Why do we have import libraries anyway?

devblogs.microsoft.com/oldnewthing/20091013-00

October 13, 2009

Raymond Chen

Last time we looked at the classical model for linking as groundwork for answering Adam's question why do we need import libraries? Why can't all the type information be encoded in the export table? At the time the model for DLLs was being developed, the classical model still was the primary means by which linking was performed. Men were men and women were women. Compilers generated object modules, and linkers *resolved symbols*, connecting the loose ends of the object modules, in order to produce an executable. The linker didn't care what language the object modules were written in; in fact it had no way of finding out, since by the time you had an object module, all that was left was a bunch of code with placeholders, and some metadata describing how those placeholders should be filled in. "If the answer to this is 'because the .dll doesn't contain all the information the linker needs' – the question becomes why don't .dlls contain all the information needed to link against them?'" Because the linker doesn't have that information when it generates the DLL file. It just has a bunch of code with placeholders. The type information was lost a long time ago. (And if you're writing code in assembly language, your concept of *type information* is radically different from that of a C or C++ programmer. What is the type of *the ECX register contains the size of the buffer on entry to the function and contains the number of unused bytes in the buffer on exit*?) Many years later, the OLE automation folks decided to encode type information in metadata in the form of a *type library*. And if that's not good enough, you can turn to the CLR, which records enough type information in its assemblies that you can, in principle, call the methods exported from it.

Mind you, this principle is pretty useless, because even with this type information, all you get are the function parameter types and return type. You still have no idea what that `bool` parameter means, for example. All that you have is a little more information to shoot yourself in the foot with.

Raymond Chen

Follow