Psychic debugging: Why your IContextMenu::InvokeCommand never gets called

devblogs.microsoft.com/oldnewthing/20120727-00

July 27, 2012

Raymond Chen

A customer reported a problem with their shell context menu extension.

> I have implemented the `IContextMenu` shell extension, but when the user selects my custom menu item, my `IContextMenu::InvokeCommand` is never called. Can anyone please let me know what the problem could be and how to fix it?

Since there really isn't much information provided in this request, I was forced to invoke my psychic powers. Actually, given what you know about shell context menu hosting, you probably know the answer too.

> My psychic powers tell me that you gave your menu item the wrong ID, or you returned the wrong value from `IContextMenu::QueryContextMenu` .

If the menu IDs do not lie in the range you described by the return value from `IContextMenu::QueryContextMenu` , then when the user chooses the menu item, the item ID will not map to your shell extension. In our sample composite context menu, observe that `CCompositeContextMenu::ReduceOrdinal` relies on the component context menu handlers putting their menu IDs in the range `idCmdFirst` through `idCmdFirst - return_value - 1` . If the two don't line up, then `CCompositeContextMenu::ReduceOrdinal` won't realize that the menu item the user selected corresponds to you. We never did hear back from the customer, so the world may never know whether my psychic prediction was correct.

Bonus chatter: When possible, use a static verb registration instead of an `IContextMenu` handler. They are much simpler to implement while still providing a good amount of expressive power.

You can provide additional information in your registration to control things like the conditions under which your verb should be shown. You can even register cascading submenus statically.

Raymond Chen

Follow