C# for Java Developers

Patrick Linskey
plinskey@gmail.com
@plinskey
Who am I?

- Patrick Linskey
- 13 years of Java experience, 11 professionally
- 2 years of professional C# experience
Who are you?

• Hopefully, Java developers who are interested in learning about C#
• ... for client, server, or mobile deployment
Apology to Readers

Much of this talk happens in Visual Studio.

Sorry about that.

I will make the code available at http://github.com/pcl -- the commit history provides some detail.
[Anders Hejlsberg]
Some Terminology

- **Common Language Runtime**, not JVM
- **Intermediate Language**, not Java bytecode
- Assembly, not JAR
- **[Attribute]**, not @Annotation
[Syntax]
Coding Conventions

• All methods use InitialCaps
• Constants use InitialCaps instead of ALL_CAPS
• Namespaces are not as consistently-scoped as are packages in the Java world
• Assembly names are first-class constructs
[Unit Testing]
MSTest vs. NUnit

• I use NUnit because it seems more familiar
• Without ReSharper, NUnit / Visual Studio integration is a bit weak
Visual Studio

- Most people seem to use it
- $$$
- ReSharper (plugin from JetBrains) is a must
- #1 awesome Visual Studio feature: program counter manipulation during debug sessions
[LINQ and Functions]
LINQ

• Can evaluate LINQ queries against database, XML, collections, etc.
  • Anything IEnumerable or IQueryable

• Also: JOIN, GROUP BY, HAVING, ...
Function Objects

- We saw Func<int, int> in fib
- Generally: Func<TArg1, TArg2, TReturn>
- Support for up to 16 arguments!
Action Objects

- `Action<TArg1, TArg2, ...>`
- Also supports 16 args!
- Equivalent to a void method return type
Tuples

- Tuple<T1, T2, T3, ...>
- Up to 8 items
- Useful for easy multi-object returns, LINQ queries
Anonymous Types

• Predate tuples

```csharp
var ratio = new {
    Numerator = 2,
    Denominator = 3
};
Console.Out.WriteLine(  
    ratio.Numerator / ratio.Denominator);
```
[Object-Orientation]
Virtual keyword

- By default, methods are resolved at compile time
- To get polymorphic method invocation, highlight the methods as ‘virtual’
The Ecosystem
Third-Party is Weak

- Outside UI widgets, nowhere near as much activity as in the Java world
- Few open-source projects
- http://stackoverflow.com is your friend
Spring.NET is great

- Lags the Java Spring Framework a bit
- I’ve only used **dependency injection** and **lifecycle management** features

- WPF (and presumably Silverlight) have their own resource management
- It’s possible to bridge the two
Yay!

- Generics are reified, and support covariance and contravariance
- Dynamic method invocation with the ‘dynamic’ pseudo-type (akin to ObjC’s ‘id’)
- `+=` syntax for all delegate fields / properties
- `Func`, `Action`, raw Delegate
More Cool Stuff

• `@` strings
• Indexers
• Partial classes (UI and codegen)
• Structs
Some Missing Features

- No anonymous inner classes (use closures / function objects instead)
- All inner classes are static
- Enums cannot have state or methods
- Can emulate methods with extension methods
Questions?

Patrick Linskey
plinskey@gmail.com
code: http://github.com/pcl