

InfoVis Systems & Toolkits



CS 4460 – Intro. to Information Visualization
September 23, 2014
John Stasko
Presented by Yi Han

Background



- In previous classes, we have examined different techniques for presenting multivariate data
 - We'll continue to show more later too
- Today we look at systems that implement these ideas and provide some of their own new visualization techniques

Agenda



- Systems providing a view or views
 - Many Eyes, Improvise, Polaris, ...
- Toolkits that can be used to build systems
 - prefuse, Piccolo, protovis, D3, ...
- Commercial systems (to come in a future class)
 - Spotfire, InfoZoom, InfoScope, Tableau, ...

Systems/Tools



- Primarily examining academic/free systems that provide preexisting views
 - Commercial systems in a later class

Many Eyes

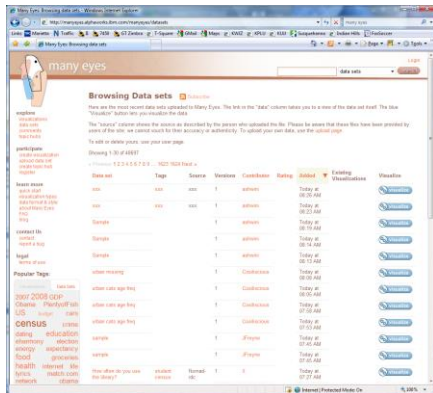


- InfoVis on the web
- Website developed from IBM's infovis group
- Motivating infovis challenges:
 - Difficulty of creating new visualizations
 - How do you discuss the visualizations?
- Project goals:
 - Enable end-user creation of visualizations
 - Foster large-scale collaborative usage

Features

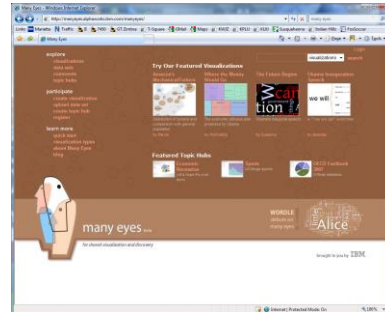


- Provides data upload capabilities and choice from library of visualizations
- Includes
 - Gallery of recently uploaded visualizations for browsing
 - Chosen highlighted visualizations
 - Attached discussion forums for each vis



Recent gallery

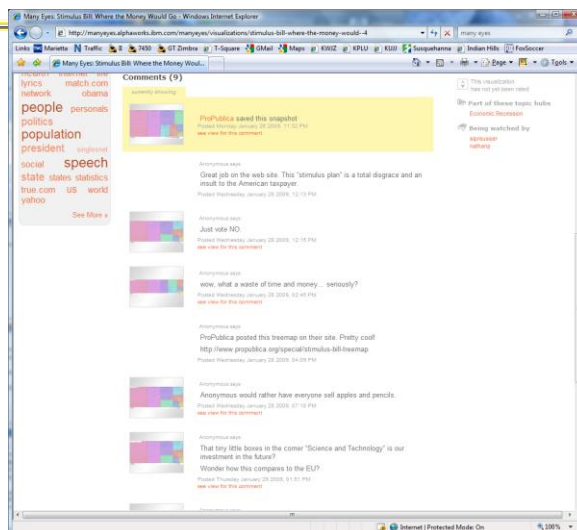
Featured visualizations



Fall 2014

CS 4460

7



Discussion forums

Fall 2014

CS 4460

8

Data



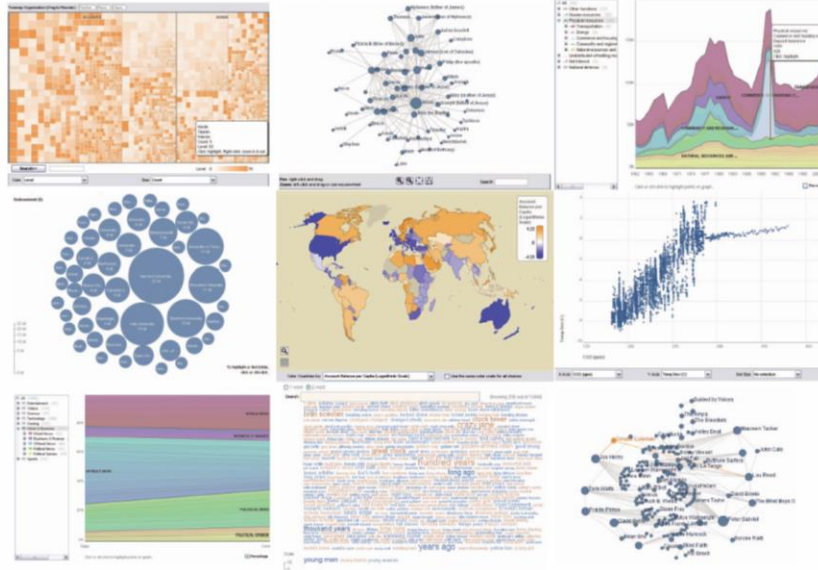
- Users upload their own data sets
 - All become public
- Format: table or unstructured text
 - Metadata allowed
- Immutable once uploaded

Visualizations



- Preloaded visualization types
 - Has grown over time
- User chooses one to combine with their data
- Provides named, typed slots that the user maps particular pieces of data to
 - System makes some reasonable guesses too

Originals

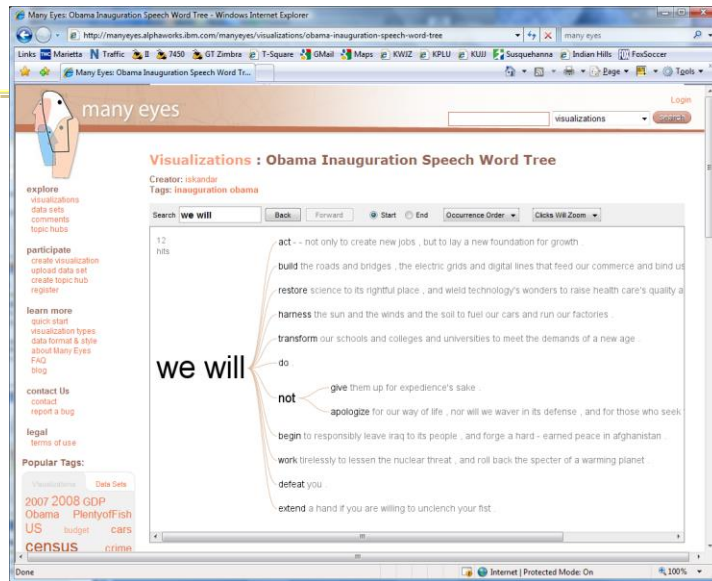


Fall 2014

CS 4460

11

WordTree



Fall 2014

CS 4460

12

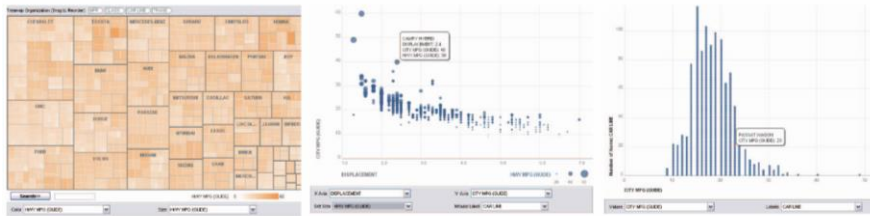


Fig. 3. Three user generated visualizations offering different perspectives on the same dataset on car fuel economy. The grey areas on the top and bottom are automatically generated by the application and allow the user to browse through different dimensions in the data.

Allows the user to control the mapping from data to image

Social Aspects



- Users identified by login ID
- Can leave comments about different visualizations
- Can take snapshot of visualization state
 - Unique URL
- “Blog this” button

Evaluation



- Quantitative, objective
- 1895 posts of March '07
- Wide variety of topics of visualizations and motivations for creating visualizations
- Does seem to be fostering discussion

Viégas et al
HICSS '08

Fall 2014

CS 4460

15

Evaluation



- Qualitative, subjective
- In-depth interviews with some ME users
- Visualizations used largely for communication and collaboration (not necessarily analysis)
 - Privacy and audience management a concern
- Highlights a number of interesting, non-expected uses of the technology

Danis et al
CHI '08

Fall 2014

CS 4460

16

Watch More



- Video of talk from Stanford HCI seminar
- <http://hci.stanford.edu/cs547/abstracts/07-08/080201-viegas-wattenberg.html>

Fall 2014

CS 4460

17

Thoughts?



- What do you think of the design choices they made?

Fall 2014

CS 4460

18

Sense.us



- Related, follow-up to Many Eyes
- Discussion and visualization of US Census data
- Go beyond Many Eyes in terms of annotation, collaboration, and discussion

Heer et al
CHI '07

Fall 2014

CS 4460

19

Components



- Doubly-linked discussion
 - Can go from visualization to threaded discussion items or vice-versa
- Graphical annotation
 - Simple graphics editor and comments (like transparent layer)
- Bookmark trail
 - Small strip of views
- Comment listings and social navigation
 - Searchable, sortable indices and links

Fall 2014

CS 4460

20

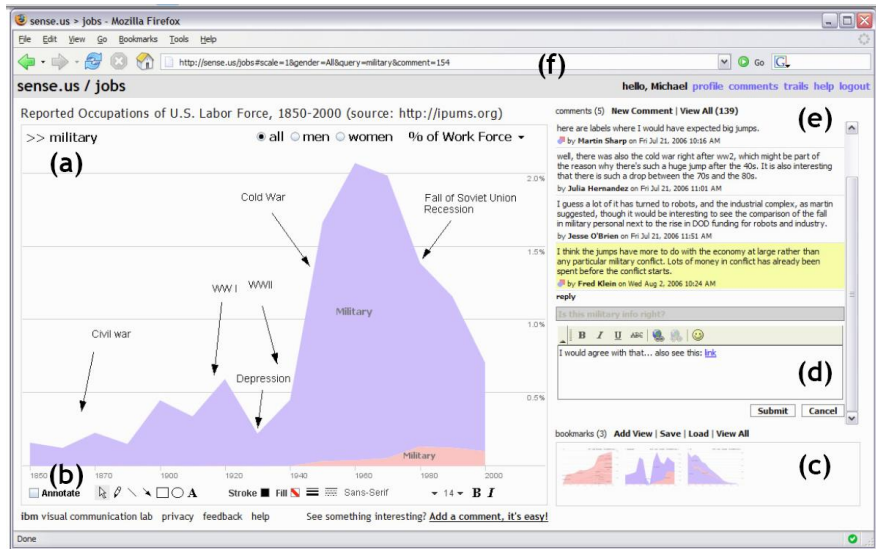


Figure 1. The sense.us collaborative visualization system. (a) An interactive visualization applet, with a graphical annotation for the currently selected comment. The visualization is a stacked time-series visualization of the U.S. labor force, broken down by gender. Here the percentage of the work force in military jobs is shown. (b) A set of graphical annotation tools. (c) A bookmark trail of saved views. (d) Text-entry field for adding comments. Bookmarks can be dragged onto the text field to add a link to that view in the comment. (e) Threaded comments attached to the current view. (f) URL for the current state of the application. The URL is updated automatically as the visualization state changes.

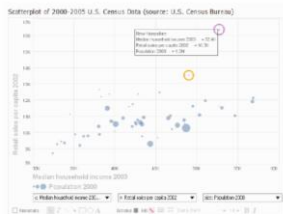


Figure 5. Scatterplot of U.S. states showing median household income (x-axis) vs. retail sales per capita (y-axis). New Hampshire and Delaware have the highest retail sales.

Video

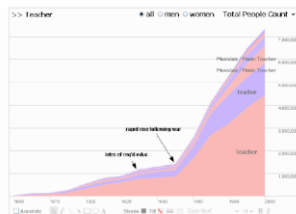


Figure 6. Visualization of the number of teachers. Annotations indicate the start of compulsory education and the rise of teachers in the post World War II era.

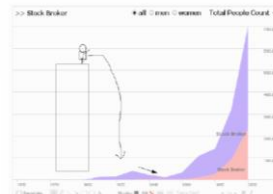


Figure 7. Annotated view of stock brokers. The attached comment reads "Great depression killed a lot of brokers".

Sample annotations

Improvise



- System used to build visualization tool
- Modular library of visualization components
- Multiple views key
- Coordination model coupled with data processing language at its heart

Weaver
InfoVis '04

Fall 2014

CS 4460

23

Key Components

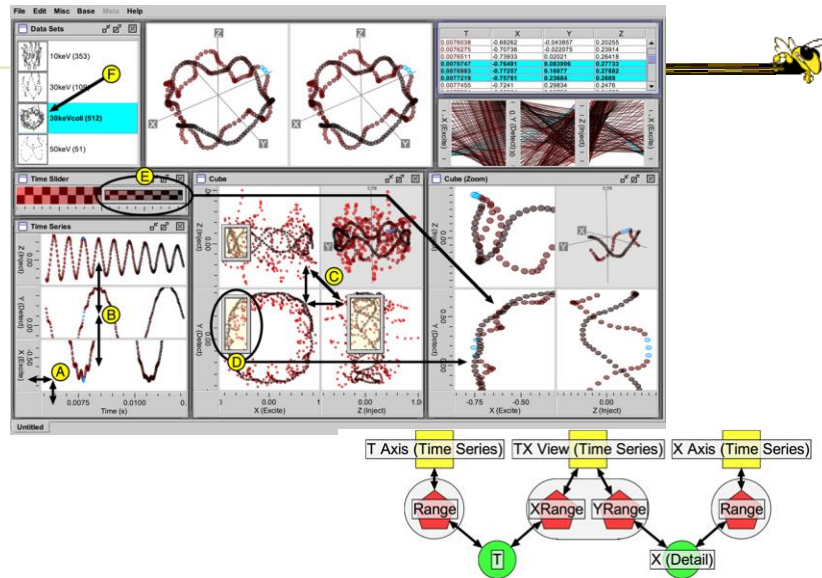


- Live Properties
 - For coordinating controls (eg, views, sliders)
 - Control defines live properties that bind to variables
- Coordinated Queries
 - Visual abstraction language
 - Visually encode data records into graphical attributes

Fall 2014

CS 4460

24



Polaris System

- Relational databases
- Pivot tables from spreadsheets
- N-dimensional data cubes
- Analytic approach is fundamental
- Provides visual representations of these concepts

Forerunner to Tableau

Interface

Database Schema: The user drags fields from the database schema to shelves to define the visual specification.

Layer Tabs: Each layer has its own tab; different transformations and mappings can be specified for each layer.

Axis Shelves: The fields placed here determine the structure of the table and the types of graphs in each table pane.

Context Menu: The context menu provides access to the data transformation and interaction capabilities of Polaris such as sorting, filtering, and aggregation.

Layer Shelf: The fields placed here determine how records are partitioned into layers.

Grouping and Sorting Shelves: The fields placed here determine how records are grouped and sorted within the table panes.

Mark Pulldown: Relations in each pane are mapped to marks of the selected type.

Retinal Property Shelves: The fields placed here determine how data is encoded in the retinal properties of the marks.

Legends: Legends enable the user to see and modify the mappings from data to retinal properties.

Video

Visualization

- Table of data (rows, columns)
- Each axis may have nested dimensions
- Table entry is a pane, and has visual marks to represent data
- Analyst drags and drops fields from database schema onto shelves of display
- Much interaction supported

Graphics



- Formal table algebra provided to describe data
- Visual mappings established from data types to appropriate (good) markings and encodings

Interaction



- Four main types
 - Deriving additional fields
 - Sorting and filtering
 - Brushing and tooltips
 - Undo and redo

Toolkits & Infrastructures



- Set of components or capabilities that allow others to put together visualization systems
- Growing trend

Fall 2014

CS 4460

31

Toolkit Design



- What would you include in a toolkit like this if you designed it?

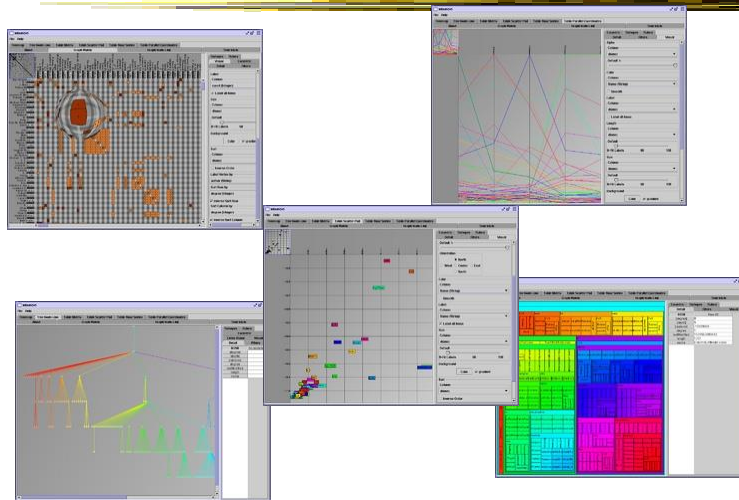
Fall 2014

CS 4460

32

InfoVis Toolkit

<http://ivtk.sourceforge.net>



Fekete
InfoVis '04

Fall 2014

CS 4460

33

Characteristics

- Coherent software architecture and set of Java components
 - Data structure is a table
 - Views: time series, parallel coordinates, scatterplots, node-link diagrams, treemaps
 - Added capabilities in color management, labeling, dynamic queries, ...

Fall 2014

CS 4460

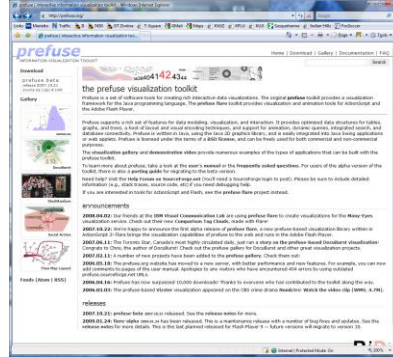
34

Prefuse



- Toolkit in Java for building information visualizations

Beginning in a series of toolkits from Jeff Heer and his research group



Heer et al
CHI '05

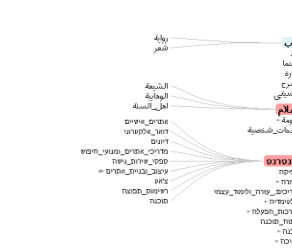
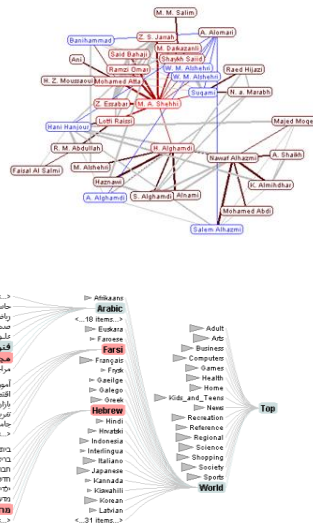
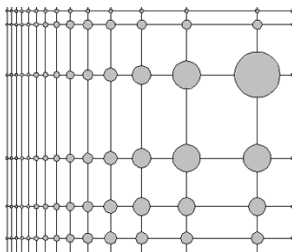
Fall 2014

CS 4460

35

Prefuse

<http://prefuse.org>



Demo
Video

Fall 2014

CS 4460

36

Characteristics



- Fine-grained building blocks for constructing visualizations (as opposed to pre-made views)
- Data model is a graph (entities & relations)
- Includes library of layout algorithms, navigation and interaction techniques
- Written in Java2d
- Follows CMS vis process model

Fall 2014

CS 4460

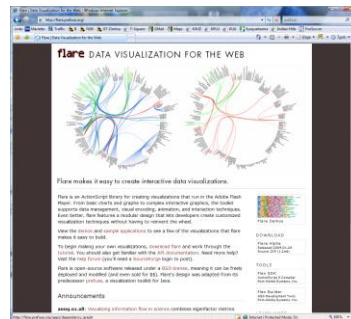
37

<http://flare.prefuse.org>

Flare



- Follow-on of prefuse implemented in ActionScript for running in Flash player



Demos

Fall 2014

CS 4460

38

Protovis

<http://vis.stanford.edu/protovis>



- Subsequent toolkit following a declarative model
 - Simple grammar of graphical primitives called *marks*
 - Marks are associated with data, using mapping to physical *properties*
 - Properties can be dynamic or static
 - Register *event handlers* to provide interactivity

Bostock & Heer
TVCG (InfoVis) '09

Heer & Bostock
TVCG (InfoVis) '10

Fall 2014

CS 4460

39

Built-in Mark Types

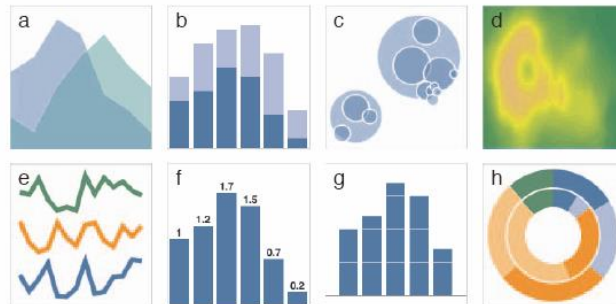


Fig. 5. Examples of built-in mark types. (a-h) Area; Bar; Dot; Image; Line; Label and Bar; Rule and Bar; Wedge.

Fall 2014

CS 4460

40

Examples



```
new pv.Panel().canvas("fig3a")
  .add(pv.Bar)
  .data([1, 1.2, 1.7, 1.5, .7, .2])
  .bottom(0).width(20)
  .height(function(d) d * 80)
  .left(function() this.index * 25)
  .root.render();
```



```
new pv.Panel().canvas("fig3b")
  .data([[1, 1.2, 1.7, 1.5, .7],
        [.5, 1, .8, 1.1, 1.3],
        [.2, .5, .8, .9, 1]])
  .add(pv.Area)
  .data(function(d) d)
  .fillStyle(pv.Colors.category19.parent)
  .bottom(function() let (c = this.cousin())
    c ? (c.bottom + c.height) : 0)
  .height(function(d) d * 40)
  .left(function() this.index * 35)
  .root.render();
```

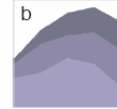
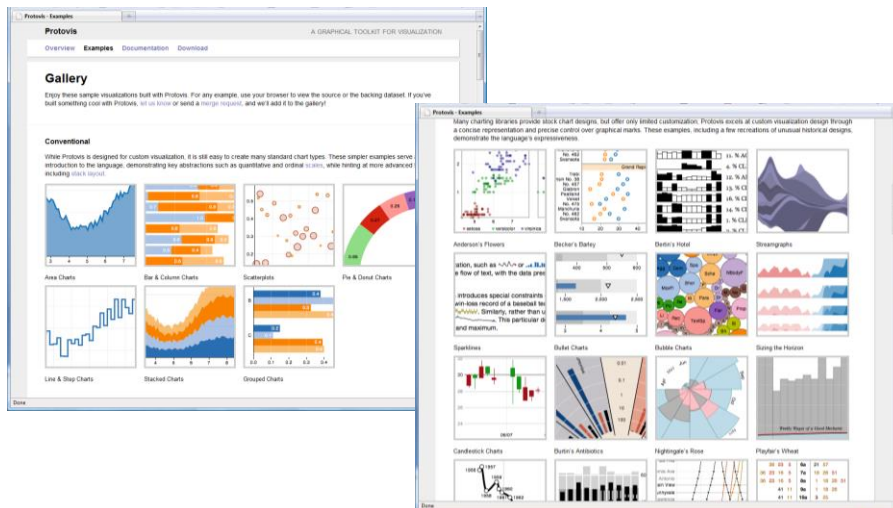


Fig. 2. Specifying two simple charts. (a) Bar. (b) Stacked area.

Gallery



D³: Data-Driven Documents



- Subsequent entry in the Heer-Bostock line of toolkits
- “Not just an infovis toolkit”
- Javascript-based
- Very similar to Protovis...
 - Except makes use explicitly of web standards such as Scalable Vector Graphics (SVG) rather than a proprietary “marks” graphics set

Fall 2014

CS 4460

43

D3 Design Pattern



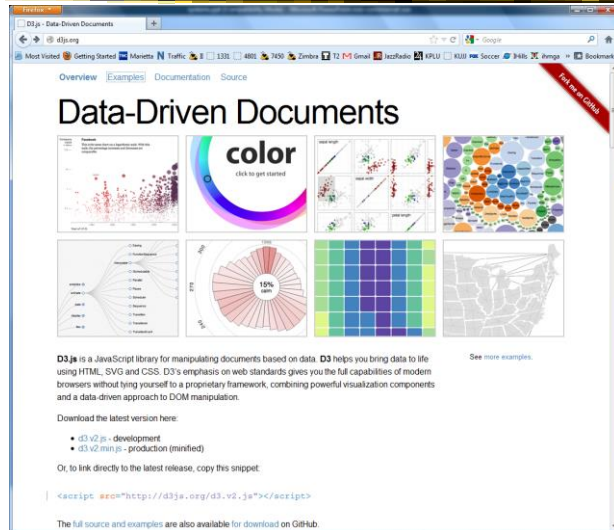
- Declarative Syntax like Protovis
- Creating/Modifying selections of the HTML DOM
- “An elegant for-loop with a bunch of useful helper functions”
- Excellent support for changing data
 - Taking advantage of CSS3 Transformations and Transitions
- Integrates seamlessly into any webpage

Fall 2014

CS 4460

44

D3 Website



Fall 2014

CS 4460

45

D3 Examples



- <https://github.com/mbostock/d3/wiki/Gallery>
- "Show Reel": <http://bl.ocks.org/1256572>
- Voronoi Diagram: <http://mbostock.github.com/d3/ex/voronoi.html>
- Force-Directed Graph: <http://mbostock.github.com/d3/ex/force.html>
- Box Plots: <http://mbostock.github.com/d3/ex/box.html>

Fall 2014

CS 4460

46

D3 Support



- Active community online
 - <https://github.com/mbostock/d3/wiki>
 - Including Mike Bostock often answering questions
- In active development
 - Though the critical features are unlikely to change

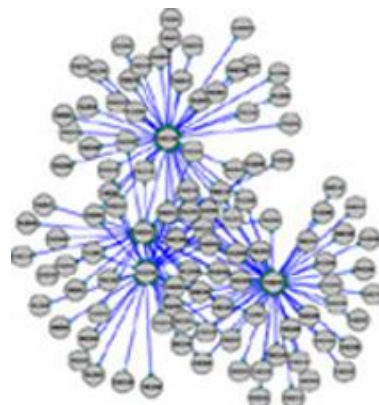
Fall 2014

CS 4460

47

Piccolo

<http://code.google.com/p/piccolo2d>



Fall 2014

CS 4460

48

Characteristics



- Graphics toolkit with built-in zooming and panning support
- Very nice for implementing infovis
- Will discuss more later in course...

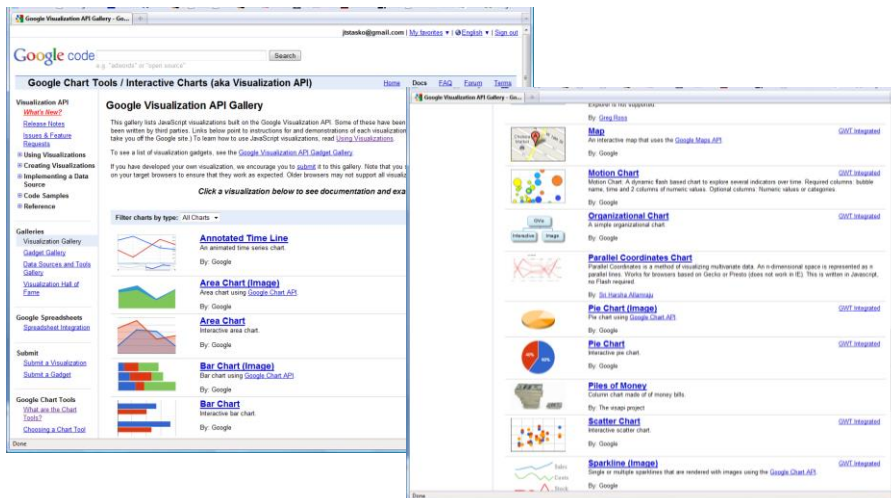
Fall 2014

CS 4460

49

<http://code.google.com/apis/visualization/documentation/>

Google Chart Tools



Fall 2014

CS 4460

50

Characteristics



- Javascript-based
- Gallery of contributed code segments
- Visualizations are interactive
- Evolving API

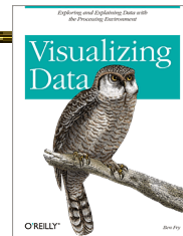
Fall 2014

CS 4460

51

Processing

<http://processing.org>



- Java based
- Unlike prior toolkits, not specifically designed for InfoVis
 - Data Reader? Layout algorithm?
 - But can definitely be used to build visualizations!
- Well documented, lots of tutorials with contributions from many people and even books

Ben Fry

Fall 2014

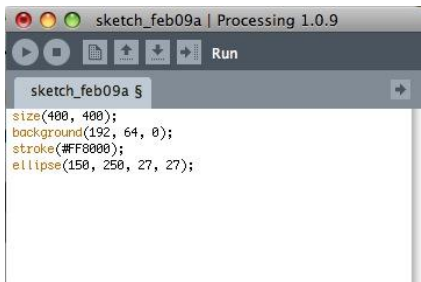
CS 4460

52

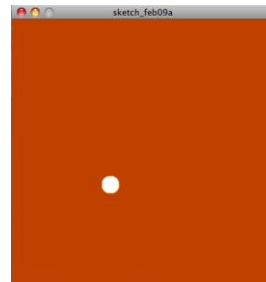
Processing: the idea



- Programming as scripting
 - PDE: processing development environment
 - A program is called a *sketch*
 - written as a list of statements

A screenshot of the Processing IDE window titled "sketch_feb09a | Processing 1.0.9". The window has a toolbar with icons for play, save, undo, redo, and run. Below the toolbar is a text area containing the following code:

```
sketch_feb09a $  
size(400, 400);  
background(192, 64, 0);  
stroke(#FF0000);  
ellipse(150, 250, 27, 27);
```



Fall 2014

CS 4460

53

The Pain of using Java2D



- Graphics repaint
- Animation
- ActionListener, EventListener ...
- Long-winded Java syntax

- Processing makes many of these things much easier

Fall 2014

CS 4460

54

Processing: Useful Functions



```
void setup() {  
  //your own code here  
}
```

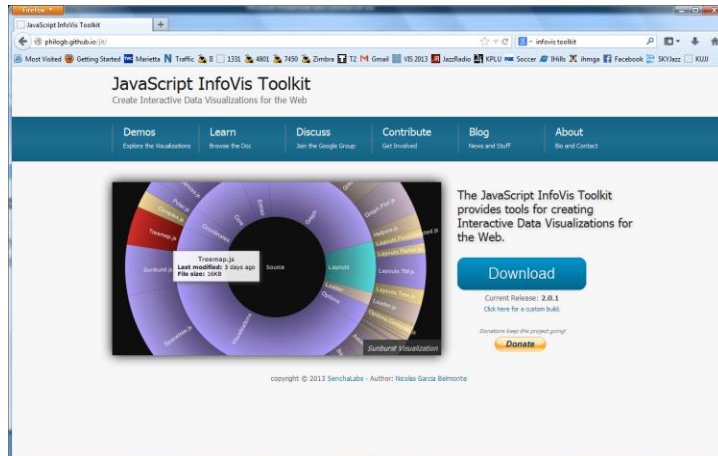
```
void draw() {  
  //your own code here  
}
```

- These are built-in functions that are called automatically.
 - The setup() block runs once.
 - The draw() block runs repeatedly: good for animation



Some new toolkits

JavaScript InfoVis Toolkit



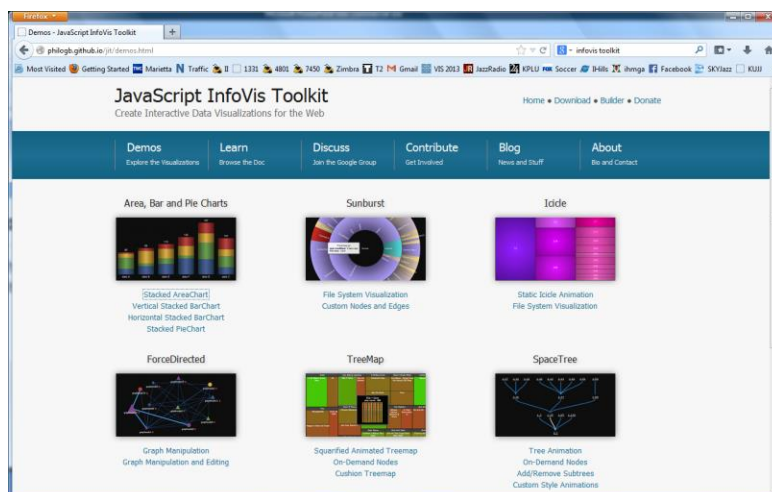
<http://philogb.github.io/jit/>

Fall 2014

CS 4460

57

Built-in Visualizations



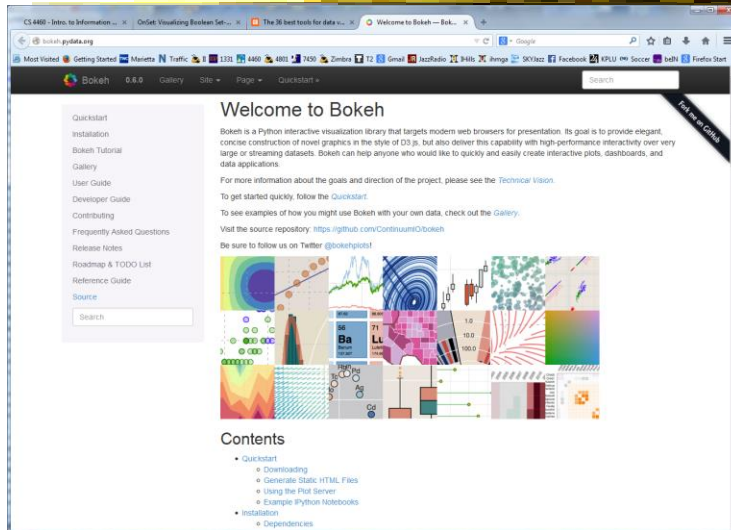
Fall 2014

CS 4460

58

Bokeh

<http://bokeh.pydata.org>



Python

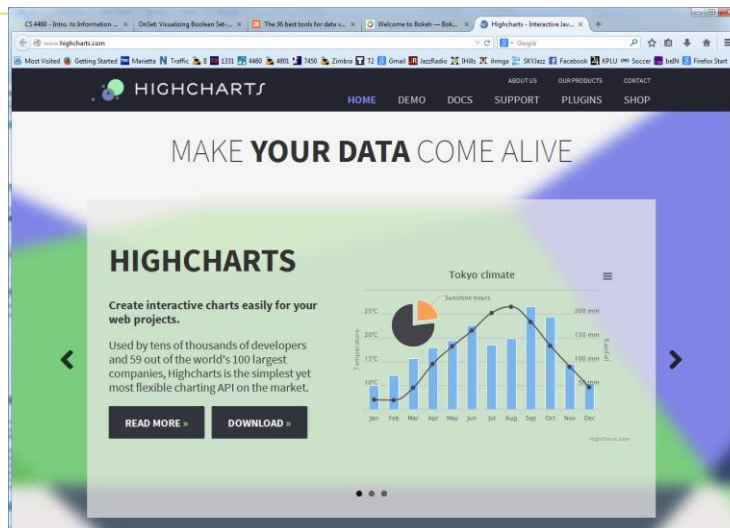
Fall 2014

CS 4460

59

Highcharts

<http://www.highcharts.com/>



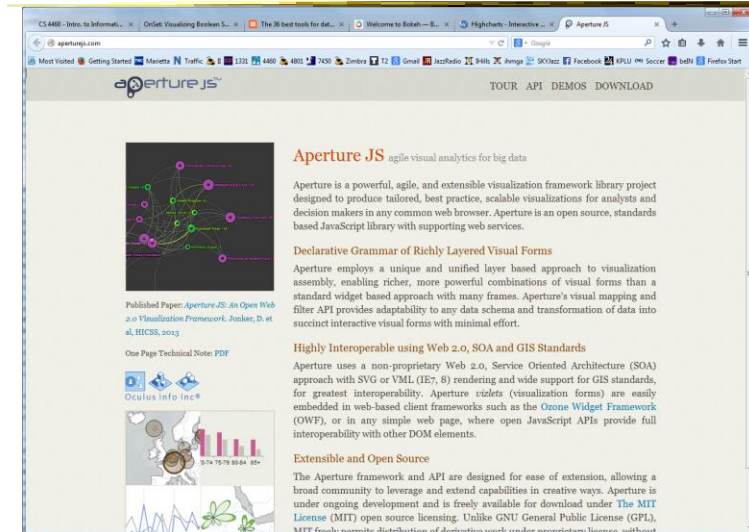
Fall 2014

CS 4460

60

Aperture.js

<http://aperturejs.com>



Fall 2014

CS 4460

61

Design Patterns

- Heer used his experience with prefuse etc to describe common design patterns for infovis

- Reference model
- Data column
- Cascaded table
- Relational graph
- Proxy tuple
- Expression
- Scheduler
- Operator
- Renderer
- Production rule
- Camera
- Dynamic query binding

Heer & Agrawala
TVCG (InfoVis) '06

Fall 2014

CS 4460

62

Reflection



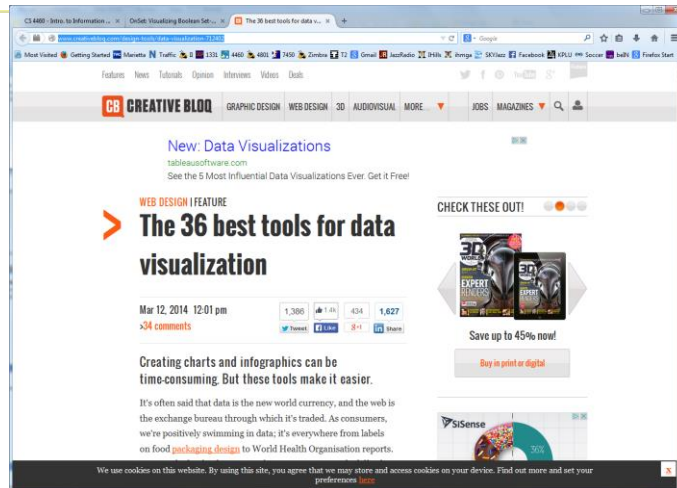
- What would you seek in a good infovis system or toolkit?

Toolkit Tradeoffs



- InfoVis-focused
 - Many fundamental techniques built-in
 - Can be faster to get something going
 - Often more difficult to implement something “different”
 - Doc is often poor
- Generic graphics
 - More flexible
 - Can customize better
 - Big learning curve
 - Doc is often better
 - Can take a long time to (re)implement basic techniques

Nice Summary



<http://www.creativebloq.com/design-tools/data-visualization-712402>

Fall 2014

CS 4460

65

More in the Future...

- Demos of commercial infovis systems
 - Spotfire
 - Table Lens/Eureka
 - SeeIt
 - InfoZoom
 - InfoScope
 - Tableau

Fall 2014

CS 4460

66

HW 2 Feedback



- Things we noticed
- Example solutions

Fall 2014

CS 4460

67

HW 3 Ongoing



- D3
- Any questions?

Fall 2014

CS 4460

68

HW 4



- Write one potential exam question
 - Short answer, sentence to about a paragraph
 - Can use a vis
 - Not multiple choice, T/F
- Include the answer

- Due next Tuesday

Fall 2014

CS 4460

69

Upcoming



- Guest lecture: Prof. Rahul Basole
 - Reading

- Design session & exam review

Fall 2014

CS 4460

70



Additional material

Fall 2014

CS 4460

71

Improvise Application



- Understand spatial and temporal patterns of movements of people and groups of people, understand commerce and travel patterns
- Focus: Social networks of people and historical data
- Alternate: Understand actions and movements of people in terrorist groups

Weaver et al
Information Visualization '07

Fall 2014

CS 4460

72

Particulars



Figure 1 The National Hotel in Roaring Spring, PA.

- Who – Historical geographers, geneologists
- Problem – Provide platform where these people can do research and more easily understand the movement patterns
- Data – Old hotel records & registers

Data Collection



Figure 2 The Rebersburg Hotel Register.

- Two hotel registers in PA
 - ~10k guest entries
 - Hundreds of place of origin (mostly NE)
- Transcribed into spreadsheets (400 hours)
- Many repeat visitors

- 6% of dataset not legible/usable

Preliminary Data Analysis



- Simple tables generated
- GIS software for mapping geographic locations

Fall 2014

CS 4460

75

Visualization

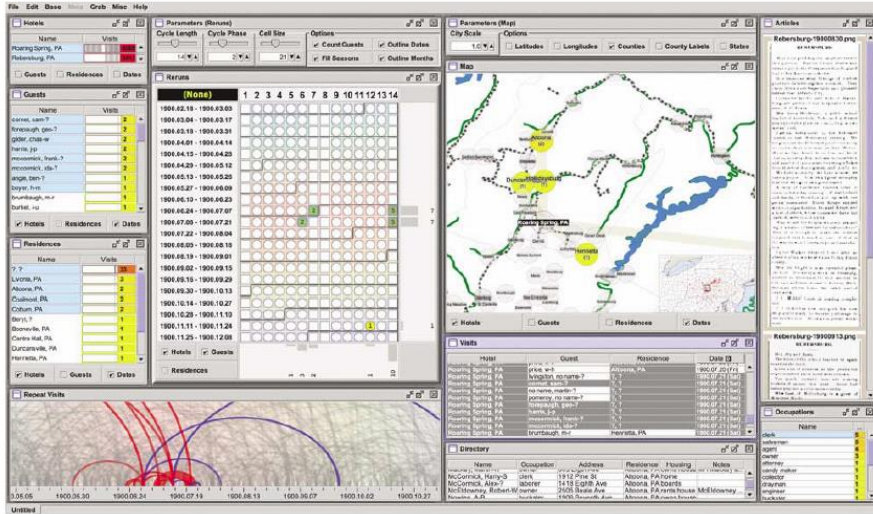


- How?
 - Consult with historical geographers
 - Iterative design
- Multiple, coordinated views

Fall 2014

CS 4460

76

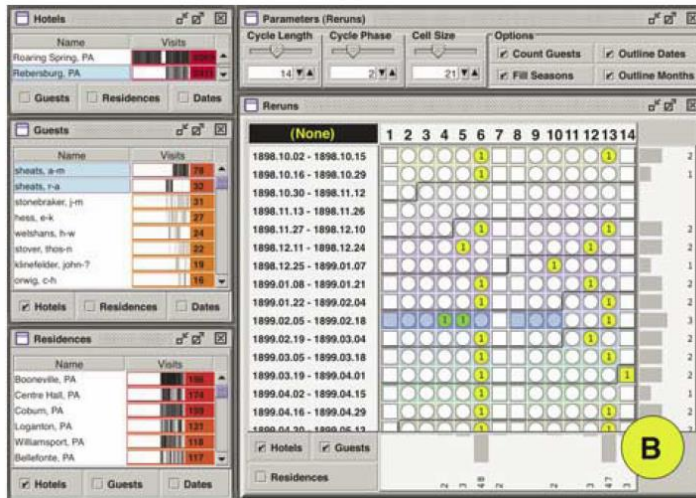


User interface

Fall 2014

CS 4460

77

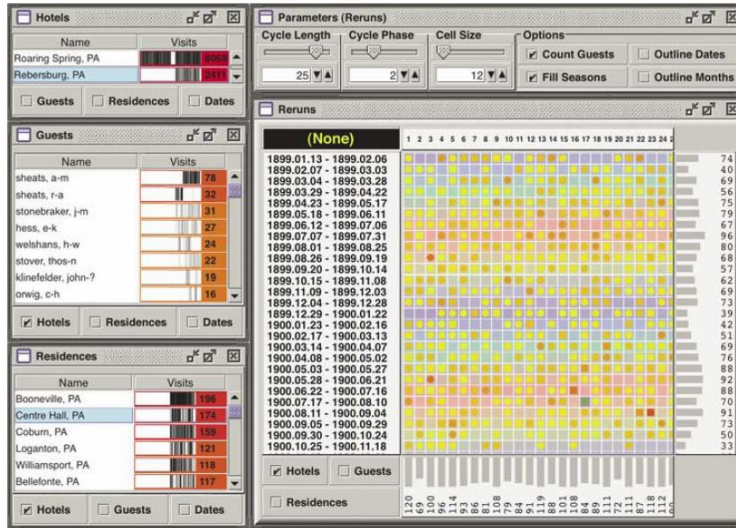


Looking for patterns of visitation

Fall 2014

CS 4460

78



Showing time of year of visits

Fall 2014

CS 4460

79

Video



Fall 2014

CS 4460

80

Data Analysis



- Through interaction with views, analysts can pose queries and look for answers
- “For these people on this date, did they ever meet prior to that?”
- Can also ask questions for unknown people (match a ?)

Fall 2014

CS 4460

81

Data Attributes Explored



- Weekly circuits
- Weather effects
- Biweekly circuits
- Particular hotel characteristics
- Groups & gatherings
- Non-periodic regular visits

Fall 2014

CS 4460

82

Microsoft Pivot



- System unveiled to great publicity from LiveLabs
 - After shut-down, has mostly disappeared
- Faceted browsing/
filtering with image-
based UI

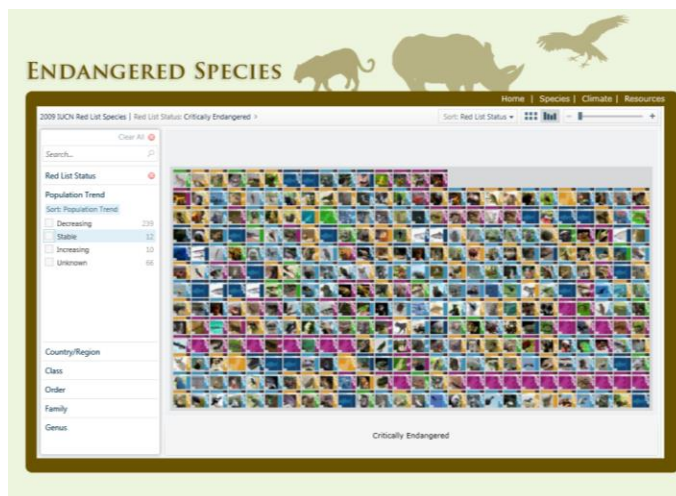


Fall 2014

CS 4460

83

Demo



Fall 2014

CS 4460

84