Data Exploration & Visualization

Module 12

Graph Visualization

Dr. ZENG Wei

DSAA 5024

The Hong Kong University of Science and Technology (Guangzhou)

Data Exploration & Visualization

Module 12: Graph Visualization

- Graph data
  - representation, terminology, tasks, challenges
- Graph visualization
  - Node-link graph: layout, graph simplification
  - Matrix visualization
  - Hybrid visualization
  - Graph visualization tools

Examples of graph

- Connections throughout our lives and the world
  - Circle of friends
  - World Wide Web
  - Road network
  - Semantic map in an AI algorithm
  - …
- Model connected set as a Graph
- Graph/network visualization is one of the oldest and most studied areas of InfoVis

Graph representation

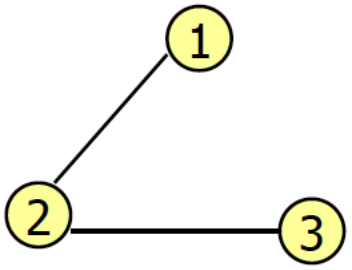
- A graph $G = (V, E)$
  - Vertices (nodes) $V := \{v_i\}$
    - Degree of vertex: number of edges that are incident to the vertex $\deg(v)$
  - Edges (links) $E \sqsubseteq \left\{ \{v_i, v_j\} \,\middle|\, v_i, v_j \in V \text{ and } i \neq j \right\}$
    - Undirected or directed

1: 2
2: 1, 3
3: 2

adjacency list

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

adjacency matrix



node-link

Graph terminology

- Graphs can have *cycles*
  - Trees are *acyclic*
- 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)

Tasks for graph visualization

- 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

Tasks for graph visualization

- 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

Tasks for graph visualization

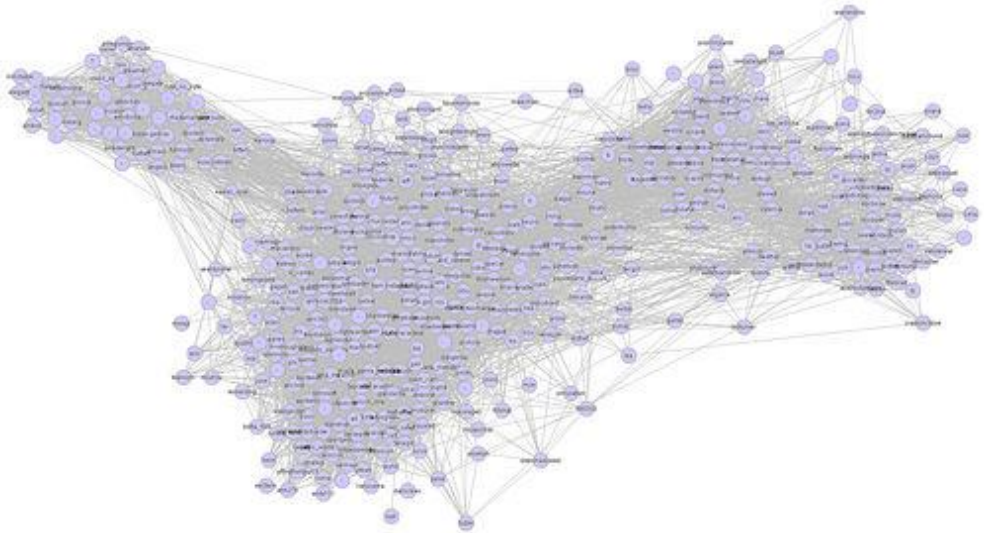
- Browsing tasks
  - Follow path
    - Follow a given path
  - Revisit
    - Return to a previously visited node
- Overview task
  - Compound exploratory task
    - Estimate size of a network
    - Find patterns

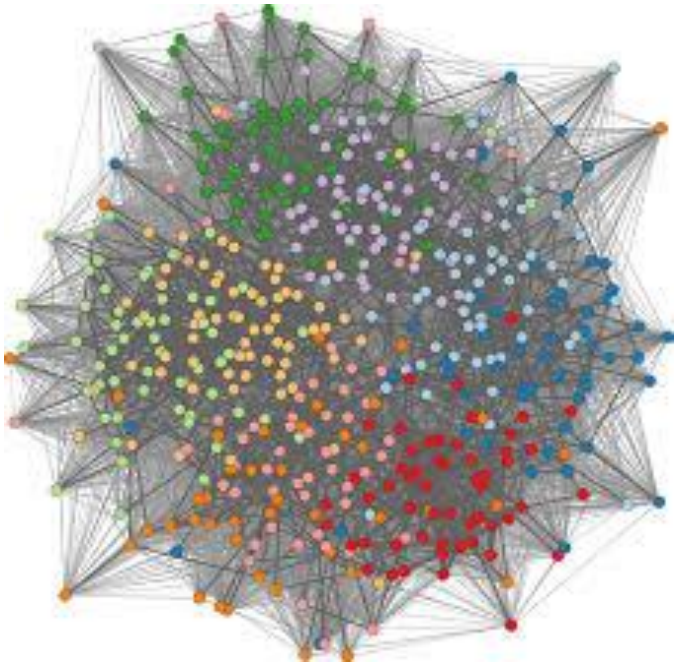
Challenges for graph visualization

- Graph layout and positioning
  - Make a concrete rendering of abstract graph
- Navigation/Interaction
  - How to support user changing focus and moving around the graph?
  - How do we allow a user to query, visit, or move around a graph?

Challenges for graph visualization

- Scalability challenge
  - May run out of space for vertices and edges (turns into “ball of string”)
    - With enough vertices and enough edges, you get…
    - A hairball! (ball of string)
  - Can really slow down algorithm





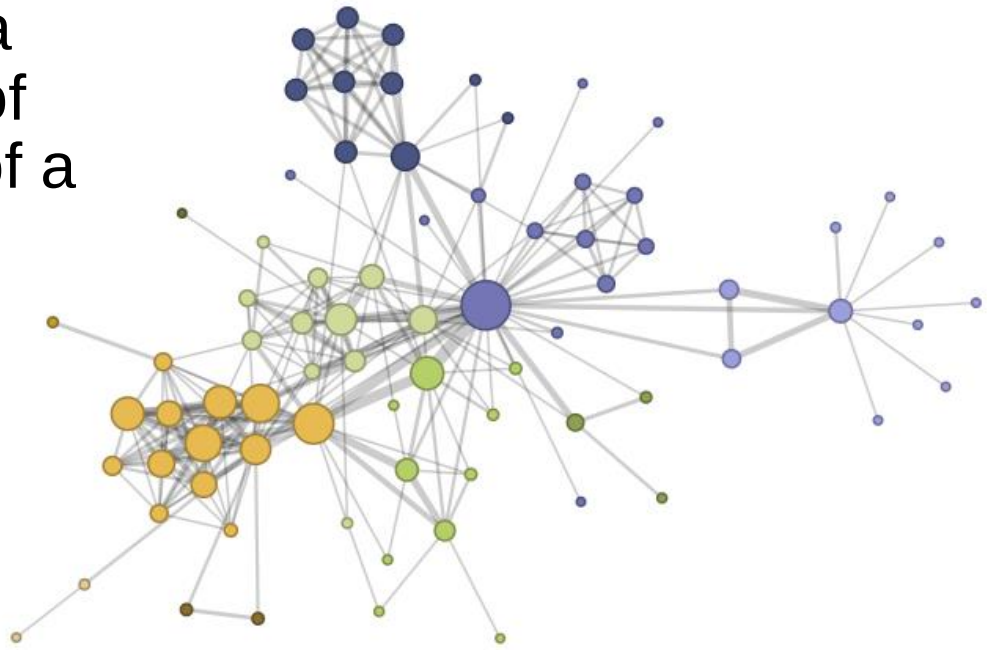
Data Exploration & Visualization

Module 12: Graph Visualization

- Graph data
  - representation, terminology, tasks, challenges
- Graph visualization
  - Node-link graph: layout, graph simplification
  - Matrix visualization
  - Hybrid visualization
  - Graph visualization tools

Node-link graph

- A drawing of a graph is a pictorial representation of the vertices and edges of a graph
  - Vertex Issues
    - Shape, Color, Size, **Location**, Label
  - Edge issues
    - Color, Size, Label
    - Form: Polyline, straight line, tapered, grid, curved, planar, upward/downward, ...



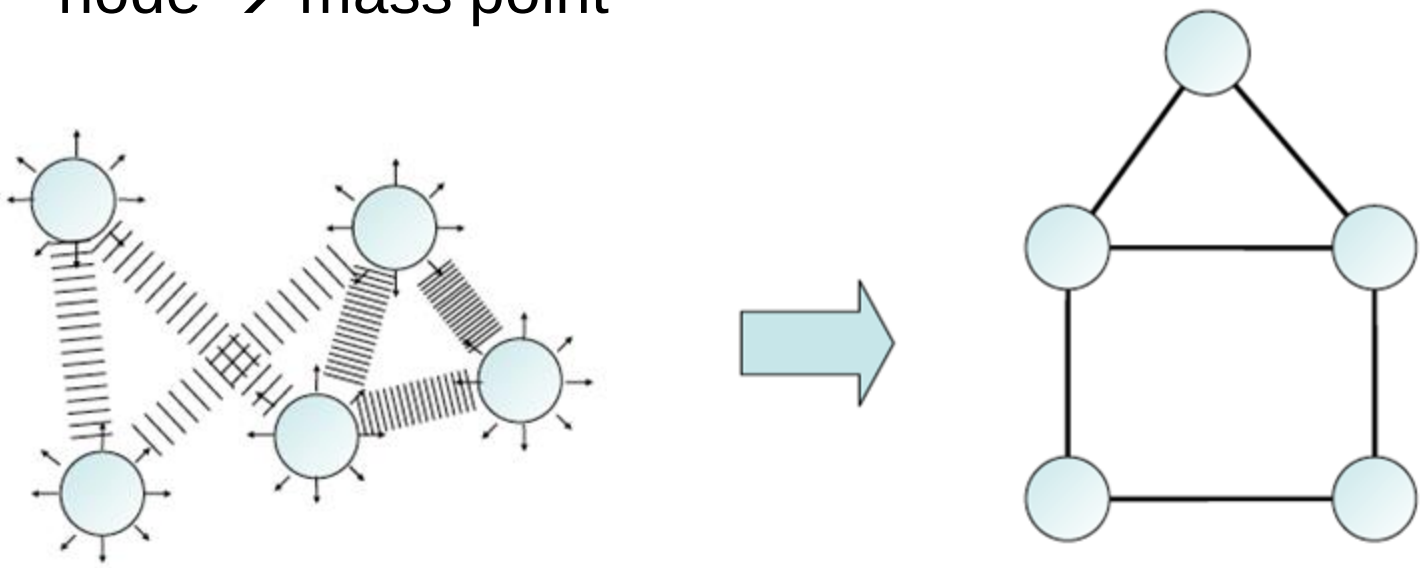
Character relations in Les Miserables
http://hci.stanford.edu/jheer/files/zoo/

Common layout techniques

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

Force-directed layout

- What about graphs without intrinsic order?
- Physical model:
  - edge → spring
  - node → mass point



Force-directed layout

- Peter Eades introduced the layout in 1984 in the paper ‘A Heuristic for Graph Drawing’
- With the effects of **spring force and gravitation**, nodes far away will be dragged near and vice versa
- The layout reaches a **balance** and becomes stable after **iterations**.

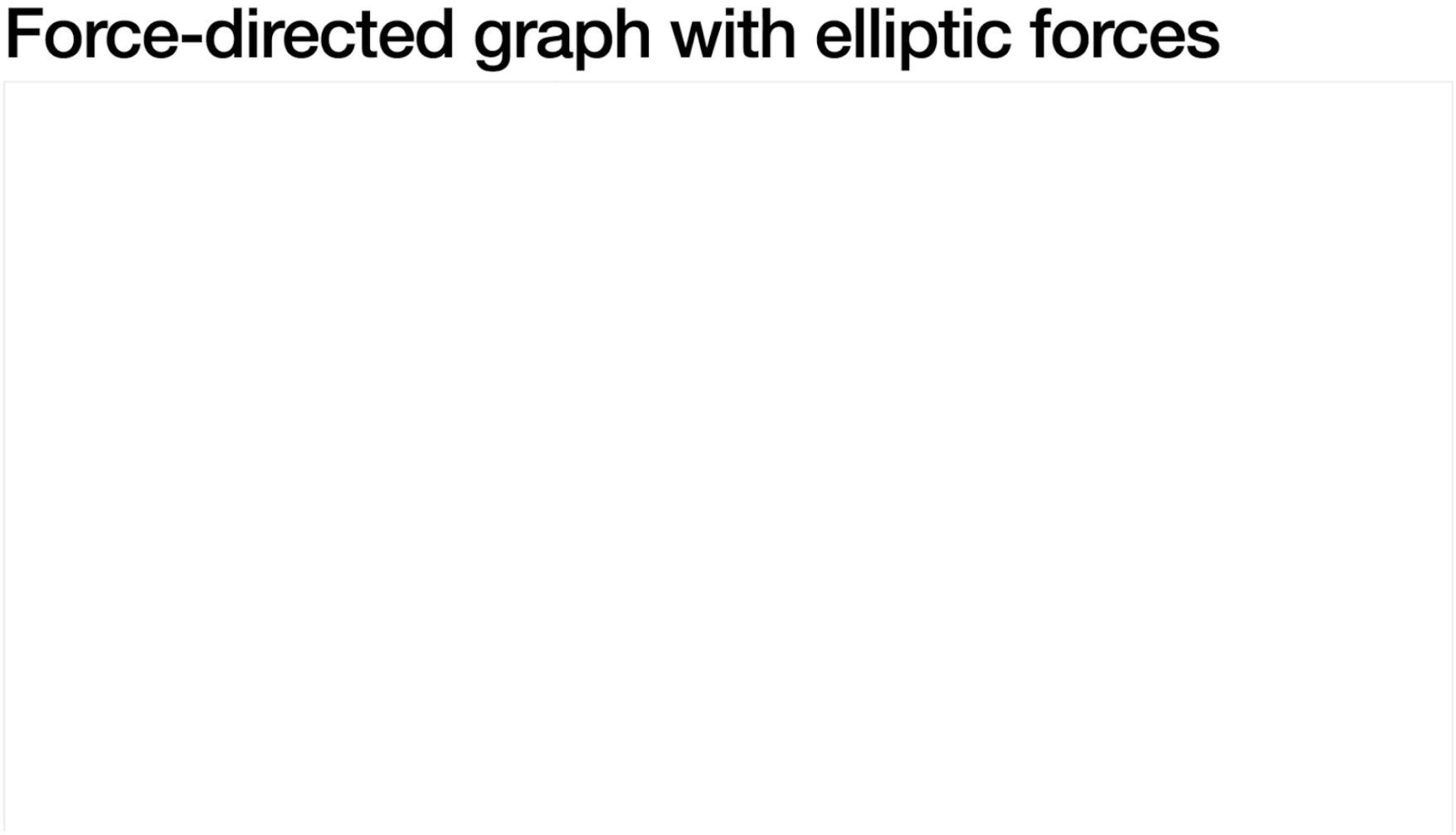
- Spring Model: $$E_s = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{1}{2} k(d(i,j) - s(i,j))^2$$

- Energy Model: $$E = E_s + \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{r w_i w_j}{d(i,j)^2}$$

Force-directed layout

- Starting positions: random or initial **configuration**
- Loop:
  - Compute **the repulsion and attraction** force for every pair of nodes
  - Accumulate the **force** (vector) for every node
  - Update nodes position **step by step** according to their forces
- Loop stops when the layout is “good enough”

Force-directed layout



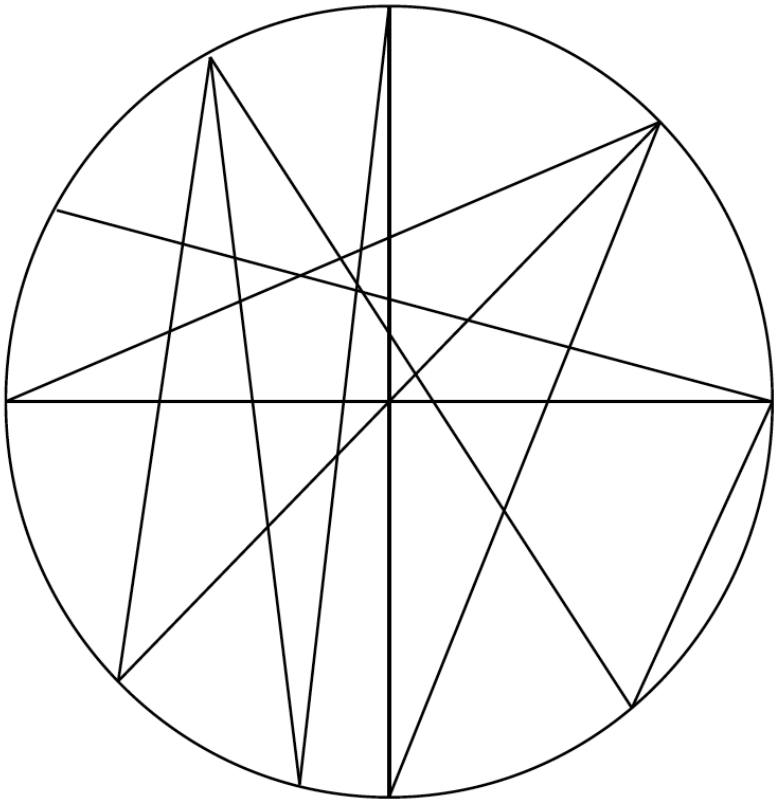
https://mbostock.github.io/protovis/ex/force.html

Force-directed layout

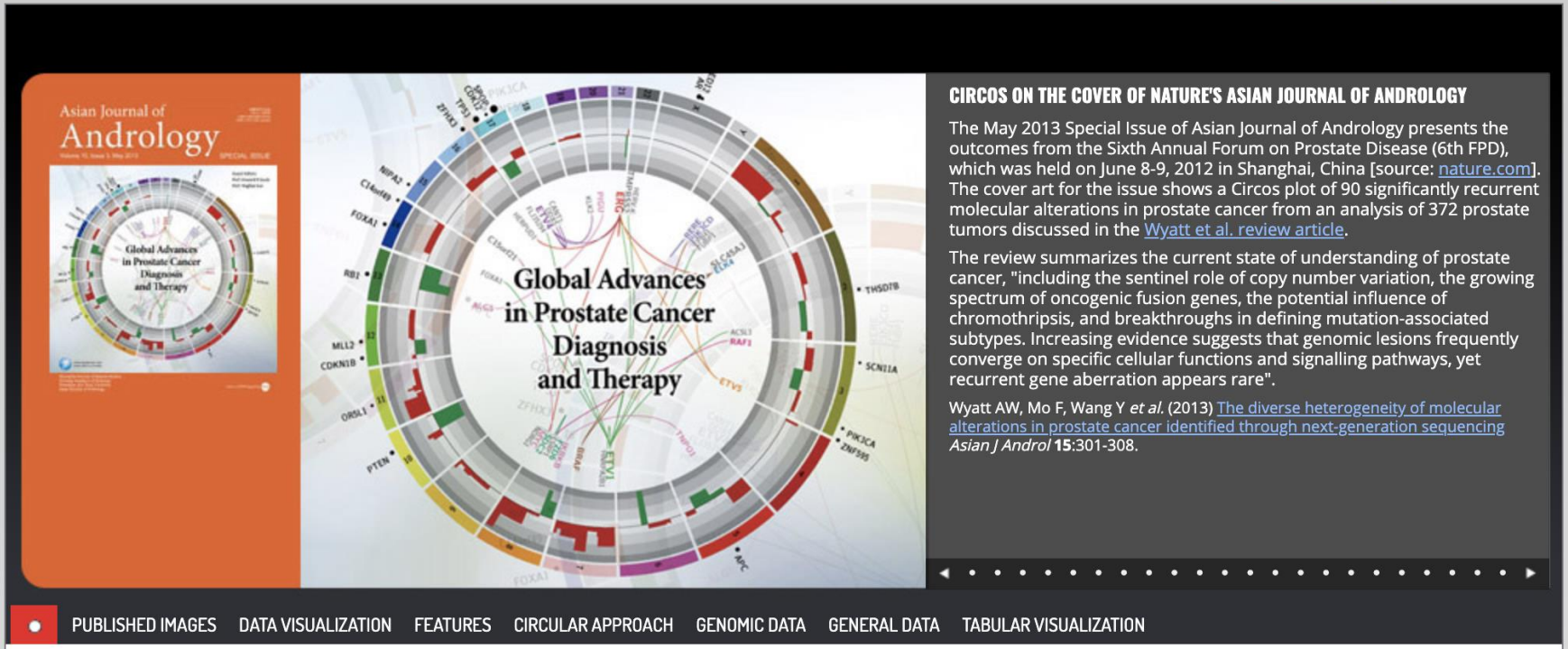
- Advantages:
  - Very flexible for any type of graphs
  - Forces can be customized
  - Easy to implement
- Disadvantages:
  - Local optimal
  - Initial configuration is important
  - Computation complexity of iterative algorithm
    - each iteration is *O(n log(n))*
- Extensions: FADE, GRIP, FMS, FM3, GVA

Circular layout

- Ultra-simple
- May not look so great
- Space vertices out around circle
- Draw lines (edges) to connect vertices
- Uses curved lines and becomes "chord diagrams"



Circos at the EMBO NGS workshop in Tunis, Sept 15–25.



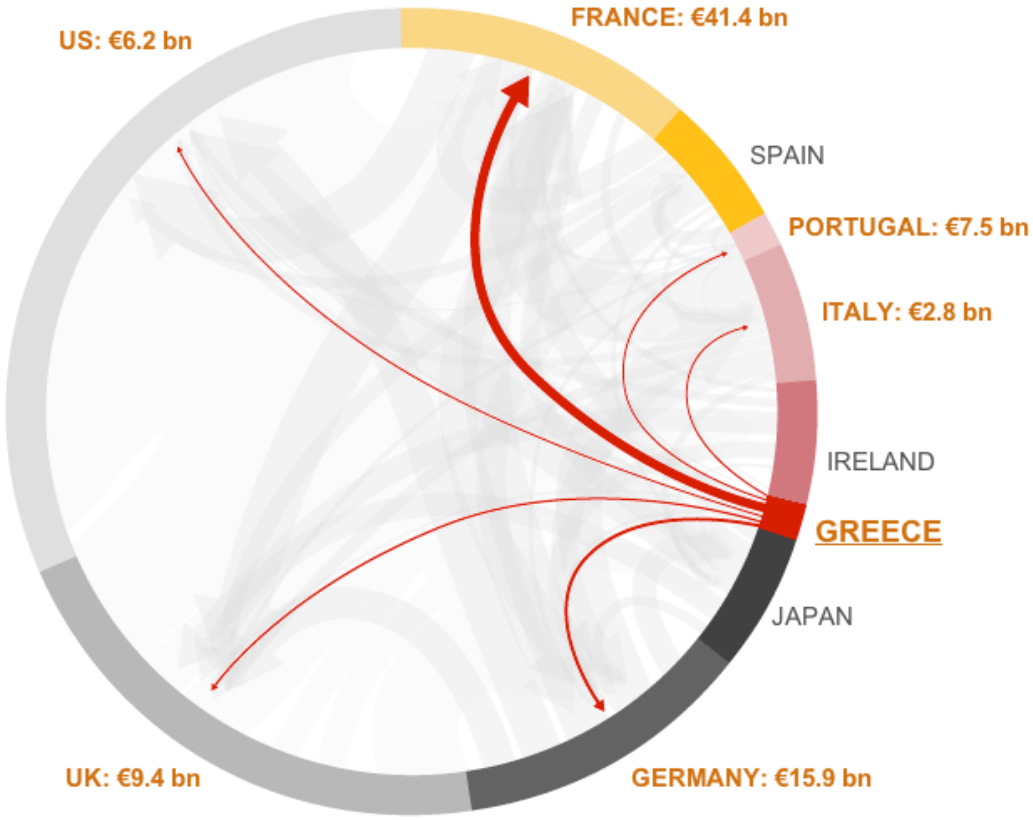
WHAT IS CIRCOS?

CIRCULAR VISUALIZATION

Circos is a software package for visualizing data and information. It visualizes data in a circular layout — this makes Circos ideal for exploring relationships between objects or positions. There are other reasons why a circular layout is advantageous, not the least being the fact that it is attractive.

Circos is ideal for creating publication-quality infographics and illustrations with a high data-to-ink ratio, richly layered data and pleasant symmetries. You have fine control each element in the figure to tailor its focus points and detail to your audience.

Circular layout



GREECE

GDP: €0.2 tn

Foreign debt: €0.4 tn

€38,073
Foreign debt per person

252% Foreign debt to GDP

166% Govt debt to GDP

Risk Status: HIGH

Greece is heavily indebted to eurozone countries and is one of three eurozone countries to have received a bail-out. Although the Greek economy is small and direct damage of it defaulting on its debts might be absorbed by the eurozone, the big fear is "contagion" - or that a Greek default could trigger a financial catastrophe for other, much bigger economies, such as Italy.

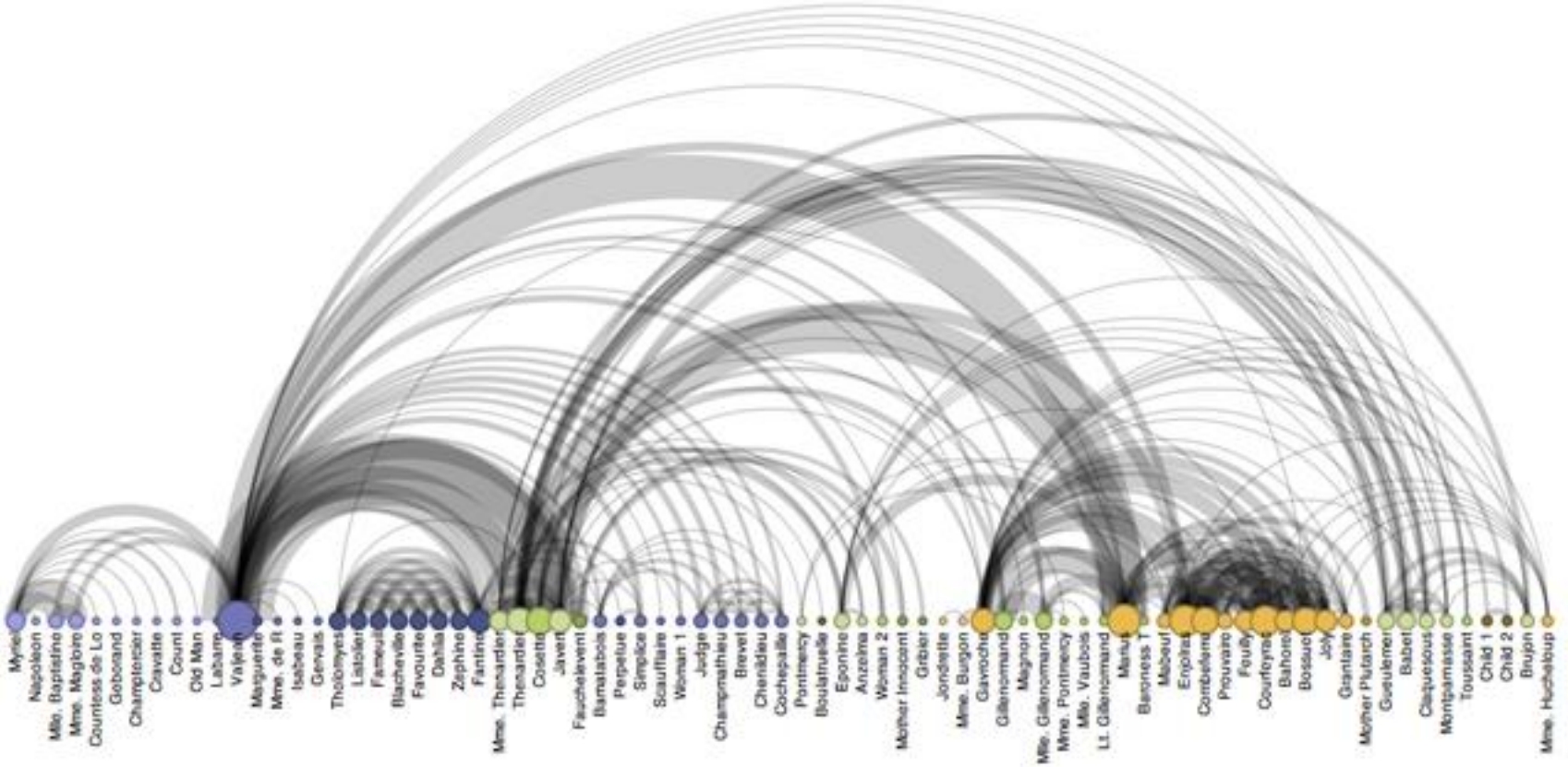
Back to introduction

Source: Bank for International Settlements, IMF, World Bank, UN Population Division

Eurozone debt web: Who owes what to whom?

http://www.bbc.co.uk/news/business-15748696

Arc diagram layout

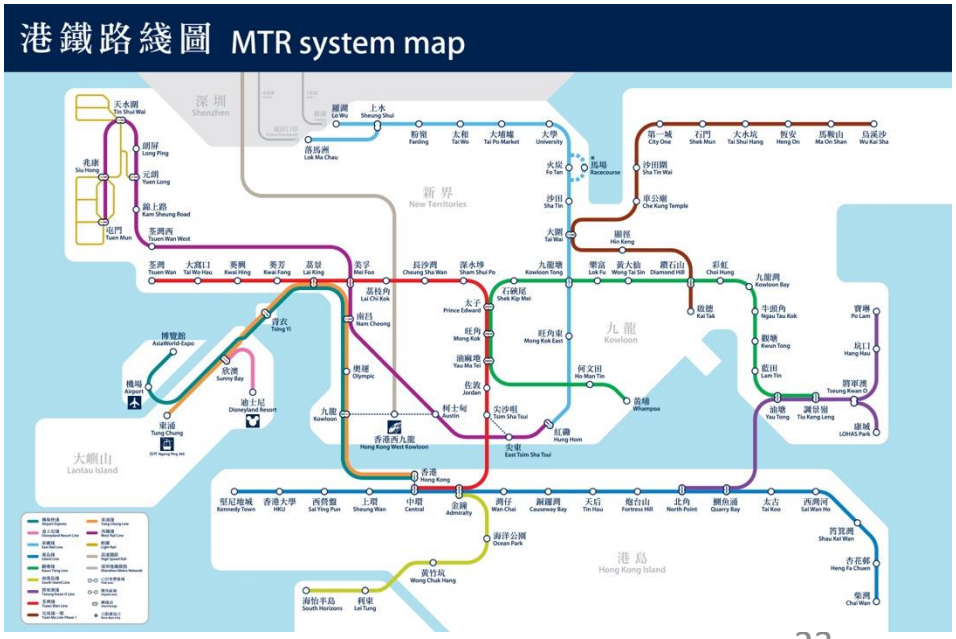


Character relations in Les Miserables

http://hci.stanford.edu/jheer/files/zoo/

Subway diagrams

- Geographic landmarks largely suppressed on maps
- These are more ***graphs*** than maps!
- Subway-style diagrams have become their own genre of network layouts



Subway diagrams

- “The genius of the London Tube Map”, Design legend Michael Bierut @ TED 2018

Aesthetic considerations

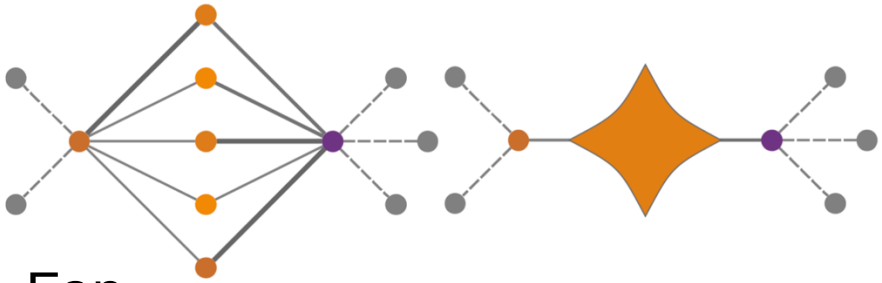
- A set of metrics to quantitatively rate the "goodness" of a graph layout
  - **Crossings:** minimize towards planar
  - **Total Edge Length:** minimize towards proper scale
  - **Area:** minimize towards efficiency
  - **Maximum Edge Length:** minimize longest edge
  - **Uniform Edge Lengths:** minimize variances
  - **Total Bends:** minimize orthogonal towards straight line

Haleem et al., 2019

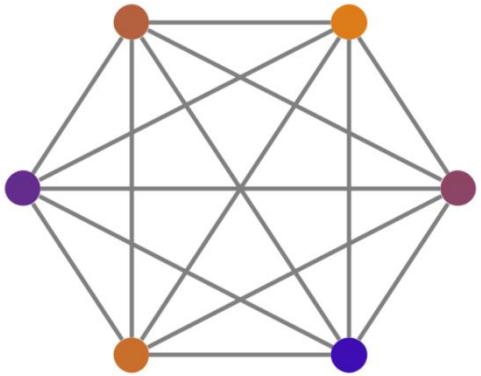
Graph simplification

- Extracting network motifs

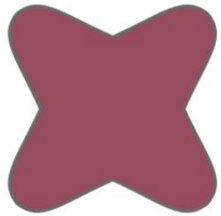
Connector



Clique



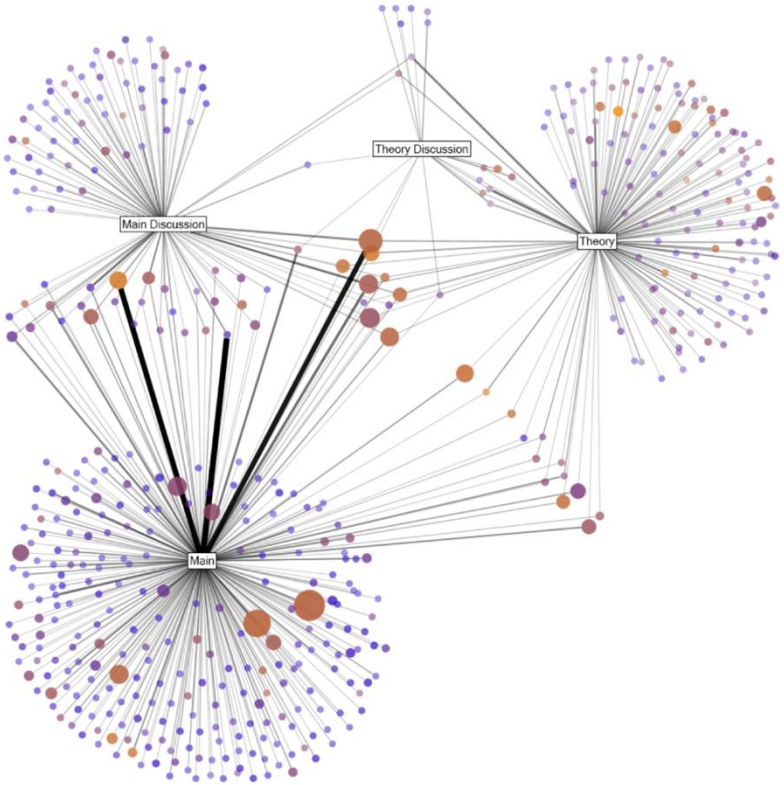
Fan

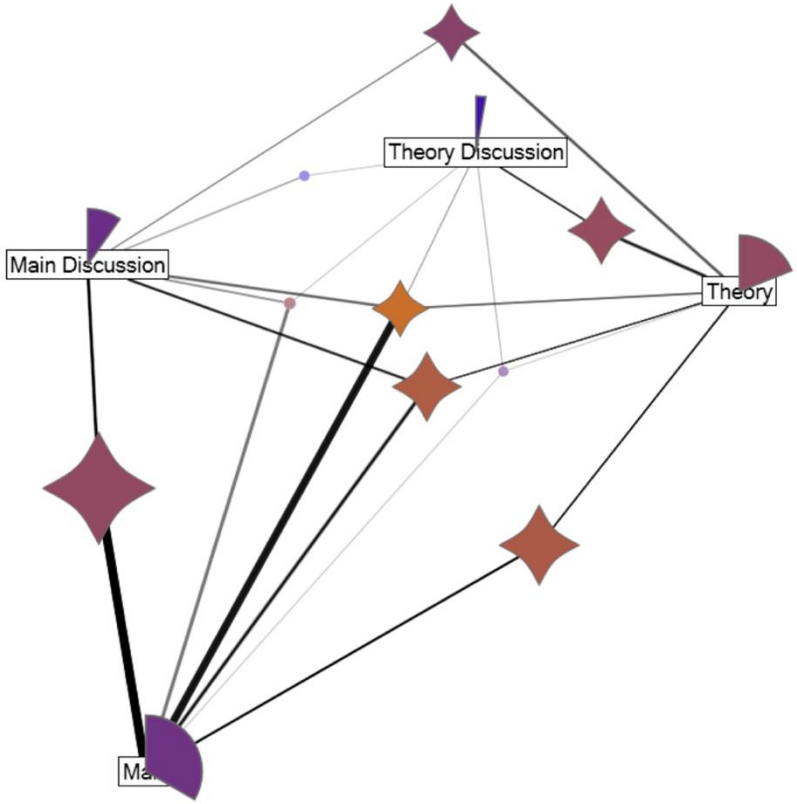


Dunne and Shneiderman, 2013

Graph simplification

- Extracting network motifs

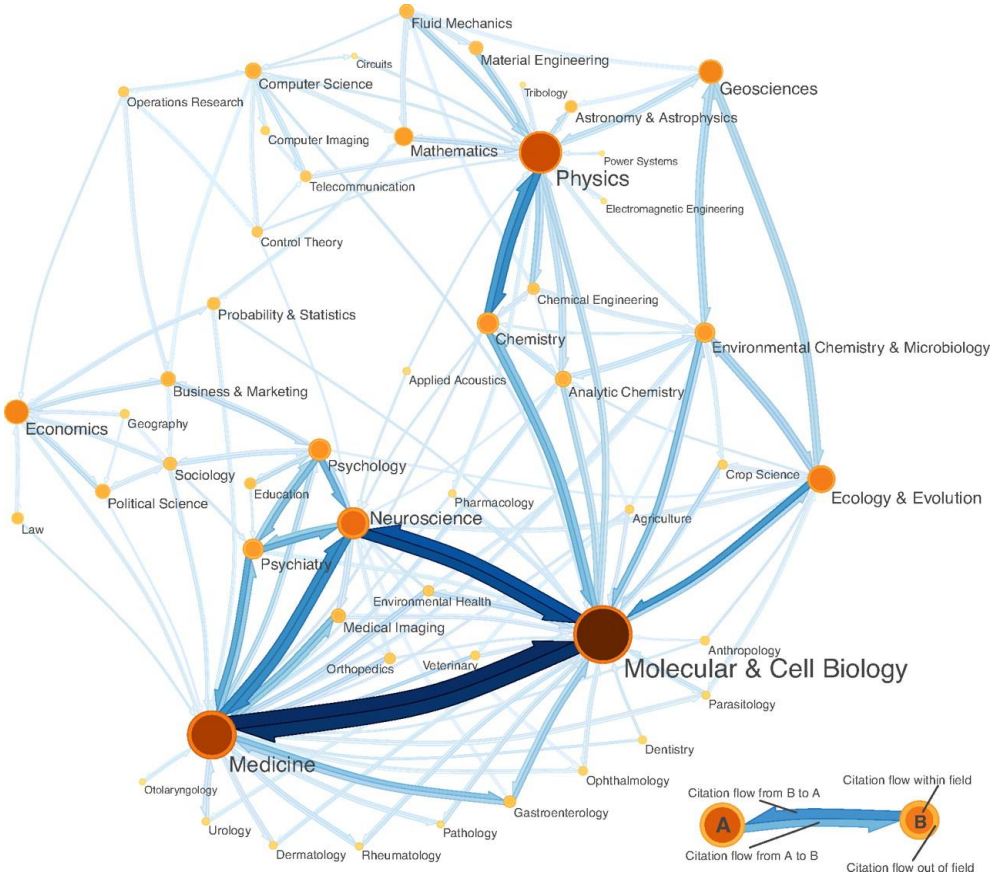




Dunne and Shneiderman, 2013

Graph simplification

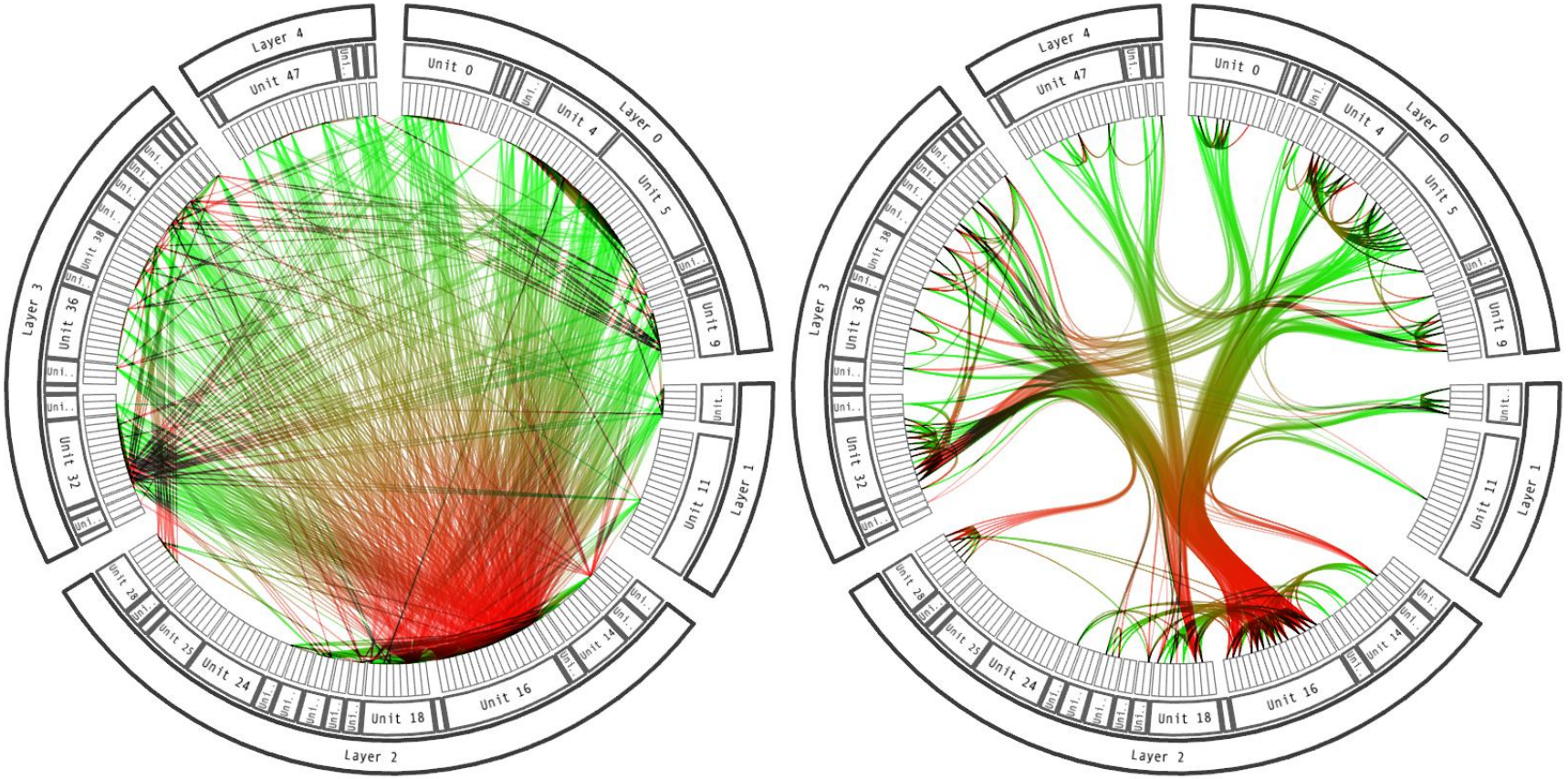
- Clustering and visualization



Rosvall and Bergstrom, 2018

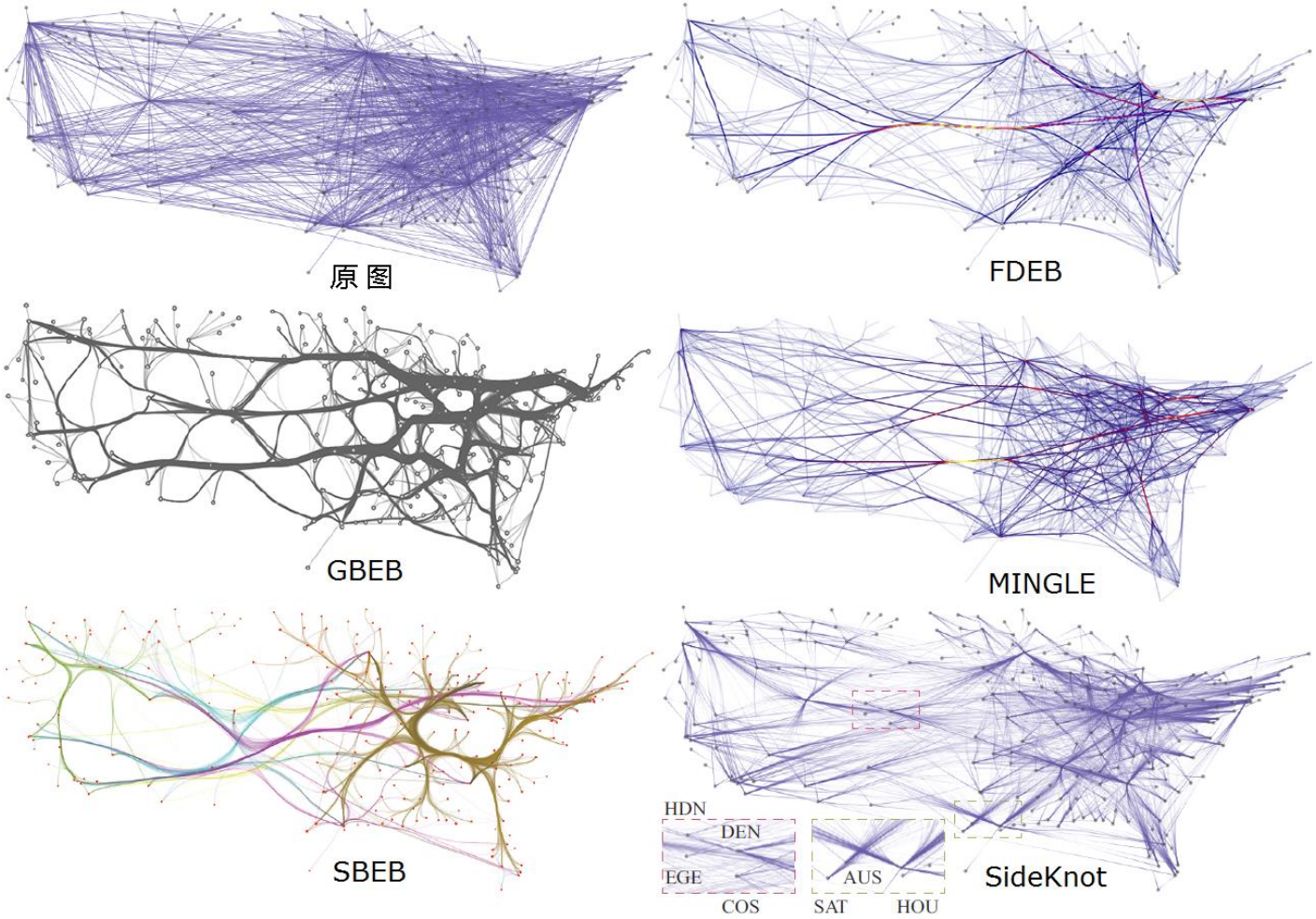
Graph simplification

- Edge bundling



Graph simplification

- Edge bundling



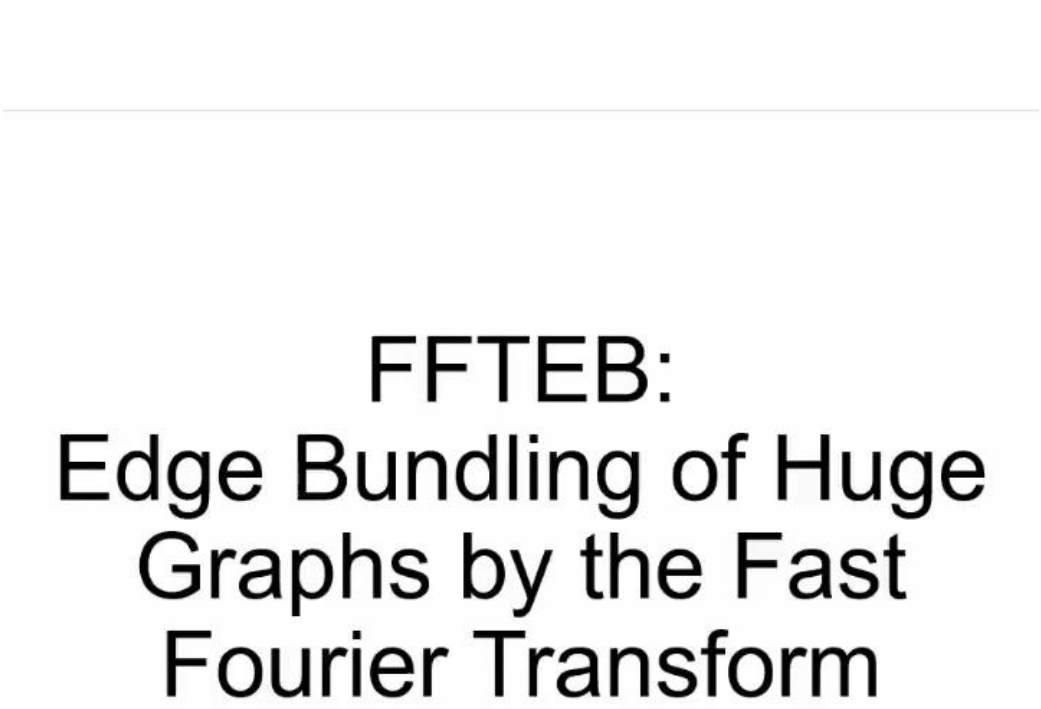
Graph simplification

- Edge bundling

Hurter et al., 2012

Graph simplification

- Edge bundling



Lhuilier et al., 2017

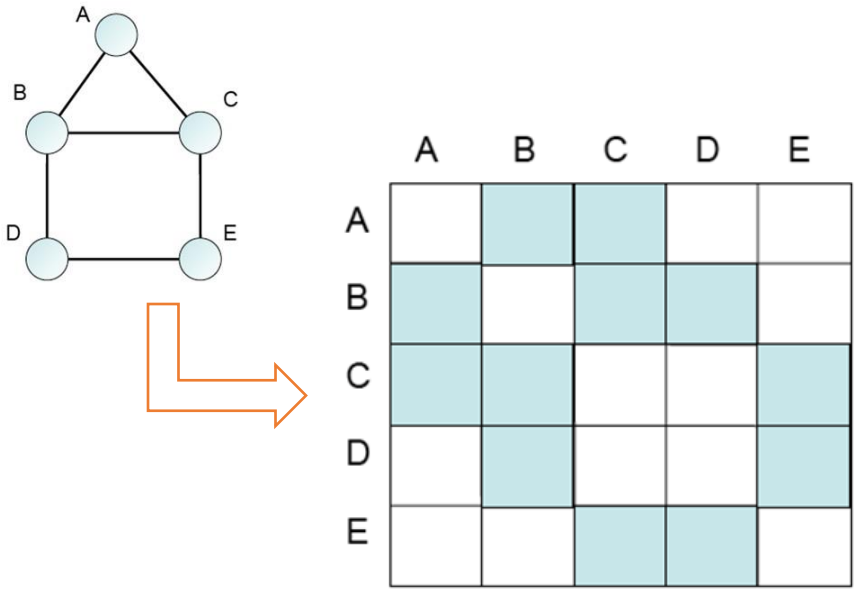
Data Exploration & Visualization

Module 12: Graph Visualization

- Graph data
  - representation, terminology, challenges, tasks
- Graph visualization
  - Node-link graph: layout, graph simplification
  - Matrix visualization
  - Hybrid visualization
  - Graph visualization tools

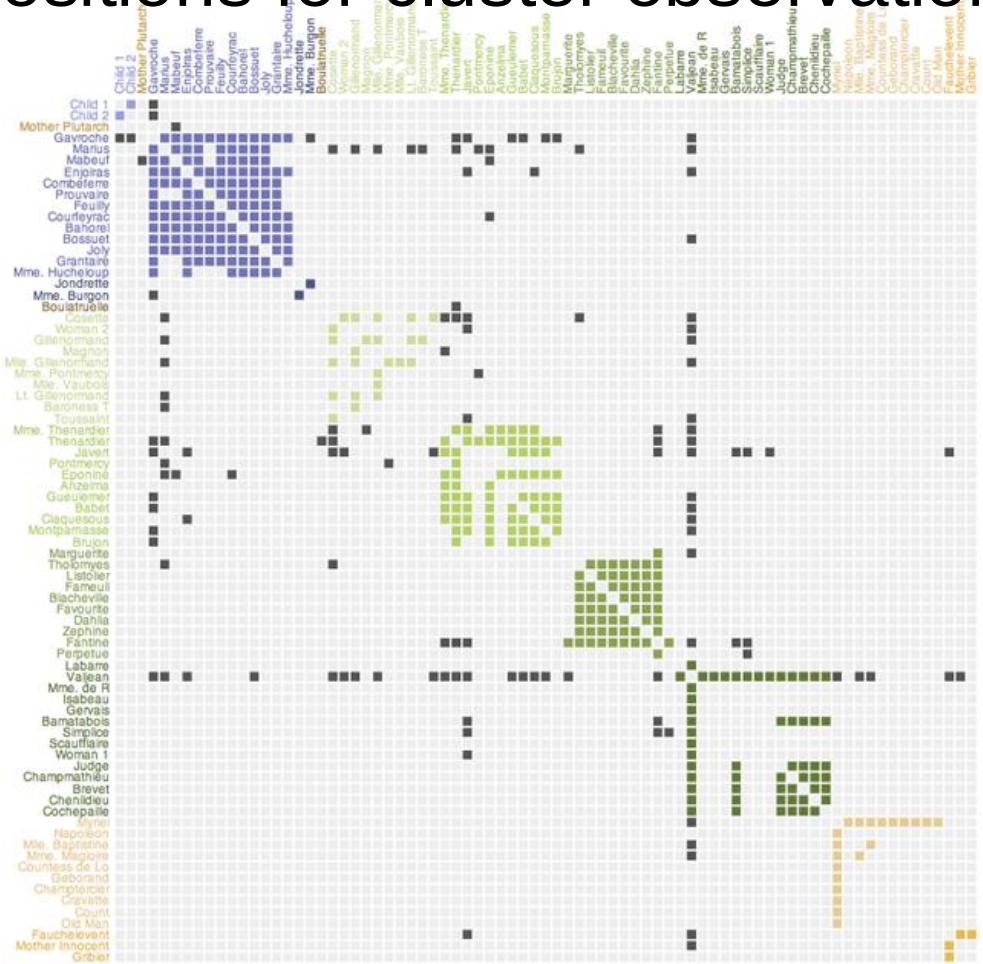
Adjacency matrix

- N × Nmatrix, representing relations among N nodes.
- Cell (i, j) represents the relation between the *i*th and the *j*th node
  - Weight
  - Direction
- Related issues
  - Ordering
  - Path finding



Adjacency matrix

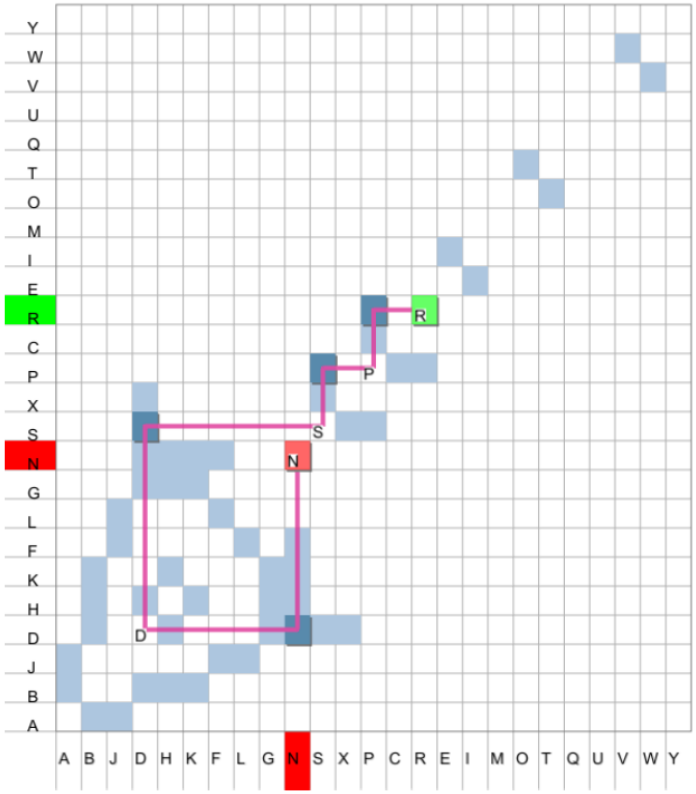
- Order positions for cluster observation

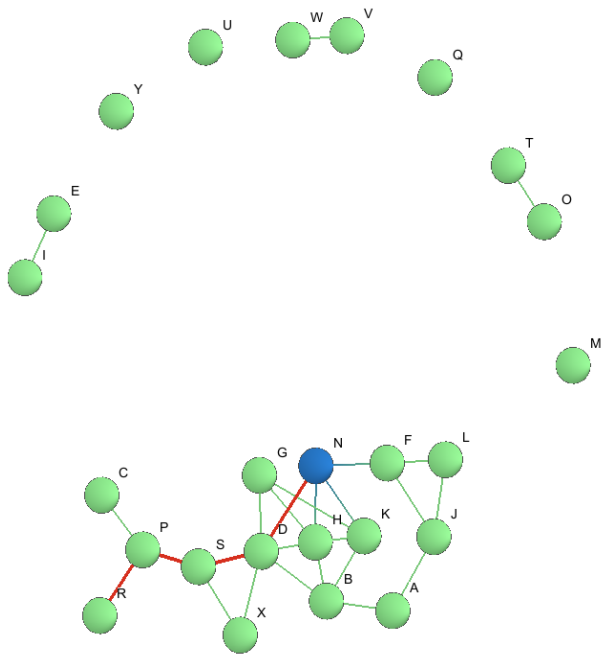


http://hci.stanford.edu/jheer/files/zoo

Adjacency matrix

- Path finding is difficult





Left: Connecting matrix entries and the diagonal.
Right: Path {N;D;S;P;R} is Highlighted in the Node-link Diagram.

Adjacency matrix

- Advantages:
  - No edge crossing, good for edge cluttered graph
  - Good visual scalability
  - Adding new items makes small changes in the visual representation

- Disadvantages:
  - Visualization is too abstract to understand
  - Difficult to follow a transitive relation path

Hybrid visualization

- Complex edge relations—node-link diagram
- Too many nodes—adjacency matrix
- What if there are many nodes, and some of them with complex edge relations?

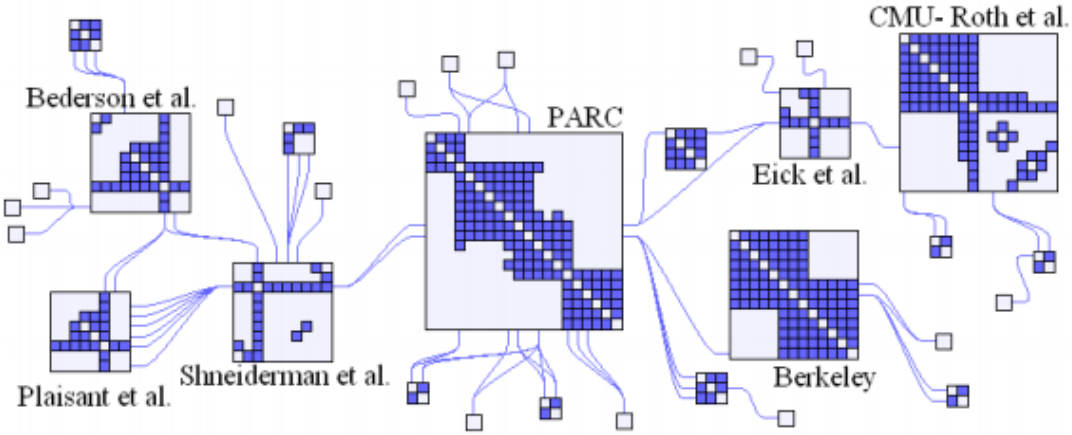


Fig. 1: NodeTrix Representation of the largest component of the Info-Vis Co-authorship Network

Henry et al., 2007

Hybrid visualization

Henry et al., 2007

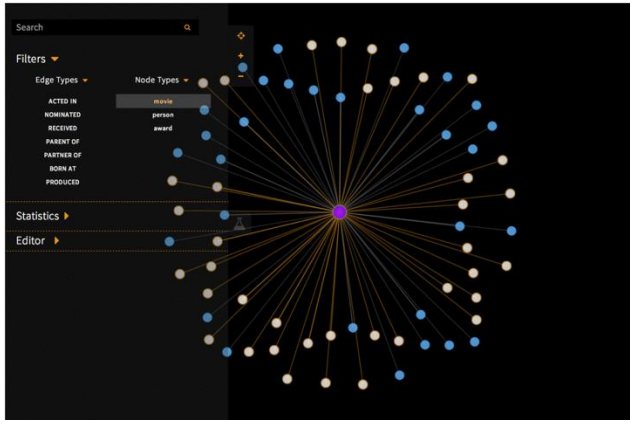
Data Exploration & Visualization

Module 12: Graph Visualization

- Graph data
  - representation, terminology, challenges, tasks
- Graph visualization
  - Node-link graph: layout, graph simplification
  - Matrix visualization
  - Hybrid visualization
  - Graph visualization tools

Graph visualization tools

- Visualization packages



Alchemy.js

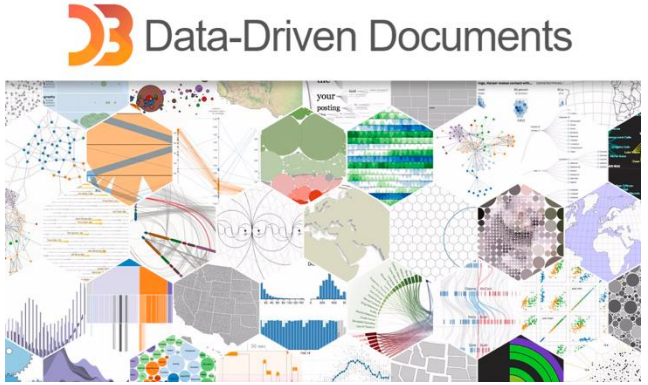
…

Sigma.js

Cytoscape.js

VivaGraphJS

…



D3.js

Pixi.js

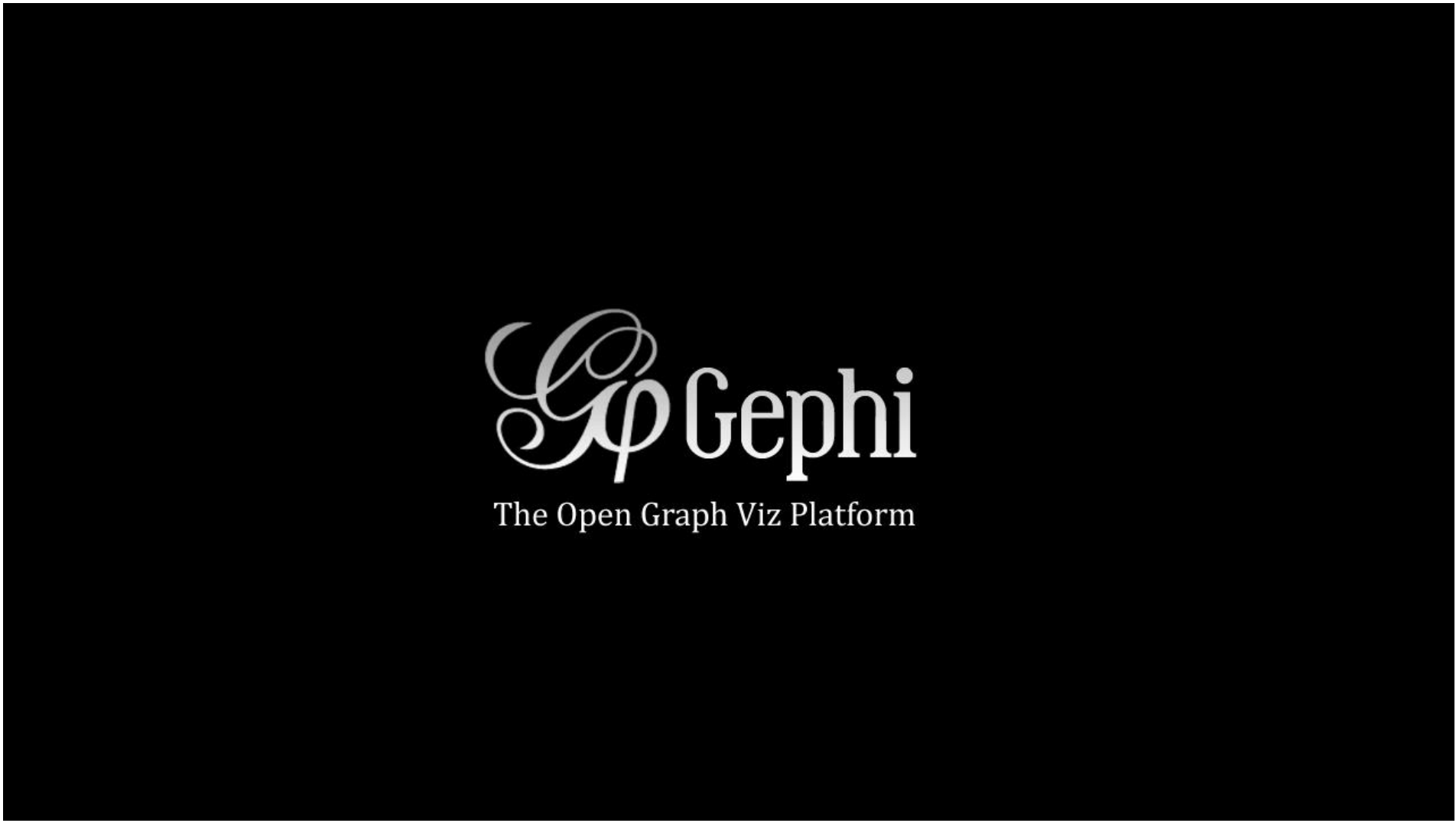
…

More and more flexible

Graph visualization tools

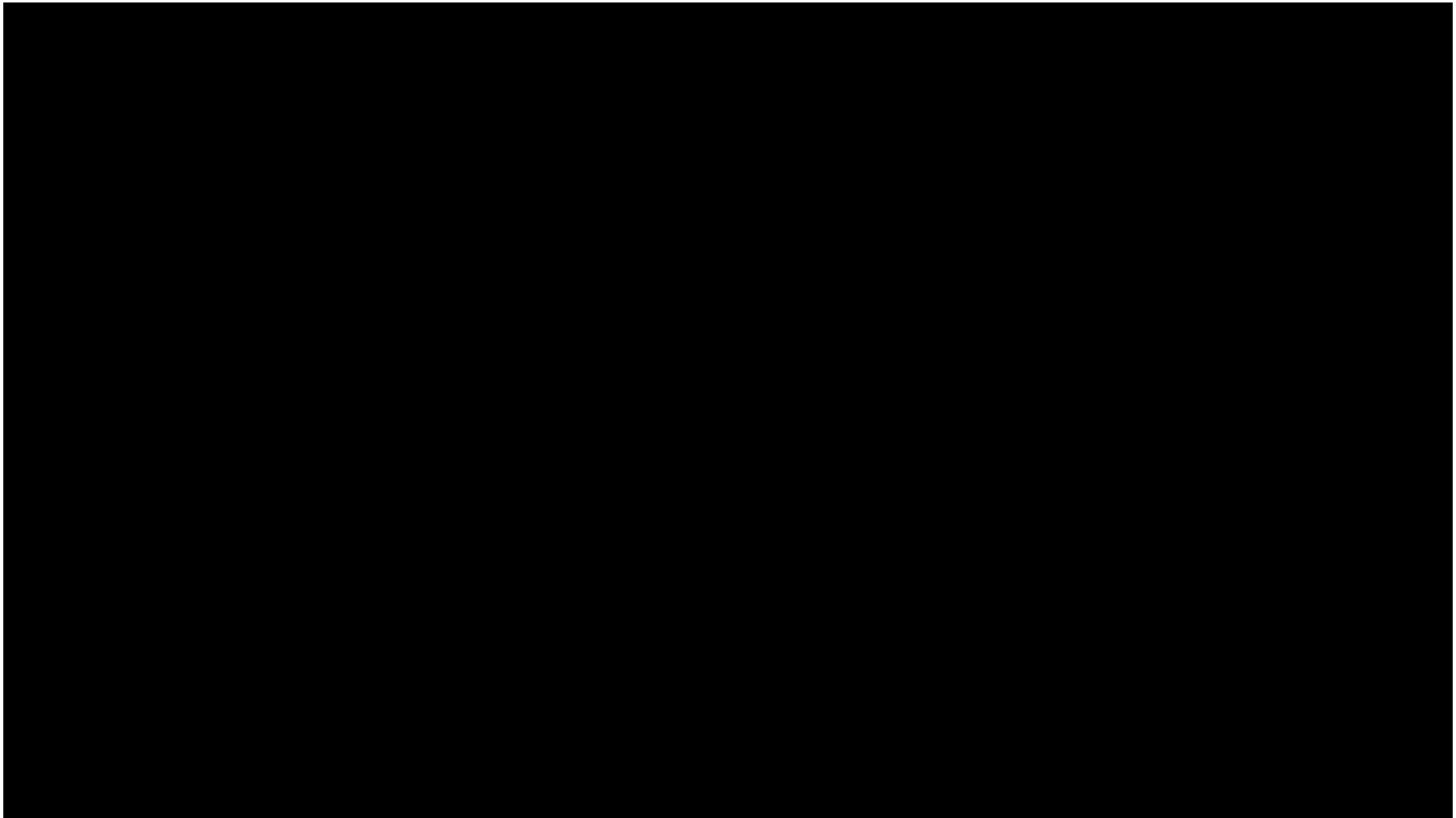
- Software
  - Gephi
  - Cytoscape
  - Palantir
  - Visone
  - …

Graph visualization tools



Examples

- A Network of Science: 150 Years of Nature Papers – Nature, 2019
  - Interactive version: https://www.nature.com/articles/d41586-019-03305-w



Summary

- Node-link diagram
  - Easy to understand topological structure, but problematic for dense graphs
  - Graph simplification (e.g., edge bundling) can help
    - Not always possible, and not always appropriate
- Matrix-based visualization
  - No node occlusion and edge crossing: good visual scalability, suitable for large graphs
  - Abstract, hard to follow paths
- No best solution
  - graph visualization is still under active research!!!
  - Hybrid methods combine advantages of node-link diagrams and matrix