G53FIV: Fundamentals of Information Visualization

Lecture 12: Visualizing Time Series, Trees and Graphs

Ke Zhou
School of Computer Science
Ke.Zhou@nottingham.ac.uk

https://moodle.nottingham.ac.uk/course/view.php?id=68644

Overview

- Visualizing Time Series Data
- Trees and Graphs

Visualizing Time Series Data

Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Traditional Time Series Visualization

NVIDIA stock vs. NASDAQ (from Yahoo! finance)

Challenges

- Temporal relationships can be highly complex
  - temporal ordering is a serious issue
  - event may occur in spatially disjoint locations
  - what came before what – cause and effect
  - what time shifts are acceptable/plausible?
- To understand temporal relationships, an analyst
  - might need to reread the paragraph many times
  - needs to cognitively make inferences between pieces of information

Tasks

- Often asked questions:
  - when was something greatest/least?
  - is there a pattern? are two series similar?
  - does a data element exist at time t, and when?
  - how long does a data element exist and how often?
  - how fast are data elements changing
  - in what order do data elements appear?
  - do data elements exist together?

(Optional Reading) Müller, Wolfgang, and Heidrun Schumann. "Visualization for modeling and simulation: visualization methods for time-dependent data-an overview." Proceedings of the 35th conference on Winter simulation: driving innovation. Winter Simulation Conference, 2003.

Taxonomy

Time	**Temporal primitives**	time points (a) (b) (c) (d) (e) (f) (g) (i)		time intervals (g) (h)
	Structure of time	linear (a) (b) (c) (d) (f) (g) (h) (i)	cyclic (e)	branching (h)
Data	**Frame of reference**	abstract (c) (d) (f) (g) (h) (i)		spatial (a) (b) (e) (i)
	Number of variables	univariate (a) (b) (f) (g) (h)		multivariate (c) (d) (e) (i)
	Level of abstraction	data (a) (b) (c) (d) (e) (f) (g) (h) (i)		data abstractions (b) (g) (i)
Representation	**Time dependency**	static (c) (d) (e) (g) (h) (i)		dynamic (a) (b) (f) (i)
	Dimensionality	2D (a) (c) (d) (g) (h) (i)		3D (b) (e) (f) (i)

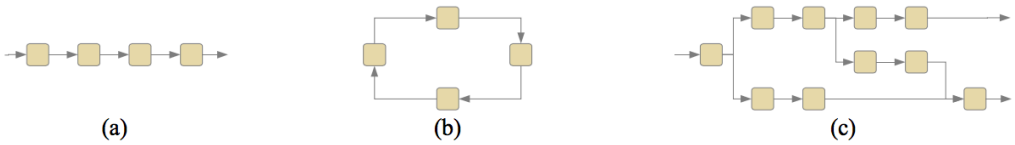
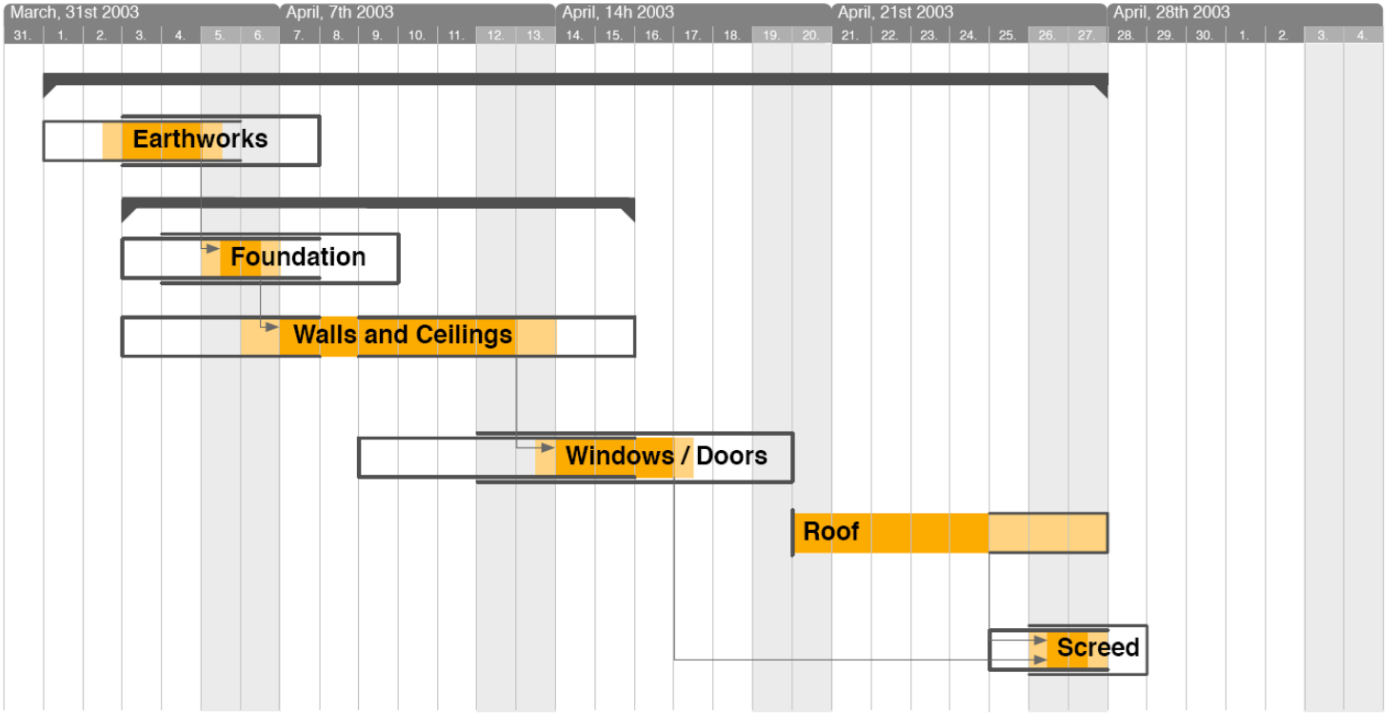


Fig. 2. Structure of time: (a) Linear time; (b) Cyclic time; (c) Branching time.

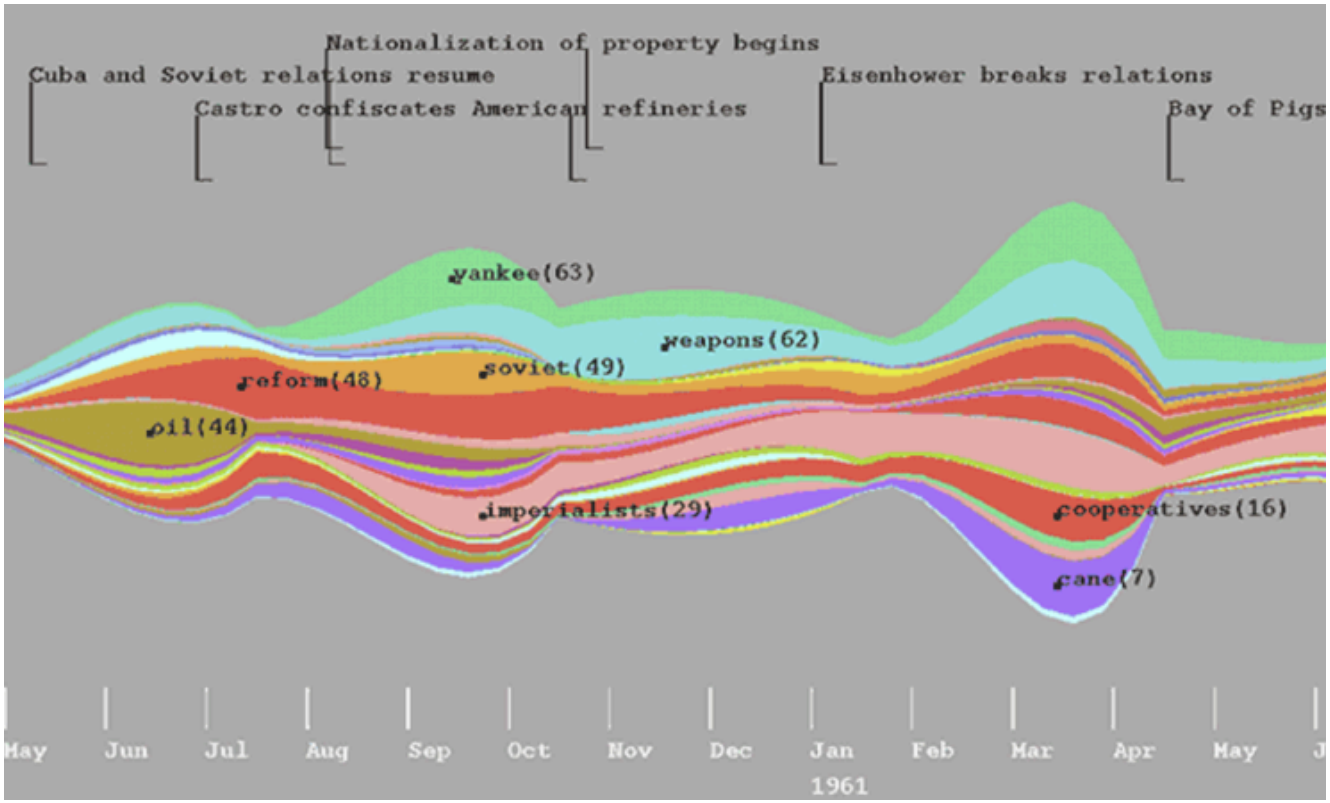
Aigner, Wolfgang, et al. "Visualizing time-oriented data - a systematic view." Computers & Graphics 31.3 (2007): 401-409.

Gantt Chart



Time	Temporal primitives	Intervals
	Structure of time	Linear, Branching
Data	Frame of reference	Abstract
	Number of variables	Univariate
	Level of abstraction	Data
Representation	Time dependency	Static
	Dimensionality	2D

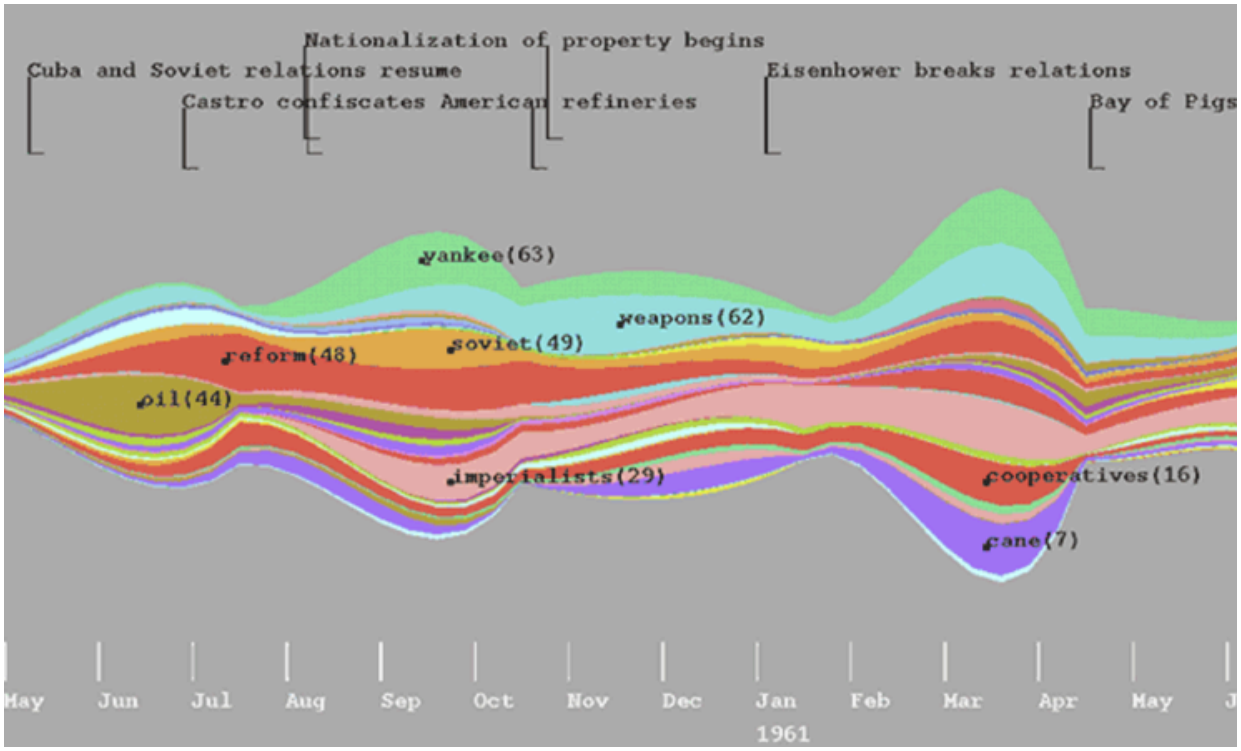
Theme River
(Stream Graphs / Stacked Area Charts)



River widens or narrows to depict changes in the collective strength of selected themes in the underlying documents. Individual themes are represented as colored "currents" flowing within the river (example: Cuban Missile crisis) .

(Optional) Havre, Susan, et al. "Themeriver: Visualizing thematic changes in large document collections." IEEE transactions on visualization and computer graphics 8.1 (2002): 9-20.

Theme River



Time	Temporal primitives	Time Points
	Structure of time	Linear
Data	Frame of reference	Abstract
	Number of variables	Multivariate
	Level of abstraction	Data
Representation	Time dependency	Static
	Dimensionality	2D

Havre, Susan, et al. "Themeriver: Visualizing thematic changes in large document collections." IEEE transactions on visualization and computer graphics 8.1 (2002): 9-20.

Cyclic Patterns

- Time data are often cyclic
  - Spiral displays are good to bring out cyclic patterns
  - One period per loop (for example, a year)

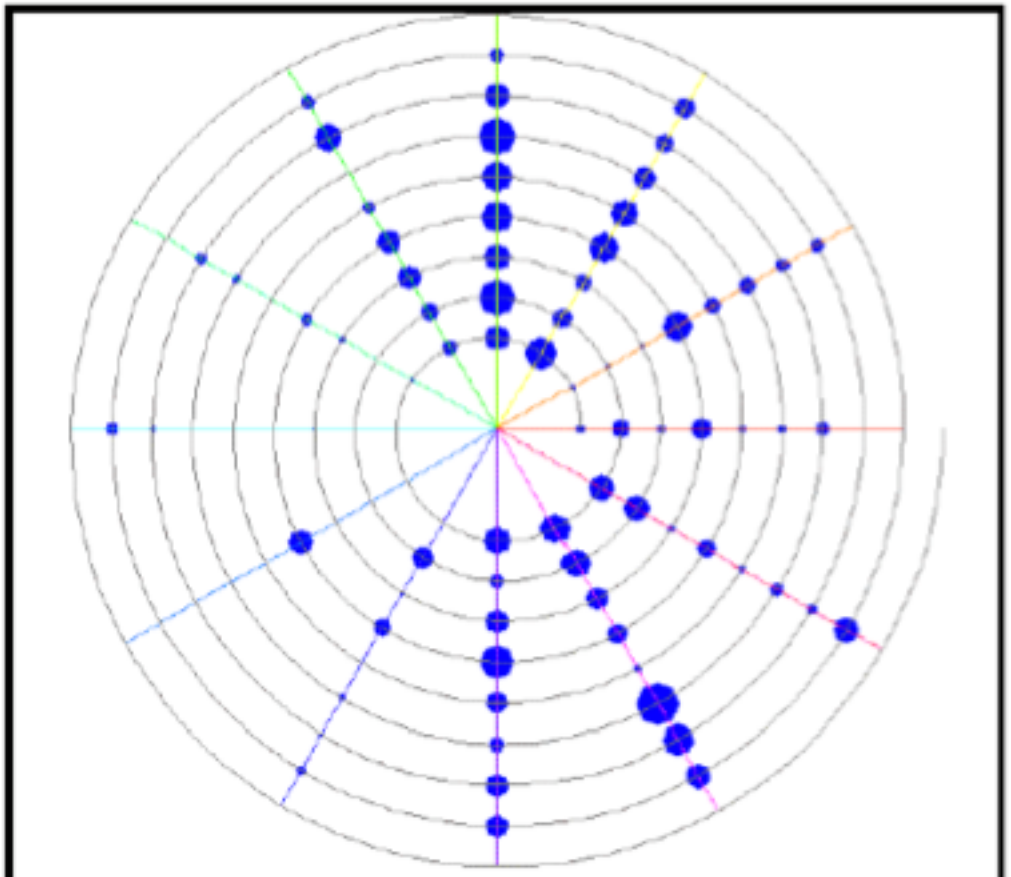


Figure 2. An indented spiral, with spokes, showing monthly consumption percentages for Baphia Capparidifolia during the period 1980 – 1988.

Carlis & Konstan UIST '98

Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Cyclic Patterns

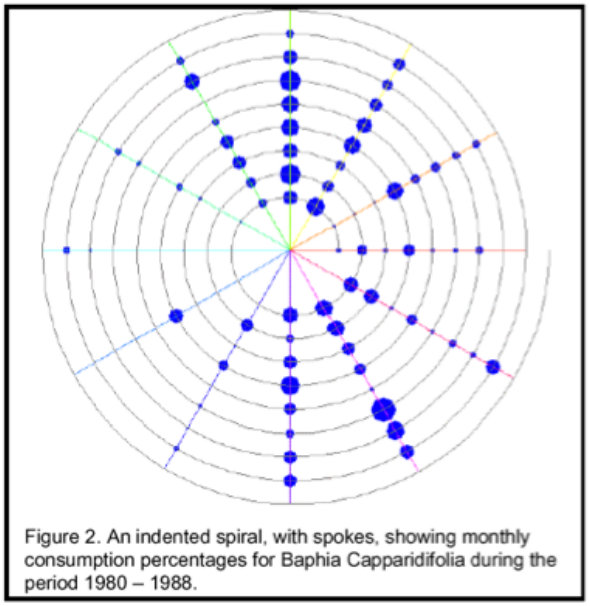


Figure 2. An indented spiral, with spokes, showing monthly consumption percentages for Baphia Capparidifolia during the period 1980 – 1988.

Time	**Temporal primitives**	**Time Points**
	Structure of time	**Cyclic**

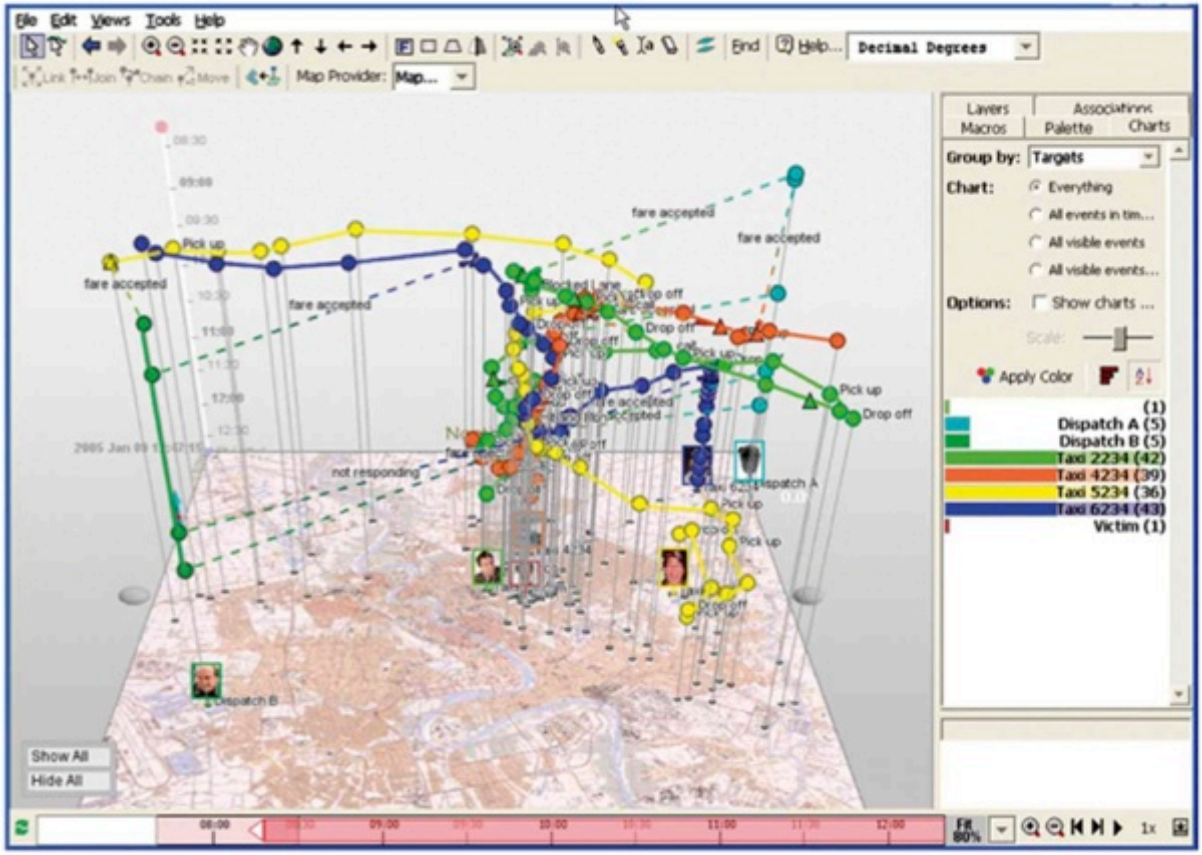
Data	**Frame of reference**	**Abstract**
	Number of variables	**Univariate**
	Level of abstraction	**Data**

Representation	**Time dependency**	**Static**
	Dimensionality	**2D**

Carlis & Konstan UIST '98

Combining Space and Time

- OculusInfo Geotime application
  - events are represented in an X,Y,T coordinate space
  - the X,Y plane shows geography
  - the vertical T axis represents time
  - events animate in time vertically through the 3-D space as the time slider bar is moved.

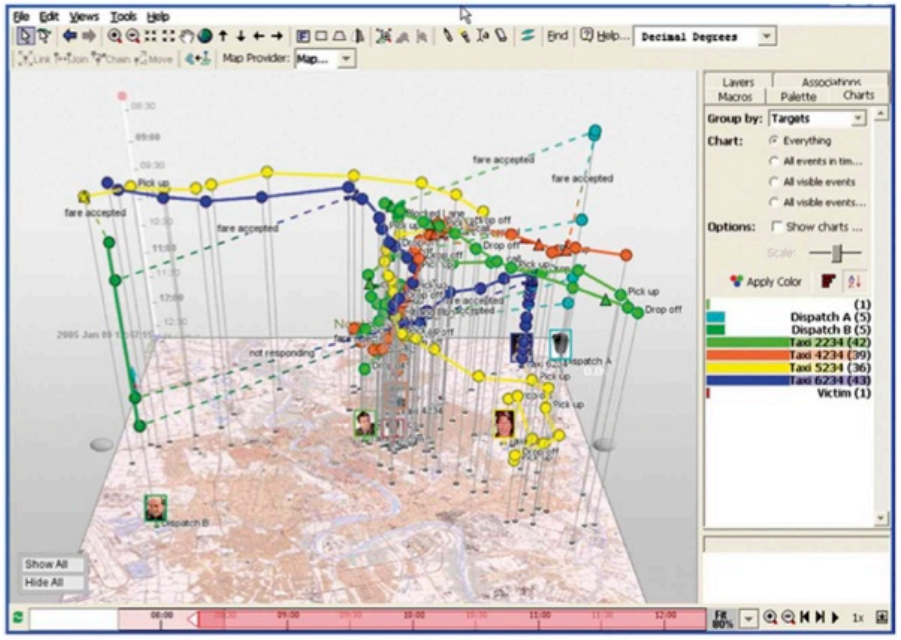


Video: https://www.youtube.com/watch?v=P9-lpE47oWc

Kapler & Wright InfoVis '04

Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

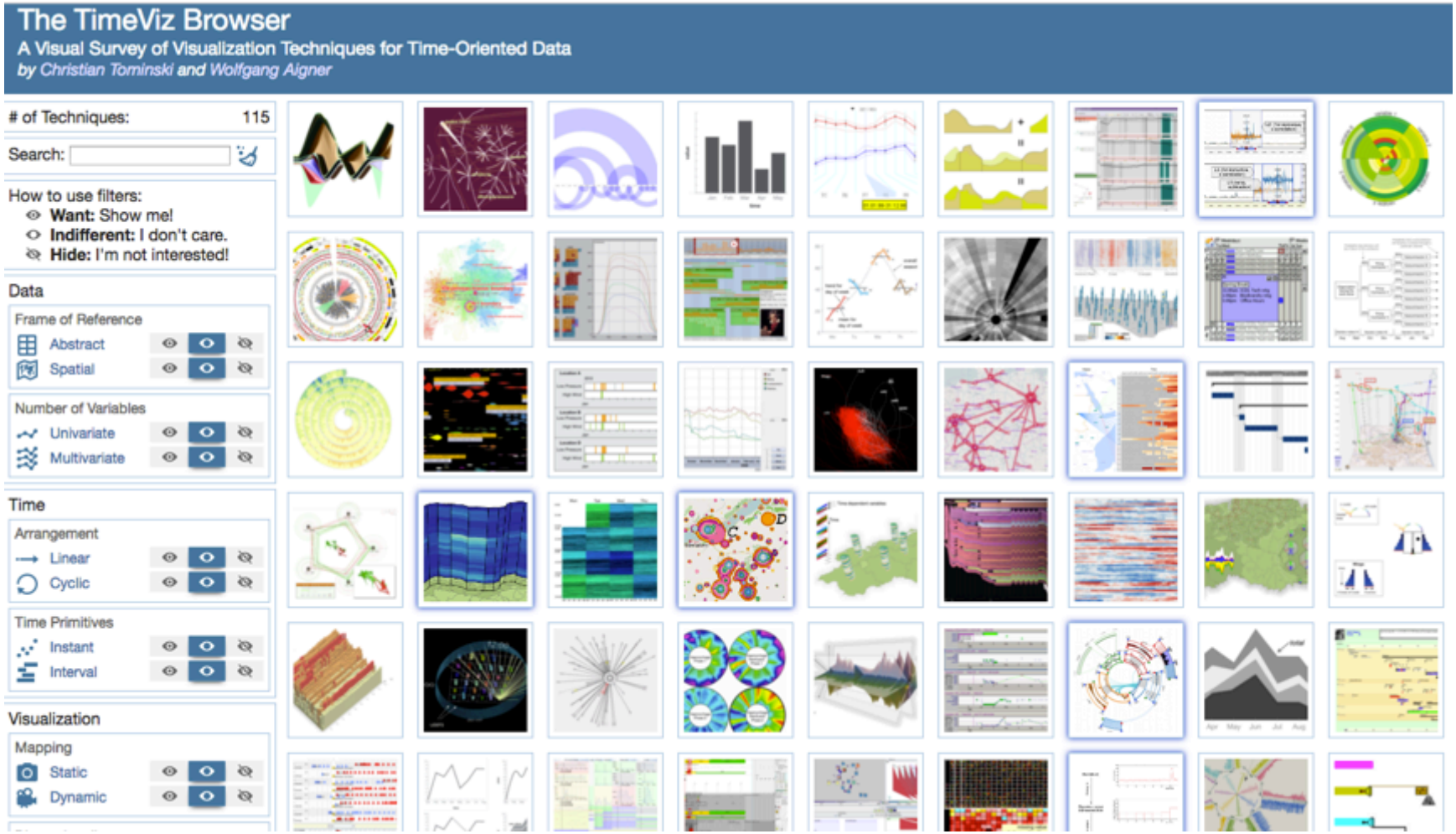
Combining Space and Time



Time	**Temporal primitives**	**Time Points**
	Structure of time	**Linear**
Data	**Frame of reference**	**Spatial**
	Number of variables	**Multivariate**
	Level of abstraction	**Data**
Representation	**Time dependency**	**Static, Dynamic**
	Dimensionality	**3D**

Kapler & Wright InfoVis '04

Visualizing Time-oriented Data
A Systematic View



http://www.timeviz.net/

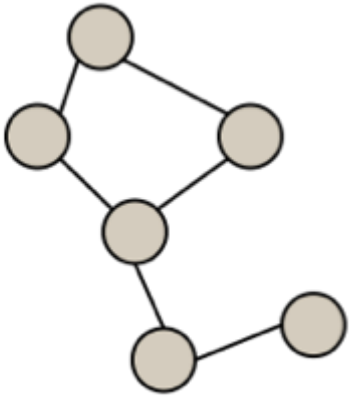
Aigner, Wolfgang, et al. "Visualizing time-oriented data—a systematic view." Computers & Graphics 31.3 (2007): 401-409.

Visualizing Trees and Graphs

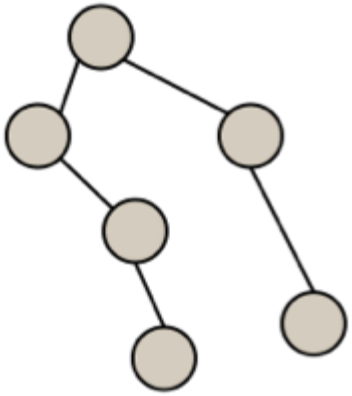
Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Graphs and Trees

- Graphs
  - Model relations among data
  - Nodes and edges



- Trees
  - Graphs with hierarchical structure
    - Connected graph with N-1 edges
  - Nodes as parents and children



Spatial Layout

- A primary concern of graph drawing is the spatial arrangement of nodes and edges.

- Often the goal is to effectively depict the graph structure:
  - Connectivity, path-following
  - Network distance
  - Clustering
  - Ordering (e.g., hierarchy level)

Many Applications

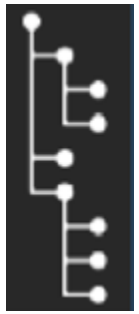
- Tournaments
- Organization Charts
- Biological Interactions (Genes, Proteins)
- Computer Networks
- Social Networks
- Integrated Circuit Design

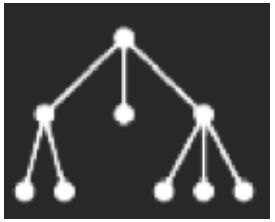
Visualizing Trees

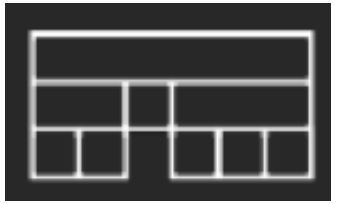
Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

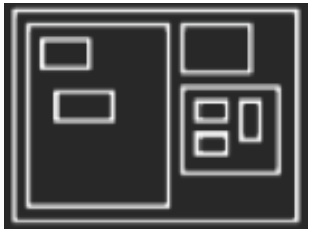
Tree Visualizations

- Indented lists
  - Linear list, indentation encodes depth
- Node-link trees
  - Nodes connected by lines/curves
- Layered diagrams
  - Relative position and alignment
- Treemaps (Enclosure diagrams)
  - Represent hierarchy by enclosure



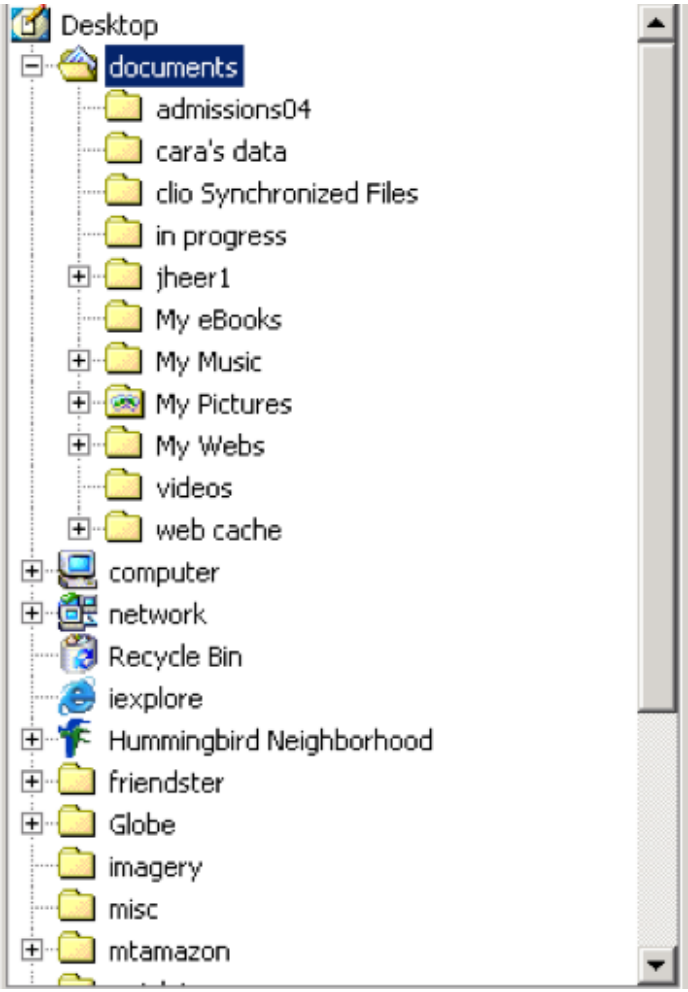






Indented List

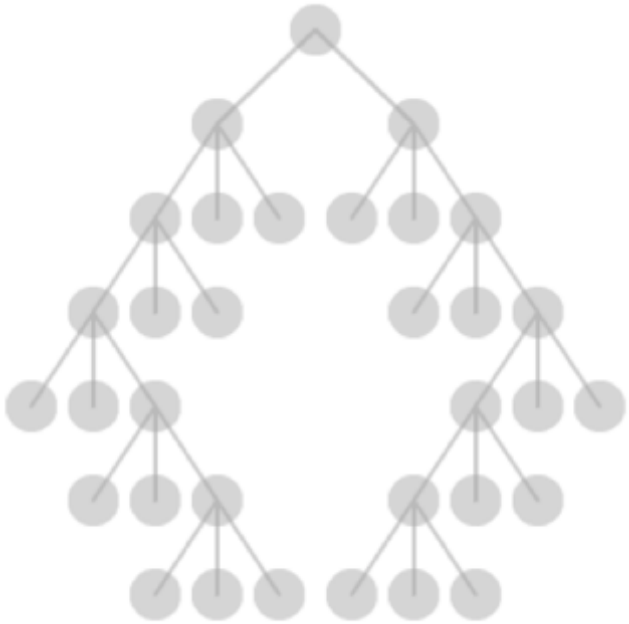
- Places all items along vertically spaced rows
- Indentation used to show parent/child relationships
- Commonly used as a component in an interface
- Breadth and depth contend for space
- Often requires a great deal of scrolling



Node-Link Trees

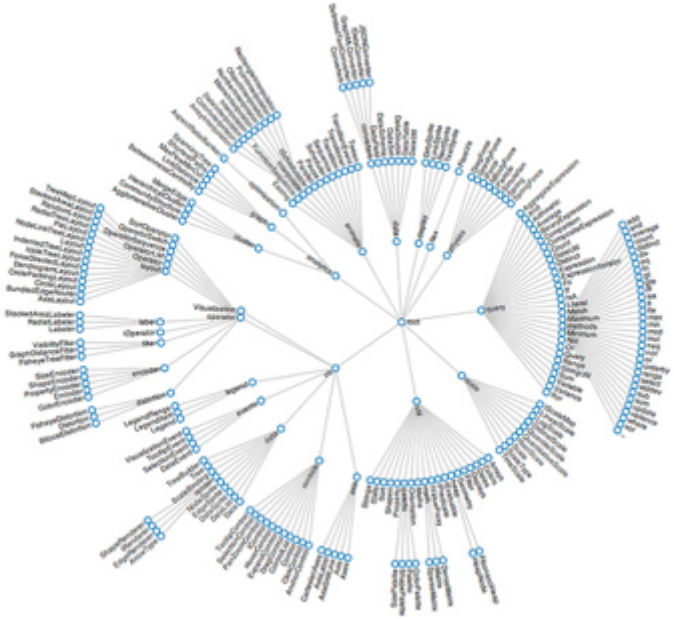
- Nodes are distributed in space, connected by straight or curved lines.
- Typical approach is to use 2D space to break apart breadth and depth.
- Often space is used to communicate hierarchical orientation (e.g., towards authority or generality)

- Reingold-Tilford algorithm can achieve linear time in presenting a compact layout

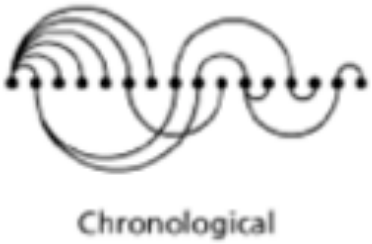


Other Node-Link Trees

- Radial layout places the root in the center.
  - The radius encodes the depth.



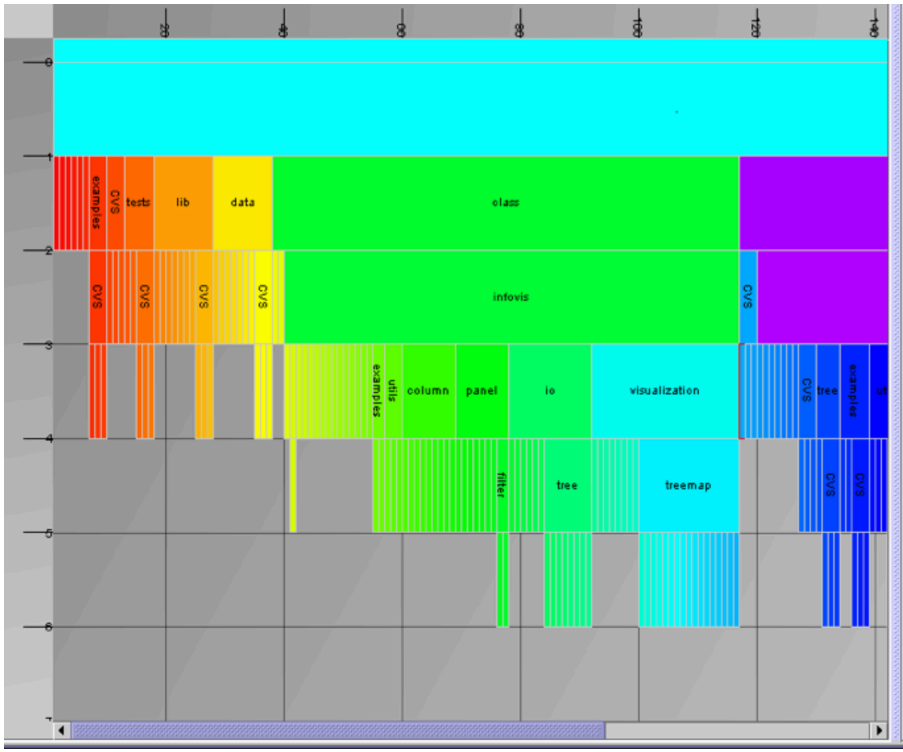
- ThreadArcs
  - combine the chronology of messages with the branching tree structure in a mixed-model visualization



(Optional Reading) Kerr, Bernard. "Thread arcs: An email thread visualization." Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on. IEEE, 2003.

Layered Diagrams

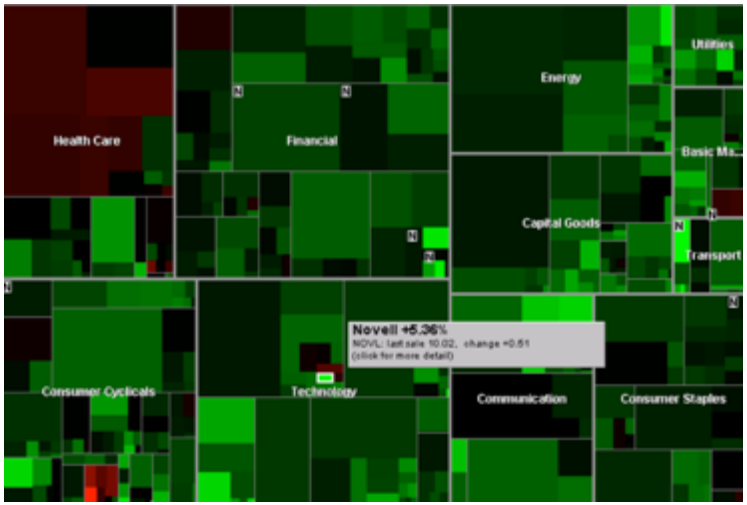
- Signify tree structure using
  - Layering
  - Adjacency
  - Alignment
- Involves recursive sub-division of space
- Higher-level nodes get a larger layer area, whether that is horizontal or angular extent.
- Child levels are layered, constrained to parent's extent

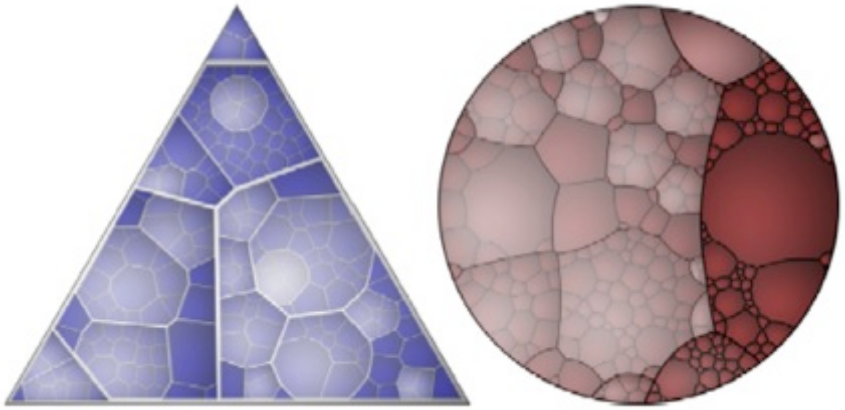


Icicle Trees

Treemaps (Enclosure Diagrams)

- Recursively fill space. Enclosure signifies hierarchy.
- Additional measures can be taken to control aspect ratio of cells.
- Often uses rectangles, but other shapes are possible, e.g., iterative Voronoi tesselation.





http://www.cs.umd.edu/hcil/treemap-history/

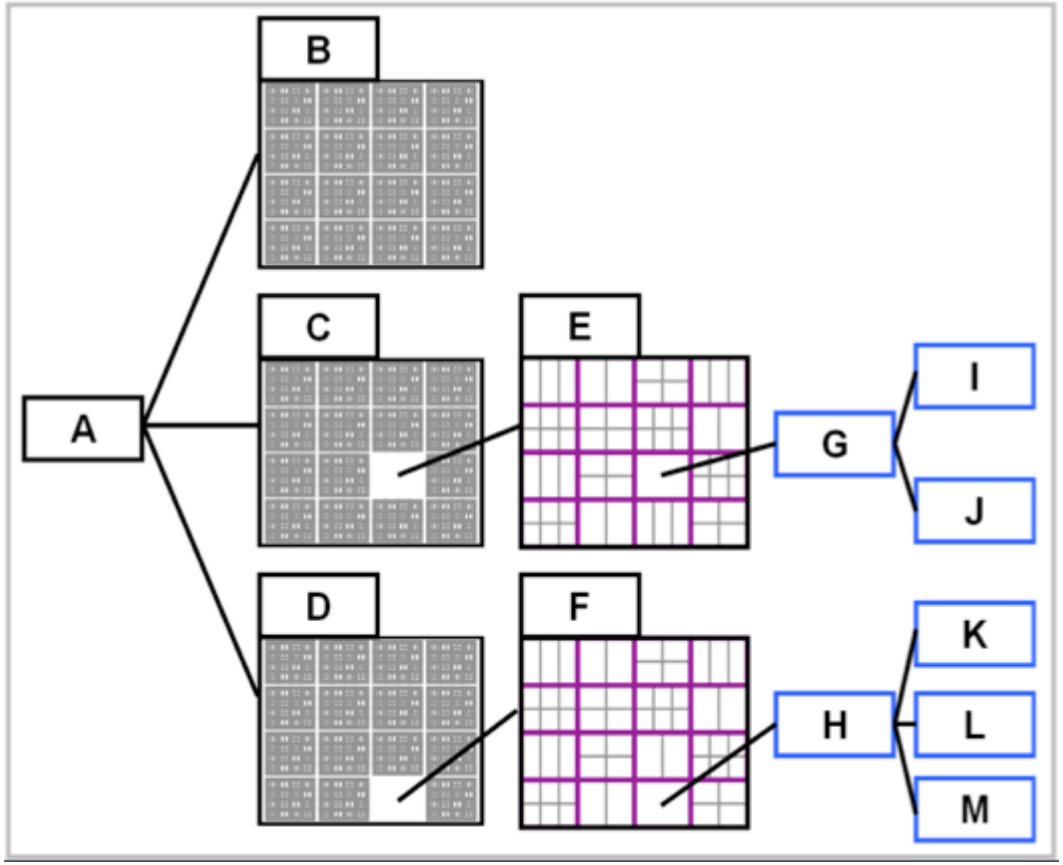
Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Hybrids

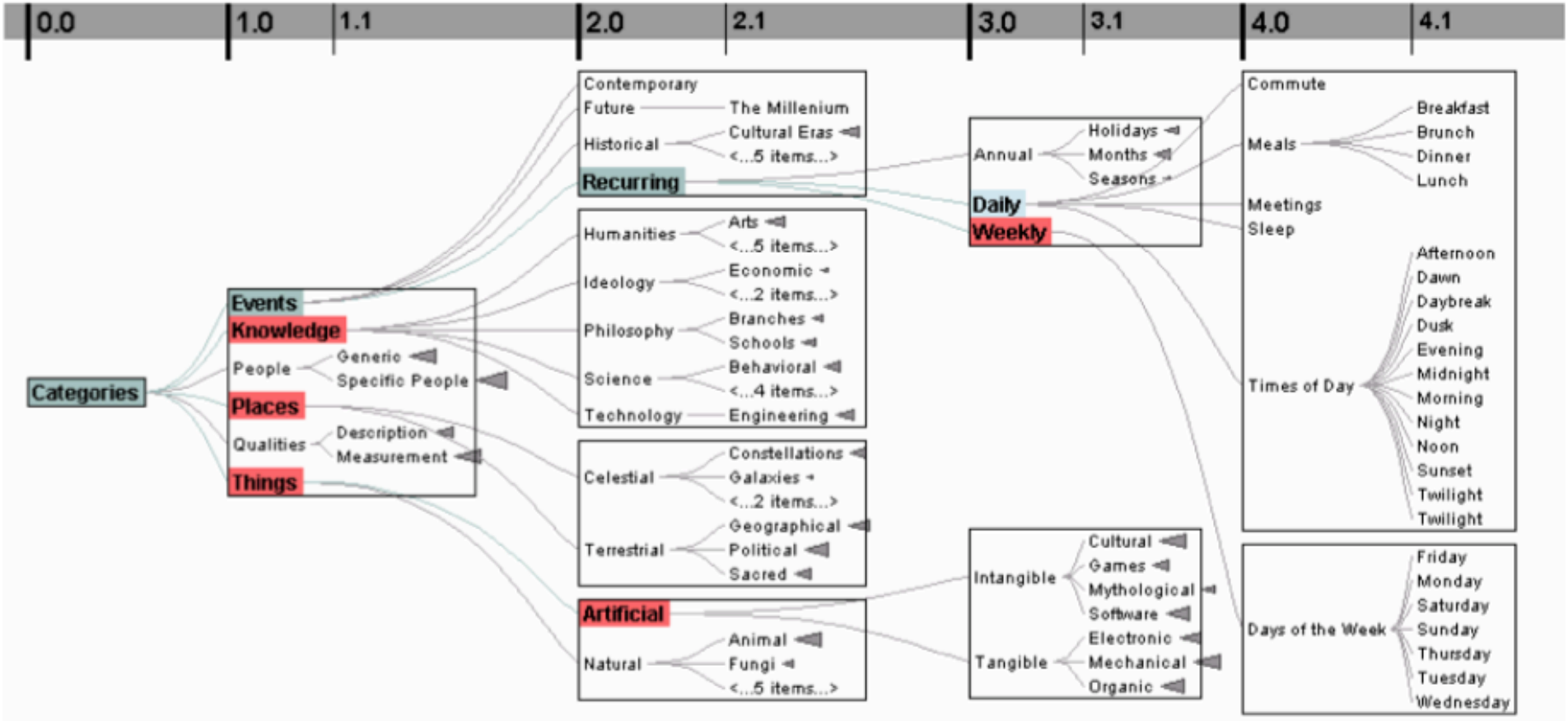
- Elastic Hierarchies
  - Node-link diagram with treemap nodes.

- Video:

https://www.youtube.com/watch?v=nvsIqYQ75yA



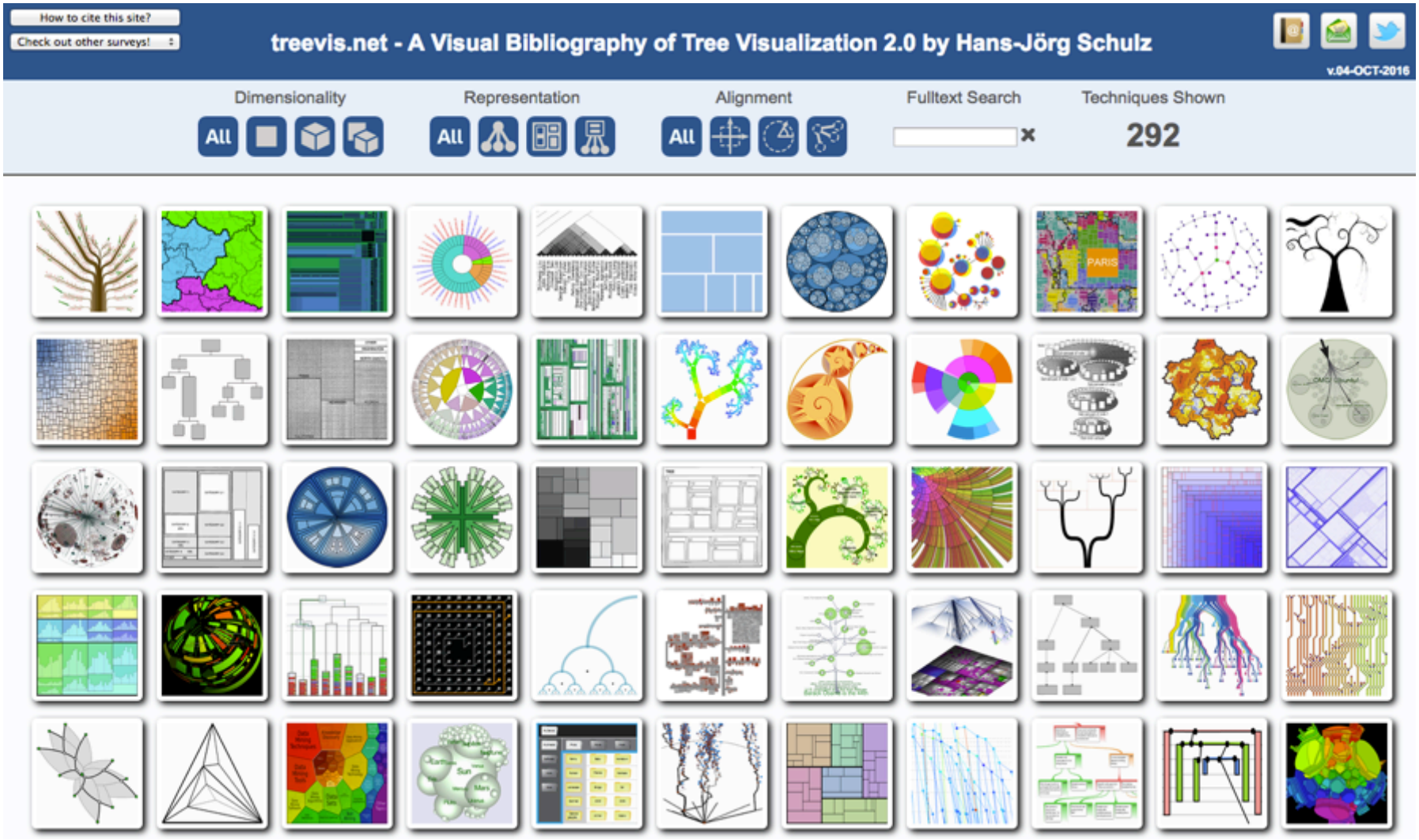
Interactive: Degree-of-interest Trees



- Cull “un-interesting” nodes on a per block basis until all blocks on a level fit within bounds.
- Attempt to center child blocks beneath parents.

Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Treevis.net

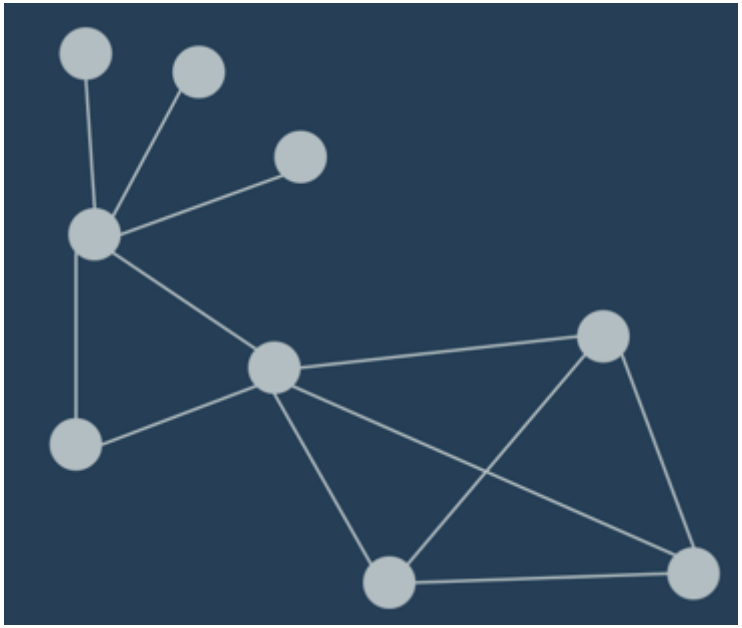


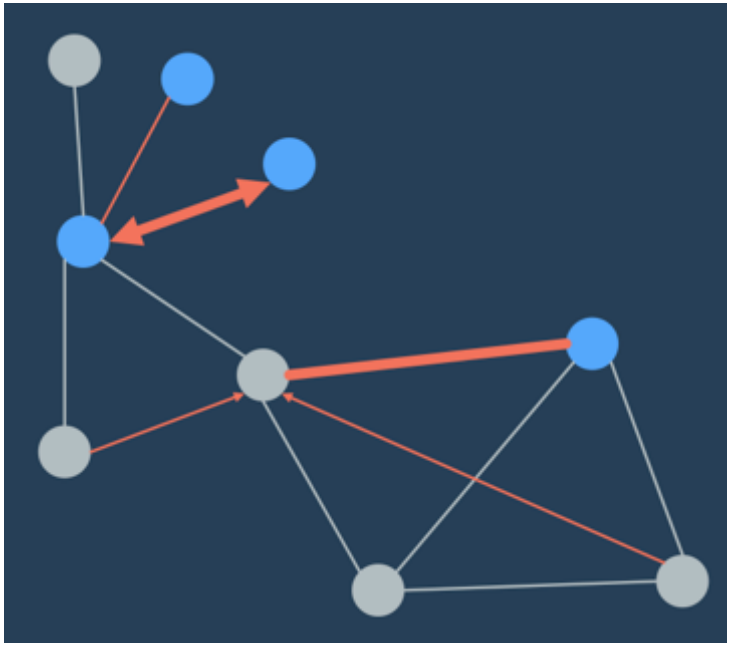
Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Visualizing Graphs

Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

What's in a Graph?



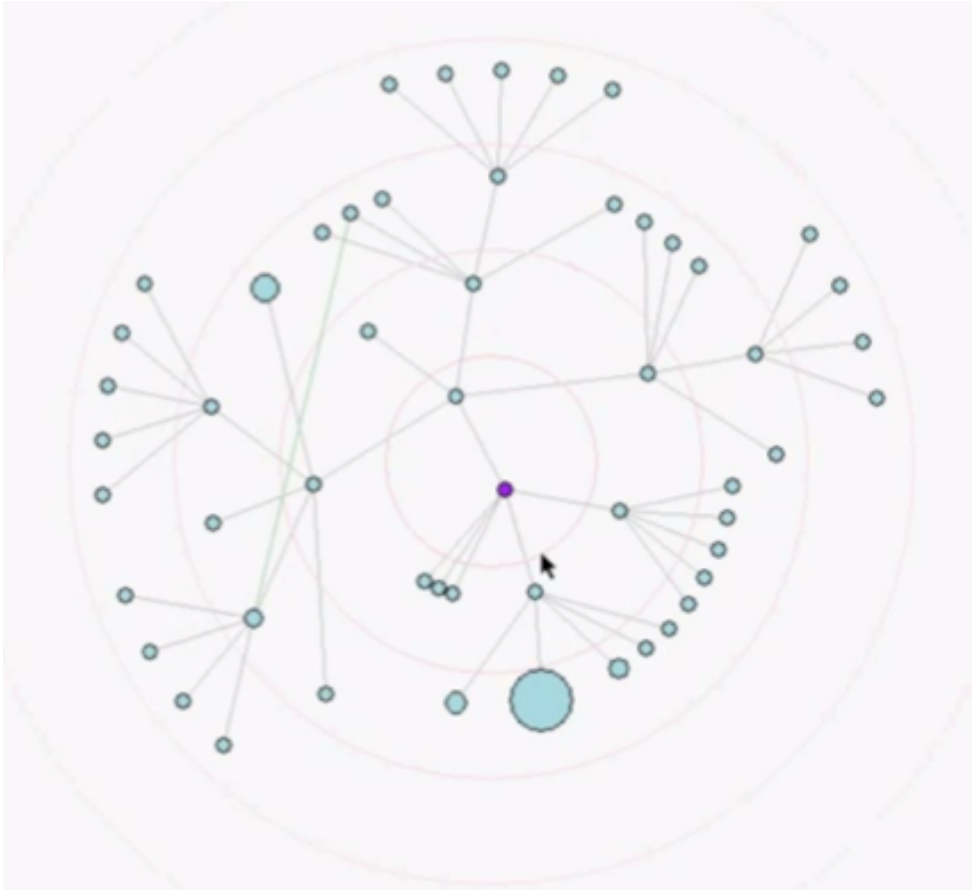


Graph Visualization

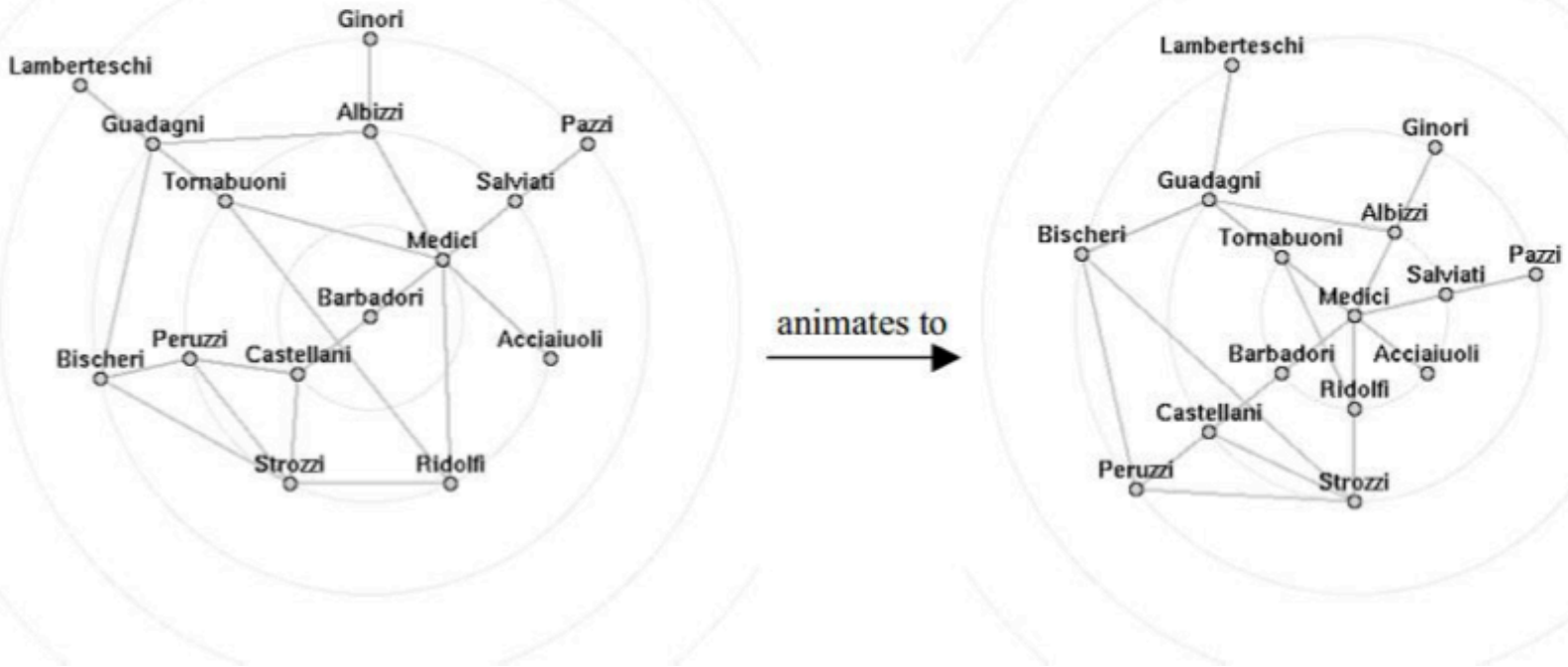
- Two representations:
  - Node-link diagrams
  - Matrices

Tree in the Graph

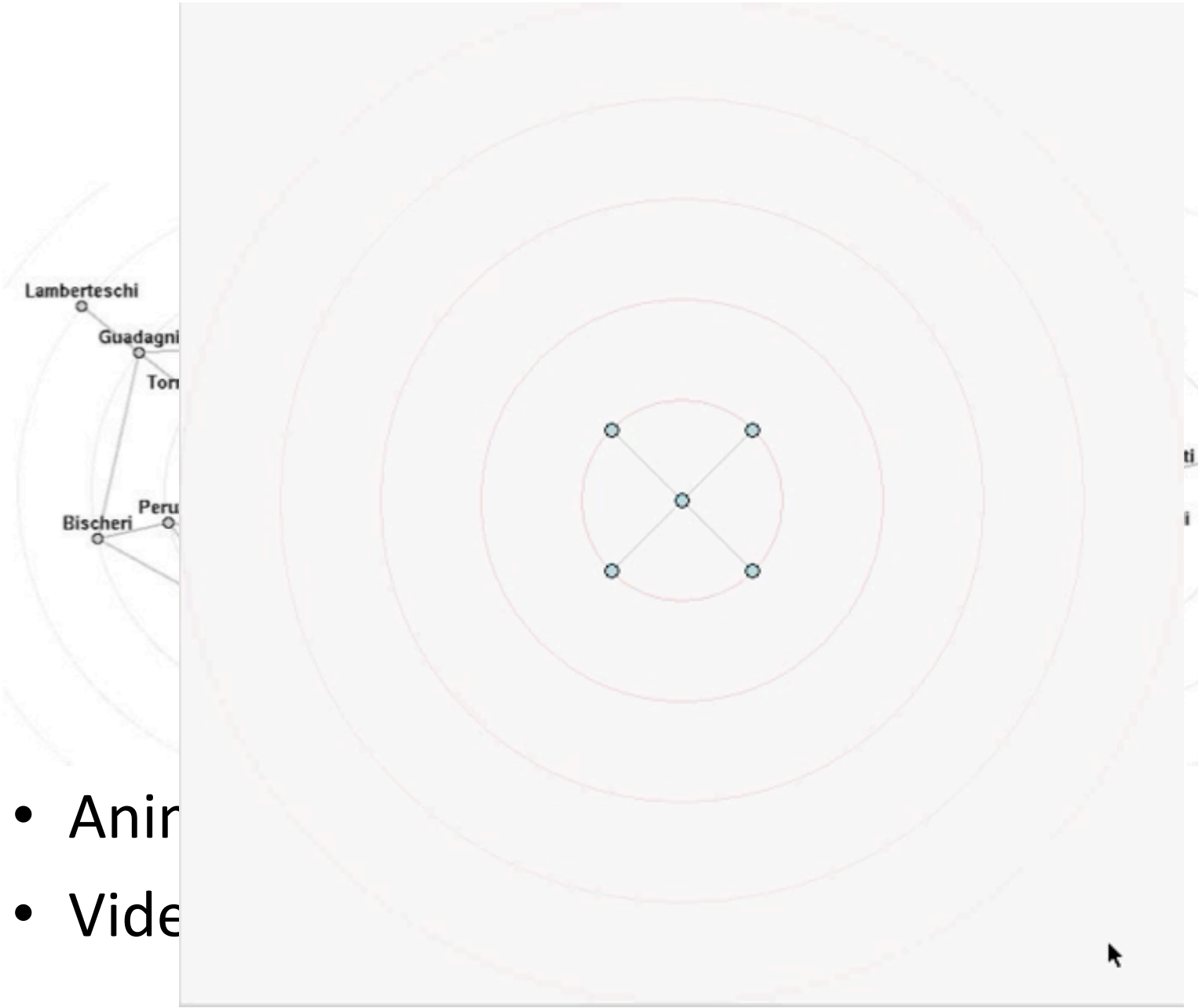
- Many graphs are tree-like or have useful spanning trees
- Spanning trees lead to arbitrary roots
- Fast tree layouts allow graph layouts to be recalculated at interactive rates



Tree in the Graph

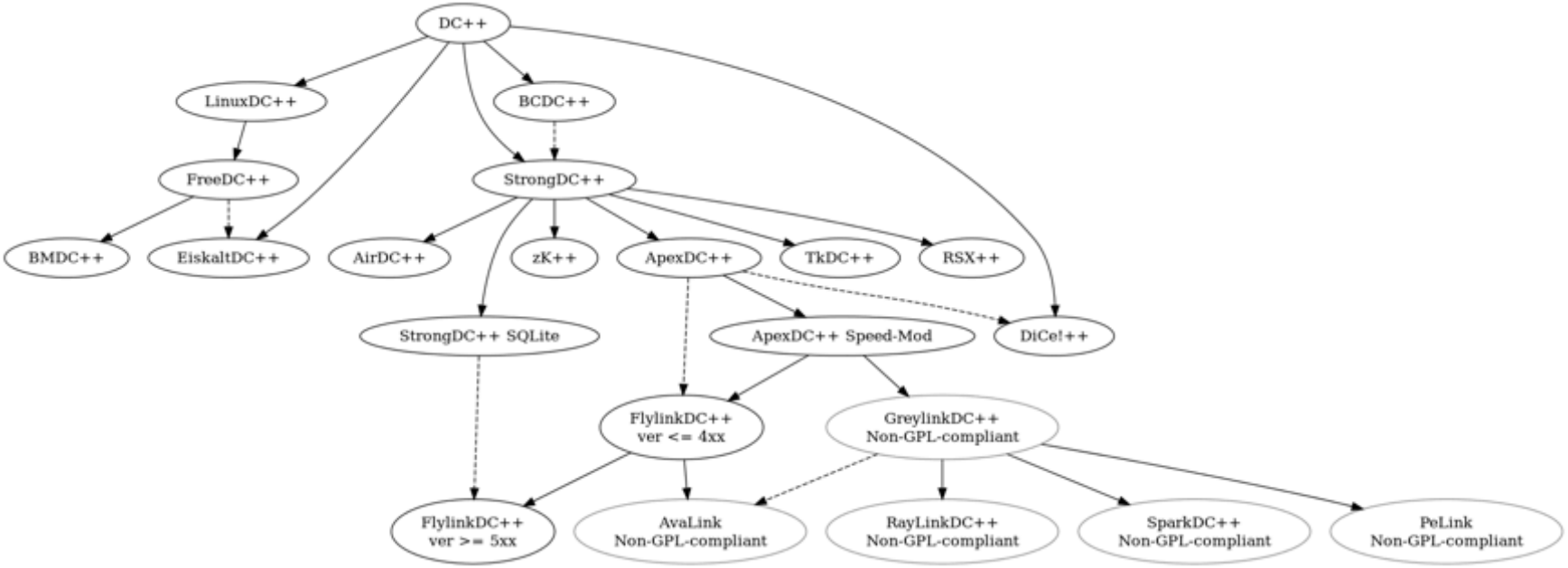


- Animated Graphs with Radial Layout
- Video: http://www.youtube.com/watch?v=OPX5iGro_lA



- Anir
- Vide

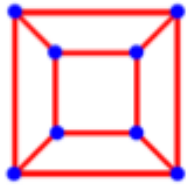
Hierarchical Graph Layout

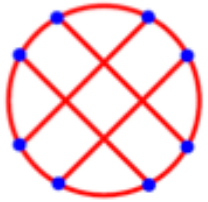


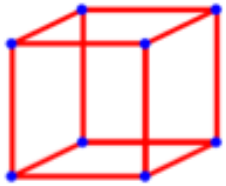
- Evolution of the DC++ tool
- Layered graph drawing
- Layout of a Direct Acyclic Graph
- Hierarchical layering based on descent

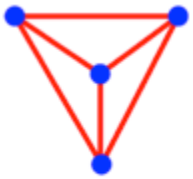
Optimization Techniques

- Treat layout as an optimization problem
  - Define layout using an energy model along with constraints: equations the layout should obey.
  - Use optimization algorithms to solve
- Commonly posed as a physical system
  - Charged particles, springs, drag force, ...
- Different constraints can be introduced
  - Minimize edge crossings
  - Minimize area
  - Minimize line bends
  - Minimize line slopes
  - Maximize smallest angle between edges
  - Maximize symmetry









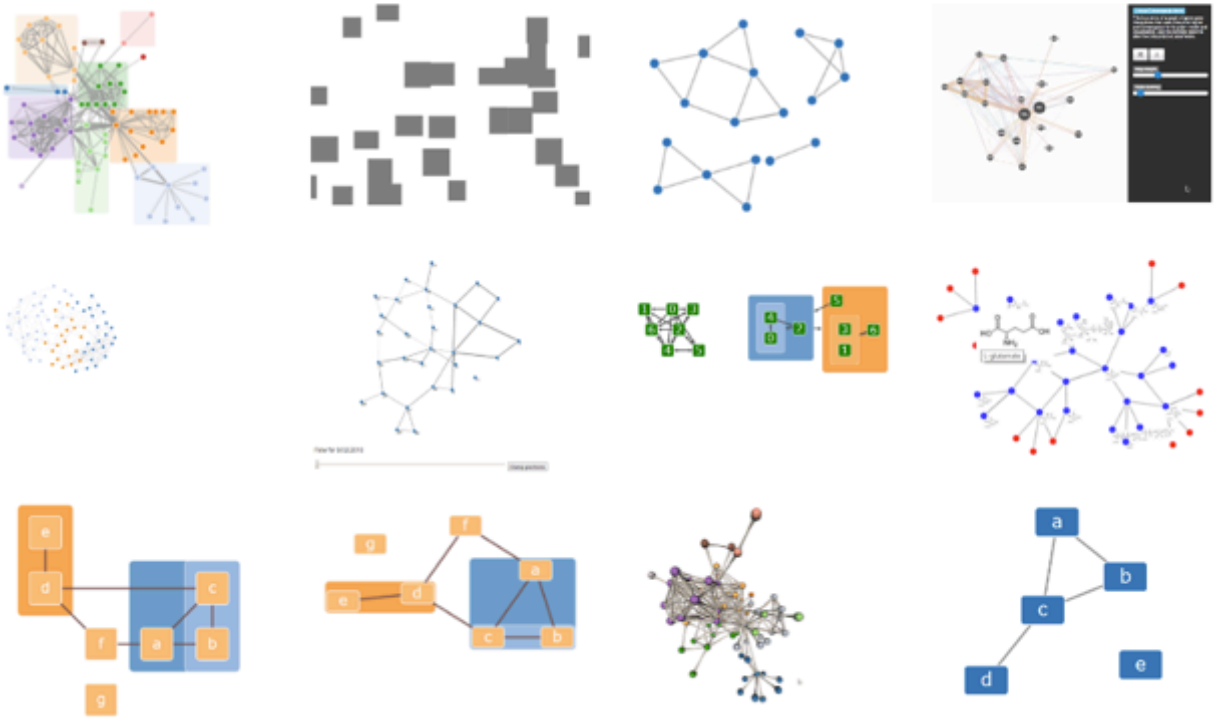
min # crossings

max symmetries

Constraint-based Optimization and Layout

cola.js

Constraint-Based Layout in the Browser

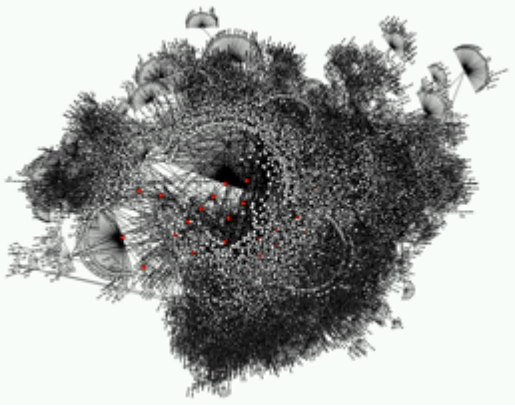


http://marvl.infotech.monash.edu/webcola/

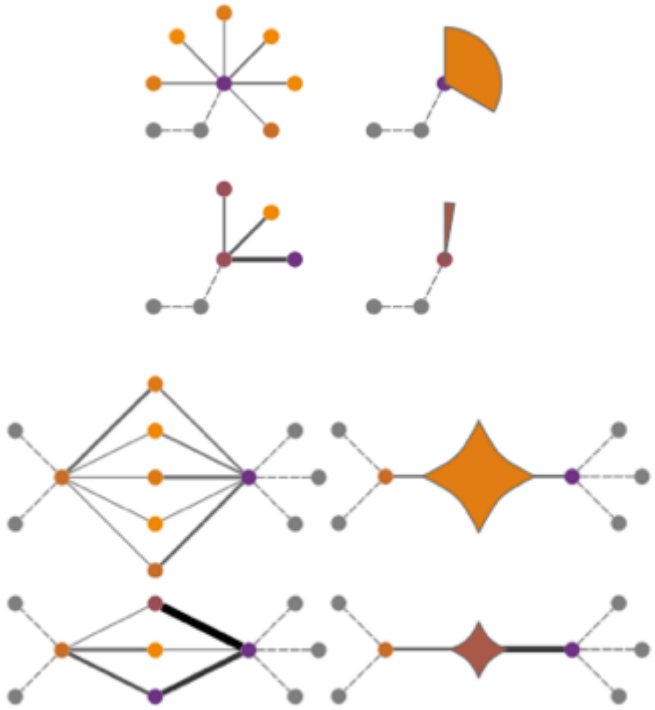
Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

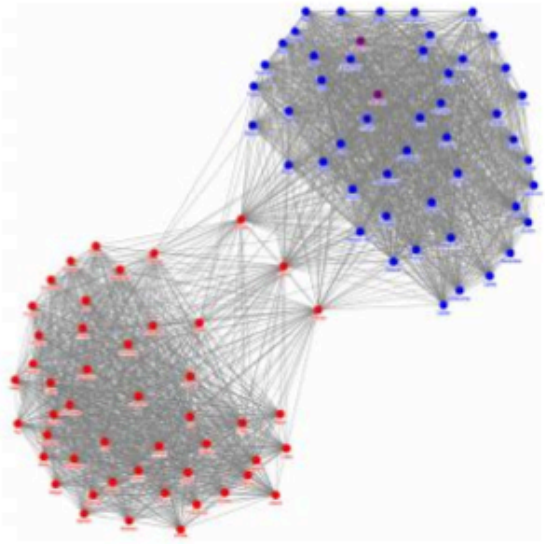
Scalability Issue

- Need to cope with messiness
- Solutions
  - Extracting network motifs
  - Taking advantage of node attributes
  - Degree-of-Interest graphs
  - Use the alternative representation: matrix

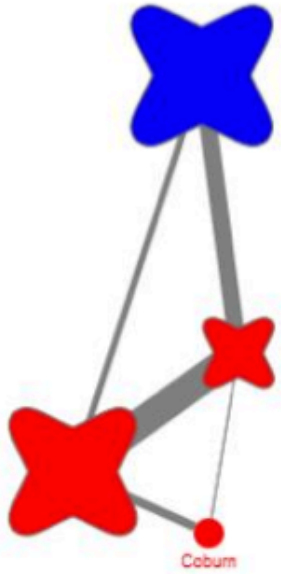


Motifs





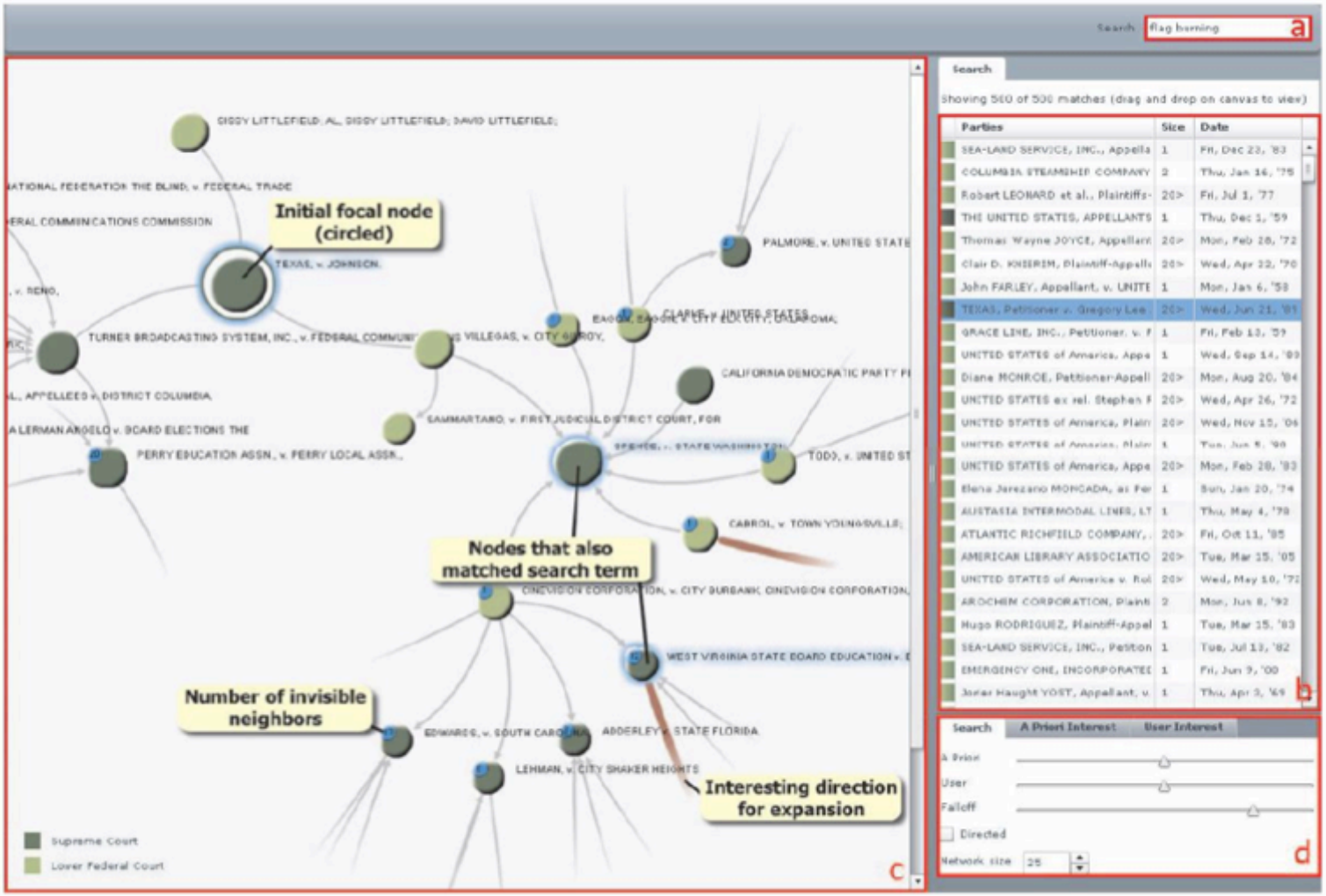
(a) 65%



(b) 65% simplified

Dunne, Cody, and Ben Shneiderman. "Motif simplification: improving network visualization readability with fan, connector, and clique glyphs." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013.

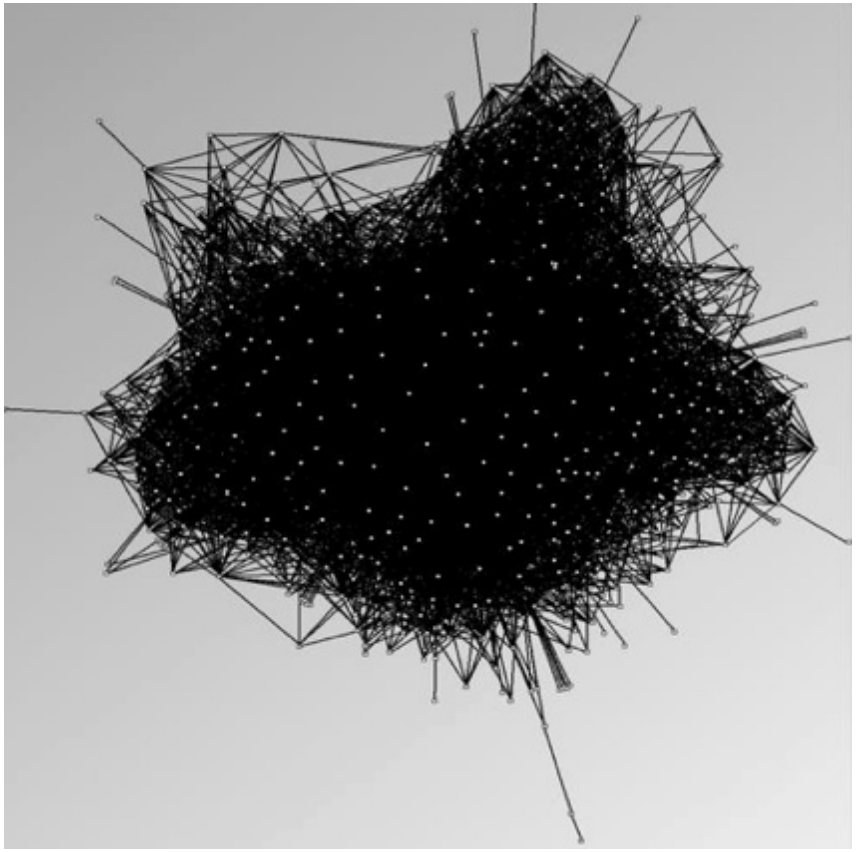
Interactive Degree-of-Interest Graphs

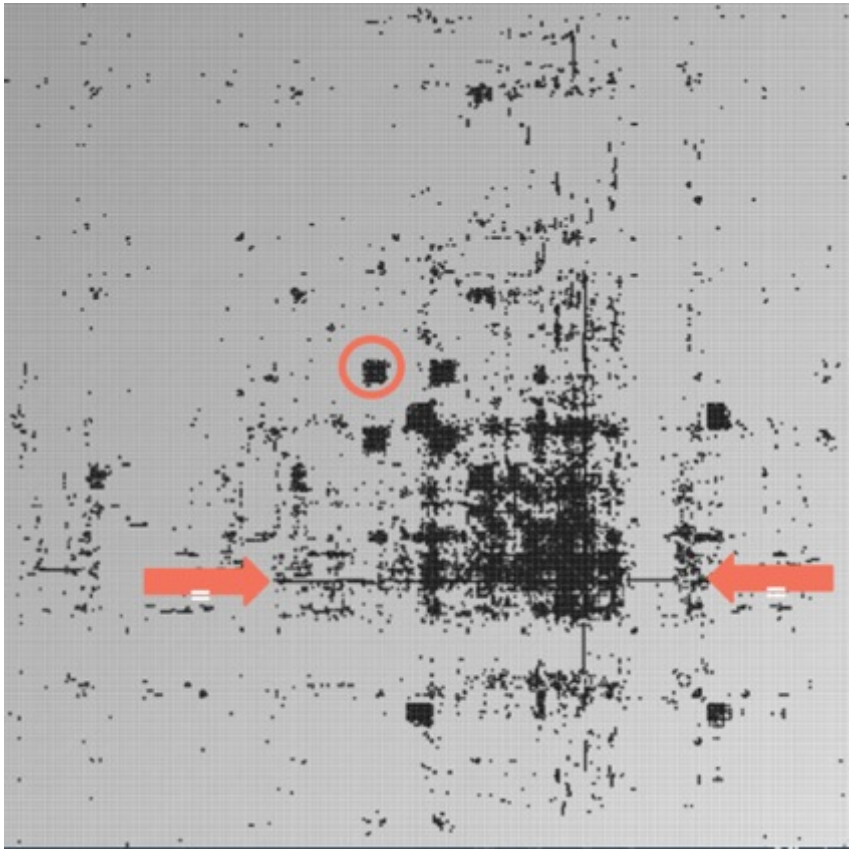


Van Ham, Frank, and Adam Perer. "“Search, show context, expand on demand”: supporting large graph exploration with degree-of-interest." IEEE Transactions on Visualization and Computer Graphics 15.6 (2009).

Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Matrices

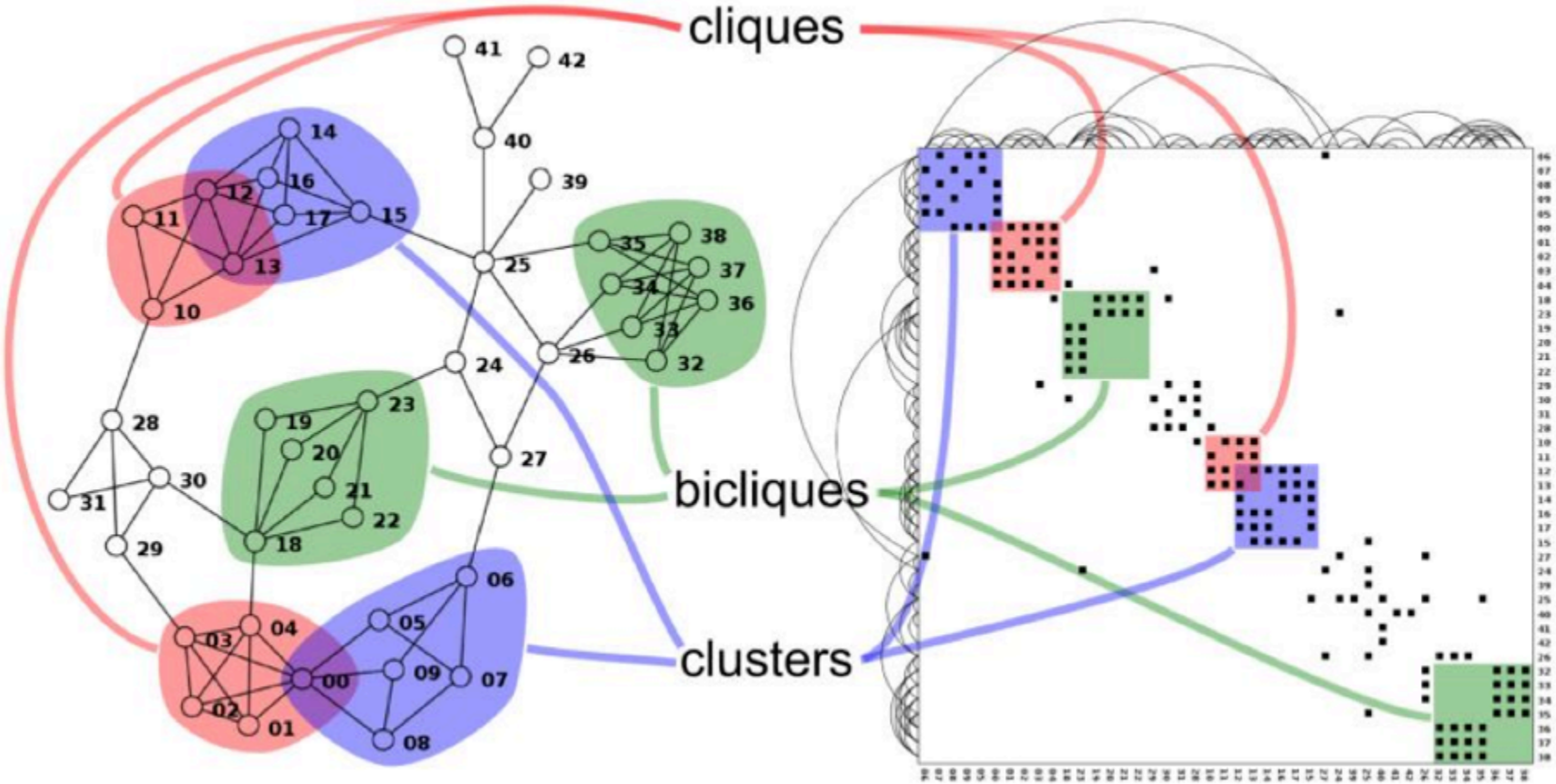




Matrix vs. Node-link

Matrix	Node-link
Require learning	**Familiar**
No overlap	Node overlap
No crossings	Link crossing
Use a lot of space	**More compact**
Dense graphs	**Sparse graphs**

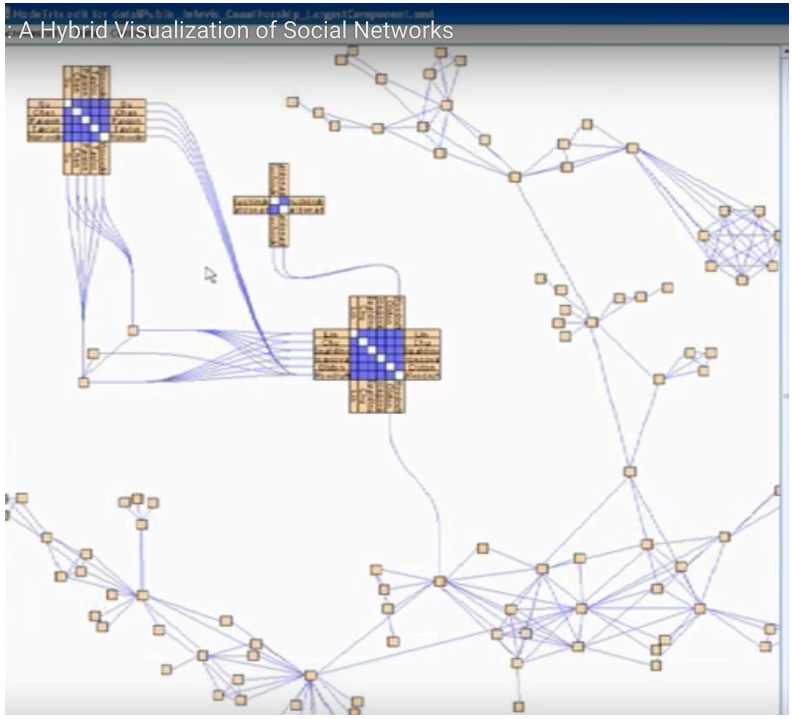
Node Link to Matrix



Dr. Ke Zhou (http://www.cs.nott.ac.uk/~pszkz/)

Hybrid

- Merging Node-link with Matrices is possible.



Video: https://www.youtube.com/watch?v=7G3MxyOcHKQ

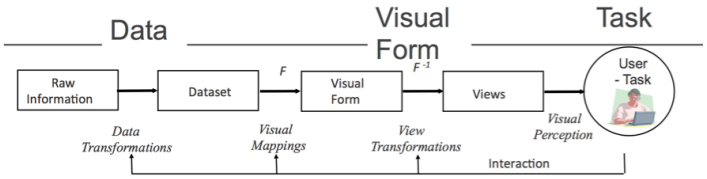
Henry, Nathalie, Jean-Daniel Fekete, and Michael J. McGuffin. "NodeTrix: a hybrid visualization of social networks." IEEE transactions on visualization and computer graphics 13.6 (2007): 1302-1309.

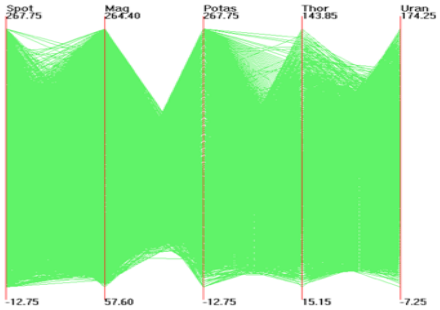
Summary

- Visualizing Tree Layout
  - Indented / Node-Link / Enclosure / Layers
- Visualizing Graph Layout
  - Tree in graph / Hierarchical graph layout
  - Layout Optimization
  - Scalability issue: motif, degree of interests, matrix
  - Matrix

Next Lecture

- Topic:
  - Recap of Fundamentals
  - Exam Review





- Next Monday (11 Mar)
  - 12:00 - 14:00
  - A25, Business South, Jubilee Campus