

Why did every Windows 3.0 DLL have an exported function called WEP?

 devblogs.microsoft.com/oldnewthing/20180919-00

September 19, 2018



Raymond Chen

One of the quirks of Windows 3.0 was this function called WEP, which was exported by every DLL. And yet if you looked at every DLL, you'd see that the function did nothing but return.

What's the deal with this function that everybody has, but which does nothing?

WEP stands for Windows Exit Procedure. Starting in Windows 3.0, Windows would call a DLL's WEP function with a single boolean parameter. It called the WEP function with `FALSE` immediately before unloading the DLL from memory, and it called the WEP function with `TRUE` immediately before shutting down. Of course, the DLL's WEP function was called only once per instance, because once you're unloaded, you're not going to be around to receive the shutdown notification.

Since the WEP function was called as part of the unload or shutdown process, you were extremely limited in what you could do. The function itself had to be in a non-moveable segment, and the import entry needed to be in the resident name table. You can't call any function that might result in a call to `LoadModule`. You can't call into another DLL at shutdown, because the other DLL might have already run its shutdown code.

In practice, everybody just returned without doing anything.

[Raymond Chen](#)

Follow

