

To: Alex Morrow/CAM/Lotus
cc: John Manopoli/CAM/Lotus
From: Noah Mendelsohn/CAM/Lotus
Date: 01-04-96 05:19:59 PM
Subject: Microsoft Support for OCX Development

Earlier this afternoon you asked me for an update on our experiences with support and documentation by Microsoft for their OCX technology. A list of our concerns from earlier this year is contained in the attached note, dated 2/9/95. As you know, most of my personal focus in the period since then has been on other matters, but I have this afternoon reviewed the situation with several of the developers here at Lotus who are working with OCX. They and I agree on the following summary of our experiences in the period since February, 1995.

- The lack of freely licensed code and reference materials for native (as opposed to MFC-based) OCX server and container implementation was and is a significant impediment to both the planning and the implementation of our OCX based products. One of our developers estimated that he would have finished his year's work at least two months earlier, had the level of documentation for OCX been equivalent to that provided for OLE 2.0. Lack of detailed information has discouraged us from implementing more comprehensive OCX support in the products that were developed during 1995; this is especially true of OCX container functions. As you know, licensing restrictions prevent many of our developers from referring to the MFC sources for information on OLE or OCX development.
- Microsoft has released some useful new materials relating to OCX development, and they have been helpful in making those materials available to us. These materials include a draft of a new book on OLE Controls by Adam Denning, and conformance guidelines for OCX implementation. While helpful, none of these directly address our need for freely licensed reference implementations of container and server functions. The book, in particular, relies heavily on MFC, and specifically suggests that the reader consult the MFC source code in certain areas; as noted above, the MFC license severely limits our ability to work with the source. Within the past month, Microsoft did release a simple, non-MFC based class framework for OCX sever development; the framework was provided to attendees of the Microsoft Internet Design Preview in early December. I have not yet had the opportunity to review that code, and cannot comment on its significance. In any case, it comes too late to affect the products that we have spent the past year designing and building.
- Mark Colan, one of our OCX developers, says that Mike McKeown of Microsoft Developer Relations informally promised that we would receive source for an OLE control container to be provided as part of a Microsoft validation suite. Mark says that we have not received that code from Microsoft, and I suspect that it is not yet available. Such a validation container would indeed be very useful, but it would probably not be a complete substitute for a production quality reference container application. We're still hoping to receive it at some time in the future.

13
+ mso
counting

My dealings with representatives from Microsoft remain cordial. They have been generally helpful with arranging for our attendance at design previews for various Microsoft system products, and also with getting us access to OCX support when available. Microsoft has recently shifted much of their corporate focus to the Internet, and they seem to be doing a better job of providing early access to code and documentation in that area. We have had representatives at recent Microsoft Internet Architecture Design Previews. The Internet code that we've received, while still shaky, appears to be reasonably up-to-date. I hope this is indicative of improvements in the level of support that we'll be receiving in the future. Unfortunately, the lack of appropriate OCX documentation was and is a significant problem.

I apologize for not being able to pull together a more detailed or authoritative analysis on short notice. I have tried to give a fair and balanced picture of events through 1995, but it is possible that I missed

PLA.
EXHIBIT NO. 8

something of significance. Please let me know if you need any further information.

Noah

=====
SECTION WITHHELD ON THE
BASIS OF ATTORNEY CLIENT
OR WORK PRODUCT PRIVILEGE

To: John Landry, Ilene Lang, Tom Lemberg
cc: Alex Morrow, Ron Sandstrom, Allen Olsen
From: Noah Mendelsohn
Date: 02/09/95 03:48:34 PM
Subject: Microsoft OCX Support: Is the Playing Field Level?

This note summarizes my concerns regarding Microsoft's support for ISV's implementing the new OLE Controls (OCX) technology.

OLE Controls, which are implemented as enhancements to OLE 2.0, are emerging as the key component architecture for the Windows operating system platform. Microsoft has also disclosed that OLE controls will be used as the basis for the desktop user interface in Cairo, the successor to Windows NT.

Microsoft has publicly committed, on numerous occasions, to ensuring a fair separation between the application and system groups at Microsoft. Specifically, they have promised to provide equivalent operating system API support and documentation to application developers working inside and outside Microsoft. I am concerned that these commitments are not being met in the case of OCX, and that Lotus and other ISVs are being put at an unfair competitive disadvantage. As you know, I have been responsible over the past two years for our technical contacts with Microsoft regarding OLE 2.0 and related technologies. Though some concerns regarding OLE 2.0 documentation and development process remain unresolved, the support we received on OLE 2.0 was generally professional, detailed, and in most cases responsive. Extensive documentation and sample code was provided for most OLE 2.0 features, without onerous licensing restrictions. I and a number of members of my group developed productive working relationships with our counterparts at Microsoft, and most of the information we were given has proven over time to be correct. Those relationships are based on the assumption, which I believe to be correct, that Microsoft and Lotus have a shared interest in seeing the features of Microsoft's operating systems exploited correctly and consistently in Lotus' products.

Recently, a number of concerns have arisen regarding Microsoft's willingness and ability to extend such support to the new OLE Controls technology. For the reasons listed below, I believe that Microsoft

CONFIDENTIAL IBM 0410075025

application developers have been given earlier and more detailed access to OCX specifications than we have had here at Lotus. These are serious concerns, and I hope that we can address them with Microsoft promptly:

- **Licensed Microsoft Tools Code is the Only Available Sample for OCX Server Implementation**

When OLE 2.0 was released, it was accompanied by an extensive reference manual in two volumes, an additional guidebook by Kraig Brockschmidt, and a number of reasonably detailed sample programs for both container and server functions. Even with that level of information, developers inside and outside Microsoft struggled to build robust implementations of OLE 2.0. Microsoft also released a version of their Foundation Classes (MFCs), which simplified implementation of OLE 2.0. The source code provided with MFC also served as a useful sample OLE implementation for some developers outside of Lotus, but licensing restrictions on the MFC source prevented its use for that purpose within Lotus. The other samples provided by the Microsoft operating system group proved adequate for most purposes, and we received reasonably good direct support from Microsoft when additional information was needed.

With OLE controls, the level of support and documentation from Microsoft has changed dramatically for the worse. MFC version 3.0 is now the *only* production quality example of an OCX server implementation available outside of Microsoft. Furthermore, the MFC's continue to be governed by licensing restrictions which prevent their use for many purposes within Lotus. Microsoft has effectively chosen to use a restrictively licensed product of their tools division as the only documentation for a critical new operating system feature.

- **Inadequate documentation of OCX Container API**

The only OCX container sample code that's available is, by Microsoft's own description, incomplete and inadequate as a guide to building production quality products. Nonetheless, Microsoft is shipping container implementations as part of their Visual C++ and Access products, and we can assume that other Microsoft containers will follow soon. Development of container support for Visual Basic 4.0 is presumed to be nearly complete. The transfer of the OLE Forms development group to the Microsoft Office group (see below) clearly suggests that Microsoft application developers have direct access to the OCX container specifications that are unavailable to Lotus.

- **The OLE Forms Feature of the Cairo OS is being developed by the Microsoft Office Applications Group**

OLE Forms are a counterpart to OLE controls and a cornerstone of the Cairo user interface architecture. We were recently informed by a Microsoft employee that responsibility for development of this operating system feature has been transferred to the Microsoft Office *applications* group. The implications of this are particularly disturbing:

- Developers of Microsoft office products have early access to information on this key operating system technology.
- Office developers have the opportunity to optimize OLE Forms to meet their own needs, at the expense of supporting competitive applications.
- An inappropriate and potentially permanent tie between Microsoft's application and operating system products is created.

- **Microsoft's "Access" application developed in direct consultation with OCX developers**

Microsoft's "Access" database product recently shipped with OCX container support. Public information on writing such a container is extremely sketchy even now, and was essentially unavailable at the time Access shipped. We were told by an OCX developer that Access developers consulted frequently and directly with the OCX development group to get the information needed to build a container. Microsoft has also told us that there is no such support structure in place for other ISVs even now that Microsoft's own products are available to customers. Although they are willing to discuss creation of such a support structure, and to provide support on a best-effort basis in the meantime, Access has already been given a significant advantage relative to competitive products like Lotus Approach. Furthermore, no commitments to any specific level of support have been made at this time.

- **Developers of key OS features transferring to and from job assignments in Microsoft applications groups**

Key developers of technologies relating to OLE 2.0 and OCX have transferred back and forth between Microsoft application and operating system groups over the past several years. Clearly, such employees are in a position to bring both specific technical information and product planning perspectives with them as they transfer. Competitors have no comparable access to the development process.

Given our earlier positive experiences with OLE 2.0, the situation described above is particularly disappointing and disturbing. Whether by design or inadvertently, Microsoft has inappropriately tied implementation and support of a key new operating system component directly to their tools and applications groups. Those groups therefore have a direct advantage when competing with Lotus, and a conflict of interest in giving us support.

I believe that we must ask Microsoft to:

- Ensure that responsibility for support and implementation of operating system features like OCX rests with the Operating Systems group at Microsoft. Specifically, conflicts of interest between Microsoft's applications (and tools) groups and their competitors must be avoided.
- Ensure that neither documentation nor sample code required to exploit operating system features carries a license more restrictive than that of the operating system APIs themselves. Microsoft should not try to avoid such responsibilities by claiming that particular Microsoft tools are required for access to OS services.
- Recommit to providing equivalent information and support for operating system features to application and tool developers inside and outside of Microsoft.
- Avoid inappropriate transfers of personnel between groups if such transfers would give an unfair competitive advantage to Microsoft products.
- Work specifically to redress any inequities which may have arisen in the particular case of OCX and related technologies.

We were visited recently by Mike Blaszcak, one of the lead OCX developers. Mike was helpful and attentive to our concerns, and his visit represented a small but significant positive step in providing access to OCX expertise for Lotus developers. Nonetheless, the concerns listed above remain unresolved at this time. Our earlier experiences with OLE suggest that Microsoft and Lotus can have a productive and mutually beneficial relationship leading to the effective use of their operating system technologies in our

products. I hope that we can work with Microsoft to provide us with access to the information required to exploit OLE controls, OLE Forms, and other Microsoft operating system technologies in our products

Noah

CONFIDENTIAL IBM 0410075028
