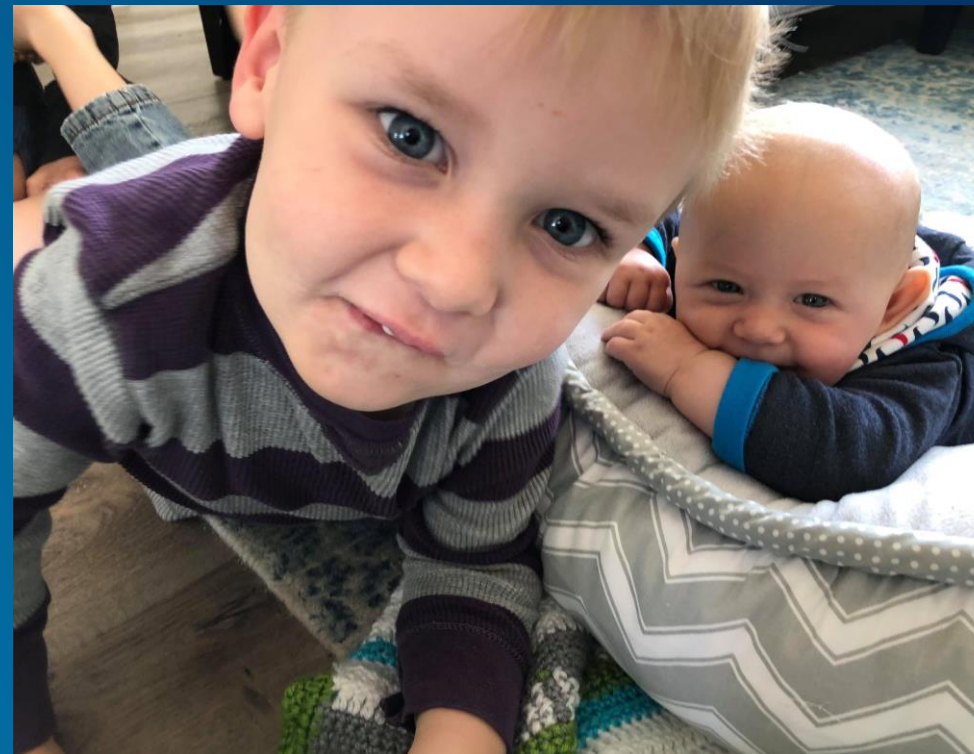
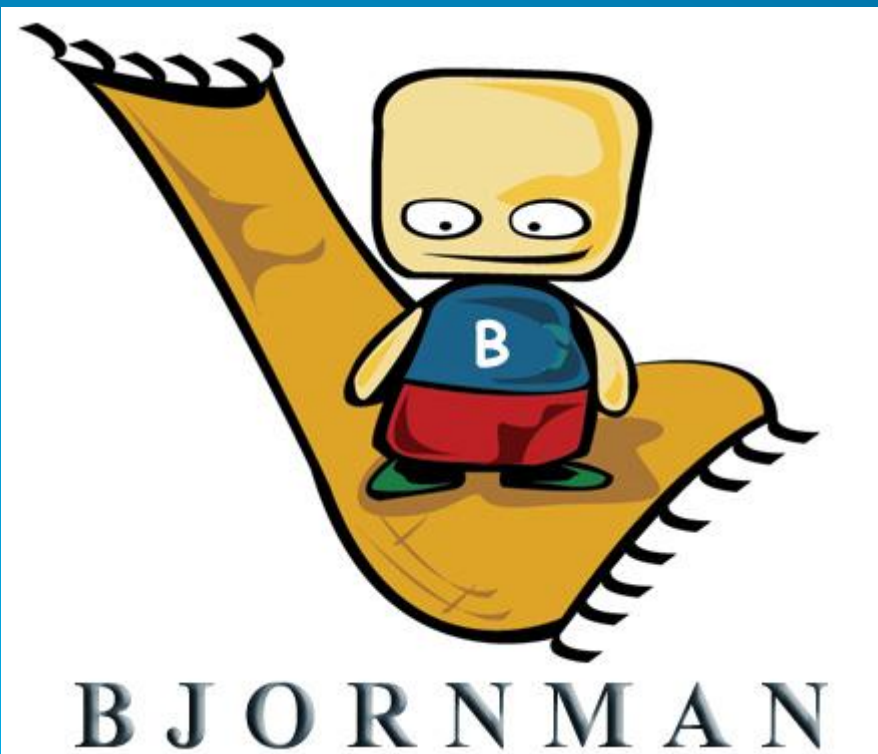


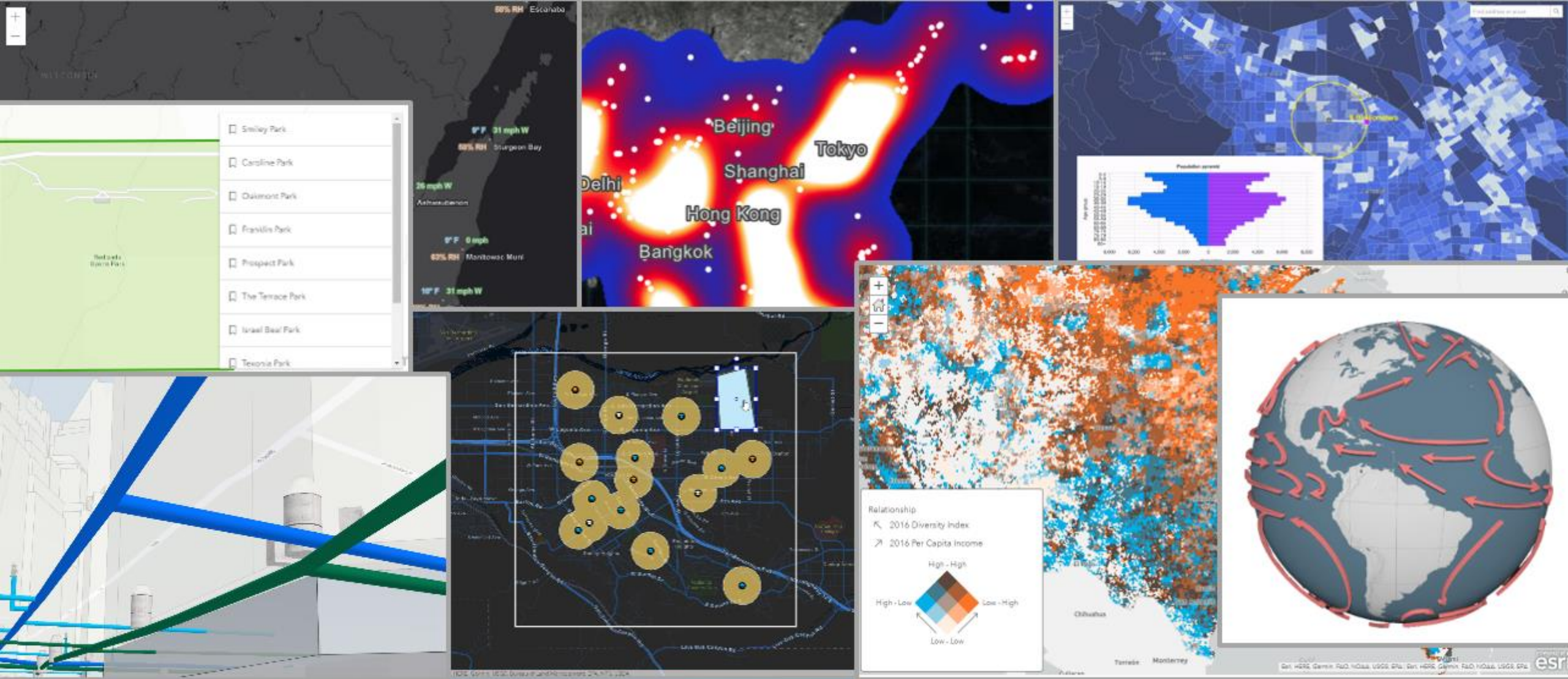


ArcGIS API for JavaScript: What's New

Bjorn Svensson, Julie Powell, Noah Sager

**GIS
INSPIRING
WHAT'S
NEXT**





4.8 & 3.25

Released July 5th

ArcGIS API for JavaScript

[Home](#)[Guide](#)[API Reference](#)[Sample Code](#)[Community](#)[Get Started](#)[Overview](#)[Release notes](#)[Get the API](#)[System requirements](#)[Community](#)[Migrating from 3.x](#)[Working with the API](#)[Visualization](#)[Reference](#)

Release notes for 4.8

[Release notes](#)

- Topics

- Labeling
- Increased limit for the number of point features displayed in a web scene
- Sky can be replaced with a color or made entirely transparent
- Underground visualization improvements
- FeatureLayer performance improvements
- Improved support for layer view queries
- Support for Request Modification
- Heatmap support
- Widget updates
 - Bookmarks widget
 - Sketch updates
 - CSS widget updates
- Support for Catalan and Hungarian locales
- Better WebMap Support
- Passing tokens on a layer

CHECK OUT THE RELEASE NOTES

New features, guide topics, samples

ArcGIS API for JavaScript

[Home](#)[Guide](#)[API Reference](#)[Sample Code](#)[Community](#)

4.8

▼ Get Started

Featured samples

[Intro to MapView \(2D\)](#)[Intro to SceneView \(3D\)](#)[Intro to layers](#)[Intro to popups](#)[Intro to widgets](#)

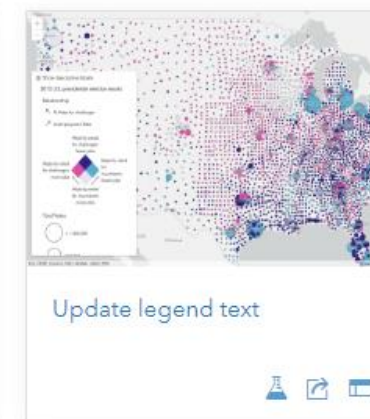
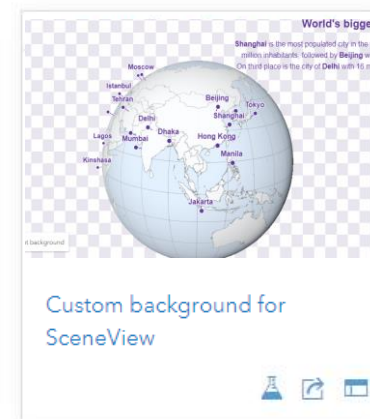
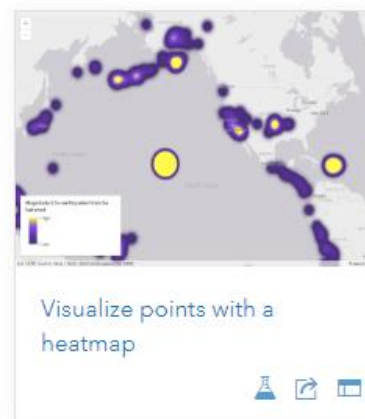
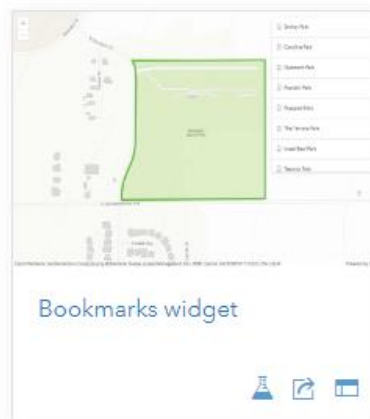
▼ Latest Samples

[Add labels to a FeatureLayer](#)[Add multiple label classes to a layer](#)[Multi-line labels](#)

12 results for Sample Code: 4.8

Tags

4.8



4.8 SAMPLES

Search for “4.8” or go under “Latest Samples”
12+ new samples

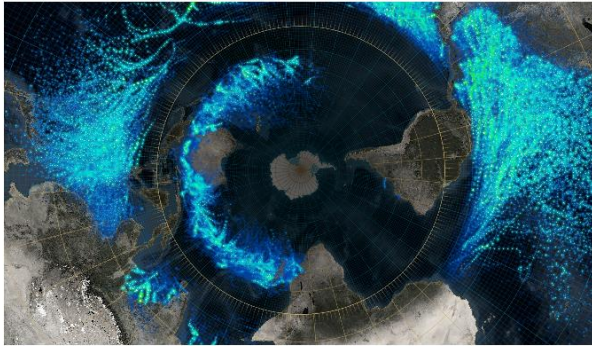
ArcGIS API for JavaScript

[Home](#) [Guide](#) [API Reference](#) [Sample Code](#) [Community](#)

- > [Get Started](#)
- > [Migrating from 3.x](#)
- > [Working with the API](#)
- > [Visualization](#)
- > [Reference](#)

Overview

The ArcGIS API for JavaScript 4.x reimagines the API in terms of its support for both 2D and 3D, its ease of use, its ability to work with map and layer web resources stored as items in the ArcGIS geoinformation model, and its support for building engaging and elegant user experiences.



Dev Summit 2018 Hurricanes app using the ArcGIS API for JavaScript 4.7

Keyboard shortcuts



?

esc

g h

g g

g a

g s

g t

spacebar

shift spacebar

g b

Site wide shortcuts

Bring up this help dialog

Dismiss this help dialog

Go to Home

Go to Guide

Go to API Reference

Go to Sample Code

Scroll to top of page

Scroll page down

Scroll page up

Scroll to bottom of page

/

esc

enter

e s

s

v l

Sample Code

Focus search

Remove search focus

Focus first search result

Explore in the Sandbox

Share in online code editor

View live example

FeatureLayer

[Constructors](#) | [Properties](#) | [Methods](#) | [Type definitions](#) | [Events](#)

```
require(["esri/layers/FeatureLayer"], function(FeatureLayer) { /* code goes here */ });
```

Class: [esri/layers/FeatureLayer](#)Inheritance: [FeatureLayer](#) → [Layer](#) → [Accessor](#)Subclasses: [StreamLayer](#)

Since: ArcGIS API for JavaScript 4.0

ArcGIS API for JavaScript

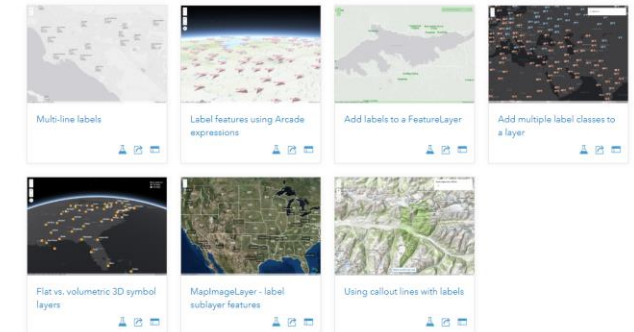
[Home](#) [Guide](#) [API Reference](#) [Sample Code](#) [Community](#)

labeling

- > [Get Started](#)
- > [Latest Samples](#)
- > [Mapping and Views](#)
- > [Layers](#)
- > [FeatureLayer](#)
- > [SceneLayer](#)
- > [MapImageLayer](#)
- > [CSVLayer](#)
- > [ImageryLayer](#)
- > [PointCloudLayer](#)
- > [Custom Layers](#)
- > [Query](#)
- > [Draw](#)
- > [Visualization](#)
- > [Popups](#)
- > [Editing](#)
- > [Graphics](#)
- > [Searching](#)

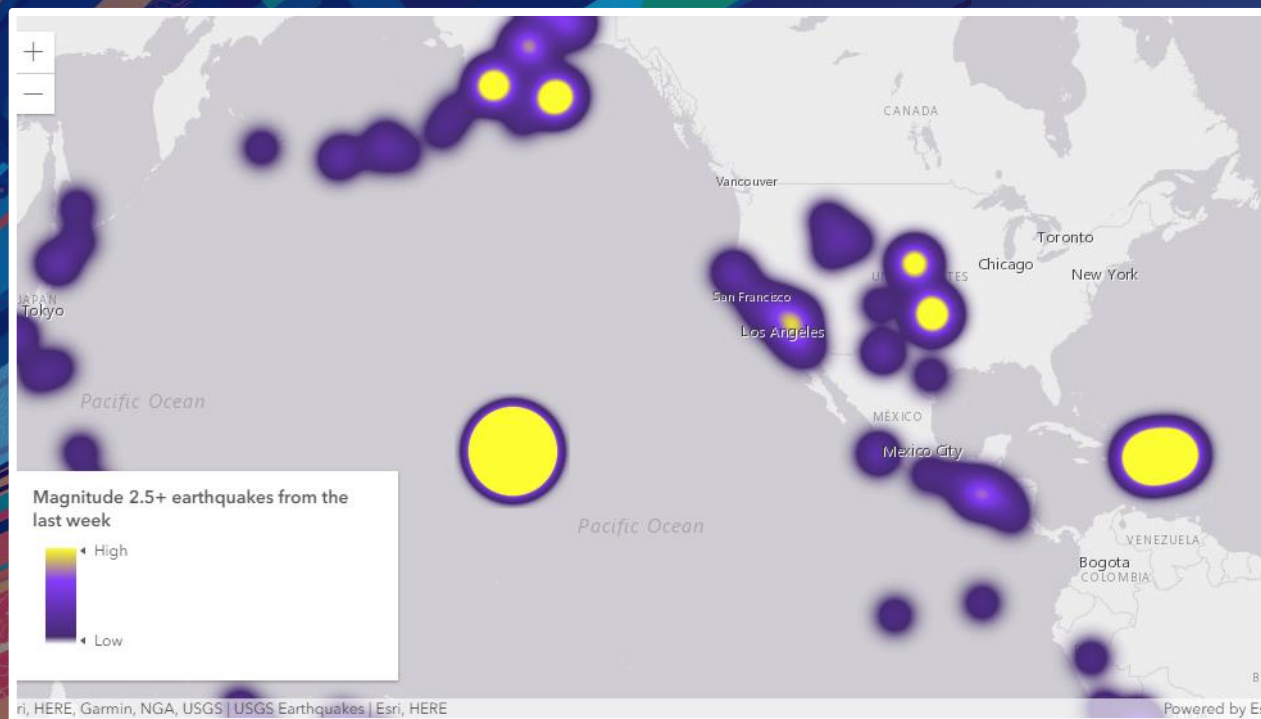
7 results for Sample Code: labeling

Tags

[labeling](#) [labels](#) [labelingInfo](#)


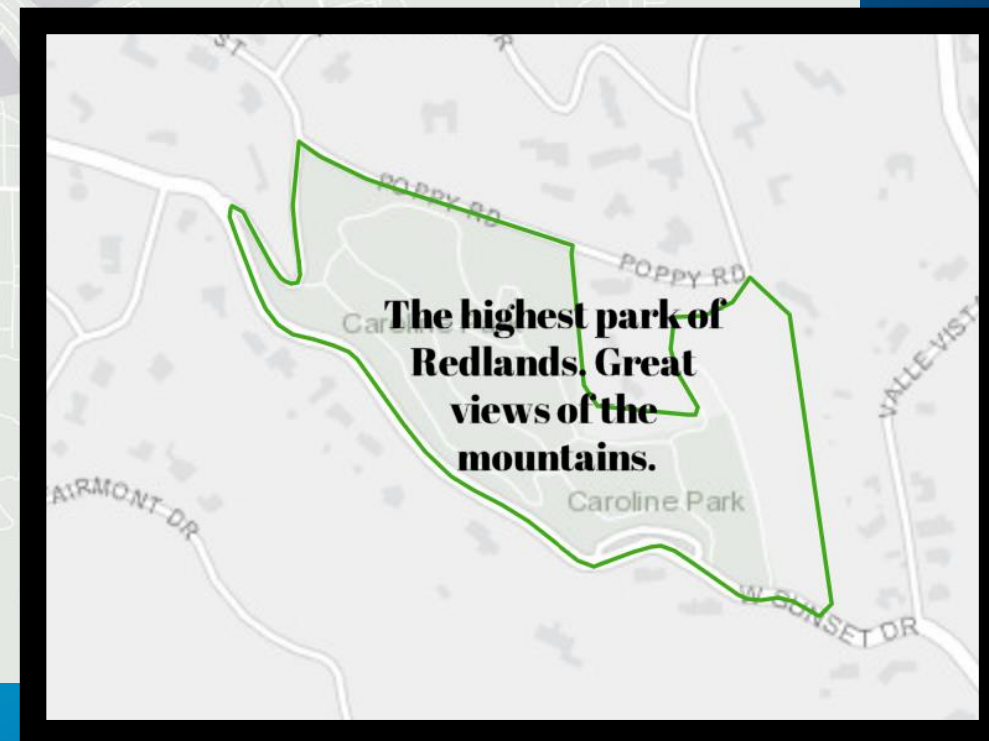
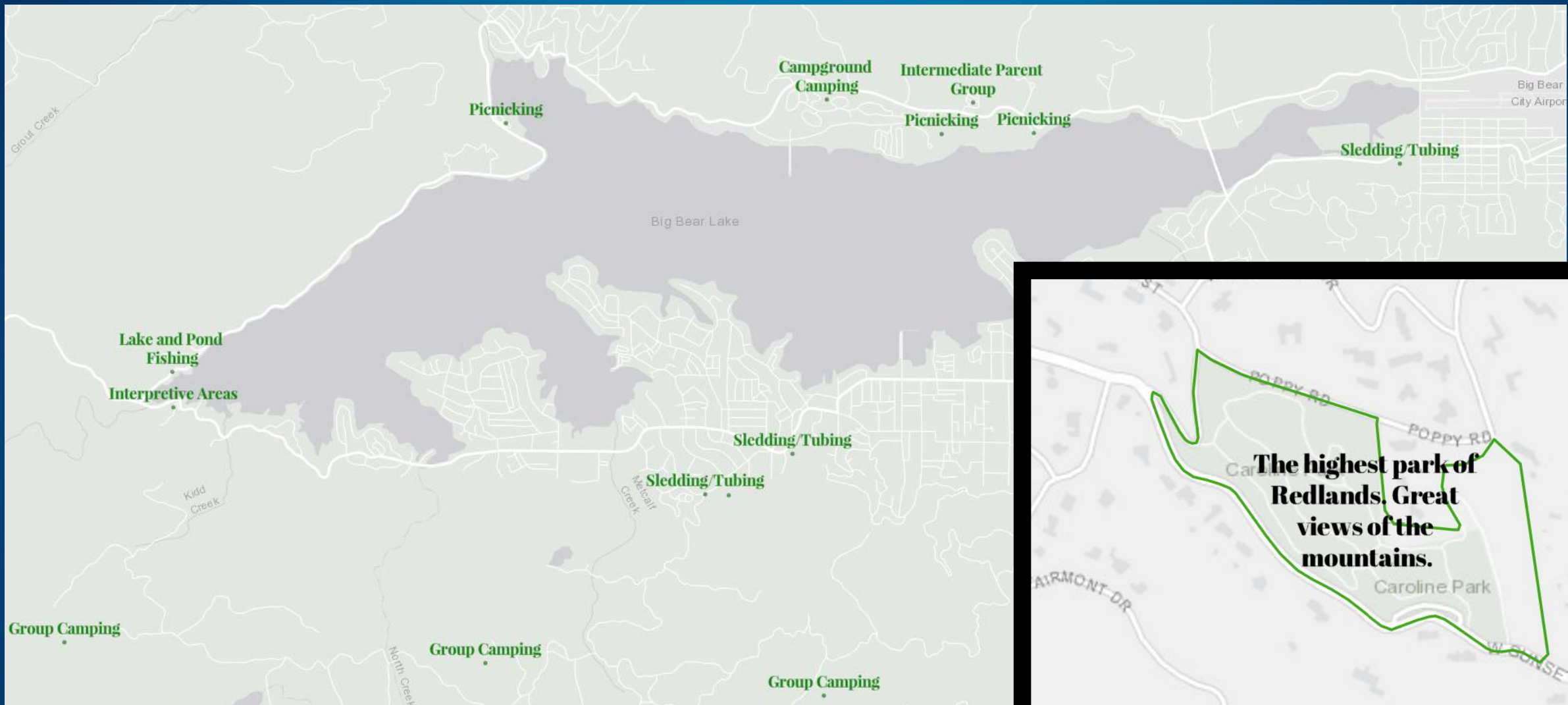
SDK IMPROVEMENTS

Guide pages, Shortcuts, Quick Links,
Sample search and tags



Visualization

Clustering, heatmap, smartmapping, Arcade expressions, labeling, ...



LABELING

2D labeling of client-side features *

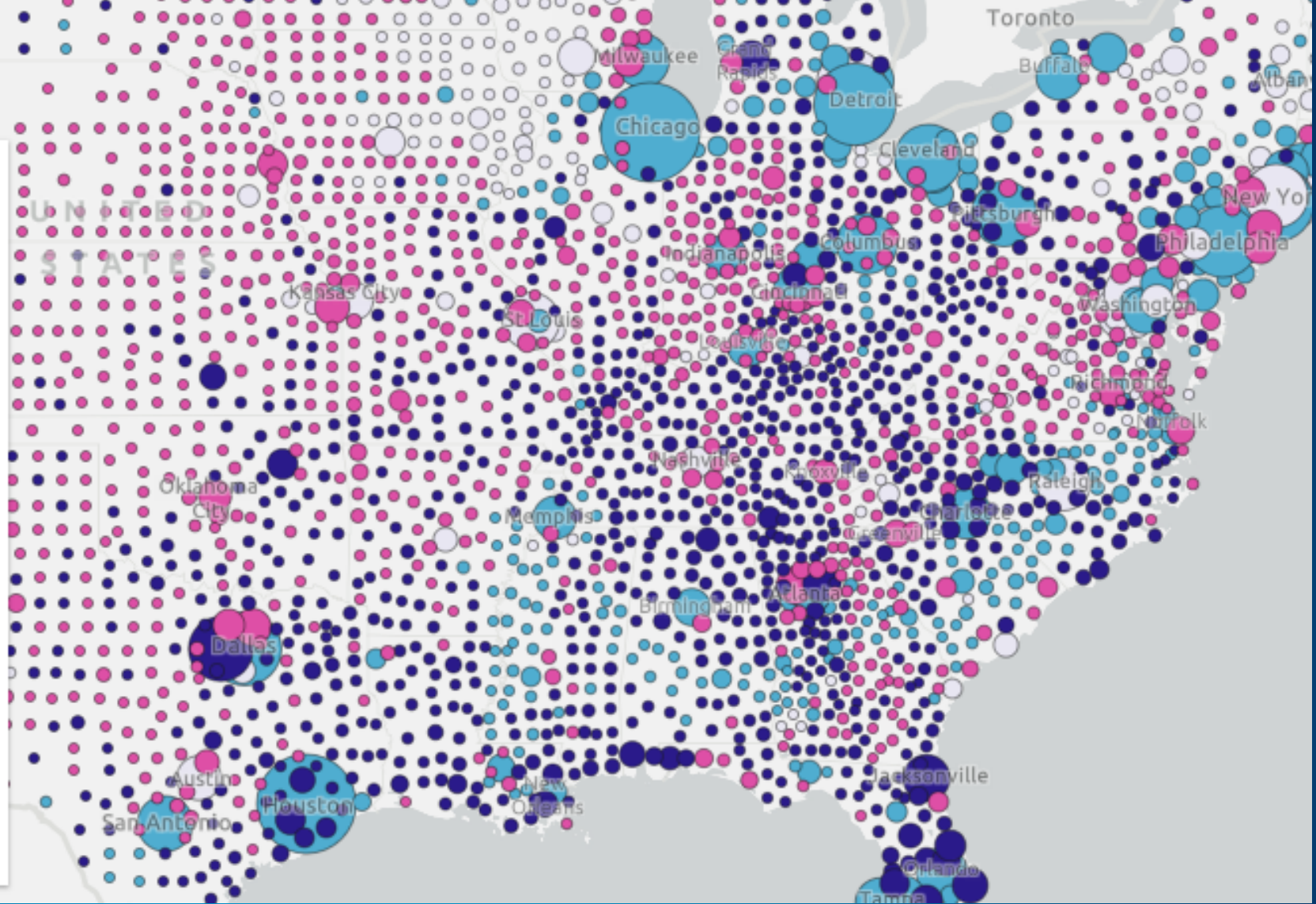
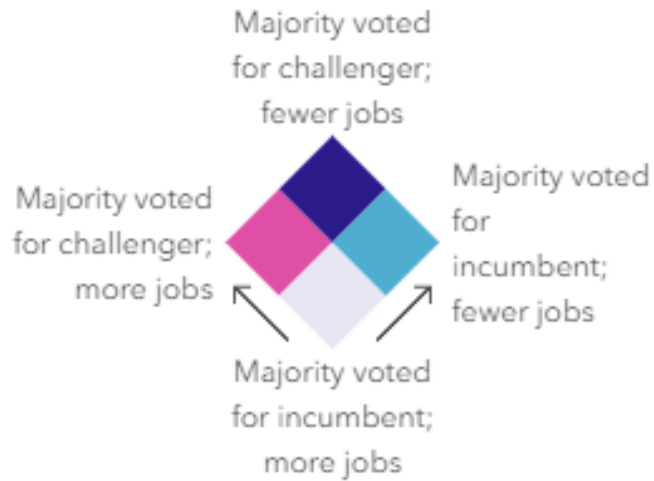
☒ Show descriptive labels

2012 U.S. presidential election results

Relationship

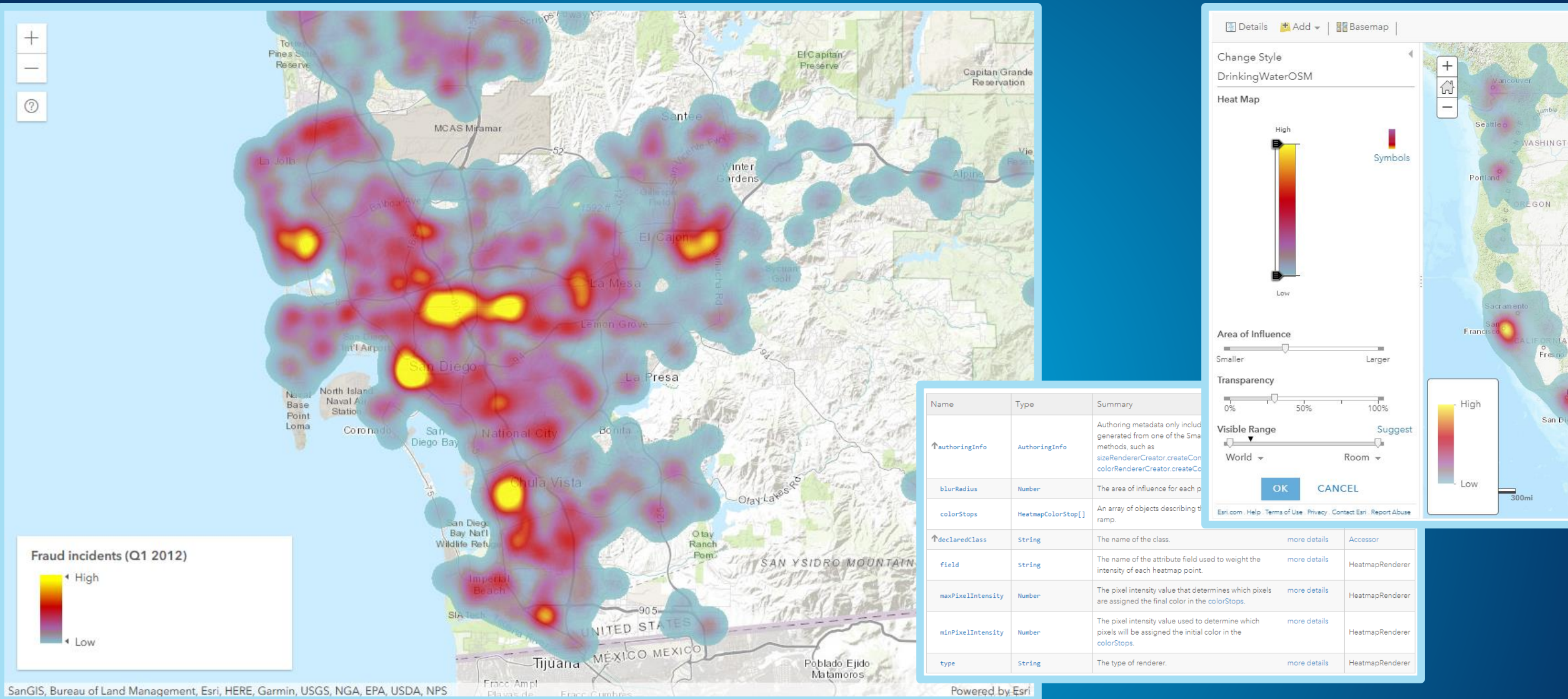
↖ % Votes for challenger

↗ Unemployment Rate



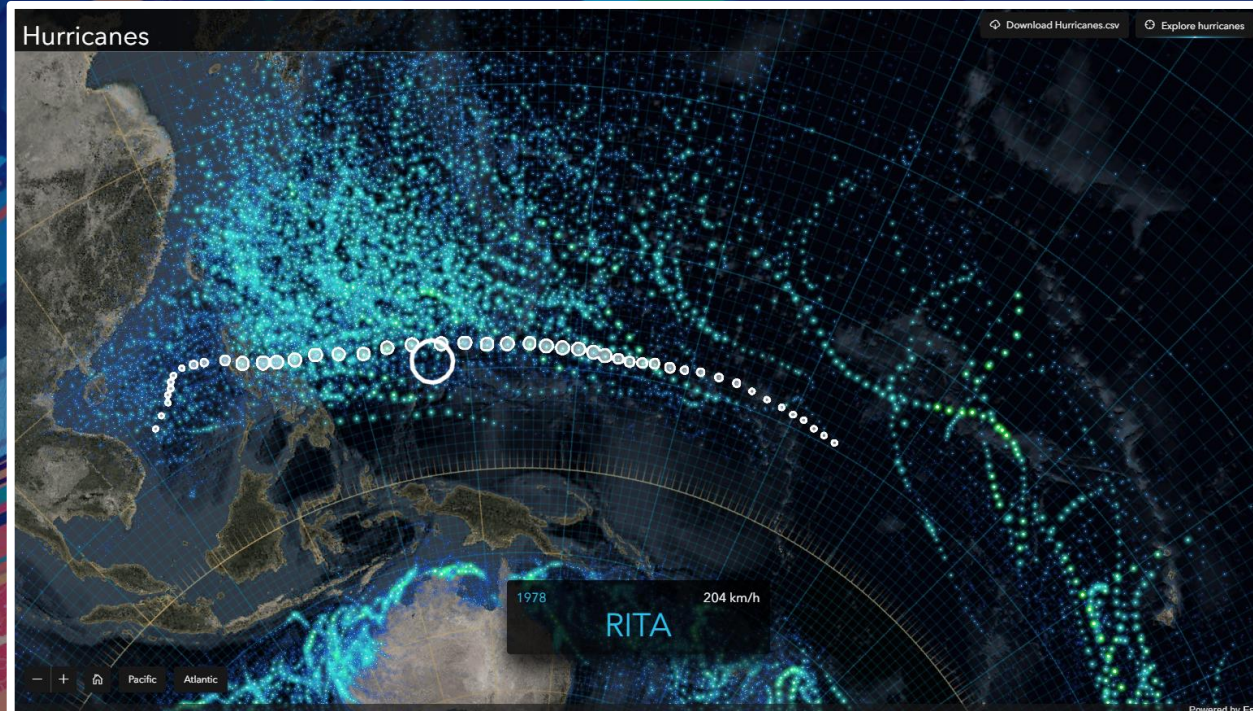
RELATIONSHIP RENDERER

a renderer for exploring the potential relationship between two numeric attributes
... a.k.a. Bivariate Choropleth Maps



HEATMAP RENDERER (2D)

visualize large, dense point datasets as a raster surface to emphasize areas with a high density of features.



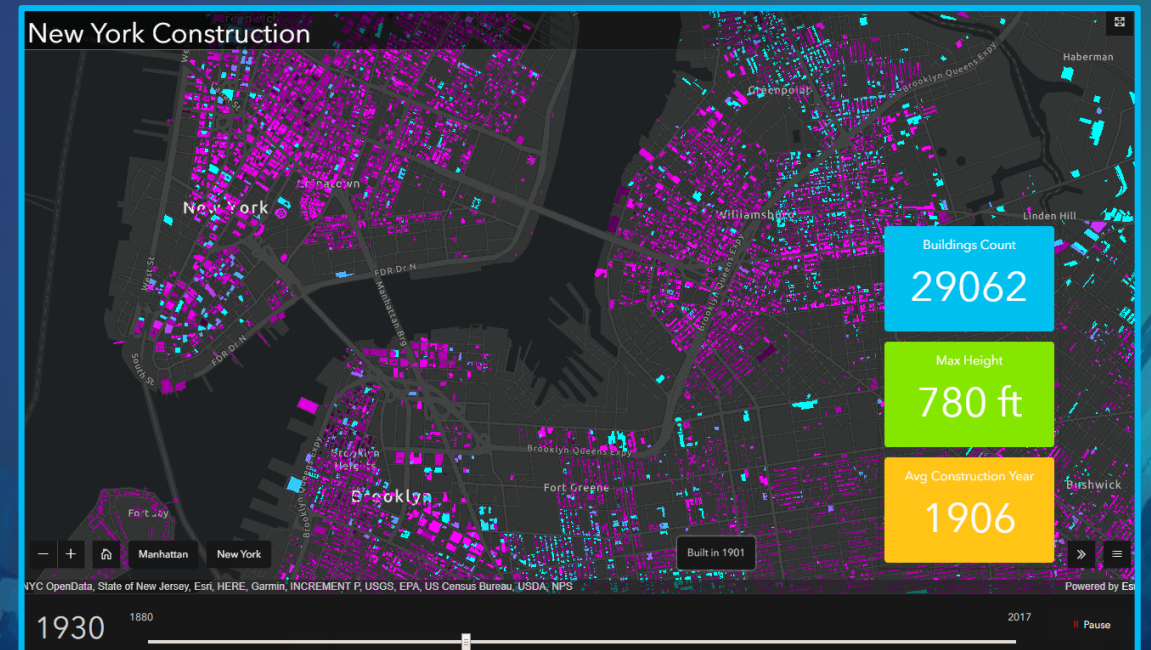
Do more with more:

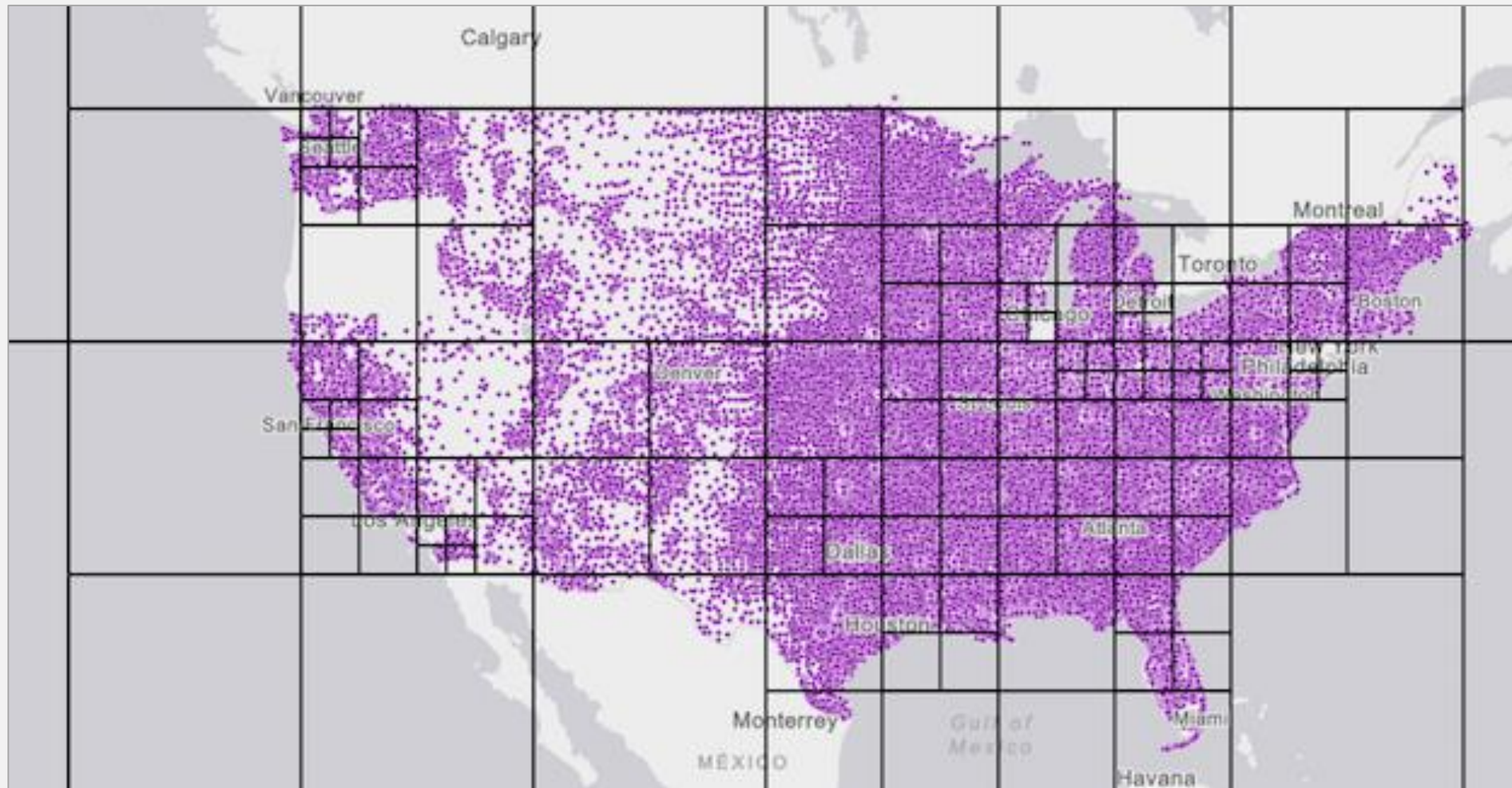
Better feature layers & client-side processing

FEATURE LAYERS

Maximizing performance: a look under the hood

1. Query in an efficient way -> feature tiles & caching
2. Minimize size of data delivered to browser-> *binary* format (PBF)
3. Fast rendering -> WebGL (opt in)

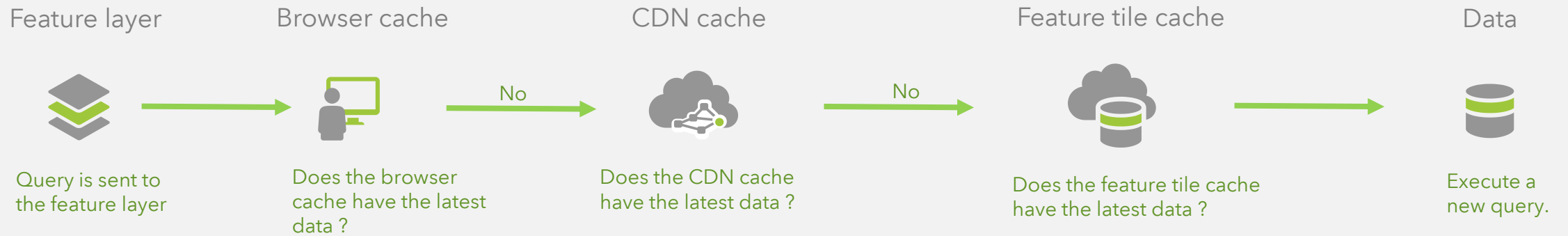




FEATURE FETCH STRATEGY

- Feature tile queries
- Progressive feature tile subdivisions
- Smaller tiles in feature dense areas

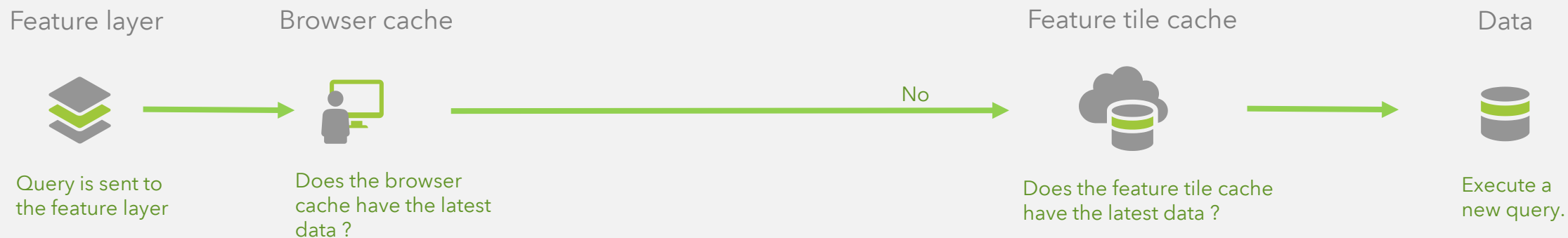
Public Layer



FEATURE TILE QUERIES AND RESPONSE CACHING

- Tile requests are consistent.
- Cache feature tiles in the browser, CDN, and Online

Private Layer

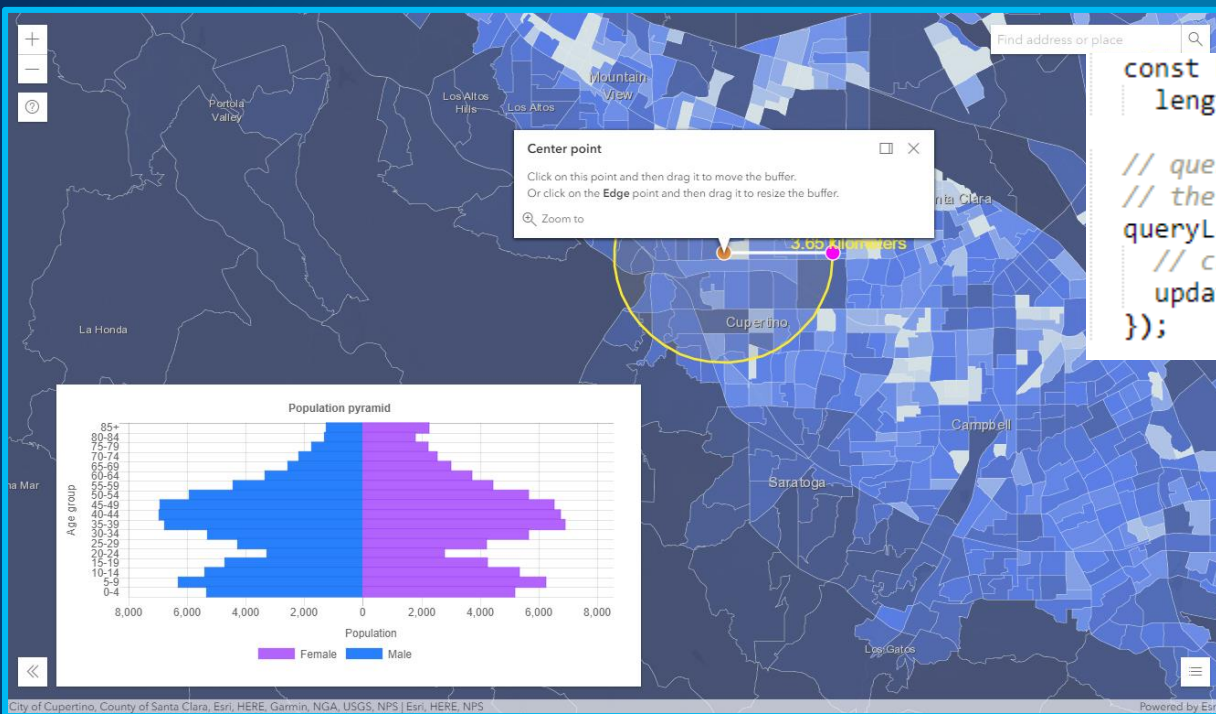


FEATURE TILE QUERIES AND RESPONSE CACHING

- Server-side caching of private data happens within Online
- Authorized users have access

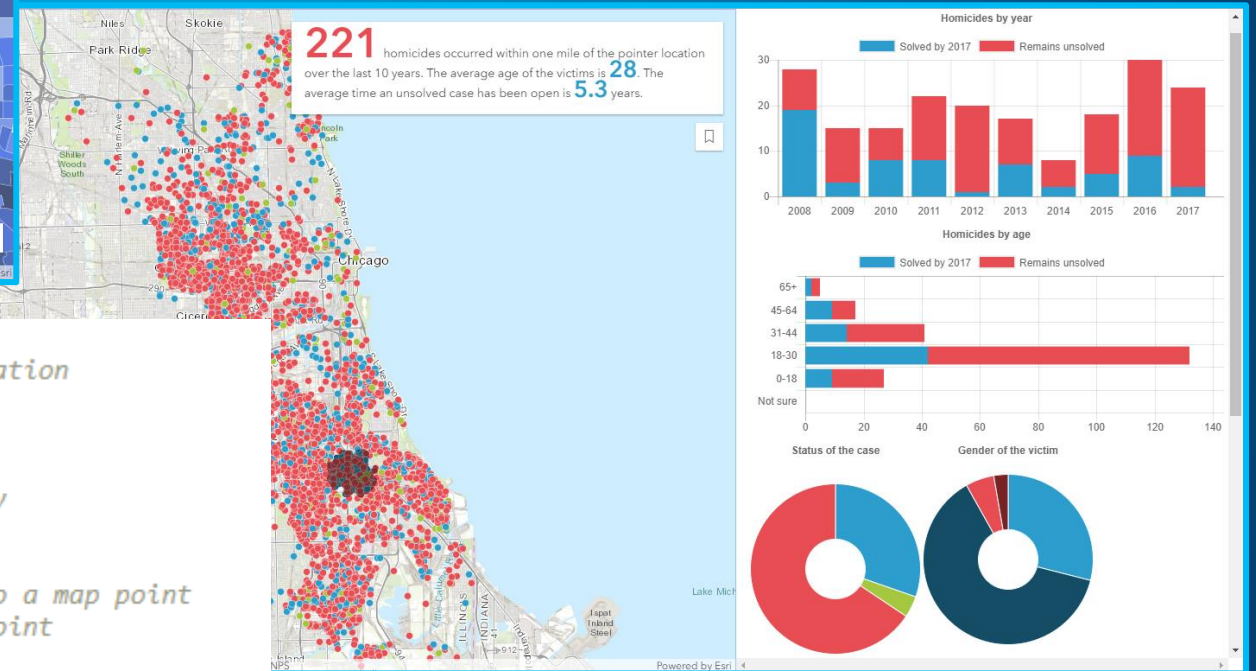
Build interactive workflows with the data

- Client-side querying & statistics
- Geometry engine
- Projection engine



```
const buffer = geometryEngine.geodesicBuffer(centerGraphic.geometry,
length, unit);
```

```
// query female and male age groups of the census tracts that intersect
// the buffer polygon on the client
queryLayerViewAgeStats(buffer).then(function(newData) {
// create a population pyramid chart from the returned result
updateChart(newData);
});
```



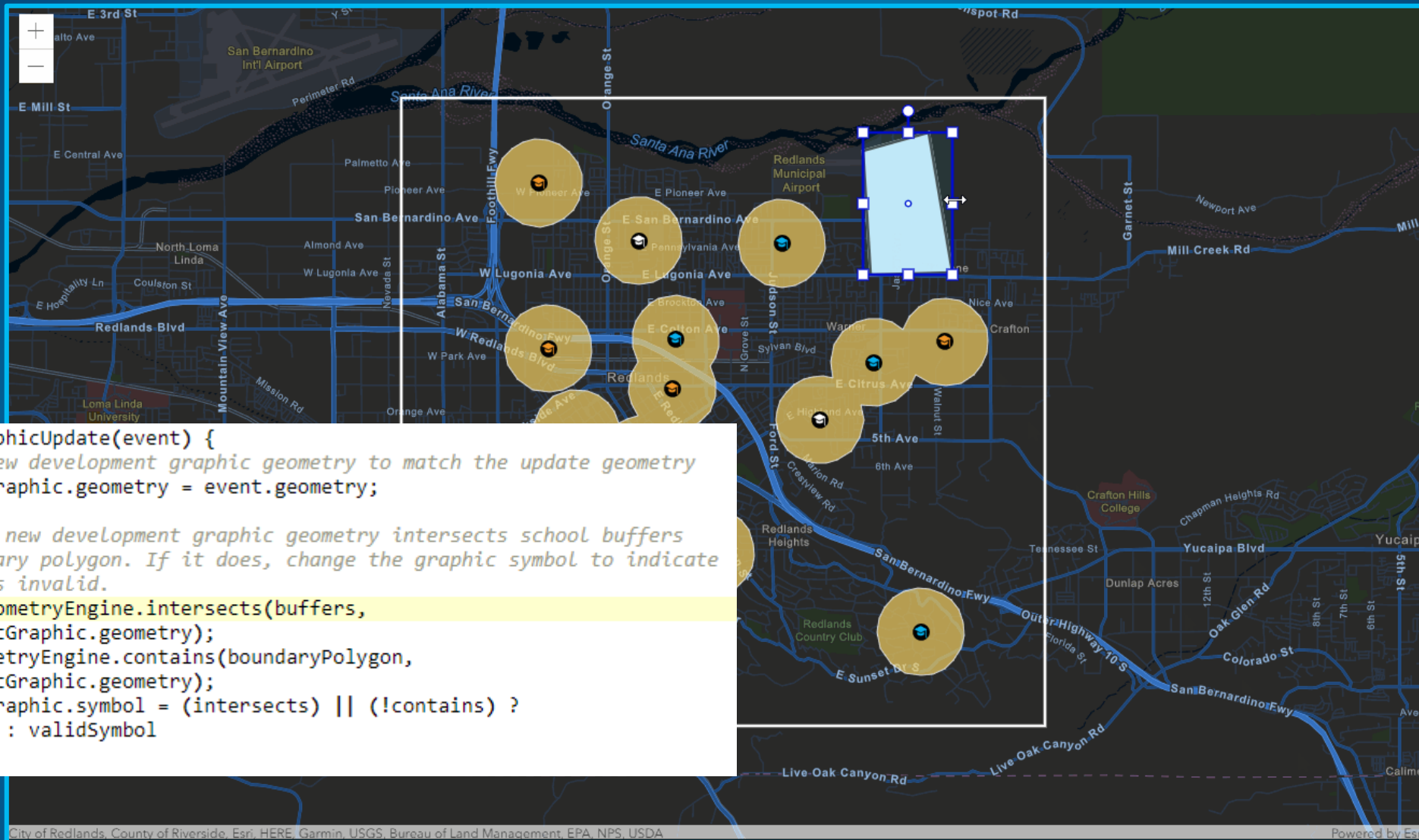
```
/**
 * Queries statistics against the layer view at the given screen location
 */
function queryStatsOnMouseMove(layerView, event) {

// create a query object for the highlight and the statistics query

const query = layerView.layer.createQuery();
query.geometry = view.toMap(event); // converts the screen point to a map point
query.distance = 1; // queries all features within 1 mile of the point
query.units = "miles";
```

CLIENT-SIDE QUERYING & STATISTICS

- Done on FeatureLayerView
- 4.8 added full spatial queries & expressions for fields & stats



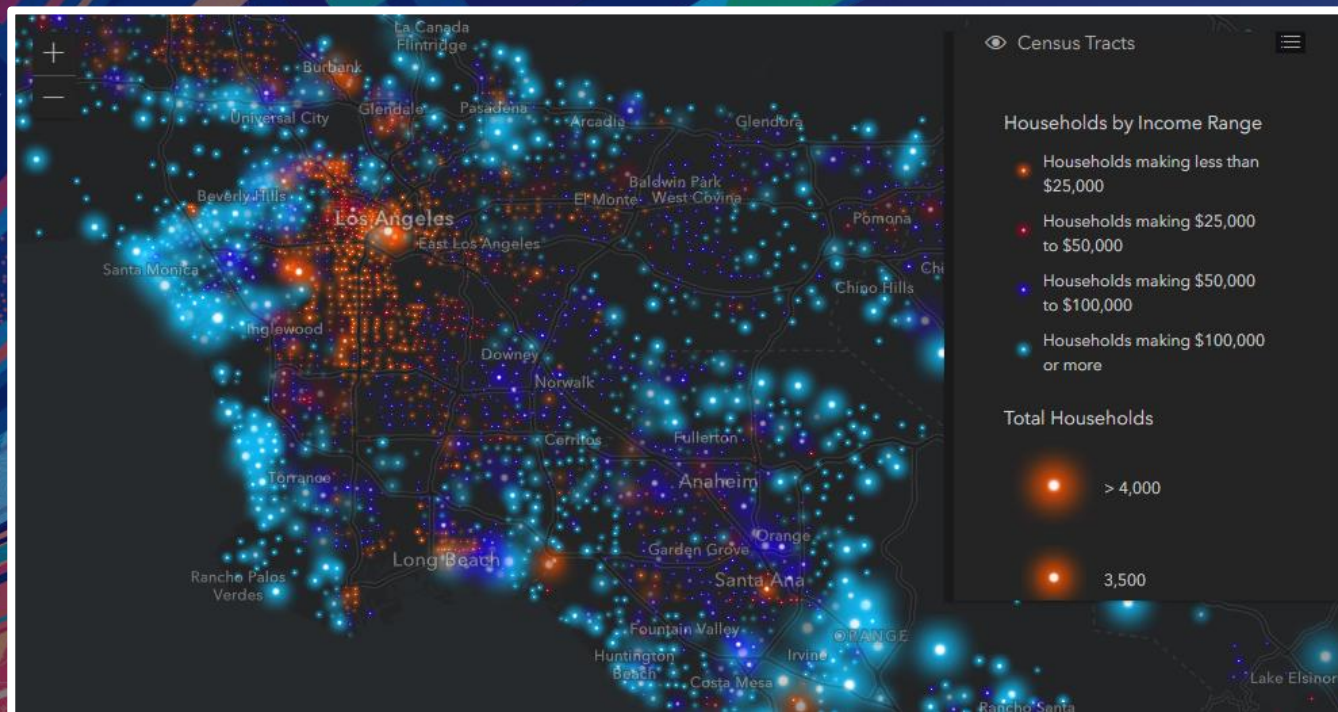
GEOMETRY ENGINE

Not new! But you can use it in new ways with other updates such as sketching.



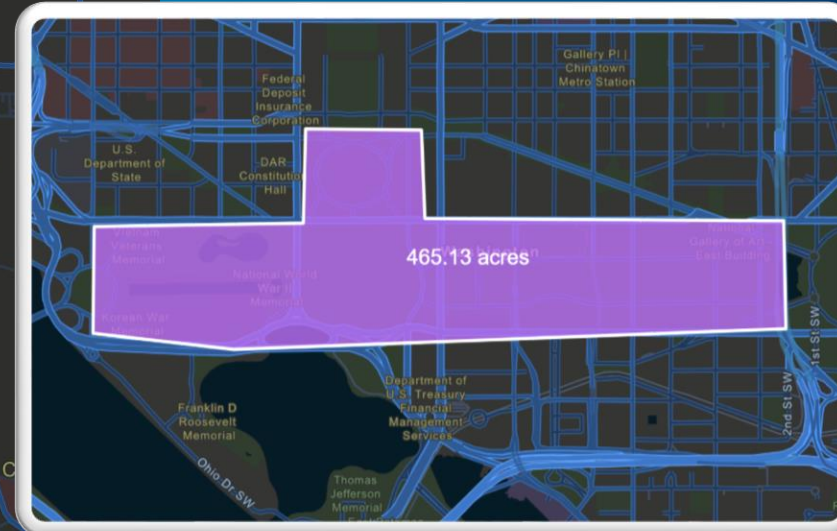
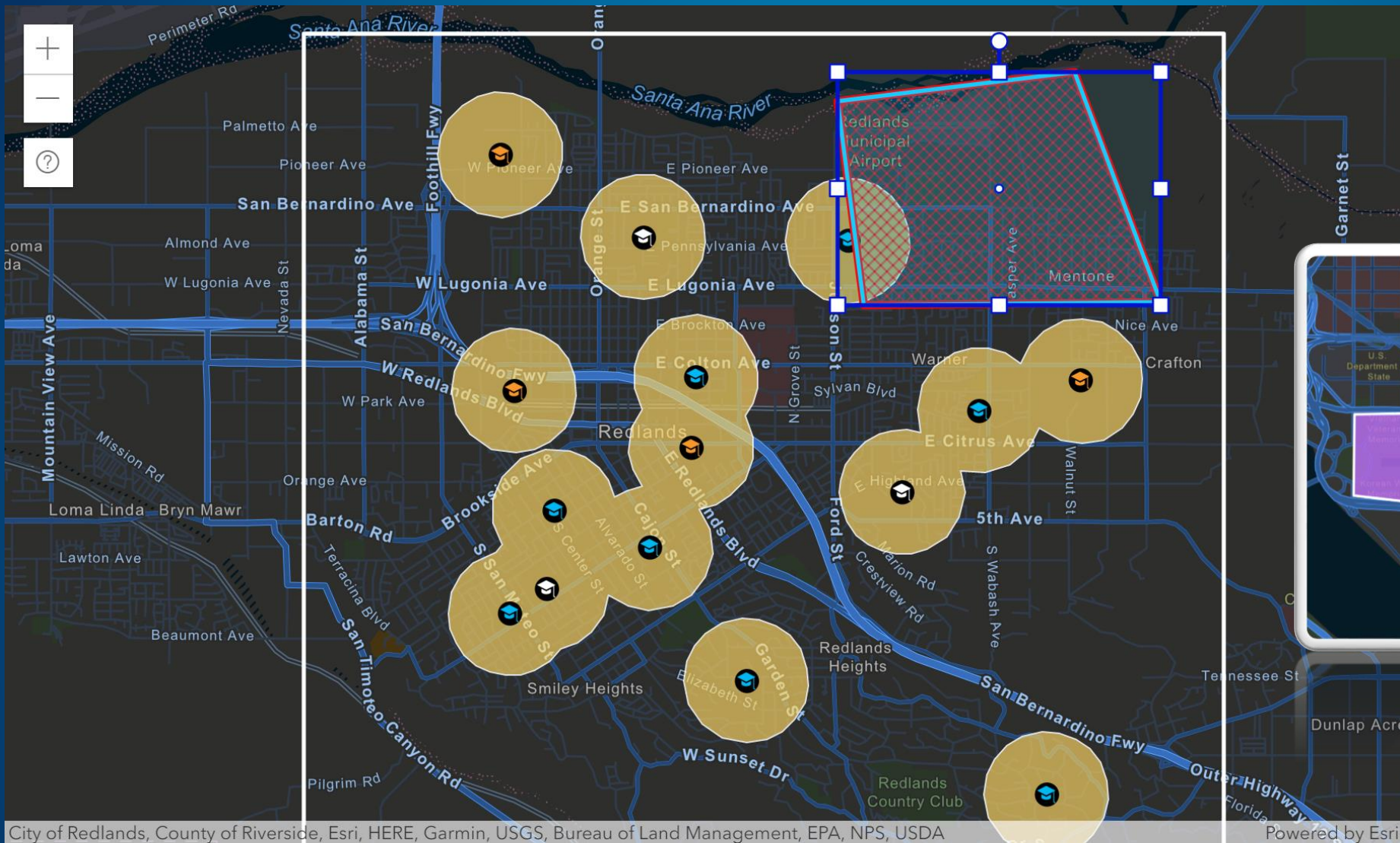
CLIENT-SIDE PROJECTION

- Same engine as ArcGIS Pro
- Uses WebAssembly



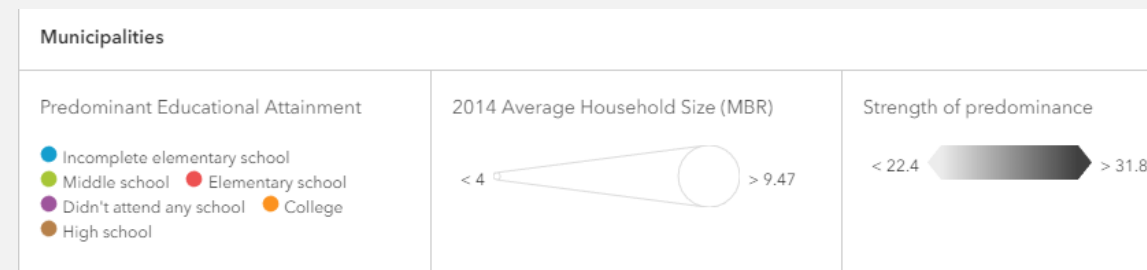
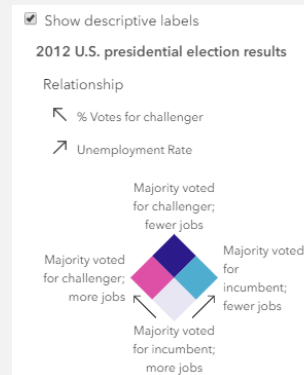
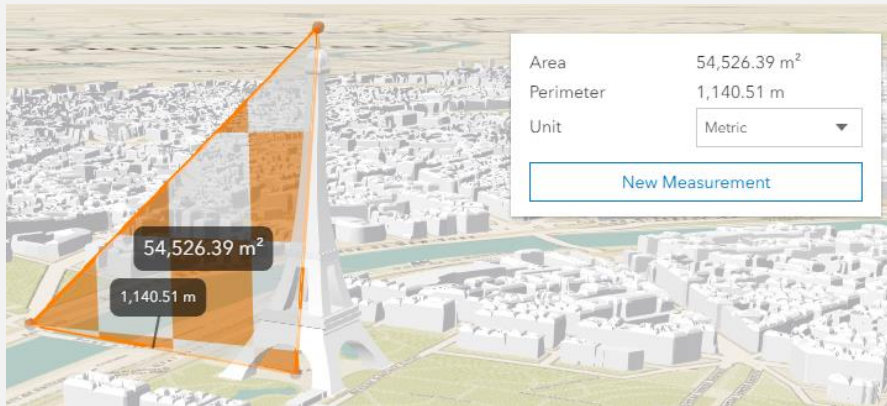
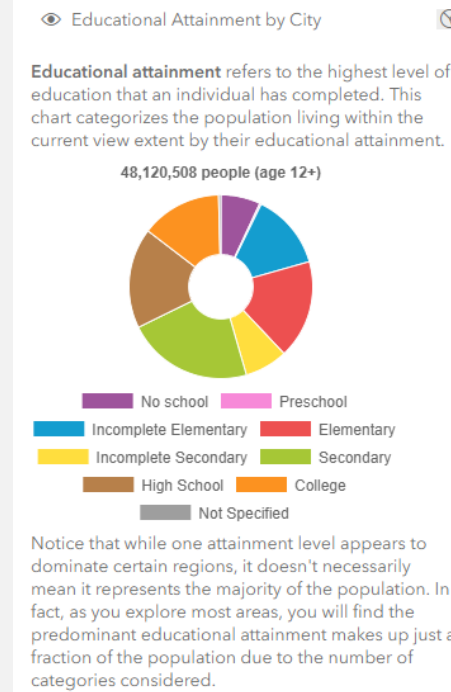
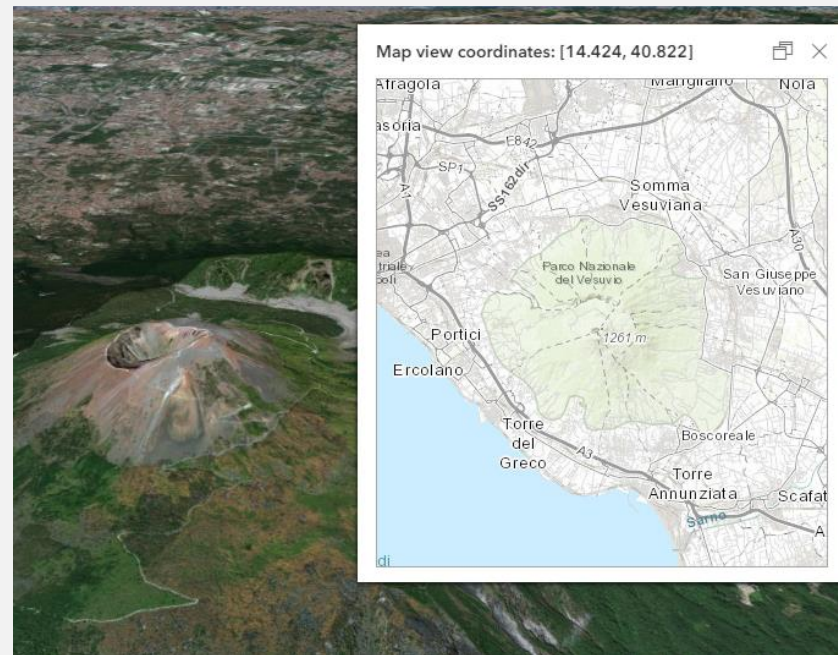
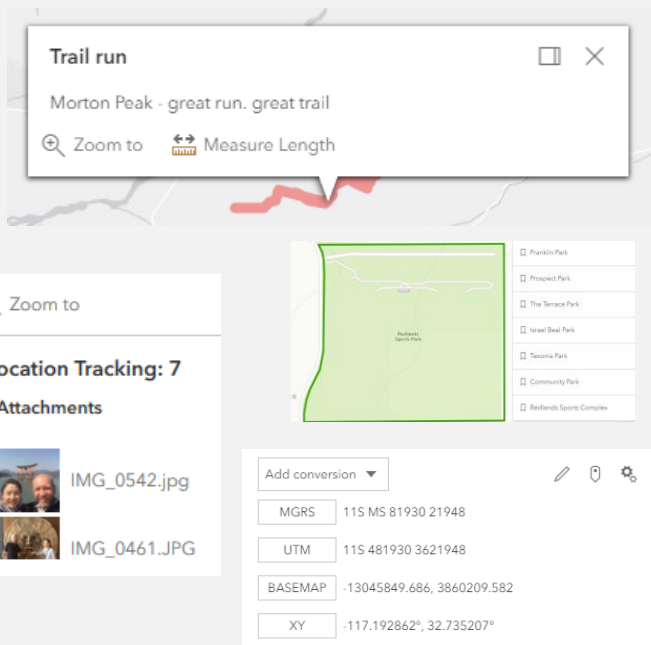
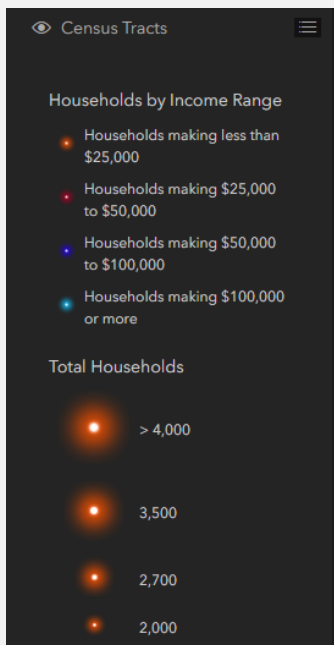
Building a great UX

Drawing & Widgets



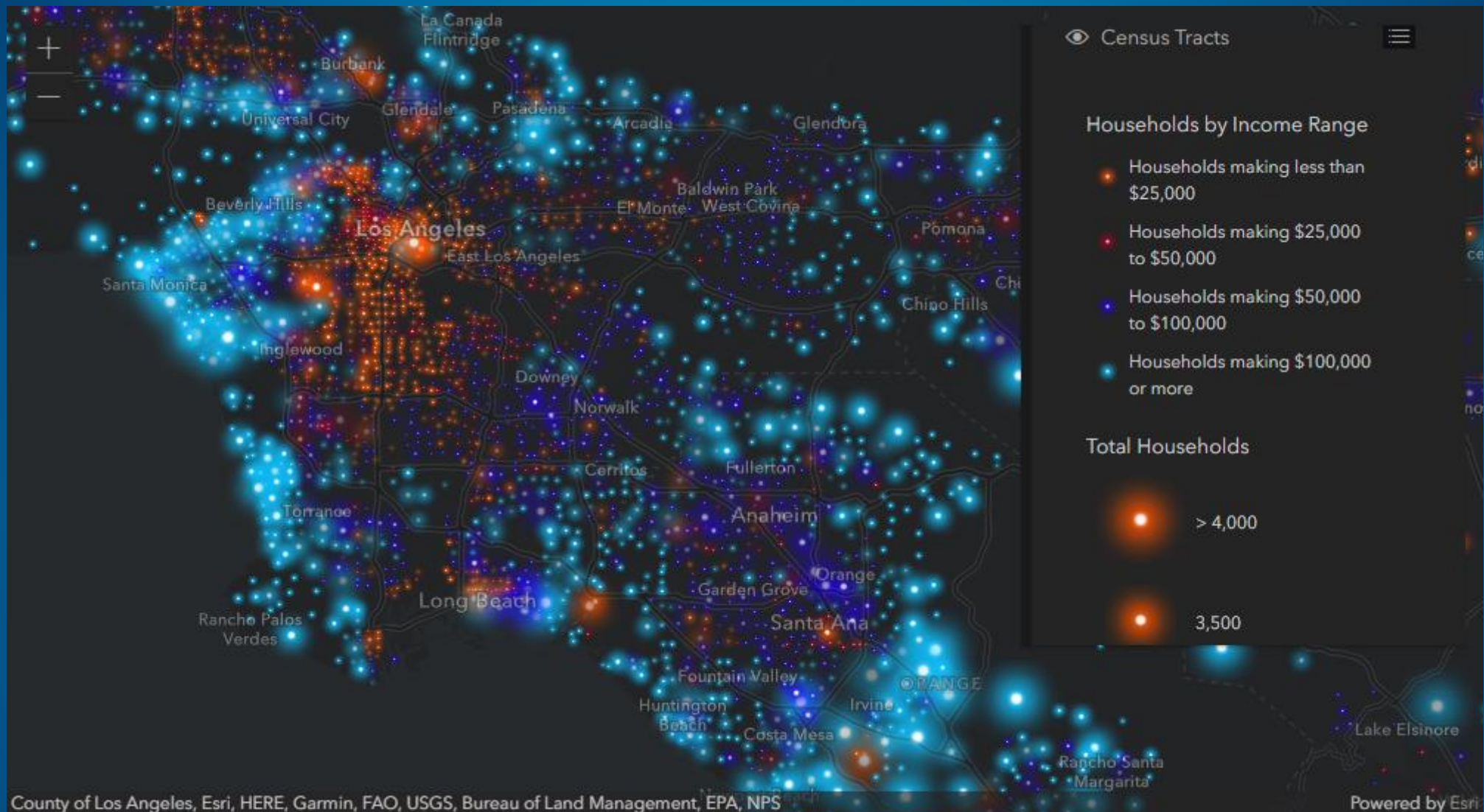
DRAW & SKETCHVIEWMODEL

Draw is the basic building block.
SketchViewModel is one way to put it
together.



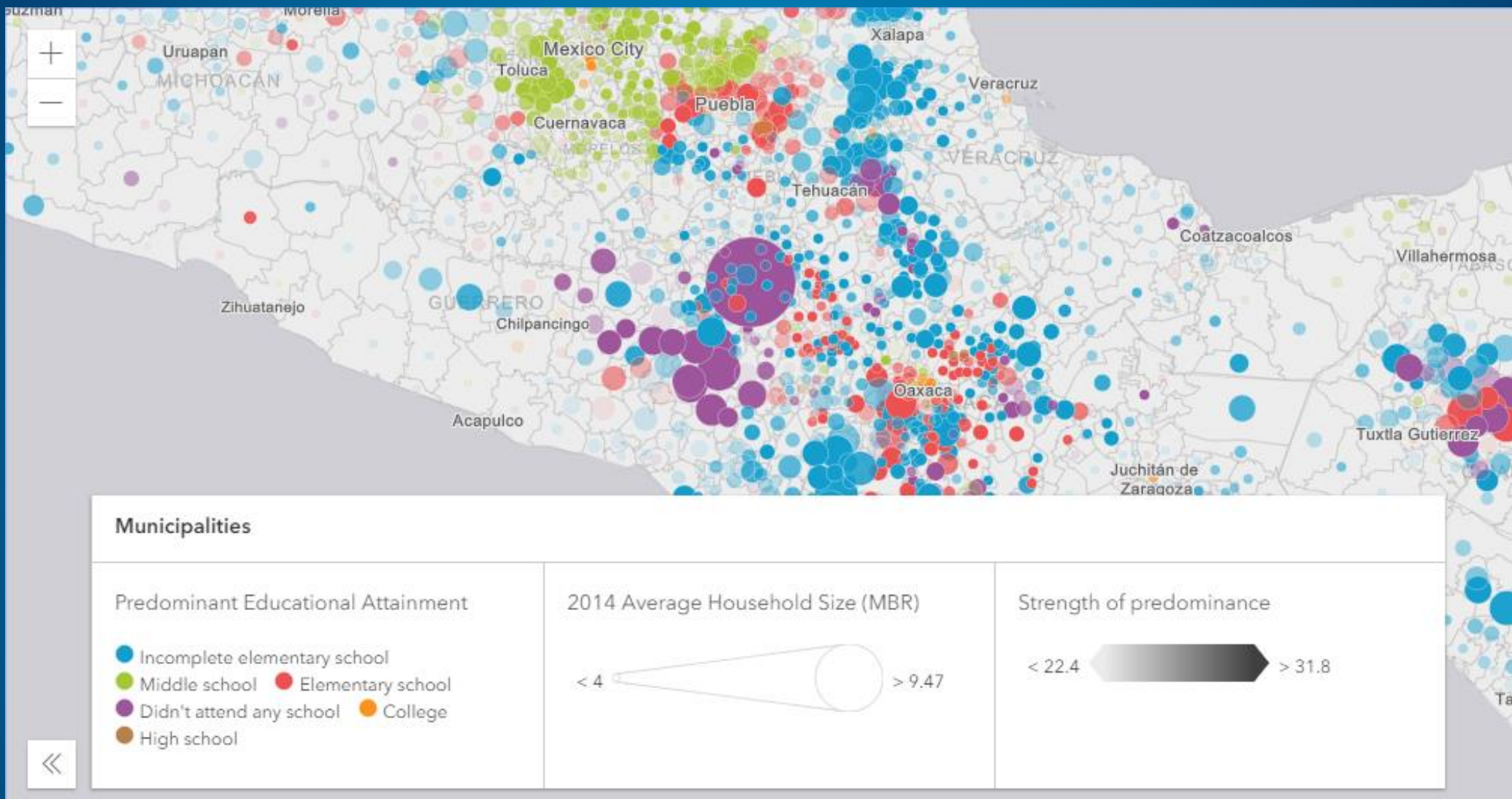
WIDGETS

LayerList, Legend, CoordinateConversion, AreaMeasurement3D, Popup, Bookmarks



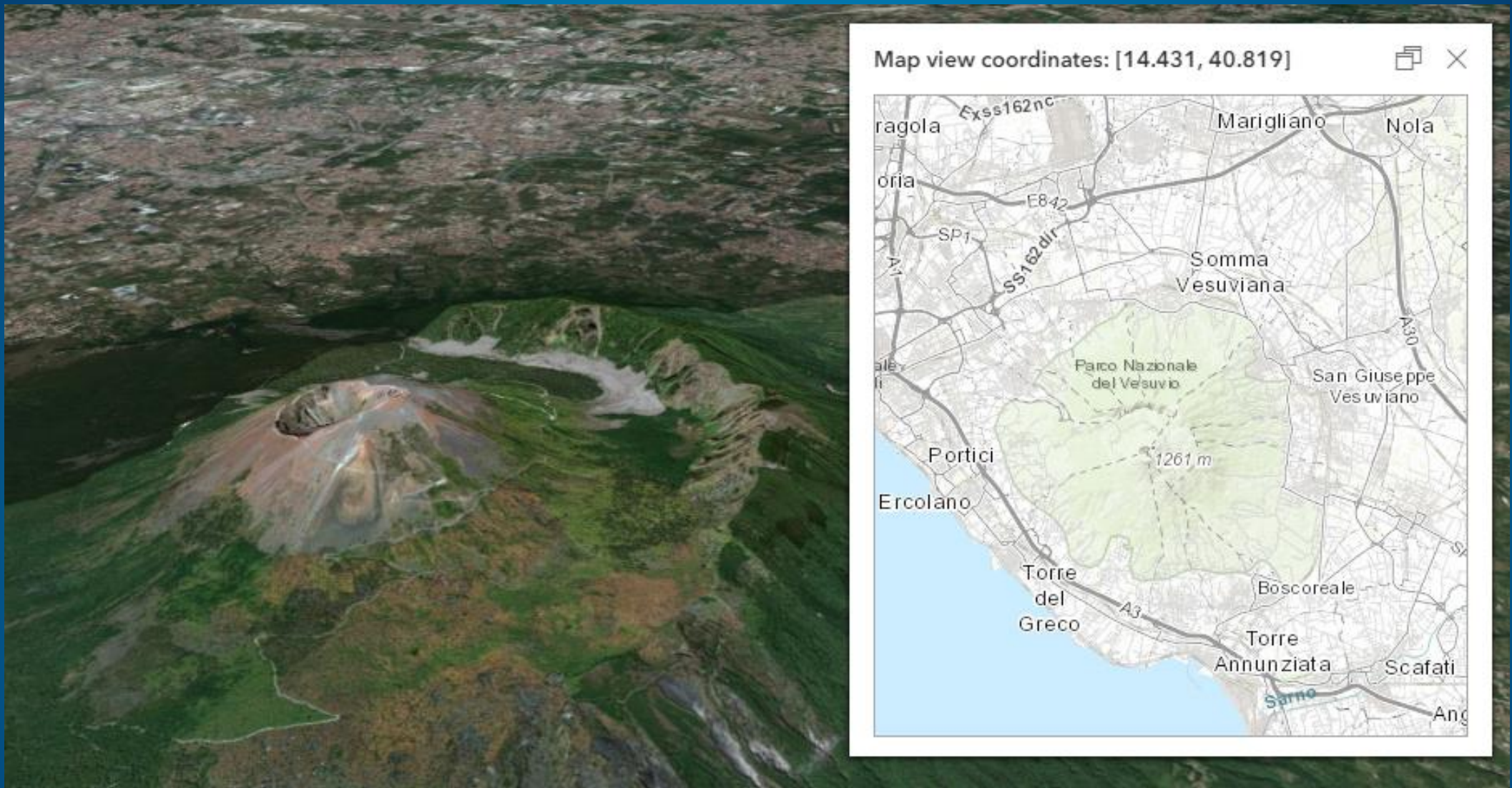
COMBINE LAYERLIST AND LEGEND WIDGETS

Add a Legend widget to a LayerList widget to create a single UI element.



LEGEND WIDGET CARD STYLE

A responsive style that renders the legend with a landscape (horizontal) layout in large views, and in a more compact card layout in small views.



POPUP WITH DOM NODE

Populate the content of a Popup using a function that returns a DOM node.



3D: expanding the reach

Better underground, 3D mobile, and more



MOBILE SUPPORT IN 3D

Officially supported devices:

- iOS — iPhone 8, iPad Pro (Safari browser)
- Android — Samsung S8, Samsung Tab S3 (Chrome browser)



> General

✓ Best practices

Best practices for using layers in maps

Best practices for sharing

Best practices for scene performance

> Account

> Content

> Use URL parameters

> Terms of use

Best practices for scene performance

Good scene performance enables your audience, such as your organization or the public, to experience your scenes the way you intended and with the most impact. You can optimize your scenes using the information here as a guide in the event you see compromised scene performance, such as slow layer loading, stuttering when navigating, or overall slowness. Scene performance optimization can be broken down into the following categories:

- Browser and hardware settings
- Viewing scenes
- Authoring scenes
- Creating scene content

In this topic

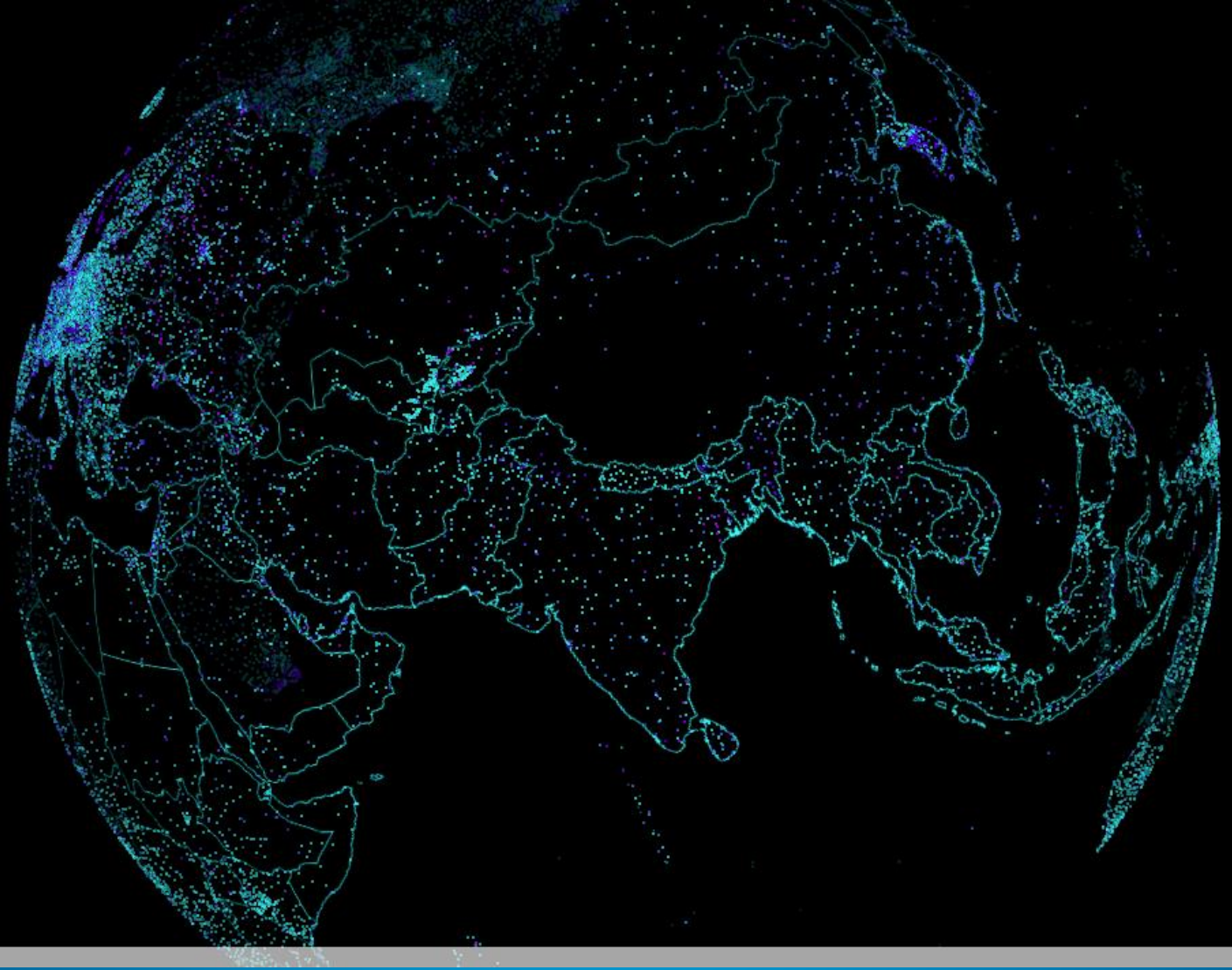
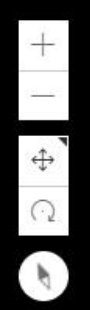
[Browser and hardware settings](#)[Viewing scenes](#)[Authoring scenes](#)[Creating scene content](#)

Browser and hardware settings

MOBILE SUPPORT IN 3D

Officially supported devices:

- iOS — iPhone 8, iPad Pro (Safari browser)
- Android — Samsung S8, Samsung Tab S3 (Chrome browser)



World airports

Powered by Esri

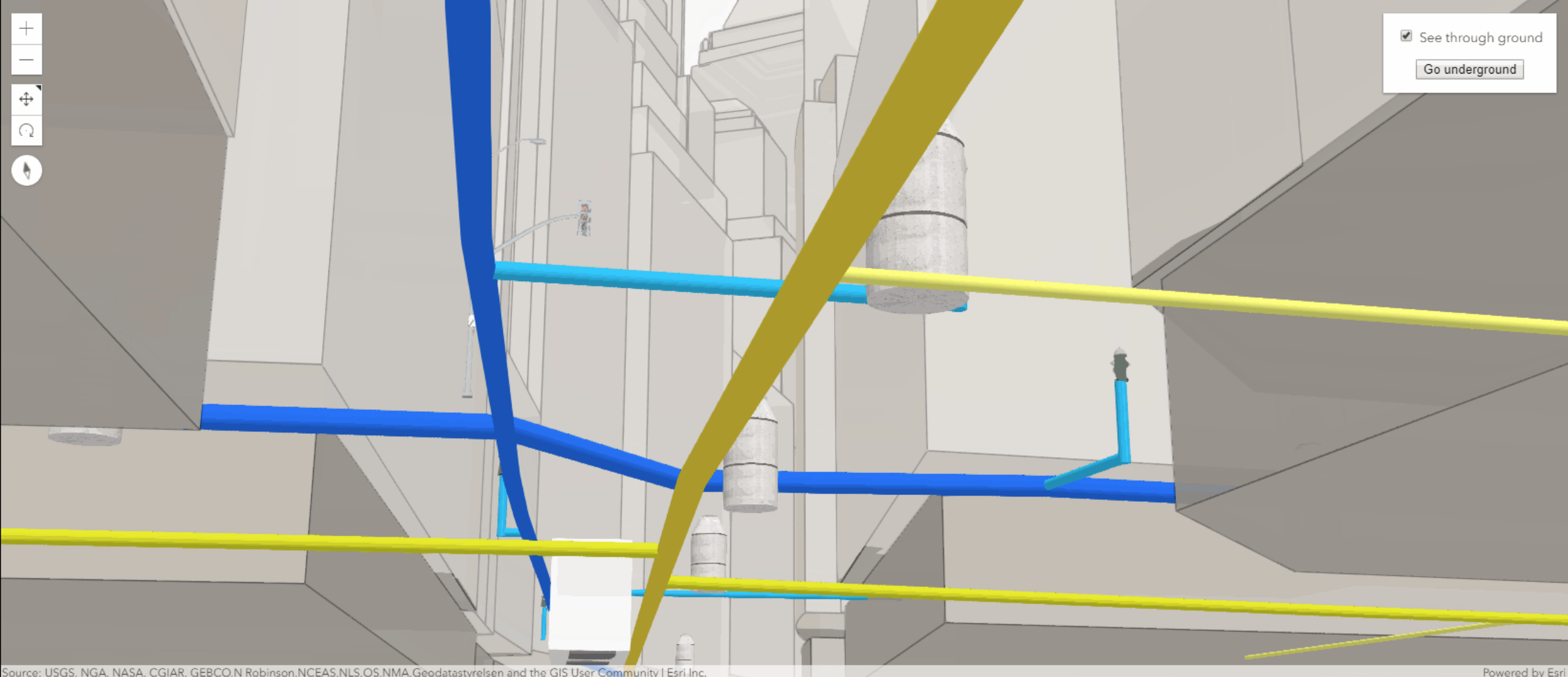
INTERACT WITH MORE FEATURES

- Point feature layers
- Dynamic loading & display
- Point scene layers



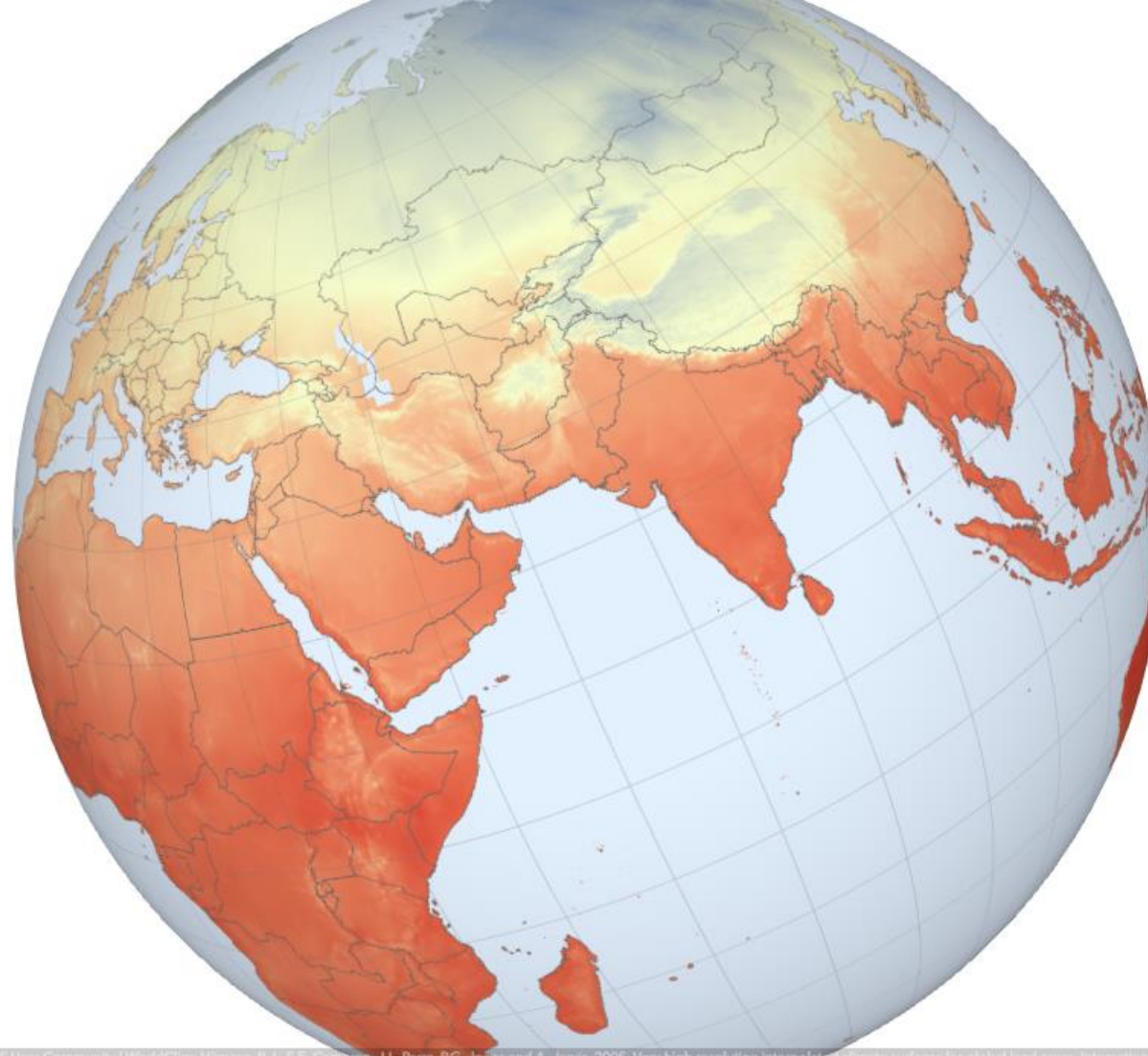
EDGE RENDERING

- For 3D Object SceneLayers (i.e. buildings)
- “Sketch” and solid styles



UNDERGROUND NAVIGATION

- Now global scenes
- Ground partial transparency
- Control the ground color (great when not using a basemap)



Layers Legend

WorldGeo Physical Climate Mean Temp January

Degrees Celsius

39.4

0

-53.6

World Countries (Generalized)

Graticule

Graticule

10-degree grid

CUSTOM BACKGROUND COLOR

Transparent or set to a custom color

GIS API for JavaScript

[Guide](#)
[API Reference](#)
[Sample Code](#)
[Community](#)

It Started

Upgrading from 3.x

Working with the API

ties

ses

3

ng

U

casting

ble

```
fromJSON()
```

Script Setup

Product Development

menting Accessor

Using npm for Custom Builds

Using **npm** (A package manager for JavaScript) is another option for developers of the JavaScript API to generate custom builds of the ArcGIS API for JavaScript. You would use the npm package of the ArcGIS API for JavaScript if you wanted to create local builds of your application.

Please note, npm is now the recommended way to install the ArcGIS API for JavaScript into a local project. Previously, we recommended using [Bower](#), but now we recommend moving to npm to take advantage of more web development tools like [Webpack](#).

Note: See the [prerequisites](#) section on configuring your system before proceeding. If your system is already configured with GNode.js, and Java you may proceed to the [Getting Started](#) section.

- Prerequisites
- Getting started
- Understanding npm
- Generate a custom build (Dojo)

Prerequisites

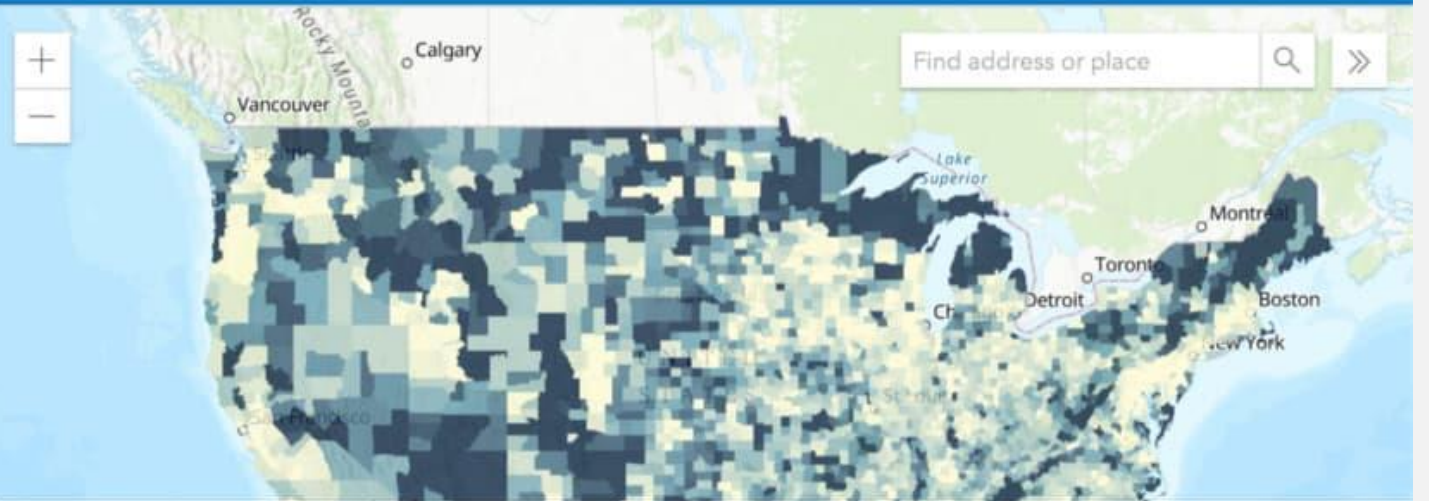


```
Total: [////////////////////] 29/29
Passed: 29 Failed: 0 Skipped: 0

Chr 63 Mac: [////////////////////] 29/29
```

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
All files	94.47	63.24	93.48	94.7	
src/app	84.42	77.78	75	83.58	
Application.ts	83.56	77.78	75	82.54	... 190,191,192
config.ts	100	100	100	100	
src/app/mapactions	96.77	61.54	100	96.08	
MapAction.ts	100	66.67	100	100	25
navRotation.ts	95.65	66.67	100	95	39
reverseGeocode.ts	95.65	50	100	94.44	45
src/app/utills	100	75	100	100	
dateUtils.ts	100	75	100	100	51
src/app/widgets	92.31	50	92	91.25	
Alert.tsx	95.24	50	100	94.74	96
Authenticate.tsx	95.65	37.5	83.33	95.24	104
UserNav.tsx	90.91	66.67	100	90	57,58
WebMapBrowser.tsx	88	50	85.71	85	53,54,55
src/app/widgets/Authenticate	75.41	50	85.71	79.17	
AuthenticateViewModel.ts	75.41	50	85.71	79.17	... 161,183,190
src/app/widgets/Authenticate/components	100	75	100	100	
AuthComponents.tsx	100	75	100	100	39,42

ARCGIS DEMO APP



```
λ arcgis create awesome-app
✓ ArcGIS Application template installed.
🌐 Installing dependencies, this could take a while...
Map: USGS, Esri, Garmin, FAO, NOAA, EPA
```

CUSTOM BUILDS AND DEV TOOLS

npm, webpack, cli

ArcGIS API for JavaScript

[Home](#) [Guide](#) [API Reference](#) [Sample Code](#) [Community](#)

- > Get Started
- > Migrating from 3.x
- ✓ Working with the API

Properties

Promises

Arcade

Labeling

View UI

Autocasting

Loadable

Using fromJSON()

TypeScript Setup

Widget Development

Implementing Accessor

Working with the ArcGIS platform

Access Secure Resources

Using Frameworks

Using npm

Using npm for Custom Builds

Using [npm](#) (A package manager for JavaScript) is another option for developers of the JavaScript API to generate custom builds of the ArcGIS API for JavaScript. You would use the npm package of the ArcGIS API for JavaScript if you wanted to create local builds of your application.

Please note, npm is now the recommended way to install the ArcGIS API for JavaScript into a local project. Previously, we recommended using [Bower](#), but now we recommend moving to npm to take advantage of more web development tools like [Webpack](#).

Note: See the [prerequisites](#) section on configuring your system before proceeding. If your system is already configured with Git, Node.js, and Java you may proceed to the [Getting Started](#) section.

- [Prerequisites](#)
- [Getting started](#)
- [Understanding npm](#)
- [Generate a custom build \(Dojo\)](#)

Prerequisites

1. [Node.js](#) (includes npm, the node package manager)
2. [git](#)
3. [Java 7 or greater](#) - required by [Closure Compiler](#) (Dojo build)

To check if these are installed:



Search packages

Share your code. npm Orgs help your team discover, share, and reuse code. [Create a free org »](#)

arcgis-js-api

4.7.2 • [Public](#) • Published 2 months ago

[Readme](#)

14 Dependencies

9 Dependents

7 Versions

arcgis-js-api

A minified, unbuilt version of the [ArcGIS API for JavaScript](#).

Features

A minified, unbuilt version of the [ArcGIS API for JavaScript](#). You can install this repo via [npm](#) and create your own custom builds with [Webpack](#) or the [Dojo Toolkit](#).

Instructions

Building an ArcGIS API for JavaScript application requires signing up for an [ArcGIS account](#).

```
npm install arcgis-js-api
```

install

```
> npm i arcgis-js-api
```

weekly downloads

113

version

4.7.2

license

SEE LICENSE IN ...

open issues

0

pull requests

0

homepage

developers.arc... github

last publish

2 months ago

USING NPM FOR CUSTOM BUILDS

Use the npm package of the ArcGIS API for JavaScript if you wanted to create local builds of your application.

ArcGIS API for JavaScript

[Home](#) [Guide](#) [API Reference](#) [Sample Code](#) [Community](#)

- > Get Started
- > Migrating from 3.x
- ▼ Working with the API

Properties

Promises

Arcade

Labeling

View UI

Autocasting

Loadable

Using fromJSON()

TypeScript Setup

Widget Development

Implementing Accessor

Working with the ArcGIS platform

Access Secure Resources

Using Frameworks

Using npm

Using webpack

Using webpack for Custom Builds

[webpack](#) is a module bundler for JavaScript. It can basically process the code in your application and bundle it up, not only into a single file, but bundled files that can be dynamically loaded as needed at runtime, for example, in an application that might have multiple routes. You can use webpack to build applications that use the ArcGIS API for JavaScript by using the [@arcgis/webpack-plugin](#).

To learn more about webpack, please refer to the [concepts](#) and [guides](#) pages from webpack.

Note: See the [prerequisites](#) section on configuring your system before proceeding. If your system is already configured with Node.js, you may proceed to the [Getting Started](#) section.

- [Prerequisites](#)
- [Getting started](#)
- [Understanding webpack](#)
- [Building your app](#)

Prerequisites

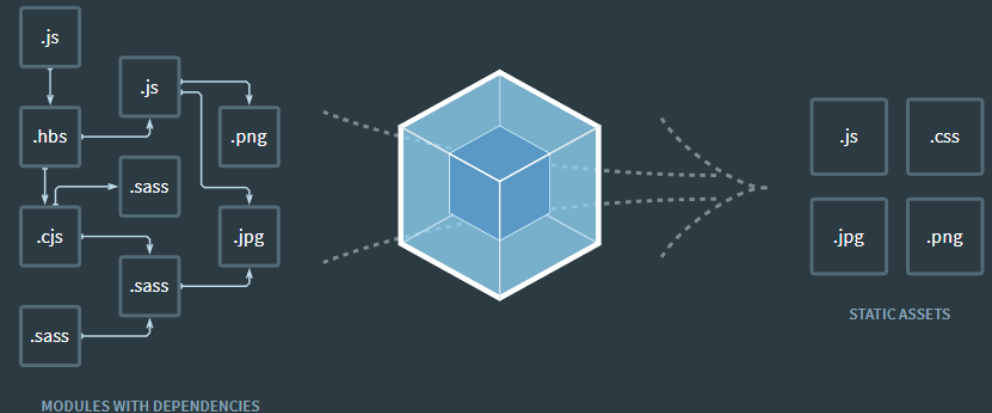
1. [Node.js](#) version `8.x.x` or higher (includes npm, the node package manager)

To check if these are installed:

```
node --version
```

[DOCUMENTATION](#) [CONTRIBUTE](#) [VOTE](#) [BLOG](#) [SEARCH](#) [GITHUB](#) [TWITTER](#)

bundle your assets



USING WEBPACK TO BUNDLE YOUR ASSETS

Use webpack to process your code and bundle it up (into one or more files) to be dynamically loaded at runtime.

CLI to build a template application and widgets using the ArcGIS API for JavaScript

javascript arcgis-js-api cli flow

22 commits

1 branch

3 releases

0 contributors

Apache-2.0

Branch: master
















New pull request

Create new file

Upload files

Find file

Clone or download

 rene rubalcava	refactor cli modules, update terminal outputs	Latest commit b95c228 a day ago
 .github	cli party	2 months ago
 __mocks__	refactor cli modules, update terminal outputs	a day ago
 flow-typed/npm	cli party	2 months ago
 src	refactor cli modules, update terminal outputs	a day ago
 templates	template app - remove service worker from dev builds, only in prod	a day ago
 .babelrc	cli party	2 months ago
 .eslintrc	cli party	2 months ago
 .flowconfig	update template application	2 months ago
 .gitignore	cli party	2 months ago
 .travis.yml	add badges and travis yml	a month ago
 CONTRIBUTING.md	cli party	2 months ago
 LICENSE	cli party	2 months ago
 README.md	add badges and travis yml	a month ago
 package.json	4.7.5	3 days ago

You can install it via `npm` with following command.

```
npm install -g @arcgis/cli
```

Once installed you can now use the `@arcgis/cli` to scaffold a new application, using the `create` command and passing an application name.

```
arcgis create awesome
```

```
~/Sites/demos
λ arcgis create awesome
✓ ArcGIS Application template installed.
✓ Done! You're ArcGIS JSAPI application has been installed!

arcgis create awesome 51.60s user 35.57s system 107% cpu 1:21.32
```

AWESOME

node_modules

src

assets

★ favicon.ico

🖼 icon.png

css

🔒 .gitkeep

🔗 index.scss

🔗 main.scss

data

📄 app.ts

widgets

App

App.tsx

Header.tsx

config.ts

index.html

index.ts

oauth-callback.html

tests

unit

widgets

App

AppViewModel.ts

App.ts

Header.ts

all.ts

tsconfig.json

.gitignore

intern.json

package-lock.json

package.json

README.md

tsconfig.json

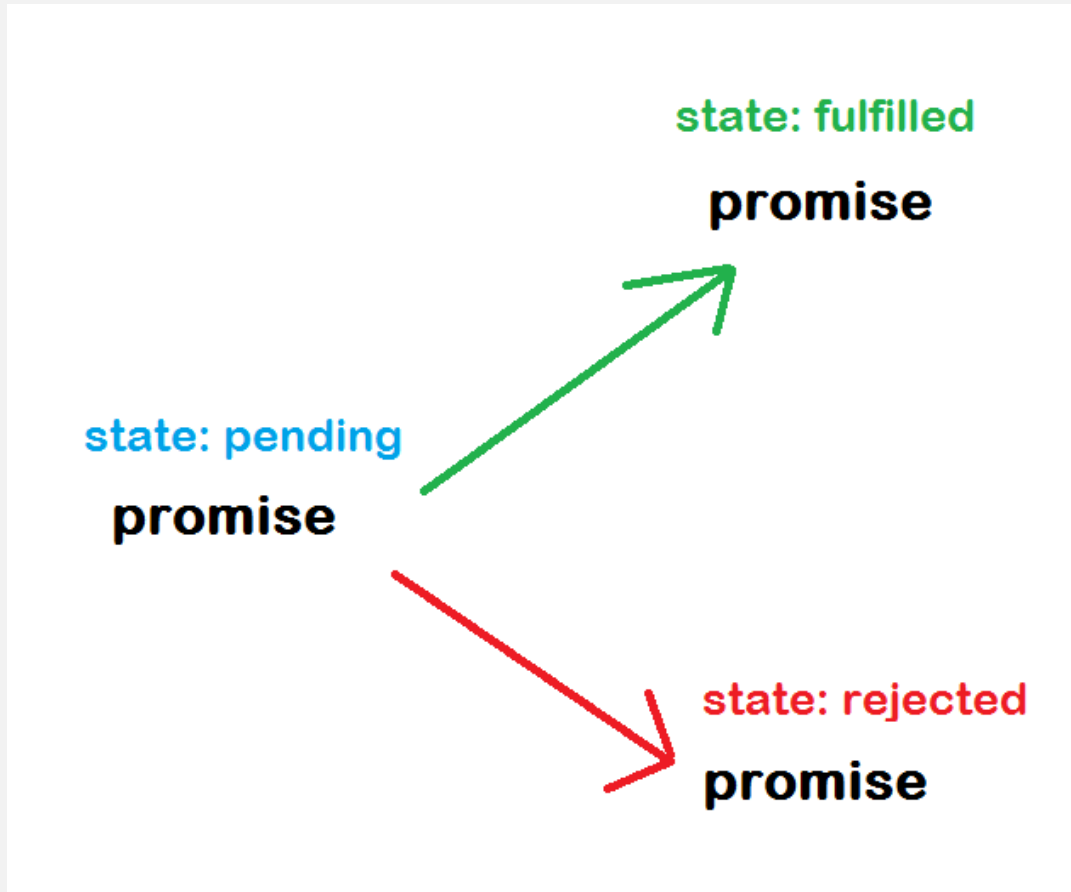
tslint.json

webpack.config.js

webpack.tests.config.js

ADD POWER WITH THE ARCGIS CLI

Scaffold new applications, custom widgets, and include unit testing, without having to click a button.



Version	then()	when()	always()	otherwise()
4.0 - 4.5	Available	Not Available	Available	Available
4.6	Deprecated	Available	Available	Available
4.7	Not Available	Available	Deprecated	Deprecated
4.8	Not Available	Available	Not Available	Not Available

ArcGIS Web API / JavaScript API / 4.8 / Guide

ArcGIS API for JavaScript

Home Guide API Reference Sample Code Community

Working with promises

Promises play an important role in the ArcGIS API for JavaScript. The better you understand promises, the more equipped you'll be to write cleaner code when working with asynchronous operations in your applications.

What is a promise?

On the most basic level, a promise is a representation of a future value returned from an asynchronous task. When the task executes, the promise allows other processes to run simultaneously, which gives users a better experience. In essence, a promise is a value that "promises" to be returned whenever the process completes. This is particularly useful when making multiple network requests where timing and download speeds can be unpredictable.

A promise is always in one of three states: *pending*, *resolved*, or *rejected*. When a promise resolves, it can resolve to a value or another promise as defined in a `callback` function. When a promise is rejected, it should be handled in an `errback` function.

MAKING BETTER PROMISES

A promise is an object that represents a potential future value.

The background is a solid blue gradient. In the top-left corner, there are several overlapping, semi-transparent geometric shapes in shades of teal, green, and red. In the bottom-right corner, there is a large, complex geometric shape composed of many smaller, overlapping rectangles and polygons in various colors including orange, yellow, green, and blue. The text "More resources" is positioned in the lower-left area of the image.

More resources



esri

THE
SCIENCE
OF
WHERE