Techniques for Using the Network File System
by Wayde Schroeder

Overview
The Network File System (NFS®), a standard feature on a NeXT computer, provides access to shared file storage, which can save hard disk storage expenses. A NeXT computer offers three ways of mounting files using NFS: mounts when the system starts up, mounts by the autonfsmount command, and mounts by hand using the mount command.

Each method has its own advantages and particular use. This article explains how and when each method of creating NFS mounts should be used and gives examples of completing some common tasks with each. The examples include mounting NFS file servers, serving folders such as /NextLibrary, auto-mounting network resources such as sound or image files, and temporarily accessing file systems from another system's optical or floppy disk drive.

If you are an expert user and know what method you will be using, skip to the relevant ªExamples of Each Mount Methodº or ªTroubleshootingº sections.

Mounts Entered into the NetInfo Domain
Creating mount entries on every personal system in your network would be an

unbearable task. One feature of the NeXT NetInfo™ system, not available on other systems using NFS, is the ability to set up mount entries, for all machines in the network, from a single entry in the network NetInfo domain.

This article will not cover how to set up a NetInfo domain. The article assumes a two-level NetInfo networkÐ one top-level domain, tagged network, and a local NetInfo domain for each system, tagged local. For more information about creating and maintaining NetInfo networks, see the *Network and System Administration* manual.

Features of Each NFS Mount Method
Mounting NFS Files at Startup
Resources that must be available at all times to the user of the system should be mounted when the system starts up. This includes shared library resources, such as /NextLibrary, and local applications, such as /LocalApps. The number of remote file systems mounted when the system starts up should be kept to a minimum, because it will lengthen the time it takes for the system to start up. System files that are required for the system to start up should not be available exclusively through an NFS mount. This would inhibit the system's ability to start up if the NFS file server is down. For an example of how to avoid this situation, see the ªExamples of Each Mount Methodº section.

Advantages and Disadvantages
· Mounts are available immediately after a system starts up. Mounting the file systems

occurs when the system starts up.

- It takes longer for a system to start up, because it will mount remote files at this time.

- The NFS file server will create mounts to itself. This causes more overhead and other problems when working in the network NetInfo domain. See the ªExamples of Each Mount Methodº section.

- You may not know that the files are NFS remote files. This can be a disadvantage in perception of the speed of a local system, given that a delay may be caused by an overloaded NFS file server, or it can be an advantage in that you will not be concerned with the remote file system.

Mounting NFS Files with autonfsmount
Many files on a network will be accessed only occasionally. Examples of these types of files are: other users' home folders, source code from a previous project, and image and sound files. Files that are not used all the time, or by every user on the system, are best mounted by the autonfsmount command. This is a way to gain access to every file system in the network, without the penalty of mounting files when you don't need them.

Advantages and Disadvantages
- Because the autonfsmount command allows the computer to mount remote file

systems only when the user needs them, no resources are wasted mounting file systems that are not used.

· An NFS client NeXT computer can start up faster because it does not have to mount network files when it starts up. When a network has hundreds of file servers, this can save a lot of time.

· The autonfsmount command knows when the files are local and does not create an NFS mount to itself. Instead, a symbolic link will be created by the autonfsmount command on the local disk.

· You will know that the file systems are remote. The /Net folder will appear as a globe containing the names of the computers that are available NFS servers. The user can tell on which computer the file system physically resides. This can help you tell where a problem might be when the NFS server is down or overloaded.

Mounting NFS Files by Hand with the mount Command
The mount command may be used from a UNIX shell window. You may wish to use this feature when you need temporary access to a remote file system and don't want to create a temporary entry in the NetInfo database. An example of this is accessing files on a remote optical disk or floppy disk.

Advantages and Disadvantages

· The mount can be done without modifying the NetInfo database.

· You can select the name of the folder where the files will be mounted.

How to Use Each Mount Method
How Mounts Occur When the System Starts

When a NeXT computer starts up, it checks NetInfo for NFS file systems that should be mounted via the following command:

```
mount -vat nfs
```

The command line arguments -vat nfs specify the following options:

-t nfs	type	Mount file systems of the NFS type
-a	all	Mount all file systems in the NetInfo database with the appropriate type
-v	verbose	Display messages while mounting

To determine which file systems to mount, entries in the mounts directories of all NetInfo databases in the hierarchy will be searched. To determine whether a mount entry is of the NFS type, the name property is examined for a value that contains a

colon, as in *machinename:*/Folder. The colon indicates an NFS mount.

A typical mount entry in NetInfo could have the following properties and values in the mounts directory:

Property   *Value*

name   *machinename*:/Folder

dir   /Local_Folder

opts   rw,bg,intr,nosuid

Note: If an opts (options) value of net or noauto is used with the mount entry, the mount will not be done when the system starts up. (For more information on the mount command and its options, see the UNIX manual page for mount.)

How autonfsmount Mounts Files
When a NeXT computer starts up, the autonfsmount command is started and runs in the background via the following command:

```
/usr/etc/autonfsmount -tm 10 -tl 43200 -a /private -m /Net -fstab
```

The command line arguments -tm 10 -tl 43200 -a /private -m /Net -fstab specify the following options:

-tm 10	retry time	Seconds between retries to mount
-tl 43200	mount time	Seconds to keep file system mounted after no access
-a /private	mount folder	Folder for actual mount
-m	NIS option	Don't use the NIS auto.master database
/Net -fstab	Access folder	Folder and map option

Here is a more detailed description of these options:

-tm 10
This is the interval in seconds between retries to mount a remote file system. Here it's set to 10 seconds; the default value is 30 seconds.

-tl 43200
This is the time in seconds to keep a file system mounted after it's no longer being accessed by the user. After a remote file system has been mounted by the autonfsmount command and not accessed for 12 hours, or 43200 seconds, it will be unmounted. This way you won't have to wait for the file system to be remounted, unless you haven't accessed the remote file system for 12 hours. In most cases, the unmounting will happen overnight, and will not disturb your work. If you're accessing files on the remote file system, it won't be unmounted.

-a /private
This is the folder where the NFS file system is actually mounted. The default folder is /tmp_mnt. The actual mount is placed in /private to provide support for NetBoot clients.The autonfsmount command creates a symbolic link from where the user sees the remote file system, /Net*machinename* in this case, to /private/Net/*machinename*, where the remote file system is mounted.

-m
This option tells the autonfsmount command not to look at the auto.master map in the NIS database (formerly called Yellow Pages or YP). If this option isn't used, the NIS map will be read.

/Net -fstab
This is the folder where the autonfsmount command will run, and the option -fstab tells the autonfsmount command where to find information about what NFS file systems are available to be mounted. Any references to files in /Net will trigger (that is, activate) the autonfsmount command. The /Net folder will appear in the File Viewer as a globe to show you that the autonfsmount command is running on this folder and giving access to folders and files across the network. If the autonfsmount command is run on any other folder, that folder will also show in the File Viewer as a globe.

The -fstab option is a special autonfsmount map that NeXT has added to the standard automount command. (For more information on autonfsmount maps, see the UNIX manual page for autonfsmount.) This map option is different from other autonfsmount maps because you will see the available remote file systems under /Net before the autonfsmount command is triggered to mount the file systems (triggering means the user has requested access to files in the remote folder where the autonfsmount command is running and caused a mount to occur). This means the names of the remote machines available through the autonfsmount command will show in the File Viewer under /Net. Other maps will not show any files below the folder where the autonfsmount command is running until you trigger the mount to occur. One way to trigger the mount is by typing the name listed in the map on the keyboard. For this to work, you must know the name of the entry in the map. The -fstab option allows the user to browse through remote file systems in the Workspace File Viewer. Using any other maps or map options for autonfsmount, such as the -hosts option, would not show any machine names under the folder where the command is running until the user has triggered the mount.

How to Mount Files by Hand
When using the UNIX mount command in a shell, you will be able to mount any NFS file systems. The command used will be:

```
mount machinename:/Remote_Folder /Local_Folder
```

Looking at the two arguments in the mount command shows what it will do.

machinename:/Remote_Folder
The name of the NFS server is *machinename*. The : tells the mount command that this is an NFS mount. No other type option is needed. Remote_Folder is the folder on the NFS server that will be mounted.

/Local_Folder
This is the folder where the remote files will be mounted and where you will see the remote files in the local File Viewer. This folder doesn't have to exist on the local system before the mount. If it doesn't exist, it will be created for the duration of the mount. If it does exist, the mount will go on top of the existing folder, and any files that may be underneath will be temporarily hidden.

Examples of Each Mount Method
Mounts That Occur When the System Starts Up
The /NextLibrary folder uses a large amount of hard disk space on each system. By sharing the contents of this folder, from one system that is an NFS server, other systems on the network can remove the local copies of this folder and mount a copy through NFS. This will free about 80 MB of disk space on each NFS client for the Extended Release and about 21 MB of disk space for the Standard Release.

You need to avoid two problems with this setup. The first problem can occur on the NFS server. The most convenient way to create an NFS mount for all systems in a network is to place the mount entry in the network domain. The problem, common to all mounts of this type created in the network NetInfo domain, is that you cannot mount a local folder directly on itself. A mount such as *server:*/NextLibrary on /NextLibrary will cause the mount command to hang on the NFS server system. This is avoided by moving the /NextLibrary folder to /nextlibrary on the NFS server and using a mount entry of *server:*/nextlibrary on /NextLibrary.

The second problem applies to the /NextLibrary folder. Similar problems apply to other folders that contain files required for the system to start up, such as: /bin, /usr/lib, /private/etc, /usr/etc. The /NextLibrary/Fonts folder is required for the login process to present the login window. Without the /NextLibrary/Fonts folder, you'll see a blank screen with no login window when the system starts. To avoid this, each client keeps a local copy of the /NextLibrary/Fonts folder. This way, if the NFS mount fails for some reason, such as when the server is down or the client is removed from the network, the systems will be able to continue. When the mount occurs, the files under the mount will be temporarily hidden for the duration of the mount.

This leaves about 4 MB of disk space used, from the /NextLibrary folder, on each client system. You could save this space by creating 11 mount entries under /NextLibrary (a mount for all the folders except Fonts) instead of the single entry for /NextLibrary. This

example uses the single mount, removing all the folders in /NextLibrary except the /NextLibrary/Fonts folder on the clients. For the server, links will be created for each folder under /NextLibrary to /nextlibrary, so the server can start up without the network.

Example #1: Mounting /NextLibrary

1. *Important:* Go to the system that will be the NFS file server.

2. Log in as root on the system that will serve the /NextLibrary folder. Using the Terminal application, open a shell window.

3. Go to the File Viewer, create a new folder in the top level, and name it /nextlibrary.

4. Check that the file system is exported. The output shown here is an example:

```
server# cat /etc/exports
/nextlibrary
```

The /etc/exports file should contain a list of the folders that are exported to NFS clients. You may see the message ª/etc/exports: No such file or directory.º If you get this message, you will need to create the file /etc/exports using your favorite text editor, add the folder or folders that should be exported, and then restart the

system. If the /etc/export file exists, you can add the new folder to be exported and run the exportfs command.The output shown here is the continuation of this example:

```
server# exportfs -av
exported /nextlibrary
```

For more information about the /etc/exports file and the exportfs command, see the *Network and System Administration* manual and UNIX manual pages.

5. Open the /NextLibrary folder and select all of the folders contained in /NextLibrary.

6. Drag the icon of a hand holding several cards to the shelf.

7. Open the new folder /nextlibrary.

8. Drag the hand-holding-cards icon from the shelf into the /nextlibrary folder. This will move the folders from /NextLibrary to /nextlibrary.

9. Select all the folders in /nextlibrary and place the hand-holding-cards icon on the shelf again.

10.   Open the empty /NextLibrary folder.

11. Press the Control key and drag the hand-holding-cards icon from the shelf to the /NextLibrary folder. This will create links from /nextlibrary to /NextLibrary. Be sure to hold the Control key down during the complete dragging operation.

12.   *Important:* Go to each of the NFS clients and complete step 13. Do not perform step 13 on the NFS server.

13.   Log in as root on a system that will be an NFS client of the /NextLibrary folder. Go to the File Viewer, select /NextLibrary, and remove all the folders except Fonts. This must be done before the mount entry is created and the files are mounted from the server.

Repeat this step for each system except the NFS server in the network NetInfo domain. If this is not done on a client, the mount will occur, but you will not save disk space on that system.

14.   Create a mount for *server:*/nextlibrary on /NextLibrary in the network NetInfo domain. Use the following properties and values:

Property Value
name   *server:*/nextlibrary

dir   /NextLibrary

opts   ro,bg,intr,nosuid

15.   After creating the mount entry, restart the computer or run the command mount -vat nfs on the server and the clients.

For more information on creating mounts, see the *Network and System Administration* manual. If you are a UNIX expert and are familiar with the UNIX fstab file format, you may want to use this shortcut for creating the mount entry into the NetInfo database.

Use your favorite editor to create a temporary fstab file. The entry in the temporary fstab file would look like this:

```
server:/nextlibrary /NextLibrary nfs ro,bg,intr,nosuid 0 0
```

Then load the entry from the file into the network NetInfo database, using the command:

```
server# niload fstab .. < /tmp/fstab.tmp
```

The name of the temporary fstab is /tmp/fstab.tmp.

Check your work with the command:

```
server# nidump fstab ..
server:/nextlibrary /NextLibrary nfs ro,bg,intr,nosuid 0 0
```

You may have more mount entries in this database, but there should be one line as shown.

Mounts Done by the autonfsmount Command
In a network environment, you will find that other users have collected reference files that you would like to use, such as sound and image files. Since these files can be large, you will not want to make copies for each user. Although these files could be kept in a user's network home folder and accessed there, in this example these files will be placed in the folder /Sound_Images on an NFS server.

Example #2: Sharing Sound and Image Files
1. *Important:* Go to the system that will be the NFS file server of the sound and image files.

2. Log in as root on the NFS server. Create the /Sound_Images folder and copy the files to be served into it. You may wish to organize the files into folders like /Sound_Images/Sounds and /Sound_Images/Images.

3. Check that the file system is exported. Using the Terminal application, open a shell window. The output shown here is an example:

```
server# cat /etc/exports
/Sound_Images
```

The /etc/exports file should contain a list of the folders that are exported to NFS clients. You may see the message: ª/etc/exports: No such file or directory.º If you get this message, you will need to create the file /etc/exports using your favorite text editor, add the folder or folders that should be exported, and then restart the system. If the /etc/export file exists, you can add the new folder to be exported and run the exportfs command.The output shown here is the continuation of this example:

```
server# exportfs -av
exported /Sound_Images
```

For more information about the /etc/exports file and the exportfs command, see the *Network and System Administration* manual and the UNIX manual pages.

4. Create a mount for server:/Sound_Images on /Net in the network NetInfo domain. Use the following properties and values:

Property	*Value*
name	*server:*/Sound_Images
dir	/Net
opts	ro,bg,net,nosuid

For more information on creating mounts, see the *Network and System Administration* manual.

Mounts Done by Hand with the mount Command
This type of mount is normally used as a temporary solution to accessing remote files or for debugging exporting problems. In this example, the optical disk will first be mounted to the local file system by hand so that the disk will not be ejected when the user logs out. This is a file system mount type 4.3, not an NFS mount. Mounting the optical disk by hand also allows you to specify the folder that the optical disk is mounted onto, rather than relying on the disk label for the mount point. This way you will know the name of the folder that should be exported. The optical disk will need to be exported independently of the hard disk. Once this is done, you will go to the system that needs access to the remote optical disk and NFS mount the folder.

Example #3: Accessing Remote Optical Drives
  1. *Important:* Go to the system that will be the NFS file server of the optical disk.

2. Log in as root. Using the Terminal application, open a shell window.

3. Mount the optical disk using the following command. Do not place the optical disk in the drive before typing the mount command shown below. You will be prompted to insert the disk after the mount command is executed.

```
server# mount /dev/od0a /optical
```

4. Check that the file system has been exported. Edit the /etc/exports file and add the entry for /optical. The output shown here is an example:

```
server# cat /etc/exports
/optical
```

The /etc/exports file should contain a list of the folders that are exported to NFS clients. You may see the message: ª/etc/exports: No such file or directory.º If you get this message, you will need to create the file /etc/exports using your favorite text editor, add the folder or folders that should be exported, then restart the system, and start again at step 2. If the /etc/export file exists, you can add the new folder to be exported and run the exportfs command. The output shown here is the continuation of this example:

```
server# exportfs -av
```

```
exported /optical
```

For more information about the /etc/exports file and the exportfs command, see the *Network and System Administration* manual and UNIX manual pages.

5. *Important:* Go to each of the NFS clients and complete step 6. Do not perform step 6 on the NFS server.

6. Log in as root on the NFS client. Using the Terminal application, open a shell window.

7. Mount the remote optical disk:

    ```
    client# mount server:/optical /Optical
    ```

    Now you can access the file from the remote optical disk, under the /Optical folder.

8. When you have finished accessing the file, remove the mount from the client:

    ```
    client# umount /Optical
    ```

    Be sure you are not accessing files in /Optical and have not opened the /Optical folder in the File Viewer. If you have, you will receive the message ªDevice Busyº

when you try to unmount the /Optical. Restarting the system is another way to accomplish this task.

9. *Important:* Go to the NFS file server on which the optical disk is located.

10.   Remove the entry for the optical disk from the /etc/exports file. Edit the file, removing the line /optical. Remove the export of the optical disk:

```
server# exportfs -u /optical
```

11. Remove the mount for the optical disk:

```
server# umount /optical
```

Be sure you are not accessing files in /optical and have not opened the /optical folder in the File Viewer. If you have, you will receive the message ªDevice Busyº when you try to unmount the /optical. Restarting the system is another way to accomplish this task.

12.   Eject the optical disk.

```
server# disk -e /dev/rod0a
```

WARNING: Do not eject the optical disk while it is still mounted. If you do, you'll need to restart the system.

Troubleshooting
Problems and Solutions Mounting NFS Files When the System Starts Up

Q: *Why does my server hang after accessing a mount in the network NetInfo domain?*

A: NFS servers cannot mount a folder onto itself. This is a common experience when creating a mount in the network NetInfo domain. The bad entry would look like this:

Property	*Value*
name	*server:/Folder*
dir	*/Folder*
opts	ro,bg,intr,nosuid

This works fine for all the machines in the domain except the server system, where this mount of a local folder onto itself will hang the mount command. For more information this situation and how to avoid this problem, see Example #1.

Q: *When the system starts up, the screen is blank. Why doesn't a login window appear?*

A: This problem may be caused by a missing /NextLibrary/Fonts folder. You will have to start up in single-user mode or to log in through the network, and restore the

/NextLibrary/Fonts folder.

Problems and Solutions with the autonfsmount Command

Q: *I created a mount in /Net without the net option. Is this a problem?*

A: Yes. Never create any mount entries in the NetInfo database on the folder /Net without using the opts option value of net. The net option tells the mount command to allow the autonfsmount command to take care of these mounts. Also, never create a mount on /Net by hand or in a script run with root privileges. If you are accessing a file in /Net, make an entry with the net option in the NetInfo database and let the autonfsmount command do its job. If a mount is created in /Net by a mount method other than the autonfsmount command, the autonfsmount command will have problems when it tries to create the link in /Net that will point to the actual mount point in /private/Net. The /Net folder, or any folder where the autonfsmount command is started, and the net option always go together, and one should never be used without the other.

Q: *Can I use autonfsmount maps through NIS (Yellow Pages)?*

A: You can use NIS database maps and other autonfsmount maps and special map options shown in the UNIX manual pages for autonfsmount. However, these maps do not have the built-in feature of the -fstab special map option. The machine names or

folders listed in maps other than -fstab won't show in the File Viewer or in ls commands run in a UNIX shell, unless the machine name or folder name is accessed by the user typing the name listed in the map directly on the keyboard. Once it is accessed, the autonfsmount command has been triggered, and the mount has occurred, you will see the machine names or folders. This problem is inherent in other maps used with autonfsmount. For example, if you start the autonfsmount command on /net -hosts, running ls /net will show nothing in /net. If you run ls /net/*hostname*, where *hostname* is a host listed in the NIS hosts.byname map, you will trigger autonfsmount to mount all the file systems that are exported on that host. Now the ls command and the File Viewer will show the hostname in /net.

To use NIS maps or local maps, change how the autonfsmount command is started in the /etc/rc script. Remove the -m option if you wish the command to reference the NIS auto.master map. Add a folder name followed by an NIS map name, or the name of a map located on the local disk.

Q: *Can I use direct maps as documented in the Sun 4.1 version of the automount command?*

A: No. The current Release 2.0 version of the autonfsmount command on a NeXT computer does not support direct maps like /-.

Q: *Why do autonfsmount entries I just created sometimes not appear in the File Viewer?*

A: If you have added a new autonfsmount entry in NetInfo for a machine, all you should need to do to mount the files is update the File Viewer with the Update Viewers menu option, and click the new machine name that appears in /Net. If the machine already has an entry for autonfsmount and you are adding a new folder to the list of available folders, the new entry will show up in the File Viewer only if the machine is mounted after the entry was created. This means if the machine is already mounted, you will have to wait until the machine is unmounted by the autonfsmount command (after 12 hours of nonuse), or you must restart the system.

Q: *Can I access autonfsmount file systems in the /private/Net folder?*

A: No. The autonfsmount command won't be triggered to mount the file system and won't keep them mounted when accessed through the /private/Net folder. Once the file system is mounted by the autonfsmount command, it could be accessed in /private/Net, but this is not recommended.

Problems and Solutions Mounting NFS Files with the mount Command

Q: *Why do I get the error message ªaccess deniedº when mounting NFS file systems?*

A: This is the most common problem with unsuccessful mounts. Be sure that the file

system is properly exported from the server. This includes checking the access rights and export options.

Q: *Why does mounting a NeXT computer from Version 3.0 or older NFS releases give the error message ªaccess deniedº?*

A: Start the rpc.mountd with the -n option in the /etc/rc script. This option is required for compatibility with old releases of NFS. For more on mountd and the -n option, refer to the UNIX manual pages.