

# Why wasn't the Windows 95 shell prototyped on Windows NT?

[devblogs.microsoft.com/oldnewthing/20120102-00](http://devblogs.microsoft.com/oldnewthing/20120102-00)

January 2, 2012



Raymond Chen

Carlos wonders why the Windows 95 shell was prototyped as 16-bit code running on the still-under-development 32-bit kernel, USER, and GDI as opposed to being prototyped as fully 32-bit code on Windows NT. There were a number of reasons, some good, some bad. One reason was that the Windows 95 shell was being developed by the Windows 95 team, which was an outgrowth of the Windows 3.1 team. That meant that they had Windows 3.1-class hardware. And the hardware requirements of Windows NT were significantly higher than the hardware requirements of Windows 3.1. Here's a table for comparison:

Platform	RAM	
	Minimum	Recommended
Windows 3.1	2MB	4MB
Windows NT 3.1	12MB	16MB
Windows 95	4MB	8MB

The Windows 3.1 team adhered to the principle that the team members' machines, as a general rule, were as powerful as the recommended hardware requirements. If you asked really nicely, you were permitted to exceed that, but not by too much, with one notable exception. Think of it as performance dogfood. If Windows was unusable on the stated recommended hardware requirements, the entire team felt it because that's what they were running. (When Windows 95 shipped, my primary machine was a 486/DX50 with 8MB of RAM. My test machine was a 386 with 4MB of RAM. The combined computing power and storage capacity of all the machines in my office is now exceeded by your cell phone.) Okay, so you just finished Windows 3.1, and all of the team members currently have 4MB machines, with a few lucky people that have a whopping 8MB of RAM. If you decided to do your prototype work on Windows NT, that would mean tripling the amount of memory in most of the computers just to meet the *minimum* requirements for Windows NT. And you can't say that "Well, you would have had to pay for all that RAM anyway," because look at

that chart: Windows 95's *final* hardware requirements were still lower than Windows NT's minimum! Spending all that money to upgrade the computers for something that was just a temporary situation anyway seemed like a bad way of spending your hardware budget. From the software development side, prototyping the new shell on Windows NT was not practical because Windows 95 introduced a whole bunch of new features to Win32, features which didn't exist in Windows NT. Part of the goal of the prototype was to exercise these new features, things we take for granted nowadays like `RegisterClassEx` and `WM_MOVING` and the Close button in the upper right corner. Those features didn't exist in Windows NT; if you wanted to develop the prototype on Windows NT, somebody would have had to port them and build a special "throwaway" version of Windows NT for the Windows 95 team to use. That means devoting some people to learning a whole new code base and development environment (and buying lots more hardware) to add features that they had no intention of shipping. It was much more cost-effective to do the prototyping of the Windows 95 shell *on Windows 95*. You could see if a design led to poor performance and deal with it before things went too far in the wrong direction. You could use those fancy new functions in kernel, USER, and GDI, which is great because that meant that you would start finding bugs in those fancy new functions, you would start finding design flaws in those fancy new functions. If the shell team needed a new feature from the window manager or the kernel, they could just ask for it, and then they could start using it and file bugs when it didn't work the way they wanted. All the effort was for real. Nothing was being thrown away except for the stuff inside `#ifdef WIN16` blocks, which was kept to a minimum. All through the shell prototyping effort, the code was compiled both with and without `#define WIN16`, and as the kernel team worked on supporting 32-bit processes, they had this program sitting there waiting to go that they could try out. And not some dorky *Hello world* program but a real program that does interesting things. (They couldn't use any of the Windows NT built-in programs because those were Unicode-based, and Windows 95 didn't support Unicode.)

Maybe those were bad reasons, but that was the thinking.

[Raymond Chen](#)

**Follow**

