

Visualization Analysis & Design

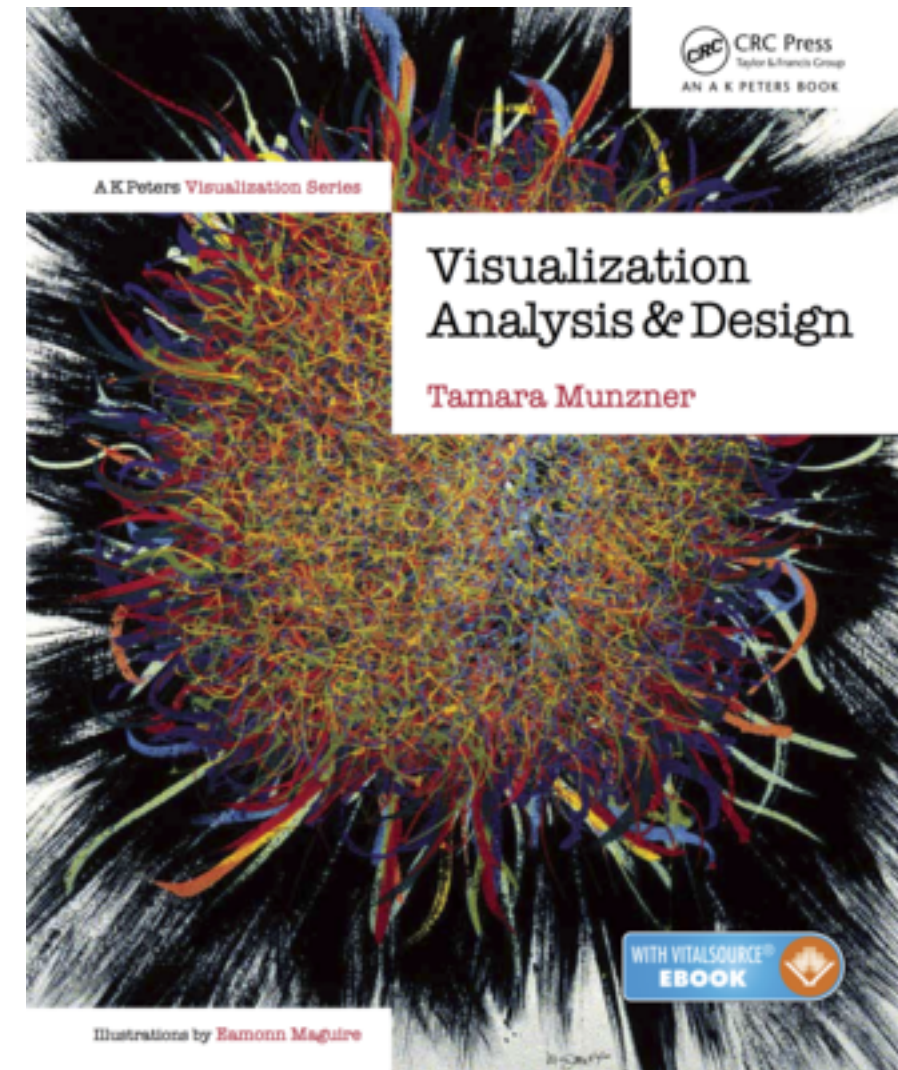
Half-Day Tutorial

Tamara Munzner

Department of Computer Science
University of British Columbia

IEEE VIS 2016 Tutorial
October 2016, Baltimore MD

<http://www.cs.ubc.ca/~tmm/talks.html#halfdaycourse16>



[@tamaramunzner](https://twitter.com/tamaramunzner)

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

Defining visualization (vis)

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Why?...

Why have a human in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

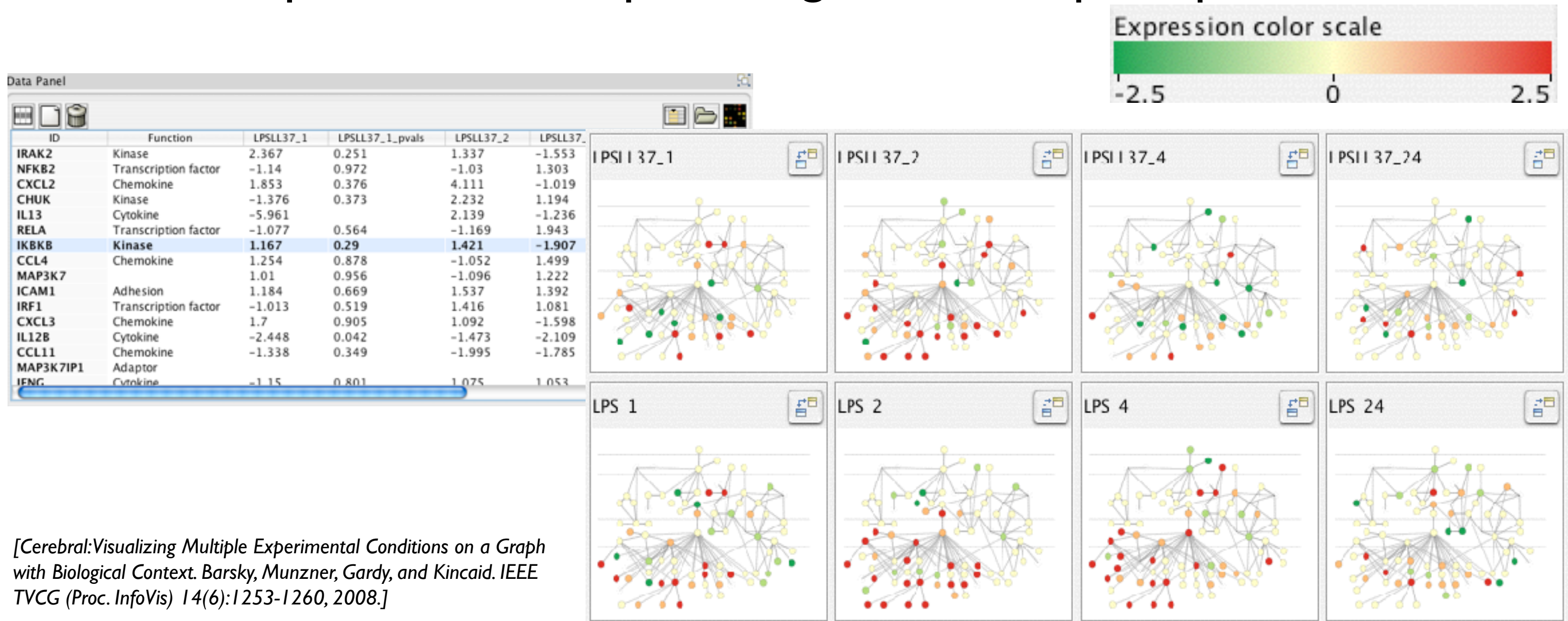
Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.

- don't need vis when fully automatic solution exists and is trusted
- many analysis problems ill-specified
 - don't know exactly what questions to ask in advance
- possibilities
 - long-term use for end users (ex: exploratory analysis of scientific data)
 - presentation of known results (ex: New York Times Upshot)
 - stepping stone to better understanding of requirements before developing models
 - help developers of automatic solution refine/debug, determine parameters
 - help end users of automatic solutions verify, build trust

Why use an external representation?

Computer-based visualization systems provide **visual representations** of datasets designed to help people carry out tasks more effectively.

- external representation: replace cognition with perception



[Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context. Barsky, Munzner, Gardy, and Kincaid. IEEE TVCG (Proc. InfoVis) 14(6):1253-1260, 2008.]

Why represent all the data?

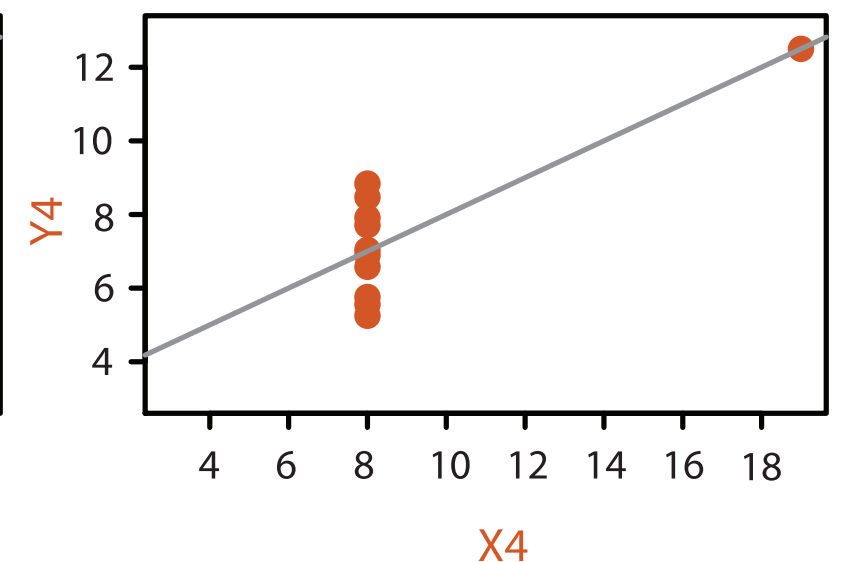
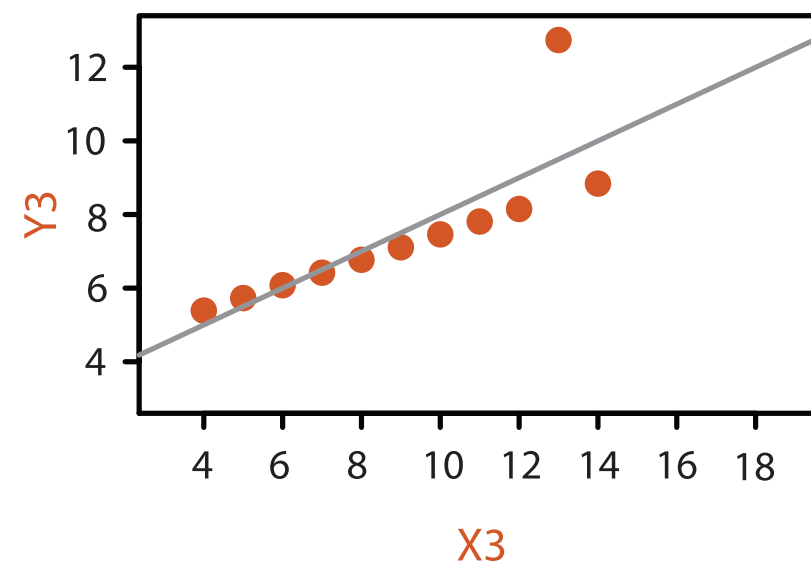
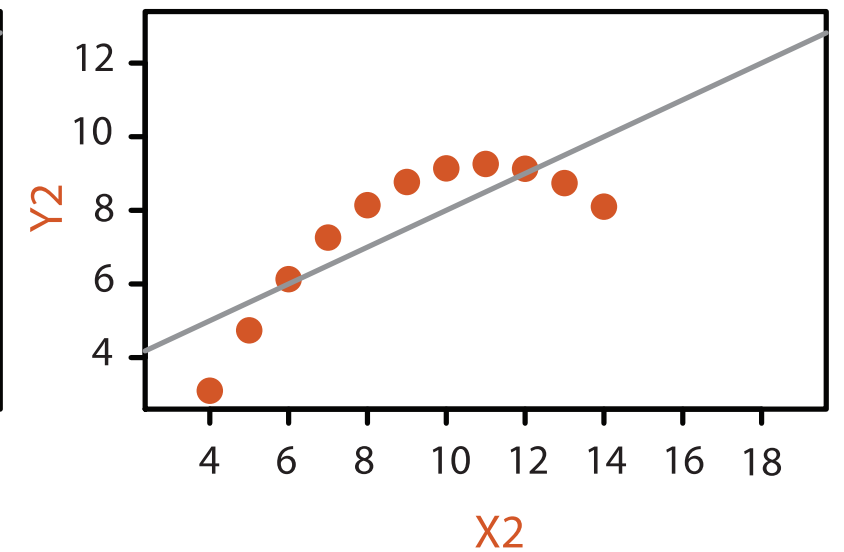
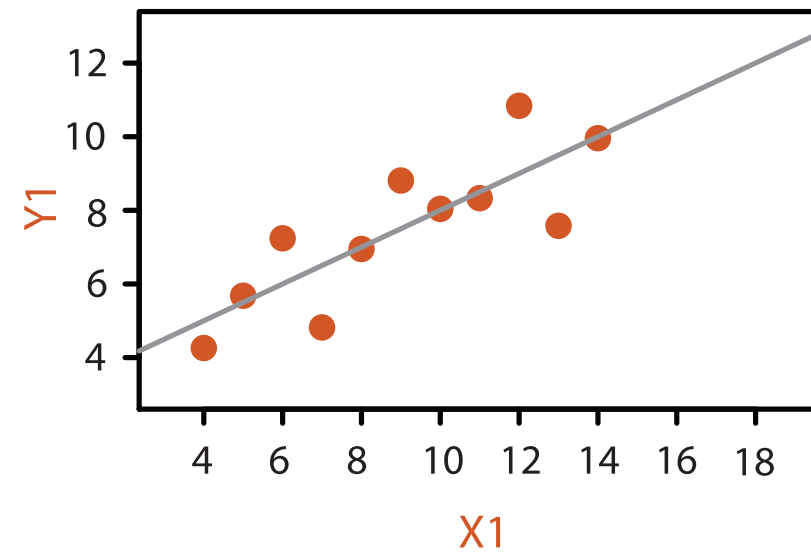
Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- summaries lose information, details matter
 - confirm expected and find unexpected patterns
 - assess validity of statistical model

Anscombe's Quartet

Identical statistics

x mean	9
x variance	10
y mean	7.5
y variance	3.75
x/y correlation	0.816



Why analyze?

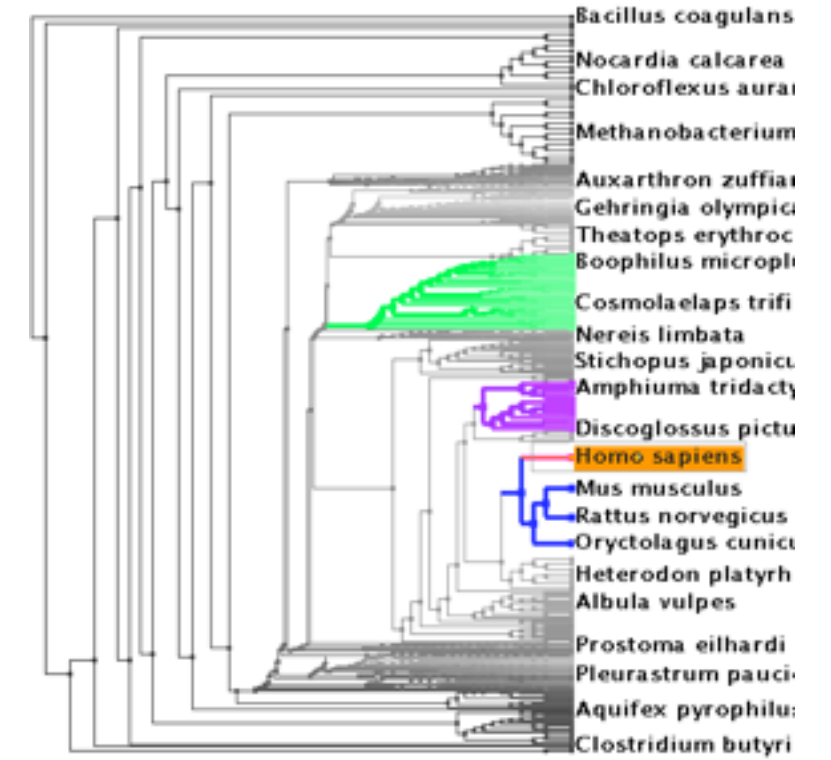
- imposes structure on huge design space
 - scaffold to help you think systematically about choices
 - analyzing existing as stepping stone to designing new
 - most possibilities ineffective for particular task/data combination

SpaceTree



[SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation. Grosjean, Plaisant, and Bederson. Proc. InfoVis 2002, p 57–64.]

TreeJuxtaposer



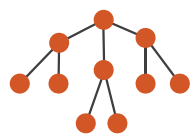
[TreeJuxtaposer: Scalable Tree Comparison Using Focus +Context With Guaranteed Visibility. ACM Trans. on Graphics (Proc. SIGGRAPH) 22:453– 462, 2003.]

What?

Why?

How?

→ Tree



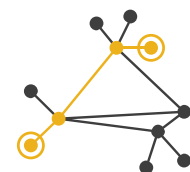
→ Actions

→ Present → Locate → Identify



→ Targets

→ Path between two nodes



→ SpaceTree

→ Encode → Navigate → Select → Filter → Aggregate



→ TreeJuxtaposer

→ Encode → Navigate → Select → Arrange



What?

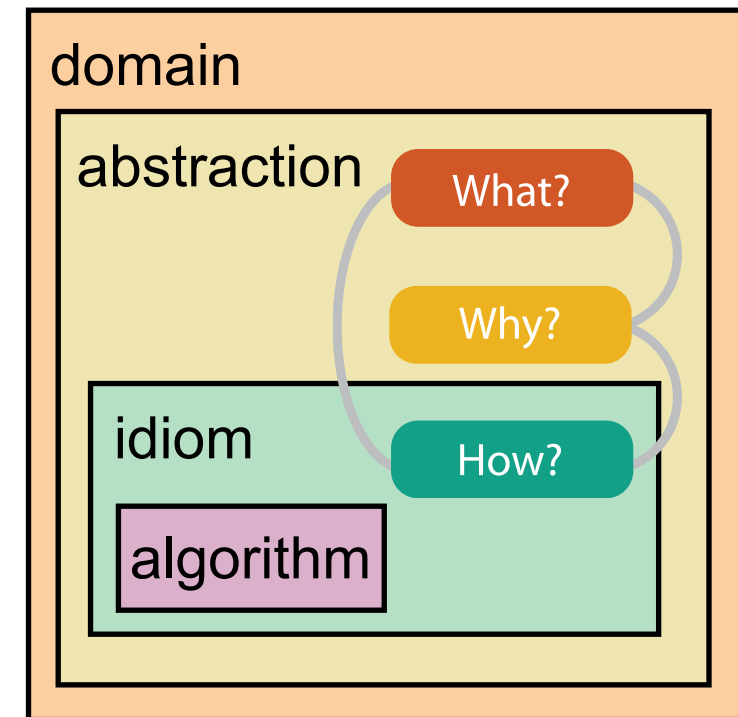
Why?

How?

Analysis framework: Four levels, three questions

- *domain situation*
 - who are the target users?
- *abstraction*
 - translate from specifics of domain to vocabulary of vis
 - **what** is shown? **data abstraction**
 - **why** is the user looking at it? **task abstraction**
- *idiom*
 - **how** is it shown?
 - **visual encoding idiom**: how to draw
 - **interaction idiom**: how to manipulate
- *algorithm*
 - efficient computation

[A Nested Model of Visualization Design and Validation.
Munzner. *IEEE TVCG* 15(6):921-928, 2009
(Proc. InfoVis 2009).]



[A Multi-Level Typology of Abstract Visualization Tasks
Brehmer and Munzner. *IEEE TVCG* 19(12):2376-2385,
2013 (Proc. InfoVis 2013).]

Why is validation difficult?

- different ways to get it wrong at each level



Domain situation

You misunderstood their needs



Data/task abstraction

You're showing them the wrong thing



Visual encoding/interaction idiom

The way you show it doesn't work

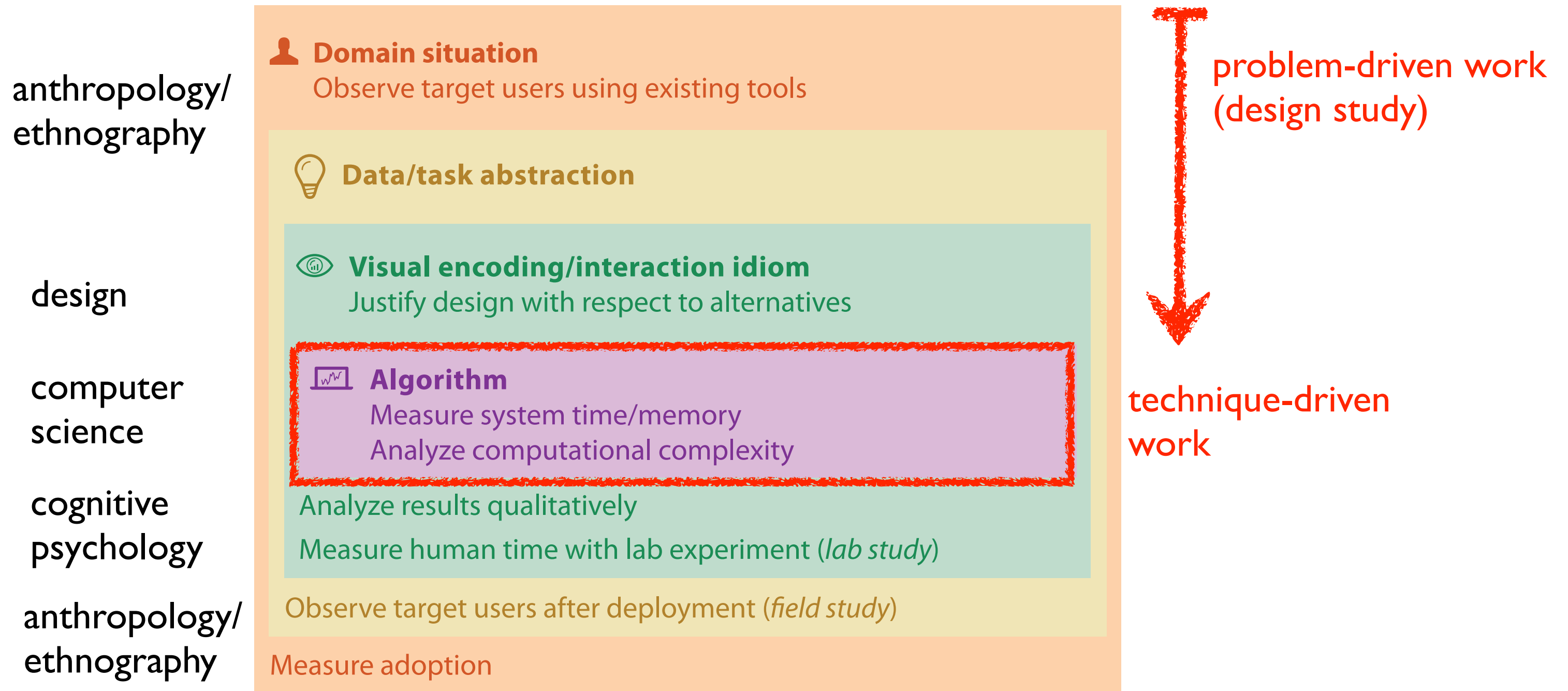


Algorithm

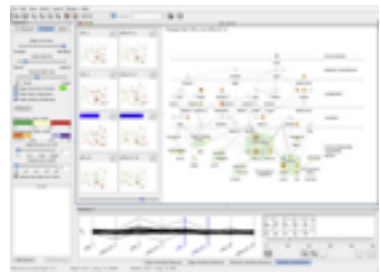
Your code is too slow

Why is validation difficult?

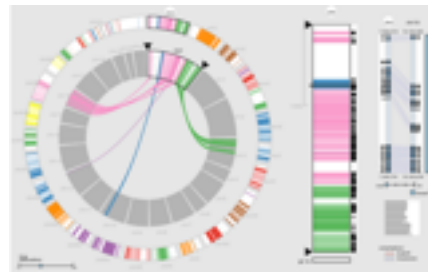
- solution: use methods from different fields at each level



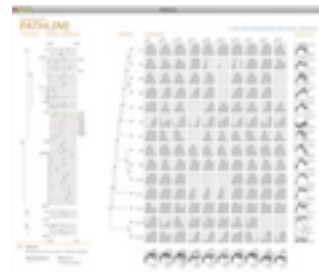
Design Studies: Lessons learned after 21 of them



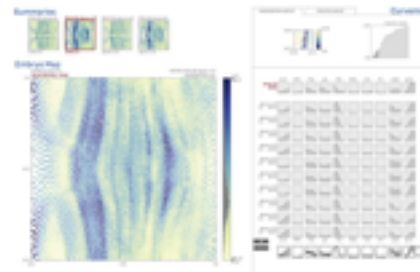
Cerebral
genomics



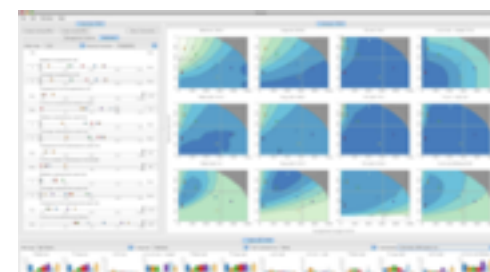
MizBee
genomics



Pathline
genomics



MulteeSum
genomics



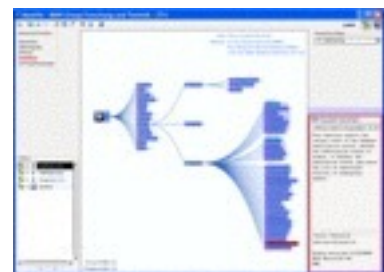
Vismon
fisheries management



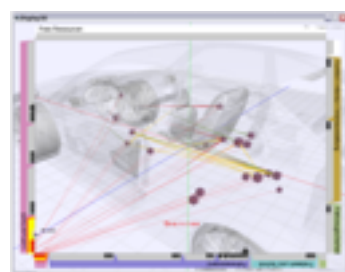
QuestVis
sustainability



WiKeVis
in-car networks



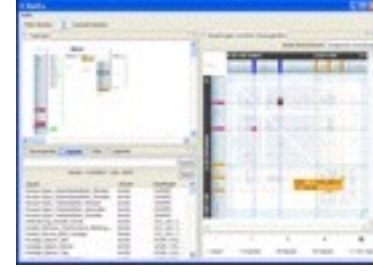
MostVis
in-car networks



Car-X-Ray
in-car networks



ProgSpy2010
in-car networks



ReEx
in-car networks



Cardiogram
in-car networks



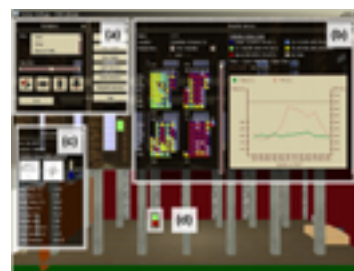
AutobahnVis
in-car networks



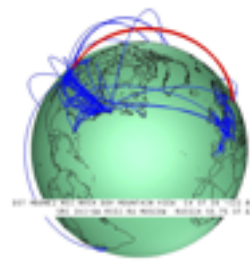
VisTra
in-car networks



Constellation
linguistics



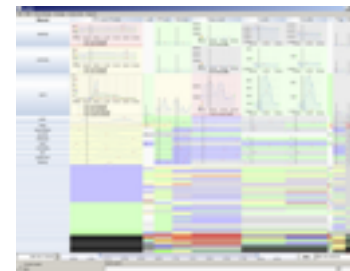
LibVis
cultural heritage



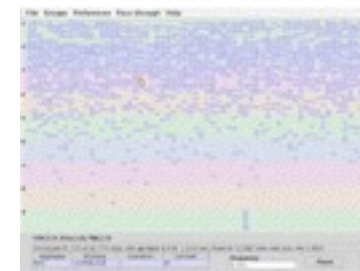
Caidants
multicast



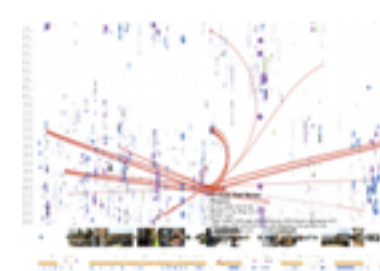
SessionViewer
web log analysis



LiveRAC
server hosting



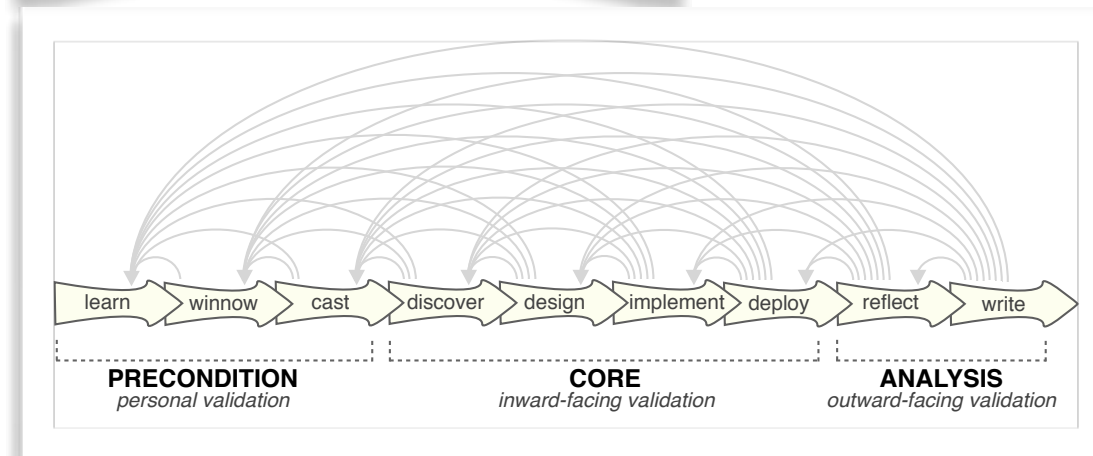
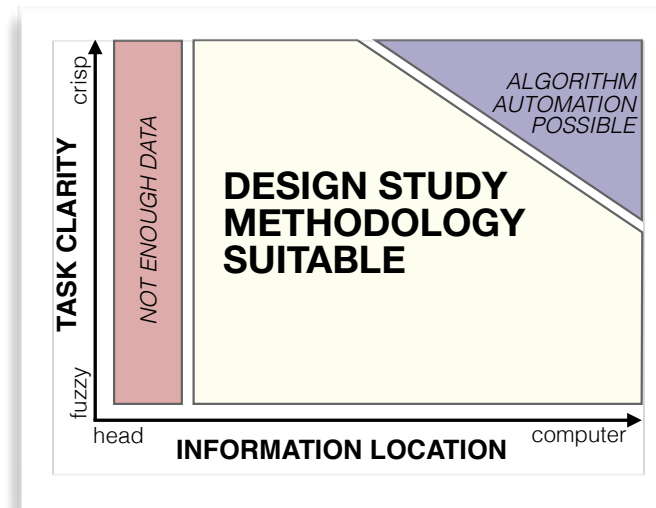
PowerSetViewer
data mining



LastHistory
music listening

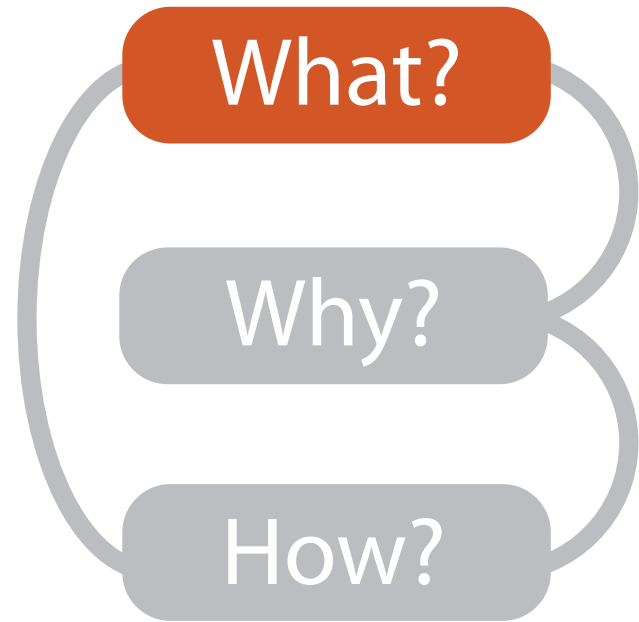
Design Study Methodology: Reflections from the Trenches and the Stacks

- definitions
- 9-stage framework
- 32 pitfalls
and how to avoid them



PF-1	premature advance: jumping forward over stages	general
PF-2	premature start: insufficient knowledge of vis literature	learn
PF-3	premature commitment: collaboration with wrong people	winnow
PF-4	no real data available (yet)	winnow
PF-5	insufficient time available from potential collaborators	winnow
PF-6	no need for visualization: problem can be automated	winnow
PF-7	researcher expertise does not match domain problem	winnow
PF-8	no need for research: engineering vs. research project	winnow
PF-9	no need for change: existing tools are good enough	winnow

[Sedlmair, Meyer, Munzner.
*IEEE Trans. Visualization and
Computer Graphics*
18(12): 2431-2440, 2012
(Proc. InfoVis 2012).]



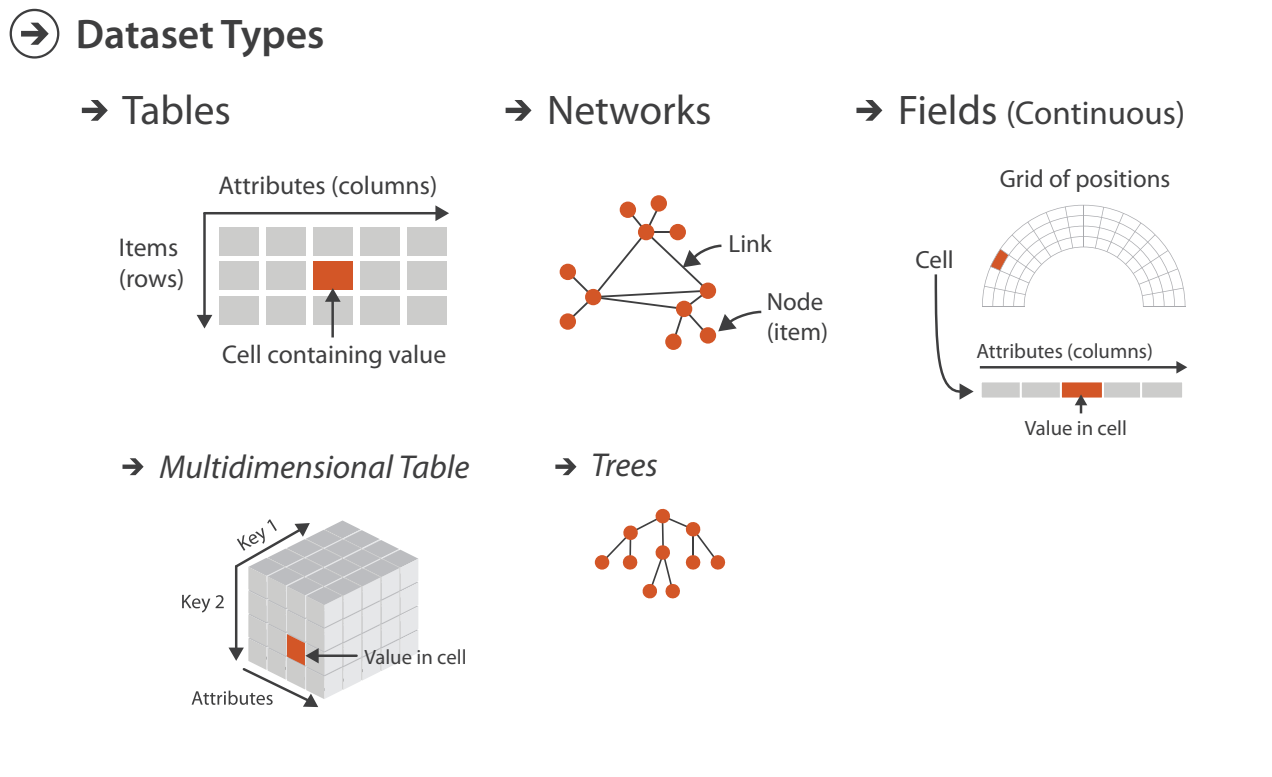
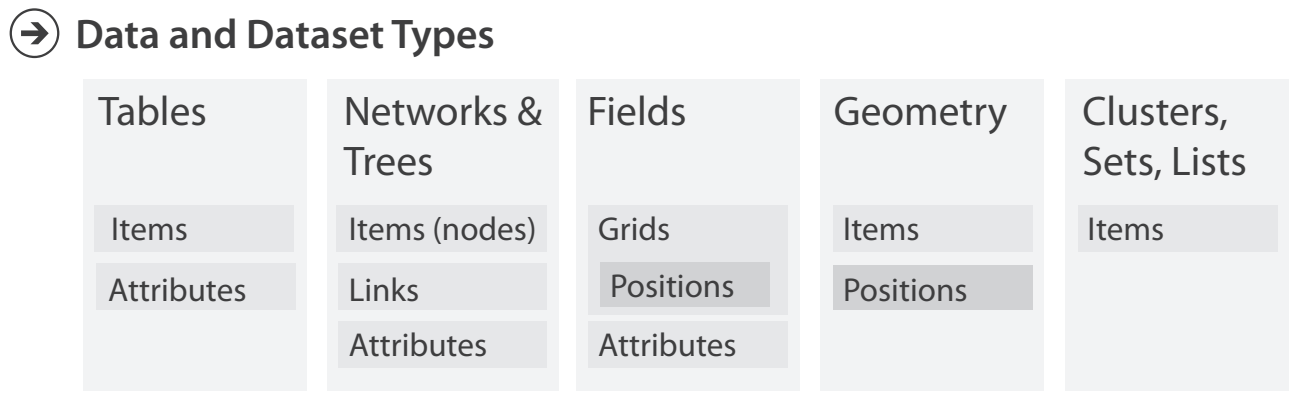
What?

Datasets

Attributes

- Data Types
 - Items
 - Attributes
 - Links
 - Positions
 - Grids

- Attribute Types
 - Categorical
 - + ● ■ ▲
 - Ordered
 - Ordinal
 - 👕 👕 👕
 - Quantitative
 - ┆ ┆ ┆

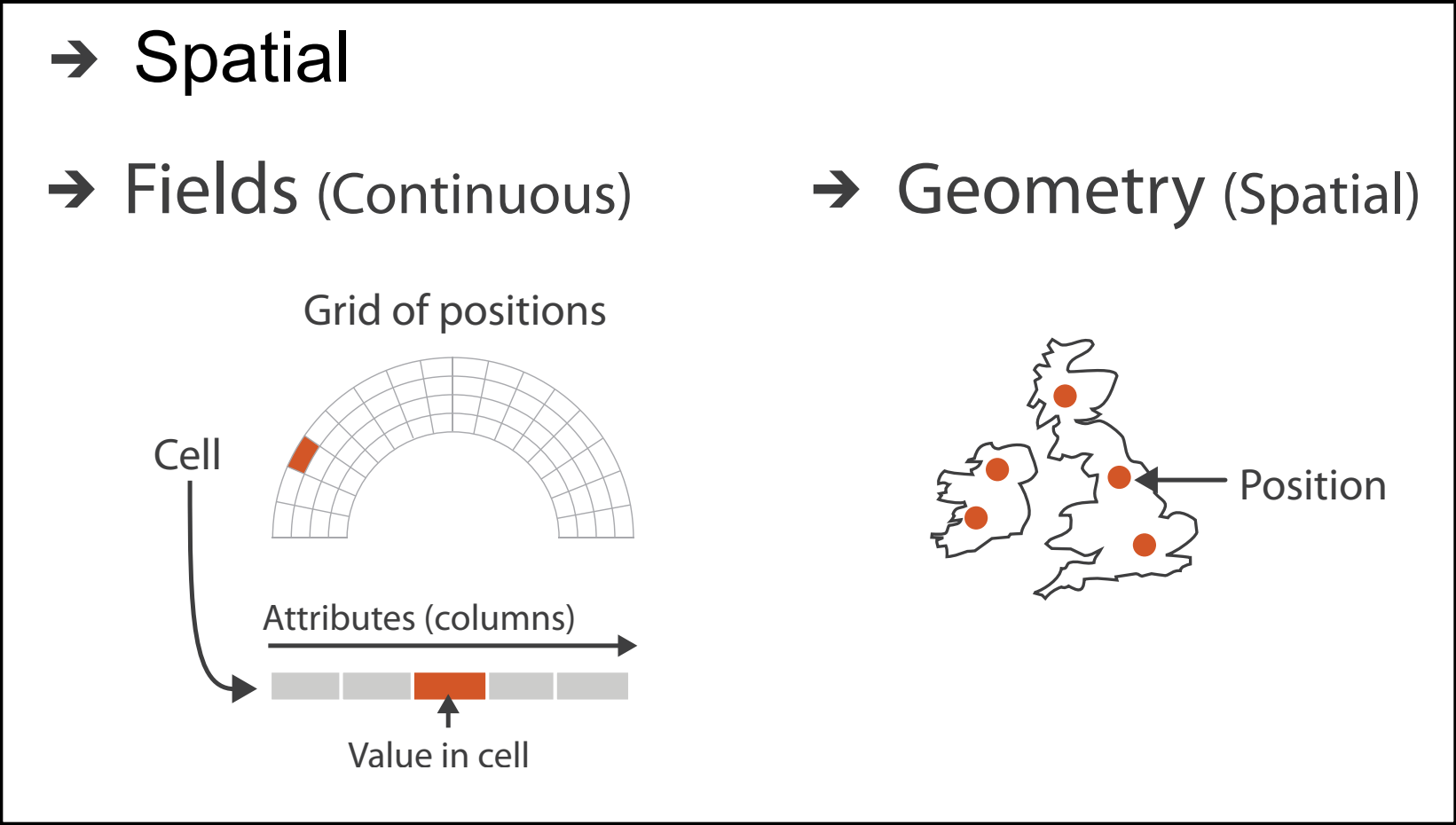
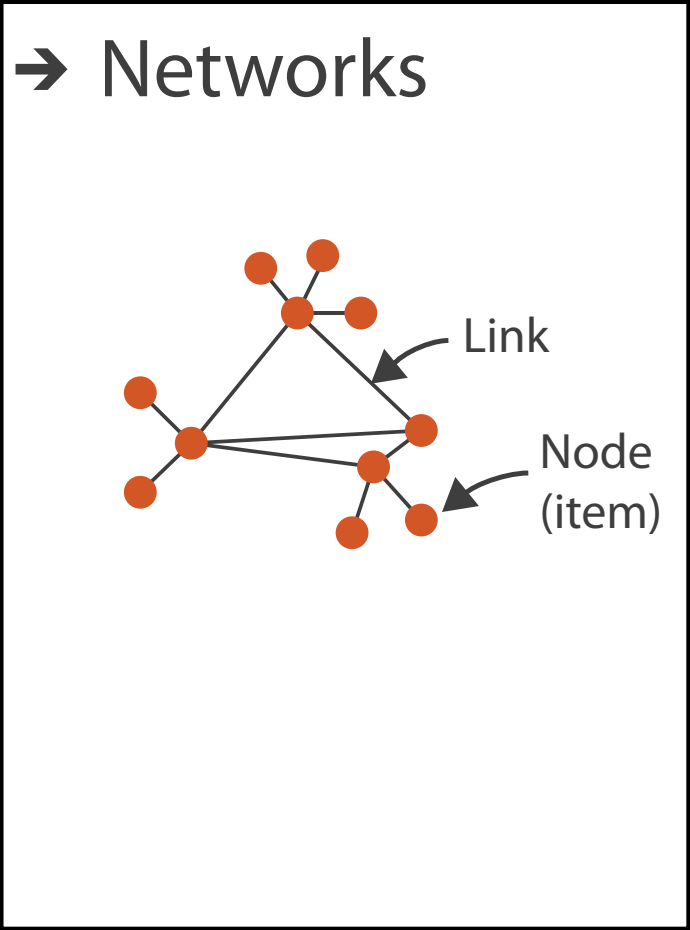
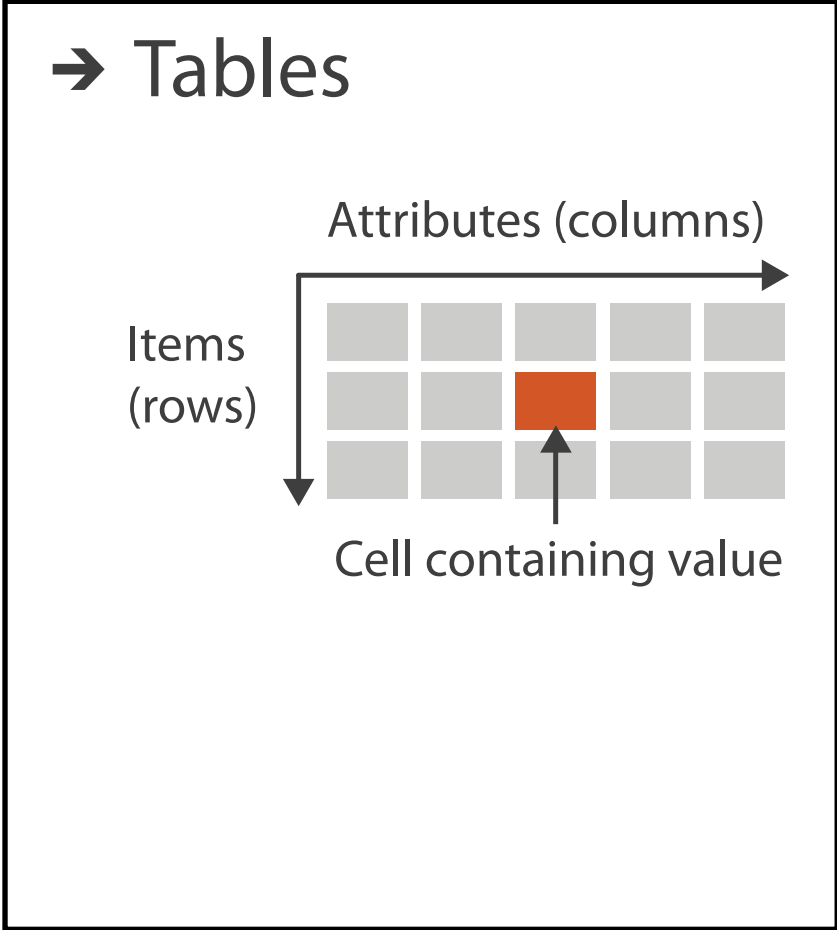


- Ordering Direction
 - Sequential
 -
 - Diverging
 - ←→
 - Cyclic
 - ↻



Three major datatypes

→ Dataset Types



- visualization vs computer graphics
 - geometry is design decision

Attribute types

➔ Attribute Types

➔ Categorical

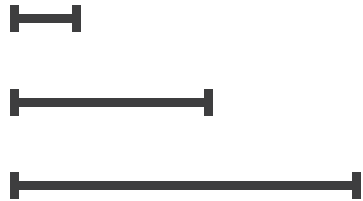


➔ Ordered

➔ *Ordinal*



➔ *Quantitative*



➔ Ordering Direction

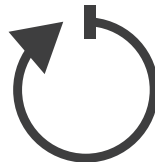
➔ Sequential

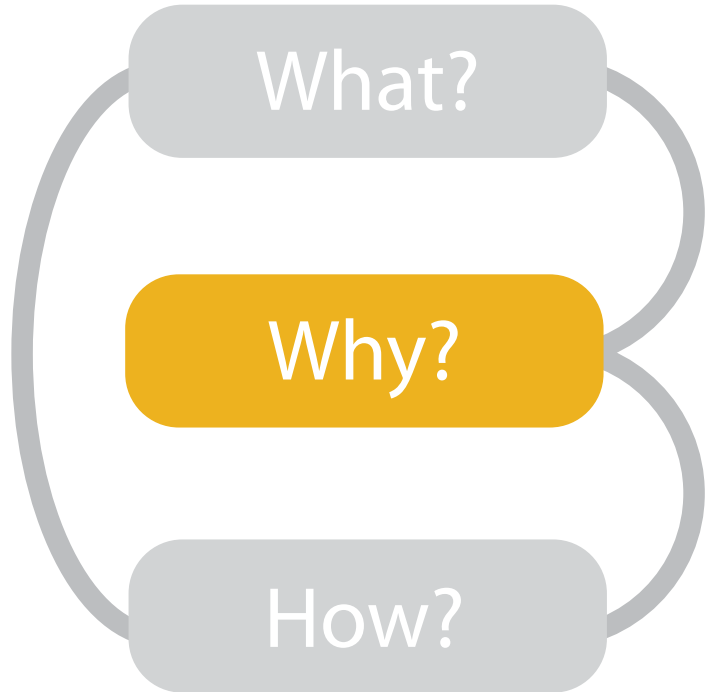


➔ Diverging



➔ Cyclic





👉 Actions

🎯 Targets

➔ **Analyze**

- ➔ Consume
 - ➔ Discover
 - ➔ Present
 - ➔ Enjoy
- ➔ Produce
 - ➔ Annotate
 - ➔ Record
 - ➔ Derive

➔ **All Data**

- ➔ Trends
- ➔ Outliers
- ➔ Features

➔ **Attributes**

- ➔ One
 - ➔ Distribution
 - ➔ Extremes
- ➔ Many
 - ➔ Dependency
 - ➔ Correlation
 - ➔ Similarity

➔ **Search**

	Target known	Target unknown
Location known	Lookup	Browse
Location unknown	Locate	Explore

➔ **Query**

- ➔ Identify
- ➔ Compare
- ➔ Summarize

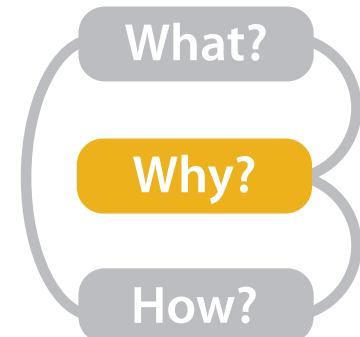
➔ **Network Data**

- ➔ Topology
- ➔ Paths

➔ **Spatial Data**

- ➔ Shape

- {action, target} pairs
 - discover distribution
 - compare trends
 - locate outliers
 - browse topology



Actions: Analyze, Query

- analyze

- consume

- discover vs present

- aka explore vs explain

- enjoy

- aka casual, social

- produce

- annotate, record, derive

- query

- how much data matters?

- one, some, all

- independent choices

- analyze, query, (search)

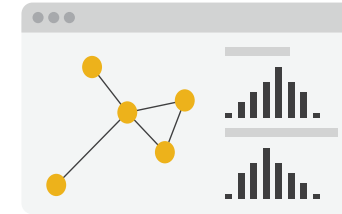
→ Analyze

- Consume

- Discover



- Present

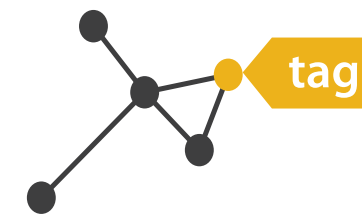


- Enjoy



- Produce

- Annotate



- Record

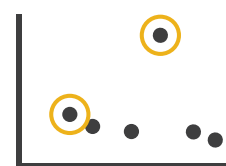


- Derive

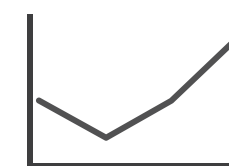


→ Query

- Identify



- Compare

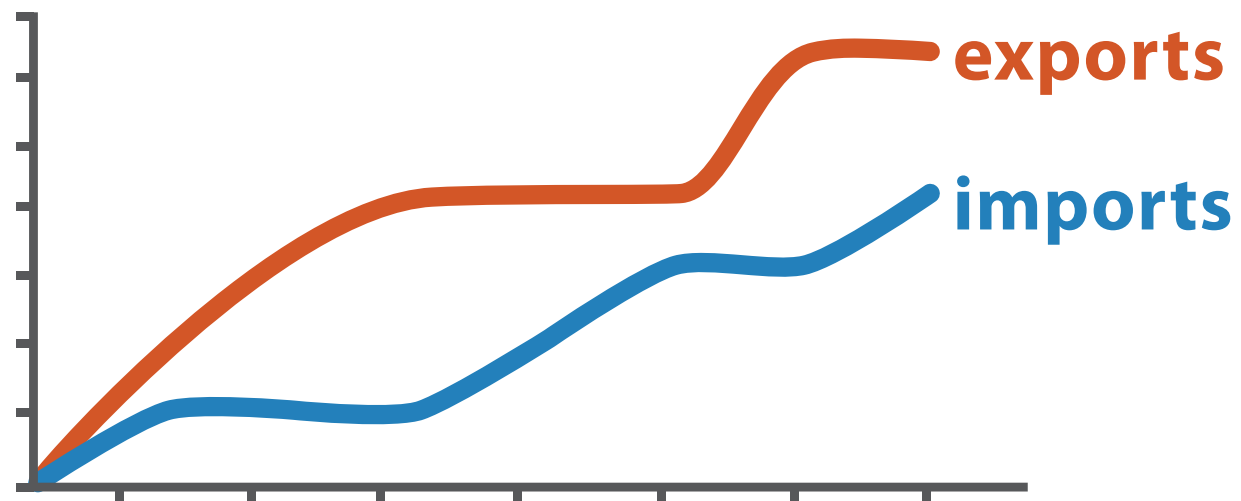


- Summarize

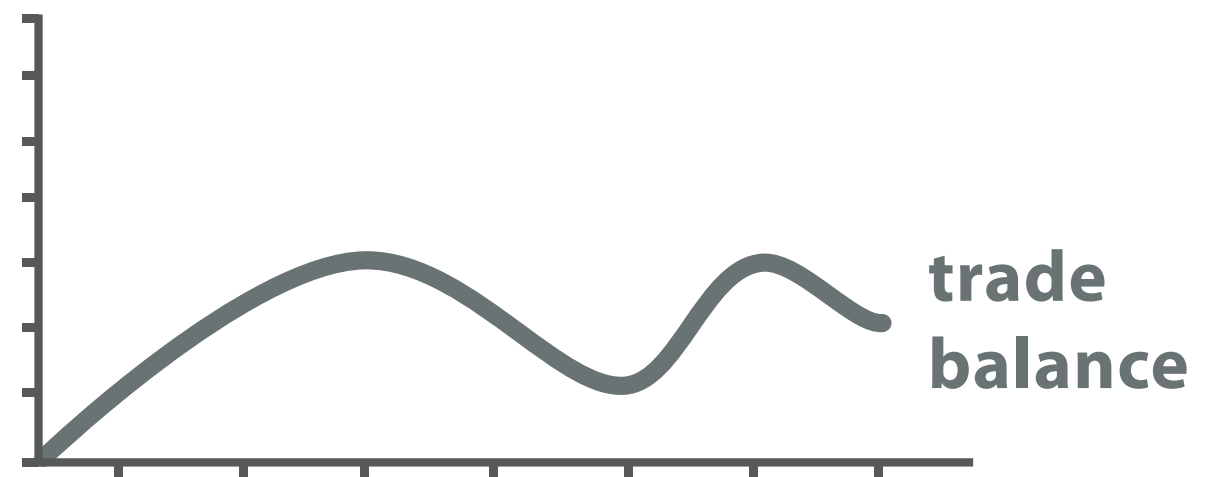


Derive

- don't necessarily just draw what you're given!
 - decide what the right thing to show is
 - create it with a series of transformations from the original dataset
 - draw that
- one of the four major strategies for handling complexity



Original Data



$$\text{trade balance} = \text{exports} - \text{imports}$$

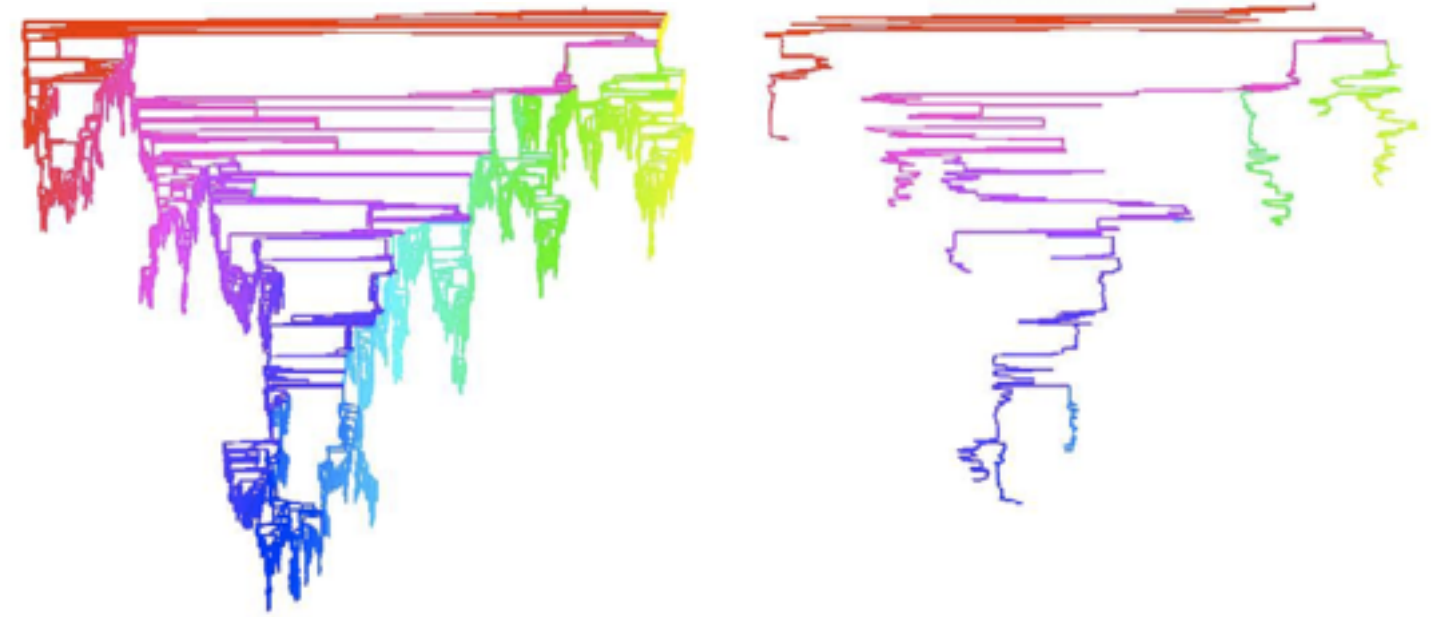
Derived Data

Analysis example: Derive one attribute

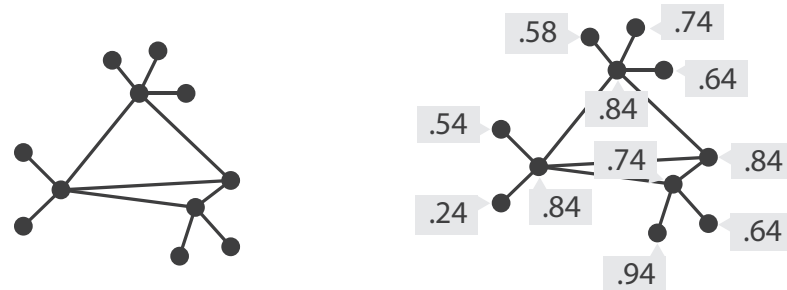
- Strahler number

- centrality metric for trees/networks
- derived quantitative attribute
- draw top 5K of 500K for good skeleton

[Using Strahler numbers for real time visual exploration of huge graphs. Auber. Proc. Intl. Conf. Computer Vision and Graphics, pp. 56–69, 2002.]



Task 1



In
Tree

➔

Out
Quantitative
attribute on nodes

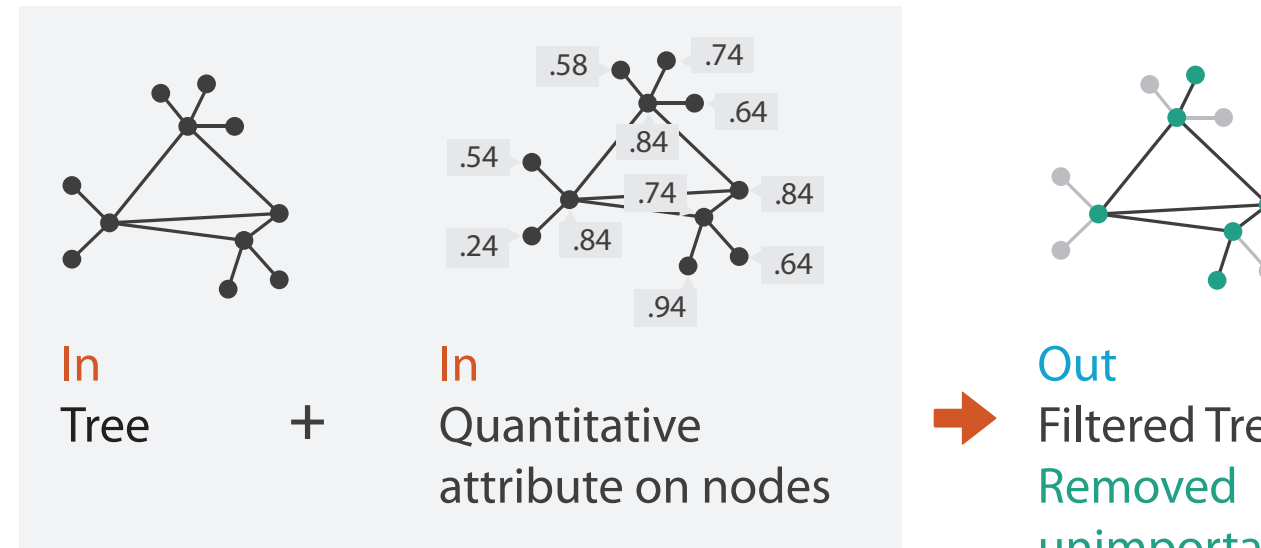
What?

- ➔ In Tree
- ➔ Out Quantitative attribute on nodes

Why?

- ➔ Derive

Task 2



In
Tree

+

In
Quantitative
attribute on nodes

➔

Out
Filtered Tree
Removed
unimportant parts

What?

- ➔ In Tree
- ➔ In Quantitative attribute on nodes
- ➔ Out Filtered Tree

Why?

- ➔ Summarize
- ➔ Topology

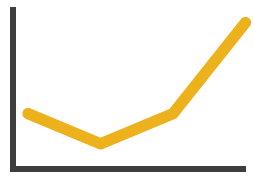
How?

- ➔ Reduce
- ➔ Filter

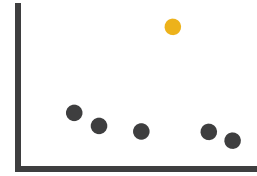
Why: Targets

→ All Data

→ Trends



→ Outliers



→ Features



→ Attributes

→ One

→ *Distribution*



→ *Extremes*

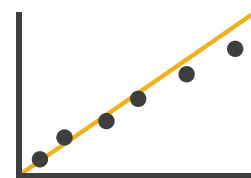


→ Many

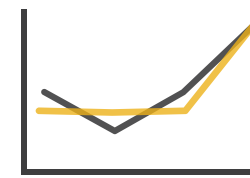
→ *Dependency*



→ *Correlation*

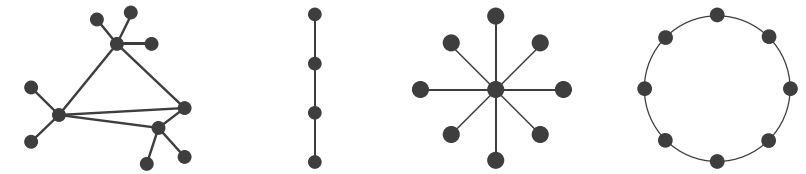


→ *Similarity*



→ Network Data

→ Topology

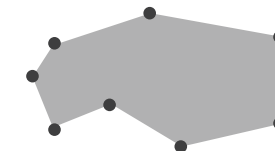


→ *Paths*



→ Spatial Data

→ Shape



How?

Encode

→ Arrange

→ Express



→ Separate



→ Order



→ Align



→ Use



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape



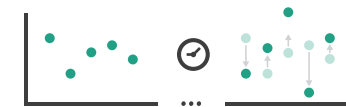
→ Motion

Direction, Rate, Frequency, ...

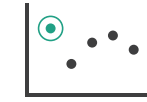


Manipulate

→ Change



→ Select



→ Navigate

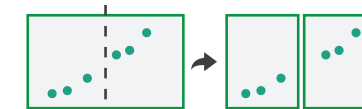


Facet

→ Juxtapose



→ Partition



→ Superimpose



Reduce

→ Filter



→ Aggregate



→ Embed



What?

Why?

How?

Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
 - *Chap 1: What's Vis, and Why Do It?*
 - *Chap 2: What: Data Abstraction*
 - *Chap 3: Why: Task Abstraction*
- *A Multi-Level Typology of Abstract Visualization Tasks*. Brehmer and Munzner. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis) 19:12 (2013), 2376–2385.
- *Low-Level Components of Analytic Activity in Information Visualization*. Amar, Eagan, and Stasko. Proc. IEEE InfoVis 2005, p 111–117.
- *A taxonomy of tools that support the fluent and flexible use of visualizations*. Heer and Shneiderman. Communications of the ACM 55:4 (2012), 45–54.
- *Rethinking Visualization: A High-Level Taxonomy*. Tory and Möller. Proc. IEEE InfoVis 2004, p 151–158.
- Visualization of Time-Oriented Data. Aigner, Miksch, Schumann, and Tominski. Springer, 2011.

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - **Marks and Channels**
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

How?

Encode

→ Arrange

→ Express



→ Separate



→ Order



→ Align



→ Use



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape



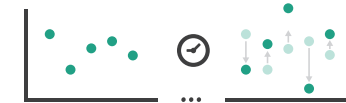
→ Motion

Direction, Rate, Frequency, ...

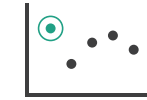


Manipulate

→ Change



→ Select



→ Navigate

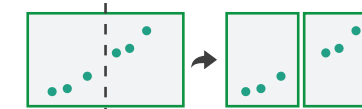


Facet

→ Juxtapose



→ Partition



→ Superimpose



Reduce

→ Filter



→ Aggregate



→ Embed



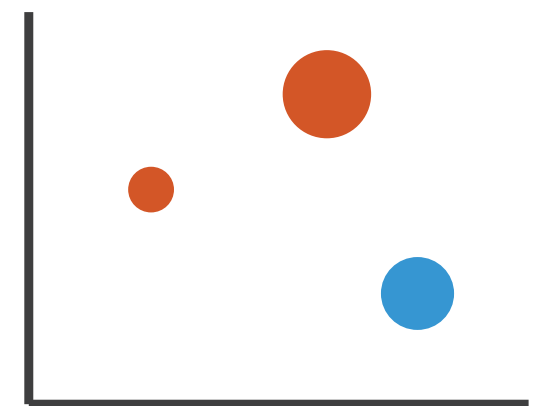
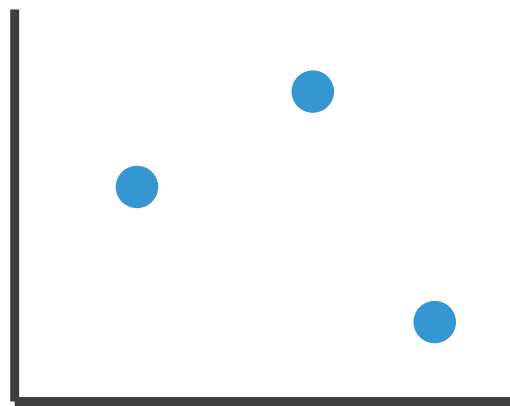
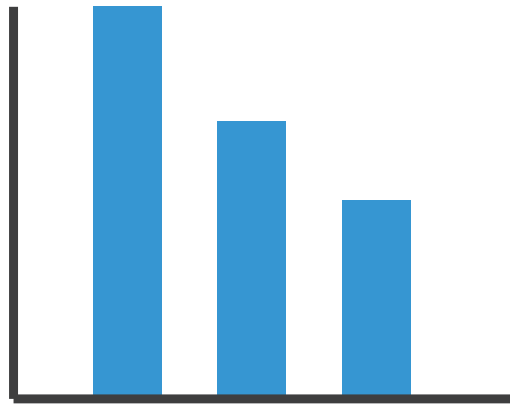
What?

Why?

How?

Visual encoding

- analyze idiom structure



Definitions: Marks and channels

- marks

- geometric primitives

→ Points



→ Lines



→ Areas



- channels

- control appearance of marks

→ Position

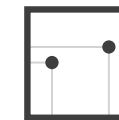
→ Horizontal



→ Vertical



→ Both



→ Color



- channel properties differ

- type & amount of information that can be conveyed to human perceptual system

- show magnitude vs. identity

- accuracy of perception

- number of discriminable bins

→ Shape



→ Tilt



→ Size

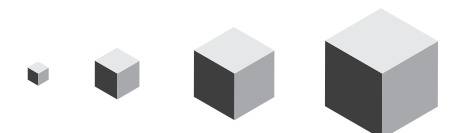
→ Length



→ Area

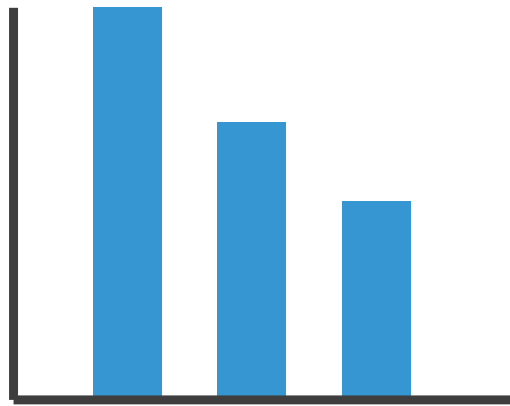


→ Volume



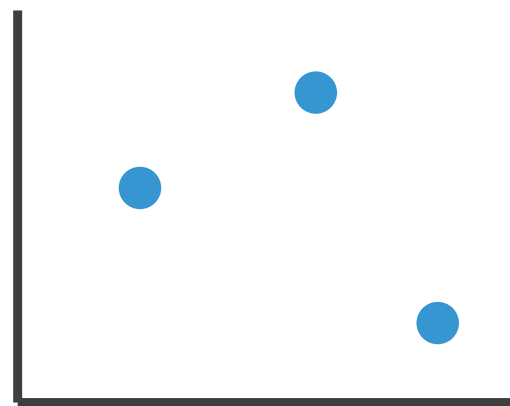
Visual encoding

- analyze idiom structure
 - as combination of marks and channels



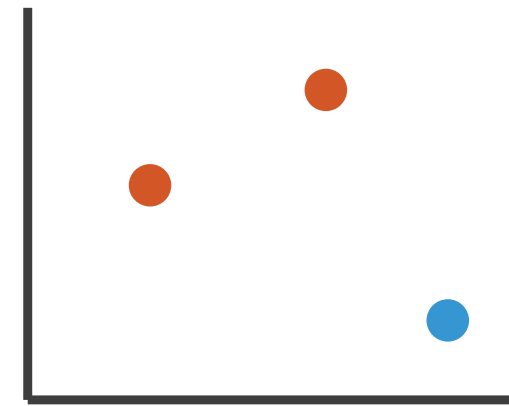
1:
vertical position

mark: line



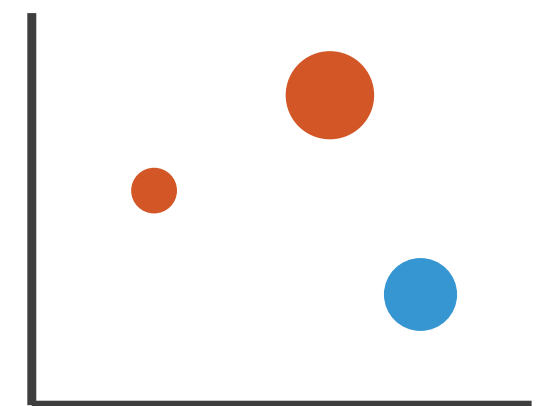
2:
vertical position
horizontal position

mark: point



3:
vertical position
horizontal position
color hue

mark: point



4:
vertical position
horizontal position
color hue
size (area)

mark: point

Channels

Position on common scale



Position on unaligned scale



Length (1D size)



Tilt/angle



Area (2D size)



Depth (3D position)



Color luminance



Color saturation



Curvature



Volume (3D size)



Same

Spatial region



Color hue



Motion





Shape




Channels: Matching expressiveness


➔ Magnitude Channels: Ordered Attributes

Position on common scale 


Position on unaligned scale 


Length (1D size) 

Tilt/angle 


Area (2D size) 

Depth (3D position) 

Color luminance 

Color saturation 

Curvature 

Volume (3D size) 

Same

Same

➔ Identity Channels: Categorical Attributes

Spatial region 

Color hue 

Motion 

Shape 

- expressiveness principle
 - match channel and data characteristics

➔ Attribute Types

➔ Categorical

➔ Ordered



➔ Ordinal

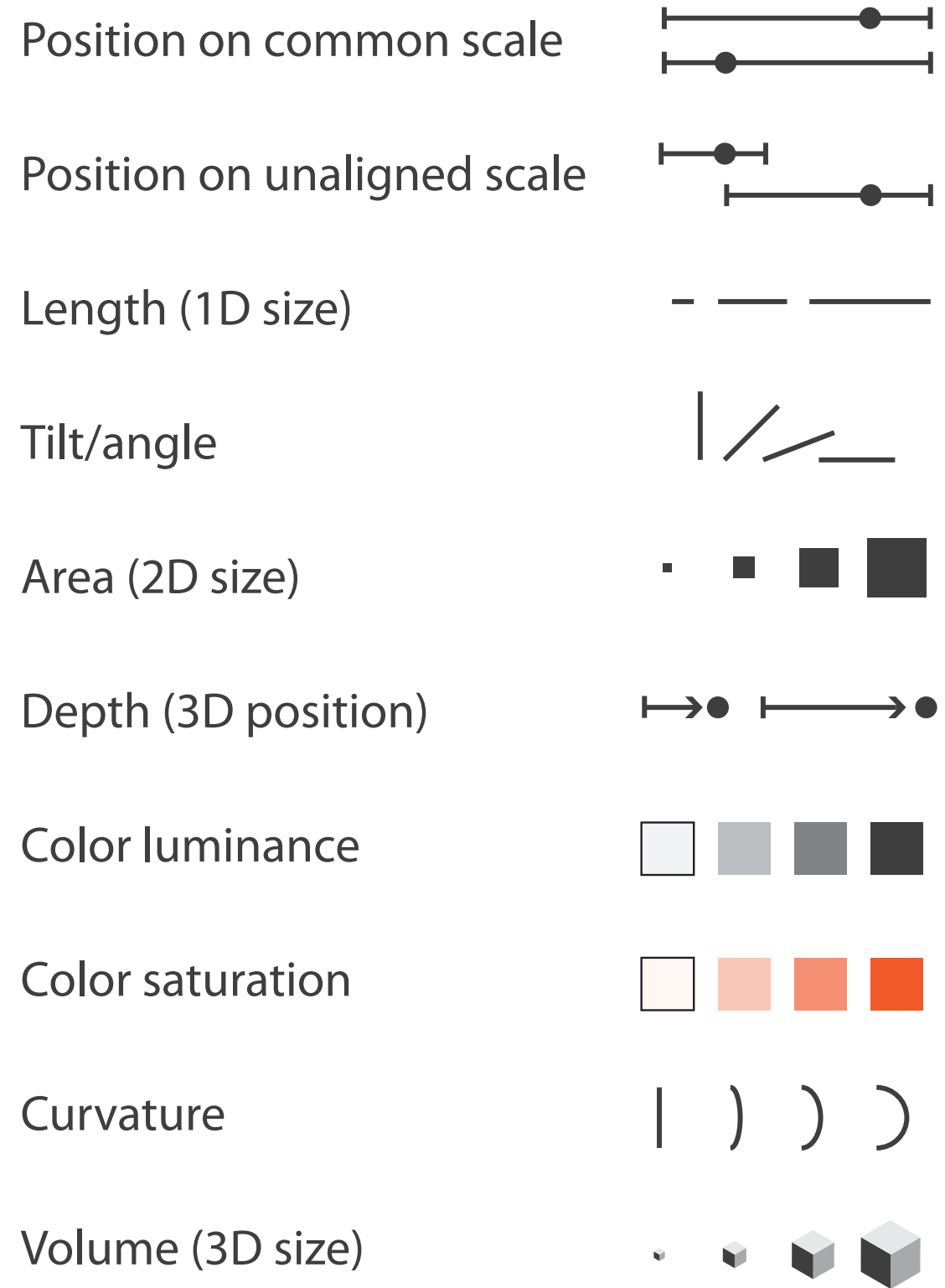


➔ Quantitative



Channels: Ranking effectiveness

➔ Magnitude Channels: Ordered Attributes



➔ Identity Channels: Categorical Attributes



- **expressiveness principle**
 - match channel and data characteristics
- **effectiveness principle**
 - encode most important attributes with highest ranked channels

Channels: Ranking effectiveness

➔ Magnitude Channels: Ordered Attributes

Position on common scale



Position on unaligned scale



Length (1D size)



Tilt/angle



Area (2D size)



Depth (3D position)



Color luminance



Color saturation



Curvature



Volume (3D size)



Same

➔ Identity Channels: Categorical Attributes

Spatial region



Color hue



Motion



Shape



Best

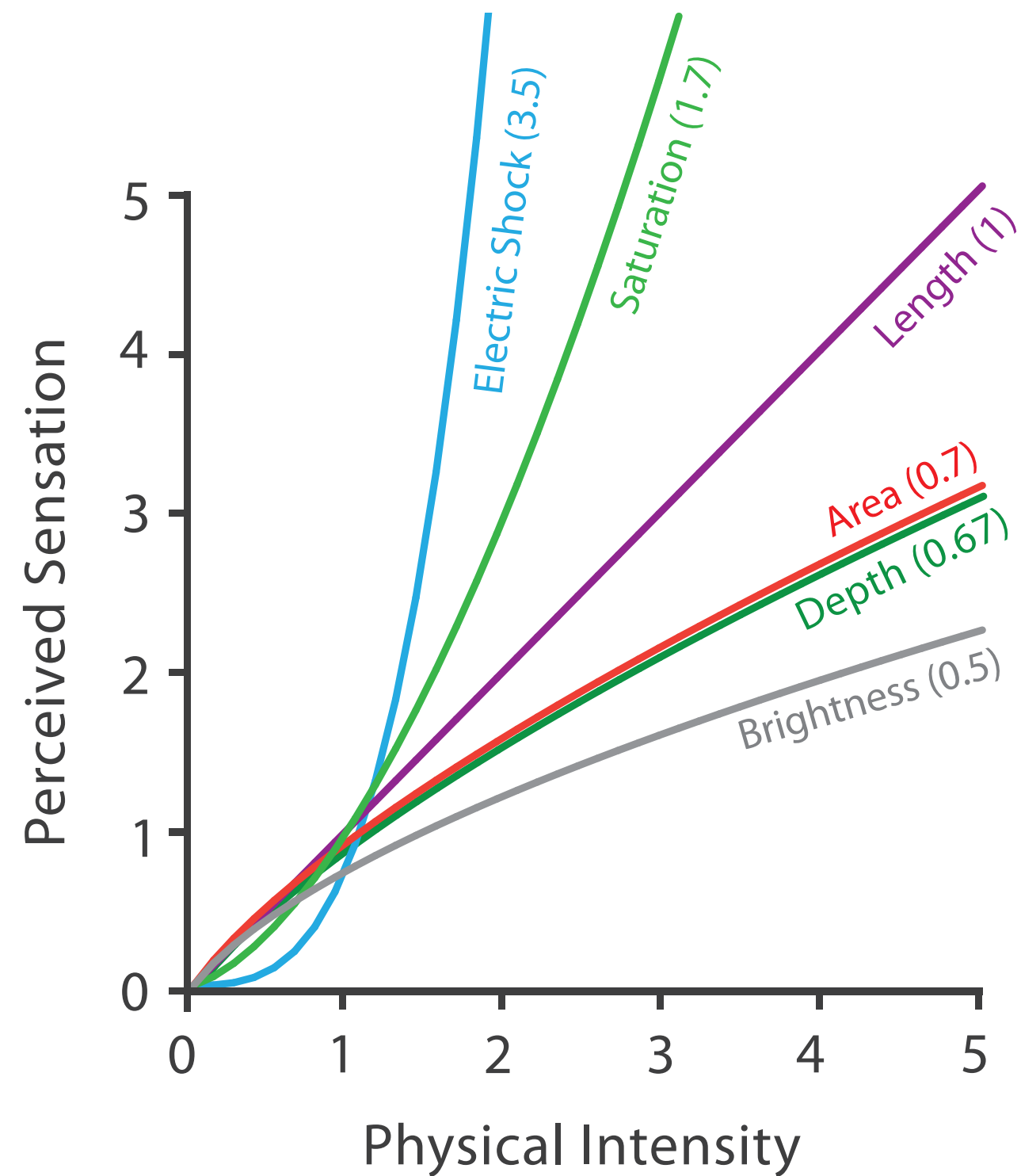
Effectiveness

Least

- **expressiveness principle**
 - match channel and data characteristics
- **effectiveness principle**
 - encode most important attributes with highest ranked channels
 - spatial position ranks high for both

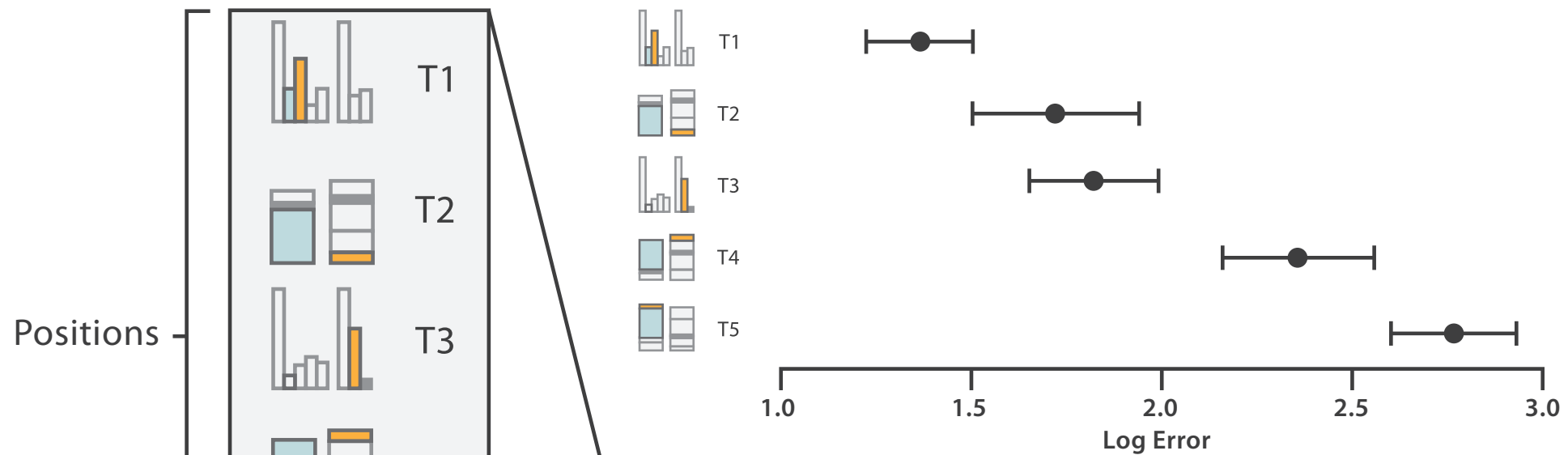
Accuracy: Fundamental Theory

Steven's Psychophysical Power Law: $S = I^N$

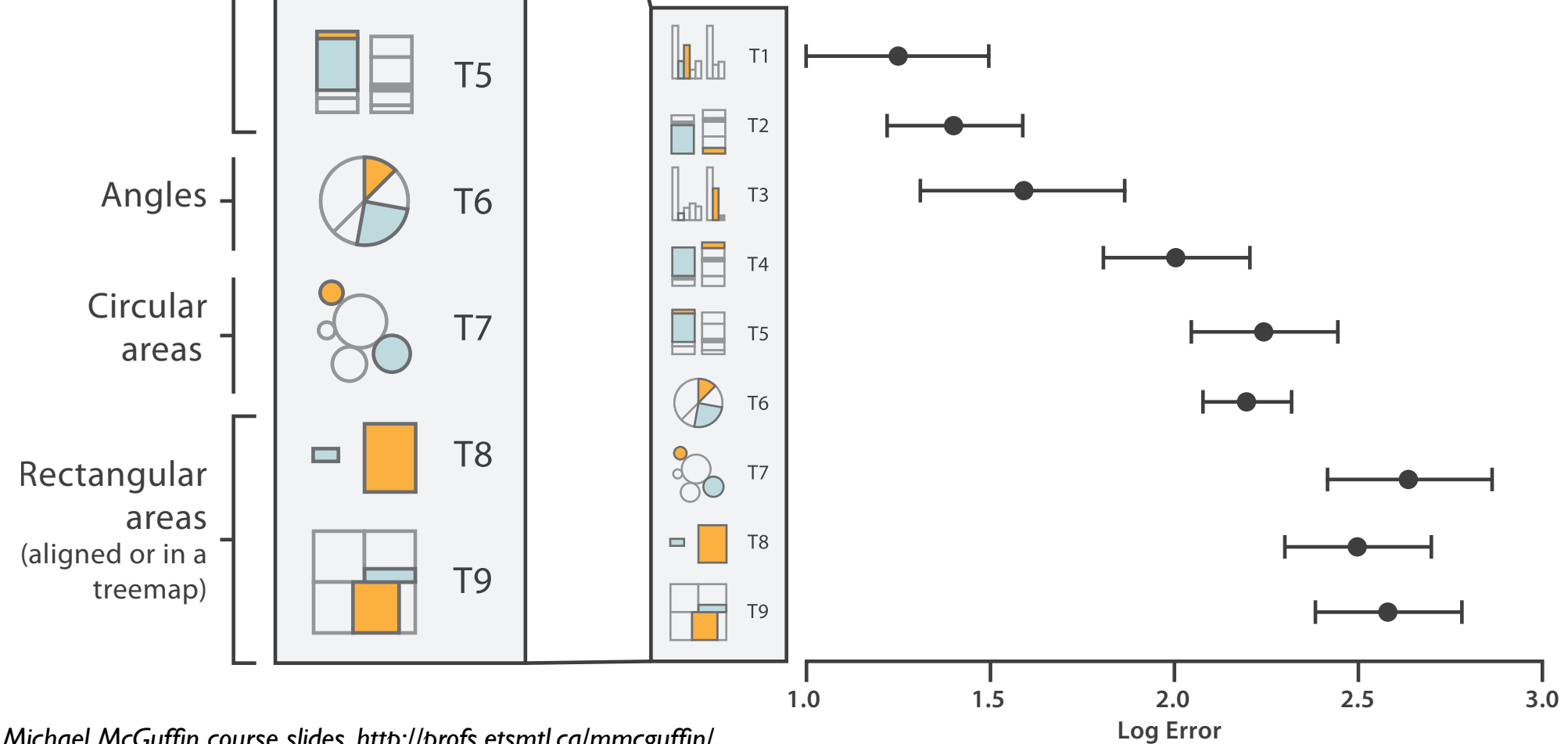


Accuracy: Vis experiments

Cleveland & McGill's Results



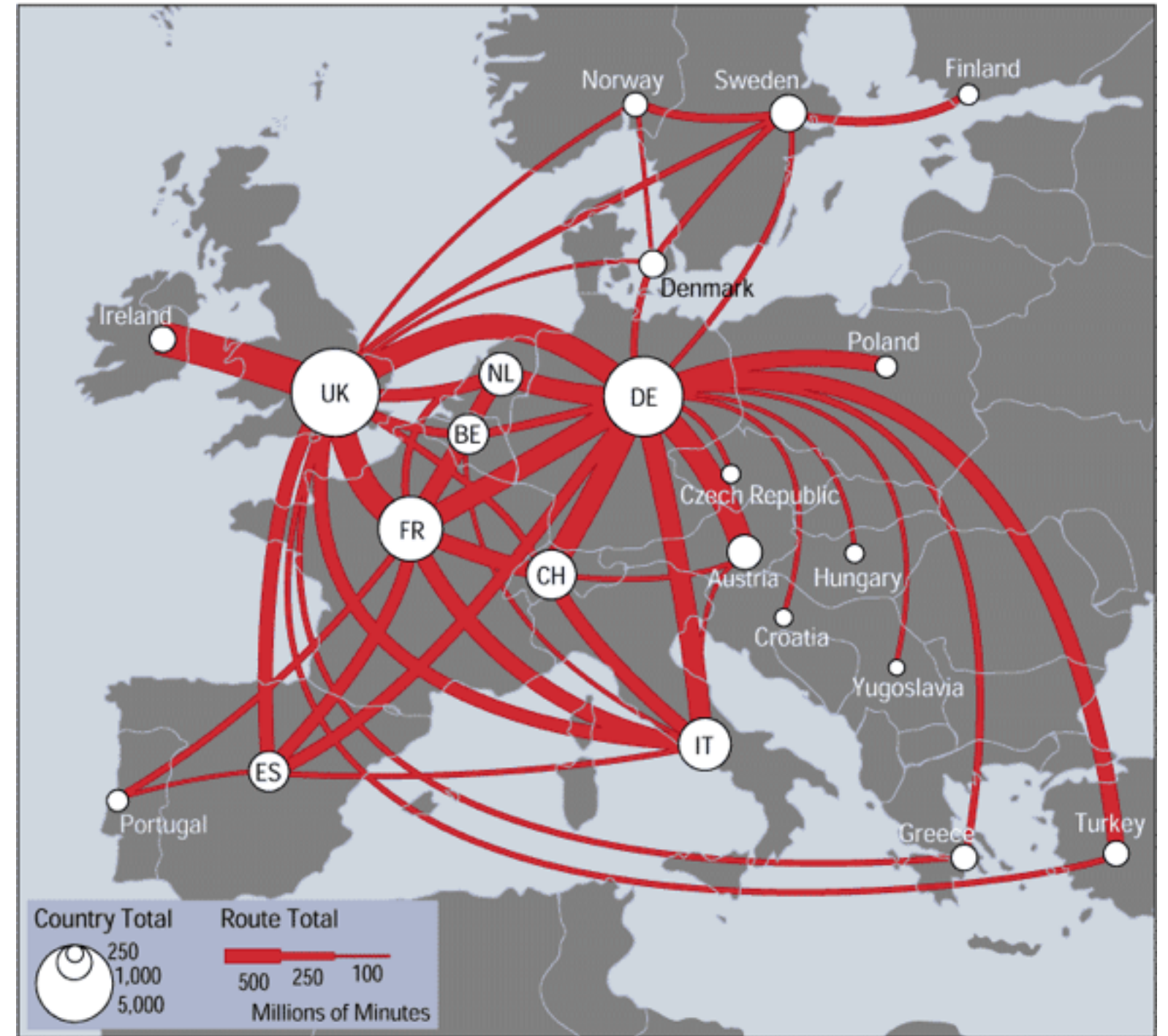
Crowdsourced Results



[Crowdsourcing Graphical Perception: Using Mechanical Turk to Assess Visualization Design. Heer and Bostock. Proc ACM Conf. Human Factors in Computing Systems (CHI) 2010, p. 203–212.]

Discriminability: How many usable steps?

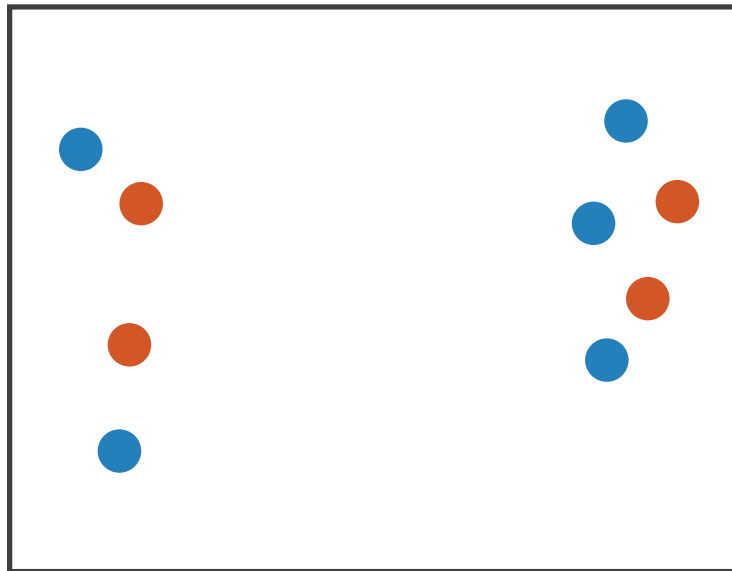
- must be sufficient for number of attribute levels to show
 - linewidth: few bins



[mappa.mundi.net/maps/maps_014/telegeography.html]

Separability vs. Integrality

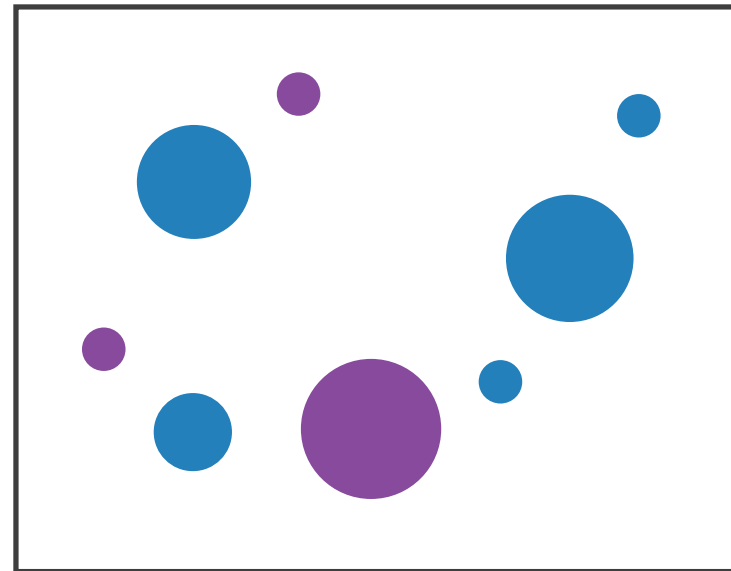
Position
+ Hue (Color)



Fully separable

2 groups each

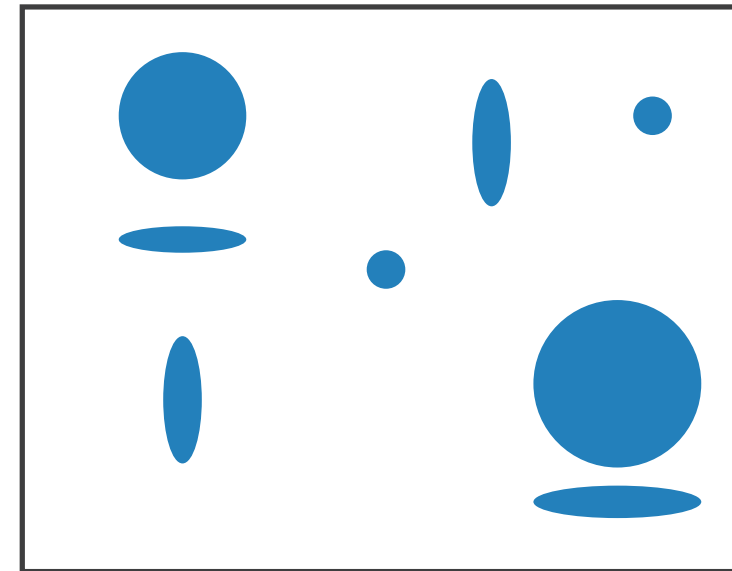
Size
+ Hue (Color)



Some interference

2 groups each

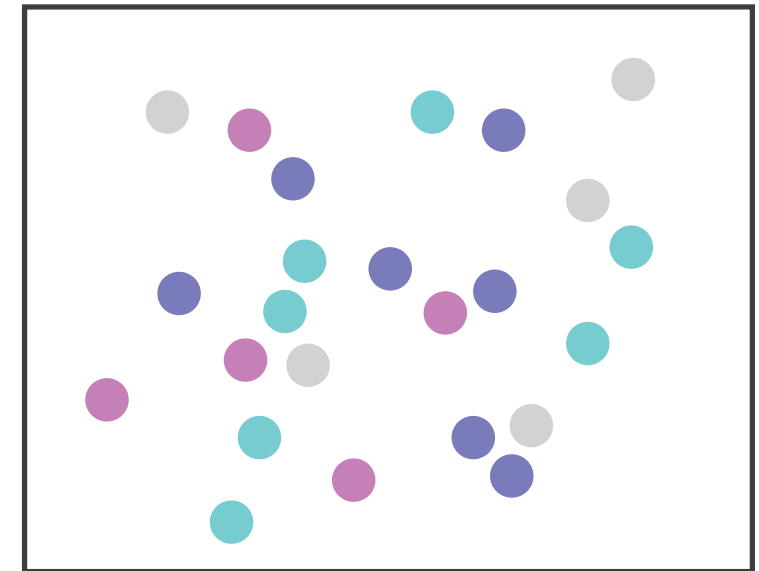
Width
+ Height



Some/significant
interference

3 groups total:
integral area

Red
+ Green

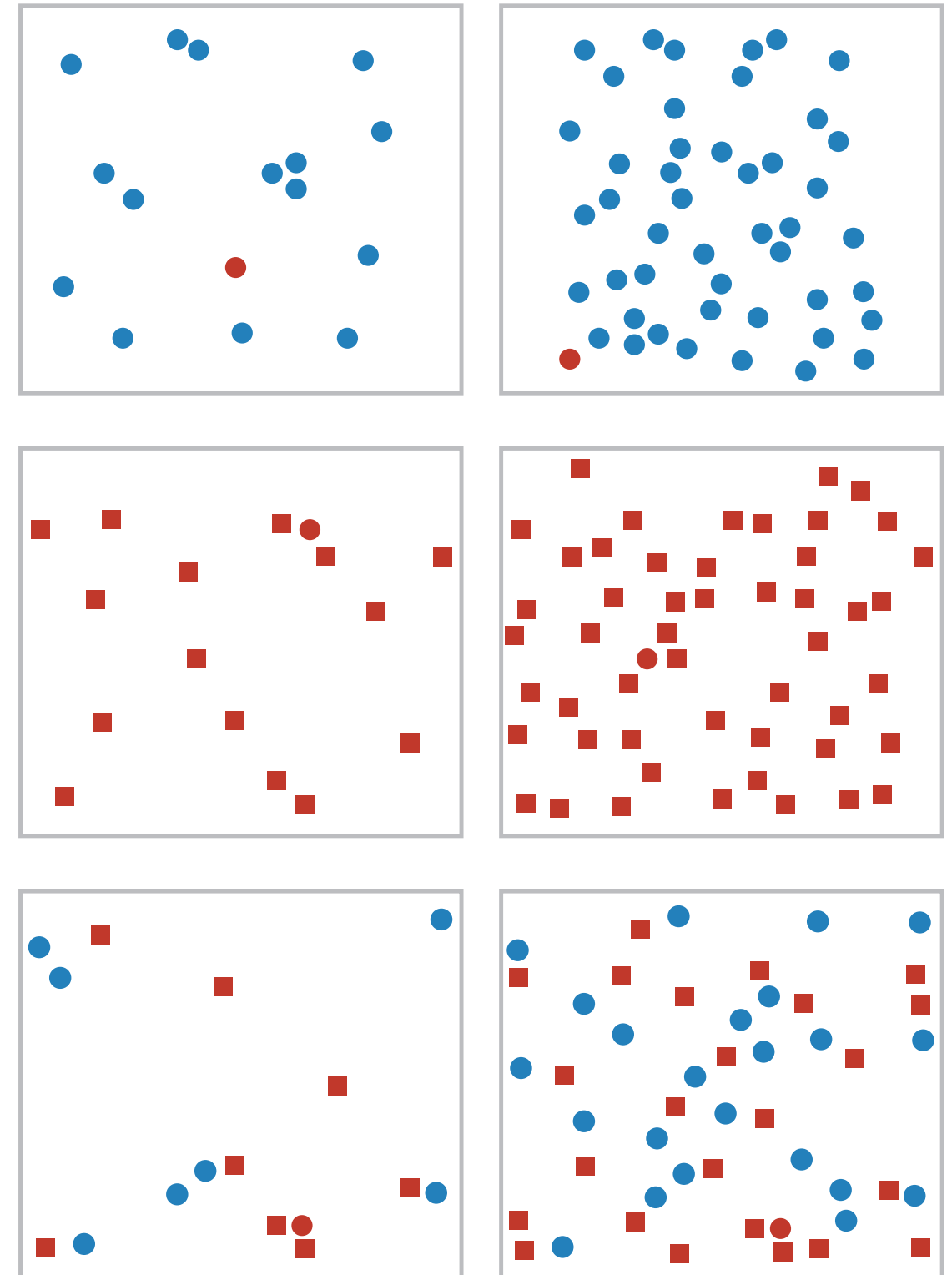


Major interference

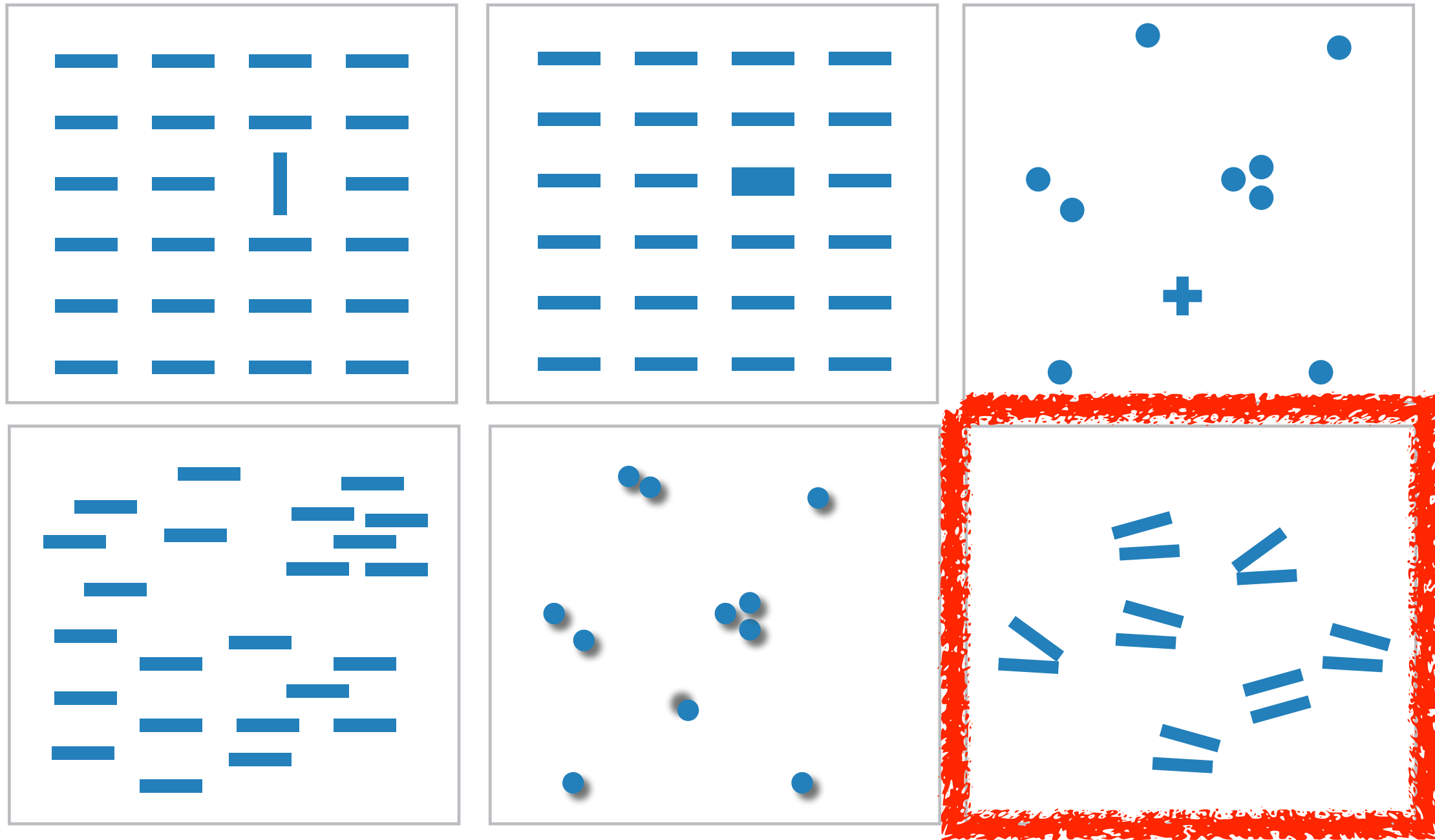
4 groups total:
integral hue

Popout

- find the red dot
 - how long does it take?
- parallel processing on many individual channels
 - speed independent of distractor count
 - speed depends on channel and amount of difference from distractors
- serial search for (almost all) combinations
 - speed depends on number of distractors



Popout



- many channels: tilt, size, shape, proximity, shadow direction, ...
- but not all! parallel line pairs do not pop out from tilted pairs

Grouping

- containment
- connection

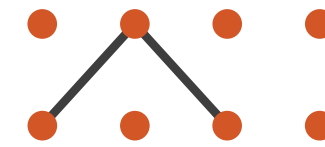
- proximity
 - same spatial region
- similarity
 - same values as other categorical channels

Marks as Links

➔ Containment



➔ Connection



➔ Identity Channels: Categorical Attributes

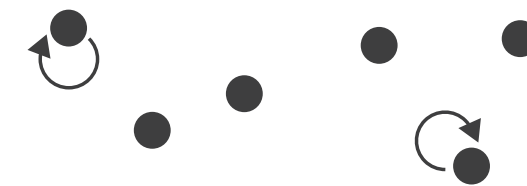
Spatial region



Color hue



Motion



Shape



Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
 - *Chap 5: Marks and Channels*
- *On the Theory of Scales of Measurement*. Stevens. Science 103:2684 (1946), 677–680.
- *Psychophysics: Introduction to its Perceptual, Neural, and Social Prospects*. Stevens. Wiley, 1975.
- *Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods*. Cleveland and McGill. Journ. American Statistical Association 79:387 (1984), 531–554.
- *Perception in Vision*. Healey. <http://www.csc.ncsu.edu/faculty/healey/PP>
- *Visual Thinking for Design*. Ware. Morgan Kaufmann, 2008.
- *Information Visualization: Perception for Design*, 3rd edition. Ware. Morgan Kaufmann /Academic Press, 2004.

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - **Arrange Tables**
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

How?

Encode

→ Arrange

→ Express



→ Separate



→ Order



→ Align



→ Use



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape



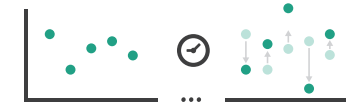
→ Motion

Direction, Rate, Frequency, ...

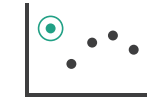


Manipulate

→ Change



→ Select



→ Navigate

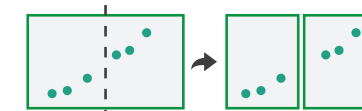


Facet

→ Juxtapose



→ Partition



→ Superimpose



Reduce

→ Filter



→ Aggregate



→ Embed



What?

Why?

How?

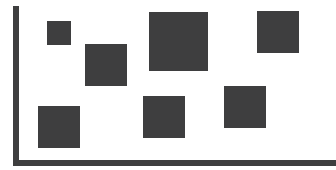
Arrange tables

→ Express Values

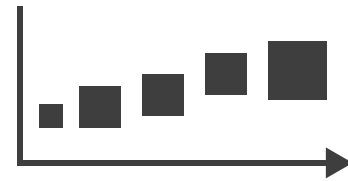


→ Separate, Order, Align Regions

→ Separate



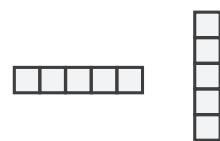
→ Order



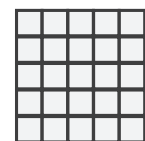
→ Align



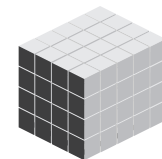
→ 1 Key *List*



→ 2 Keys *Matrix*



→ 3 Keys *Volume*

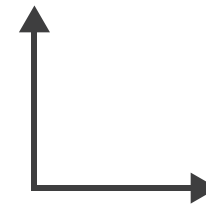


→ Many Keys *Recursive Subdivision*

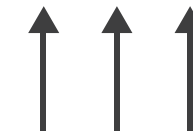


→ Axis Orientation

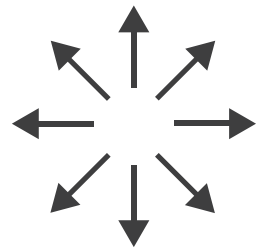
→ Rectilinear



→ Parallel

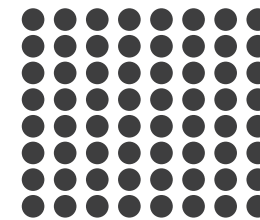


→ Radial



→ Layout Density

→ Dense



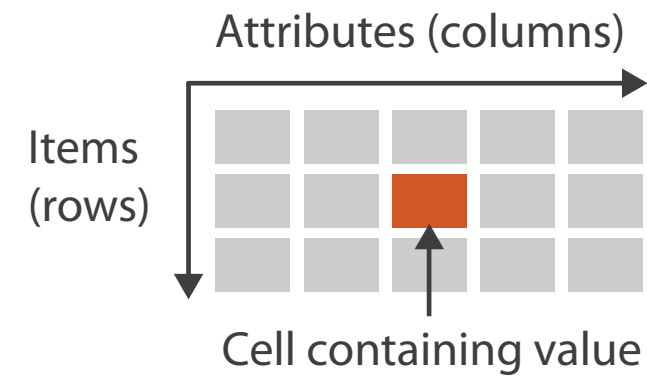
→ Space-Filling



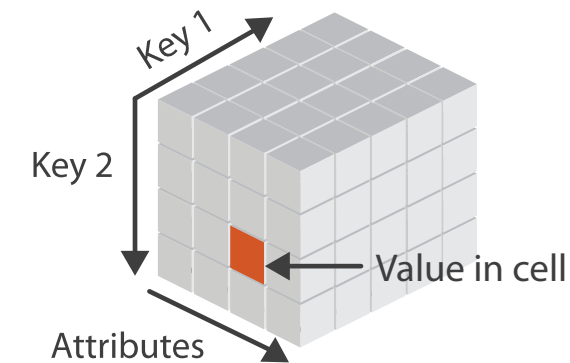
Keys and values

- key
 - independent attribute
 - used as unique index to look up items
 - simple tables: 1 key
 - multidimensional tables: multiple keys
- value
 - dependent attribute, value of cell
- classify arrangements by key count
 - 0, 1, 2, many...

→ Tables

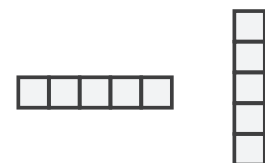


→ *Multidimensional Table*

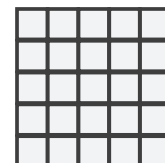


⊙ Express Values

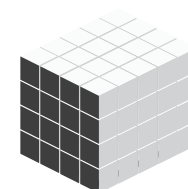
→ 1 Key
List



→ 2 Keys
Matrix



→ 3 Keys
Volume



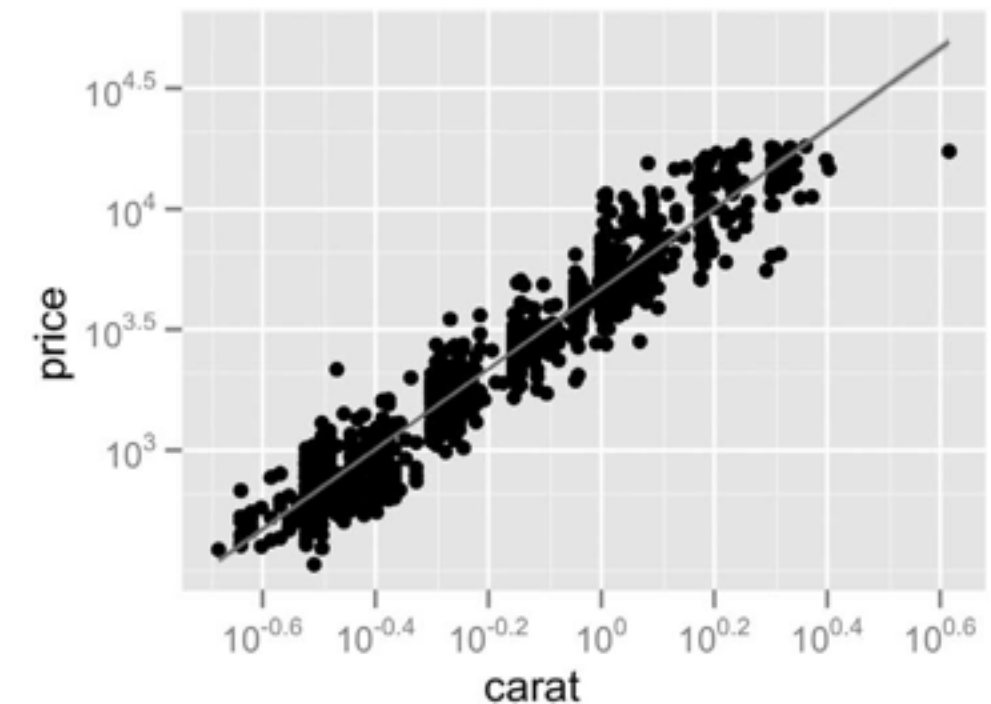
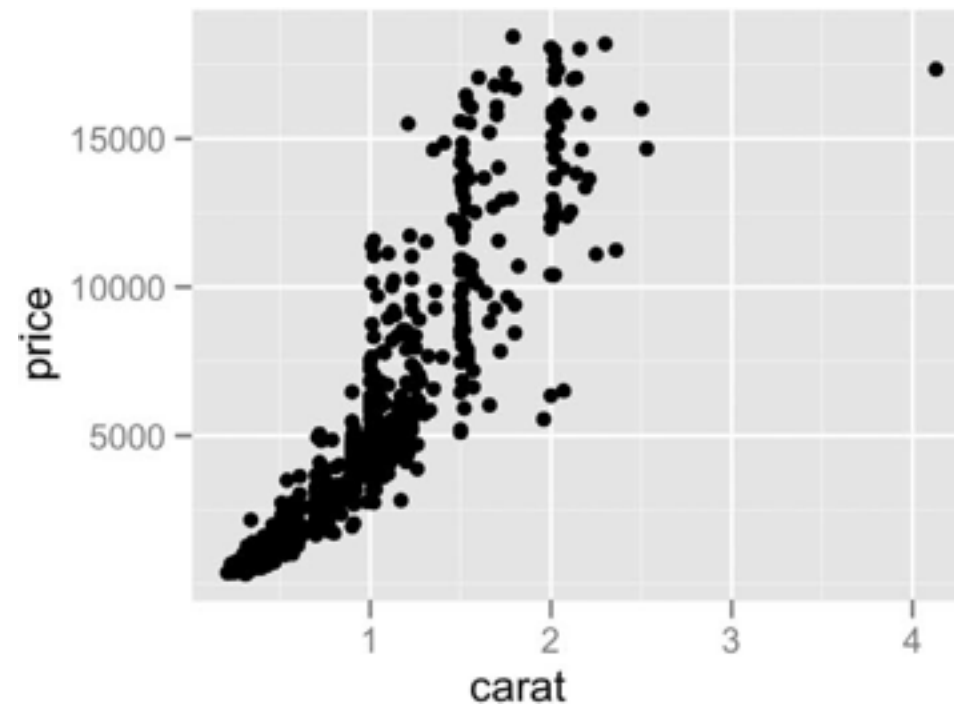
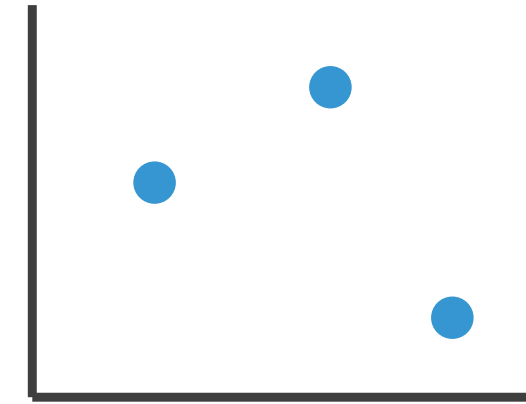
→ Many Keys
Recursive Subdivision



Idiom: scatterplot

- *express* values
 - quantitative attributes
- no keys, only values
 - data
 - 2 quant attribs
 - mark: points
 - channels
 - horiz + vert position
 - tasks
 - find trends, outliers, distribution, correlation, clusters
 - scalability
 - hundreds of items

➔ Express Values

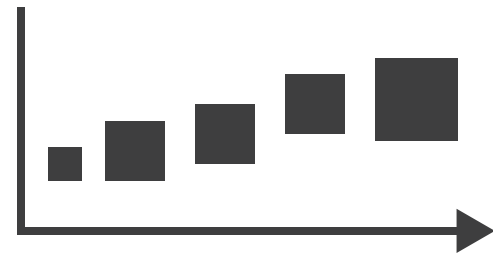


Some keys: Categorical regions

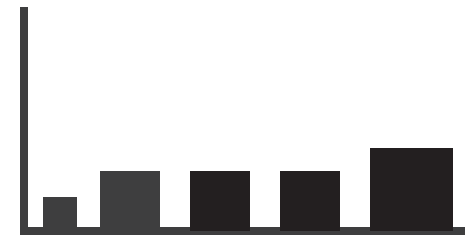
→ Separate



→ Order

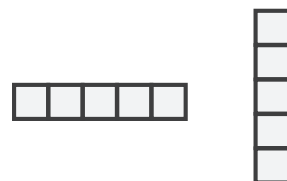


→ Align

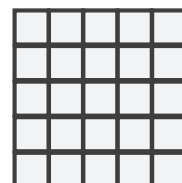


- **regions:** contiguous bounded areas distinct from each other
 - using space to *separate* (proximity)
 - following expressiveness principle for categorical attributes
- use ordered attribute to *order* and *align* regions

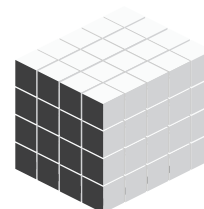
→ 1 Key
List



→ 2 Keys
Matrix



→ 3 Keys
Volume



→ Many Keys
Recursive Subdivision



Idiom: bar chart

- one key, one value

- data

- 1 categ attrib, 1 quant attrib

- mark: lines

- channels

- length to express quant value

- spatial regions: one per mark

- separated horizontally, aligned vertically

- ordered by quant attrib

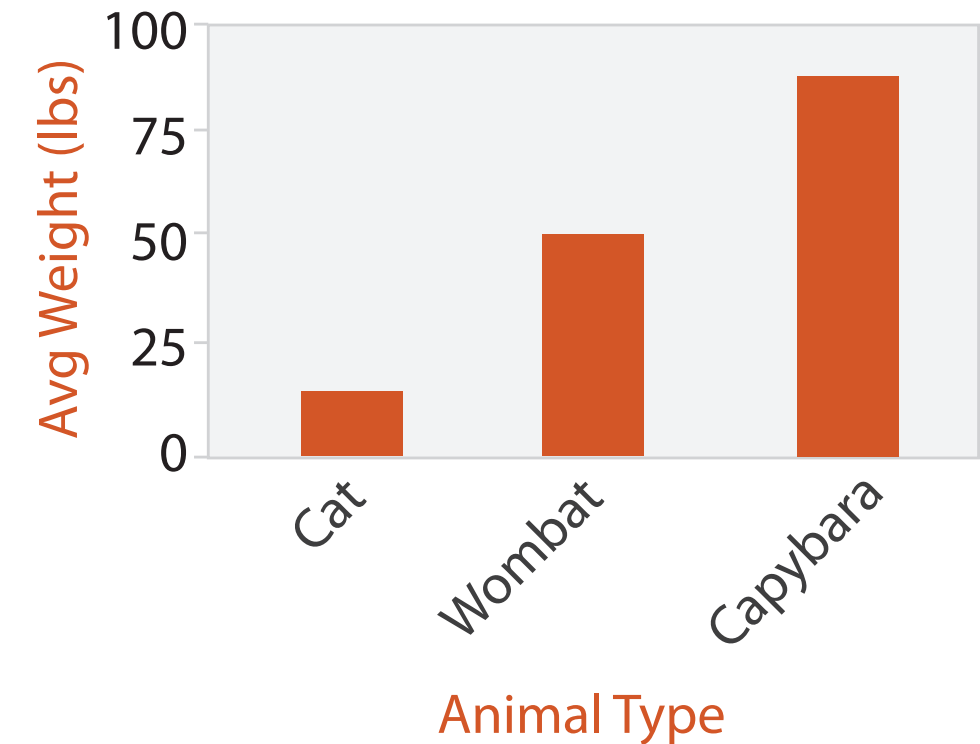
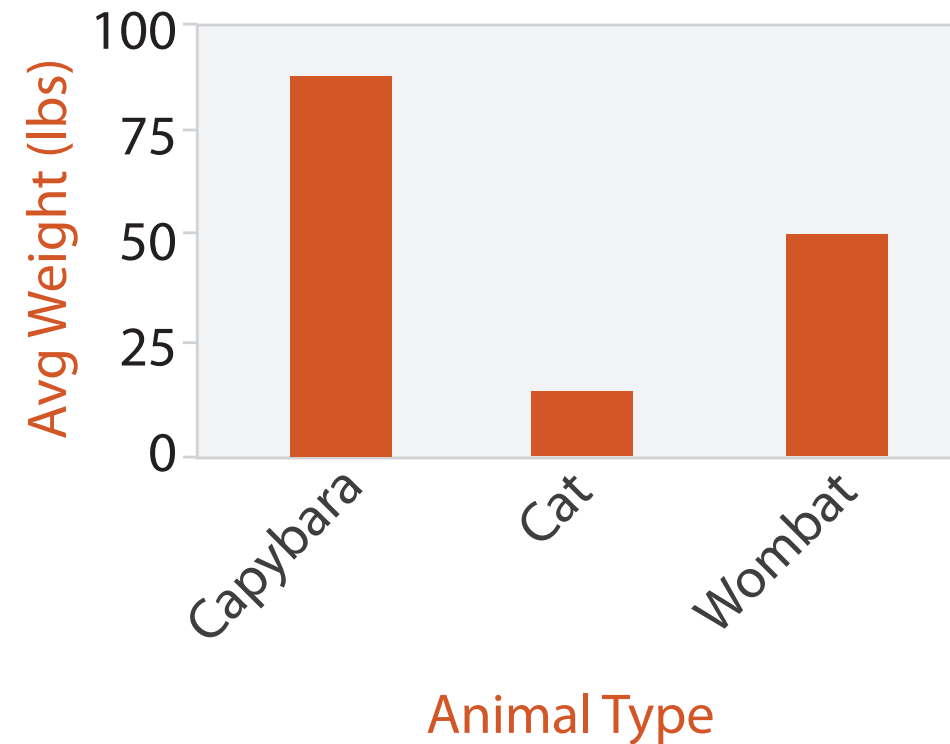
- » by label (alphabetical), by length attrib (data-driven)

- task

- compare, lookup values

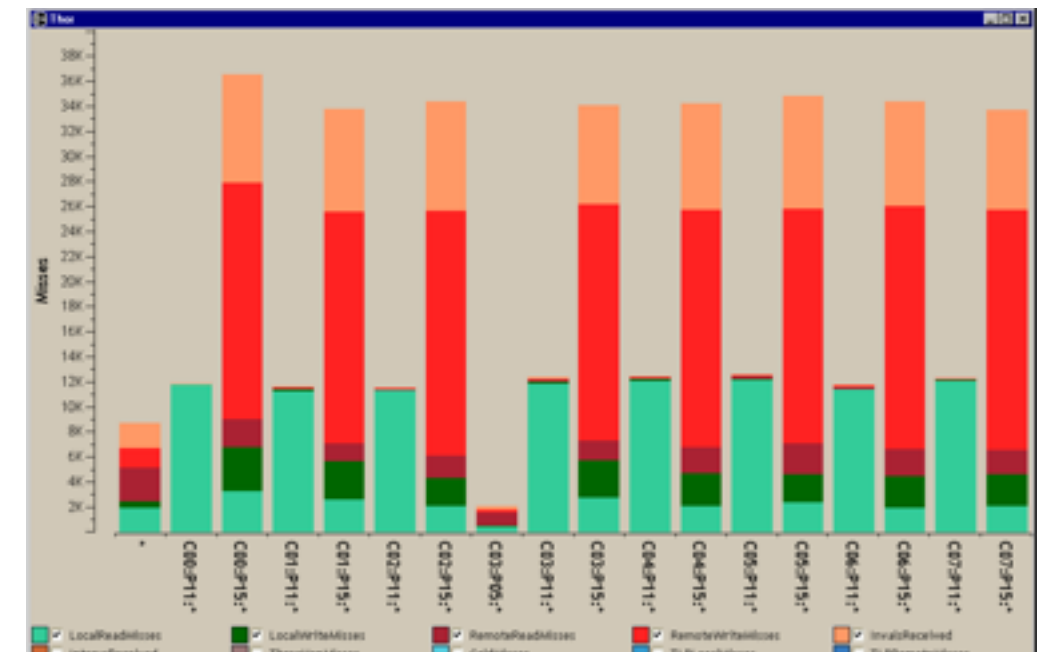
- scalability

- dozens to hundreds of levels for key attrib



Idiom: stacked bar chart

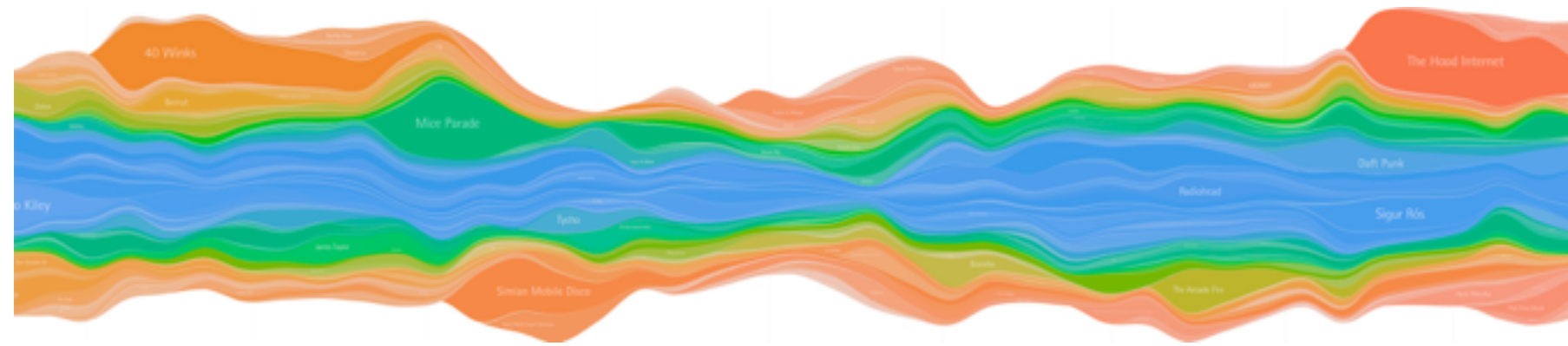
- one more key
 - data
 - 2 categ attrib, 1 quant attrib
 - mark: vertical stack of line marks
 - glyph: composite object, internal structure from multiple marks
 - channels
 - length and color hue
 - spatial regions: one per glyph
 - aligned: full glyph, lowest bar component
 - unaligned: other bar components
 - task
 - part-to-whole relationship
 - scalability
 - several to one dozen levels for stacked attrib



[Using Visualization to Understand the Behavior of Computer Systems. Bosch. Ph.D. thesis, Stanford Computer Science, 2001.]

Idiom: streamgraph

- generalized stacked graph
 - emphasizing horizontal continuity
 - vs vertical items
 - data
 - 1 categ key attrib (artist)
 - 1 ordered key attrib (time)
 - 1 quant value attrib (counts)
 - derived data
 - geometry: layers, where height encodes counts
 - 1 quant attrib (layer ordering)
 - scalability
 - hundreds of time keys
 - dozens to hundreds of artist keys
 - more than stacked bars, since most layers don't extend across whole chart



[Stacked Graphs Geometry & Aesthetics. Byron and Wattenberg. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2008) 14(6): 1245–1252, (2008).]

Idiom: line chart / dot plot

- one key, one value

- data

- 2 quant attribs

- mark: points

- line connection marks between them

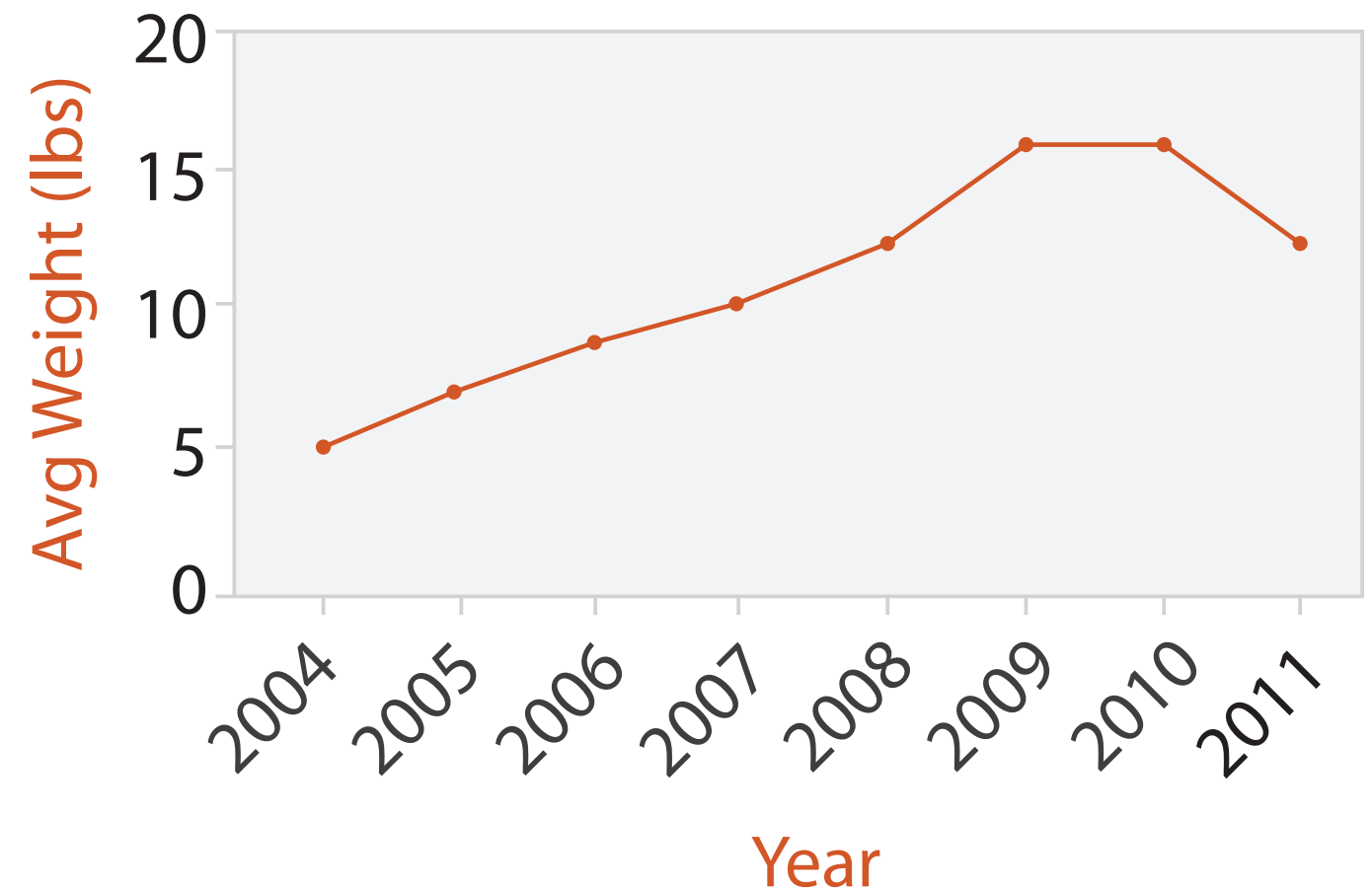
- channels

- aligned lengths to express quant value
- separated and ordered by key attrib into horizontal regions

- task

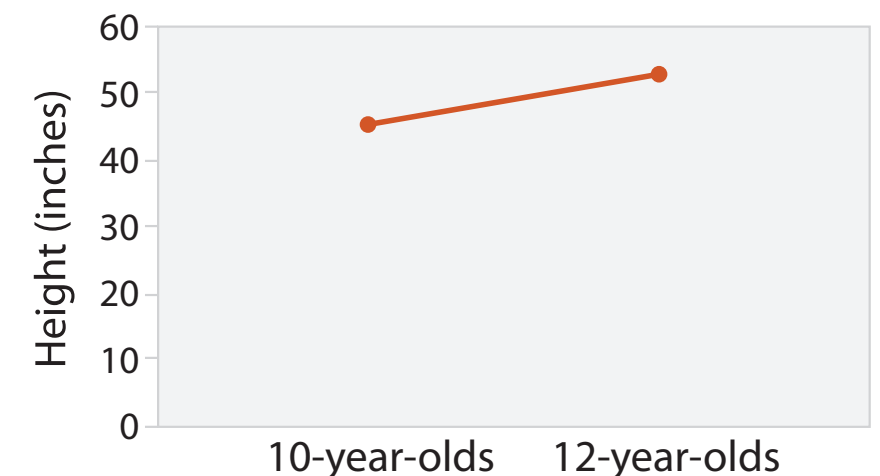
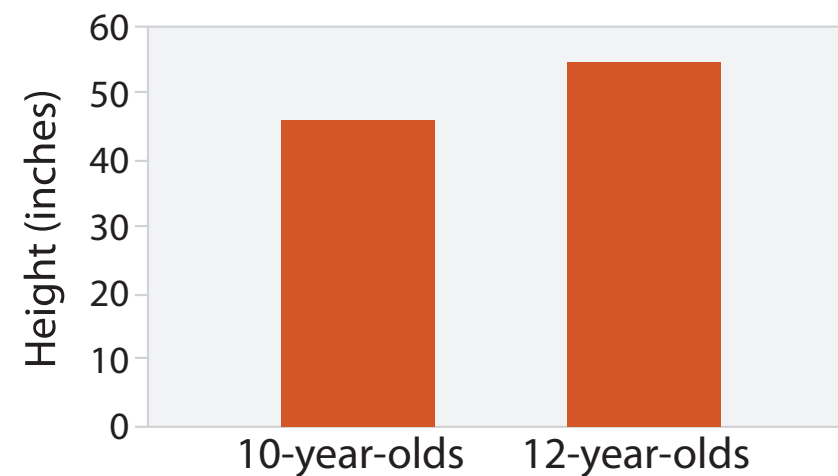
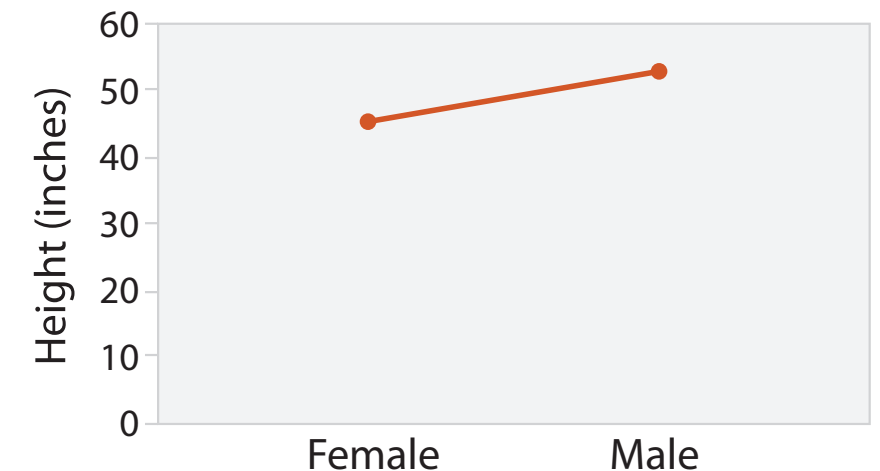
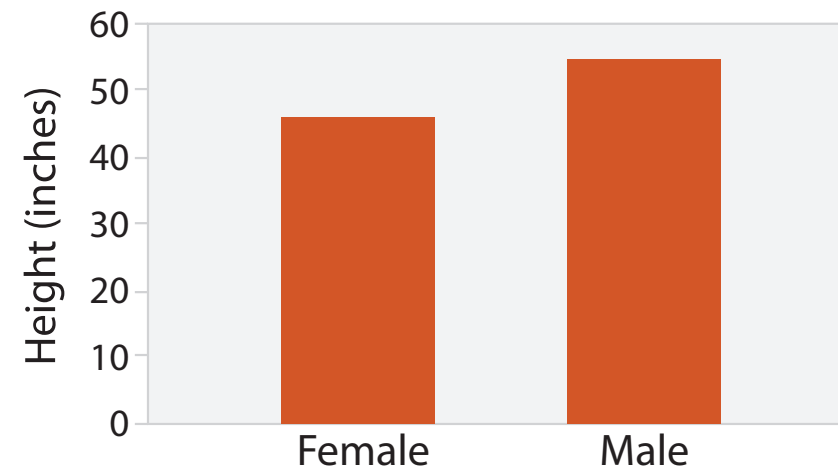
- find trend

- connection marks emphasize ordering of items along key axis by explicitly showing relationship between one item and the next



Choosing bar vs line charts

- depends on type of key attrib
 - bar charts if categorical
 - line charts if ordered
- do not use line charts for categorical key attribs
 - violates expressiveness principle
 - implication of trend so strong that it overrides semantics!
 - “The more male a person is, the taller he/she is”



after [Bars and Lines: A Study of Graphic Communication. Zacks and Tversky. *Memory and Cognition* 27:6 (1999), 1073–1079.]

Idiom: heatmap

- two keys, one value

- data

- 2 categ attribs (gene, experimental condition)
- 1 quant attrib (expression levels)

- marks: area

- separate and align in 2D matrix
 - indexed by 2 categorical attributes

- channels

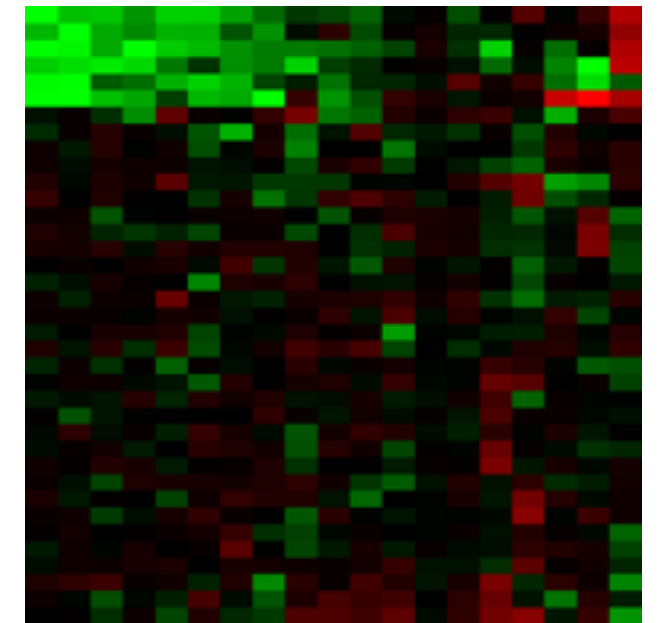
- color by quant attrib
 - (ordered diverging colormap)

- task

- find clusters, outliers

- scalability

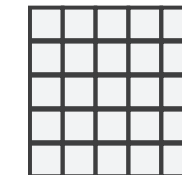
- 1K categorical levels, 1M items; only ~10 quantitative attribute levels



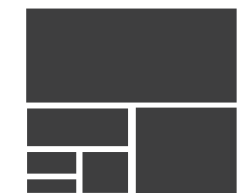
→ 1 Key
List



→ 2 Keys
Matrix

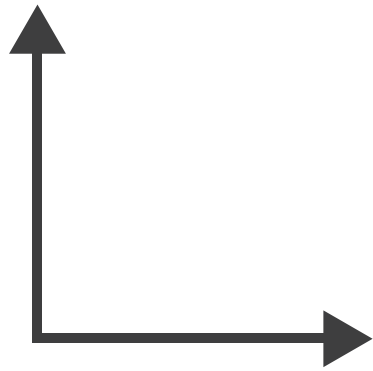


→ Many Keys
Recursive Subdivision

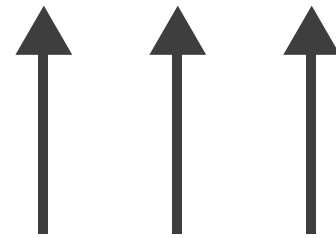


➔ Axis Orientation

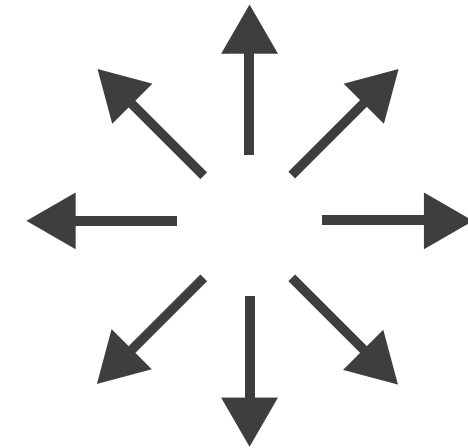
➔ Rectilinear



➔ Parallel



➔ Radial



Idioms: scatterplot matrix, parallel coordinates

- scatterplot matrix (SPLOM)

- rectilinear axes, point mark
- all possible pairs of axes
- scalability
 - one dozen attribs
 - dozens to hundreds of items

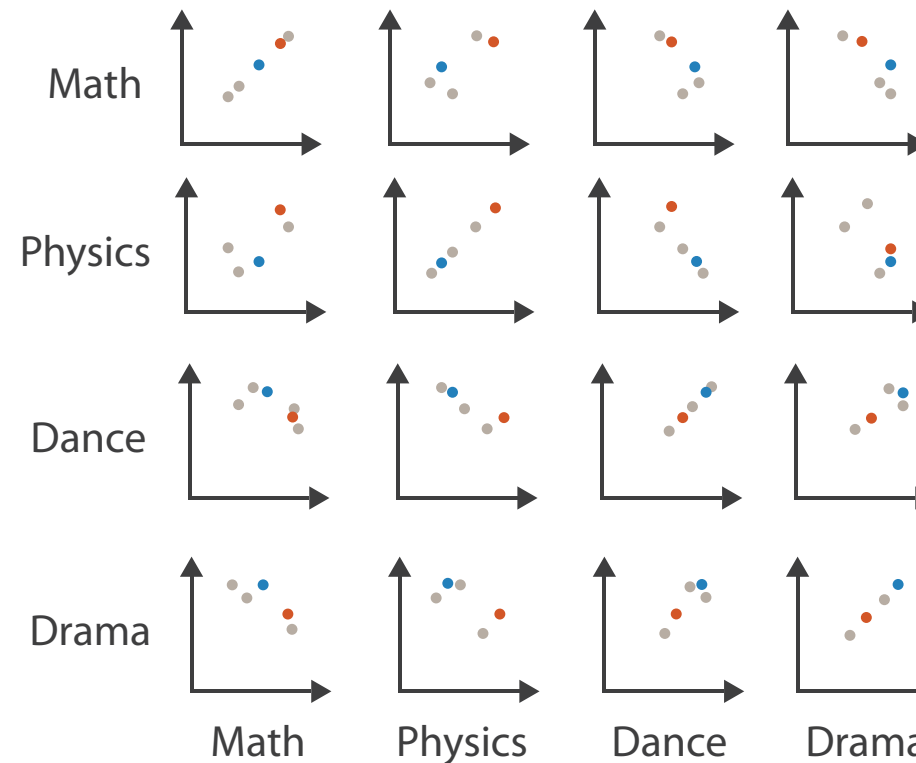
- parallel coordinates

- parallel axes, jagged line representing item
- rectilinear axes, item as point
 - axis ordering is major challenge

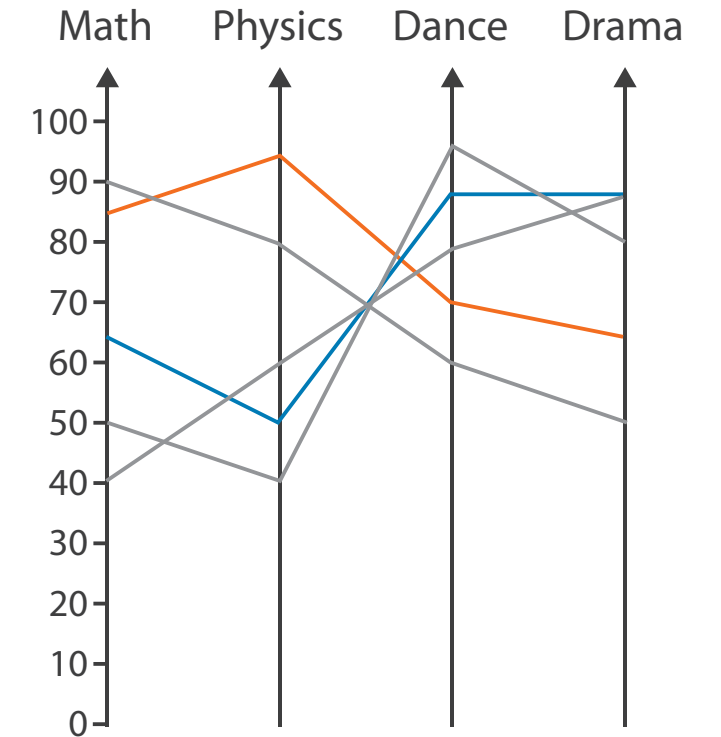
- scalability

- dozens of attribs
- hundreds of items

Scatterplot Matrix



Parallel Coordinates

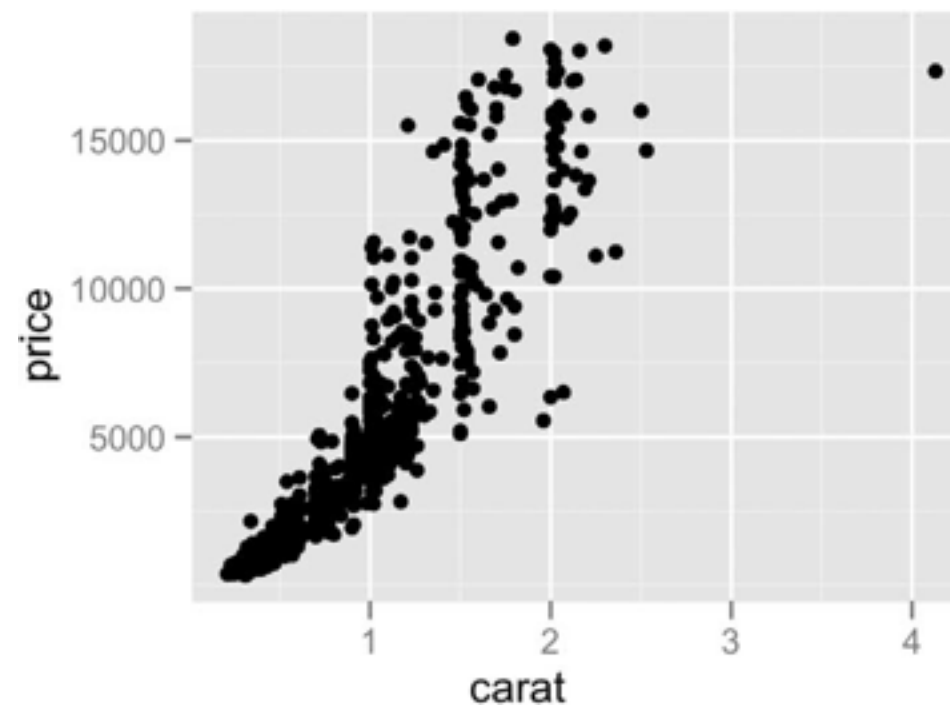


Table

Math	Physics	Dance	Drama
85	95	70	65
90	80	60	50
65	50	90	90
50	40	95	80
40	60	80	90

Task: Correlation

- scatterplot matrix
 - positive correlation
 - diagonal low-to-high
 - negative correlation
 - diagonal high-to-low
 - uncorrelated
- parallel coordinates
 - positive correlation
 - parallel line segments
 - negative correlation
 - all segments cross at halfway point
 - uncorrelated
 - scattered crossings



[A layered grammar of graphics. Wickham. *Journ. Computational and Graphical Statistics* 19:1 (2010), 3–28.]

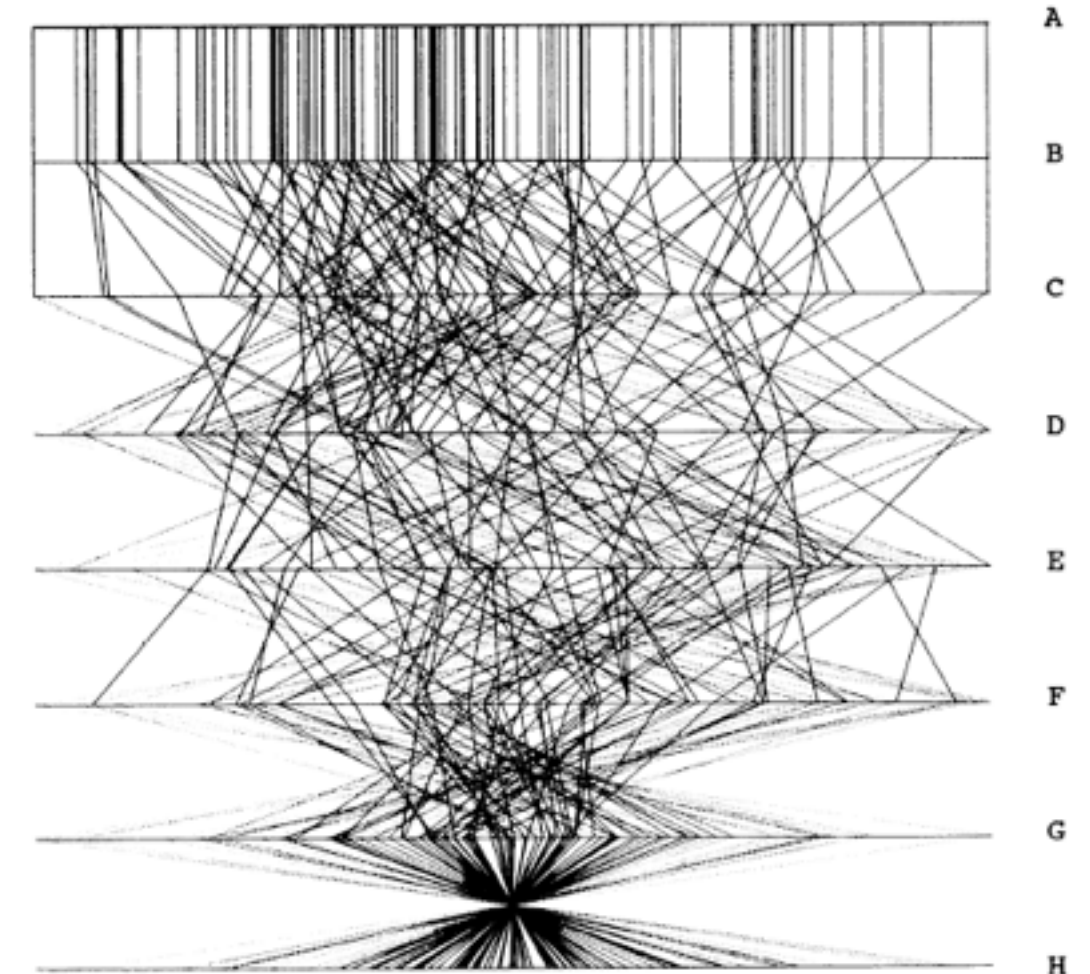
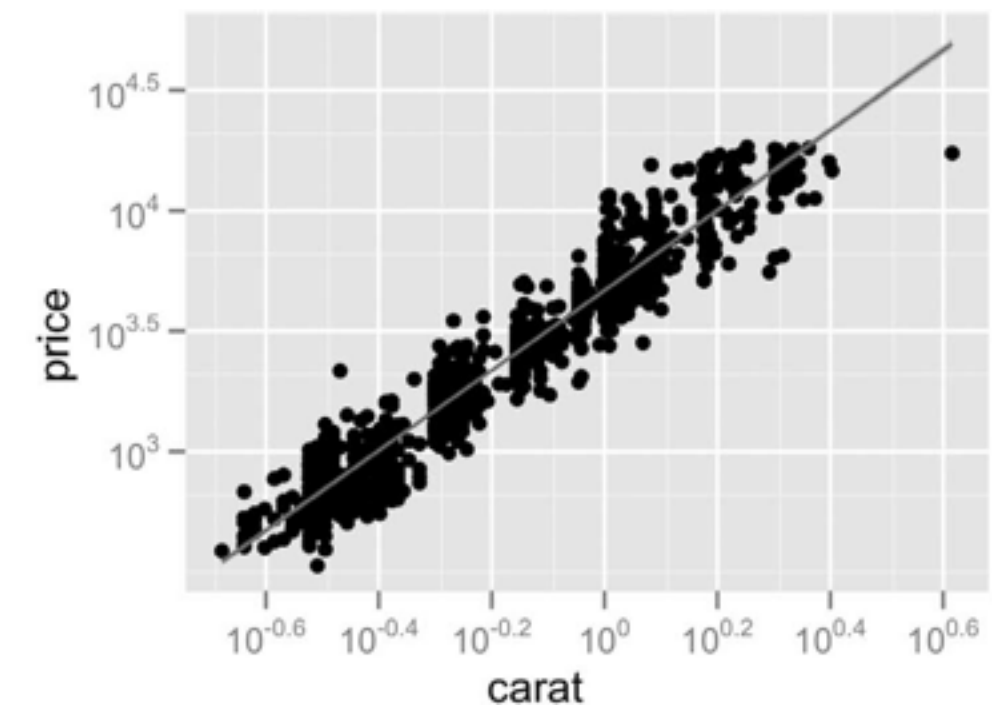


Figure 3. Parallel Coordinate Plot of Six-Dimensional Data Illustrating Correlations of $\rho = 1, .8, .2, 0, -.2, -.8,$ and -1 .

[Hyperdimensional Data Analysis Using Parallel Coordinates. Wegman. *Journ. American Statistical Association* 85:411 (1990), 664–675.]

Idioms: pie chart, polar area chart

- pie chart

- area marks with angle channel

- accuracy: angle/area much less accurate than line length

- arclength also less accurate than line length

- polar area chart

- area marks with length channel

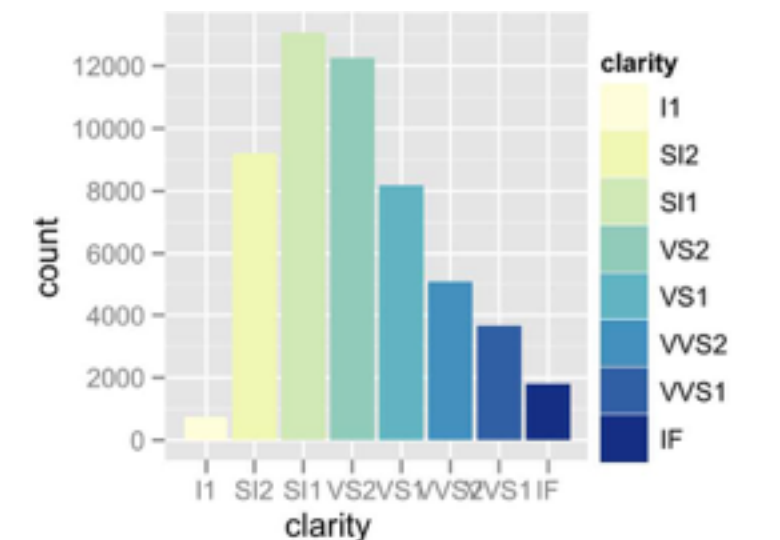
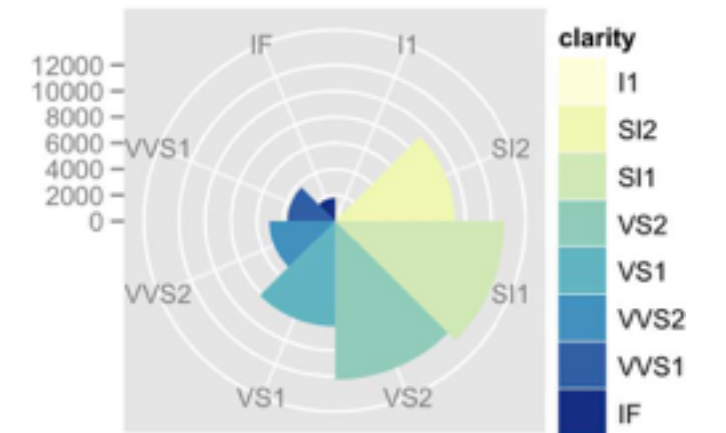
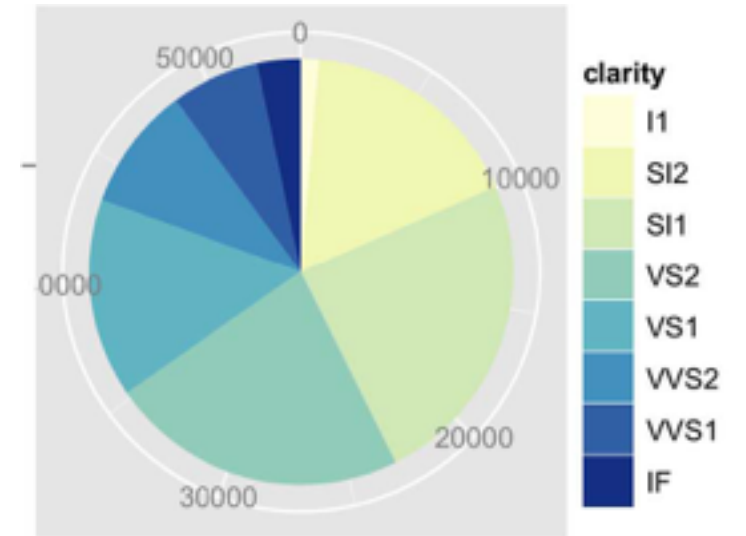
- more direct analog to bar charts

- data

- I categ key attrib, I quant value attrib

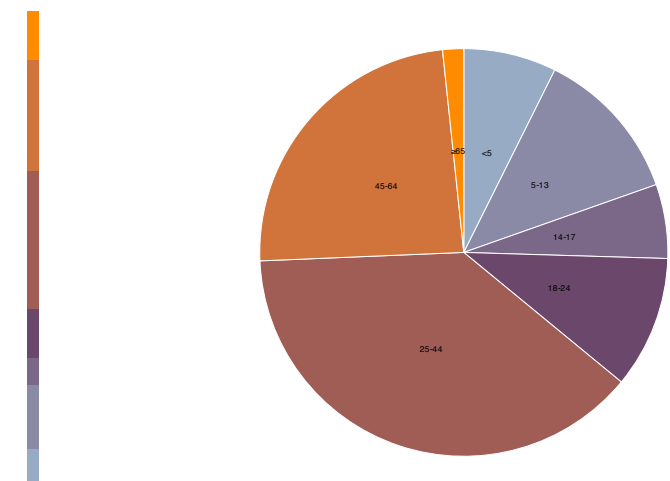
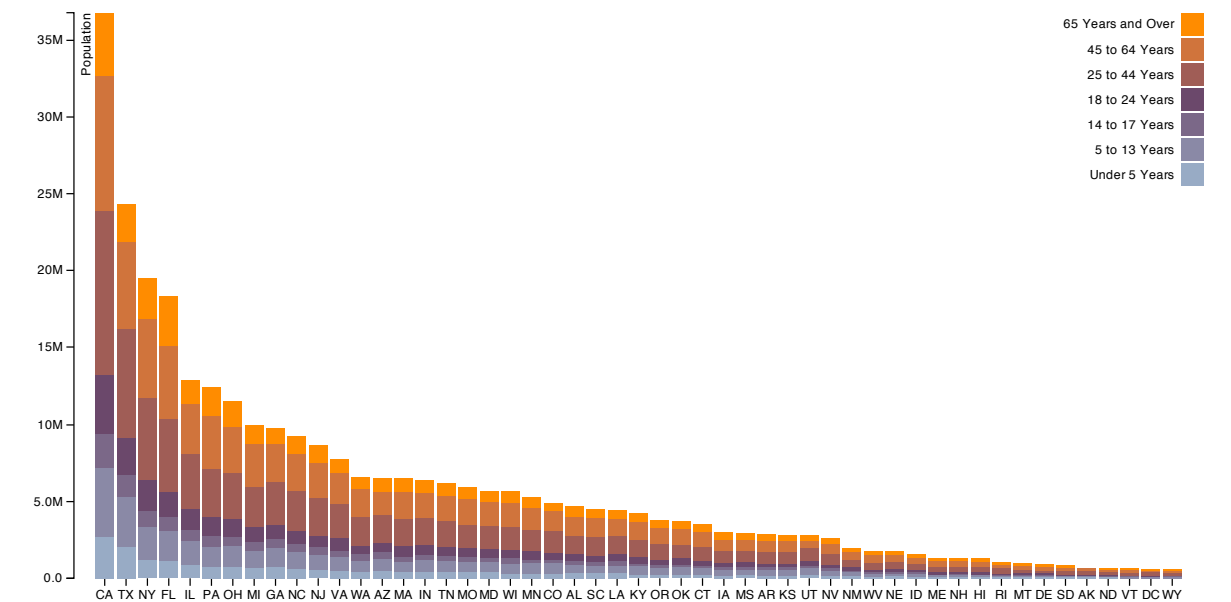
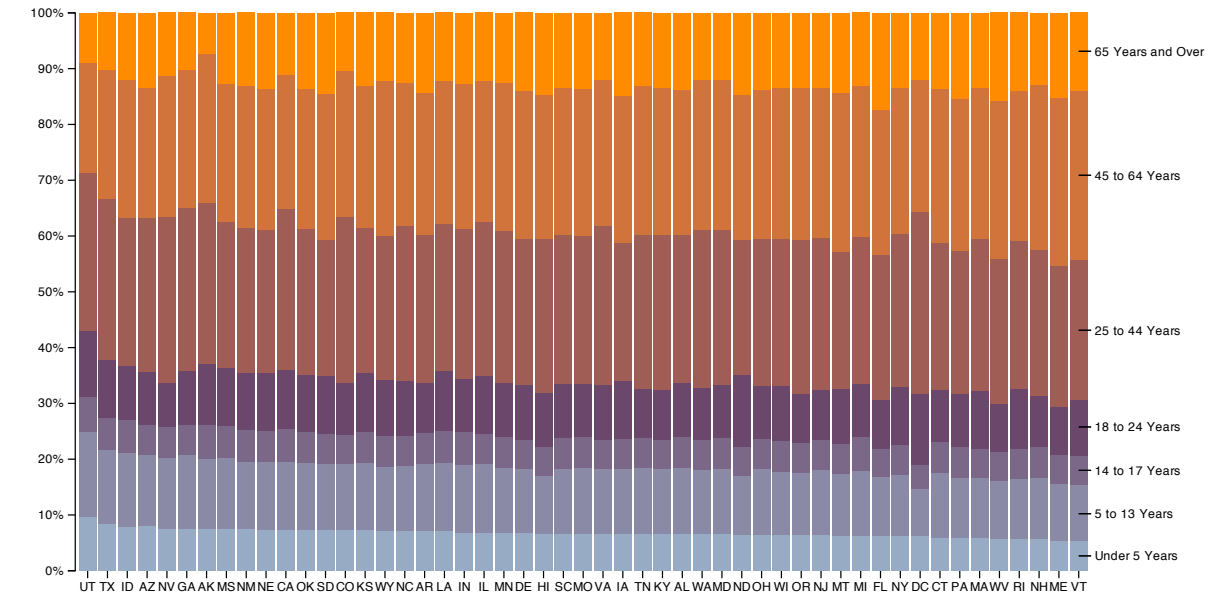
- task

- part-to-whole judgements



Idioms: **normalized stacked bar chart**

- task
 - part-to-whole judgements
- **normalized stacked bar chart**
 - stacked bar chart, normalized to full vert height
 - single stacked bar equivalent to full pie
 - high information density: requires narrow rectangle
- **pie chart**
 - information density: requires large circle



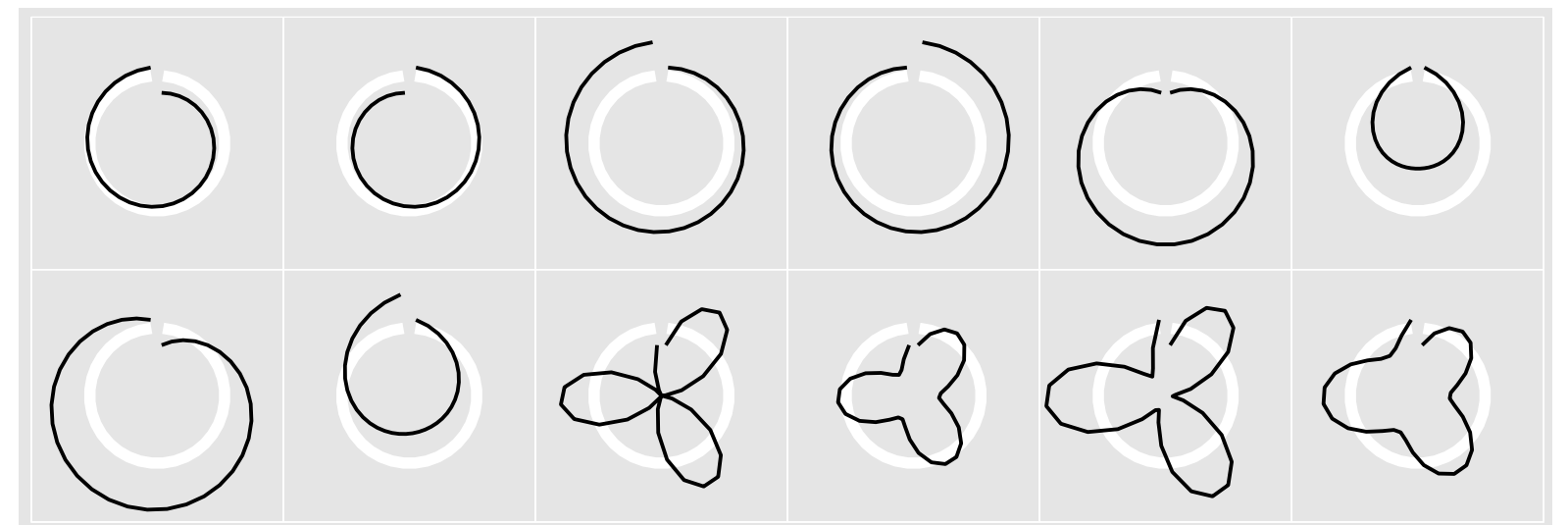
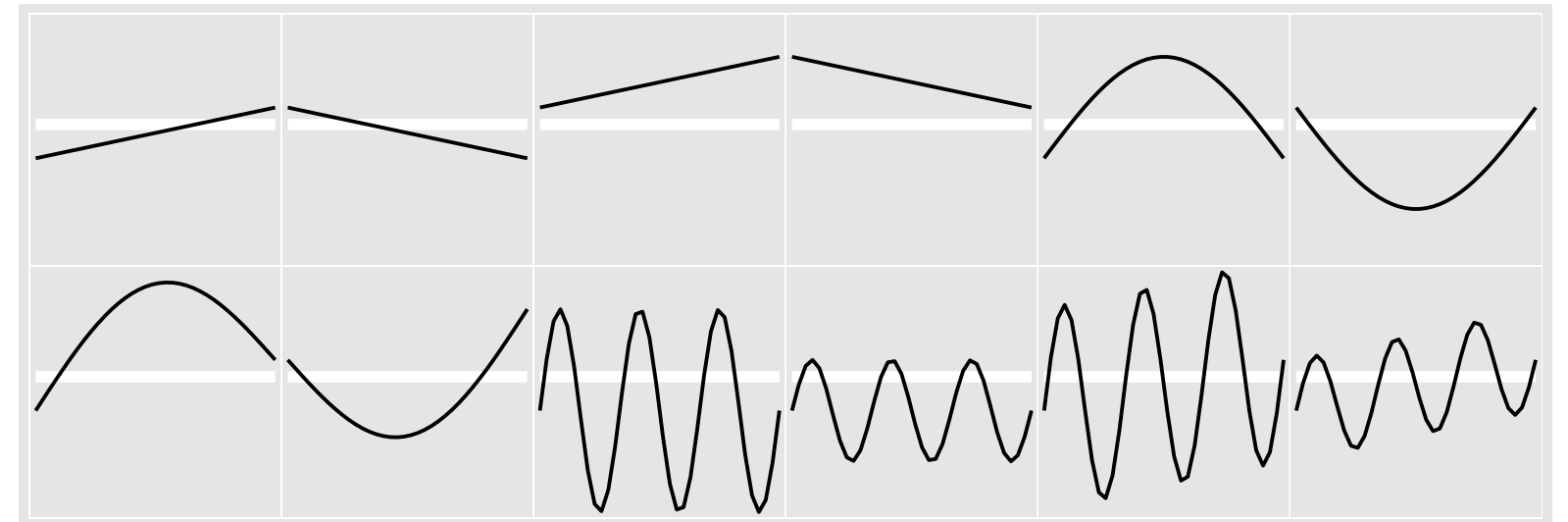
<http://bl.ocks.org/mbostock/3887235>,

<http://bl.ocks.org/mbostock/3886208>,

<http://bl.ocks.org/mbostock/3886394>.

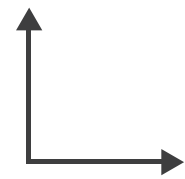
Idiom: **glyphmaps**

- rectilinear good for linear vs nonlinear trends
- radial good for cyclic patterns



➔ Axis Orientation

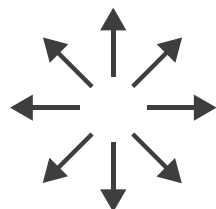
➔ Rectilinear



➔ Parallel



➔ Radial



[Glyph-maps for Visually Exploring Temporal Patterns in Climate Data and Models. Wickham, Hofmann, Wickham, and Cook. *Environmetrics* 23:5 (2012), 382–393.]

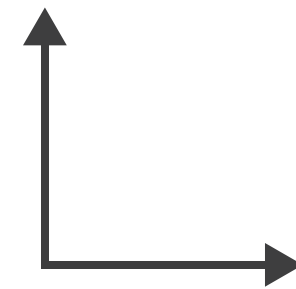
Orientation limitations

- **rectilinear: scalability wrt #axes**
 - 2 axes best
 - 3 problematic
 - more in afternoon
 - 4+ impossible
- **parallel: unfamiliarity, training time**
- **radial: perceptual limits**
 - angles lower precision than lengths
 - asymmetry between angle and length
 - can be exploited!

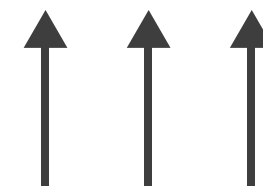
[Uncovering Strengths and Weaknesses of Radial Visualizations - an Empirical Approach. Diehl, Beck and Burch. IEEE TVCG (Proc. InfoVis) 16(6):935–942, 2010.]

➔ **Axis Orientation**

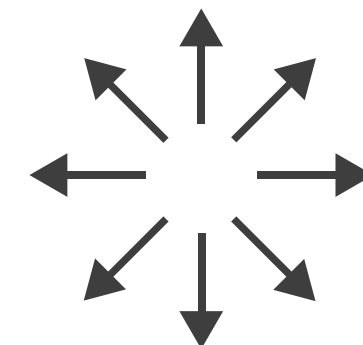
➔ Rectilinear



➔ Parallel



➔ Radial



Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
–*Chap 7: Arrange Tables*
- Visualizing Data. Cleveland. Hobart Press, 1993.
- *A Brief History of Data Visualization*. Friendly. 2008.
<http://www.datavis.ca/milestones>

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - **Arrange Spatial Data**
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

Arrange spatial data

→ Use Given

→ Geometry

→ *Geographic*

→ *Other Derived*

→ Spatial Fields

→ *Scalar Fields (one value per cell)*

→ *Isocontours*

→ *Direct Volume Rendering*

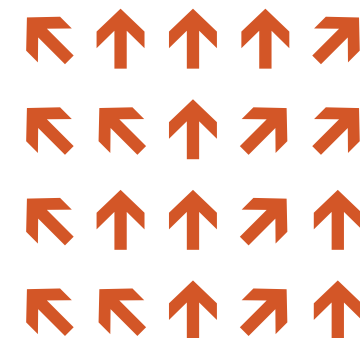
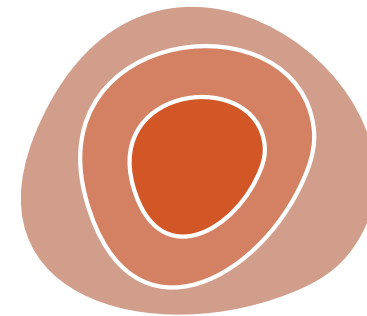
→ *Vector and Tensor Fields (many values per cell)*

→ *Flow Glyphs (local)*

→ *Geometric (sparse seeds)*

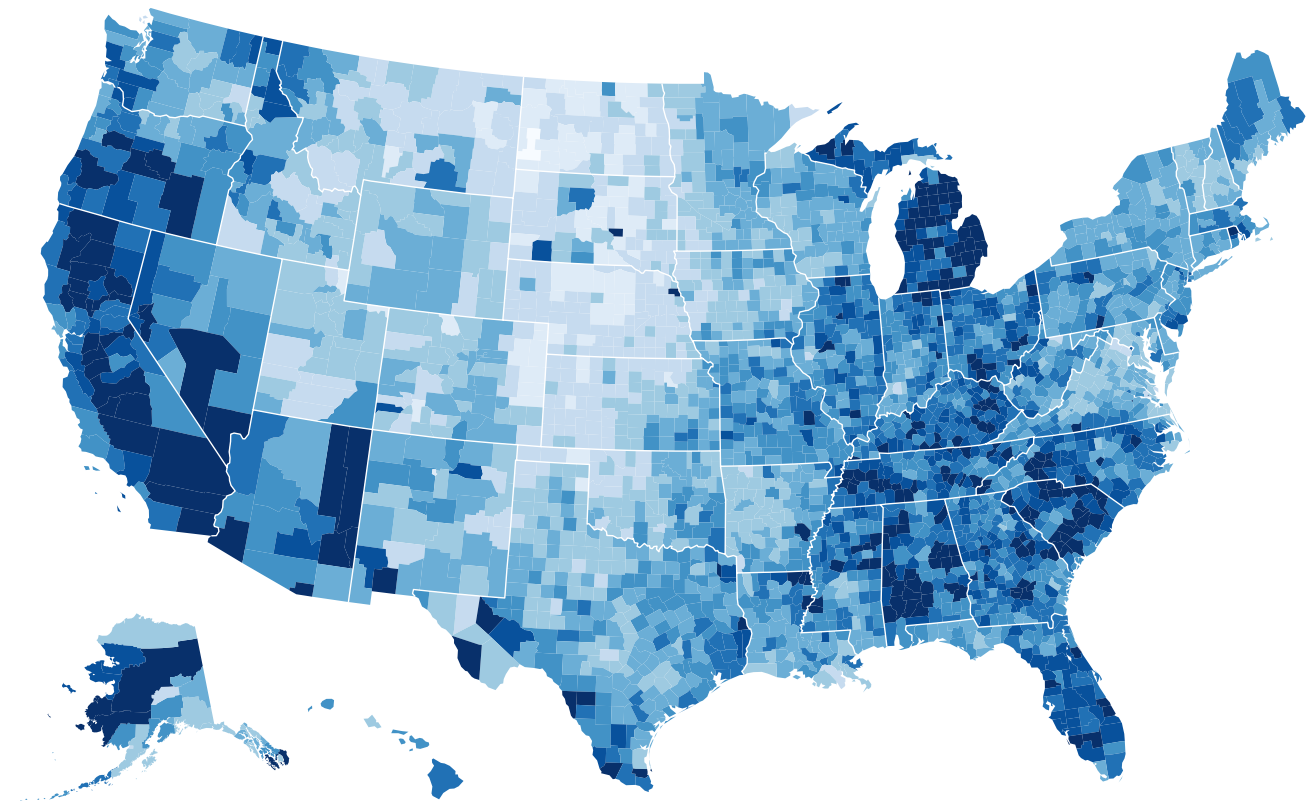
→ *Textures (dense seeds)*

→ *Features (globally derived)*



Idiom: **choropleth map**

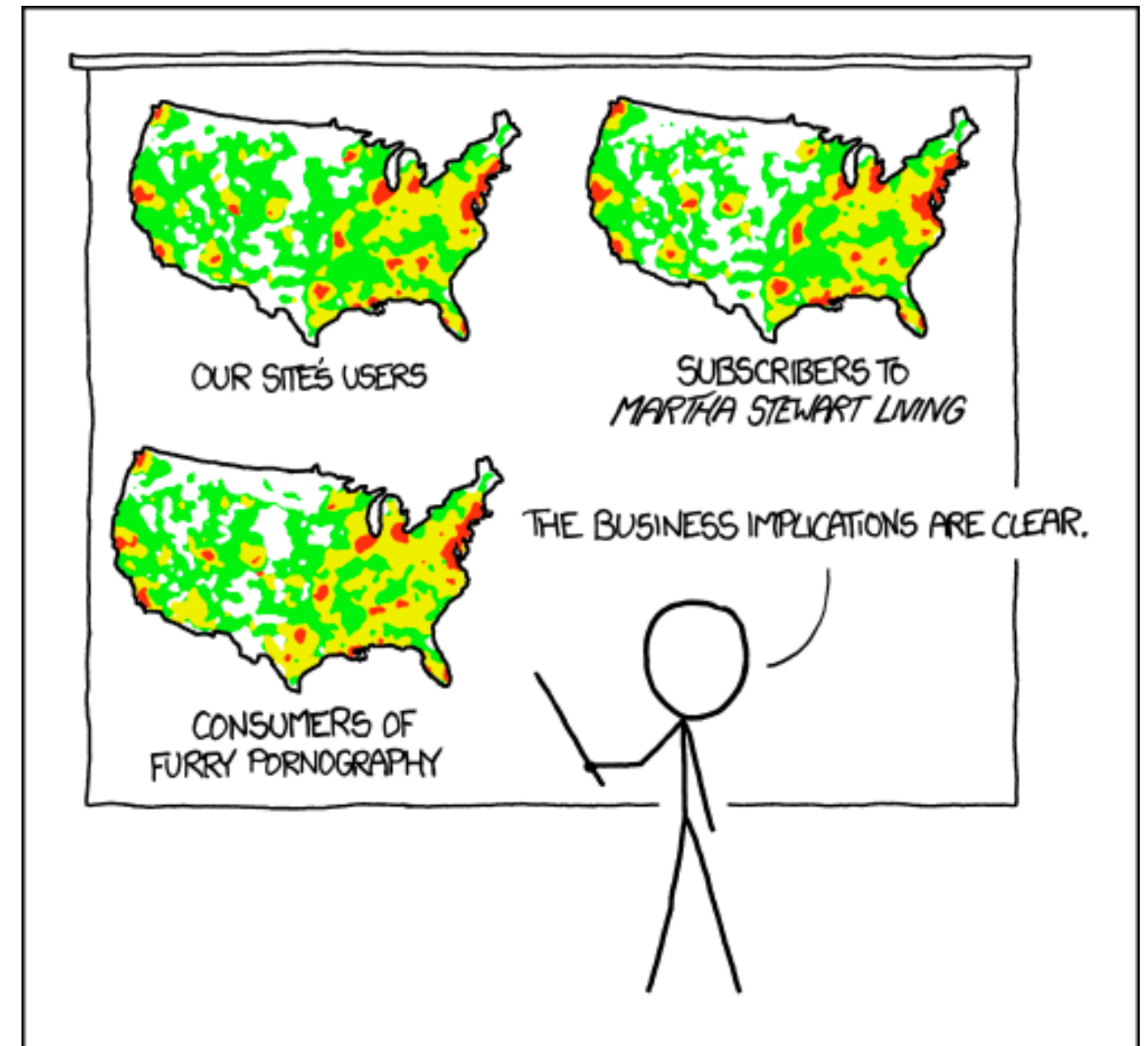
- *use* given spatial data
 - when central task is understanding spatial relationships
- data
 - geographic geometry
 - table with 1 quant attribute per region
- encoding
 - use given geometry for area mark boundaries
 - sequential segmented colormap *[more later]*



<http://bl.ocks.org/mbostock/4060606>

Beware: Population maps trickiness!

- consider when to normalize by population density
- general issue
 - absolute counts vs relative/normalized data

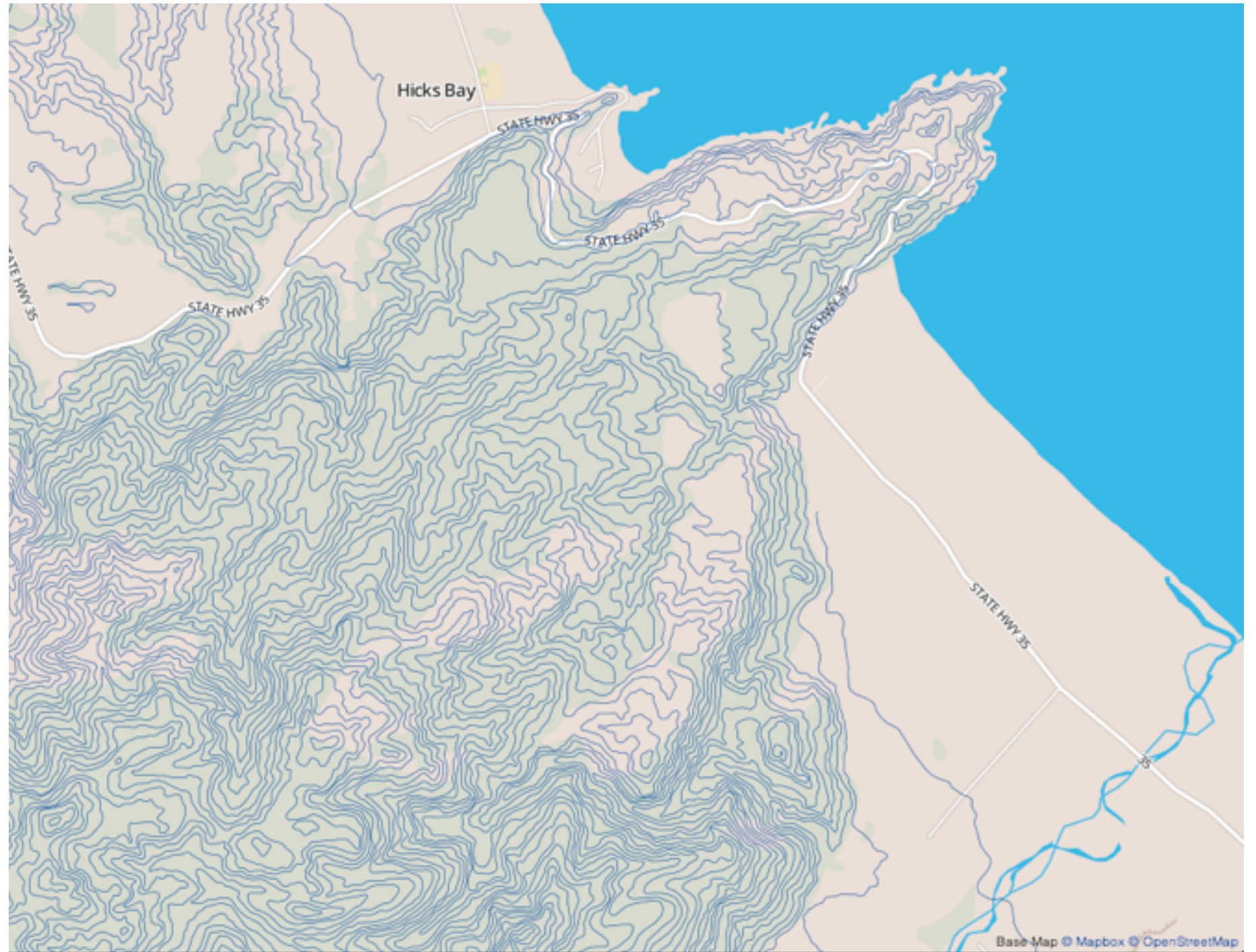


PET PEEVE #208:
GEOGRAPHIC PROFILE MAPS WHICH ARE
BASICALLY JUST POPULATION MAPS

[<https://xkcd.com/1138>]

Idiom: **topographic map**

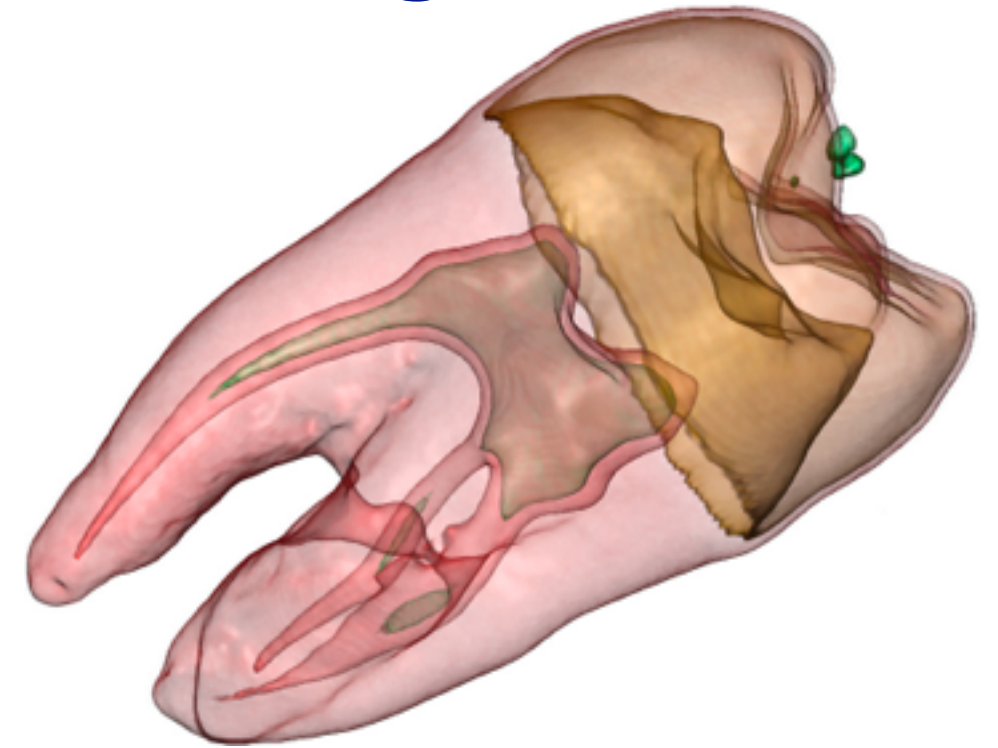
- data
 - geographic geometry
 - scalar spatial field
 - 1 quant attribute per grid cell
- derived data
 - isoline geometry
 - isocontours computed for specific levels of scalar values



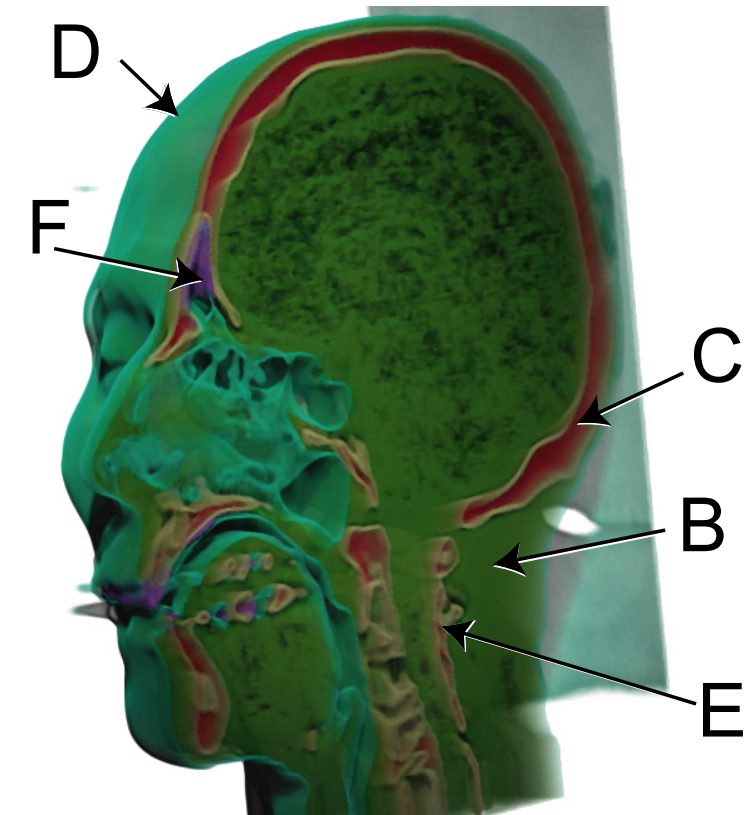
Land Information New Zealand Data Service

Idioms: **isosurfaces**, **direct volume rendering**

- **data**
 - scalar spatial field
 - 1 quant attribute per grid cell
- **task**
 - shape understanding, spatial relationships
- **isosurface**
 - derived data: isocontours computed for specific levels of scalar values
- **direct volume rendering**
 - transfer function maps scalar values to color, opacity
 - no derived geometry



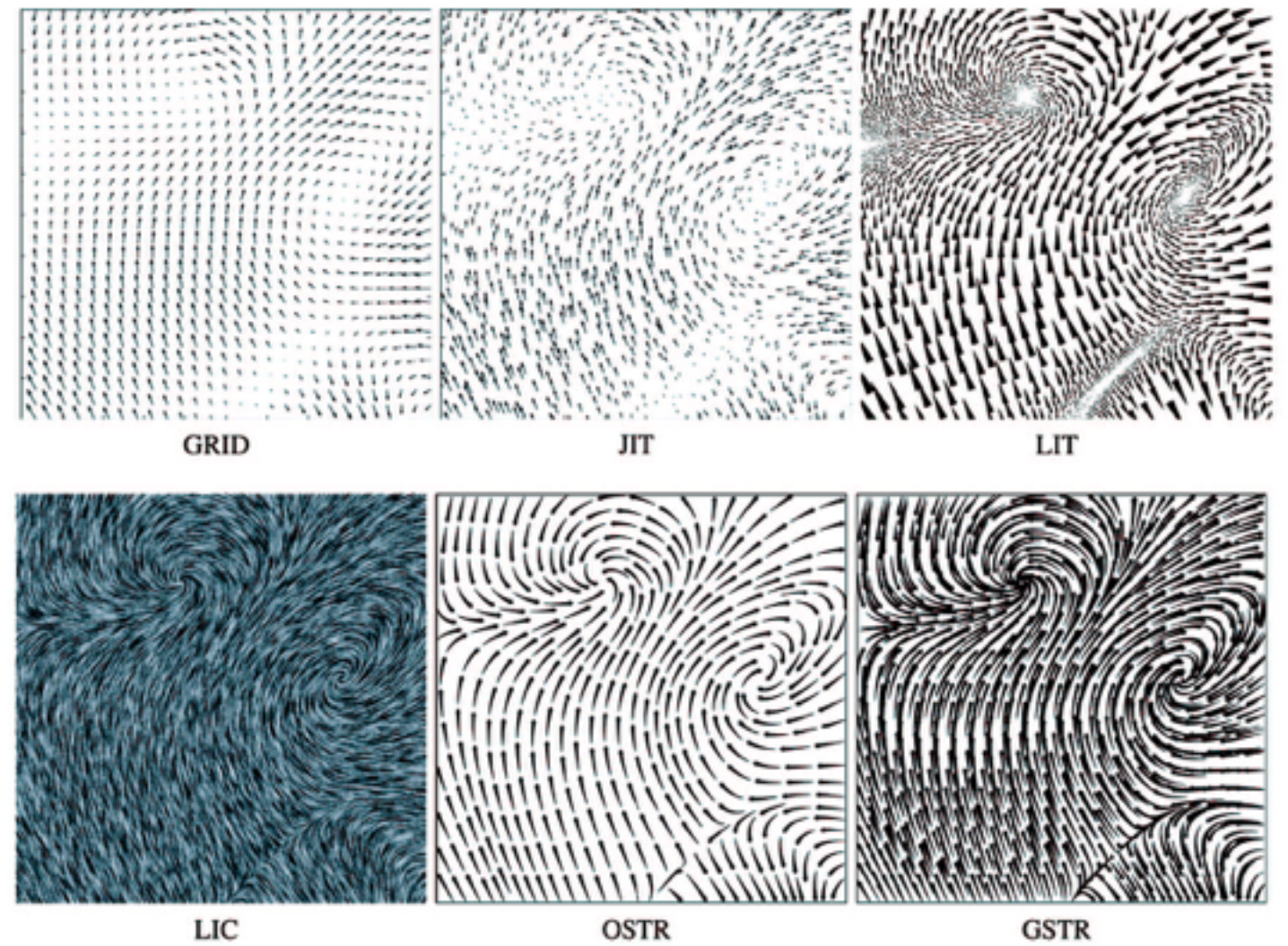
[Interactive Volume Rendering Techniques. Kniss. Master's thesis, University of Utah Computer Science, 2002.]



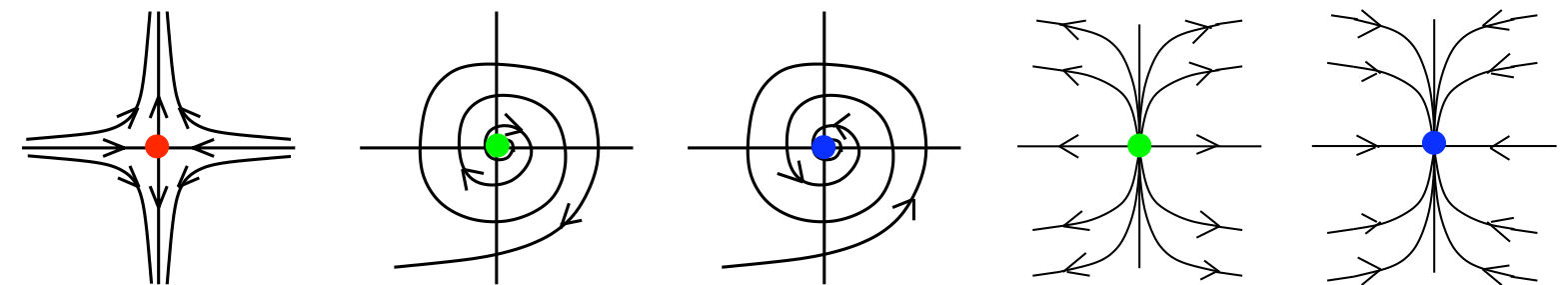
[Multidimensional Transfer Functions for Volume Rendering. Kniss, Kindlmann, and Hansen. In The Visualization Handbook, edited by Charles Hansen and Christopher Johnson, pp. 189–210. Elsevier, 2005.]

Vector and tensor fields

- data
 - many attribs per cell
- idiom families
 - flow glyphs
 - purely local
 - geometric flow
 - derived data from tracing particle trajectories
 - sparse set of seed points
 - texture flow
 - derived data, dense seeds
 - feature flow
 - global computation to detect features
 - encoded with one of methods above



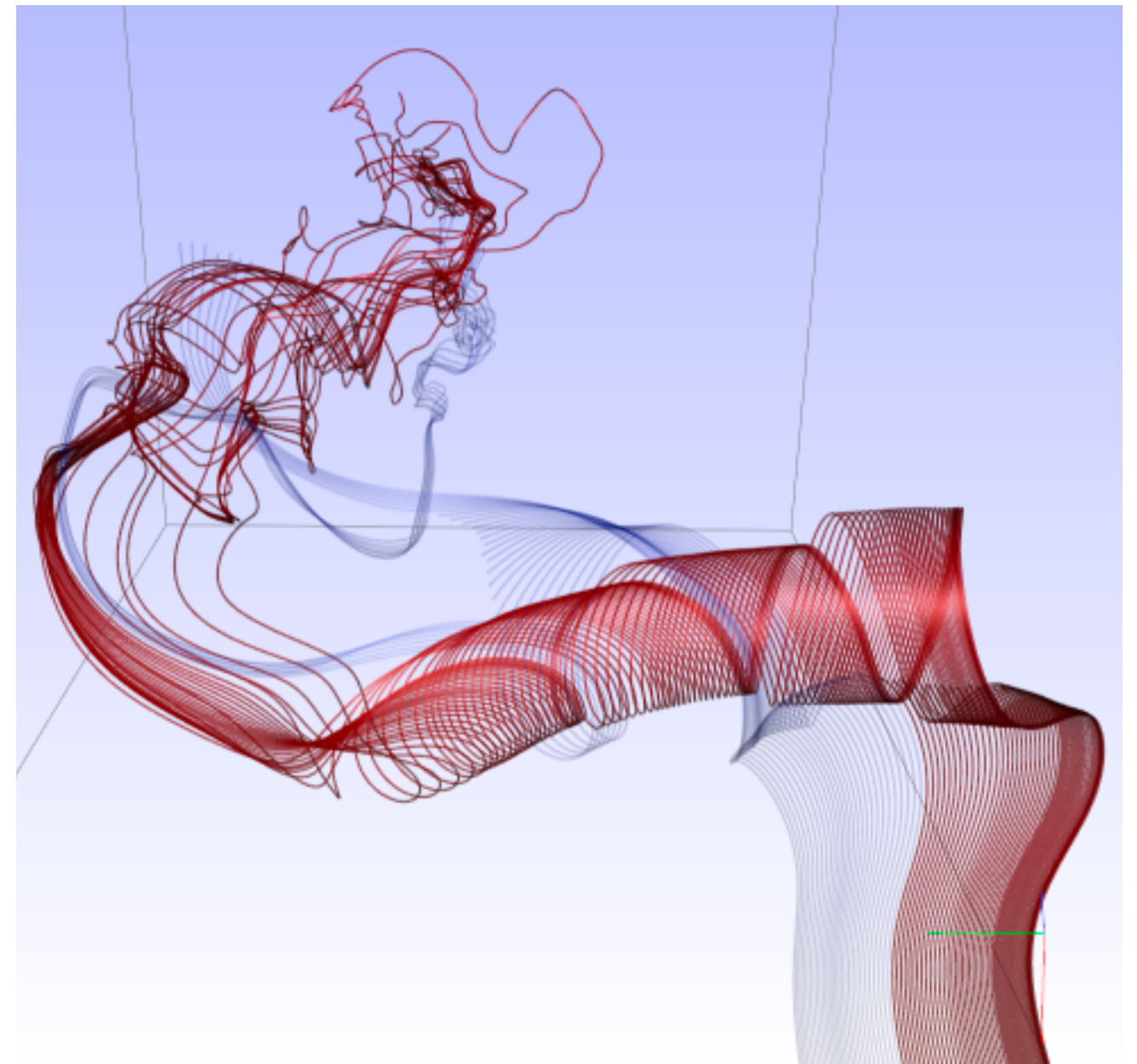
[Comparing 2D vector field visualization methods: A user study. Laidlaw et al. IEEE Trans. Visualization and Computer Graphics (TVCG) 11:1 (2005), 59–70.]



[Topology tracking for the visualization of time-dependent two-dimensional flows. Tricoche, Wischgoll, Scheuermann, and Hagen. Computers & Graphics 26:2 (2002), 249–257.]

Idiom: **similarity-clustered streamlines**

- data
 - 3D vector field
- derived data (from field)
 - streamlines: trajectory particle will follow
- derived data (per streamline)
 - curvature, torsion, tortuosity
 - signature: complex weighted combination
 - compute cluster hierarchy across all signatures
 - encode: color and opacity by cluster
- tasks
 - find features, query shape
- scalability
 - millions of samples, hundreds of streamlines



[Similarity Measures for Enhancing Interactive Streamline Seeding. McLoughlin, Jones, Laramee, Malki, Masters, and Hansen. IEEE Trans. Visualization and Computer Graphics 19:8 (2013), 1342–1353.]

Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Oct 2014.
–*Chap 8: Arrange Spatial Data*
- How Maps Work: Representation, Visualization, and Design. MacEachren. Guilford Press, 1995.
- Overview of visualization. Schroeder and. Martin. In The Visualization Handbook, edited by Charles Hansen and Christopher Johnson, pp. 3–39. Elsevier, 2005.
- Real-Time Volume Graphics. Engel, Hadwiger, Kniss, Reza-Salama, and Weiskopf. AK Peters, 2006.
- Overview of flow visualization. Weiskopf and Erlebacher. In The Visualization Handbook, edited by Charles Hansen and Christopher Johnson, pp. 261–278. Elsevier, 2005.

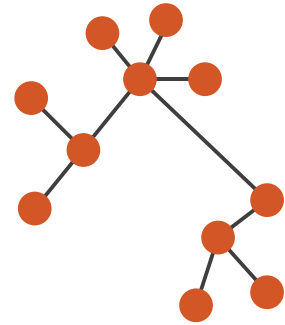
Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - **Arrange Networks and Trees**
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

Arrange networks and trees

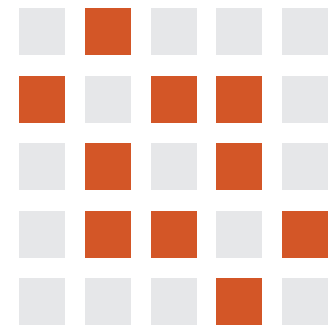
→ Node–Link Diagrams Connection Marks

✓ NETWORKS ✓ TREES



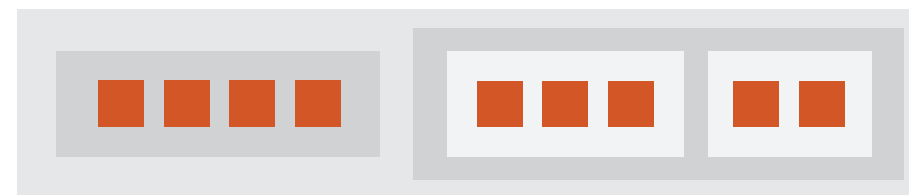
→ Adjacency Matrix Derived Table

✓ NETWORKS ✓ TREES



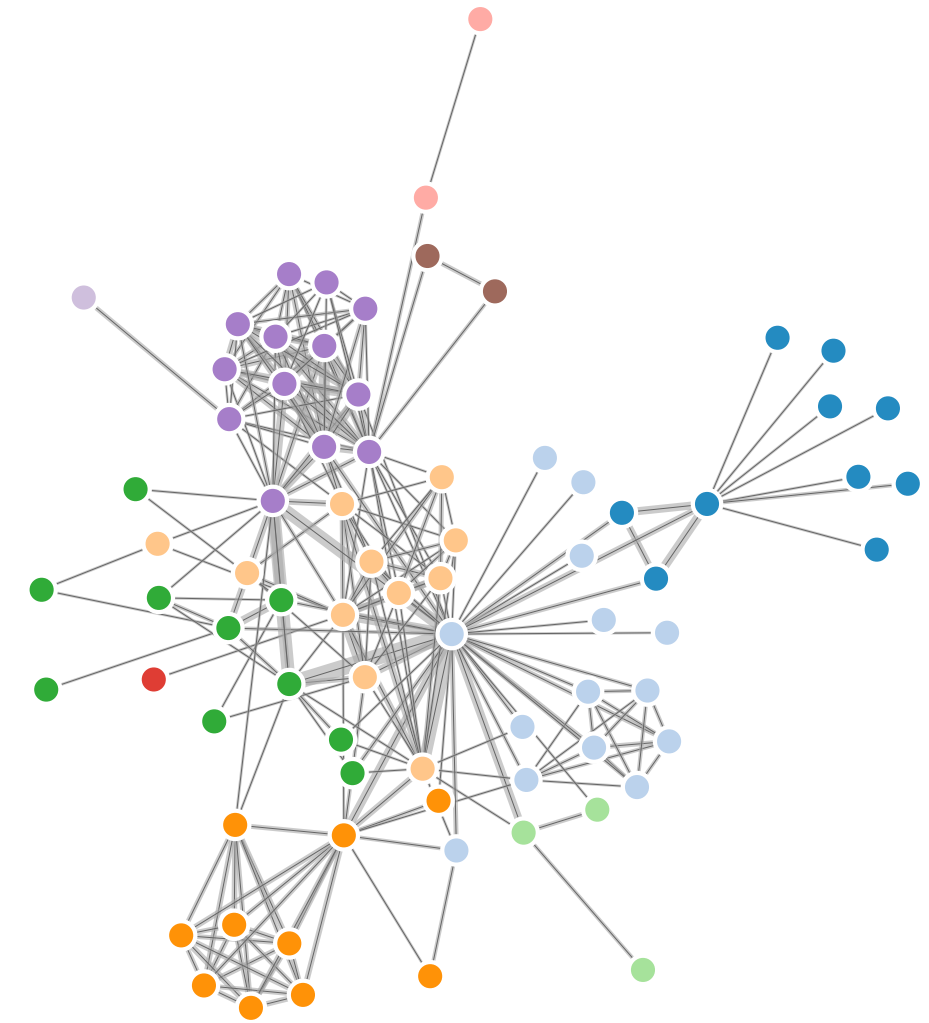
→ Enclosure Containment Marks

✗ NETWORKS ✓ TREES



Idiom: **force-directed placement**

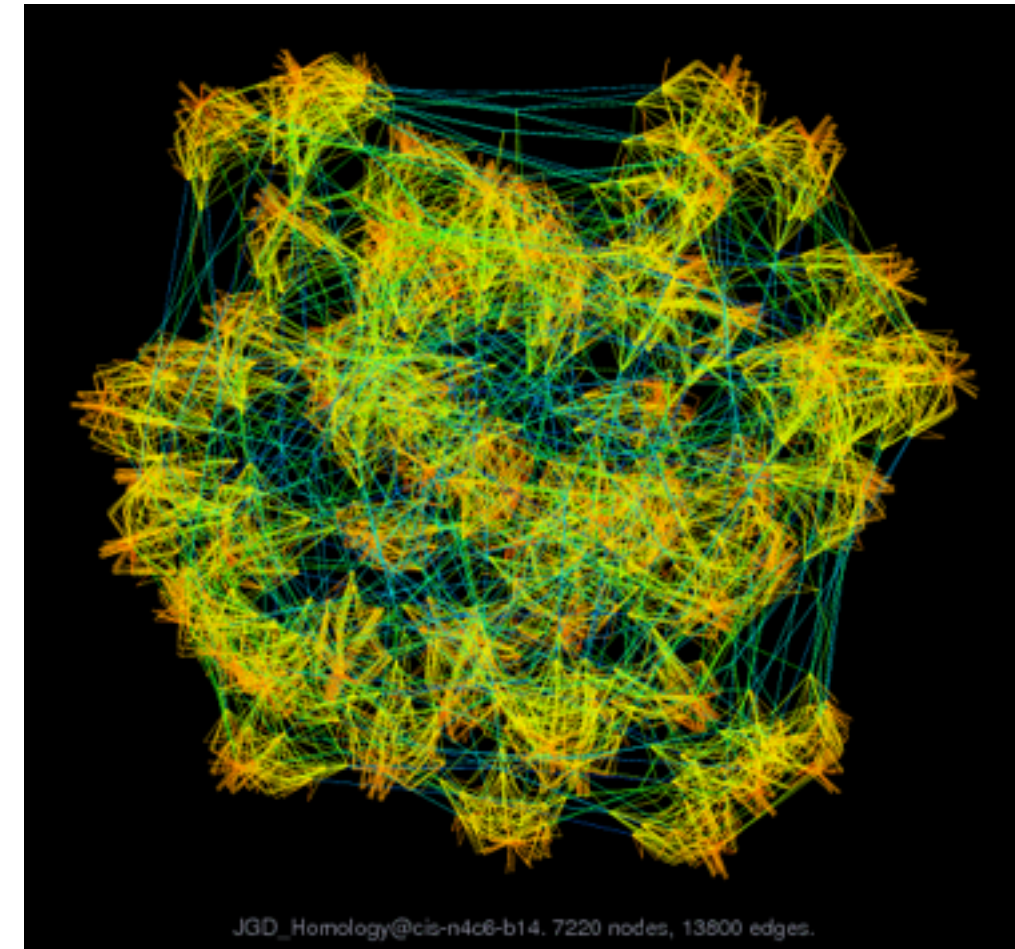
- visual encoding: node-link diagram
 - link connection marks, node point marks
- algorithm: energy minimization
 - analogy: nodes repel, links draw together like springs
 - optimization problem: minimize crossings
- spatial position: no meaning directly encoded
 - sometimes proximity meaningful
 - sometimes proximity arbitrary, artifact of layout algorithm
- tasks
 - explore topology; locate paths, clusters
- scalability
 - node/edge density $E < 4N$



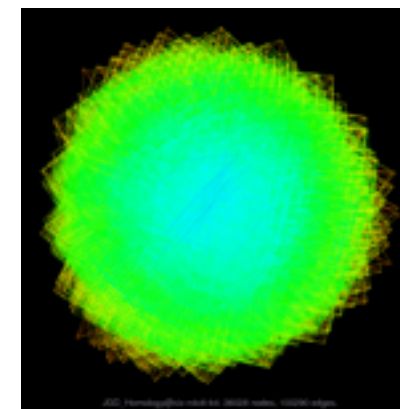
<http://mbostock.github.com/d3/ex/force.html>

Idiom: **sfdp** (multi-level force-directed placement)

- data
 - original: network
 - derived: cluster hierarchy atop it
- considerations
 - better algorithm for same encoding technique
 - same: fundamental use of space
 - hierarchy used for algorithm speed/quality but not shown explicitly
- scalability
 - nodes, edges: 1K-10K
 - hairball problem still hits eventually



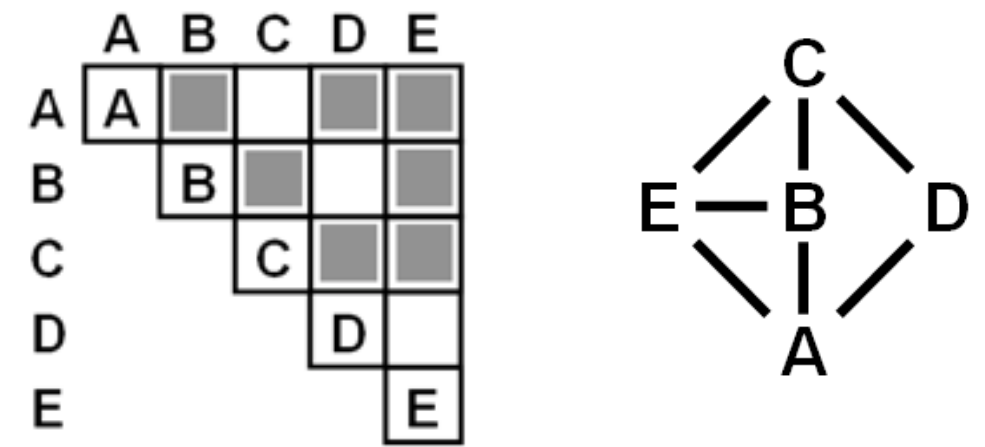
[Efficient and high quality force-directed graph drawing. Hu. The Mathematica Journal 10:37–71, 2005.]



<http://www.research.att.com/yifanhu/GALLERY/GRAPHS/index1.html>

Idiom: adjacency matrix view

- data: network
 - transform into same data/encoding as heatmap
- derived data: table from network
 - 1 quant attrib
 - weighted edge between nodes
 - 2 categ attribs: node list x 2
- visual encoding
 - cell shows presence/absence of edge
- scalability
 - 1K nodes, 1M edges



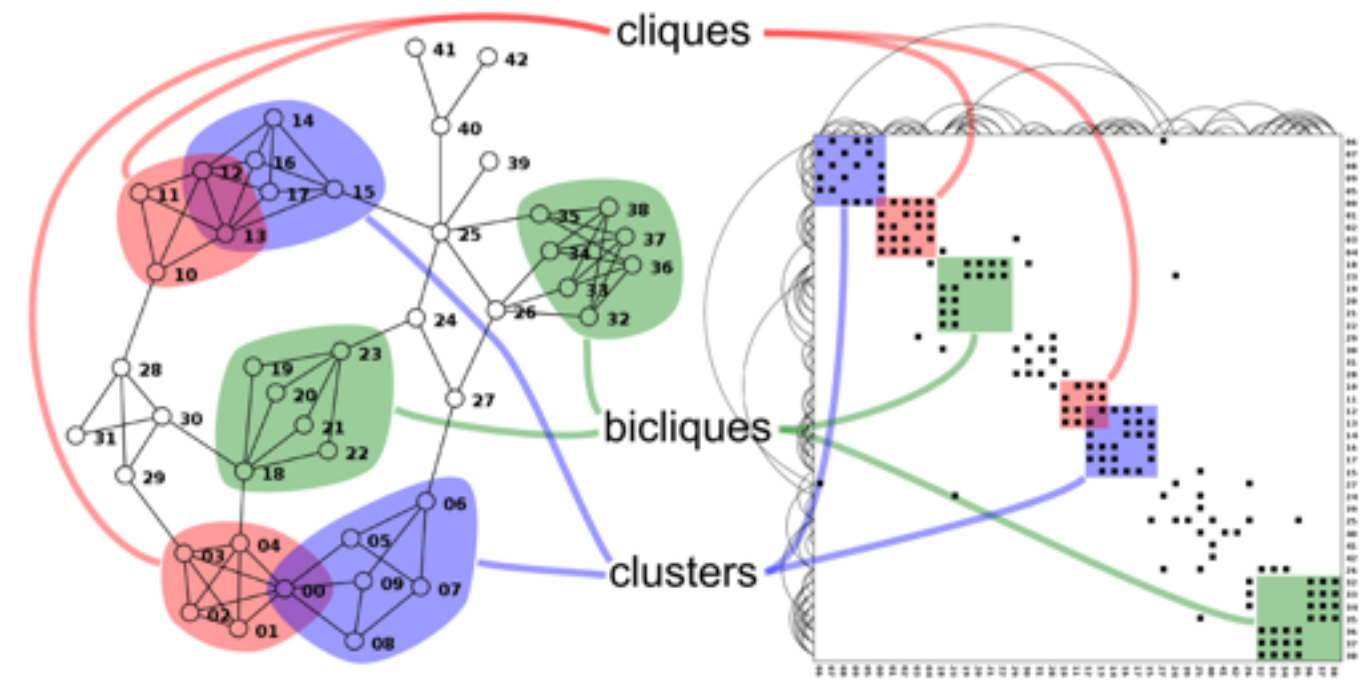
[NodeTrix: a Hybrid Visualization of Social Networks. Henry, Fekete, and McGuffin. IEEE TVCG (Proc. InfoVis) 13(6):1302-1309, 2007.]



[Points of view: Networks. Gehlenborg and Wong. Nature Methods 9:115.]

Connection vs. adjacency comparison

- adjacency matrix strengths
 - predictability, scalability, supports reordering
 - some topology tasks trainable
- node-link diagram strengths
 - topology understanding, path tracing
 - intuitive, no training needed
- empirical study
 - node-link best for small networks
 - matrix best for large networks
 - if tasks don't involve topological structure!

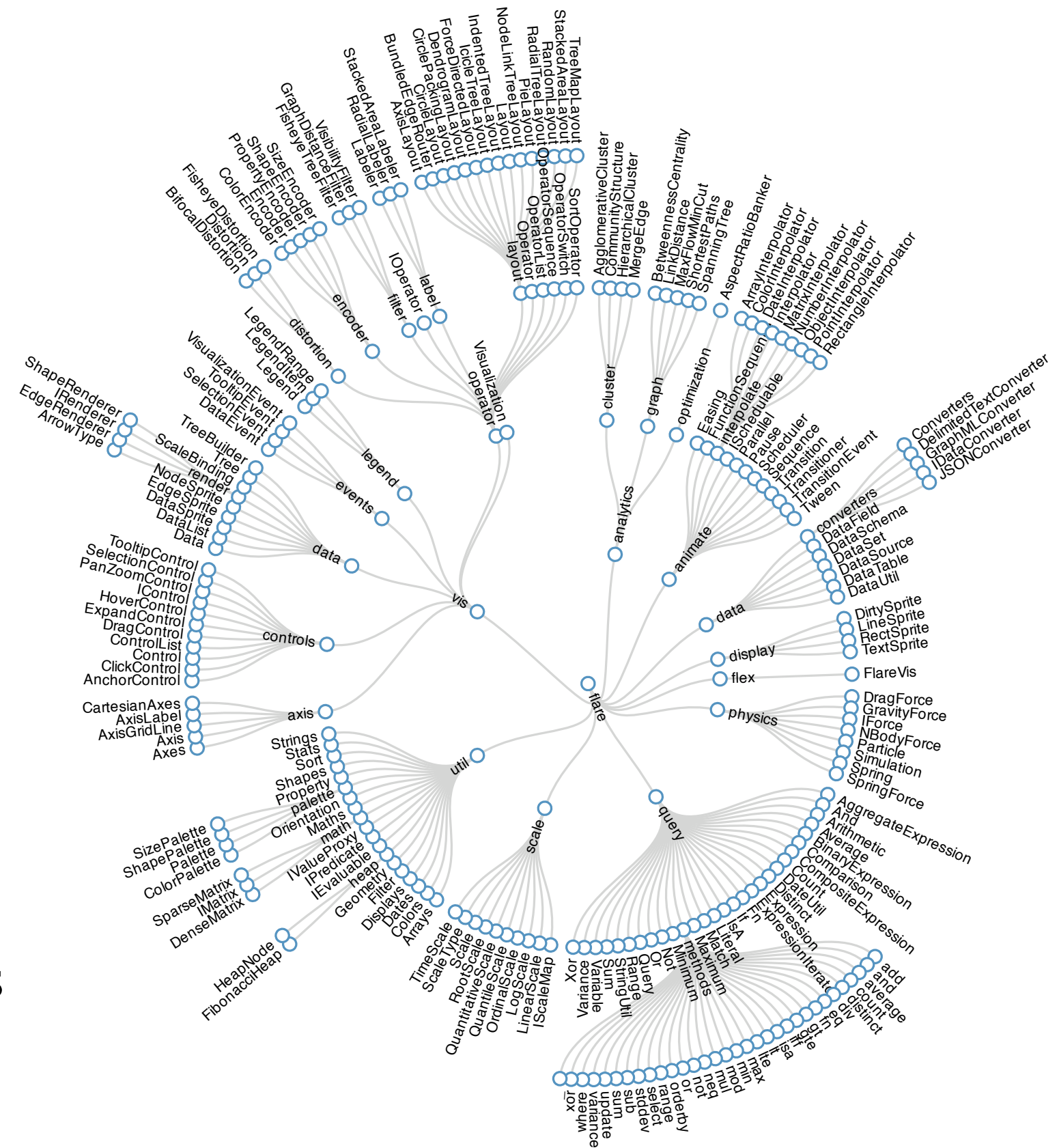


<http://www.michaelmcguffin.com/courses/vis/patternsInAdjacencyMatrix.png>

[On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis. Ghoniem, Fekete, and Castagliola. Information Visualization 4:2 (2005), 114–135.]

Idiom: radial node-link tree

- data
 - tree
- encoding
 - link connection marks
 - point node marks
 - radial axis orientation
 - angular proximity: siblings
 - distance from center: depth in tree
- tasks
 - understanding topology, following paths
- scalability
 - 1K - 10K nodes



Idiom: **treemap**

- data
 - tree
 - 1 quant attrib at leaf nodes
- encoding
 - area containment marks for hierarchical structure
 - rectilinear orientation
 - size encodes quant attrib
- tasks
 - query attribute at leaf nodes
- scalability
 - 1M leaf nodes

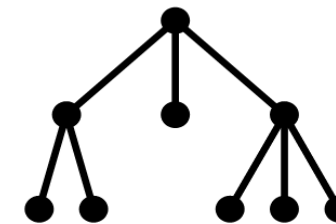
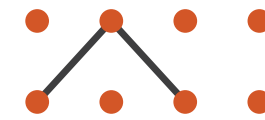


http://tulip.labri.fr/Documentation/3_7/userHandbook/html/ch06.html

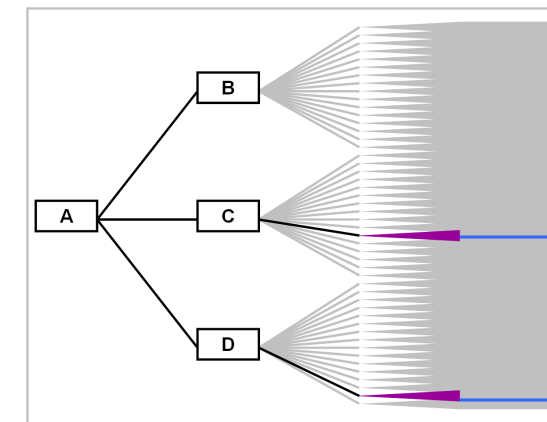
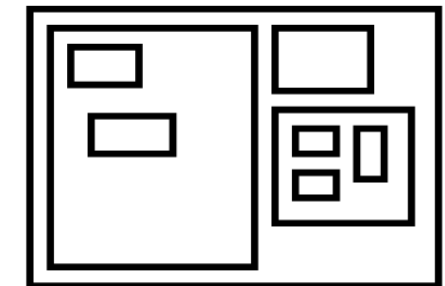
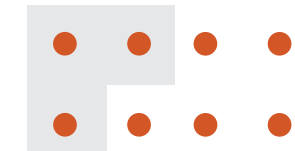
Link marks: Connection and containment

- marks as links (vs. nodes)
 - common case in network drawing
 - 1D case: connection
 - ex: all node-link diagrams
 - emphasizes topology, path tracing
 - networks and trees
 - 2D case: containment
 - ex: all treemap variants
 - emphasizes attribute values at leaves (size coding)
 - only trees

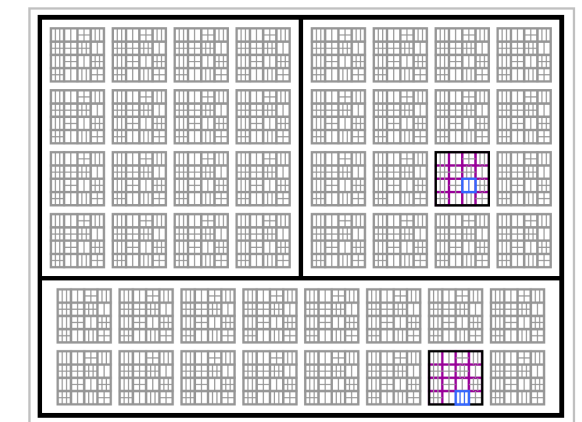
➔ Connection



➔ Containment



Node-Link Diagram



Treemap

[Elastic Hierarchies: Combining Treemaps and Node-Link Diagrams. Dong, McGuffin, and Chignell. Proc. InfoVis 2005, p. 57-64.]

Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
–*Chap 9: Arrange Networks and Trees*
- Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. von Landesberger et al. Computer Graphics Forum 30:6 (2011), 1719–1749.
- Simple Algorithms for Network Visualization: A Tutorial. McGuffin. Tsinghua Science and Technology (Special Issue on Visualization and Computer Graphics) 17:4 (2012), 383–398.
- Drawing on Physical Analogies. Brandes. In Drawing Graphs: Methods and Models, LNCS Tutorial, 2025, edited by M. Kaufmann and D. Wagner, LNCS Tutorial, 2025, pp. 71–86. Springer-Verlag, 2001.
- <http://www.treevis.net> Treevis.net: A Tree Visualization Reference. Schulz. IEEE Computer Graphics and Applications 31:6 (2011), 11–15.
- Perceptual Guidelines for Creating Rectangular Treemaps. Kong, Heer, and Agrawala. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis) 16:6 (2010), 990–998.

Outline

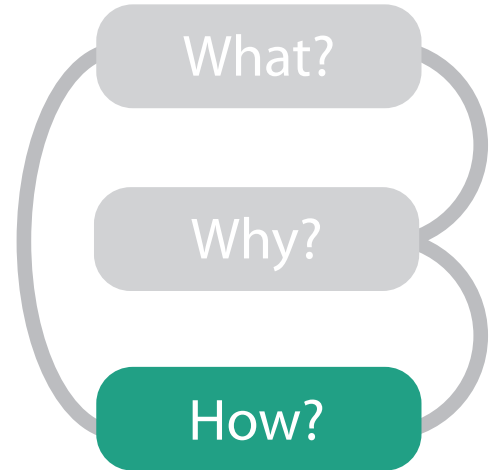
- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context
- **Break** 3:40-4:15pm

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - **Map Color and Other Channels**
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

Idiom design choices: First half

Encode

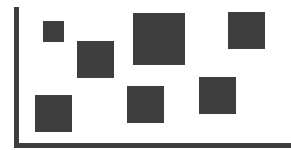


→ Arrange

→ Express



→ Separate



→ Order



→ Align



→ Use



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape



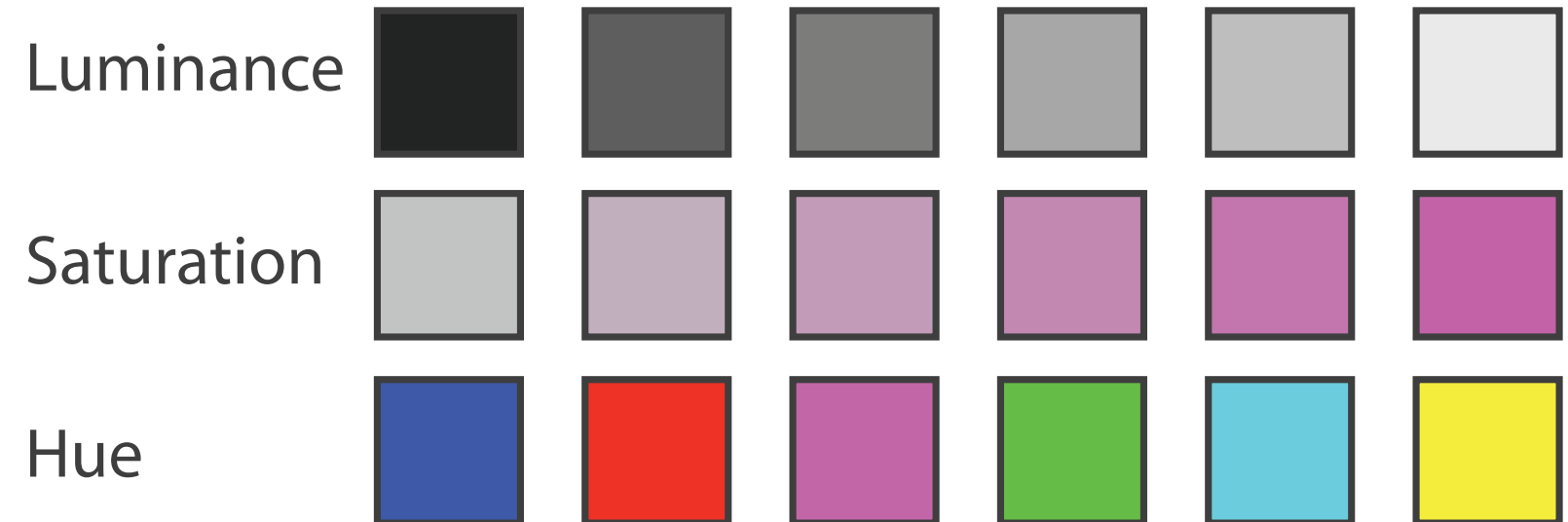
→ Motion

Direction, Rate, Frequency, ...



Decomposing color

- first rule of color: do not talk about color!
 - color is confusing if treated as monolithic
- decompose into three channels
 - ordered can show magnitude
 - luminance
 - saturation
 - categorical can show identity
 - hue



Luminance

- need luminance for edge detection
 - fine-grained detail only visible through luminance contrast
 - legible text requires luminance contrast!
- HLS better than RGB for encoding but beware
 - L lightness \neq L^* luminance



Lightness information



Color information



Corners of the RGB color cube



L from HLS

All the same

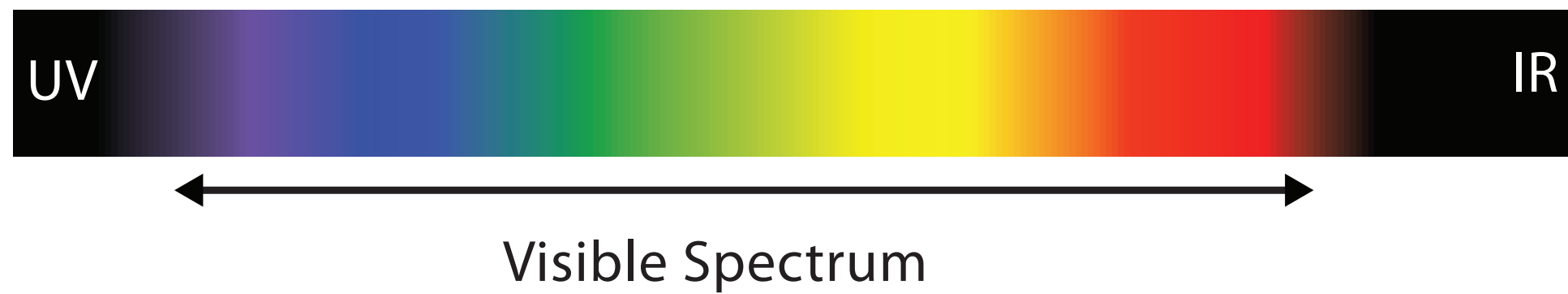
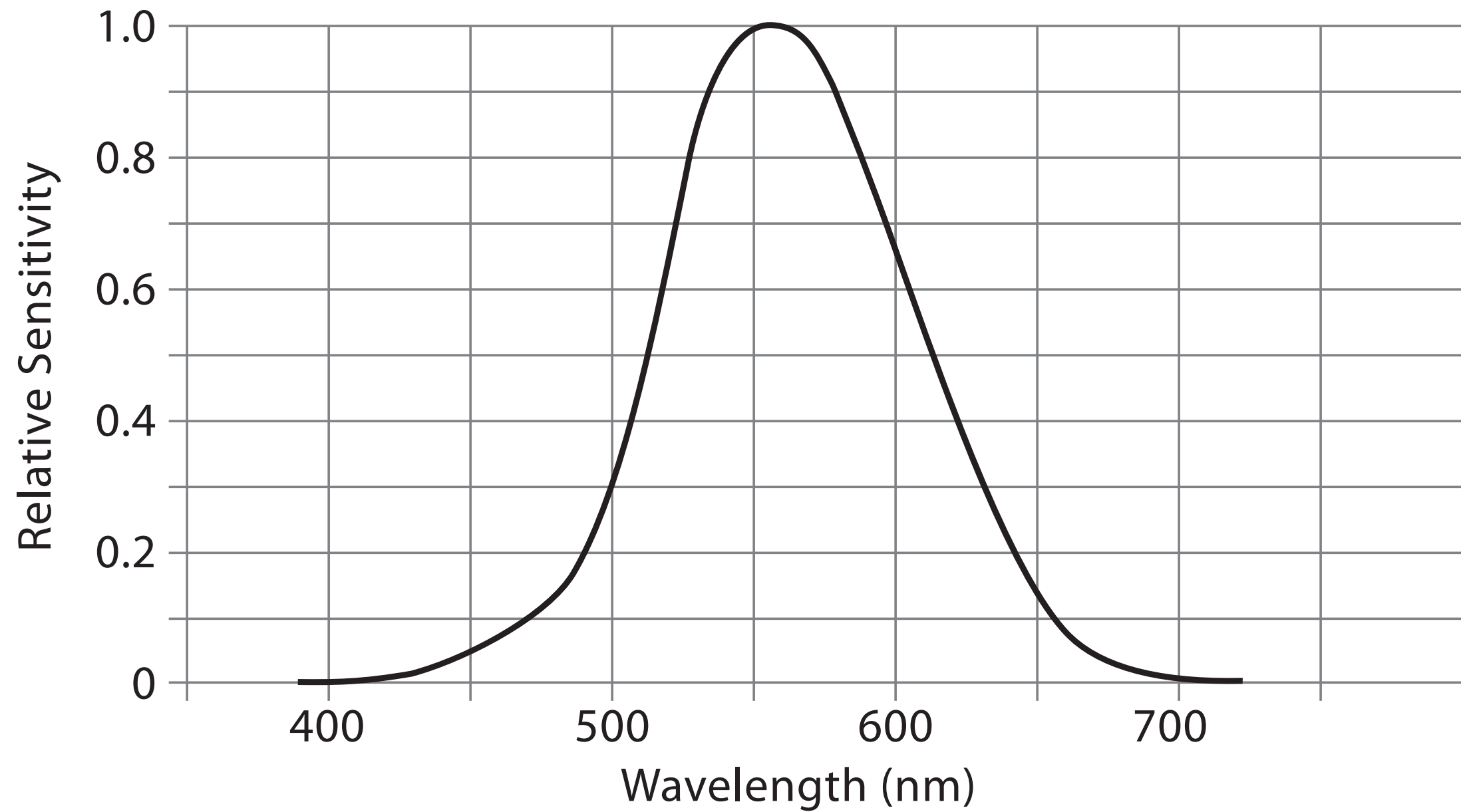


Luminance values



[Seriously Colorful: Advanced Color Principles & Practices. Stone.Tableau Customer Conference 2014.]

Spectral sensitivity



Colormaps

→ Categorical



→ Ordered

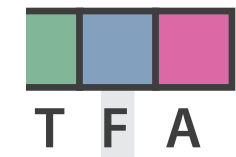
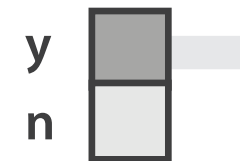
→ *Sequential*



→ *Diverging*

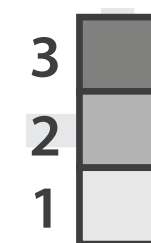
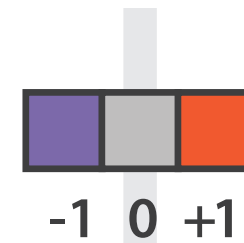


Binary



Categorical

Diverging



Sequential

after [Color Use Guidelines for Mapping and Visualization. Brewer, 1994.
<http://www.personal.psu.edu/faculty/c/a/cab38/ColorSch/Schemes.html>]

Colormaps

→ Categorical

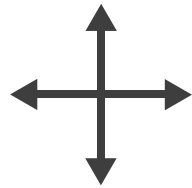


→ Ordered

→ *Sequential*



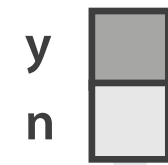
→ Bivariate



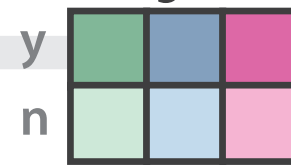
→ *Diverging*



Binary

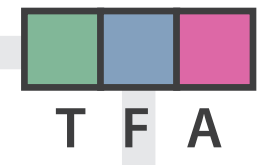


Categorical



Binary

Categorical

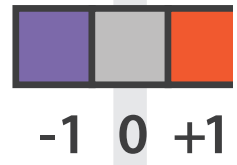


Diverging

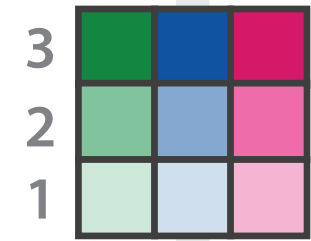


Binary

Diverging

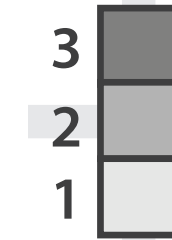


Categorical



Sequential

Sequential



after [Color Use Guidelines for Mapping and Visualization. Brewer, 1994. <http://www.personal.psu.edu/faculty/c/a/cab38/ColorSch/Schemes.html>]

Colormaps

→ Categorical



→ Ordered

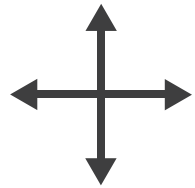
→ Sequential



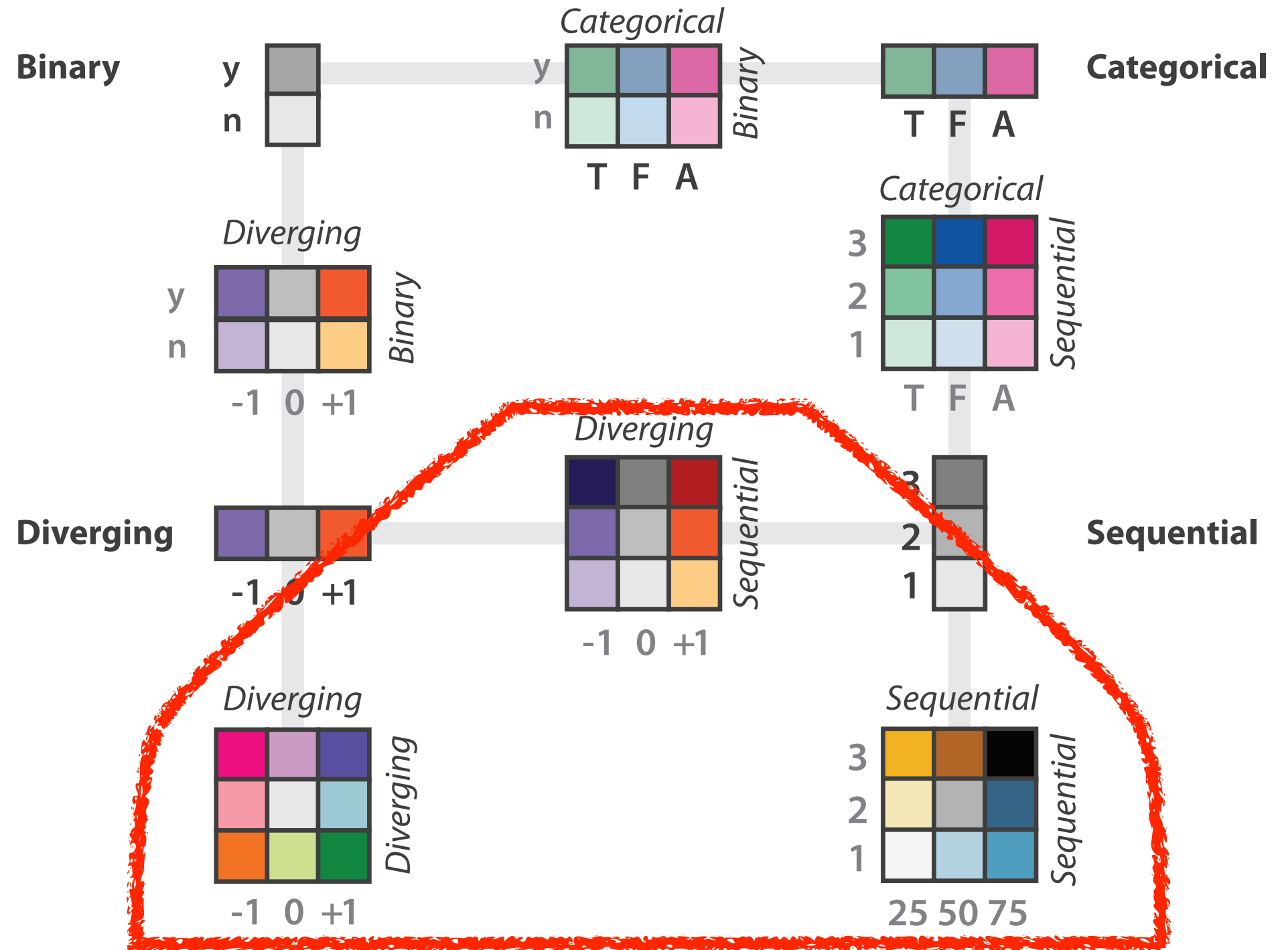
→ Diverging



→ Bivariate



use with care if more than two levels (binary)!



after [Color Use Guidelines for Mapping and Visualization. Brewer, 1994. <http://www.personal.psu.edu/faculty/cl/cab38/ColorSch/Schemes.html>]

Colormaps

→ Categorical



→ Ordered

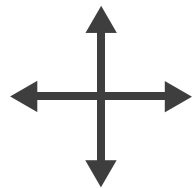
→ Sequential



→ Diverging



→ Bivariate



- color channel interactions

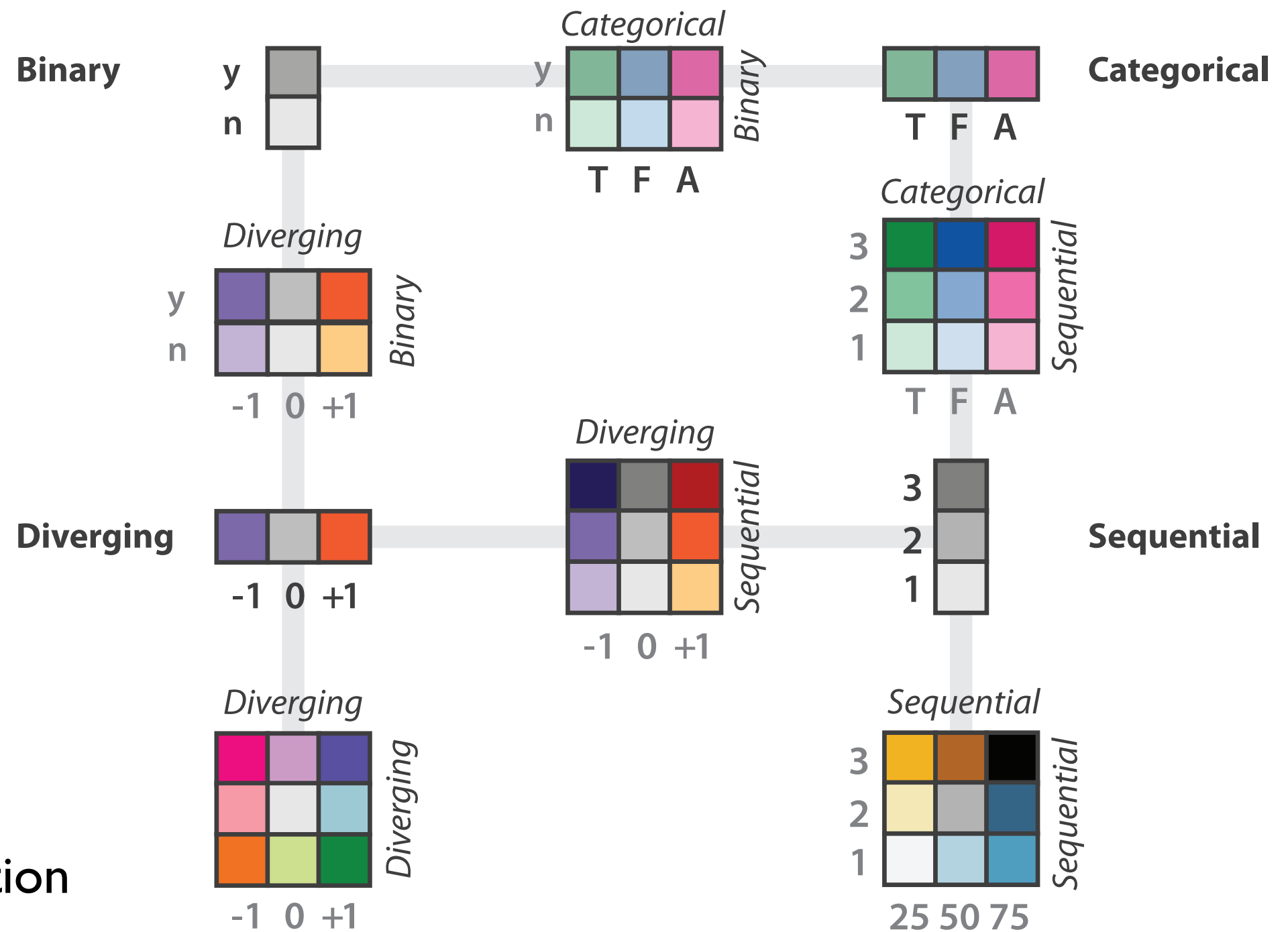
- size heavily affects salience

- small regions need high saturation

- large need low saturation

- saturation & luminance: 3-4 bins max

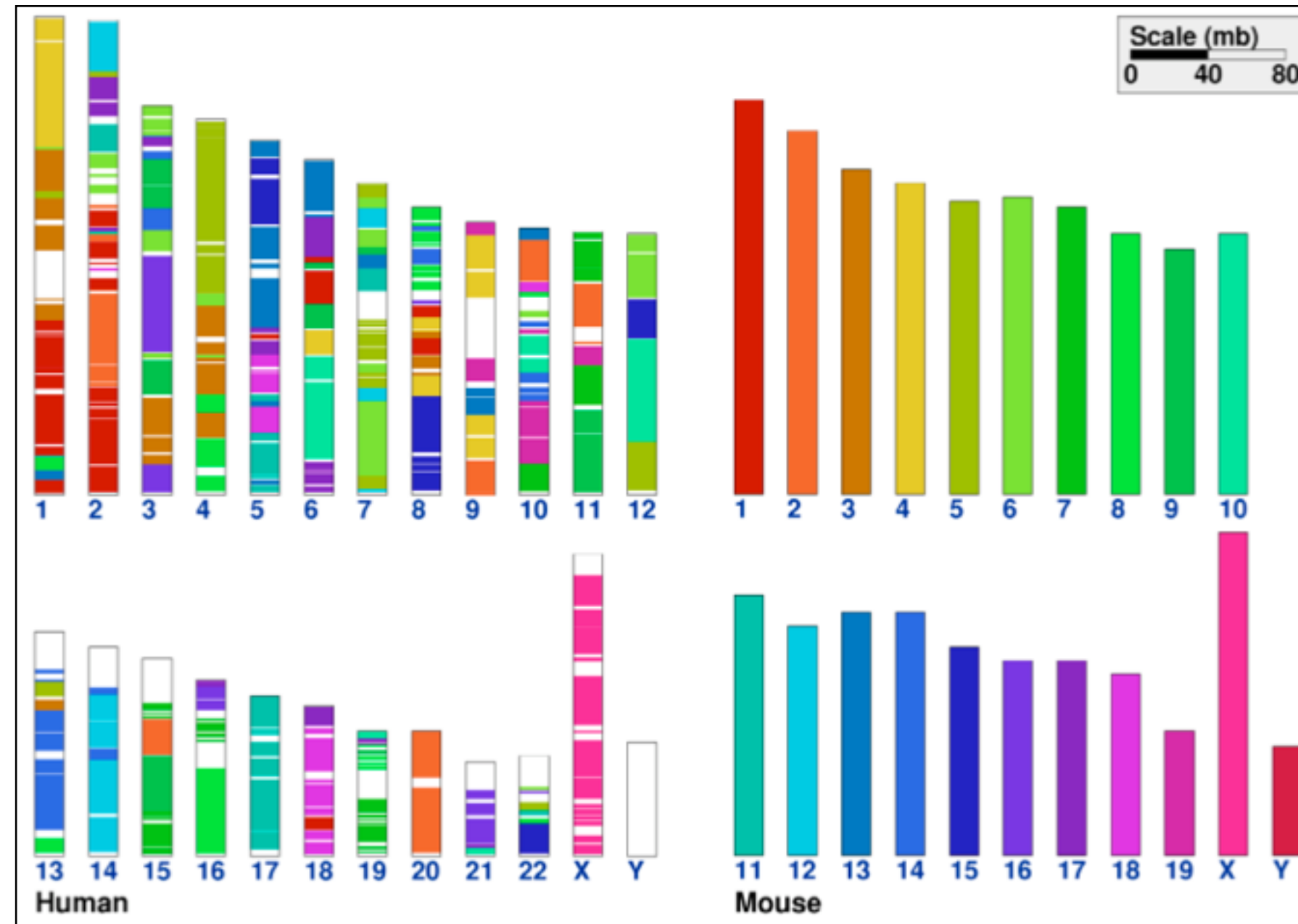
- also not separable from transparency



after [Color Use Guidelines for Mapping and Visualization. Brewer, 1994. <http://www.personal.psu.edu/faculty/cl/cab38/ColorSch/Schemes.html>]

Categorical color: Discriminability constraints

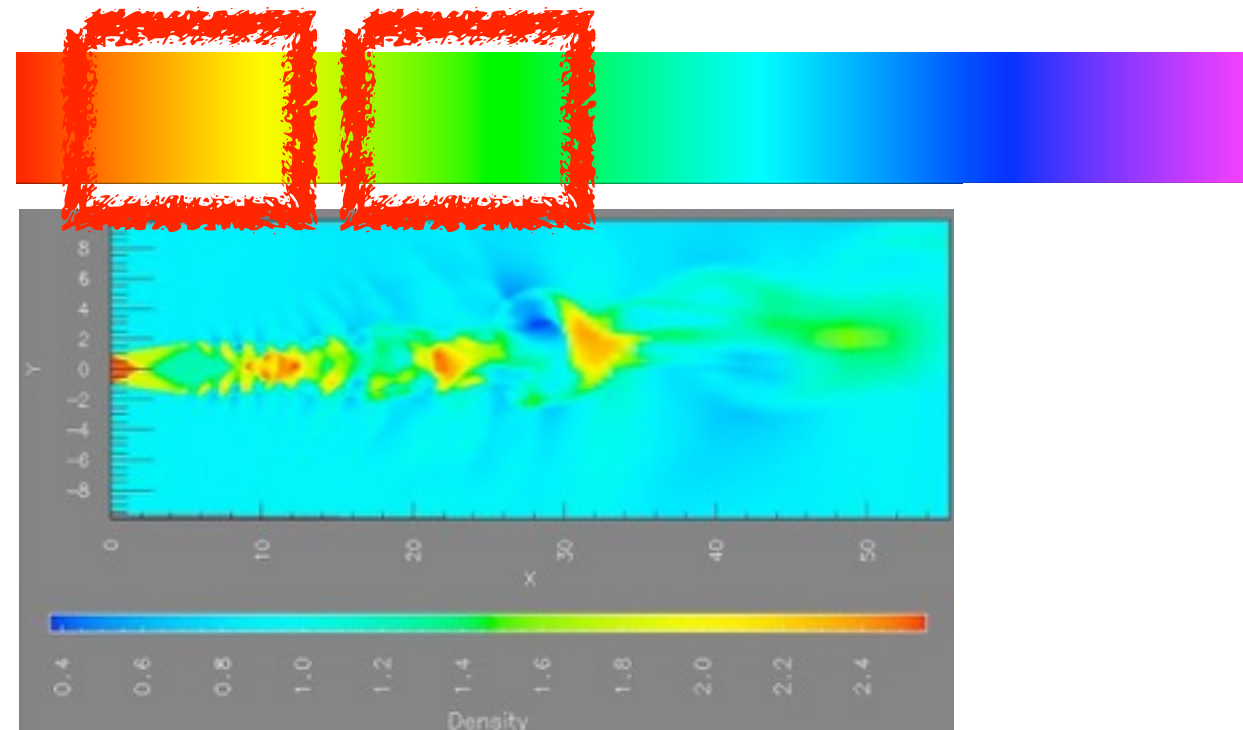
- noncontiguous small regions of color: only 6-12 bins



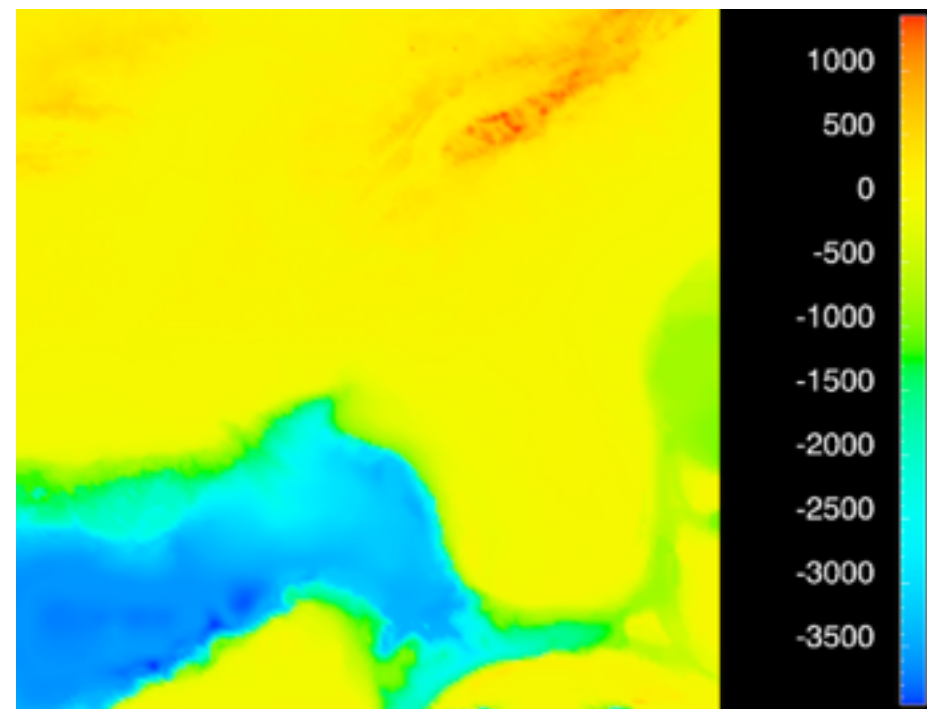
[Cinteny: flexible analysis and visualization of synteny and genome rearrangements in multiple organisms. Sinha and Meller. *BMC Bioinformatics*, 8:82, 2007.]

Ordered color: Rainbow is poor default

- problems
 - perceptually unordered
 - perceptually nonlinear
- benefits
 - fine-grained structure visible and nameable



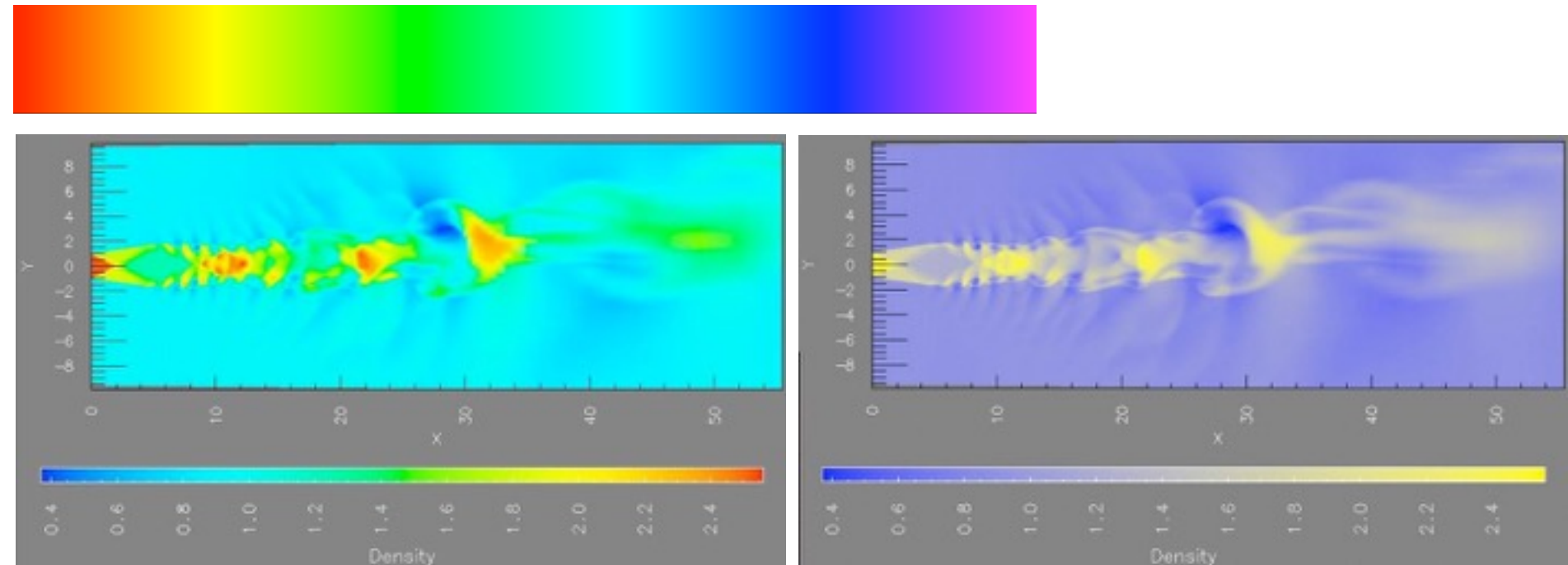
[A Rule-based Tool for Assisting Colormap Selection. Bergman, Rogowitz, and Treinish. Proc. IEEE Visualization (Vis), pp. 118–125, 1995.]



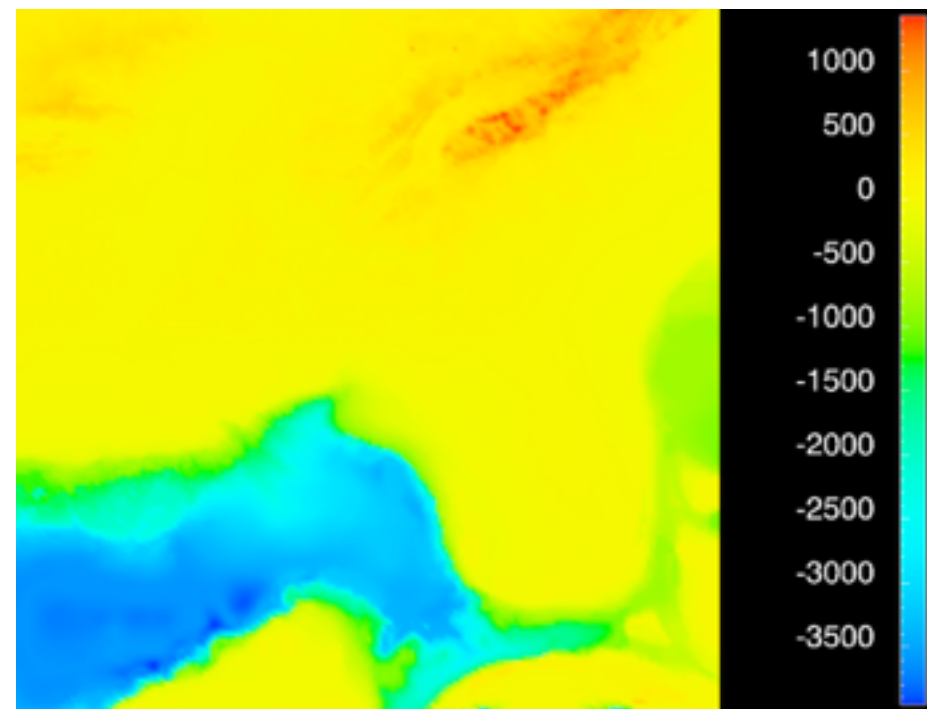
[Why Should Engineers Be Worried About Color? Treinish and Rogowitz 1998. <http://www.research.ibm.com/people/llloyd/color/color.HTM>]

Ordered color: Rainbow is poor default

- problems
 - perceptually unordered
 - perceptually nonlinear
- benefits
 - fine-grained structure visible and nameable
- alternatives
 - large-scale structure: fewer hues



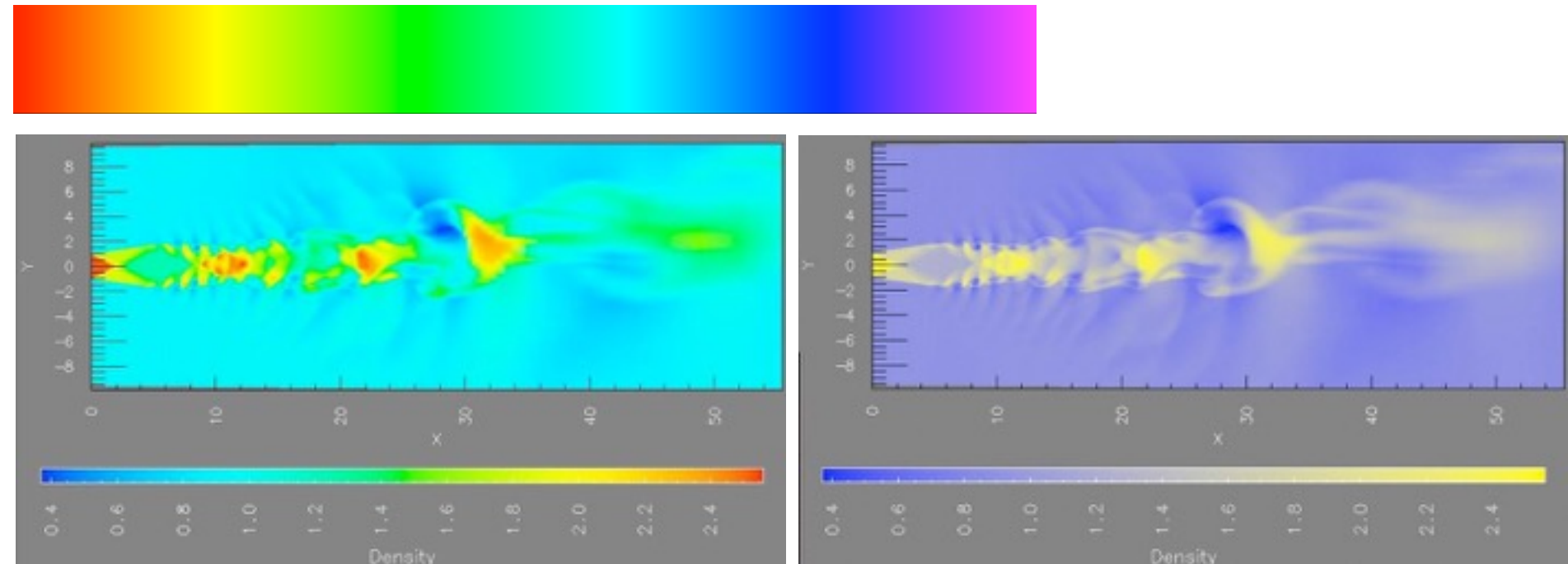
[A Rule-based Tool for Assisting Colormap Selection. Bergman, Rogowitz, and Treinish. Proc. IEEE Visualization (Vis), pp. 118–125, 1995.]



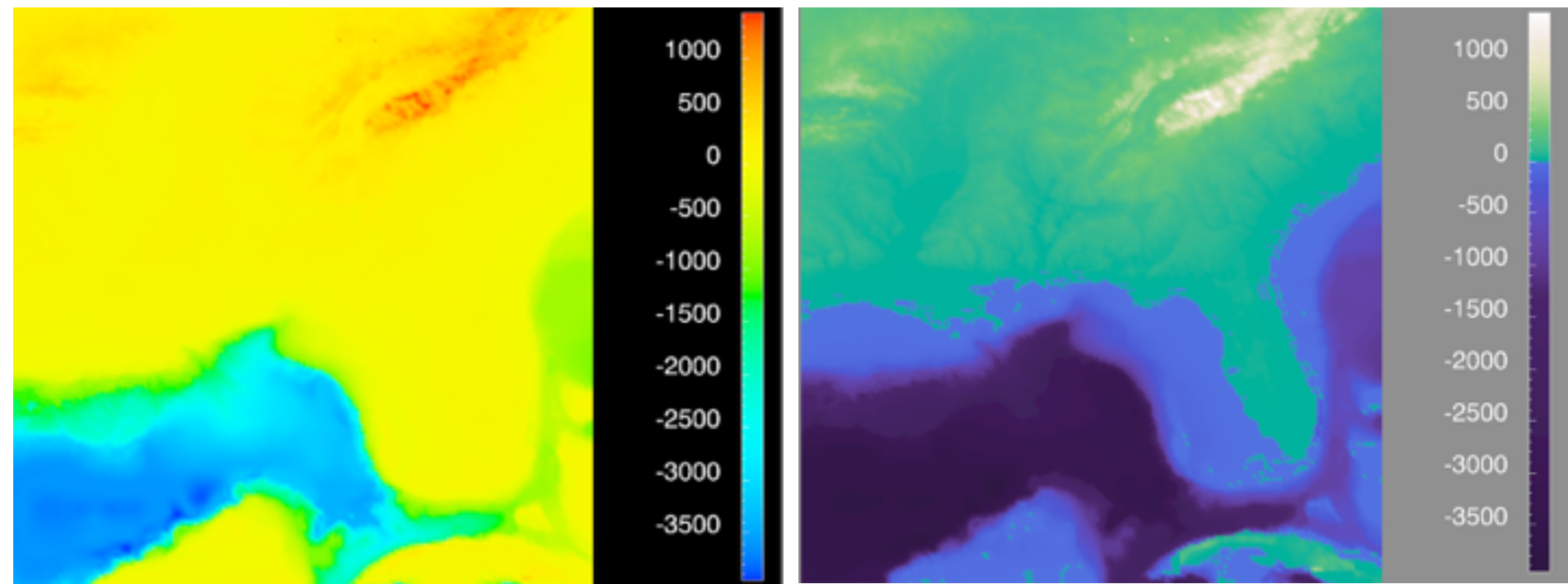
[Why Should Engineers Be Worried About Color? Treinish and Rogowitz 1998. <http://www.research.ibm.com/people/llloyd/color/color.HTM>]

Ordered color: Rainbow is poor default

- problems
 - perceptually unordered
 - perceptually nonlinear
- benefits
 - fine-grained structure visible and nameable
- alternatives
 - large-scale structure: fewer hues
 - fine structure: multiple hues with monotonically increasing luminance [eg viridis]



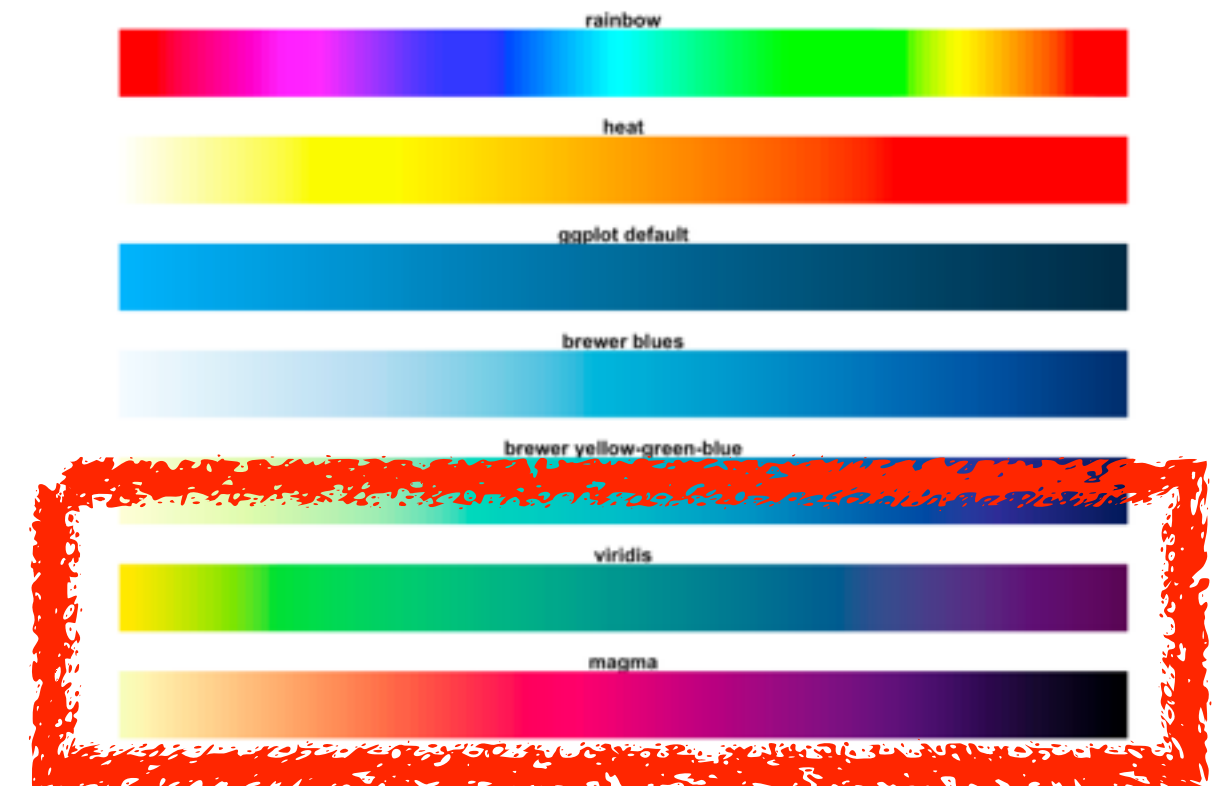
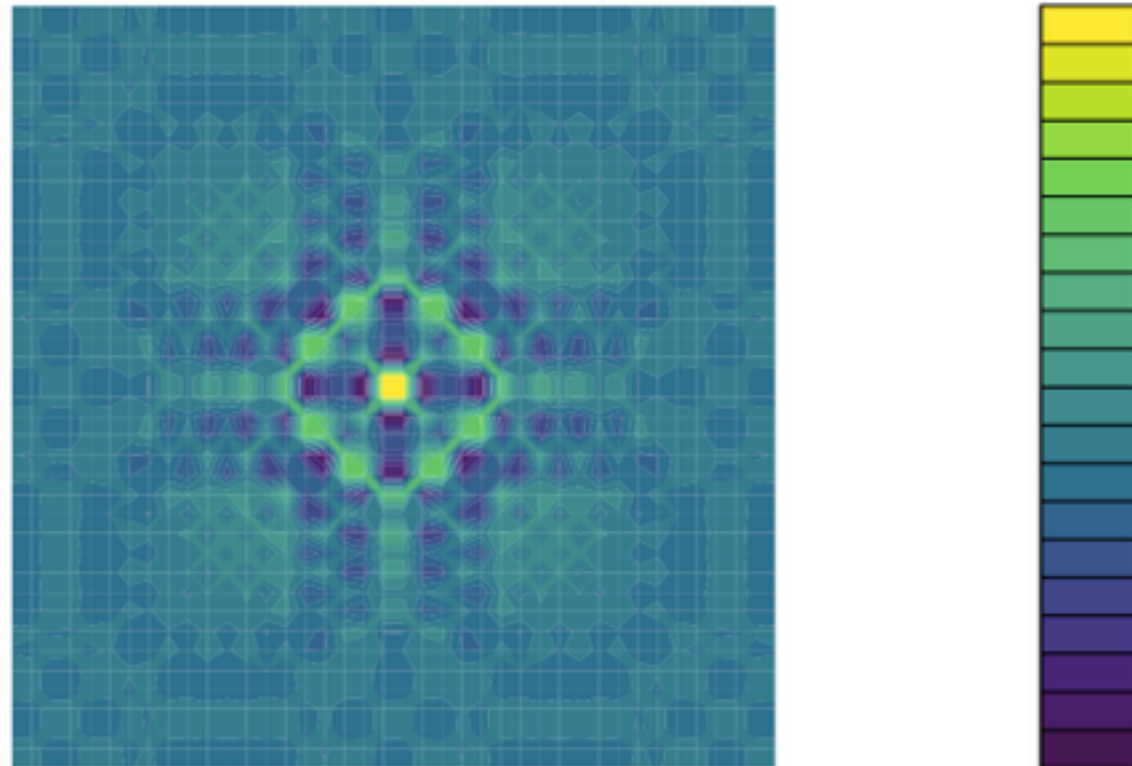
[A Rule-based Tool for Assisting Colormap Selection. Bergman, Rogowitz, and Treinish. Proc. IEEE Visualization (Vis), pp. 118–125, 1995.]



[Why Should Engineers Be Worried About Color? Treinish and Rogowitz 1998. <http://www.research.ibm.com/people/lloyd/color/color.HTM>]

Viridis / Magma

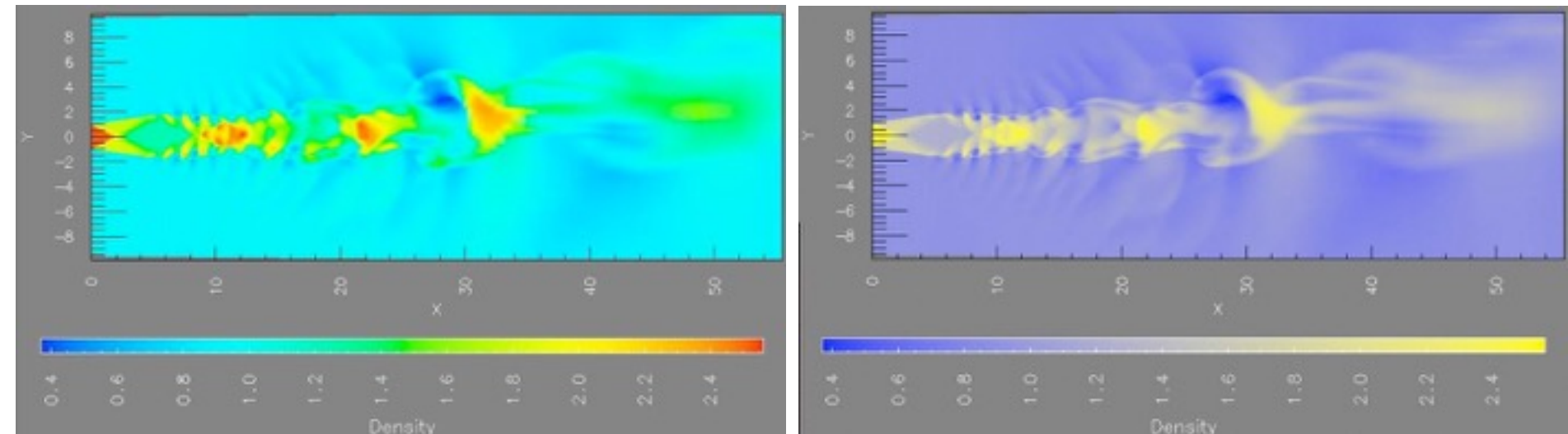
- monotonically increasing luminance, perceptually uniform
- colorful, colourblind-safe
 - R, python, D3



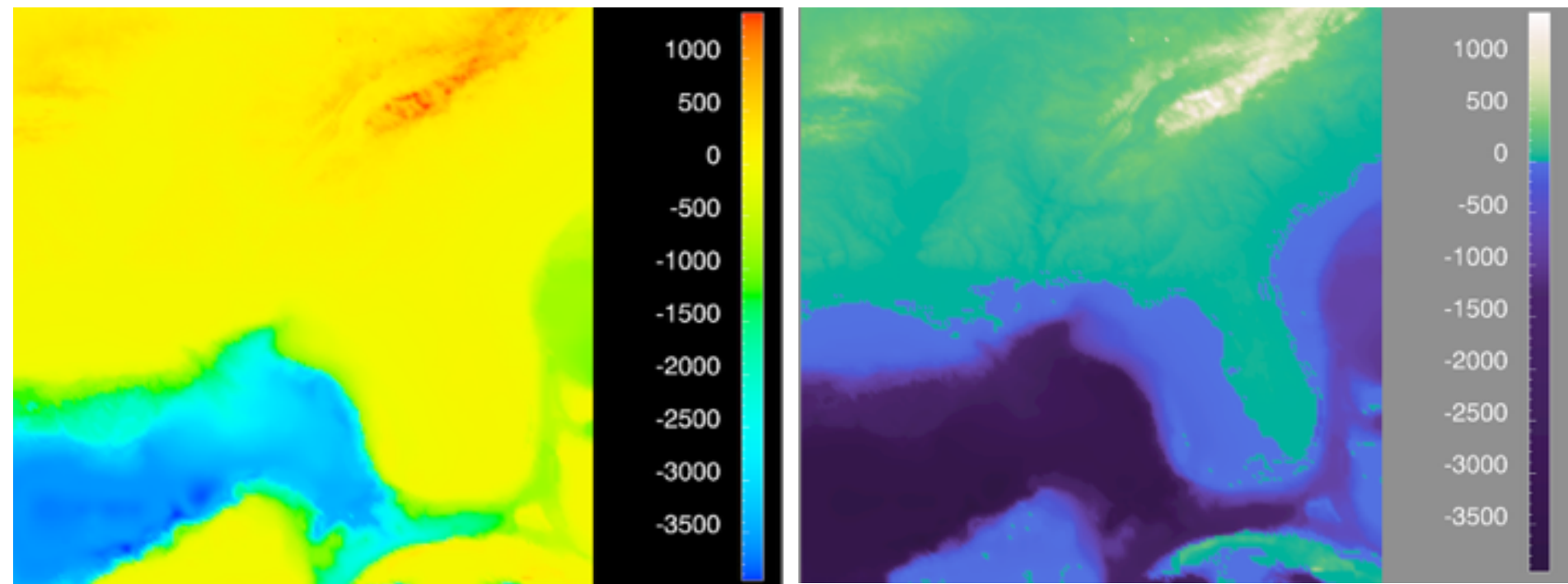
<https://cran.r-project.org/web/packages/viridis/vignettes/intro-to-viridis.html>

Ordered color: Rainbow is poor default

- problems
 - perceptually unordered
 - perceptually nonlinear
- benefits
 - fine-grained structure visible and nameable
- alternatives
 - large-scale structure: fewer hues
 - fine structure: multiple hues with monotonically increasing luminance [eg viridis]
 - categorical: segmented saturated rainbow is good!



[A Rule-based Tool for Assisting Colormap Selection. Bergman, Rogowitz, and Treinish. Proc. IEEE Visualization (Vis), pp. 118–125, 1995.]



[Why Should Engineers Be Worried About Color? Treinish and Rogowitz 1998. <http://www.research.ibm.com/people/lloyd/color/color.HTM>]



[Transfer Functions in Direct Volume Rendering: Design, Interface, Interaction. Kindlmann. SIGGRAPH 2002 Course Notes]

Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
 - Chap 10: Map Color and Other Channels*
- ColorBrewer, Brewer.
 - <http://www.colorbrewer2.org>
- *Color In Information Display*. Stone. IEEE Vis Course Notes, 2006.
 - <http://www.stonesc.com/Vis06>
- A Field Guide to Digital Color. Stone. AK Peters, 2003.
- *Rainbow Color Map (Still) Considered Harmful*. Borland and Taylor. IEEE Computer Graphics and Applications 27:2 (2007), 14–17.
- Visual Thinking for Design. Ware. Morgan Kaufmann, 2008.
- Information Visualization: Perception for Design, 3rd edition. Ware. Morgan Kaufmann / Academic Press, 2004.
- <http://www.r-bloggers.com/using-the-new-iridis-colormap-in-r-thanks-to-simon-garnier/>

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - **Manipulate: Change, Select, Navigate**
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

How?

Encode

→ Arrange

→ Express



→ Order



→ Use



→ Separate



→ Align



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape



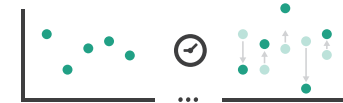
→ Motion

Direction, Rate, Frequency, ...

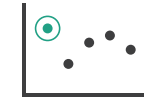


Manipulate

→ Change



→ Select



→ Navigate

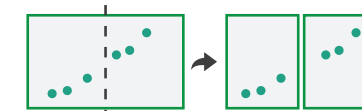


Facet

→ Juxtapose



→ Partition



→ Superimpose



Reduce

→ Filter



→ Aggregate



→ Embed



What?

Why?

How?

How to handle complexity: 1 previous strategy + 3 more

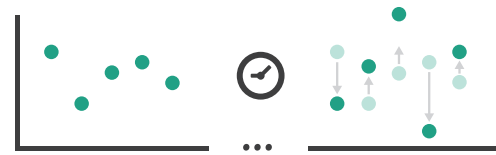
→ *Derive*



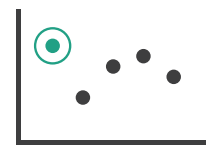
- derive new data to show within view
- change view over time
- facet across multiple views
- reduce items/attributes within single view

Manipulate

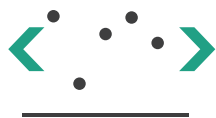
→ Change



→ Select

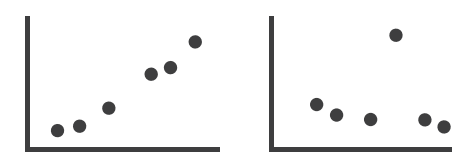


→ Navigate

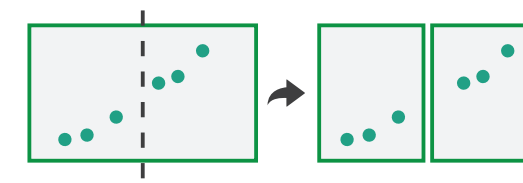


Facet

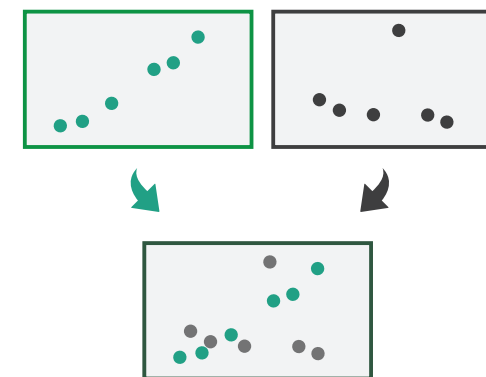
→ Juxtapose



→ Partition



→ Superimpose



Reduce

→ Filter



→ Aggregate

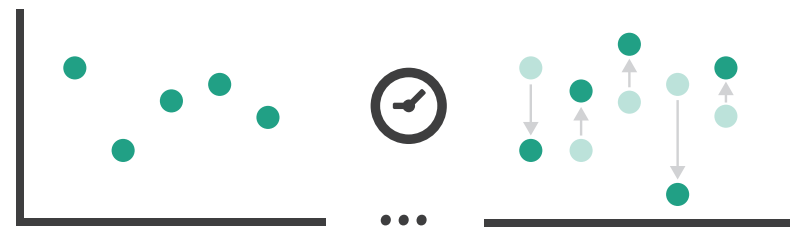


→ Embed

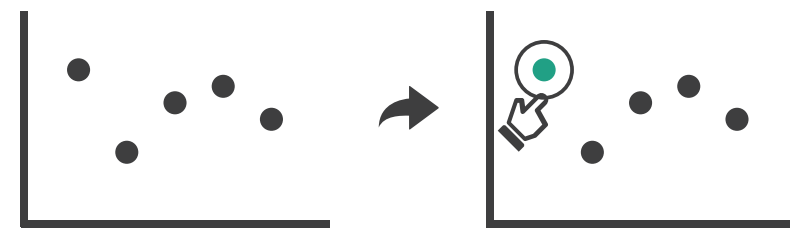


Manipulate

→ Change over Time



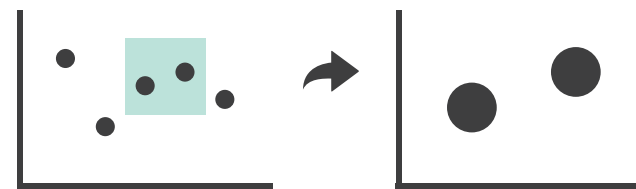
→ Select



→ Navigate

→ Item Reduction

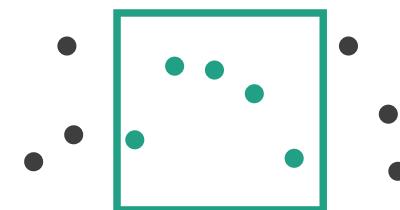
→ Zoom
Geometric or Semantic



→ Pan/Translate

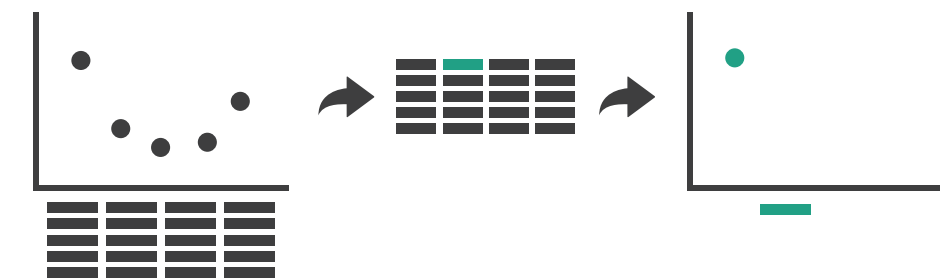


→ Constrained

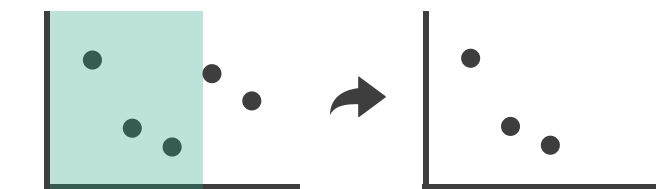


→ Attribute Reduction

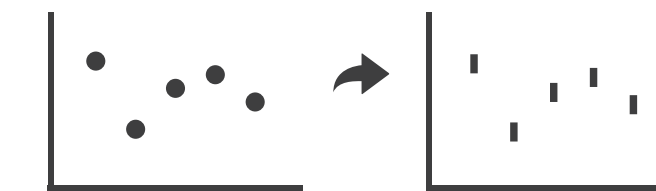
→ Slice



→ Cut



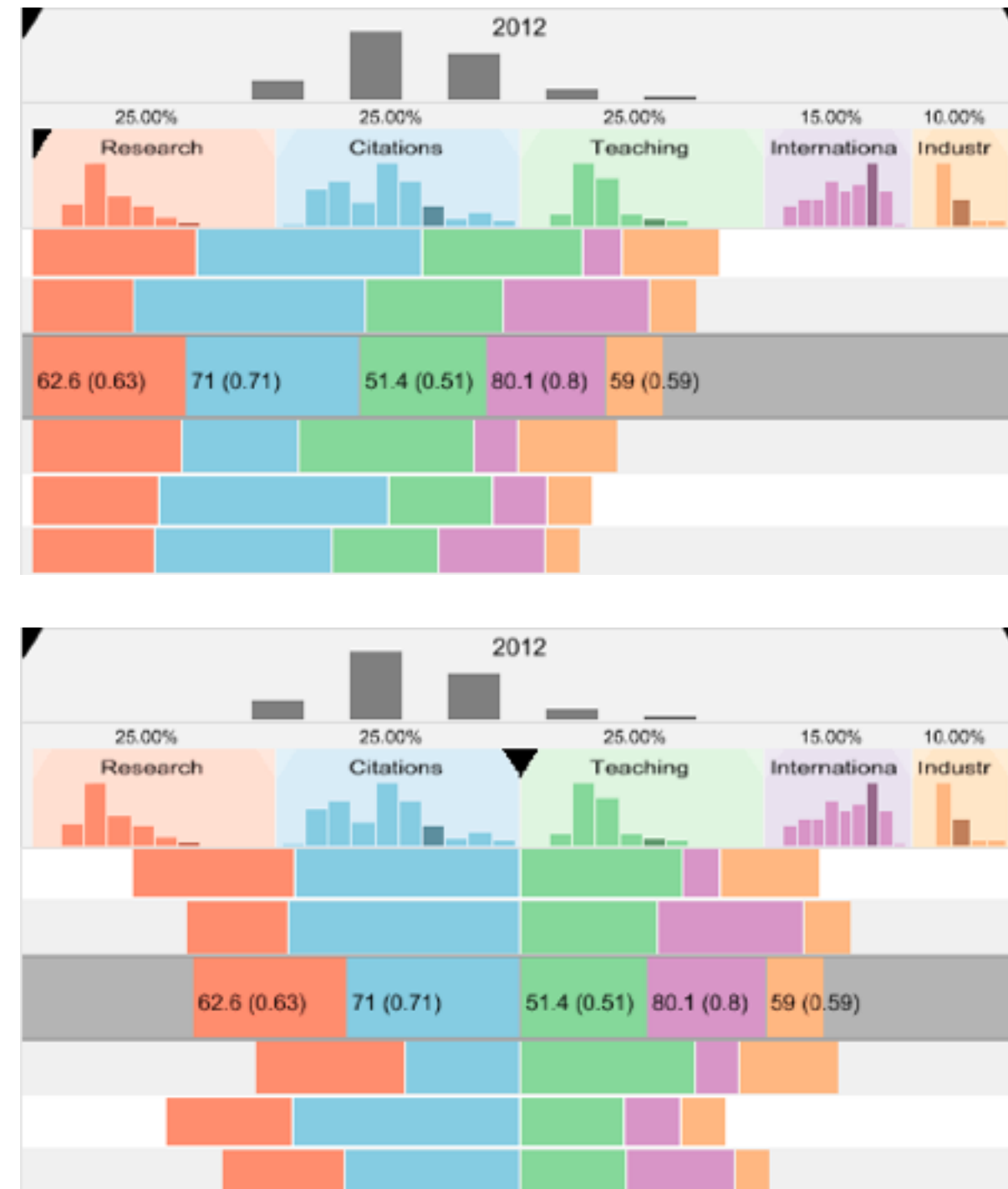
→ Project



Idiom: **Realign**

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

System: **LineUp**



[LineUp: Visual Analysis of Multi-Attribute Rankings. Gratzl, Lex, Gehlenborg, Pfister, and Streit. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2013) 19:12 (2013), 2277–2286.]

Navigate: Changing item visibility

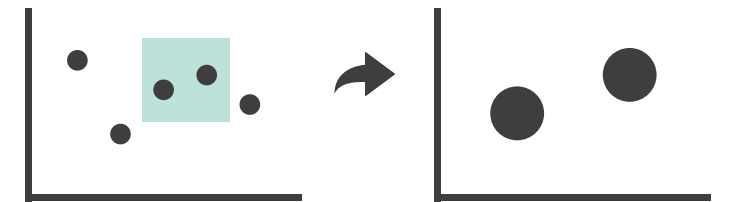
- change viewpoint
 - changes which items are visible within view
 - camera metaphor
 - zoom
 - geometric zoom: familiar semantics
 - semantic zoom: adapt object representation based on available pixels
 - » dramatic change, or more subtle one
 - pan/translate
 - rotate
 - especially in 3D
 - constrained navigation
 - often with animated transitions
 - often based on selection set

➔ Navigate

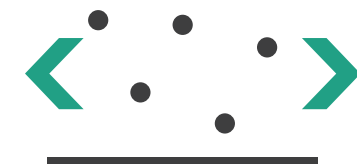
➔ Item Reduction

➔ Zoom

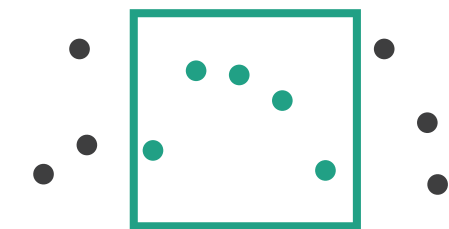
Geometric or *Semantic*



➔ Pan/Translate



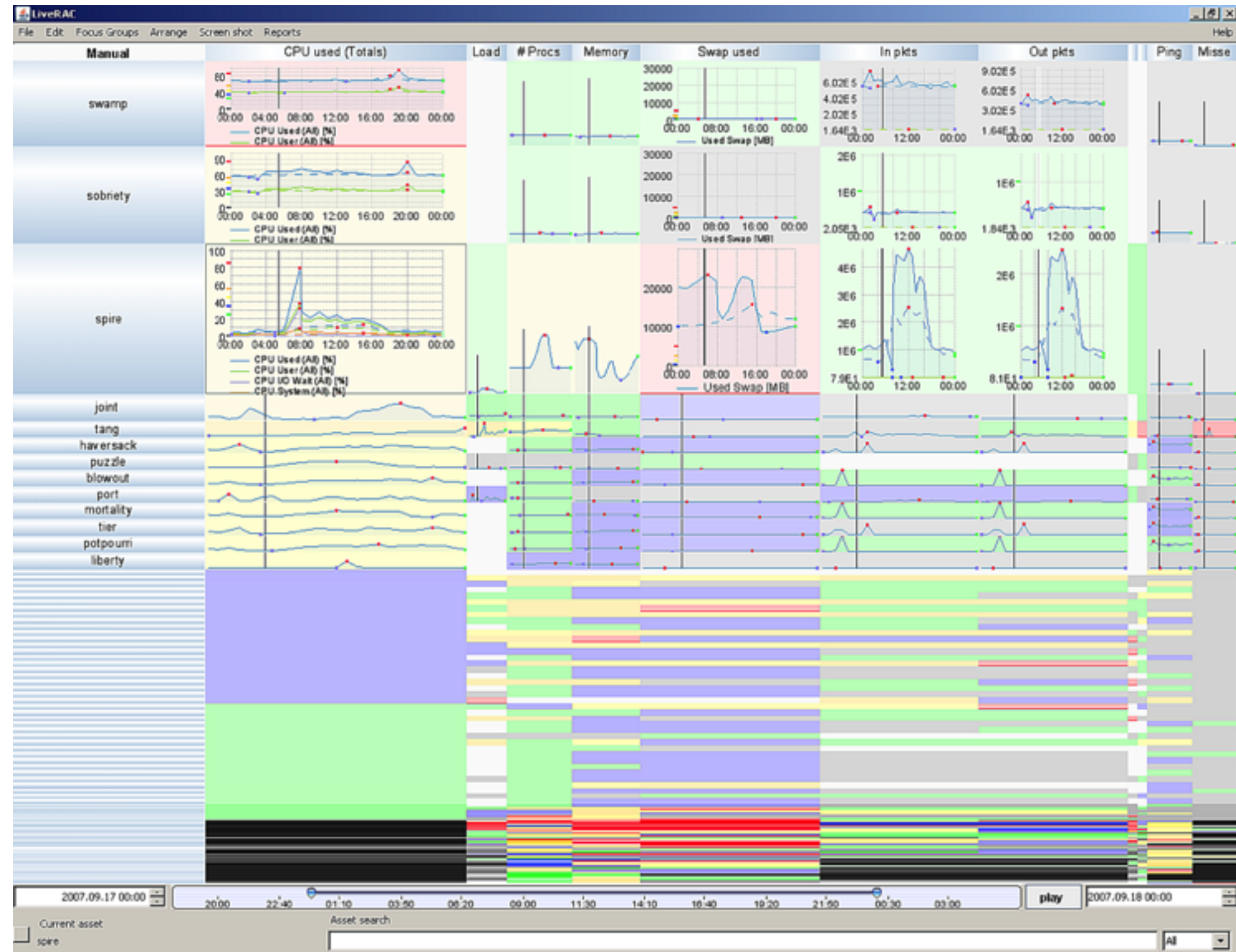
➔ Constrained



Idiom: Semantic zooming

System: LiveRAC

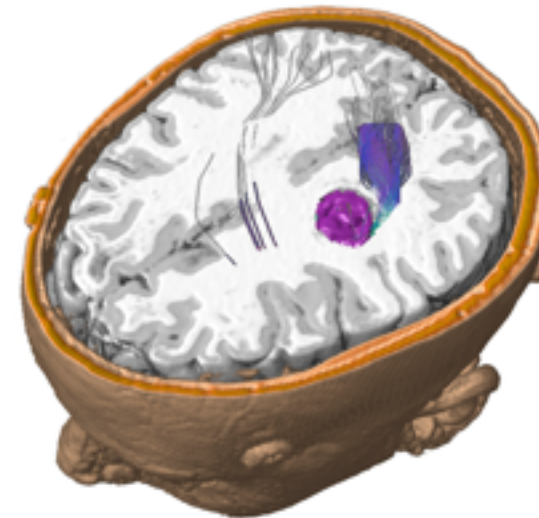
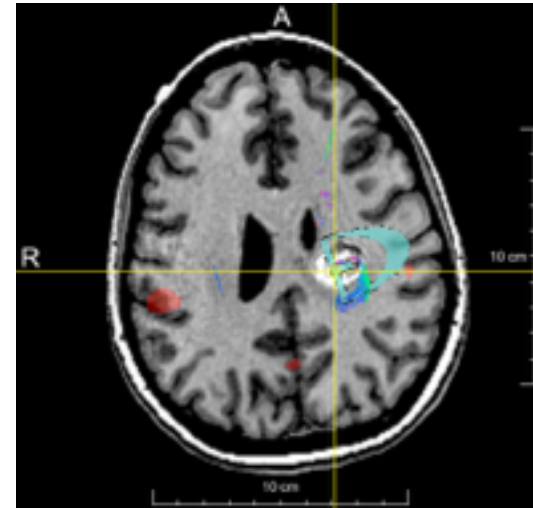
- visual encoding change
 - colored box
 - sparkline
 - simple line chart
 - full chart: axes and tickmarks



[LiveRAC - Interactive Visual Exploration of System Management Time-Series Data. McLachlan, Munzner, Koutsofios, and North. Proc. ACM Conf. Human Factors in Computing Systems (CHI), pp. 1483–1492, 2008.]

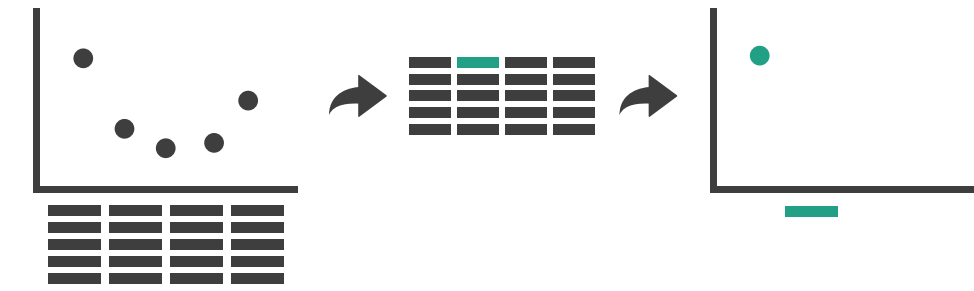
Navigate: Reducing attributes

- continuation of camera metaphor
 - slice
 - show only items matching specific value for given attribute: slicing plane
 - axis aligned, or arbitrary alignment
 - cut
 - show only items on far side of plane from camera
 - project
 - change mathematics of image creation
 - orthographic
 - perspective
 - many others: Mercator, cabinet, ...

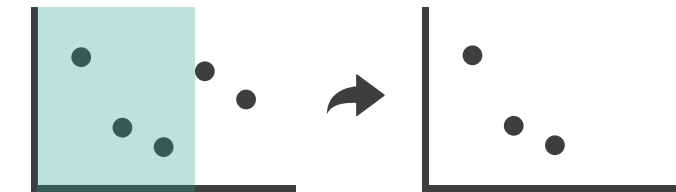


→ Attribute Reduction

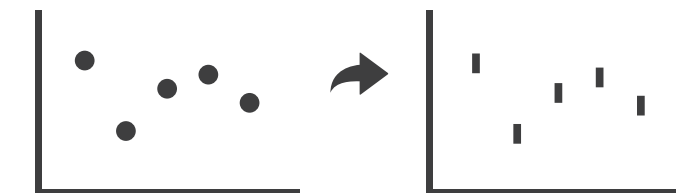
→ *Slice*



→ *Cut*



→ *Project*



Further reading

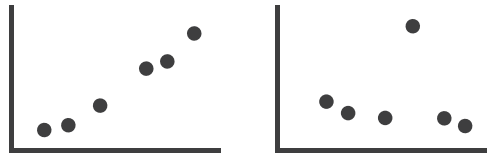
- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
 - Chap 11: Manipulate View*
- *Animated Transitions in Statistical Data Graphics*. Heer and Robertson. IEEE Trans. on Visualization and Computer Graphics (Proc. InfoVis07) 13:6 (2007), 1240–1247.
- *Selection: 524,288 Ways to Say “This is Interesting”*. Wills. Proc. IEEE Symp. Information Visualization (InfoVis), pp. 54–61, 1996.
- *Smooth and efficient zooming and panning*. van Wijk and Nuij. Proc. IEEE Symp. Information Visualization (InfoVis), pp. 15–22, 2003.
- *Starting Simple - adding value to static visualisation through simple interaction*. Dix and Ellis. Proc. Advanced Visual Interfaces (AVI), pp. 124–134, 1998.

Outline

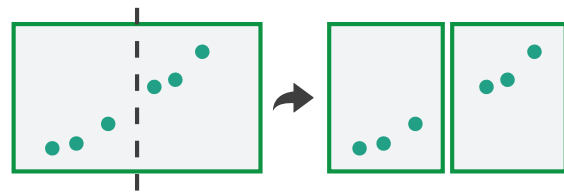
- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - **Facet: Juxtapose, Partition, Superimpose**
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

Facet

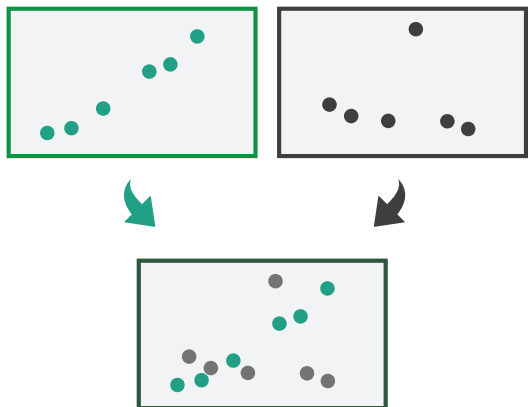
→ Juxtapose



→ Partition



→ Superimpose



Juxtapose and coordinate views

→ Share Encoding: Same/Different

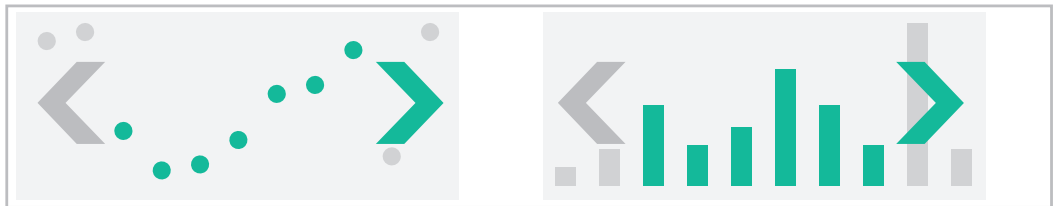
→ *Linked Highlighting*



→ Share Data: All/Subset/None



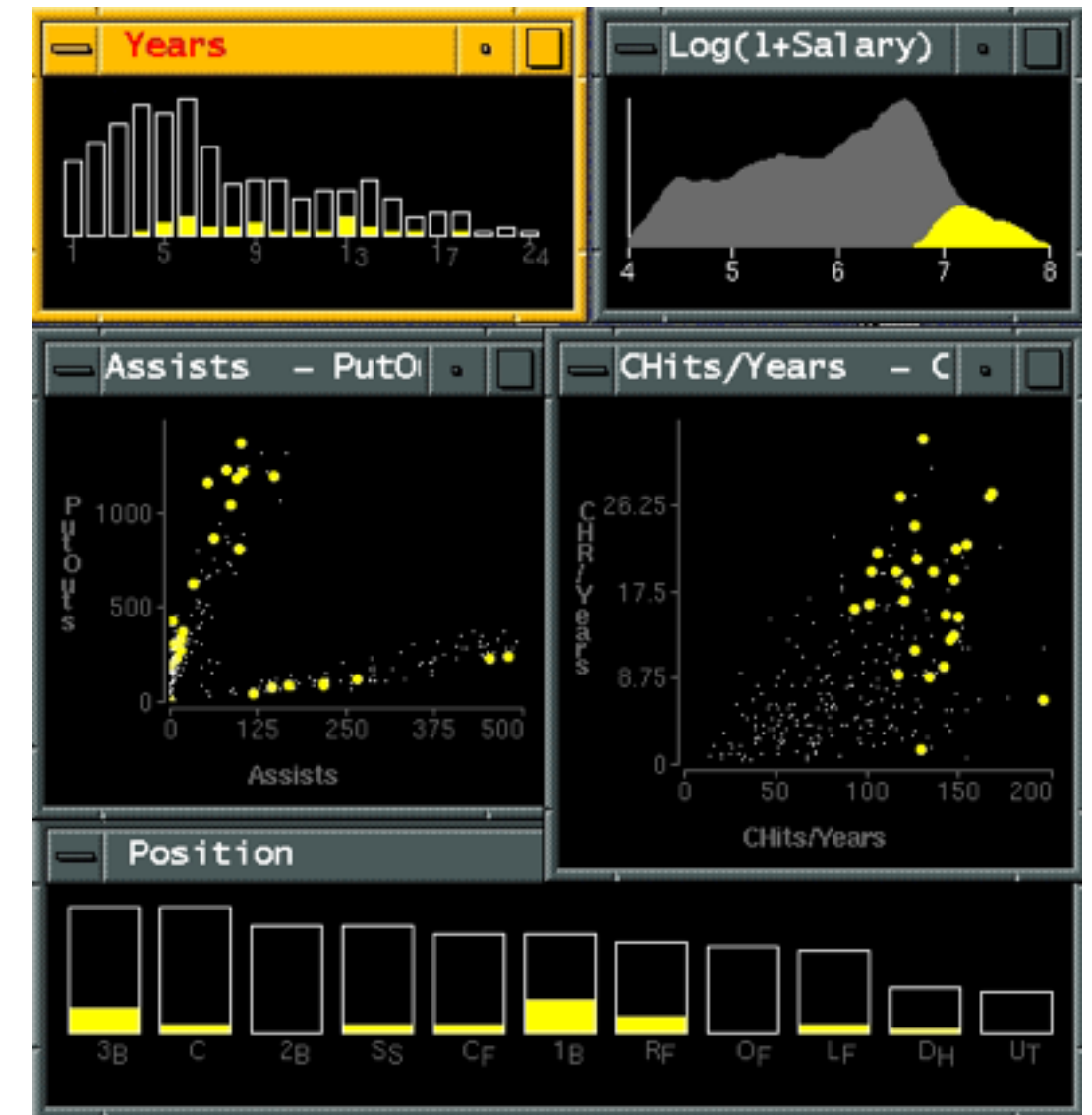
→ Share Navigation



Idiom: **Linked highlighting**

System: **EDV**

- see how regions contiguous in one view are distributed within another
 - powerful and pervasive interaction idiom
- encoding: different
 - *multiform*
- data: all shared



[Visual Exploration of Large Structured Datasets. Wills. Proc. New Techniques and Trends in Statistics (NTTS), pp. 237–246. IOS Press, 1995.]

Idiom: **bird's-eye maps**

System: **Google Maps**

- encoding: same
- data: subset shared
- navigation: shared
 - bidirectional linking

- differences
 - viewpoint
 - (size)

- **overview-detail**

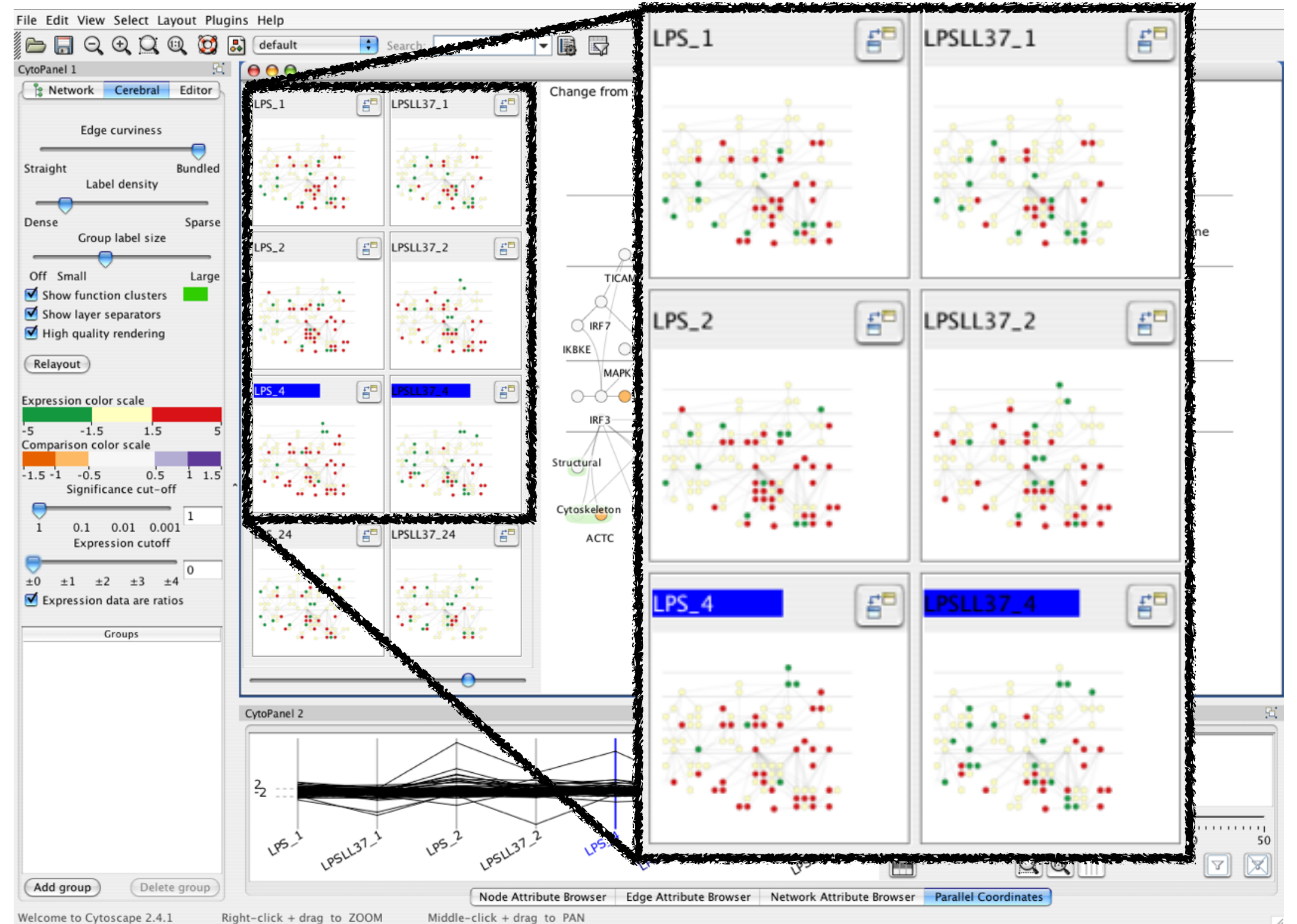


[A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. Cockburn, Karlson, and Bederson. *ACM Computing Surveys* 41:1 (2008), 1–31.]

Idiom: Small multiples

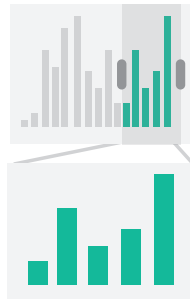
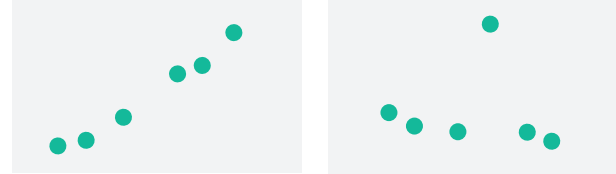


System: Cerebral

- encoding: same
- data: none shared
 - different attributes for node colors
 - (same network layout)
- navigation: shared



[Cerebral: Visualizing Multiple Experimental Conditions on a Graph with Biological Context. Barsky, Munzner, Gardy, and Kincaid. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2008)* 14:6 (2008), 1253–1260.]

Coordinate views: Design choice interaction

		Data		
		All	Subset	None
Encoding	Same	Redundant	 Overview/ Detail	 Small Multiples
	Different	 Multiform	 Multiform, Overview/ Detail	No Linkage

- why juxtapose views?

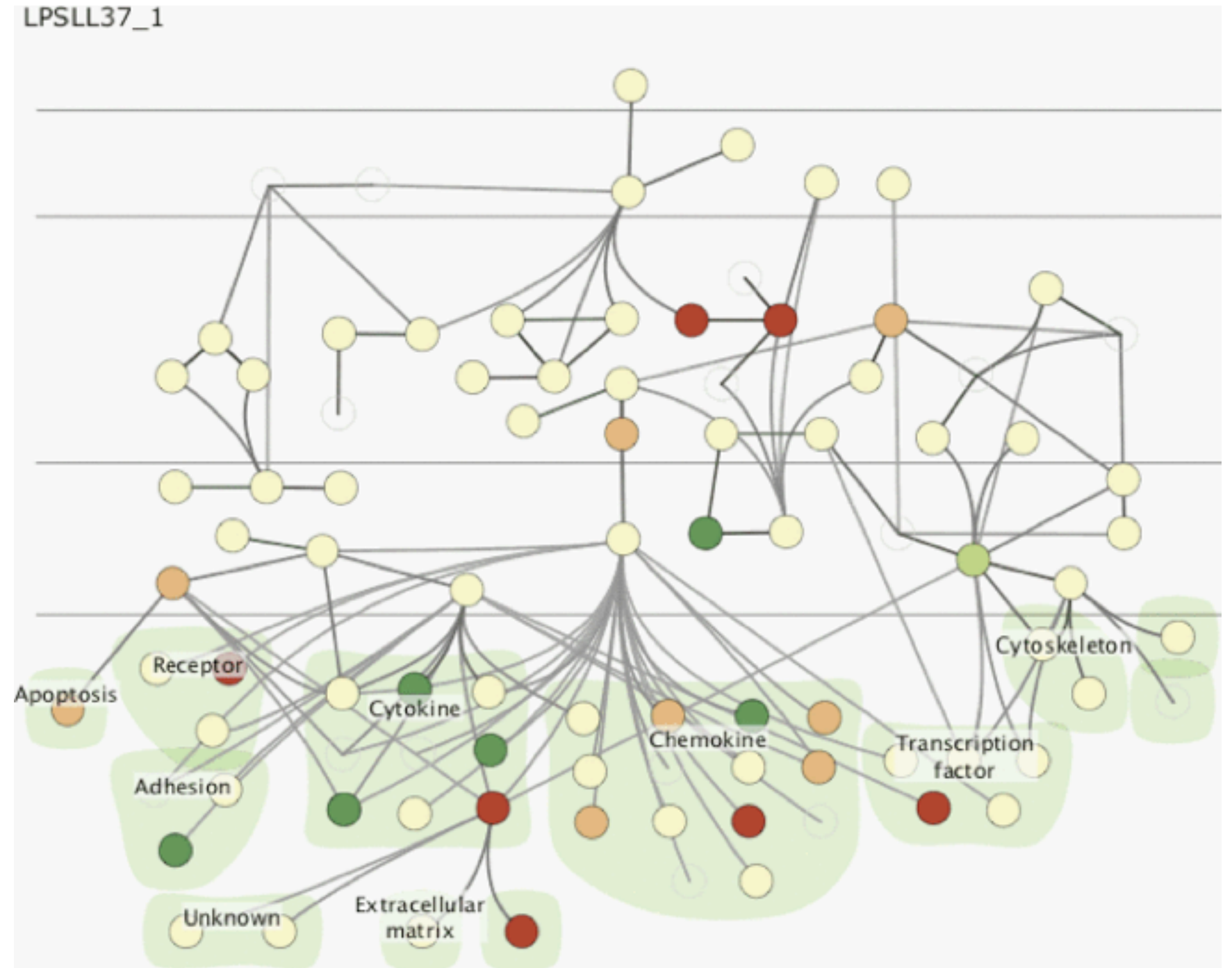
- benefits: eyes vs memory

- lower cognitive load to move eyes between 2 views than remembering previous state with single changing view

- costs: display area, 2 views side by side each have only half the area of one view

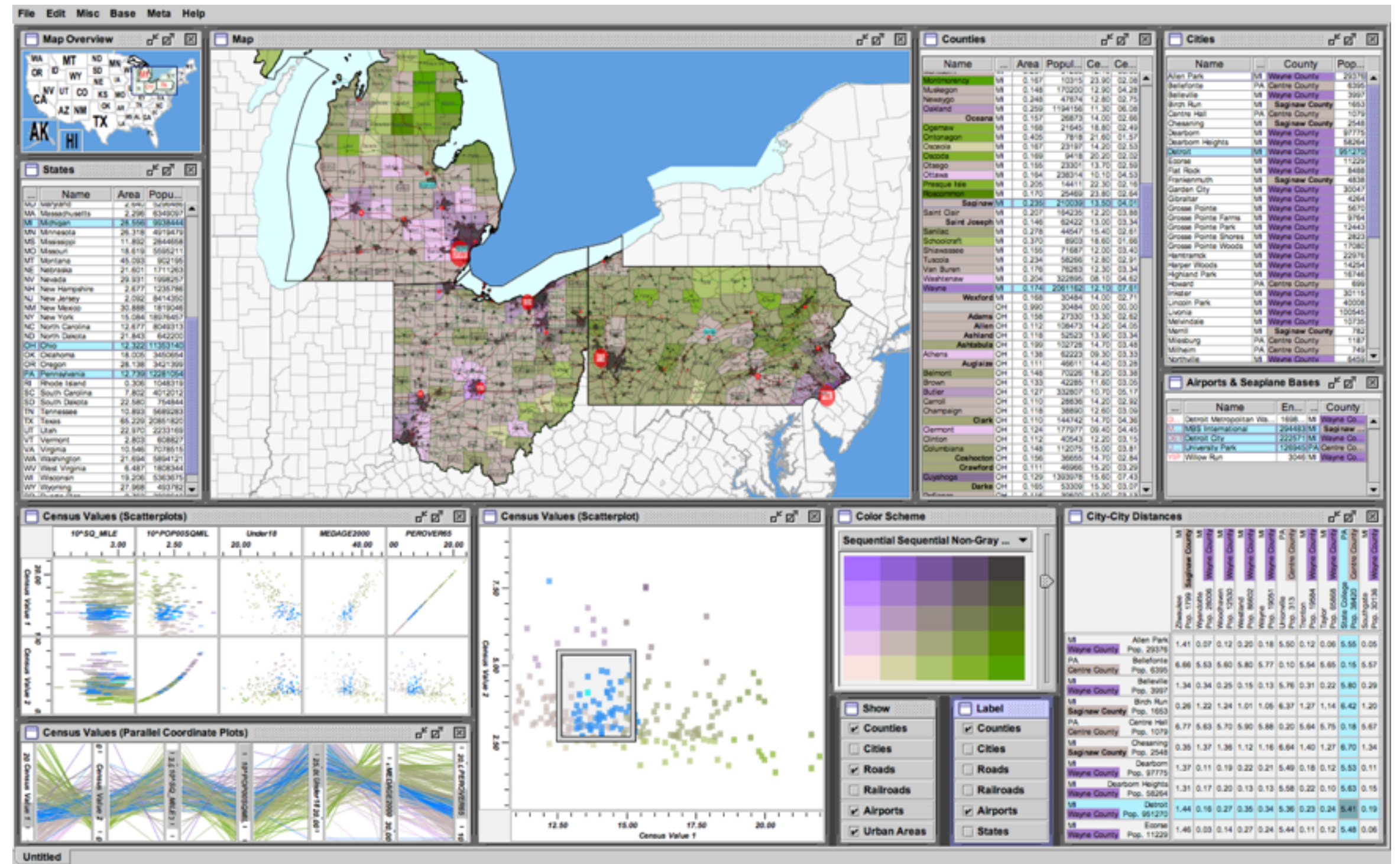
Why not animation?

- disparate frames and regions: comparison difficult
 - vs contiguous frames
 - vs small region
 - vs coherent motion of group
- safe special case
 - animated transitions



System: **Improvise**

- investigate power of multiple views
 - pushing limits on view count, interaction complexity
 - how many is ok?
 - open research question
 - reorderable lists
 - easy lookup
 - useful when linked to other encodings

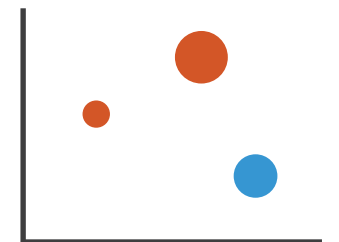
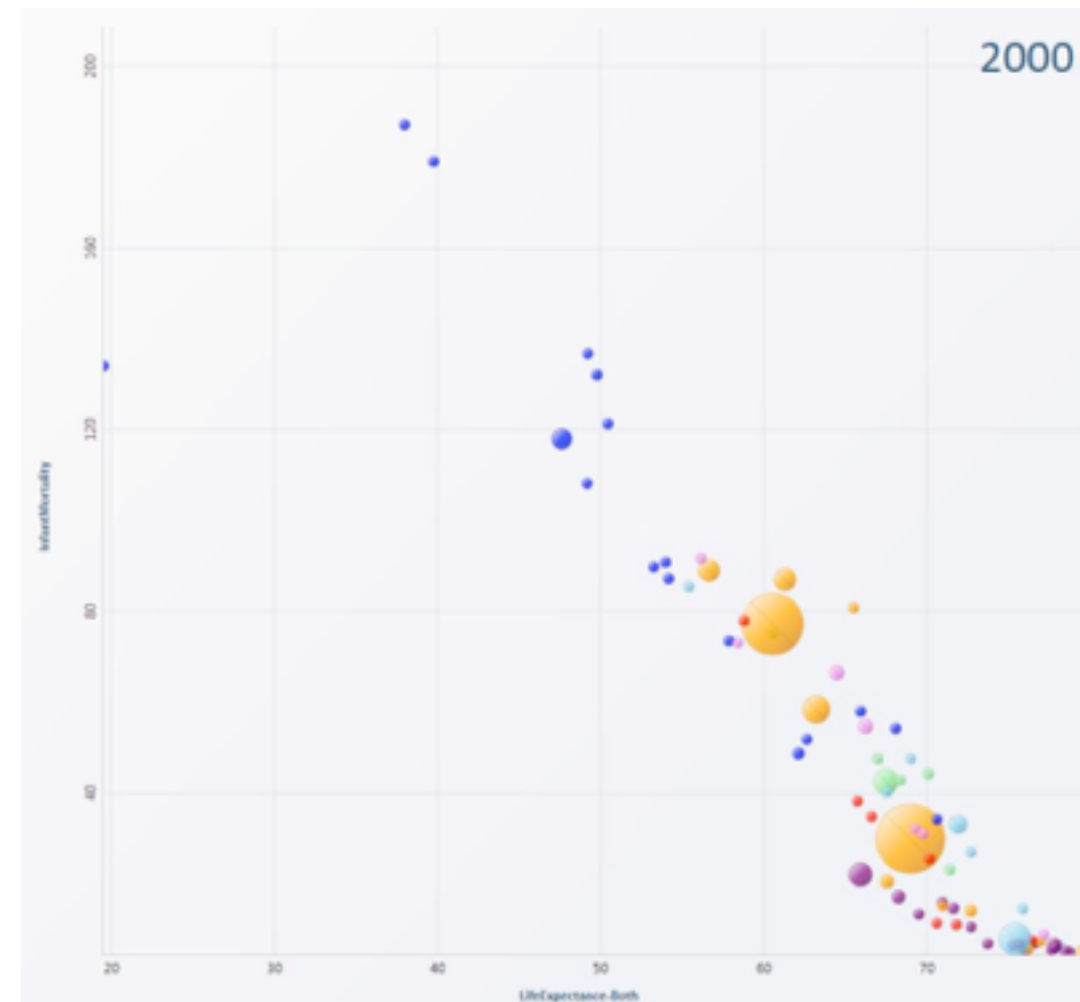
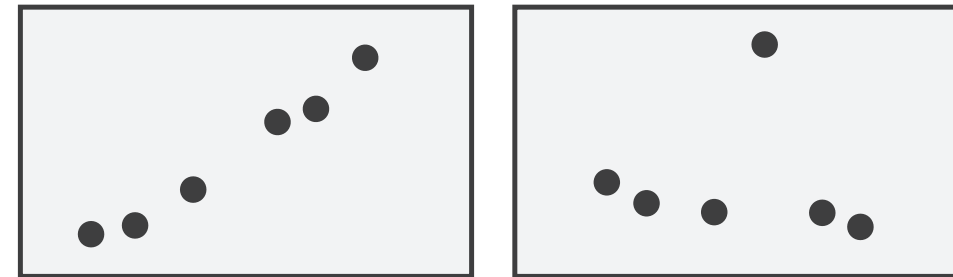


[Building Highly-Coordinated Visualizations In Improvise. Weaver. Proc. IEEE Symp. Information Visualization (InfoVis), pp. 159–166, 2004.]

Partition into views

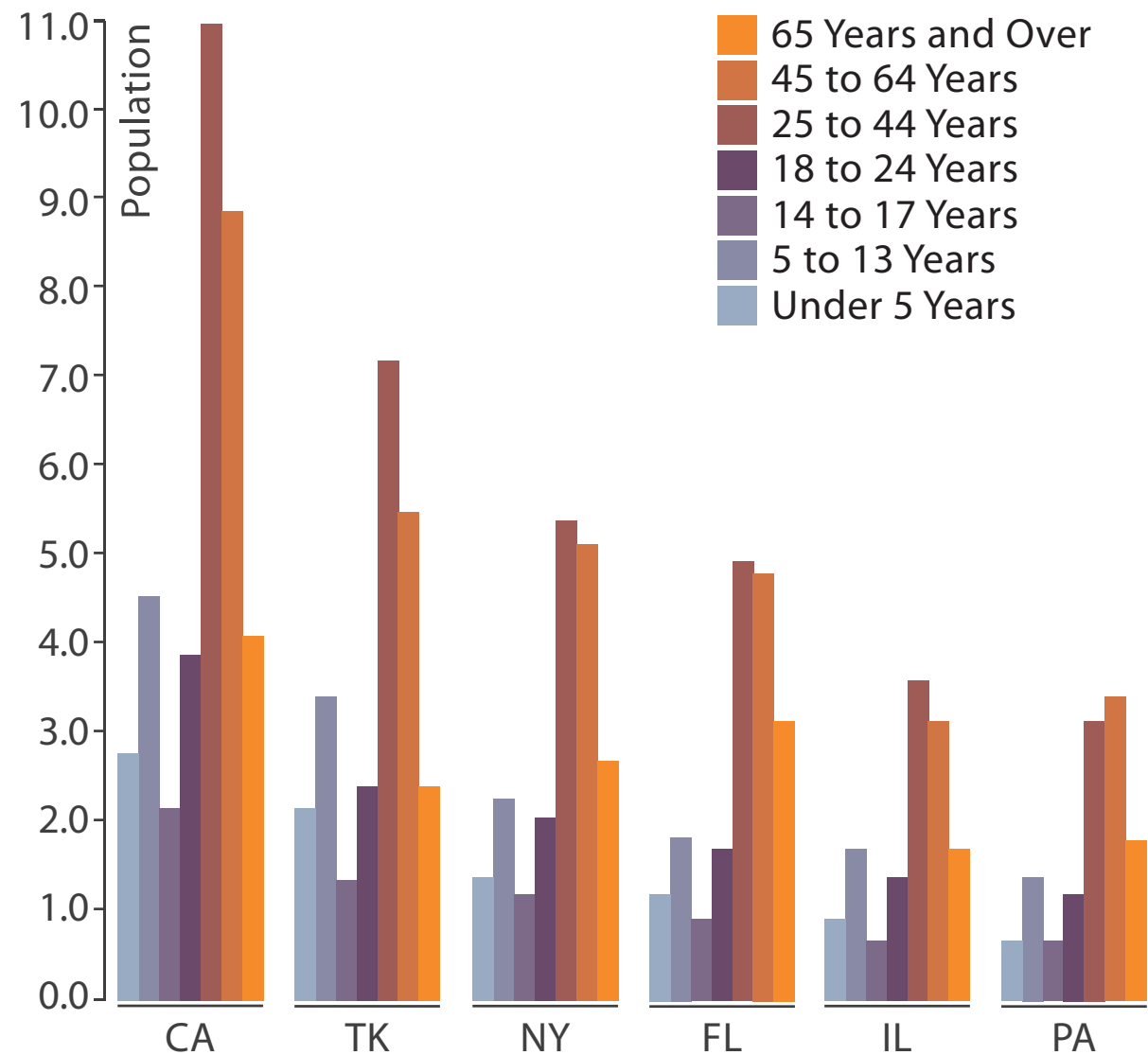
- how to divide data between views
 - split into regions by attributes
 - encodes association between items using spatial proximity
 - order of splits has major implications for what patterns are visible
- no strict dividing line
 - *view: big/detailed*
 - contiguous region in which visually encoded data is shown on the display
 - *glyph: small/iconic*
 - object with internal structure that arises from multiple marks

➔ Partition into Side-by-Side Views

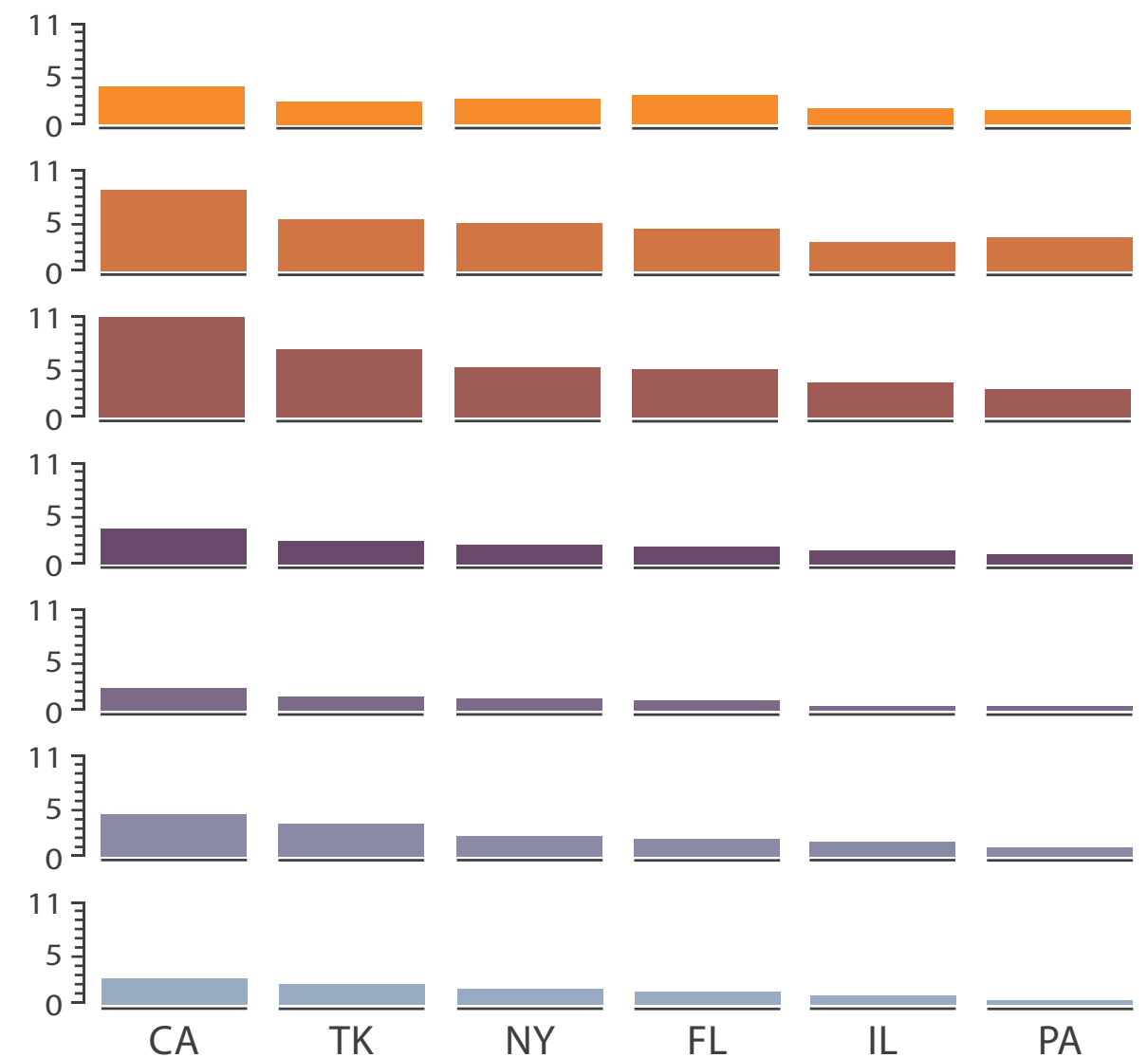


Partitioning: List alignment

- 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



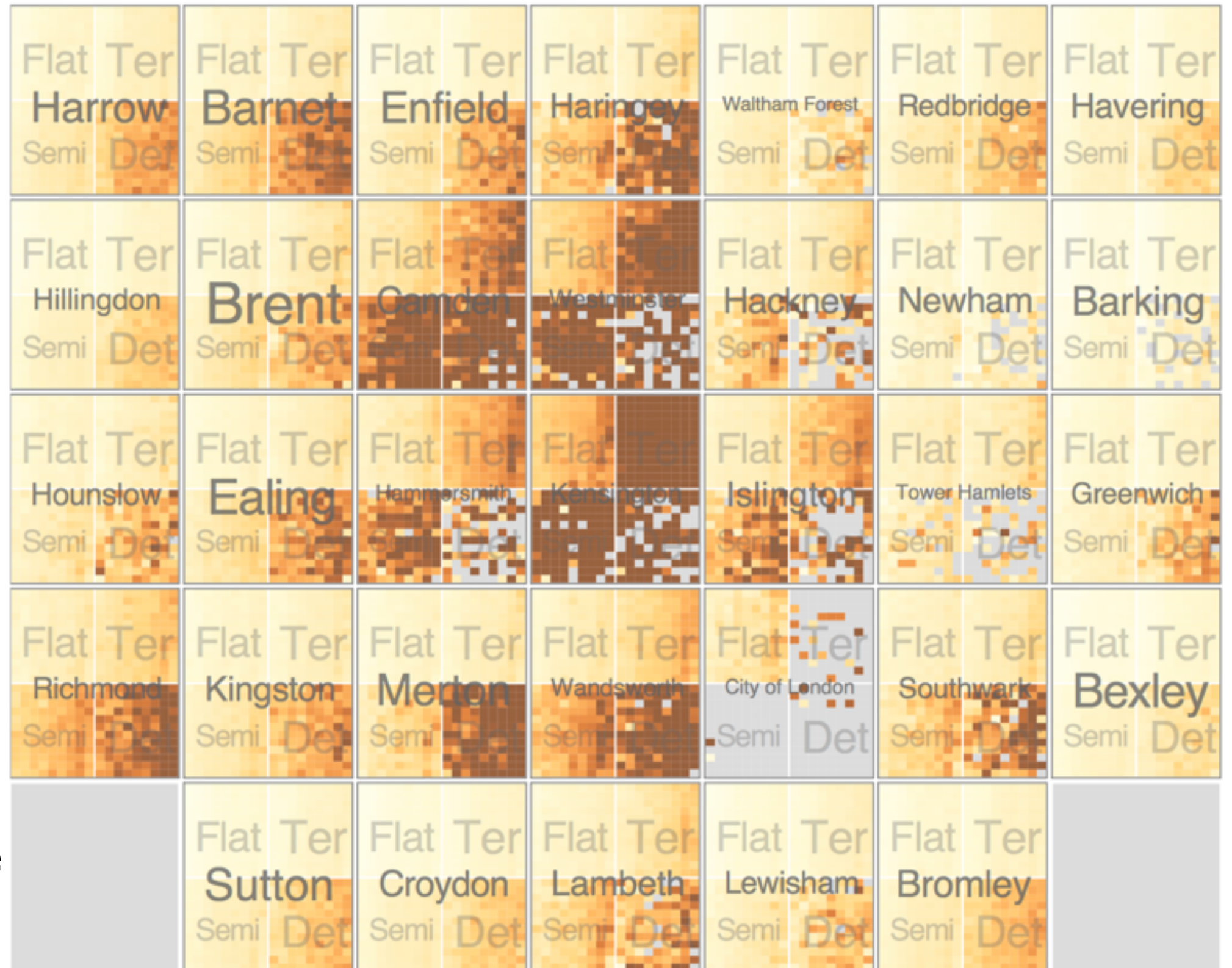
- small-multiple bar charts
 - split by age into regions
 - one chart per region
 - compare: easy within age, harder across states



Partitioning: Recursive subdivision

System: **HIVE**

- split by neighborhood
- then by type
- 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



Partitioning: Recursive subdivision

System: **HIVE**

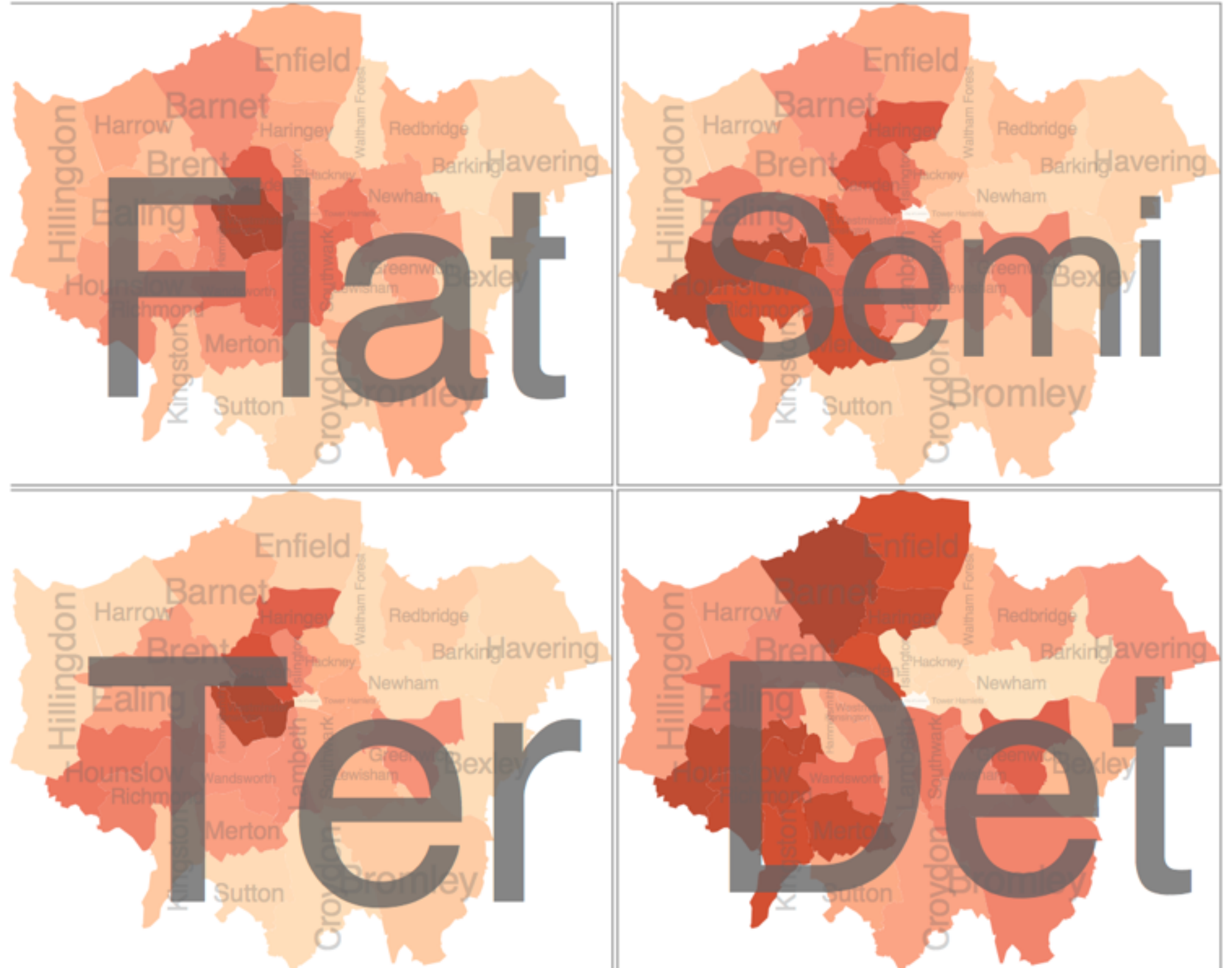
- switch order of splits
 - type then neighborhood
- switch color
 - by price variation
- type patterns
 - within specific type, which neighborhoods inconsistent



Partitioning: Recursive subdivision

System: **HIVE**

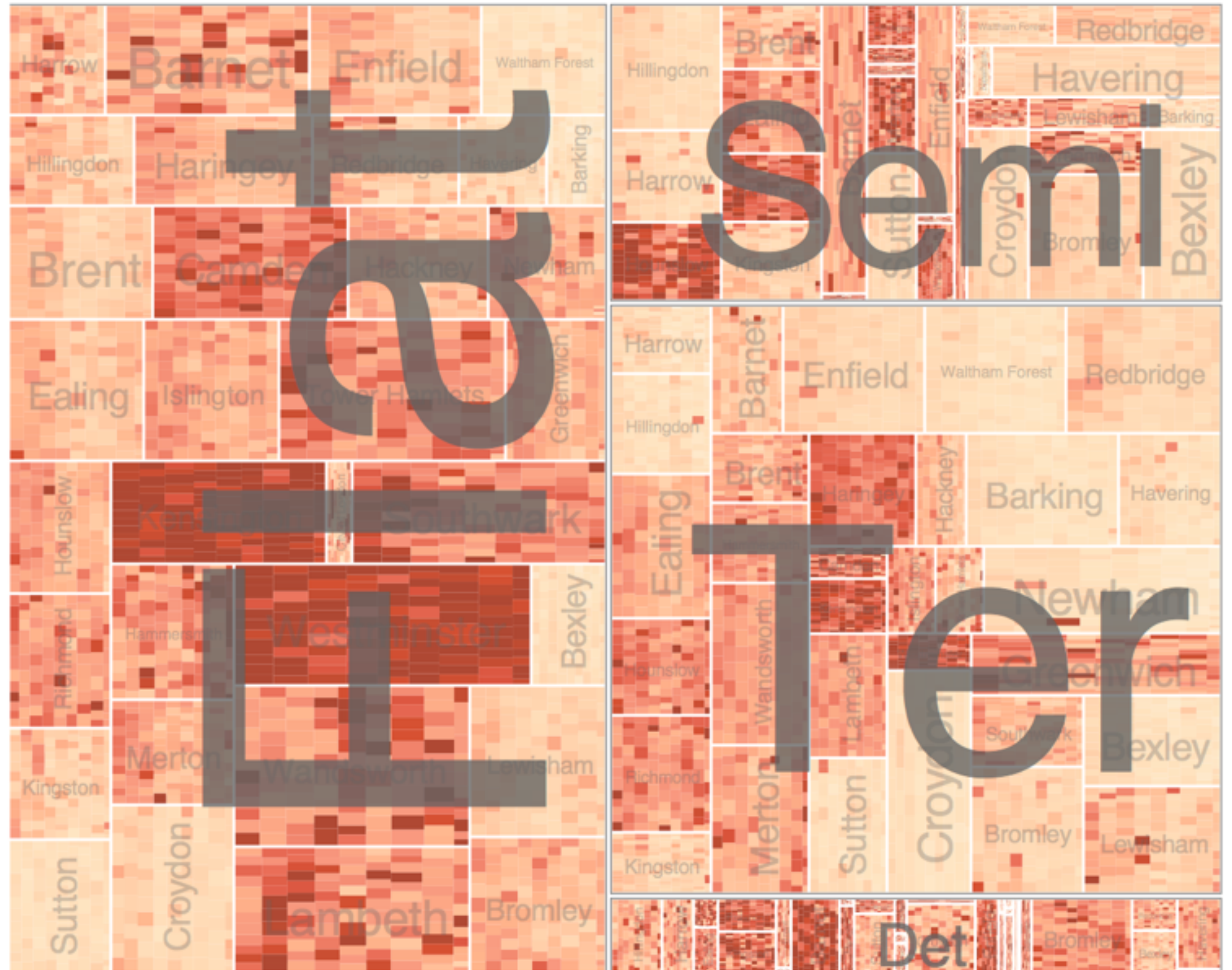
- different encoding for second-level regions
– choropleth maps



Partitioning: Recursive subdivision

System: **HIVE**

- size regions by sale counts
 - not uniformly
- result: treemap



Superimpose layers

- *layer*: set of objects spread out over region

- each set is visually distinguishable group

- extent: whole view

- design choices

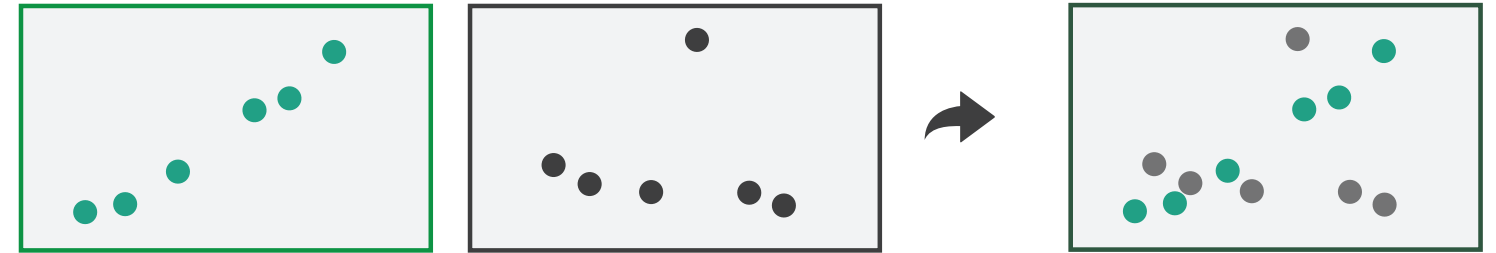
- how many layers, how to distinguish?

- 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”
 - check luminance contrast with greyscale view

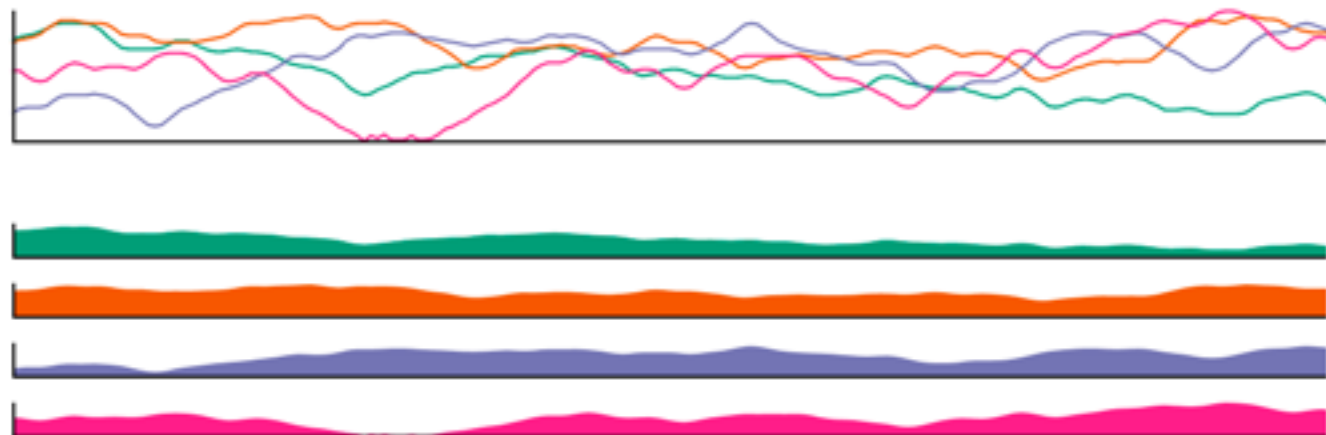


[Get it right in black and white. Stone. 2010.

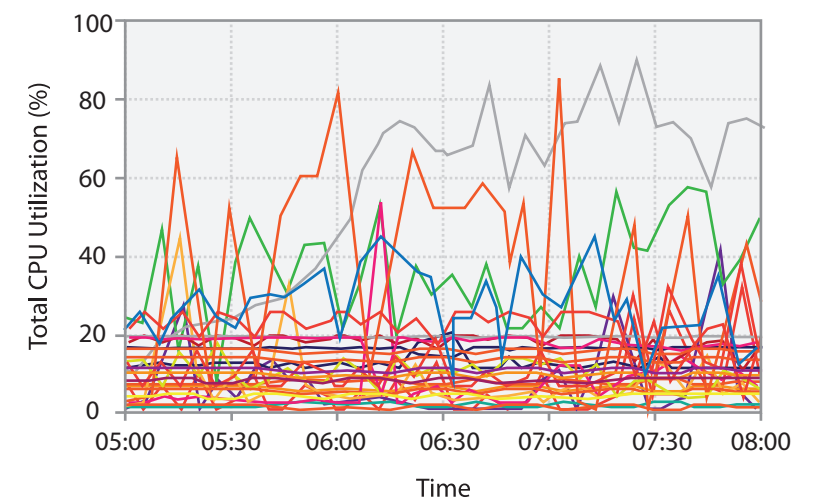
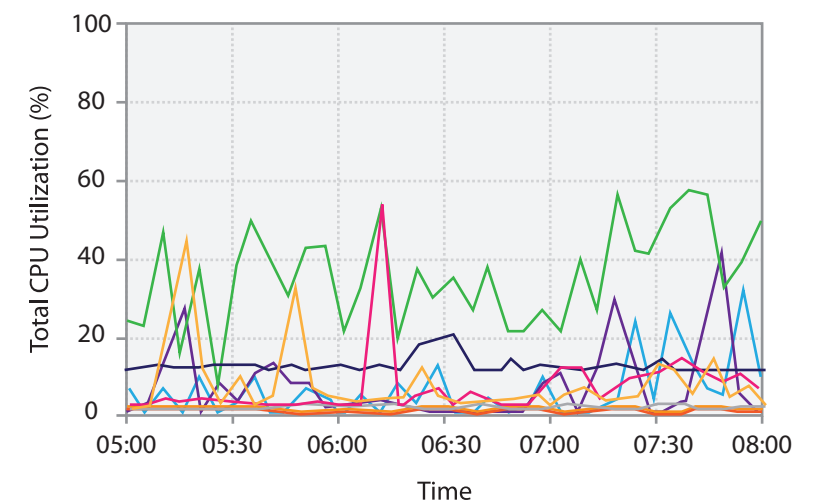
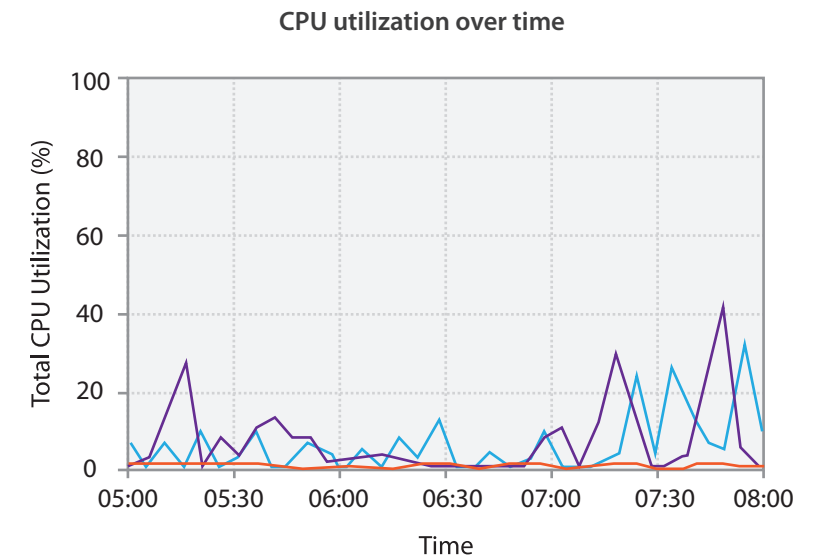
<http://www.stonesc.com/wordpress/2010/03/get-it-right-in-black-and-white>]

Superimposing limits

- few layers, but many lines
 - up to a few dozen
 - but not hundreds
- superimpose vs juxtapose: empirical study
 - superimposed for local, multiple for global
 - tasks
 - local: maximum, global: slope, discrimination
 - same screen space for all multiples vs single superimposed



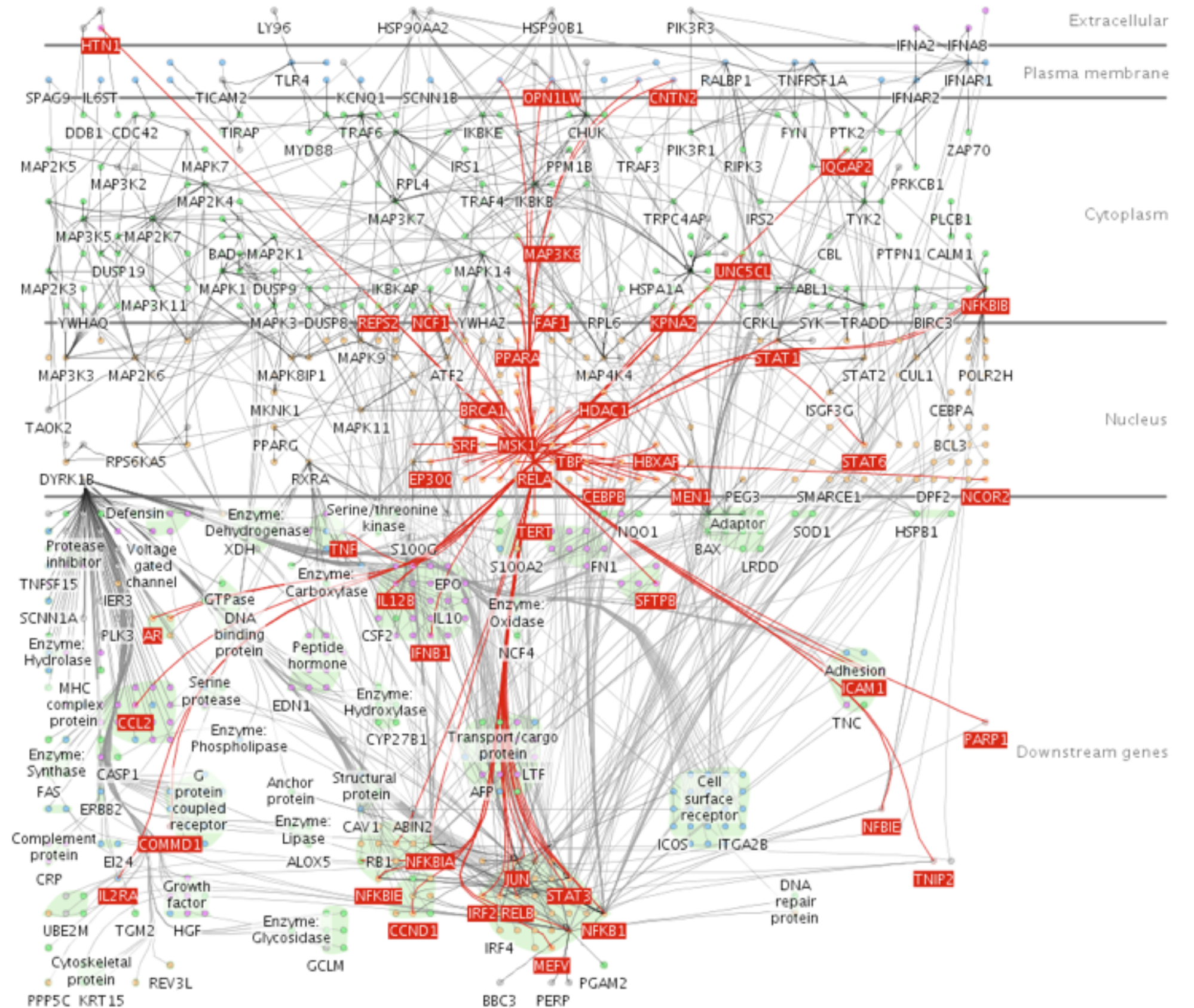
[Graphical Perception of Multiple Time Series. Javed, McDonnell, and Elmqvist. IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE InfoVis 2010) 16:6 (2010), 927–934.]



Dynamic visual layering

- interactive, from selection
 - lightweight: click
 - very lightweight: hover
- ex: 1-hop neighbors

System: Cerebral



[Cerebral: a Cytoscape plugin for layout of and interaction with biological networks using subcellular localization annotation. Barsky, Gardy, Hancock, and Munzner. *Bioinformatics* 23:8 (2007), 1040–1042.]

Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
–*Chap 12: Facet Into Multiple Views*
- *A Review of Overview+Detail, Zooming, and Focus+Context Interfaces*. Cockburn, Karlson, and Bederson. ACM Computing Surveys 41:1 (2008), 1–31.
- *A Guide to Visual Multi-Level Interface Design From Synthesis of Empirical Study Evidence*. Lam and Munzner. Synthesis Lectures on Visualization Series, Morgan Claypool, 2010.
- *Zooming versus multiple window interfaces: Cognitive costs of visual comparisons*. Plumlee and Ware. ACM Trans. on Computer-Human Interaction (ToCHI) 13:2 (2006), 179–209.
- *Exploring the Design Space of Composite Visualization*. Javed and Elmqvist. Proc. Pacific Visualization Symp. (PacificVis), pp. 1–9, 2012.
- *Visual Comparison for Information Visualization*. Gleicher, Albers, Walker, Jusufi, Hansen, and Roberts. Information Visualization 10:4 (2011), 289–309.
- *Guidelines for Using Multiple Views in Information Visualizations*. Baldonado, Woodruff, and Kuchinsky. In Proc. ACM Advanced Visual Interfaces (AVI), pp. 110–119, 2000.
- *Cross-Filtered Views for Multidimensional Visual Analysis*. Weaver. IEEE Trans. Visualization and Computer Graphics 16:2 (Proc. InfoVis 2010), 192–204, 2010.
- *Linked Data Views*. Wills. In Handbook of Data Visualization, Computational Statistics, edited by Unwin, Chen, and Härdle, pp. 216–241. Springer-Verlag, 2008.
- *Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications*. Borgo, Kehrer, Chung, Maguire, Laramée, Hauser, Ward, and Chen. In Eurographics State of the Art Reports, pp. 39–63, 2013.

Outline

- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - Embed: Focus+Context

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

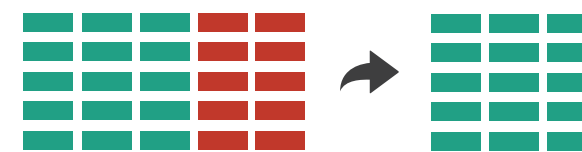
Reducing Items and Attributes

① Filter

→ Items



→ Attributes

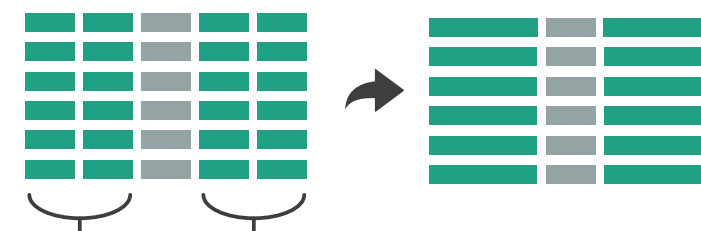


② Aggregate

→ Items



→ Attributes

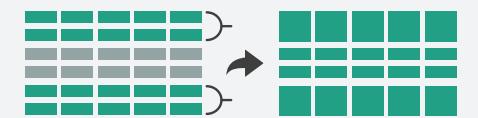


Reduce

① Filter



② Aggregate



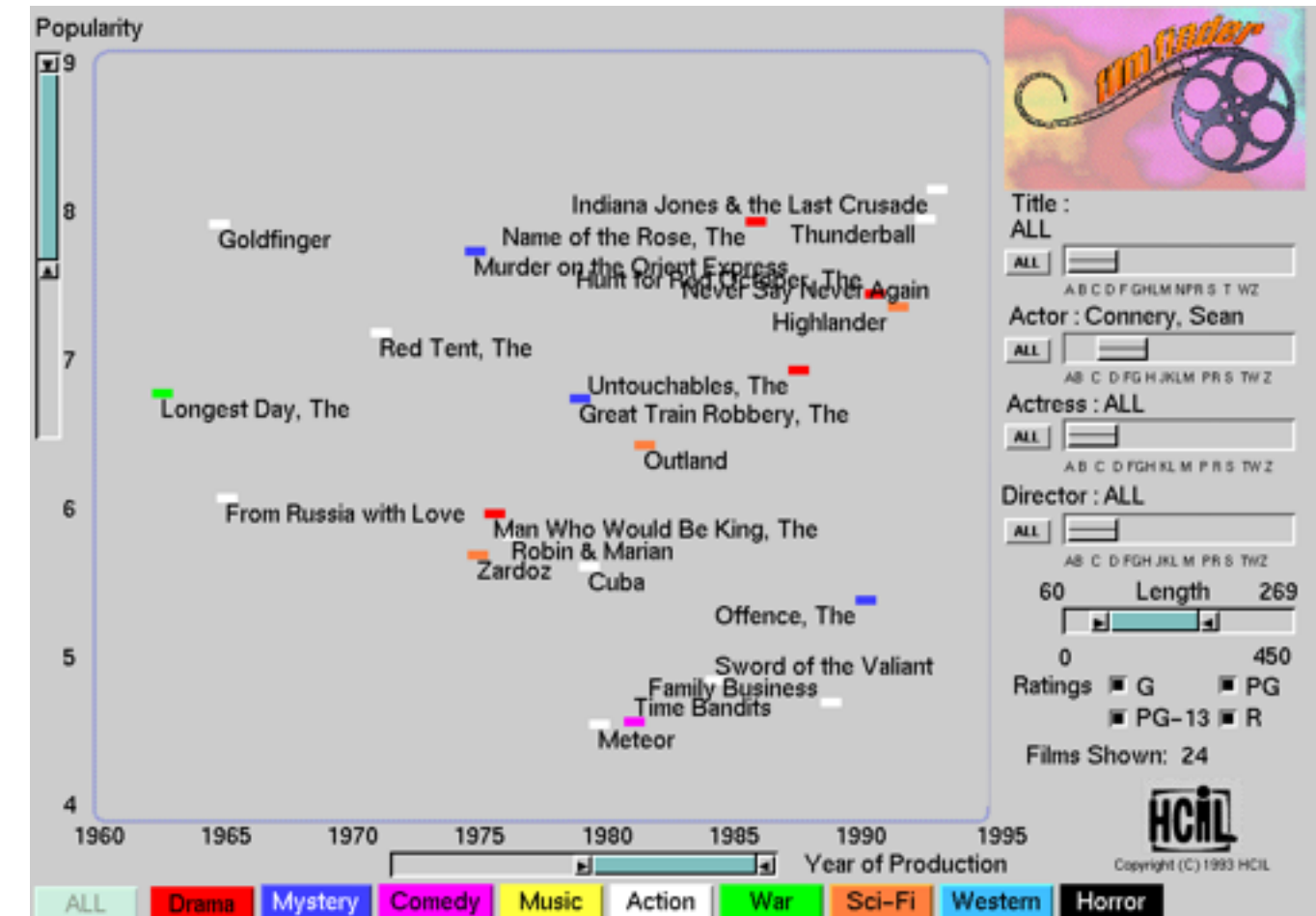
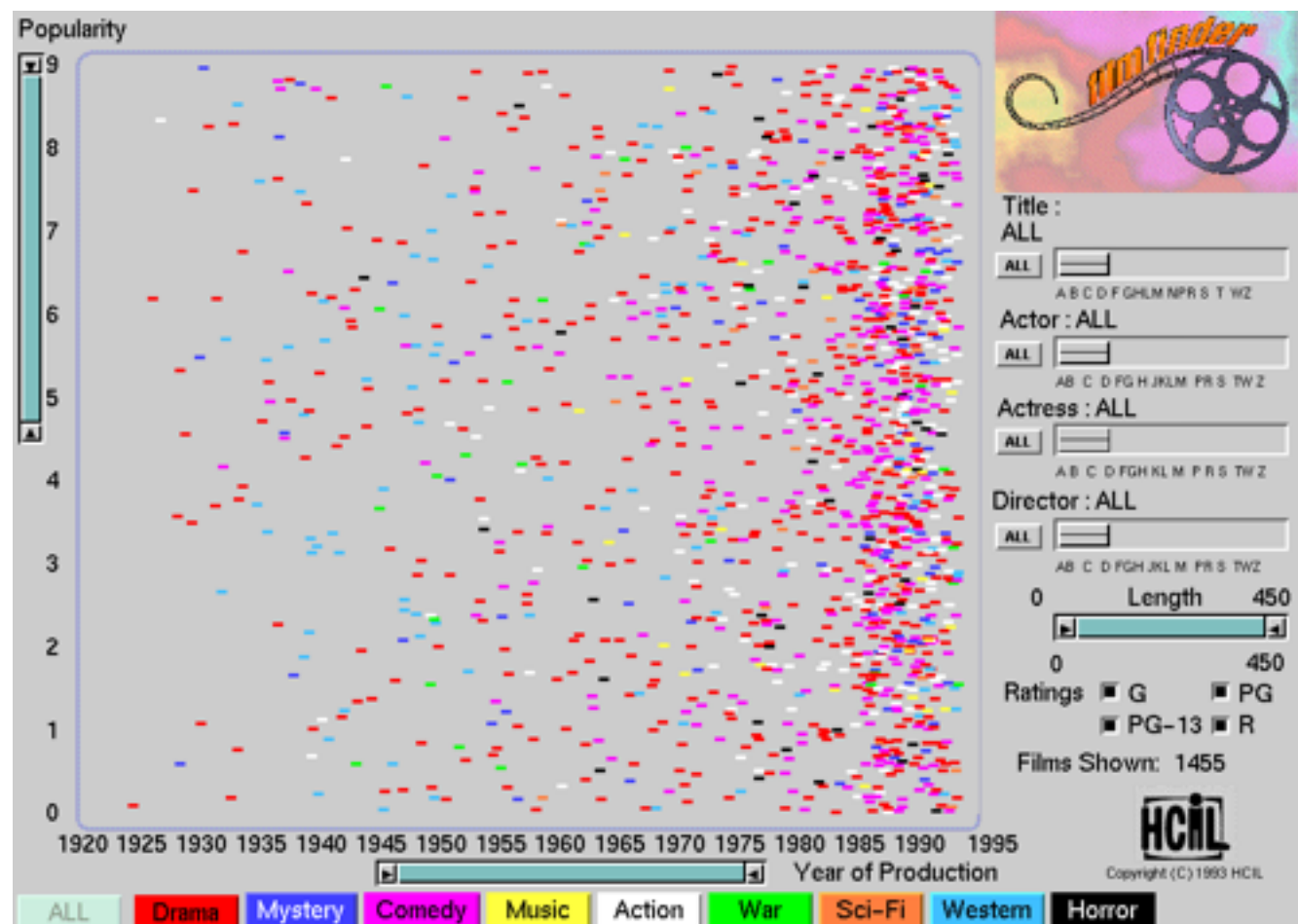
③ Embed



Idiom: dynamic filtering

System: FilmFinder

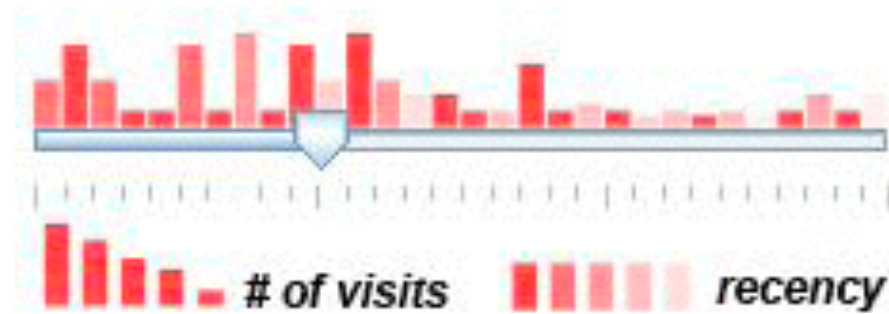
- item filtering
- browse through tightly coupled interaction
 - alternative to queries that might return far too many or too few



[Visual information seeking: Tight coupling of dynamic query filters with starfield displays. Ahlberg and Shneiderman. Proc. ACM Conf. on Human Factors in Computing Systems (CHI), pp. 313–317, 1994.]

Idiom: **scented widgets**

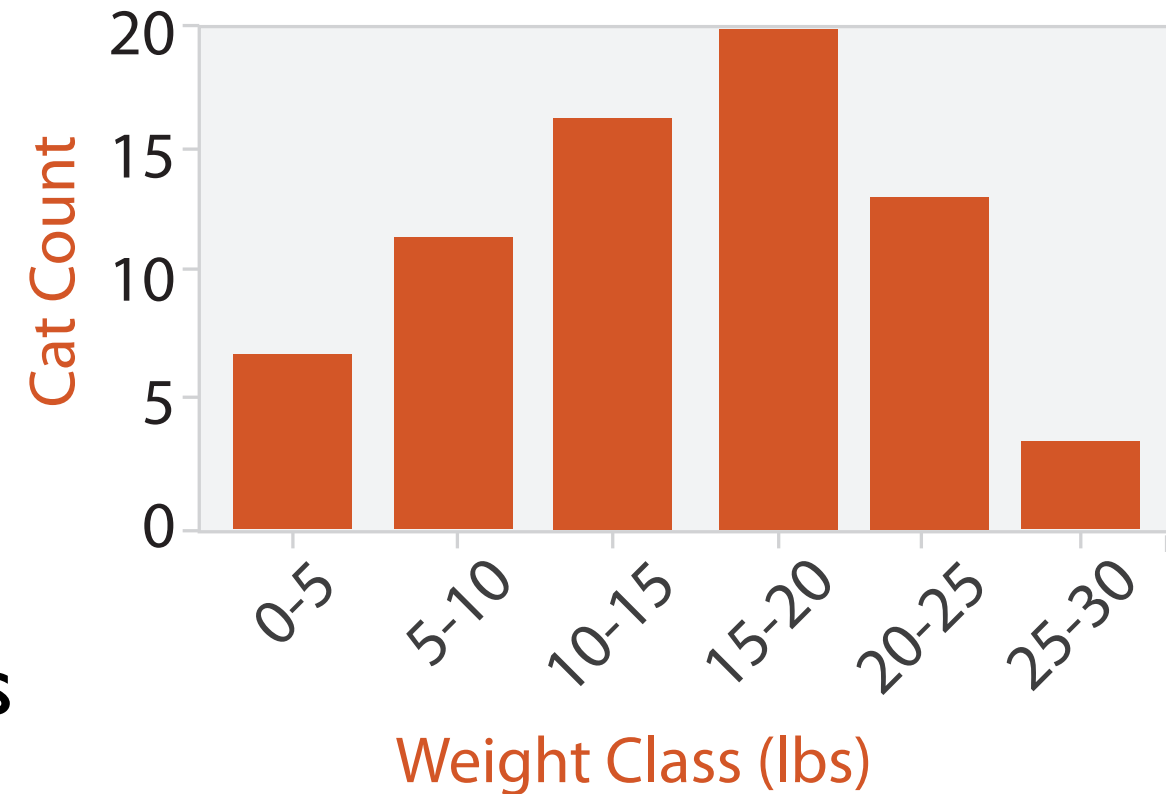
- augment widgets for filtering to show *information scent*
 - cues to show whether value in drilling down further vs looking elsewhere
- concise, in part of screen normally considered control panel



[Scented Widgets: Improving Navigation Cues with Embedded Visualizations. Willett, Heer, and Agrawala. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2007)* 13:6 (2007), 1129–1136.]

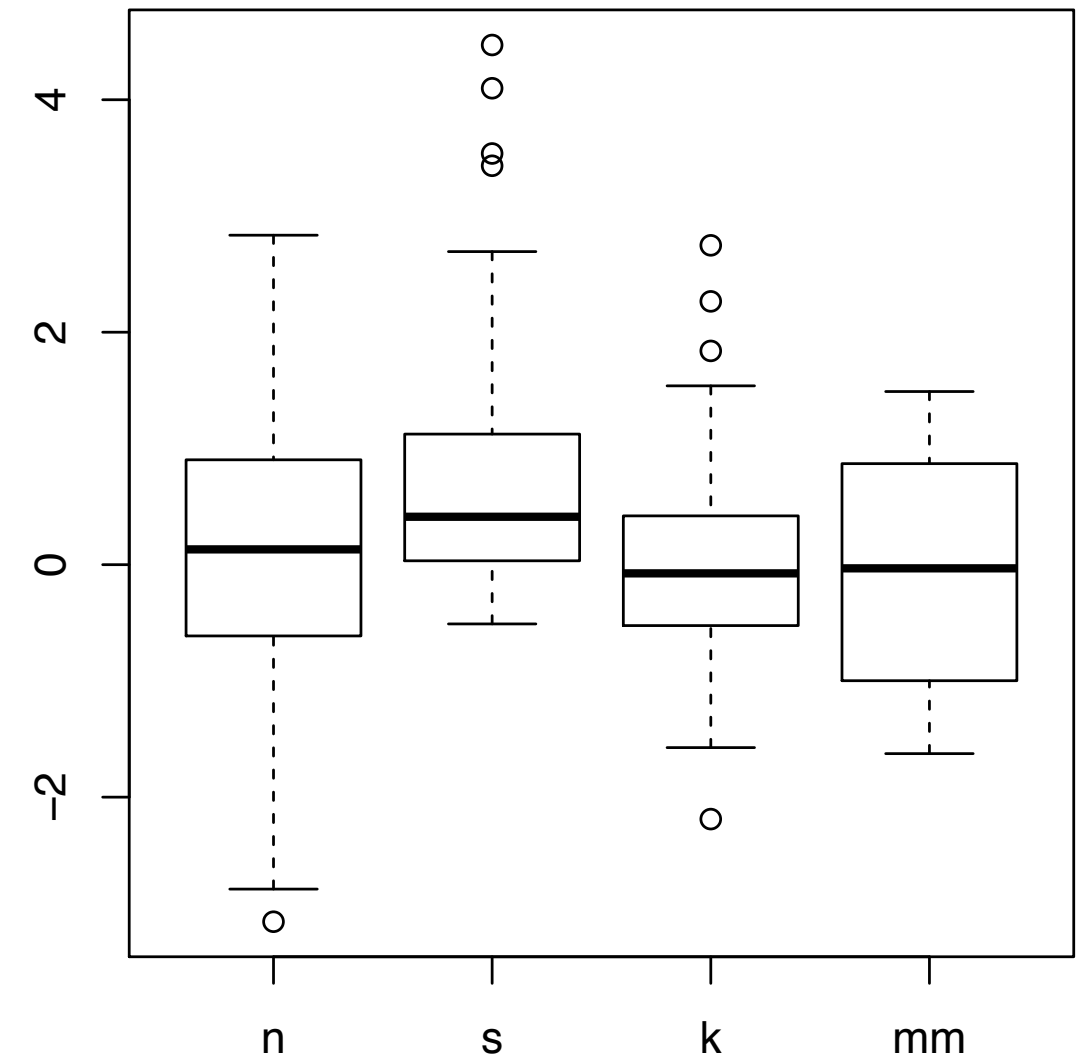
Idiom: **histogram**

- 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**

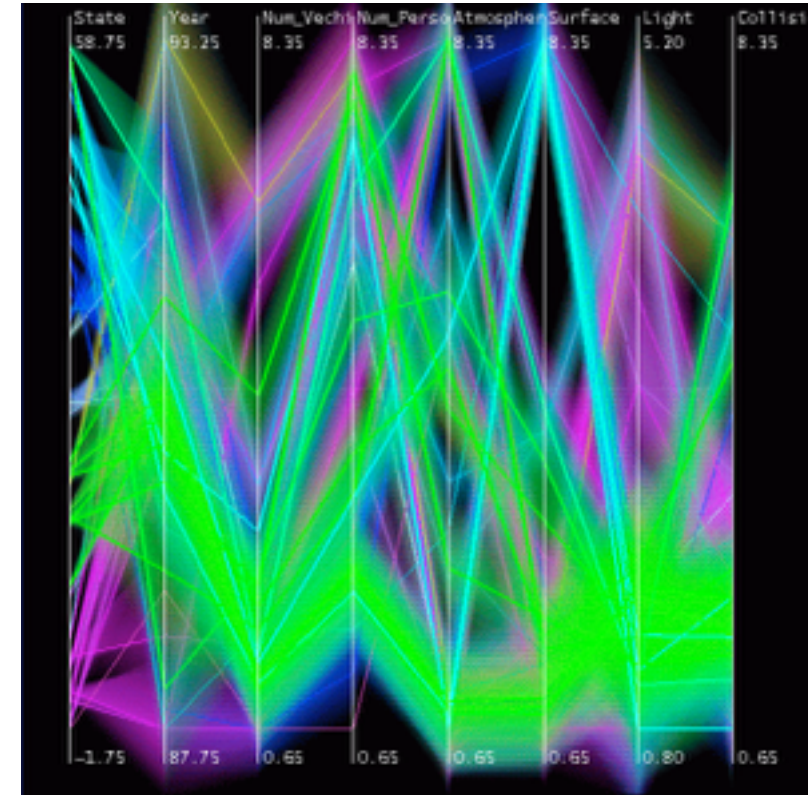
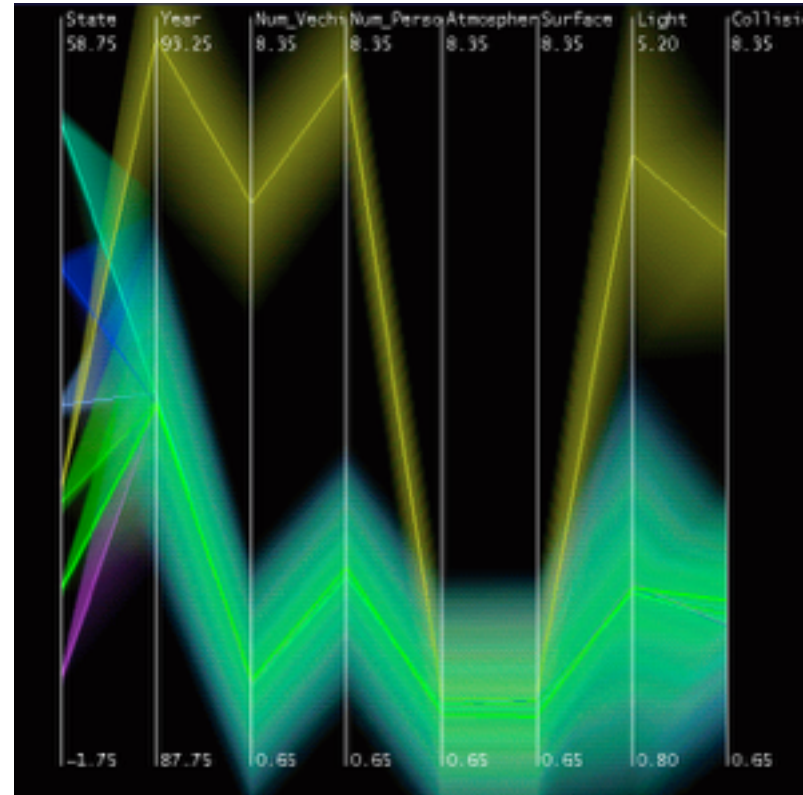
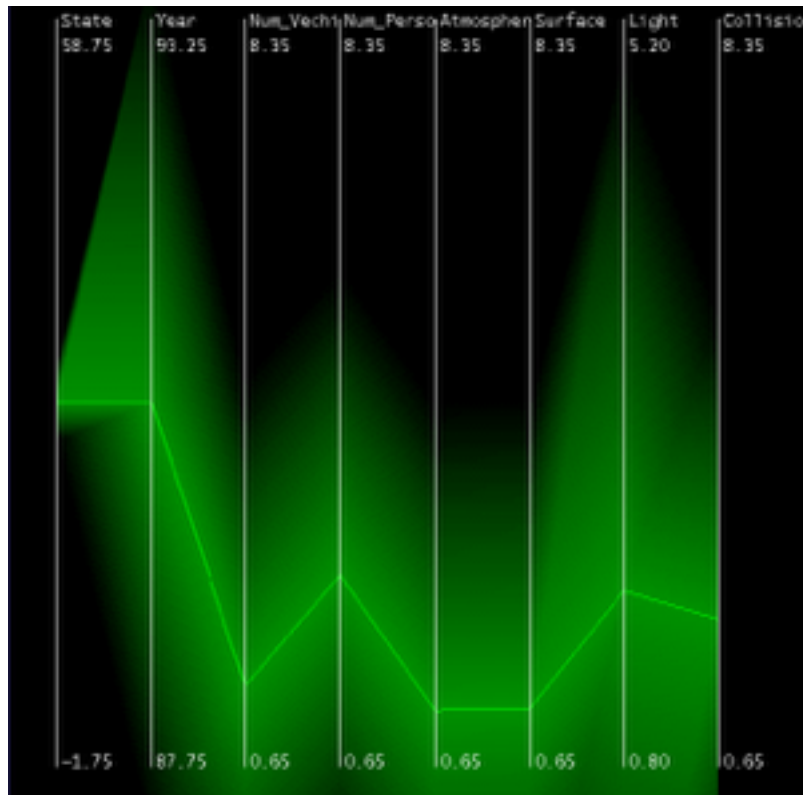
- static item aggregation
- task: find distribution
- data: table
- derived data
 - 5 quant attribs
 - median: central line
 - lower and upper quartile: boxes
 - lower upper fences: whiskers
 - values beyond which items are outliers
 - outliers beyond fence cutoffs explicitly shown



[40 years of boxplots. Wickham and Stryjewski. 2012. had.co.nz]

Idiom: Hierarchical parallel coordinates

- dynamic item aggregation
- derived data: *hierarchical clustering*
- encoding:
 - cluster band with variable transparency, line at mean, width by min/max values
 - color by proximity in hierarchy



[Hierarchical Parallel Coordinates for Exploration of Large Datasets. Fua, Ward, and Rundensteiner. Proc. IEEE Visualization Conference (Vis '99), pp. 43– 50, 1999.]

Dimensionality reduction

- attribute aggregation
 - derive low-dimensional target space from high-dimensional measured space
 - 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

Tumor
Measurement Data

data: 9D measured space

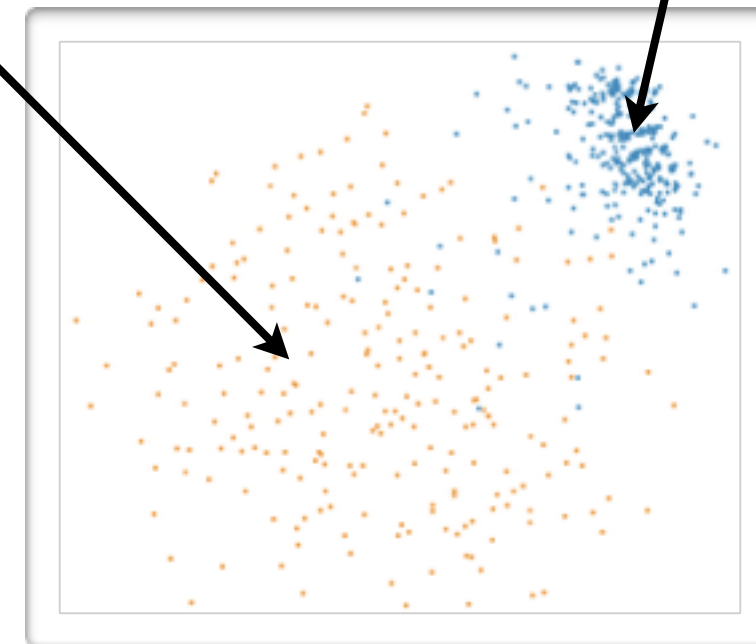


DR



Malignant

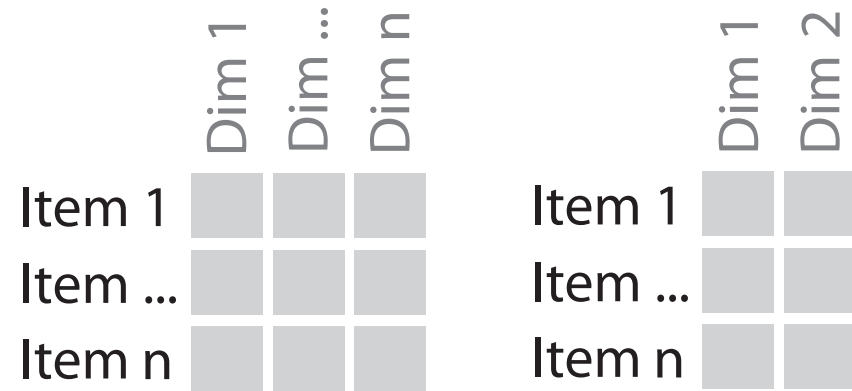
Benign



derived data: 2D target space

Idiom: Dimensionality reduction for documents

Task 1



In HD data → **Out** 2D data

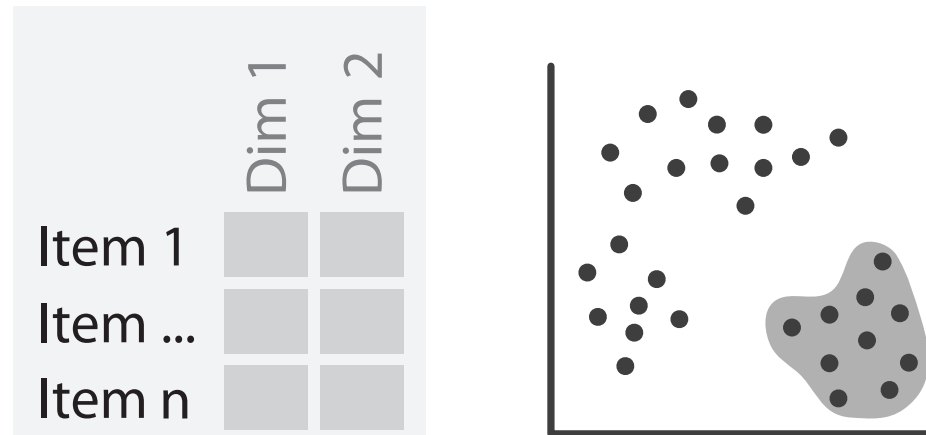
What?

Why?

- **In** High-dimensional data
- **Out** 2D data

- Produce
- Derive

Task 2



In 2D data → **Out** Scatterplot Clusters & points

What?

Why?

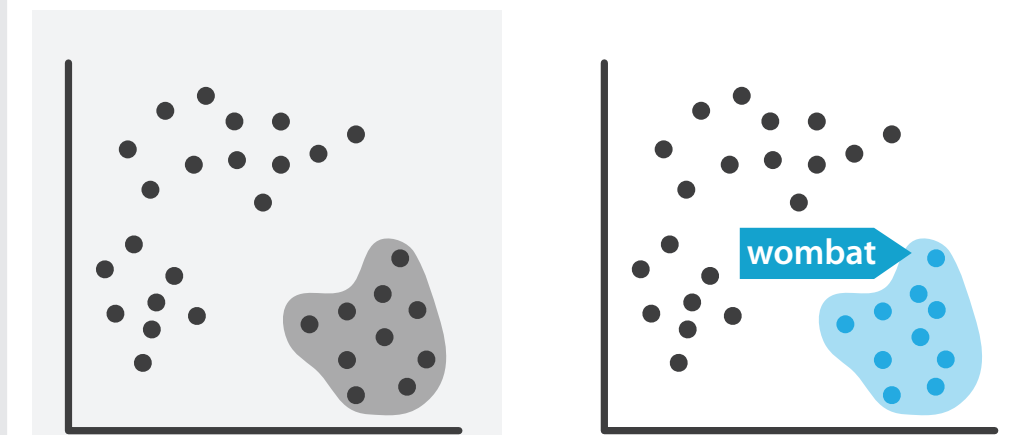
How?

- **In** 2D data
- **Out** Scatterplot
- **Out** Clusters & points

- Discover
- Explore
- Identify

- Encode
- Navigate
- Select

Task 3



In Scatterplot Clusters & points → **Out** Labels for clusters

What?

Why?

- **In** Scatterplot
- **In** Clusters & points
- **Out** Labels for clusters

- Produce
- Annotate

Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
 - Chap 13: Reduce Items and Attributes*
- *Hierarchical Aggregation for Information Visualization: Overview, Techniques and Design Guidelines*. Elmqvist and Fekete. IEEE Transactions on Visualization and Computer Graphics 16:3 (2010), 439–454.
- *A Review of Overview+Detail, Zooming, and Focus+Context Interfaces*. Cockburn, Karlson, and Bederson. ACM Computing Surveys 41:1 (2008), 1–31.
- *A Guide to Visual Multi-Level Interface Design From Synthesis of Empirical Study Evidence*. Lam and Munzner. Synthesis Lectures on Visualization Series, Morgan Claypool, 2010.

Outline

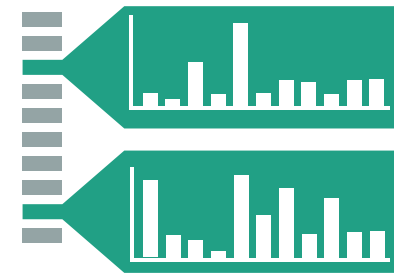
- **Session 1** 2:00-3:40pm
 - Analysis: What, Why, How
 - Marks and Channels
 - Arrange Tables
 - Arrange Spatial Data
 - Arrange Networks and Trees
- **Session 2** 4:15pm-5:50pm
 - Map Color and Other Channels
 - Manipulate: Change, Select, Navigate
 - Facet: Juxtapose, Partition, Superimpose
 - Reduce: Filter, Aggregate
 - **Embed: Focus+Context**

Embed: Focus+Context

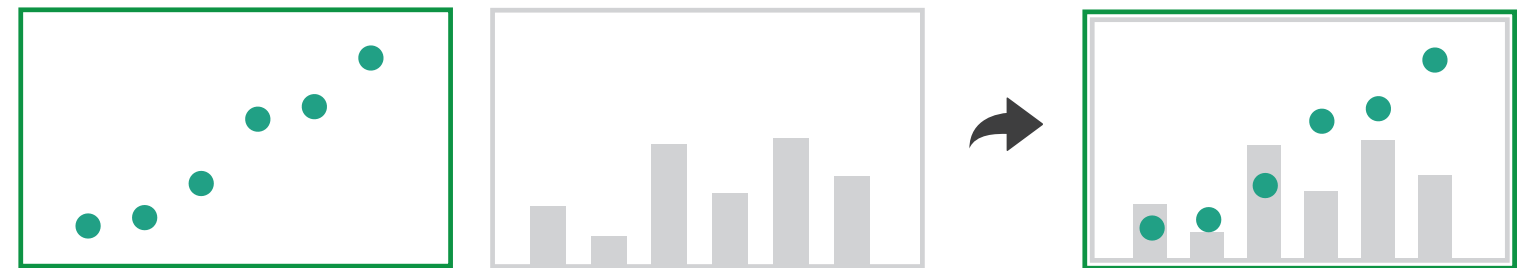
- combine information within single view
- elide
 - selectively filter and aggregate
- superimpose layer
 - local lens
- distort geometry
 - to make room for context

→ Embed

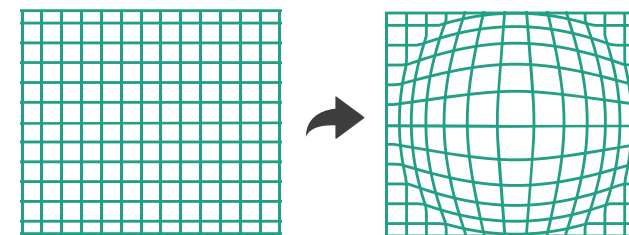
→ Elide Data



→ Superimpose Layer

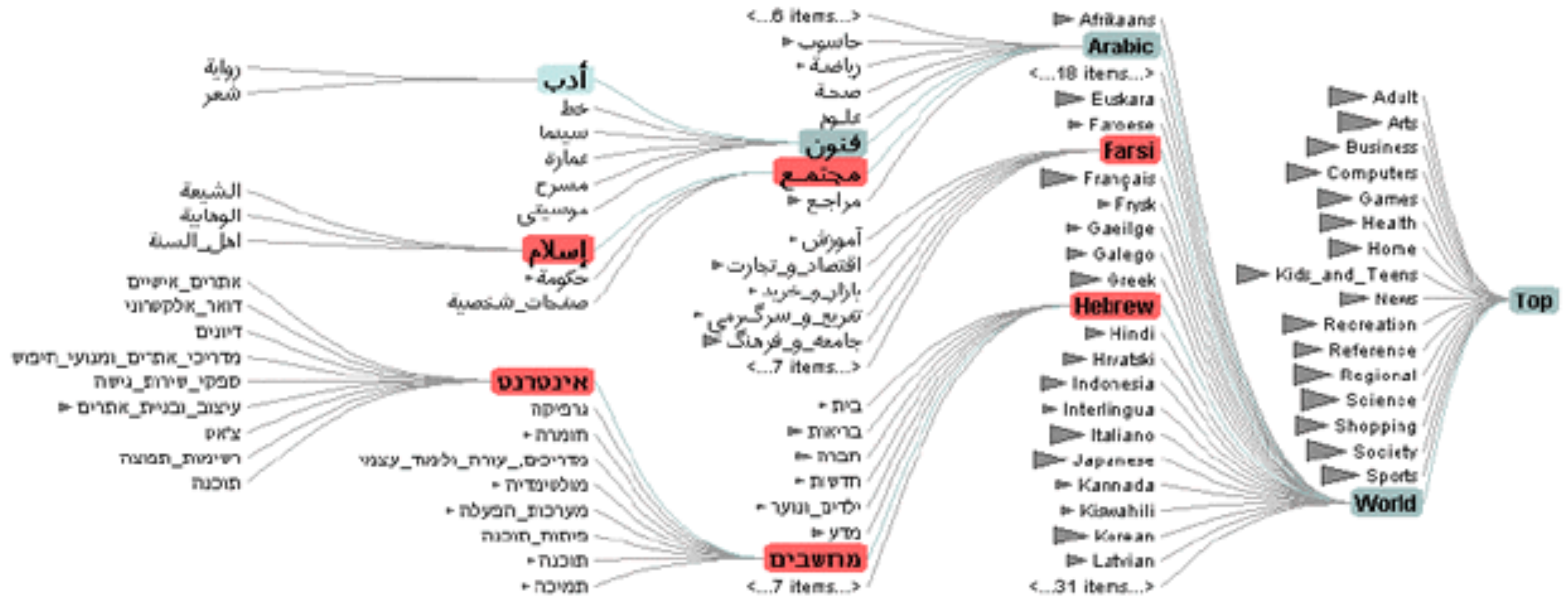


→ Distort Geometry



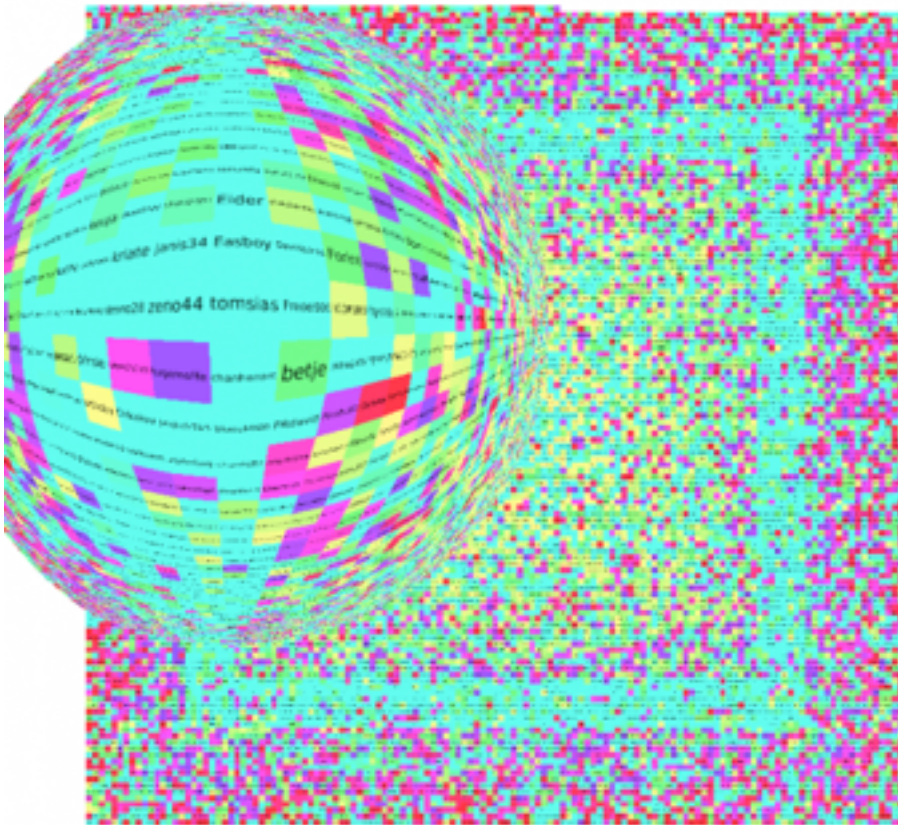
Idiom: DOI Trees Revisited

- elide
 - some items dynamically filtered out
 - some items dynamically aggregated together
 - some items shown in detail

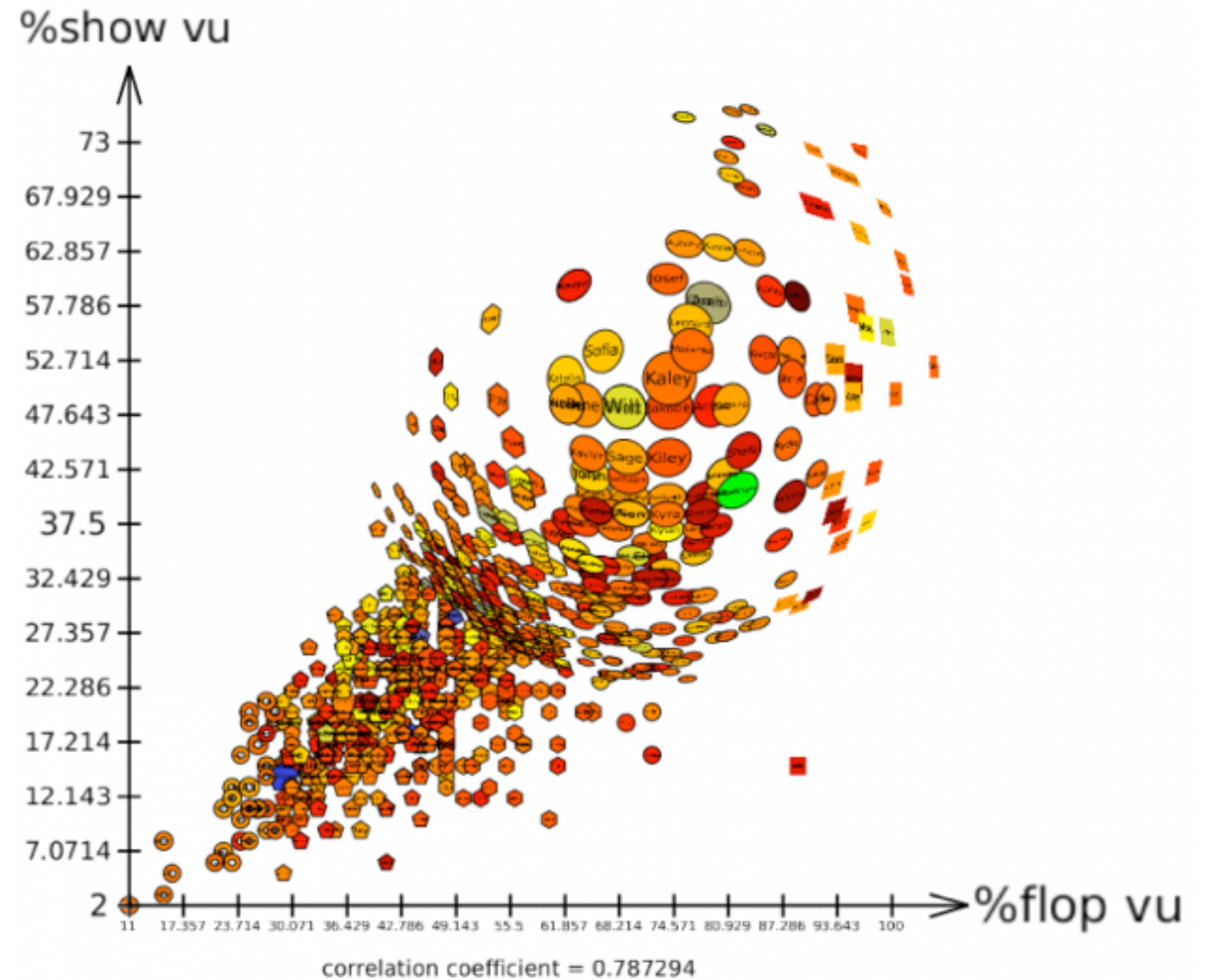


Idiom: **Fisheye Lens**

- distort geometry
 - shape: radial
 - focus: single extent
 - extent: local
 - metaphor: draggable lens



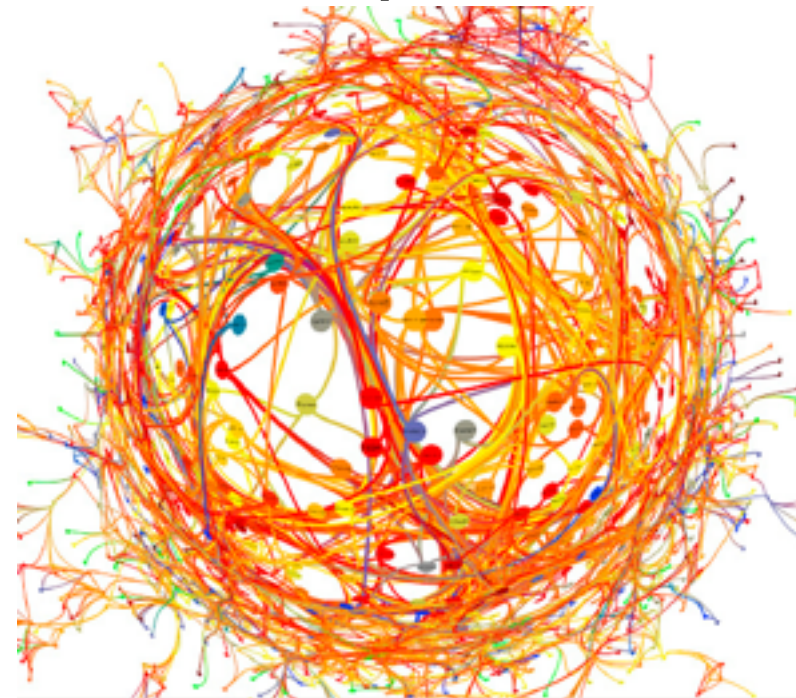
<http://tulip.labri.fr/TulipDrupal/?q=node/351>
<http://tulip.labri.fr/TulipDrupal/?q=node/371>



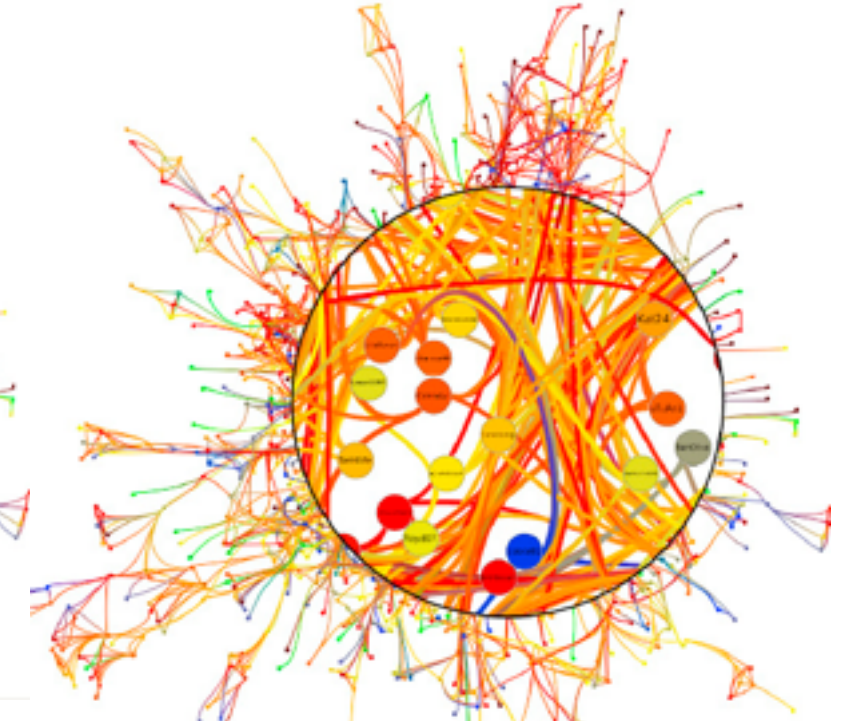
Distortion costs and benefits

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

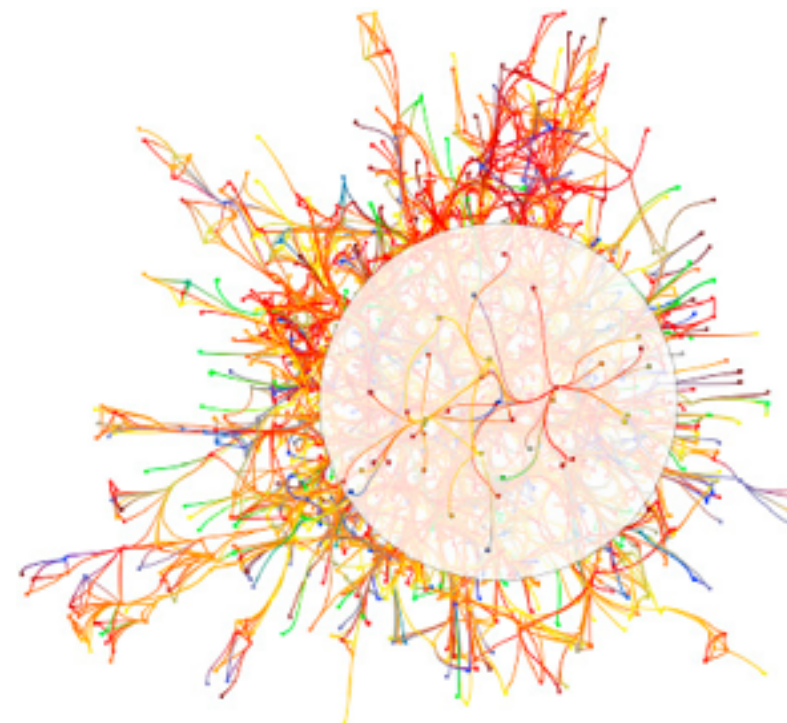
fisheye lens



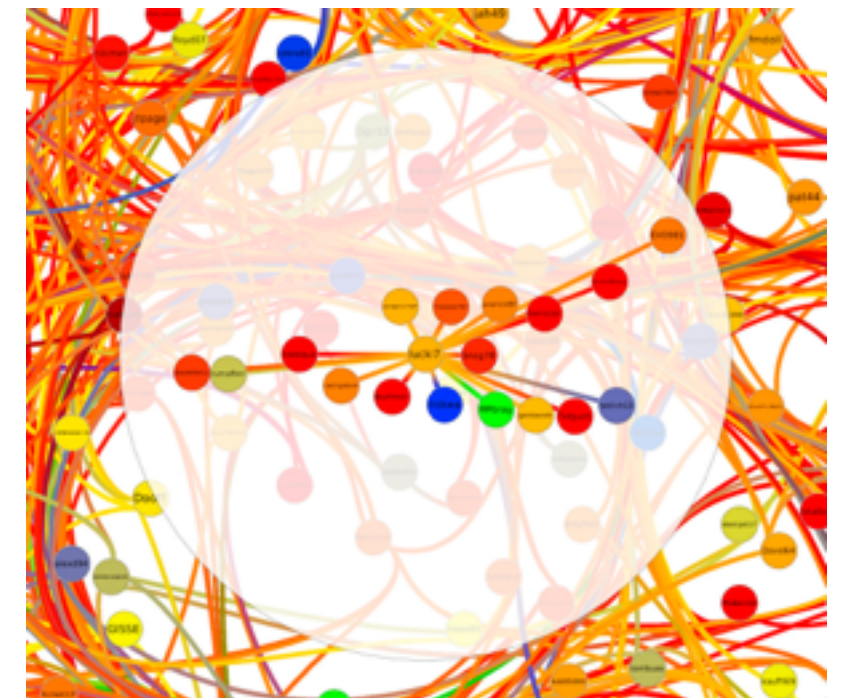
magnifying lens



neighborhood layering



Bring and Go



Further reading

- Visualization Analysis and Design. Munzner. AK Peters Visualization Series, CRC Press, Nov 2014.
 - *Chap 14: Embed: Focus+Context*
- *A Fisheye Follow-up: Further Reflection on Focus + Context*. Furnas. Proc. ACM Conf. Human Factors in Computing Systems (CHI), pp. 999–1008, 2006.
- *A Review of Overview+Detail, Zooming, and Focus+Context Interfaces*. Cockburn, Karlson, and Bederson. ACM Computing Surveys 41:1 (2008), 1–31.
- *A Guide to Visual Multi-Level Interface Design From Synthesis of Empirical Study Evidence*. Lam and Munzner. Synthesis Lectures on Visualization Series, Morgan Claypool, 2010.

Not covered today

- **Rules of Thumb**

- No unjustified 3D

- Power of the plane, dangers of depth
 - Occlusion hides information
 - Perspective distortion loses information
 - Tilted text isn't legible

- No unjustified 2D

- Resolution over immersion

- Overview first, zoom and filter, details on demand

- Function first, form next

What?

Datasets

Attributes

domain

abstraction

What?

Why?

idiom

How?

algorithm

Why?

Actions

Targets

→ Data Types

→ Items

→ Data and D

Tables

Items

Attributes

→ Analyze

→ Consume

→ Discover



→ Present



→ Enjoy



→ All Data

→ Trends



→ Outliers

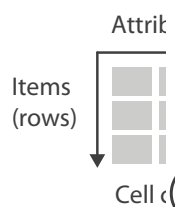


→ Features



→ Dataset Typ

→ Tables



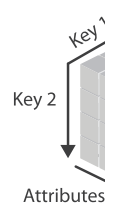
→ Produce

→ Annotate



→ Search

→ Multidir



	Target
Location known	
Location unknown	

→ Arrange

→ Express



→ Separate



→ Order



→ Align



→ Use



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



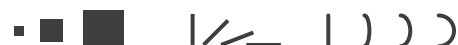
→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape



→ Motion

Direction, Rate, Frequency, ...



How?

Encode

Manipulate

Facet

Reduce

→ Arrange

→ Express



→ Separate



→ Order



→ Align



→ Use



→ Map

from **categorical** and **ordered** attributes

→ Color

→ Hue



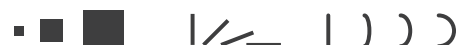
→ Saturation



→ Luminance



→ Size, Angle, Curvature, ...



→ Shape

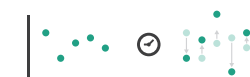


→ Motion

Direction, Rate, Frequency, ...



→ Change



→ Select



→ Navigate



→ Juxtapose



→ Partition



→ Superimpose



→ Filter



→ Aggregate



→ Embed



What?

Why?

→ Geometr



→ Query

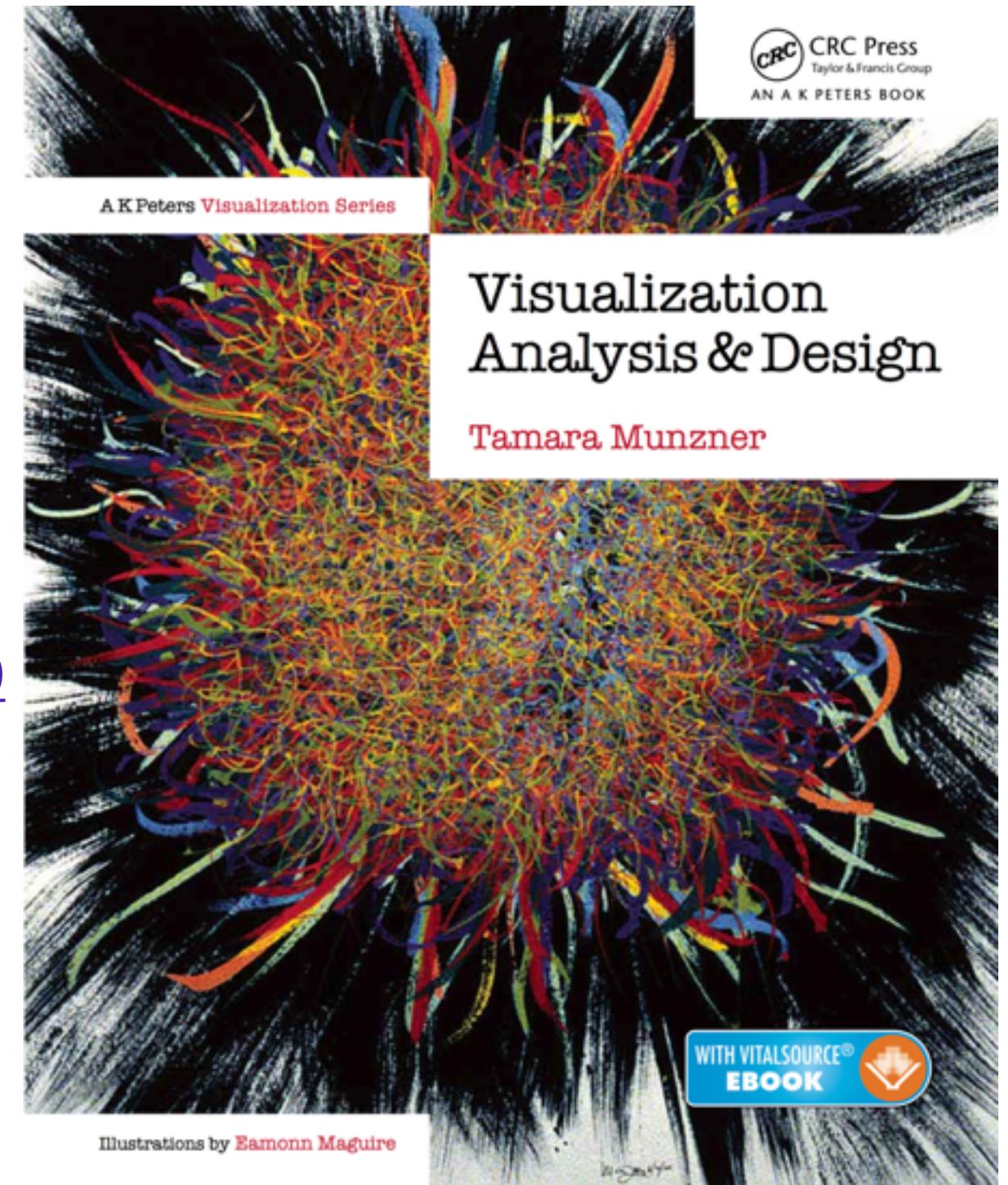
→ Identify



More Information

@tamaramunzner

- this tutorial
<http://www.cs.ubc.ca/~tmm/talks.html#halfdaycourse16>
- book
<http://www.cs.ubc.ca/~tmm/vadbook>
 - 20% promo code for book+ebook combo:
HVN17
 - <http://www.crcpress.com/product/isbn/9781466508910>
 - illustration acknowledgement: Eamonn Maguire
- full courses, papers, videos, software, talks
<http://www.cs.ubc.ca/group/infovis>
<http://www.cs.ubc.ca/~tmm>



Visualization Analysis and Design.
Munzner. A K Peters Visualization Series, CRC Press, Visualization Series, 2014.