

Printing the name and position of the focused item on the desktop

 devblogs.microsoft.com/oldnewthing/20160530-00

May 30, 2016



Raymond Chen

Today's Little Program prints the name and position of the focused item on the desktop. Remember, Little Programs do little to no error checking.

```

#define UNICODE
#define _UNICODE
#include "stdafx.h"
#include <windows.h>
#include <shlobj.h>
#include <exdisp.h>
#include <shlwapi.h>
#include <atlbase.h>
#include <atlalloc.h>
#include <stdio.h>

int __cdecl wmain(int argc, wchar_t **argv)
{
    CCoInitialize init;
    CComPtr<IFolderView> spView;
    FindDesktopFolderView(IID_PPV_ARGS(&spView));
    CComPtr<IShellFolder> spFolder;
    spView->GetFolder(IID_PPV_ARGS(&spFolder));

    int iItem;
    if (FAILED(spView->GetFocusedItem(&iItem))) {
        wprintf(L"Sorry, no focused item.\n");
        return 0;
    }

    CComHeapPtr<ITEMID_CHILD> spidl;
    spView->Item(iItem, &spidl);

    STRRET str;
    spFolder->GetDisplayNameOf(spidl, SHGDN_NORMAL, &str);
    CComHeapPtr<wchar_t> spszName;
    StrRetToStr(&str, spidl, &spszName);

    wprintf(L"Focused item is %ls\n", static_cast<LPWSTR>(spszName));
    spszName.Free();

    spFolder->GetDisplayNameOf(spidl, SHGDN_FORPARSING, &str);
    StrRetToStr(&str, spidl, &spszName);
    wprintf(L"Parsing name is %ls\n", static_cast<LPWSTR>(spszName));

    POINT pt;
    spView->GetItemPosition(spidl, &pt);
    wprintf(L"Position is %d, %d\n", pt.x, pt.y);

    return 0;
}

```

We actually have most of the necessary pieces lying around already.

After initializing COM and getting the desktop folder view, we get the underlying `IShellFolder` because we're going to need it in order to interpret the pidsl that come out later.

We ask the view for the index of the focused item. If it can't cough one up, then we apologize and exit.

Otherwise, we use that index to get the corresponding pidl and then ask the folder to convert it into a normal name (which is the name shown under the icon) and a parsing name (which for files is the full path name).

Finally, we ask for the item position.

There you have it, a little program that identifies the current focused item on the desktop.

Raymond Chen

Follow

