

TITLE: Confessions of a DDK Developer

The OS/2 Device Driver Development Kit falls short, but it's a start

STEVE MASTRIANNI

Photograph: STEVE MASTRIANNI

One cold February afternoon, after a long wait, I received a letter from IBM announcing the beta OS/2 2.0 Device Driver Development Kit. I eagerly signed and returned the form with a check for \$15, and less than one week later, I finally had the elusive OS/2 DDK. I'd been hearing rumors about it for several months. IBM kept promising to deliver it, but delays dragged the process out, and IBM had to repeatedly ask developers to have patience. Sources close to IBM say that the problem was not the availability of the driver code. Instead, no one was given the responsibility to gather the code and produce the DDK. Rumors of legal problems with Microsoft also circulated. At last, the problems were resolved.

#### A Dubious Beginning

The DDK came on one CD-ROM, and the Win-OS2 drivers came on two floppy disks. I was eager to find out what took up all the space, so I began to install it immediately on a machine that was running the December 2.1 beta code. The installation process asked for 50 MB of disk space. I had 52 MB free, so I figured I'd just make it.

I was wrong.

About 95 percent of the way through the installation, I got a general protection fault and that feared message "the system is stopped." Judging by the address of the fault, it appeared to be in a device driver, and my worst fears were realized when I found that the first several tracks of my disk were rewritten with data. The disk was no longer bootable. The FAT (file allocation table) was gone, as was all my data. Fortunately, this was a spare system that didn't contain any critical information. (I later found out that this was a known bug with the 2.1 beta code. The device-driver code, I later discovered, requires 54 MB of disk space.)

I reformatted the hard disk and reinstalled DOS and the 2.1 beta code. I reinstalled much of my previous software, but I left approximately 100 MB free. This time the installation went perfectly, duly placing a gob of driver source code on my disk. All the drivers were installed: printer, SCSI, display, virtual, and physical. The installation program lets you select only the drivers you want. That helps conserve disk space, but locating a particular driver is a tedious job.

#### What You Get

The DDK includes the previously unreleased Microsoft CL386, a 32-bit C compiler that was used internally to create the 32-bit code for OS/2 2.0. IBM also used CL386 to build all the VDDs (virtual device drivers) for OS/2 2.x. Also included is a copy of MASM (Microsoft Macro Assembler) 5.1, although some examples require MASM 6.0. I was annoyed by this. Why couldn't the examples all use the same tools?

A better question yet: Why does IBM develop OS/2 with different compilers and assemblers? This requires device-driver developers to install Microsoft C 5.1, Microsoft C 6.0, CL386, IBM's C Set/2, MASM 5.1, and MASM 6.0. In its defense, IBM says that it is working to correct this problem in a subsequent release of the DDK. Also, some simple tools are included with the DDK, such as a program that lets you quickly change a file's extended attributes, a hardware palette-display program, a display test tool, a printer test tool, TOUCH, SED, MAPSYM, LIB, and the resource compiler, RCPP.

As for drivers, the DDK includes dozens. In the category of video-display drivers, the DDK includes a 16-bit VGA driver, a 16-bit 8514/A driver, a 32-bit VGA driver, a 32-bit Super VGA driver, and a 32-bit device-independent driver. The virtual video counterparts are also included. Base video handlers

for VGA, 8514/A, CGA, XGA (Extended Graphics Array), and EGA top off the list of video drivers. The two floppy disks contain the source code for two seamless video drivers based on a Windows 3.0 and Windows 3.1 VGA driver.

In the printer category, IBM supplies only two drivers: a PostScript driver and a plotter driver, which appear to be OS/2 1.3 source code. This makes sense, because IBM still supports these drivers on OS/2 1.3. You must compile the PostScript driver with CL386. For disk drives, the DDK includes the source code for the floppy driver, ASPI (advanced SCSI programming interface) driver, IDE driver, PS/2 SCSI driver, PS/2 floppy driver, and PS/2 DASD (Direct Access Storage Device) driver. For CD-ROM drives, the DDK includes the source code for the Hitachi, Toshiba, NEC, and Sony SCSI CD-ROM drives, as well as CD-ROM Device Manager. Drivers for the Mouse Systems mouse, PC Mouse/Logitech mouse, and VisiOn mouse are also included, although a virtual mouse driver is missing.

#### Hunting Around

The first place I looked was the PDD (physical device driver) section, since these are my favorite OS/2 device drivers. I've long been an advocate of writing PDDs in C. I supply a C-callable library that allows a PDD written in C to call the register-based OS/2 device helpers, and I was curious to see what IBM offered.

IBM included a library with source code and make files, but absolutely no documentation. If you're a good MASM programmer, you can probably figure out the more-complex-than-necessary assembly language code and macros, and the way to call the functions by reading the assembly language include file--if you have the time.

Also, the library contained several undocumented device-helper function calls. The library needs to be fully documented, and the documentation file placed on the CD-ROM for reference. If the function calls are unsupported, IBM should make note of these so that developers won't use them in production code. But they should be documented nonetheless. I was pleased to see much of the driver code written in C. Since most driver operations are quick, it makes almost no sense to write them in assembly language. Drivers written in C can be written in half the time it takes to write one using MASM, and they're easier to debug and support.

I didn't find the PDD reference, which is the PDD writer's bible. Nor did I find the presentation driver and VDD references. What's confusing about these omissions is that IBM has already released these in the Professional Developer's Kit. Not all driver writers, however, have need for the PDK, so they're left with virtually no documentation. IBM says it intends to supply these documents on future releases of the DDK.

Trying to make sense of the DDK by wading through over 50 MB of code is extremely tedious. The DDK CD-ROM should include the necessary navigational information in INF format. I'd also like to see the Control Program Reference manual on the DDK CD-ROM, since most of the calls to the device driver are performed with these APIs. When testing my drivers, I always have to refer to Control Program Reference because I find it impossible to remember all the parameters and their ordering.

Also noticeably absent from the DDK is an example of a file-system driver. The information about OS/2's IFS (installable file system) is sketchy at best, and it can be obtained only with special permission. Although a small skeleton IFS is downloadable via CompuServe, IBM needs to do a better job in the IFS area by providing sample code and IFS documentation as a standard part of the DDK CD-ROM. There's just no excuse for not providing the code.

#### Other Improvements Needed

The DDK is a good start, but it's extremely difficult to use without good documentation. Because IBM has committed to regular releases of the DDK, it should take the time to make the product better. Many of the source code examples contain hundreds of lines of code with no documentation whatsoever, and some never even state what the function does. IBM must provide more documentation to let developers understand exactly what is on the CD and how

to find it. Nobody has the time to fumble around the CD looking for something that may or may not be there.

A debugger such as ASDT32 should be provided on the CD. The DDK is also lacking in examples of character drivers, such as a serial or parallel driver, a data acquisition driver, or a simple memory-mapped driver. These "other" drivers make up over half of the devices that will require OS/2 device drivers. The DDK should also include a tutorial for building device drivers with sample code. All these documents, including the device-driver references, should be supplied in several formats (e.g., INF, Read/2, ASCII, and PostScript).

IBM must do a better job of supporting device-driver writers, especially since this has traditionally been OS/2's Achilles' heel. Currently, IBM operates a small BBS where device-driver writers can ask questions and download sample code. However, the BBS is a single-user system and lacks the benefit of ongoing message threads, like you find on CompuServe or BIX. Periodically, the questions and answers are merged into a file that can be downloaded, but this is not as helpful as a continuing message thread on CompuServe. IBM has announced that CompuServe is the official public support forum for OS/2 2.x. Therefore, IBM should support device drivers in the device-driver forum there, where a much wider audience of developers can benefit from the message traffic.

#### Waking Up?

IBM may finally be getting the message. A special group within the company has been formed to promote OS/2 device drivers--both publicly and internally. The DDK has become a product, and it now has a product manager assigned to it. IBM plans quarterly releases of the DDK, and each release should get better and better. Future releases will include drivers and tools for the following areas: pen computing, multimedia, XGA, 8514/A, SCSI, mice, keyboards, IFS, serial and parallel ports, touchscreens, and PCMCIA, as well as a large selection of tools and on-line documentation.

IBM has scheduled a three-day OS/2 DDK conference this month. The OS/2 2.x device-driver developers from the Boca Raton labs will attend to give talks and meet with developers. Perhaps the release of the DDK and the conference signal a turning point for OS/2 device-driver developers. Stay tuned.

Steve Mastrianni is president of Personal Systems Software in Canton, Connecticut. He specializes in device drivers, operating systems, and real-time applications for OS/2 and Windows NT. You can reach him on BIX as "smastrianni," on the Internet at smastrianni@bix.com, or on CompuServe at 71501,1652.