How do I ShellExecute a file, but with a specific program instead of the default program?

December 20, 2017

Raymond Chen

The usual way to launch a file is to use `ShellExecute`, which opens the file in the default program registered to handle that type of file. But what if you want to override the default?

We saw some time ago that if you know the ProgId of the program you want to run, you can specify it in the `lpClass` of the `SHELLEXECUTEINFO` to force the file to be treated as if it were a specific kind of file.

Okay, but what if you aren't sure that the program you want to run has registered a ProgId at all? Or if you don't know what that ProgId is? For example, maybe you searched the hard drive for executable files and put them all in a list, and let the user pick one, and then you want to run that program to open the file.

In that case, you are in a bit of a pickle because you don't know how an arbitrary program expects its command line to be formatted in order to open a file. Fortunately, most programs which can open files will accept the file name on the command line with no other options, so you will have a high chance of success if you simply enclose the name of the file you want to open in quotation marks (in case it contains spaces), and then pass that as the command line. When calling `ShellExecute`, you pass `lpFile` equal to the program you want to run, and `lpParameters` equal to the file you want to open, enclosed in quotation marks if necessary.

Exercise: Does the path to the program need to be fully-qualified?

Exercise: Why do you have to quote the file you want to open, but not the program itself?

One reason you may want to use `ShellExecute` instead of just going straight to `CreateProcess` is if you need elevation. You can pass `lpOperation` equal to `runas` to get `ShellExecute` to do the work of prompting the user for elevation.

Raymond Chen

Follow

2/2