

# Why doesn't the Open Files list in the Shared Folders snap-in show all my open files?

[devblogs.microsoft.com/oldnewthing/20110823-00](http://devblogs.microsoft.com/oldnewthing/20110823-00)

August 23, 2011



Raymond Chen

A customer wanted a way to determine which users were using specific files on their server. They fired up the Shared Folders MMC snap-in and went to the *Open Files* list. They found that the results were inconsistent. Some file types like `.exe` and `.pdf` did show up in the list when they were open, but other file types like `.txt` did not. The customer asked for an explanation of the inconsistency and for a list of which file types work and which ones don't. The customer is confusing two senses of the term *open file*. From the file system point of view, an open file is one that has an outstanding handle reference. This is different from the user interface concept of "There is an open window on my screen showing the contents of the file." The *Open Files* list shows files which are open in the file system sense, not in the user interface sense. Whether a file shows up in the *Open Files* list depends on the application that is used to open the file (in the user interface sense). Text files are typically opened by Notepad, and Notepad reads the entire contents of the file into memory and closes the file handle. Therefore, the file is open (in the file system sense) only when it is in the process of being loaded or saved. There is no comprehensive list of which types of files fall into which category because the behavior is not a function of the file type but rather a function of the application being used to view the file. (If you open a `.txt` file in Word, I believe it will keep the file system handle open until you close the document window.)

The customer seemed satisfied with the explanation. They ran some experiments and observed that *Hey, check it out, if I load a really big text file into Notepad, I can see it show up in the Open Files list momentarily*. They never did come back with any follow-up questions, so I don't know how they went about solving the original problem. (Maybe they used a SACL to audit who was opening the files.)

Raymond Chen

**Follow**

