

Deleted but not yet forgotten

 devblogs.microsoft.com/oldnewthing/20040607-00

June 7, 2004



Raymond Chen

Earlier, I discussed the interactions of the various FILE_SHARE_* flags, then Larry Osterman took this as inspiration to give a history of the file sharing flags.

If a file is opened with delete-sharing enabled and you delete the file, the file doesn't actually go away until the open handles are closed. In the meantime, the file hangs around as a zombie. (Under Unix, a deleted file with open handles is removed from the directory and merely floats around in the happy sea of inodes with no name.)

Why doesn't the file go away?

Well, one reason is that device drivers can ask for the name of the file that corresponds to an open handle. If the directory entry were removed, then there would be no name to return! (What would you expect to be returned from GetModuleFileName if the module no longer exists? Should it be possible for GetModuleFileName to return ERROR_FILE_NOT_FOUND?)

Another reason is that if power were to be lost while a "forgotten but not lost" handle was still open, you would now have lost clusters on the disk.

And a final reason is that a "pending delete" file isn't actually gone for good. A driver can "undelete" the file by clearing the delete-on-close flag!

Raymond Chen

Follow

