Data Visualization (CSCI 627/490)

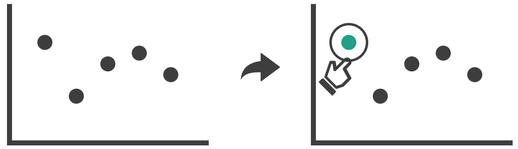
Networks

Dr. David Koop

Interaction Overview

Change over Time

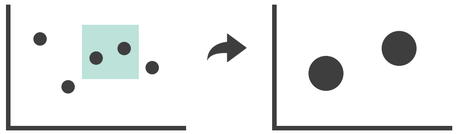
Select



Navigate

Item Reduction

- *Zoom*
  Geometric or *Semantic*



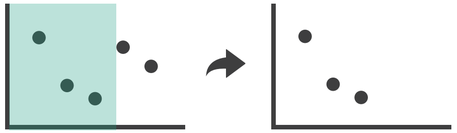
- *Pan/Translate*

- *Constrained*

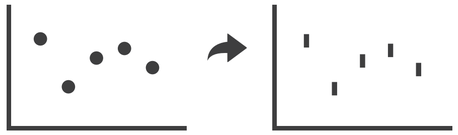
Attribute Reduction

- *Slice*

- *Cut*

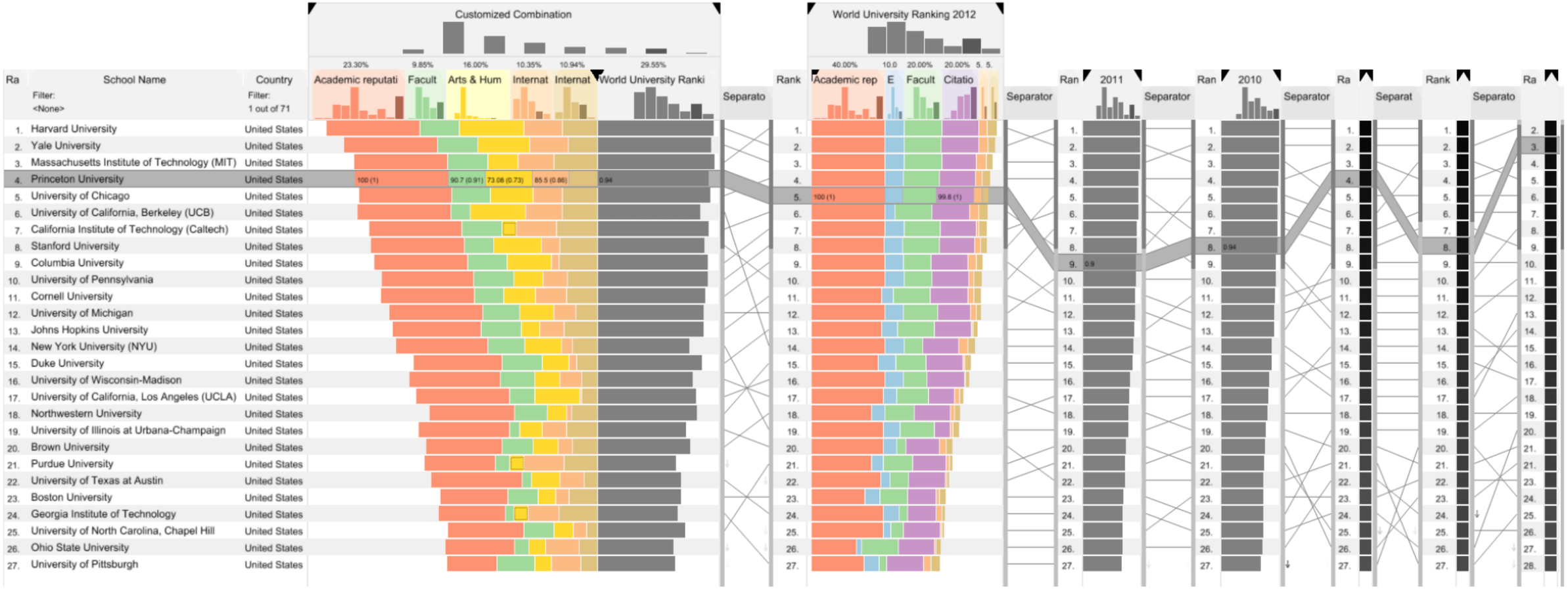


- *Project*



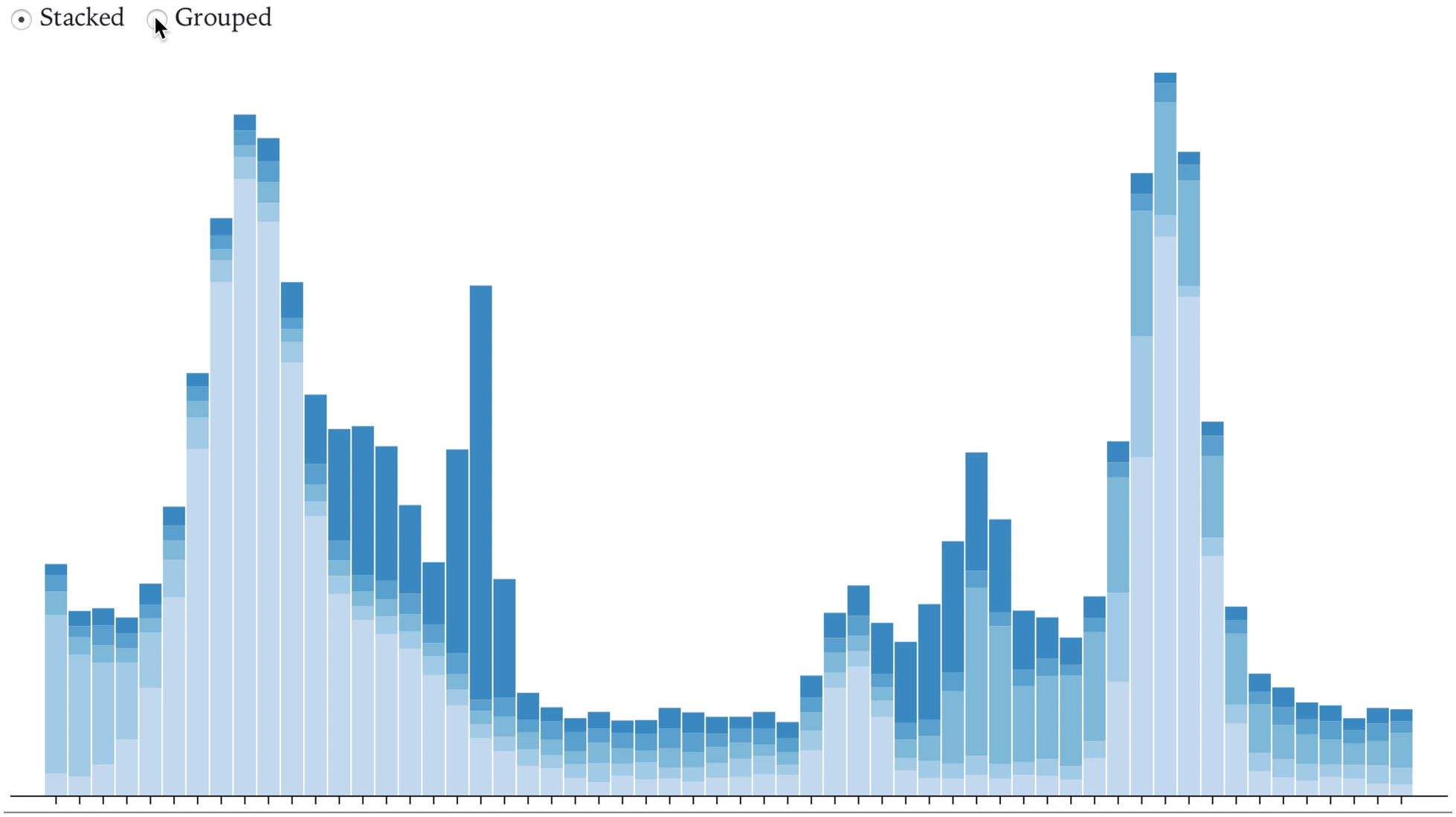
[Munzner (ill. Maguire), 2014]

Sorting & Slope Graphs: LineUp



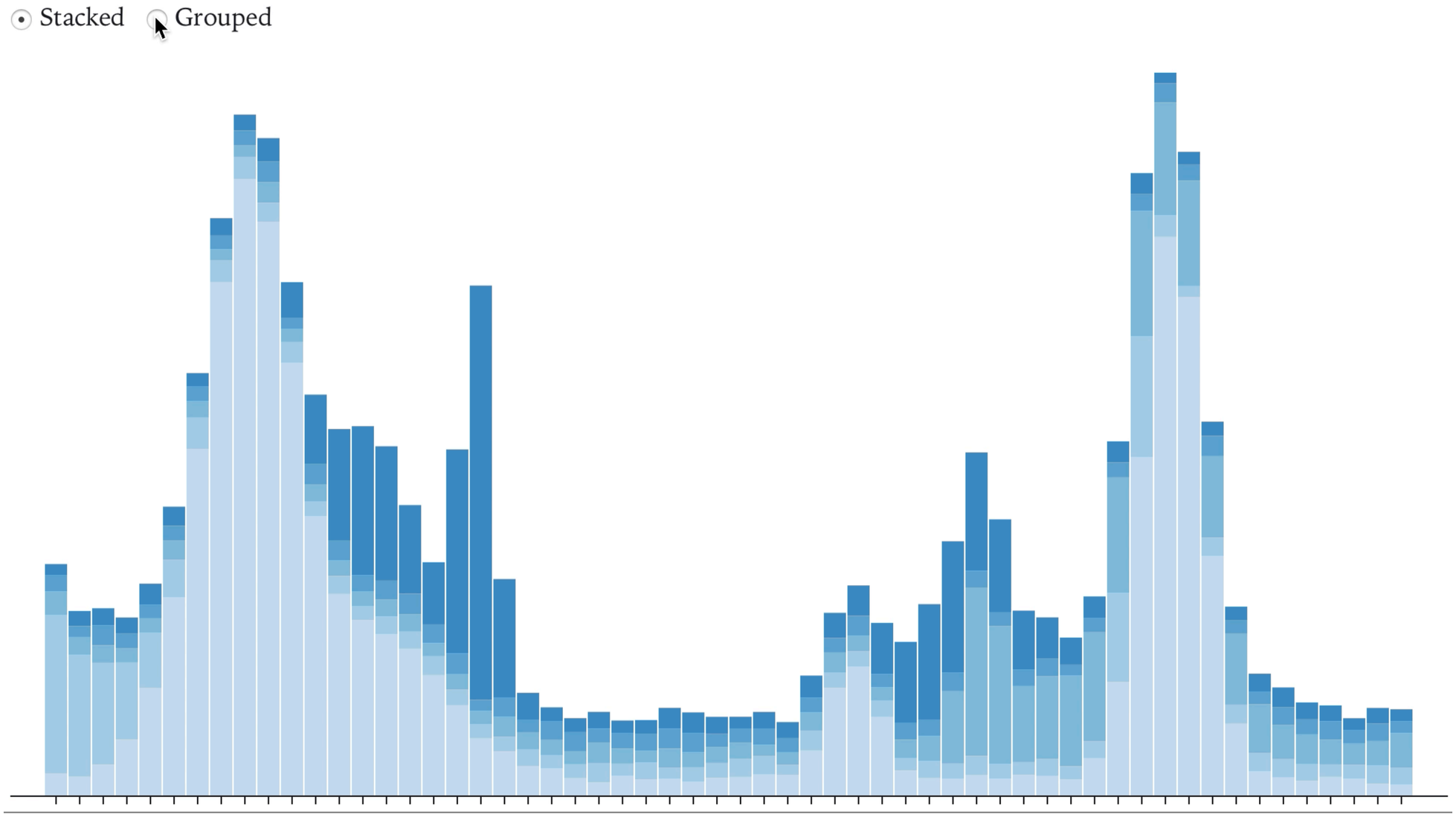
[Gratzl et al., 2013]

Animated Transitions



[M. Bostock]

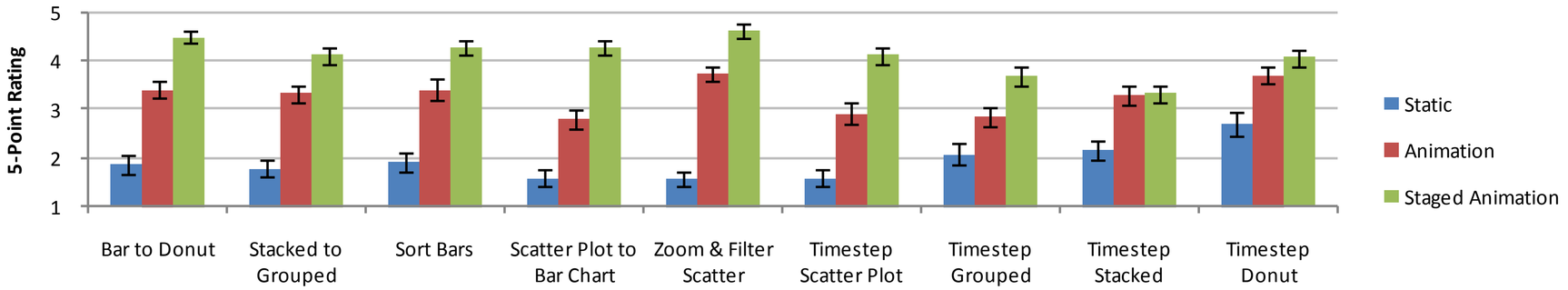
Animated Transitions



[M. Bostock]

Heer and Robertson Study

- User Preferences: Staged animation > animation > static transitions



- Animation improves graphical perception
- Staging is better (do axis rescaling before value changes)
- Avoid axis rescaling when possible

[Heer and Robertson, 2007]

Selection

- Selection is often used to initiate other changes
- User needs to select something to drive the next change
- What can be a selection target?
  - Items, links, attributes, (views)
- How?
  - mouse click, mouse hover, touch
  - keyboard modifiers, right/left mouse click, force
- Selection modes:
  - Single, multiple
  - Contiguous?

Highlighting

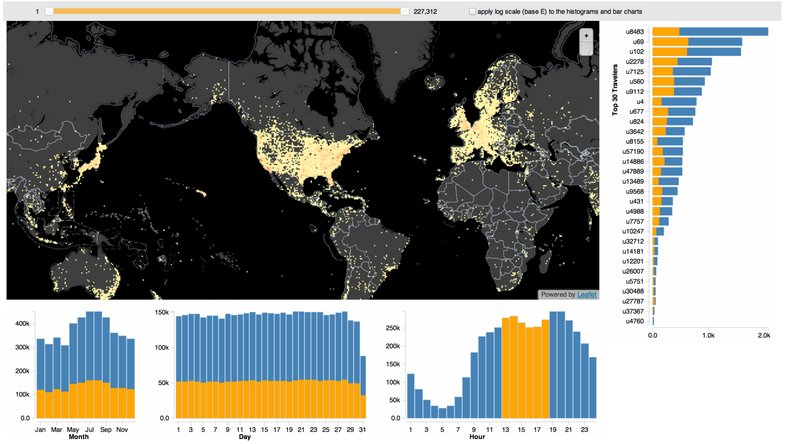
- Selection is the user action
- Feedback is important!
- How? Change selected item's visual encoding
  - Change color: want to achieve visual popout
  - Add outline mark: allows original color to be preserved
  - Change size (line width)
  - Add motion: marching ants

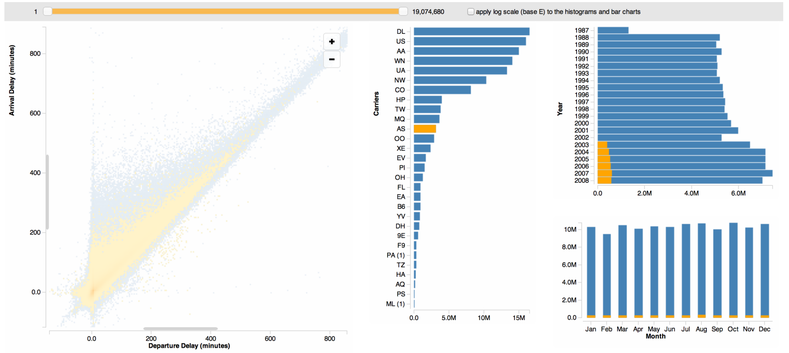
Highlighting

- Selection is the user action
- Feedback is important!
- How? Change selected item's visual encoding
  - Change color: want to achieve visual popout
  - Add outline mark: allows original color to be preserved
  - Change size (line width)
  - Add motion: marching ants

Interaction Latency

- The Effects of Interactive Latency on Exploratory Visual Analysis, Z. Liu and J. Heer, 2014
- Brush & link, select, pan, zoom

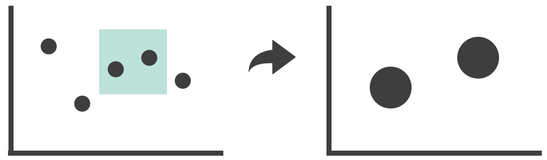




- 500ms added latency causes significant cost
  - decreases user activity and dataset coverage
  - reduces rate of observations, generalizations, and hypotheses

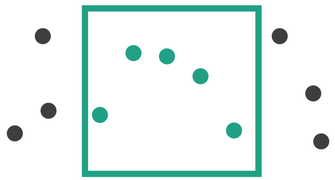
Navigation

- Item Reduction
  - *Zoom*
    Geometric or *Semantic*



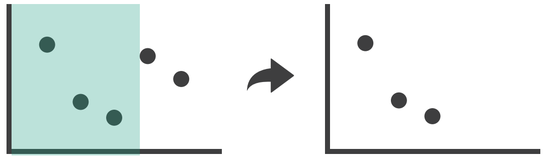
  - *Pan/Translate*

  - *Constrained*

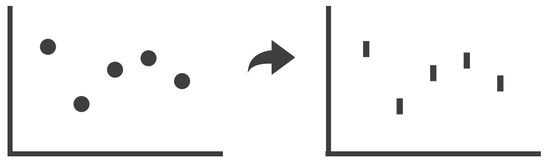


- Attribute Reduction
  - *Slice*

  - *Cut*



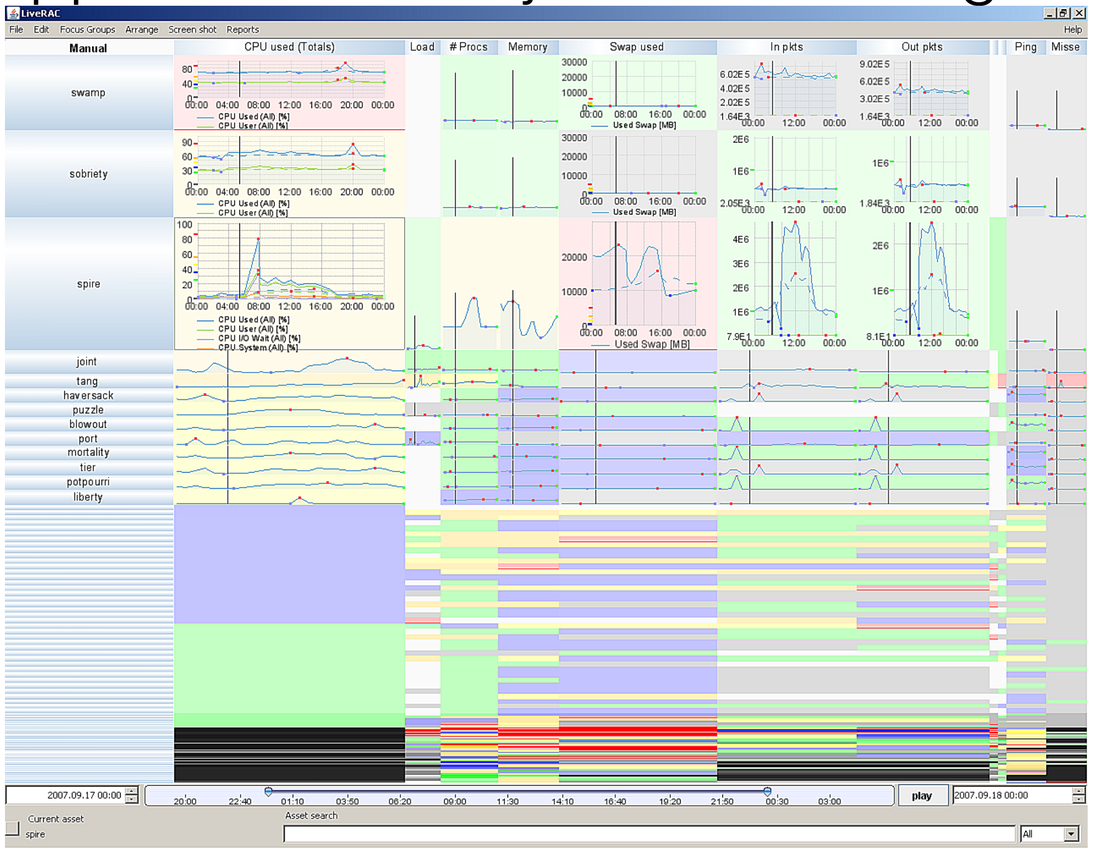
  - *Project*



[Munzner (ill. Maguire), 2014]

Zooming

- Geometric Zooming: just like a camera
- Semantic Zooming: visual appearance of objects can change at different scales
- LiveRAC Example: (focus + context)

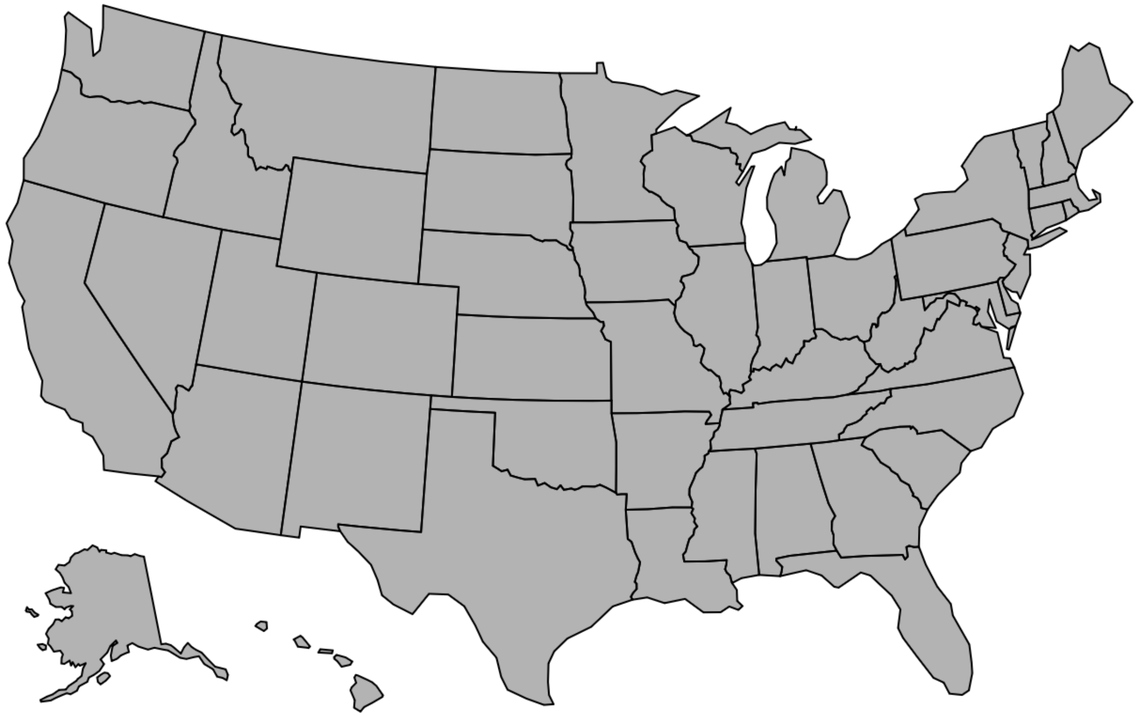


[McLachlan et al., 2008]

Projects

- Proposal feedback up on Blackboard
- General comments:
  - Attributes not in the dataset (where are you getting external data?)
  - Tasks not well suited to visualization (data queries)
  - Visualization creativity: think about custom designs
- Designs:
  - Three different good designs
  - One bad design
  - Sketch these
  - Progress on implementation

Assignment 4



- Choropleth Maps
  - Use D3 for Part 1
  - Can use either D3 or Plot for Part 2
  - Make sure the colormap is appropriate!
- Treemap [627]
- Two resources:
  - Courselet (Plot in python)
  - Observable Notebook on Maps

Networks

Networks

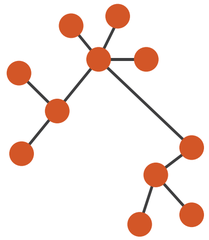
- Why not graphs?
  - Bar graph
  - Graphing functions in mathematics
- Network: nodes and edges connecting the nodes
- Formally, G = (V,E) is a set of nodes V and a set of edges E where each edge connects two nodes.
- Nodes == items, edges connect items
- **Both** nodes and edges may have **attributes**

Arrange Networks and Trees

Node–Link Diagrams

Connection Marks

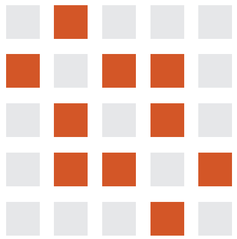
✔ NETWORKS ✔ TREES



Adjacency Matrix

Derived Table

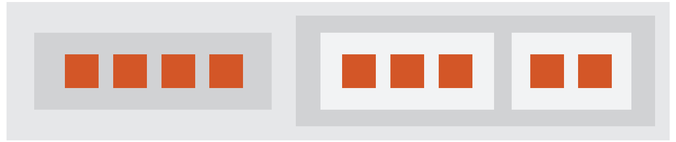
✔ NETWORKS ✔ TREES



Enclosure

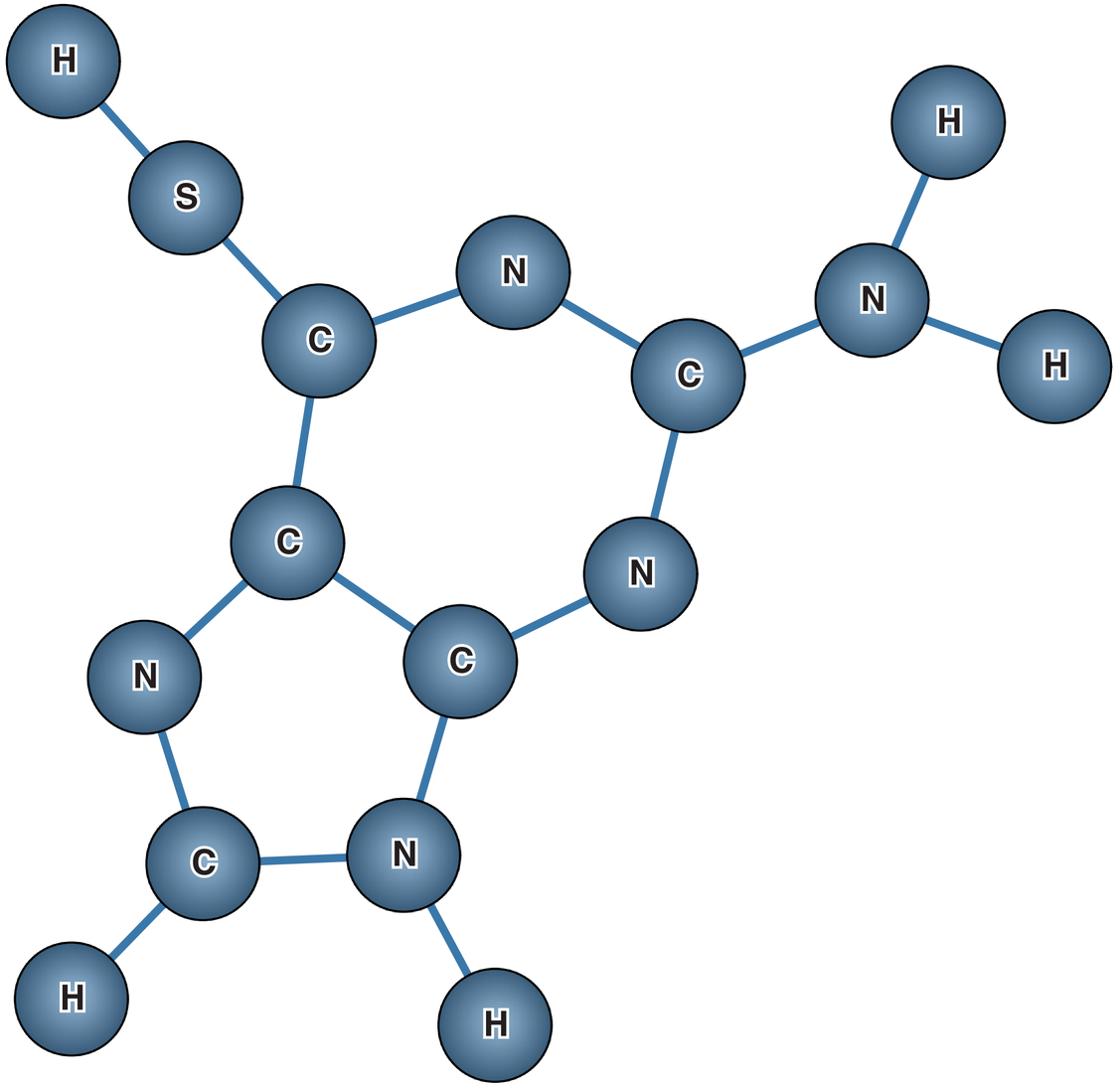
Containment Marks

✘ NETWORKS ✔ TREES

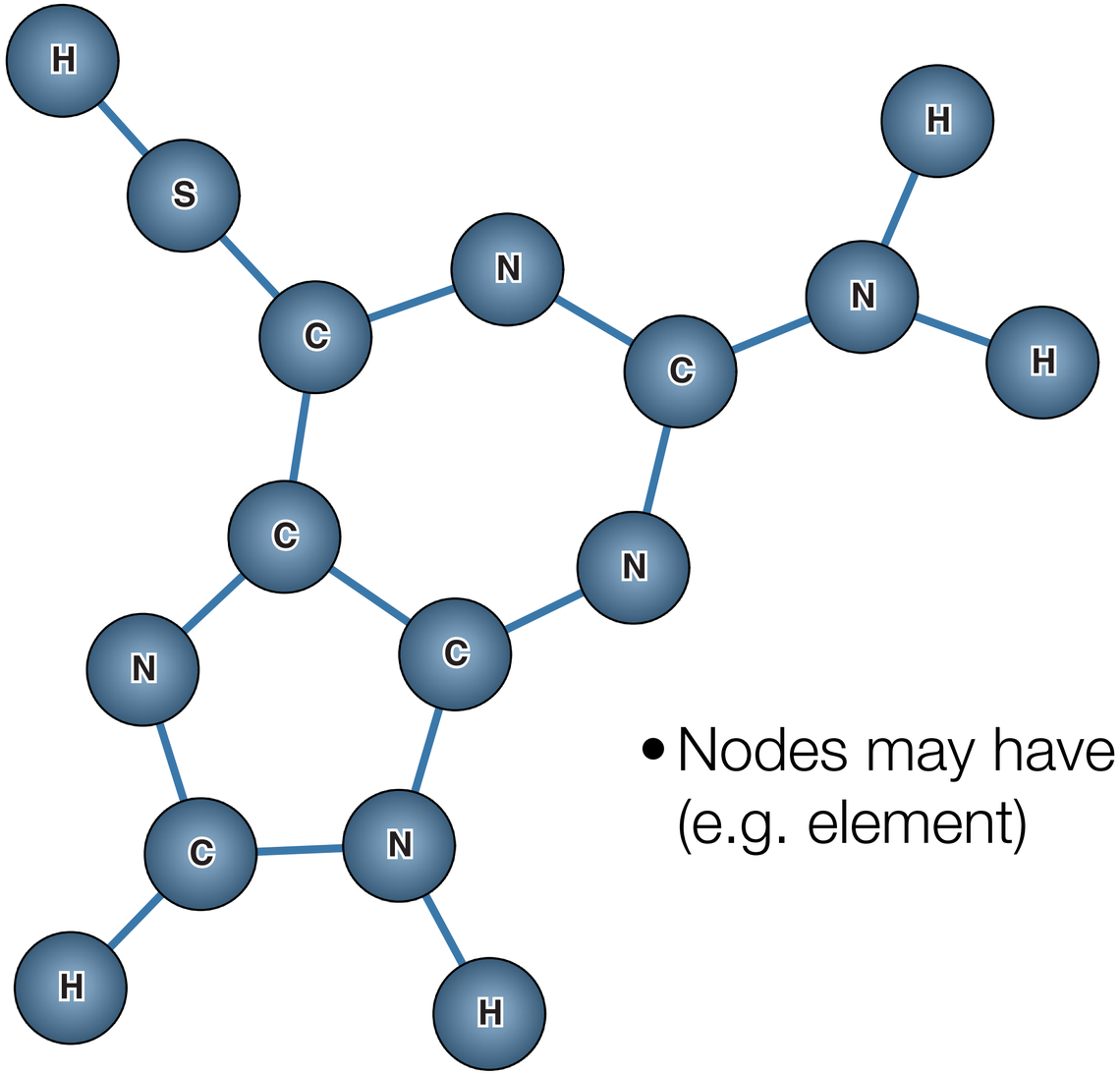


[Munzner (ill. Maguire), 2014]

Molecule Graph

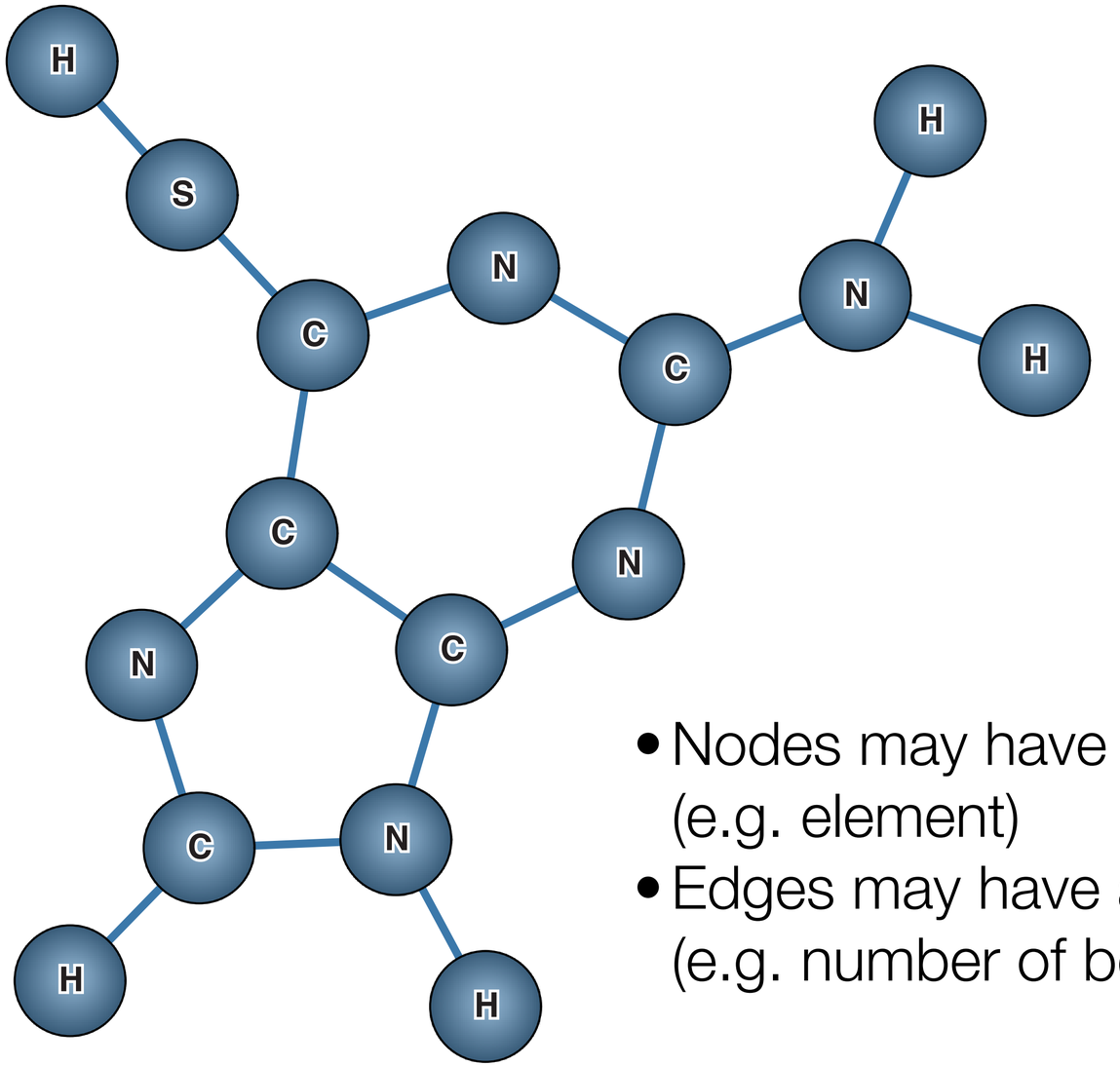


Molecule Graph



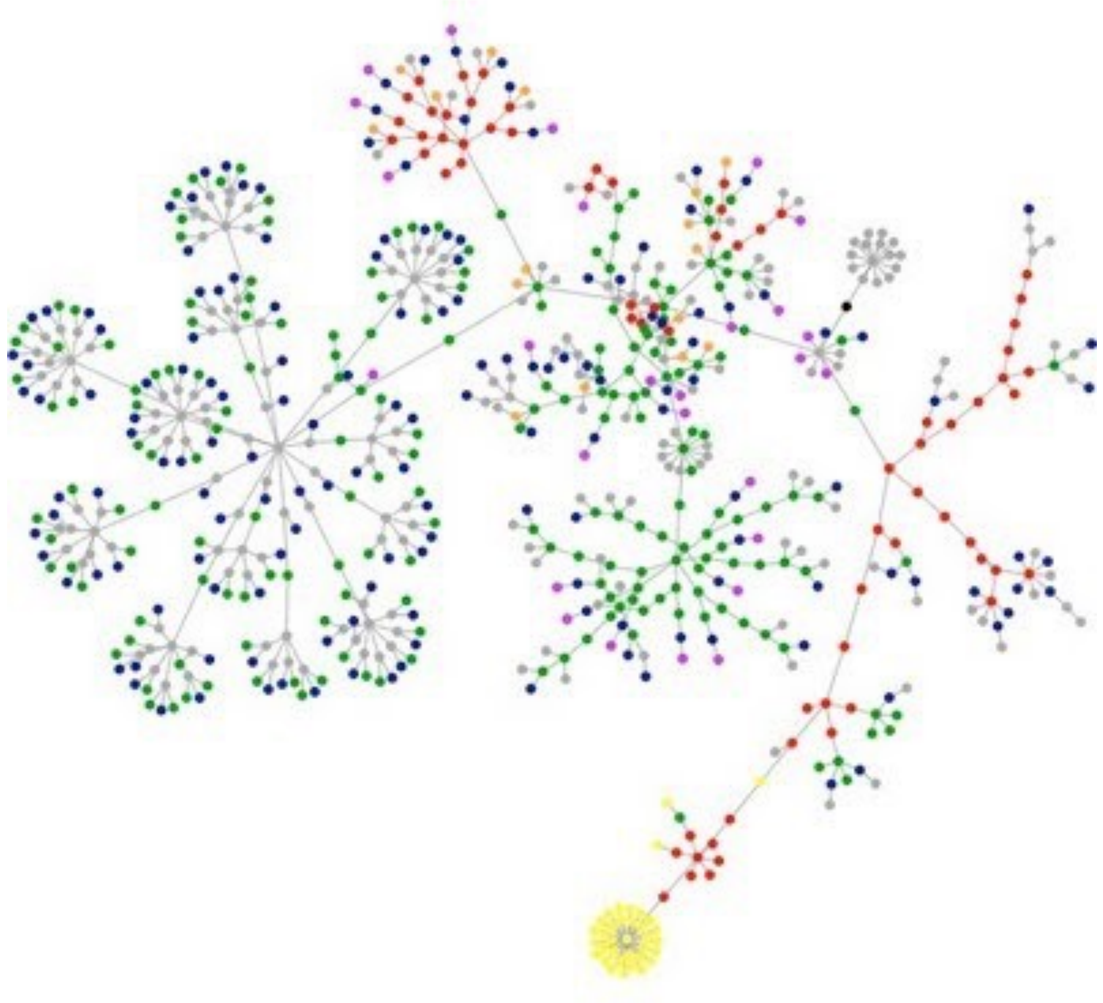
- Nodes may have attributes (e.g. element)

Molecule Graph



- Nodes may have attributes (e.g. element)
- Edges may have attributes (e.g. number of bonds)

Web Sites as Graphs (amazon.com)



[M. Salathe, 2006]

Social Networks

[P. Butler, 2010]

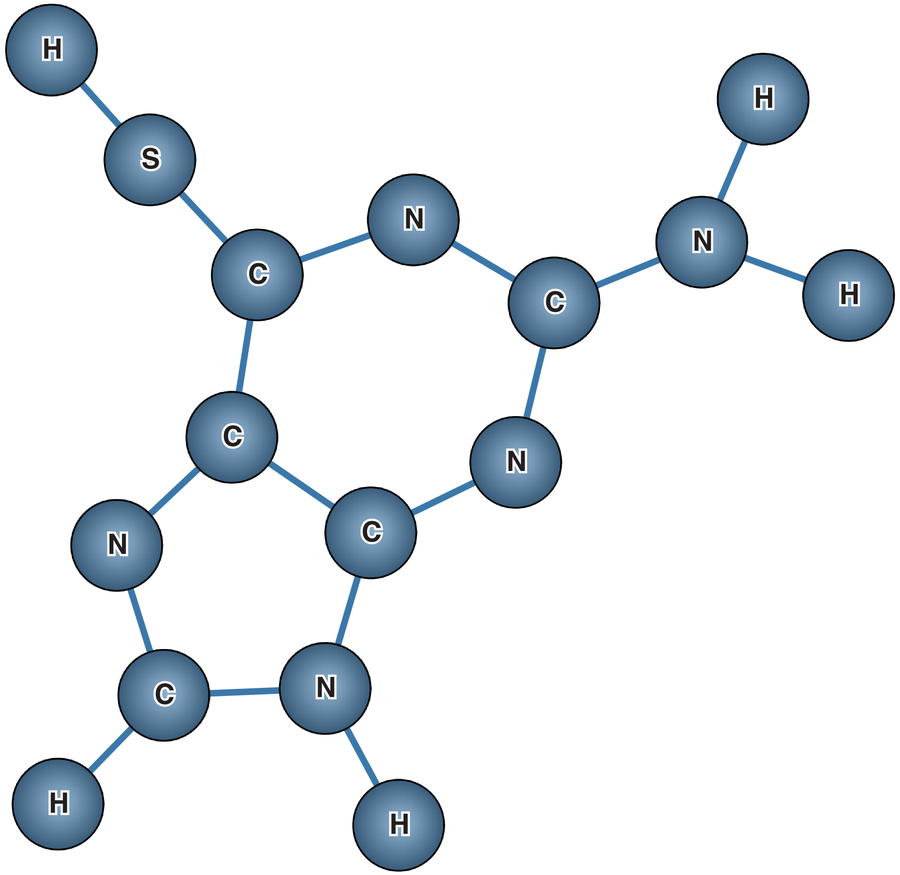
Networks as Data

Nodes

ID	Atom	Electrons	Protons
0	N	7	7
1	C	6	6
2	S	16	16
3	C	6	6
4	N	7	7

Edges

ID1	ID2	Bonds
0	1	1
1	2	1
1	3	2
3	4	1

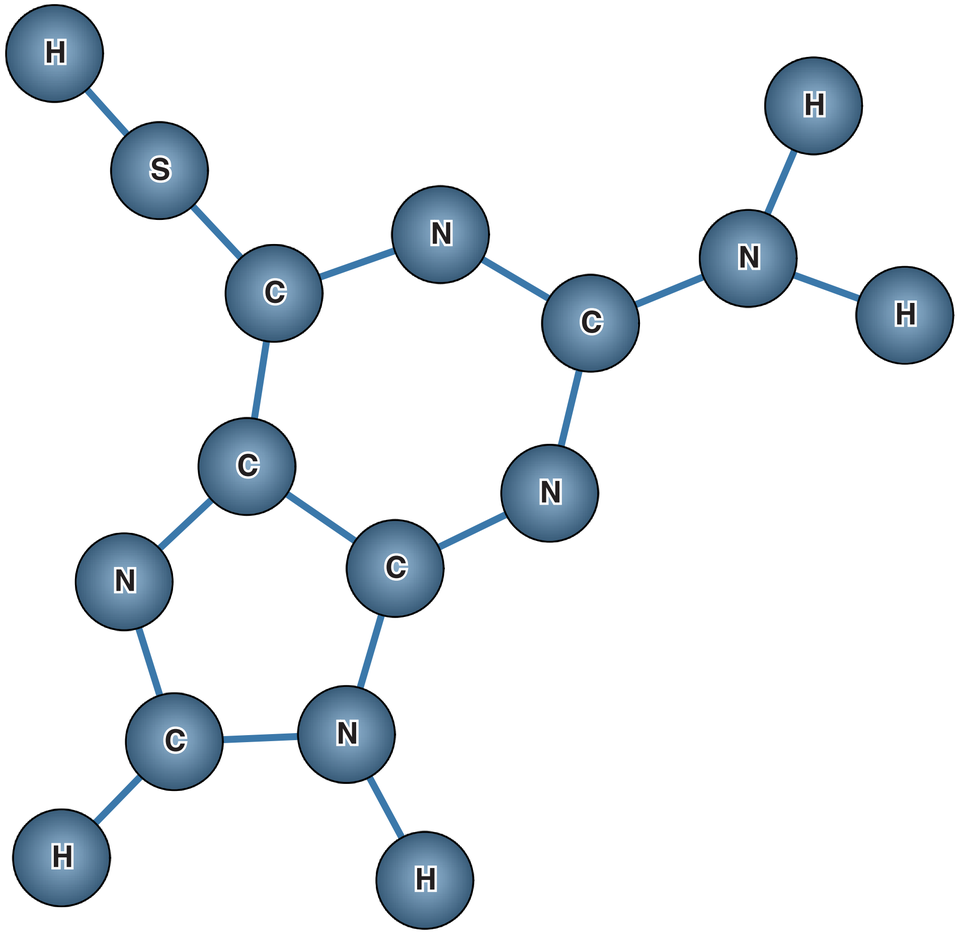


Node-Link Diagrams

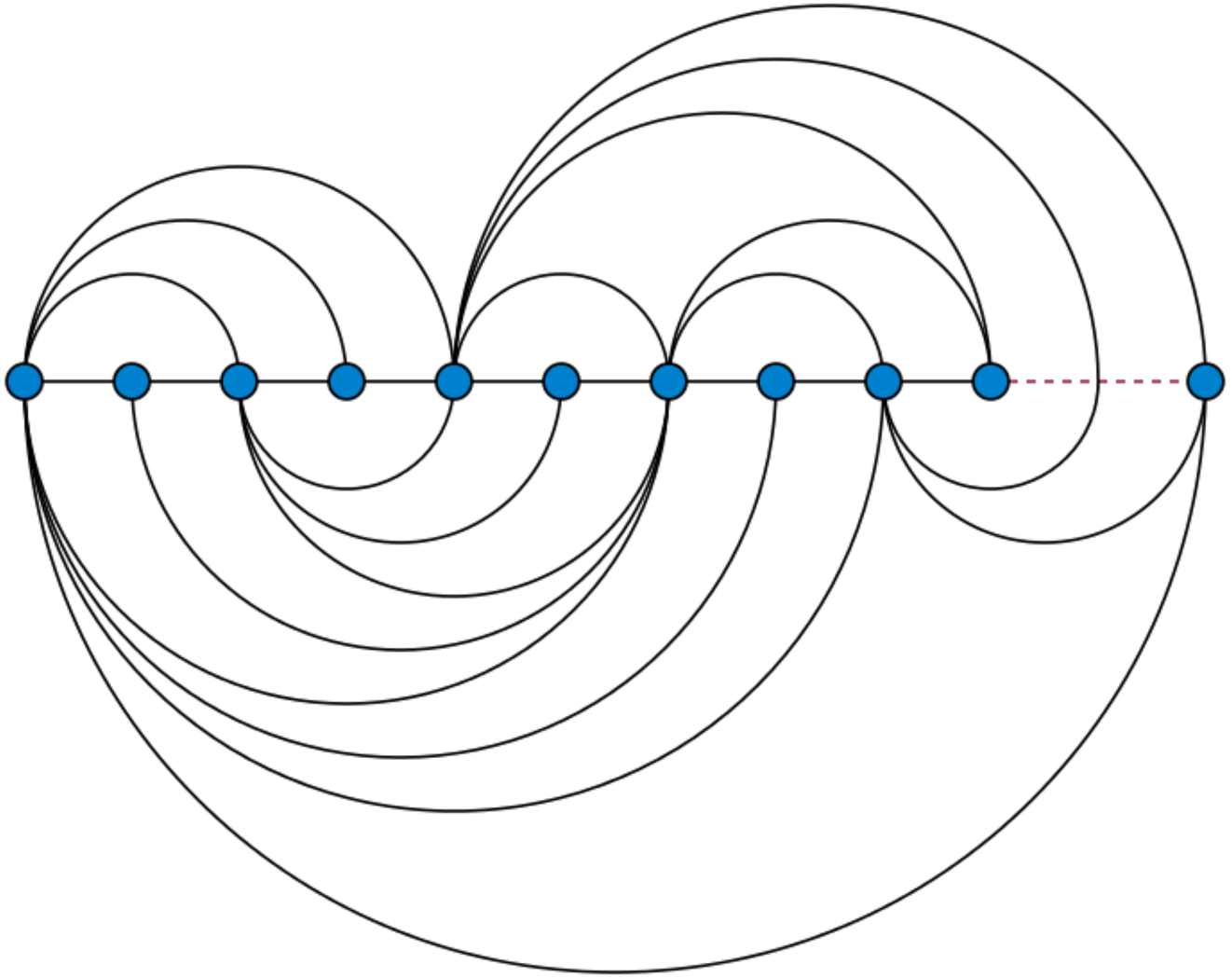
- Data: nodes and edges
- Task: understand connectivity, paths, structure (topology)
- Encoding: nodes as point marks, connections as line marks
- Scalability: hundreds

- …but high **density** of links can be problematic!

- Issue with the encoding?



Arc Diagram



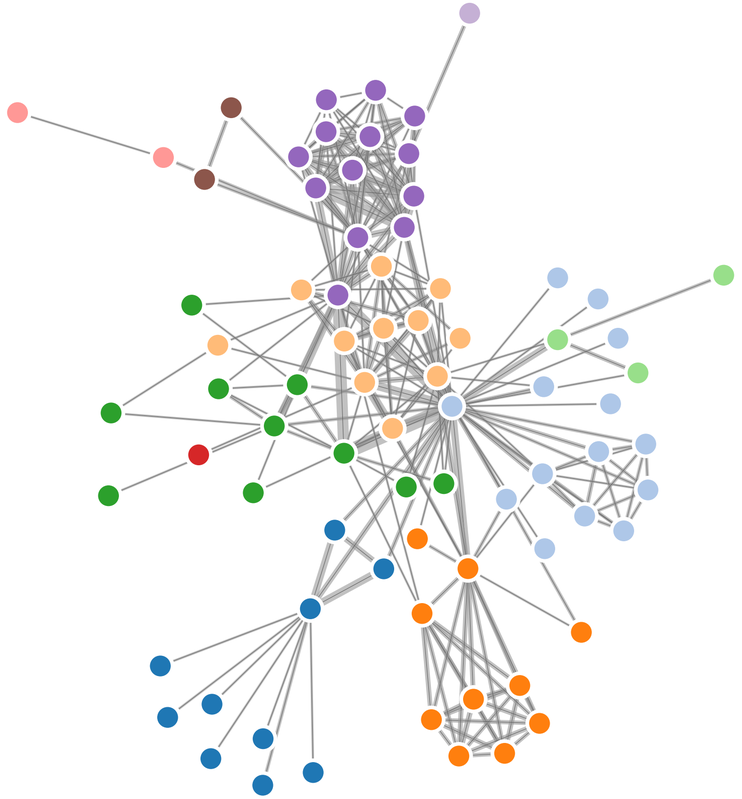
[D. Eppstein, 2013]

Network Layout

- Need to use spatial position when designing network visualizations
- Otherwise, nodes can **occlude** each other, links hard to distinguish
- How?
  - With bar charts, we could order using an attribute…
  - With networks, we want to be able to see connectivity and topology (not in the data usually)
- Possible metrics:
  - Edge crossings
  - Node overlaps
  - Total area

Force-Directed Layout

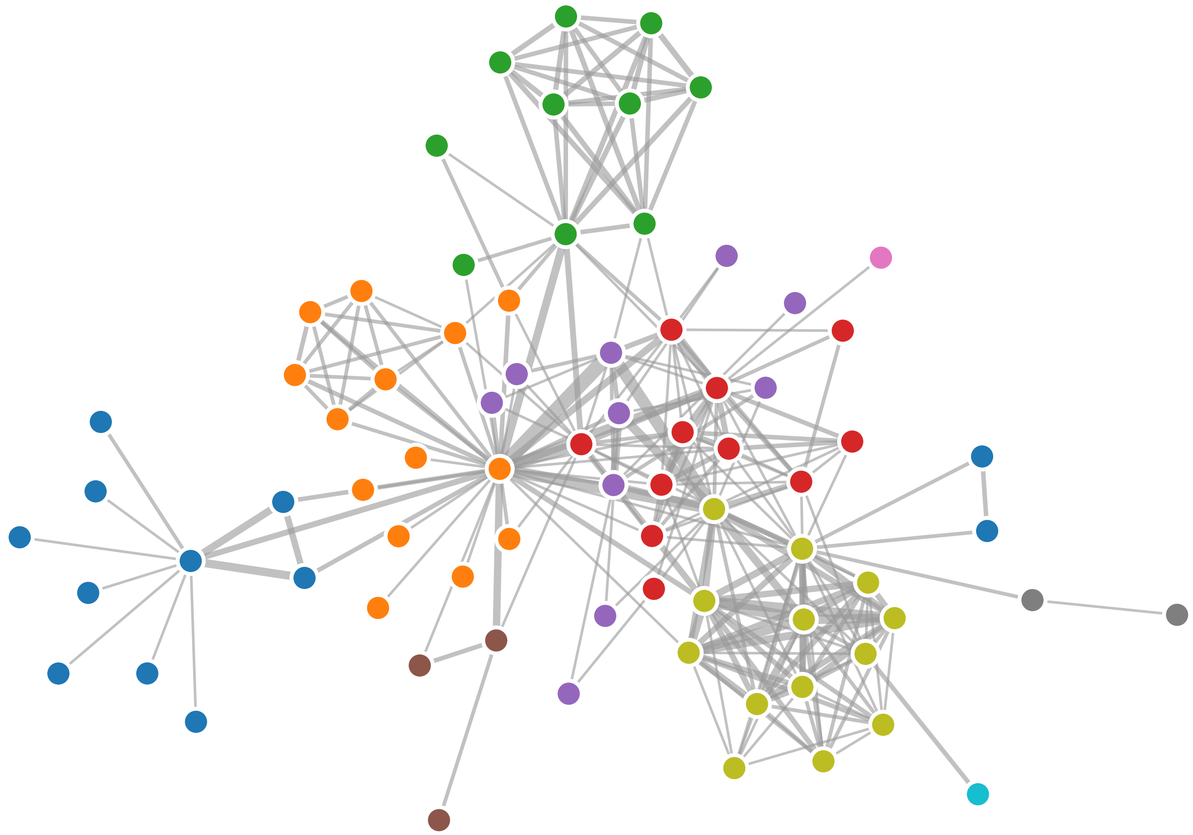
- Nodes push away from each other but edges are springs that pull them together
- Weakness: nondeterminism, algorithm may produce difference results each time it runs



[M. Bostock, 2017]

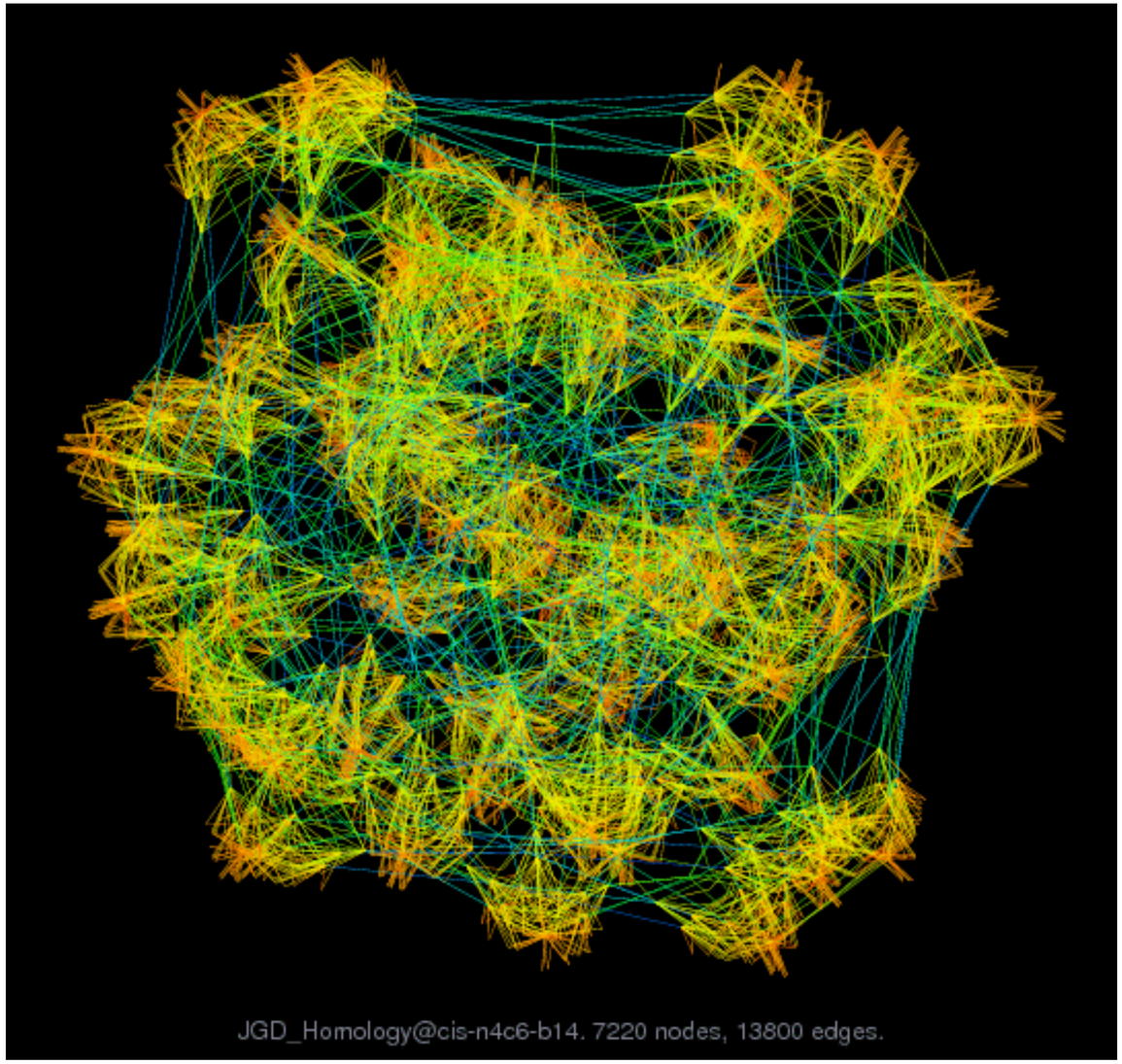
Constraint-Based Optimization (CoLa)

- Higher quality layout
- More **stable** in interactive applications (no "jitter")
- Allows user specified constraints such as alignments and grouping
- Can avoid overlapping nodes
- Provides flow layout for directed graphs
- May be **less scalable** to very large graphs
- Can route edges around nodes



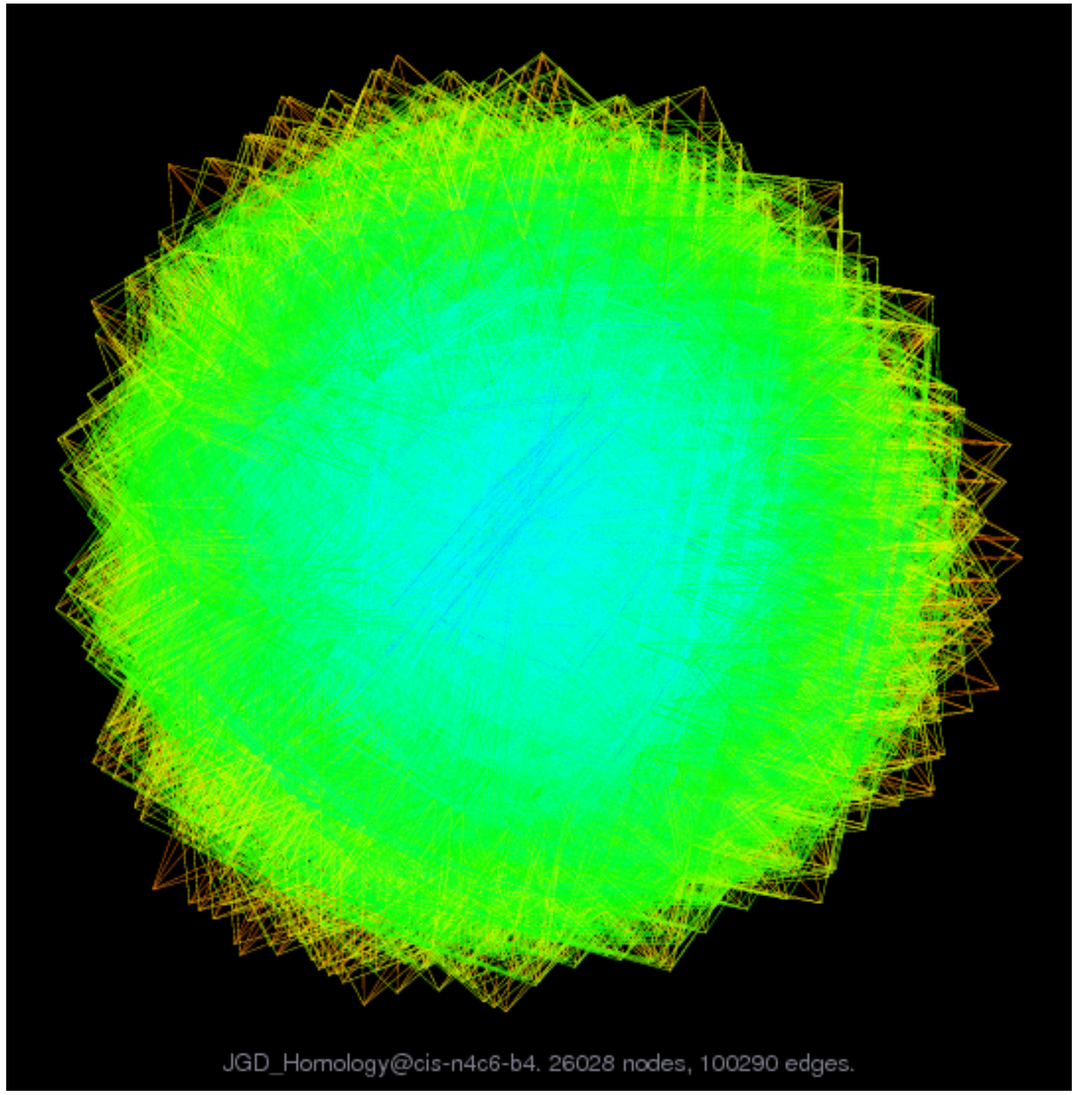
[T. Dwyer et al. (WebCoLa); M. Bostock (Example), 2018]

sfdp



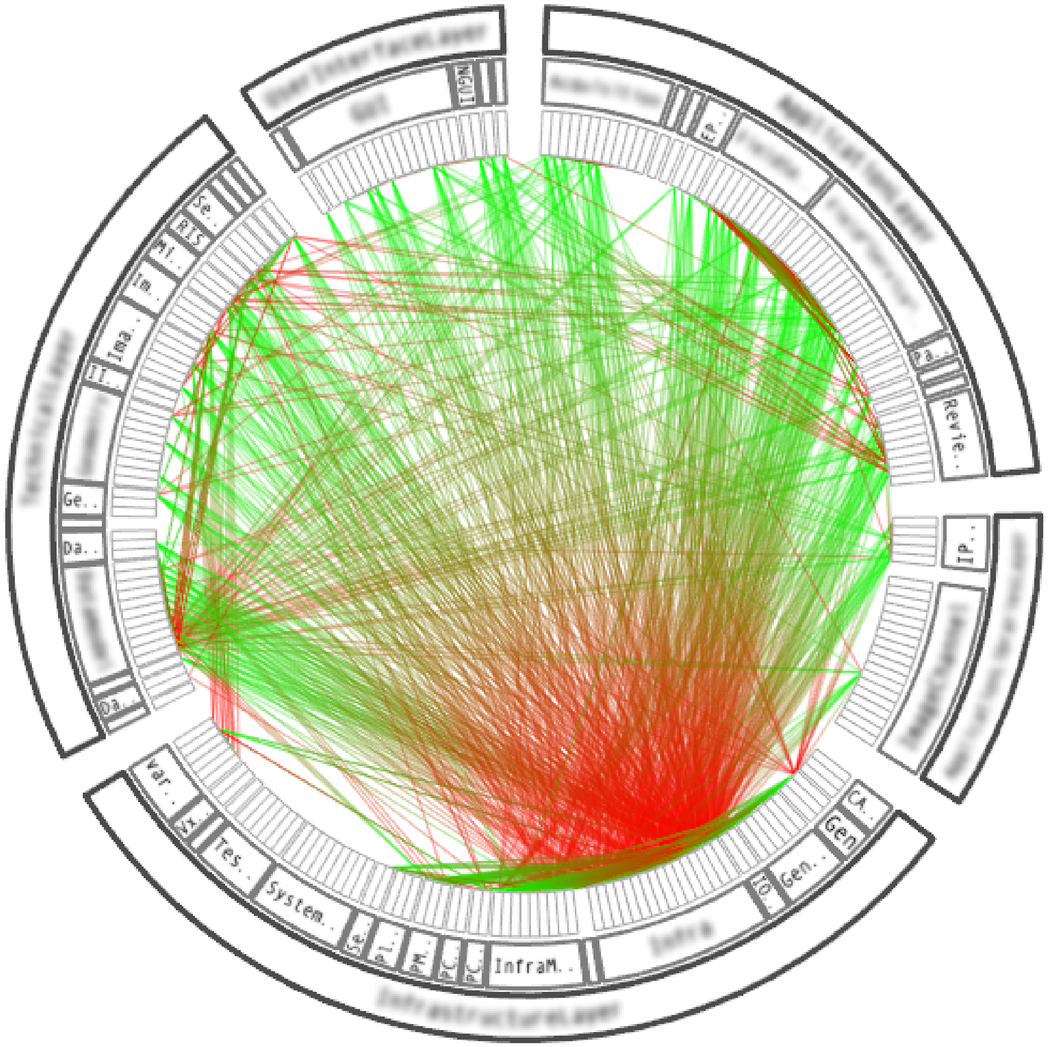
[Hu, 2005]

“Hairball”



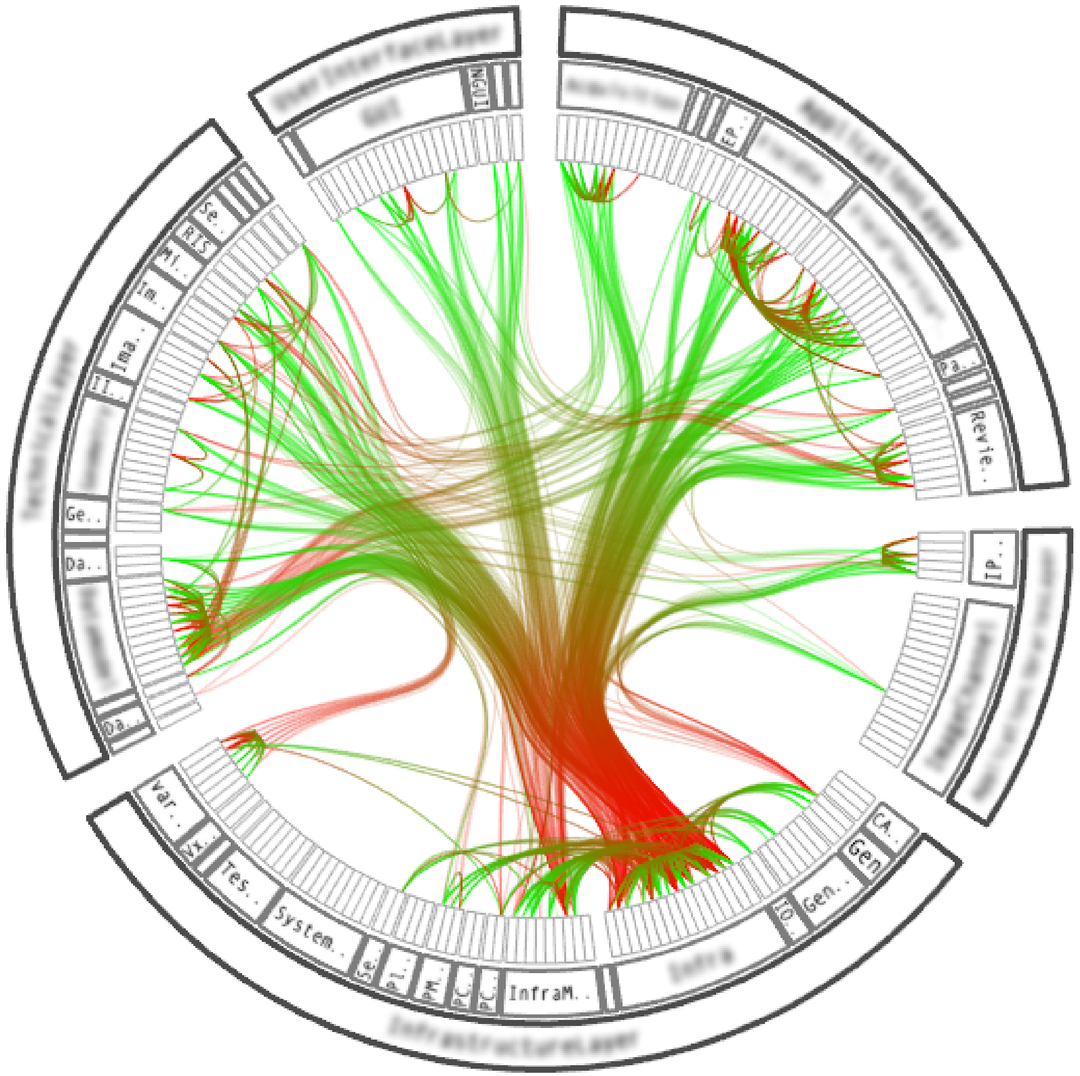
[Hu, 2014]

Hierarchical Edge Bundling



[Holten, 2006]

Hierarchical Edge Bundling



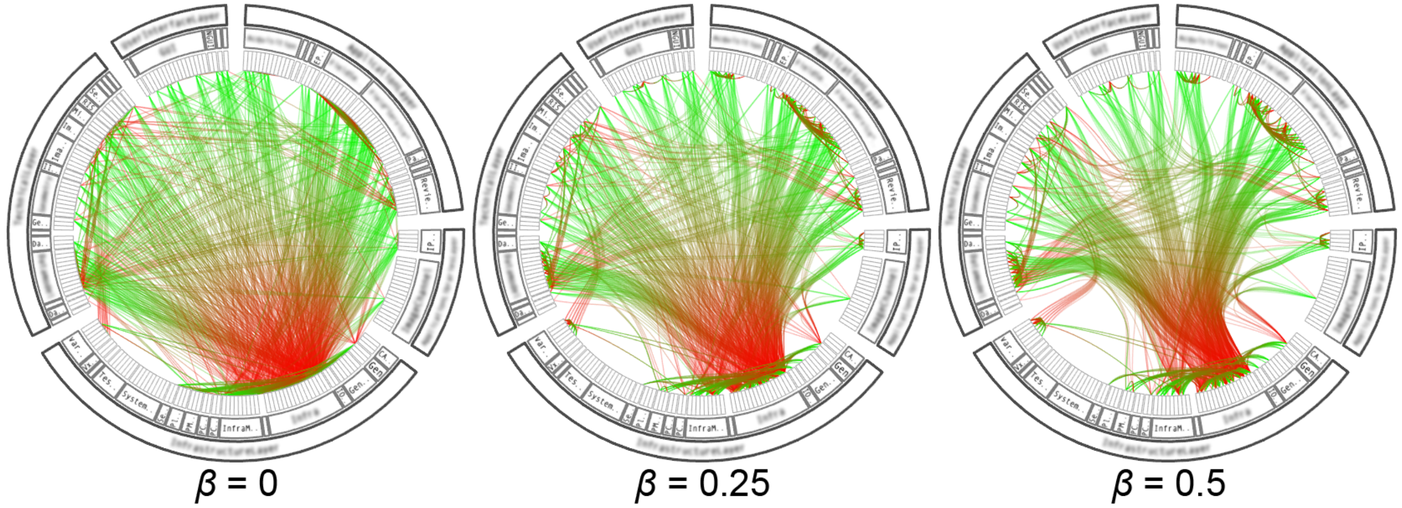
[Holten, 2006]

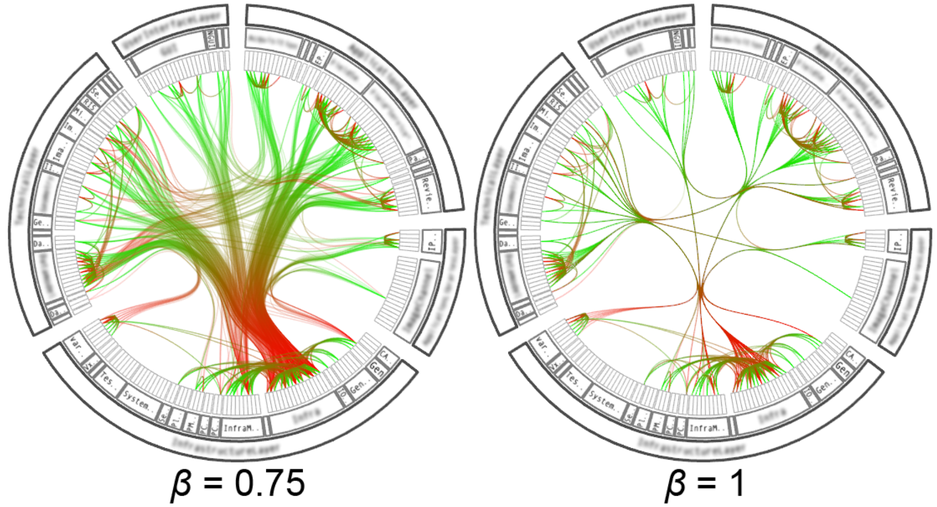
Hierarchical Edge Bundling

- Flexible and generic method
- Reduces visual clutter when dealing with large numbers of adjacency edges
- Provides an intuitive and continuous way to control the strength of bundling.
  - Low bundling strength mainly provides low-level, node-to-node connectivity information
  - High bundling strength provides high-level information as well by implicit visualization of adjacency edges between parent nodes that are the result of explicit adjacency edges between their respective child nodes

[Holten, 2006]

Bundling Strength



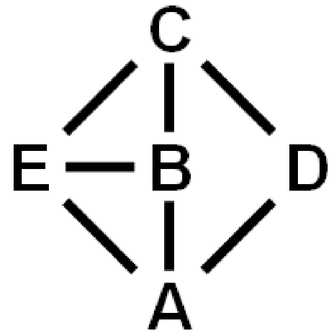


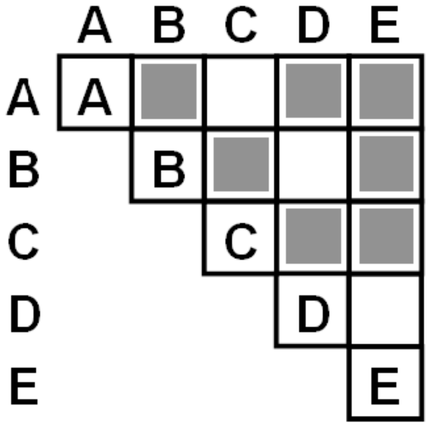
[Holten, 2006]

Adjacency Matrix

- Change network to tabular data and use a matrix representation
- Derived data: nodes are keys, edges are boolean values
- Task: lookup connections, find well-connected clusters
- Scalability: millions of edges

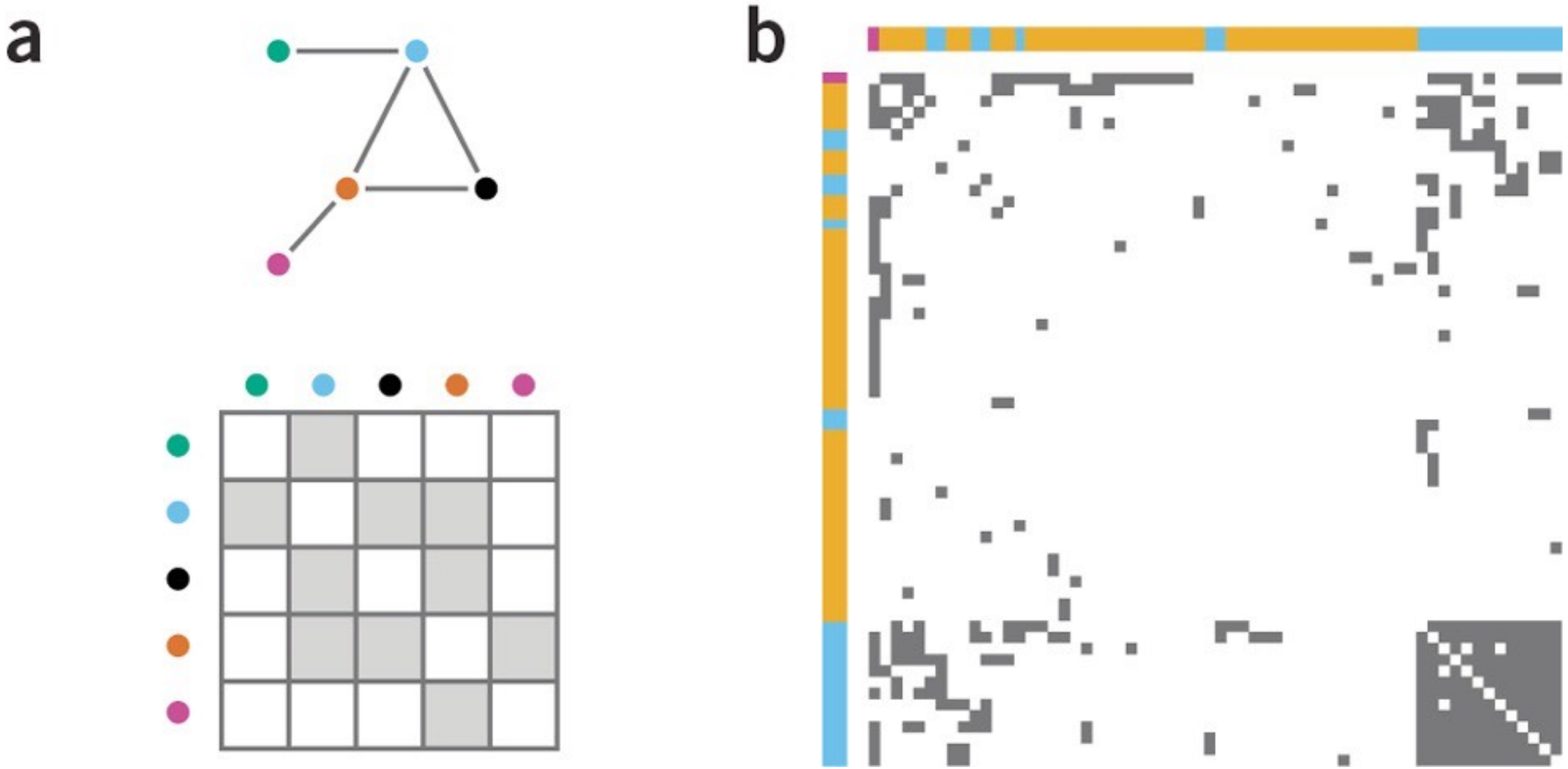
- Can encode **edge weight**, too





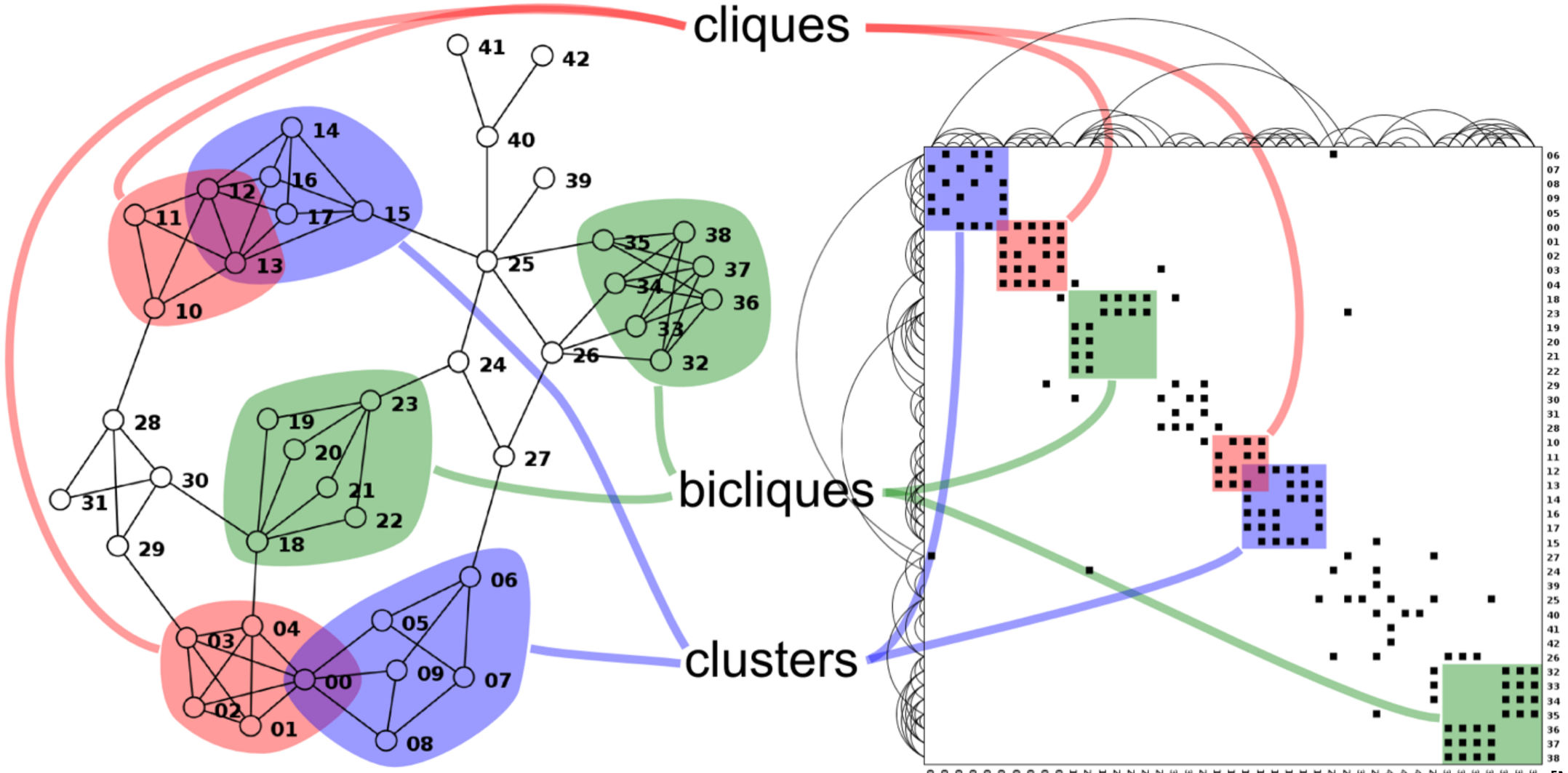
[Henry et al., 2007]

Cliques in Adjacency Matrices



[Gehlenborg and Wong]

Structures from Adjacency Matrices



[McGuffin]

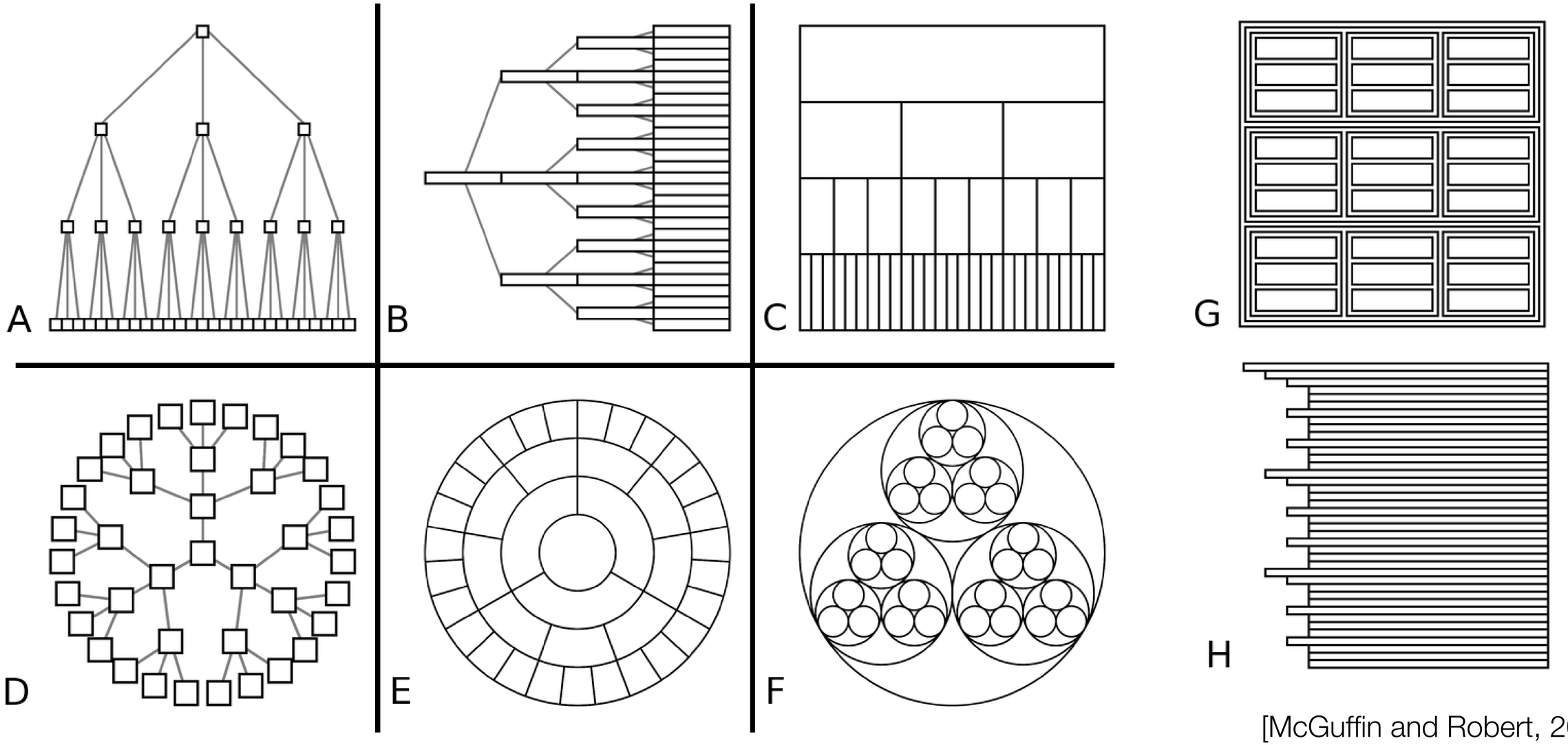
Node-Link or Adjacency Matrix?

- Empirical study: For most tasks, node-link is better for small graphs and adjacency better for large graphs
- Multi-link paths are hard with adjacency matrices
- Immediate connectivity or neighbors are ok, estimating size (nodes & edges also ok)
- People tend to be more familiar with node-link diagrams
- Link density is a problem with node-link but not with adjacency matrices

Trees

- Trees are directed acyclic **networks**
  - each edge has a direction: the origin is the parent, the destination is the child
  - cannot get back to a node after leaving it
- …plus each node has **at most one** parent node
- A tree has a **root** (every other node hangs off it)
- Can consider enclosure in trees using parent-child relationships

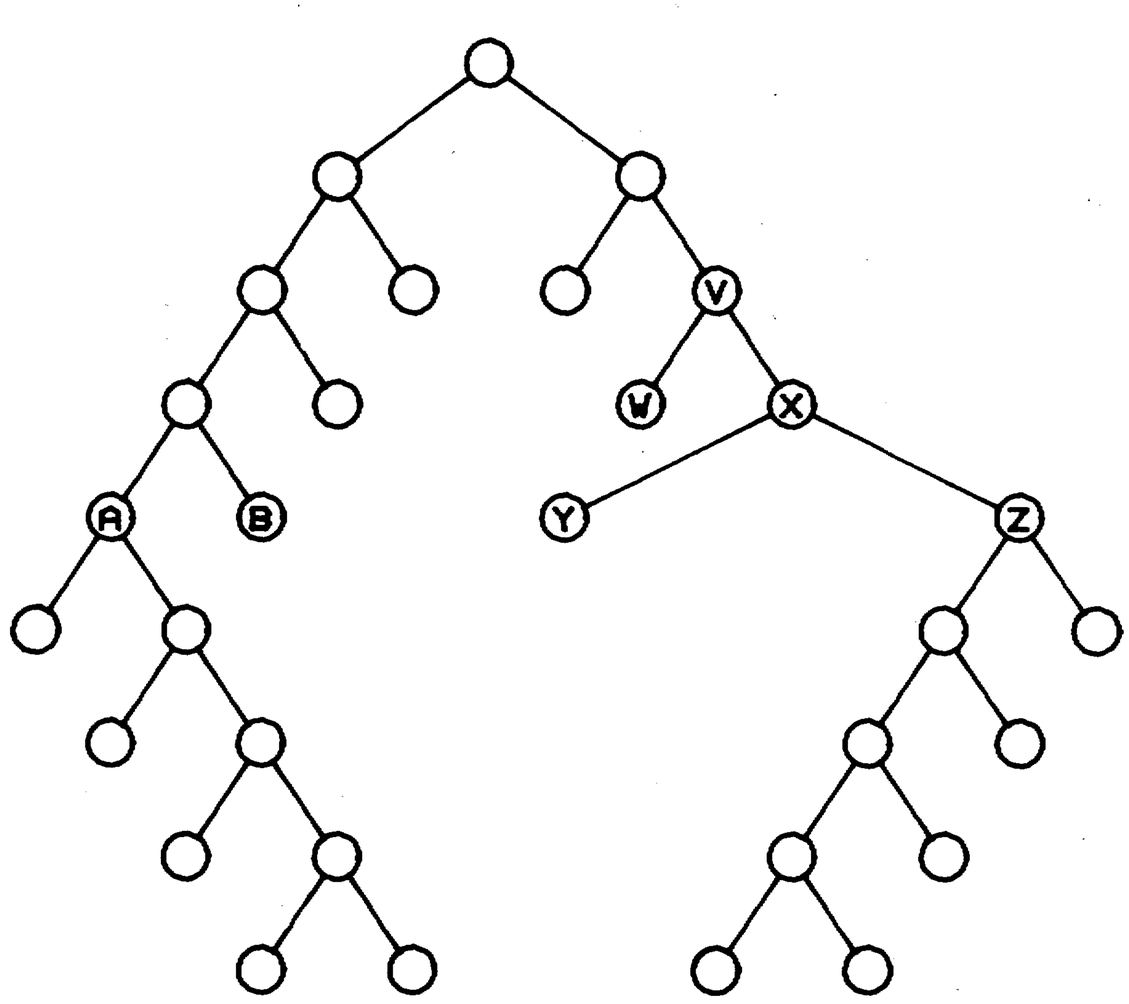
Tree Visualizations



[McGuffin and Robert, 2010]

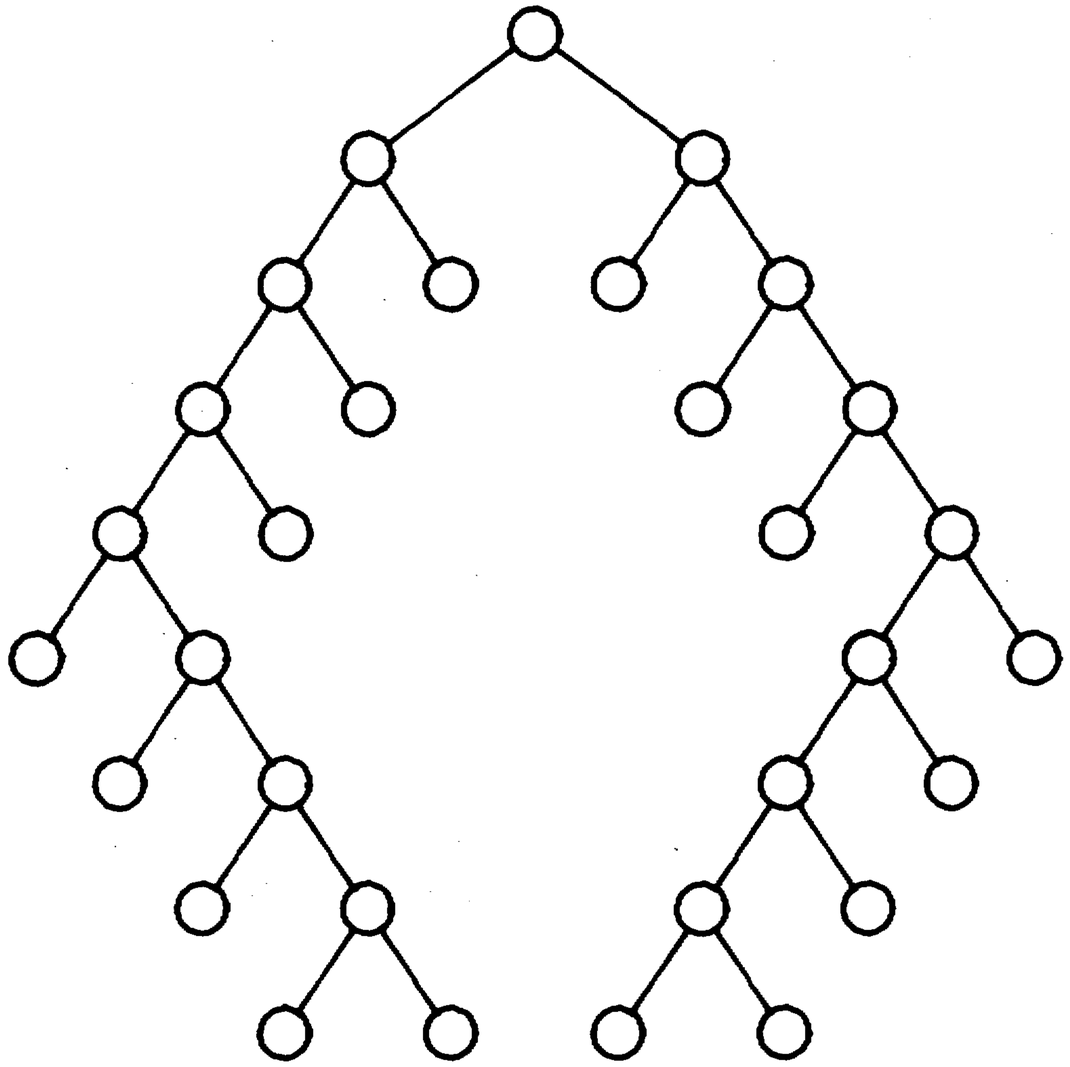
Node-Link Diagram

- Trees are networks
- …but we have more structure
- Horizontal or vertical
- Idea 1: partition space for each node via recursion
- Idea 2: “Tidy” Drawing
  - Wetherell & Shannon: Don’t waste space (overlapping parent nodes is ok)
  - Reingold and Tilford: Keep symmetry, subtrees look similar



[WS Alg., Reingold and Tilford, 1981]

Reingold-Tilford Algorithm

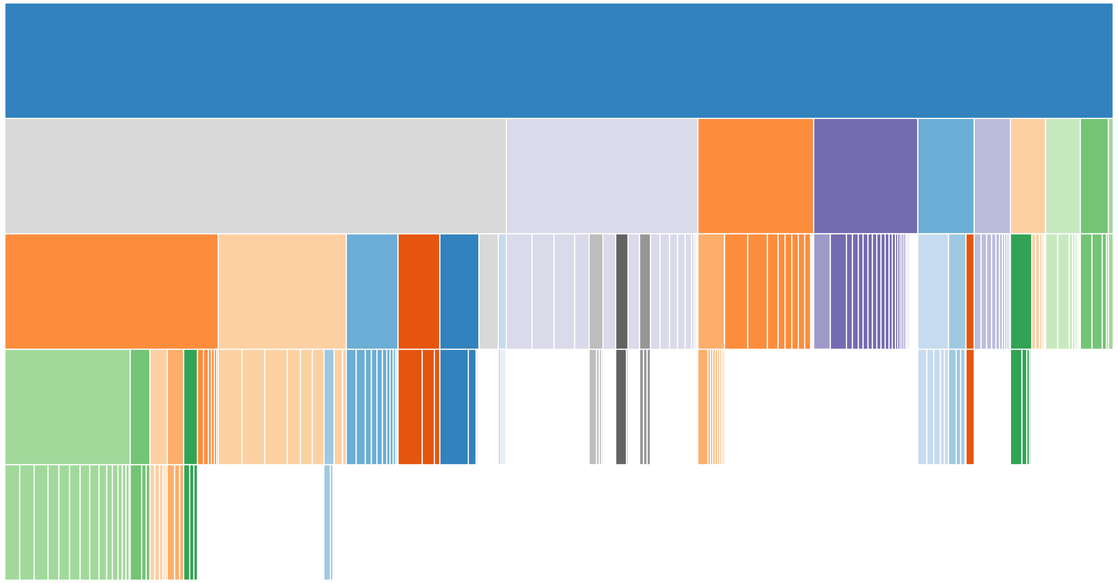


- Recurse on left and right subtrees
- Shift subtree over as long as it doesn't overlap
- Place parent centered above the subtrees
- Originally, only binary trees, extended by Walker

[Reingold and Tilford, 1981]

Icicle Plot

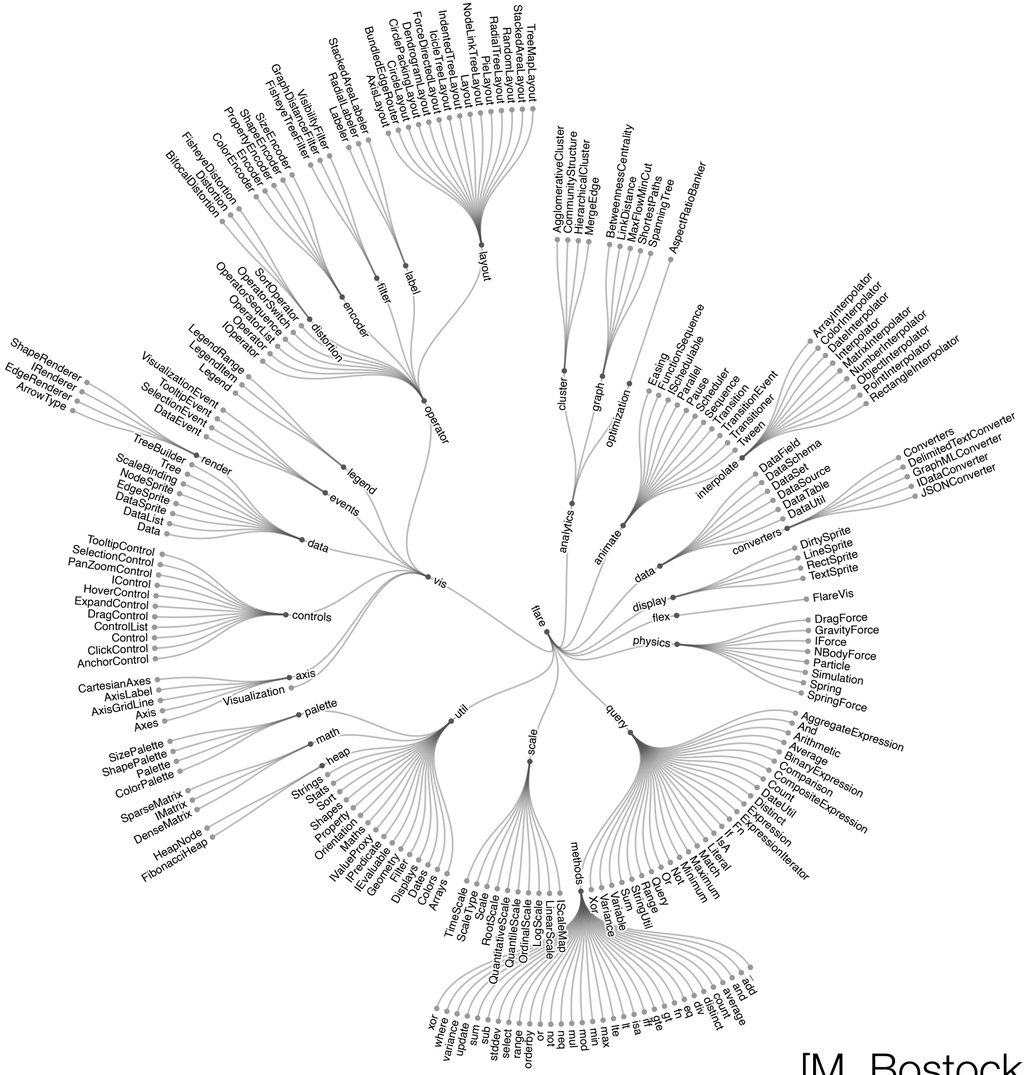
- Line marks
- Vertical position shows depth
- Horizontal position shows links and sibling order
- Scalability: 1 pixel leaves, but harder to label



[Bostock, 2011]

Radial Node-Link

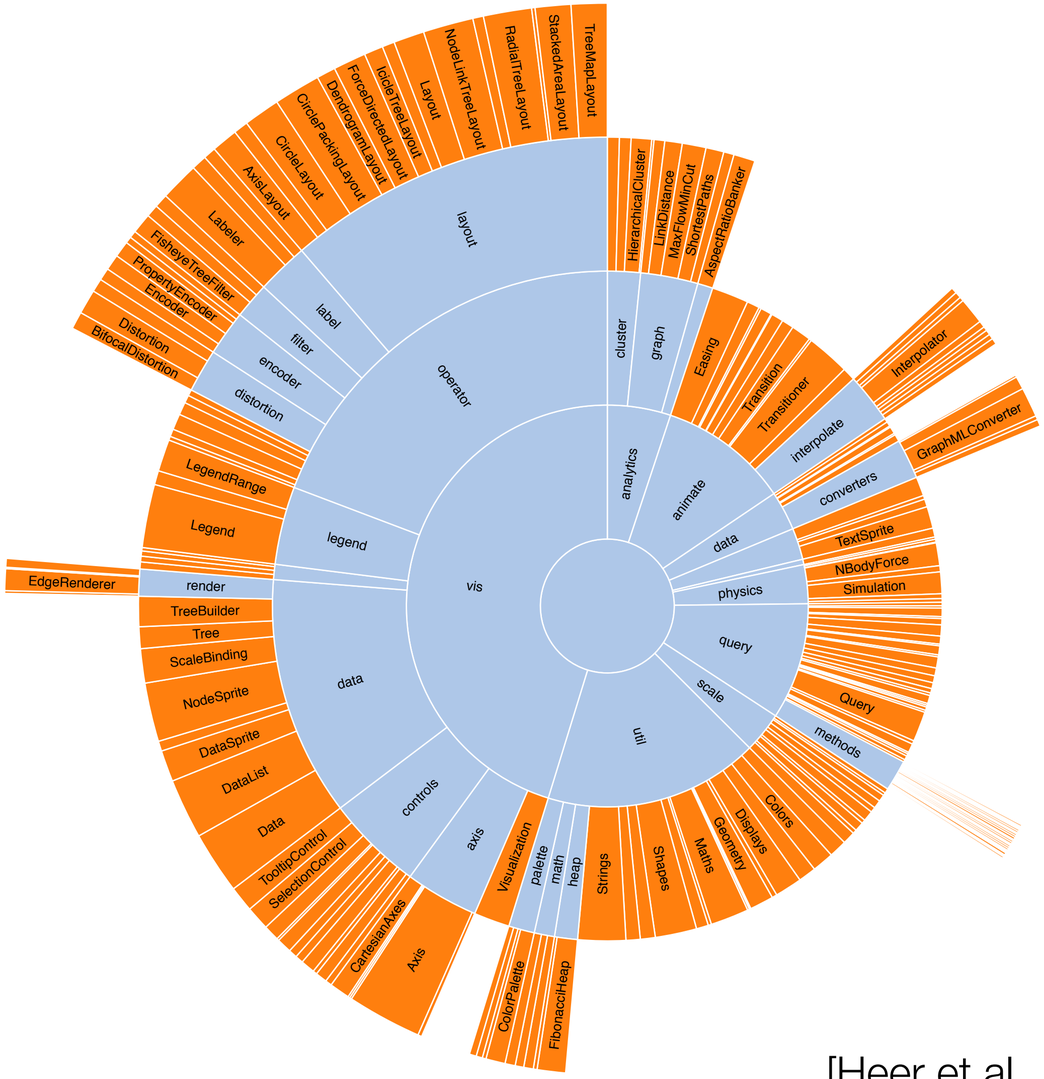
- Use polar coordinates instead of rectilinear
- Same layout algorithms work (e.g. Reingold-Tilford)
- Benefit: space usage, labels



[M. Bostock, 2017]

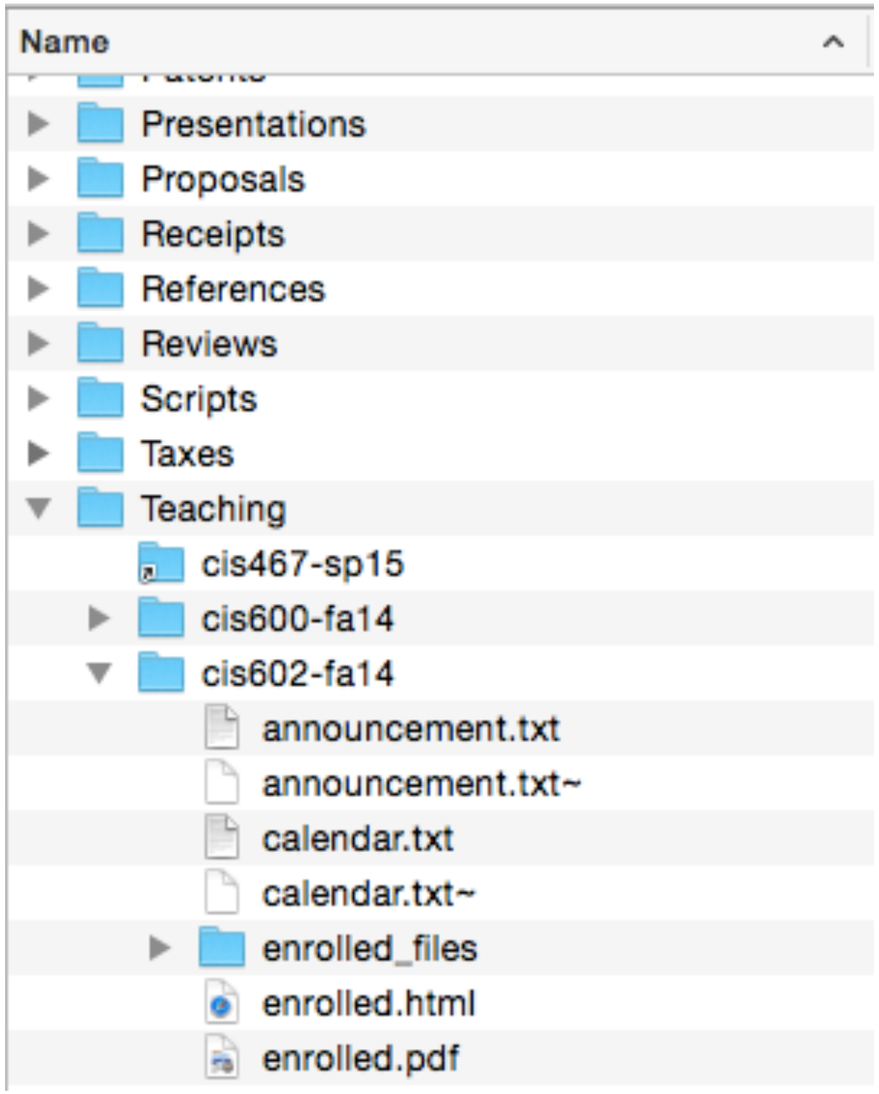
Sunburst

- Icicle plot in a radial layout
- Reading labels?
- Intuitive navigation



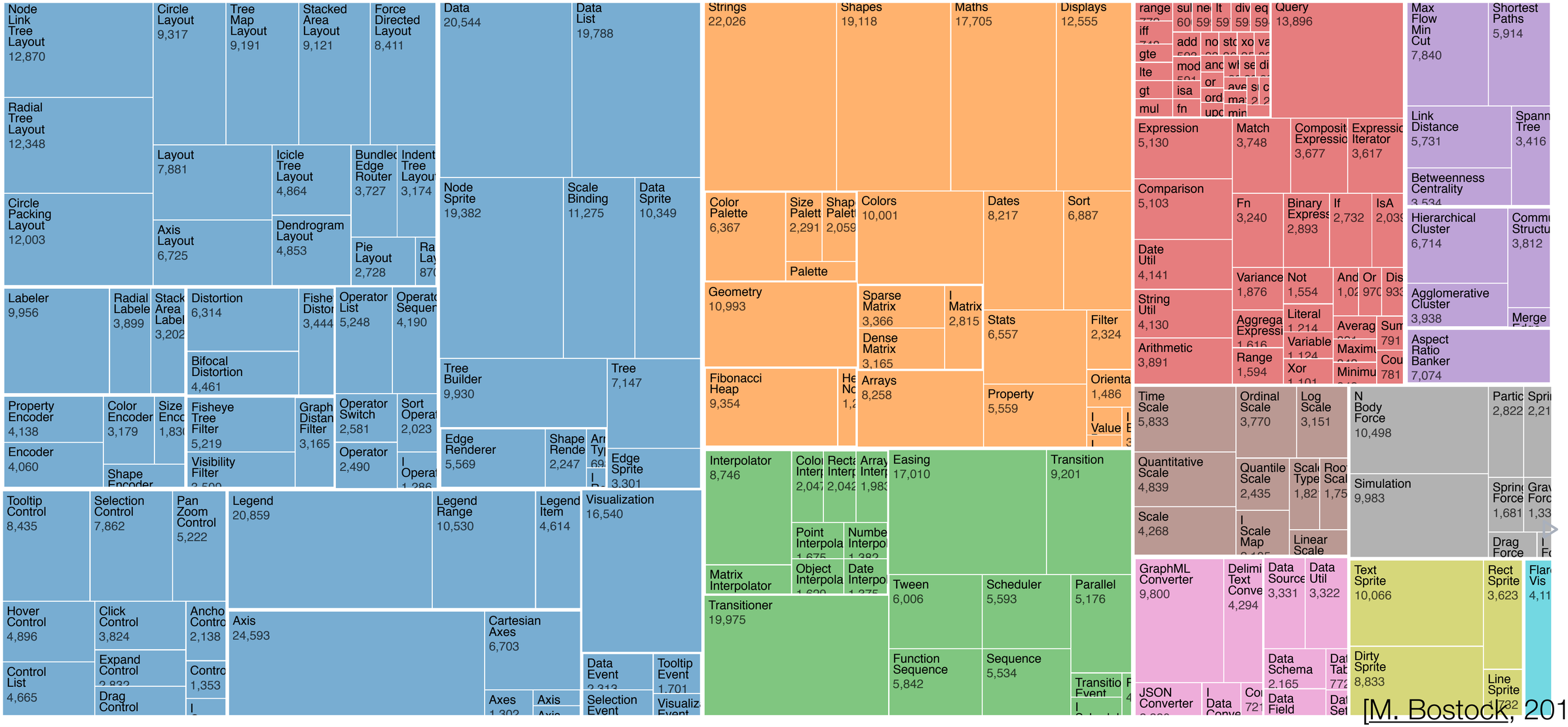
[Heer et al., 2012]

Indented Outline



- Like a filesystem tree
- Use horizontal position to show depth, vertical positions show sibling/order

Treemap



[M. Bostock, 2017]