Starting out right with Xamarin

and other useful tips, tricks and resources
p.s. this is just my opinion

RYAN DAVIS

QUEENSLAND C# MOBILE DEVELOPERS MEETUP

2015 06 23

whoami

- Ryan Davis
- Hobby Developer
- Work in ~~Information Management~~ LINQPad all day baby

- Blog: ryandavis.io
- Github: @rdavisau
  - sockets-for-pcl, sockethelpers – write socket comms in shared code
  - jsondatacontext-linqpad – json data context driver for LINQPad

current swag status: fanatic

to cover

- Mobile principles to live by
- Pop Quiz – your first big questions
- Spotlight on some useful libraries
- Roundup of resources.

-= mobile principles to live by =-

#1

know platform

~~love~~ thy ~~neighbour~~

why Xamarin, again?

(not at all MECE)

Xamarin because:

- Write code in powerful, expressive languages
- Take advantage of the .NET platform and ecosystem
- TARGET ALL THE PLATFORMS!!
- Share as much code as we can
- Avoid Java and Objective-C and swift

- ~~Don’t want to have to learn about the native platforms.~~

not all mobile platforms are created equal

Xamarin is not designed to relieve you of the need to understand the underlying platform:

- Android, iOS and WP differ on many levels – UI, app lifecycle, sandboxing and restrictions

Each platform might be more or less suited to a given application or programming paradigm

- MVVM and databinding – WinRT
- Custom drawing and animation – iOS
- Doing whatever you want – Android.

forunately, we can make use of native docs

Xamarin’s iOS/Android frameworks are C# idiomatic, but otherwise almost identical to the native ObjC/Java counterparts

- A book or course targeting iOS or Android native is still relevant, especially for frameworks like UIKit and CoreAnimation
- Tutorials and documentation are easily adapted

Probably the biggest deviation from native API naming:

- tableView:cellForRowAtIndexPath:	→	GetCell(,)
NATIVE iOS		XAMARIN iOS

. : You have access to a lot of platform resources for learning – see the resource roundup for suggestions

-= mobile principles to live by =-

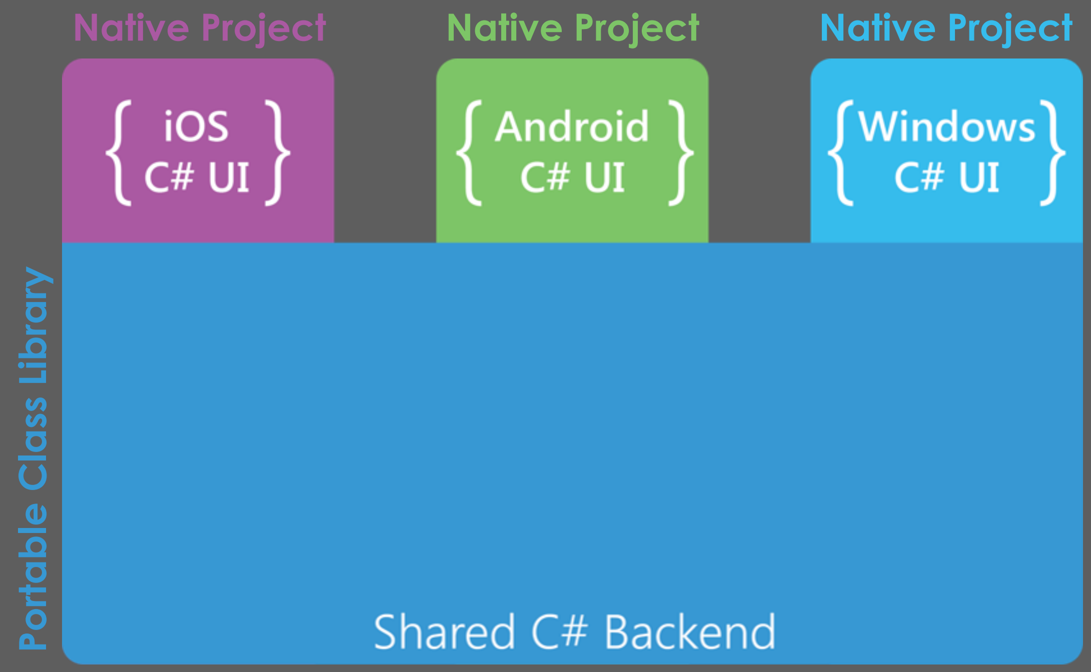
#2

cross-platform
begin with the^end in mind

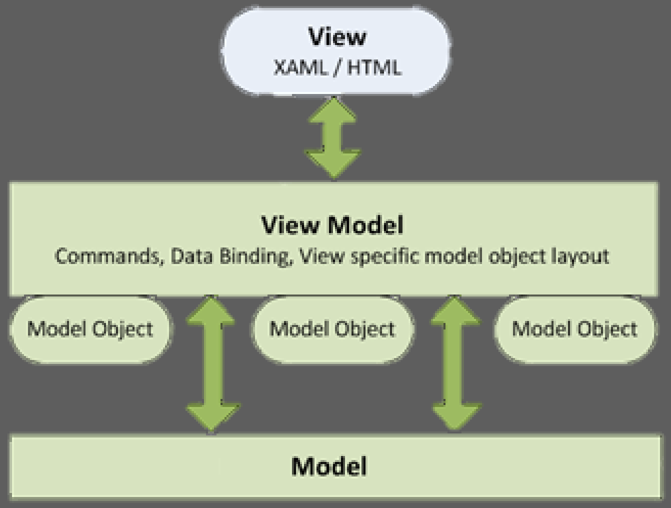
MVVM + PCL is a match made in heaven

The MVVM + PCL method is proven and good

- By now, the community has settled on the combination of MVVM and a portable core assembly as an effective way to write cross platform apps.



From a Xamarin Presentation

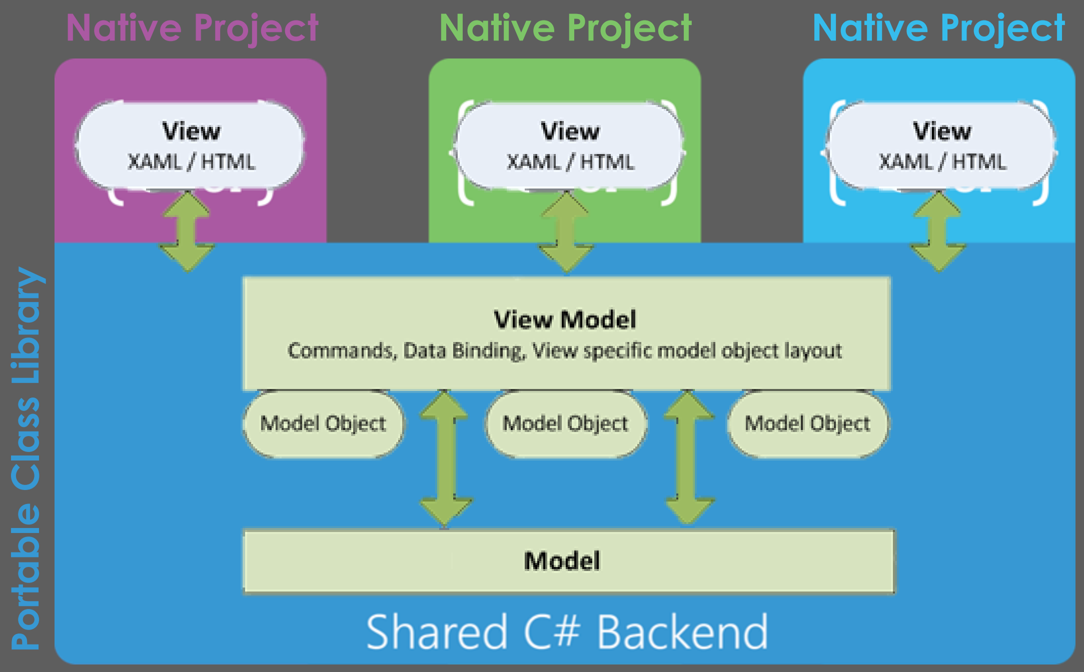


From a google image result for MVVM

MVVM + PCL is a match made in heaven

The MVVM + PCL method is proven and good

- By now, the community has settled on the combination of MVVM and a portable core assembly as an effective way to write cross platform apps.



go MVVM with a PCL from the beginning

Keeping your viewmodels in the core project and your views in native projects encourages Good Habits

- You cannot accidentally use platform-specific and non-PCL APIs
- You cleanly separate your viewmodel and views, and your code goes into the viewmodel by default
- You can write the majority of your application logic without fiddling with UI

and you can do it in LINQPAD if you want

You can access platform-specific functionality transparently using PCL shims.

-= mobile principles to live by =-

#3 try not to re-invent the wheel

cross-platform mobile is new, but not that new

- There are a lot of new considerations that didn’t necessarily apply to the old world of desktop and MVC.
- Many problems have already been solved
  - We do get the benefit of the existing .NET libraries, where PCL-compatible
  - There are a lot of Xamarin libraries out there already

dial before you dig

A little research may save you a lot of hours!

- There probably isn’t a need for another MVVM or DI framework
  - MvvmCross, ReactiveUI, MVVMLight
  - TinyIoC, Ninject, Splat, Autofac, Unity
- There very well could be a need for more cross-platform abstractions
- See the resource roundup for a catalogue of useful libraries and abstractions.

-= mobile principles to live by =-

#4 win friends and influence people

aka. get involved in the community

the Xamarin OSS ecosystem is strong

- Xamarin is new for us, but there are a lot of people who have been doing cool stuff with the platform for a long time
- For newcomers, it can be hard to know what’s out there, which libraries are reliable, etc. it can help to start following the work of others
- For a good summary of what’s happening, subscribe to the Weekly Xamarin Newsletter

giving back has lots of benefits

- Credibility in the community
- Credibility with future clients / employers
- No cost: http://resources.xamarin.com/open-source-contributor.html

  "In order to ensure that all open source projects that want to add or maintain compatibility with Xamarin have access to the tools they need, we offer open source projects with app store compatible licenses non-commercial subscriptions to our products for the purposes of developing, testing, and maintaining compatibility with Xamarin. Active developers on established open source projects can fill out this form to apply"

- You'll feel good helping people

POP QUIZ

Answering the tough questions

question #1

question #1

NATIVE

vs

HYBRID/ WEB

question #2

PCL

PORTABLE CLASS LIBRARY

- A class library that works against a ‘portable’ subset of .NET APIs - those supported by all the platforms you are targeting.
- The de-facto standard for cross-platform Xamarin/Mobile apps.
- Just use this.

vs

SHARED

- Essentially, a hardened version of linked files
- Still useful for sharing XAML between Windows Store/Phone apps
- Seriously, just use a PCL

question #3

Xamarin

vs

Xamarin.Forms

question #3

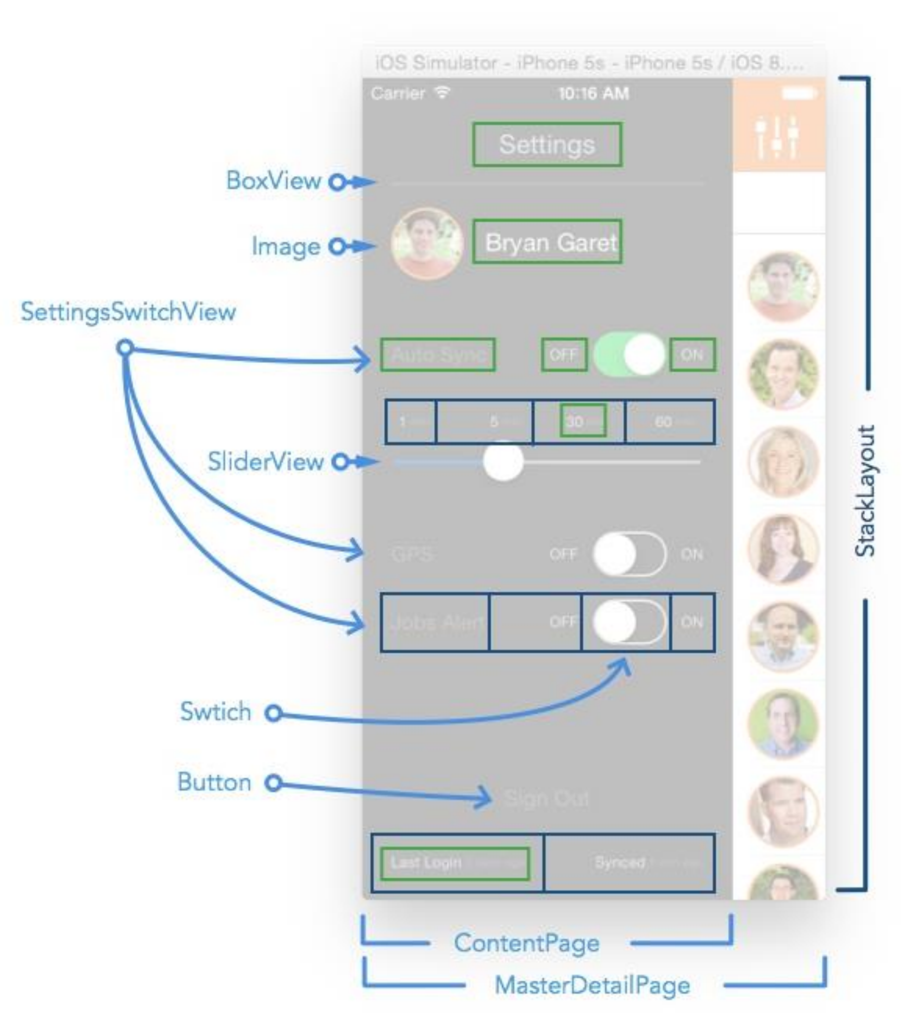
Xamarin

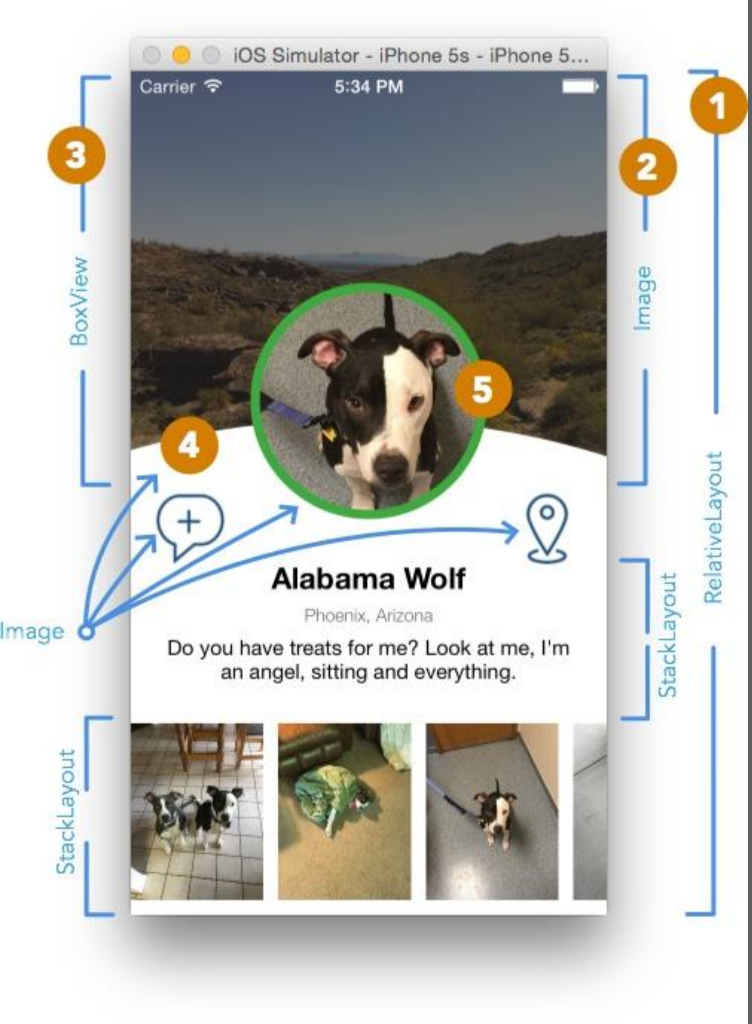
vs

Xamarin.Forms

- Guidance at http://xamarin.com/forms for which platform to use when
- You can mix and match Xamarin.Forms pages with native views
  - https://github.com/xamarin/xamarin-forms-samples/tree/master/Forms2Native
  - https://github.com/xamarin/xamarin-forms-samples/tree/master/Native2Forms
- Xamarin.Forms might be more capable than you thought
  - https://www.syntaxismyui.com/xamarin-forms-in-anger/

question #3 – Xamarin.Forms in Anger

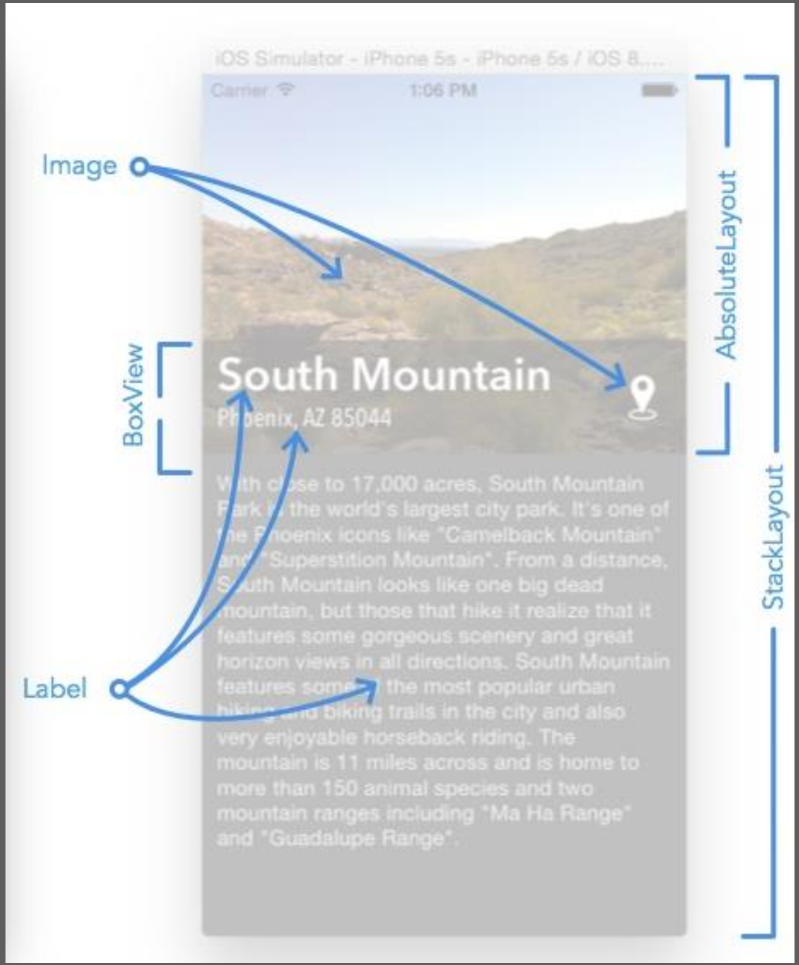




Alabama Wolf

Phoenix, Arizona

Do you have treats for me? Look at me, I'm an angel, sitting and everything.



South Mountain

Phoenix, AZ 85044

With close to 17,000 acres, South Mountain Park is the world's largest city park. It's one of the Phoenix icons like "Camelback Mountain" and "Superstition Mountain". From a distance, South Mountain looks like one big dead mountain, but those that hike it realize that it features some gorgeous scenery and great horizon views in all directions. South Mountain features some of the most popular urban hiking and biking trails in the city and also very enjoyable horseback riding. The mountain is 11 miles across and is home to more than 150 animal species and two mountain ranges including "Ma Ha Range" and "Guadalupe Range".

question #4

Visual Studio

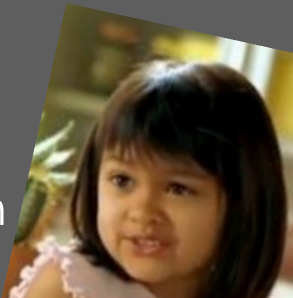
vs

Xamarin Studio

- Resharper
- Good dark theme
- Windows Phone/Store
- Easy to work with Azure and ASP.NET resources if your app involves them
- It’s familiar

- Better designer experience, particularly iOS
- Faster build/deploy times for iOS
- You can make changes to your files while the app is running!
- Sketches…
- “Update NuGet Packages”

I use both



question #5

Parallels vs Bootcamp

ESX vs Hyper-V

question #5 (setting up your dev box)

The true challenge of cross platform:

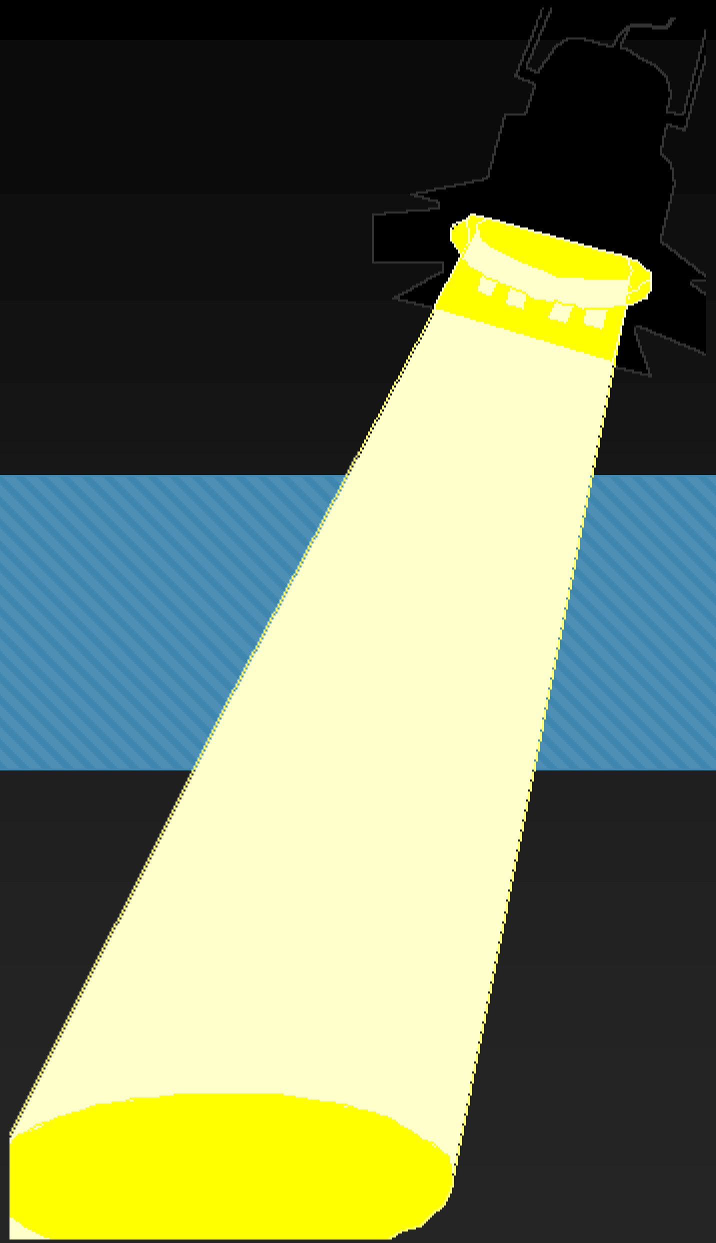
- You need a Mac to build/run iOS apps
- You need to be running Windows + Hyper-V to run the WinPhone emulator
- Android is chill and runs everywhere, thanks buddy

What can we do?

- My advice - Get a Mac and keep your options open
  - Use Bootcamp, you can boot Windows direct or use Parallels/VMWare from OSX
    (Parallels seems to perform best, this result has been consistent for several releases now)
  - If you're brave, you can also boot your OSX partition in VMWare when running in Bootcamp: https://pigiuz.wordpress.com/2013/11/22/how-to-make-a-vm-boot-your-osx-partition-from-windows/

Spotlight

on some cool li s

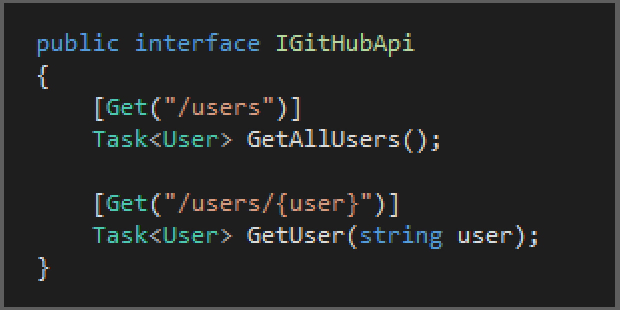


how to use `HttpClient` to query a JSON API

- Step 1: **Reconsider.**

hello ModernHttpClient and Refit

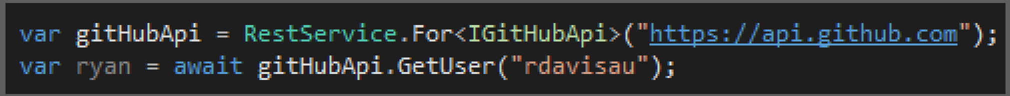
- **ModernHttpclient -** HttpClient implementations that use platform-native HTTP clients for :rocket:
  - Looks like an HttpClient, acts like the best native implementation!
- **Refit -** The automatic type-safe REST library for Xamarin and .NET



```
public interface IGitHubApi
{
    [Get("/users")]
    Task<User> GetAllUsers();

    [Get("/users/{user}")]
    Task<User> GetUser(string user);
}
```

Create an interface for your API, annotated with verb and URL stem



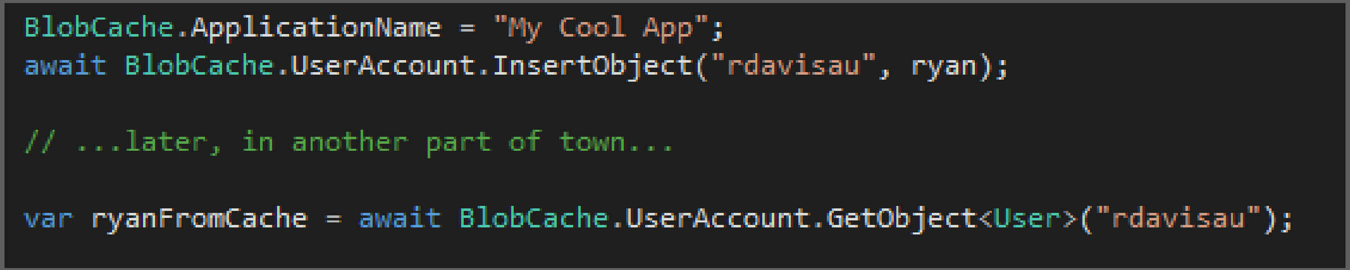
```
var gitHubApi = RestService.For<IGitHubApi>("https://api.github.com");
var ryan = await gitHubApi.GetUser("rdavisau");
```

Type-safe API methods, async, (de)serialization come for free!

  - Supports all the best verbs, body parameters, authentication, static and dynamic request headers and generic interface definitions.

persisting data - Akavache

- Cross-platform file system access is Hard but it has been solved
  - https://github.com/dsplaisted/PCLStorage
  - Consider whether you need file system or just to save stuff
- Akavache gives you async-friendly key-value persistence + more

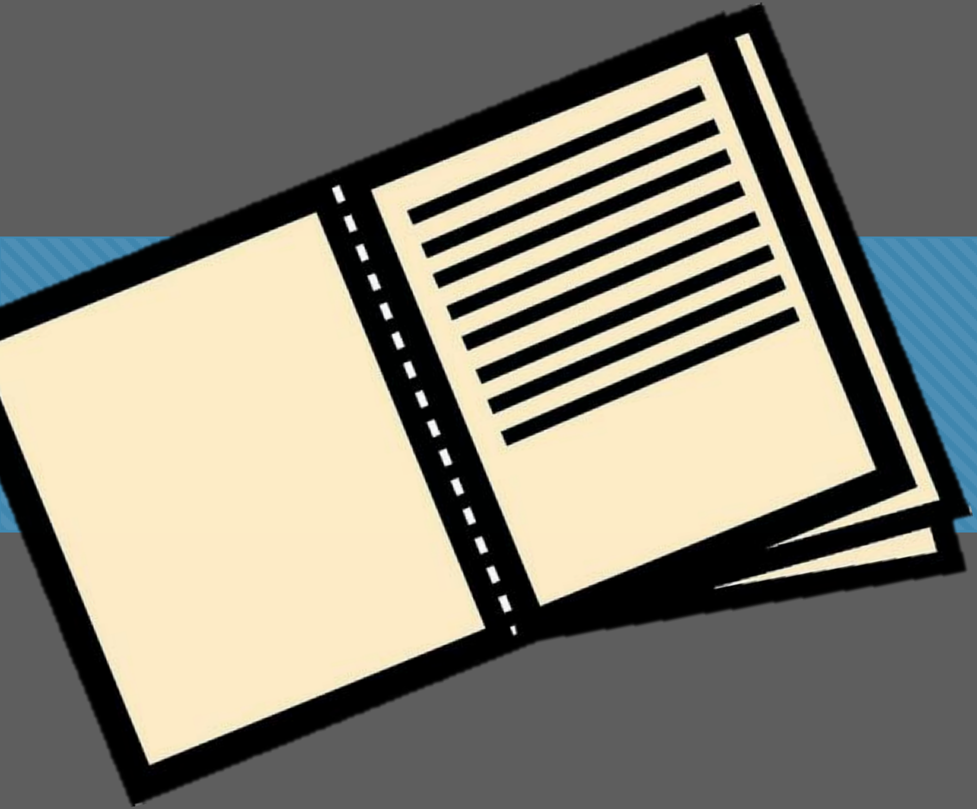


```
BlobCache.ApplicationName = "My Cool App";
await BlobCache.UserAccount.InsertObject("rdavisau", ryan);

// ...later, in another part of town...

var ryanFromCache = await BlobCache.UserAccount.GetObject<User>("rdavisau");
```

  - Has a GetAllObjects<T>() method
  - Facilitates caching and expiry: GetOrFetchLatest(string key, Func<Task<T>, DateTimeOffset expiry)
  - Specific methods for encrypted storage e.g. user creds



Resource roundup

people to watch, follow, reach out to etc.

(not comprehensive)

Subscribe to the Weekly Xamarin Newsletter!
http://www.weeklyxamarin.com

- Globals
  - **@paulcbetts** – Paul Betts, developer of ReactiveUI, Refit, ModernHttpClient, Akavache, Splat, Fusillade, etc.
  - **@jamesmontemagno** – James Montegamo, developer evangelist @ Xamarin, behind the Plugin for Xamarin push, lots of Android Material Design stuff, generally cool enthused dude
  - **@mikebluestein** – old timer, ex-Xamarin, big on CocosSharp and now Apple Watch
  - **@robgibbens** - Rob Gibbens, XU instructor, writes top blog posts @ http://arteksoftware.com/
  - **@saramgsilva** – Sara Silva, New Xamarin MVP, Azure focussed samples, etc.
  - **@shanselman** – Scott Hanselman, MSFT.. just because
  - **@xamarinhq** – of course.

Join the public Xamarin Slack room!
https://xamarinchat.herokuapp.com/

- Locals
  - **@glenntstephens** – Glenn Stephens, runs this here meetup, XU instructor, long time Monotouch
  - **@geoffreyhuntley** – Sydney, long timer? Active in the Xamarin/.NET OSS scene
  - **@kent_boogaart** – Kent Boogart, serious coder, under-marketed, active blog - http://kent-boogaart.com/
  - **@rid00z** – Michael Ridland, new Xamarin MVP

cross-platform / general

- MVVM
  - MvvmCross – https://github.com/MvvmCross/MvvmCross - Popular cross-platform MVVM library
  - ReactiveUI – https://github.com/reactiveui/ReactiveUI - MVVM library based on the Reactive Extensions and FRP principles
  - MVVM Light Toolkit – http://www.mvvmlight.net/ - Another MVVM library
- Helpful Libraries
  - Akvache - https://github.com/akavache/Akavache - An asynchronous, persistent key-value store
  - Splat - https://github.com/paulcbetts/splat - “A library to make things cross-platform that should be”
    (logging, service location, colours, images, geometry)
  - ModernHttpClient - https://github.com/paulcbetts/ModernHttpClient - HttpClient that uses platform-native HTTP clients for 🚀
  - Refit - https://github.com/paulcbetts/refit - The automatic type-safe REST library for Xamarin and .NET
  - Polly - https://github.com/michael-wolfenden/Polly - express transient exception handling policies such as Retry, Retry Forever, Wait and Retry or Circuit Breaker in a fluent manner
  - Lager - https://github.com/flagbug/Lager - cross-platform settings storage
  - Reactive Extensions - https://github.com/Reactive-Extensions/Rx.NET - a library for composing asynchronous and event-based programs using observable sequences and LINQ-style query operators.

cross-platform / general

(continued)

- Platform Abstractions
  - Xamarin Plugins - https://github.com/xamarin/plugins - catalogue of x-platform NuGet packages
  - XF-Labs - https://github.com/XLabs/Xamarin-Forms-Labs - XF-specific controls and a few platform abstractions
  - SimplyMobile - https://github.com/sami1971/SimplyMobile - Collection of abstracted mobile functionalities
  - Xamarin Component Store - https://components.xamarin.com/ - some paid

- On the 'Bait and Switch' / 'Advanced PCL' method
  - http://blogs.msdn.com/b/dsplaisted/archive/2012/08/27/how-to-make-portable-class-libraries-work-for-you.aspx
  - http://log.paulbetts.org/the-bait-and-switch-pcl-trick/
  - http://ericsink.com/entries/pcl_bait_and_switch.html
  - http://ryandavis.io/creating-and-publishing-a-cross-platform-library/ ← for excruciating detail on making one

xamarin.forms

- Good articles, series, books etc.
  - **Creating a Xamarin Forms App** [12 part series]
    http://adventuresinxamarinforms.com/2014/11/10/creating-a-xamarin-forms-app-part-1-introduction/
  - **Creating Mobile Apps with Xamarin.Forms (Preview)** [it's a book, free]
    http://developer.xamarin.com/guides/cross-platform/xamarin-forms/creating-mobile-apps-xamarin-forms/
  - **Xamarin Forms Kickstarter** [it's a book too, paid]
    http://www.xforms-kickstarter.com/
  - **Xamarin Forms in Anger** [still-ongoing series]
    https://www.syntaxismyui.com/xamarin-forms-in-anger/
  - **Xamarin Forms Recipes** [still-ongoing series?]
    https://www.syntaxismyui.com/category/xamarin-forms/recipe-xamarin-forms/
  - **Supercharging Xamarin Forms with Custom Renderers** [4-part series]
    http://www.wintellect.com/devcenter/jprosise/supercharging-xamarin-forms-with-custom-renderers-part-1
  - **Getting Started with Xamarin Forms Labs** [blog post]
    http://www.codenutz.com/getting-started-xamarin-forms-labs-xaml-mvvm-ioc/
- Xamarin's XF Developer Portal
  http://developer.xamarin.com/guides/cross-platform/xamarin-forms/

game dev

- CocosSharp and Monogame
  - **Xamarin's got game (development)** [introductory blog post]
    http://blog.xamarin.com/xamarins-got-game-development/
  - **Introduction to Game Development with Xamarin** [official Xamarin documentation]
    http://blog.xamarin.com/xamarins-got-game-development/
  - **Creating a basic game in CocosSharp**
    http://ryandavis.io/creating-a-basic-game-in-cocossharp-pt1/ [3 part series. it really is a basic game though]
    https://github.com/rdavisau/meetup-pop [code and slidepack]
  - **Intro to CocosSharp** [official Xamarin presentation]
    http://forums.xamarin.com/discussion/28879/cocossharp-intro-slides
- cocos2d
  - **Ray Wenderlich's cocos2d posts** [there are lots]
    http://www.raywenderlich.com/tag/cocos2d
  - **Learn cocos2d** [book, paid]
    http://www.apress.com/9781430244165

native platforms

- iOS
  - **Beginning iPhone Development** [book, paid, 40% off right now?] - http://www.apress.com/9781484202005
  - **Ray Wenderlich's site** - http://www.raywenderlich.com/
  - **iOS Dev Center** [apple's own] - https://developer.apple.com/devcenter/ios/
- Droid
  - **Pro Android, 5th Edition** [currently 'alpha', full release 20/05/2015] - http://www.apress.com/9781430246800
  - **Programming Mobile Apps for Android, Part 1** [Coursera, began 17/06] - https://www.coursera.org/course/androidpart1
  - **developer.android.com** [it's official] - http://developer.android.com/index.html
- Windows Phone
  - **Windows Phone 8.1 Dev for Absolute Beginners** [MVA] - http://bit.ly/VnpgCO
  - **Building Apps for Windows Phone 8.1 Jump Start** [MVA] - http://bit.ly/1poSSeE

+ approximately 999,999,999,999 courses on Pluralsight

training / misc

- Xamarin University - http://xamarin.com/university
- Pluralsight - http://www.pluralsight.com/
- Microsoft Virtual Academy - http://www.microsoftvirtualacademy.com/

- Xamarin Evolve conference videos
  - 2014: https://evolve.xamarin.com/
  - 2013: http://xamarin.com/evolve/2013

- Planet Xamarin – Xamarin Community Blogs
  - http://planet.xamarin.com/

Also check out PaintCode for iOS:
http://www.paintcodeapp.com/

questions / thanks