Developing Offline Apps with ArcGIS Runtime SDKs

Euan Cameron
Justin Colville
Will Crick
Offline App Capabilities
Disconnected Use of the ArcGIS Platform

- Viewing and Interacting with Maps
- Querying Data
- Editing Features
  - Synchronization
- Find places and locations
- Get directions
- Pre-planned or On-demand Workflows
- Occasionally Connected Scenarios
- All Runtime SDKs*
ArcGIS Runtime Architecture

Application

API

Core

Smartphones

Tablets

Desktops

ArcGIS

Local Server
ArcGIS Runtime Architecture

Application

API

Core

Smartphones

Tablets

Desktops

ArcGIS

Local Server
Offline Capabilities
Maps and Data
The ArcGIS Platform – Working Offline

- Offline Apps Need Maps and Data!
- All Compute Platforms Support Going Offline
  - Online and Portal
  - Server
  - Desktop
- New Workflows
  - Critical for You to Understand These Workflows
From ArcGIS Online and Portal

Author

Synchronize

Prepare

Download

View
Query
Analyze
Edit

Prepare

Download

View
Query
Analyze
Edit
ArcGIS Online Data Licensing

- Always Review Data Usage Restrictions
- Supported Esri Services
  - Basemaps
- Esri Services Not Supported Offline
  - Geocoding
  - Routing
Offline With ArcGIS Server

Diagram showing the process of publishing a service, downloading the service, and synchronizing it with a mobile device for offline use. The diagram includes steps for authoring maps, publishing services, and viewing, querying, analyzing, and editing data offline.
Offline With Desktop

Developing Offline Apps with ArcGIS Runtime SDKs
Data Preparation
Mixing and Matching

• Possible to Take a Hybrid Approach
  - Online basemaps
  - ArcGIS Server feature services
  - Geocoding and routing from Desktop

• The Hybrid Approach Will Be Common in Your Solutions
  - Optimize content delivery to match solution
Optimizing The Data Provisioning Process

• Features and Basemap Content
  - On Demand – Retrieves data from the server when requested
    - Uses server processing power and bandwidth for every client request
  - Side-loading - Content created ahead of time and provisioned onto the device
    - Can help alleviate server load at busy times of the day
    - Commonly used for basemaps
    - Can work with feature data if the workflow is right
      - How often and how many features change on the backend?

• Networks and Locators
  - Side-loading is the only option
APIs for Building Offline Apps
Working With Basemaps Offline

• ArcGIS Desktop Can Create Tile Caches

• Esri Basemaps
  - http://tiledbasemaps.arcgis.com/
  - AGOL basemap group

• ArcGIS Server Tiled Map Services

• Services Limit Extent
  - Administrator Sets the Limit
  - Workable Size
    - Compression quality
  - Estimations

• Consider Reusing Offline Basemaps
Taking ArcGIS Online Basemaps Offline
Working With Operational Data Offline
It is all about the features

- Read only access
- Work with Attachments and Relates
- Edit Features
- Use Popups
- Sync Changes with a Server
Syncing With a Server

- **Send Edits**
  - Versioned Data – download only
  - Per Layer Sync – non versioned data
  - Per Geodatabase Sync – versioned data (download only)

- **Control Over Sync Direction**
  - None
  - Download changes only
  - Upload edits only
  - Download changes and upload edits - bidirectional
Generating a Geodatabase and Showing Features

- **Feature**
  - `geometry()`
  - `attributes()`
- **Graphic**
- **GdbFeature**
- **GeodatabaseTask**
  - `fetchFeatureServiceInfo()`
  - `new GenerateParams(fsInfo) - generateGeodatabase(params)`
  - `Geodatabase.getSyncParams() - syncGeodatabase()`
- **Geodatabase**
  - `isSyncEnabled()`
  - `getFeatureTables()`
  - `hasLocalEdits()`
- **GeodatabaseFeatureTable**
- **FeatureLayer**
  - `getFeatures(x,y)`
  - `selectFeatures()`
  - `hideFeature()`
  - `isNewFeature()`
  - `queryRelated()`
  - `getAttachments()`
  - `getFields()`
- **FeatureTable**
  - `add()`
  - `update()`
  - `delete()`
  - `query()`
  - `getFields()`
Offline and Online Features

- Feature
  - Graphic
  - GdbFeature
- Layer
  - FeatureLayer
  - FeatureTable
- GeodatabaseFeatureTable
- GeodatabaseFeatureServiceTable
Working With Features

• **A Feature is a Feature**
  - No Difference for Online and Offline
  - Programming model in general is the same

• **Editing Features is Different**
  - Local Geodatabase saves edits in the database
  - Feature Service posts edits to the server when you tell it
Offline Features
How Does Sync Work?

The Flow of Data

1. Feature service with sync enabled
2. Generate geodatabase enabled for sync
3. Adds, edits, deletes
4. Generate delta geodatabase
5. Upload delta
6. Retrieve response
7. Apply response to geodatabase
Working With Locators Offline

- Prepared for Offline Use Using ArcGIS Desktop
- Programming Model Similar to Online Locators
- High Performance
Offline Locators
Working With Networks Offline

• Prepared for Offline Use Using ArcGIS Desktop
• Programming Model Similar to Online Networks
• High Performance
Offline Networks
App development

• Killed/exited apps
  - Hold on to params (or with iOS jobid) and resubmit same job (with use cached job=true)
    - If job still exists on the server, will not resubmit job
    - If partial download on disk, will not resubmit job
    - If error, will resubmit job

• Backgrounded iOS apps
  - Go to iOS session!

• Large data over 3G
  - Check for wifi access for ExportTilecCacheTask??
Licensing Offline Capabilities for Your Apps

• Viewing Vector and Raster Data
  - Basic License
• Offline Feature Editing, Geocoding and Networks
  - Standard License
Your Common Questions

- Is Versioned Data Supported?
- Are Related Tables Supported?
- What About Attachments?
- Will My Existing ArcGIS Services Work?
- How Does Conflict Detection Work?
Questions?
Thank you...

• Please fill out the session survey:

First Offering ID: 2308

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