Strategies for Building Mobile Applications Using ArcGIS Web APIs

Kelly Hutchins, Julie Powell, Chris Mahlke
Why do we care?

- 378K iPhones sold per day
- 371K Babies born per day
- 562K iOS devices
- 700K Android devices activated per day
- 200K Nokia smartphones
- 143k Blackberry devices

Agenda

Mobile Implementation Patterns
Defining a Mobile Strategy
Mobile Web Apps
Hybrid Apps
Native Apps
Implementation Patterns

- Web
- Hybrid
- Native
- Multiple

ArcGIS API for Flex
ArcGIS API for JavaScript
ArcGIS API for Silverlight*
Implementation Patterns

- Web
  - Hybrid
- Native
- Multiple

ArcGIS API for Flex
ArcGIS API for JavaScript
ArcGIS API for Silverlight*
Implementation Patterns

- Web
- Hybrid
- Native
- Multiple

ArcGIS Native SDKs for iOS | Android | Windows Phone | Windows Mobile
Implementation Patterns

- Web
- Hybrid
- Native
- Multiple

The Whole Enchilada
The right tools + the right ingredients
Defining a Mobile Strategy: Considerations

- Platform Support
- User Experience & Design
- Feature Requirements
- Development Expertise
- Performance Requirements
- Deployment
Design Guidelines

* Embrace the platform
* Clearly define: targeted user & feature requirements
* Attention to detail
* Seamless OS integration
* Custom artwork/branding

Careful!
* Fat fingers (larger hit areas)
* Performance: images, memory management, etc
* Compact, optimized
* Design for mobile, not resizing from full website

Mobile Apps and Responsive Design
Looking for Design Ideas?

http://www.mobile-patterns.com
Agenda

Defining a Mobile Strategy
Mobile Web Apps
Hybrid Apps
Native Apps
Target multiple platforms

iOS
Target multiple platforms

BlackBerry
Target multiple platforms

Windows
Target multiple platforms

Android
Compact Build

The compact version of the ArcGIS API for JavaScript is designed for building applications where bandwidth speeds and network latency are an issue. For example, on a mobile device, where you don't want the application to download. This build is also a great option if you want to use a JavaScript other than Dojo.

To use the compact build, add the following script tag to your application:

```html
<script src="http://serverapi.arcgis.com/jsapi/arcgis/3.3compact/"/>
```

Primary differences between compact build and standard build.

1. The compact build removes the dependency on the dojox namespace upon initialization. As a result, you don't need the dojo dijit module. It won't be loaded. A side-effect of this is that some of the widgets are provided.
Demo
Building Mobile Apps with the JS API
Device Access

Beginning with the GeoLocation API
Devices can present rich, device-aware features and experiences.
Offline Storage

The HTML5 App Cache enables web apps to work offline.
Frameworks

jQuery

Sencha

dōjō
Testing & Debugging

- User Agent
- Emulators
- Remote Debugger
- Device Library
Demo
Debugging
Remote debugging

- Chrome for Android
- Firefox for Android
- Safari Web Inspector

Utilities > Settings > Safari > Advanced

To use the Web Inspector, connect to Safari on your computer using a cable and access your iPhone from the Develop menu. You can enable the Develop menu in Safari’s Advanced Preferences on your computer.
<table>
<thead>
<tr>
<th>Pro</th>
<th>Con</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rapid Development cycle</td>
<td>Browser/Device fragmentation</td>
</tr>
<tr>
<td>Build once</td>
<td>Access to device hardware and software limited ****</td>
</tr>
<tr>
<td>Easy to build</td>
<td>Native apps tend to have smoother experience and slicker look and feel</td>
</tr>
<tr>
<td>Potentially retrofit existing site.</td>
<td>Difficult to monetize.</td>
</tr>
</tbody>
</table>
Agenda
Defining a Mobile Strategy
Mobile Web Apps
Hybrid Apps
Native Apps
Hybrid

- Hybrid apps are native apps with embedded HTML/JS/CSS
- Hybrid apps have all the advantages of native apps
- Hybrid consist of web and native portions
Hybrid

- Hybrid apps are native apps with embedded HTML/JS/CSS
- Hybrid apps have all the advantages of native apps
- Hybrid consist of web and native portions
Hybrid

- Hybrid apps are native apps with embedded HTML/JS/CSS
- Hybrid apps have all the advantages of native apps
- Hybrid consist of web and native portions
Hybrid

- Accelerometer
- Camera
- Capture
- Compass
- Connection
- Contacts
- Device
- Events
- File
- Geolocation
- Globalization
- InAppBrowser
- Media
- Notification
- Splashscreen
- Storage
Adobe PhoneGap Build System

HTML5  CSS3  JS

PhoneGap Build System

Adobe / Yohei Shimomae
http://phonegap.com/about/artwork/
Adobe PhoneGap Build System

HTML5  CSS3  JS

PhoneGap Build System

Adobe / Yohei Shimomae
http://phonegap.com/about/artwork/
What’s the Scoop?
Release Builds

Adobe Flex SDK
ArcGIS API for Flex

Package for Target Platform

Export Release Build

Project: BasicMap
Location: BasicMap.mxml

Target platforms
- Google Android
- Apple iOS
- BlackBerry Tablet OS

Export
Export to folder: 
(in /BasicMap) 
Base filename: BasicMap

Export as:
- Signed packages for each target platform
- Signed AIR package for installation on desktop
- Intermediate AIRi package that must be manually signed later

Deploy Application: Depends on Platform
UX & Device Integration

• Sizing your app:
  - Stage.stageWidth, Stage.stageHeight, Capabilities.screenDPI

• Orientation

• Full screen

• Touch input/Gestures

• Native extensions for Adobe AIR, i.e.
  - Network info
  - Push Notifications

• File Storage
Sensors

Examples

- **Geolocation**: latitude and longitude, heading, speed
- **Accelerometer**
- **CameraUI and CameraRoll**
- **Gestures**
Development with the ArcGIS API for Flex

• Offline workflows
  - LocalTileLayer (ArcGISLocalTiledLayer)
  - Tile Packages (TPK)
• Map Rotation
• Sparkifying components & skinning
  - Geocoder component
• Mobile Samples
Demo
Flex Mobile Dev
## Hybrid

<table>
<thead>
<tr>
<th>Pro</th>
<th>Con</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rapid Development cycle.</td>
<td>Debugging can be challenging</td>
</tr>
<tr>
<td>Build once</td>
<td>Access to device hardware and software limited ****</td>
</tr>
<tr>
<td>Easy to build</td>
<td>Native apps tend to have smoother experience and slicker look and feel</td>
</tr>
<tr>
<td>Potentially retrofit existing site.</td>
<td></td>
</tr>
</tbody>
</table>
Agenda
Defining a Mobile Strategy
Mobile Web Apps
Hybrid Apps
Native Apps
## Native

<table>
<thead>
<tr>
<th>Programming Language</th>
<th>Apple</th>
<th>Windows</th>
<th>Android</th>
<th>BlackBerry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obj-C, C/C++</td>
<td>XCode</td>
<td>Visual Studio</td>
<td>Eclipse</td>
<td>Eclipse</td>
</tr>
<tr>
<td>C/C++</td>
<td>.app</td>
<td>.xap</td>
<td>.apk</td>
<td>.alx, .cod</td>
</tr>
<tr>
<td>Java, C/C++</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Distribution</th>
<th>App Store</th>
<th>Windows MarketPlace</th>
<th>Google Play</th>
<th>BlackBerry App World</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Native

<table>
<thead>
<tr>
<th>Programming Language</th>
<th>Apple</th>
<th>Windows</th>
<th>Android</th>
<th>BlackBerry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obj-C, C/C++</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C/C++</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java, C/C++</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IDE</th>
<th>XCode</th>
<th>Visual Studio</th>
<th>Eclipse</th>
<th>Eclipse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary Executable</td>
<td>.app</td>
<td>.xap</td>
<td>.apk</td>
<td>.alx, .cod</td>
</tr>
<tr>
<td>Distribution</td>
<td>App Store</td>
<td>Windows MarketPlace</td>
<td>Google Play</td>
<td>BlackBerry App World</td>
</tr>
</tbody>
</table>
ArcGIS Runtime

- Family of SDKs for multiple platforms
  - Consistent capabilities
- Native to the platform
  - For building great apps
- Lightweight and fast
- Powerful
- Easy
SDK Platforms

- **Android 10.1.1**
  - Java/Android

- **Windows Mobile 10.1.1**
  - .NET

- **iOS 10.1.1**
  - Objective-C
  - Cocoa Touch

- **Windows Phone 10.1.1**
  - .NET/XAML

- **WPF 10.1.1**
  - .NET/XAML

- **JavaSE 10.1.1**
  - Java
  - Win/Linux

**Core Runtime**
## Native

<table>
<thead>
<tr>
<th>Pro</th>
<th>Con</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast</td>
<td>Several codebases to maintain</td>
</tr>
<tr>
<td>Access to all native hardware</td>
<td>Access to device hardware and software limited ****</td>
</tr>
<tr>
<td>UX</td>
<td>Native apps tend to have smoother experience and slicker look and feel</td>
</tr>
<tr>
<td></td>
<td>Clandestine “app” store policies</td>
</tr>
</tbody>
</table>
Final Thoughts

- No single approach delivers all the benefits all the time.

- A mobile strategy should consider the developing trends in the mobile space.

- Recommendation: A flexible solution

- Utilize Esri’s existing mobile applications and samples
Everyone will be connected...
• Julie Powell (@JuliePowellGIS)
• Chris Mahlke (@ChrisMahlke)
• Kelly Hutchins (@kellyhuthcins)

http://esriurl.com/survey

Offering ID: