ask the user to open the Properties dialog box for a Dfs link and go to the Dfs tab. You can then help the user check the link's status and determine which physical share the client currently uses.

Definitely DFS

Kathy Ivens

(Reprinted from WindowsItPro Magazine)

As soon as you're comfortable with Dfs, you'll find many uses for it. Besides preventing user headaches, the service offers an easy way to achieve failover and load balancing for important files. Dfs is also a great way to tidy up document storage and help users find all the documents they need. If your company's human resources (HR) documents, for example, are scattered on servers across the enterprise, you might want to use Dfs to let users access all those documents within the hierarchy of one Dfs root. This solution is easier than moving the documents, and users who already use UNC paths to get to specific files won't scream and yell—a win-win situation.