

Visualizing Relationships: Graphs, Trees and Networks

Lyn Bartram
IAT 814 week 7
19.10.2011

These slides have been largely adapted from B. Zupan, J. Heer and M. Hearst



Administrivia

- Final project
- <http://www.sfu.ca/siatclass/IAT355/Spring2014/Assignments/finalProject355.html>
- Team project: you must clear it with me.

Visualizing Relations

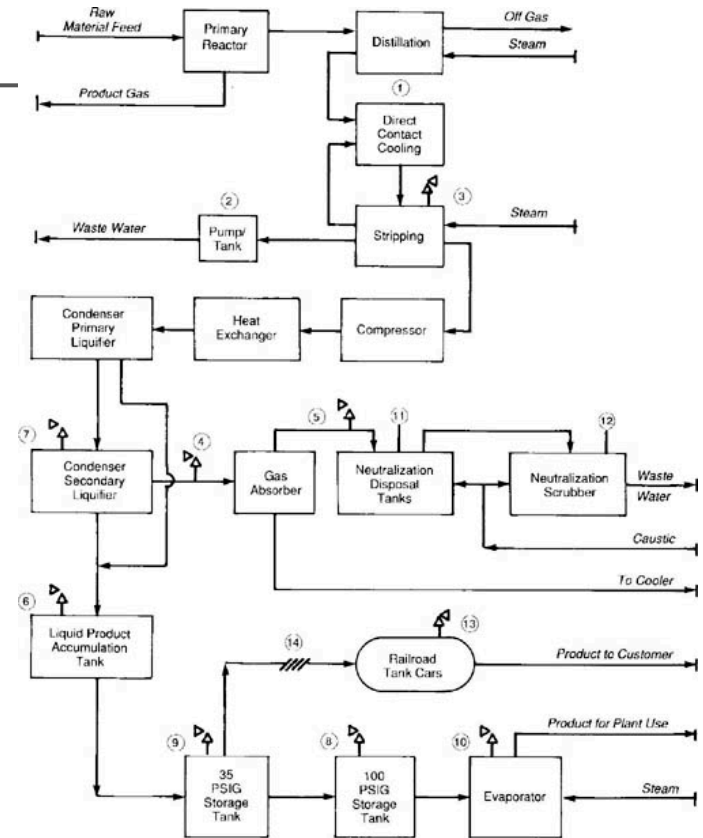
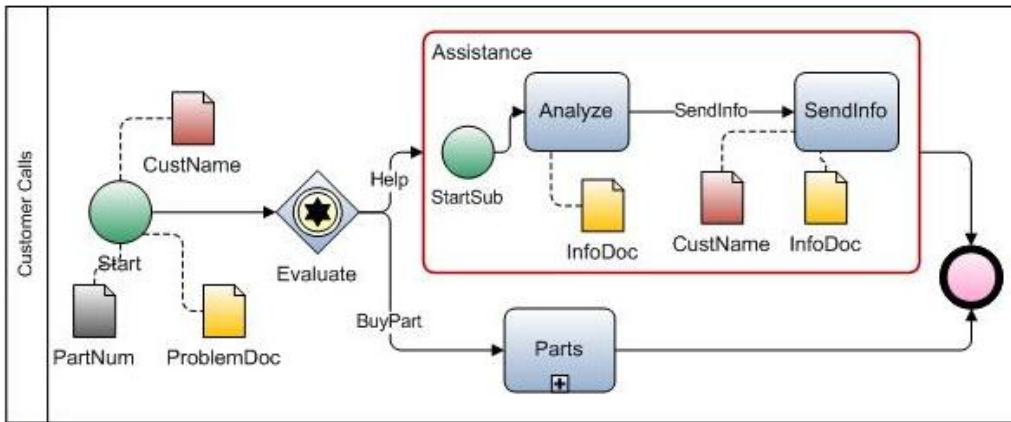
Why relations? Isn't all data inherently relational?

- Visualising data: seeing the patterns between the data values and attributes that emerge and associate or disassociate in some way
 - Visualising relations: when how one datum relates to another is an element in itself
-
- We want to see the overall structure of the data set
 - Patterns emerge from structure as well as from values/attributes

Common Applications

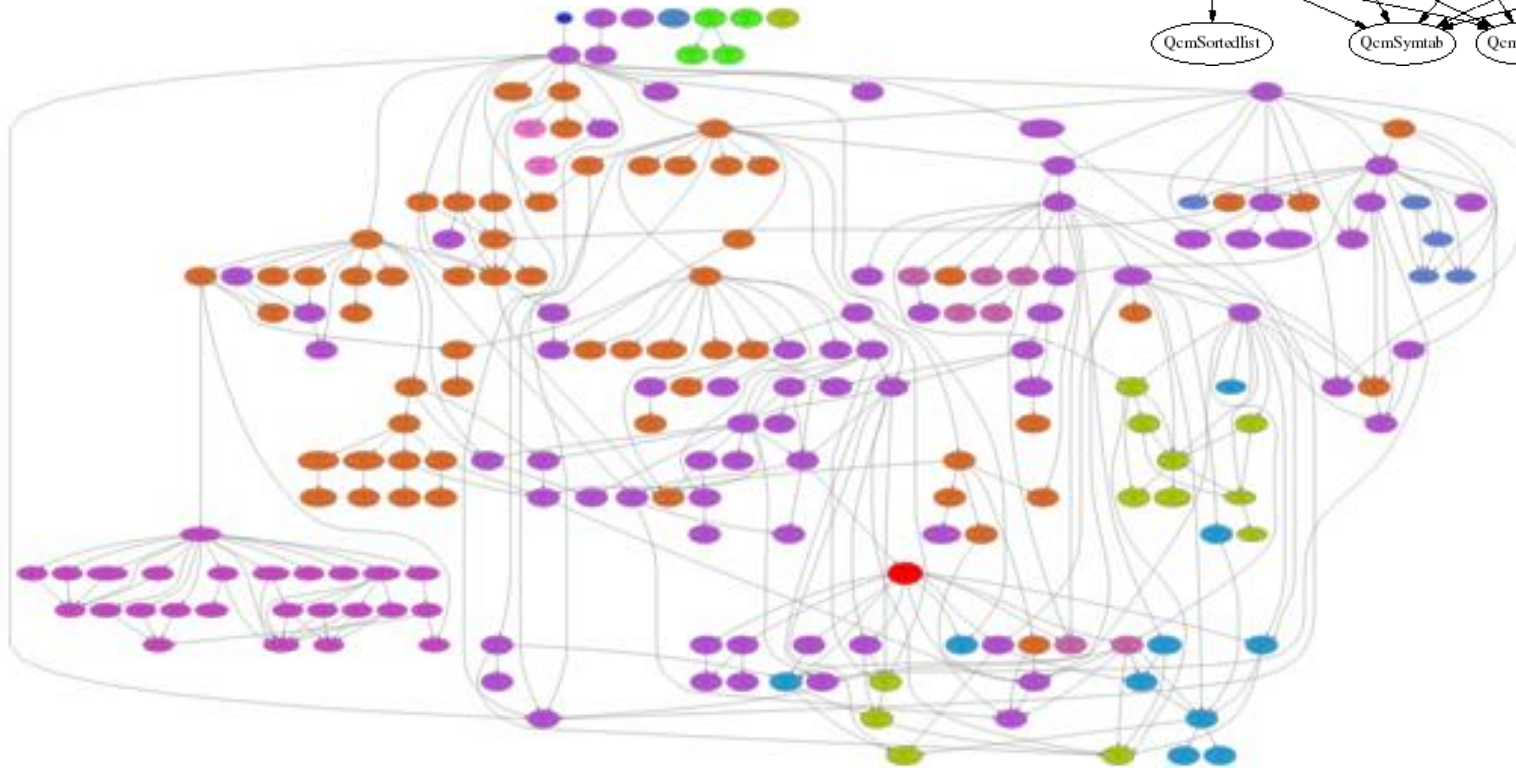
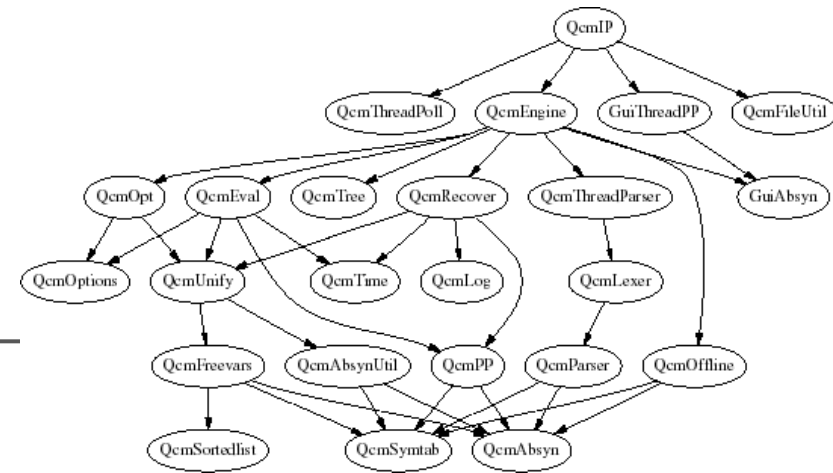
- Process Visualization (e.g., Visio)
- Dependency Graphs
- Biological Interactions (Genes, Proteins)
- Computer Networks
- Social Networks
- Concept maps
- Ontologies
- Simulation and Modeling
- Probability maps

Process flow diagrams



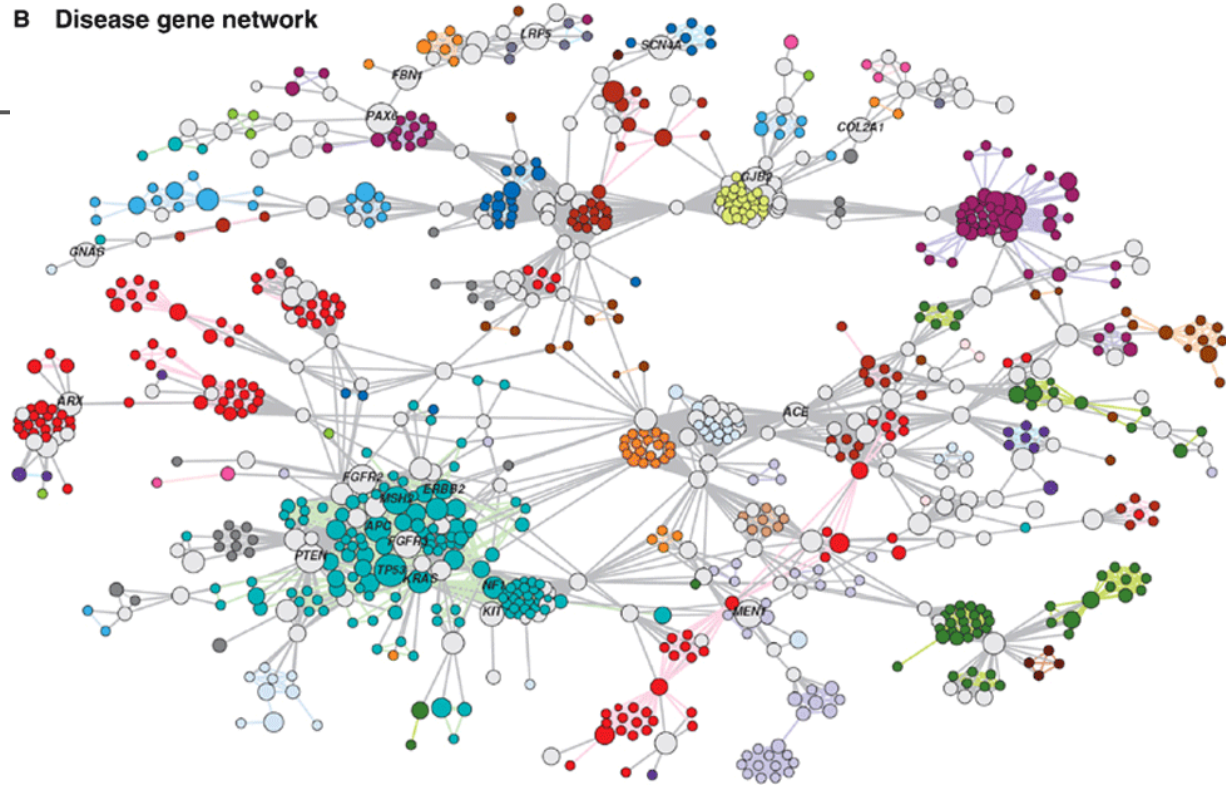
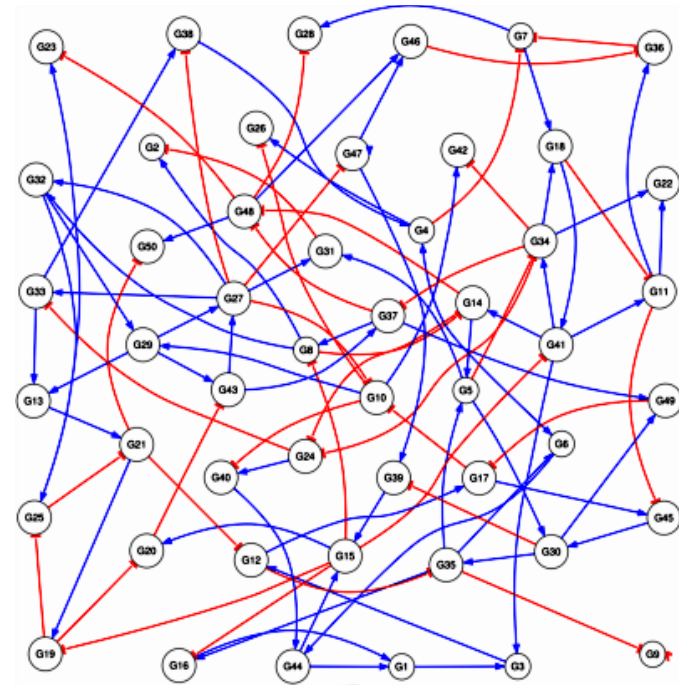
Page-1

Dependency graphs

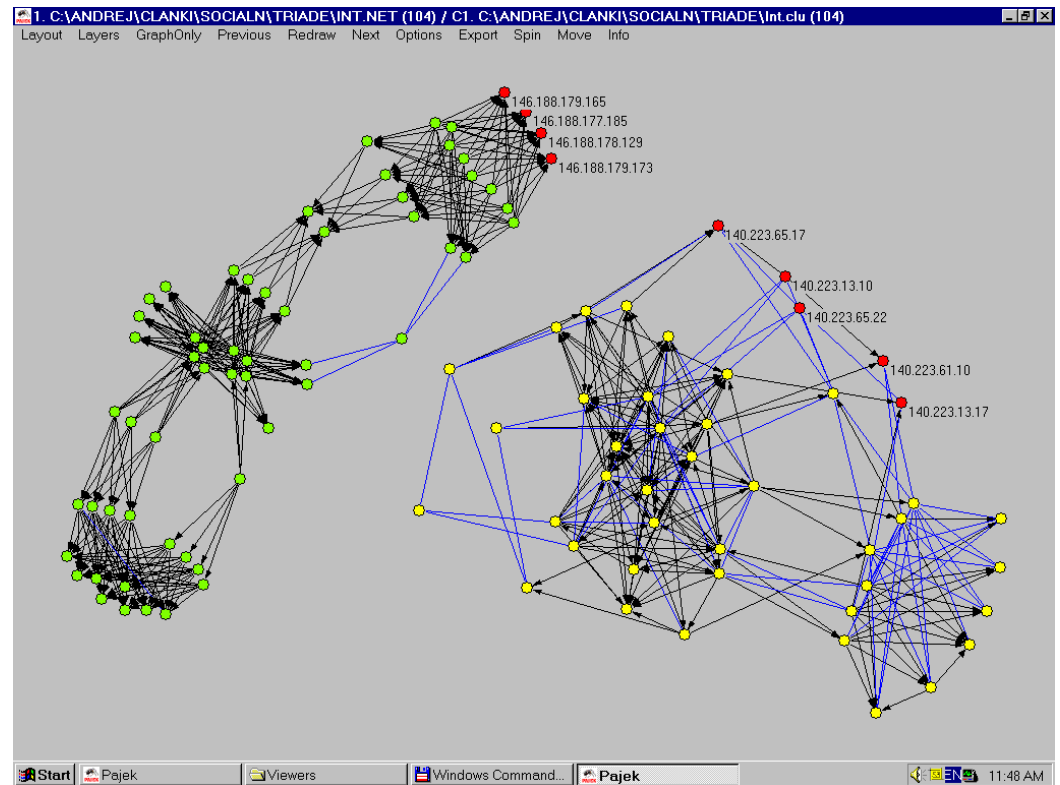
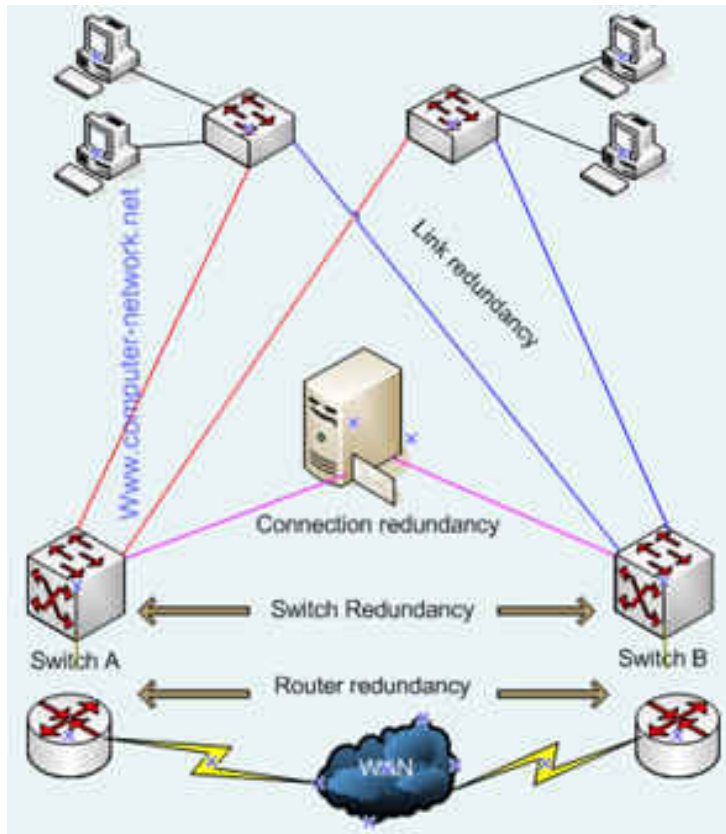


Gene networks

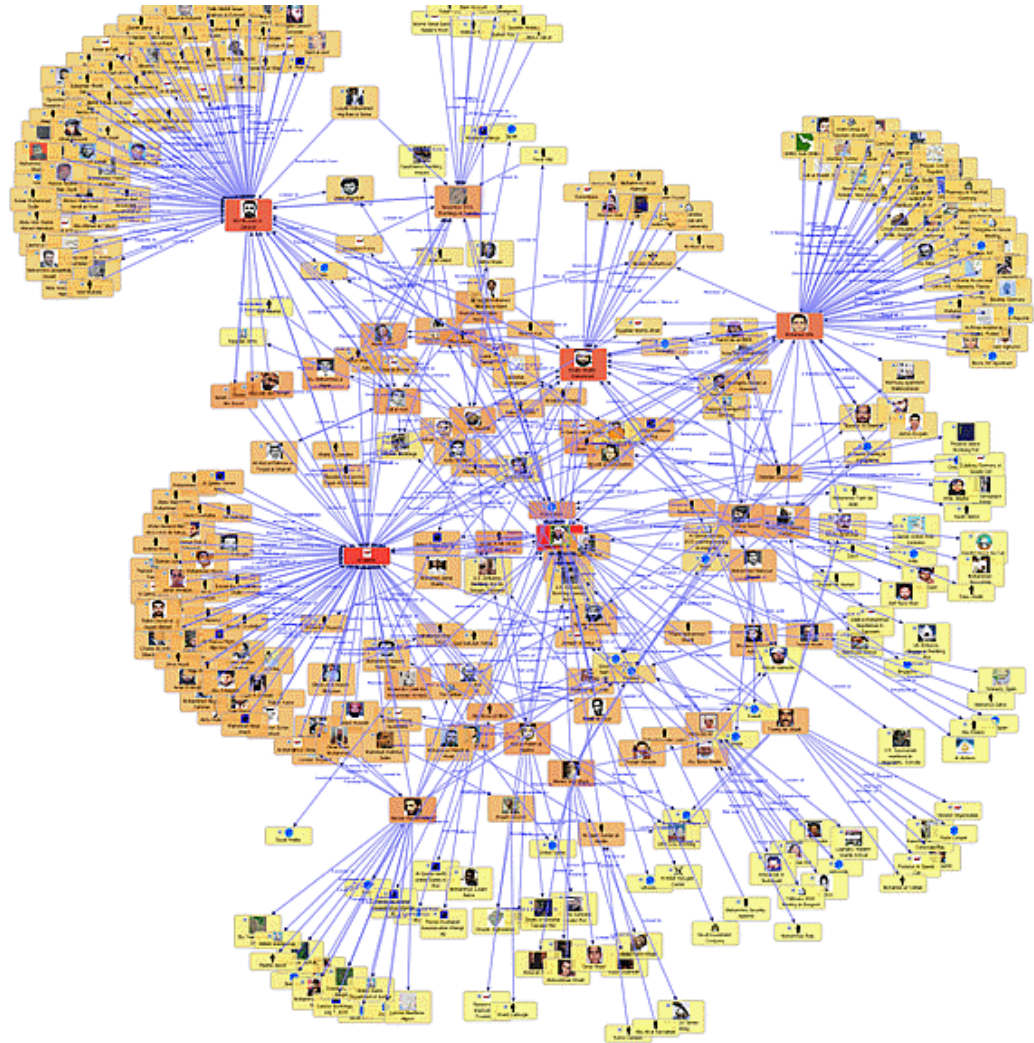
B Disease gene network



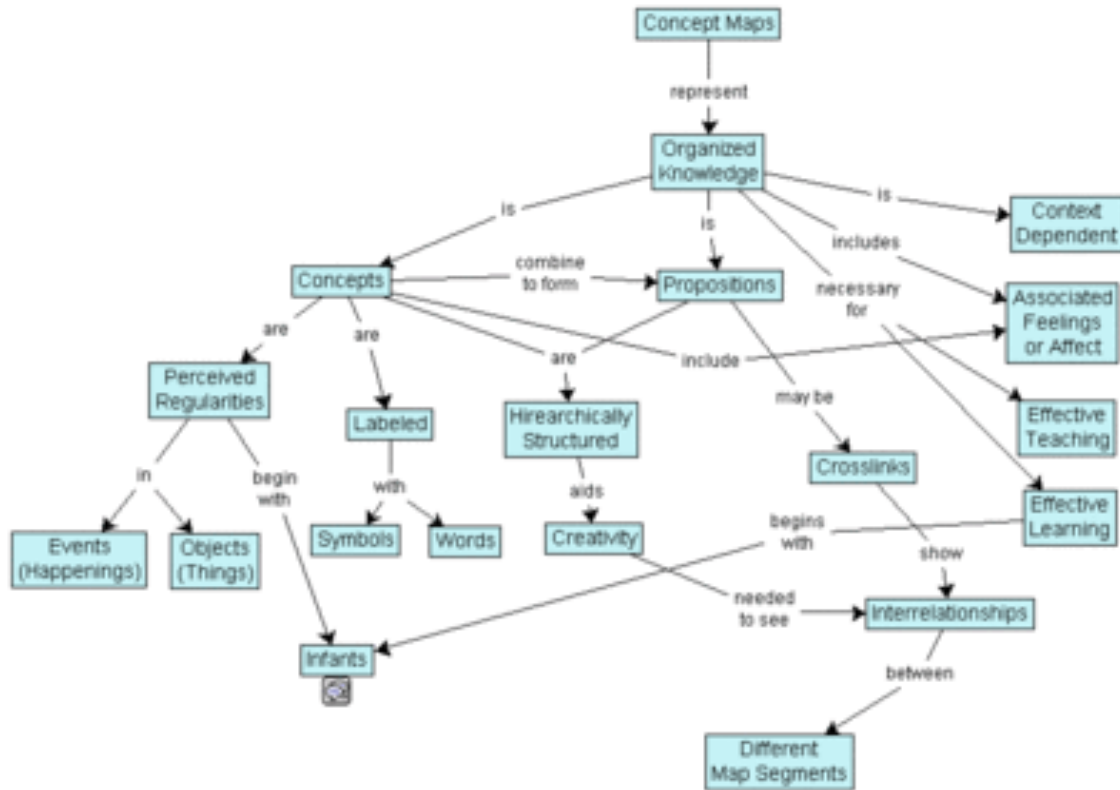
Computer networks



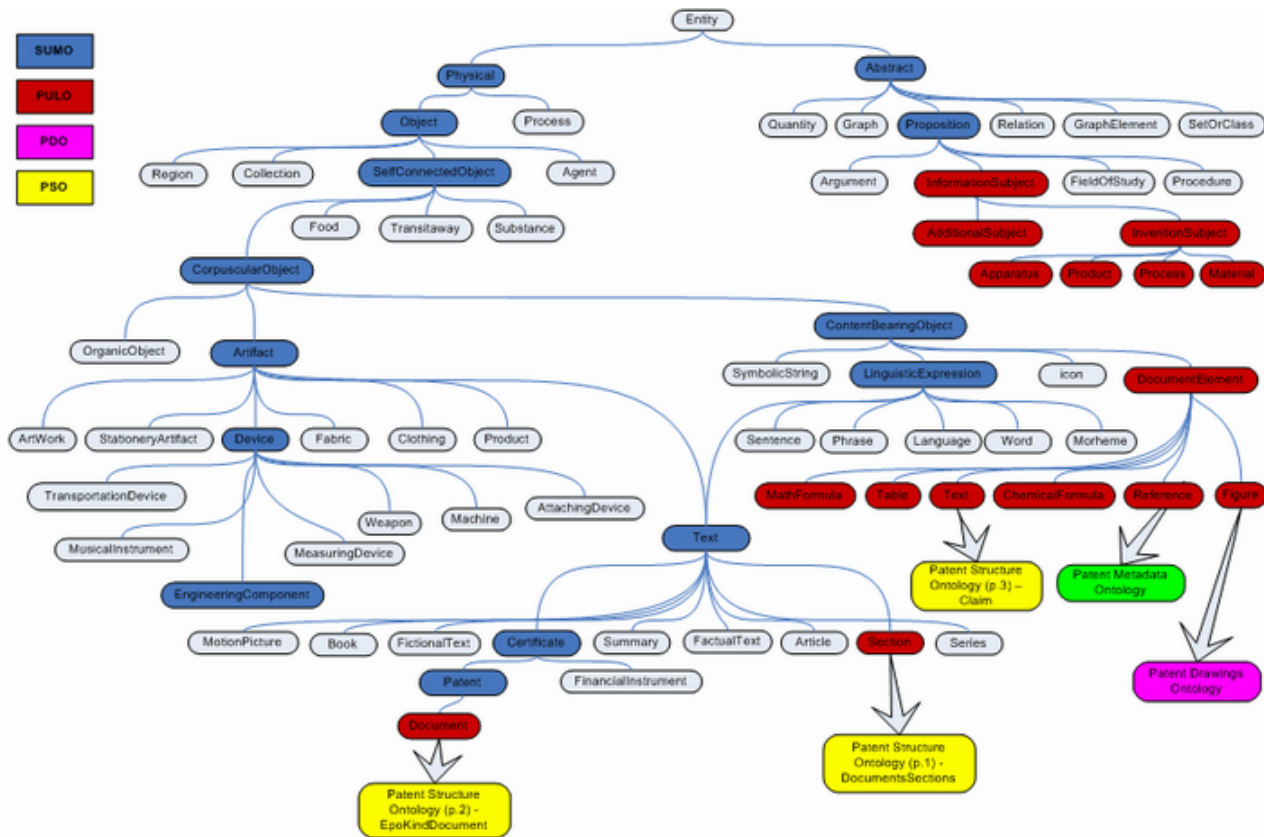
Social networks



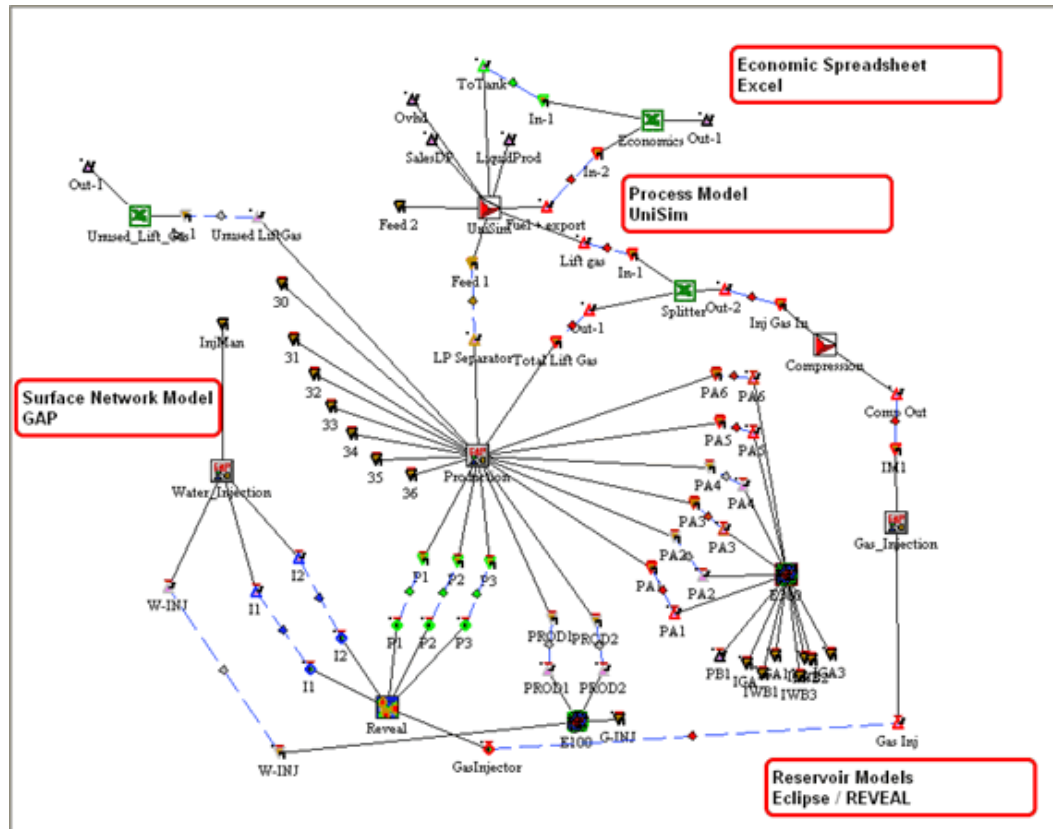
Concept maps



Ontologies

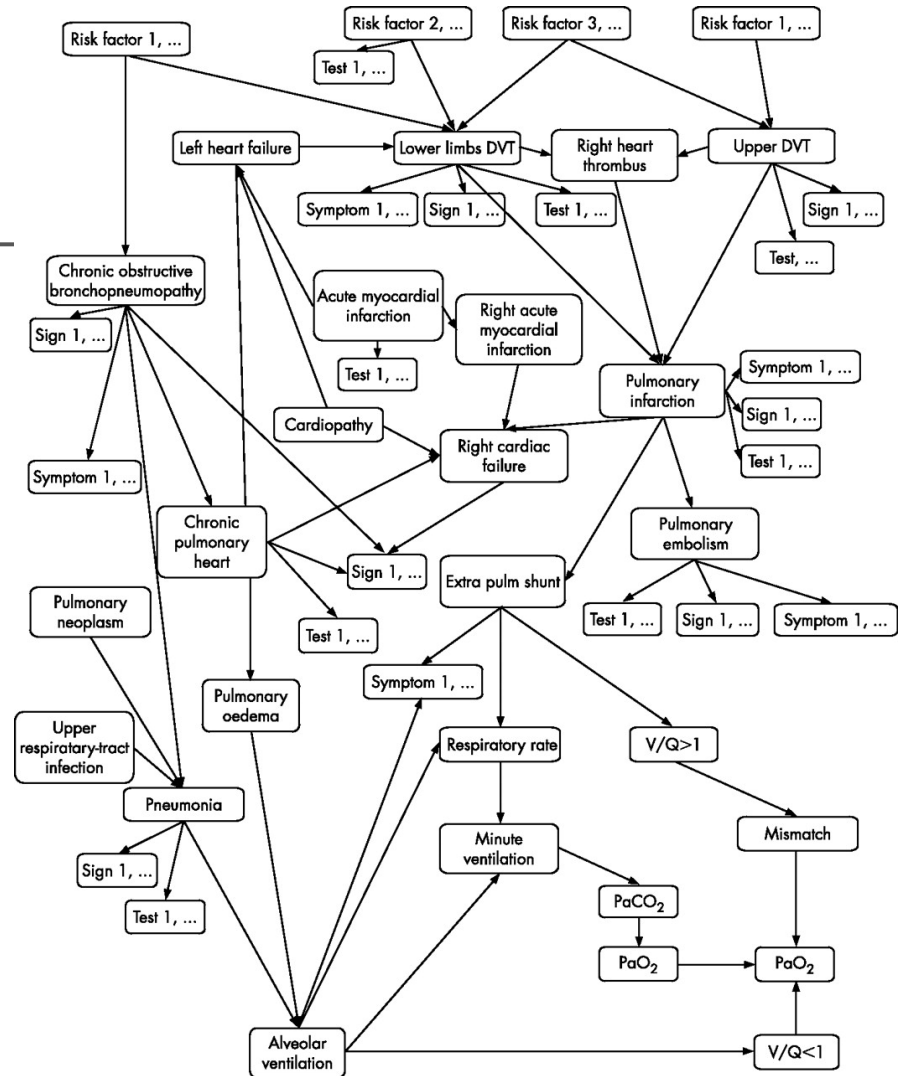


Simulation graphs



Probability maps

- Bayesian network



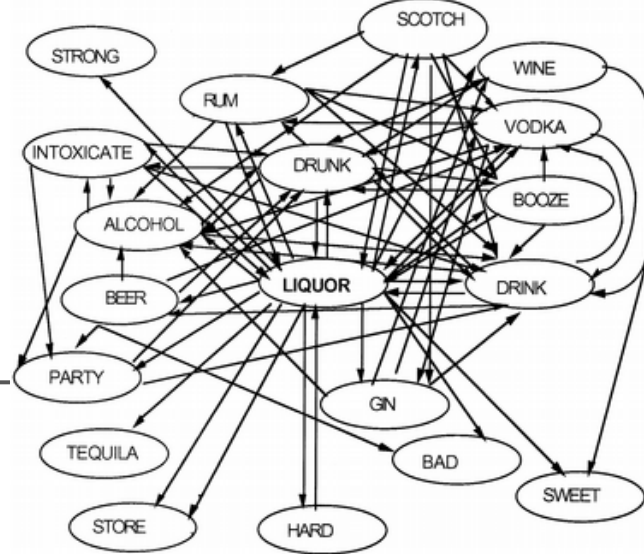
When is Graph Visualization Applicable?

- Ask the question: *is there an inherent relation among the data elements to be visualized?*
 - If YES – then the data can be represented by nodes of a graph, with edges representing the relations.
 - If NO – then the data elements are “unstructured” and goal is to use visualization to analyze and discover relationships among data.

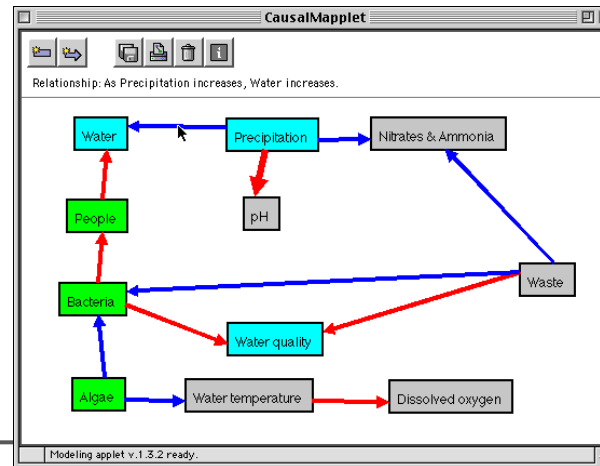
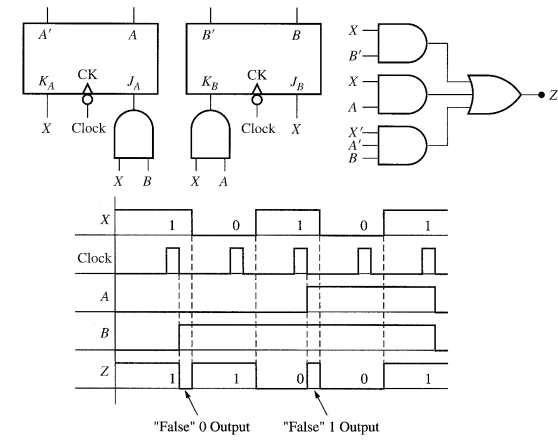
Source: Herman, Graph Visualization and Navigation in Information Visualization: a Survey

What kinds of relations?

This is speculative and incomplete !



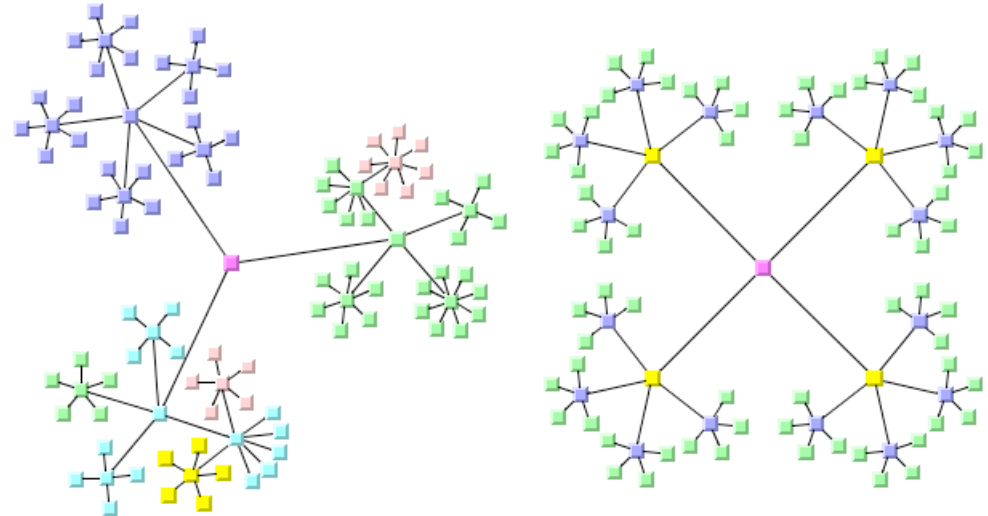
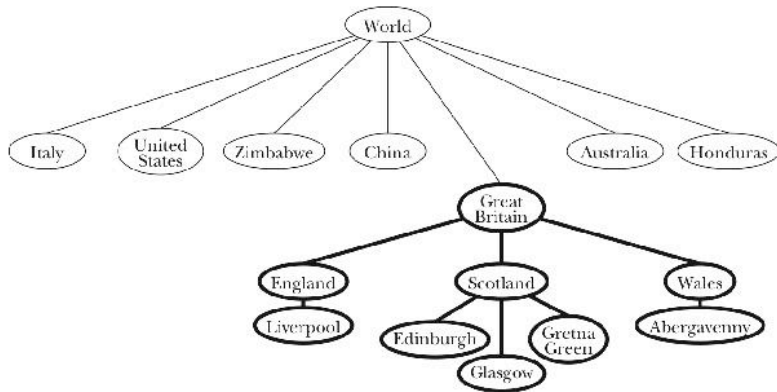
- Associative (IS-A)
 - A is like B in some way
- Ordinal/Sequential
 - A > B, precedes B, in some way
- Time
- Influence
 - Causal
 - dependency



Design Choices

- Spatial encoding and management
 - Position
 - Density (space filling)
- Visual features
- Properties of data points/sets
- Topology!
- Line marks for **connectivity** (links)
- Area marks for **containment** (hierarchy)
- Properties of relations

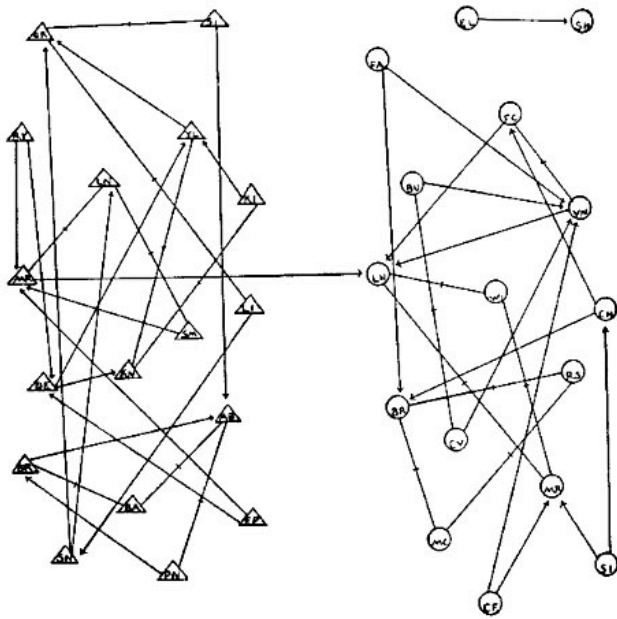
Node-link diagrams



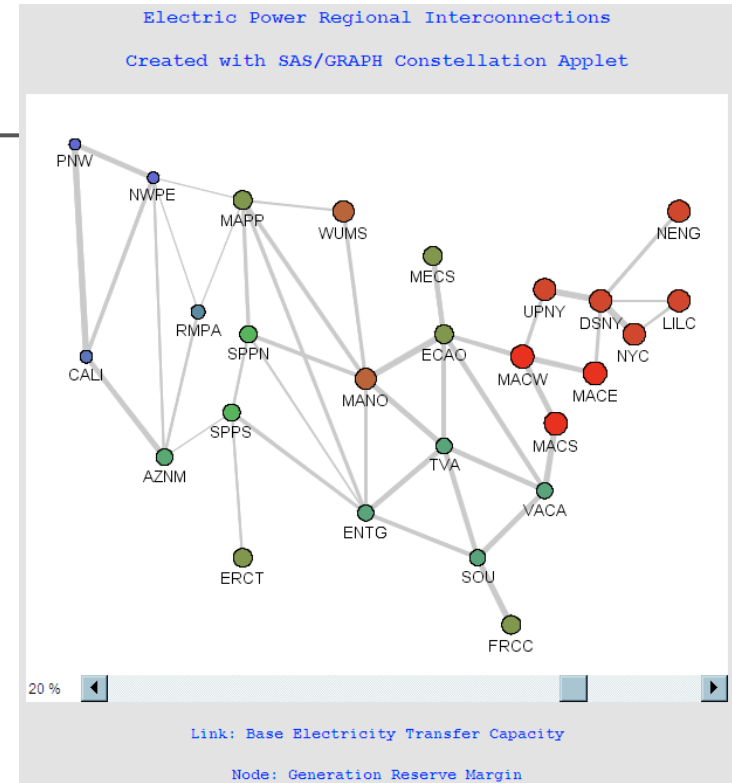
B. Harvey & M. Wright. Simply Scheme, Introducing Computer Science

[IBM Java Vis blog](#)

Node-link diagrams

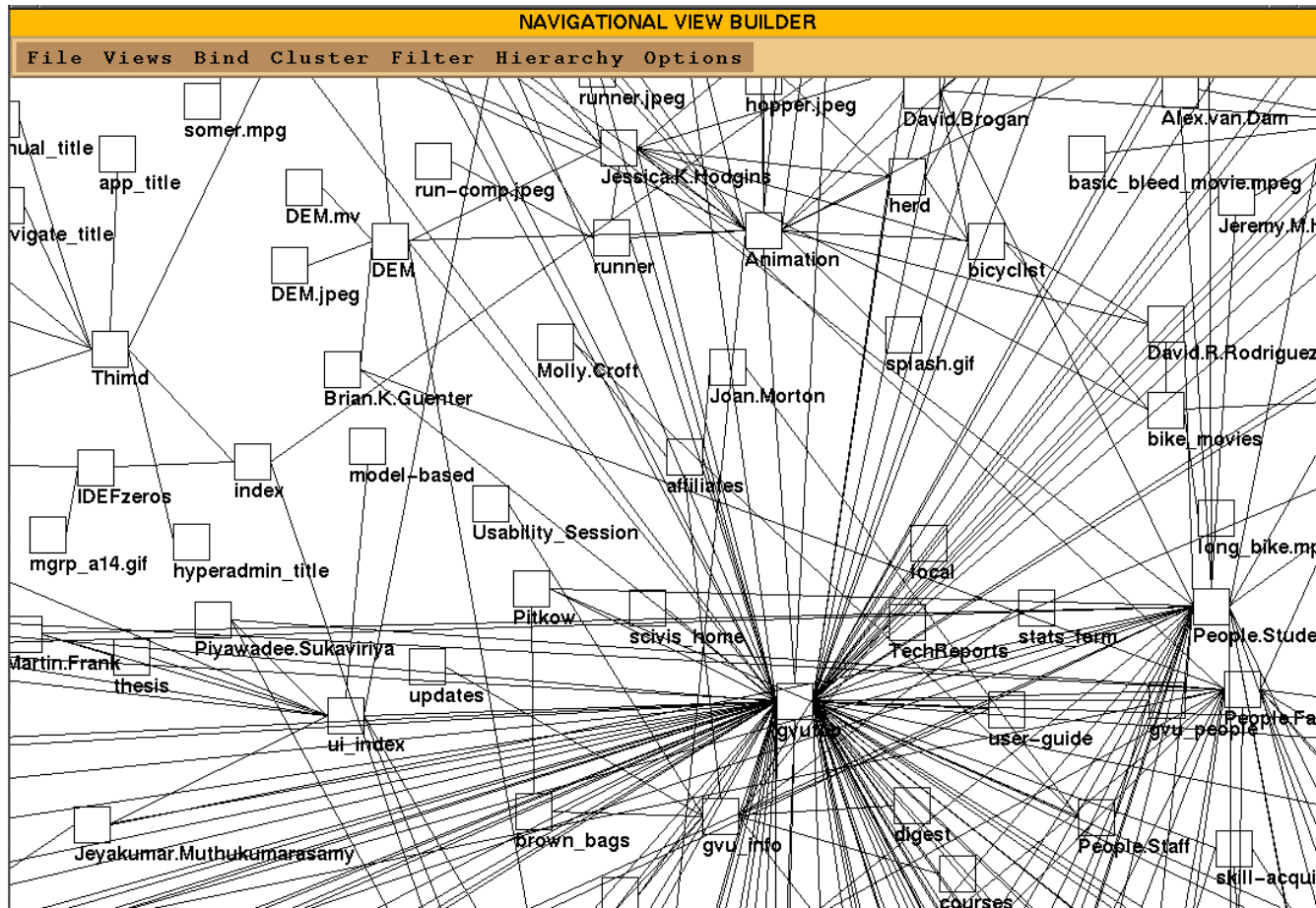


Blog.visual.ly

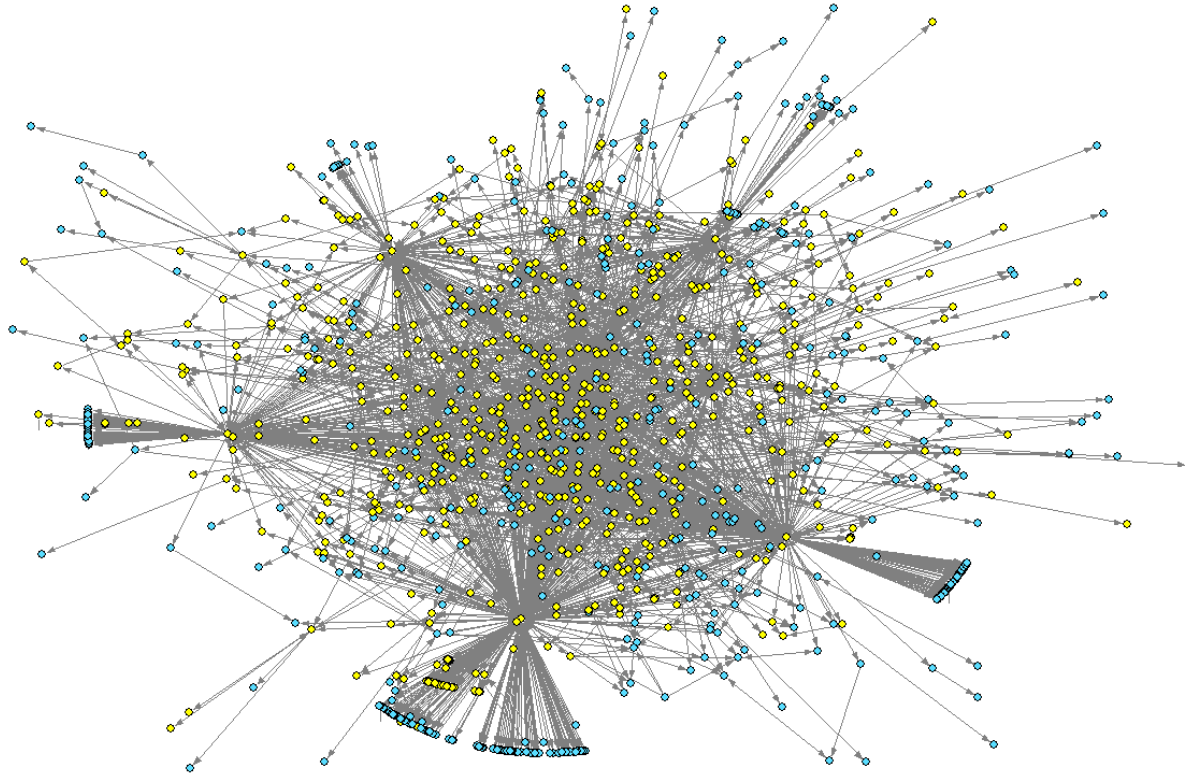


[SAS Constellation Tool](#)

Graphs: the problem



Graph Layout: The Problem



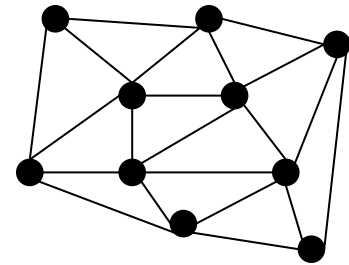
Topology-related tasks

- Finding all possible paths
- Shortest path
- Distance/proximity:
 - All adjacent nodes $<n>$ hops away
 - Degree (how many nodes connect in or out)
- Flow through the network (directional)
- Finding bridges that connect clusters or components

Graph and Tree Data Structures

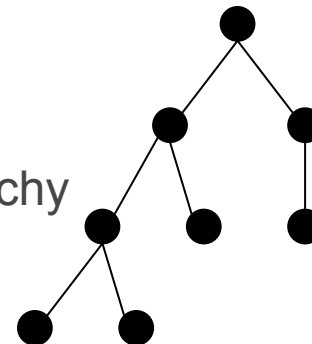
- **Graphs:**

- Representations of structured, connected data
- Consist of a set of **nodes** (data) and a set of **edges** (relations)
- Edges can be **directed** or **undirected**

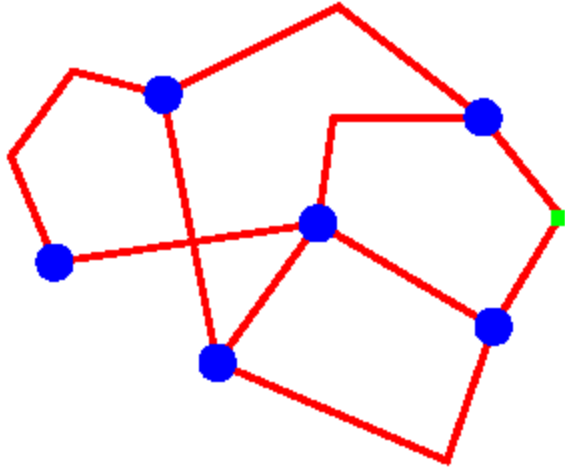


- **Trees:**

- Graphs with a specific structure
 - connected graph with $n-1$ edges
- Representations of data with natural hierarchy
- Nodes are either **parents** or **children**

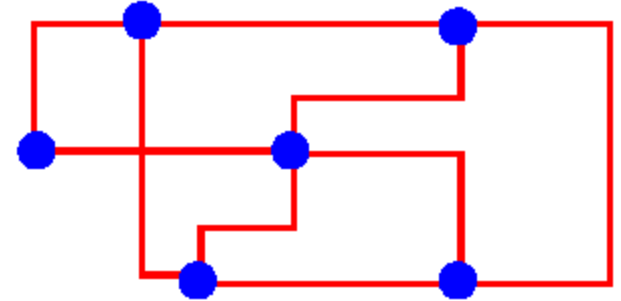


Traditional Graph Drawing

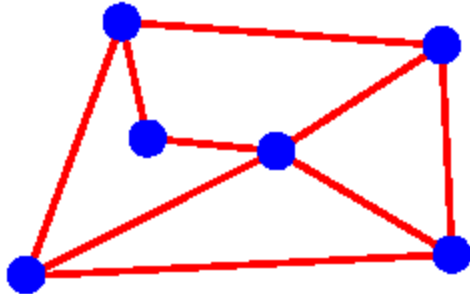


poly-line graphs (includes bends)

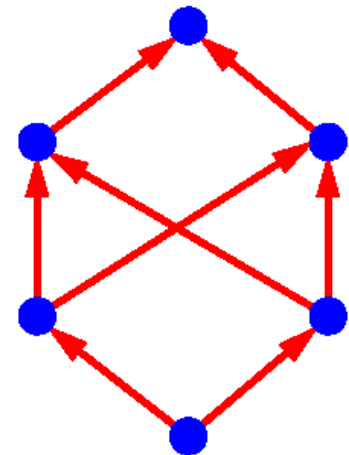
orthogonal drawing



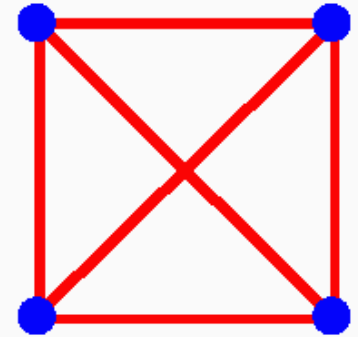
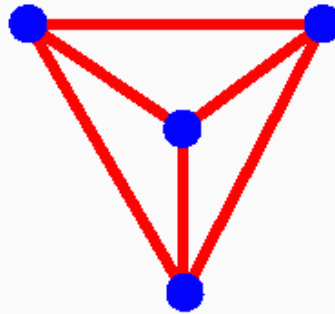
planar, straight-line drawing



upward drawing of DAGs

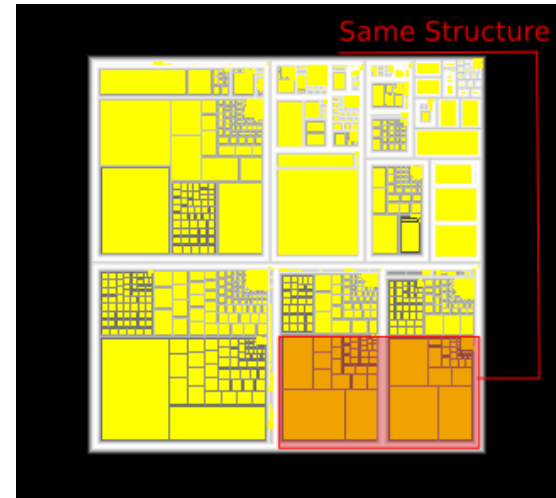
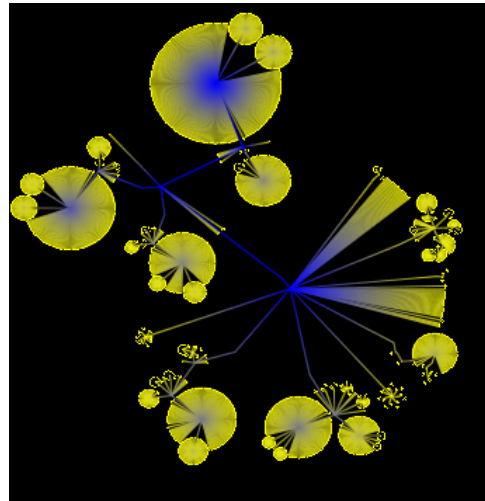
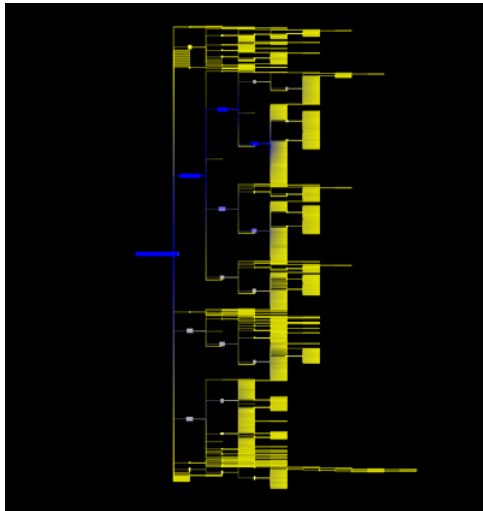


Aesthetic constraints



- Minimize link crossings
- Minimize link lengths
- Minimize link bends
- Maximize symmetries
- Minimize link crossings
- Mathematically difficult to do everything
- Often unsuitable for interactive visualisation
 - Approximation algorithms very complex
 - Unless you only need to compute layout once
- Precompute layout, or compute once at the beginning of an application then support interaction

-
- Tree with 3238 nodes



[3 different views of the same graph, made using Tulip](#)

Graph Layout

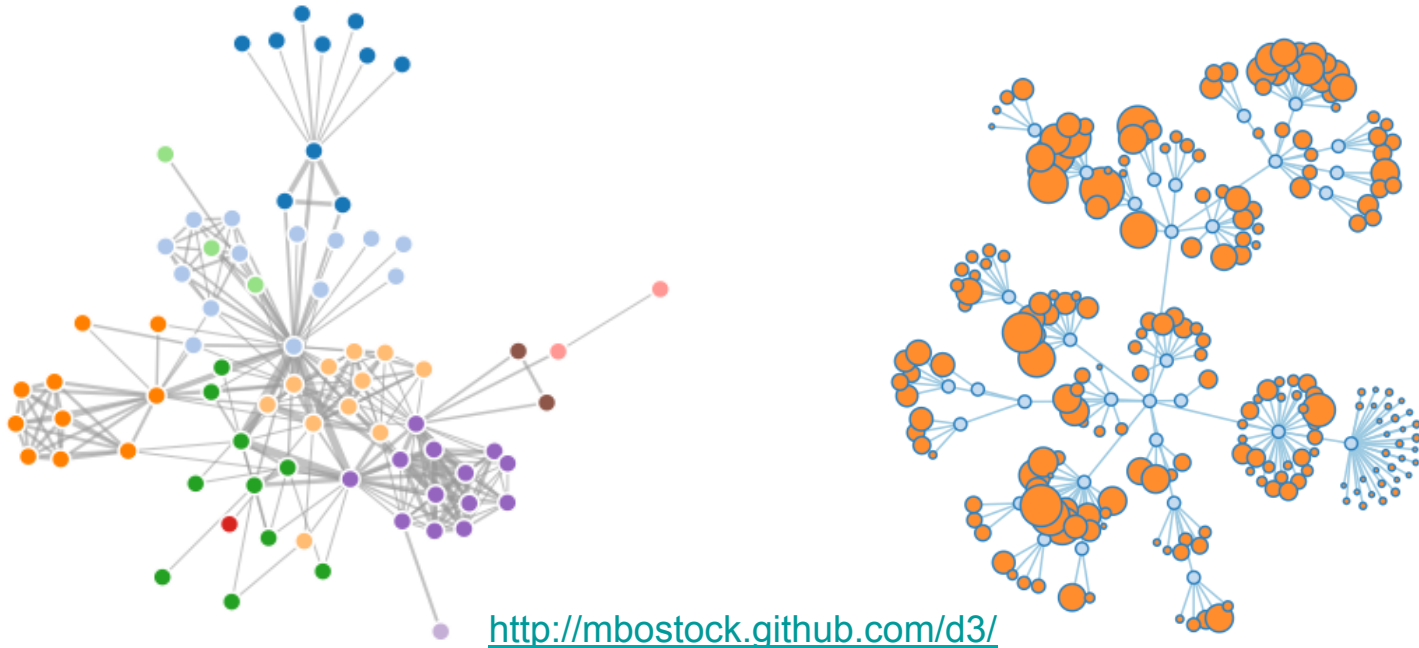
- How to position the nodes and edges?
 - Space does not directly encode VALUE
 - The **primary** concern with graphs
- Other visual features common
 - Colour
 - Size
 -

Layout Approaches

- Optimization-based techniques
 - Includes spring-embedding / force-directed layout
- Hierarchical graph layout
 - Tree-ify the graph - then use tree layout
- Radial graph layout
- Adjacency matrices
- Structurally-independent layout
- On-demand revealing of subgraphs
- Distortion-based views
 - Hyperbolic browser

Force-directed placement

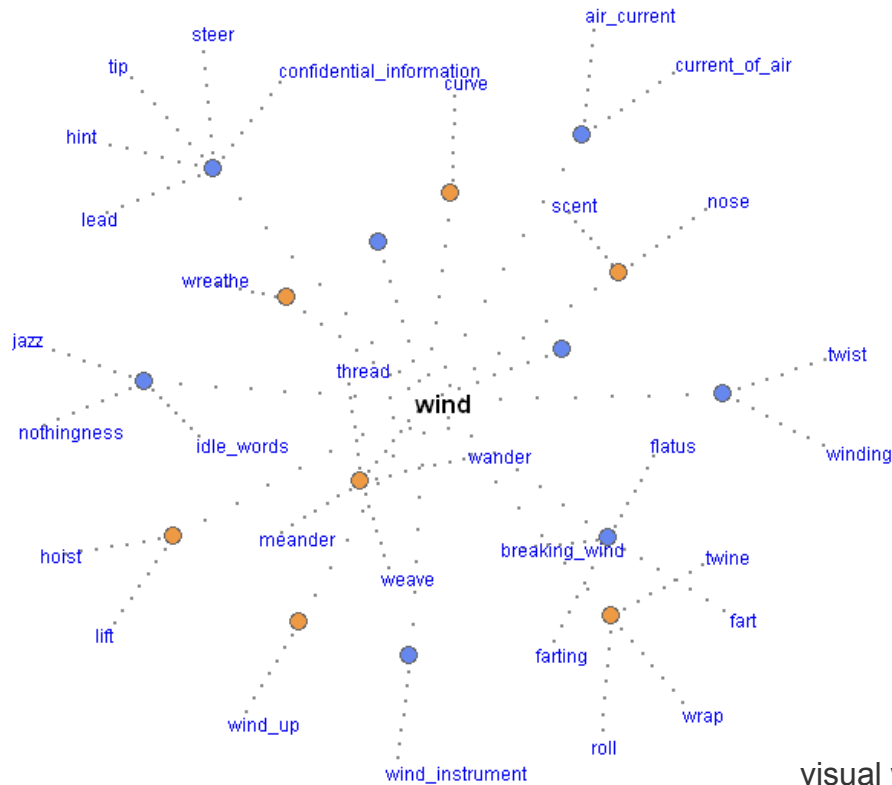
- D3 graphs using force directed layout, colour and size



Optimization-based layout

- Specify constraints for layout
 - Series of mathematical equations
 - Hand to “solver” which tries to optimize the constraints
- Examples
 - Minimize edge crossings, line bends, etc
 - Multi-dimensional scaling (preserve multi-dim distance)
 - Force-directed placement (use physics metaphor)
- Benefits
 - General applicability
 - Often customizable by adding new constraints

Example: Force-Directed Layout



Uses physics model to layout graph,
Nodes repel each other, edges act as springs, and some amount of friction or drag force is used.

Special techniques to dampen “jitter”.

visual wordnet <http://www.kylescholz.com/projects/wordnet>

visuwords <http://www.visuwords.com/>

Force-directed placement

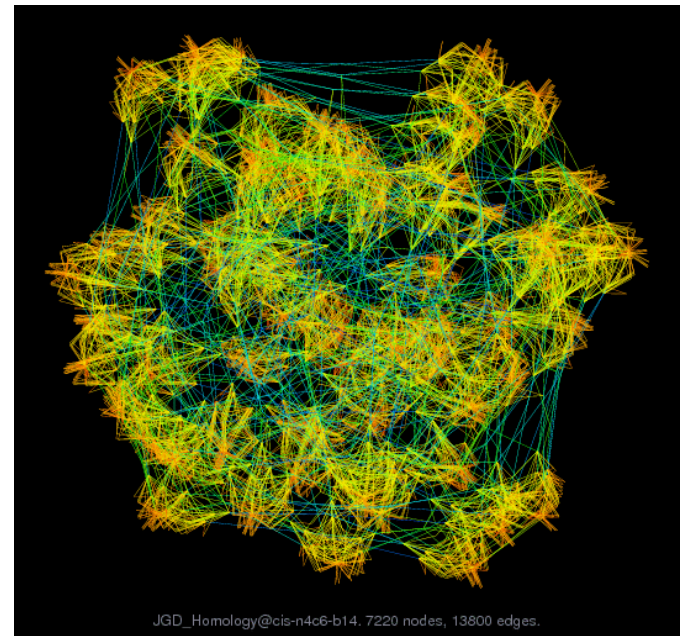
- Widely used
- Use an algorithm of “springs” and “forces” to iteratively position nodes to some specified tolerance
- Simple to explain and implement
- Clustering “behaviour”
 - exploits perceptual cue of proximity
 - can be due to strong links **or** strong repelling (dissimilar) forces: ambiguous

Force-directed placement: issues

- Scalability
- Nondeterministic : never the same graph twice!
- Layout algorithm can be **brittle**:
 - Gets stuck in non-optimal configuration
 - Requires lots of parameter tweaking
- Jittery: algorithm doesn't "settle"
- Approximate constraint satisfaction
- Running time; "organic" look not always desired

Multi-level layouts

- **Coarsen** network into derived hierarchy of successively simpler networks to capture the most essential aspects of the original's structure
- Layout simplest version first, successively refine
- Introduce implicit **hierarchy**

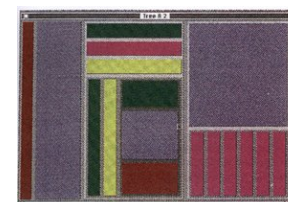
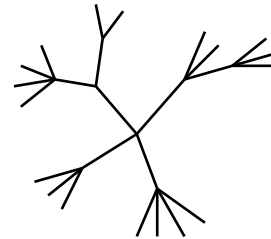
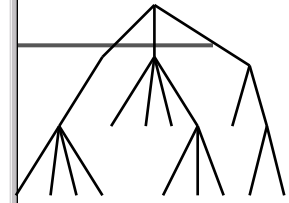
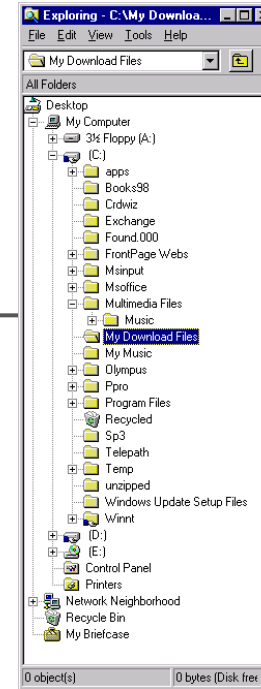


Tree layouts

- Topological structure with hierarchy
- Parent-child relations are paramount
- can have explicit sibling links but usually implicit

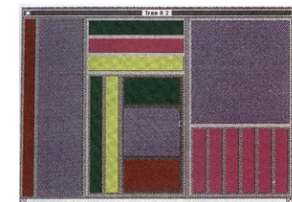
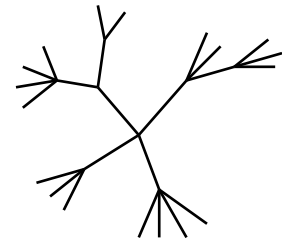
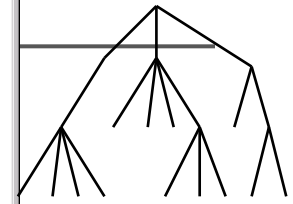
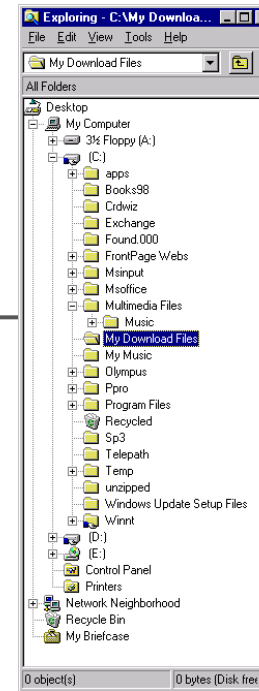
Hierarchical Information

- Pervasive
 - File / Directory systems on computers
 - Classifications / Taxonomies / Controlled Vocabularies
 - Software Menu structure
 - Organization charts
 - ...

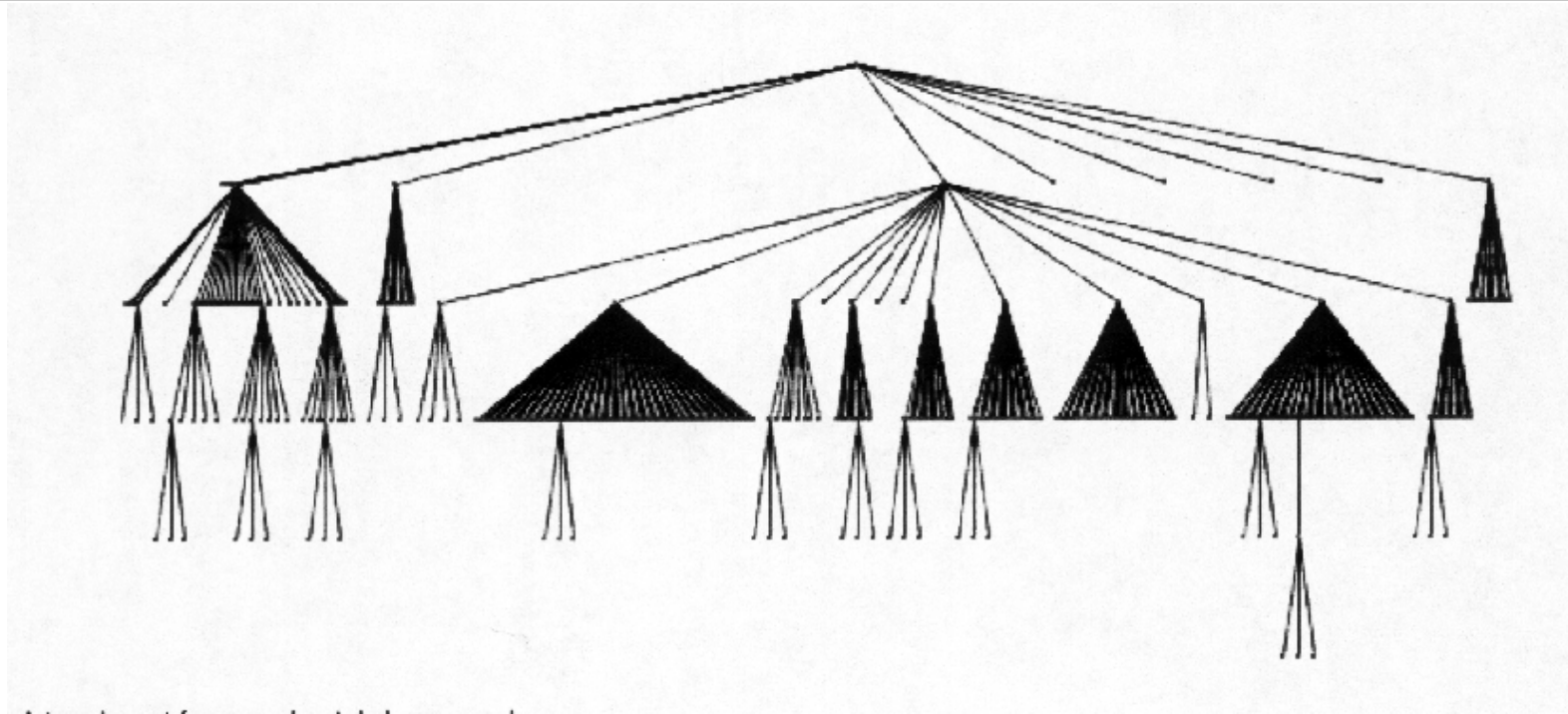


Hierarchical Information

- Indented Outlines
 - Good for Searching Bad for Structure
- Node-Link Trees
 - Top-to-Bottom Layout
 - 2D
 - 3D : ConeTree
- Radial Layout
 - 2D : SunBurst, Hyperbolic Trees
 - 3D : H3 & Walrus
- Space-Filling Treemaps



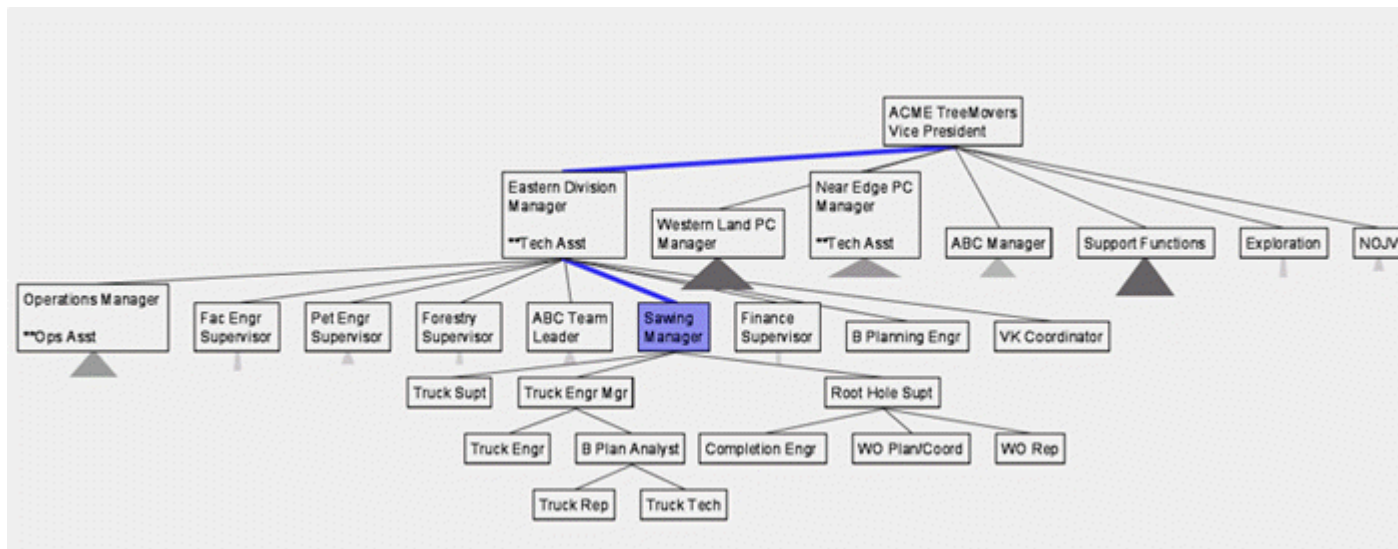
Hierarchical Data – Traditional Node-Link Layout



Allocate Space proportional to # of Children at Different Levels

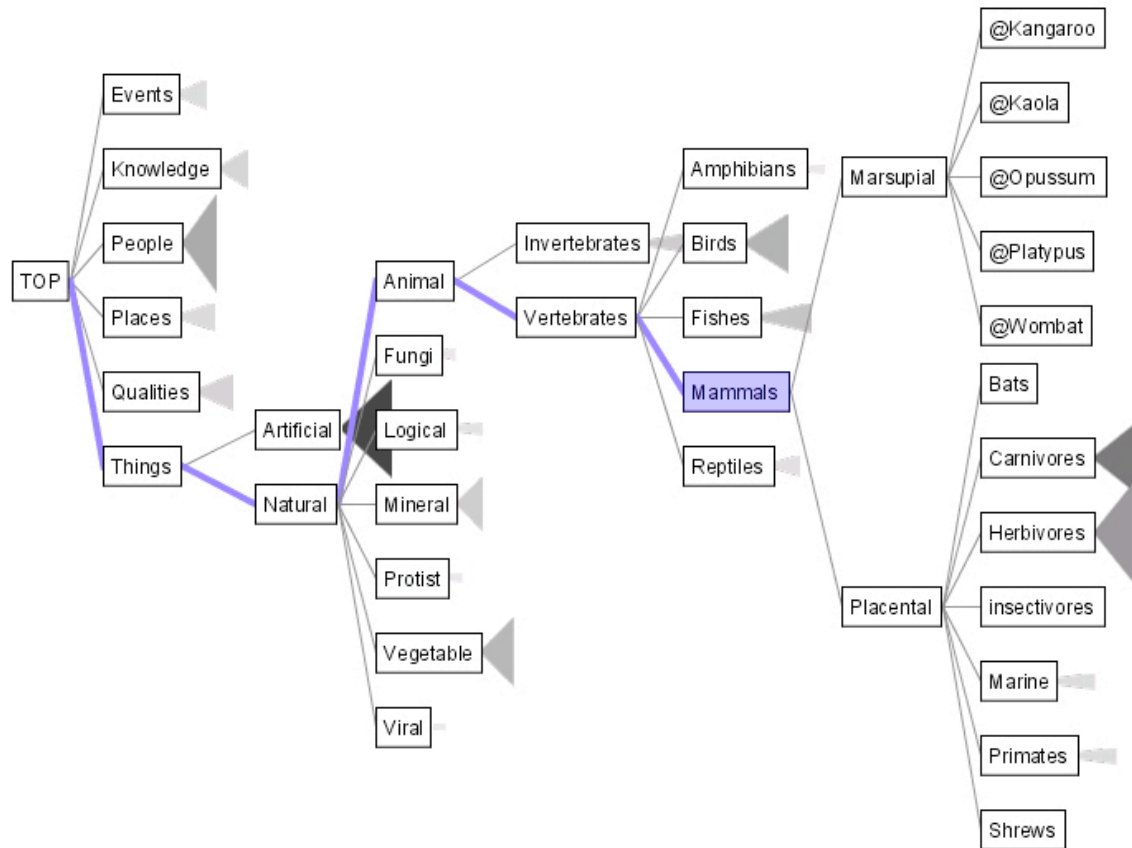
Traditional Node-Link Layout → SpaceTree

- HCI Lab – University of Maryland <http://www.cs.umd.edu/hcil/spacetree/>



<http://www.scils.rutgers.edu/~aspoerri/Teaching/InfoVisResources/videos/>

Space Tree cont.



3D ConeTree

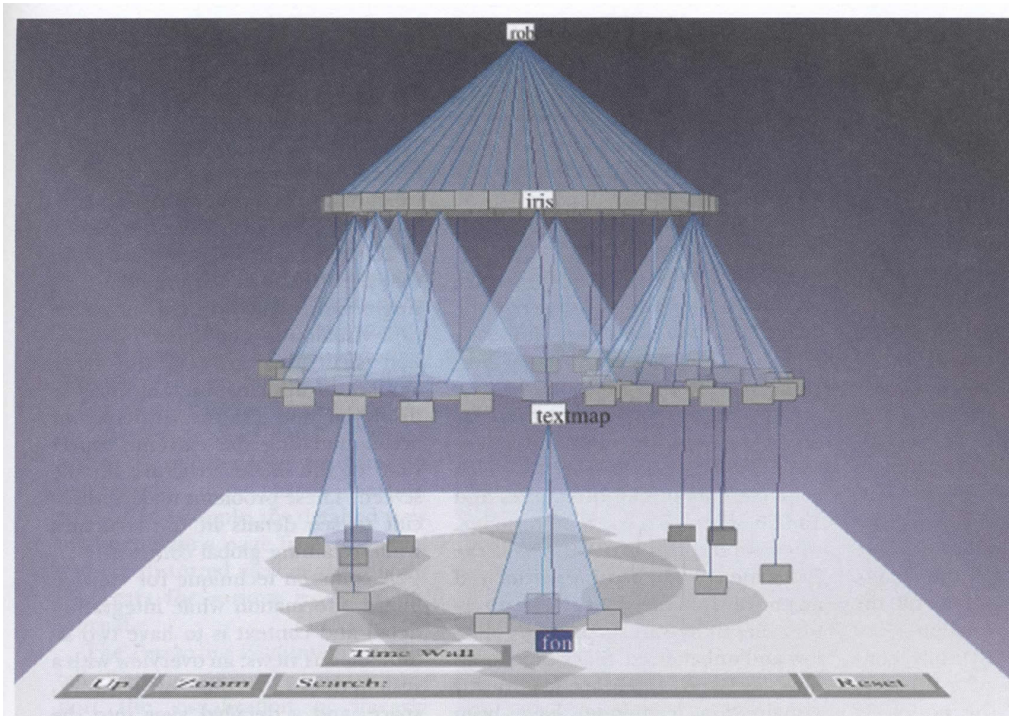


FIGURE 7

Cone Tree visualization of a directory hierarchy

- Positive
- Higher Information Density
- Smooth animation

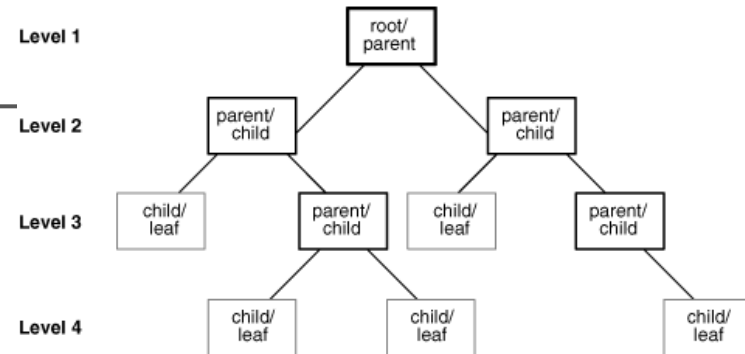
- Negative
- Occlusion
- Non-trivial to implement
- Requires horsepower

3D used to increase Information Density

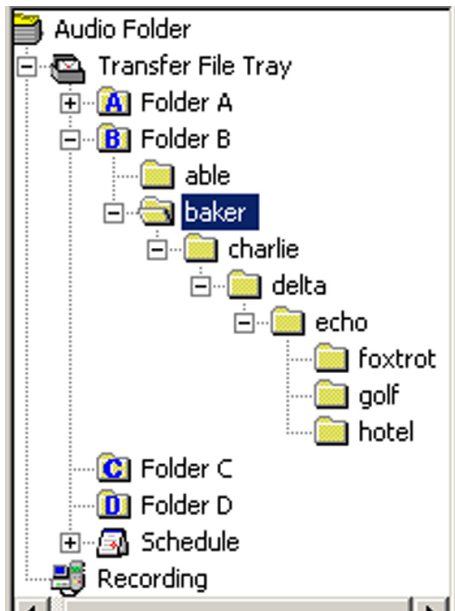
Children laid out in a cylinder “below” parent

Tree-link diagrams

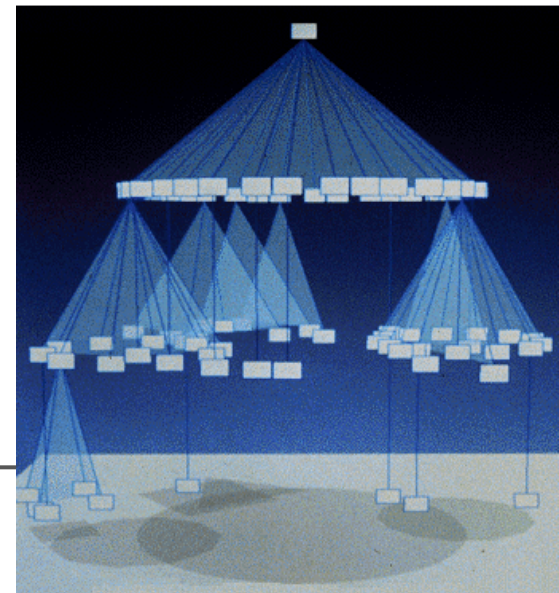
Classical representation



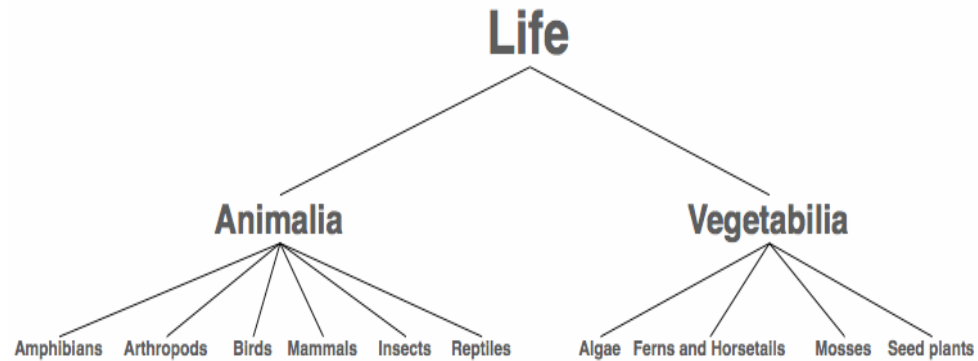
Tree view



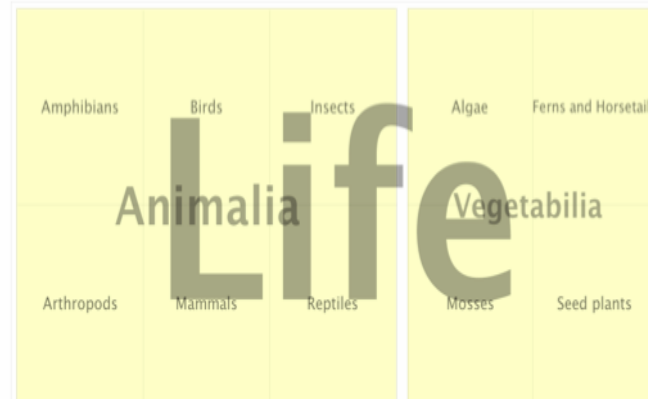
Cone tree



Tree – containment diagrams

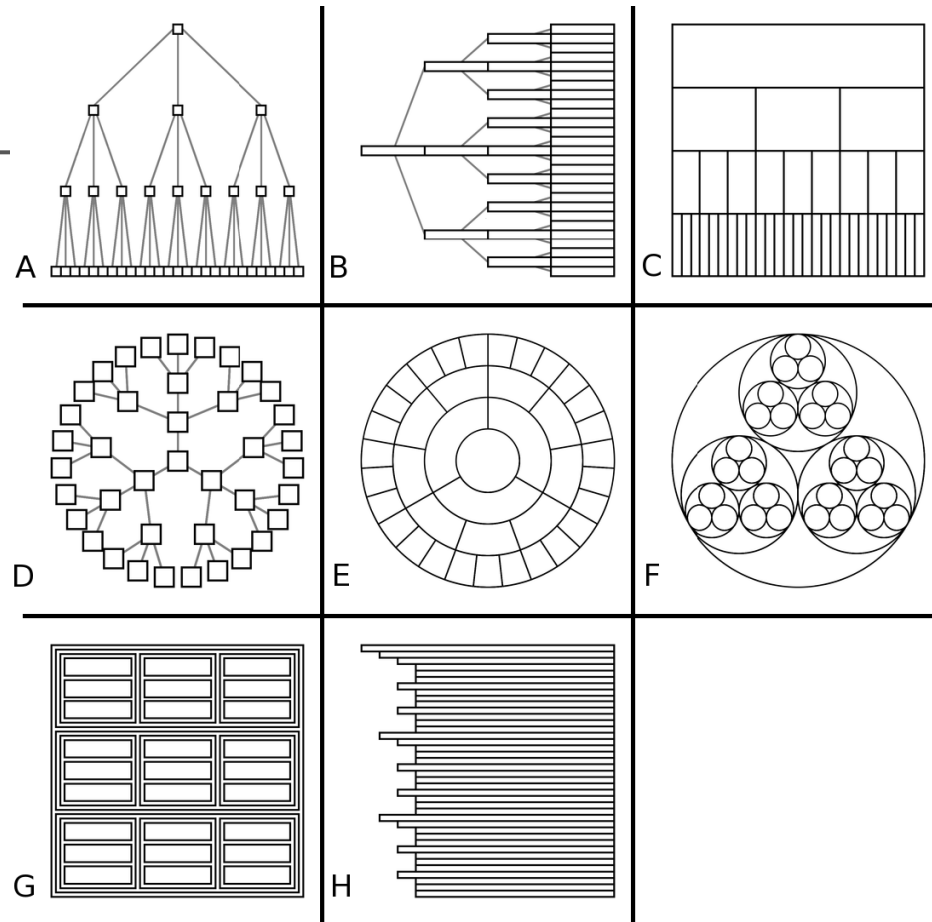


Treemap



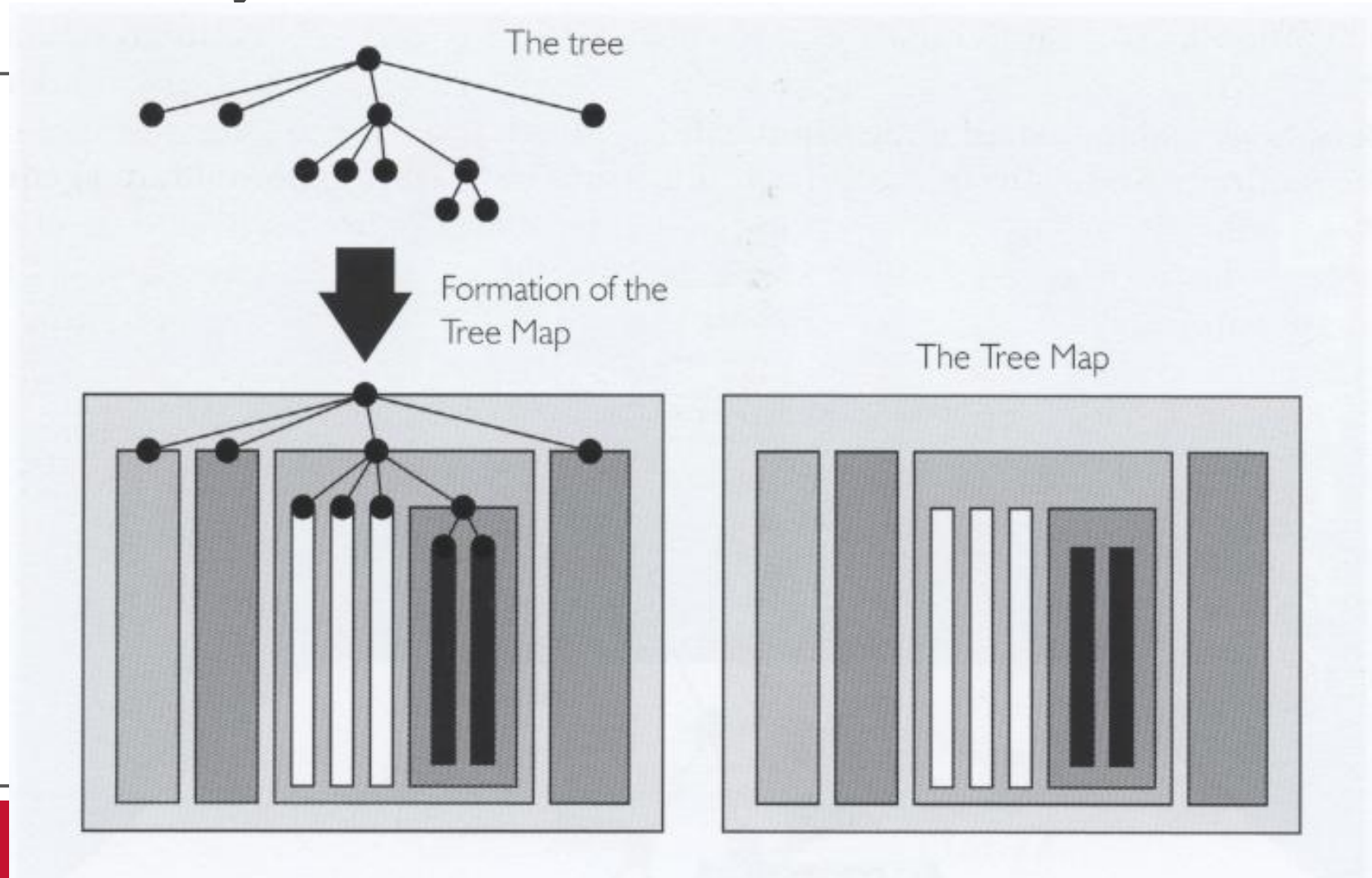
Tree layouts

- Eight visual encoding idioms showing the same tree dataset, using different combinations of visual channels.

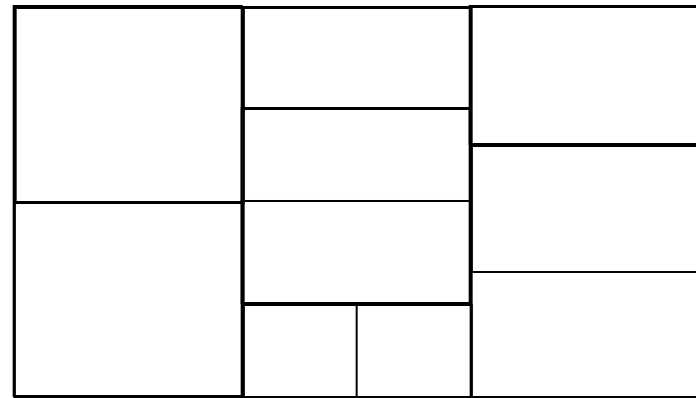
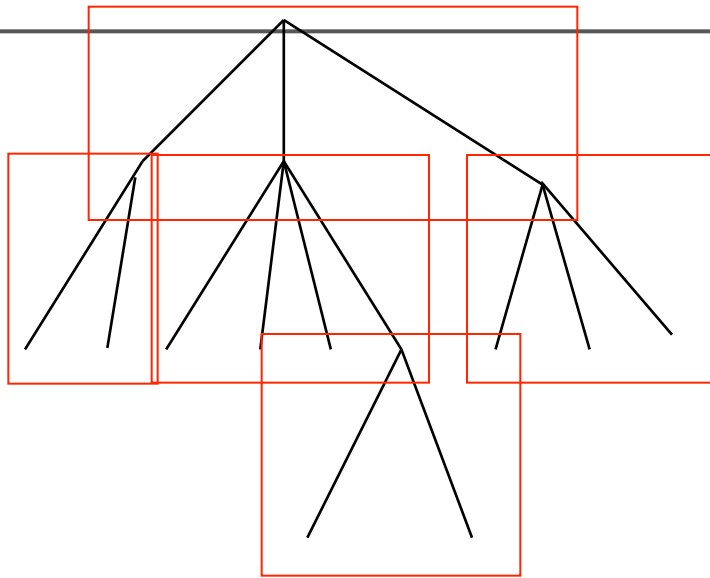


[T. Munzner, Visualization Analysis and Design]

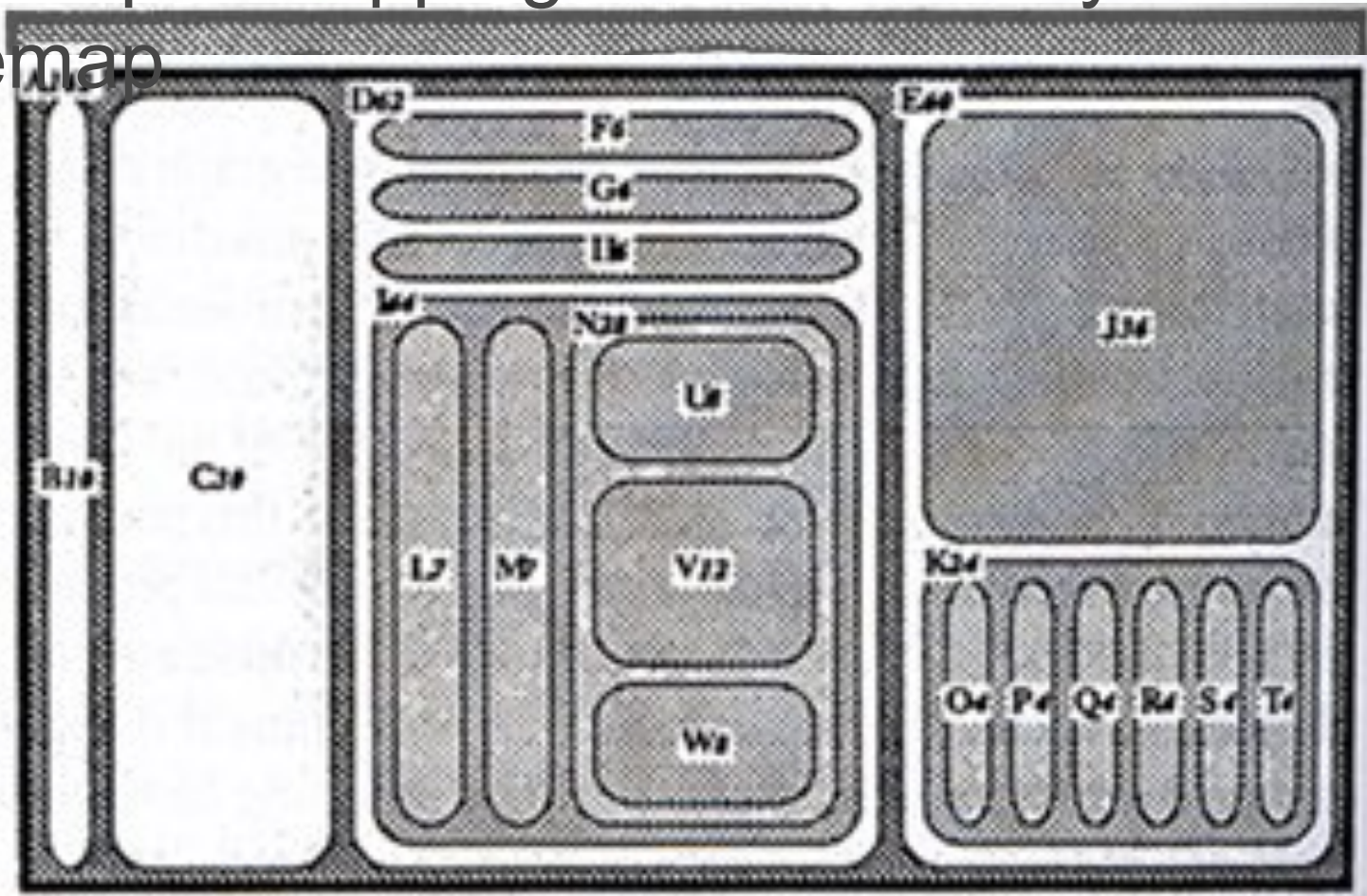
Tree layouts: containment



Treemaps – “Slice & Dice”



Treemap – Mapping Traditional Layout into Treemap



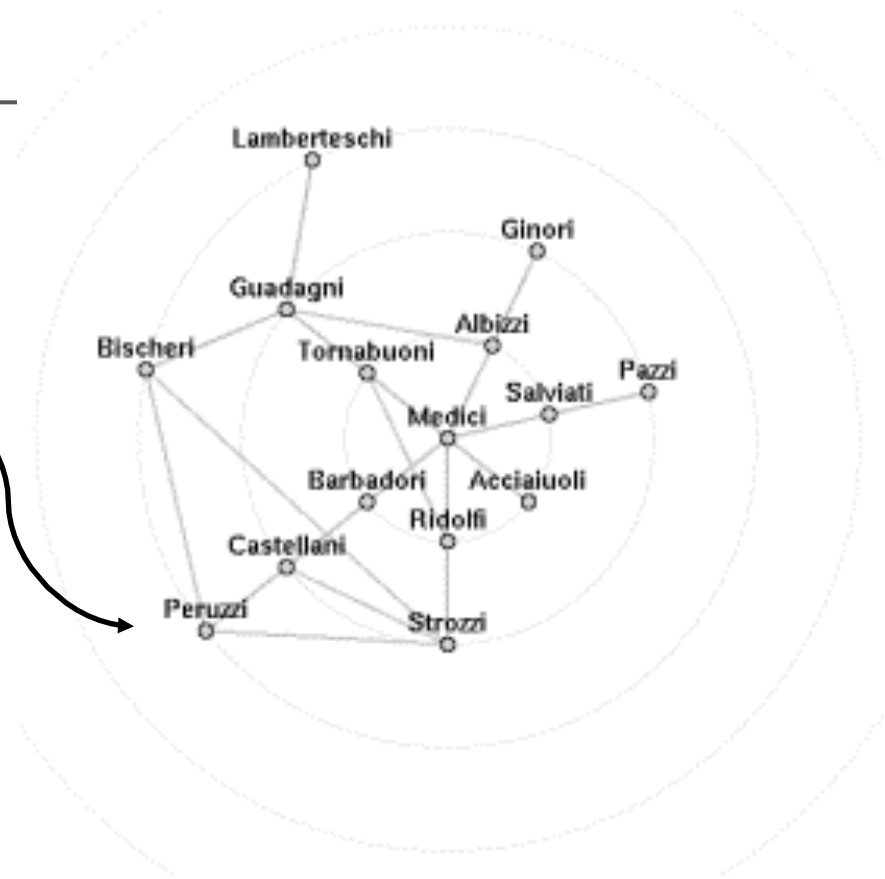
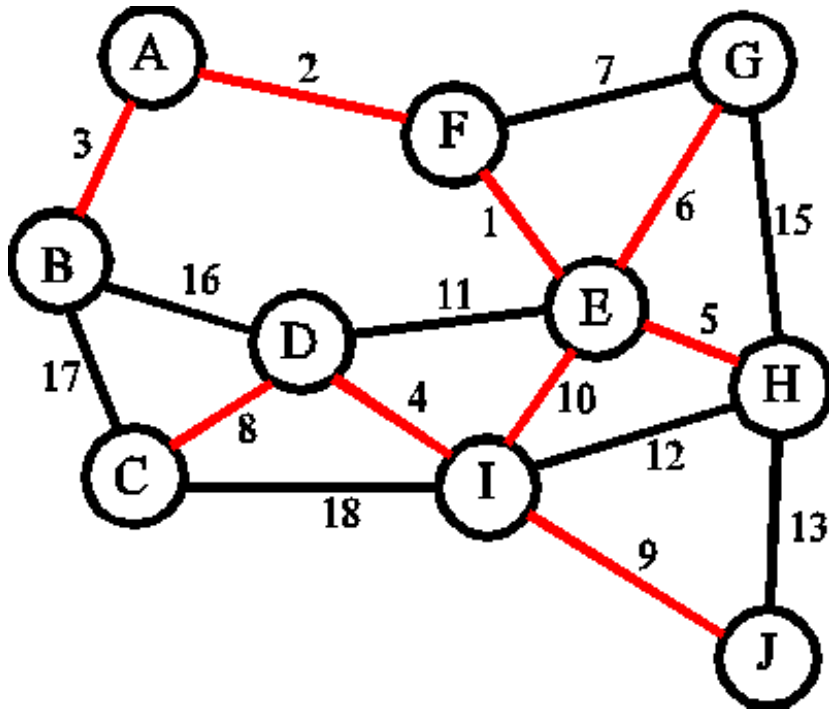
Treemaps

- Which Problem do Treemaps aim to address?
 - Visualize hierarchical structure as well as content of (atom) nodes
- What are Treemaps' main design goals?
 - **Space-filling** (High Data / Ink Ratio)
 - “Structure” is represented using **Enclosure / Containment**
 - “Content” is represented using **Area**
- Pre-attentive, Early Visual Processes Used?
 - Position, Size = Area, Color and Containment

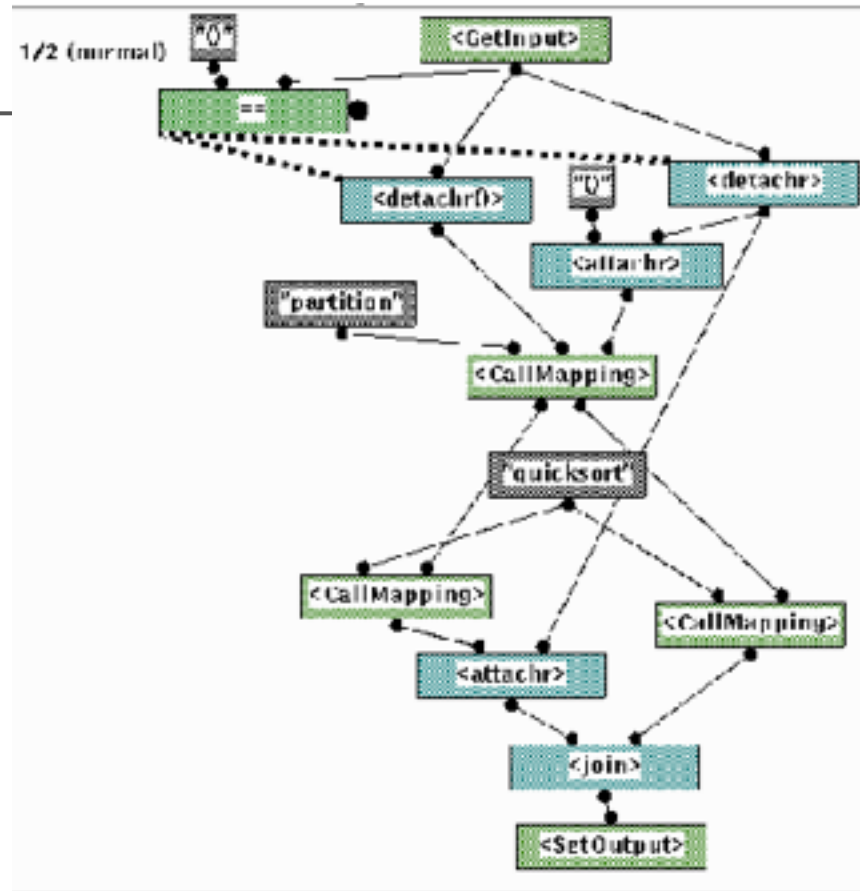
Tree-based graph layout

- Select a tree-structure out of the graph
 - Breadth-first-search tree
 - Minimum spanning tree
 - Other domain-specific structures
- Use a tree layout algorithm
- Benefits
 - Fast, supports interaction and refinement
- Drawbacks
 - Limited range of layouts

Tree-ify the graph



Tree-ify the graph

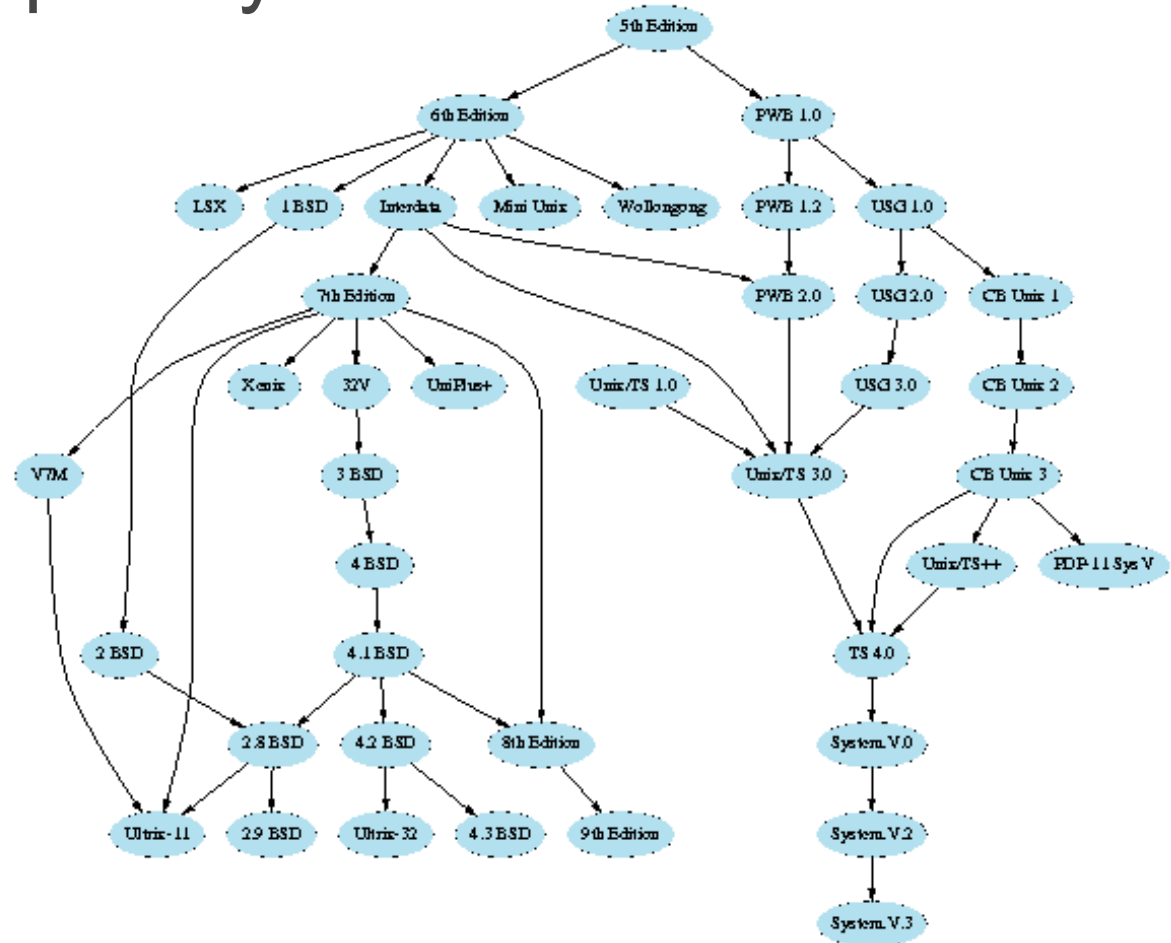


Hierarchical graph layout

- Use directed structure of graph to inform layout
- Order the graph into distinct levels
 - this determines one dimension
- Now optimize within levels
 - determines the second dimension
 - minimize edge crossings, etc
- The method used in graphviz's "dot" algorithm
- Great for directed acyclic graphs, but often misleading in the case of cycles

Hierarchical Graph Layout

- Evolution of the UNIX operating system
- Hierarchical layering based on descent

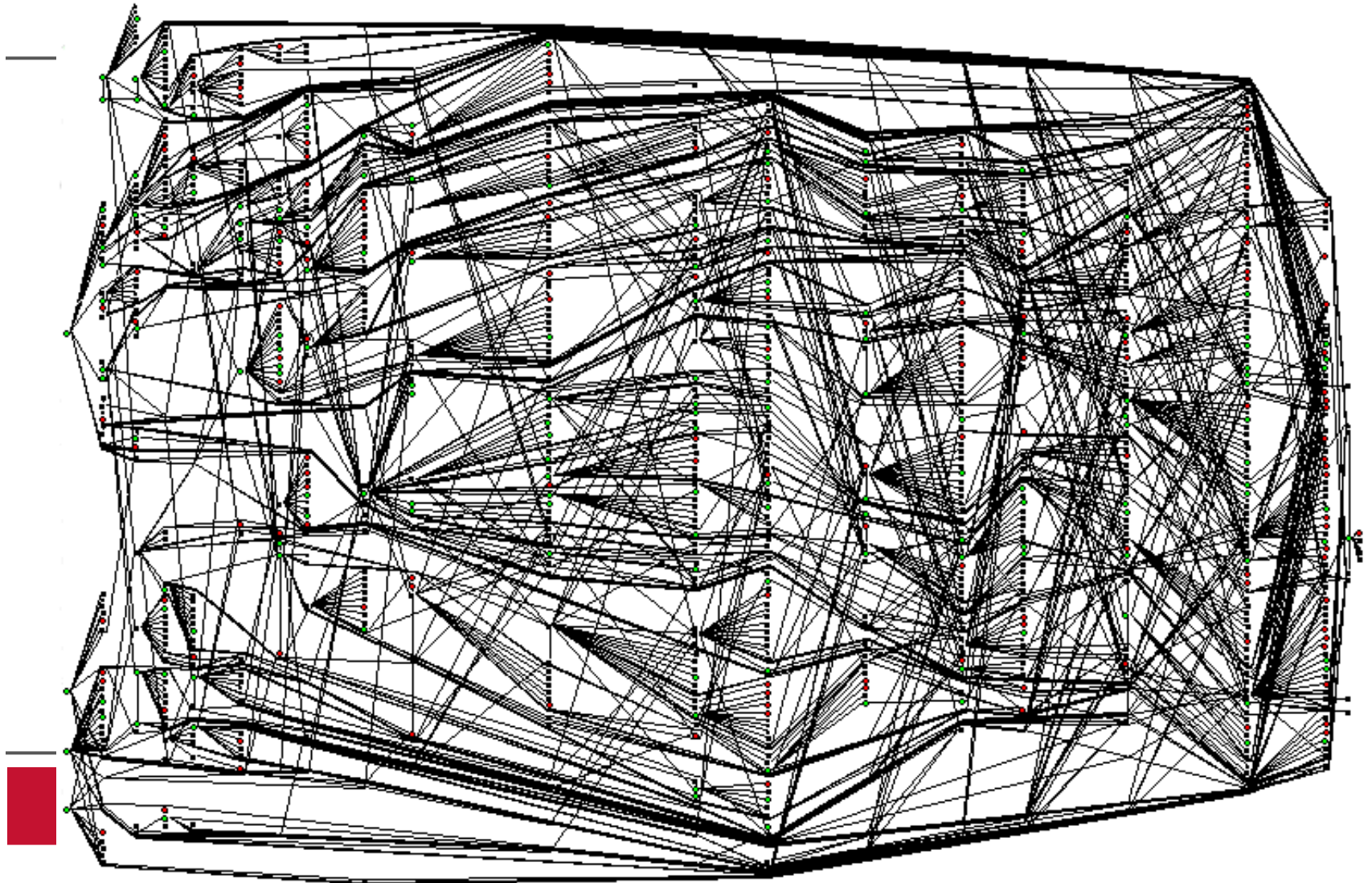


Partial Map of Gnutella Network - 7/27/00

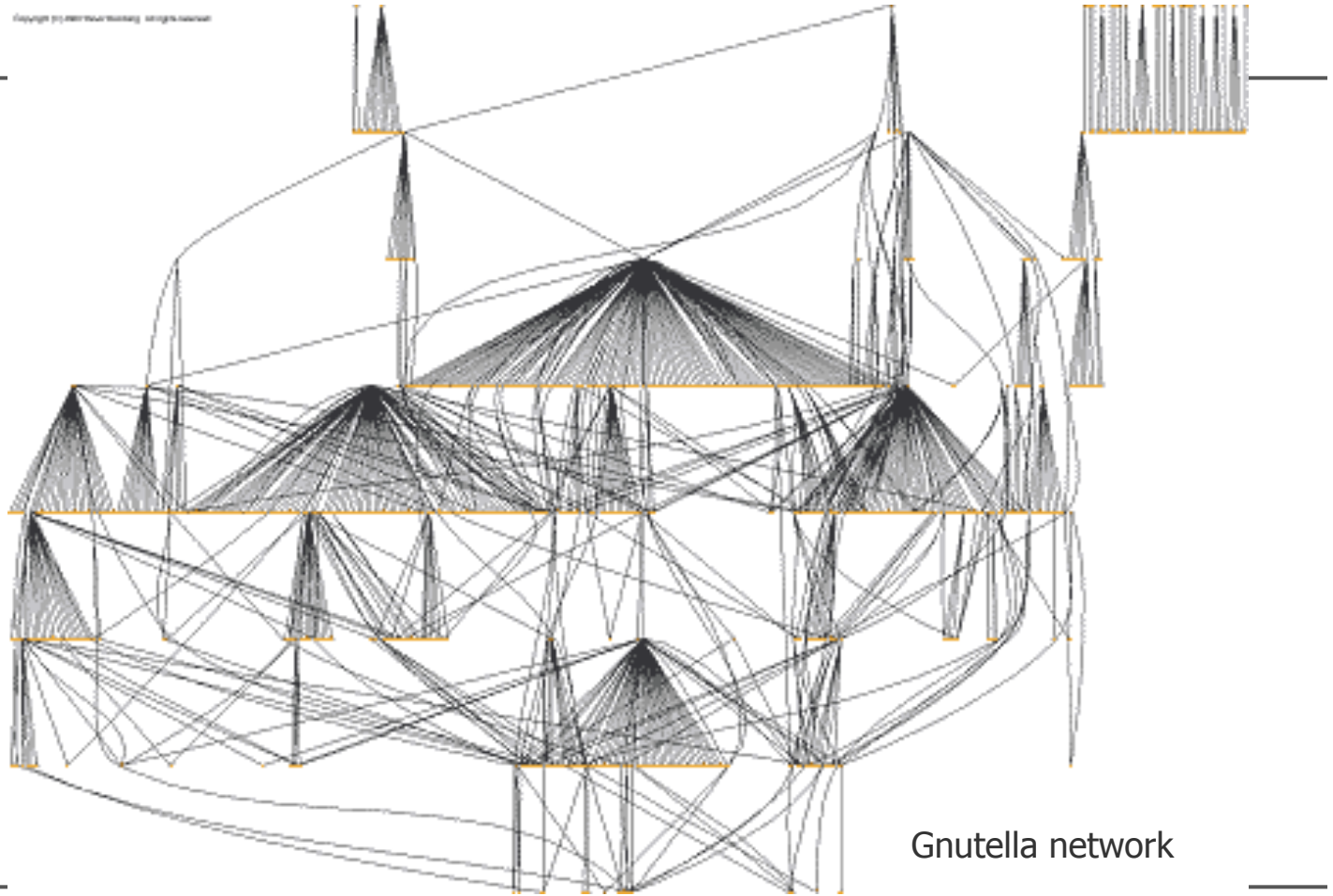
Clip2 Distributed Search Services

<http://dss.clip2.com>

(c)2000 Clip2.com, Inc.

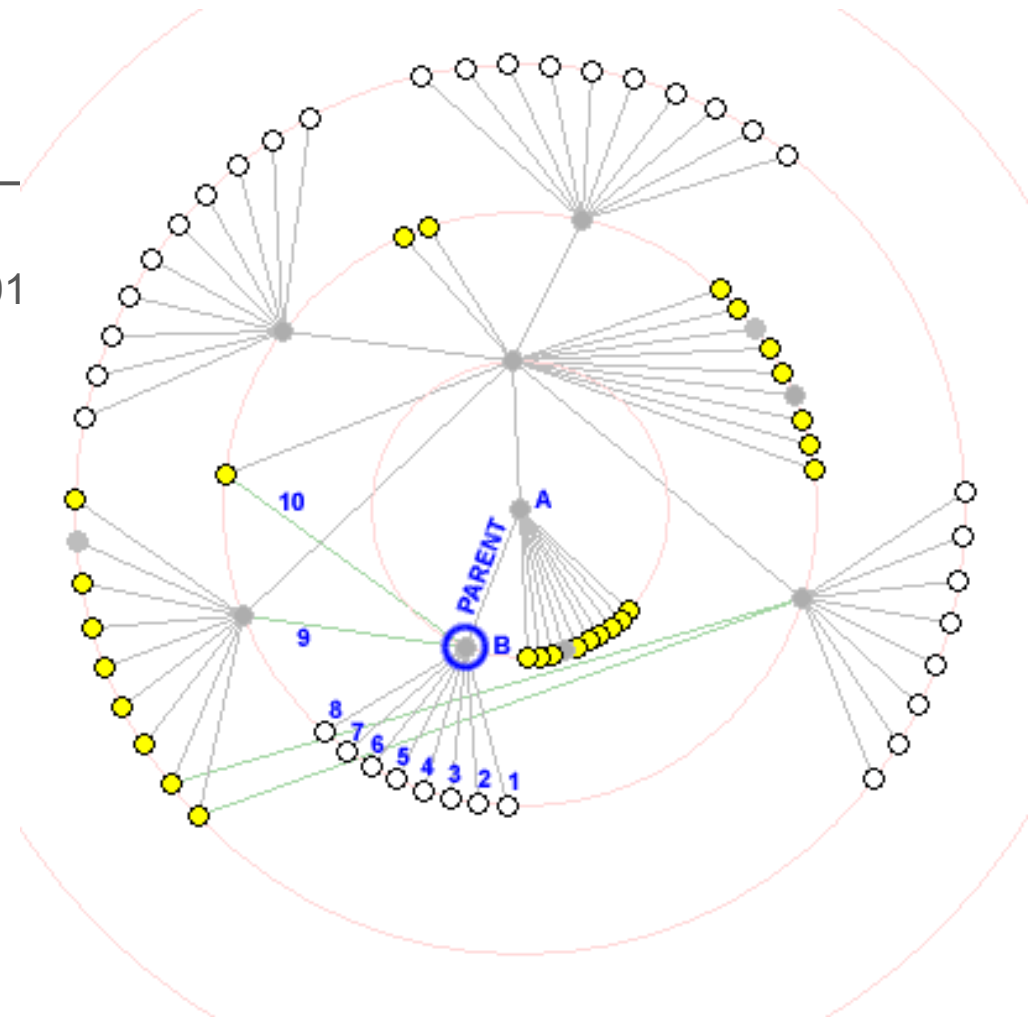
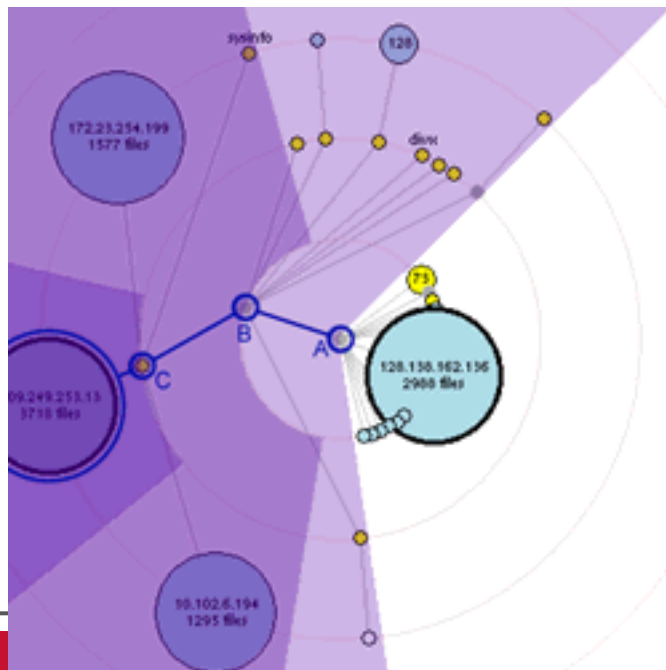


Hierarchical graph layout

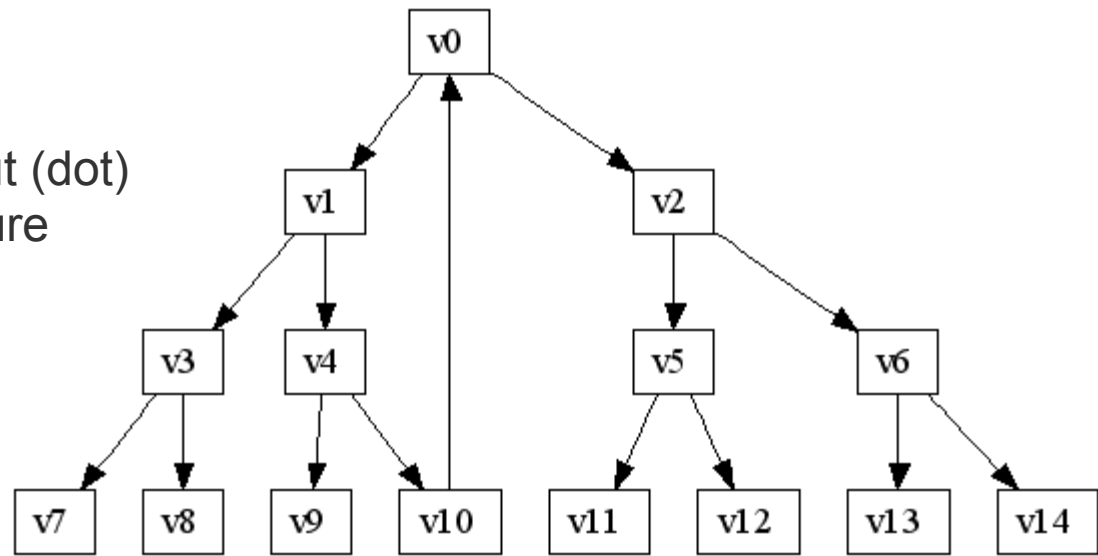


Radial Layout

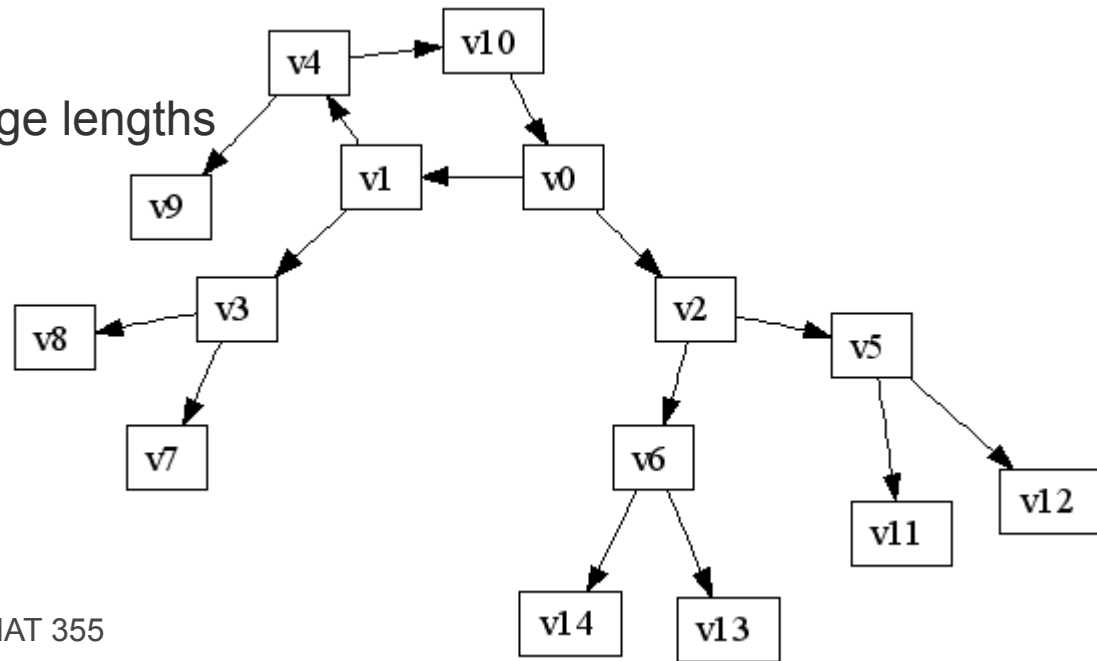
- **Animated Exploration of Graphs with Radial Layout**, Yee et al., 2001
- Gnutella network



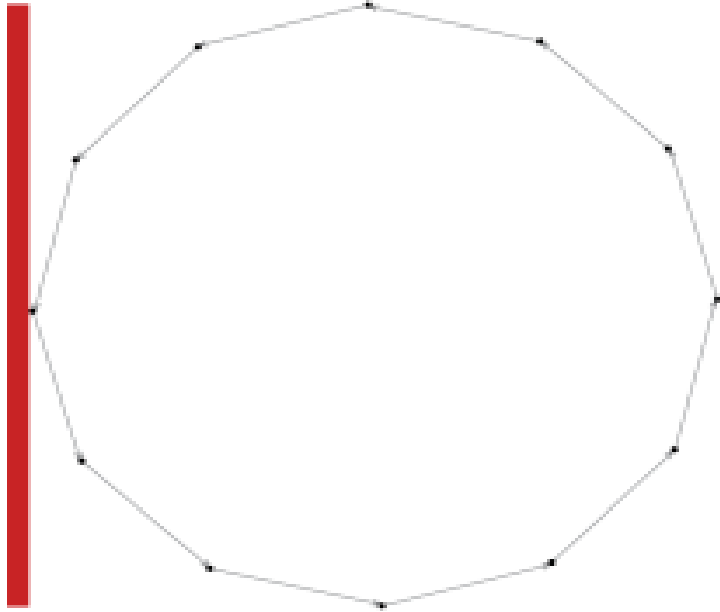
Typical Sugiyama layout (dot)
- preserves tree structure



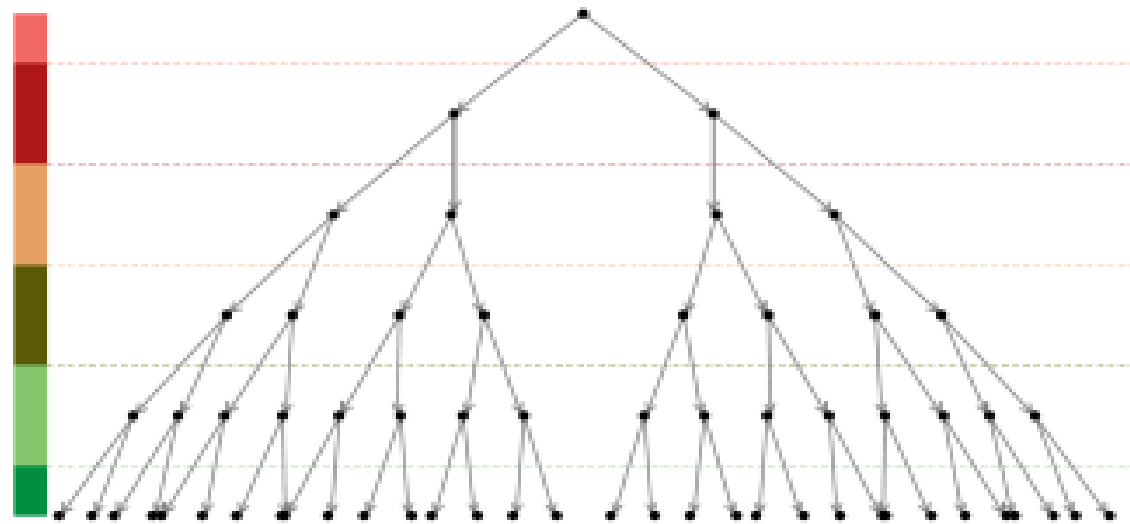
Alternative method
- preserves uniform edge lengths



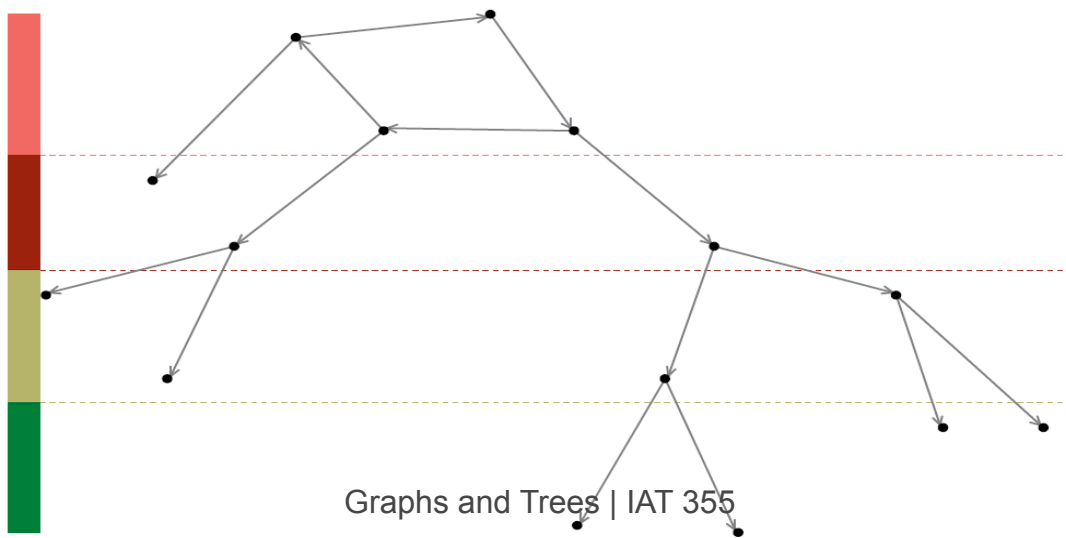
slide borrowed from Tim Dwyer



(a) 1 band

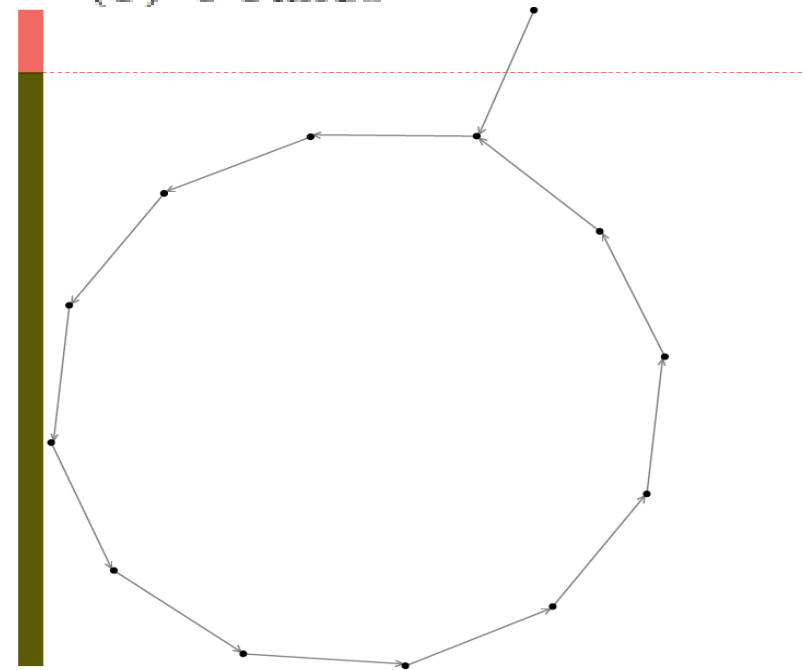


(b) 6 bands



Graphs and Trees | IAT 355

(c) 4 bands

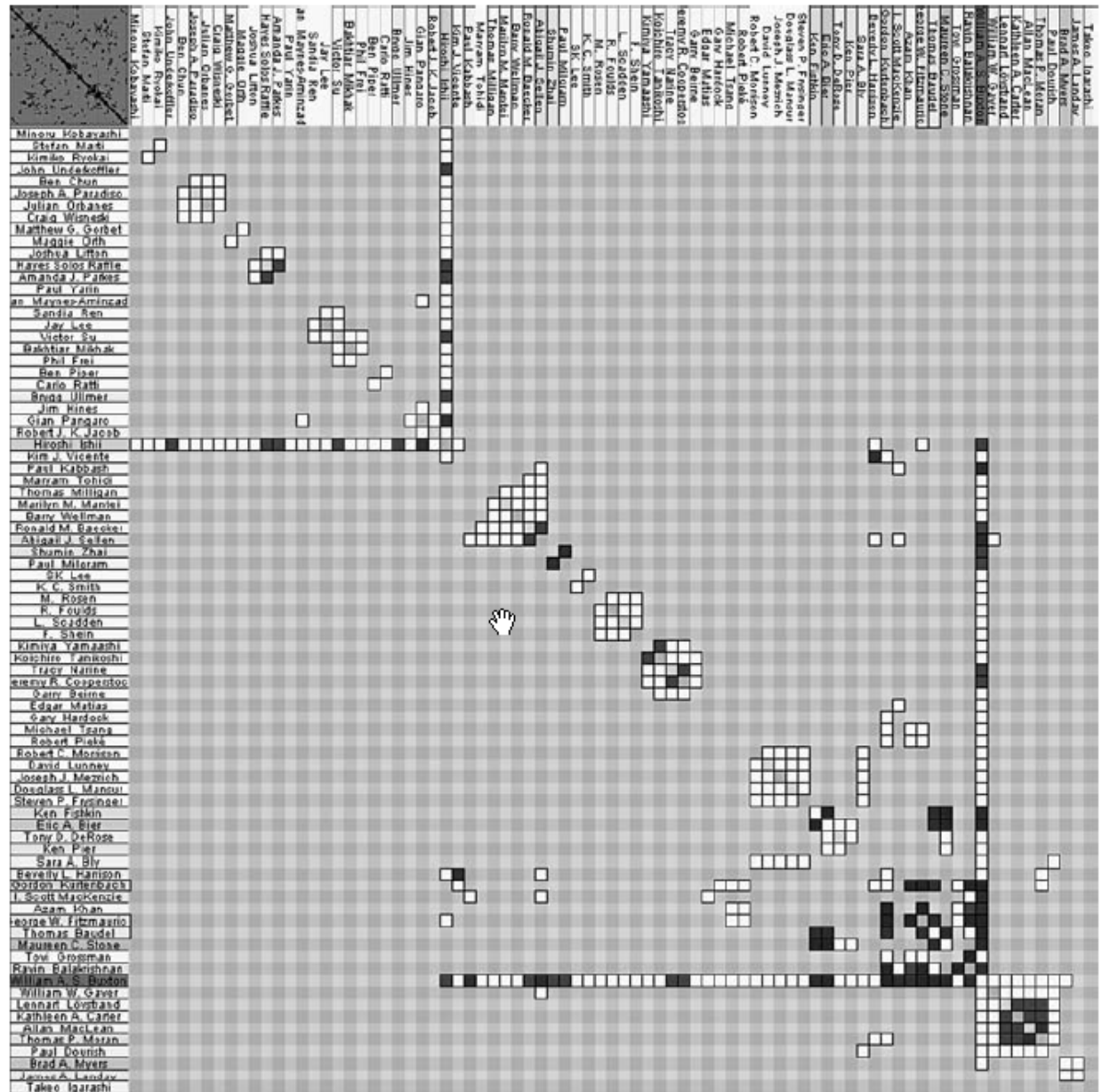


(d) 2 bands

Adjacency Matrices

- So far, only looked at node-link diagrams
- Often doesn't scale well due to edge-crossings, occlusion, etc. --> hard to read
- One solution: adjacency matrix
 - show graph as table
 - nodes as rows/columns
 - edges as table cells

Matrices

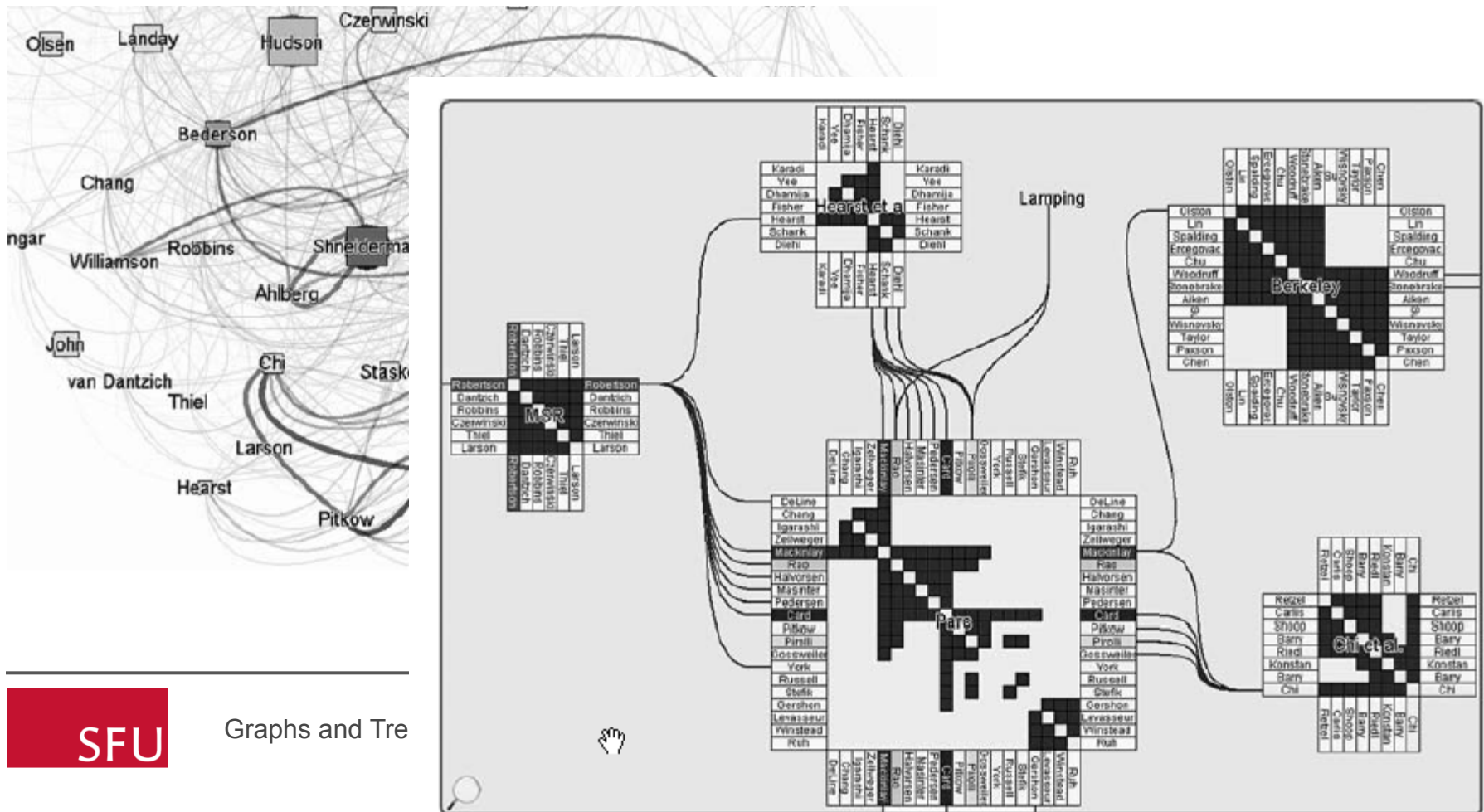


<http://www.informaworld.com/s>

Graphs and Trees | 171 | 333

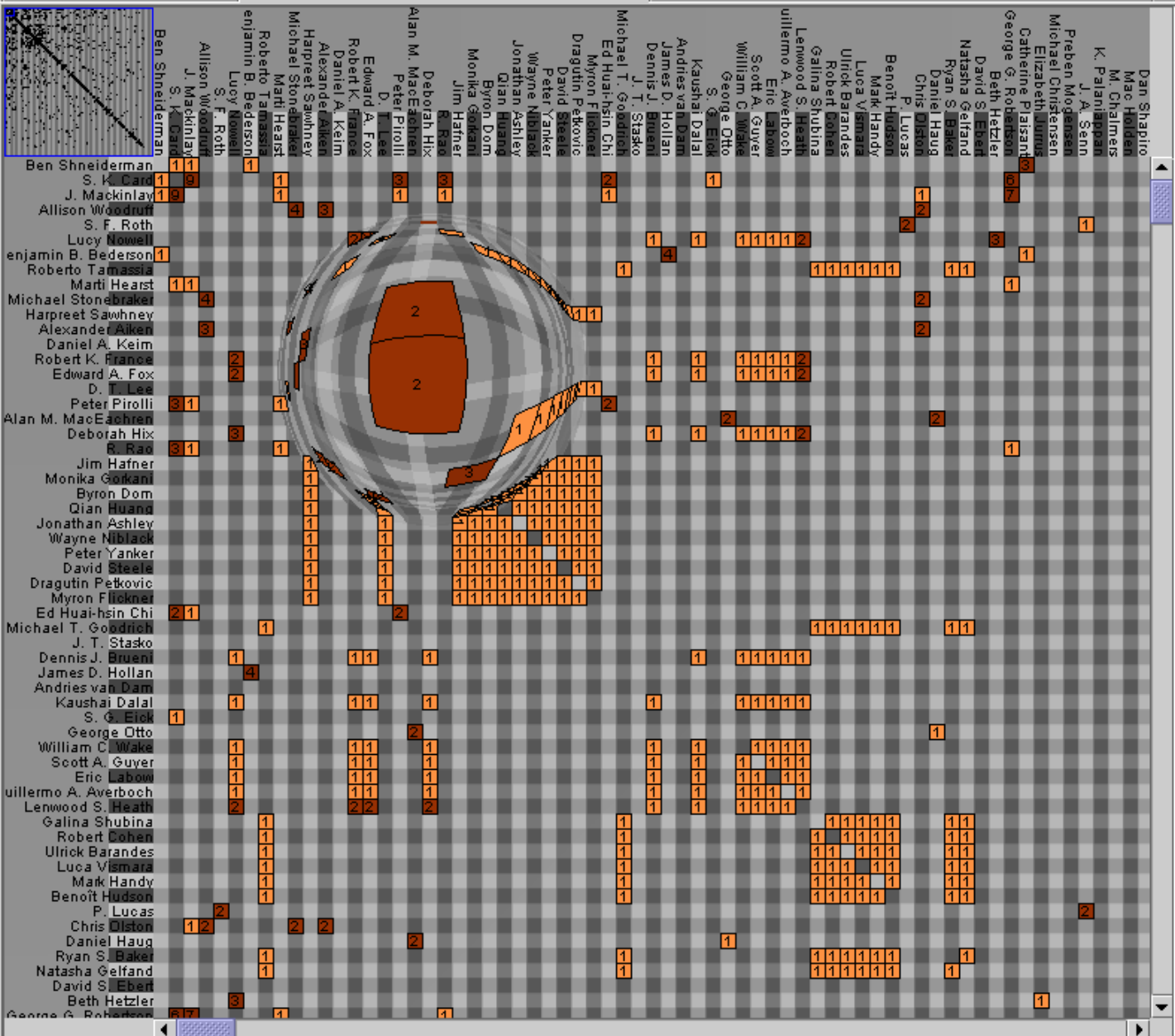
Matrices with Submatrices

- <http://www.informaworld.com/smpp/content~content=a789632485~db=all>



File Help

- Treemap
- Tree Node-Link
- Table Matrix
- Table Scatter Plot
- Table Time Series
- Table Parallel Coordinates



Tree Icicle

Fisheyes Rulers

Visual Exentric

Detail Filters

Label

Column

count (Integer)

Label all items

Size

Column

(None)

Default

0=Fit Labels 50

Background

Color gradient

Sort

Column

(None)

Inverse Order

Label Vertex by

author (String)

Sort Row by

degree (Integer)

Inverse Sort Row

Sort Column by

degree (Integer)

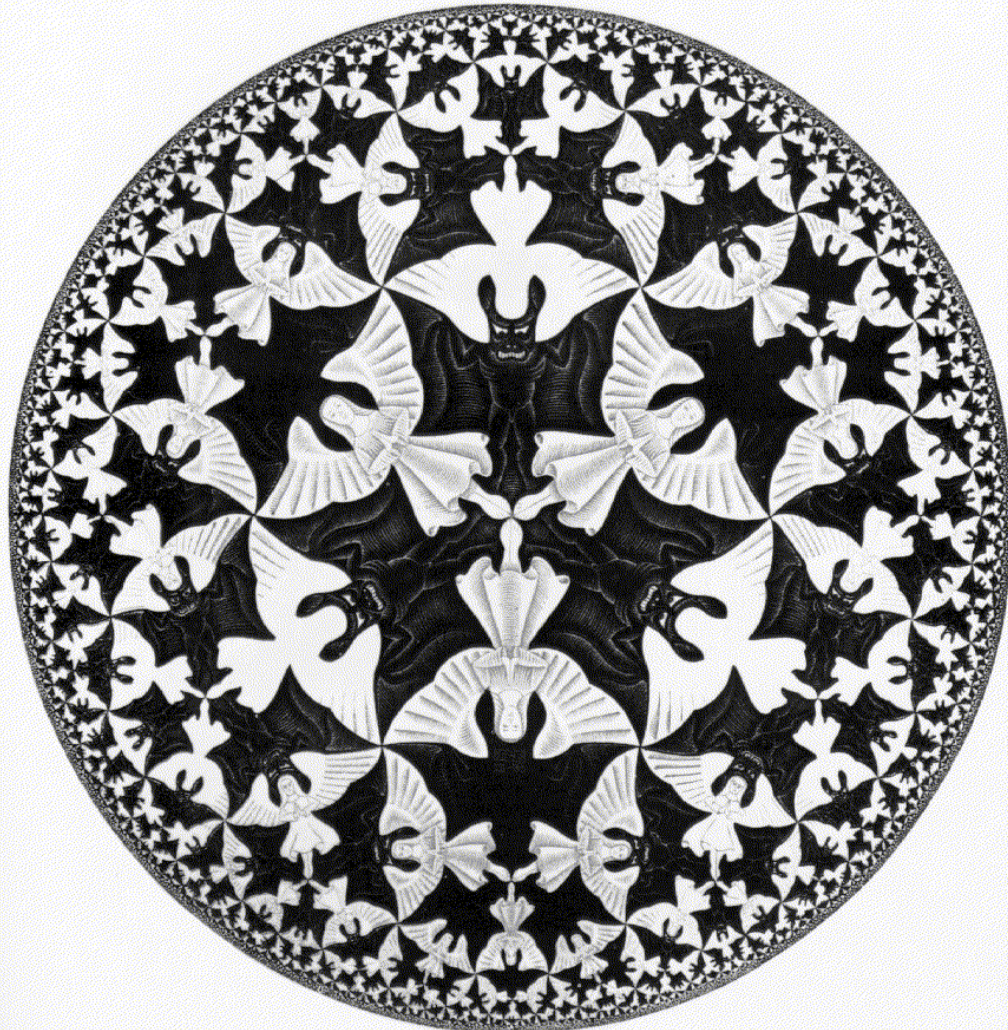
Inverse Sort Column

What about other ways to optimise screen space?

We can distort the space interactively !



Hyperbolic Browser: Inspiration



Using Distortion and Focus + Context

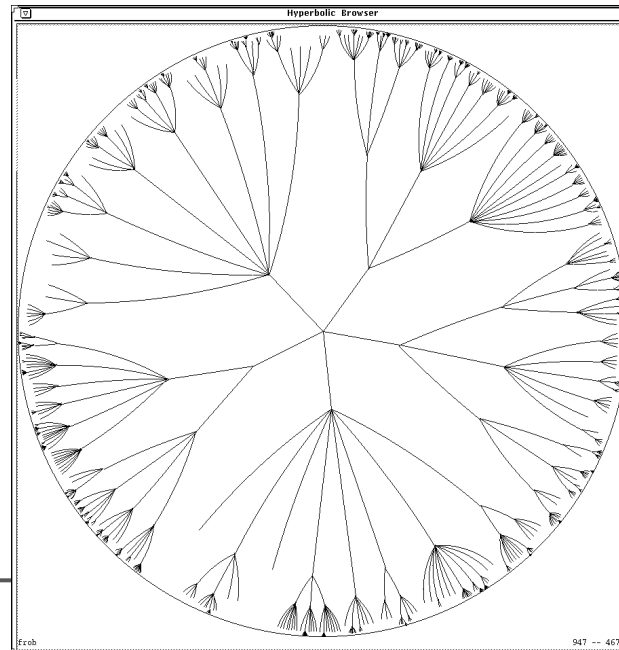
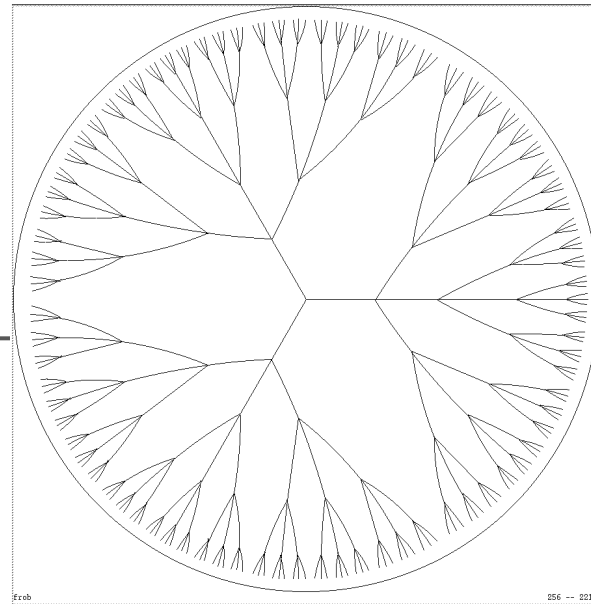
- The Hyperbolic Tree Browser

The Hyperbolic Browser: A Focus + Context Technique for Visualizing Large Hierarchies, Lamping & Rao, CHI 1996.

- <http://www.inxight.com/products/sdks/st/>
- Uses non-Euclidean geometry as basis of focus + context technique
 - The hyperbolic browser is a projection into a Euclidean space – a circle
 - The circumference of a circle increases at a linearly with radius ($2\pi r$)
 - The circumference of a circle in hyperbolic space increases exponentially
- Exponential growth in space available with linear growth of radius
 - Makes tree layout easy
- Size of objects decreases with growth of radius
- Reduces expense of drawing trees when cut-off at one pixel

Initial Layout

- Root mapped at center
- Multiple generations of children mapped out towards edge of circle
- Drawing of nodes cuts off when less than one pixel



User orientation on refocus

- Problem
 - Hyperbolic Geometry can allow disorienting rotations of objects when refocusing
- Solution one:
 - Preserve initial angular orientation of parent to child nodes
- Solution two:
 - Preserve left to right orientation of parent to child nodes beginning with initial display
- Note: both rely on relative geometric consistency

Structurally-Independent Layout

- Ignore the graph structure.
- Base the layout on other attributes of the data
- Examples:
 - Geography
 - Time
- Benefits
 - Often very quick layout
 - Optimizes communication of particular features
- Drawbacks
 - May or may not present structure well

Fisheye techniques

- Based on camera fisheye lens model
- Original 1D – Furnas
- Distort screen space around area of interest – FOCUS
- Use interaction to expand and collapse sub-graphs – HIERARCHY

-
- Continuous Zoom (levels of hierarchy)

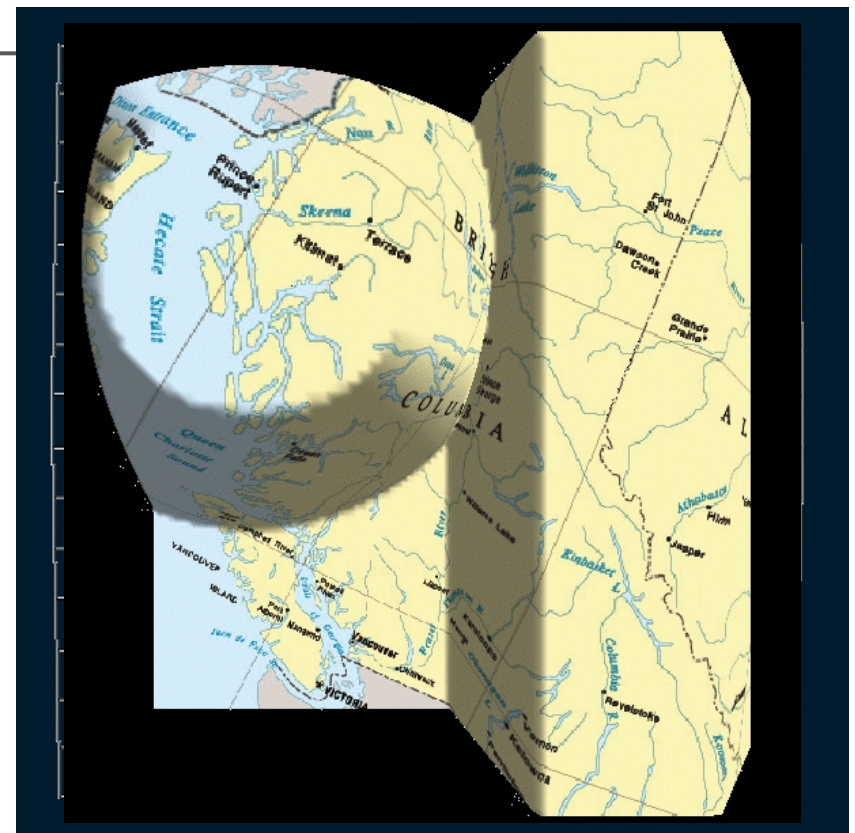
19. The Continuous Zoom: A Constrained
Fisheye Technique for Viewing and
Navigating Large Information Spaces

Lyn Bartram, Albert Ho, John Dill,
Frank Henigman
Simon Fraser University



Contact:
lyn@cs.sfu.ca

Time: 4:20



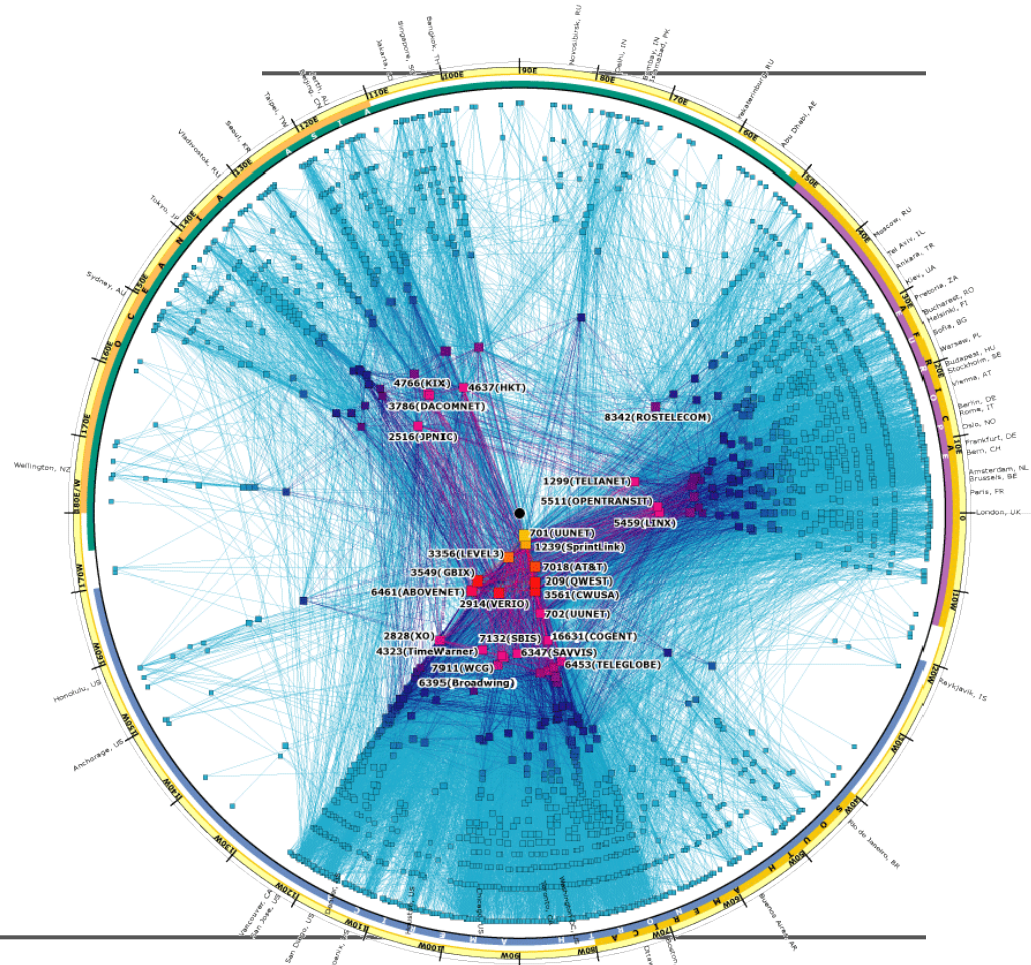
Structurally-Independent Layout

- Ignore the graph structure.
- Base the layout on other attributes of the data
- Examples:
 - Geography
 - Time
- Benefits
 - Often very quick layout
 - Optimizes communication of particular features
- Drawbacks
 - May or may not present structure well

Structurally Independent Layout

- The “Skitter” Layout
 - Internet Connectivity
- Angle = Longitude
 - geography
- Radius = Degree
 - # of connections

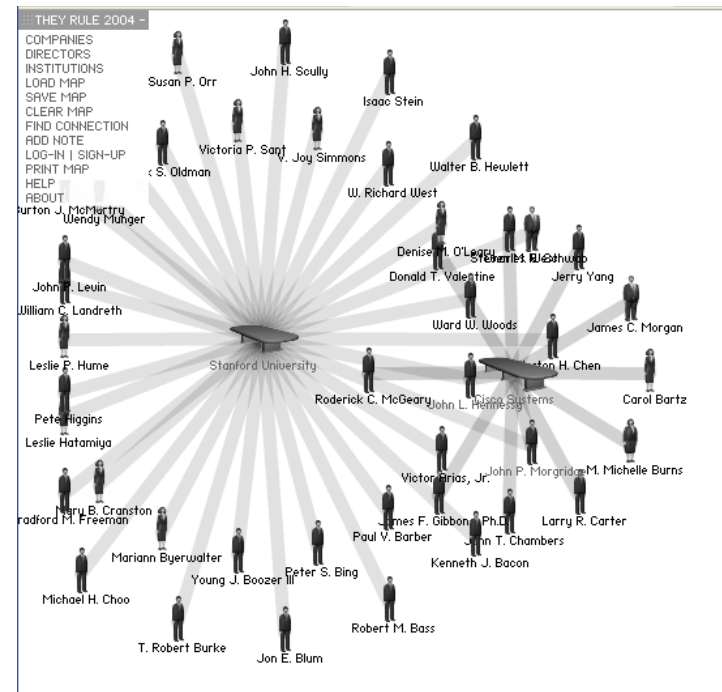
Skitter, www.caida.org



http://www.caida.org/research/topology/as_core_network/2007/images/ascore-simple.2007_big.png

Progressive Disclosure

- Only show subsets that are currently selected
- <http://www.theyrule.net/>
- <http://kylescholz.com/projects/wordnet/wordnet2.html>

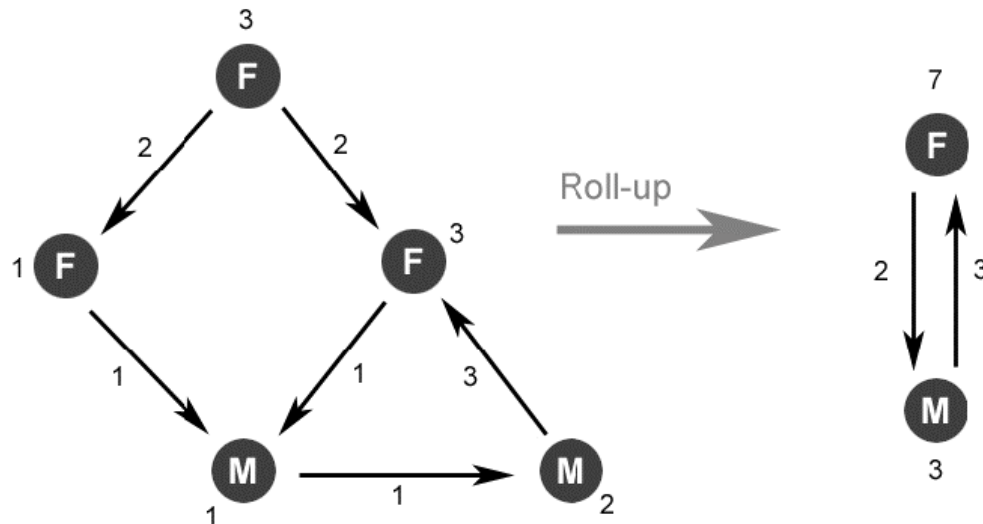


Problem: Multivariate Graphs

- What if you want to associate information with the nodes and edges?
- Typical approach: vary
 - Size of nodes
 - Color of nodes
 - Fatness of edges
 - Colors of edges
- However, it's hard to make quantitative comparisons when these retinal cues are spread throughout the graph.

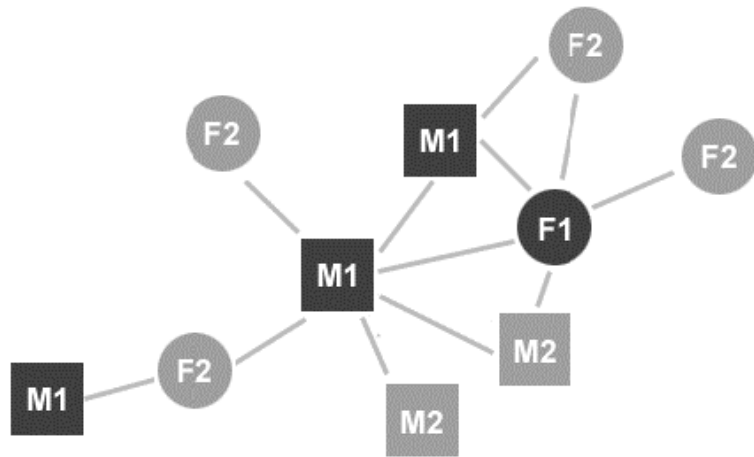
Solution: Wattenberg's Pivot Graphs

- Use “roll-up” idea from OLAP to compress and re-express graph data.
 - Aggregate all nodes that have the same values on each of those dimensions, and aggregate edges accordingly.
- In graph below,
 - F = Female, M = Male, Numbers mean counts

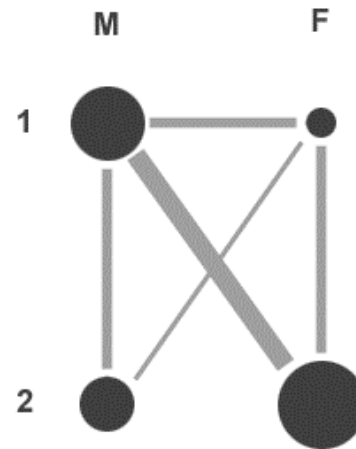


Multidimensional Pivot Graphs

- What is added, and what is lost, from this transformation?



Node and Link Diagram



PivotGraph Roll-up

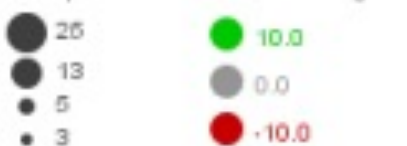
X-Axis:
 Gender

Y-Axis:
 Location

Flip X/Y

Clear

People In vs. Out Degree



Relationships Rel. / Person



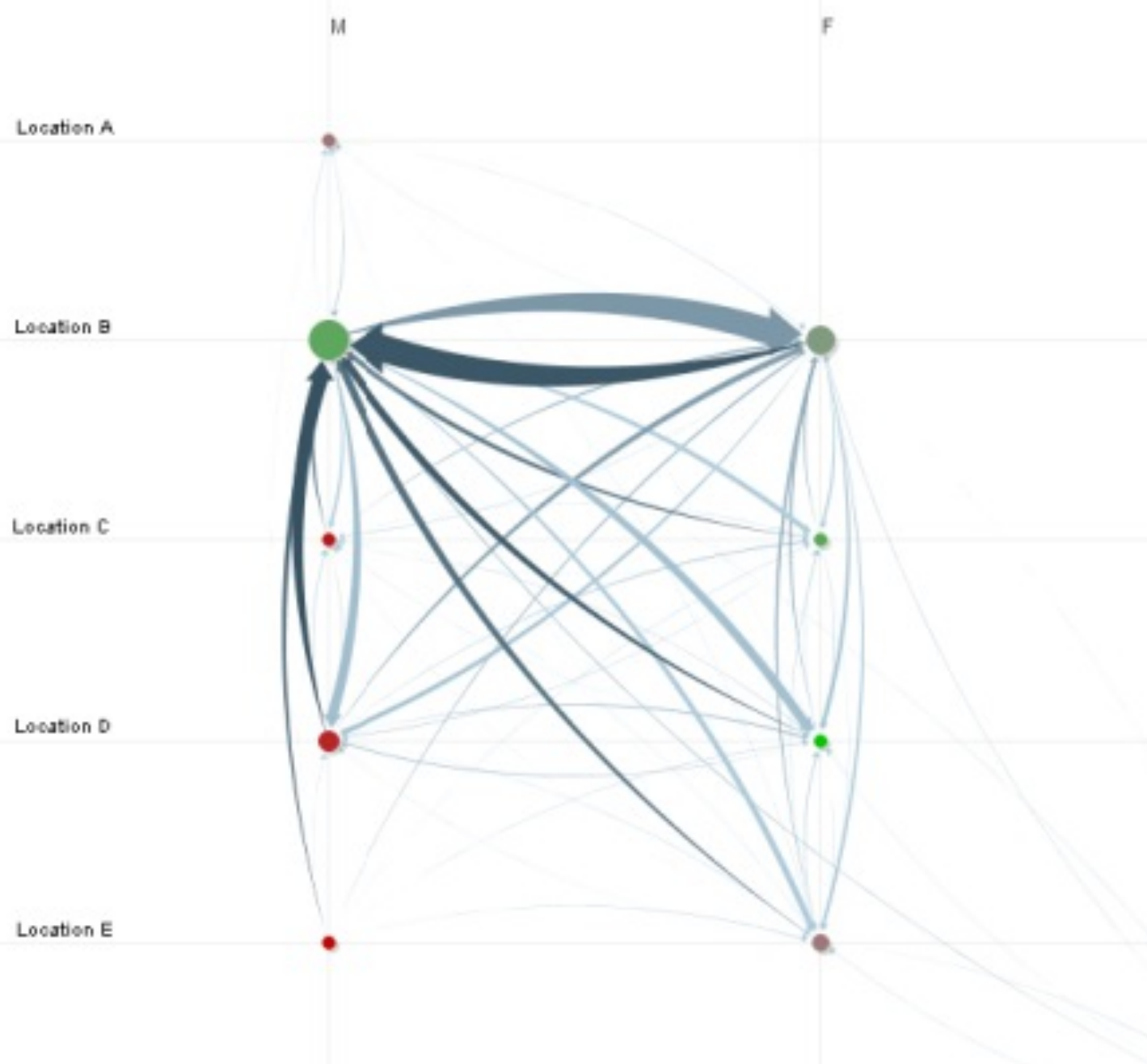
Select:

Gender
 (All Values)

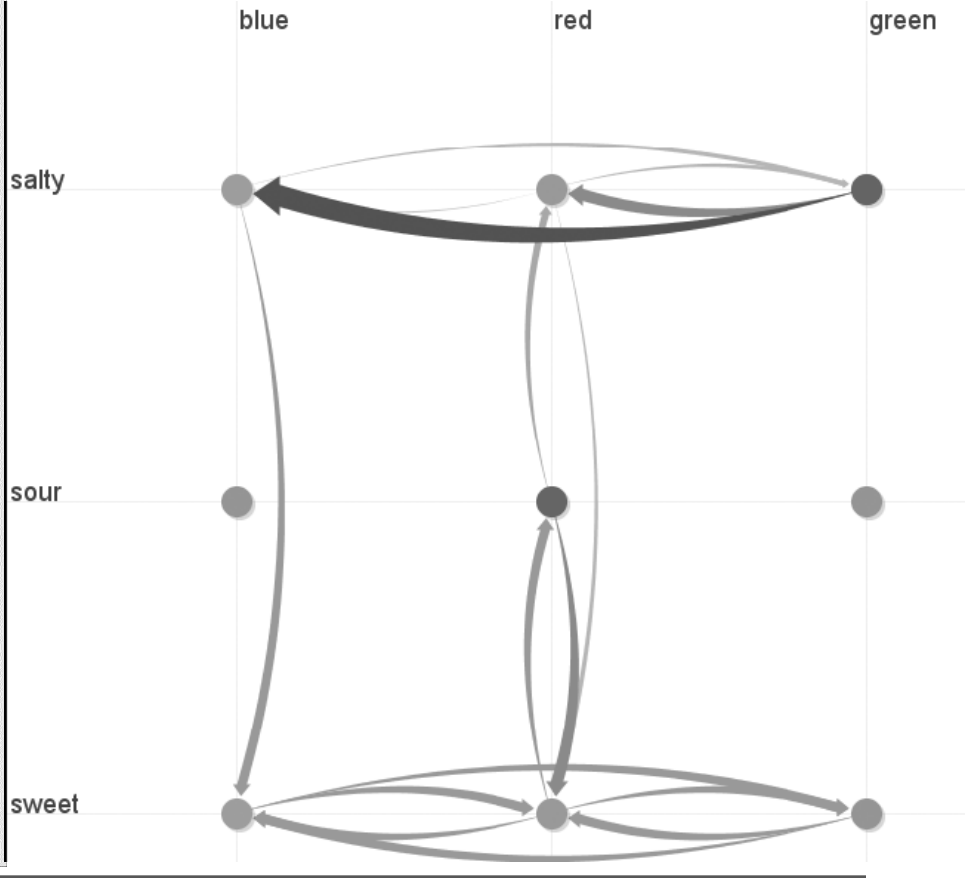
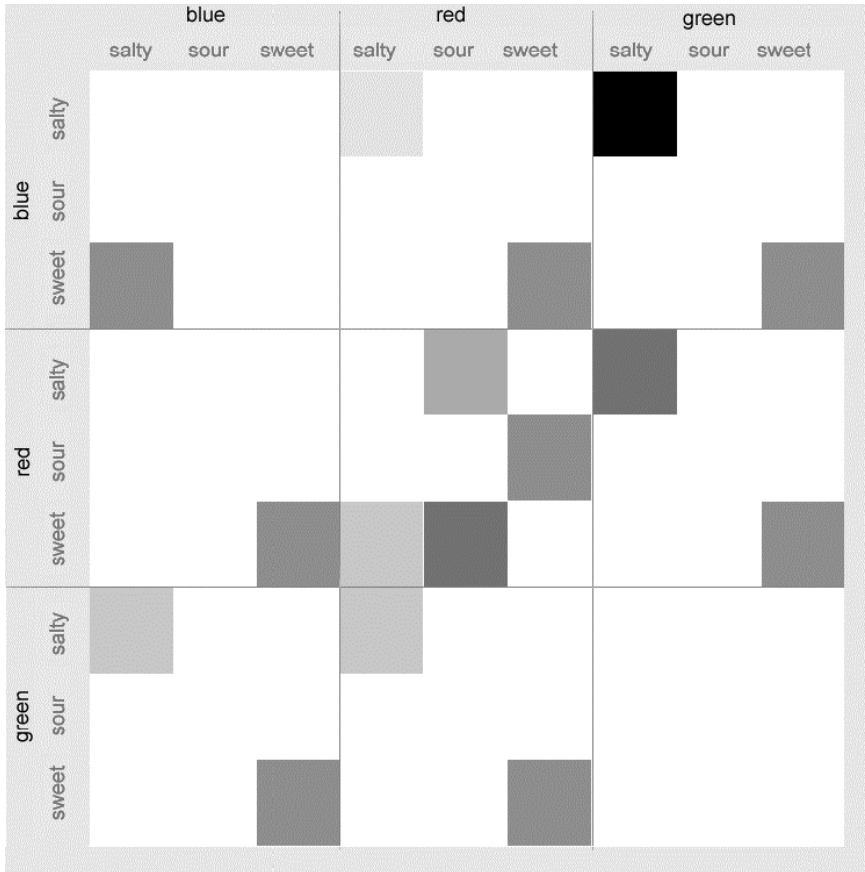
Legacy
 (All Values)

Department
 (All Values)

Level



Compare 2D Pivot Graph with 2D Matrix



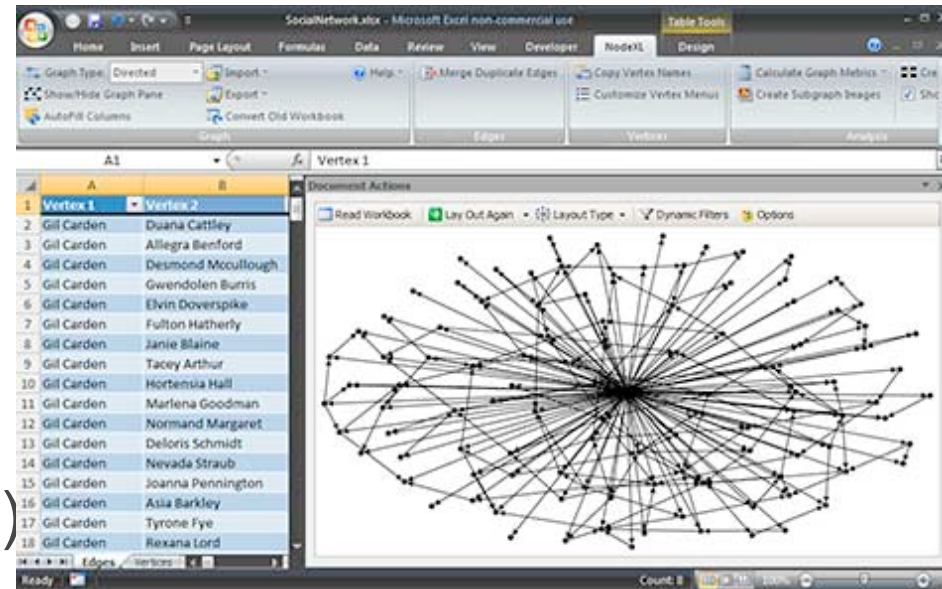
Issues with Pivot Graphs

- Disconnected components may become connected
- Acyclic graphs may obtain cycles

New toolkits!

Networks for excel by Marc Smith et al. at Microsoft research

- Used to be called .Netmap
- Now called NodeXL
- Requires windows-specific software
- (Search on “excel NodeXL”)
- Chart Tamer for Excel
- Stephen Few et al.



.NetMap: Edges Worksheet

The screenshot shows the Microsoft Excel interface with the 'Table Tools' ribbon active. The ribbon includes tabs for 'Home', 'Insert', 'Page Layout', 'Formulas', 'Data', 'Review', 'View', 'Developer', 'Add-Ins', and 'Design'. The 'Table Tools' ribbon is divided into 'Menu Commands', 'Toolbar Commands', 'Custom Toolbars', and 'C#UNG 2008'. The 'C#UNG 2008' section contains a 'Show Graph' checkbox which is checked. Below the ribbon, the formula bar shows 'F3' and 'Yes'. The worksheet contains a table with the following data:

	A	B	C	D	E	F	G	H	I
1	Vertex 1	Vertex 2	Color	Width	Opacity	Visible?			
2	16899266	14258203				Yes			Edges Worksheet

.NetMap: Vertices Worksheet

	A	B	C	D	E	F	G	H	I	J	K
1	Vertex	Color	Shape	Radius	Opacity	Image ID	Primary Label	Secondary Label	Tooltip	Visible	
2	16899266		Circle					16899266	16899266		
3	15522825		Circle					15522825	15522825		
4	11305793		Circle					11305793	11305793		
5	42502628		Circle					42502628	42502628		
6	41929652		Circle					41929652	41929652		
7	41901803		Circle					41901803	41901803		
8	41000857		Circle					41000857	41000857		

Vertex Radius
Enter an optional vertex radius between 0.1 and 50.

visuSites.xlsx - Microsoft Excel non-commercial use

Table Tools

Home Insert Page Layout Formulas Data Review View Add-Ins Design

Get External Data Refresh All Edit Links Connections

Sort Filter Clear Reapply Advanced

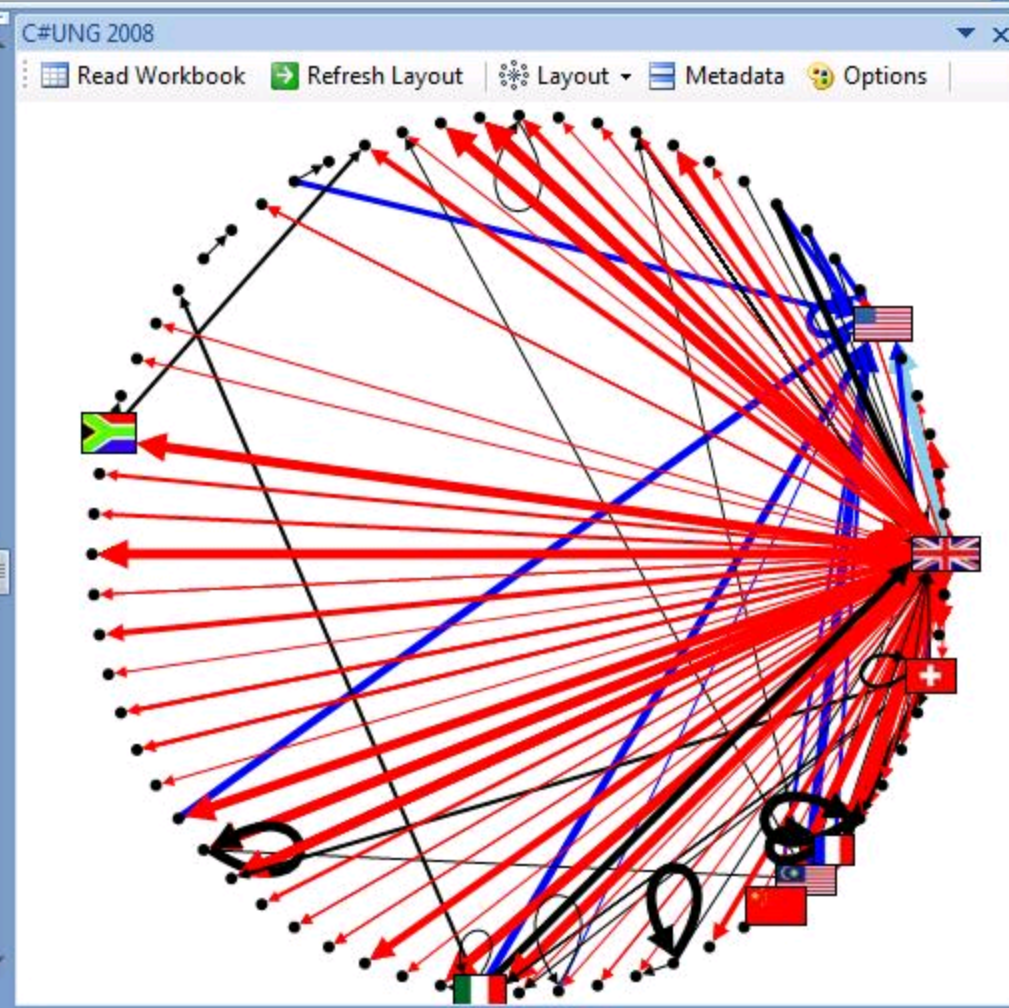
Text to Columns Remove Duplicates Data Validation Consolidate What-If Analysis

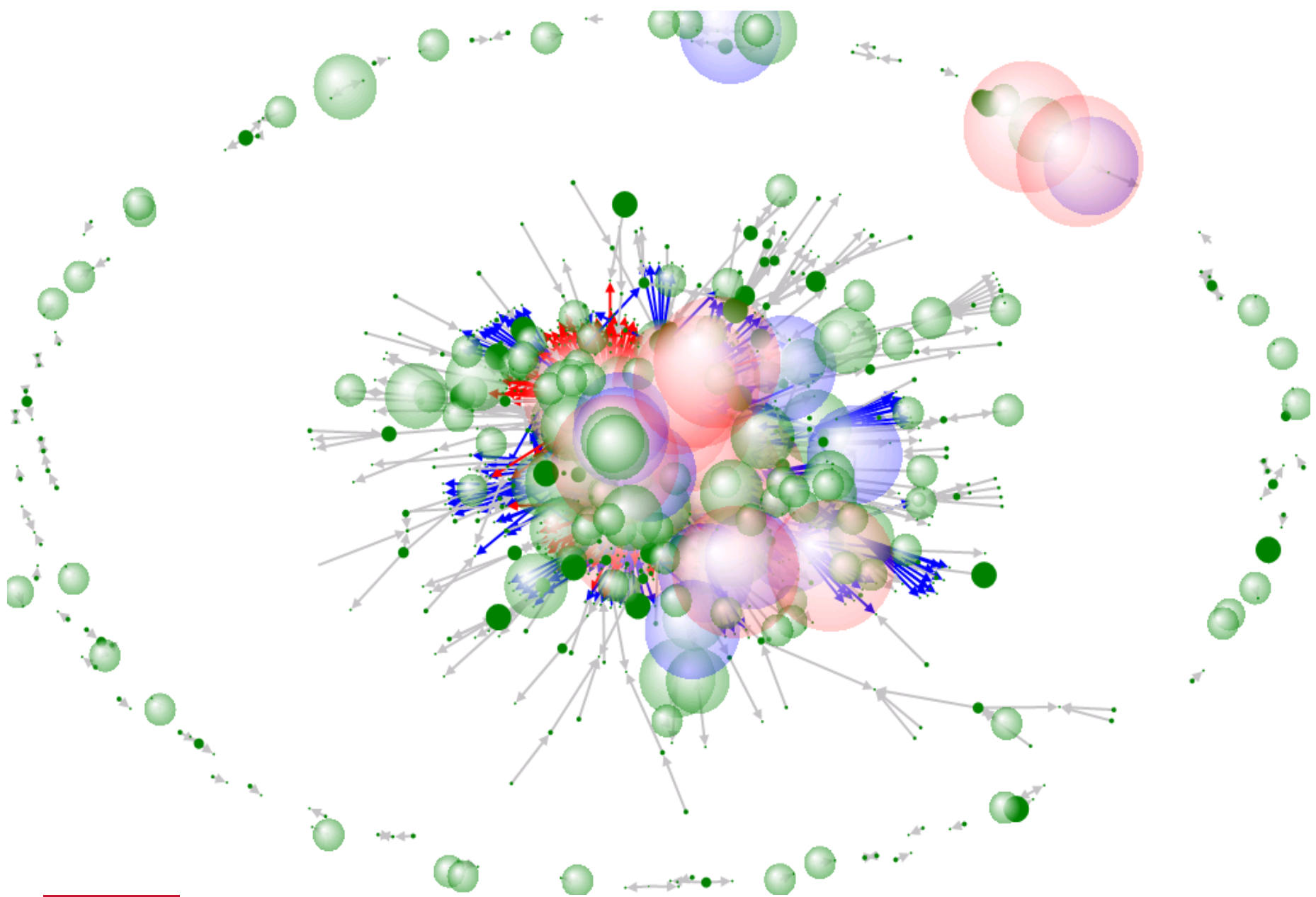
Group Ungroup Subtotal Outline

Data Analysis Analysis

	Vertex 1	Vertex 2	Color	Width	Opacity	Visible?
232	UK	US	Sky Blue	2		
233	UK	US	Sky Blue	3		
234	UK	US	Sky Blue	5		
235	UK	US	Sky Blue	5		
236	UK	US	Sky Blue	5		
237	UK	US	Sky Blue	5		
238	UK	US	Sky Blue	5		
239	UK	US	Sky Blue	5		
240	UK	US	Sky Blue	5		
241	UK	US	Sky Blue	5		
242	UK	US	Sky Blue	2		
243	UK	US	Sky Blue	5		
244	UK	US	Sky Blue	5		
245	UK	US	Sky Blue	5		
246	UK	US	Sky Blue	5		
247	UK	US	Sky Blue	1		
248	US	US	Blue	4		
249	Zambia	US	Blue	3		
250	UK	Vatican	Red	1		
251	UK	Venezuela	Red	1		
252	UK	Venezuela	Red	1		
253	UK	Venezuela	Red	5		
254	Malaysia	Vietnam		1		

Edge Opacity
 Enter an optional edge opacity between 0 (transparent) and 10 (opaque).





Data

- Overview
- Rank Nodes
- Rank Edges
- Plot Nodes
- Find Communities
- Edge Types
- Details
- Settings

Community

Newman's Community

Disable

ID	Number of Nodes
0	28
1	13
2	27
3	14
4	22
5	34
6	21

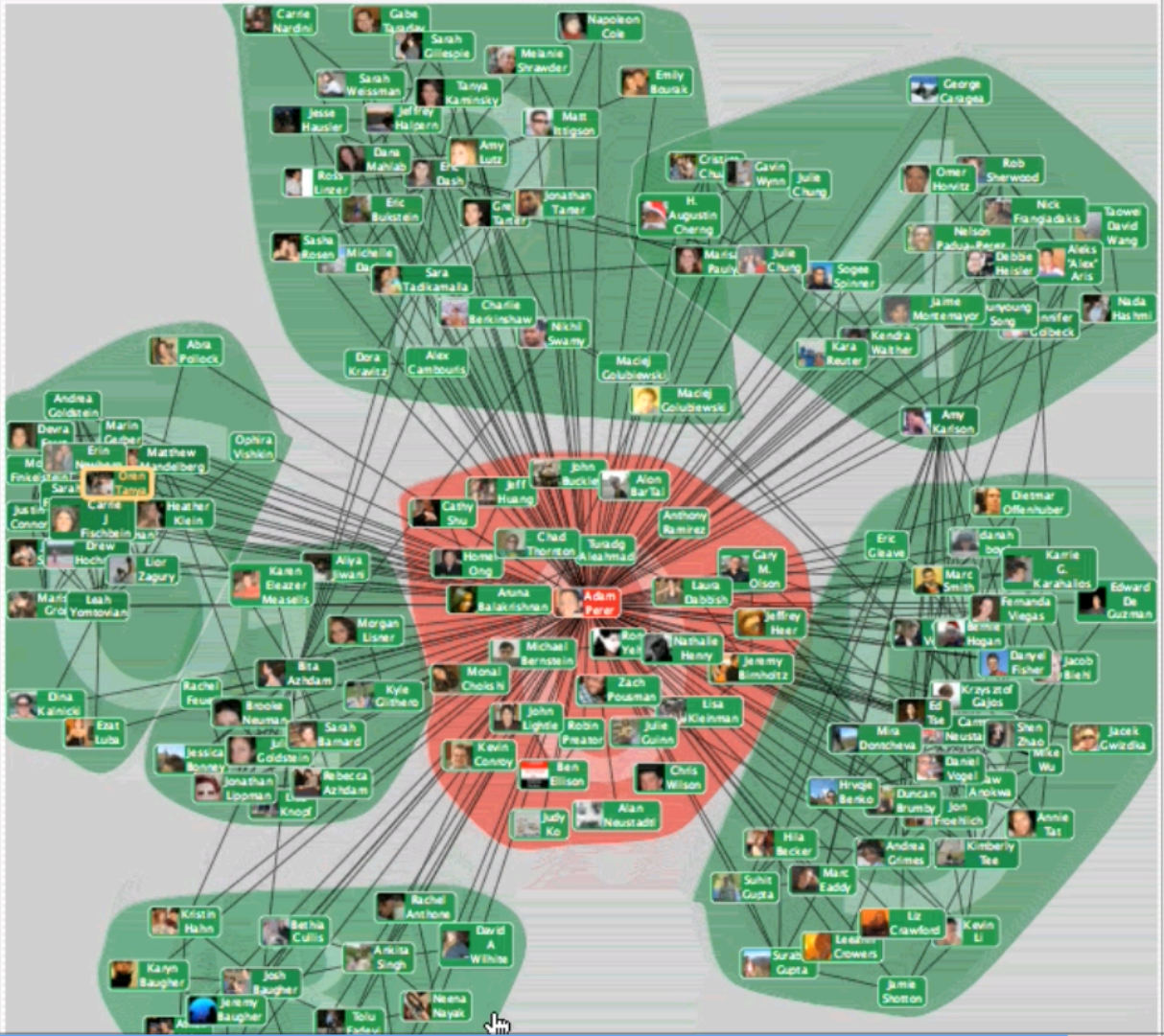
Unselect Table

Show Subgraph ...

Compress Graph

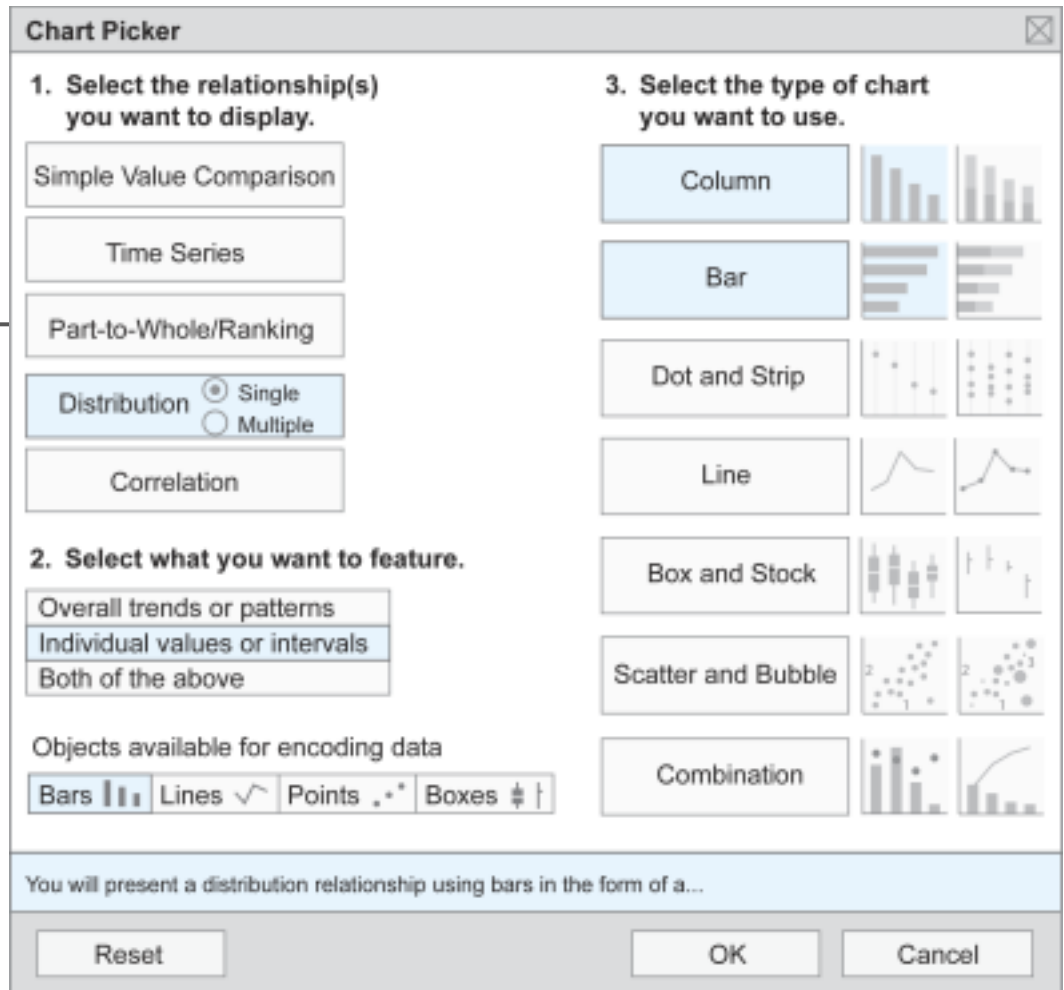
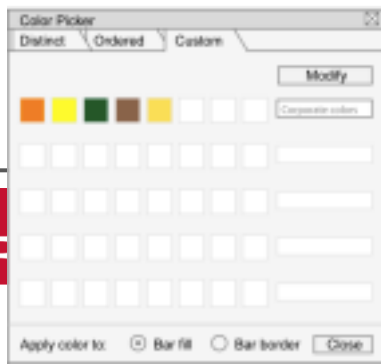
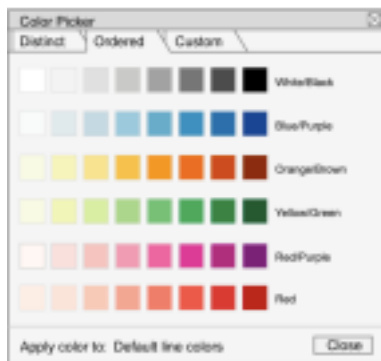
Network Visualization

Layout: Font Size: Edge Size: Label Size:



Navigation controls including back, forward, and play buttons, along with a volume slider.

Chart Tamer



- Stephen Few + XL³
- Function-based