Graph and Network Visualization

CS 4460 – Intro. to Information Visualization
October 21, 2014
John Stasko

Connections

- Connections throughout our lives and the world
  - Circle of friends
  - Delta's flight plans
  - …
- Model connected set as a *Graph*

What is a Graph?

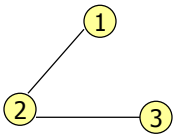
- Vertices (nodes) connected by
- Edges (links)

Adjacency list

1: 2
2: 1, 3
3: 2

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

Adjacency matrix



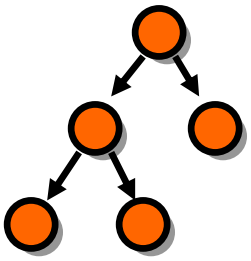
Drawing

Graph Terminology

- Graphs can have *cycles*
- Graph edges can be *directed* or *undirected*
- The *degree* of a vertex is the number of edges connected to it
  - *In-degree* and *out-degree* for directed graphs
- Graph edges can have values (*weights*) on them (nominal, ordinal or quantitative)

Trees are Different

- Subcase of general graph
- No cycles
- Typically directed edges
- Special designated root vertex



Graph Uses

- In information visualization, any number of data sets can be modeled as a graph
  - US telephone system
  - World Wide Web
  - Distribution network for on-line retailer
  - Call graph of a large software system
  - Semantic map in an AI algorithm
  - Set of connected friends
- Graph/network visualization is one of the oldest and most studied areas of InfoVis

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

Layout Examples

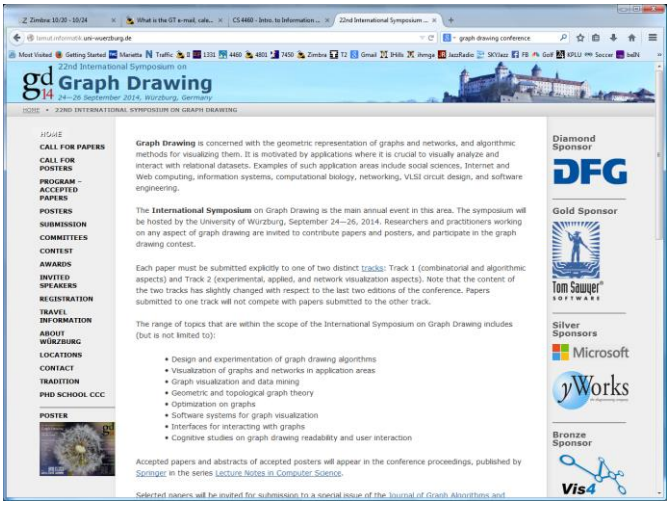
- Homework assignment
- Let's judge!

Results

- What led to particular layouts being liked more?
- Discuss

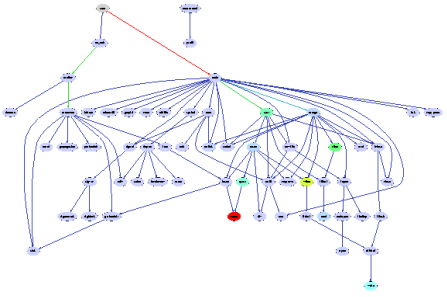
Layout Algorithms

Entire research community's focus

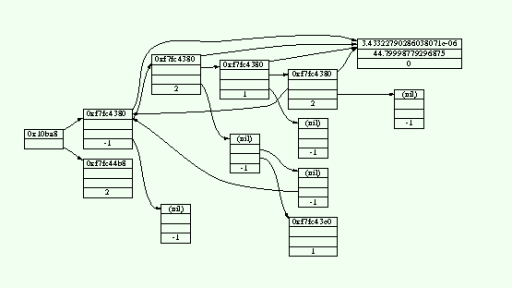


Vertex Issues

- Shape
- Color
- Size
- Location
- Label



Edge Issues



- Color
- Size
- Label
- Form
  - Polyline, straight line, orthogonal, grid, curved, planar, upward/downward, ...

Aesthetic Considerations

- ***Crossings*** -- minimize towards planar
- ***Total Edge Length*** -- minimize towards proper scale
- ***Area*** -- minimize towards efficiency
- ***Maximum Edge Length*** -- minimize longest edge
- ***Uniform Edge Lengths*** -- minimize variances
- ***Total Bends*** -- minimize orthogonal towards straight-line

Which Matters?

- 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

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 about User Tasks?

- So what do people want to do with or learn from network visualizations?
  - Recurring theme of this class: Too often this is neglected

Graph Vis Task Taxonomy

- Start with Amar et al '05 low-level tasks
- Then add four types of other tasks (next pages)

Graph Vis Task Taxonomy

- 1. Topology-based tasks
  - Adjacency
    Find the set of nodes adjacent to a node
  - Accessibility
    Find the set of nodes accessible to a node
  - Common connection
    Given nodes, find the set of nodes connected to all
  - Connectivity
    Find shortest path
    Identify clusters
    Identify connected components

Graph Vis Task Taxonomy

- 2. Attribute-based tasks
  - On the nodes
    - Find the nodes having a specific attribute value
  - On the edges
    - Given a node, find the nodes connected only by certain kinds of edges

Graph Vis Task Taxonomy

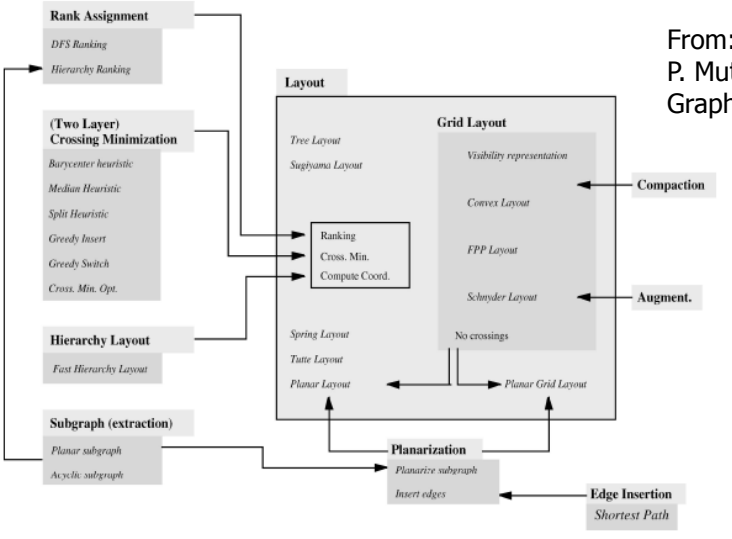
- 3. Browsing tasks
  - Follow path
    - Follow a given path
  - Revisit
    - Return to a previously visited node
- 4. Overview task
  - Compound exploratory task
    - Estimate size of a network
    - Find patterns

Layout Heuristics

- Layout algorithms can be
  - polyline edges
  - planar
    - No edge crossings
  - orthogonal
    - horizontal and vertical lines/polylines
  - grid-based
    - vertices, crossings, edge bends have integer coords
  - curved lines
  - hierarchies
  - circular
  - ...

Types of Layout Algorithms

From:
P. Mutzel, et al
Graph Drawing '97



Common Layout Techniques

- Hierarchical
- Force-directed
- Circular
- Geographic-based
- Clustered
- Attribute-based
- Matrix

We will discuss many of these further in the slides to come

Scale Challenge

- May run out of space for vertices and edges (turns into "ball of string")
- Can really slow down algorithm

- Sometimes use *clustering* to help
  - Extract highly connected sets of vertices
  - Collapse some vertices together

Navigation/Interaction Challenge

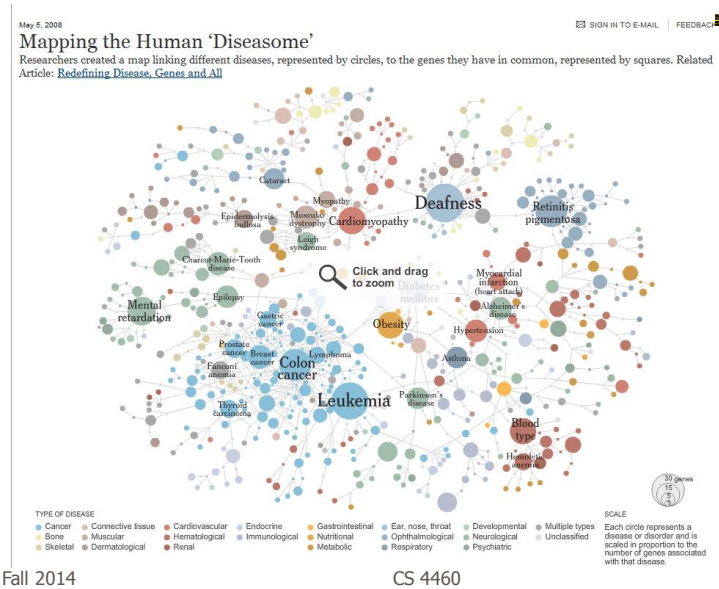
- How do we allow a user to query, visit, or move around a graph?
- Changing focus may entail a different rendering

Graph Drawing Uses

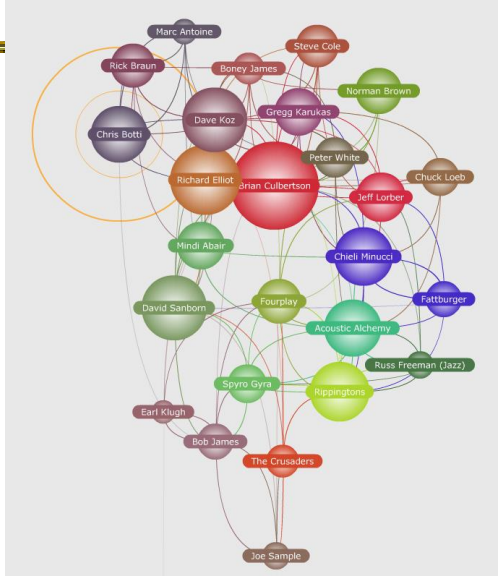
- Many domains and data sets can benefit significantly from nice graph drawings
- Let's look at some examples…

http://www.nytimes.com/interactive/2008/05/05/science/20080506_DISEASE.html

Human Diseases



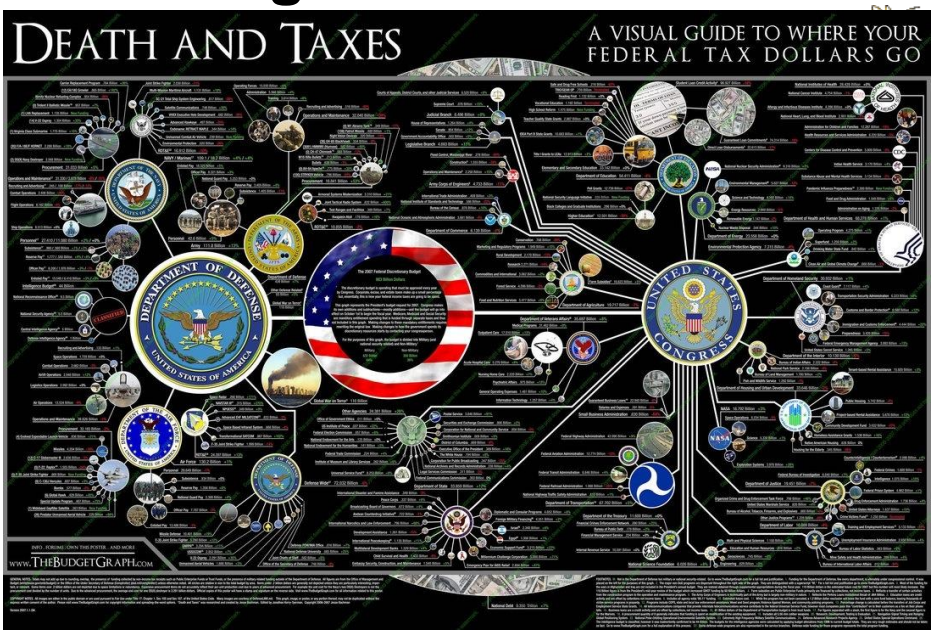
Music Artists



http://www.liveplasma.com/

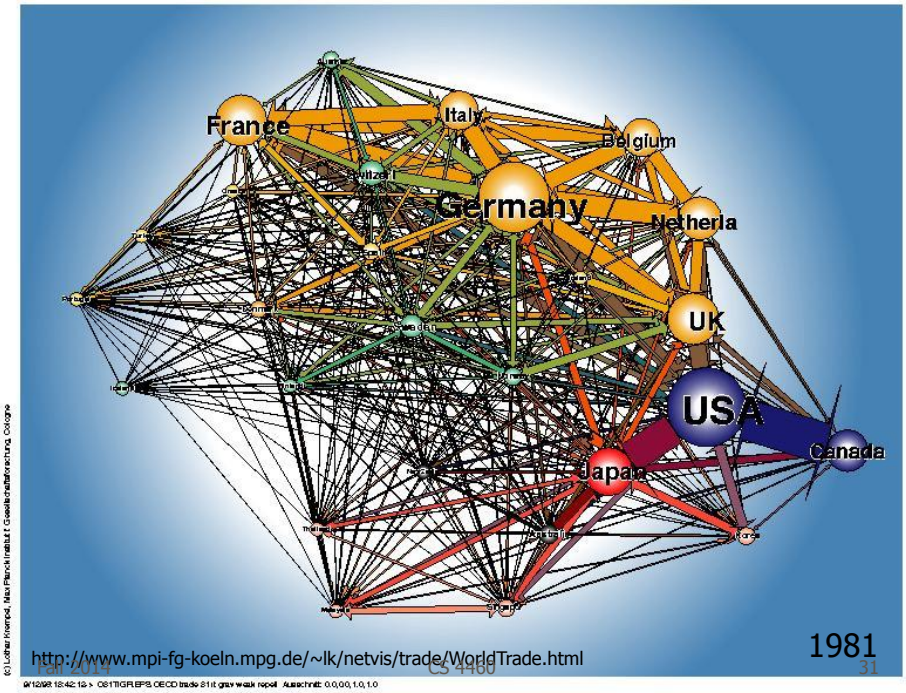
http://mibi.deviantart.com/art/Death-and-Taxes-2007-39894058

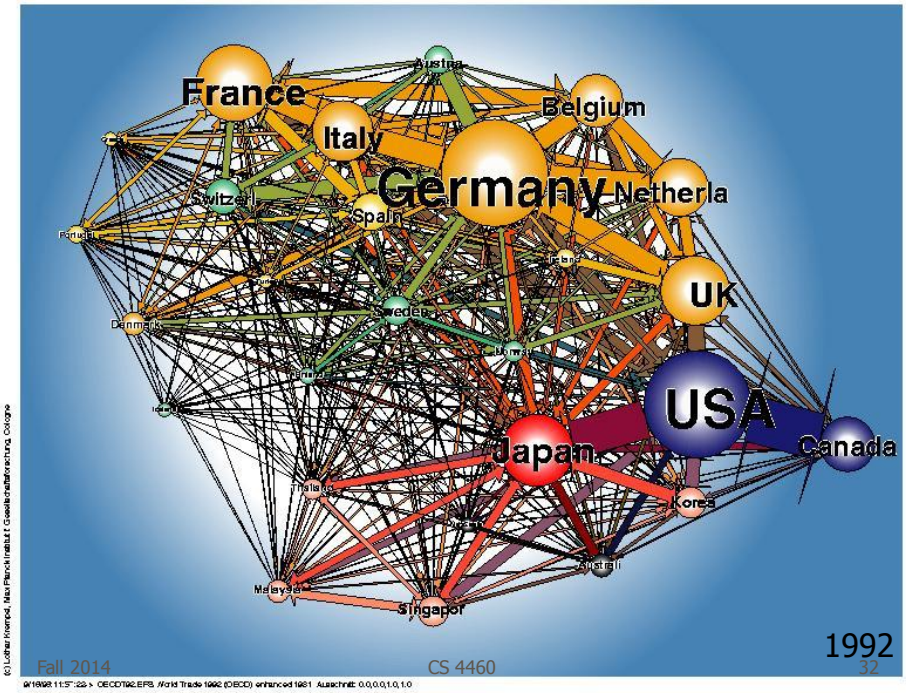
US Budget



Social Analysis

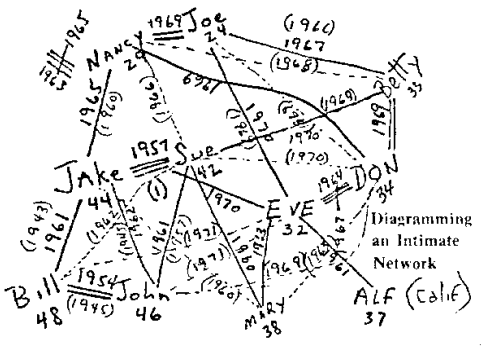
- Facilitate understanding of complex socio-economic patterns
- Social Science visualization gallery (Lothar Krempel):
  - http://www.mpi-fg-koeln.mpg.de/~lk/netvis.html
- Next slides: Krempel & Plumper's study of World Trade between OECD countries, 1981 and 1992


France
Italy
Belgium
Germany
Netherla
UK
USA
Canada
Japan
http://www.mpi-fg-koeln.mpg.de/~lk/netvis/trade/WorldTrade.html
1981


France
Austria
Belgium
Italy
Germany
Netherla
Spain
UK
USA
Canada
Japan
Korea
Singapor
1992

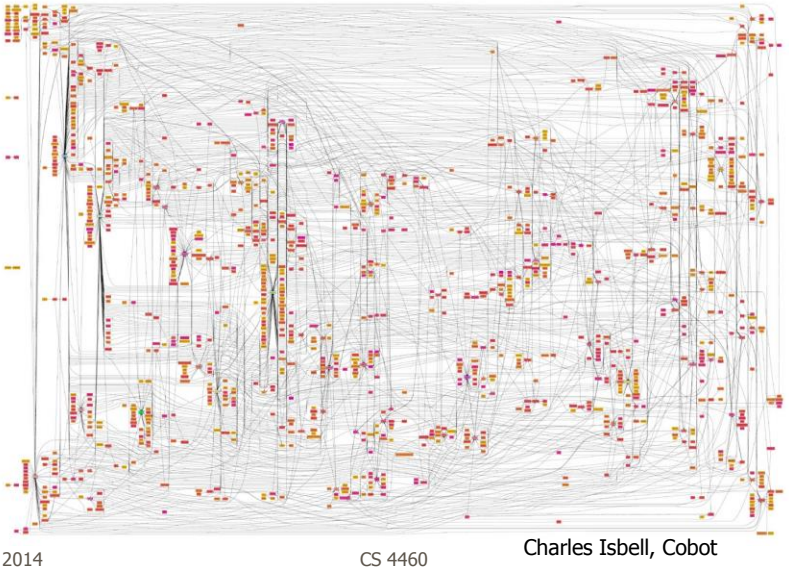
Social Network Visualization

- Social Network Analysis
  - http://www.insna.org



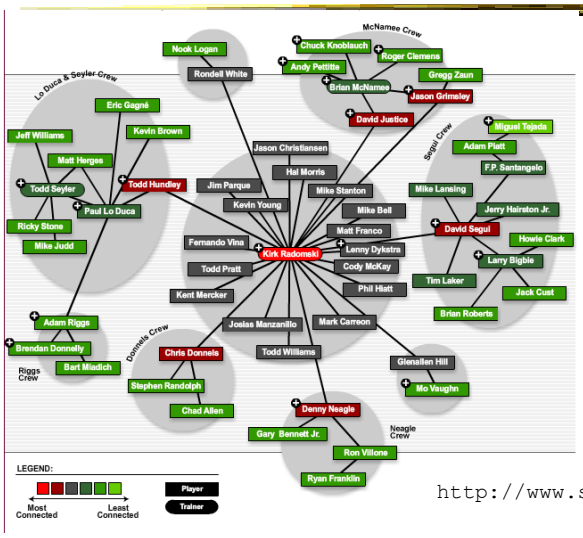
Hot topic again
Why?
Terrorists
Facebook

People connections



Charles Isbell, Cobot

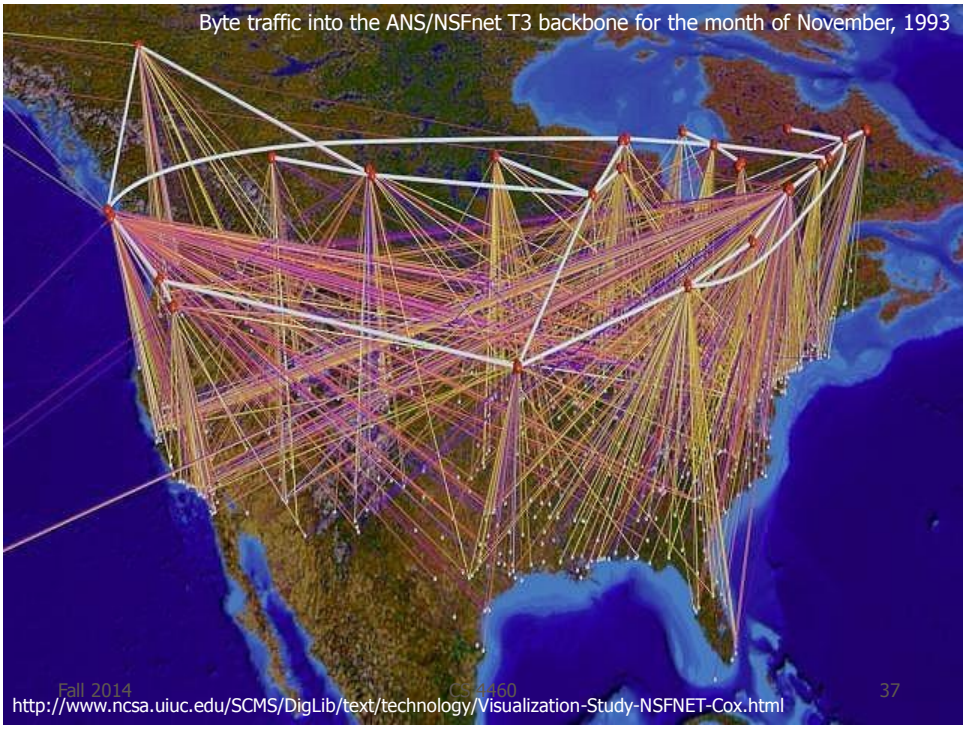
Steroids in MLB



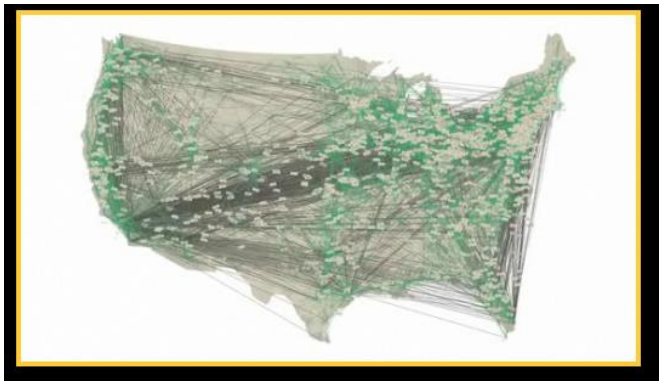
http://www.slate.com/id/2180392/

Geo Applications

- Many problems and data sets have some geographic correspondence



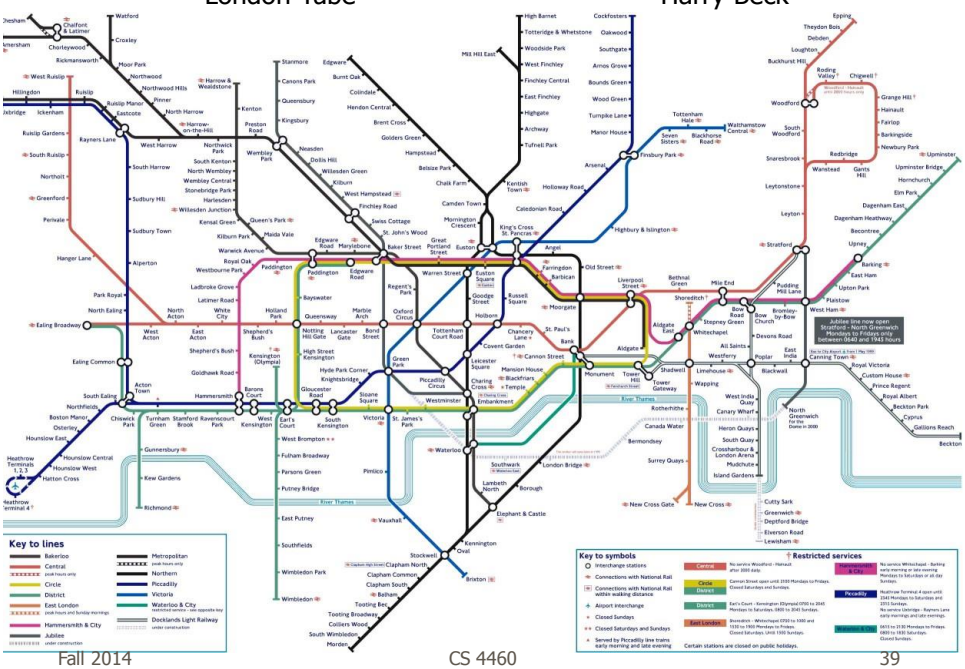
Follow the Money

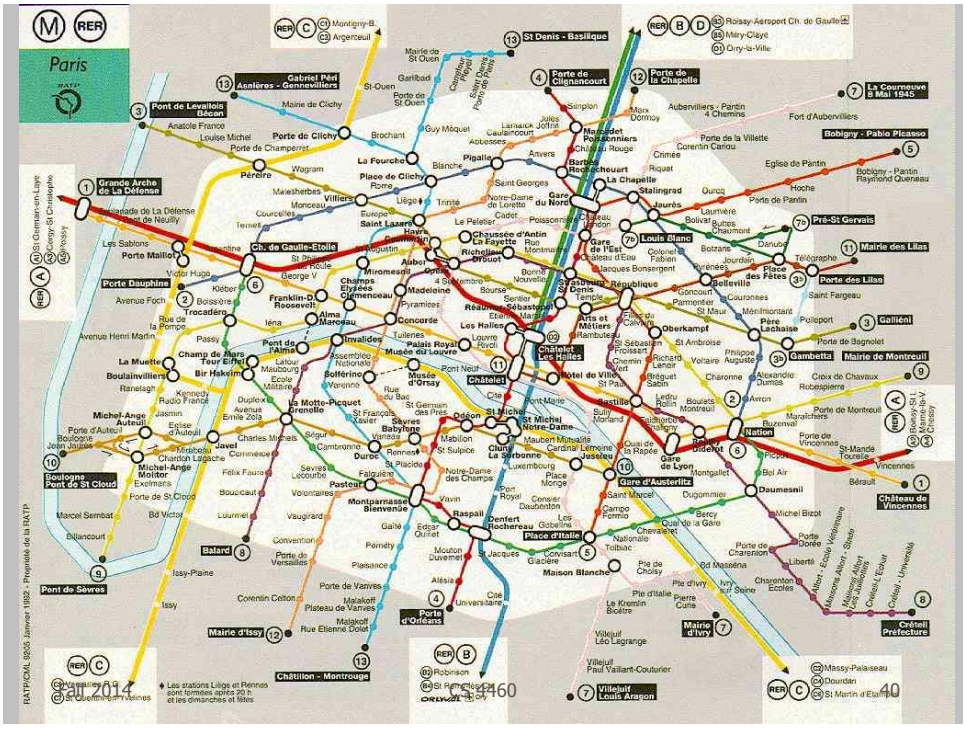


Where does a dollar bill go?

http://www.nsf.gov/news/special_reports/scivis/follow_money.jsp

London Tube Harry Beck





Atlanta
MARTA

3 Subway Diagrams

- Geographic landmarks largely suppressed on maps, except water (rivers in London & Paris) and asphalt (highways in Atlanta)
  - Rather fitting, no?
- These are more *graphs* than maps!

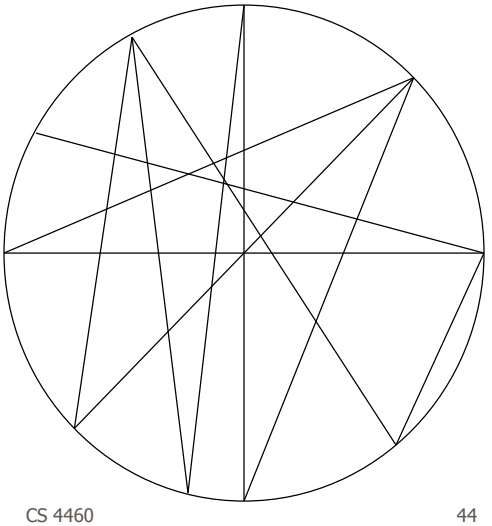
But Is It InfoVis?

- I generally don't consider a pure graph layout (drawing) algorithm to be InfoVis
  - Nothing wrong with that, just an issue of focus
- For InfoVis, I like to see some kind of interaction or a system or an application…
  - Still, understanding the layout algorithms is very important for infovis
  - Let's look at a few…

Circular Layout

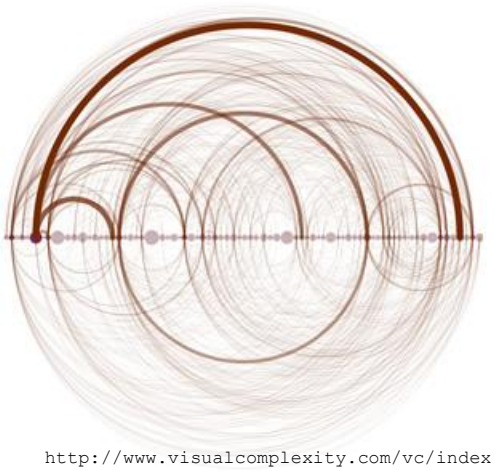
Ultra-simple
May not look so great

Space vertices out around circle
Draw lines (edges) to connect vertices



Arc Diagram Layout

Wattenberg
InfoVis '02



http://www.visualcomplexity.com/vc/index.cfm?method=Arc%20Diagrams

Tree Layout

- Run a breadth-first search from a vertex
  - This imposes a spanning tree on the graph
- Draw the spanning tree

- Simple and fast, but obviously doesn't represent the whole graph

Hierarchical Layout

Often called Sugiyama layout

Try to impose hierarchy on graph
- Reverse edges if needed to remove cycles
- Introduce dummy nodes

Put nodes into layers or levels
Order l->r to minimize crossings

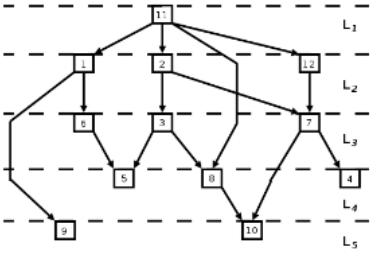


Figure: A graph showing a layered layout, created with the Sugiyama heuristic, with the layers shown. The bends in the edges correspond to dummy nodes.

http://www.csse.monash.edu.au/hons/se-projects/2006/Kieran.Simpson/output/html/node7.html#sugiyamaexample

Force-directed Layout

- Example of constraint-based layout technique
- Impose constraints (objectives) on layout
  - Shorten edges
  - Minimize crossings
  - …
- Define through equations
- Create optimization algorithm that attempts to best satisfy those equations

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

Force-directed Example

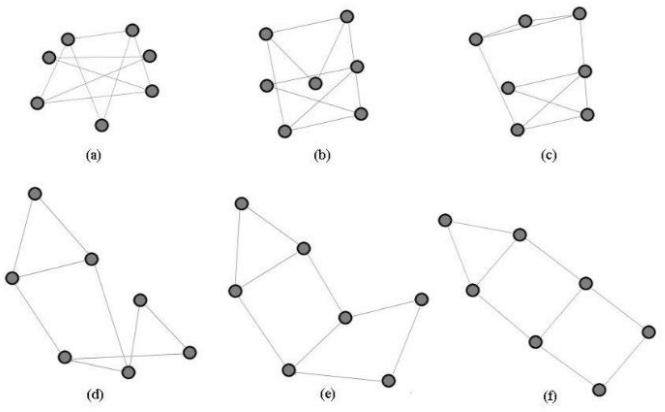
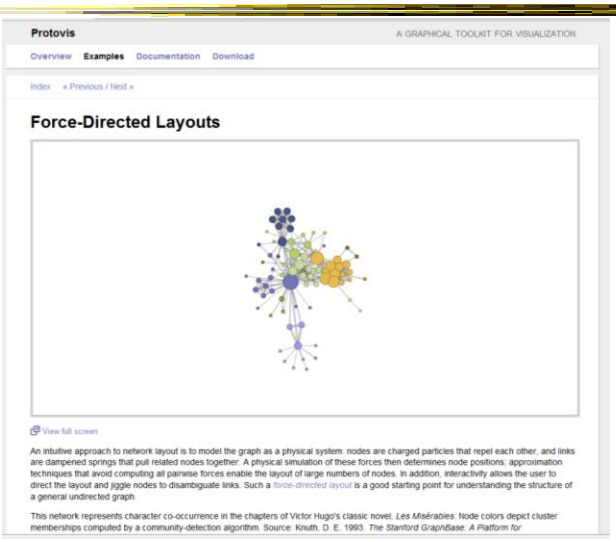


Figure 2: A graph drawing through a number of iterations of a force directed algorithm.

http://www.cs.usyd.edu.au/~aquigley/3dfade/

http://vis.stanford.edu/protovis/ex/force.html

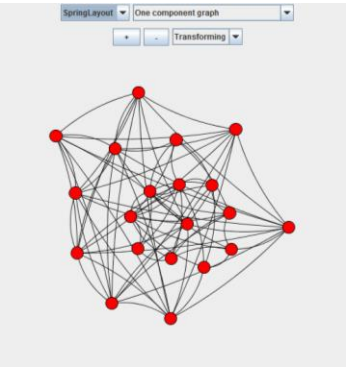
In Action



Variant

Images from JUNG

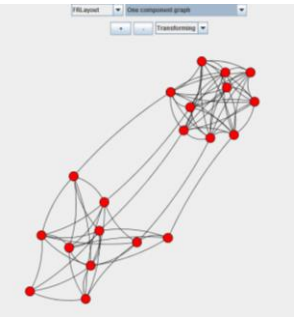
- Spring layout
  - Simple force-directed spring embedder



Variant

Images from JUNG

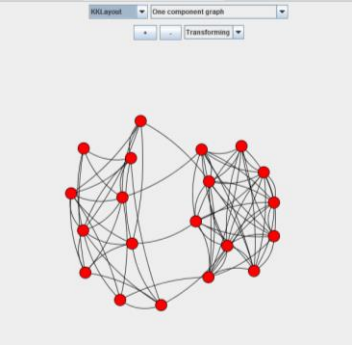
- Fruchterman-Reingold Algorithm
  - Add global temperature
  - If hot, nodes move farther each step
  - If cool, smaller movements
  - Generally cools over time



Variant

Images from JUNG

- Kamada-Kawai algorithm
  - Examines derivatives of force equations
  - Brought to zero for minimum energy



Other Applications

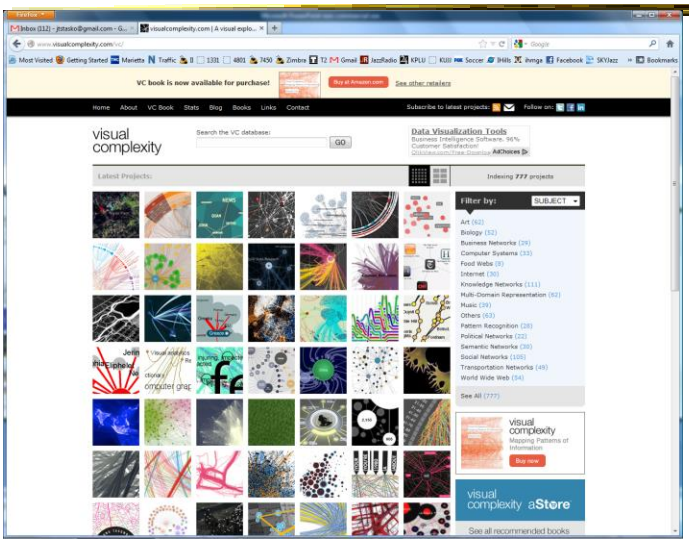
- Email
- How would you visualize all email traffic in CoC between pairs of people?
- Solutions???

Possible Solutions

- Put everyone on circle, lines between
  - Color or thicken line to indicate magnitude
- Use spring/tension model
  - People who send a lot to each other are drawn close together
  - Shows clusters of communications

http://www.visualcomplexity.com

Mucho Examples



Interaction

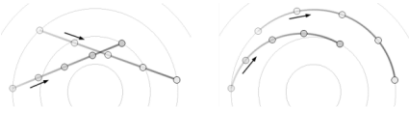
- One of the key ways we move beyond graph layout to graph visualization (InfoVis) is interaction with the graph

Focus of Graph

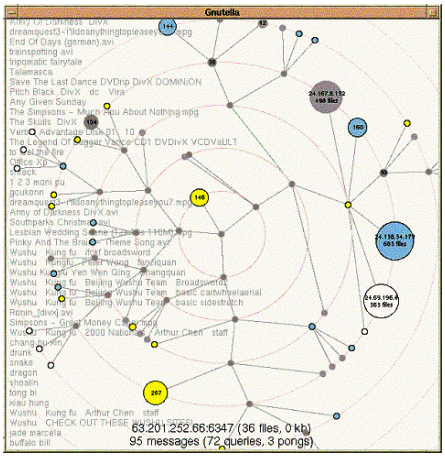
- Particular node may be focus, often placed in center for circular layout
- How does one build an interactive system that allows changes in focus?
  - Use animation
  - But intuition about changes not always right

Focus Change Animation

Straight linear interpolation of focus changes not as appealing as changes along polar coordinates



Yee, Fisher, Dhamija, Hearst
InfoVis '01



Video

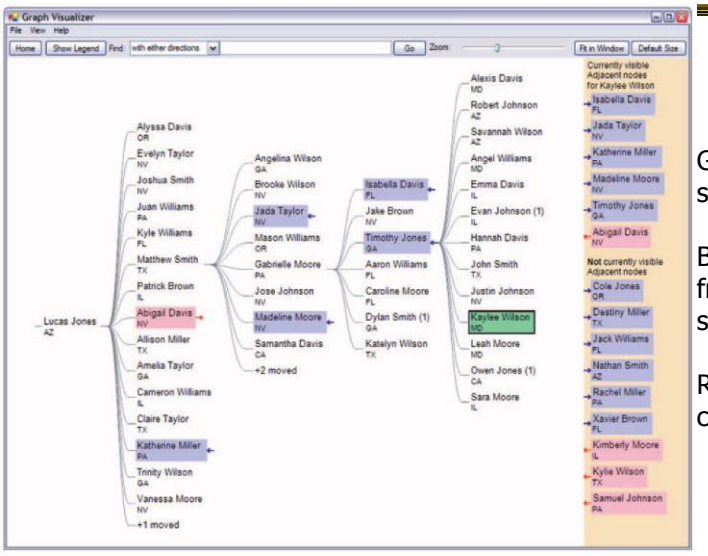
TreePlus

- Don't draw entire graph
- Have a focus vertex, then incrementally expand and show connections (min span tree) from there
- Interaction:
  - Single-click: show connections via highlight
  - Double-click: new focus vertex
  - Smooth animated change in focus
- "Plant a seed and watch it grow"

Lee et al *TVCG* '06

TreePlus

Video



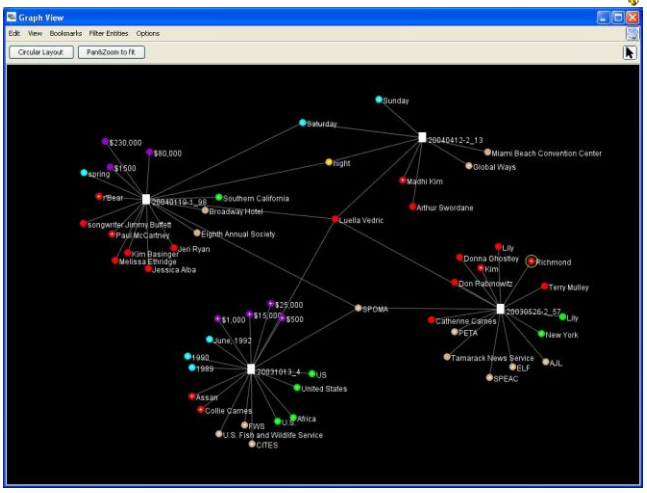
Green – current selection

Blue – vertices from current selection

Red – vertices to current selection

Jigsaw's Graph View

Don't draw everything, but allow the viewer to interactively explore (expand & compress) the graph



Stasko, Görg & Liu
Information Visualization '08

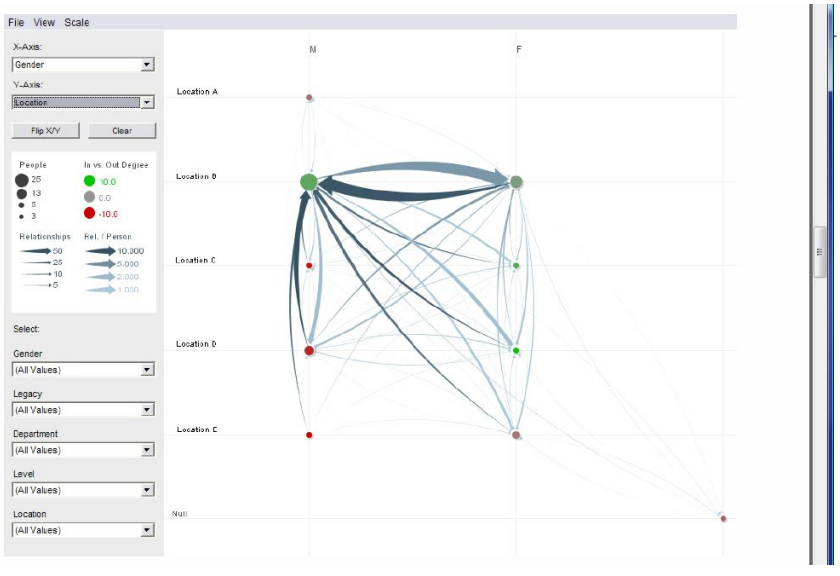
Recent Trends in GraphViz

- Attributes of nodes influence geometric positioning
  - Not just some arbitrary layout
- Utilize graph statistical analysis too
- Largely driven by interest in social network analysis

PivotGraph

- Cluster on common node attributes
  - Put all A's together, all B's together, …
- "Roll up" nodes
  - Draw edge from A to B depending on how many edges from some A to some B
- Position nodes into a grid based on attributes

Wattenberg
CHI '06



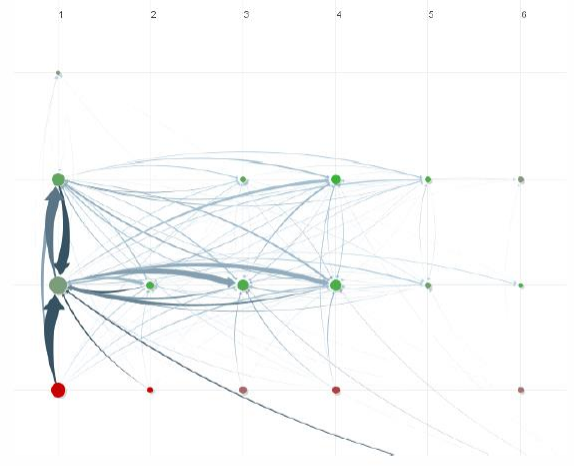


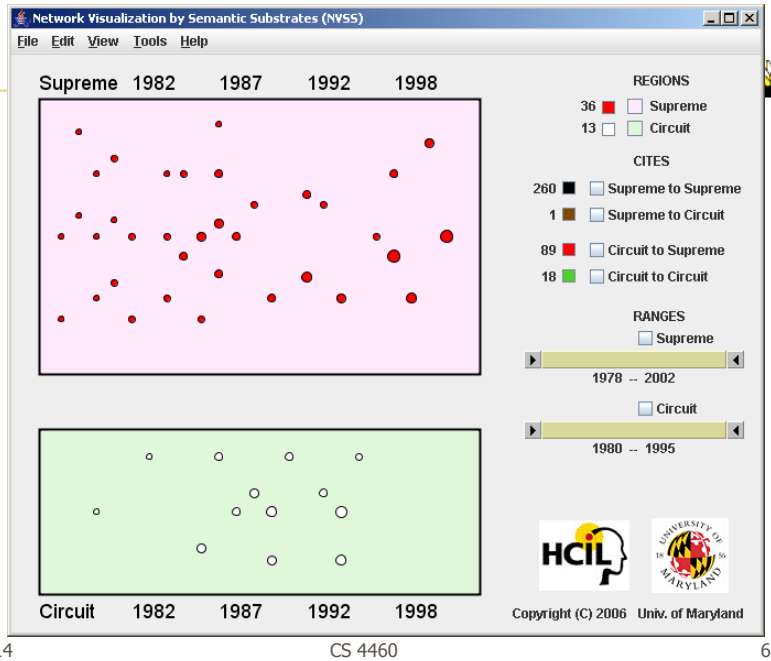
Figure 10. *Communication network of people in a large company. X-axis is division, y-axis is office geography. The division in the leftmost column has far more cross-location communication than the others.*

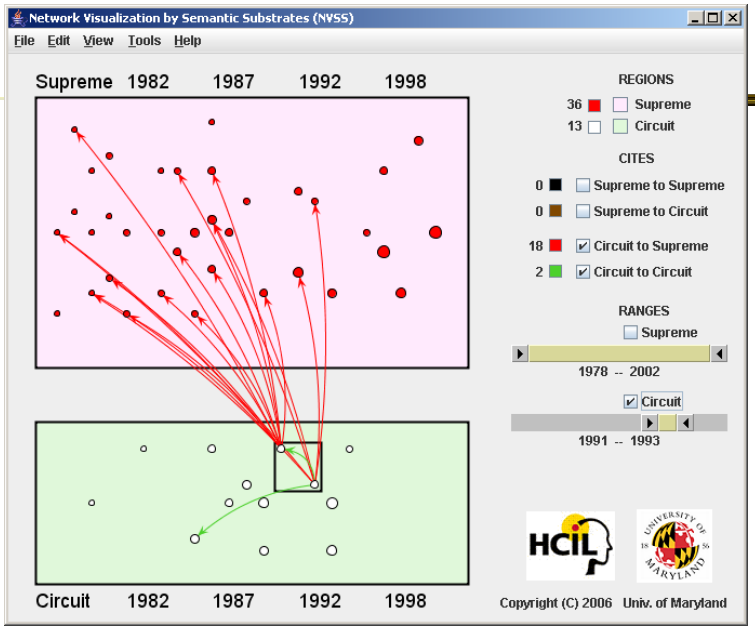
http://www.cs.umd.edu/hcil/nvss/

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

Shneiderman & Aris
TVCG (InfoVis) '06


Network Visualization by Semantic Substrates (NVSS)
File Edit View Tools Help
Supreme 1982 1987 1992 1998
Circuit 1982 1987 1992 1998
REGIONS
36 Supreme
13 Circuit
CITES
260 Supreme to Supreme
1 Supreme to Circuit
89 Circuit to Supreme
18 Circuit to Circuit
RANGES
Supreme
1978 -- 2002
Circuit
1980 -- 1995
HCIL
Copyright (C) 2006 Univ. of Maryland

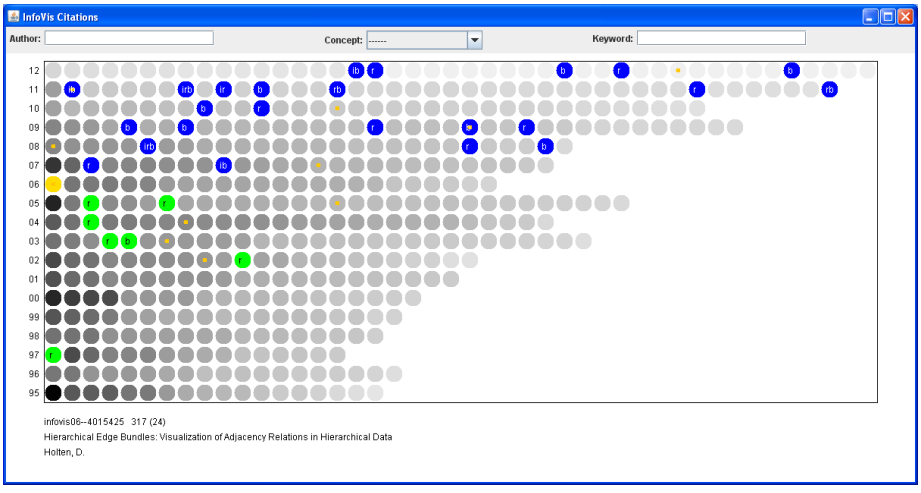

Network Visualization by Semantic Substrates (NVSS)
File Edit View Tools Help
Supreme 1982 1987 1992 1998
Circuit 1982 1987 1992 1998
REGIONS
36 Supreme
13 Circuit
CITES
0 Supreme to Supreme
0 Supreme to Circuit
18 Circuit to Supreme
2 Circuit to Circuit
RANGES
Supreme
1978 -- 2002
Circuit
1991 -- 1993
HCIL
Copyright (C) 2006 Univ. of Maryland

CiteVis

- Showing InfoVis Conference paper citation patterns
  - Papers are graph vertices
  - A cites B is graph edge
- Attribute-based layout
  - Year x Number of citations
- Uses color & interaction to show citations rather than drawn links

Stasko, Choo, Han, Hu, Pileggi, Sadana & Stolper
InfoVis poster '13

http://www.cc.gatech.edu/gvu/ii/citevis



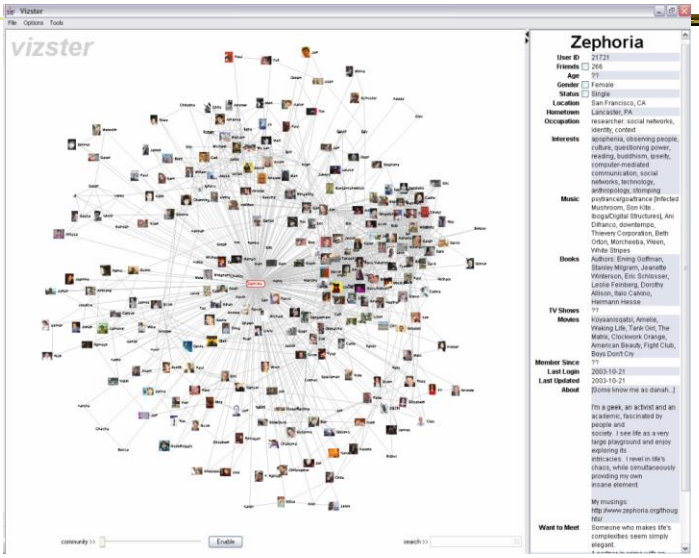
Vizster

- Visualize social networking sites like friendster, myspace, facebook
- Implementation
  - Crawled 1.5 million members (Winter 2003)
  - Written in Java using the *prefuse* tookit (http://prefuse.sourceforge.net)
- Oppose Shneiderman's mantra. Instead:

  "Start with what you know, then grow."

Heer & boyd
InfoVis '05

Visualization



Combining Features

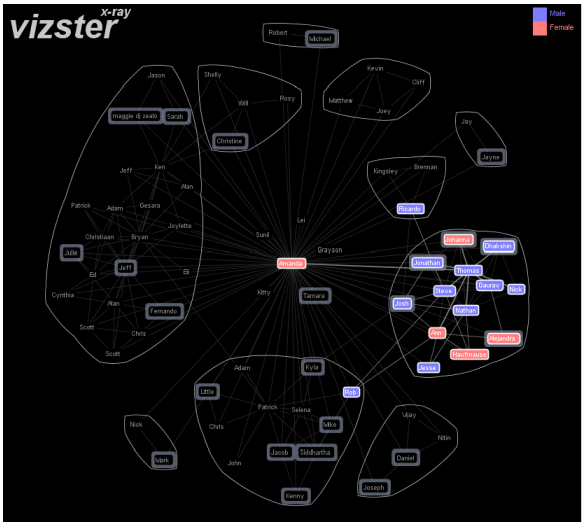
Video

Colors: Gender

Halo: Search for "student"

Highlight: Friends of selection

Blobs: Communities

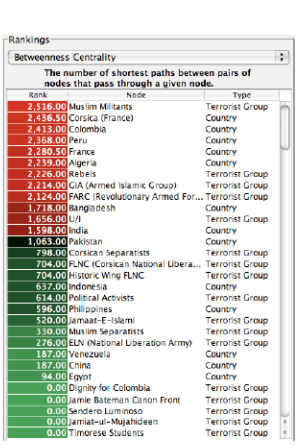


http://www.cs.umd.edu/hcil/socialaction/

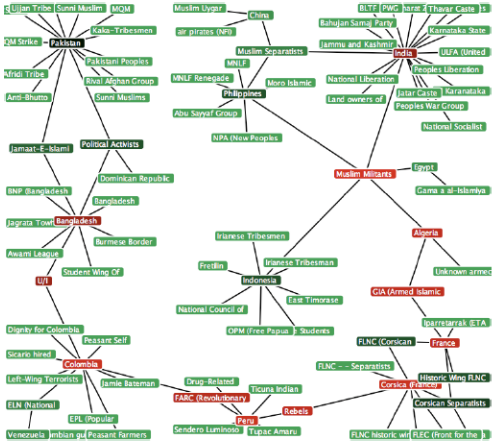
SocialAction

- Combines graph structural analysis (ranking) with interactive visual exploration
- Multiple coordinated views
  - Lists by ranking for analysis data
  - Basic force-directed layout for graph vis

Perer & Shneiderman
TVCG (InfoVis) '06



(a) Ordered list of 97 nodes in the largest connected component of the terrorism network in 1996. The nodes are ranked according to their betweenness centrality.



(b) Network visualization of the same 97 nodes, colored according to their ranking. The nodes with highest betweenness rankings, sometimes referred to as "gatekeepers", are painted red.

Figure 1.

Social Network Attributes

- **Bary center** – total shortest path of a node to all other nodes
- **Betweenness centrality** – how often a node appears on the shortest path between all other nodes
- **Closeness centrality** – how close a node is compared to all other nodes
- **Cut-points** – the subgraph becomes disconnected if the node is removed
- **Degree** – number of connections for node
- **HITs** – "hubs and authorities" measure
- **Power centrality** – how linked a node is to rest of network

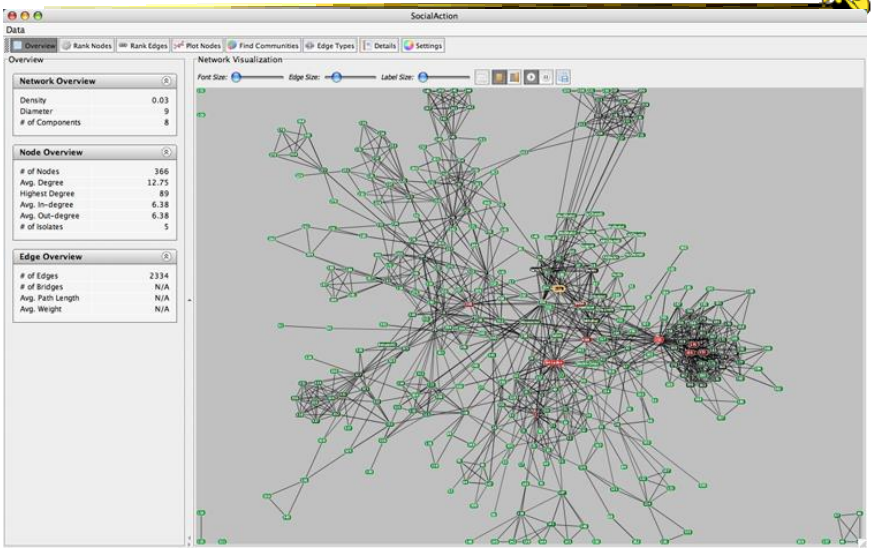
Attribute Ranking

- Run these measures on all nodes and rank them
- Sort the rankings and show in lists and scatterplots
- Allow user to filter based on rankings
- Can aggregate rankings for cohesive subgroups of nodes

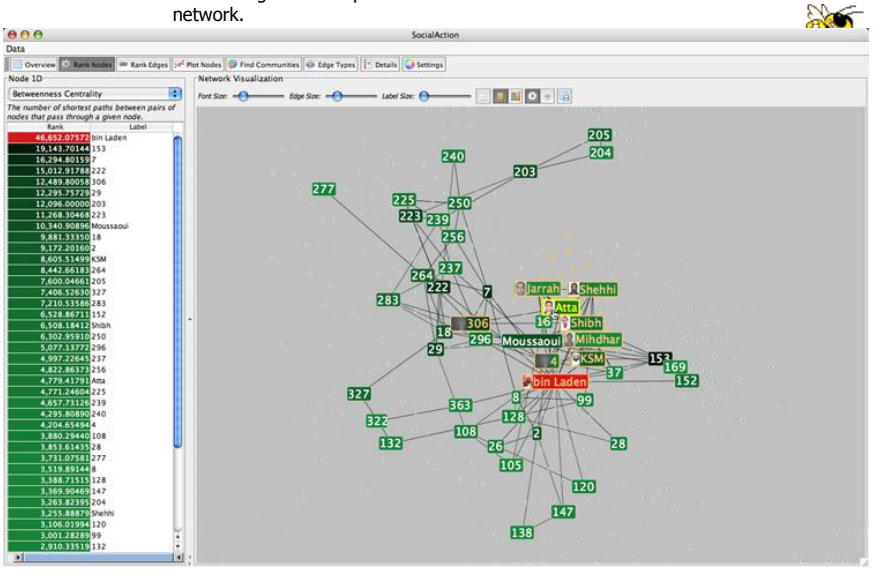
Graph Visualization

- Standard node-link
- Node positions remain constant across different metric views to promote comprehension
- Links can have types
- Coherent subgroups can be aggregated (like in Vizster)
  - Uses Newman's community identification algo

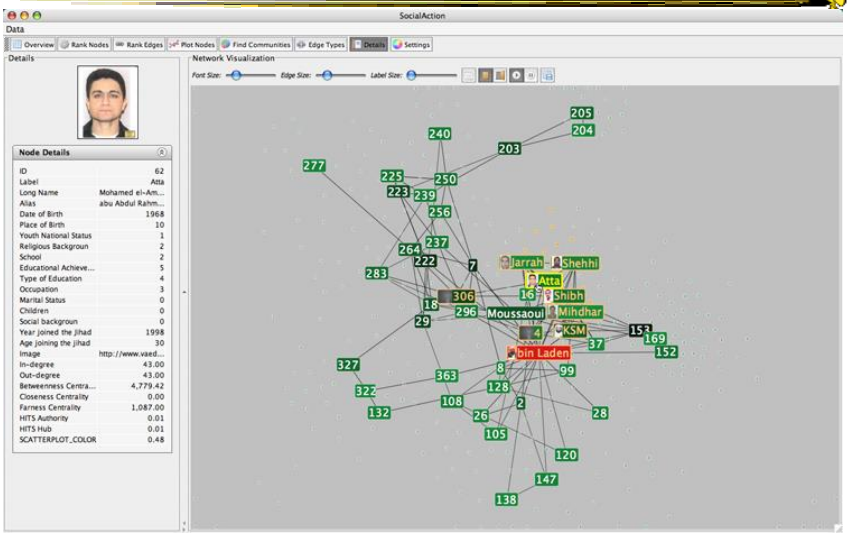
Users begin with an overview of the entire social network. On the left side, overview statistics that describe the overall structure are presented. On the right, the network is visualized using a force directed algorithm.



The gatekeepers are found using a statistical algorithm. Users filter out the unimportant nodes using a dynamic slider which simplifies the visualization while maintaining the node positions and structure of the network.

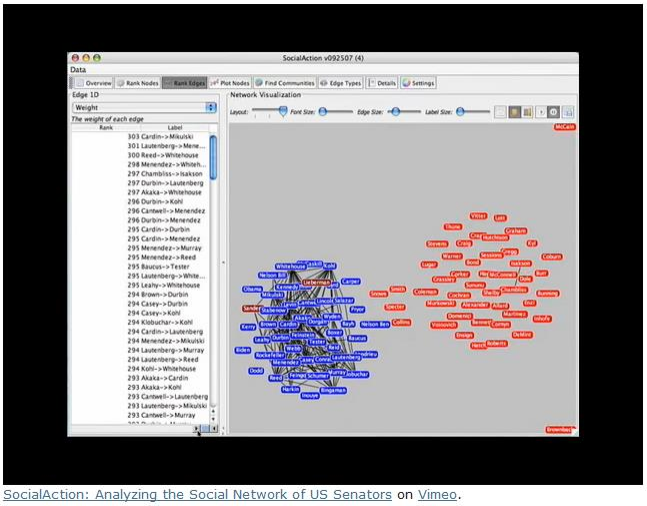


Labels are always given priority so users can understand what the data represents. When user selects a node, neighbors are highlighted and details appear on the left. In order to protect sensitive information, node labels have been anonymized except for those individuals publicly identified in the Zacarias Moussaoui trial.



http://www.cs.umd.edu/hcil/socialaction/

Senate Voting Patterns



Video

SocialAction: Analyzing the Social Network of US Senators on Vimeo.

Implementation

- Jung
  - Network data structures and algorithms
- Prefuse
  - Graph drawing
- Piccolo
  - Scatterplot and Matrix views

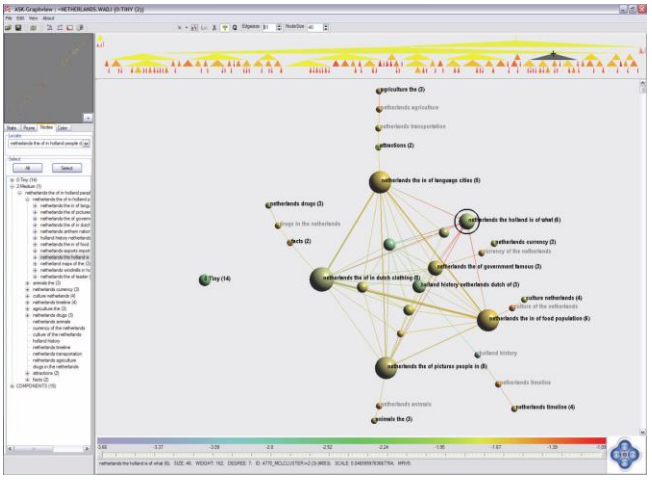
Comments

- One of my favorite recent InfoVis papers
- Not too innovative on the vis technique side, but wonderful application and synthesis of useful capabilities
- Actually, a very nice *visual analytics* example
- Good subsequent paper on case studies evaluation of it (on our later Eval day)

Really Big Graphs

- May be difficult to keep all in memory
- Often visualized as "hairballs"
- Smart visualizations do structural clustering, so you see a high-level overview of topology

ASK-GraphView



Uses clustering algorithms to construct a hierachy

Abello, van Ham & Krishnan
TVCG (InfoVis) '06

Alternate Big Graph Approach

- Show some of the details, rather than high level structure
- Allow users to focus on particular nodes
- Adapt DOI algorithm from trees to graphs
- Rely heavily on interaction

- Different paradigm: "Search, show context, expand on demand"

van Ham & Perer
TVCG (InfoVis) '09

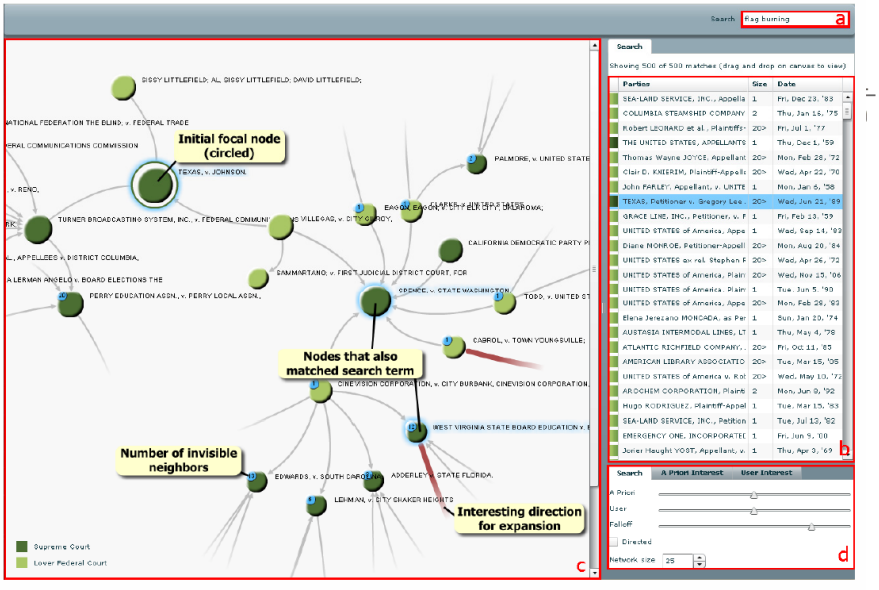


Fig. 3. Basic user interface layout. A user types a query in the searchbox (a) which yields a number of hits presented in tabular form (b). One of these hits can then be dragged to the main screen (c) which shows the subgraph centered on that node. Other nodes that matched the user's search are highlighted in blue. Users can adapt the balance between different components of the DOI function and the size of the subgraph in a separate panel (d).

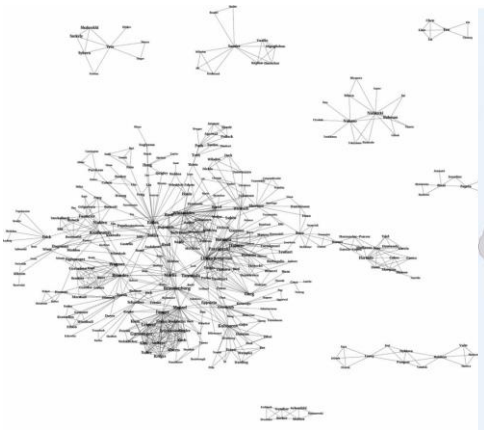
Graphs as Maps

- Represent a large graph as a map
- Maintain inherent structure and relationships between nodes
- Follow standard cartographic representations

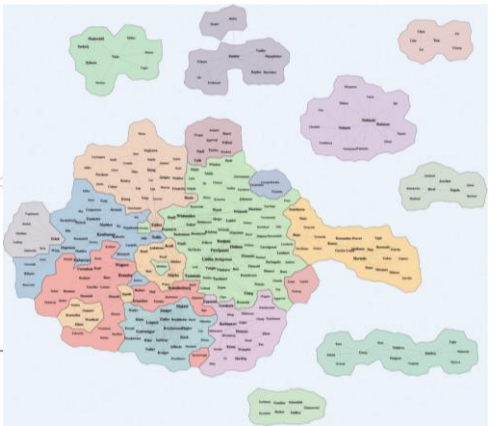
Gansner, Hu & Kobourov
IEEE CG&A (PacificVis) '10

http://www2.research.att.com/~yifanhu/MAPS/imap.html

Both Representations



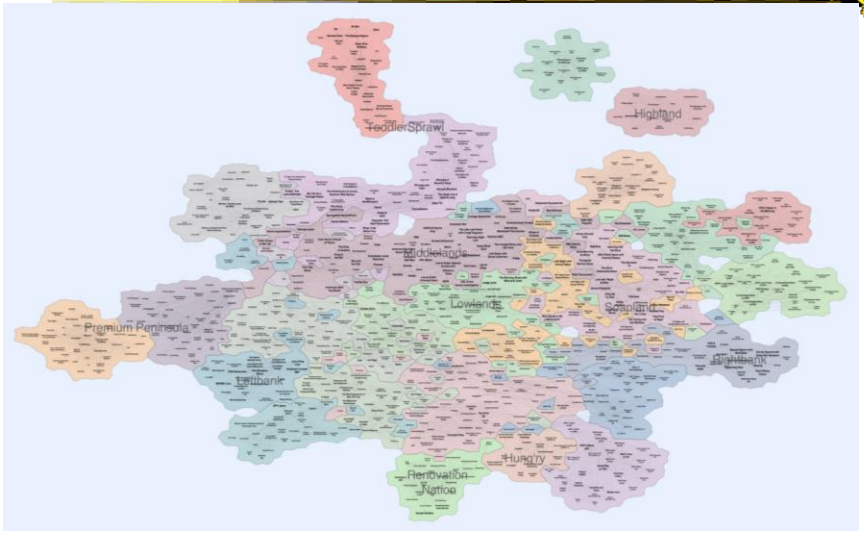
Node-link

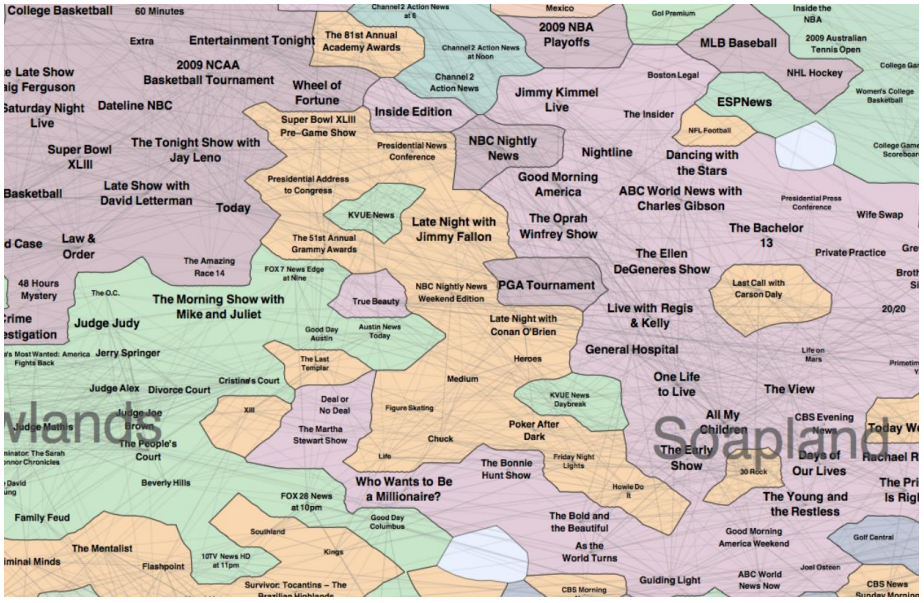


Map

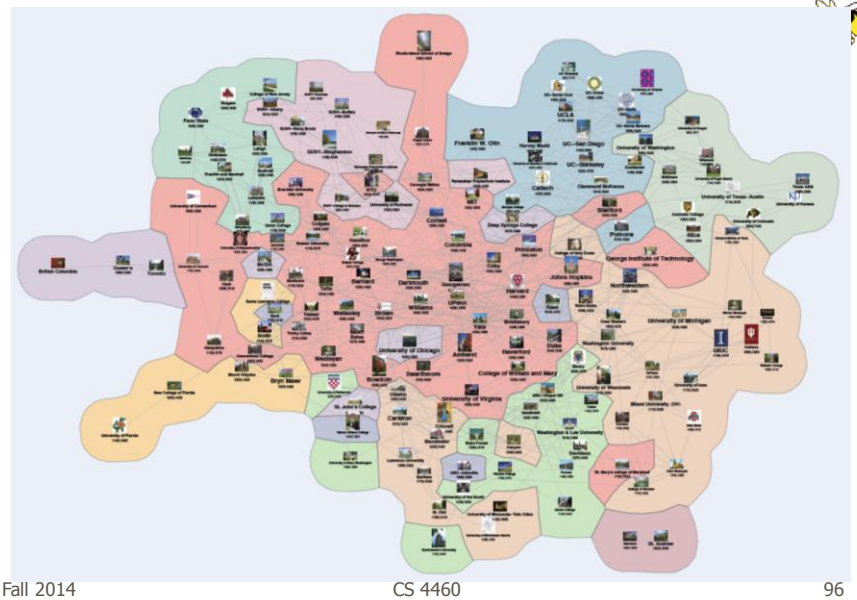
Music Graph/Map

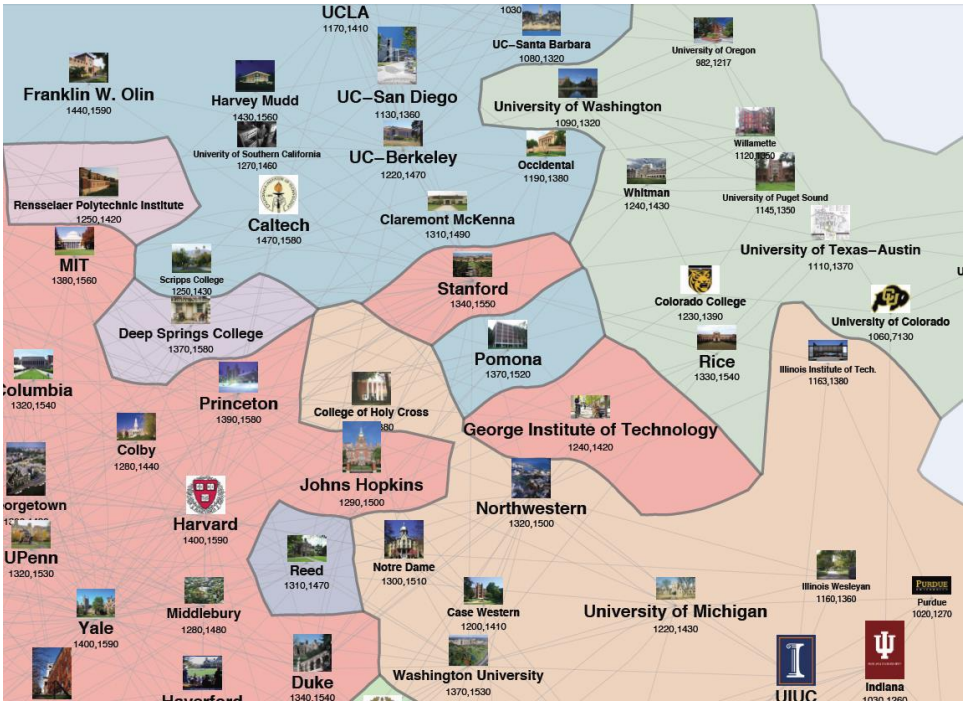
TV Shows




College Basketball
60 Minutes
Extra
Entertainment Tonight
The 81st Annual Academy Awards
2009 NBA Playoffs
MLB Baseball
2009 Australian Tennis Open
2009 NCAA Basketball Tournament
Wheel of Fortune
Channel 2 Action News
Jimmy Kimmel Live
Boston Legal
NHL Hockey
ESPNews
Dateline NBC
Super Bowl XLIII Pre-Game Show
Inside Edition
The Insider
Saturday Night Live
Super Bowl XLIII
The Tonight Show with Jay Leno
Presidential News Conference
NBC Nightly News
Nightline
Dancing with the Stars
Basketball
Late Show with David Letterman
Today
Presidential Address to Congress
Good Morning America
ABC World News with Charles Gibson
KVUE News
Late Night with Jimmy Fallon
The Oprah Winfrey Show
Wife Swap
The Bachelor 13
Law & Order
The 51st Annual Grammy Awards
The Ellen DeGeneres Show
Private Practice
The Amazing Race 14
FOX 7 News Edge at Nine
48 Hours Mystery
The O.C.
NBC Nightly News Weekend Edition
PGA Tournament
Last Call with Carson Daly
The Morning Show with Mike and Juliet
True Beauty
Live with Regis & Kelly
20/20
Judge Judy
Good Day Austin
Austin News Today
Late Night with Conan O'Brien
Jerry Springer
The Last Templar
Heroes
General Hospital
Life on Mars
Judge Alex
Divorce Court
Cristina's Court
Medium
One Life to Live
The View
Deal or No Deal
KVUE News Daybreak
All My Children
CBS Evening News
The People's Court
The Martha Stewart Show
Figure Skating
Poker After Dark
Chuck
The Early Show
Days of Our Lives
Beverly Hills
Life
The Bonnie Hunt Show
Friday Night Lights
Howie Do It
Who Wants to Be a Millionaire?
FOX 28 News at 10pm
The Young and the Restless
Family Feud
Good Day Columbus
The Bold and the Beautiful
Good Morning America Weekend
Golf Central
Southland
As the World Turns
The Mentalist
Kings
10TV News HD at 11pm
Flashpoint
Guiding Light
ABC World News Now
Joel Osteen
CBS News

Colleges





Drawing Graphs Better

- Can we do clever "tricks" to make dense graphs more readable?

Hierarchical Edge Bundles

- Bundle edges that go from/to similar nodes together
  - Like wires in a house
- Uses B-spline curves for edges
- Reduces the clutter from many edges

Holten
TVCG (InfoVis) '06

Example

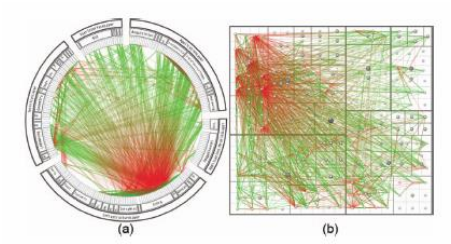


Fig. 11. A software system and its associated call graph (caller = green, callee = red). (a) and (b) show the system without bundling using a radial and a squarified treemap layout (node labels disabled), respectively. (a) and (b) mainly show hot spots; the actual connectivity information is more difficult to discern due to visual clutter.

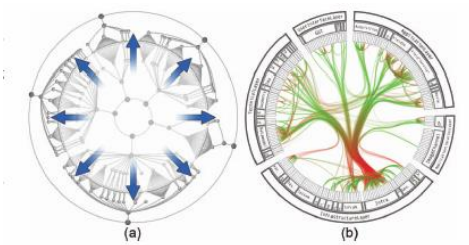


Fig. 12. Radial layout construction. (a) A radial tree layout is used for the inner circle and subsequently mirrored to the outside; (b) the inner layout is hidden and its structure is used to guide the adjacency edges. An icicle plot based on the mirrored layout is used to show the hierarchy.

Example

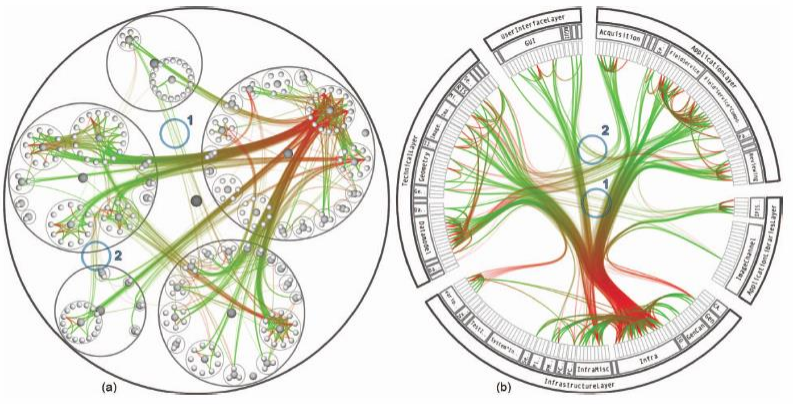
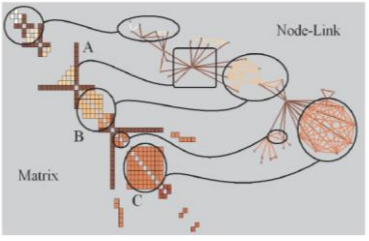


Fig. 13. A software system and its associated call graph (caller = green, callee = red). (a) and (b) show the system with bundling strength $\beta = 0.85$ using a balloon layout (node labels disabled) and a radial layout, respectively. Bundling reduces visual clutter, making it easier to perceive the actual connections than when compared to the non-bundled versions (figures 2a and 11a). Bundled visualizations also show relations between sparsely connected systems more clearly (encircled regions); these are almost completely obscured in the non-bundled versions. The encircled regions highlight identical parts of the system for (a), (b), and figure 15.

Matrix Representations

- There has been renewed interest in matrix representations of graphs recently
- I think the regularity, symmetry, and structure of a matrix are a win – people understand them well, but they don't scale up really well



MatrixExplorer

- Provides matrix view in combination with node-link and various operations for gaining different perspectives

Fig. 1. MatrixExplorer showing two synchronized representations of the same network: matrix on the left and node-link on the right.

Node Reordering

Extremely important operation with matrix representations

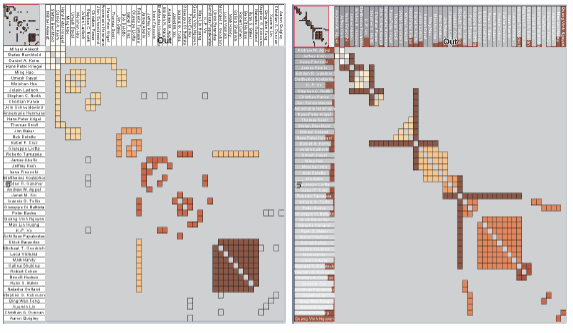
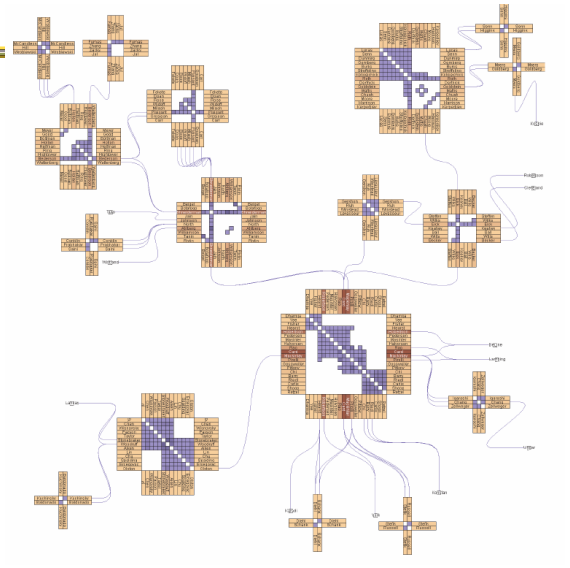


Fig. 6. Initial order (left) and TSP order (right). Colors represent clusters found by the user. Clusters are different in the two representations. Users found more clusters with TSP order. Headers red indicators (right) represents the distance between adjacent rows/columns.

NodeTrix

Hybrid of matrix
and node-link



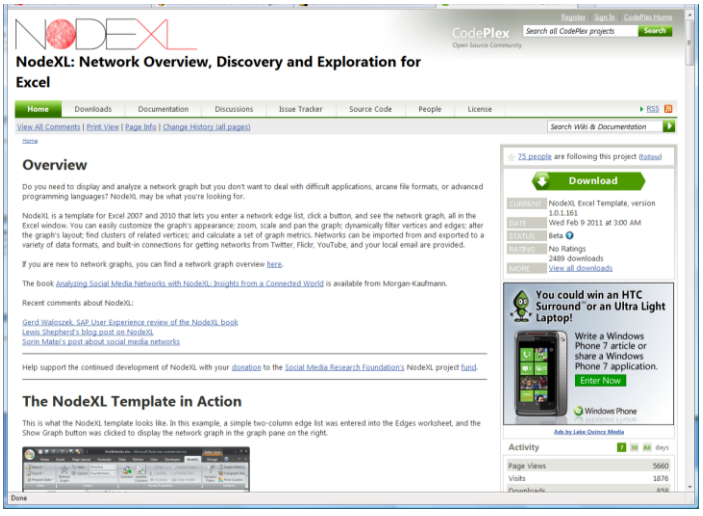
Henry & Fekete
TVCG (InfoVis) '07

Simplifying Input

- Make it easier to input graphs and then explore them

NodeXL

http://nodexl.codeplex.com/



Characteristics

- Plug-in for MS Excel
- Includes many network layout and network analysis metrics
- Data import:
  - List out vertices and edges in Excel columns
  - Native importers for email, Twitter, YouTube, etc.

Smith et al
C&T '09

Non-Network Data?

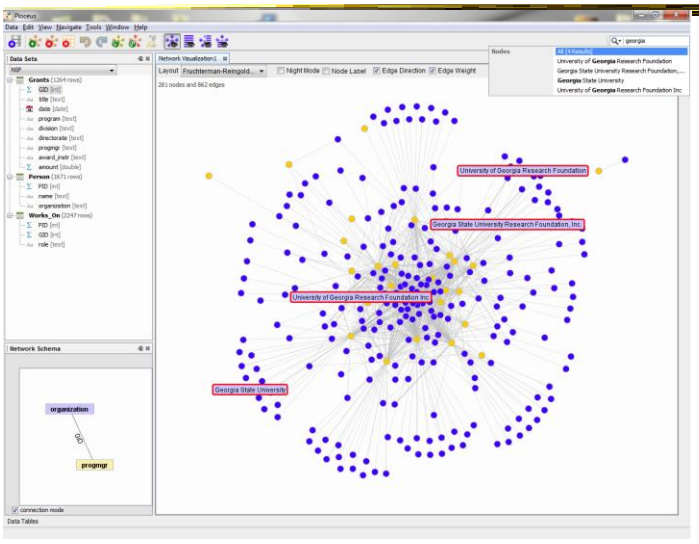
- But what if you don't have vertex-edge data to begin?
  - May just have tabular data from spreadsheet or database
- Still may want to explore data modeled as a graph
  - Consider DB of NSF grants (PIs, institution, PM, amount, …)
  - Look for clusters, patterns, connections, …

Ploceus

Liu, Navathe, Stasko
VAST '11, *Information Visualization* '14

- Framework and system for modeling and visualizing tabular data as network
- Allow user to model data as graph interactively through direct manipulation
  - What are vertices, edges, edge weights, …
- Visualizes graph on-the-fly (different layouts and network metrics)
- Advanced ops (project, aggregate, slice-n-dice) can be specified interactively too

Ploceus



Video

Graph Drawing Support

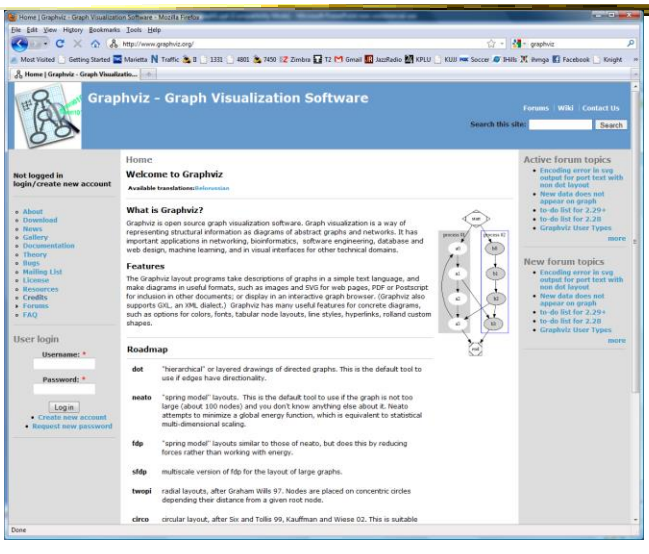
- Libraries
  - JUNG (Java Universal Network/Graph Framework)
  - Graphviz (formerly dot?)
- Systems
  - Gephi
  - TouchGraph

http://jung.sourceforge.net/

JUNG

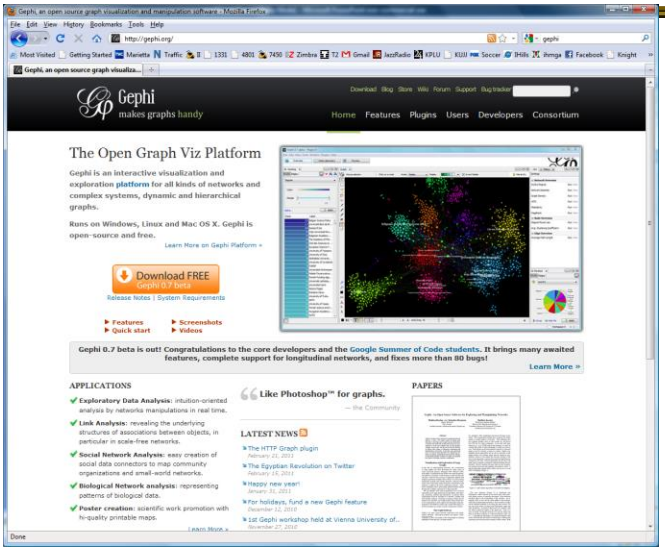
http://www.graphviz.org

Graphviz



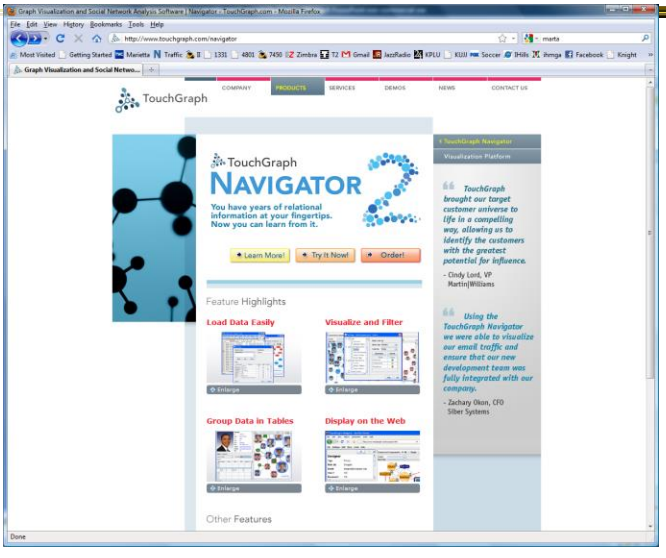
http://gephi.org

Gephi



http://www.touchgraph.com/navigator

TouchGraph

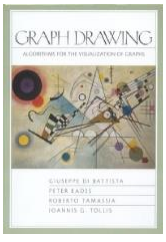


Graph Visualization Resource

- Very nice overview & survey
  - Herman et al, *IEEE TVCG* '00
  - but a little dated now

Graph Drawing Resources

- Book
  - diBattista, Eades, Tamassia, and Tollis, *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice Hall, 1999



- Tutorial (talk slides)
  - http://www.cs.brown.edu/people/rt/papers/gd-tutorial/gd-constraints.pdf
- Web links
  - http://graphdrawing.org

Upcoming

- Hierarchies and Trees
  - Reading
    Munzner chapter 9
- Text and Documents 1
  - Reading

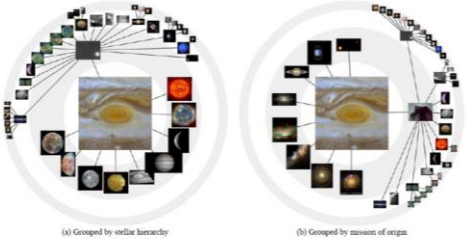
Additional Material

MoireGraph

- Uses radial layout not terribly unlike hyperbolic tree, but no hyperbolic geometry
- Impose levels on graph by doing min span tree from some node
- Put root at center, nodes at subsequent levels further out radially, with descreasing space for each
- Interaction is key

Jankun-Kelly & Ma
InfoVis '03

Sample Views



(a) Grouped by stellar hierarchy (b) Grouped by mission of origin

Figure 6: Image databases such as the NASA Planetary Photojournal posses several metadata attributes that group the same images differently depending on the metadata used. A MoireGraph can animate between displays of the different visual node graphs. In this image, only spanning tree edges are shown.

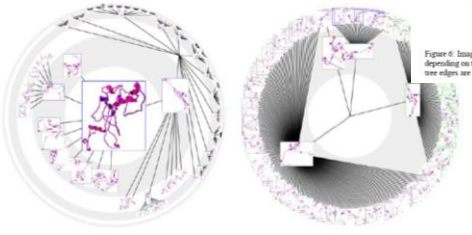
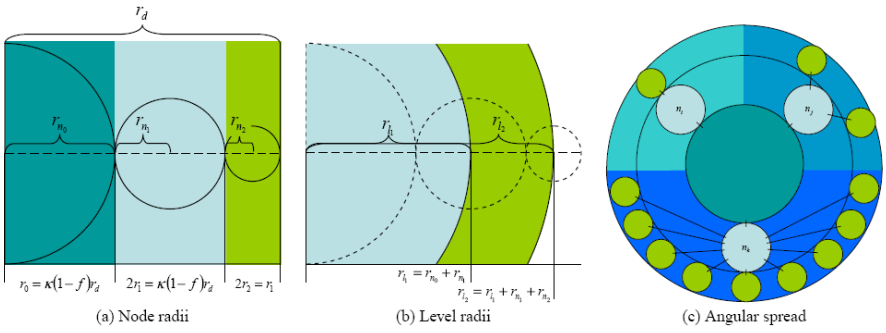


Figure 7: MoireGraphs for 284 configurations of proteins from an optimization process. the node without an image is a dummy node connecting the forest of configuration trees. As the optimizer progresses, more configurations are generated from a candidate configuration forming the configuration tree on the left. The right tree partitions the proteins into the current optimization status which is also encoded by their colored outline (right image): waiting (green), active (red), retired (blue), or culled (black).

Layout Principles

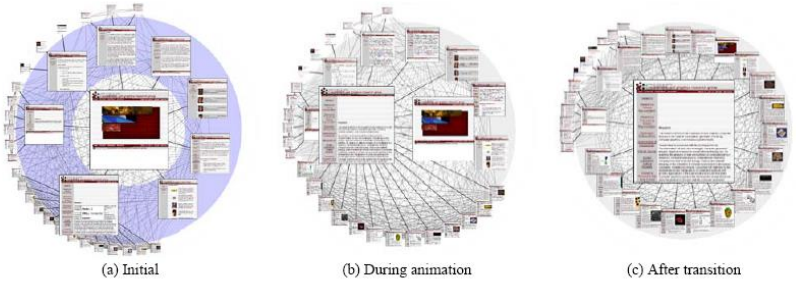


(a) Node radii (b) Level radii (c) Angular spread

Decreasing exponential space for outer layers

Spreading the "children" nodes

Navigation and interaction…



(a) Initial (b) During animation (c) After transition

Figure 5: Animated Navigation. Selecting a node in a MoireGraph changes the focus. The angular coordinates of a node and the node's size are interpolated during the animation.

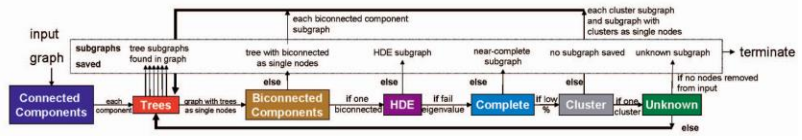
Video

Case Study

- NicheWorks
  - Interactive Visualization of Very Large Graphs
    Graham Wills
    Lucent (at that time)

TopoLayout

- Topological features are detected recursively inside a graph
- Their subgraphs are collapsed into single nodes, forming a hierarchy
- Each feature drawn with an algorithm tuned for its topology



Archambault, Munzner, & Auber
TVCG '07

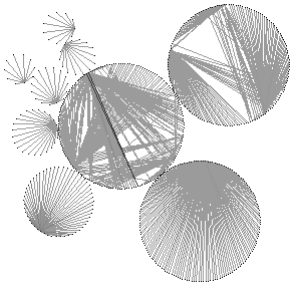
Big Graphs

- 20,000 - 1,000,000 Nodes
- Works well with 50,000
- Projects
  - Software Engineering
  - Web site analysis
  - Large database correlation
  - Telephone fraud detection

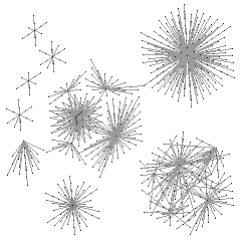
Features

- Typical interactive operations
- Sophisticated graph layout algorithm
  - 3 Layouts
    - Circular
    - Hexagonal
    - Tree
  - 3 Incremental Algorithms
    - Steepest Descent
    - Swapping
    - Repelling

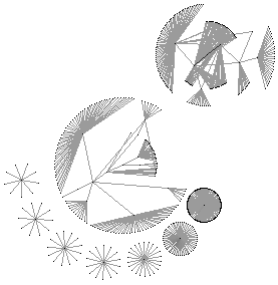
Web Site Example



Circle layout

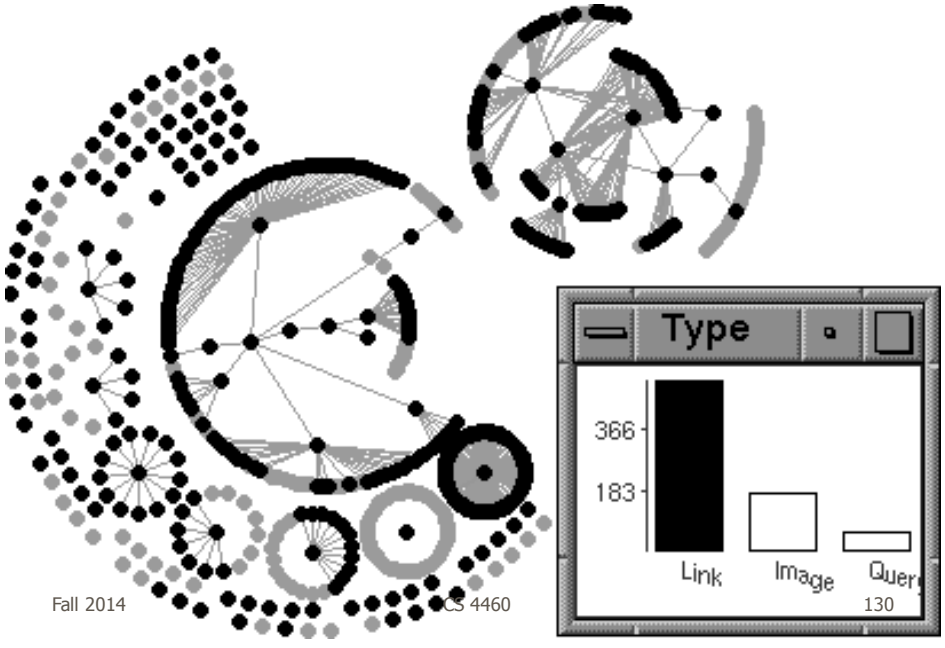


Hexagonal layout

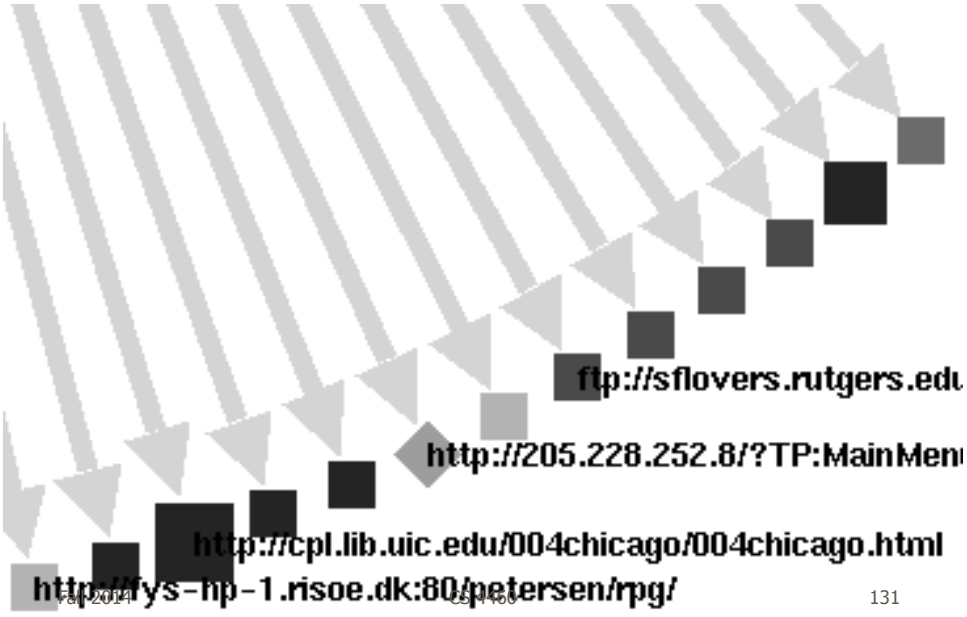


Tree layout

Interface

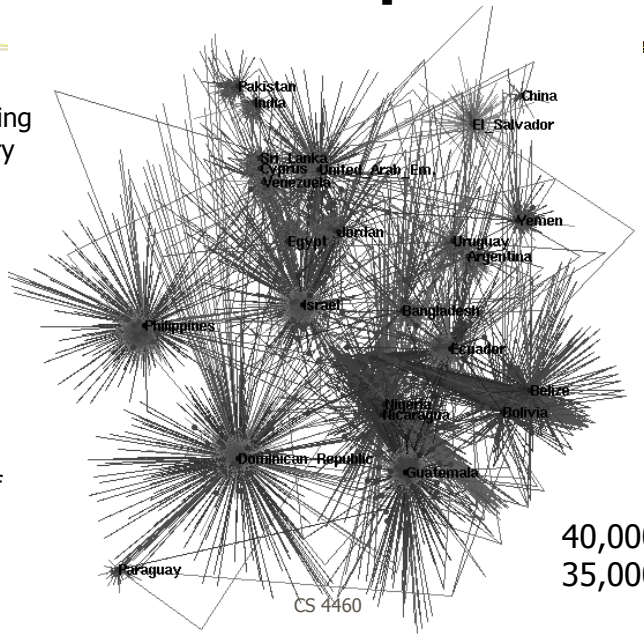


Interface



Phone Fraud Example

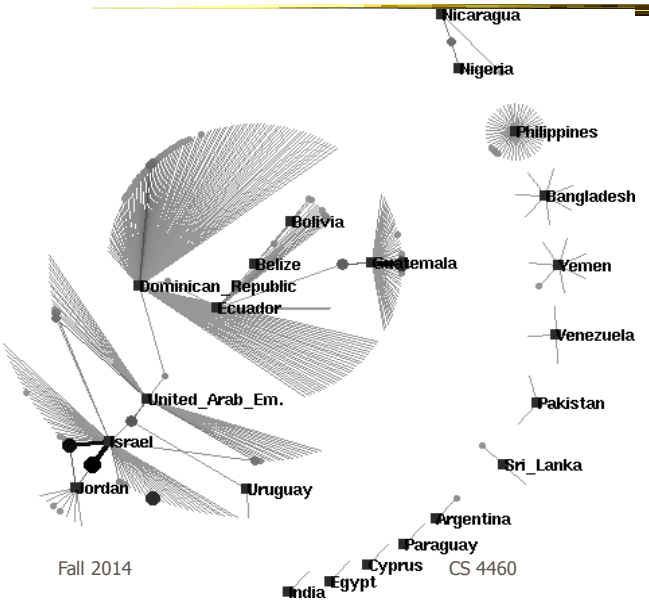
Shown are people calling that country



Length of edge is duration of call

40,000 calls
35,000 callers

Fraud Example

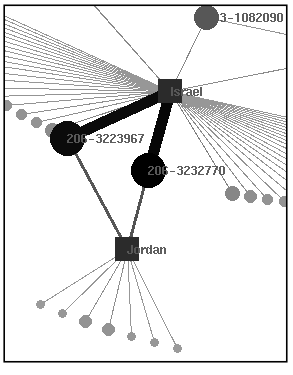


Filtering for people who made multiple calls and spent a significant amount of time on the phone

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

Note the two people calling Israel and Jordan

Fraud Example



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

More Neat Stuff

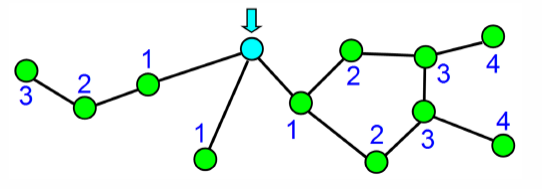
- http://willsfamily.org/gwills/
- Lots of interesting application areas
- More details on NicheWorks

PNNL's Graph Vis Work

- Graph Signatures
- Goal is to characterize the different styles of nodes in graph based on their local connectivity patterns

Wong et al
TVCG '06

Compute Signature



1. Run BFS from each node
2. Count how many nodes are 1, 2, 3,… steps away
   That is node's signature (3-d is recommended)
3. DO MDS to project into 2D scatterplot
4. Run k-means to detect different clusters. (9 is recommended)

(3, 3, 3, 2)
4-d signature of selected node

Example 1

1

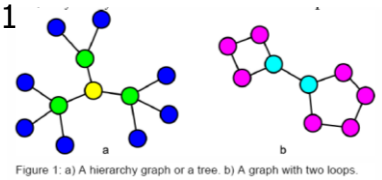


Figure 1: a) A hierarchy graph or a tree. b) A graph with two loops.

2

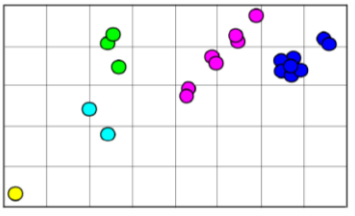


Figure 4: A scatterplot generated by projecting the 2-degree node signatures that represent the local topology of graph nodes in Figure 1 onto a 2D space.

3

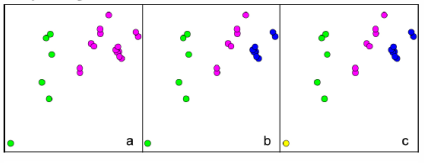


Figure 5: Results of K-mean on the scatterplot shown in Figure 4 where k = a) 2, b) 3, and c) 4.

Example 2

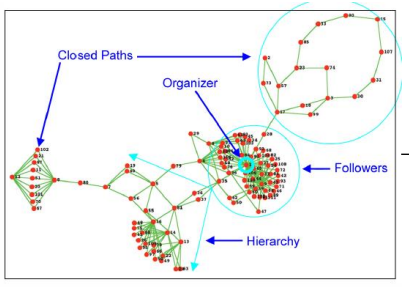


Figure 6: A force-directed layout of GD96B.

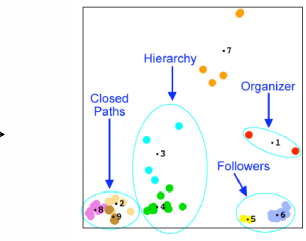


Figure 7: A 2D scatterplot generated by classical MDS using the signature vectors extracted from GD96B.

Example 2

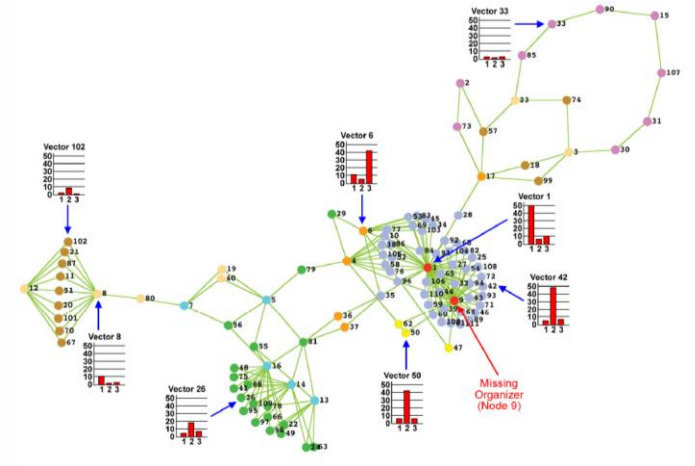


Figure 8: Nodes that share the same color belong to the same cluster identified in Figure 7. Eight signatures (represented as bar graphs) are selected to highlight the general topology of the seven clusters. Notice the previously missing organizer (node 9 in red) hidden among a sea of followers in Figure 6.