

How do I write a program that can be run either as a console or a GUI application?

devblogs.microsoft.com/oldnewthing/20090101-00

January 1, 2009



Raymond Chen

You can't, but you can try to fake it.

Each PE application contains a field in its header that specifies which subsystem it was designed to run under. You can say `IMAGE_SUBSYSTEM_WINDOWS_GUI` to mark yourself as a Windows GUI application, or you can say `IMAGE_SUBSYSTEM_WINDOWS_CUI` to say that you are a console application. If you are GUI application, then the program will run without a console.

The subsystem determines how the kernel prepares the execution environment for the program. If the program is marked as running in the console subsystem, then the kernel will connect the program's console to the console of its parent, creating a new console if the parent doesn't have a console. (This is an incomplete description, but the details aren't relevant to the discussion.) On the other hand, if the program is marked as running as a GUI application, then the kernel will run the program without any console at all.

There are some people who want to write what I call an "opportunistic" console program. These are programs that will use the console of their parent if available, but do not want a console created for them if not. The kernel doesn't support this type of program, but that hasn't stopped some people from coming up with clever workarounds. Note that if such a program type were introduced, it would create problems with programs such as `cmd.exe` and Explorer which change their behavior depending on what subsystem a program belongs to. These programs would have to be modified to understand a new pseudo-subsystem called "both".

I've also seen requests for what I call a "dynamic" console program. These are programs that want to decide at run time whether they want a console or not. For example, a program might want to run with a console only if a special command line switch is passed. To do this the kernel would have to have psychic powers: It would somehow have to know whether to hook up a console to your program or not (which happens before the program begins executing)

based on something that happens in the future (when your program actually runs and parses its command line and decides whether it wants to run as a console or a GUI program). Again, people have come up with workarounds (see earlier link).

Raymond Chen

Follow

