

Visual Studio™

Magazine

Turn Your .NET App Into Java

The idea is obvious in retrospect, yet so outrageous that it's no wonder it took so long for someone to act on it. C# and Java have significant similarities, and even if you've never examined MSIL or Java bytecode, you can imagine that they do pretty much the same thing. Those few who have acted on these observations have built relatively simple static translators between C# and Java source code. What Mainsoft has done with Visual MainWin, however, goes several steps beyond that.

Visual MainWin is a plug-in to Visual Studio that lets you write applications in any .NET language, then convert the .NET IL into Java bytecode dynamically. The process is so seamless that you can even use the Visual Studio debugger on runtime Java code; Visual MainWin converts the bytecode back to .NET IL to engage the debugger.

You start by selecting a Visual MainWin project from the New Projects list, and configure your application server if necessary. Then write your app in the .NET language of your choice. You can add Java references, and the documentation describes other structures

Visual MainWin 1.6

Mainsoft

Web: www.mainsoft.com

Phone: 800-624-6946; 408-200-4000

Price: \$5,000 per developer (two-year license); \$2,500 per server (distribution license)

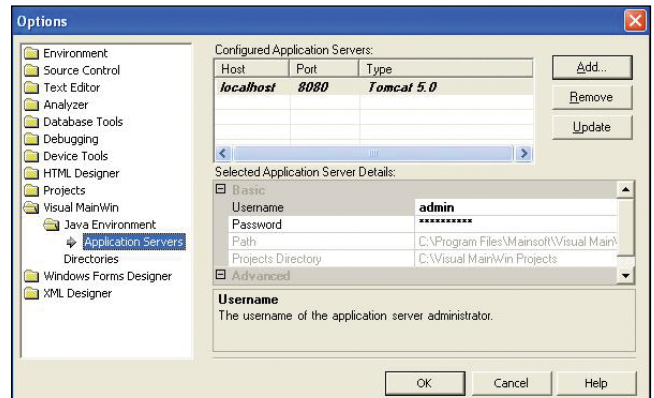
Quick Facts: Converts .NET applications into Java bytecode.

Pros: Seamless Visual Studio integration; highly intuitive in actual use.

Cons: Not quite as automatic a translation as it first appears.

you should pay attention to when writing your code. Building is equivalent to building a .NET project (and a command-line build is also available).

Differences between the .NET and Java platforms make it impossible to perform strict one-to-one mapping. For example, exceptions from the .NET platform are derived from the System.Exceptions class, while there are two possible classes in Java, depending on whether the exception is raised from the app or the underlying



Convert .NET Apps to Java. Mainsoft's Visual MainWin lets Visual Studio developers create a .NET application and translate it to an equivalent Java app dynamically. You can even set up and configure a Java application server from within the Visual Studio environment.

JVM. There's no way to catch language exceptions when .NET code is translated into Java bytecode.

Don't let these minor hiccups deter you from employing Visual MainWin. For those times when you must deploy an application as a J2EE app, you don't need to bring on Java developers. Instead, you can develop the app in a .NET language and run it through Visual MainWin to meet the deployment needs.

The Developer Edition, which supports only Apache Tomcat, is freely downloadable from <http://dev.mainsoft.com> (it also lacks source-code integration and some third-party control support). Both this and the Enterprise Edition require Visual Studio 2003, and come bundled with both a Java SDK to execute Java on your development system and Tomcat to work with a Web application.

Pricing for Visual MainWin Enterprise Edition, which includes support of commercial application servers, is \$5,000 per developer and \$2,500 per server for a distribution license (there are no runtime fees for internal use). This is a fair deal for a product that can save hundreds of thousands of dollars in development costs under the right circumstances. **VSM**

Peter Varhol writes a Java column for *Java Pro* magazine, and he writes and speaks on architecture and development topics using both Java and .NET.

Reprinted with permission from Visual Studio, September 2005. © Fawcette Technical Publications. All Rights Reserved. On the Web at www.enterprise-architect.net. FosteReprints: 1-866-879-9144