How the shell converts an icon location into an icon

May 5, 2010

Raymond Chen

A customer had trouble getting an icon to display for a registered file type.

> In my resource file, I specify the icon like this:
>
> ```
> 101 ICON awesome.ico
> ```
>
> And when I register my file type, I set it like this:
>
> ```
> [HKEY_CLASSES_ROOT\.xyz\DefaultIcon]
> @="C:\path\to\awesome.exe,101"
> ```
>
> However, when I view an `.xyz` file, my awesome icon doesn't appear. On the other hand, if I change the `101` to a `0`, then it works. Why? Isn't the number in the resource file the resource ID?

Why yes, in fact, the number in the resource file is indeed the resource ID. But the number after the comma in the `DefaultIcon` *isn't*.

The format of shell icon locations (used most visibly by `DefaultIcon`, but also used in other places) is `«file»,«index»`, where the index is a zero-based index of the icon in your resources. In other words, an index of zero means "Give me the first icon in the file." The resource compiler sorts icons numerically by resource ID, so the first icon in the file is the icon with the numerically lowest resource ID. In the above example, apparently there are no icons with resource IDs less than 101; that would explain why asking for icon index zero results in the awesome icon.

The function behind all this icon hunting is `ExtractIcon`. The icon location string is split up at the comma into a path and an integer, and that path and integer are passed to the `ExtractIcon` function. Since the `ExtractIcon` function interprets the integer as the icon index, that's what the integer in your icon location string means.

In Windows 95, the `ExtractIcon` function was enhanced so that you could also specify an icon by its resource ID by passing its *negative* as the icon index. In other words, if you place a negative number after the comma, then its absolute value is interpreted as the resource ID.

We learned some time ago that there's a hole in this model: It doesn't work if you want a resource ID of 1, because <u>the value −1 has a special meaning to `ExtractIcon`</u>: Instead of extracting an icon, it tells you how many icons there are.

The workaround for this is actually straightforward, and you should be able to figure it out on your own based on information provided in this article. How do you specify that you want the icon whose resource ID is 1?

[Raymond Chen](#)

Follow