ArcGIS Runtime SDK for .NET
Building Windows Apps

Morten Nielsen
Mike Branscomb
ArcGIS Runtime SDK for .NET Parts I and II

• Part I
  - Building Windows Apps
• Part II
  - Building Xamarin Apps
ArcGIS Runtime session tracks at DevSummit 2018

- ArcGIS Runtime SDKs share a common core, architecture and design
- Functional sessions promote common capabilities and workflows
  - An Introduction to the API and Architecture
  - Working with Your Portal
  - Building 3D Applications
  - Network Analysis
  - Styling Maps
  - Analysis
  - Building Great User Experiences
  - Working with Maps Online and Offline
  - Editing Your Data Online and Offline
  - Integrating Imagery
- Product sessions promote platform-specific development experiences
- Demo theaters highlight examples of specific technical capabilities
Dev Summit 2018

• Watch "ArcGIS Runtime SDK for .NET: Building Apps"

• Topics covered:
  - Overview and history
  - Architecture
  - What is UWP?
  - What is Xamarin?
  - Getting Started
  - Installation / IDE integration
  - Building a simple app
  - Migrating from previous major release (10.2.x)
  - Road ahead for 2017…
Agenda

- New in 2017-2018
- Architecture of the SDK
- Toolkit
- Tips and Tricks
- Cross platform development
- Roadmap for 2018
New in 2017-2018
The road ahead…  

Dev Summit 2017

- Improved support for Virtual Machines w. support for Software Rendering
- Visual Studio 2017 integration
- Simpler Callout API
- Upcoming features across the runtimes:
  - Toolkit out of beta
  - Take an entire map offline
  - Improved Raster support
  - Full ArcGIS Server Dynamic Layer support
  - Improved Label support
  - Coordinate notation formatter
  - WMTS, WebTiledLayer, S57 Layers
  - Related tables and more…
What’s new

• v100.1 – June 2017
  - Raster on Android and iOS
  - Take a map offline
  - Related tables
  - Image Services
  - OpenStreetMap
  - Bing
  - WMTS
  - Camera controllers in 3D
  - Labeling
  - Service Area (online)
  - Closest Facility (online)
  - Coordinate Notation
  - Map Service DynamicLayer

• v100.2 – December 2017
  - Layers: ENC, WMS
  - Tables: Shapefile, Geopackage
  - Raster: Geopackage
  - Export Vector Tile Packages
  - Offline maps preplanned workflow
  - Dynamic feature layer rendering
  - Line of Sight
  - Viewshed
  - Service Area (local)
  - Closest Facility (local)
  - Statistics Queries
  - Transformations
  - Transactional editing
  - SketchEditor enhancements
  - Time

• v100.2.1 – February 2018
  - Bug fixes
  - Raster datasets and tile packages in mobile map packages
  - SceneView WGS84 basemaps
  - WMS versions <1.3

• Patch – February 2018
  - ArcGIS Runtime Local Geoprocessing Service Startup Patch
    - Windows 7 SP1
    - Windows Server 2008 R2
  - Patches:
    - Local Server SDK v100.0 – v100.2
    - .NET SDK v10.2.7
    - Qt SDK 10.2.6
    - Java SDK 10.2.4
    - WPF SDK v10.2.5
Architecture
ArcGIS Runtime SDK for .NET

Architecture Diagram

Public SDK

- WPF
- UWP
- iOS
- Android

Internal

- ArcGIS Runtime SDK for .NET (C#)
  - .NET Framework
  - .NET Core
  - Mono / Xamarin

OS

- Win32
- Windows Runtime
- iOS
- Android

Interop

ArcGIS Runtime Core (C++)

Common UI

- Xamarin Forms

Common SDK

- Maps, scenes, layers, symbols, feature data, geocoding, routing, portal, rasters, offline(syncing), geometry, geometry engine...

Where all the magic happens
ArcGIS Runtime SDK for .NET

A simpler view…

- WPF
- UWP
- iOS
- Android

<table>
<thead>
<tr>
<th>WPF</th>
<th>Xamarin Forms</th>
<th>.NET</th>
</tr>
</thead>
</table>

• Native UI
  • MapView, SceneView

• Common SDK
  • Maps, scenes, layers, symbols, feature data, geocoding, routing, portal, rasters, offline/syncing, geometry, geometry engine…

Operating System
Demo: Toolkit
Toolkit

- https://github.com/Esri/arcgis-toolkit-dotnet
- Early beta of toolkit v100.x on Nuget.org
- AppVeyor build for Pull Requests and master
- Code analyzers enforce coding guidelines
- Beta update planned for Q1 2018
  - Compass
  - Legend
  - ScaleLine
  - TimeSlider

Roadmap…

- Popup UI
- Table of contents
- Template picker / editing toolbar
- Measure toolbar
Tips and Tricks
Tips and Tricks

• New samples
• Animating graphics and features
• Renderers
• Labeling
• Running Python scripts
New samples
New Samples
Animating graphics and features
Frame-by-frame animation

16ms
Frame-by-frame animation

Frame-based event classes:

- WPF and UWP: CompositionTarget.Rendering
- Android: Android.Animation.TimeAnimator
- iOS: CoreAnimation.CADisplayLink

Full source-code in “GeoEvent Server Streaming Sample”

https://github.com/Esri/arcgis-runtime-demos-dotnet
Renderers
Renderers
The hidden secrets...

```json
{
  "renderer": {
    "type": "heatmap",
    "blurRadius": 10,
    "colorStops": [
      { "ratio": 0, "color": [0,0,0,0] },
      { "ratio": 0.01, "color": [133,193,200,0] },
      { "ratio": 1, "color": [255,255,200,0] },
    ],
    "maxPixelIntensity": 1000,
    "minPixelIntensity": 0
  }
}
```

Renderer myRenderer = Renderer.FromJson(json);

https://developers.arcgis.com/web-map-specification/objects/heatmapRenderer/
https://developers.arcgis.com/web-map-specification/objects/visualVariable/
Labeling
Labeling

https://developers.arcgis.com/web-map-specification/objects/labelingInfo/

https://developers.arcgis.com/arcade/
Python Scripts
Python

• ArcGIS Runtime Local Server

```python
# Import necessary modules
import arcpy
import os

# Set variables from input parameters.
workspace = arcpy.GetParameterAsText(0)

# Set the current workspace
arcpy.env.workspace = str(workspace)

# Create the output table.
outTableFc = arcpy.CreateTable_management('in_memory', 'datasets')

# Set the field names and definitions for the output table.
arcpy.AddField_management(outTableFc, 'NAME', 'TEXT', '', '', 260)
arcpy.AddField_management(outTableFc, 'DATATYPE', 'TEXT', '', '', 260)

# Create an insert cursor to insert rows into the output table.
fields = ['NAME', 'DATATYPE']
cur = arcpy.da.InsertCursor(outTableFc, fields)

# Feature
datasets = arcpy.ListFeatureClasses()

# For each dataset insert the name into a new row in the table.
for dataset in datasets:
    cur.insertRow((dataset, 'Feature Class'))

# Mosaic
datasets = arcpy.ListDatasets(feature_type='mosaic')
```
Cross Platform Dev
ArcGIS Runtime SDK for .NET

A simpler view...

- **WPF**
- **UWP**
- **iOS**
- **Android**

**Native UI**
- *MapView, SceneView*

**Common SDK**
- *Maps, scenes, layers, symbols, feature data, geocoding, routing, portal, rasters, offline/syncing, geometry, geometry engine...*
Demo: Cross Platform
.NET Standard
Resources
Resources

- Samples  https://github.com/Esri/arcgis-runtime-samples-dotnet
- DevLabs  https://developers.arcgis.com/labs/
- Example apps  https://developers.arcgis.com/example-apps/
- Demos  https://github.com/Esri/arcgis-runtime-demos-dotnet
- Toolkit  https://github.com/Esri/arcgis-toolkit-dotnet
- Geonet  https://community.esri.com/ > ArcGIS Runtime SDK for .NET
Minimum system requirements at next release

• WPF:
  - .NET Framework version 4.6.1
• Visual Studio:
  - Visual Studio 2015 (WPF, Xamarin.Android, and Xamarin.iOS)
  - Visual Studio 2017 (UWP)
• Xamarin.Android
  - API level 19 (Android 4.4)
• Xamarin.iOS
  - Development for iOS 10
• UWP:
  - 1709 (OS build 16299) / ‘Fall Creators Update’