

Why do text files end in Ctrl+Z?

 devblogs.microsoft.com/oldnewthing/20040316-00

March 16, 2004



Raymond Chen

Actually, text files don't need to end in Ctrl+Z, but the convention persists in certain circles. (Though, fortunately, those circles are awfully small nowadays.) This story requires us to go back to CP/M, the operating system that MS-DOS envisioned itself as a successor to. (Since the 8086 envisioned itself as the successor to the 8080, it was natural that the operating system for the 8086 would view itself as the successor to the primary operating system on the 8080.) In CP/M, files were stored in "sectors" of 128 bytes each. If your file was 64 byte long, it was stored in a full sector. The kicker was that the operating system tracked the size of the file as the number of sectors. So if your file was not an exact multiple of 128 bytes in size, you needed some way to specify where the "real" end-of-file was. That's where Ctrl+Z came in. By convention, the unused bytes at the end of the last sector were padded with Ctrl+Z characters. According to this convention, if you had a program that read from a file, it should stop when it reads a Ctrl+Z, since that meant that it was now reading the padding. To retain compatibility with CP/M, MS-DOS carried forward the Ctrl+Z convention. That way, when you transferred your files from your old CP/M machine to your new PC, they wouldn't have garbage at the end.

Ctrl+Z hasn't been needed for years; MS-DOS records file sizes in bytes rather than sectors. But the convention lingers in the "COPY" command, for example.

[Raymond Chen](#)

Follow

