Topic Notes

Graphs and Networks 1

CS 7450 - Information Visualization
March 8, 2011
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?

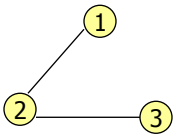
- Vertices (nodes) connected by
- Edges (links)

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



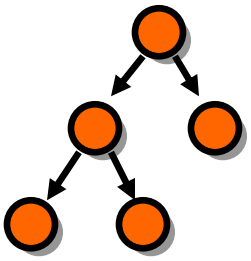
Drawing

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

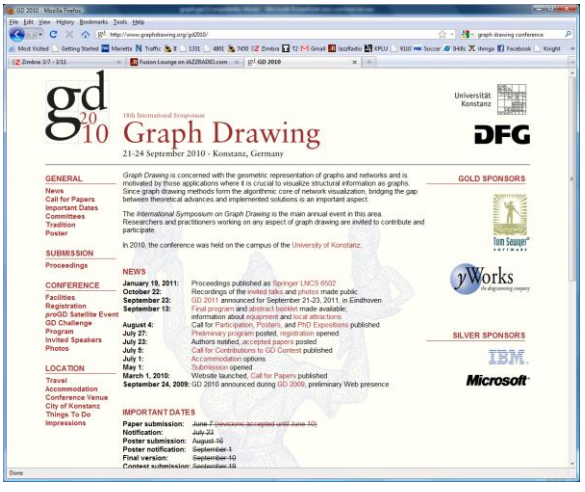
- Homework assignment
- Let's judge!

Results

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

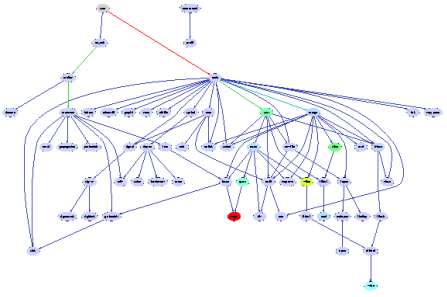
Layout Algorithms

Entire research community's focus



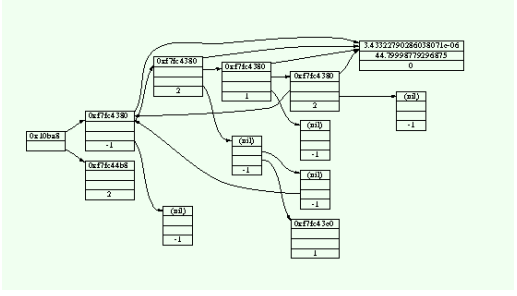
Vertex Issues

- 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

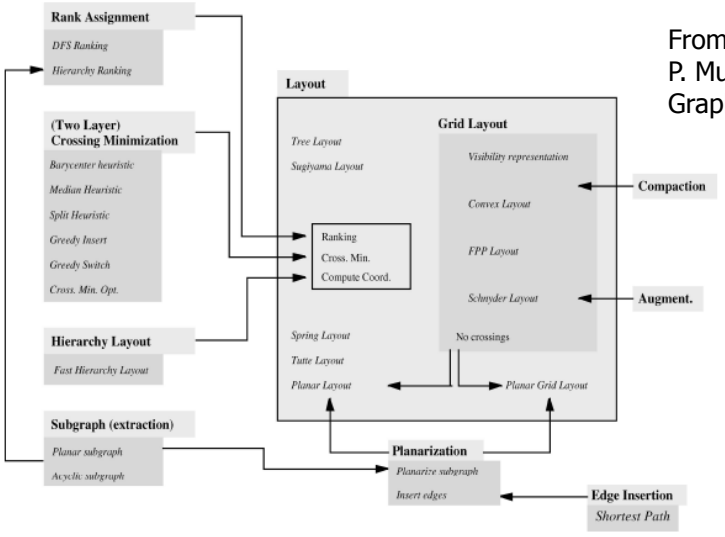
3) For every link you can follow it from source to destination

4) Clusters and outliers are identifiable

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

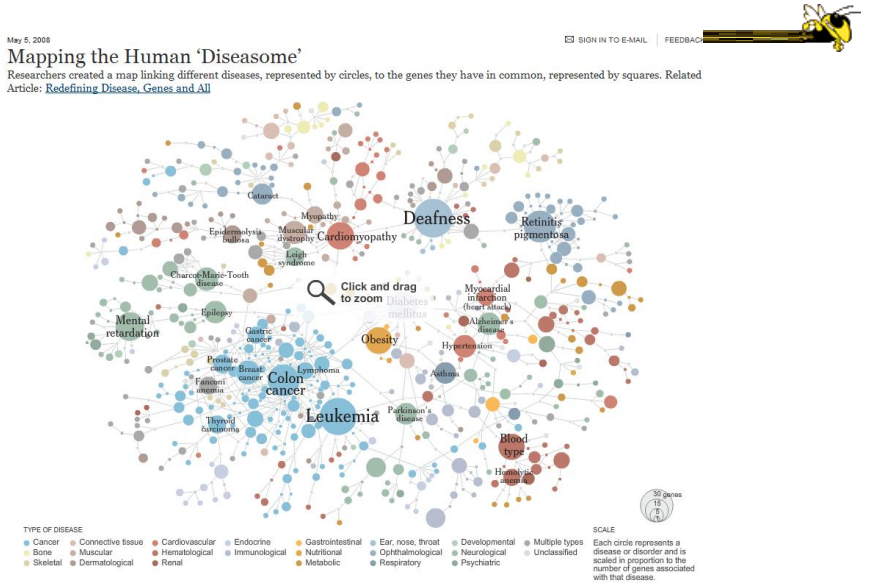
- How do we allow a user to query, visit, or move around a graph?
- Changing focus may entail a different rendering

Graph Drawing Uses

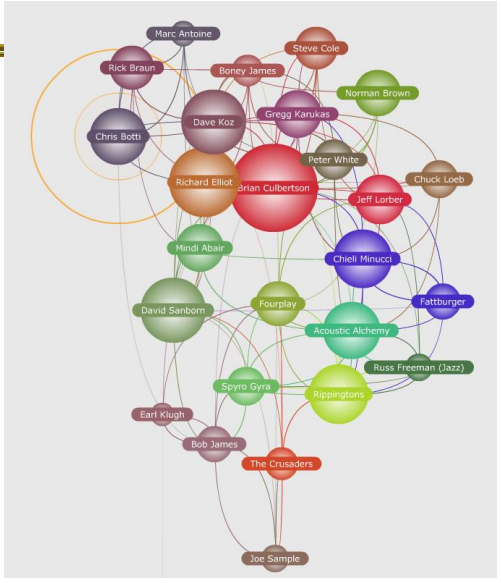
- Many domains and data sets can benefit significantly from nice graph drawings
- Let's look at some examples…

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

Human Diseases



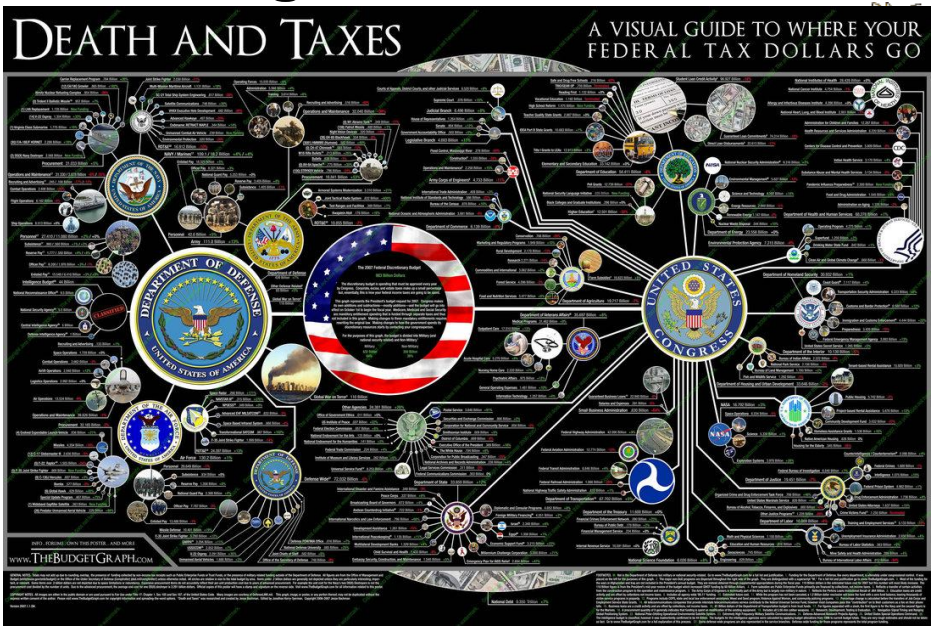
Music Artists



http://www.liveplasma.com/

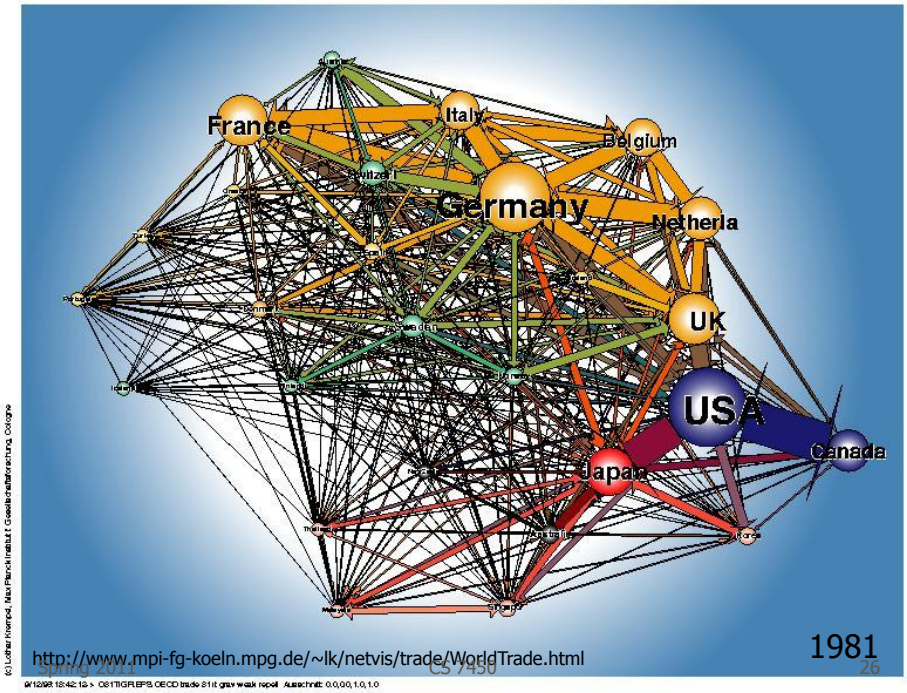
http://mibi.deviantart.com/art/Death-and-Taxes-2007-39894058

US Budget

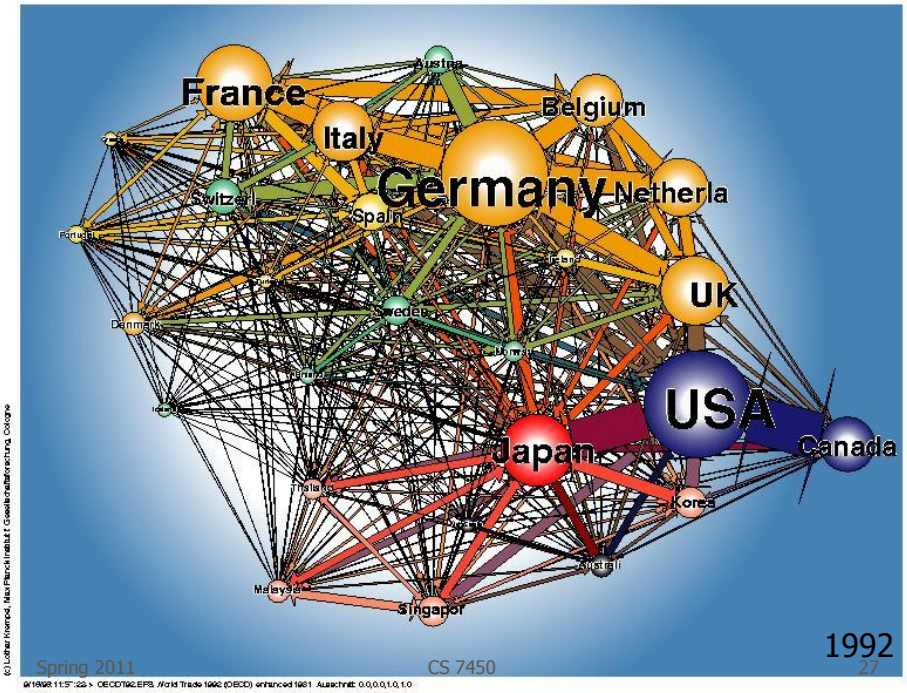


Social Analysis

- 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

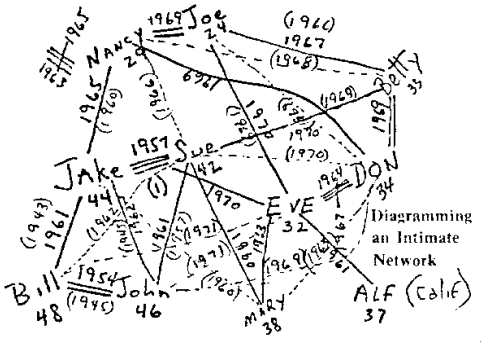


http://www.mpi-fg-koeln.mpg.de/~lk/netvis/trade/WorldTrade.html



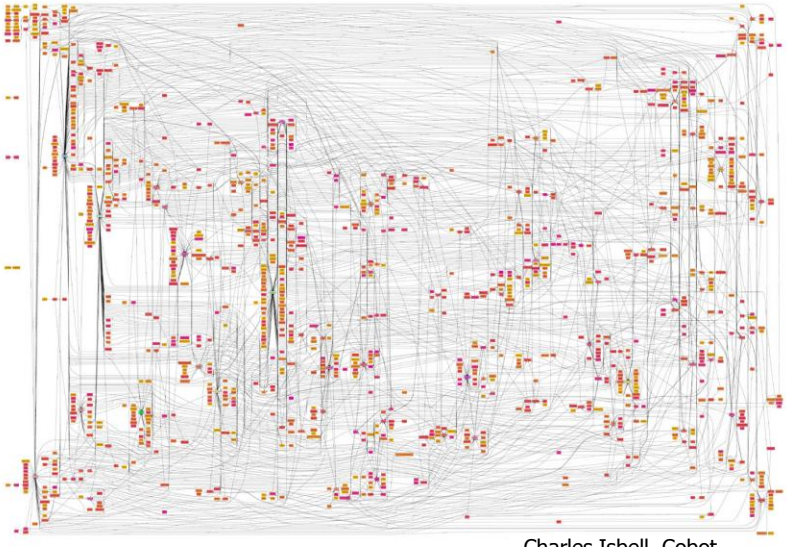
Social Network Visualization

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



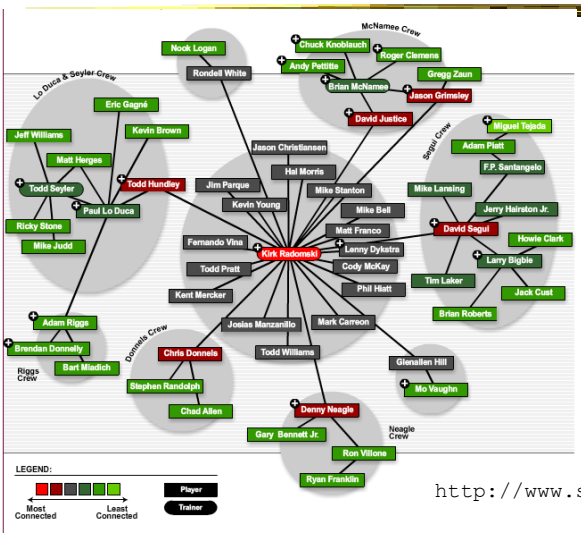
Hot topic again
Why?
Terrorists
Facebook

People connections



Charles Isbell, Cobot

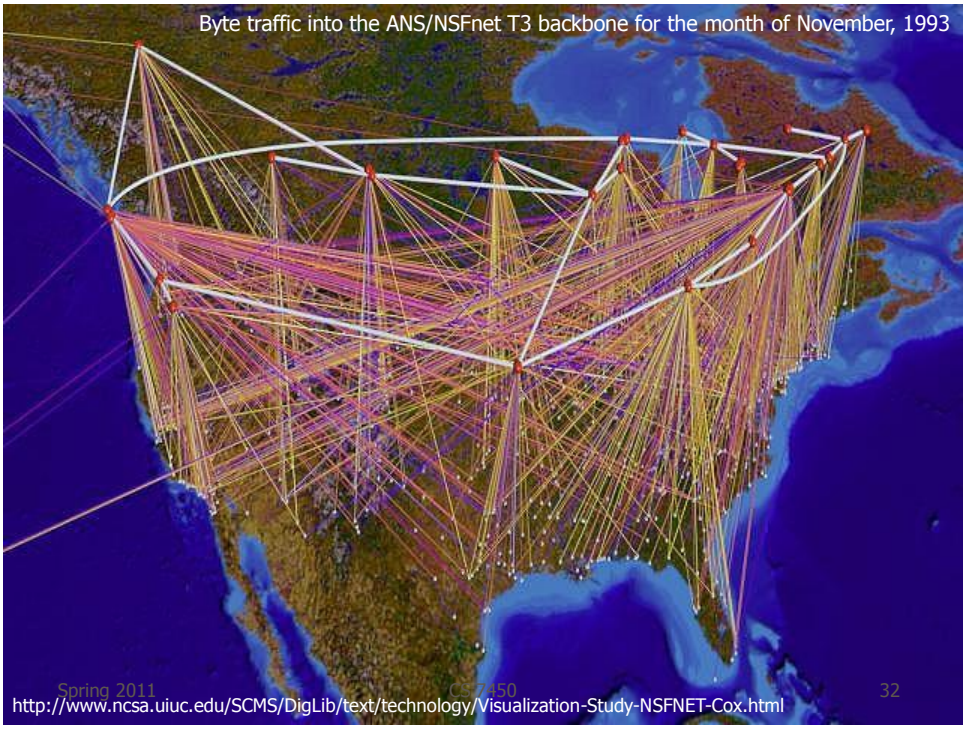
Steroids in MLB



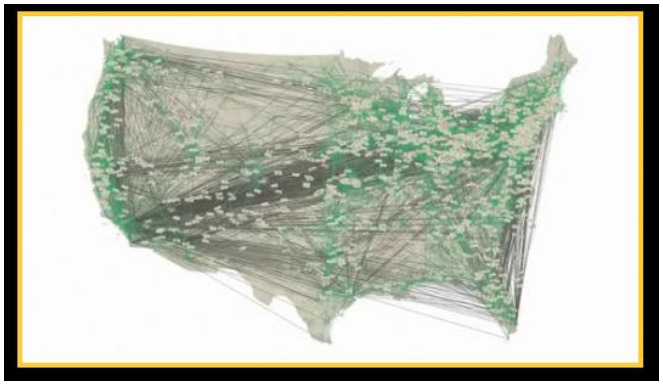
http://www.slate.com/id/2180392/

Geo Applications

- Many problems and data sets have some geographic correspondence



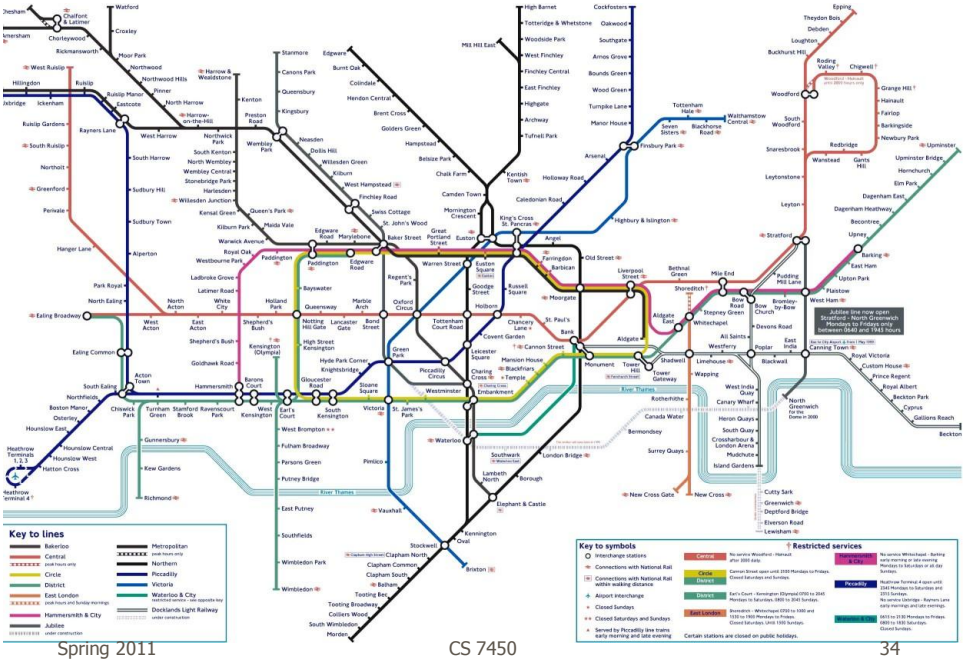
Follow the Money

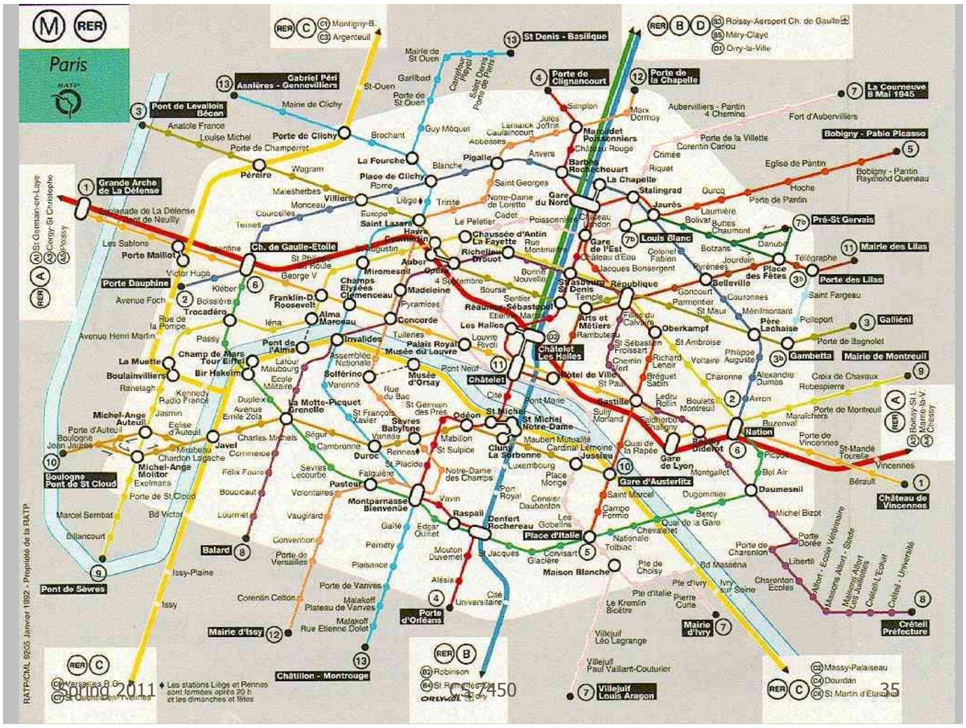


Where does a dollar bill go?

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

London Tube Harry Beck





Atlanta MARTA

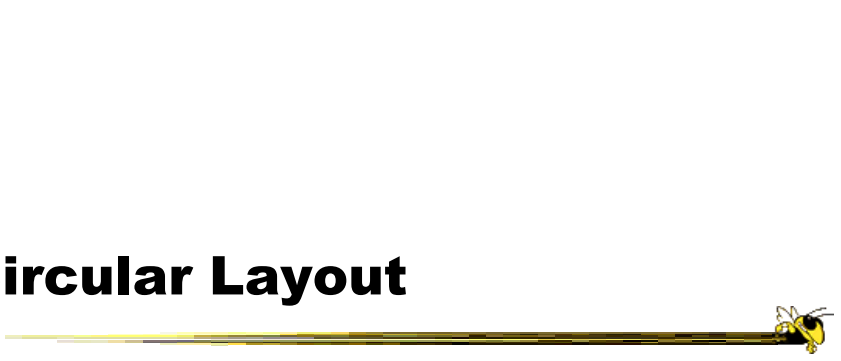
3 Subway Diagrams

- 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?

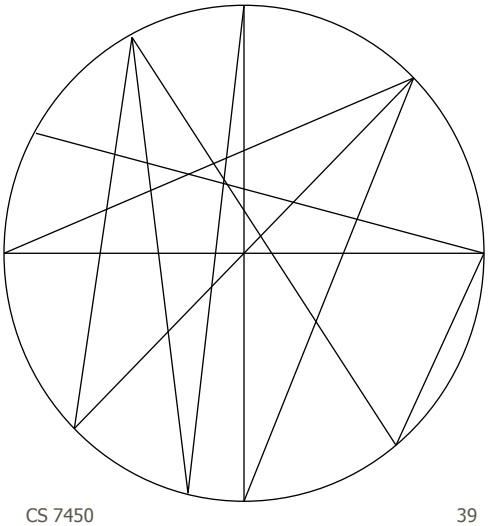
- I generally don't consider a pure graph layout (drawing) algorithm to be InfoVis
  - Nothing wrong with that, just an issue of focus
- For InfoVis, I like to see some kind of interaction or a system or an application…
  - Still, understanding the layout algorithms is very important for infovis
  -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



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

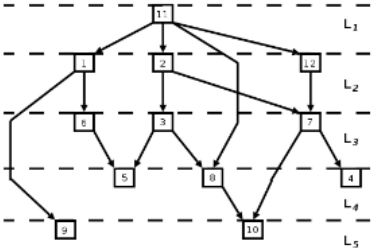


Figure: A graph showing a layered layout, created with the Sugiyama heuristic, with the layers shown. The bends in the edges correspond to dummy nodes.

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

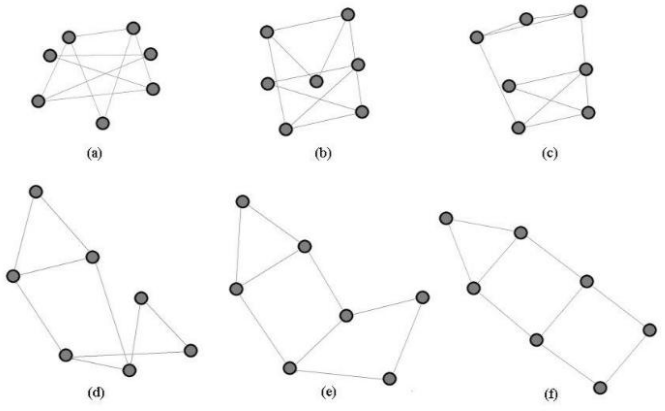
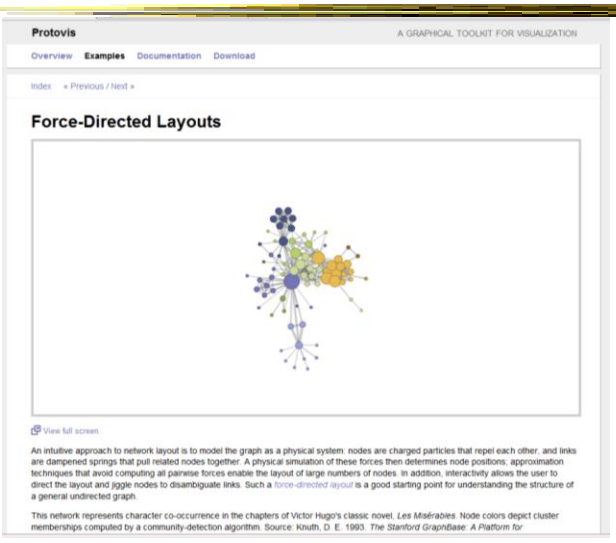


Figure 2: A graph drawing through a number of iterations of a force directed algorithm.

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

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

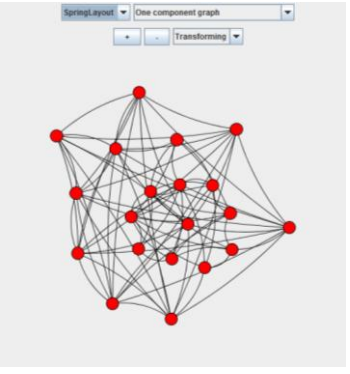
In Action



Variant

Images from JUNG

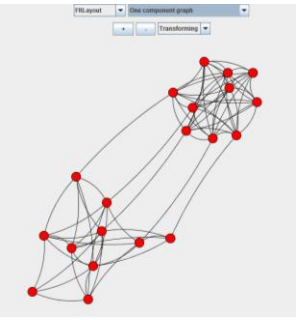
- Spring layout
  - Simple force-directed spring embedder



Variant

Images from JUNG

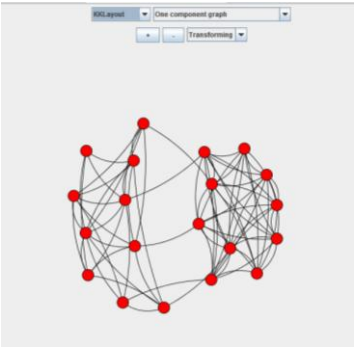
- Fruchterman-Reingold Algorithm
  - Add global temperature
  - If hot, nodes move farther each step
  - If cool, smaller movements
  - Generally cools over time



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

Case Study

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

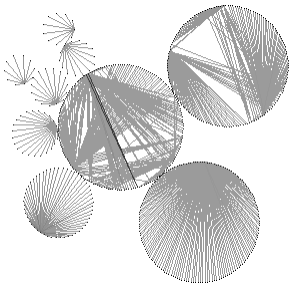
Big Graphs

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

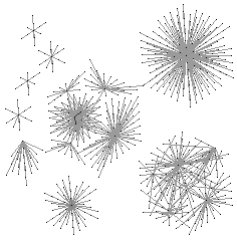
Features

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

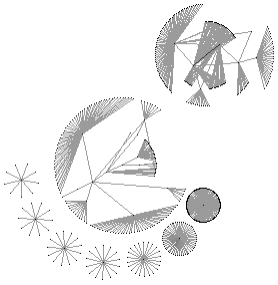
Web Site Example



Circle layout

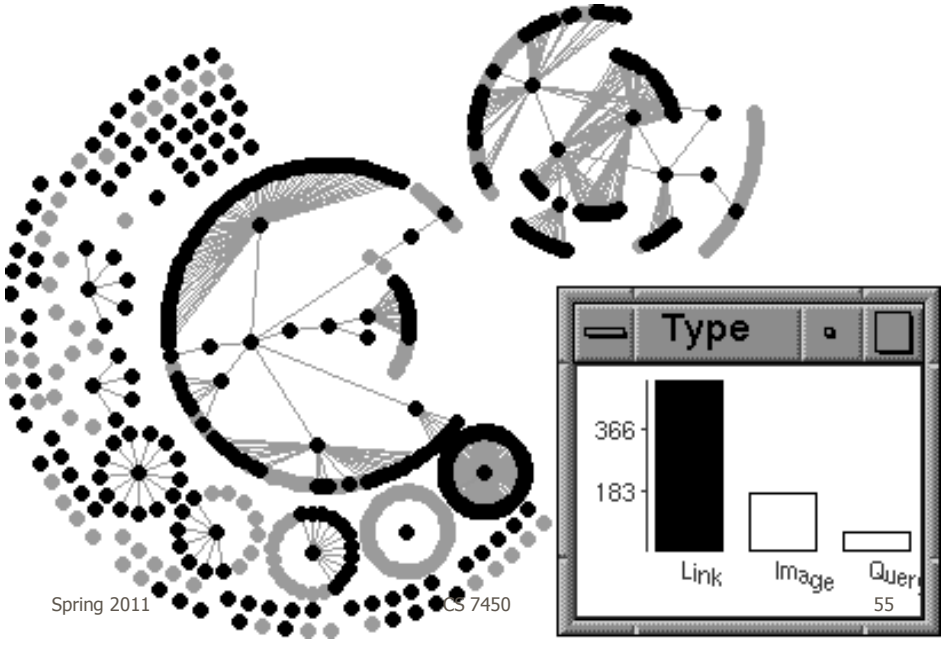


Hexagonal layout

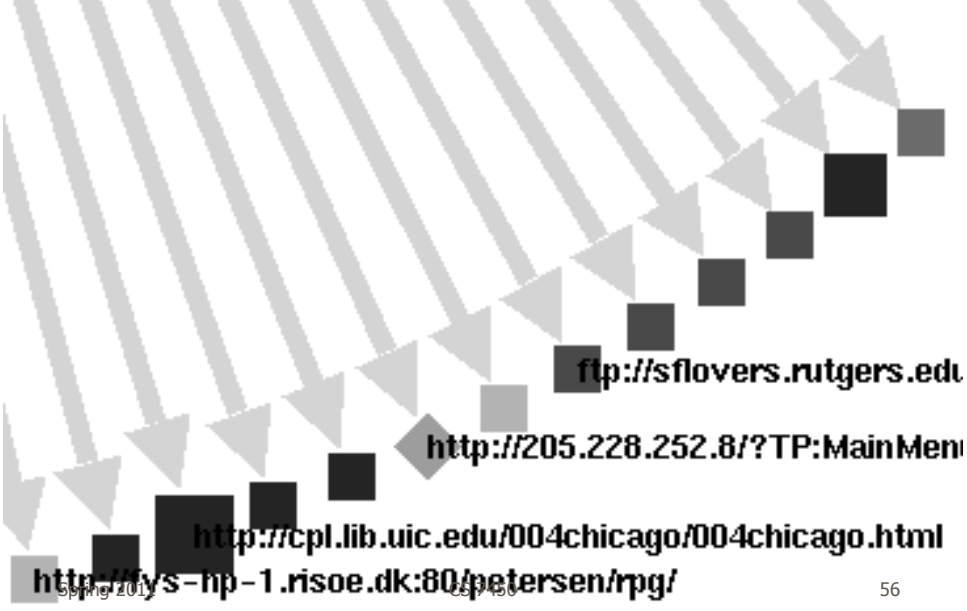


Tree layout

Interface

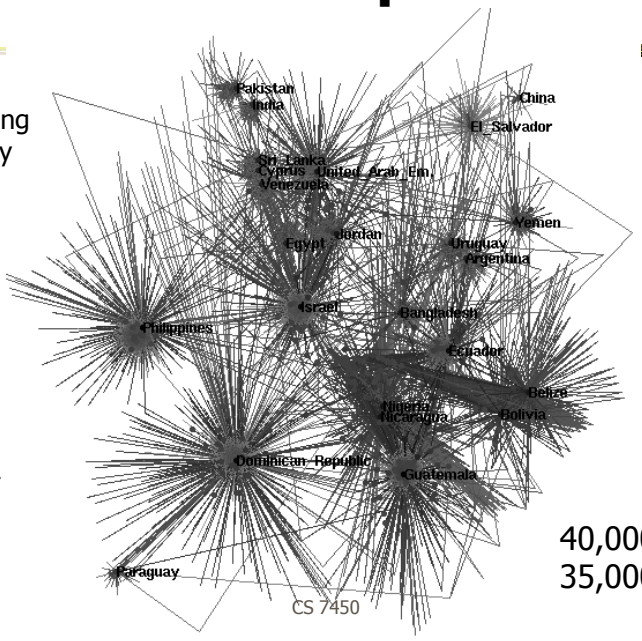


Interface



Phone Fraud Example

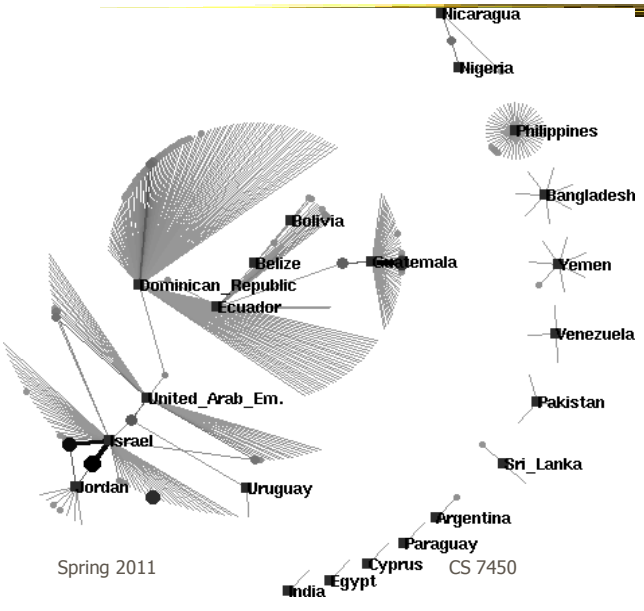
Shown are people calling that country



Length of edge is duration of call

40,000 calls
35,000 callers

Fraud Example

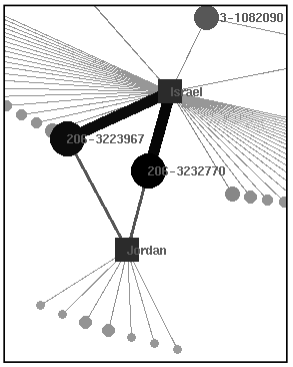


Filtering for people who made multiple calls and spent a significant amount of time on the phone

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

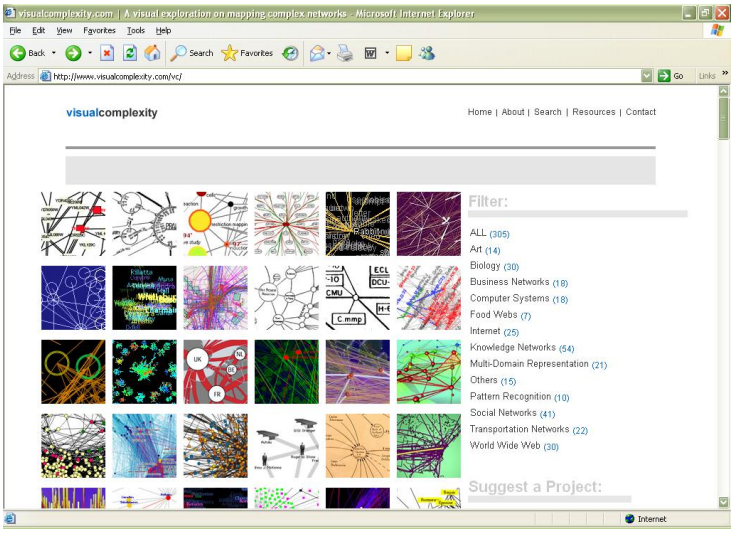
When bills came to US, they would ignore and move on

More Neat Stuff

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

http://www.visualcomplexity.com

Mucho Examples

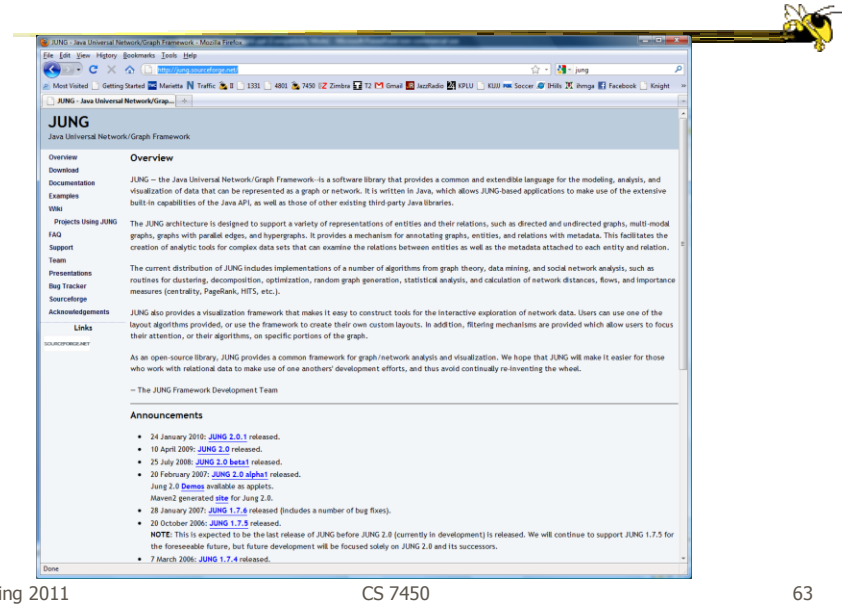


Graph Drawing Support

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

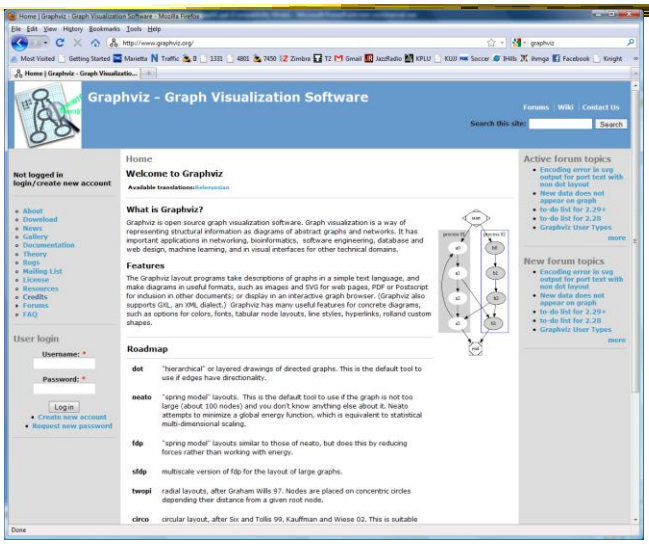
http://jung.sourceforge.net/

JUNG



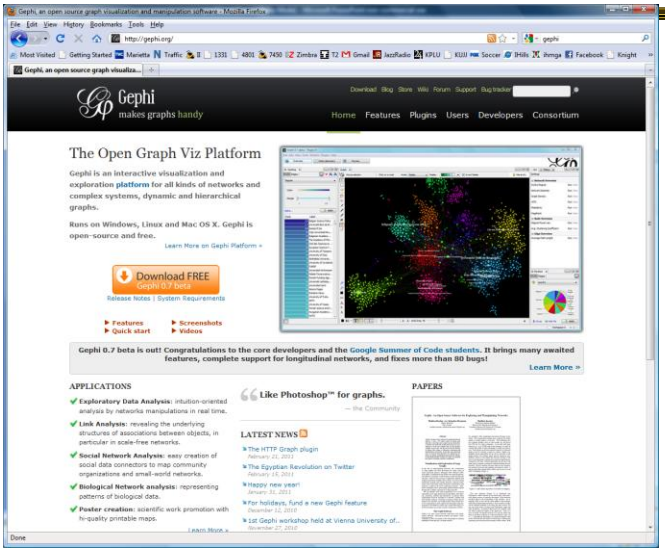
http://www.graphviz.org

Graphviz



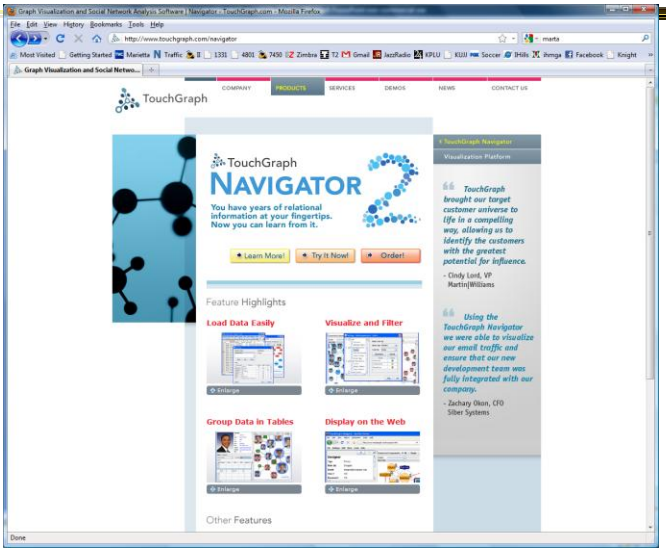
http://gephi.org

Gephi

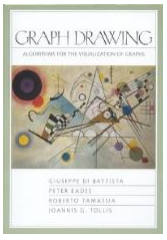


http://www.touchgraph.com/navigator

TouchGraph



Graph 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

- Graphs and Networks 2
  - Reading
    - Perer & Shneiderman
- Text and Documents 1
  - Reading
    - Ward chapter 9
    - Wattenberg & Viegas '08