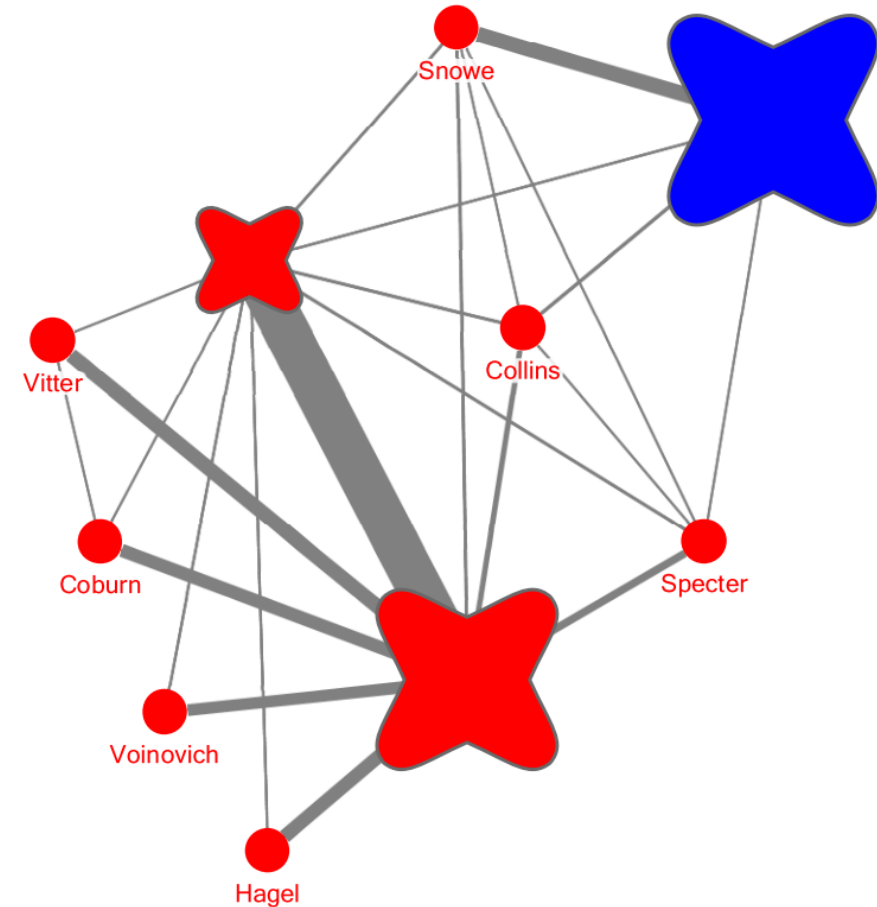
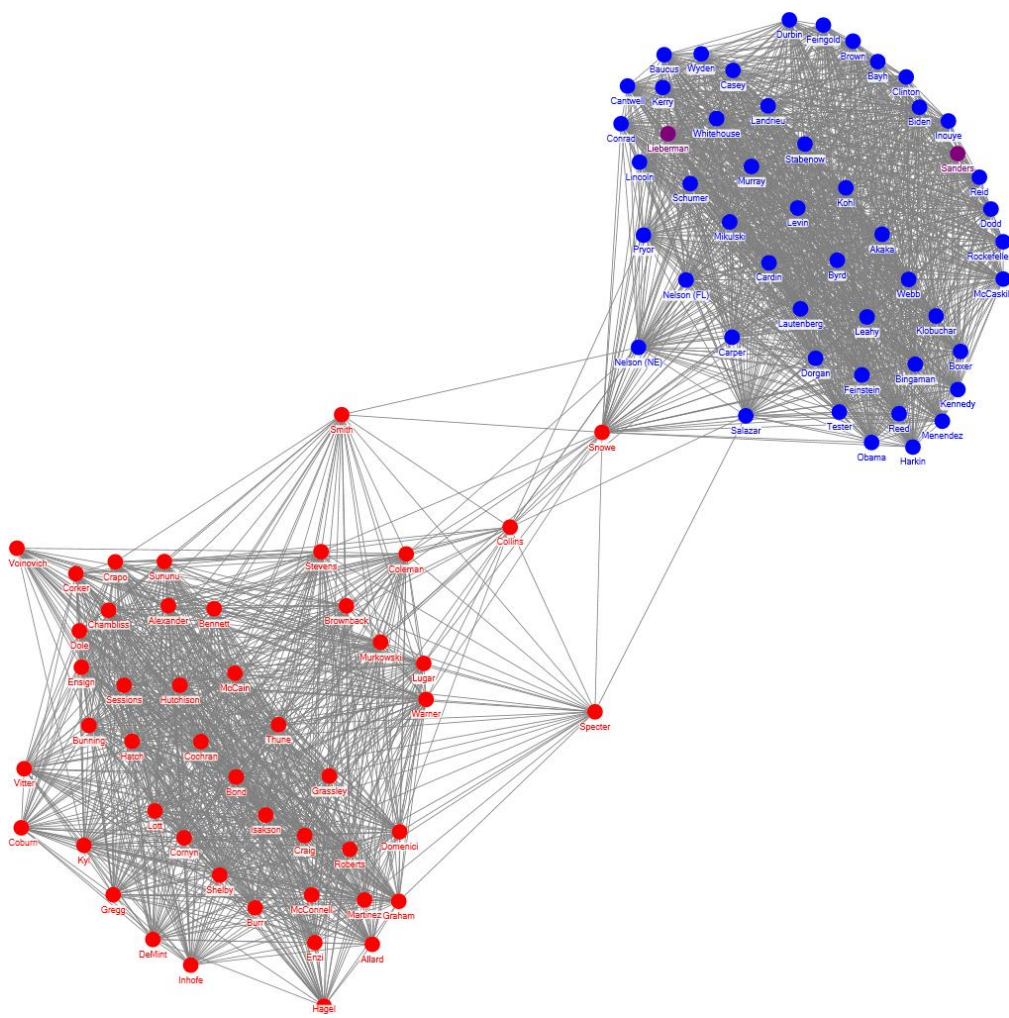


# CS 7295-01 Special Topics on Visualization in Network Science



**Professor Cody Dunne**

<https://codydunne.github.io/cs7295-f17/>

c.dunne@northeastern.edu

## **SVG**

- Elements are part of the DOM
- CSS
- Drawing with vectors
- Animations & effects built in
- Standard syntax, accessibility
- Responsive to resolution/text changes
- Export as vector graphics

## **Canvas**

- Elements drawn programmatically
- Drawing by pixels
- Animation not built in
- Can be fuzzy at other resolutions
- High-performance for many, small objects – especially on mobile
- Easy export to PNG

## **WebGL**

- ...
- Even higher performance for many, small objects

## SVG

- D3

## Canvas

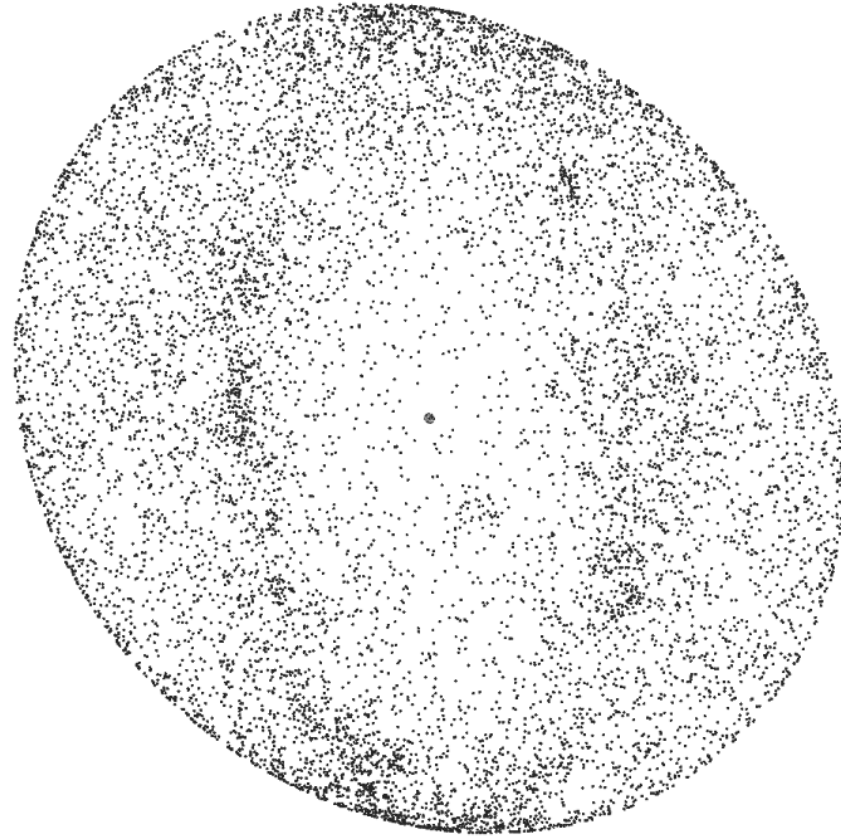
- D3 (sort of)
- Pixi.js

## WebGL

---

|                             | SVG | Canvas     | WebGL |
|-----------------------------|-----|------------|-------|
| <a href="#">D3</a>          | Y   | Sort of... | N     |
| <a href="#">Two.js</a>      | Y   | Y          | Y     |
| <a href="#">Three.js</a>    | Y   | Y          | Y     |
| <a href="#">Pixi.js</a>     | N   | Y          | Y     |
| <a href="#">NetworkCube</a> | N   | N          | Y     |

# Comparison of SVG, Canvas, WebGL, etc.

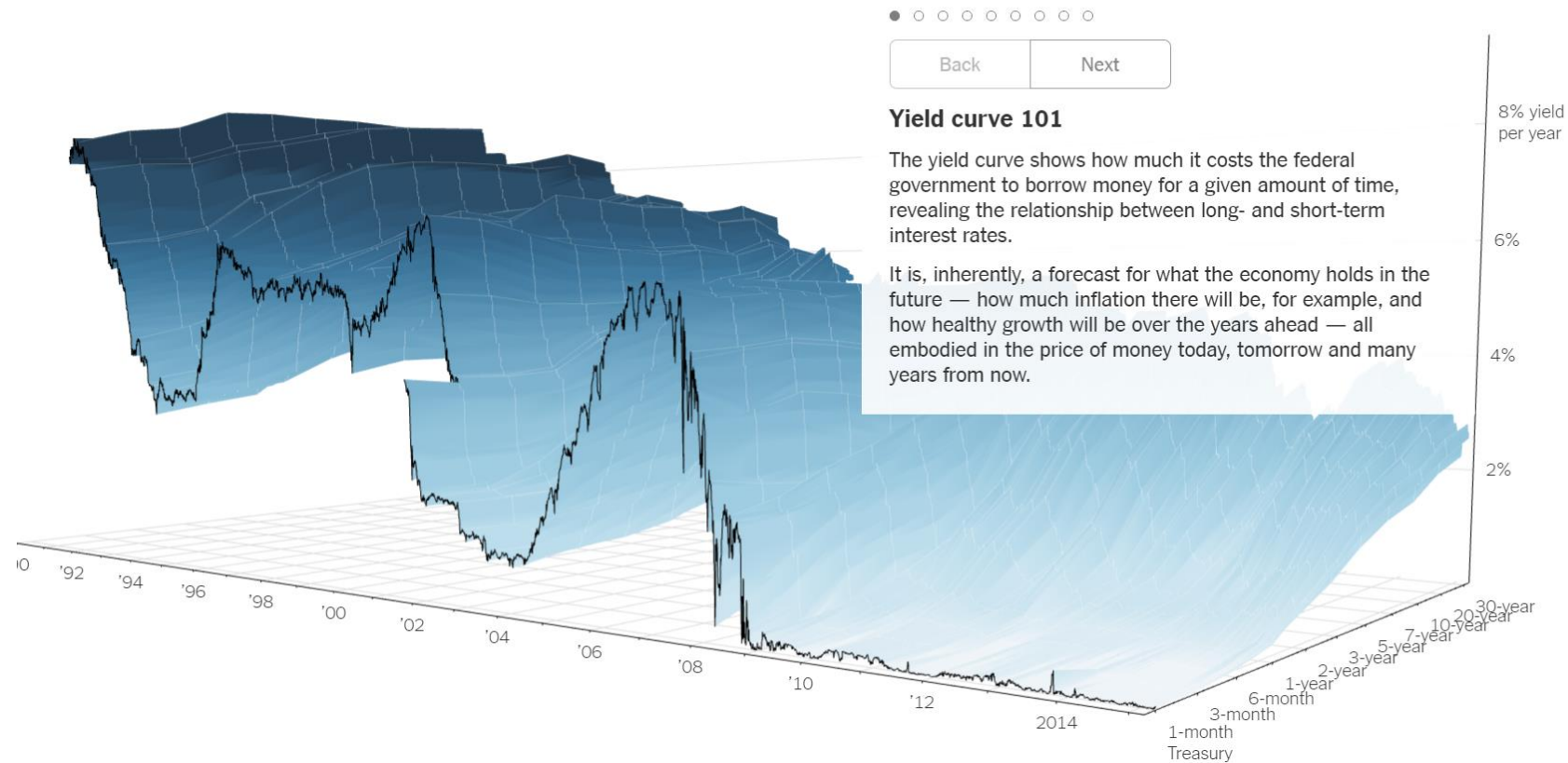


# Hybrid approaches

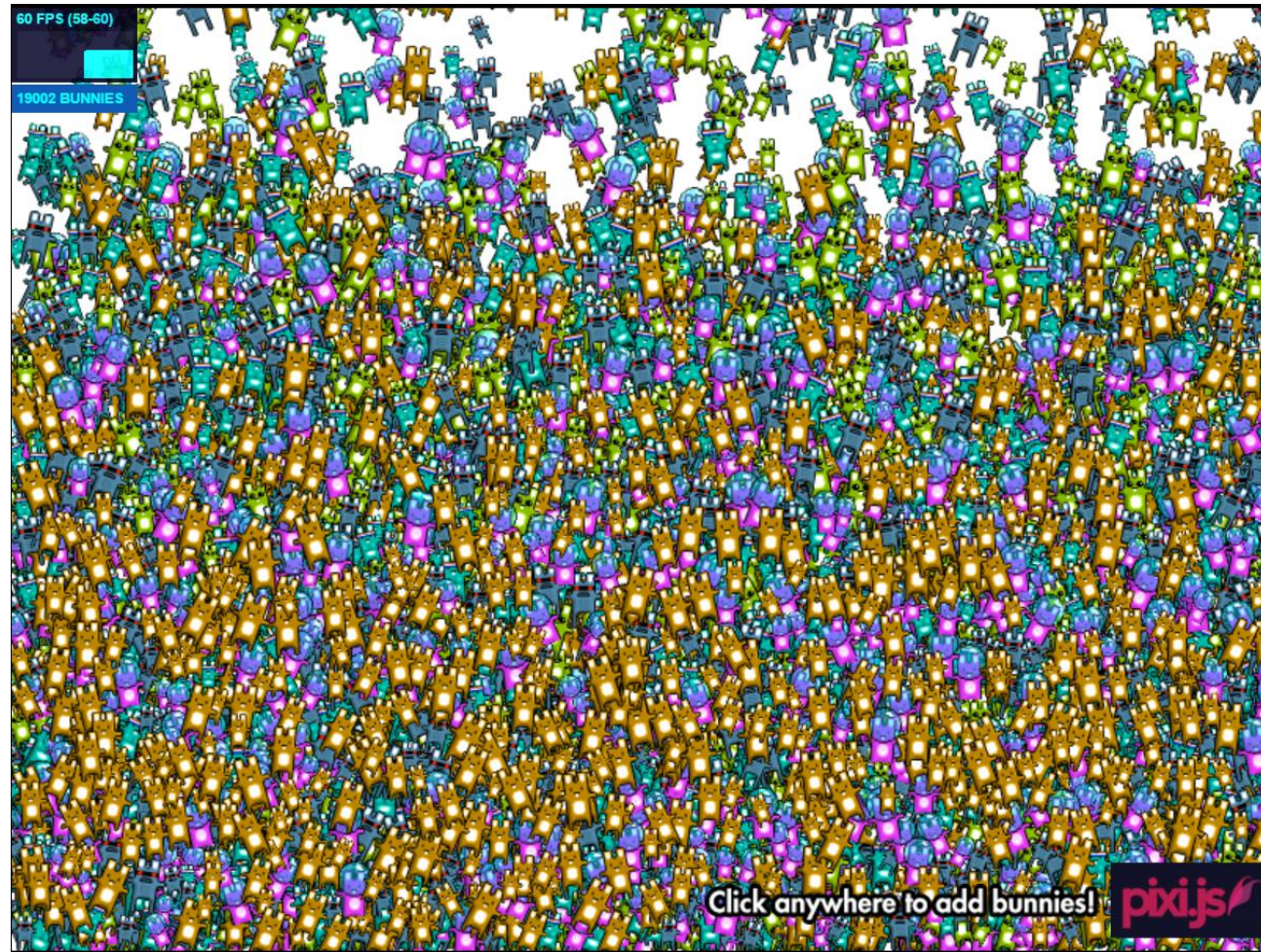
SHARE

## A 3-D View of a Chart That Predicts The Economic Future: The Yield Curve

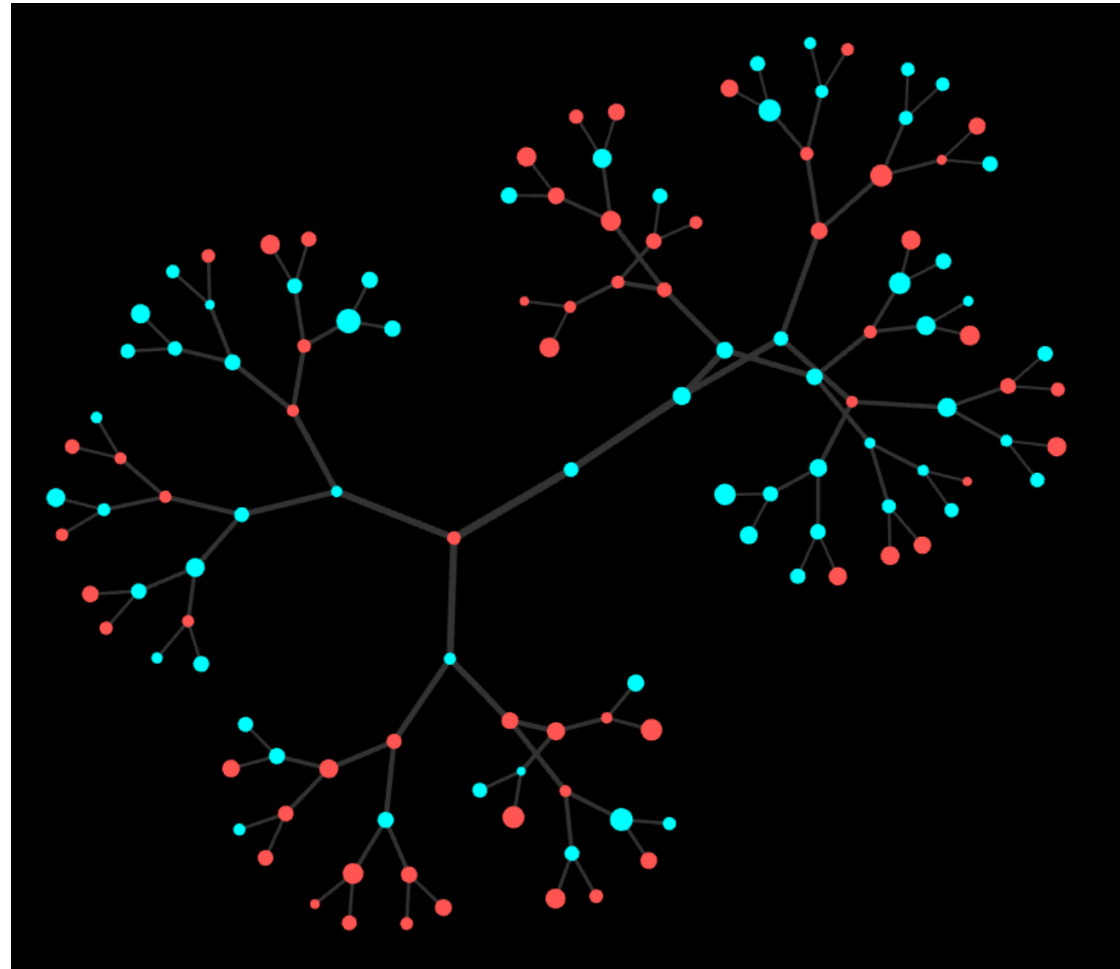
By GREGOR AISCH and AMANDA COX MARCH 18, 2015



# Pixi.js Bunnymark



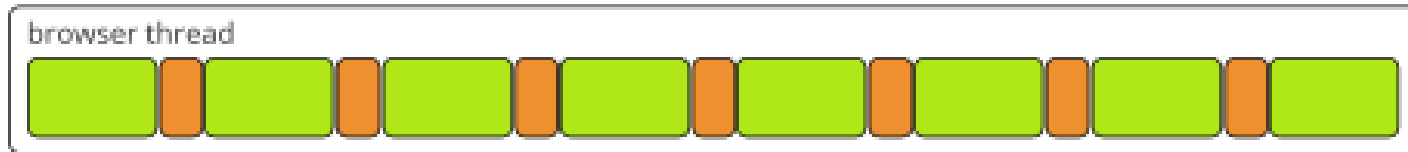
# [Ngraph](#)/[VivaGraphJS](#) + [Pixi.js](#)



# [Ngraph/VivaGraphJS](#) + [Pixi.js](#) layout in web worker

render time      layout time

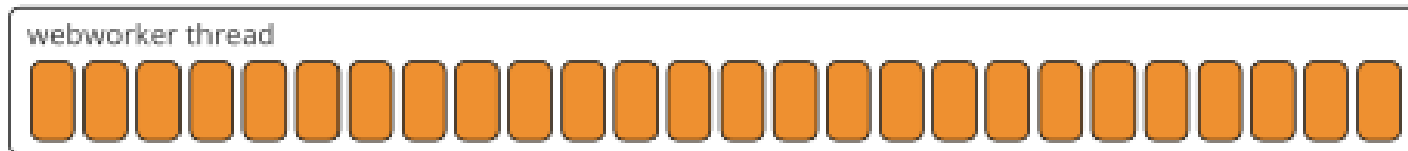
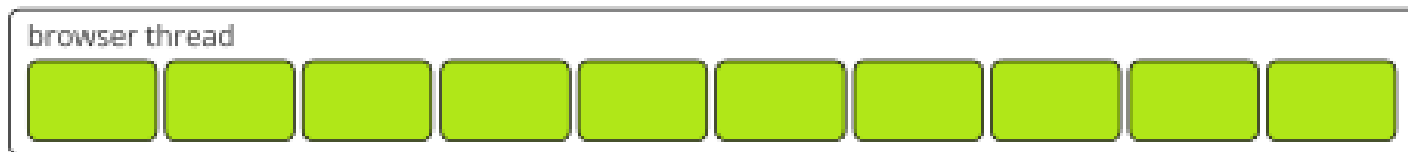
## single threaded solution



8 renders, 7 layouts

[Without web workers](#)

## multi-threaded solution



10 renders, 26 layouts

[Web workers](#)