

Chapter 7 - User State Migration

When you migrate client workstations to Microsoft® Windows® 2000 Professional, an important part of that process is saving and restoring user data and settings.

This chapter addresses the specific issues related to migrating user data and settings, referred to as user state, from Microsoft® Windows® 95, Microsoft® Windows® 98, and Microsoft® Windows NT® Workstation version 4.0. It contains planning and implementation steps for saving user state and importing it to the computer after it has Windows 2000 Professional installed on it. This process is referred to as user state migration.

Because of the savings gained in maintaining user productivity and workplace morale, it is highly recommended that you save your user's state, and restore it after installing the new operating system.

In This Chapter

Planning User State Collection and Deployment

Collecting User State

Testing and Pilots

Operating System Deployment and User State Restoration

Microsoft User State Migration Tool

Planning User State Collection and Deployment

User data files are those important work files that the user needs to get their job done. Settings contain application-specific and user-specific information that, if not migrated, can cause the user to spend considerable time reconfiguring their applications after migration. Additionally, application settings supply the user with pointers and menus that can be essential for user productivity.

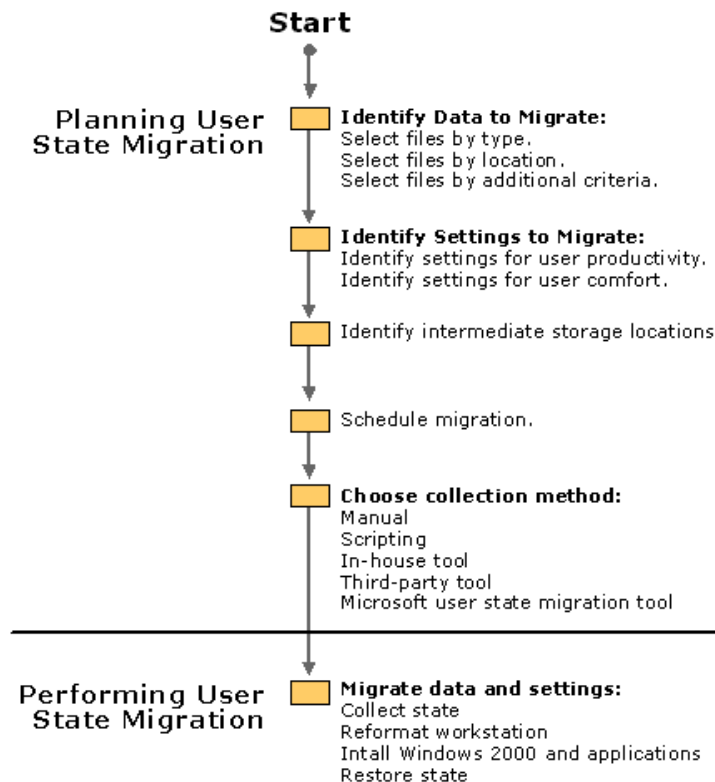
When you migrate a computer operating system from Windows 95, Windows 98, or Windows NT Workstation 4.0 to Windows 2000 Professional, and if system policy was used on the previous operating system, the persistence of registry settings, old files, drivers, dynamic-link libraries (DLLs), and such can be problematic. The client registry contains settings not created within approved Windows 2000 Professional Group Policy trees; these settings are not removed on the client when the network's domain controller is upgraded.

To avoid these problems, it is recommended that you perform a clean installation of the Microsoft® Windows® 2000 operating system on the computer, rather than doing an upgrade. This ensures that the old registry settings do not persist and cause unexpected results. Additionally, if you replace a user's computer due to hardware upgrade or system failure, user state migration is a critical step in this process. Any time you do a clean installation of Windows 2000 Professional onto a client workstation, user state migration is required.

Planning is one of the most important steps for a successful user state migration. In your migration plan:

- Prepare for data migration
- Identify the data to be migrated.
- Identify the settings to be migrated.
- Determine where and how the data will be stored during migration.
- Schedule migration.
- Identify the best way to collect user state data.
- Determine the best way to restore user state data.

Figure 7.1 illustrates these tasks.



If your browser does not support inline frames, [click here](#) to view on a separate page.

Figure 7.1 User State Migration Process

Preparing for Data Migration

For migration to go smoothly, prepare and backup user files, and handle any issues involved in programs that use synchronization mechanisms (such as e-mail). The time directly prior to migration is a good time for users to get their files into a stable state. For example, if version control software is used, it is recommended that user's check in all the files they have checked out. If all in-progress documents are supposed to be saved to a specific network folder (such as F:\EngineeringDrafts), make sure the users save all relevant files to that folder.

While not part of the migration itself, backing up user data files is always important when making significant changes to a workstation. If user data is not collected during migration, it can still be retrieved from the backup. If the user state collection retrieves data only within a known folder (for example, My Documents), users need to move their important documents to that folder. If the folder is a network share, no migration of the files is needed as long as the migrated system also has access to the share.

Best Practice It is a good idea to search the drive for important files that the user might have missed. These can often be located by looking for specific file types (like DOC files).

It is recommended that the user send all e-mail prior to migration. If the user uses mail synchronization between two mail stores, mail synchronization files might not migrate well. This is not a consideration if the server has been updated with the user's latest e-mail messages, or if the user stores e-mail only on the server or only on the workstation. It is an issue only if mail is stored in both places and synchronization is used. Along with e-mail, My Briefcase, Outlook, Notes, and any other feature that employs synchronization must be synchronized prior to the migration.

Identifying Data to Migrate

When identifying what data to migrate, consider these questions:

- What data files do the users work with regularly?
- Where do the users save their files? Are they located in a single folder, or scattered across a drive?

One way to determine which folders to migrate is by known locations. These can be locations that the system is aware of, like My Documents and Favorites, or company-specified locations, such as \EngineeringDrafts.

Another way to discover which files to migrate is to identify the applications the users use and then look for files with corresponding file types. An e-mail package and a productivity suite like Microsoft® Office® are common. These applications normally use specified file name extensions. For example, Microsoft® Word primarily uses .doc file name extension. However, it also uses other file types, such as templates (DOT files), on a less frequent basis.

Create a list of important file types based on common applications used in your organization. A good starting point for finding files types to include is to look at the registered file types on the desktop image. The list of registered file types is found in Folder Options.

To find the registered file types

1. Double-click the **My Computer** icon on the desktop.
2. Click the **Tools** menu.
3. Click Folder Options.
4. Click the **File Types** tab to display the registered files types, as shown in Figure 7.2.

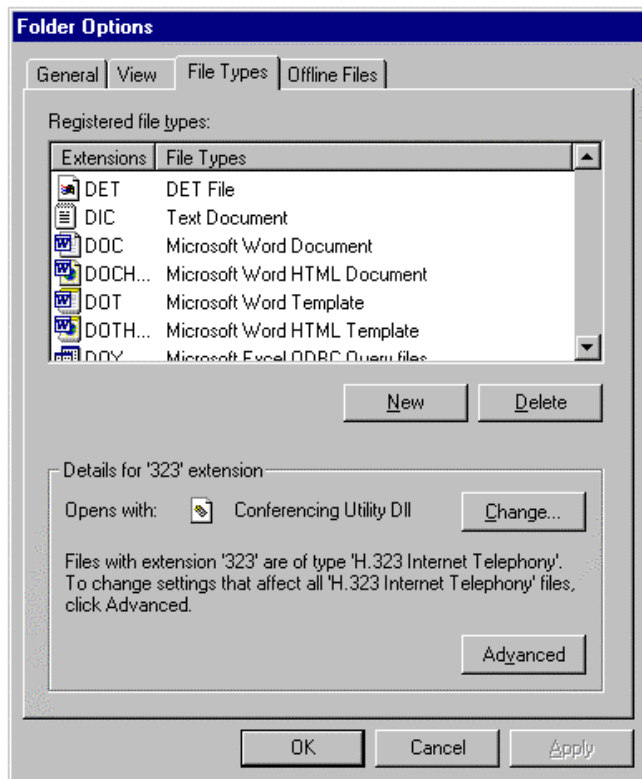


Figure 7.2 Registered File Types

Best Practice Do not migrate the actual applications; it is best to reinstall them from a software distribution point. For more about software distribution points, see "Improving Software Installation in Your IT Infrastructure," in this book.

Identify Settings to Migrate

Consider the following questions when identifying which settings to migrate:

- Which settings do the users need so they can work?

- Which settings make the work environment more comfortable, allowing users to be productive?
- Which settings reduce support desk calls after the migration?

Note If you are migrating a Windows NT 4.0 user profile to a Windows 2000 Professional environment and have been using system policy, be aware that persistence of system policy settings in the user registry hive and nonstandard access controls on the user registry can cause unexpected results. You might prefer to recreate the profile, rather than try to migrate it and sort out the potential problems.

User Settings for Productivity

Create a list of the important settings the user needs to become productive immediately after the migration. These settings might include an e-mail server and account, remote access connections, accessibility features, and Internet connectivity. One good place to find the relevant settings is in your organization's new employee system configuration handbook.

Locating application-specific settings can be time consuming, because various applications store settings in different locations. Therefore, limit this list to only those settings that the user must have to be productive.

Some applications, such as those in the Microsoft® Office® office suite, provide tools that scan the registry and then display settings and their storage location in an easy-to-read format. Other applications require you to compare registry entries before and after an installation to trace the settings. User settings are usually stored in HKEY_CURRENT_USER \Software<Company Name>\<Application Name>. To compare before and after images of the registry to find these entries, you can use the tool Sysdiff.exe, available on the Web resources page at <http://windows.microsoft.com/windows2000/reskit/webresources>.

Caution Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows 2000 Professional. To configure or customize Windows 2000 Professional, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

It is not always cost-effective to collect and restore all user-specific registry settings. To determine the cost-effectiveness of collecting settings:

- Learn the number of users to be migrated.
- Multiply this number by the time it will take the user to reconfigure these settings.
- Multiply this number by the users' average hourly wage to determine the cost of not migrating the users' application settings.

For example, if you are migrating 100 users and you estimate that it will take each user four hours to reconfigure their settings (if the settings are not migrated), and the average hourly wage for these employees is \$15, your formula would be: $100 \times 4 \times 15 = \$6,000$. Compare this figure to the costs involved in tracing, collecting, and restoring the settings.

User Settings for User Comfort

While the value of doing so might not be immediately apparent, it is always worthwhile to capture the settings that make the user feel comfortable with their system. Desktop settings, display settings, and folder options are common examples of these settings. While you might not see these settings as essential, if these settings are not migrated, users will spend time setting their environment the way they prefer it. Productivity is lost while the user adjusts these settings.

Migrating these settings makes the desktop familiar to the user, which helps them feel more comfortable with the new system, and to learn about it more easily. Not migrating these settings results in lost employee productivity time, lower employee satisfaction with the migration, and a potential help desk calls.

Use discretion when deciding which settings to migrate. For example, while it is important to migrate some settings for user comfort (for example, their wallpaper), it is less important to migrate things like the pixel size of their toolbar.

Some settings might require moving a specific file. For example, if the desktop background setting is a photo of the user's child either move the file during the migration, or inform the user that they will have to replace this image after migration. Microsoft now offers a User State Migration Tool (USMT). To download it, see the User State Migration Tool link on the Web Resources page at <http://windows.microsoft.com/windows2000/reskit/webresources>.

If you will not migrate these types of settings, make that clear when you announce the migration. Set expectations early and clearly to reduce user frustration and help desk calls.

Determining Where Data Is Stored During Migration

Set aside a resource for storing user data and settings during the migration period. The section below helps you determine how much disk space you need for an intermediate storage location. Key things to consider are:

- E-mail: If the user deals with a large amount of e-mail, or keeps e-mail on their local computer instead of on a mail server, this e-mail will often take up as much disk space as all of their other files combined — easily amounting to over 100 MB. (Note: This is not a factor if their e-mail is stored on a server only.)
- User documents: Frequently, all of a user's documents fit into 50 MB of space, depending on the types of files they work with. This estimate assumes typical office work such as word processing documents and spreadsheets.
- User system settings: 5 MB is usually adequate to save these registry settings.

Considering the above factors, and based on typical office work, Table 7.1 gives some general guidelines. Check this against your own environment to produce a realistic picture of your own intermediate storage requirements. Additional factors, such as document types, make a substantial difference in these calculations. For example, an architectural firm where Computer-Aided Design (CAD) files are a common file type needs much more space than a law firm dealing primarily with word processing documents. Of course, if your employees already store many of their documents on file servers, much less space is needed for migration.

Table 7.1 Disk Space Requirements for Intermediate Stores

User Type	Rough Storage Estimate
Desktop user storing e-mail on server	50 MB to 75 MB
Desktop user with local e-mail storage	150 MB to 200 MB
Laptop user	150 MB to 300+ MB

- **Best Practice** Allow a minimum buffer of 20 percent additional space on the intermediate store. For performance sake, put the intermediate store on high-speed drives. Ensure that this is the only task the store is performing and that it has an optimized (high-speed) network connection.

Scheduling Migration

Migration is best done by moving groups of users in phases. Migrate a group or team, and make sure their migration is successful before starting the next group. This gives you a chance to make any necessary modifications to your plan between groups.

Depending on available resources, it is generally best to not migrate more than 100 people at a time. If you calculate that each user is going to have at least 50 MB of state information, and you are migrating 100 users at a time, your plan must include the storage of 5 GB of user state data on an intermediate data store. This user state data needs to be copied to the intermediate store before the operating system deployment, and then copied back afterwards. This copying process can have an impact on network bandwidth. Migrating 1,000 users at once using these parameters requires 50 GB of intermediate data storage, and could have a significant impact on network performance for all users. Phasing keeps network resource requirements (bandwidth and hard disk storage) to a manageable size.

Collecting User State

After you have identified which data and settings will be migrated and where they are located, decide your method for collecting the user state from the workstations. Collection can be done using:

- Manual collection
- Scripting files
- In-house tools
- Third-party tools
- Microsoft User State Migration Tool (USMT)

Some benefits and costs of each collection method are described in the following sections. Use this information to identify which method, or combination of methods, best suits your organization. Factors to consider include the size of your organization, the level of desktop management in place, the number of users to be migrated, the randomness of file locations on workstations, the type of technical personnel you have available for the migration, and the amount of time you can dedicate to this process.

Collecting User State Manually

Most organizations today collect user state data manually. This procedure requires a technician to visit each machine that is to be migrated. When data is set up along expected guidelines (for example, all documents are stored in My Documents), manual collection can be efficient. When using manual collection methods, settings are not typically migrated. The technician simply helps the user set up their new machine.

However, consider that when collecting user state manually, important files and settings can be missed when users disperse data throughout local drives. Treatment of special cases often requires the user or technician to be aware of additional data locations and have extra time to deal with it. These special cases extend the amount of time the migration takes. Users' experience from migration to migration tends to be inconsistent, because different technicians might have more or less time and knowledge to deal with specific issues. This inconsistency is likely to increase help desk calls, and increase problems due to random human error.

Labor costs in this choice can be very high; therefore, it is often beneficial to combine manual collection with other, automated scripts or tools.

Collecting User State Using Scripting

Scripting can significantly reduce the cost of migration. If you combine scripting with a manual approach, you reduce the costs of an exclusively manual approach while providing the flexibility to handle special situations. With a script file, a technician does not need to spend significant time at each machine. The expense for each additional migrated user is far less than with manual collection. Development costs for a script file are usually fairly low, and most medium to large organizations have in-house personnel who can write such a script file. This combination lends itself to a consistent experience for the users.

However, this method can only handle the cases that are specified in the script. Users frequently have to spend more time getting their systems ready for the migration. If they make mistakes in preparation (for example, forgetting to move a file to a designated folder), they are likely to lose that data.

Collecting User State Using In-House Tools

Some large organizations develop their own in-house migration tool. This frequently gives the best migration experience for the user, because it is customized to the specific environment. Fixes can be made quickly when new applications are adopted or bugs are found. Such a tool can capture user settings and files, and restore them. To see an option for building such a tool, see the Windows Script Host link on the Web Resources page at <http://windows.microsoft.com/windows2000/reskit/webresources>.

This option usually requires a significant investment in development time. If you do not have in-house personnel who can create this tool, it is probably too expensive to hire programmers to do this. Even if you do have in-house programmers, compare the cost of tool development with other, less expensive methods.

Collecting State Using Third Party Tools

Currently available third-party tools can handle most needed settings, and are extensible for in-house application settings. These tools often provide multiple types of rules to specify which files to migrate. While fairly thorough, such tools are not targeted to your specific environment.

Initial cost for this option is often expensive. There is usually a per-seat fee, so the cost of the migration increases as you migrate more users.

Collecting State Using the Microsoft User State Migration Tool (USMT)

Microsoft now offers a User State Migration Tool (USMT) that you can download from the Web Resources page at <http://windows.microsoft.com/windows2000/reskit/webresources>.

The USMT offers multiple rules for including various file types and settings and is easily customized. The USMT does not include a per-seat fee, so migration costs do not escalate with additional users.

The USMT is a command-line tool with no graphical user interface. By default, USMT collects only a limited group of data and settings. Expect to customize this default set, which requires technical personnel with knowledge about the registry.

For more information about USMT, see "Microsoft User State Migration Tool" later in this chapter.

Testing and Pilots

After you have identified which data and settings to migrate and determined the best methods for collecting user state data, set up a test environment in a lab and test your migration method. You need a domain controller running Microsoft® Windows® 2000 Server or Microsoft® Windows® 2000 Advanced Server, and at least one workstation for each type of operating system from which you want to migrate. For example, if you are migrating some workstations from Windows 98 and some workstations from Windows NT Workstation 4.0, you need to test at least one workstation for each.

After you have thoroughly tested your collection, operating system deployment, and restoration processes, conduct a pilot with a small group of users. If unexpected problems arise, you can address them before going any further. Make sure that all data and settings have migrated as expected. Only after you are fully satisfied with the success of your pilot migration should you begin a full migration.

The pilot also gives you an opportunity to test your space estimates for intermediate data storage. After you have migrated a few typical user states to a server, note the space required and adjust your initial calculations accordingly.

User Education

Prior to migration, set user expectations to match the results you learned obtained from your pilot. If you will move all user data into My Documents, explain this in advance so they can find their files. If you are not migrating particular settings, tell them in advance so they know what to expect. The more closely you can mirror their previous environment, the happier they will be with the migration, the less support they need, and the sooner they can resume productivity.

Operating System Deployment and User State Restoration

After user state has been collected, the easiest and quickest way to deploy Windows 2000 Professional is to use disk imaging and broadcast technology, such as Ghost, SysPrep, or Remote Installation Service (RIS). For more information about RIS, see "Deploying and Using Remote OS Installation" in this book. Your deployment plan should include some method of matching each saved user state to the appropriate computer. For example, some computers store their serial number in a manner accessible by software; you can also use the MAC address on a computer's network interface card as an identifier. If neither of these schemes works for you, use your network map to match user state to its appropriate recipient computer.

After the new operating system and any needed applications have been installed, the user profile and its associated directories need to be created on the target computer, and write the stored user state data to it. Simply having the user log on creates the user profile. If Group Policy is used for application distribution, the applications assigned to the user are also installed at this point. For more information about application distribution, see "Improving Software Installation in Your IT Structure" in this book.

Security and File Relocation Considerations

Migration is a good opportunity to move towards a low-cost desktop by moving the user's data files into their My Documents folder. For a discussion about lowering the total cost of ownership of workstations using IntelliMirror™ technologies, see "Meeting Your Desktop Management Needs with Windows 2000 Professional Change and Configuration Management".

When moving to a more highly managed environment (for example, limiting the user to storing files in the My Documents folder), the user's access rights change. Problems can occur if user data files have not been relocated to the user's My Documents folder. Files stored in folders the user can no longer access are essentially lost to that user. If you are moving to a secured desktop, move user data files to their My Documents folder. Maintain the folder hierarchy within My Documents so the user can still easily find their files.

When migrating a Windows NT workstation with an NTFS drive, files can have individual access control lists (ACLs). These ACLs often do not migrate with the files. Instead, the files will assume the default ACLs of the folder to which they are copied.

Best Practice If there are multiple users on the same computer who share files in common folders, relocate the common folders to a network share and assign the appropriate access permissions.

If the computer hard drive configuration was changed, you might not be able to restore files to exactly the same drive or directory structure as that from which they were collected. For example, if you have replaced two small drives with one large drive, the second drive is not available to receive user data. In this case, the files have to be relocated.

When files are relocated, they can end up in a folder that already has a file with the same name. This causes a name conflict. The best way to handle this problem is to rename the conflicting file. It is important when renaming the file to not lose the meaning of the original name. When long file names are allowed, the best way to do this is to add #1 to the end of the file name. Increment the 1 to a 2 if another file collides, and so on. For example, if you have three files named example.doc, name them as example.doc, example#1.doc, example#2.doc. (Note: Microsoft User State Migration Tool automatically handles file name renaming as explained here.)

Do the same for folder names that conflict. Do not simply combine the two folders, as this can cause files to be mixed together inappropriately. Cleaning up from such a combination is often very complex, and it is much easier for the user to combine the folders later if that's what they want.

One way to avoid name collision is to include as much of the original path as possible in the new location. For example, if the full path of the original file and file name was D:\EngineeringDrafts\example.doc, but the new root location is C:\Documents and Settings\\My Documents, the new full path and file name should be C:\Documents and Settings\\My Documents\EngineeringDrafts\example.doc.

File relocation can also cause problems with Most Recently Used lists (MRUs) and shortcuts to documents. If the target of the shortcut or MRU has moved, the shortcut no longer works. Fixing the MRUs and shortcuts requires retargeting them to the new location of the document. Be sure to keep a list of all moved files and checking each MRU and shortcut.

Best Practice The best solution to this is not to move MRUs or shortcuts. Instead, have the user recreate their own shortcuts and MRUs. If you do move them, educate the user so that they understand why some of them might no longer work.

The enhanced security of Windows 2000 Professional can mean that registry settings that had been accessible with Windows 95 or Windows 98 are no longer accessible. When running Windows 2000 Professional with nonadministrative, nonpower user permissions, users have write access only to three locations: HKEY_CURRENT_USER, User Profile, and a shared documents location. If there are settings outside this hive, they cannot change those settings without Administrator access. The user might be able to simply run the application to import all their old settings if the settings are saved in a REG file. Repackaging the application into a Windows Installer package and then assigning that package to the user, enables the settings to be installed. (For more information about Windows Installer packages, see "Improving Software Installation in Your IT Structure" in this book.) An easier way is to have someone with Administrator access (or a program running with that security context) restore the settings.

Applications that write settings outside of HKEY_CURRENT_USER need to be handled on a case-by-case basis. Sometimes it is acceptable to change the access on a particular registry hive, while other times this can cause unacceptable registry access security issues. The best solution is to work with the software vendor to determine when a revised version of the application will be available.

Installing or Upgrading Over User Settings

If the application to which the user settings are related is not installed until after user state migration, it is probable that the user settings will be overwritten. While some applications check whether user settings already exist and preserve them, this is not common. Upgrading to a new version of an application as part of the migration can also cause user settings to be lost.

To avoid overwriting user settings, do one of the following (listed in order of preference):

- Create and deploy a desktop image composed of the new Windows 2000 Professional operating system and your common applications.
- Install the applications separately from the operating system, before restoring the user state.
- If you are upgrading the application as part of the migration, upgrade the application before collecting the user state and installing the new operating system.

- Install the new operating system, restore user state, and then upgrade the application, so that user-specified application settings are retained.

Translating and Relocating Settings

Because Windows 98 and Windows 2000 Professional do not represent Display Font information the same way in the registry, most settings must be translated during migration, while others need relocation.

Simply copying the HKEY_CURRENT_USER \Control Panel\Desktop\WindowsMetrics subkey during migration causes problems because entries, such as **IconFont** are not correctly translated. In general, leave entries like this behind or have code written to translate them. Translation code is usually not cost effective. USMT handles most operating system and shell translation issues. For in-house applications, review and experiment to determine the best way to achieve the desired results.

Most subkeys and entries are in different locations in all versions earlier to Windows 2000 Professional. They do not need translation — they need relocation. Desktop icons are a good example of this. The My Computer icon is stored in the value of the HKEY_CURRENT_USER \Software \Classes \CSIDL\{20D04FE0-3AEA-1069-A2D8-08002B30309D}**DefaultIcon (Default)** entry in Windows 98, but is stored in the value of the HKEY_CURRENT_USER \Software \Microsoft \Windows \CurrentVersion \Explorer \CLSID {20D04FE0-3AEA-1069-A2D8-08002B30309D}**DefaultIcon (Default)** entry in Windows 2000 Professional. Tools like Sysdiff.exe, included with the Windows 2000 Professional Resource Kits, can be used to compare before and after images of the registry to find these entries.

Microsoft User State Migration Tool

The Microsoft User State Migration Tool consists of two executable files, ScanState.exe, LoadState.exe, and three migration rule information files Applet.inf, Usermig.inf, and Sysfiles.inf. ScanState.exe collects user data and settings based on the information contained in Applet.inf and Sysfiles.inf. LoadState.exe deposits this user state data on a computer running a fresh (not upgraded) installation of Windows 2000 Professional based on the information contained in usermig.inf. Additional INF files can be created for additional migration requirements. Office.inf is included as an example of one of these additional migration INF files — it handles some of the settings for Office.

With no modification to its default settings, the tool:

- Migrates the user's system settings including:
 - Dial-up networking or remote access
 - Display settings (wallpapers, colors, fonts)
 - Microsoft® Internet Explorer version 4.0 or Microsoft® Internet Explorer 5 settings
 - Accessibility settings (user must re-start computer for these settings to take effect)
 - Network shares
 - Folder options
- Migrates the user's Microsoft® Outlook® 98, Microsoft® Outlook® 97, or Microsoft® Outlook® 2000 MAPI settings.
- Migrates some of the common user document types to My Documents.

It is easy to modify what is included in the state that ScanState.exe collects. The tool can be instructed to collect or leave specified files, folders, registry entries, or registry subtrees. (For rules on modifying the scripts, see "Migration Rule INF File" later in this chapter.)

Table 7.2 User State Migration Tool Requirements

Type of system	Requirements
Source system	Windows 95, Windows 98, or Windows NT Workstation 4.0 Access to the intermediate store
Intermediate store	Storage based on data to be migrated. (50 MB is often enough for typical users, but this can vary.)
Target system	Windows 2000 Professional Access to the intermediate store Appropriate amount of drive space to receive the user state data

Using the USMT

In the simplest case, the default migration rule INF file (Sysfiles.inf) is used as is for the migration. The administrator must create a script run on the client workstation. The script can: must:

- Run ScanState.exe on the client workstation, copying the user state data to an intermediate store. You can send a shortcut to the script to the users and instruct them to run this program when they leave for the evening, or you can deploy the script automatically or on a schedule.
- Reformat the disk and install Windows 2000 Professional and applications as needed. This can be automated by using disk-imaging software.
- Run LoadState.exe as the local administrator on the client workstation to restore the user settings. (This can be done as a scheduled task running in the local administrator context.)
When the user logs on, the last of their user state data is reset.

If you want to modify the default parameters in the Sysfiles.inf INF file, the steps are as follows:

- Identify which application settings to migrate.
- Identify which file types, folders, or specific files to migrate.
- Locate these settings and files and create a Migration Rule INF file for them.
- Run ScanState.exe, copying the user state data to an intermediate store. You can send a shortcut to the script to the users and instruct them to run it when they leave for the evening, or you can deploy the script automatically or on a schedule.
- Reformat the disk and install Windows 2000 Professional and any necessary applications. This can be automated by using disk-imaging software.
- Run LoadState.exe as administrator to restore the user settings. (You can do this as a scheduled task running in the administrator context.)
When the user logs on, the last of their user state data is reset.

The remainder of this chapter explains the syntax and usage of the User State Migration Tool.

ScanState.exe Syntax

This section explains the syntax and usage of ScanState.exe.

```
scanstate [/c /i input.inf]* [/l scanstate.log] [/v verbosity_level] [/f] [/u] [/x] migration_path
```

Where:

Flag	Resulting Action
/c	continues past filename_ too_long errors. Log files in Longfile.log.
/i	input.inf specifies an INF file containing rules that define which user state data to collect. Multiple INF files can be specified.
/l	log file specifies a file in which to log errors.
/v	enables verbose output. The syntax is /v # where # is 1 to 7, with 1 being the least verbose, and 7 the most.
/u	specifies that user settings will be migrated. This is a switch for troubleshooting only.
/f	specifies that files will be migrated. This is a switch for troubleshooting only.
	migration_path is a path to the location where files should be read/written.

Example:

```
scanstate \\filesrv\migration\elizabeth /i .\applet.inf /i .\sysfiles.inf /i .\files.inf
```

By default, user settings, system settings, and files are migrated. Use the /u, /f, and /x flags only for troubleshooting. Multiple Migrate Rule INF files can be specified. If you specify more than one, the Sysfiles.inf Migration Rule INF file must be included.

LoadState.exe Syntax

This section explains the syntax and usage of LoadState.exe.

```
loadstate [/i input.inf]* [/l loadstate.log] [/v #] [/f] [/u] [/x] migration_path
```

Where:

Flag	Resulting Action
/i	input.inf specifies an INF file containing rules to define what state to migrate. Multiple INF files can be specified.
/l	log file specifies a file in which to log errors.
/v	enables verbose output. The syntax is /v # where # is 1 to 7, with 1 being the least verbose, and 7 the most.
/x	specifies that no files or settings will be migrated. This is a switch for troubleshooting only.
/u	specifies that user settings will be migrated. This is a switch for troubleshooting only.
/f	specifies that files will be migrated. This is a switch for troubleshooting only.
	migration_path is a path to the location from which the files should be read.

Example:

```
loadstate \\filesrv\migration\elizabeth /i .\usermig.inf
```

By default, user settings, system settings, and files are migrated. The /u, /f, and /x flags should only be used for troubleshooting. Multiple migrate rule INF files can be specified.

Migration Rule INF File

This section explains the rules and syntax for Migration Rule INF files (Applet.inf, Usermig.inf, and Sysfiles.inf). It is recommended that only administrators who are familiar with scripting and the registry attempt to modify the default Migration Rule INF file. Thorough test the INF file after any modifications.

Special Folder and Environment Variable Support

The Migration Rule INF file supports path substitutions for the special folders defined by the CSIDL value set. When the name of the CSIDL is inserted between percent marks in the INF file, (for example %CSIDL_SYSTEM% or %CSIDL_PERSONAL%), USMT uses them. Any environment variables are also supported if the user and administrator define them the same.

Example:

```
%CSIDL_PERSONAL%\*.txt would match C:\Documents and Settings
```

File Include Rules

Additional files and folders can be included in the migration by adding a **CopyFiles** label to the **[Copy This State]** section of a Migration Rule INF file. The **CopyFiles** label is followed by a section name. The section contains the files and folders to be migrated to the destination machine.

Example:

```
[Copy This State]
CopyFiles=Copy Trip Reports
[Copy Trip Reports]
; The next line copies the TripRpts folder non-recursing
DIR=C:\TripRpts
; The next line copies everything in the TrpRpts folder-recursing ONLY
DIR=C:\TripRpts\*
; The next line copies all files on the user's desktop in a folder called TripRpts
; matching the 99*.DOC wildcard search.
%CSIDL_DESKTOP%\TripRpts\99*.DOC
```

File Exclude Rules

If you do not want to migrate some files or folders, add a **DelFiles** label to the **[Copy This State]** section of a Migration Rule INF file to have those files and folders excluded from the migration. The **DelFiles** label is followed by a section name. The section contains the files and folders that must not be copied to the destination. If ScanState.exe was originally going to copy the specified items, now it does not. Note that the file and folder specified in this section are not actually deleted from either the original or destination computer.

Example:

```
[Copy This State]
DelFiles=Trip Reports DelFiles
[Trip Reports DelFiles]
; We don't want to migrate the July '99 trip report
%CSIDL_DEKSTOP%\TripRpts\9907*.DOC
```

Write to a Different Location Than Read

Often migration is used to move towards a more managed desktop. To this end, administrators want to move many of the user's documents into the My Documents folder, CSIDL_PERSONAL. This is accomplished by adding the destination folder on the target computer to the end of a CopyFiles section. This new folder is used as the root for the files. You can also copy to folders other than "My Documents" by re-rooting to the destination folder. Direct naming (such as C:\DATA) is supported. For more information about moving files, see "Registry Move and Copy Rules" later in this chapter.

Example:

```
[Copy This State]
CopyFiles=Copy Trip Reports
[Copy Trip Reports]
; The next line copies the TripRpts folder
DIR=C:\TripRpts, %CSIDL_PERSONAL%
; The next line copies everything in the TrpRpts folder
DIR=C:\TripRpts\*, %CSIDL_PERSONAL%
; The next line copies all files on the user's desktop in a folder called ; TripRpts
; matching the 99*.DOC wildcard search.
%CSIDL_DESKTOP%\TripRpts\99*.DOC, %CSIDL_PERSONAL%
```

In the previous example all of the files are moved to the user's My Documents folder indicated by CSIDL_PERSONAL. The C:\TripRpts folder would be relocated under My Documents. (C:\Documents and Settings\\My Documents\TripRpts).

Wildcards

The Migration Rule INF file supports Microsoft® MS-DOS® style wildcard searches with a few extensions. For example, the character * matches zero or more characters, ? matches a single character. If the pattern string contains back slashes, the fully qualified path name of the target file is matched against the pattern; otherwise, only the file name+extension portion is matched. Multiple wildcard characters are allowed in the pattern string. If the pattern is intended to match a directory only, it should have a **dir=tag** statement at the beginning of the line.

Examples:

- C:\TripRpts\2000\August.* matches c:\TripRpts\2000\august.doc and c:\TripRpts\2000\august.xls
- It does not match c:\TripRpts\august.doc.
- August.* matches c:\TripRpts\2000\august.doc, c:\TripRpts\2000\august.txt, and c:\TripRpts\august.doc.
- *.doc matches c:\TripRpts\2000\august.doc and c:\TripRpts\august.doc.
- c:\2000*.* matches c:\TripRpts\2000\august.doc, c:\TripRpts\2000\august.txt, and c:\ExpenseRpts\2000\morefiles.ext.
- c:\TripRpts matches a file named c:\TripRpts but not a directory named c:\TripRpts.
- dir=c:\TripRpts matches directory TripRpts but not file TripRpts.
- To get all DOC files on the system, use *.doc.
- To get all DOC files on drive c:\, use c:\.doc or c:*.*.doc

```
Settings\%username%\My Documents\august.txt
```

Registry Include Rules

Additional registry entries or even entire registry subtrees can be included in the migration by adding an **AddReg** label to the **[Copy This State]** section of a Migration Rule INF file. The **AddReg** label is followed by a section name. The section contains registry subkeys and entries to be copied to the destination computer. These entries will overwrite any corresponding registry entries on the destination computer.

Example:

```
[Copy This State]
AddReg=SpaceCadet AddReg
[SpaceCadet AddReg]
HKR\Software\Microsoft\Plus!\Pinball\ SpaceCadet\*=
```

Registry Exclude Rules

If some entries should not be migrated, add a **DelReg** label to the **[Copy This State]** section of a Migration Rule INF file. The **DelReg** label is followed by a section name. The section contains registry entries that should not be copied to the destination.

Note Using this rule does not actually delete any entries from the destination computer.

Example:

```
[Copy This State]
DelReg=SpaceCadet DelReg
[SpaceCadet DelReg]
HKR\Software\Microsoft\Plus!\Pinball\ SpaceCadet [Table Exe]=
```

Registry Move and Copy Rules

Often, a registry entry must be moved during the migration. If upgrading from Windows 95 or Windows 98 to Windows 2000 Professional, most registry subkeys and entries must be changed, moved, deleted, or added. This is done by adding the **RenReg** label in the **[Copy This State]** section. The **RenReg** label is followed by a section name. This section contains registry entries to be copied to the destination machine and renamed or moved.

Example:

```
[Copy This State]
RenReg= Outlook98&97 RenReg
[Outlook98&97 RenReg]
HKR\Software\Microsoft\Windows Messaging Subsystem\Profiles\*= \ HKR\Software\Microsoft\Windows
NT\CurrentVersion\Windows Messaging Subsystem\Profiles
```


Note When a back slash (\) is the last character on a line in a Migration Rule INF file, it indicates that the command continues on the next line.

When the value of a registry entry is a pointer to a file that also needs to be migrated, the **RegFile** label in the **[Copy This State]** section can copy the file indicated by the registry value. The **RegFile** label is followed by a section name. This section contains registry entries. Each entry listed must contain a file name. All file names listed are copied to the destination computer. The file path in the entry on the destination computer is set to the actual location of the file (in case the file was moved). An entry can be listed under both the **RenReg** and **RegFile** sections. **RegFile** can also move files.

Example:

```
[Copy This State]
RegFile=Desktop RegFile
[Desktop RegFile]
HKR\Control Panel\Desktop [Wallpaper]=
; This next line would move the wallpaper image to My Documents.
; You would NOT have both of these in the same file.
HKR\Control Panel\Desktop [Wallpaper]=, %CSIDL_PERSONAL%
```

Rule Precedence

When multiple rules can be applied to the same file or folder, an algorithm is used to find the best matching rule. In general, the one that *most closely describes the file or folder in question wins*. If two rules describe the file or folder equally well, the **INCLUDE** rule wins. The full path and file name are considered when trying to find the closest match. Equally matching rules often have unexpected results. Registry rules are applied in the following order:

- **DelReg** can be specified to ScanState.exe or LoadState.exe.
- **AddReg**, **RenReg**, and **RegFile** can be specified to either ScanState.exe or LoadState.exe. However, if they are specified to LoadState.exe and not ScanState.exe, the specified entry might not be copied. In the case of **RegFile**, the file might not be copied.
- Anything listed under **DelReg** is not copied to the destination computer.
- Anything listed under **RenReg** is renamed. If the same key is listed multiple times, only one rename rule applies.
- Anything listed under **RegFile** must have the path in the value fixed. Listing an entry under **RegFile** multiple times is the same as listing it once.
- Anything listed under **AddReg** overwrites the corresponding key on the destination computer. Listing an entry multiple times is the same as listing it once.
- If an entry exists on the destination computer, it is not overwritten (unless the entry is listed in an **AddReg** section).

An entry can have one rule from each of the set of **RenReg**, **RegFile**, and **AddReg** rules. In that case, all the specified rules are applied simultaneously. However, all rules applied simultaneously must be at the same depth in the path. For example, you might apply an **AddReg** and **RenReg** rule to the HKEY_LOCAL_MACHINE \Software \Microsoft \Exchange* subkey. If you applied an **AddReg** rule to the HKEY_LOCAL_MACHINE \Software \Microsoft \Exchange* subkey and a **RenReg** rule to the HKEY_LOCAL_MACHINE \Software \Microsoft \Exchange \Client subkey, the **RenReg** rule takes precedence because it provides the more specific description.

[Send feedback to Microsoft](#)

[© 2004 Microsoft Corporation. All rights reserved.](#)