

Adding GeoJSON Support to ArcGIS Pro with KoopJS

Ryan Cooper

Parks, Recreation and Cultural Resources

City of Raleigh

@maptastik | ryancooper729@gmail.com

GISDay - 11/18/2020

What is GeoJSON?

What is GeoJSON?

- JavaScript Object Notation
- Data stored as name/key-value pairs
- Text-based and Human-readable
- I/O supported by multiple languages
- Common server/API response format

PARKID	NAME	ACRES
4	Lions	41.4
111	Lockwood	3.8
185	Abbotts Creek	29.1



```
{  
  PARKID: 4,  
  NAME: "Lions",  
  ACRES: 41.4  
}
```

What is GeoJSON?

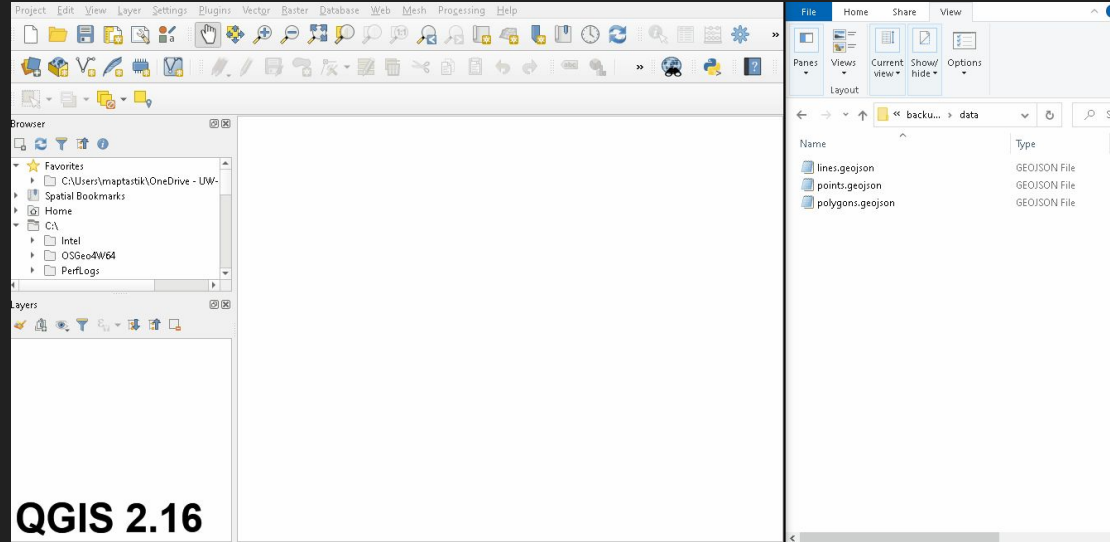
- JSON structure for vector geodata
- IETF standard (RFC 7946)
- Supports Point, LineString, Polygons, Multi-types
- filename.geojson

PARKID	NAME	ACRES	GEOMETRY
4	Lions	41.4	POINT(-78.61,35.8)
111	Lockwood	3.8	POINT(-78.6,35.8)
174	Abbotts Creek	29.1	POINT(-78.58,35.9)

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "PARKID": 4,
        "NAME": "Lions",
        "ACRES": 41.4
      },
      "geometry": {
        "type": "Point",
        "coordinates": [-78.61, 35.8]
      }
    },
    {
      "type": "Feature",
      "properties": {
        "PARKID": 11,
        "NAME": "Lockwood",
        "ACRES": 3.8
      },
      "geometry": {
        "type": "Point",
        "coordinates": [-78.6, 35.8]
      }
    },
    {
      "type": "Feature",
      "properties": {
        "PARKID": 185,
        "NAME": "Abbotts Creek",
        "ACRES": 29.1
      },
      "geometry": {
        "type": "Point",
        "coordinates": [-78.58, 35.9]
      }
    }
  ]
}
```

GeoJSON in ArcGIS Pro?

- No built-in method to directly access GeoJSON file
- Cannot “drag-and-drop” into map
- JSON to Features Tool provides indirect support (copies features to new feature class)



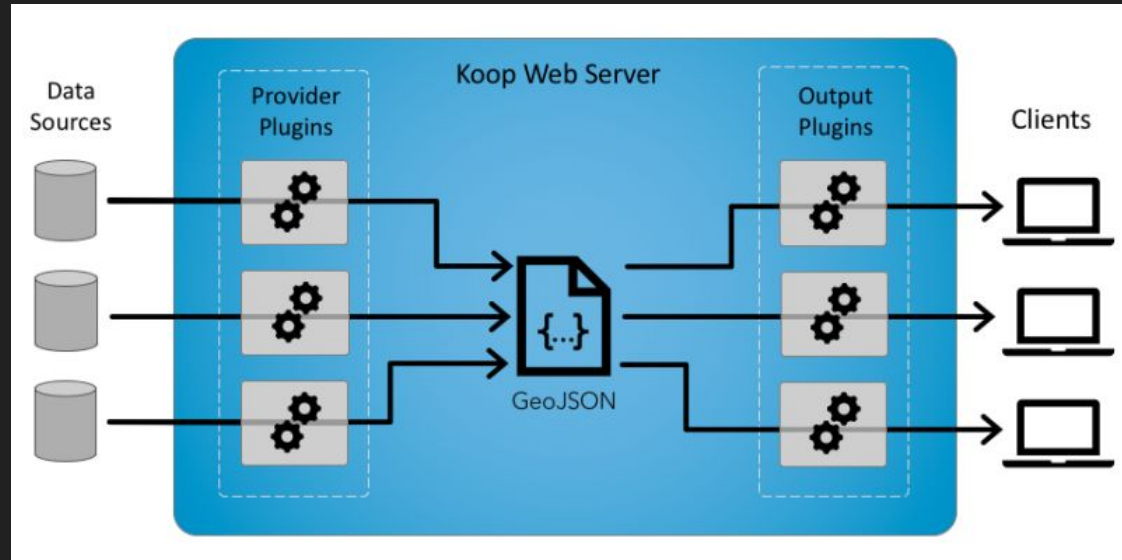
How can we fix this?



With Koop!

What is Koop?

- “[A] JavaScript toolkit for connecting spatial APIs”
- Translates input data to GeoJSON
- Serves FeatureService & MapService
- Esri, but Open Source
- Plugin architecture allows for support for a variety of input sources and output types



<https://koopjs.github.io>

Setup

- Prerequisites installs
 - Git
 - Node
 - Node Package Manager (npm)
- Install Koop CLI
 - Command-line tool for creating a new, basic Koop server
 - Open Node.js command prompt (Windows) or terminal (OSX, Linux)
 - `npm install -g @koopjs/cli`

Create a GeoJSON Koop server

1. Create a new Koop server
 - `koop new app geojson-services`
2. Install GeoJSON Provider plugin in geojson-services server
 - `npm install @koopjs/provider-file-geojson`
3. Register the GeoJSON Provider with the server
 - `/src/index.js`
 - `const geojsonProvider = require('@koopjs/provider-file-geojson')`
`koop.register(geojsonProvider)`
4. Create data directory in server root and add GeoJSON files
5. Start server
 - `npm start`
6. Add layer in Pro (Add Data>Data from path)
 - `http://localhost:8080/geojson-services/rest/services/<<file name>>/FeatureServer/0`

Links

- GeoJSON
 - [IETF 7946 - GeoJSON Specification](#)
 - [geojson.io \(Browser tool for authoring and viewing GeoJSON\)](#)
- KoopJS
 - [Official Website](#)
 - [Quick Start](#)
 - [koop-cli](#)
 - [Provider Plugin List](#)
 - [GeoJSON Provider Plugin](#)
- Miscellaneous
 - [Download and Install Git](#)
 - [Download and Install NodeJS/NPM](#)