ESRI DEVELOPER SUMMIT

10-12 November | Berlin, Germany

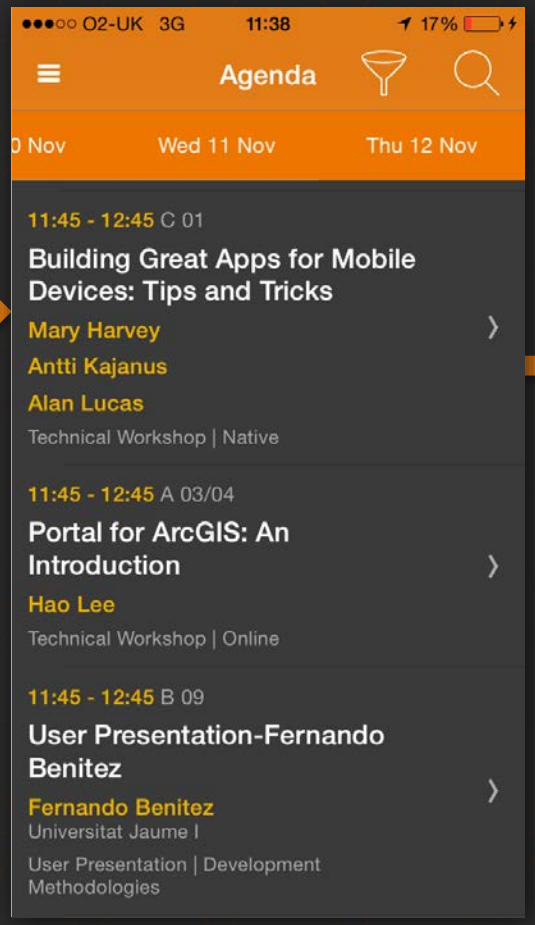
Preview of ArcGIS Runtime and Xamarin

Antti Kajanus, Stuart McIlreavy

Please Take Our Survey!

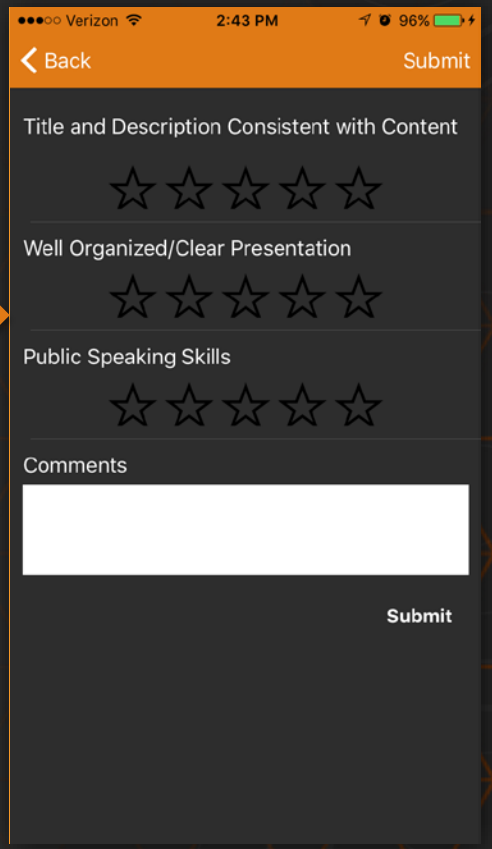
Download the Esri Events app and find your event

Select the session



Select “Technical Workshop Survey”

Complete Answers and Select “Submit”



Agenda

- Xamarin Overview
- Why Xamarin?
- ArcGIS Runtime and Xamarin
- Demos
- Why not Xamarin?
- Questions

Xamarin overview

Xamarin - overview

what can you do with it?

- Build .NET apps for iOS and Android
  - Write code using C# (or F#)
    - Share code across iOS, Android, and Windows
    - Leverage platform-specific capabilities as needed
  - Write UI markup with Xamarin Forms XAML
    - Share markup across iOS, Android, and Windows*
    - Different XAML stack than WinRT and WPF
  - Write UI specifically for iOS and Android
    - Use full platform specific UI stack
    - Requires knowledge about targeting platforms

** ArcGIS Runtime supports iOS and Android while Xamarin supports other targets too.*

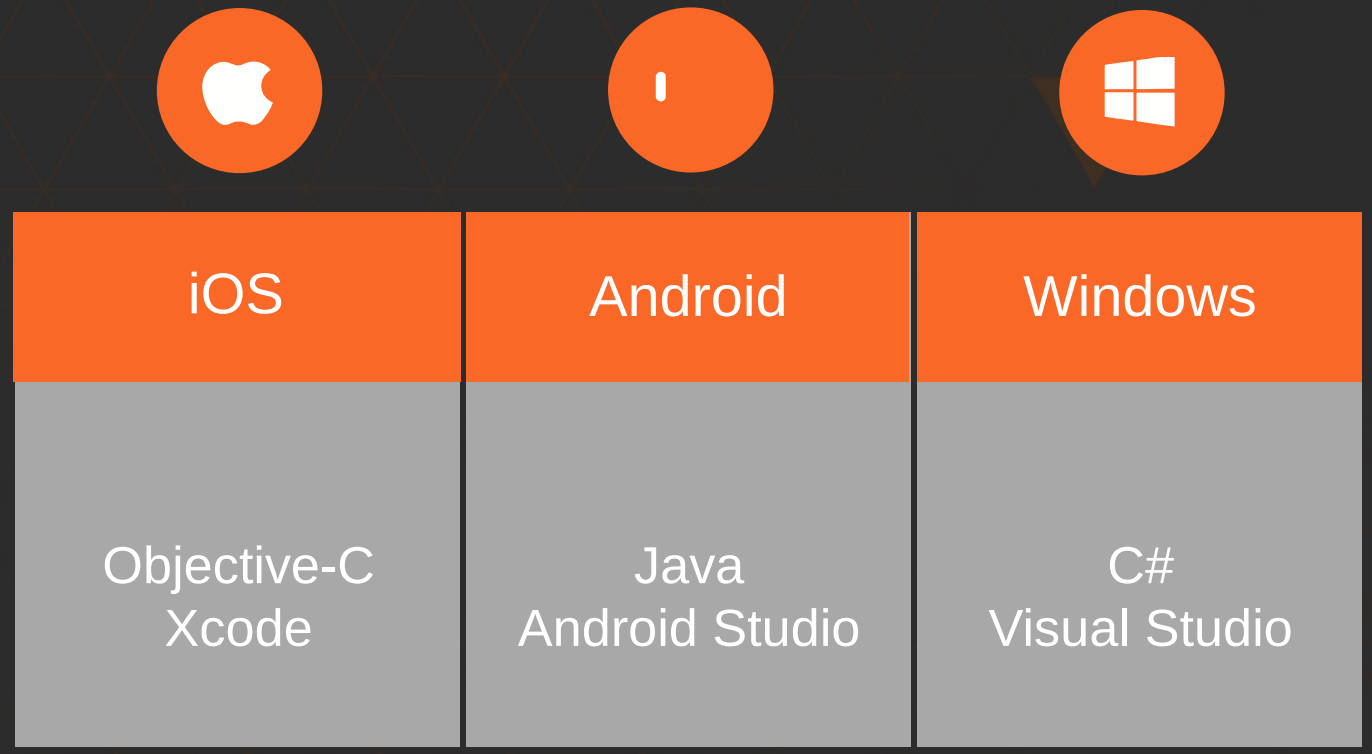
Xamarin - overview

But what is it?

- Software suite for cross-platform .NET development
  - Libraries for iOS and Android
    - Mono – open-source .NET implementation for non-Windows platforms
    - Platform bindings – surface OS APIs to C# (and F#)
    - Xamarin Forms – cross-platform UI library
  - Development tools for Windows and OS X
    - Visual Studio extension
    - Xamarin Studio
    - Xamarin Android Player

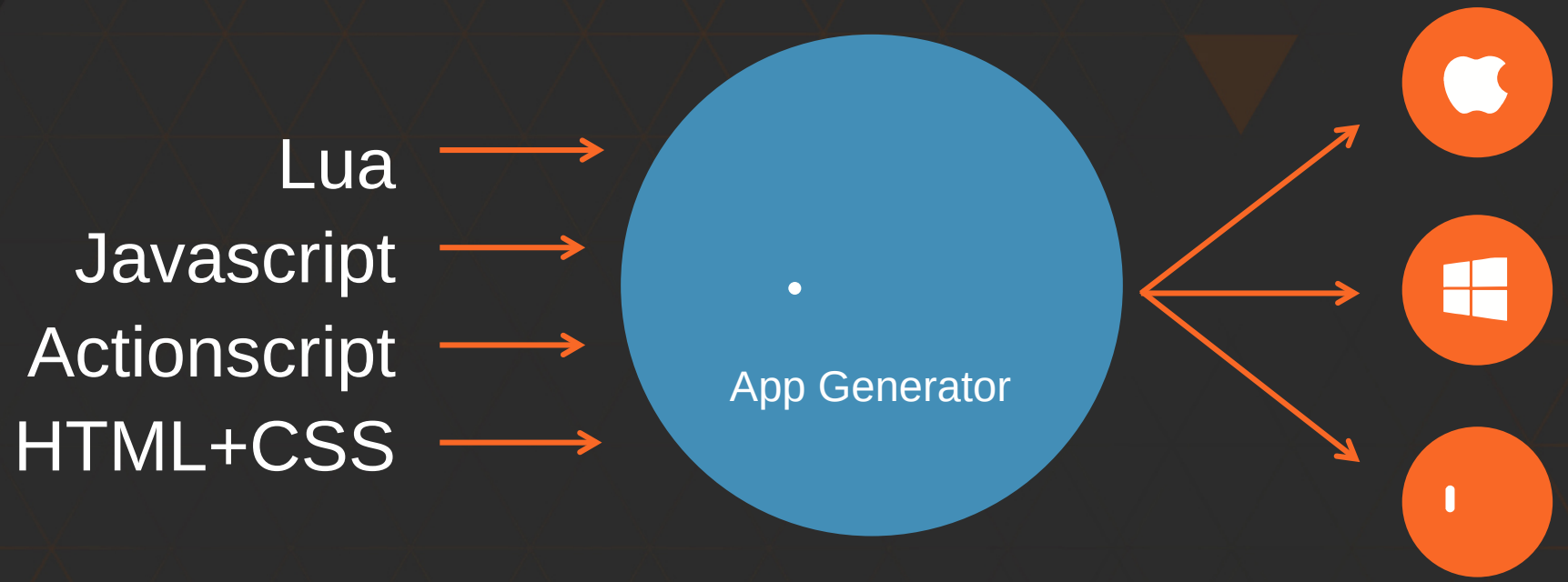
Why Xamarin?

Silos



- No shared code - Many languages & development environments - Multiple teams

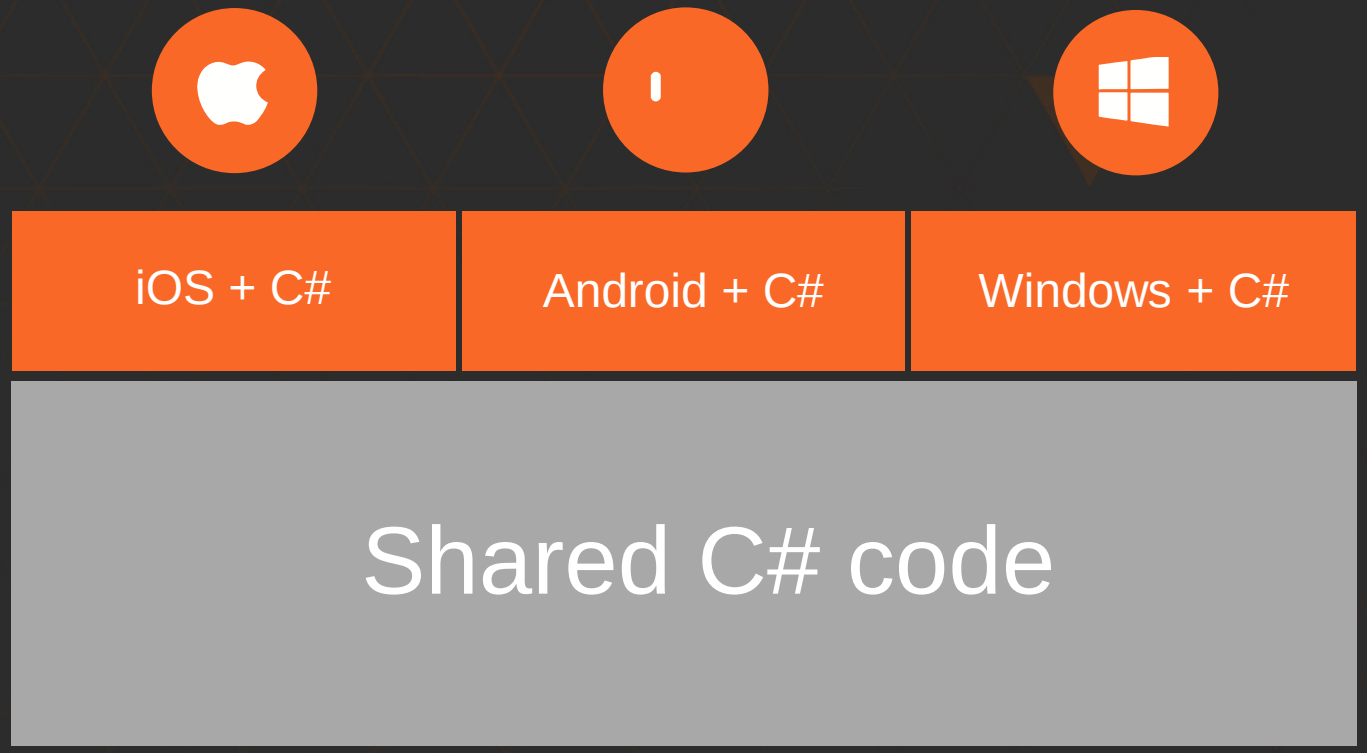
Application generators



Limited native API access - Slow performance - Poor user experience

Xamarin's approach

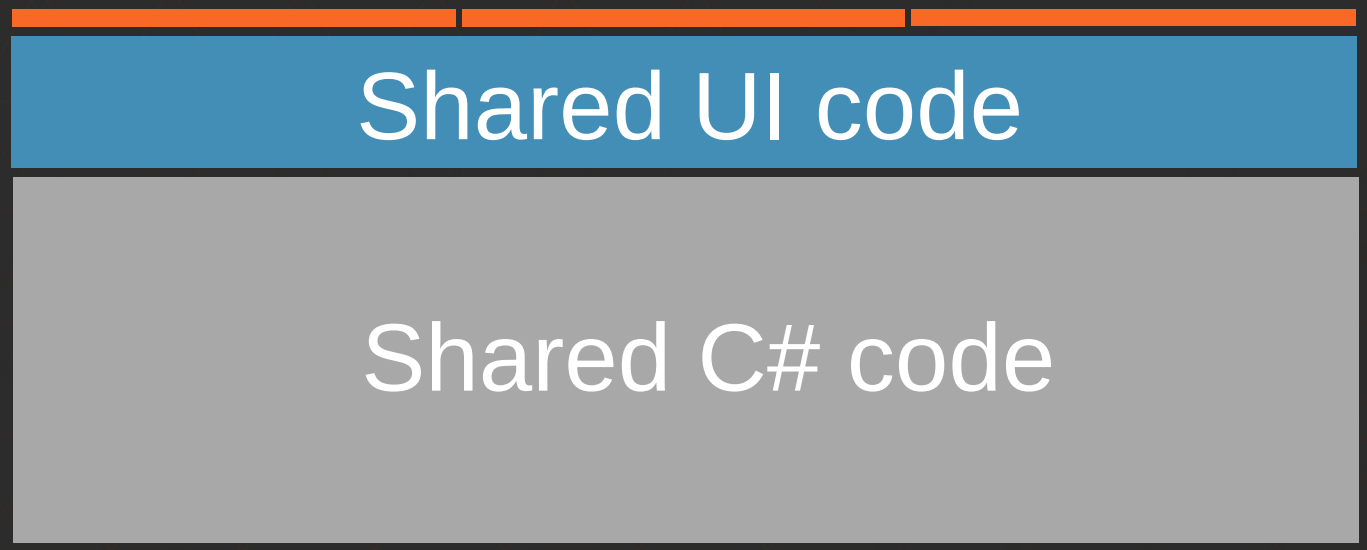
Xamarin.IOS / Xamarin.Android



Shared C# codebase - 100% native API access - High performance

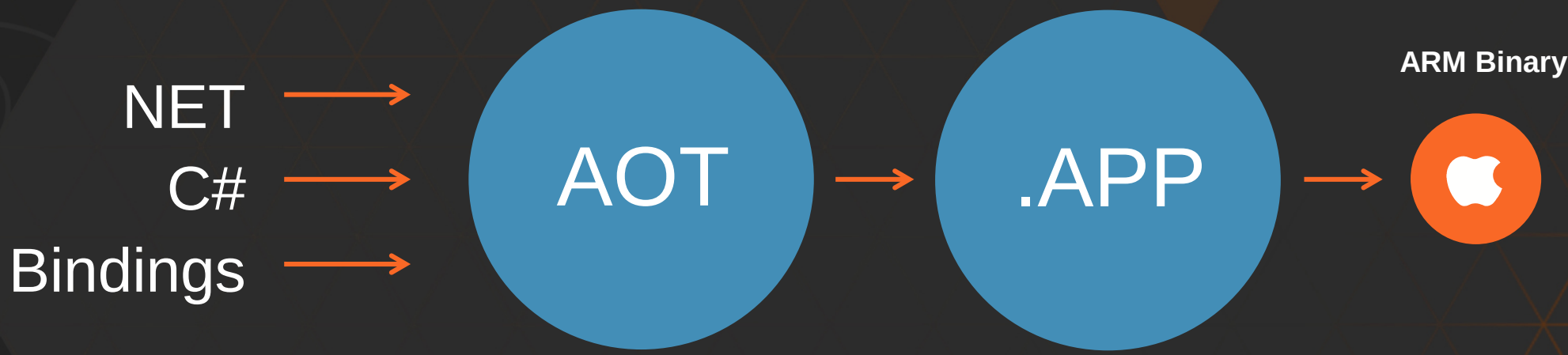
Xamarin's approach

Xamarin.Forms



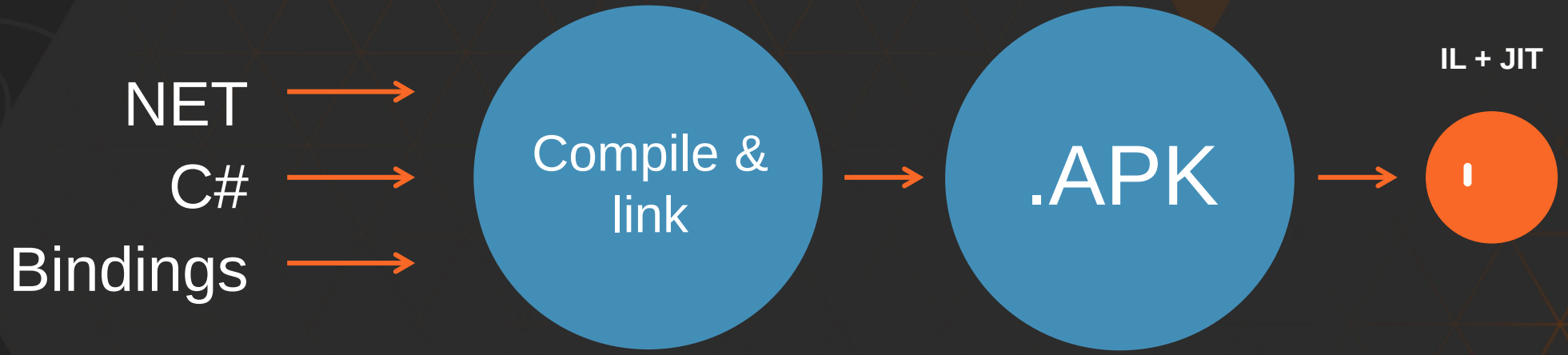
… and even more shared code!

It's native



It feels native because it is.

It's native



It feels native because it is.

Why Xamarin?

Advantages – code sharing and re-use

- Large amount of code is re-usable cross-platform
  - C# logic often portable
  - Language features identical cross-platform
  - Most BCL classes available cross-platform

- XAML situationally portable
  - Same markup features (binding, data context, etc)
  - Portability pairings – iOS / Android, Windows Store / Windows Phone, Desktop / Universal Windows Platform
  - ArcGIS Runtime XAML almost completely portable

Why Xamarin?

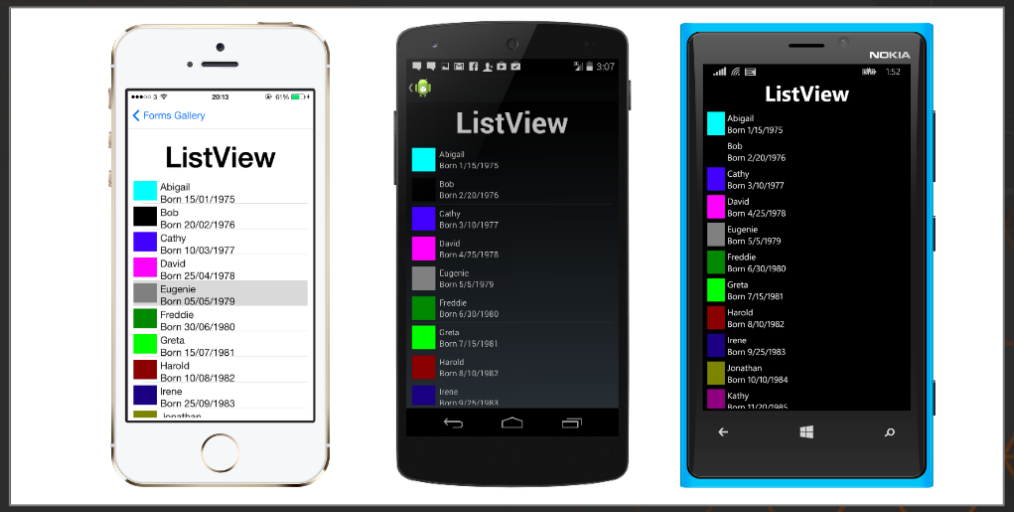
Advantages – full power of the platform(s)

- All platform capabilities available
  - Windows platform APIs available natively
  - iOS and Android platform APIs surfaced through bindings
    - Can get as platform-specific as needed
    - Can ostensibly do anything that's possible with the native APIs

Why Xamarin?

advantages – native rendering

- Windows components are native
- Android and iOS components exposed by Xamarin bindings are native
- Xamarin Forms components map to native components internally



Why Xamarin?

advantages – C#

- Managed
- Strongly typed with dynamic capabilities
- Richly featured

```
1 reference
private async Task LoadPortal()
{
    StatusMessage = "Initializing Portal...";
    var portal = await ArcGISPortal.CreateAsync();
    PortalInfo = portal.ArcGISPortalInfo;
    IsLoadingPortal = false;
    await LoadMaps(portal);
}
```

```
0 references
private double? getFirstX(IEnumerable<MapPoint> points)
{
    return points?.First()?.X;
}
```

```
Action<Map, Layer> mapAddLayer = (m, lyr) =>
    {
        if (!m.Layers.Contains(lyr))
            m.Layers.Add(lyr);
    };
```

Which one to pick?

Xamarin.Forms

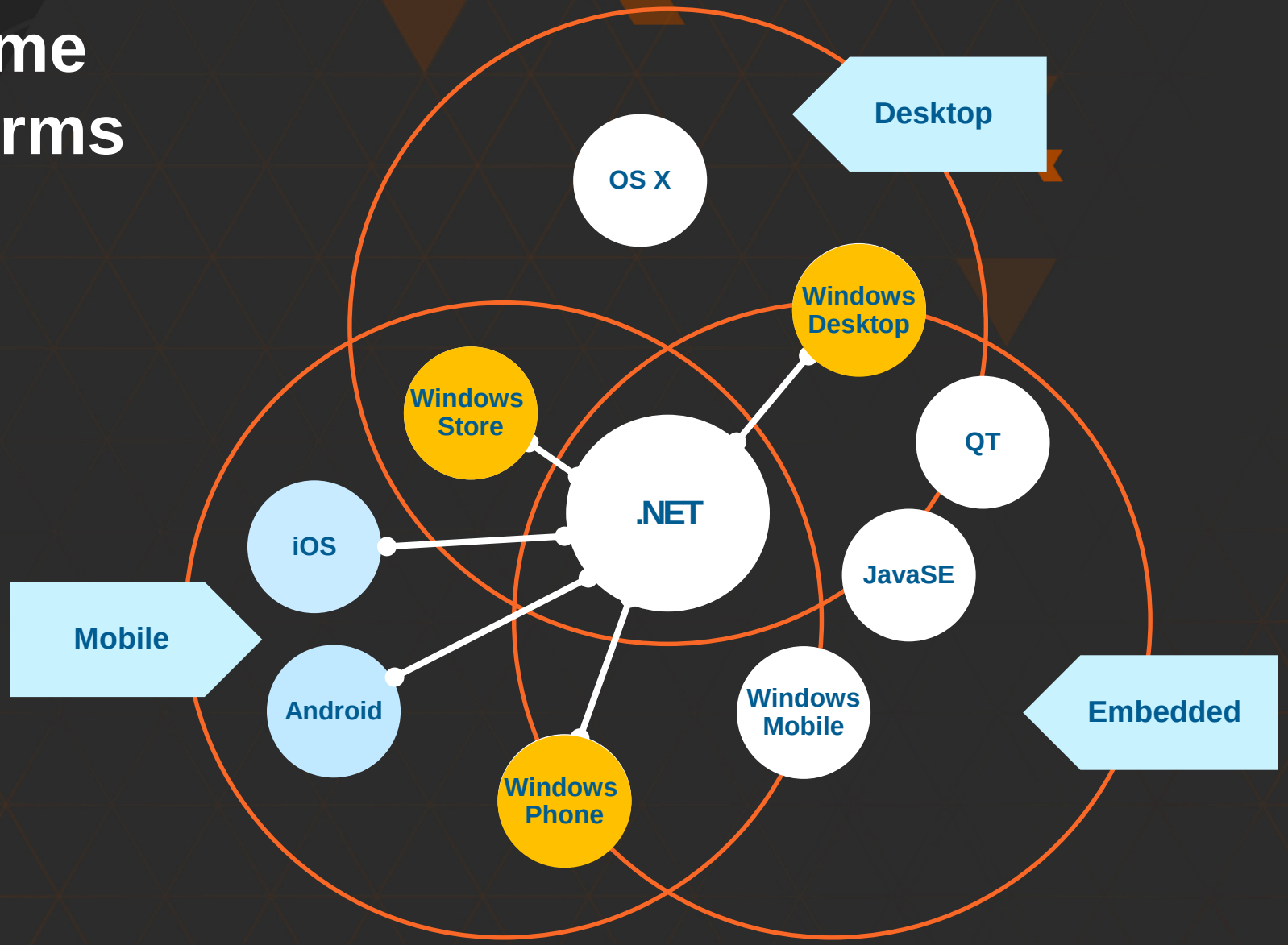
- Apps that require little platform-specific functionality
- Apps where code sharing is more important than custom UI
- Time until delivery

Xamarin.iOS / Xamarin.Android

- Apps that uses many platform-specific APIs
- Apps where custom UX is more important than code sharing
- Apps that require specialized interaction

Xamarin and ArcGIS

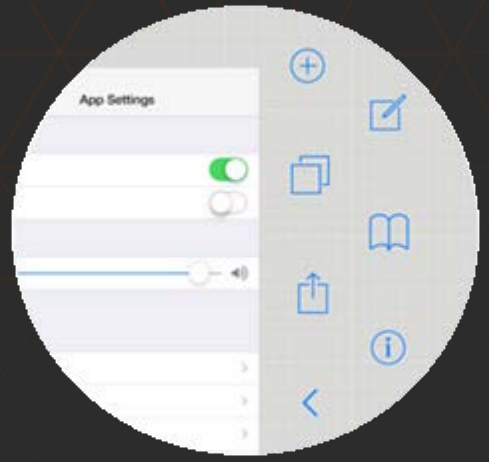
Runtime platforms



ArcGIS Runtime

Work Offline

Take your data offline to view, search, get directions, and edit and sync data



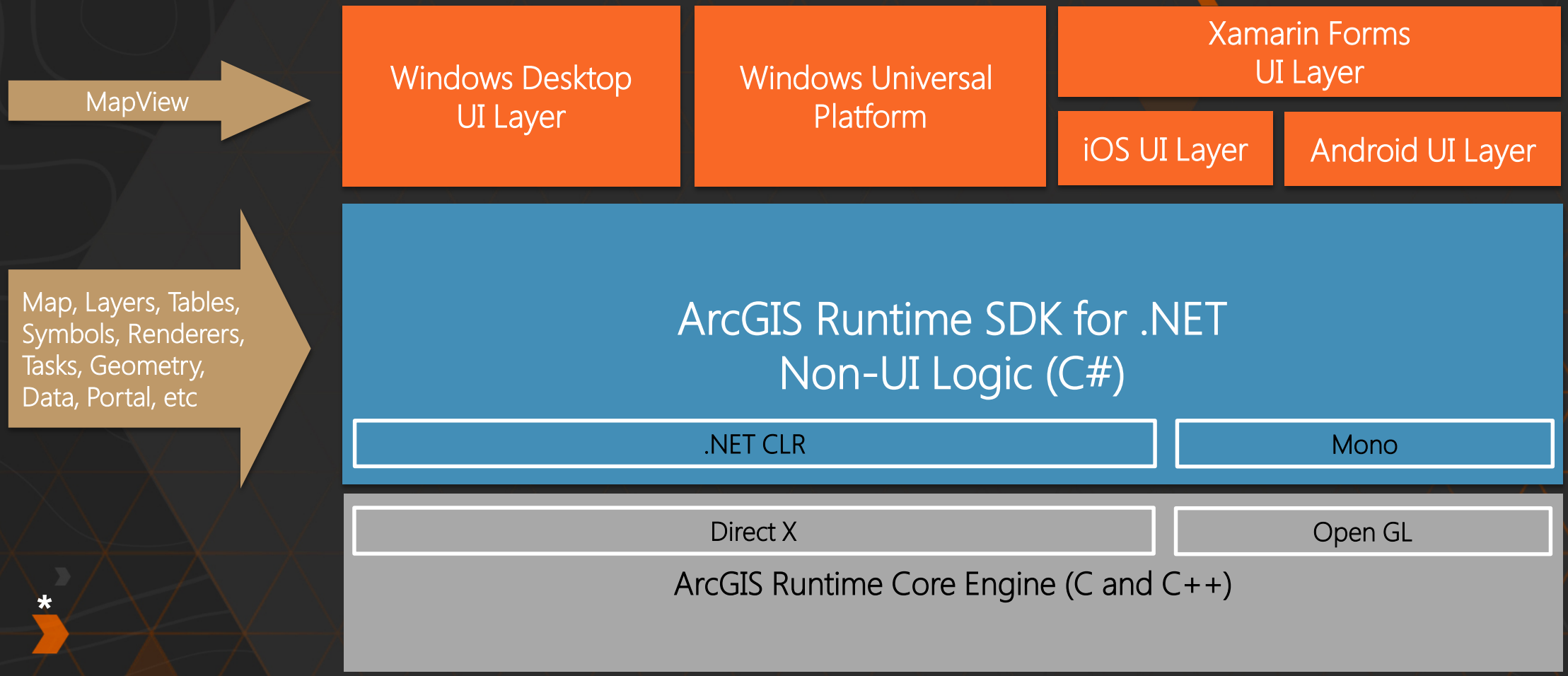
Native User Experience

Build apps that match the UI and UX conventions of their platforms.

Access Native APIs

Access native device capabilities like the GPS and compass and access user data like calendars and contacts

The ArcGIS Runtime for Xamarin/.NET

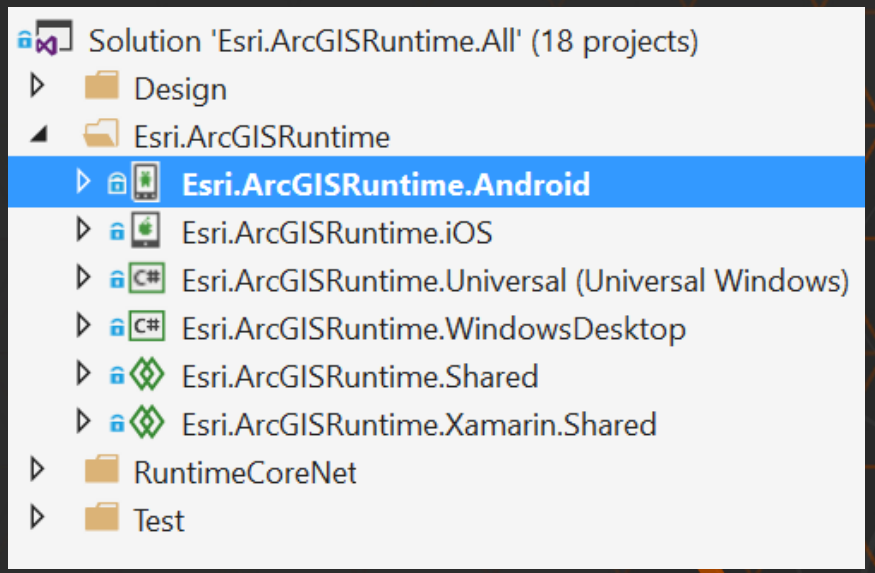


*

The ArcGIS Runtime for Xamarin

Why code-sharing works

- One Common API surface
  - Same API on Windows Desktop, Windows Universal Platform, iOS, and Android
  - Same underlying code, same functionality
  - Most code becomes shareable cross-platform
- Streamlined Development
  - Changes inherently apply to all platforms
  - All platforms remain in sync
- Tooling in Visual Studio
  - Shared projects



Demos

The ArcGIS Runtime(s) for Xamarin (and .NET)

Why not Xamarin?

Why not Xamarin?

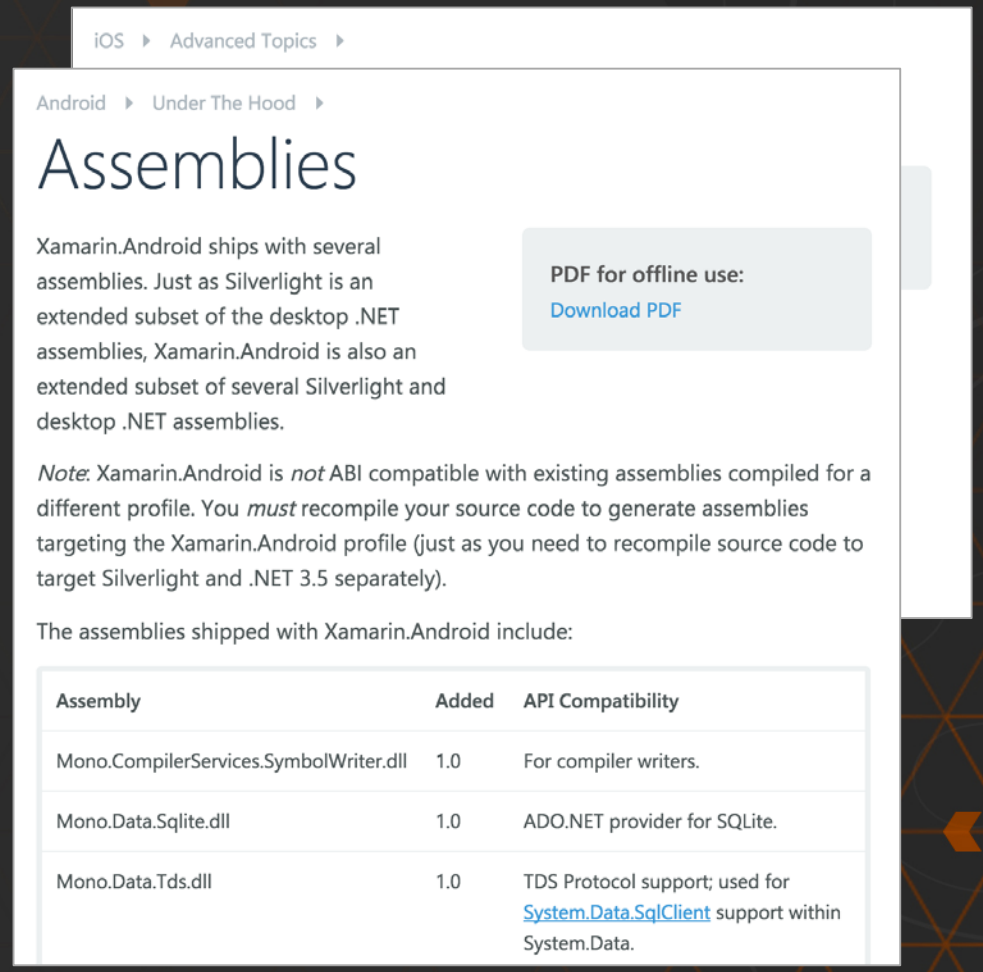
Challenges – can't touch this

- ArcGIS Runtime bits not available yet
  - Targeting to release public beta by Q1
  - Final release is targeting to be released next year
  - Sorry – you'll have to wait
- Xamarin supports
  - We support iOS and Android from them
  - You should use .NET directly for Windows Platforms

Why not Xamarin?

Challenges – multiple dev stacks

- .NET and Xamarin dev platforms are separate
  - Independent .NET Framework implementations
  - Xamarin only provides a subset of full .NET
  - Additional limitations within Xamarin's implementation
  - Xamarin Studio in addition to Visual Studio
  - Xamarin Visual Studio extension



iOS ▸ Advanced Topics ▸

Android ▸ Under The Hood ▸

Assemblies

Xamarin.Android ships with several assemblies. Just as Silverlight is an extended subset of the desktop .NET assemblies, Xamarin.Android is also an extended subset of several Silverlight and desktop .NET assemblies.

PDF for offline use:
Download PDF

Note: Xamarin.Android is *not* ABI compatible with existing assemblies compiled for a different profile. You *must* recompile your source code to generate assemblies targeting the Xamarin.Android profile (just as you need to recompile source code to target Silverlight and .NET 3.5 separately).

The assemblies shipped with Xamarin.Android include:

Assembly	Added	API Compatibility
Mono.CompilerServices.SymbolWriter.dll	1.0	For compiler writers.
Mono.Data.Sqlite.dll	1.0	ADO.NET provider for SQLite.
Mono.Data.Tds.dll	1.0	TDS Protocol support; used for System.Data.SqlClient support within System.Data.

Why not Xamarin?

Challenges – some code is not shareable

- Platform-specific code and markup sometimes necessary
  - Three different XAML "flavors"
  - BCL differences
    - E.g. file system
  - Non-BCL differences
    - E.g. gesture recognition

```
<Window xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:esri="http://schemas.esri.com/arcgis/runtime/2013"
```

```
<Page xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:esri="using:Esri.ArcGISRuntime.Controls"
```

```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             xmlns:esri="clr-namespace:Esri.ArcGISRuntime.Controls;assembly=Esri.ArcGISRuntime"
```

Why not Xamarin?

Challenges – two dev OSes needed

- Both OS X and Windows required for targeting all platforms
  - Windows for Windows Desktop, Store, Phone
  - Either Windows or OS X for Android
  - OS X (or Windows + OS X) for iOS
    - For build host and simulator

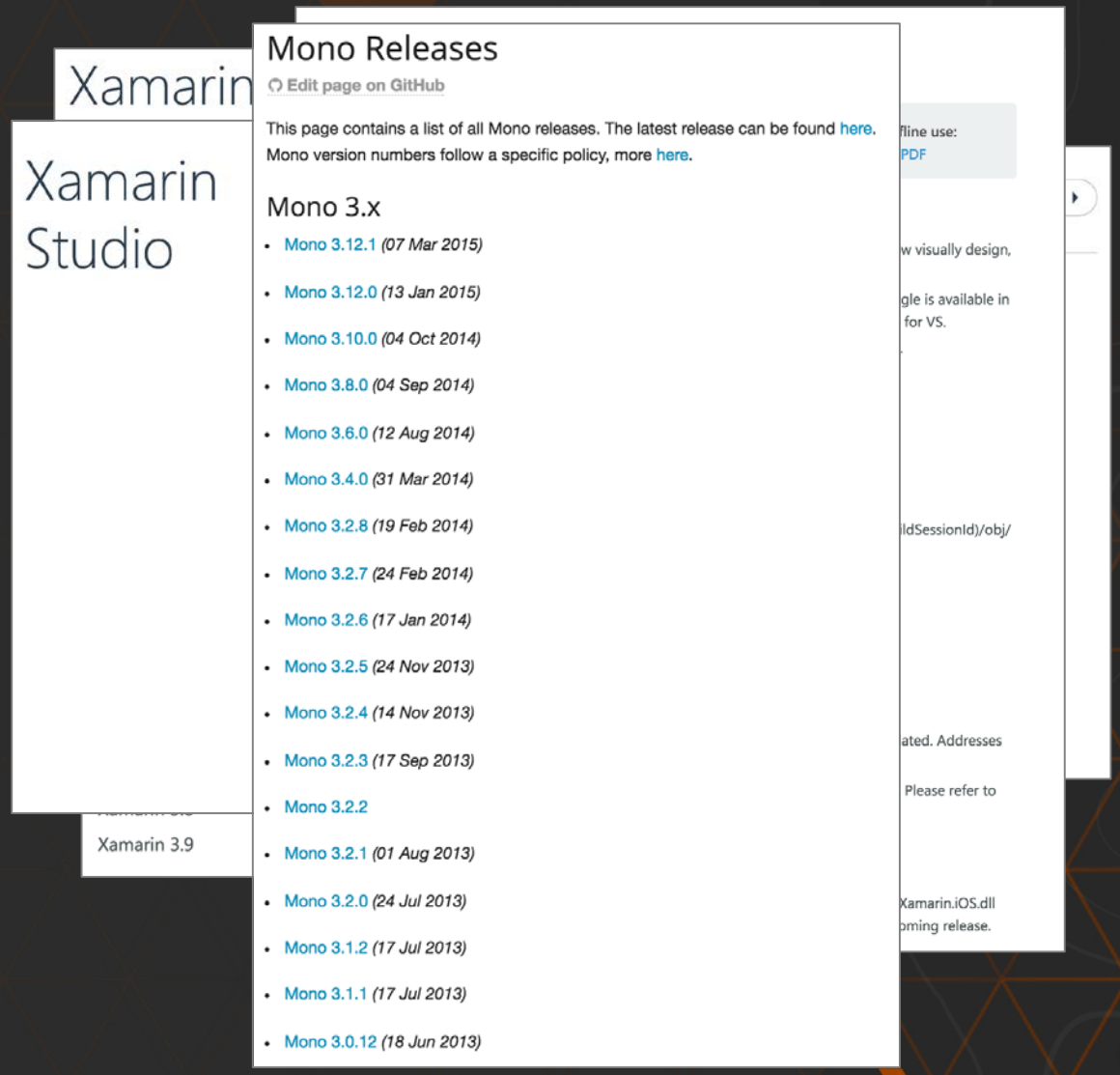
	Desktop	Store	Phone	iOS	Android
Windows	✓	✓	✓		✓
OS X				✓	✓

*

Why not Xamarin?

Challenges – tooling maturity

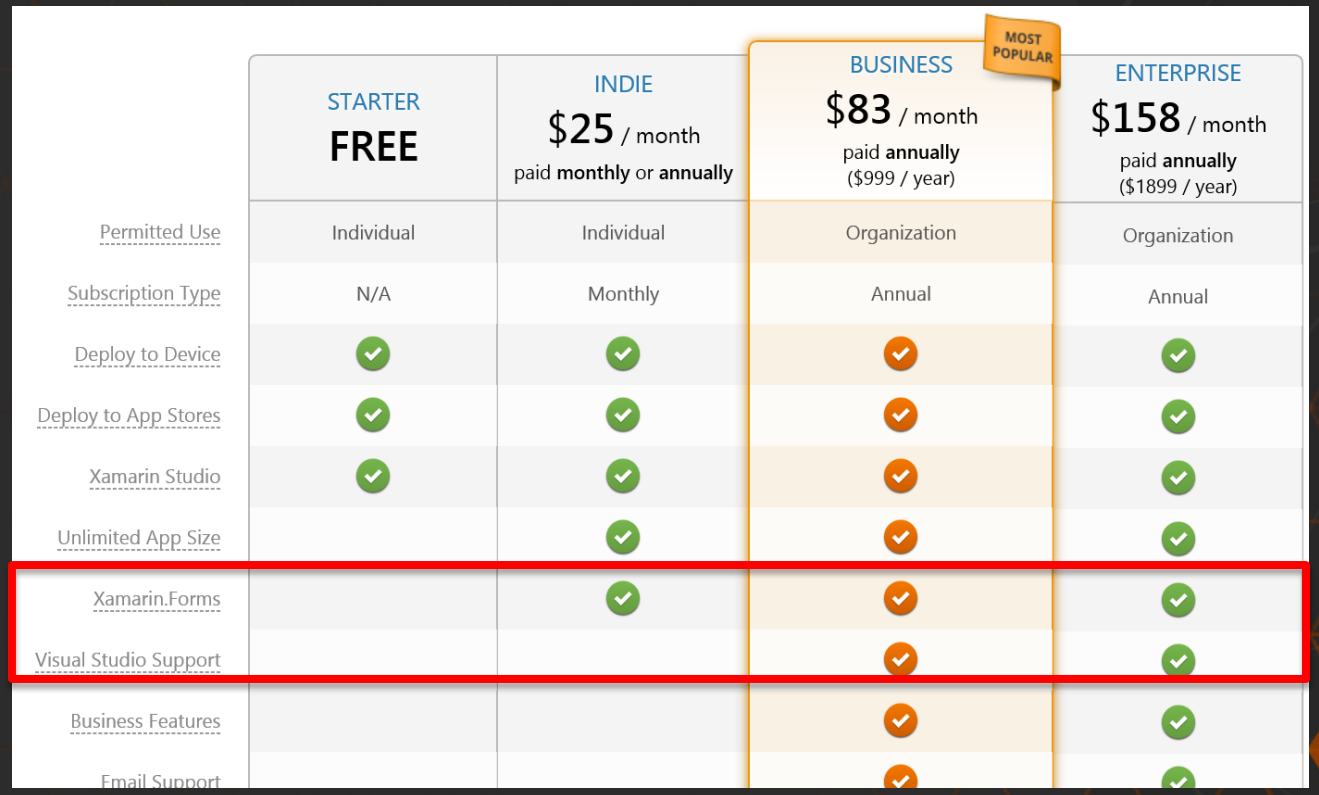
- Xamarin tooling still maturing
  - Unexpected behavior not uncommon
  - Dev-deploy-debug cycle can be much longer than on Windows platforms
  - Frequent updates



Why not Xamarin?

Challenges – tooling maturity

- Xamarin is not free of charge
  - Free version limits app size, excludes VS and Xamarin Forms
  - Free trial available
  - Pricing is subscription-based, available monthly or annually



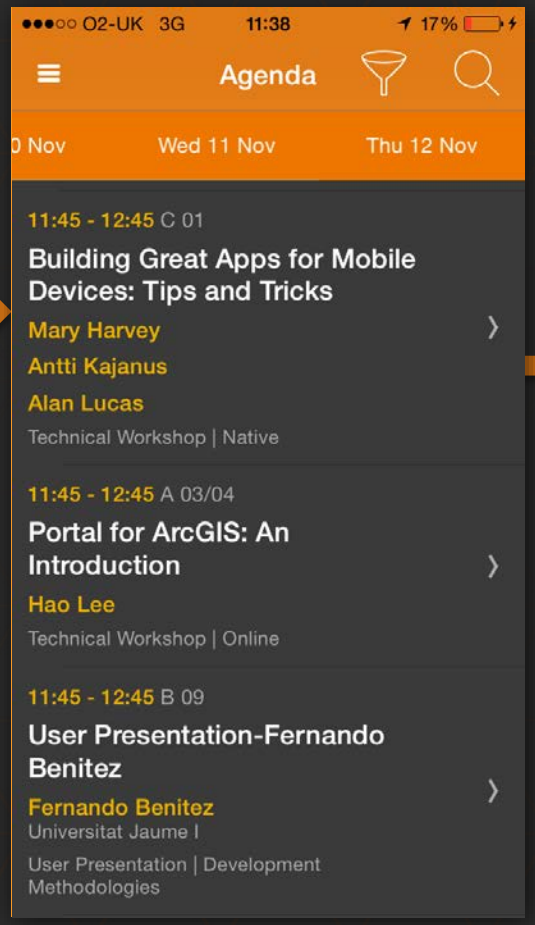
	STARTER **FREE**	INDIE $25 / month paid **monthly** or **annually**	BUSINESS $83 / month paid **annually** ($999 / year) — MOST POPULAR	ENTERPRISE $158 / month paid **annually** ($1899 / year)
Permitted Use	Individual	Individual	Organization	Organization
Subscription Type	N/A	Monthly	Annual	Annual
Deploy to Device	✓	✓	✓	✓
Deploy to App Stores	✓	✓	✓	✓
Xamarin Studio	✓	✓	✓	✓
Unlimited App Size		✓	✓	✓
Xamarin.Forms		✓	✓	✓
Visual Studio Support			✓	✓
Business Features			✓	✓
Email Support			✓	✓

SUMMARY

Please Take Our Survey!

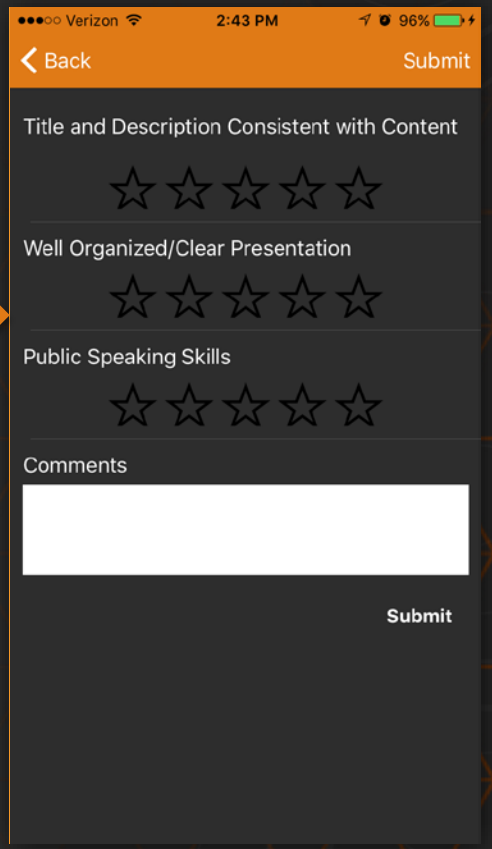
Download the Esri Events app and find your event

Select the session



Select “Technical Workshop Survey”

Complete Answers and Select “Submit”



QUESTIONS

Thank You to Our

Gold Sponsor


esri®