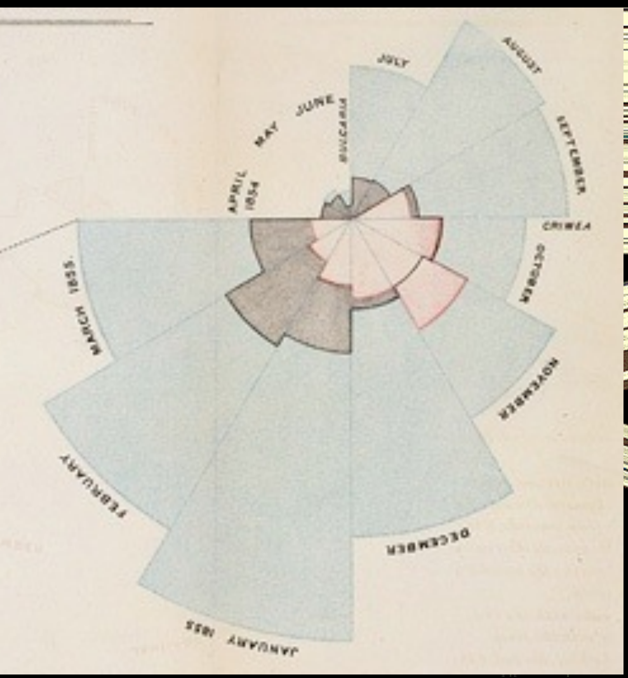
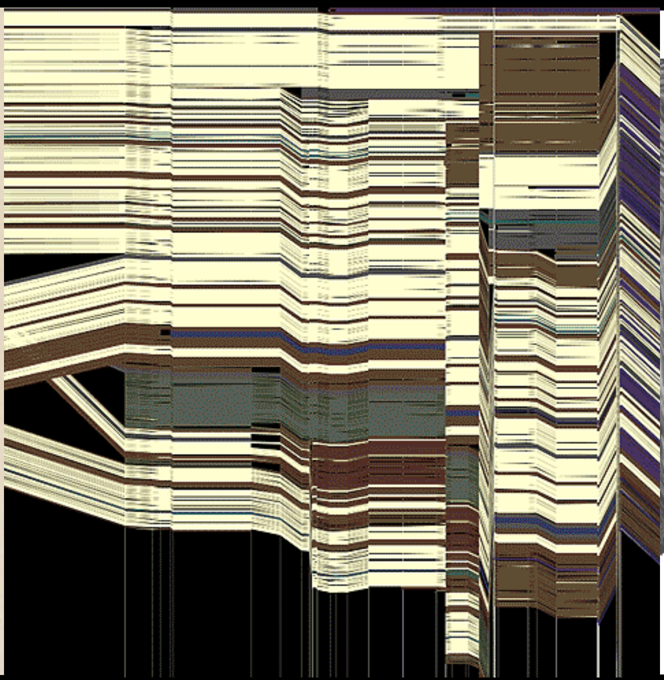
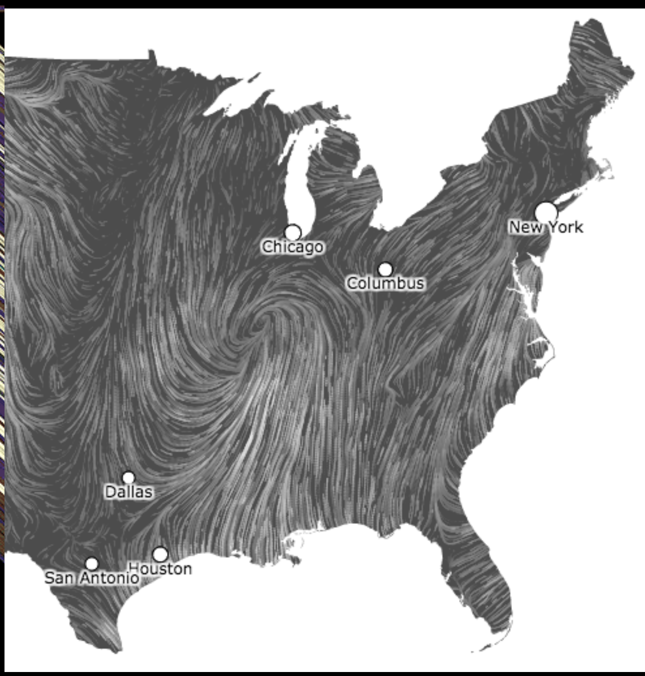
CSE 442 - Data Visualization

Visualization Tools







Matthew Conlen University of Washington

How do people create visualizations?

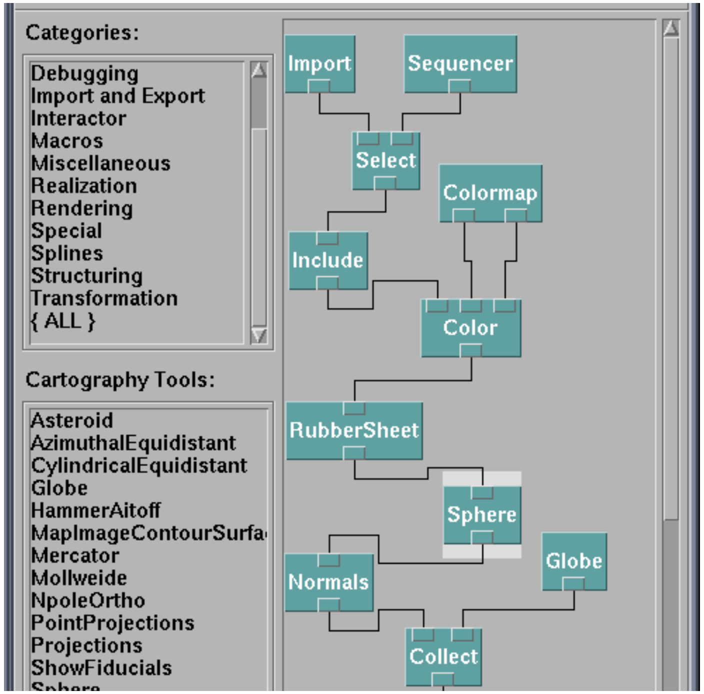


Chart Typology

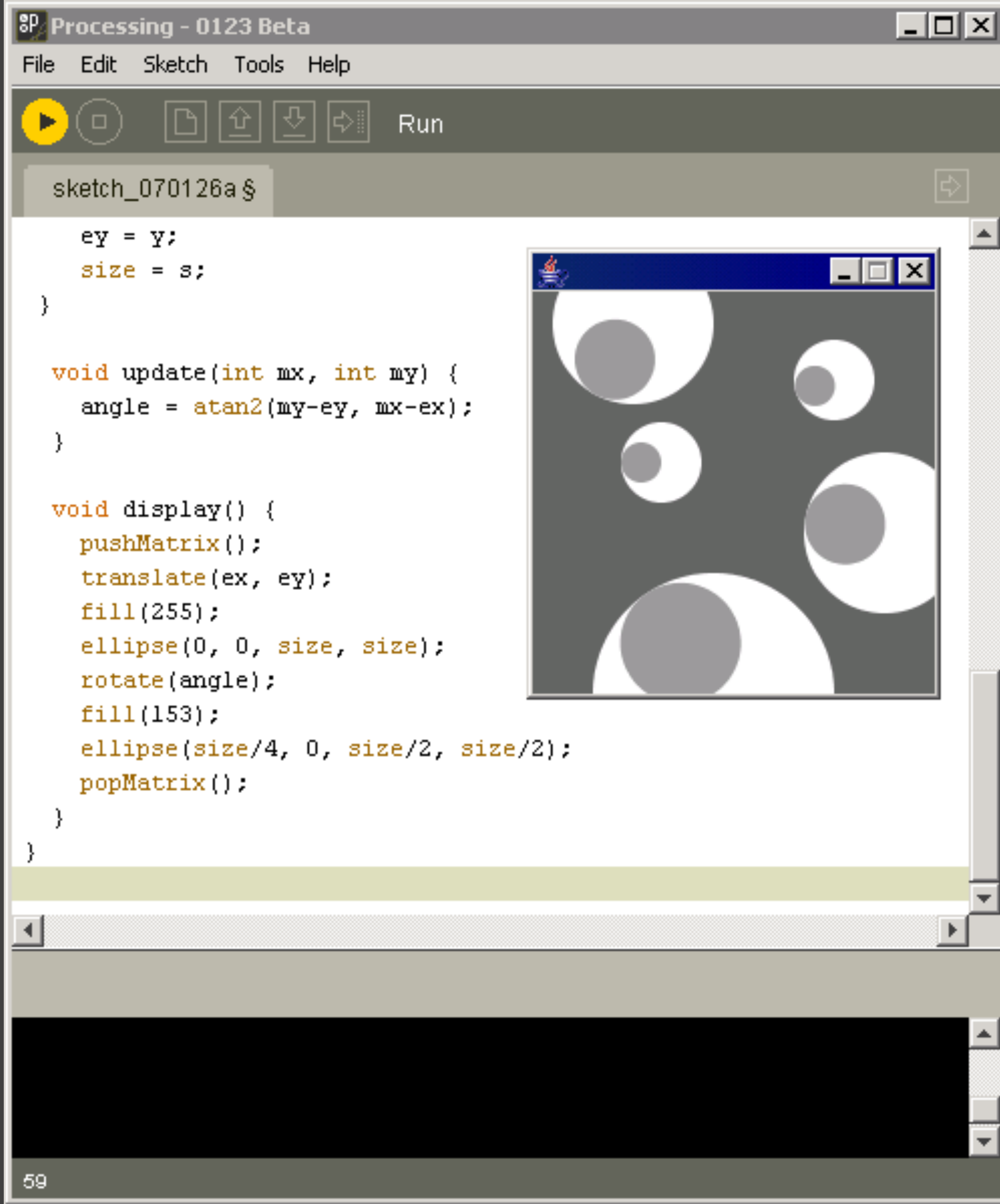
Pick from a stock of templates
Easy-to-use but limited expressiveness
Prohibits novel designs, new data types

Component Architecture

Permits more combinatorial possibilities
Novel views require new operators,
which requires software engineering

Graphics APIs

Processing, OpenGL, Java2D



http://processing.org

US Air Traffic, Aaron Koblin

Graphics APIs

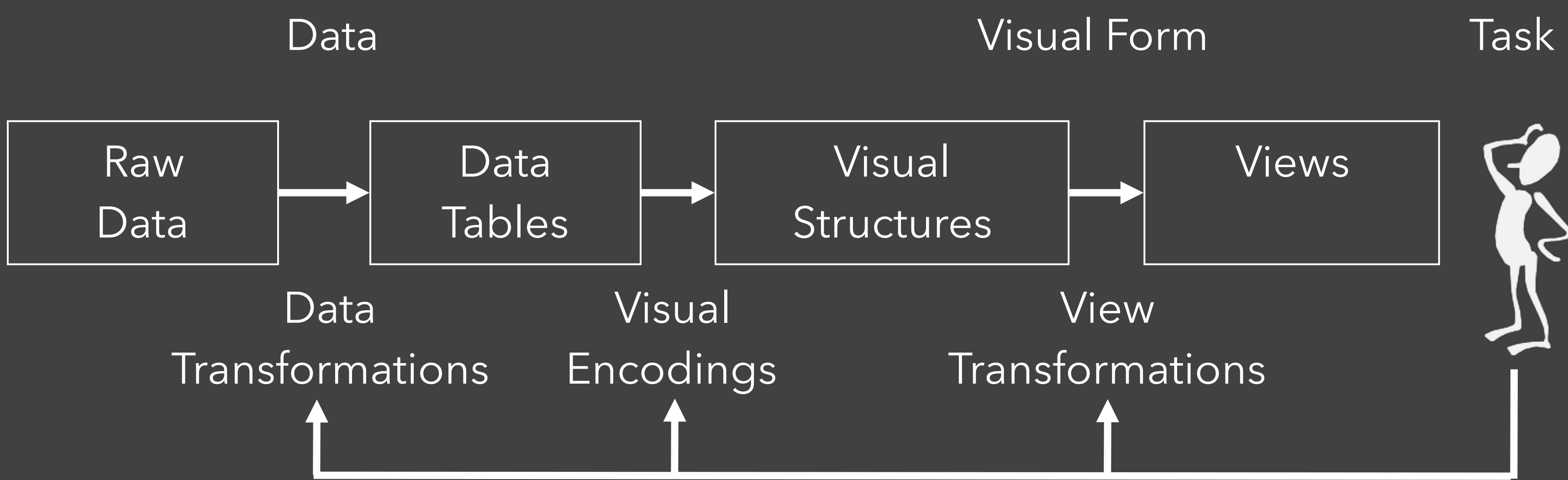
Processing, OpenGL, Java2D

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

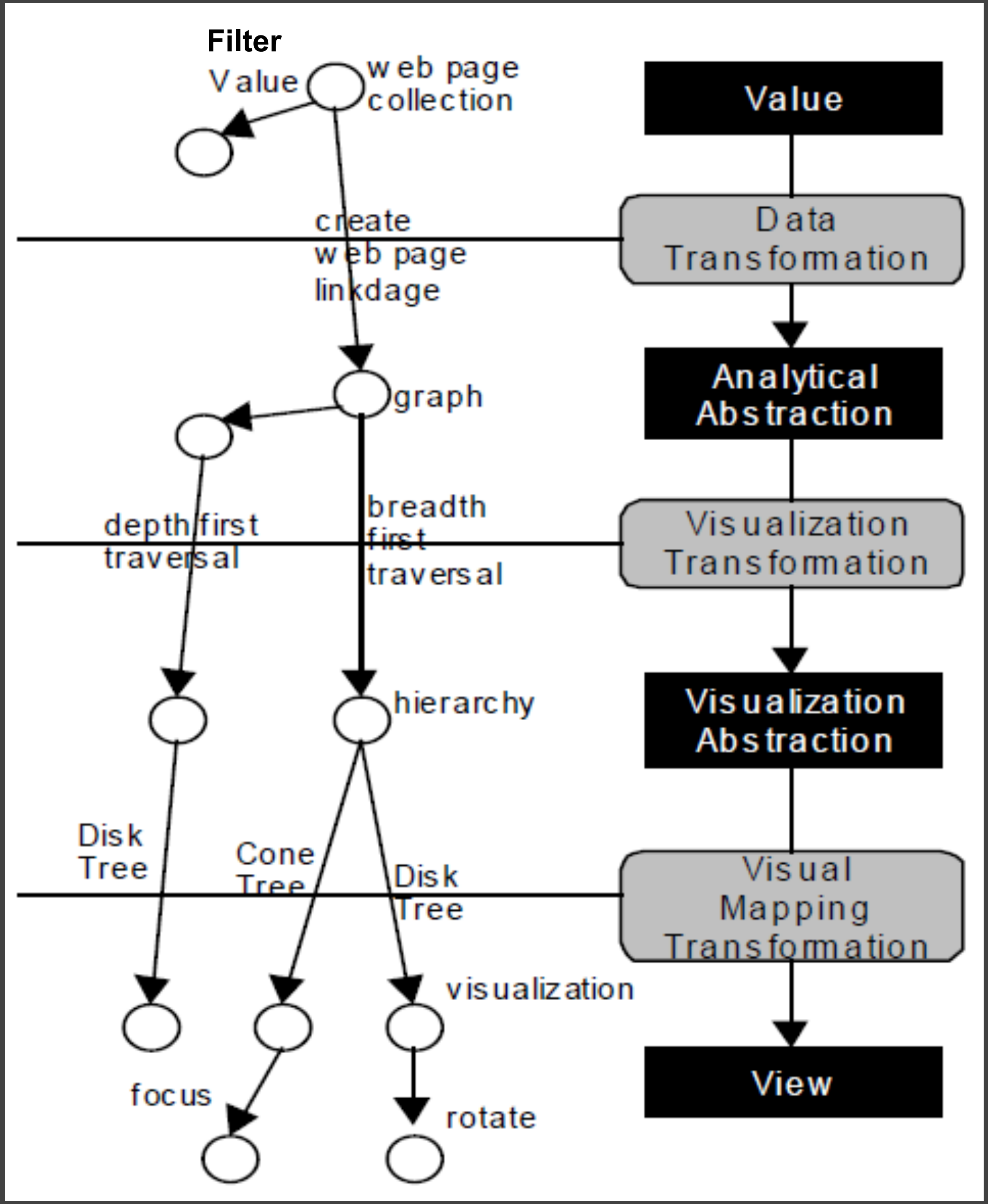
Processing, OpenGL, Java2D



Data
Visual Form
Task
Raw Data
Data Tables
Visual Structures
Views
Data Transformations
Visual Encodings
View Transformations

Data State Model

[Chi 98]



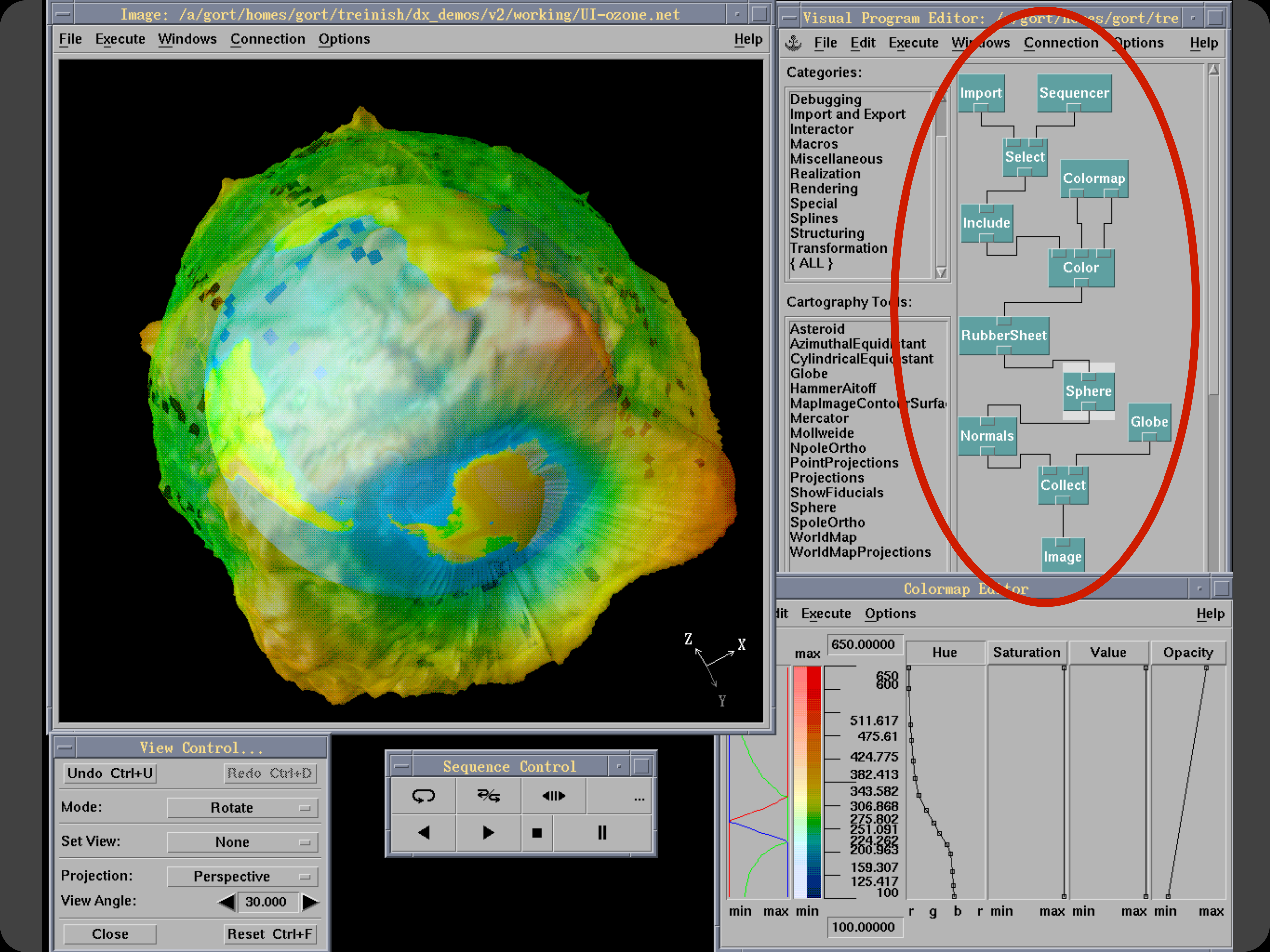
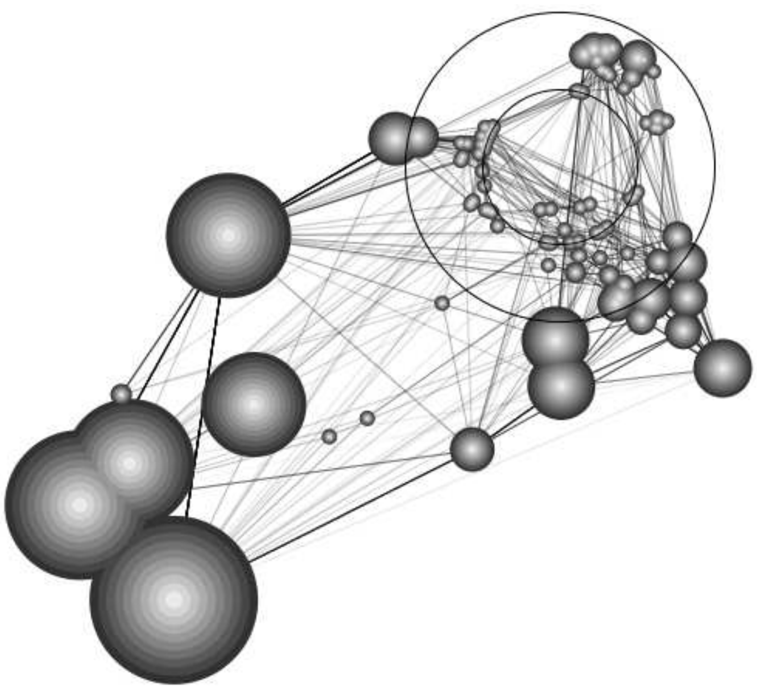


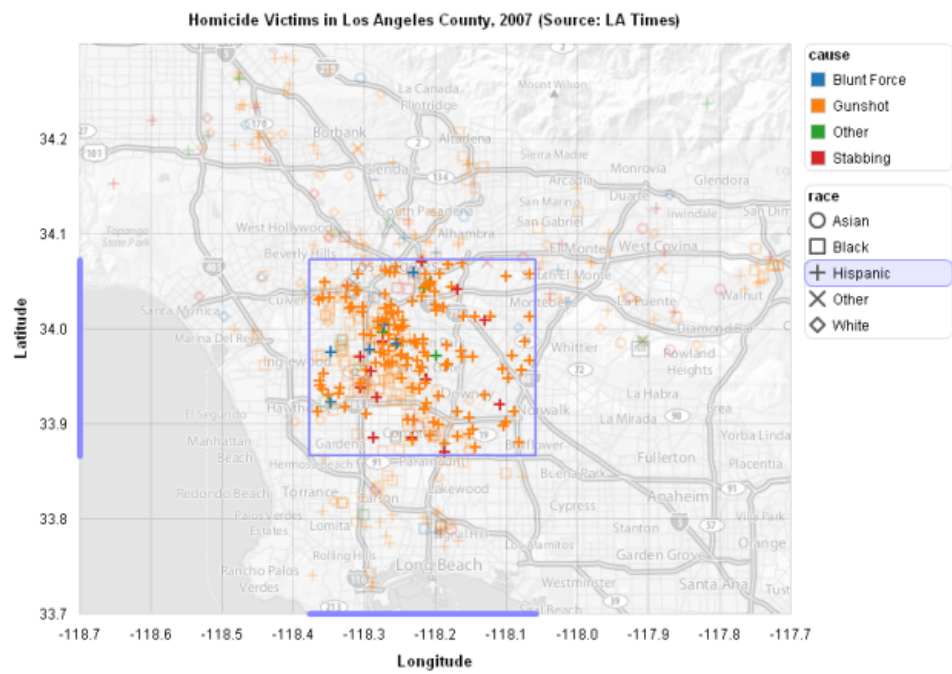
Image: /a/gort/homes/gort/treinish/dx_demos/v2/working/UI-ozone.net
File Execute Windows Connection Options Help
Visual Program Editor
File Edit Execute Windows Connection Options Help
Categories:
Debugging
Import and Export
Interactor
Macros
Miscellaneous
Realization
Rendering
Special
Splines
Structuring
Transformation
{ ALL }
Asteroid
AzimuthalEquidistant
Globe
HammerAitoff
Mercator
Mollweide
NpoleOrtho
PointProjections
Projections
ShowFiducials
Sphere
SpoleOrtho
WorldMap
WorldMapProjections
Import
Sequencer
Select
Colormap
Include
Color
RubberSheet
Sphere
Globe
Normals
Collect
Image
Colormap Editor
Execute Options Help
max
650.00000
Hue
Saturation
Value
Opacity
650
600
511.617
475.61
424.775
382.413
343.582
306.868
275.802
251.091
224.262
200.963
159.307
125.417
100
min max min
r g b r min max min max min max
100.00000
View Control...
Undo Ctrl+U
Redo Ctrl+D
Mode: Rotate
Set View: None
Projection: Perspective
View Angle: 30.000
Close
Reset Ctrl+F
Sequence Control
Z
X
Y

Prefuse & Flare

Operator-based toolkits for visualization design

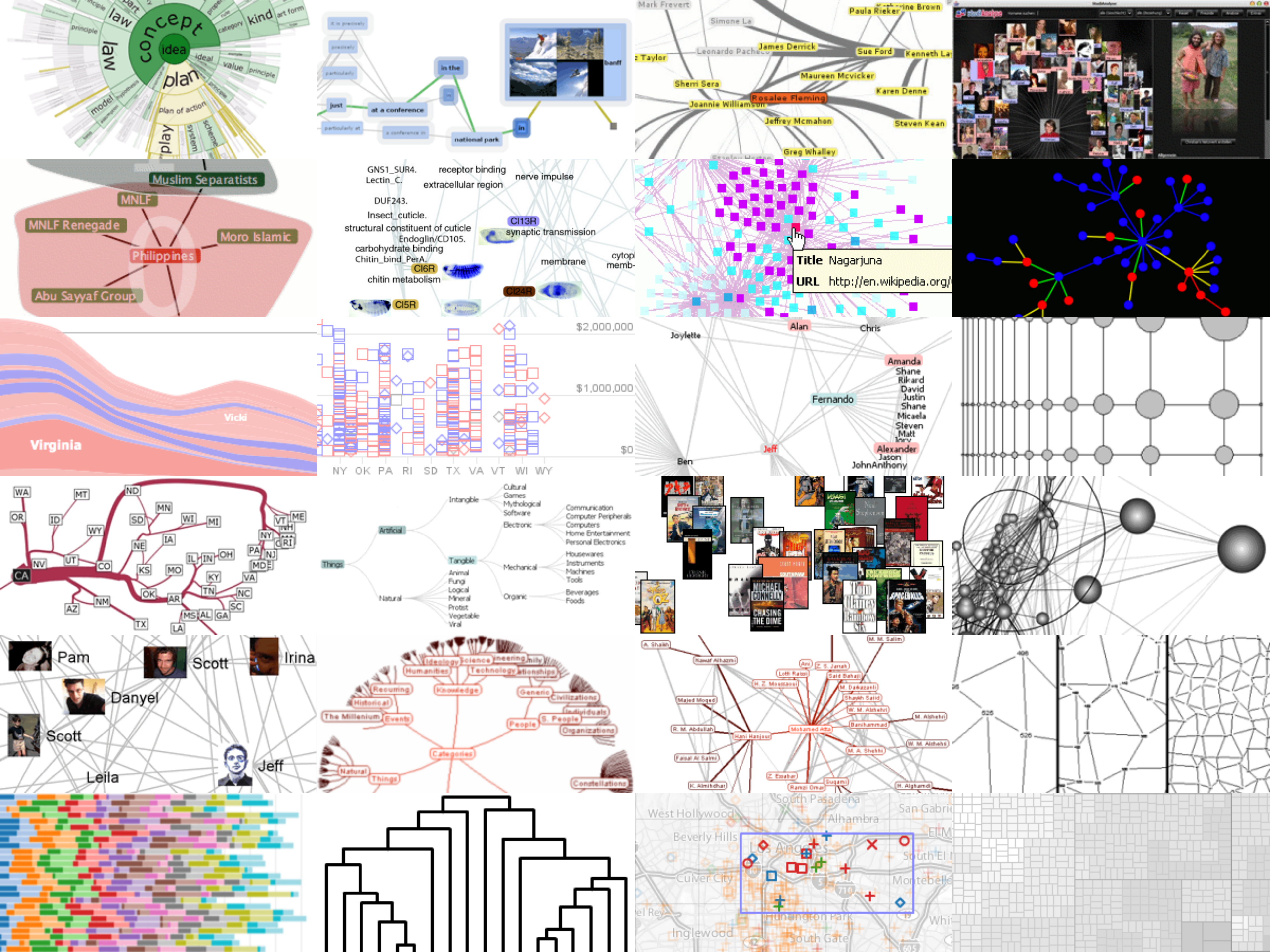
Vis = (Input Data -> Visual Objects) + Operators

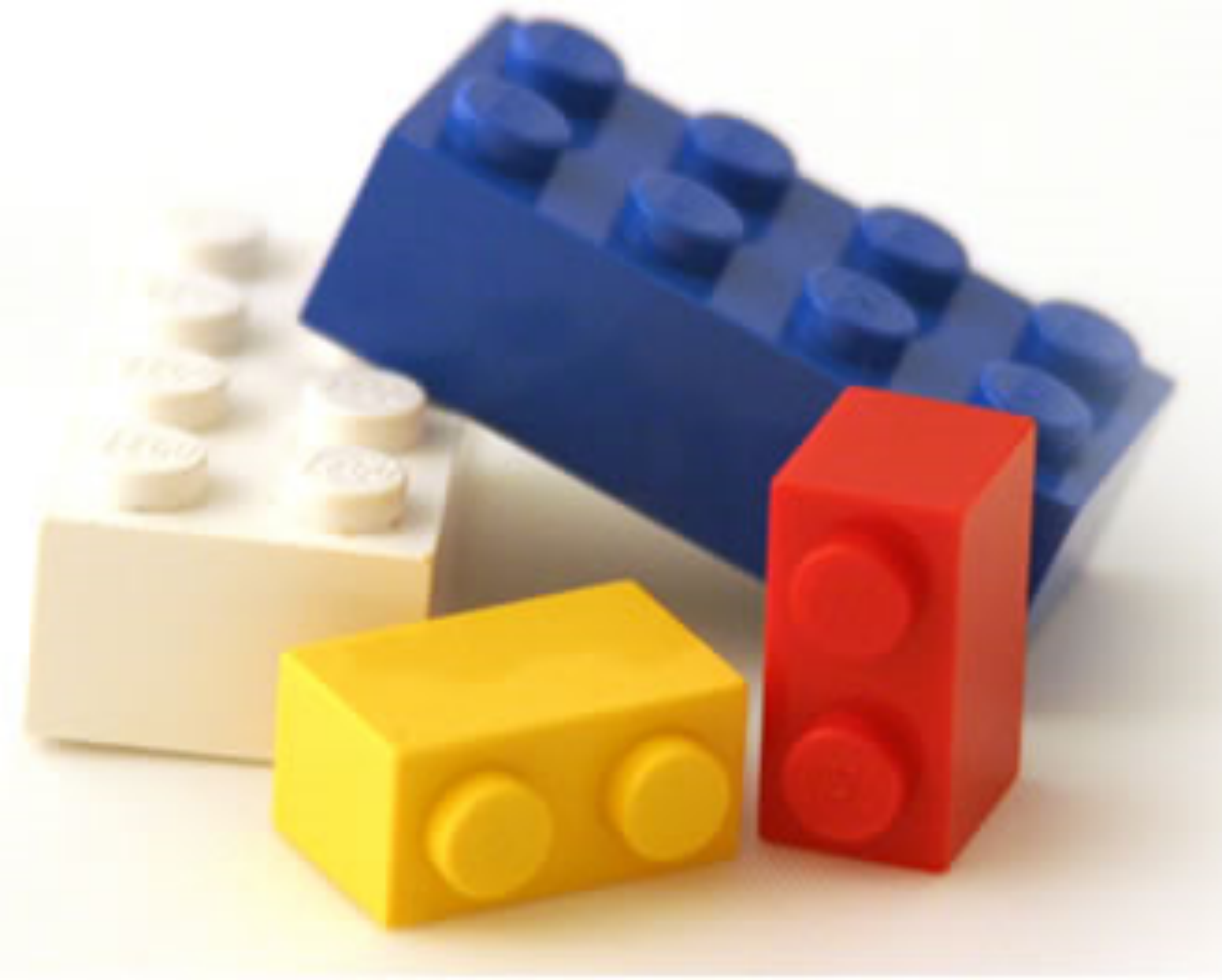


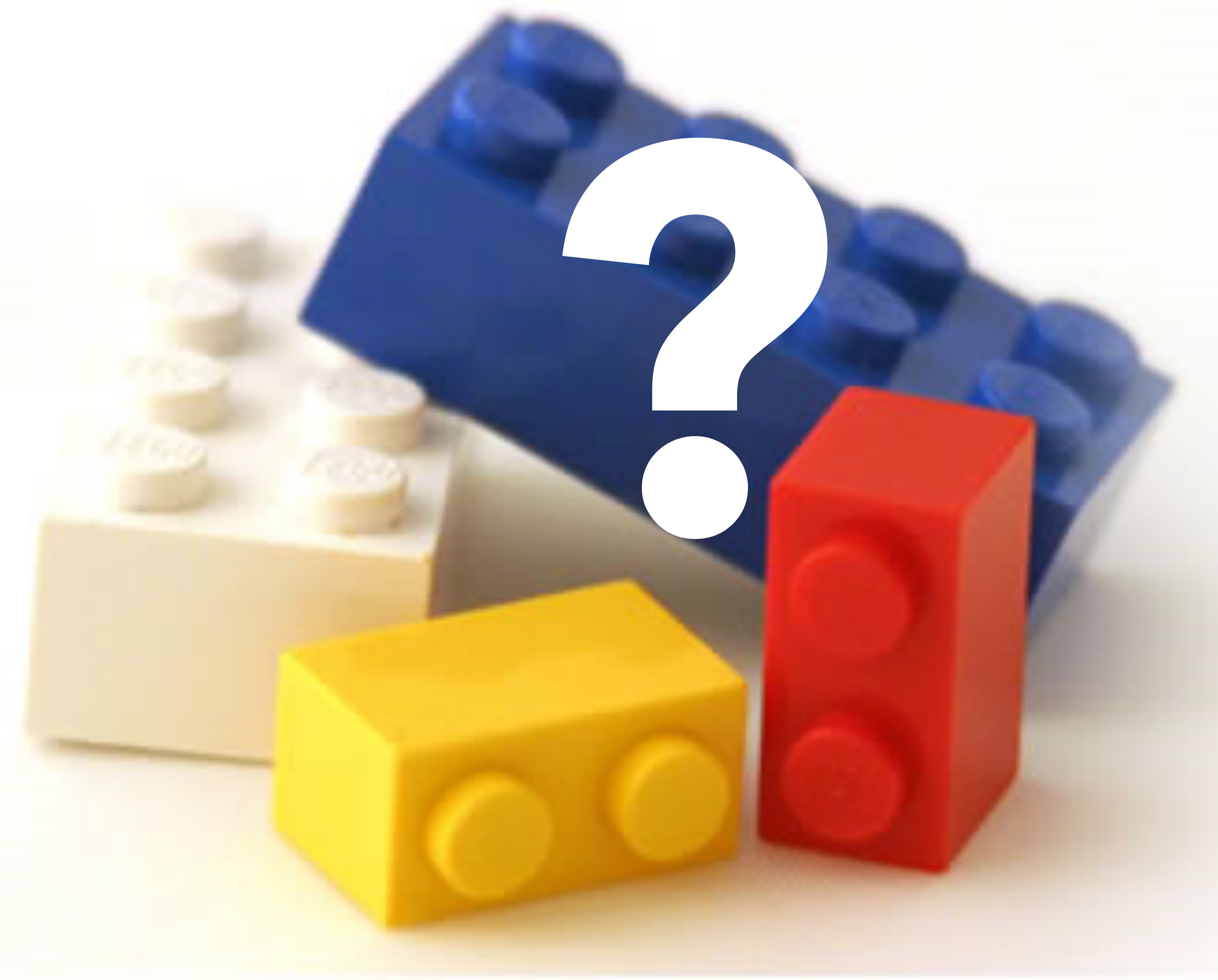


Prefuse (http://prefuse.org)

Flare (http://flare.prefuse.org)


concept
idea
law
kind
plan
plan of action
play
system
model
principle
value
ideal
in the
just
at a conference
national park
in
banff
Paula Rieker
Katherine Brown
Simone La
James Derrick
Leonardo Pacheco
Sue Ford
Kenneth Lay
Taylor
Sherri Sera
Maureen Mcvicker
Rosalee Fleming
Karen Denne
Joannie Williamson
Jeffrey Mcmahon
Steven Kean
Greg Whalley
Muslim Separatists
MNLF
MNLF Renegade
Moro Islamic
Philippines
Abu Sayyaf Group
GNS1_SUR4.
Lectin_C.
receptor binding
extracellular region
nerve impulse
DUF243.
Insect_cuticle.
structural constituent of cuticle
Endoglin/CD105.
carbohydrate binding
Chitin_bind_PerA.
chitin metabolism
CI13R
synaptic transmission
CI6R
membrane
CI24R
CI5R
Title Nagarjuna
URL http://en.wikipedia.org/
Vicki
Virginia
$2,000,000
$1,000,000
$0
NY OK PA RI SD TX VA VT WI WY
Joylette
Alan
Chris
Amanda
Shane
Rikard
David
Justin
Fernando
Micaela
Steven
Matt
Jeff
Alexander
Jason
Ben
JohnAnthony
WA
MT
ND
OR
ID
MN
SD
WI
MI
WY
VT
ME
NH
NY
RI
IA
NE
PA
NJ
NV
UT
IL
IN
OH
CA
CO
MD
KS
MO
KY
VA
OK
NC
AR
AZ
NM
TN
SC
TX
MS
AL
GA
LA
Things
Artificial
Intangible
Cultural
Games
Mythological
Software
Electronic
Communication
Computer Peripherals
Computers
Home Entertainment
Personal Electronics
Tangible
Mechanical
Housewares
Instruments
Machines
Tools
Natural
Animal
Fungi
Logical
Mineral
Protist
Vegetable
Viral
Organic
Beverages
Foods
Pam
Scott
Irina
Danyel
Scott
Leila
Jeff
Humanities
Technology
Knowledge
Recurring
Historical
The Millenium
Events
Generic
Civilizations
People
Organizations
Categories
Natural
Things
Constellations
Mohamed Atta
Hani Hanjour
South Pasadena
West Hollywood
Alhambra
San Gabriel
Beverly Hills
Los Angeles
Culver City
Montebello
Huntington Park
Inglewood
South Gate





Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Chart Typologies

Excel, Many Eyes, Google Charts

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Chart Typologies

Data Sets : State Quick Facts

Uploaded By: zinggoat

Created at: Friday May 18, 3:08 PM

Data Source: US Census Bureau

Description:

Tags: people census

view as text edit data set

	People QuickFacts	Population 2005 estimate	Population percent change April 1 2000 to July 1 2005	Population 2000	Population percent change 1990 to 2000	Persons under 5 years old percent 2004	Persons under 18 years old percent 2004	Persons 65 years old and over percent 2004
1	Alabama	4557808	0.03	4447100	0.1	0.07	0.24	0.13
2	Alaska	663661	0.06	626932	0.14	0.08	0.29	0.06
3	Arizona	5939292	0.16	5130632	0.4	0.08	0.27	0.13
4	Arkansas	2779154	0.04	2673400	0.14	0.07	0.25	0.14
5	California	36132147	0.07	33871648	0.14	0.07	0.27	0.11
6	Colorado	4665177	0.08	4301261	0.31	0.07	0.26	0.1
7	Connecticut	3510297	0.03	3405565	0.04	0.06	0.24	0.14
8	Delaware	843524	0.08	783600	0.18	0.07	0.23	0.13
9	Florida	17789864	0.11	15982378	0.24	0.06	0.23	0.17
10	Georgia	9072576	0.11	8186453	0.26	0.08	0.26	0.1
11	Hawaii	1275194	0.05	1211537	0.09	0.07	0.24	0.14
12	Idaho	1429096	0.1	1293953	0.29	0.07	0.27	0.11
13	Illinois	12763371	0.03	12419293	0.09	0.07	0.26	0.12

Choosing a visualization type for State Quick Facts

Analyze a text

Tag Cloud

How are you using your words? This enhanced tag cloud will show you the words popularity in the given set of text.

Learn more

Wordle

Wordle is a toy for generating "word clouds" from text that you provide. The clouds give greater prominence to words that appear more frequently in the source text.

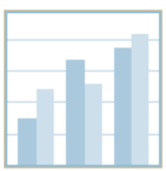
Learn more

Word Tree

See a branching view of how a word or phrase is used in a text. Navigate the text by zooming and clicking.

Learn more

Compare a set of values



Bar Chart

How do the items in your data set stack up? A bar chart is a simple and recognizable way to compare values. You can display several sets of bars for multivariate comparisons.

Learn more

Block Histogram

This versatile chart lets you get a quick sense of how a single set of data is distributed. Each item in the data is an individually identifiable block.

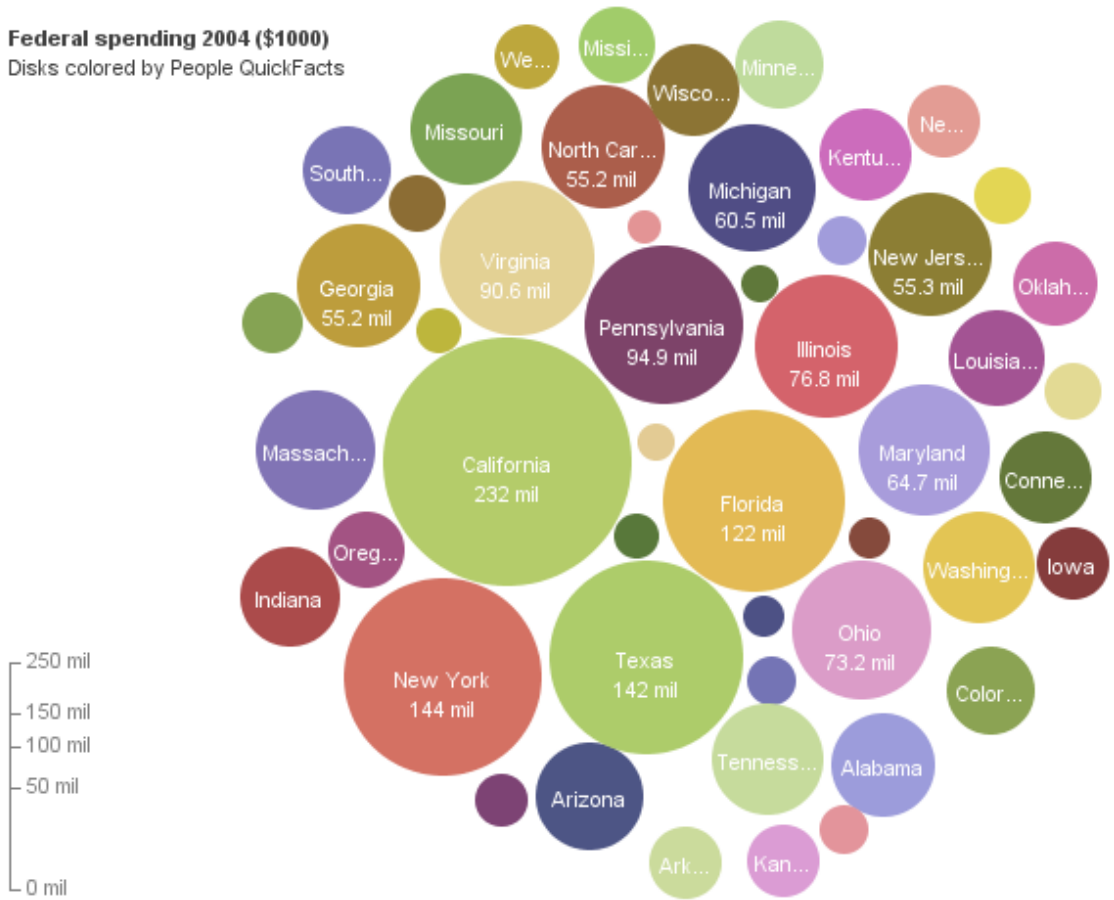
Learn more

Visualizations : Federal Spending by State, 2004

Creator: Anonymous
Tags: census people

People QuickFac...
Click to select,
Ctrl-Click: multiple
Shift-Click: range

- Alabama
- Alaska
- Arizona
- Arkansas
- California
- Colorado
- Connecticut
- Delaware
- Florida
- Georgia
- Hawaii
- Idaho
- Illinois
- Indiana
- Iowa
- Kansas
- Kentucky
- Louisiana
- Maine
- Maryland



Search>>

To highlight or find totals
click or ctrl-click.

Bubble Size: Federal spending 2004 ($1000) | Label: People QuickFacts | Color: People QuickFacts

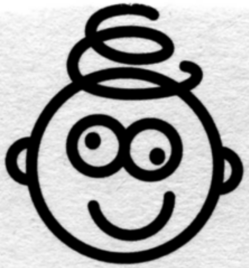
- Retail sales per capita 2002
- Minority-owned firms percent of total 1997
- Women-owned firms percent of total 1997
- Housing units authorized by building permits 2004
- Federal spending 2004 ($1000)
- Land area 2000 (square miles)
- Persons per square mile 2000
- FIPS Code

Data file ... Census Bureau

This data set
has not yet been rated

full image

rate this



MAD LIBS®

MY MUSIC LESSON

Every Wednesday, when I get home from school, I have a piano lesson. My teacher is a very strict house (NOUN). Her name is Hillary Clinton (CELEBRITY (FEMALE)). Our piano is a Steinway Concert tree (NOUN) and it has 88 cups (PLURAL NOUN). It also has a soft pedal and a/an Smily (ADJECTIVE) pedal. When I have a lesson, I sit down on the piano AlBERTO (NOUN) and play for 16 minuts (PERIOD OF TIME). I do scales to exercise my cats (PLURAL NOUN), and then I usually play a minuet by Johann Sebastian washington (CELEBRITY (LAST NAME)). Teacher says I am a natural Haunted House (NOUN) and have a good musical leg (PART OF THE BODY). Perhaps when I get better I will become a concert vet (PROFESSION) and give a recital at Carnegie hospital (TYPE OF BUILDING).

[M]ost charting packages channel user requests into a **rigid array of chart types**. To atone for this lack of flexibility, they offer a kit of post-creation editing tools to return the image to what the user originally envisioned. **They give the user an impression of having explored data rather than the experience**.

Leland Wilkinson
The Grammar of Graphics, 1999

Chart Typologies

Excel, Many Eyes, Google Charts

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

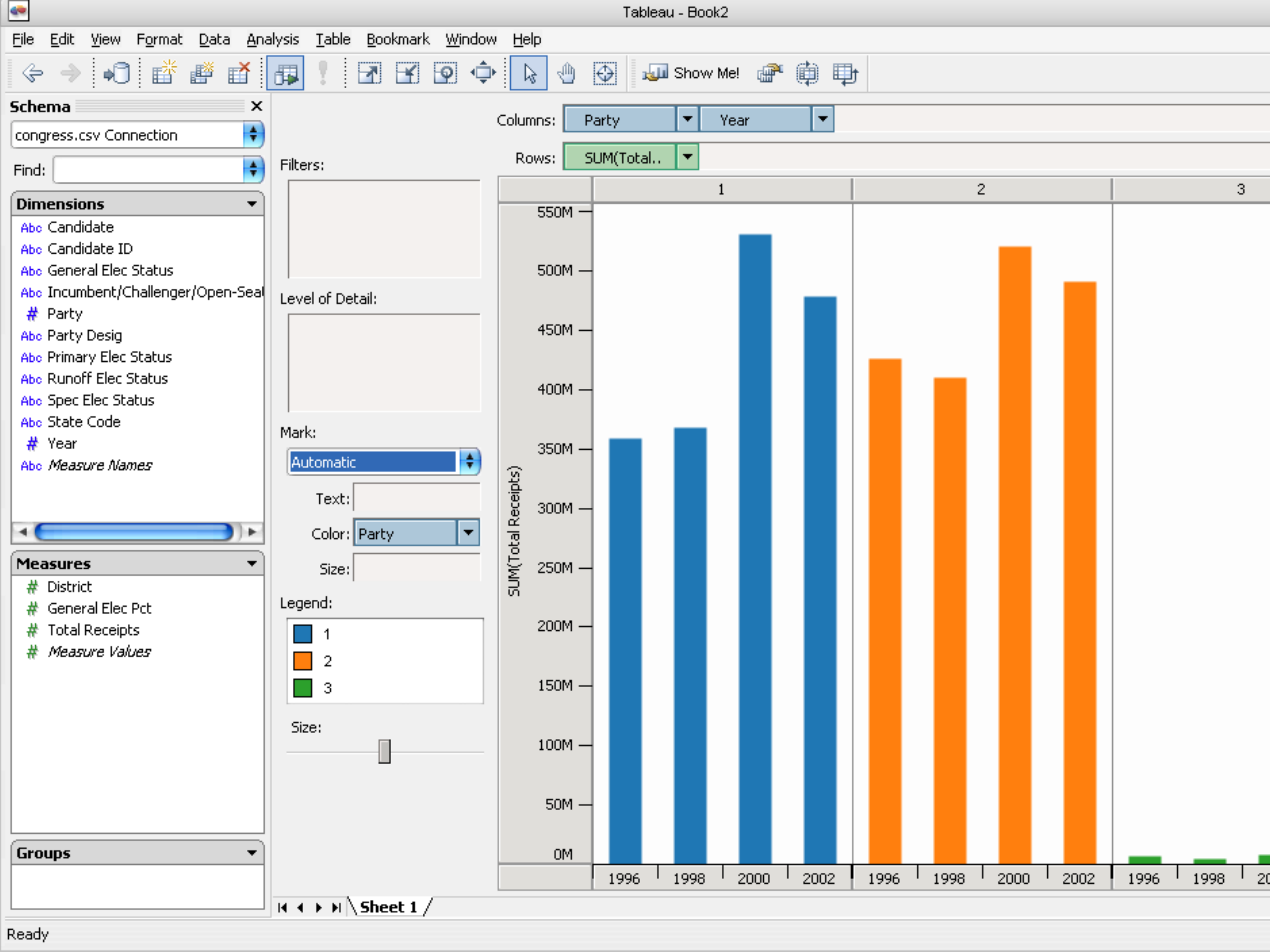
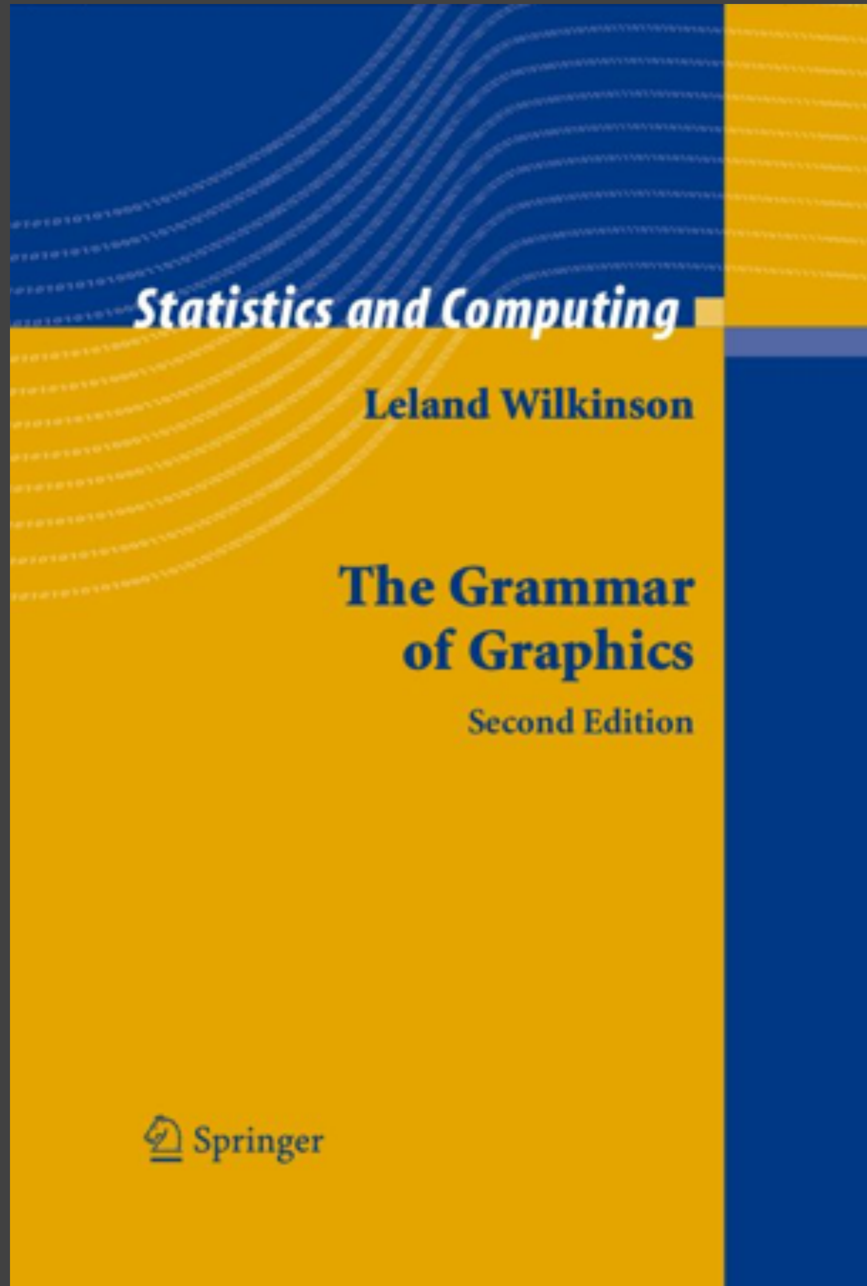
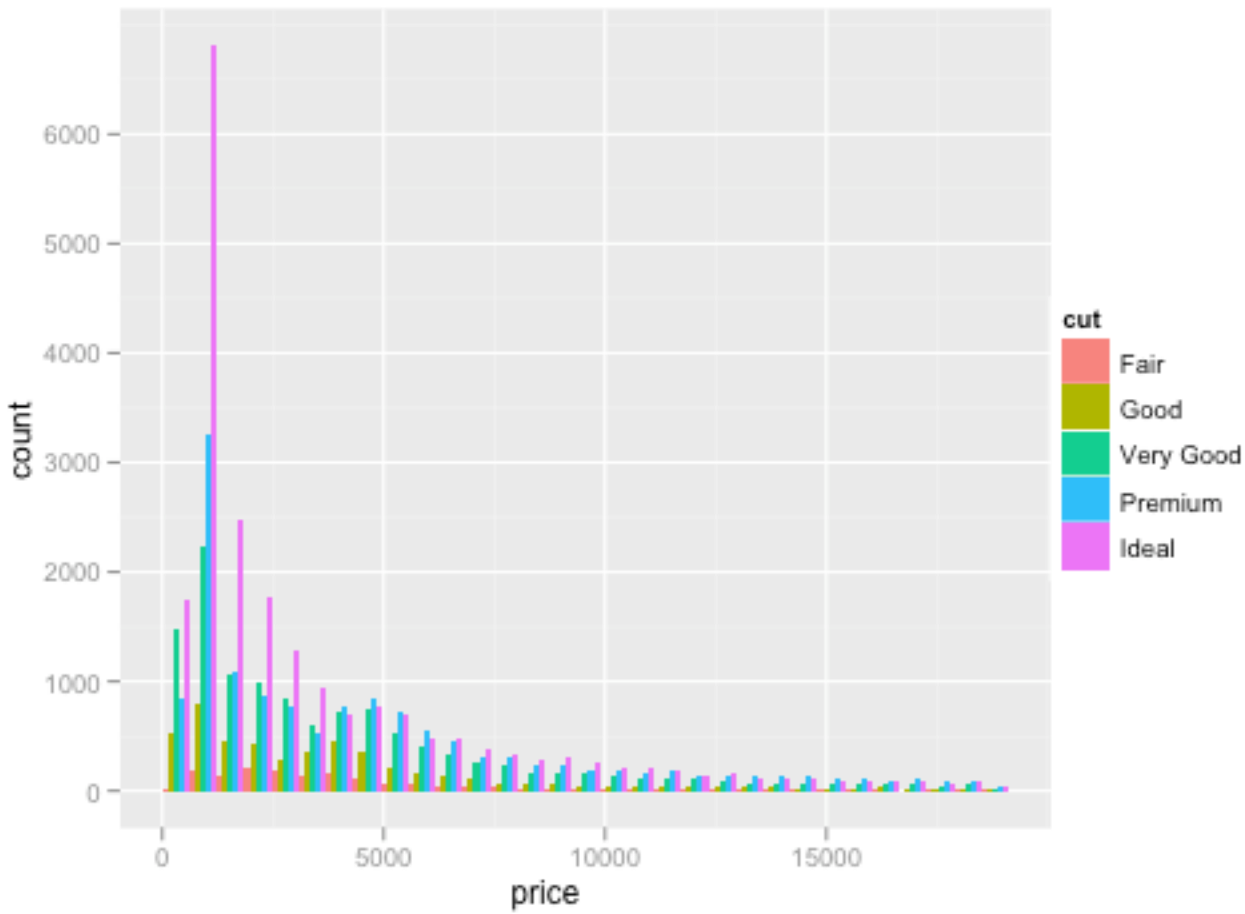


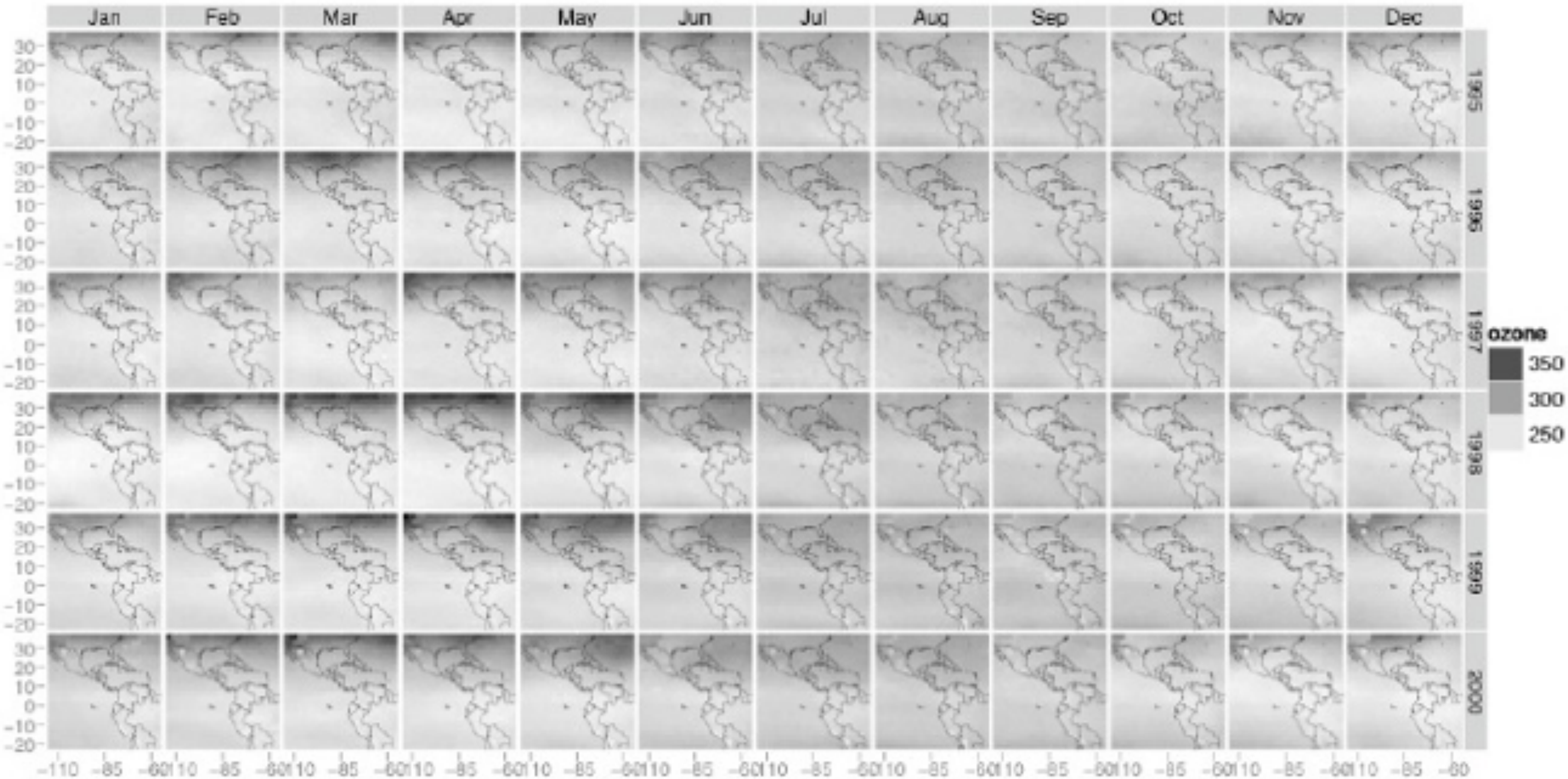
Tableau - Book2
File Edit View Format Data Analysis Table Bookmark Window Help
Show Me!
Schema
congress.csv Connection
Find:
Dimensions
Abc Candidate
Abc Candidate ID
Abc General Elec Status
Abc Incumbent/Challenger/Open-Seat
Party
Abc Party Desig
Abc Primary Elec Status
Abc Runoff Elec Status
Abc Spec Elec Status
Abc State Code
Year
Abc Measure Names
Measures
District
General Elec Pct
Total Receipts
Measure Values
Groups
Filters:
Level of Detail:
Mark:
Automatic
Text:
Color: Party
Size:
Legend:
1
2
3
Size:
Columns: Party Year
Rows: SUM(Total..
1
2
3
550M
500M
450M
400M
350M
300M
250M
200M
150M
100M
50M
0M
SUM(Total Receipts)
1996
1998
2000
2002
1996
1998
2000
2002
1996
1998
Sheet 1
Ready


Statistics and Computing
Leland Wilkinson
The Grammar of Graphics
Second Edition
Springer

```
ggplot(diamonds, aes(x=price, fill=cut))
+ geom_bar(position="dodge")
```




```
ggplot(diamonds, aes(x=price, fill=cut))
+ geom_bar(position="dodge")
```




```
qplot(long, lat, data = expo, geom = "tile", fill = ozone,
  facets = year ~ month) +
scale_fill_gradient(low = "white", high = "black") + map
```

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Ease-of-Use

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Ease-of-Use

Expressiveness

Ease-of-Use

Expressiveness

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

?

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Ease-of-Use

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Visualization Grammars

Protovis, D3.js

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

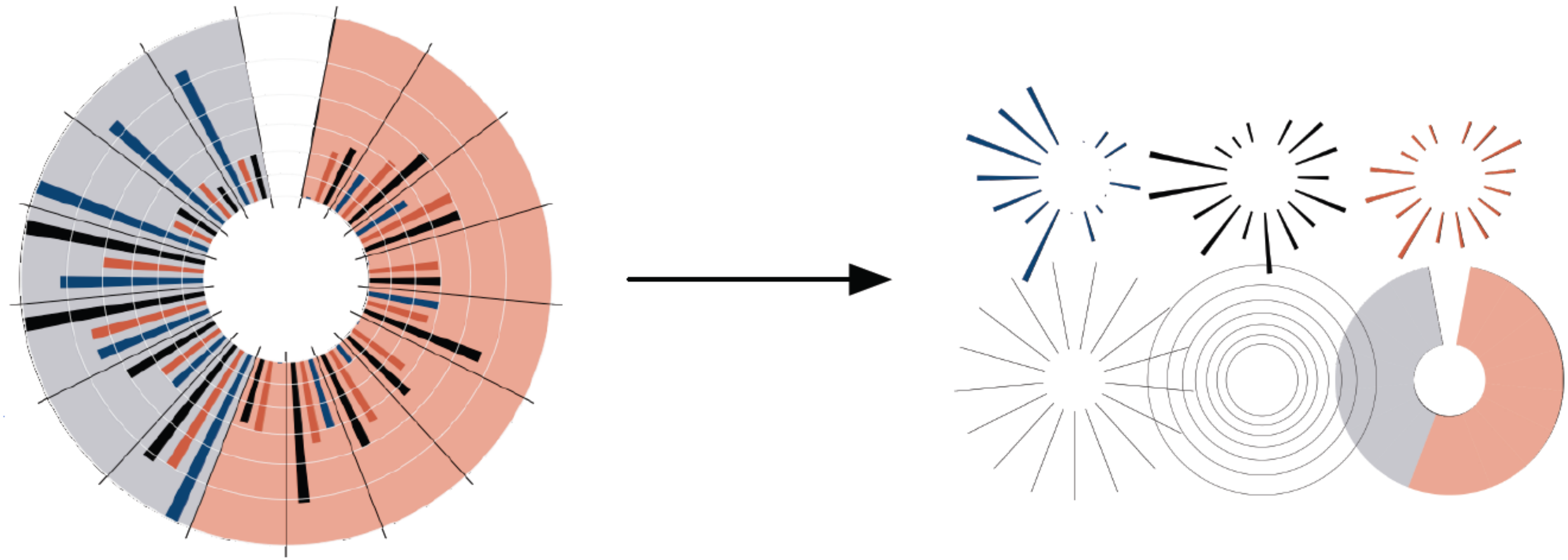
Expressiveness

Protovis & D3

Today's first task is not to invent wholly new [*graphical*] techniques, though these are needed. Rather we need most vitally to recognize and reorganize the **essential of old techniques**, to **make easy their assembly in new ways**, and to **modify their external appearances to fit the new opportunities**.

J. W. Tukey, M. B. Wilk
Data Analysis & Statistics, 1965

Protovis: A Grammar for Visualization



A graphic is a composition of data-representative marks.

with **Mike Bostock**, **Vadim Ogievetsky, Jeff Heer**

Visualization Grammar

Visualization Grammar

Data	Input data to visualize

Visualization Grammar

Data	Input data to visualize
Transforms	Grouping, stats, projection, layout

Visualization Grammar

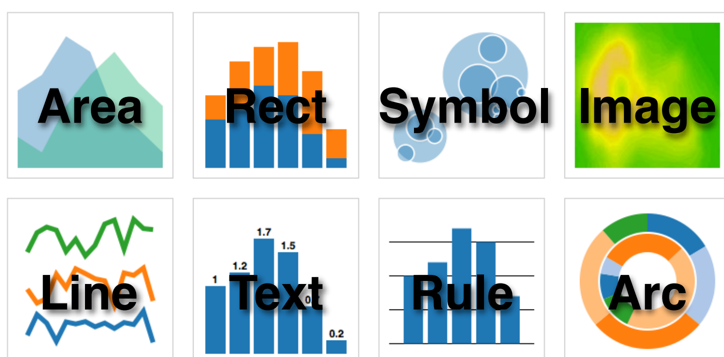
Data	Input data to visualize
Transforms	Grouping, stats, projection, layout
Scales	Map data values to visual values

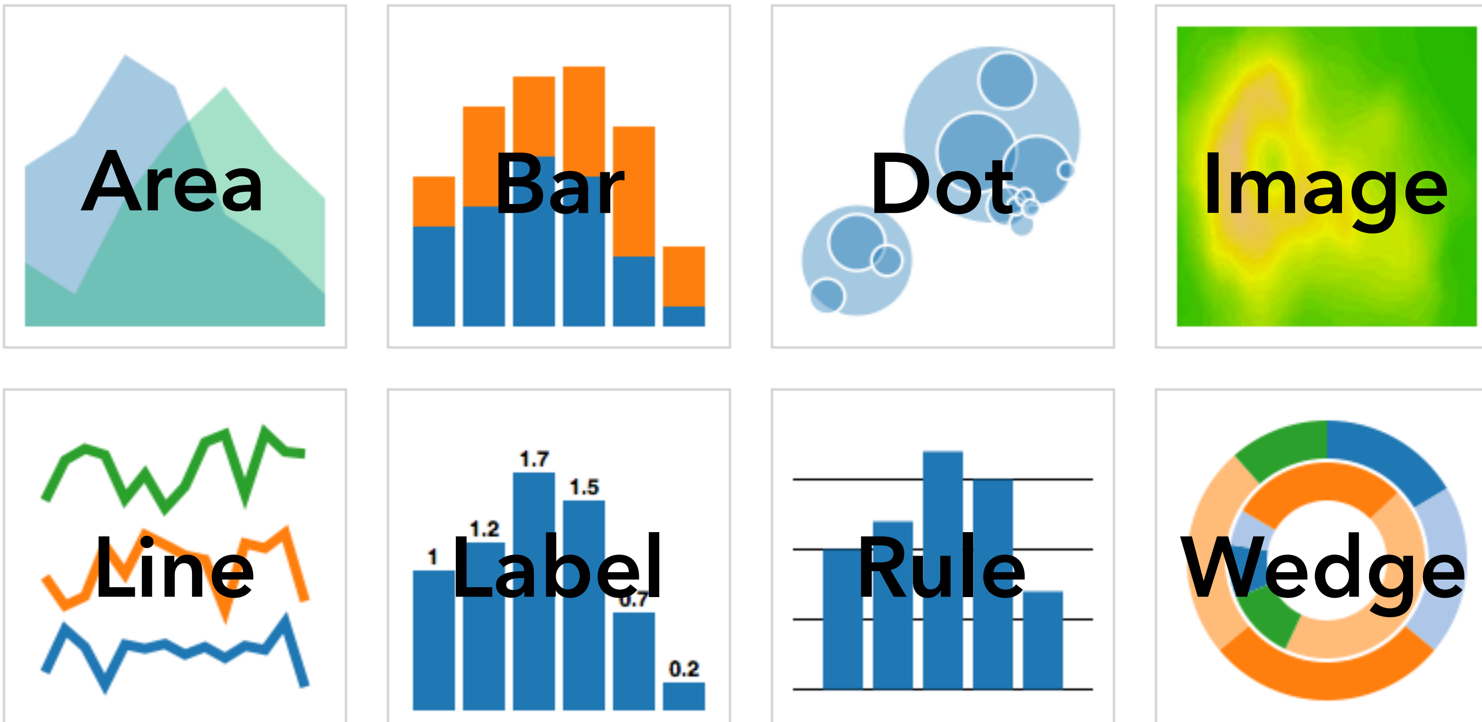
Visualization Grammar

Data	Input data to visualize
Transforms	Grouping, stats, projection, layout
Scales	Map data values to visual values
Guides	Axes & legends visualize scales

Visualization Grammar

Data	Input data to visualize
Transforms	Grouping, stats, projection, layout
Scales	Map data values to visual values
Guides	Axes & legends visualize scales
Marks	Data-representative graphics





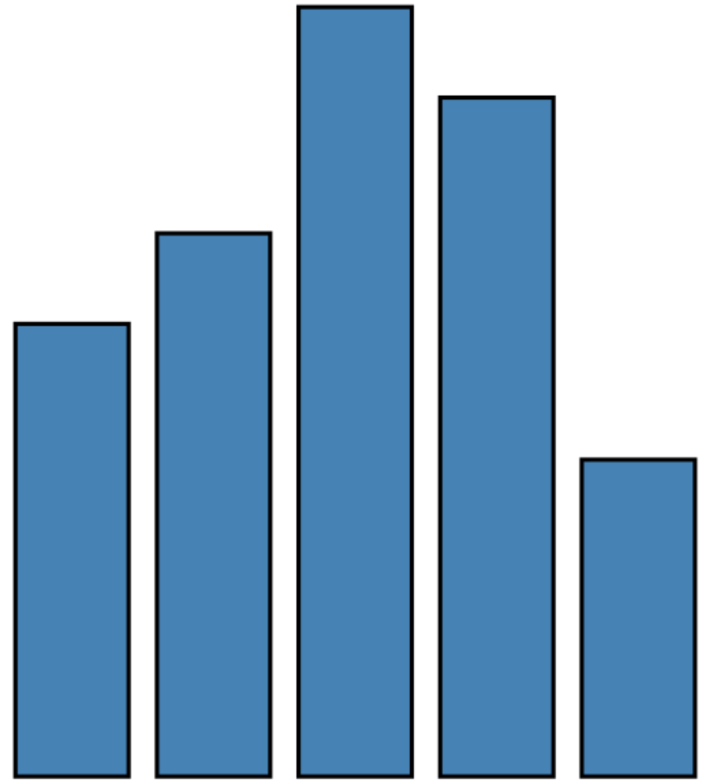
MARKS: Protovis graphical primitives

MARK $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	λ
visible	λ
left	λ
bottom	λ
width	λ
height	λ
fillStyle	λ
strokeStyle	λ
lineWidth	λ
...	λ

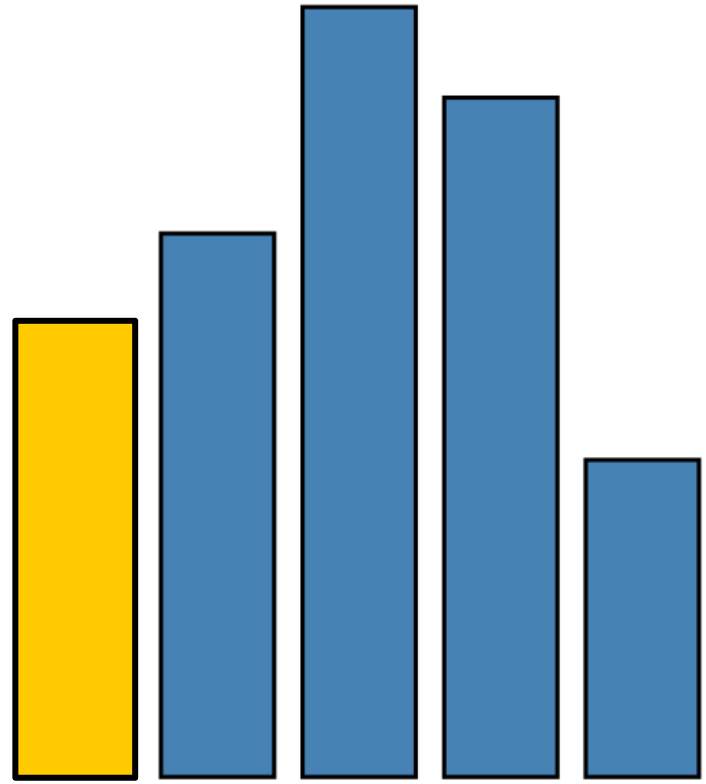
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	λ: index * 25				
bottom	0				
width	20				
height	λ: datum * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



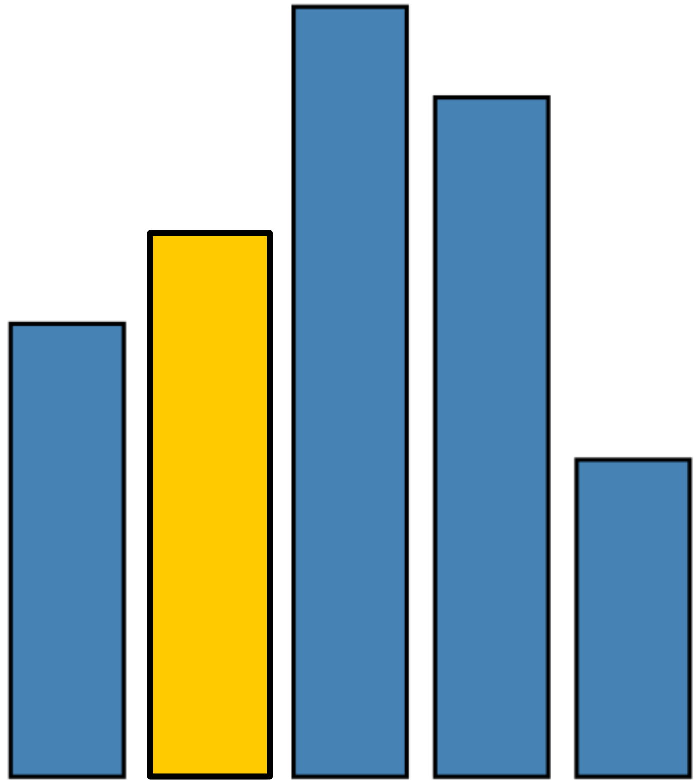
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	0 * 25				
bottom	0				
width	20				
height	1 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



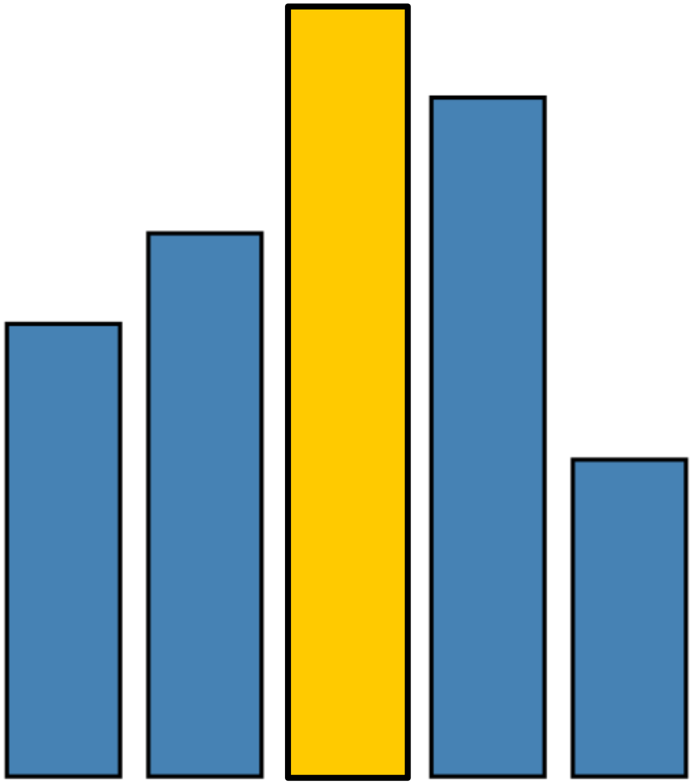
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	1 * 25				
bottom	0				
width	20				
height	1.2 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



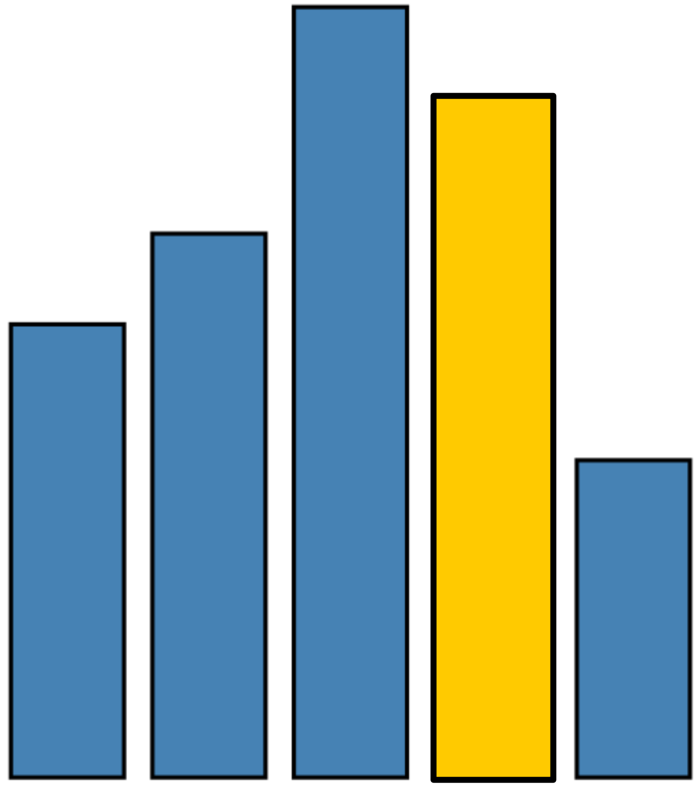
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	2 * 25				
bottom	0				
width	20				
height	1.7 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



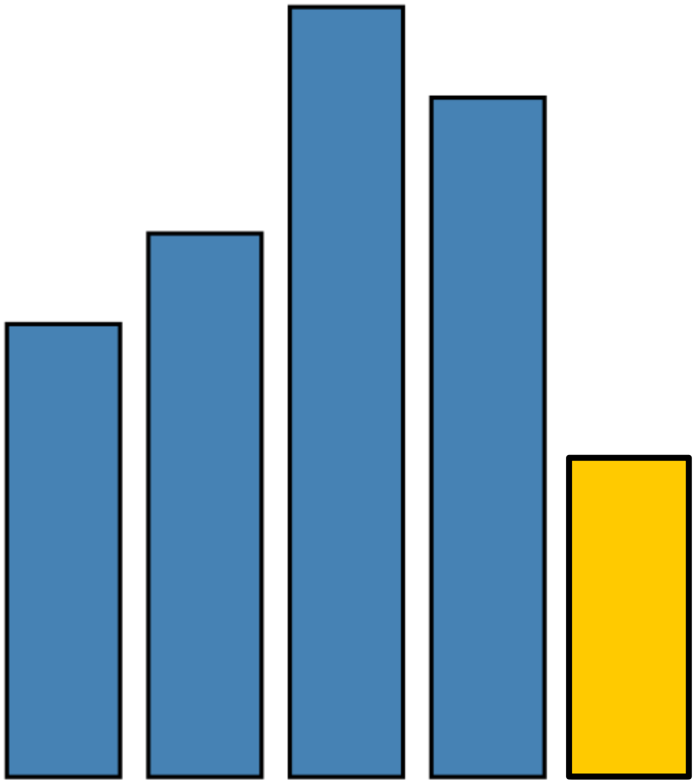
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	3 * 25				
bottom	0				
width	20				
height	1.5 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



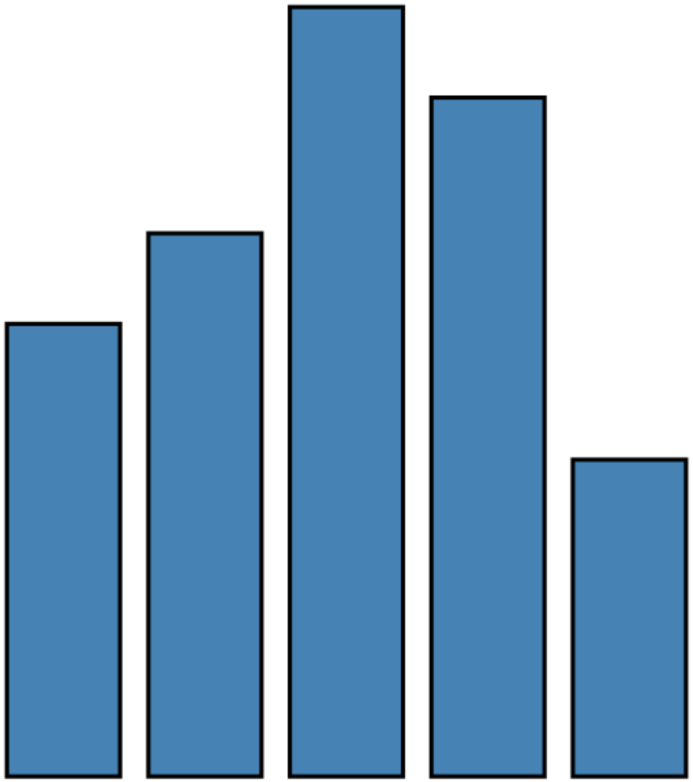
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	4 * 25				
bottom	0				
width	20				
height	0.7 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				

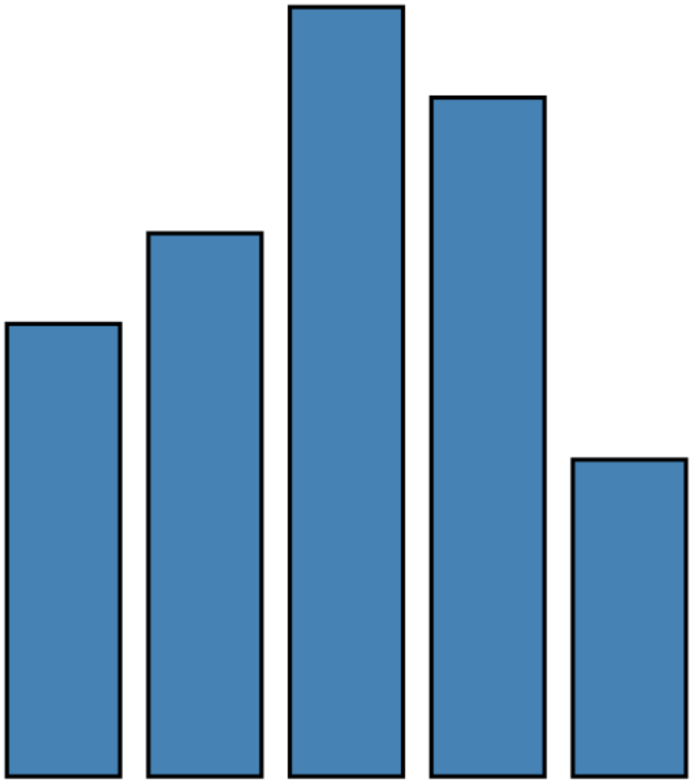


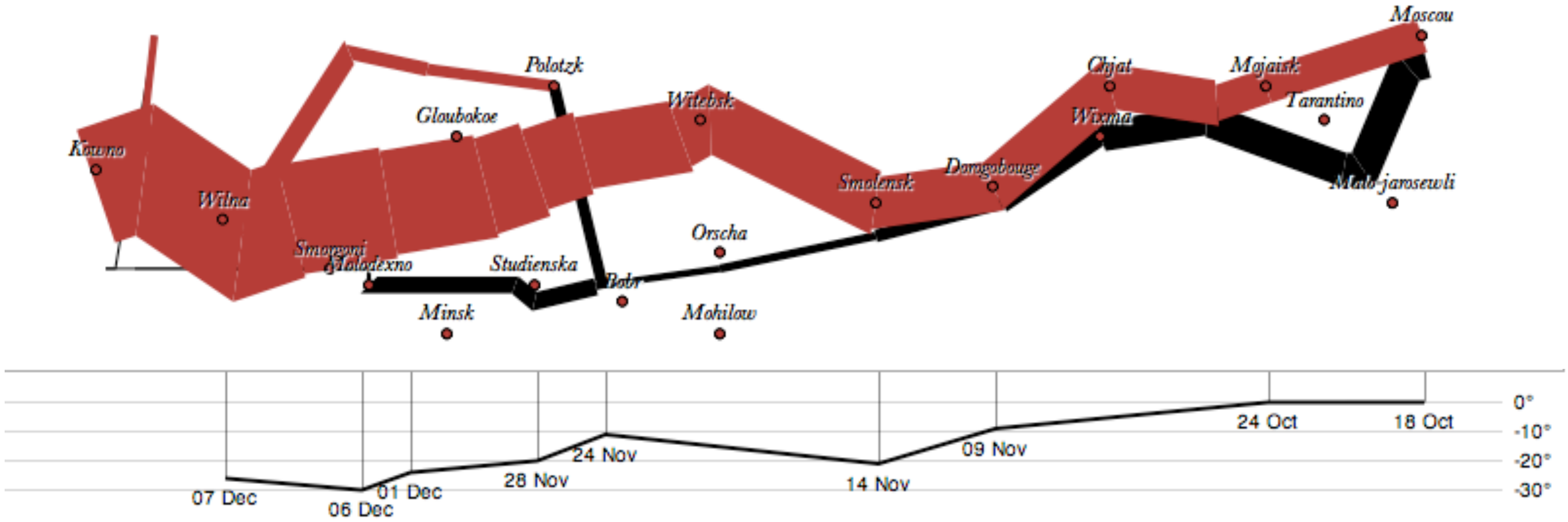
RECT $\lambda : \boldsymbol{D} \rightarrow \boldsymbol{R}$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	λ: index * 25				
bottom	0				
width	20				
height	λ: datum * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



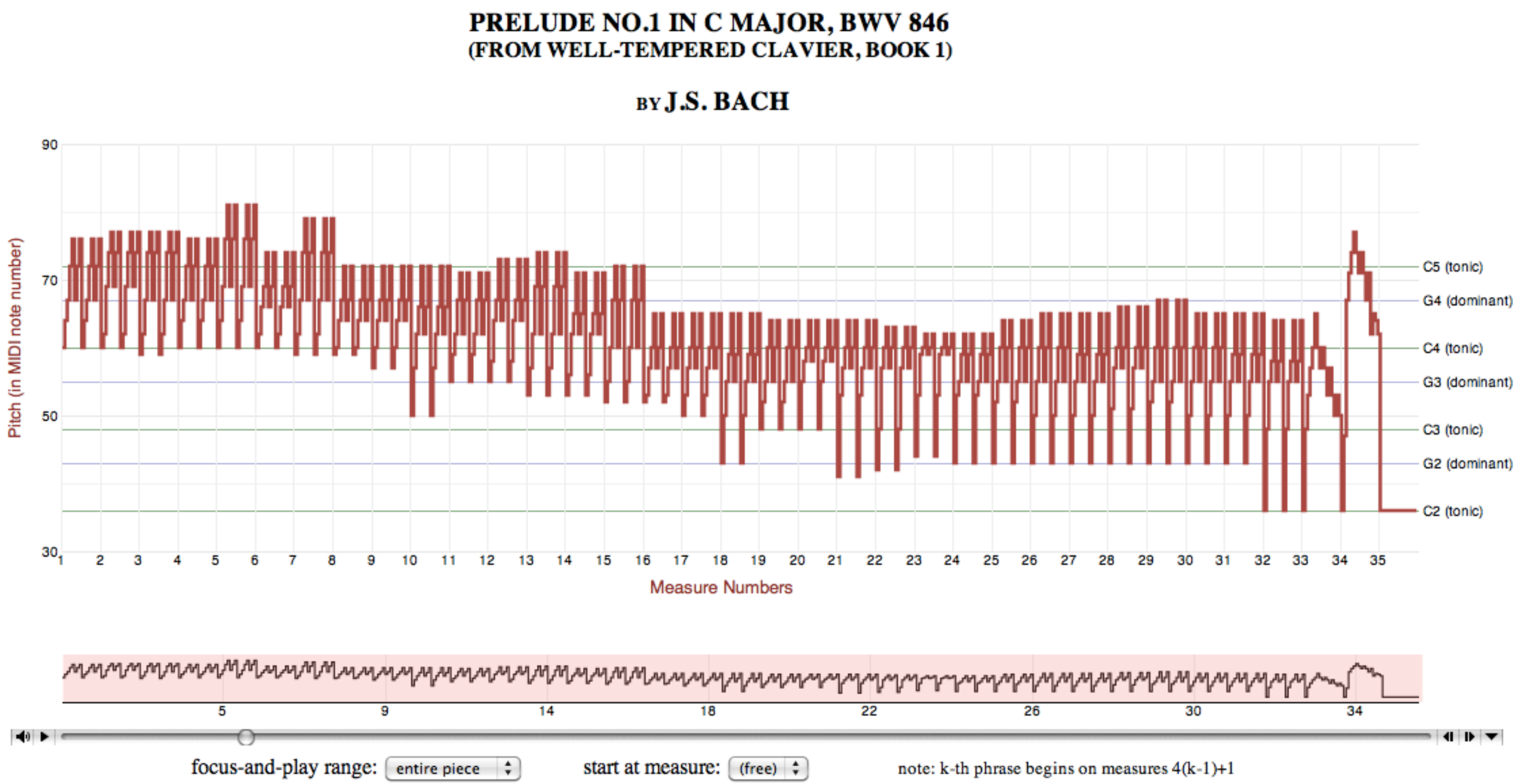
```
var vis = new pv.Panel();
vis.add(pv.Bar)
  .data([1, 1.2, 1.7, 1.5, 0.7])
  .visible(true)
  .left((d) => this.index * 25);
  .bottom(0)
  .width(20)
  .height((d) => d * 80)
  .fillStyle("blue")
  .strokeStyle("black")
  .lineWidth(1.5);
vis.render();
```



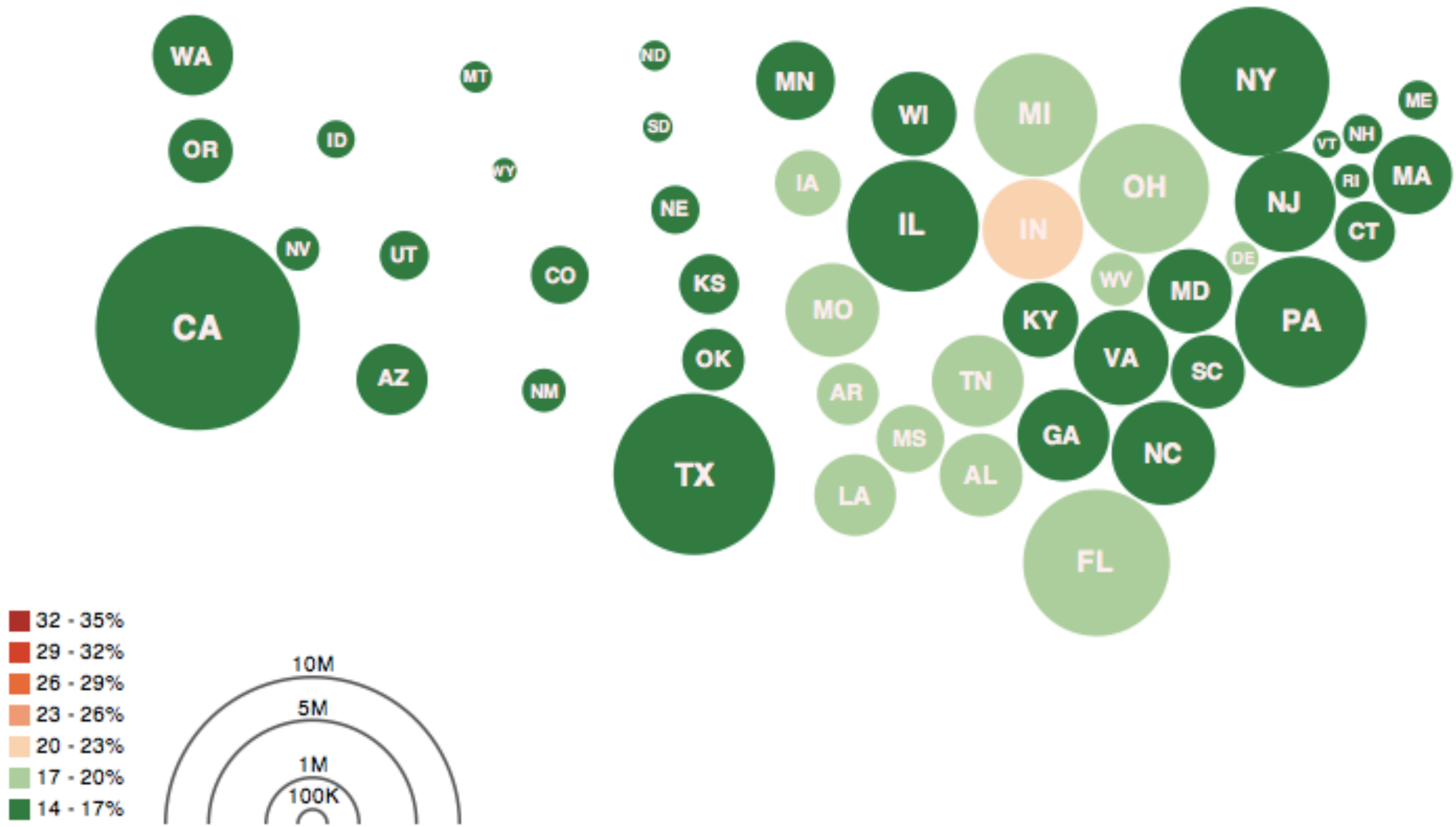



```
var army = pv.nest(napoleon.army, "dir", "group“);
var vis = new pv.Panel();

var lines = vis.add(pv.Panel).data(army);
lines.add(pv.Line)
  .data(() => army[this.idx])
  .left(lon).top(lat).size((d) => d.size/8000)
  .strokeStyle(() => color[army[paneIndex][0].dir]);

vis.add(pv.Label).data(napoleon.cities)
  .left(lon).top(lat)
  .text((d) => d.city).font("italic 10px Georgia")
  .textAlign("center").textBaseline("middle");
```

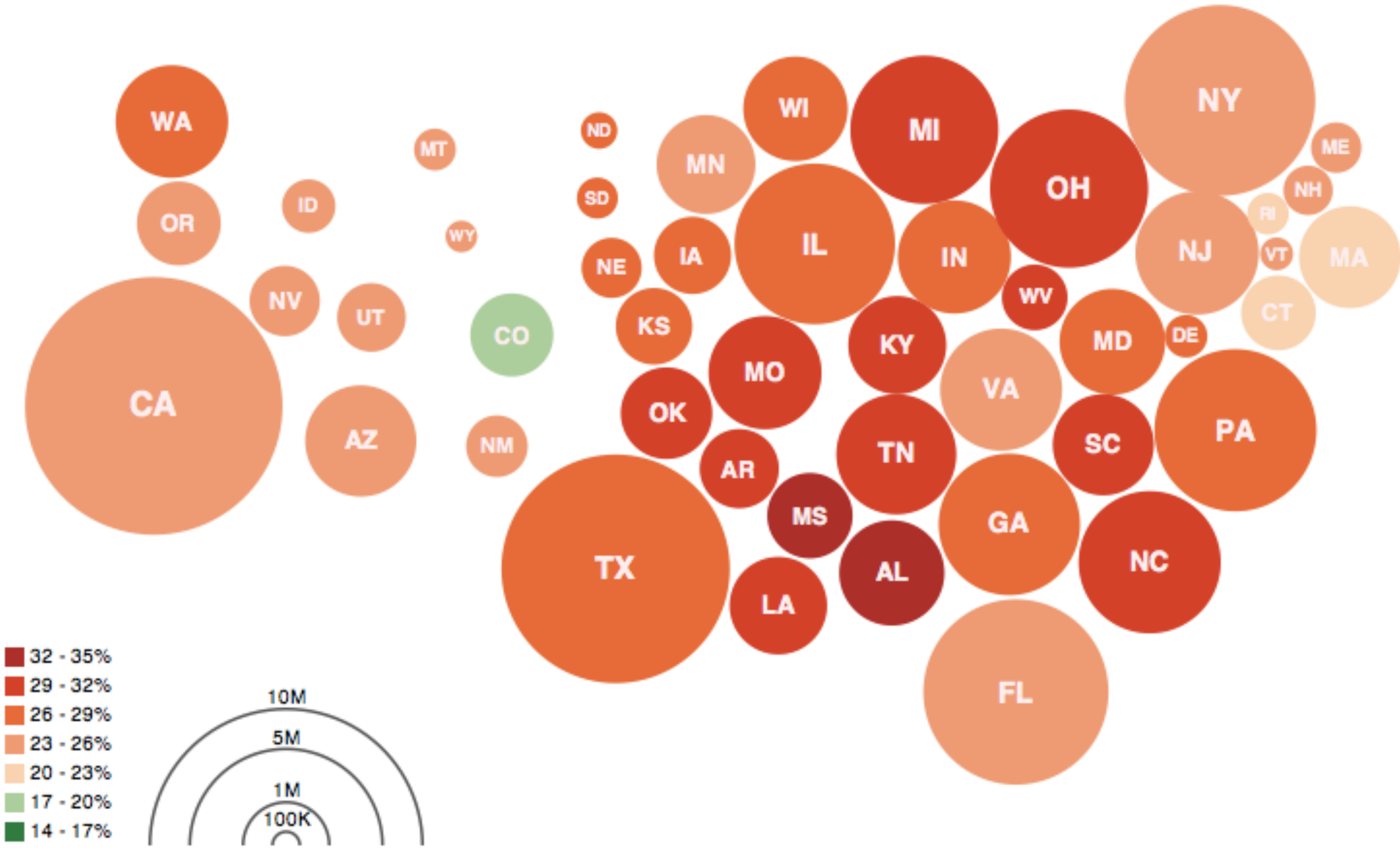
```
vis.add(pv.Rule).data([0,-10,-20,-30])
  .top((d) => 300 - 2*d - 0.5).left(200).right(150)
  .lineWidth(1).strokeStyle("#ccc")
  .anchor("right").add(pv.Label)
   .font("italic 10px Georgia")
   .text((d) => d+"°").textBaseline("center");

vis.add(pv.Line).data(napoleon.temp)
  .left(lon).top(tmp) .strokeStyle("#0")
 .add(pv.Label)
  .top((d) => 5 + tmp(d))
  .text((d) => d.temp+"° "+d.date.substr(0,6))
  .textBaseline("top").font("italic 10px Georgia");
```



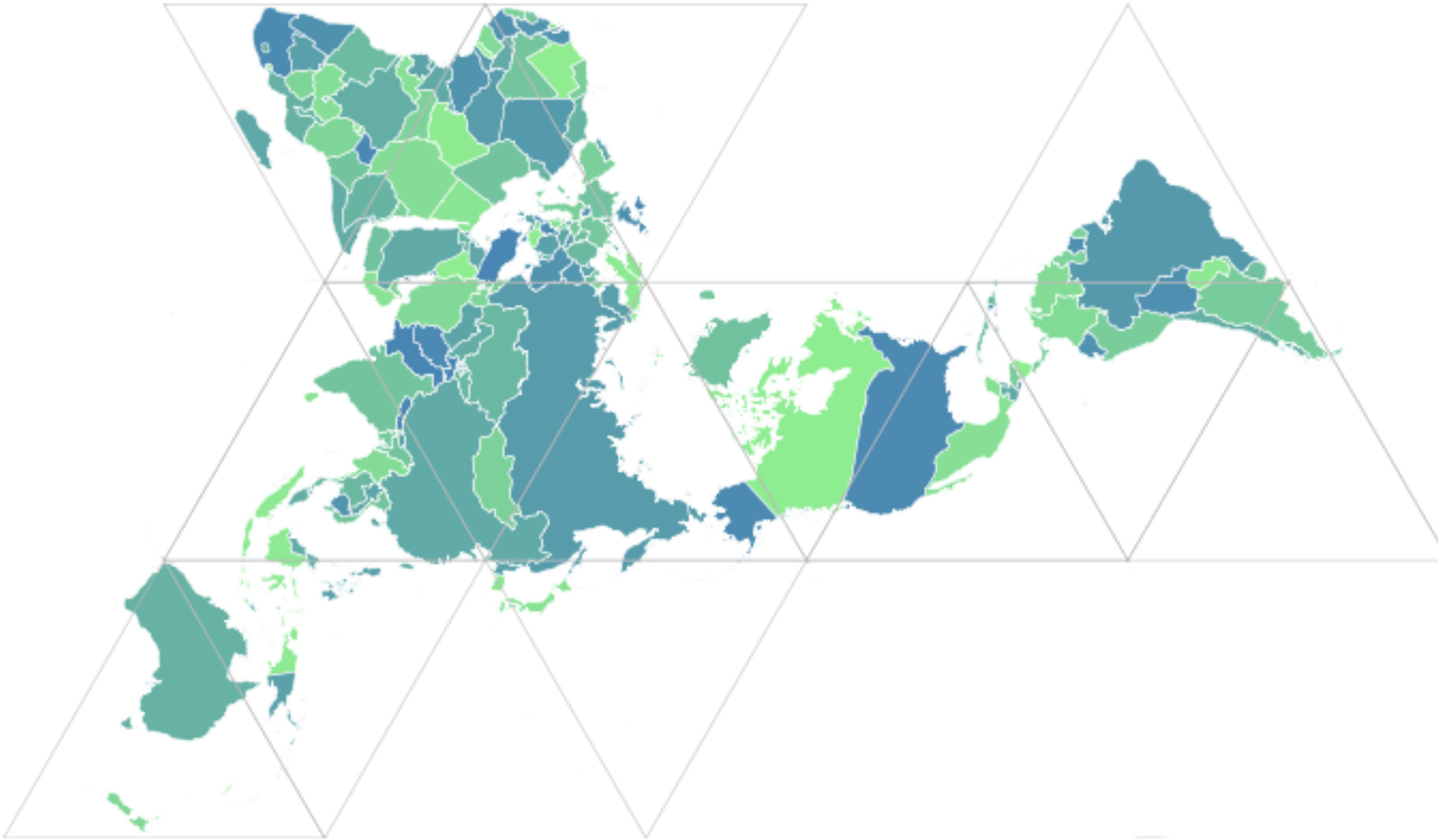
Bach's Prelude #1 in C Major | Jieun Oh



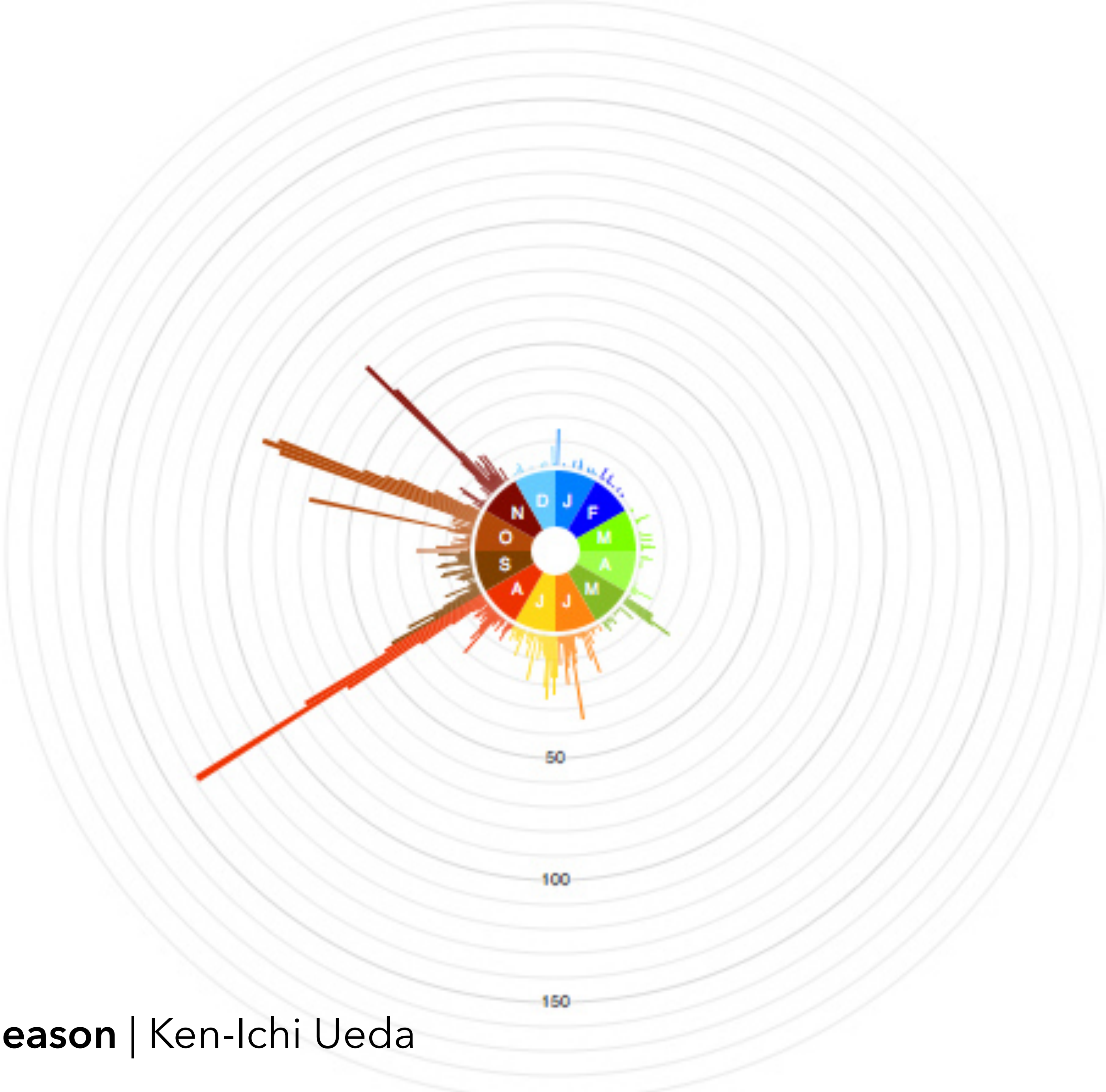
Obesity Map | Vadim Ogievetsky



Obesity Map | Vadim Ogievetsky

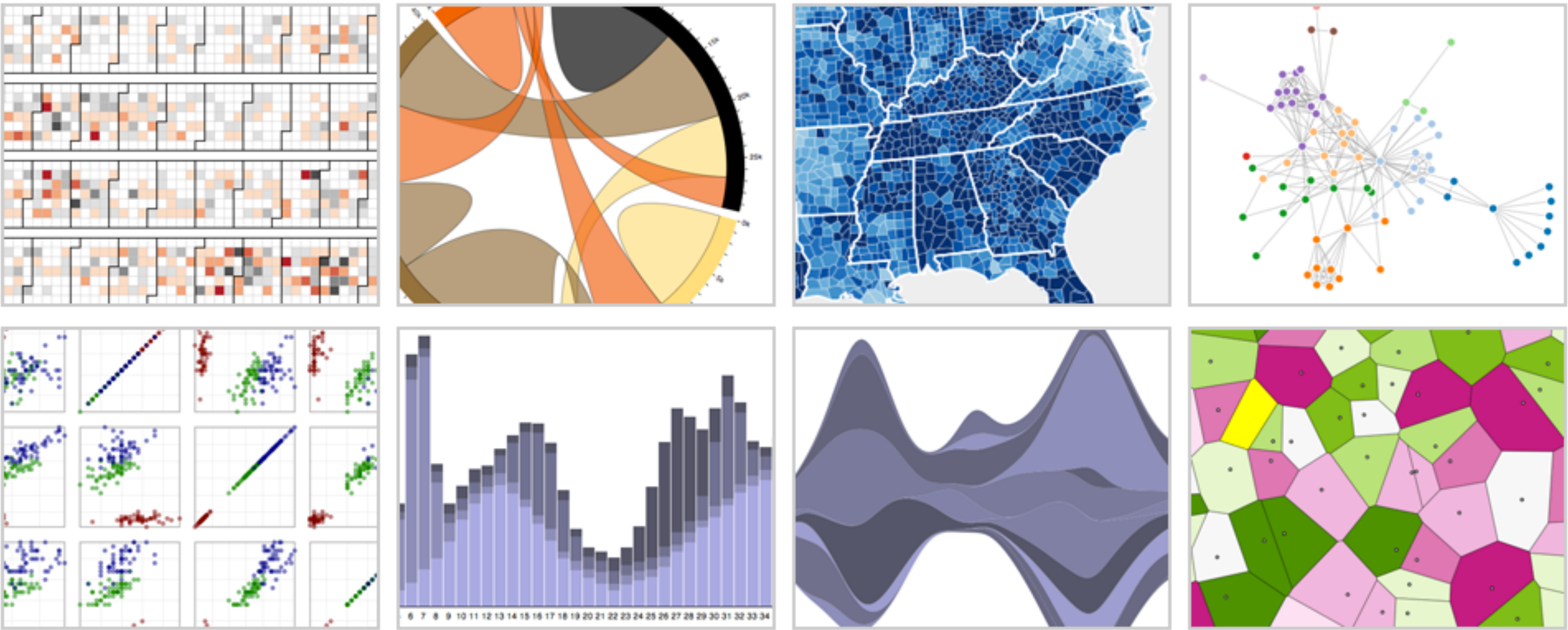


Dymaxion Maps | Vadim Ogievetsky



FlickrSeason | Ken-Ichi Ueda

d3.js Data-Driven Documents



with **Mike Bostock**, Jason Davies, Vadim Ogievetsky, Jeff Here

Protovis

Specialized mark types

+ Streamlined design
- Limits expressiveness
- More overhead (slower)
- Harder to debug
- Self-contained model

Specify a scene (nouns)

+ Quick for static vis
- Delayed evaluation
- Animation, interaction are more cumbersome

Protovis

Specialized mark types

+ Streamlined design
- Limits expressiveness
- More overhead (slower)
- Harder to debug
- Self-contained model

Specify a scene (nouns)

+ Quick for static vis
- Delayed evaluation
- Animation, interaction are more cumbersome

D3

Bind data to DOM

- Exposes SVG/CSS/…
+ Exposes SVG/CSS/…
+ Less overhead (faster)
+ Debug in browser
+ Use with other tools

Transform a scene (verbs)

- More complex model
+ Immediate evaluation
+ Dynamic data, anim, and interaction natural

D3 Selections

The core abstraction in D3 is a ***selection***.

D3 Selections

The core abstraction in D3 is a ***selection***.

```
// Add and configure an SVG element (<svg width="500" height="300">)
var svg = d3.append("svg")      // add new SVG to page body
    .attr("width", 500)         // set SVG width to 500px
    .attr("height", 300);       // set SVG height to 300px
```

D3 Selections

The core abstraction in D3 is a ***selection***.

```
// Add and configure an SVG element (<svg width="500" height="300">)
var svg = d3.append("svg")          // add new SVG to page body
    .attr("width", 500)             // set SVG width to 500px
    .attr("height", 300);           // set SVG height to 300px

// Select & update existing rectangles contained in the SVG element
svg.selectAll("rect")               // select all SVG rectangles
    .attr("width", 100)             // set rect widths to 100px
    .style("fill", "steelblue");    // set rect fill colors
```

Data Binding

Selections can *bind* **data and DOM elements**.

var values = [{…}, {…}, {…}, …]; // input data as JS objects

Data Binding

Selections can ***bind* data and DOM elements**.

```
var values = [ {…}, {…}, {…}, … ]; // input data as JS objects

// Select SVG rectangles and bind them to data values.
var bars = svg.selectAll("rect.bars").data(values);
```

Data Binding

Selections can ***bind* data and DOM elements**.

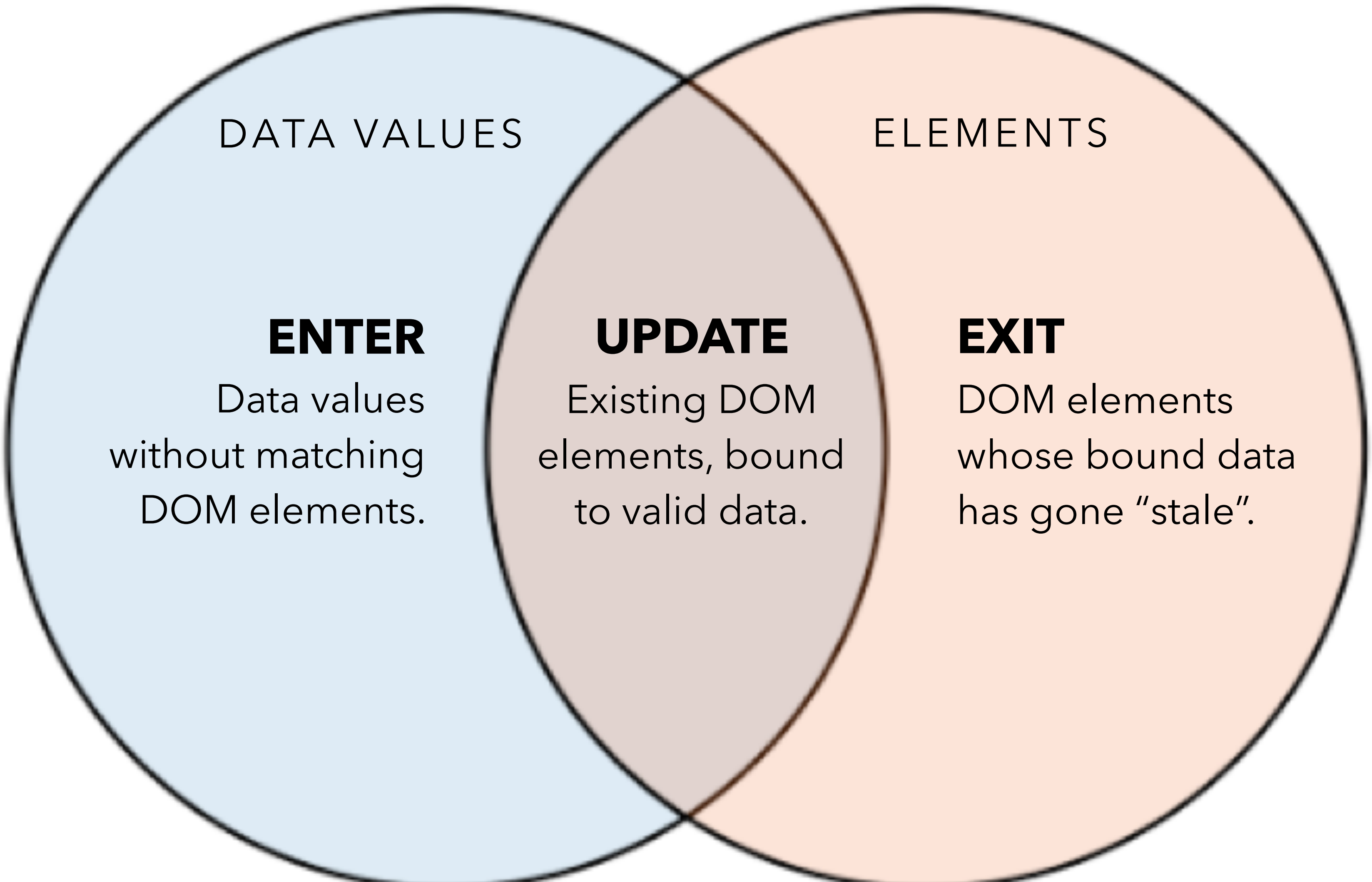
```
var values = [ {…}, {…}, {…}, … ]; // input data as JS objects

// Select SVG rectangles and bind them to data values.
var bars = svg.selectAll("rect.bars").data(values);

// What if the DOM elements don't exist yet? The enter set represents data
// values that do not yet have matching DOM elements.
bars.enter().append("rect").attr("class", "bars");
```

Data Binding

Selections can ***bind* data and DOM elements**.

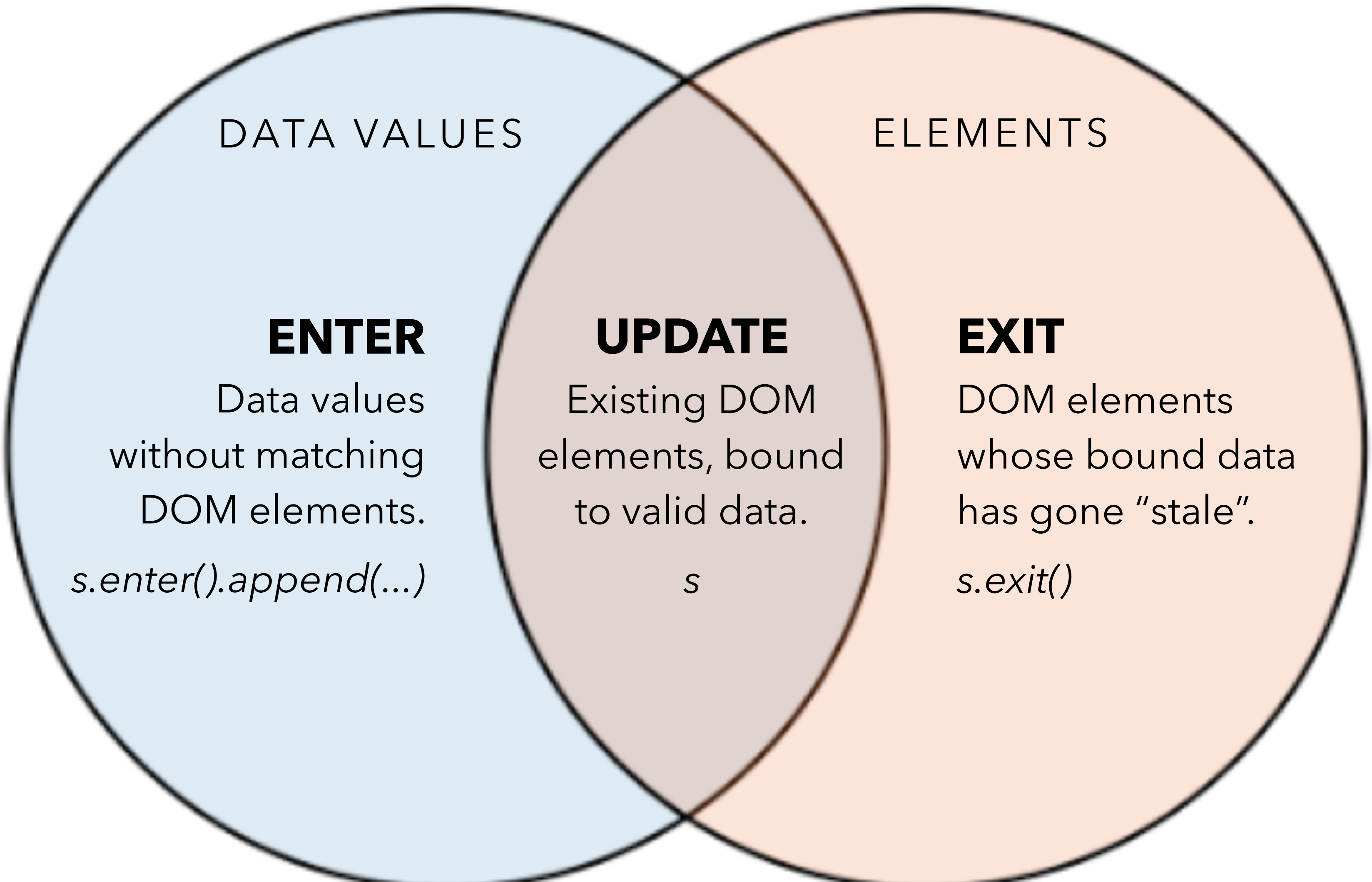
```
var values = [ {…}, {…}, {…}, … ]; // input data as JS objects

// Select SVG rectangles and bind them to data values.
var bars = svg.selectAll(“rect.bars”).data(values);

// What if the DOM elements don’t exist yet? The enter set represents data
// values that do not yet have matching DOM elements.
bars.enter().append(“rect”).attr(“class”, “bars”);

// What if data values are removed? The exit set is a selection of existing
// DOM elements who no longer have matching data values.
bars.exit().remove();
```

The Data Join



The Data Join

var s = d3.selectAll(...).data(...)



D3 Modules

Data Parsing / Formatting (JSON, CSV, …)
Shape Helpers (arcs, curves, areas, symbols, …)
Scale Transforms (linear, log, ordinal, …)
Color Spaces (RGB, HSL, LAB, …)
Animated Transitions (tweening, easing, …)
Geographic Mapping (projections, clipping, …)
Layout Algorithms (stack, pie, force, trees, …)
Interactive Behaviors (brush, zoom, drag, …)

Many of these correspond to future lecture topics!

Ease-of-Use

Expressiveness

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Visualization Grammars

Protovis, D3.js

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Administrivia

A2: Exploratory Data Analysis

Use visualization software to form & answer questions

First steps:

Step 1: Pick domain & data

Step 2: Pose questions

Step 3: Profile the data

Iterate as needed

Deliverable: A sequence of annotated visualizations that clearly communicate your findings.

Due by 11:59pm

Tonight!

Tutorials

Introduction to **Web Development** & **D3.js**

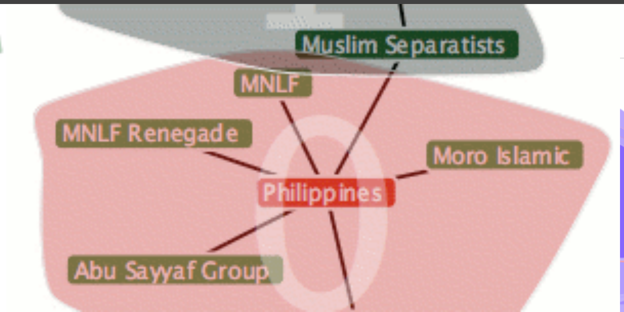
Next Lecture

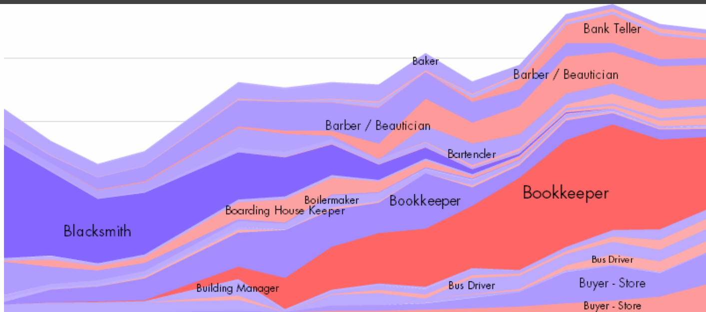
A3: Interactive Prototype

Create an interactive visualization. Choose a driving question for a dataset and develop an appropriate visualization + interaction techniques, then deploy your visualization on the web.

Due by *11:59pm* on **Monday, February 10**.

Work in project teams of 3-5 people.



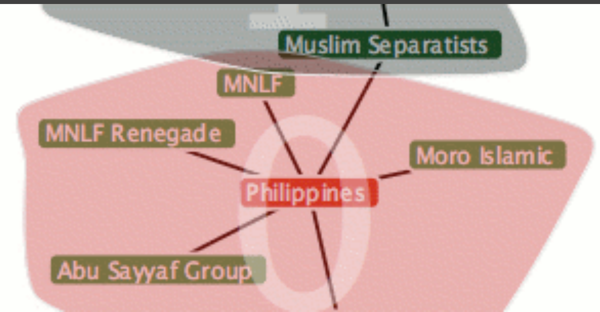


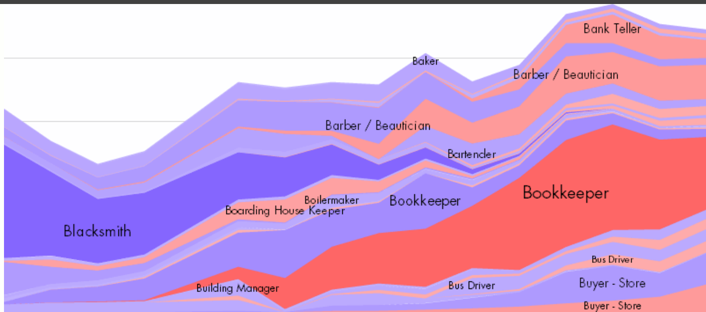
Requirements

Interactive. You must implement interaction methods! However, this is not only selection / filtering / tooltips. Also consider annotations or other narrative features to draw attention and provide additional context.

Web-based. D3 is encouraged, but not required. Deploy your visualization using GitHub pages.

Write-up. Provide design rationale on your web page.





A3 & Final Project Team

Form a **team of 3-5** for A3 and the Final Project.

Start thinking about your Final Project, too!

A3 is open-ended, but you can use it to start exploring your FP topic if you like.

Submit signup form by **Friday 1/31, 11:59pm**.

If you do not have team mates, you should:

- Use the facilities on Canvas/Piazza
- piazza.com/washington/winter2020/cse442
- Stay after class to meet potential partners

Team Member Roles

We encourage you to structure team responsibilities!

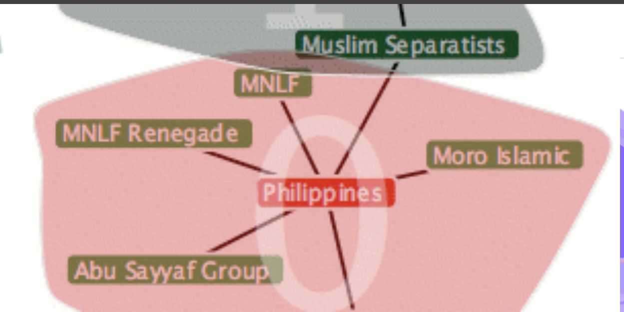
Coordinator: Organize meetings, track deadlines, *etc.*

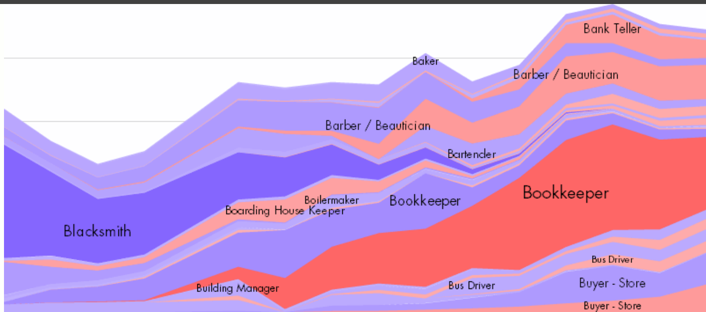
Data Lead: Data wrangling, management, distillation

Tech Lead: Manage code integration, GitHub repo

UX Lead: Visualization/interaction design & evaluation

One may have multiple roles, share work across roles…



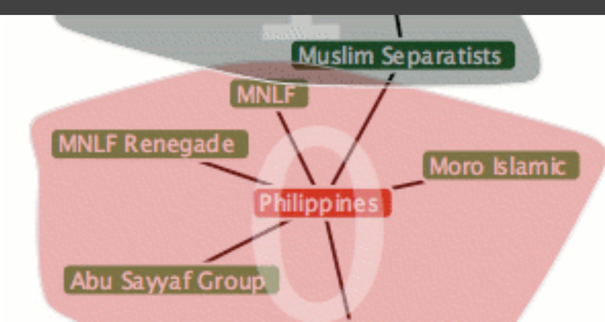


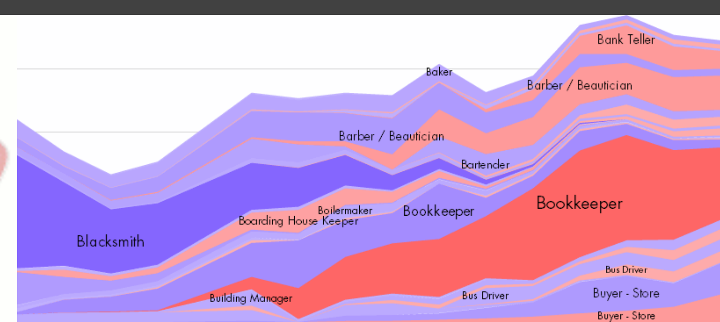
Interactive Prototype Tips

Start now. It will take longer than you think.

Keep it simple. Choose a *minimal* set of interactions that enables users to explore and generate interesting insights. Do not feel obligated to convey *everything* about the data: focus on a compelling subset.

Promote engagement. How do your chosen interactions reveal interesting observations?





Interactive Prototype Tips

Prototyping is a valuable activity. Feel free to use this exercise to your advantage for the final project.

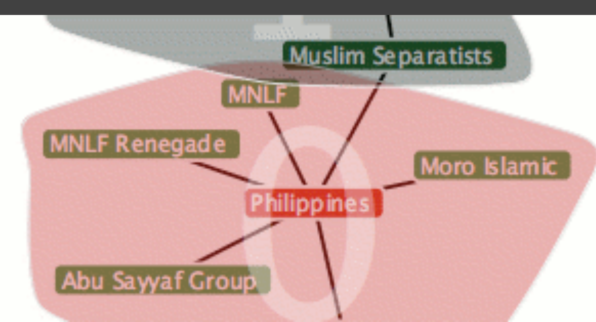
Final Project

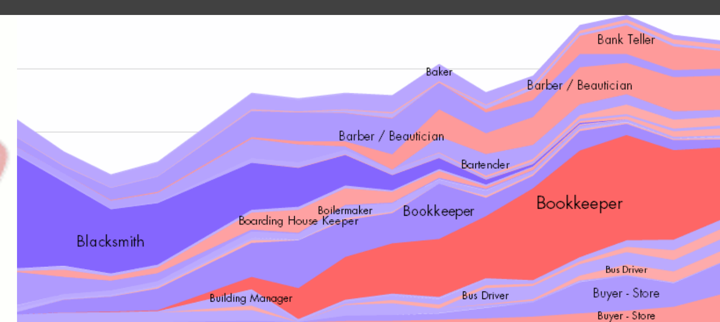
Interactive dashboard for non-expert users.

(*Politics, Sports, Climate, Finance, Entertainment, …*)

NEW! Explorable explanation of technical topic.

(Algorithm, Theorem, Science, …)





A Visualization Tool Stack

Chart Typologies

Excel, Many Eyes, Google Charts

Visual Analysis Grammars

VizQL, ggplot2

Visualization Grammars

Protovis, D3.js

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Chart Typologies

Excel, Many Eyes, Google Charts

Charting Tools

Visual Analysis Grammars

VizQL, ggplot2

Visualization Grammars

Protovis, D3.js

Declarative Languages

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Programming Toolkits

Chart Typologies
Excel, Many Eyes, Google Charts

Charting
Tools

Visual Analysis Grammars
VizQL, ggplot2

Visualization Grammars
Protovis, D3.js

Declarative
Languages

Component Architectures
Prefuse, Flare, Improvise, VTK

Graphics APIs
Processing, OpenGL, Java2D

Programming
Toolkits

What is a Declarative Language?

Programming by describing *what*, not *how*

Separate **specification** (*what you want*) from **execution** (*how it should be computed*)

In contrast to **imperative programming**, where you must give explicit steps.

TODAY'S PAPER | VIDEO | MOST POPULAR | TIMES TOPICS | MOST RECENT

Subscribe to The Times | Welcome, jmheer

The New York Times

Tuesday, October 26, 2010 Last Update: 3:50 PM ET

Search

ING DIRECT

Subscribe to Hom

— 2010 Midterm Elections —

Tea Party Vow to Deter Voter Fraud Is Called Scare Tactic

By IAN URBINA 2:19 PM ET

Voting rights group say that Tea Party members' plan to question voters' eligibility at the polls is intended to suppress minority and poor voters.

Post a Comment | Read (355)

Glaxo Pays $750 Million Fine for Tainted Products

By GARDINER HARRIS and DUFF

Joshua Kristal for The New York Times

Painting at 99, With No Compromises

By ROBIN FINN

An exhibition celebrating Will Barnet's centennial year traces his evolution as a modern American artist.

OPINION »

OP-ED CONTRIBUTOR

Humans to Asteroids: Watch Out!

How to keep near-Earth objects from hitting us.

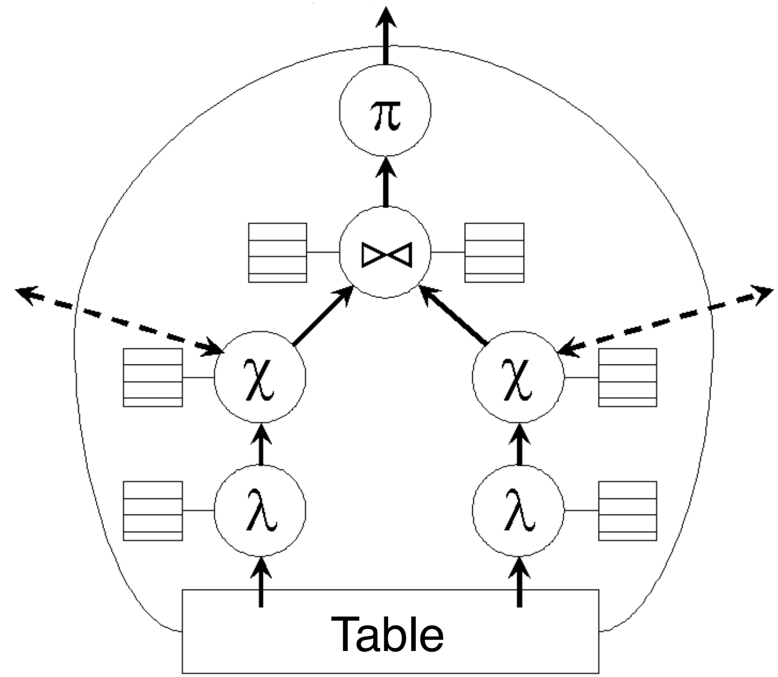
- Brooks: No Second Thoughts | Comments (200)
- Herbert: The Corrosion of America
- Cohen: Turkey Steps Out
- Editorial: Mortgage Mess
- Bloggingheads: Jon Stewart's Power

MARKETS » At 3:56 PM ET

S.&P. 500 | Dow | Nasdaq

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<!--[if IE]><![endif]-->
▼<html>
  ▶<head>…</head>
  ▼<body id="home" style="visibility: visible; ">
     <script src="http://connect.facebook.net/en_US/all.js"></scrip
     <div id="fb-root"></div>
     <a name="top"></a>
    ▼<div id="shell">
      ▶<ul id="memberTools">…</ul>
       <!-- ADXINFO classification="text_ad" campaign="nyt2010-circ-
      ▶<div class="tabsContainer">…</div>
       <!-- close .tabsContainer -->
      ▶<div id="page" class="tabContent active">…</div>
       <!--close page -->
     </div>
     <!--close shell -->
    ▶<script type="text/javascript" language="JavaScript">…</script
     <img src="/adx/bin/clientside/
     1b1a1755Q2F,.tWdC,KiQ2BCQ5C6iSM.KCYQ2FMadYYQ24xN!Q2B!xQ24WN4aN
     <script src="http://graphics8.nytimes.com/js/test/wto/wt_capi.
     charset="utf-"></script>
     <span id="wto_script"></span>
    ▶<script type="text/javascript"></script>
     <img src="http://graphics8.nytimes.com/ads/blank.gif">
     <script type="text/javascript" src="http://graphics8.nytimes.c
```

HTML / CSS




```sql
SELECT customer_id, customer_name,
    COUNT(order_id) as total
FROM customers
INNER JOIN orders ON
    customers.customer_id
    = orders.customer_id
GROUP BY customer_id, customer_nam
HAVING COUNT(order_id) > 5
ORDER BY COUNT(order_id) DESC
```

SQL

Why Declarative Languages?

Faster iteration. Less code. Larger user base.

Better visualization. *Smart defaults.*

Reuse. *Write-once, then re-apply.*

Performance. *Optimization, scalability.*

Portability. *Multiple devices, renderers, inputs.*

Programmatic generation.
Write programs which output visualizations.
Automated search & recommendation.

Chart Typologies
Excel, Many Eyes, Google Charts

Charting Tools

Visual Analysis Grammars
VizQL, ggplot2

Visualization Grammars
Protovis, D3.js

Declarative Languages

Component Architectures
Prefuse, Flare, Improvise, VTK

Graphics APIs
Processing, OpenGL, Java2D

Programming Toolkits

Chart Typologies

Excel, Many Eyes, Google Charts

Charting Tools

Visual Analysis Grammars

VizQL, ggplot2, ***Vega-Lite***

Visualization Grammars

Protovis, D3.js, ***Vega***

Declarative Languages

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Programming Toolkits

Chart Typologies ?!

Excel, Many Eyes, Google Charts

Charting Tools

Visual Analysis Grammars

VizQL, ggplot2, ***Vega-Lite***

Visualization Grammars

Protovis, D3.js, ***Vega***

Declarative Languages

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Programming Toolkits

Visual Analysis Grammars

VizQL, ggplot2, ***Vega-Lite***

Visualization Grammars

Protovis, D3.js, ***Vega***

Declarative Languages

Component Architectures

Prefuse, Flare, Improvise, VTK

Graphics APIs

Processing, OpenGL, Java2D

Programming Toolkits

Interactive Data Exploration

Tableau, ***Lyra, Polestar, Voyager***

Graphical Interfaces

Visual Analysis Grammars

VizQL, ggplot2, ***Vega-Lite***

Visualization Grammars

Protovis, D3.js, ***Vega***

Declarative Languages

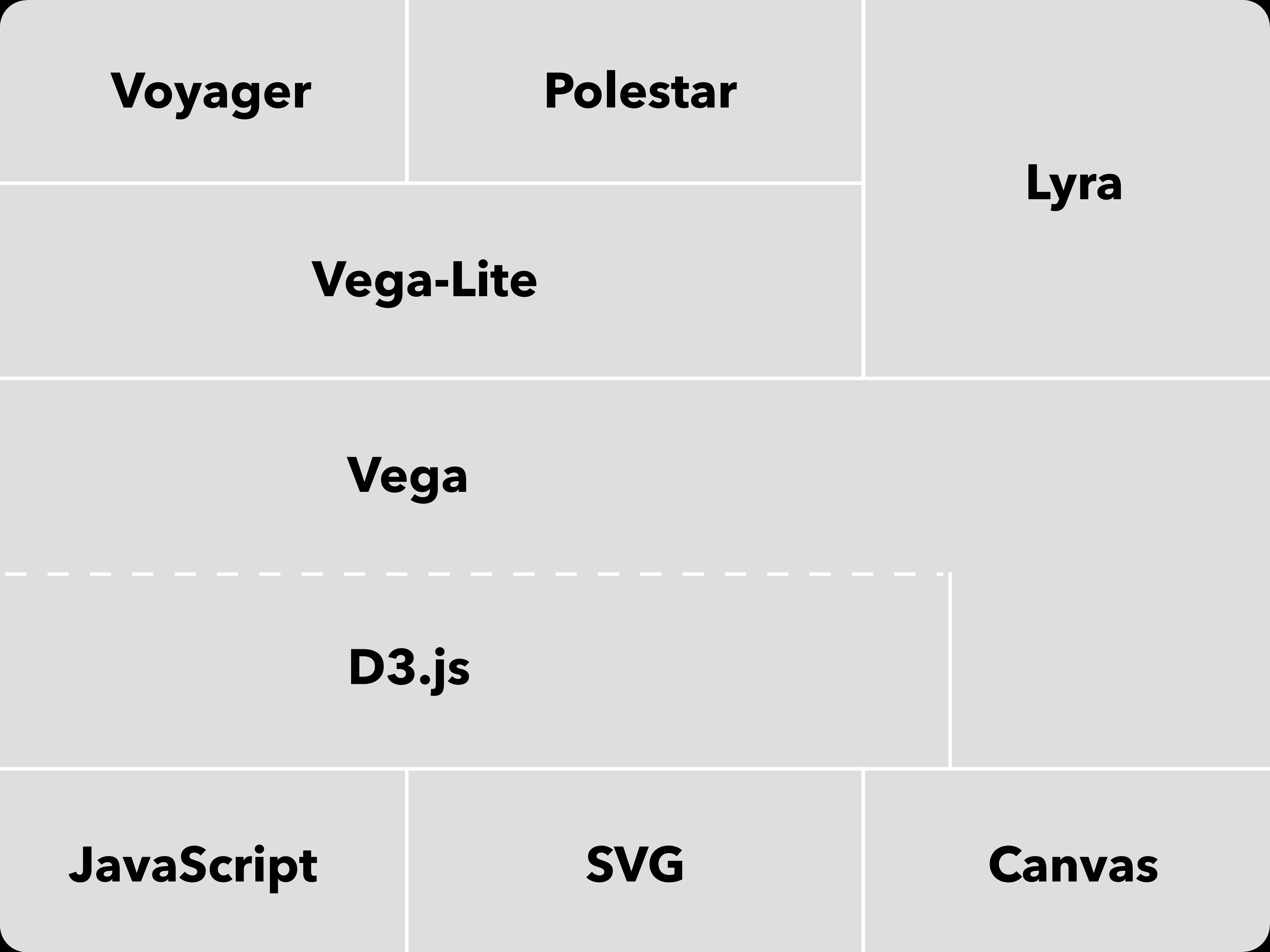
Component Architectures

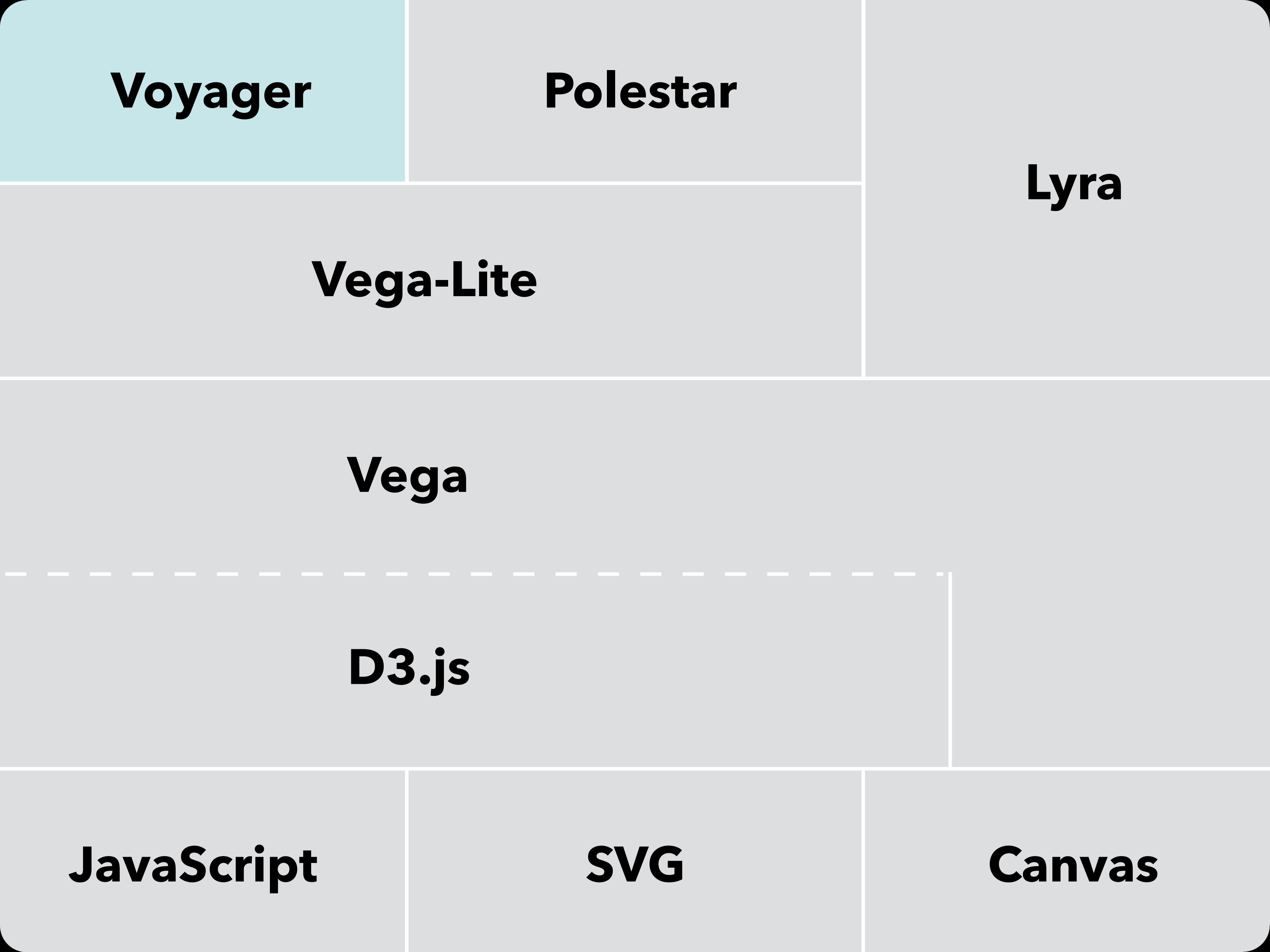
Prefuse, Flare, Improvise, VTK

Programming Toolkits

Graphics APIs

Processing, OpenGL, Java2D


Voyager
Polestar
Lyra
Vega-Lite
Vega
D3.js
JavaScript
SVG
Canvas


Voyager
Polestar
Lyra
Vega-Lite
Vega
D3.js
JavaScript
SVG
Canvas

Voyager

Reduce tedious manual specification

Voyager

Reduce tedious manual specification

Support early-stage data exploration

Encourage *data coverage*

Discourage *premature fixation*

Voyager

Reduce tedious manual specification

Support early-stage data exploration

Encourage *data coverage*

Discourage *premature fixation*

Approach: browse a gallery of visualizations

Voyager

Reduce tedious manual specification

Support early-stage data exploration
Encourage *data coverage*
Discourage *premature fixation*

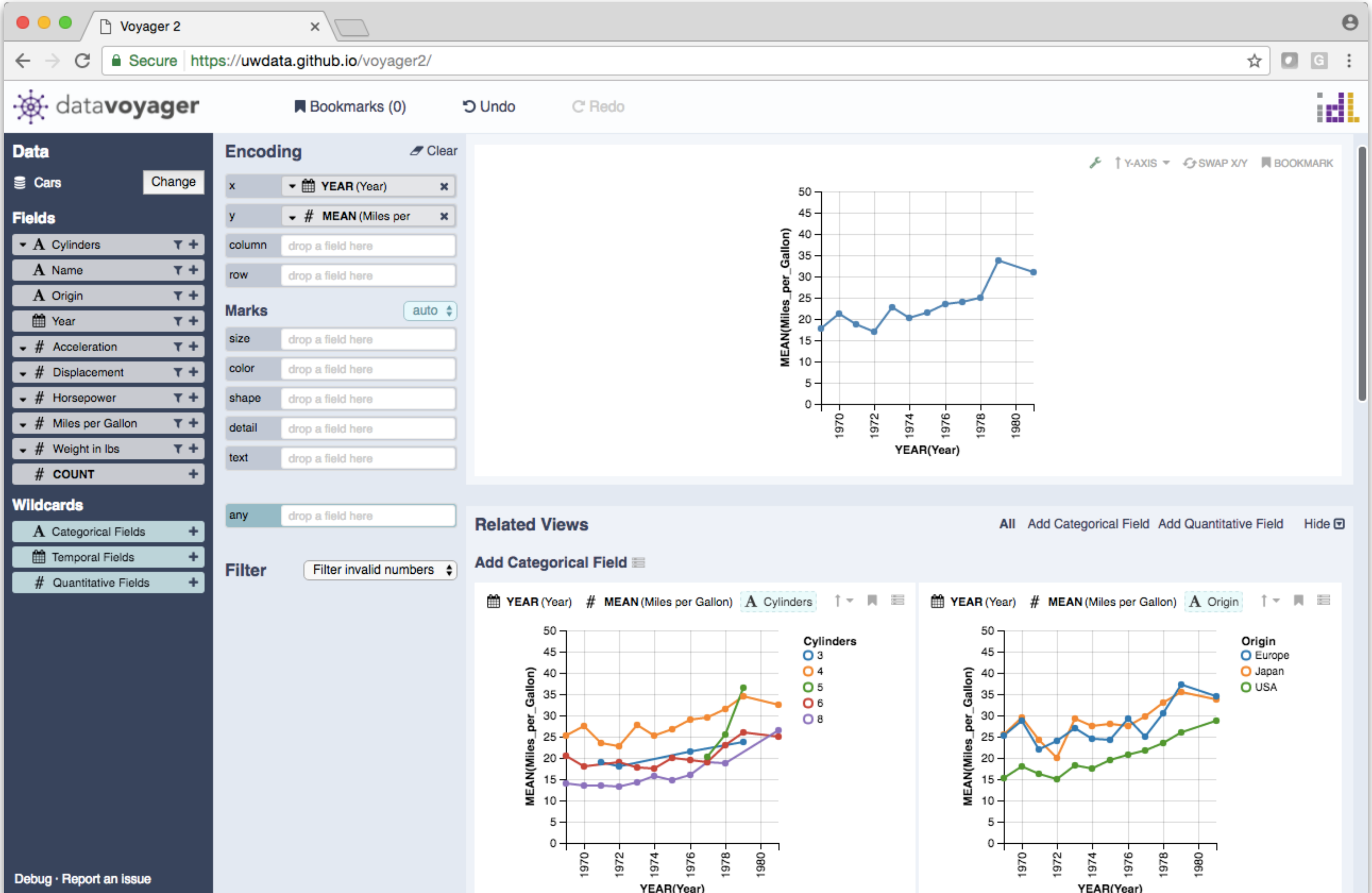
Approach: browse a gallery of visualizations
Challenge - *combinatorial explosion!*

Voyager

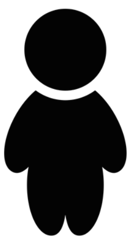
Reduce tedious manual specification

Support early-stage data exploration
Encourage *data coverage*
Discourage *premature fixation*

Approach: browse a gallery of visualizations
Challenge - *combinatorial explosion!*
Automatic recommendation of useful views
+ **end-user steering** to focus exploration

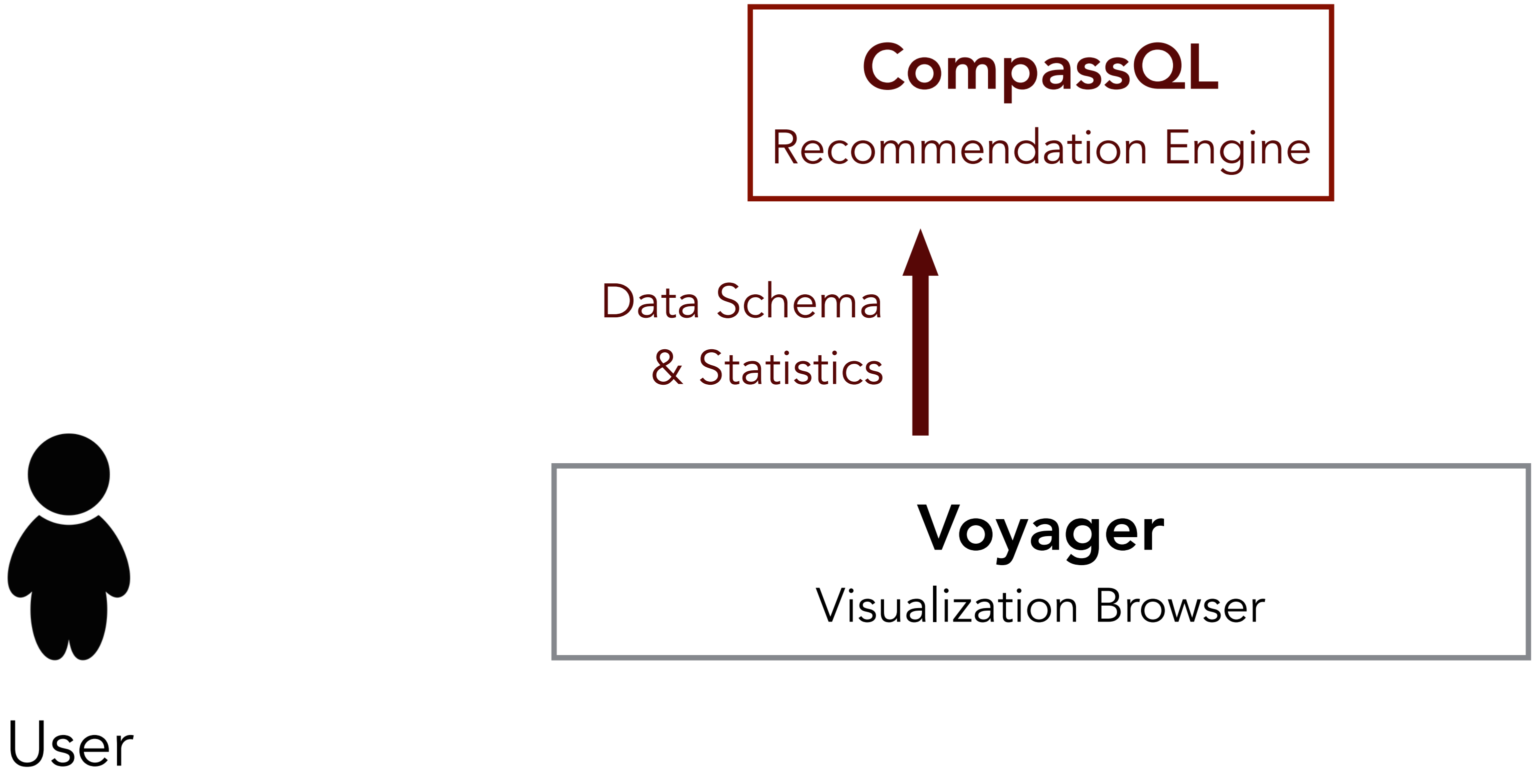


Voyager. Wongsuphasawat et al. *InfoVis'15, CHI'17*



User


Data Set
Voyager
Visualization Browser
User


CompassQL
Recommendation Engine
Data Schema
& Statistics
Voyager
Visualization Browser
User

1. Select **data variables**
2. Apply **transformations**
3. Pick visual **encodings**

CompassQL

Recommendation Engine

Data Schema
& Statistics

Voyager

Visualization Browser

User

Constrain & rank choices by **data type**, **statistics** & **perceptual principles.**

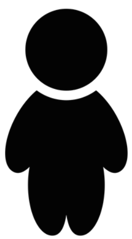
CompassQL

Recommendation Engine

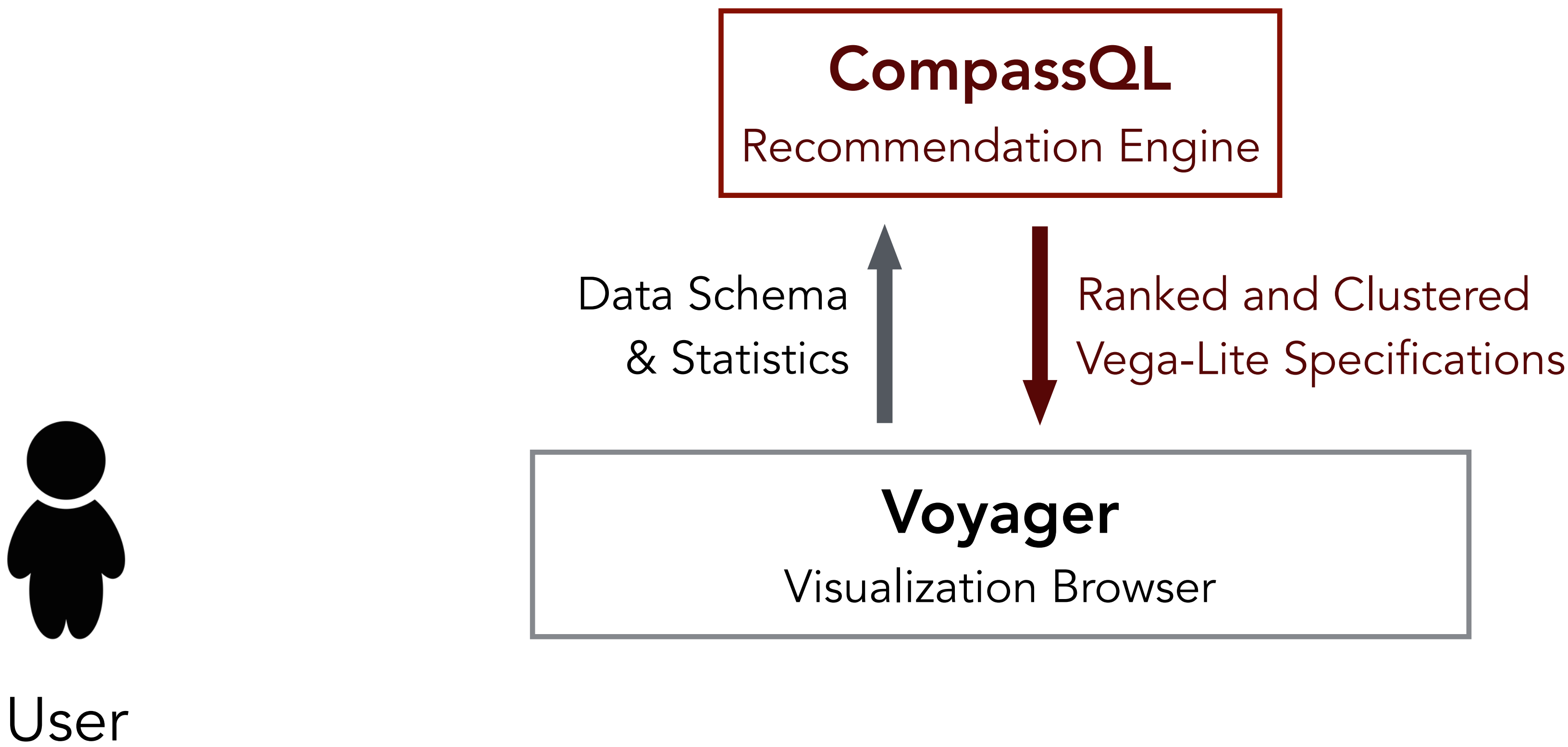
Data Schema
& Statistics

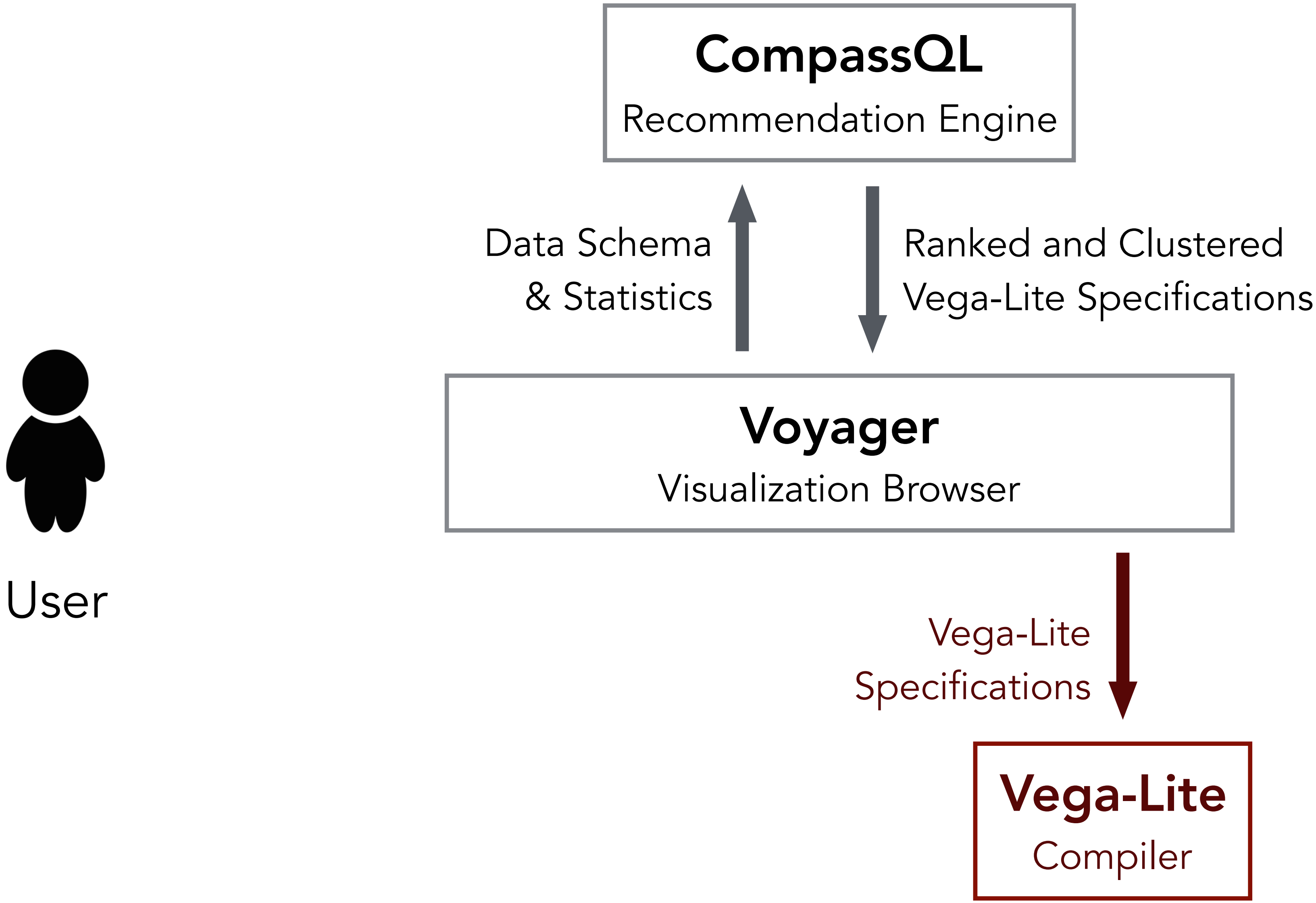
Voyager

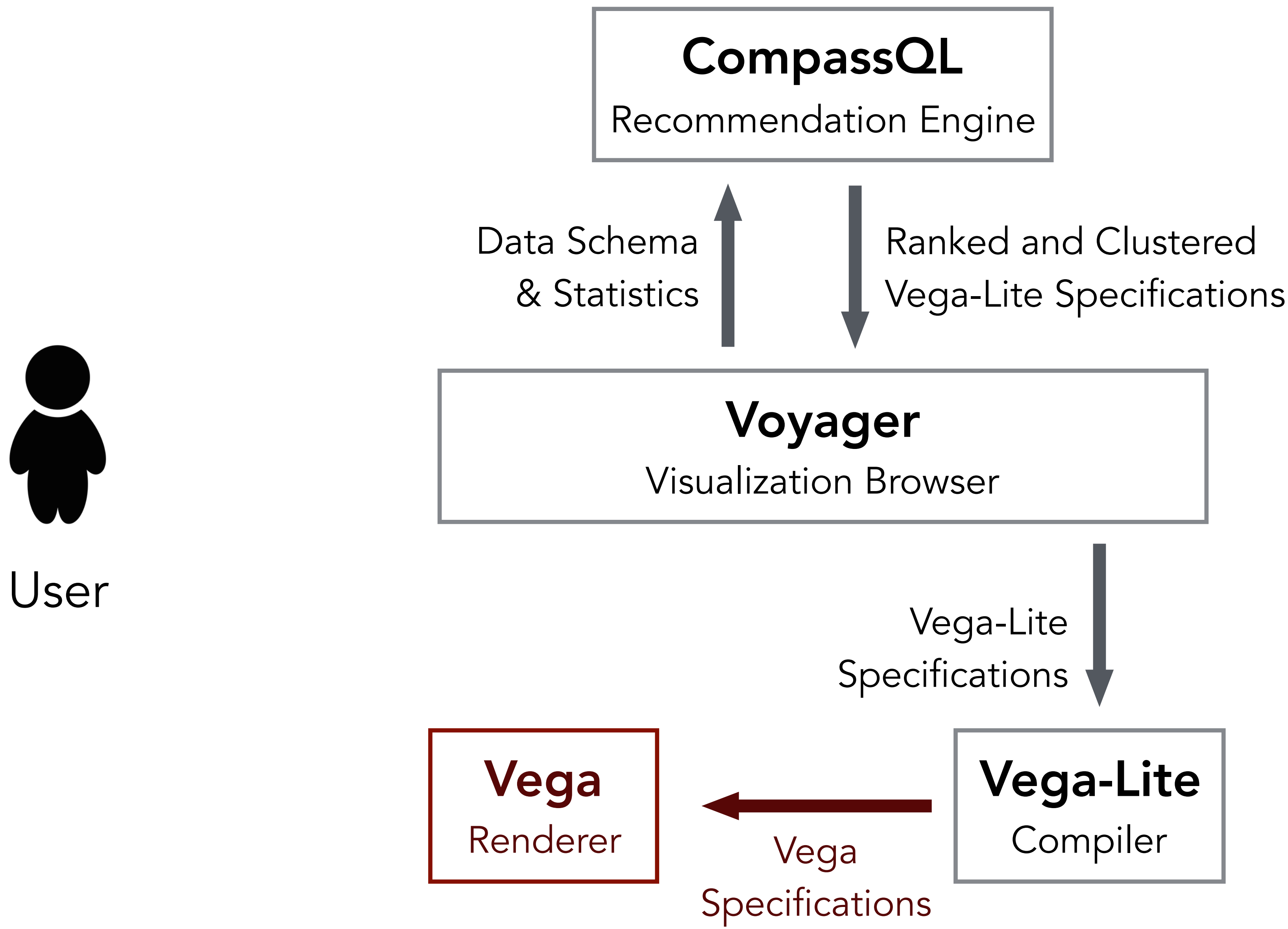
Visualization Browser

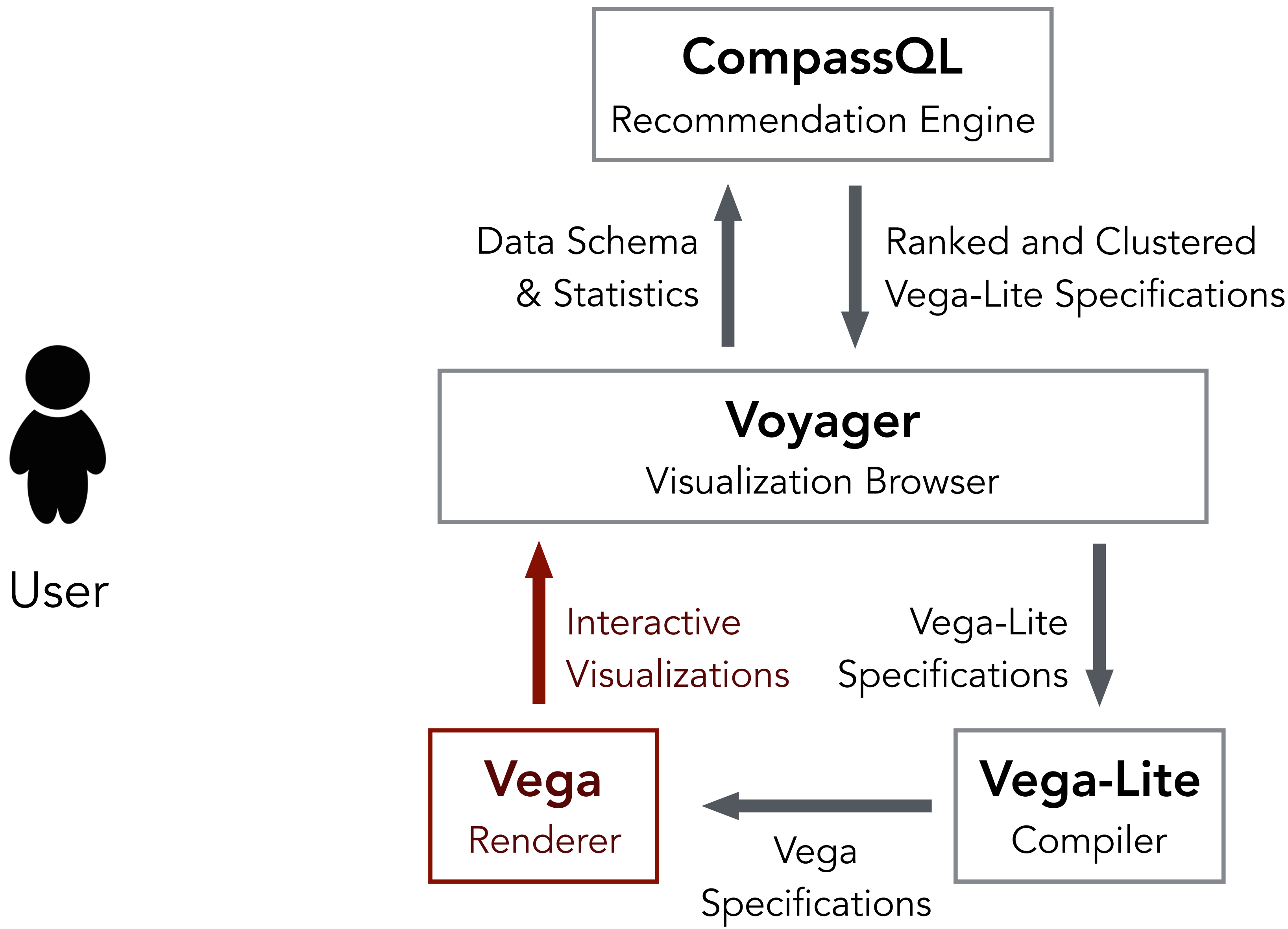


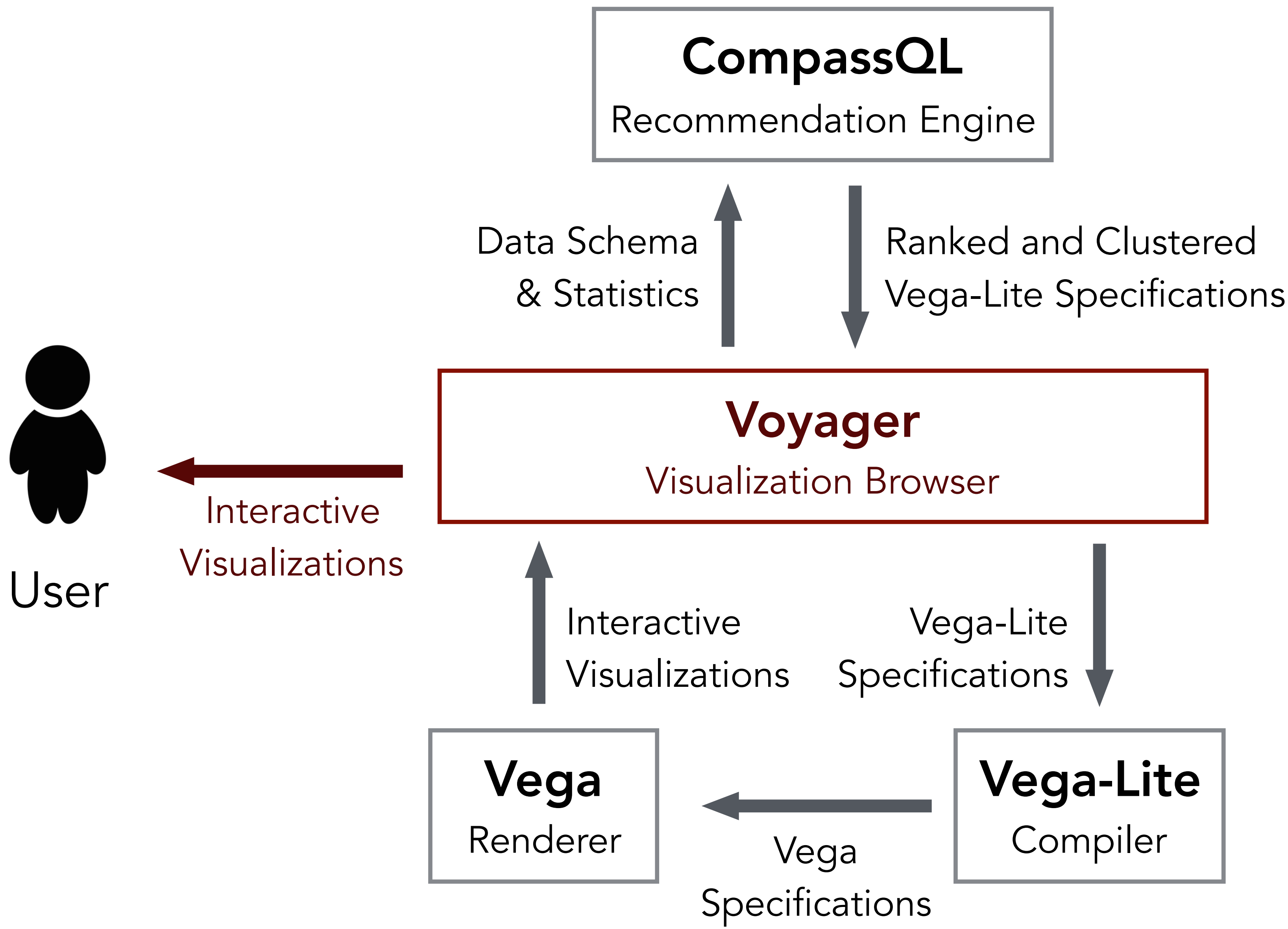
User

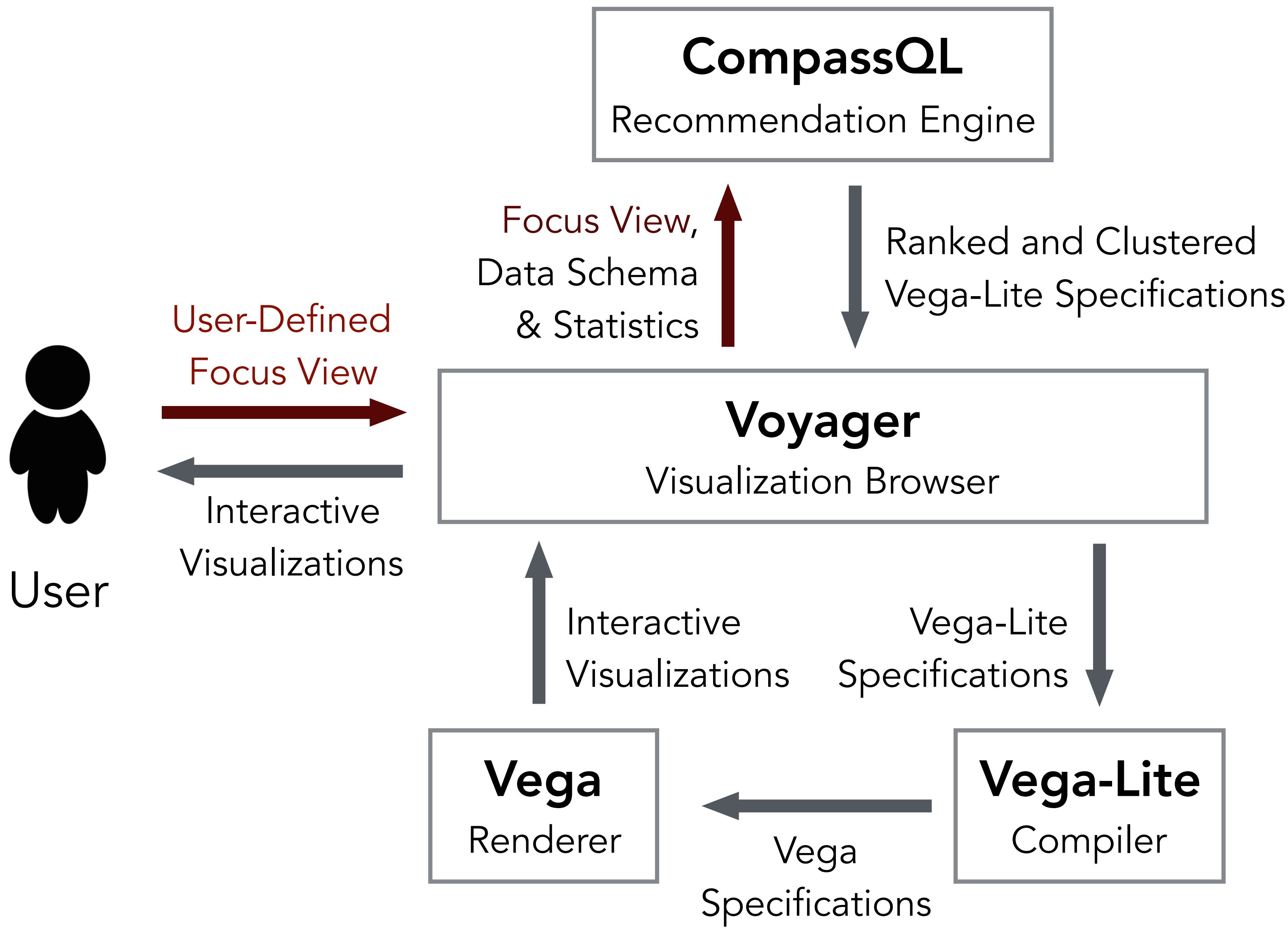

CompassQL
Recommendation Engine
Data Schema
& Statistics
Ranked and Clustered
Vega-Lite Specifications
Voyager
Visualization Browser
User


CompassQL
Recommendation Engine
Data Schema
& Statistics
Ranked and Clustered
Vega-Lite Specifications
User
Voyager
Visualization Browser
Vega-Lite
Specifications
Vega-Lite
Compiler


CompassQL
Recommendation Engine
Data Schema
& Statistics
Ranked and Clustered
Vega-Lite Specifications
User
Voyager
Visualization Browser
Vega-Lite
Specifications
Vega
Renderer
Vega
Specifications
Vega-Lite
Compiler


CompassQL
Recommendation Engine
Data Schema
& Statistics
Ranked and Clustered
Vega-Lite Specifications
User
Voyager
Visualization Browser
Interactive
Visualizations
Vega-Lite
Specifications
Vega
Renderer
Vega
Specifications
Vega-Lite
Compiler

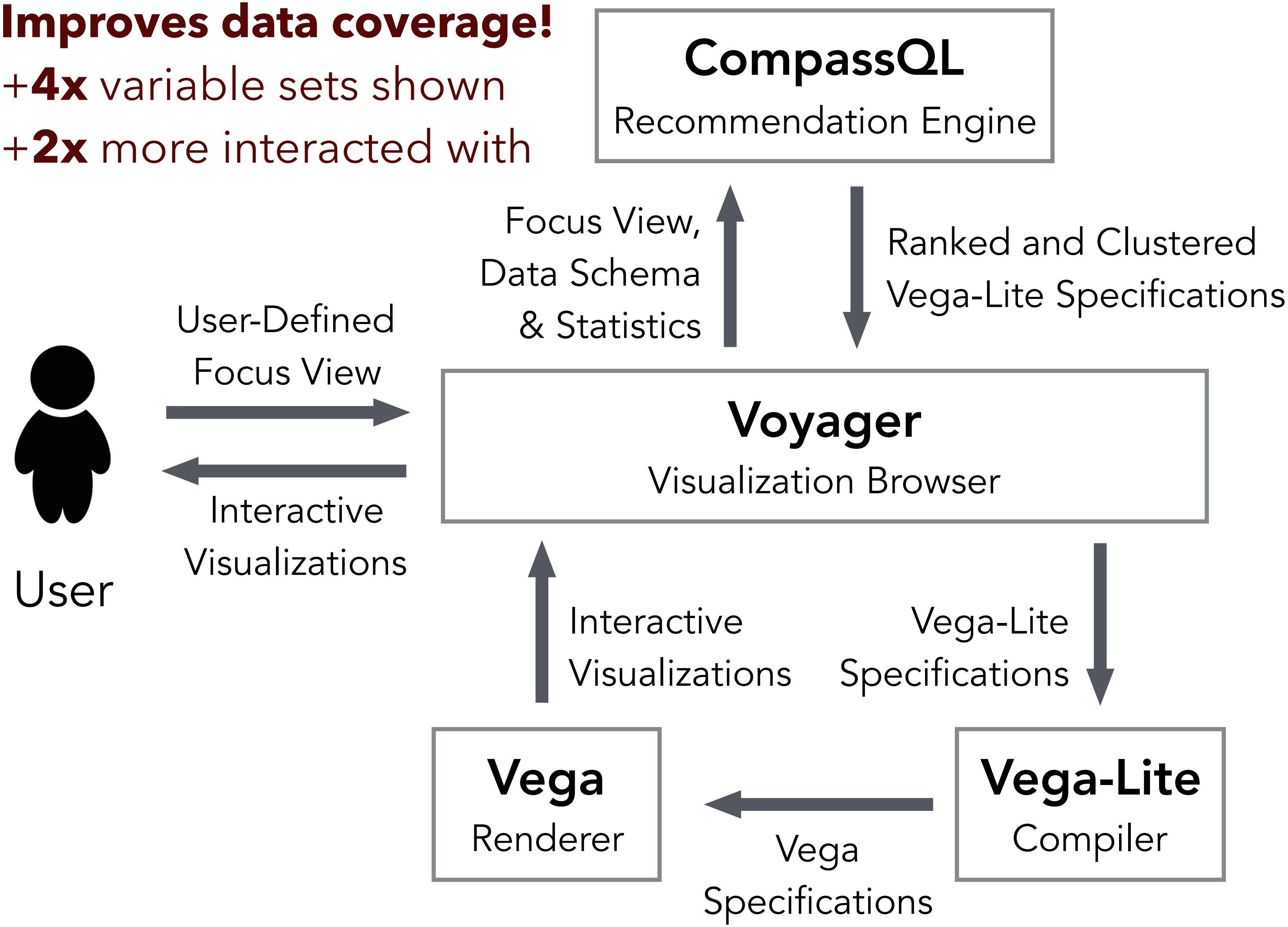

CompassQL
Recommendation Engine
Data Schema
& Statistics
Ranked and Clustered
Vega-Lite Specifications
Voyager
Visualization Browser
User
Interactive
Visualizations
Interactive
Visualizations
Vega-Lite
Specifications
Vega
Renderer
Vega-Lite
Compiler
Vega
Specifications

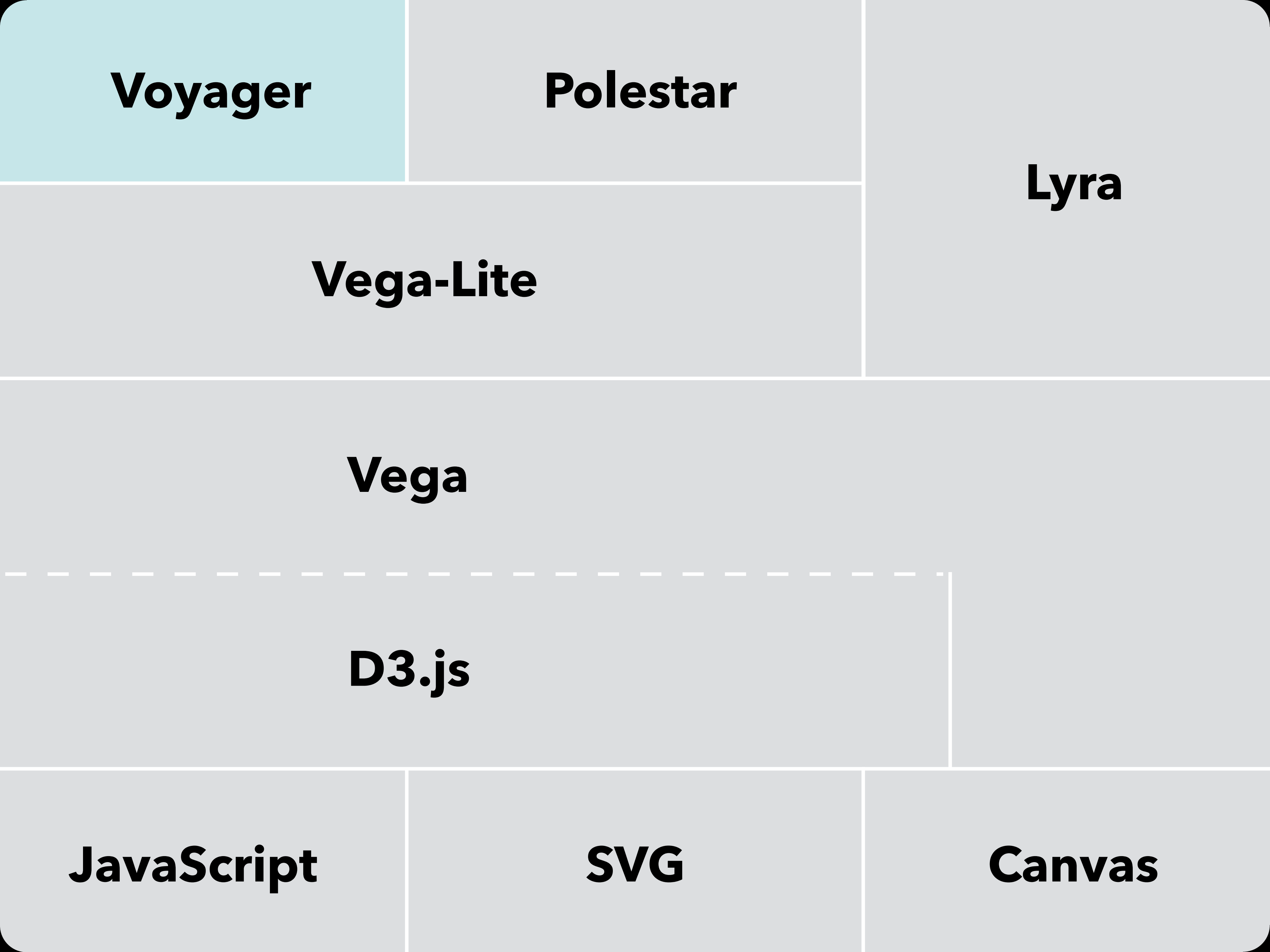

CompassQL
Recommendation Engine
Focus View,
Data Schema
& Statistics
Ranked and Clustered
Vega-Lite Specifications
User-Defined
Focus View
User
Interactive
Visualizations
Voyager
Visualization Browser
Interactive
Visualizations
Vega-Lite
Specifications
Vega
Renderer
Vega
Specifications
Vega-Lite
Compiler

Improves data coverage!

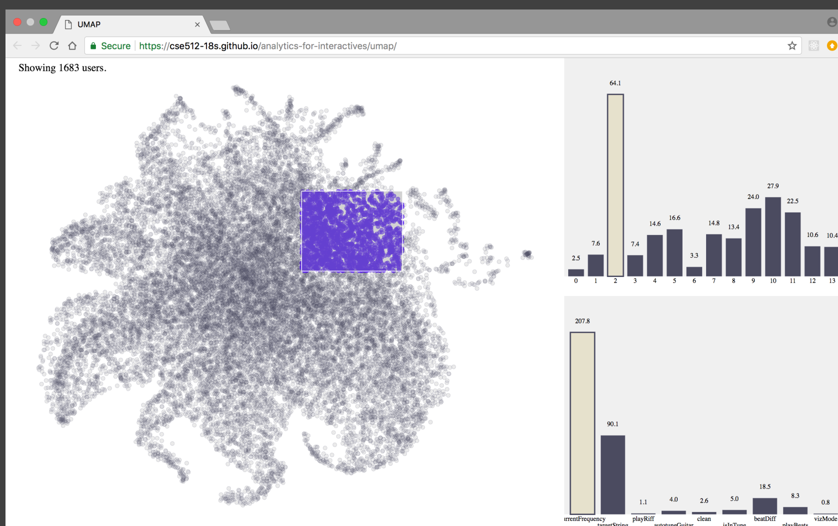
+**4x** variable sets shown

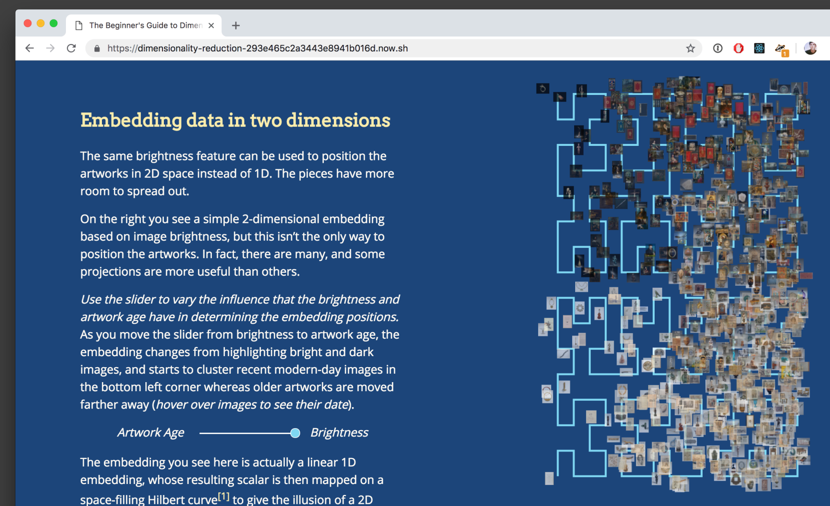
+**2x** more interacted with



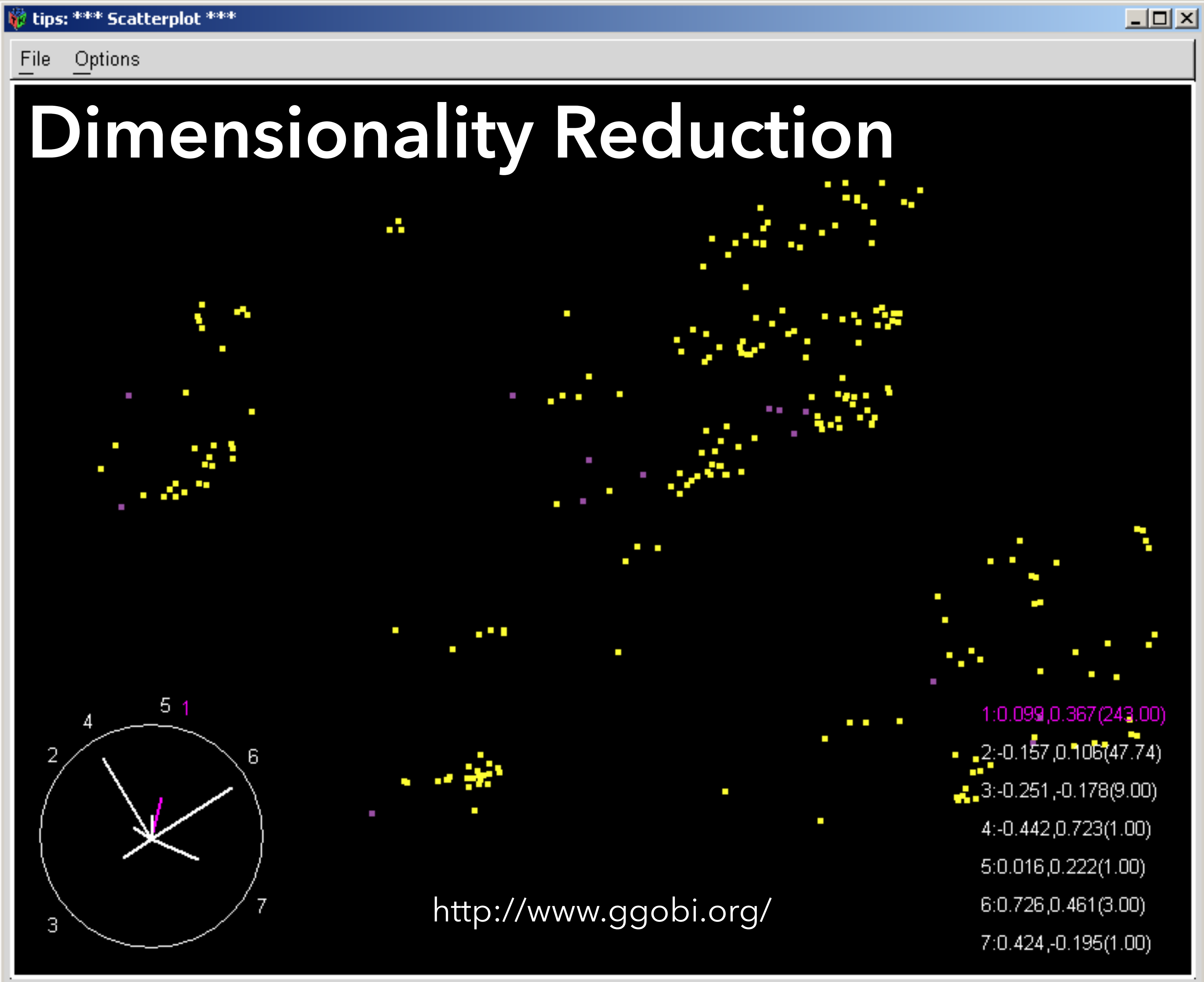

Voyager
Polestar
Lyra
Vega-Lite
Vega
D3.js
JavaScript
SVG
Canvas

Evaluating Interactive Articles



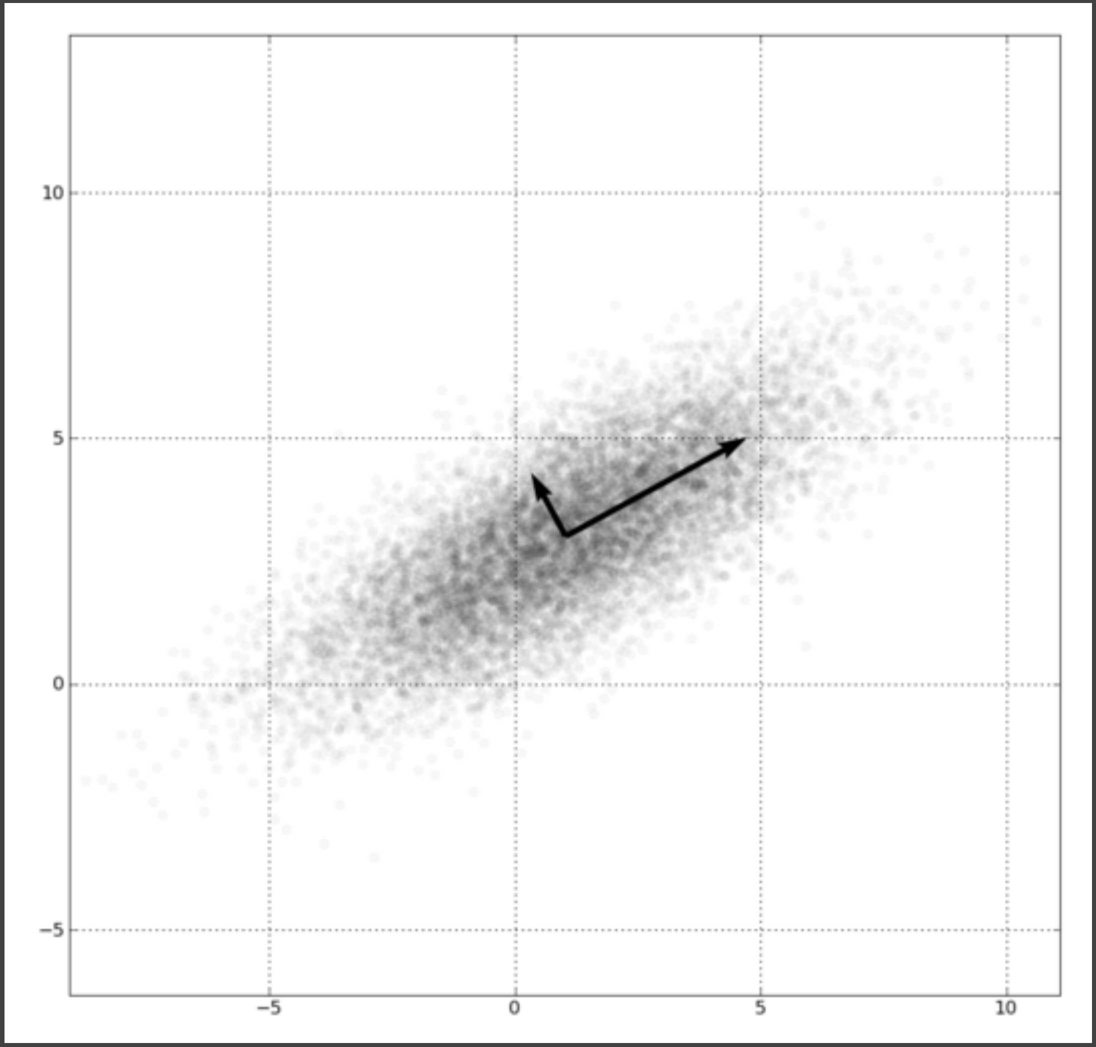


Dimensionality Reduction



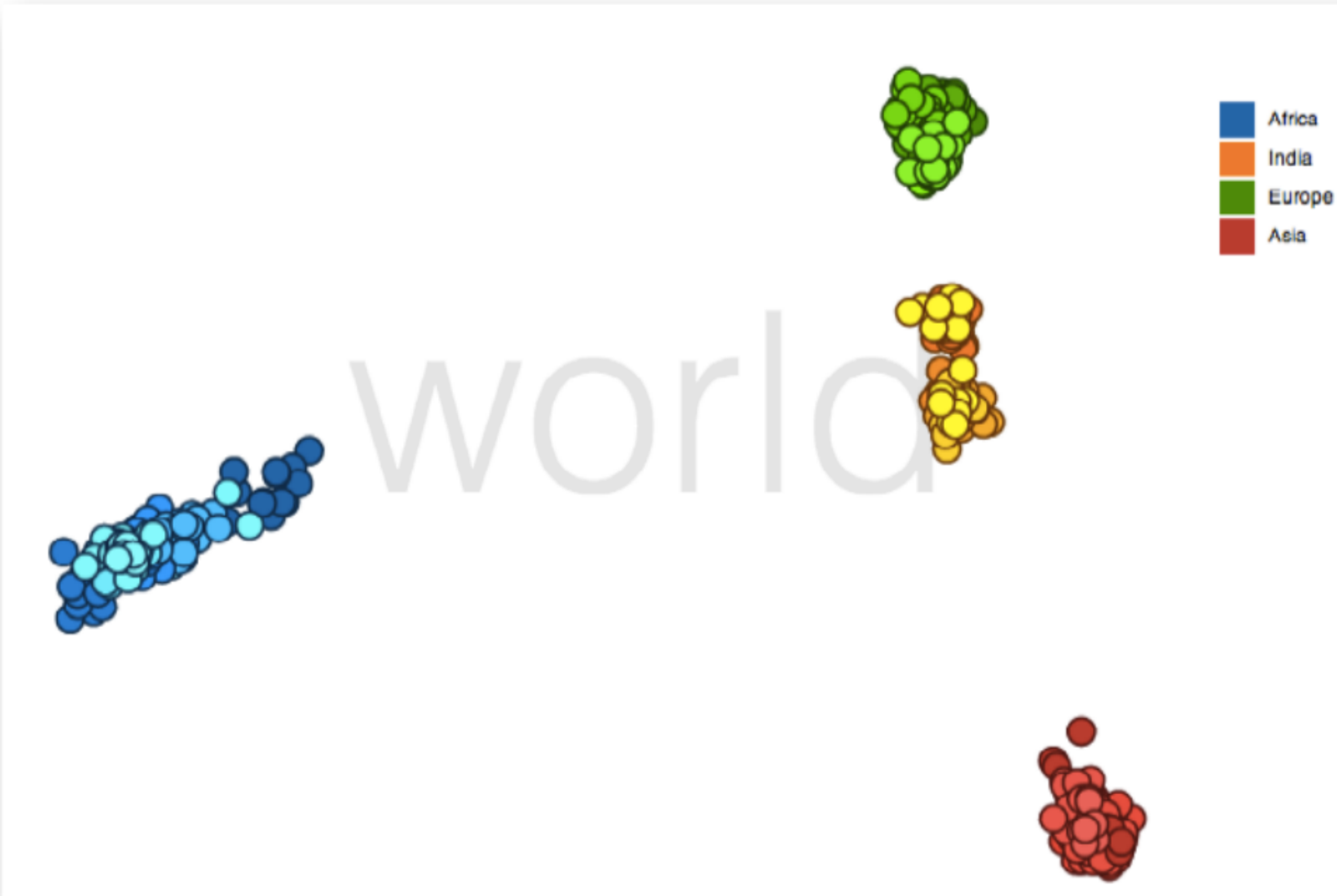
http://www.ggobi.org/

Principal Components Analysis



1. Mean-center the data.
2. Find ⊥ basis vectors that maximize the data variance.
3. Plot the data using the top vectors.

PCA of Genomes [Demiralp et al. '13]



Many Reduction Techniques!

General Strategies:

Matrix Factorization

Nearest Neighbor (Topological) Methods

Popular Techniques:

Principal Components Analysis (PCA)

t-Dist. Stochastic Neighbor Embedding (t-SNE)

Uniform Manifold Approx. & Projection (UMAP)

The Beginner's Guide to Dimensionality Reduction

Explore the methods that data scientists use to visualize high-dimensional data.

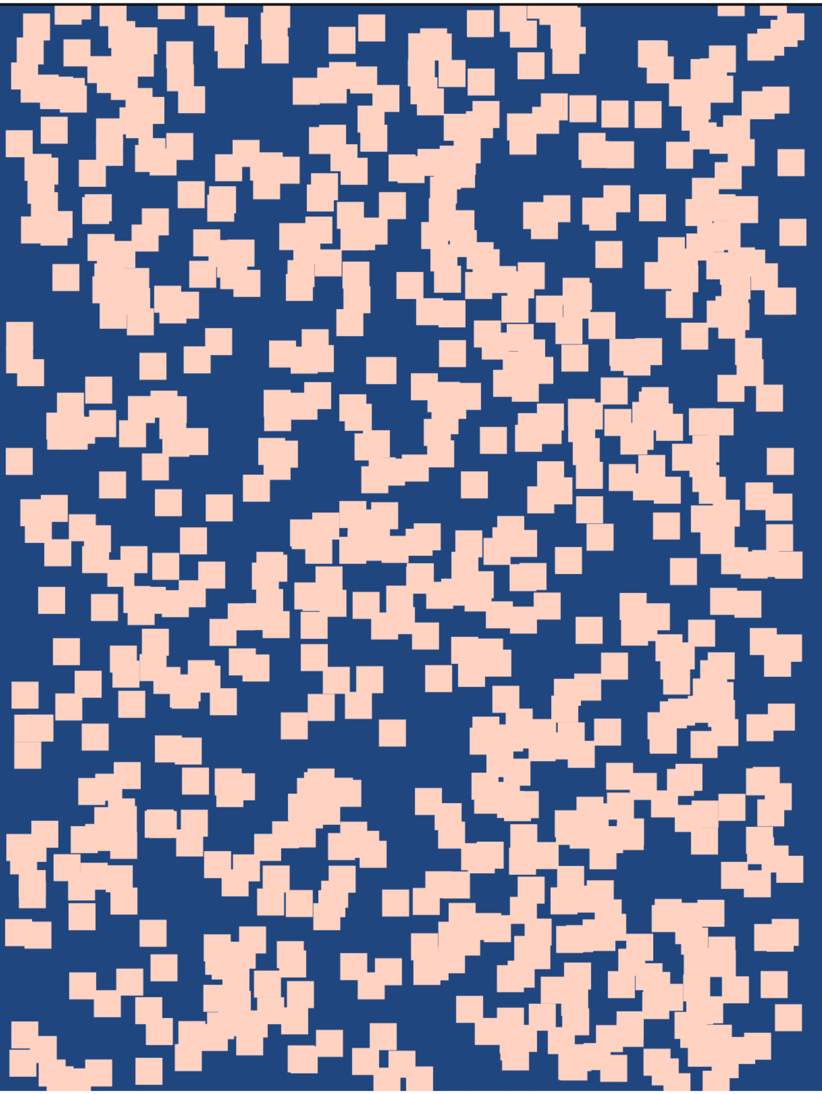
By: Matthew Conlen and Fred Hohman

July 16, 2018

Dimensionality reduction is a powerful technique used by data scientists to look for hidden structure in data. The method is useful in a number of domains, for example document categorization, protein disorder prediction, and machine learning model debugging[2].

The results of a dimensionality reduction algorithm can be visualized to reveal patterns and clusters of similar or dissimilar data. Even though the data is displayed in only two or three dimensions, structures present in higher dimensions are maintained, at least roughly[7].

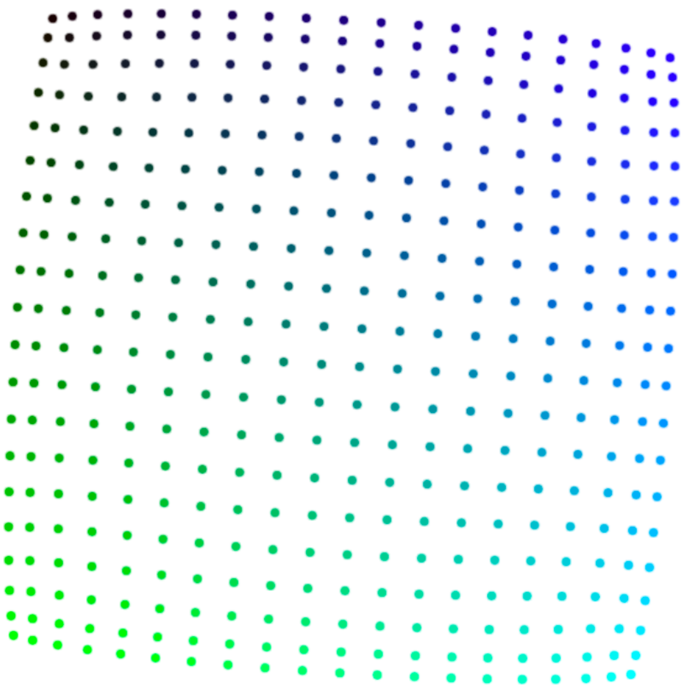
The technique is available in many applications, for



Distill ABOUT PRIZE SUBMIT

How to Use t-SNE Effectively

Although extremely useful for visualizing high-dimensional data, t-SNE plots can sometimes be mysterious or misleading. By exploring how it behaves in simple cases, we can learn to use it more effectively.



Step 1,910

A square grid with equal spacing between points. Try convergence at different sizes.

Points Per Side 20

Perplexity 10

Epsilon 5

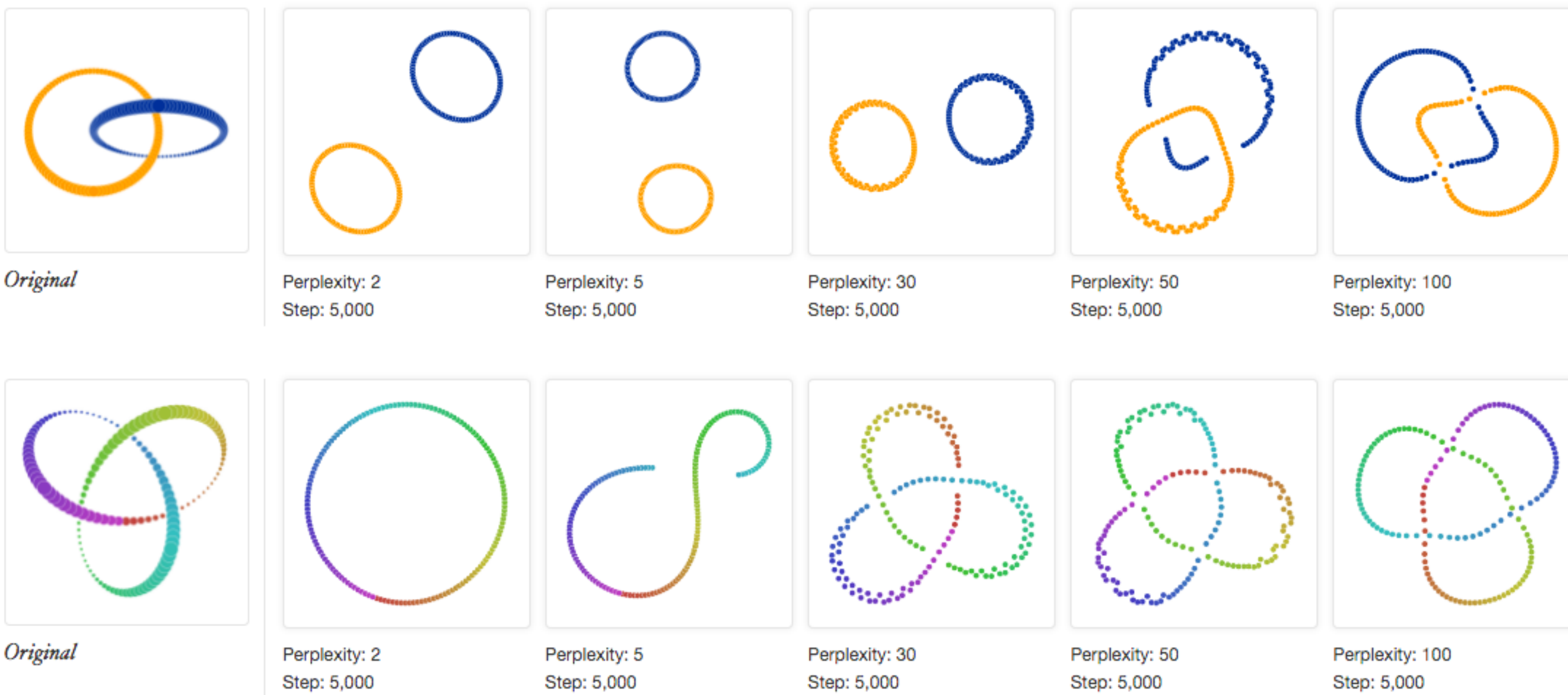
distill.pub

MARTIN WATTENBERG Google Brain
FERNANDA VIÉGAS Google Brain
IAN JOHNSON Google Cloud

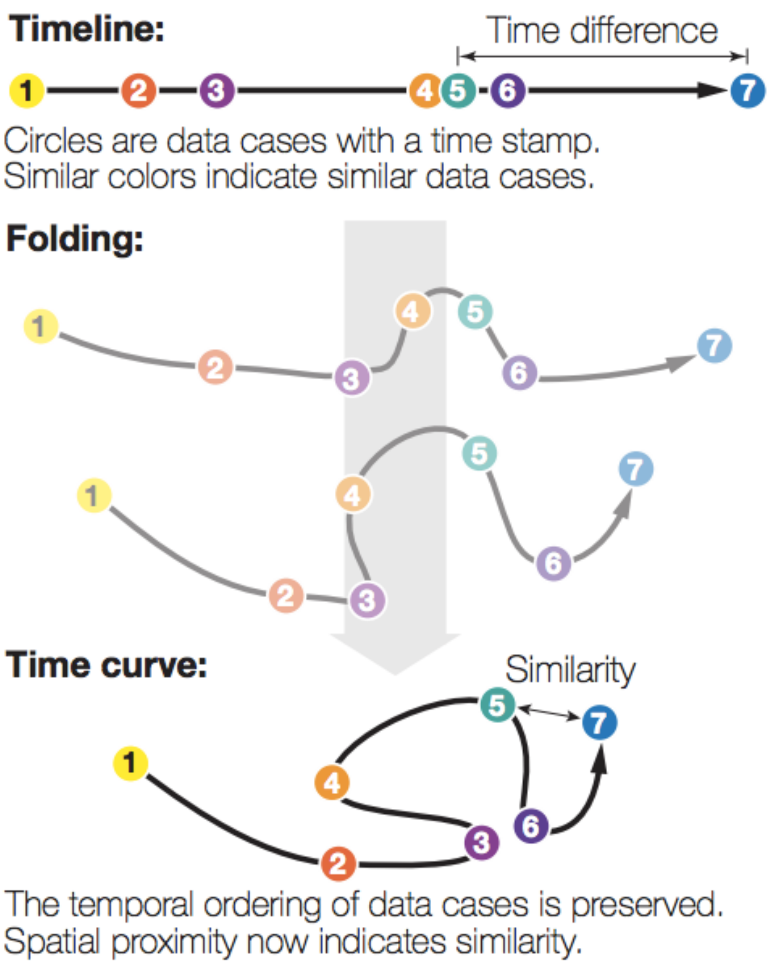
Oct. 13 2016

Citation: Wattenberg, et al., 2016

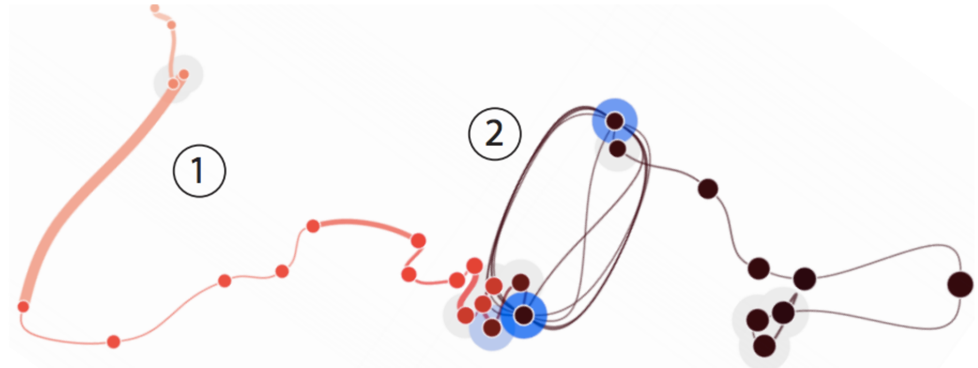
Visualizing t-SNE [Wattenberg et al. '16]



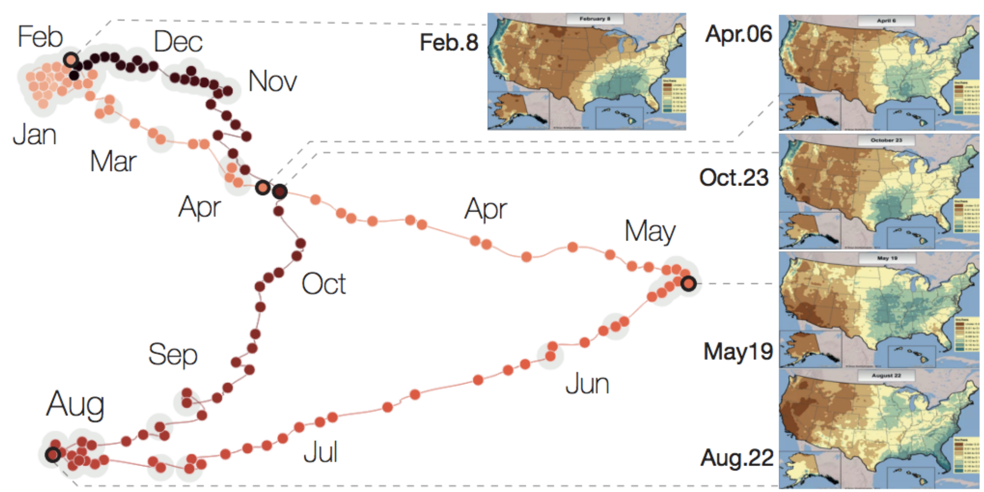
Time Curves [Bach et al. '16]



(a) Folding time



Wikipedia "Chocolate" Article



U.S. Precipitation over 1 Year