CS 519: Scientific Visualization

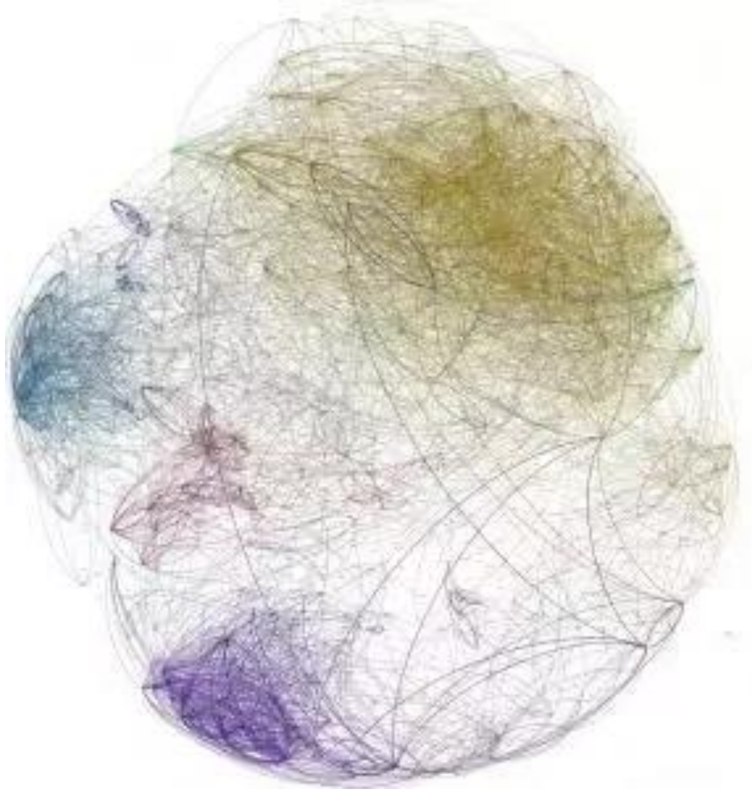
Network Visualization: Visual Analysis of Large Graphs

Eric Shaffer

Some slides adapted from:
Alexandru Telea, *Data Visualization Principles and Practice*

Large Networks Are Problematic

- 2012 webgraph:
  3.5 billion pages
  128 billion links
- Probably don't have enough pixels
- Even if we did, probably don't have enough cognitive capacity



Not the webgraph

Graph Preprocessing

- Idea: We can visualize smaller graphs

Let's make the big graph into a small graph

…try to keep the most important parts

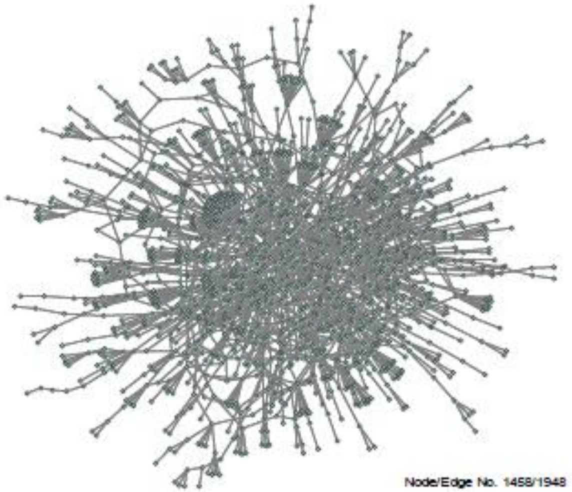
- Two approaches:
  - Graph filtering: remove unimportant parts
  - Graph aggregation: merge similar graph elements together

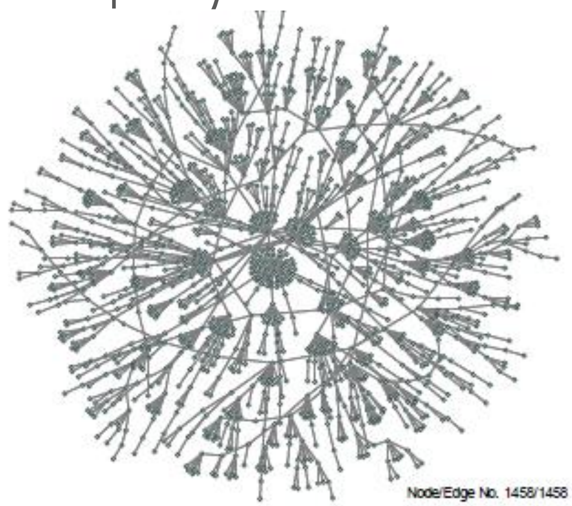
Graph Filtering

JIA Y., HOBEROCK J., GARLAND M., HART J.: **On the visualization of social and other scale-free networks.** IEEE Transactions on Visualization and Computer Graphics 14, 6 (2008)

Removes edges in order of increasing betweeness centrality

Preserves connectivity
Preserves graph features (e.g. cliques)

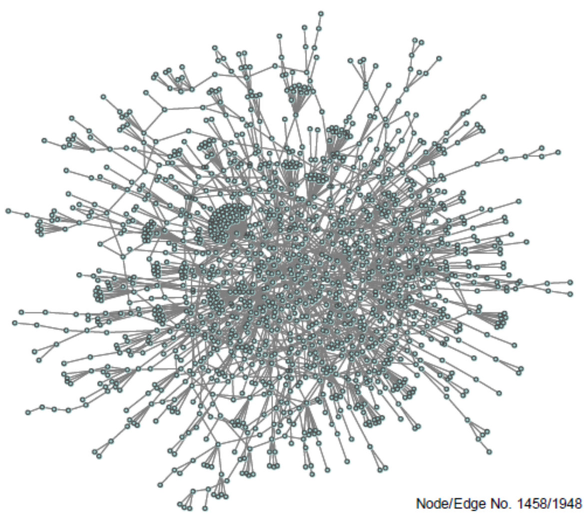


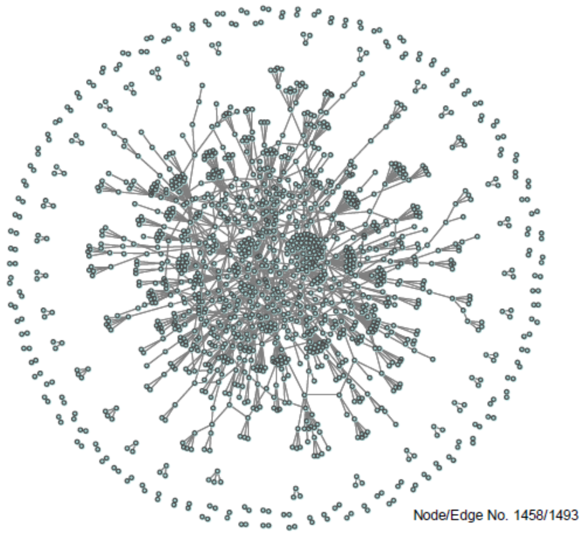


Betweeness Centrality

- Betweeness centrality (BC) ranks edges
  - How often they appear on shortest paths
  - High BC → important communication tunnels
  - Low BC → less important
- Remove low BC edges
  - Keeps "back bone" of the graph

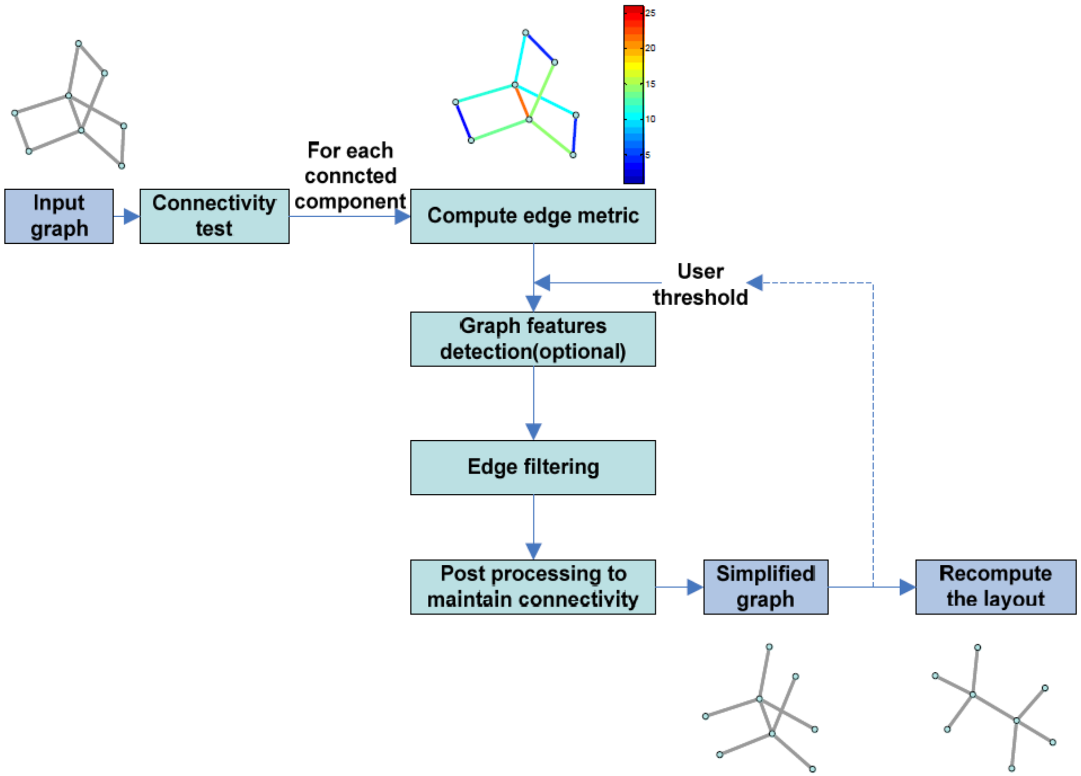
Simple Edge Filtering is Insufficient





Need to maintain connectivity…possibly other important features

Workflow



Input graph
Connectivity test
For each conncted component
Compute edge metric
User threshold
Graph features detection(optional)
Edge filtering
Post processing to maintain connectivity
Simplified graph
Recompute the layout

Betweeness Centrality is Expensive

Graph $G= (V, E)$, $|V| = n$, $|E| = m$

- Betweenness centrality [Freeman 1977]
- Relies on computing All-Pairs Shortest Paths
- Complexity $O(m*n)$ for unweighted graph [Brandes 01]

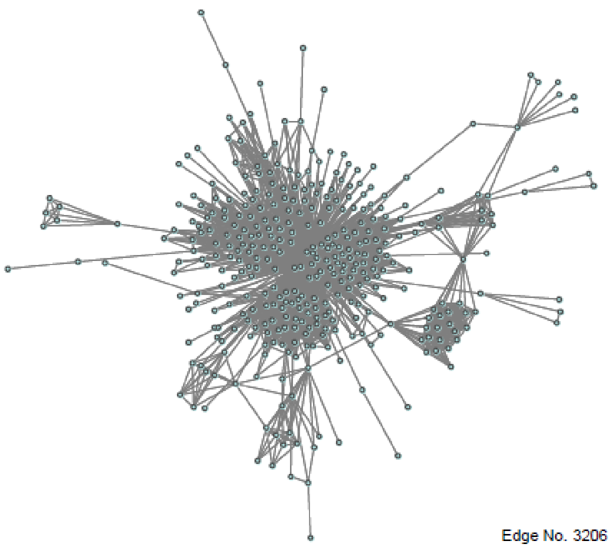
For huge graphs

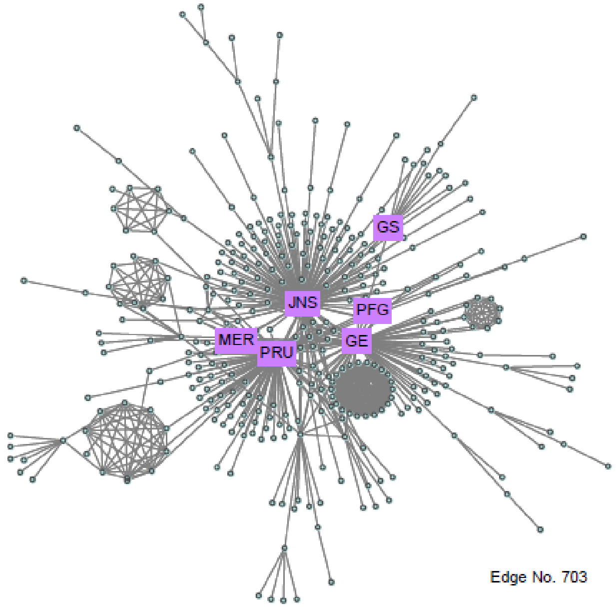
- Approximated with random sampling [Jacob et al. 05]
  - $O((m+n)*log(n))$ with $C*log(n)$ samples where C is a constant

- For our edge filtering purpose
  - Only relative orders of BC are needed
  - Select $C*log(n)$ highest degree hub nodes

Graph Feature Detection

- Graph features
  - Cliques
    - NP-Complete problem
    - Fast approximation *O(m*n)* [Chiricota et al. 03]
- User defined features





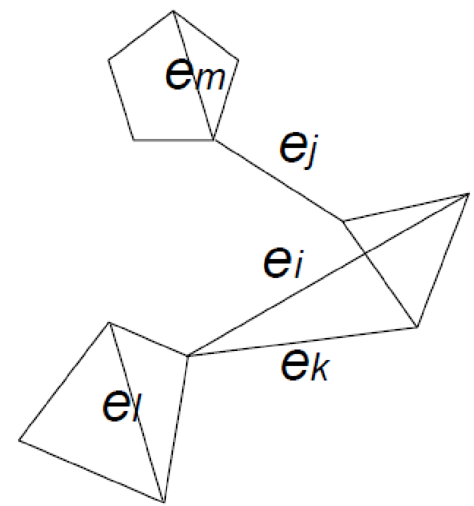
Edge Filtering

Edges	BC Metric
…	…
e_h	*1.3*
e_i	*1.1*
e_j	*1.2*
e_k	*1.15*
e_l	*1.21*
e_m	*1.09*
…	…

Sort →

Edges	BC Metric
e_m	*1.09*
e_i	*1.1*
e_k	*1.15*
e_j	*1.2*
e_l	*1.21*
e_h	*1.3*
…	…

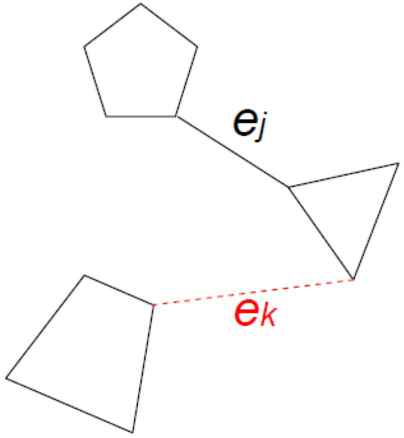
Threshold $t = 1.25$



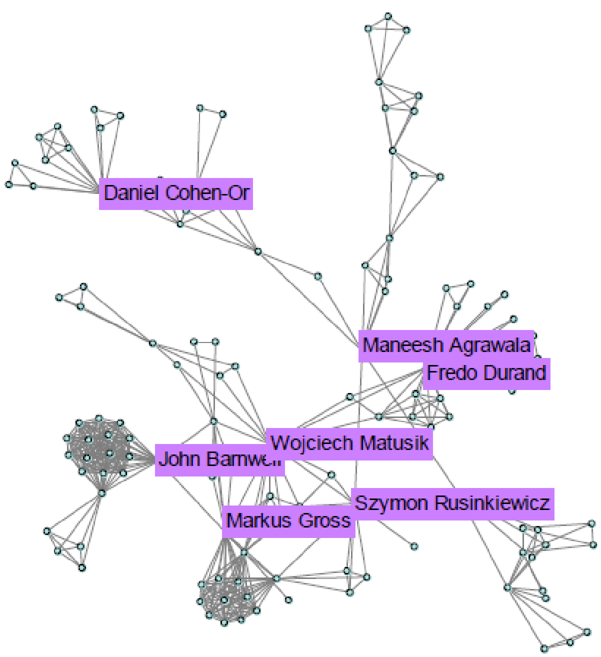
Recover Connectivity

e_l	1.21
e_j	1.2
e_k	1.15
e_i	1.1
e_m	1.09
Removed Edges	BC Metric

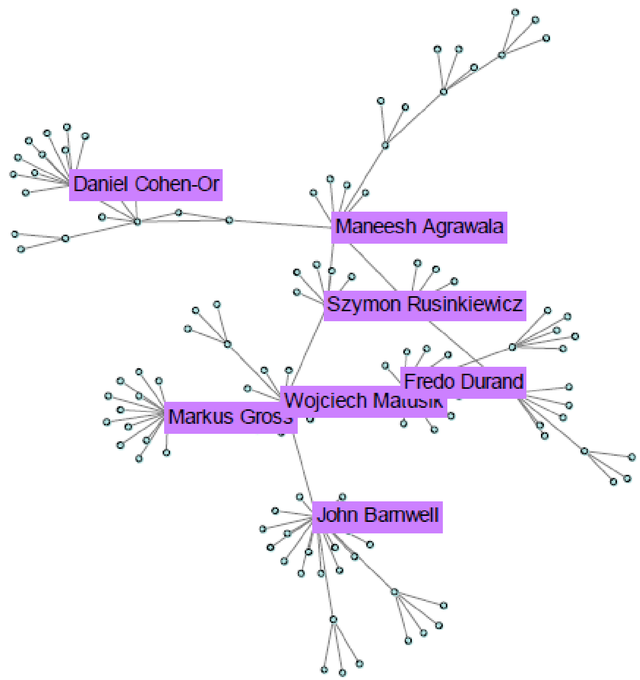
stack



Recompute the Layout

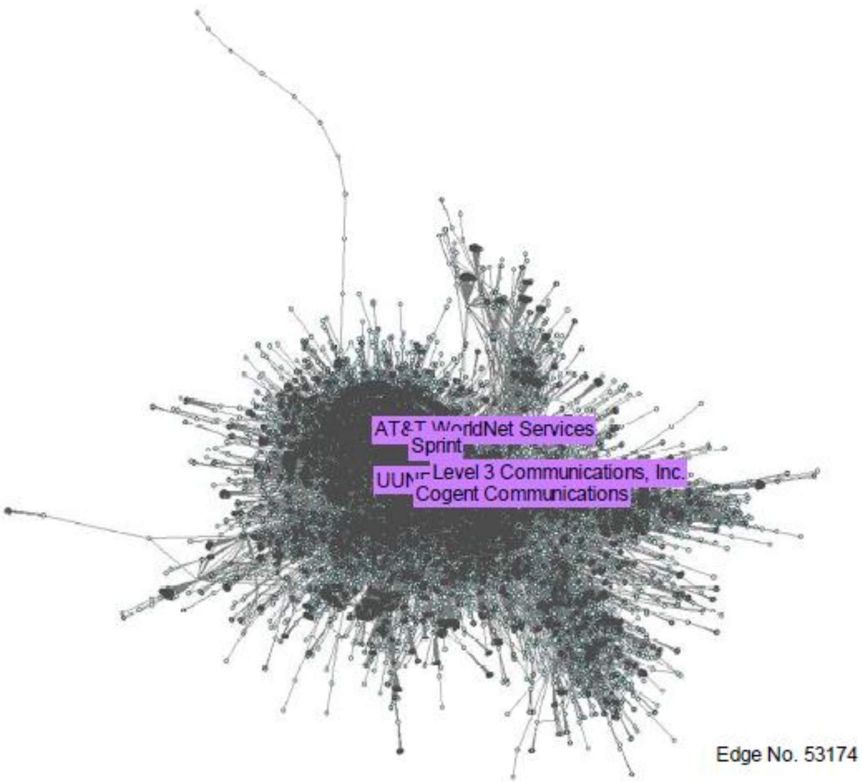


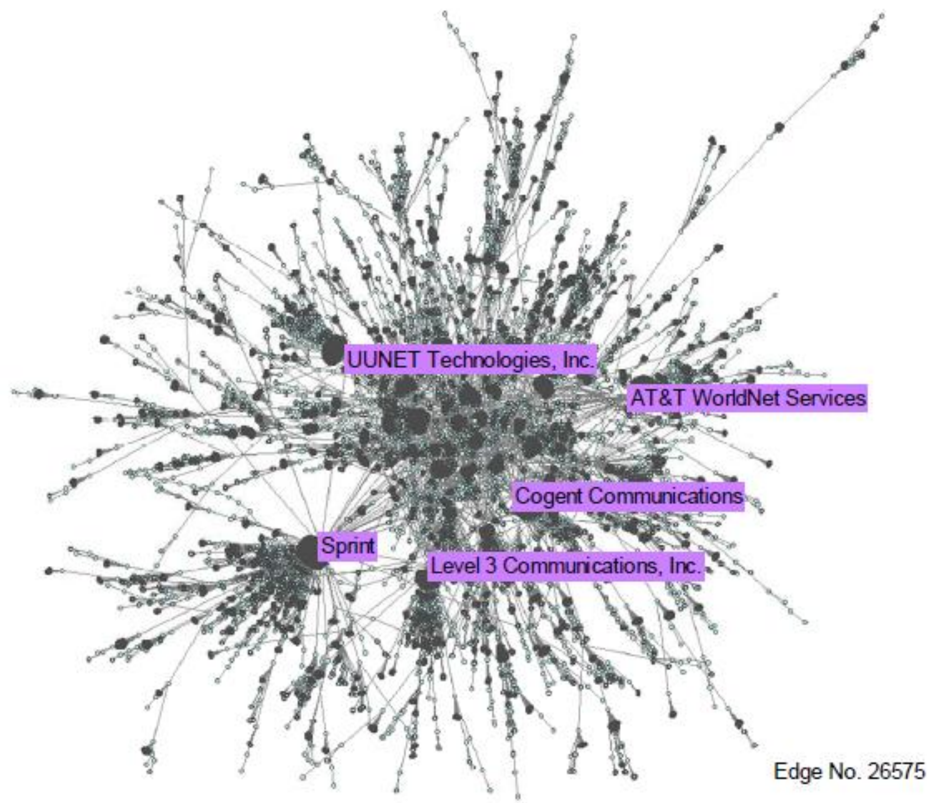
Edge No. 441



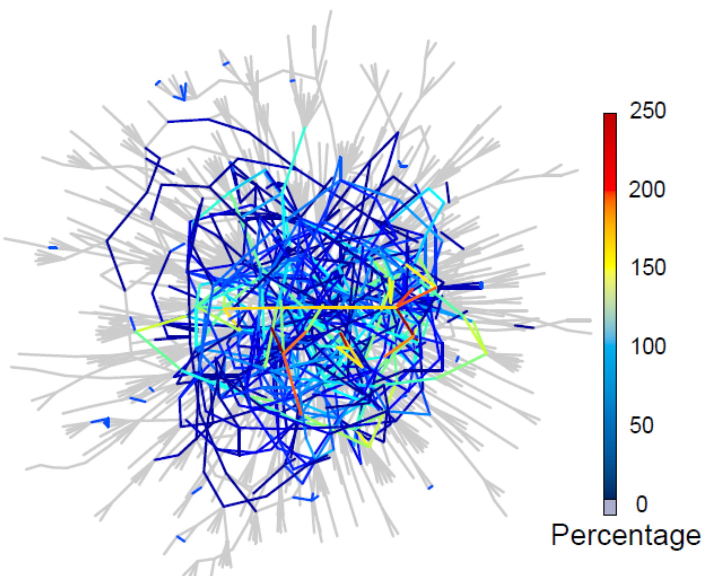
Edge No. 129

Fixing the Hairball….

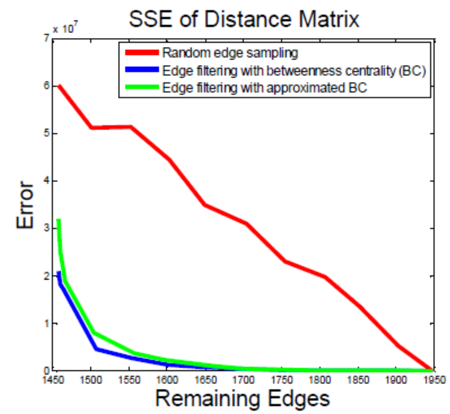


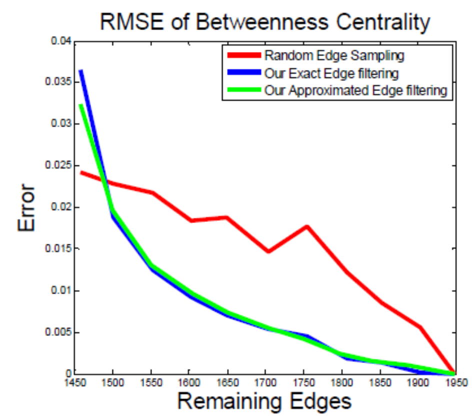


Empirical Data on Error



Relative error of BC approximation

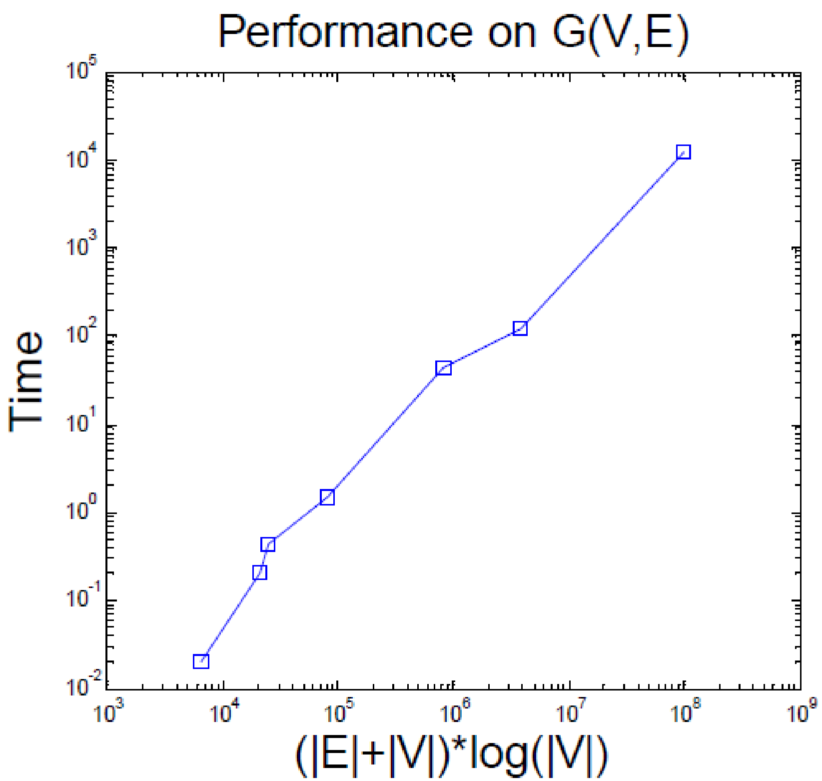




- Error measured as

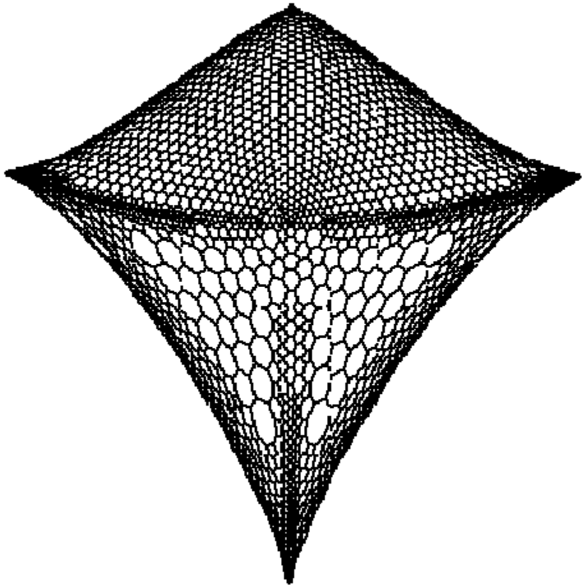
Performance

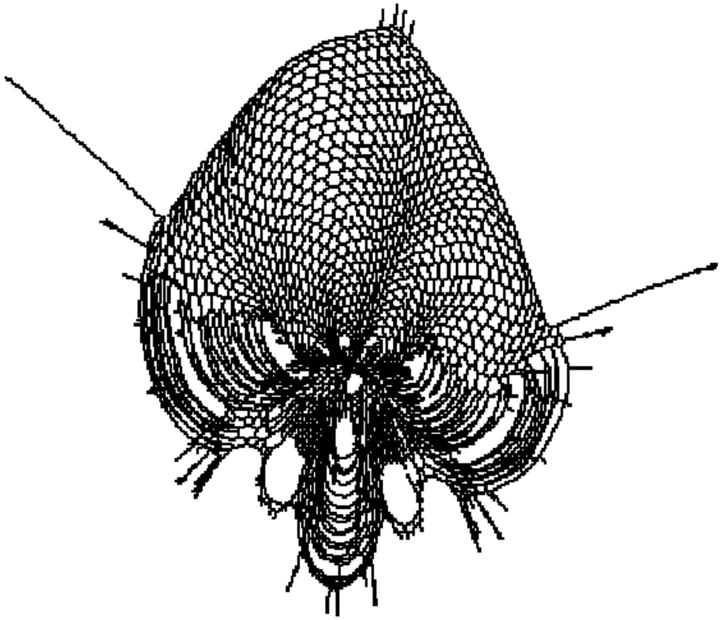
Graph	Nodes	Edges	Timing
siggraph07	328	773	0.02s
sp500-038	365	3206	0.20s
bo	1458	1948	0.44s
cg_web	2269	8131	1.50s
as-rel.071008	26242	53174	43.66s
hep-th	27400	352021	120.72s
flickr	820878	6625280	12442.70s



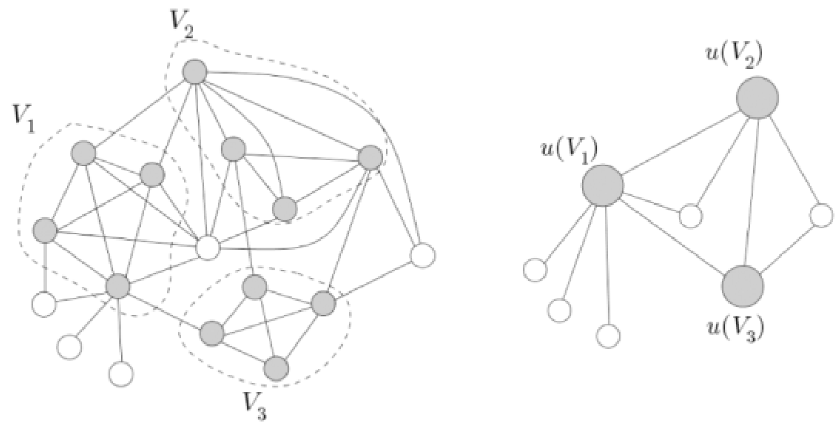
Limitations

- Doesn't work well for non-power law graphs
  - Including planar graphs
- Obviously doesn't show entire data set

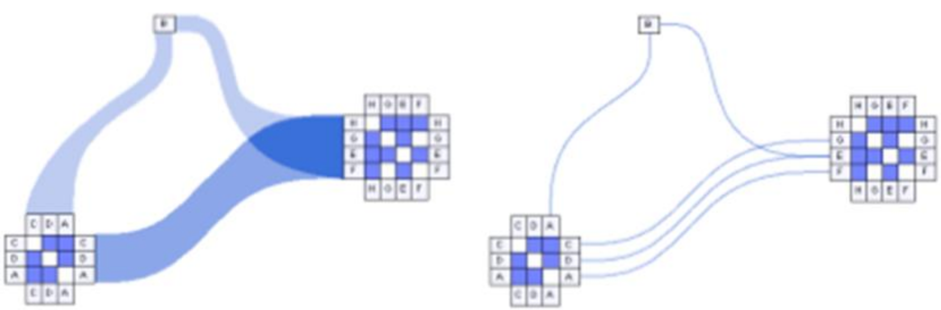




Graph Aggregation



(a) (b)



aggregated edges original edges

- Graph aggregation
  - Produces a simpler/smaller 'cluster graph' from a large one
  - Vertices: partitioned between disjoint clusters
  - Edges:
    - intra-cluster: relate nodes in same cluster
    - inter-cluster: relate clusters
  - many clustering methods (strongly-connected components, data based, …)
- Visualize
  - The cluster graph
  - Cluster internals using a cluster icon (e.g. a matrix plot)

AggregationTechniques for Large Graphs

- We have seen some techniques already
  - Multi-level force-directed placement
  - Matrix representations and re-orderings
- Some other methods
  - Edge bundling
    - Visually group similar edges to alleviate occlusion
  - Graph splatting
    - Force directed layout meets volume rendering

Matrix Aggregation

N. Elmqvist, T.-N. Do, H. Goodell, N. Henry, J.-D. Fekete. **ZAME: Interactive Large-Scale Graph Visualization.** In *Proceedings of the IEEE*

Pacific Visualization Symposium, pp. 215-222, 2008.

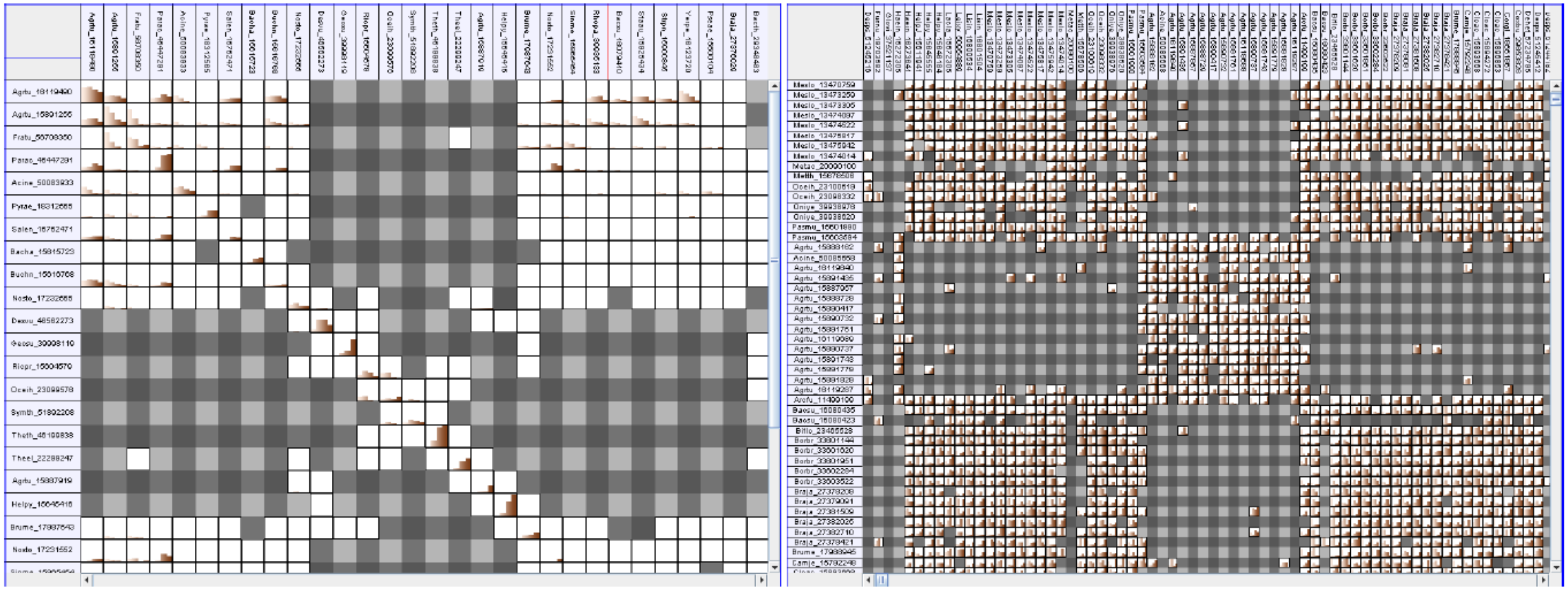
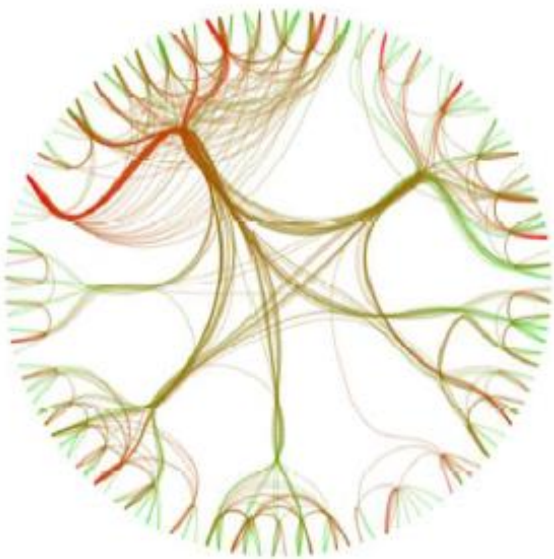


Figure 1: A protein-protein interaction dataset (100,000 nodes and 1,000,000 edges) visualized using ZAME at two different levels of zoom.

Graph Aggregation: Edge Bundling

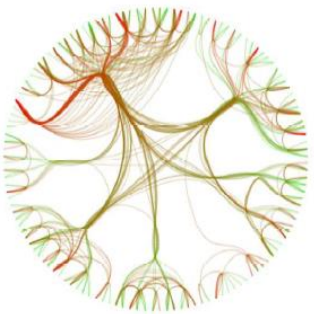
- Edge bundles are clusters of similar edges
- Many approaches…usually cluster vertices
  - Edges between clusters follow similar paths
- Some metrics
  - Shortest path distance to a "hub-node"
  - Remove high-BC edges to discover clusters
  - Lots of others



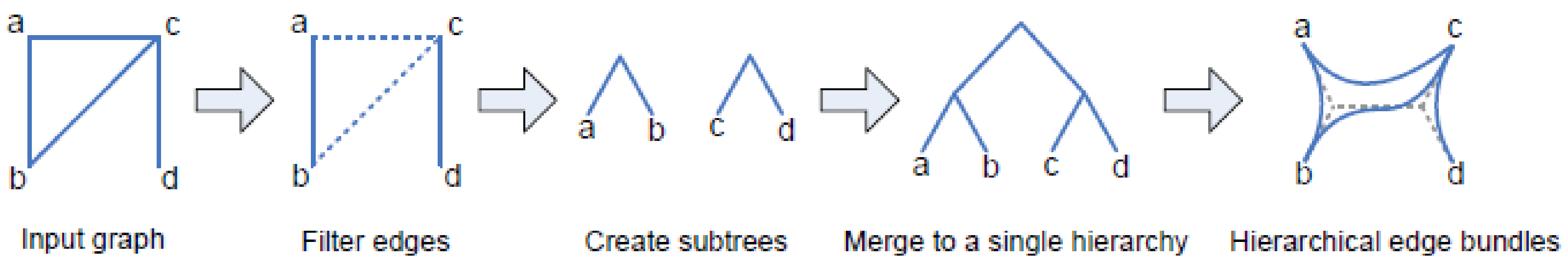
Hierarchical Edge Bundling: Example

Yuntao Jia, Michael Garland, John C. Hart:
Social Network Clustering and Visualization using Hierarchical Edge Bundles. Comput. Graph. Forum 30(8): 2314-2327 (2011)

1. Generate a hierarchical structure of vertex clusters
2. Vertices are placed radially around circle
   1. Positions from in-order traversal of hierarchy
   2. Root nodes of clusters in in interior, leaves on the perimeter
3. Edges are B-Spline curves
   1. Control points are hierarchy node layout positions along shortest tree path between the two nodes



Balanced Hierarchy Construction



- Filter edges by removing highest-bc first
- Construct communities by merging in increasing BC order of removed edges

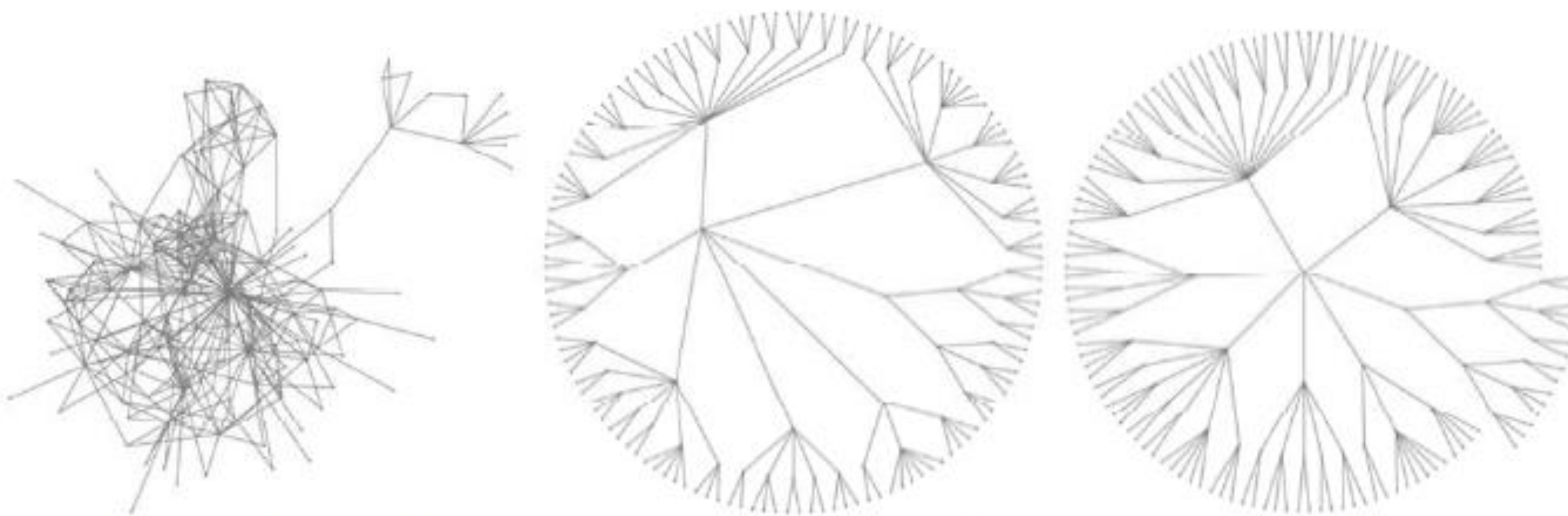
BC Calculation

$$BC(v) = \sum_{u \neq v \neq w \in V} \sigma_{u,w}(v)/\sigma_{u,w}$$

where $\sigma_{u,w}$ counts the number of shortest paths between u and w, and $\sigma_{u,w}(v)$ counts only the ones containing v.

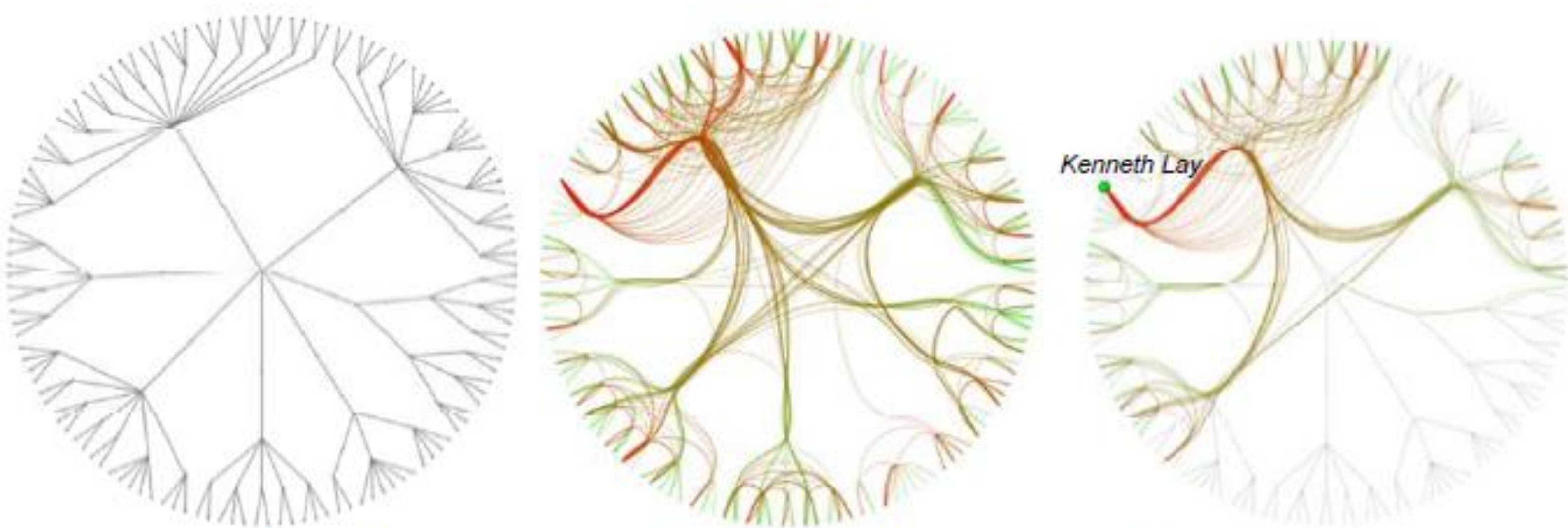
- An edge a,b is removed only if min(deg(a),deg(b)) > 1
- And BC(a,b) > 1

Examples



Original Graph Extracted Hierarchy Fixed Root

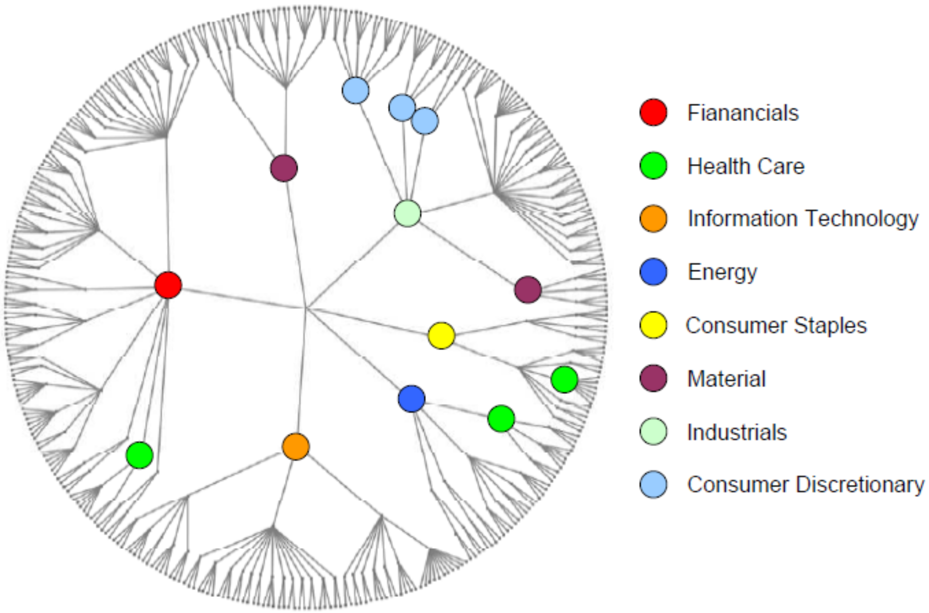
Enron E-Mail Graph



Fixed Depth　Edge Bundles　Selected visualization

- Enron scandal 2001
- 389 e-mails, 132 employees
- Red = sender, Green = recipient
- Can select node to see which communities that person contacted

Community Discovery



Generated Hierarchy (with user supplied labels)

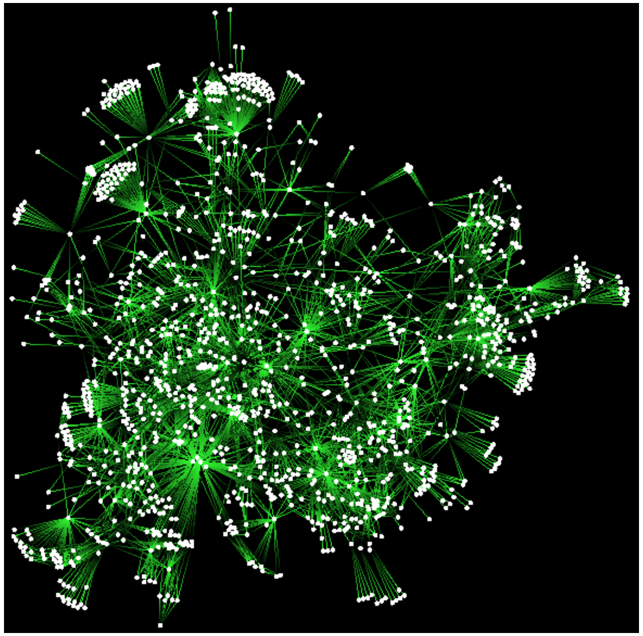
Figure 8 visualizes the undirected graph "sp500-38," which represents 3,206 cross correlations of price fluctuation of 365 stocks from the S&P 500. Our method is able to recognize different stock sectors and put them near to each other in the hierarchy. The hierarchical edge bundles visualization reveals that financial stocks affect all other kind of stocks except energy, consumer staples and health stocks, which are relatively independent.

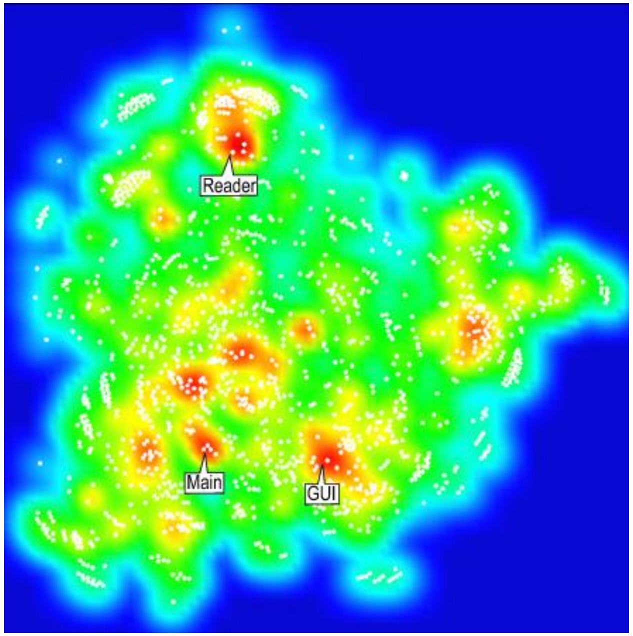
Graph Splatting

- node-link layouts get easily cluttered; remove problem by not drawing edges
- key idea: transform discrete dataset into a continuous one!
- do a force-directed layout
- convolve nodes (and optionally edges) with a Gaussian filter (radius r = simplification level)

$$f(x,y) = \Sigma_{i=1}^{N} k_i e^{-\left(\frac{|x-x_i|}{r}\right)^2}$$

- render the resulting density signal using SciVis methods

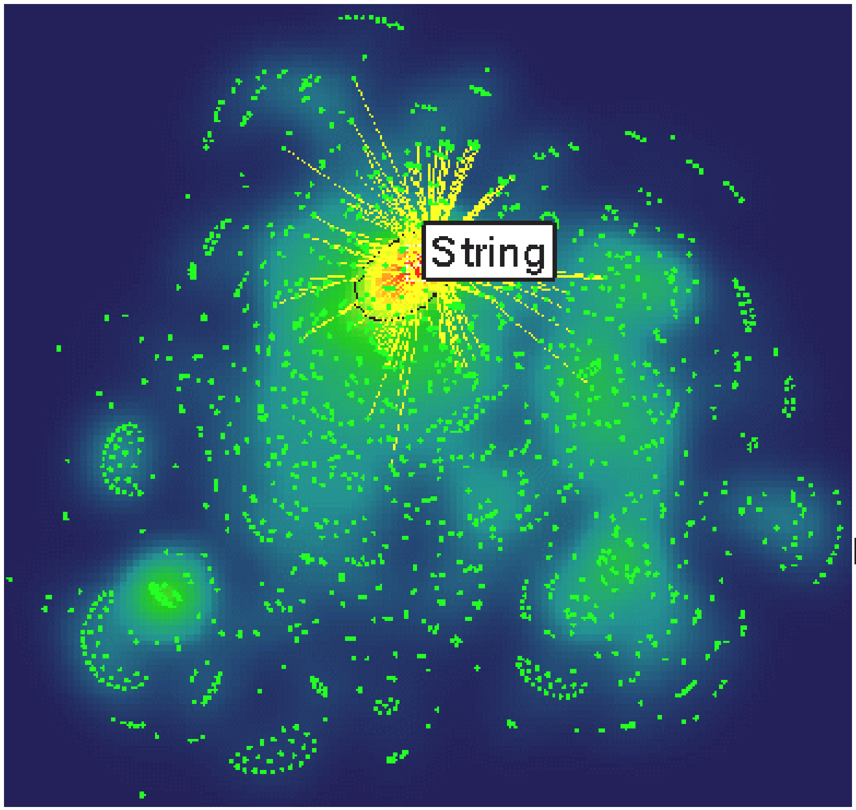




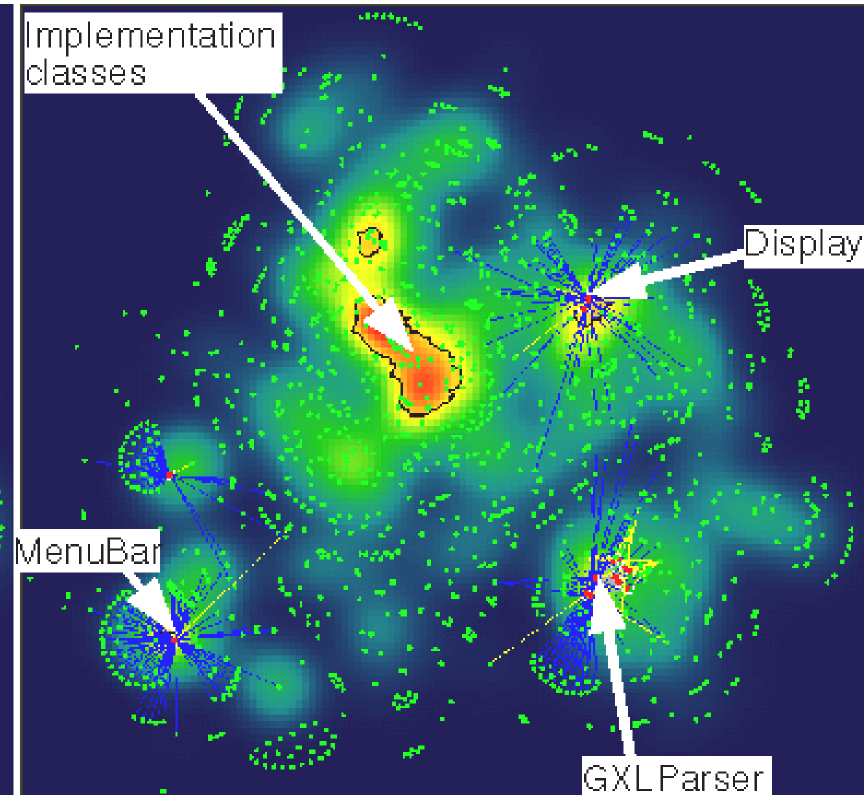
Java call graph (5000 nodes, 2000 relations), courtesy Nokia Research

Graph Splatting

- use factor k_i to specify the importance of i^{th} node
- example: k_i = number of requirements / provisions of a component (out/ingoing edges)
- high densities denote strongly connected clusters / important nodes



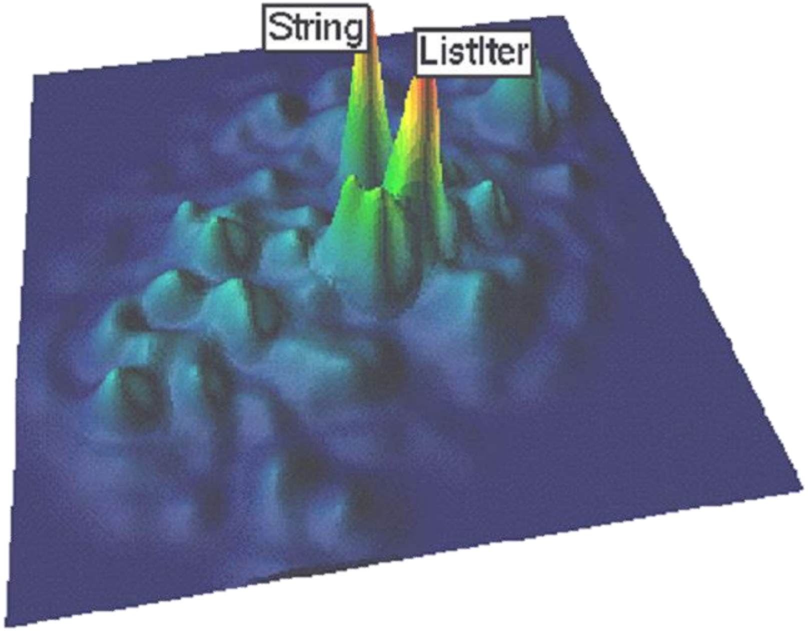
components using String class



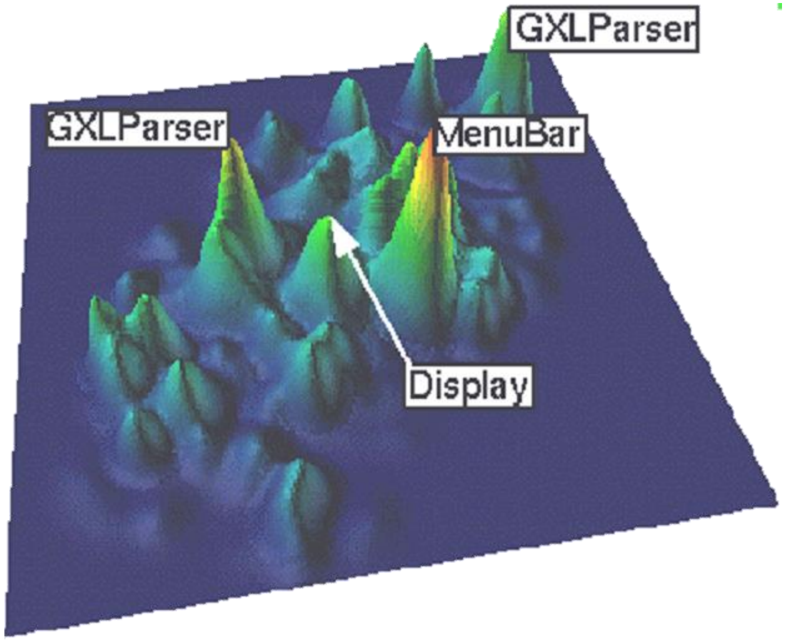
package requirements

Graph Splatting

- the splat field can be visualized also as a height plot
- other SciVis techniques possible too (isolines, clipping…)
- however: the result quality strongly depends on the layout quality



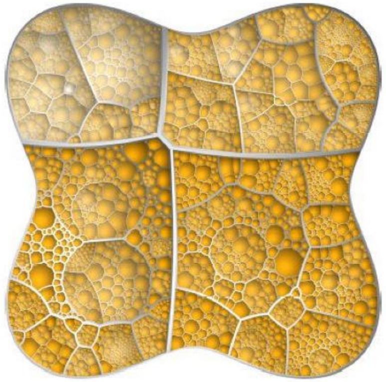
components using String class



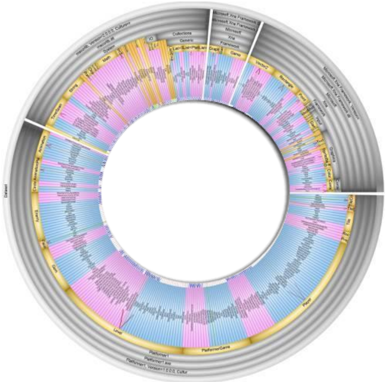
package requirements

Other Techniques

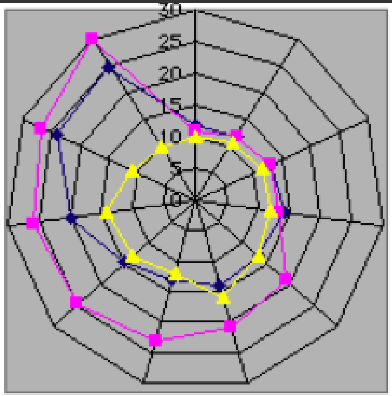
directionally constrained layouts



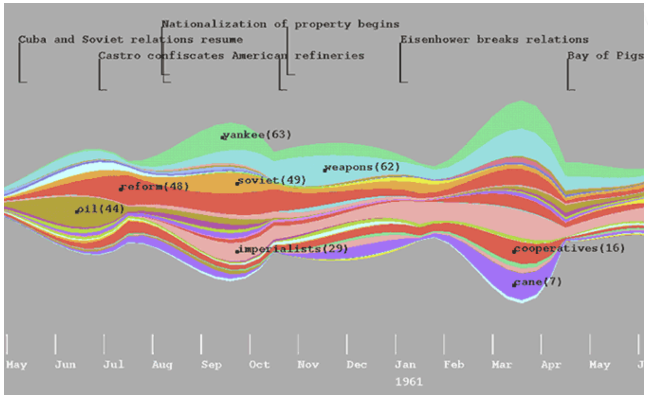
Voronoi treemaps



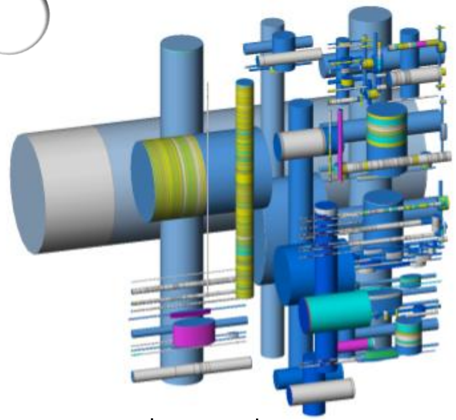
radial icicle plots



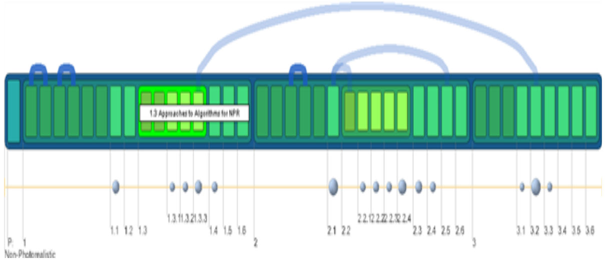
star plots



theme rivers



beam trees



arc trees