The challenges in changing the way Explorer hosts shell extensions

devblogs.microsoft.com/oldnewthing/20111114-00

November 14, 2011

Raymond Chen

Various types of shell extensions such as thumbnail extractors are run in a separate process, and if one of those shell extensions crashes, it takes out the COM Surrogate rather than the main Explorer process. Anonymous wondered if this model could be extended to all types of shell extensions, perhaps not all at once, but gradually. The dangers of extending this model to existing shell extensions are compatibility (of course) and re-entrancy. The thumbnail extractor interface was lucky in that the only parameter an extractor received was an `IShellItem` representing the object for which the caller wishes to retrieve a thumbnail. There's no foothold into the user interface, which means that it can be moved to a place that has no user interface. Unfortunately, most shell extensions are not so lucky. Many of them receive some user interface object (usually a window handle) as a parameter. If those types of shell extensions have historically been hosted in-process, then the implementors of those shell extensions will do things like take that window handle and subclass it (possibly walking around the window hierarchy for a while until it finds a window it wants to subclass). Moving the shell extension into another process would break it, because you can't subclass windows in another process.

Even if you manage to find an interface that is given no foothold into the user interface, you still may not be able to move it to a host process due to the danger of re-entrancy. When you invoke a COM method call from a single-threaded apartment (and all UI work is done in single-threaded apartments), and the object that is the recipient of the call lives on another thread or even in another process, COM will send the request to that other thread and pump messages waiting for the reply. This means that a method call which previously never pumped messages now does, opening windows of re-entrancy, and the great thing about windows of re-entrancy is that you never hit them yourself, but your customers somehow manage to find them without any problem.

Raymond Chen

Follow