ArcGIS API for JavaScript: What's New

Bjorn Svensson, Julie Powell, Noah Sager
Release notes for 4.8

- 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
12 results for Sample Code: 4.8

Tags
4.8

Bookmarks widget
Visualize points with a heatmap
Custom background for SceneView
Update legend text

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

SDK IMPROVEMENTS

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

Clustering, heatmap, smartmapping, Arcade expressions, labeling, ...
The highest park of Redlands. Great views of the mountains.
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
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
Feature tile queries and response caching

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

Public Layer

Feature layer ➔ Browser cache ➔ CDN cache ➔ Feature tile cache ➔ Data

1. Query is sent to the feature layer
2. Does the browser cache have the latest data?
   - No ➔ Does the CDN cache have the latest data?
     - No ➔ Does the feature tile cache have the latest data?
       - No ➔ Execute a new query.
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
• Done on FeatureLayerView
• 4.8 added full spatial queries & expressions for fields & stats
function checkGraphicUpdate(event) {
    // update the new development graphic geometry to match the update geometry
    newDevelopmentGraphic.geometry = event.geometry;

    // Check if the new development graphic geometry intersects school buffers
    // or the boundary polygon. If it does, change the graphic symbol to indicate
    // the update is invalid.
    Intersects = geometryEngine.intersects(buffers,
        newDevelopmentGraphic.geometry);
    contains = geometryEngine.contains(boundaryPolygon,
        newDevelopmentGraphic.geometry);
    newDevelopmentGraphic.symbol = (Intersects || (!contains) ?
        invalidSymbol : validSymbol)
}
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.
Add a Legend widget to a LayerList widget to create a single UI element.
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)

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

Browser and hardware settings

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

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

npm, webpacks, cli, promises
CUSTOM BUILDS AND DEV TOOLS

npm, webpack, cli
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.
USING WEBPACK TO BUNDLE YOUR ASSETS

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
```
ADD POWER WITH THE ARCGIS CLI

Scaffold new applications, custom widgets, and include unit testing, without having to click a button.
A promise is an object that represents a potential future value.
More resources