# PBO Workshop

## Data-Driven Documents With d3

Ben Racine [1]

[1] Cornerstone Systems NW

November 2, 2011

# Introductions

**Favorite Tools?**

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

**Any web developers?**

- Javascript experience?

# Introductions

## Favorite Tools?

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

## Any web developers?

- Javascript experience?

# Introductions

## Favorite Tools?

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

## Any web developers?

- Javascript experience?

# Introductions

## Favorite Tools?

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

## Any web developers?

- Javascript experience?

# Introductions

## Favorite Tools?

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

## Any web developers?

- Javascript experience?

# Introductions

## Favorite Tools?

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

## Any web developers?

- Javascript experience?

# Introductions

## Favorite Tools?

- Database and/or spreadsheet tools
- DSL and/or general programming languages
- Visualization tools

## Any web developers?

- Javascript experience?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Introductions

## Browser Poll

- Chrome
- Firefox 3+
- IE9
- Safari
- None of the above?

# Background

## jQuery + Protovis ≈ d3

- Any jQuery experience?
  - d3 is similar, but can also target the SVG (xml-like image format)
  - They both do some fancy functional programming to make it possible for us to declaratively (and efficiently) reach into the dom tree
- Any Protovis exposure by any chance?

# Background

## jQuery + Protovis ≈ d3

- Any jQuery experience?
  - d3 is similar, but can also target the SVG (xml-like image format)
  - They both do some fancy functional programming to make it possible for us to declaratively (and efficiently) reach into the dom tree
- Any Protovis exposure by any chance?

# Background

## jQuery + Protovis ≈ d3

- Any jQuery experience?
    - d3 is similar, but can also target the SVG (xml-like image format)
    - They both do some fancy functional programming to make it possible for us to declaratively (and efficiently) reach into the dom tree
- Any Protovis exposure by any chance?

# Background

## jQuery + Protovis ≈ d3

- Any jQuery experience?
  - d3 is similar, but can also target the SVG (xml-like image format)
  - They both do some fancy functional programming to make it possible for us to declaratively (and efficiently) reach into the dom tree
- Any Protovis exposure by any chance?

# Background

## jQuery + Protovis ≈ d3

- Any jQuery experience?
  - d3 is similar, but can also target the SVG (xml-like image format)
  - They both do some fancy functional programming to make it possible for us to declaratively (and efficiently) reach into the dom tree
- Any Protovis exposure by any chance?

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  :
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  :
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  ⋮
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  ⋮
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  ⋮
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  :
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
  - API: https://github.com/mbostock/d3/wiki/API-Reference
  - Examples: http://mbostock.github.com/d3/ex/
  - Source: https://github.com/mbostock/d3.git

  :
- Google message group
- SVG Specification (v1.1)
- Twitter: @*i3enhamin*, @*mbostock*

# Resources

- Github: http://mbostock.github.com/d3/
    - API: https://github.com/mbostock/d3/wiki/API-Reference
    - Examples: http://mbostock.github.com/d3/ex/
    - Source: https://github.com/mbostock/d3.git

  :
- Google message group
- SVG Specification (v1.1)
- Twitter: *@i3enhamin*, *@mbostock*

# Clone or Download Slides, Source Code and Exercises

```
if you are a git user
    git clone git@github.com:benracine/d3_cisnet_tutorial.git
else
    https://github.com/benracine/d3_cisnet_tutorial/downloads
end
```

# Canonical Test to Ensure Installation

## Canonical Test

- Navigate to an exercise file on your hard-drive in your browser
- Open up your browser's web developer tools
- Go to console
- Enter d3 and you should see "object" in the response

# Briefly playing in the console

- navigate to http://mbostock.github.com/d3/
- let's change the color of the hyperlinks
  - open console; d3.selectAll("a").style("color","red")

# Exercise-01.html: Hello World

- just raw html (i.e. no SVG)
- include the main d3 file in line 5
  - this is the ’core’ module
  - the default build of d3.js includes the core, scale, svg and behavior modules
  - others include:
    - time
    - geo
    - csv

# Exercise-01.html: Hello World

- all d3 commands live in a unified d3 namespace
- d3 supports CSS3 notation, i.e. can select by:
  - can select by tag ("div")
  - class (".awesome")
  - containment ("parent child")
  - selectors can be intersected (".this.that" for logical AND) or unioned (".this, .that" for logical OR)
- note the difference between d3.select and d3.selectAll
- notice the method chaining has already begun
- elements can be accessed directly
  - through the each call

# Exercise-01.html: Hello World

- text is an "operator"
- operators can both get or set
- class: toggling of css classes
- style: sets the css style property, can be run w/ priority levels
- property: example, a slider value
- by default, D3 supports svg, xhtml, xlink, xml and xmlns namespaces. Additional namespaces can be registered by adding to d3.ns.prefix.

# Exercise-01.html: Hello World

- can be set as either constants or as functions
- when used to set document content, the operators return the current selection, so you can chain multiple operators together in a concise statement.
- d3.select("") $\approx$ $("") $\approx$ jQuery("")

# Exercise-02.html:: Including an SVG Element

- width and height could be related to the width and height of the window in order to be self-adjusting
- think of the svg element as a canvas with a transformed coordinate system
- g element is means of containing other svg elements
- tranform is a handy way of moving the coordinate system to a desired location
    - coord system
    - origin is the top-left
    - x is positive to the right
    - y is positive down

# Exercise-02.html:: Including an SVG Element

- svg:circle self explanatory
  - refer to the SVG spec for relevant circle attributes
- note the use of the svg namespace variable to cache a selection of interest

# Exercise-03.html: Combining with CSS Selections

- Concepts
    - CSS3 selector notation in the style section ≈ in the d3.select("") command
    - Appending is fairly self-explanatory
    - Transforms, coordinate transform to make it easier to think of
    - Note that the origin is in the top-left corner and that positive y is down.

- Jumping ahead, I will commonly account for this, not with a transform, but with an appropriate scale, which we will get to shortly
- Note: a side effect of this transform is that by not setting the x,y variables on the circle... it defaults to zero. This zero is now the point that is transformed to the middle of the page though.
- Namespaces, explain that svg:svg <– first one is a namespace, second one is the element itself svg:g is kind of like a div in html:... just a bag in which to group other things in note: you give them uniqueness through class or id
- Attr, addressed in previous slide
- Appropriate use of namespace variables

# Exercises-05.html through Exercise-08.html: Skipping for now

- d, i, this...
- event listeners can take many forms
- you can listen for different types of events
- click, mouseover, submit, etc.
- there's a subtlety of attaching to multiple functions to the same event...
- i.e. click.foo maps to one function, click.bar maps to another function

# Exercises-05.html through Exercise-08.html: Skipping for now

- exercise-05.html: skip tweens and get to data bindings
- exercise-06.html: notice that we're scaling the whole image,
- exercise-07.html: listen to user events, i.e watch the mouse move
- exercise-08.html: mouse fading events
- exercise-09.html: html-based bar-chart to emphasize that it's not just for SVG canvases

# Exercise-09.html: Bar Chart

- Bar Chart with html: Elements
- have to explain scales...
- Concepts

# Exercise-09.html: Bar Chart

- how to draw a bar graph
- the basics of appending
- Identity function
- Functional programming
- Data binding selections
- Update
- Enter
- Exit

# Exercise-11.html: 2d Array into HTML Table

- foo

# Exercise-12.html: 2d Array into SVG Bar Chart

- 2d Array into SVG Bar Chart
- he's used rangebands and ordinal scales
- RangeBands
- Linear vs. ordinal scales

# Exercise-13.html: Axes Elements

- axes elements are the stars
- do a quick little demo of changing the results to show off how flexible it is
- Axes Elements

# Extras

- Transition ≈ a non-instantaneous transformation with extra attributes:
    - Duration
    - Delay
- Ease
- Interpolate
- Tween (exercise-05.html if we get a chance)
- Call and each for control flow

# Conclusion