Receiving a notification any time the selection changes in an Explorer window

devblogs.microsoft.com/oldnewthing/20140922-00

Raymond Chen

Today's Little Program merely prints a message whenever the user changes the selection on the desktop. I chose the desktop for expediency, since it saves me the trouble of coming up with a way for the user to specify which Explorer window they want to track. Also, all I do is print a message saying "Selection changed!"; actually getting the selection was covered earlier in both C++ and script.

Remember that Little Programs do little to no error checking.

```
#define STRICT
#include <windows.h>
#include <ole2.h>
#include <shlobj.h>
#include <shdispid.h>
#include <atlbase.h>
#include <stdio.h>
class CShellFolderViewEventsSink :
    public CDispInterfaceBase<DShellFolderViewEvents>
{
public:
 CShellFolderViewEventsSink() { }
 HRESULT SimpleInvoke(
    DISPID dispid, DISPPARAMS *pdispparams, VARIANT *pvarResult)
 {
  switch (dispid) {
  case DISPID_SELECTIONCHANGED:
   printf("Selection changed!\n");
   break;
  }
  return S_OK;
 }
};
int __cdecl wmain(int, wchar_t **)
{
 CCoInitialize init;
 CComPtr<IShellFolderViewDual> spFolderView;
 GetDesktopAutomationObject(IID_PPV_ARGS(&spFolderView));
 CComPtr<CShellFolderViewEventsSink> spSink;
 spSink.Attach(new CShellFolderViewEventsSink());
 spSink->Connect(spFolderView);
 MessageBox(NULL, TEXT("Click OK when bored."), TEXT("Title"), MB_OK);
 spSink->Disconnect();
 return 0;
}
```

Our `CShellFolderViewEventsSink` simply prints the message whenever it receives a `DISPID_SELECTIONCHANGED` event.

Sure, this program isn't useful on its own, but you can incorporate into a program that uses an Explorer Browser so that your application can do something based on the current selection. (For example, if your program is using an Explorer Browser to let the user select files for upload, you can display the total file size of the current selection.)

Raymond Chen

Follow