Developing Mobile Apps with the ArcGIS Runtime SDK for .NET

Rich Zwaap
Morten Nielsen
Agenda

• The ArcGIS Runtime
• Getting started with .NET
• Mapping
• Editing
• Going offline
• Geocoding and routing
• Using Native Device Capabilities
The ArcGIS Runtime
ArcGIS Runtime

Runtime built using C++
EXPLOITS THE CAPABILITIES OF THE DEVICE

Functionality exposed to developers via an API
native to the platform
INTUITIVE TO LEARN

Common functionality set and conceptual model
EASES MULTI PLATFORM DEVELOPMENT
Device Platforms

PHONE  TABLET  LAPTOP  DESKTOP  EMBEDDED
Runtime Platforms

- iOS
- Android
- Windows Phone
- Windows Desktop
- Windows Store
- JavaSE
- QT
- .NET
- Desktop
- Embedded
Runtime Architecture

- Platform API wraps C++ core
- You write code in the language of the platform (C#, VB, Java, Objective-C, etc)
- No need to be concerned with details of Core
Runtime SDK for .NET

- APIs for three platforms
- Desktop, Store (tablet), Phone
- One common API surface

ArcGIS Runtime API

C++ ‘Runtime Core’

Windows Desktop API
Windows Store app API
Windows Phone API
A Native App

- Works connected and disconnected from the internet
- Exploits device capabilities
- Works with sensors connected to the device
Devices devices devices...
Getting started with the .NET Runtime

where do I get started?
ArcGIS Runtime SDK

- Conceptual doc, API reference, samples, and the developer community
  - Start here: [http://developers.arcgis.com/net](http://developers.arcgis.com/net)
  - Beta community: [https://betacommunity.esri.com](https://betacommunity.esri.com)
- GitHub: Samples, Toolkit, Offline app, Portal Viewer app
ArcGIS for Developers
http://developers.arcgis.com

- SDK Content – Documentation, API Reference, Samples
- Register for a developer account (free)
  - Register applications
    - ClientID for licensing Runtime apps at Basic level
  - Create cloud-hosted services
  - Receive 50 credits per month for development and testing
  - Download ArcGIS Runtime SDKs
- Can also log in with an ArcGIS Online account
ArcGIS Runtime SDK for .NET Developer’s Guide
http://developers.arcgis.com/net/desktop/guide

• Getting started
  - System requirements
  - Installation
• Tutorials
• High-level concepts
• Platform-specific content
  - Desktop
  - Store
  - Phone
• Sign up for beta at https://betacommunity.esri.com/
Developer Resources

developers.arcgis.com
Mapping

I can haz map in my appz?
Adding a map

- **Map** class
  - Defines layers to be shown
  - Also specifies initial extent and spatial reference
  - Data object – not rendered directly
Adding a map

- Different layer types for different data sources
  - **ArcGISTiledMapServiceLayer** - tiled ArcGIS for Server map services
  - **ArcGISDynamicMapServiceLayer** - dynamic ArcGIS for Server map services
  - **ArcGISImageServiceLayer** - ArcGIS for Server image services
  - **FeatureLayer** - layers within map services and geodatabases
  - **ArcGISLocalTiledLayer** - tile packages
  - More classes for non-ArcGIS data sources (Bing, OpenStreetMap, GeoRSS, CSV, etc)
Adding a map

- **MapView** class
  - Control that defines appearance of map within application (position on page, width, height, etc)
  - Takes **Map** as a property
  - Also specifies:
    - Display of user location (GPS)
    - Editing behavior via the **Editor** class
    - Map navigation methods and events

```xml
<esri:MapView ExtentChanged="MapView_ExtentChanged"
<esri:MapView.LocationDisplay>
<location:LocationDisplay IsEnabled="True" AutoPanMode="Navigation" />
</esri:MapView.LocationDisplay>
<esri:MapView.Editor>
<esri:Editor />
</esri:MapView.Editor>
<esri:Map>
<layers:ArcGISTiledMapServiceLayer ServiceUrl="http://services.arcgisonline.com/arcgis/rest/services/World_Topographic/MapServer" />
</esri:Map>
</esri:MapView>
```
Mapping
let’s try it...
Editing
and MVVM, and location display, and...
Going Offline
pulling the plug
Taking Data Offline – Tiled Map Services

- Use `ExportTileCacheTask` to take tiled map services offline
  - `GenerateTileCacheAsync` - creates a tile package (.tpk) or compact cache
  - `DownloadTileCacheAsync` - downloads a .tpk or compact cache
  - `GenerateTileCacheAndDownloadAsync` - does both in one call

```csharp
var task = new ExportTileCacheTask(new Uri(onlineTiledLayer.ServiceUri));
var downloadResult = await task.GenerateTileCacheAndDownloadAsync(
    generateOptions,
    downloadOptions,
    TimeSpan.FromSeconds(3),
    CancellationToken.None,
    onGenerateUpdate,
    onDownloadUpdate);
```

Developing Mobile Apps with the ArcGIS Runtime SDK for .NET
Taking Data Offline – Feature Services

- **GeodatabaseSyncTask** - take feature services offline
  - `GenerateGeodatabaseAsync` - creates a geodatabase
  - Use `ArcGISHttpClient` to download the result

```csharp
GeodatabaseSyncTask gdbTask = new GeodatabaseSyncTask(new Uri(featureServiceUrl));
var result = await gdbTask.GenerateGeodatabaseAsync(
    gdbParameters,
    // Parameters for the operation
    onGenerateCompleted,
    // Callback to handle operation completion
    TimeSpan.FromSeconds(3),
    // Interval to check status
    onGenerateProgress,
    // Callback to handle operation status updates
    CancellationToken.None);
    // Token to handle cancellation
```
Going Offline
let’s pull the plug…
Sync – Reconnecting with the Data Source

- **GeodatabaseSyncTask.SyncGeodatabaseAsync**
  - Push updates from the client and download changes from the service
  - Only changes (deltas) are downloaded/uploaded

```csharp
var gdbTask = new GeodatabaseSyncTask(new Uri(featureServiceUrl));
var result = await gdbTask.SyncGeodatabaseAsync(
    syncParameters,
    featureLayer.FeatureTable.Geodatabase, // Operation parameters
    onSyncCompleted,
    onUploadCompleted,
    TimeSpan.FromSeconds(3), // Geodatabase to sync
    onStatusUpdate,
    CancellationToken.None); // Callback for completion of sync operation
    // Callback for changes being uploaded to server
    // Frequency of sync status checks
    // Callback for status updates
    // Token for handling cancellation
```
Sync
plugging back in
Fully Disconnected Workflows

- Create Runtime Content from ArcMap and include in application
- `ArcGISLocalTiledLayer` - tile packages
- `FeatureLayer` - geodatabases and shapefiles (coming soon)

```xml
<esri:MapView>
  <esri:Map>
    <layers:ArcGISLocalTiledLayer Path="ms-appx://Data/Tiles/USA.tpk" />
  </esri:Map>
</esri:MapView>
```

```csharp
var gdb = await Geodatabase.OpenAsync(@"Data\Features\SalesData.gdb"); // 1.) Open the geodatabase
var table = gdb.FeatureTables.Where(t => t.Name == "Stores").First(); // 2.) Find the stores table
var featureLayer = new FeatureLayer() { FeatureTable = table }; // 3.) Create a layer to display the table
map.Layers.Add(featureLayer); // 4.) Add the layer to the map
```
Disconnected Workflows: Using Runtime Content
Routing & Geocoding
Native Device Capabilities

extending your device with your devices
What’s coming…
Future Releases

- Raster layers – create from local raster data (GDAL)
- New vector layers – create from local vector data (geopackage, shapefile, kml)
- Analysis functions - add to your layers
- 3D viewing
- More offline capabilities
  - Feature service table – equivalence with ArcGISFeatureLayer
  - Versioned data support
- Better support for authoring/reading maps

- @ Next release: Some APIs will release items before others…
- @ Next, Next release: APIs will be back in sync
UC Runtime Sessions
ArcGIS Runtime SDK Sessions - Thursday

<table>
<thead>
<tr>
<th>Session Name</th>
<th>Time</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArcGIS Runtime SDKs: The Road Ahead</td>
<td>1:30pm – 2:45pm</td>
<td>Room 07 A/B</td>
</tr>
</tbody>
</table>
Thank you...

- Please fill out the session survey:

  Offering ID: 1665

  Online - www.esri.com/ucsessionsurveys
  Paper - pick up and put in drop box