

Is there a tool to convert ANSI C code to C#?

Asked 15 years, 7 months ago Modified 15 years, 7 months ago

Viewed 7k times



5



I need to import some ANSI C code into a project I'm working on. For reasons I prefer not to go into, I want to refactor the code to C# rather than trying to wrap the original code. It will take me perhaps a couple of days to do the work by hand, but before I start, is there an automated tool that can get me most of the way there?

I'm a cheapskate and the work I'm doing is pro bono, so free tools only please.

c#

c

Share

Improve this question

Follow

asked May 11, 2009 at 17:52



[Tim Long](#)

13.7k ● 20 ● 86 ● 155

Glad to see you admit you're a cheapskate. ;-) – [peSHlr](#) Aug 2, 2009 at 20:33

I'm actually not a cheapskate on my own accord, but I'm giving my time to that particular project for free and so I'm not prepared to spend money on it. When I do paid work, I try to

use the best tools I can afford. – [Tim Long](#) Aug 3, 2009 at 16:03

4 Answers

Sorted by:

Highest score (default) ◆



2



If manual refactoring is "only" going to take a few days, that would get my vote. Depending on what the C code is doing and how it is written (pointers, custom libraries, etc.) an automated converter may just make a mess. And untangling that mess could be a larger task than just converting by hand.

Share Improve this answer

answered May 11, 2009 at 18:38

Follow



[Mark Maslar](#)

1,121 ● 4 ● 16 ● 28



1



Setting up, cleaning up, refactoring, and just plain making converted code (if there is even a converter available) would probably be something in the magnitude of weeks or months, so you'll be better served just to go ahead with the manual rewrite.

Share Improve this answer

answered May 11, 2009 at 18:03

Follow



[Michael Meadows](#)

28.4k ● 4 ● 49 ● 63



There is an experimental [CLI Back-End and Front-End](#) for GCC. It is already capable of compiling a subset of C

1 programs into CIL, the byte-code that the CLR runs.



(The webpage makes it seem like the code was only developed over a few months and then ignored since then, but it's out of date; [ST Microelectronics](#) is continuing maintenance and development.)



You don't specify *why* you want a C to C# translator, but if you just want to get C and C# to play together without P/Invoke or COM, it might be good enough.

Share Improve this answer

answered May 11, 2009 at 22:52

Follow



[ephemient](#)

204k ● 39 ● 288 ● 399



1

It may make sense to start by getting the existing code to compile as managed C++ aka C++/CLI. Assuming that went smoothly enough, then you have a working, testable foundation on which to build. Move key features to their own classes, and as needed, rewrite as C# along the way.



Share Improve this answer

answered May 12, 2009 at 3:55

Follow



[RBerteig](#)

43.3k ● 7 ● 90 ● 130