



Tips for working with disconnected web mapping apps

Andy Gup, Javier Abadia



Why disconnected JavaScript?

Mobile web map used in areas of intermittent or no internet

Ability to reload or restart web map in areas of intermittent or no internet

Lightweight cross-browser functionality

Need a full featured, robust offline solution?

ArcGIS Runtimes for iOS, Android, Qt and .NET!

Includes integrated support for offline editing and synchronization.

Also fully supports related tables, sub-types, domains and much more.

TrailYelper

Mtn Biking/Hiking app



Use Cases

When would you consider offline use of web maps?

- Viewing simple maps
- Lightweight data collection
 - VGI
 - Simple editing
- Devices
 - laptop
 - smartphone / tablet

What would the developer need to do?

Enhancing an online app with offline functionality

- Keep a local copy of HTML/CSS, [all] Javascript code and other static resources (imgs)
- Keep a local copy of map data...
 - Basemap tiles
 - feature layers
- Keep temporary local copy of edits made to the feature layers...
 - including new attachments
- ...and “teach” the JS API to use the local data instead of fetching/pushing data through the network

Is it possible?

Yes, but it can be complicated!

- HTML5 application cache mechanism
 - cache HTML, CSS, images and .js files
- HTML5 storage APIs
 - indexed db (50Mb +)
 - async, store key,value pairs, more capacity
 - Increasing support (more when using shim)
 - Websql (50Mb +)
 - Spec no longer maintained. Chrome, Safari, Opera
 - localStorage
 - sync, store key,value pairs, low capacity (<5Mb)
 - good browser support

Is it possible?

What mechanisms can we use to do what we need to do?

- HTML5 application cache mechanism
 - cache HTML, CSS, images and .js files
- HTML5 storage APIs
 - indexed db (50Mb +)
 - async, store key,value pairs, more capacity
 - Increasing support (more using shim)
 - Websql (50Mb +)
 - Spec no longer maintained. Chrome, Safari, Opera
 - localStorage
 - sync, store key,value pairs, low capacity (<5Mb)
 - good browser support

good for...

Restarts and reloads

Basemap tiles

Feature edits

Min Zoom Level (farthest from ground)

0

Current Zoom Level

16

Max Zoom Level (closer to ground)

19

Level	Tile Count	Size Mb (aprox.)
13	2	0.07 Mb
14	4	0.14 Mb
15	6	0.21 Mb
16	12	0.43 Mb
17	35	1.25 Mb
18	104	3.72 Mb
19	375	13.42 Mb
Total	551	19.71 Mb

Prepare for Offline

Delete All Tiles

Go Offline

Usage: 0.13

Using proxy: Yes



Demos

Tiles, TPKLayer

Disconnected JavaScript workflows

Scenario 1 – online > offline > online

Simplest scenario, if we don't need to support “browser reload”
(scenario 2)

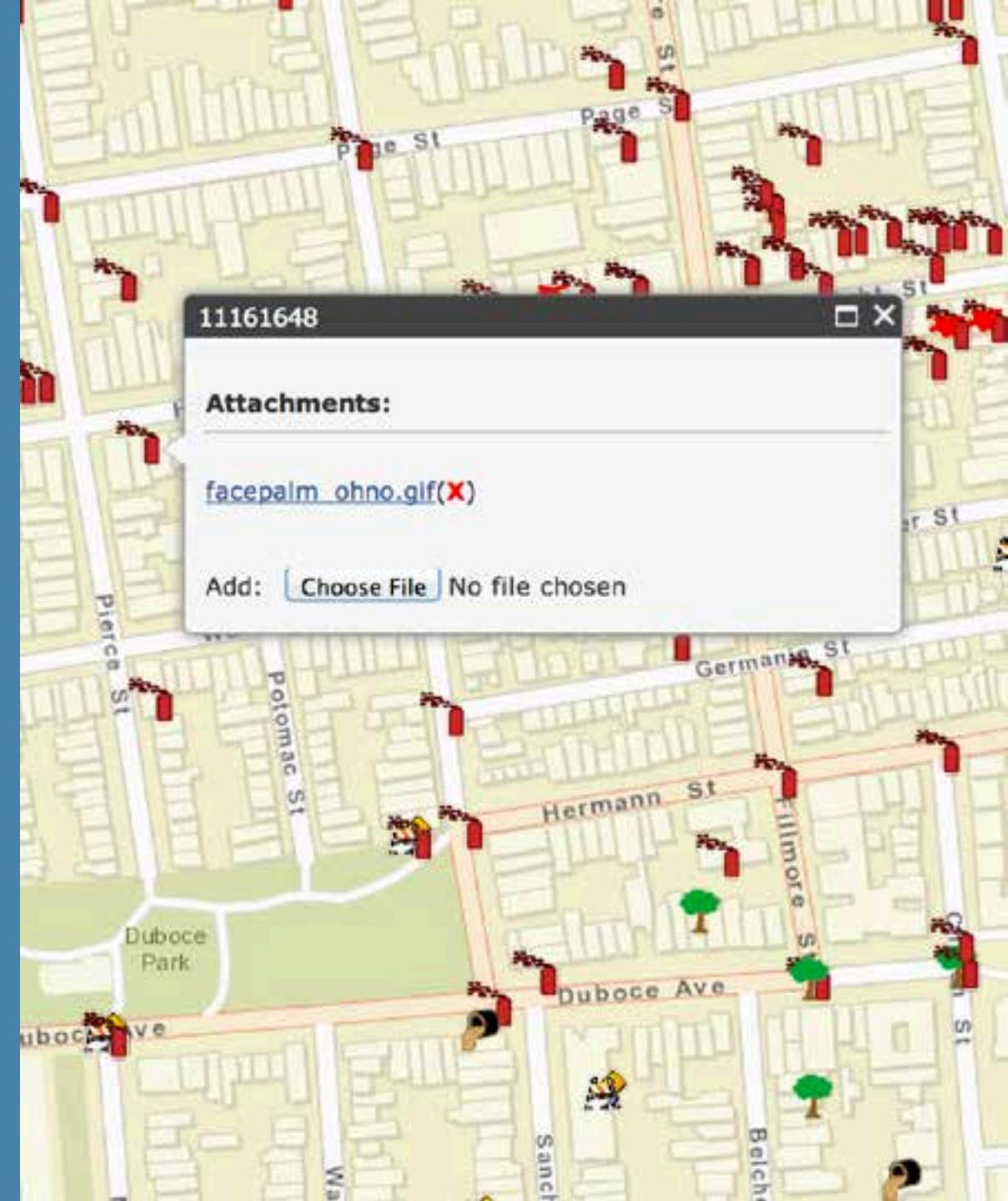
1. no need to cache code and static resources
(e.g. css, html, js already loaded online)
2. Feature layer already temporarily cached via mode SNAPSHOT

Scenario 2 – online > offline > online (+ restart/reload)

Scenario 3 – offline tiles-only (TPKLayer)

Feature Editing

Attachments included!



Caniuse.com

- Candidate Recommendation

data client-side, allows indexed database query.

	IE	Firefox	Chrome
8.0			
9.0		28.0	33.0
10.0		29.0	34.0
11.0		30.0	35.0
		31.0	36.0
		32.0	37.0
		33.0	38.0

Offline-editing-js project

Lightweight libraries and sample apps:

<https://github.com/Esri/offline-editor-js>

Recap

Reminder: the ArcGIS Runtimes already have built-in, robust support for full offline use cases.

Carefully examine your workflows: intermittent vs no internet.

Offline support in JavaScript is challenging and some things not possible or recommended.

Questions?

Andy Gup

@agup

agup@esri.com

Javier Abadia

@javierabadia

javier.abadia@esri.es



esri

Understanding our world.