

Taxes: Files larger than 4GB

 devblogs.microsoft.com/oldnewthing/20080108-00

January 8, 2008



Raymond Chen

Nowadays, a hard drive less than 20 gigabytes is laughably small, but it used to be that the capacity of a hard drive was measured in megabytes, not gigabytes. Today, video files and databases can run to multiple gigabytes in size, and your programs need be prepared for them. This means that you need to use 64-bit file offsets such as those used by the function `SetFilePointerEx` (or `SetFilePointer` if you're willing to fight with the somewhat roundabout way it deals with the high 32 bits of the offset). It also means that you need to pay attention to the `nFileSizeHigh` of the `WIN32_FIND_DATA` structure. For example, if your program rejects files smaller than a minimum size, and I give you a file that is exactly four gigabytes, and you check only the `nFileSizeLow`, then you will think that the file is too small even though it is actually enormously huge.

More indirectly, it means that you can't blindly map an entire file into memory. Many programs simplify file parsing by mapping the entire file into memory and then walking around the file using pointers. This breaks down on 32-bit machines once the file gets to be more than about a gigabyte and a half in size, since the odds of finding that much contiguous free address space is pretty low. You'll have to map it in pieces or use some other parsing method entirely.

Raymond Chen

Follow

