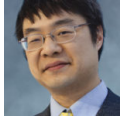


The case of the oplock deadlock poppycock

 devblogs.microsoft.com/oldnewthing/20181029-00

October 29, 2018



Raymond Chen

A customer reported that their application would sometimes hang trying to open a file. The call to `CreateFile` would simply hang, rather than failing with `ERROR_SHARING_VIOLATION` or some other error code.

One case where a `CreateFile` can hang is when there is an oplock on the file that needs to be broken.

Explorer takes an oplock on files when generating thumbnails. This allow Explorer to be notified when another application wants access to the file in a manner that conflicts with the the thumbnail extractor, so it can abandon the thumbnail operation and close the file, thereby permitting the other application's `CreateFile` call to succeed. If Explorer hadn't used an oplock, then an application that wants to access the file would get an immediate sharing violation.

The thread that owns the oplock is stuck here:

```
kernel32!CreateFileW+0x379
ole32!CFileStream::Init_OpenOrCreate+0x10d
ole32!CFileStream::InitWorker+0xb5
ole32!DfFromName+0x8f
ole32!DfOpenDocfile+0x1f2
ole32!DfOpenStorageEx+0x183
ole32!StgOpenStorageEx+0x9d
contoso!DllUnregisterServer+0xa9c0
SHELL32!CIconAndThumbnailOplockWrapper::Extract+0x14
SHELL32!CShellItem::_GetThumbnail+0x6f
... and so on ...
```

The shell took an oplock on the file and then called the thumbnail provider to extract the thumbnail. The thumbnail provider called back into `StgOpenStorageEx`, presumably to open the file in order to get the thumbnail. But that open call also hung.

The thumbnail extractor passed these flags to `StgOpenStorageEx`:

- `STGM_READWRITE`

- `STGM_ SHARE_ DENY_ WRITE`
- `STGM_ TRANSACTED`

The interesting flag here is `STGM_ READWRITE`. The oplock takes read access with deny-write, and the attempt to open the file for writing conflicts with the oplock. This causes the `CreateFile` to block waiting for the oplock to be released. But the oplock can't be released until the thumbnail extraction is complete. Deadlock.

The customer confirmed that changing `STGM_ READWRITE` to `STGM_ READ` fixes the problem. "But they say that they need write access."

We never did learn why the customer requires write access in order to extract a thumbnail from a file. I mean, does that mean that they cannot extract thumbnails from read-only files, or from CD-ROM drives?

We strongly encouraged them to find a way to extract thumbnails that doesn't require write access. But if that is impossible for some reason, they can work around it by indicating that their thumbnail extractor is not compatible with oplocks.

```
[HKEY_CLASSES_ROOT\CLSID\{xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}]  
NoOplock=REG_DWORD:1
```

~~For apps packaged with the Desktop Bridge, you can make [this declaration in your manifest](#):~~

```
<desktop2:DesktopPreviewHandler ... NoOplock="true"/>
```

Correction: The declaration above is for preview handlers, not thumbnail handlers.

Raymond Chen

Follow

