



Developing a Windows 8 application with F#

Gustavo Guerra

London Windows Store App Developer Group



Learn On The Go

Demo

Why F#?

- Functional First
 - Immutability by default
 - Powerful abstractions (LINQ+++)
- Strongly Type System
 - Friends don't let friends use null
 - Option type
 - Whole class hierarchies in handful of lines
 - Discriminated unions
 - Record types

Why F#?

- Powerful but Light Syntax
 - Significant whitespace
 - Everything is an expression
 - Pattern matching
 - Type Inference
 - Almost feels like a dynamic language like python
- Easier to Understand
 - Declarative feel
 - Explicit compilation order
 - Signature files

Why F#?

- REPL
 - Testability
 - More productive than compile-run cycle
 - Similar to dynamic languages (Python, Ruby, etc...)
- All your types are belong to us
 - Type Providers

Why F#?

- Very good Async support
 - What's in C# 4.5
 - More composable
 - Built-in cancellation
 - Started explicitly
- <http://tomasp.net/blog/csharp-async-gotchas.aspx>

Windows 8

- No Support for WinRT in F#
 - View has to be done in C#
 - F# used for logic and view model in Portable Class Library
 - Data binding works nicely
- F# compilation to WinJS
 - <https://github.com/ZachBray/FunScript>
 - <http://www.websharper.com>

Want to know more?

- <http://fsharp.org>
- <http://fsharpforfunandprofit.com/>
- F#unctional Londoners Meetup Group
 - Two meetups each month (one of them is Hands-On)
 - F# on iPad and iPhone with Xamarin Studio with Neil Danson - Thursday, 30th May