

# Analisi e Visualizzazione di Reti Complesse

**DV08 - Interactive Views**

**Prof. Rossano Schifanella**

**DV08 - Interactive Views - a.a. 2023-2024**



# How to handle complexity: 1 previous strategy

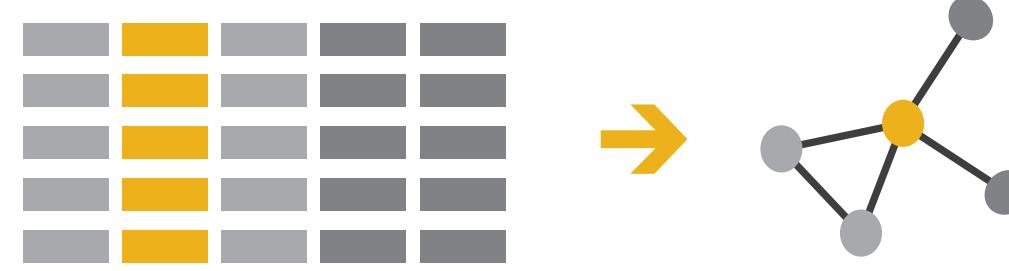
→ *Derive*



**derive new data to show within the view**

# How to handle complexity: 1 previous strategy + 2 more

## → Derive

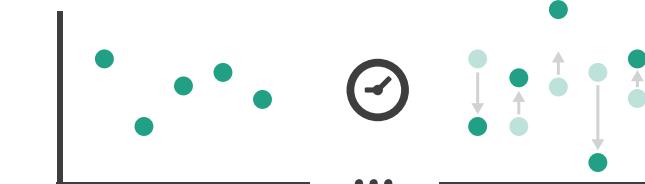


derive new data to show within the view

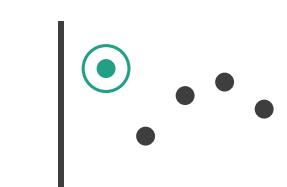
change view over time  
facet across multiple views

## Manipulate

### → Change



### → Select



### → Navigate

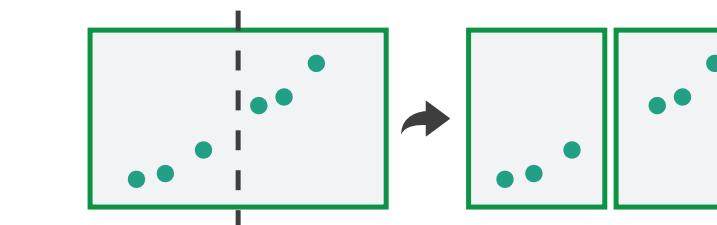


## Facet

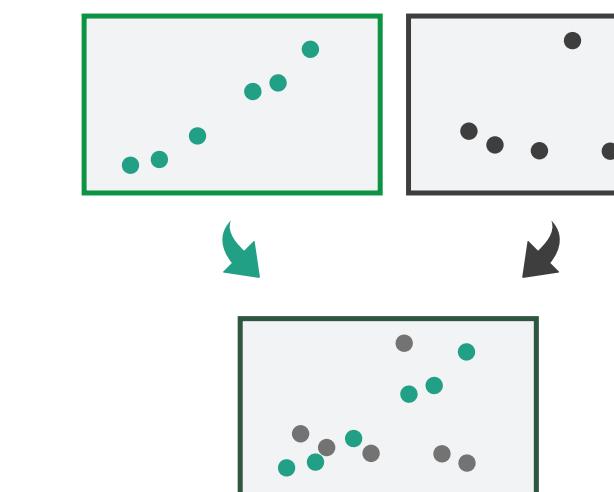
### → Juxtapose



### → Partition

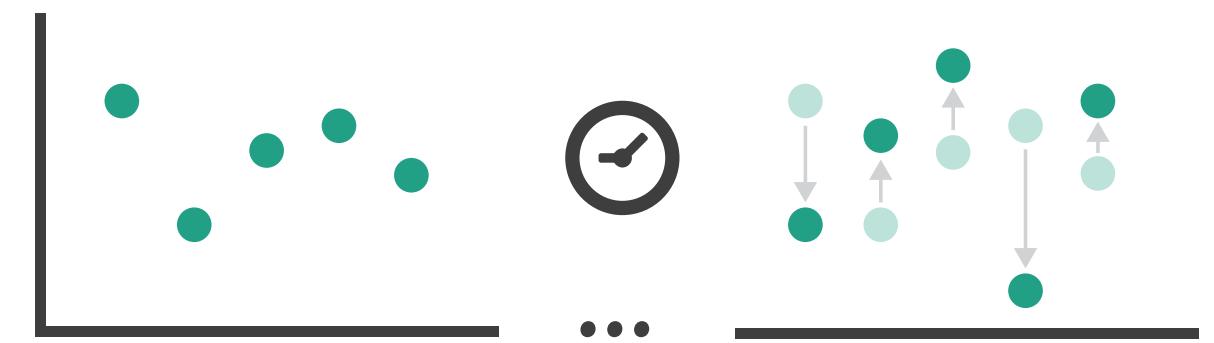


### → Superimpose



# Manipulate

## → Change over Time



# Change over time

- change any of the other choices
  - visual encoding itself
  - parameters
  - arrange: rearrange, reorder
  - aggregation level, what is filtered...
- interaction entails **change**
- **powerful & flexible**

# Idiom: Change parameters

- **widgets and controls**
  - sliders, buttons, radio buttons, checkboxes, dropdowns/comboboxes
- **pros**
  - clear affordances, self-documenting (with labels)
- **cons**
  - uses screen space
  - design choices
    - separated vs interleaved
      - controls & canvas

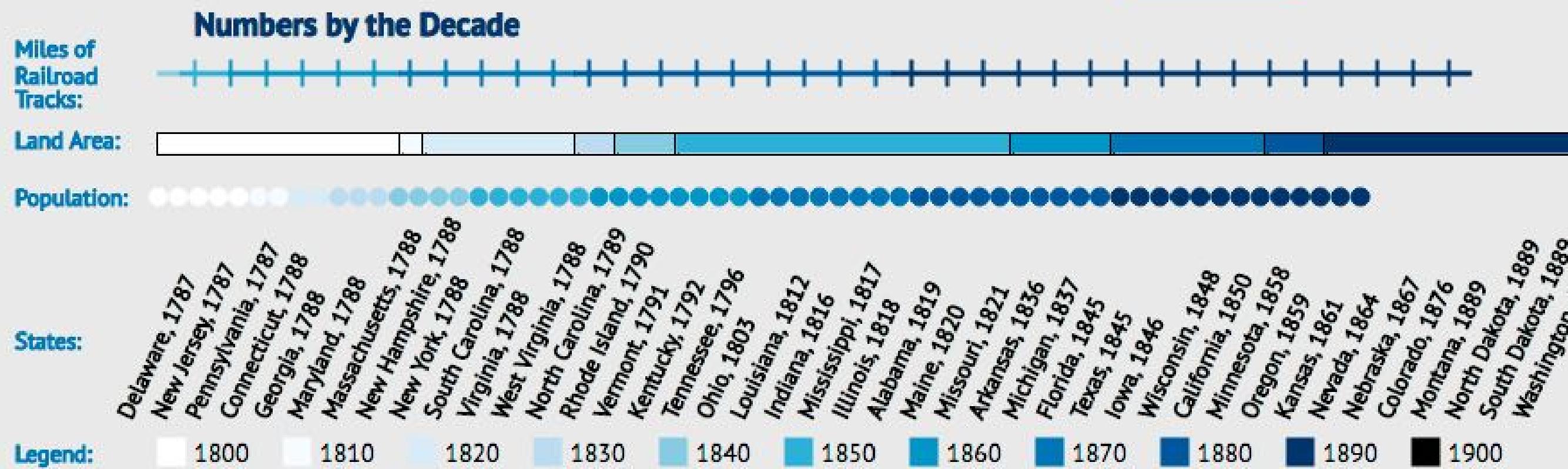
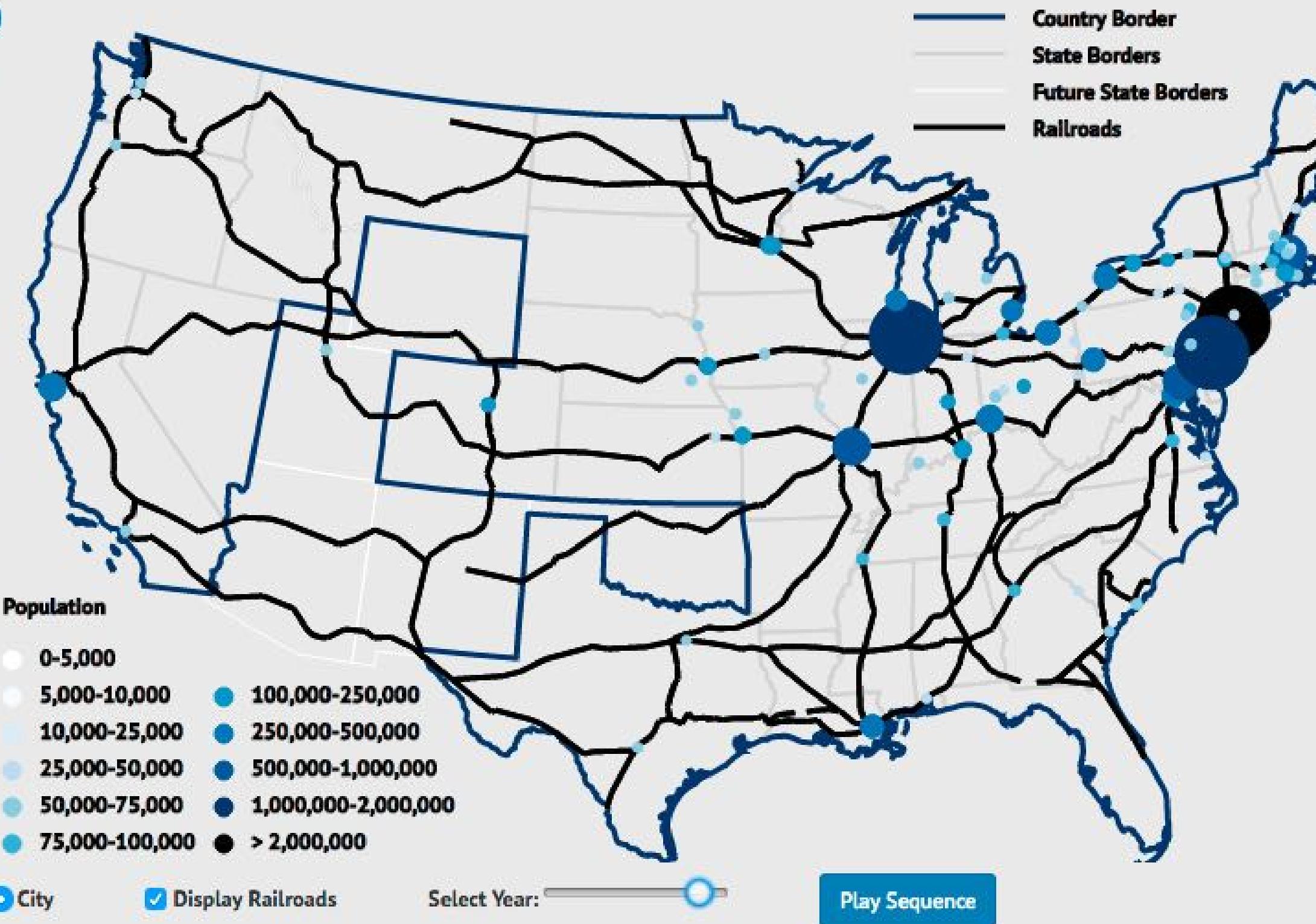
# The Growth of a Nation

## Or....how the railroads changed the face of America in the 1800's

The following visualization shows land, population and railroad growth in 19th Century America.

### The Year 1890

By 1890, there were more than 165,000 miles of railroad tracks across the US. The massive railroad corporations had paved the way for other corporations that grew out of the industrial revolution of that time such as the Oriental Telephone and Standard Oil Companies. New technologies such as the telephone, light bulb and motion pictures were invented and became more mainstream.



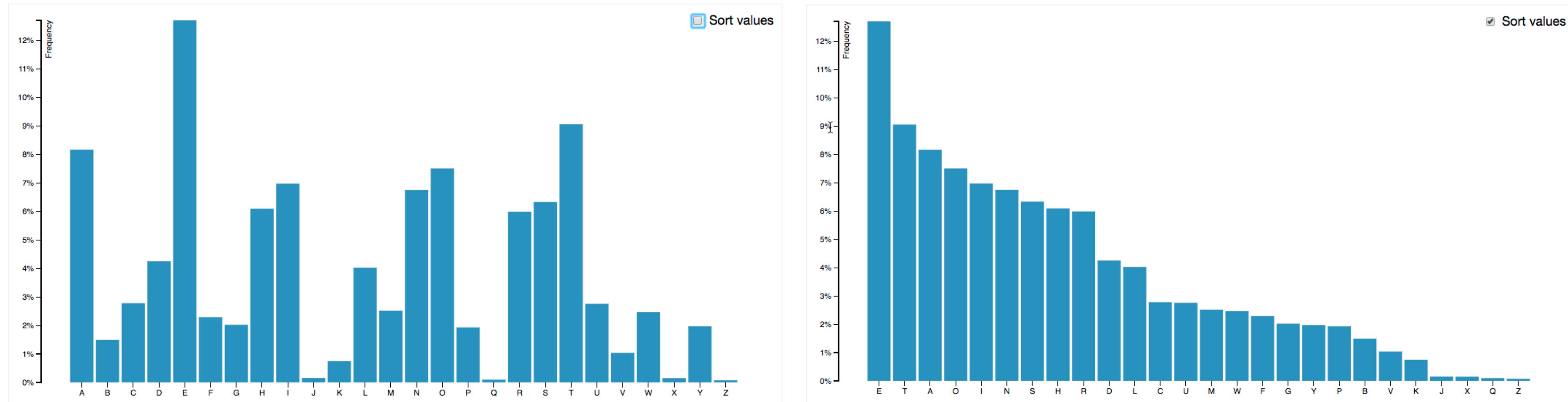
### Largest cities

New York, NY
Chicago, IL
Philadelphia, PA
Brooklyn, NY
St. Louis, MO
Boston, MA
Baltimore, MD
San Francisco, CA
Cincinnati, OH
Cleveland, OH
Buffalo, NY
New Orleans, LA
Pittsburgh, PA
Washington, VA
Detroit, MI
Milwaukee, WI
Newark, NJ
Minneapolis, MN
Jersey City, NJ
Louisville, KY
Omaha, NE
Rochester, NY
St. Paul, MN
Kansas City, MO
Providence, RI
Denver, CO
Indianapolis, IN
Allegheny, PA
Albany, NY
Columbus, OH
Syracuse, NY
New Haven, CT
Worcester, MA
Toledo, OH
Richmond, VA
Paterson, NJ
Lowell, MA
Nashville, TN
Scranton, PA
Fall River, MA
Cambridge, MA
Atlanta, GA
Memphis, TN
Wilmington, DE
Dayton, OH
Troy, NY
Grand Rapids, MI
Reading, PA
Camden, NJ
Trenton, NJ
Lynn, MA
Lincoln, NE
Charleston, SC
Hartford, CT
St. Joseph, MO
Evansville, IN
Los Angeles, CA
Des Moines, IA
Bridgeport, CT
Oakland, CA
Portland, OR
Saginaw, MI
Salt Lake City, UT
Lawrence, MA
Springfield, MA
Manchester, NH
Utica, NY
Hoboken, NJ
Savannah, GA
Seattle, WA
Peoria, IL
New Bedford, MA
Erie, PA
Somerville, MA
Harrisburg, PA
Kansas City, KS
Dallas, TX
Sioux City, IA
Elizabeth, NJ

Population State County

# Idiom: Change order/arrangement

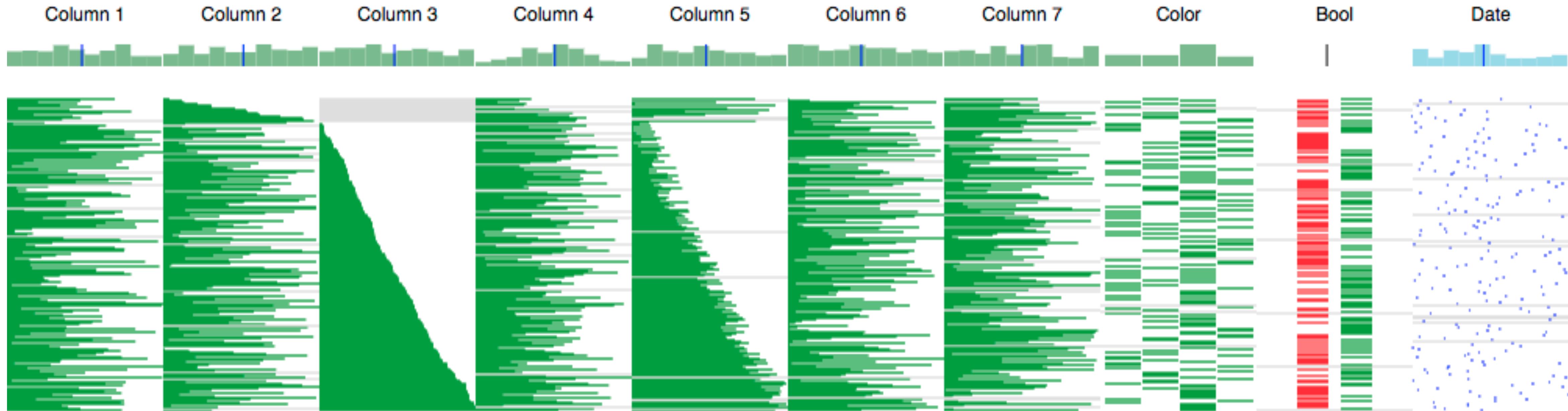
- what: simple table
- how: data-driven reordering
- why: find extreme values, trends



[Sortable Bar Chart] <https://observablehq.com/@d3/sortable-bar-chart>

# Idiom: Reorder

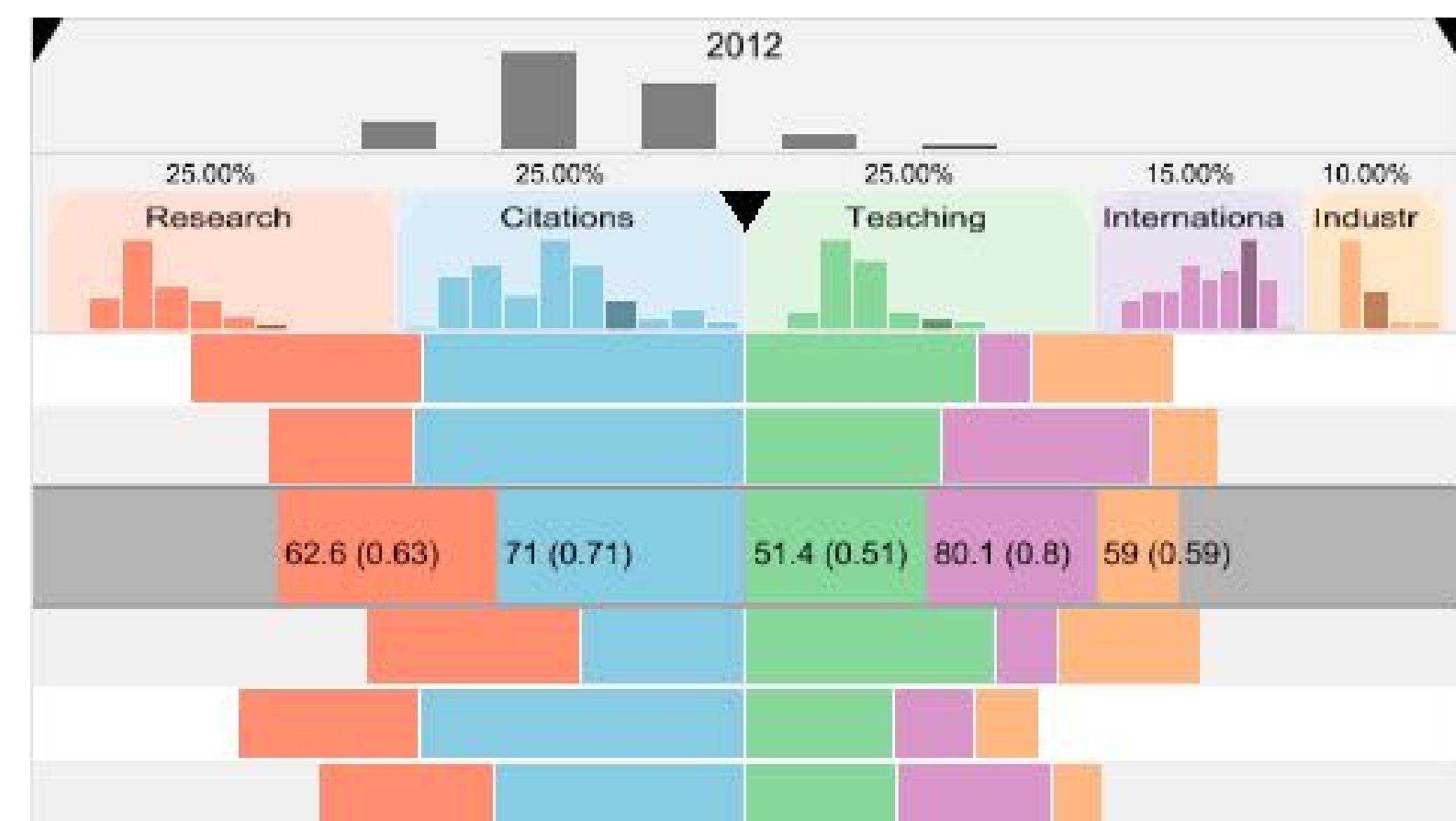
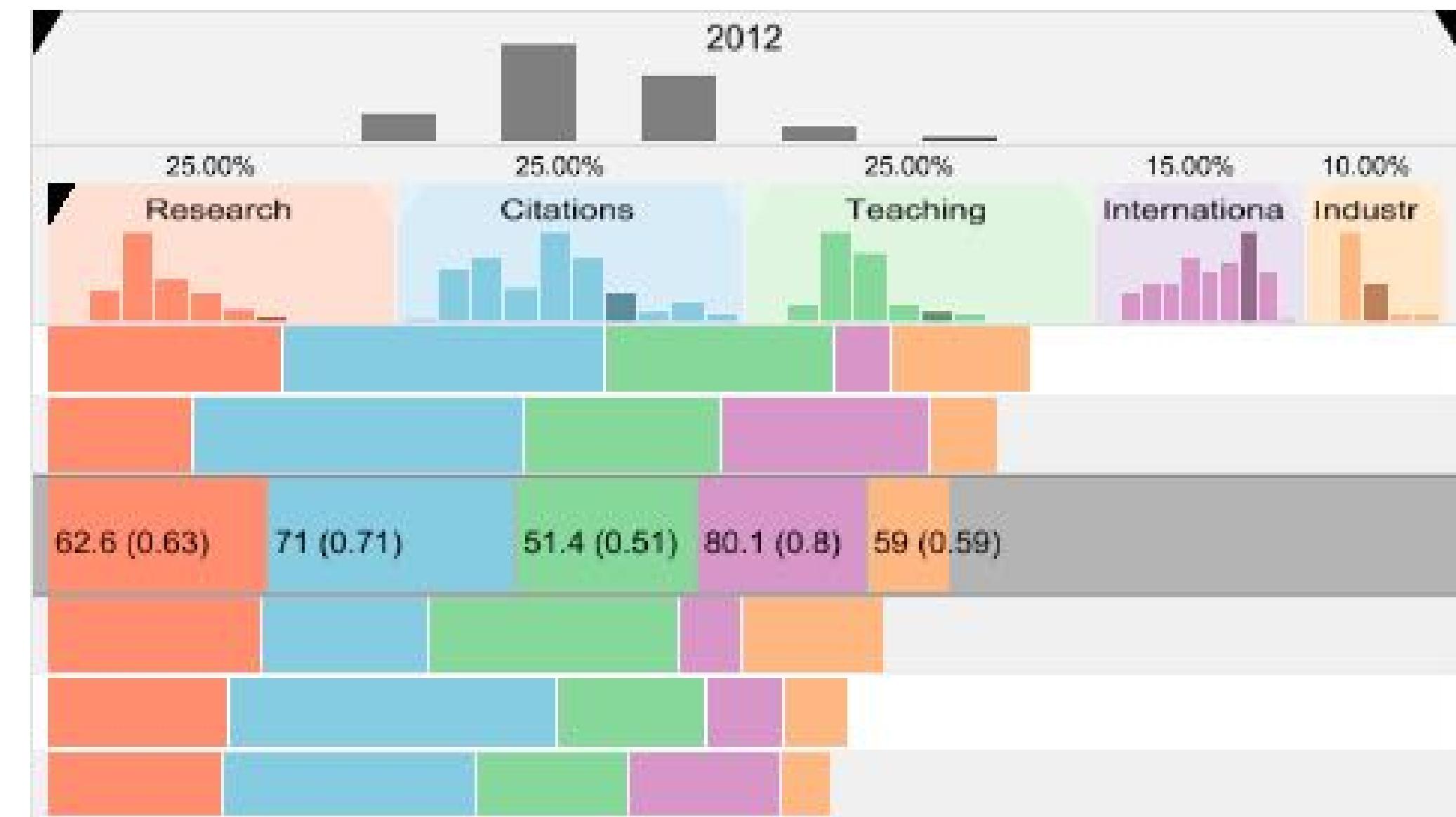
- what: table with many attributes
- how: data-driven reordering by selecting column
- why: find correlations between attributes



## System: LineUp

# Idiom: Change alignment

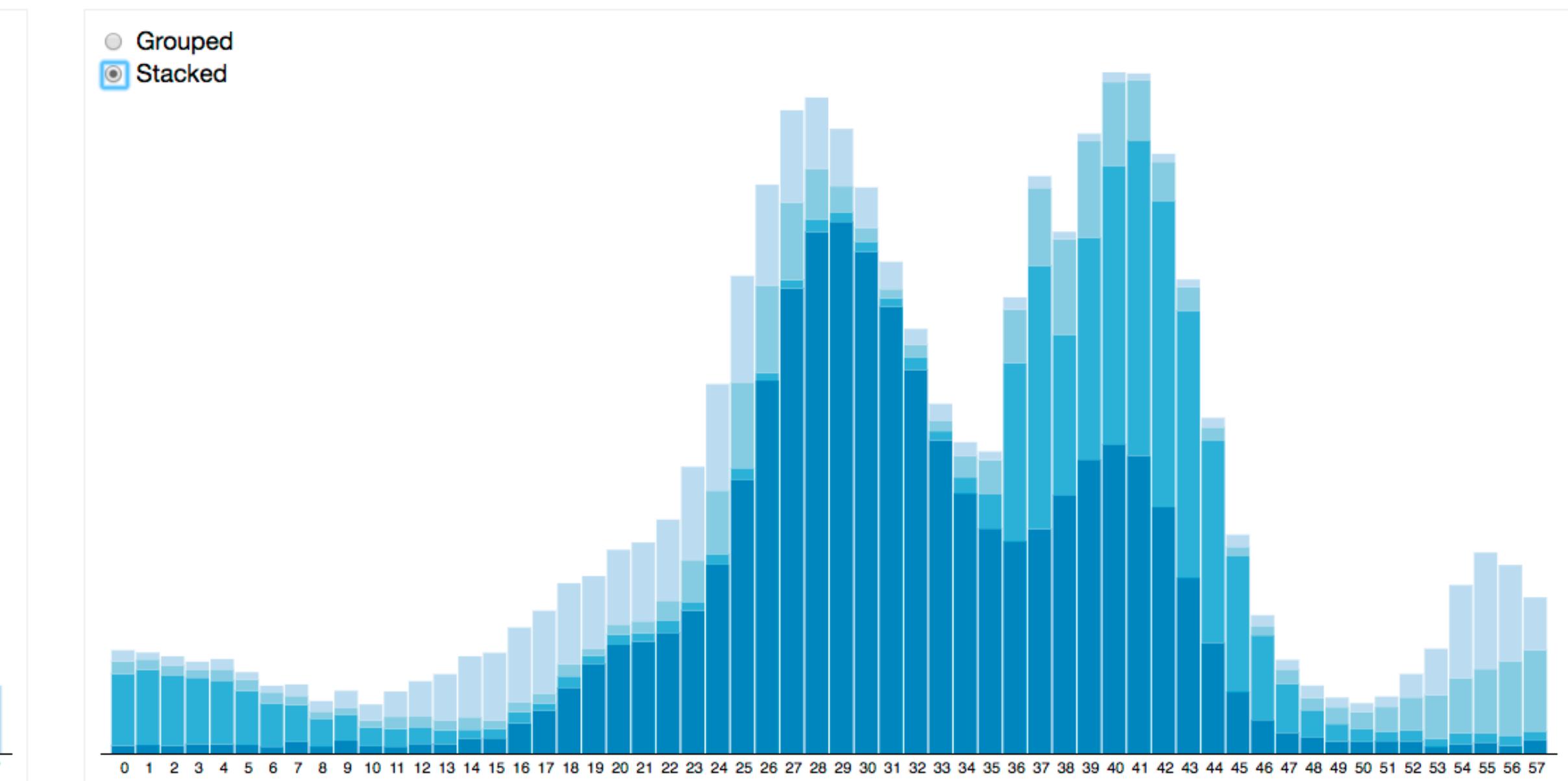
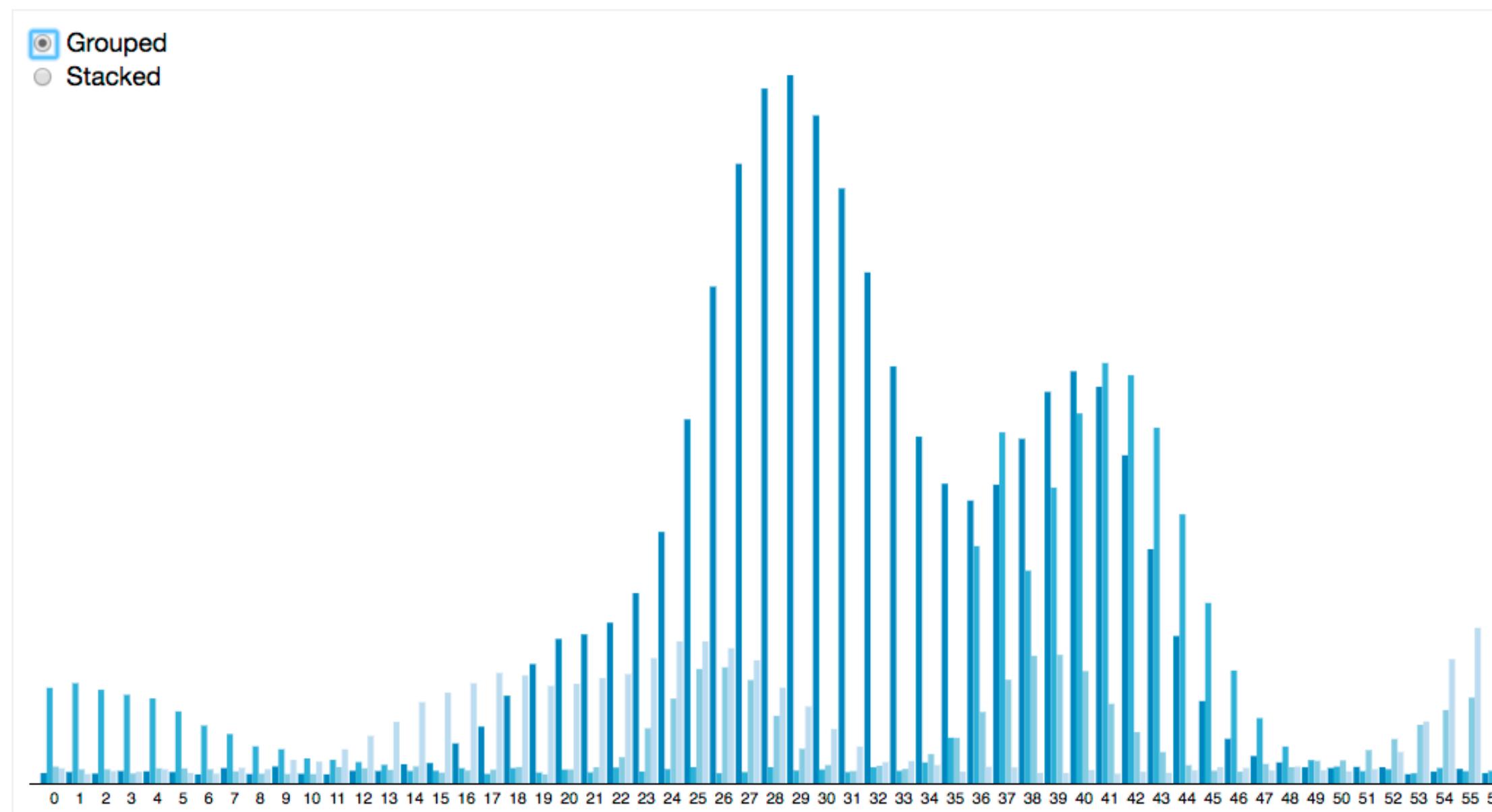
- stacked bars
  - easy to compare
    - first segment
    - total bar
  - align to different segment
    - supports flexible comparison



[LineUp: Visual Analysis of Multi-Attribute  
Rankings. Gratzl, Lex, Gehlenborg, Pfister, and Streit.  
IEEE Trans. Visualization and Computer Graphics]

# Idiom: Animated transitions - visual encoding change

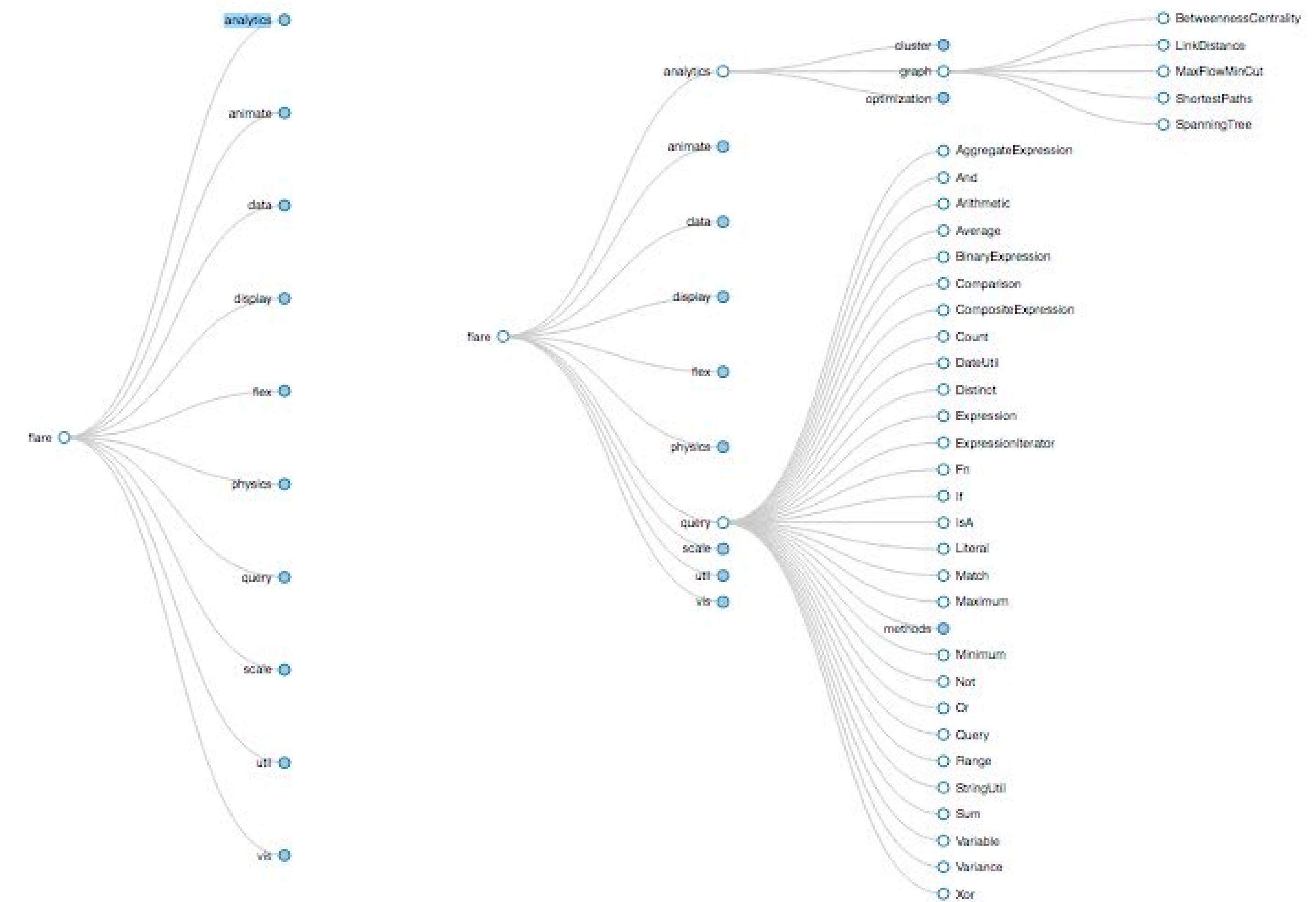
- smooth transition from one state to another
  - alternative to jump cuts, supports item tracking
    - best case for animation
  - staging to reduce cognitive load



[Stacked to Grouped Bars] <https://observablehq.com/@d3/stacked-to-grouped-bars>

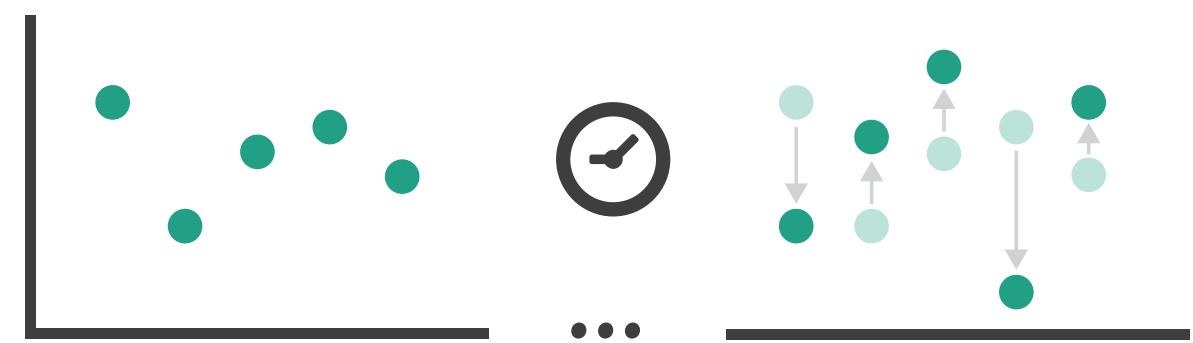
# Idiom: Animated transition - tree detail

- animated transition
  - network drilldown/rollup

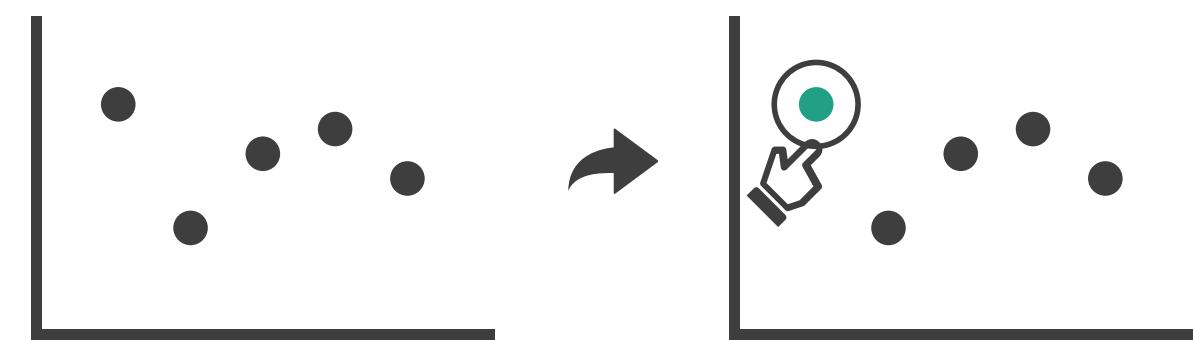


# Manipulate

## → Change over Time



## → Select



# Interaction technology

- what do you design for?
  - mouse & keyboard on desktop?
    - large screens, hover, multiple clicks
  - touch interaction on mobile?
    - small screens, no hover, just tap
  - gestures from video / sensors?
    - ergonomic reality vs movie bombast
  - eye tracking?

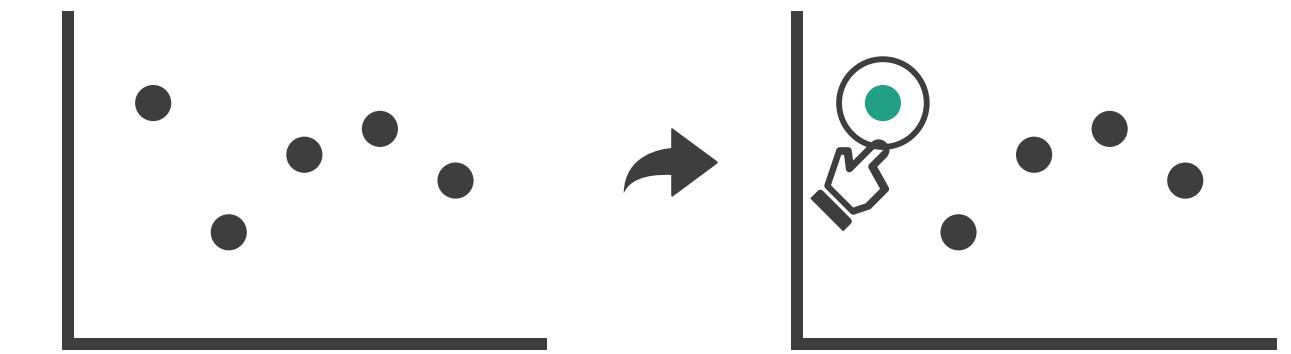


**Data visualization and the news - Gregor Aisch (37 min)**  
[vimeo.com/182590214](https://vimeo.com/182590214)



**I Hate Tom Cruise - Alex Kauffmann (5 min)**  
[www.youtube.com/watch?v=QXLfT9sFcbc](https://www.youtube.com/watch?v=QXLfT9sFcbc)

## → Select

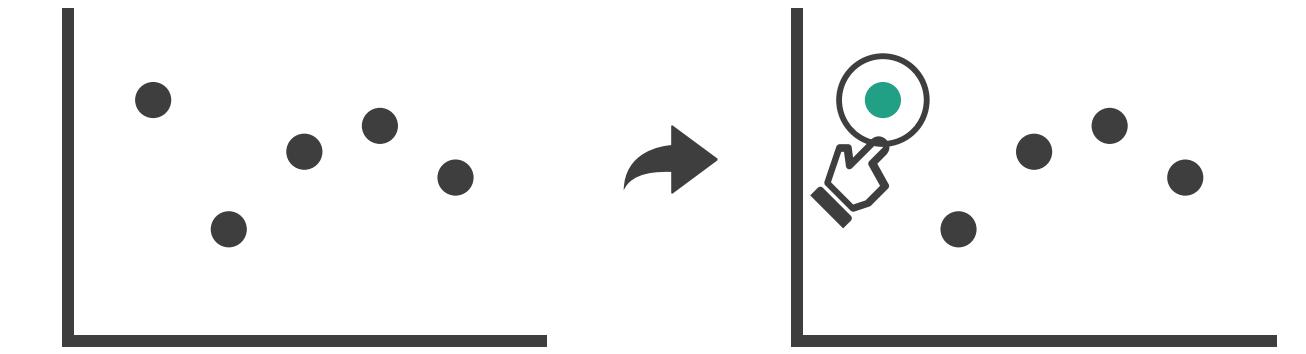


# Selection

- **selection**: basic operation for most interaction
- design choices: how many selection types?
- **interaction modalities**
  - **click/tap** (heavyweight) vs **hover** (lightweight but not available on most touchscreens)
  - **multiple click types** (shift-click, option-click, ...)
  - **proximity** beyond click/hover (touching vs nearby vs distant)
- **application semantics**
  - **adding** to selection set vs **replacing** selection
  - can selection be **null**?
    - ex: toggle so nothing selected if click on background
  - **primary** vs **secondary** (ex: source/target nodes in network)
  - **group membership** (add/delete items, name group, ...)

→ Select

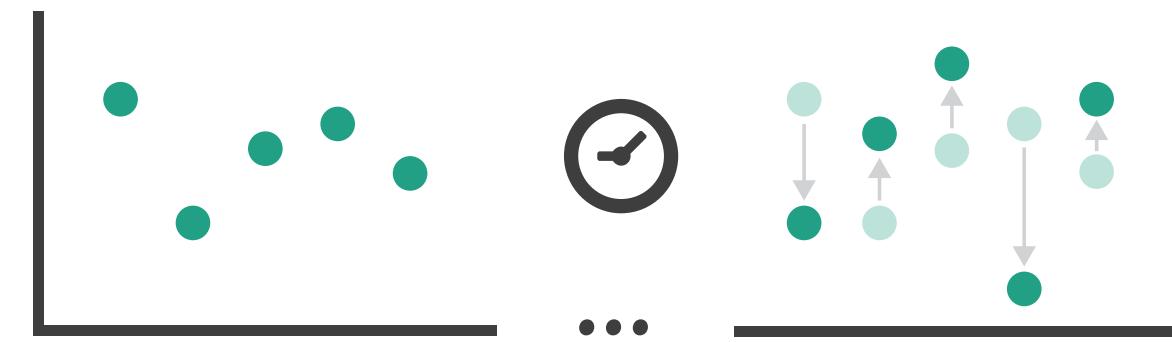
# Highlighting



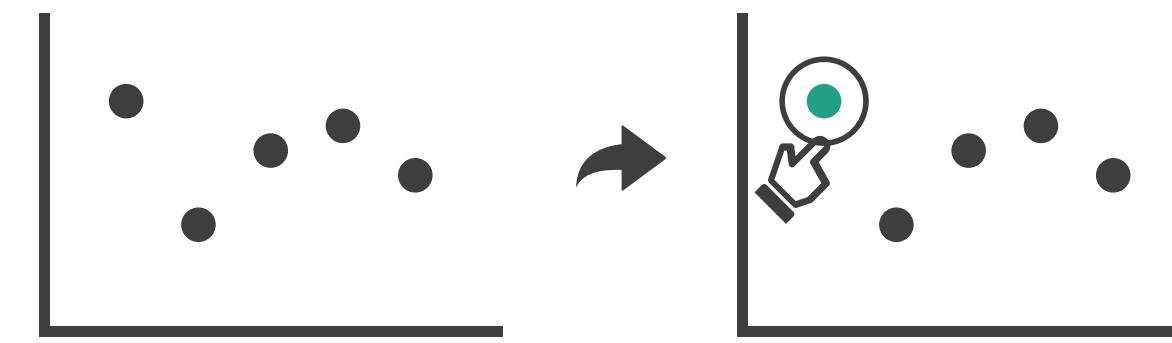
- **highlight**: change visual encoding for selection targets
  - visual feedback closely tied to but separable from selection (interaction)
- design choices: **typical visual channels**
  - change **item color**
    - but hides existing color coding
  - add **outline mark**
  - **change size** (ex: increase outline mark linewidth)
  - **change shape** (ex: from solid to dashed line for link mark)
- unusual channels: motion
  - motion: usually avoid for single view
    - with multiple views, could justify to draw attention to other views

# Manipulate

## → Change over Time



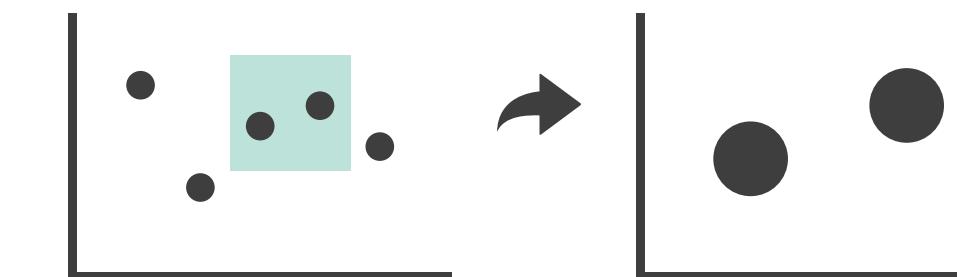
## → Select



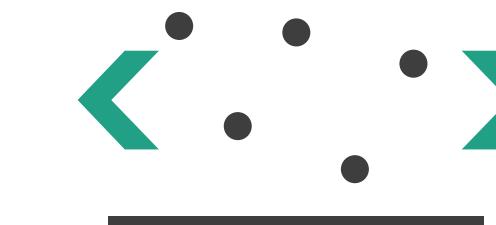
## → Navigate

### → Item Reduction

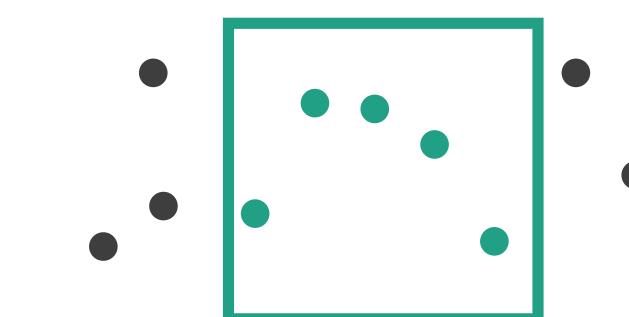
→ Zoom  
Geometric or Semantic



### → Pan/Translate



### → Constrained



# Navigate: Changing viewpoint/visibility

- **change viewpoint**

- changes which items are visible within view

- **camera metaphor**

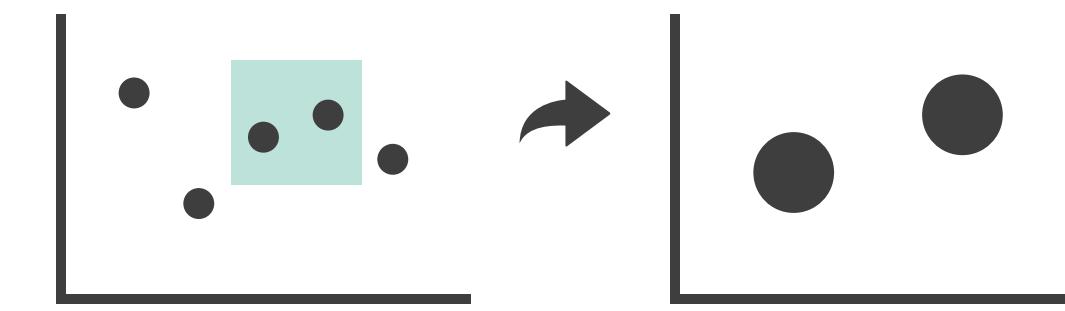
- pan/translate/scroll
  - move up/down/sideways

→ Navigate

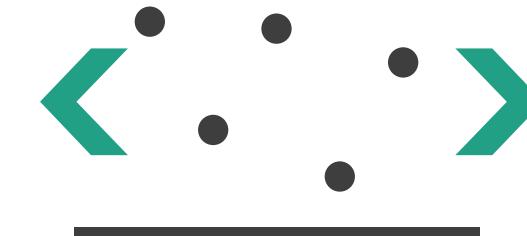
→ Item Reduction

→ Zoom

*Geometric* or *Semantic*

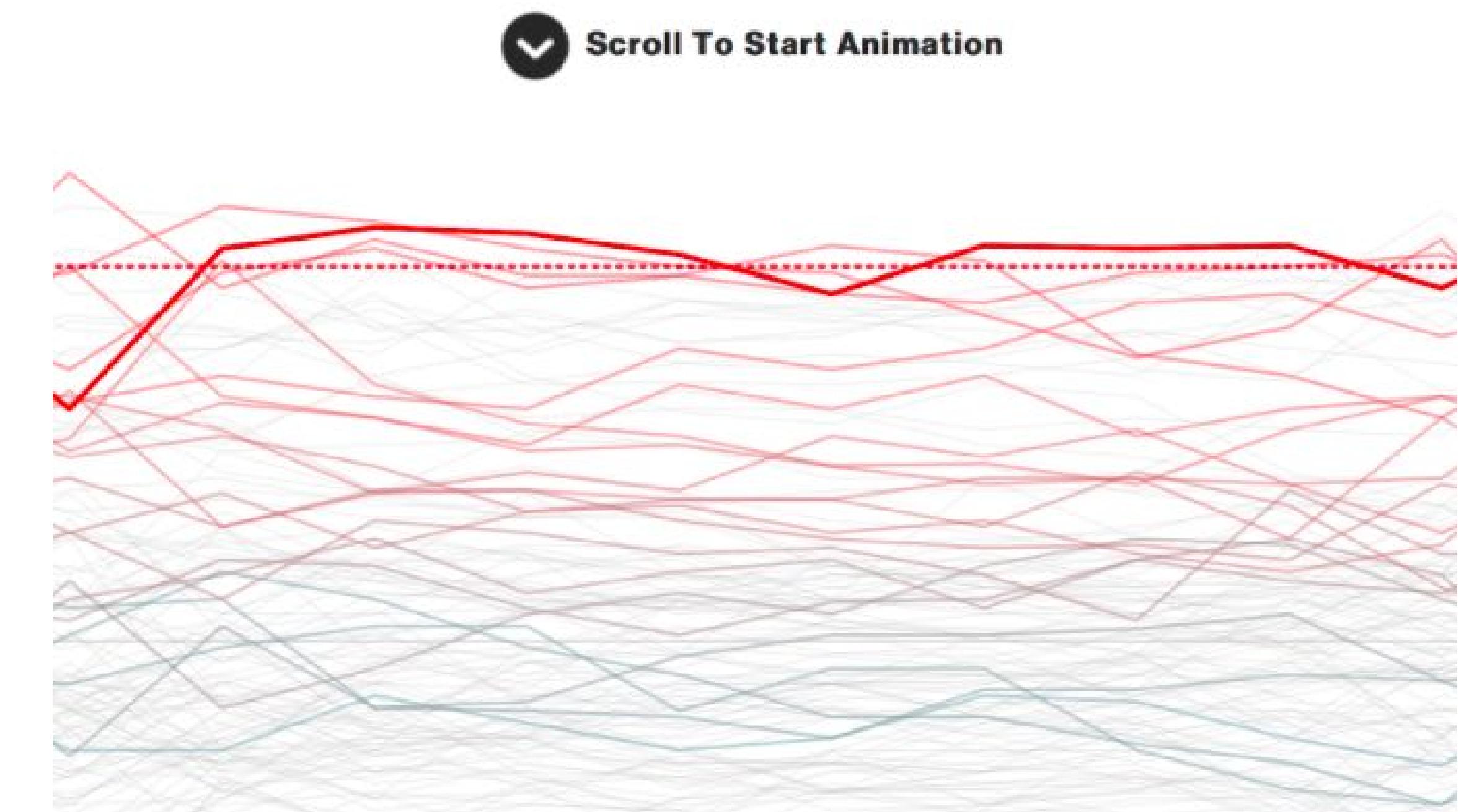


→ Pan/Translate



# Idiom: Scrollytelling

- **how:** navigate page by scrolling  
(panning down)
- **pros:**
  - familiar & intuitive, from standard web browsing
  - linear (only up & down) vs possible overload of click-based interface choices
- **cons:**
  - full-screen mode may lack affordances
  - **scrolljacking**, no direct access
  - unexpected behaviour
  - continuous control for discrete steps



[How to Scroll, Bostock](<https://bost.ocks.org/mike/scroll/>)  
<https://eagereyes.org/blog/2016/the-scrollytelling-scourge>  
<https://www.vev.design/blog/scrollytelling-examples/>

# Navigate: Changing viewpoint/visibility

- **change viewpoint**

- changes which items are visible within view

- **camera metaphor**

- pan/translate/scroll
  - move up/down/sideways

- **rotate/spin**

- typically in 3D

- **zoom in/out**

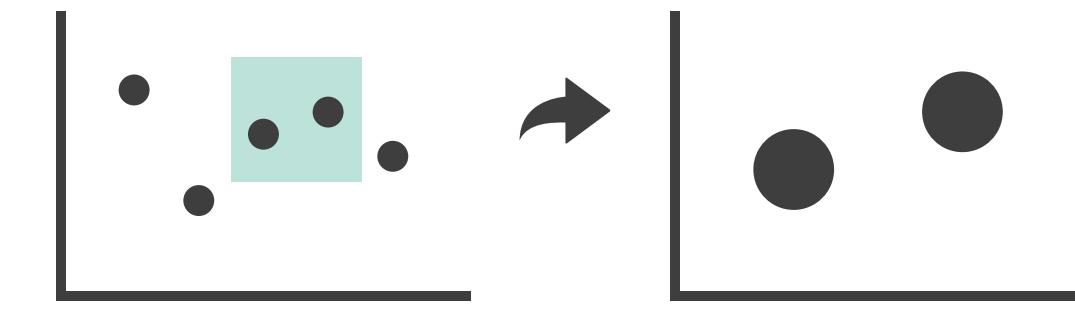
- enlarge/shrink world == move camera closer/further
- geometric zoom: standard, like moving physical object

→ Navigate

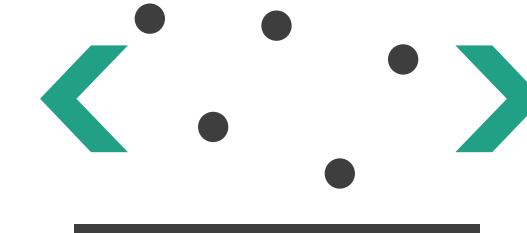
→ Item Reduction

→ Zoom

*Geometric* or *Semantic*



→ Pan/Translate



# Navigate: Unconstrained vs constrained

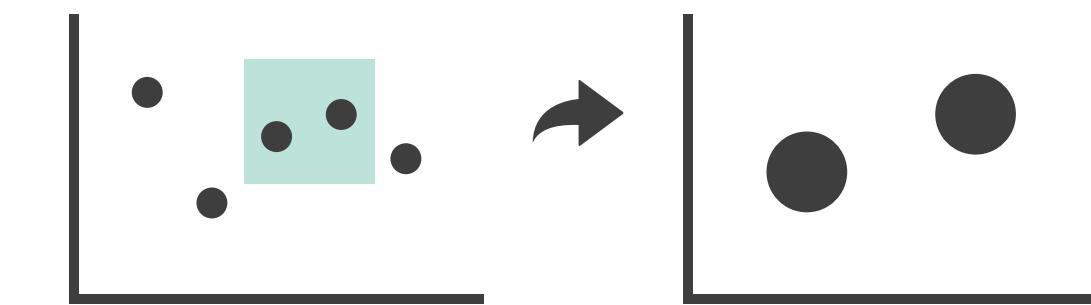
- **unconstrained navigation**
  - easy to implement for designer
  - hard to control for user
    - easy to overshoot/undershoot
- **constrained navigation**
  - typically uses animated transitions
  - trajectory automatically computed based on selection
    - just click; selection ends up framed nicely in final viewport

→ Navigate

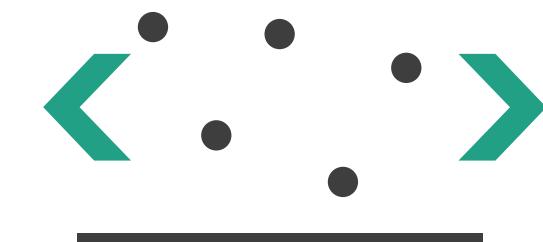
→ Item Reduction

→ Zoom

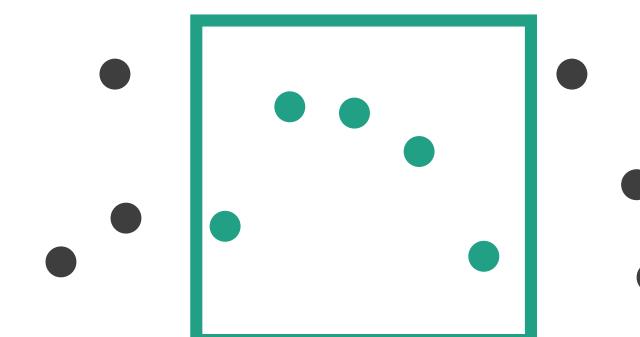
*Geometric* or *Semantic*



→ Pan/Translate



→ Constrained



# Idiom: Animated transition + constrained navigation

- example: **geographic map**
  - simple zoom, only viewport changes, shapes preserved



[g Box] <https://observablehq.com/@d3/zoom-to-bounding-box>

# Interaction benefits

- **interaction pros**
  - major advantage of **computer-based** vs paper-based visualization
  - **flexible, powerful, intuitive**
    - **exploratory data analysis**: change as you go during analysis process
    - **fluid task switching**: different visual encodings support different tasks
  - **animated transitions** provide excellent support
    - empirical evidence that animated transitions help people stay oriented

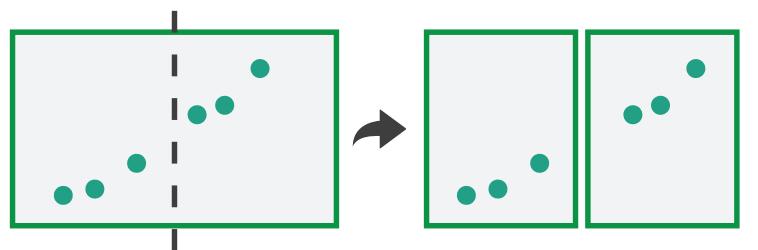
# Interaction limitations

- interaction has a **time cost**
  - sometimes minor, sometimes significant
  - degenerates to human-powered search in worst case
- **remembering** previous state imposes **cognitive load**
- controls may **take screen real estate**
  - or invisible functionality may be difficult to discover (lack of affordances)
- users may **not interact as planned** by designer
  - NYTimes logs show ~90% don't interact beyond scrollytelling - Aisch, 2016

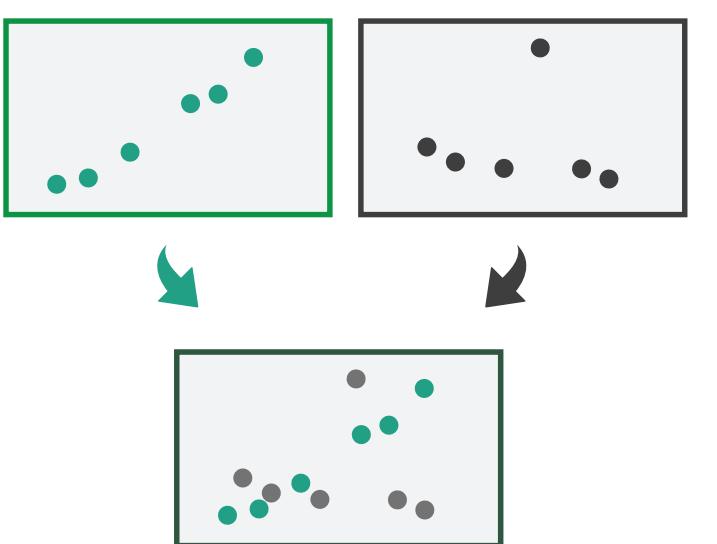
# Facet (multiple views)



→ **Partition**



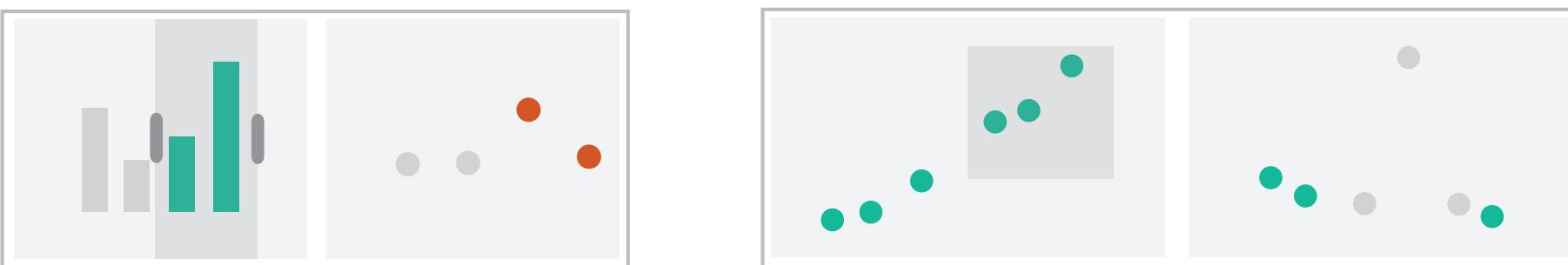
→ **Superimpose**



# Juxtapose and coordinate views

→ Share Encoding: Same/Different

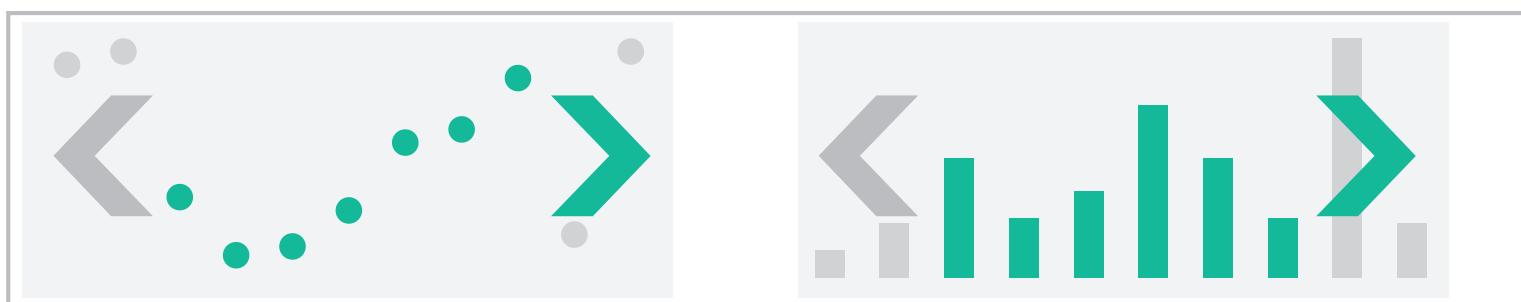
→ *Linked Highlighting*



→ Share Data: All/Subset/None

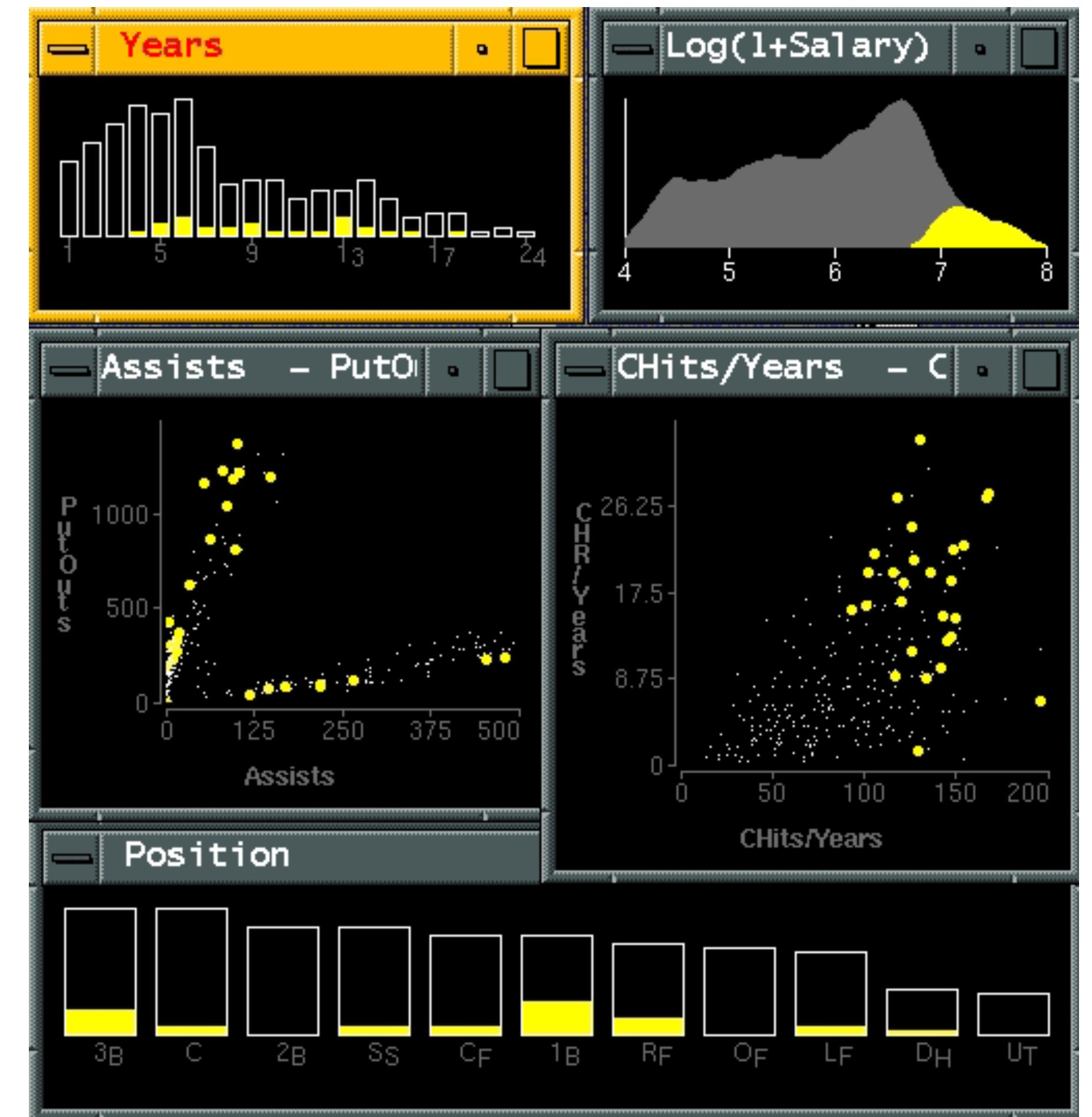


→ Share Navigation



# Idiom: Linked highlighting

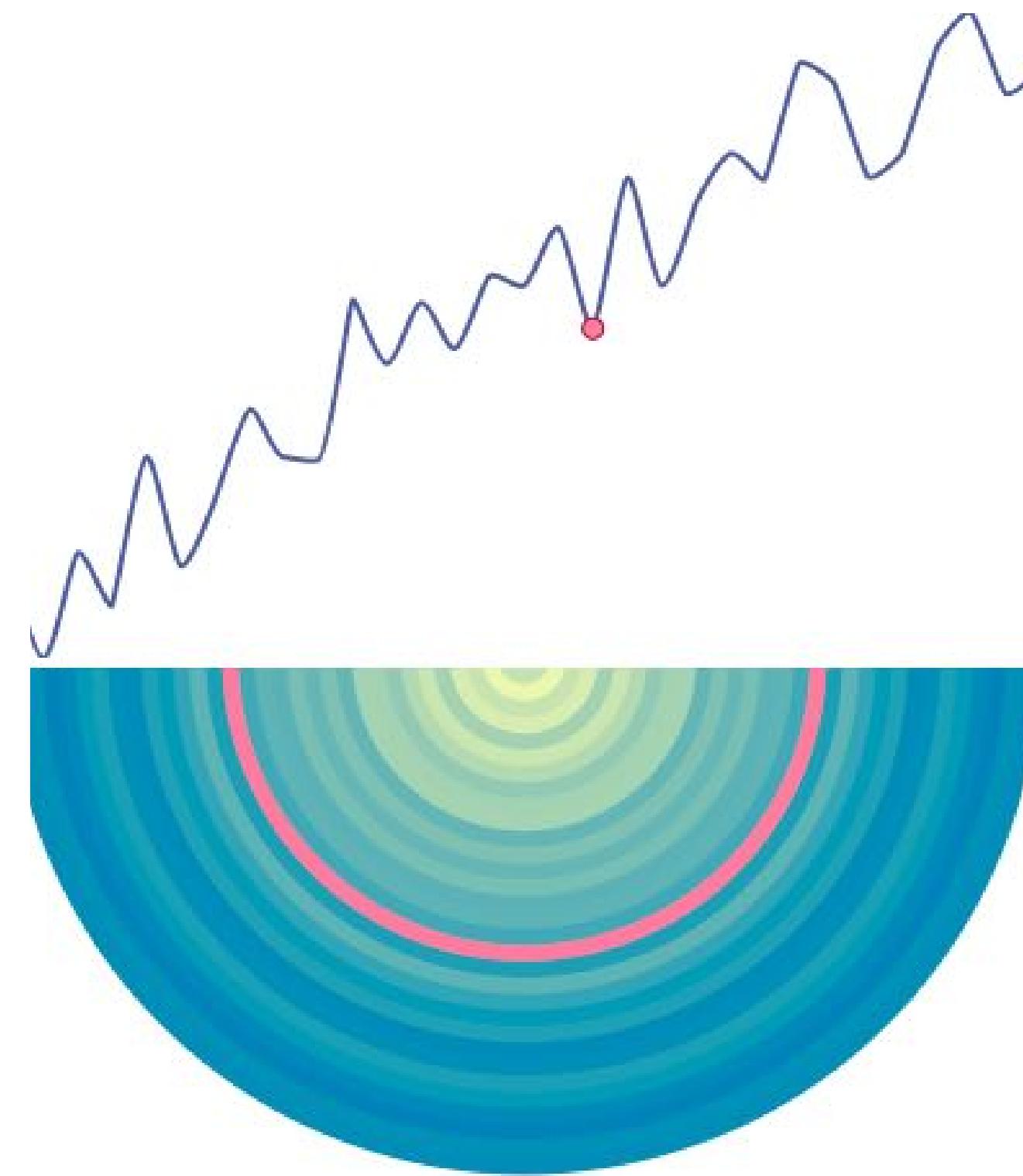
- see how regions contiguous in one view are distributed within another
  - powerful and pervasive interaction idiom
- encoding: different
  - multiform
- data: all shared
  - all items shared
  - different attributes across the views
- aka: **brushing and linking**



[Visual Exploration of Large Structured Datasets. Wills.  
Proc. New Techniques and Trends in Statistics (NTTS)]

# Linked views: Directionality

- unidirectional vs bidirectional linking
  - bidirectional almost always better!



<http://pbeshai.github.io/linked-highlighting-react-vega-redux/>

# Idiom: Overview-detail views

- encoding: **same or different**
  - ex: same (birds-eye map)
- data: subset shared
  - viewpoint differences:  
subset of data items
- navigation: shared
  - **bidirectional** linking
- other differences
  - (window size)

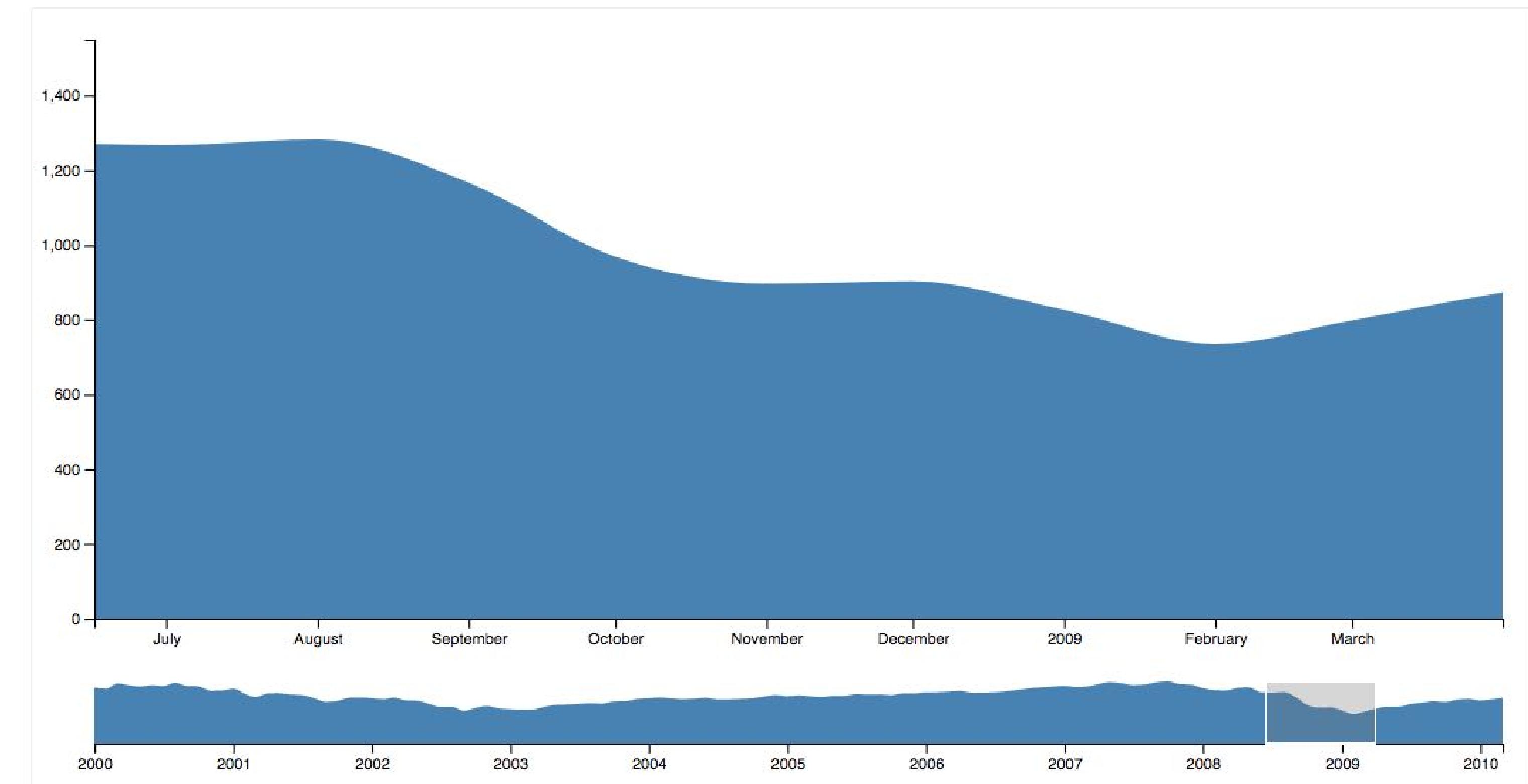
System: Google Maps



[A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. Cockburn, Karlson, and Bederson. ACM Computing Surveys 41:1 (2008), 1–31.]

# Idiom: Overview-detail navigation

- encoding: **same or different**
- data: subset shared
- navigation: **shared**
  - **unidirectional linking**
  - select in small overview, change extent in large detail view



<https://observablehq.com/@uwdata/interaction>

# Idiom: Tooltips

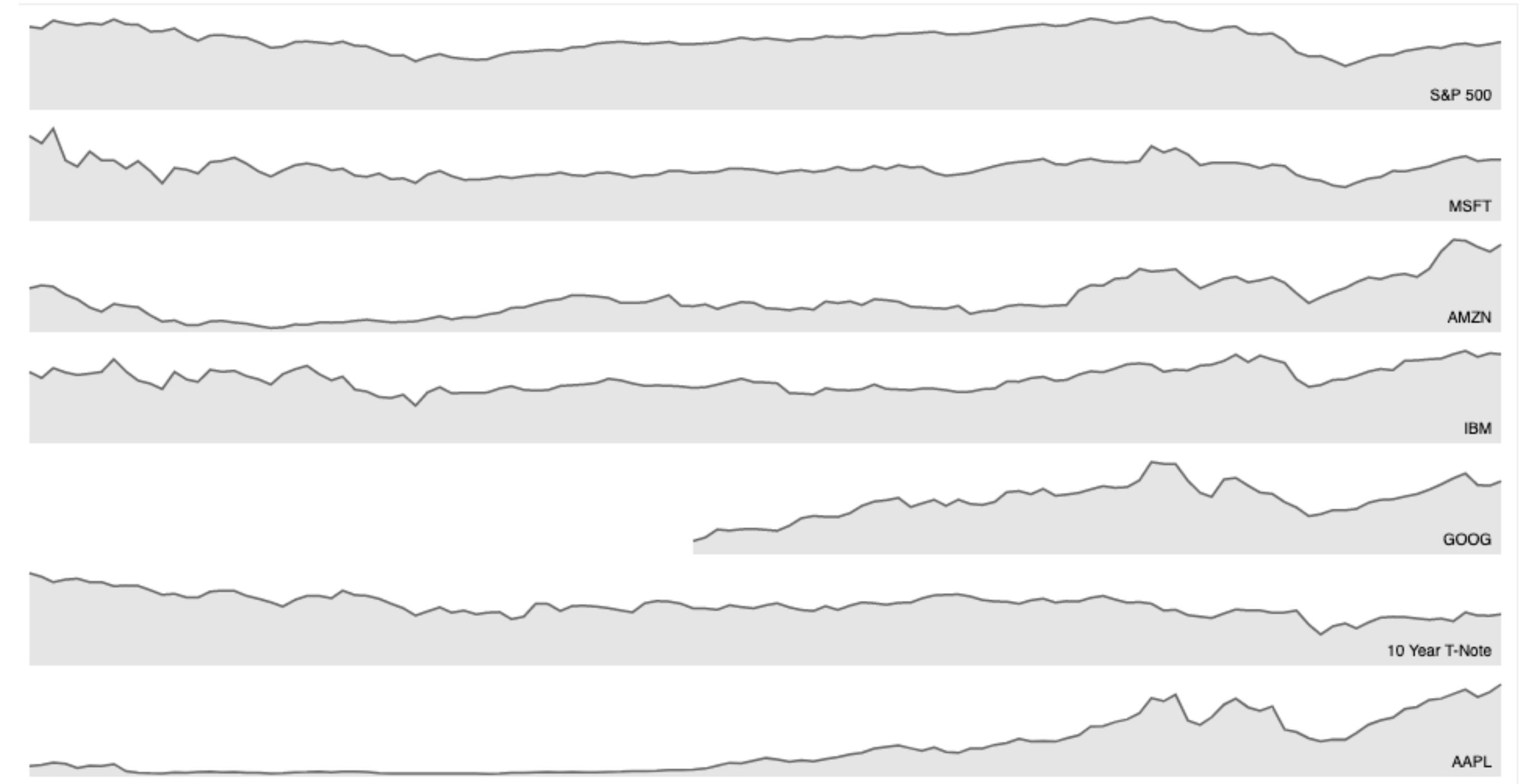
- **popup information for selection**
  - **hover or click**
  - specific case of detail view:  
provide useful additional detail on demand
  - beware: does not support overview!
    - always consider if there's a way to visually encode directly to provide overview
    - “If you make a rollover or tooltip, assume nobody will see it. If it's important, make it explicit.”
      - Gregor Aisch, NYTimes



[<https://www.highcharts.com/demo/dynamic-master-detail>]

# Idiom: Small multiples

- encoding: same
  - ex: line charts
- data: none shared
  - different slices of dataset
    - items or attributes
    - ex: stock prices for different companies



<https://observablehq.com/@nyuviz/small-multiples>

[https://projects.flowingdata.com/tut/linked\\_small\\_multiples\\_demo/](https://projects.flowingdata.com/tut/linked_small_multiples_demo/)

# Interactive small multiples

- **linked highlighting:** analogous item/attribute across views
  - same year highlighted across all charts if hover within any chart

## The Rise and Decline of Ask MetaFilter

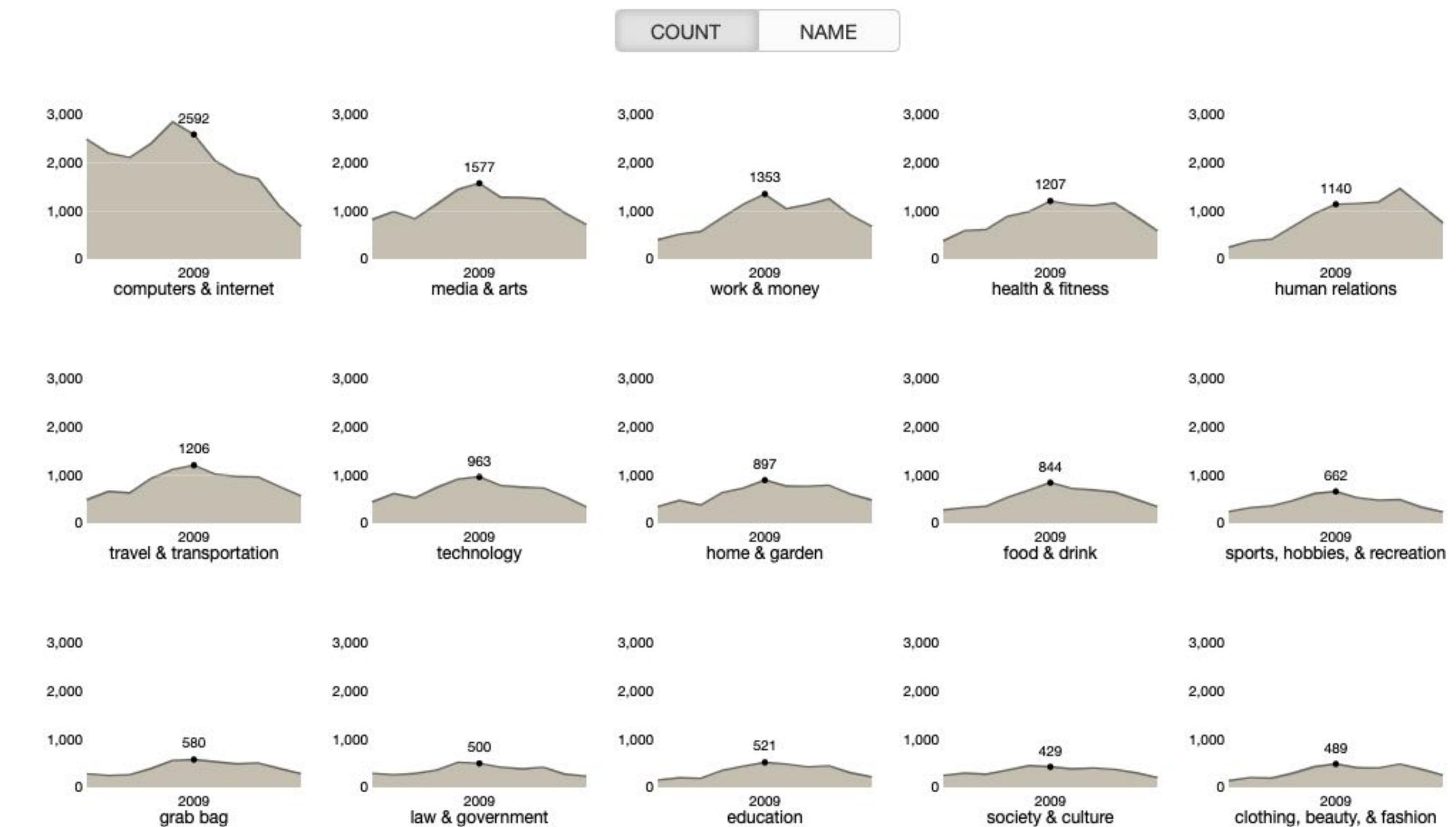
Metafilter's revenue has been on the decline, but has its content dried up as well?

Here we look at new posts on Ask Metafilter by category.

Categories like **computers & internet** have been dropping in use for a long time, most likely due to competition like Stack Overflow.

Other smaller categories have had consistent use patterns until more recently.

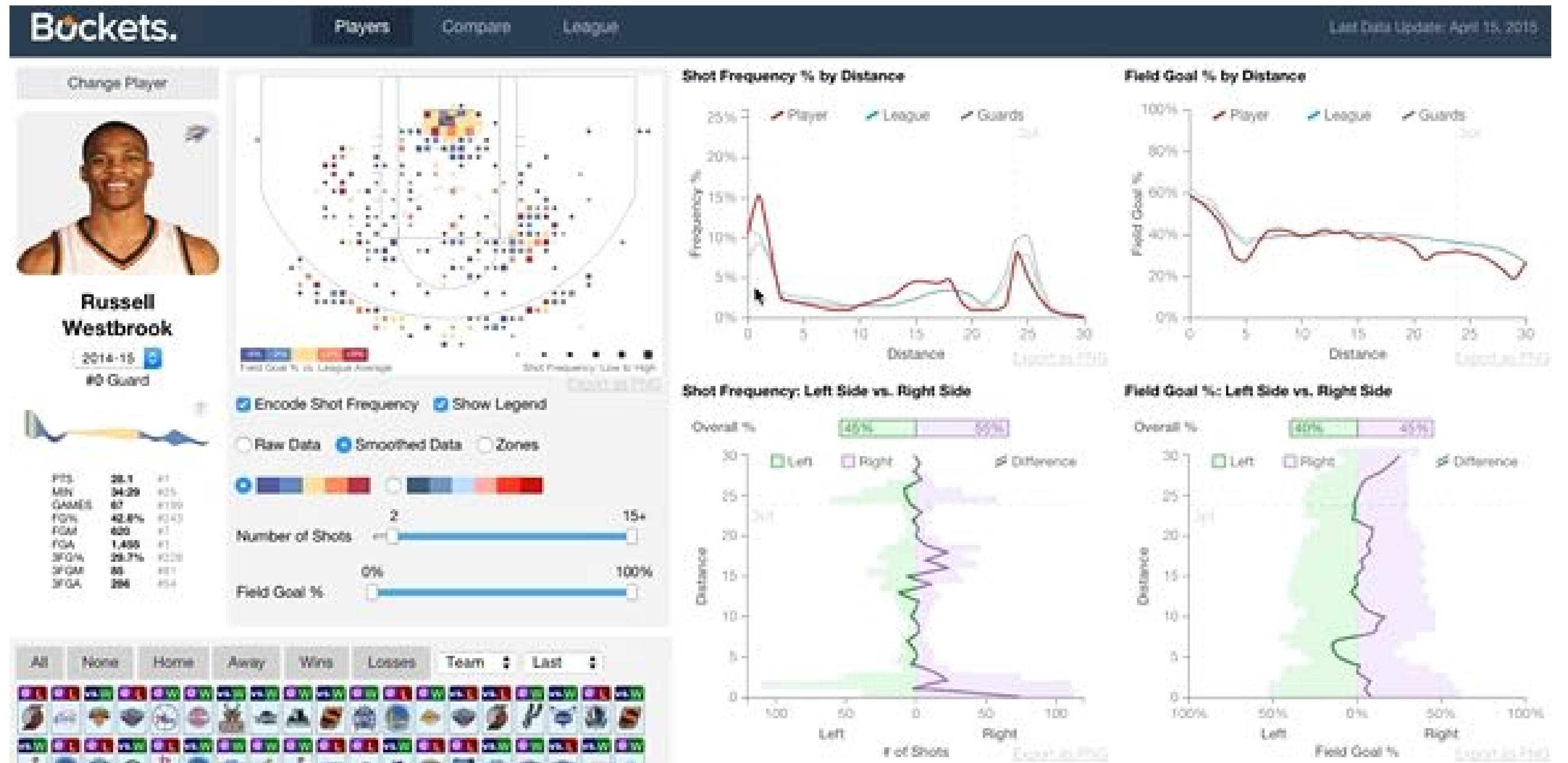
Disclaimer: 2014 is included, even though the year is not over yet.



[<https://blog.scottlogic.com/2017/04/05/interactive-responsive-small-multiples.html>]

[[http://projects.flowingdata.com/tut/linked\\_small\\_multiples\\_demo/](http://projects.flowingdata.com/tut/linked_small_multiples_demo/)]

# Example: Combining many interaction idioms

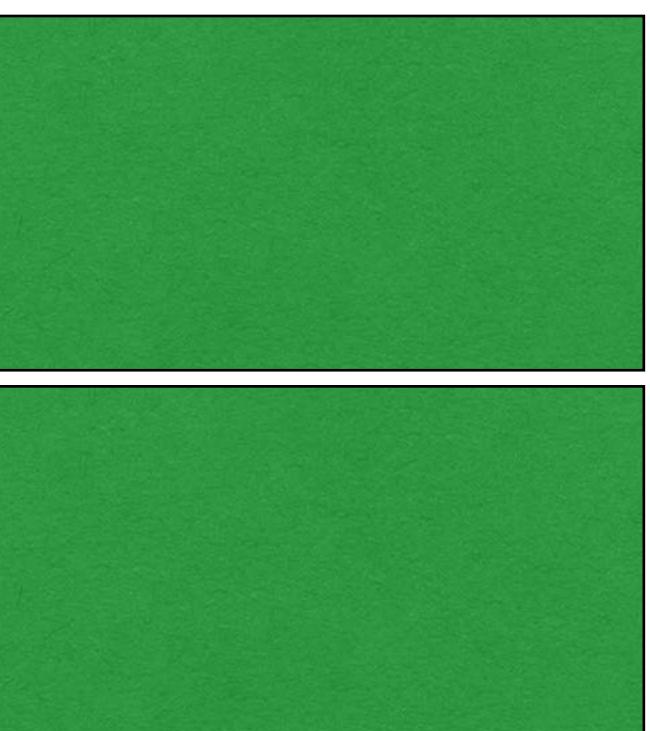
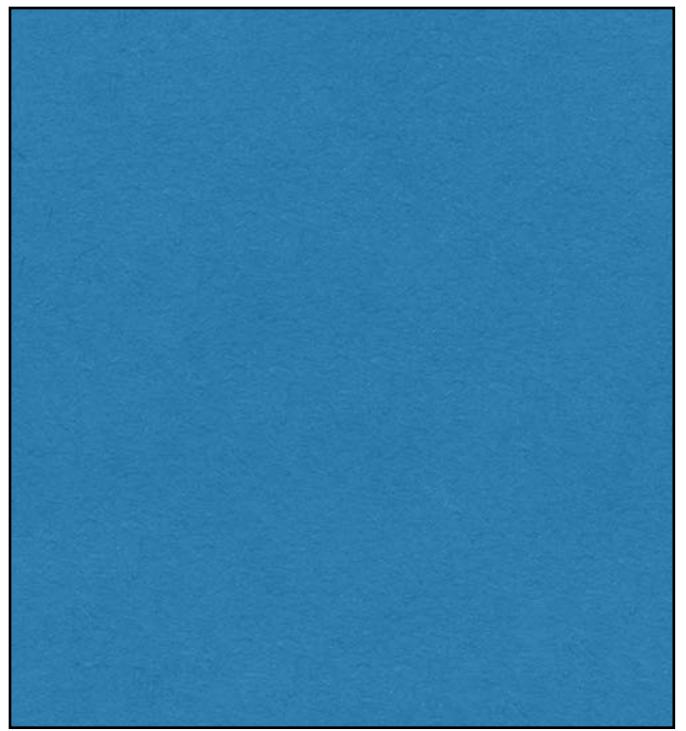


- multiform
- multidirectional linked highlighting of small multiples
- tooltips

<http://buckets.peterbeshai.com/>

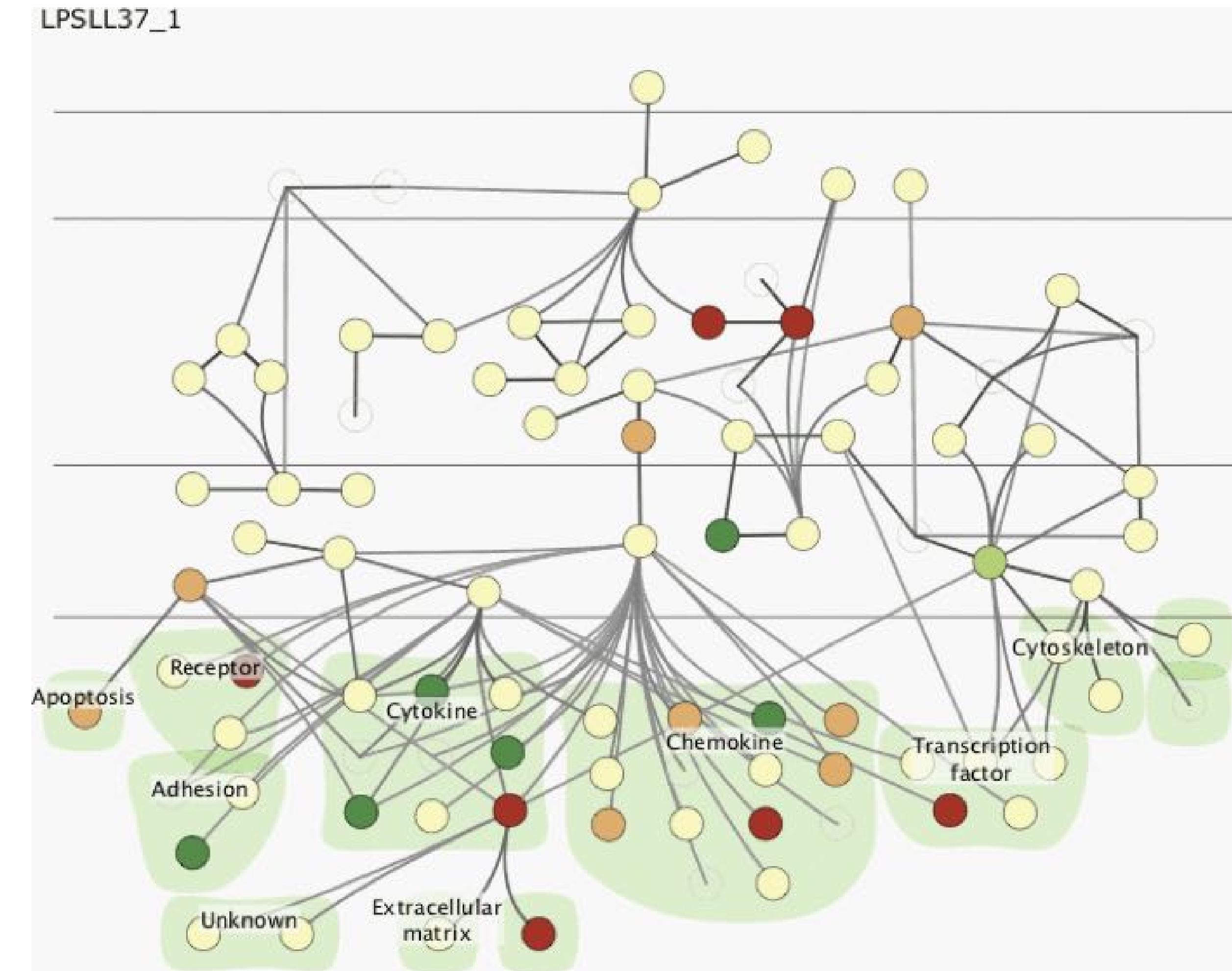
# Juxtapose views: tradeoffs

- juxtapose costs
  - display area
    - 2 views side by side: each has only half the area of one view
- juxtapose benefits
  - cognitive load: eyes vs memory
    - lower cognitive load: move eyes between 2 views
    - higher cognitive load: compare single changing view to memory of previous state



# Juxtapose vs animate

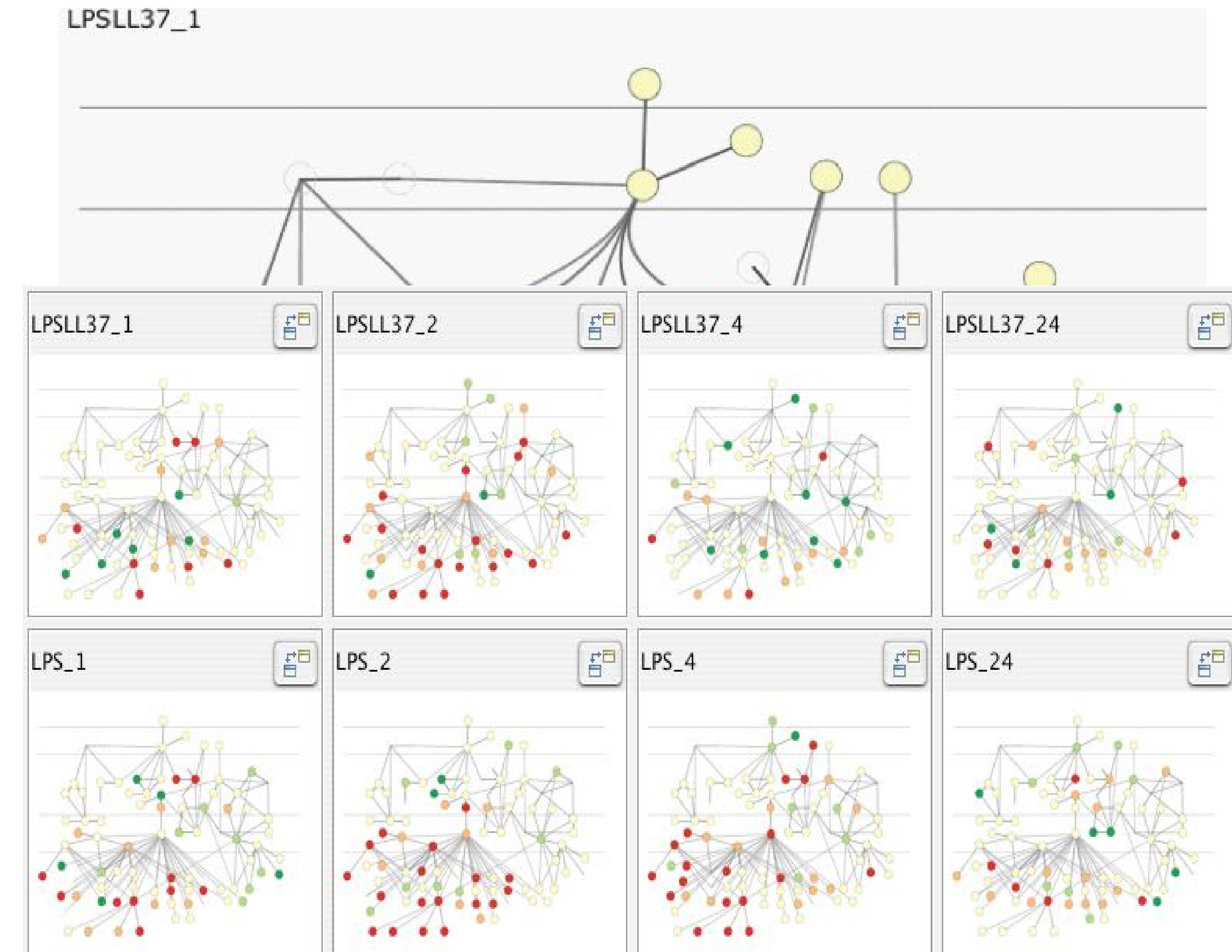
- **animate:** hard to follow if many scattered changes or many frames
- vs **easy special case:** animated transitions



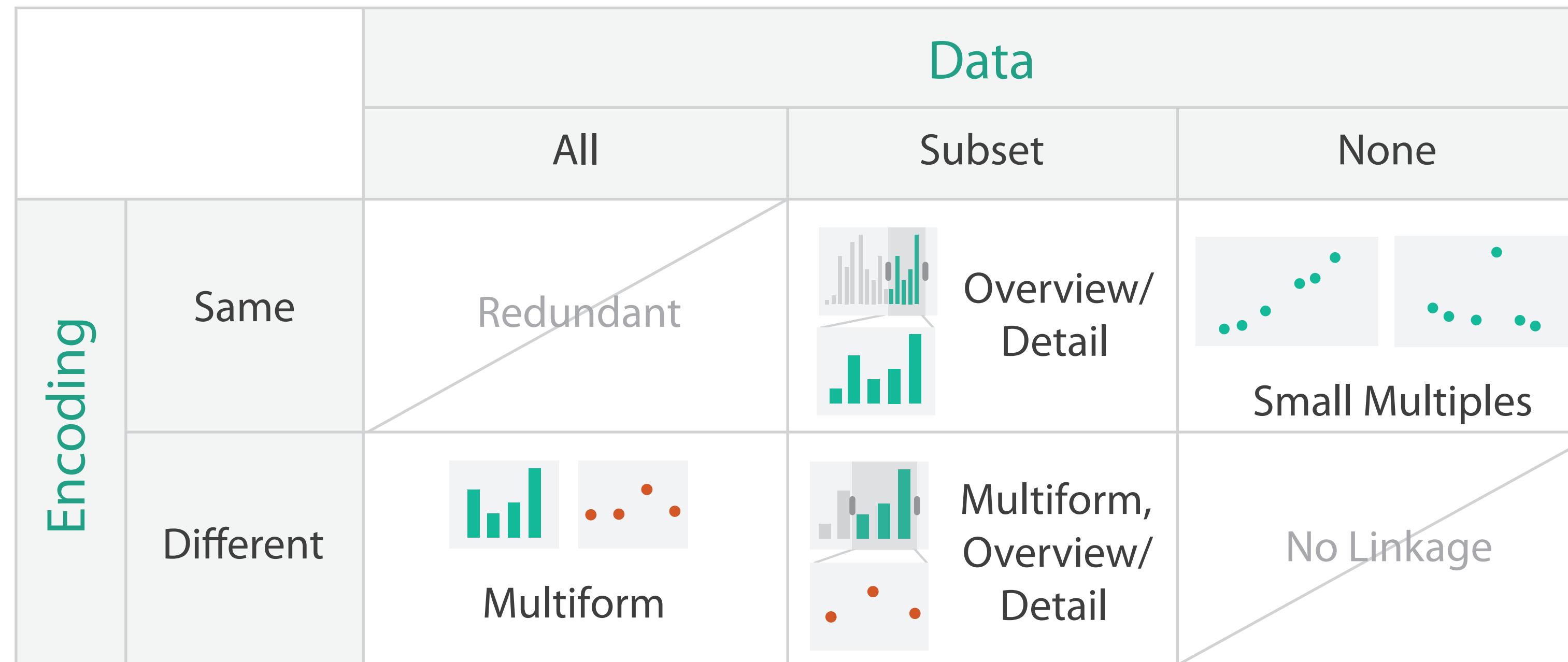
Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context. Barsky, Munzner, Gardy, and Kincaid. IEEE Trans. Visualization and Computer Graphics

# Juxtapose vs animate

- animate: hard to follow if many scattered changes or many frames
  - vs easy special case: animated transitions
- juxtapose: easier to compare across small multiples
  - different conditions (color), same gene (layout)



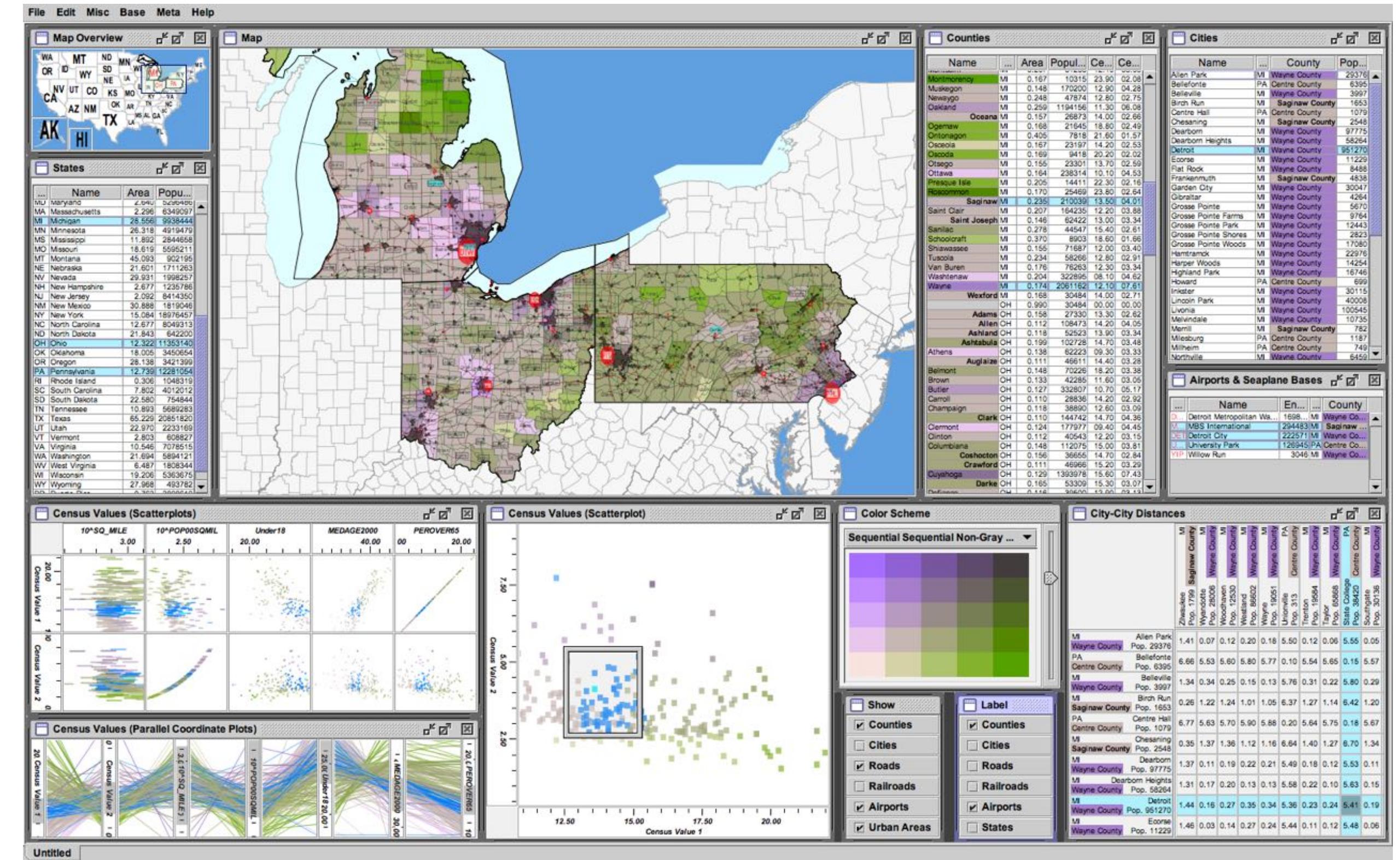
# View coordination: Design choices



# Idiom: Reorderable lists

- list views
  - easy lookup
  - useful when linked to other views
- how many views is ok vs too complex?
  - open research question

System: Improvise

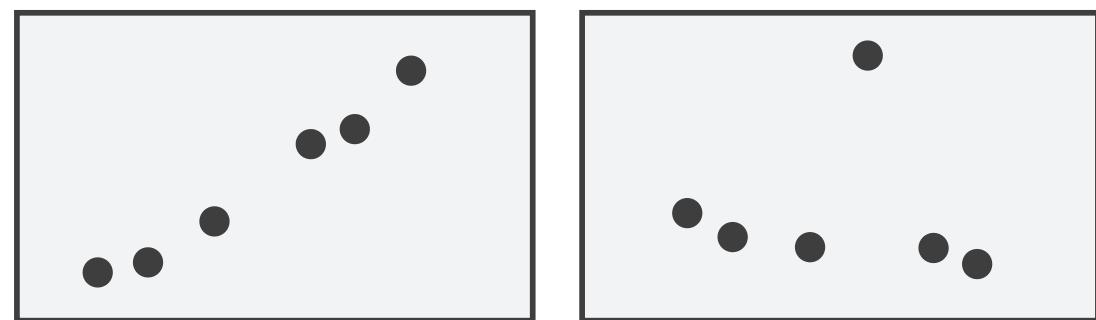


[Building Highly-Coordinated Visualizations In Improvise]

# Partition into views

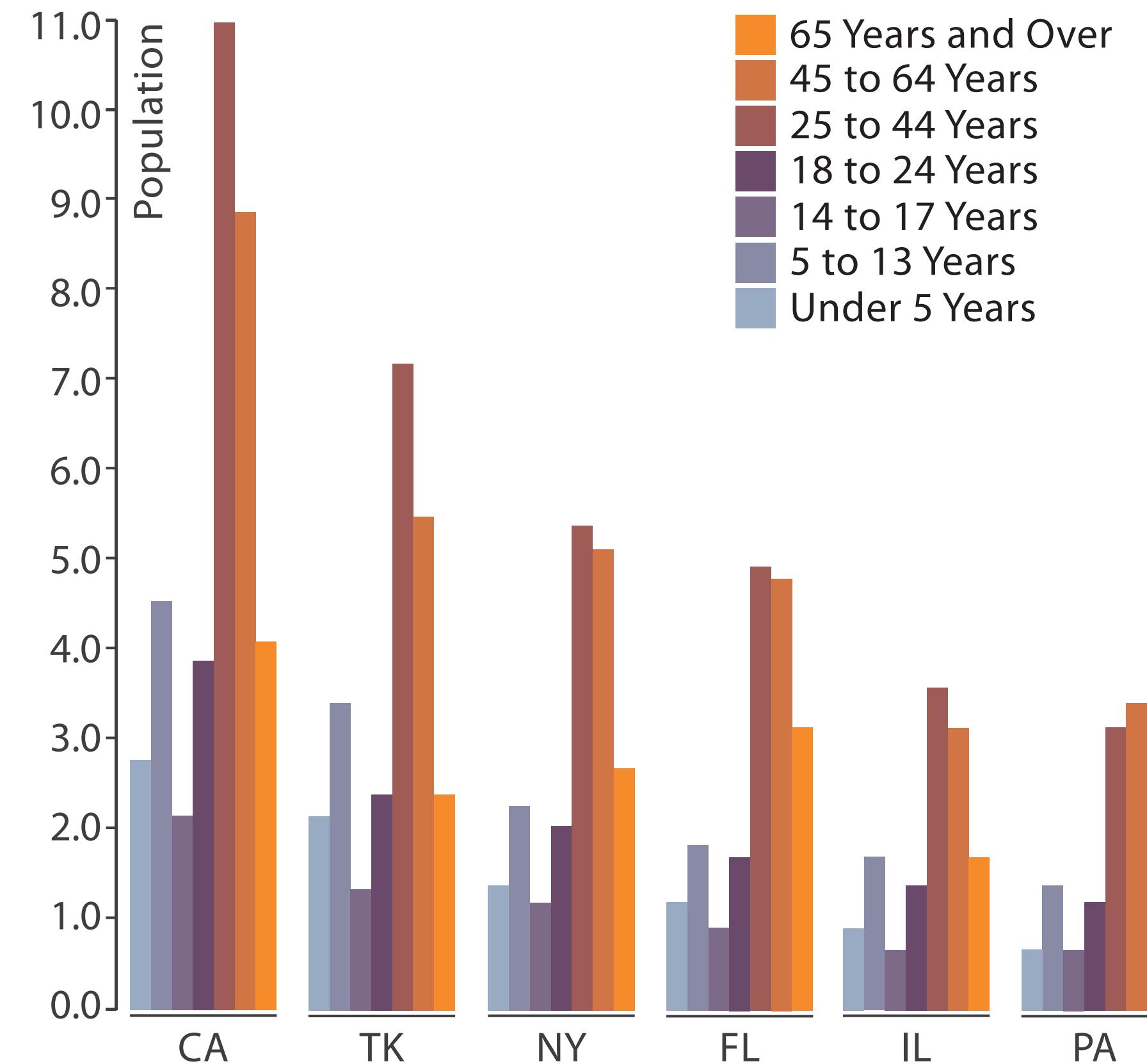
- how to **divide data between views**
  - **split** into regions by attributes
  - encodes association between items using **spatial proximity**
  - **order of splits** has major implications for what patterns are visible

## → Partition into Side-by-Side Views



# Partitioning: Grouped vs small-multiple bars

- single bar chart with grouped bars
  - **split by state** into regions
    - complex glyph within each region showing all ages
  - compare: easy within state, hard across ages
- **small-multiple bar charts**
  - **split by age** into regions
    - one chart per region
  - **compare**: easy within age, harder across states

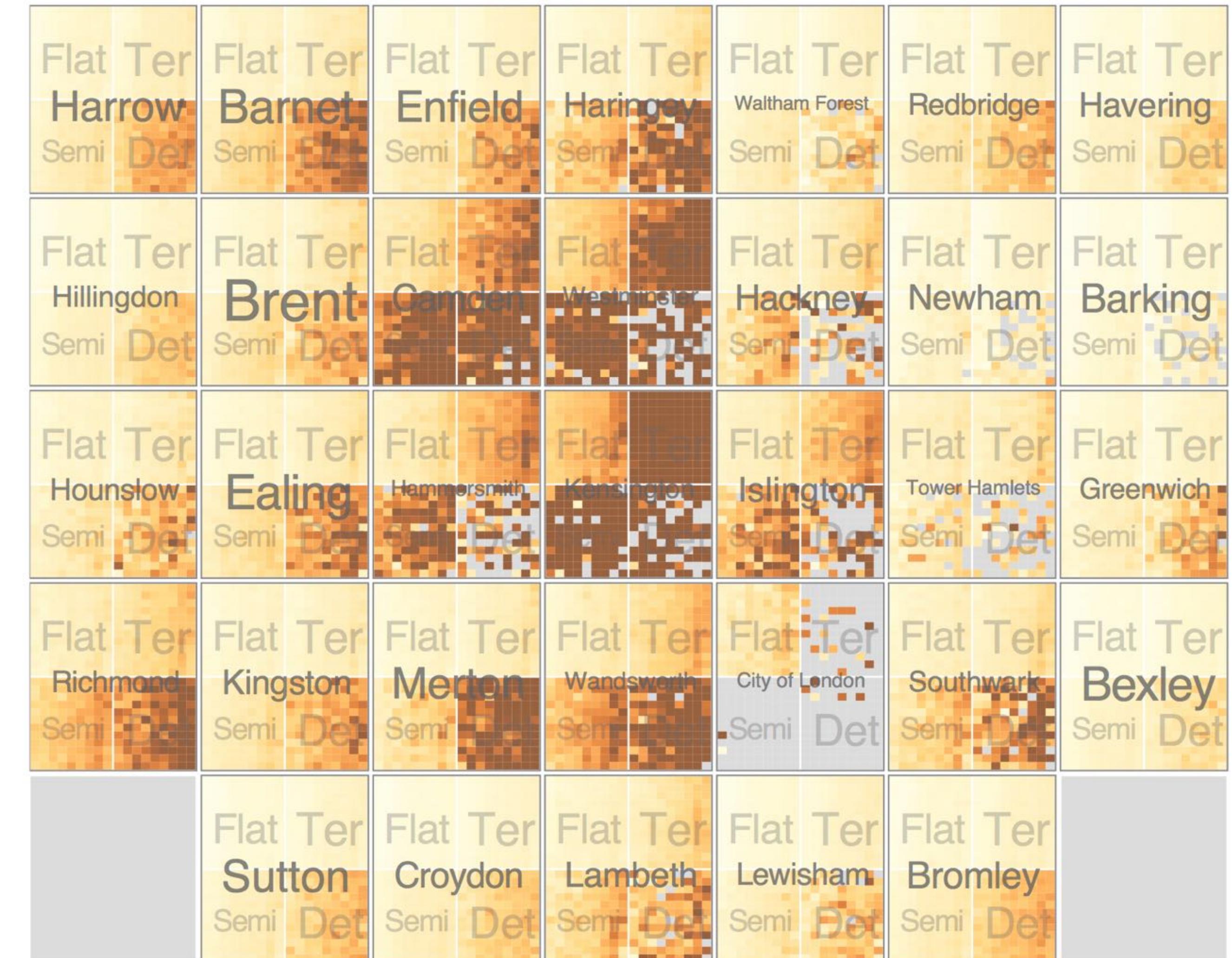


[<https://observablehq.com/@d3/grouped-bar-chart/2>]

System: HIVE

# Partitioning: Recursive subdivision

- split by neighborhood
- then by type
  - flat, terrace, semi-detached, detached
- then time
  - years as rows
  - months as columns
- color by price
- neighborhood patterns
  - where it's expensive
  - where you pay much more for detached type



[Configuring Hierarchical Layouts to Address Research Questions. Slingsby, Dykes, and Wood. IEEE Transactions on Visualization and Computer Graphics]

# Partitioning: Recursive subdivision

- switch order of splits
  - type then neighborhood
- switch color
  - by price variation
- type patterns
  - within specific type, which neighborhoods inconsistent

System: HIVE

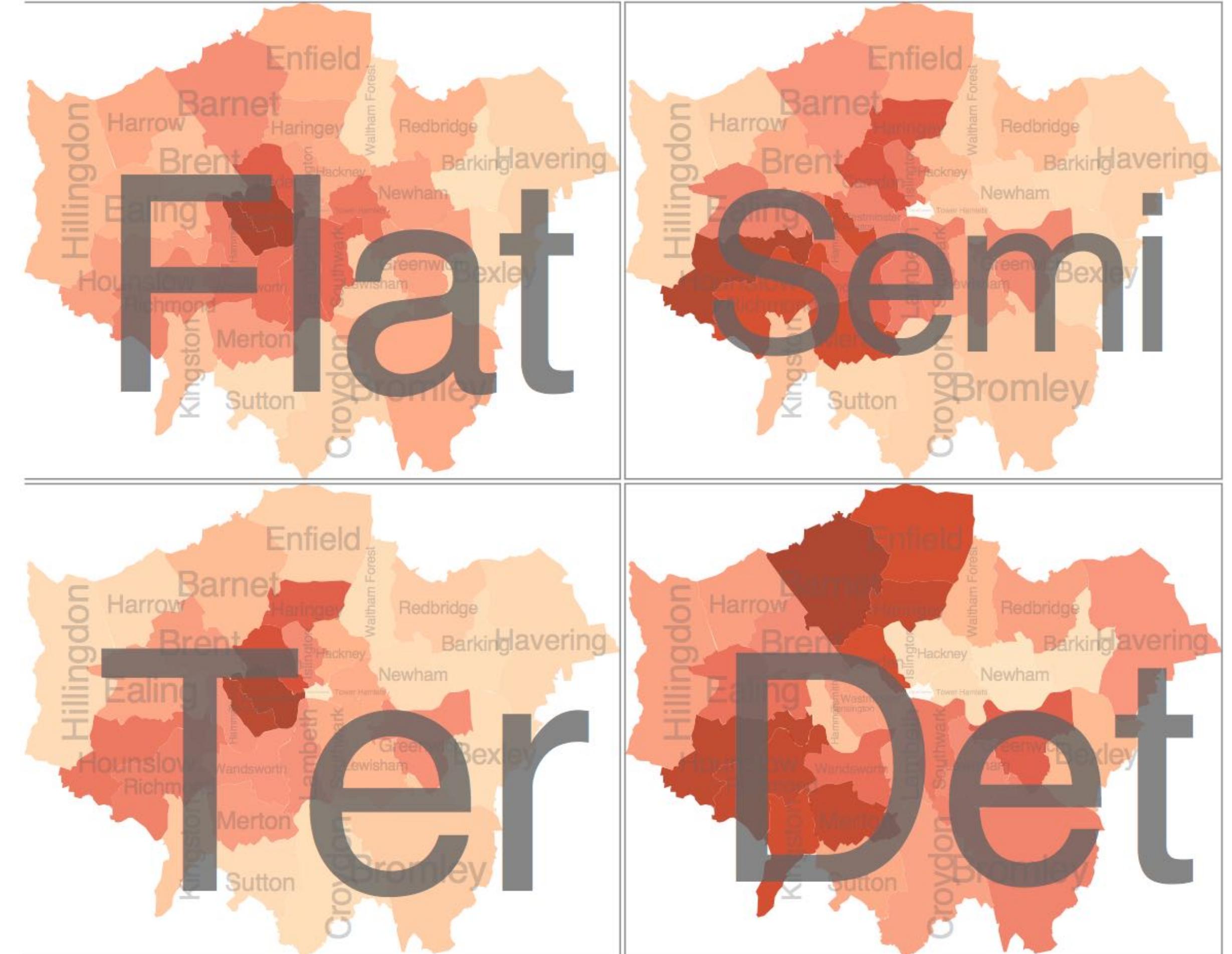


[Configuring Hierarchical Layouts to Address Research Questions. Slingsby, Dykes, and Wood. IEEE Transactions on Visualization and Computer Graphics]

# Partitioning: Recursive subdivision

- different encoding for second-level regions
  - choropleth maps

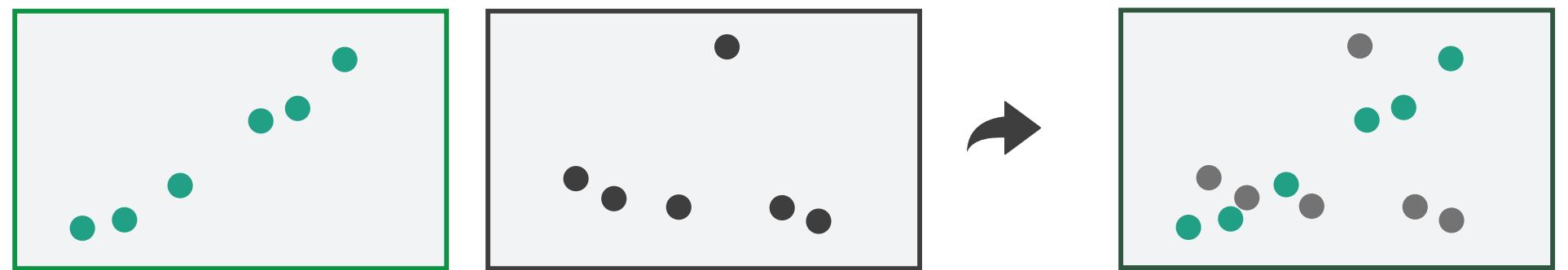
System: HIVE



# Superimpose layers

- **layer**: set of objects spread out over region
  - each set is visually distinguishable group
  - extent: whole view
- design choices
  - **how many layers, how to distinguish?**
    - encode with different, nonoverlapping channels
    - two layers achievable, three with careful design
  - small static set, or dynamic from many possible?

## → Superimpose Layers



# Static visual layering

- **foreground layer:** roads
  - hue, size distinguishing main from minor
  - high luminance contrast from background
- **background layer:** regions
  - desaturated colors for water, parks, land areas
- user can **selectively focus attention**

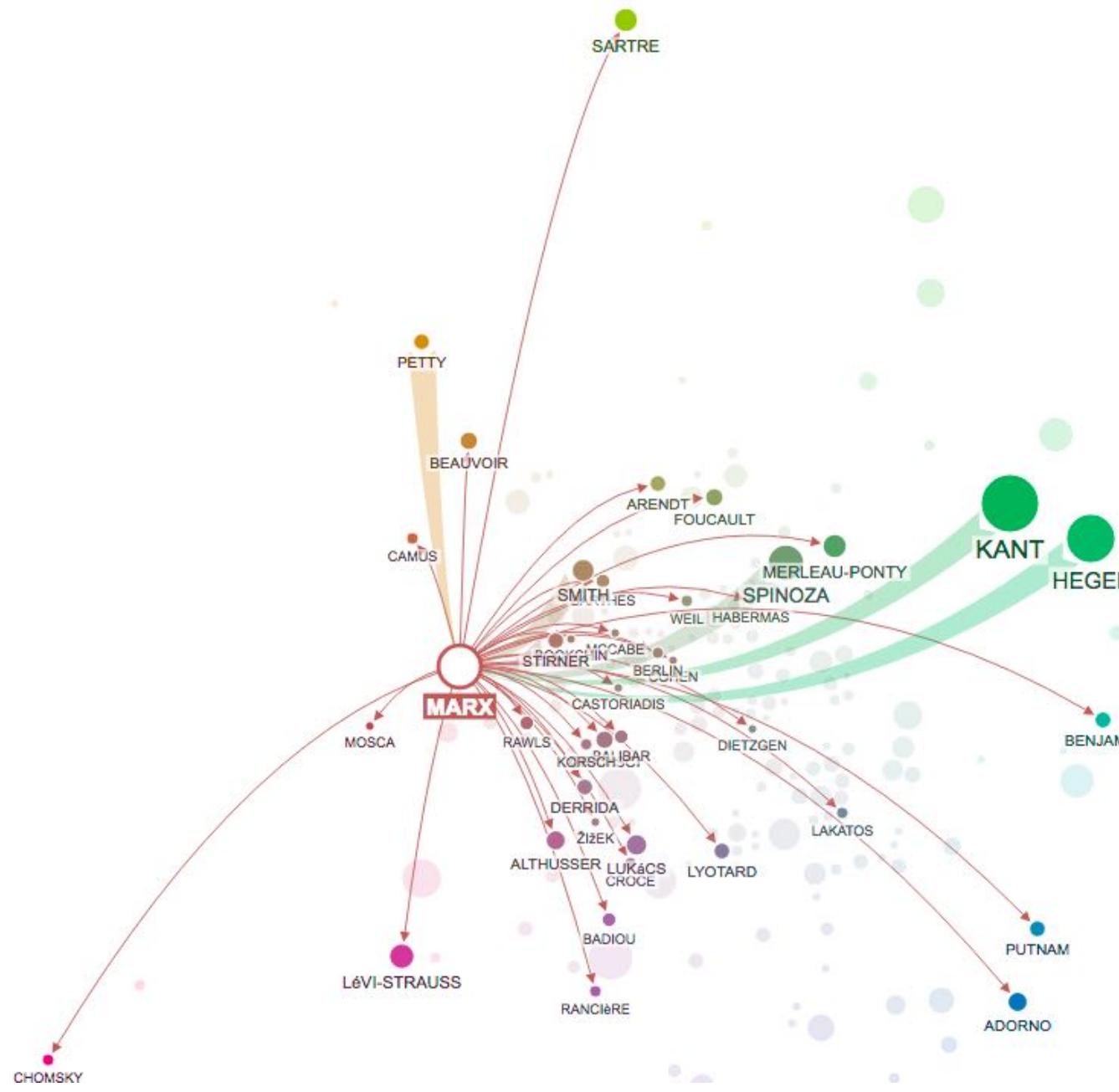


[Get it right in black and white. Stone. 2010.  
<http://www.stonesc.com/wordpress/2010/03/get-it-right-in-black-and-white>]

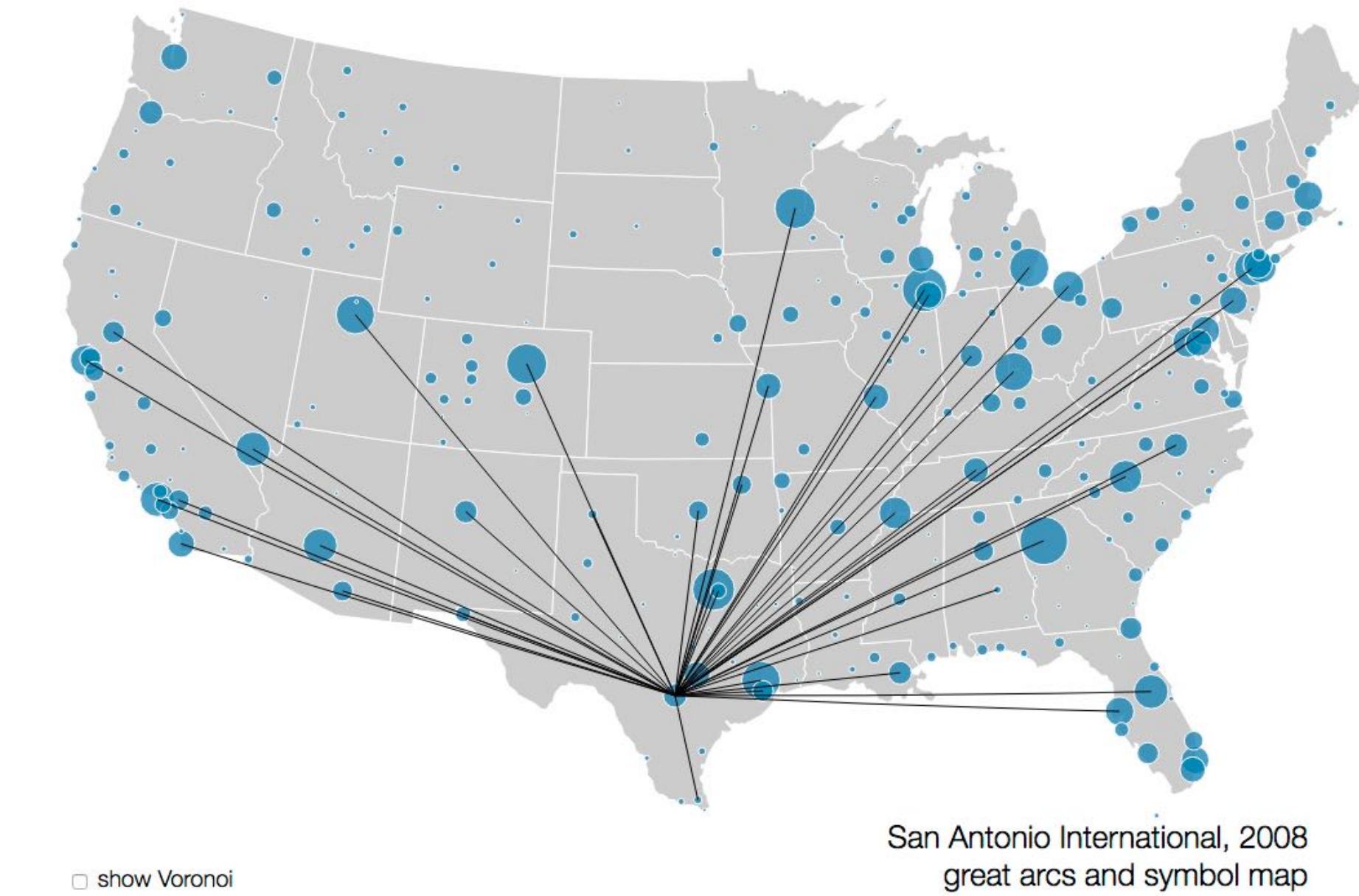
# Dynamic visual layering

- interactive, based on selection
  - one-hop neighbour highlighting

click (heavyweight)



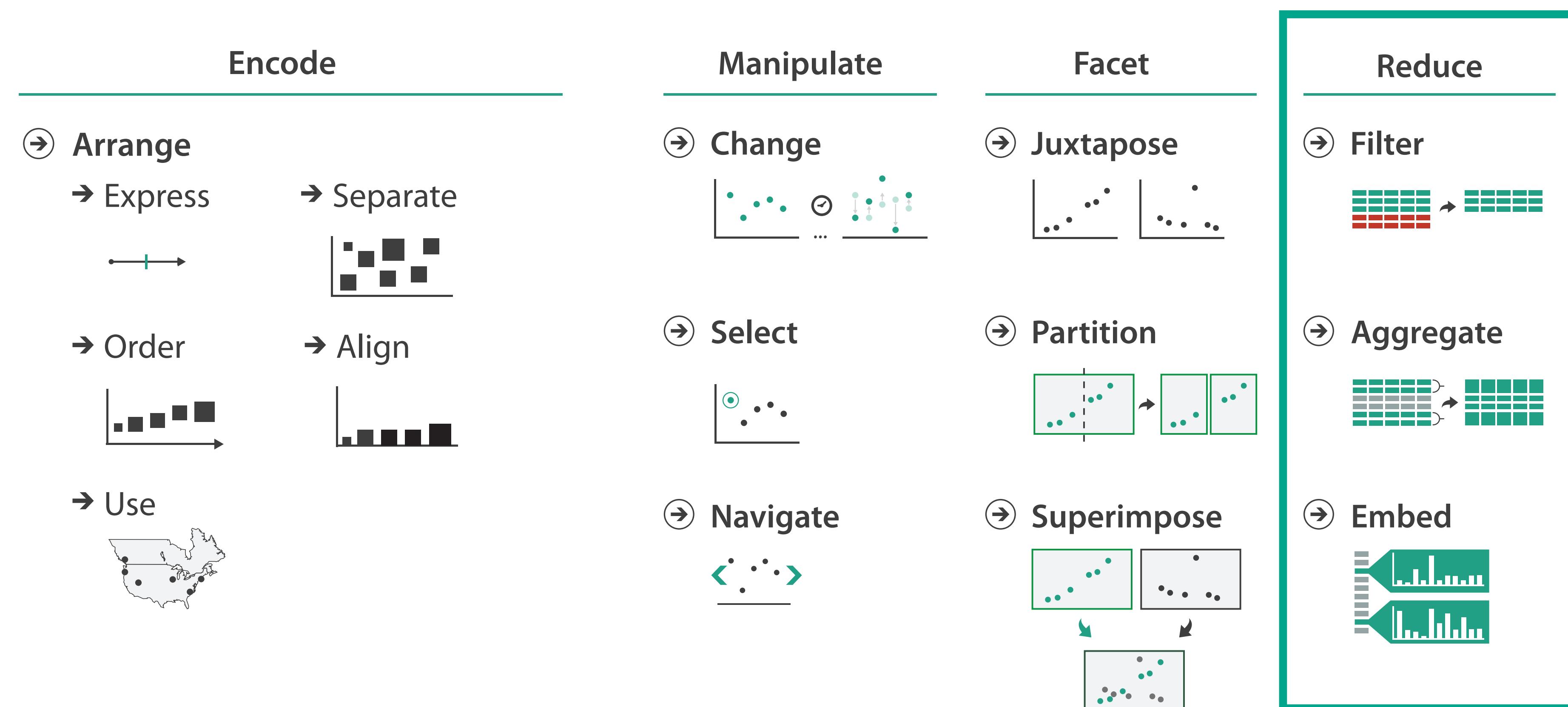
hover (fast)



<https://mariandoerk.de/edgemaps/demo/>

<http://mbostock.github.io/d3/talk/20111116/airports.html>

# How: Idiom design choices



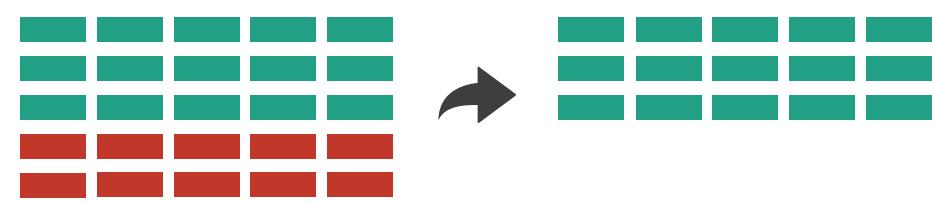
# Reduce items and attributes

- **reduce/increase: inverses**
- **filter**
  - pro: straightforward and intuitive
    - to understand and compute
  - con: out of sight, out of mind

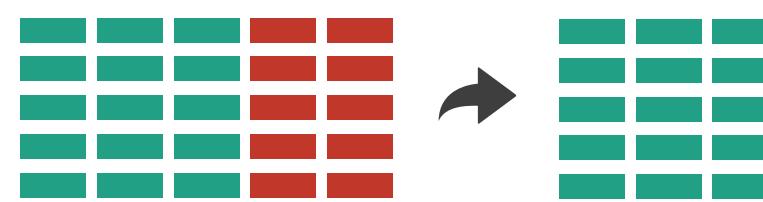
## Reducing Items and Attributes

### ④ Filter

→ Items



→ Attributes



# Reduce items and attributes

- **reduce/increase: inverses**
- **filter**
  - pro: straightforward and intuitive
    - to understand and compute
  - con: out of sight, out of mind
- **aggregation**
  - pro: inform about whole set
  - con: difficult to avoid losing signal
- **not mutually exclusive**
  - combine filter, aggregate
  - combine reduce, change, facet

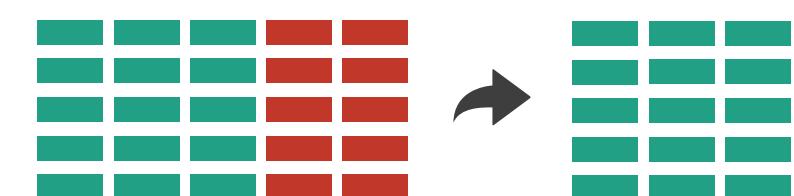
## Reducing Items and Attributes

### → Filter

→ Items

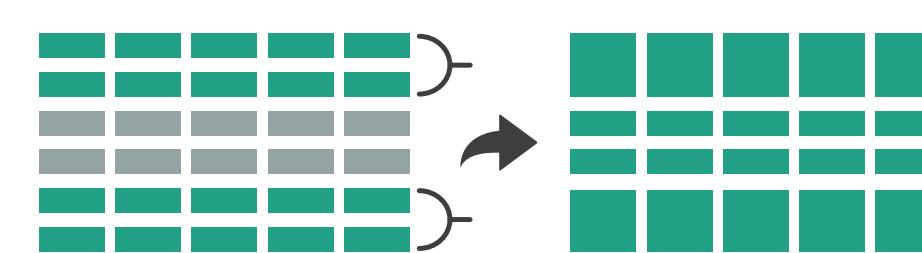


→ Attributes

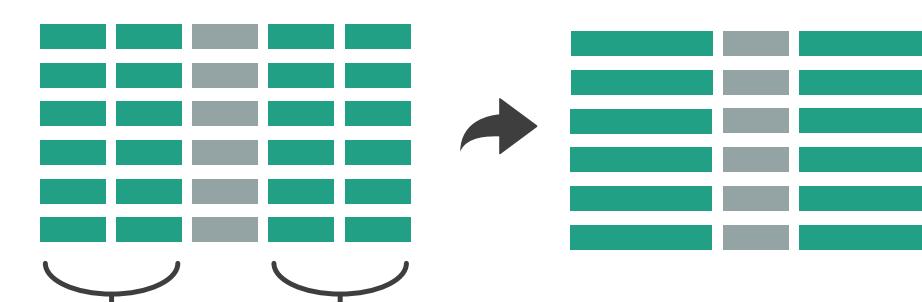


### → Aggregate

→ Items



→ Attributes



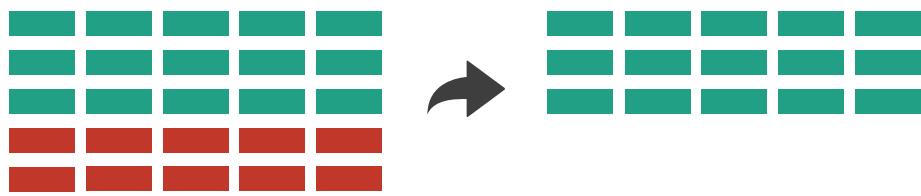
# Filter

- **eliminate some elements**
  - either items or attributes
- according to **what?**
  - any possible function that partitions dataset into two sets
  - attribute values bigger/smaller than x
  - noise/signal
- **filters vs queries**
  - **query**: start with nothing, add in elements
  - **filters**: start with everything, remove elements
  - **best approach depends on dataset size**

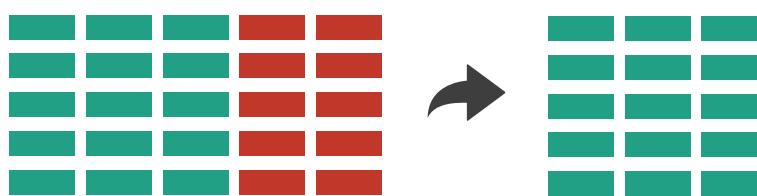
## Reducing Items and Attributes

### → Filter

→ Items

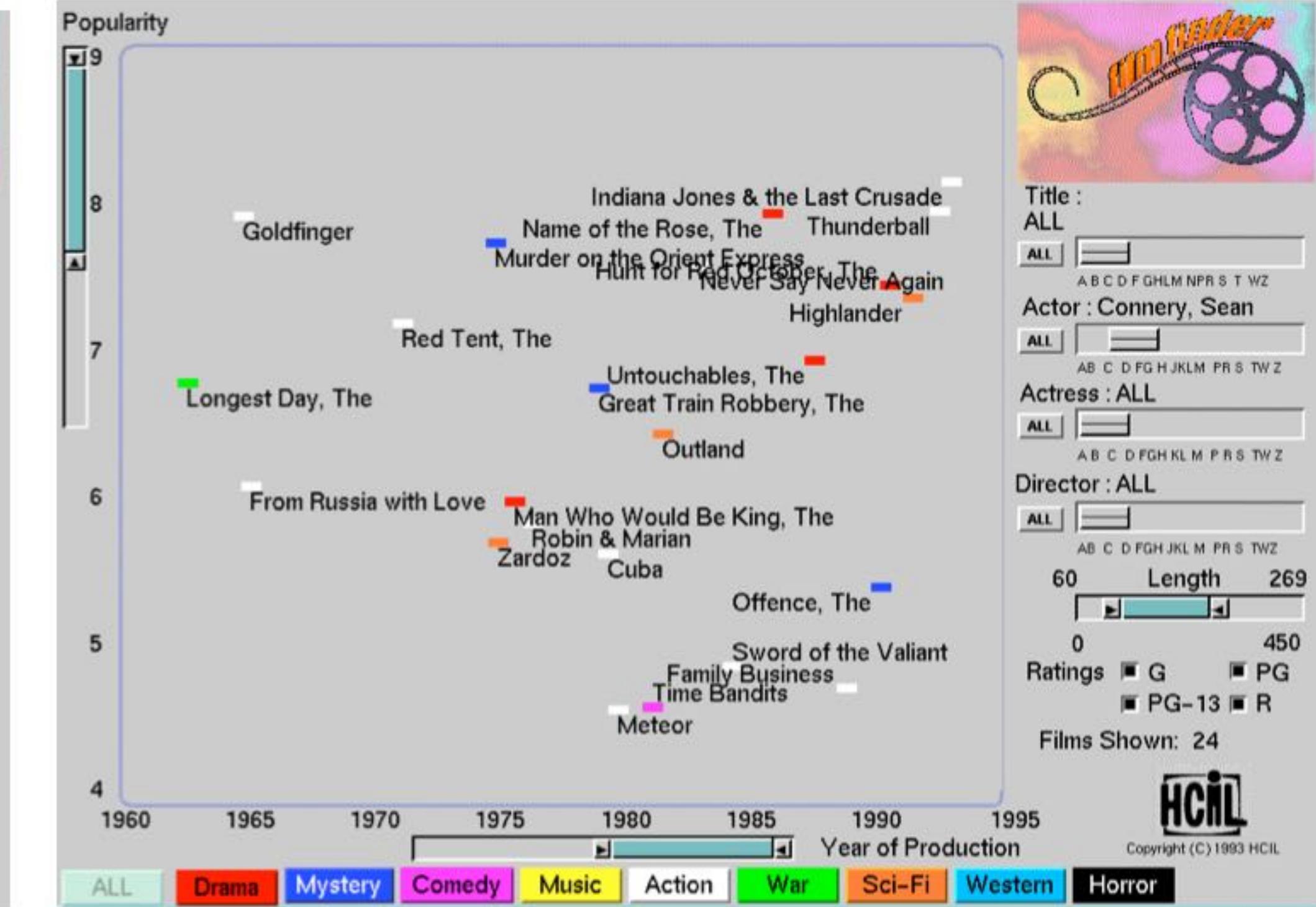
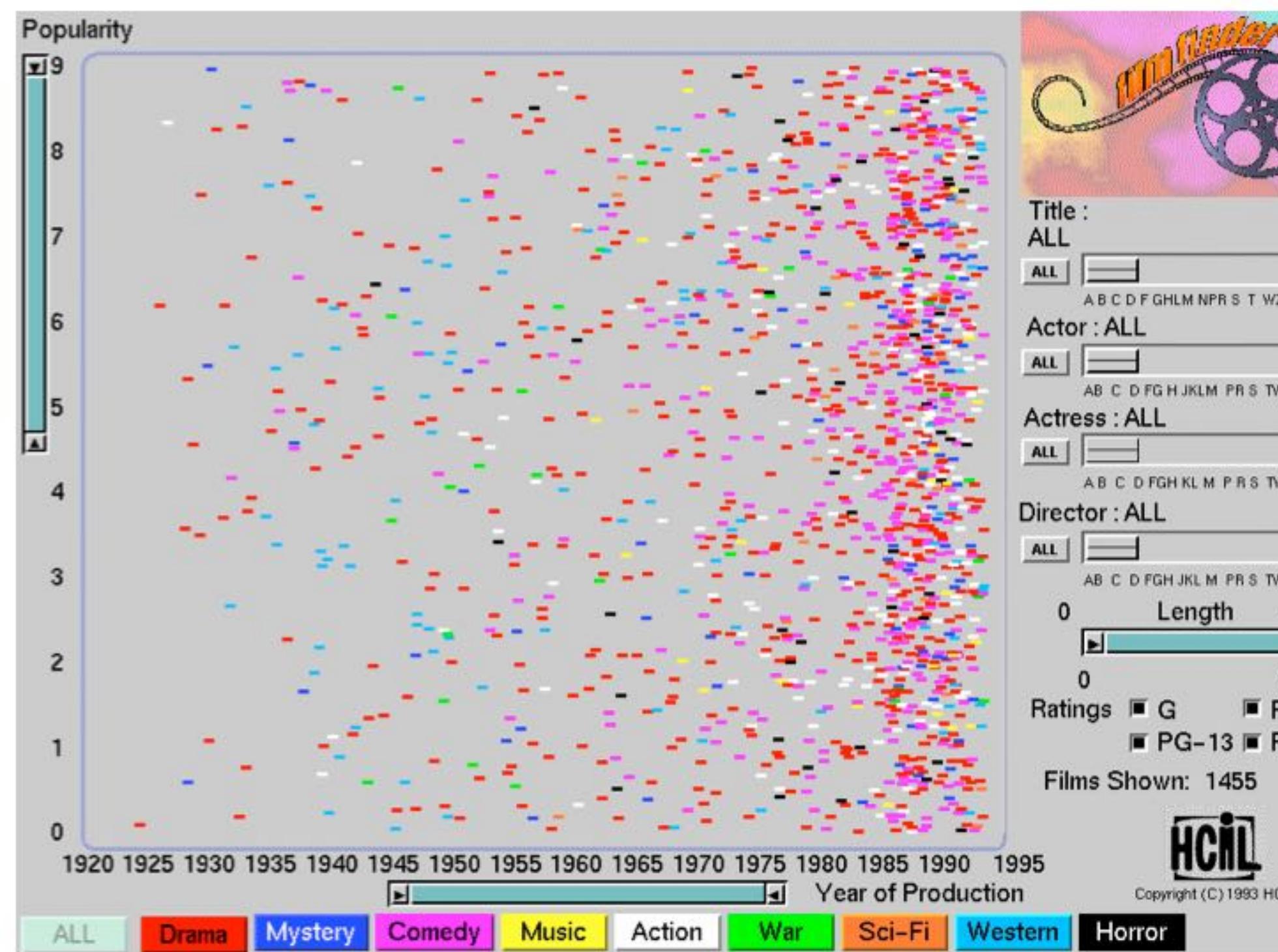


→ Attributes



# Idiom: FilmFinder

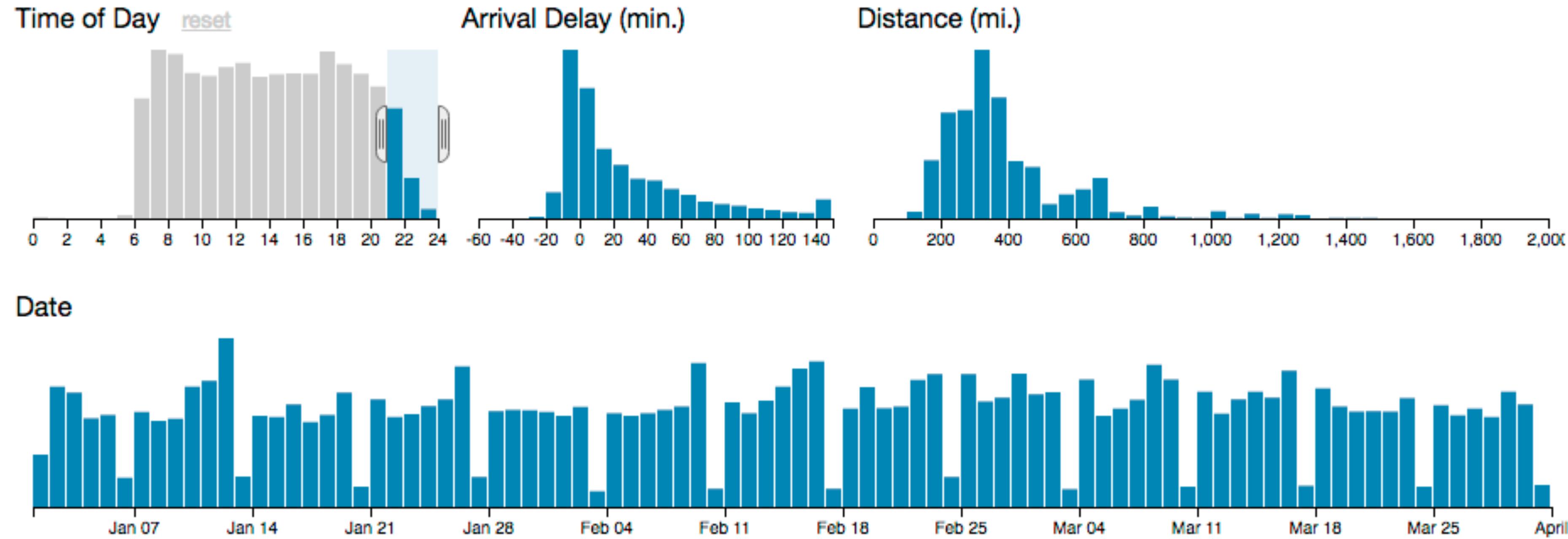
- dynamic queries/filters for items
  - tightly coupled interaction and visual encoding idioms, so user can immediately see results of action



[Ahlberg & Schneiderman, Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. CHI 1994.]

# Idiom: cross filtering

- item filtering
- coordinated views/controls combined
  - all selected histogram bisliders update when any ranges change



<http://square.github.io/crossfilter/>

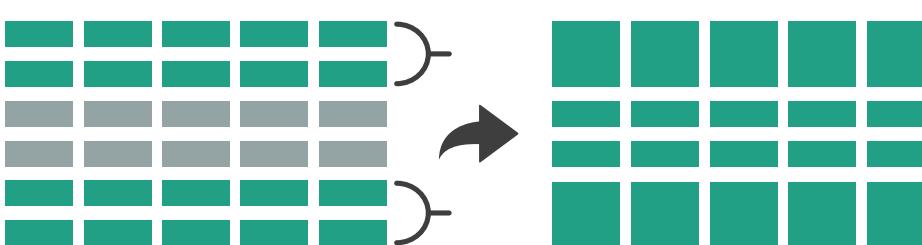
<https://observablehq.com/@uwdata/interaction>

# Aggregate

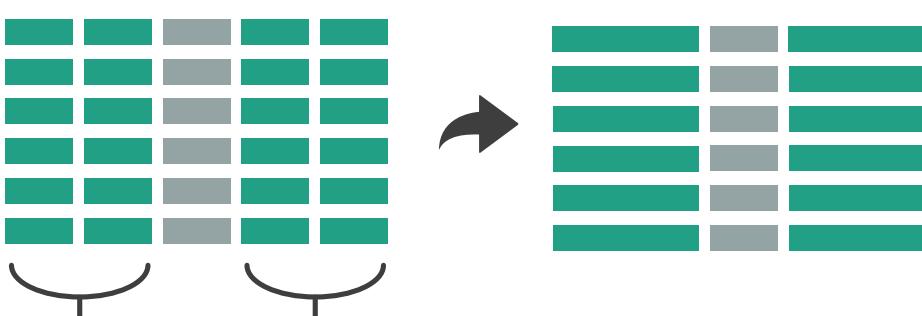
- a group of elements is represented by a **smaller number of derived elements**

➔ Aggregate

→ Items

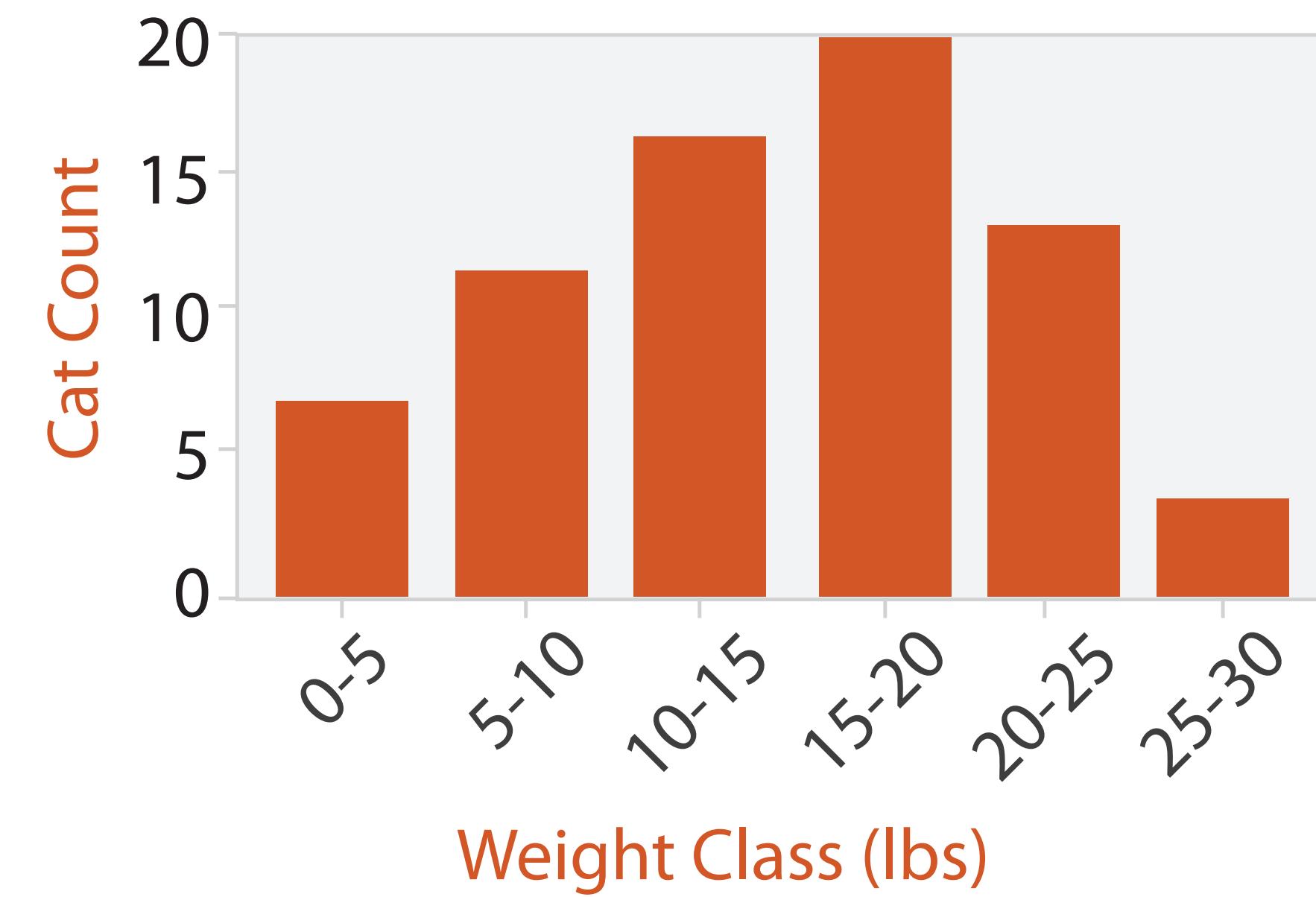


→ Attributes



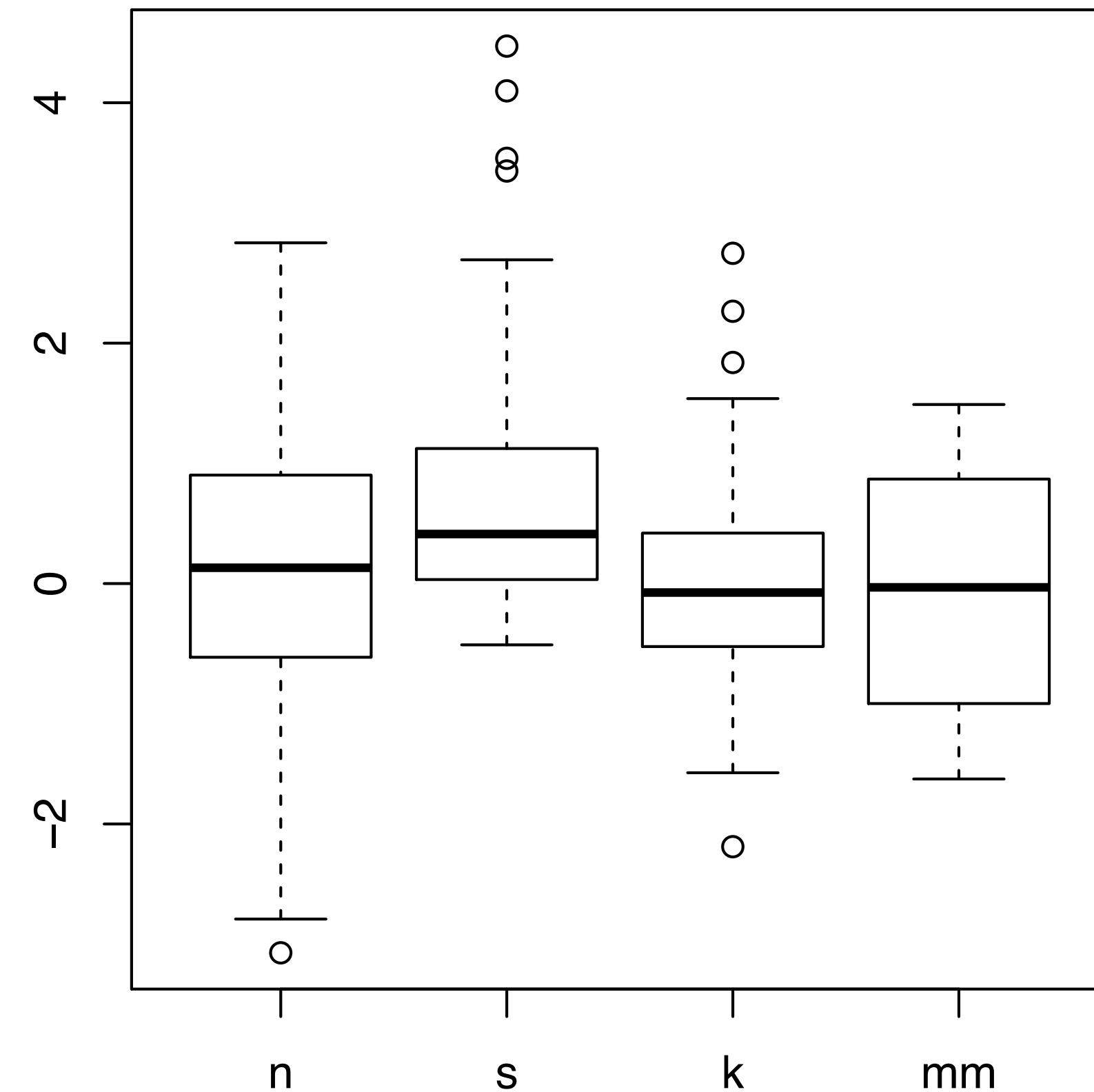
# Idiom: histogram

- static item aggregation
- task: find distribution
- data: table
- derived data
  - new table: keys are bins, values are counts
- bin size crucial
  - pattern can change dramatically depending on discretization
  - opportunity for interaction: control bin size on the fly



# Idiom: boxplot

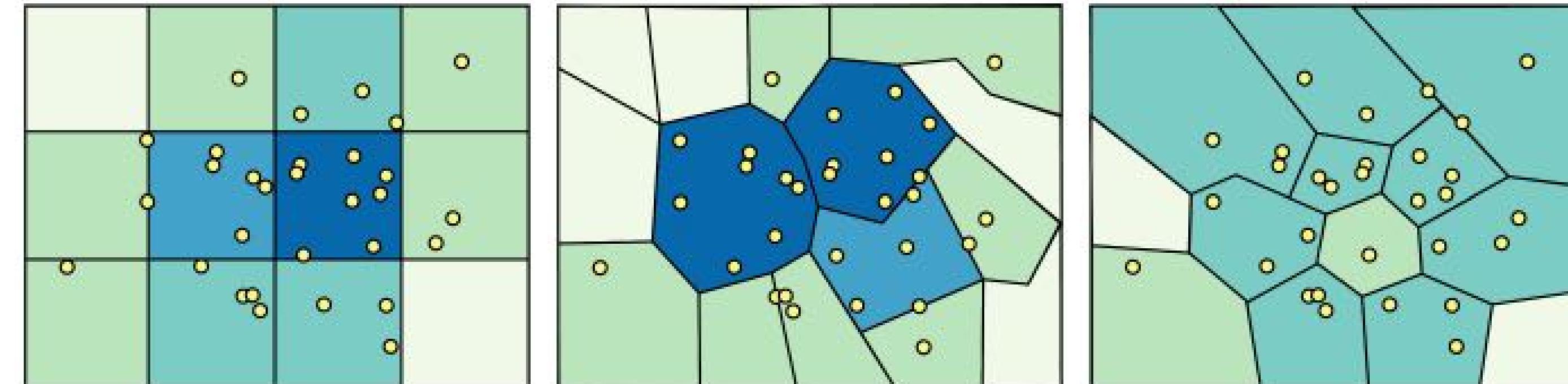
- static item aggregation
- task: find distribution
- data: table
- derived data
  - 5 quant attrs
    - median: central line
    - lower and upper quartile: boxes
    - lower upper fences: whiskers
      - values beyond which items are outliers
  - outliers beyond fence cutoffs explicitly shown
- scalability
  - unlimited number of items!



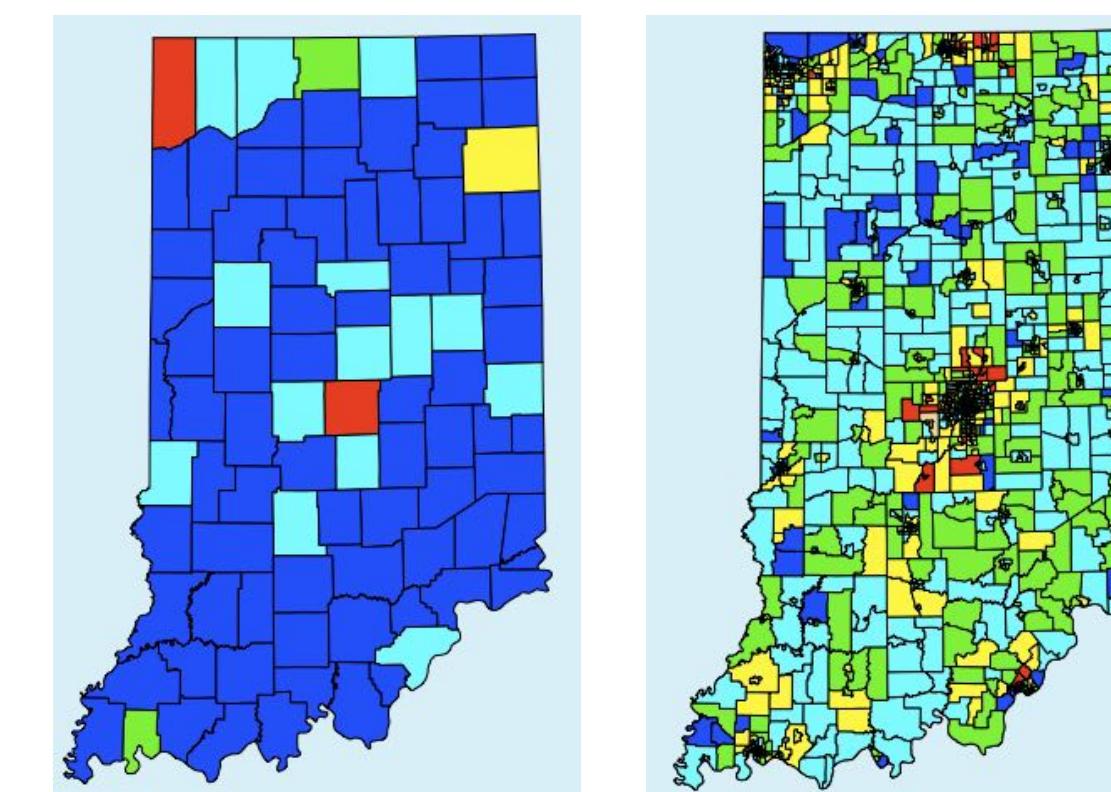
[40 years of boxplots. Wickham and Stryjewski. 2012]

# Spatial aggregation

- MAUP: Modifiable Areal Unit Problem
  - changing boundaries of cartographic regions can yield dramatically different results
  - zone effects



- scale effects

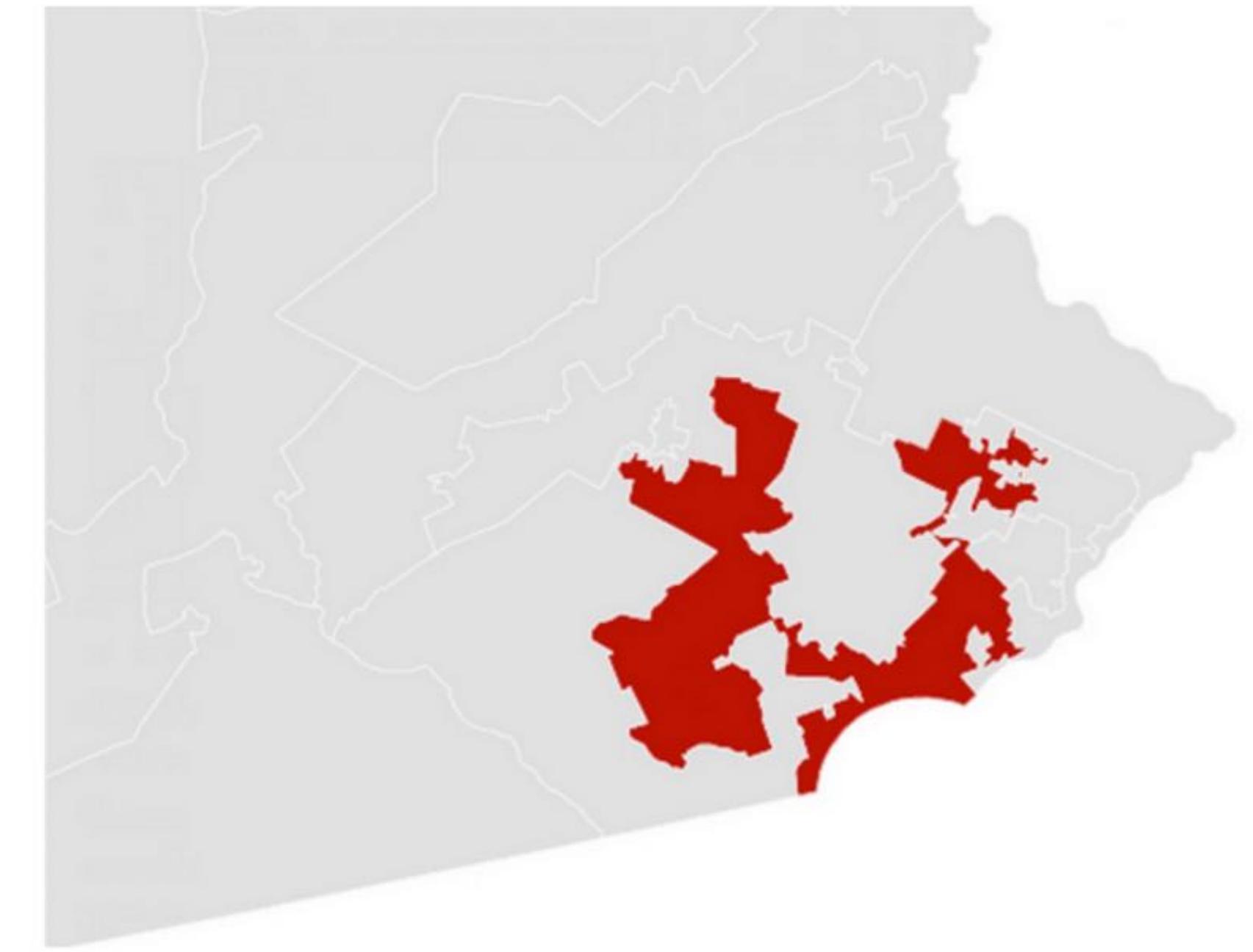
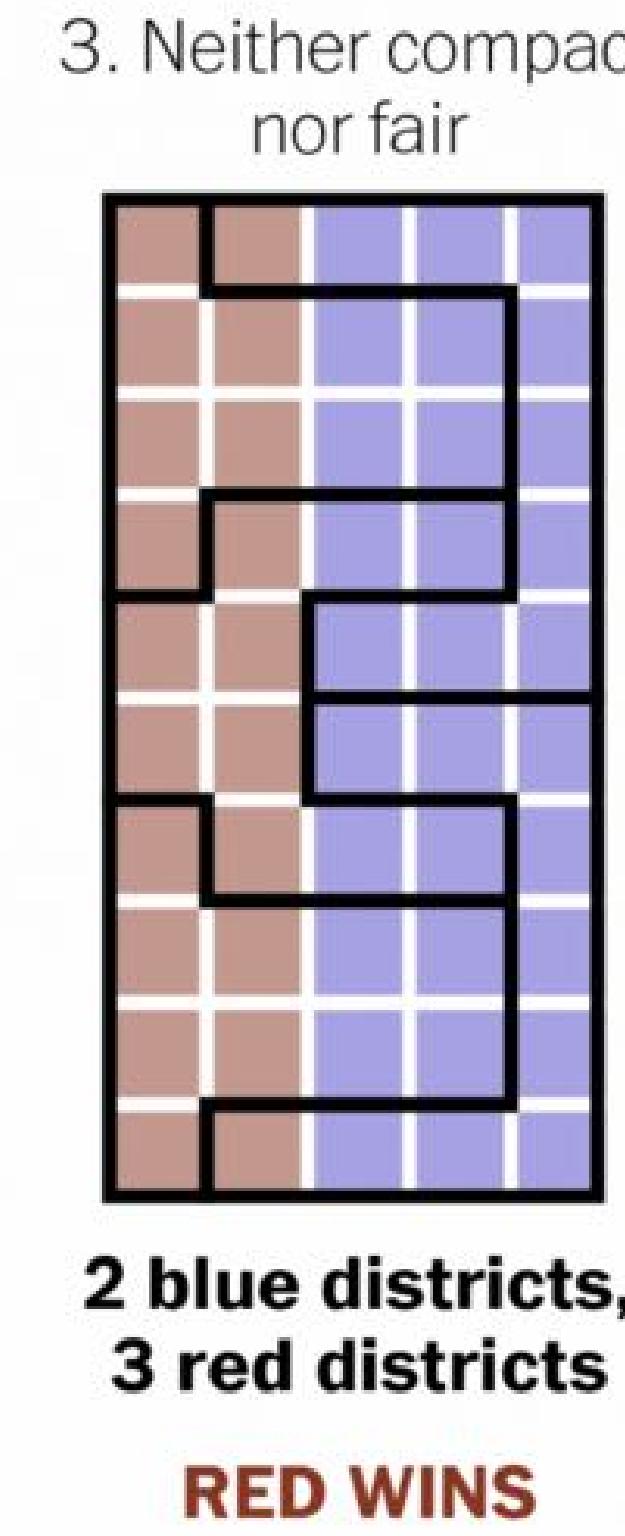
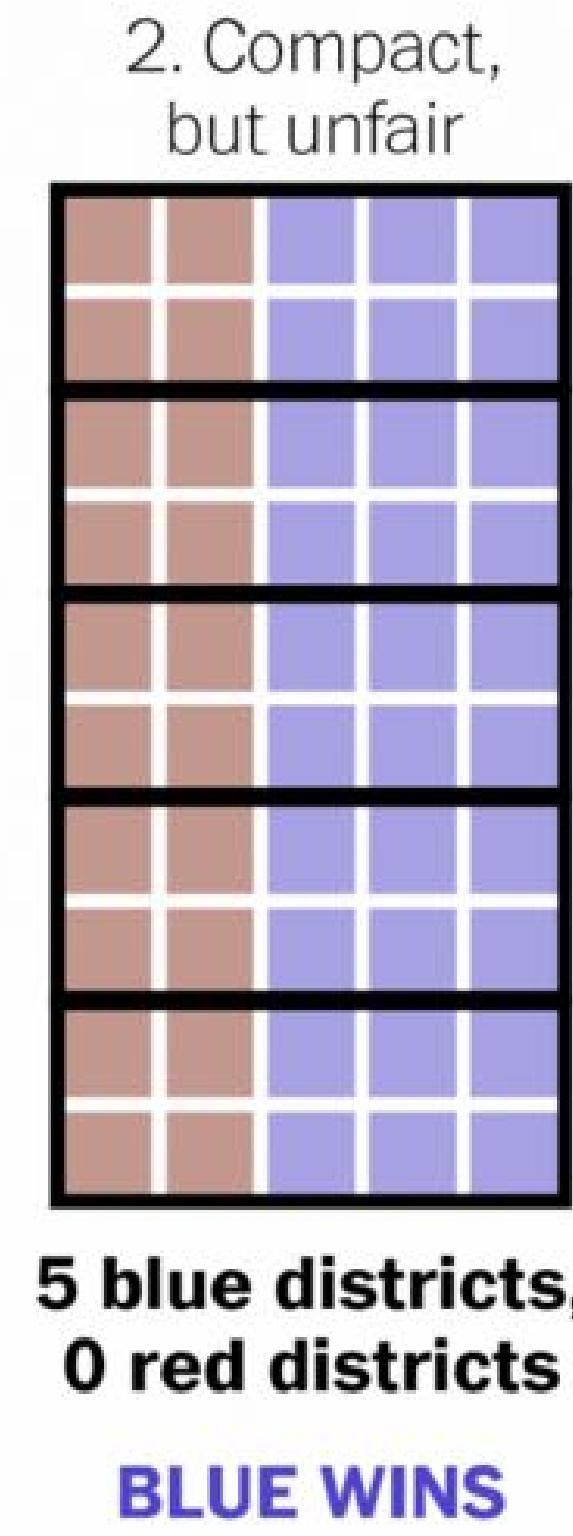
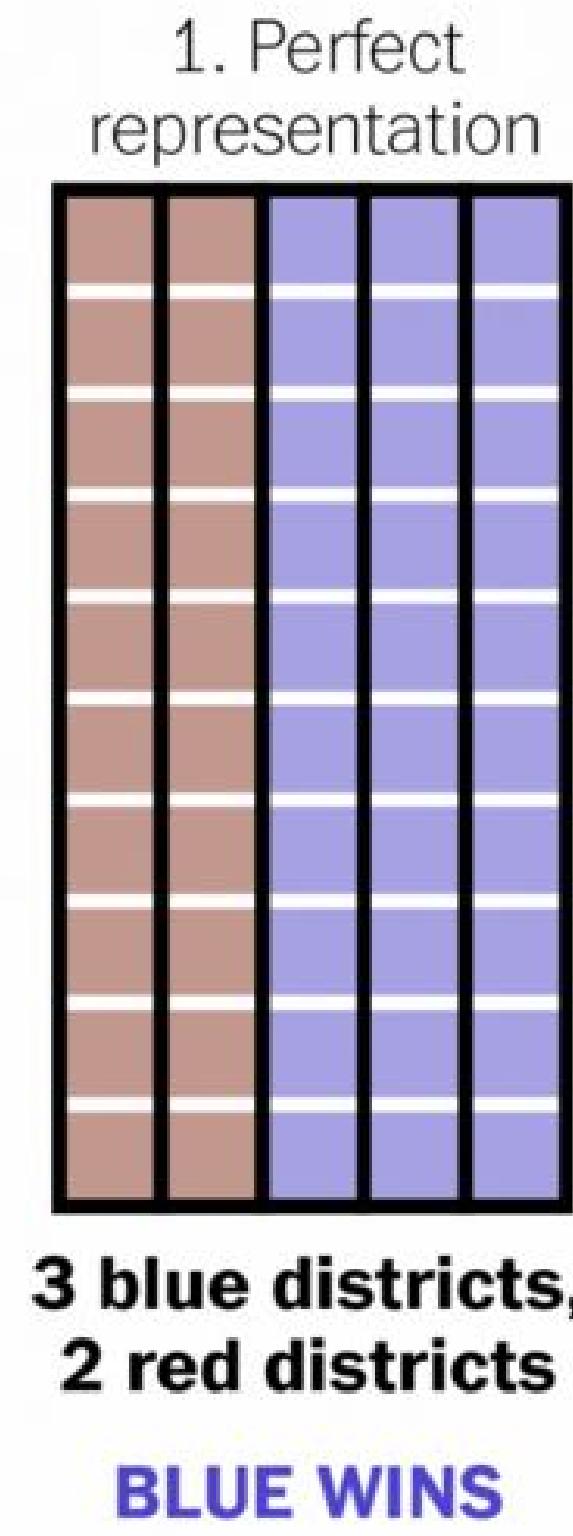
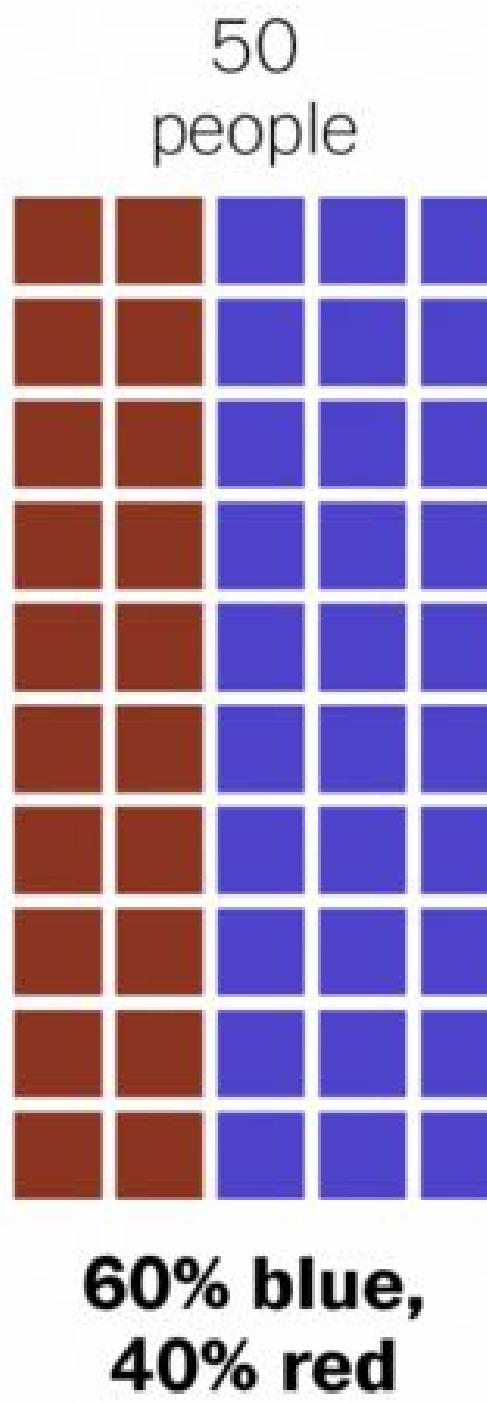


<https://blog.cartographica.com/blog/2011/5/19/the-modifiable-areal-unit-problem-in-gis.html>

# Gerrymandering: MAUP for political gain

## Gerrymandering, explained

Three different ways to divide 50 people into five districts



A real district in Pennsylvania:  
Democrats won 51% of the vote but  
only 5 out of 18 house seats

# Dynamic aggregation: Clustering

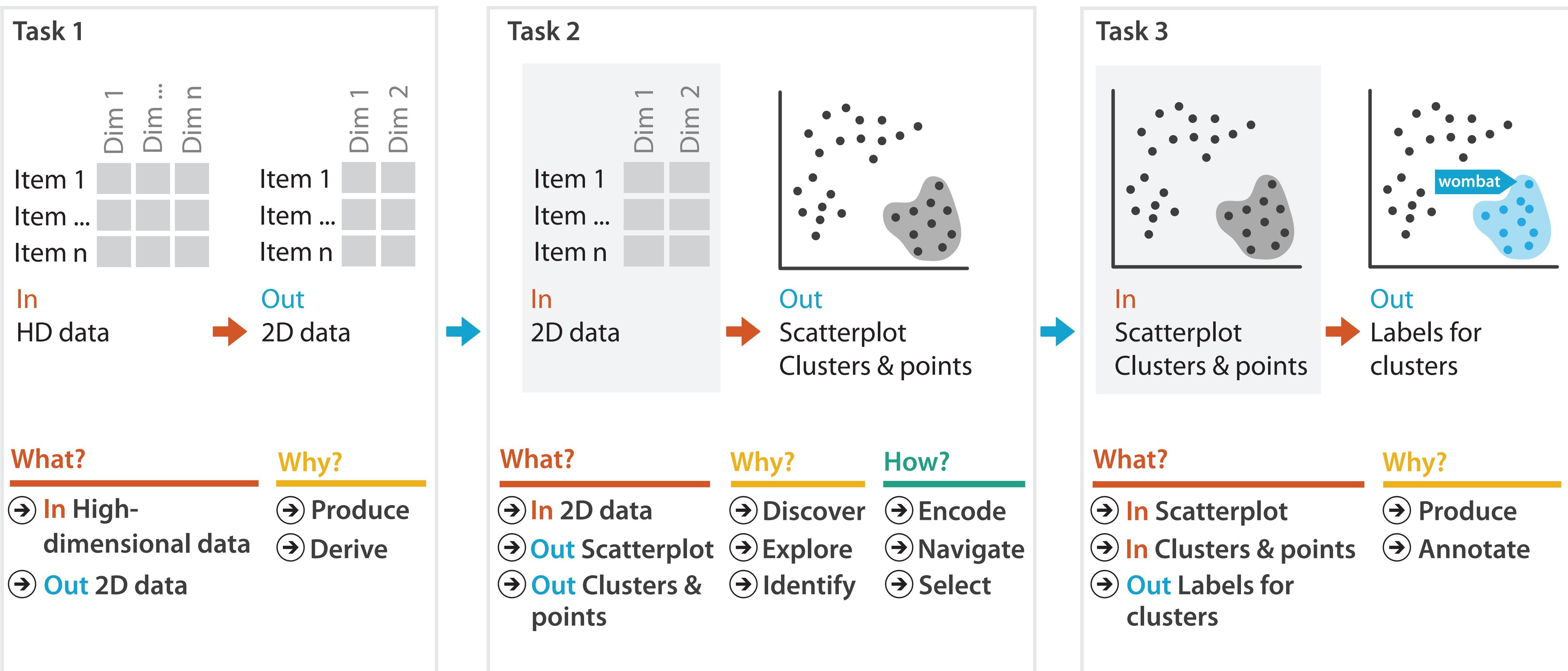
- **clustering:** classification of items into similar bins
  - based on similarity measure
  - hierarchical algorithms produce "similarity tree": cluster hierarchy
    - agglomerative clustering: start w/ each node as own cluster, then iteratively merge
- **cluster hierarchy:** derived data used w/ many dynamic aggregation idioms
  - cluster more homogeneous than whole dataset
    - statistical measures & distribution more meaningful

# Attribute aggregation: dimensionality reduction

- attribute aggregation
  - derive low-dimensional target space from high-dimensional measured space
    - capture most of variance with minimal error
  - use when you can't directly measure what you care about
    - true dimensionality of dataset conjectured to be smaller than dimensionality of measurements
    - latent factors, hidden variables

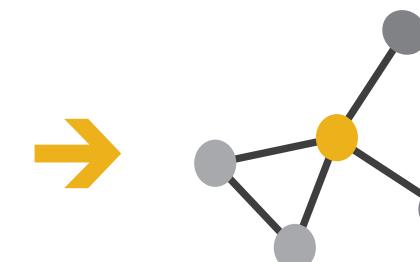
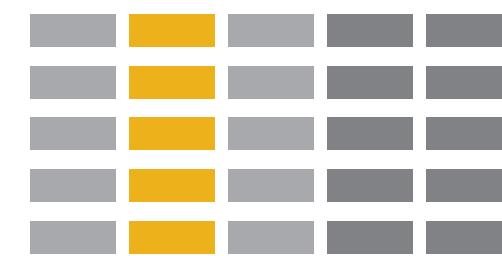


# Idiom: Dimensionality reduction for documents



# How to handle complexity: 4 strategies

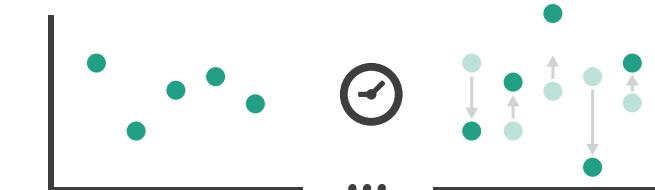
## → Derive



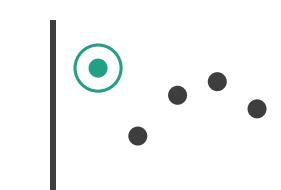
derive new data to show within view  
change view over time  
facet across multiple views  
reduce items/attributes within single view

## Manipulate

### → Change



### → Select

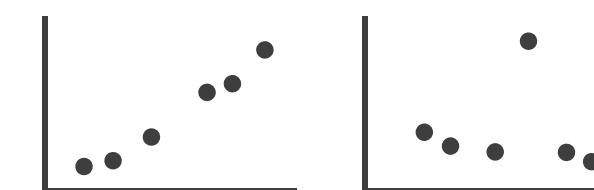


### → Navigate

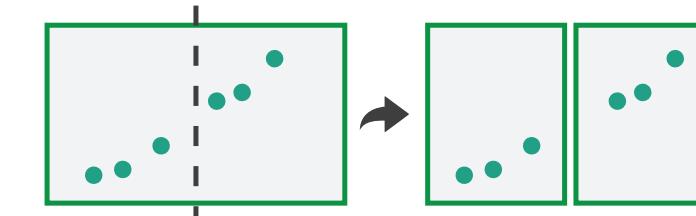


## Facet

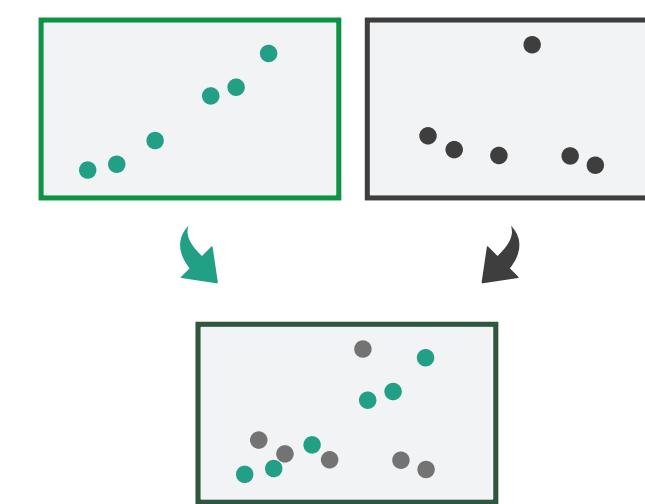
### → Juxtapose



### → Partition

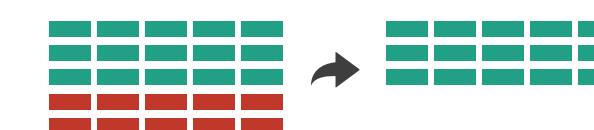


### → Superimpose

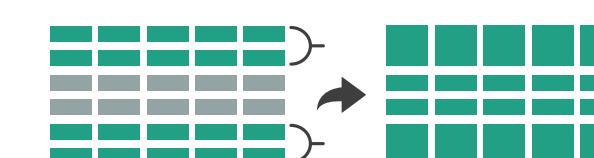


## Reduce

### → Filter



### → Aggregate



### → Embed



# Embed: Focus+Context

- combine focus + context info within single view
  - vs standard navigation within view
  - vs multiple views
- elide data
  - selectively filter and aggregate

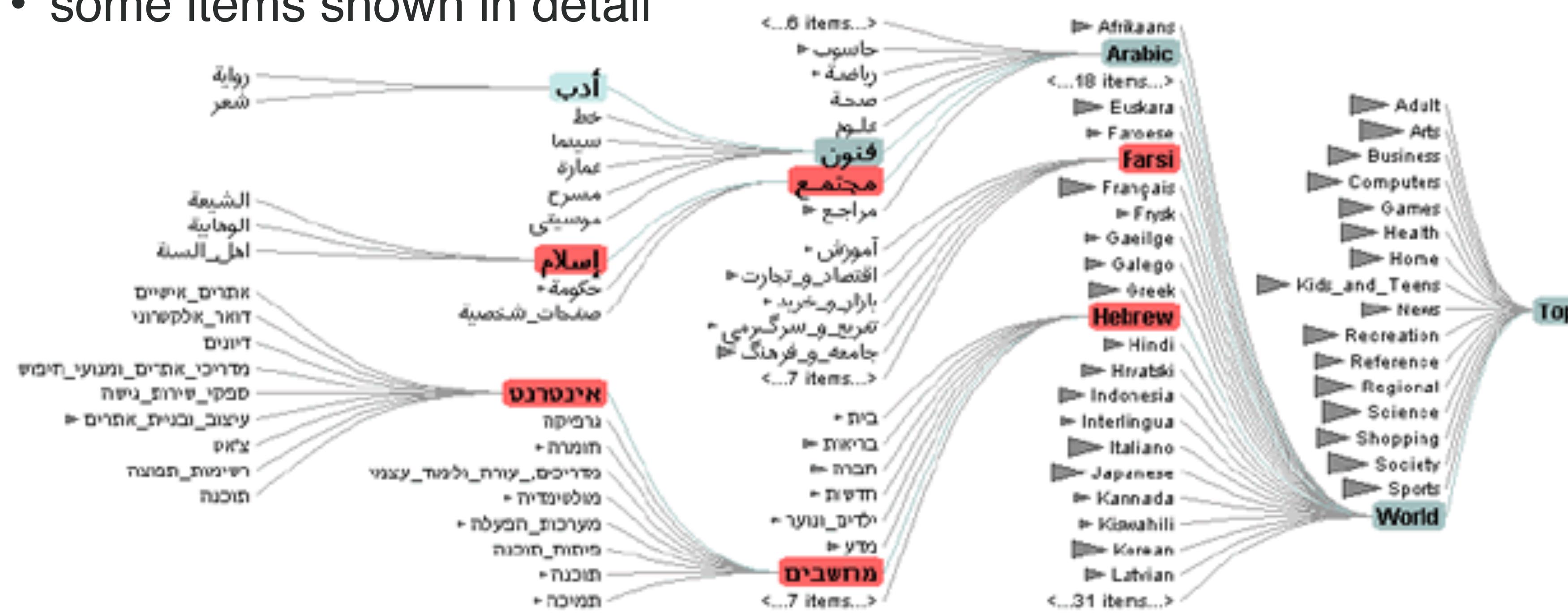
➔ Embed

→ Elide Data



# Idiom: DOI Trees Revisited

- focus+context choice: elide
  - some items dynamically filtered out
  - some items dynamically aggregated together
  - some items shown in detail



[DOI Trees Revisited: Scalable, Space-Constrained Visualization of Hierarchical Data. Heer and Card. Proc. Advanced Visual Interfaces (AVI), pp. 421–424, 2004.]

# Embed: Focus+Context

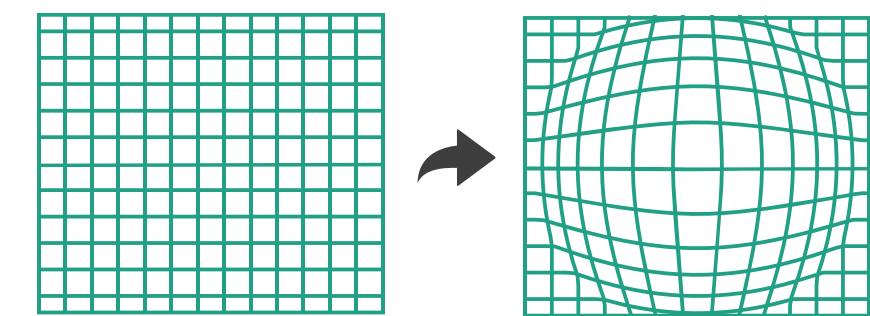
- combine focus + context info within single view
  - vs standard navigation within view
  - vs multiple views
- elide data
  - selectively filter and aggregate
- distort geometry
  - carefully chosen to integrate F+C

➔ Embed

→ Elide Data

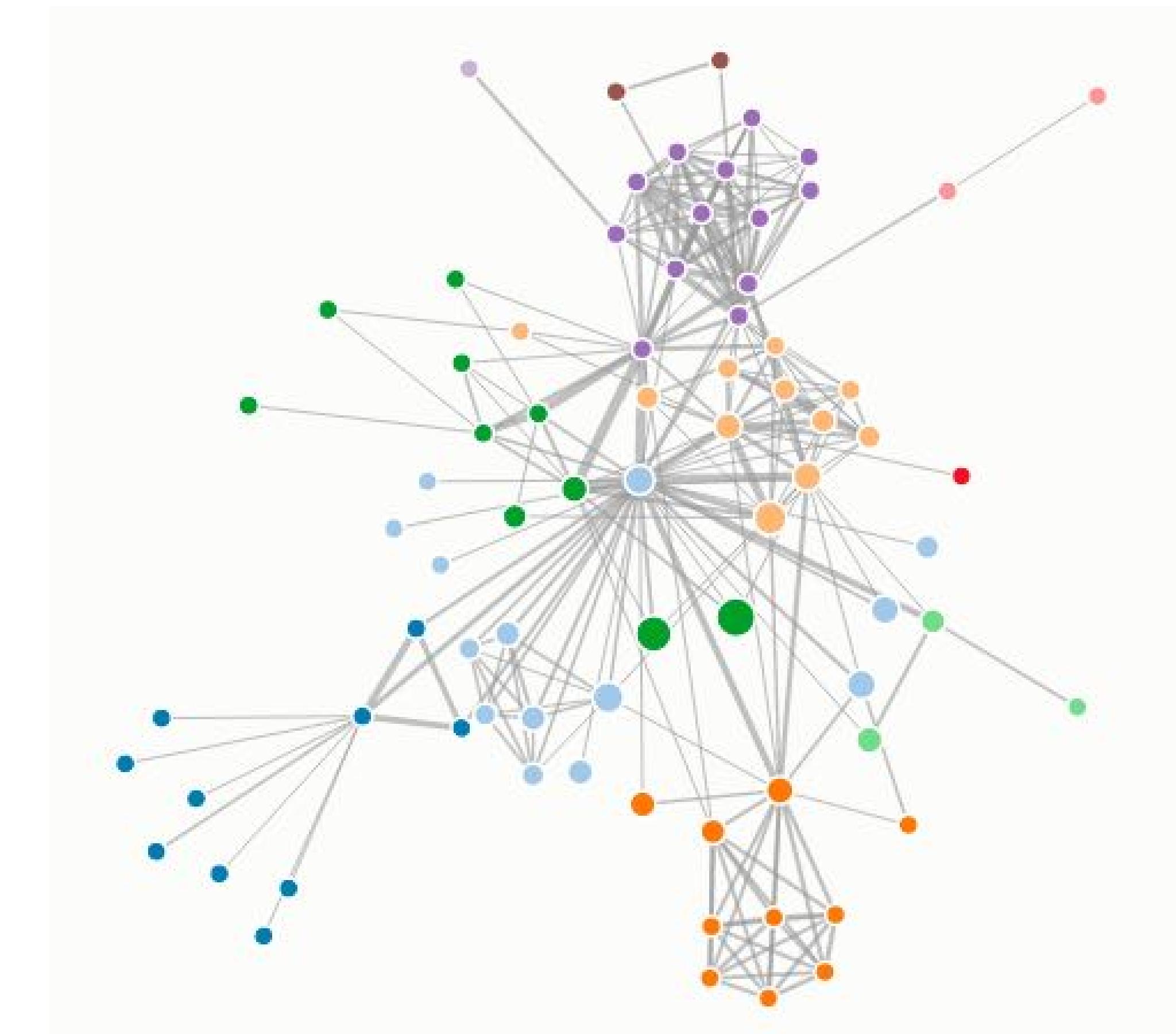


→ Distort Geometry



# Idiom: Fisheye Lens

- F+C choice: distort geometry
  - shape: radial
  - focus: single extent
  - extent: local
  - metaphor: draggable lens



[D3 Fisheye Lens] <https://bostocks.org/mike/fisheye/>

# Embed: Focus+Context

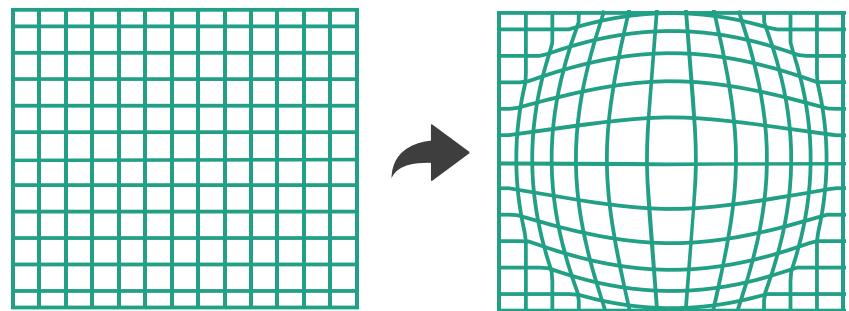
- combine focus + context info within single view
  - vs standard navigation within view
  - vs multiple views
- elide data
  - selectively filter and aggregate
- distort geometry:  
design choices
  - region shape: radial, rectilinear, complex
  - how many regions: one, many
  - region extent: local, global
  - interaction metaphor

➔ Embed

→ Elide Data



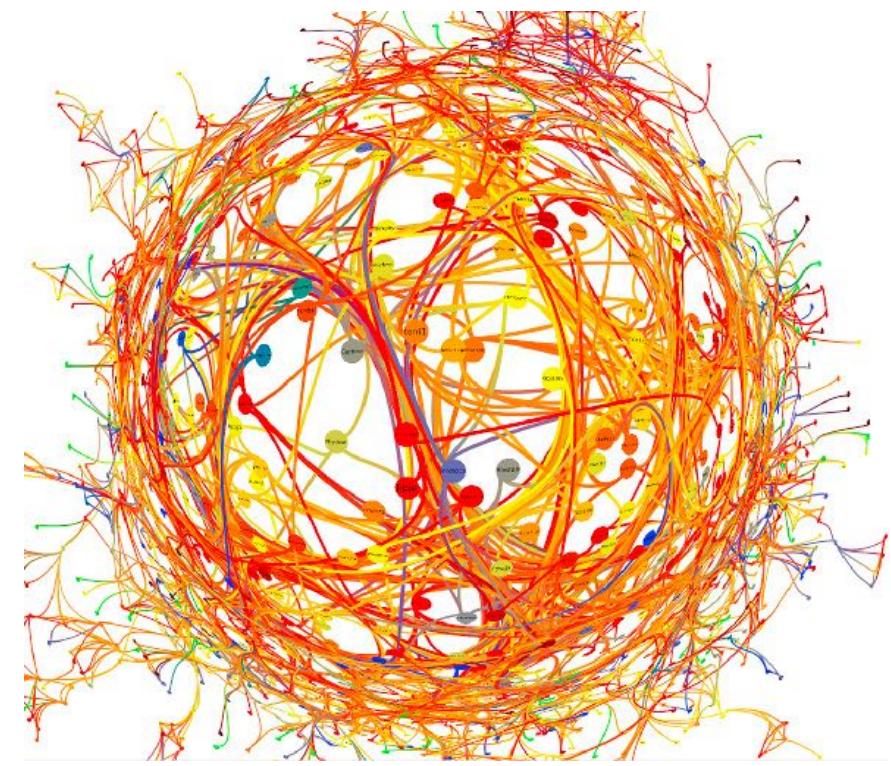
→ Distort Geometry



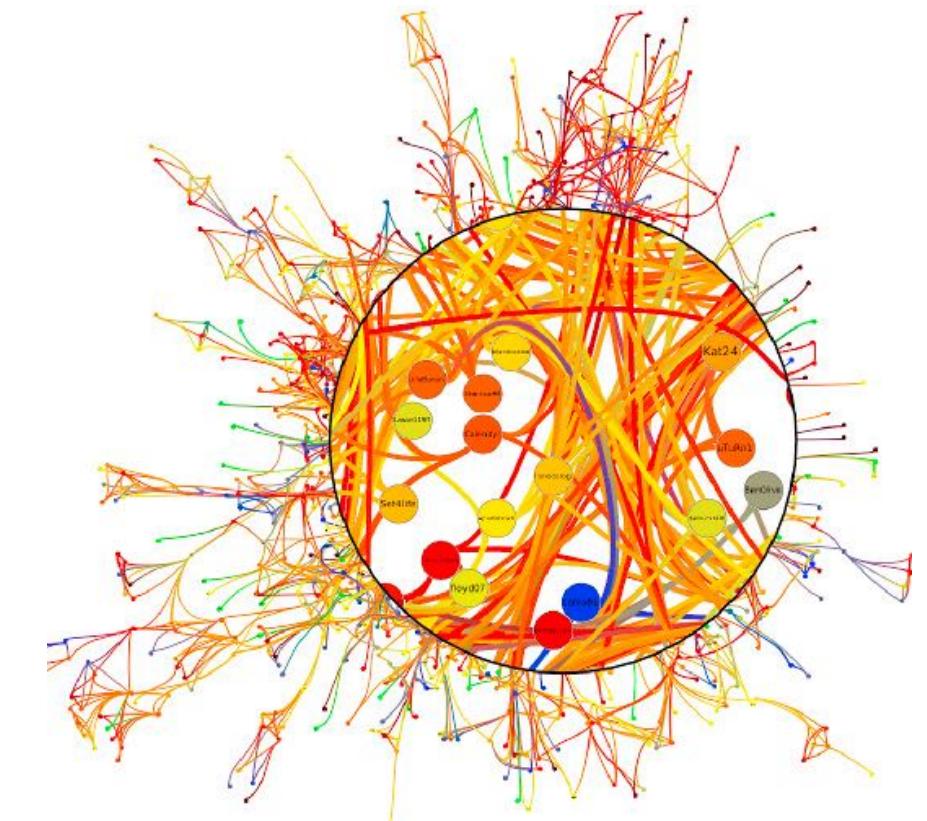
# Distortion costs and benefits

- benefits
  - combine focus and context information in single view
- costs
  - length comparisons impaired
    - topology comparisons unaffected: connection, containment
  - effects of distortion unclear if original structure unfamiliar
  - object constancy/tracking may be impaired

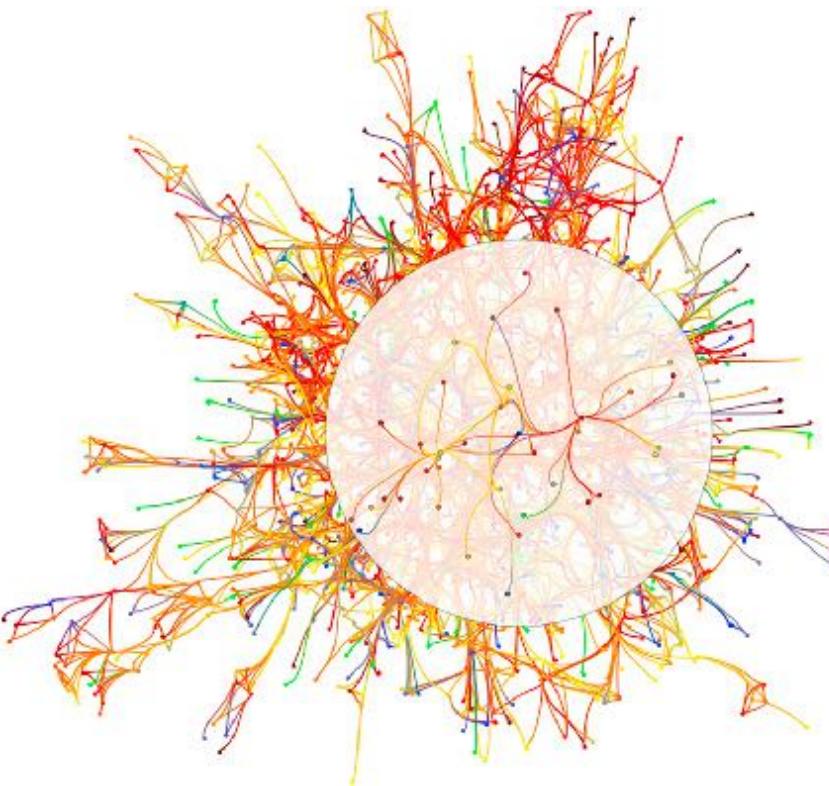
fisheye lens



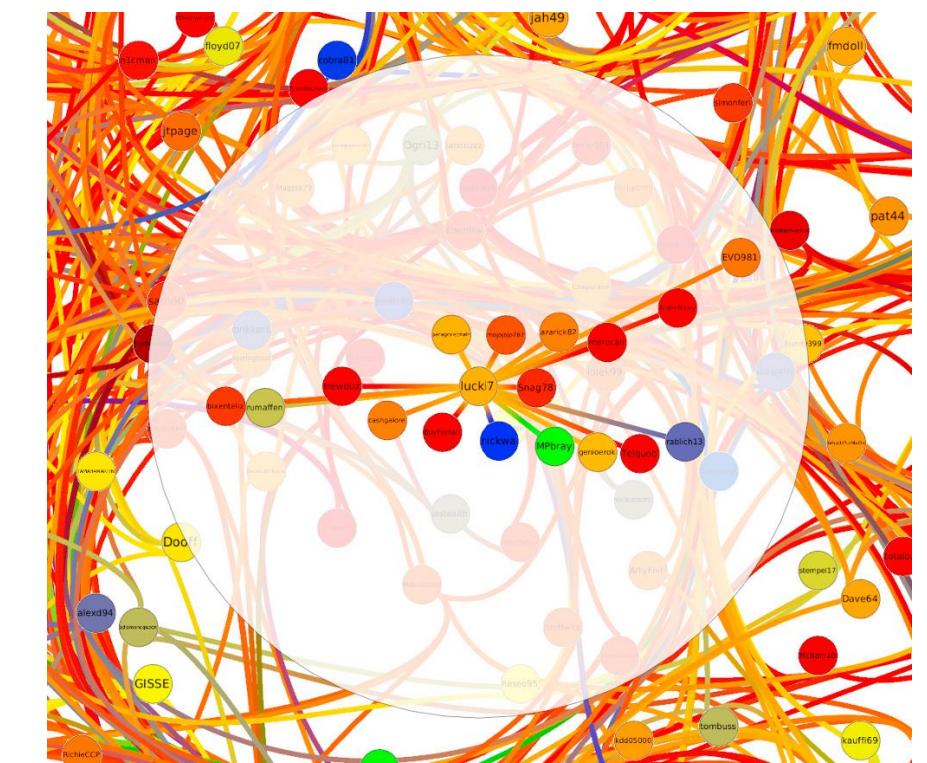
magnifying lens



neighborhood layering



Bring and Go



[Living Flows: Enhanced Exploration of Edge-Bundled Graphs Based on GPU-Intensive Edge Rendering. Lambert, Auber, and Melançon.]

# Questions?

:::::::



@rschifan



rossano.schifanella@unito.it



<http://www.di.unito.it/~schifane>