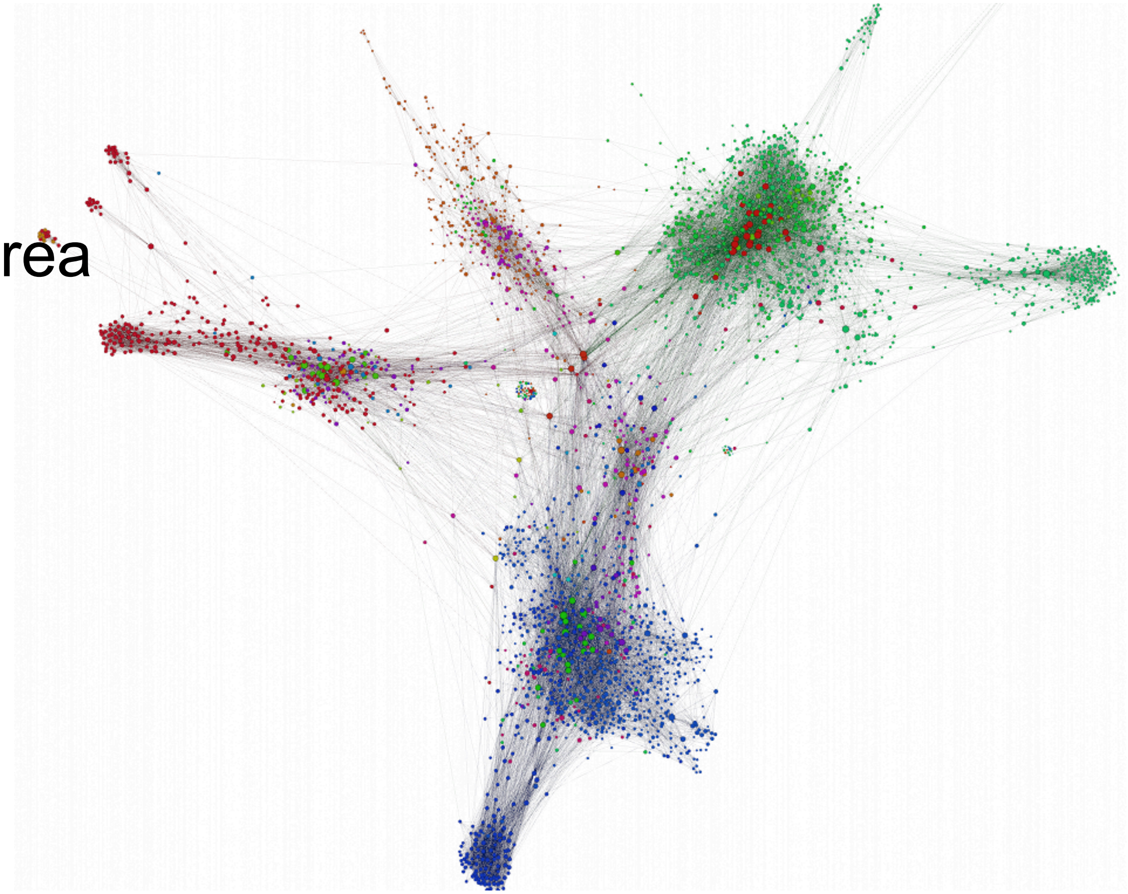
Module #7b:
Graph Visualization

Key Issues

- Size of graph
- Planarity
- Minimization of graph area
- Aesthetics
- Algorithm complexity
- Predictability



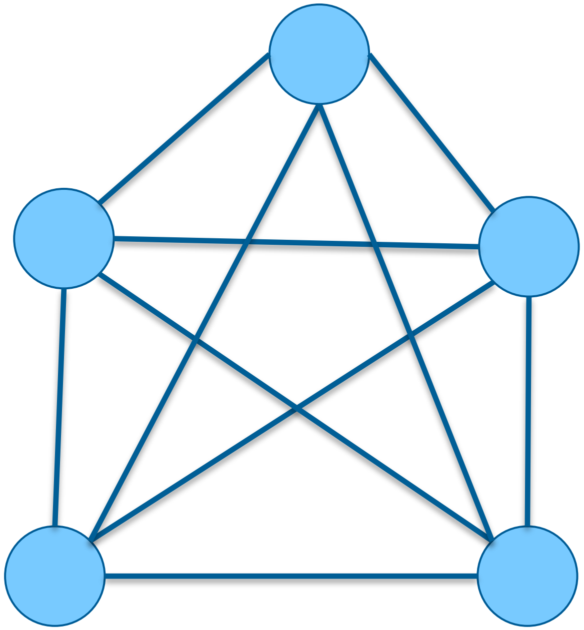
Visualization Challenges

Minimize

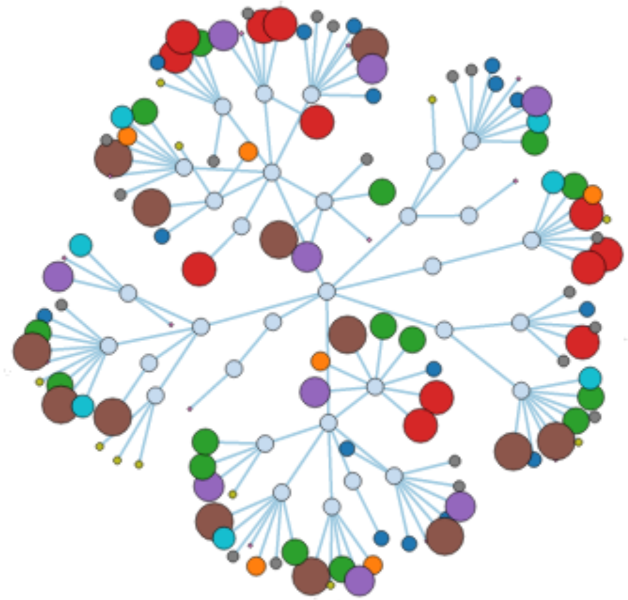
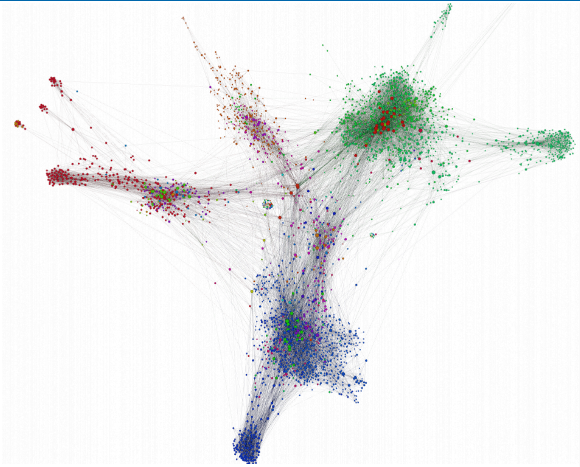
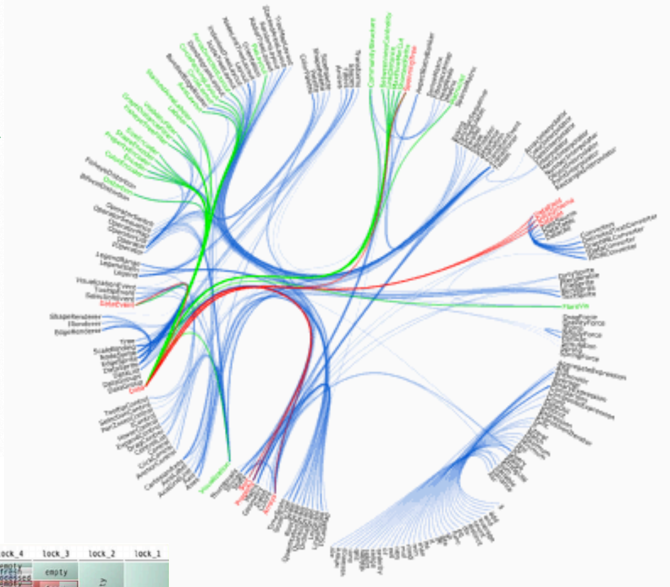
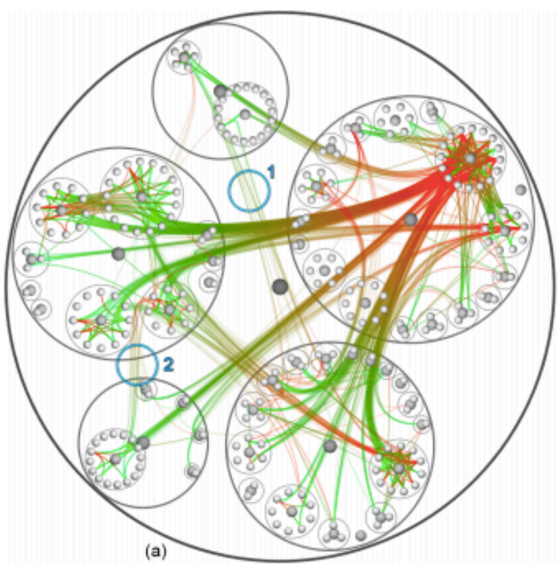
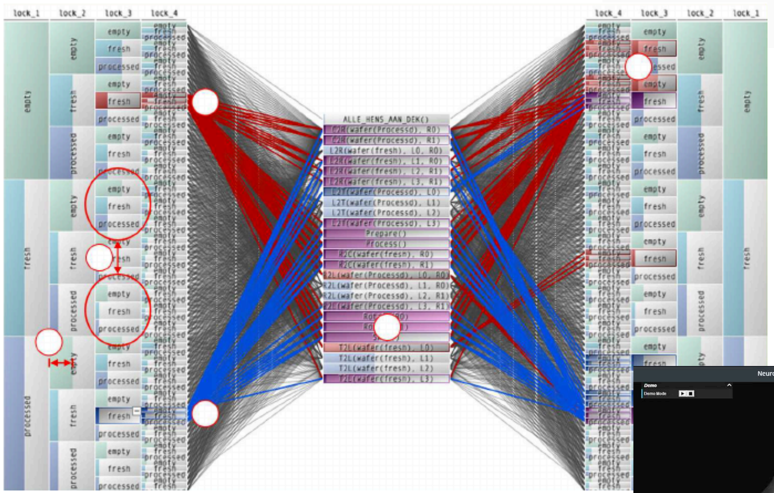
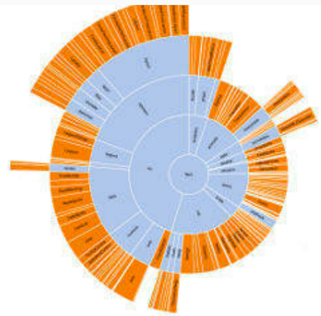
- edge crossings
- area
- line bends
- line slopes
- total edge length
- max edge length
- edge length variance

Maximize

- smallest angle between
- edges symmetry



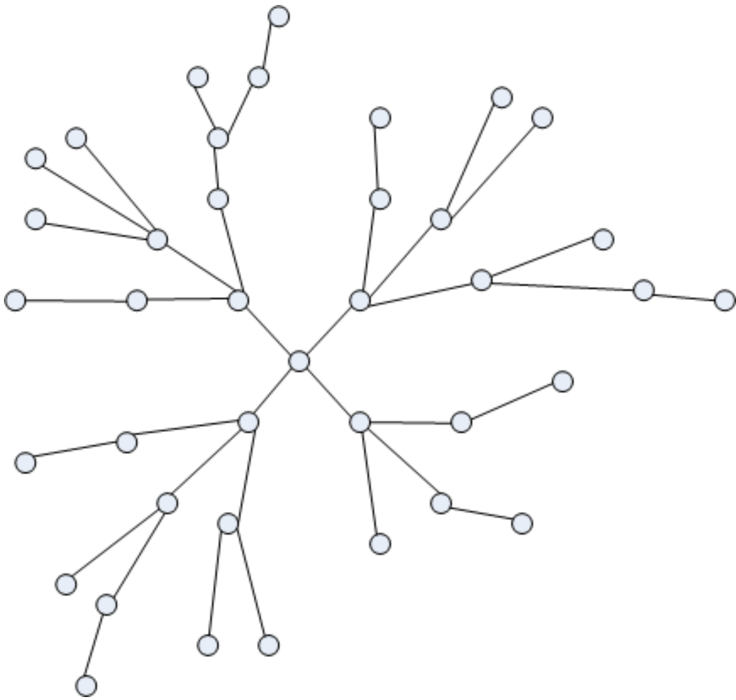
Graph Visualization


Node-link diagrams

- Node-link diagrams: the trivial way to graph data
  - Represent the relations among the data elements in form of lines

- Focus on:
  - the computation of the coordinates of the nodes
  - the representation of the lines.



Node-link diagrams

- A clean layout should comply with:
  1. Nodes and edges should be evenly distributed.
  2. Edge-crossings should be minimized.
  3. Depict symmetric sub-graphs in the same way.
  4. Minimize the edge bending ratio
  5. Minimize the edge lengths, which helps readers detecting the relations among different nodes faster.

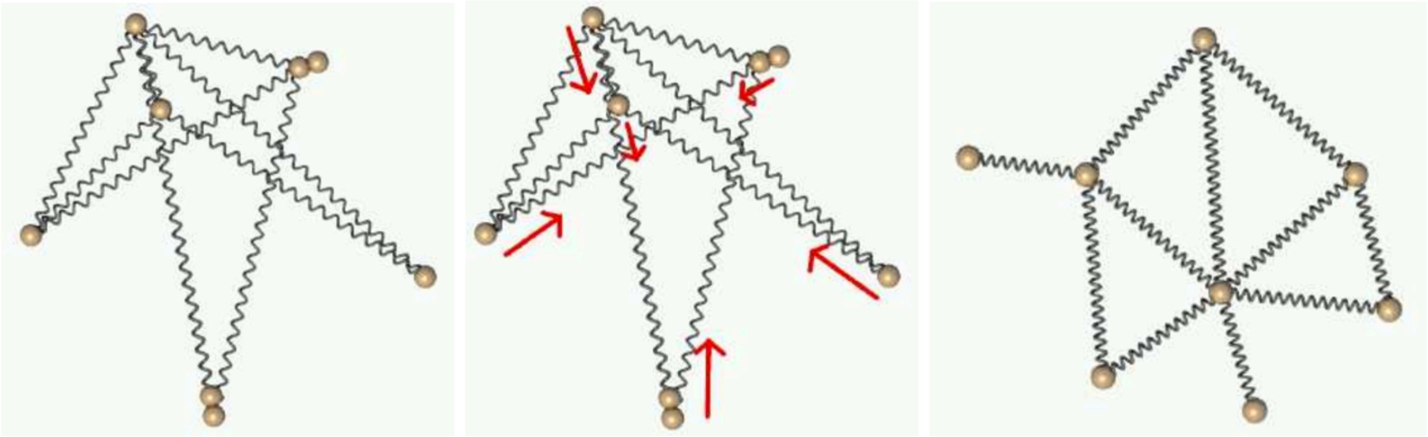
- In cases where the data is inherently structured distribute the nodes into different layers.
  - This increases the understandability of the underlying graph.
  - Ex: in data-flow diagrams it is recommended to separate the graph elements into different layers in a way that the final representation reflects the original semantics.

Spring Layout Algorithm

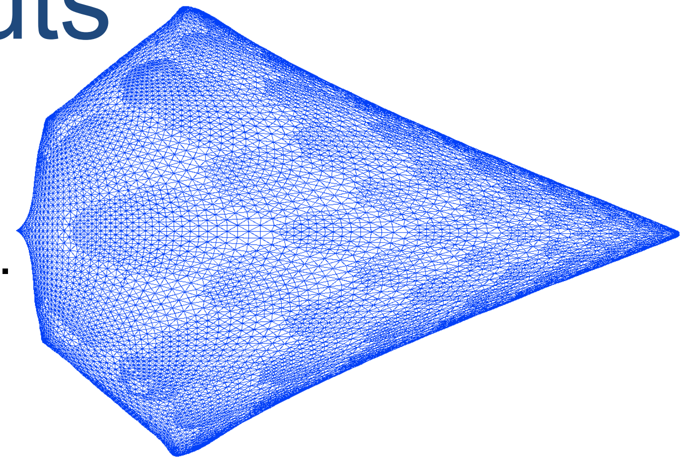
- The spring layout algorithm is widely known as **force-directed layout.**
- Originally proposed by Eades in 1984
- One of the most popular node-link layouts due to:
  - Simplicity
  - Ability to produce a symmetric layout
  - Good-quality results
  - Intuitive
  - Interactivity
  - Strong theoretical foundations
- Disadvantages:
  - Running time O(n3)
  - Local minima

Spring Layout Algorithm

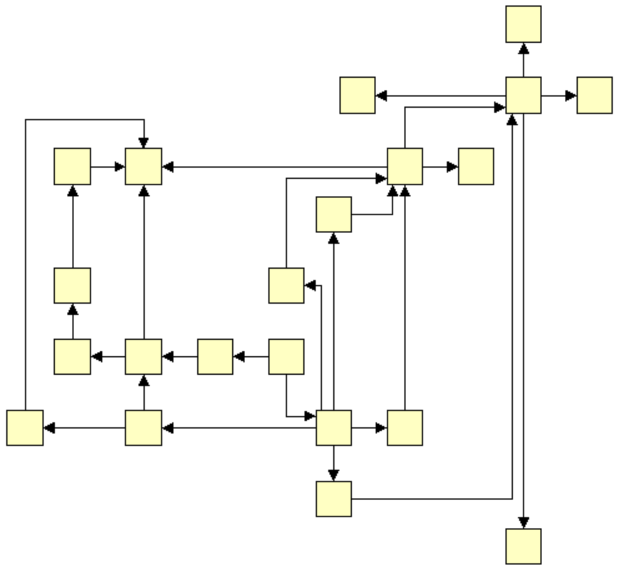
- The spring layout algorithm represents the graph as a physical system
  - graph nodes considered charged particles connected to each other using a set of springs
  - each node is associated with two types of forces:
    1. Attraction forces
    2. Repulsive forces
- The attraction force is applied to the neighbor nodes which are connected by a spring, while the repulsive force is applied to all graph nodes



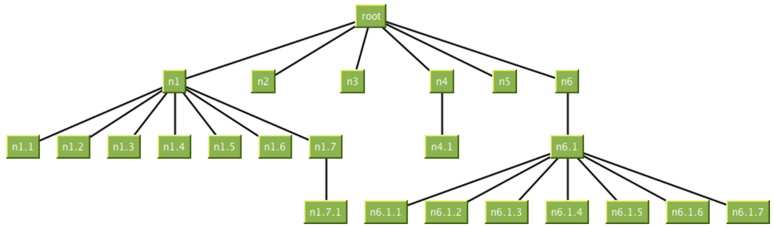
Graph Layouts



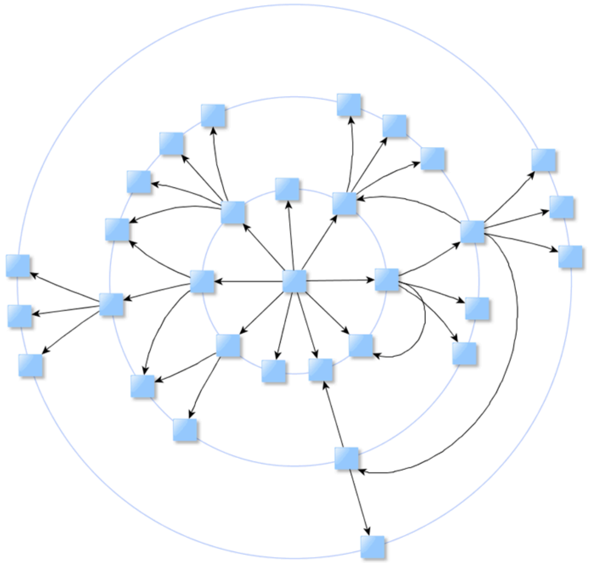
- **Spectral layout:** use as coordinates the eigenvectors of a matrix such as the Laplacian derived from the adjacency matrix of the graph.



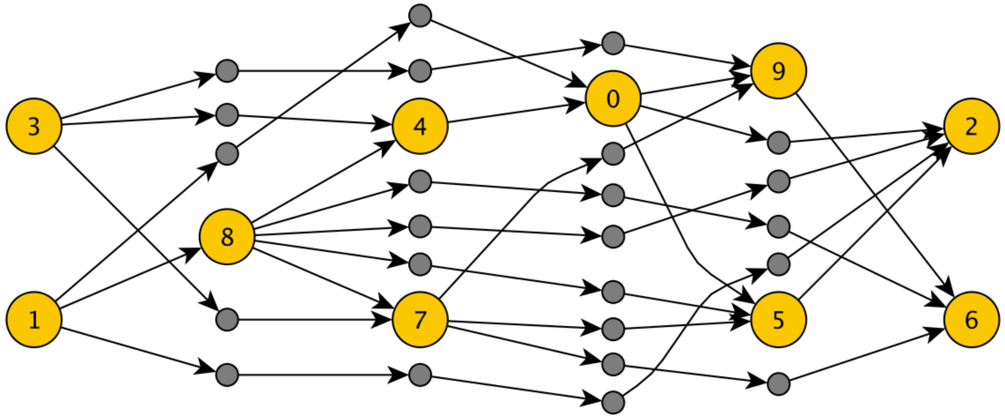
- **Orthogonal layout:** allows the edges of the graph to run horizontally or vertically, parallel to the coordinate axes of the layout
- **Tree layout** algorithms these show a rooted tree-like formation



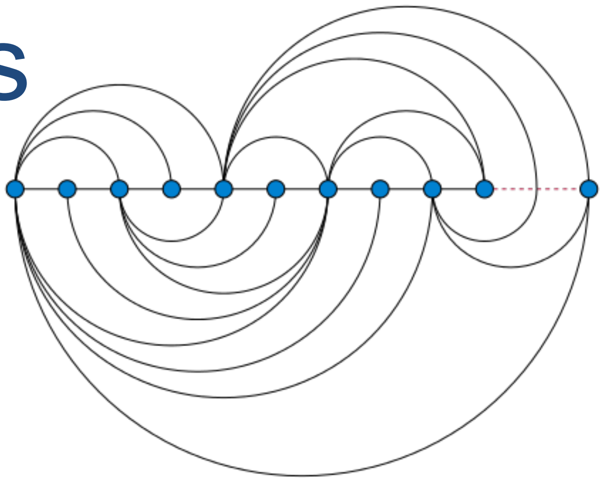
Graph Layouts



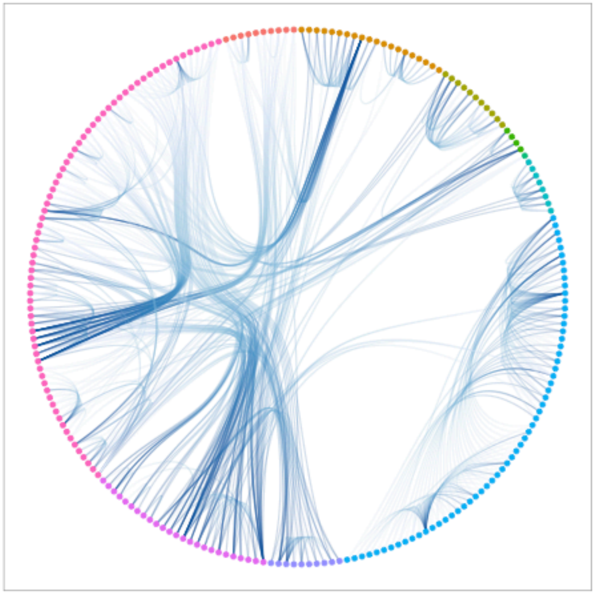
- **Balloon layout**: children of each node in the tree are drawn on a circle surrounding the node, with the radii of these circles diminishing at lower levels in the tree so that these circles do not overlap.
- **Layered graph:** drawing methods (often called Sugiyama-style drawing) directed acyclic graphs or graphs that are nearly acyclic.



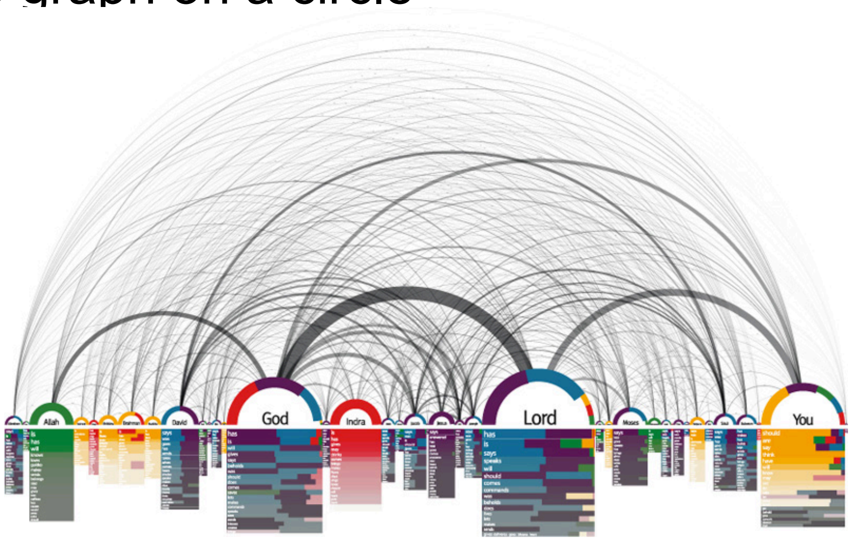
Graph Layouts



- **Arc diagrams:** a layout proposed in the 1960s where vertices are placed on a line; edges may be drawn as semicircles above or below the line.



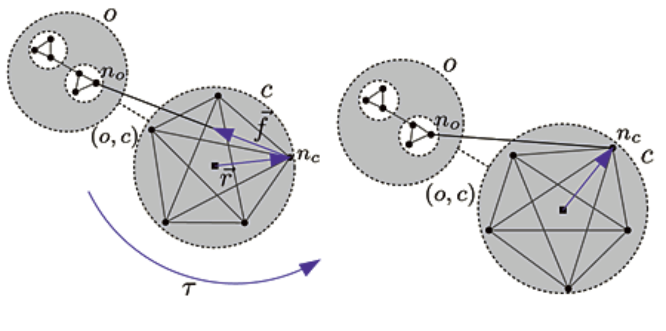
- **Circular layout:** methods place the vertices of the graph on a circle



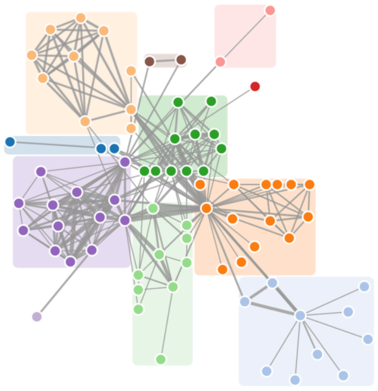
Topological Feature-Based Layout

- Topological feature-based layout: multi-level, feature-based approach.

- Four main steps:
  1. **Decomposition phase:** graph is decomposed into many sub-graphs based on the topological features of each internal sub-graph.
  2. **Feature layout phase**: meta-nodes or the grouped sub-graphs are laid out using one of the layout algorithms
  3. **Crossing reduction phase:** eliminate the crossing ratio in the produced layout.
  4. **Overlap elimination phase**: change the node sizes in the final layout to ensure that no nodes overlap each other.



(a) (b)



JOHNS HOPKINS
WHITING SCHOOL of ENGINEERING

© The Johns Hopkins University 2016, All Rights Reserved.