

Why does SHFileOperation have internal error codes for DVD?

devblogs.microsoft.com/oldnewthing/20100322-00

March 22, 2010



Raymond Chen

erangi asks why the SHFileOperation function has internal error codes DE_DEST_IS_DVD and DE_SRC_IS_DVD if DVDs hadn't been invented at the time it was first written a long time ago.

As my colleague Christopher Davis explained, the `SHFileOperation` function originally came from the old File Manager of Windows 3.0, code which was written before Win32 error codes were invented. File Manager error codes and Win32 error codes had a common starting point (the MS-DOS error codes), but evolved under divergent paths. When no suitable MS-DOS error code existed for a situation that might arise during file copying, the File Manager folks invented an error code for it. Meanwhile, when no suitable MS-DOS error code existed for a situation that might arise in Win32, the Win32 folks invented an error code for it. That the two sets of error codes would come up with different meanings for the same numerical value is to be expected, since in both cases, the number was “available for use.”

Okay, so when the DVD error codes were added, why weren't they added to `winerror.h` to make them official instead of adding them to the internal error list?

Well, for one, these are internal error codes (which happen to be exposed to applications in an accidental way). Why bother making official error codes for things which were meant to be internal anyway?

“Dear kernel team, please add this error code to `winerror.h`.”

— *Okay, how should we document it?*

“Oh, it's not documented.”

— *Then why the heck do you need it in `winerror.h`?*

Second, there may be considerations that are not immediately obvious from the raw list of internal error codes. For example, I noticed that there is one error code called `ERRORONDEST` which is “or”d with other error codes. The case of `DE_ROOTDIR | ERRORONDEST` is

specifically called out, but it's highly likely that the more general case applies. For example, `DE_FILENAME_TOO_LONG | ERRORONDEST` probably means that a file name on the destination was too long. If the file copy engine were to switch entirely to Win32 error codes, all the uses of internal error codes would have to be broken up into two parts, one for the Win32 error code and another for the boolean that specifies whether the Win32 error code applies to the source or destination. This means changing all the functions which pass or return internal error codes, which can get particularly tricky if you were passing the error code as a parameter to `SendMessage` or `PostMessage` or some other function where you've already used up all the bits of expressiveness. (You'd have to put the information into a structure and then worry about keeping track of whose job it is to free that structure.)

Third, there's the principle of proportionate response: Sure, the person who wanted to add DVD handling to the file copy engine could have torn apart and rewritten the way error codes are handled inside the copy engine. But would that have been a proportionate response to a request to add DVD error handling to the copy engine? "Here you go, I added DVD error handling, and I completely redesigned the way errors are handled."

It's like asking someone to come fix a broken light switch and discovering that they rewired your house while they were there. Maybe they did a good job, or maybe they accidentally introduced a short circuit somewhere in a little-used closet. It's even more exciting when they don't even tell you that they rewired your house. You test the light switch, it works, and you thank them for a job well done. Then two months later, you visit that closet, turn on the light switch, and all the outlets on the second floor explode.

Now, the principle of proportionate response is not a law of the universe. It's just a principle, and sometimes principles need to be broken. For example, the principle of proportionate response also results in a frog being boiled alive. Sometimes you just have to get out of the pot. But apparently this was not one of those times.

Chris did point out that the new Vista copy engine returns `HRESULT`s rather than goofy internal undocumented error codes, so at least things are better now. The frog has been taken out of the pot.

Sidebar: It was only after I had written up this message that I realized that erangi had already asked this question at the bottom of the original blog entry. If I had known that, I wouldn't have bothered writing up this entry, because I don't like it when people ask the same question in multiple places, especially since my response is based purely on speculation, guesswork that you too could have performed.

Raymond Chen

Follow



