

to function (such as hardware interrupts and faults)....

48

From ralph Tue Oct 10 08:58:10 1989
To: aaron jaywant marcw ralph richp
Subject: Re: lotus and win 3
Date: Tue Oct 10 08:58:08 1989

>From russw Mon Oct 9 08:29:28 1989
To: ralph steveh
Subject: Re: lotus and win 3
Cc: billing camerom davidw jodyw marcw paulma petern philba richab russw russw
Date: Mon Oct 9 08:26:18 1989

ralph, your points are great--but for I think we have a major problem on our hands:
a. Lotus 1-2-3 3.0 may be a stupid architecture in terms of what they have done at ring 0--but by eliminating the option of backpatching the installed base of 1-2-3 3.0 Lotus has a big problem on their hands.

I had not heard such a strong message from you guys about it not being possible to field-update existing re. 3 owners. Without this available Lotus would have to really do a more re-release to existing purchasers--not just roll-in the fixes to new purchasers. this is very different than just revising the rational run-time.

It is such a big problem, to my mind it may no longer be possible to get a workable strategy with lotus for co-existing with win386--and that is a big problem for us--unless you want to go to war with lotus over the issue--or find a way to run their existing product.

b. It does sound to me like Lotus is in a box in terms of VCPI or any other method for running with "appropriate" evolution of vcpi for virtualization--since they won't work. your points are great here. It may be that we will have to go to war with lotus publically to take the heat off of windows386--but I sure don't want to do this at the moment given the volatility of lotus. Also your points about even worse behaved vcpi apps is great.

Billg would like us to write a letter or something to the vcpi club (or rational) spelling out what we would like them to tell their users to stop doing (like the ring 0 stuff) so that that we can have these products get upgraded to be "cleaner" so that they can write vxd's and run. I will work with you on that this week.

c. OS/2 does not need to support Dos Extenders. Any app that is dos extended (except for some vertical market stuff) also provides an os/2 version of the app. For example, Lotus 1-2-3 3 ships with the os/2 version in the same box. 32-bit os/2 will be an answer for Borland paradox.

d. The bottom line, is that win386 may be a big loser unless we can figure out how to run Lotus 1-2-3 under it. We do have the current MARCV

strategy of telling users to run them in 286 mode. If the market buys that--and lotus relaxes a little--then maybe that is the answer to the problem. If the market insists on win386 running rel. 3--then either they have a huge upgrade problem on their hands or we do. someone loses.

49

From ralph Tue Oct 10 08:58:31 1989
To: aaron jaywant marcw ralph richp
Subject: VCPI support under Windows/386
Date: Tue Oct 10 08:58:28 1989

>From ralph Mon Sep 25 14:31:25 1989
To: billg
Cc: ralph
Subject: VCPI support under Windows/386
Date: Mon Sep 25 14:31:22 1989

Russw asked me to send you information on running Lotus 1-2-3 ver 3.0 under Windows/386.

As far as us writing a virtual device to support Lotus without any modifications on their part, I would guess that it would be very difficult. DOS extenders attempt to directly modify the GDT, LID, IDT, and page tables. This is exactly why VCPI was defined -- It lets each DOS extender take control of the machine so it can set up it's own protected mode state. If we wrote a virtual device to run Lotus (not support VCPI, just run 1-2-3), we could not allow the rational extender to modify the state of the GDT, IDT, or page tables. Therefore, we would have to do some pretty fancy virtualization such as creating a "virtual IDT" and trapping on accesses to specific selectors. The work required to do this would take several man months and I could not guarantee it would ever work properly.

As far as VCPI goes, it is a horrible design when running multiple virtual machines. It forces the environment to give up all protection and transfer control of the machine to another operating environment. Below is a document I wrote several months ago that details what is wrong with VCPI.

Please send me mail or call if you have any further questions.

VCPI = DESQVIEW
Why We Shouldn't Follow Quarterdeck Off the Cliff

CONFIDENTIAL

First, let me point out that I like the idea of being compatible with existing DOS extended applications. I know that users will scream and yell if we are not. However, the current Windows/386 architecture is fundamentally different from Quarterdeck's. Desqview will work on an 8086, 80286, and 80386. On a 386 it will work better if you run OEMM, and the quarterdeck LIMULATOR. OEMM is the only protected mode portion of Desqview. It provides special hooks so that old applications can be displayed in a window and, of course,

D:\TMP\DH003597

Tue Oct 10 10:37:36 1989

Plaintiff's Exhibit

7517

Comes V. Microsoft

X113902

7
provides EHM. The rest of the virtualization is all done in virtual 8086 mode using the standard Desqview kernel. This includes the PIC, COM, and Keyboard virtualization as well as scheduling and network support. Windows/386 has NO code that runs in virtual 8086 mode.

VCP1 is an interface that allows two autonomous protected mode control programs to operate independently of each other. As the VCP1 spec states: "Each control program maintains its own set of system tables (GDT, LDT, IDT, page tables, etc.)" This is all well and good for Desqview since it doesn't care that it's LHMulator is inactive when a DOS extended application is executing. However Desqview can not run a "Desqview oblivious" DOS extended application in a window. In other words, Desqview would be reduced to 8086 style virtualization which they do better than any other product I have seen.

An example of a specific technical problem Windows/386 has with VCP1 involves hardware interrupts. Since Desqview has no protected mode hardware interrupt code (except to reflect the interrupt to V86 mode), a VCP1 DOS extender can handle hardware interrupts correctly by simply reflecting them into the virtual 8086 code. Windows/386 on the other hand fully virtualizes the 8259A and must intercept all I/O to ports 20h, 21h, A0h, and A1h. Since VCP1 does not provide a mechanism for IOPM sharing, the Windows/386 architecture will not work with it. Similar problems are involved with the display, keyboard, COM ports, LPT ports, mouse, and networks. The Virtual Display Device would require major reworking if it was not able to trap the display I/O ports when an application was running.

VCP1 is nice method of allowing DOS extenders to cooperate with LHMulators. However, Windows/386 is an operating system. It is very difficult to imagine the current Windows/386 architecture ever working with VCP1 as it is currently defined. If we need to change the VCP1 specification then the main goal -- working with current Rational System DOS extended applications -- is lost. If we change the Windows/386 architecture, it will require an almost complete rewrite of all devices and large portions of the Virtual Machine Manager (Kernel). Also, one of the nicest potential features of Windows/386 would be lost. The current architecture will allow devices such as network transports to be moved into protected mode. VCP1 will not allow interrupting devices to reside in protected mode.

The ability to define our own API for our DOS extender will give us more control over multi-tasking applications since we can virtualize I/O under our current extender specification. In other words, we will be able to run protected mode applications in a window while Desqview can not. From a purely technical standpoint I would say restricting apps to our private API is the best way to go. From a marketing standpoint, compatibility with the National extender may be essential. Technically, VCP1 support is out of the question unless we wanted to start over with a Mondrian-like architecture and run Windows using a 386 DOS extender.

MARCU

8
Think of it this way: How hard would it be to "shut down" OS/2 for a little while and then start it back up? We have the same problem. Windows/386 uses the protection features of the 80386 to run virtual machines. When the protection goes away, so does Windows/386.

Desqview has reached the end of it's rope. Since it lives in real mode it must use EMS, XMS, and other sludge methods to get at extended memory. Windows/386 lives there very happily. Our architecture is fundamentally easier to extend than theirs. We have installable devices -- they don't. VCP1 does not allow the control program to take any advantage of the protection features of the 386.

CONFIDENTIAL

D:\VMP\DH003597.

Tue Oct 10 10:37:36 1989

X113903