

Re: Why doesn't C# allow incremental compilation like Java?

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-04/msg02965.html>

- *From:* mwelsh1118@xxxxxxxxxx
 - *Date:* 20 Apr 2007 16:36:03 -0700
-

I've looked into that, but it's not possible to do within Visual Studio (it always compiles projects to dll's). Regardless, it still makes the project the minimum unit of compilation. It seems like ideally the time it takes to re-compile should be proportional to the number of changed classes within a project. Obviously changing the non-private interface forces the dependencies of a changed class to be re-compiled, but if I add a single statement to a single method of a single class it takes exactly the same amount of time to compile as if I'd added a statement to every class in the project.

Mark

On Apr 19, 10:31 pm, "Nicholas Paldino [.NET/C# MVP]" <m...@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:

That's not true, really. You could compile into individual netmodules (using the /target:module switch) and then assemble them into an assembly using the assembly linker tool (al.exe).

You should be able to configure VS.NET 2005 to output a module by changing the project file (specifically, the inputs to the csc task).

Hope this helps.

--
- Nicholas Paldino [.NET/C#MVP]
- m...@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

<mwelsh1...@xxxxxxxxxx> wrote in message

news:1177027683.624336.7030@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Why doesn't C# allow incremental compilation like Java?

Re: Why doesn't C# allow incremental compilation like Java?

Specifically, in Java I can compile single .java files in isolation. The resulting individual .class files can be grouped into .jar files.

In C#, there appears to be no analog. I have to compile all my .cs files into a single .dll.

This has serious drawbacks in terms of compilation. With Eclipse, I change a file and only that file is re-compiled. With Visual Studio, I change a file and have to manually re-compile, which may take 10+ seconds for large projects (even if all I did was add a space to a comment).

Why was this design decision made? And is there any way to speed up Visual Studio 2005 C# compilation? Are there plans to have background compilation in future versions of Visual Studio?

Thanks.