PLAINTIFF'S EXHIBIT
462
Comes v. Microsoft

Cc: bradsi genfac janfac
Date: Mon Nov 26 14:48:39 1990

The Facilities Department can be contacted via e-mail on the "genfac" alais. The "facman" alais is no longer in use.

In regards to your request concerning the Bldg 3 2nd floor mens room, I am having a plumber out to correct a clog in the urinal. It looks like it will need to be pulled off the wall to repair correctly. For the urinal to be clogged this bad, something has physically been put down the urinal path. Please be patient for one more day while repairs are effected.

"janfac" will ensure that the area is cleaned up in a sanitary manner.

Thank you for your patience!

127
From philba Mon Nov 26 14:53:50 1990
To: bens bradsi
Cc: cameronm davidw paulma richab
Subject: Re: to share or not to share, that is the question
Date: Mon Nov 26 15:50:19 1990

You bring up some valid points however there is one other point that needs to be understood: ISVs that have access to our code often take advantage of internal data-structures, bugs or other quirks of the implementation. This prevents us from changing the code and adds a serious burden of compatibility beyond the API.

I favor simply doing a better job of documenting.

| >From bens Mon Nov 26 12:22:46 1990
| To: bradsi
| Cc: cameronm davidw paulma philba richab
| Subject: to share or not to share, that is the question
| Date: Mon Nov 26 12:19:34 1990
|
| Issue:
| Should MS Apps (and other ISVs) have access to Windows source code?
|
| Complications:
| 1) If MS Apps have access to the Windows sources, then all ISVs should
| have access, else we are subject to restraint-of-trade complaints
| (to say nothing of the morality of the situation).
| 2) If outside ISVs have access to our sources, then we make it much easier
| for another company to come along and clone Windows.
|
| I talked to some apps guys on a recent recruiting trip, and they made the
| following very good point:

EXH. 39 DATE 11/01/01
WITNESS Abel
SUSAN ZIELIE

Windows is poorly documented

The state machine that is USER.EXE is barely documented in the SDK. Any ISV that wants to write a great Windows app ends up looking at the source code (like our Apps group), unassembling the DLLs, or writing experiment code to divine the actual behavior of the system.

Arguments against giving ISVs access to Windows source code usually boil down to:

> An ISV will use some undocumented feature of Windows, or directly access internal data, in such a way that future versions of Windows will be forced to support this bad behavior, restricting MS ability to innovate in Windows.

I claim that letting an ISV looking at the source code is the *best* way to avoid this problem:

1) An ISV that unassembles Windows to figure out its behavior is effectively looking at source code, but without the benefits of source comments. This approach is more work for the ISV, and gives MS no opportunity to guide the ISV. With source code, there are generally comments discussing rationale for the behavior of the system.

2) An ISV that writes test apps to divine Windows behavior is really on thin ice. Either the ISV spends a great deal of effort writing test code to be certain Windows is fully understood, or the ISV may end up making assumptions which are not correct. Since the former approach is a great deal of work, and the ISV is never sure when to stop ("do I really understand how this works now?"), most ISVs will end up in the latter situation. These are the most dangerous apps, since they are most dependant upon the exact behavior of a specific release of Windows.

The key problem is that our documentation does not provide sufficient depth of coverage. The key question is:

> How much would it cost to provide sufficient documentation, and is that any different, really, from providing source code?

The key difficulty in writing really great documentation is anticipating all the questions an ISV might have about the behavior of the system. Presumably, there is a level of documentation which is great enough that an ISV would have to perform only a small amount of experimentation.

Solutions:

A. Status Quo
   + Simple
   - MS Apps have (unfair) advantage over other ISVs.

B. Make Windows sources available for a fee, with a restrictive licensing

| agreement (only available to N trusted employees, must be kept on
| a secure server, no derivative of these sources may be shipped, etc.)
| + All ISVs are equal
| - Cloning risk
|
| C. Disallow all ISVs, including MS Apps, from looking at Windows sources
| (in practice, this would be a very hard thing to do, and certainly
| runs counter to the spirit of MS).
| + All ISVs are equal
| - Difficult to make happen at MS
| - Reduces information flow on Windows to ISVs
|
| D. Write great documentation.
| + All ISVs are equal
| + Avoids cloning risk
|
| Conclusion
| Given that we want to make Windows programming as attractive as possible,
| I vote for (B) making the sources available. This gives us a little
| extra incentive to keep enhancing Windows, so that it does not become
| a stationary target for cloners, but otherwise benefits the Windows
| ISV community.
|
| -- bens
|

128
From cliffw Mon Nov 26 14:57:51 1990
To: bradsi
Subject: XL Bugs
Date: Mon Nov 26 14:55:23 1990

The PREVIEW bugs have been forwarded to the appropriate people.

The Style Combo box display problem is probably a WIN 3 installation bug.
I think this is the problem because I just checked a the beta version
on a 386/25 in the configuration lab, and there were no problems.

If you re-install Win 3 (or I will do it if you wish) the problem should
go away. If it doesn't go away, please contact me.

129
From korys Mon Nov 26 15:03:58 1990
To: dwgroup
Subject: Printers in 3/2165
Date: Mon Nov 26 15:01:16 1990

\\prntsrv\aplw - Apple LaserWriter plus
\\prntsrv\ljiii - HP laserjet III