IAT 814
Knowledge Visualization

Networks, Graphs and Trees

Lyn Bartram

SFU

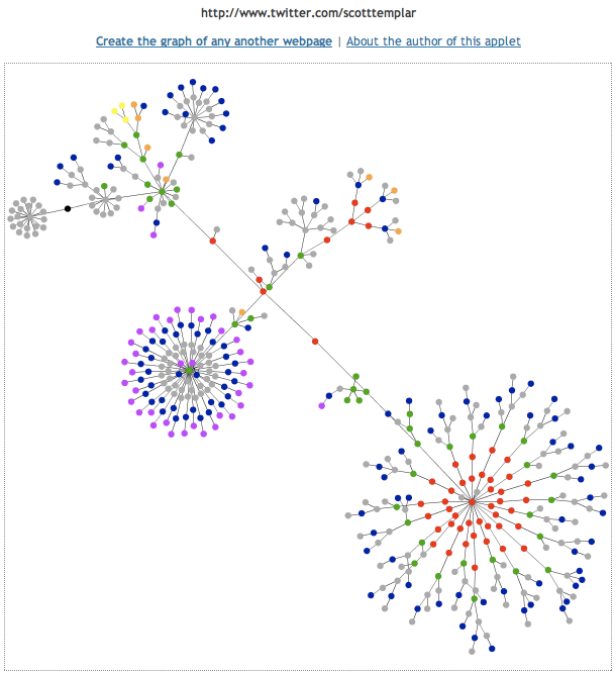
SCHOOL OF INTERACTIVE ARTS + TECHNOLOGY

Administrivia

- Assignment 2 presentations next week
- 5 each day
  - 15 minutes each , including time for discussion
  - Remember your two questions !!!
- Assignment 3 out tonight (you'll like this one).

We live in a connected world

- Online Social networks: Facebook, Twitter ~ people connected online
- Information networks: WWW ~ web pages connected through hyperlinks
- Computer networks: The internet ~ computers and routers connected through wired/wireless connections
- What is a network? “any collection of objects in which some pairs of these objects are connected by links” [Easley and Kleinberg, 2011]



Visualizing Relations

Why relations? Isn't all data inherently relational?

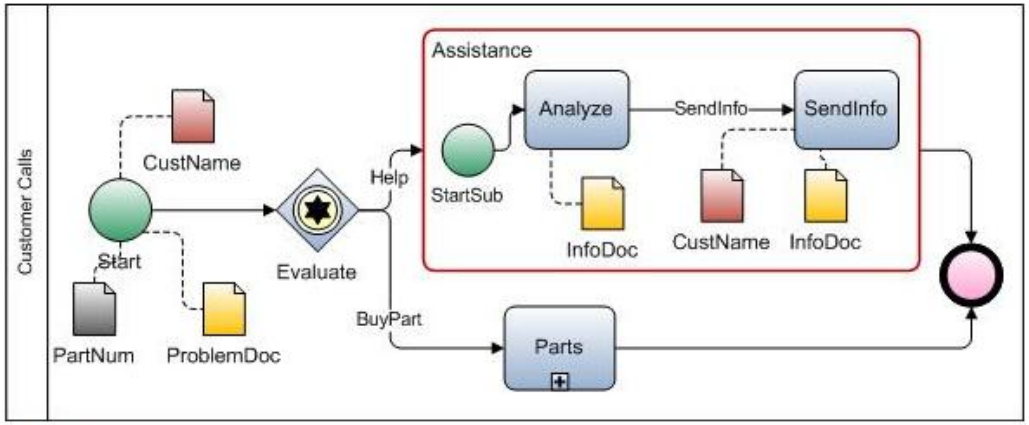
- Visualising data: seeing the patterns between the data values and attributes that emerge and associate or disassociate in some way
- Visualising relations: when how one datum relates to another is an element in itself

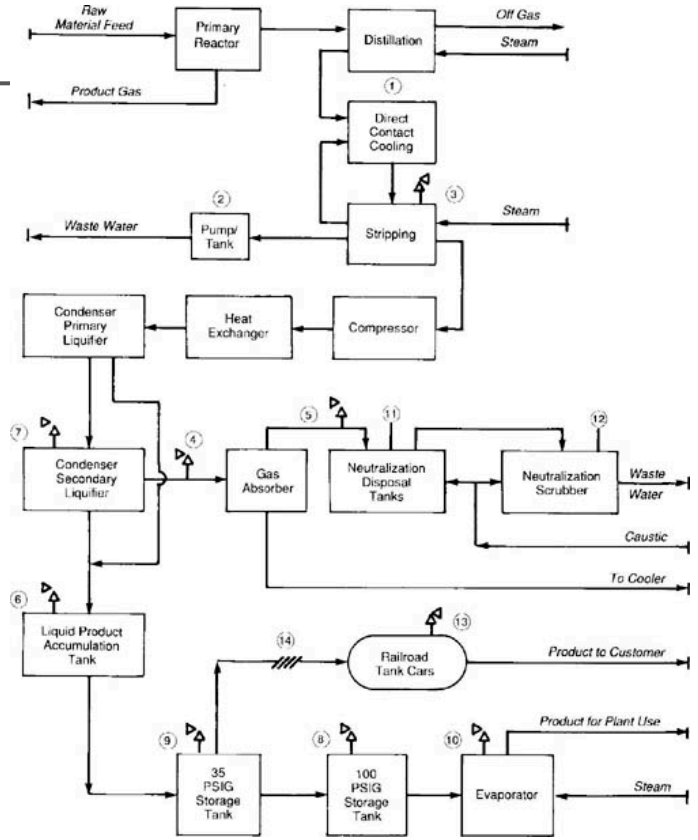
- We want to see the overall structure of the data set
- Patterns emerge from structure as well as from values/ attributes

Common Applications

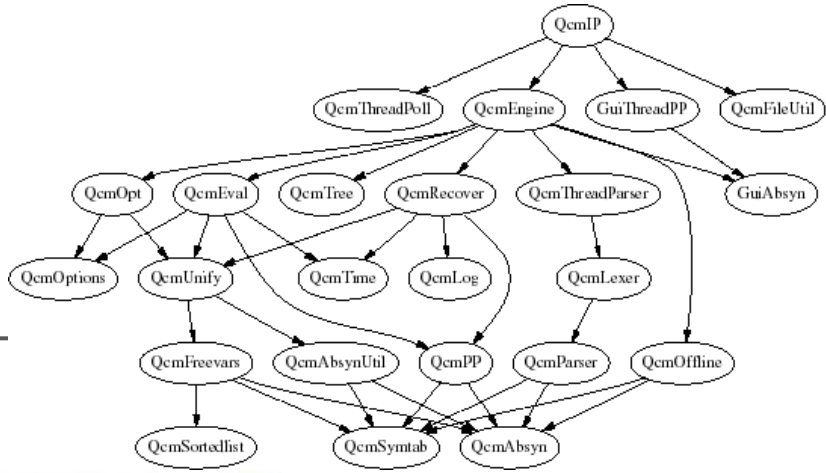
- Process Visualization (e.g., Visio)
- Dependency Graphs
- Biological Interactions (Genes, Proteins)
- Computer Networks
- Social Networks
- Concept maps
- Ontologies
- Simulation and Modeling
- Probability maps

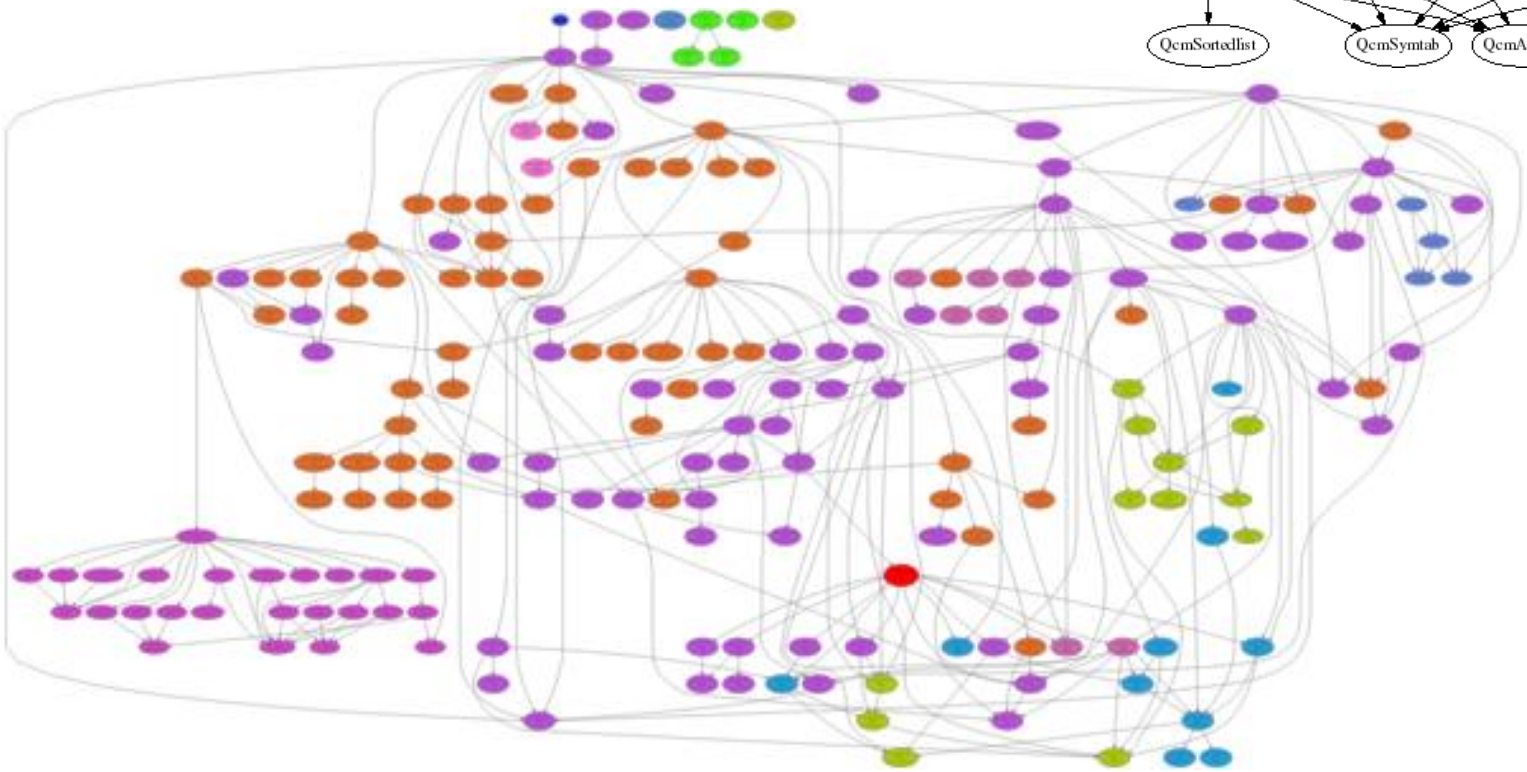
Process flow diagrams



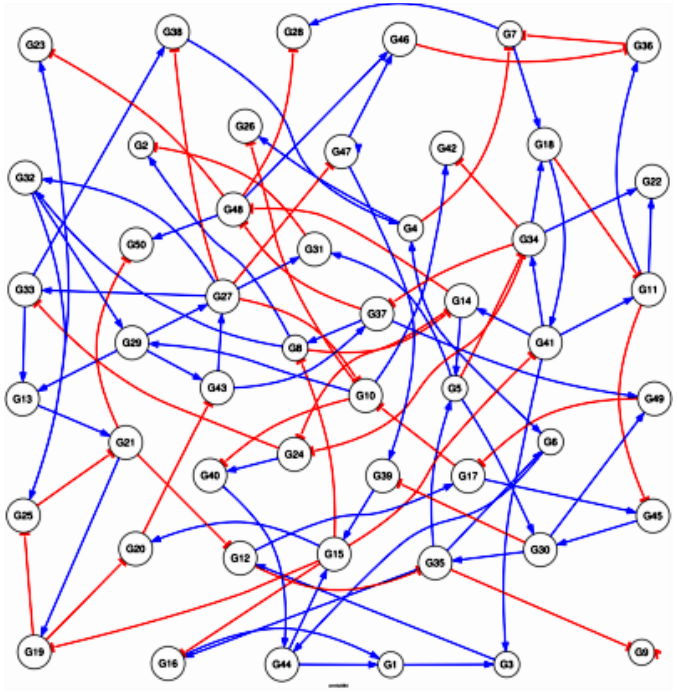


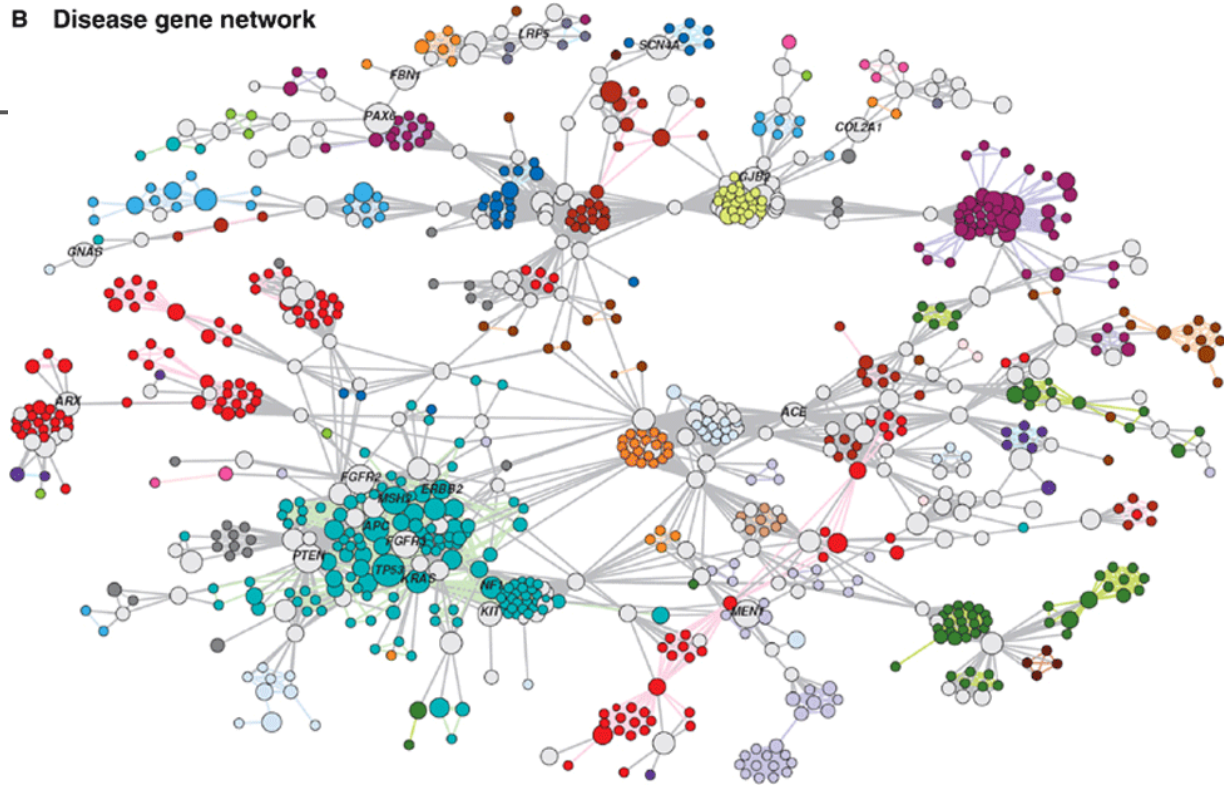
Dependency graphs



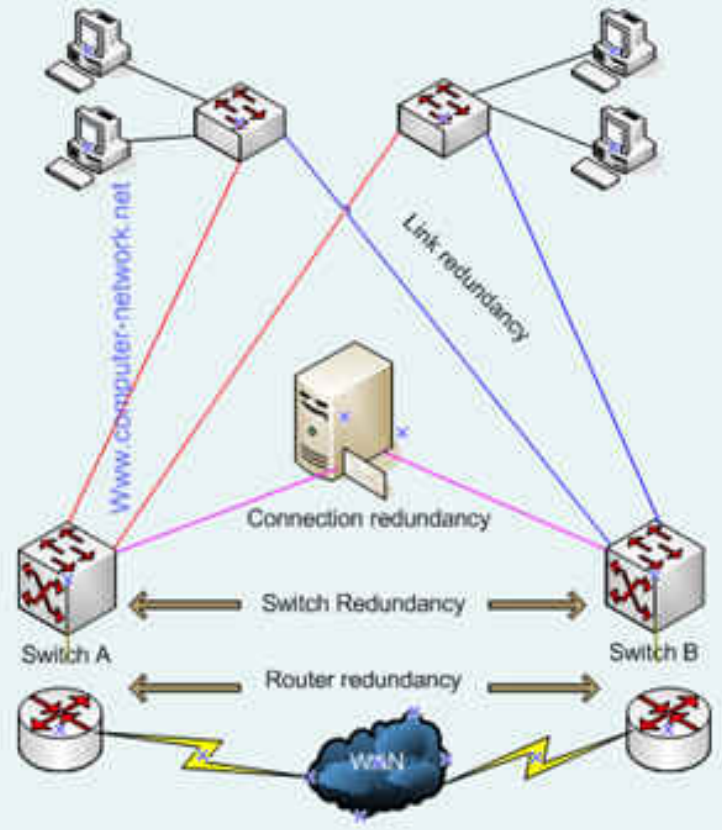


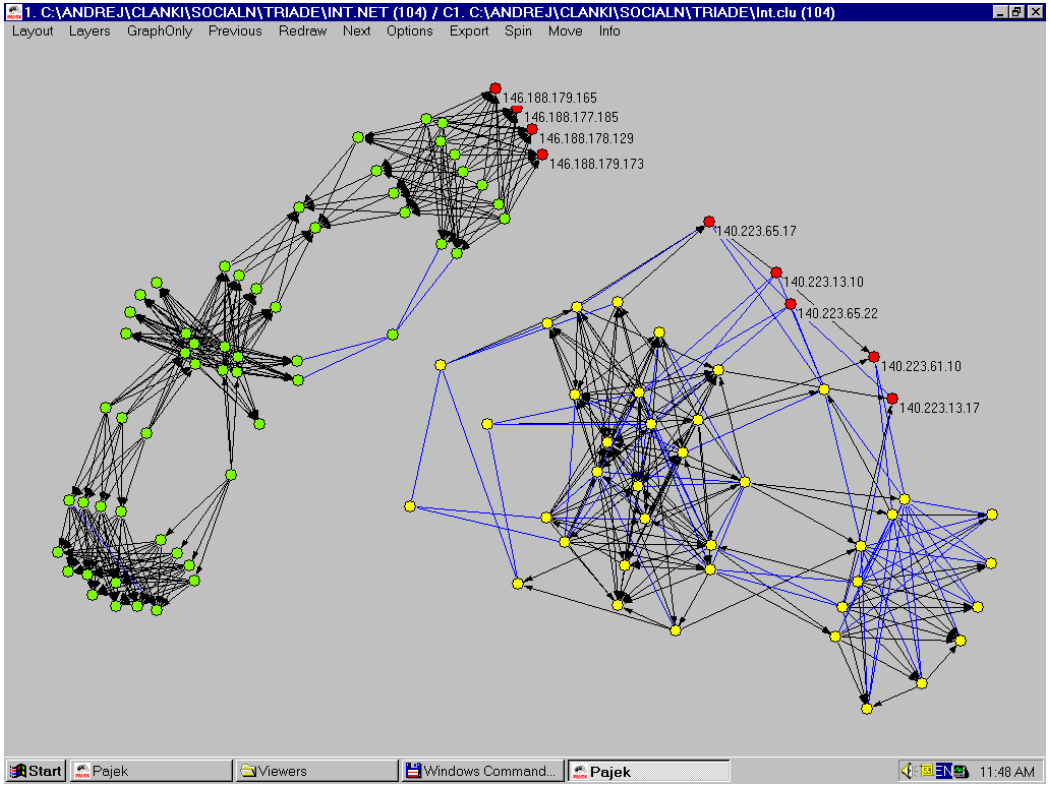
Gene networks





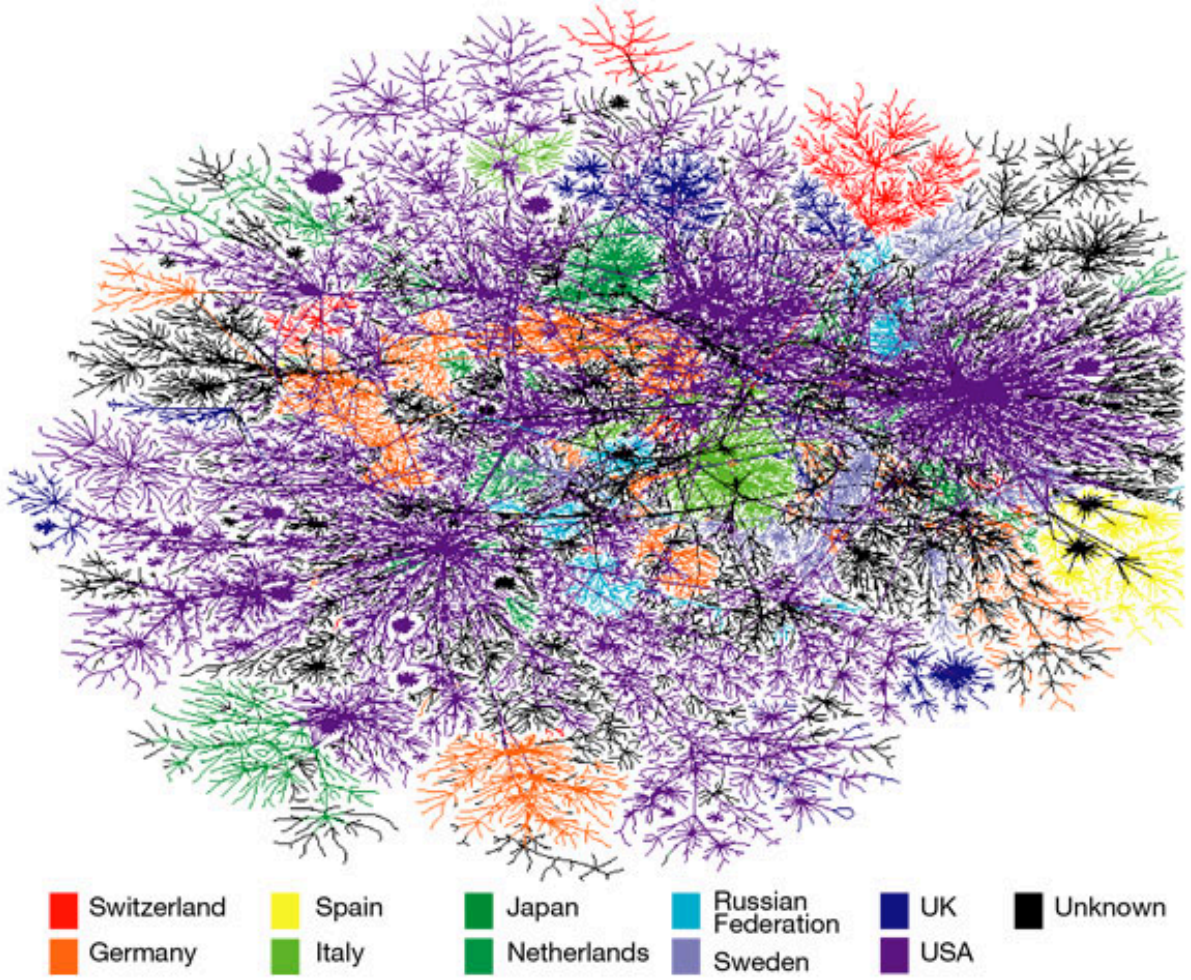
Computer networks





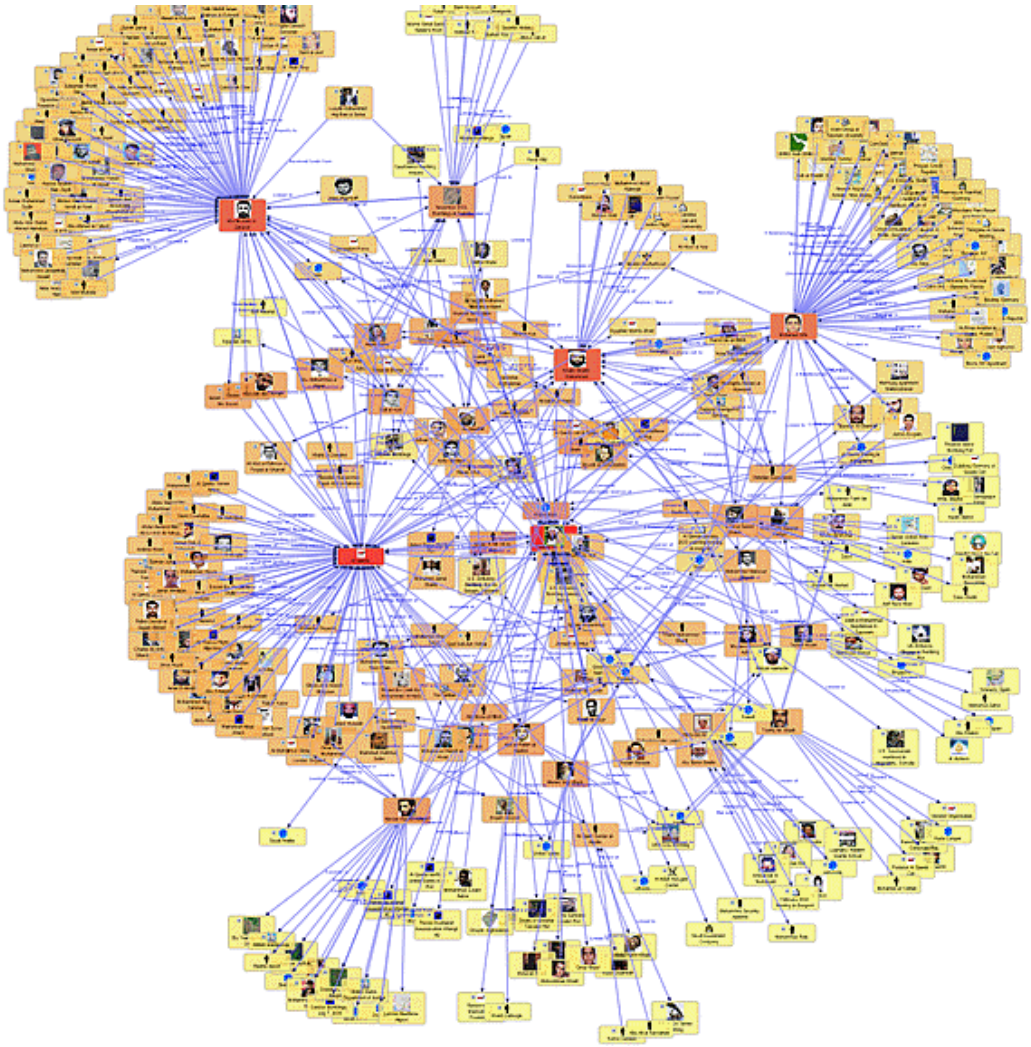
Internet

- What does the Internet look like?
  - Email paths

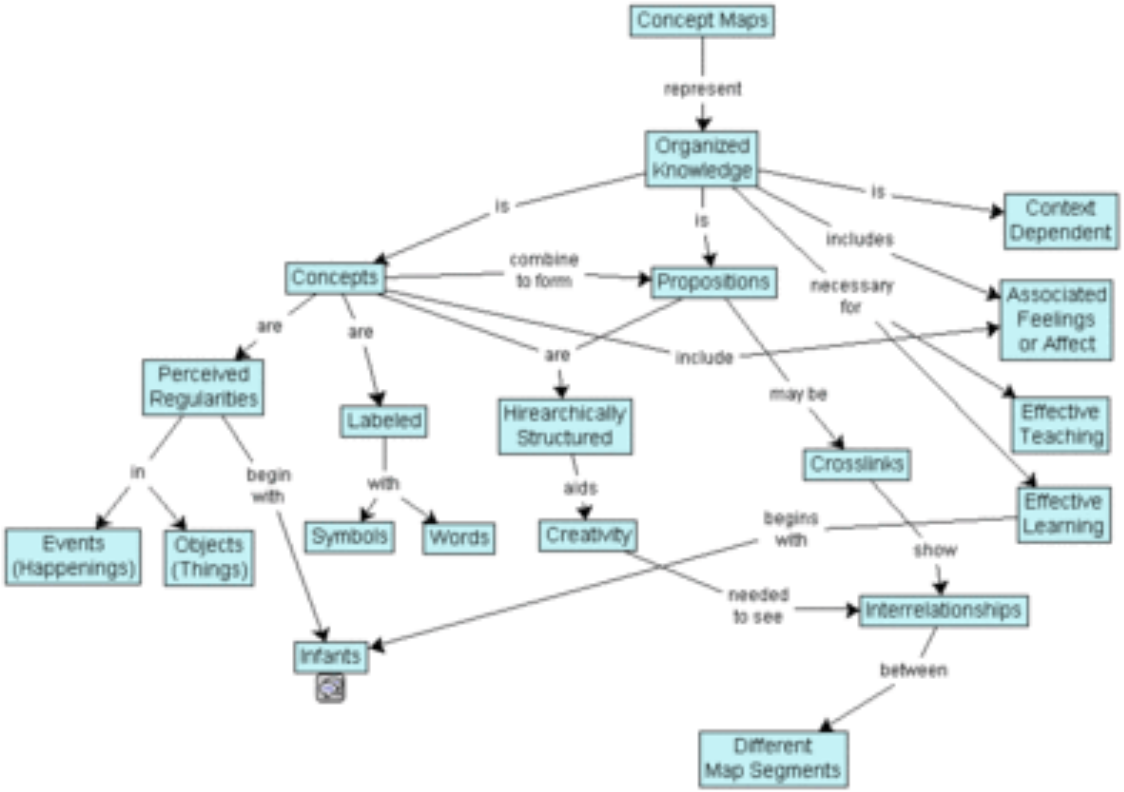


Nature, 406, 353-354(27 July 2000)

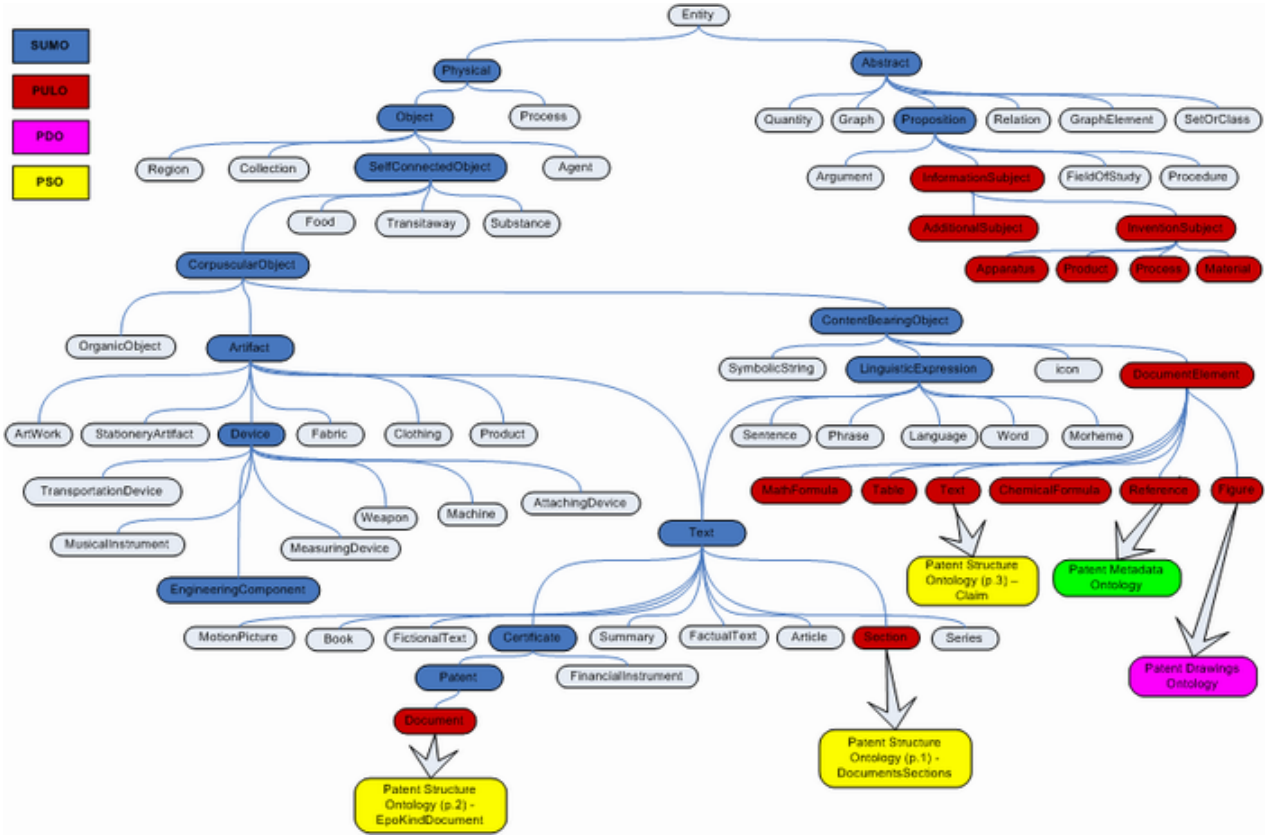
Social networks



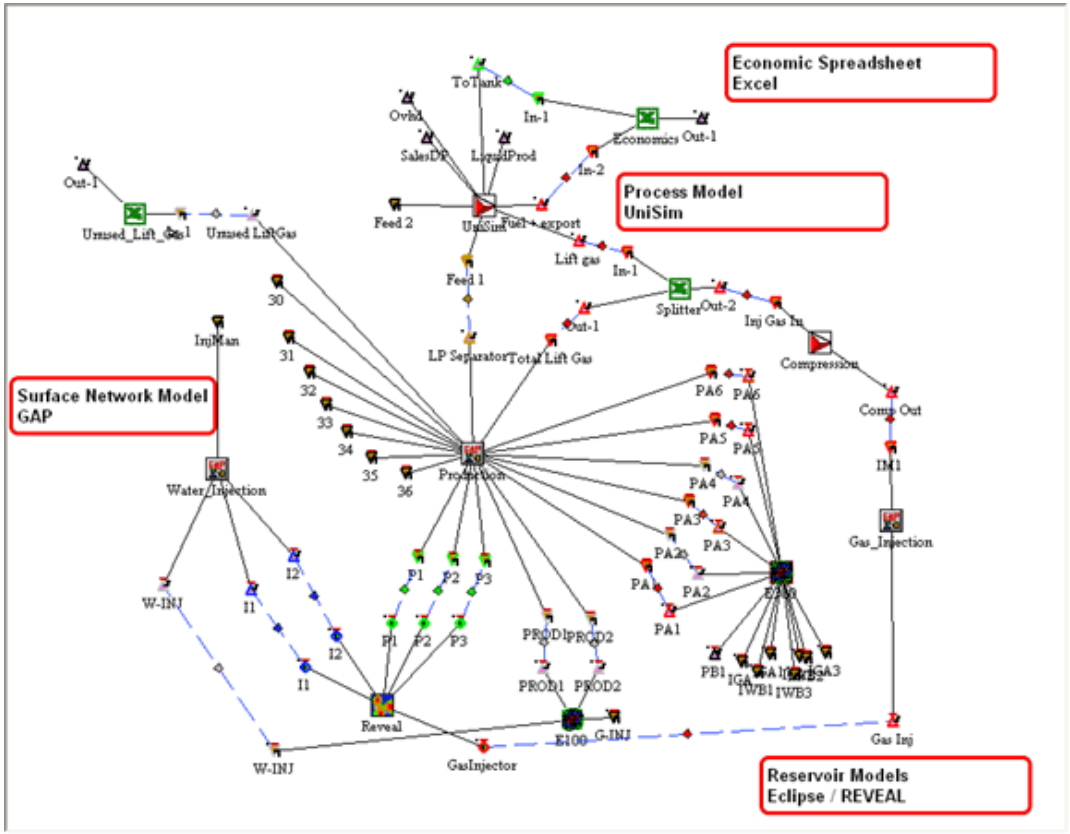
Concept maps



Ontologies

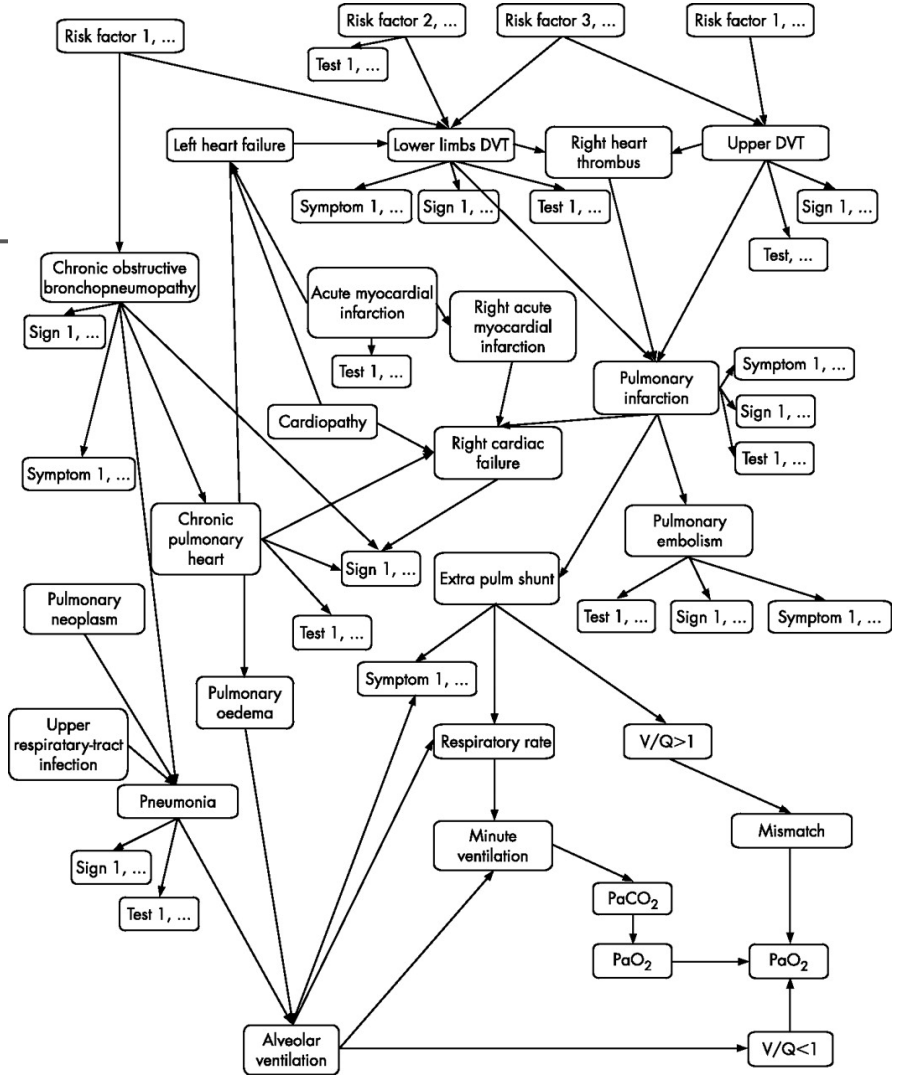

SUMO
PULO
PDO
PSO
Entity
Physical
Abstract
Object
Process
Quantity
Graph
Proposition
Relation
GraphElement
SetOrClass
Region
Collection
SelfConnectedObject
Agent
Argument
InformationSubject
FieldOfStudy
Procedure
Food
Transitaway
Substance
AdditionalSubject
InventionSubject
Apparatus
Product
Process
Material
CorpuscularObject
ContentBearingObject
OrganicObject
Artifact
SymbolicString
LinguisticExpression
icon
DocumentElement
ArtWork
StationeryArtifact
Device
Fabric
Clothing
Product
Sentence
Phrase
Language
Word
Morheme
TransportationDevice
Weapon
Machine
AttachingDevice
MathFormula
Table
Text
ChemicalFormula
Reference
Figure
MusicalInstrument
MeasuringDevice
Text
Patent Structure Ontology (p.3) – Claim
Patent Metadata Ontology
EngineeringComponent
MotionPicture
Book
FictionalText
Certificate
Summary
FactualText
Article
Section
Series
Patent
FinancialInstrument
Patent Drawings Ontology
Document
Patent Structure Ontology (p.1) - DocumentsSections
Patent Structure Ontology (p.2) - EpoKindDocument

Simulation graphs

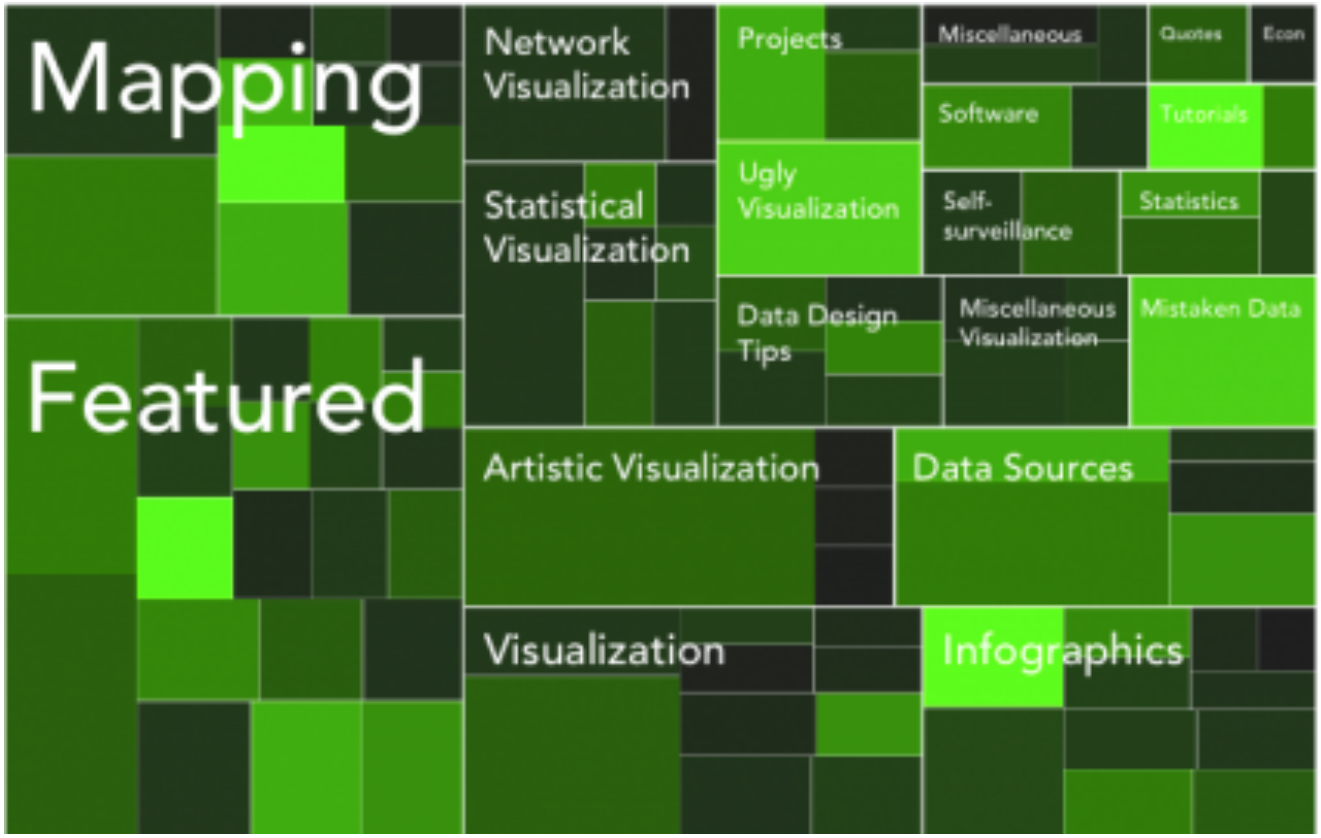


Probability maps

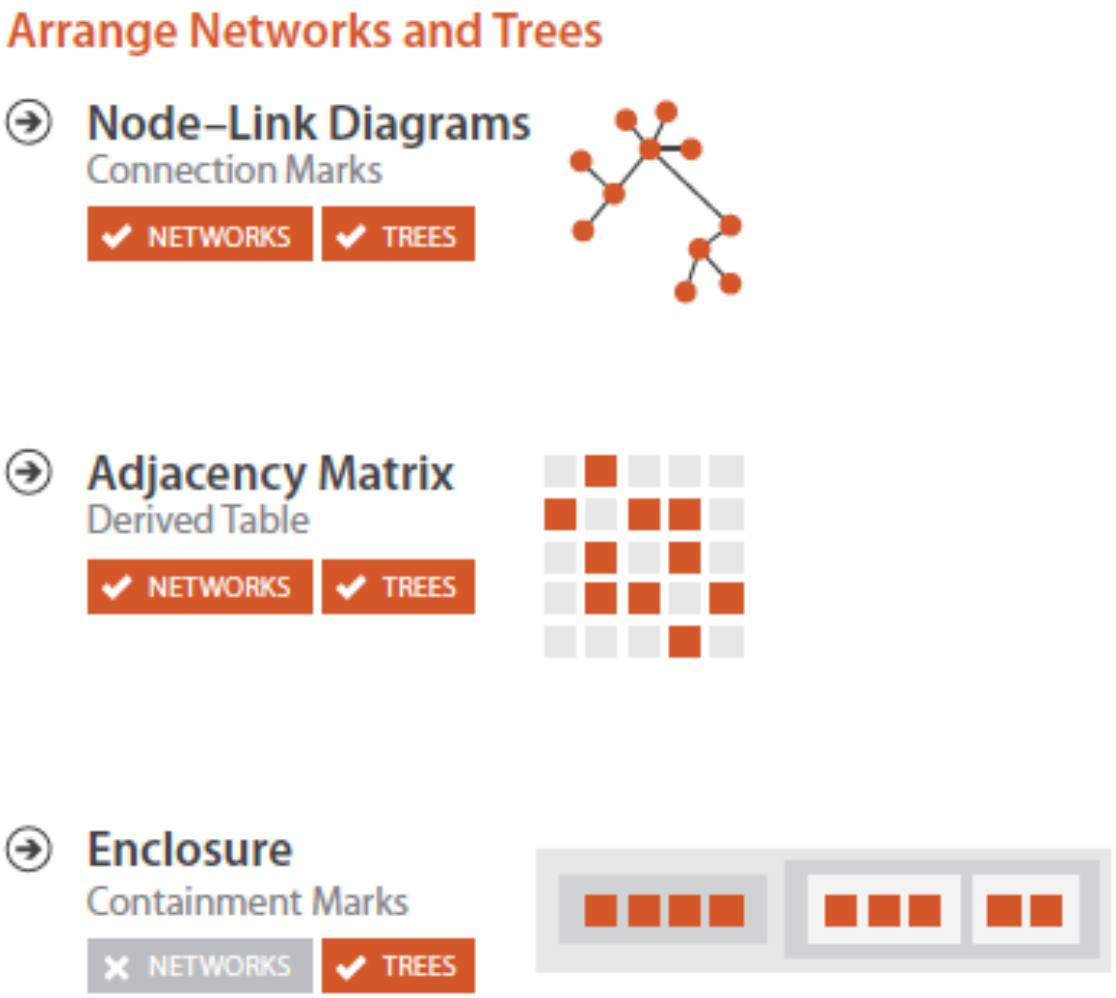
- Bayesian network



Is this a network?



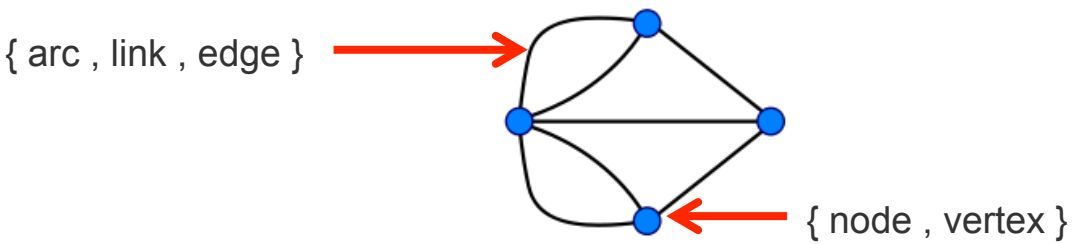
- Several ways to represent relations
- Choice depends on relation, task and scale



T. Munzner, Visualization Analysis and Design.

Graphs as Network Models

- Graphs are only ONE way to represent networks
  - Most powerful
  - Most popular
- Graphs are well suited for topology-related problems
  - Distance through the network (hops), propagation, clusters defined by connectivity



points	lines	
vertices	edges, arcs	math
nodes	links	computer science
sites	bonds	physics
actors	ties, relations	sociology

Source: L. Adamic SNA class @coursera

Network Visualization, QMSS @ Columbia

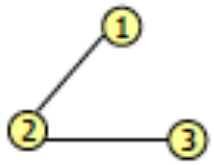
@denisparra

A formal definition

- A graph is a way of specifying **relationships** among a **collection** of **items**.
- A graph consists of a **set of objects**, called **nodes** or **vertices**, with certain pairs of these **objects** connected by **links** called **edges**.
  - Easley and Kleinberg, 2011

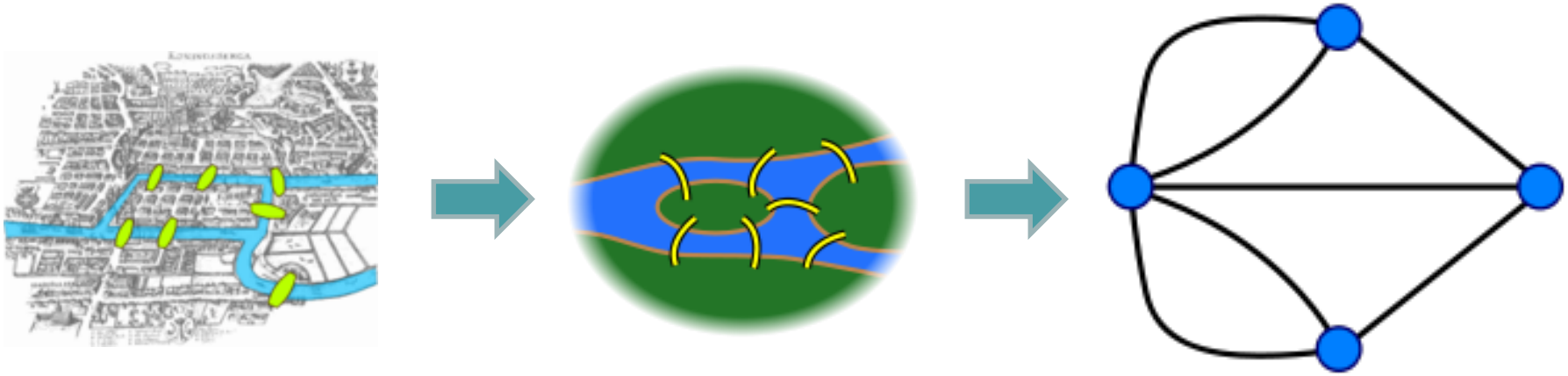
	1	2	3
1	0	1	0
2	1	0	1
3	0	1	0

Adjacency matrix



A bit of history: Graph models

- Around 1735, the mathematician Venn Euler set the foundation for graph theory by creating a model to represent the problem of the “7 bridges of Königsberg”



Source: http://en.wikipedia.org/wiki/Seven_Bridges_of_K%C3%B6nigsberg and “Linked” by A-L. Barabasi

Tree Data

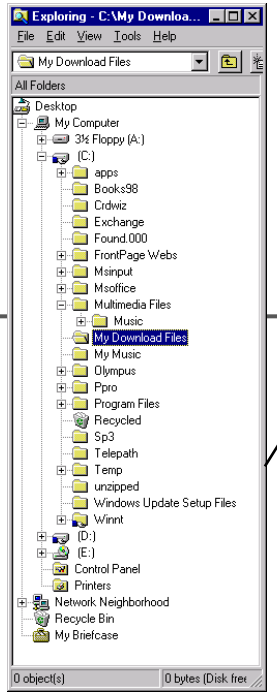
A tree can be thought of as simply a special kind of network.

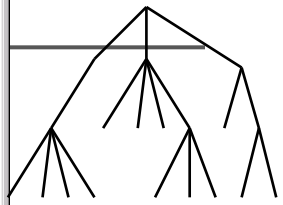
Tree layouts

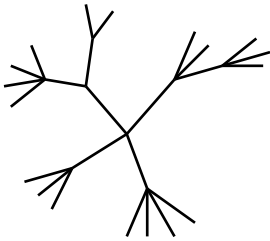
- Topological structure with hierarchy
- Parent-child relations are paramount
- can have explicit sibling links but usually implicit

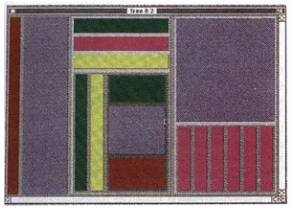
Hierarchical Information

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





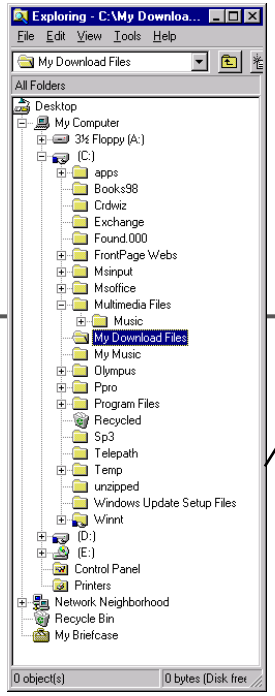


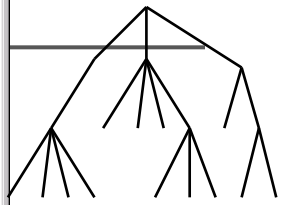


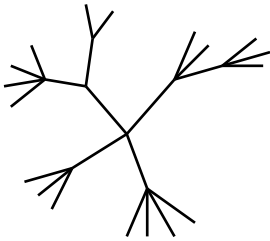
SFU

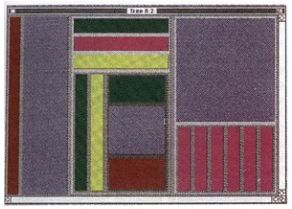
Hierarchical Information

- 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



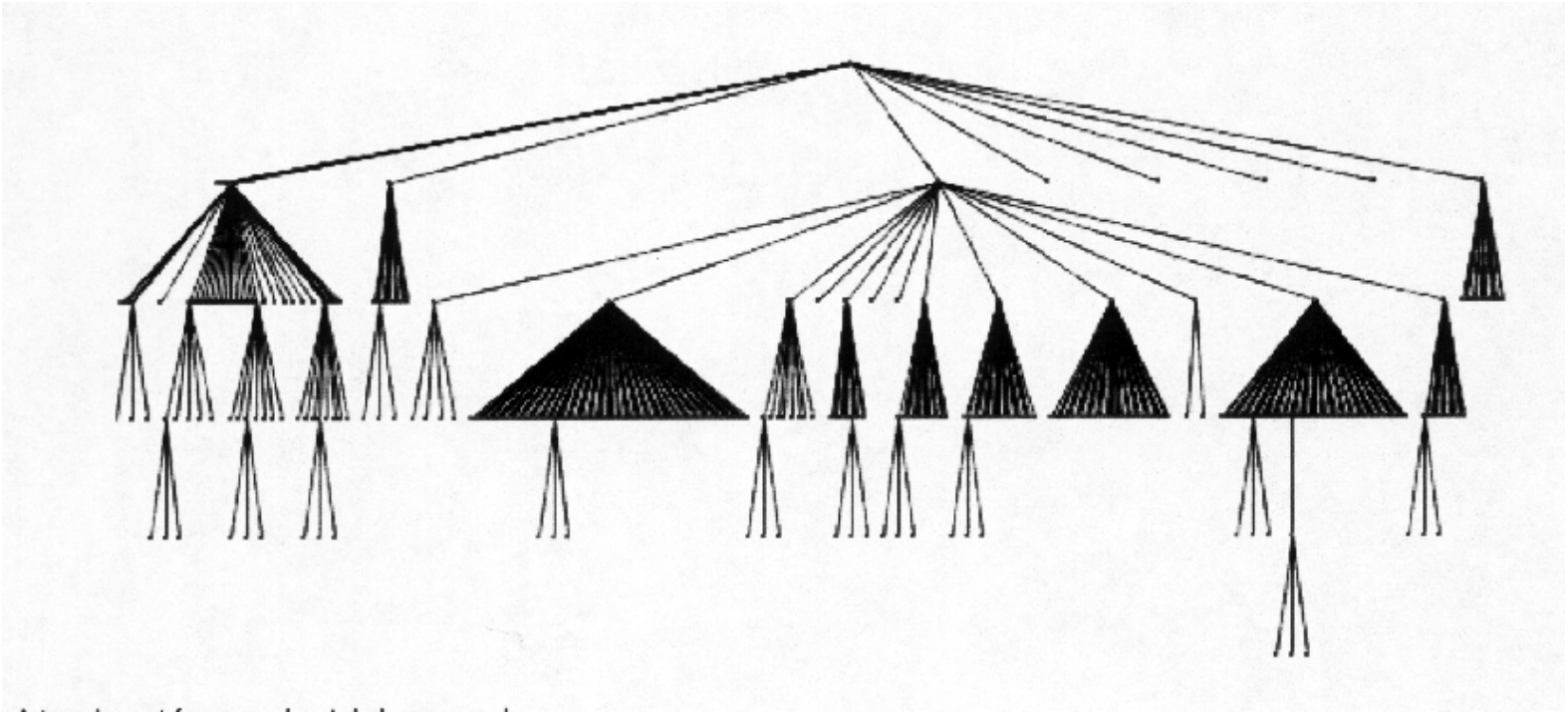






SFU

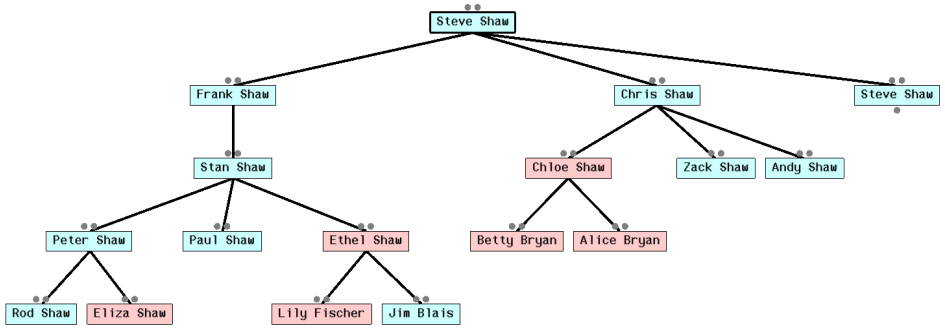
Hierarchical Data – Traditional Node-Link Layout



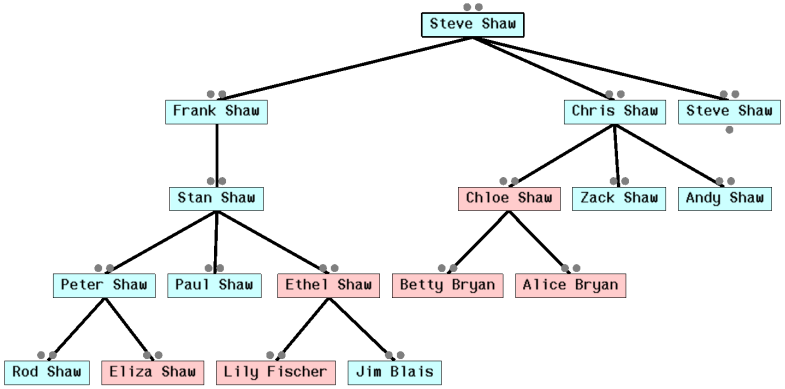
Allocate Space proportional to # of Children at Different Levels

SFU

Classical/Layered

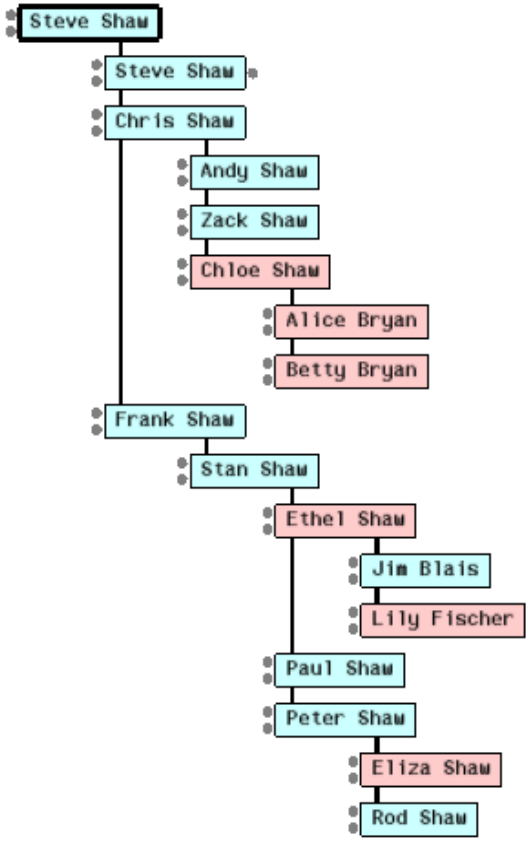


A naïve and easy-to-program layout: each subtree has an interval in x that is not overlapped by the neighboring subtrees. A postorder depth-first-traversal combines the intervals of subtrees to yield the interval of a parent node.



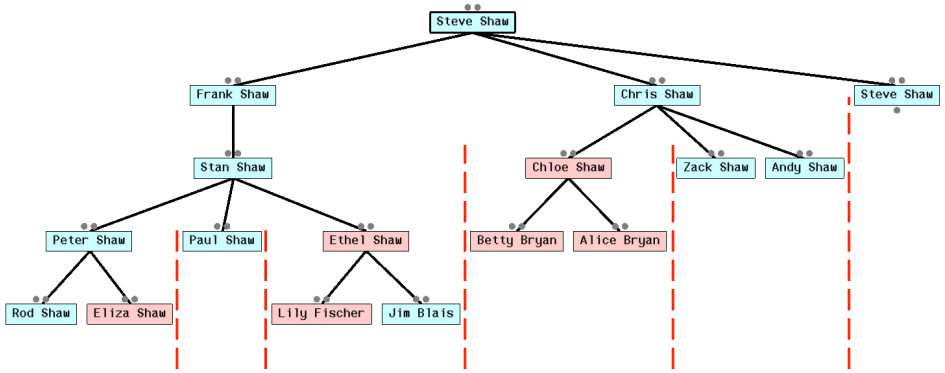
A "Reingold Tilford" style layout : saves space in x by moving subtrees together as much as possible. (For details, see section 3 of Christoph Buchheim, Michael Jünger and Sebastian Leipert, "Improving Walker's Algorithm to Run in Linear Time", Proceedings of Symposium on Graph Drawing (GD) 2002, pages 344-353.) The is more complicated to program.

Indented Outline List

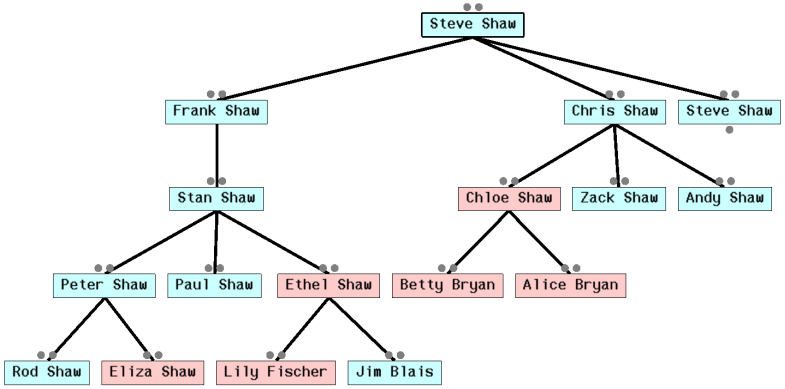


Another easy-to-program layout: a preorder depth-first-traversal encounters nodes in order of their y coordinates, and the x coordinate of each node is proportional to its depth.

Classical/Layered

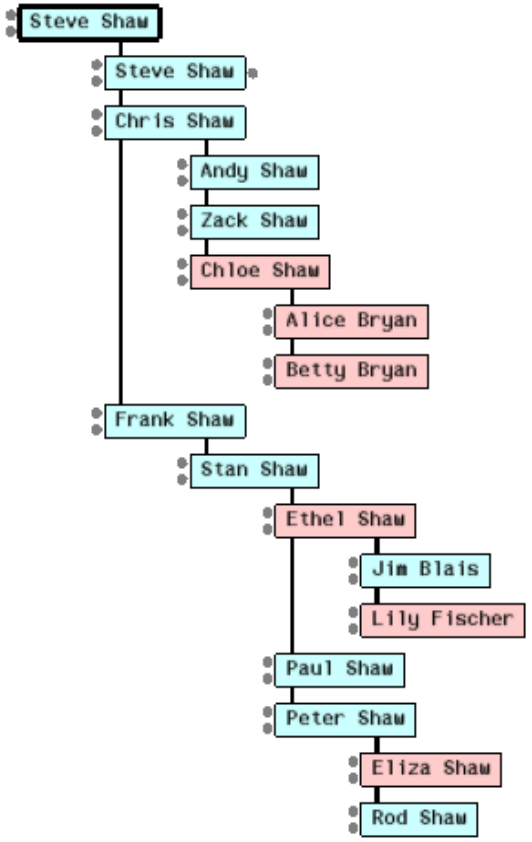


A naïve and easy-to-program layout: each subtree has an interval in x that is not overlapped by the neighboring subtrees. A postorder depth-first-traversal combines the intervals of subtrees to yield the interval of a parent node.



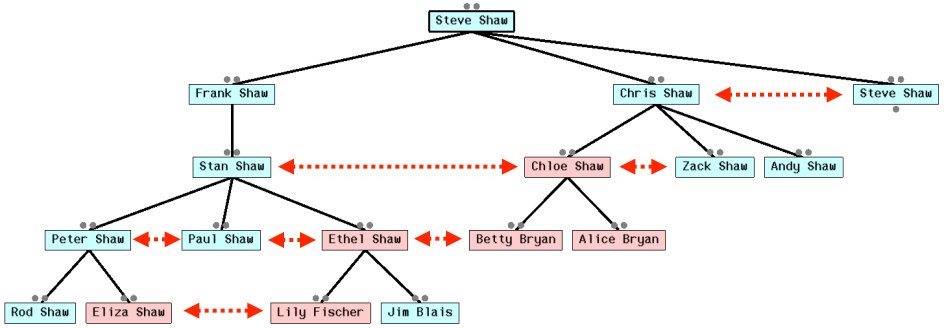
A "Reingold Tilford" style layout : saves space in x by moving subtrees together as much as possible. (For details, see section 3 of Christoph Buchheim, Michael Jünger and Sebastian Leipert, "Improving Walker's Algorithm to Run in Linear Time", Proceedings of Symposium on Graph Drawing (GD) 2002, pages 344-353.) The is more complicated to program.

Indented Outline List

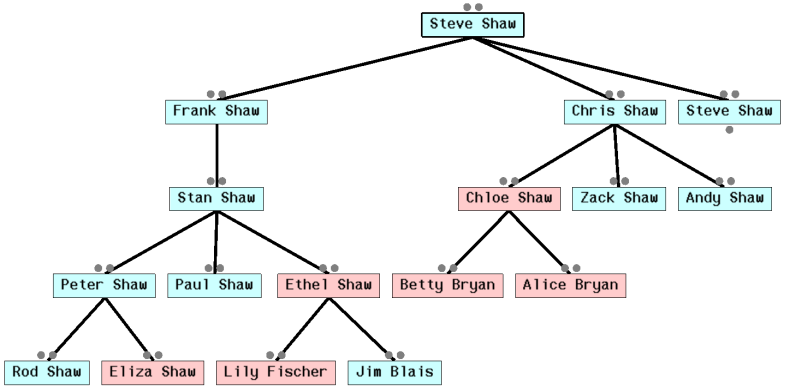


Another easy-to-program layout: a preorder depth-first-traversal encounters nodes in order of their y coordinates, and the x coordinate of each node is proportional to its depth.

Classical/Layered

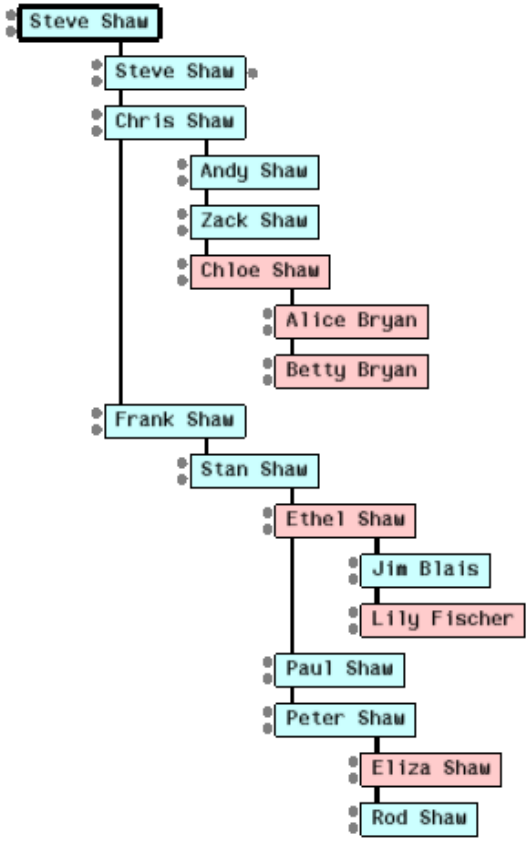


A naïve and easy-to-program layout: each subtree has an interval in x that is not overlapped by the neighboring subtrees. A postorder depth-first-traversal combines the intervals of subtrees to yield the interval of a parent node.



A "Reingold Tilford" style layout : saves space in x by moving subtrees together as much as possible. (For details, see section 3 of Christoph Buchheim, Michael Jünger and Sebastian Leipert, "Improving Walker's Algorithm to Run in Linear Time", Proceedings of Symposium on Graph Drawing (GD) 2002, pages 344-353.) The is more complicated to program.

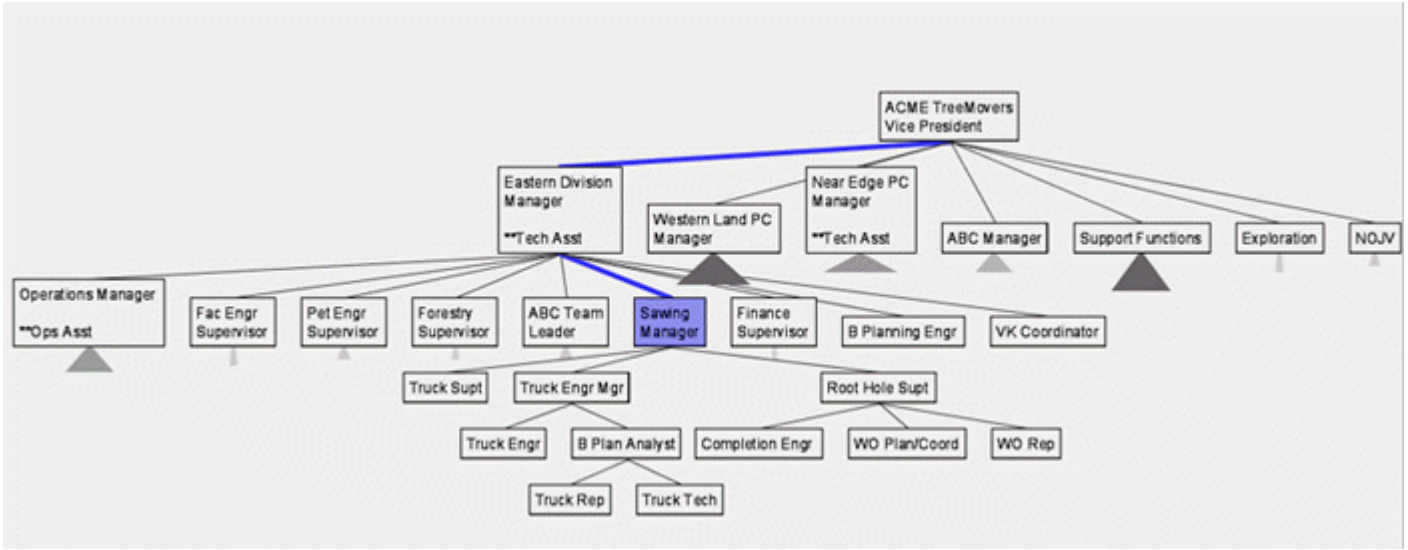
Indented Outline List



Another easy-to-program layout: a preorder depth-first-traversal encounters nodes in order of their y coordinates, and the x coordinate of each node is proportional to its depth.

Traditional Node-Link Layout ➔ SpaceTree

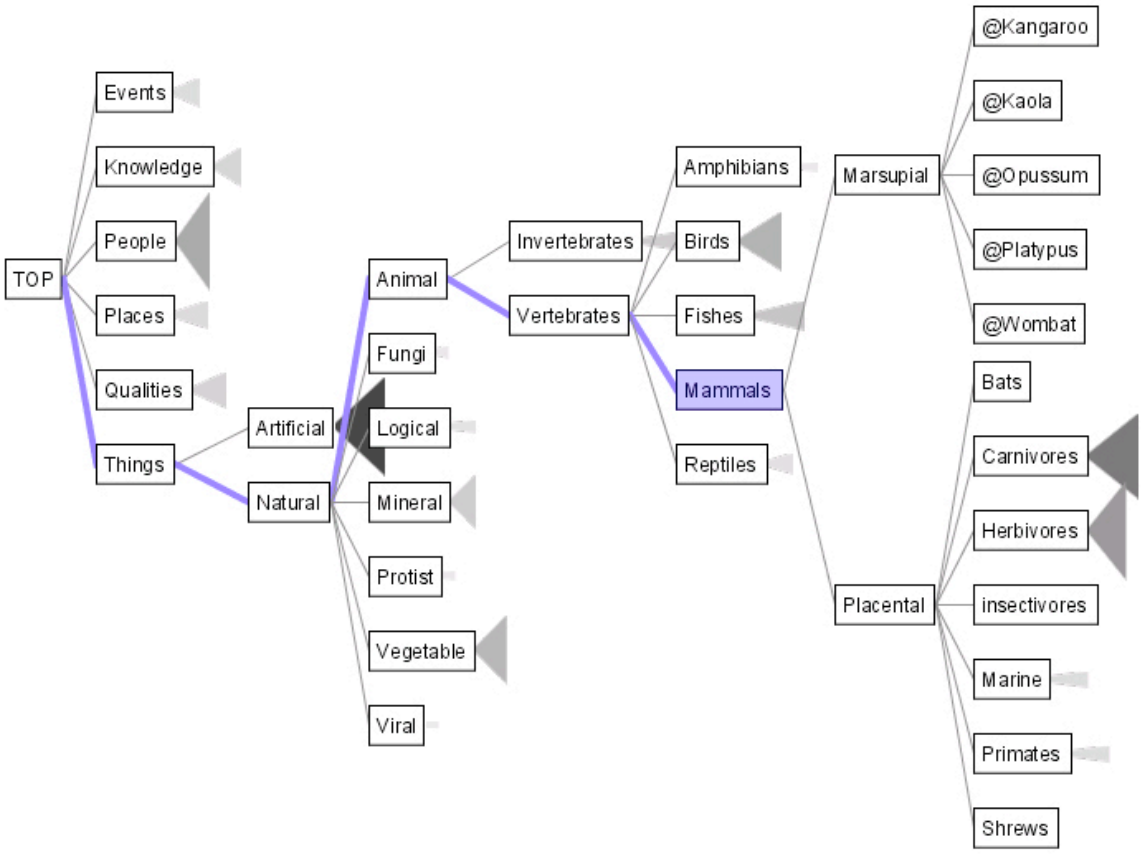
•HCI Lab – University of Maryland http://www.cs.umd.edu/hcil/spacetree/



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

SFU

Space Tree cont.



SFU

3D ConeTree

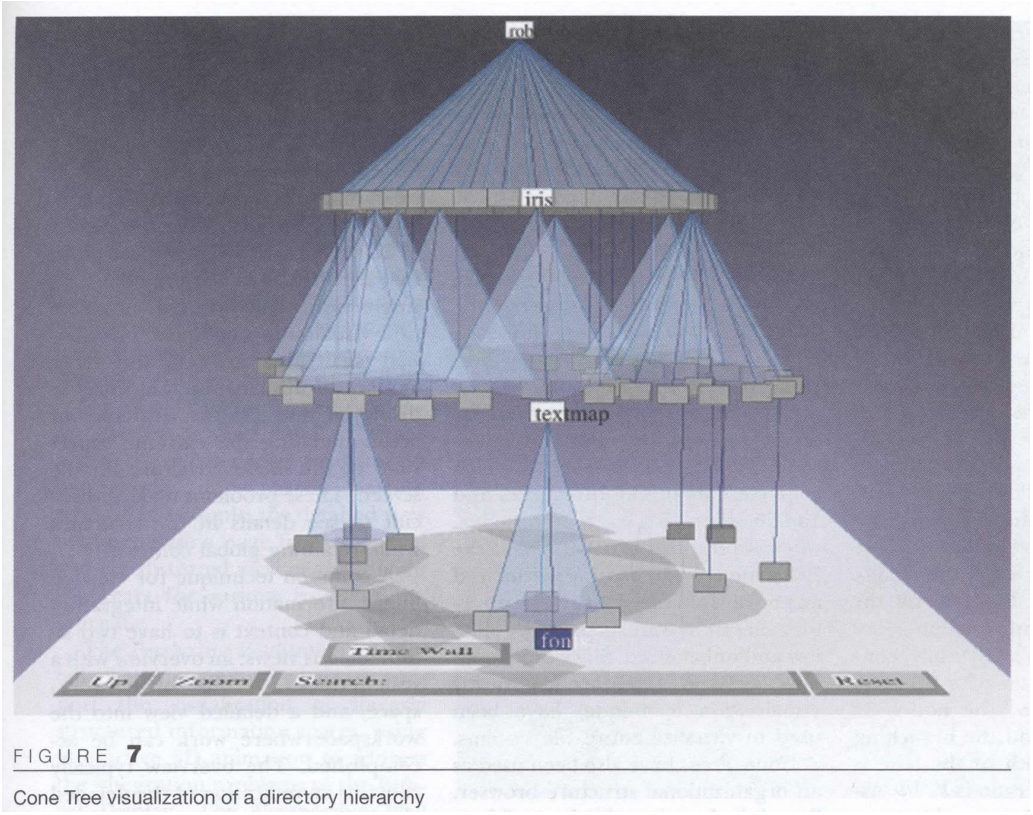


FIGURE 7

Cone Tree visualization of a directory hierarchy

- Positive
- Higher Information Density
- Smooth animation

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

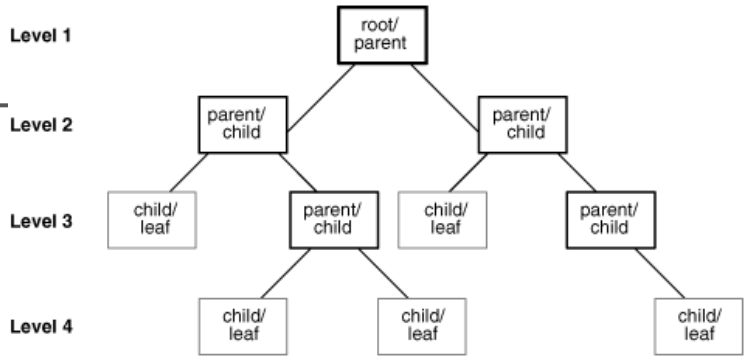
3D used to increase Information Density

Children laid out in a cylinder “below” parent

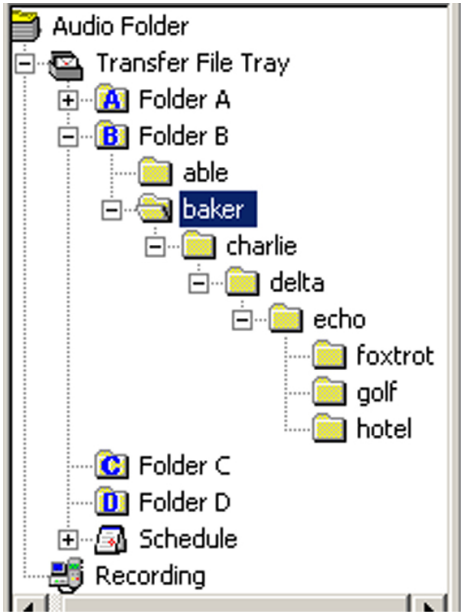
SFU

Tree-link diagrams

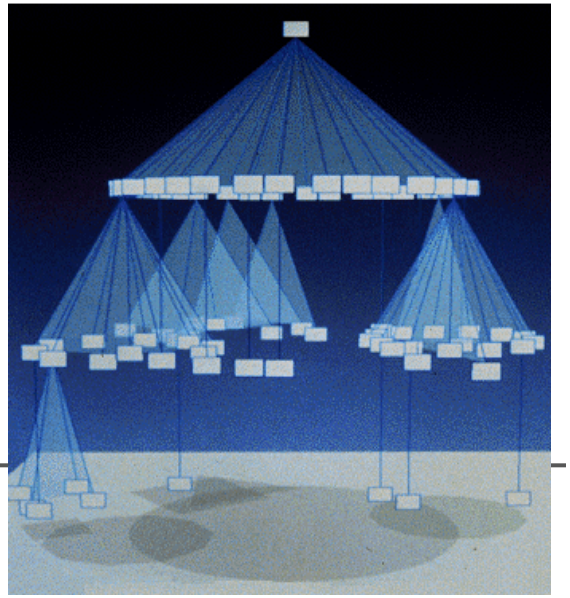
Classical representation



Tree view

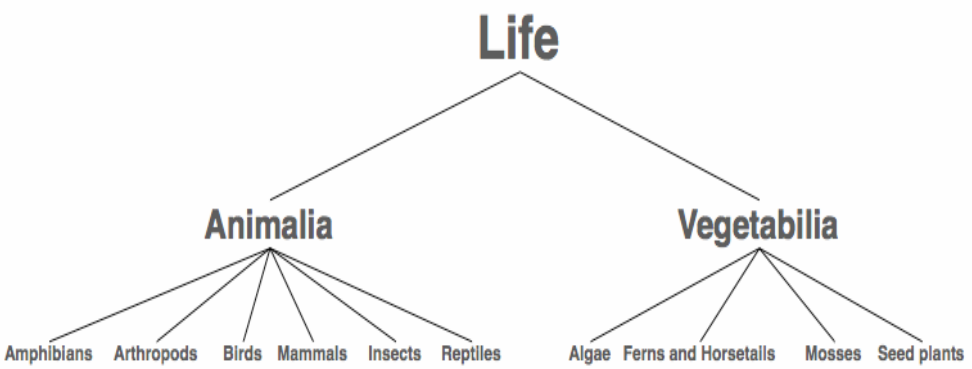


Cone tree

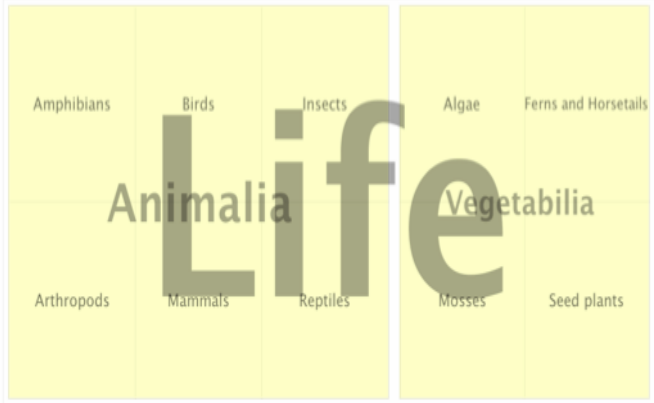


SFU

Tree – containment diagrams

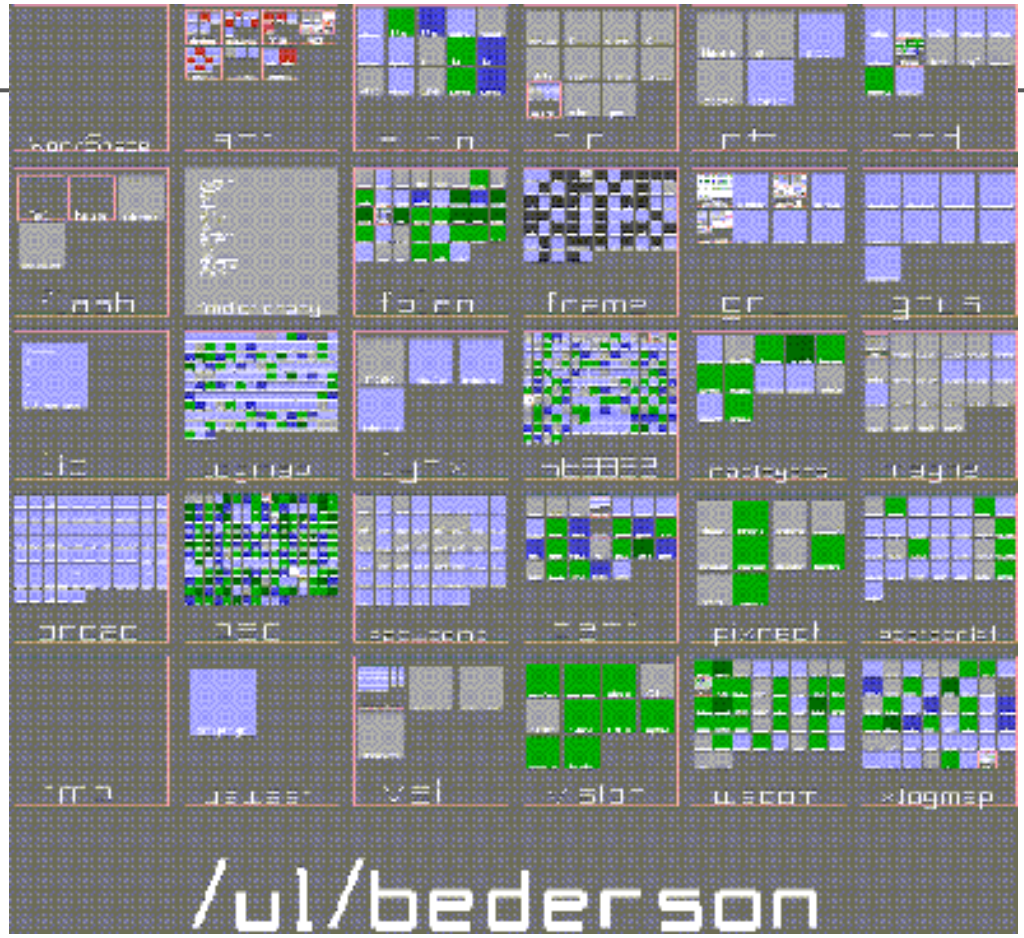


Treemap



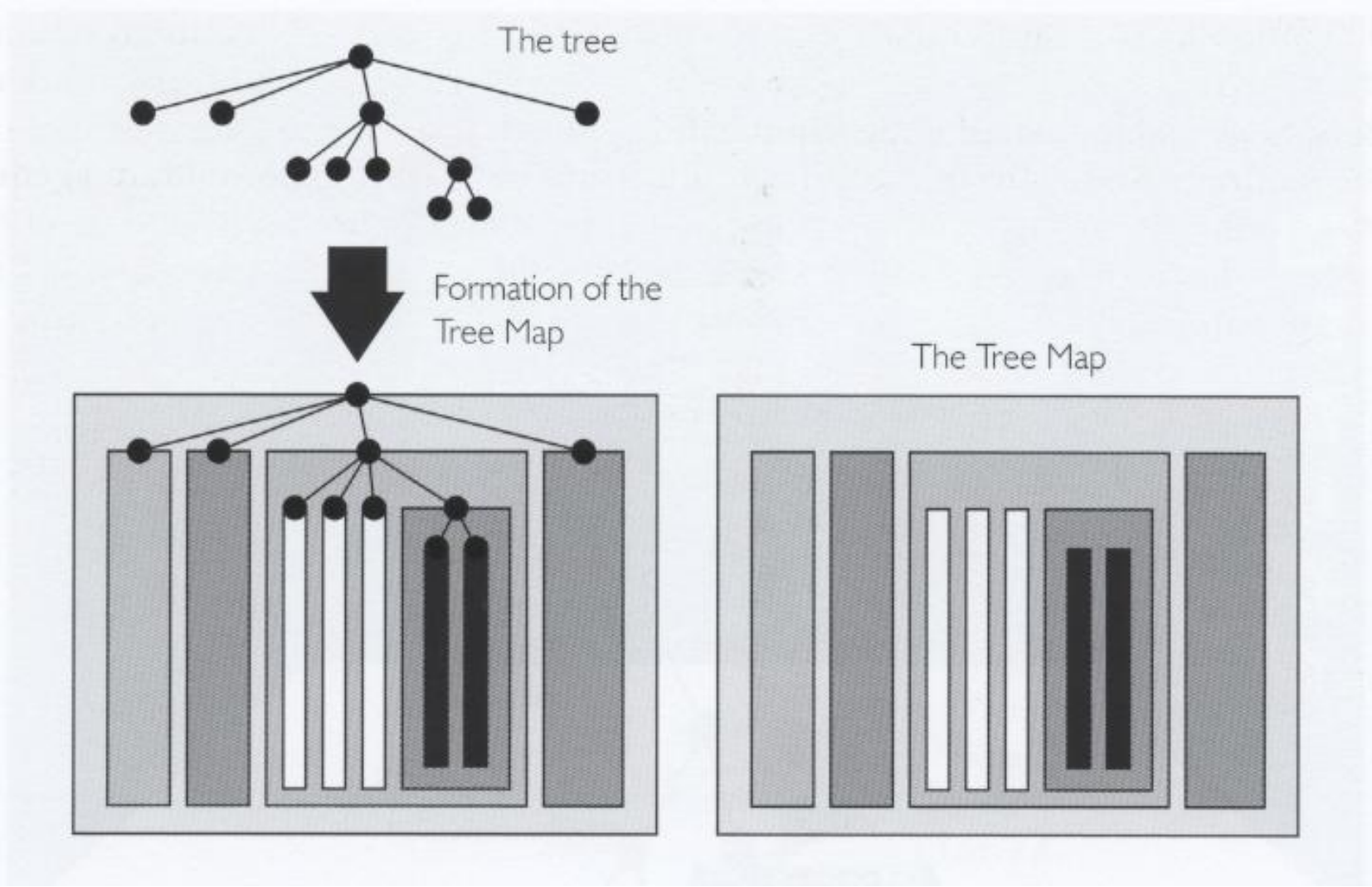
SFU

- Treemaps
- Recursive, repeated subdivision of space

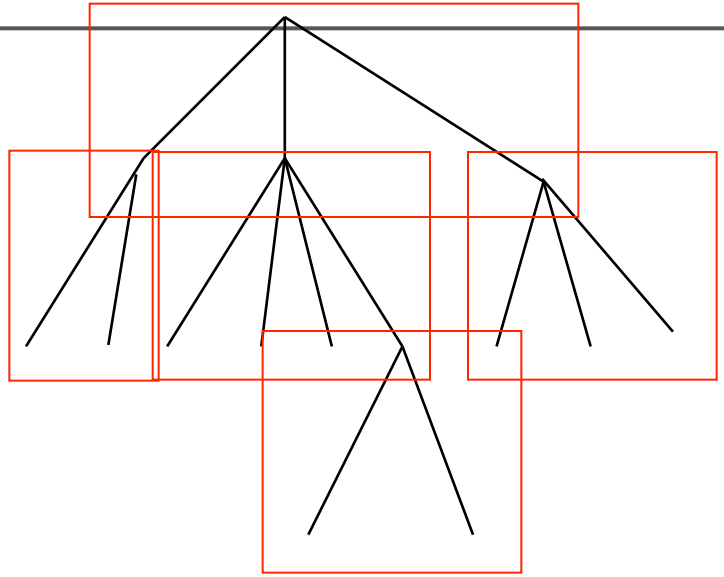


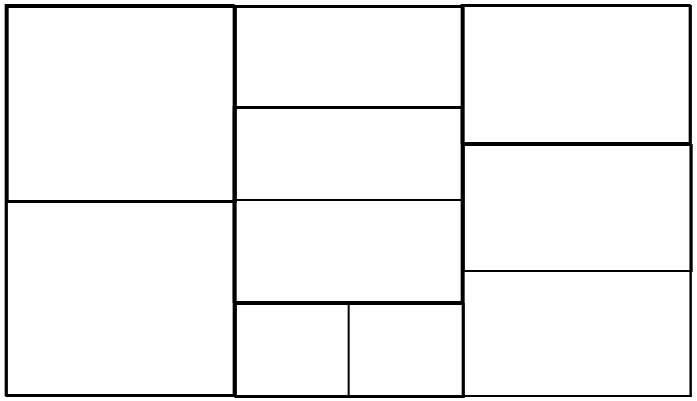
SFU

Tree layouts: containment

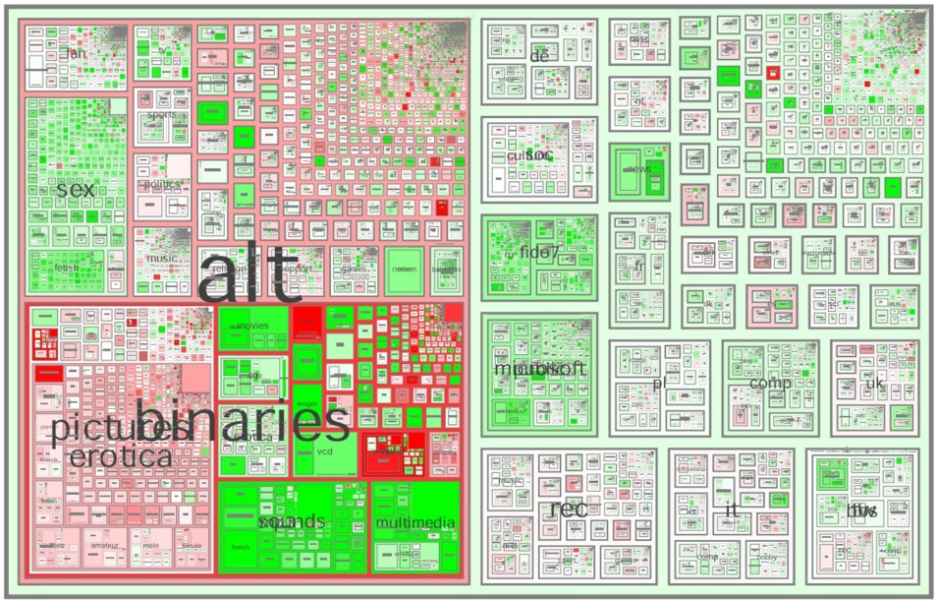


Treemaps – “Slice & Dice”

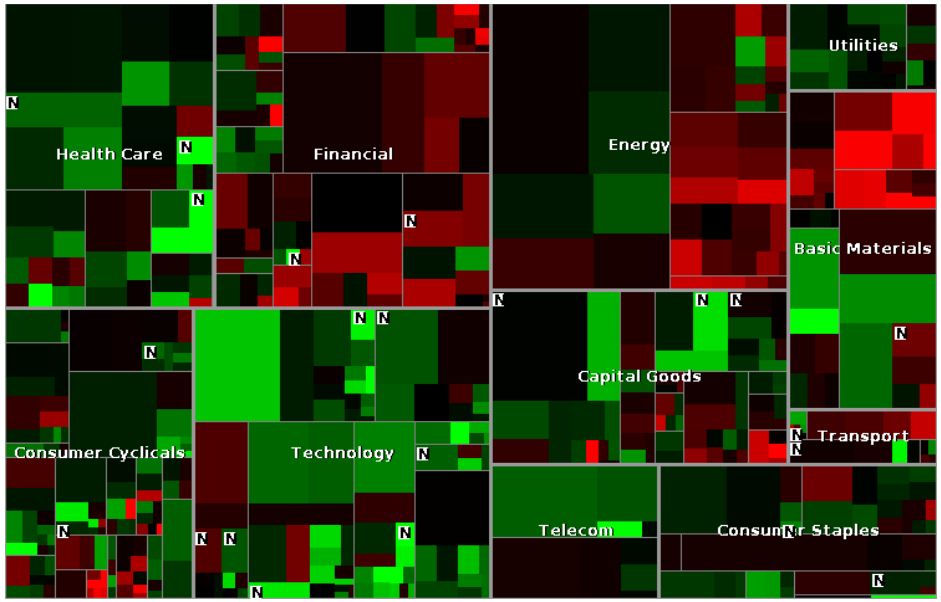




Treemaps use **containment** marks rather than **connectivity** marks



Marc Smith and Andrew Fiore, 2001



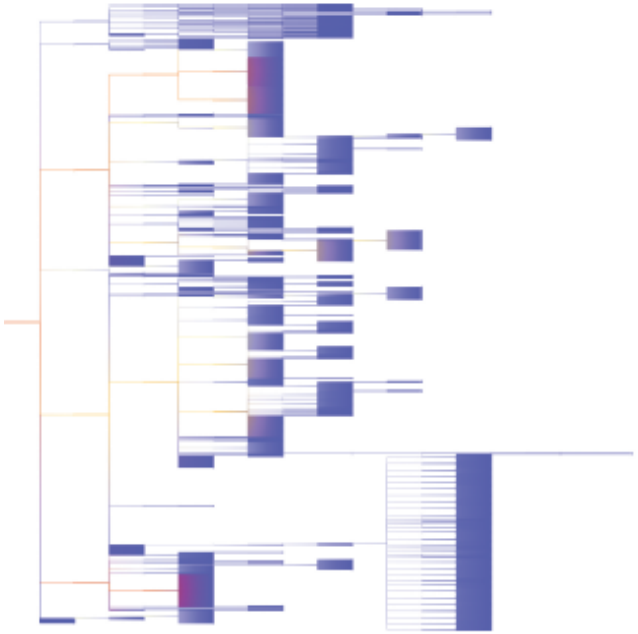
Martin Wattenberg, 1998
and
http://www.smartmoney.com/marketmap/

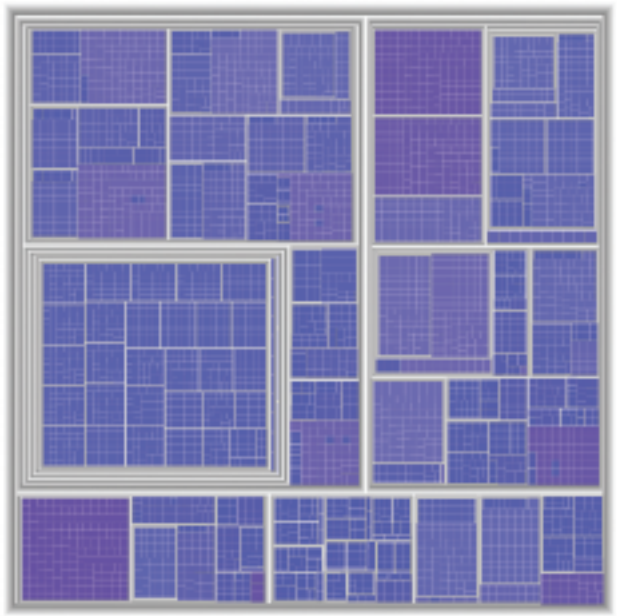
Treemaps

- 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

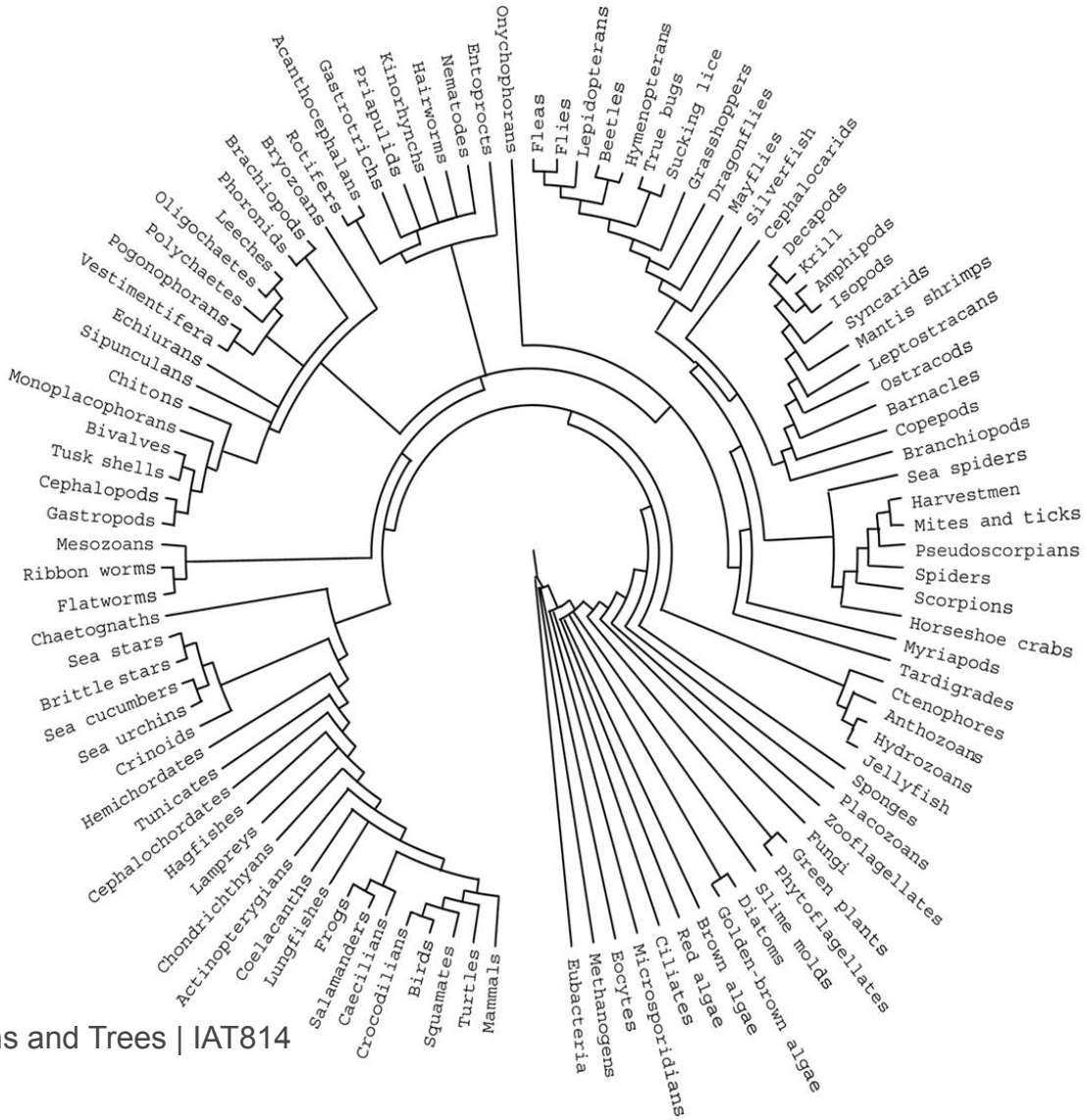
Treemaps

- Treemaps scale well to large number of nodes



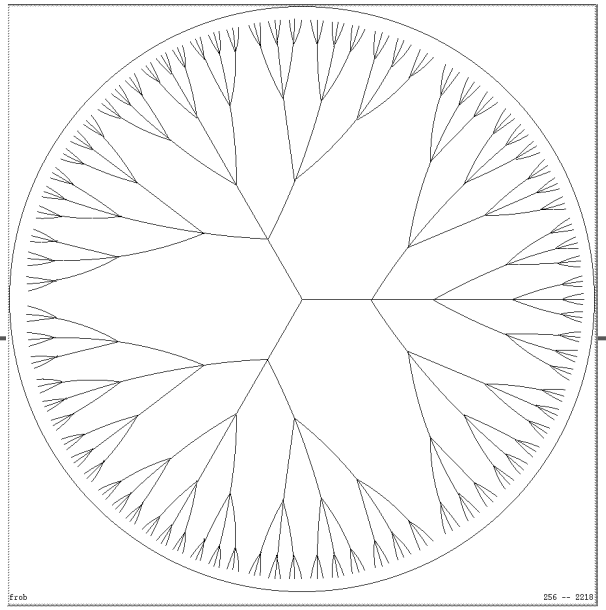


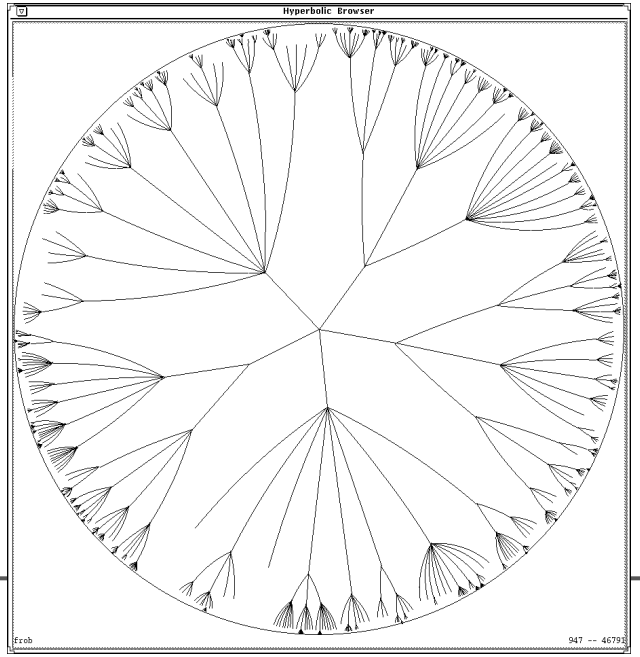
Trees


Onychophorans
Fleas
Flies
Lepidopterans
Beetles
Hymenopterans
True bugs
Sucking lice
Grasshoppers
Dragonflies
Mayflies
Silverfish
Cephalocarids
Decapods
Krill
Amphipods
Isopods
Syncarids
Mantis shrimps
Leptostracans
Ostracods
Barnacles
Copepods
Branchiopods
Sea spiders
Harvestmen
Mites and ticks
Pseudoscorpians
Spiders
Scorpions
Horseshoe crabs
Myriapods
Tardigrades
Ctenophores
Anthozoans
Hydrozoans
Jellyfish
Sponges
Placozoans
Zooflagellates
Fungi
Green plants
Phytoflagellates
Slime molds
Diatoms
Golden-brown algae
Brown algae
Red algae
Ciliates
Microsporidians
Eocytes
Methanogens
Eubacteria
Mammals
Turtles
Squamates
Birds
Crocodilians
Caecilians
Salamanders
Frogs
Lungfishes
Coelacanths
Actinopterygians
Chondrichthyans
Lampreys
Hagfishes
Cephalochordates
Tunicates
Hemichordates
Crinoids
Sea urchins
Sea cucumbers
Brittle stars
Sea stars
Chaetognaths
Flatworms
Ribbon worms
Mesozoans
Gastropods
Cephalopods
Tusk shells
Bivalves
Monoplacophorans
Chitons
Sipunculans
Echiurans
Vestimentifera
Pogonophorans
Polychaetes
Oligochaetes
Leeches
Phoronids
Brachiopods
Bryozoans
Rotifers
Acanthocephalans
Gastrotrichs
Priapulids
Kinorhynchs
Hairworms
Nematodes
Entoprocts

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





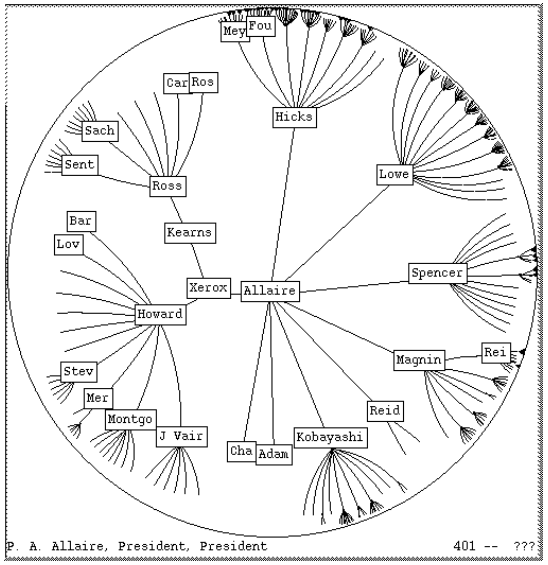
User orientation on refocus

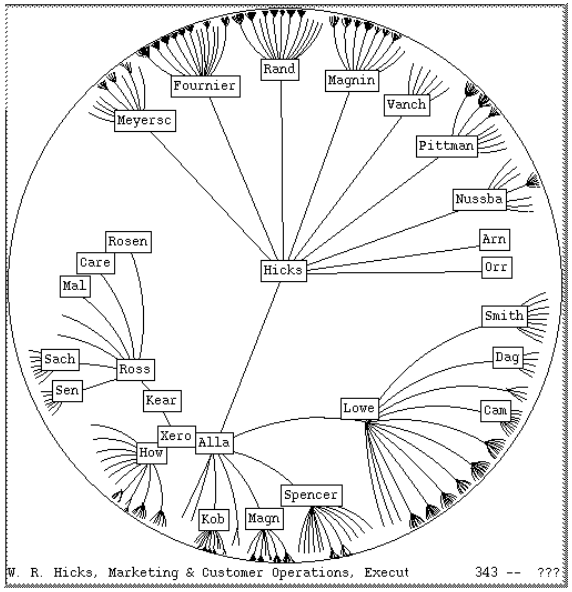
- Problem
  - Hyperbolic Geometry can allow disorienting rotations of objects when refocusing
- Solution one:
  - Preserve initial angular orientation of parent to child nodes
- Solution two:
  - Preserve left to right orientation of parent to child nodes beginning with initial display

- Note: both rely on relative geometric consistency

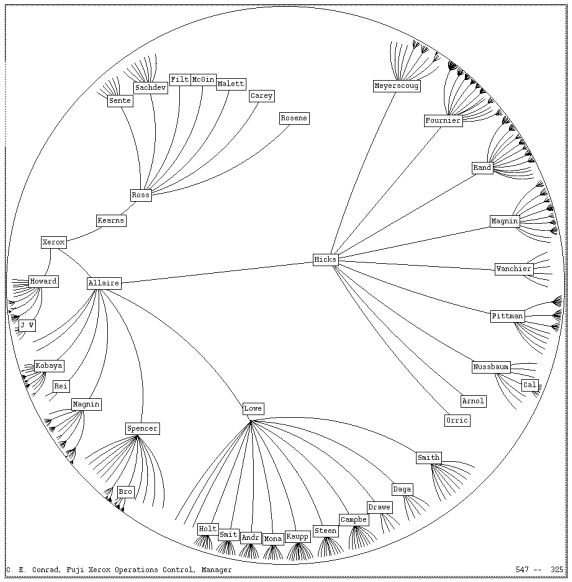
User orientations - Solutions

Preserving Angular Orientation



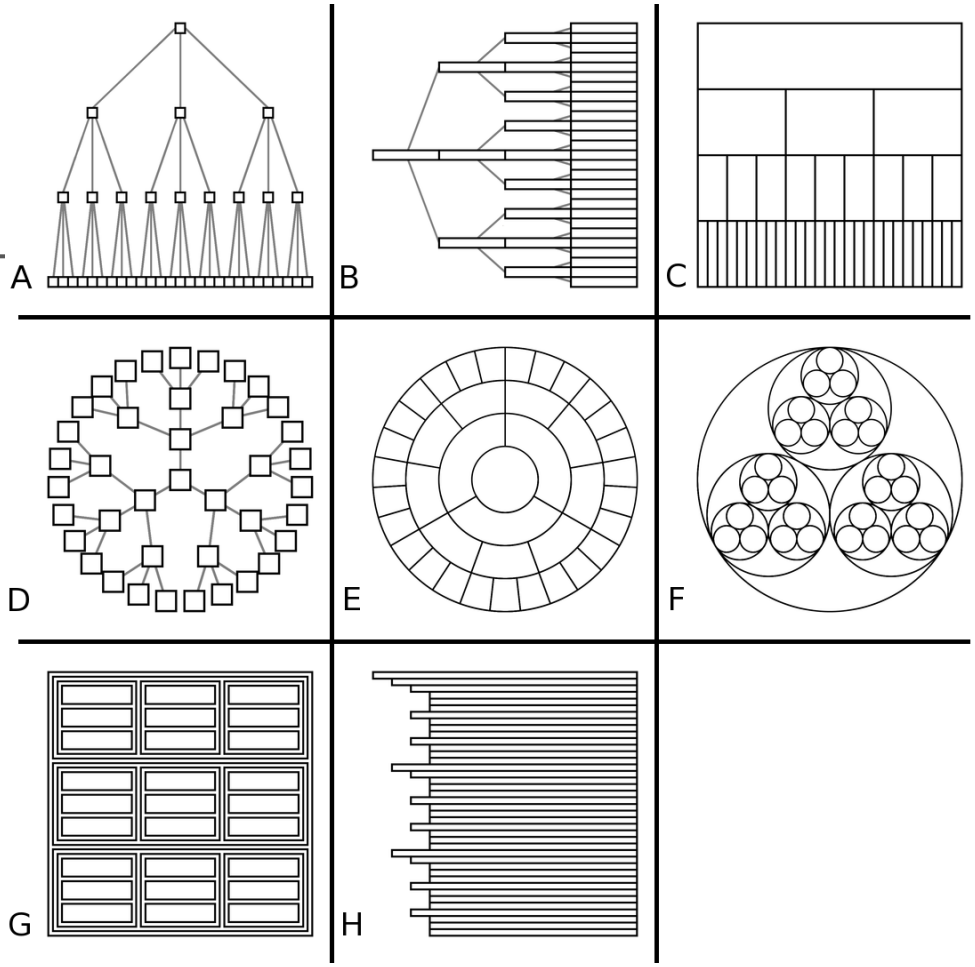


Left to Right Ordering



Trees

- Different tree idioms use containment and/or connectivity
- Spatial position also meaningful in some
- Different visual channels

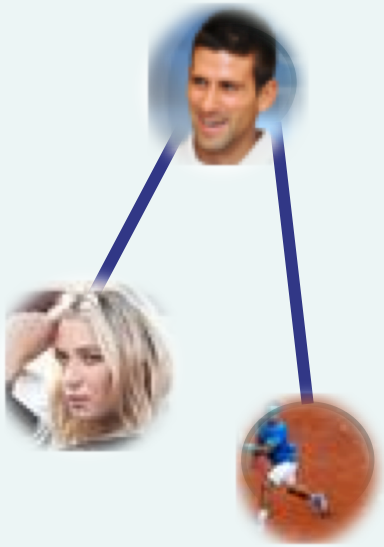
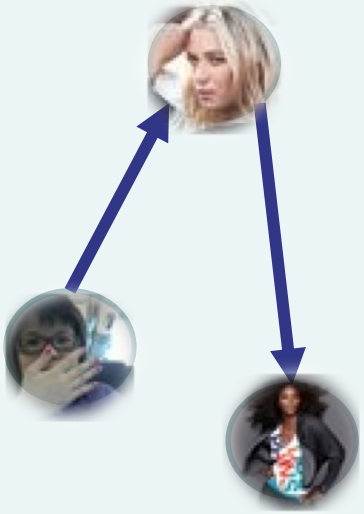
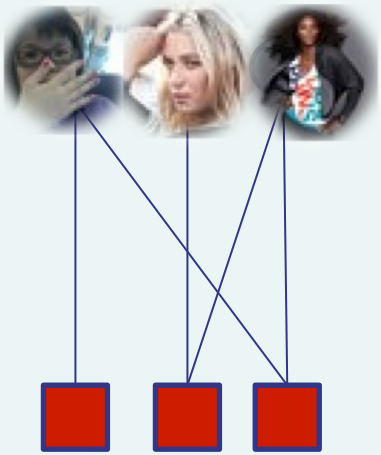


Michael McGuffin and Jean-Marc Robert, 2010

Graphs are more complicated

Graph Terminology

- Graphs can have cycles
- Edges can be directed or undirected
- Degree of a vertex = # connected nodes
  - In-degree and out-degree for directed graphcs
- Graph edges can have values (weights)
  - Nominal (N), ordinal (O), quantitative (Q)

Undirected (Facebook friendships)	Directed (Twitter following)	multimode (Amazon user-product)	Weighted (Facebook likes)	and more ….
				

Network Analysis measures

Node-level metrics

- Centrality
  - (In/Out) Degree
  - Betweenness
  - Closeness
  - Eigenvector
- Clustering coefficient

Graph-level metrics

- Size
- Diameter (longest *path*)
- Average path length
- Average [*node metric*]

- These are only a few representative measures
- http://www.slideshare.net/gcheliotis/social-network-analysis-3273045

Interpretation of measures

	Interpretation in Social Networks
Degree	How many people can this person reach directly?
Betweenness	How likely is this person to be the most direct route between two people in the network?

Source: http://www.slideshare.net/gcheliotis/social-network-analysis-3273045 slide 24

Interpretation of measures

	Interpretation in Social Networks
Closeness	How fast can this person reach everyone in the network?
Eigenvector	How well is this person connected to other well-connected people?

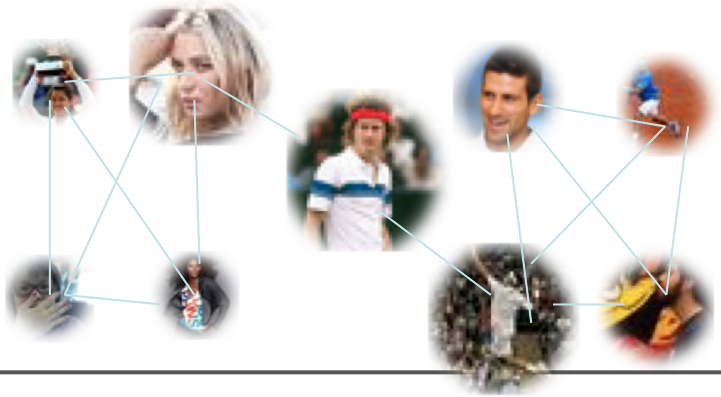
Source: http://www.slideshare.net/gcheliotis/social-network-analysis-3273045 slide 24

Two more concepts…

- Total possible number of edges in a network
  - #edges = n * (n -1) /2 (undirected network)
  - #edges = n * (n -1) (directed network)

- (Shortest) Path: the shortest sequence of edges to be followed to reach a node B from a node A in a network.

Which is the length of the shortest path between Rafa Nadal and Sharonpova?

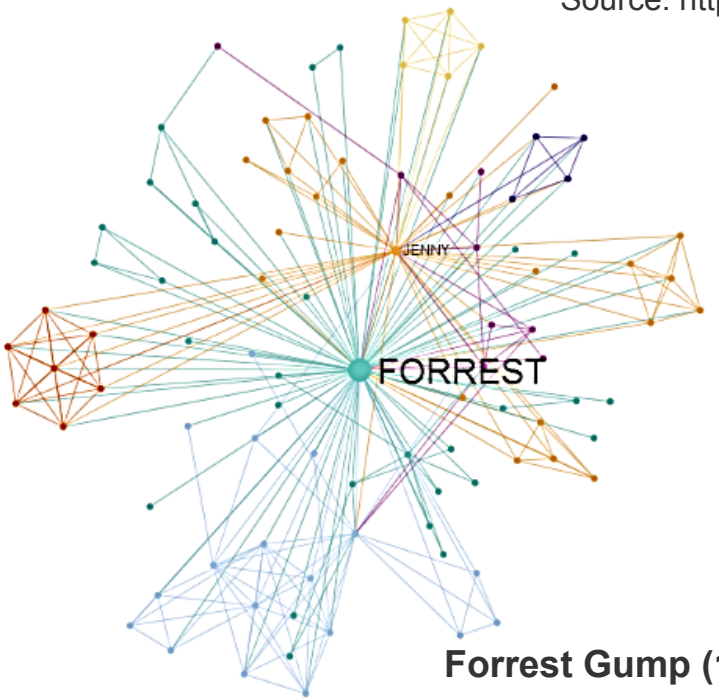


SFU

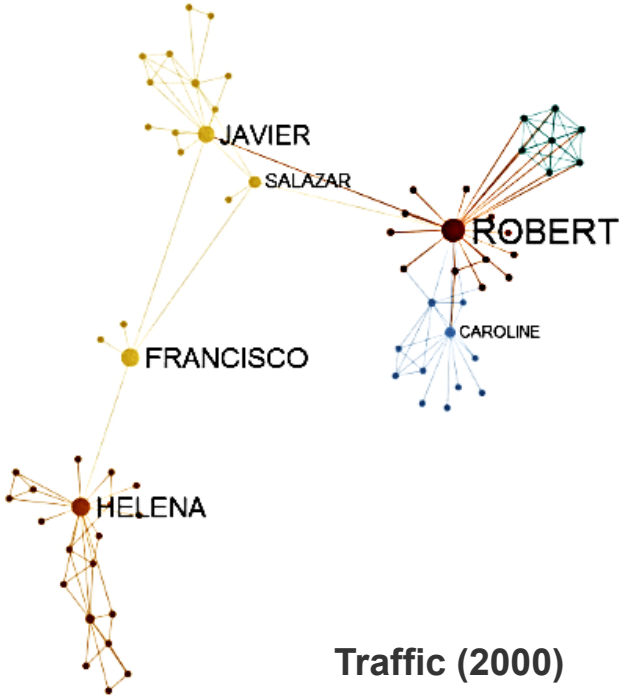
Using these concepts in network analysis

- compare these 2 movie networks (characters' interactions)

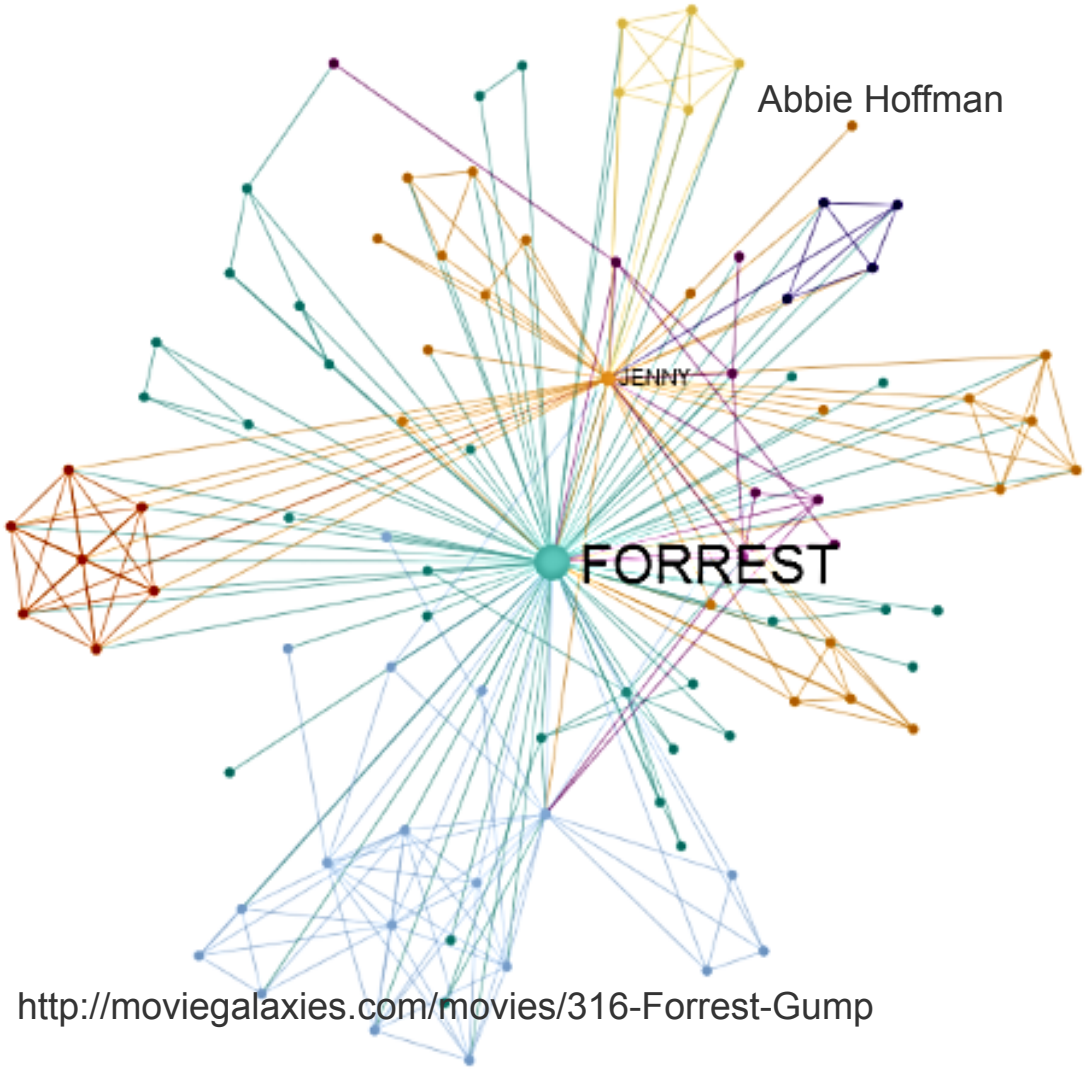
Source: http://moviegalaxies.com



Forrest Gump (1994)



Traffic (2000)



http://moviegalaxies.com/movies/316-Forrest-Gump

Network metrics:

- Size: 94/271
- Density: 0.06
- Diameter: 4
- Clustering coefficient: 0.8
- Avg. Path Length: 1.99

Node metrics:

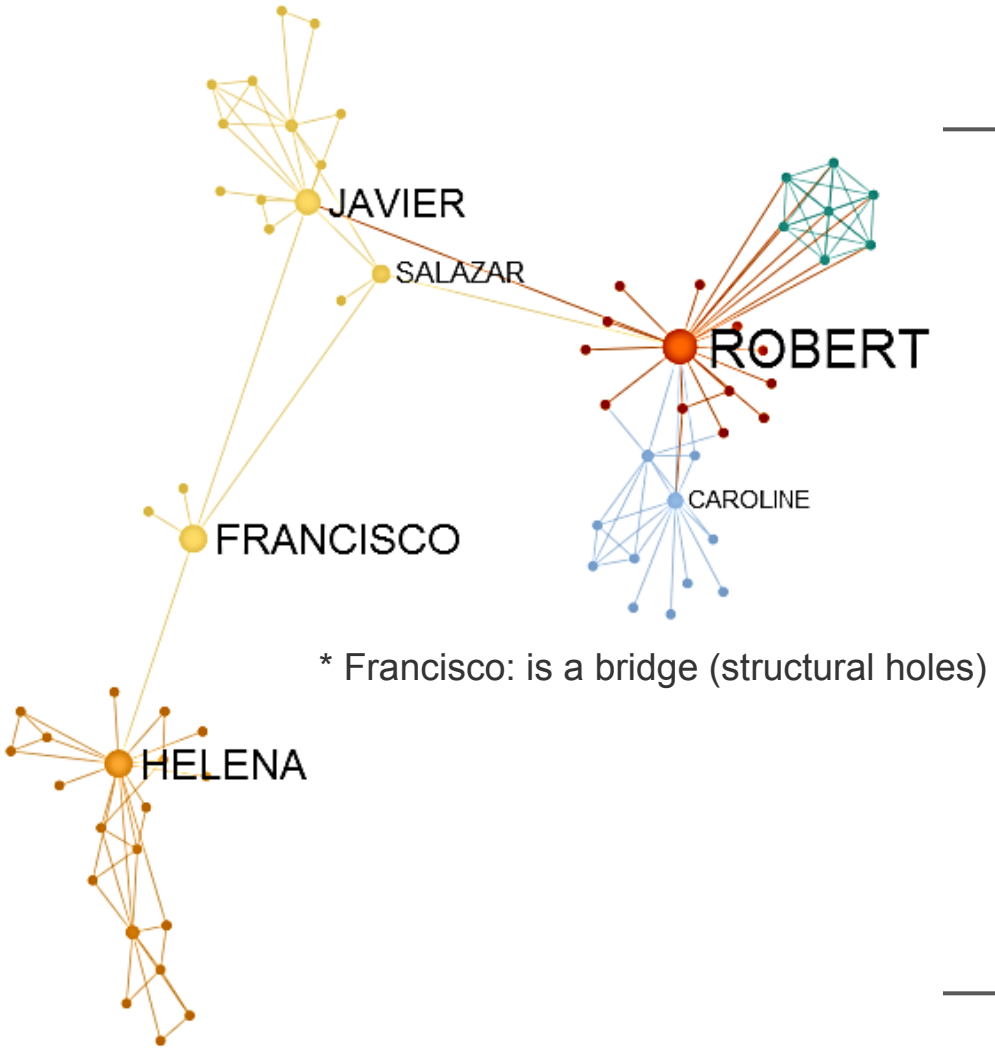
Forrest

- Degree: 89
- Betweetnness: 3453.8

Abbie Hoffman

- Degree: 6
- Betweenness: 0

Traffic (2000)



* Francisco: is a bridge (structural holes)

Network metrics:

- Size: 68
- Density: 0.04
- Diameter: 7
- Clustering coefficient: 0.55
- Avg. Path Length: 3.54

Node metrics:

Robert

- Degree: 24
- Betweetnness: 1437.7

Francisco

- Degree: 5
- Betweenness: 1031

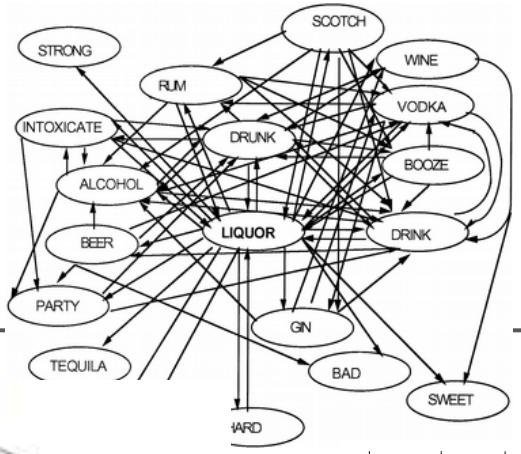
http://moviegalaxies.com/movies/837-Traffic

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.

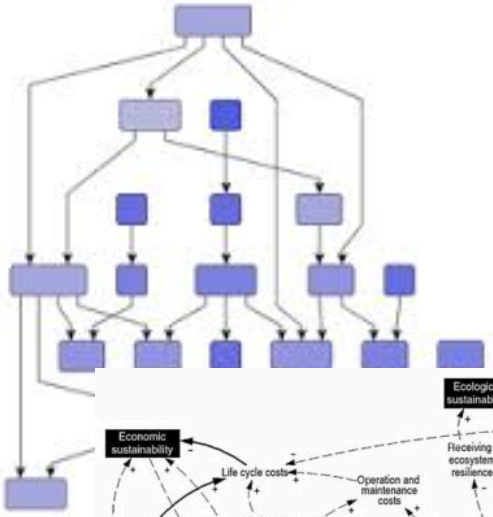
    Source: Herman, Graph Visualization and Navigation in Information Visualization: a Survey

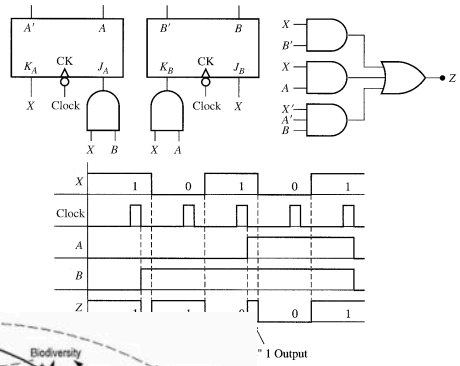
What kinds of relations?

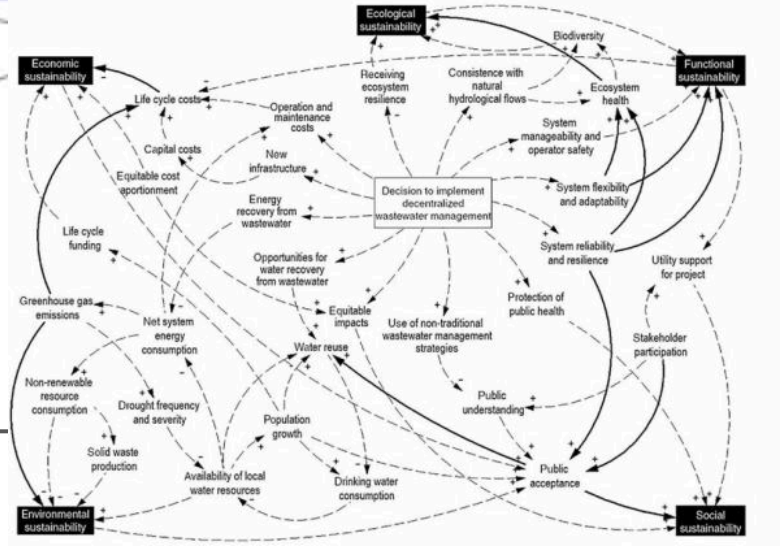


This is speculative and incomplete !

- Associative (IS-A)
  - A is like B in some way
- Ordinal/Sequential
  - A > B, precedes B, in so
  - Time
- Hierarchy
  - A is the parent of B and C
- Influence
  - Causal
  - dependency





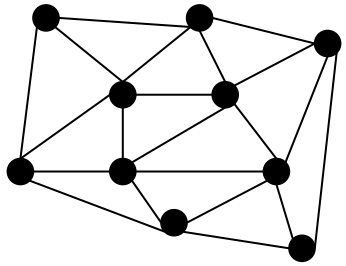


Graph Visualization Challenges

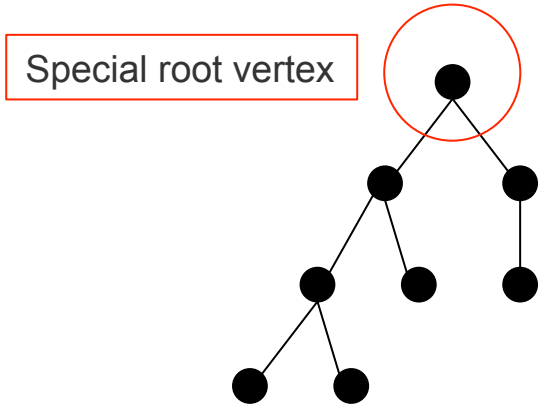
- Graph layout and positioning
  - Make a concrete rendering of abstract graph
- Navigation/Interaction
  - How to support user changing focus and moving around the graph
- Scale
  - Above two issues not too bad for small graphs, but large ones are much tougher

Graph and tree data structures

- Graphs:
  - Structured, connected data



- Trees are different
  - N-1 edges
  - No cycles
  - hierarchical data
  - Directed edges (implicit)



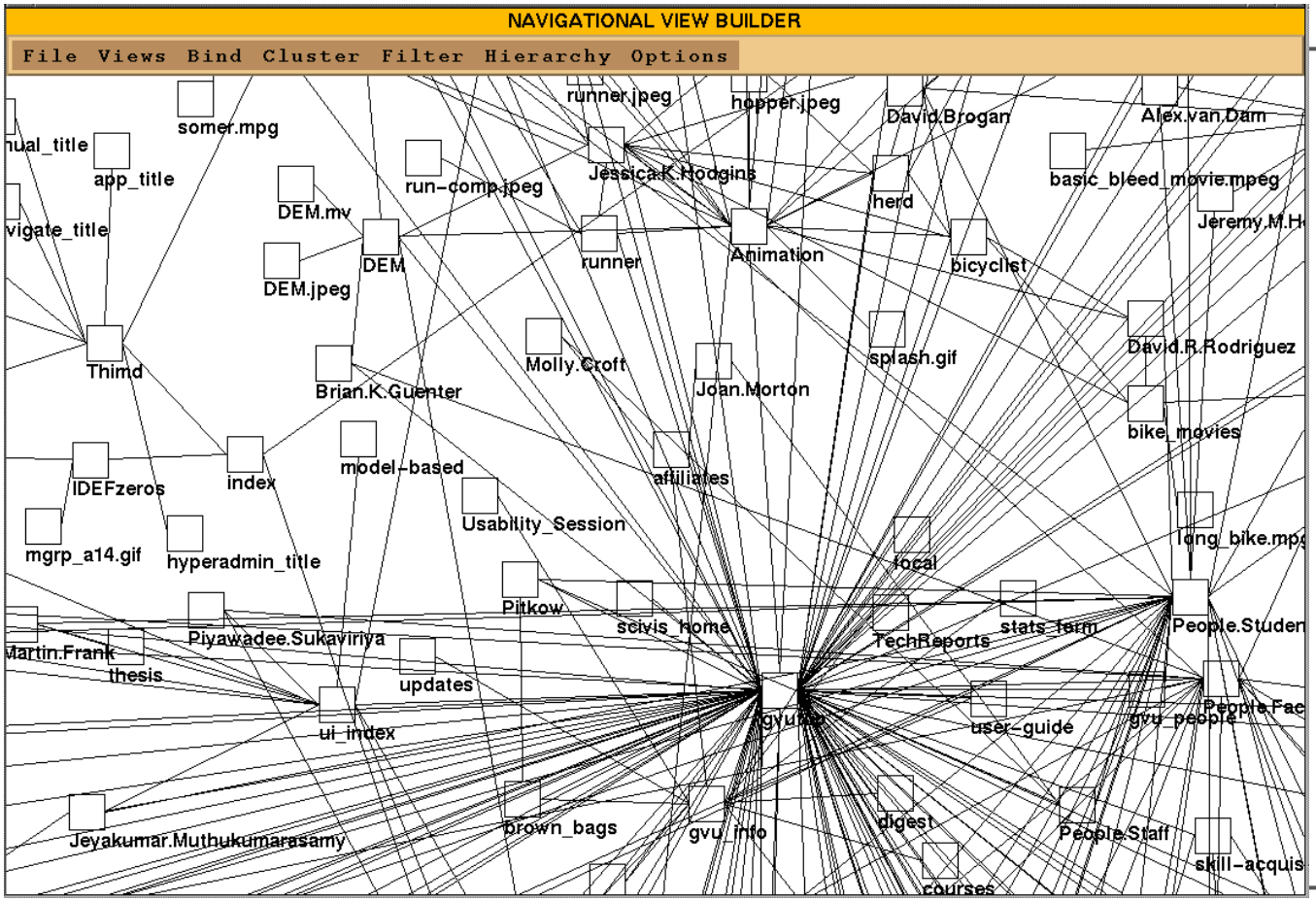
Graph Layout

- How to position the nodes and edges?
  - Avoid clutter
  - Maintain appropriate relations

- The **primary** concern with networks
  - while inheriting other issues such as color, size, etc

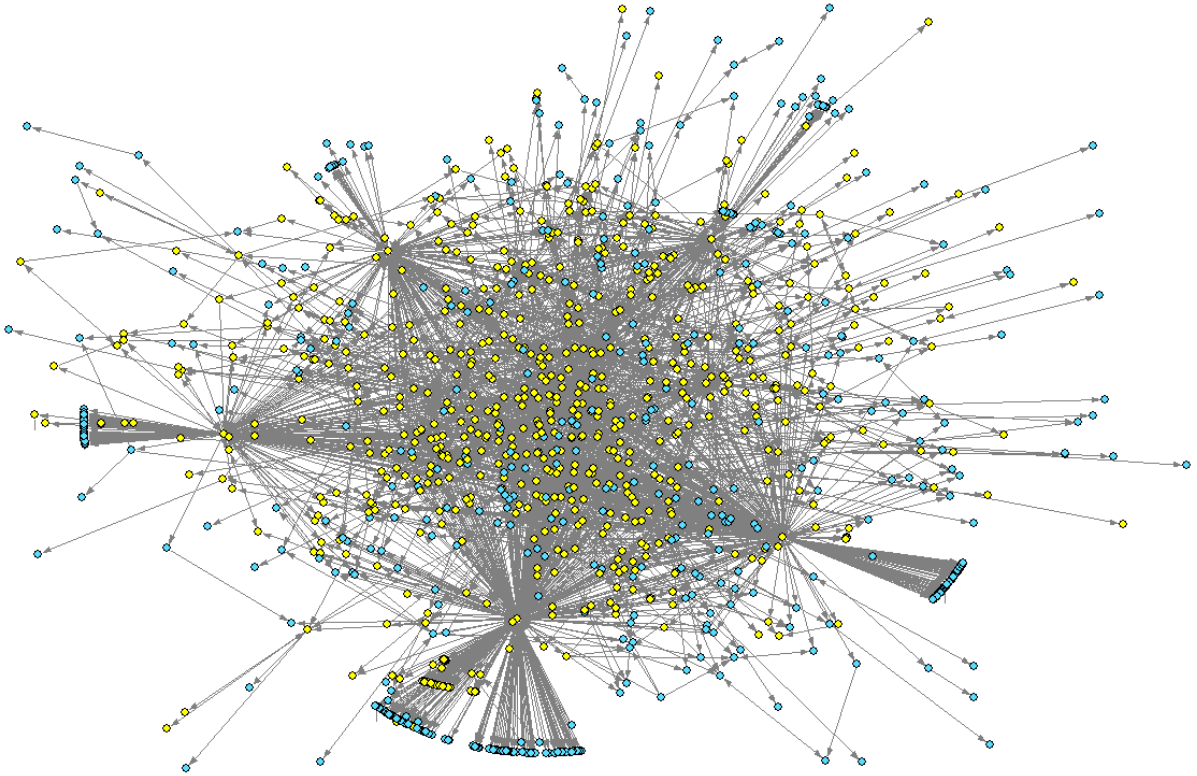
- The topic of the Graph Drawing conference (as well as numerous InfoVis papers) and even multiple books.

Graphs: the problem



- YUK
- .....

The Hairball Problem

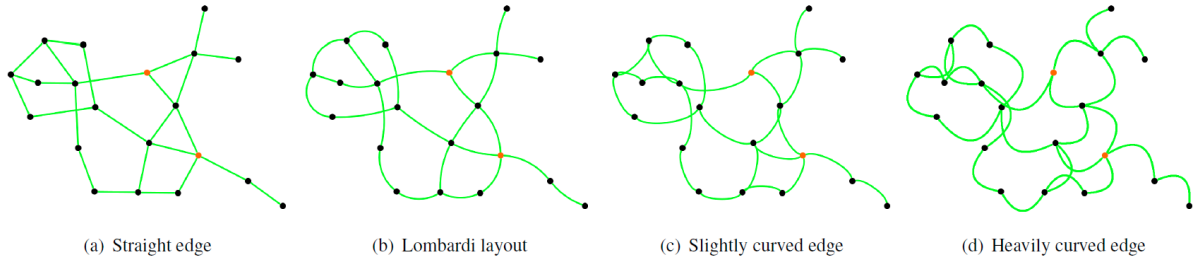


Vertex/Node choices

- Colour
- Size
- Label
- Form/Shape
- **Location**

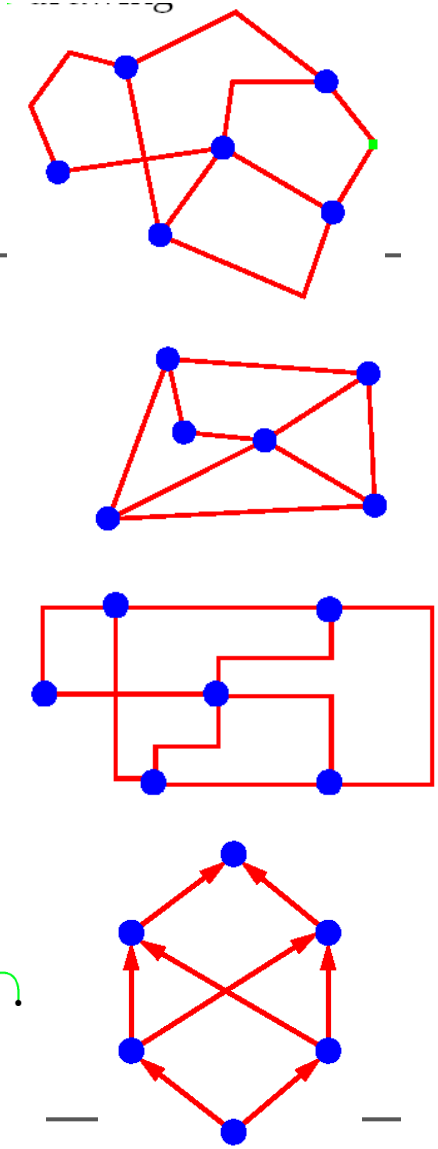
Edge options

- Colour
- Thickness
- Label
- Form
  - Polyline, straight, orthogonal, grid, curved, planar up/down …

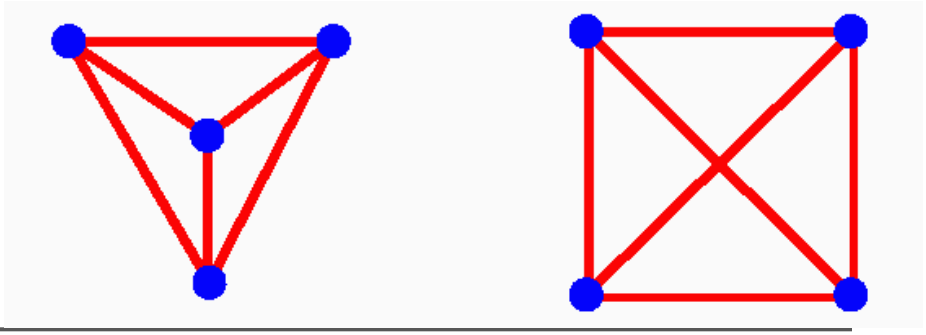


(a) Straight edge (b) Lombardi layout (c) Slightly curved edge (d) Heavily curved edge

Fig. 1. Examples of edge types used in the cured edge study.



Aesthetic constraints



- Minimize link crossings
- Minimize link lengths
- Area – minimize towards efficiency
- Minimize link bends
- uniform edge lengths – minimise variances
- Minimize lomgest edge

- Mathematically difficult to do everything !

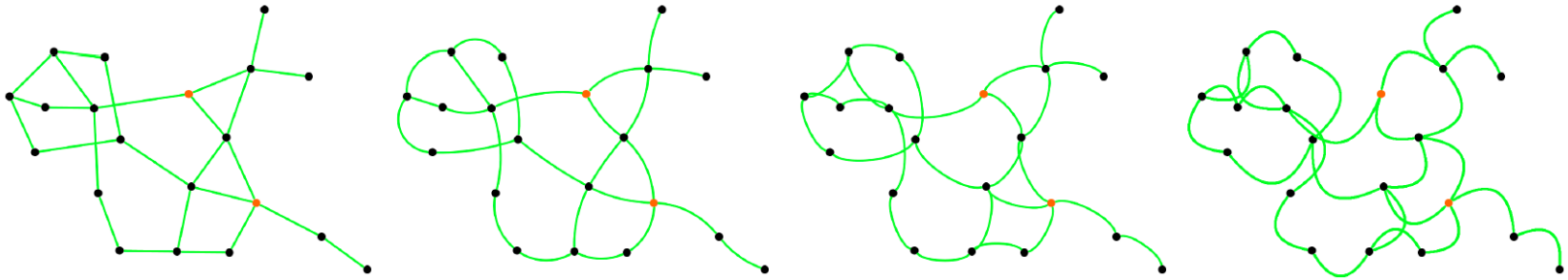
- Often unsuitable for interactive visualisation
  - Approximation algorithms very complex
  - Unless you only need to compute layout once
- Precompute layout, or compute once at the beginning of an application then support interaction

Which is most important?

- Various studies examined which of the aesthetic factors matter most and/or what kinds of layout/vis techniques look best
  - Purchase, Graph Drawing ’97
  - Ware et al, Info Vis 1(2)
  - Ghoniem et al, Info Vis 4(2)
  - van Ham & Rogowitz, TVCG ’08
- Results mixed: Edge crossings do seem important
- BUT
- A recent study (Xu et al 2012) found edge crossings less important than path distance and smoothness
- Curved lines may be preferred but performance is better with straight ..

Aesthetics are not yet defined fully [Xu et al. 2012]

- Lombardi layout a reasonable approach,
  - Uses curved lines only when straight lines are not optimal



(a) Straight edge (b) Lombardi layout (c) Slightly curved edge (d) Heavily curved edge

Fig. 1. Examples of edge types used in the cured edge study.

Kai Xu, Chris Rooney, Peter Passmore, Dong-Han Ham, and Phong H. Nguyen. A User Study on Curved Edges in Graph Visualization , InfoVis 2012.

Shneiderman's NetViz Nirvana

1. Every node is visible
2. For every node you can count its degree
3. For every link you can follow it from source to destination
4. Clusters and outliers are identifiable

- But – what do people actually want to do with graph visualizations??
- Recall the basic vis tasks of [Amar & Stasko]
  - Lecture 5B
- 4 additional types of graph tasks

Graph Visualization Tasks [Stasko]

1. Topology-based tasks
   - Adjacency (find the set of adjacent nodes)
   - Accessibility (find the set of nodes accessible to a node given a limit of hops)
   - Common connection (given nodes, find the set of nodes connected to all members)
   - Connectivity
     - Find shortest path
     - Identify bridges

Graph Visualization Tasks [Stasko]

2. Attribute-based tasks
   - Nodes(find the set of nodes with a specific attribute)
   - Edges (Given a node, find the nodes connected only by certain kinds of edges)

Graph Visualization Tasks [Stasko]

3. Browsing tasks
   - Follow a path
   - Revisit a previously visited node or path
   - Explore different paths to same end –end connections

4. Overview and summary tasks
   - Compound exploratory task
     - Estimate network size and shape
     - Find patterns
     - Find clusters

Common Layout Approaches

- Force-directed
- Hierarchical
- Circular
- Geographic-based
- Clustered
- Attribute-based
- Matrix
- (this list is not meant to be exhaustive)

Optimization-based layout

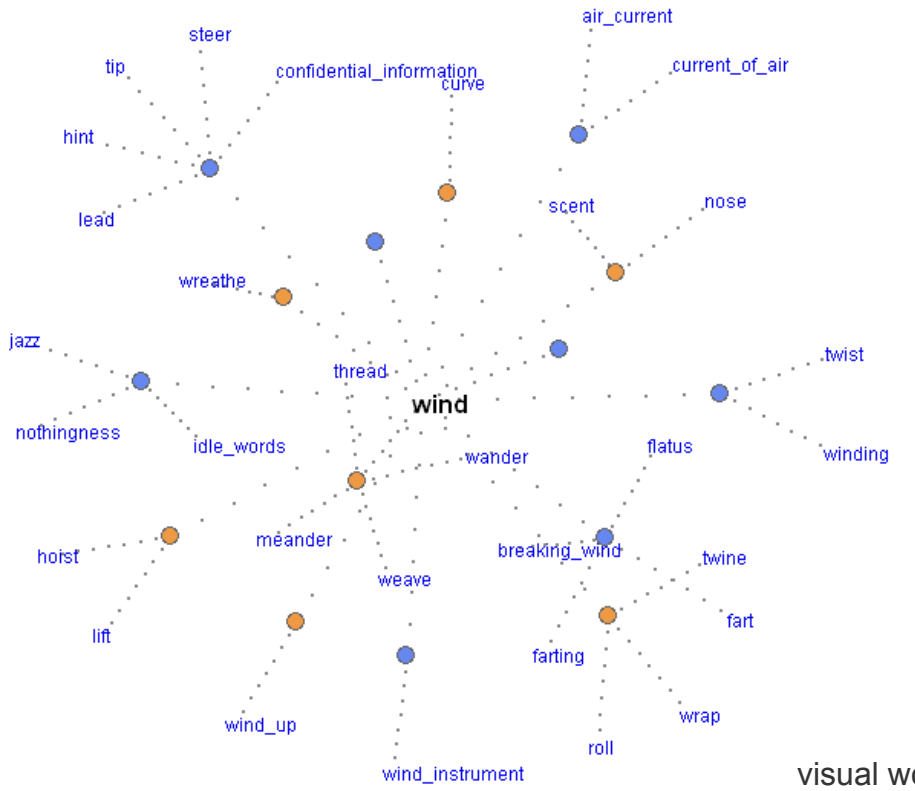
- Specify constraints for layout
  - Series of mathematical equations
  - Hand to “solver” which tries to optimize the constraints
- Examples
  - Minimize edge crossings, line bends, etc
  - Multi-dimensional scaling (preserve multi-dim distance)
  - Force-directed placement (use physics metaphor)
-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

http://bl.ocks.org/benzguo/4370043

Force – directed layout

- Spring model (common)
  - Edges – Springs (gravity attraction)
  - Vertices – Charged particles (repulsion)
- Equations for forces
- Iteratively recalculate to update positions of vertices
- Seeking local minimum of energy
  - Sum of forces on each node is zero

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.

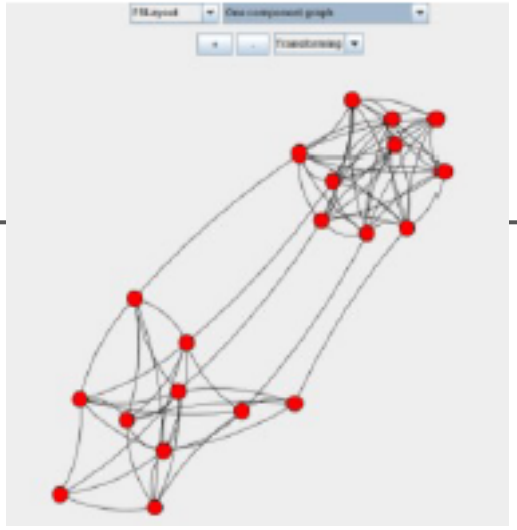
Special techniques to dampen “jitter”.

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

visuwords http://www.visuwords.com/

Variants of force layouts

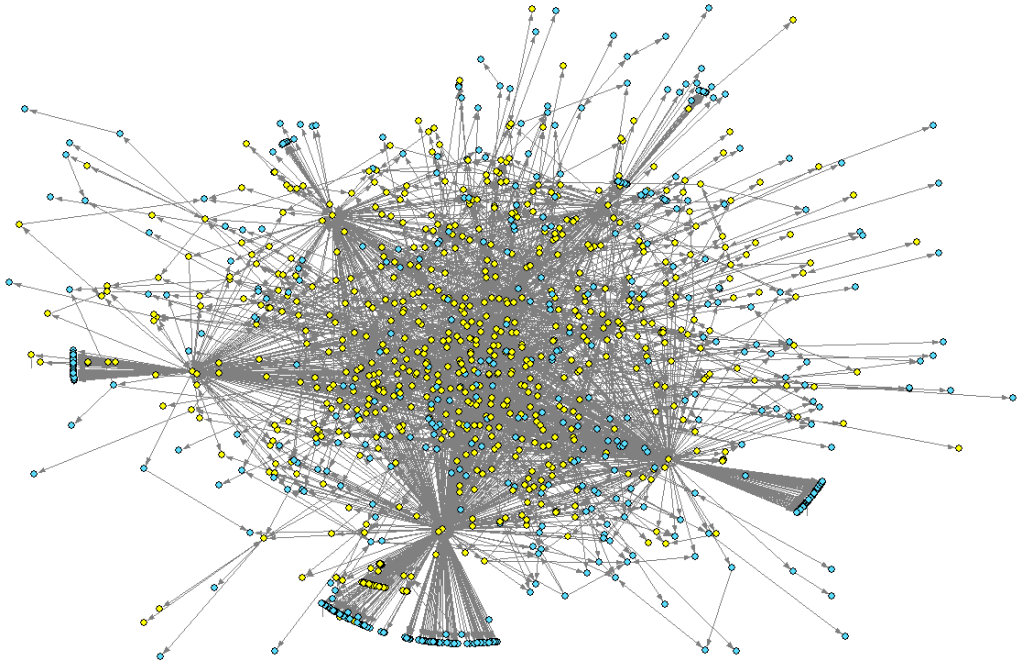
- Address issues of randomness
- Use calculations to control how layout moves with changes in nodes
  - Temperature to model how much nodes move
- Demos at JUNG



Issues

Idiom	Force-Directed Placement
What: Data	Network.
How: Encode	Point marks for nodes, connection marks for links.
Why: Tasks	Explore topology, locate paths.
Scale	Nodes: dozens/hundreds. Links: hundreds. Node/link density: $L < 4N$

- Popular and easy
- Nondeterministic
  - mall changes in input can have large changes in output
  - randomness, no constraints on maintaining geometric proximity
- Jitter
- Scalability: the “hairball” problem

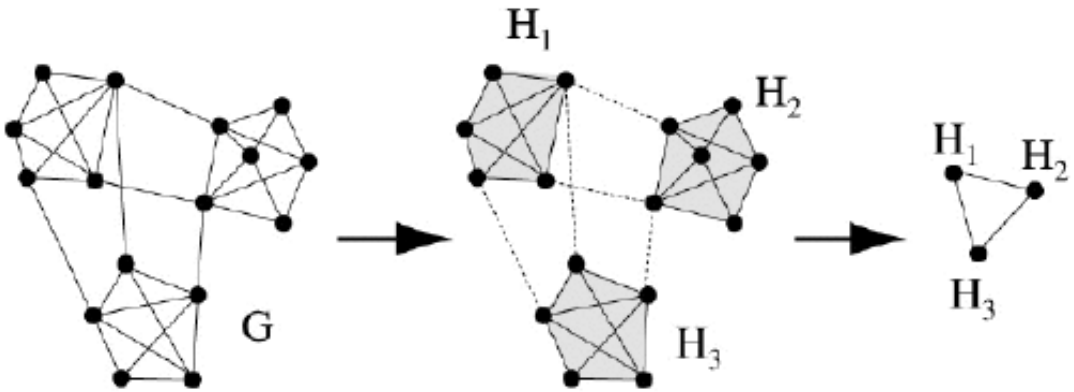


Small-world networks

- high clustering, small path length
  - vs. random uniform distribution
- examples
  - social networks, movie actors, Web, ...
- multiscale small-world networks
  - exploit these properties for better layout

Small world coarsening

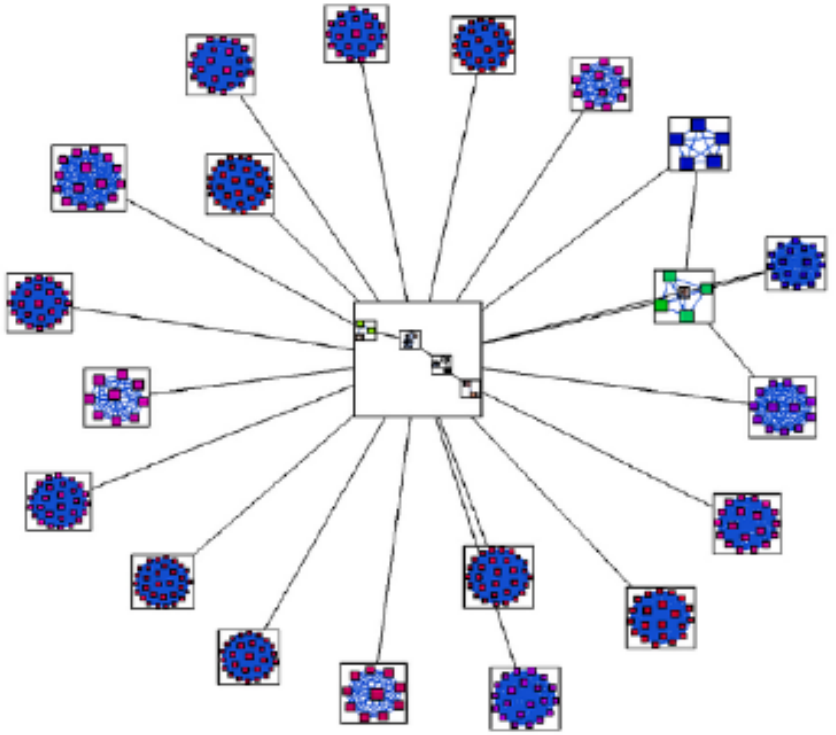
- remove low-strength edges
- maximal disconnected subgraphs
- *Quotient graph*: subgraph = higher-level node



Auber et al. Multiscale Visualization of Small World Networks. Proc. InfoVis 2003, p 75-81.]

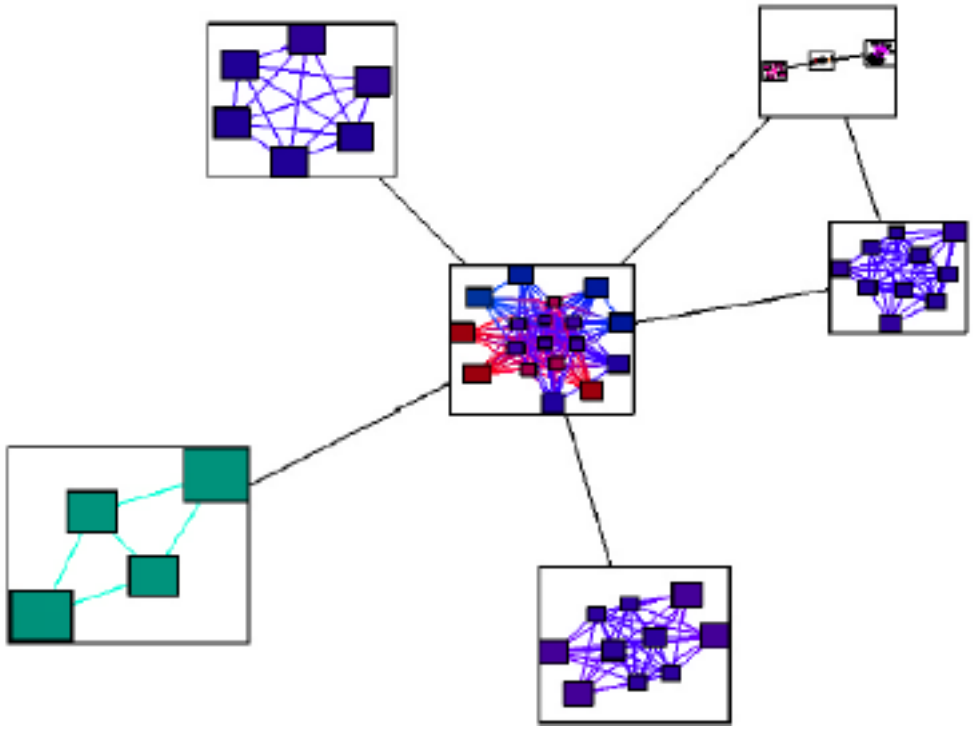
Nested Quotient Graphs

- Visual encoding:
  - Subgraph(s) laid out in metanode



Nested Quotient Graphs

- Pro: very evocative of structure
- Con: doesn't scale past 3 levels



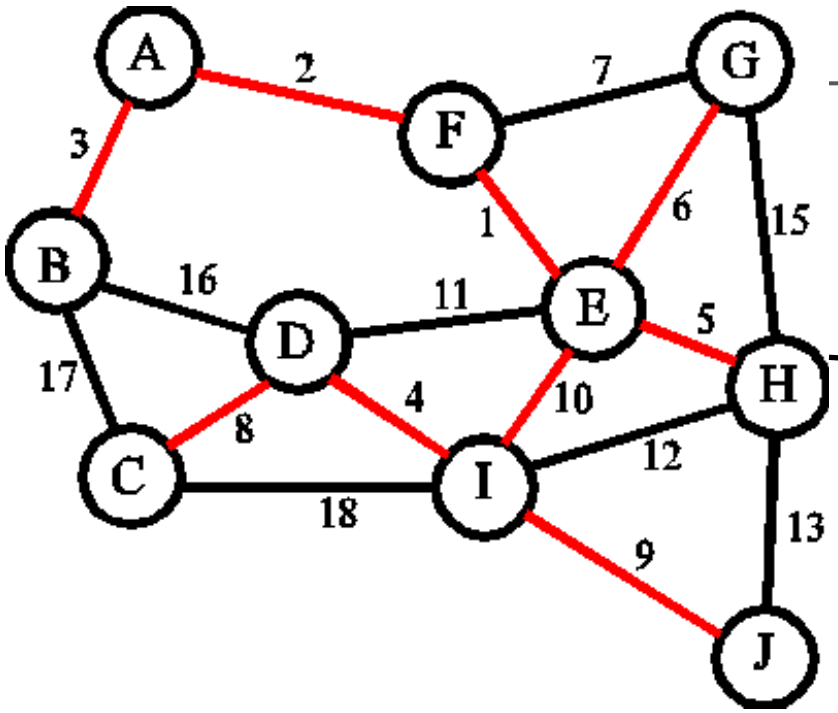
Tree-based graph layout: add hierarchy

- 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

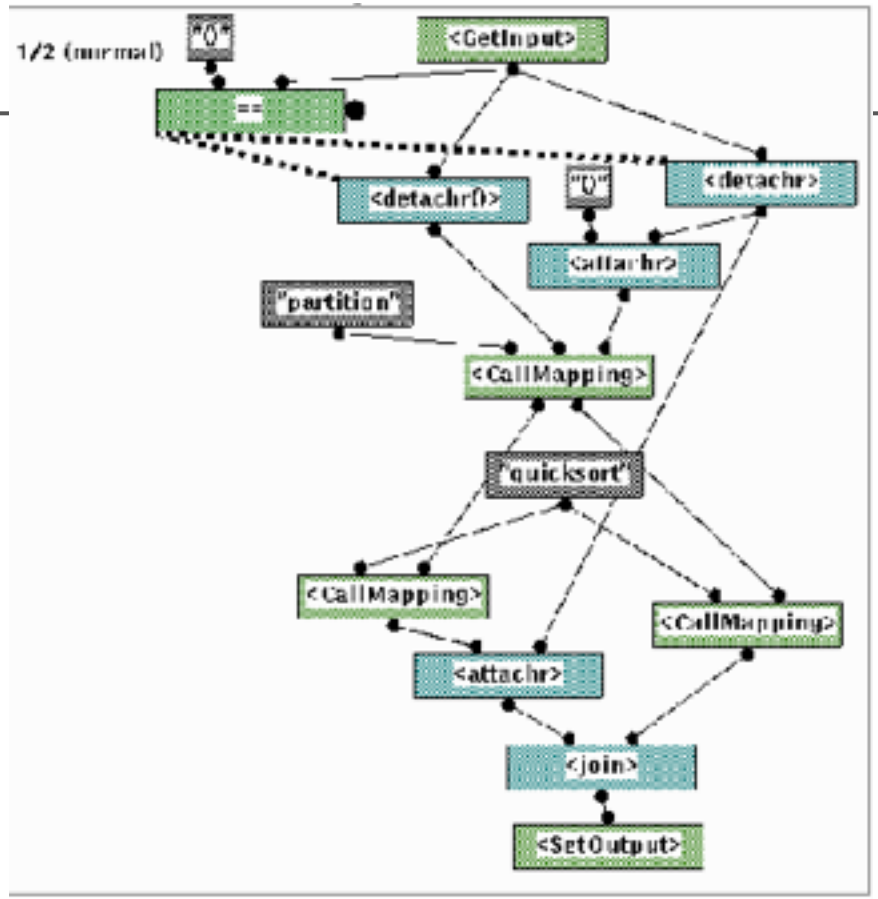
Hierarchical graph layout

- 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

Tree-ify the graph

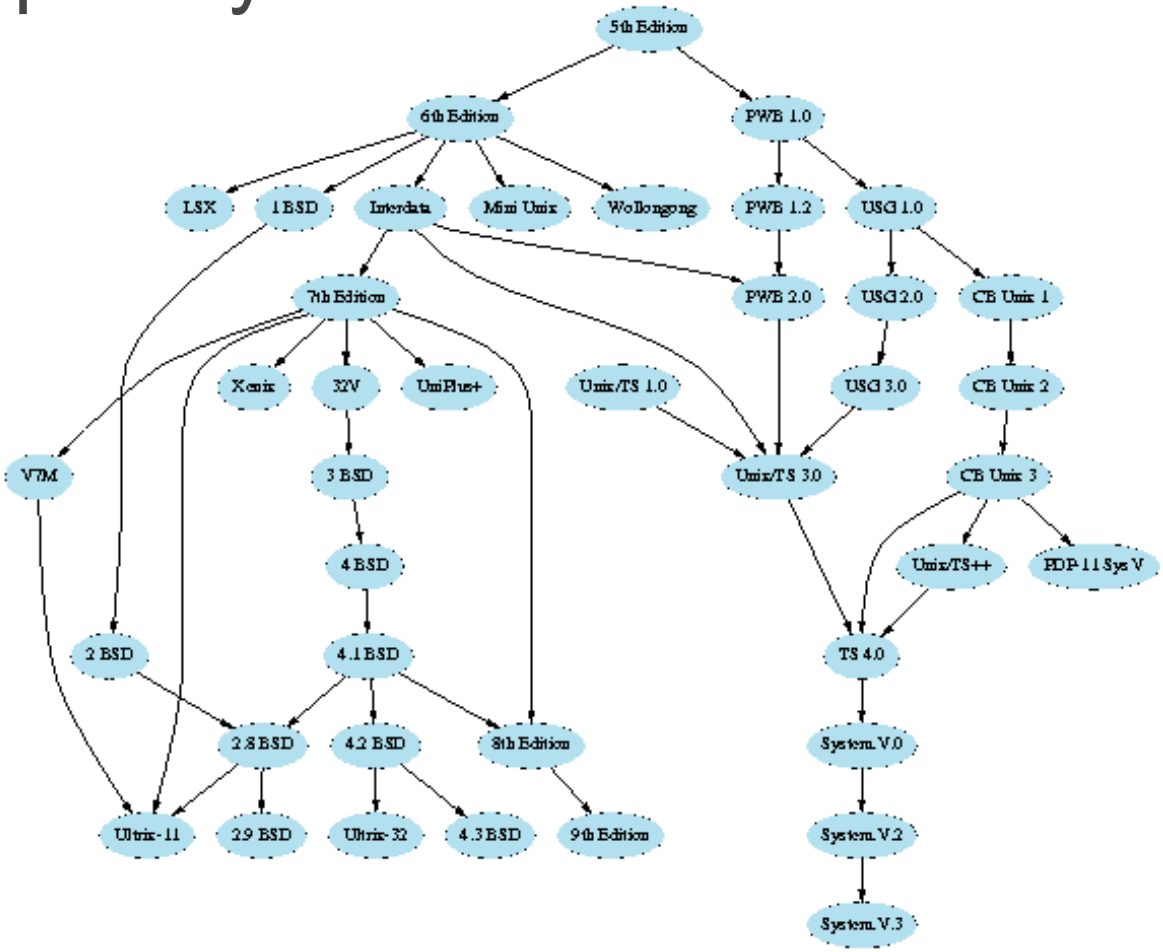


Tree-ify the graph



Hierarchical Graph Layout

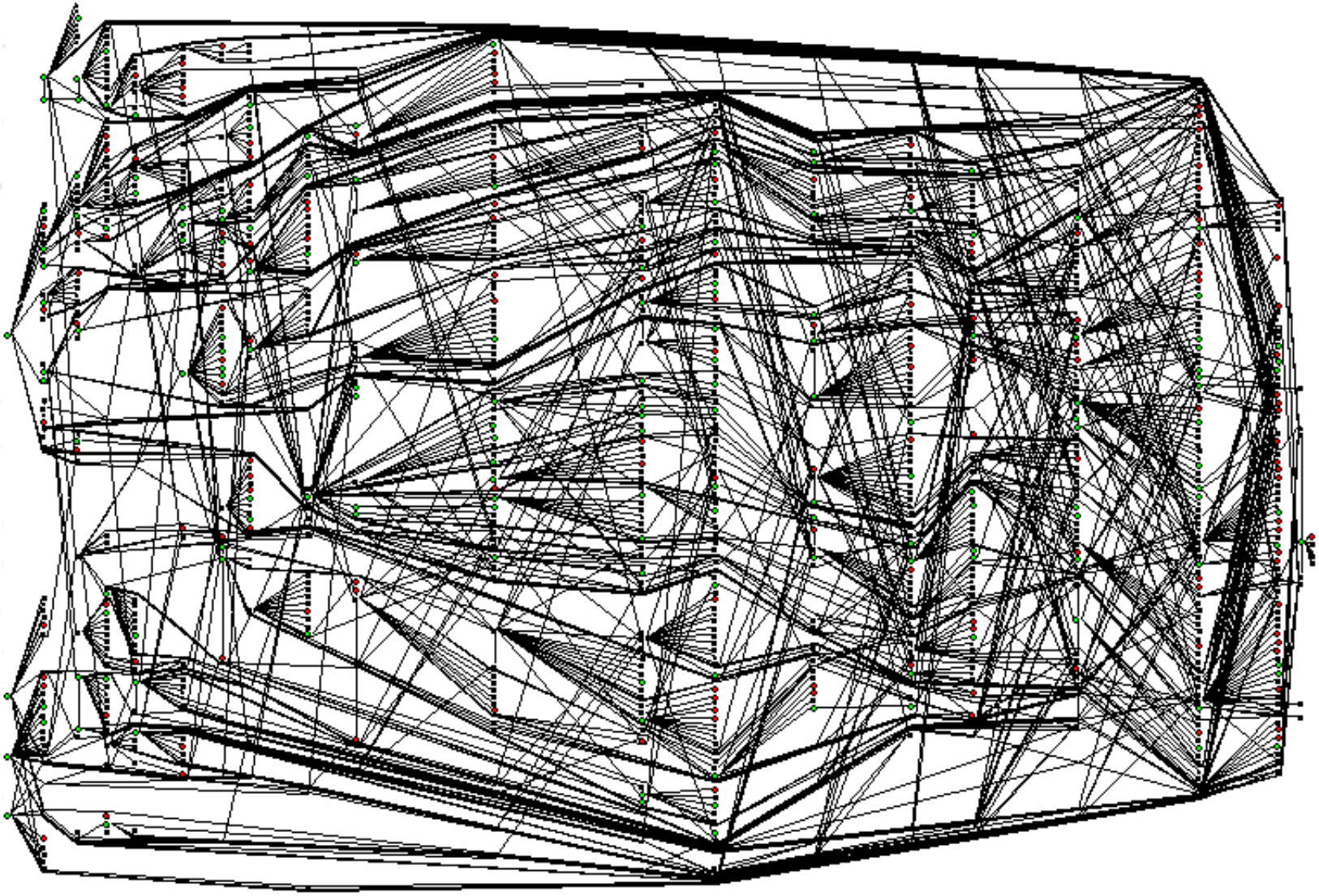
- Evolution of the UNIX operating system
- Hierarchical layering based on descent



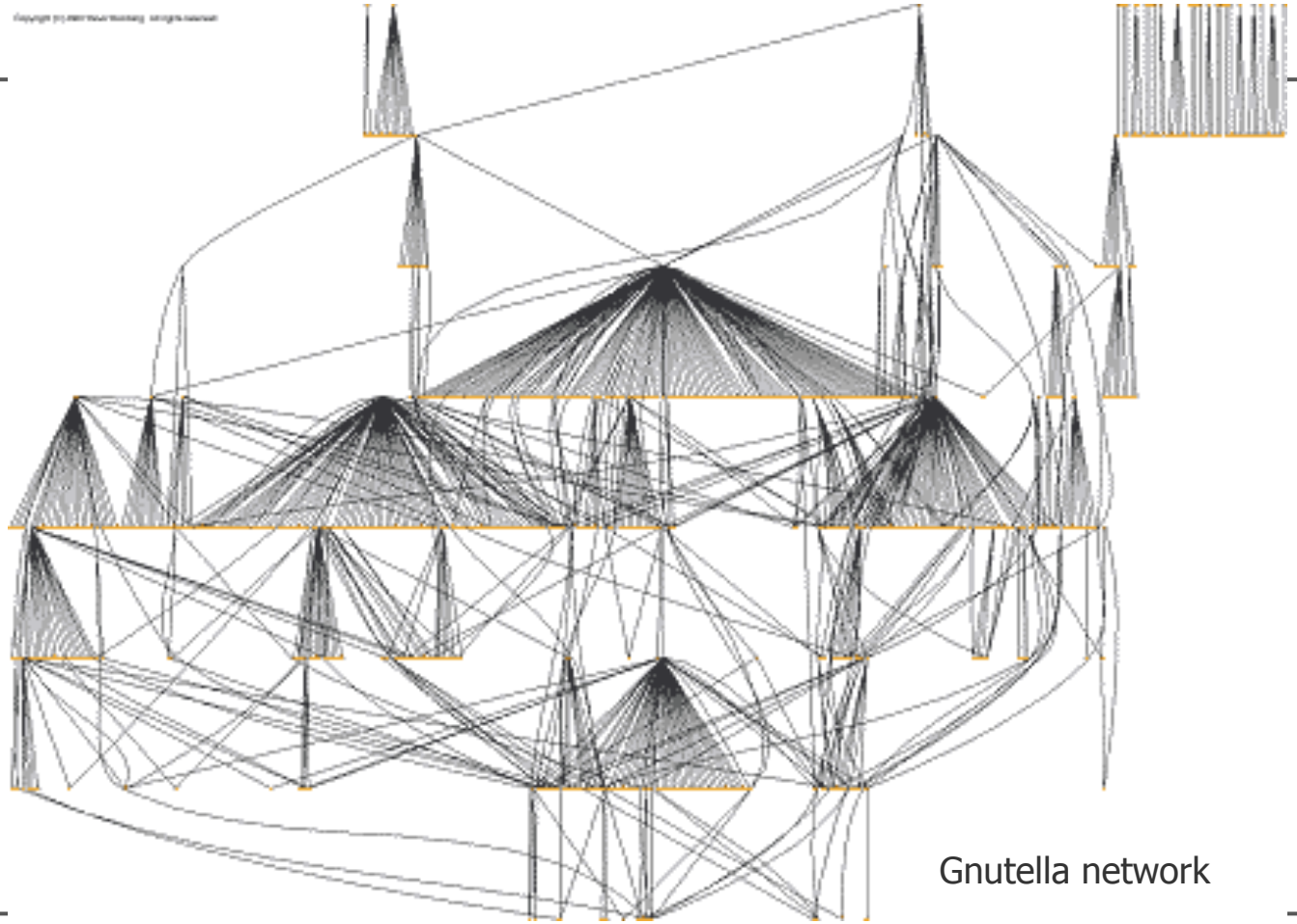
Partial Map of Gnutella Network - 7/27/00

Clip2 Distributed Search Services

http://dss.clip2.com

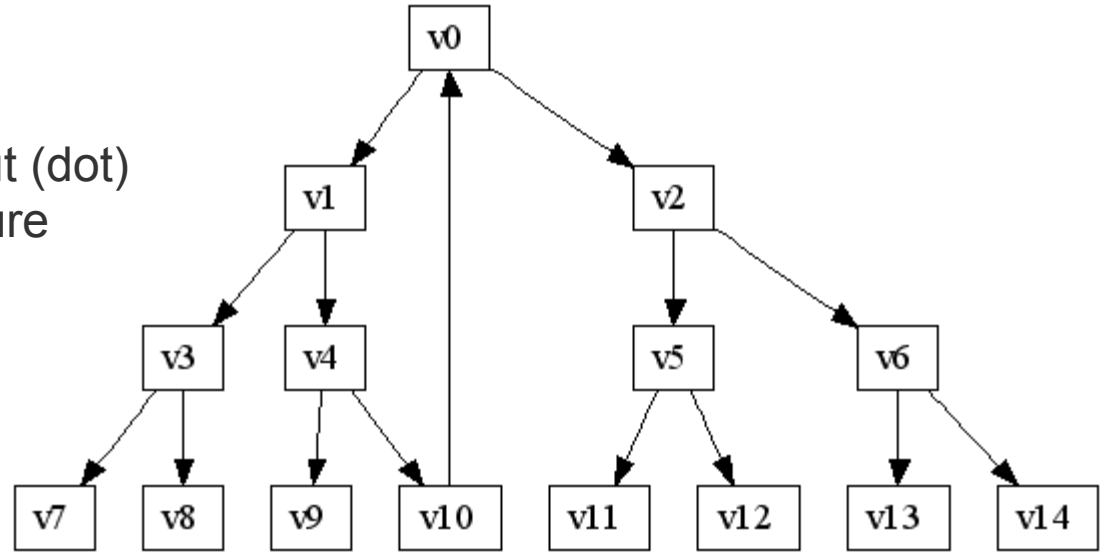


Hierarchical graph layout

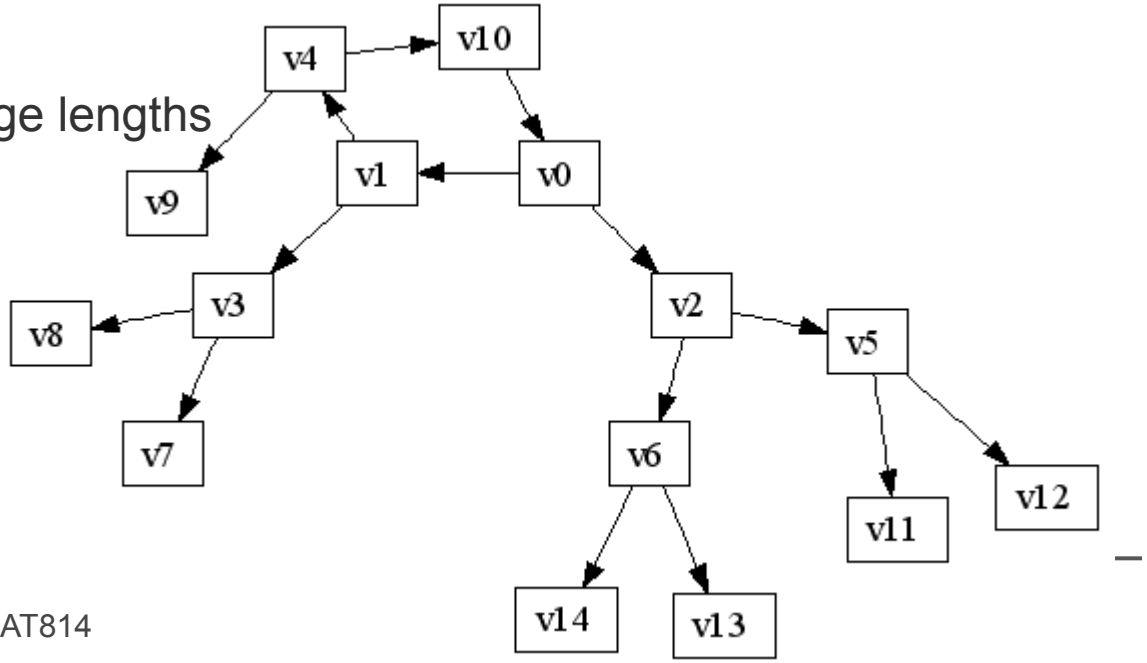


Gnutella network

Typical Sugiyama layout (dot)
- preserves tree structure



Alternative method
- preserves uniform edge lengths



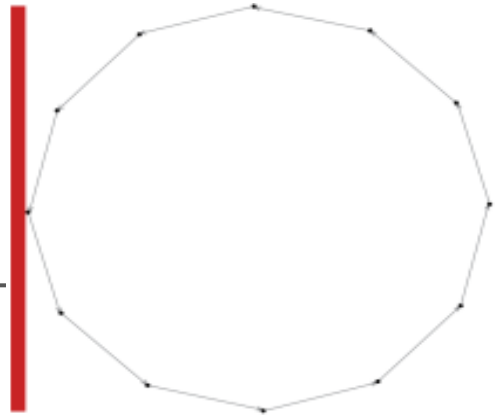
slide borrowed from Tim Dwyer and Marti Hearst

Examples

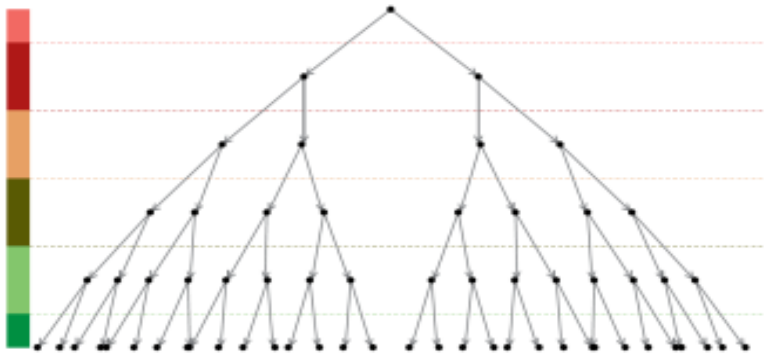
Layered Graph Layouts

Directed Acyclic Graphs

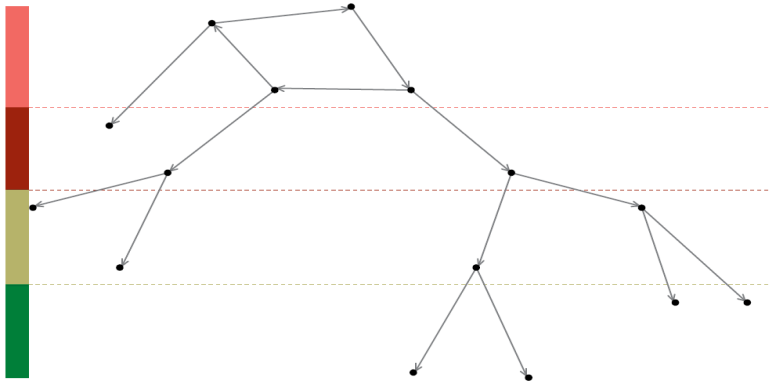
Identify “ bands”



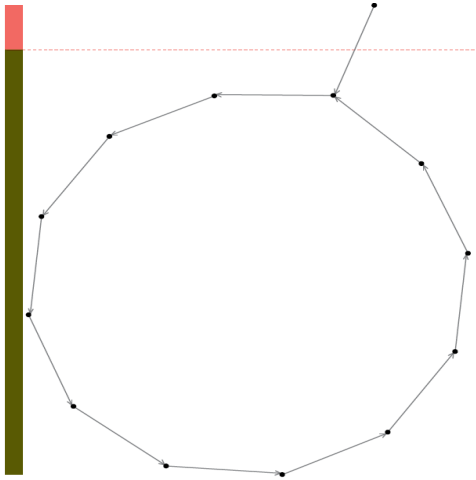
(a) 1 band



(b) 6 bands



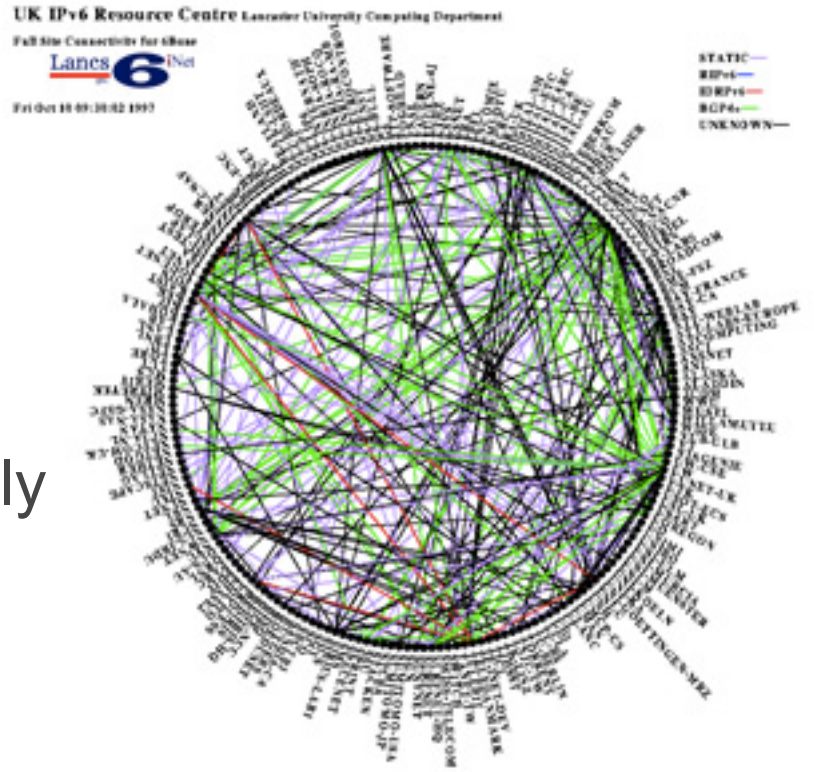
(c) 4 bands

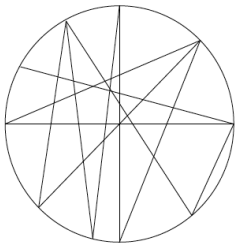


(d) 2 bands

Radial layouts

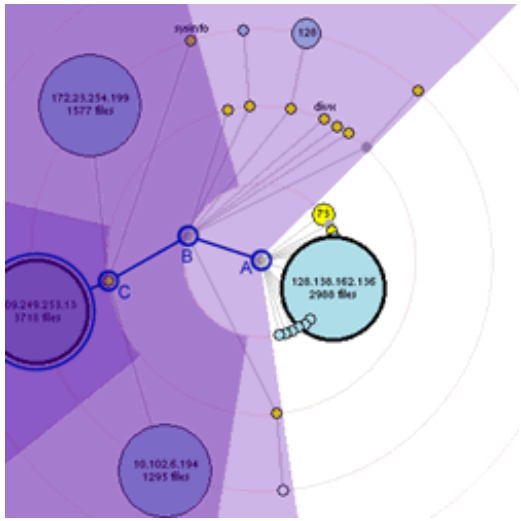
- Very simple
- Plot nodes around the circumference
- Edges connect them
  - Minimise crossings
- Can get very complex quickly

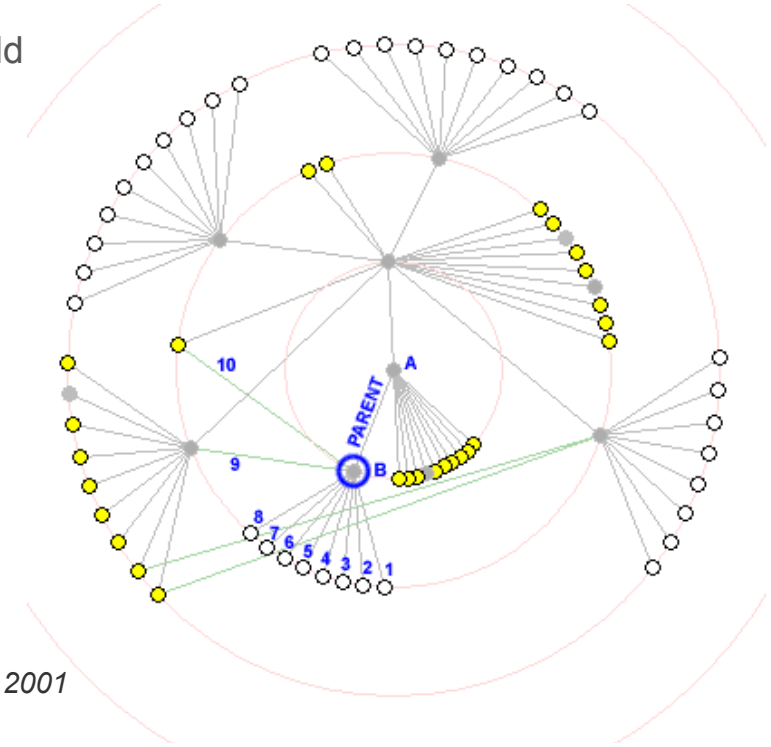




Radial Layout

- Data abstraction: Hierarchical network
- encoding technique: radial layout
- interaction technique: animated transitions from old new layout



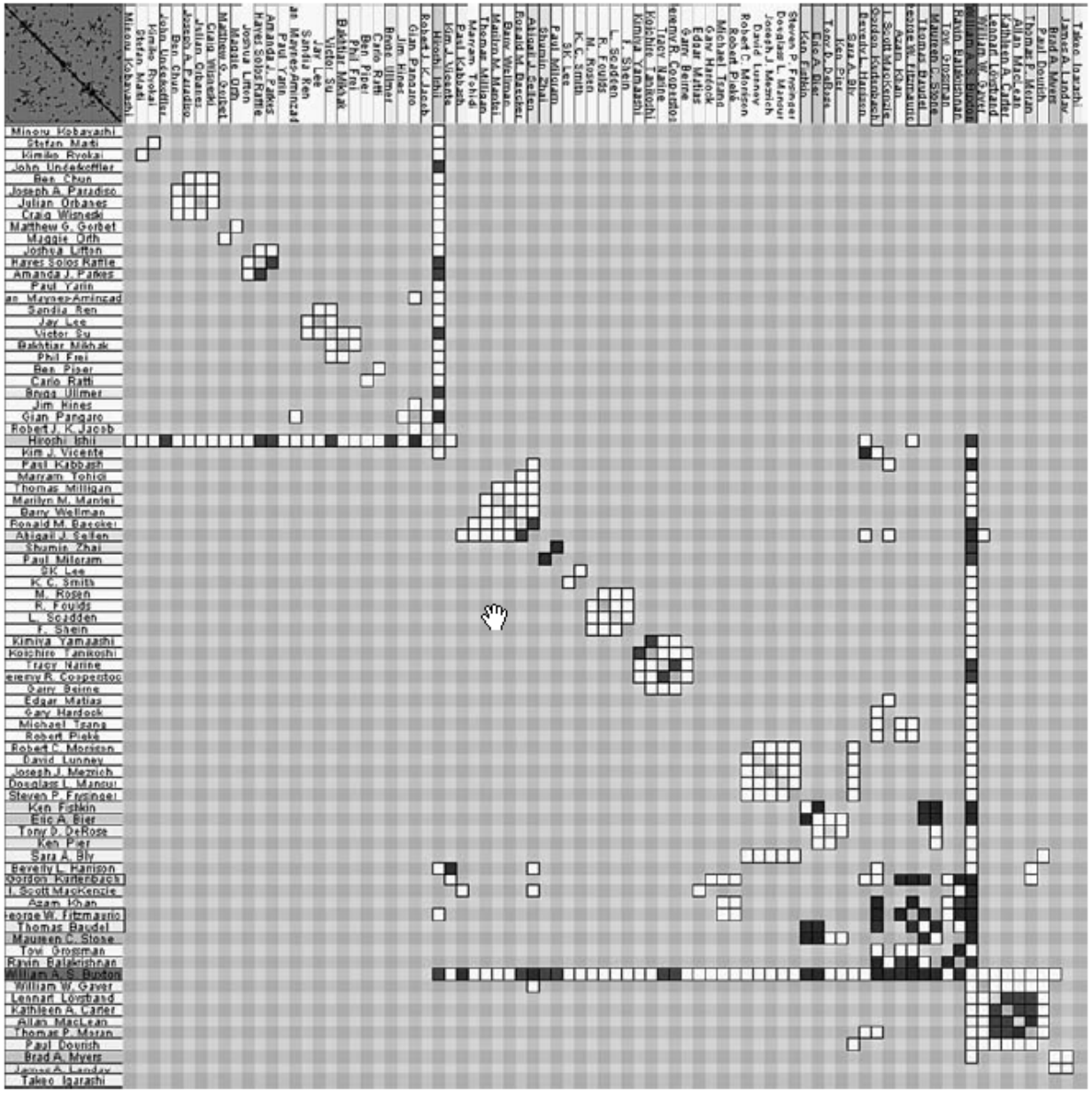


Animated Exploration of Graphs with Radial Layout, Yee et al., 2001

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

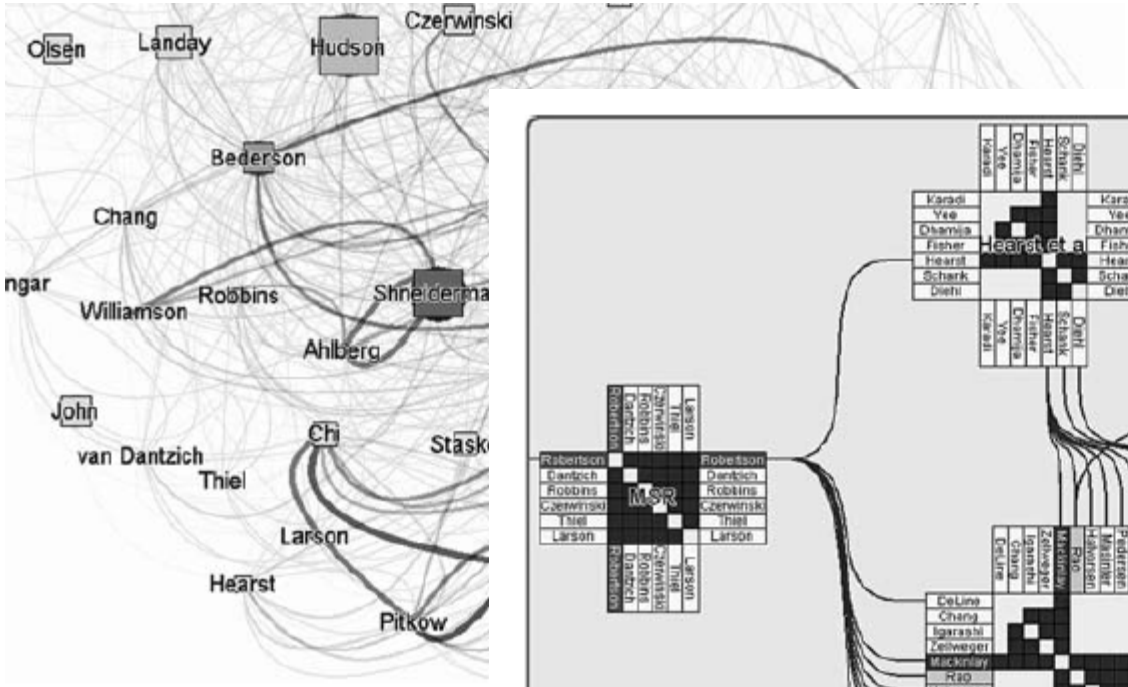


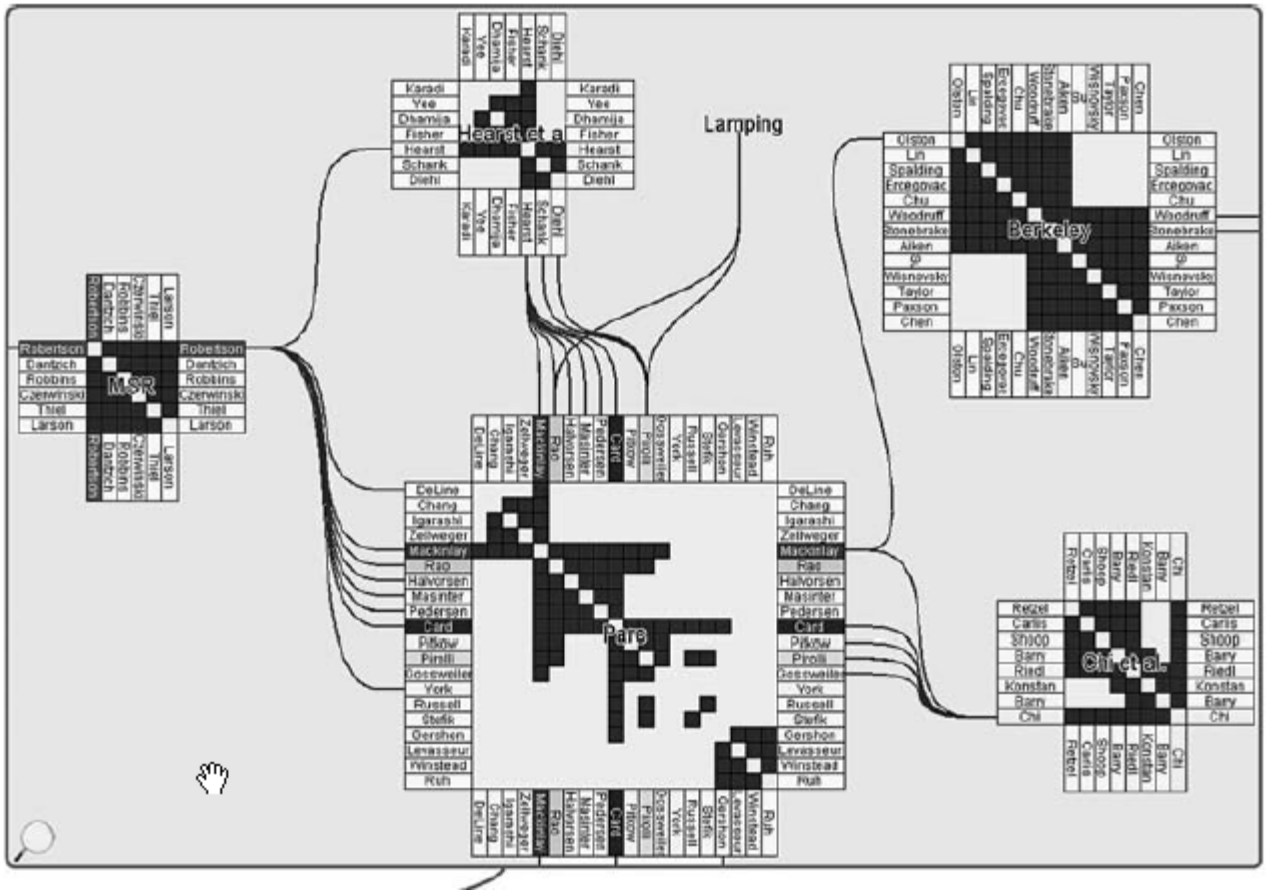
http://www.informaworld.com/s

20 Years of Four HCI Conferences: A Visual Exploration
Henry et al. IJHCI 2007

Matrices with Submatrices

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



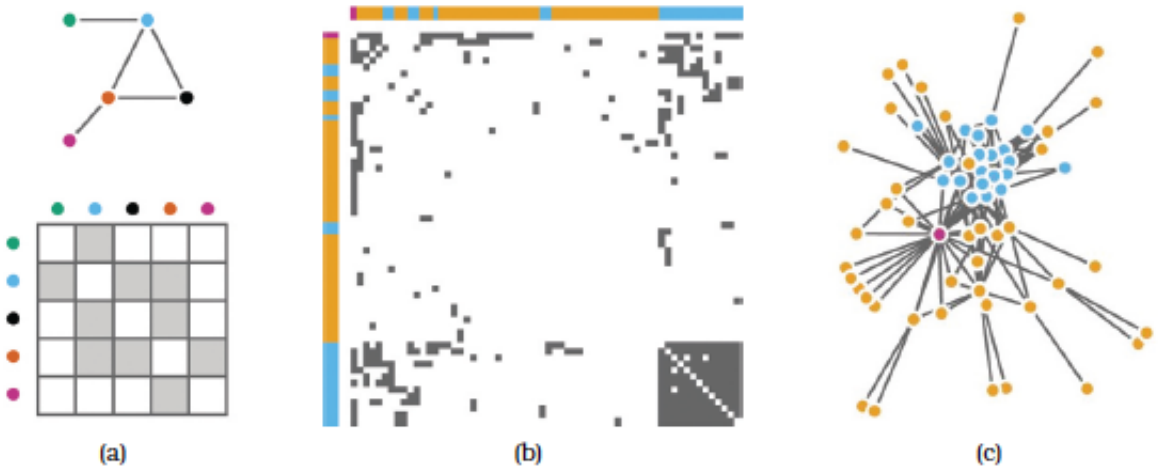


Node-link graphs

Idiom	**Multilevel Force-Directed Placement (sfdp)**
What: Data	Network.
What: Derived	Cluster hierarchy atop original network.
What: Encode	Point marks for nodes, connection marks for links.
Why: Tasks	Explore topology, locate paths and clusters.
Scale	Nodes: 1000–10,000. Links: 1000–10,000. Node/link density: $L < 4N$.

- Not very scaleable
- Re-ordering can disorient
- Good for smaller networks

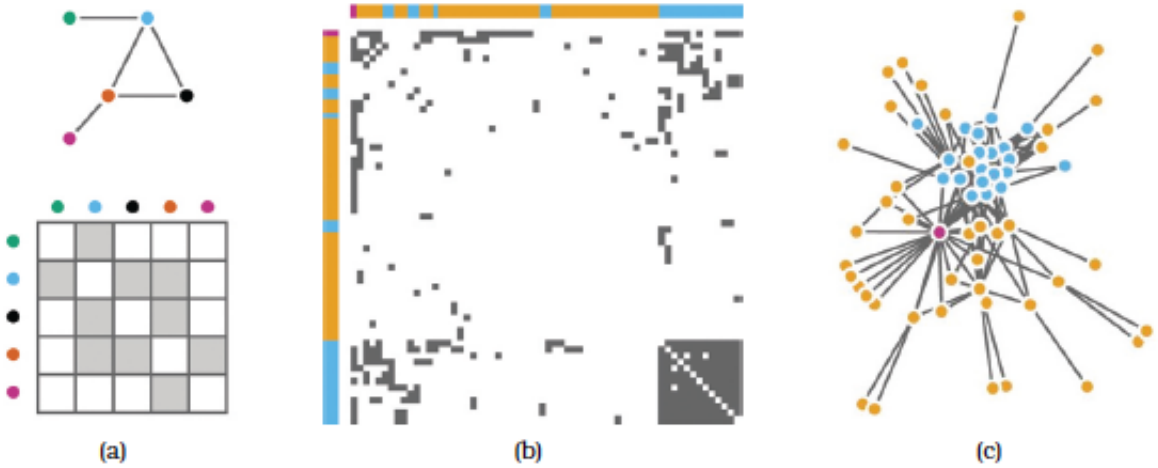
- Good for topological tasks
- Perceptually intuitive
- Perceptually unstable (some)



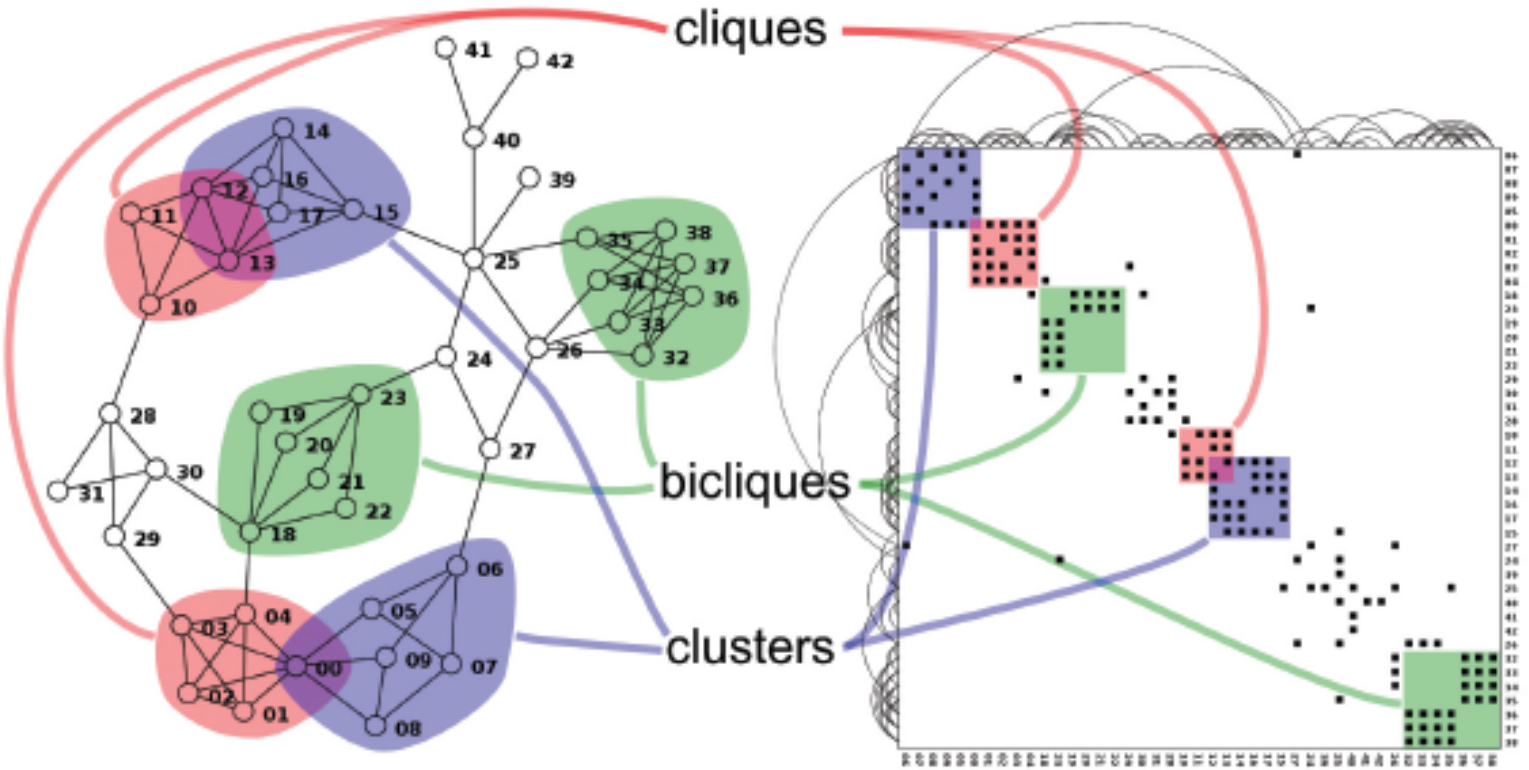
Matrices

Idiom	Adjacency Matrix View
What: Data	Network.
What: Derived	Table: network nodes as keys, link status between two nodes as values.
How: Encode	Area marks in 2D matrix alignment.
Scale	Nodes: 1000. Links: one milllion.

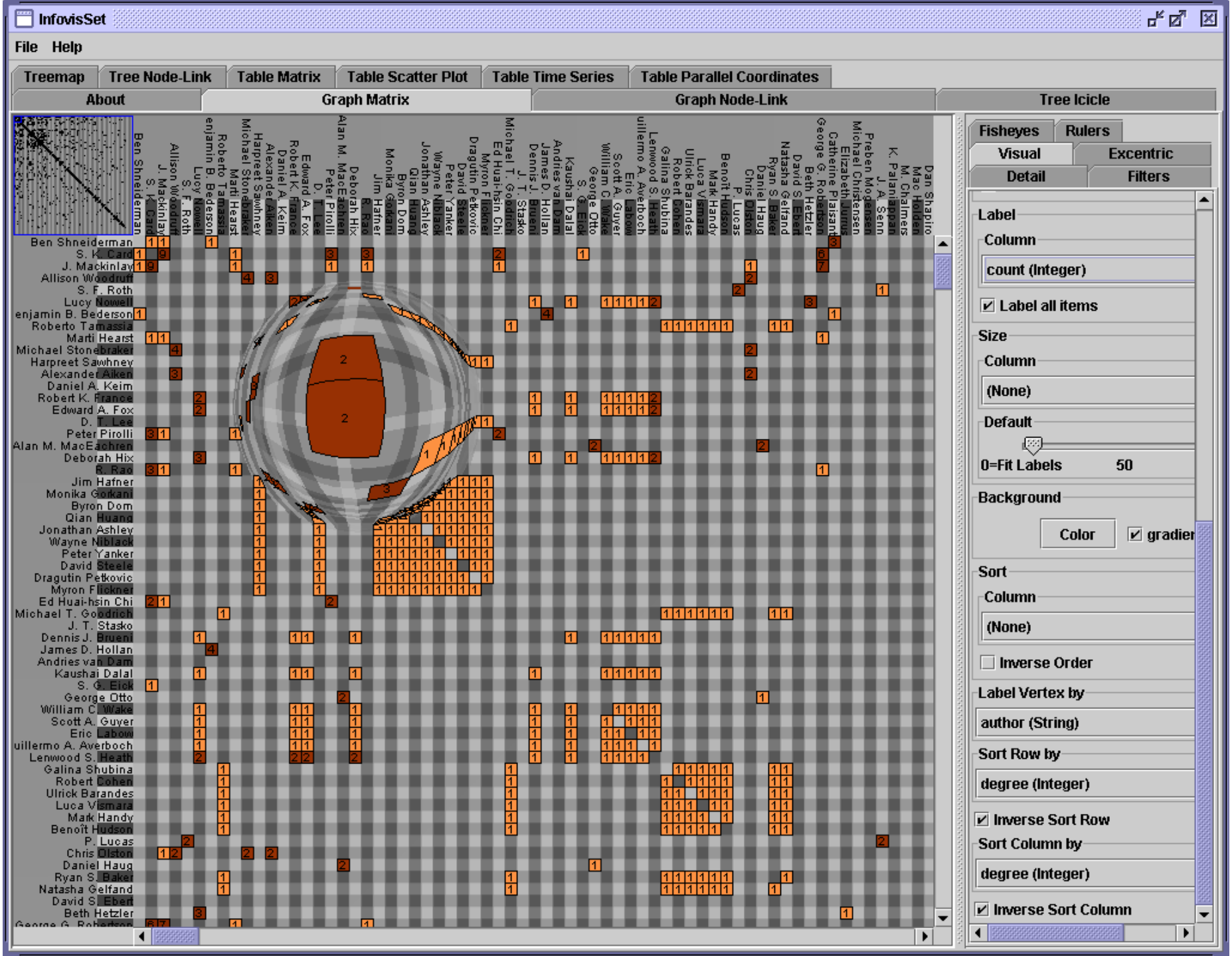
- Very scaleable
- Good for link density
- Easily support re-ordering
- Fast node-lookup
- Perceptually stable
- Hard to learn



The two views can co-exist

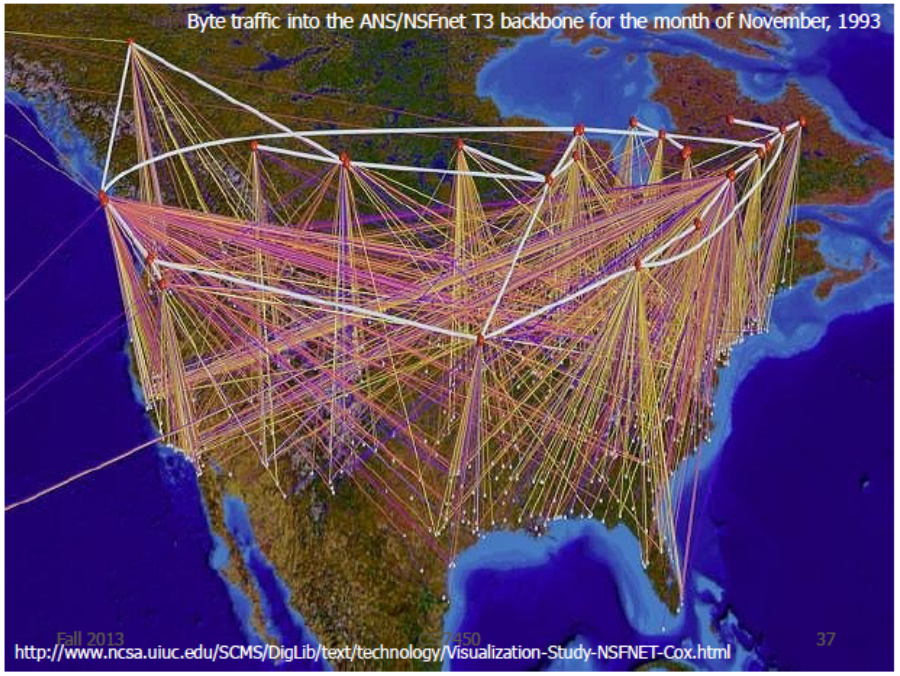


Michael J. McGuffin. "Simple Algorithms for Network Visualization: A Tutorial." Tsinghua Science and Technology (Special Issue on Visualization and Computer Graphics) 17:4 (2012), 383–398


InfovisSet
File Help
Treemap
Tree Node-Link
Table Matrix
Table Scatter Plot
Table Time Series
Table Parallel Coordinates
About
Graph Matrix
Graph Node-Link
Tree Icicle
Ben Shneiderman
S. K. Card
J. Mackinlay
Allison Woodruff
S. F. Roth
Lucy Nowell
enjamin B. Bederson
Roberto Tamassia
Marti Hearst
Michael Stonebraker
Harpreet Sawhney
Alexander Aiken
Daniel A. Keim
Robert K. France
Edward A. Fox
D. T. Lee
Peter Pirolli
Alan M. MacEachren
Deborah Hix
R. Rao
Jim Hafner
Monika Gorkani
Byron Dom
Qian Huang
Jonathan Ashley
Wayne Niblack
Peter Yanker
David Steele
Dragutin Petkovic
Myron Flickner
Ed Huai-hsin Chi
Michael T. Goodrich
J. T. Stasko
Dennis J. Brueni
James D. Hollan
Andries van Dam
Kaushal Dalal
S. G. Eick
George Otto
William C. Wake
Scott A. Guyer
Eric Labow
uillermo A. Averboch
Lenwood S. Heath
Galina Shubina
Robert Cohen
Ulrick Barandes
Luca Vismara
Mark Handy
Benoît Hudson
P. Lucas
Chris Olston
Daniel Haug
Ryan S. Baker
Natasha Gelfand
David S. Ebert
Beth Hetzler
George G. Robertson
Catherine Plaisant
Elizabeth Jurnus
Preben Mogensen
Michael Christensen
K. Palaniappan
J. A. Senn
M. Chalmers
Mac Holden
Dan Shapiro
Fisheyes
Rulers
Visual
Excentric
Detail
Filters
Label
Column
count (Integer)
Label all items
Size
Column
(None)
Default
0=Fit Labels
50
Background
Color
gradier
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

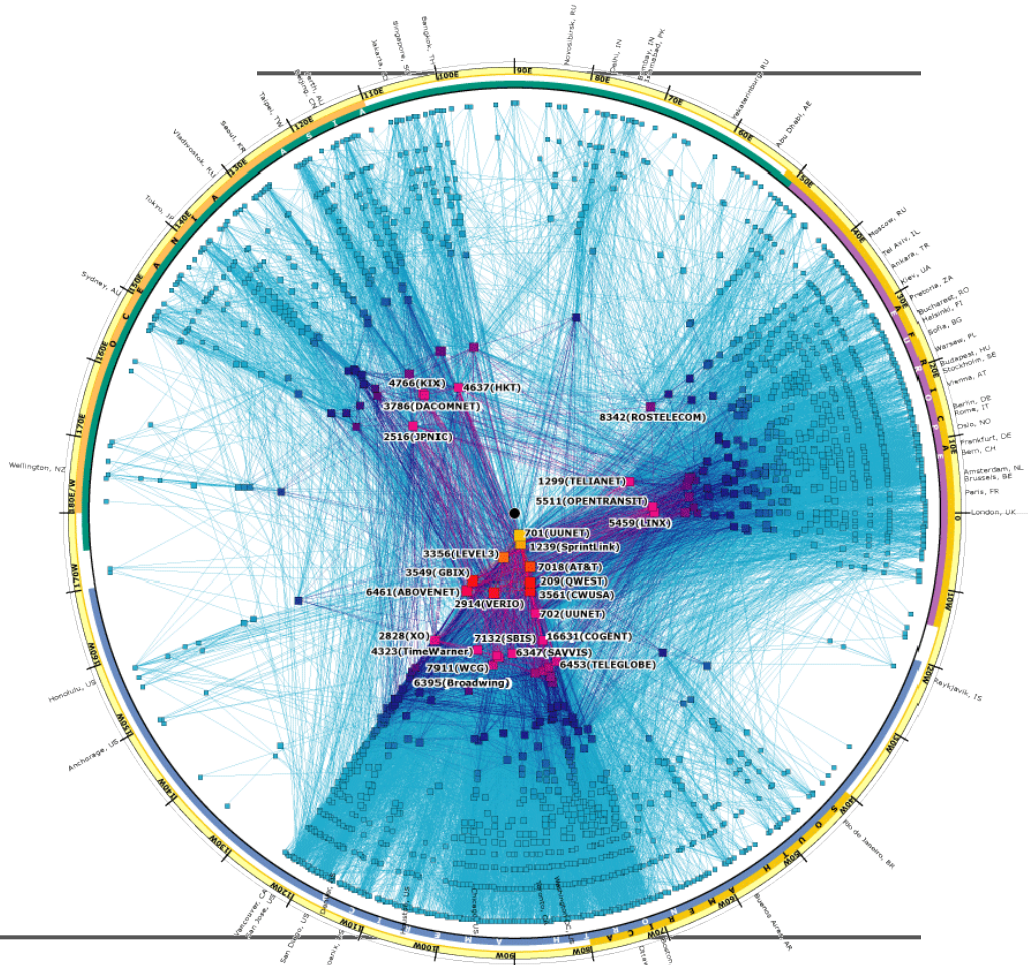
Structurally-Independent Layout

- Ignore the graph structure.
- Base the layout on other attributes of the data
  - 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

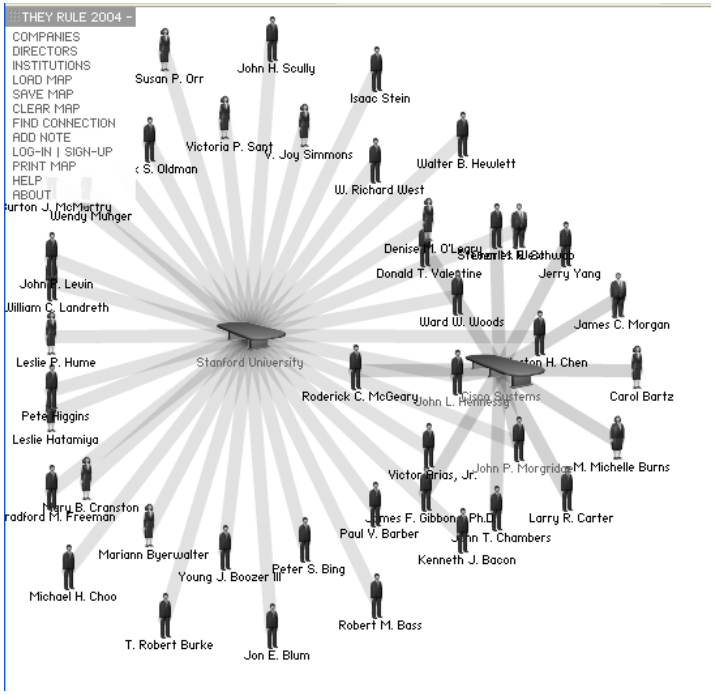


Skitter, www.caida.org

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



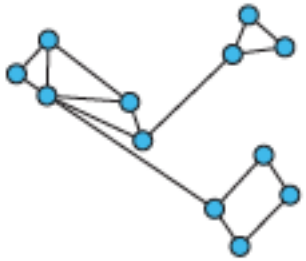
Aggregate/compound methods

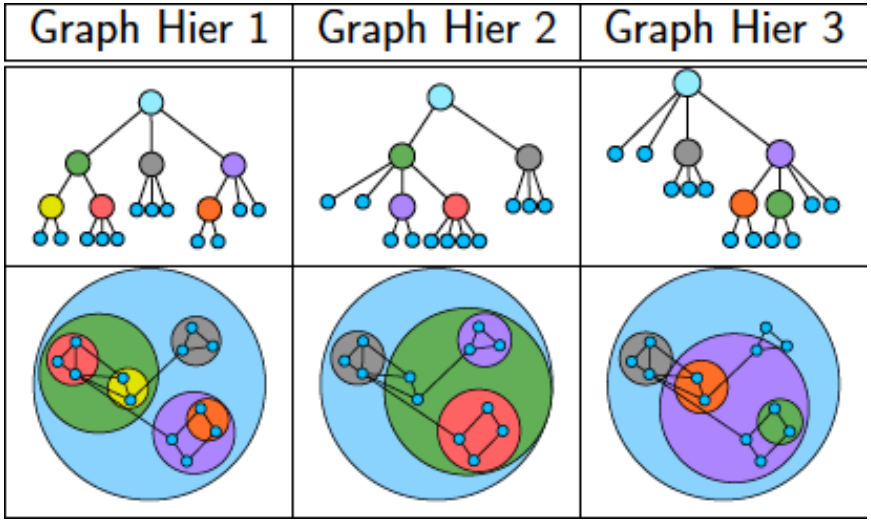
- What about issues of scale?
  - Too many dimensions
  - Too much data!

- Add more sophisticated grouping and attribute cues to graph
- More scalable solutions make use of clustering and hierarchy

Multilevel (compound) graphs

- Data abstraction: create a cluster hierarchy on top of original graph (*coarsening*)





Archambault et al. GrouseFlocks: Steerable Exploration of Graph Hierarchy Space. IEEE Trans. Visualization and Computer Graphics 14(4):900-913 200

Containment and connectivity

- Compound networks can use both marks
- Interaction: expand metanodes to change the “cut”

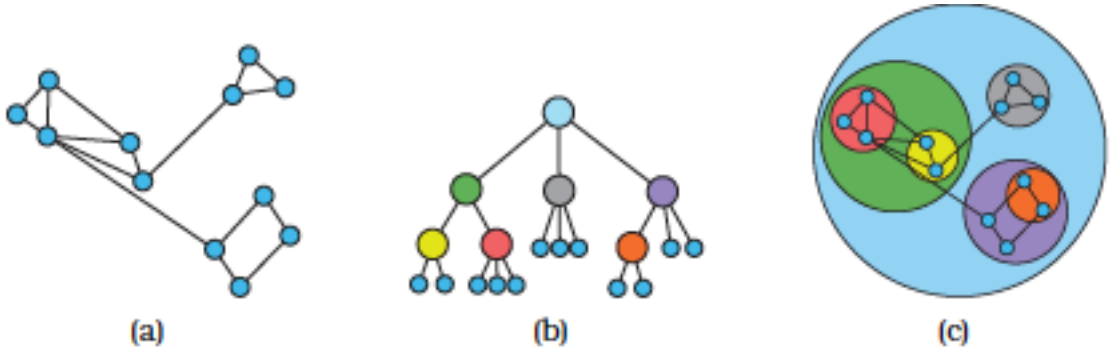
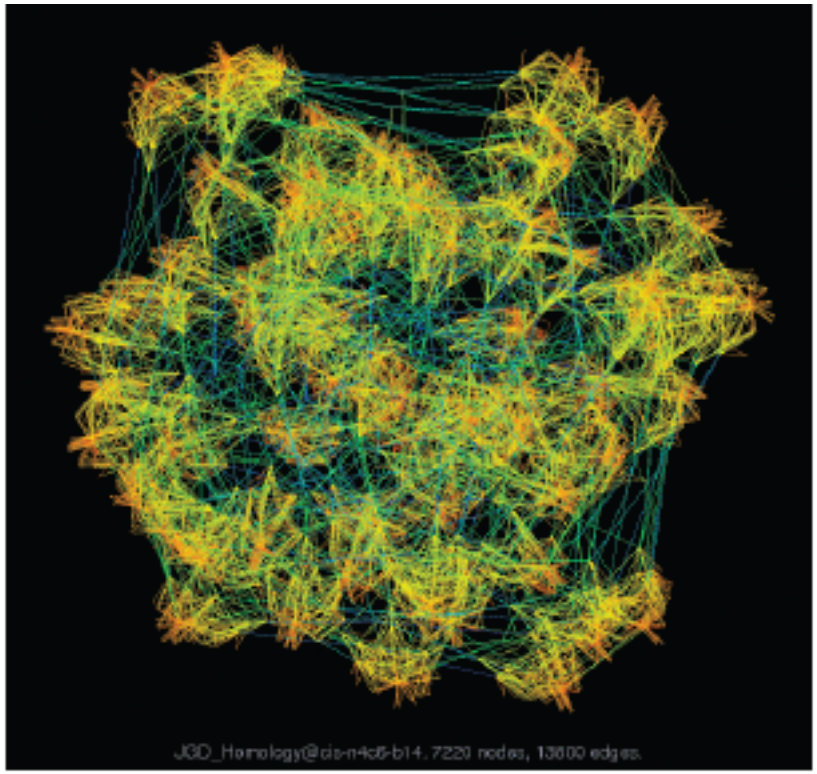


Figure 9.10. GrouseFlocks uses containment to show graph hierarchy structure. (a) Original graph. (b) Cluster hierarchy built atop the graph, shown with a node–link layout. (c) Network encoded using connection, with hierarchy encoded using containment. From [Archambault et al. 08, Figure 3].

Multi-level network drawing [Hu 2005]

- “Coarsen” network
- Multilevel scalable force-directed placement (sfdp)
- Lays out simplified version of network and successively adds detail

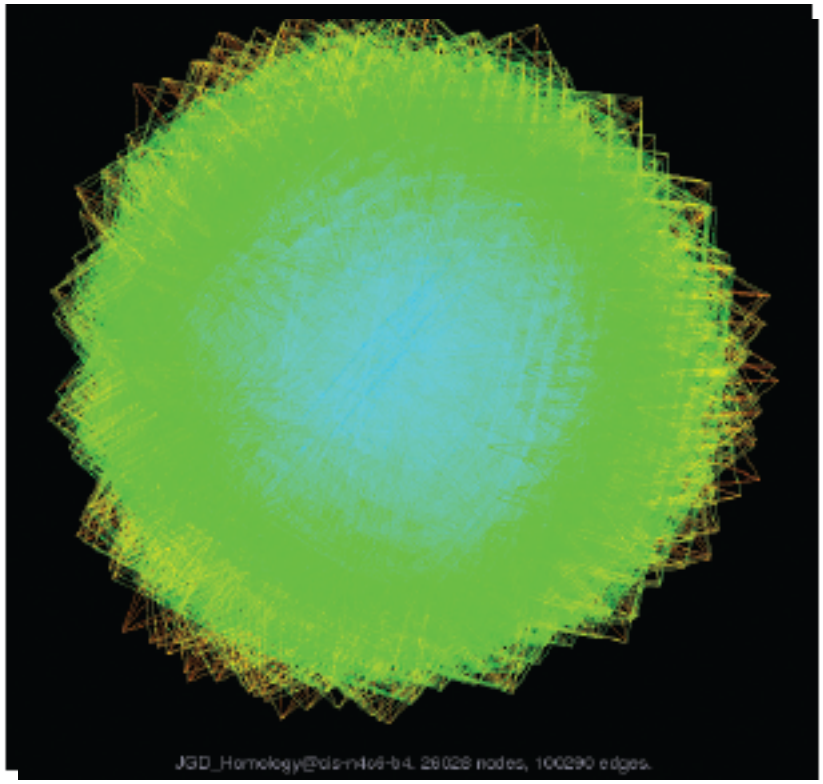
- 7220 nodes
- 13800 edges



Multi-level network drawing [Hu 2005]

- At some point this approach fails as well

- 26028 nodes
- 100,209edges



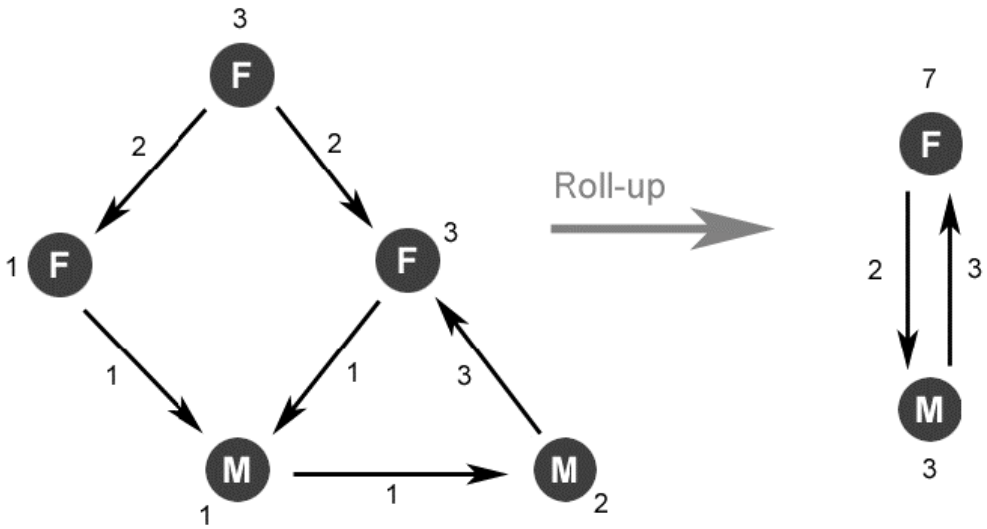
Problem: Multivariate Graphs

- 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

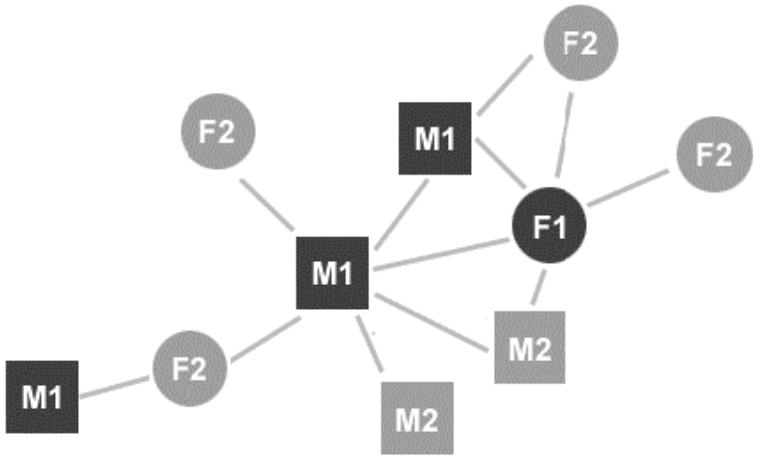


SFU

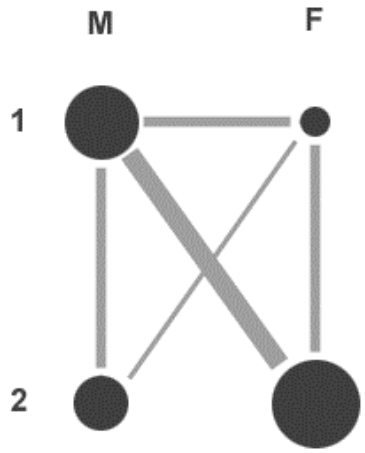
Visual Exploration of Multivariate Graphs, Wattenberg, IEEE Infovis 2009

Multidimensional Pivot Graphs

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



Node and Link Diagram



PivotGraph Roll-up

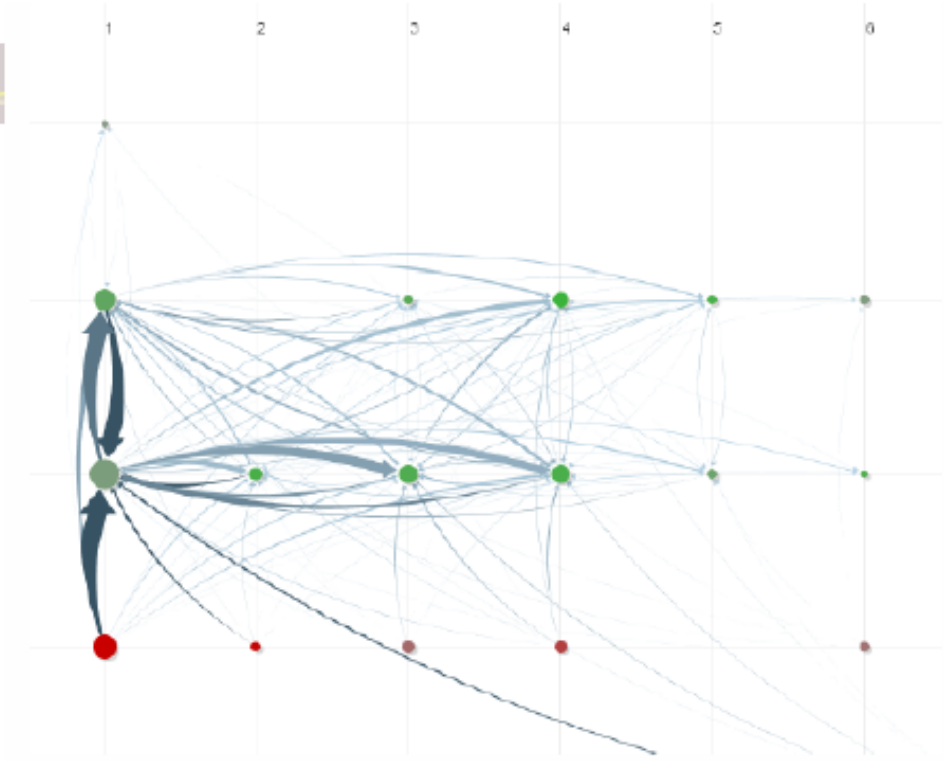
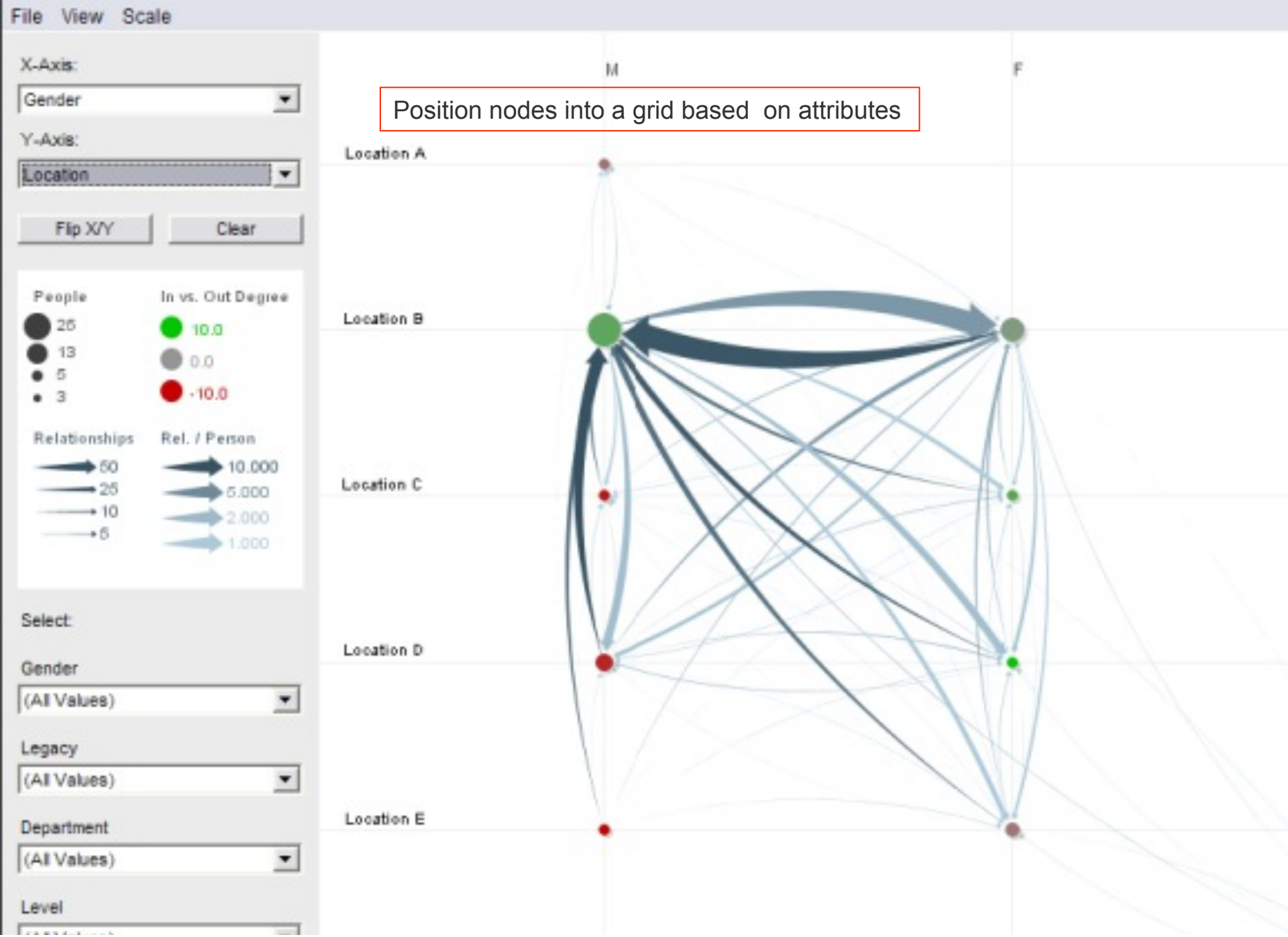
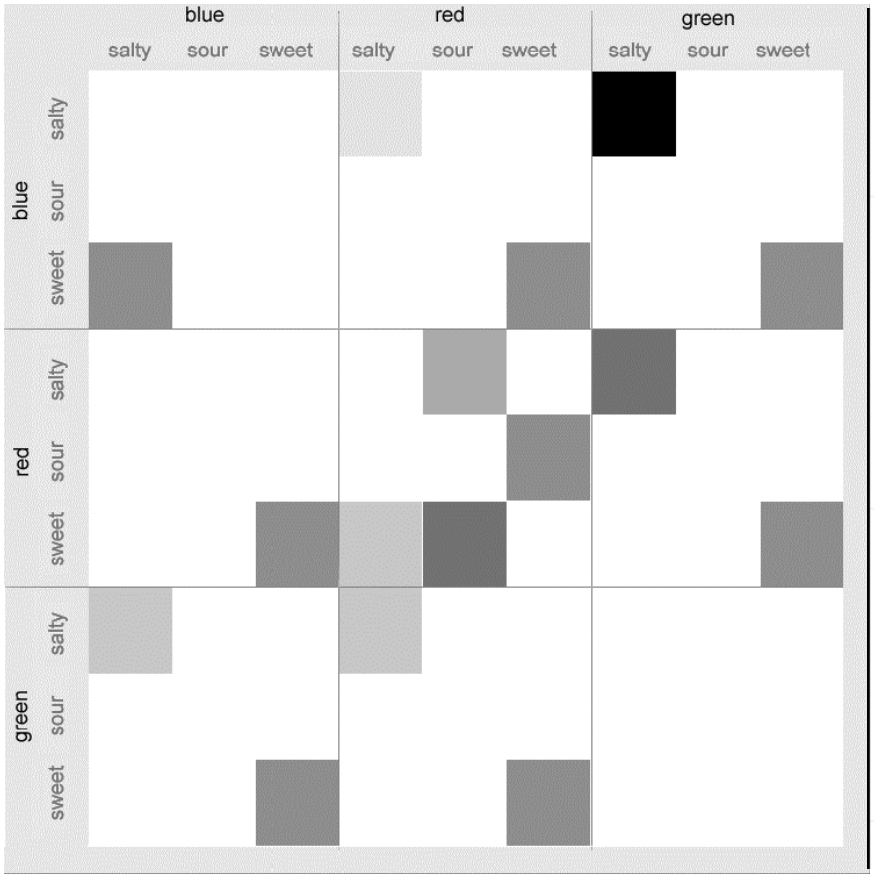
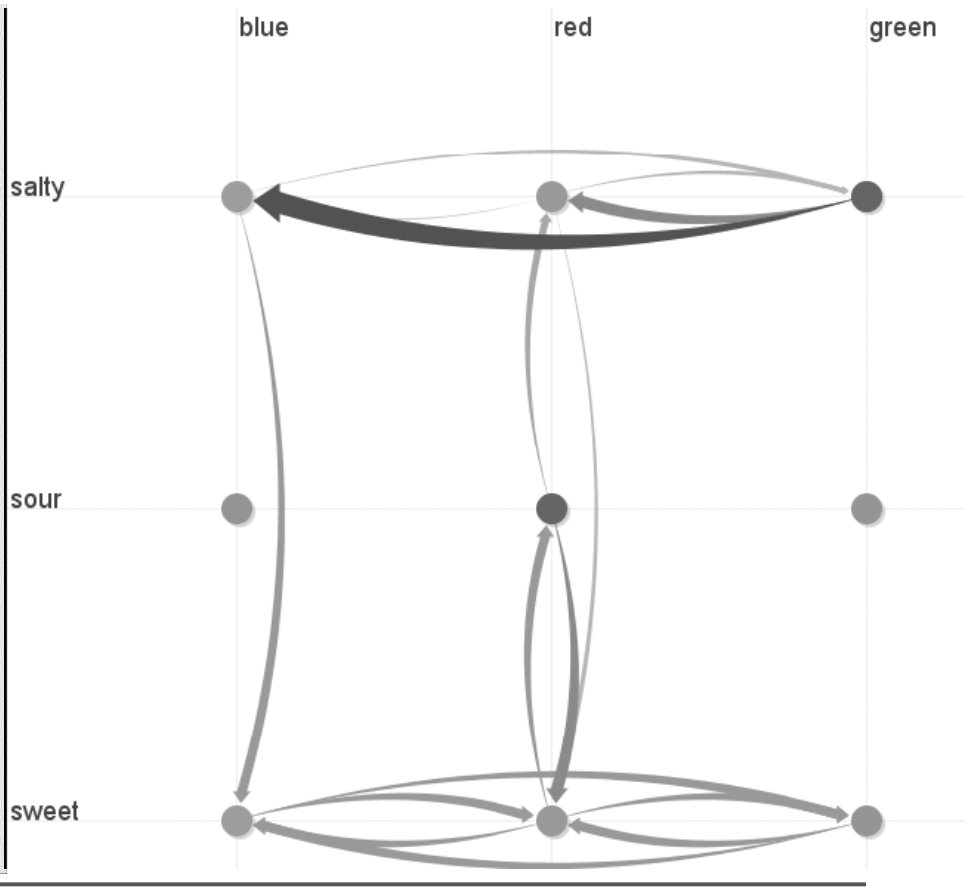


Figure 10. *Communication network of people in a large company. X-axis is division, y-axis is office geography. The division in the leftmost column has far more cross-location communication than the others.*


File View Scale
X-Axis:
Gender
Y-Axis:
Location
Flip X/Y
Clear
People
25
13
5
3
In vs. Out Degree
10.0
0.0
-10.0
Relationships
50
25
10
5
Rel. / Person
10.000
5.000
2.000
1.000
Select:
Gender
(All Values)
Legacy
(All Values)
Department
(All Values)
Level
Position nodes into a grid based on attributes
M
F
Location A
Location B
Location C
Location D
Location E

Compare 2D Pivot Graph with 2D Matrix





Issues with Pivot Graphs

- More compact than matrix view
- Shows aggregate view!
- but

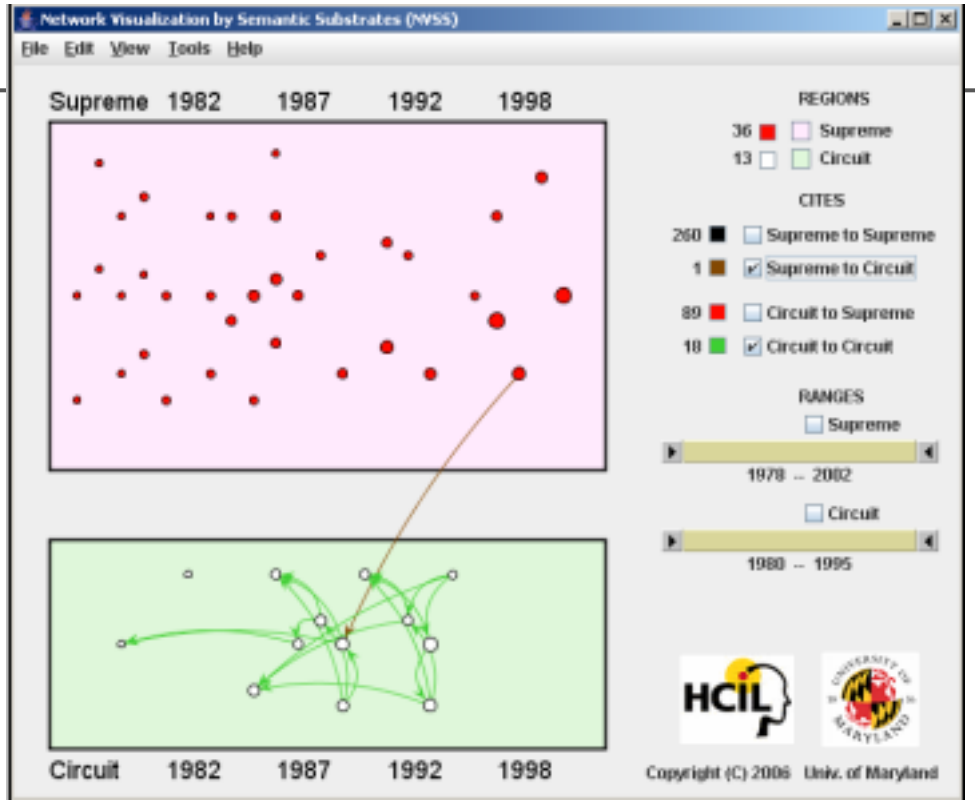
- Disconnected components may become connected
- Acyclic graphs may obtain cycles

Semantic Substrates

- Group nodes into regions
  - According to an attribute
  - Categorical, ordinal, or binned numerical
- In each region:
  - Position nodes according to some other attribute(s)
- Give users control of link visibility

Semantic substrates

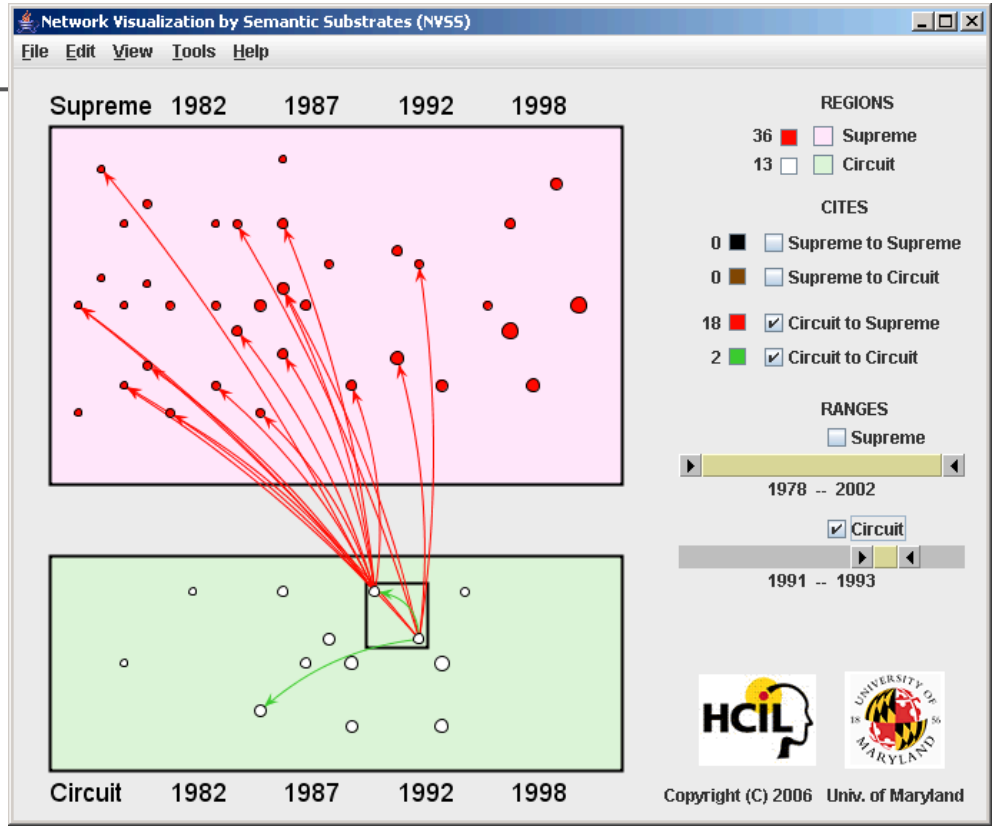
- Specify node groups as substrates (regions to place nodes
- Control link visibility



Shneiderman, B. and Aris, A., Network Visualization by Semantic Substrates, (Proceedings of IEEE Visualization/Information Visualization) IEEE Transactions on Visualization and Computer Graphics 12(5), 733-740, 2006

Semantic substrates

- Specify node groups as substrates (regions to place nodes
- Control link visibility



Shneiderman, B. and Aris, A., Network Visualization by Semantic Substrates, (Proceedings of IEEE Visualization/Information Visualization) IEEE Transactions on Visualization and Computer Graphics 12(5), 733-740, 2006

How might we use graph vis?

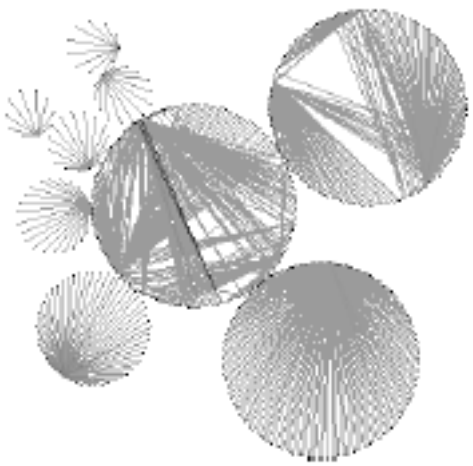
Case study (Stasko)

- NicheWorks (Graham Wills, Lucent): interactive visualization of very large graphs
  - 20,000 – 1M nodes
  - Software energinerring
  - Web analysis
  - Telephone fraud

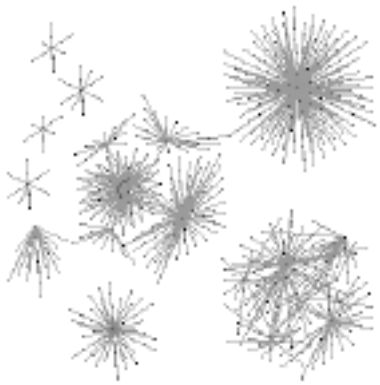
Features

- Sophisticated layout algorithm
  - Circular
  - Hexagonal
  - Tree
- 3 incremental algorithms
  - Steepest descent
  - Swapping
  - repelling

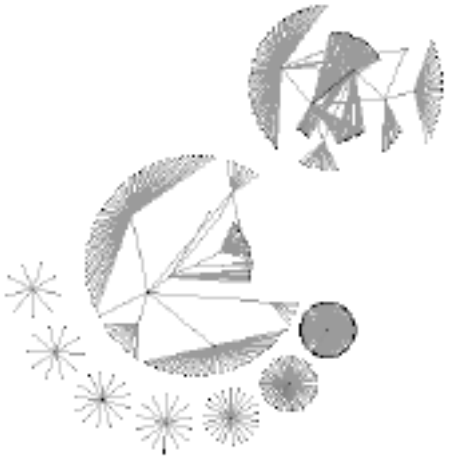
Website example



Circle layout

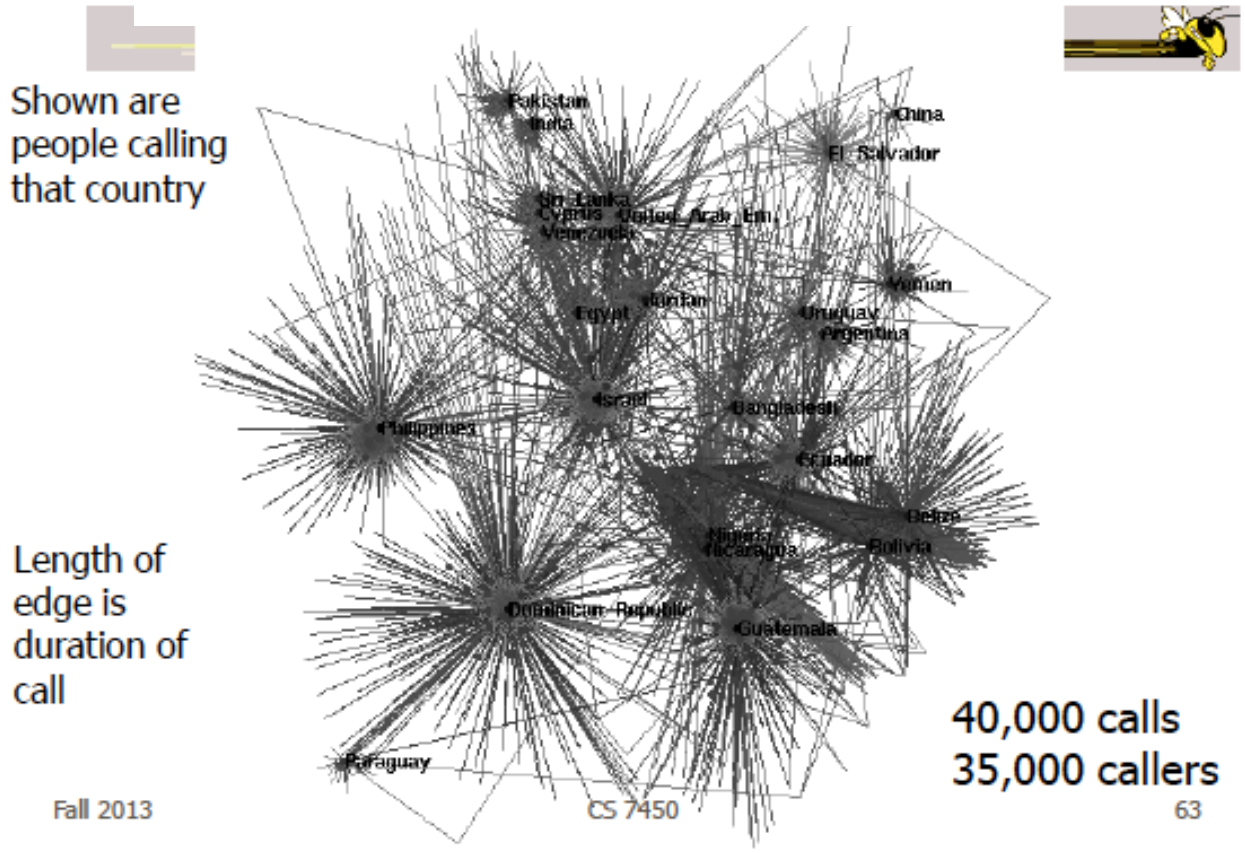


Hexagonal layout

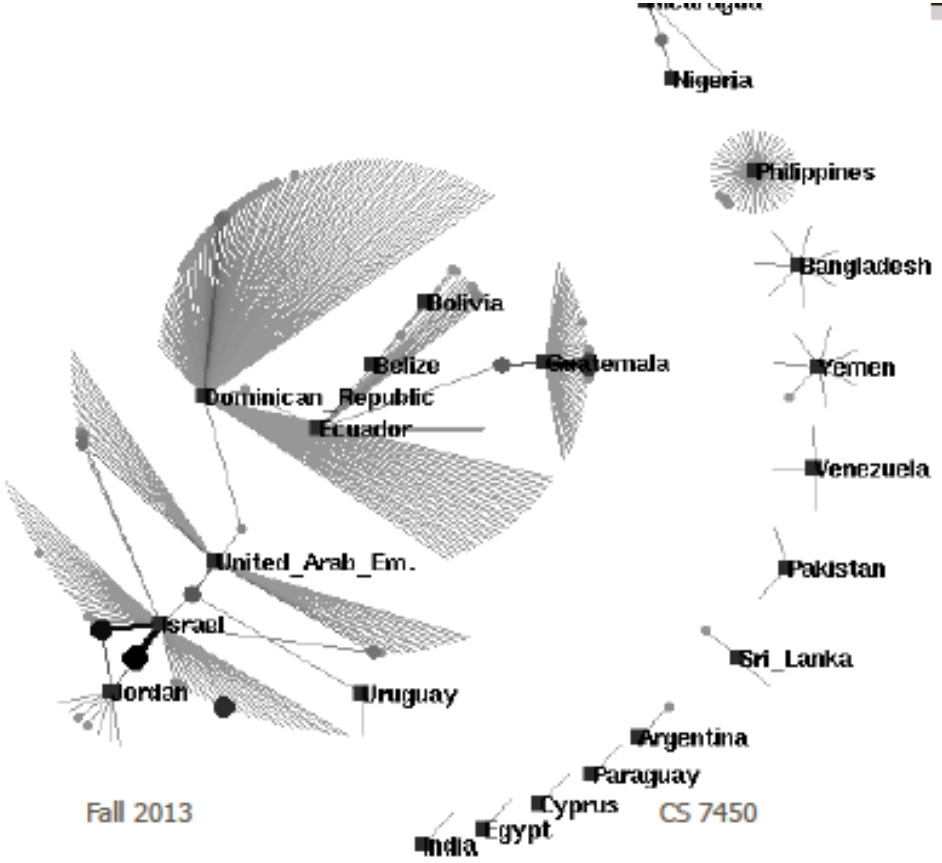


Tree layout

Phone Fraud example



Fraud example

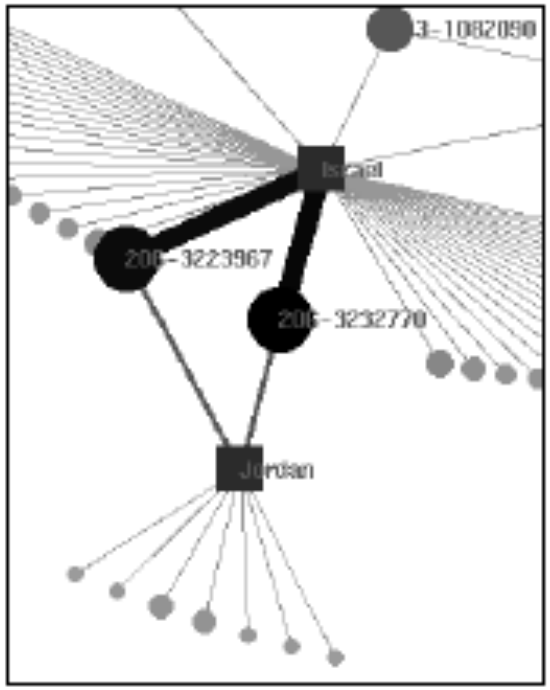


Filtering for people who made multiple calls and spent a significant amount of time on the phone

Playing with parameters like these is important because fraudsters know how to evade

Note the two people calling Israel and Jordan

Fraud example cont.



Zooming in, we notice they have similar calling patterns and numbers (likely part of same operation)

Illegal to call between Israel and Jordan at the time, so fraudsters set up rented apts in US and charge Israeli and Jordanian business people for 3rd party calling

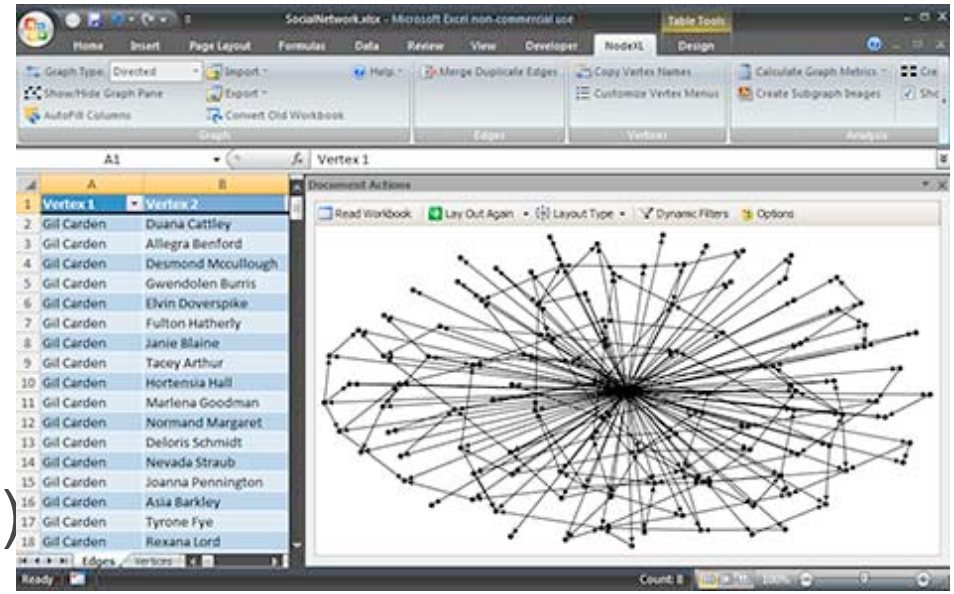
When bills came to US, they would ignore and move on

New toolkits!

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

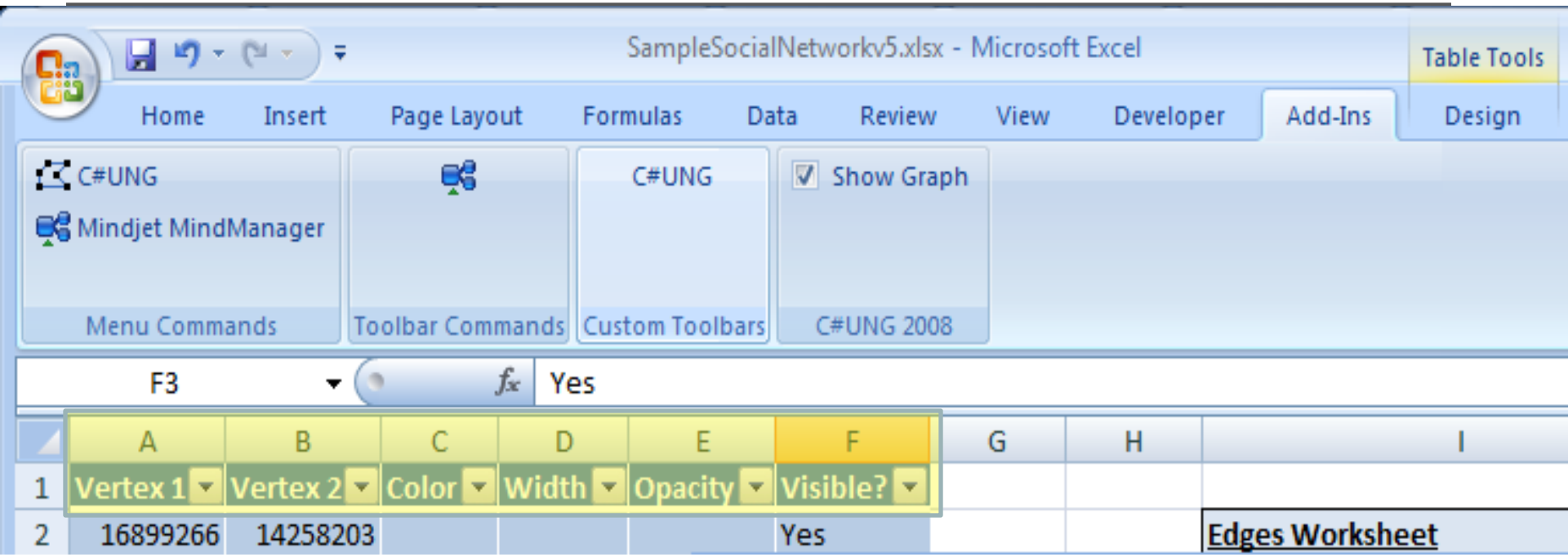
- Used to be called .Netmap
- Now called NodeXL
- Requires windows-specific software

- (Search on “excel NodeXL”)

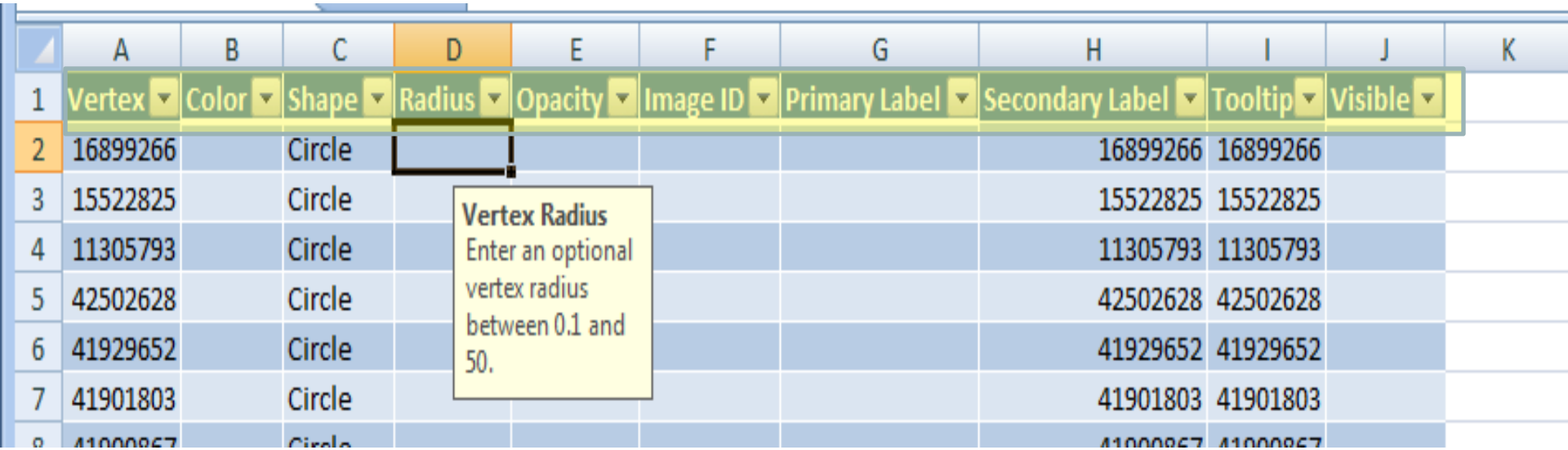


- Chart Tamer for Excel
- Stephen Few et al.

.NetMap: Edges Worksheet



.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		

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 | Connections | Properties | Edit Links | Sort | Filter | Clear | Reapply | Advanced | Text to Columns | Remove Duplicates | Data Validation | Consolidate | What-If Analysis | Group | Ungroup | Subtotal | Data Analysis

E236

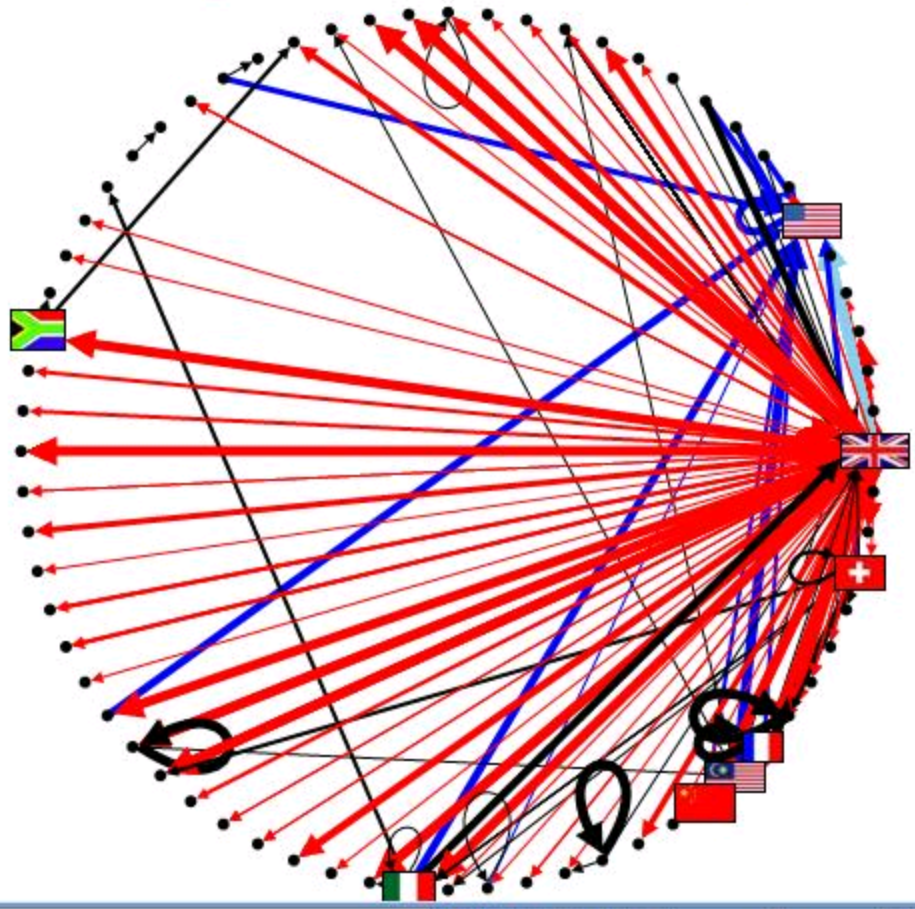
	Vertex 1	Vertex 2	Color	Width	Opacity	Visible?
232	UK	US	Sky Blue	2		
233	UK	US	Sky Blue	3		
234	UK	US	Sky Blue	5		
235	UK	US	Sky Blue	5		
236	UK	US	Sky Blue	5		
237	UK	US	Sky Blue	5		
238	UK	US	Sky Blue	5		
239	UK	US	Sky Blue	5		
240	UK	US	Sky Blue	5		
241	UK	US	Sky Blue	5		
242	UK	US	Sky Blue	2		
243	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).

Edges | Vertices | Images

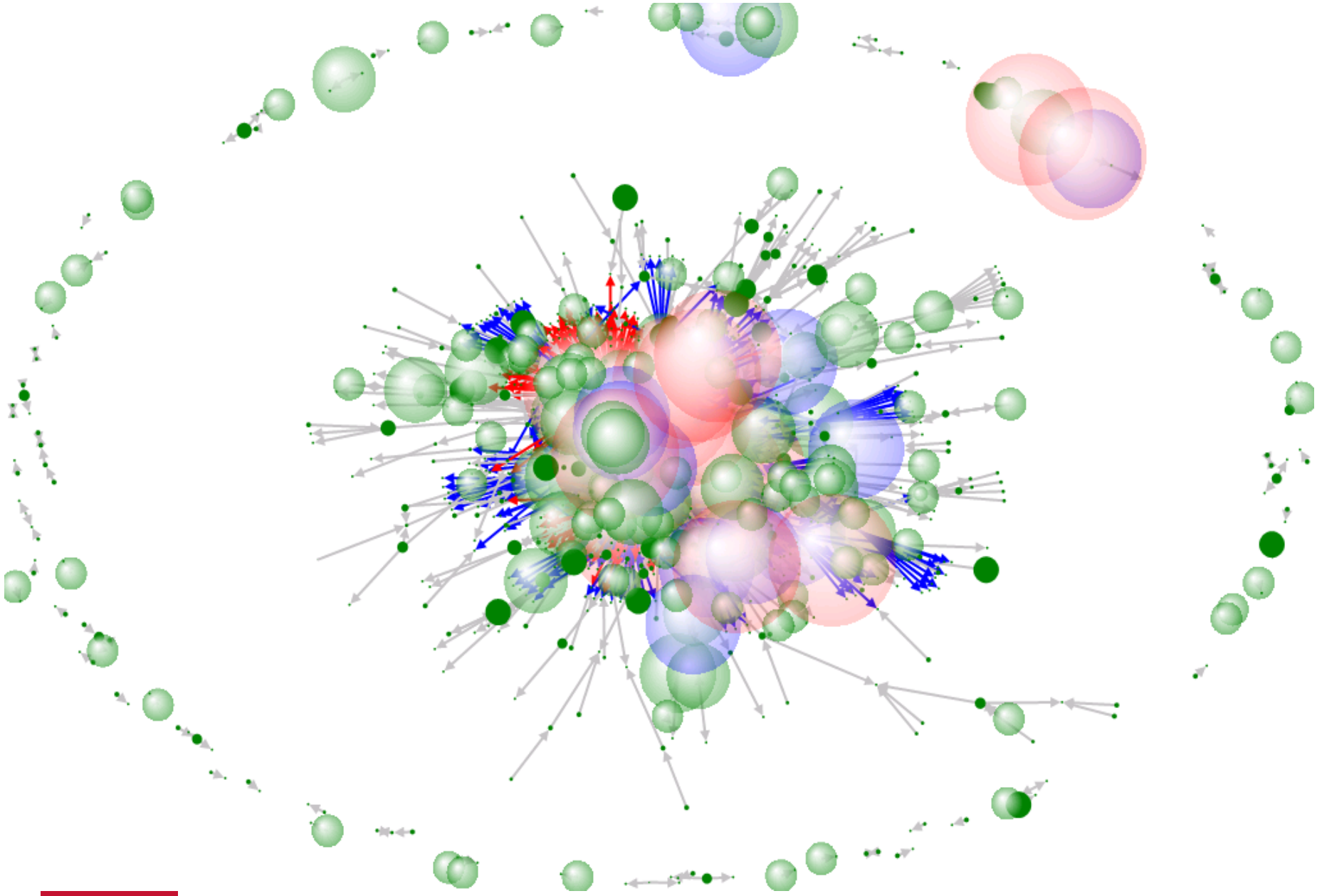
C#UNG 2008

Read Workbook | Refresh Layout | Layout | Metadata | Options

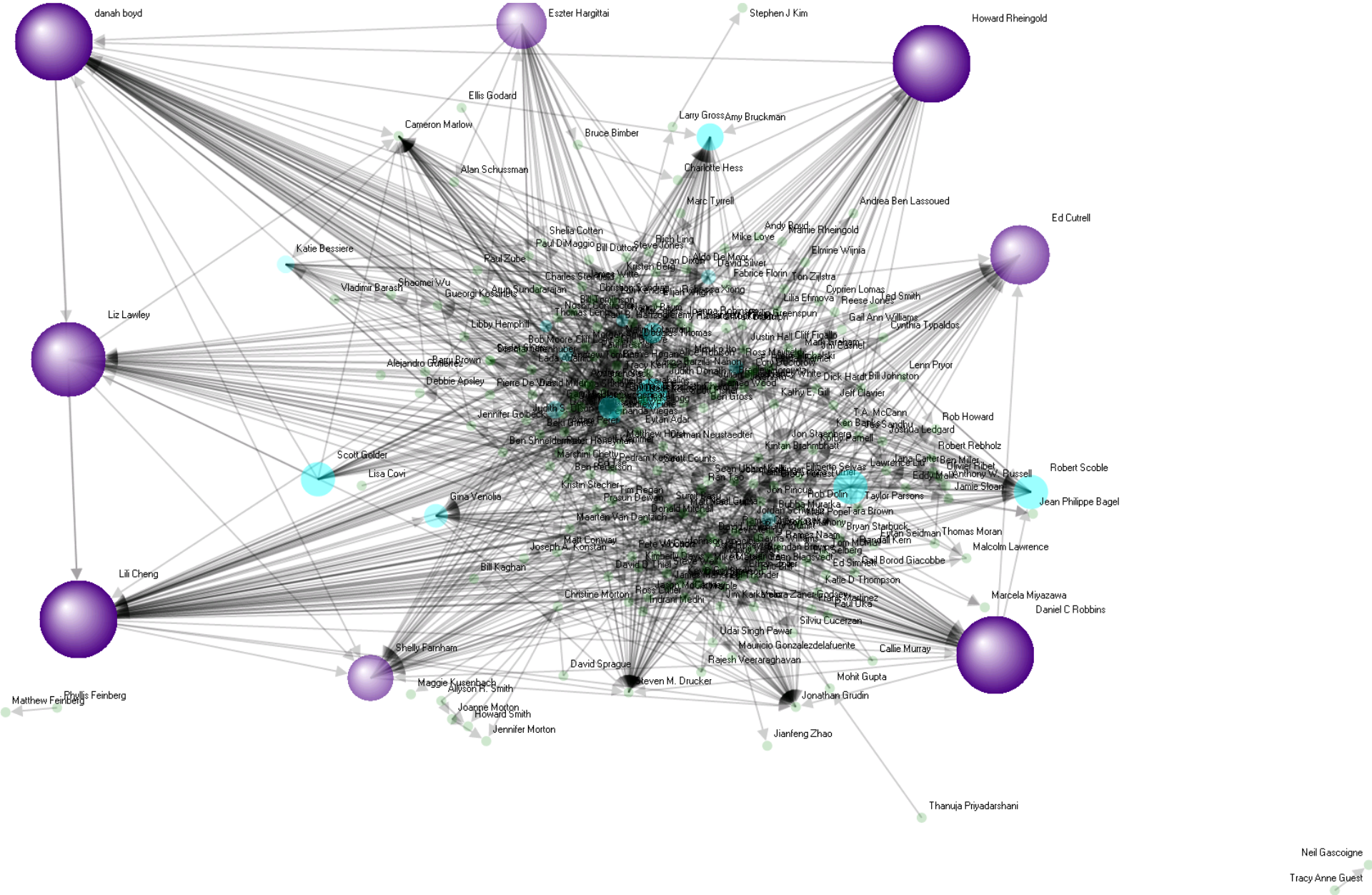


Ready

100%



Marc's Facebook Graph



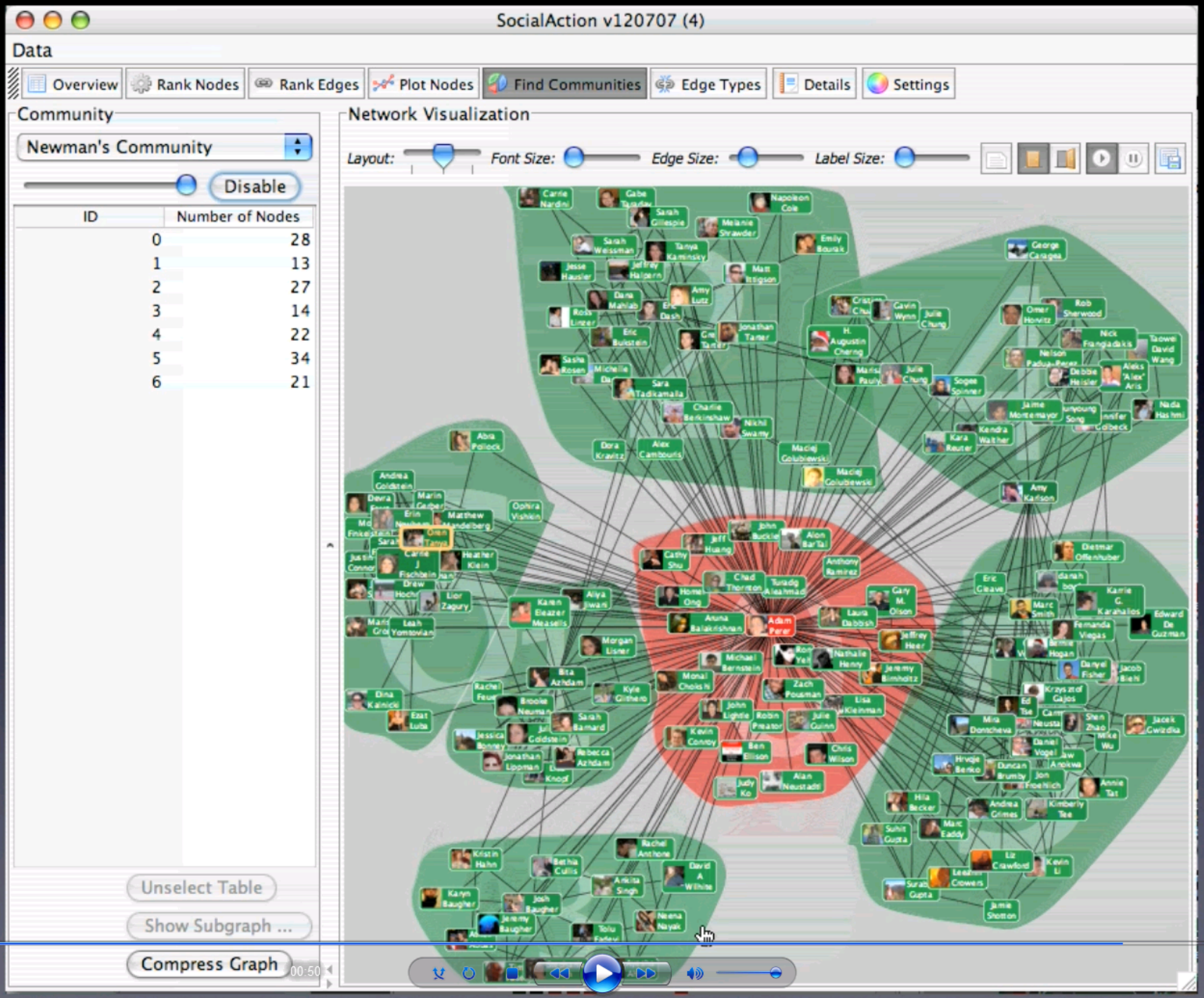
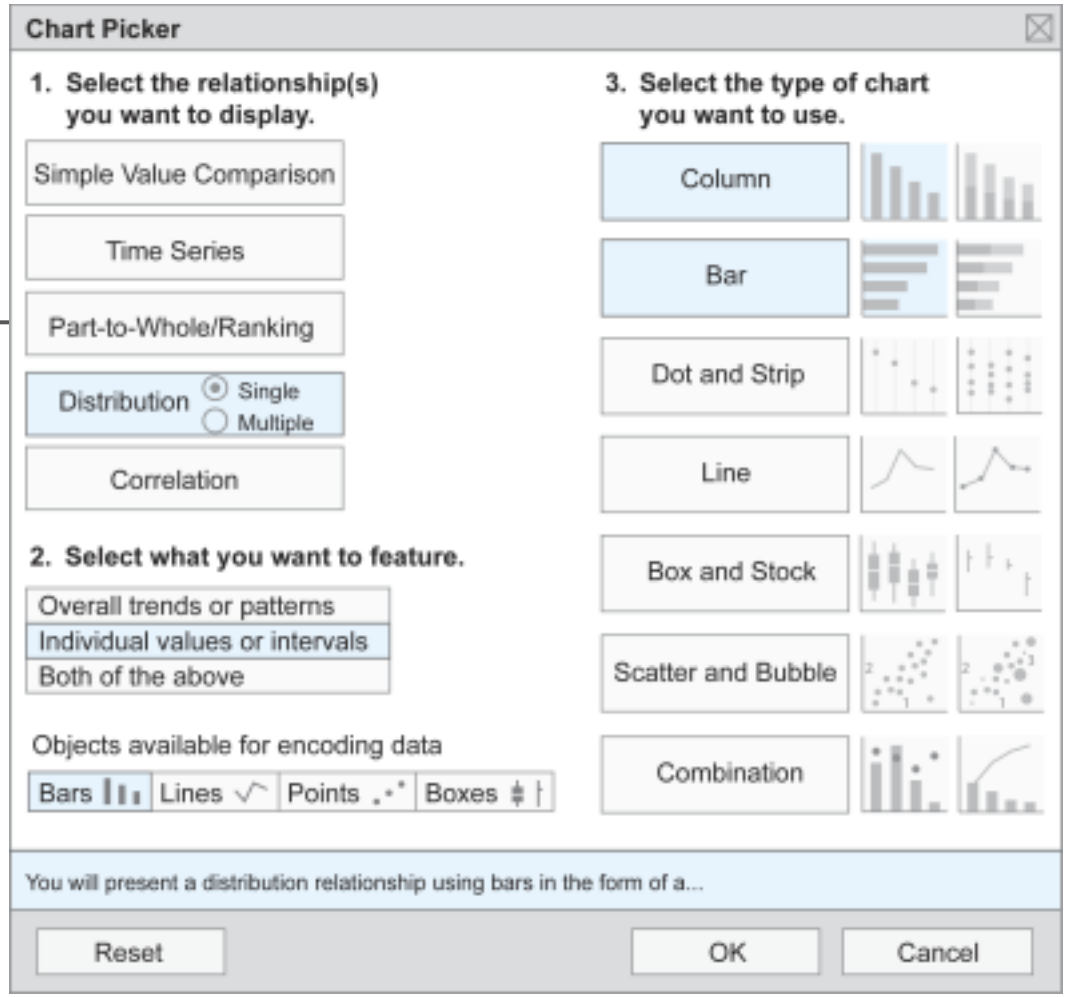

SocialAction v120707 (4)
Data
Overview
Rank Nodes
Rank Edges
Plot Nodes
Find Communities
Edge Types
Details
Settings
Community
Newman's Community
Disable
ID
Number of Nodes
0 28
1 13
2 27
3 14
4 22
5 34
6 21
Unselect Table
Show Subgraph ...
Compress Graph
Network Visualization
Layout:
Font Size:
Edge Size:
Label Size:

Chart Tamer



- Stephen Few + XL^3
- Function-based