

MEMORANDUM

TO: Frank King
Rick Faulk
Frank Ingari
David Reed
Bruce Johnston
Al Stoddard

FROM: Semmes Walsh

SUBJECT: Windows vs. OS/2 Comparison

DATE: October 18, 1989

Two documents are attached, which compare OS/2 and Windows. Both are part of a Lotus/IBM project to develop a presentation for use within IBM, conveying the relative advantages of OS/2.

The first document is what we're providing to IBM. Entitled "Comparison of OS/2 and Windows", it compares the two environments from a technical as well as a business perspective. At its heart is a comparative chart listing OS/2 and Windows 3's differences and their impact on developers and end users. The chart is followed by a non-technical explanation of each of the differences.

The material for the first document was compiled primarily from discussions with Ray Ozzie and David Reed. Thanks go to David for editing and reviewing the paper as a whole.

The second document, entitled "Why OS/2", is the current version of the IBM internal presentation, which makes use of the material we have provided. It is IBM Confidential material so please use discretion in distributing it.

Please let me know if you have questions or suggestions about either of the documents.

Semmes



LOTUS
001391

CONFIDENTIAL IBM 0610002243

Plaintiff's Exhibit

5065

Comes V. Microsoft

TRM 751023R75B

COMPARISON OF OS/2 AND WINDOWS

SUMMARY

- OS/2 is the best choice for the modern application environment.

Personal computers are now the platform for mission critical line of business applications, supporting multiple applications on a single PC, with transparent network communications.

OS/2 provides the required enabling technologies for this application environment today, and these enabling technologies will be developed even further as OS/2 matures.

Windows provides insufficient enabling technologies, and Windows's future expansion is severely constrained by DOS's limitations and Windows's 4 year old internal architecture.

- There is no hardware platform where Windows is clearly better than OS/2

Windows requires only 1 to 1.5Mb less RAM than OS/2 to support multiple concurrent applications. (Note: recent performance benchmarks indicate that the difference, if there is any, is even smaller than this). As a percentage of total RAM, this difference decreases as more concurrent applications are added.

Windows's weaknesses relative to OS/2 are most pronounced on the 80286, despite its positioning as the superior low end GUI environment.

Two of the most widely touted features of Windows 3 - paged virtual memory and multiple concurrent DOS sessions - do not work on a 80286.

- User and support costs are lower without Windows

Windows applications can interfere with each other. This can cause data loss, and it raises support costs.

If OS/2 is the final destination, going to Windows only adds cost, because migrating out of Windows into OS/2 requires training and applications upgrades.

- From a developer's perspective, OS/2 is the preferable environment

OS/2 provides better tools for complex applications. All Lotus development is done under OS/2; even for DOS or Windows products.

OS/2 is easier and safer for corporate developers to use for almost any application.

The differences between OS/2 and Windows equate to sizeable development costs in the migration between environments, in either direction.

Market Development

Lotus Strictly Private

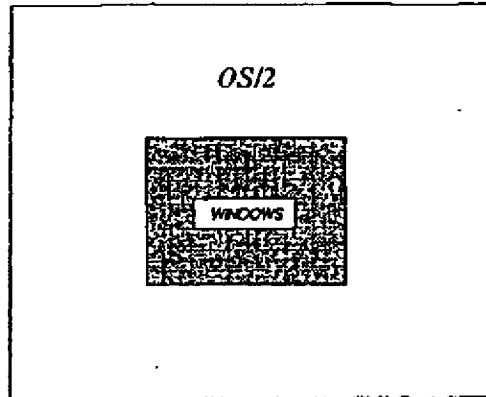
LOTUS
001392

CONFIDENTIAL IBM 0610002244

IBM 7510238759

COMPARISON OF OS/2 AND WINDOWS

OS/2 - WINDOWS PROFILE



- **Overview of the differences** (Specific differences and what they mean appear on following pages.)
 1. OS/2 capabilities are a "superset" of Windows capabilities.
 2. Even the shared capabilities of OS/2 and Windows are implemented differently.
- **Implications for applications migration**
 1. Applications that are written to take advantage of OS/2's power would be extremely difficult and costly to retrofit into Windows.
 2. Windows applications require significant reengineering to run in OS/2, *even if OS/2 compatibility is a design criterion*. As an example, in a Lotus OS/2 product originally built under Windows (and designed from the outset for OS/2 portation), 50% of the program code had to be reviewed and/or rewritten in order to allow the product to run under OS/2.
 3. Even simple OS/2 applications, which use only the capabilities that have counterparts in both OS/2 and Windows, would require pervasive changes in order to run in Windows. Virtually every line of code would have to be changed. Estimates for 3 Lotus OS/2 products run from 20 to 100 man years to convert to Windows.

Market Development

Lotus Strictly Private

LOTUS
001393

CONFIDENTIAL-IBM 0610002245

IBM 7510238760

COMPARISON OF OS/2 AND WINDOWS

DIFFERENCES BETWEEN WINDOWS 3 AND OS/2

	Architectural Compatibility	Code Compatibility	Product Impact	User Impact
System installation / configuration	-	-	-	X
Multiple DOS sessions	-	-	-	X
Private memory / address space protection	2	2	3	X
Task scheduling, preemption, & threads	3	3	0/3	X
Shared memory	3	3	0/3	X
File system	2	2	2	X
Inter-program communications (IPC)	3	3	3	X
Window management	2	3	3	X
Graphics	3	3	3	X
Font metric details / outline fonts	3	2	1/3	X
National language support	3	2	1/2	X
Timers	3	3	0/1	X
Coordinate system origin and direction	3	3	3	-
Metrics	1	1	0/3	-
Dialog definition	1	3	2	-
Icons, bitmaps, colors, resource files	2	2	2	-
Dynamic link libraries	2	2	2	-
Input model	1	2	2	-
Dialog boxes	1	2	2	-
Controls	1	2	2	-
Printing	2	2	2	-
Call names	1	2	2	-
Positioning of characters relative to drawing position	3	2	1	-
Clipboard	1	2	1	-
Clock	1	1	0/1	-
Event messages	2	2	2	-
Subroutine linkage	1	3	2/3	-

Architectural compatibility:	Code compatibility	Product Impact	User Impact
"Are the basic mechanisms the same?"	"How much recoding is required?"	"What is the impact on development resource requirements?"	"Does this make the application look or behave differently to the user?"
1 = same; 2 = derived (similar but different) 3 = different	1 = Cosmetic difference 2 = Minor rethinking required 3 = Major impact	1 = Low 2 = Medium 3 = High	X = unavoidable impact on user
0/... = Has impact only if the application uses this capability of OS/2 or Windows. - = Not applicable			

Market Development

Lotus Strictly Private

LOTUS
001394

CONFIDENTIAL IBM 0610002246

IBM 7510238761

COMPARISON OF OS/2 AND WINDOWS

What the differences mean

- TO USERS -

The following differences make applications look or behave differently to the user as well as affecting the developer. These are the differences that are marked with an X in the "User Impact" column of the chart.

- **System installation / configuration**

For the end user, configuring and installing Windows is very different from configuring and installing OS/2. This will cause end user support costs to rise significantly when users migrate from Windows to OS/2.

- **Multiple DOS sessions**

The 60386 provides hardware support for multiple concurrent DOS sessions. Windows 3 takes advantage of this capability, whereas the current releases of OS/2 do not. For this reason, Windows 3/386 users with 80386 machines can run multiple DOS applications concurrently. However, these DOS sessions are prone to the weaknesses of Windows task scheduling and address space protection, relative to OS/2 (these are described individually below.)

- **Private memory / address space protection**

OS/2 provides each application with its own "protected" memory space, whereas Windows allows all applications to use the same memory space. This creates the likelihood of inter-application conflict in Windows, which does not exist in OS/2. This issue is separate from the use of "protected mode" memory, and is true even when Windows 3 uses "protect mode" memory.

OS/2's superior protection between applications makes it a preferable environment for developers, because it creates multiple "interference free" environments in which the developer can work concurrently. For Windows users, this quality in Windows causes system crashes and data loss that are not possible under OS/2.

- **Task scheduling, preemption, and threads**

Under OS/2, the operating system allocates time to each application and to each process within an application. This is called "preemptive" multitasking, and it guarantees system time to every process, with the priority between processes ultimately being determined by the user (using config.sys).

Windows uses non-preemptive multi tasking, so that Windows applications contend with each other for system time. In this environment, one application or process can seize the system and deny system access to other applications until that application or process is finished. Similarly, when a process that has control of the system depends on the completion of an external event (such as receipt of a message from the network), this one process can hold up the whole system while it waits for the event to occur.

As a result, Windows users often find that their applications really don't "multi-task" under Windows. This weakness in Windows is especially disruptive in networked environments.

- **Shared memory**

Shared memory permits communication between different applications or between different parts of the same application. As contrasted with DDE, which is also used to communicate between applications, shared memory is used to communicate large banks of data quickly, whereas DDE is used for relatively slower, lower volume communication.

LOTUS
001395

Market Development

Lotus Strictly Private

CONFIDENTIAL IBM 0610002247

IBM 7510238762

COMPARISON OF OS/2 AND WINDOWS

OS/2 has a mechanism for dealing with shared memory that is different and much easier to use than Windows's. As a result users find that Windows performs very poorly in an environment where large databases must be shared instantaneously between applications (e.g. on a database server with many clients).

- **File system**

OS/2 provides a number of advantages over Windows's file system, which is limited to what DOS provides. Improvements in OS/2 include longer file names (256 characters in OS/2 1.2 as opposed to 8 in DOS/Windows), and a high performance file system, which result in increased ease of use, enhanced DBMS and network performance, and increased reliability and failure recovery.

- **Inter-program communications**

OS/2 supports Named Pipes, a inter-program communication protocol that facilitates network communications between applications or between parts of a single application. Named Pipes allows for local/remote transparency, so that it is not necessary for to decide in advance whether parts of an application will be run on the same or on different machines. Support of Named Pipes makes OS/2 an ideal platform for certain classes of applications that rely on this type of communications, especially client/server applications.

The capabilities that Named Pipes provides to OS/2 are not available in DOS; they are very difficult and sometimes impossible to replicate in Windows. As a result, Windows is an inappropriate platform for certain client / server applications.

- **Window management**

OS/2 provides tools for managing windows, control bars, and other window elements which are more structured and more powerful than the tools in Windows. As a result, it is easier for developers to create customized, SAA compliant windows under OS/2 than under Windows. As a result, users will find that customized applications developed under OS/2 will tend to look more uniform than if they had been developed under Windows, unless their Windows developers are highly expert and attentive to inter-application consistency.

- **Graphics**

OS/2 provides tools that make it easier to manipulate advanced geometric shapes or complex collections of shapes. As a result, graphics programs are far easier to develop under OS/2 than under Windows. This translates to the user as slower performance on many graphics applications.

- **Font metric details / outline fonts**

The tools for describing font attributes in OS/2 are at the same time more general purpose, and more sophisticated than they are in Windows. As a result, users will find it much easier to get typeset quality output from OS/2 applications than from Windows applications.

- **National language support**

OS/2 much more powerful tools than Windows for internationalizing applications, especially in areas where the application interacts with the operating system. As a result, it takes longer and is more costly to deliver foreign language versions of Windows products to end users.

- **Timers**

Windows handles timing through event messaging, whereas OS/2 takes advantage of threading and preemptive multitasking, which are not available in Windows. As a result of this difference, Windows is not an appropriate environment for real time applications such as process control or robotics.

Market Development

Lotus Strictly Private

LOTUS
001396

CONFIDENTIAL IBM 0610002248

IBM 7510238763

COMPARISON OF OS/2 AND WINDOWS

What the differences mean

- TO DEVELOPERS -

These differences affect developers and increase developer costs of migrating, but they can be hidden from users.

- **Call names and subroutine linkage**

Call names and the arguments they take are different in OS/2 and Windows, (call names and arguments are key classes of "verbs" and "nouns" in a programming language). Calling sequences are sets of instructions that are used to invoke subroutines. Windows and OS/2 use different calling sequences. Windows uses special calling sequences which are only available on Microsoft compilers.

These differences make it necessary to translate substantial pieces of a program from one environment in order to allow it to run in the other environment.

- **Clipboard**

OS/2 and Windows require different coding techniques.

- **Clock**

The mechanism by which a program accesses the system clock is different in OS/2 and Windows.

- **Controls**

The style of coding required for managing controls - scroll bars, radio buttons, minimize/maximize icons, and the like - is different and easier in OS/2.

- **Coordinate system origin and direction**

Windows maps screen or page locations beginning at the lower left; OS/2 begins at the lower right. This and other differences in location referencing have a major development impact.

- **Dialog boxes**

OS/2 includes more powerful tools for managing dialog boxes than Windows.

- **Dialog definition**

The style of coding required for managing dialog boxes is different and easier in OS/2.

- **Dynamic link libraries**

Dynamic link libraries are sets of program code that can be dynamically loaded as needed when the application is running. In OS/2 and Windows, there is a significant difference in the programming required to manage dynamic link libraries, which may have an effect ranging from major to minor, depending on the specific libraries being used.

- **Event messages**

When events such as keystrokes, mouse movement, communications, or DDE messaging occur, Windows and OS/2 respond differently. They generate different types of messages, carrying different information in different formats. As a result, a developer has to change the way the application responds to events, in order to move it from one operating system to the other.

Market Development

Lotus Strictly Private

LOTUS

001397

CONFIDENTIAL IBM 0610002249

IBM 7510238764

COMPARISON OF OS/2 AND WINDOWS

- **Icons, bitmaps, colors, resource files**
The mechanisms and tools available to deal with icons, bitmaps, color palette management (how an application selects and uses colors) and resource files all differ in their syntax and presentation.
- **Netbios**
Windows and OS/2 use different call names to make use of Netbios.
- **Positioning of characters relative to drawing position**
This is another example of the issue described under coordinate system origin and direction, which is described above.
- **Printing**
The capabilities and properties of printer drivers in OS/2 and Windows differ considerably. Each system provides a different set of tools for setting up printer default parameters and for finely controlling the printer. In order to get high quality output for a given printer, the developer has to write specially for the printer/operating system combination, and the resulting driver is not portable to the other operating system.
- **Subroutine linkage**
See description under call names, above.

Market Development

Lotus Strictly Private

LOTUS
001398

CONFIDENTIAL IBM 0610002250

IBM 7510238765