

Re: Semi-newbie question on Visual C++ vs. Visual C#

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2007-02/msg01013.html>

- *From:* Joseph M. Newcomer <newcomer@xxxxxxxxxxxx>
 - *Date:* Sun, 11 Feb 2007 13:03:42 -0500
-

I used to teach a Win16, then Win32 API programming course. I also taught an MFC course. In the Win16/Win32 courses, we could get the students to create an app that vaguely resembled a proper Windows application by late Wednesday afternoon. No menu prompts, no toolbars, no status bar, but hey, it almost worked right.

In the MFC course, the first lab is "build an MFC application". It takes 5 minutes to get to where we were after three days, and our first app is built before the morning break. ANd it has toolbar, status bar, menu prompts, etc.

The cost for the low-end version of VS should justify going for the real thing. I was buying the Microsoft C/C++/MFC system from my personal funds back when it was a great deal more expensive than it is today (allowing for inflation). Free software is often worth exactly what you pay for it.

joe

On Sun, 11 Feb 2007 15:36:45 GMT, MrAsm <mrasm@xxxxxxx> wrote:

On Sun, 11 Feb 2007 02:18:35 GMT, Steven O. <null@xxxxxxx> wrote:

I now plan to develop some simple scientific and mathematical applications, mostly for self-study purposes, but some of them may turn commercial at some point. The guts of the programs will be the mathematical logic and processing, but I do need a GUI front-end for data input, setting parameters, results display, etc. I've downloaded both Visual C++ Express and Visual C# Express, but haven't explored them yet, and I'm trying to decide which one to pursue.

I believe that with Visual C++ Express you **can't** use MFC. You may download the Platform SDK (separate download) and develop your GUI code using that SDK (not MFC).

But if you find it hard to develop GUIs using VC6 and MFC (with all the wizards and IDE support), I think you will have hard-time to develop using "pure" Win32 Platform SDK...

Re: Semi-newbie question on Visual C++ vs. Visual C#

In terms of underlying code for program logic, I'd prefer to work in C++, both because I know it already, and because it is more universal than C#. (So, for example, it would be easier to port the code to some other environment down the road.)

There are C# projects in development under GNU/Linux (like Mono and DotGNU). But I don't know their level of compatibility with Microsoft's C# and IDE...

On the other hand, I don't want to waste time learning GUI programming again.

If you don't want to learn GUI programming, then choosing Visual C# 2005 Express would be a better option. It's just drag-and-drop and add event handlers... :) Very RAD.

I want to be able to slap together a basic GUI using drag-and-drop style development for GUI components. A very quick peek at Visual C++ Express does not immediately reveal a drag-and-drop development tool, but I may have missed something. (Have not even installed Visual C# Express yet.) So, in the end, I'll go with whichever language makes its easier to put the GUI together, and if that means learning C#, I'll do that.

So you should learn C#.

And consider also that C# is basically C++ without pointers and with some interesting features (like garbage collector, built-in reflection, a huge library – the .NET framework, etc.)

However, if you want to implement very complex data structures, you may also consider support of libraries like STL or Boost. In this case, for what I know, you have to use C++ (maybe there's a STL.NET, but I think it is under development...).

I think that building the core of your app in C++ and building the GUI using C# RAD-system, and call C++ code from C# (using C-interface DLL or COM) would be the best choice for you.

MrAsm

Joseph M. Newcomer [MVP]
email: newcomer@xxxxxxxxxxxxx
Web: <http://www.flounder.com>

Re: Semi-newbie question on Visual C++ vs. Visual C#

MVP Tips: http://www.flounder.com/mvp_tips.htm