

Microsoft Corporation
16011 NE 36th Way
Box 97017
Redmond, WA 98073-9717

Tel 206 882 8080
Telex 160520
Fax 206 883 8101

Microsoft Memo

TO: Jeremy Butler, Michael Cooper, Martin Dunsmuir, Bill Gates, Frank Gaudette, Rob Glaser, Jeff Harbers, David Jones, Joachim Kempin, Adrian King, Jon Lazaruz, Gordon Letwin, Paul Maritz, Nathan Myhrvold, Peter Neupert, Scott Oki, Jeff Raikes, Darryl Rubin, John Sabol, Jon Shirley, Charles Simonyi, Tandy Trower, Mark Ursino, Russ Werner, Greg Whitten, Min Yee, The Waggener Group

FROM: Steve Ballmer

DATE: May 23, 1988

RE: **WINDOWS VERSUS PRESENTATION MANAGER:
A DEVELOPERS PERSPECTIVE**

Attached is a fantastic article written by David Weise.

Plaintiff's Exhibit
8089
Comes V. Microsoft

**RESTRICTED
CONFIDENTIAL**

X 174510
CONFIDENTIAL

12026066

Microsoft

Windows versus Presentation Manager: A Developer's Perspective

Jonl
 Bills
 CC: Jeff
 Jonl
 Eric Staff
 by David Weise Systems Staff
 TWG
 Joffr

This quickly thrown together evolving epistle is one Windows\PM developer's ruminations on Windows, the Presentation Manager, and where each is headed. I have tried to keep the technical argot to a middling level, so that all readers may learn from this reconnaissance of runes. I'll let you know at the bottom if I think I succeeded.

As quick background material, Windows consists of three pieces that sit on top of DOS. These are the kernel (OS functionality), GDI (graphics) and User (window manager). Under PM the Windows kernel gets replaced by OS/2, GDI gets replaced by the engine, and User mostly got ported right over. To keep this article manageable I'll mostly ignore OS/2, give User just a short summary (User has evolved from Windows to PM, the code has matured and new functionality has been added), and concentrate on dramatic differences between Windows and PM (which basically boil down to more memory and real graphics capabilities).

We start with the obligatory comparison chart:

	Windows	PM
Advantages:	memory usage	memory available richer graphics
Disadvantages:	memory available	memory usage

The largest complaint against Windows is that 640K is really not a lot space to write a complex and interesting application in, especially since Windows takes up a lot of space itself. We can counter this argument somewhat by pointing out that Excel and PageMaker are complex and interesting applications that run just fine. In addition since Windows was originally designed to run on a 256K machine (yes, on a 256K machine you could even bring up Write and print from it!) Windows itself occupies a minimum of space. If you have a machine with an extra 1Meg of expanded memory all of that memory goes to applications. This is what I mean by Windows having better memory usage. In theory PM will not have memory restrictions since OS/2 can address up to 16 Megabytes. However you have to have at least 2 Megabytes in your machine before you can even consider running a complex and interesting PM application.

The graphics engine in PM puts the graphics capabilities of Windows to shame. This did not come cheap, in either code size or manpower. To start with, the graphics capabilities are broken into two different parts in PM: the Engine which Microsoft is responsible for, and GPI which IBM Hursley is responsible for. The Engine is about 200K of fine hand-tuned assembler and GPI is about 225K of code. This total of 425K is to be compared to Windows GDI which is about 100K. In addition the display driver went from 30K to 150K, again all of it in assembler. So we're talking 575K to 130K, PM versus Windows.

The drawing primitives in Windows are lines, ellipses aligned along the coordinate axes, and boxes. PM's primitives include lines, ellipses oriented any which way, boxes, three point arcs, bezier curves (a type of spline), and any conic you wish to name. Just as importantly PM makes full use of 16 colors on EGAs, Windows only uses 8 colors (which is why it always has a Romper Room brightness to it). Concepts from PostScript have also been liberally thought about, in particular the concepts of "paths" and "areas" have been implemented. Paths and areas are a way of grouping together drawing primitives to make more complex primitives. A good example of this is trying to draw large letters. Let's say you want to make a "G" the size of this page. You could draw the outline using lines and arcs end to end and store this outline for future use. Whenever you needed a "G" you could simply make a single call to draw the path that represented the outline. This ability to build more

12026067

X 174511
CONFIDENTIAL

RESTRICTED
CONFIDENTIAL

complex objects out of primitives is a major advantage of PM over Windows. In fact this ability was viewed to be so important that there are three additional ways to build complex objects! Without going into detail they are Journaling (done by the engine), Segments (done by GPI), and MetaFiling.

Not only are the graphics richer, but they are also done right. As an example on a Macintosh use MacPaint to draw different ellipses. You will notice double pixels, witches hats, pimples, and worst of all missing pieces from some ellipses! Things are not much better on a Sun. Our lines and curves are the best to found anywhere, and they merely are outward evidence of internals that are just as polished. But let me stick to comparing just two products instead of 5 products.

What applications will benefit from better graphic capabilities? To be fair, spreadsheets and word processors will probably look the same. Page composition and desktop publishing will improve but only to the extent the the graphics they push around with text will be better. The true power comes to applications that are drawing or design oriented; Windows PaintBrush and Windows Graph are examples of these. AutoCAD is an example of the type of application that will simply shine on PM -- Computer Aided Design. (CAD stands for Computer Aided Design, CAM is Computer Aided Manufacturing, and CASE is Computer Aided Software Engineering - or some such drivel.) The richer graphics should convince the CAD cabal to seriously consider porting their applications to PM. To put it bluntly Windows just doesn't have the power for a sophisticated CAD application, which is why no one is developing one. Most CAD applications are run on midsize computers such as Vaxen or expensive workstations from Sun. The tools Microsoft has yet to bring to the party are drivers for high resolution displays, and a story how 64K segment limits are an advantage over a linear address space. If we can do this then Sun Workstations will lose some of their reason for existence. And this brings us to an important point that deserves its own paragraph.

PM is not just user friendliness brought to OS/2. PM with OS/2 is the beginning of the attack on workstations and mini-computers by Microsoft. We all know that personal computers are becoming more powerful every year, that the performance per price just keeps getting better. The power in an IBM model 80 is equivalent to the power of mid-size computers of a decade ago. Furthermore Microsoft is dedicated to the concept of a computer in every home and office desk (and of course we're gonna sell the software on all of these computers!). Yet a simple extrapolation is interesting; in a decade we're going to have present day mid-size computers in our homes and offices. But Microsoft doesn't presently write software for these systems, so we'd be squeezed out. Of course this analysis is somewhat simplistic but it gets the point across. A different way of looking at things is that as personal computers get more sophisticated they should take over the workstation/midsize market and that it is only natural Microsoft should dominate that software market as well. Any way the point is this, PM plays a very important role expanding our markets, and is the future.

Is Windows going to be blown away by all of this? ABSOLUTELY NOT!! Windows answers most of today's problems today. Not only are there sophisticated applications available for Windows (such as Excel, PageMaker, Opus, and Notes from Lotus), but there is an increasing number of support/utility applications coming out for Windows. Our marketing people at Spring Comdex were suprised at the number of Window applications out there that they didn't know about. We have sold an incredible number of Windows at the retail level, at the OEM level, and in the international market. We have sold an amazing amount of SDKs. The functionality and integration of present Windows applications are probably sufficient to satisfy the needs many end users. As pointed out above spreadsheets and word processors are going to look pretty much the same under PM and Windows. Many people are not going to need the power of OS/2 for a long while. Nor are many going to be able to justify the expense of OS/2 and PM. On a machine with 3 or less megabytes of RAM Windows will simply out perform PM in terms functionality of the number of applications you can run.

Other reasons the beachhead of Windows will not be washed away by the sands of time are Windows386 and the Microsoft office. Windows386 allows Windows applications and ANY old DOS applications to run side by side. Windows can do this for only a small subset of DOS apps. Windows386 also does preemptive multi-tasking of old applications. For those users who need this functionality now Windows386 is the only real solution. The Microsoft office is what all those people in building 3 are doing. They are building sophisticated applications that are meant to blow away the competition, that to work together well by having the capability of communicate with each other, and that are all Windows applications.

Windows is the present for as long as it takes for the present to become the future.

12026068

RESTRICTED
CONFIDENTIAL

(continued on next page)

X 174512
CONFIDENTIAL

In conclusion we can see that my title was a straw man. What I have tried to develop here is that Windows and PM are not in competition, but that rather PM is the logical extension of Windows to OS/2, an extension that makes use of the addition functionality present in OS/2. There is a large class of computer users who will never switch to OS/2 in the near term simply because DOS 3 and 640K adequately meet their needs and budget. Windows is the solution for these people. We should not think of Windows VERSUS PM, we should be thinking Windows AND PM.

This discussion has gone on longer than I expected, and yet I was only able to touch on highlights and generalities of Windows and PM. There are many more interesting technical, marketing, and personal points (such as PM applications that really understand the concept and power of networks and multiprocessor machines) about each that perhaps others can talk about in future articles for DosGetNews(). □

X 174513
CONFIDENTIAL

12026069

RESTRICTED
CONFIDENTIAL