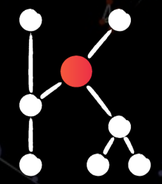


MAPS IN PRACTICE

KIRELL BENZI, PH.D

@KirellBenzi

www.kirellbenzi.com

Web mapping

Also known as Web GIS (geographic information systems)

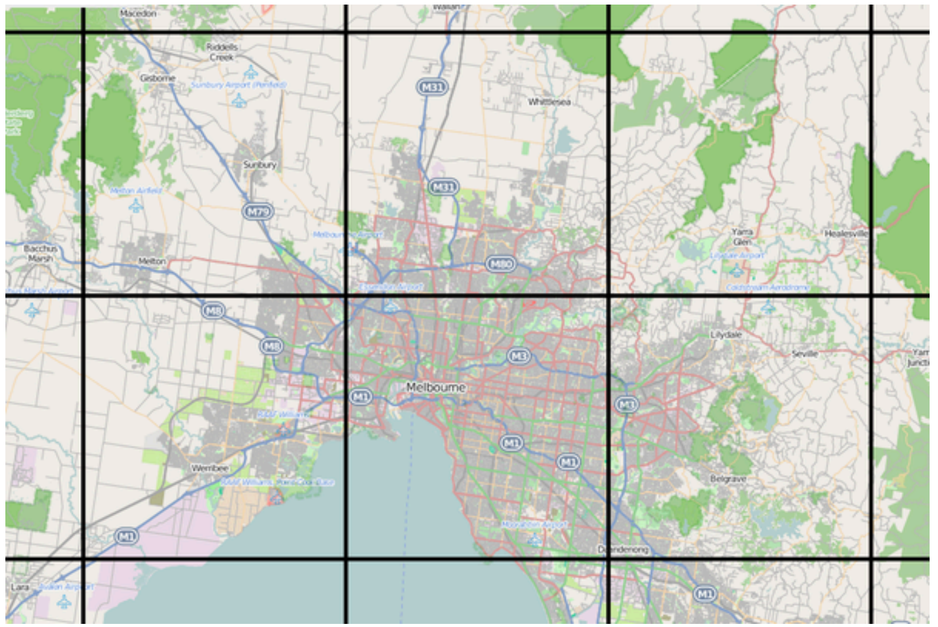
Client-server interaction to serve and consume a geographical dataset

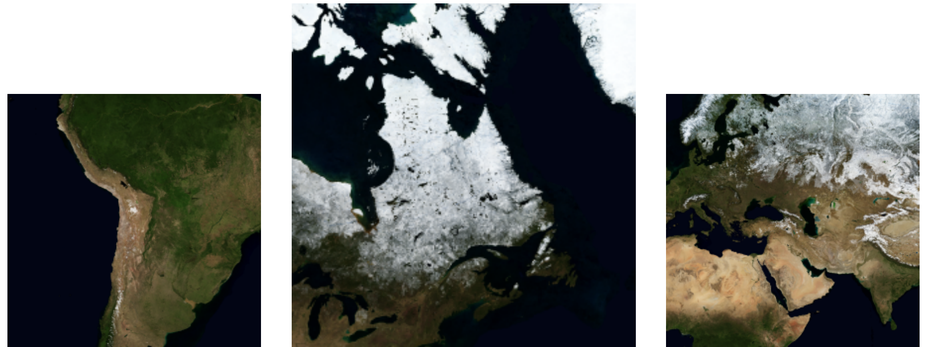
Examples: Google Maps, Apple Maps, Waze, OpenStreetMap, Here

Tiled web map

Cut a very high resolution image into a hierachy a squared tiled for each zoom level

Displayed in a browser by seamlessly joining dozens of individually requested image files on demand



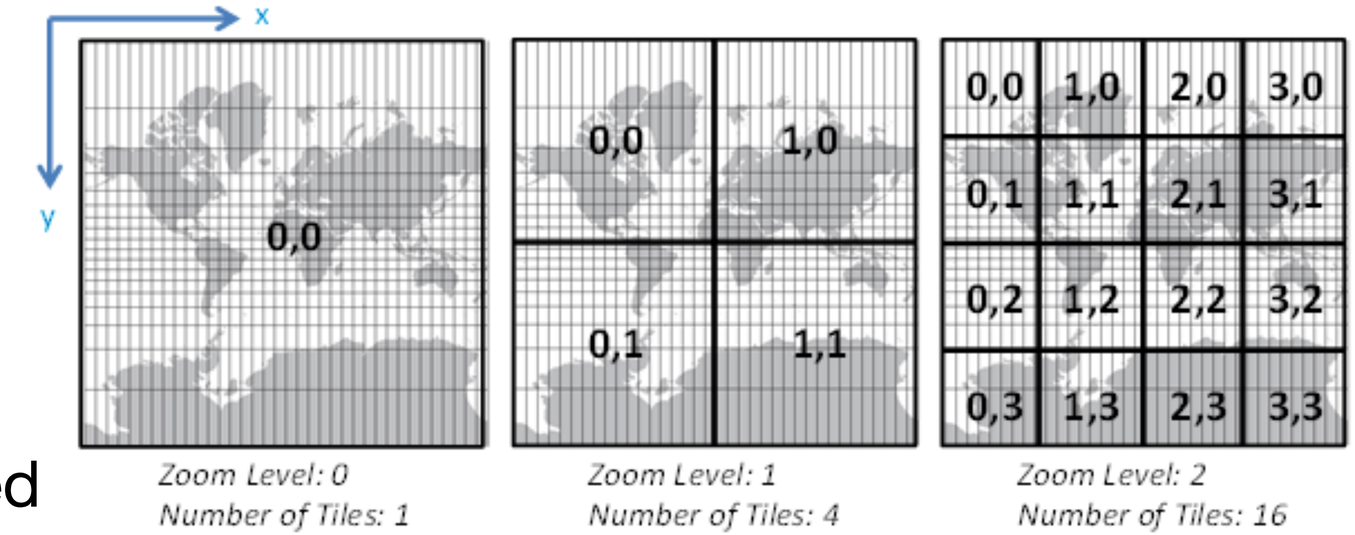


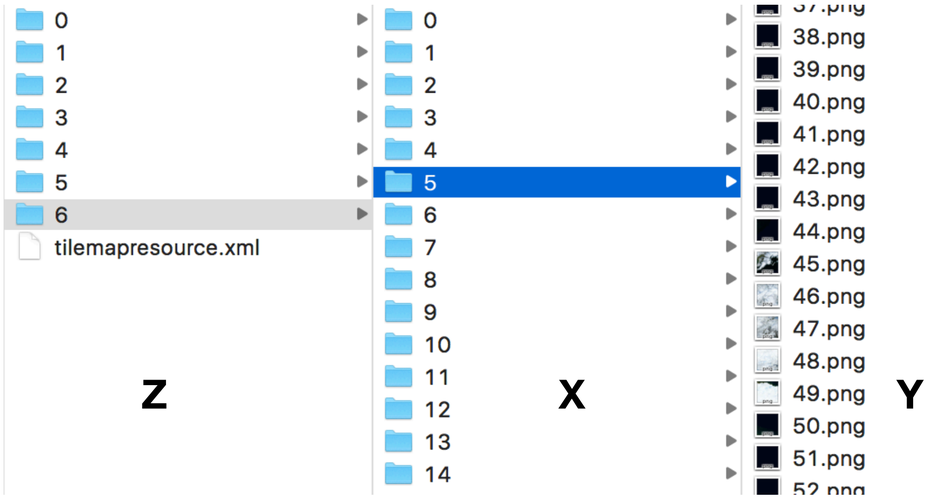
Tiled web map

Tiles are 256x256 pixels

Each zoom level doubles in both dimensions, so a single tile is replaced by 4 tiles when zooming in

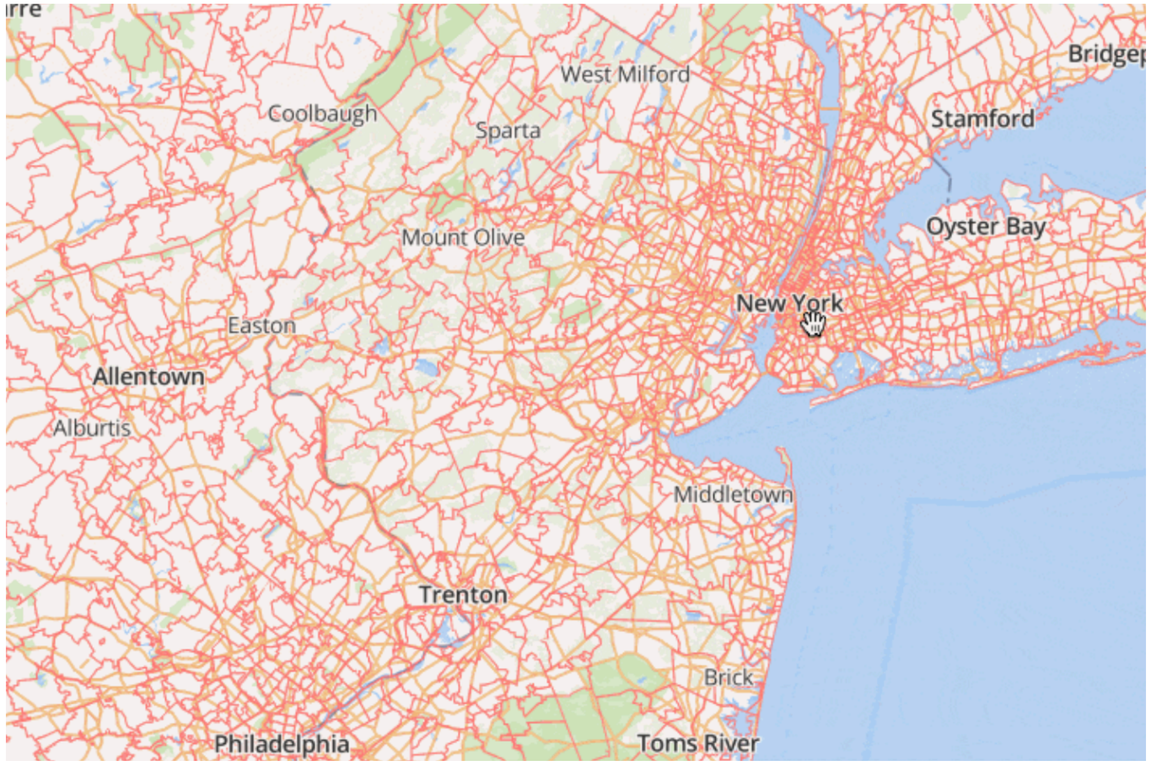
Uses Web Mercator projection: latitude limits of around 85 degrees





Vector tiles

- Newer format, replace bitmaps
- Contain geometries and metadata – like road names, place names, house numbers – in a compact, structured format
- Data transfer is greatly reduced
- Better customization
- Rasterisation can be performed directly in the client

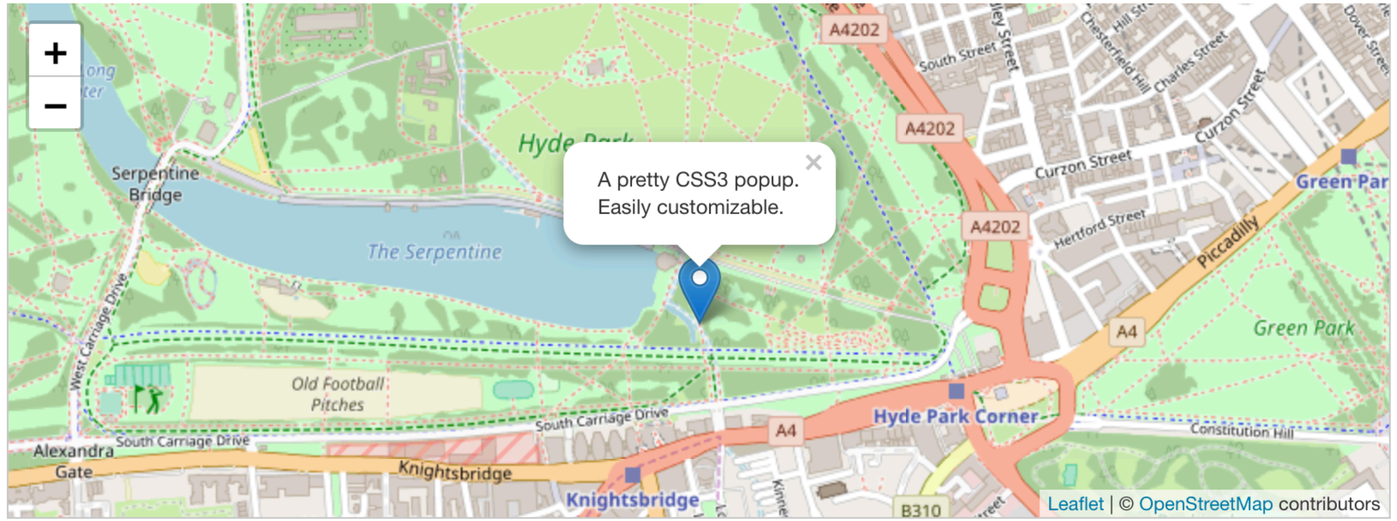


Libraries

GDAL, to create, convert, do anything with maps (C++, Python)

Leaflet.js, leading open-source JS library for interactive maps

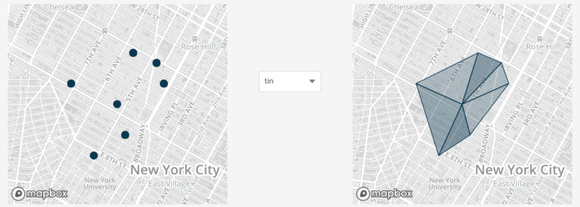
Turf.js advanced geospatial analysis




```
var map = L.map('map').setView([51.505, -0.09], 13);

L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {
    attribution: '&copy; <a href="http://osm.org/copyright">Open
}).addTo(map);

L.marker([51.5, -0.09]).addTo(map)
    .bindPopup('A pretty CSS3 popup.<br> Easily customizable.')
    .openPopup();
```



Advanced geospatial analysis for browsers and Node.js

Simple

Modular, simple-to-understand JavaScript functions that speak GeoJSON

Modular

Turf is a collection of small modules, you only need to take what you want to use

Fast

Takes advantage of the newest algorithms and doesn't require you to send data to a server

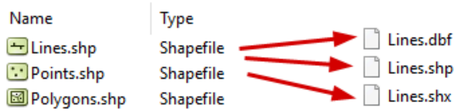
GIS formats

1 SHP (Shapefile)

The shapefile is BY FAR the most common geospatial file type you'll encounter. All commercial and open source accept shapefile as GIS formats. **It's become the industry standard**.

But you'll need a complete set of three files that are mandatory to make up a shapefile. The three required files are – SHP is the feature geometry, SHX is the shape index position and DBF is the attribute data.

You can optionally include these files but are not completely necessary. PRJ is the projection system metadata, XML is the associated metadata, SBN is the spatial index for optimizing queries and SBX helps loading times.



2 KMZ/KML (Keyhole Markup Language)

KML stands for Keyhole Markup Language. This GIS format is XML-based and is primarily used for Google Earth. KML was developed by Keyhole Inc which was later acquired by Google.

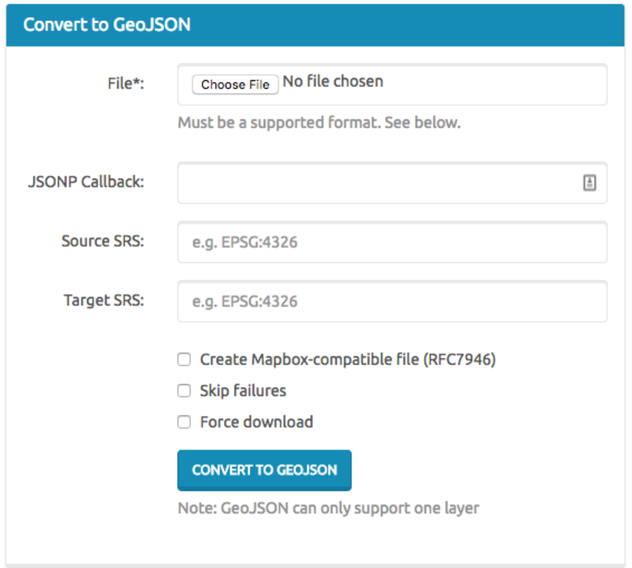
KMZ (KML-Zipped) replaced KML as being the default Google Earth geospatial format because it is a compressed version of the file. KML/KMZ became an international standard of the Open Geospatial Consortium in 2008.

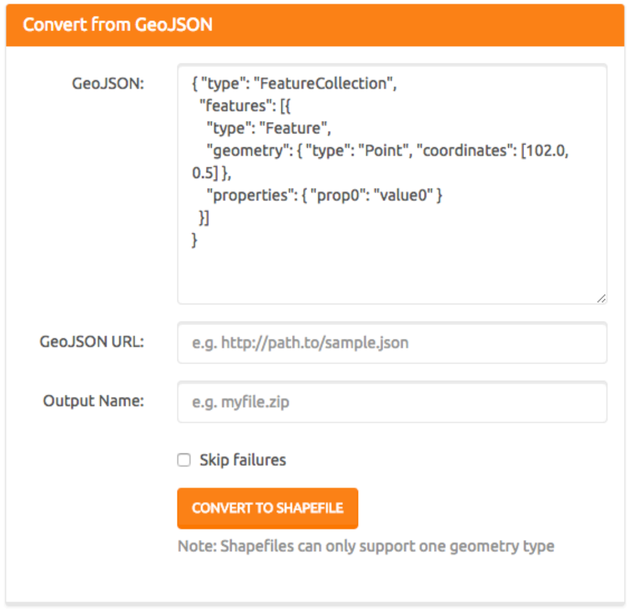
The longitude, latitude components (decimal degrees) are as defined by the World Geodetic System of 1984 (WGS84). The vertical component (altitude) is measured in meters from the WGS84 EGM96 Geoid vertical datum.

FireStations.kmz

OGRE

ogr2ogr web client



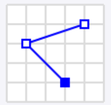
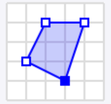
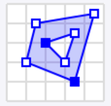


GeoJSON

GeoJSON is a format for encoding geographic data structures with their non-spatial attributes

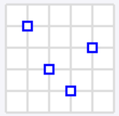
Open standard format

Based on JavaScript Object Notation (JSON).

Point		{ "type": "Point", "coordinates": [30, 10] }
LineString		{ "type": "LineString", "coordinates": [[30, 10], [10, 30], [40, 40]] }
Polygon		{ "type": "Polygon", "coordinates": [[[30, 10], [40, 40], [20, 40], [10, 20], [30, 10]]] }
		{ "type": "Polygon", "coordinates": [[[35, 10], [45, 45], [15, 40], [10, 20], [35, 10]], [[20, 30], [35, 35], [30, 20], [20, 30]]] }

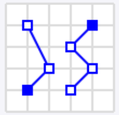
GeoJSON composites

MultiPoint



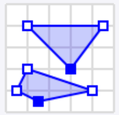
```
{ "type": "MultiPoint",
    "coordinates": [
        [10, 40], [40, 30], [20, 20], [30, 10]
    ]
}
```

MultiLineString



```
{ "type": "MultiLineString",
    "coordinates": [
        [[10, 10], [20, 20], [10, 40]],
        [[40, 40], [30, 30], [40, 20], [30, 10]]
    ]
}
```

MultiPolygon



```
{ "type": "MultiPolygon",
    "coordinates": [
        [
            [[30, 20], [45, 40], [10, 40], [30, 20]]
        ],
        [
            [[15, 5], [40, 10], [10, 20], [5, 10], [15, 5]]
        ]
    ]
}
```

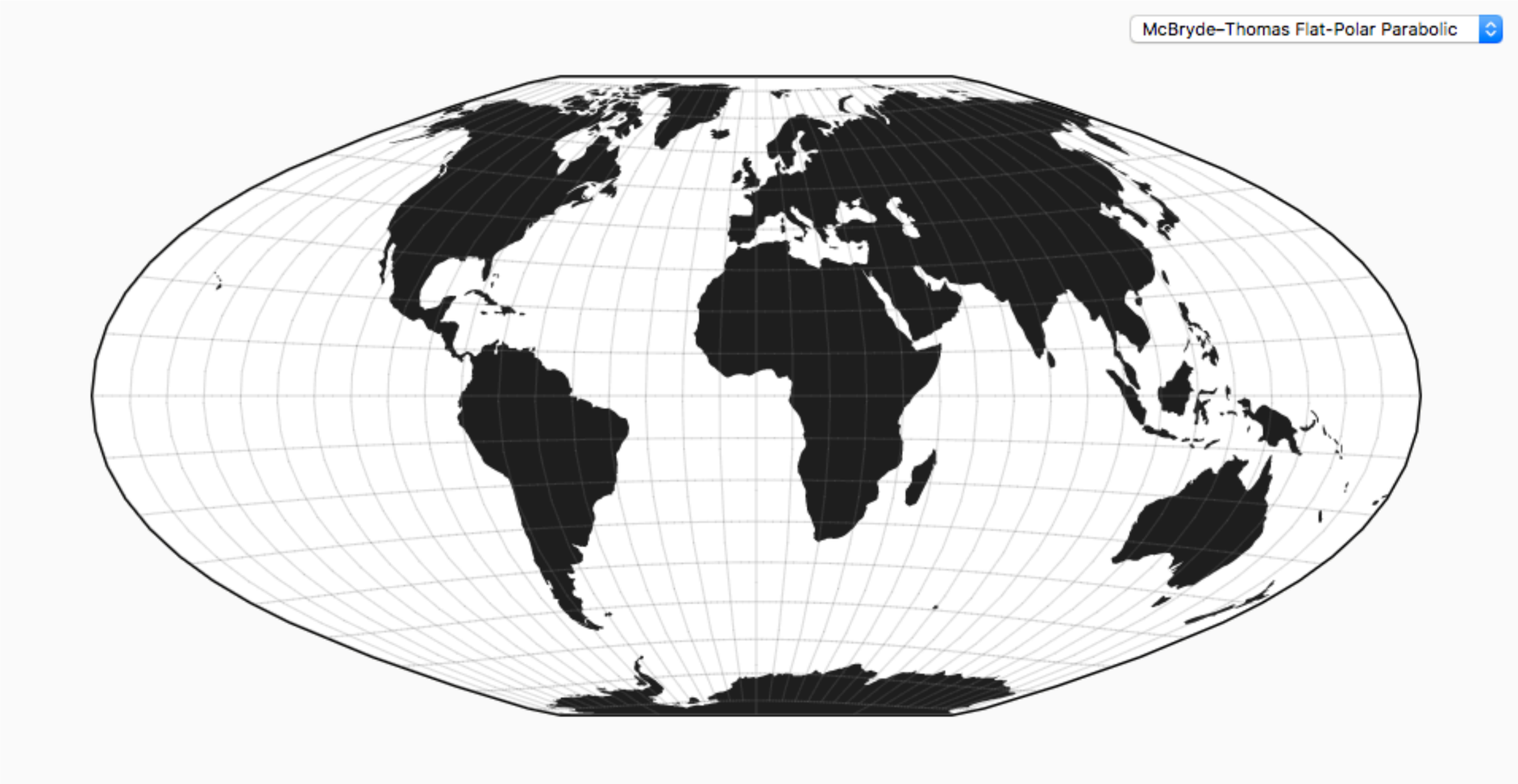
Wikipedia

GeoJSON features

```
{ "type": "FeatureCollection",
    "features": [
      { "type": "Feature",
        "geometry": {
          "type": "Point",
          "coordinates": [102.0, 0.5]
          },
          "properties": {
            "prop0": "value0"
          }
        },
      { "type": "Feature",
        "geometry": {
          "type": "LineString",
          "coordinates": [[102.0, 0.0], [103.0, 1.0], [104.0, 0.0], [105.0, 1.0]]
          },
        "properties": {
          "prop0": "value0",
          "prop1": 0.0
          }
```

Group both geometry and arbitrary properties such as names, roads, etc.

D3 projections



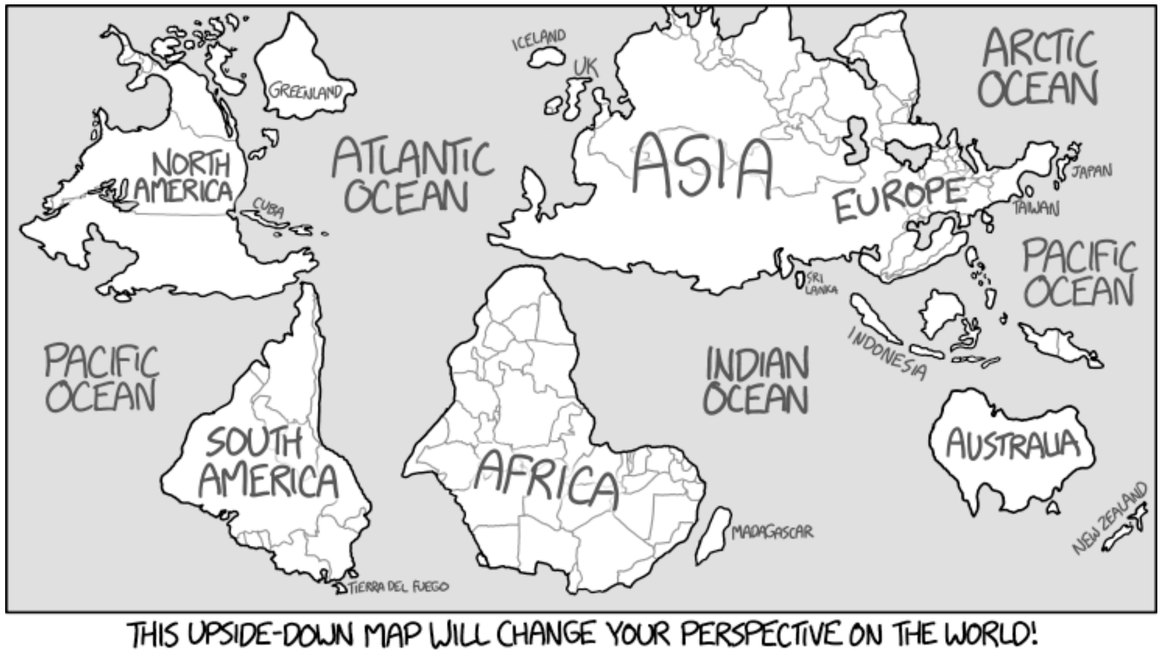
Bostock

How does it work?

The core object is ***d3.geoPath()***

Generates an SVG path data string, attribute “**d**” (or render to Canvas)

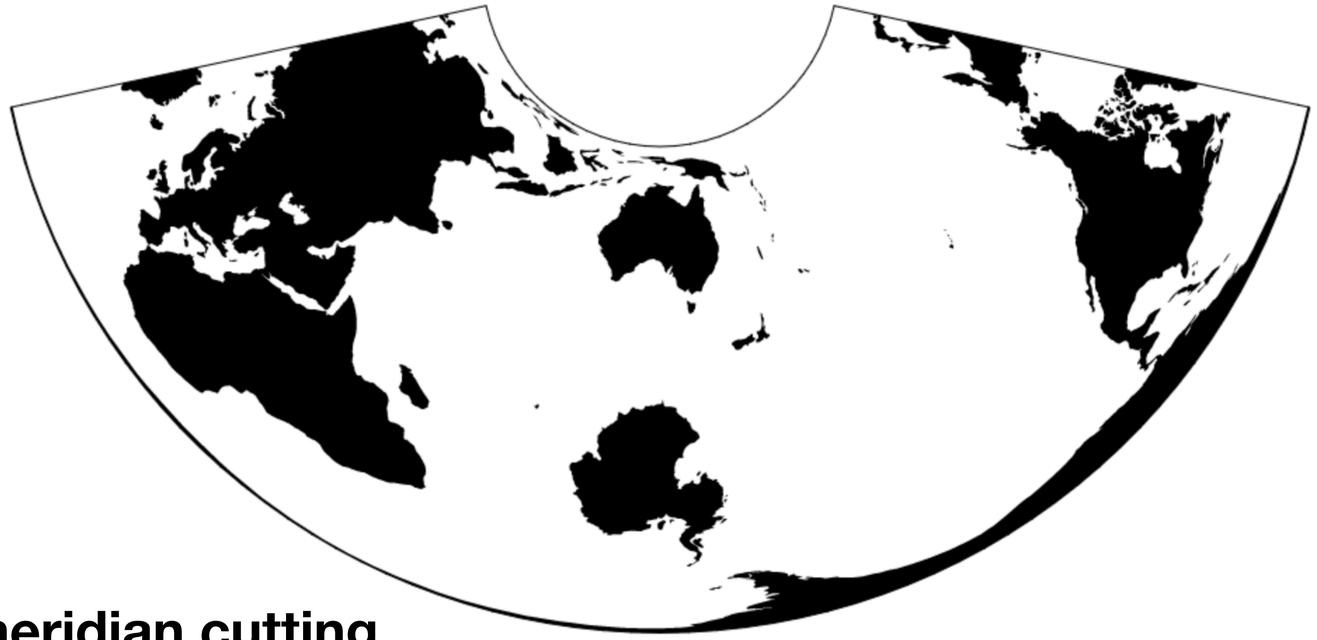
Relies on a **projection** to map the input to the correct position on screen



d3 projection

In a continous world, it is just a point transformation that takes latitude and longtiude values as input

In real-world discrete geometry, it is much more complex



Anti-meridian cutting

Example

```
let path = d3.geoPath()
             .projection(d3.geoAzimuthalEqualArea());

d3.json("/mbostock/raw/4090846/world-110m.json", function(error, world) {
 if (error) throw error;

 svg.insert("path", "world")
     .datum(topojson.feature(world, world.objects.land))
     .attr("class", "land")
     .attr("d", path);
});
```

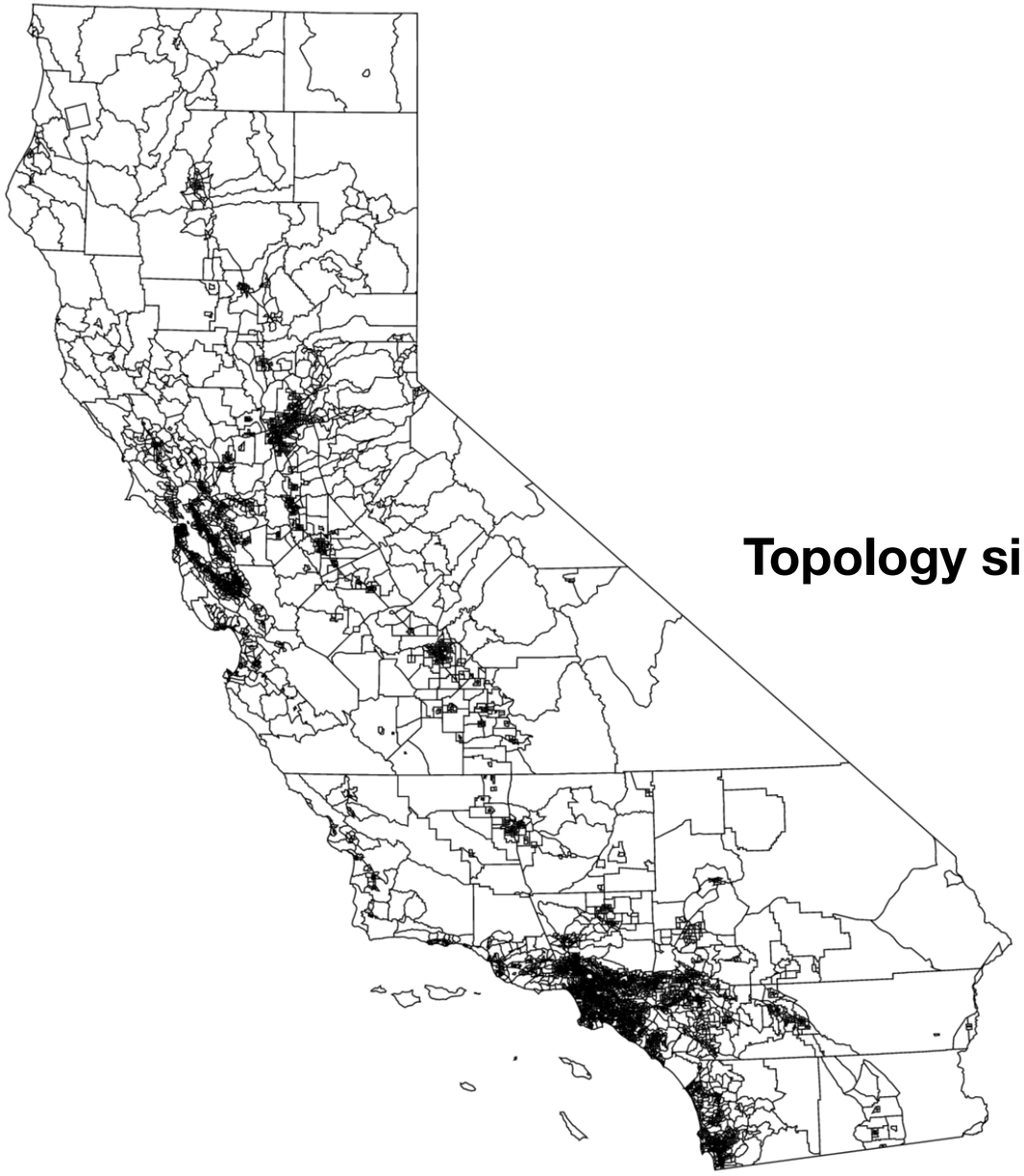
TopoJSON

Extension of GeoJSON that encodes topology

GeoJSON is not efficient because we refine the geometry of each object even if they share the same coordinates

Geometries in TopoJSON files are stitched together from shared line segments called **arcs**

Helps for topology simplification, cartograms!



Topology simplification

Bostock

TopoJSON example

Origine_Point
"coordinates": [0,0]

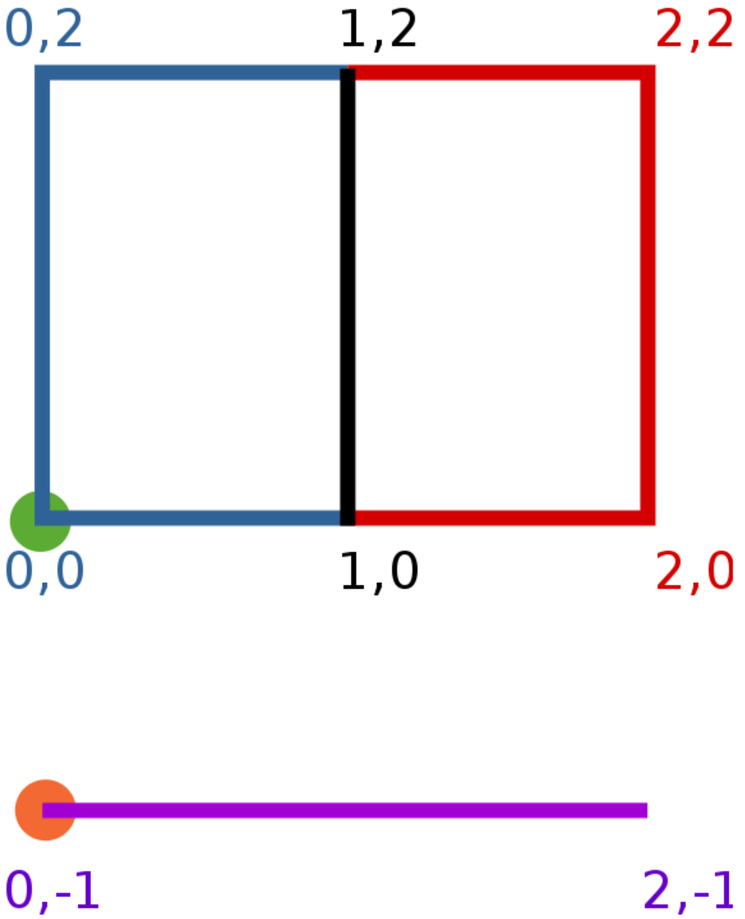
Under_Point
"coordinates": [0,-1]

Under_LineString
"arcs": [3]

Left_Polygon
"arcs": [[0,1]]

Right_Polygon
"arcs": [[2,-1]]

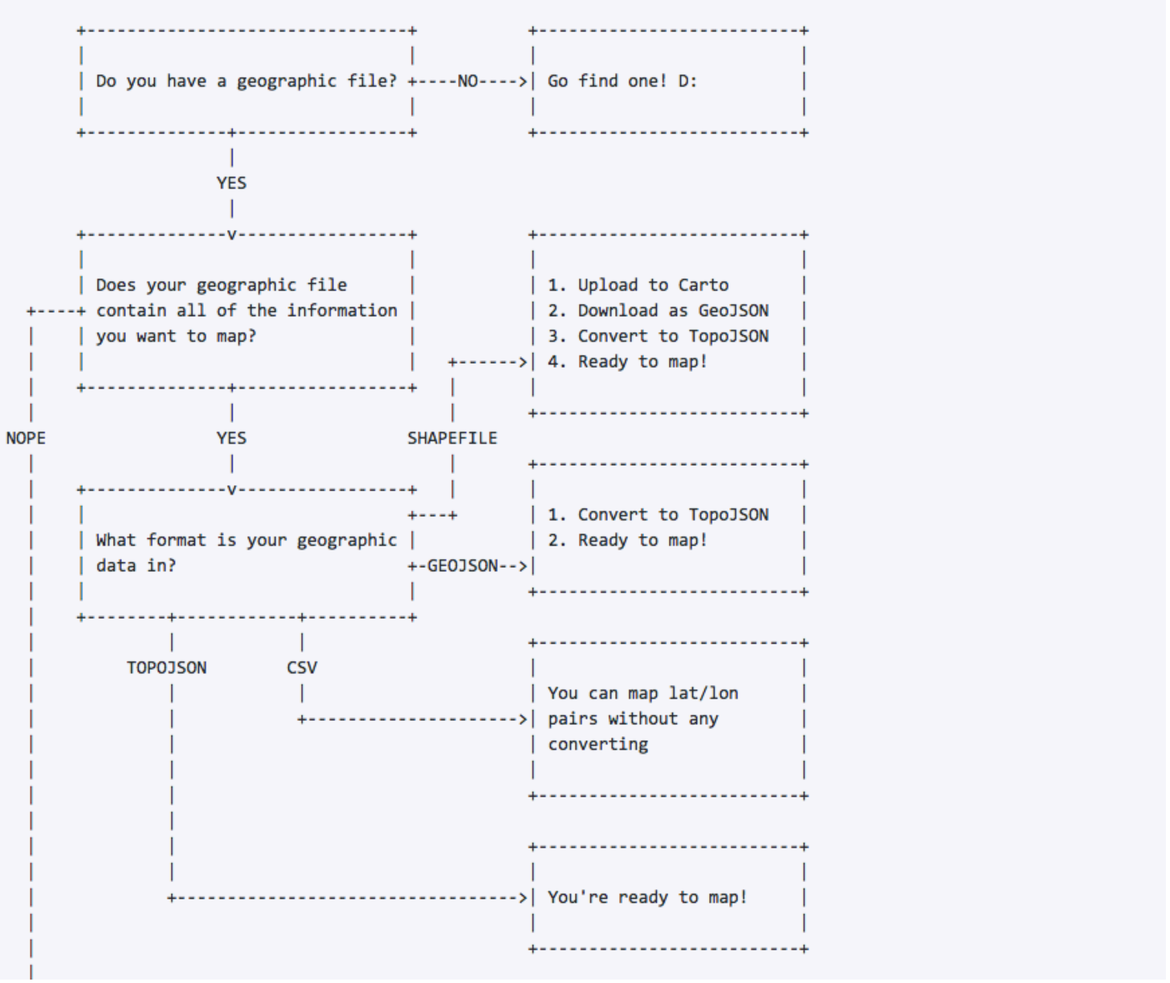
```
"arcs":[
    [[1,2],[0,-2]],
    [[1,0],[-1,0],[0,2][1,0]],
    [[1,2],[1,0],[0,-2],[-1,0]],
    [[0,-1],[2,0]]
]
```

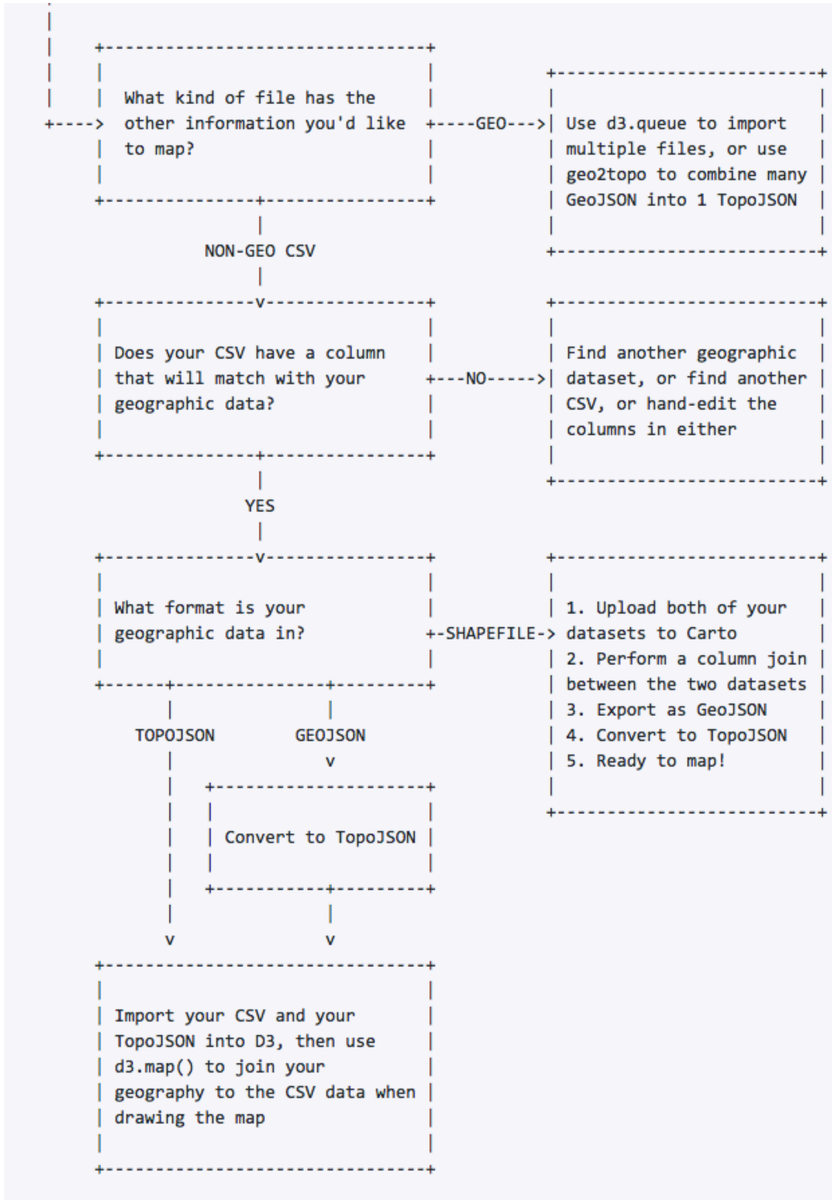


Wikipedia

A topology of the United States; dots indicate arc endpoints.

Bostock





TopoJSON guide

Search Data

Find Bus Routes, Park Services, Schools

Near City, Country

Total Datasets 100,489

Organizations Worldwide Sharing Open Data 5,524

Browse by Categories

Safe
- Crime
- Disaster
- Emergency Response

Prosperous
- Demographics
- Economy
- Education

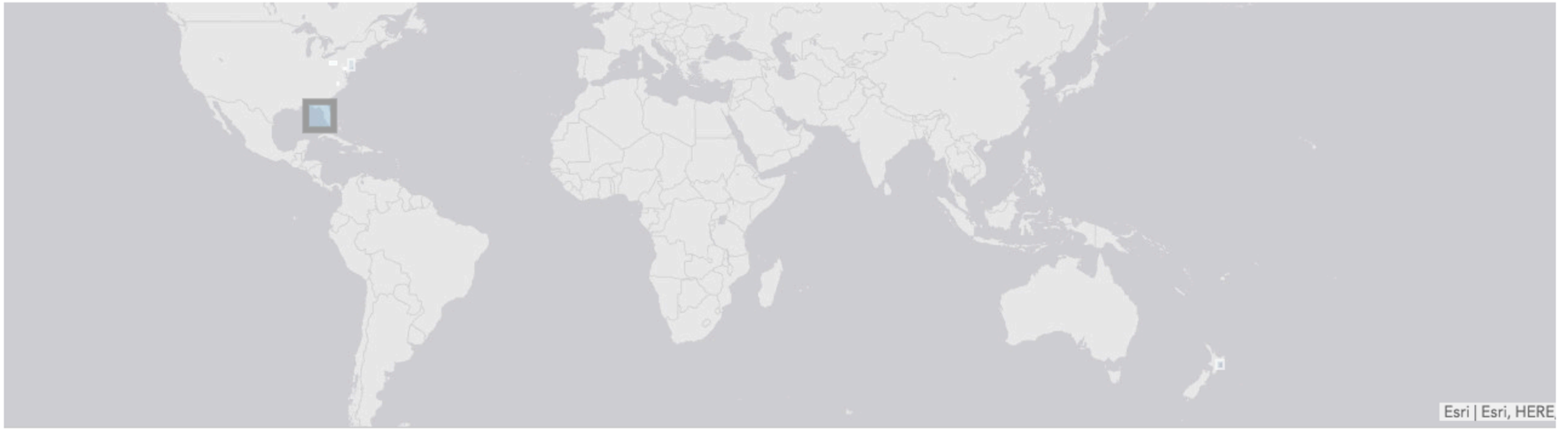
Sustainable
- Climate
- Energy
- Infrastructure

Healthy
- Agriculture
- Disease
- Health-Care

Liveable
- Culture
- Housing
- Transportation

Well-Run
- Boundaries
- Financial
- Planning & Land Use

ArcGIS Hub Community Initiatives Open Data Sustainable Near My Data Sign In



Views

My Items

Items

Open Data

Filter By

Tags

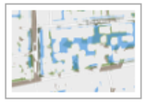
Sustainability **(332)**
.Sd **(216)**
Arcgis **(216)**
Service Definition **(216)**
Wildlife **(216)**

‹ 1 - 10 of 1,304 results ›

Relevance

SUSTAIN - Bioretention

Shared by CM_3RWW



This dataset illustrates 3 Rivers Wet Weather's (3RWW) 2013 analysis from the EPA System Urban Stormwater Treatment and Analysis Integration (SUSTAIN) model for the ALCOSAN service area. The data is intended to help municipalities comply with certain wet weather regulations, particularly the Pennsylvania DEP's geographic

Custom License 12/19/2016 Spatial Dataset 160,349 Rows

GeoJSON

Sustainable Forestry

Shared by aknight_cosspp

This dataset was developed to provide scientific decision support and help measure progress for preservation of resources associated with Florida Forever Measure G1: The number of acres acquired that are available for sustainable forest management; and Measure G2: The number of acres of state-owned forestland managed for

Custom License 1/11/2018 Raster Dataset

Leaflet: L.esri.MapService

Leaflet + d3

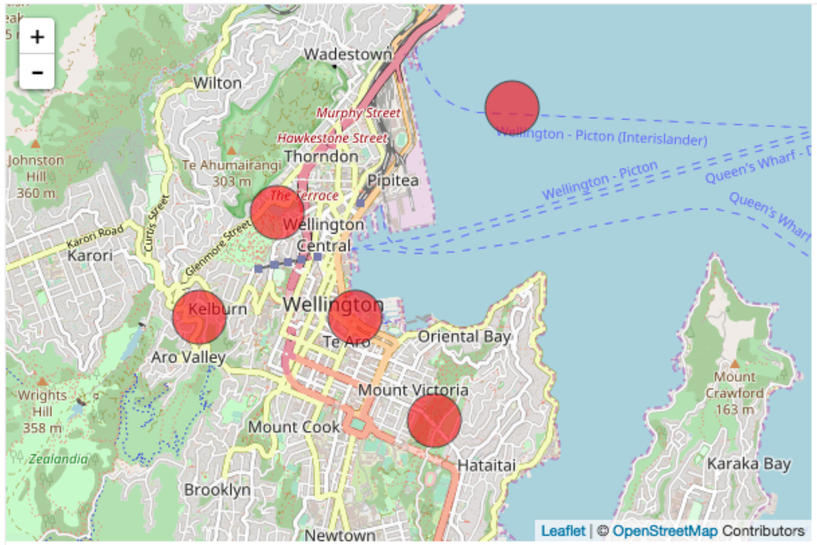
```
var states = [{
    "type": "Feature",
    "properties": {"party": "Republican"},
    "geometry": {
        "type": "Polygon",
        "coordinates": [[
            [-104.05, 48.99],
            [-97.22,  48.98],
            [-96.58,  45.94],
            [-104.03, 45.94],
            [-104.05, 48.99]
        ]]
    }
}, {
    "type": "Feature",
    "properties": {"party": "Democrat"},
    "geometry": {
        "type": "Polygon",
        "coordinates": [[
            [-109.05, 41.00],
            [-102.06, 40.99],
            [-102.03, 36.99],
            [-109.04, 36.99],
            [-109.05, 41.00]
        ]]
    }
}];

L.geoJSON(states, {
    style: function(feature) {
        switch (feature.properties.party) {
            case 'Republican': return {color: "#ff0000"};
            case 'Democrat':   return {color: "#0000ff"};
        }
    }
}).addTo(map);
```

d3noob's Block 9267535
Updated December 25, 2017

Popular / About

Map using leaflet.js and d3,js overlaid



Homework

- Read Interactive Data Visualization for the Web chapter 12
- Fill the project proposal form