DATA-DRIVEN DOCUMENTS
(d3.js, D3, D3)

It's just a toolkit...
It’s just a toolkit…

a really powerful toolkit, but you still have to do all of the programming and design.

WHAT DOES D3 DO?

• A Javascript framework for generating, styling and animating web content based on data.

• What does D3 do?
  • Load data into the browser’s memory
  • Create, Update, and Remove web page elements based on data (referred to as “binding data to elements on the web page”)
  • Transform and Style those page elements based on data
  • Transition elements between states in response to user input
WHAT DOES D3 DO?

- D3 provides a large offering of data-to-visual mappings for you to use in transforming and styling your data visualization.

<table>
<thead>
<tr>
<th>Axes</th>
<th>Geographies</th>
<th>Scales</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brushes</td>
<td>Hierarchies</td>
<td>Shapes</td>
</tr>
<tr>
<td>Chords</td>
<td>Interpolators</td>
<td>Time Formats</td>
</tr>
<tr>
<td>Colors</td>
<td>Number Formats</td>
<td>Time Intervals</td>
</tr>
<tr>
<td>Dragging</td>
<td>Paths</td>
<td>Transitions</td>
</tr>
<tr>
<td>Easings</td>
<td>Polygons</td>
<td>Voronoi Diagrams</td>
</tr>
<tr>
<td>Forces</td>
<td>Quadtrees</td>
<td>Zooming</td>
</tr>
</tbody>
</table>

- While D3 provides a framework to apply these mappings, it is up to you to design how to use these rules in your visualization.

D3.JS - ORIGINS

- D3 was originally written by Mike Bostock, Vadim Ogievetsky, and Jeff Heer
- Bostock started working on D3 as a PhD Student at Stanford
- D3 Predecessors (Jeff Heer’s work):
  - ProtoVis & Prefuse - Java toolkit
  - Flare - ActionScript that runs on Adobe Flash Player
- D3’s success stems from:
  - piggy-backing on web technologies
  - thorough documentation
  - growing community of programmers
  - Visit https://bl.ocks.org/ - the D3 community showcase
The Gold Standard in Web Visualization

- D3.js appears everywhere on the web:
  - From data journalism pieces at the *New York Times*, *Wall Street Journal*, and *Washington Post* to...
  - The blogs and data visualization groups scattered about the web (e.g. [http://pudding.cool/](http://pudding.cool/) and [http://bocoup.com/](http://bocoup.com/) are worth a look)

- D3 certainly owes some of its popularity to riding the *everything-on-the-web* trend
- Nevertheless, the declarative way in which you can express relationships between data and visual elements is *fundamentally superior to any other library!*

D3.js - AN EXAMPLE
D3.js - An Example

```javascript
<script>
  var numericData = [1, 2, 4, 8, 16];
  var svg = d3.select('svg');
  // Add rectangles
  svg.selectAll('rect')
    .data(numericData)
    .enter()
    .append('rect');
</script>
```
D3.js - An Example

```javascript
var numericData = [1, 2, 4, 8, 16];
var svg = d3.select("svg");

// Add rectangles
svg.selectAll("rect")
    .data(numericData)
    .enter()
    .append("rect")
    .attr("fill", "#ff7f50")
    .attr("width", 300)
    .attr("height", 400);
```

D3.js - An Example

```javascript
var numericData = [1, 2, 4, 8, 16];
var svg = d3.select("svg");

// Add rectangles
svg.selectAll("rect")
    .data(numericData)
    .enter()
    .append("rect")
    .attr("fill", "#ff7f50")
    .attr("width", 300)
    .attr("height", 400);
```
D3.js - AN EXAMPLE

```javascript
var numericData = [1, 2, 4, 8, 16];
var svg = d3.select('svg');

// Add rectangles
svg.selectAll('rect')
  .data(numericData)
  .enter()
    .append('rect')
      .attr('fill', '#FF7F00')
      .attr('width', 50)
      .attr('height', function(d) {
        return (1 + d) * 50;
      })
      .attr('y', 20)
      .attr('x', function(d, i) {
        return (i + 40) * 50;
      });
```

D3.js - AN EXAMPLE

```javascript
var numericData = [1, 2, 4, 8, 16];
var svg = d3.select('svg');

// Add rectangles
svg.selectAll('rect')
  .data(numericData)
  .enter()
    .append('rect')
      .attr('fill', '#FF7F00')
      .attr('width', 50)
      .attr('height', function(d) {
        return (1 + d) * 50;
      })
      .attr('y', 20)
      .attr('x', function(d, i) {
        return (i + 40) * 50;
      });
```
LAB PROCEDURE

Before Class

- Read Chapters 5 and 6 - Interactive Data Visualization for the Web by Scott Murray
- Git pull example code (https://github.gatech.edu/CS-4460/Labs.git)

In-Class

- Open Lab 3 instruction page (https://github.gatech.edu/CS-4460/Labs/wiki)
- Work through activities
  - First thing, start a simple http server with python at 03_lab directory

After Class

- If you run out of time, finish all lab activities
- Lab Solutions will be posted Monday nights