Depicting Networks and Trees

The yEd Graph Editor www.yworks.com

Archambault, D. "Visual Analytics: an Introduction", Swansea University, 2020

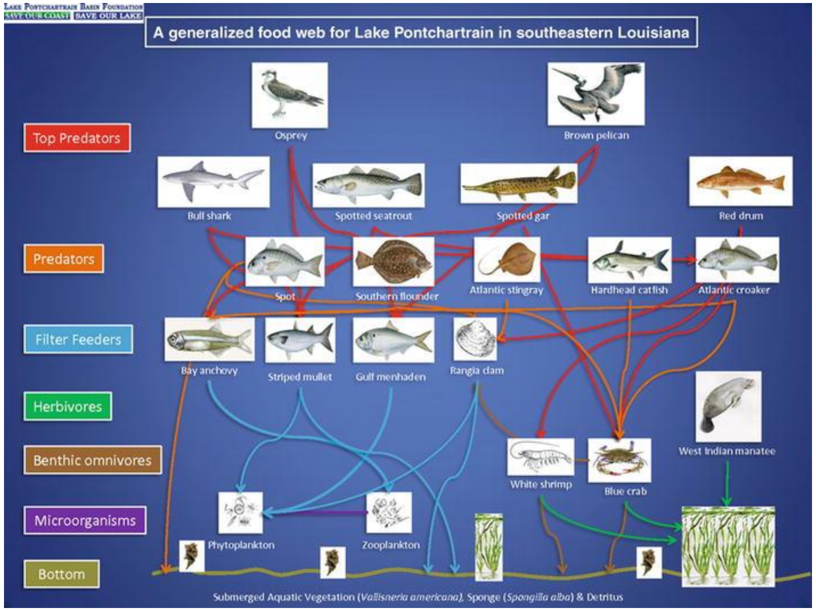
Kerren, A. "Information Visualisation: Perception", Linnaeus University, 2020

Networks/Graphs

Network: relationships between objects

Graph: edges between nodes

Examples:
- train routes between cities
- friendships between people
- management of employees
- links between webpages
- predator-prey relationships



Weis J.S. (2016) Predator–Prey Relationships. Encyclopaedia of Estuaries.

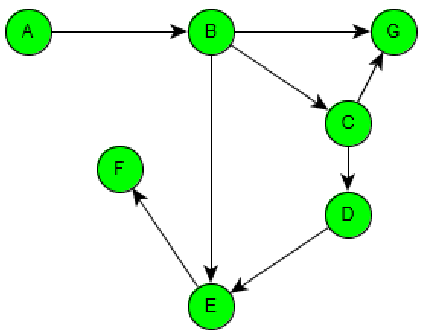
Graph Data: directed

	A	B	C	D	E	F	G
A	0	1	0	0	0	0	0
B	0	0	1	0	1	0	1
C	0	0	0	1	0	0	1
D	0	0	0	0	1	0	0
E	0	0	0	0	0	1	0
F	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0

graph adjacency matrix

A	B		
B	C	E	G
C	D	G	
D	E		
E	F		

graph adjacency list



graph drawing

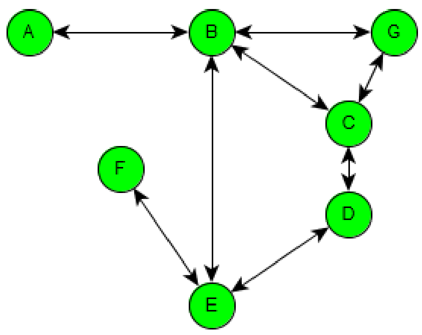
Graph Data: bi-directed

	A	B	C	D	E	F	G
A	0	1	0	0	0	0	0
B	1	0	1	0	1	0	1
C	0	1	0	1	0	0	1
D	0	0	1	0	1	0	0
E	0	1	0	1	0	1	0
F	0	0	0	0	1	0	0
G	0	1	1	0	0	0	0

graph adjacency matrix

A	B			
B	A	C	E	G
C	A	D	G	
D	C	E		
E	B	D	F	
F	E			
G	B	C		

graph adjacency list



graph drawing

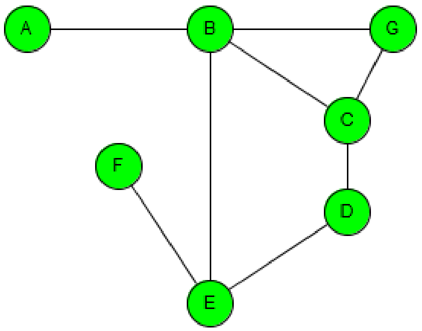
Graph Data: undirected

	A	B	C	D	E	F	G
A	0	1	0	0	0	0	0
B	1	0	1	0	1	0	1
C	0	1	0	1	0	0	1
D	0	0	1	0	1	0	0
E	0	1	0	1	0	1	0
F	0	0	0	0	1	0	0
G	0	1	1	0	0	0	0

graph adjacency matrix

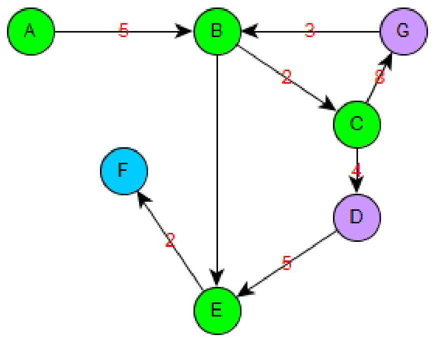
A	B		
B	C	E	G
C	D	G	
D	E		
E	F		

graph adjacency list



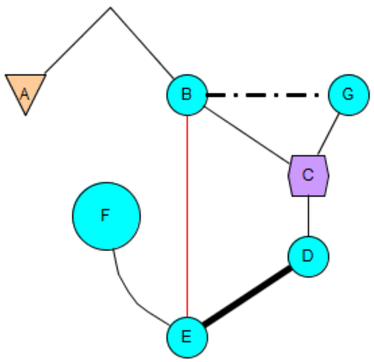
graph drawing

- Graph have nodes (vertices) and edges
- Edges can be directed or undirected
- The degree of a node: number of connecting edges
- For directed graphs (or ‘digraphs’)
  - in-degree: number of incoming edges
  - out-degree: number of outgoing edges
- Directed graphs can have cycles
- Edges can have attributes (particularly numerical weights)
- Nodes can have attributes
- Trees are graphs with a hierarchical structure



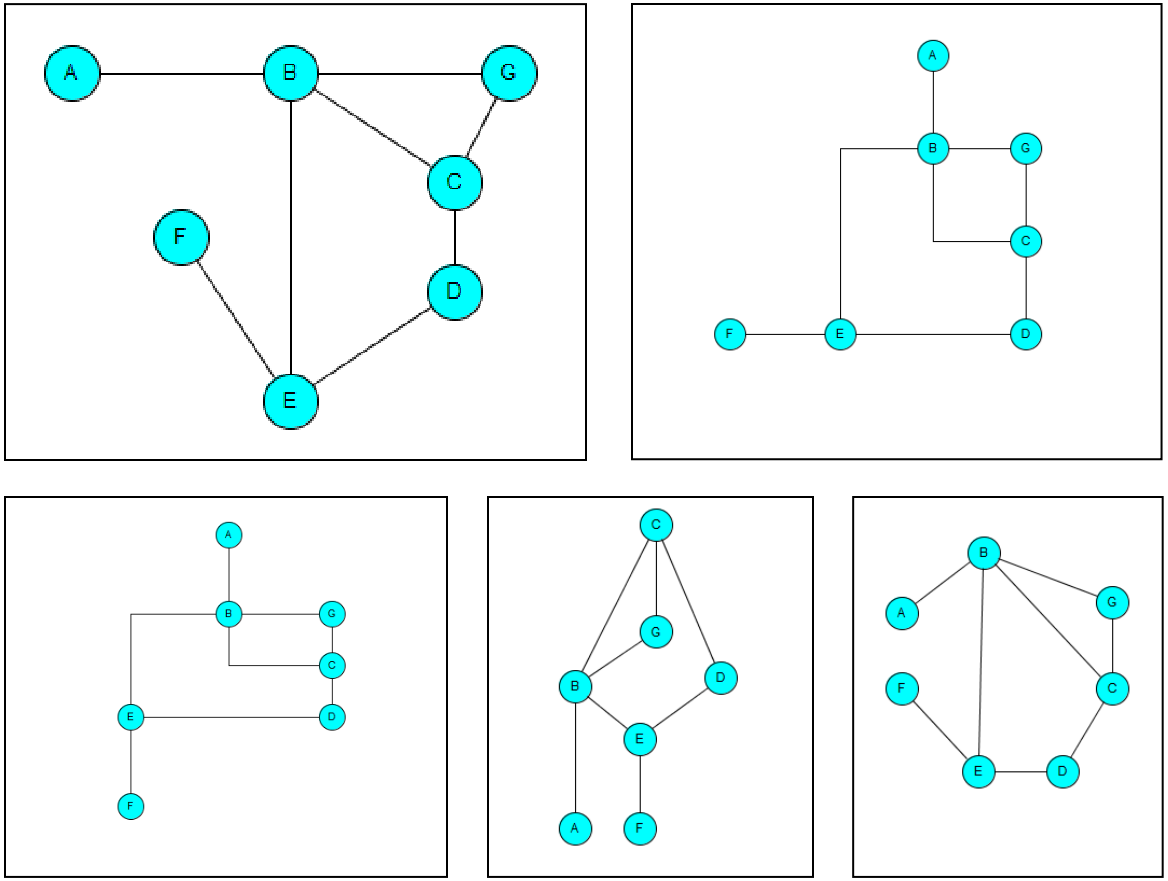
- Graph drawing: placing the nodes on the plane
- Graph visualisation: providing interactive features for exploring the graph

- Graph visual encoding:
  - nodes: shape, colour, size
  - edges: straight/curved/polyline, colour, thickness, texture



Approaches to Graph Drawing

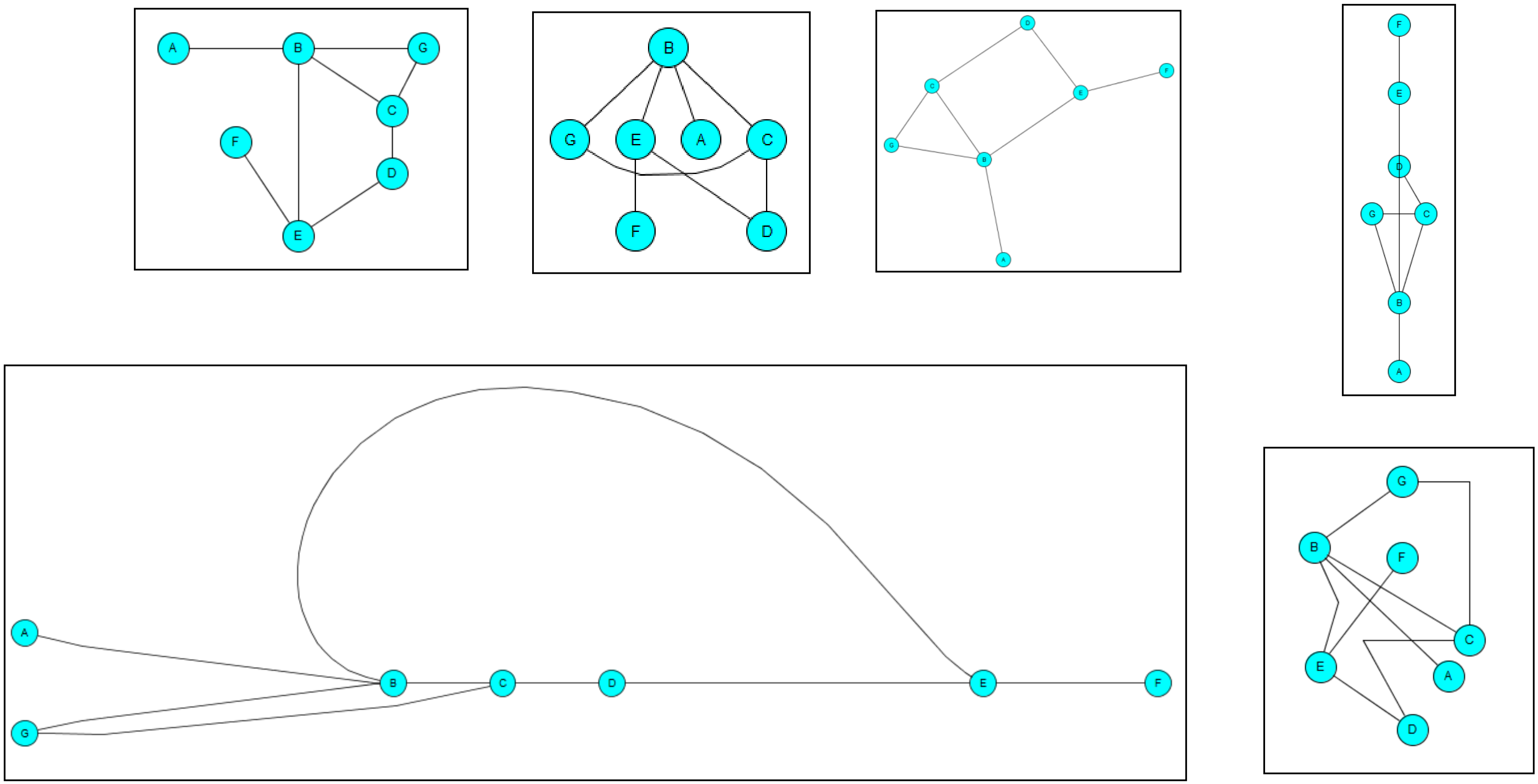
- Straight line
- Polyline
- Orthogonal
- Grid
- Upward
- Circular



Several graph layout methods, embodied in graph layout algorithms

See: Handbook of Graph Drawing and Visualisation,
https://cs.brown.edu/people/rtamassi/gdhandbook/

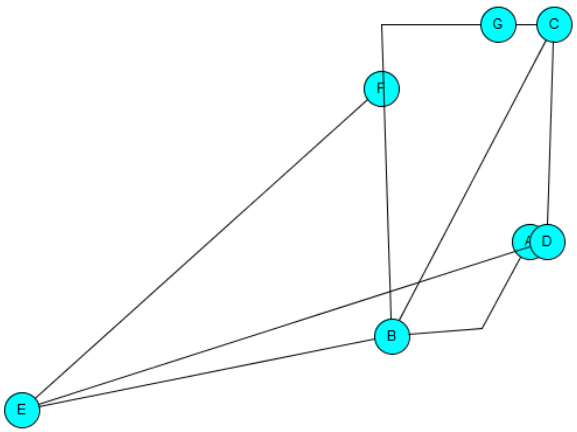
Choose algorithm to best support understanding

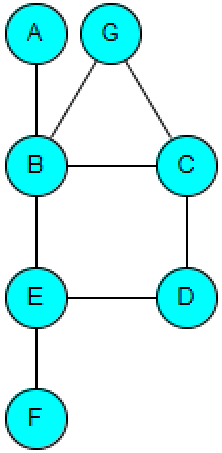


https://www.yworks.com/products/yed

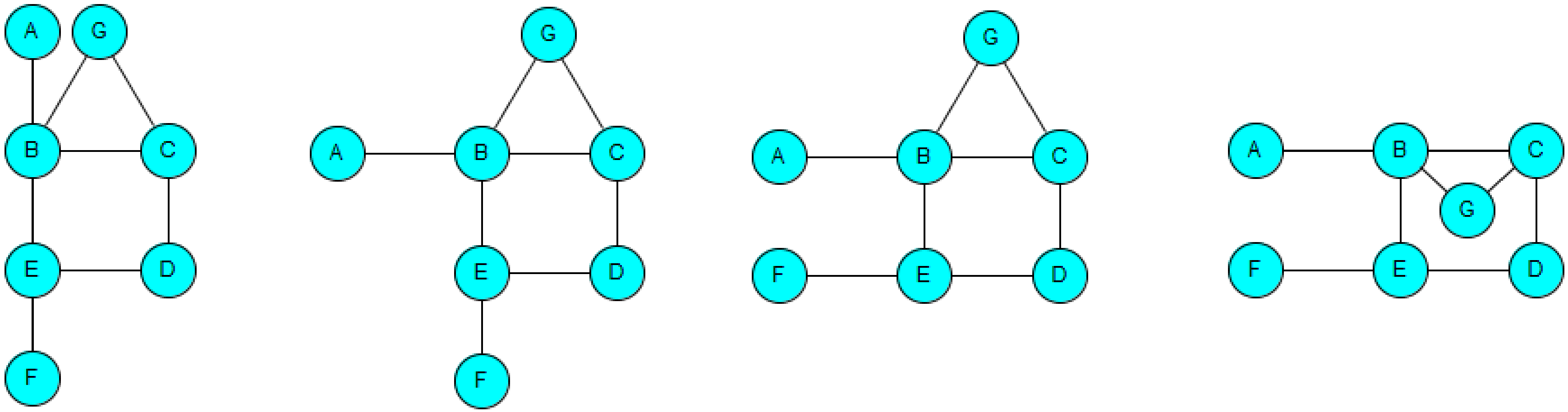
Graph drawing aesthetics/ principles

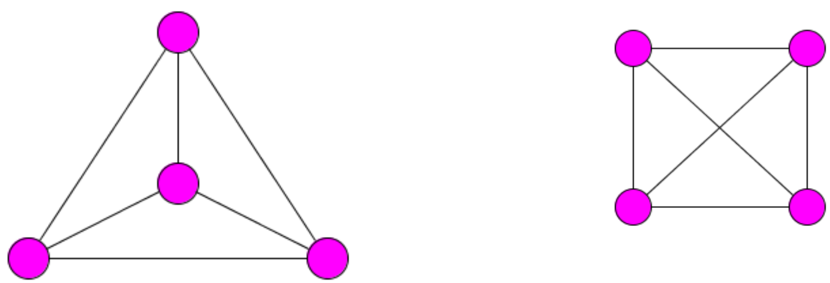
- Reduce
  - node overlap
  - node/edge overlap
  - edge crossings
  - total area
  - edge lengths: maximum, variance, total
  - edge bends: number
- Increase
  - depiction of symmetry
  - minimum angle at nodes





Aesthetic conflicts





Conforming to aesthetic criteria can also be computationally expensive

Thus: aesthetics can only be *heuristic guidelines*, not *mandatory requirements*

Types of graph

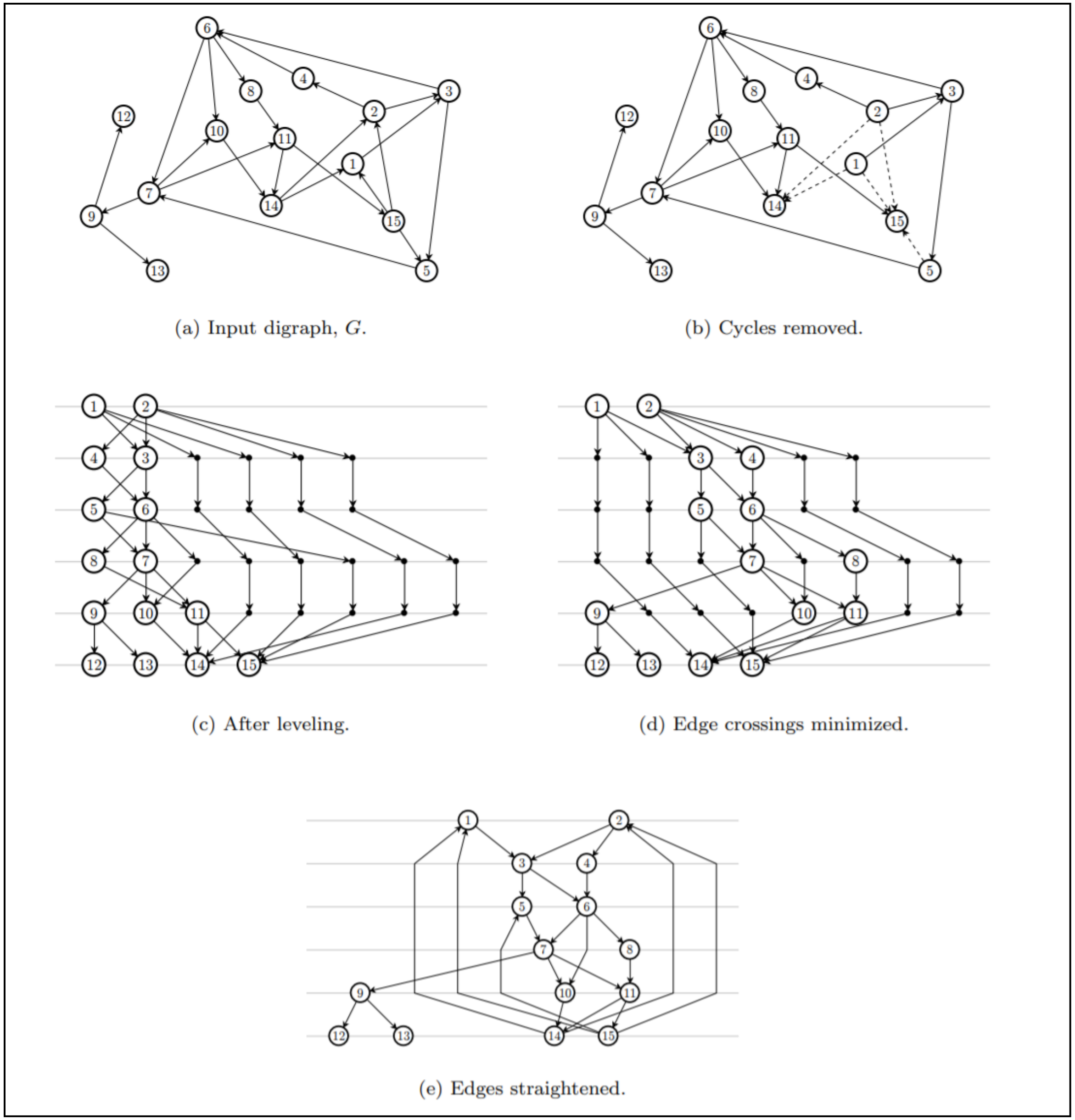
- **Directed Acyclic:** directed edges, no cycles
- **Planar:** can be drawn with no edge crossings
- **General graphs:** no assumptions
- **Trees:** strict hierarchy

Directed Acyclic graphs

K. Sugiyama, S. Tagawa and M. Toda, Methods for Visual Understanding of Hierarchical System Structures, in IEEE Transactions on Systems, Man, and Cybernetics, 11(2):109-125, Feb. 1981.

1. If the graph has cycles, reverse the direction so that the cycles are removed
   - (remember which ones; you will need to re-reverse the direction at the end)
2. Assign nodes to vertical layers
   - create dummy nodes so that each edge only traverses one layer
   - directed edges go from one layer to the next
3. Order the nodes horizontally within each layer to minimise crossings
4. Move the nodes horizontally within each layer to straighten edges
5. Re-reverse the direction of the edges changed in step 1

Healy & Nikolov, Ch 13, Handbook of Graph Drawing and Visualization, 2013,



(a) Input digraph, G.

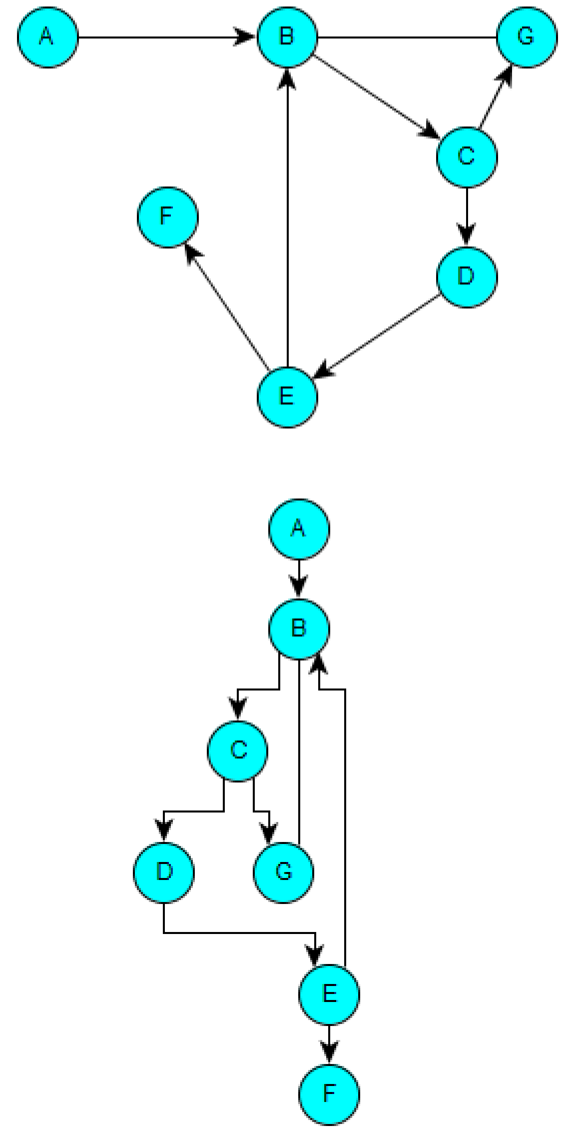
(b) Cycles removed.

(c) After leveling.

(d) Edge crossings minimized.

(e) Edges straightened.

Healy & Nikolov, Ch 13, Handbook of Graph Drawing and Visualization, 2013



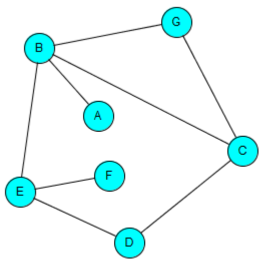
yEd graph editor (hierarchical layout)

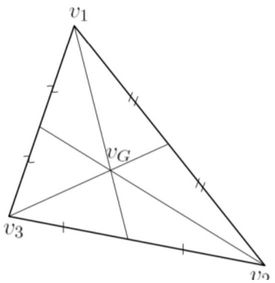
Planar graphs

Tutte, W. T. (1963), How to Draw a Graph. Proceedings of the London Mathematical Society, s3-13: 743–767

Drawing a “3-connected” planar graph with no edge crossings

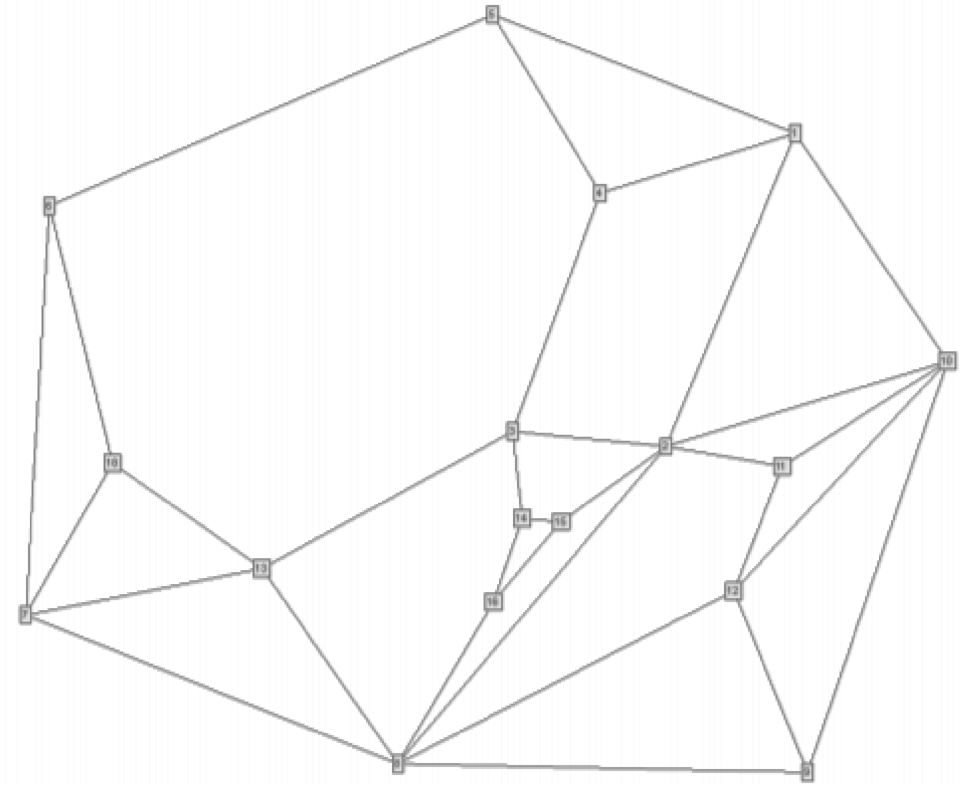
1. Nodes on the outer face placed at vertices of a convex polygon
2. Each internal node is placed at the barycentre average of its neighbours, solving a set of linear equations



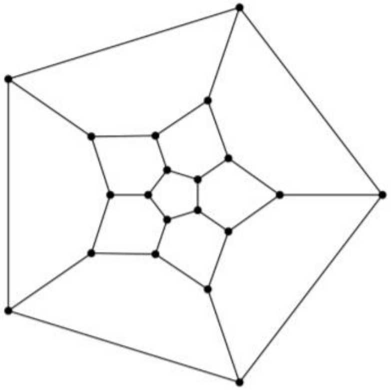


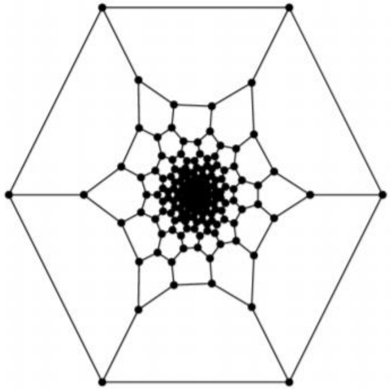
Face: set of nodes connected in a loop
Outer face: connected nodes that are unbounded
Barycentre: point where three medians of a triangle meet
3-connected: you need to delete at least 3 nodes to disconnect the graph

K. Pavlou, https://www2.cs.arizona.edu/~kpavlou/Tutte_Embedding.pdf (accessed 27/05/21)



Vismara, Ch 6, Handbook of Graph Drawing and Visualization, 2013
http://cs.brown.edu/people/rtamassi/gdhandbook/

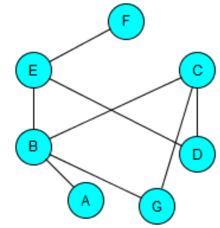


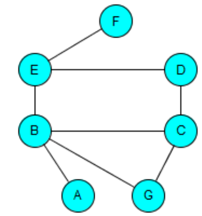


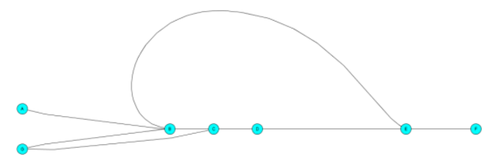
K. Pavlou, https://www2.cs.arizona.edu/~kpavlou/Tutte_Embedding.pdf (accessed 27/05/21)

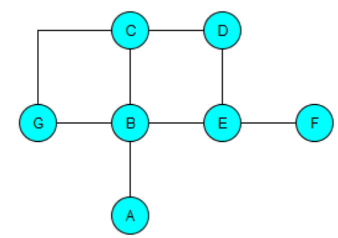
General graphs

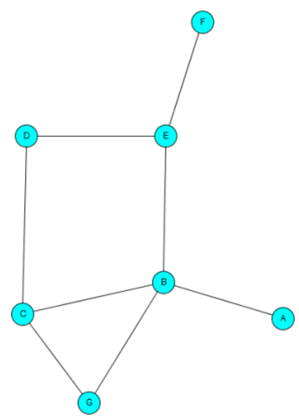
- No assumptions made
- Typically stochastic approach
  - randomly place nodes
  - improvement by iteration
  - may be non-deterministic
- Different principles
  - Circular
  - Radial
  - Orthogonal
  - Force-directed











yEd graph editor (radial, orthogonal, organic layout)

General Graphs: force-directed layout

Peter Eades. A heuristic for graph drawing. Congressus Numerantium, 42:149–160, 1984

Fruchterman, T. M. J.; Reingold, E. M.. Graph Drawing by Force-Directed Placement, Software – Practice & Experience, 21(11): 1129–1164, 1991.

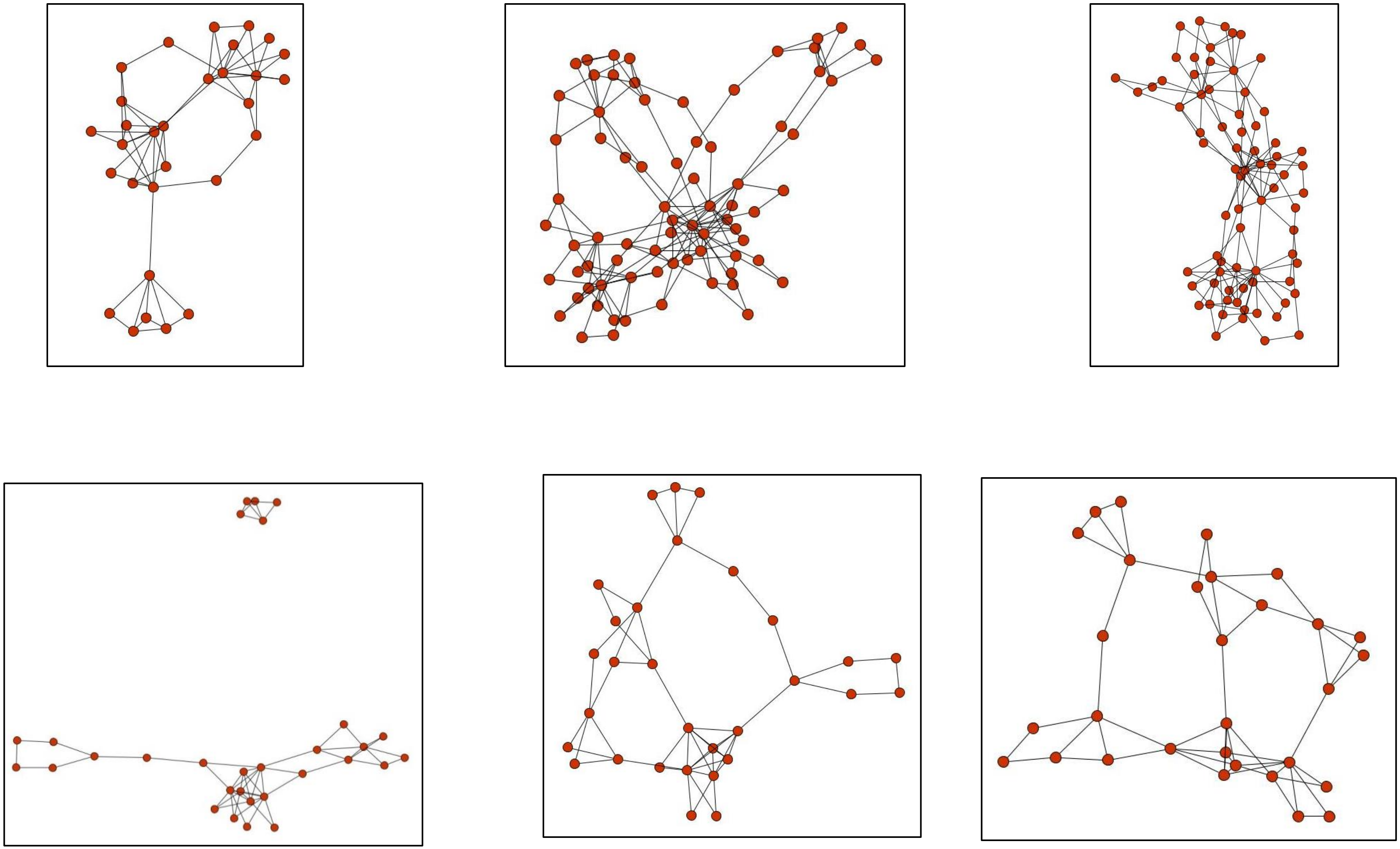
Nodes: steel rings

Edges: springs

Connected nodes: attractive force

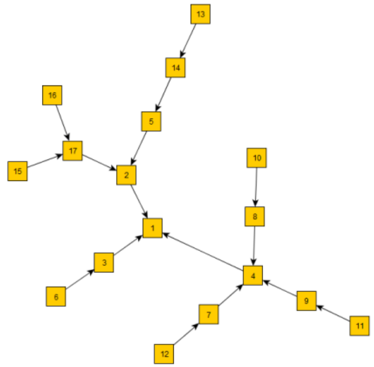
Unconnected nodes: repulsive forces

1. Start with random placement of nodes
2. Calculate the energy represented by the attractive and repulsive forces
3. Move nodes until there is minimum energy
4. F&R: "temperature adjustment" - adjustments become smaller as layout improves



Purchase, Stirling, Archambault. "Proximity, Communities, and Attributes in Social Network Visualisation”, 2020

Trees



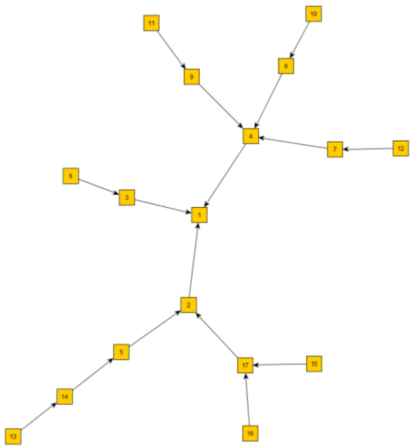
Any two nodes are connected by only one path

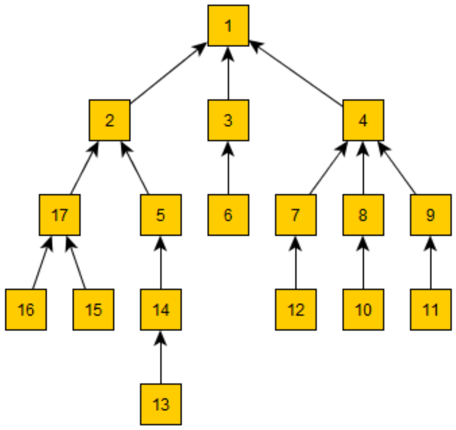
Every node has one unique parent

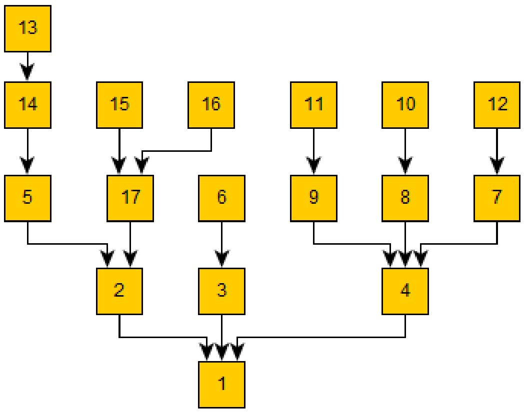
Connected: a path between all pairs of nodes

The number of edges is one less than the number of nodes

Directed or undirected





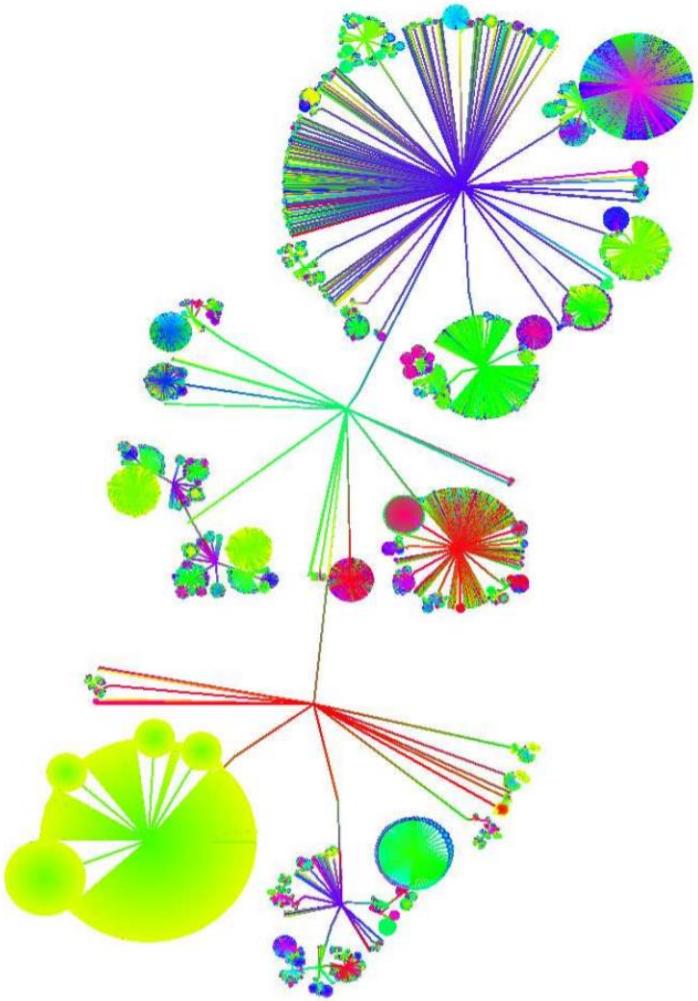


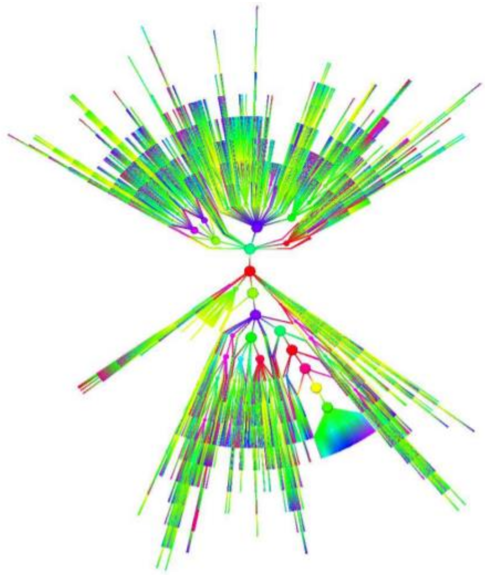
yEd graph editor (circular, organic, hierarchical, tree layout)

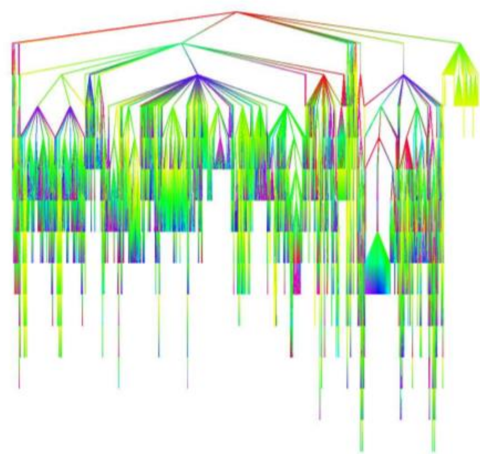
Other Tree approaches

- **Bubble trees**: S. Grivet, D. Auber, J.-P. Domenger, G. Melançon. Bubble Tree Drawing Algorithm. International Conference on Computer Vision and Graphics, 2004.
- **Scalable Tree Visualisation**: T. Munzner, F. Guimbretiere, S. Tasiran, L. Zhang, and Y. Zhou. TreeJuxtaposer: Scalable Tree Comparison Using Focus+Context with Guaranteed Visibility, ACM Transactions on Graphics 22(3):453-462, 2003.
- **Cone trees**: G. G. Robertson, J. D. Mackinlay, and S. K. Card, "Cone trees: Animated 3D visualizations of hierarchical information," ACM SIGCHI conference on Human Factors in Computing Systems, pp. 189–194, 1991.
- **Tree maps**: B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. IEEE Conference on Visualization '91, pp. 284-291, 1991.
  - **Cushion Tree maps:** J. J. Van Wijk and H. Van de Wetering, "Cushion treemaps: visualization of hierarchical information," *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis'99)*, pp. 73-78, 1999.

Bubble trees







maximising angular resolution
identifies symmetric sub-trees

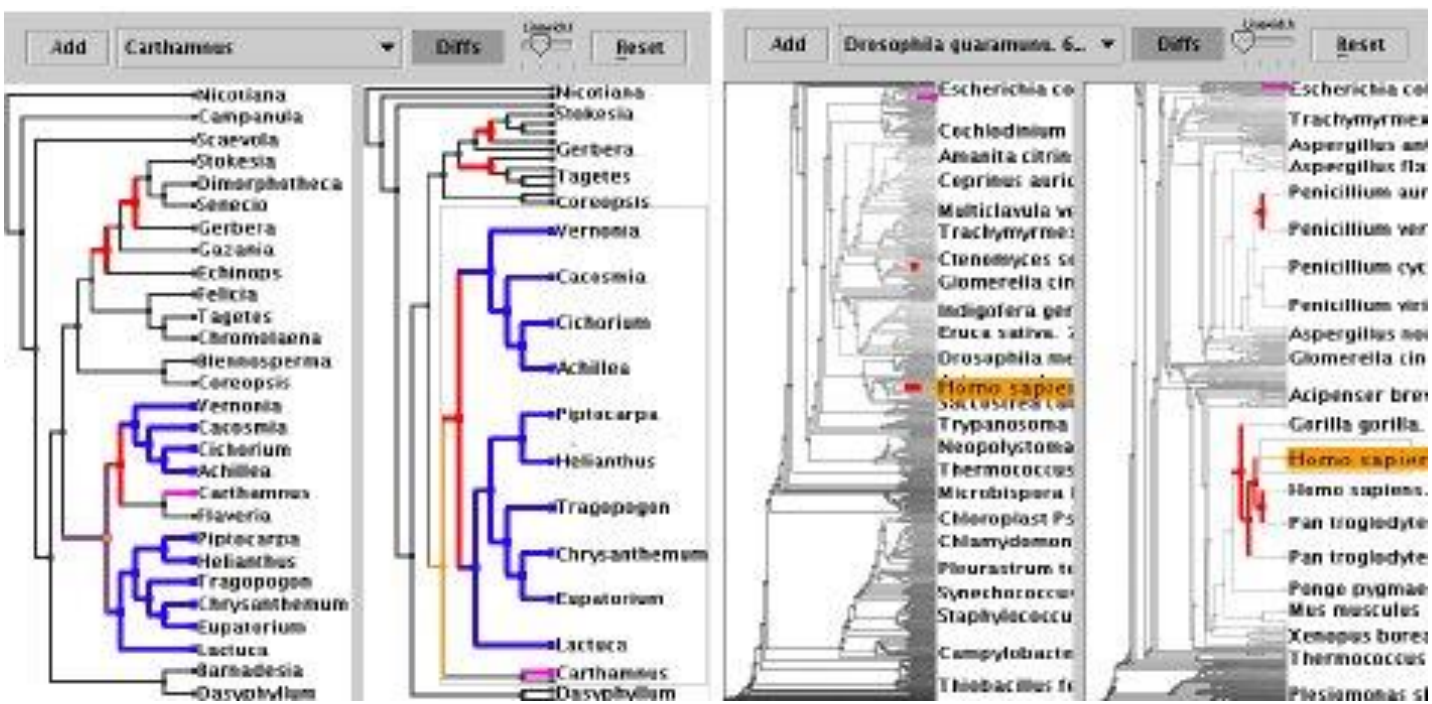
S.Grivet, D.Auber, J-P.Domenger, G.Melançon. Bubble Tree Drawing Algorithm, 2004

TreeJuxtaposer

Interactive tree comparisons

Choose areas to stretch, and areas to compress

"We have presented a system that allows interaction with and detailed structural comparisons between trees of over 100,000 nodes each, and browsing single trees of half a million nodes"



Munzner, Guimbretiere, Tasiran, Zhang, Zhou. TreeJuxtaposer: Scalable Tree Comparison Using Focus+Context with Guaranteed Visibility, 2003.

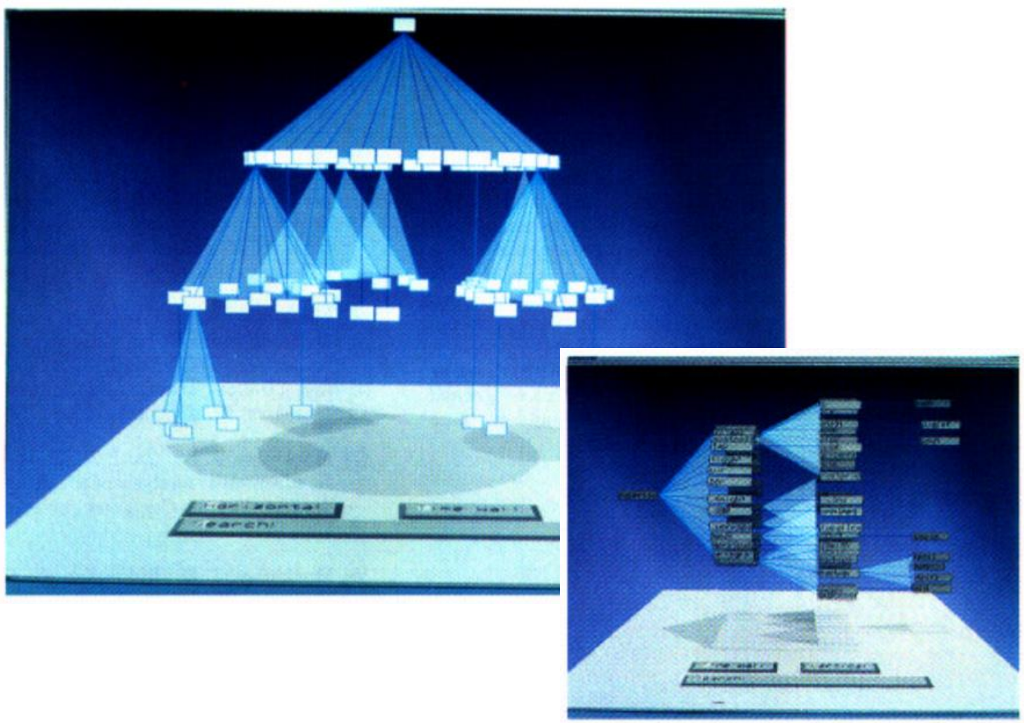
Cone Trees

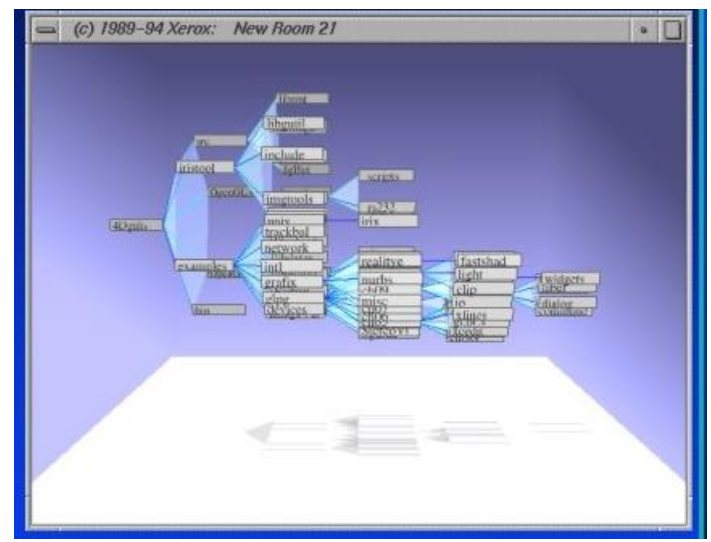
3D extension of typical tree visualisations

Tree levels are arranged on circular ‘disks’

Animations bring nodes of interest to the front (by spinning the disks)

Up to 10 levels, 1000 nodes





Robertson, Mackinlay, Card, “Cone trees: Animated 3D visualizations of hierarchical information,” 1991

Tree Maps

Space-filling representation

Hierarchical order shown by rectangle containment

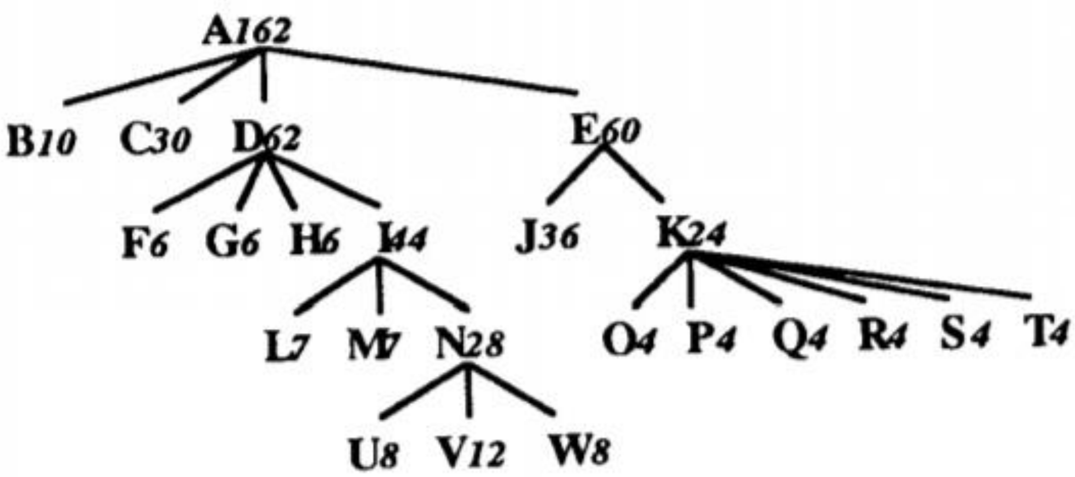


Figure 2. Tree Diagram

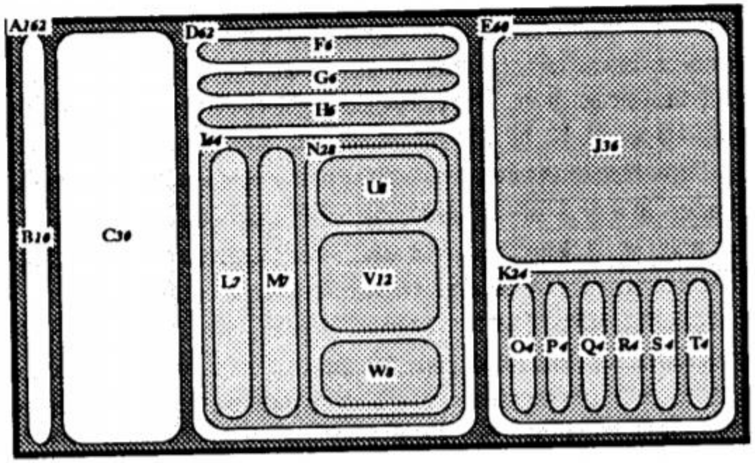


Figure 4. Nested Tree-Map

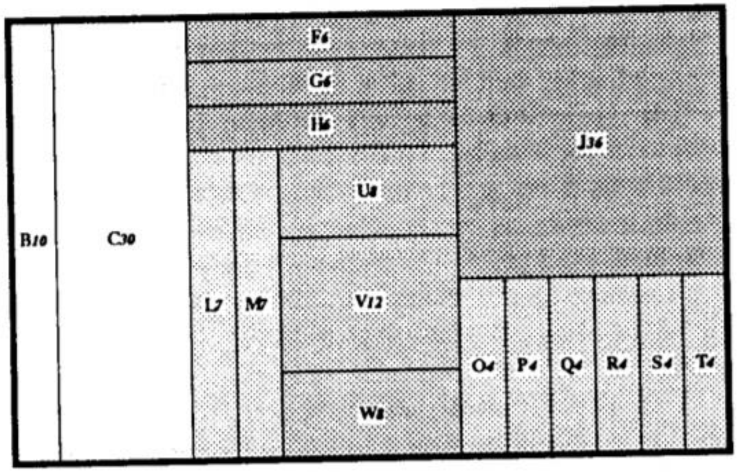
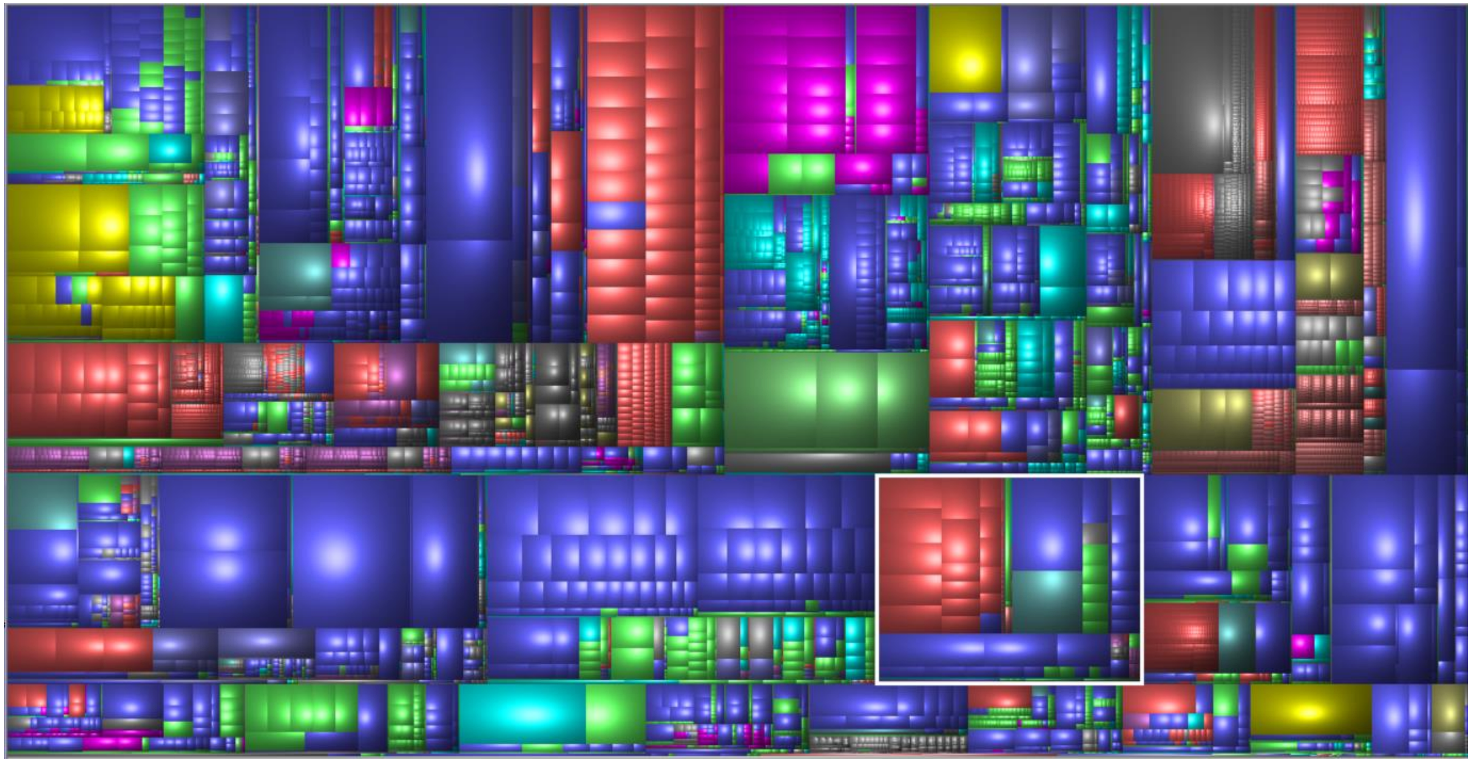


Figure 5. Tree-Map

Johnson, Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures, 1991.

Cushion Tree Maps



WinDirStat

File hierarchies on a disk

Colours: different file types (.pdf, .docx etc.)

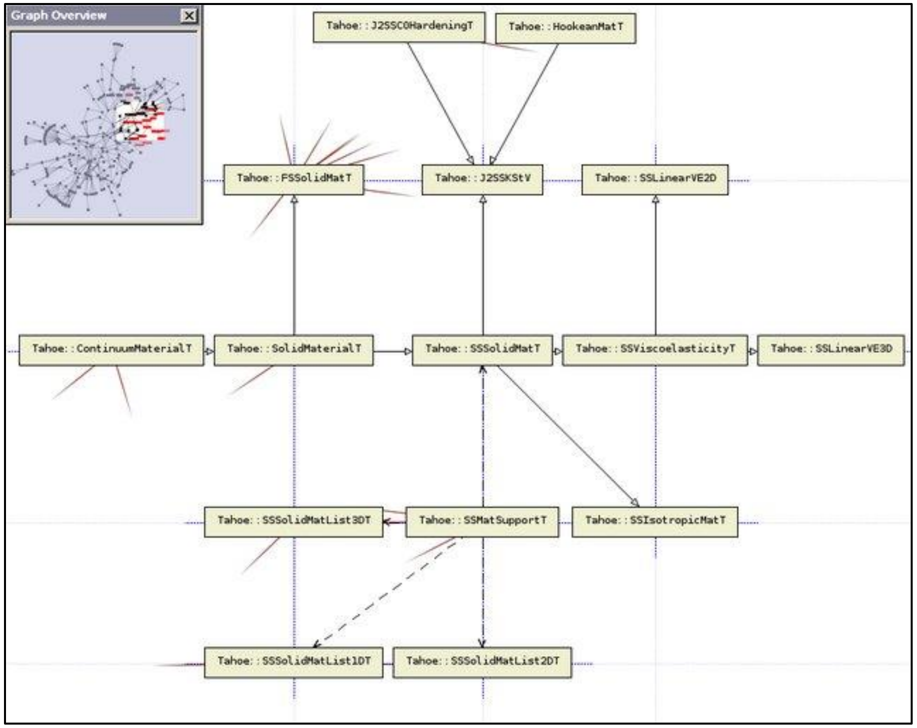
https://windirstat.net/

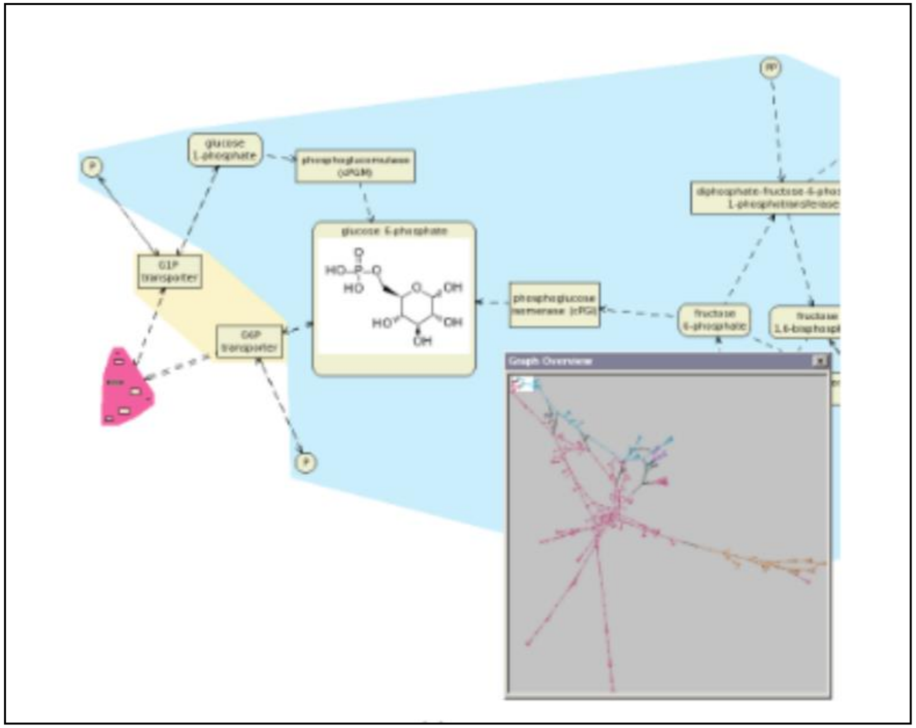
Graph Visualisation

Many interactive techniques for interaction with graphs are designed to address the problem of exploring very large graphs

- **Overview+Detail:** T. Dwyer, et al. "Exploration of Networks using overview+detail with Constraint-based cooperative layout," in IEEE TVCG, vol. 14, no. 6, pp. 1293-1300, 2008.
- **Edge Clustering:** W. Cui et al. “Geometry-Based Edge Clustering for Graph Visualization“. In Proceedings of Information Visualization 2008.
  - **Edge Bundling:** D. Holten. “Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data.” IEEE TVCG, 12(5):741-748, 2006.
- **Dense Networks:** A. Nocaj, M. Ortmann and U. Brandes: Untangling the Hairballs of Multi-Centered, Small-World Online Social Media Networks. Journal of Graph Algorithms and Applications 19(2):595-618, 2015.

Overview+Detail



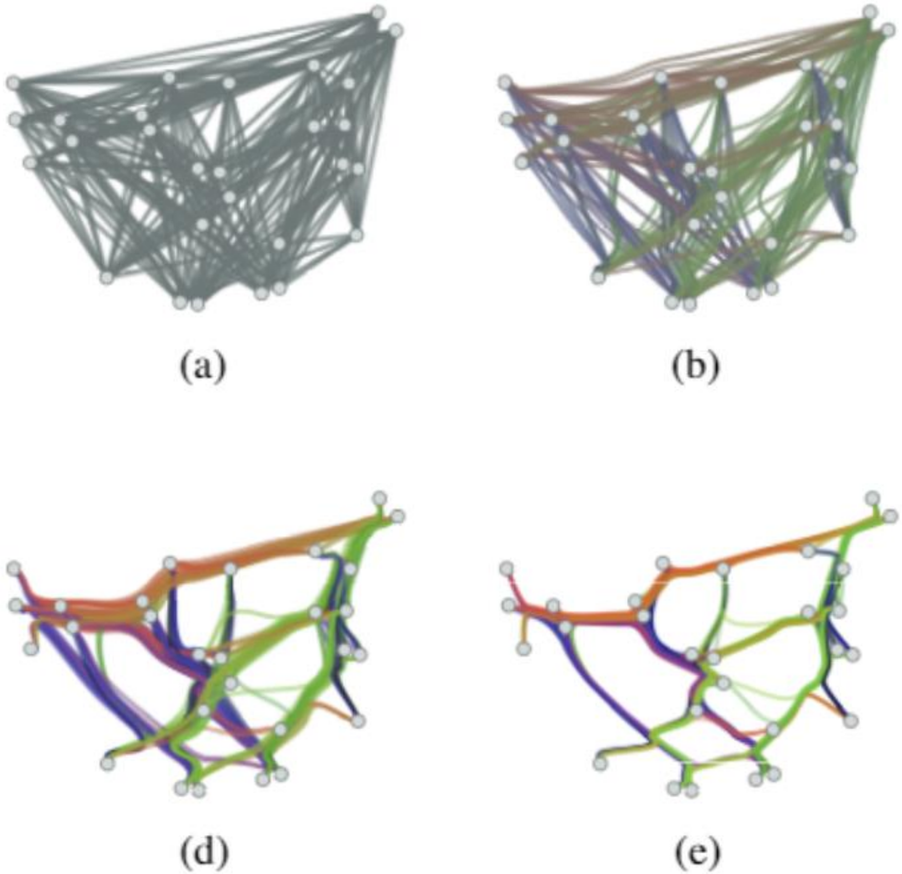


Small window shows the overview and context

Main display shows the detail

Dwyer et al., "Exploration of Networks using overview+detail with Constraint-based cooperative layout," 2008.

Edge Clustering



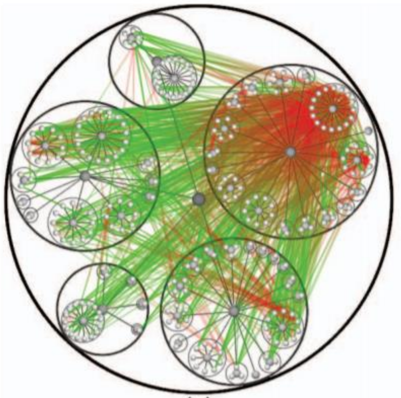
"capture the underlying edge patterns and generate informative and less cluttered layouts"

Groups edges into bundles; all edges in a bundle go through the same point

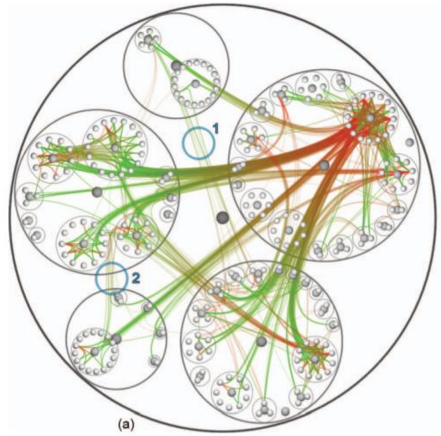
Loss of detailed information, but indicative structure shown (and details recoverable)

Cui et al. "Geometry-Based Edge Clustering for Graph Visualization", 2008.

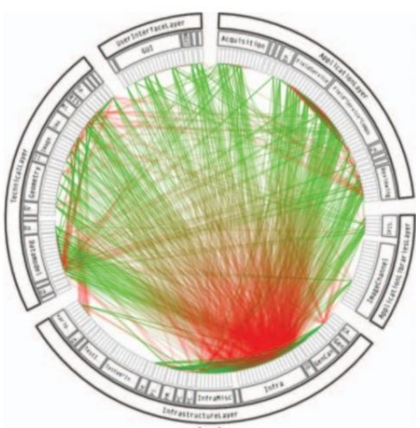
Edge Bundling



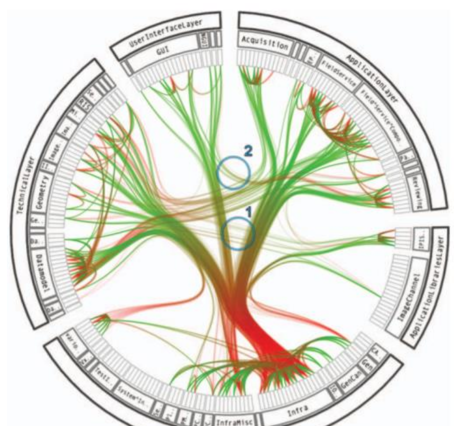
(a)



(a)



(a)



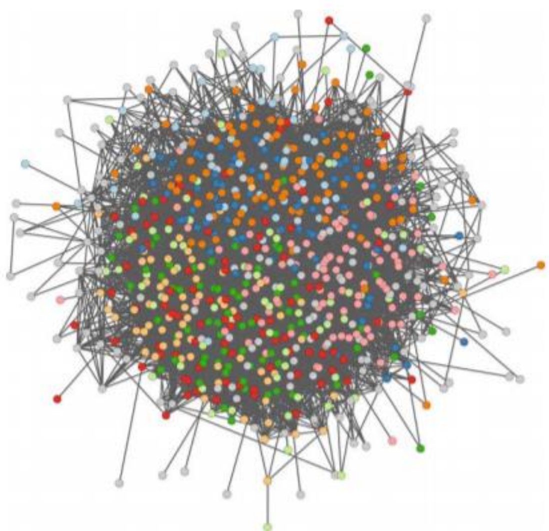
(b)

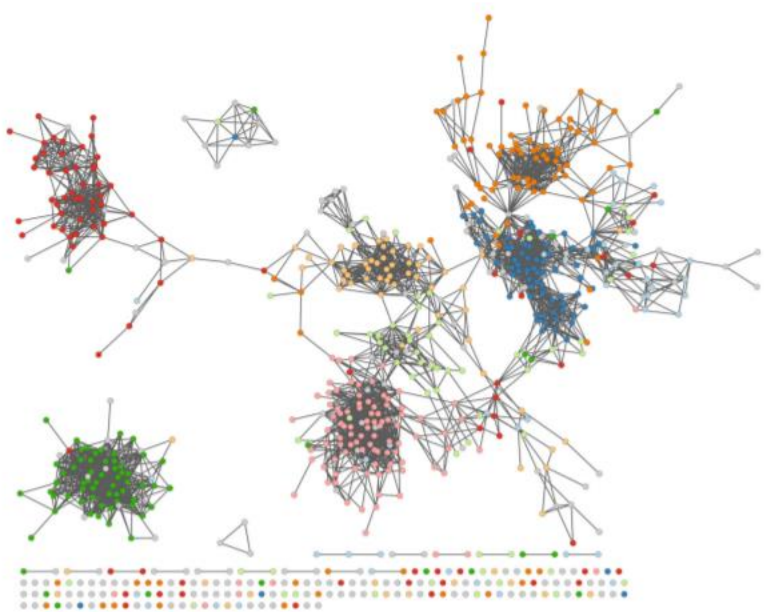
Particular focus on hierarchies

"quickly gaining insight in the adjacency relations present in hierarchically organized systems…aesthetically pleasing."

D. Holten. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data, 2006

Dense Networks





Identifying communities in social networks – finding the ‘backbone’ between strong communities

Only keep edges that are important:

- support short cycles (length 3)
- key in defining communities

Then use a layout algorithm on the reduced graph

Nocaj, Ortmann Brandes: Untangling the Hairballs of Multi-Centered, Small-World Online Social Media Networks, 2015.

- **Graphs:** abstract data structures
  - directed, undirected, connected, trees, planar
- **Graph drawings:** visual representations of graphs
  - node-link: algorithms, aesthetics
  - trees: node-link & space-filling
- **Graph visualisations:** interactive techniques for exploring graphs

Depicting Networks and Trees