

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

One of the most mysterious parts of Windows NT is the Registry. Often, even experienced NT users and administrators have only a vague notion of what it stores and how it organizes data. That users must view the Registry's contents through NT's Registry editors (Regedt32 and Regedit) supplied with NT does not make the Registry's mysteriousness any less surprising. The Registry is large, and its organization often seems to verge on the haphazard.

Knowing where the Registry displays different types of information makes the Registry less intimidating. Let's start with an overview of its structure and then look at the specific values each of its major data branches contains. (I won't talk about just documented or undocumented keys, or describe only data that you can't access from system administration utilities, because such restrictions lead to notions that the Registry is somehow more mysterious than it really is.) I'll conclude with a brief section that provides some useful Registry settings. For information about NT's Registry editors and how to back up the Registry, see Christa Anderson, "Care and Feeding of the Registry," December 1996.

The Registry's Structure

This section introduces the Registry, so if you're already familiar with the Registry, skip ahead to the next section. Because the Registry is a database, its structure is much like that of a logical disk volume. The Registry contains keys, which are similar to a disk's directories, and values, which compare to files on a disk. A key is a container that can consist of other keys (subkeys) or values. Values, on the other hand, store data. Top-level keys are root keys. Throughout the article, I'll use subkey and key interchangeably (only the root keys are not subkeys).

Both keys and values borrow their naming convention from the file system. Thus, you can uniquely identify a value with the name mark, which is stored in a key called trade, with the name trade\mark. One exception to this naming scheme is each key's unnamed value. Regedit displays the unnamed value as Default; Regedt32 uses <No Name>.

Values store different kinds of data and can be one of the 11 types listed in Table 1. The majority of Registry values are either REG_DWORD, REG_BINARY, or REG_SZ. Values of type REG_DWORD can store numbers or Booleans (on/off values); REG_BINARY values can store numbers larger than 32 bits, or raw data such as encrypted passwords; REG_SZ values store strings (Unicode, of course) that can represent names, filenames, paths, and types.

TABLE 1: Registry Value Types

Value Type	Description
REG_NONE	No value type
REG_SZ	Unicode NULL terminated string
REG_EXPAND_SZ	Unicode NULL terminated string that can have embedded environment variables
REG_BINARY	Arbitrary length binary data
REG_DWORD	32-bit number
REG_DWORD_BIG_ENDIAN	32-bit number, high byte first
REG_LINK	Unicode symbolic link
REG_MULTI_SZ	Array of Unicode strings

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

REG_RESOURCE_LIST	Hardware resource description
REG_FULL_RESOURCE_DESCRIPTOR	Hardware resource description
REG_RESOURCE_REQUIREMENTS_LIST	Resource requirements

The REG_LINK type is particularly interesting because it lets a value transparently point at another key or value. When you traverse the Registry through a link, the path searching continues at the target of the link. For example, if \Root1\Link has a REG_LINK value of \Root2\RegKey and RegKey contains the value RegValue, two paths identify RegValue: \Root1\Link\RegValue and \Root2\RegKey\RegValue. NT prominently uses Registry links: Three of the six Registry root keys, listed in Table 2, are links to subkeys within the three non-link root keys.

TABLE 2: Registry Root Keys		
Root Key	Description	Link
HKEY_CURRENT_USER	Information related to currently logged-on user	Subkey under HKEY_USERS corresponding to currently logged-on user
HKEY_USERS	Contains subkeys for all local user accounts	Not a link
HKEY_CLASSES_ROOT	Contains file association and OLE registration information	HKEY_LOCAL_MACHINE\SOFTWARE\Classes
HKEY_LOCAL_MACHINE	All static and dynamic system configuration information	Not a link
HKEY_DYN_DATA	Performance counters	Not a link
HKEY_CURRENT_CONFIG	Information about hardware configuration	HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Hardware Profiles\Current

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

Registry Root Keys

You can chart the organization of the Registry via the data stored within it. The six root keys divide the Registry data into categories. (You cannot add new root keys or delete existing ones.) Data associated with the currently logged-on user (HKEY_CURRENT_USER), information about all the accounts on the machine (HKEY_USERS), file association and Object Linking and Embedding (OLE) registration information (HKEY_CLASSES_ROOT), system-related information (HKEY_LOCAL_MACHINE), performance data (HKEY_DYN_DATA), and some information about the current hardware profile (HKEY_CURRENT_CONFIG) comprise the six data categories.

Why do root key names begin with an H? The root key names represent Win32 handles (H) to keys (KEY). Throughout the rest of the article, I'll abbreviate the root key names. For example, HKLM represents HKEY_LOCAL_MACHINE. Table 3 lists the root key names and corresponding abbreviations.

TABLE 3: Registry Root Key Abbreviations

Root Key	Abbreviation
HKEY_CURRENT_USER	HKCU
HKEY_USERS	HKU
HKEY_CLASSES_ROOT	HKCR
HKEY_LOCAL_MACHINE	HKLM
HKEY_DYN_DATA	HKDD
HKEY_CURRENT_CONFIG	HKCC

HKEY_CURRENT_USER

The HKCU root key contains data regarding the preferences and software configuration of the locally logged-on user. Within HKCU, you find the 10 subkeys shown in Table 4. Whenever a user logs on, HKCU is created as a link to the user's key under HKEY_USERS.

TABLE 4: HKEY_CURRENT_USER Subkeys

Subkey	Description
AppEvents	Sound/event associations
Console	Command window shortcut settings (e.g., width, height, colors)
Control Panel	Screen saver, desktop scheme, keyboard, and mouse settings
Environment	Environment variable definitions
Keyboard Layout	Keyboard layout setting (e.g., US, UK)
Network	Network drive mappings and settings
Printers	Printer connection settings
Software	User-specific software preferences
UNICODE Program Groups	User-specific start menu group definitions

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

Windows 3.1 Migration Status	File status data for systems that upgrade from Windows 3.x to NT 4.0
------------------------------	--

HKCU\AppEvents contains two subkeys: EventLabels, where you find event names (e.g., mail arrival, window minimize), and Schemes, where you find sound and event associations. Under Schemes\Apps you find groups of event keys whose values can point at wave files. You can easily change these settings via the Control Panel Sounds applet.

HKCU\Console contains a subkey for each Command Prompt shortcut in the user's account. Under these subkeys, you find all the properties (e.g., foreground and background text colors, window size, edit mode) for the command window that's created when you execute a particular shortcut. You can access all these values through the Properties menu item on the individual command prompt windows. HKCU\Control Panel contains GUI settings such as desktop and screen-saver parameters, cursor file associations, and window attributes such as colors and geometry settings. As the key's name suggests, you can edit most of these values through Control Panel applets; however, you must edit some values via a Registry editor. For example, to make the window focus follow the mouse, you must set the value of HKCU\ControlPanel\Mouse\ActiveWindowTracking to 1 (and reboot for the change to take effect). To tell Windows how long (in milliseconds) to pause before it displays the cascading menus of the Start menu, you must edit HKCU\Control Panel\Desktop\MenuShowDelay.

You find environment variable definitions in HKCU\Environment. You can change these definitions with the Control Panel System applet under the Environment tab.

HKCU\Network and HKCU\Printers contain network drive-letter mapping information and printer connection data, respectively. You can set these values through Explorer, File Manager, and the Control Panel Printers applet.

The heftiest subkey under HKCU is Software. Most applications create subkeys under HKCU\Software that consist of the vendor's name (e.g., Microsoft) and contain subkeys for the vendor's applications (e.g., Windows NT). Subkeys and values within the application keys are where programs locate user-dependent information, such as most recently used (MRU) menu items, appearance characteristics, and usage preferences.

The HKCU\UNICODE Program Groups and HKCU\Windows 3.1 Migration Status subkeys contain upgrade information if you've upgraded the system from a previous version of NT or from Windows 3.x. NT 4.0 does not use the UNICODE Program Groups subkey, and the subkey doesn't contain any information if you've never installed a previous version of NT. Upgraded machines may display obsolete program group data under this subkey. The Windows 3.1 Migration Status subkey contains information about whether Windows 3.x .grp and .ini files have been converted to NT 4.0 format.

HKEY_USERS

HKU contains a subkey for each user who has a local account on the system, as I alluded to in the description of HKCU. The .DEFAULT subkey contains the HKCU settings that the system account uses. They are in effect when the logon box appears. The other user subkeys are named with the Security Identifier (SID) of the user's account that they serve.

HKEY_CLASSES_ROOT

The HKCR root key first appeared in the Windows 3.1 Registry; Microsoft migrated HKCR to the NT 4.0 Registry for compatibility purposes. HKCR consists of two types of information: file extension associations and OLE class

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

registrations. A key exists for every registered filename extension. Most keys contain a REG_SZ value that points at another key in HKCR containing the association information for the class of files that extension represents. For example, if you install Microsoft Word, the .doc subkey has an unnamed value, "Word.Document.6". If you look at the Word.Document.6 subkey, you find an unnamed value that describes the file type (which Explorer's file-association window uses) and keys that associate that type of files to icons (DefaultIcon); other keys specify dynamic data exchange (DDE) commands created whenever you open, create, or print Word.Document.6 files. Keys without defined unnamed values have DDE command information stored in subkeys.

HKCR keys such as Word.Document.6 also contain OLE registration information. That way, OLE client applications can look up and establish communication with OLE server applications to support functionality such as inserting an Excel spreadsheet into a Word document. CLSID subkeys store registration numbers as very long representations of OLE registration identifiers.

HKEY_LOCAL_MACHINE

HKLM is the most interesting but often least understood root key of the Registry--HKLM contains an incredible amount of unrelated information grouped under five subkeys: HARDWARE, SAM, SECURITY, SOFTWARE, and SYSTEM.

The HKLM\HARDWARE subkey maintains descriptions of the system's hardware and all hardware device-to-driver mappings. NTDETECT on x86 machines, or ARC firmware on RISC machines, collects information on the system's hardware characteristics as the machine boots. NTDETECT or ARC passes this information on to NT once NT's image has been started. NT then stores this information in the HKLM\HARDWARE\DESCRIPTION subkey. As device drivers start up and claim devices, they inform NT so that it can associate devices with the drivers that control them. NT places this mapping data in the HKLM\HARDWARE\DEVICEMAP subkey. Serving a similar purpose, HKLM\HARDWARE\OWNERMAP associates the system's buses (e.g., PCI and ISA) to drivers that control them. Finally, device drivers inform NT of system resources that they claim for their devices. Such resources include port addresses, physical memory ranges, and interrupt numbers. NT keeps track of this information in the HKLM\HARDWARE\RESOURCEMAP subkey to prevent conflicts. Windows NT Diagnostics (Winmsdp.exe) lets you view Registry hardware information that it obtains by simply reading values out of the HARDWARE key.

HKLM\SAM holds local and domain account information, such as user passwords, group definitions, and domain associations. By default, this key is unreadable by even the system administrator account. Looking inside HKLM\SAM is not very revealing because the data is undocumented and the passwords are encrypted with a one-way mapping (e.g., you cannot determine a password from its encrypted form).

HKLM\SECURITY stores user and group policies. Examples of policies include whether a particular user is allowed to reboot the machine, load device drivers, back up files, or access the system remotely. SECURITY's information is also encrypted. HKLM\SAM is linked into the SECURITY subkey under HKLM\SECURITY\SAM.

Like HKCU\Software, applications use HKLM\SOFTWARE to store private settings. HKLM\SOFTWARE uses the same naming convention I described for HKCU\Software, but the type of data stored is usually different. Because the HKLM root key is the same for all users who log on, it serves as a repository for system-wide program settings. The information usually includes paths to application files and directories and licensing, and expiration date information.

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

One particularly interesting subkey is HKLM\SOFTWARE\Microsoft\Windows NT\Current Version. Here you can find the NT build number, whether the version is uniprocessor or multiprocessor, and the system root directory path. If you installed a service pack, its name appears in CSDVersion. Current Version has a useful subkey: Winlogon. By modifying entries in Winlogon (I'll describe how to modify entries at the end of the article), you can set up the system to automatically log on a user whenever the system boots.

Another HKLM\SOFTWARE subkey is Windows\Current Version. This key is a Windows 95-compatibility key that contains system software parameters. For example, the Explorer key includes information about desktop name-space extensions such as Network Neighborhood and My Computer. Applications put pointers to their uninstall programs in the Uninstall key. And AppPaths is where NT stores the paths of applications it knows about. Executing an AppPaths program from the Start menu's Run dialog box launches the program by looking at its hard-wired location.

NT's command central is under HKLM\SYSTEM. NT Setup creates the HKLM\SYSTEM\Setup subkey, which points subsequent invocations of Setup at the System's root partition. NT uses the Setup\SystemSetupInProgress value to determine whether to be in Setup or regular operation mode. Another subkey under HKLM\SYSTEM is DISK. It is present on only systems that have run NT's Disk Administrator program. HKLM\SYSTEM\DISK is where Disk Administrator stores information about drive letter mappings, volume sets, mirrored volumes, and striped sets.

HKLM\SYSTEM also contains two or more subkeys with the prefix ControlSet and another subkey called CurrentControlSet. NT links CurrentControlSet to the ControlSet subkey that corresponds to the profile the system used in the boot of the current session. The other ControlSet subkeys represent configurations such as Last Known Good Configuration, a copy of the last profile the system successfully booted with. You can look at the value Current under HKLM\SYSTEM>Select to find out which ControlSet subkey CurrentControlSet maps to. Other values under Select point at control sets associated with Last Known Good Configuration, and the control set that last resulted in a failed boot attempt.

Within HKLM\SYSTEM\CurrentControlSet are the four subkeys listed in Table 5. NT keeps its static configuration information in the Control subkey, which contains about 30 different subkeys of its own. One of Control's noteworthy subkeys is ComputerName, which displays the system's name under ActiveComputerName. Control\CrashControl is a handy subkey for device driver developers and systems administrators. It contains values that give NT directions for what to do when the machine goes down, including whether to produce a crash dump and whether to immediately reboot.

TABLE 5: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet Subkeys

Subkey	Description
Control	Static NT tuning and configuration parameters
Enum	Information collected when drivers and services are started
Hardware Profiles	Video-related configuration information
Services	Startup and error control for device drivers and services

Control\hivelist contains the paths to files where NT stores Registry information. Control\hivelist values point at the files for HKLM\SAM, HKLM\SECURITY, HKLM\SOFTWARE, HKLM\SYSTEM, HKU\DEFAULT, and individual user accounts.

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

Control\ProductOptions deserves mention: It's the subkey that contains the ProductType value, which identifies whether the system is a workstation ("WinNT") or a server ("ServerNT"). Microsoft applications check the ProductType value and adjust their behavior according to its setting (for more information about this Registry value, see "Inside the Difference Between Windows NT Workstation and Windows NT Server," November 1996).

Control\Session Manager contains a variety of interesting parameters. Values for this key include BootExecute, which can point at a program that will automatically execute early in the system boot, and LicensedProcessors, which is the number of processors that the system's license supports (two for NT Workstation and four for NT Server). NT uses only the number of licensed processors, even if the system has more.

The Control\Session Manager\Environment subkey contains system-level environment variables. The Control\Session Manager\SubSystems subkey keeps pointers to the files that the NT environment subsystems (Win32, WOW, OS/2, and Posix) use.

Control\Session Manager's Executive and Memory Management subkeys contain values for advanced system tuning. For instance, Executive holds values that can direct NT to create additional operating system worker threads. Another value stored there, PriorityQuantumMatrix, has an enticing name that implies the ability to fine-tune NT's scheduling algorithm, but the value actually stores encrypted NT beta and release candidate expiration dates. Memory Management holds memory subsystem tuning parameters. One setting, PagingFiles, directs the system to the location of the paging files; you can use other settings to override internal defaults that specify the amounts of memory set aside for various internal operations.

The final key I'll mention under HKLM\SYSTEM\CurrentControlSet\Control is WOW. It contains entries related to the execution of command windows, including the path to the command window executable, ntvdm.exe, in the cmdline value.

HKLM\SYSTEM\CurrentControlSet\Services is the control center for NT OS's dynamically added parts: Win32 services and kernel-mode device drivers. Every service and device driver that NT ships with support for and any service or driver that you install later has a key under Services. A Services subkey typically contains several values from the list shown in Table 6. A few Services subkeys allow a driver or service to control when it will be loaded in the NT boot sequence. The required Start value is the primary order controller. NT loads services and drivers in three phases, each of which corresponds to a particular Start definition. The first phase, Boot, occurs just after NTOSKRNL starts. At this time, the system loads only those drivers essential to NT's boot. The second phase, System, is when the system loads the majority of device drivers. The system is still in its text mode (blue screen) during this phase. The system initiates the third phase, Auto, about the time the Win32 subsystem starts. You can identify approximately when the Win32 services start by the appearance of the system logon dialog box.

TABLE 6: Services Values and Subkeys

Value	Description
DisplayName	Name shown in Control Panel's Services or Devices applets
ImagePath	Pathname of service or driver file if it's not in SystemRoot\System32\Drivers

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

ErrorControl	Indicates action NT will take if service or device reports an error when it starts
Start	0=Boot Start (core drivers) 1=System Start (at blue screen) 2=Auto Start (after Win32 starts approximately when logon prompt appears) 3=Demand Start 4=Disabled
Type	Kernel-mode driver, File System, Win32 Service, etc.
Group, Tag	Controls load ordering for Boot Start and System Start drivers
DependOnService, DependOnGroup	Controls load ordering for Auto Start Drivers
Parameters	Stores driver/service-dependent private settings

Developers use other Services subkey values (Group, Tag, DependOnService, DependOnGroup) to fine-tune the start location of a driver or service within a boot phase. They need these values when dependencies exist between drivers or services.

Drivers and services often have a Parameters subkey that contains private settings. For example, the Browser service's Parameters subkey is a value that denotes whether the browser is the domain master browser. The Parameters subkey of the Busmouse driver stores the number of buttons and its sample rate.

HKEY_DYN_DATA

HKDD is a fake key--it doesn't really exist. It serves as a convenient doorway to device driver, Win32 application, and native NT performance counters via the Registry API. When a Win32 program queries a value or key in HKDD, the request gets routed as an I/O request to the appropriate driver or Win32 program, which returns information that looks like the result of an authentic Registry access. The Performance Monitor (Perfmon) program accesses this root key to provide the intricate performance information it displays.

HKEY_CURRENT_CONFIG

HKCC, a new root key in NT 4.0, is a link to HKLM\SYSTEM\CurrentControlSet\Hardware Profiles\Current, which contains the configuration data for the hardware profile in use on the system. Microsoft added HKCC to NT to let applications that access this key run on both Windows 95 and NT. To create, configure, and change hardware profiles, you can use Control Panel's System, Services, and Device applets.

Registry Gems

Now that you understand the basic structure of the Registry, let's look at a few handy settings for data stored in the Registry. As always, before you try any of the following suggestions, back up your Registry. Editing Registry entries incorrectly can cause systemwide problems that may require you to reinstall NT to correct them.

Auto-logon. If you have a private NT system (e.g., a machine you use at home), you can configure NT to automatically log you on when you boot NT. To enable auto-logon, you must modify four values in HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon. First, set the AutoAdminLogon

Inside the Windows NT Registry

Mark Russinovich

(Reprinted from WindowsItPro Magazine)

value to 1. Then specify appropriate strings for DefaultDomainName, DefaultPassword, and DefaultUserName. The next time you reboot, you'll automatically be logged on.

Tuning a workstation for server-like workloads. NT Workstation and NT Server have vastly different performance characteristics because of the internal tuning NT performs. You cannot access most tuning parameters, but you can find a few in the Registry. If you run NT Server and double-click the Server entry of the Services tab in Control Panel's Network applet, you get a dialog box that lets you determine what type of applications you want to tune the machine for. You can choose among Minimize Memory Used, Balance, Maximize Throughput for File Sharing, and Maximize Throughput for Network Applications. Systems running NT Workstation do not present this dialog box. The options change two Registry values: HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\LargeSystemCache and HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters\Size. Table 7 lists the value data you need to enter in the Registry of a system running NT Workstation to achieve the same tuning options the NT Server dialog box offers.

TABLE 7: Tuning a Workstation

Target Tuning Option	LargeSystemCache	Size
Minimize Memory Used	0	1
Balance	0	2
Maximize Throughput for File Sharing	1	3
Maximize Throughput for Network Applications	0	3

Previewing bitmaps in their icons. How many times have you wished you could get a quick look at what's in a bitmap file without opening it? You can, with a simple Registry setting. Just change the value of HKCR\Paint.Picture\DefaultIcon to "%1". Reboot for the change to take effect.

After the Tour

These few tips conclude a whirlwind tour of the Registry. If you want to learn more about the Registry, get Microsoft Windows NT Workstation Resource Kit or Microsoft Windows NT Server Resource Kit, which include extensive online documentation about the Registry. If your interest lies in programming the Registry API, refer to the Win32 software development kit (SDK).