2IMV20

Encoding: tables, trees, and graphs

Technische Universiteit
Eindhoven
University of Technology

Recap

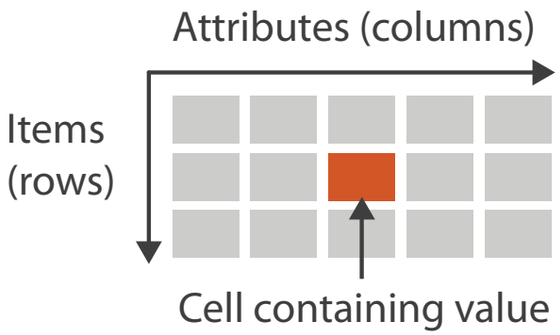
Design framework

- **What** is shown?
  - data abstraction
- **Why** is the user looking at it?
  - task abstraction
- **How** is it shown?
  - visual encoding and interaction

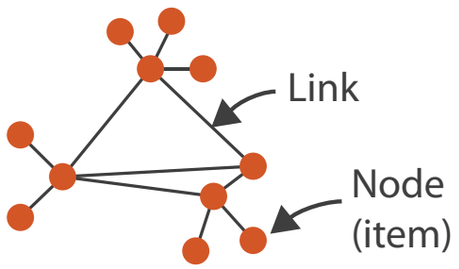
Munzner, **A nested model for visualization design and validation**. IEEE Trans. Visualization and Computer Graphics, 15(6):921-928, 2009.

What: Datasets and data types

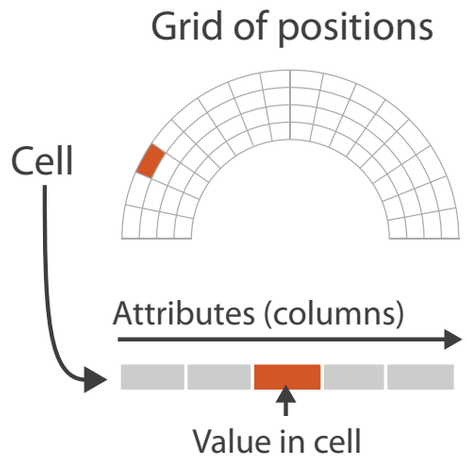
Tables



Networks

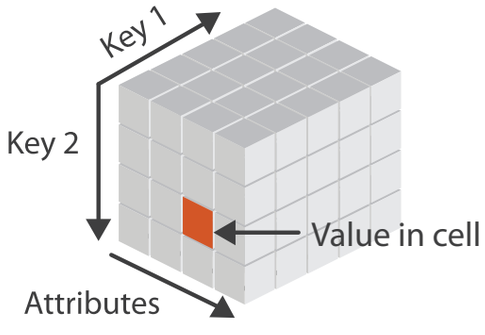


Fields

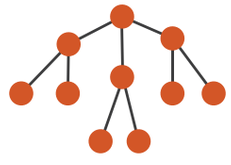


Geometry

Multi-dimensional table



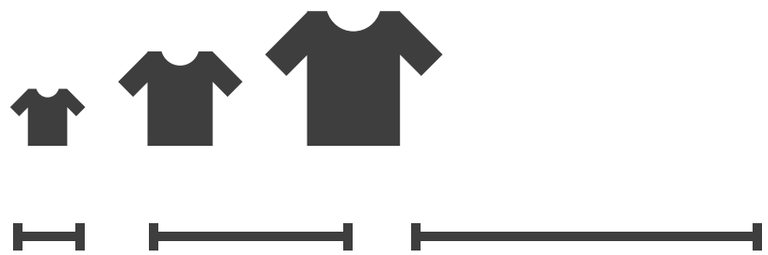
Trees



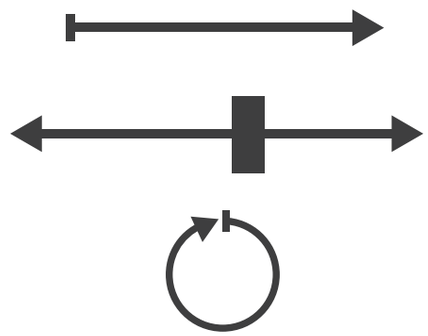
What: attribute types

- Categorical

- Ordered
  - ordinal
  - quantitative



- Ordering direction
  - sequential
  - diverging
  - cyclic



Why: Tasks

- User goals
  - High level actions: analyze
    - ➔ Consume
      - ➔ *Discover*
      - ➔ *Present*
      - ➔ *Enjoy*

    - ➔ Produce
      - ➔ *Annotate*
      - ➔ *Record*
      - ➔ *Derive*

  - Mid level actions: search

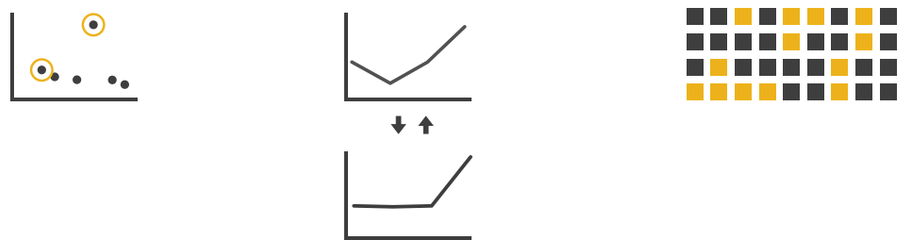
➔ **Search**

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

  - Low level actions: query

➔ **Query**

➔ Identify ➔ Compare ➔ Summarise

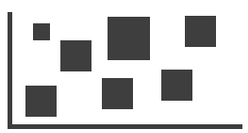


How: encoding preview

Arrange

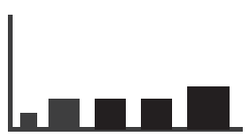
➔ Express

➔ Separate



➔ Order

➔ Align



➔ Use

Map

from categorical and ordered attributes

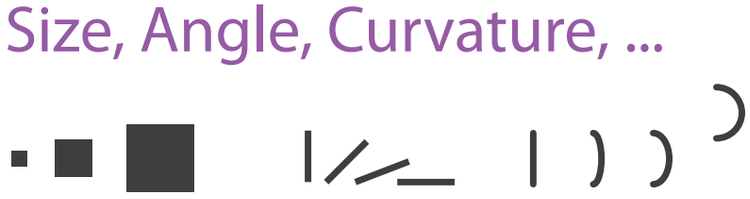
➔ Color

➔ *Hue*

➔ *Saturation*

➔ *Luminance*

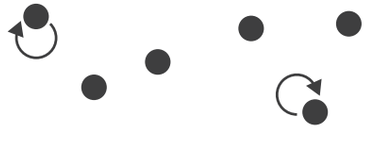
➔ Size, Angle, Curvature, ...



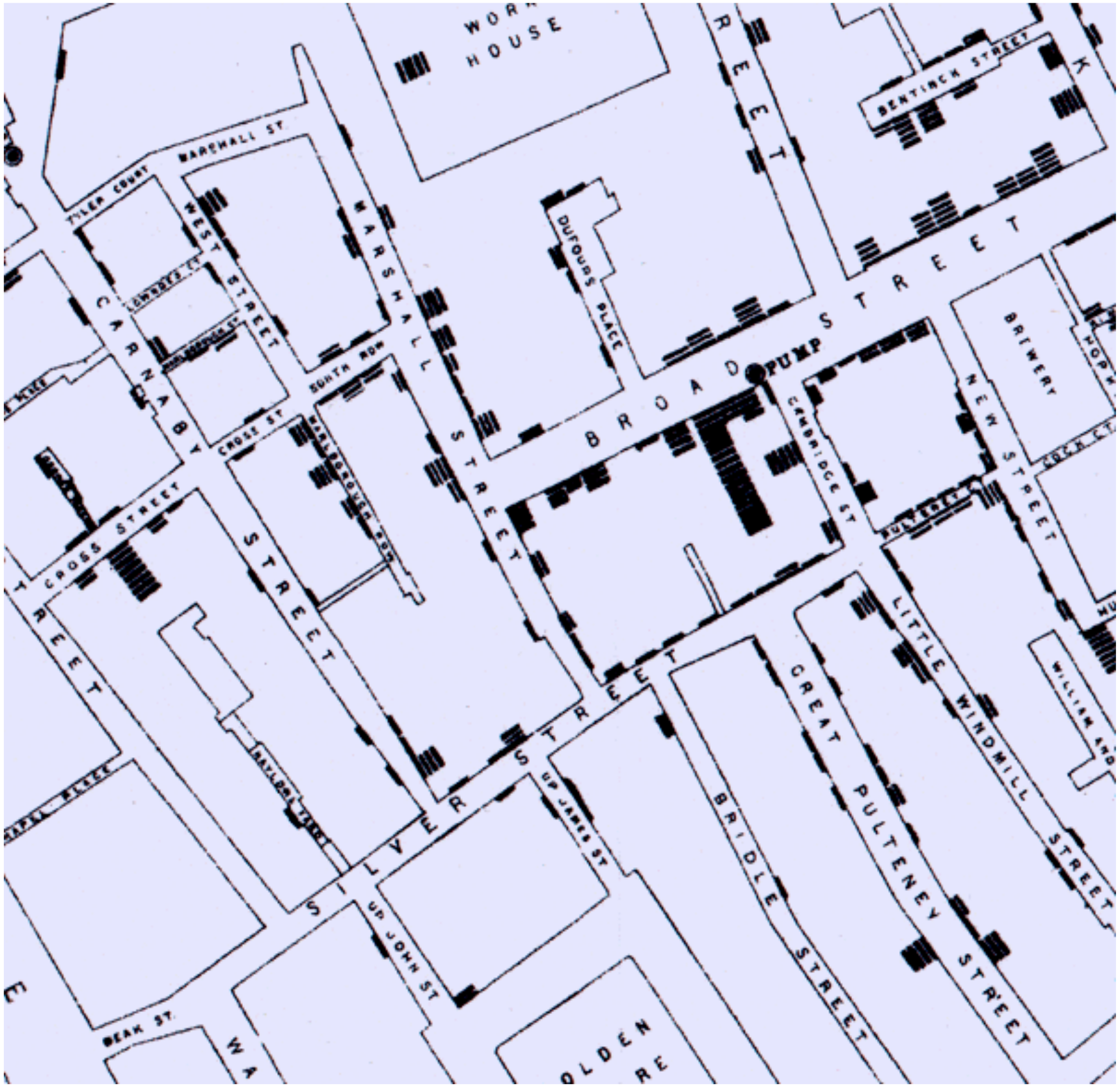
➔ Shape

➔ Motion

Direction, Rate, Frequency, ...

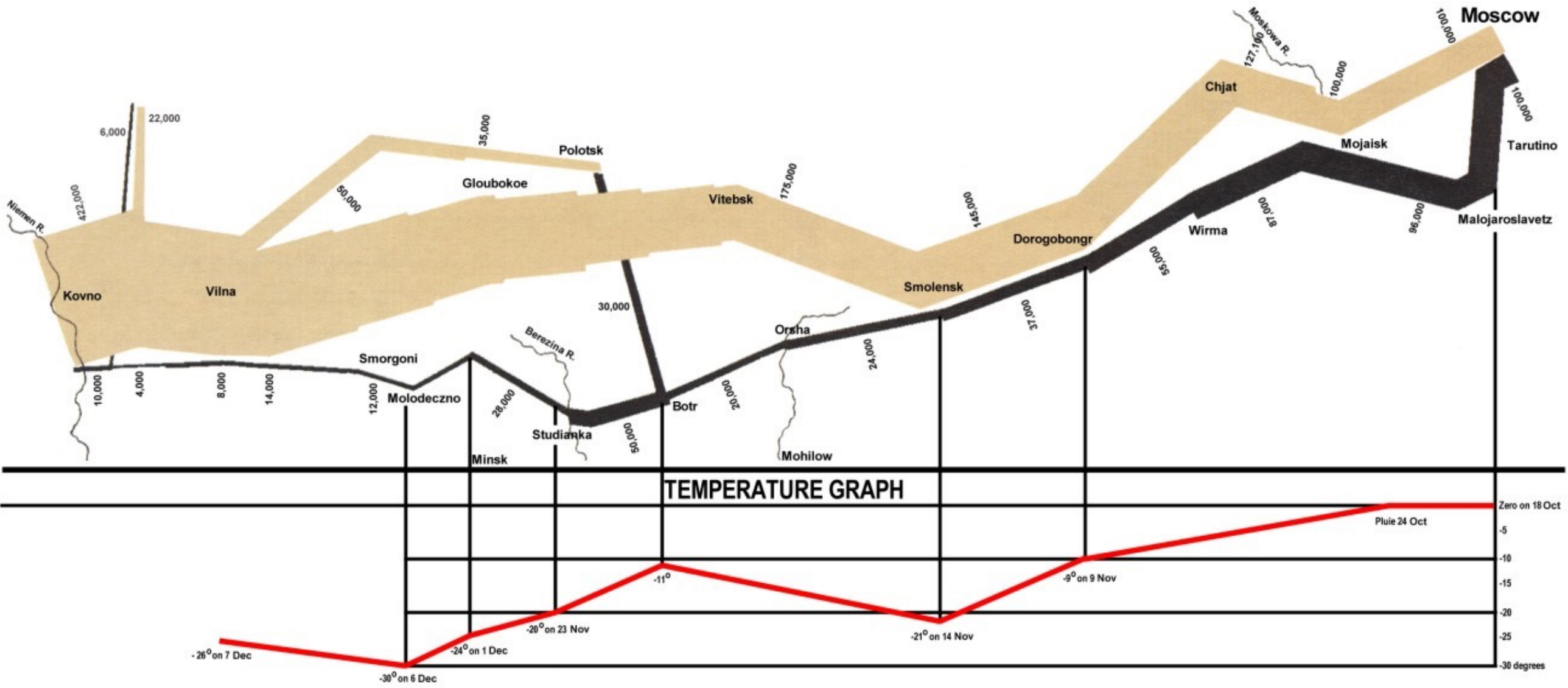


Historical examples: Cholera Map of London



J. Snow, 1854

Historical examples: Napoleon's retreat from Russia



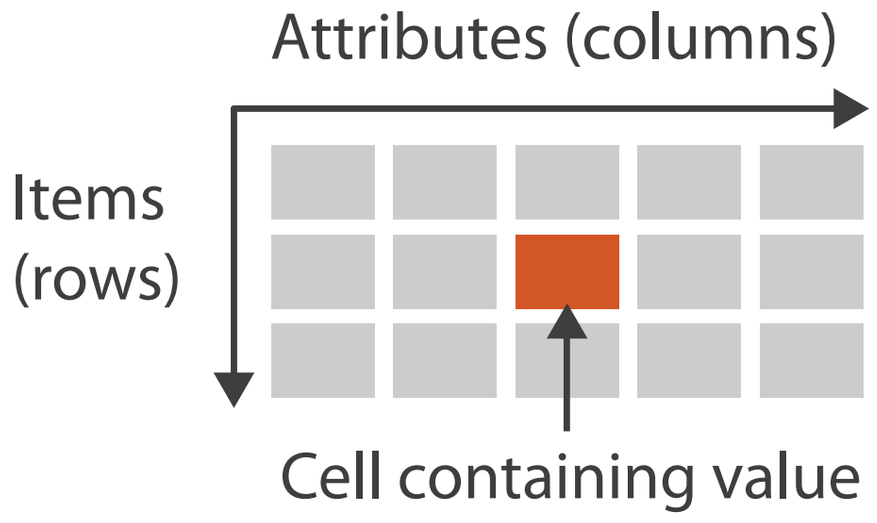
C.J. Minard, 1861

Tabular data

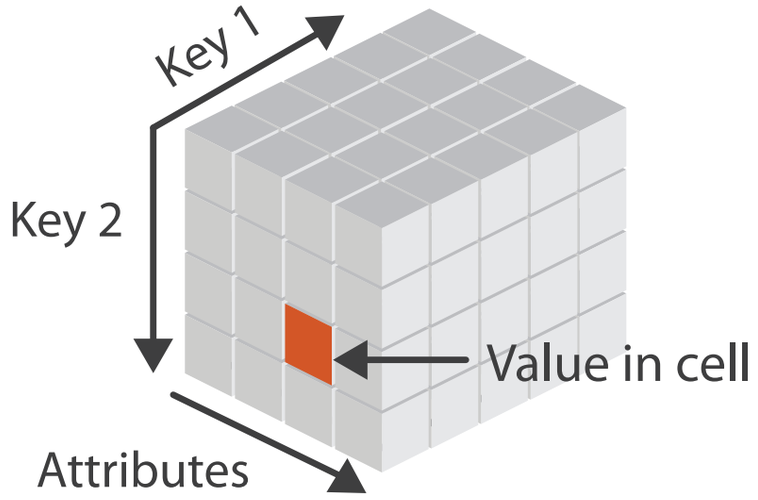
Vehicle Name	Small/Sporty/ Compact/Large Sedan	Sports Car	SUV	Wagon	Minivan	Pickup	AWD	RWD	Retail Price	Dealer Cost	Engine Size (l)	Cyl	HP	City MPG	Hwy MPG	Weight	Wheel Base	Len	Width
Toyota 4Runner SR5 V6	0	0	1	0	0	0	0	0	27710	24801	4	6	245	18	21	4035	110	189	74
Toyota Avalon XL 4dr	1	0	0	0	0	0	0	0	26560	23693	3	6	210	21	29	3417	107	192	72
Toyota Avalon XLS 4dr	1	0	0	0	0	0	0	0	30920	27271	3	6	210	21	29	3439	107	192	72
Toyota Camry LE 4dr	1	0	0	0	0	0	0	0	19560	17558	2.4	4	157	24	33	3086	107	189	71
Toyota Camry LE V6 4dr	1	0	0	0	0	0	0	0	22775	20325	3	6	210	21	29	3296	107	189	71
Toyota Camry Solara SE 2dr	1	0	0	0	0	0	0	0	19635	17722	2.4	4	157	24	33	3175	107	193	72
Toyota Camry Solara SE V6 2dr	1	0	0	0	0	0	0	0	21965	19819	3.3	6	225	20	29	3417	107	193	72
Toyota Camry Solara SLE V6 2dr	1	0	0	0	0	0	0	0	26510	23908	3.3	6	225	20	29	3439	107	193	72
Toyota Camry XLE V6 4dr	1	0	0	0	0	0	0	0	25920	23125	3	6	210	21	29	3362	107	189	71
Toyota Celica GT-S 2dr	0	1	0	0	0	0	0	0	22570	20363	1.8	4	180	24	33	2500	102	171	68
Toyota Corolla CE 4dr	1	0	0	0	0	0	0	0	14085	13065	1.8	4	130	32	40	2502	102	178	67
Toyota Corolla LE 4dr	1	0	0	0	0	0	0	0	15295	13889	1.8	4	130	32	40	2524	102	178	67
Toyota Corolla S 4dr	1	0	0	0	0	0	0	0	15030	13650	1.8	4	130	32	40	2524	102	178	67
Toyota Echo 2dr auto	1	0	0	0	0	0	0	0	11560	10896	1.5	4	108	33	39	2085	93	163	65
Toyota Echo 2dr manual	1	0	0	0	0	0	0	0	10760	10144	1.5	4	108	35	43	2035	93	163	65
Toyota Echo 4dr	1	0	0	0	0	0	0	0	11290	10642	1.5	4	108	35	43	2055	93	163	65
Toyota Highlander V6	0	0	1	0	0	0	1	0	27930	24915	3.3	6	230	18	24	3935	107	185	72
Toyota Land Cruiser	0	0	1	0	0	0	1	0	54765	47986	4.7	8	325	13	17	5390	112	193	76
Toyota Matrix XR	0	0	0	1	0	0	0	0	16695	15156	1.8	4	130	29	36	2679	102	171	70
Toyota MR2 Spyder convertible 2dr	0	1	0	0	0	0	0	1	25130	22787	1.8	4	138	26	32	2195	97	153	67
Toyota Prius 4dr (gas/electric)	1	0	0	0	0	0	0	0	20510	18926	1.5	4	110	59	51	2890	106	175	68
Toyota RAV4	0	0	1	0	0	0	1	0	20290	18553	2.4	4	161	22	27	3119	98	167	68
Toyota Sequoia SR5	0	0	1	0	0	0	1	0	35695	31827	4.7	8	240	14	17	5270	118	204	78
Toyota Sienna CE	0	0	0	0	1	0	0	0	23495	21198	3.3	6	230	19	27	4120	119	200	77
Toyota Sienna XLE Limited	0	0	0	0	1	0	0	0	28800	25690	3.3	6	230	19	27	4165	119	200	77
Toyota Tacoma	0	0	0	0	0	1	0	1	12800	11879	2.4	4	142	22	27	2750	103	*	*
Toyota Tundra Access Cab V6 SR5	0	0	0	0	0	1	1	0	25935	23520	3.4	6	190	14	17	4435	128	*	*
Toyota Tundra Regular Cab V6	0	0	0	0	0	1	0	1	16495	14978	3.4	6	190	16	20	3925	128	*	*

Tabular data

- Key
  - independent attribute, used as index
- Value
  - dependent attribute, value of cell
- Also called **multivariate** data



➔ *Multidimensional Table*



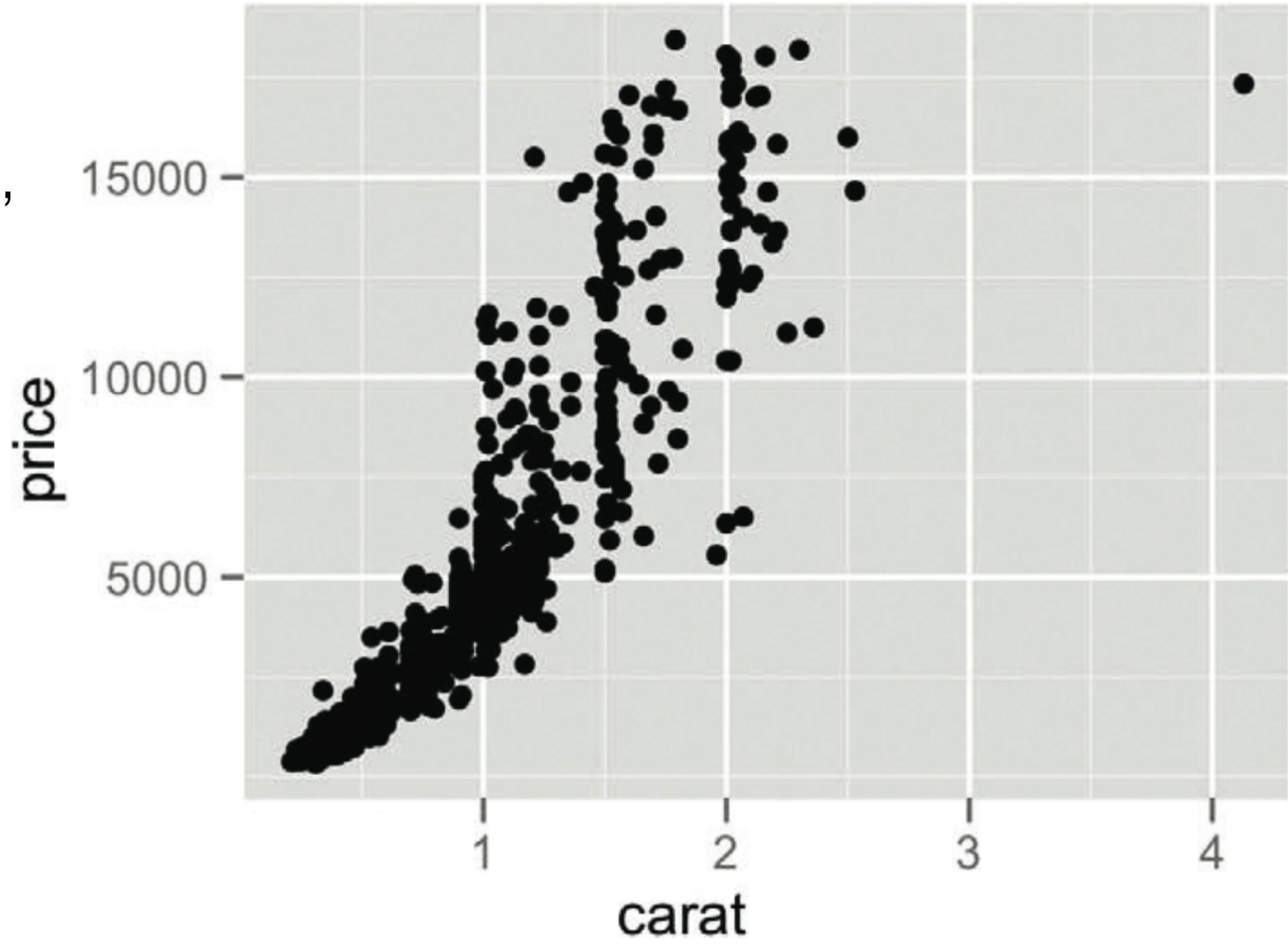
Tabular data: visualization techniques

- Point-based
  - project records from n-dimensional data space to k-dimensional display space
  - example: scatterplot
- Line-based
  - link points corresponding to particular record together with lines
  - example: parallel coordinate plot
- Region-based
  - convey values by filled polygons (size, shape, color,…)
  - example: bar chart

Point-based techniques

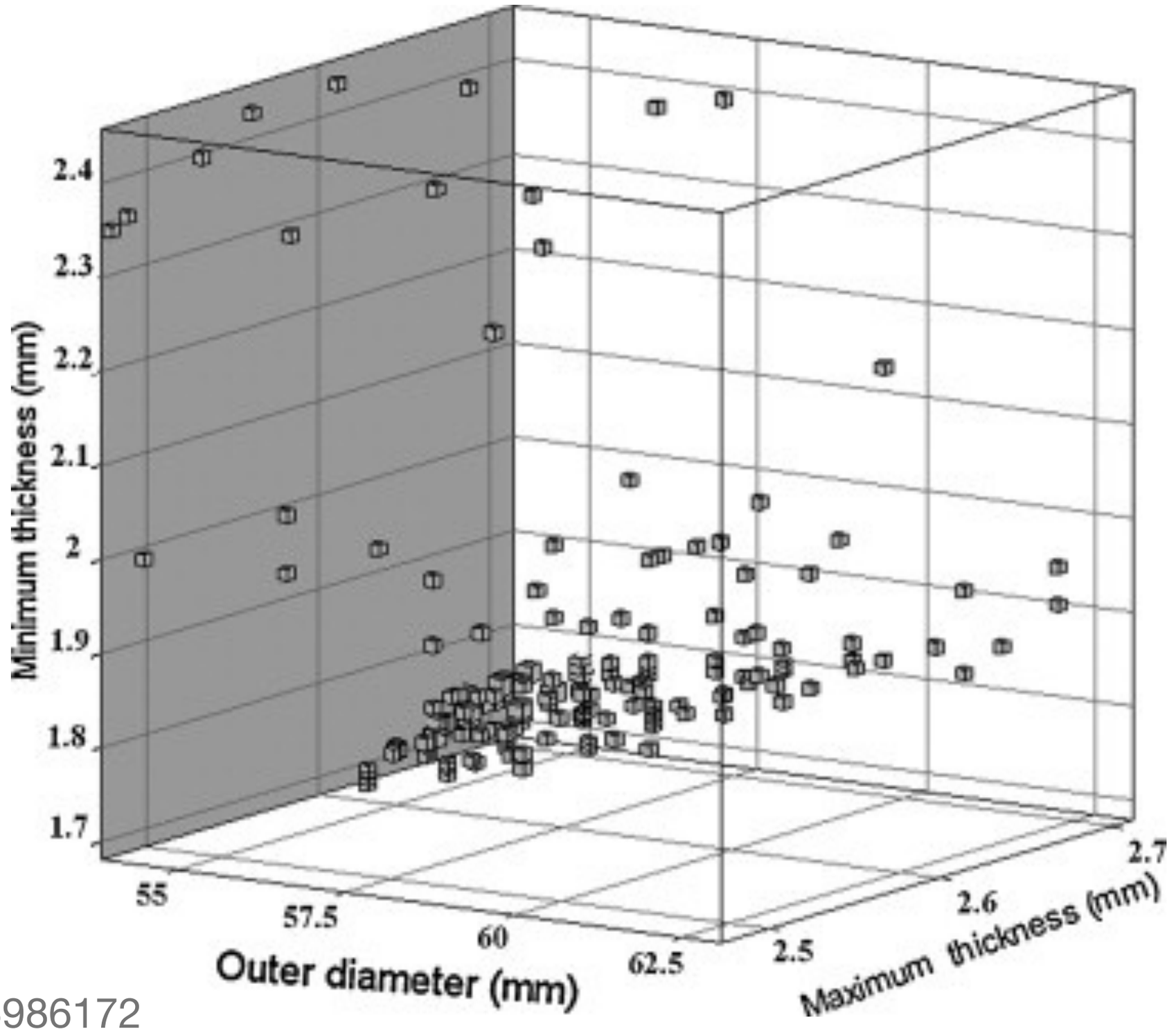
Scatter plot

- What
  - two quantitative value attributes
- How
  - horizontal/vertical position + shape
- Task
  - find trends, outliers, distribution, correlation, clusters
- Scale
  - hundreds of items



Encoding multiple keys in a scatterplot

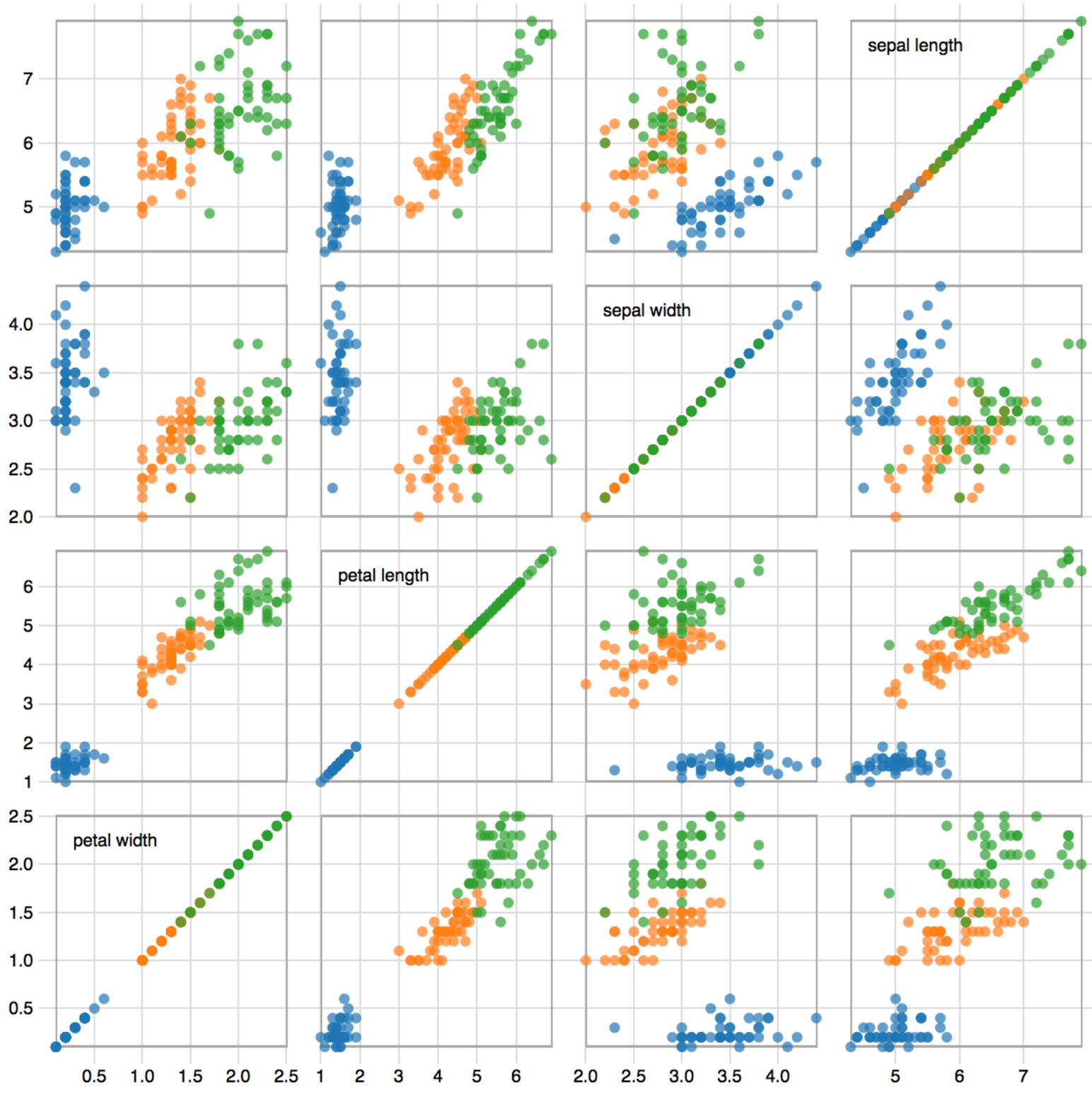
- 3D a good idea?



http://bl.ocks.org/hlvoorhees/5986172

Scatter plot matrix (SPLOM)

- What
  - multi-key table
- How
  - 2D matrix of scatterplots
- Task
  - find correlation, trends, outliers, clusters
- Scale
  - hundreds of items
  - about a dozen attributes



http://bl.ocks.org/mbostock/3213173

Line-based techniques

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

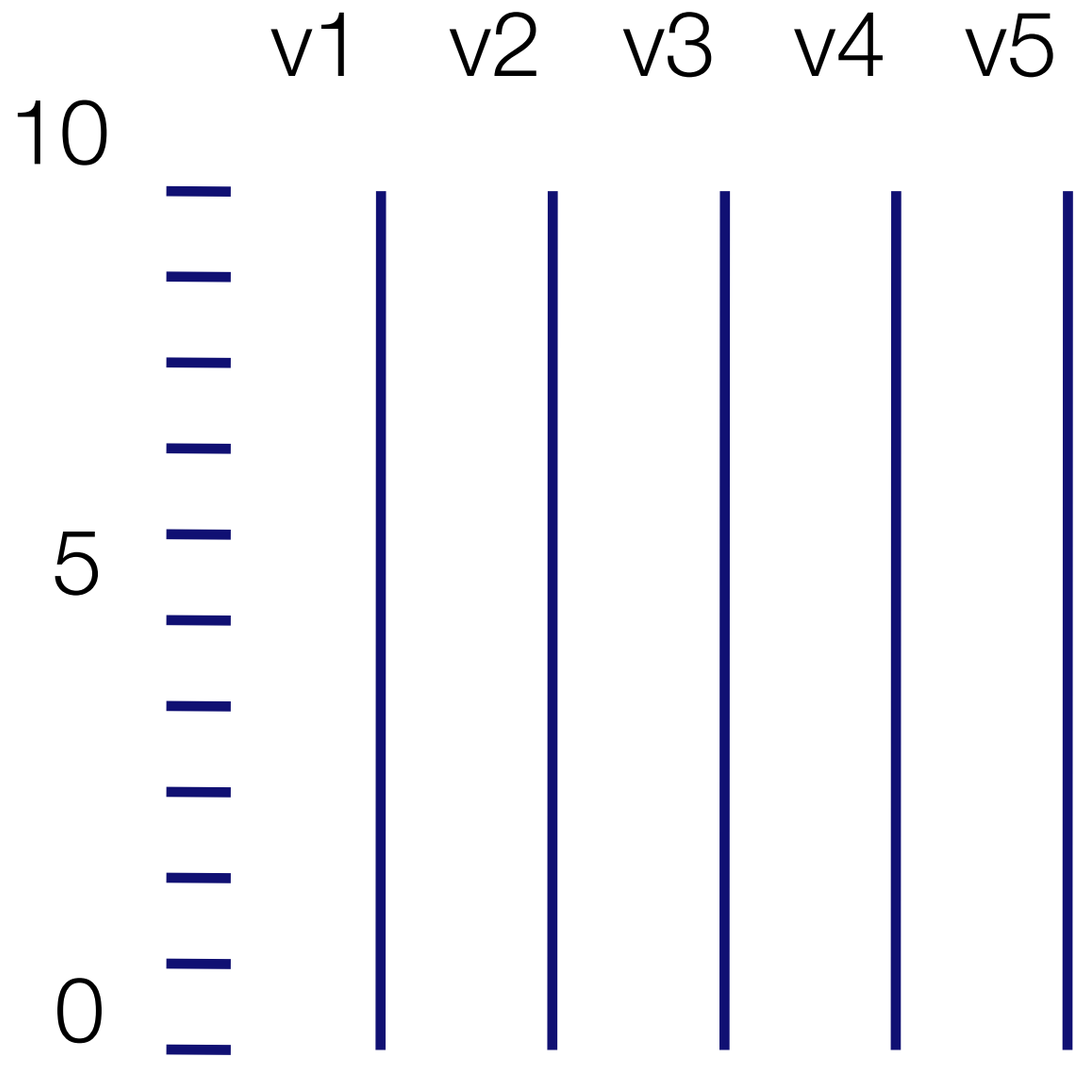
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes



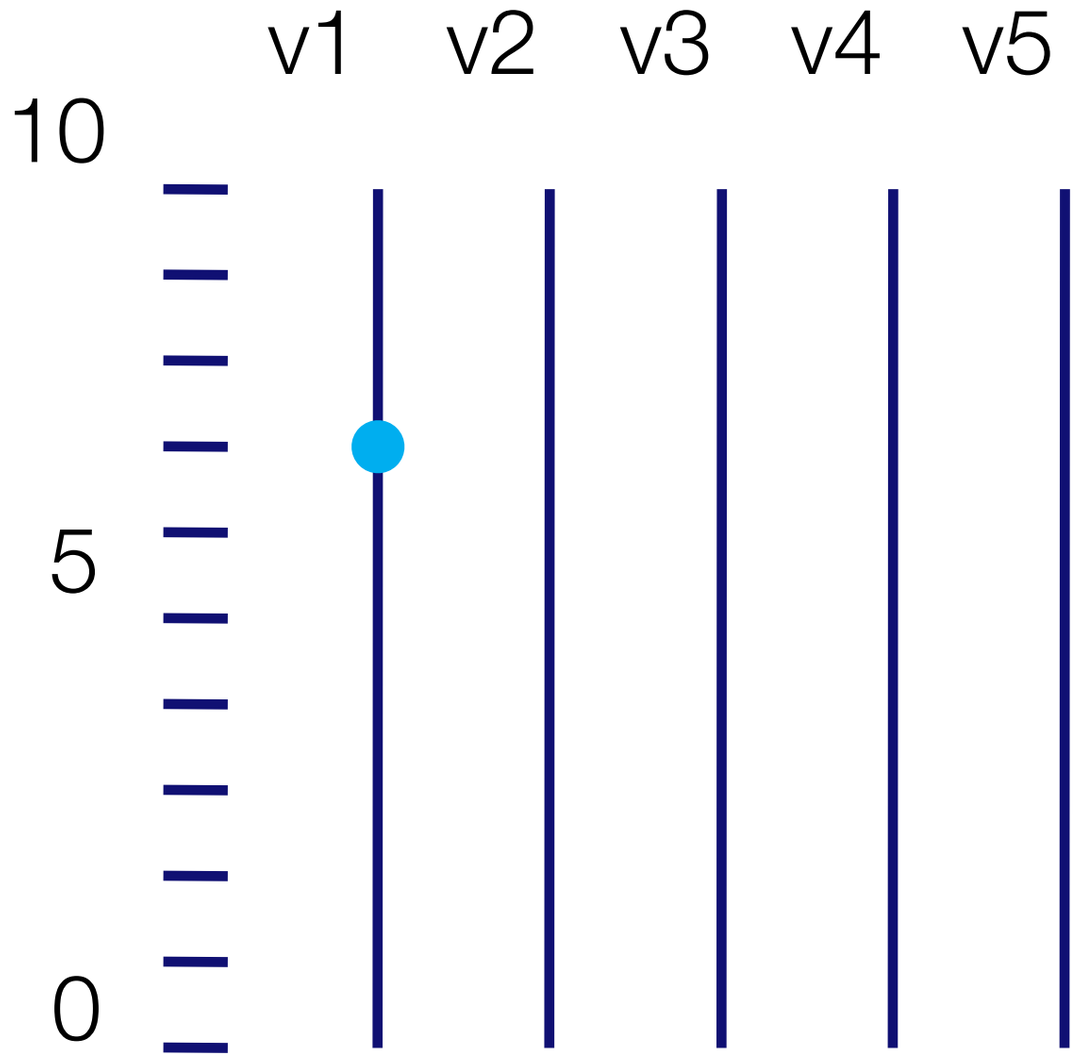
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis



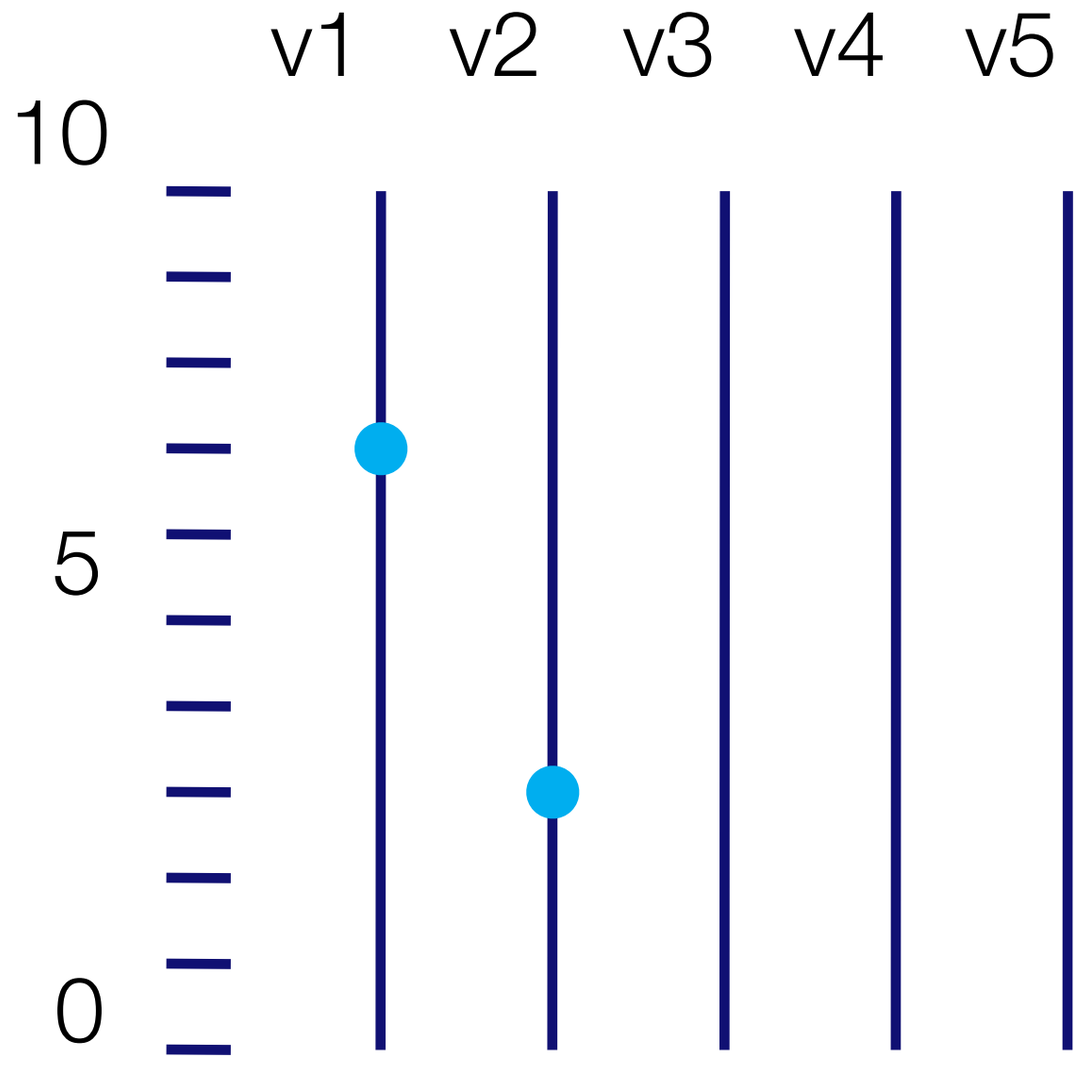
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis



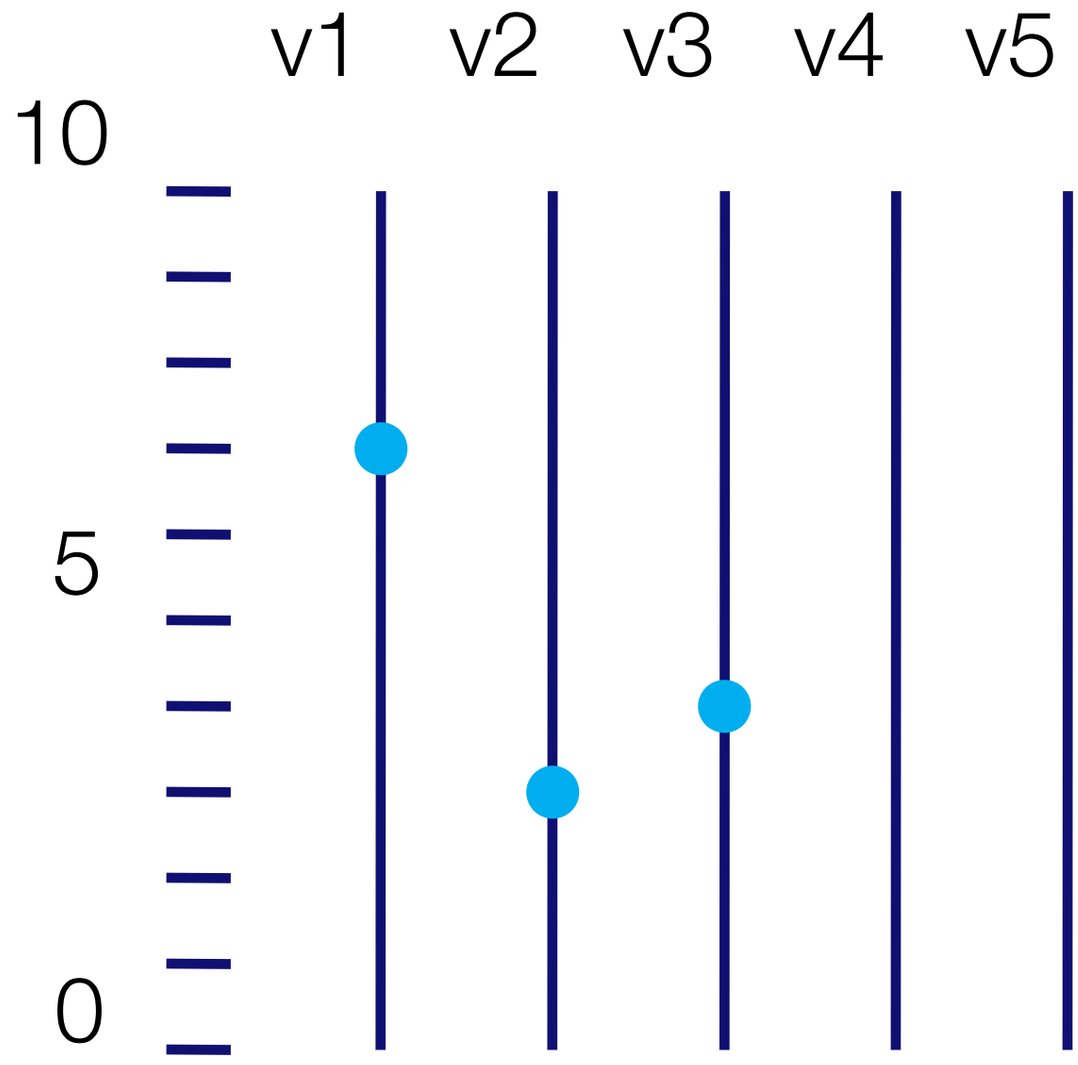
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis



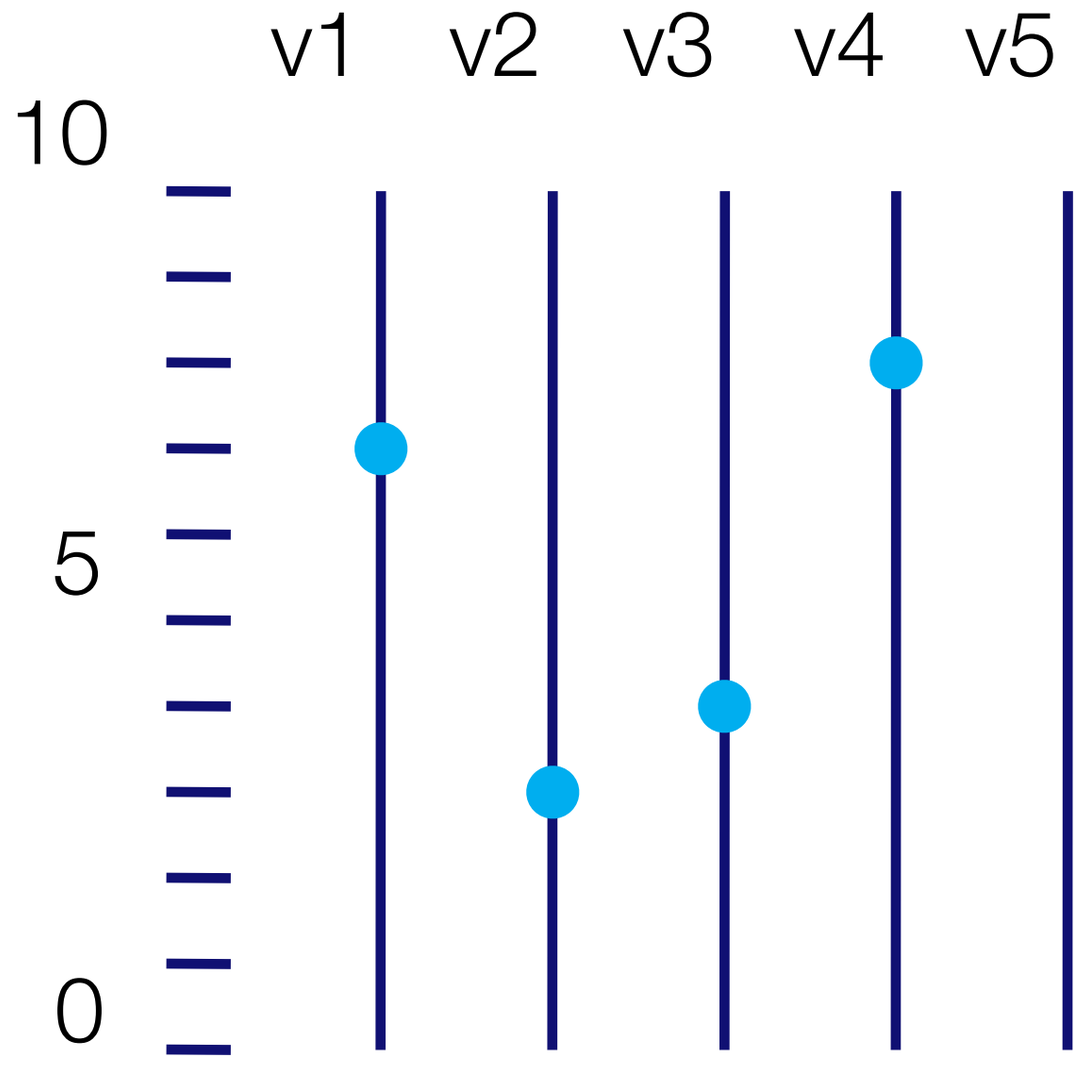
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis



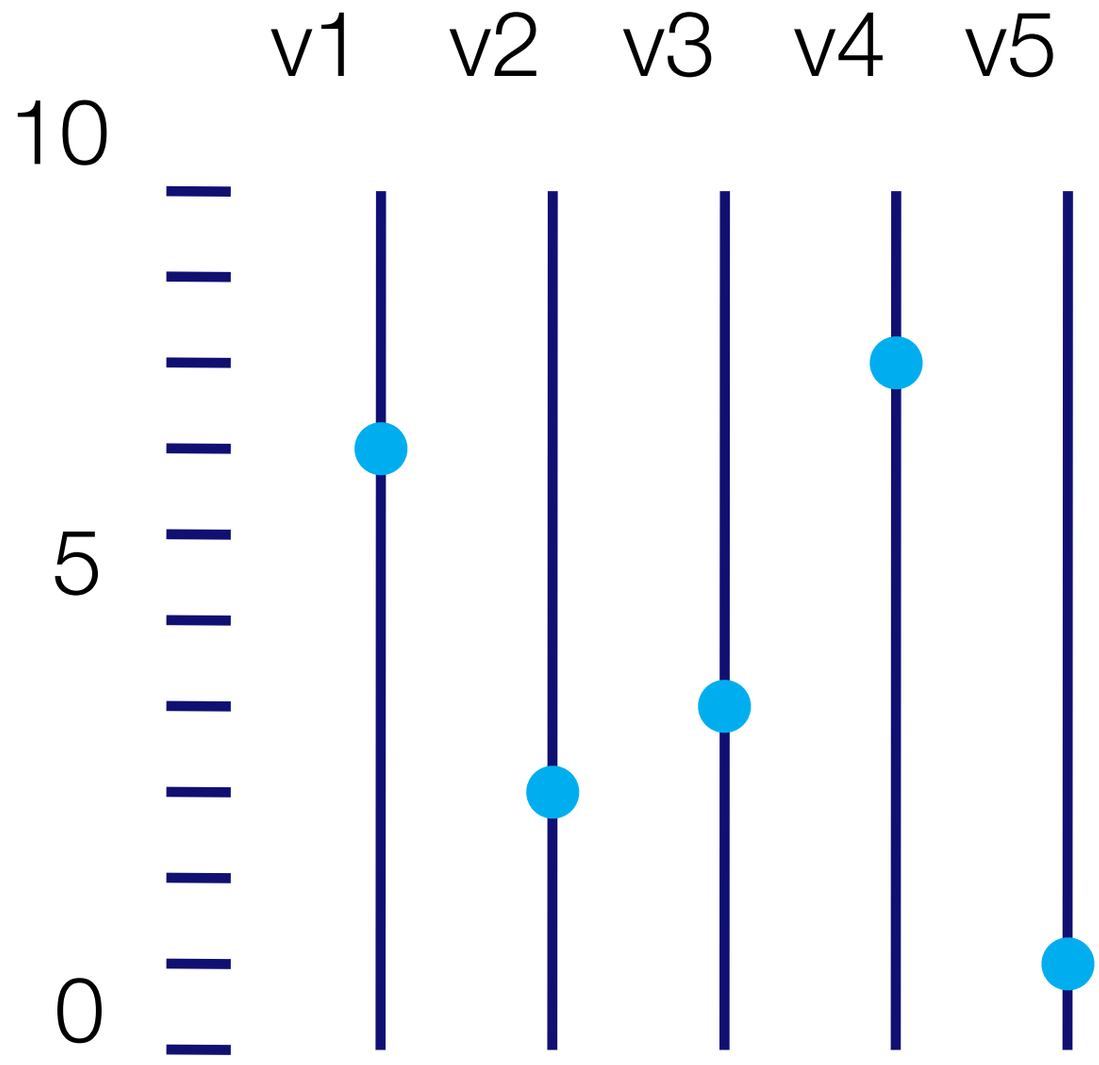
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis



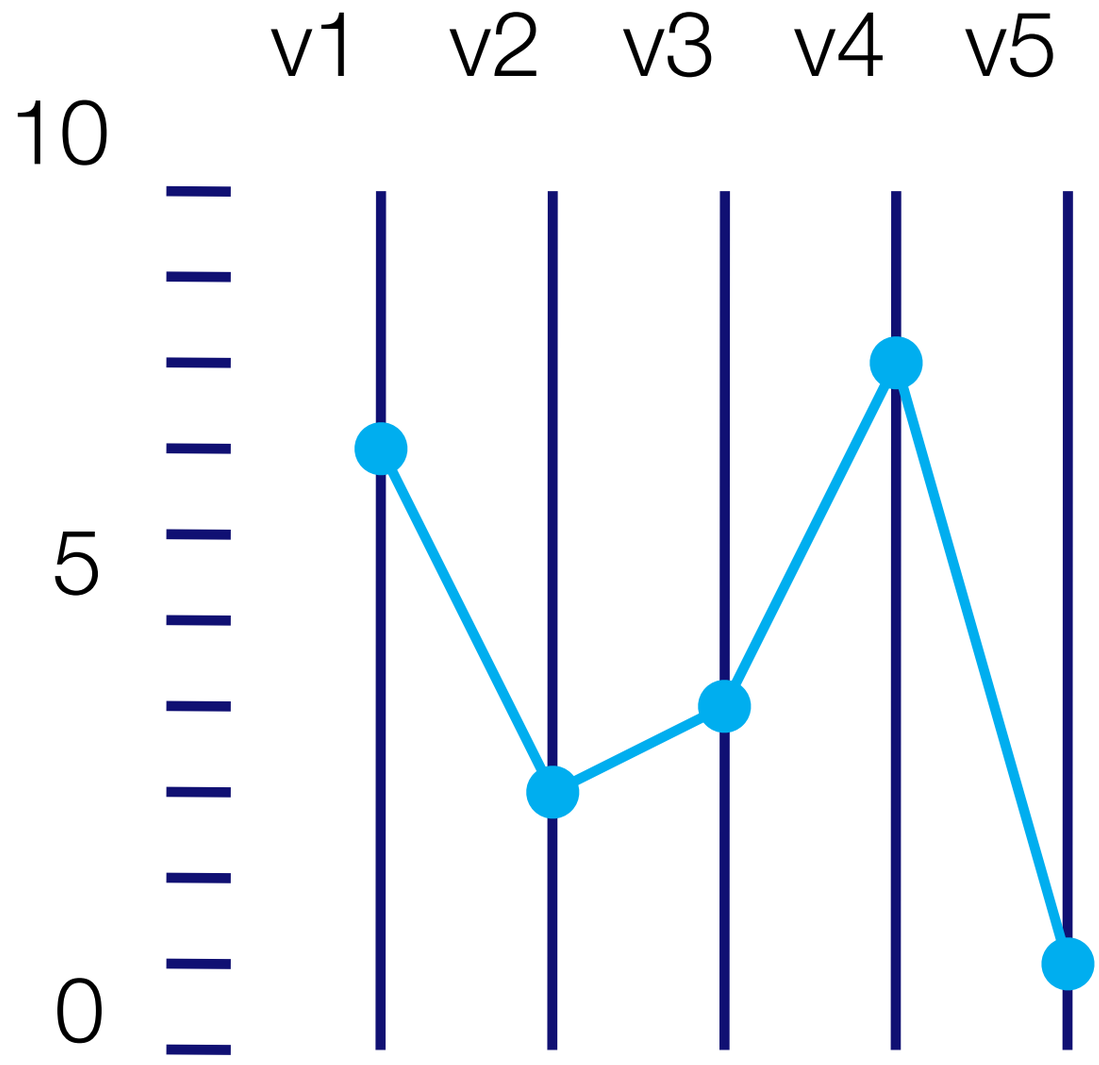
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis
- connect points of one data item by polyline



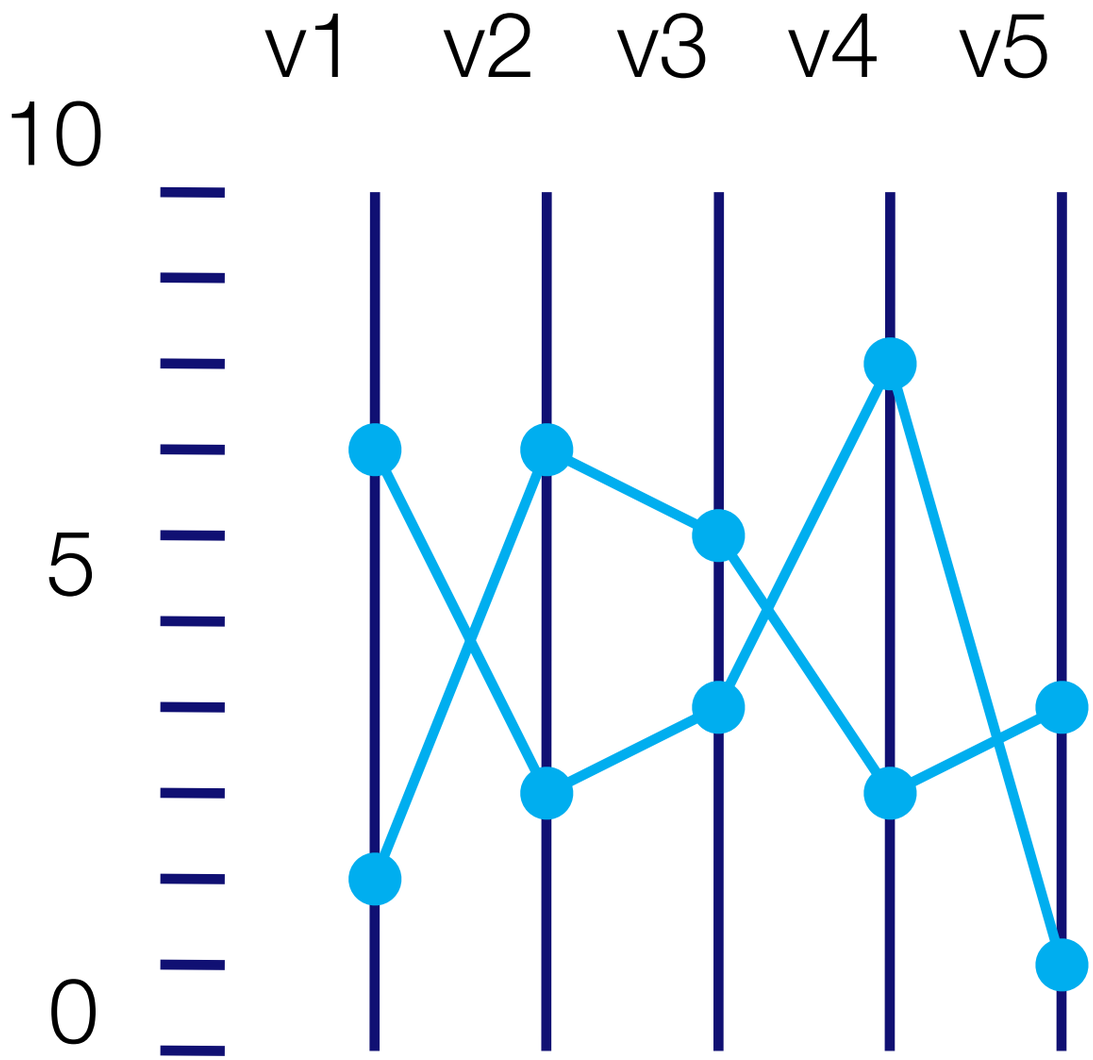
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

- encode variables as parallel axes
- 1-D scatter plot per axis
- connect points of one data item by polyline



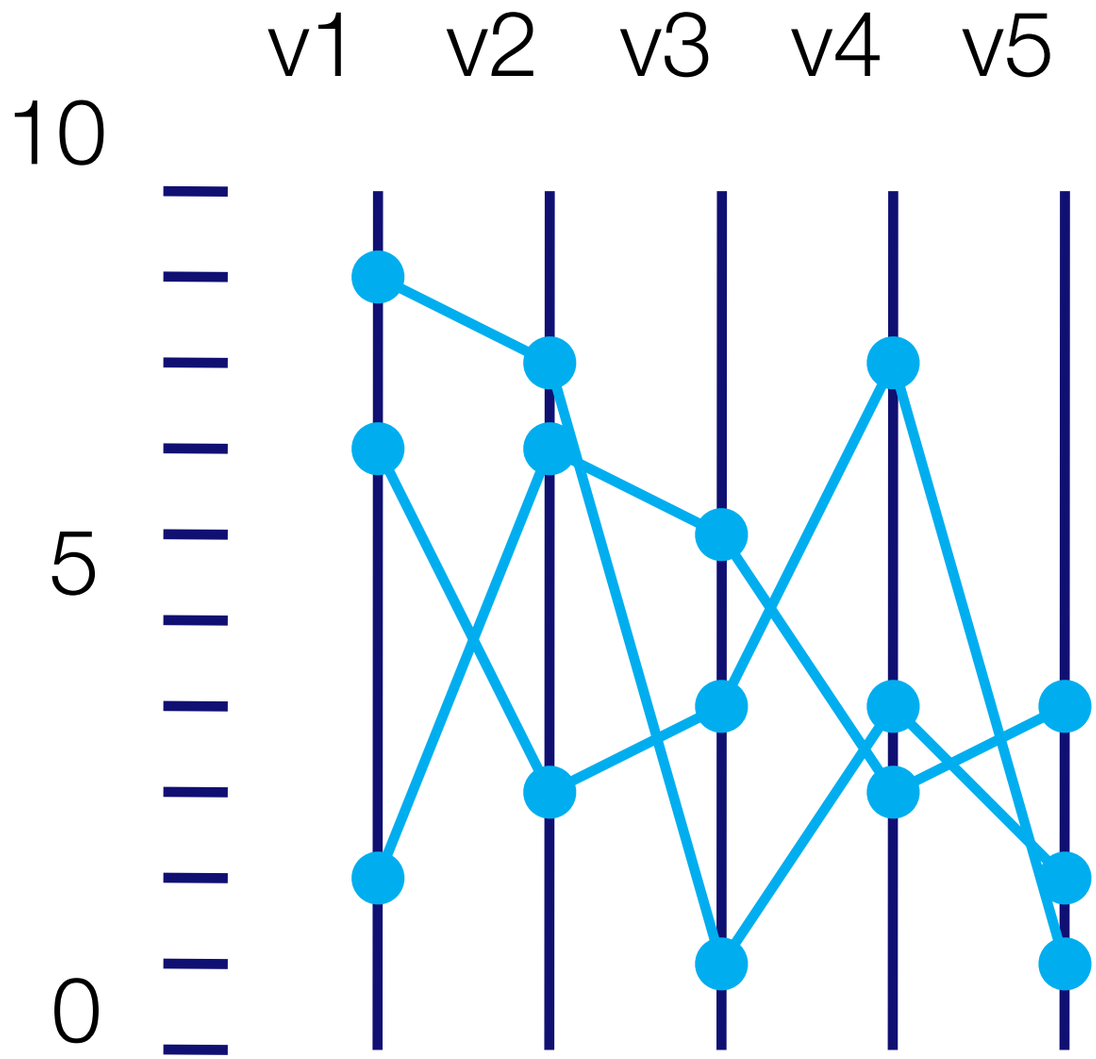
Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

Parallel coordinate plot (PCP)

Construction

	v1	v2	v3	v4	v5
d1	7	3	4	8	1
d2	2	7	6	3	4
d3	9	8	1	4	2

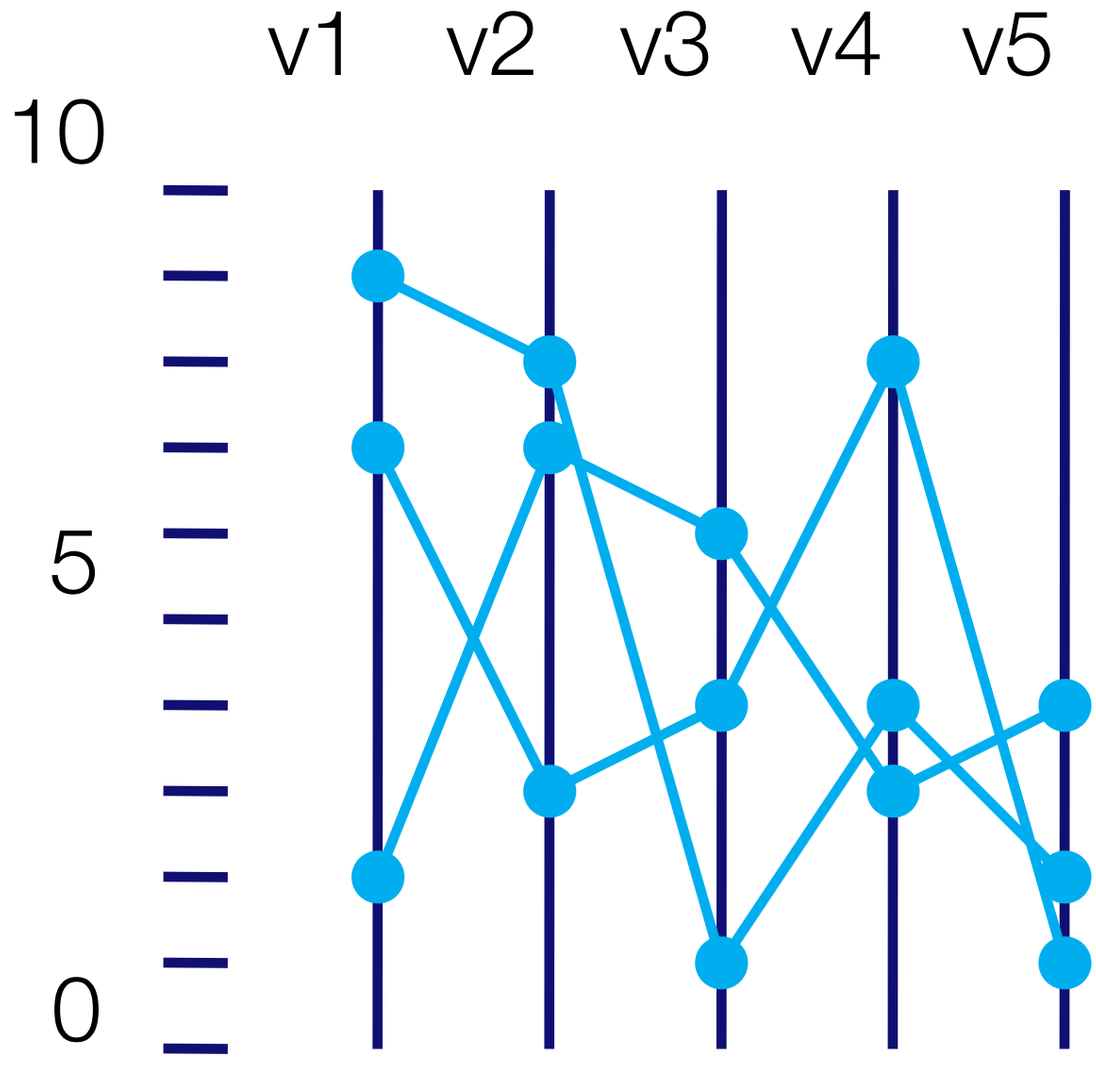
- encode variables as parallel axes
- 1-D scatter plot per axis
- connect points of one data item by polyline



Inselberg, The plane with parallel coordinates. The Visual Computer 1(3):69-91, 1985

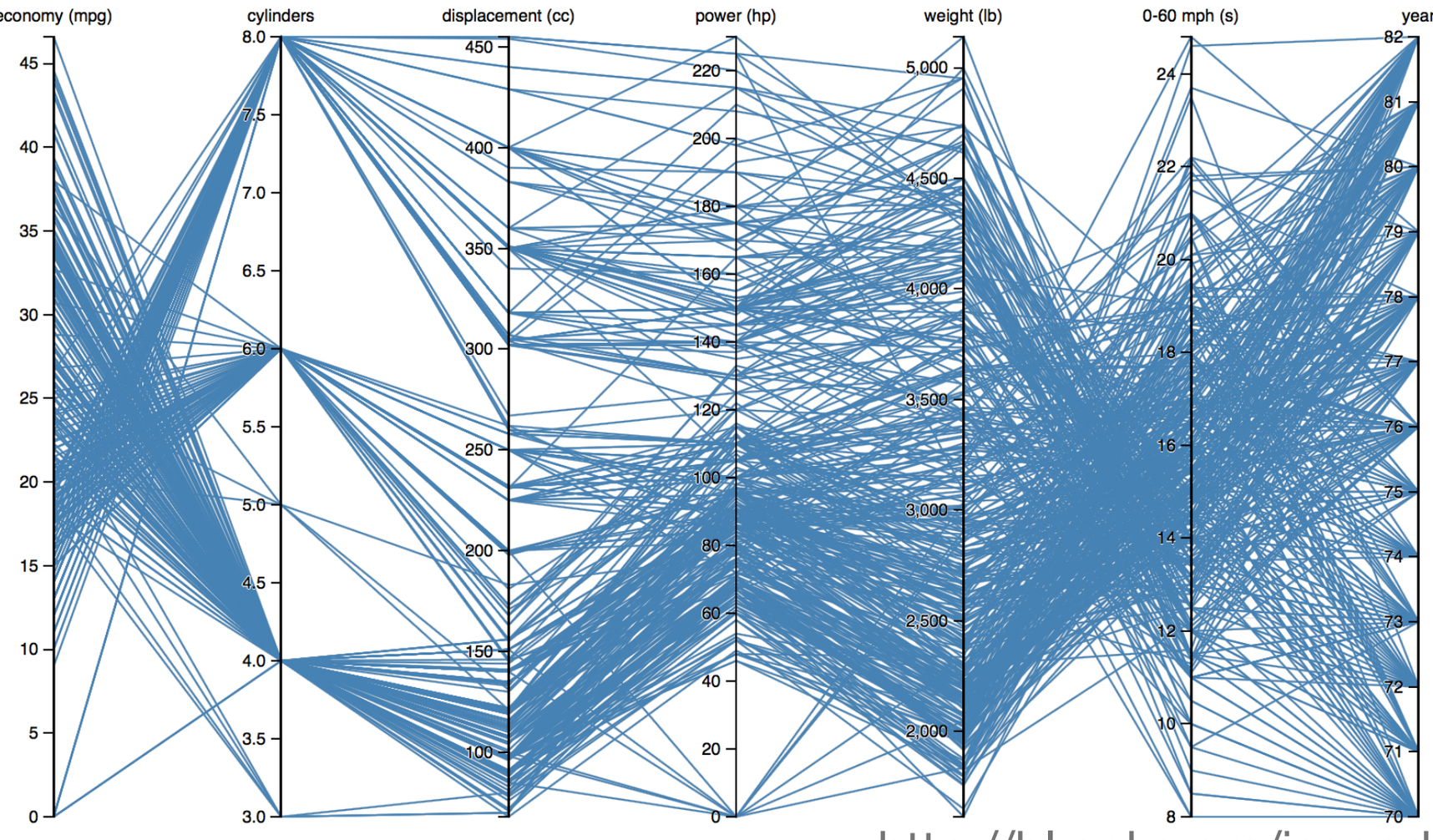
PCP

- What
  - many value attributes
- How
  - parallel layout: horizontal position for axes; vertical position for value; connection lines between points of same item
- Task
  - find trends, outliers, correlation, extremes
- Scale
  - hundreds of items
  - about a dozen attributes



PCP interpretation

- Look for
  - clusters of similar lines (correlation)
  - similar crossing points (negative correlation)
  - isolated lines or lines with different slopes than neighbors (outliers)

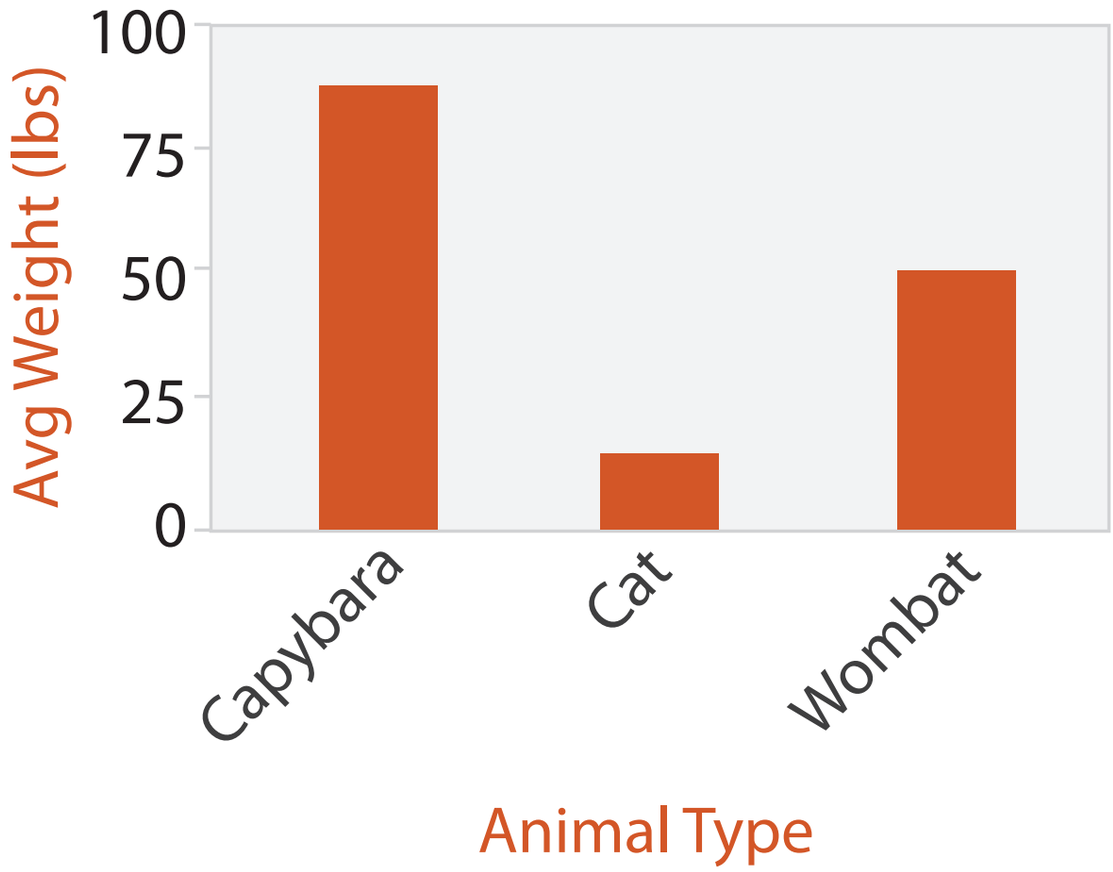


http://bl.ocks.org/jasondavies/1341281

Region-based techniques

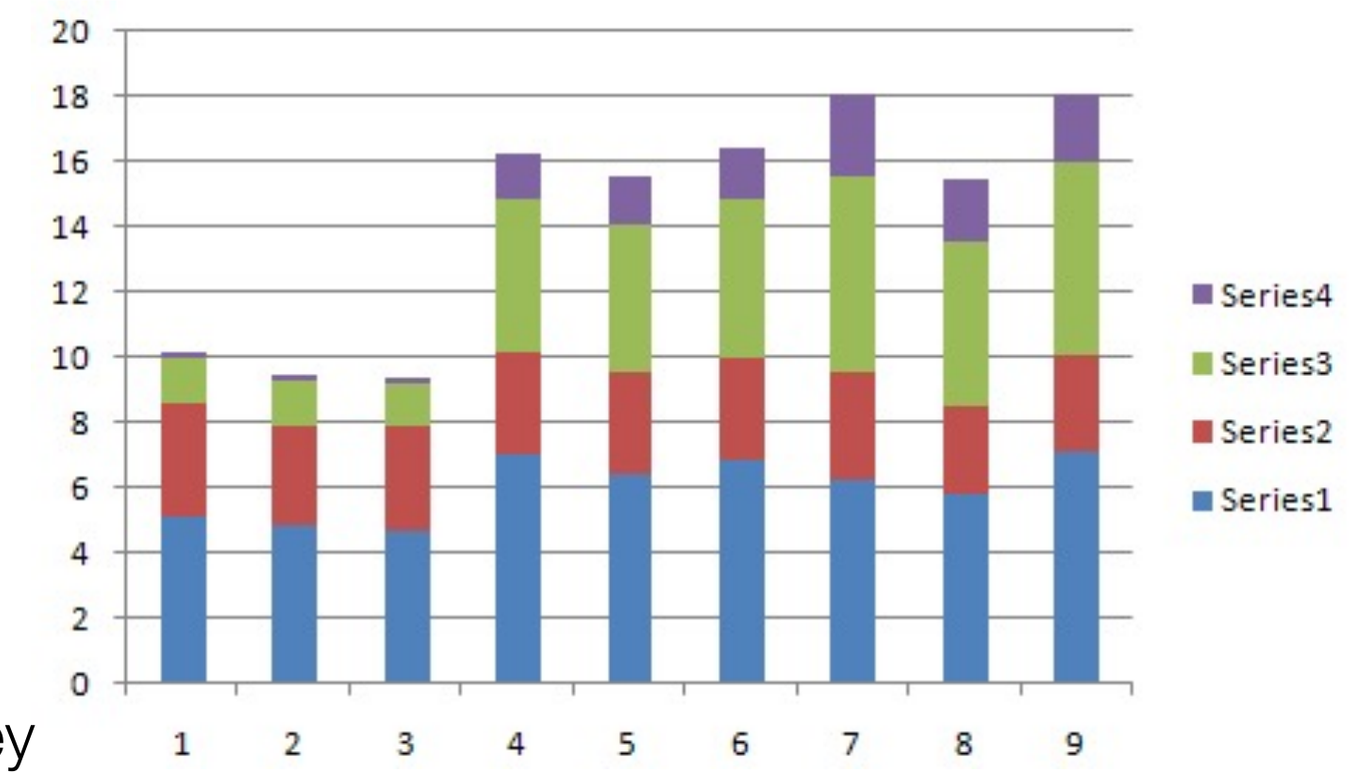
Bar chart

- What
  - one quantitative value attribute
  - one categorical key attribute
- How
  - rectangular shape
  - value as length
  - key as horizontal position
- Task
  - lookup and compare
- Scale
  - dozens to hundreds of levels for key attribute



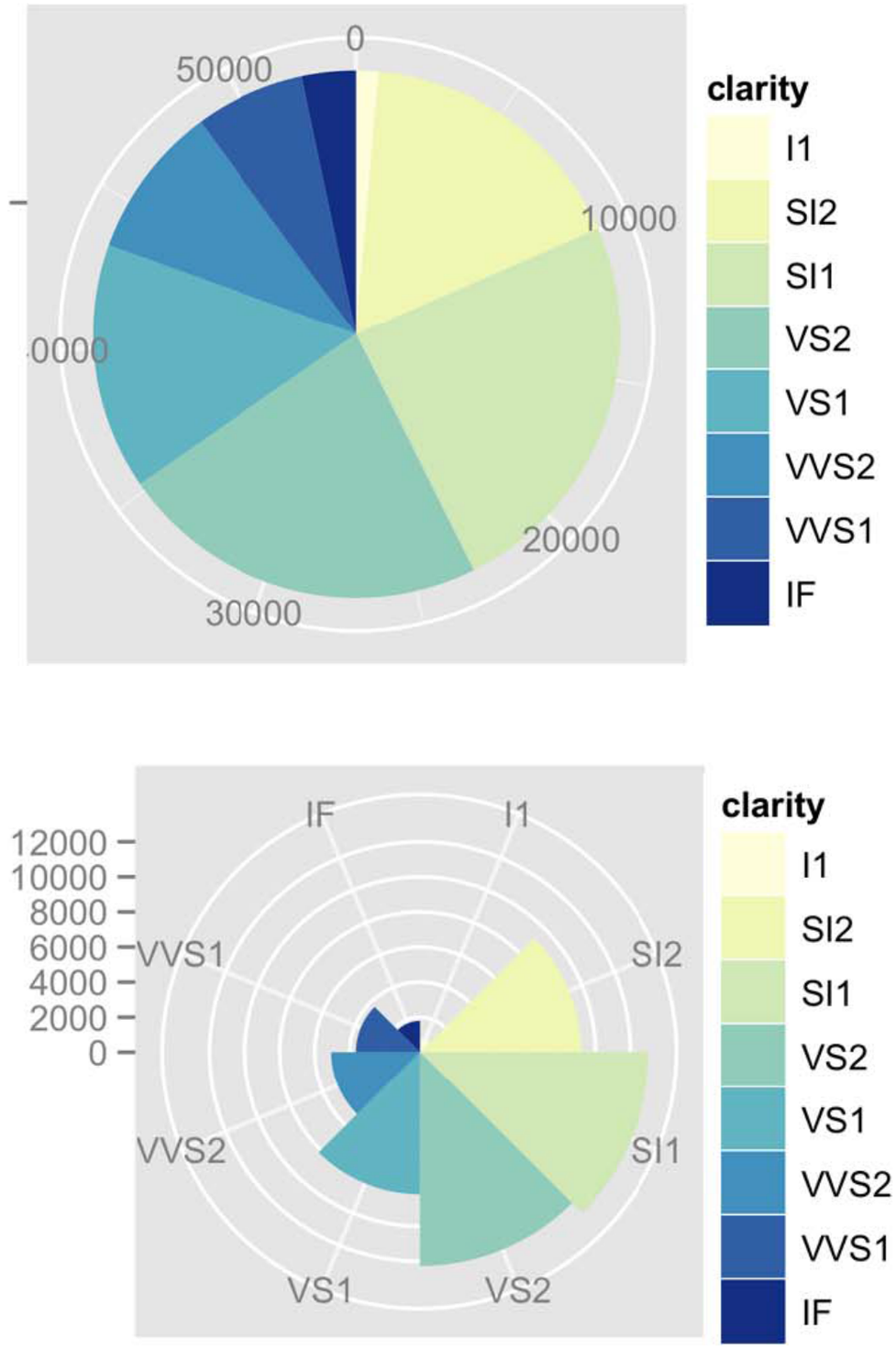
Stacked bar chart

- What
  - one quantitative attribute value
  - two categorical keys
- How
  - rectangle with length-encoded subcomponents of value attribute for each category of secondary key
- Task
  - part-to-whole relationship
- Scale
  - dozens to hundreds of levels for key attribute on main axis
  - several to one dozen for key attribute along stacked axis

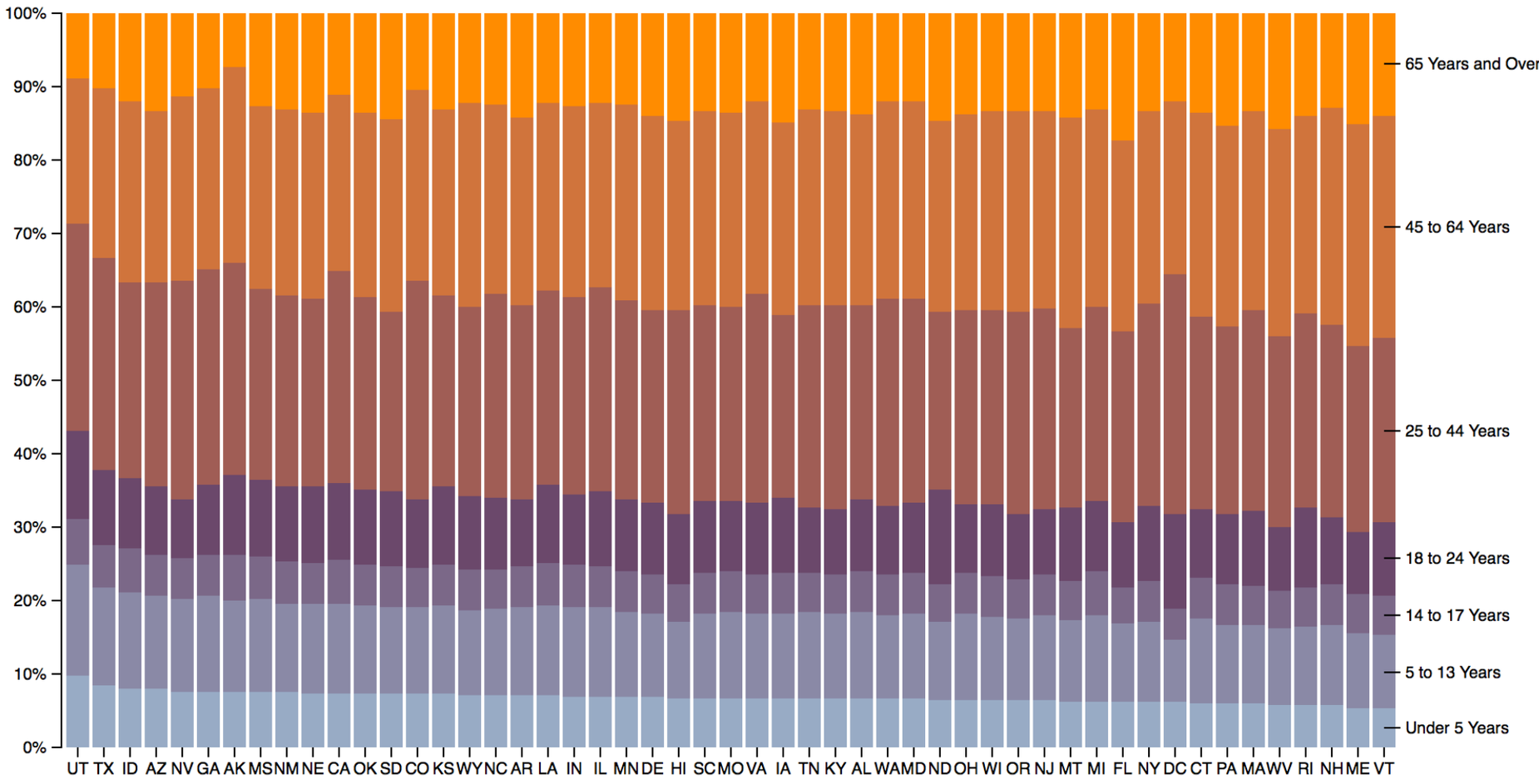


Pie chart, polar area chart

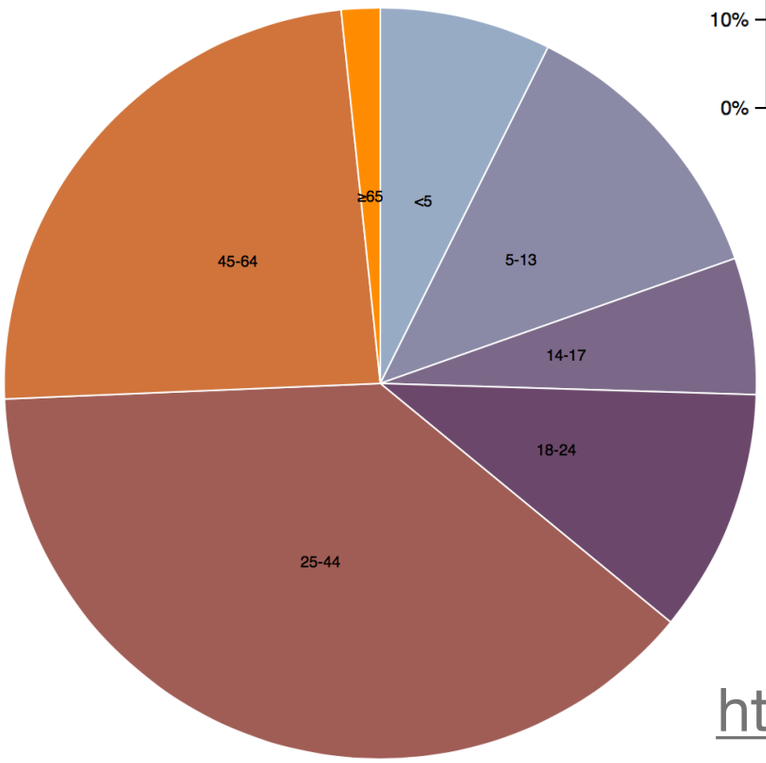
- What
  - one quantitative attribute
  - one categorical attribute
- How
  - Pie chart
    - area shapes with **angle** channel
  - Polar area chart
    - area shapes with **length** channel
- Task
  - part-whole relationship
- Scale
  - dozen categories



Pie chart vs. normalized bar chart



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

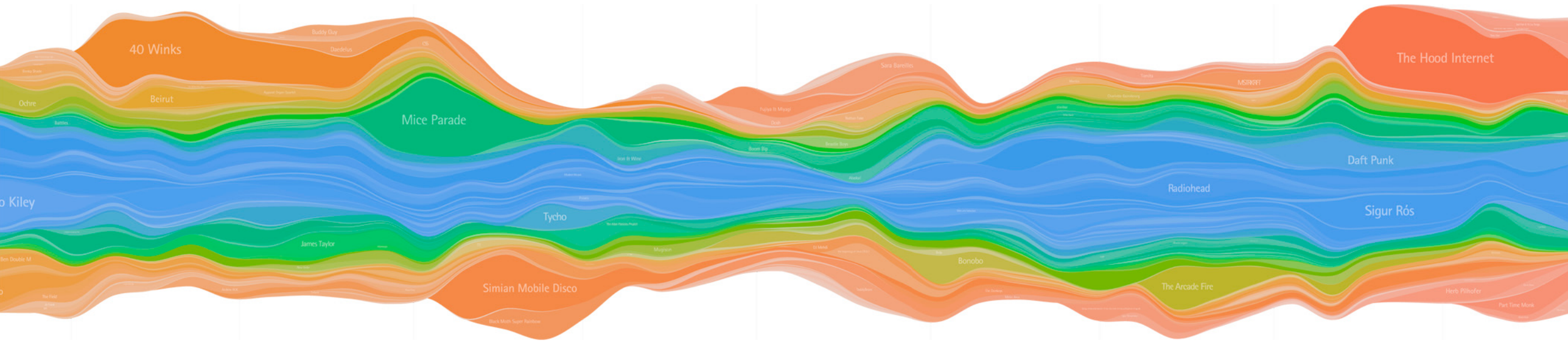


http://bl.ocks.org/mbostock/3887235

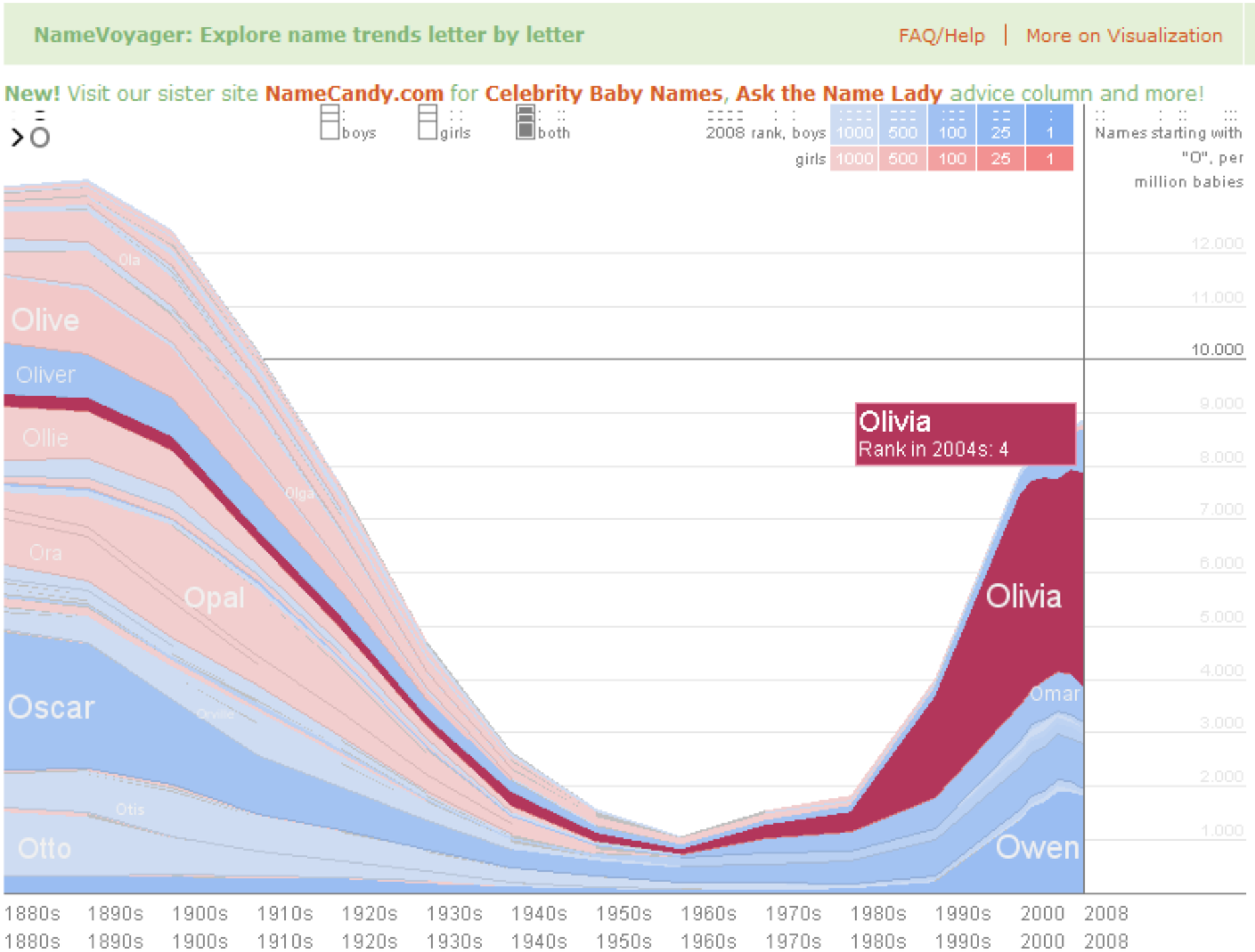
Streamgraphs

- What
  - one quantitative value attribute
  - one ordered key attribute (time)
  - one categorical key attribute
- How
  - derived geometry showing layers
  - height encodes count
  - derived attribute for layer ordering
- Scale
  - hundreds of time keys
  - dozens to hundreds of categorical key values

Byron & Wattenberg. Stacked graphs geometry & aesthetics. IEEE Trans. Visualization and Computer Graphics 14(6):1245-1252, 2008.



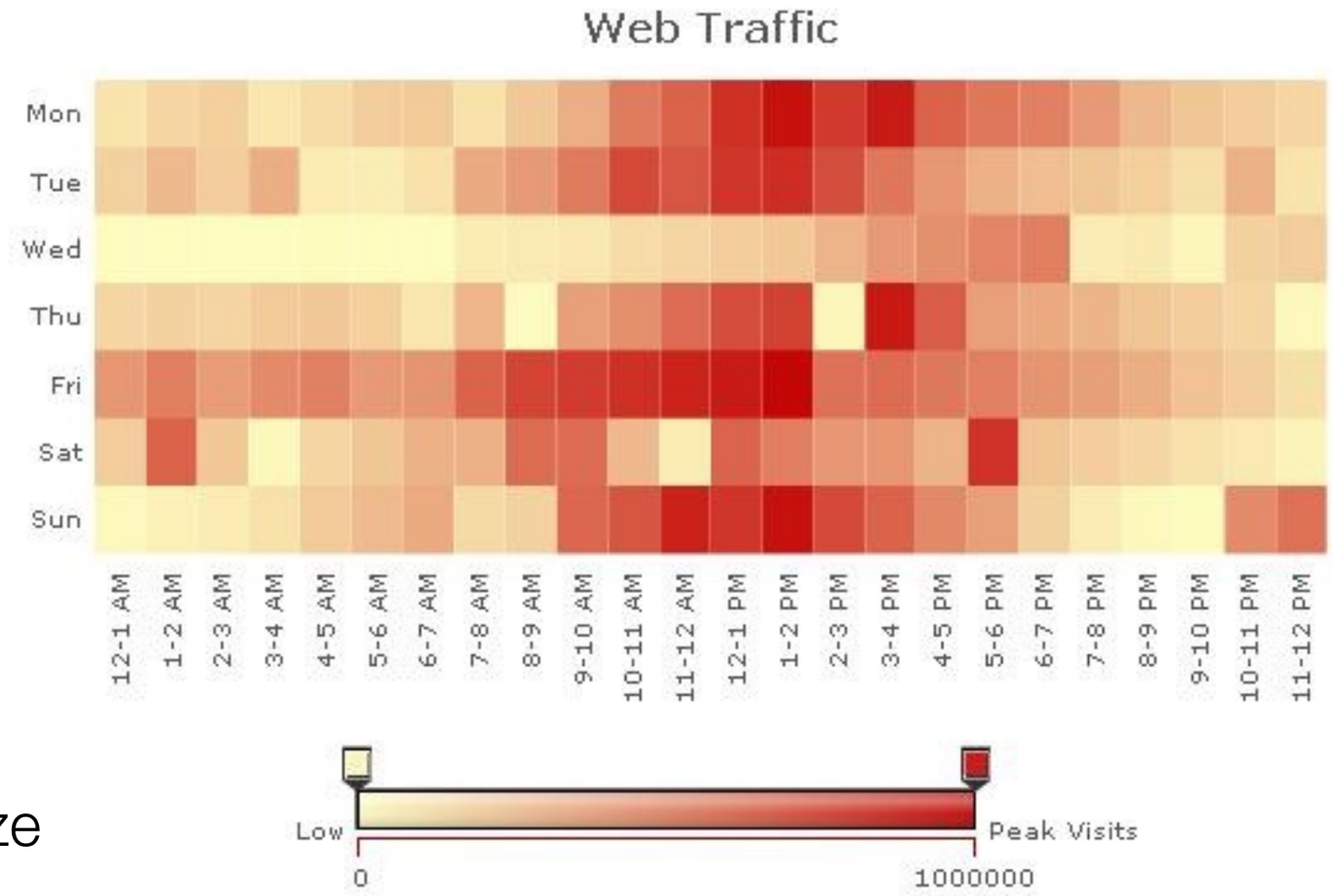
Interactive example: NameVoyager



http://www.babynamewizard.com/voyager

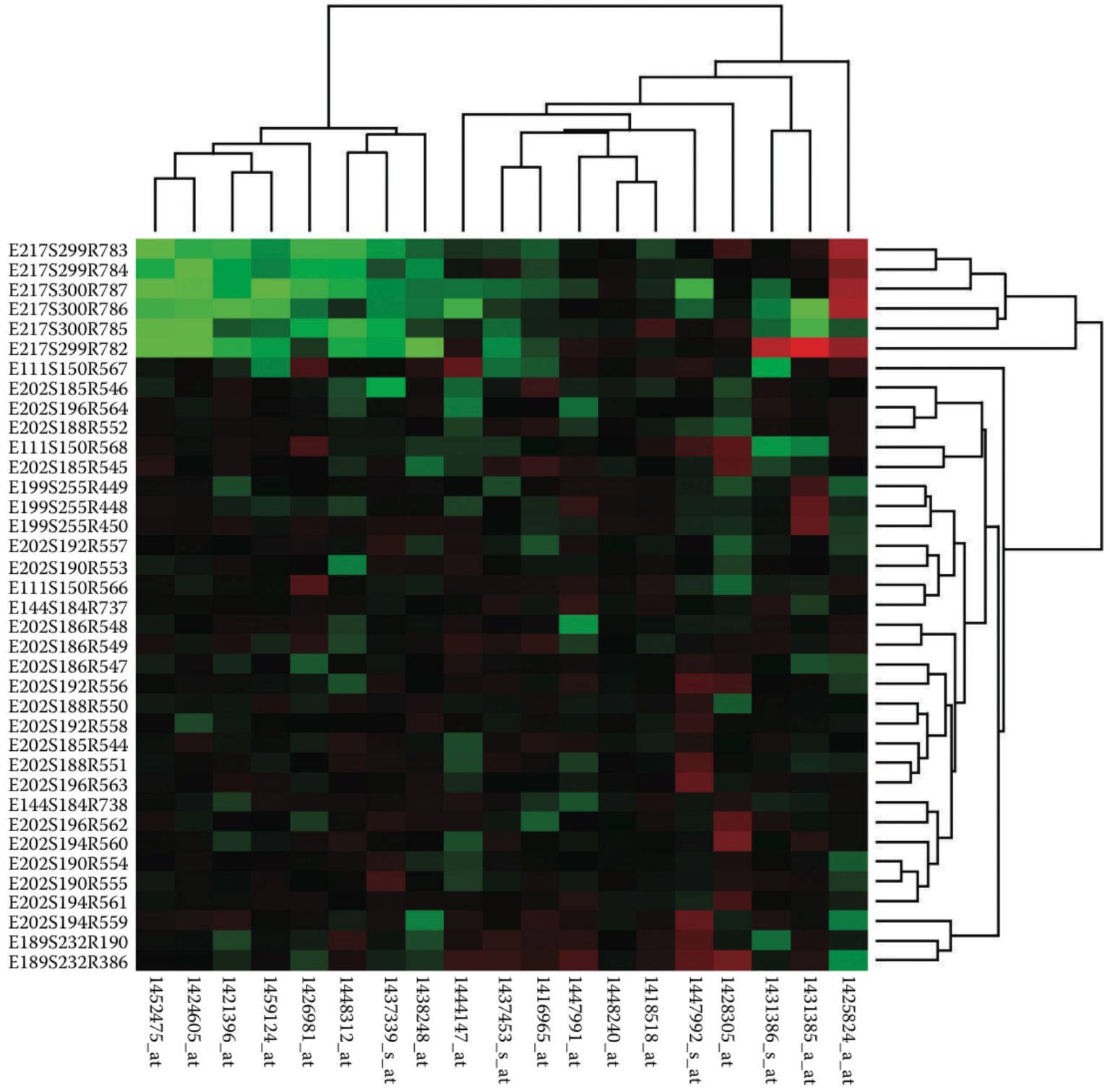
Heatmap

- What
  - two categorical key attributes
  - one quantitative value attribute
- How
  - 2D matrix arrangement of colored squares
- Task
  - find clusters, outliers; summarize
- Scale
  - 1M items
  - hundreds of categorical levels
  - quantitative levels: 3-11

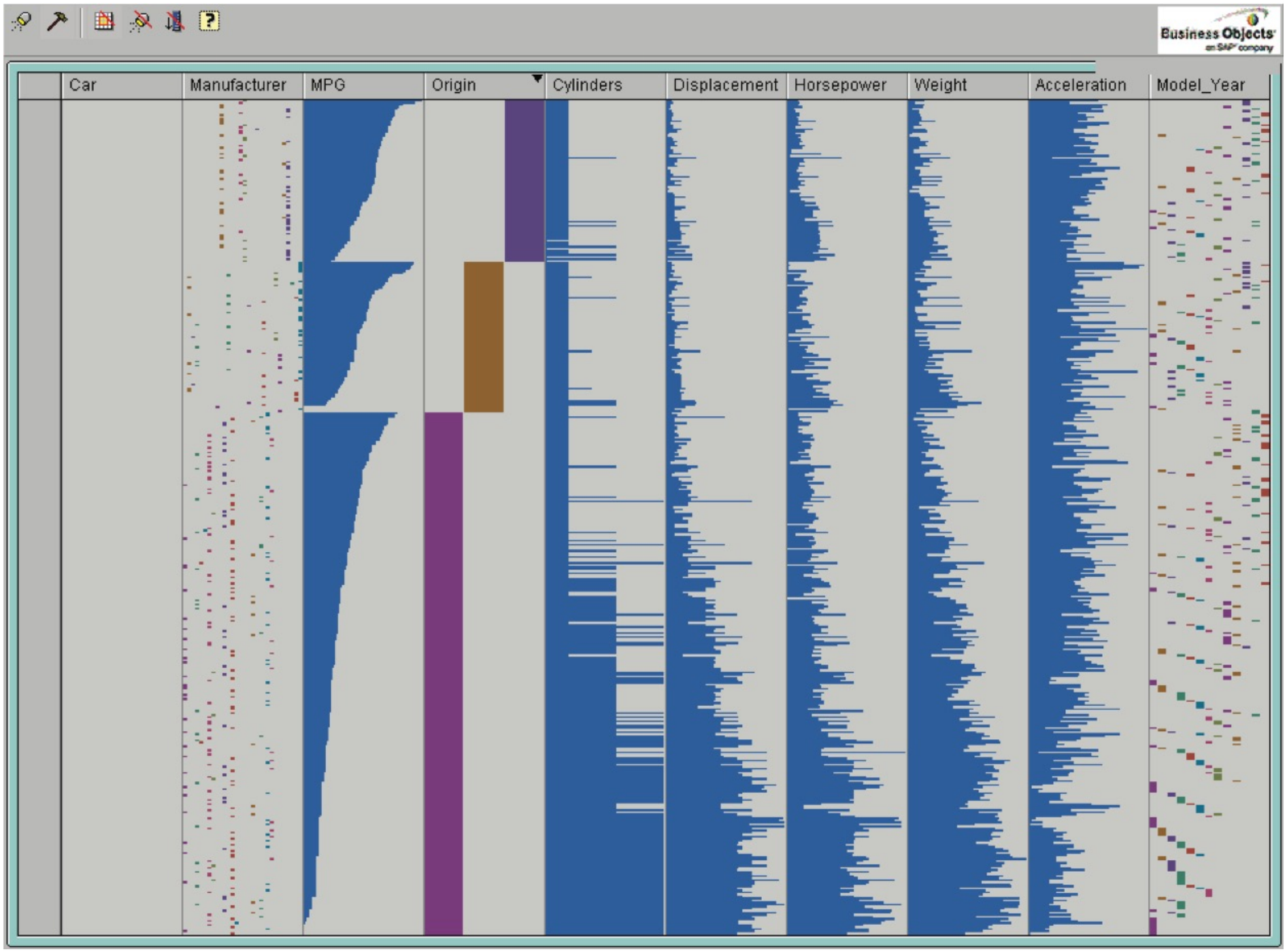


Clustered heatmap

- In addition to standard heatmap
  - derive two cluster hierarchies
  - order rows and columns by hierarchy

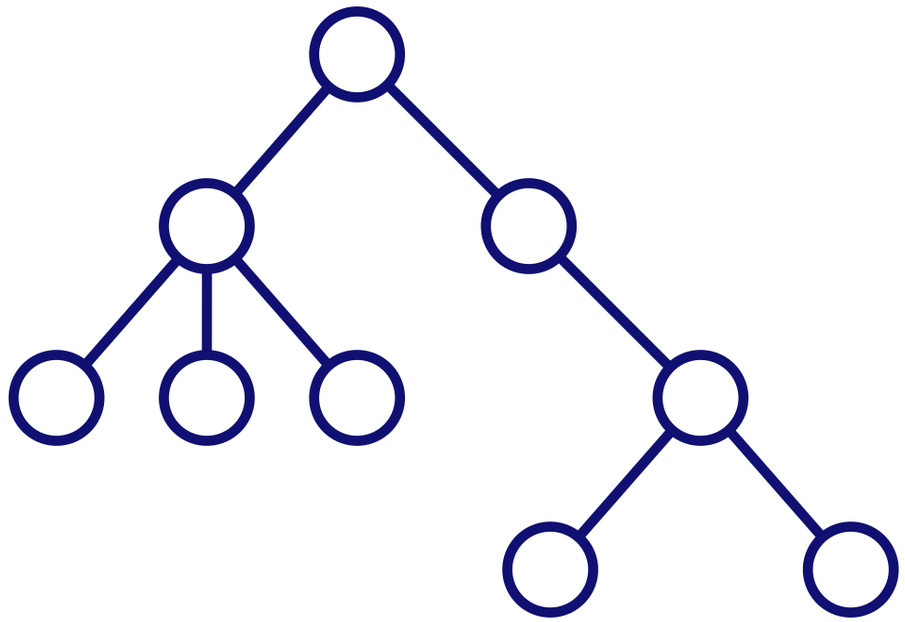


Tabular display: Table Lens



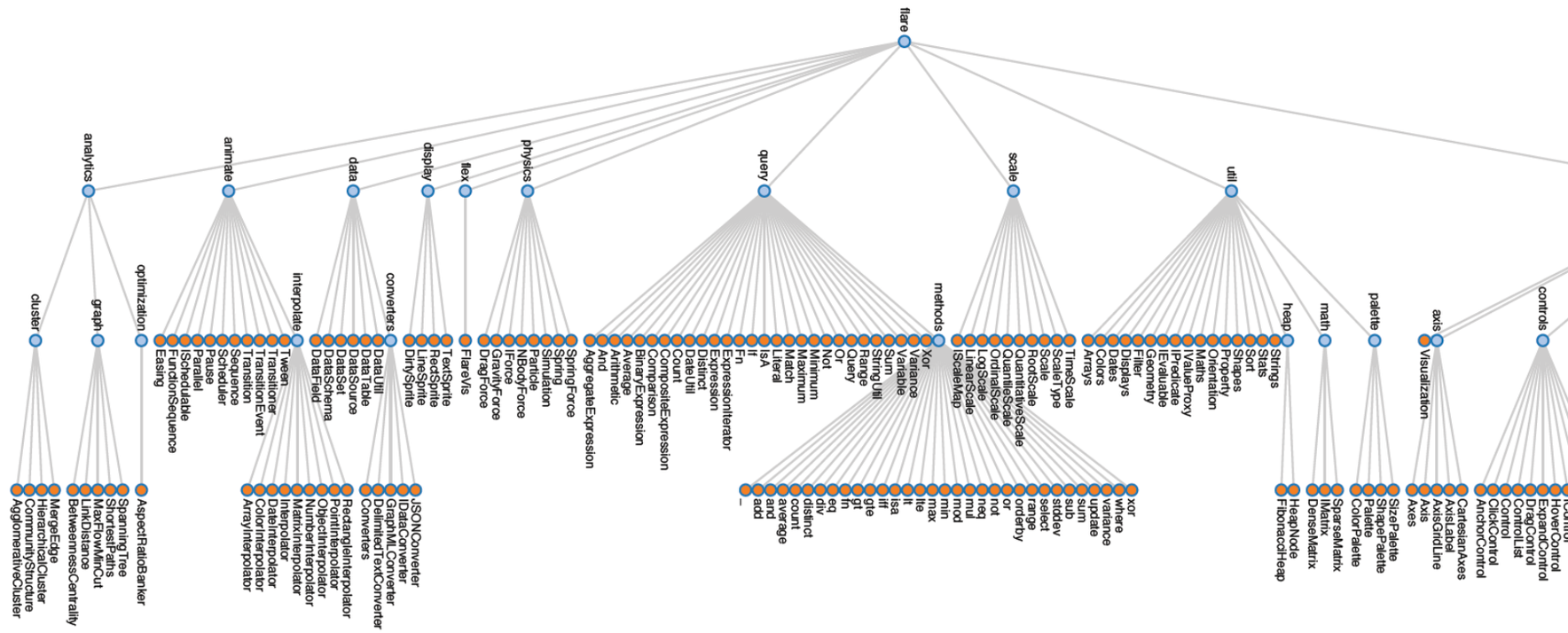
Rao et al., The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. Proc. SIGCHI Conf. Human Factors in Computing Systems: Celebrating Interdependence, pp. 318-322, 1994 (https://youtu.be/ZDY9YCYv7z8)

Trees



Node-link tree

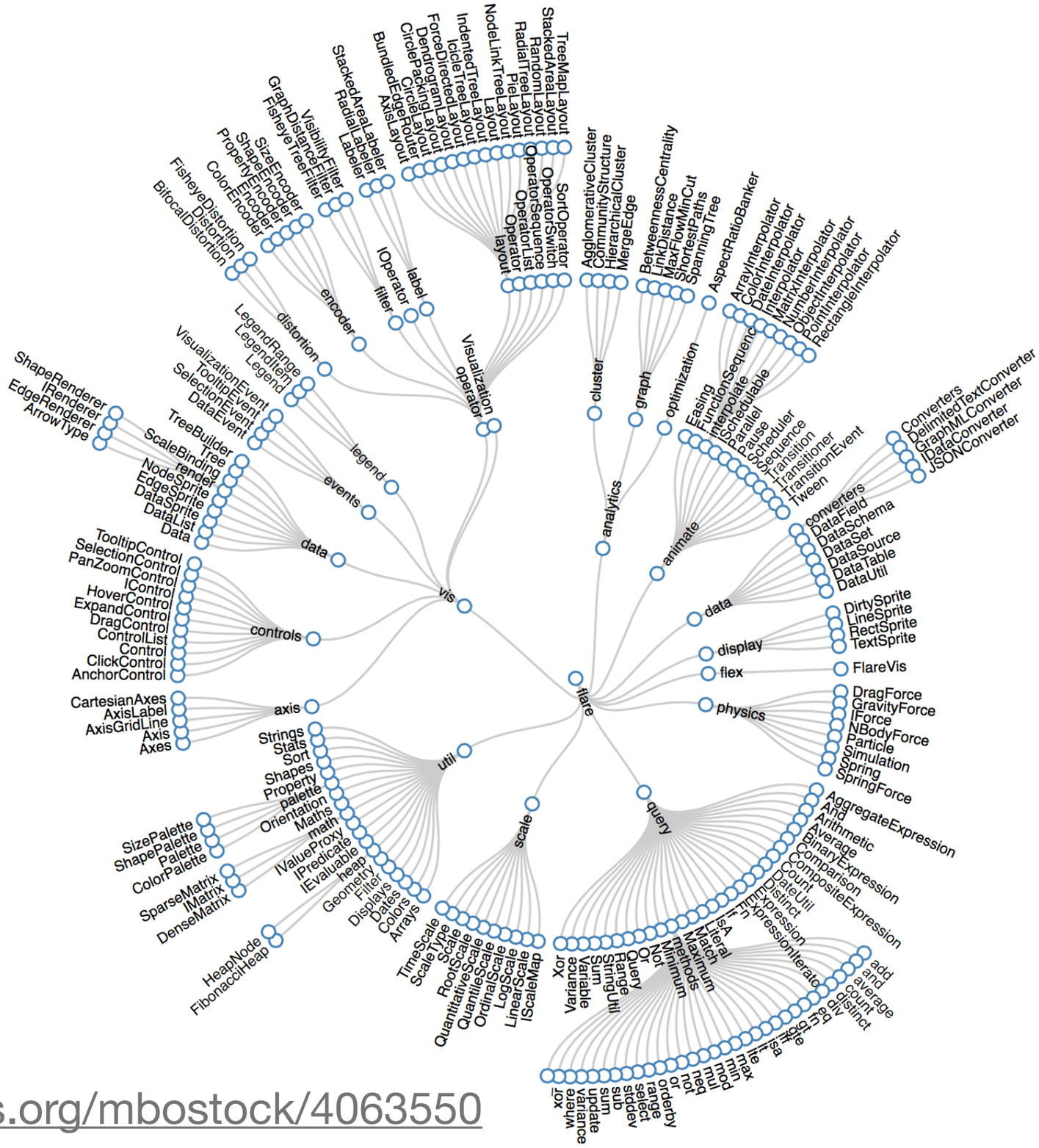
- What
  - tree
- How
  - point nodes
  - link connections
  - root at top
- Task
  - understand topology, follow paths
- Scale
  - 1000-10000 nodes



Flare package hierarchy (an infovis toolkit: http://flare.prefuse.org)

Radial node-link tree

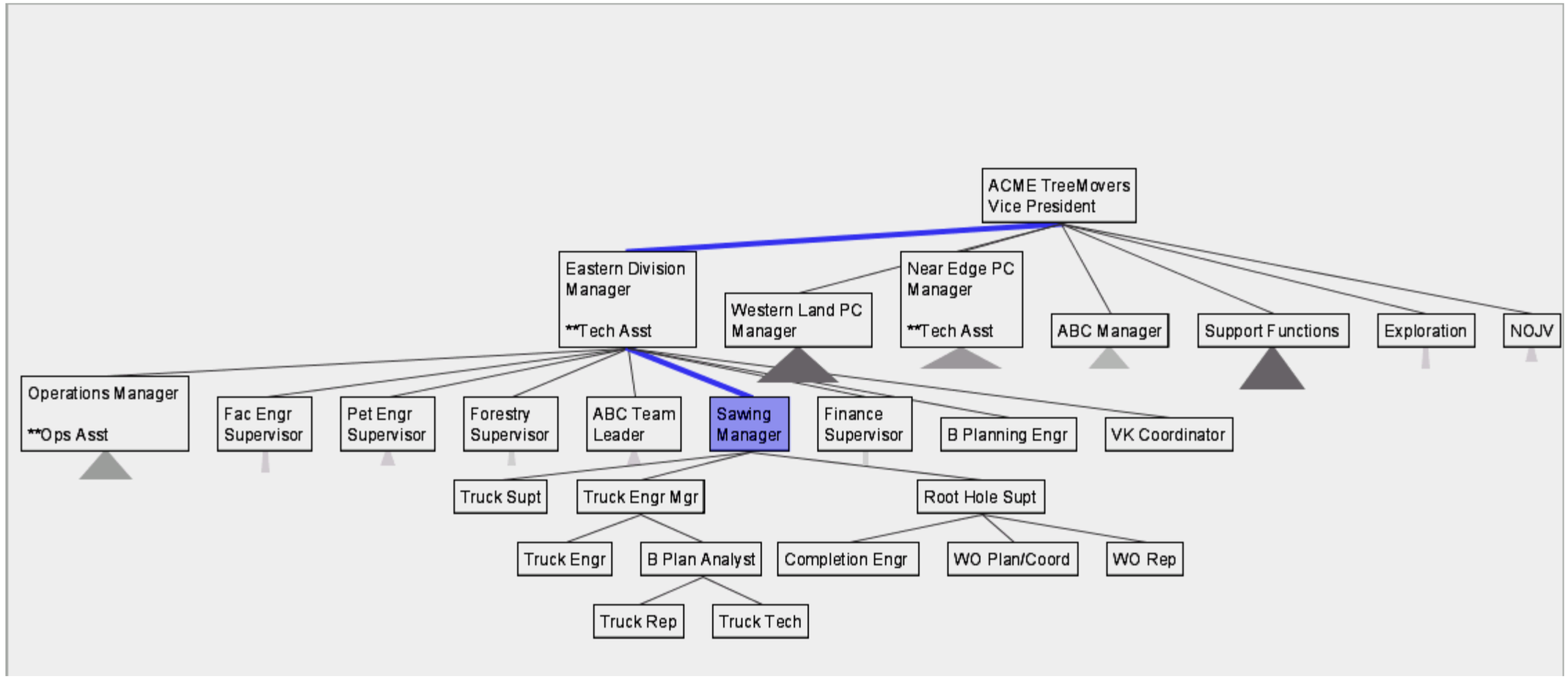
- What
  - tree
- How
  - point nodes
  - link connections
  - radial layout
- Task
  - understand topology
    follow paths
- Scale
  - 1000-10000 nodes



http://bl.ocks.org/mbostock/4063550

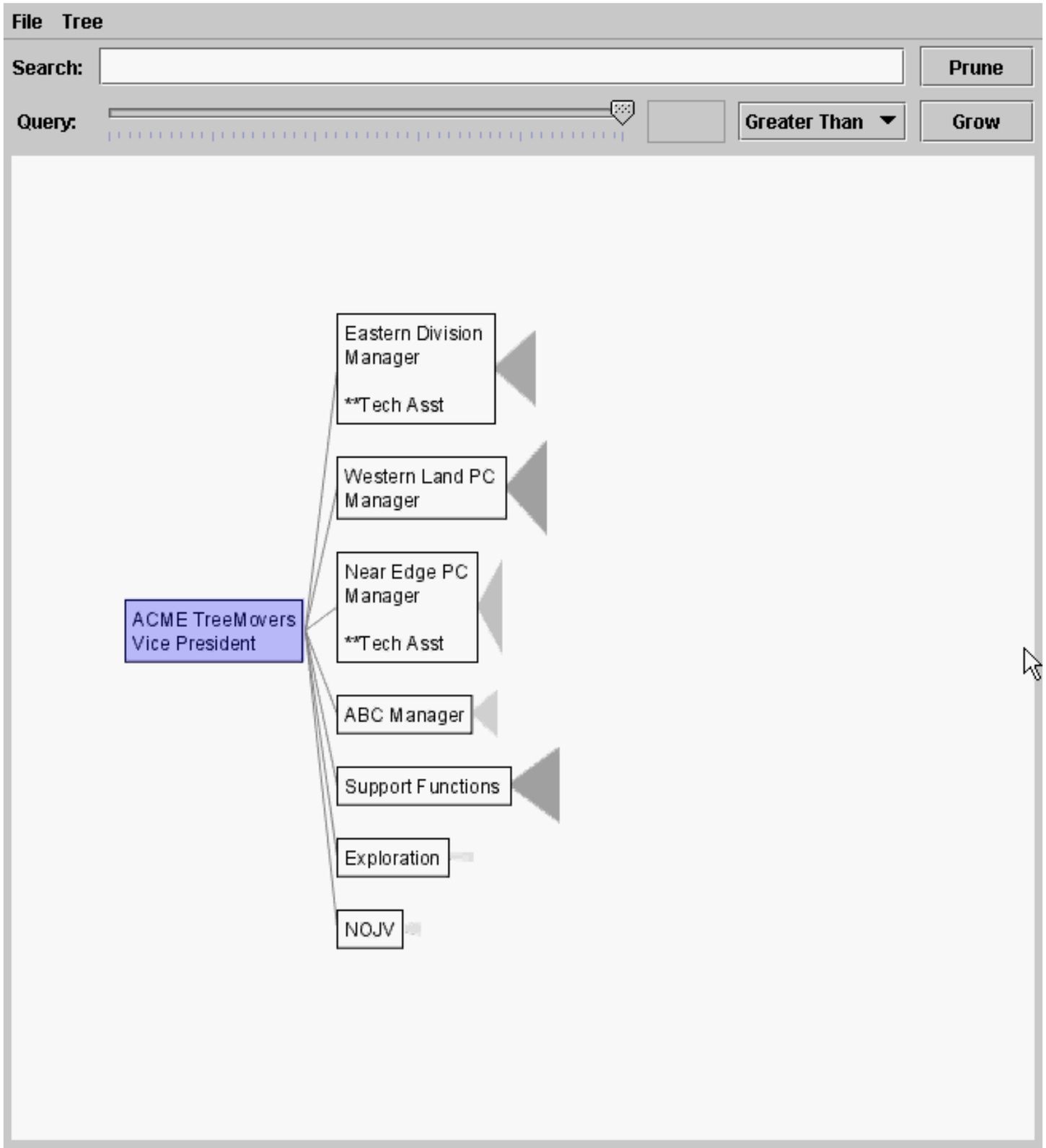
Space Tree

- Conventional 2-D layout + animation + collapse/uncollapse
- Subtrees are triangles (preview icons)
  - height/width corresponds to depth/average width subtree
  - shading corresponds to number of nodes inside

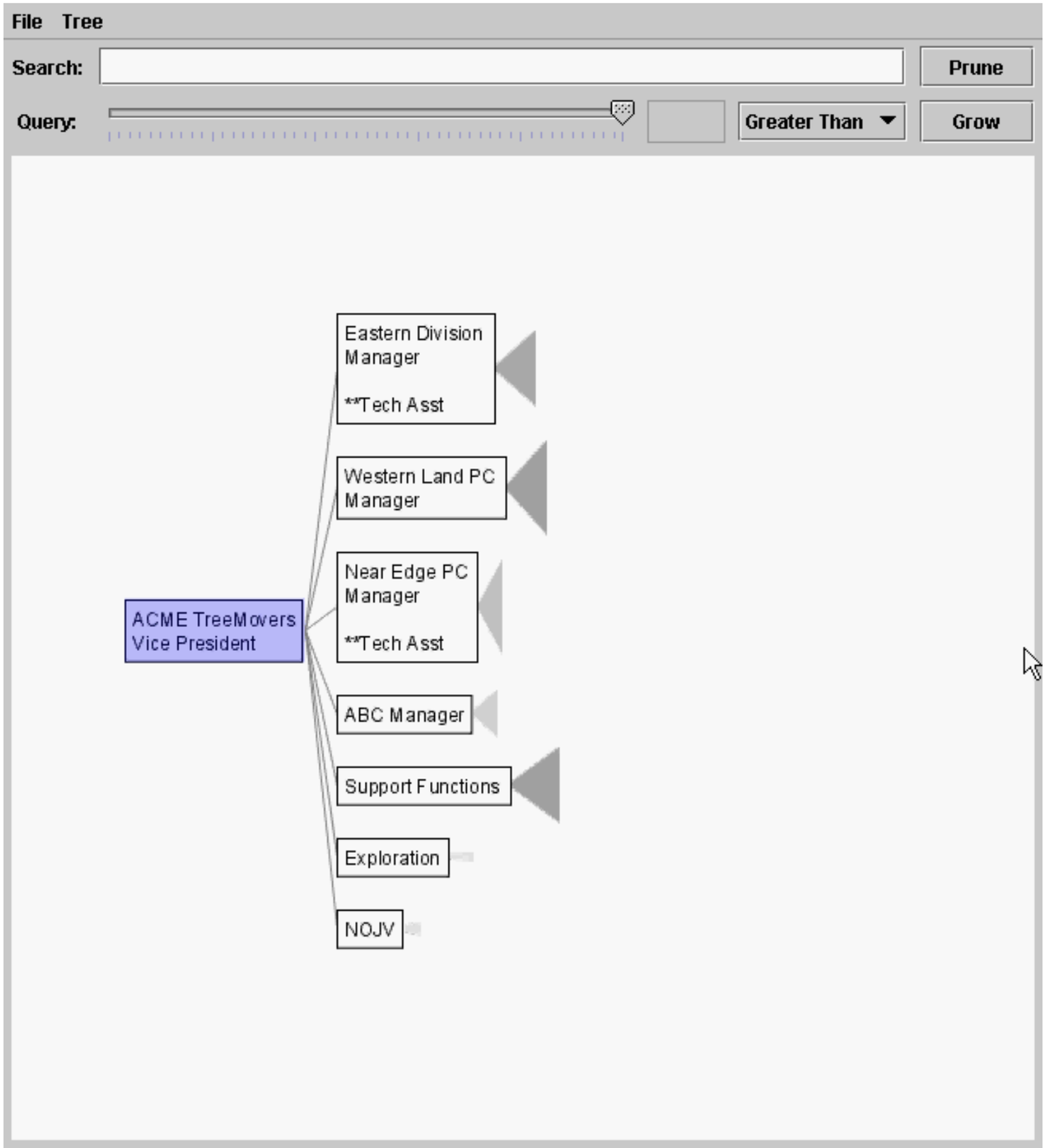


Plaisant et al., SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation. Proc. IEEE Symp. Information Visualization, pp. 57-64, 2002

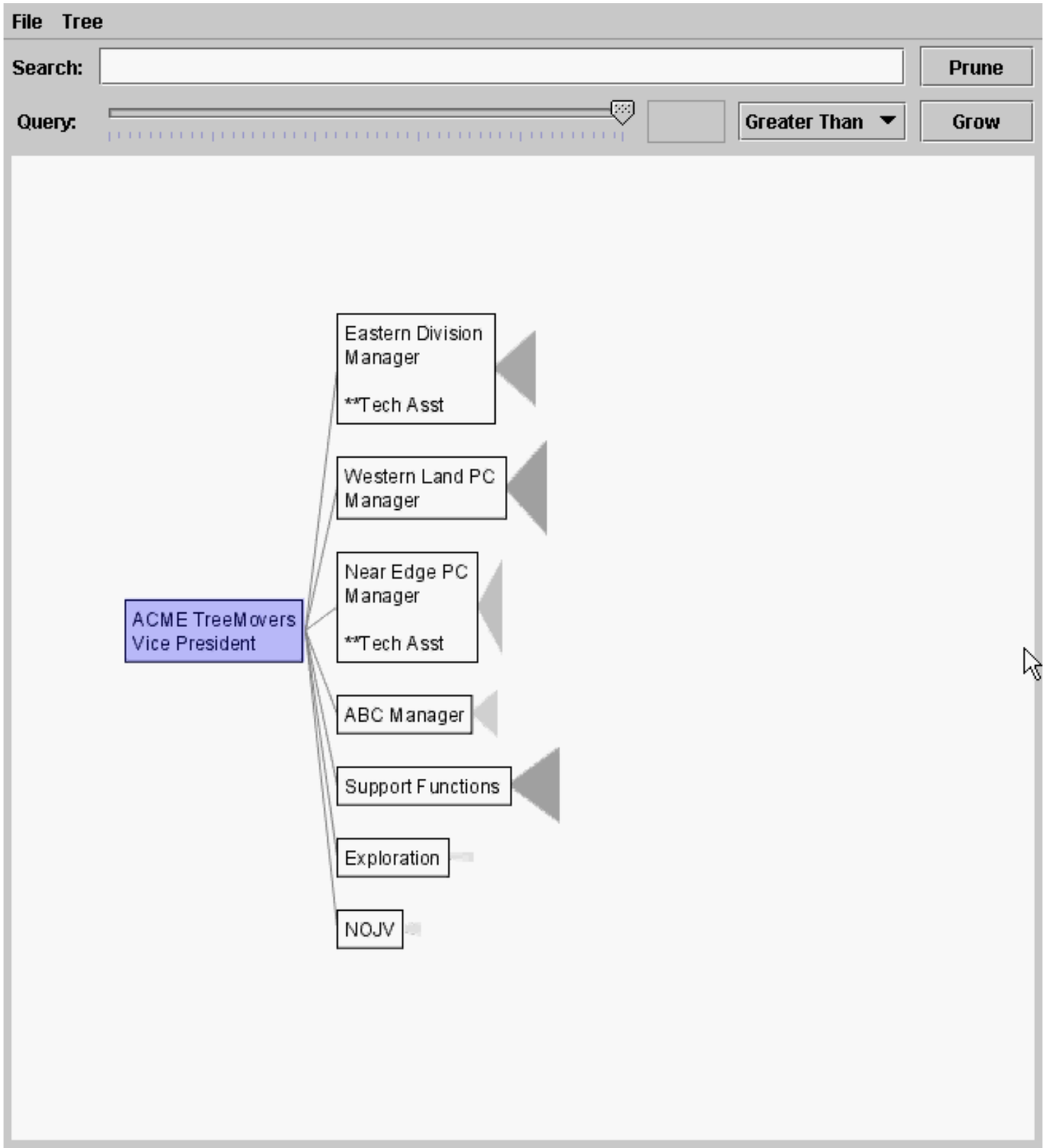
Space Tree example



Space Tree example

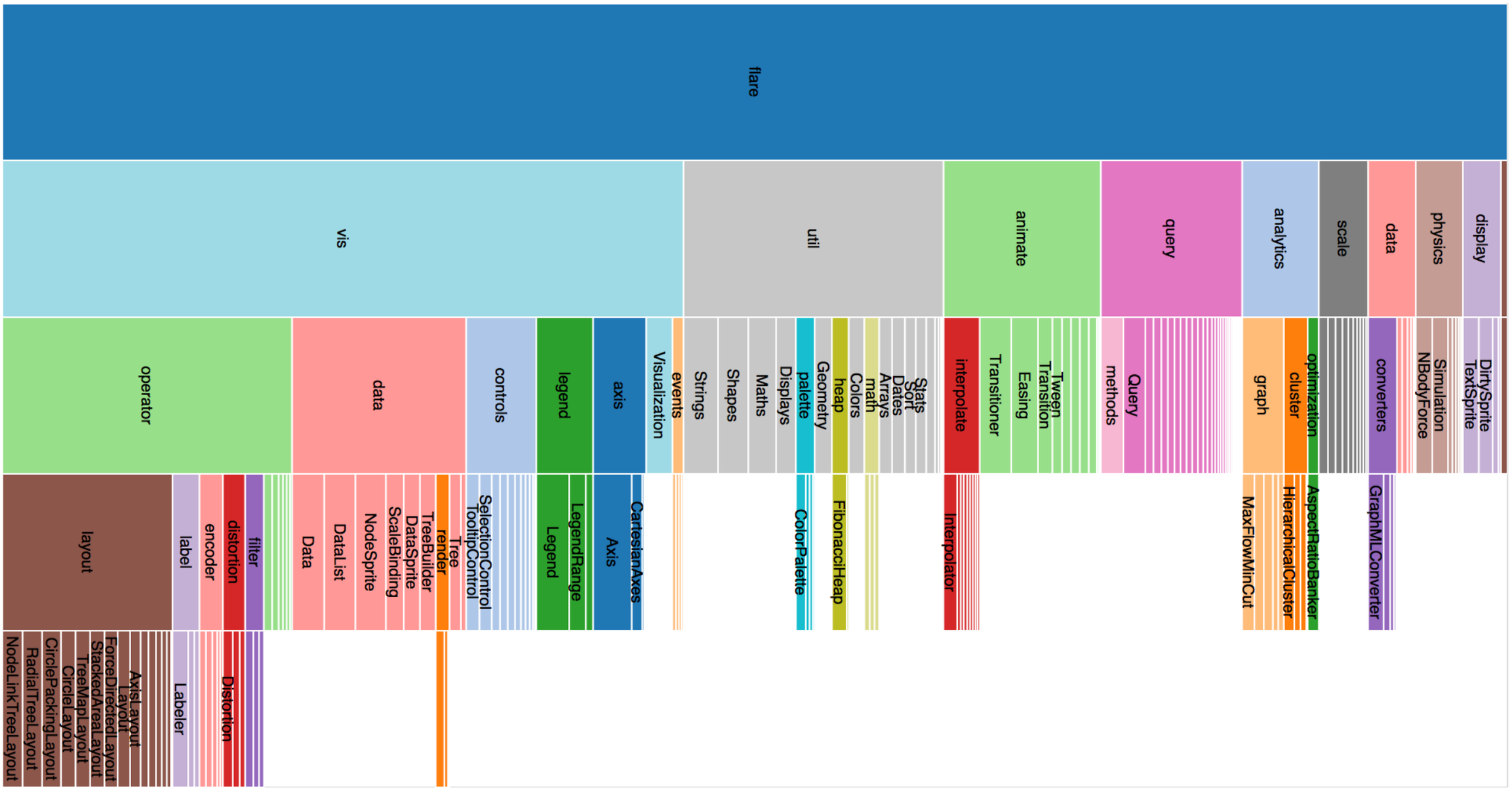


Space Tree example



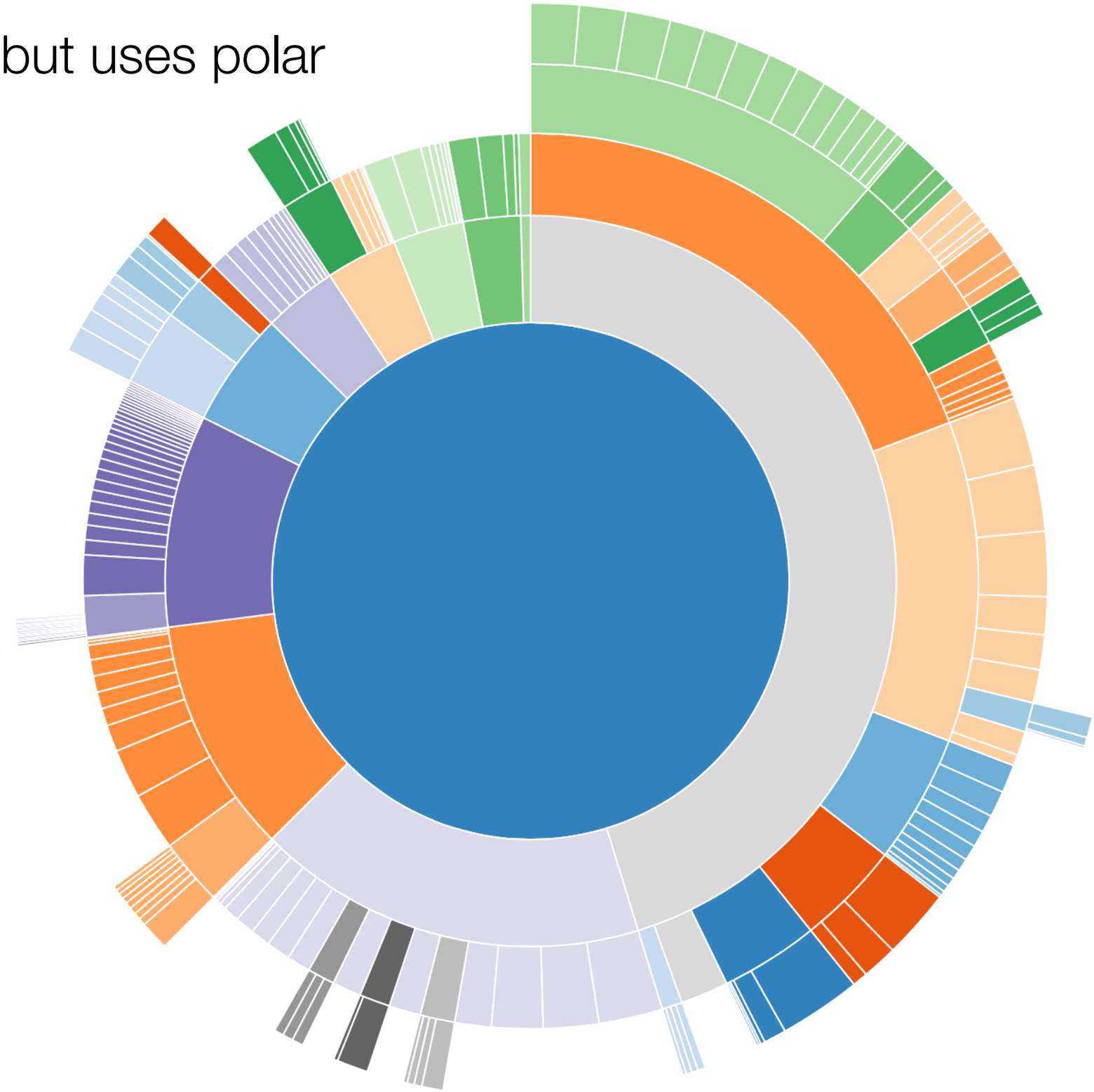
Icicle tree

- How
  - vertical position shows depth
  - horizontal position shows links and sibling order



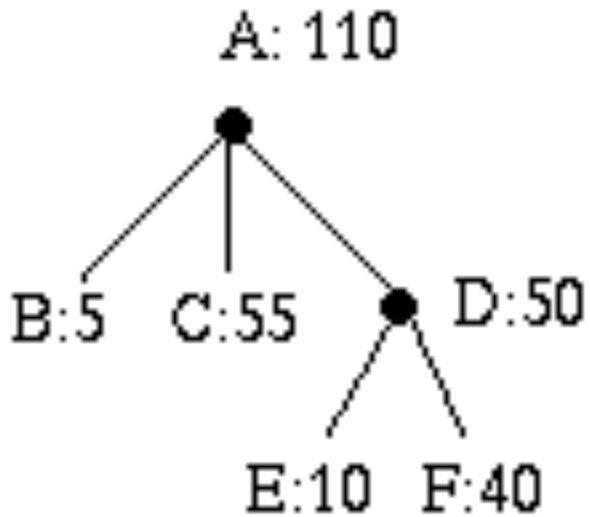
Sunburst

- Equivalent to Icicle tree, but uses polar coordinates



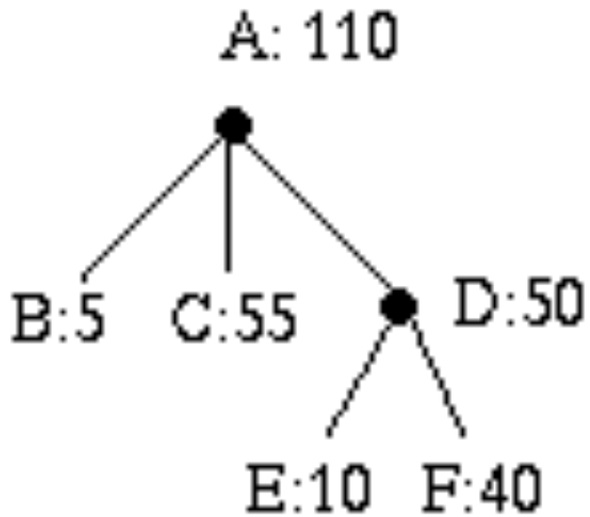
Treemap

- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links



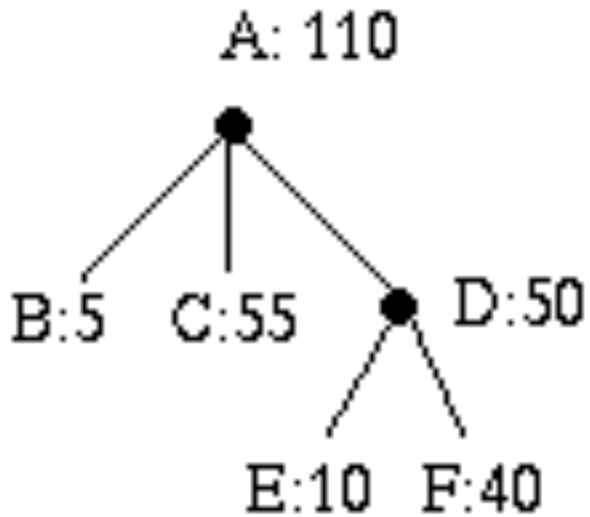
Treemap

- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links



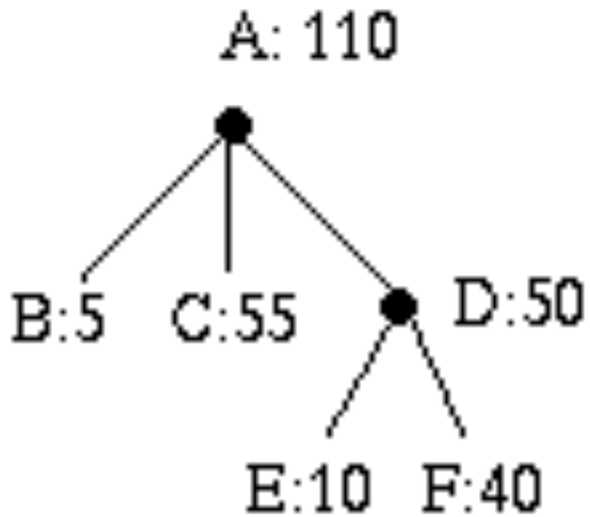
Treemap

- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links



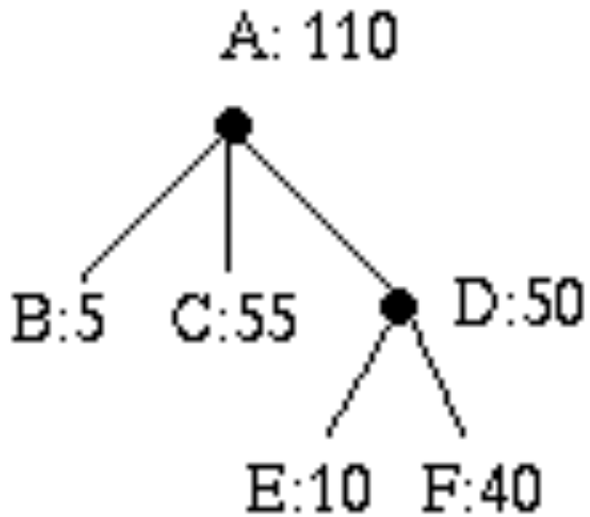
Treemap

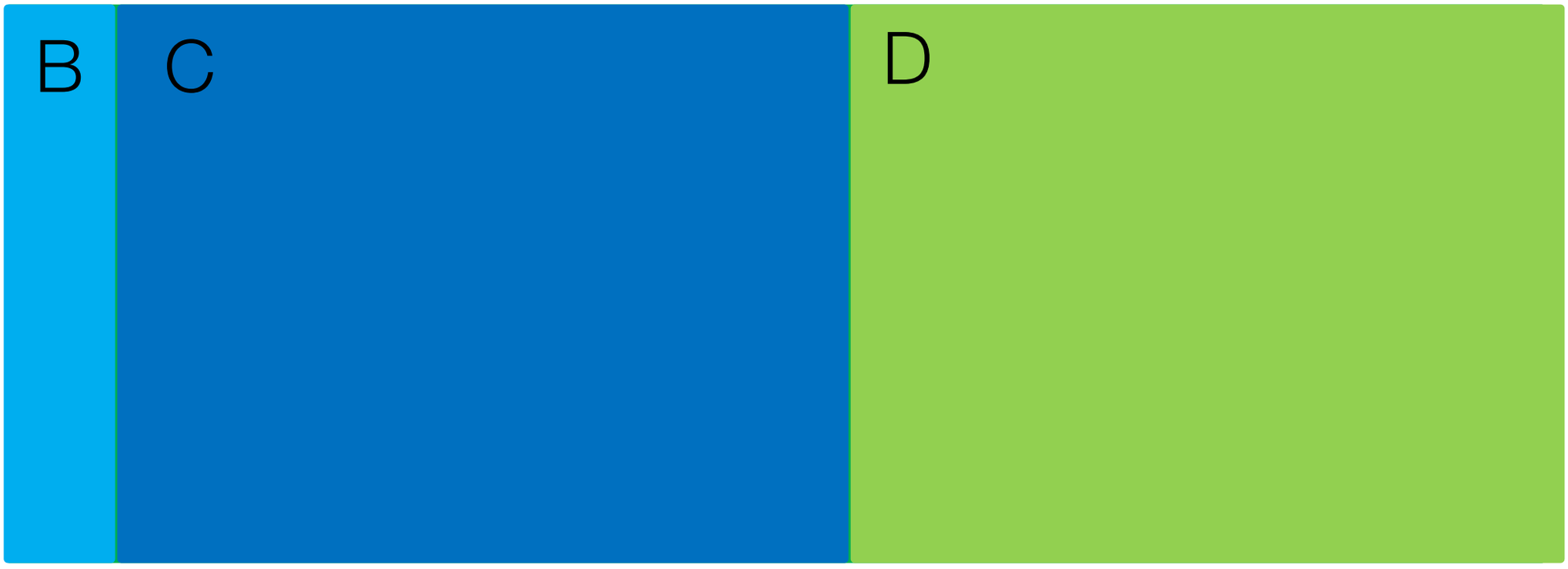
- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links



Treemap

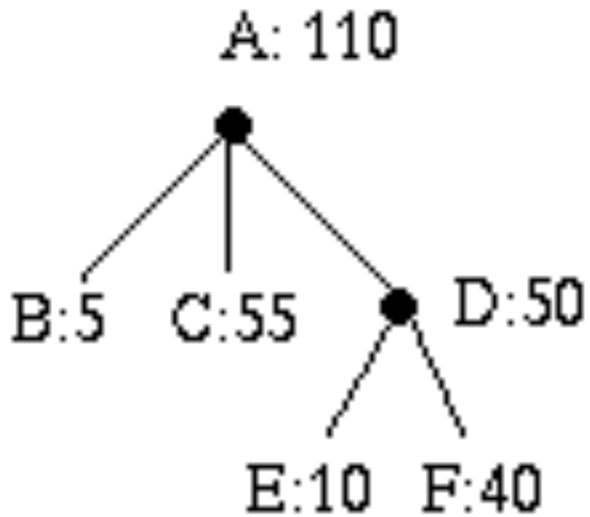
- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links

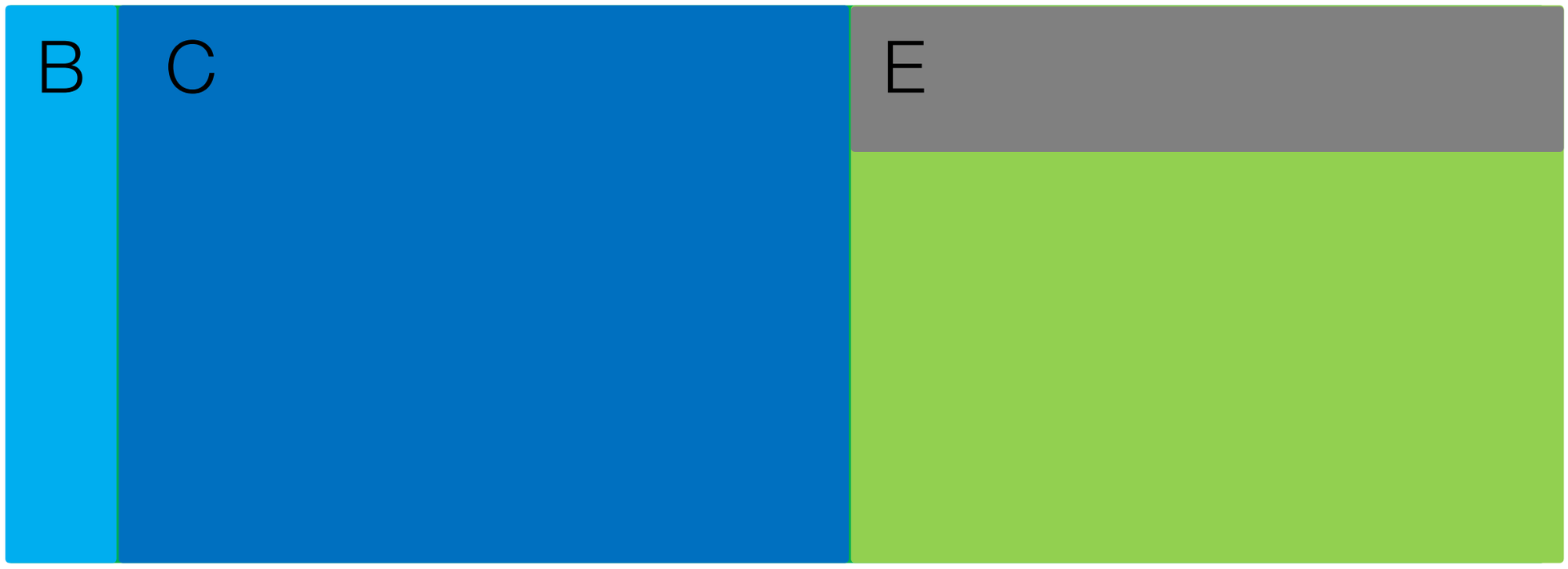




Treemap

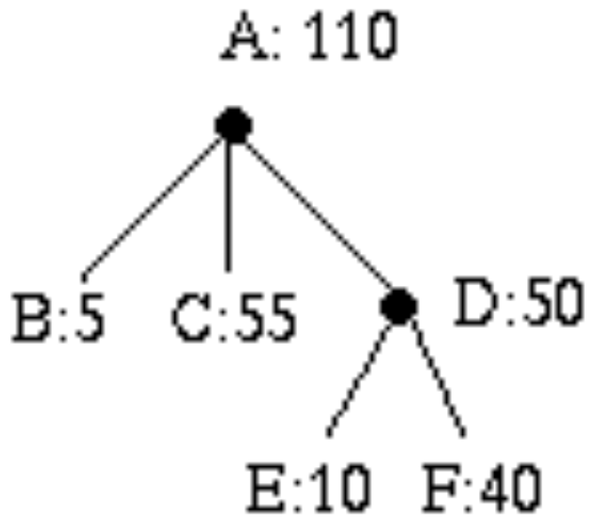
- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links





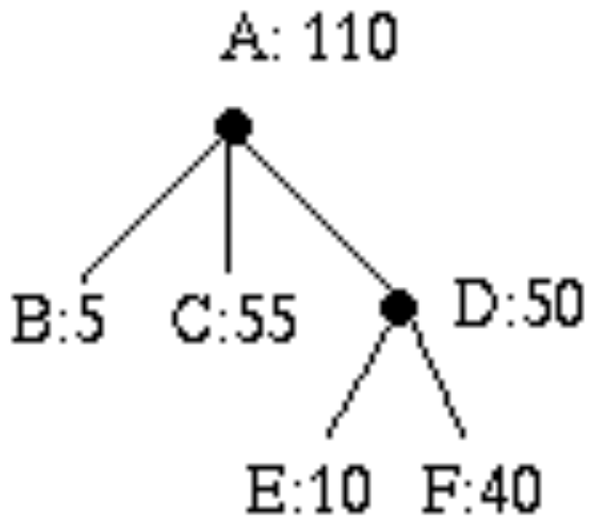
Treemap

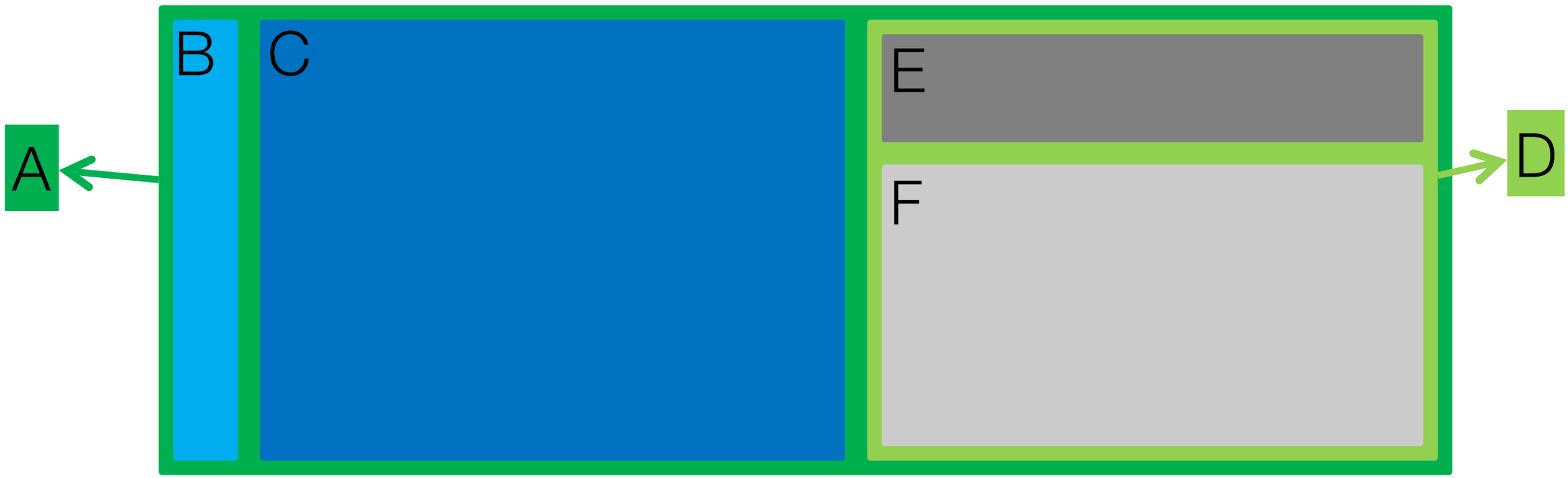
- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes
- Scale
  - million nodes and links



Treemap with borders

- How
  - areas and containment, rectangular layout
- Task
  - query attributes at leaf nodes, understand hierarchy
- Scale
  - million nodes and links

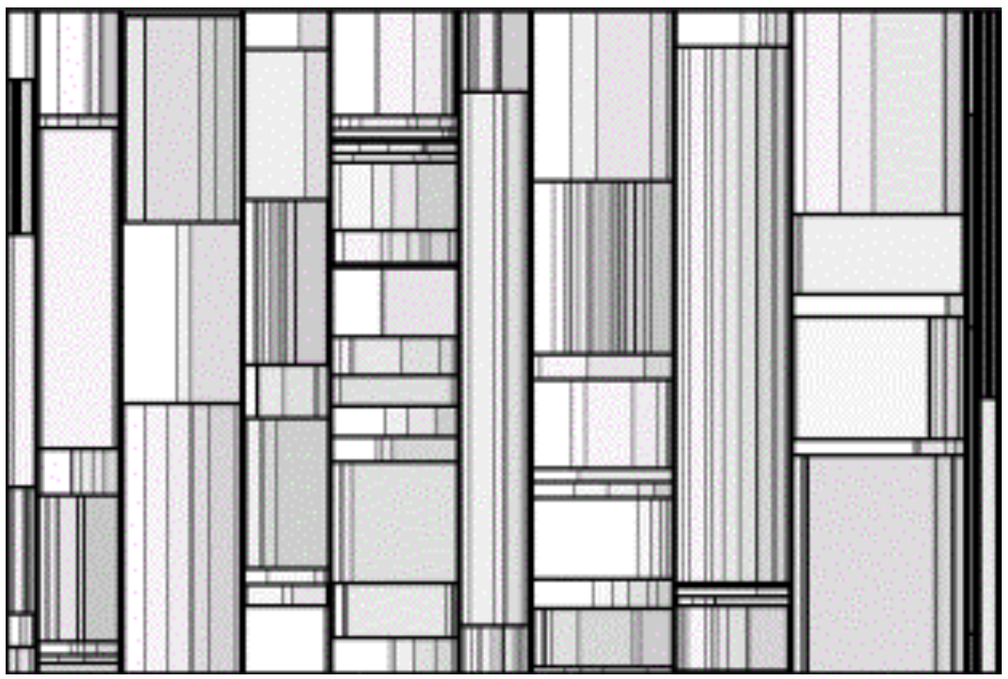




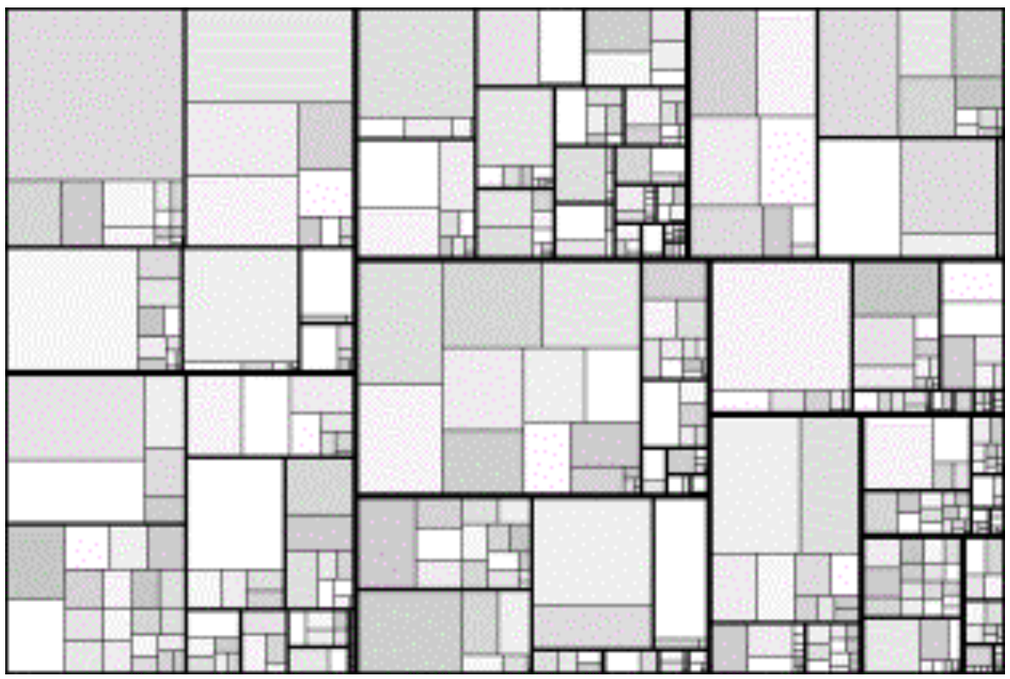
Treemap layout

- Long-thin aspect ratios can occur
- Squarified layout generates nicer rectangles

Bruls, Huizing, van Wijk. Squarified treemaps. Proc. joint Eurographics and IEEE TCVG Symposium on Visualization, pp. 33-42, 2000.

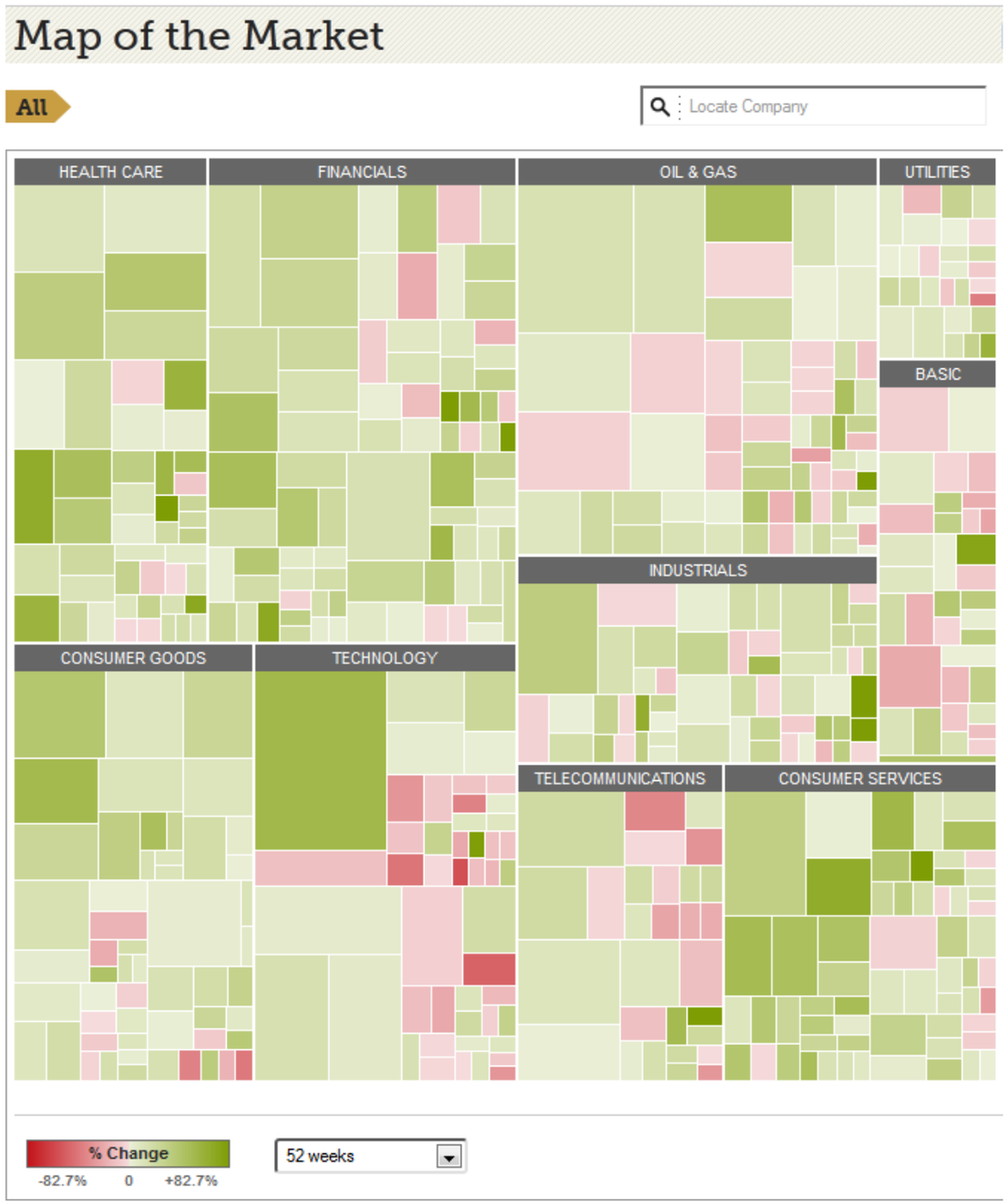


Slice-and-dice layout



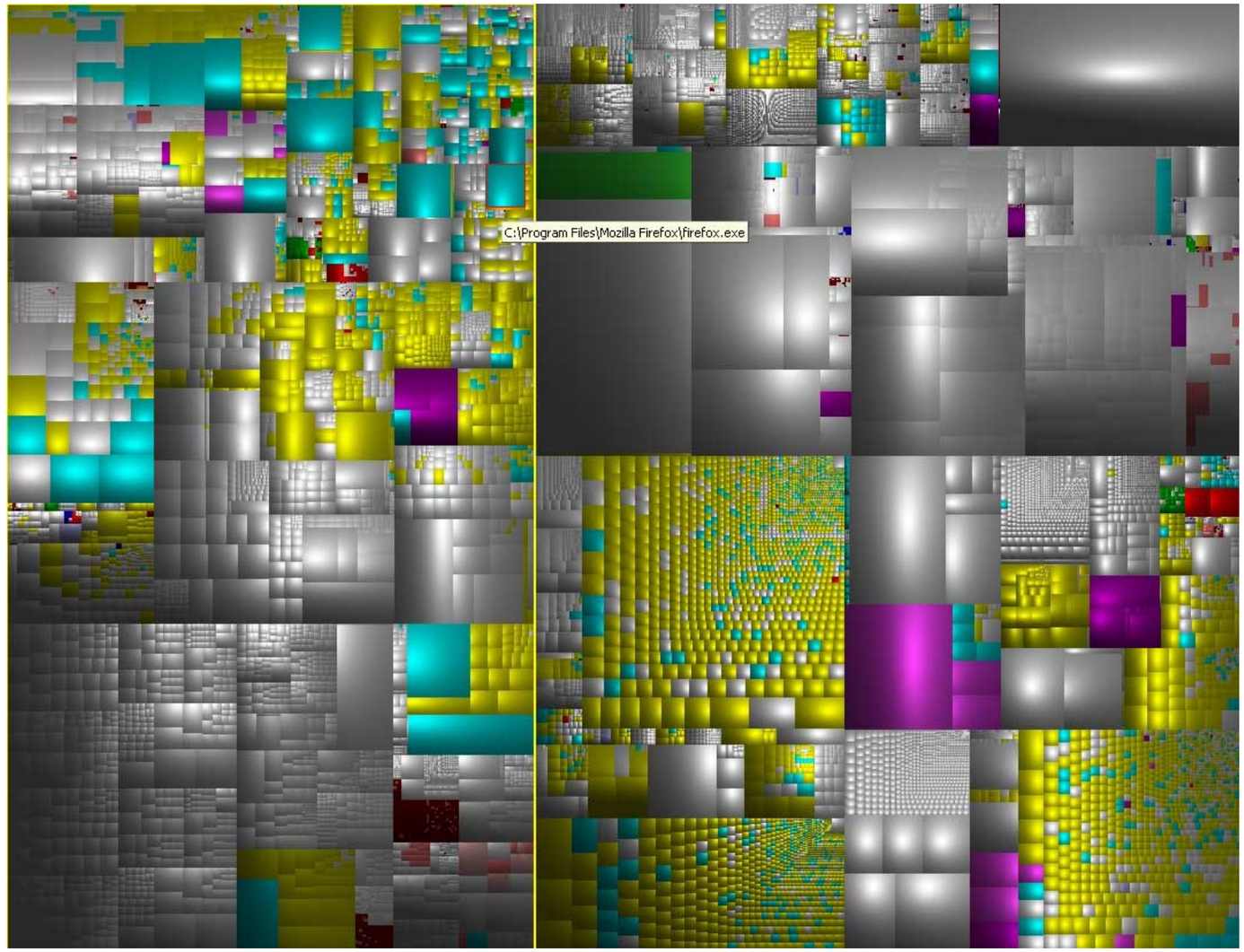
Squarified layout

Treemap applications



http://www.smartmoney.com/map-of-the-market/

https://finviz.com/map.ashx



Sequoiaview

http://www.win.tue.nl/sequoiaview

http://bl.ocks.org/mbostock/4063582

Summary

- Node-link diagrams or space-filling techniques?
- Depends on the properties of data
  - Node-link typically better at exposing structure of information
  - Space-filling good for focusing on one or two additional variables of cases

Networks

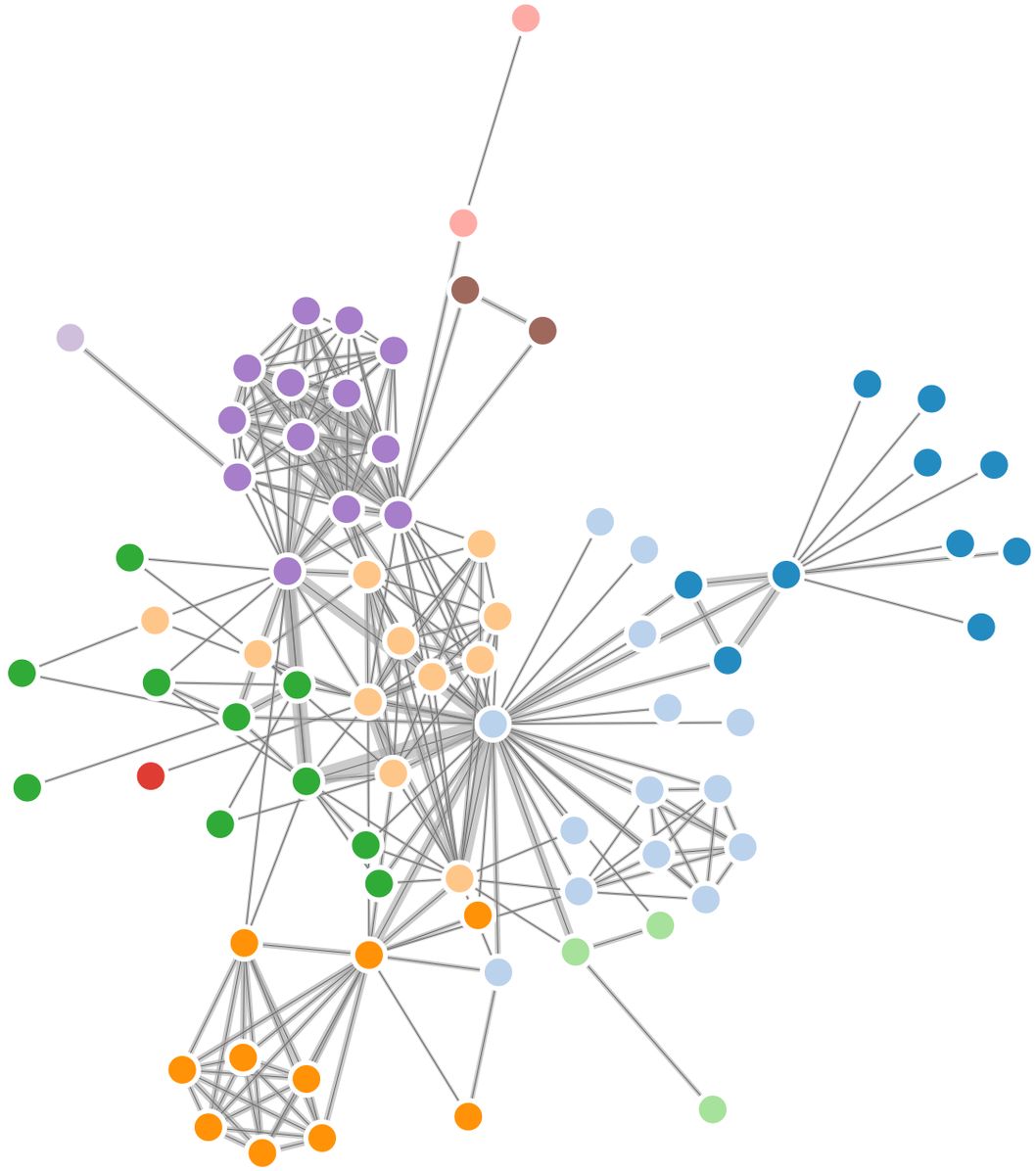

vizster
Michael
Kevin
Cliff
Robert
Jay
Jayne
Matthew
Joey
Rosy
Jason
Shelly
Will
Kingsley
Brennan
Sarah
maggie dj zealo
Christine
Ricardo
Johanna
Dhakshin
Ken
Alan
Grayson
Jonath
Thomas
Gaurav
Nick
Jeff
Sunil
Lei
Steve
Nathan
Josh
Amanda
Joylette
Adam
Alejandra
Patrick
Bryan
Kitty
Tamara
Ann
Haufmause
Christiaan
Bi
Jesse
Jeff
Kyla
Rob
Julie
Ed
Fernando
Alan
Mike
Cynthia
Adam
Patrick
Selena
Vijay
Nitin
Chris
Scott
Siddhartha
Scott
Little
Jacob
Daniel
Chris
John
Kenny
Joseph
Nick
Mark

Requirements

- Planarity
  - draw graph in plane without edge crossings
- Aesthetic rules
  - edges/node evenly distributed
  - edges should be straight
  - isomorphic substructures displayed in same manner
  - minimize edge crossings
- Predictability
  - another run of algorithm gives same layout
- Time complexity

Force-directed layout

- What
  - graph
- How
  - node points and links
  - spatial position has no meaning
  - define forces on nodes and edges
  - define energy on drawing and minimize total energy
- Task
  - explore topology, locate paths
- Scale
  - dozens to hundreds of nodes and edges



http://bl.ocks.org/mbostock/4062045

Vertex force

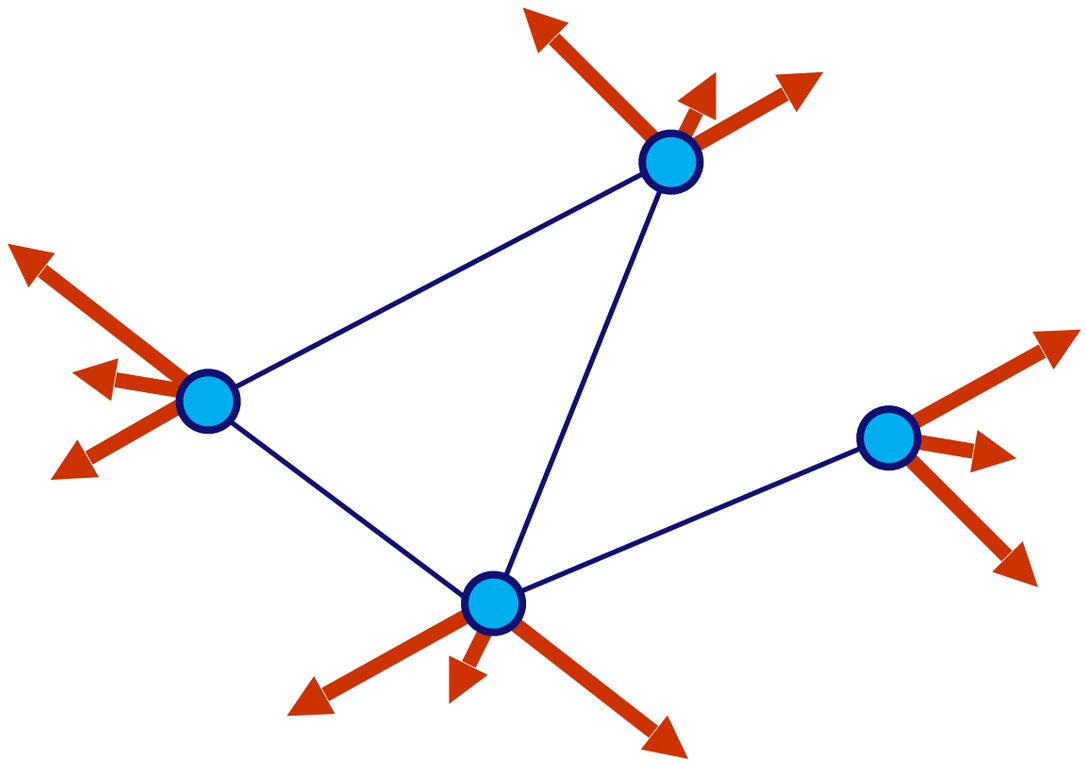
- Repelling force between vertices i and j
- Prevents vertices from coming too close

repulsion strength

Euclidian distance

$$g_{ij}(x) = \frac{r_{ij}}{d(i,j)^2} \frac{x_i - x_j}{d(i,j)}$$

$$g_{ij}(y) = \frac{r_{ij}}{d(i,j)^2} \frac{y_i - y_j}{d(i,j)}$$



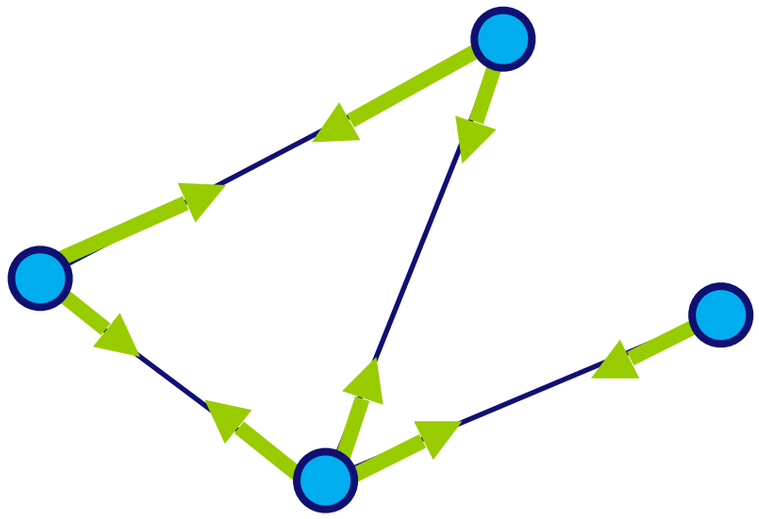
Edge force

- Spring force on edges
- Attracts vertices connected by edge
- Prevents these vertices from getting too far apart

spring tension spring length

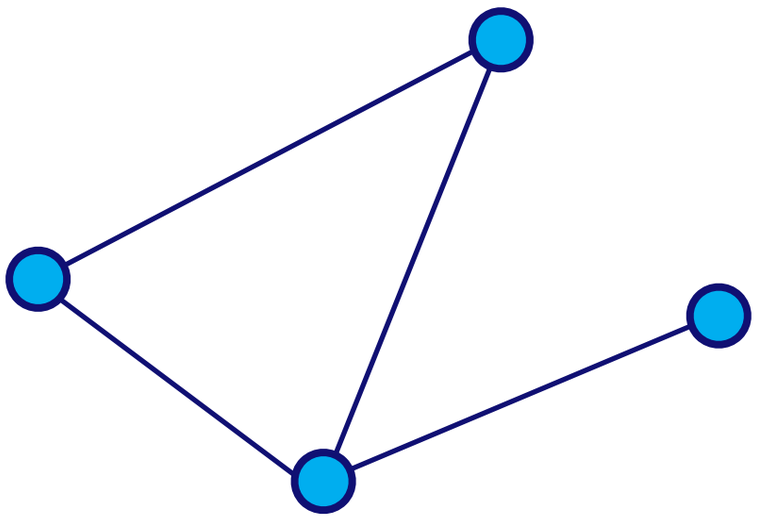
$$f_{ij}(x) = k_{ij}(d(i,j) - s_{ij})\frac{x_i - x_j}{d(i,j)}$$

$$f_{ij}(y) = k_{ij}(d(i,j) - s_{ij})\frac{y_i - y_j}{d(i,j)}$$



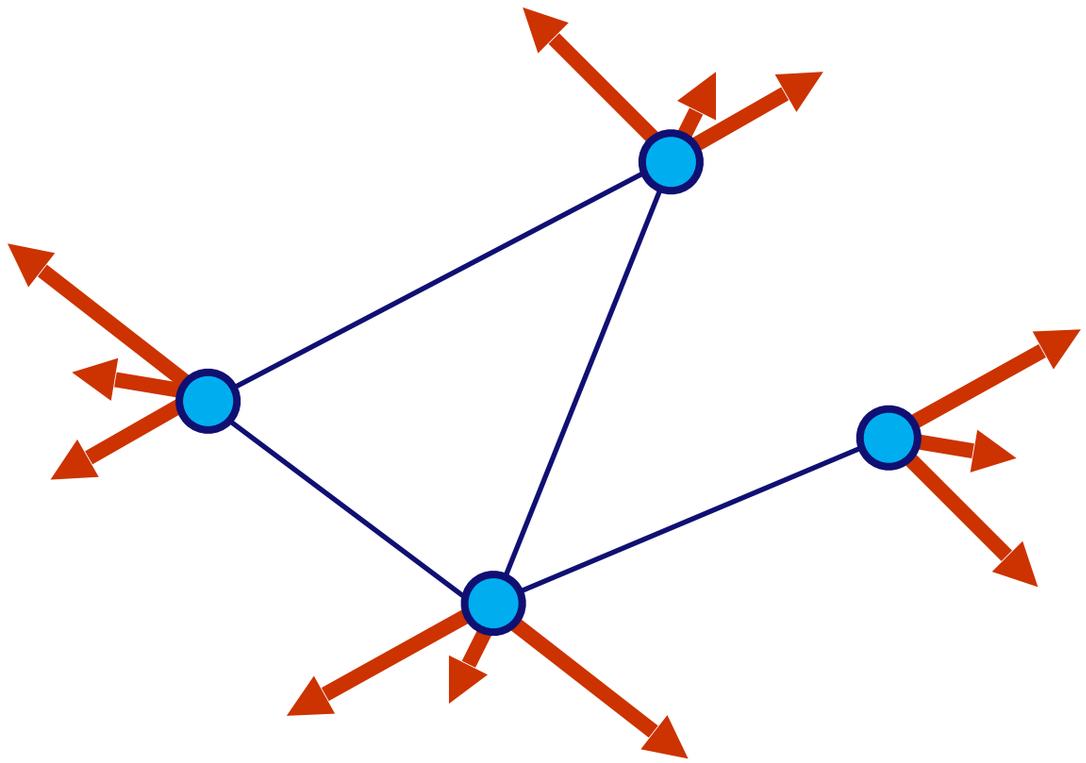
Forces in action

- Compute repulsion and attraction forces for all nodes



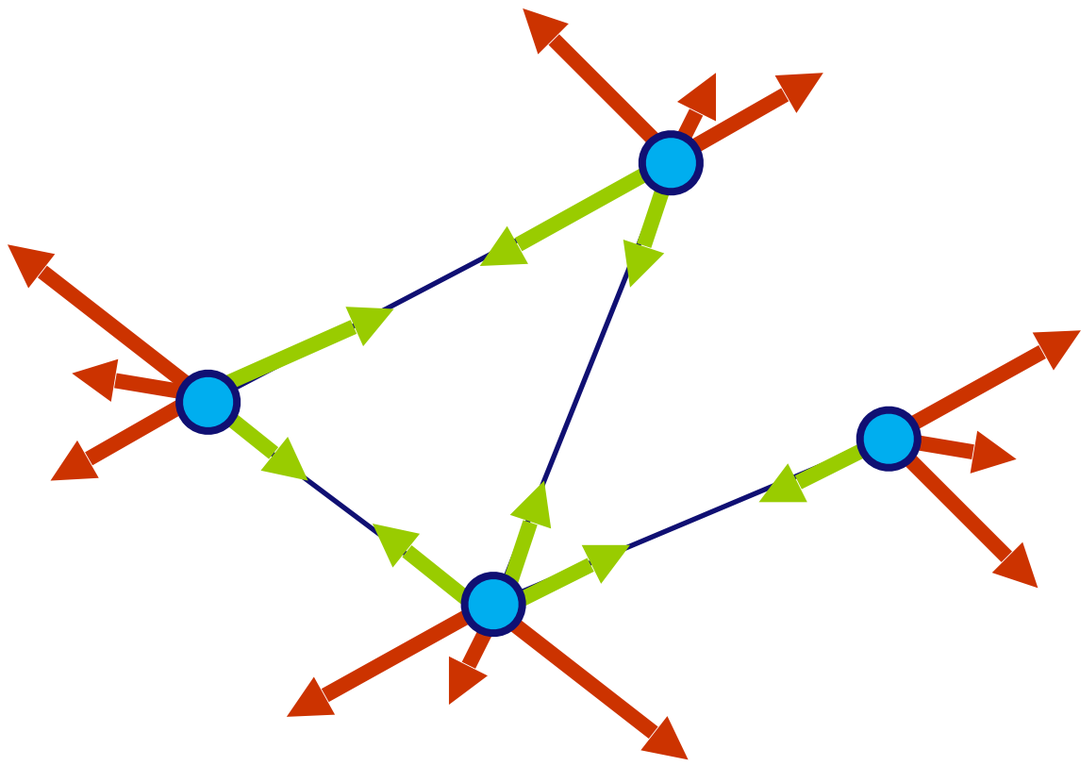
Forces in action

- Compute repulsion and attraction forces for all nodes



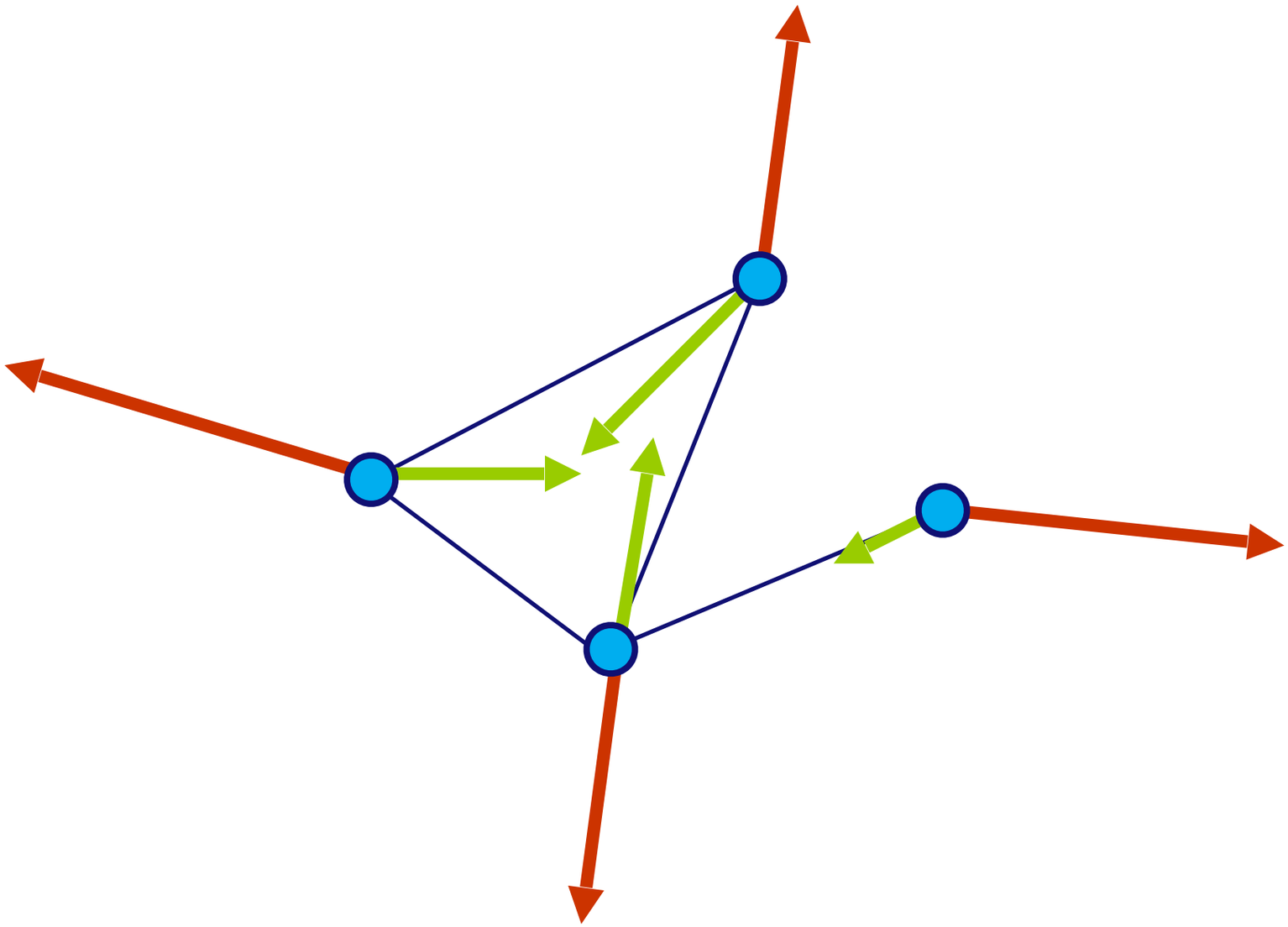
Forces in action

- Compute repulsion and attraction forces for all nodes



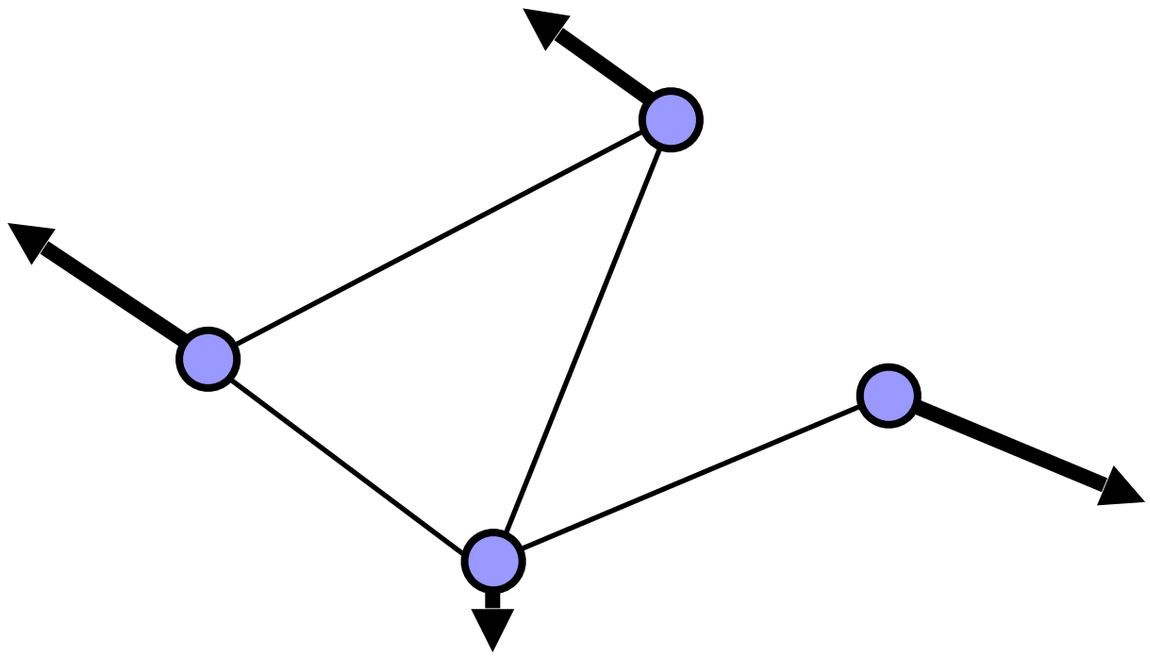
Forces in action

- Compute net repulsion and attraction forces



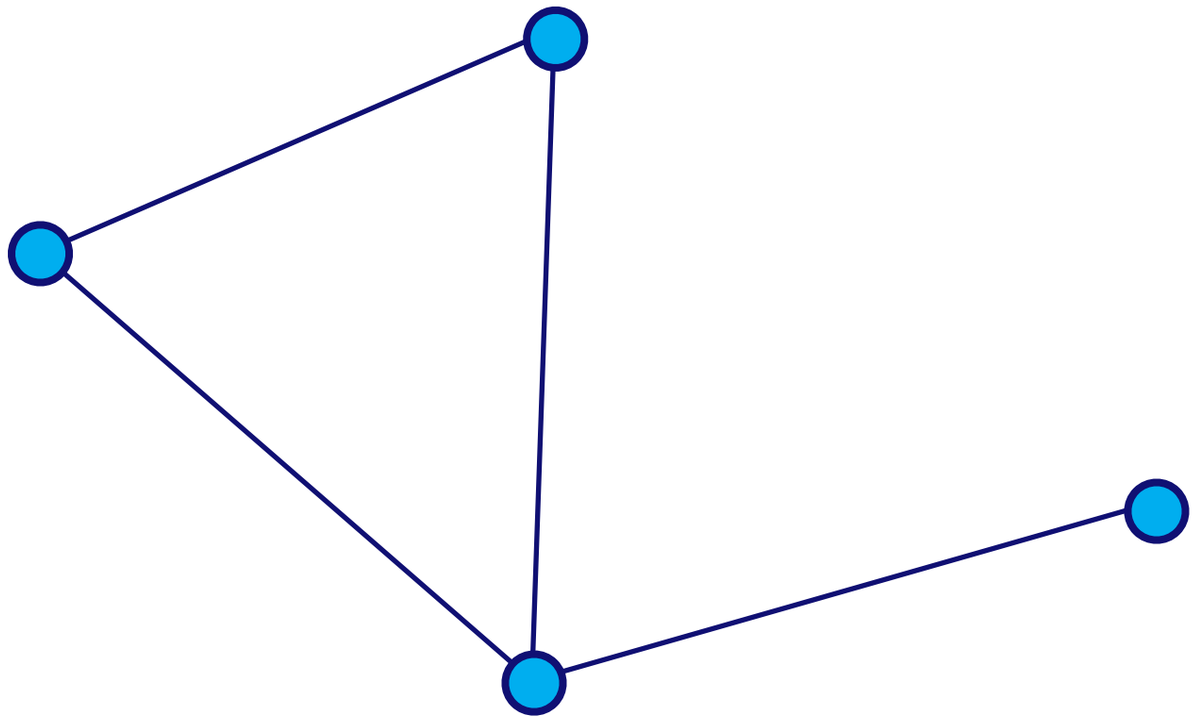
Forces in action

- Compute net force



Forces in action

- Move vertices to their new positions

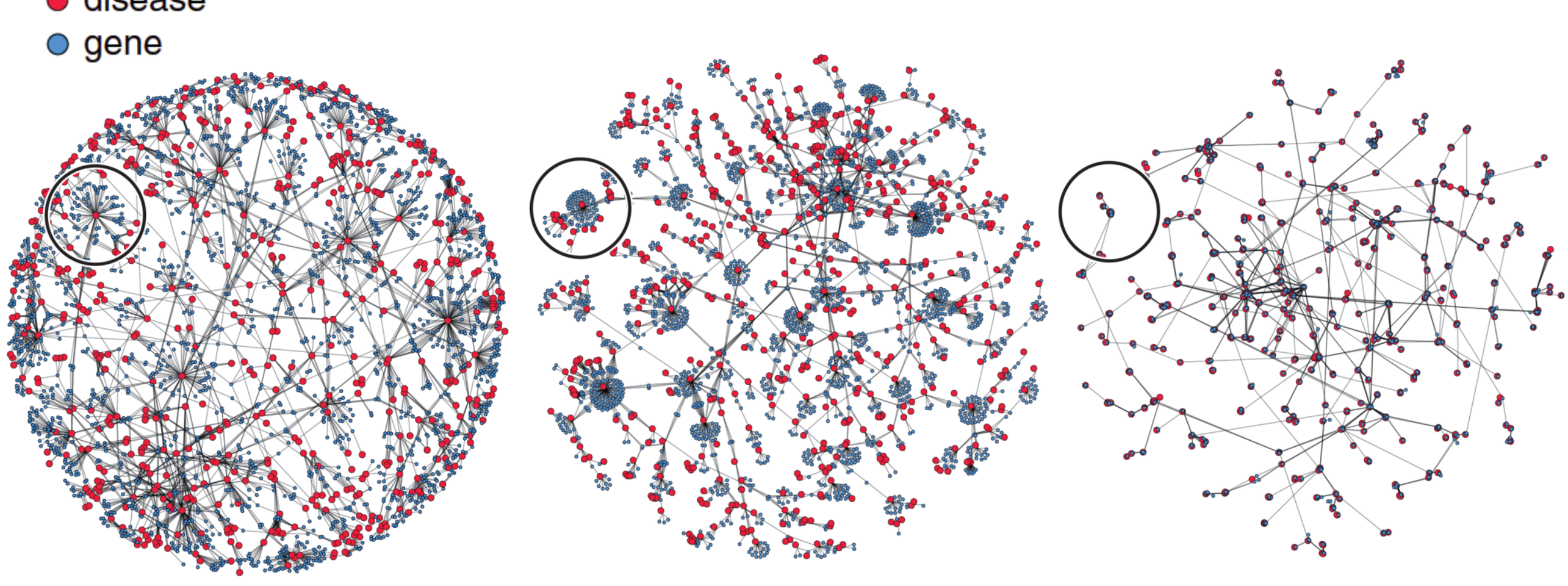


Termination

- Fixed number of iterations
- Total energy below some threshold
- Local minimum
- User input

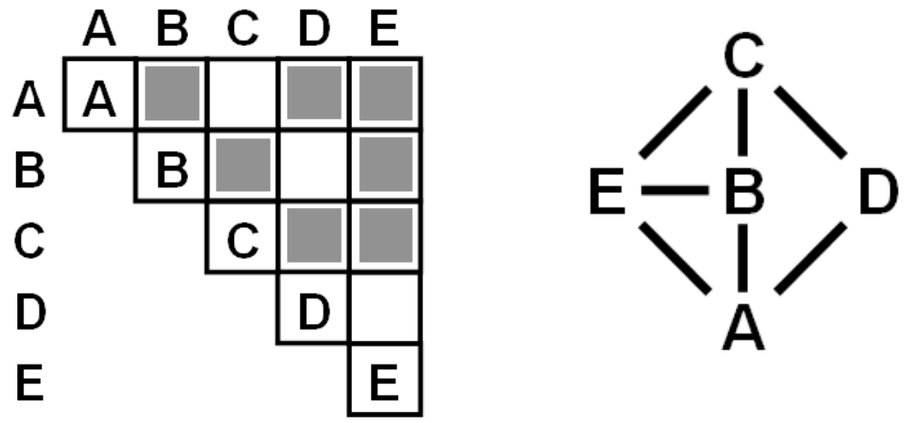
Force-directed layout limitations

- Different variants result in different layouts
- Randomization: different layout for same data on different runs
- Many nodes and edges: hairball
- Spatial proximity of nodes in final layout can be artifact

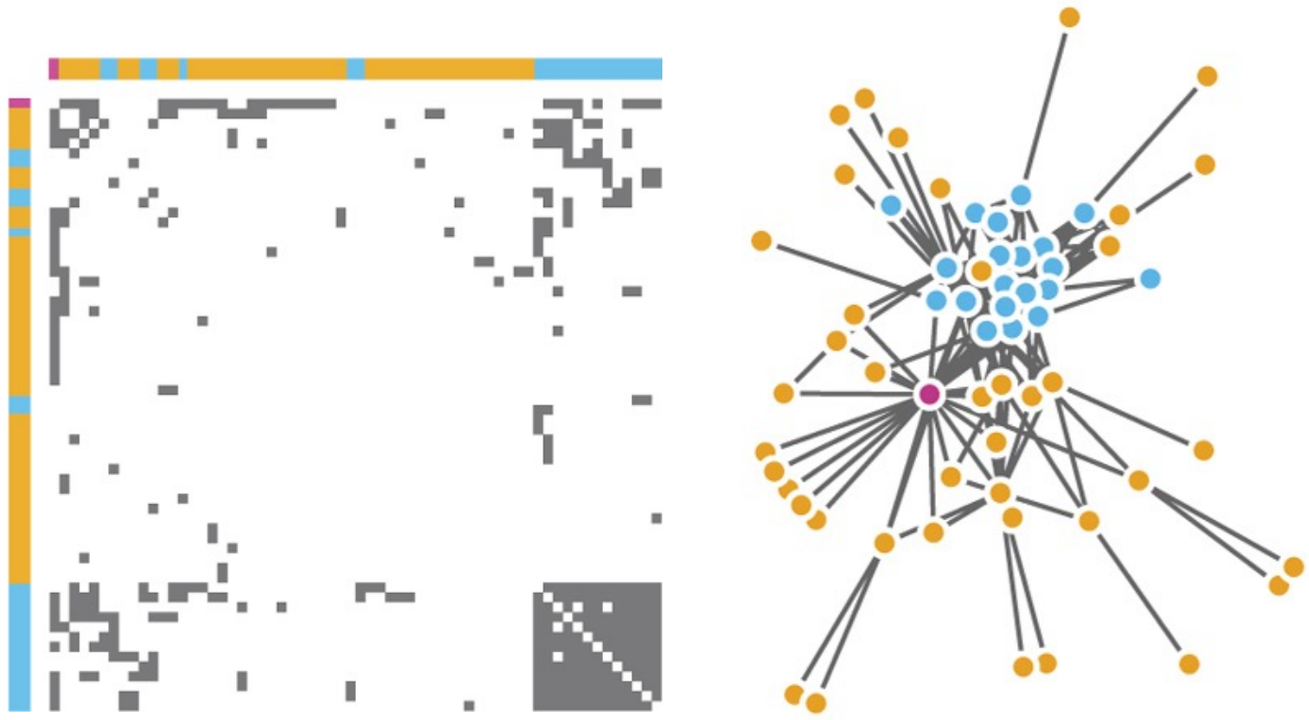


Adjacency matrix view

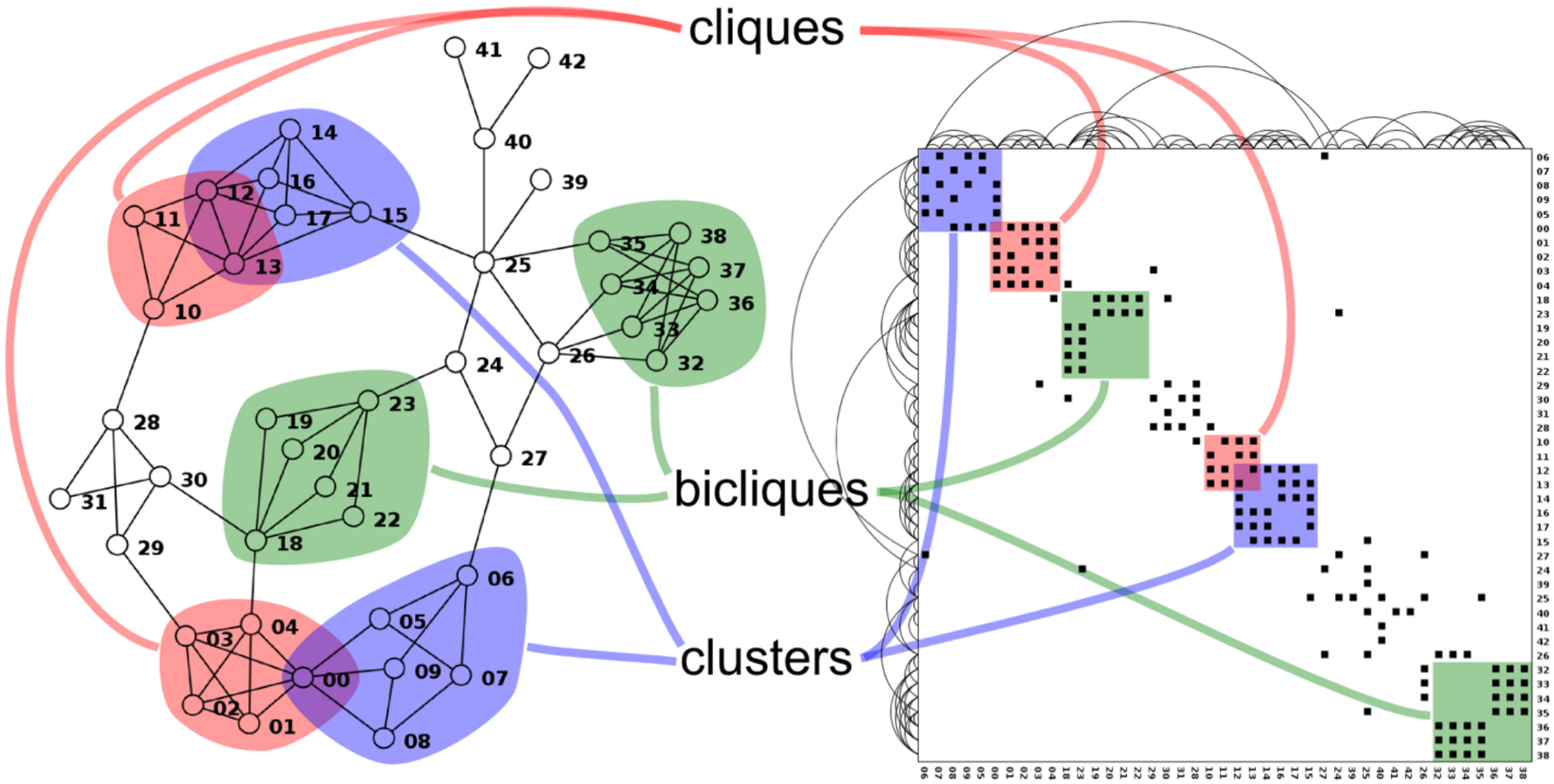
- What
  - graph
  - derived data
    - node categories
    - edge weights
- How
  - same as heatmap
  - cell shows presence/absence of edge
- Scale
  - 1000 nodes, million edges



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



Node-link versus Adjacency matrix



- Node-link
  - topology, path tracing
  - intuitive
- Adjacency matrix
  - predictable, scalable
  - supports reordering

Reading this week

A Design Space of Visualization Tasks

Hans-Jörg Schulz, Thomas Nocke, Magnus Heitzler, and Heidrun Schumann

Abstract—Knowledge about visualization tasks plays an important role in choosing or building suitable visual representations to pursue them. Yet, tasks are a multi-faceted concept and it is thus not surprising that the many existing task taxonomies and models all describe different aspects of tasks, depending on what these task descriptions aim to capture. This results in a clear need to bring these different aspects together under the common hood of a general design space of visualization tasks, which we propose in this paper. Our design space consists of five design dimensions that characterize the main aspects of tasks and that have so far been distributed across different task descriptions. We exemplify its concrete use by applying our design space in the domain of climate impact research. To this end, we propose interfaces to our design space for different user roles (developers, authors, and end users) that allow users of different levels of expertise to work with it.

Index Terms—Task taxonomy, design space, climate impact research, visualization recommendation

1 Introduction

As the field of information visualization matures, a phase of consolidation sets in that aims to pull together multiple individual works of research under a common conceptual hood. This hood can take on different shapes and forms, one of which is the *design space*. Such a design space realizes a descriptive generalization that permits to specify a concrete instance – be it a layout [8], a visualization [46], or a combination of visualizations [28] – by making design choices along a number of independent design dimensions. Even last year's InfoVis conference recognized the increasing importance of design spaces by dedicating an entire session to them.

Yet, information visualization is more than the visual representation alone. It also takes into account the tasks the user wishes to pursue with the visual representation. The literature contains a wealth of classifications, taxonomies, and frameworks that describe these tasks: lists of verbal task descriptions, mathematical task models, domain-specific task collections, and procedural task combinations into workflows. All of these serve the respective purpose well for which they have been developed. However, the research question of how to consolidate them under the hood of one common design space is still open, even though it has been shown on a smaller scale that such a combination into a common framework can be a useful endeavor [9, 21].

In this paper, we aim to give a first answer to this research question by contributing such a design space for visualization tasks. This contribution is twofold. First, it derives an abstract design space that a visualization task design space for climate impact research based on structured interviews with eight domain experts and two visualization developers. This design space is then utilized to recommend visualizations that are suitable to pursue a given task in that field.

The remainder of this paper is organized as follows: The related work is summarized in Section 2 and from its discussion, we derive our task design space in Section 3. We then debate its properties, limitations, and applications in Section 4. This also includes examples of how some of the existing task taxonomies can be expressed as parts of our design space. After this conceptual part, Section 5 details the use case example of how to apply the general design space to the application domain of climate impact research and how to draw concrete benefits from it. With this example, we aim to show a feasible way for the adaptation of the design space that can be transferred to other application domains as well. We conclude this paper by briefly sharing our personal experience from working with the design space and pointing out directions for future work in Section 6.

2 Related Work

The concept of *tasks* exhibits numerous facets that are also reflected in the existing body of research on that topic. Commonly, *visualization tasks* are understood as activities to be carried out interactively on a visual data representation for a particular reason. The investigation of visualization tasks has the aim to **establish recurring tasks** in order

DOI:10.1145/1743546.1743567

Article development led by acmqueue
queue.acm.org

A survey of powerful visualization techniques, from the obvious to the obscure.

BY JEFFREY HEER, MICHAEL BOSTOCK, AND VADIM OGIEVETSKY

A Tour Through the Visualization Zoo

THANKS TO ADVANCES in sensing, networking, and data management, our society is producing digital information at an astonishing rate. According to one estimate, in 2010 alone we will generate 1,200 exabytes—60 million times the content of the Library of Congress. Within this deluge of data lies a wealth