

The Old New Thing

How does Explorer deal with recent files that were renamed?

8 Oct 2012 7:00 AM

25

Roni wonders how Explorer manages to keep track of files that were moved or renamed. Specifically, "opening a shortcut to a renamed file actually updates the shortcut's destination and opens the renamed file. How is this done? Is it an NTFS feature?"

This feature has been around since Windows 95. If the target of a shortcut no longer exists, the shell tries to *resolve* the shortcut; i.e., find the object, wherever it ended up moving to. As I explained several months before the question was posted, the algorithm used by the shell varies depending on the operating system and the file system and your domain policies. Possibly also the phase of the moon, one can never be sure.

It's not that Explorer actually keeps track of the files as they move around, just in case you had a shortcut to them. Rather, the shortcut remembers enough information about the file so that if the file moves, Explorer can try looking for it.

The fact that shortcuts can resolve targets means that shortcuts are a handy tool for keeping track of files that might move around. If you want to keep track of a file, you can just create a shortcut to it (you don't even need to save it in a file), and when it comes time to find the file, you just resolve the shortcut.

Blog - Comment List MSDN TechNet

Comments

**Mm**

8 Oct 2012 7:06 AM

#

>> It's not that Explorer actually keeps track of the files as they move around, just in case you had a shortcut to them.

And luckily it does so.. otherwise the classic "rename old file, copy new file" would update the shortcut which is not what one wants.

**Lockwood**

8 Oct 2012 7:15 AM

#

I remember the torch animation.

"Trying to find new target..."

torch scanning

"The file MyGame.exe has been moved or deleted. Do you want the shortcut to now

point to RandomThingIShotWhileOnHoliday.jpg as the file closest matching? Yes/No"



Vilx-

8 Oct 2012 7:38 AM

#

Odd. I don't know if it's because I'm a programmer or what, but I've never seen this feature used in real life.



Muzer

8 Oct 2012 7:40 AM

#

I find it irritating when you click on a shortcut that doesn't exist and it offers to replace the shortcut with the found one (if it does find one), even when you don't have write access to the shortcut. This might have changed, of course, my only experience with that was XP.



alegr1

8 Oct 2012 8:11 AM

#

>otherwise the classic "rename old file, copy new file" would update the shortcut which is not what one wants.

In NTFS it probably will do that. NTFS shortcut keeps a file GUID. If you move a file and replace it with the new file with the same name, the GUID will belong to the old file.



alegr1

8 Oct 2012 8:12 AM

#

"xpclient: and even then, it doesn't place that file in the sort order"



John

8 Oct 2012 9:04 AM

#

@Vilx: I've never made use of this feature, either. Then again I don't create many personal shortcuts.



jader3rd

8 Oct 2012 9:24 AM

#

I didn't know it kept track of a rename. In the Vista timeframe I'd import some pictures, and then with a combination of Tab and F2 start renaming them. Many times I'd outpace Explorer (I could see it struggle with the renames), and then when I'd add some metadata tags (which was a much richer experience in Vista than in 7) Explorer would fail with errors saying that it couldn't find the file; sometimes the error was with the new name, sometimes with the old.

It's the reason why I wish that Explorer would make the changes asynchronously, and assume that it was going to work.



Giulia Q

8 Oct 2012 9:46 AM

#

@vilx: when this feature works, you won't even notice you used it :)



Giovanni

8 Oct 2012 10:49 AM

#

I wish Windows creates relative shortcuts so I can use them on a DVD-R.

[Oh, you mean SLDF_HAS_RELPATH, which is on by default? -Raymond]



AndyCadley

8 Oct 2012 11:53 AM

#

@alegr1: It doesn't matter if you're using NTFS, because the path to a file that the shortcut pointed to is still valid and thus there is no need to "resolve" the shortcut. Although the guid no doubt helps in all those cases where the target does go missing.



Skyborne

8 Oct 2012 12:30 PM

#

One time, the flashlight-finding-a-folder animation came up on our Windows 95 machine and my mom seemed surprised at it, saying something like, "it thinks it's dark in there?" I guess to her, it should be bright inside the computer because the screen is lit up.



Joshua

8 Oct 2012 1:07 PM

#

[Oh, you mean SLDF_HAS_RELPATH, which is on by default? -Raymond]

Try this. Burn such a DVD-R. Move it to a second DVD drive. Access the file via the shortcut. Grind-grind-grind oops no disk in first drive. Then it tries to resolve the second drive (which does work).

At least that's the way it worked last time I tried it.

[Oh, now I understand the question. You want to suppress the initial probe and go straight to the relative path. I don't think that's possible. -Raymond]

**Joshua**

8 Oct 2012 1:54 PM

#

[Oh, now I understand the question. You want to suppress the initial probe and go straight to the relative path. I don't think that's possible. -Raymond]

I'll bet a sufficiently deranged shortcut would work (primary path goes somewhere off of \\.\pipe or similar game), but good luck getting explorer to make one.

I solved the problem by shipping a tiny BAT file that could perform the relative lookup. So much easier.

**jas88**

9 Oct 2012 4:29 AM

#

"You want to suppress the initial probe and go straight to the relative path."

I think he was after a shortcut with a relative destination *instead of* an absolute one - which seems a reasonable thing to want, indeed arguably a more reasonable expectation for most purposes than absolute ones, since drive letters aren't constant and I'm sure the vast majority of links are intra-filesystem.

A batch/command file is a functional workaround, but far from ideal - surely there is a more elegant way?

[While I agree that it's an interesting feature, it's outside the scope of what shortcuts were originally designed for. -Raymond]

**JustSomeGuy**

9 Oct 2012 5:31 AM

#

I have a vague recollection that one version of OS/2 easily handled moving files around while still keeping shortcuts up to date. Or it may have been Workplace Shell.

Does anyone know about this?

**Giovanni**

9 Oct 2012 5:42 AM

#

"I think he was after a shortcut with a relative destination *instead of* an absolute one - which seems a reasonable thing to want, indeed arguably a more reasonable expectation for most purposes than absolute ones, since drive letters aren't constant and I'm sure the vast majority of links are intra-filesystem.

A batch/command file is a functional workaround, but far from ideal - surely there is a more elegant way?"

Si, si! When I put DVD-R in other PC then drive letter is changing. I was thinking relative shortcut is nice feature but Windows no support it. Batch file is not so nice way.

* There is little problem in blog. People with no account are showing "kurakuraninja" if mouse is over their name in post title.

**Joshua**

9 Oct 2012 8:08 AM

#

* There is little problem in blog. People with no account are showing "kurakuraninja" if mouse is over their name in post title.

I don't think that's a bug.

**Ian Boyd**

9 Oct 2012 10:20 AM

#

Not to be confused with [Distributed Link Tracking] (technet.microsoft.com/.../cc736811.aspx), which can find a shortcut target if you:

- rename it
- move it to a different folder
- move it to a different volume
- move it to a different computer
- rename the computer the target is on
- rename the share that the target is on

**Gabe**

9 Oct 2012 10:35 PM

#

Ian Boyd: Wouldn't Explorer just use the distributed link tracking service?

**David Walker**

10 Oct 2012 11:01 AM

#

I thought the Distributed Link Tracking Client was only useful on a domain-connected computer where there is a Distributed Link Tracking Server running in the same domain. Is that right or wrong?

**Ian Boyd**

10 Oct 2012 12:10 PM

#

@Gabe

> Wouldn't Explorer just use the distributed link tracking service?

i assume because Explorer was invented before NTFS, and has to run on file systems other than NTFS, they needed their own way to resolve shortcuts to the [best of their limited ability.](technet.microsoft.com/.../2006.09.windowsconfidential.aspx)

**Joshua**

10 Oct 2012 4:23 PM

#

[While I agree that it's an interesting feature, it's outside the scope of what shortcuts were originally designed for. -Raymond]

My first encounter with shortcuts encountered it. Had some program (probably game) on floppy and needed to start it with some custom options. Drive letter was A:. Guess what happened when I put the floppy into another machine that still had a 5.25" floppy as A:.

**Gabe**

10 Oct 2012 10:22 PM

#

Ian: The last paragraph of Raymond's article that you linked to actually implies that link tracking is indeed used by Explorer. And here's proof, from MSDN's documentation on the IShellLink::Resolve method:

"If distributed link tracking is not available or fails to find the link object, Resolve attempts to find it with search heuristics." (from msdn.microsoft.com/.../bb774952.aspx)

**Ian Boyd**

11 Oct 2012 7:34 AM

#

@Gabe: i don't believe Explorer is "using" Distributed Link Tracking. If the Distributed Link Tracking client service is running, and it manages to resolve the shortcut for you - then Explorer never *realizes* that the shortcut was broken. DLT intercepts requests for a shortcut and resolves it for you, at the file system level.

Only if DLT could not resolve the link (service not running, file moved too far from it's capabilities), then the shortcut is passed (broken as it is) to the calling process.

You could test this by trying to call `CreateFile` on a broken shortcut from your own application. If the shortcut is resolved, then DLT fixed it for you. Otherwise the file will not be found.

*[The file system doesn't know about shortcuts. That's purely a shell concept. The `IShellLink::Resolve` method first asks the DLT to find the target. If not found, then it uses its fallback algorithm. (I'm still not sure what mental model you're using where `CreateFile` on a *.lnk file magically opens a file different from the one you requested, and even if it did, that wouldn't fix the shortcut target.) - Raymond]*