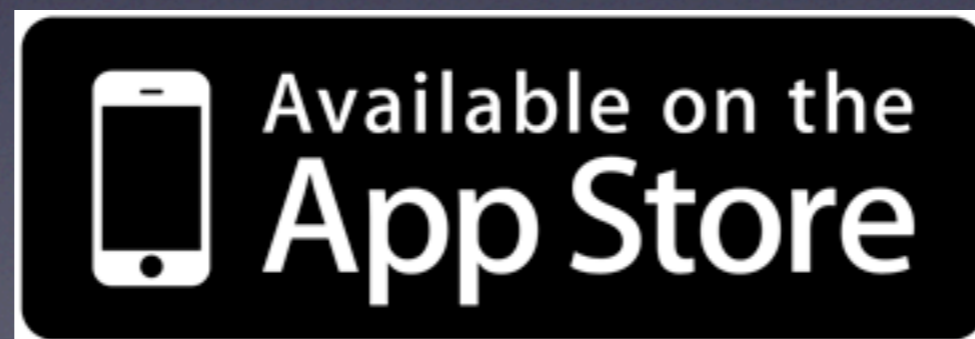


# Developing mobile apps in .NET

MonoTouch and MonoDroid



Apple really needs a mascot



# Topics

- A bit of background, what you need to get started
- MonoTouch
- A little MonoDroid
- Cross-platform options
- Other stuff

# The marketing slide

- Chances are, you know this.
- iPhone: there are lots. Still the “cool” device
- Android: there are even more lots, and it’s growing quickly
- WP7: new but promising

# Background

- iPhone apps are normally written in Objective-C (same as most Mac apps)
- Apple provides a free tool chain (XCode)
- Objective-C is “ok”, but feels very 1995
  - Semi-manual memory management
  - Strange syntax
- Under the hood it's all ARM assembly!

# What you need (iOS edition)

- An Intel Mac\*
- XCode (free from Apple)\*\*
- iDevices (iPodTouch, iPhone, iPad)\*\*
- A good idea helps. As does some design skills.

\* Costs money. No way to get around this tho (...VMWare? ...)

\*\* Costs money if you want to put it on a device.

# Rules

- Can't interpret (no python, ruby, .NET)
  - Not well enforced (Lua in most games)
- Can't JIT (can't execute from data space)
- Can't download code and run it.
  - Strictly enforced
- Nice published rules around AppStore approvals

# Side note:

## What about 3.3.1?

- Clause 3.3.1: Thou shalt not use anything but C, C++, Objective-C or Javascript (via the browser).
- Removed mid-2010.
- **Was never enforced.**
- Purpose: Piss off Adobe. Worked great!

# Non-Apple options

aka “I don’t wanna use XCode”

- HTML-based: PhoneGap, Appcelerator, web apps
- Flash (kinda)
- .NET-based: MonoTouch, Unity



# What you need

## (MonoTouch Edition)

- All of the previous stuff, and...
- Mono
- MonoDevelop
- MonoTouch\*\*

\* Costs money. No way to get around this tho (...VMWare? ....)

\*\* Costs money if you want to put it on a device.

# What you need to know

- .NET (but you know this) - C# 3
- CocoaTouch
- Some Objective-C (to read samples)
  
- Don't underestimate the size of CocoaTouch

# Architecture

YOUR APP

COCOA TOUCH

OTHER LIBS

OpenGLES

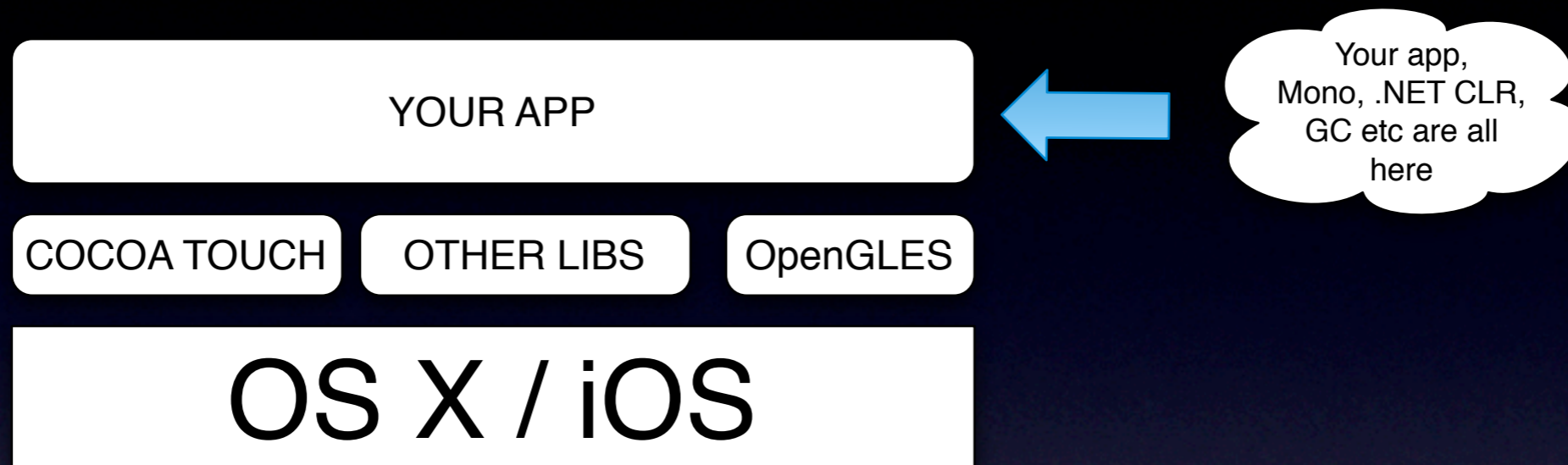
OS X / iOS

# MonoTouch



- Uses the Mono libraries (Silverlight profile)
- AOT compiler
  - Like JIT, but at compile time with a linker
- Full access to everything Apple provides

# Architecture

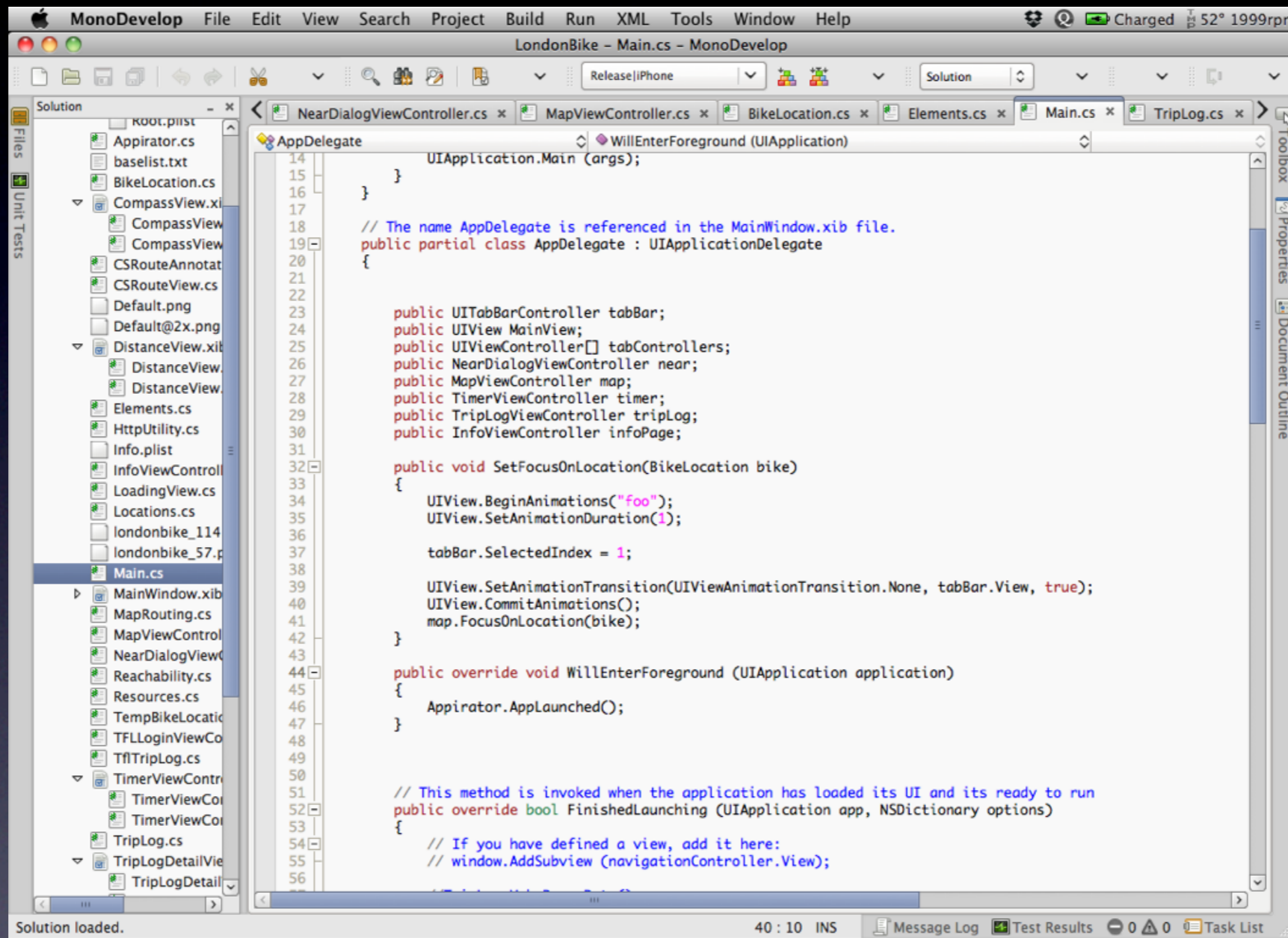


- Everything is in the binary
- End result is identical to an XCode app
  - If they embedded Mono, that is!
  - Binary is about 2MB larger than normal

# MonoTouch

- Wraps ALL of the CocoaTouch libraries
  - Enough to use Apple docs
- ..and you have all the (non-visual) bits of .NET too (up to C# 3.5 - 4.0?)
- You can call into Obj-C using btouch (eg Three20, the Facebook lib)
- You will never “port” winforms/WPF to this. Don't try.

# MonoDevelop



- Mac-based IDE; Open Source

# iPhone Apps

- Window-based apps (UIKit/CocoaTouch)
  - Navigation-based apps - lists, lots of lists
  - Forms (single canvas)
- OpenGL apps
  - Full control over screen etc.
  - Usually games



# CocoaTouch

- Very rich API - equivalent to .NET + WPF + WCF + more
- Strongly MVC (and other patterns)
- Similar to Cocoa (desktop), but with new touch-based classes (normally start with UI vs NS)
- Some things are very easy to do

# OpenGL

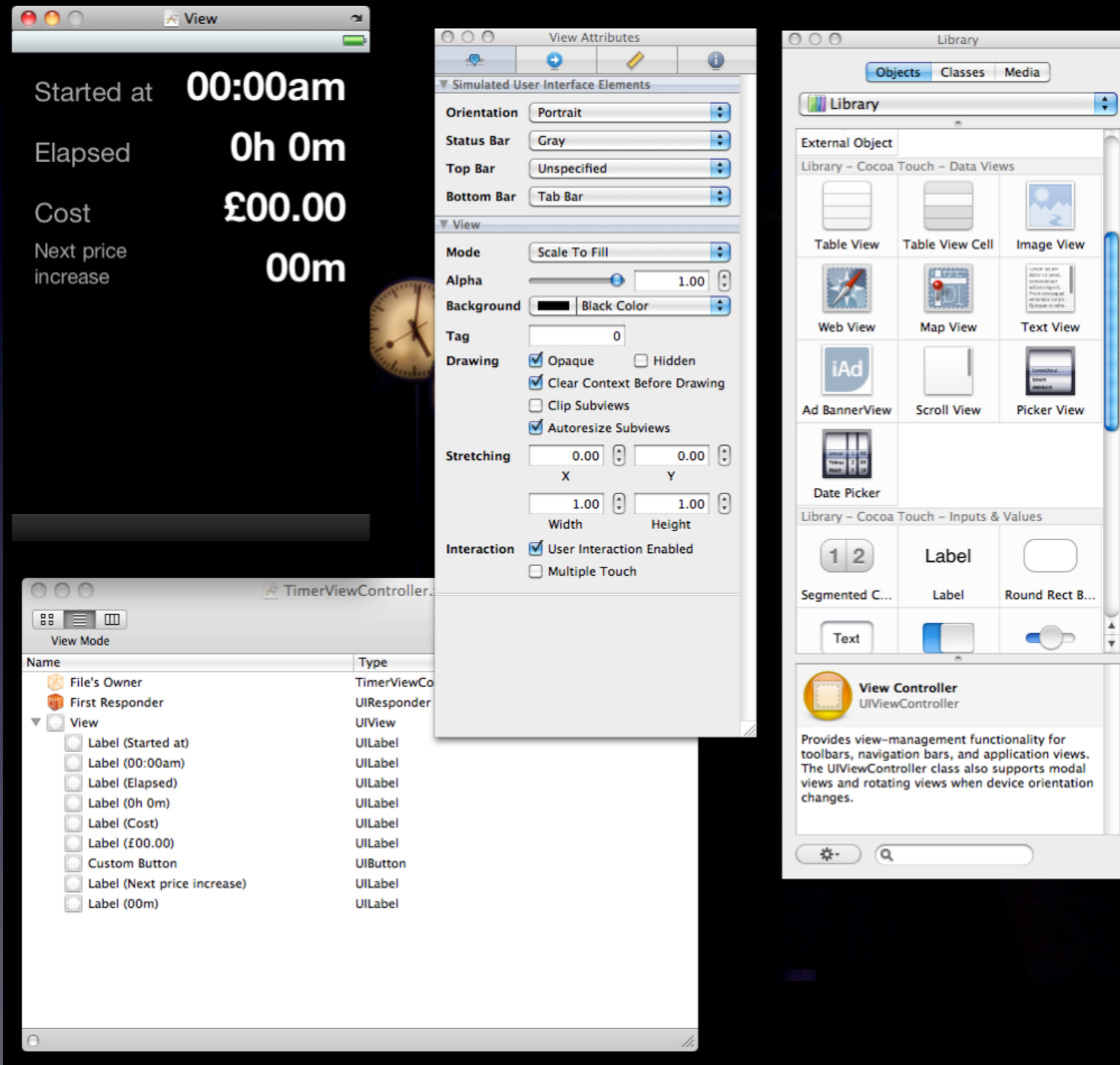
- Go find @majicdave (who wrote Chopper / Chopper 2)
- Bring him beer
- Talk OpenGL (it's the same calls as Objective-C)



# Building interfaces

- Interface Builder (part of XCode)
  - NIBs and XIBs
  - Very strong i18n support
- Do it in code
- End result is the same, it's an object graph

# Interface Builder



# Code

```
UIApplication.SharedApplication.StatusBarStyle = UIStatusBarStyle.BlackOpaque;
tabBar = new UITabBarController(); MainView = tabBar.View;
window.AddSubview(MainView);

near = new NearDialogViewController();
map = new MapViewController();
tripLog = new TripLogViewController();
infoPage = new InfoViewController();

timer = new TimerViewController{
    TabBarItem = new UITabBarItem("Timer", Resources.Timer, 1)
};

tabControllers = new UIViewController[] {

    new UINavigationController(near) {
        TabBarItem = new UITabBarItem("Near", Resources.Near, 0)
    },
    new UINavigationController(map) {
        TabBarItem = new UITabBarItem("Map", Resources.Map, 2)
    },
    timer,
    new UINavigationController(tripLog) {
        TabBarItem = new UITabBarItem("Trip Log", Resources.TripLog, 3)
    },
    new UINavigationController(infoPage) {
        TabBarItem = new UITabBarItem("Info", Resources.Info, 4)
    }
};

tabBar.SetViewControllers(tabControllers, false);

if (localNotification)
{
    tabBar.SelectedIndex = 2;
}

window.MakeKeyAndVisible ();
```

# Demo

- Lets write something simple
- Demo a few less simple things

# Design

- You are going to need some design skills
  - Yours
  - Apples - HIG
  - Pay someone
- Tapworthy is worth buying
  - Josh is coming to WebStock

# Libraries

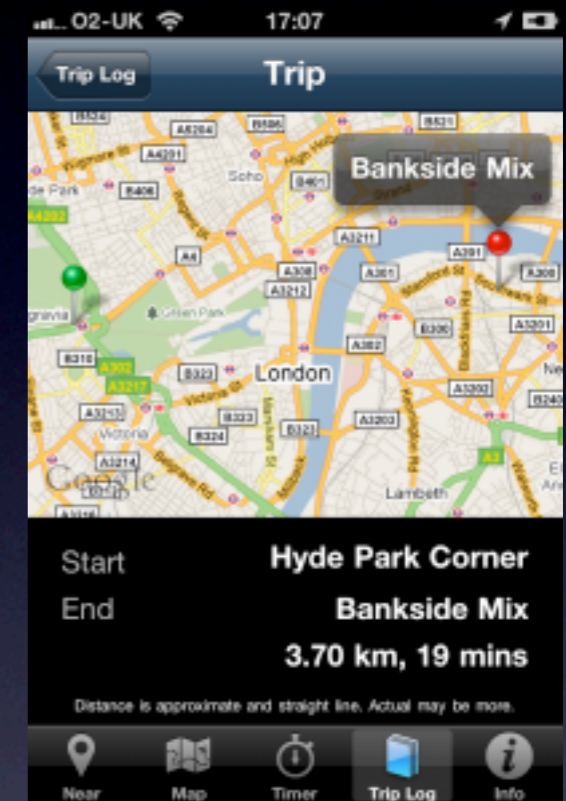
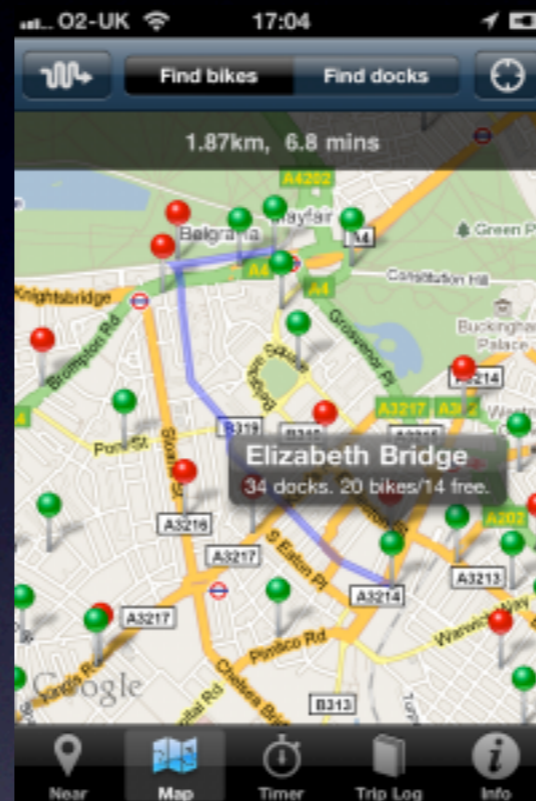
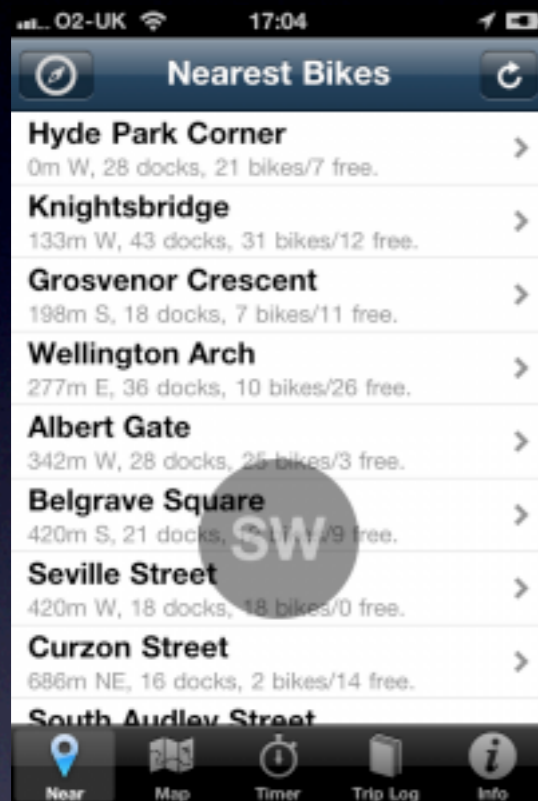
- SQLite (included), various ORMs
- AddressBook, Audio, AVFoundation, CoreAnimation, CoreFoundation, CoreGraphics, CoreLocation, GameKit
- MapKit, MediaPlayer, MessageUI, OpenGL ES, StoreKit
- New lib's usually available within 24-48 hours, eg in iOS 4.2



# Some apps

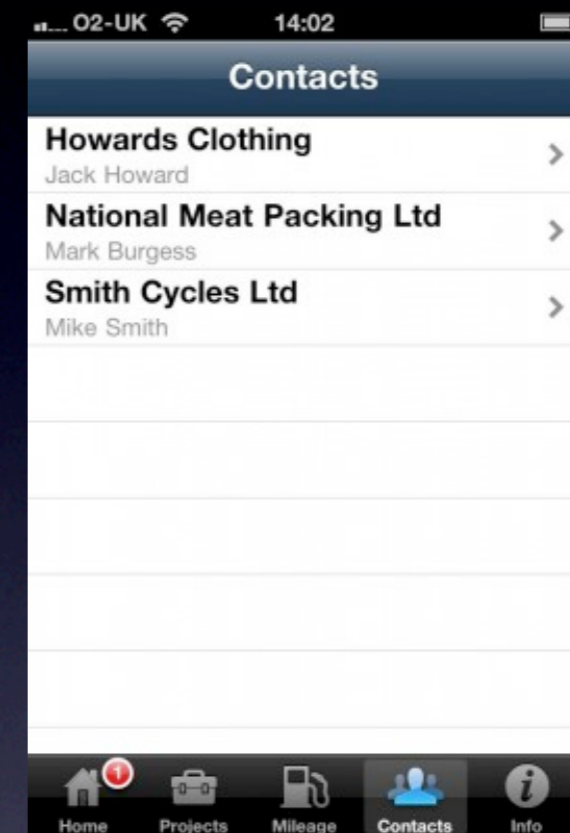
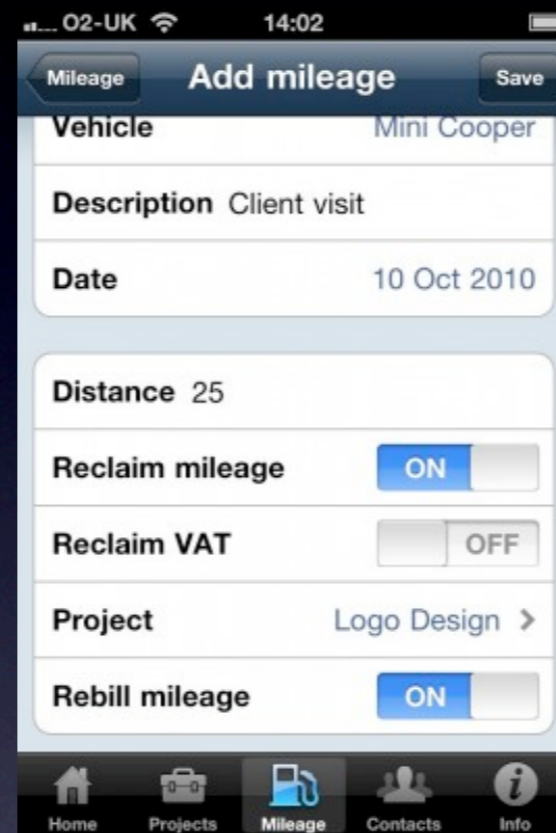
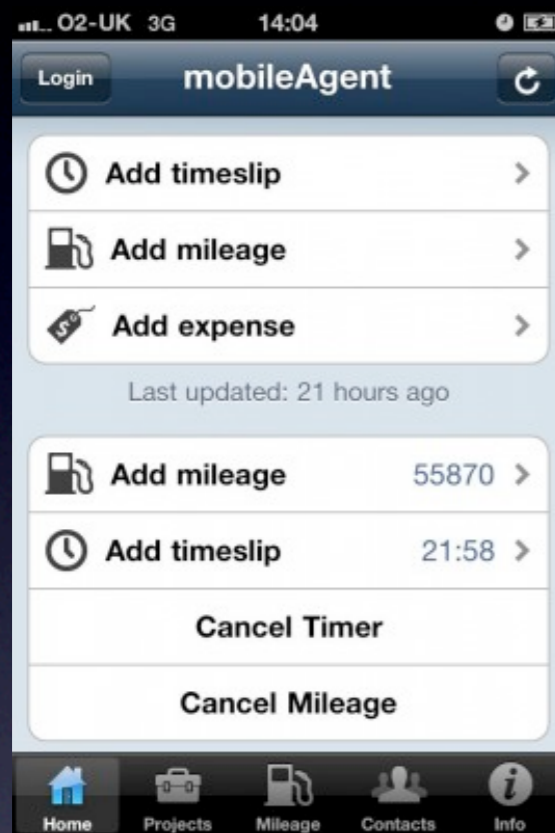
- Very small sample
- All look and work exactly like “normal” apps
- More at:
  - [monotouch.net/Apps](http://monotouch.net/Apps)

# London Bike App



- Find bikes in the London Cycle Scheme
- CoreLocation, lists, MapKit, AppEngine backend

# mobileAgent



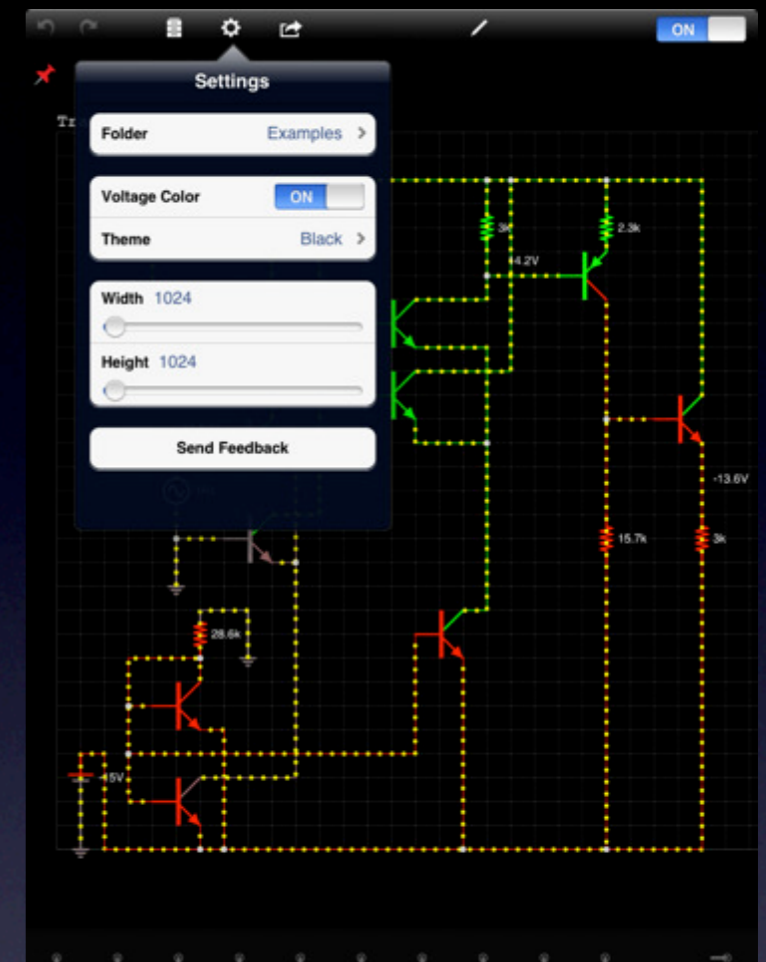
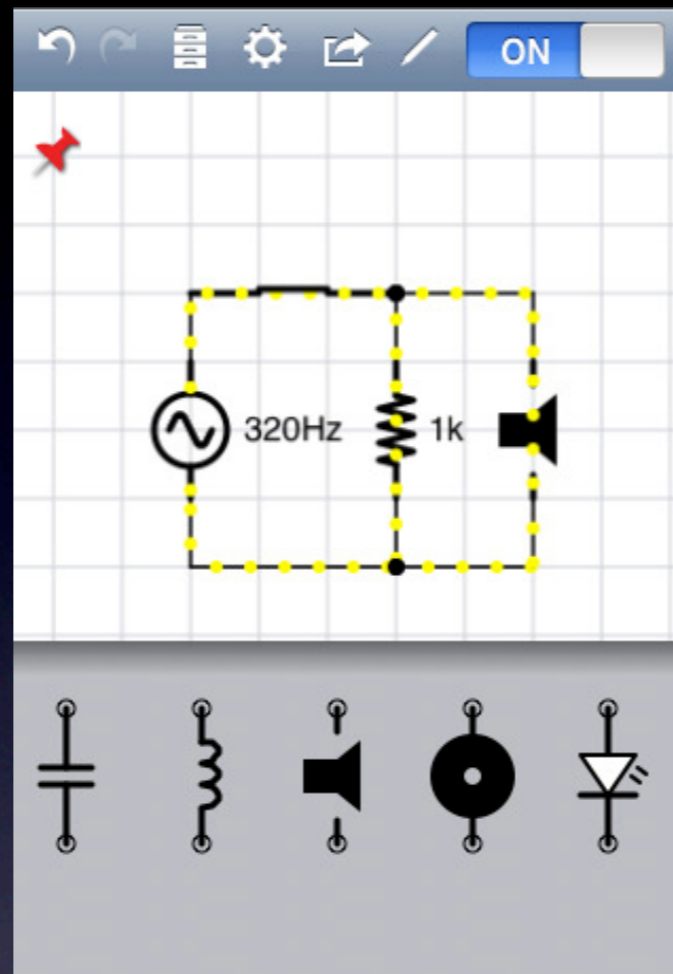
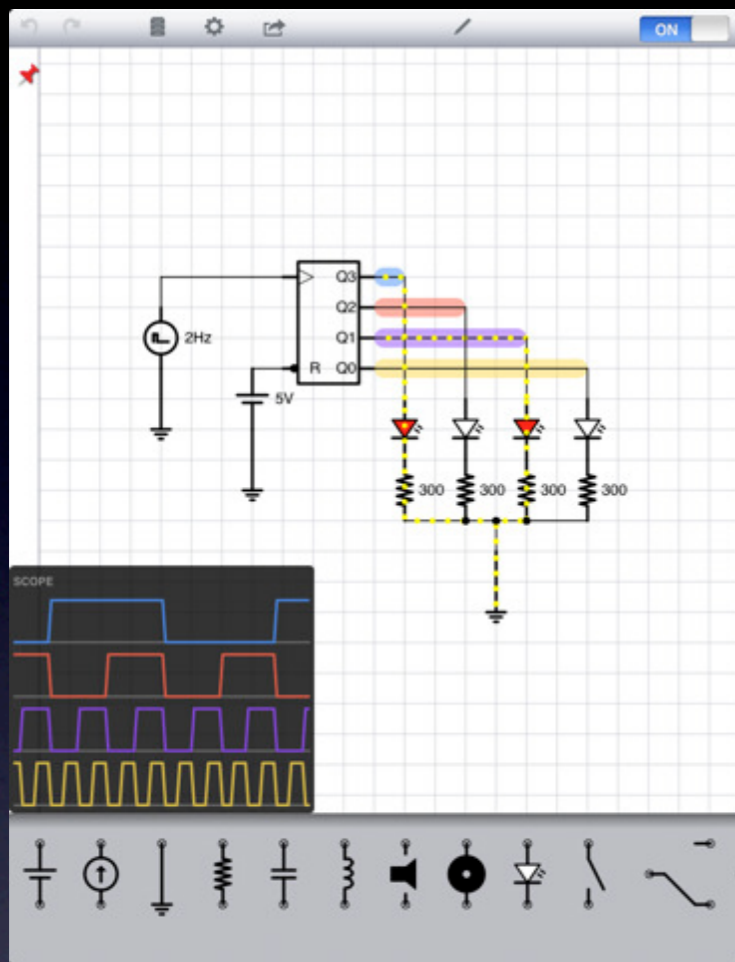
- Client for FreeAgent (UK-focused Xero competitor)
- SQLite, lists, lots of XML processing, Camera

# Quicklytics



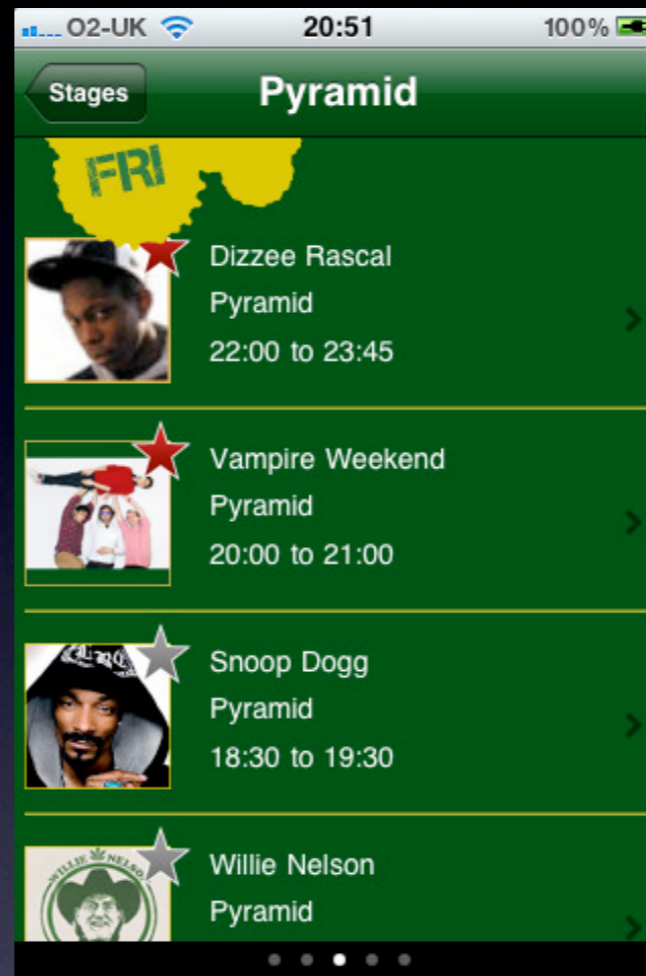
- iPhone client for Google Analytics
- Google OAuth, highly visual

# iCircuit



- Realtime circuit simulator (iPad and iPhone)

# Festival Star

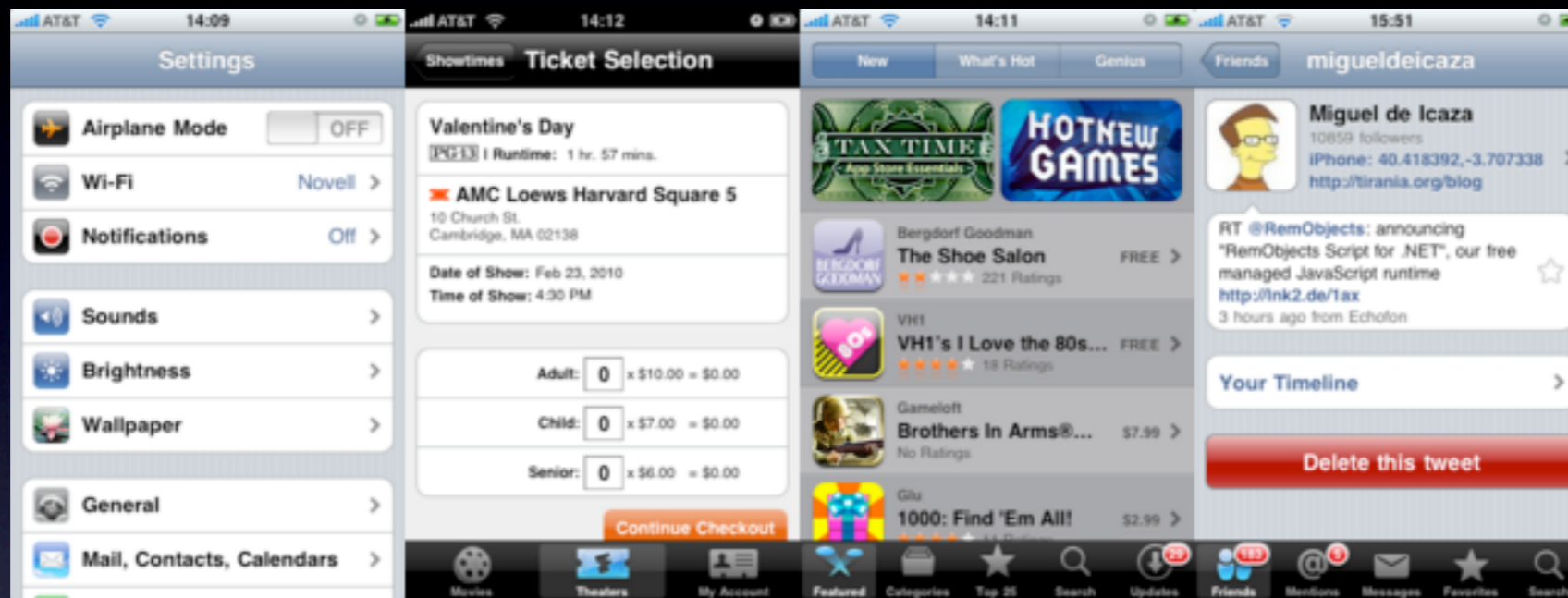


- Line-ups etc for all major UK festivals
- Big backend database of acts, music etc

# Handy stuff

- MonoTouch.Dialog
- Blogs, IRC, active mailing list
- Unity (same idea, game framework)
- XNATouch

# MonoTouch.Dialog



- These are all lists (UITableView)
- Makes list applications trivial to write
- OSS/GitHub, active development
- Noise about porting it to MonoDroid.



# Demo

- MonoTouch.Dialog

# Support / Books

- Blogs (quite a few)
- IRC (core dev's hang out on it EST/CST)
- Book - Pro iPhone Programming with MonoTouch
  - Another on the way

# MonoDroid

- Still very much in CLOSED beta
- Use .NET to write Android apps
- Different limitations
  - Can JIT! Can deploy Mono as a framework to the phone. No AOT.
- Windows-based - VS2010 plugin





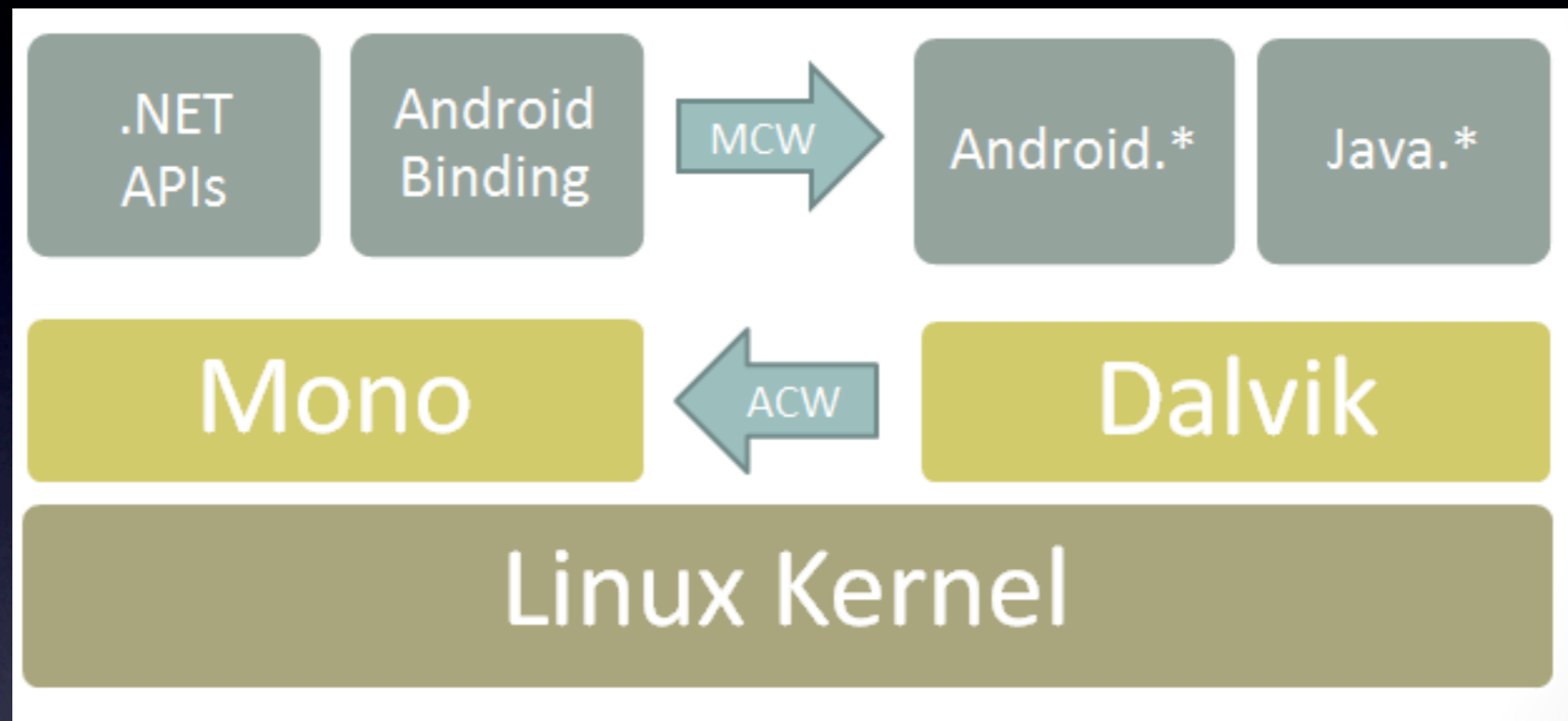
# MonoDroid

- Same .NET profile as MonoTouch (and WP7): Silverlight, most of .NET 4.0
- Wraps all Android API's
  - (by RTM, anyway...)
- Can call to Java and native code
- Deploy Mono with app or as shared lib via the AppStore

# What you need

- A Windows machine - VM works too
- The Android SDK (free)
- Visual Studio 2010 (paid version, not express)
- MonoDroid

# Architecture



- Mono is a peer to Dalvik
- Interop between mono, dalvik and native code

# What you'll need to learn

- Android API's (Activities, Intents, Services, Content providers, project structures, XML layout etc)
- A bit of Java to read examples
- Android UI conventions

# Crossplatform

- Android, iPhone, WP7, iPad
- UI would change. Interactions would change. Plumbing code and framework might not.
- Native app in all cases, no “lowest common denominator”
- MonoTouch. Dialog between all platforms would help a lot



# Gotchas

- You WILL need to write the UI on each platform
- You most likely will need to write a lot of the glue code on each one
- You will need to abstract away a fair bit of stuff (IoC might help here)
- It's not write once, run anywhere.

# If you must have a single code base

- Think again. And then again.
- Look at SenchaTouch, jQueryTouch etc
- Consider PhoneGap or Appcelerator to wrap the webapp

# Summary

- If you need single codebase, cross platform:
  - Write a webapp. Be prepared to be underwhelmed.
- If you want native, and have .NET skills
  - Mono\* is a good way to do it
  - It's NOT a silver bullet tho!
- Otherwise, consider the native tools

# Get MT on the cheap!

- Discount code from Novell
- 10% off until 15th January
  - Normally \$530
  - Now \$477
- Code: (Ask me)

# Sign up for MonoDroid

- [www.monodroid.net](http://www.monodroid.net)
- They add people fairly often
- If you have a urgent need, and can make a case, email them!

... and one more thing



# MonoMac

- MonoMac
- OpenSource Mono bindings for Cocoa (desktop mac apps)
- Takes a lot of ideas from MonoTouch
- Write full Mac apps in C#, embed Mono
- Aiming to work on the Mac AppStore
- [mono-project.com/MonoMac](http://mono-project.com/MonoMac)