SQUATCHIN WITH D3.JS

Michael Bowman
Applied Information Sciences
@webslingerm
http://github.com/bowmanmc/squatchin

SQUATCHIN
WITH D3.JS

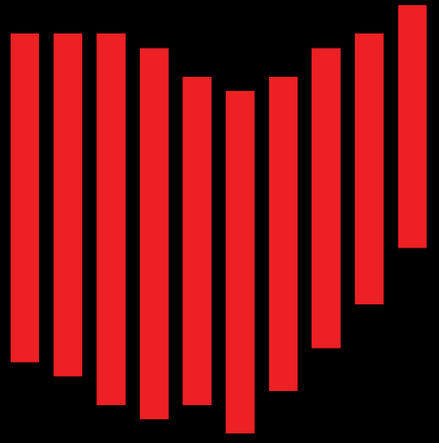
Brought to you in part by...

Applied Information Sciences

- WPAFB Contractor
- Full Stack Web Developers
- http://workatais.com

SQUATCHIN
WITH D3.JS

Brought to you in part by. . .



Dayton Data Visualization Meetup

- Quarterly Lunchtime Meetings
- http://meetup.com/daytondv

SQUATCHIN
WITH D3.JS

What You'll Learn Today

- How to find ~~silly~~ awesome data sets
- How to make a map with d3.js
- How to use PhantomJS to export SVG elements
- Why you would want to do such a thing

Let's Go

SQUATCHIN

SQUATCHIN
WITH D3.JS

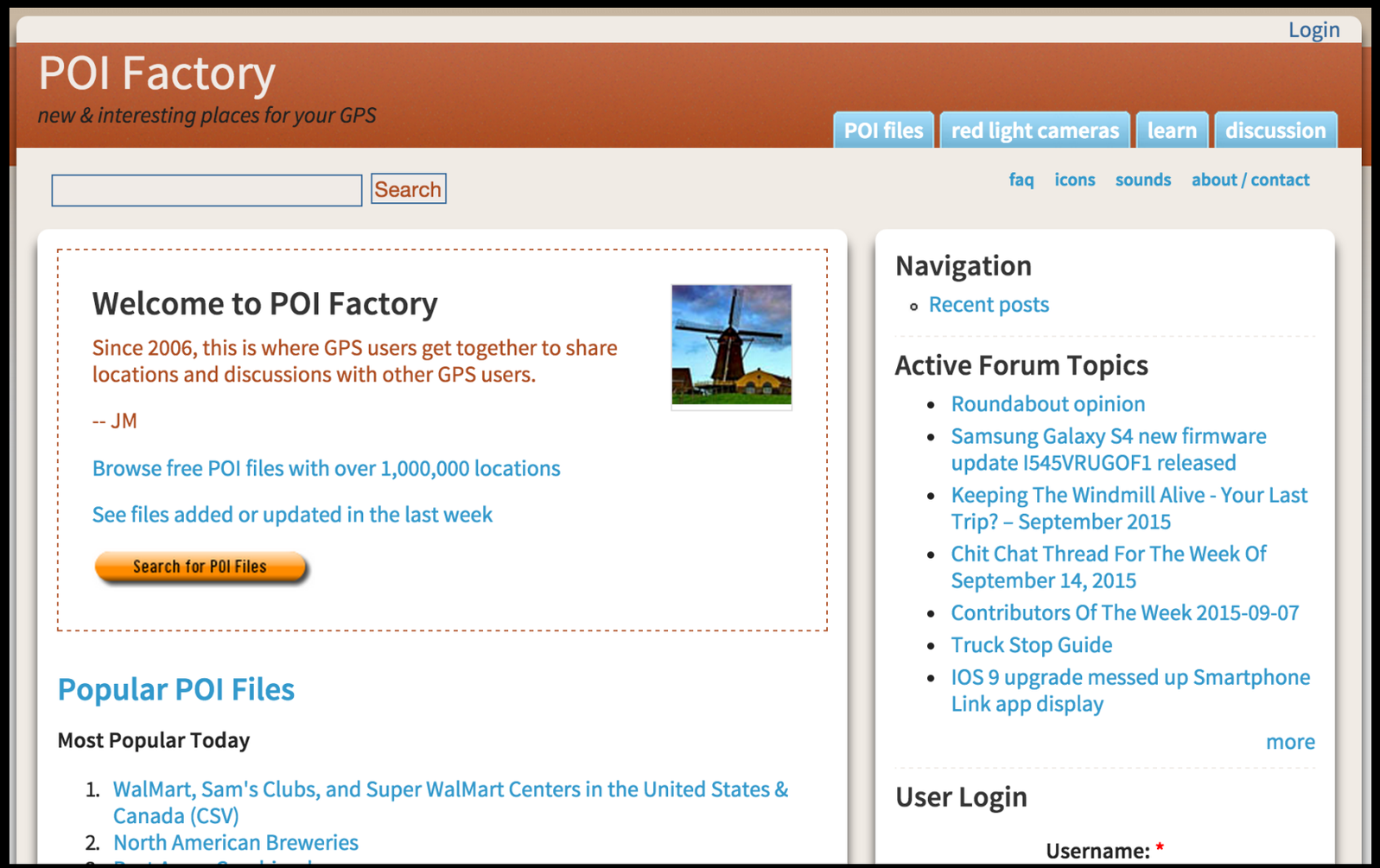
Sightings?

SQUATCHIN WITH D3.JS

SQUATCHIN WITH D3.JS

POI Factory

http://www.poi-factory.com



FINDING BIGFOOT

SQUATCHIN WITH D3.JS

BFRO.net

http://www.bfro.net/news/google_earth.asp

BFRO
The Bigfoot Field Researchers Organization

Google Earth Tool for Bigfoot Reports: The BFRO.net Layers

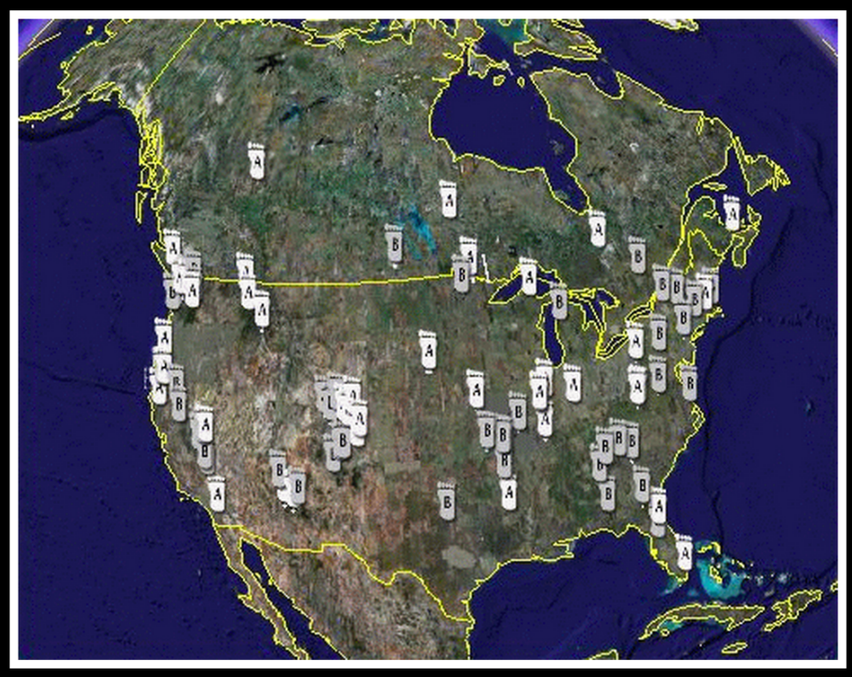
- Home
- Features
- Reports
- Media Articles
- Report Form
- FAQs
- Departments

Click on the links below to download BFRO.net Layers for Google Earth. You only need to install these layers one time in your Google Earth program, and they will automatically update themselves every time you launch Google Earth and periodically while you keep it open.

- **New Additions to BFRO.net**
  This layer shows the positions of reports added to the BFRO database over the past three months. Some of the reports are recent incidents, but many are not.
- **All Reports on BFRO.net**
  This layer shows *all* published reports in the BFRO database that have specified locations. (which is almost all of them) All of the reports are relevant for geographic analysis because they show patterns of possible habitiation areas.

Google Earth is a free, downloadable software program that shows images of the Earth from the sky. Google Earth can show the landscape in three dimensions from any angle. It also allows the user to zoom in and fly through the landscape like a flight simulator.

A Google Earth "layer" displays a set of geographic points overlayed on satellite



Click the links to the left to download the self-updating BFRO.net Layers.

SQUATCHIN
WITH D3.JS

Investigating the BFRO Data

- 3,556 Combined Sightings
- 12 Scatological Evidence
- 1 Stolen Fish

SQUATCHIN
WITH D3.JS

Investigating the BFRO Data

- 3 Classes of Bigfoot Sightings
- Class A - No Chance of Misidentification
- Class B - Long Distance, Sound Only
- Class C - All others

See http://www.bfro.net/gdb/classify.asp

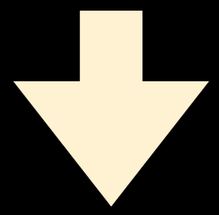
SQUATCHIN
WITH D3.JS

Cleaning Up the BFRO Data

- KML -> TSV
- FCC's Census Block API
  https://www.fcc.gov/developers/census-block-conversions-api
- description, timestamp, longitude, latitude, reportId, class, zip, city, state
- D3 has some handy utilities for delimited files

SQUATCHIN WITH D3.JS

```
52      this.getSightings = function() {
53          var deferred = $.Deferred();
54          d3.tsv('data/bfrodb.min.tsv', function(error, response) {
55              deferred.resolve(response);
56          });
57          return deferred.promise();
58      };
```




```
▼ 0: Object
    city: "Valdez-Cordova"
    class: "A"
    description: "Report 637: Campers' encounter just after dark in the Wrangell
    latitude: "61.5"
    longitude: "-142.9"
    reportId: "637"
    state: "AK"
    timestamp: "2000-06-16T12:00:00Z"
    zip: "02261"
```

SQUATCHIN
WITH D3.JS

```
43
44          map.getOutlines().then(function(outlines) {
45              map.drawOutlines(outlines);
46              map.getSightings().then(function(sightings) {
47                  map.drawSightings(sightings);
48              });
49          });
```

SQUATCHIN WITH D3.JS

SVG Primer

- Scalable Vector Graphics - W3C - 1999
- Widely Adopted

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
8								[1] 4.3	
[2] 9			45					4.4	
[2] 10	[2] 12	42	46			8.4		4.4.4	
[2] 11	[2] 13	43	47	9	34	9.2	8	46	47
	[2] 14	44	48		35				
		45	49		36				
		46	50						

SQUATCHIN
WITH D3.JS

SVG Primer

```svg
<?xml version="1.0" standalone="no"?>

<svg width="100px" height="100px" version="1.1" xmlns="http://www.w3.org/2000/svg">

  <path d="M10 10 H 90 V 90 H 10 L 10 10"/>

  <!-- Points -->
  <circle cx="10" cy="10" r="2" fill="red"/>
  <circle cx="90" cy="90" r="2" fill="red"/>
  <circle cx="90" cy="10" r="2" fill="red"/>
  <circle cx="10" cy="90" r="2" fill="red"/>

</svg>
```

https://developer.mozilla.org/en-US/docs/Web/SVG/Tutorial/Paths

SQUATCHIN
WITH D3.JS

D3.js Primer

- Toolkit for building interactive visualizations with SVG

http://theikeablog.com/2008/08/29/ikea-brooklyn/

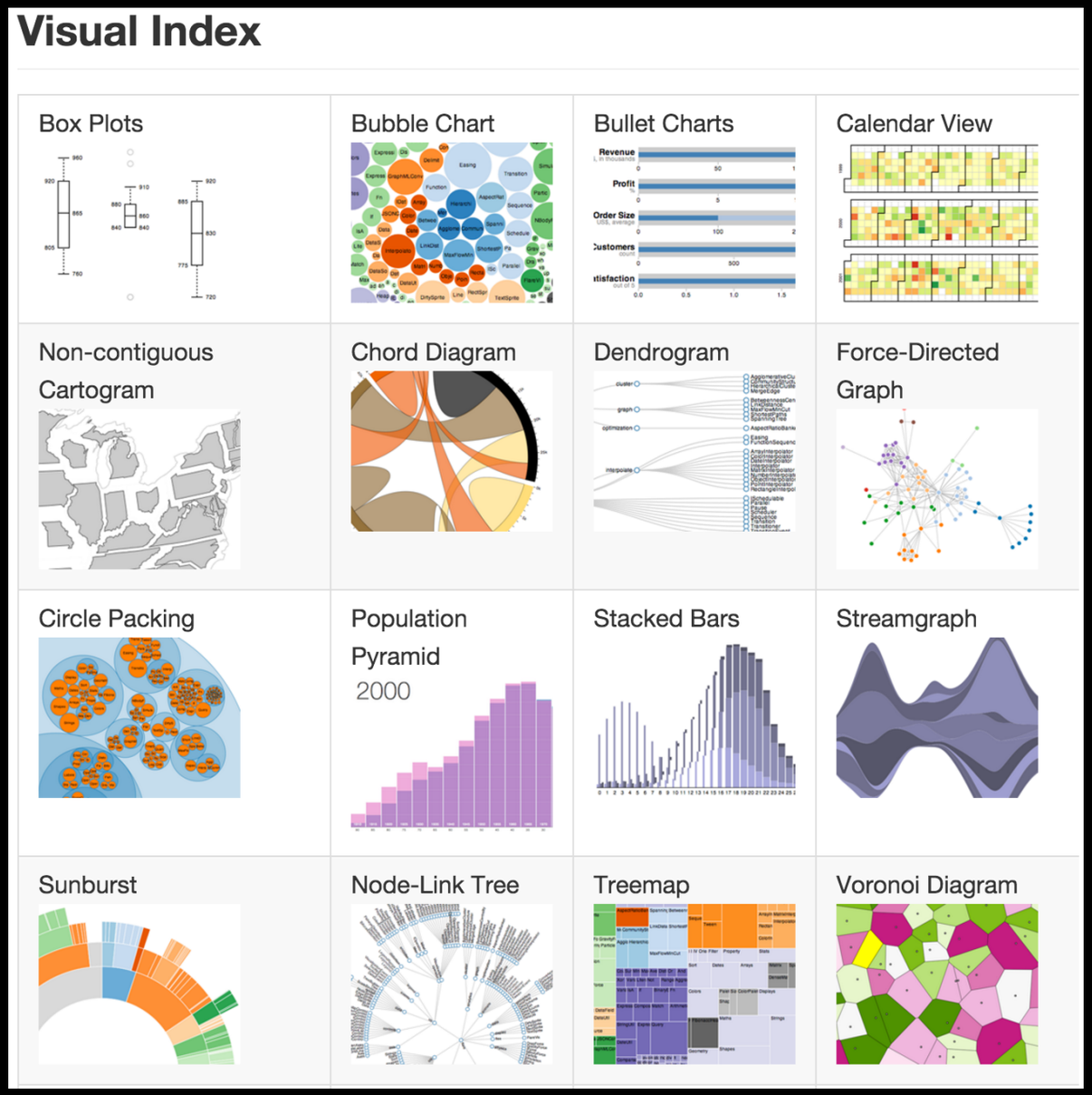
VS

http://www.popularwoodworking.com/woodworking-daily/woodworking-hand-tools-sloyd-list

SQUATCHIN WITH D3.JS

D3.js Primer

- http://d3js.org
- https://github.com/mbostock/d3/wiki/Gallery



SQUATCHIN
WITH D3.JS

OK, but *why*?

Rasters

- Google Maps, MapBox
- Tiles are images

Vectors

- Stylable
- Interactive

SQUATCHIN
WITH D3.JS

Let's Do This!

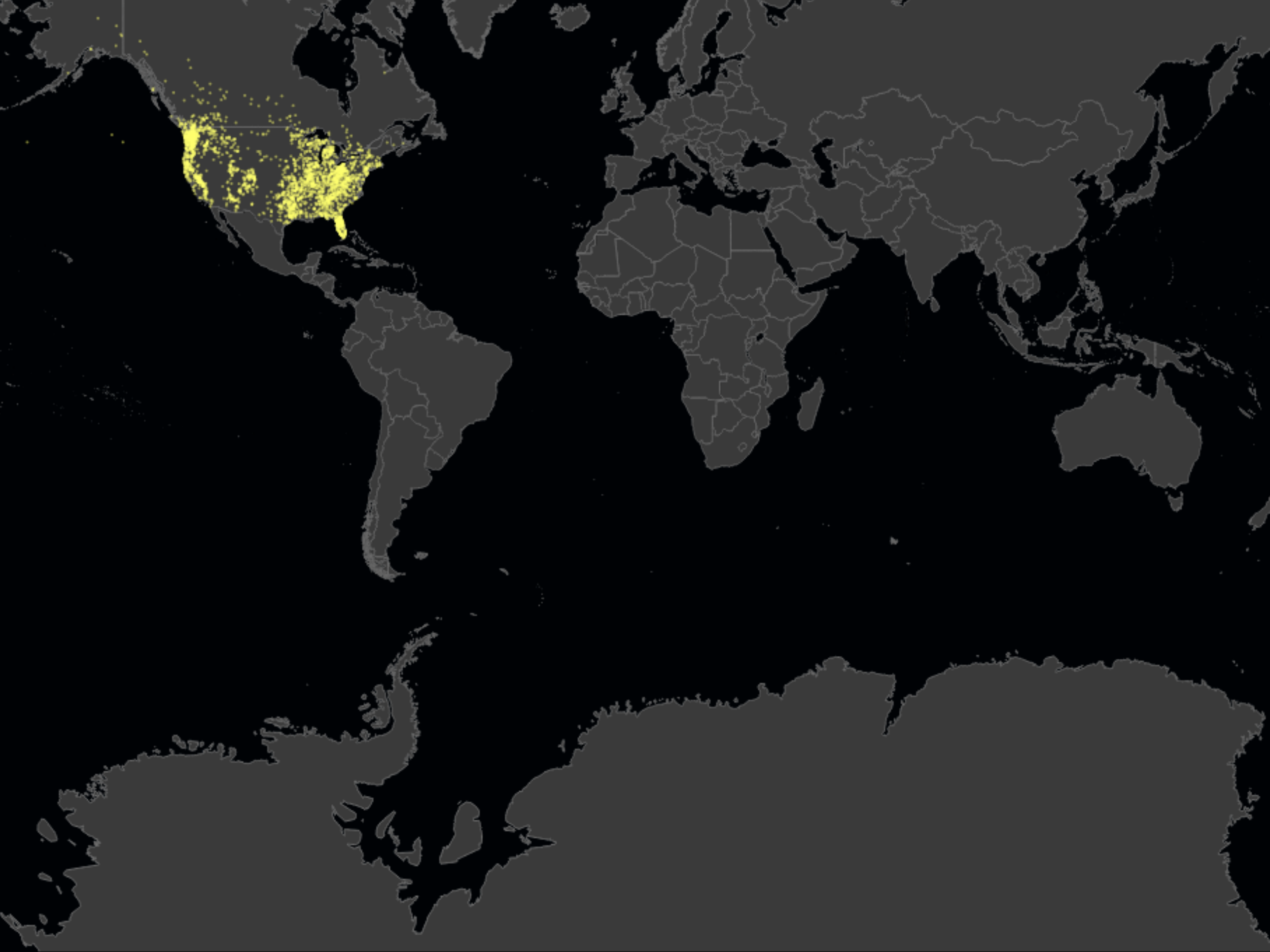
SQUATCHIN WITH D3.JS

```
17          map.projection = d3.geo.mercator();
18          map.path = d3.geo.path().projection(map.projection);
```

```
125     this.drawOutlines = function(data) {
126         var map = this;
127
128         map.land.selectAll('path')
129             .data(data.features)
130             .enter().append('path')
131                 .attr({
132                     'class': 'land',
133                     'd': map.path
134                 });
135     };
```

SQUATCHIN WITH D3.JS

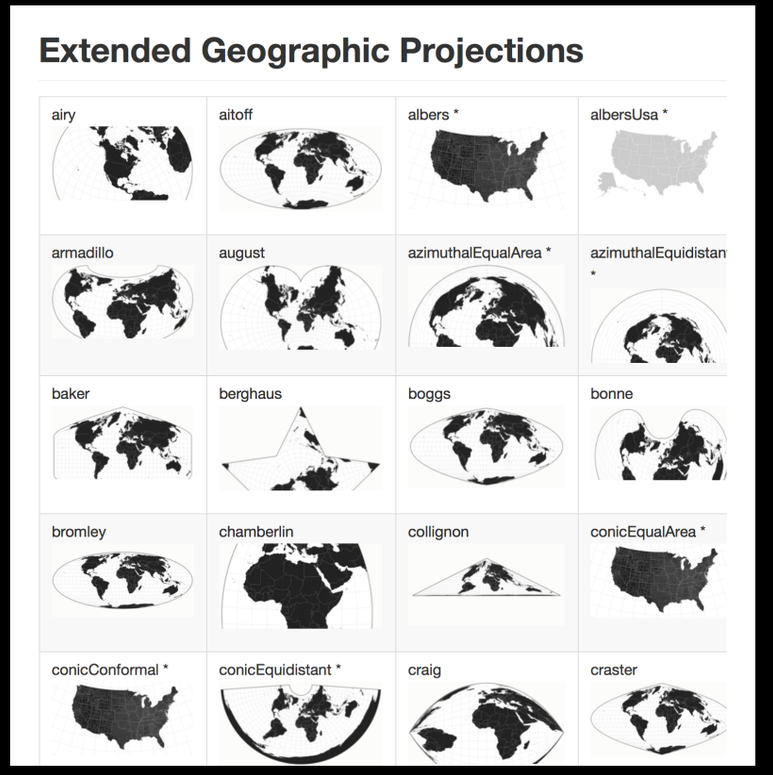
```
102        map.sightings.selectAll('circle')
103            .data(data)
104            .enter().append('circle')
105                .attr('r', 1)
106                .attr('class', 'pin')
107                .attr('transform', function(d) {
108                    return "translate(" + map.projection([
109                        d.longitude,
110                        d.latitude
111                    ]) + ")";
112                });
```



SQUATCHIN WITH D3.JS

Designing the Map

- Limit Area to USA/CAN
- Projection - Airy's Minimum Error https://github.com/d3/d3-geo-projection
- Grid
- Group Sightings by Class



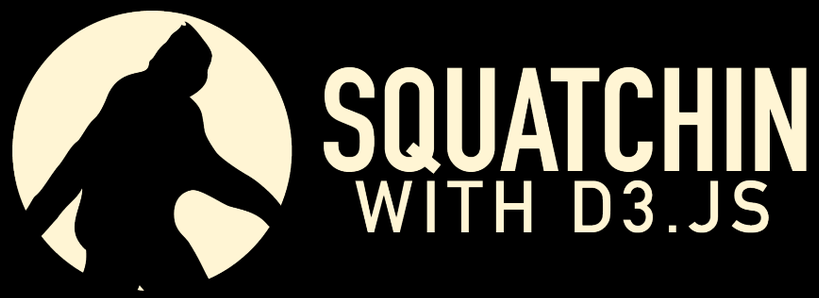
See "Designing Beautiful Maps with D3.js - Michael Bowman"

SQUATCHIN WITH D3.JS

```
20        map.svg = d3.select(map.divSelector).append('svg')
21            .attr({
22                'id': 'map',
23                'width': map.width,
24                'height': map.height
25            });
26
27        map.grid = map.svg.append('g').attr({
28            'id': 'grid'
29        });
30        map.land = map.svg.append('g').attr({
31            'id': 'land'
32        });
33        map.classc = map.svg.append('g').attr({
34            'id': 'class-c'
35        });
36        map.classb = map.svg.append('g').attr({
37            'id': 'class-b'
38        });
39        map.classa = map.svg.append('g').attr({
40            'id': 'class-a'
41        });
42
```

```
<svg id="map" width="1558" height="1074">
  <g id="grid">…</g>
  <g id="land">…</g>
  <g id="class-c">…</g>
  <g id="class-b">…</g>
  <g id="class-a">…</g>
</svg>
```

See “Designing Beautiful Maps with D3.js - Michael Bowman”




```
124     this.drawOutlines = function(data) {
125         var map = this;
126
127         var center = {
128             'lon': -90,
129             'lat': 80
130         };
131         var r = [center.lon * -1, center.lat * -1];
132         console.log('Rotate to: ' + JSON.stringify(r));
133         map.projection.scale(1).translate([0, 0]).rotate(r);
134
135         var b = map.path.bounds(data),
136             s = 0.98 / Math.max((b[1][0] - b[0][0]) / map.width, (b[1][1] - b[0][1]) / map.height),
137             t = [(map.width - s * (b[1][0] + b[0][0])) / 2, (map.height - s * (b[1][1] + b[0][1])) / 2];
138         map.projection.scale(s).translate(t);
139
140         var graticule = d3.geo.graticule()
141             .extent([
142                 // left, bottom (lon, lat)
143                 [-180, -30],
144                 // right, top   (lon, lat)
145                 [180, 90]
146             ])
147             .step([5, 5]); // how many degrees between graticule lines
148
```

SQUATCHIN WITH D3.JS

```
149        map.grid.append('path')
150            .datum(graticule)
151            .attr('class', 'graticule')
152            .attr('d', map.path);
153
154        map.land.selectAll('path')
155            .data(data.features)
156            .enter().append('path')
157                .attr({
158                    'class': 'land',
159                    'd': map.path
160                });
161    };
```

SQUATCHIN
WITH D3.JS

Pretty Good… but…

- That thing looks like it was designed by a defense contractor…
- What about interactivity?

SQUATCHIN
WITH D3.JS

Squatchin on CodePen

- Fork/Remix this map on CodePen and check out the other web versions
- http://codepen.io/collection/DadbNM
- http://codepen.io/webslingerm/collections

SQUATCHIN
WITH D3.JS

Breaking Out of the Browser

- Why?
  - Use other design tools (Adobe Illustrator, Sketch)
  - Caching, Post Processing (SVGO), Fun
- How?
  - PhantomJS

SQUATCHIN WITH D3.JS

PhantomJS

SOURCE CODE DOCUMENTATION API EXAMPLES FAQ

Full web stack
No browser required

PhantomJS is a headless WebKit scriptable with a JavaScript API. It has **fast** and **native** support for various web standards: DOM handling, CSS selector, JSON, Canvas, and SVG.

Download v2.0 Get started

```
// Simple Javascript example

console.log('Loading a web page');
var page = require('webpage').create();
var url = 'http://phantomjs.org/';
page.open(url, function (status) {
  //Page is loaded!
  phantom.exit();
});
```

Community: Read the release notes Join the mailing list Report bugs

PhantomJS is an optimal solution for

HEADLESS WEBSITE TESTING
Run functional tests with frameworks such as Jasmine, QUnit, Mocha, Capybara,

SCREEN CAPTURE
Programmatically capture web contents, including SVG and Canvas. Create web site

PAGE AUTOMATION
Access and manipulate webpages with the standard DOM API, or with usual libraries like jQuery.

NETWORK MONITORING
Monitor page loading and export as standard HAR files. Automate performance analysis using YSlow

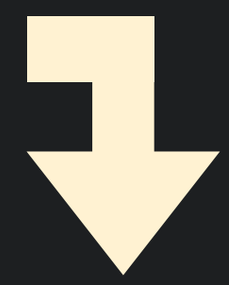
SQUATCHIN WITH D3.JS

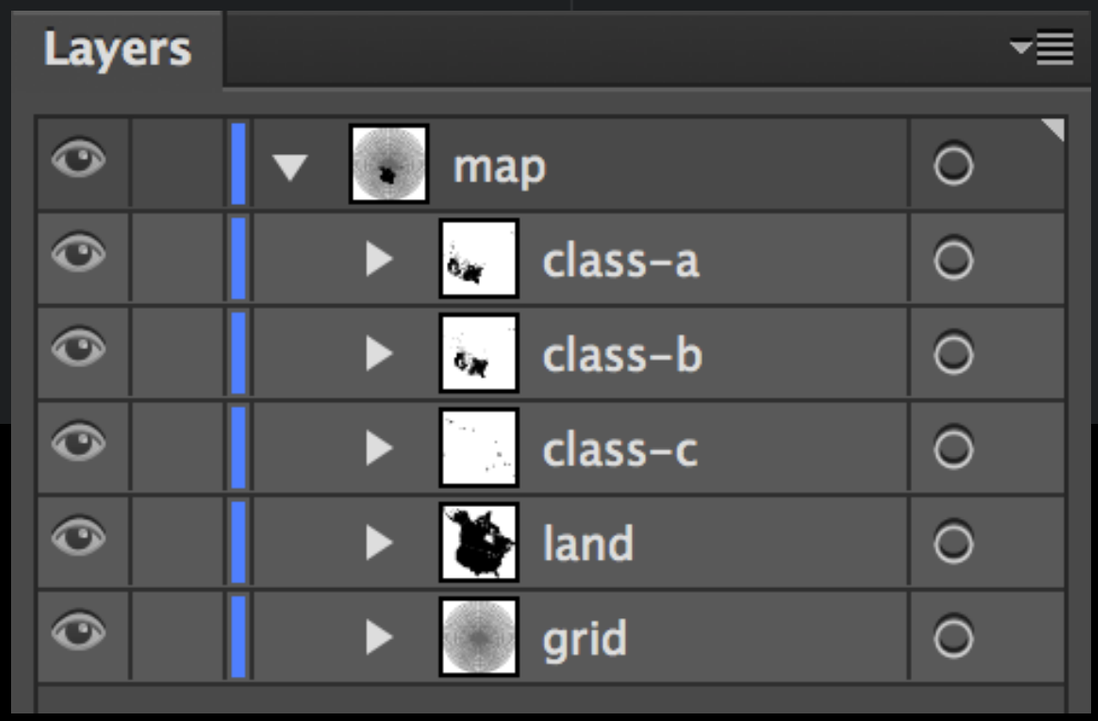
```
/**
 * savemap.js
 * Finds the map on the page and writes it out to an svg file
 * To Run:
 *    npm install -g phantomjs
 *    phantomjs savemap.js
 */
var fs = require('fs');
var page = require('webpage').create();
var url = 'file://' + fs.absolute('./app/index.html');

page.open(url, function(status) {

    var svg = page.evaluate(function() {
        var serializer = new XMLSerializer();
        var el = document.getElementById('map');
        return serializer.serializeToString(el);
    });
    var filename = 'map.svg';

    fs.write(fs.absolute('./svg/' + filename), svg, 'w');
    phantom.exit();
});
```

SQUATCHIN
WITH D3.JS

Importing into Illustrator

```
27        map.grid = map.svg.append('g').attr({
28            'id': 'grid'
29        });
30        map.land = map.svg.append('g').attr({
31            'id': 'land'
32        });
33        map.classc = map.svg.append('g').attr({
34            'id': 'class-c'
35        });
36        map.classb = map.svg.append('g').attr({
37            'id': 'class-b'
38        });
39        map.classa = map.svg.append('g').attr({
40            'id': 'class-a'
41        });
42
```

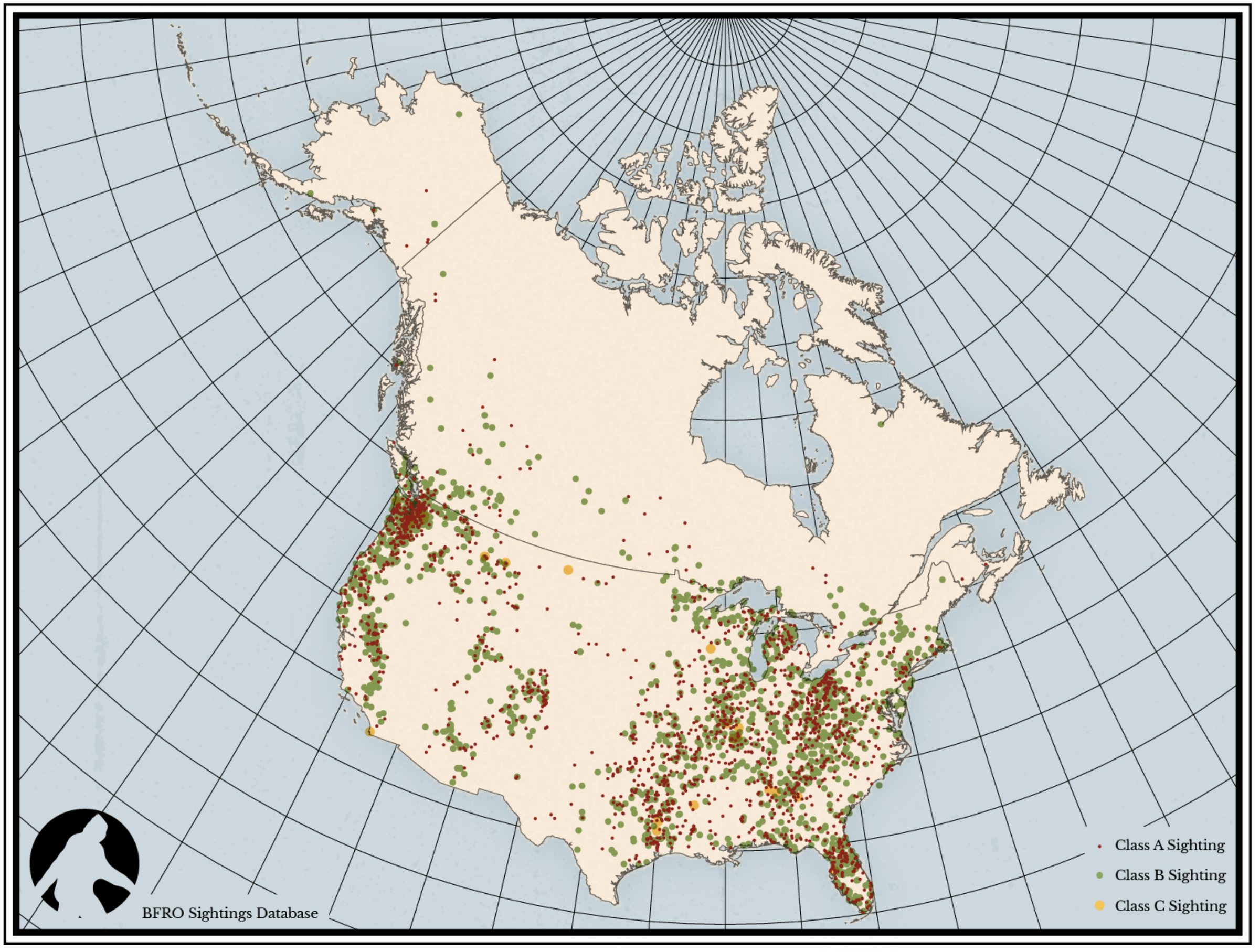




SQUATCHIN
WITH D3.JS

Tweaking Layout and Styles

- Tweak Colors
- Frame Map
- Add Title Logo
- Add Legend


BFRO Sightings Database
Class A Sighting
Class B Sighting
Class C Sighting

SQUATCHIN WITH D3.JS

Additional Resources

- Filament Group SVG to PNG
  https://www.npmjs.com/package/svg-to-png
- Export with CSS Styles
  http://spin.atomicobject.com/2014/01/21/convert-svg-to-png
- D3.geo Blocks
  http://bl.ocksplorer.org/#/search/d3.geo
- Designing Beautiful Maps with D3.js
  https://github.com/bowmanmc/designing_maps
- BFRO
  http://www.bfro.net
- TriState Bigfoot
  http://www.tristatebigfoot.com
- Sasquatch Genome Project
  http://www.sasquatchgenomeproject.org/
- Josh Stevens Map
  http://www.joshuastevens.net/visualization/squatch-watch-92-years-of-bigfoot-sightings-in-us-and-canada

SQUATCHIN WITH D3.JS

Additional Resources

- Interactive Data Visualization for the Web
  http://chimera.labs.oreilly.com/books/1230000000345

- D3.js in Action
  https://www.manning.com/books/d3-js-in-action

- Designing Better Maps
  http://www.amazon.com/Designing-Better-Maps-Guide-Users/dp/1589480899

- Data.Gov
  http://www.data.gov/

- QGIS
  http://www.qgis.org/en/site

- D3.js
  https://github.com/d3
  https://github.com/mbostock/d3

SQUATCHIN WITH D3.JS

Thank You!

Michael Bowman
Applied Information Sciences
@webslingerm
http://github.com/bowmanmc/squatchin