Thread affinity of user interface objects: Addendum

devblogs.microsoft.com/oldnewthing/20121109-00

November 9, 2012

Raymond Chen

Some time ago, I discussed the thread affinity of device contexts, noting that the `GetDC` and `ReleaseDC` functions must be called from the same thread. The same restriction also applies to printer DCs. Printer DCs must be created and destroyed on the same thread. The printing team imposed this additional rule in order to make it easier for printer driver vendors to author their drivers. (Printer driver developers have a habit of calling `CoInitialize` in their implementation of `CreateDC` and `CoUninitialize` in their implementation of `DeleteDC`.) Given that printer drivers are a third-party extension point, it's probably in your best interest to treat printer DCs as having hard thread affinity, since who knows what the printer driver is going to do if you try to access it from multiple threads. In other words, the thread that creates the printer DC should be the only thread which prints to it, and should also be the thread which destroys the printer DC when printing is complete.

That's not saying that all your printing has to be done from one thread. If you want, you can create multiple printer DCs, each on different threads. Just make sure to use each printer DC only on the thread that created it. Your printer driver will thank you.

Raymond Chen

Follow