




























Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>

Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>Introduction to Registry Value Entries</u>
	<u>CurrentControlSet\Select Subkey</u>
	<u>Standard Entries for CurrentControlSet\Services Subkeys</u>
	<u>Other Registry Keys</u>
	<u>System Components with No Additional Entries</u>
	<u>LAN Manager vs. Windows NT Parameters</u>
	<u>Windows 3.1 .INI Settings in Windows NT</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>














Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Overview of CurrentControlSet\Control Subkeys</u>
	<u>Control Value Entries</u>
	<u>BootVerificationProgram</u>
	<u>GroupOrderList</u>
	<u>HiveList</u>
	<u>Keyboard Layout</u>
	<u>NetworkProvider</u>
	<u>NIs</u>
	<u>Print</u>
	<u>ServiceGroupOrder</u>
	<u>Session Manager</u>
	<u>VirtualDeviceDrivers</u>
	<u>Windows</u>
	<u>WOW</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>




































Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Overview of Network Adapter Cards</u>
	<u>NetRules Subkey Entries</u>
	<u>Linkage Subkey Entries for Network Components</u>
	<u>Parameters Subkey Entries for Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>


























Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Overview of Device Drivers</u>
	<u>File System Drivers and Recognizers</u>
	<u>Disk, Serial, and Parallel Port Entries</u>
	<u>Mouse and Keyboard Driver Entries</u>
	<u>SCSI Miniport Driver Entries</u>
	<u>Sound Driver Entries</u>
	<u>Video Device Driver Entries</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>













Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Overview of network services</u>
	<u>Alert service entries</u>
	<u>AppleTalk and MacFile entries for SFM</u>
	<u>BootVerification service entries</u>
	<u>Browser service entries</u>
	<u>DiskPerf service entries</u>
	<u>DLC system driver entries</u>
	<u>EventLog service entries</u>
	<u>NBF (NetBEUI) transport entries</u>
	<u>NetLogon service entries</u>
	<u>NetworkProvider service entries</u>
	<u>NWLink transport entries</u>
	<u>Redirector (Rdr) service entries</u>
	<u>Remote Access Service (RAS) entries</u>
	<u>Replicator service entries</u>
	<u>Schedule service entries</u>
	<u>Server service entries</u>
	<u>TCP/IP transport entries</u>
	<u>UPS service entries</u>
	<u>Workstation service entries</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>

Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Overview of Microsoft Mail</u>
	<u>Address Book Entries for Mail</u>
	<u>Custom Commands Entries for Mail</u>
	<u>Custom Messages Entries for Mail</u>
	<u>Custom Menus Entries for Mail</u>
	<u>Microsoft Mail Entries</u>
	<u>MMF Entries for Mail</u>
	<u>Mac FileTypes Entries for Mail</u>
	<u>MS Proofing Tools Entries for Mail</u>
	<u>Providers Entries for Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>













Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>Overview of Microsoft Schedule+</u>
	<u>Microsoft Schedule+ Entries</u>
	<u>Microsoft Schedule+ Appt Books Entries</u>
	<u>Microsoft Schedule+ Archives Entries</u>
	<u>Microsoft Schedule+ Exporter Entries</u>
	<u>Microsoft Schedule+ Importer Entries</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>

Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Overview of User Preferences</u>
	<u>Hive Information for User Profiles</u>
	<u>Console Entries for Users</u>
	<u>Color, Pattern, and Screen Saver Entries for Users</u>
	<u>Cursors Entry Values for Users</u>
	<u>Desktop Entry Values for Users</u>
	<u>Environment Variable Entries for Users</u>
	<u>Extensions Entries for Users</u>
	<u>File Manager Software Settings</u>
	<u>International Entry Values for Users</u>
	<u>Keyboard and Keyboard Layout Entries for Users</u>
	<u>Mouse Entries for Users</u>
	<u>Multimedia and Sound Entries for Users</u>
	<u>Network Entries for Users</u>
	<u>Program Manager Entries for Users</u>
	<u>Windows Help Entries</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>

Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Overview of Fonts</u>
	<u>Font Drivers Entries</u>
	<u>FontCache Entries</u>
	<u>Fonts Entries</u>
	<u>GRE_Initialize Entries</u>
	<u>FontSubstitutes Entries</u>
	<u>TrueType Entries for Users</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>















Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Overview of Printing</u>
	<u>Printing Entries for Users</u>
	<u>Control\Print Entries for the Computer</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>

Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Overview of Subsystems</u>
	<u>Microsoft OS/2 Version 1.x Software Registration Entries</u>
	<u>Windows Software Registration Entries</u>
	<u>WOW Software Registration Entries</u>
	<u>Windows NT Software Registration Entries</u>
	<u>Winlogon Registry Entries</u>
	<u>Summary of Other Registry Keys</u>

Microsoft Windows NT Registry Entry Guide Contents

	<u>List of Device Driver and Service Entries</u>
	<u>List of Software and User Preference Entries</u>
	<u>List of Key Names</u>
	<u>Overview Topics</u>
	<u>CurrentControlSet\Control Subkeys</u>
	<u>Network Adapter Cards</u>
	<u>Device Drivers</u>
	<u>Network Services</u>
	<u>Microsoft Mail</u>
	<u>Microsoft Schedule+</u>
	<u>User Preferences</u>
	<u>Fonts</u>
	<u>Printing</u>
	<u>Subsystems</u>
	<u>Summary of Other Registry Keys</u>
	<u>Overview of Other Registry Keys</u>
	<u>HKEY_LOCAL_MACHINE\HARDWARE Subtree</u>
	<u>HKEY_LOCAL_MACHINE\SAM Subtree</u>
	<u>HKEY_LOCAL_MACHINE\SECURITY Subtree</u>
	<u>HKEY_LOCAL_MACHINE\SOFTWARE Subtree</u>
	<u>HKEY_CLASSES_ROOT</u>

Introduction to Registry Value Entries

Wherever possible, you should use the Control Panel and the applications in the Administrative Tools program group to make changes to the Microsoft Windows NT system configuration. When you need to use REGEDT32.EXE to view, add, or change a setting, use the information in this online Guide.

Caution: Using Registry Editor incorrectly can cause serious problems, including corruptions that may make it necessary to reinstall Windows NT.

Not all entries that appear in this Help file may be found in the Registry for a particular computer. For many entries, the system uses the default value unless you add the entry to the Registry and specify another value.

Note: In general, if you change values for any entries in the CurrentControlSet, you must restart the computer for the changes to take effect. If you change values for entries under HKEY_CURRENT_USER using Registry Editor, you may have to log off and log back on for the changes to take effect.

The Registry is structured as a set of four subtrees of keys that contain per-computer and per-user databases. The per-computer information includes information about hardware software installed on the specific computer. The per-user information includes the information in user profiles, such as desktop settings, individual preferences for certain software, and personal printer and network settings. In versions of Windows for MS-DOS, per-computer information was saved in WIN.INI and SYSTEM.INI files, but it was not possible to save separate information for individual users.

In the Windows NT Registry, each individual key can contain data items called *value entries* and can also contain additional *subkeys*. In the Registry structure, keys are analogous to directories, and the value entries are analogous to files.

HKEY_LOCAL_MACHINE

Contains information about the local computer system, including hardware and operating system data such as bus type, system memory, device drivers, and startup control data.

HKEY_CLASSES_ROOT

Contains object linking and embedding (OLE) and file-class association data (equivalent to the Registry in Windows for MSDOS).

HKEY_CURRENT_USER

Contains the user profile for the user who is currently logged on, including environment variables, personal program groups, desktop settings, network connections, printers, and application preferences.

HKEY_USERS

Contains all actively loaded user profiles, including HKEY_CURRENT_USER, which always refers to a child of HKEY_USERS, and the default profile. Users who are accessing a server remotely do not have profiles under this key on the server; their profiles are loaded into the Registry on their own computers.

Each of the root key names begins with HKEY_ to indicate to software developers that this is a *handle* that can be used by a program. A handle is a value used to uniquely identify a resource so that a program can access it.

A value entry has three parts: the name of the value, the data type of the value, and the value itself, which can be data of any length. The three parts of value entries always appear in the following order:

<name> : <data type> : <value>

A value entry cannot be larger than about 1 MB. The data type of a value entry is an integer, the meaning of which is agreed to by convention. Values from 0 to 0x7fffffff are reserved for definition by the system. Values from 0x80000000 to 0xffffffff are reserved for use by applications.

The predefined data types for Registry entries are the following:

REG_BINARY

Raw binary data. Most hardware component information is stored as binary data and can be displayed in WinMSD in an easy-to-read format. In Registry Editor, the data is displayed in hexadecimal format. For example:

Component Information : REG_BINARY : 00 00 00...

REG_DWORD

Data represented by a number that is 4 bytes long. Many parameters for device driver and services are this type and can be displayed in Registry Editor in binary, hex, or decimal format. For example, entries for service error controls are this type:

ErrorControl : REG_DWORD : 0x1

REG_EXPAND_SZ

An expandable data string, which is text that contains a variable to be replaced when called by an application. For example, for the following value, the string %SystemRoot% will be replaced by the actual location of the directory containing the Windows NT system files:

File : REG_EXPAND_SZ : %SystemRoot%\file.exe

REG_MULTI_SZ

A multiple string. Values that contain lists or multiple values in human-readable text are usually this type. Entries are separated by NULL characters. For example, the following value entry specifies the binding rules for a network transport:

bindable : REG_MULTI_SZ : dlcDriver dlcDriver non non 50

REG_SZ

A sequence of characters representing human-readable text. For example, a components description is usually this type:

DisplayName : REG_SZ : Messenger

CurrentControlSet\Select Subkey

The Select subkey maintains information about the control sets for the currently selected computer.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \Select

The Select subkey contains the following named values:

Current **REG_DWORD**

Range: $0xN$, where N identifies a control set

Identifies the control set from which the CurrentControlSet subkey is derived. If this value is 0x1, for example, the subkey producing the CurrentControlSet is ControlSet001.

Default **REG_DWORD**

Range: $0xN$, where N identifies a control set

Identifies the default control set. If this value is 0x1, for example, the default control set is ControlSet001.

Failed **REG_DWORD**

Range: $0xN$, where N identifies a control set

Identifies the control set number of the control set that was last rejected and replaced with a LastKnownGood control set.

LastKnownGood **REG_DWORD**

Range: $0xN$, where N identifies a control set

Identifies the last control set that successfully started the system. If this value is 0x1, for example, the last control set known to be good is ControlSet001.














Overview of CurrentControlSet\Control Subkeys

This key contains parameters that control system startup, such as subsystems to load, the size and location of paging files, and so on.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control

Standard Control subkeys:

	<u>Control Value Entries</u>
	<u>BootVerificationProgram</u>
	<u>GroupOrderList</u>
	<u>HiveList</u>
	<u>Keyboard Layout</u>
	<u>NetworkProvider</u>
	<u>Nls</u>
	<u>Print</u>
	<u>ServiceGroupOrder</u>
	<u>Session Manager</u>
	<u>VirtualDeviceDrivers</u>
	<u>Windows</u>
	<u>WOW</u>

Note: The system must be restarted for any changes in the Control key to take effect.

Control Value Entries

The Control subkey itself can contain the following value entries.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control

Current User **REG_SZ**

Range: *Username*

Default: *Username*

Specifies the username for the currently logged-on user.

RegistrySizeLimit **REG_DWORD**

Range: *Size in bytes*

Default: 8 MB (That is, 25 percent of the default **PagedPoolSize**.)

Specifies the total amount of space that can be consumed by Registry data.

The system ensures that the value for **RegistrySizeLimit** is at least 4 MB and no greater than about 80 percent of the size of **PagedPoolSize**. Setting **RegistrySizeLimit** to 0xffffffff sets the value to be as large as 80 percent of **PagedPoolSize** (under the Control\Session Manager\MemoryManagement key). You can allow for a bigger Registry by setting the initial **PagedPoolSize**, or you can set the value of **RegistrySizeLimit**. If you want a very large Registry, you want to set both. However, for all but a few domain controllers, **RegistrySizeLimit** never needs to be changed.

RegistrySizeLimit must have a type of REG_DWORD and a data length of 4 bytes, or it will be ignored.

SystemStartOptions **REG_SZ**

Range: *String*

Contains the text of system arguments passed to the system by the firmware. These values can be used to determine whether the debugger is enabled, the options set for ports and speed, and so on. For example, the following value could be defined for **SystemStartOptions**:

c:\winnt="Windows NT" /DebugPort=com1 /DebugBaudRate=56000

In this example, "Windows NT" in the first part of the string indicates the *SystemRoot* specified under the Windows NT\CurrentVersion subkey in the Software area of the Registry. The remaining portion of this string is interpreted by the system to define the COM port and baud rate for debugging.

BootVerificationProgram Control Entries

The BootVerificationProgram key is used to update the last known good configuration during system startup. This entry describes a program that will be called by the service controller to establish the last known good configuration. Such a program can be written by the system administrator.

This program can, for example, query another server. If the response is not as expected, the program can call the NotifyBootConfigStatus() function with FALSE, causing the system to restart using the LastKnownGood control set. Or the program might cause the system to run without saving the current configuration as the LastKnownGood control. Conversely, if the SQL server responds as expected, the program can call the NotifyBootConfigStatus() function with TRUE, causing the current configuration to be saved as the LastKnownGood control.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Control
            \BootVerificationProgram
```

Any entry added under this subkey must have the following format:

ImagePath **Reg_SZ** or **REG_EXPAND_SZ**

Range: *Filename*

Default: (no entry)

Specifies the filename for a startup verification program.

You must also specify a value of 1 for the **ReportBootOK** entry under the following Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
    \Windows NT
        \CurrentVersion
            \WinLogon
```

When the value of **ReportBootOK** is set to 0, it disables the automatic (default) startup acceptance, which happens after the first successful logon.

As an alternative, if you want to verify system startup from a remote location, the BootVerification service supplied with Windows NT can be used instead of the BootVerificationProgram. The BootVerification service cannot be used in conjunction with entries in the BootVerificationProgram subkey.

Note: Do not change values in the BootVerificationProgram key unless you need a custom verification program to satisfy specific startup criteria at your site. Otherwise, accept the default procedures for verifying system startup.

GroupOrderList Control Entries

The entries in the GroupOrderList key specify the ordering of services within groups.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \GroupOrderList
```

For a service listed under CurrentControlSet\Services, the value of the **Group** entry plus any **Tag** entry determines the order in which the service is loaded. But not all services have a **Tag** entry, and not all groups have an entry in the GroupOrderList subkey. The ServiceGroupOrder subkey specifies the order for loading groups.

The entries in the key are all of type REG_BINARY. These are the default entries that define the order within groups:

- Base
- Filter
- Keyboard Class
- Keyboard Port
- Ndis
- Pointer Class
- Pointer Port
- Primary Disk
- SCSI Miniport
- Video

HiveList Control Entries

The location of the files that contain Registry information is reported in the HiveList key.

Registry path:

```
HKEY_CURRENT_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \hivelist
```

All data types are REG_SZ. The following are the default entries:

\REGISTRY\MACHINE\HARDWARE

\REGISTRY\MACHINE\SAM=

\Device\Harddisk0\Partition1\SystemRoot\SYSTEM32\CONFIG\SAM

\REGISTRY\MACHINE\SECURITY=

\Device\Harddisk0\Partition1\SystemRoot\SYSTEM32\CONFIG\SECURITY

\REGISTRY\MACHINE\SOFTWARE=

\Device\Harddisk0\Partition1\SystemRoot\SYSTEM32\CONFIG\SOFTWARE

\REGISTRY\MACHINE\SYSTEM=

\Device\Harddisk0\Partition1\SystemRoot\SYSTEM32\CONFIG\SYSTEM

\REGISTRY\USER\DEFAULT=

\Device\Harddisk0\Partition1\SystemRoot\SYSTEM32\CONFIG\DEFAULT

\REGISTRY\USER\SID_#=

\Device\Harddisk0\Partition1\SystemRoot\SYSTEM32\CONFIG\ADMIN000

Keyboard Layout Control Entries

The Keyboard Layout key maintains the mapping of keyboard layout names to keyboard layout DLL names.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Keyboard Layout

The Keyboard Layout key is used by the system to determine which DLL to load. The mapping may contain duplicated keyboard layout DLL names if that DLL implements the layout for more than one language (in other words, a keyboard layout may have more than one name). Individual preferences are stored under HKEY_CURRENT_USER\Keyboard Layout.

The Keyboard Layout key contains an entry in this format:

KeyboardLayout **REG_SZ**

Range: *DLL filename*

By convention, a keyboard layout name is a null-terminated string of 8 hexadecimal digits representing a language ID (in the last four digits) and a variation number (in the first four digits).

For example, the language ID of Swiss German is 0x807, so, by convention, keyboard layout names could be 00000807, 00010807, and so on.

The DosKeybCodes subkey is used by the system to convert Windows NT keyboard layout names into MS-DOS-style two-character keyboard layout names as used by the **kb16** command (equivalent to **keyb** in MS-DOS). The system must automatically switch the VDM keyboard layout whenever the Windows NT keyboard layout is changed.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Keyboard Layout
 \DosKeybCodes

KeyboardLayout **REG_SZ**

Range: *Two-character code*

These entries define the two-character equivalent for each keyboard layout.

DosKeybCodes Entries

DosKeybCodes Entries

00000405=cz	0000041B=sl
00000406=dk	0000041D=sv
00000407=gr	00000807=sg
00000409=us	00000809=uk
0000040A=sp	0000080A=la
0000040B=su	0000080C=be
0000040C=fr	00000813=be
0000040E=hu	00000816=po
0000040F=us	00000C0C=cf
00000410=it	00001009=us
00000413=nl	0000100C=sf
00000414=no	00010409=dv
00000415=pl	0001040A=sp
00000416=br	00010C0C=cf
00000419=ru	00020409=us
0000041A=yu	

NetworkProvider Control Entries

Windows NT supports a common interface that allows networks from several different vendors (that is, network providers) to operate on a single computer at the same time. The NetworkProvider subkey provides a list of the available network providers that use the Microsoft network-independent APIs.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \NetworkProvider
        \Order
```

ProviderOrder REG_SZ

Range: *Comma-separated list of key names*

Default: LanmanWorkstation (when a single network is installed)

Specifies the order for accessing available network providers, as defined by choosing the Networks button in the Network dialog box in Control Panel. Each entry in this list is in the form of a Registry Services subkey name for a service or driver that is associated with a particular network provider. The Services subkey associated with a network provider should contain a **Group** entry with the value of NetworkProvider, and must also contain a NetworkProvider subkey with information for the Multiple Provider Router.

See also:

[NetworkProvider Service Entries](#)

Nls Control Entries

This key contains subkeys that define information for languages and code pages.

In the Code Page and Language subkeys, all code pages and languages supported under Windows NT are listed, so applications can check these keys to find all "supported" languages. However, only the entries with filenames in the data fields are actually installed in the system. So applications must check the data fields to find out if a specific code page is actually installed in the system.

CodePage Entries

The entries identify the files for available code pages. If there is no value following the entry name, that code page is not installed in the system.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Nls
        \CodePage
```

Each entry has the following format:

CodePageID **REG_SZ**

Range: *Filename*

As shown in the following list, **437**=C_437.NLS indicates the file for code page 437, and **1252**=C_1252.NLS indicates the file for code page 1252, and so on. **ACP** is the default ANSI code page; **OEMCP** is the default OEM code page; and **MACCP** is the default Macintosh code page.

10000=c_10000.nls

10006=

10007=

10029=

1026=

1250=

1251=

1252=c_1252.nls

1253=

1254=

437=c_437.nls

500=

850=c_850.nls

852=

855=

857=

860=

861=

863=

865=

866=

869=

875=

ACP=1252

MACCP=10000

OEMCP=437

Language Entries

The entries identify the files for available languages. If there is no value following the entry name, that language is not installed in the system.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Nls
        \Language
```

Each entry has the following format:

LanguageID **REG_SZ**

Range: *Filename*

Default: 0409 (the default language ID)

As shown in the following table, I_INTL.NLS is the file for language 0409, I_INTL.NLS is the file for language 0809, and so on.

0405=I_intl.nls

0406= I_intl.nls

0407=I_intl.nls

0408=

0409=I_intl.nls

040a=I_intl.nls

040b= I_intl.nls

040c=I_intl.nls

040e=I_intl.nls

040f=I_intl.nls

0410=I_intl.nls

0413=I_intl.nls

0414= I_intl.nls

0415=I_intl.nls

0416=I_intl.nls

0419=I_intl.nls

041b=I_intl.nls

041d=I_intl.nls

041f=

0807=I_intl.nls

0809=I_intl.nls

080a=I_intl.nls

080c=I_intl.nls

0810=I_intl.nls

0813=I_intl.nls

0814=I_intl.nls

0816=I_intl.nls

0c07= I_intl.nls

0c0a=I_intl.nls

0c0c=l_intl.nls

1009=l_intl.nls

100c=l_intl.nls

1409=l_intl.nls

1809=l_intl.nls

28=l_intl.nls

OEMLocale Entries

The following key, if present, can contain entries that an OEM adds to customize its locale:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Nls
        \OEMLocale
```

This section of the Registry is only checked if a specific locale ID is not found in the default locale file (LOCALE.NLS). If this key is present, each entry has the following format:

OEMlocale **REG_SZ**

Range: *filename*

Default: This subkey is not present, so there is no default.

Specifies the locale ID and a filename for the OEM locale.

ServiceGroupOrder Control Entries

The ServiceGroupOrder key specifies the order to load various groups of services.

Order within groups is specified using the value of **Tag** under the specific Services subkeys and the values in the GroupOrderList subkey. For example, when you start Windows NT, the Boot Loader scans the Registry for drivers with a **Start** value of 0 (which indicates that these drivers should be loaded but not initialized before the Kernel) and a **Type** value of 0x1 (which indicates a Kernel device driver such as a hard disk or other low-level hardware device driver). The drivers are then loaded into memory in the order specified as the **List** value in the ServiceGroupOrder subkey.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Control
            \ServiceGroupOrder
```

List **REG_MULTI_SZ**

Range: *Group names* (specifies the order for loading drivers into memory)

Default:

SCSI miniport
port
Primary disk
SCSI class
SCSI CDROM
class filter
boot file system
Base
Keyboard Port
Pointer Port
Keyboard Class
Pointer Class
Video
file system
Event log
Streams Drivers
NDIS
TDI
NetBIOSGroup
NetDDEGroup
extended base
network

Session Manager Control Key

The Session Manager subkey contains the global variables used by the Session Manager.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Control

\Session Manager

Several subkeys appear under the Session Manager key:



Session Manager Control Entries



DOS Devices Control Entries



Environment Control Entries



KnownDLLs Control Entries



Memory Management Control Entries



Subsystem Startup Control Entries



Windows Startup Control Entries



WOW Startup Control Entries

Session Manager Control Entries

The Session Manager key itself contains several entries.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Session Manager

BootExecute **REG_MULTI_SZ**

Default: autocheck autochk *

Specifies programs to run during startup. For example, if CONVERT.EXE has been used to convert the file system on a hard disk drive, this value is added to **BootExecute** so that conversion occurs when the system is restarted:

BootExecute = autocheck autoconv \Dos\Devices\X: /FS:NTFS

CriticalSectionTimeout **REG_DWORD**

Default: 0x278d00

Specifies the deadlock time-out for critical sections. Usually, retail installations of Windows NT will not time-out and detect deadlocks.

GlobalFlag **REG_DWORD**

Default: 0x21100000

Controls various Windows NT internal operations. You can change this value to disable the OS/2 subsystem if you want to run bound applications in a VDM, rather than under the OS/2 subsystem. Set this value to 20100000 to disable the OS/2 subsystem.

See NTEXAPI.H FLG_ definitions in the Windows NT SDK.

ObjectDirectories **REG_MULTI_SZ**

Default:

\DosDevices

\Windows

\RPC Control

Lists the object directories to create during startup. Do not edit these entries.

DOS Devices Control Entries

The DosDevices subkey lists the built-in symbolic links to create at startup.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Control

 \Session Manager

 \DOS Devices

Entries in this key have the data type of REG_SZ. The following list shows the default entries under this key.

AUX=\DosDevices\COM1

MAILSLOT=\Device\MailSlot

NUL=\Device\Null

PIPE=\Device\NamedPipe

PRN=\DosDevices\LPT1

TELNET=\Device\Telnet

UNC=\Device\Mup

Environment Control Entries

The Environment subkey defines environment variables that the system creates and that are used by Windows NT Logon and Program Manager.

Caution: Use extreme care in changing these entry values. If the operating system cannot find the files specified for a subsystem, you will not be able to run non-Windows NT applications.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Session Manager
 \Environment

ComSpec **REG_EXPAND_SZ**

Range: *Filename*

Default: %systemRoot%\SYSTEM32\CMD.EXE

Defines the path and filename for the Windows NT command interpreter (the equivalent of MS-DOS COMMAND.COM).

Os2LibPath **REG_EXPAND_SZ**

Range: *Filename*

Default: %systemRoot%\SYSTEM32\os2\dll

Defines the path for the Microsoft OS/2 version 1.x library.

Path **REG_EXPAND_SZ**

Range: *Filename*

Default: %systemRoot%\SYSTEM32; %SystemRoot%

Defines the path variable for Windows NT logon and Program Manager.

Windir **REG_EXPAND_SZ**

Range: *Filename*

Default: %systemRoot%\SYSTEM32\CMD.EXE

Defines the path for the executable for WOW, as used by Windows NT logon and Program Manager.

KnownDLLs Control Entries

The KnownDLLs key defines the set of DLLs that are first searched during system startup. In general these are system DLLs that are loaded from disk into a section of memory and are checked for integrity. These DLLs consume some resources, even if no application loads them.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Session Manager
        \KnownDLLs
```

driverName **REG_SZ**

Range: *DLL Filename*

This series of entries defines a driver name and the corresponding DLL filename. The following shows the default entries:

```
advapi32=advapi32.dll
comdlg32=comdlg32.dll
crtdll=crt.dll
DllDirectory=%SystemRoot%\system32
gdi32=gdi32.dll
kernel32=kernel32.dll
lz32=lz32.dll
olecli32=olecli32.dll
olesvr32=olesvr32.dll
rpcrt4=rpcrt4.dll
shell32=shell32.dll
user32=user32.dll
version=version.dll
```

Memory Management Control Entries

The Memory Management subkey defines paging options.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Session Manager
 \Memory Management

The paging file parameters should be defined by using the System icon in Control Panel and choosing the Virtual Memory button.

IoPageLockLimit **REG_DWORD**

Range: *Number of bytes*

Default: 512K

Specifies the limit of the number of bytes that can be locked for I/O operations. When this value is 0, the system uses the default (512K). The maximum value is about the equivalent of physical memory minus pad, which is 7 MB for a small system and grows as the amount of memory grows. For a 64 MB system, pad is about 16 MB; for a 512 MB system, pad is about 64 MB.

LargeSystemCache **REG_DWORD**

Range: *Number*

Default: 0

Specifies, for a nonzero value, that the system favor the system-cache working set rather than the processes working set. Set this value by choosing the Windows NT Advanced Server installation base.

NonPagedPoolSize **REG_DWORD**

Range: *Number of bytes*

Default: 0

Specifies the size of nonpaged pool in bytes. When this value is 0, the system uses the default size (based on physical memory). The maximum amount is about 80 percent of physical memory.

PagedPoolSize **REG_DWORD**

Range: *0 to 128 MB*

Default: 0x3000000 (32 MB)

Specifies the size of paged pool in bytes. When this value is 0, the system uses the default size (32 MB). See also **RegistrySizeLimit** in Control Value Entries.

PagingFiles **REG_MULTI_SZ**

Range: *System_Paging_Files*

Default: C:\pagefile.sys 27

Specifies page file information set by choosing the System icon in Control Panel.

Subsystem Startup Control Entries

Defines the subsystem settings established at startup.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Session Manager
 \SubSystems

These values should only be maintained by the system. You should not need to manually define these settings.

Debug **REG_EXPAND_SZ**

Range: *Names*

Default: (no value)

Optional **REG_MULTI_SZ**

Range: *Subsystem names*

Default: Os2 Posix

Defines subsystems that are only loaded when the user starts an application that requires this subsystem.

Os2 **REG_EXPAND_SZ**

Range: *Path and filename*

Default: %SystemRoot%\system32\os2ss.exe

Defines the path to the executable file used to start the Microsoft OS/2 version 1.x subsystem.

Posix **REG_EXPAND_SZ**

Range: *Path and filename*

Default: %SystemRoot%\system32\psxss.exe

Defines the path to the executable file used to start the POSIX subsystem. (There are no additional POSIX entries in the Registry.)

Required **REG_MULTI_SZ**

Range: *Name*

Default: Debug Windows

Windows **REG_EXPAND_SZ**

Range: *Path and filename for the executable file used to start the Win32 subsystem*

Default:

%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,3072
Windows=On SubSystemType=Windows ServerDll=basesrv,1
ServerDll=winsrv:GdiServerDllInitialization,4 ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ServerDll=mmsndsrv,5 ProfileControl=Off
MaxRequestThreads=16

VirtualDeviceDrivers Control Entries

The VirtualDeviceDriver key contains a list of Win32 DLLs that serve as virtual device drivers (VDD). Each **VDD** entry results in loading that VDD when a virtual MS-DOS machine (VDM) is being created for running an application created for MS-DOS or 16-bit versions of Microsoft Windows.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Control
            \VirtualDeviceDrivers
```

VDD REG_MULTI_SZ

Range: *Filenames*

Default: None--the Windows NT VDDs are built into the system.

Specifies valid Win32 DLLs that are virtual device drivers.

Windows Startup Control Entries

This key contains entries that define the system directories for the Win32 subsystem (32-bit Windows).

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \Windows

Directory **REG_EXPAND_SZ**

Range: *Directory name*

Default: %SystemRoot%

Defines the directory for Windows NT.

SystemDirectory **REG_EXPAND_SZ**

Range: *Directory name*

Default: %SystemRoot%\system32

Defines the directory for the Windows NT system files.

The Registry entries for starting the Win32 subsystem are defined under the **Required** and **Windows** value entries in the Session Manager\Subsystem key.

WOW Startup Control Entries

The following values control startup parameters that affect MSDOS-based applications and applications created for 16-bit Windows 3.1.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Control
 \WOW

Cmdline **REG_EXPAND_SZ**

Range: *Path and switches*

Default: %SystemRoot%\system32\ntvdm -f%SystemRoot%\system32 -a

Defines the command line that runs when an MS-DOS-based application runs under Windows NT. This command line continues to run until the related application is closed. The following switches can be included:

- a = Specifies a command to pass to the VDM
- f = Specifies the directory to find NTVDM.EXE
- m = Hides the VDM console window
- w = Specifies the WOW VDN

KnownDLLs **REG_SZ**

Range: *DLL filenames*

Default: shell.dll commdlg.dll mmsystem.dll olecli.dll olesvr.dll ddeml.dll win87em.dll toolhelp.dll lanman.driv netapi.dll pmspl.dll wowdeb.exe

Defines a list of known DLLs for use by the WOW VDM that provide compatibility for non-Win32 applications. When the system searches for DLLs to load, it compares the requested DLL with those in the **KnownDLLs** list and then loads the matching DLL from the *SystemRoot\SYSTEM32* directory.

If you want to replace a DLL, you must delete the name from this list, so that the system will search elsewhere for the DLL. The files USER.DLL, GDI.DLL, and SYSTEM.DRV are not included in this list, because these are required Windows NT system files and their location cannot be changed.

LPT_timeout **REG_SZ**

Range: *Number of seconds*

Default: 15

Defines how many seconds after the LPT port has been used that Windows NT waits before grabbing the port, closing it, and flushing the output. This value should only be needed for applications that use BIOS and do not close the port. It is used only by MS-DOS-based applications.

Size **REG_SZ**

Range: *Number in megabytes*

Default: 0

Defines the amount of memory to be given to each individual MS-DOS VDM. The default of 0 gives the VDM as much memory as Windows NT determines is necessary, depending upon the memory configuration.

Wowcmdline **REG_EXPAND_SZ**

Range: *Path and switches*

Default: %SystemRoot%\system32\ntvdm -m -w -f%SystemRoot%\system32 -a %SystemRoot%\system32\krnl386

Defines the command line that runs when a 16-bit Windows-based application is started. The switches instruct Windows NT to start either an MS-DOS VDM or a WOW VDM, as follows:

- a = Specifies a command to pass to the VDM
- f = Specifies the directory to find NTVDM.EXE
- m = Hides the VDM console window
- w = Specifies the WOW VDM

Wowsize REG_SZ

Range: *Number in megabytes*

Default: Depends on RISC-based computer's system memory

For RISC-based computers, defines the amount of memory provided in a VDM when a WOW session is started. This value is not used on x86-based computers, where Windows NT allocates the memory needed when it is asked for.

The default size chosen for a RISC-based computer depends on the amount of system memory on the computer. For each MB specified, the system uses 1.25 MB, so setting **Wowsize** to 4 MB causes the VDM to allocate 5 MB, although applications can only use 4 MB. You can override the following defaults:

For less than 12 MB system memory (small), **Wowsize** = 3 MB

For 12 - 16 MB system memory (medium), **Wowsize** = 6 MB

For more than 16 MB system memory (large), **Wowsize** = 8 MB

Caution: Setting **Wowsize** to a value lower than 3 MB will cause most applications to fail.

Standard Entries for CurrentControlSet\Services Subkeys

The Services subkeys contain parameters for the device drivers, file system drivers, and Win32 service drivers.

Registry path:






HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services

The name of each Services subkey is the name of the service, which is also the root of the name of the file from which the service is loaded. For example, for the serial mouse, the service name and Services subkey name is Sermouse. The file from which this is loaded is *SystemRoot\SYSTEM32\DRIVERS\SERMOUSE.SYS*.

All service names are defined under HKEY_LOCAL_MACHINE\SOFTWARE. The names of the Windows NT built-in network services such as the Alerter and Browser services are defined under the Microsoft\Windows NT\CurrentVersion subkey in the Software area of the Registry.

Each Services key can have additional subkeys. Many services have a Linkage subkey, which provides data for binding network components, as described in [Linkage Subkey Entries for Network Components](#). Many services also have a Parameters subkey that contains entries defined by the service with values for configuring the specific service.

The following standard value entries appear for each Services subkey:

	<u>ErrorControl</u>
	<u>Group, DependOnGroup, DependOnService, and Tag</u>
	<u>ImagePath and ObjectName</u>
	<u>Start</u>
	<u>Type</u>

Values for Parameters subkeys and other service-specific entries are described in these topics:

[Registry Entries for Network Adapter Cards](#)

[Registry Entries for Device Drivers](#)

[Registry Entries for Services](#)

ErrorControl

This basic value entry appears for each Services subkey in the CurrentControlSet.

ErrorControl **REG_DWORD**

Range: *Error constant*

Specifies the level of error control for the service as follows:

0x3 (Critical) -- Fail the attempted system startup. If the startup is not using the LastKnownGood control set, switch to LastKnownGood. If the startup attempt is using LastKnownGood, run a bug-check routine.

0x2 (Severe) -- If the startup is not using the LastKnownGood control set, switch to LastKnownGood. If the startup attempt is using LastKnownGood, continue on in case of error.

0x1 (Normal) -- If the driver fails to load or initialize, startup should proceed, but display a warning.

0x0 (Ignore) -- If the driver fails to load or initialize, start up proceeds. No warning is displayed.

Group, DependOnGroup, DependOnService, and Tag

These basic value entries appear for each Services subkey in the CurrentControlSet.

Group **REG_DWORD**

Range: *Group name*

Default: (null)

Specifies the name of the group of which the particular service is a member.

DependOnGroup **REG_DWORD**

Range: *Group name*

Default: (empty)

Specifies zero or more group names. If one or more groups is listed, at least one service from the named group must be loaded before this service is loaded.

DependOnService **REG_DWORD**

Range: *Service name*

Default: (empty)

Specifies zero or more Services subkey names. If a service is listed here, that named service must be loaded before this service is loaded.

Tag **REG_DWORD**

Specifies a load order within a given group. The value of **Tag** specifies a number that is unique within the group of which the service is a member. The related *GroupName* entry under the Control\GroupOrderList subkey specifies a list of tags, in load order.

For example, the following services that are members of the Primary Disk group could have these values: **Tag**=4 for the Abiosdsk subkey, **Tag**=2 for Atdisk, **Tag**=1 for Cpqarray, and **Tag**=3 for Floppy. The value for **Primary Disk** under the GroupOrderList subkey will use these **Tag** values to specify the defined order for loading these services. As another example, each SCSI miniport service has a unique **Tag** value that is used as an identifier in the **SCSI miniport** value under the GroupOrderList subkey to define which SCSI adapter to load first.

ImagePath and ObjectName

These basic value entries appear for each Services subkey in the CurrentControlSet.

ImagePath REG_DWORD

Range: *Path and filename* (For adapters, this value is ignored.)

Default: For a driver, *systemroot\SYSTEM32\DRIVERS\driverName.SYS*

For a service, *systemroot\SYSTEM32\serviceName.EXE* (where *driverName* or *serviceName* is the same as the related Services subkey name)

ObjectName REG_DWORD

Range: *Object name*

Default: *subkeyName*

Specifies an object name. If **Type** specifies a WIN32 Service, this name is the account name that the service will use to log on when the service runs. If **Type** specifies a Kernel driver or file system driver, this name is the Windows NT driver object name that the I/O Manager uses to load the device driver.

Start

This basic value entry appears for each Services subkey in the CurrentControlSet.

Start REG_DWORD

Range: *Start constant*

Specifies the starting values for the service as follows:

0x0 (Boot) = Loaded by Kernel loader. Represents a part of the driver stack for the boot (startup) volume and must therefore be loaded by the Boot Loader.

0x1 (System) = Loaded by I/O subsystem. Represents a driver to be loaded at Kernel initialization.

0x2 (Auto load) = Loaded by Service Control Manager. To be loaded or started automatically for all startups, regardless of service type.

0x3 (Load on demand) = Loaded by Service Control Manager. Available, regardless of type, but will not be started until the user starts it (for example, by using the Devices icon in Control Panel).

0x4 (Disabled) = Loaded by Service Control Manager, but not to be started under any conditions.

The **Start** value is ignored for adapters. If **Type** is a Win32 Service value, the **Start** value must specify an Auto, Demand, or Disabled value.

Type (Services)

This basic value entry appears for each Services subkey in the CurrentControlSet.

Type **REG_DWORD**

Range: *Type constant*

Specifies the type of service as follows:

0x1 -- a Kernel device driver.

0x2 -- a file system driver, which is also a Kernel device driver.

0x4 -- a set of arguments for an adapter.

0x10 -- a Win32 program that can be started by the Service Controller and that obeys the service control protocol. This type of Win32 service runs in a process by itself.


0x20 -- a Win32 service that can share a process with other Win32 services.


For example, when you start Windows NT, the Boot Loader scans the Registry for drivers with a **Start** value of 0 (which indicates that these drivers should be loaded but not initialized before the Kernel) and a **Type** value of 0x1 (which indicates a Kernel device driver such as a hard disk or other low-level hardware device driver). The drivers are then loaded into memory in the order specified as the **List** value in CurrentControlSet\Control\ServiceGroupOrder.


Registry Entries for Network Adapter Cards

Windows NT supports network adapter drivers under the NDIS 3.0 specification (Network Device Interface Specification).

The following sections describe entries in the other areas of the Registry that contain configuration information for network adapter cards and their drivers, including:

 NetRules subkeys under HKEY_LOCAL_MACHINE\SOFTWARE subkeys for drivers and adapters.

 Linkage subkey entries under HKEY_LOCAL_MACHINE\SYSTEM subkeys for drivers and adapters, defining information about bindings for the component.

 Parameter subkey entries under HKEY_LOCAL_MACHINE\SYSTEM subkeys for network card adapters, defining specific information such as the IRQ number, I/O base address, and other details.

The information provided here is chiefly to be used for informational and troubleshooting purposes. The settings for either the NDIS driver or the network adapter card should be changed using the Network icon in Control Panel.

The CurrentControlSet\Services subkey for a network driver or adapter card includes the standard entries found in the Services subkeys, with the following default values:

ErrorControl = 0x1 (normal)

Start = 0x3 (load on demand)

Type = 0x1 or 0x4 (driver or adapter)

See Also:

[Registry Entries for Services](#)









[TCP/IP Transport Entries](#)

NetRules Subkey Entries

During network reconfiguration (when you choose the Network icon in Control Panel), the system reads the values stored in the NetRules subkeys for information used to bind the network components. The Registry path for these value entries is the following:

- For adapter card drivers and network services, where the *driverName* subkey is the name of the network card's driver, as defined by the system:
HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules
- For network adapter cards, where the *netcard#* subkey is a number, beginning with 01 for the first network adapter:
HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard#*\NetRules

These are the entry values that can appear:

	<u>Bindable</u>
	<u>Bindform</u>
	<u>Class</u>
	<u>Hidden</u>
	<u>Interface</u>
	<u>Review</u>
	<u>Type</u>
	<u>Use</u>

Bindable

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard#*\NetRules

This value entry can appear under the NetRules subkey for network adapter cards.

Bindable REG_MULTI_SZ

Range: *fromClass toClass* Yes|No Yes|No *value*

Defines a possible binding and its constraints. For example:

 bindable : REG_MULTI_SZ : ndisDriver ndisAdapter non exclusive 100

This example specifies that components of class "ndisDriver can be bound to those of class "ndisAdapter. For the other fields:

non = The component of class ndisDriver can accept other bindings

exclusive = The component of class ndisAdapter cannot accept other bindings.

100 = The relative importance (weight) of this binding; that is, in cases of competition, it will be discarded in favor of other bindings whose weight is greater.

Because this value entry is a REG_MULTI_SZ, as many criteria for binding as necessary can be defined by a single component.

This value entry is optional, because there are a few predefined binding rules, and binding rules defined anywhere in the system apply to all network component classes.

Bindform

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard#*\NetRules

This value entry can appear under the NetRules subkey for network adapter cards.

Bindform REG_SZ

Range: *ObjectName* Yes|No Yes|No [*container|simple|streams*]

The *ObjectName* field contains the name (or name prefix) by which the component is identified by the system. This value must be the same as the name in the related CurrentControlSet\Services subkey. Names for adapters are created by the system and override the **Bindform** setting.

The first Yes|No pair indicates whether the component is to receive binding information directly in its Linkage subkey. The second Yes|No pair indicates whether the device name is supposed to appear in generated binding strings.

The final optional value in this entry indicates how binding device names are constructed. This value is required for software components.

Class

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\driverName\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards\netcard#\NetRules

This value entry can appear under the NetRules subkey for network adapter cards.

Class **REG_MULTI_SZ**

Range: *NewClassName OldClassName|basic [Yes|No]*

Allows a component to define a new class. As many new classes as necessary can be defined by any component.

Note: These classes are not related to the OLE and DDE classes defined under HKEY_LOCAL_MACHINE\SOFTWARE\Classes.

Class names do not need to be defined within any particular component. The system adds the new definition to its database without regard to origin. The order of **Class** entries is irrelevant. However, results are indeterminate if classes are referred to that are not defined anywhere in the system.

This entry is optional, because there are a few predefined classes, and class definitions made anywhere in the system apply to all network components. Because any network component can define new classes, be careful that the names used are unique within all possible installable network components. The following shows the predefined class names in the first release of Windows NT.

This list, of course, cannot be exhaustive.

Predefined class = Adapter card type

ee16Driver; ee16Adapter = Intel EtherExpress 16 LAN adapter

elinkiiAdapter; elinkiiDriver = 3Com Etherlink II adapter

ibmtokDriver; ibmtokAdapter = IBM Token Ring adapter

lanceDriver; dec101Adapter = DEC Lance adapter

lt200Driver; lt200Adapter = Daystar Digital LocalTalk adapter

ne2000Driver; ne2000Adapter = Novell NE2000 adapter

proteonDriver; p1390Adapter = Proteon adapter

ubDriver; ubAdapter = Ungermann-Bass Ethernet NIUpc adapter

wdlanDriver; smcisaAdapter = SMC (WD) adapter

The final optional value indicates whether this class is a "logical end-point for the bindings protocol; the default value is No.

Hidden

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard#*\NetRules

This value entry can appear under the NetRules subkey for network adapter cards.

Hidden REG_DWORD

Range: *0 or 1*

Suppresses the display of the component (adapter or network software) in the Network dialog box in Control Panel.

Usually, all networking components discovered in the Registry are displayed in the two list boxes in the Network dialog box in Control Panel. Setting this value to 1 prevents the item from being displayed, which means it cannot be configured or removed by the user.

Interface

This value entry can appear under the NetRules subkey for network adapter cards.

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard*\NetRules

Interface REG_MULTI_SZ

Range: *interfaceName* *upperClass* "*objectName*" *namingMethod*

Allows a single component to make available more than one type of capability to other components in the system. This value contains these items:

interfaceName = The tokenized name of the secondary interface.

upperClass = The class to which the interface belongs. (*LowerClass* is the same as the primary interface.)

objectName = The Windows NT device name to be created.

namingMethod = Determines how the bindings appear.

Review

This value entry can appear under the NetRules subkey for network adapter cards.

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard#*\NetRules

Review REG_DWORD

Range: 0 or 1

Indicates whether a component requests bindings review. If set to 1 (or nonzero), the system reinvokes this component's .INF file after bindings have been changed. This allows network components to modify the binding information or request additional information from administrators about the new or altered connections.

Type (NetRules)

This value entry can appear under the NetRules subkey for network adapter cards.

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard*\NetRules

Type REG_SZ

Range: *component className* [/lowerClass]

Defines the type of the component in terms of abstract network component classes. If the optional lower class name is absent, the first (or upper level) class type name is used for both its upper and lower classes.

This value is required for network software and network adapter cards. The component types are the following:

Adapter = A piece of hardware

Driver = A software component associated directly with a piece of hardware

Transport = A software component used by services

Service = A software component providing capability directly to user applications

Basic = A token used to represent a fundamental class name (that is, a class with no parent)

Use

This value entry can appear under the NetRules subkey for network adapter cards.

Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft*driverName*\CurrentVersion\NetRules

-Or-

HKEY_LOCAL_MACHINE\SOFTWARE
 \Microsoft\Windows NT\CurrentVersion
 \NetworkCards*netcard#*\NetRules

Use REG_SZ

Range: *service*[*driver*]*transport*[*adapter*] [Yes|No] [Yes|No]

Defines the role played by the component. If this entry is absent, the value of Service is assumed. This value entry only appears for software items.

A hardware device is automatically assumed to be an adapter. Each network component may identify itself as a driver, transport, or service to clarify its role. This distinction is as follows:

driver = Exists only to support one or more adapters. If no bindings are generated (or permitted by the user) that include a particular driver, that driver is not loaded. However, no error is generated, since no "denial of service has occurred.

service = Provides end-user functionality, and every attempt is made to support its operation. An EventLog entry is generated if a service is present in the system for which there is no available transport (the number of possible bindings is zero).

transport = Exists only to support services. Like a driver, it is not loaded unless necessary.

The final two Yes|No values in this entry are optional; if present, each must be either Yes or No. The first value indicates whether driver group names are used instead of specific driver dependencies. The second value indicates whether transport group names are used instead of specific transport dependencies. These values cause the system to generate references to dependencies based upon their group names, not by their specific service names.

For example, the LanmanServer is marked as Yes Yes; this means that its transport and driver dependencies are at the group level; so LanmanServer will be loaded if any one of its transport dependencies and any one of its driver dependencies successfully load.

Linkage Subkey Entries for Network Components

Each network component that the system knows is required can be given bindings, which define the relationships between network software components.

Caution: Bindings are created by the system and should not be altered by users.

Whether the bindings actually appear in the Registry depends upon the **Bindform** value for the component in its NetRules subkey. The binding information is stored in three value entries in the Linkage subkey for a component under its Services subkey in the following Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \ServiceName

 \Linkage

If a binding is disabled, the settings are stored under a Disabled subkey under the Linkage subkey. These values are in exactly the same format as for active bindings.

For a network component, there might be more than one subkey under its Services key: one for the driver and one for the network adapter card, plus additional subkeys for services that might be installed with the network adapter. Also, entries for adapter cards for AppleTalk, DLC (data link control), NWLink, and TCP/IP are described in their respective sections in Registry Entries for Network Services.

Bind **REG_MULTI_SZ**

Range: *ObjectName ObjectName ...*

Each string in this value entry is the name of a Windows NT object created by the underlying software. The names are based on the object names declared in the **Bindform** entry under the NetRules subkey.

Export **REG_MULTI_SZ**

Range: *ObjectName ObjectName ...*

Each string in this value entry indicates the name that should be added in the system to allow access to the corresponding bound object. The names are based on the object names declared in the **Bindform** entry under the NetRules subkey.

Route **REG_MULTI_SZ**

Range: *"Name of service" "Name of service"...*

Each string in this value entry indicates the exact path through the binding protocol represented by the binding. The names of services are used, surrounded by quotation marks.

Each element of these multistring values has a one-to-one correspondence with the others; that is, **Bind[1]** is to **Export[1]** and **Route[1]** as **Bind[2]** is to **Export[2]** and **Route[2]**.

In addition to generating binding information, the system determines each network component's complete set of dependencies, and stores them in the related subkey in the CurrentControlSet\Services key. For examples using these value entries, see "Bindings for Network Components in Chapter 11, "Overview of the Windows NT Registry, in the *Windows NT Resource Guide*.

Parameters Subkey Entries for Network Adapter Cards

Each network adapter card has a Parameters subkey with value entries that contain the settings for interrupt number, I/O port, and other parameters. These entries are found under subkeys for specific adapter cards in this Registry path:

HKEY_LOCAL_MACHINE\SYSTEM










 \CurrentControlSet

 \Services

 adapter name#

 \Parameters

These values should all be set by choosing the Network icon in Control Panel. The descriptions here are for informational purposes only.

	<u>BusNumber and BusType</u>
	<u>CableType and CardSpeed</u>
	<u>CardType</u>
	<u>DMACHannel, InterruptNumber, and IOBaseAddress</u>
	<u>MediaType</u>
	<u>MaximumPacketSize</u>
	<u>MemoryMapped and MemoryMappedBaseAddress</u>
	<u>NetworkAddress and Transceiver</u>
	<u>Default Settings for Network Adapter Card Parameters</u>

BusNumber and BusType

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
```

These values can appear in the Parameters subkey for a network adapter card.

BusNumber **REG_DWORD**

Range: *Number*

Default: Usually 0 (depends on the installation)

Defines the bus number, beginning with 0 in the common case where the computer has one bus type, whether it is ISA, EISA, MCA, or TurboChannel. For the rare computer that has more than one bus, bus number 2 has the value of 1, and so on.

This value should be maintained by the system. You cannot change it using the Network dialog box in Control Panel.

BusType **REG_DWORD**

Range: *Number*

Specifies the bus type for the computer, as in the following list:

0 = MIPS (Jazz-Internal bus)

1 = ISA bus

2 = EISA bus

3 = MCA bus

4 = TcChannel bus

This value should be maintained by the system. You cannot change it using the Network dialog box in Control Panel.

CableType and CardSpeed

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
```

These values can appear in the Parameters subkey for a network adapter card.

CableType **REG_DWORD**

Range: *1 or 2*

Specifies the cable type as unshielded twisted pair (UTP=1) or shielded twisted pair (STP=2), for Proteon 1390 adapter cards.

CardSpeed **REG_DWORD**

Range: *4 or 16 megabits per second*

Specifies card speed as 4 or 16 megabits per second in hex (0x4 or 0x10), for Proteon 1390 adapter cards.

CardType

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
```

CardType **REG_DWORD**

Range: *Number*

Specifies the card installed in the system. For some manufacturers, all their network cards use the same driver, which checks the value of **CardType** to determine the network card model number of the installed card. For example, the following shows **CardType** values for related network card model numbers.

DEC:

- 1 = DEC100
- 2 = DEC20x
- 3 = DEC PC
- 4 = DEC Station
- 5 = DEC422
- 7 = DEC101

Proteon:

- 1 = Proteon 1390
- 2 = Proteon 1990

Ungermann-Bass:

- 2 = UB PC
- 3 = UB EOTP
- 4 = UBPS

This value should be maintained by the system. You cannot change it using the Network dialog box in Control Panel.

DMAChannel, InterruptNumber, and IOBaseAddress

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 adapter name#
 \Parameters

DMAChannel **REG_DWORD**

Range: 5, 6, or 7

Specifies the DMA channel used by the adapter card.

InterruptNumber **REG_DWORD**

Range: *IRQ number*

Specifies the interrupt level (IRQ) for the adapter card. IRQ5 is a common choice (0x5).

IOBaseAddress **REG_DWORD**

Range: *Number in hex*

For some adapters, this entry specifies the I/O port base address as a hexadecimal string. For other adapters, this entry specifies whether this is the primary adapter card (1) or a secondary card (2).

MediaType

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
```

MediaType **REG_DWORD**

Range: *Number*

Specifies the network type, as follows:

- 1 = Ethernet
- 2 = Token Ring
- 3 = ARCnet
- 4 = FDDI network
- 5 = Apple LocalTalk

This value should be maintained by the system. You cannot change it using the Network dialog box in Control Panel.

MaximumPacketSize

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

adapter name#

\Parameters

MaximumPacketSize **REG_DWORD**

Range: *Number*

Specifies the maximum packet size that the IBM Token Ring adapter is allowed to transmit. Use this parameter when sending data across bridges that may have smaller packet sizes available on the destination network than on the native network. You cannot change this parameter using the Network dialog box in Control Panel.

MemoryMapped and MemoryMappedBaseAddress

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 adapter name#

 \Parameters

MemoryMapped **REG_DWORD**

Range: *0 or 1*

Specifies whether the adapter card is memory mapped.

MemoryMappedBaseAddress **REG_DWORD**

Range: *Memory address in hex*

Specifies the base memory (I/O) address used by the adapter card. This number must match the card's memory address settings as specified by its manufacturer.

NetworkAddress and Transceiver

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 adapter name#
 \Parameters

NetworkAddress **REG_SZ**

Range: *Number*

Specifies the address the network adapter should use instead of the burned-in address. For example, for the IBM Token Ring card, this value is 40000000203. You can change this value for Token Ring adapters using the Network dialog box in Control Panel, but not for Ethernet adapters.

Transceiver **REG_DWORD**

Range: *1 or 2*

Specifies the transceiver as External (1) or OnBoard (2). This value should be 1 for a DEC/Intel/Xerox (DIX) connection.

Default Settings for Network Adapter Card Parameters

The following summarizes default settings for various card types.

Settings are not required for EtherLink / MC, DEC PC, IBM Token Ring 16/4A, Proteon P1990, Novell NE3200, SMC/Western Digital 8013EA, and the Ungermann-Bass Ethernet NIUps (MC) or Ethernet NIUps/EOTP (short MC).

3Com EtherLink II, EtherLink II/TP, EtherLink II/16, or EtherLink II/16 TP:

InterruptNumber = 3

IoBaseAddress = 0x300

Transceiver = Internal

MemoryMapped = Off

3Com EtherLink 16/16 TP

InterruptNumber = 5

IoBaseAddress = 0x300

MemoryMappedAddress = 0xD0000

MemorySize = 16

Transceiver = Internal

DEC EtherWORKS Turbo EISA

SlotNumber = 1

DEC EtherWORKS LC, EtherWORKS Turbo, or EtherWORKS Turbo/TP

InterruptNumber = 5

MemoryAddress = 0xD0000

IoBaseAddress = Primary

IBMTOKEN Ring 16/4

IoBaseAddress = Primary

Novell NE2000

InterruptNumber = 3

IoBaseAddress = 0x300

ProteonP1390

InterruptNumber = 5

IoBaseAddress = 0xa20

DMACHannel = 5

CableType = STP

CardSpeed = 16

SMC/Western Digital 8003EP, 8013EWC, or 8013WB

InterruptNumber = 3

MemoryBaseAddress = 0xD000

IoBaseAddress = 0x300

Ungermann-Bass Ethernet NIUpc (long) or Ethernet NIUpc/EOTP (short)

InterruptNumber = 5

IoBaseAddress = 0x368

MemoryMappedAddress = 0xD8000

Registry Entries for Device Drivers

The following device drivers have additional value entries that can be specified in the Registry, in addition to changes that you can make using the Devices icon in Control Panel. This list is not exhaustive, but shows examples of the types of services and drivers that appear in the Registry.



File system drivers and recognizers:



Disk, serial, and parallel port devices:



Keyboard and mouse devices:



SCSI miniport devices:



Sound drivers






Video display devices:

Note: You can view current settings for these device drivers by choosing the Devices button in WinMSD.

File System Drivers and Recognizers

Each file system supported by Windows NT is made up of three components: the file system driver, the file system utility DLL, and the file system recognizer used during startup to determine the file systems present on the system. All necessary elements and settings are recognized automatically by Windows NT. You can configure file system drivers by choosing the Devices icon in Control Panel. The following file system drivers and recognizers are provided for these file systems:

- Cdfs and Cdfs_Rec: compact disc file system (CDFS)
-  Fastfat and Fat_Rec: File Allocation Table (FAT)
-  Ntfs and Ntfs_Rec: Windows NT file system (NTFS)
-  Pinball and HPFS_Rec: High-performance file system (HPFS)

The Registry path for settings that control file system drivers is the following, where *DriverName* is the file system driver minus the filename extension:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \DriverName
```

The standard entries and default values for the file system drivers are:

ErrorControl = 0x1 (Normal)
Group = Boot file system
Start = 0x4 (disabled)
Type = 0x2 (file system driver)

The file system recognizer determines whether the file system should be loaded. The Registry path for file system recognizers is the following, where *RecognizerName* is the file system driver minus the filename extension:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \RecognizerName
```

The standard entries for the file system recognizers are:

ErrorControl = 0 (startup halts)
Group = Boot file system
Start = 0x1 (system)
Type = 0x8 (file system recognizer)







The file system drivers and recognizers do not add any additional Registry entries besides the standard entries described in CurrentControlSet\Services Subkeys.

Note: If CONVERT.EXE has been used to convert the file system on a hard disk drive, this value is added to CurrentControlSet\Control\SessionManager so that conversion occurs when the system is restarted:

BootExecute = autocheck autoconv \Dos\Devices\X: /FS:NTFS

Disk, Serial, and Parallel Port Entries

This section provides general information about the Description entries for adapters in the Hardware key. Then specific information is presented about the DeviceMap subkey entries for AtDisk and for serial and parallel ports. Finally, Services subkey entries for parallel and serial ports are described, including specific entries for multiport serial I/O cards.

-  [Hardware Description Entries for Adapters](#)
-  [DeviceMap Subkey Entries for AtDisk](#)
-  [DeviceMap Subkey Entries for Serial and Parallel Ports](#)
-  [Parallel Subkey Entries in the Services Key](#)
-  [Serial Subkey Entries in the Services Key](#)
-  [Multiport Serial I/O Card Entries in the Services Key](#)

Hardware Description Entries for Adapters

The following shows the path for all *MultifunctionAdapter* entries:

```
HKEY_LOCAL_MACHINE\HARDWARE
  \Description
    \System
      \MultifunctionAdapter
        \0
          \ControllerName
            \0
```

The entries in this portion of the Registry contain data discovered by the Hardware Recognizer or provided from the ARC database that describes controllers for hard disks, display devices, the keyboard, pointing devices, and serial and parallel ports. Administrators cannot usefully modify entries in the Hardware key. This data is volatile (destroyed and recreated each time the system starts) and is useful only for informational purposes. You can use WinMSD to view this information in a more usable format.

Each subkey contains information in the following format:

Component Information **REG_BINARY**

Range: *System-defined*

Identifies version information plus other data for the associated subkey entry.

Configuration Data **REG_UNKNOWN**

Range: *System-defined*

Contains binary information related to the hardware component, such as I/O port addresses and IRQ number. This entry is not present if no such data is available for a particular subkey.

Identifier **REG_SZ**

Range: *Device type name*

Contains the name of a component, if specified.

The following samples from the *MultifunctionAdapter* subkeys describe a system that has a keyboard, a Microsoft InPort bus mouse or Microsoft Mouse Port mouse, and a Microsoft serial mouse (on COM1), all connected to the ISA bus. This sample is for informational purposes only, since these subkeys are volatile and are therefore recreated each time you start Windows NT.



Keyboard Adapter Example



Bus or Port Mouse Adapter Example



Serial Mouse Adapter Example

Keyboard Adapter Example

In this example, the **Identifier** value specifies the keyboard type name, which is typically PCAT_ENHANCED to indicate a 101/102-key enhanced keyboard. The keyboard type name is mainly informational, since the actual keyboard type and subtype are retrieved from the keyboard-specific data in **Configuration Data**.

Registry path:

HKEY_LOCAL_MACHINE\HARDWARE

 \Description

 \System

 \MultifunctionAdapter

 \0

 \KeyboardController

 \0

 \KeyboardPeripheral

 \0

Configuration Data : REG_UNKNOWN : *Device data (keyboard type, subtype)*

Identifier : REG_SZ : PCAT_ENHANCED

Bus or Port Mouse Adapter Example

This example shows a typical **Identifier** value for the pointer type name can also be MICROSOFT PS2 MOUSE (also known as the Mouse Port mouse) or MICROSOFT BUS MOUSE under this subkey:

```
HKEY_LOCAL_MACHINE    \HARDWARE
                        \Description
                          \System
                            \MultifunctionAdapter
                              \0
                                \PointerController
                                  \0
                                    \PointerPeripheral
                                      \0
```

Identifier : REG_SZ : MICROSOFT INPORT MOUSE

Serial Mouse Adapter Example

This example shows a typical entry for a serial mouse, under this subkey:

HKEY_LOCAL_MACHINE\HARDWARE

 \Description

 \System

 \MultifunctionAdapter

 \0

 \SerialController

 \0

 \PointerPeripheral

 \0

Identifier : REG_SZ : MICROSOFT SERIAL MOUSE

DeviceMap Subkey Entries for AtDisk

AtDisk is the driver for non-SCSI hard disk controllers on x86-based computers.

Note: The Abiosdisk driver has no Hardware key and no parameters that users can set under CurrentControlSet\Services.

The following Registry path can contain subkeys named Controllerx, where x starts at 0 and increases:

```
HKEY_LOCAL_MACHINE\HARDWARE
  \DeviceMap
    \AtDisk
```

These Controllerx subkeys are created for each non-SCSI hard disk controller on the system. As with all hardware data, these subkeys are volatile and so are recreated each time you start the system.

Under the Controllerx subkeys are subkeys named Disky, where y starts at 0 (zero) and increases. These subkeys are created for each actual disk controlled by the particular controller.

Under the Disky subkeys are the following value entries, which can be extremely helpful in reporting disk problems:

Firmware revision **REG_SZ**

Range: *Free format*

Defined by the disk manufacturer to identify the version of the on-board code used to control the disk.

Identifier **REG_SZ**

Range: *Free format*

Defined by the disk manufacturer to identify the make and model of the disk.

Number of cylinders **REG_DWORD**

Range: *A hex value*

The number of cylinders on the drive.

Number of heads **REG_DWORD**

Range: *A hex value*

The number of heads on the drive.

Sectors per track **REG_DWORD**

Range: *A hex value*

The number of sectors that exist on a track. These are typically 512-byte sectors.

The standard entries for AtDisk under the CurrentControlSet\Services subkey are the following:

ErrorControl = 0 (startup halts)

Group = Primary disk

Start = 0x0 (boot)

Tag = 0x2

Type = 0x1 (Kernel driver)

DeviceMap Subkey Entries for Serial and Parallel Ports

Remember that the entries in HKEY_LOCAL_MACHINE\HARDWARE are recreated each time the system is started. The entries in these subkeys are described here for informational purposes only.

ParallelX REG_SZ

Range: *A string, typically LPTy*

Registry path:

HKEY_LOCAL_MACHINE\HARDWARE
 \DEVICEMAP
 \PARALLEL PORTS

Specifies that the Windows NT Parallel device X is the actual device for the MSDOS name LPTy. These value entries are used to determine all the parallel ports available on the system.

SerialX REG_SZ

Range: *A string, typically COMy*

Registry path:

HKEY_LOCAL_MACHINE\HARDWARE
 \DeviceMap
 \Serialcomm

Specifies that the Windows NT Serial device X is the actual device for the MSDOS name COMy. These value entries are used to determine all the communication ports available on the system.

Parallel Subkey Entries in the Services Key

The following subkeys and values can be found under the following key:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Parallel
```

Subkeys and values under the Parallel subkey can be used to configure parallel (printer) ports in addition to information that the Hardware Detector finds at system startup.

This can also be used to override values determined by the Hardware Detector. If the **PortAddress** value entry is the same as a system-detected port, this data in the current control set will be used instead of the data found by the system. Entries appear in the System log in Event Viewer if this is occurring.

The Parallel subkey contains a subkey named Parameters, under which is a set of subkeys typically named Parallelx where x is some whole number. A system administrator must place these subkeys and values into the Registry. There exists no tool other than Registry Editor to define and manipulate these values.

For example, under the Parallel2 subkey, the following value entries can appear.

The first two of these value entries are required. If the section does not include both, the port is not configured and an error appears in the System log in Event Viewer. If an entry is placed into the Registry with just these values, the port will be driven using polling algorithms.

DosDevices **REG_SZ**

Range: *Free-form string*

Specifies the name used to access the parallel port from the command prompt or from within an application. A typical value would be LPT3.

PortAddress **REG_DWORD**

Range: *A hex value*

Denotes the address of the first register of the parallel port. A typical **PortAddress** in this case would be 0x278.

The following values are optional:

DisablePort **REG_DWORD**

Range: *0 or 1*

Default: 0

If the value is a 1, the device will be deleted after the port is reset during initialization. No access to the port will be allowed.

Interrupt **REG_DWORD**

Range: *A hex value*

Denotes the IRQ that the particular device would interrupt on. A typical **Interrupt** value in this case would be 0x5.

The standard entries for the Parallel subkey are the following:

ErrorControl = 0 (startup halts)

Group = Extended base

Start = 0x2 (autoload)

Type = 0x1 (Kernel device driver)

Serial Subkey Entries in the Services Key

The following subkeys and values can be found under the following key:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Serial
```

Subkeys and values under the CurrentControlSet\Services subkey can be used to configure serial ports in addition to information that the Hardware Detector finds at system startup. This can also be used to override values determined by the Hardware Detector. If the **PortAddress** value entry is the same as a system-detected port, this data in the current control set will be used instead of the data found by the system. Entries appear in the System log in Event Viewer if this is occurring.

The standard entries for the Serial subkey are the following:

ErrorControl = 0 (startup halts)

Group = Extended base

Start = 0x2 (autoload)

Type = 0x1 (Kernel device driver)



Deleting Serial Ports



Parameters for Serial Ports



Multiport Serial I/O Card Entries in the Services Key

Parameters for Serial Ports

The Serial subkey contains a subkey named Parameters, under which is a set of subkeys typically named SerialX where X is a whole number. A system administrator must place these subkeys and values into the Registry. You can only define and manipulate these values by using Registry Editor.

For example, under the Serial2 subkey, the following value entries can appear.

The first three of these value entries are required. If the subkey does not include all three, the port is not configured, and an error appears in the System log in Event Viewer.

DosDevices **REG_SZ**

Range: *Free-form string*

Specifies the name used to access the communication port from the command prompt or from within an application. A typical value would be COM3.

Interrupt **REG_DWORD**

Range: *A hex value*

Denotes the IRQ that the particular device would interrupt on. A typical **Interrupt** value in this case would be 0x4.

PortAddress **REG_DWORD**

Range: *A hex value*

Denotes the address of the first register of the serial device. A typical **PortAddress** in this case would be 0x3e8.

The following entry values are optional:

DisablePort **REG_DWORD**

Range: *0 or 1*

Default: 0

If the value is 1, the device will be deleted after the port is reset during initialization. No access to the port will be allowed.

ForceFifoEnable **REG_DWORD**

Range: *0 or 1*

Default: 1

If the value is 1 and the hardware supports a FIFO buffer (for example, the NS 16550AFN), the driver enables the FIFO. Not all FIFOs are reliable. If the application or the user notices lost data or no data transmission, it is recommended that this value be set to 0.

Multiport Serial I/O Card Entries in the Services Key

In addition to controlling the standard serial ports included with most PCs, the Microsoft serial driver can be used to control many *dumb* multiport serial cards. Dumb denotes that the control includes no onboard processor.

At least the following two additional value entries are used for each port on the multiport card. Each of these two entries must be included for each port of the multiport board under subkey entries in CurrentControlSet\Services:

InterruptStatus **REG_DWORD**

Range: *A hex value*

Denotes the address of the interrupt status register that indicates which port on the multiport card is actually requesting an interrupt. To determine the appropriate value, consult the manufacturer's installation guide.

PortIndex **REG_DWORD**

Range: *A hex value*

Denotes which port on the card this information is for. These values start at 1 and increase. Typically these would be the same as the values inscribed on the connector for the multiport.

Certain multiport boards, such as Digiboard non-MCA bus cards, use a different scheme to determine which port is interrupting. These boards should include the following value entry in the configuration data:

Indexed **REG_DWORD**

Range: *Should be 1*

Denotes that this board uses an indexed interrupt notification scheme as opposed to a bitmapped method. To determine whether this entry should be included, consult the board's manufacturer.



Example 1: 4-port CONTROL Hostess 550 board



Example 2: 8-port Digiboard communications board

Example 1: Controlling Multiport Serial I/O Cards

If you have a 4-port COMTROL Hostess 550 board configured to use address 0x500 with an interrupt of 0x2, the values in the Registry would be as follows, under CurrentControlSet\Services\SerialX:

Serial2 subkey:

PortAddress = REG_DWORD 0x500
Interrupt = REG_DWORD 2
DosDevices = REG_SZ COM3
InterruptStatus = REG_DWORD 0x507
PortIndex = REG_DWORD 1

Serial3 subkey:

PortAddress = REG_DWORD 0x508
Interrupt = REG_DWORD 2
DosDevices = REG_SZ COM4
InterruptStatus = REG_DWORD 0x507
PortIndex = REG_DWORD 2

Serial4 subkey:

PortAddress = REG_DWORD 0x510
Interrupt = REG_DWORD 2
DosDevices = REG_SZ COM5
InterruptStatus = REG_DWORD 0x507
PortIndex = REG_DWORD 3

Serial5 subkey:

PortAddress = REG_DWORD 0x518
Interrupt = REG_DWORD 2
DosDevices = REG_SZ COM6
InterruptStatus = REG_DWORD 0x507
PortIndex = REG_DWORD 4

Example 2: Controlling Multiport Serial I/O Cards

Certain multiport boards, such as Digiboard non-MCA bus cards, use a different scheme to determine which port is interrupting. These boards should include the value entry **Indexed** in the configuration data for each port under its subkey in CurrentControlSet\Services. This entry indicates that the board uses an indexed interrupt notification scheme as opposed to a bitmapped method.

For example, if you have an 8-port Digiboard communications board configured to be at address 0x100 with an interrupt of 0x3, the values in the Registry would be as follows, under CurrentControlSet\Services\SerialX:

Serial2 subkey:

PortAddress = REG_DWORD 0x100
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM3
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 1

Serial3 subkey:

PortAddress = REG_DWORD 0x108
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM4
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 2

Serial4 subkey:

PortAddress = REG_DWORD 0x110
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM5
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 3

Serial5 subkey:

PortAddress = REG_DWORD 0x118
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM6
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 4

Serial6 subkey:

PortAddress = REG_DWORD 0x120
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM7
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 5

Serial7 subkey:

PortAddress = REG_DWORD 0x128
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM8
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 6

Serial8 subkey:

PortAddress = REG_DWORD 0x130
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM9
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 7

Serial9 subkey:

PortAddress = REG_DWORD 0x138
Interrupt = REG_DWORD 3
DosDevices = REG_SZ COM10
InterruptStatus = REG_DWORD 0x140
Indexed = REG_DWORD 1
PortIndex = REG_DWORD 8

Deleting Serial Ports

You might need to delete one or more COM ports, which you can only do by deleting the related Registry information in the DeviceMap and Services areas of the Registry. To delete entries in the DeviceMap subkey, you must be logged on as a member of the Administrators group.

To delete a COM port

1. In Registry Editor, delete the appropriate value entry for the COM port under the following Registry path:

HKEY_LOCAL_MACHINE\Hardware\DeviceMap\SerialComm

The value entries in this subkey are used to determine all the communication ports available on the system. The values are of the form **Serialyyy=COMx**, where COMx can be COM1 to COM256, and Serialyyy can be from Serial0 through any large positive number such as Serial15000.

You can identify the value entry you want to delete based on the list that appears when you choose the Ports icon in Control Panel.







2. Delete the COM port's related Serialyyy key in the following Registry path:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Serial\Parameters\Serialyyy

If you do not delete the related Services subkey for Serialyyy, the COMx port continues to appear in the Ports list in Control Panel each time the system starts, even though there is no related DeviceMap subkey.

Mouse and Keyboard Driver Entries

Parameters in this section are for the mouse and keyboard class and port drivers, including the following:

-  Microsoft Bus Mouse Port Driver Entries
-  Intel 8042 Port Driver Entries
-  Microsoft Serial Mouse Port Driver Entries
-  Mouse Class Driver Entries
-  Keyboard Class Driver Entries
-  DeviceMap Entries for the Keyboard and Mouse

Microsoft Bus Mouse Port Driver Entries

The following value entries for the Microsoft bus mouse are found in this subkey:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Busmouse

 \Parameters

MouseDataQueueSize **REG_DWORD**

Range: $\geq 0x1$

Default: 0x64 (100)

Specifies the number of mouse events to be buffered internally by the driver, in nonpaged pool. The allocated size, in bytes, of the internal buffer is this value times the size of the MOUSE_INPUT_DATA structure (defined in NTDDMOU.H). Consider increasing the size if the System log in Event Viewer frequently contains this message from the Busmouse source: "The ring buffer that stores incoming mouse data has overflowed (buffer size is configurable via the Registry).

NumberOfButtons **REG_DWORD**

Range: $\geq 0x1$

Default: 0x2 (two buttons)

Specifies the number of buttons on the bus mouse. If the number of buttons detected at startup time and placed in the Registry is incorrect, this value can be used to override it.

PointerDeviceBaseName **REG_SZ**

Range: *Base port device name*

Default: PointerPort

Specifies the base name for the device object(s) created by the bus mouse port device driver. The device driver also writes information about the device objects into HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap so that the pointer class driver can locate the pointer port device objects.

SampleRate **REG_DWORD**

Range: *Mouse sample rate in Hz*

Default: 0x32 (50 Hz)

Specifies the sample rate for the bus mouse. Intended for future use. This value might have no effect in the first release of Windows NT.

Intel 8042 Port Driver Entries

The i8042prt driver handles the keyboard and mouse port mouse (also known as a PS/2-compatible mouse) for the Intel 8042 controller.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \i8042prt
 \Parameters

KeyboardDataQueueSize **REG_DWORD**

Range: $\geq 0x1$

Default: 0x64 (100)

Specifies the number of keyboard events to be buffered internally by the driver, in nonpaged pool. The allocated size, in bytes, of the internal buffer is this value times the size of the KEYBOARD_INPUT_DATA structure (defined in NTDDKBD.H). Consider increasing the size if the System log in Event Viewer contains the following message from the i8042prt source: "The ring buffer that stores incoming keyboard data has overflowed (buffer size is configurable via the Registry).

KeyboardDeviceBaseName **REG_SZ**

Range: *Base keyboard port device name*

Default: KeyboardPort

Specifies the base name for the keyboard device object(s) created by the i8042prt device driver. The device driver also writes information about the device objects into HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap so that the keyboard class driver can locate the keyboard port device objects.

MouseDataQueueSize **REG_DWORD**

Range: $\geq 0x1$

Default: 0x64 (100)

Specifies the number of mouse events to be buffered internally by the driver, in nonpaged pool. Consider increasing the size if the System log in Event Viewer contains the following message from the i8042prt source: "The ring buffer that stores incoming mouse data has overflowed (buffer size is configurable via the Registry).

MouseResolution **REG_DWORD**

Range: *Mouse resolution specifier*

Default: 0x3

Used in specifying the mouse port mouse resolution, where 2 to the power of **MouseResolution** specifies counts-per-millimeter.

MouseSynchIn100ns **REG_DWORD**

Range: *Time, in 100 nanosecond units*

Default: 10000000 (1 second)

Specifies the length of time after which the next mouse interrupt is assumed to indicate the start of a new mouse packet (partial packets are discarded). This allows the driver to synchronize its internal notion of the mouse packet state with the hardware state, in the event that a mouse interrupt has been lost. Consider modifying this value if the system behaves as if there are random mouse events occurring (for example, button clicks when no mouse button has been pressed).

NumberOfButtons **REG_DWORD**

Range: $\geq 0x1$

Default: 0x2

Specifies the number of buttons on the mouse port mouse. If an incorrect number of buttons is detected at startup time and placed in the Registry, this value can be used to override it.

PointerDeviceBaseName **REG_SZ**

Range: *Base mouse port device name*

Default: PointerPort

Specifies the base name for the pointer device object(s) created by the mouse port mouse driver. The device driver also writes information about the device object into this Registry path:

HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap

so that the pointer class driver can locate the pointer port device objects.

PollStatusIterations **REG_DWORD**

Range: *Number*

Default: 1

Specifies the maximum number of times to check the i8042 controller status register for interrupt verification before dismissing the interrupt as spurious. This value can be used to work around a problem experienced on some hardware (including the Olivetti MIPS computers) where the keyboard interrupt is raised before the Output Buffer Full bit is set in the i8042 status register.

Increase this value if the system seems to suddenly stop taking keyboard interrupts. This can happen if a keyboard interrupt is mistakenly dismissed as spurious, when instead it just took too long to set Output Buffer Full after raising the interrupt. Increasing the value of **PollStatusIterations** results in a longer execution time for the Interrupt Service Routine if the keyboard interrupt truly is spurious (there is a 1 microsecond delay following each check for Output Buffer Full).

To determine whether the driver is taking keyboard interrupts, press the NumLock key. If the NumLock light on the keyboard turns on or off, this indicates that the i8042prt driver handled the keyboard interrupt correctly.

OverrideKeyboardType **REG_DWORD**

Range: $\geq 0x0$

Default: 0x4 (Enhanced 101-key or 102-key keyboard)

This entry is not usually present. When present, it specifies the keyboard type (overriding the keyboard type detected during system initialization). Add this value entry if the detected keyboard type is incorrect in the Registry. Type values 0x2 and 0x4 indicate an enhanced 101-key or 102-key keyboard, or compatible; other values typically indicate an old-style AT keyboard (83, 84, or 86 keys), or compatible.

OverrideKeyboardSubtype **REG_DWORD**

Range: $\geq 0x0$

Default: 0x0

This entry is not usually present. When present, it specifies the OEM-dependent keyboard subtype (overriding the keyboard subtype detected during system initialization).

PollingIterations **REG_DWORD**

Range: $\geq 0x400$

Default: 0x400

Specifies the standard number of times to poll the hardware (in polling mode) before giving up and timing out the operation. Consider increasing this value if the driver fails to initialize or work correctly

and the System log in Event Viewer contains the following message from the i8042prt source: "The operation on ... timed out (time out is configurable via the Registry).

PollingIterationsMaximum REG_DWORD

Range: $\geq 0x400$

Default: 0x2EE0

Specifies the maximum number of times to poll the hardware (in polling mode) before giving up and timing out the operation. This value is used instead of **PollingIterations** when an old-style AT keyboard is detected (see **OverrideKeyboardType** in this section).

Consider increasing this value if the driver fails to initialize or work correctly and the System log in Event Viewer contains the following message from the i8042prt source: "The operation on ... timed out (time out is configurable via the Registry).

ResendIterations REG_DWORD

Range: $\geq 0x1$

Specifies the maximum number of times a hardware operation will be retried before timing out. Consider increasing this value if the driver fails to initialize or work correctly and the System log in Event Viewer contains the following message from the i8042prt source: "Exceeded the allowable number of retries (configurable via the Registry) on device ...

Default: 0x3

SampleRate REG_DWORD

Range: *Mouse sample rate in Hz*

Default: 0x3C (60 Hz)

Specifies the sample rate for the mouse. Intended for future use. This value might have no effect in the first release of Windows NT.

Microsoft InPort Bus Mouse Port Driver Entries

The value entries for the Microsoft InPort bus mouse are found in the following subkey:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Inport

 \Parameters

HzMode **REG_DWORD**

Range: *Mouse sample rate mode specifier*

Default: 0x2 (selects 50 Hz)

Specifies the value written to the Microsoft InPort mode register to set the mouse sample rate. Might be used in the first release of Windows NT, but should become obsolete.

MouseDataQueueSize **REG_DWORD**

Range: *>= 0x1*

Default: 0x64 (100)

Specifies the number of mouse events to be buffered internally by the driver, in nonpaged pool. The allocated size, in bytes, of the internal buffer is this value times the size of the MOUSE_INPUT_DATA structure (defined in NTDDMOU.H). Consider increasing the size if the System log in Event Viewer contains the following message from the InPort source: "The ring buffer that stores incoming mouse data has overflowed (buffer size is configurable via the Registry).

NumberOfButtons **REG_DWORD**

Range: *>= 0x1*

Default: 0x2

Specifies the number of buttons on the Microsoft InPort bus mouse. If the number of buttons detected at startup time and placed in the Registry is incorrect, this value can be used to override it.

PointerDeviceBaseName **REG_SZ**

Range: *Base port device name*

Default: PointerPort

Specifies the base name for the device object(s) created by the Microsoft InPort bus mouse device driver. The device driver also writes information about the device objects into this path:

HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap

so that the pointer class driver can locate the pointer port device objects.

SampleRate **REG_DWORD**

Range: *Mouse sample rate in Hz*

Default: 0x32 (50 Hz)

Specifies the sample rate for the Microsoft InPort bus mouse. Intended for future use. This value might have no effect in the first release of Windows NT.

Microsoft Serial Mouse Port Driver Entries

The value entries for the Microsoft serial mouse are found in the following subkey:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Services
            \Sermouse
                \Parameters
```

MouseDataQueueSize **REG_DWORD**

Range: $\geq 0x1$

Default: 0x64 (100)

Specifies the number of mouse events to be buffered internally by the driver, in nonpaged pool. The allocated size, in bytes, of the internal buffer is this value times the size of the MOUSE_INPUT_DATA structure (defined in NTDDMOU.H). Consider increasing the size if the System log in Event Viewer contains the following message from the Sermouse source: "The ring buffer that stores incoming mouse data has overflowed (buffer size is configurable via the Registry).

NumberOfButtons **REG_DWORD**

Range: $\geq 0x1$

Default: 0x2

Specifies the number of buttons on the serial mouse. If the number of buttons detected at startup time and placed in the Registry is incorrect, this value can be used to override it.

OverrideHardwareBitstring **REG_DWORD**

Range: 0x1 or 0x2

This entry is not usually present. When present, it specifies that regardless of whether it was actually detected, a serial mouse is present on the system. Add this value to tell the driver to assume the serial mouse is on COM1 (specified by the value 0x1) or COM2 (specified by the value 0x2). This entry is useful if the serial mouse has not been automatically detected.

PointerDeviceBaseName **REG_SZ**

Range: *Base port device name*

Default: PointerPort

Specifies the base name for the device object(s) created by the serial mouse device driver. The device driver also writes information about the device objects into HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap so that the pointer class driver can locate the pointer port device objects.

SampleRate **REG_DWORD**

Range: *Mouse sample rate in Hz*

Default: 0x28 (1200 baud)

Specifies the sample rate for the serial mouse. Intended for future use. This value might have no effect in the first release of Windows NT.

Mouse Class Driver Entries

The value entries for the mouse class driver are found in the following subkey:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Services
            \Mouclass
                \Parameters
```

ConnectMultiplePorts **REG_DWORD**

Range: *0x0 or 0x1*

Default: 0x1 (The events generated by up to the **MaximumPortsServiced** number of pointing devices on the system will all be fed to the Windows subsystem in a single input stream.)

Specifies the type of connection between class and port device objects. This parameter is mainly of interest to device driver writers.

The value 0x0 specifies a 1:1 relationship between class device objects and port device objects. (That is, one class device object is created by the driver and connected to one port device object; the maximum number of objects created and connected to an associated port object is determined by the value of **MaximumPortsServiced**.) The value 0x1 specifies a 1:many relationship between a single class device object and multiple port device objects. (That is, one class device object is created by the driver and then connected to multiple port device objects, up to a maximum specified by **MaximumPortsServiced**.)

MaximumPortsServiced **REG_DWORD**

Range: *>= 0x1*

Default: 0x3 (The class driver will service up to three pointing devices.)

Specifies the number of port devices the mouse class device driver will connect to and service. The class device driver handles hardware-independent operations on a specific class of devices (in this case, the mouse and other pointing devices). The port drivers manage the hardware-specific operations.

MouseDataQueueSize **REG_DWORD**

Range: *>= 0x1*

Default: 0x64 (100)

Specifies the number of mouse events to be buffered internally by the driver, in nonpaged pool. The allocated size, in bytes, of the internal buffer is this value times the size of the MOUSE_INPUT_DATA structure (defined in NTDDMOU.H). Consider increasing the size if the System log in Event Viewer frequently contains the following message from the Mouclass source: "The ring buffer that stores incoming mouse data has overflowed (buffer size is configurable via the Registry).

PointerDeviceBaseName **REG_SZ**

Range: *Base class device name*

Default: PointerClass

Specifies the base name for the device object(s) created by the mouse class device driver. The device driver also writes information about the device object into

HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap so that the pointer class device object(s) can be easily located.

Keyboard Class Driver Entries

The value entries for the keyboard class driver are found in the following subkey:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Services
            \Kbdclass
                \Parameters
```

ConnectMultiplePorts **REG_DWORD**

Range: *0x0 or 0x1*

Default: 0x0 (The events generated by up to the **MaximumPortsServiced** number of keyboard devices on the system will feed separate input streams. In the first release of Windows NT, the Windows subsystem only reads from a single keyboard input stream.)

Specifies the type of connection between class and port device objects. This parameter is mainly of interest to device driver writers.

The value 0x0 specifies a 1:1 relationship between class device objects and port device objects. (That is, one class device object is created by the driver and connected to one port device object; the maximum number of objects created and connected to an associated port object is determined by the value of **MaximumPortsServiced**.) The value 0x1 specifies a 1:many relationship between a single class device object and multiple port device objects (That is, one class device object is created by the driver and then connected to multiple port device objects, up to a maximum specified by **MaximumPortsServiced**.)

KeyboardDataQueueSize **REG_DWORD**

Range: *>= 0x1*

Default: 0x64 (100)

Specifies the number of keyboard events to be buffered internally by the driver, in nonpaged pool.

The allocated size, in bytes, of the internal buffer is this value times the size of the `KEYBOARD_INPUT_DATA` structure (defined in `NTDDKBD.H`). Consider increasing the size if the System log in Event Viewer contains the following message from the Kbdclass source: "The ring buffer that stores incoming keyboard data has overflowed (buffer size is configurable via the Registry).

KeyboardDeviceBaseName **REG_SZ**

Range: *Base class device name*

Default: KeyboardClass

Specifies the base name for the keyboard device object(s) created by the keyboard class device driver. The device driver also writes information about the device objects into `HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap` so that the keyboard class device objects are easily located.

MaximumPortsServiced **REG_DWORD**

Range: *>= 0x1*

Default: 0x3 (The class driver will service up to three keyboard devices.)

Specifies the number of port devices the keyboard class device driver will connect to and service. The class device driver handles hardware-independent operations on a specific class of devices (in this case, keyboard devices). The port drivers manage the hardware-specific operations.

DeviceMap Entries for the Keyboard and Mouse

The following DeviceMap descriptions are for informational purposes only, since the DeviceMap subkeys are volatile and are recreated each time you start Windows NT. Administrators cannot modify DeviceMap entries.

These DeviceMap entries are used by the Windows subsystem to locate the pointer and keyboard class devices, and by the pointer and keyboard class drivers to locate the associated pointer and keyboard port devices. Information is placed in the DeviceMap subkey by the keyboard and pointer class and port drivers.

The format for each of these entries is the following:

Name of class device object : REG_SZ : Registry path to driver's Services

The keyboard class information appears in the following Registry path:

```
HKEY_LOCAL_MACHINE\HARDWARE
  \DeviceMap
    \KeyboardClass
```

There can be one or more of these entries. Each entry specifies the name of a device object created by the keyboard class driver to represent the class device, followed by the Registry path to the driver's Services subkey.

Default:

```
\Device\KeyboardClass0 : REG_SZ : \Registry\Machine\System\ControlSet001\Services\Kbdclass
```

The keyboard port information appears in the following Registry path:

```
HKEY_LOCAL_MACHINE\HARDWARE
  \DeviceMap
    \KeyboardPort
```

There can be one or more of these entries. Each entry specifies the name of a device object created by the keyboard port driver(s) to represent the physical keyboard (port) device, followed by the Registry path to the driver's Services subkey.

Default:

```
\Device\KeyboardPort0 : REG_SZ : \Registry\Machine\System\ControlSet001\Services\i8042prt
```

The mouse class information appears in the following Registry path:

```
HKEY_LOCAL_MACHINE\HARDWARE
  \DeviceMap
    \PointerClass
```

There can be one or more of these entries. Each entry specifies the name of a device object created by the pointer (mouse) class driver to represent the class device, followed by the Registry path to the driver's Services subkey.

Default:

```
\Device\PointerClass0 : REG_SZ : \Registry\Machine\System\ControlSet001\Services\Mouclass
```

The mouse port information appears in the following Registry path:

```
HKEY_LOCAL_MACHINE\HARDWARE
  \DeviceMap
    \PointerPort
```

There can be one or more of these entries. Each entry specifies the name of a device object created by the pointer port driver(s) to represent the physical pointing (port) device, followed by the Registry path to the driver's Services subkey.

Default (assumes mouse port, Microsoft InPort, and serial pointing devices are connected):

\Device\PointerPort0 : REG_SZ :

\Registry\Machine\System\ControlSet001\Services\i8042prt

\Device\PointerPort1 : REG_SZ :

\Registry\Machine\System\ControlSet001\Services\Inport

\Device\PointerPort2 : REG_SZ :

\Registry\Machine\System\ControlSet001\Services\Sermouse

SCSI Miniport Driver Entries

The basic SCSI miniport driver entries in the Registry are found under subkeys in the following path:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

Each subkey's name is the same as the driver's filename minus the .SYS filename extension; for example, FD8XX, which is the entry for all Future Domain 800-series SCSI adapter.

List of SCSI miniport device drivers

The contents of a SCSI miniport subkey are standard for all SCSI miniport drivers, with these basic value entries:

ErrorControl = 0x01 (startup continues)

Group = SCSI Miniport

Start = 0x00 (Auto Start)

Tag = Optional (determines the load order of SCSI miniport drivers)

Type = 0x01 (device driver).

For each SCSI miniport key, there can be one or more subkeys named Parameters\Device or Parameters\Device N , where $N = 0, 1, 2$, and so on. The value of N corresponds to the SCSI host adapter number. If the subkey name is Device, the value is globally defined. If the subkey name is Device N , the value only pertains to the particular SCSI host adapter.

The SCSI miniport driver recognizes several optional value entries that can be defined under the Parameters\Device subkeys for SCSI adapters:



Basic SCSI Parameters\Device Entries



SCSI Parameters\Device Entries for Troubleshooting



DriverParameter Values for SCSI Device Drives

Basic SCSI Parameters\Device Entries

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Device name

 \Parameters

InitiatorTargetId **REG_DWORD**

Range: *Number*

Default: Uninitialized

Sets the SCSI bus host adapter ID. It is used by host adapters that can set the initiator ID from software.

MaximumLogicalUnit **REG_DWORD**

Range: *Number*

Default: 8

Controls the number of logical units per target controller that are scanned for by the SCSI miniport driver. Most devices only support one logical unit, and some devices may fail if more than one logical unit is scanned for.

ScsiDebug **REG_DWORD**

Range: *Number*

Default: 0

This value is used to set the value of the ScsiDebug variable, which controls the verbosity of DebugPrint, with 0 being the least verbose. This is used for debugging.

SCSI Parameters\Device Entries for Troubleshooting

The following value entries are used to fix problems such as device time-outs or controller detection errors but will reduce I/O performance. These value entries can be abbreviated. For example, a value entry of **Disable** will cause **DisableSynchronousTransfers**, **DisableTaggedQueuing**, **DisableDisconnects**, and **DisableMultipleRequests** to be set.

Note: The system must be restarted before these options take effect.

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Device name
 \Parameters

BreakPointOnEntry **REG_DWORD**

Range: 0 or 1

Default: 0 (disabled)

A DbgBreakPoint() call is immediately made inside of SpParseDevice. This is used for debugging.

DisableDisconnects **REG_DWORD**

Range: 0 or 1

Default: 1 (enabled)

Disables disconnects on the SCSI bus. It causes all requests to be executed sequentially.

DisableMultipleRequests **REG_DWORD**

Range: 0 or 1

Default: 1 (enabled)

Prevents the SCSI miniport driver from sending more than one request at a time per SCSI device.

DisableSynchronousTransfers **REG_DWORD**

Range: 0 or 1

Default: 1 (enabled)

Disables synchronous data transfers on the SCSI bus.

DisableTaggedQueuing **REG_DWORD**

Range: 0 or 1

Default: 1 (enabled)

Disables SCSI-II tagged command queuing on the host adapter.

DriverParameter *Data type is specific to driver*

Range: A string

A pointer to this data is passed to the SCSI miniport driver in a miniportFindAdapter routine. It is the fourth parameter, ArgumentString. A miniport driver uses this data to define the IRQ number for the SCSI host adapter, but other applications for the data are possible.

The data type for this value is defined by the specific SCSI miniport driver developer. If the data type is REG_SZ, the Unicode string is converted to an ANSI string before transferring it to the SCSI miniport driver.



DriverParameter Values for SCSI Device Drives

DriverParameter Values for SCSI Device Drives

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Device name
 \Parameters

The following drivers currently use the **DriverParameter** value entry:

Aha154x :

BusOnTime=xx

Default: 7

xx is the bus on time in microseconds for the card. Valid values are 2-15. The value is usually adjusted downward when DMA transfers from the Adaptec card are interfering with other DMA transfers.

FD8XX

IRQ=xx

Default: 5

xx is the IRQ the card should use. Valid values are 0, 3, 4, 5, 10, 11, 12, 14, 15. This value should match the jumper settings on the card.

Numbers 0, 3, and 5 are for the short cards (850, 845); the rest are for the 885 card only. A value of 0 indicates the card should not use any interrupts and will poll.

T128:

IRQ=xx

Default: 5

xx is the IRQ the card should use. Valid values are: 0, 3, 5, 7, 10, 12, 14, and 15. This value should match the jumper settings on the card.

Numbers greater than 7 are for the T128F card only. A value of 0 indicates the card should not use any interrupts and will poll.

T13B:

IRQ=xx

Default: 5

xx is the IRQ the card should use. Valid values are 0, 3, 5, and 7. This value should match the jumper settings on the card.

A value of 0 indicates the card should not use any interrupts and will poll.

TMV1:

IRQ=xx

Default: 10

xx is the IRQ the card should use. Valid values are: 2, 3, 4, 5, 6, 7, 10, 11, 12, and 15.

Wd33c93:

IRQ=xx; DMA=yy

Default: **IRQ=10; DMA=6**

xx is the IRQ the card should use. Valid values are: 3, 4, 5, 10, 11, 12, and 15.

yy is the DMA channel the card should use. Valid values are: 5, 6, and 7.

SCSI miniport device drivers

The Registry includes entries for at least the following SCSI miniport device drivers:

Ahaxxx = Adaptec 154x and 174x SCSI adapters

DptScsi = DPT SCSI adapter

Fd16_700, Fd7000ex, Fd8xx = Future Domain MCS 600/MCS 700, TMC-7000ex, and 800-series SCSI adapters

Ncr53c9x, Ncr700, Ncr710 = NCR SCSI controller and adapters

Oliscsi = Olivetti SCSI adapter

Sparrow = SCSI adapter

Spock = SCSI adapter

T128 and T13B = Trantor SCSI adapters

Ultraxxx = UltraStor 124, 14f, and 24f SCSI adapters

Wd33c93 = Maynard SCSI adapter

Video Device Driver Entries

This section describes the entries for video device drivers under the DeviceMap subkey and under the CurrentControlSet\Services subkeys for specific video drivers.



Video Information in the DeviceMap Subkey



Video Driver Entries in the Services Subkey



DefaultSettings.xxx Entries for Video Drivers



VgaCompatible Values for Video Drivers



Other Value Entries for Video Drivers

Video Information in the DeviceMap Subkey

The hardware device mapping for video is under the following subkey:

HKEY_LOCAL_MACHINE\HARDWARE

\DeviceMap

\Video

This information is volatile and is reconstructed at startup by the video port driver. It can change from startup to startup based on external factors, such as failure to initialize a video adapter or the addition of other video cards to the system.

This subkey contains the mappings from Windows NT logical video devices to the physical device they represent in the CurrentControlSet\Services subkey. This mapping allows the system to find the right display driver for the currently installed video device.

*%device_object_name% **REG_SZ** Registry path for device*

Indicates the first logical video device is the first physical adapter. For example, the following entry indicates that the first logical video device is the second physical XGA adapter:

Video\Device0 =

\Registry\Machine\CurrentControlSet\Services\XGA\Device1

In this example, the value indicates that the second logical video device is the first physical VGA adapter:

Video\Device1 =

\Registry\Machine\CurrentControlSet\Services\Vga\Device0

These values point to entries in the Services subkey, as described [Video Driver Entries in the Services Subkey](#).

Sound Card Driver Entries in the Services Key

Usually, sound drivers consist of two parts: a front-end for installation and request processing, consisting of files such as SNDBLST.DLL or SYNTH.DLL; and a kernel driver for communicating with the hardware, consisting of files such as SNDBLST.SYS or SYNTH.SYS.

There is also a helper driver named MMDRV.DLL that transforms most low-level calls to Wave, MIDI, and AUX devices into calls to kernel mode drivers.

These are the installable sound kernel drivers for Windows NT:

MIPSSND.SYS (MIPS ARCSound 100/150 built-in sound)

MVAUDIO.SYS (Media Vision Pro Audio Spectrum 16 and Pro Studio 16)

MVOPL3.SYS (Synthesizer driver for Media Vision Pro Audio Spectrum 16 and Pro Studio 16)

SNDBLST.SYS (SoundBlaster 1.5 and compatibles)

SNDSYS.SYS (Windows sound system and Compaq Business Audio)

SYNTH.SYS (Ad Lib and OPL3 MIDI synthesizer driver)

For each installed sound driver, several parameters are stored in the Registry, based on choices made by using the Drivers icon in Control Panel.

Registry path for sound system driver entries:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \SoundDriverName
        \Parameters
```

The actual key name in the Registry is the same as the filename of the related kernel driver. For example, for SNDBLST.SYS, the key name is Sndblst.

Some drivers can write over the values they receive on the basis of information read from the hardware. The following shows some typical values found in the Parameters subkey for sound cards:

Configuration Error REG_DWORD

Range: 0, 1, 2, 3, or 4

Specifies an error that occurred during install. This value is only relevant if the driver fails to load. (Not all drivers write this information into the Registry.)

0 = Nonspecific error

1 = Hardware not found (usually the wrong I/O port was assumed)

2 = Specified interrupt was incorrect or did not work

3 = Specified DMA channel was incorrect or did not work

4 = Hardware is present but not working

DmaChannel REG_DWORD

Default: Depends on the sound card not user-configurable.

Defines the DMA channel settings for transferring digitized sound.

Interrupt REG_DWORD

Range: *Interrupt number*

Default: 0xa (configured for Interrupt 10)

Defines the interrupt number used by the hardware.

LeftLineInAtten REG_DWORD

Range: *Number*

Specifies the current volume level of the left channel of the line-in input.

Port **REG_DWORD**

Range: *Address*

Default: 0x220 for SNDBLST.SYS

Defines the I/O port start address used to communicate with the hardware.

RightLineInAtten **REG_DWORD**

Range: *Number*

Specifies the current volume level of the right channel of the line-in input.

Video Driver Entries in the Services Subkey

The port driver portion of the video driver is hardware-independent and contains operating system-specific code. Therefore, the port driver, VIDEOprt.SYS, can support one or more video devices. The Services\Videoprt subkey has no added parameters, and its standard entries are the following:

ErrorControl = 0x1 (Normal)

Group = Video

Start = 0x1 (system)

Type = 0x1 (Kernel driver)

The specific subkey for each video driver contains all the information required to initialize and program the device properly. If several adapters can be handled by a single driver, the subkeys Device1, Device2, and so on will contain information for the other devices. The Registry path looks like this, where *VideoDriverName* is the name of a specific video device driver:




```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \VideoDriverName
        \Device0
```

For example, the following subkey contains information for the first logical device of type VGA:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Vga
        \Device0
```

Examples of VideoDriverName subkeys

The following values can be set in a video driver subkey:

-  DefaultSettings.xxx Entries for Video Drivers
-  VgaCompatible Values for Video Drivers
-  Other Value Entries for Video Drivers

VideoDriverName subkeys

The *VideoDriverName* subkeys for drivers in Windows NT include the following. This is not an exhaustive list:

- Ati
- ET4000
- Jazzgxxx
- S3
- Trident
- Vga
- Wdvg
- Xga

DefaultSettings.xxx Entries for Video Drivers

Registry path:
HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \VideoDriverName
 \Device0

DefaultSettings.BitsPerPel REG_DWORD

Range: *Number of bits per pixel*

Contains the number of colors for the mode requested by the user. For example, for the v7vram miniport, the following value yields a 256-color mode:

DefaultSettings.BitsPerPel = 8

DefaultSettings.Interlaced REG_DWORD

Range: *0 or 1*

Determines whether the mode requested by the user is interlaced. For example, for the v7vram miniport:

DefaultSettings.Interlaced = 0x0 (FALSE)

DefaultSettings.VRefresh REG_DWORD

Range: *Number Hz*

Contains the refresh rate of the mode requested by the user. For example, for the et4000 miniport:

DefaultSettings.VRefresh = 72

DefaultSettings.XResolution REG_DWORD

Range: *Number of pixels*

Contains the width of the mode requested by the user. For example, for the et4000 miniport:

DefaultSettings.Xresolution = 1024

DefaultSettings.YResolution REG_DWORD

Range: *Number of pixels*

Contains the height of the mode requested by the user. For example, for the et4000 miniport:

DefaultSettings.Yresolution = 768

Other Value Entries for Video Drivers

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \VideoDriverName
        \Device0
```

DeviceData **REG_BINARY**

Range: *Binary code*

Contains binary data specific to the Windows display driver. For example, for the VGA miniport:

DeviceData = 05a0 5075 8ef0 8456 c8dd

InstalledDisplayDrivers **REG_MULTI_SZ**

Range: *Driver names*

Contains a list of names of display drivers that can function with this miniport, depending on which mode is selected by the user.

Note: Display driver names do not contain the .DLL filename extension.

The system attempts to initialize the adapter by calling each display driver, using the user-selected parameter. If the combination of display driver and monitor do not support the mode requested by the user, the display driver fails to initialize and the system tries the next display driver. If all display drivers fail to initialize, the system calls the first display driver again to set the adapter to any mode it can.

For example, for the et4000 miniport:

InstalledDisplayDrivers = "vga" "vga256" "vga64k"

For the S3 miniport:

InstalledDisplayDrivers = "s3"

Monitor **REG_SZ**

Range: *Monitor name*

This entry is reserved to contain the name of the VESA VDIIF monitor information file for the monitor connected to the adapter. No such files are provided with Windows NT in the current version.

If a file is supplied and a value is added, the miniport can load this file to determine the exact timings of the monitor connected to the physical device.

For example, for the XGA miniport, if a NEC4FS monitor was attached to the XGA card:

Monitor = "NEC4fg.vdb"

VgaCompatible Values for Video Drivers

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \VideoDriverName
        \Device0
```

VgaCompatible **REG_DWORD**

Range: 0 or 1

Determines whether the driver supports all of the VGA functionality required to perform full-screen operations.

If this value is 1, the driver supports full-screen applications in x86-based computers. If this value is 0, the VGA miniport (described under the Vga subkey) will also be used to enable and disable full-screen modes for non-Windows applications.

As a general rule, all drivers for SVGA adapters should set this value to 1, because they must implement all the VGA functionality to perform extended save/restore of all registers. A video accelerator designed as an accelerator working independently of the VGA miniport (via pass-through) can set this to 0 and let the VGA miniport do all the full-screen work.

For example, for the et4000 miniport:

VgaCompatible = 0x1 (TRUE)

For the S3 miniport:

VgaCompatible = 0x0 (FALSE)

This functionality is not required for other computer platforms, such as RISC-based computers, because the VDM sessions are emulated using NTVDM.EXE, and there are no full-screen sessions.




















Registry Entries for Network Services

The following services have additional value entries that can be specified in the Registry. You can view current settings for these services by choosing the Services button in WinMSD.

This section describes parameters for these services under the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services subkey. Services not included here do not have hidden parameters that you can set.

Some of these services also have configuration information described in [NetRules Subkey Entries](#).

Note: Wherever possible, choose the Services icon in Control Panel or use Server Manager in Windows NT Advanced Server to change values for these services.

	Alert service entries
	AppleTalk and MacFile entries for SFM
	BootVerification service entries
	Browser service entries
	DiskPerf service entries
	DLC system driver entries
	EventLog service entries
	NBF (NetBEUI) transport entries
	NetLogon service entries
	NetworkProvider service entries
	NWLink transport entries
	Redirector (Rdr) service entries
	Remote Access Service (RAS) entries
	Replicator service entries
	Schedule service entries
	Server service entries
	TCP/IP transport entries
	UPS service entries
	Workstation service entries

Alerter Service Entries

Entries for this service appear under the following subkey:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Alerter

 \Parameters

AlertNames **REG_MULTI_SZ**

Range: *List*

Default: NULL







Specifies the list of users to whom administrator alerts are sent. This value can be set using the Server Manager.

AppleTalk and MacFile Service Entries for SFM

Services For Macintosh (SFM) does not appear in the Registry until you install SFM using the Network icon in Control Panel (from the Windows NT Advanced Server or Resource Guide disks). After installation, the SFM value entries appear under several Services subkeys: AppleTalk, MacFile, MacPrint, and MacSrv. You should let the system maintain entries in the MacPrint or MacSrv service. However, the AppleTalk and MacFile services contain definable parameters described in this section.

You should use the Network icon in Control Panel to configure SFM, and use File Manager to administer file services, Server Manager to administer server services, and Print Manager to administer print services for SFM.

SFM is included with Windows NT Advanced Server and on the Windows NT Resource Guide disk.

	<u>AppleTalk Entries for SFM</u>
	<u>AppleTalk Parameter Entries</u>
	<u>Adapter Card Entries for AppleTalk</u>
	<u>MacFile Entries for SFM</u>
	<u>MacFile Parameters Entries</u>
	<u>Type_Creators Entries for MacFile</u>

AppleTalk Entries for SFM

The values for the AppleTalk service are found in this Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM  
    \CurrentControlSet  
        \Services  
            \AppleTalk
```

Values for the AppleTalk Parameters and Adapters subkeys are described here. The settings in the Linkage subkey and the TCP/IP settings in the Winsock subkey for AppleTalk are maintained by the system and should not be changed by the user. For changes to these values to take effect, you must restart the computer.

The following are the defaults for the basic service entries:

DependOnGroup=NDIS

DependOnService=Lance

DisplayName=AppleTalk Transport

ErrorControl=0x1

ImagePath=\SystemRoot\system32\drivers\sfmataalk.sys

Start=0x2

Type=0x1

AppleTalk Parameter Entries

This is the Registry path for the AppleTalk parameters:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \AppleTalk
        \Parameters
```

These values set port, zone, and router information.

DefaultPort **REG_SZ**

Range: *Adapter name*

Default: the first adapter found.

Specifies the network on which the SFM service names are registered. If the AppleTalk protocol is not routing, only Macintosh workstations connected to this network can see the file and print services.

During initial installation, the default port is set automatically to the first Ethernet adapter found, or to the first Token Ring adapter, or to a LocalTalk adapter (in that order).

DesiredZone **REG_SZ**

Range: *Zone name*

Default: There is no default.

Specifies the zone in which the SFM service is present. If this value is not set, SFM is present in the default zone for that network.

EnableRouter **REG_DWORD**

Range: *0 or 1*

Default: 0

Tells the AppleTalk protocol whether routing needs to be started on this computer. If routing is started, Macintosh workstations connected to any of the networks that this computer is on should be able to use the file and print servers for Macintosh.

Important: This value is critical. Unless absolutely essential, do not set this value to 1.

Adapter Card Entries for AppleTalk

The entries for AppleTalk that are specific to network adapter cards are found under the following Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Services
            \AppleTalk
                \Adapters
                    \adapter_name
```

There is one subkey for each adapter that is AppleTalk-compatible on the computer. These entries are found in each *Adapter_Name* subkey. For changes to these values to take effect, you must restart the computer.

AarpRetries **REG_DWORD**

Range: *Number*

Default: 0xa

Specifies the maximum number of AppleTalk address-resolution protocol packets to be sent by the AppleTalk protocol.

DdpCheckSums **REG_DWORD**

Range: *0 or 1*

Default: 0

Tells the AppleTalk protocol whether to compute checksums in the DDP layer. If this entry is 1, the AppleTalk protocol uses sums in the DDP layer.

DefaultZone **REG_SZ**

Range: *Zone name*

Default: There is no default.

Contains the default zone for this network if this adapter is seeding the network. If the adapter is seeding the network, the default zone is chosen when you configure SFM using the Network icon in Control Panel.

NetworkRangeLowerEnd **REG_DWORD**

Range: *1 to 65279*

Default: There is no default.

Specifies the lower network number of the network range if this adapter is seeding the network. If the adapter is seeding the network, this number is set by using the Network icon in Control Panel to configure SFM.

NetworkRangeUpperEnd **REG_DWORD**

Range: *1 to 65279*

Default: There is no default.

Specifies the upper network number of the network range for this network if this adapter is seeding the network. If the adapter is seeding the network, this number is set by using the Network icon in Control Panel to configure SFM.

PortName **REG_SZ**

Range: *AdapterName@ComputerName*

Default: There is no default.

Specifies the name used to identify the AppleTalk protocol running on a particular adapter on a

computer.

SeedingNetwork REG_DWORD

Range: *0 or 1*

Default: 0

Used by the AppleTalk protocol during startup. If this value is 0, this adapter is not seeding the network and the AppleTalk protocol ignores any seeding information for the adapter, if specified. If this value is 1, the AppleTalk protocol reads all seeding information and seeds the network, if valid.

ZoneList REG_MULTI_SZ

Range: *List of zones*

Default: There is no default.

This value is relevant only when the adapter is seeding the network. The network is seeded with this list of zones by the AppleTalk protocol. Changes can be made by using the Network icon in Control Panel to configure SFM.

MacFile Entries for SFM

The MacFile subkey contains the main entries for the AppleTalk File Protocol (AFP) server. All configuration information for the file server is in the following subkey:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \MacFile
```

The MacFile\Parameters subkey includes Type_Creators, Icons, and Extensions subkeys. You should let the system maintain entries in the Icons or Extensions subkeys. This section describes value entries for the Parameters and Parameters\Type_Creator subkeys.

The basic entries for the MacFile services are the following:

DependOnGroup=

DependOnService=AppleTalk LanManWorkStation NetLogon

DisplayName=File Server for Macintosh

ErrorControl=0x1

ImagePath=%SystemRoot%\system32\sfmsvc.exe

ObjectName=LocalSystem

Start=0x2

Type=0x20

MacFile Parameters Entries

The Registry path for MacFile parameters is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \MacFile
        \Parameters
```

The following value entries specify server options, which can be set from the Server Manager. All of these entries are added to the Registry when changes to the default values occur. For changes to these values to take effect, you must start File Server for Macintosh using the Devices icon in Control Panel.

LoginMsg **REG_SZ**

Range: *1-198 characters*

Default: There is no default.

Specifies the message you want Macintosh users to see when they log on to the Windows NT Advanced Server network.

MaxSessions **REG_DWORD**

Range: *1 - 0xffffffff (unlimited)*

Default: 0xff 255 (in decimal)

Specifies the maximum number of user sessions that the file server for Macintosh can accommodate.

PagedMemLimit **REG_DWORD**

Range: *0x3e8 - 0x3e800 (1000K to 256000K)*

Default: 0x4e20 (20000K)

Specifies the maximum amount of page memory that the file server for Macintosh uses. Performance of the MacFile service increases with an increase in this value. However, the value should not be set lower than 1000 KB. It is especially important that you are well acquainted with memory issues before changing this resource parameter. You cannot change this value from Server Manager.

NonPagedMemLimit **REG_DWORD**

Range: *0xff - 0x3e80 (256K to 16000K)*

Default: 0xfa0 (4000)

Specifies the maximum amount of RAM that is available to the file server for Macintosh. Increasing this value helps performance of the file server, but decreases performance of other system resources.

ServerName **REG_SZ**

Range: *Server name*

Default: There is no default.

Specifies the name of the server running SFM on a Windows NT Advanced Server network. Use the server's Windows NT Advanced Server name as the default if you need to add this entry.

ServerOptions **REG_DWORD**

Default: bits 1, 2, and 3 set to on.

Specifies server options that are set in Server Manager. If needed for repair purposes, change Bits 1 through 3; do not change any of the other bits. When on, Bit 1 allows guest logons, Bit 2 allows cleartext passwords, and Bit 3 allows Macintosh users to save passwords on their workstations.

Volume **REG_MULTI_SZ**

Range: *MaxUses Properties Password Path*

Each entry specifies information about a Macintosh-accessible volume on the server on a Windows NT Advanced Server network. You should add Macintosh-accessible volumes using File Manager.

MaxUses = Specifies the maximum number of simultaneous workstations that can be connected to the file server. The upper limit is unlimited (0xffffffff). The practical limits are based on the server hardware and network media.

Properties = Specifies security options as a decimal value. When Bit 1 is set to On, the volume is read-only. When Bit 16 is set to On, guests can use this volume. The default is 1000000000000000 (in binary notation) (the volume is read-only; guests can use this volume).

Password = Contains the encrypted password. Do not change this value. If a user forgets a password, you can delete this entry, thus removing a password requirement from the user's account. Then the user can specify a new password at logon.

Path = Specifies the path of the volume's root directory. If a volume has been deleted, the path may still be valid; consequently, you should not delete this value. If volumes are deleted using File Manager, you can delete this value.

See **MacCP** in the NLS\CodePage subkey for information about the Macintosh codepage.

Type_Creators Entries for MacFile

The values in the following Registry subkey list all the Macintosh type-creators that are associated with MS-DOS-style filename extensions:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \MacFile

 \Parameters

 \Type_Creators

Change these values from the File Manager by choosing the Associate command from the MacFile menu. The value entries that appear in the Registry for each type of creator have the following format:

REG_MULTI_SZ

Range: Creator=*Value* Type=*Value* Comment=*Value*

The three values appear for each entry. The value for Creator= must have from 1 through 4 characters. The value for Type= must have from 1 through 4 characters. The value for Comment= must have from 0 through 29 characters.



Default Entries for Type_Creators

Default Entries for Type_Creators

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \MacFile

 \Parameters

 \Type_Creators

- 1** = CREATOR=XCEL TYPE=XLC3 Comment=Excel Chart
- 10** = CREATOR=XCEL TYPE=XLM4 Comment=MS Excel 4.0 Macrosheet
- 11** = CREATOR=XCEL TYPE=XLW4 Comment=MS Excel 4.0 WorkSpace
- 12** = CREATOR=XCEL TYPE=XLA Comment=Excel 4.0 Add-in MacroFile
- 13** = CREATOR=XCEL TYPE=XLT Comment=Excel 4.0 Template File
- 14** = CREATOR=MSWD TYPE=WDBN Comment=MS Word 4.0 Document
- 15** = CREATOR=MSWD TYPE=TEXT Comment=MS Word 4.0 Document
- 16** = CREATOR=MSPJ TYPE=MPP Comment=MS Project 1.x
- 17** = CREATOR=MSPJ TYPE=MPX Comment=Project 1.x Xchange Format
- 18** = CREATOR=MSPJ TYPE=MPP Comment=MS Project 1.x
- 19** = CREATOR=MSPJ TYPE=MPC Comment=MS Project 1.x Calendars
- 2** = CREATOR=XCEL TYPE=XLS3 Comment=Excel Spreadsheet
- 20** = CREATOR=MSPJ TYPE=MPV Comment=MS Project 1.x Views
- 21** = CREATOR=MSPJ TYPE=MPW Comment=MS Project 1.x WorkSpaces
- 22** = CREATOR=ALD3 TYPE=ALB3 Comment=Mac Pagemaker Publication
- 23** = CREATOR=ALD3 TYPE=ALT3 Comment=Mac Pagemaker Template
- 24** = CREATOR=ALD3 TYPE=TIFF Comment=Mac Pagemkr Tiff Graphics
- 25** = CREATOR=MORE TYPE=TEXT Comment=Symantec More File
- 26** = CREATOR=FOX+ TYPE=F+DB Comment=FoxBase Plus
- 27** = CREATOR=ALD2 TYPE=PUBF Comment=Mac PageMaker
- 28** = CREATOR=ARTZ TYPE=EPSF Comment=Adobe Illustrator
- 29** = CREATOR=SIT! TYPE=SIT! Comment=Aladdin Stuffit
- 3** = CREATOR=XCEL TYPE=XLM3 Comment=Excel Macrosheet
- 30** = CREATOR=PPT2 TYPE=SLD2 Comment=MS Power Point 2.0
- 31** = CREATOR=PPT3 TYPE=SLD3 Comment=MS Power Point 3.0
- 32** = CREATOR=L123 TYPE=LWK3 Comment=Lotus 1-2-3
- 33** = CREATOR=LMAN TYPE=DEXE Comment=LMAN Executables
- 4** = CREATOR=XCEL TYPE=XLW3 Comment=Excel Workspace
- 5** = CREATOR=XCEL TYPE=XLA Comment=Excel Add-in MacroFile
- 6** = CREATOR=XCEL TYPE=SLM3 Comment=Excel Template File
- 7** = CREATOR=XCEL TYPE=TEXT Comment=MS Excel 3.0
- 8** = CREATOR=XCEL TYPE=XLC4 Comment=MS Excel 4.0 Chart
- 9** = CREATOR=XCEL TYPE=XLS4 Comment=MS Excel 4.0 Spreadsheet

AppleTalk Winsock Entries

The TCP/IP settings in the Winsock subkey for AppleTalk are defined by the system and should not be changed by the user. The following shows the defaults:

HelperDllName=C:\winnt\system32\sfmwshat.dll

Mapping=

MaxSockAddrLength=0x8

MinSockAddrLength=0x6

See Also

[Windows Sockets for TCP/IP](#)

BootVerification Service Entries

The Services subkey for the BootVerification service does not appear by default in the Registry. You can add this key if you want to verify system startup from a remote location using the BOOTVRFY.EXE program provided with Windows NT. This service can be started from a remote computer. The service tells the Windows NT service controller that it should save the current startup system configuration as the LastKnownGood control set, and then the service terminates itself.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Services
            \BootVerification
```

When you add the BootVerification key using Registry Editor, add the following value entries:

ErrorControl : REG_DWORD : 0x1

ImagePath : REG_EXPAND_SZ : bootvrfy.exe

ObjectName : REG_SZ : LocalSystem

Start : REG_DWORD : 0x3

Type : REG_DWORD : 0x2

You must also specify a value of 1 for the **ReportBootOK** entry under the following Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
    \Windows NT
        \CurrentVersion
            \WinLogon
```

When the value of **ReportBootOK** is set to 0, it disables the automatic (default) startup acceptance, which happens after the first successful logon.

This alternative method of verifying system startup cannot be used in conjunction with BootVerificationProgram.

Browser Service Entries

The parameters that control network bindings for the Browser service are described in [NetRules Subkey Entries](#).

Under the following Registry path, two parameters are found:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

\Browser

\Parameters



DGRcvr Entries for the Browser Driver

CacheHitLimit **REG_DWORD**

Range: 0 to 256

Default: 1

Describes the number of NetServerEnum requests required to qualify that the response to a NetServerEnum request be cached. If the browser receives more than **CacheHitLimit** NetServerEnum requests with a particular set of parameters, it caches the response and returns that value to the client.

CacheResponseSize **REG_DWORD**

Range: 0 to xffffff

Default: 10

Specifies the maximum number of responses kept for each transport. To disable this feature, set this value to 0.

IsDomainMasterBrowser **REG_SZ**

Range: Boolean

For TCP/IP, specifies a workstation within a workgroup which can be included in global LMHOSTS file. When this parameter is set to Yes, it forces the elevation of a workstation's priority for the browser. This helps with WAN browsing.

This value should be set on a few systems for the workgroup, placing mappings for each in the global LMHOSTS file. For example, in a workgroup with 20 members, set this value on three of the computers to earn a better chance to act as master browsers. This facilitates remote browsing ability for workstations in remote domains whose domain master browser has successful mappings for these special workgroup members.

MaintainServerList **REG_SZ**

Range: Boolean or Auto

Default: Auto, if none is present. (This server contacts the Master Browse Server, and the Master Browse Server tells this server whether it should become a browse server.)

If this value is No, this server is not a browse server. If this value is Yes, this server becomes a browse server. It attempts to contact the Master Browse Server to get a current browse list. If it cannot find the Master Browse Server, it forces an election and is, of course, a candidate to become the master.

If **MaintainServerList** is Auto, this server may or may not become a browse server, depending on the results of the Registry exchange with the Master Browse Server.

If **MaintainServerList** is set to Yes, the computer is configured to always be a backup browser.

QueryDriverFrequency **REG_DWORD**

Range: 0 to 900

Default: 30

Indicates the time after which a browser maser will invalidate its NetServerEnum response cache and the frequency that a master browser will query the browser driver to retrieve the list of servers. Increasing this time makes browsing somewhat faster, but browse information will not necessarily be 100 percent accurate to the minute. Lowering this time makes browse response more accurate, but will increase the CPU load on the browse master.

DGRcvr Entries for the Browser Driver

The following Browser driver parameters are found under this Registry path for the Datagram Receiver:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

\DGRcvr

\Parameters

BrowserServerDeletionThreshold REG_DWORD

BrowserDomainDeletionThreshold REG_DWORD

Range: 0 to 0xffffffff

Default: 0xffffffff

If more than **BrowserServerDeletionThreshold** servers (or **BrowserDomainDeletionThreshold**) servers (or domains) are flushed in a 30-second interval, this causes an event to be generated.

FindMasterTimeout REG_DWORD

Range: 0 to 0xffffffff seconds

Default: 0xffffffff

Specifies the maximum number of seconds that FindMaster requests should be allowed to take. If you have a slow LAN, you may want to increase this value, but only if directed by Microsoft Product Support Services.

GetBrowserListThreshold REG_DWORD

Range: *Number*

Default: 0xffffffff (That is, never log events.)

Represents the threshold that the Browser uses before logging an error indicating that too many of these requests have been "missed. If more requests than the value of **GetBrowserServerList** are missed in an hour, the Browser logs an event indicating that this has happened.

MailslotDatagramThreshold REG_DWORD

Range: *Number*

Default: 0xffffffff (That is, never log events.)

Represents the threshold that the Browser uses before logging an error indicating that too many of these requests have been "missed. If more mailslots than the value of **MailslotDatagramThreshold** are missed in an hour, the Browser logs an event indicating that this has happened.

DiskPerf Service Entries

The DiskPerf subkey entries determines whether disk performance statistics are maintained by the system. If the **Start** value is 0 (boot), then statistics are counted and are reported by Performance Monitor and similar tools. Collecting disk performance statistics can take up to 1.5 percent of the disk throughput on a system with a slow processor (such as an 20 MHz 80386 computer) but should have negligible impact on a system with a faster processor (such as a 33 MHz i486 and above).

Turn DiskPerf on or off only by using the Diskperf utility; for example, type **diskperf -y** at the command prompt.

There are no parameters that users can set. The following are the default values for the standard entries:

ErrorControl = 0x1 (Normal)

Group = Filter

Start = 0x4 (disabled)

Type = 0x1 (Kernel driver)

DLC System Driver Entries

The DLC subkey does not appear unless this service is installed. In Windows NT, the Data Link Control (DLC) protocol only needs to be installed on computers that access IBM mainframes (usually with 3270 applications) or on print servers that print directly to Hewlett-Packard printers. Network printers such as the HP III si use the DLC protocol, for example, because the frames received are easy to take apart.

The DLC driver depends on an having an NDIS group service available and is bound to the network adapter card through the NDIS device driver. Each adapter that supports the DLC protocol has a subkey under the DLC\Parameters subkey. With Registry Editor, you can modify the following parameters for the DLC system driver. The path for these parameters is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \DLC
        \Parameters
          \adapter name
```

The following **TxTick** parameters are multipliers for the T1, T2, and Ti values, which represent time periods in milliseconds that are used when a station or SAP is opened. If the time period value is between 1 and 5, the time delay is computed as follows:

$$(\text{time period}) * \mathbf{TxTickOne} * 40 \text{ milliseconds}$$

If the time period value is between 6 and 10, the time delay is computed as follows:

$$(\text{time period} - 5) * \mathbf{TxTickTwo} * 40 \text{ milliseconds}$$

When computing the short-tick values (**TxTickOne**) and the long-tick values (**TxTickTwo**), the resulting values for T1, T2, and Ti should generally follow this rule: $T2 < T1 < Ti$.

T1Tick{One|Two} REG_DWORD

Range: 1 to 255 milliseconds

Default: **T1TickOne** = 5; **T1TickTwo** = 25

Specifies the delay before retransmitting an I frame if not acknowledged.

T2Tick_{One|Two} REG_DWORD

Range: 1 to 255 milliseconds

Default: **T2TickOne** = 1; **T2TickTwo** = 10

ies the delay before acknowledging frames in the receive window if the receive window has not been filled.

TiTick{One|Two} REG_DWORD

Range: 1 to 255 milliseconds

Default: **TiTickOne** = 25; **TiTickTwo** = 125

Specifies the delay before testing an inactive station to determine if it is still active.

Swap REG_DWORD

Range: 0 or 1

Default: 1

Used when talking over Ethernet to support certain Token Ring-to-Ethernet bridges for swapping of the Destination Address when using DLC over a Token Ring network. If this value is 0, the adapter addresses presented at the API interface are not bit-flipped before being put online.

UseDixOverEthernet REG_DWORD

Range: 0 or 1

Default: 0

Specifies the default mode for the connection-oriented and connectionless 802.2 LLC (Logical Link Control) frames sent on Ethernet. If this value is 1, the DLC driver users DIX encoding in the frames to be transmitted instead of the 802.3 Ethernet format.

Note: Additional parameters that were included in the Microsoft LAN Manager COMTOKR utility are not defined for DLC, because Windows NT does not have the same memory limitations as MS-DOS.

EventLog Service Subkeys

The Services subkey for EventLog contains at least three subkeys for the three types of logs-- Application, Security, and System. These *Logfile* subkeys contain subkeys that define the location of the related event message file and the supported types of events, as follows:



The Application key contains subkeys for installed applications and services that write to the Application event log.



The Security key contains subkeys for each of the security subsystem components.



The System key contains subkeys for device drivers.

Each of the three *Logfile* subkeys for the EventLog service can contain the value entries described in this section. The Registry path for these entries is the following, where *logfile* is System, Application, or Security.

HKEY_CURRENT_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Eventlog

 \logfile

These entries are described for informational purposes only. This information is usually maintained by Event Viewer. New keys under the Application key can only be added in meaningful ways by using the Win32 Registry APIs.



EventLog Service Entries

EventLog Service Entries

Registry path:

HKEY_CURRENT_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Eventlog
 \Logfile

File **REG_SZ**

Range: *Path and filename*

Default: \SystemRoot\system32\config\filename

Specifies the fully qualified path name of the file for this log. This value can be set in Event Viewer.

MaxSize **REG_DWORD**

Range: *Number in kilobytes*

Default: 512

Specifies the maximum size of the log file. This value can be set using the Event Viewer.

Retention **REG_DWORD**

Range: *Number of seconds*

Default: 604800 (7 days)

Specifies that records that are newer than this value will not be overwritten. This is what causes a log full event. This value can be set using the Event Viewer.

Sources **REG_MULTI_SZ**

Range: *Names of source applications*

Default: None. This value is dynamically maintained by the EventLog service.

Specifies the applications, services, or groups of applications that write events to this log. Each source is a subkey of the *Logfile* key.

The *Source* subkeys under a *Logfile* key are created by the applications that write events in the related event log. These subkeys contain information specific to the source of the event under the following types of value entries.

EventMessageFile **REG_EXPAND_SZ**

Range: *Filename*

Specifies the path and filename for the event identifier message file.

CategoryMessageFile **REG_EXPAND_SZ**

Range: *Filename*

Specifies the path and filename for the category message file. The category and event identifier message strings may be in the same file.

CategoryCount **REG_DWORD**

Range: *Number*

Specifies the number of categories supported.

TypesSupported **REG_DWORD**

Range: *Number*

Specifies a bitmask of supported types.

NBF (NetBEUI) Transport Entries

The startup parameters for the NetBEUI (NBF) transport are found under the following subkey:

HKEY_LOCAL_MACHINE\SYSTEM

\Services






\NBF

\Parameters

Note: The parameters that control network bindings for this service are described in NetRules Subkey Entries. You should set the Export and Bind parameters by using the Network application in Control Panel.

The Initxxx entries for NBF define the initial allocation and the size of free memory for items. The Maxxxx entries define the upper limits. Within these ranges, the system autotunes performance. By default, the NBF service uses all the resources necessary to handle client requests, and when it is not actively working, it doesn't use many resources. Set Initxxx values to control initial allocation, which can make the system a little faster when you know a server will be busy. Set the Maxxxx values to control limits when you don't want the server to be too busy or to use too much memory for networking.

With Registry Editor, you can modify the following startup parameters for the NBF transport:

	<u>Defaultxxx Entries for NBF</u>
	<u>Initxxx Entries for NBF</u>
	<u>LLCxxx Entries for NBF</u>
	<u>MAXxxx Entries for NBF</u>
	<u>Other Entries for NBF</u>

Defaultxxx Entries for NBF

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

\Services

\NBF

\Parameters

DefaultT1Timeout **REG_DWORD**

Range: *100-nanosecond units*

Default: 6000000 (600 milliseconds)

Specifies the initial value for the T1 timeout. T1 controls the time that NBF waits for a response after sending a logical link control (LLC) poll packet before resending it. Adjust this parameter only if NBF will be connecting over slow networks or to slow remote computers (although NBF does adapt).

DefaultT2Timeout **REG_DWORD**

Range: *100-nanosecond units*

Default: 1500000 (150 milliseconds)

Specifies the initial value for the T2 timeout. T2 controls the time that NBF can wait after receiving an LLC poll packet before responding. It must be much less than T1; one-half or less is a good general rule. Adjust this parameter only if NBF will be connecting over slow networks or to slow remote computers.

DefaultTiTimeout **REG_DWORD**

Range: *100-nanosecond units*

Default: 300000000 (30 seconds)

Specifies the initial value for the Ti timeout. Ti is the inactivity timer. When it expires, NBF sends an LLC poll packet to ensure that the link is still active. Adjust this parameter only if NBF is connecting over networks with unusual reliability characteristics, or over slow networks or to slow computers.

Initxxx Entries for NBF

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \Services
 \NBF
 \Parameters

InitAddresses **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies the number of initial addresses to allocate within any memory limits that might imposed on NBF. Addresses correspond to NetBIOS names. An address is for the actual name, and an address file is for a TDI (Transport Driver Interface) client using that name; so usually you have the same number, but if two users open the same address, that is two address files but only one address.

Set this parameter if you know that a large number of addresses are needed. Otherwise, the system automatically allocates space for addresses as needed.

InitAddressFiles **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies the number of initial address files to allocate within any memory limits that might imposed on NBF. Set this parameter if you know that a large number of address files are needed. Otherwise, the system automatically allocates space for address files as needed.

InitConnections **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 1

Specifies the number of initial connections (NetBIOS sessions) to allocate within any memory limits that might imposed on NBF. Set this parameter if you know that a large number of connections are needed. Otherwise, the system automatically allocates space for connections as needed.

InitLinks **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 2

Specifies the number of initial LLC links to allocate within any memory limits that might imposed on NBF. Typically, you have one connection per LLC link to another network adapter card, because the redirector puts all links to a computer into one connection. However, you may have more if two computers are communicating with each other or if a NetBIOS application is running. Set this parameter if you know that a large number of links are needed. Otherwise, the system automatically allocates space for links as needed.

InitReceiveBuffers **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 5

Specifies the number of initial receive buffers to allocate. Receive buffers are used by NBF when it calls NDIS TransferData for received datagrams. Usually, this value is allocated as needed, but you can use this parameter to preallocate memory if you know a large number of datagram frames will be received.

InitReceivePackets **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 10

Specifies the number of initial receive packets to allocate. Receive packets are used by NBF when it calls NDIS TransferData for received data. Usually, this value is allocated as needed, but you can use this parameter to preallocate memory if you know a large number of UI frames will be received.

InitRequests REG_DWORD

Range: *1 or higher; 0 = no limit*

Default: 5

Specifies the number of initial requests to allocate within any memory limits that might imposed on NBF. Requests are used for in-progress connect requests, remote adapter status requests, find name requests, and so on. Set this parameter if you know that a large number of requests are needed. Otherwise, the system automatically allocates space for requests as needed.

InitSendPackets REG_DWORD

Range: *1 or higher; 0 = no limit*

Default: 30

Specifies the number of initial send packets to allocate. Send packets are used by NBF whenever it sends connection-oriented data on behalf of a client. Usually, this value is allocated as needed, but you can use this parameter to preallocate memory if you know a large number of data frames are needed or if you see a lot of "send packets exhausted messages when using Performance Monitor.

InitUIFrames REG_DWORD

Range: *1 or higher; 0 = no limit*

Default: 5

Specifies the number of initial UI frames to allocate. UI frames are used by NBF to establish connections and for connectionless services such as datagrams. Usually, this value is allocated as needed, but you can use this parameter to preallocate memory if you know a large number of UI frames are needed.

LLCxxx Entries for NBF

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

\Services

\NBF

\Parameters

LLCMaxWindowSize **REG_DWORD**

Range: *Number of frames*

Default: 10

Specifies the number of LLC lframes that NBF can send before polling and waiting for a response from the remote. Adjust this parameter only if NBF is communicating over a network whose reliability often changes suddenly.

LLCRetries **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 8

Specifies the number of times that NBF will retry polling a remote workstation after receiving a T1 timeout. After this many retries, NBF closes the link. Adjust this parameter only if NBF is connecting over networks with unusual reliability characteristics.

MAXxxx Entries for NBF

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \Services
 \NBF
 \Parameters

MaxAddresses **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies the maximum number of addresses that NBF allocates within any memory limits that might imposed on NBF. Addresses are NetBIOS names that are registered on the network by NBF. An address is for the actual name, and an address file is for a TDI client using that name.

Use this optional parameter to fine-tune use of NBF memory. Typically this parameter is used to control address resources with an unlimited NBF.

MaxAddressFiles **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies maximum number of address files that NBF allocates within any memory limits that might imposed on NBF. Each address file corresponds to a client opening an address.

Use this optional parameter to fine-tune use of NBF memory. Typically this parameter is used to control address files with an unlimited NBF.

MaxConnections **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies the maximum number of connections that NBF allocates within any memory limits that might imposed on NBF. Connections are established between NBF clients and similar entities on remote computers.

Use this optional parameter to fine-tune use of NBF memory. Typically this parameter is used to control connection resources with an unlimited NBF.

MaximumIncomingFrames **REG_DWORD**

Range: *1 or higher; 0 = off*

Default: 2

Used in some cases to control how many incoming frames NBF will receive before it sends an acknowledgment to a remote machine. In general, NBF automatically senses when to send acknowledgments, however when communicating with some Microsoft Lan Manager or Lan Server remote computers configured with a very low value for **maxout**, this parameter can be set to an equal or lower value to improve network performance. (This parameter corresponds roughly to the Microsoft LAN Manager **maxin** parameter.) A value of 0 turns off this hint, causing NBF to revert to usual behavior. For communication with most all remotes, this parameter isn't used.

MaxLinks **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies the maximum number of links that NBF allocates within any memory limits that might imposed on NBF. Links are established for every remote adapter to which NBF communicates.

Use this optional parameter to fine-tune use of NBF memory. Typically this parameter is used to

control link resources with an unlimited NBF.

MaxRequests **REG_DWORD**

Range: *1 or higher; 0 = no limit*

Default: 0 (no limit)

Specifies the maximum number of requests that NBF allocates within any memory limits that might imposed on NBF. Requests are used by NBF to control send, receive, connect, and listen operations.

Use this optional parameter to fine-tune use of NBF memory. Typically this parameter is used to control request resources with an unlimited NBF.

Other NBF Entries

Registry path:
HKEY_LOCAL_MACHINE\SYSTEM
 \Services
 \NBF
 \Parameters

AddNameQueryRetries **REG_DWORD**

Range: *Number*

Default: 3

Specifies the number of times that NBF will retry sending ADD_NAME_QUERY and ADD_GROUP_NAME_QUERY frames. Adjust this parameter only if NBF is registering addresses on a network that drops many packets.

AddNameQueryTimeout **REG_DWORD**

Range: *100-nanosecond units*

Default: 5000000

Specifies the time-out between NBF sending successive ADD_NAME_QUERY and ADD_GROUP_NAME_QUERY frames. Adjust this parameter only if NBF is registering addresses on a network with slow computers or over a slow network.

GeneralRetries **REG_DWORD**

Range: *Number*

Default: 3

Specifies the number of times that NBF will retry sending STATUS_QUERY and FIND_NAME frames. Adjust this parameter only if NBF is operating on a network that drops many packets.

GeneralTimeout **REG_DWORD**

Range: *100-nanosecond units*

Default: 5000000

Specifies the time-out between NBF sending successive STATUS_QUERY and FIND_NAME requests. Adjust this parameter only if NBF is operating on a network with slow computers or over a slow network.

NameQueryRetries **REG_DWORD**

Range: *Number*

Default: 3

Specifies the number of times that NBF will retry sending NAME_QUERY frames. Adjust this parameter only if NBF is connecting to computers over a network that drops many packets.

NameQueryTimeout **REG_DWORD**

Range: *100-nanosecond units*

Default: 5000000

Specifies the time-out between NBF sending successive NAME_QUERY frames. Adjust this parameter only if NBF is connecting to slow computers or over a slow network.

QueryWithoutSourceRouting **REG_DWORD**

Range: *0 or 1*

Default: 0 (false)

When you are using NBF over a Token Ring driver, this parameter instructs NBF to send half the

queries without including source routing information when connecting to a remote computer. This supports bridging hardware that cannot forward frames that contain source routing information.

UseDixOverEthernet REG_DWORD

Range: *0 or 1*

Default: 0 (false)

Specifies whether NBF should use DIX encoding when bound to an Ethernet MAC. When using DIX encoding is enabled, NBF cannot talk to computers that use the standard IEEE 802.3 encoding.

WanNameQueryRetries REG_DWORD

Range: *Number*

Default: 5

Specifies the number of times that NBF will retry sending NAME_QUERY frames when connecting with RAS. Adjust this parameter only if NBF is connecting to computers over a network that drops many packets.

NetLogon Service Entries

The Registry path for the parameters for the NetLogon service is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Netlogon
        \Parameters
```

Note: The NetLogon share name should also be in the path for logon scripts.

PulseInterval **REG_DWORD**

Range: *60 to 3600 seconds*

Default: 300

Specifies how long a domain controller waits before sending each update notice to Windows NT Advanced Servers.

When this value is not specified in the Registry, NetLogon determines optimal values depending on the domain controller's load.

Randomize **REG_DWORD**

Range: *15 to 120 seconds*

Default: 30

Specifies the amount of time a Windows NT Advanced Server domain controller uses to stagger requests sent to the domain controller. This value is used by every server in the domain. When the domain controller sends update message to the servers, it includes the **Randomize** value in the message. The servers receiving that message will wait a maximum of that many seconds before responding to that message.

When this value is not specified in the Registry, NetLogon determines optimal values depending on the domain controller's load.

Scripts **REG_SZ**

Range: *Pathname*

Default: NULL

Specifies the fully qualified path name to where logon scripts reside. This value can be set using the Services icon in Control Panel or the Server Manager.

Update **REG_SZ**

Range: *Yes or No*

Default: No

When this value is set to Yes, NetLogon fully synchronizes the database each time it starts.

NetworkProvider Service Entries

If more than one network is present under Windows NT, each network has a Services subkey that include a value for Group of NetworkProvider plus its own subkey named NetworkProvider.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Service or Driver Key Name
        \NetworkProvider
```

The following entry values should appear under the NetworkProvider subkey for each network the computer can use:

The NetworkProvider subkey under CurrentControlSet\Control provides a list of the available network providers.

ProviderName **REG_SZ**

Range: *Name*

Default: Defined by network vendor

Specifies the name of the Provider, which is displayed as the network name in Browse dialog boxes. This name is defined by the network vendor, and is usually some variation of the product name.

ProviderPath **REG_SZ**

Range: *DLL pathname*

Default: Defined by network vendor

Specifies the full path of the DLL that implements the network provider.

NWLink Transport Entries (IPX/SPX)

NWLink is an implementation of the IPX/SPX protocols popular in NetWare networks. In addition, the module NWNBLink provides support for the Novell implementation of the NetBIOS protocol. With the Registry Editor, you can modify the following:



NWNBLink Entries for Microsoft Extensions to Novell NetBIOS

These are parameters for the NetBIOS component of NWNBLink, including modification of parameters supporting the Microsoft extensions to Novell NetBIOS



NWNBLink Entries for Novell NetBIOS or Microsoft Extensions

These are parameters for the IPX/SPX component of NWLink



NWLink Entries for IPX/SPX for the Network Adapter Card



NWLink Entries for Global IPX/SPX Parameters

The last two sections contain parameters that affect the use of NWLink in a Token Ring network

Caution: All entries have reasonable defaults that usually should not need to be modified. Be careful when modifying an entry, because any change can easily affect the performance of a conversation between the sender and receiver.

The NWLink keys do not appear in the Registry unless this service is installed using the Network icon in Control Panel. After the service is installed, not all entries appear by default in the Registry. If the entry is not there, the default value for that entry is used.

NWNBLink Entries for Microsoft Extensions to Novell NetBIOS

The Microsoft Extensions to Novell NetBIOS are included to enhance the performance of the traditional Novell NetBIOS protocol. NWNBLink can detect automatically whether it is talking to a Novell NetBIOS implementation that does not understand these extensions. In such a case, NWNBLink will fall back to the standard Novell NetBIOS protocol currently used in NetWare networks.

However, significant performance gains can be realized if the extensions are used (for example, if the NetBIOS conversation occurs between two Windows NT computers).

The Registry path for these value entries is the following:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \NWNBLink

 \Parameters



ACKxxx Entries for NBNWLink



Other NWNBLink Entries

ACKxxx Entries for NBNWLink

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \NWNBLink
 \Parameters

AckDelayTime **REG_DWORD**

Range: 50 to 65535 *milliseconds*

Default: 250 (no entry = default)

Determines the value of the delayed acknowledgment timer.

AckWindow **REG_DWORD**

Range: 0 to 65535 *frames*

Default: 2 (no entry = default)

Specifies the number of frames to receive before sending an acknowledgment. The **AckWindow** entry is used as a clocking mechanism on networks in which the sender is networked on a fast LAN, but the receiver is networked on the other side utilizing a slower link. By automatically forcing acknowledgments, the sender can keep sending frames continually. If both the sender and receiver are located on a fast link, you can set **AckWindow** to 0 to turn off sending an acknowledgment to the sender. Alternatively, NWNBLink can be set to dynamically determine whether to use the **AckWindow** parameter based on the setting of **AckWindowThreshold**. Related parameter:

AckWindowThreshold.

AckWindowThreshold **REG_DWORD**

Range: 0 to 65535 *milliseconds*

Default: 500 (no entry = default)

Specifies the threshold value for the round-trip time that defines when **AckWindow** will be ignored. The round trip time is an estimate of how long it takes for a frame to be sent and received from a workstation. NWNBLink determines this estimate and uses it as a basis for determining whether it is necessary to send automatic acknowledgments. If **AckWindowThreshold** is set to 0, NWNBLink relies on the **AckWindow** entry. Related parameters: **AckWindow**.

Other NWNBLink Entries

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \NWNBLink
 \Parameters

EnablePiggyBackAck **REG_DWORD**

Range: 0 or 1

Default: 1 (true--enable piggybacking acknowledgments; no entry = default)

Allows the receiver to piggyback acknowledgments. Piggybacking acknowledgments can occur when the receiver has detected the end of a NetBIOS message. When the sender and receiver are not participating in two-way NetBIOS traffic, you should set **EnablePiggyBackAck** to 0. An example of one-way traffic is a stock update application, where a server constantly sends NetBIOS messages to clients but the client does not need to respond.

If **EnablePiggyBackAck** is set to 1 but there is no back traffic, NWNBLink waits the number of milliseconds determined by **AckDelayTime** before sending the acknowledgment, and then it turns off support for piggybacking acknowledgments. If the workstation at some point starts sending as well as receiving data, NWNBLink turns support back on for piggybacking acknowledgments. Related parameter: **AckDelayTime**.

Extensions **REG_DWORD**

Range: 0 or 1

Default: 1 (true; no entry = default)

Specifies whether to use NWNBLink extensions discussed in this section.

RcvWindowMax **REG_DWORD**

Range: 1 to 49152 frames

Default: 4 (no entry = default)

Specifies the maximum number of frames the receiver can receive at one time. The value specified by **RcvWindowMax** is sent to the sender during session initialization to give the sender an upper bound on the number of frames that can be sent at one time. Related parameters: **AckDelayTime**, **AckWindow**, **AckWindowThreshold**, **EnablePiggyBackAck**, and **RcvWindowMax**.

NWNBLink Entries for Novell NetBIOS or Microsoft Extensions

The Registry path for these value entries is the following:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \NWNBLink
 \Parameters

BroadcastCount **REG_DWORD**

Range: 1 to 65535

Default: 3 (no entry = default)

Specifies the number of times to send a broadcast. If **Internet** is set to 1, the **BroadcastCount** is doubled. Related parameter: **BroadcastTimeout**.

BroadcastTimeout **REG_DWORD**

Range: 1 to 65535 *half-seconds*

Default: 1 (no entry = default)

Specifies the time between sending find-name requests. This value is not affected if **Internet** is set to 1. Related parameter: **BroadcastCount**.

ConnectionCount **REG_DWORD**

Range: 1 to 65535

Default: 5 (no entry = default)

Specifies the number of times to send a connection probe. A connection probe is sent by the initiator of a session if a connection could not be made to the remote computer. If **Internet** is set to 1, the **ConnectionCount** is doubled. Related parameter: **ConnectionTimeout**.

ConnectionTimeout **REG_DWORD**

Range: 1 to 65535 *half-seconds*

Default: 2 (no entry = default)

Specifies the time between sending connection probes when initiating a session.

InitialRetransmissionTime **REG_DWORD**

Range: 1 to 65535 *milliseconds*

Default: 500 (no entry = default)

Specifies the initial value for the retransmission time. Related parameter: **RetransmitMax**.

Internet **REG_DWORD**

Range: 0 or 1

Default: 1 (true; no entry = default)

Specifies whether to change the packet type from 0x04 to 0x14 (Novell WAN broadcast).

KeepAliveCount **REG_DWORD**

Range: 1 to 65535

Default: 8 (no entry = default)

Specifies the number of times to send a session-alive frame before timing out if there is no response. Related parameter: **KeepAliveTimeout**.

KeepAliveTimeout **REG_DWORD**

Range: 1 to 65535 *half-seconds*

Default: 60 (no entry = default)

Specifies the time between sending session-alive frames. Related parameter: **KeepAliveCount**.

RetransmitMax **REG_DWORD**

Range: 1 to 65535

Default: 8 (no entry = default)

Specifies the maximum number of times the sender should retransmit before assuming that something is wrong with the link. Related parameter: **InitialRetransmissionTime**.

NWLink Entries for IPX/SPX for the Network Adapter Card

These parameters are specific for each binding of NWLink to a network adapter card. The Registry path for these value entries is the following:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \NWLinkIPX

 \NetConfig

 \Driver01



Sourcexxx Entries for NWLinkIPX for NetCards



Other Entries for NWLinkIPX for NetCards

Other Entries for NWLinkIPX for NetCards

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \NWLinkIPX
        \NetConfig
          \Driver01
```

AdapterName **REG_DWORD**

Range: *Name*

Specifies the name of the adapter that NWLink will use. This parameter is set when you choose a network adapter card to bind NWLink using the Network icon in Control Panel. In this release, only one card is supported.

BindSap **REG_DWORD**

Range: *Type field*

Default: 8137 (in hex)

Specifies the Ethertype if the frame format is Ethernet II. The Ethertype field is only relevant if Ethernet II frames are to be sent or received. You can choose the frame type using the Network icon in Control Panel. Related parameter: **PktType**.

EnableFuncaddr **REG_DWORD**

Range: *Boolean*

Default: 1 (true)

When set to 1, this parameter specifies that the IPX functional address will be enabled if this card is a Token Ring card. If this value is 0, the IPX functional address will not be added. The IPX functional address is C00000800000 (hex). Novell has been phasing out use of this, but it is still in use in some places. It is up to the application to take advantage of the IPX functional address. In most instances, the broadcast address (FFFFFFFFFFFF hex) is used instead.

MaxPktSize **REG_DWORD**

Range: *0 to 65535*

Default: 0

Specifies the maximum frame size the network adapter card should be allowed to transmit. If this number is 0, NWLink will get this information from the card driver. This parameter allows the administrator to make the maximum transmit size for a card smaller than the card driver allows. A scenario in which you might want to change this entry is in an environment in which the network adapter card on one side of a conversation is on a link that has a larger frame size than the link on the other side of a conversation—for example, if the sending station is linked to a 16 Mbps Token Ring and the receiving station is linked to an Ethernet network.

NetworkNumber **REG_DWORD**

Range: *Number*

Default: 0

Specifies the network number (in hex) to be used for this adapter. If this number is 0, the NWLink will get the network number from the network as it is running. This parameter is set using the Network icon in Control Panel. IPX network numbers are 4 bytes (8 hex characters) long. An example of an IPX network number is AABBDFFF. You should not have to enter a specific value because NWLink will determine it for you. Make sure to get the network number for your IPX subnet from the network administrator if you want to enter a specific number.

PktType **REG_DWORD**

Range: 0, 1, 2, 3, or 4

Default: 1 (802.3)

NWLink supports Ethernet, Token Ring, FDDI, and Arcnet topologies. The **PktType** parameter specifies the packet form to use. The valid values (and their related packet forms) are the following:

0 = Ethernet_II

1 = Ethernet_802.3

2 = 802.2

3 = SNAP

4 = Arcnet

If the adapter is an Ethernet adapter, choose between values 0 through 3. If the adapter is either a Token Ring or FDDI adapter, choose between s 2 and 3. If you are using an Arcnet adapter, choose value 4. If the adapter is a Token Ring or FDDI adapter, values 0 and 1 will work the same as value 2.

Related parameter: **BindSap**.

Sourcexxx Entries for NWLinkIPX for NetCards

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \NWLinkIPX
        \NetConfig
          \Driver01
```

SourceRouteBcast **REG_DWORD**

Range: *0 or 1*

Default: 0

Specifies the source route to be used when transmitting a packet to the broadcast MAC (Media Access Control) address (FFFFFFFFFFFF hex). If this value is 0, the packet will be transmitted to the single-route broadcast (0xC2, 0x70). If the value is 1, the packet will be transmitted to the all-routes broadcast (0x82, 0x70). Related parameters: **SourceRouteDef**, **SourceRouting**, and **SourceRouteMCast**.

SourceRouteDef **REG_DWORD**

Range: *0 or 1*

Default: 0

Specifies the source route to be used when transmitting a package to a unique MAC address that is not in the source routing table. If the MAC address is in the source routing table, the route in the table will be used. If this value is 0, the packet will be transmitted to the single-route broadcast (0xC2, 0x70). If the value is not 0, the packet will be transmitted to the all-routes broadcast (0x82, 0x70). Related parameters: **SourceRouteBcast**, **SourceRouting**, and **SourceRouteMCast**.

SourceRouteMcast **REG_DWORD**

Range: *Boolean*

Default: 0

Specifies the source route to be used when transmitting a packet to a multicast MAC address (C000xxxxxxx). If this value is 0, the packet will be transmitted to the single-route broadcast (0xC2, 0x70). If the value is not 0, the packet will be transmitted to the all-routes broadcast (0x82, 0x70). Related parameters: **SourceRouteBcast**, **SourceRouteDef**, and **SourceRouting**.

SourceRouting **REG_DWORD**

Range: *Boolean*

Default: 0 (false--do not use source routing)

Specifies whether to use source routing. This parameter is only used if the adapter is a Token Ring adapter. If there are no source routing bridges on the Token Ring, disable this entry to disable all of the source routing logic. Related parameters: **SourceRouteBcast**, **SourceRouteDef**, and **SourceRouteMCast**.

NWLink Entries for Global IPX/SPX Parameters

The following parameters are global for the entire transport. The Registry path for these value entries is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \NWLinkIPX
        \Parameters
```

ConnectionCount **REG_DWORD**

Range: 1 to 65535

Default: 10

Specifies the number of times the probe will be sent when SPX is trying to connect to a remote node. If no response is received after the probes are sent, an error will occur. Related parameter:

ConnectionTimeout.

ConnectionTimeout **REG_DWORD**

Range: 1 to 65535 *half-seconds*

Default: 2 (1 second)

Specifies the time between connection probes when SPX is trying to connect to a remote node.

Related parameter: **ConnectionCount.**

KeepAliveCount **REG_DWORD**

Range: 1 to 65535

Default: 8

Specifies how many times to send a keep-alive probe before timing out if there is no response.

Related parameter: **KeepAliveTimeout.**

KeepAliveTimeout **REG_DWORD**

Range: 1 to 65535 *half-seconds*

Default: 12 (6 seconds)

Specifies the time that the local side should wait before sending a probe to the remote to verify that the SPX connection is still alive. Related parameter: **KeepAliveCount.**

RipAgeTime **REG_DWORD**

Range: 1 to 65535 *minutes*

Default: 5 minutes

IPX maintains an RIP cache in order to locate computers on a remote network. The **RipAgeTime** entry informs IPX how long to wait before requesting an RIP update for an entry. This timer is reset when an RIP announcement is received for an entry in the RIP cache.

RipCount **REG_DWORD**

Range: 1 to 65535

Default: 5

When the RIP protocol layer is trying to find a route on the network, this parameter specifies how many times to send a request before giving up. Related parameter: **RipTimeout**

RipTimeout **REG_DWORD**

Range: 1 to 65535 *half-seconds*

Default: 1 (1 half-second)

Specifies the timeout between RIP request packets being sent out when the RIP protocol layer is trying to find a route on the network. Related parameter: **RipCount**.

RipUsageTime **REG_DWORD**

Range: *1 to 65535 minutes*

Default: 15 minutes

IPX maintains a RIP cache in order to locate computers on a remote network. The **RipUsageTime** entry informs IPX how many minutes to wait before an entry in the RIP cache will be deleted from the cache. This timer is reset when a packet is sent to the remote computer.

SourceRouteUsageTime **REG_DWORD**

Range: *1 to 65535 minutes*

Default: 10

Specifies the number of minutes an unused entry can remain in the Token Ring source routing cache before it is flushed.

WindowSize **REG_DWORD**

Range: *1 to 10 SPX packets*

Default: 4

Specifies the window to use in the SPX packets. SPX uses the Allocation field of the SPX packet to tell the remote how many receives are available for receiving data. The **WindowSize** entry specifies what value to put in the SPX Allocation field.

Redirector (Rdr) Service Entries

The subkey for the Rdr (redirector) service has the following Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
    \CurrentControlSet
        \Services
            \Rdr
                \Parameters
```

For the search buffer sizes defined in the following entries: if the buffer passed for the search is less than the **LowerSearchThreshold** value, the system requests **LowerSearchThreshold** bytes of data from the server. If the buffer size is between the value of **LowerSearchThreshold** and **UpperSearchBufferSize**, the system uses the buffer size. On a slow link (such as a RAS link), if it will take more than five seconds to retrieve data, the Redirector service uses the user's requested buffer size.

ConnectTimeout **REG_DWORD**

Range: *Number of seconds*

Default: 300 (5 minutes)

Specifies the maximum amount of time the redirector will wait for a connect or disconnect to complete.

LowerSearchBufferSize **REG_DWORD**

Range: *Number of kilobytes*

Default: 16K

Specifies the number of bytes the redirector will use for small searches.

LowerSearchThreshold **REG_DWORD**

Range: *Number of kilobytes*

Default: 16K

Specifies the number of bytes below which the redirector will request a search of **LowerSearchBufferSize**. If the search size is larger than this (but below the **UpperSearchBufferSize**), the redirector will use the **UpperSearchBufferSize**.

StackSize **REG_DWORD**

Range: *Number of kilobytes*

Default: 4

Sets the default IRP stack size for the redirector.

UpperSearchBufferSize **REG_DWORD**

Range: *Number of kilobytes*

Default: 32K

Specifies the number of bytes the redirector will use for large searches.

UseAsyncWriteBehind **REG_DWORD**

Range: *0 or 1*

Default: 1 (true)

Enables the asynchronous-write-behind variation of the write-behind optimization.

UseWriteBehind **REG_DWORD**

Range: *0 or 1*

Default: 1 (true)

Enables the write-behind optimization.






Remote Access Service (RAS) Entries

The RemoteAccess subkey is created in the Registry when you install RAS on a server, using the Network icon in Control Panel. The default values in RemoteAccess and its subkeys work well for all Windows NT operations such as copying files, using network resources, and sending and receiving electronic mail. However, for some systems, you may want to adjust individual parameters to suit your particular performance and security needs.

Initially, there are no value entries in the Registry for the Remote Access key or its subkeys until you add them with new settings. (The only exception is **EnableNetbiosGateway**, the NetBIOS parameter.) Unlisted value entries are set to their default values.

For information on Remote Access configuration files and other parameters, see Appendix B, "Configuration Files, in the *Windows NT Remote Access Service Administrator's Guide*.

The subkeys under the CurrentControlSet\Services\RemoteAccess key include the following entries:

-  Remote Access Parameters Subkey Entries
-  RAS NetBIOSGateway Subkey Entries
-  RAS AsyncMAC Subkey Entries
-  RAS RasHub Subkey Entries
-  RAS RasMan Subkey Entries

Remote Access Parameters Subkey Entries

The Parameters subkey for Remote Access has the following Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \RemoteAccess
        \Parameters
```

For changes to these values to take effect, you must stop and restart the Remote Access service. The functions and settings of these value entries are as follows:

AuthenticateRetries **REG_DWORD**

Range: *0 to 10*

Default: 2

Sets the maximum number of unsuccessful retries allowed if the initial attempt at authentication fails.

AuthenticateTime **REG_DWORD**

Range: *20 to 600 seconds*

Default: 120 seconds

Sets the maximum time limit within which a user must be successfully authenticated. If the client does not initiate the authentication process within this time, the user is disconnected.

CallbackTime **REG_DWORD**

Range: *2 to 12 seconds*

Default: 2 seconds

Sets the time interval that the server waits before calling the client back when the Callback feature has been set. Each client communicates the value of its own callback time when connecting to a Remote Access server. If this value is not communicated (that is, if the client does not communicate a value for the callback time, as with Remote Access 1.0 and 1.1 clients), the value of the **CallbackTime** parameter becomes the default.

EnableAudit **REG_DWORD**

Range: *0 or 1*

Default: 1 (enabled)

Determines whether Remote Access auditing is turned on or off. If this feature is enabled, all audits are recorded in the Security event log, which you can view using Event Viewer.

NetbiosGatewayEnabled **REG_DWORD**

Range: *0 or 1*

Default: 1 (enabled)

Caution: Do not change this value in Registry Editor, because various network bindings must also be changed. This parameter should only be changed by using the RAS Setup program.

Makes the server function like a NetBIOS gateway, allowing clients to access the LAN. If disabled, remote clients can access only the resources on the Remote Access server in a point-to-point connection; dial-in users cannot see the network or access network resources.

RAS NetBIOSGateway Subkey Entries

The Registry path for these entries is the following:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

\RemoteAccess

\Parameters

\NetbiosGateway



MAXxxx Entries for RAS NetBIOSGateway



RemoteListen Entry for RAS NetBIOSGateway



Enablexxx and Disablexxx Entries for RAS NetBIOSGateway



Other Entries for RAS NetBIOSGateway

MAXxxx Entries for RAS NetBIOSGateway

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \RemoteAccess
 \Parameters
 \NetbiosGateway

MaxBcastDgBuffered **REG_DWORD**

Range: 16 to 255

Default: 32

Sets the number of broadcast datagrams that the gateway buffers for a client. If you're using an application that communicates extensively through multicast or broadcast datagrams, increase this parameter so that the Remote Access server can deliver all datagrams reliably.

MaxDgBufferedPerGroupName **REG_DWORD**

Range: 1 to 255

Default: 10

Sets the number of datagrams that can be buffered per group name. Increasing this value buffers more datagrams per group name but also takes up more virtual memory.

MaxDynMem **REG_DWORD**

Range: 131072 to 4294967295

Default: 655350

Sets the amount of virtual memory used to buffer NetBIOS session data for each remote client.

Because the Remote Access server is a gateway between the slow line and the LAN, data is stored (buffered) in its memory when coming from the fast line (LAN) before it is forwarded to the slow line (asynchronous line).

The Remote Access server minimizes the usage of the system's physical memory by locking only a minimal set of pages (about 64K per client) and making use of virtual memory (up to **MaxDynMem**) to buffer the rest of the data. So, as long as there is enough space on the hard disk to expand PAGEFILE.SYS, you can increase this value if needed.

If you have an application with a LAN (fast) sender and an asynchronous (slow) receiver, and if the sender is sending more data at a time than the Remote Access server can buffer in **MaxDynMem**, the Remote Access server tries to apply a form of NetBIOS level flow control by not submitting NCB.RECEIVE on the session until it has enough buffer space to get incoming data. For this reason, if you have such an application, you should increase your NetBIOS SEND/RECEIVE timeouts so that the fast sender can keep pace with the slow receiver.

MaxNames **REG_DWORD**

Range: 1 to 255

Default: 255

Sets the number of unique NetBIOS names each client can have, with a limit of 255 names for all clients together.

Remote clients running Windows NT and Windows for Workgroups may need as many as seven or eight names each. To accommodate these workstations, set the **MaxNames** value to 8 and reduce the number of ports on the Remote Access server. If you have Windows NT or Windows for Workgroups clients dialing in to servers running Remote Access version 1.1 or earlier, set this parameter to 8 or greater.

MaxSessions REG_DWORD

Range: *1 to 255*

Default: 255

Sets the maximum number of simultaneous NetBIOS sessions each client can have, with a limit of 255 sessions for all clients together. If you have multiple clients connecting simultaneously with each running 4 or 5 sessions, decrease the value of this parameter so that the total number of sessions does not exceed 255.

RemoteListen Entry for RAS NetBIOSGateway

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \RemoteAccess
        \Parameters
          \NetbiosGateway
```

RemoteListen **REG_DWORD**

Range: 0 to 2

Default: 1 (messages)

Sets the remote NCB_LISTEN capability.

0 = Disables a client's ability to post NCB_LISTEN for any NetBIOS name. Because every remote listen posted consumes one session, setting this parameter to 0 saves sessions.

1 = Messages. Allows clients to post NCB_LISTEN on Windows NT Advanced Server message aliases only. If a remote client is running the Messenger service, it can then receive messages from LAN users, printers, and the like.

2 = All. Enables NCB_LISTEN for all remote client NetBIOS names, allowing clients to run NetBIOS server applications. This setting allows all clients to function as NetBIOS servers on the network.

It is best to leave the **RemoteListen** parameter set to the default, 1 (messages). Allowing NCB_LISTEN capability on remote clients can significantly drain system resources and therefore is not recommended.

If the **RemoteListen** parameter is set to 2, Remote Access posts an NCB_LISTEN on all NetBIOS names of Remote Access clients. Because the average Windows NT Advanced Server workstation has about seven or eight NetBIOS names assigned to it, the total number of NetBIOS names for which an NCB_LISTEN would be posted is 7 or 8 * 64 (the maximum number of clients per Remote Access server), which exceeds the 255 maximum.

Enablexxx and Disablexxx Entries for RAS NetBIOSGateway

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \RemoteAccess
 \Parameters
 \NetbiosGateway

DisableMcastFwdWhenSessionTraffic **REG_DWORD**

Range: 0 or 1

Default: 1 (enabled)

Allows NetBIOS session traffic (for example, Windows NT applications) to have priority over multicast datagrams (such as server messages). In other words, multicast datagrams are transferred only when there is no session traffic. Unless you're using an application that depends on multicast datagrams, leave this parameter enabled.

EnableBroadcast **REG_DWORD**

Range: 0 or 1

Default: 0 (disabled)

Determines whether broadcast datagrams are forwarded to remote workstations. Broadcast datagrams are not often useful and take up too much bandwidth on a slow link. Unless you're using an application that relies on broadcast datagrams, leave this parameter disabled.

EnableNetbiosSessionsAuditing **REG_DWORD**

Range: 0 or 1

Default: 0 (disabled)

Enable this parameter to record in the event log the establishment of NetBIOS sessions between the remote clients and the LAN servers. Enable this parameter to track the NetBIOS resources accessed on the LAN.

Other Entries for RAS NetBIOSGateway

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \RemoteAccess
 \Parameters
 \NetbiosGateway

AutoDisconnect **REG_DWORD**

Range: *0 to 60000 seconds (1000 minutes)*

Default: 1200 seconds (20 minutes)

Sets the time interval after which inactive connections are terminated. Inactivity is measured by lack of NetBIOS session data transfer, such as copying files, accessing network resources, and sending and receiving electronic mail. You may want to set this value to 0 seconds if clients are running NetBIOS datagram applications. Setting this value to 0 turns off **AutoDisconnect**.

MultiCastForwardRate **REG_DWORD**

Range: *-1 (disabled); 0 to 32,676 seconds*

Default: 5

Governs the multicasting of group name datagrams to all remote workstations. This parameter filters datagrams sent on group names by forwarding them at a specified time interval.

The value -1 disables forwarding. The value 0 guarantees delivery of group name datagrams. The value n forwards datagrams every n seconds, when $1 \leq n \leq 32,676$.

If the **EnableBroadcast** parameter is set to 0, broadcasts are not forwarded even if the **MultiCastForwardRate** parameter is set to a positive number (in this case, only multicast datagrams are forwarded). The line becomes overloaded. If **MultiCastForwardRate** is set to -1, broadcasts are still not forwarded even if **EnableBroadcast** is set to 1. See also **EnableBroadcast** in [Enablexxx and Disablexxx Entries for RAS NetBIOSGateway](#).

To save bandwidth for session traffic, filter the datagrams. However, if you have an application based on multicast datagrams, set this parameter to 0. This value guarantees delivery of all datagrams sent on group names from the LAN to the remote client.

NumRecvQueryIndications **REG_DWORD**

Range: *1 to 32*

Default: 3

Allows a Remote Access client to initiate multiple network connections simultaneously. If a remote client is running a NetBIOS application that does multiple NCB.CALL commands simultaneously, increase this parameter to improve performance.

RcvDgSubmittedPerGroupName **REG_DWORD**

Range: *1 to 32*

Default: 3

Determines the number of NetBIOS commands of the type Receive Datagram that can be submitted simultaneously per group name on the LAN stack. Keep this setting as small as possible to minimize the amount of memory consumed by system resources. Each datagram command received locks about 1.5K of physical memory in the system.

SizWorkBufs **REG_DWORD**

Range: *1024 to 65536*

Default: 4500

Sets the size of work buffers. The default setting is optimized for the server message block (SMB) protocol, the protocol between the workstation and the server running on the Windows NT Advanced Server system.

RAS AsyncMAC Subkey Entries

The Registry path for these entries is the following:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \AsyncMacn

 \Parameters

For changes to take effect, you must restart the computer.

MaxFrameSize REG_DWORD

Range: 576 to 1514

Default: 1514

Determines the maximum frame size. Use smaller frames for noisy links. A lower setting sends less data per frame, slowing performance. Do not change this parameter for previous versions of the Remote Access service. The value is negotiated between the server and Windows NT clients.

RAS RasHub Subkey Entries

The Registry path for the RasHub subkey is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \RasHub
        \Parameters
```

The subkeys RasHub01 and RasHub02 show, for example, that this installation of Remote Access is configured for two COM ports. In configuring ports, you can determine whether clients have access to the Remote Access server only (point-to-point connection) or to the network.

For changes to take effect, you must restart the computer.

NetworkAddress REG_SZ

Range: "xxxxxx"

Reassigns the first four bytes of the 6-byte IEEE address. For example, for the address "03-1F-2C-81-92-34" only the first four bytes are looked at.

Some applications depend on an IEEE adapter address being available. However, because the Remote Access Service uses modems (not real Ethernet adapters), it does not have an IEEE Ethernet address per se. This parameter lets you manually set an IEEE adapter address for Remote Access adapter bindings where applications demand it.

RAS RasMan Subkey Entries

The Registry path for the RasMan key is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \RasMan
        \Parameters
```

Logging REG_DWORD

Range: 0 or 1

Default: 0

Turns on information tracking for the modem using the DEVICE.LOG file. Set this value to 1 if you have modem problems that you cannot solve following documented procedures in the *Microsoft Windows NT Remote Access Administrator's Guide*. Logging begins the next time you dial in to connect through RAS (you do not need to restart your computer for the DEVICE.LOG file to be created).

Replicator Service Entries

The Registry path that contains entries for the Replicator service is the following:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Replicator

 \Parameters



Export and Import Entries for the Replicator Service



Other Entries for the Replicator Service

Export and Import Entries for the Replicator Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Replicator
 \Parameters

ExportList **REG_SZ**

Range: *List*

Default: (none)

Lists an unlimited number of servers or domains that receive notices when the export directory is updated. These servers subsequently replicate from the export server. If no *List* value is specified, the export server sends a notice to its domain. Separate multiple *List* names with a semicolon (;). This value is ignored if the value of **Replicate** is 2 (Import).

Do not use the UNC name when you specify a computername; that is, do not include two backslashes (\\) at the beginning of the name.

Use the Replicator controls in Server Manager or the Server icon in Control Panel to set this value.

ExportPath **REG_SZ or REG_EXPAND_SZ**

Range: *Pathname*

Default: %SystemRoot%\System32\Repl\Export

Specifies the export path. All files to be replicated must be in a subdirectory of the export directory.

This value is ignored if the value of **Replicate** is set to 2 (Import). Use the Replicator controls in Server Manager or the Server icon in Control Panel to set this value, which cannot be a UNC name.

ImportList **REG_SZ**

Range: *List*

Lists an unlimited number of servers or domains that receive notices when the import directory is updated. These servers subsequently replicate from the import server. If no *List* value is specified, updates come from the import server's domain. Separate multiple *List* names with a semicolon (;). This value is ignored if the value of **Replicate** is 1 (Export).

Do not use the UNC name when you specify a computername; that is, do not include two backslashes (\\) at the beginning of the name.

Use the Replicator controls in Server Manager or the Server icon in Control Panel to set this value.

ImportPath **REG_SZ or REG_EXPAND_SZ**

Range: *Pathname*

Default: %SystemRoot%\System32\Repl\Import

Specifies the path on the import server to receive replicas from the export servers. This value is ignored if the value of **Replicate** is 1 (Export). Use the Replicator controls in Server Manager or the Server icon in Control Panel to set this value, which cannot be a UNC name.

Other Entries for the Replicator Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Replicator
 \Parameters

CrashDir **REG_SZ**

Range: *First-level directory name*

This item is temporarily recorded in the Registry by the Replicator service. If it remains after a system repair, you can delete this entry using Registry Editor.

GuardTime **REG_DWORD**

Range: *0 to one-half of Interval minutes*

Default: 2 minutes

Sets the number of minutes an export directory must be stable (no changes to any files) before import servers can replicate its files.

This option applies only to directories with tree integrity.

Interval **REG_DWORD**

Range: *1 to 60 minutes*

Default: 5 minutes

Sets how often an export server checks the replicated directories for changes. This option is ignored on import servers.

Pulse **REG_DWORD**

Range: *1 to 10 cycles*

Default: 3

Specifies how often the export server repeats sending the last update notice. These repeat notices are sent even when no changes have occurred, so that import servers that missed the original update notice can receive the notice. The server waits the equivalent of (**Pulse** * **Interval**) minutes before sending each repeat notice.

Random **REG_DWORD**

Range: *1 to 120 seconds*

Default: 60

Specifies the maximum time that the import servers can wait before requesting an update. An import server uses the export server's value of **Random** to generate a random number of seconds (from 0 to the value of **Random**). The import server waits this long after receiving an update notice before requesting the replica from the export server. This prevents the export server from being overloaded by simultaneous update requests.

Replicate **REG_DWORD**

Range: *1, 2, or 3*

Default: 3

Specifies the Replicator action, according to the following:

- 1 = Export--the server maintains a master tree to be replicated.
- 2 = Import--the server receives update notices from the export server.
- 3 = Both-- the server is to export and import directories or files.

Use the Replicator controls in Server Manager or the Server icon in Control Panel to set this value.

Schedule Service Entries

There are no parameters that can be added for the Schedule service in this path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Schedule

You use the Schedule service to submit a job such as an executable or batch file to run at a later time.

You must define access controls on the Schedule key itself if you want to run in an account that is not an Administrator account.

Server Service Entries

With Registry Editor, you can modify the startup parameters for the Server service. Unless otherwise noted, these parameters are found in this path:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

\LanmanServer

\Parameters



Enablexxx Entries for the Server Service



Initxxx Entries for the Server Service



MAXxxx Entries for the Server Service



MINxxx Entries for the Server Service



Scavxxx Entries for the Server Service



Sessxxx Entries for the Server Service



Threadxxx Entries for the Server Service



Other Entries for the Server Service

The parameters that control network bindings for this service are described in [NetRules Subkey Entries](#).

Other Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanServer
 \Parameters

AlertSched **REG_DWORD**

Range: *1 to 65535 minutes*

Default: 5

Specifies in Microsoft LAN Manager and in Windows NT how often the server checks alert conditions and sends needed alert messages.

Announce **REG_DWORD**

Range: *1 to 65535 seconds*

Default: 240

Specifies how often a nonhidden server announces itself to the network. More frequent announcements keep client server tables more up to date, but cost network overhead and processing on client computers, because clients must process every announcement.

AnnDelta **REG_DWORD**

Range: *0 to 65535 milliseconds*

Default: 3000

Specifies the time by which the announcement period can vary. This helps to prevent several servers from continuously announcing simultaneously, thereby reducing network load peaks.

Comment **REG_SZ**

Range: *Text string*

Default: NULL

Provides the server's comment. This is sent in announcements and returned to NetServerGetInfo requests.

Disc **REG_DWORD**

Range: *0 to infinite minutes*

Default: 15 minutes

Specifies the amount of idle time that a circuit is allowed before being disconnected. If the virtual circuit has any open files or searches, it is not automatically disconnected. If this parameter is set to a low value, it saves server resources but hinders performance because of clients' overhead in reconnecting. This is equivalent to **autoDisconnect** in Microsoft LanManager.

DiskSpaceThreshold **REG_DWORD**

Range: *0 to 99 percent*

Default: 10 percent

Specifies the percentage of free disk space remaining before an alert is sent.

ErrorThreshold **REG_DWORD**

Range: *1 to 65535*

Default: 10

Sets the number of errors that can occur within an **AlertSched** interval before the server sends an

alert message.

Hidden REG_BINARY

Range: *0 or 1*

Default: 0 (false)

If this parameter is disabled, the server's name and comment can be viewed by others on the domain.
If enabled, the server's name and comment will not be announced.

IRPStackSize REG_DWORD

Range: *1 to 12*

Default: 4

Specifies the number of stack locations in I/O Request Packets (IRPs) used by the server. It may be necessary to increase this number for certain transports, MAC drivers, or local file system drivers. Each increment costs 36 bytes of memory per work item (that is, #work items * 36 bytes = total memory cost).

LinkInfoValidTime REG_DWORD

Range: *0 to 100,000 seconds*

Default: 60

Specifies the amount of time during which the transport link information is still valid. If more than this amount of time has passed since the last query, the server requires transport link information.

NetworkErrorThreshold REG_DWORD

Range: *1 to 100 percent*

Default: 5 percent

Triggers an alert whenever the percentage of failing network operations relative to total network operations exceeds this value during the **AlertSched** interval.

NumBlockThreads REG_DWORD

Range: *1 to 10 threads*

Default: (depends on configuration)

Specifies the number of threads set aside by the server to service requests that can block the thread for a significant amount of time. Larger values can increase performance but use more memory. A value that is too large can impede performance by causing excessive task switching.

OpenSearch REG_DWORD

Range: *1 to 2048 searches*

Default: 2048

Specifies the maximum number of outstanding searches on the server, per connection. A single client can have up to the **OpenSearch** number of active searches. This includes all types of searches, including MS-DOS, OS/2, and Windows NT.

OplockBreakWait REG_DWORD

Range: *10 to 180 seconds*

Default: 35

Specifies the time that the server waits for a client to respond to an oplock break request. Smaller values can allow detection of crashed clients more quickly but can potentially cause loss of cached data.

RawWorkItems REG_DWORD

Range: *1 to 512 items*

Default: (depends on configuration)

Specifies the number of special work items for raw I/O that the server uses. A larger value for this parameter can increase performance but costs more memory.

SizReqBuf **REG_DWORD**

Range: 512 to 65536 bytes

Default: 4356

Specifies the size of request buffers that the server uses. Small buffers use less memory; large buffers may improve performance.

Users **REG_DWORD**

Range: 1 to infinite

Default: 0xFFFFFFFF (infinite)

Specifies the maximum number of users that can be simultaneously logged on to the server.

XactMemSize **REG_DWORD**

Range: 64 KB to 16 MB0

Default: 1 MB

Specifies the maximum amount of virtual memory used by the Xactsrv service. A larger value for this parameter helps ensure that memory is available for downlevel clients but costs virtual address space and potentially costs pageable memory.

Enablexxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanServer
 \Parameters

EnableFCBopens **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Specifies whether MS-DOS File Control Blocks (FCBs) are folded together, so multiple remote opens are performed as a single open on the server. This saves resources on the server.

EnableOplockForceClose **REG_DWORD**

Range: 0 or 1

Default: 0 (false)

If a client has an opportunistic lock (oplock) and does not respond to an oplock break, there are two possible behaviors:

0 = False: fail the second open, thereby limiting access to the file. (This is typical behavior for a client running LAN Manager version 2.0.)

1 = True: force closed the open instance of the client that has the oplock, risking the loss of cached data. (This is typical behavior for a client running LAN Manager version 2.1.)

This parameter selects between the two.

EnableOplocks **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Specifies whether the server allows clients to use oplocks on files. Oplocks are a significant performance enhancement, but have the potential to cause lost cached data on some networks, particularly wide-area networks.

EnableRaw **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Specifies whether the server processes raw Server Message Blocks (SMBs). If enabled, this allows more data to be transferred per transaction and improves performance. However, it is possible that processing raw SMBs can impede performance on certain networks. This parameter is automatically tuned by the server.

EnableSoftCompat **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Specifies whether the server maps a request to a normal open request with shared-read access when the server receives a compatibility open request with read access. Mapping such requests allows several MS-DOS computers to open a single file for read access. However, this feature can potentially cause functionality problems with some MS-DOS applications.

Initxxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanServer
 \Parameters

InitConnTable **REG_DWORD**

Range: 1 to 128

Default: 8

Specifies the initial number of tree connections to be allocated in the connection table. The server automatically increases the table as necessary, so setting the parameter to a higher value is an optimization.

InitFileTable **REG_DWORD**

Range: 1 to 256

Default: 16

Specifies the initial number of file entries to be allocated in the file table of each server connection.

InitSearchTable **REG_DWORD**

Range: 1 to 2048

Default: 8

Specifies the initial number of entries in the connection's search table.

InitSessTable **REG_DWORD**

Range: 1 to 64

Default: 4

Specifies the initial number of session entries to be allocated in the session table of each server connection.

InitWorkItems **REG_DWORD**

Range: 1 to 512

Default: (depends on configuration)

Specifies the initial number of receive buffers, or work items, used by the server. Allocating work items costs a certain amount of memory initially, but not as much as having to allocate additional buffers later.

MAXxxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanServer
 \Parameters

MaxFreeConnections **REG_DWORD**

Range: *2 to 8 items*

Default: (Depends upon configuration)

Specifies the maximum number of free connection blocks maintained per endpoint.

MaxLinkDelay **REG_DWORD**

Range: *0 to 100,000 seconds*

Default: 60

Specifies the maximum time allowed for a link delay. If delays exceed this number, the server disables raw I/O for this connection.

MaxKeepSearch **REG_DWORD**

Range: *10 to 10000 seconds*

Default: 1800

Specifies the maximum time during which an incomplete MS-DOS search will be kept by the server. Larger values ensure better interoperability with MS-DOS utilities such as tree-copy and delete-node. However, larger values can cause unusual local behavior (such as a failure of a local directory-delete operation) and higher memory use on the server.

MaxMpxCt **REG_DWORD**

Range: *1 to 100 requests*

Default: 50

Provides a suggested maximum to clients for the number of simultaneous requests outstanding to this server. A higher value can increase server performance but requires higher use of server work items.

MaxNonpagedMemoryUsage **REG_DWORD**

Range: *1 MB to infinite bytes*

Default: (depends on system and server configuration)

Specifies the maximum size of nonpaged memory that the server can have allocated at any time. Adjust this parameter if you want to administer memory quota control.

MaxPagedMemoryUsage **REG_DWORD**

Range: *1 MB to infinite bytes*

Default: (depends on system and server configuration)

Specifies the maximum size of pageable memory that the server can have allocated at any time. Adjust this parameter if you want to administer memory quota control.

MaxWorkItems **REG_DWORD**

Range: *1 to 512 items*

Default: (depends on configuration)

Specifies the maximum number of receive buffers, or work items, the server can allocate. If this limit is reached, the transport must initiate flow control at a significant performance cost.

MaxWorkItemIdleTime **REG_DWORD**

Range: *10 to 1800 seconds*

Default: 300

Specifies the amount of time that a work item can stay on the idle queue before it is freed.

MINxxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanServer
 \Parameters

MinFreeConnections **REG_DWORD**

Range: *2 to 5 items*

Default: (depends upon configuration)

Specifies the minimum number of free connection blocks maintained per endpoint.

MinFreeWorkItems **REG_DWORD**

Range: *0 to 10 items*

Default: 2

Specifies the minimum number of available receive work items that are needed for the server to begin processing a potentially blocking SMB. A larger value for this parameter ensures that work items are available more frequently for nonblocking requests, but it also increases the likelihood that blocking requests will be rejected.

MinKeepSearch **REG_DWORD**

Range: *5 to 5000 seconds*

Default: 480

Specifies the minimum amount of time that the server will keep incomplete MSDOS searches, even if more search entries are needed. This parameter only matters when the server is near the maximum number of open searches it is allowed.

MinLinkThroughput **REG_DWORD**

Range: *0 to infinite bytes per second*

Default: 0

Specifies the minimum link throughput allowed by the server before it disables raw and opportunistic locks for this connection.

MinRcvQueue **REG_DWORD**

Range: *0 to 10 items*

Default: 2

Specifies the minimum number of free receive work items needed by the server before it begins allocating more. A larger value for this parameter helps ensure that there will always be work items available, but a value that is too large is simply inefficient.

Scavxxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \LanmanServer

 \Parameters

ScavTimeout **REG_DWORD**

Range: *1 to 300 seconds*

Default: 30

Specifies the time that the scavenger remains idle before waking up to service requests. A smaller value for this parameter improves the response of the server to various events but costs CPU cycles.

ScavQosInfoUpdateTime **REG_DWORD**

Range: *0 to 100,000 seconds*

Default: 300

Specifies the time that can pass before the scavenger goes through the list of active connections to update the link information.

Sessxxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \LanmanServer

 \Parameters

SessConns REG_DWORD

Range: *1 to 2048 connections*

Default: 2048

Specifies the maximum number of tree connections that can be made on the server via a single virtual circuit.

SessOpens REG_DWORD

Range: *1 to 2048 files*

Default: 2048

Specifies the maximum number of files that can be open on a single virtual circuit.

SessUsers REG_DWORD

Range: *1 to 64 users*

Default: 32

Specifies the maximum number of users that can be logged on to a server via a single virtual circuit.

Threadxxx Entries for the Server Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanServer
 \Parameters

ThreadCountAdd REG_DWORD

Range: *0 to 10 threads*

Default: (depends on configuration)

The server uses one worker thread per processor for the computer it is running on. This parameter indicates how many additional threads the server should use. More threads can improve performance but cost memory. Too many threads can hurt performance by causing excessive task switching.

ThreadPriority REG_DWORD

Range: *0, 1, 2, or 15*

Default: 1

Specifies the priority of all server threads in relation to the base priority of the process. Higher priority can give better server performance at the cost of local responsiveness. Lower priority balances server needs with the needs of other processes on the system. Values 0 to 2 are relative to normal or background processes. The default value of 1 is equivalent to the foreground process. A value of 15 runs the server threads at real-time priority--which is not recommended.

TCP/IP Transport Entries

The various TCP/IP keys do not appear in the Registry unless TCP/IP is installed using the Network icon in Control Panel.

With Registry Editor, you can modify the following parameters for the TCP/IP transport. This section does not include TCP/IP parameters that can be set using the Networks application in Control Panel.

The startup parameters defined in this section are found in these subkeys of HKEY_LOCAL_MACHINE\SYSTEM\System\CurrentControlSet\Services:



TCPIP\Parameters



__adapter_name#\Parameters\TCPIP, where *adapter_name#* indicates a Services subkey for a network adapter card



Ftpsvc\Parameters



NBT\Parameters



Streams\Parameters



Windows Sockets for TCP/IP

The parameters that control network bindings for this service are described in NetRules Subkey Entries.
See Also:

IsDomainMasterBrowser in Browser Service Entries

TCP/IP Parameters Subkey Entries

The entries for TCP/IP parameters appear under the following Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Services

\Tcpip

\Parameters



TCPxxx Entries for TCP/IP Parameters



UDPxxx Entries for TCP/IP Parameters



Other Entries for TCP/IP Parameters

Other Entries for TCP/IP Parameters

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Tcpip
 \Parameters

ArpCacheLife **REG_DWORD**

Range: *Number of Seconds*

Default: 600 (10 minutes)

Determines the default lifetime for entries in the ARP cache table. Once an entry is placed in the ARP cache, it is allowed to remain there until its lifetime expires or until its table entry is reused because it is the oldest entry.

ArpCacheSize **REG_DWORD**

Range: *Number*

Default: 62

Determines the maximum number of entries that the ARP cache table can hold. The ARP cache is allowed to grow dynamically until this size is reached. After the table reaches this size, new entries can only be added by replacing the oldest entries that exist.

DatabasePath **REG_EXPAND_SZ**

Range: *Pathname*

Default: %SystemRoot%\system32\drivers\etc

Determines where TCP/IP and NBT look for the hosts, services, networks, protocols, NETRC, and LMHOSTS files.

DefaultGateway **REG_SZ**

Range: *IP address*

Specifies the IP address of the system's default gateway. Set this parameter by choosing the Network icon in Control Panel.

DNSLookupOrder **REG_DWORD**

Range: *1, 2, 3, or 4*

Default: 1

Specifies the order that the workstation follows in trying to resolve hostnames. Set this value in the TCP/IP Connectivity dialog box after choosing the Network icon in Control Panel. Values are:

- 1 = DNS then hosts file
- 2 = hosts file then DNS
- 3 = DNS only
- 4 = hosts only

IpEnableRouter **REG_DWORD**

Range: *0 or 1*

Default: 0 (false)

Determines whether IP routing is enabled between local interfaces. IP routing is always enabled for packets generated by the local host. This parameter determines whether, for packets received on an interface and not destined for this host, IP will attempt to actively forward the packets to hosts that can be reached via its other interfaces.

IpReassemblyTimeout REG_DWORD

Range: *Number of seconds*

Default: 60 seconds

Determines how long IP accepts fragments when attempting to reassemble a previously fragmented packet. That is, if a packet is fragmented, all of the fragments must make it to the destination within this time limit; otherwise, the fragments will be discarded and the packet will be lost.

NameServer REG_SZ

Range: *String*

Specifies an array of DNS server IP addresses. Set this parameter by choosing the Network icon in Control Panel.

SearchList REG_SZ

Range: *String*

Specifies an array of DNS domain name suffixes for DNS queries. Set this parameter by choosing the Network icon in Control Panel.

TCPxxx Entries for TCP/IP Parameters

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Tcpip
 \Parameters

TcpDisableReceiveChecksum **REG_DWORD**

Range: *0 or 1*

Default: 0 (false, that is, checksums will be checked on receives)

Specifies whether Checksums is disabled on receive.

TcpDisableSendChecksum **REG_DWORD**

Range: *0 or 1*

Default: 0 (false, that is, checksums will be generated on sends)

Specifies whether Checksums is disabled on send.

TcpKeepCnt **REG_DWORD**

Range: *Number in seconds*

Default: 120

Specifies how often TCP/IP will generate keep-alive traffic. When TCP/IP determines that no activity has occurred on the connection within the specified time, it generates keep-alive traffic to probe the connection. After trying **TcpKeepTries** number of times to deliver the keep-alive traffic without success, it marks the connection as down.

TcpKeepTries **REG_DWORD**

Range: *Number*

Default: 20

Specifies the maximum number of times that TCP/IP will attempt to deliver keep-alive traffic before marking a connection as down.

TcpLogLevel **REG_DWORD**

Range: *Number*

Default: 16 (log everything)

Specifies how verbose TCP/IP should be about logging events in the event log. The highest level of verbosity is 16, and 1 is the lowest level. The following shows general information about these levels:

1 = Only the most critical errors

4 = Serious protocol violations

8 = Nonserious protocol violations

12 = Information about unusual events

16 = Information about unusual events that some networks normally allow

TcpMaxConnectAttempts **REG_DWORD**

Range: *Number*

Default: 3

Specifies the maximum number of times TCP/IP attempts to establish a connection before reporting failure. The initial delay between connection attempts is 3 seconds. This delay is doubled after each attempt.

TcpMaxRetransmissionAttempts REG_DWORD

Range: *Number*

Default: 7

Specifies the maximum number of times that TCP/IP attempts to retransmit a piece of data on an established connection before ending the connection. The initial delay before retransmitting is based on the current estimate TCP/IP makes of the round-trip time on the connection. This delay is doubled after each retransmission. Acknowledgment of the data results in a recalculation of the estimate for the round-trip time.

TcpNumConnections REG_DWORD

Range: *Number*

Default: 64

Specifies the maximum number of TCP endpoints.

TcpRecvSegmentSize REG_DWORD

Range: *Bytes*

Default: 1460

Specifies the maximum receive segment size.

TcpSendDownMax REG_DWORD

Range: *Number*

Default: 16384

Specifies the maximum number of bytes queued by TCP/IP.

TcpSendSegmentSize REG_DWORD

Range: *Bytes*

Default: 1460

Specifies the maximum send segment size.

TcpWindowSize REG_DWORD

Range: *Number*

Default: 8192

Sets the size of the TCP send and receive windows, which is the amount of data that can be accepted in a single transaction. This parameter is important in transferring files between a client and a server and is critical for performance for one-way traffic, such as for FTP.

UDPxxx Entries for TCP/IP Parameters

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Tcpip
 \Parameters

UdpDisableReceiveChecksum **REG_DWORD**

Range: *0 or 1*

Default: 0 (false, that is, checksums will be checked on receives)

Specifies whether Checksums is disabled on receive.

UdpDisableSendChecksum **REG_DWORD**

Range: *0 or 1*

Default: 0 (false, that is, checksums will be generated on sends)

Specifies whether Checksums is disabled on send.

UdpNumConnections **REG_DWORD**

Range: *Number*

Default: 64

Specifies the maximum number of UDP endpoints.

Adapter Card Parameters for TCP/IP

These parameters for TCP/IP are specific to individual network adapter cards. These appear under the following Registry path, where *adapter name#* refers to the Services subkey for the specific adapter card:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
          \Tcpip
```



Broadcast Entries for Adapter Cards with TCP/IP



Maximum Transmission Unit Entries for Adapter Cards



Other TCP/IP Entries for Adapter Cards

Broadcast Entries for Adapter Cards with TCP/IP

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 adapter name#
 \Parameters
 \Tcpip

BroadcastType **REG_DWORD**

Range: 0 or 1

Default: 1 (all 1's)

Determines whether broadcast packets contain all 0's or all 1's as the broadcast address. The most common broadcast type is all 1's. The all-0's setting is provided for compatibility with BSD 4.2 systems.

ForwardBroadcasts **REG_DWORD**

Range: 0 or 1

Default: 0 (false)

Specifies whether broadcasts should be forwarded between adapters. If enabled, broadcasts seen by this interface are forwarded to other IP interfaces.

Other TCP/IP Entries for Adapter Cards

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
          \Tcpip
```

IPAddress **REG_SZ**

Range: *IP address*

Specifies the IP address for the specific network adapter card. Set this value by choosing the Network icon in Control Panel.

KeepAlive **REG_DWORD**

Range: *0 or 1*

Default: 1 (true)

Determines whether TCP connections that request keep-alive packets result in keep-alive packets being sent. This feature is used to determine when inactive connections can be disconnected. When a connection becomes inactive, keep-alive packets are periodically exchanged. When 20 consecutive keep-alive packets go unanswered, the connection is broken. This disconnect is initiated by the endpoint that is sending keep-alive packets.

SubnetMask **REG_SZ**

Range: *IP address*

Specifies the subnet mask for this specific network adapter card. Set this parameter by choosing the Network icon in Control Panel.

Trailers **REG_DWORD**

Range: *0 or 1*

Default: 0 (false)

Specifies whether the trailer format is used. This feature provides compatibility with BSD 4.2 systems. When this feature is enabled, TCP/IP header information follows the data area of IP packets.

Maximum Transmission Unit Entries for Adapter Cards

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \adapter name#
        \Parameters
          \Tcpip
```

MTU **REG_DWORD**

Range: *Number in octets*

Default: 0 (That is, use the value supplied by the adapter.)

Specifies the maximum transmission unit size of an interface. Each interface used by TCP/IP may have a different MTU value specified. The MTU is usually determined through negotiation with the lower driver, using that lower driver's value. However, that value may be overridden.

Ideally, the MTU should be large enough to hold any datagram in one frame. The limiting factor is usually the technology making the transfer. Some technologies limit the maximum size to as little as 128; Ethernet limits transfers to 1500; and proNet-10 allows as many as 2044 octets per frame.

Datagrams larger than the MTU value are automatically divided into smaller pieces called fragments; size is a multiple of eight octets. Fragmentation usually occurs somewhere through which the traffic must pass whose MTU is smaller than the encapsulated datagram. If fragmentation occurs, the fragments travel separately to the destination computer, where they are automatically reassembled before the datagram is processed.

RouterMTU **REG_DWORD**

Range: *Number in octets*

Default: 0 (That is, use the value supplied by the lower interface.)

Specifies the maximum transmission unit size that should be used when the destination IP address is on a different subnet. Each interface used by TCP/IP may have a different **RouterMTU** value specified. In many implementations, the value of **RouterMTU** is set to 576 octets. This is the minimum size that must be supported by any IP node. Because modern routers can usually handle MTUs larger than 576 octets, the default value for this parameter is the same value as that used by **MTU**.

FTP Server Service Entries for TCP/IP

The following Registry path contains parameters that affect the behavior of the FTP server service component:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Ftpsvc
        \Parameters
```

The Ftpsvc subkey does not appear until you install the FTP service using the Network icon in Control Panel. Also, you must restart the FTP server service (Ftpsvc) using the Services icon in Control Panel for any changes to these values to take effect.

There can also be an AccessCheck subkey under Ftpsvc, which allows access to FTP for new users. If the AccessCheck subkey exists, but cannot be opened, the user is refused FTP services. If the subkey exists but can only be opened for read access, the user is granted read-only FTP access. If the subkey does not exist, it is not used to influence FTP access. By default, this subkey does not exist and therefore has no impact on FTP operations. An administrator can create this Registry subkey and attach specific access controls, which will serve to control user access to the FTP service.



FTP Login Parameters for TCP/IP



Other FTP Parameters for TCP/IP

FTP Login Parameters for TCP/IP

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Ftpsvc
 \Parameters

AllowAnonymous **REG_DWORD**

Range: *0 or 1*

Default: 1 (true anonymous logins are allowed)

Controls anonymous logins. Anonymous logins are only allowed if this value is non-zero (true).

AnonymousOnly **REG_DWORD**

Range: *0 or 1*

Default: 0 (false non-anonymous logins are allowed)

When this value is 1, only anonymous logins are allowed. Otherwise, non-anonymous logins are allowed as well.

AnonymousUserName **REG_SZ**

Range: *UserName*

Default: "Guest"

Contains the anonymous login alias. When a user attempts an anonymous login, the username specified (anonymous) is mapped to this Registry value for authentication and impersonation.

LogAnonymous **REG_DWORD**

Range: *0 or 1*

Default: 0 (false do not log successful anonymous logins)

When this value is 1, all successful anonymous logins are logged to the system event log.

LogNonAnonymous **REG_DWORD**

Range: *0 or 1*

Default: 0 (false, that is, do not log successful non-anonymous logins)

When this value is 1, all successful non-anonymous logins are logged to the system event log.

Other FTP Parameters for TCP/IP

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \Ftpsvc
 \Parameters

AnnotateDirectories **REG_DWORD**

Range: *0 or 1*

Default: 0 (false do not send directory annotations)

When this value is 1, every time a user changes directories (that is, sends the server a CWD command), an attempt is made to open a file named ~FTPSVC~.CKM in the new directory. If this file is found, its contents are sent to the user as part of the successful reply to the CWD command. This may be used to attach annotations to specific directories.

This value is used as a default for new users. Users can toggle their own personal annotate directories flag with the site-specific CKM command (SITE CKM).

ConnectionTimeout **REG_DWORD**

Range: *Seconds*

Default: 600 (10 minutes)

Specifies the time to allow clients to remain idle before forcibly disconnecting them. This prevents idle clients from consuming server resources indefinitely.

This value may be set to 0 if time-outs are not to be enforced. If set to 0, idle clients may remain connected indefinitely.

ExitMessage **REG_SZ**

Range: *Message*

Default: "Goodbye."

Specifies a signoff message sent to an FTP client upon receipt of a QUIT command.

GreetingMessage **REG_MULTI_SZ**

Range: *Strings*

Default: None (no special greeting message)

Specifies the message (if this value exists in the Registry) to be sent to new clients after their account has been validated. In accordance with de facto Internet behaviour, if a client logs on as anonymous and specifies an identity starting with a '-' (minus), then this greeting message is not sent.

HomeDirectory **REG_EXPAND_SZ**

Range: *Path*

Default: C:\

Specifies the initial home directory for new clients. After a new client is validated, an attempt is made to change to this directory with the Chdir command. If this directory is inaccessible, the client is refused FTP services. If Chdir is successful, then an attempt is made to change to a directory with the same name as the clients username. If this fails, an attempt is made to change to a directory called DEFAULT. If this fails, the current directory is left at home.

If a new client connects and finds the home directory is inaccessible, an event is written to the event log.

MaxClientsMessage **REG_SZ**

Range: *Message*

Default: "Maximum clients reached, service unavailable."

Specifies the message (if this value exists in the Registry) to be sent to a client if the maximum number of clients has been reached or exceeded. This indicates that the server is currently servicing the maximum number of simultaneous clients and is refusing additional clients. See

MaxConnections.

MaxConnections **REG_DWORD**

Range: *0 or 1*

Default: 20

Specifies the maximum number of simultaneous clients the server will service. This value may be set to 0 if there is to be no limit on simultaneous clients.

MsdosDirOutput **REG_DWORD**

Range: *0 or 1*

Default: 1 (truedirectory listings will look like MS-DOS)

When this value is 1, the output of the LIST command (usually sent as a result of a DIR command from the client) will look like the output of the MS-DOS dir command. If this value is 0, the output of the LIST command looks like the output of the Unix ls command.

This value also controls slash flipping in the path sent by the PWD command. When this value is 1 (true), the path contains backward slashes (\). If this value is 0 (false), the path contains forward slashes (/).

This value is used as a default for new users. Users can toggle their own personal MS-DOS directory output flag with the site-specific DIRSTYLE command (SITE DIRSTYLE).

ReadAccessMask **REG_DWORD**

Range: *BitFields*

Default: 0 (all read access denied)

This value is a bitmask and controls the read ability of the various disk volumes in the system. Drive A corresponds to bit 0, drive B corresponds to bit 1, drive C corresponds to bit 2, and so on. A user may only read from a specific volume if the corresponding bit is set.

WriteAccessMask **REG_DWORD**

Range: *BitFields*

Default: 0 (all write access denied)

This value is a bitmask and controls the write ability of the various disk volumes in the system. Drive A corresponds to bit 0, drive B corresponds to bit 1, drive C corresponds to bit 2, and so on. A user may only write to a specific volume if the corresponding bit is set.

NBT Parameters for TCP/IP

NBT is NetBIOS over TCP/IP. Parameters for TCP/IP are also configured under NBT in the following Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \NBT
        \Parameters
```

MaxPreload **REG_DWORD**

Range: *Number*

Default: 100

Specifies the maximum NBT number of entries for LMHOSTS that are preloaded into the NBT NetBIOS name cache. LMHOSTS is a file located in the directory specified by **DatabasePath**.

NbProvider **REG_SZ**

Range: *DLL name*

Default: `_tcp`

This value is only present for network services that provide a NetBIOS interface. Its presence causes LanMan information to be maintained for the transport. The value data represents the RPC provider string used to select the proper DLL for interfacing RPC to the transport.

NbtKeepAlive **REG_DWORD**

Range: *Number in seconds*

Default: 0 (Do not generate NBT keep-alive traffic.)

Specifies how often NBT will generate keep-alive traffic. When NBT determines that no activity has occurred on a connection for the specified time interval, it will generate keep-alive traffic to probe the connection. If TCP/IP is unable to deliver this traffic, it marks the connection as down and notifies NBT.

PermanentName **REG_SZ**

Range: *Unique name*

Default: The value of *IPAddress* in dotted decimal

Specifies the permanent name of the NetBIOS node for NBT. In many NetBIOS implementations, this is the MAC address. This name must be unique.

ScopeID **REG_SZ**

Range: *String*

Specifies the RFC1001/1002 scope name for this system. Set this value by choosing the Network icon in Control Panel.

Streams Parameters for TCP/IP

The TCP/IP parameter for Streams are found under the following Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Services

 \Streams

 \Parameters

MaxMemoryUsage **REG_DWORD**

Range: *Number of bytes*

Default: No limit

Specifies the maximum amount of memory that can be allocated to the Streams environment. Once this limit is reached, Streams will fail allocation requests made by Streams-based drivers.

Windows Sockets Entries for TCP/IP

All Windows Sockets parameters can be set by choosing the Network icon in Control Panel. These parameters are found in two locations, as shown here.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \Winsock
        \Parameters
```

Transports **REG_Multi_SZ**

Range: *Strings*

Default: Depends on installation

Contains the Registry key names of installed transports that support Windows Sockets. If TCP/IP is the only installed transport that supports Windows Sockets, then this value is Tcpip. The Windows Sockets DLL uses the strings in **Transports** to find information about each transport.

Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \TCPIP
        \Parameters
          \Winsock
```

HelperDllName **REG_EXPAND_SZ**

Range: *Path and filename*

Default: Depends on the transport; %SystemRoot%\system32\wshtcpip.dll for TCP/IP.

Specifies the name of the Windows Sockets helper DLL for the TCP/IP transport. This value is set by the Windows Sockets DLL and is not a user settable parameter.

IRPStackSize **REG_DWORD**

Range: *Number*

Default: 4

Specifies the number of IRP stack locations needed by AFD, the driver used for Windows Sockets. The default is sufficient for all existing transports, but new transports may be developed that need more IRP stack locations.

Mapping **REG_BINARY**

Default: Depends on transport

Identifies the address families, socket types, and protocols supported by the transport. This value is set by the Windows Sockets DLL and is not a user settable parameter.

MaxSockAddrLen **REG_DWORD**

Range: *Octets*

Specifies the maximum length of socket addresses for the INET sockets family. This value is set by the Windows Sockets DLL and is not a user settable parameter.

MinSockAddrLen **REG_DWORD**

Range: *Octets*

Specifies the minimum length of socket addresses for the INET sockets family. This value is set by the Windows Sockets DLL and is not a user settable parameter.

See Also:

[AppleTalk Winsock Entries](#)

UPS Service Entries

The Registry does not contain information for the UPS service until the user checks the Uninterruptible Power Supply Is Installed checkbox in the UPS dialog box and then chooses the OK button. Changes to settings should be made by using the UPS icon in Control Panel.

The UPS service will not start unless the UPS subkey is present in the Registry, all parameters are present in the Registry, and all values are within the correct range. If any of these elements are missing or in error, a message announces that the UPS service is not correctly configured. All corrections can be made using the UPS icon in Control Panel.

The UPS parameters remain in the Registry if the user uninstalls UPS.

The path for the UPS subkey is the following:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Services
      \UPS
```

BatteryLife **REG_DWORD**

Range: *2 to 720 minutes*

Default: 2 (minutes)

Specifies the life of the UPS backup battery when fully charged.

CommandFile **REG_EXPAND_SZ**

Range: *Filename*

Default: (empty)

Specifies the name of a command file to execute immediately before shutting down.

FirstMessageDelay **REG_DWORD**

Range: *0 to 120 seconds*

Default: 5 (seconds)

Specifies the number of seconds between initial power failure and the first message sent to the users. If power is restored within the **FirstMessageDelay** time, no message is sent, although the event is logged.

MessageInterval **REG_DWORD**

Range: *5 to 300 seconds*

Default: 120 (seconds)

Specifies the number of seconds between messages sent to users to inform them of power failure.

Options **REG_DWORD**

Range: *Value*

Defines the bit mask for messages related to options in the UPS dialog box.

Bit Masks for UPS Messages

Port **REG_SZ**

Range: *Port name*

Default: COM1:

Specifies the name of the serial port the UPS is connected to.

RechargeRate **REG_DWORD**

Range: *1 to 250 minutes*

Default: 100 (minutes)

Specifies the recharge rate of the UPS backup battery.

Bit Masks for UPS Messages

0x00000001 = Installed

0x00000002 = PowerFailSignal

0x00000004 = LowBatterySignal

0x00000008 = CanTurnOff

0x00000010 = PosSigOnPowerFail

0x00000020 = PosSigOnLowBattery

0x00000040 = PosSigShutOff

0x00000080 = CommandFile

Workstation Service Entries

You can modify the startup parameters for the Workstation service using the Registry Editor. Unless otherwise indicated, these value entries are found in the following Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

The parameters that control network bindings for this service are described in [NetRules Subkey Entries](#).



[Bufxxx Entries for the Workstation Service](#)



[Illegal Datagram Entries for the Workstation Service](#)



[Lockxxx Entries for the Workstation Service](#)



[MAXxxx Entries for the Workstation Service](#)



[Usexxx Entries for the Workstation Service](#)



[Other Named Pipe Entries for the Workstation Service](#)



[Other Entries for the Workstation Service](#)

Bufxxx Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

BufFilesDenyWrite **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Specifies whether the redirector should cache files that are opened with only FILE_SHARE_READ sharing access. Usually, if a file is opened with FILE_SHARE_READ specified, the file cannot be buffered because other processes may also be reading that file. This optimization allows the redirector to buffer such files. This optimization is safe because no process can write to the file.

Disable this parameter if it is necessary to preserve the strict semantics of the sharing modes specified.

BufNamedPipes **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Indicates whether the redirector should buffer character-mode named pipes.

Disable this parameter to guarantee that all pipe write operations are flushed to the server immediately and to disable read ahead on character-mode named pipes.

BufReadOnlyFiles **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Specifies whether the redirector should cache files that are read-only. Usually, if a read-only file is opened, the file cannot be buffered because other processes may also be reading that file. This optimization allows the redirector to buffer such files. This optimization is safe because no process can write to the file. However, another user can modify the file to enable writing to the file, causing loss of data.

Disable this parameter if it is necessary to preserve the strict semantics of the sharing modes specified.

Illegal Datagram Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

IllegalDatagramResetTime **REG_DWORD**

Range: *Number of seconds*

Default: 60

Specifies the span of time during which the number of illegal datagram events is counted. Because Windows NT logs all illegal datagrams, it is possible for the event log to be filled with a proliferation of these in a short amount of time. This entry and the **NumIllegalDatagramEvents** entry work together to limit the number of illegal datagrams that are recorded in the log within a certain span of time.

NumIllegalDatagramEvents **REG_DWORD**

Range: *Number of events*

Default: 5

Specifies the maximum number of datagram events to be logged within the span of time specified by the **IllegalDatagramResetTime** parameter. Because Windows NT logs all illegal datagrams, the event log can be filled with a proliferation of these in a short time. This entry and the **IllegalDatagramResetTime** entry work together.

Lockxxx Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

LockIncrement **REG_DWORD**

Range: *Number of milliseconds*

Default: 10

This parameter is not used for Win32 applications. However, if OS/2-based applications request that a lock operation waits forever, and if the lock cannot be immediately granted on a non-LAN Manager version 2.0 server, this parameter controls the rate at which the redirector ramps back the failed lock operations.

This parameter should not be changed unless you are running an OS/2-based application that requests lock operations that might fail.

LockMaximum **REG_DWORD**

Range: *Number of milliseconds*

Default: 500

Used to configure the lock backoff package. This parameter exists to prevent an errant application from "swamping" a server with nonblocking requests where there is no data available for the application.

LockQuota **REG_DWORD**

Range: *Bytes of data*

Default: 4096 (bytes)

Specifies the maximum amount of data that is read for each file using this optimization if the **UseLockReadUnlock** parameter is enabled.

Increase this value if your application performs a significant number of lock-and-read style operations. (This means performing lock operations and immediately reading the Contents of locked data.) It is conceivable that you could cause the system to run out of paged pool, but only by increasing this value to a few megabytes and by using an application that locks millions-of-byte ranges.

MAXxxx Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

MaxCmds **REG_DWORD**

Range: 0 to 255

Default: 15

Specifies the maximum number of work buffers that the redirector reserves for performance reasons.

Increase this value to increase your network throughput. If your application performs more than 15 simultaneous operations, you might want to increase this value. Because this parameter actually controls the number of execution threads that can be simultaneously outstanding at any time, your network performance will not always be improved by increasing this parameter. Each additional execution threads takes about 1K of nonpaged pool if you actually load up the network. Resources will not be consumed, however, unless the user actually makes use of them.

MaxCollectionCount **REG_DWORD**

Range: 0 to 65535 bytes

Default: 16

Specifies the threshold for character-mode named pipes writes. If the write is smaller than this value, the write will be buffered. Adjusting this value may improve performance for a named-pipe application (but it will not affect SQL server applications).

Usexxx Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

Use512ByteMaxTransfer **REG_DWORD**

Range: 0 or 1

Default: 0 (false)

Specifies whether the redirector should only send a maximum of 512 bytes in a request to an MS-Net server regardless of the servers-negotiated buffer size. If this parameter is disabled, request transfers from the Windows NT redirector could cause the MS-Net server to crash.

UseLockReadUnlock **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Indicates whether the redirector uses the lock-and-read and write-and-unlock performance enhancements.

When this value is enabled, it generally provides a significant performance benefit. However, database applications that lock a range and don't allow data within that range to be read will suffer performance degradation unless this parameter is disabled.

UseOpportunisticLocking **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Indicates whether the redirector should use opportunistic-locking (oplock) performance enhancement. This parameter should be disabled only to isolate problems.

UseRawRead **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Enables the raw-read optimization. This provides a significant performance enhancement on a local area network

UseRawWrite **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Enables the raw-write optimization. On a LAN, this provides a significant performance enhancement.

UseUnlockBehind **REG_DWORD**

Range: 0 or 1

Default: 1 (true)

Indicates whether the redirector will complete an unlock operation before it has received confirmation from the server that the unlock operation has completed. Disable this parameter only to isolate problems or to guarantee that all unlock operations complete on the server before completing the application's unlock request.

UseWriteRawData **REG_DWORD**

Range: *0 or 1*

Default: 1 (true)

Enables the raw-write-with-data optimization. This allows the redirector to send 4 KB of data with each write-raw operation. This provides a significant performance enhancement on a local area network.

Other Named Pipe Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

CharWait **REG_DWORD**

Range: *0 to 65535 milliseconds*

Default: 3600

Specifies time to wait for an instance of a named pipe to become available when opening the pipe.

Increase this value if your pipe server application is typically very busy.

CollectionTime **REG_DWORD**

Range: *0 to 65535000 milliseconds*

Default: 250

Specifies the maximum time that write-behind data will remain in a character-mode pipe buffer.

Changing this value may cause a named pipe application's performance to improve (but it does not affect SQL Server applications).

PipeIncrement **REG_DWORD**

Range: *Number of milliseconds*

Default: 10

Controls the rate at which the redirector "backs off on failing nonblocking pipe reads.

This parameter is used to prevent an errant application from swamping a server with nonblocking requests where there is no data available for the application. You can use the backoff statistics to tune this parameter to be more efficient for an application that uses nonblocking named pipes (except for SQL Server applications).

PipeMaximum **REG_DWORD**

Range: *Number of milliseconds*

Default: 500

Controls the maximum time at which the redirector "backs off on failing non-blocking pipe reads.

This parameter exists to prevent an errant application from swamping a server with nonblocking requests where there is no data available for the application. You can use the backoff statistics to tune this parameter to be more efficient for an application that uses nonblocking named pipes (except for SQL Server applications).

SizCharBuf **REG_DWORD**

Range: *64 to 4096 bytes*

Default: 512

Specifies the maximum number of bytes that will be written into a character-mode pipe buffer.

Adjusting this value may improve performance for a named-pipe application (but it will not affect SQL server applications).

Timeout Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

CacheFileTimeout **REG_DWORD**

Range: *Number of seconds*

Default: 10

Specifies the maximum time that a file will be left in the cache after the application has closed the file.

Increase the value of this parameter if you are performing operations on the server that could cause files to be reopened more than 10 seconds after the application has closed them. For example, if you are performing a build over the network, you should increase this parameter's value.

SessTimeout **REG_DWORD**

Range: *10 to 65535 seconds*

Default: 45

Specifies the maximum amount of time that the redirector allows an operation that is not long-term to be outstanding. The redirector uses this value to establish the extra time to wait after the expected length of time for the SMB response. The time that the redirector actually waits for a server to respond to an SMB is roughly $[(\text{SMB size} + \text{size of data sent or received}) / \text{bytes per second}] +$

SessTimeOut. This is a system-wide parameter used for all protocols, including TCP/IP. However, **SessTimeOut** does not apply to certain types of SMBs such as transaction commands that have their own time-out variable in the SMB.

Other Entries for the Workstation Service

Registry path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet
 \Services
 \LanmanWorkstation
 \Parameters

DormantFileLimit **REG_DWORD**

Range: *Number of files*

Default: 45

Specifies the maximum number of files that should be left open on a share after the application has closed the file.

This parameter exists because the default configuration of LAN Manager servers only allow a total of 60 open files from remote clients and 50 from each client workstation. Because the Windows NT redirector may keep files open in the cache after an application has closed the file, this means that the redirector may overload a misconfigured LAN Manager server. To correct this problem, either reduce this value, or increase the values for the LAN Manager server's **maxSessopens** and **maxOpens** parameters.

KeepConn **REG_DWORD**

Range: *1 to 65535 seconds*

Default: 600

Specifies the maximum amount of time that a connection can be left dormant. This parameter is the redirector equivalent of the **Disc** parameter in the Services\LanmanServer\Parameters subkey.

As a general rule, try increasing this value if your application closes and opens UNC files to a server less frequently than 10 minutes apart. This decreases the number of reconnections made to a server.

LogElectionPackets **REG_DWORD**

Range: *0 or 1*

Default: 0 (false)

Specifies whether the Browser should generate events when election packets are received.

MailslotBuffers **REG_DWORD**

Range: *Number of buffers*

Default: 5

Specifies the maximum number of buffers available to process mailslot messages. If your application uses many mailslot operations, set this higher to avoid losing mailslot messages.

OtherDomains **REG_SZ**

Range: *DomainNames*

Default: (none)

Specifies the Microsoft LAN Manager domains to be listed for browsing.

ReadAheadThroughput **REG_DWORD**

Range: *Kilobytes per second*

Default: 0xffffffff

Specifies the throughput required on a connection before the cache manager is told to enable read ahead.

ServerAnnounceBuffers REG_DWORD

Range: *Number*

Default: 20

Specifies the maximum buffers used to process server announcements. If your network has many servers, you can increase this value to avoid losing server announcements.

This parameter is found under the LanmanWorkstation\Parameters\Static subkey.

Transports REG_MULTI_SZ

Range: *List*

Default: None

Lists the transports that the redirector services and is found under the LanmanWorkstation\Linkage subkey. You should modify it by choosing the Network icon in Control Panel.

UtilizeNtCaching REG_DWORD

Range: *0 or 1*

Default: 1 (true)

Indicates whether the redirector uses the cache manager to cache the contents of files. Disable this parameter only to guarantee that all data is flushed to the server immediately after it is written by the application.

Registry Entries for Microsoft Mail

The parameters used by the Microsoft Mail application provided with Windows NT appear under this subkey:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \Mail

This key includes the following subkeys:

-  Address Book Entries for Mail
-  Custom Commands Entries for Mail
-  Custom Messages Entries for Mail
-  Custom Menus Entries for Mail
-  Microsoft Mail Entries
-  MMF Entries for Mail
-  Mac FileTypes Entries for Mail
-  MS Proofing Tools Entries for Mail
-  Providers Entries for Mail

Many of the entries in these subkeys have default values and won't be present in the Mail subkeys. To change the appearance and behavior of the Mail application, use the Mail menu commands instead of editing the Mail entries directly. Some of the options that you specify in the Mail application are stored in your mail message file (.MMF) instead of the Mail Registry entries.

These keys are created in HKEY_CURRENT_USER when you first run Mail. If your system previously contained a Windows for MS-DOS version of MSMAIL.INI, its contents are migrated to the Registry when you first run Mail under Windows NT.

Address Book Entries for Mail

Entries in this subkey are used by the Address Book support functions in the Mail program. Most of the entries for this subkey use default values specified by the Mail program, and the Address Book subkey might not be present under the Mail key. The entries in this subkey control the default address directory displayed in the Address Book. Do not change any of these entries if they appear.

This is the Registry path for this subkey:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \Address Book
```

Custom Commands Entries for Mail

Each entry under this key specifies a custom command that can be installed into one of the Mail menus at run-time. These entries can appear both in the Microsoft Mail key and in the SHARED32.INI file in the directory defined under the Microsoft Mail subkey as the value of **SharedExtensionsDir**.

This is the Registry path for this subkey:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Custom Commands

The Custom Commands subkey can contain one or more of the following entries:

tag **REG_SZ**

This specifies the descriptive identifier for the command in the following format:

tag = *version;menu;name;position;DLL name, ordinal;command; event map;status text;Help file;help context;*

For example:

IC1= 3.0;help;&Out of Office;10;<ExtsDir>BINEXT\OOF32.DLL;3;;Out of Office Email Notification

version = The version of Mail that the custom command is compatible with.

menu = The menu where the custom command is to be added, such as File or Help.

name = The command name to appear on the menu. Include an ampersand just before the letter that is to serve as an ALT+key accelerator.

position = The zero-relative position within its menu where the command is to appear. A value of -1 places the command at the end of the menu.

DLL name = Name or path of the DLL where the custom command resides. This entry can optionally be followed by a comma and the ordinal of the entry point to call (default is ,1).

command = The command string passed as one of the parameters to the DLL entry point for the command.

event map = A sequence of up to 16 digits identifying the Mail events that the custom command supports. Each can be 0 or 1 to indicate if the DLL is to be called for a specific event. Currently three events are defined; the rest are reserved and must be zero (or, as in the examples above, this whole section can be empty): The first digit means Mail startup; the second digit means Mail exit; and the third digit means the arrival of a new message.

status text = Text to be displayed in the Mail status bar when the user highlights the command in the menu.

Help file = Windows NT Help file to be invoked when the user presses F1 while the command is selected. The specified filename is passed to the Windows NT Help program. (optional)

help context = Passed to the Windows NT help program along with the Help file name. Use -1 (Help file index) if there is no specific entry in the Help file for this command. (optional)

<ExtsDir> = A special token that can appear within certain fields. Expands to the value of **SharedExtensionsDir** in the Microsoft Mail key. Used to refer to DLLs that are installed in a shared extensions directory on the network. Valid for the *DLL name*, *command*, and *Help file* subfields.

Custom Messages Entries for Mail

This subkey is similar in many ways to the Custom Commands subkey. Each entry specifies a custom message type to be installed into a Mail menu at run-time.

These entries can appear both in the Microsoft Mail key and in the SHARED32.INI file in the directory defined by the **SharedExtensionsDir** entry under the Microsoft Mail key.

This is the Registry path for this subkey:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \Custom Messages
```

The Custom Messages subkey can contain one or more of the following entries.

class **REG_SZ**

Specifies a string uniquely identifying the message type. Mail places this string in messages and calls custom message DLLs based on its value. Each entry is in the following format:

class = *version;menu;name;position;DLL name;command;operation map; status text;Help file;help context;*

version = The version of Mail that the custom message is compatible with.

menu = The menu where the custom command for the message type is to be added, such as File or Mail.

name = The command name to appear on the menu. Include an ampersand just before the letter that is to serve as an ALT+key accelerator.

position = The zero-relative position within its menu at which the command is to appear. A value of -1 places the command at the end of the menu.

DLL name = Name or path of the DLL in which the custom command resides.

command = Command string passed as one of the parameters to the DLL entry point for the command.

operation map = Sequence of up to 16 digits. Each can be 0, 1, or 2, where 0 indicates that Mail is to perform its standard operation on the custom message; 1 indicates that the DLL is to be called to handle the operation; and 2 indicates that the operation is to be completely disabled. Currently eight operations are defined; the rest are reserved and must be zero. In the following list, 0 is the leftmost digit:

- 0 = Compose (menu command defined in this entry)
- 1 = File.Open
- 2 = Mail.Reply
- 3 = Mail.Reply to All
- 4 = Mail.Forward
- 5 = File.Print
- 6 = File.Save as
- 7 = Arrival of new mail

status text = Text to be displayed in the Mail status bar when the user highlights the command in the menu.

Help file = Windows NT Help file to be invoked when the user presses F1 while the command is selected. The specified filename is passed to the Windows NT Help program. (optional)

help context = Passed to the Windows NT Help program along with the Help filename. Use -1 (Help file index) if there is no specific entry in the Help file for this command. (optional)

<*ExtsDir*> = A special token that can appear within certain fields. Expands to the value of **SharedExtensionsDir** in the Microsoft Mail key. Used to refer to DLLs that are installed in a shared extensions directory on the network. Valid for the *DLL name*, *command*, and *Help file* fields.

Custom Menus Entries for Mail

This subkey can contain any number of entries (within reason). You can use these menus to group custom commands and/or messages under a single, distinctive menu item.

This is the Registry path for this subkey:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \Custom Menus
```

The following information can appear both in the Microsoft Mail key and in SHARED32.INI in the directory defined by **SharedExtensionsDir** under the Microsoft Mail key. The SHARED32.INI file is examined first.

tag **REG_SZ**

Identifies the menu to someone reading these values but serves no other purpose. Specifies a menu name to be added to the Mail menu bar. This is the format:

Tag=version;name;name to follow;status text

For example:

tools=3.0;&Tools;Window;Useful development toys

version = The version of Mail with which the menu is compatible; 3.0 is the current version.

name = The menu name to be added to the menu bar. Include an ampersand just before the letter that is to serve as an ALT+key accelerator.

name to follow = Name of an existing menu. The new menu is added directly before it.

status text = Text to be displayed in the Mail status bar when the user highlights the menu name.

Microsoft Mail Entries

This subkey is used to define the appearance and behavior of the Mail program. This is the Registry path for this subkey:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \Microsoft Mail
```

This key also appears under HKEY_USERS\DEFAULT, but its only contents are **MigrateIni** and **MigrateIniPrint**.

-  Font and Window Size Entries for Microsoft Mail
-  Login and Password Entries for Microsoft Mail
-  Mail Spooler Entries for Microsoft Mail
-  Message File Entries for Microsoft Mail
-  Migration from MAIL.INI for Microsoft Mail
-  Print Entries for Microsoft Mail
-  SharedExtensionsDir Entry in Microsoft Mail
-  Other Entries in Microsoft Mail

Font and Window Size Entries for Microsoft Mail

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Microsoft Mail

FixedFont **REG_SZ**

Range: *facename, size, 0 or 1, 0 or 1*

Default: Courier New, 9, 0, 0

Identifies the fixed-pitch font used to display the body text of a mail message. This entry has four parts, each separated by a comma: typeface name (not the font file name), point size, flag for bold, and flag for italic. The Change Font command on the View menu toggles between the Normal font and the Fixed font.

NormalFont **REG_SZ**

Range: *facename, size, 0 or 1, 0 or 1*

Default: Helv, 10, 0, 0 (The FontSubstitutes subkey defines the mapping of the Helv font to a font present on the local computer.)

Identifies the default font (normally proportionally spaced) used to display Mail messages. This entry has four parts, each separated by a comma: typeface name (not the font file name), point size, flag for bold, and flag for italic. The Change Font command in the View menu toggles between the Normal font and the Fixed font. The latter is useful for viewing messages that were created using a fixed-pitch font. Both entries affect only message body text, not the message envelope text or folder lists; Mail uses Helv 8 for that purpose.

Window **REG_SZ**

Range: *Left Top Right Bottom Zoom Toolbar Statusbar Scrollbars*

Default: window size and location determined by Windows NT, zoom state normal (restored), toolbar on, status bar on, scroll bars on.

Specifies the zero-relative position within its menu where the main Mail window is to appear. This entry consists of eight numbers that govern the display of the main Mail window. This information is written when you exit Mail, and the changes you made while Mail was running are lost.

The format for this entry is as follows:

Left Top Right Bottom = The first four numbers are pixel coordinates for the four sides of the main window in this order: left, top, right, and bottom.

Zoom = The zoom value is:

- 1 = main window is in a normal (restored) state
- 2 = maximized (zoomed)
- 3 = minimized (by icon)

Toolbar Statusbar Scrollbars = Determines when the toolbar, status bar, and scroll bars are displayed on the main window:

- 0 = corresponding bar is not displayed
- 1 = bar is displayed

Login and Password Entries for Microsoft Mail

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Microsoft Mail

Login **REG_SZ**

Range: *mailbox name*

Default: (blank)

Identifies the default User Name (up to 10 characters) displayed in the Mail Sign In dialog box used to log into Mail. If you set both the **Login** and **Password** entries, the Login dialog is not displayed when you start Mail, and your mailbox is immediately displayed. If you set just **Login**, Mail prompts for your password only.

NewMsgsAtStartup **REG_SZ**

Range: *0 or 1*

Default: 0

Specifies whether Mail is to check for new mail messages in the foreground as soon as the user logs in. Set this entry to 1 to have Mail download new messages as quickly as possible when it is started. If this entry is 0, Mail checks for new messages in the background (as is usually the case when the Mail application is being used).

Password **REG_SZ**

Range: *password*

Default: (blank)--you are prompted for a password by the Mail program.

Use this entry and the **Login** entry to provide Mail with your account information, without being required to type this information into the Mail Sign In dialog box each time. If there is no password, leave the value for this entry blank, but do not omit the entry. Omitting the entry means you want to type your password each time in the Mail Sign In dialog box when you start Mail.

Security **REG_SZ**

Range: *0 or 1*

Default: 0

If this entry is 1, Mail prompts for your password whenever its window is restored from its iconic state. That is, if Mail is minimized and you double-click on the icon, you must re-enter your password before you (or anyone else) can see your messages.

ServerPassword **REG_SZ**

Range: *password*

Identifies the password used to connect to the server specified by the **ServerPath** entry. This entry should be used to specify the password for the file share if the form of the **ServerPath** entry is specified using the universal naming conventions (UNC) and the server, share, or the path name contains spaces. If this entry is present, the value for the **ServerPath** entry is interpreted literally, and any spaces present in the value for the entry is used when dynamically connecting to the workgroup postoffice.

If the **ServerPassword** entry is used, do not specify a password for the **ServerPath** entry (the password will be misinterpreted, and the resulting path to the workgroup postoffice will be invalid). The **ServerPassword** entry is written by the Mail program when connecting for the first time to the workgroup postoffice and is stored in the Microsoft Mail key file in encrypted format. Do not change

this entry.

Mail Spooler Entries for Microsoft Mail

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Microsoft Mail

CheckLatencyInterval **REG_SZ**

Range: *seconds*

Default: 30 seconds

Affects the mail spooler's latency checking, which is intended to prevent spooler background processing from interfering with foreground work. If the specified length of time passes without the spooler having any work to do, the latency algorithm is reinitialized.

ForceScanInterval **REG_SZ**

Range: *seconds*

Default: 300 seconds (5 minutes)

Affects the mail spooler's latency checking, which is intended to prevent the spooler background processing from interfering with foreground work. If the designated length of time passes without the spooler getting an opportunity to do outstanding work, idle time is requested more frequently (based on the value of **ScanAgainInterval**), and eventually idle time is used whenever it can.

IdleRequiredInterval **REG_SZ**

Range: *seconds*

Default: 2 seconds

Affects the mail spooler's latency checking, which is intended to prevent the spooler's background processing from interfering with foreground work. The spooler defers its work temporarily if the system has serviced an interactive request such as a keyboard entry or mouse movement within this interval, to avoid starting a transfer when the user is busy.

PollingInterval **REG_SZ**

Range: *minutes*

Default: 10

Gives the default for the Check for New Mail Every *n* Minutes option in the Mail Options dialog box. The value the user enters in the dialog box is written to the user's mail message file (.MMF)--this value is used to define how often the Mail spooler checks for new mail messages.

PumpCycleInterval **REG_SZ**

Range: *seconds*

Default: 60 seconds, or the number of minutes specified in the Mail Options dialog box

Permits the spooler to check for new mail more often than once per minute, or to override the polling interval value defined in the user's mail message file.

ScanAgainInterval **REG_SZ**

Range: *seconds*

Default: 2

Affects the mail spooler latency checking to prevent spooler background processing from interfering unduly with foreground work. When the spooler defers work because of higher priority, interactive tasks, it rechecks the availability of the system at this interval.

SpoolerBackoffInterval REG_SZ

Range: *milliseconds*

Default: 2000 (two seconds)

Specifies the amount of time the mail spooler waits before retrying an operation that has failed because of a transient mail server error condition, such as a locked file.

SpoolerReconnectInterval REG_SZ

Range: *seconds*

Default: 60 (one minute)

Specifies the amount of time the mail spooler waits before retrying an operation that has failed because of a fatal mail server error condition, such as a lost network connection.

Message File Entries for Microsoft Mail

Registry path:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \Mail
 \Microsoft Mail

ExportMmfFile **REG_SZ**

Range: *filename*

Identifies the path and filename for a .MMF file pointing to the last place a mail folder was exported to. This entry is written by the Mail program and is used as a default value for display in the Export Folder dialog box when you choose Export Folders from the File menu in Mail.

LocalIMMF **REG_SZ**

Range: *0 or 1*

Default: 1

Specifies the location where the user's mail message file (.MMF) is created when the user runs Mail for the first time. If this entry is 0, the user's Mail messages are stored in the postoffice on the server. If this entry is 1, the user's .MMF file is created locally in the *SystemRoot* directory rather than on the postoffice the first time Mail is run. Also, this value set to 1, in conjunction with the **NoServerOptions** entry, prevents .MMF files from being stored in the postoffice.

NoServerOptions **REG_SZ**

Range: *0 or 1*

Default: 0

If this entry is 1, the Server button in the Mail Options dialog box is unconditionally disabled. On the Mail server, this button calls up another dialog box that enables the user to relocate the .MMF file. Together with the **LocalIMMF** entry, this entry prevents .MMF files from being stored on the postoffice.

OfflineMessages **REG_SZ**

Range: *Pathname*

Default: The path specified in the Mail Options Server dialog for a local message file.

Defines the file location when you choose to store your message file somewhere other than the postoffice. When you start up without connecting to the postoffice, this entry locates the file quickly (without presenting a File Browse dialog box). The entry is removed when you store your message file at the postoffice.

Note: If you start online and your .MMF file is not on the postoffice, this entry is not used. Use the Mail Server Options dialog box to move your .MMF file.

OldStorePath **REG_SZ**

Range: *Pathname*

Contains the original path to a file that was originally stored in a place other than the postoffice. This entry is written temporarily by the Mail transport while you are moving your message file (using the Mail Options Server dialog box). This entry is removed after the move completes successfully and only appears if the system crashes during a move.

ServerPath **REG_SZ**

Range: *Pathname*

Or *\\server\share\path password*

Or *server/share:path*

If this entry is present, Mail searches for the postoffice in the specified directory. The first form, with a normal path name, works on all networks--the connection to the file share containing the network postoffice must be made before running the Mail program. The second form (UNC) works only on Windows NT and on Microsoft networks and compatibles. If you use the UNC form, Mail connects dynamically (without using a drive) to the file server where the postoffice resides. (If the specified UNC name contains any spaces, the password for the share needs to be specified for the **ServerPassword** entry.) The third form works only on Novell NetWare networks and uses an unused drive letter to connect dynamically to the NetWare file server where the postoffice resides. If no value is specified for the **ServerPath** entry, Mail asks the user for a path to the workgroup postoffice and writes a new value for this entry.

SharedFolders REG_SZ

Range: *0 or 1*

Default: 1

Enables the use of Mail shared folders. If this entry is 1, the user can access shared folders. If this entry is 0, shared folders are unavailable to the user.

Migration from MAIL.INI for Microsoft Mail

Registry path:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \Mail
 \Microsoft Mail

MigrateIni **REG_SZ**

Range: *0 or 1*

Default: 1 (yes)

Specifies whether to migrate the Mail .INI files created by a Windows for MSDOS version of Mail for use under Windows NT. This entry is saved in the HKEY_USERS\DEFAULT\Software\Microsoft\Mail subkey. In HKEY_CURRENT_USER, this entry is deleted after the user first runs Mail.

MigrateIniPrint **REG_SZ**

Range: *0 or 1*

Default: 1 (yes)

Specifies whether to migrate the Mail .INI print information created by a Windows for MS-DOS version of Mail for use under Windows NT. This entry is saved in the HKEY_USERS\DEFAULT\Software\Microsoft\Mail subkey. In HKEY_CURRENT_USER, this entry is deleted after the user first runs Mail.

Print Entries for Microsoft Mail

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Microsoft Mail

Multi-Message **REG_SZ**

Range: *0 or 1*

Default: 1 (That is, the check box is checked by default.)

Defines the last setting chosen for the Print Multiple Notes On A Page check box in the Print dialog box, which appears when you print messages from Mail. Set this entry to 0 for that option to appear unchecked by default.

Printer **REG_SZ**

Range: *printer name, driver name, port*

Default: the default printer specified in Print Manager

This is the printer that appears in the Mail Print dialog box and is used by Mail when printing messages.

SharedExtensionsDir Entry in Microsoft Mail

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Microsoft Mail

SharedExtensionsDir **REG_SZ**

Range: *Pathname*

Or *\\server\share\path password*

Default: (blank)

If this entry is present, Mail searches for shared custom commands and messages in the specified directory. The SHARED32.INI file in that directory identifies the shared extensions to load. Additional extensions can be entered in the Microsoft Mail key. Administrators often provide a common share point for extensions to Mail to simplify updating the extensions, and this entry makes it work. The first form with a normal path name works on all networks. The second form works only on Windows NT and on Microsoft networks and compatibles. If the second form is used, Mail connects dynamically (without using a drive) to the file server where the shared extensions reside.

Other Entries in Microsoft Mail

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Microsoft Mail

DemosEnabled **REG_SZ**

Range: *0 or 1*

Default: 0 (The Mail demos are not provided with Windows NT.)

Specifies whether the Demos menu option is to be displayed in the Help menu. If the value of the entry is 1, the Demos menu option is displayed in the Help menu. If this entry is 0, the Demos menu option is not shown.

GALOnly **REG_SZ**

Range: *0 or 1*

Default: 0

If this entry is set to 1, the Mail address book displays only the Global Address List and the personal address book, thereby providing a flat address list of all the users visible from your postoffice. You must be running against a PC Mail 3.0 or higher postoffice with global address list support for this to work. **GALOnly** is currently supported only by the PC Mail name service provider.

StripGatewayHeaders **REG_SZ**

Range: *0 or 1*

Default: 1 (You don't see gateway information.)

If this entry is 1, message header text that appears above the dashed line is stripped from PC Mail messages that arrive via a gateway. Set this value to 0 if you want to see the extended information supplied by the gateways, which typically includes items such as message identifiers specific to the foreign mail system.

Message header text supplied by native PC Mail clients is always stripped. Only the PC Mail transport supports this entry.

WG **REG_SZ**

Range: 0 or 1

Default: 1

Specifies whether the version of the Mail program running on the computer is the Mail program provided with Windows NT. This entry is used internally by the Mail application and is written by the Mail program when it is executed.

MMF Entries for Mail

Most entries under this key affect automatic compression of the Mail message file, which by default has the filename extension of .MMF. When enabled, automatic compression uses idle time on your PC to recover disk space freed by the deleted messages and returns the disk space to the file system. You should not need to change the default values for entries in this subkey.

Registry path:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \MMF
```

Kb_Free_Start_Compress **REG_SZ**

Range: *kilobytes*

Default: 300

Background compression starts when at least this much recoverable space is detected in your message file. Both **Percent_Free_Start_Compress** and this entry are always active. The first entry to trigger starts the compression.

Kb_Free_Stop_Compress **REG_SZ**

Range: *kilobytes*

Default: 100.

Background compression stops when there is less than the indicated amount of recoverable space in your message file. This avoids the unnecessary difficulty in trying to recover the last little bit of free space. Both this entry and **Percent_Free_Stop_Compress** are always active. The first entry to trigger stops the compression.

No_Compress **REG_SZ**

Range: *0 or 1*

Default: 0 (That is, background compression is enabled.)

Specifies whether background compression is to be disabled. A value of 1 disables background compression of the .MMF message store.

Percent_Free_Start_Compress **REG_SZ**

Range: *percent*

Default: 10

Background compression starts when the amount of recoverable space rises above this percentage of the total file size. Both **Kb_Free_Start_Compress** and this entry are always active. The first one to trigger starts the compression.

Percent_Free_Stop_Compress **REG_SZ**

Range: *percent*

Default: 5

Background compression stops when the amount of recoverable space falls below this percentage of the total .MMF file size. Both this entry and **Kb_Free_Stop_Compress** are always active. The last one to trigger stops the compression.

Secs_Till_Fast_Compress **REG_SZ**

Range: *seconds*

Default: 600 seconds (That is, ten minutes of system inactivity.)

The background compression algorithm has a fast mode and a slow mode. Background compression begins in the slow mode to avoid slowing system response time. After a number of seconds of system inactivity indicated by this entry, the compression switches to fast mode. Any user activity changes the setting back to slow mode.

See also **AppInit_DLLs** in [Windows Software Registration Entries](#)

Mac FileTypes Entries for Mail

Entries in this subkey map the Macintosh file type and creator tags to MSDOS eight-plus-three filenames.

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Mail

 \Mac FileTypes

Mail uses these entry values to determine what application to launch on a file attachment that has been sent from a Macintosh mail client. There are two alternate forms for the entries:

 creator:type=extension

Or:

 :type=extension

Both the creator and type are sequences of four characters (possibly including blanks). For example,

 :TEXT=DOC

launches the application associated with the extension .DOC (Word for Windows, for example) on any Macintosh file of type TEXT.

MS Proofing Tools Entries for Mail

The MS Proofing Tools subkey defines spelling values for Mail.

Registry path:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \MS Proofing Tools
```

CustomDict REG_SZ

Range: *entry name*

Default: (no default)

Specifies the name of an entry in the [MS Proofing Tools] section of the WIN.INI file. That entry in turn gives the fully qualified path to a file containing your custom dictionary. The custom dictionary contains spellings not found in the standard dictionary but that were added using the Add button in the Spelling dialog box. This entry lets Mail take advantage of a custom dictionary you may have already created with another Microsoft application, such as Microsoft Word for Windows.

Spelling REG_SZ

Range: *keyname*

Specifies the name of an entry in the MS Proofing Tools subkey that defines filenames for the spelling checker DLL and dictionary. The entries for **Spelling** are in this format:

Spelling *NNNN,M*

There is no space after the comma. In this format, *NNNN* is the four-digit language identifier of the current Windows NT version as defined in the Control\NLS\Language subkey, and *M* is the spelling dictionary type.

The Registry path for the key that this entry refers to is the following, by default:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \MS Proofing Tools
          \Spelling NNNN,0
```

This subkey contains an entry in the following form:

Spelling *NNNN,0* REG_SZ

Range: *DLLfilename, Dictionary filename*

Default: MSPELL32.DLL,MSP32_XX.LEX (In these values, *XX* is usually the two letters identifying the language version of Windows NT defined in the DosKeybCodes subkey.

Specifies the fully qualified path to the spelling checker DLL and dictionary. This entry lets Mail use the same dictionary you may already be using with another Microsoft application, such as Microsoft Word for Windows. Windows NT does supply a dictionary.

Providers Entries for Mail

Entries in the Providers subkey for Mail define settings that service providers use with Microsoft Mail front-end programs. For this release, there are service providers for Microsoft Mail for PC LANs. Service providers for other mail systems may be available later.

Registry path:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Mail
        \Providers
```

Logon **REG_SZ**

Range: *DLL name*

Default: MSSFS32

Identifies a single DLL that contains the logon and session management code for your mail system.

This value is often, but not necessarily, the same as the **Transport** and **Name** entries. This value is the base name of the DLL, without the .DLL filename extension, but include a path if the DLL is not in a directory on the user's path or in the directory containing the Mail executable file.

Name **REG_SZ**

Range: *DLL name*

Default: MSSFS32 PABNSP32

Identifies one or more DLLs that contain functions required to browse system and personal user lists.

One of the values is often, but not necessarily, the same as the **Logon** and **Transport** entries. Enter the base name of the DLL, without the .DLL filename extension, but include a path if the DLL is not in a directory on the user's path or in the directory containing the Mail executable file.

The order of providers in this entry is significant. When Mail is attempting to resolve ambiguous names typed in a message and finds an exact match in the first provider in the list, it will not go on to query the rest. Placing the personal address book provider first can save time in that process.

SharedFolders **REG_SZ**

Range: *DLL name*

Default: MSSFS32 (It is unlikely that any DLL other than MSSFS will have this functionality.)

Identifies a single DLL that contains functions required to read and write messages in Microsoft PC Mail shared folders.

Transport **REG_SZ**

Range: *DLL name*

Default: MSSFS32

Identifies a single DLL that contains the functions necessary to send and receive mail on your mail system. It is often, but not necessarily, the same as the **Logon** and **Name** entries. Enter the base name of the DLL, without the .DLL filename extension, but include a path if the DLL is not in a directory on the user's path or in the directory containing the Mail executable file.

Registry Entries for Microsoft Schedule+

The settings used by Microsoft Schedule+ to track basic information about the user's schedule, such as display and general option settings, current window positions, and printer information are stored under the following key:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \Schedule+

The Schedule+ key contains the following subkeys:

-  Microsoft Schedule+ Entries
-  Microsoft Schedule+ Appt Books Entries
-  Microsoft Schedule+ Archives Entries
-  Microsoft Schedule+ Exporter Entries
-  Microsoft Schedule+ Importer Entries

Most of these entries have built-in defaults. You should not need to change the Schedule+ settings. To change the appearance and behavior of Schedule+, use the appropriate Schedule+ menu commands. Many values are for saving settings between sessions.

These keys are created in HKEY_CURRENT_USER when you first run Schedule+. If your system previously contained a Windows for MS-DOS version of SCHDPLUS.INI, the contents are migrated to the Registry when you first run Schedule+ under Windows NT.

Microsoft Schedule+ Entries

This key defines the appearance and behavior of Microsoft Schedule+.

Registry path:

HKEY_CURRENT_USER

\Software



\Microsoft

\Schedule+

\Microsoft Schedule+

This key also appears under HKEY_USERS\DEFAULT, but its only contents are **MigrateIni** and **MigrateIniPrint**.

Microsoft Mail entries are defined in the following topics:

-  [Color Entries for Microsoft Schedule+](#)
-  [Defaultxxx Entries for Microsoft Schedule+](#)
-  [Export Entries for Microsoft Schedule+](#)
-  [File and Path Entries for Microsoft Schedule+](#)
-  [Font and Window Entries for Microsoft Schedule+](#)
-  [Import Entries for Microsoft Schedule+](#)
-  [Migration from .INI Files for Microsoft Schedule+](#)
-  [Polling and Update Entries in Microsoft Schedule+](#)
-  [Task List Entries in Microsoft Schedule+](#)
-  [Other Entries in Microsoft Schedule+](#)

Color Entries for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

Color Numbers for Schedule+

ApptBookColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 3 (Yellow)

Specifies the preference setting for the background color of the Appointment Book. The color number corresponds (in order) to the colors shown in the Display dialog box available from the Options menu.

ApptBookLinesColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 1 (Black)

Specifies the preference setting for the color of the lines in the Appointment Book. The color number corresponds to non-dithered colors in the Display dialog box available from the Options menu.

OtherColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 7 (Red)

Specifies the preference setting for the color of other users' appointments in the Planner. The color number corresponds to non-dithered colors in the Display dialog box available from the Options menu.

PageBackgroundColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 11 (Gray)

Specifies the preference setting for the background color of the Schedule+ window. The color number corresponds to non-dithered colors in the Display dialog box available from the Options menu.

PlannerColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 2 (White)

Specifies the preference setting for the background color of the Planner window. The color number corresponds to colors in the Display dialog box available from the Options menu.

PlannerLinesColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 1 (Black)

Specifies the preference setting for the color of the lines in the Planner. The color number corresponds to non-dithered colors in the Display dialog box available from the Options menu.

UserColor **REG_SZ**

Range: *colornumber(1-17)*

Default: 4 (Blue)

Specifies the preference setting for the color of your own appointments in the Planner. The color number corresponds to non-dithered colors in the Display dialog box available from the Options menu.

Defaultxxx Entries for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

DefaultPrinter **REG_SZ**

Range: *printer name, driver name, port*

Indicates the current default printer port and its network path as specified in Print Manager. This is the default printer Schedule+ uses for printing schedule information.

DefaultRemindAgain **REG_SZ**

Range: *0 or 1*

Default: 0

Defines the default state of the Remind Again check box. If this entry is 1, you are reminded again of your appointments at the requested intervals. If this entry is 0, you are reminded of your appointment only once.

DefaultRemindAgainAmount **REG_SZ**

Range: *timeunits*

Default: 5

Specifies the default number of time units to wait (interval) before reminding you of appointments again.

DefaultRemindAgainUnits **REG_SZ**

Range: *minutes, hours, days, weeks, or months*

Default: minutes

Specifies the type of time units used in the **DefaultRemindAgainAmount** entry.

Export Entries for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

ExportNoNotes **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates whether the user chose to export notes. If this entry is 0, notes are exported. If this entry is 1, the notes are not exported.

ExportRange **REG_SZ**

Range: *0 or 1*

Indicates the range of schedule information to be exported. If this entry is 0, the entire schedule file is exported. If this entry is 1, a particular range is exported.

ExportType **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates the current default file type for exporting your schedule. If this entry is 0, the default file type for exporting your schedule is the Schedule+ format. If the entry is 1, the file type for export is Text.

File and Path Entries for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

AppPath **REG_SZ**

Range: *Pathname*

Default: *SystemRoot\SYSTEM32* directory

Specifies the location of the Schedule+ program and execution files. Microsoft Mail uses this path to find Schedule+ when you receive a meeting request.

CreateFileFirstTime **REG_SZ**

Range: *0 or 1*

Default: 0

Specifies whether an online calendar (.CAL) file should be created for a first-time Schedule+ user. If this entry is 1, an online calendar (.CAL) file is created the first time a user signs on to Schedule+. If 0 (as set automatically the first time you run Schedule+), an online calendar file is not created automatically.

LocalPath **REG_SZ**

Range: *Pathname*

Specifies the location of the last user's local calendar (.CAL) file.

Font and Window Entries for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

AppointmentView **REG_SZ**

Range: *state left top right bottom*

Specifies the state (1=normal, 2=maximized, 3=iconic) and the coordinates for the position of the Appointment Book window on the screen. These five numbers are written by the Schedule+ application when you exit and are used to restore the window to the last displayed position. The coordinates are pixel coordinates for the four sides of the Appointment Book window.

LargeFont **REG_SZ**

Range: *0 or 1*

Default: 0

Specifies the preference setting for the font size of text displayed in the Appointment Book and Planner. If this entry is 1, the font size of text is 10 points. If this entry is 0, the font is 8 points.

MainWindow **REG_SZ**

Range: *state left top right bottom*

Specifies the state (1=normal, 2=maximized, 3=iconic) and the coordinates for the position of the Schedule+ application window on the screen. These five numbers are written by the Schedule+ application when you exit, and are used to restore the Schedule+ window to the last displayed position. The coordinates are pixel coordinates for the four sides of the main window.

NoStatusBar **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates the preference setting for displaying the status bar. If this entry is 1, status bar is not displayed. If set to 0, the status bar is displayed.

RequestSummary **REG_SZ**

Range: *state left top right bottom*

Specifies the state (1=normal, 2=maximized, 3=iconic) and the coordinates for the position of the Messages window on the screen. These five numbers are written by the Schedule+ application when you exit and are used to restore the Messages window to the last displayed position. The coordinates are pixel coordinates for the four sides of the Messages window.

WindowOrder **REG_SZ**

Range: *0 1 or 1 0*

Default: 0 1

Indicates the current display order of Schedule+ windows. The Schedule+ window is represented by 0, and the Messages window is 1. The first value for the **WindowOrder** entry indicates the window on top, and the second entry identifies the window behind the top window.

Import Entries for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

ImportDoNotAddDuplicates **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates whether the user chose to import duplicate appointments. If this entry is 0, duplicate appointments are imported. If this entry is 1, your duplicate appointments are not imported.

ImportDoNotAskAboutConflicts **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates whether the user chose to be asked about conflicting appointments during the import process. If this entry is 0, you are prompted for each conflicting appointment during the import process--in this case, you are asked whether to add each conflicting appointment. A value of 1 indicates that you are not asked about conflicts; they are added automatically.

ImportType **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates the current default file type for importing a schedule file. If this entry is 0, the file type for importing your schedule is the Schedule+ format. If the entry is 1, the file type is the Windows NT Calendar format.

Migration from .INI Files for Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

MigrateIni **REG_SZ**

Range: 0 or 1

Default: 1 (yes)

Specifies whether to migrate the Schedule+ .INI files created by a Windows for MS-DOS version of Schedule+ for use under Windows NT. This entry is saved in this subkey:

 HKEY_USERS\DEFAULT\Software\Microsoft\Mail

In HKEY_CURRENT_USER, this entry is deleted after the user first runs Mail.

MigrateIniPrint **REG_SZ**

Range: 0 or 1

Default: 1 (yes)

Specifies whether to migrate the Schedule+ .INI print information created by a Windows for MS-DOS version of Schedule+ for use under Windows NT. This entry is saved in this subkey:

 HKEY_USERS\DEFAULT\Software\Microsoft\Mail

In HKEY_CURRENT_USER, this entry is deleted after the user first runs Mail.

Polling and Update Entries in Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

PollTime **REG_SZ**

Range: *centiseconds*

Default: 6000 centiseconds (one minute)

 Specifies the frequency for checking the server for schedule file changes.

ReminderPollTime **REG_SZ**

Range: *minutes*

Default: 15

 Specifies the frequency for polling the server for alarm changes.

UpdatePostOfficeTime **REG_SZ**

Range: *centiseconds*

Default: 6000 centiseconds (one minute)

 Specifies the frequency for updating the postoffice on the server after a change is made.

Task List Entries in Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

ShowActiveTasks **REG_SZ**

Range: *0 or 1*

Default: 0 (That is, all tasks are displayed.)

Indicates whether the Task list is showing all tasks or only active tasks, as specified from the Tasks menu. If only active tasks are displayed, this value is 1.

TaskSortOrder **REG_SZ**

Range: *0, 1, 2, -1, -2, or -3*

Default: 0

Specifies the current sort order for tasks, according to the following:

0 = Tasks are sorted by priority.

1 = Tasks are sorted by due date.

2 = Tasks are sorted by description.

-1 = Tasks are sorted by reverse description.

-2 = Tasks are sorted by reverse due date.

-3 = Tasks are sorted by reverse priority.

TaskSortSecond **REG_SZ**

Range: *0, 1, 2, -1, -2, or -3*

Default: 0

Specifies the secondary sort order for tasks. If this entry is 0, the second sort order is by priority; if 1 the second sort order, using the same values as specified for **TaskSortOrder**.

ViewNotByProject **REG_SZ**

Range: *0 or 1*

Default: 0

Indicates whether the tasks in the Task list are currently displayed by project. If this entry is 1, the tasks are not displayed by project.

Other Entries in Microsoft Schedule+

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+

CopyTime **REG_SZ**

Range: *minutes*

Default: 15 minutes

Specifies the time interval that Schedule+ copies your online .CAL file to your local .CAL file (occurs in idle time).

LocalUser **REG_SZ**

Range: *username*

Specifies the name of the last user to use the Schedule+ software on this computer.

DemosEnabled **REG_SZ**

Range: *0 or 1*

Default: 0 (The Schedule+ demos are not provided with Windows NT.)

Specifies whether the Demos menu option is to be displayed in the Help menu. If the entry is 1, the Demos menu option is displayed in Help menu. If this entry is 0, the Demos menu option is not shown.

StartupOffline **REG_SZ**

Range: *0 or 1*

Default: 0 (That is, Schedule+ is started online.)

Specifies whether Schedule+ should start up using the offline scheduling information, or whether the online schedule should be used. If this entry is 1, Schedule+ is started offline.

Color Numbers for Schedule+

1=Black	10=Dark gray
2=White	11=Light gray
3=Yellow	12=Bright blue
4=Blue	13=Bright green
5=Green	14=Bright blue-green
6=Blue-green	15=Bright red
7=Red	16=Bright violet
8=Violet	17= Bright yellow
9=Khaki	

Microsoft Schedule+ Appt Books Entries

Schedule+ uses this subkey to track the Appointment books of other Schedule+ users that you had open when you exited Schedule+.

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+ Appt Books

Count **REG_SZ**

Range: *number*

Indicates the number of other users' Appointment Books you had open when you exited Schedule+. More entries appear in this subkey when the number is nonzero.

Microsoft Schedule+ Archives Entries

Schedule+ uses this subkey to track the Archive files that you had open when you exited Schedule+.

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+ Archives

Count **REG_SZ**

Range: *number*

Indicates the number of Archive files you had open when you exited Schedule+. More entries appear in this subkey when the number is nonzero.

Microsoft Schedule+ Exporter Entries

Schedule+ uses this subkey to specify settings for exporters.

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+ Exporters

Key **REG_SZ**

Range: *DLL name*

Identifies a single exporter DLL for Schedule+. The available files can be found on CompuServe. The Key name of this entry can be any string.

Microsoft Schedule+ Importer Entries

Schedule+ uses this subkey to specify settings for importers.

Registry path:

HKEY_CURRENT_USER

 \Software

 \Microsoft

 \Schedule+

 \Microsoft Schedule+ Importers

Key **REG_SZ**

Range: *DLL name*

Identifies a single importer DLL for Schedule+. The available files can be found on CompuServe. The Key name of this entry can be any string.

Registry Entries for User Preferences

Information about Registry entries for user preferences about the following topics can be found in this section:

-  [Hive Information for User Profiles](#)
-  [Console Entries for Users](#)
-  [Color, Pattern, and Screen Saver Entries for Users](#)
-  [Cursors Entry Values for Users](#)
-  [Desktop Entry Values for Users](#)
-  [Environment Variable Entries for Users](#)
-  [Extensions Entries for Users](#)
-  [File Manager Software Settings](#)
-  [International Entry Values for Users](#)
-  [Keyboard and Keyboard Layout Entries for Users](#)
-  [Mouse Entries for Users](#)
-  [Multimedia and Sound Entries for Users](#)
-  [Network Entries for Users](#)
-  [Program Manager Entries for Users](#)
-  [Windows Help Entries](#)

The information presented here is primarily for troubleshooting, showing the default entry values and explaining the meaning of important entries.

There are no hidden values that you can set for user preferences. That is, all of these values can be set using the icons in Control Panel or the tools in the Administrative Tools group, or other programs provided with Windows NT.

However, some entry values do not appear in the Registry unless you use Control Panel or other tools to change the default values.

All Registry paths shown here are for HKEY_CURRENT_USER, to show how you can view entries for the currently logged on user. However, most of these entries also appear in HKEY_USERS\DEFAULT, where changing entries will change values for the default user profile.

Hive Information for User Profiles

Information about user profile files appears in the following keys:



The HiveList subkey lists all hives that are active but not profiles that are not active.



The ProfileList subkey lists all the profiles known on the computer, whether or not the profiles are active, under the following Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \ProfileList
          \SID_#
```

Each installed user profile has its own subkey under the ProfileList subkey, and that subkey contains the following entry:

ProfileImagePath **REG_EXPAND_SZ**

Range: *Profile hive filename*

Default: %SystemRoot%\system32\config\hiveFilename

Specifies the path and filename for the hive for this user. The hive file name that is the value for **ProfileImagePath** includes a portion of the username associated with that *SID_#*, so that you can identify the user to which it belongs.

Sid **REG_BINARY**

Range: *Number assigned by system*

Console Entries for Users

The Console key contains several subkeys that define screen size and buffer size for character-based screens in Windows NT. These subkeys appear under the following Registry path:

HKEY_CURRENT_USER

\Console

\subkeyNames

The Command Prompt subkey does not appear unless the current user has changed the screen colors or font for the command prompt and also checked the Save Configuration options. Use the commands on the Control menu in the command prompt to change these values.

Command Prompt (All data types are REG_DWORD):

FontFamily=0x30

FontSize=0xc0008

FullScreen=0x1

PopupColors=0xf5

QuickEdit=0

ScreenBufferSize=0x190050

ScreenColors=0x9f

WindowsPosition=0x150004

WindowSize=0x190050

Configuration (All data types are REG_SZ):

CommandRecallBufferSize=50

FillAttr=0x07

ScreenBufferColumns=80

ScreenBufferRows=25

WindowColumns=80

WindowRows=25

Introducing Windows NT:

FullScreen=0x1

Microsoft QBASIC:

FullScreen=0x1

Color, Pattern, and Screen Saver Entries for Users

This section describes the subkeys that contain settings for user preferences related to the desktop.



Colors Entry Values



Color Schemes Entry Values



Patterns Entry Values



Screen Saver Subkey Entry Values

Colors Entry Values

The Colors subkey specifies the color as a series of three numbers for each area of the Windows screen, in the following Registry path:

```
HKEY_CURRENT_USER  
  \Control Panel  
    \Colors
```

Each entry has a REG_SZ data type. The following lists the defaults for each entry under the Colors subkey:

ActiveBorder=192 192 192

ActiveTitle=0 0 128

AppWorkSpace=255 255 255

Background=255 255 255

ButtonFace=192 192 192

ButtonHighlight=255 255 255

ButtonShadow=128 128 128

ButtonText=0 0 0

GrayText=128 128 128

Hilight=0 0 128

HilightText=255 255 255

InactiveBorder=192 192 192

InactiveTitle=192 192 192

InactiveTitleText=0 0 0

Menu=255 255 255

MenuText=0 0 0

Scrollbar=192 192 192

TitleText=255 255 255

Window=255 255 255

WindowFrame=0 0 0

WindowText=0 0 0

Color Schemes Entry Values

The entries in the Color Schemes subkey define the colors for each element of specific color schemes, as set by choosing the Color icon in Control Panel. These entries appear under the following Registry path:

```
HKEY_CURRENT_USER
    \Control Panel
        \Color Schemes
```

The Current subkey specifies the current color scheme, based on those listed in the Color Schemes subkey.

The Custom Colors subkey defines the custom colors in the color palette, as set by choosing the Color icon in Control Panel. The entries are designated ColorA through ColorP, and all have the value FFFFFFFF by default.

Each entry in these subkeys has a REG_SZ data type.

The following list shows the defaults for each color scheme.

Arizona=

804000,FFFFFF,FFFFFF,0,FFFFFF,0,808040,C0C0C0,FFFFFF,4080FF,C0C0C0,0,C0C0C0,C0C0C0,808080,0,808080,808000,FFFFFF,0,FFFFFF

Black Leather Jacket=

0,C0C0C0,FFFFFF,0,C0C0C0,0,800040,808080,FFFFFF,808080,808080,0,10E0E0E0,C0C0C0,808080,0,808080,0,FFFFFF,0,FFFFFF

Bordeaux=

400080,C0C0C0,FFFFFF,0,FFFFFF,0,800080,C0C0C0,FFFFFF,FF0080,C0C0C0,0,C0C0C0,C0C0C0,808080,0,808080,800080,FFFFFF,0,FFFFFF

Cinnamon=

404080,C0C0C0,FFFFFF,0,FFFFFF,0,80,C0C0C0,FFFFFF,80,C0C0C0,0,C0C0C0,C0C0C0,808080,0,808080,80,FFFFFF,0,FFFFFF

Designer=

7C7C3F,C0C0C0,FFFFFF,0,FFFFFF,0,808000,C0C0C0,FFFFFF,C0C0C0,C0C0C0,0,C0C0C0,C0C0C0,808080,0,C0C0C0,808000,0,0,FFFFFF

Emerald City=

404000,C0C0C0,FFFFFF,0,C0C0C0,0,408000,808040,FFFFFF,408000,808040,0,C0C0C0,C0C0C0,808080,0,808080,8000,FFFFFF,0,FFFFFF

Fluorescent=

0,FFFFFF,FFFFFF,0,FF00,0,FF00FF,C0C0C0,0,FF80,C0C0C0,0,C0C0C0,C0C0C0,808080,0,808080,0,FFFFFF,0,FFFFFF

Hotdog Stand=

FFFF,FFFF,FF,FFFFFF,FFFFFF,0,0,FF,FFFFFF,FF,FF,0,C0C0C0,C0C0C0,808080,0,808080,0,FFFFFF,FFFFFF,FFFFFF

LCD Default Screen Settings=

808080,C0C0C0,C0C0C0,0,C0C0C0,0,800000,C0C0C0,FFFFFF,800000,C0C0C0,0,C0C0C0,C0C0C0,7F8080,0,808080,800000,FFFFFF,0,FFFFFF

LCD Reversed - Dark=

0,80,80,FFFFFF,8080,0,8080,800000,0,8080,800000,0,8080,C0C0C0,7F8080,0,C0C0C0,800000,FFFF,828282,FFFFFF

LCD Reversed - Light=

800000,FFFFFF,FFFFFF,0,FFFFFF,0,808040,FFFFFF,0,C0C0C0,C0C0C0,800000,C0C0C0,C0C0C0,7F8080,0,808040,800000,FFFFFF,0,FFFFFF

Mahogany=

404040,C0C0C0,FFFFFF,0,FFFFFF,0,40,C0C0C0,FFFFFF,C0C0C0,C0C0C0,0,C0C0C0,C0C0C0,8080

80,0,C0C0C0,80,FFFFFF,0,FFFFFF

Monochrome=

C0C0C0,FFFFFF,FFFFFF,0,FFFFFF,0,0,C0C0C0,FFFFFF,C0C0C0,C0C0C0,0,808080,C0C0C0,808080,0,808080,0,FFFFFF,0,FFFFFF

Ocean=

808000,408000,FFFFFF,0,FFFFFF,0,804000,C0C0C0,FFFFFF,C0C0C0,C0C0C0,0,C0C0C0,C0C0C0,808080,0,0,808000,0,0,FFFFFF

Pastel=

C0FF82,80FFFF,FFFFFF,0,FFFFFF,0,FFFF80,FFFFFF,0,C080FF,FFFFFF,808080,C0C0C0,C0C0C0,808080,0,C0C0C0,FFFF00,0,0,FFFFFF

Patchwork=

9544BB,C1FBFA,FFFFFF,0,FFFFFF,0,FFFF80,FFFFFF,0,64B14E,FFFFFF,0,C0C0C0,C0C0C0,808080,0,808080,FFFF00,0,0,FFFFFF

Plasma Power Saver=

0,FF0000,0,FFFFFF,FF00FF,0,800000,C0C0C0,0,80,FFFFFF,C0C0C0,FF0000,C0C0C0,808080,0,C0C0C0,FFFFFF,0,0,FFFFFF

Rugby=

C0C0C0,80FFFF,FFFFFF,0,FFFFFF,0,800000,FFFFFF,FFFFFF,80,FFFFFF,0,C0C0C0,C0C0C0,808080,0,808080,800000,FFFFFF,0,FFFFFF

The Blues=

804000,C0C0C0,FFFFFF,0,FFFFFF,0,800000,C0C0C0,FFFFFF,C0C0C0,C0C0C0,0,C0C0C0,C0C0C0,808080,0,C0C0C0,800000,FFFFFF,0,FFFFFF

Tweed=

6A619E,C0C0C0,FFFFFF,0,FFFFFF,0,408080,C0C0C0,FFFFFF,404080,C0C0C0,0,10E0E0E0,C0C0C0,0,808080,0,C0C0C0,8080,0,0,FFFFFF

Valentine=

C080FF,FFFFFF,FFFFFF,0,FFFFFF,0,8000FF,400080,FFFFFF,C080FF,C080FF,0,C0C0C0,C0C0C0,808080,0,808080,FF00FF,0,FFFFFF,FFFFFF

Wingtips=

408080,C0C0C0,FFFFFF,0,FFFFFF,0,808080,FFFFFF,FFFFFF,4080,FFFFFF,0,808080,C0C0C0,808080,0,C0C0C0,808080,FFFFFF,0,FFFFFF

Patterns Entry Values

HKEY_CURRENT_USER

\Control Panel

\Patterns

The Patterns subkey contains entries that define the color values for the bitmap patterns, as set by choosing the Desktop icon. Each value is a set of eight numbers, corresponding to the colors in the eight basic elements of the pattern.

Each entry has a REG_SZ data type.

The following list shows the default colors for the default patterns.

(None)=(None)

50% Gray=170 85 170 85 170 85 170 85

Boxes=127 65 65 65 65 65 127 0

Critters=0 80 114 32 0 5 39 2

Diamonds=32 80 136 80 32 0 0 0

Paisley=2 7 7 2 32 80 80 32

Pattern=224 128 142 136 234 10 14 0

Quilt=130 68 40 17 40 68 130 1

Scottie=64 192 200 120 120 72 0 0

Spinner=20 12 200 121 158 19 48 40

Thatches=248 116 34 71 143 23 34 113

Tulip=0 0 84 124 124 56 146 124

Waffle=0 0 0 0 128 128 128 240

Weave=136 84 34 69 136 21 34 81

Screen Saver Subkey Entry Values

HKEY_CURRENT_USER

\Control Panel

\ScreenSaverName

The various *Screen Saver* subkeys define user preferences for specific screen savers. All entries have a REG_SZ data type. The following table summarizes the default entries under the *Screen Saver* subkeys.

Screen Saver.Marquee

BackgroundColor=0 0 128

CharSet=0

Font=Times New Roman

Mode=1

Size=24

Speed=14

Text=Your text goes here.

TextColor=255 0 255

Screen Saver.Mystify

Active1=1

Active2=1

Clear Screen=1

EndColor1=255 255 255

EndColor2=255 255 255

Lines1=7

Lines2=12

StartColor1=0 0 0

StartColor2=0 0 0

WalkRandom1=1

WalkRandom2=1

Screen Saver.Stars

Density=50

WarpSpeed=10

See Also:

[Screen Saver Settings for the Desktop](#)

Cursors Entry Values for Users

HKEY_CURRENT_USER

\Control Panel

\Cursor

The Cursor subkey contains entries that specify the .ANI or .CUR files containing custom cursors defined using the Cursor icon in Control Panel. There are no entries in this key unless the user changes cursor styles in Control Panel. All data types are REG_SZ. The following lists the names for possible default entries:

AppStarting

Arrow

CrossHair

IBeam

No

SizeAll

SizeNESW

SizeNS

SizeNWSE

SizeWE

Wait

Desktop Entry Values for Users

The Desktop key contains entries that control the appearance of the screen background and the position of windows and icons on the screen.





Registry path:

HKEY_CURRENT_USER

\Control Panel

\Desktop

To change most of these entries, use the Desktop icon in Control Panel. The Desktop subkey can contain the following kinds of entries:

-  Icon Settings for the Desktop
-  Screen Saver Settings for the Desktop
-  Wallpaper and Background Settings for the Desktop
-  Other Settings for the Desktop

Other Settings for the Desktop

Registry path:

HKEY_CURRENT_USER

\Control Panel

\Desktop

BorderWidth **REG_SZ**

Range: *number*

Default: 3

Sets the width of the borders around all the windows that have sizeable borders. The possible range is 1 (narrowest) to 49 (widest).

CoolSwitch **REG_SZ**

Range: *Boolean*

Default: 1

Turns fast task switching on or off. To change this entry, choose the Desktop icon from Control Panel, and check or clear the Fast ALT+TAB Switching option in the Task List dialog box.

CursorBlinkRate **REG_SZ**

Range: *milliseconds*

Default: 530

Indicates how much time elapses between each blink of the selection cursor.

GridGranularity **REG_SZ**

Range: *number*

Default: 0

Specifies the size of the grid used to position windows on the screen. The possible range is 0 through 49, in units of 8 pixels.

Wallpaper and Background Settings for the Desktop

Registry path:

HKEY_CURRENT_USER

\Control Panel

\Desktop

Pattern **REG_SZ**

Range: *b1 b2 b3 b4 b5 b6 b7 b8*

Default: (None) (This string appears when no pattern is specified.)

Specifies a pattern for the screen background. The 8 numeric values define a bitmap 8 pixels wide and 8 pixels high. Each decimal value represents a byte, and each byte represents a row of 8 pixels, where 0 sets the corresponding pixel to the background color, and 1 sets the corresponding pixel to the foreground color (specified by the **Background** and **WindowText** values in the Colors subkey, respectively).

For example, if you set the *b1* value to the decimal value 175, the top row of pixels in the bitmap appears as the binary equivalent (10101111).

TileWallpaper **REG_SZ**

Range: *Boolean*

Default: 0

Specifies that the desktop wallpaper is tiled across the screen if this value is 1, or centered if this value is 0.

Wallpaper **REG_SZ**

Range: *bitmap-filename*

Default: "(None) (This string appears when no pattern is specified.)

Supplies the filename for the bitmap on the screen background. Include the path if the file is not in the *SystemRoot* or *SystemRoot\SYSTEM32* directory.

Screen Saver Settings for the Desktop

Registry path:

HKEY_CURRENT_USER

\Control Panel

\Desktop

ScreenSaveActive **REG_SZ**

Range: *Boolean*

Default: 0

Specifies whether a screen saver should be displayed if the system is not actively being used. Set this value to 1 to use a screen saver; 0 turns off the screen saver.

ScreenSaverIsSecure **REG_SZ**

Range: *Boolean*

Default: 0

Specifies whether a password is assigned to the screen saver.

ScreenSaveTimeout **REG_SZ**

Range: *seconds*

Default: 900

Specifies the amount of time that the system must be idle before the screen saver appears.

SCRNSAVE.EXE **REG_SZ**

Range: *Filename*

Default: (None)

Specifies the screensaver executable filename.

Icon Settings for the Desktop

Registry path:

HKEY_CURRENT_USER

\Control Panel

\Desktop

IconSpacing **REG_SZ**

Range: *pixels*

Default: 75

Specifies the number of pixels that appear horizontally between icons. A larger number increases the space between icons.

IconTitleFaceName **REG_SZ**

Range: *fontname*

Default: Helv

Specifies the font used to display icon titles. Change this value if the icon title is difficult to read.

IconTitleSize **REG_SZ**

Range: *number*

Default: 9

Specifies the size of the font used to display icon titles. Change this value if the icon title is difficult to read.

IconTitleStyle **REG_SZ**

Range: *Boolean*

Default: 0

IconTitleWrap **REG_SZ**

Range: *Boolean*

Default: 1

Specifies whether to wrap icon titles. A value of 1 allows icon title wrapping and increases icon vertical spacing by three lines; 0 turns off icon title wrapping.

Environment Variable Entries for Users

The Environment subkey contains the user environment variables, as defined by choosing the System icon in Control Panel. Changes to these variables take effect the next time a non-Windows NT-based application is run or the command prompt is used.

Registry path:

HKEY_CURRENT_USER
 \Environment

The default is the environment variables defined in the user's profile at startup.

Extensions Entries for Users

The Extensions subkey identifies personal preferences for document files with corresponding command lines, so that opening a document file in File Manager automatically starts the application. The extensions are found in the following Registry path:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Windows NT
        \CurrentVersion
          \Extensions
```

The following default entries are defined. All have a REG_SZ data type.

bmp=pbrush.exe ^.bmp

crd=cardfile.exe ^.crd

ini=notepad.exe ^.ini

pcx=pbrush.exe ^.pcx

rec=recorder.exe ^.rec

trm=terminal.exe ^.trm

txt=notepad.exe ^.txt

wri=write.exe ^.wri

Note: The extension information for all users can be viewed and modified in HKEY_CLASSES_ROOT. This is where you will find the filetypes and extension information for File Manager.




File Manager Software Settings

The File Manager subkey contains the user preferences for the appearance of items in File Manager.

Registry path:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \File Manager
 \Settings

The following entries can appear. Most items have a default setting and do not appear unless the user makes changes in File Manager.

	<u>Confirmxxx Settings for File Manager</u>
	<u>Typeface Settings for File Manager</u>
	<u>Other Settings for File Manager</u>

Confirmxxx Settings for File Manager

Registry path:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \File Manager
 \Settings

ConfirmDelete **REG_SZ**

Range: *Boolean*

Default: 1 (enabled)

Specifies whether the user is to be prompted to confirm file deletion requests.

ConfirmFormat **REG_SZ**

Range: *Boolean*

Default: 1 (enabled)

Specifies whether the user is to be prompted to confirm formatting requests.

ConfirmMouse **REG_SZ**

Range: *Boolean*

Default: 1 (enabled)

Specifies whether the user is to be prompted to confirm mouse drag-and-drop requests.

ConfirmReplace **REG_SZ**

Range: *Boolean*

Default: 1 (enabled)

Specifies whether the user is to be prompted to confirm file replacement requests.

ConfirmSubDel **REG_SZ**

Range: *Boolean*

Default: 1 (enabled)

Specifies whether the user is to be prompted to confirm subdirectory deletion requests.

ConfirmSystemHiddenReadOnly **REG_SZ**

Range: *Boolean*

Default: 1 (enabled)

Specifies whether the user is to be prompted to confirm for system, hidden, or read-only file changes.

Other Settings for File Manager

Registry path:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \File Manager
 \Settings

AddOns **REG_SZ**

Range: *Boolean*

Default: (none)

dir1 **REG_SZ**

Range: *Comma-separated list*

Default: 0,0,522,249,-1,-1,1,0,202,2033,261,C:\WINNT*.*

The current directory settings.

NumButtons **REG_SZ**

Range: *Number*

Default: 15000000

ToolbarWindow **REG_SZ**

Default:

CD000000CE000000FFFFFFFFE000000FF000000FFFFFFFFF9101000092010000FFFFFFFF9401000
0950100009601000097010000FFFFFFFF52040000FFFFFFFF6B0000006A0000006C000000FFFFFFFF
F5D020000

Contains user-defined settings for the toolbar, as defined in the Options menu in File Manager.

Window **REG_SZ**

Range: *Numbers*

Default: 0,0,640,480, , ,2

Specifies the size and position of the window and whether it is maximized when opened. Use the mouse to move and size the window.

Typeface Settings for File Manager

Registry path:

HKEY_CURRENT_USER
 \Software
 \Microsoft
 \File Manager
 \Settings

Face **REG_SZ**

Range: *Typeface*

Default: MS Sans Serif.

Specifies the name of the typeface used for desktop items.

FaceWeight **REG_SZ**

Range: *Number*

Specifies 700 for bold or bold italic, 400 for regular or italic.

LowerCase **REG_DWORD**

Range: *0, 1, 4, 8*

Specifies values for lowercase variables checked in the Fonts dialog box, as follows:

0 = No options checked

0x1 = Lowercase for FAT drives

0x4 = Italic

0x8 = Lowercase for all drives

Size **REG_SZ**

Range: *Number*

Default: 8

The point size for the typeface.

International Entry Values for Users

The International subkey describes how to display dates, times, currency, and other items for a specific country, under the following Registry path:

HKEY_CURRENT_USER
 \Control Panel
 \International

The International\Sorting Order key is not used in this version of Windows NT.

The following summarizes entry values under this subkey. All data types are REG_SZ. To change any of these items, choose the International icon in Control Panel.

iCountry **REG_SZ**

Range: *country*

Specifies the country code. This number matches the country's international telephone code, except for Canada, which is 2. The U.S. English default is 1.

iCurrDigits **REG_SZ**

Range: *number*

Specifies the number of digits to put after the decimal separator in currency. The U.S. English default is 2.

iCurrency **REG_SZ**

Range: *number*

Specifies a positive currency format, where 0 = \$2, 1 = 2\$, 2 = \$ 2, and 3 = 2 \$. The U.S. English default is 0. The actual currency symbol is specified by the **sCurrency** value.

iDate **REG_SZ**

Range: *number*

Specifies a numerical date format for compatibility with Windows 2.x, where 0 = 12/31/90, 1 = 31/12/90, and 2 = 90/12/31. The U.S. English default is 0. The actual date divider is specified by the **sShortDate** value.

iDigits **REG_SZ**

Range: *number*

Specifies the number of digits to display after the decimal separator in numbers. The U.S. English default is 2.

iLZero **REG_SZ**

Range: *0 or 1*

Specifies whether to put leading zeros in decimal numbers, where 0 = .7 and 1 = 0.7. The U.S. English default is 1. The actual decimal separator is specified by the **sDecimal** value.

iMeasure **REG_SZ**

Range: *0 or 1*

Specifies the measurement system as metric or English, where 0 = metric and 1 = English. The U.S. English default is 1.

iNegCurr **REG_SZ**

Range: *number*

Specifies a negative number format.

iNegCurr -- Negative Number Format

The U.S. English default is 1. The actual currency symbol is specified by the **sCurrency** value.

iTime REG_SZ

Range: *number*

Specifies whether to format time using a 12-hour or 24-hour clock, where 0 = 1:00 (12-hour clock) and 1 = 13:00 (24-hour clock). The U.S. English default is 0. The actual time separator is specified by the **sTime** value.

iTLZero REG_SZ

Range: *number*

Specifies whether to put leading zeros in time, where 0 = 9:15 and 1 = 09:15. The U.S. English default is 0. The actual time separator is specified by the **sTime** value.

Locale REG_SZ

Range: *number*

Specifies the current user's locale ID for the local language preferences, based on values defined in CurrentControlSet\Control\Nls\Language. The U.S. English default is 00000409.

s1159 REG_SZ

Range: *string*

Specifies the time marker to use in time strings before noon in the 12-hour time format. The U.S. English default is AM.

s2359 REG_SZ

Range: *string*

Specifies the time marker to use in time strings after noon in the 12-hour format or that follows all times in the 24-hour format. The U.S. English default is PM.

sCountry REG_SZ

Range: *string*

Specifies the name of the country whose standard value you want to use. The U.S. English default is United States.

sCurrency REG_SZ

Range: *string*

Specifies the currency symbol you want to use. The U.S. English default is \$.

sDate REG_SZ

Range: *string*

Specifies the symbol separating numbers for the short date. The U.S. English default is /.

sDecimal REG_SZ

Range: *string*

Specifies the punctuation used to separate the fractional part of a decimal number from the whole number part. The U.S. English default is . (a period).

sLanguage REG_SZ

Range: *string*

Specifies the language you want to work in. Windows applications that provide language specific tasks, such as sorting or spell checking, use this entry. The U.S. English default is enu.

Values for Locales

sList REG_SZ

Range: *string*

Specifies the character used to separate items in a list. In U.S. English, the most common separator is a comma. The U.S. English default is , (comma).

sLongDate REG_SZ

Range: *format*

Specifies your choices for the long date formats, including abbreviations for the words and separators. Control Panel accepts only certain format combinations. Therefore, you should use Control Panel to change these entries. The U.S. English default is dddd, MMMM dd, yyyy (that is, Friday, June 1, 1990).

sLongDate Values

sShortDate REG_SZ

Range: *format*

Specifies a choice for the short date format, including abbreviations for the words and separators, according to the list described for **sLongDate**. Control Panel accepts only certain format combinations. Therefore, you should use Control Panel to change this setting. The U.S. English default is M/d/yy (that is, 6/1/90).

sThousand REG_SZ

Range: *string*

Specifies the symbol used to separate thousands. For example, if the value is a comma, the number appears as 3,000. The U.S. English default is , (comma).

sTime REG_SZ

Range: *string*

Specifies the character used to separate the hours, minutes, and seconds in time. For example, if the value is a colon, the time appears as 15:29:31. The U.S. English default is : (a colon).

iNegCurr Negative Number Format

0 = (\$1)	6 = 1-\$	12 = \$ -1.1
1 = -\$1	7 = 1\$-	13 = 1.1- \$
2 = \$-1	8 = -1 \$	14 = (\$ 1.1)
3 = \$1-	9 = -\$ 1	15 = (1.1 \$)
4 = (1\$)	10 = 1 \$-	
5 = -1\$	11 = \$ 1-	

sLongDate Values

d = Day (1-31)
dd = Day (01-31)
ddd = Day (Mon-Sun)
dddd = Day (Monday-Sunday)
M = Month (1-12)
MM = Month (01-12)
MMM = Month (Jan-Dec)
MMMM = Month (January-December)
yy = Year (00-99)
yyyy = Year (1900-2040)

Values for Locales

Values for the locales supported for the first version of Windows NT are the following:

csy = Czech	frc = French (Canadian)
dan = Danish	frs = French (Swiss)
dea = German (Austrian)	hun = Hungarian
des = German (Swiss)	isl = Icelandic
deu = German	ita = Italian
ell = Greek	its = Italian (Swiss)
ena = English (Australia)	nlb = Dutch (Belgian)
enc = English (canada)	nld = Dutch
eng = English (U.K.)	non = Norwegian (Nynorsk)
eni = English (Irish)	nor = Norwegian (Bokmal)
enu = English (U.S.)	plk = Polish
enz = English (New Zealand)	ptb = Portuguese (Brazilian)
esm = Spanish (Mexican)	ptg = Portuguese
esn = Modern Spanish	rus = Russian
esp = Castilian Spanish	sky = Slovak
fin = Finnish	svc = Swedish
fra = French	sve = Swedish
frb = French (Belgian)	trk = Turkish

Keyboard and Keyboard Layout Entries for Users

The Keyboard entry contains user preferences as defined by choosing the Keyboard icon in Control Panel.

Registry path:

HKEY_CURRENT_USER

\Control Panel

\Keyboard



Keyboard Key



Keyboard Layout Key



Keyboard Layout\Substitutes key

Keyboard Key

Registry path:

HKEY_CURRENT_USER
 \Control Panel
 \Keyboard

InitialKeyboardIndicators **REG_SZ**

Range: *Number*

Default: 0

Specifies initial values for keys. 0 means that NumLock is turned off after the user logs on; 2 means Numlock is turned on after the user logs on. This value is set during log off or shutdown to preseve the state of the Numlock key at that time.

KeyboardDelay **REG_SZ**

Range: *0 to 3*

Default: 1

Establishes how much time elapses after you hold down a key before the key starts to repeat. The values 0 through 3 provide a linear scale from the smallest delay supported by the keyboard driver to the largest delay. Typically, 0 represents 250 milliseconds, and 3 represents 1 second, with a 20 percent accuracy.

KeyboardSpeed **REG_SZ**

Range: *0 to 31*

Default: 31

Sets how much time elapses between repetitions of a character on the display when you hold down a keyboard key. The values 0 through 31 provide a linear scale from the slowed repeat rate supported by the keyboard driver to the fastest repeat rate. Typically, 0 represents 2 per second, and 31 represents 30 per second.

Keyboard Layout key

The Keyboard Layout key records the user's preferred layout, which is loaded and activated by the system when the user logs on.

Registry path:

HKEY_CURRENT_USER
 \Control Panel
 \Keyboard Layout

When the user logs off, the user's current keyboard layout is stored here. The value for the entry is based on those defined in CurrentControlSet\Control\NLS\Keyboard Layout. To change the keyboard layout, choose the Windows NT Setup icon from the Main program group.

Active **REG_SZ**

Range: *KeyboardLayout*

Default: 00000409 (for standard U.S. English)

Keyboard Layout\Substitutes Key

Registry path:

HKEY_CURRENT_USER
 \Control Panel
 \Keyboard Layout
 \Substitutes

The Keyboard Layout\Substitutes key is empty by default. This subkey records a mapping between keyboard layout names. The system checks the user's Substitutes subkey when loading the keyboard driver, and if a substitute is specified, the corresponding layout name is substituted. For example, an entry such as the following under the Keyboard Layout\Substitutes subkey indicates that the user prefers the Dvorak U.S. English keyboard layout (00010409) to the standard U.S. English keyboard layout (00000409).

00000409 : REG_SZ : 00010409

Mouse Entries for Users

Registry path:

HKEY_CURRENT_USER
 \Control Panel
 \Mouse

To change these entries, choose the Mouse icon from Control Panel.

DoubleClickSpeed **REG_SZ**

Range: *milliseconds*

Default: 686

Sets the maximum time between clicks of the mouse button that the system permits for one double-click. The lower the value for this entry, the less time you have to click twice to double-click.

MouseSpeed **REG_SZ**

Range: *0 or 1 or 2*

Default: 1

Sets the relationship between mouse and cursor movement when the value of either **MouseThreshold1** or **MouseThreshold2** is exceeded. When this occurs, cursor movement accelerates according to the value of **MouseSpeed**.

0 = No acceleration.

1 = The cursor is moved twice the normal speed when mouse movement exceeds the value of **MouseThreshold1**.

2 = The cursor is moved twice the normal speed when the mouse movement exceeds the value of **MouseThreshold1**, or four times the normal speed if mouse movement exceeds **MouseThreshold2**.

MouseThreshold1 **REG_SZ**

MouseThreshold2 **REG_SZ**

Range: *pixels*

Default: **MouseThreshold1**=6; **MouseThreshold2**=10

These entries set the maximum number of pixels that the mouse can move between mouse interrupts before the system alters the relationship between mouse and cursor movement. If the mouse movement exceeds the threshold defined by **MouseThreshold1** and if **MouseSpeed** is greater than 0, the system moves the cursor at twice the normal speed. If the mouse movement exceeds the threshold defined by **MouseThreshold2** and if **MouseSpeed** is 2, the system moves the cursor at four times the normal speed.

SwapMouseButtons **REG_SZ**

Range: *Boolean*

Default: 0

Specifies whether to swap the right and left mouse buttons. If the value is 1, the buttons are swapped.

Multimedia and Sound Entries for Users

Values related to user preferences for multimedia items in Control Panel are found in the following Registry path:

HKEY_CURRENT_USER
 \Control Panel

The information here is for troubleshooting reference. All changes should be made using the Devices, Drivers, MIDI Mapper, and Sound icons in Control Panel. The following lists the default entries. All are REG_SZ data types.

MMCPL:

H=230
NumApps=20
W=442
X=88
Y=84

Sound:

Beep=yes

Sounds:

Enable=1
SystemAsterisk=chord.wav,Asterisk
SystemDefault=ding.wav,Default Beep
SystemExclamation=chord.wav,Exclamation
SystemExit=chimes.wav,Windows Logoff
SystemHand=chord.wav,Critical Stop
SystemQuestion=chord.wav,Question
SystemStart=tada.wav,Windows Logon

Network Entries for Users

This section describes the user preferences and settings for the network.



Network Connection Entries for Users



Summary of Network Software Entries for Users



Network\Persistent Connections Subkey

Network Connection Entries for Users

The following Registry path contains the list of specific shares to reconnect when the user logs on:

HKEY_CURRENT_USER
 \Network

The Network subkey does not appear unless you are connected to a shared directory when the Reconnect At Logon option was checked in File Manager. There is a subkey for each shared directory to be reconnected at system startup. The name of the subkey is the drive-letter designated for the connection. Each such subkey can contain the following entries:

ConnectionType **REG_DWORD**

Default: 0x1

ProviderName **REG_SZ**

Range: *Network name*

Default: Microsoft Windows Network

Specifies the network provider for the path to the shared directory.

RemotePath **REG_SZ**

Range: *UNC sharename*

Specifies the UNC name for the shared directory.

UserName **REG_SZ**

Range: *username*

Default: (blank) (That is, the name of currently logged on user is assumed.)

Specifies the username under which connection was made to the shared directory if a name was added to the Connect As box in the Connect Network Drive dialog box in File Manager.

Summary of Network Software Entries for Users

The following path contains subkeys with settings for user preferences related to Event Viewer, Server Manager, User Manager, and User Manager for Domains:

HKEY_CURRENT_USER
 \SOFTWARE
 \Microsoft
 \Windows NT
 \CurrentVersion
 \Network

The following summarizes default settings in the Network subkeys. All entries have REG_SZ data types.



Definition of Network Software Parameters

Event Viewer:

Filter=
Find=
IfNT=1
LogType=0
Module=System
SaveSettings=1
SortOrder=0
Window=132 126 504 282 0

Server Manager:

SaveSettings=1

User Manager:

SaveSettings=1

User Manager for Domains:

Confirmation=1
SaveSettings=1
SortOrder=0
Window=132 90 480 258 0

Definition of Network Software Parameters

The following defines most of the common parameters for these applications:

HKEY_CURRENT_USER

 \SOFTWARE

 \Microsoft

 \Windows NT

 \CurrentVersion

 \Network

ApplicationName

Confirmation **REG_SZ**

Range: *Boolean*

Specifies whether the application requests user confirmation for actions such as deletions or other value changes.

SaveSettings **REG_SZ**

Range: *Boolean*

Specifies whether options selected in the application are saved when the application is closed.

SortOrder **REG_SZ**

Range: *Boolean*

Specifies the sort order followed by the application, where 1 specifies sort by username, and 0 specifies sort by full name.

Window **Reg_SZ**

Range: *Pixel location for window*

Specifies window location when application was last closed as four numbers plus 1 or 0 to indicate whether the window was iconized.

Network\Persistent Connections Subkey

The Persistent Connections subkey contains entries that control the restoration of network connections, under this Registry path:

```
HKEY_CURRENT_USER
  \SOFTWARE
    \Microsoft
      \Windows NT
        \CurrentVersion
          \Network
            \Persistent Connections
```

DriveMappingLetter **REG_SZ**

Range: *UNC sharename*

An entry appears for each connection to a shared network directory.

Order **REG_SZ**

Range: *drive-letter order*

Specifies the order for the shared directory connections.

SaveConnections **REG_SZ**

Range: *Yes or No*

Default: Yes

Contains the value set by the Reconnect At Logon check box in the Connect Network Drive dialog box in File Manager.

The following additional parameter is found in this Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \Network
          \World Full Access Shared Parameters
```

ExpandLogonDomain **REG_SZ**

Range: *Yes or No*

Default: Yes

Specifies whether the Shared Directories list is expanded by default in the Connect Network Drive dialog box. This is the value set in the Connect Network Drive dialog box in File Manager by checking the Expand By Default check box.

The following parameter is used by the Windows NT administrative applications:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft\Windows NT\CurrentVersion
    \Network\Shared Parameters
```

Slow Mode **REG_SZ** *String*

Stores information about which servers and domains are across a Low Speed Connection. User Manager for Domains, Server Manager, and Event Viewer read this information unless explicitly told whether to start in Low Speed Connection mode. The cache is updated each time one of these applications is started or set to a new, nonlocal focus, or when the user explicitly changes the Low Speed Connection setting. This is an LRU cache of up to 20 focus targets. The first entry is the most

recently used.

This is a shared state between users, so if one user changes the cached setting for a target focus, other users get that setting by default. The user must be a member of a group with Power Users or better privileges to have access to this subkey.

Default: CLOSEDOMAIN;h;FARDOMAIN;l;\CLOSEMACHINE;h;\FARMACHINE;1

Program Manager Entries for Users



Program Groups Entries



Program Manager Software Groups Entries



Restrictions Entries for Program Manager



Program Manager Settings Entries

Program Groups Entries

The Program Group key contains subkeys that define the contents of all personal program groups in Program Manager, under this Registry path:

```
HKEY_CURRENT_USER\SOFTWARE  
  \Microsoft\Windows NT\CurrentVersion  
    \Program Manager
```

Common groups are defined under the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE  
  \Microsoft\Program Groups
```

The information stored in these subkeys is in binary format, so you cannot easily edit it from Registry Editor. To change the content of program groups, use the mouse and keyboard techniques in Program Manager.

Program Manager Software Groups Entries

The Groups key specifies group numbers for the defined program groups, under this Registry path:

```
HKEY_CURRENT_USER
  \SOFTWARE
    \Microsoft
      \Windows NT
        \CurrentVersion
          \Program Manager
            \Groups
```

These are the default entries. All data types are REG_SZ.

Group1=Main

Group2=Accessories

Group3=Administrative Tools

Group4=Games

Group5=Startup

Restrictions Entries for Program Manager

The Restrictions subkey defines restrictions for activities in Program Manager. Restrictions can be defined for users in User Profile Editor.

Registry path:

```
HKEY_CURRENT_USER\  
    Software\  
        \Microsoft\  
            \Windows NT\  
                \CurrentVersion\  
                    \Program Manager\  
                        \Restrictions
```

EditLevel **REG_DWORD**

Range: *Number*

Default: 0

Sets restrictions for what users can modify in Program Manager. You can specify one of the following values.

- 0 = Allows the user to make any change. (This is the default value.)
- 1 = Prevents the user from creating, deleting, or renaming groups. If you specify this value, the New, Move, Copy, and Delete commands on the File menu are not available when a group is selected.
- 2 = Sets all restrictions in **EditLevel=1** and prevents the user from creating or deleting program items. If you specify this value, the New, Move, Copy, and Delete commands on the File menu are not available at all.
- 3 = Sets all restrictions in **EditLevel=2** and prevents the user from changing command lines for program items. If you specify this value, the text in the Command Line box in the Properties dialog box cannot be changed.
- 4 = Sets all restrictions in **EditLevel=3** and prevents the user from changing any program item information. If you specify this value, none of the areas in the Properties dialog box can be modified. The user can view the dialog box, but all of the areas are dimmed.

NoClose **REG_DWORD**

Range: *0 or 1*

Default: 0

Disables the Exit Windows command on the File menu if this value is 1. Users cannot quit Program Manager through the File Menu or the Control menu (the Exit Windows and Close commands will be dimmed), or by using ALT+F4.

NoFileMenu **REG_DWORD**

Range: *0 or 1*

Default: 0

Removes the File menu from Program Manager if this value is 1. All of the commands on that menu are unavailable. Users can start the applications in groups by selecting them and pressing ENTER, or by double-clicking the icon. Unless you have also disabled the Exit Windows command, users can still quit Windows by using the Control menu or ALT+F4.

NoRun **REG_DWORD**

Range: *0 or 1*

Default: 0

Disables the Run command on the File menu if this value is 1. The Run command is dimmed on the

File menu, and the user cannot run applications from Program Manager unless the applications are set up as icons in a group.

NoSaveSettings REG_DWORD

Range: *0 or 1*

Disables the Save Settings on Exit command on the Options menu if this value is 1. The Save Settings command is dimmed on the Options menu, and any changes that the user makes to the arrangement of windows and icons are not saved when Windows NT is restarted. This setting overrides the **SaveSettings** value in the Program Manager subkey.

Default: 0

Restrictions REG_DWORD

Range: *0 or 1*

Default: 0

Turns restrictions on or off.

ShowCommonGroups REG_DWORD

Range: *0 or 1*

Default: 0x1

Controls whether common program groups are displayed.

Program Manager Settings Entries

Registry path:

HKEY_CURRENT_USER\
 Software
 \Microsoft
 \Windows NT
 \CurrentVersion
 \Program Manager
 \Settings

AutoArrange **REG_DWORD**

Range: *0 or 1*

Default: 0x1

If the AutoArrange command is checked on the Options menu in Program Manager, this value is 1, and the icons in each group are automatically arranged when you run Program Manager.

display.drv **REG_SZ**

Range: *filename*

Default: vga.drv

Defines the video display driver used.

MinOnRun **REG_DWORD**

Range: *0 or 1*

Default: 0x1

If the Minimize On Use command is checked on the Options menu in Program Manager, this value is 1, and Program Manager is iconized when you run another application.

Order **REG_SZ**

Default: 5 4 2 1 3

SaveSettings **REG_DWORD**

Range: *0 or 1*

Default: 0x1

If the Save Settings On Exit command on the Options menu is checked in Program Manager, this value is 1, and Program Manager saves the current configuration when you close Windows.

Window **REG_SZ**

Default: 68 63 636 421 1

Four numbers that indicate the pixel position of the window when Program Manager is opened, followed by a 1 if the window is maximized.

Windows Help Entries

This key includes settings that specify the size and placement of the Windows Help window and dialog boxes, as well as the color of text that, when chosen, displays a macro, pop-up window, or new panel. Entries are found under the following Registry path:

```
HKEY_CURRENT_USER
    \Software
        \Microsoft
            \Windows Help
```

M_WindowPosition **REG_SZ**

H_WindowPosition **REG_SZ**

Range: *number,number, number,number,number*

Default: H_WindowPosition=[213,160,213,160,0]

M_WindowPosition=[212,4,425,476,0]

These entries define the default size and position of the main Help window and the History dialog box. The first and second *number* values define the x and y coordinates of the upper-left corner for the default position of the main Help window and the dialog boxes. The third and fourth *number* values define the default width and height (in pixels) of the Help window and the History dialog box. The fifth *number* specifies whether the main Help window is maximized, where 1 is maximized, and 0 is the default height and width of the window. Changing this value has no effect on the size of a dialog box, because it cannot be maximized.

To change this entry, move or size the main Help window or the History dialog box.

Maximized **REG_SZ**

Default: 0

XI **REG_SZ**

Default: 166

Xr **REG_SZ**

Default: 474

Yd **REG_SZ**

Default: 444

Yu **REG_SZ**

Default: 120

IFJumpColor **REG_SZ**

Range: *red-value green-value blue-value*

Specifies the color of text that, when chosen, leads to a new panel of information in another help file.

IFPopupColor **REG_SZ**

Range: *red-value green-value blue-value*

Specifies the color of text that, when chosen, displays a pop-up panel (the type of panel used for glossary definitions) located in a different help file.

JumpColor **REG_SZ**

Range: *red-value green-value blue-value*

Specifies the color of text that leads to a new panel of help information. For example, entering 000

000 000 as values results in black text on a white background.

MacroColor REG_SZ

Range: *red-value green-value blue-value*

Specifies the color of text that, when chosen, runs a Help macro.



PopupColor REG_SZ

Range: *red-value green-value blue-value*

Specifies the color of text that, when chosen, displays a pop-up panel (the type of panel used for glossary definitions). For example, entering 000 000 000 as values results in black text on a white background.

Registry Entries for Fonts

This section describes entries in subkeys that concern the fonts available to all users on a computer.

	<u>Font Drivers Entries</u>
	<u>FontCache Entries</u>
	<u>Fonts Entries</u>
	<u>GRE_Initialize Entries</u>
	<u>FontSubstitutes Entries</u>
	<u>TrueType Entries for Users</u>

Font Drivers Entries

The Font Drivers subkey in the following Registry path can contain references to external font drivers:

HKEY_LOCAL_MACHINE\SOFTWARE

 \Microsoft

 \Windows NT

 \CurrentVersion

 \Font Drivers

You should not need to modify this entry directly. Your font vendor should supply an installation program for adding and removing drivers.

Driver description **REG_SZ**

Range: *Driver filename or pathname*

Lists external font drivers installed on the system. Windows NT does not include any external font drivers. The bitmap, vector, and TrueType drivers are built in and do not appear on this list.

FontCache Entries

The FontCache subkey contains entries that define parameters for font caching.

Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \FontCache
```

The value entries in the FontCache subkey can greatly influence the amount of memory used by the system. However, these values should not be modified, except in the rare case where you must tune the performance for an international version of Windows NT or for specialized cases such as a print shop, where you may be manipulating large character sets.

MaxSize REG_DWORD

Range: *Number of kilobytes*

Default: 0x80

Specifies the maximum amount of address space reserved per font cache.

MinIncrSize REG_DWORD

Range: *Number of kilobytes*

Default: 0x4

Specifies the minimum amount of memory committed each time a font cache is grown.

MinInitSize REG_DWORD

Range: *Number of kilobytes*

Default: 0x4

Specifies the minimum amount of memory initially committed per font cache at the time of creation.

Fonts Entries

The following Registry path is for entries describing the fonts used for displaying information in applications created for Windows NT or versions of Windows for MS-DOS:

```
HKEY_LOCAL_MACHINE    \SOFTWARE
    \Microsoft
        \Windows NT
            \CurrentVersion
                \Fonts
```

Entries in the Fonts key have the following format:

Font Name **REG_SZ**

Range: *font filename*

These value entries define the installed fonts and their related filenames. These are the default value entries:

Arial=ARIAL.FOT
Arial Bold=ARIALBD.FOT
Arial Bold Italic (TrueType)=ARIALBI.FOT
Arial Italic (TrueType)=ARIALI.FOT
Courier 10,12,15 (VGA res)=COURE.FON
Courier New (TrueType)=COUR.FOT
Courier New Bold (TrueType)=COURBD.FOT
Courier New Bold Italic (TrueType)=COURBI.FOT
Courier New Italic (TrueType)=COURI.FOT
Modern (Plotter)=MODERN.FON
MS Sans Serif 8,10,12,14,18,24 (VGA res)=SSERIFE.FON
MS Serif 8,10,12,14,18,24 (VGA res)=SERIFE.FON
Roman (Plotter)=ROMAN.FON
Script (Plotter)=SCRIPT.FON
Small Fonts (VGA res)=SMALLE.FON
Symbol (TrueType)=SYMBOL.FOT
Symbol 8,10,12,14,18,24 (VGA res)=SYMBOLE.FON
Times New Roman (TrueType)=TIMES.FOT
Times New Roman Bold (TrueType)=TIMESBD.FOT
Times New Roman Bold Italic (TrueType)=TIMESBI.FOT
Times New Roman Italic (TrueType)=TIMESI.FOT
WingDings (TrueType)=WINGDING.FOT

GRE_Initialize Entries

The following Registry path is for entries describing the fonts used for character-based programs:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \GRE_Initialize
```

The FONTS.FON and FIXEDFON.FON entries do not affect the console, but they do affect menus and dialog boxes and some applications such as Notepad.

Caution: Editing these entries can cause menus and dialog boxes to display improperly. Unlike versions of Windows for MS-DOS, changing these default fonts will render poor results, because the font set under Windows NT is closely tied to the driver.

FONTS.FON REG_SZ

Range: *Filename*

Specifies the filename of the default system font.

FIXEDFON.FON REG_SZ

Range: *Filename*

Specifies the filename of the default system fixed-width font.

OEMFONT.FON REG_SZ

Range: *Filename*

Specifies the filename of the default OEM (or console) font.

FontSubstitutes Entries

The entries in the FontSubstitutes subkey define substitute typeface names for fonts under the following Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \FontSubstitutes
```

You should not need to modify these entries. This subkey is usually used by applications with a special need to equate font names.

Alternate name **REG_SZ**

Range: *Actual name*

Default: **Helv**=MS Sans Serif; **Helvetica**=Arial; **Times**=Times New Roman; **Tms Rmn**=MS Serif

Specifies the alternate typeface name. For example, the following entry means that Helv is an alternative typeface name that can be used to refer to the MS Sans Serif font:

Helv=MS Sans Serif

TrueType Entries for Users

The entries in the TrueType subkey describe options that affect the use and display of TrueType fonts in Windows-based applications. This is the Registry path:

```
HKEY_CURRENT_USER
  \Software
    \Microsoft
      \Windows NT
        \CurrentVersion
          \TrueType
```

The TrueType subkey can contain the following entries:

TTEnable REG_SZ

Range: *Boolean*

Default: 1

Controls whether TrueType fonts are available. Setting this value to 1 makes TrueType fonts available in your Windows-based applications. Setting this value to 0 turns off TrueType fonts so they are unavailable in applications.

TTonly REG_SZ

Range: *Boolean*

Default: 0

Specifies whether to make only TrueType fonts available in Windows-based applications. If this value is set to 1, only TrueType fonts are available. If this value is set to 0, all fonts installed on your system are available. To change this entry, choose the Fonts icon from Control Panel.

Registry Entries for Printing

Always use Print Manager to change configuration settings for all printers. There are no parameters for printing that can be set only through the Registry.

The Registry contains printer information in these locations:



The per-user settings for the current default printer are stored under this key:

HKEY_CURRENT_USER\Printers



The hardware-specific information about drivers and print processors is stored under the following key, where *Hardware* represents the subkey for a specific Windows NT platform, such as Windows NT x86 or Windows NT R4000:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet

\Control\Print



Printing Entries for Users



Control\Print Entries for the Computer



Print Environments Entries



Print Monitors Entries



Printers Entries



Print Providers Entries

Printing Entries for Users

The following Registry path contains a description of the default printer, as selected by the current user:

HKEY_CURRENT_USER

\Printers

The following Registry path contains the user preferences for Print Manager:

HKEY_CURRENT_USER

\Software

\Microsoft

\Windows NT

\CurrentVersion

\Print Manager

These standard entries appear:

Network REG_DWORD

Default: 0x1

Print Manager REG_BINARY

Default:

Save Settings REG_DWORD

Default: 0x1

Control\Print Entries for the Computer

The principal information for printers appears under the following Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Control

 \Print

This path contains additional subkeys that define supporting DLLs, drivers, and other necessary information for installed printers. These subkeys can include Environments, Forms, Monitors, Printers, and Providers. An OEM print provider might also add subkeys under this Registry path.

Print Environments Entries

The following Registry path contains these subkeys, each of which can contain a value entry defining the directory that contains the appropriate drivers:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Print
        \Environments
          \Windows NT environment description
            \Drivers
              \Print Processors
```

Environment description keys include the following:

```
\Windows NT Alpha_AXP
\Windows NT R4000
\Windows NT x86
```

A *PrinterDriverName* subkey under the related Drivers subkey can contain these values:

Configuration File **REG_SZ**

Default: *Installed DLL filename*

Data File **REG_SZ**

Default: *Installed .PPD filename*

Driver **REG_SZ**

Default: *Installed driver DLL filename*

Version **REG_DWORD**

Default: *Version number*

A *Print Processors* subkey (and subsequent subkeys) under this same Registry path contains an entry of the following type:

Driver **REG_SZ**

Default: winprint.dll

Print Monitors Entries

The entry in the following path defines the DLL filename for the appropriate print monitor:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Print
        \Monitors
          \Provider Network Port
```

Driver **REG_SZ**

Range: *DLL filename*

The Options subkey for the provider network port contains the following entries, all of which are REG_DWORD:

Adapter=0

ConnectionType=0x1

DlcBufferSize=0x27100

DlcT1Timer=0x5

DlcT2Timer=0x2

DlcTiTimer=0x3

EventLogging=0x7

LinkStationsUsed=0x40

StatusUpdateInterval=0x3c

Printers Entries

Each installed printer has a subkey in the following Registry path:

```
HKEY_LOCAL_MACHINE\SYSTEM
  \CurrentControlSet
    \Control
      \Print
        \Printers
          \printer name
```

The following entries can appear under such a subkey:

Attributes : REG_DWORD: 0x1
Datatype : REG_SZ: RAW
Default DevMode : REG_BINARY:
Description : REG_SZ: *driver description on port*
Location : REG_SZ :
Name : REG_SZ : *user defined*
Parameters : REG_SZ :
Port : REG_SZ : *port name*
Print Processor : REG_SZ: WinPrint
Printer Driver : REG_SZ : *driver name selected in Setup*
Priority : REG_DWORD: 0x1
Security : REG_BINARY :
Separator File : REG_SZ :
Share Name : REG_SZ :
StartTime : REG_DWORD : 0
Status : REG_DWORD: 0
UntilTime : REG_DWORD: 0

Print Providers Entries

Each print service provider has a subkey in the following Registry path:

HKEY_LOCAL_MACHINE\SYSTEM

 \CurrentControlSet

 \Control

 \Print

 \Providers

 Print Services Name

The default subkey for a Windows NT network is LanMan Print Service. Such a subkey contains the following entry:

Name **REG_SZ**

Range: *DLL filename*

Default: win32spl.dll

Registry Entries for Subsystems

This section describes software registration entries related to Windows NT subsystems.



Microsoft OS/2 Version 1.x Software Registration Entries



Windows Software Registration Entries



WOW Software Registration Entries



Windows NT Software Registration Entries



Winlogon Registry Entries

Microsoft OS/2 Version 1.x Software Registration Entries

The basic software information for the Microsoft OS/2 version 1.x subsystem is found in the following Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \OS/2 Subsystem for NT
```

The OS/2 Subsystem key contains several subkeys, but initially only the \config.sys subkey contains an entry. This subkey contains the OS/2 CONFIG.SYS information stored after an OS/2 application has been run on the system.

If there is no OS/2 CONFIG.SYS file, this subkey contains the following entry:

PROTSHELL=C:

```
  \os2
  \pmshell.exe c:
  \os2
  \os2.ini c:
  \os2
  \os2sys.ini
  %SystemRoot%
  \system32
  \cmd.exe
```

SET COMSPEC=%SystemRoot%

```
  \system32
  \cmd.exe
```

The Os2 subkey under CurrentControlSet\Control\Session Manager\SubSystems defines the path to the executable file used to start the OS/2 subsystem. The **Os2LibPath** value under the Session Manager\Environment subkey defines the directory path for the OS/2 library. These entries are described in CurrentControlSet\Control Subkeys.

To disable the OS/2 subsystem, set the value of **GlobalFlag** to 20100000 in the following subkey:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet
  \Control\Session Manager
```

Windows Software Registration Entries

The Windows subkey under the following path defines some values used by applications created to run under Windows for MS-DOS:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \Windows
```

AppInit_DLLs **REG_SZ**

Causes all the specified DLLs (one or many) to be attached to all Windows-based applications. This means that once this is set for a session, upon restarting the system, all the Windows-based applications that run in that session will load the specified DLLs. For example, an applications developer can use it to attach the Microsoft Call/Attribute Profiler to all Windows-based applications by calling CAPSETUP.EXE, which sets the **AppInit_DLLs** so the user doesn't have to do it through Registry Editor.

DeviceNotSelectedTimeout **REG_SZ**

Range: *Seconds*

Default: 15

Specifies how much time the system waits for a device to be switched on. If the device is not switched on during this time, the system won't print to the device. For some devices, the system immediately posts an error message if the device is not already switched on. This entry only defines the system default value. To change the value for a particular printer, use Print Manager.

Spooler **REG_SZ**

Range: *Boolean*

Default: Yes

Specifies whether output to the printer is to be sent through Print Manager. Changing this value to No turns off Print Manager.

swapdisk **REG_SZ**

Range: *drive:directory*

Default: The directory pointed to by the TEMP environment variable; if there is no TEMP variable, the default is the boot directory of your first hard disk (usually C:).

Provides the name of the disk drive and directory to which Windows for MS-DOS in standard mode swaps non-Windows-based applications.

TransmissionRetryTimeout **REG_SZ**

Range: *Seconds*

Default: 45

Specifies the default amount of time for attempted transmission retries. If a successful transmission does not occur during this time, Print Manager displays a message stating that the printer is not receiving characters. This setting serves only as the system default value. To change the value for a particular printer, use Print Manager.

WOW Software Registration Entries

Software registration values for the WOW subsystem appear under the following Registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Microsoft
    \Windows NT
      \CurrentVersion
        \WOW
```

The WOW subkeys have the same names as headings in the SYSTEM.INI file, and the values are the same items as were contained in the Windows for MS-DOS SYSTEM.INI file. All are REG_SZ value types.

The information provided here is for troubleshooting only. These are the WOW subkeys and their default values.

Boot

Lists drivers and Windows 3.x modules used by some 16-bit Windows-based applications, with these entries and default values to map Windows 3.x drivers to Windows NT:

comm.drv = comm.drv

display.drv = vga.drv

drivers = mmsystem.dll

fixedfon = vgafix.fon

keyboard.drv = keyboard.drv

language.dll = (empty)

mouse.drv = mouse.drv

network.drv = lanman.drv

oemfonts.fon = vgaoem.fon

shell = progman.exe

sound.drv = sound.drv

system.drv = system.drv

boot.description

Used by some 16-bit Windows-based applications, with these kinds of entries and default values:

display.drv = VGA

keyboard.typ = Enhanced 101/102 key U.S. and Non U.S. keyboards

language.dll = English (American)

network.drv = LAN Support

system.drv = MS-DOS or PC-DOS System

Compatibility

Used to translate 16-bit Windows APIs and messages to 32-bit equivalents.

MAILSPL = 0x40000000 (for MS Mail)

SHOPPER = 0x20000000 (for Clip-Art Windows Shopper)

SHADOW = 0x10000000 (for BeyondMail installation)

WINPROJ = 0x80000000 (for Microsoft Project)

ESCAPES = 0x1000000 (for Micrographix Escapes)

HIRES = 0x00100000 (for HIRES display cards)

Keyboard

Contains information about the keyboard used by some 16-bit Windows-based applications, with these kinds of entries and default values:

keyboard.dll = (empty)

subtype = (empty)

type = 4

NonWindowsApp

Contains information used by non-Windows-based applications. This is handled automatically by Windows NT. Unless you manually added values to SYSTEM.INI in Windows 3.x, this subkey is empty.

Standard

Contains entries specific to running Windows 3.x in standard mode, used by some 16-bit Windows-based applications. Unless you manually added values to SYSTEM.INI in Windows 3.x, this subkey is empty.

The [386Enh] section is read from the SYSTEM.INI file. There is no Registry equivalent.

Windows NT Software Registration Entries

Printing and fonts are described in separate sections. Other software registration entries for Windows NT are found in this Registry Path

HKEY_LOCAL_MACHINE\SOFTWARE

 \Microsoft

 \Windows NT

 \CurrentVersion

These are the basic entries in this key:

CurrentBuild, CurrentType, and CurrentVersion **REG_SZ**

Range: *Descriptionsr*

Identifies the current version of the Windows NT.

InstallDate **REG_DWORD**

Range: *Number*

Describes the installation date for Windows NT.

PathName **REG_SZ**

Range: *Pathname*

Default: C:\winnt

Defines the path where Windows NT system files are saved.

RegisteredOrganization and RegisteredOwner **REG_SZ**

Range: *Name*

Defines the registered organization and owner, as specified during Setup.

SoftwareType **REG_SZ**

Range: *Type*

Default: SYSTEM

Defines the type of software for this version.

SourcePath **REG_SZ**

Default: Path for installation media during Setup.

Defines the source of the Setup files.

SystemRoot **REG_SZ**

Default: Specified during Setup (typically, C:\winnt)

Defines the path for the directory containing the Windows NT files.

Winlogon Registry Entries

The Registry value entries that control the logon sequence for starting Windows NT are found under the following Registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE
    \Microsoft
        \Windows NT
            \CurrentVersion
                \Winlogon
```

AutoAdminLogon **REG_SZ**

Range: *0 or 1*

Default: 0

Specifies automatic logon if this value is 1. You must also add the value entry **DefaultPassword** with a value for the user listed under **DefaultUserName** for automatic logon to work.

When **AutoAdminLogon** is used, Windows NT automatically logs on the specified user when the system is started, bypassing the CTRL+ALT+DEL logon dialog box.

DefaultDomainName **REG_SZ**

Range: *Domain name*

Default: NEWDOMAIN

Specifies the name of the last successfully logged on domain.

DefaultPassword **REG_SZ**

Range: *Password*

Specifies the password for the user listed under **DefaultUserName**. Used during automatic logon.

DefaultUserName **REG_SZ**

Range: *Username*

Specifies the name of the last successfully logged on user. If values are defined for **DefaultPassword** and **AutoAdminLogon**, this is the user who is logged on by default during automatic logon.

LegalNoticeCaption **REG_SZ**

Range: *String*

Default: (none)

Specifies a caption for a message to appear when the user presses CTRL+ALT+DEL during logon. Add this value entry if you want to add a warning to be displayed when a user attempts to log on to a Windows NT system. The user cannot proceed with logging on without acknowledging this message.

To specify text for the message, you must also specify a value for **LegalNoticeText**.

LegalNoticeText **REG_SZ**

Range: *String*

Default: (none)

Specifies for a message to appear when the user presses CTRL+ALT+DEL during logon. Add this value entry if you want to add a warning to be displayed when a user attempts to log on to a Windows NT system. The user cannot proceed with logging on without acknowledging this message.

To include a caption for the logon notice, you must also specify a value for **LegalNoticeCaption**.

ReportBootOk **REG_SZ**

Range: *0 or 1*

Default: 1

When this value is set to 0, it disables the automatic (default) startup acceptance, which happens after the first successful logon. This value must be 0 if you use alternate settings in the BootVerification or BootVerificationProgram keys.

Shell REG_SZ

Range: *Executable names*

Default: taskman,progman,wowexec

Specifies executables that are run by USERINIT and that are expected to be in the users shell program. If for some reason WinLogon cannot start the entries listed in **Userinit**, then WinLogon will execute the entries in **Shell** directly.

System REG_SZ

Range: *Executable names*

Default: lsass.exe,spoolss.exe

Specifies executables to be run by WinLogon in the system context. These are activated during system initialization.






Userinit REG_SZ

Range: *Executable names*

Default: USERINIT,NDDEAGNT.EXE

Specifies executables to be run by WinLogon when a user logs on. These executables are run in the user context. The first entry (USERINIT) is responsible for executing the shell program. NDDEAGNT.EXE is needed to run NetDDE.

Summary of Other Registry Keys

	<u>HKEY_LOCAL_MACHINE\HARDWARE Subtree</u>
	<u>HKEY_LOCAL_MACHINE\SAM Subtree</u>
	<u>HKEY_LOCAL_MACHINE\SECURITY Subtree</u>
	<u>HKEY_LOCAL_MACHINE\SOFTWARE Subtree</u>
	<u>HKEY_CLASSES_ROOT</u>

HKEY_LOCAL_MACHINE\HARDWARE Subtree

The HKEY_LOCAL_MACHINE\HARDWARE subtree contains the hardware data in the Registry that is computed at system startup. This includes information about hardware components on the system board and about the interrupts hooked by specific hardware devices.

The Hardware key contains distinct and important sets of data in three subkeys -- Description, DeviceMap, and ResourceMap.

All information in HKEY_LOCAL_MACHINE\HARDWARE is *volatile*, which means that the settings are computed each time the system is started and then discarded when the system is shut down.

Applications and device drivers use this key to read information about the system components, store data directly into the DeviceMap section, and store data indirectly into the ResourceMap section.

Note: Do not try to edit the data in HKEY_LOCAL_MACHINE\HARDWARE, because this is a volatile key. Also, much of the information appears in binary format, making it difficult to decipher.

To view data about a computer's hardware in an easy-to-read format for troubleshooting, run WinMSD, and choose the Devices button. WinMSD extracts the information from the Registry and renders it in a more readable format.



Hardware Description Subkey



Hardware DeviceMap Subkey



Hardware ResourceMap Subkey

Hardware Description Subkey

The Description subkey under HKEY_LOCAL_MACHINE\HARDWARE displays information from the hardware database built by the firmware, NTDETECT, and the Executive itself:



If the computer is a RISC-based computer, this database is a copy of the ARC configuration database taken from the firmware.



If the computer is an x86-based computer, this database contains the data found by the Hardware Recognizer, which is a program that runs as part of the Windows NT startup sequence. The Hardware Recognizer on x86-based computers is NTDETECT.COM.



If the target computer is not PC-compatible, the OEM provides its own version of NTDETECT.COM as the Hardware Recognizer.

The Hardware Recognizer for x86-based computers detects the following:

- Bus/adapter type
- Communication ports
- Floating point coprocessor
- Floppy drives
- Keyboard
- Machine ID
- Mouse
- Parallel ports
- SCSI adapters
- Video adapter

Network adapter cards are not detected as part of startup but are instead detected during Windows NT Setup or if you choose the Network icon in Control Panel to install a new network adapter. See also [Registry Entries for Network Adapter Cards](#).

The *MultifunctionAdapter* subkey under the Description key contains several subkeys, each corresponding to specific bus controllers on the local computer. Each of these subkeys describes a class (or type) of controller, including controllers for disk drives, display, keyboard, parallel ports, pointing devices, serial ports, and SCSI devices. The subkey's path describes the type of component. The numbering for hardware components is 0-based, which means that, for example, the first (or only) disk controller appears under the 0 subkey.

The name of the *MultiFunctionAdapter* key depends on the bus type. For example, the subkey name for ISA and MCA buses appears as MultiFunctionAdapter. For EISA buses, the key name is EisaAdapter, and for TurboChannel buses, the name can be TcAdapter.

For each detected hardware component, the optional **Component Information** and **Configuration Data** value entries store version and configuration data in binary format. The **Identifier** entry contains the name of a component, if specified. For details about these entries, see [Registry Entries for Device Drivers](#)

Hardware DeviceMap Subkey

Under HKEY_LOCAL_MACHINE\HARDWARE\DeviceMap, each *Device* subkey contains one or more values to specify the location in the Registry for specific driver information for that kind of component.

The value for each *Device* subkey describes an actual port name or a Services subkey in the following path:

HKEY_LOCAL_MACHINE\SYSTEM

 \ControlSet nnn

 \Services

The related Services subkey contains information about a device driver. That *Service* subkey contains the information a system administrator might need for troubleshooting and is also the information presented about the device in WinMSD.

Hardware ResourceMap Subkey

The ResourceMap subkey under HKEY_LOCAL_MACHINE\HARDWARE maps device drivers to resources that the drivers use. Each ResourceMap subkey contains data reported by the device driver about its use of I/O ports, I/O memory addresses, interrupts, DMA channels, and so on. The data in the ResourceMap subkey is volatile, so this subkey is recreated each time you start Windows NT.

Under the ResourceMap subkey, there are *DeviceClass* subkeys for the general class (or type) of devices. Each of these subkeys contains one or more *DriverName* subkeys with information about a specific driver. For example, Sermouse is the *DriverName* subkey under the PointerPort *DeviceClass* subkey. The driver names in these subkeys match the services listed in the following path:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services

Note: If you need to resolve resource conflicts, use WinMSD to view the data from these subkeys in an easily readable format.

HKEY_LOCAL_MACHINE\SAM Subtree

The HKEY_LOCAL_MACHINE\SAM subtree contains the user and group account information in the Security Account Manager (SAM) database for the local computer. For a computer that is running Windows NT Advanced Server, this key also contains security information for the domain. This information is what you see in User Manager, and it also appears in the lists of users and groups when you use the Security menu commands in File Manager.

This key is mapped to HKEY_LOCAL_MACHINE\SECURITY\SAM, so changes made in one key automatically appear in the other key.

Note: The information in this database is in binary format and cannot be changed using Registry Editor. If you want to change user account or global group account information, use User Manager or User Manager for Domains to add or remove users or to change security information for the local computer or for the domain.

HKEY_LOCAL_MACHINE\SECURITY Subtree

The HKEY_LOCAL_MACHINE\SECURITY subtree contains security information for the local computer, including user rights, password policy, and the membership of local groups, as set in User Manager.

The subkey HKEY_LOCAL_MACHINE\SECURITY\SAM is mapped to HKEY_LOCAL_MACHINE\SAM, so changes made in one key automatically appear in the other key.

Note: The information in this database is in binary format and cannot be changed using Registry Editor. If you want to change user account or global group account information, use User Manager or User Manager for Domains to add or remove users or to change security information for the local computer or for the domain.

HKEY_LOCAL_MACHINE\SOFTWARE Subtree

The HKEY_LOCAL_MACHINE\SOFTWARE subtree contains specific configuration information about software on the local computer. The entries under this handle, which apply for anyone using this particular computer, show what software is installed on the computer and also define file associations and OLE information. The HKEY_CLASSES_ROOT handle is an alias for the subtree rooted at HKEY_LOCAL_MACHINE\SOFTWARE\Classes.

This key contains, for example, the information you add by using the Associate command in File Manager, information added during installation for specific Windows-based applications, and information about applications installed with Windows NT.

The HKEY_LOCAL_MACHINE\SOFTWARE subtree contains several subkeys. The Classes, Program Groups, and Secure subkeys are described here, plus general information about the various *Description* subkeys that might appear in a Registry.

The key HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft and its subkey named Windows NT\CurrentVersion are of particular interest. These subkeys contain information about software that supports services built into Windows NT, as well as data about the version and type of the current release (multiprocessor versus uniprocessor). For example, it is possible to run a Windows NT uniprocessor kernel on a multiprocessor computer, but you don't get any multiprocessor benefits by doing so. To quickly check which kernel type is running on a computer, see the data in the Registry under the Windows NT\CurrentVersion subkey.

HKEY_CLASSES_ROOT

HKEY_CLASSES_ROOT contains information about file associations and OLE. This is the same data as in the Classes subkey under HKEY_LOCAL_MACHINE\SOFTWARE.

The sole purpose for HKEY_CLASSES_ROOT is to provide compatibility with the Windows 3.1 registration database.

As with Windows for MS-DOS, the Windows NT File Manager includes an Associate dialog box for associating a filename extension with a specific application. Windows NT stores these associations in the Windows NT Registry. The Associate dialog box should be used whenever possible to define filename associations.

HKEY_LOCAL_MACHINE\SOFTWARE\Classes Subkey

The Classes subkey defines types of documents, providing information on filename-extension associations and OLE information that can be used by Windows shell applications and OLE applications. HKEY_CLASSES_ROOT displays the same information as stored under this subkey.

Important: The OLE information must be created by the specific application, so you should not change this information using Registry Editor. If you want to change filename-extension associations, use the Associate command in File Manager.

The Classes subkey contains two kinds of subkeys:



Filename-extension subkeys, which specify the application associated with files that have the selected extension.



Class-definition keys, which specify the shell and OLE properties of a class (or type) of document. These subkeys can describe shell and protocol properties for each class of document. If an application supports DDE, the Shell subkey can contain Open and Print subkeys that define DDE commands for opening and printing files, similar to the OLE and DDE information stored in the registry database under versions of Windows for MSDOS. For example, **cardfile.exe /p %1** is the print command, and the **%1** parameter stands for the selected filename in File Manager when the command is carried out.

HKEY_LOCAL_MACHINE\SOFTWARE\Description Subkeys

The various HKEY_LOCAL_MACHINE\SOFTWARE\Description subkeys contain the names and version numbers of the software installed on the local computer. (Information about the configuration of these applications is stored on a per-user basis under HKEY_CURRENT_USER.)

During installation, applications record this information in the following form:

HKEY_LOCAL_MACHINE\SOFTWARE

 \<CompanyName>

 \<ProductName>

 \<Version>

The HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft subkey is one variation of this form.

Note: The information in each subkey is added by the related application. Do not edit entries in these subkeys unless directed to do so by your application vendor.

HKEY_LOCAL_MACHINE\SOFTWARE\Program Groups Subkey

The Program Groups subkey under HKEY_LOCAL_MACHINE\SOFTWARE contains the common program groups -- that is, those used in common by all users of the local computer. (The program groups for an individual user can be viewed under HKEY_CURRENT_USER, and the default personal program groups can be viewed in HKEY_USERS\DEFAULT.) Each subkey under the Program Groups subkey is the name of a common program group, and its value is binary data describing that program group.

If you want to change the content of common program groups, use the menu commands or mouse techniques provided in Program Manager, or the User Profile Editor in Windows NT Advanced Server.

This is the default Registry subkeys:

HKEY_LOCAL_MACHINE\SOFTWARE

Program Groups

Applications

HKEY_LOCAL_MACHINE\SOFTWARE\Secure Subkey

The Secure subkey provides a convenient place for applications to store configuration information that should not be changed by anyone except an administrator. This is the Registry path:

HKEY_LOCAL_MACHINE\SOFTWARE
Secure

Windows 3.1 Migration Status Subkey

This subkey is used during installation to track what portions of the Microsoft Windows 3.1 information have successfully migrated to the Windows NT Registry. There are no user-definable value entries in this subkey.

This is the Registry path:

```
HKEY_LOCAL_MACHINE\SOFTWARE  
    Windows 3.1 Migration Status
```

System Components with No Additional Entries

The following system components do not write parameters in the Registry, besides the basic information shown in CurrentControlSet\Services:

- Abiosdsk
- Beep
- ClipSrv
- Cpqarray
- Dellarry
- Delldsa
- Diskperf
- File system drivers
- Floppy driver
- MUP utility
- NetDDE
- Null
- Printing (all keys)
- Schedule service
- SCSI class drivers
- Simbad
- SNMP
- Telnet and Telnetsys
- Videoprt
- Windbg

LAN Manager vs. Windows NT Parameters

This list describes the comparable value entries in the Windows NT Registry that replace Microsoft LAN Manager parameters.

LAN Manager parameters that relate to MS-DOS are not listed. Also, because Windows NT dynamically allocates memory for network operations, many LAN Manager parameters are no longer relevant.

In the following lists, N/A indicates that the LAN Manager parameter has no corresponding Registry entry or user-interface selection in Windows NT.

Lan Manager vs. Windows NT Settings for Services

In LAN Manager, service parameter settings are found in the file LANMAN.INI. In Windows NT, these settings are stored in the Registry.

In Windows NT, these entries are in the Registry in this path:

HKEY_LOCAL_MACHINE\SYSTEM
 \CurrentControlSet\Services*Service name*\Parameters

Alert Service

LANMAN.INI [alerter] = CurrentControlSet\Services\LanmanServer\Parameters

sizalertbuf = N / A

Messenger Service

LANMAN.INI [messenger] = CurrentControlSet\Services\LanmanServer\Parameters

logfile = Messages are not stored

sizmessbuf = Set by operating system

Netlogon Service

LANMAN.INI [Netlogon] = CurrentControlSet\Services\Netlogon\Parameters

pulse = **PulseInterval**

randomize = **Randomize**

scripts = Use Server Manager or Services in Control Panel

update = **Update**

Remote Access Service

LANMAN.INI [RemoteAccess] = CurrentControlSet\Services\RemoteAccess\Parameters

audit = **EnableAudit**

authenticateretries = **AuthenticateRetries**

authenticatetime = **AuthenticateTime**

**LANMAN.INI [RemoteAccess] = CurrentControlSet\Services\RemoteAccess\Parameters
 \NetbiosGateway\Parameters**

autodisconnect = **Autodisconnect**

enablebroadcast = **EnableBroadcast**

maxdynmem = **MaxDynMem**
maxnames = **MaxNames**
maxsessions = **MaxSessions**
multicastforwardrate = **MulticastForwardRate**
numworkbufs = N / A
raslannets = N / A
remotelisten = **RemoteListen**
sizeworkbufs = **SizWorkbufs**

Replicator Service

LANMAN.INI [Replicator] = CurrentControlSet\Services\Replicator\Parameters

exportpath = Use Server Manager to set export path
exportlist = Use Server Manager to enter To List names
guardtime = **Guardtime**
importpath = Use Server Manager to set import path
importlist = Use Server Manager to enter To List names
interval = **Interval**
logon = N / A
password = N / A
pulse = **Pulse**
random = **Random**
replicate = Use Server Manager to set

Server Service

LANMAN.INI [server] = CurrentControlSet\Services\LanmanServer\Parameters

accessalert = N / A
alertnames = Use Server Manager to list users to receive alert messages (corresponds to **Alertnames** in the Services\Alerter subkey)
alertsched = **AlertSched**
auditing = Use User Manager for Domains to set audit policy
autodisconnect = **Disc**
autopath = N / A
autoprofile = N / A
diskalert = **DiskspaceThreshold**
erroralert = **ErrorThreshold**
guestacct = Use User Manager to enable guest account
logonalert = N / A
maxauditlog = Set using Event Viewer (corresponds to **MaxSize** in Services\Eventlog subkey)
maxchdevjob = N / A
maxnetbios = N / A
maxsessopens = **SessOpens**
maxsessreqs = **MaxMpxCt**
netioalert = **NetworkErrorThreshold**

noauditing = Use User Manager to set
numadmin = N / A
numfiletasks = N / A
security = No share security
sizreqbuf = **SizReqBuf**
srvanndelta = **AnnDelta**
srvannounce = **Announce**
srvcomment = **Comment**
srvhidden = **Hidden**
userpath = N / A

Server heuristics in LAN Manager = CurrentControlSet\Services\LanmanServer\Parameters

5 = **EnableFCOpens**
6 = **ThreadPriority**
10 = **ScavTimeout**
11 = **EnableSoftCompat**
15 = **OplockBreakWait**

Autotuned server parameters = CurrentControlSet\Services\LanmanServer\Parameters

maxshares = N / A
maxusers = N / A
numbigbuf = N / A
numreqbuf = N / A

SNMP Service

LANMAN.INI [snmp] = CurrentControlSet\Services\Snmp\Parameters

authtrap = **EnableAuthenticationTraps**
communitynames = **ValidCommunities**
hostmasks = No hostmasks
trapcommunity = **TrapConfiguration**
trapdestinations = **TrapConfiguration**
version = N/A

UPS Service

LANMAN.INI [ups] (servers only) = CurrentControlSet\Services\UPS

batterytime = **BatteryLife**
cmdfile = **CommandFile**
messdelay = **FirstMessageDelay**
messtime = **MessageInterval**
recharge = **RechargeRate**
signals = **Options**

Workstation Service

LANMAN.INI [workstation] = CurrentControlSet\Services\LanmanWorkstation\Parameters

charcount = **MaxCollectionCount**

chartime = **CollectionTime**

charwait = **CharWait**

computername, domain, othdomains = Must be set through Network option in Control Panel

keepconn = **KeepConn**

keepsearch = Set to 5 minutes in Windows NT

maxcmds = Windows NT does not limit number of commands except as specified by the server

maxerrorlog = Error logs sizes set in Event Viewer

maxthreads = **MaxCmds**

maxwrkcache = Set by Windows NT cache manager

numalerts = Not applicable in Windows NT

numbcharbuf = Windows NT allocates buffers as needed.

numdgrambuf = **MailslotBuffers**

numservices = No limit to number of services in Windows NT.

numworkbuf = No limit to number of work buffers in Windows NT

sesstimeout = **SessTimeout**

sizcharbuf = **SizCharBuf**

sizerror = Self-configuring in Windows NT

sizworkbuf = No limit to the size of workstation buffers in Windows NT except as specified by server.

wrkservices = Use Services option in Control Panel to set for Windows NT

LAN Manager workstation heuristics =

CurrentControlSet\Services\LanmanWorkstation\Parameters

0 = **UseOpportunisticLocking**

5 = **UseLockReadUnlock**

11 = **UseRawRead** and **UseRawWrite**

19 = **Use512ByteMaxTransfer**

NetBEUI Protocol

LANMAN.INI [netbeui] = CurrentControlSet\Services\Nbf\Parameters

dlcretries = **LLCRetries**

localring = **QueryWithoutSourceRouting**

ncbs = **Size**

netbiosretries = **AddNameQueryRetries, GeneralRetries, NameQueryRetries**

netbiostimeout = **AddNameQueryTimeout, GeneralTimeout, NameQueryTimeout**

sessions = **Size**

t1 = **DefaultT1Timeout**

t2 = **DefaultT2Timeout**

ti = **DefaultTiTimeout**

TCP/IP

LANMAN.INI [tcpip] = CurrentControlSet\Services\adapter_name#Parameters\Tcpip

bcastaddr = **BroadcastType**

ipaddress0 = **IpAddress0**

scope = **ScopeID** (under \Services\NBT\Parameters)

subnetmask0 = **Subnetmask0**

LANMAN.INI [tcpip] = CurrentControlSet\Services \TCPIP\Parameters

defaultgateway = **DefaultGateway**

Lan Manager vs. Windows NT Settings for Adapter Cards

In LAN Manager, these entries are found in [adaptercard] section of the PROTOCOL.INI file.

In Windows NT, these entries are in the Registry in this path:

HKEY_LOCAL_MACHINE\SYSTEM

\CurrentControlSet\Services\adapter_name#Parameters

However, you can change most of settings for network adapter cards by using the Network option in Control Panel.

3Com Etherlink II

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter_name#Parameters

drivename = **Drivename**

interrupt = **IRQ**

ioaddress = **IoBaseAddress**

maxtransmits = N / A

netaddress = **NetworkAddress**

transceiver = **Transceiver**

xmitbufs = N / A

3Com EtherLink 16

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter_name#Parameters

drivename = **DriverName**

iobase = **IoBaseAddress**

netaddress = **NetworkAddress**

Compaq 32-Bit DualSpeed Token Ring

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter_name#Parameters

drivename = **Drivename**

earlyrelease = **Earlyrelease**

maxframesize = **MaxFramesize**

maxreceives = N / A

maxrequests = **MaxInternalReqs**

maxtransmits = N / A

netaddress = **NetworkAddress**
product_id = **ProductId**

DEC DEPCA

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter name#Parameters

drivername = **DriverName**
maxmulticast = **MaxMulticastList**
maxtransmits = N / A

IBM Token-Ring Adapters

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter name#Parameters

drivername = **DriverName**
earlyrelease = **Earlyrelease**
maxtransmits = N / A
netaddress = **NetworkAddress**
primaryor alternate = **IObaseAddress**
product_id = N / A
ram (Adapter and Adapter II only) = N / A
recvbuffers = N / A
recvbuffer size = N / A
xmitbuffers = N / A
xmitbuffer size = N / A

Intel EtherExpress 16

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter name#Parameters

drivername = **Drivername**
ioaddress = **IObaseAddress**

Novell NE2000

LANMAN.INI [adapterCard] = CurrentControlSet\Services\adapter name#Parameters

drivername = **Drivername**
interrupt = **IRQ**
iobase = **IOBaseAddress**

Device Driver and Service Entries

Scroll to see more entries. Click an entry to go to the topic for that entry.



[List of Software and User Preference Entries](#)



[List of Key Names](#)

[AarpRetries](#)

[AckDelayTime](#)

[AckWindow](#)

[AckWindowThreshold](#)

[AdapterName](#)

[AddNameQueryRetries](#)

[AddNameQueryTimeout](#)

[AlertNames](#)

[AlertSched](#)

[AllowAnonymous](#)

[AnnDelta](#)

[AnnotateDirectories](#)

[Announce](#)

[AnonymousOnly](#)

[AnonymousUsername](#)

[ArpCacheLife](#)

[ArpCacheSize](#)

[AuthenticateRetries](#)

[AuthenticateTime](#)

[AutoDisconnect](#)

[BatteryLife](#)

[Bind](#)

[Bindable](#)

[Bindform](#)

[BindSap](#)

[BootExecute](#)

[BreakPointOnEntry](#)

[BroadcastCount](#)

[BroadcastTimeout](#)

[BroadcastType](#)

[BrowserDomainDeletionThreshold](#)

[BrowserServerDeletionThreshold](#)

[BufFilesDenyWrite](#)

[BufNamedPipes](#)

[BufReadOnlyFiles](#)

[BusNumber](#)

[BusType](#)

[CableType](#)

CacheFileTimeout
CacheHitLimit
CacheResponseSize
CallbackTime
CardSpeed
CardType
CategoryCount
CategoryMessageFile
CharWait
Class (NetRules)
Cmdline
CodePageID
CollectionTime
CommandFile
Comment
ComponentInformation
ComSpec
ConfigurationData
Configuration Error (Sound)
ConnectionCount (NWNBLink for Novell)
ConnectionCount (NWLink for Global IPX/SPX)
ConnectionTimeout (NWNBLink for Novell)
ConnectionTimeout (NWLink for Global IPX/SPX)
ConnectionTimeout
ConnectMultiplePorts (Mouse)
ConnectMultiplePorts (Keyboard)
ConnectTimeout
CrashDir
CriticalSectionTimeout
Current
CurrentUser
DatabasePath
DdpChecksums
DebugFlags (FTP)
Default
DefaultGateway
DefaultPort
DefaultSettings.BitsPerPel
DefaultSettings.Interlaced
DefaultSettings.VRefresh
DefaultSettings.XResolution
DefaultSettings.YResolution
DefaultTxTimeout
DefaultZone

DependOnGroup
DependOnService
DesiredZone
DeviceData
Directory
DisableDisconnects
DisableMcastFwdWhenSessionTraffic
DisableMultipleRequests
DisablePort (Parallel)
DisablePort (Serial)
DisableSynchronousTransfers
DisableTaggedQueueing
Disc
DiskSpaceThreshold
DMACchannel (Netcard)
DmaChannel (Sound)
DNSLookupOrder
DormantFileLimit
DosDevices (Parallel)
DosDevices (Serial)
driverName
DriverParameter
DSP Version
EnableAudit
EnableBroadcast
EnableFCBopens
EnableFuncaddr
EnableNetbiosSessionsAuditing
EnableOplockForceClose
EnableOplocks
EnablePiggyBackAck
EnableRaw
EnableRouter
EnableSoftCompat
ErrorControl
ErrorThreshold
EventMessageFile
ExitMessage
Export (Linkage)
ExportList (Replicator)
ExportPath (Replicator)
Extensions
Failed
File

FindMasterTimeout
FirmwareRevision
FirstMessageDelay
ForceFifoEnable
ForwardBroadcasts
GeneralRetries
GeneralTimeout
GetBrowserListThreshold
GlobalFlag
GreetingMessage
Group
GuardTime
HelperDllName
Hidden (NetRules)
Hidden (Server service)
HomeDirectory
HzMode
Identifier (Hardware Description)
Identifier (DeviceMap)
IllegalDatagramResetTime
ImagePath (BootVerrificationProgram)
ImagePath (Services)
ImportList
ImportPath
Indexed
InitAddresses
InitAddressFiles
InitConnections
InitConnTable
InitFileTable
InitialRetransmissionTime
InitiatorTargetId
InitLinks
InitReceiveBuffers
InitReceivePackets
InitRequests
InitSearchTable
InitSendPackets
InitSessTable
InitUIFrames
InitWorkItems
InstalledDisplayDrivers
Interface
Internet

Interrupt (Parallel)
Interrupt (Serial)
Interrupt (Sound)
InterruptNumber (Adapter card)
InterruptStatus(Multiport serial)
Interval
IoBaseAddress
IoPageLockLimit
IPAddress
IpEnableRouter
IpReassemblyTimeout
IRPStackSize (Windows Sockets)
IRPStackSize (Server service)
IsDomainMasterBrowser
KeepAlive
KeepAliveCount (NWNBLink for Novel)
KeepAliveCount (NWLink for Global IPX/SPX)
KeepAliveTimeout (NWNBLink for Novel)
KeepAliveTimeout (NWLink for Global IPX/SPX)
KeepConn
KeyboardDataQueueSize (i8042)
KeyboardDataQueueSize (keyboard)
KeyboardDeviceBaseName (i8042)
KeyboardDeviceBaseName (keyboard)
KeyboardLayoutName
KnownDLLs
LanguageID
LargeSystemCache
LastKnownGood
LeftLineInAtten
LinkInfoValidTime
List
LLCMaxWindowSize
LLCRetries
LockIncrement
LockMaximum
LockQuota
LogAnonymous
LogElectionPackets
Logging
LoginMsg
LogNonAnonymous
LowerSearchBufferSize
LowerSearchThreshold

LPT_timeout
MailslotBuffers
MailslotDatagramThreshold
MaintainServerList
Mapping
MaxAddresses
MaxAddressFiles
MaxBcastDgBuffered
MaxClientsMessage (FTP)
MaxCmds
MaxCollectionCount
MaxConnections (NBF)
MaxConnections (TCP/IP)
MaxDgBufferedPerGroupName
MaxDynMem
MaxFrameSize
MaxFreeConnections
MaximumIncomingFrames
MaximumLogicalUnit
MaximumPacketSize
MaximumPortsServiced (Mouse)
MaximumPortsServiced (Keyboard)
MaxKeepSearch
MaxLinkDelay
MaxLinks
MaxMemoryUsage
MaxMpxCt
MaxNames
MaxNonpagedMemoryUsage
MaxPagedMemoryUsage
MaxPktSize
MaxPreload
MaxRequests
MaxSessions (MacFile)
MaxSessions (RAS)
MaxSize
MaxSockAddrLen
MaxWorkItemIdleTime
MaxWorkItems
MediaType
MemoryMapped
MemoryMappedBaseAddress
MessageInterval
MinFreeConnections

MinFreeWorkItems
MinKeepSearch
MinLinkThroughput
MinRcvQueue
MinSockAddrLen
Monitor
MouseDataQueueSize (bus mouse)
MouseDataQueueSize (i8042)
MouseDataQueueSize (InPort mouse)
MouseDataQueueSize (serial mouse)
MouseDataQueueSize (mouse class driver)
MouseResolution
MouseSynchIn100ns
MsdosDirOutput
MTU
MultiCastForwardRate
NameQueryRetries
NameQueryTimeout
NameServer
NbProvider
NbtKeepAlive
NetBIOSGatewayEnabled
NetworkAddress
NetworkAddress
NetworkErrorThreshold
NetworkNumber
NetworkRangeLowerEnd
NetworkRangeUpperEnd
NonPagedMemLimit
NonPagedPoolSize
NumberOfButtons (bus mouse)
NumberOfButtons (i8042)
NumberOfButtons (InPort mouse)
NumberOfButtons (serial mouse)
NumberOfCylinders
NumberOfheads
NumBlockThreads
NumIllegalDatagramEvents
NumRecvQueryIndications
ObjectDirectories
ObjectName
OEMlocale
OpenSearch
OplockBreakWait

Optional
Options
Os2
Os2LibPath
OtherDomains
OverrideHardwareBitstring
OverrideKeyboardSubtype
OverrideKeyboardType
PagedMemLimit
PagedPoolSize
PagingFiles
ParallelX
Path
PermanentName
PipeIncrement
PipeMaximum
PktType
PointerDeviceBaseName (bus mouse)
PointerDeviceBaseName (i8042)
PointerDeviceBaseName (InPort mouse)
PointerDeviceBaseName (serial mouse)
PointerDeviceBaseName (mouse class driver)
PollingIterations
PollingIterationsMaximum
PollStatusIterations
Port (UPS)
Port (Sound)
PortAddress (Parallel)
PortAddress (Serial)
PortIndex
PortName
Posix
ProviderName
ProviderOrder
ProviderPath
Pulse
PulseInterval
QueryDriverFrequency
QueryWithoutSourceRouting
Random
Randomize
RawWorkItems
RcvDgSubmittedPerGroupName
RcvWindowMax

ReadAccessMask
ReadAheadThroughput
RechargeRate
RegistrySizeLimit
RemoteListen
Replicate
Required
ResendIterations
Retention
RetransmitMax
Review
RightLineInAtten
RipAgeTime
RipCount
RipTimeout
RipUsageTime
Route
RouterMTU
SampleRate (bus mouse)
SampleRate (i8042)
SampleRate (InPort mouse)
SampleRate (serial mouse)
ScavQosInfoUpdateTime
ScavTimeout
ScopeID
Scripts
ScsiDebug
SearchList
Sectorspertrack
SeedingNetwork
SerialX
ServerAnnounceBuffers
ServerName
ServerOptions
SessConns
SessOpens
SessTimeout
SessUsers
SizCharBuf
Size
SizReqBuf
SizWorkBufs
SourceRouteBcast
SourceRouteDef

SourceRouteMcast
SourceRouteUsageTime
SourceRouting
Sources
StackSize
Start
SubnetMask
Swap
SystemDirectory
SystemStartOptions
TxTicky
Tag
TcpDisableReceiveChecksum
TcpDisableSendChecksum
TcpKeepCnt
TcpKeepTries
TcpLogLevel
TcpMaxConnectAttempts
TcpMaxRetransmissionAttempts
TcpNumConnections
TcpRecvSegmentSize
TcpSendDownMax
TcpSendSegmentSize
TcpWindowSize
ThreadCountAdd
ThreadPriority
Trailers
Transceiver
Transports (Windows Sockets)
Transports (Workstation service)
Type (Services)
Type (NetRules)
TypesSupported
UdpDisableReceiveChecksum
UdpDisableSendChecksum
UdpNumConnections
Update
UpperSearchBufferSize
Use
Use512ByteMaxTransfer
UseAsyncWriteBehind
UseDixOverEthernet
UseDixOverEthernet
UseLockReadUnlock

UseOpportunisticLocking

UseRawRead

UseRawWrite

Users

UseUnlockBehind

UseWriteBehind

UseWriteRawData

UtilizeNtCaching

VDD

VgaCompatible

VolumeName (MacFile)

WanNameQueryRetries

Windir

Windows

WindowSize

Wowcmdline

Wowsize

WriteAccessMask

XactMemSize

ZoneList

Software and User Preference Entries

Scroll to see more entries. Click an entry to go to the topic for that entry.



[List of Device Driver and Services Entries](#)



[List of Key Names](#)

[Active](#)

[AddOns](#)

[AppInit_DLLs](#)

[AppointmentView](#)

[AppPath](#)

[ApptBookColor](#)

[ApptBookLinesColor](#)

[AutoArrange](#)

[AutoAdminLogon](#)

[BorderWidth](#)

[CheckLatencyInterval](#)

[ConfigurationFile](#)

[Confirmation](#)

[ConfirmDelete](#)

[ConfirmFormat](#)

[ConfirmMouse](#)

[ConfirmReplace](#)

[ConfirmSubDel](#)

[ConfirmSystemHiddenReadOnly](#)

[ConnectionType](#)

[CoolSwitch](#)

[CopyTime](#)

[Count \(Schedule+ ApptBooks\)](#)

[Count \(Schedule+ Archives\)](#)

[CreateFileFirstTime](#)

[CurrentBuild, CurrentType, and CurrentVersion](#)

[CursorBlinkRate](#)

[CustomDict](#)

DataFile

DefaultDomainName

DefaultPassword

DefaultPrinter

DefaultRemindAgain

DefaultRemindAgainAmount

DefaultRemindAgainUnits

DefaultUserName

DemosEnabled (Mail)

DemosEnabled (Schedule+)

DeviceNotSelectedTimeout

dir1

display.drv

DoubleClickSpeed

DriveMappingLetter

Driver (Print Environments)

Driver (Print Monitors)

DriverDescription (Fonts)

EditLevel

ExpandLogonDomain

ExportMmfFile

ExportNoNotes

ExportRange

ExportType

Face

FaceWeight

FIXEDFON.FON

FixedFont

FontName

FONT.S.FON

ForceScanInterval

GALOnly

GridGranularity

IconSpacing

IconTitleFaceName

IconTitleSize

IconTitleStyle

IconTitleWrap

iCountry

iCurrDigits

iCurrency

iDate

iDigits

IdleRequiredInterval

iLZero

iMeasure

ImportDoNotAddDuplicates

ImportDoNotAskAboutConflicts

ImportType

iNegCurr

InitialKeyboardIndicators

InstallDate

iTime

iTLZero

Kb_Free_Start_Compress

Kb_Free_Stop_Compress

KeyboardDelay

KeyboardSpeed

LargeFont

LegalNoticeCaption

LegalNoticeText

Locale

LocalMMF

LocalPath

LocalUser

Login

Logon

LowerCase

MainWindow

MaxSize

MigrateIni (Mail)

MigrateIni (Schedule+)

MigrateIniPrint (Mail)

MigrateIniPrint (Schedule+)

MinIncrSize

MinOnRun

MouseSpeed

MouseThreshold1

MouseThreshold2

Multi-Message

Name

Network

NewMsgsAtStartup

No_Compress

NoClose

NoFileMenu

NormalFont

NoRun

NoSaveSettings

NoServerOptions

NoStatusBar

NumButtons

OEMFONT.FON

OfflineMessages

OldStorePath

Order (Persistent Connections)

Order (Program Manager)

OtherColor

PageBackgroundColor

Password

PathName

Pattern

Percent_Free_Start_Compress

Percent_Free_Stop_Compress

PlannerColor

PlannerLinesColor

PollingInterval

PollTime

Printer

PrintManager

ProfileImagePath

ProviderName

PumpCycleInterval

RegisteredOrganization and RegisteredOwner

ReminderPollTime

RemotePath

ReportBootOk

RequestSummary

Restrictions

s1159

s2359

SaveConnections

SaveSettings (Network software)

SaveSettings (Progarm Manager)

SaveSettings Printing)

ScanAgainInterval

sCountry

ScreenSaveActive

ScreenSaverIsSecure

ScreenSaveTimeOut

SCRNSAVE.EXE

sCurrency

sDate

sDecimal

Secs_Till_Fast_Compress

Security

ServerPassword

ServerPath

SharedExtensionsDir

SharedFolders (Mail message file)

SharedFolders (Mail providers)

Shell

ShowActiveTasks

ShowCommonGroups

Sid# (ProfileImage)

Size

sLanguage

sList

sLongDate

Slow Mode

SoftwareType

SortOrder

SourcePath

Spelling

SpellingNNNN,0

Spooler

SpoolerBackoffInterval

SpoolerReconnectInterval

sShortDate

StartupOffline

sThousand

sTime

StripGatewayHeaders

SwapDisk

SwapMouseButtons

System

SystemRoot

TaskSortOrder

TaskSortSecond

TileWallpaper

ToolbarWindow

TransmissionRetryTimeout

Transport

TTEnable

TTonly

UpdatePostOfficeTime

UserColor

Userinit

UserName

Version

ViewNotByProject

Wallpaper

WG

Window (Mail)

Window (File Manager)

Window (Network software)

Window (Program Manager)

WindowOrder (Schedule+)

List of Key Names

Scroll to see more names. Click a name to go to the topic for that key name.



List of Software and User Preference Entries



List of Device Driver and Services Entries

AccessCheck

Address Book

Alerter

AppleTalk

Applications

AsyncMacn

AtDisk

Boot

Boot.description

BootVerification

BootVerificationProgram

Browser

Busmouse

Classes

CodePage

Color Schemes

Colors

Compatibility

Console

Control (CurrentControlSet)

Cursors

Custom Commands

Custom Menus

Custom Messages

Description (Adapters)

Description (Hardware)

Desktop

DeviceMap

DGRcvr

Disabled

DiskPerf

DLC

DosKeybCodes

Environment (System)

Environment (User)

EventLog

Extensions

File Manager

File system drivers in Services key

Font

Font Drivers

FontCache

FontSubstitutes

Ftpsvc

GRE_Initialize

GroupOrderList

Groups

H_WindowPosition

Hardware

HiveList

i8042prt

IFJumpColor

IFPopupColor

IniFileMapping

Inport

International

JumpColor

Kbdclass

Keyboard

Keyboard

Keyboard Layout

KeyboardClass

KeyboardLayout

KeyboardPort

KnownDLLs

Language

LanmanServer

LanmanWorkstation

Linkage

M_WindowPosition

Mac FileTypes

MacFile

MacroColor

Mail

Maximized

Memory Management

Microsoft Schedule+

Microsoft Schedule+ ApptBooks

Microsoft Schedule+ Archives

Microsoft Schedule+ Exportes

Microsoft Schedule+ Importes

MicrosoftMail

MMF

Mouclass

Mouse

MS Proofing Tools

NBF

NBLinkIPX

NBNWLink

NBT

NCPA

NetbiosGateway

NetConfig

NetLogon

Network

Network adapter cards in Services key

NetworkProvider (Services)

NetworkProvider\Order (Control)

Nls\CodePage

Nls\Language

NonWindowsApp

OEMlocale

OS/2 Subsystem for NT

Parallel

Parallel Ports

Patterns

Persistent Connections

PointerClass

PointerPort

PopupColor

ProfileList

Program Groups

ProgramManager

Providers (Mail)

Randomize

RasHub

RasMan

Rdr

RemoteAccess

Replicator

ResourceMap

SAM

Schedule (Services)

Schedule+

Screen Saver

Scripts

Scsi device

Secure

Security

Select (CurrentControlSet

Serial

Serialcomm

Serialx

Sermouse

ServiceGroupOrder

Services

SessionManager

Settings (File Manager)

Settings (Program Manager)

Shared Parameters

Sid#

Sndblst

Software

Spelling

Standard

Static

Streams

Substitutes

Subsystems

Tcpip

TrueType

TTonly

Update

UPS

VGA

Video (DeviceMap)

Video Driver (Services)

Videoprt

VirtualDeviceDrivers

Windows (Startup)

[Windows \(Software Registration\)](#)

[Windows 3.1 Migration Status](#)

[Windows Help](#)

[Windows NT](#)

[WinLogon](#)

[Winsock](#)

[World Full Access Shared Parameters](#)

[WOW \(Startup\)](#)

[WOW \(Software Registration\)](#)

[Xi](#)

[Xr](#)

[Yd](#)

[Yu](#)

Windows 3.1 .INI Settings in Windows NT

If you install Windows NT as an upgrade over Windows 3.1, all the settings from various initialization files are copied into the Registry, including CONTROL.INI, PROGMAN.INI, SYSTEM.INI, WIN.INI, WINFILE.INI, and others. You can see where the Windows initialization files are mapped in the Registry by viewing the subkeys and value entries under this path:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Microsoft
      \Windows NT
        \CurrentVersion
          \IniFileMapping
```

When you install an application created for 16-bit Microsoft Windows, the application's setup program creates its own .INI file or creates entries for the WIN.INI or SYSTEM.INI file in the same way that it does for any versions of Windows for MSDOS. These entries are not updated in the Registry, because these applications do not know how to access the Windows NT Registry. For this reason, basic SYSTEM.INI, WIN.INI, and WINFILE.INI files appear in the *SystemRoot* directory in Windows NT.

If a Windows-based application tries to write to WIN.INI, SYSTEM.INI, or any other section listed in the IniFileMapping key, and if the application uses the Windows NT Registry APIs, the information is stored in the Registry. If the application writes to other sections of the .INI file or tries to open the .INI file directly without using the Windows NT Registry APIs, the information is saved in an .INI file.

To find mapping information in the HKEY_LOCAL_MACHINE\Software key, the system looks up the *filename.ext* of the initialization file. If a match is found, it looks under the mapped key for the specific application name and a variable name, and if necessary it continues to look for keys whose value entries are the variable names. If no mapping for either the application name or filename is found, the system looks for an .INI file to read and write its contents.

In the entries in the IniFileMapping key, the following symbols are used:

- ! Forces all writes to go to both the Registry and to the .INI file on disk.
- # Causes the Registry value to be set to the value in the Windows 3.1 .INI file whenever a new user logs in for the first time after Setup, if Windows NT was installed on a computer that had Windows 3.1 already installed.
- @ Prevents any reads from going to the .INI file on disk if the requested data is not found in the Registry.

USR Stands for HKEY_CURRENT_USER, and the text after the prefix is relative to that key.

SYS Stands for HKEY_LOCAL_MACHINE\SOFTWARE, and the text after the prefix is relative to that key.



Copyright Information

Information in this online document is subject to change without notice.

Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Microsoft Corporation.

© 1985-1993 Microsoft Corporation. All rights reserved.

Microsoft is a registered trademark and Windows NT is a trademark of Microsoft Corporation.

