# How can I tell that a directory is really a recycle bin?

devblogs.microsoft.com/oldnewthing/20080918-00

September 18, 2008

Raymond Chen

Here's a question inspired by an actual customer question:

> I need a function that, given a path, tells me whether it is a Recycle Bin folder. I tried using functions like `SHGetSpecialFolderPath` with `CSIDL_BITBUCKET`, but that doesn't work because the Recycle Bin is a virtual folder that is the union of the Recycle Bins of all drives.

The customer noted that they don't want to hard-code the words `RECYCLED` and `RECYCLER`, which is a good decision because the name of the directory depends on many things. I mentioned earlier that it depends on the file system. It also depends on whether the drive is accessed locally or remotely; network-based Recycle Bin folders follow yet another naming scheme. It may even depend on what operating system the user is running. No, hard-coding the name of the Recycle Bin folders is not a good idea.

The `SHDESCRIPTIONID` structure tells you a little more about a shell folder. In addition to the "description ID", it also gives you a `CLSID`, and it is the `CLSID` that is relevant here.

```
#include <windows.h>
#include <shlobj.h>
#include <tchar.h>
#include <stdio.h>


HRESULT
GetFolderDescriptionId(LPCWSTR pszPath, SHDESCRIPTIONID *pdid)
{
  HRESULT hr;
  LPITEMIDLIST pidl;
  if (SUCCEEDED(hr = SHParseDisplayName(pszPath, NULL,
                                        &pidl, 0, NULL))) {
    IShellFolder *psf;
    LPCITEMIDLIST pidlChild;
    if (SUCCEEDED(hr = SHBindToParent(pidl, IID_IShellFolder,
                                    (void**)&psf, &pidlChild))) {
      hr = SHGetDataFromIDList(psf, pidlChild,
                     SHGDFIL_DESCRIPTIONID, pdid, sizeof(*pdid));
      psf->Release();
    }
    CoTaskMemFree(pidl);
  }
  return hr;
}


int __cdecl wmain(int argc, WCHAR **argv)
{
  SHDESCRIPTIONID did;
  if (SUCCEEDED(GetFolderDescriptionId(argv[1], &did)) &&
      did.clsid == CLSID_RecycleBin) {
    printf(“is a recycle bin\n”);
  } else {
    printf(“is not a recycle bin\n”);
  }
  return 0;
}
```

The `GetFolderDescriptionId` function takes the path to a folder and converts it to an `ITEMIDLIST` just so it can call `SHGetDataFromIDList` to obtain the `SHDESCRIPTIONID`. All we care about in this case is whether the `CLSID` is the Recycle Bin or not.

```
C:\> checkrecycle C:\Windows
is not a recycle bin
C:\> checkrecycle C:\RECYCLER\S-1-5-21-2127521184-1604012920-1887927527-72713
is a recycle bin
```

Of course, now that I told you how to do it, I have to tell you that this is not how to do it. This is another example of a customer having a problem, solving half of it, and asking for help with the other half, unaware that they are approaching the problem from the wrong direction. Next time, we'll look at the customer's *real* problem.

Raymond Chen

**Follow**