PLAINTIFF'S EXHIBIT
2002
Comes v. Microsoft

Debra Vogt

From: Steven Sinofsky
To: Bill Gates
Subject: RE: Marginal cost for Mac
Date: Friday, January 28, 1994 7:23PM

It is very hard to hide some of the Mac specific features behind Windows APIs. This is how the Mac differentiates itself and how we improve Windows as well. Publish and Subscribe are really very different than OLE in the protocol of how they are implemented. The visualizations are different, and the interface is different. It is hard to think how a developer can write to one API and have it behave two completely different ways in the same application.

We also have the problem that Mac OLE is not the same API as Windows OLE. The assumption was that Mac OLE had to work with existing Mac applications, not WLM applications. Right or wrong that is what the situation is. WLM will be doing a really good job getting OLE to work with the Win32 APIs (except for the DLL problem they are working on), and that should be how we get OLE on our Mac applications in the next releases.

Other features like QuickDraw/GX printing also require more APIs and different protocols. I don't know how to maintain both the Windows GDI model and the new GX model (very different) in the same API at the same time.

OpenDoc will be another big issue. Again, it isn't just a different API on the same functionality. It has things in it that OLE does not (non-rectangular regions, Apple file format translation manager, AOCE envelopes, etc.). Since OLE doesn't express these concepts for Windows it is hard to see how one would have the existing OLE APIs implement this functionality.

For all of this we should be analyzing the Mac system as it evolves and improving and innovating what makes sense. Whenever we do this, we should make sure it gets expressed in both the Mac and Windows through WLM. This sort of makes our Windows APIs on the Mac the union of the features and gives more features to windows. This is very hard and should only be a goal, not a requirement.

I'm making a bigger deal out of the OLE stuff than it warrents. Still the majority of the effort goes to plain old Mac configuration issues (printing support, smaller screeens, multiple monitors (boy Windows should have that feature), file and path name munging (FindFile is a lot of work), and config management on various incarnations of the system).

The whole linking issue on the Mac is still being looked at for improvements. The problems arise from the fact that there is a need to keep debug information in Windows formats and Mac formats (for the low level Macsbug debugger), and resources are totally different (the file system as two forks, one for resources and one for data) which can't really be represented on DOS as one file (they use two files). The current system also has the problem where you essentially need two copies of your executable (because a translation is being done), which is very bad on disk space and SLOW because these are really big files (4MB+debug information of another 4MB) that get copied around. The situation will improve, but I'm not sure how much. Wings would love to figure out how to use their ilink and will be trying. This is not bcause of a lack of trying. They want to finish the first release of the product soon too.

I don't know how we will be able to sell Wings to anyone but people we don't want to sell it to. It is simply not the solution that corporate america wants. My suggestion is that we avoid corporations, since they want a "compile and go" solution (like the XVT library or what Bedrock claimed to be). It is hard for them to accept a product that costs a lot and gives them an 80% solution, especially when things like custom controls, and DLLs will be hard to port from Windows. The people to go after are the current crop of MacApp developers--the ones rogerh knows really well. They have been alienated by Apple (Bedrock mess) and would like to be making money on Windows. These people like MFC a lot (I gave a talk at their conference last year). They won't pay as much as corporations, but can live with an 80% solution. They will use MFC all the way and not complain about it (corporations don't want to "train" people in C++). These are the Mac zealots of today--they see their friends making money on Windows, but don't know how to get there without giving up their Mac market, that is why they bet on bedrock.

You still get to use the Mac toolbox if you need to in WLM. There are APIs that convert a Windows HWND to a Mac Window, DCs, bitmaps, etc can all be converted for the most part. This is another strength of WLM. Just like MFC lets you call Windows directly without messing up the "state" of the class library, WLM lets you call the toolbox without having to jump through hoops. This is how you can implement Mac specific things easily.

With Wings, I think, optimistically, there is a real chance to undermine the Mac. Features that are Mac-specific could lose a lot of luster if they don't come from Wings and Wings is the way all the smaller commercial mac programs get written. People will just skip doing things like balloon help and Apple drag and drop if it means too much work for too small a profit margin. I really believe that the number of religious Mac people is decreasing as they see their volume move to Windows. Unit sales for all the famous Mac apps are bigger or close on Windows (Pagemaker, Adobe, even Claris I believe). We've gotten a lot of bad press for our apps, but that doesn't stop sales. Some Windows idioms make our apps better mac apps (keyboard access to menus, for example).

Corporates will always want the "full insulation" from the native API. If portability is not a primary concern, then they will just go ahead and use goofy things like XVT for compiled applications, or some of the cross platform interpreted pseudo-windowing systems. We just can't deliver all of Windows on the Mac and we can't deliver all of the Mac seamlessly. THis is what makes WLM amazingly useful, but also makes it impossible to sell. W emade this tradeoff because we wanted to use the Windows APIs exactly--not a Mac interpretation of them. We will have great tools and for developers that stick to straight forward MFC (as a single EXE) they will have great luck getting up and running in little time (the development environment IDE will make it easy to switch a project to the Mac).

We need to have PowerPC soon as well. We need to stop this ridiculous marketing about the CPU--like it is really going to revolutionize the Mac. It will still be system 7.x under the hood. The chip can only be so much faster--they didn't change the laws of physics.

From: Bill Gates
To: Steven Sinofsky
Subject: RE: Marginal cost for Mac
Date: Friday, January 28, 1994 4:13PM

This is a valuable reply. My view is that we need to reinforce the resources that try and HIDE these problems rather than get groups to solve them all. For example publish and subscrube can be viewed as specific uses of OLE so that once you have OLE they are almost no work to build. I dont understand about no incremental linking - why is 68k different from MS format - use OUR EXE format to build and then convert the whole thing in one pass to Mac format - its not an instruction set specific thing (perhaps the semantics of the .EXEs dont allow this???).

I am still trying to figure out how we can get lots of ISVs to use Wings without actually strengthening MAC again Windows by making it a nice Windows clone and letting people like LOTUS easily attack us.