

What's the difference between the COM and EXE extensions?

 devblogs.microsoft.com/oldnewthing/20080324-00

March 24, 2008



Raymond Chen

Commenter Koro asks [why you can rename a COM file to EXE without any apparent ill effects](#). (James MAstros [asked a similar question](#), though there are additional issues in James' question which I will take up at a later date.)

Initially, the only programs that existed were COM files. The format of a COM file is... um, none. There is no format. A COM file is just a memory image. This “format” was inherited from CP/M. To load a COM file, the program loader merely sucked the file into memory unchanged and then jumped to the first byte. No fixups, no checksum, nothing. Just load and go.

The COM file format had many problems, among which was that programs could not be bigger than about 64KB. To address these limitations, the EXE file format was introduced. The header of an EXE file begins with [the magic letters “MZ”](#) and continues with other information that the program loader uses to load the program into memory and prepare it for execution.

And there things lay, with COM files being “raw memory images” and EXE files being “structured”, and the distinction was rigidly maintained. If you renamed an EXE file to COM, the operating system would try to execute the header as if it were machine code (which didn't get you very far), and conversely if you renamed a COM file to EXE, the program loader would reject it because the magic MZ header was missing.

So when did the program loader change to ignore the extension entirely and just use the presence or absence of an MZ header to determine what type of program it is? Compatibility, of course.

Over time, programs like `FORMAT.COM`, `EDIT.COM`, and even `COMMAND.COM` grew larger than about 64KB. Under the original rules, that meant that the extension had to be changed to EXE, but doing so introduced a compatibility problem. After all, since the files had been COM files up until then, programs or batch files that wanted to, say, spawn a command

interpreter, would try to execute `COMMAND.COM` . If the command interpreter were renamed to `COMMAND.EXE` , these programs which hard-coded the program name would stop working since there was no `COMMAND.COM` any more.

Making the program loader more flexible meant that these “well-known programs” could retain their COM extension while no longer being constrained by the “It all must fit into 64KB” limitation of COM files.

But wait, what if a COM program just happened to begin with the letters MZ? Fortunately, that never happened, because the machine code for “MZ” disassembles as follows:

```
0100 4D          DEC    BP
0101 5A          POP    DX
```

The first instruction decrements a register whose initial value is undefined, and the second instruction underflows the stack. No sane program would begin with two undefined operations.

Raymond Chen

Follow

