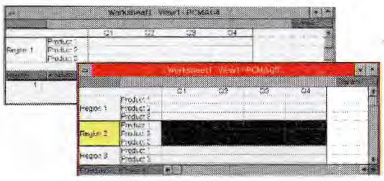
# NT unleashes the

**Some eight years after Intel first launched a 32-bit processor, Microsoft has launched its first 32-bit operating system. NT packs a powerful punch with multitasking and multiprocessor support. Although**



**the wait for Lotus Improv has been shorter, the finished product is no less exciting. And while Lotus was porting Improv, Adobe was**


**developing the Windows version of its best selling Macintosh package, Photoshop.**


Windows might be a fine user interface, but for performance-oriented users, it leaves a lot to be desired. For a start, it's slow. Staring at the hourglass isn't just boring, it's unproductive. And despite the improvements of version 3.1, Windows applications still have a habit of crashing. Now help is at hand. Enter Windows NT.

Windows NT is a full 32-bit, multitasking operating system that runs DOS, Windows, Windows NT and OS/2 text-mode applications across a raft of hardware architectures.

Microsoft first talked about NT early in 1990. At the time, the Joint Development Agreement with IBM still held, and NT was the 'New Technology' that would be at the heart of OS/2 3.0. That plan fell apart as Microsoft and IBM split, and Microsoft finally followed the route that OS/2 should have taken—a full 32-bit multitasking operating system with Windows as its front end.

Microsoft's strategy for Windows NT hasn't changed substantially since 1990. It's positioned to complement Windows 3.1, and is targeted at server systems and high-end workstations.

This common heritage with Windows is the greatest strength of Windows NT. Windows NT is directly competing with OS/2, which is also targeted at servers and higher-end workstations, and being part of the mainstream Windows family must help it.

NT is portable across different processor architectures, and supports multiprocessor machines. OS/2 is expected to do the same by early next year. Both systems support network operating systems that originate in the IBM/Microsoft Joint Development Agreement—LAN Manager for NT and LAN Server for OS/2.

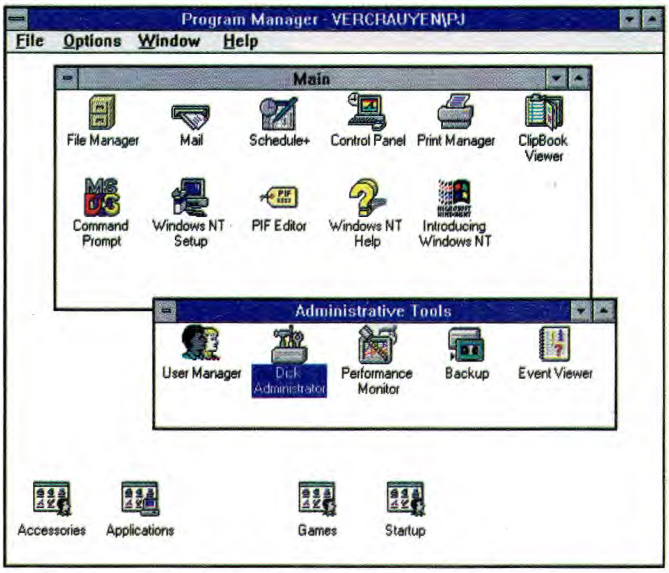
The scalability of NT will attract developers of applications that need all the computing power they can get. Server database

### ********** NT at a glance **********

- ★ **Microsoft's first full 32-bit operating system.**
- ★ **Support for pre-emptive multitasking.**
- ★ **Uses the familiar Windows 3.x-style user interface.**
- ★ **DOS, Windows and text-mode OS/2 application compatibility.**
- ★ **Robust, enterprise-wide security from unauthorised access and Trojan applications provided by C2-level security.**
- ★ **Windows NT includes client and server functionality.**
- ★ **Out-of-the-box, peer-to-peer networking.**
- ★ **Consistent programming interface allows rapid cross-platform development across RISC and Intel architectures.**
- ★ **Symmetric multiprocessor support for server and computationally-intensive client applications.**
- ★ **NTFS: a new high-performance, server-oriented file system with added security, recovery, POSIX compliance and huge disk volume support.**

T LOOKS

# power of 32-bit multitasking




**As you'd expect, NT looks very much like Windows 3.1 or Windows for Workgroups—the only new Program Manager group is for admin tools.**

systems, high-end 3D graphics and video, scientific and engineering systems are all likely candidates. Properly written NT software can make full use of multiple processors.

NT's ability to run applications across different architectures is also attractive to developers. Currently, NT runs on Intel, MIPS and DEC Alpha processors. It offers a consistent programming interface across multiple systems, all with the same graphics architecture. All developers need to do is take the source code across and recompile.

At the end of May, the final product should be released. In this article, we are testing the last beta version, which Microsoft says is feature complete and optimal in performance.

### Windows NT and NT Advanced Server

In a late change in the target markets for the new operating system, NT has been divided into two products: Windows NT 3.1 for power-user workstations and Windows NT Advanced Server for file, database, communications, messaging and application servers on high-end hardware. The difference between the two is in the extra services built into NT Advanced Server, comprising LAN Manager, additional fault-tolerant features, like disk mirroring, duplexing and RAID Level 5 support, distributed security, central network administration, Macintosh connectivity and remote access services.

This division doesn't mean that Windows NT 3.1 has been stripped down. It still includes, for example, integral networking and data and user access security, along with support for uninterruptible power supplies. But it does allow Microsoft to target 'enterprise computing' and MIS customers with NT Advanced Server and the forthcoming 32-bit SQL Server and SNA Server for NT. At the same time, it still tempts high-end PC users with a better Windows 3.1 than Windows 3.1. In either version, NT gives all customers a chance to move away from the Intel architecture towards RISC architectures based on new MIPS and DEC processors.

No matter what technical advantages NT promises in the future, it needs to run existing Windows and DOS applications as seamlessly as possible and as fast as they run now, while running true NT applications even faster. It needs to be easy to install and administer for the average PC user. And it needs to offer MIS managers the detailed access control and security features required for enterprise systems. It must also integrate smoothly with existing networks, which might include Novell servers with DOS clients, as well as peer-to-peer systems using Windows for Workgroups.

### Installation

Installation was painless, apart from watching the PC reboot twice to start the graphical portion of the setup program and to convert the initial FAT file system installed to NTFS.

For our test machine, we chose a high-end PC clone with a 66MHz 486DX2 processor with 16Mb of RAM and an additional external hard disk on an Adaptec 1542B SCSI host adapter to increase storage capacity.

The new beta release requires 12Mb of RAM on Intel-based machines and 16Mb on RISC systems, and around 80Mb of free disk space including 20Mb for the virtual memory swapfile. These requirements will probably be the same for the final release version, although Microsoft is still talking about a version that will run in 8Mb of RAM. Support is added for IDE and ESDI hard disks to that of SCSI hard disks, but a SCSI adapter is still needed for CD-ROM drives.



The display driver for the Tseng ET4000-based video board in our test system was chosen from the extensive list provided with NT, which covers hardware from Cirrus Logic, Dell, IBM, MIPS, S3, Trident, Tseng, VESA, Video Seven and Western Digital/Paradise in a variety of resolutions, colours and refresh rates. For this review, resolutions of 800 by 600 and 1,024 by 768 in 16 and 256 colours were used.

NT can be installed on top of an existing DOS, OS/2, Windows 3.1 or Windows for Workgroups system, and attempts to migrate existing system information sensibly. The NT Boot Loader can be set up during installation to let you choose which operating system to start at boot-time, and


information from the Windows Registration Database, as well as the Windows INI files is carried across. In addition, all Program Manager group files listed in PROGMAN.INI are migrated unless their names clash with NT's own program groups.

On the test machine, however—with a new unformatted SCSI drive—none of this was necessary. Installation was performed from the 21 high-density floppy disks, rather than CD-ROM, to provide a real comparison with OS/2 2.0's installation, also from the 21 disks. The comparison definitely favours NT.

Setup looks and works just like any Windows setup program, and automatically detects SCSI and network adapters where possible. The only unusual features are the options to set up faster NTFS or OS/2's HPFS file systems, instead of the DOS FAT file system, and the need for unique computer and user names that are used by the NT security system.

## User interface

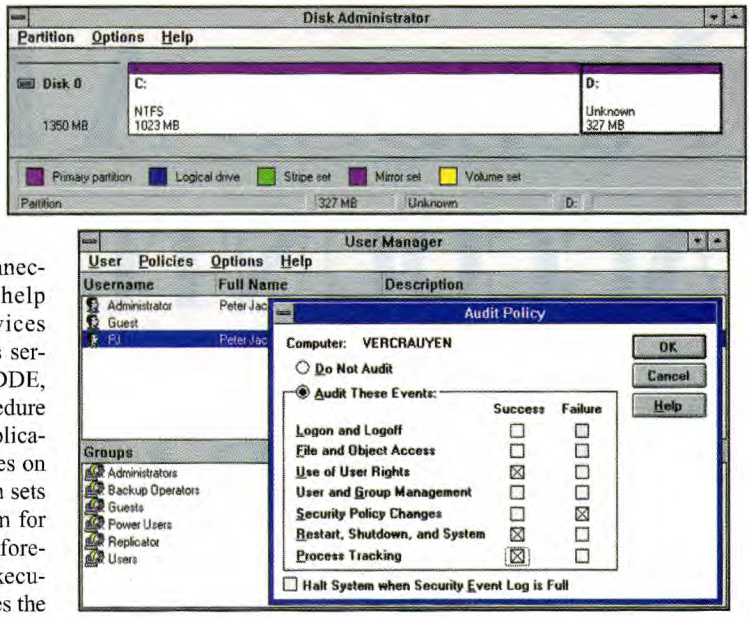
The familiarity of the NT user interface is one of its major advantages, although it can also be a disadvantage in view of the differences caused by true multitasking and the new security features. A tutorial program called Introducing Windows NT—which is DOS-based—is included to give an introduction to the networking, security and administration features of NT, although most new users will just start fiddling around with the familiar objects on the desktop. Along with some new NT-specific utilities, all the old utilities are there, although some of these have been changed to add functionality or bring them into line with Windows for Workgroups.

For example, NT's File Manager now works like that in Windows for Workgroups, with a toolbar, network drive connections and directory sharing, as well as security features if the NT partition uses NTFS. The clumsy Clipbook utility for networked OLE is also there, as are 32-bit versions of the Mail and Schedule+ applications from Windows for Workgroups.

Familiar utilities that have been changed substantially include the Control Panel, which now has a redesigned Network option and new icons called Server, Services, System and UPS. The Server tool manages shared resources, user connections and file-locks to help administer a server. Services starts, stops and configures services, including Network DDE, the Clipbook, Remote Procedure Calls and the Directory Replicator that duplicates directories on different computers. System sets the default operating system for the Boot Loader, specifies foreground and background execution priorities and configures the virtual memory swapfile. And UPS configures an uninterruptible power supply connected to a COM port.

Print Manager has also undergone big changes. It now includes the printing architecture from the Win32 API; at long last, there's no need to select a local printing port, like LPT1, and redirect it for network printing, and no need to install printer drivers locally to print on a remote printer. Network browsing is provided to help find the right printer, and remote printer administration allows any network printer to be reconfigured from any workstation.

The completely new utilities include a simple 3270 Emulator for connection to IBM hosts over Token Ring or Ethernet networks, as well as a set of system control tools.




**Disk Administrator is a Windows-based version of FDISK and FORMAT, while the User Manager is the system manager's access control tool.**

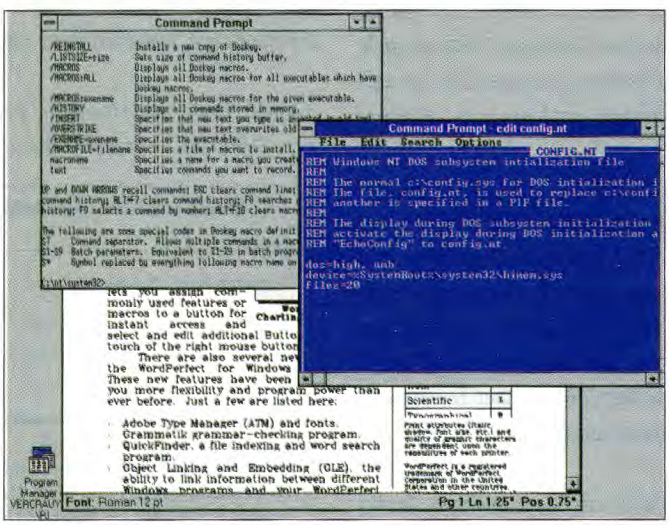
The User Manager program is used to control user access to resources on the local machine or on machines connected into a network domain. This facility also handles the Program Manager group settings for individual users. With NT, it's possible to configure groups so that only certain group icons appear when a particular user logs on.

These settings are shown visually in the Program Manager window by new group icons—personal group icons for a single user have a small picture of a head added, while groups that can be accessed by any of the machine's users have a small picture of a computer. Newly-installed applications are automatically placed in common groups until their status is changed in the User Manager.

Two other administrative tools are concerned with disk management. The Disk Administrator program combines the functions of FDISK and FORMAT in setting up and formatting disk partitions, and can also convert existing partitions to NTFS format if required, while the Backup utility backs up and restores data from and to the local machine or a remote machine on the network. Only SCSI tape drives are supported at present.

The final two tools, Event Viewer and Performance Monitor, give the system administrator information about the machine's behaviour. The Event Viewer logs significant hardware and software events, while the Performance Monitor tracks levels of processor, memory and disk usage on any networked machine, and provides alerts if any stray outside bounds, as well as producing usage reports and logs.

Of course, users who just want to boot NT and run applications won't be concerned with these tools, and in big networked installations a single network administrator will take responsibility for setting up users, user groups and access control for directories, files and programs.




**Each DOS session has its own virtual screen and separate colour palette.**

## Running applications

When it comes to running applications, NT delivers on its promise of running existing DOS

# Windows NT Lab Report


With an operating system like Windows NT, the main performance issue is the overhead the system places on the applications. Windows NT running two processes will impose an additional load for controlling and switching between them. On top of this, Windows NT runs not only its own 32-bit software, but also DOS and 16-bit Windows programs.

One simple test is to compare identical 32-bit code running under DOS as well as under NT, which is made possible by DOS Extender technology. PharLap's DOS Extender 5.1 has recently been made available to work with the 32-bit C compiler that Microsoft provides with the NT Software Development Kit. It enables you to take C or C++ code (not for graphics) and run it in both environments. (The extender also lets you run the compiler itself under DOS.)

We took our 32-bit processor test and merely changed the output code. The main test code was identical under NT and DOS; the same compiler switches and object modules were used to produce two programs, one under NT and the other under DOS using the PharLap extender. The difference in performance was negligible, with the NT program falling behind a quarter of a percent from the DOS program. For serious 32-bit performance, NT doesn't get in the way.

With a multitasking operating system, the overhead will probably rise as you load more tasks onto the system.

We used up to 15 instances of our processor test to assess the overhead by totalling the throughput for the various instances. NT is designed to handle such high loads best when the machine uses more than one processor. A typical application of such a multiprocessor machine would be a file server. If a heavy database task was communicating with a server, and suddenly a large demand for file transfers came through, the load balancing should distribute the various running threads within milliseconds to give the optimum overall throughput.

Our simple processor test tends to show that this is the case. We used an AST Manhattan SMP machine to assess the multiprocessor capabilities of NT. It had three 50MHz 486 processors, each with a 256K cache, and 64Mb of RAM. Loading three instances of the processor test gave a score for each equal to that for just one instance. Clearly, each instance was running on one processor. Load more instances, and each one attains a throughput equal to the total possible (with three instances) divided by the number of tests running. You don't get two or more instances 'stuck' on one processor. Time is apportioned to each thread according to its priority, and as running a thread is equal to loading a set of registers, the threads can be executed on any processor at any time, balancing the load almost evenly.

All the tests mentioned so far have involved 32-bit code. So how good is NT at running 16-bit code? A standard DOS 16-bit version of our processor test loses just two per cent in performance when running under NT. But it's a different story with Windows applications.

We tested processor, disk and graphics performance of Windows 16-bit code running under both Windows and NT. The processor test gave a disappointing result under NT. The Windows test showed a drop in performance of 58 per cent compared with running under Windows 3.1. Microsoft told us that there's a known bug with floating-point support and that this could be the cause.

Our test has 0.5 per cent of instructions as floating-point—a mix derived from application profiling. Sure enough, taking these instructions out of the code gave essentially the same performance under Windows 3.1 and NT. Microsoft assured us that this problem will be fixed in the final release.

Graphics gains 26 per cent and disk performance is multiplied by a massive factor of 15. This is almost certainly due to the better memory management and cacheing of NT. Our test only used an 8Mb file; a more relevant test would be under server conditions—something we shall do for a later article.

We also ran 32-bit versions of the same Windows graphics and disk tests under NT. These showed seven and 55 per cent improvements, respectively, over the 16-bit tests. Considering that the 16-bit results were themselves faster than when run under Windows 3.1, the overall gains show that NT lives up to expectations, delivering high 32-bit performance.

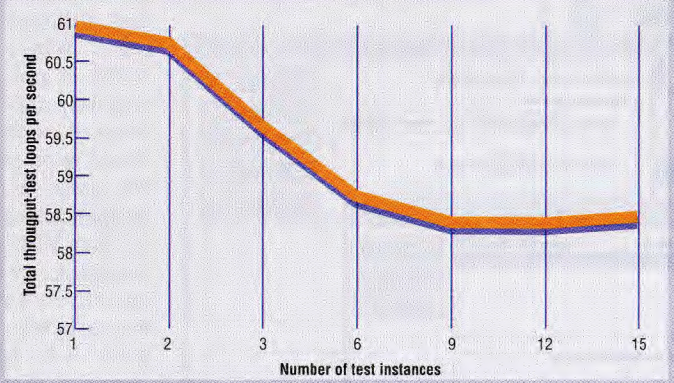
When you add to this raw, single-process performance, the outstanding multitasking and multiprocessor support, Microsoft's Windows NT is an excellent example of how an operating system should be built.

*—Edward Henning*

## PC Labs benchtests

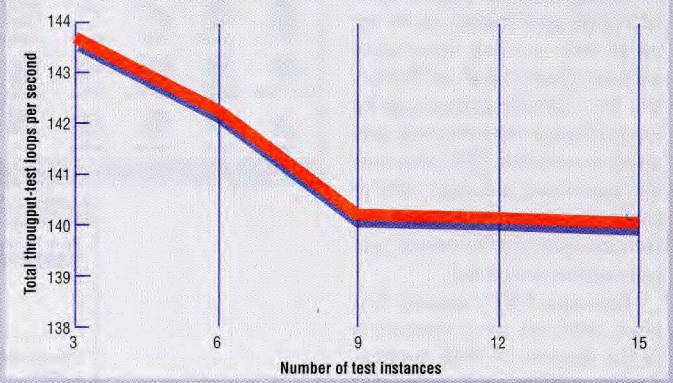

***PC Magazine* NT Bench comprises three new tests. They're based on profiling application behaviour, and test the processor and memory system, graphics and hard disk. Using the same basic source code, the tests have been compiled, where relevant, under DOS 16-bit and 32-bit, Windows 3.1 16-bit, Windows NT 32-bit and OS/2 32-bit environments.**

## Total processor throughput: (Compaq)




**The graph shows the result of loading multiple instances of our processor test onto a Compaq Deskpro 66M, which runs one 66MHz 486DX2 processor. At 15 instances, the total throughput showed only a four per cent drop from the optimum of having just one instance running. The proportion of time given to each instance was the same, but this could change given different priorities in a real situation.**

## Total processor throughput: (AST SMP)




**The graph shows the effect of running our processor tests on a machine with multiple processors—we used an AST Manhattan Symmetric Multiprocessor machine with three 50MHz 486 processors installed. Due to the design of our tests, we were only able to display a maximum of 15 instances on-screen at a time. The results show a drop in total throughput of 2.5 per cent from the optimum of three instances.**

and Windows applications unchanged, at least most of the time.

There are no actual copies of DOS and Windows on an NT system, just DOS and Win16 subsystems that sit alongside the Win32, 16-bit OS/2 and POSIX subsystems on top of the NT executive and kernel routines. The familiar DOS Command Prompt icon is still there, but it doesn't load DOS. Instead, it provides a command-line interface to NT as a whole. From this command prompt users can start any application supported by NT, including 32-bit and 16-bit Windows programs, DOS programs, text-based OS/2 programs or POSIX-compliant programs. There's also a full set of 32-bit NT commands, including most of those in DOS 5.0, with additional commands mainly concerned with networking and other 16-bit commands required for backwards compatibility with DOS 5.0 and OS/2.

But most users will treat the Command Prompt as if it was DOS. When it loads, the AUTOEXEC.BAT file in the root directory of the boot drive is parsed along with the CONFIG.NT and AUTOEXEC.NT files in the NT System32 subdirectory. Any settings in AUTOEXEC.BAT, like environment and path variables, are held in common and used by all Command Prompt instances, while the CONFIG.NT and AUTOEXEC.NT contents only apply to a particular Prompt window.

Using the PIF editor, each MS-DOS application can be set up to use its own separately named CONFIG and AUTOEXEC files. Device drivers can be loaded high in the usual way, and memory-resident TSR programs are supported, although they're run in the Command Prompt session in which they're started, and can only be used there.

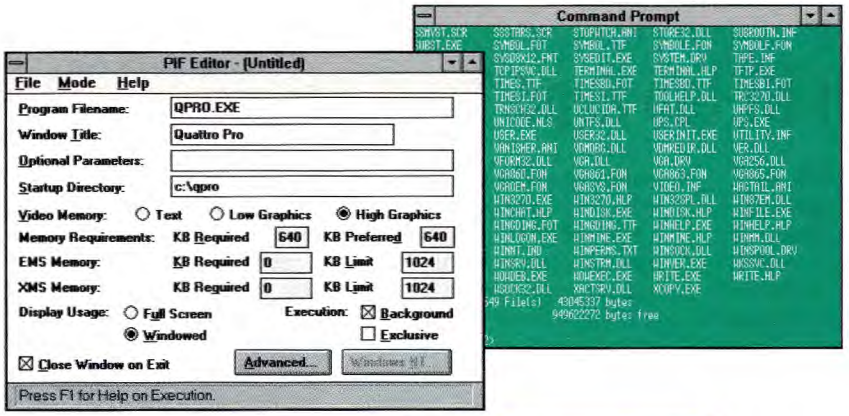
Because of NT's security features, there are some limitations in the support of DOS applications. Any DOS program expecting direct access to system hardware, like fax card drivers or communications software, won't get it: NT is set up to isolate applications completely from the PC. Such programs will only work with brand-new NT device drivers, which will come, in most cases, from the software developers themselves. However, Microsoft does supply drivers to virtualise the serial hardware for DOS comms programs. Graphics-mode DOS applications will only run full-screen, and are frozen when reduced to background icons, although text-mode programs can be run in both foreground and background either in a window or full-screen.

On the whole, DOS support and multitasking is excellent and fast. The high memory area is used to give around 616K of free memory for applications, As standard, 1Mb of extended RAM is supplied, and expanded memory support is there for programs that need it. There's full Windows-style control over the size and position of each Command Prompt window, as well as the text size and colour in the window. And the DOS environment really is seamless, so that DOS programs—like any other NT-supported program—can be run using File Manager icons, the File/Run menu command or directly from the Command Prompt. With the MS-DOS support in NT, Microsoft has successfully challenged the 'better DOS than DOS' that was one of OS/2 2.0's big plus points.

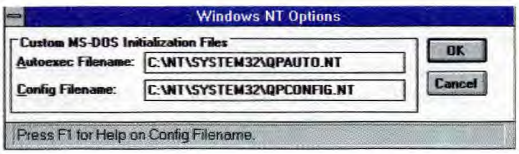
As you'd expect, the Windows support in NT is superior to that in OS/2 2.0. Most 16-bit Windows applications we tested ran smoothly under NT; the main exception was IBM's integrated Legato package, which lost its toolbar icons and also failed to respond to pull-down menu commands. This incompatibility was useful, in that it gave a chance to test NT's crash prevention. When Legato apparently locked up waiting for a menu event that never arrived, it was easy to switch away from the locked program with Alt-Tab and use the Task List to kill it. Pressing Ctrl-Alt-Del with a locked program in the foreground also gives the option to kill the task, return to NT or try the frozen application again.

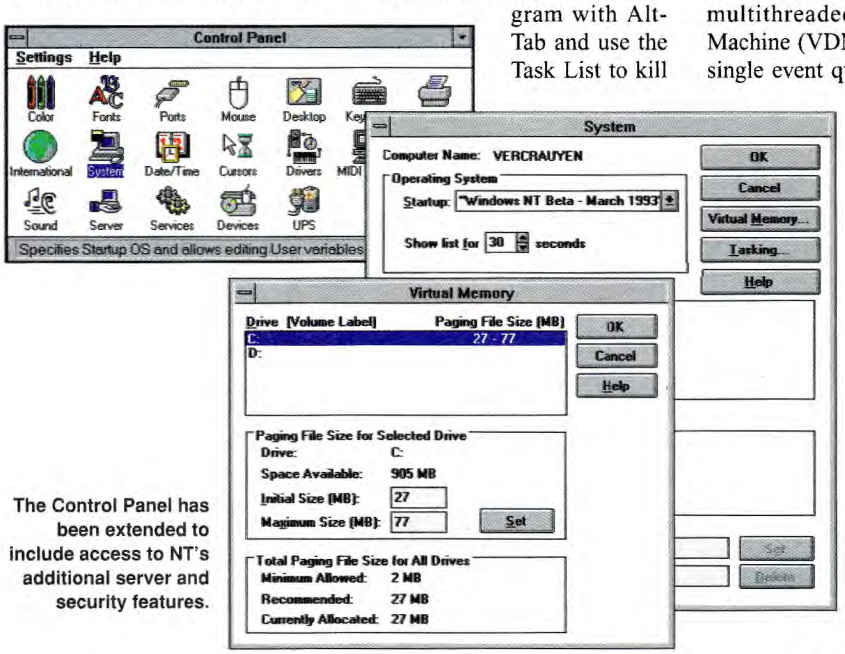
The other test of NT's crash prevention came while running a DOS program in the background. A problem with the Tseng screen driver severely corrupted the display, both in the DOS program and on the NT desktop. However, calling up the Task List and killing the DOS task restored both the screen display and the system. With Windows 3.1, a screen driver problem like this would usually require a hard reset.

However, 16-bit Windows applications aren't as protected from crashes as new 32-bit ones will be. While each 32-bit Windows program has its own protected memory space and its own event queue—so that a single application attempting to step outside its memory bounds or missing an event won't affect other programs—all 16-bit Windows programs run in a single multithreaded Virtual DOS Machine (VDM) session with a single event queue. This means that although the Win16 VDM is preemptively multitasked with Win32 or other NT tasks, 16-bit applications in the VDM multitask cooperatively as they do in Windows 3.1. It also means that a single 16-bit Windows program missing an event can hang the entire




NT still uses Windows-style PIF files—extra flexibility comes with separate CONFIG and AUTOEXEC files that are available for each application.




The Control Panel has been extended to include access to NT's additional server and security features.

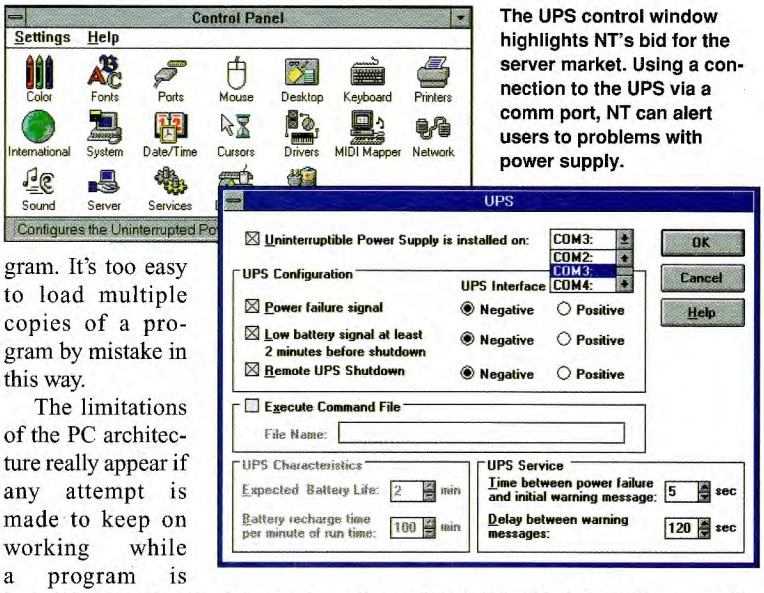

VDM and all applications running in it, although the rest of the NT system won't be affected. The performance of 16-bit Windows software is also affected by the need to convert 16-bit Windows API calls to 32-bit Win32 calls and convert the returned data back to 16-bit form. This process is called 'thunking', and imposes a time penalty on every API call an application makes. In practice, this penalty wasn't noticeable on the review machine in comparison with native Windows 3.1, partly because of processor speed and, presumably, partly because of the additional RAM that was installed to run NT.

The OS/2 and POSIX application subroutines weren't tested, but again the OS/2 support is seamless. The NT Registry and the CONFIG.SYS file in the root directory are scanned for OS/2 configuration data, and if none is found, reasonable defaults are used. Any 16-bit OS/2 1.x or 2.x text-mode application can be run from the Program Manager or Command Prompt, and any 'family' application that can run under both DOS and OS/2 is run in the OS/2 subsystem. But this is only on Intel-based PCs; only real-mode OS/2 programs will run in DOS emulation mode on MIPS and DEC processors.

**NT appears much slicker than OS/2, with better tools for managing system resources and networks**

### Architectural limitations

It has to be said that the standard Windows user interface and the PC architecture aren't particularly suited to multitasking. A new cursor icon combining an hourglass and the usual pointer is used to indicate that a program is loading, so the user can get on with something else in the meantime. But if another program icon is double-clicked while this icon is displayed, there's no indication that the second program is loading. On several occasions during the review, the second application didn't seem to load, when in fact it did load, but in a window hidden behind the windows of the Program Manager or another program. It's too easy to load multiple copies of a program by mistake in this way.




**The UPS control window highlights NT's bid for the server market. Using a connection to the UPS via a comm port, NT can alert users to problems with power supply.**

The limitations of the PC architecture really appear if any attempt is made to keep on working while a program is installing in the background. When trying to type into a word processor window while a program was being installed from floppy disk, we experienced significant lags between typing and screen display, as the floppy and hard disk activity tied up the processor. These time-lags are too long to do useful work during intensive disk activity, even with the foreground priority on maximum.

The multitasking works reasonably sensibly when a background application needs to prompt the user for some action. For example, while attempting to work in a word processor while installing a Windows program in the background, the installer prompt window for a new disk to be inserted popped up on top of the foreground application. This is helpful, but can be confusing. Switching to the installer program, caused the prompt window to disappear, and putting in the required disk and pressing Enter had no effect. We had to switch back to the previous foreground application to find the prompt window again and tell the installer to continue.

To see the full benefits of Windows NT multitasking we will have to wait for the Win32 applications that will appear when NT itself is officially released. Until then, the seamless integration of DOS and 16-bit Windows applications is done neatly, but is no more secure in theory than it is in OS/2 2.0. Like IBM's operating system, NT runs existing Windows programs side by side in a single VDM, with fragile protection from each other, while DOS programs are run safely in separate sessions.

### Conclusion

This beta version of NT is more stable than OS/2. There was none of the frequent operating system crashes that dogged OS/2, and none of the odd 'vanishing window' effects when trying to run Windows programs seamlessly. NT also appears much slicker, with better tools for managing system resources and networks and a more natural way of configuring DOS sessions for DOS applications using familiar techniques from DOS 5.0. Even the provision of standard SCSI, CD-ROM and video drivers for common clone hardware compares well with IBM's lack of third-party support.

If the latest beta of Windows NT 3.1 is any guide, Microsoft has delivered what it promised: a platform for high-performance applications that will run old applications successfully in the interim, and which has enough attractive new features to tempt those looking at OS/2 2.0 for workstations and those considering Unix for servers. The last group, in particular, will be waiting to see if Windows NT Advanced Server really is scalable across multiple processors and if it really can deliver added performance on RISC machines with full backwards application compatibility.

There are some cruel ironies in the arrival of NT 12 months or more behind OS/2 2.0. When OS/2 1.0 was launched in 1987, it was widely criticised for requiring a 286 with 2Mb of RAM, and Microsoft and IBM—later, IBM alone—laboured to make its 32-bit OS/2 2.0 run on meagrely specified 386 machines with 4Mb of RAM.

Thanks to the spread of Windows, the accelerating pace of hardware development and the bloating size of applications, machines capable of running NT aren't the outlandish and expensive things they would have been only a year ago.

At that time, Microsoft was tentatively suggesting that NT would be an expensive, server-based operating system purely because it required 12Mb of RAM and an 80Mb of disk. Today, it can offer the same operating system for PC workstations without anyone laughing. What's more, Microsoft can promise an upgrade price of around $50 (around £35) to move up to NT from Windows 3.1 or Windows for Workgroups.

Microsoft isn't perfect, and neither is NT. But Microsoft has the software developers' support and Windows has the momentum. Customers are now willing to wait for Microsoft in the same way they used to wait for IBM. With a bandwagon like this, and a new operating system that's this good, NT looks unstoppable.

*—Peter Jackson*

### Windows NT

**Microsoft UK**
Microsoft Place,
Winnersh Triangle,
Wokingham,
Berkshire RG11 5TP
(0734) 270001
**Price (ex. VAT)** TBA

**In Brief** Microsoft has succeeded in creating an excellent 32-bit operating system with multitasking and multiprocessor support. Competitive upgrades and the power of Microsoft's marketing will guarantee its success.

Circle **310** on reader service card

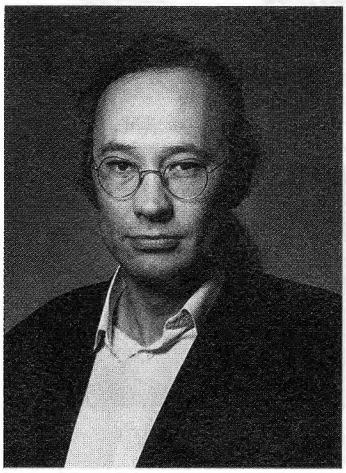
# NT redefines the PC

At a recent trade show I demonstrated some machines running Windows NT, together with some tests of ours and common pieces of software like Word and File Manager.

I leant on one machine, covering the front label with my hand, and let several people play around with the machine and its software, so they could get a feel for the system. I then asked them to guess what it was. A fast, high-performance 486 was the usual response. A couple of people even asked, 'Have you got an early Pentium in here?'

**We had two non-Intel PCs running NT, and it didn't seem to matter that they contained RISC processors. They simply delivered superior performance.**

They were all wrong. The PC was a MIPS machine, with not an Intel processor in sight. Sitting a couple of feet away was a DEC Alpha PC and, although the version of NT on it at the time was pretty flaky, it was clear that given a solid version, people's reactions would be the same.



Windows NT isn't the only familiar aspect of the DEC Alpha machine. Look inside and you'll see a PC bus with standard adapter sockets and Compaq's Q-Vision graphics card.

So here we had two non-Intel PCs running NT, and it didn't seem to matter that they contained RISC processors. They simply delivered superior performance.

But this improved performance doesn't apply across the board. DOS programs and 16-bit Windows programs all run in emulation mode under Insignia Solutions' SoftPC. While this is the best known and most successful Intel emulator, even with its proven track record it can never give performance up to the standard of the processor on which it's running. The performance is still acceptable, however—particularly on the Alpha, although at the time of writing we haven't been able to test this machine for speed, as the system just isn't ready.

Where you do get the full performance of RISC processors is with NT programs that have been recompiled for the new platform. Here, Microsoft has probably got a major success on its hands.

The one thing missing from the PC and workstation world has been consistent programming and user interfaces across different platforms. Unix in its many forms just hasn't been able to deliver. But suddenly, Microsoft looks as though it has come up with the Holy Grail of desktop computing.

We've ported our tests across these three platforms, and while they contain a few tens of thousands of lines of C code, the experience was nothing less than a joy. The only obstacles we encountered were a result of the different compilers having varying sensitivities to ambiguous coding: where one would give only a warning, another would give an error. The result was an improvement in the quality of the code.

It may well prove a little more difficult to port major applications, but not much. Developers I've spoken to tell similar tales—I've even heard of some gratuitously fiddling with their code, probably not quite believing that life can be this easy.

Many developers are interested in the ability to port across to workstations, and in particular to the DEC Alpha because of its high performance. When NT was first talked about, we heard that many Unix developers were interested in it because of the ability to port 32-bit applications 'down' to the PC architecture. Well, the traffic is clearly going to be in two directions, with NT providing the glue that binds PCs and workstations together.

We may well have to re-think our definition of a PC. We usually call an Intel and IBM-compatible system a PC. Not an Apple Macintosh, not an Archimedes, and certainly not one of these strange RISC workstations. But what does IBM-compatible mean nowadays? Not very much, really.

Intel will remain in the forefront for a considerable time to come—this isn't going to wipe it out, with everyone running Alpha machines. But it will dent Intel's market share and significantly blur the edges of the PC definition. As I've written before, the PCI bus system is clearly the one that should be the standard for future PCs, and both Intel and DEC are designing their next processors with this in mind.

So, is a PC now to be defined as a Windows NT machine with PCI and ISA buses, regardless of the processor it's running?

Well, this all depends on end users taking up the system, but there are plenty of reasons for them to do so. The architecture of NT, minus some bells and whistles and C2 security, will become the new mainstream PC operating system—currently code-named Chicago. Undiluted NT will then be for those users who require extra power or functionality.

The other major plus is one I wrote about last year. I mentioned the old joke about how 'this is the year Unix is finally going to take off'. Well, here's a prediction that you can read here first.

This is the year that Unix finally starts to die.

And about time, too. ■