

How can I write to a file only as long as nobody's looking?

 devblogs.microsoft.com/oldnewthing/20130829-00

August 29, 2013



Raymond Chen

A customer wanted to know how to detect that the user has opened Notepad to view a particular file. They had come up with method based on polling and sniffing the Notepad title bar, but they found that it consumed a lot of CPU. (They hadn't noticed yet that it doesn't localize, and that it can trigger false positives since Notepad shows only the file name and not the full path in the title bar.) Okay, let's step back and make sure we understand the problem, because this sounds like the sort of thing where the customer is looking for an answer rather than a solution. After all, why Notepad? What about some other text editor? What about Explorer's preview pane? The scenario here is that the program is generating some sort of output file, and they want the program to exit if somebody double-clicks the output file in an attempt to open it in Notepad. The customer wasn't specifically interested in Notepad, but since that was the most common case, that's all that they were really interested in. One option is to open the file in `FILE_SHARE_READ` mode. This allows other applications to open the file for reading even while you are writing. If, on the other hand, you expect the user to try to edit the output file and save the result, then they will either encounter a sharing violation (if you opened in deny-write mode) or overwrite the changes that the generator program had made in the meantime.

The customer said that they were fine with the program just exiting if somebody tried to look at the output file while it was being generated. In that case, they could use an oplock, specifically a Read-Write oplock. The Read-Write oplock grants you exclusive access to the file, but notifies you if somebody else tries to access the same file, so that you can close your handle and let the other application gain access.

Raymond Chen

Follow

