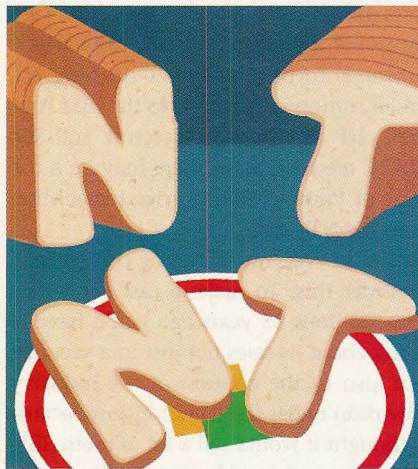


NT: The Greatest Thing Since Sliced Bread?

IT'S OFFICIAL. Windows NT is finally out of the oven. The pricing has been set and delivery dates have been promised. Those promises (back in May) were that the base system would ship within 60 days and the Advanced Server version would ship 30 days later. So there's every chance that by the time you read this, there'll be piles of Windows NT packages at your local computer store.



After a false rumor that Microsoft might price NT at a low-ball \$99, some people were probably disappointed when the announced prices were quite a bit higher—\$495 for the desktop version; \$295 if you're upgrading from Windows, DOS or OS/2. Wariness is understandable: Think back to the OS/2 fiasco when the product simply refused to move until IBM slashed its \$320 price down to a rock-bottom \$49. Is this another case of snatching defeat from the jaws of victory?

I don't think so. The key difference here is that NT,

unlike OS/2, is not the only product in the family. Right around the corner is the Chicago technology that Microsoft will be using in DOS 7.0 and Windows 4.0. Earliest betas for software developers are scheduled for early fall and products based on it should be out in 1994.

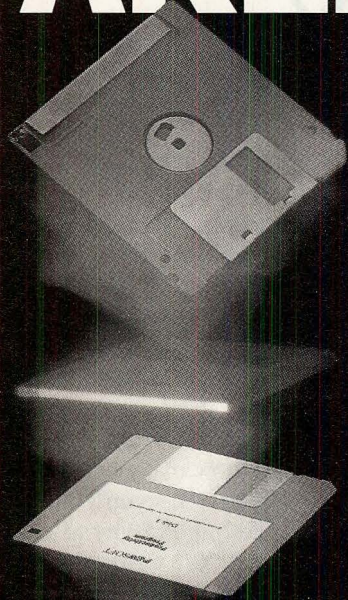
Chicago is a rewrite of the same Win32 API (application programming interface) seen on NT. Compared with NT, it leaves out the security (how many of us need government-certified C2 security on our personal machines?), portability to RISC processors and the OS/2 character-mode and POSIX subsystems. And it wouldn't surprise me if Unicode support were trimmed out or if the networking were unbundled. What Chicago leaves *in* is 32-bit flat virtual memory, preemptive multitasking, threads, pipes and semaphores—all the really good stuff in Win32. That becomes Windows 4.0. Subtract the graphical user interface and you get DOS 7.0.

Is there anyone left who wonders why developers are so excited about NT? It's not just because NT is expected to do extremely well. It's also because with Chicago, most native 32-bit NT applications should run under Win 4.0 right out of the box; character-mode apps should run even under DOS 7.0. This is a page right out of the IBM marketing manual: Let customers buy as much hardware as they want but provide assurance that all the same software will run on every system.

Back to the future

Remember the System/360 from 25 years ago? Every machine from every vendor was different; no two even had the same instruction set or operating system. Getting even minuscule improvements in performance meant changing to a different system. System/360 was revolution-

HEADS, IT'S REAL. TAILS, IT'S FAKE.



It's your choice. A little more than 50 percent of all business software in use today is pirated. If you buy it, you could end up with virus-ridden, phony software that has no documentation or product support.

Selling or copying pirated software without authorization is against the law, with severe criminal and civil penalties including imprisonment of up to five years, fines of up to \$250,000, or both. If you suspect the sale or use of pirated software, call the BSA Anti-Piracy hotline:

BSA (800) 688-BSA1 (2721)



© 1993 Business Software Alliance. All rights reserved.

Douglas A. Hamilton

ary: It signified the first time a vendor acknowledged the problem and offered a solution. Back then, someone might have pointed to IBM's ultra-high-end Model 91 and asked how it could possibly be successful. So big, so expensive—who would buy it? Similarly, today NT is questioned for being too big, too slow, too expensive.

To set the record straight, NT is not too big, slow or expensive, even on a modest 486 machine. On a \$7,000 DEC Alpha, NT absolutely screams! That's less than I paid for a 16MHz 386 six years ago. Also, Pentiums are appearing, the new Mips R4400 promises to be a screamer running NT, and if rumors of NT ports to the IBM PowerPC, HP Precision Architecture and Sun SPARC are true, soon there may be a torrent of competitively priced machines running NT—fast.

For an even more striking comparison, consider this: To make a case for selling top-end 360s 25 years ago you'd have to think about its sales not just in a vacuum, but also in the context of the industry. IBM didn't offer the Model 91 just because it thought it would sell a lot of them. IBM also did it so there'd be a Model 91—something so powerful that no matter what a customer's needs were, IBM could satisfy them. Likewise, NT offers assurance that there is a way to go if you outgrow regular Windows or DOS—and with the rate of change in this industry, that's a certainty.

We can also see Microsoft applying another standard strategy of hardware vendors; namely, building the high-end, full-function version first and then cutting its cost. It's a great strategy because it forces the engineers to push the envelope of what's possible. And that high-end positioning on the first product rubs off on the cost-reduced version that comes later.

Not too big a price to pay

The NT pricing did two things: First, it created a price umbrella for Windows 4.0 and DOS 7.0. If NT cost \$99, how would Microsoft price DOS? Would they pay you to take it? Second, it was a signal to IBM that Microsoft would not participate in the absurdly destructive price war that IBM has been waging with OS/2.

We all want a good deal, but there really is such a thing as too good a deal. Unless you buy at a close-out sale, it's no good

if the vendor isn't paid enough to have the incentive or ability to do a good job. By all estimates, IBM's \$49 price for OS/2 couldn't cover its manufacturing costs or development, nor could retailers profit. If OS/2 were the pricing model for advanced operating systems, there would be no incentive to develop better ones.

Pricing for the system also sets expectations for the applications. As a major applications vendor, Microsoft is aware that people think about the price of an application in relation to what they paid for the system. (But I've also heard other vendors express a sense of relief at the NT pricing.)

All in all, Windows NT and Win32 are positioned to win.

Coming in the next decade

At Spring Comdex, DEC exhibited an exciting compiler technology for translating existing off-the-shelf Intel x86 application binaries to Alpha binaries and optimizing the result. The technique does not depend on having access to the original source. Used on an Intel program that produced Mandelbrot patterns, the Alpha version outran the original running on the 486 beside it; that's far better than with the standard Insignia x86 emulation code that Microsoft currently packages with RISC versions of NT.

DEC's been using this translation technology for converting VAX applications into native Alpha versions for some time, but this was the first time it was shown on an Intel application. If DEC can refine this technology sufficiently for general release, it answers any concern about the performance of Intel apps on a non-Intel platform. Intel, beware: The next 10 years may be a lot different from the last.

Finally, like all things, this column comes to an end. You'll still see me in print here in *WINDOWS Magazine* from time to time, but this is my last regular column. Next month, John Ruley will take over this space with a new how-to column called "eNterprise Windows." Thanks for the long and enjoyable run. ■

Douglas Hamilton is president of Hamilton Laboratories (Wayland, Mass.) and author of the Hamilton C Shell. Reach Douglas on WIX as hamilton or care of Editor at the address on page 16.