

What does the SEE_MASK_UNICODE flag in ShellExecuteEx actually do?

 devblogs.microsoft.com/oldnewthing/20140227-00

February 27, 2014



Raymond Chen

Somebody with a rude name wonders what the SEE_MASK_UNICODE flag does. It does nothing. The flag was introduced when porting the Windows 95 shell to Windows NT. It happened further back in history than I have permission to access the Windows source code history database, but I can guess how it got introduced. One of the things that the porting team had to do was make Unicode versions of all the ANSI functions that Windows 95 created. Sometimes this was done by creating separate A and W versions of a function. Sometimes this was done by having separate A and W versions of an interface. Sometimes by adding additional fields to the A version of a structure with a flag that says whether the ANSI or Unicode members should be used. My guess is that the porting team initially decided to make `ShellExecuteEx` use that third model, where the `SHELLEXECUTEINFO` structure had a `SHELLEXECUTEINFOEX` extension with Unicode strings, and the mask specified whether the caller preferred you to use the ANSI strings or the Unicode strings. Presumably they decided to change course and switch to having separate `SHELLEXECUTEINFOA` and `SHELLEXECUTE-INFOW` structures. But when they switched from one model to the other, they left that flag behind, probably with the intention of removing it once all existing callers had been updated to stop passing the flag, but they never managed to get around to it.

So the flag is just sitting in the header file even though nobody pays any attention to it.

Raymond Chen

Follow

