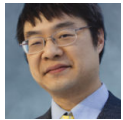


What happens if I have multiple asynchronous `ReadDirectoryChangesW` calls outstanding on the same directory handle?

 devblogs.microsoft.com/oldnewthing/20160713-00

July 13, 2016



Raymond Chen

A customer ran into an issue with the `ReadDirectoryChangesW` function. It was rather complicated, but one corner of the issue boiled down to the following: The customer had multiple asynchronous calls to `ReadDirectoryChangesW` outstanding. What if anything can be said about the results?

First, we'll give the answer while wearing kernel-colored glasses: The kernel completes the calls to `ReadDirectoryChangesW` in the order they were issued, so that the earliest call receives the first available batch of changes, and the next call receives the next available batch of changes, and so on.

Now let's take off our kernel-colored glasses and see what this means for the application.

Even though the kernel completes the calls in the order they were issued, the fact that the scheduler can preempt a thread at any time means that even though the two operations complete in sequence, the two threads that handle the completion are at the mercy of the scheduler, and they will race against each other, so the end result is unpredictable anyway.

[Raymond Chen](#)

Follow

