When programs assume that the system will never change, episode 1, redux

devblogs.microsoft.com/oldnewthing/20200526-00

Raymond Chen

Back in underline episode 1, we learned about programs that grabbed undocumented resources from system files. Today's case is similar, but the culprit is not a program that shipped to customers. Rather, it's a customer who is using icons from the system in their own internal deployments.

The customer reported that at some point between Windows 7 and the current version of Windows, the icon they were using is no longer present, and the remaining icons were renumbered. As a result, the shortcuts they are deploying to their customers show up with the wrong icon.

That icon was removed in the normal course of product development. The scenarios that required the icon no longer existed, so the icon was removed.

I found it interesting that in the customer's impact statement, they never acknowledged that they were doing something wrong in the first place. They cheerfully stated that they reached in and grabbed icons that were lying around in undocumented places, and explained that any change to those undocumented icons would result in a break in their workflow.

I advised that the customer should stop using undocumented icons from undocumented locations. If they want to deploy shortcuts with specific icons, they should deploy the icons as well. Figure out what icon they want for the shortcut, create a DLL with that icon, and deploy that DLL along with the shortcuts.

The customer liaison passed the suggestion along, with a gentle reminder that what they are doing was never supported.

The request to put all the undocumented icons back the way they were in Windows 7 was rejected. For one thing, putting them back would in turn break customers who had improperly relied on the icon positions as they were in earlier versions of Windows 10. In

other words, given two sets of customers who both did the wrong improper thing, it's impossible to make both sets of customers happy, since the two sets are relying on contradictory requirements.

Since it is known how many customers are unhappy by the way the icons are set up in Windows 10 (namely: one), but it is not known how many customers would be made unhappy if the icons were moved back to the way they were in Windows 7. We could be trading one happy customer for a dozen unhappy customers.

We elected to leave things as they were. The one unhappy customer was given instructions on how to achieve their goal in a supported way, and they'll be happier about it anyway, because they can deploy their fix immediately, instead of having to wait for the update to be developed and deployed to all their systems.

Raymond Chen

Follow