

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

Since its inception, Windows NT has been available in two forms: as a workstation operating system (Windows NT Workstation) and as a server operating system (Windows NT Server). Microsoft sells the two platforms with different licenses and different bundled software and portrays them as being suited for different workloads. The question then is what is different when you remove the software and licenses from consideration? Microsoft has stated publicly that Workstation and Server "share the same kernel architecture," and a recent article on Microsoft's Web site goes further, describing how Workstation and Server are tuned in a variety of ways to suit their distinct roles: Workstation is for desktop interactive use, and Server is for file, print, and Web serving. Given this information, the reasonable assumption is that Microsoft constructed Workstation and Server from different, although possibly in many respects shared, code bases. However, a binary comparison of the installation CDs reveals that the code base is the same.

This article will describe what files on the setup disks are different, and I'll then show that only a Registry key or two determines which type of runtime tuning the kernel and user-level applications will perform. A detailed study of the operating system and device driver tuning differences follows, and the article concludes with a tour of applications that check, for one reason or another, to see whether they are running on Server or on Workstation.

## File Differences

NT's core operating system and support components are on the setup CDs under directories that identify processor-specific versions. The NT 4.0 final release has about 2500 unique files when you combine Workstation and Server's files for any particular processor set.

A binary comparison of corresponding processor subdirectories from a Server CD and a Workstation CD flags about 200 files as inconsistent. Roughly 100 of these flagged files ship with Server but not with Workstation. This number includes files related to Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS) administration, JET database integration, license management, log viewing, Macintosh volume management, network client administration, NetWare migration, remote-system policy management, remote boot management, domain user management, and domain configuration management. Finally, the Server logon bitmaps are only on the Server CD.

Another 10 flagged files ship with Workstation but not Server. Local machine user management and a different version of Disk Manager Help files make up the bulk of this group. In addition, the workstation logon bitmaps ship only on the Workstation CD, of course.

The other 90 or so flagged files have different content for each platform and are related to text-based inf setup files. Most differences in these files result from NT Setup's internal management of virtual setup disks that have different names for Workstation and Server.

The remaining differences are the result of file entries related to components that ship on one platform but not the other. Beneath the processor-specific subdirectories on the Workstation and Server CDs are inetsrv directories. They are where Internet Information Server (IIS) installs from on a Server setup and where Peer Web Services installs from on a Workstation setup. Except for a few setup files, the files that make up these products are identical at the binary-level. Server ships with a network monitor and Microsoft's FrontPage Web site-creation software, neither of which comes with Workstation.

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

## How Does the Operating System Know What It Is?

NT's core is in the file `ntoskrnl.exe`. It consists of the NT microkernel, the Executive, the Object Manager, the Memory Manager, the Process Manager, and other low-level system components. Because the `ntoskrnl.exe` image is identical on the Workstation and Server CDs and because Microsoft and empirical evidence show that these subsystems behave differently on installed workstations and servers, the NT OS must dynamically obtain the factor that determines Workstation or Server behavior. In fact, very early in the operating system's initialization, it queries the Registry to determine whether the platform is a workstation or a server.

ProductType	System
Winnt	Workstation
Servernt	Server
Lanmannt	Advanced Server

On NT 3.51, only one Registry value separates Workstation from Server:

```
\hkey_local_machine\system\currentcontrolset\control\productoptions\producttype
```

The Microsoft Developer's Network Knowledge Base article Q124305 documents this entry and defines it as a string that can have one of the three values shown in Table 1. Setup reads the value early in system initialization and later initializes the NT Memory Manager in the `MmInitSystem()` routine, where it sets a global variable, `Mm ProductType`, to reflect `ProductType`'s value. Internally, the operating system does not distinguish between a server and an advanced server. So for either type, `MmProductType` is set to 1; for Workstation, `MmProductType` is set to 0.

On NT 4.0, two Registry values play a role in separating Workstation from Server, and the `ProductType` value's role is carried forward from NT 3.51. An 8-byte additional value (two `DWORDs`) is involved, and its primary purpose it is to encrypt operating system expiration date information. However, the value has a component that must be set according to the `ProductType` value. A violation of this rule results in a blue-screen error during boot. This "system\_license\_violation" error makes the system inoperative.

Another change from NT 3.51 is that NT 4.0 creates Registry notification work items for the two values in question. If somebody attempts to change these values, a system worker thread is notified to write the original value over the change. Attempting to alter `ProductType` also results in the system popping up a message box stating that the user has tried to violate the licensing agreement.

Other operating system subsystems and even device drivers can query the system's type by invoking the exported Memory Manager function `MMIThis AnNtAsSystem()`. It simply returns the `MmProductType` value.

User-level applications have two options for determining the operating system configuration. The first is just to read the `ProductType` Registry value directly. The second is to call a native NT function,

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

RtlGetNtProductType(), that is in the ntdll.dll. When called for the first time after a system boot, RtlGetNtProductType() reads the Product Type Registry value and caches the result in an internal variable that it returns to the caller. The internal variable is 1 if the Winnt string was read, 2 if Lanmannt was read, and 3 if Servernt was obtained. Subsequent calls avoid reading the Registry by returning the cached variable.

## Operating System Tuning

NT initialization accesses the Mm ProductType variable and the MmIsThis AnNtAsSystem() function several times to directly affect the values of about 25 internal variables and indirectly affect many more. The basic philosophy the tuning evidences is that when NT is acting as a server, the system's responsiveness to file- and network-related work needs to be of primary concern, and memory footprint is secondary in importance. Workstation aims responsiveness at systems that run multiple applications simultaneously, and memory footprints are kept smaller to accommodate more applications, so responsiveness of system-related functions can suffer. The NT Executive, the Memory Manager, the File System Runtime, the Process Manager, the I/O Manager, the Cache Manager, and the Object Manager perform optimizations according to the product type.

## Product Type and System Size

The product type directly affects the system size identifier variable, Mm SystemSize, which has repercussions for many other dynamic tuning variables. It can have one of three values: Mm SmallSystem, MmMediumSystem, and MmLargeSystem.

System size	Workstation thresholds	Server thresholds
Small	0MB-19MB	0MB-19MB
Medium	20MB-32MB	20MB-64MB
Large	>32MB	>64MB

The system size is the basis for tuning throughout the operating system and affects such things as the number of worker threads to create and the amount of memory to set aside for specific subsystem tasks. Table 2 shows the amount of memory necessary to rate a system as small, medium, or large. Note that the threshold for a large Server is double the large threshold for Workstation.

The NT Executive is in charge of creating system worker threads, and it initializes them in the routine, Exp WorkerInitialization(), which uses both the system size and the product type to determine how many threads to create. An NT system has three types of worker threads, each aimed at different priorities of work: delayed worker threads perform low-priority tasks; critical worker threads perform jobs that must be completed as soon as possible and run at a realtime priority; and one hypercritical worker thread exists only for specific, system-related operations such as exited-process cleanup.

Ideally, the number of worker threads needs to be high enough that they pick up work tasks as soon as the tasks are assigned. The tradeoff is that idle worker threads needlessly use system resources.

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

Table 3 shows how many critical and delayed worker threads the system creates by default for different parameters. On medium and large systems, a server has twice as many critical worker threads as a workstation.

System size	Delayed worker threads	Critical worker threads	
		Workstation	Server
Small	3	3	3
Medium	3	3	6
Large	3	5	10

However, by changing Registry settings under the key:

`\hkey_local_machine\system\currentcontrolset\control\sessionmanager\executive`

an administrator can direct a workstation to have just as many, or even more, worker threads than a default server configuration. The `AdditionalCriticalWorkerThreads` value under this key controls the number of extra critical worker threads that are created, and you can set it to a value up to 16. Similarly, `AdditionalDelayedWorkerThreads` controls the number of extra delayed worker threads created, and it can also be a value up to 16.

Once worker threads are started, they sleep until an item that they need to process is placed on a work queue. The form of sleep that Server worker threads perform differs from that of Workstation worker threads: Server threads sleep with their stacks locked into memory, whereas Workstation worker threads can have their stacks paged to disk. This optimization means that Server worker threads are generally more responsive when work arises, because they never have a delay reading their stacks in from the disk. However, Server threads always contribute to the in-memory footprint of the operating system.

## Product Type and Memory Manager

The Memory Manager uses the product type value to make tuning decisions in several places. The first decision occurs in `MmInitSystem()`, where the Memory Manager sets the variables `MmThrottleTop` and `MmThrottleBottom`. The system's lazy modified-page writer thread uses these variables in its determination of whether to write pages to the paging file if they have been changed out, in anticipation of a future need to do so.

On a server, these values are about twice as high as on a workstation because a workstation is likely to have more paging activity as processes and threads of different interactive applications are started and stopped. Servers are assumed to run a few unchanging applications that have fairly stable memory requirements, so the anticipation for paging activity is lower than for workstations.

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

The next place where the Memory Manager tunes each product type is in initializing the system's working-set manager. The working-set manager is a background thread that trims the in-memory footprint of applications to fit within certain ranges.

On workstations with less than 32MB of memory, a flag, MiDoPeriodic AgressiveTrimming, is set to TRUE. This setting causes the working-set manager to trim the footprints of active processes every second or so. This trimming gives more applications a chance to simultaneously squeeze at least part of their required data and code into memory.

On servers and large workstations, trimming is much more conservative and occurs only in situations where memory usage and page faulting are extremely high. In addition, during process creation on servers, processes are flagged so that the working-set manager gives them more physical memory than they would get on a workstation.

The final place where the Memory Manager accounts for running on a server rather than a workstation is where the Memory Manager determines the size of the operating system's pool of pageable memory. The pageable pool is the area from which the operating system and device drivers allocate data. Once this pool is exhausted, the system will fail to function properly. On Server, the paged pool size is a minimum of 50MB.

## Tunneling

The file system runtime performs an interesting optimization that Microsoft has introduced with NT 4.0: To preserve long file names in the face of legacy 16-bit applications that would otherwise destroy them, the NT 4.0 file system supports the notion of long file name tunneling. Tunneling is necessary when a 16-bit application, such as a word processor, maintains the current version of a document in a temporary file. When the user saves the document, the program may delete the original and rename the temporary to the original file's name.

In the absence of tunneling, the renaming of the temporary file replaces the original long filename with the short-name form. When tunneling is in effect, the file system typically remembers delete operations for 15 seconds, and if a new short filename file is created with the name of a file that has recently been deleted, the file is automatically assigned the long name of the recently deleted file. On a server, the default number of remembered delete operations is 1024, but on a workstation, the number is only 256.

You can explain this difference if you assume that servers are likely to serve file systems to large numbers of clients that will probably have much more activity over short periods than a workstation file system will have. You can override the default number in the Registry value `\hkey_local_machinesystem\currentcontrolset\control\file system\maximumtunnelentries`. Also, under the same key, by setting the value `MaximumTunnelEntryAgeIn Seconds`, you can tune the time-based window of recall for delete operations.

## Foreground Execution Quantum

One dramatic optimization that the Process Manager performs is that it sets the foreground execution quantum on a server to twice that of the foreground quantum on a workstation. And the background quantum on a server, which is identical to its foreground quantum, is six times larger than the corresponding quantum on a workstation. This difference means that on a server, threads execute for longer periods without interruption, and fewer threads generally get a share of a processor over a given duration. Because the implicit assumption is that servers will be running fewer and less

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

interactive applications than workstations, this optimization usually aids server throughput by doing away with needless thread-switching overhead.

## Lookaside Lists

The remaining optimizations that the operating system performs involve the creation of lookaside lists, which are new in NT 4.0. Lookaside lists are private stashes of fixed-sized chunks of memory created for specific tasks. This approach avoids an expensive call to the system-wide memory allocator when memory is available in a private store. In some cases, this approach can even avoid a lengthy wait for a disk operation to complete as the system makes room for an allocation out of physical memory. The Object Manager creates a stash for allocating internal object names, the Cache Manager creates a stash for the disk write-behind and read-ahead threads, and the I/O manager creates lookaside lists for I/O commands. In all these cases, the lookaside lists are larger if you configure the system as a server, because performance of these various operations in a server environment is considered more important than the impact of removing this memory from the general pool.

## Device Driver Tuning

Only four device drivers that ship with NT behave differently depending on the platform they are running on. These four drivers are `afd.sys`, `srv.sys`, `nwlnknb.sys`, and `ntfs.sys`.

`afd.sys` is the device driver responsible for managing Microsoft's Winsock TCP/IP communications protocol. `afd.sys` performs two modifications.

1. `afd.sys` sets the size of the network transfer frame to 4KB if `afd.sys` is running on Workstation and 64KB if this device driver is running on Server. A larger size means that communications will usually be faster but also that allocated buffers will have a negative impact on the system's available memory.
2. `afd.sys` modifies a variable according to the product type. This change occurs where this driver sets the limit on simultaneous network file transfers to 2 if it is running on Workstation. But `afd.sys` checks the Registry entry:

```
\hkey_local_machine\system\currentcontrolset\control\services\afd\parameters\maxactivetransmitfilecount
```

for the limit, if `afd.sys` is running on Server. This variation exists purely to limit the functionality of Workstation, rather than to optimize performance.

`srv.sys` is the network server device driver that controls network connections to a machine. Its check of the product type is also for functionality limitation. About 10 of its variables are restricted to certain values if this driver is running on a Workstation.

The most notable of these variables is `SrvMaxUsers`. On a Server, `SrvMaxUsers` is set to an effectively infinite value. But on a workstation, `SrvMaxUsers` has an upper limit of 10, allowing a maximum of 10 simultaneous network connections to the machine. This limit applies whether the connections are file transfers or network logons.

Unlike `srv.sys`, `nwlnknb.sys`, the NetWare network NetBIOS communications protocol driver, queries the system type for performance-optimization purposes. `nwlnknb.sys` maintains an internal cache of network routes. On Server systems, this cache is three times larger than on workstations, apparently because a server will need more network connections.

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

ntfs.sys is the file system driver for disks that are formatted with the native NT File System (NTFS), and this driver checks the product type for tuning. During its initialization, a call is made to MmIsThisAnNtAsSystem(), and based on the result, ntfs.sys sets the size of 11 lookaside lists that it creates for memory caching. Each lookaside list is dedicated to a different temporary data structure that must be allocated during file system operations. Again, in anticipation of more file system activity, the lists are twice as big on large servers (according to the MmSystemSize variable), as on large workstations.

## User-Level Application Tuning

Figure 1 shows a list of the user-level components that query the system type. These components are too numerous to allow in-depth explanation of their behavior. However, I can highlight a few.

Winlogon checks the product type to determine which splash screen to display, lanman.bmp (for 16-color video modes) or lanma256.bmp for Server logon, and winnt.bmp or winnt256.bmp for Workstation logon. Explorer checks the product type and shows a bitmap along the left side of the Start menu. The bitmap reads Windows NT Server or Windows NT Workstation, as appropriate.

The last application in Figure 1 is windisk.exe, the NT disk administrator program. It reads the Registry directly to determine which platform it is executing on.

If Windisk is on a Server, it provides a Fault Tolerance menu that has entries you can use to create striped sets with parity, and mirrored drives. On a Workstation, Windisk does not make this menu available. This difference explains why only Servers can create and manage fault-tolerant disks.

Several Microsoft products have setup programs that check the product type. Notable examples are BackOffice and IIS. BackOffice refuses to install any BackOffice suite programs, SQL Server, Exchange Server, IIS, and System Management Server (SMS), if the product is on a Workstation.

No technical reason lies behind this limitation: Tests show that these applications function properly if the setup program is foiled into installing them on a Workstation.

As I mentioned earlier, the version of IIS that ships with the NT 4.0 CDs checks the product type and installs as Peer Web Services if its setup program detects that it is on a Workstation, and as IIS if it detects that it is on a Server.

## No Blanket Statements

Although NT Workstation and NT Server share the same bits, the few tweaks that the system makes according to the product type can significantly affect the performance of the systems, so that given the same workloads, one or the other will perform better. The tuning that takes place seems to aim Workstation and Server at different, clearly-defined workloads. However, making blanket statements about the suitability of this tuning for all situations that can arise in actual installations is impossible. For example, some people argue that, with the advent of Common Gateway Interface (CGI), a Web

FIGURE 1: User-level components that query the system type
services.exe
samsrv.dll
msgina.dll
msafd.dll
localspl.dll
wkssvc.dll
netlogon.dll
explorer.exe
browser.dll
winlogon.exe
windisk.exe

# Inside the Difference Between Windows NT Workstation and Windows NT Server

Mark Russinovich  
(Reprinted from WindowsItPro Magazine)

serving NT Server would do better to tune itself as an interactive Workstation than as a single application Server.

NT's predefined optimizations may be suitable for many cases. However, Microsoft was shortsighted in not giving systems administrators control over the optimizations, rather than foisting on them one-size-fits-all solutions.