I wrote the original blue screen of death, sort of

devblogs.microsoft.com/oldnewthing/20140910-00

September 10, 2014

Raymond Chen

We pick up the story with Windows 95. As I noted, the blue `Ctrl`+`Alt`+`Del` dialog was introduced in Windows 3.1, and in Windows 95; it was already gone. In Windows 95, hitting `Ctrl`+`Alt`+`Del` called up a dialog box that looked something like this:

Close Program ×

Explorer
Contoso Deluxe Composer [not responding]
Fabrikam Chart 2.0
LitWare Chess Challenger
Systray
WARNING: Pressing CTRL+ALT+DEL again will restart your computer.
You will lose unsaved information in all programs that are running.

End Task
Shut Down
Cancel

(We learned about Systray some time ago.)

Whereas Windows 3.1 responded to fatal errors by crashing out to a black screen, Windows 95 switched to showing severe errors in blue. And I'm the one who wrote it. Or at least modified it last.

I was responsible for the code that displayed blue screen messages: Asking the kernel-mode video driver to switch into text mode, filling the screen with a blue background, drawing white text, waiting for the user to press a key, restoring the screen to its original contents, and reporting the user's response back to the component that asked to display the message.[1]

When a device driver crashed, Windows 95 tried its best to limp along despite a catastrophic failure in a kernel-mode component. It wasn't a blue screen of death so much as a blue screen of lameness. For those fortunate never to have seen one, it looked like this:

```
 Windows


An exception 0D has occurred at 0028:80014812. This was called from 0028:80014C34. It
may be possible to continue normally.
* Press any key to attempt to continue.
* Press CTRL+ALT+DEL to restart your computer. You will
  lose any unsaved information in all applications.
```

Note the optimistic message “It may be possible to continue normally.” Everybody forgets that after Windows 95 showed you a blue screen error, *it tried its best to ignore the error and keep running anyway*. I mean, sure your scanner driver crashed, so scanning doesn’t work any more, but the rest of the system seems to be okay.

(Imagine if you did that today. “Press any key to ignore this kernel panic.”)

Technically, what happened was that the virtual machine manager abandoned the *event* currently in progress and returned to the event dispatcher. It’s the kernel-mode equivalent to swallowing exceptions in window procedures and returning to the message loop. If there was no event in progress, then the current application was terminated.

Sometimes the problem was global, and abandoning the current event or terminating the application did nothing to solve the problem; all that happened was that the next event or application to run encountered the same problem, and you got another blue screen message a few milliseconds later. After about a half dozen of these messages, you most likely gave up hope and hit `Ctrl`+`Alt`+`Del`.

Now, that’s what the message looked like originally, but that message had a problem: Since the addresses at which device drivers were loaded into the kernel were not predictable, having the raw address didn’t really tell you much. If you were someone who was told, “This senior executive got this crash message, can you figure out what happened?”, all you had to work with was a bunch of mostly useless numbers.

That someone might have been me.

To help with this problem, I tweaked the message to include the name of the driver, the section number, and the offset within the section.

```
 Windows
An exception 0D has occurred at 0028:80014812 in VxD CONTOSO(03) + 00000152. This
was called from 0028:80014C34 in VxD CONTOSO(03) + 00000574. It may be possible to
continue normally.
* Press any key to attempt to continue.
* Press CTRL+ALT+DEL to restart your computer. You will
  lose any unsaved information in all applications.
```

Now you had the name of the driver that crashed, which might give you a clue of where the problem is, even if you knew nothing else. And somebody with access to a MAP file for the driver could now look up the address and identify which line crashed. Not great, but better than nothing, and before I made this change, nothing is what you had.

So you could say that I wrote the Windows 95 blue screen of death lameness to make my own life easier.

Bonus chatter: Later, someone (I forget whether it was me, so let's say it was one of my colleagues) added some more code to inspect the crashing address, and if it was inside the kernel heap manager, the message changed slightly:

```
 Windows
A 32-bit device driver has corrupted critical system memory, resulting in an exception 0D
at 0028:80001812 in VxD VMM(01) + 00001812. This was called from 0028:80014C34 in
VxD CONTOSO(03) + 00000575.
* Press any key to attempt to continue.
* Press CTRL+ALT+DEL to restart your computer. You will
  lose any unsaved information in all applications.
```

In this case, the sentence "It may be possible to continue normally" disappeared. Because we knew that, odds are, it won't be.

Bonus chatter: Nice job, Slashdot. You considered posting a correction, but your correction was also wrong. At least you realized your mistake.

[1] Since this code ran in the kernel, it didn't have access to keyboard layout information. It doesn't know that if you are using the Chinese-Bopomofo keyboard layout, then the way to type "OK" is to press `C`, followed by `L`, followed by `3`. Not that it would help, because there is no IME in the kernel anyway. As much as possible, the responses were mapped to language-independent keys like `Enter` and `ESC`.

Raymond Chen

Follow