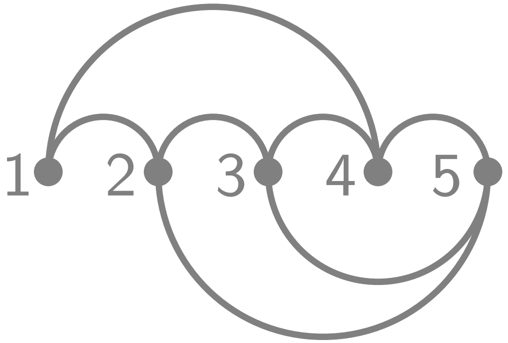
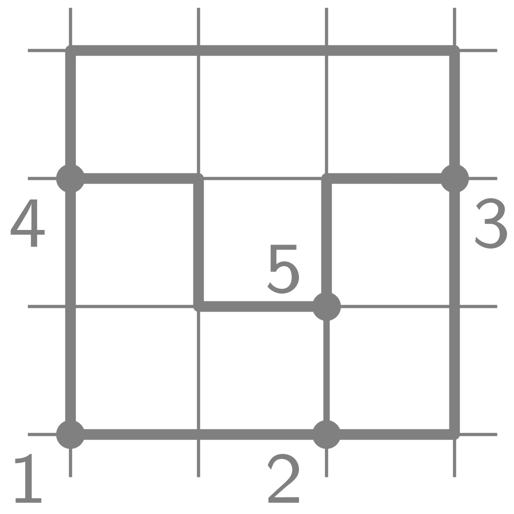
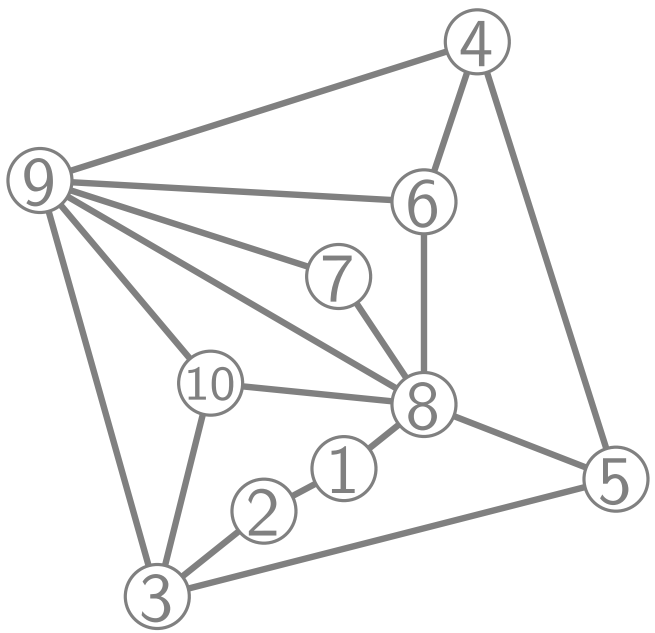
Visualization of Graphs

Lecture 1a: The Graph Visualization Problem





Johannes Zink



Summer semester 2024

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Tutorials:

- Oksana Firman (M4, room 01.005, oksana.firman@uni-wuerzburg.de)

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Tutorials:

- Oksana Firman (M4, room 01.005, oksana.firman@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 24)
- one exercise sheet each week (Friday to Friday; first sheet appears today)

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Tutorials:

- Oksana Firman (M4, room 01.005, oksana.firman@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 24)
- one exercise sheet each week (Friday to Friday; first sheet appears today)
- 20 points per sheet
- average score 50% or more $\Rightarrow$ bonus of 0.3 grade points

Organizational

Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Tutorials:

- Oksana Firman (M4, room 01.005, oksana.firman@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 24)
- one exercise sheet each week (Friday to Friday; first sheet appears today)
- 20 points per sheet
- average score 50% or more $\Rightarrow$ bonus of 0.3 grade points
- submit solutions online (WueCampus)
- we recommend using LaTeX – template on WueCampus!

Organizational

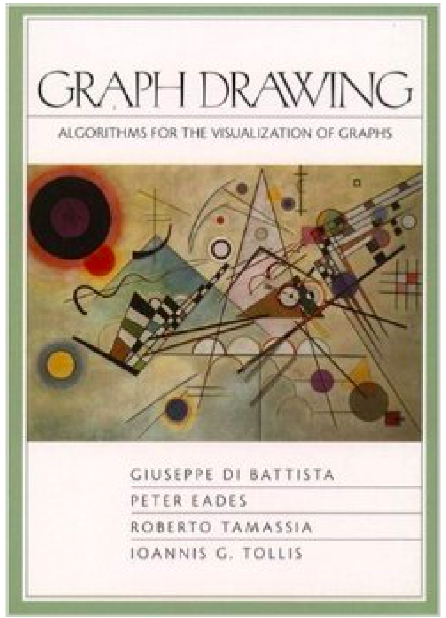
Lectures:

- Johannes Zink (M4, room 01.007, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

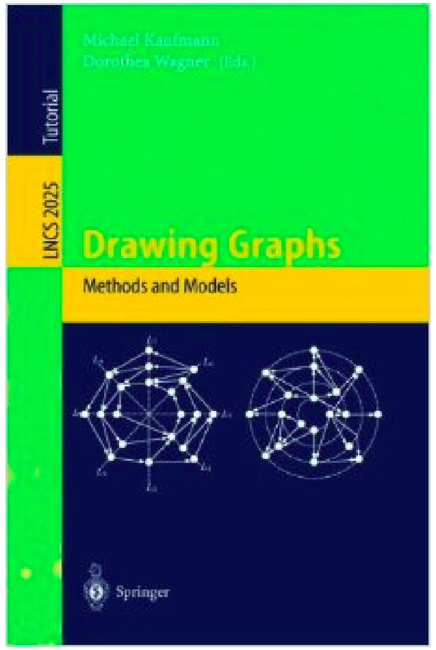
Tutorials:

- Oksana Firman (M4, room 01.005, oksana.firman@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 24)
- one exercise sheet each week (Friday to Friday; first sheet appears today)
- 20 points per sheet
- average score 50% or more $\Rightarrow$ bonus of 0.3 grade points
- submit solutions online (WueCampus)
- we recommend using LaTeX – template on WueCampus!
- discussions and solutions...

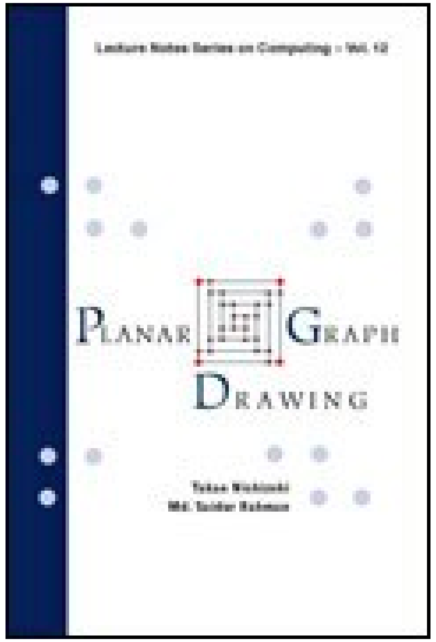
Books



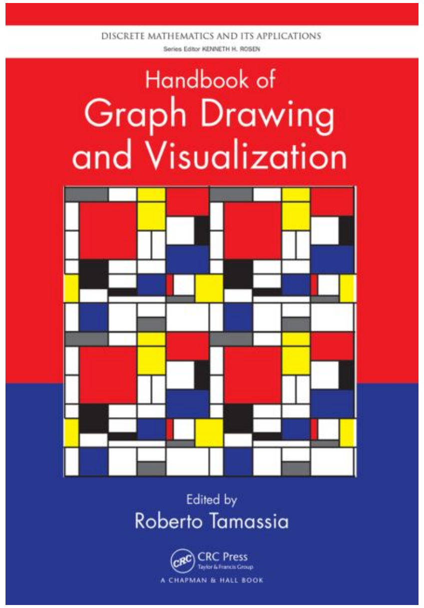
G. Di Battista, P. Eades, R. Tamassia, I. Tollis:
Graph Drawing: Algorithms for the Visualization of Graphs
Prentice Hall, 1998



M. Kaufmann, D. Wagner:
Drawing Graphs: Methods and Models
Springer, 2001

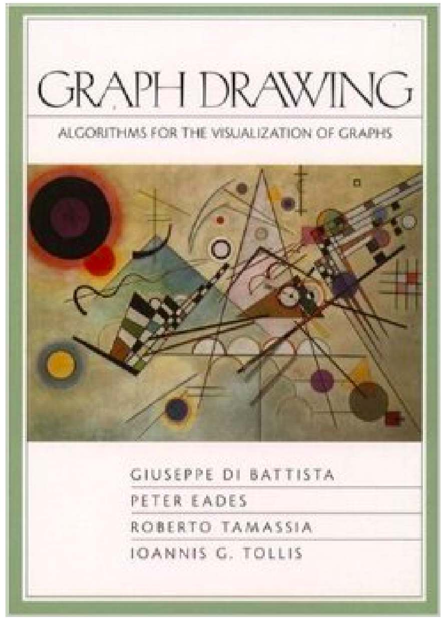


T. Nishizeki, Md. S. Rahman:
Planar Graph Drawing
World Scientific, 2004



R. Tamassia:
Handbook of Graph Drawing and Visualization
CRC Press, 2013
`http://cs.brown.edu/people/rtamassi/gdhandbook/`

Books

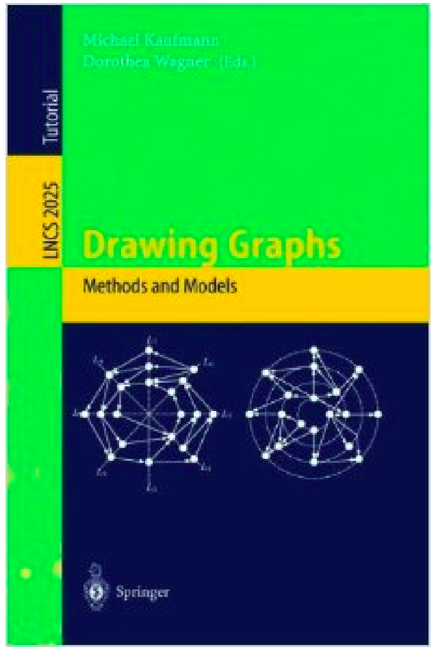


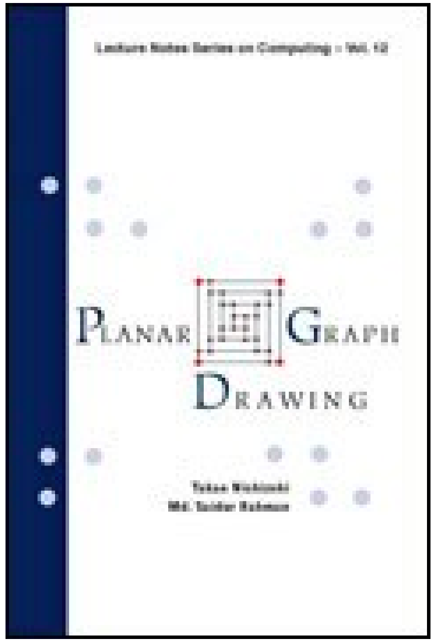
G. Di Battista, P. Eades, R. Tamassia, I. Tollis:
Graph Drawing: Algorithms for the Visualization of Graphs
Prentice Hall, 1998

[GD]

[DG]

M. Kaufmann, D. Wagner:
Drawing Graphs: Methods and Models
Springer, 2001



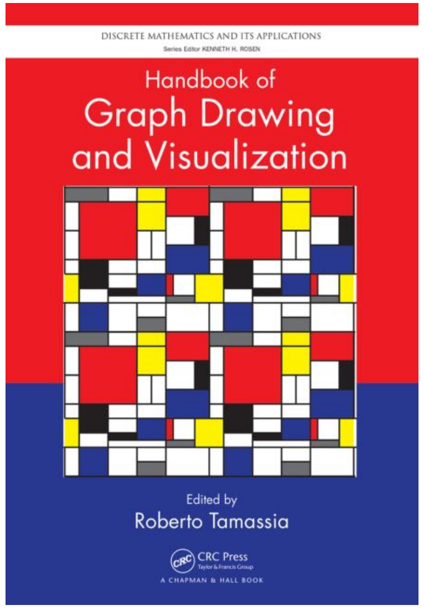


T. Nishizeki, Md. S. Rahman:
Planar Graph Drawing
World Scientific, 2004

[PGD]

[HGDV]

R. Tamassia:
Handbook of Graph Drawing and Visualization
CRC Press, 2013
`http://cs.brown.edu/people/rtamassi/gdhandbook/`



What Is This Course About?

Learning objectives

What Is This Course About?

Learning objectives

- Overview of graph visualization

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:

- Given a graph G, visualize it with a drawing Γ

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:

- Given a graph G, visualize it with a drawing Γ

Here:

- Reducing the visualization problem to its **algorithmic core**

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:

- Given a graph G, visualize it with a drawing Γ

Here:

- Reducing the visualization problem to its **algorithmic core**

graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:

- Given a graph G, visualize it with a drawing Γ

Here:

- Reducing the visualization problem to its **algorithmic core**

graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

- modeling
- data structures

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:

- Given a graph G, visualize it with a drawing Γ

Here:

- Reducing the visualization problem to its **algorithmic core**

graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

- modeling
- data structures

- divide & conquer, incremental
- combinatorial optimization (flows, ILPs)
- force-based algorithm

What Is This Course About?

Learning objectives

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:

- Given a graph G, visualize it with a drawing Γ

Here:

- Reducing the visualization problem to its **algorithmic core**

graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

- modeling
- data structures
- divide & conquer, incremental
- combinatorial optimization (flows, ILPs)
- force-based algorithm
- proofs

What Is This Course About?

Topics

- Drawing Trees and Series-Parallel Graphs
- Force-Based Drawing Algorithms and Tutte Embedding
- Straight-Line Drawings of Planar Graphs
- Upwards Planar Drawings
- Orthogonal Grid Drawings
- Contact Representations
- Hierarchical Layouts of Directed Graphs
- Visibility Representations
- The Crossing Lemma
- Linear Layouts
- Beyond Planarity
- Octilinear Drawings for Metro Maps

Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$,
  where each edge is a pair of vertices.

Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$, where each edge is a pair of vertices.

Representation?

Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$, where each edge is a pair of vertices.

Representation?

- Set notation

$$V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$$

$$\begin{aligned} E(G) = \{&\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \\ &\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \\ &\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \\ &\{v_9, v_{10}\}\} \end{aligned}$$

Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$,
  where each edge is a pair of vertices.

Representation?

- Set notation

$$V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$$

$$\begin{aligned} E(G) = \{&\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \\ &\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \\ &\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \\ &\{v_9, v_{10}\}\} \end{aligned}$$

- Adjacency list

v_1:	v_2, v_8	v_6:	v_4, v_8, v_9
v_2:	v_1, v_3	v_7:	v_8, v_9
v_3:	v_2, v_5, v_9, v_{10}	v_8:	$v_1, v_5, v_6, v_7, v_9, v_{10}$
v_4:	v_5, v_6, v_9	v_9:	$v_3, v_4, v_6, v_7, v_8, v_{10}$
v_5:	v_3, v_4, v_8	v_{10}:	v_3, v_8, v_9

Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$, where each edge is a pair of vertices.

Representation?

- Set notation

$$V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$$

$$\begin{aligned} E(G) = \{&\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \\ &\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \\ &\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \\ &\{v_9, v_{10}\}\} \end{aligned}$$

- Adjacency list

v_1 : v_2, v_8
v_2 : v_1, v_3
v_3 : v_2, v_5, v_9, v_{10}
v_4 : v_5, v_6, v_9
v_5 : v_3, v_4, v_8
v_6 : v_4, v_8, v_9
v_7 : v_8, v_9
v_8 : $v_1, v_5, v_6, v_7, v_9, v_{10}$
v_9 : $v_3, v_4, v_6, v_7, v_8, v_{10}$
v_{10} : v_3, v_8, v_9

- Adjacency matrix

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$, where each edge is a pair of vertices.

Representation?

- Set notation

$$V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$$

$$E(G) = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \{v_9, v_{10}\}\}$$

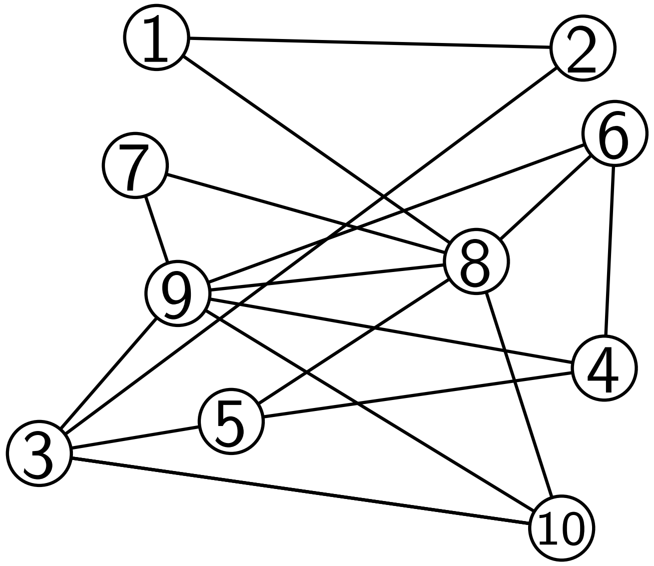
- Adjacency list

v_1:	v_2, v_8	v_6:	v_4, v_8, v_9
v_2:	v_1, v_3	v_7:	v_8, v_9
v_3:	v_2, v_5, v_9, v_{10}	v_8:	$v_1, v_5, v_6, v_7, v_9, v_{10}$
v_4:	v_5, v_6, v_9	v_9:	$v_3, v_4, v_6, v_7, v_8, v_{10}$
v_5:	v_3, v_4, v_8	v_{10}:	v_3, v_8, v_9

- Adjacency matrix

$$\begin{pmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0
\end{pmatrix}$$

- Drawing



Graphs and Their Representations

What is a graph?

- graph G
- vertex set $V(G) = \{v_1, v_2, \ldots, v_n\}$
- edge set $E(G) = \{e_1, e_2, \ldots, e_m\}$, where each edge is a pair of vertices.

Representation?

- Set notation

$$V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$$

$$\begin{aligned} E(G) = \{&\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \\ &\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \\ &\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \\ &\{v_9, v_{10}\}\} \end{aligned}$$

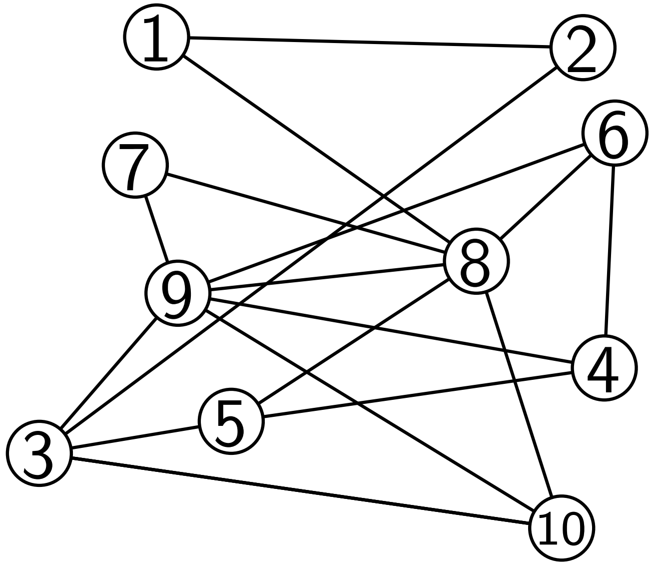
- Adjacency list

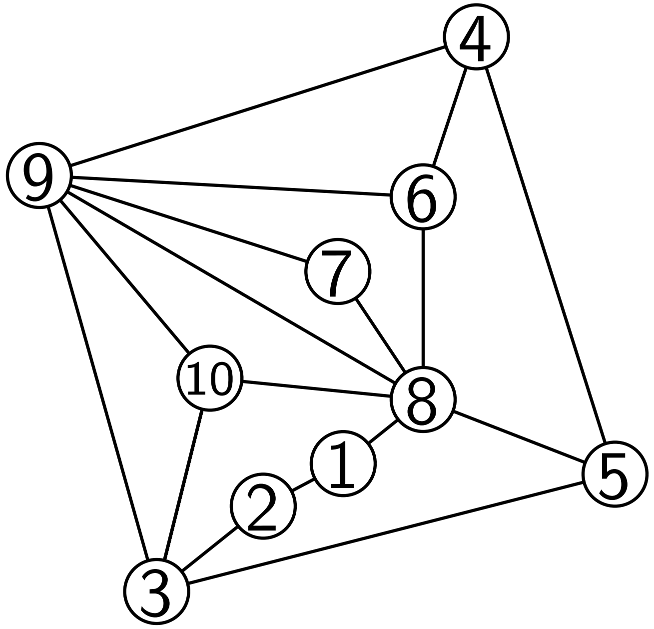
v_1 :	v_2, v_8	v_6 :	v_4, v_8, v_9
v_2 :	v_1, v_3	v_7 :	v_8, v_9
v_3 :	v_2, v_5, v_9, v_{10}	v_8 :	$v_1, v_5, v_6, v_7, v_9, v_{10}$
v_4 :	v_5, v_6, v_9	v_9 :	$v_3, v_4, v_6, v_7, v_8, v_{10}$
v_5 :	v_3, v_4, v_8	v_{10} :	v_3, v_8, v_9

- Adjacency matrix

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

- Drawing





Why to Draw Graphs?

Why to Draw Graphs?

Graphs are a mathematical representation of
real physical and abstract networks.

Why to Draw Graphs?

Graphs are a mathematical representation of real physical and abstract networks.

Physical networks

- Metro systems
- Road networks
- Power grids
- Telecommunication networks
- Integrated circuits
- ...

Why to Draw Graphs?

Graphs are a mathematical representation of
real physical and abstract networks.

Physical networks

- Metro systems
- Road networks
- Power grids
- Telecommunication networks
- Integrated circuits
- ...

Abstract networks

- Social networks
- Communication networks
- Phylogenetic networks
- Metabolic networks
- Class/Object Relation Digraphs (UML)
- ...

Why to Draw Graphs?

Graphs are a mathematical representation of
real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualizations!

Why to Draw Graphs?

Graphs are a mathematical representation of
real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualizations!
- Visualizations help with the **communication** and **exploration** of networks.

Why to Draw Graphs?

Graphs are a mathematical representation of
real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualizations!
- Visualizations help with the **communication** and **exploration** of networks.
- Some graphs are too big to draw them by hand.

Why to Draw Graphs?

Graphs are a mathematical representation of
real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualizations!
- Visualizations help with the **communication** and **exploration** of networks.
- Some graphs are too big to draw them by hand.

We need algorithms that draw graphs automatically to make networks more accessible to humans.

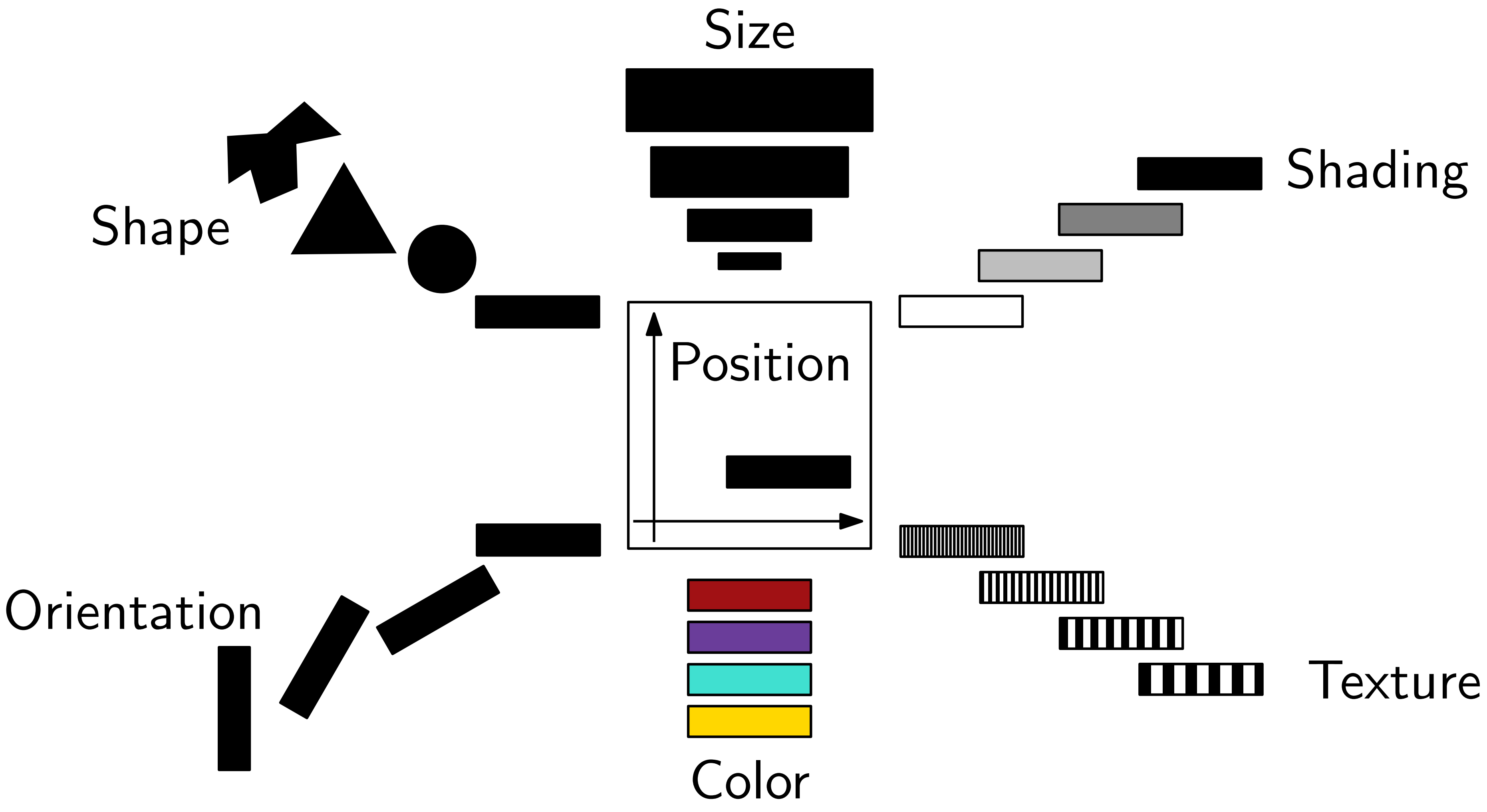
What Are We Interested In?

What Are We Interested In?

- Jacques Bertin defined *visualization variables* (1967)

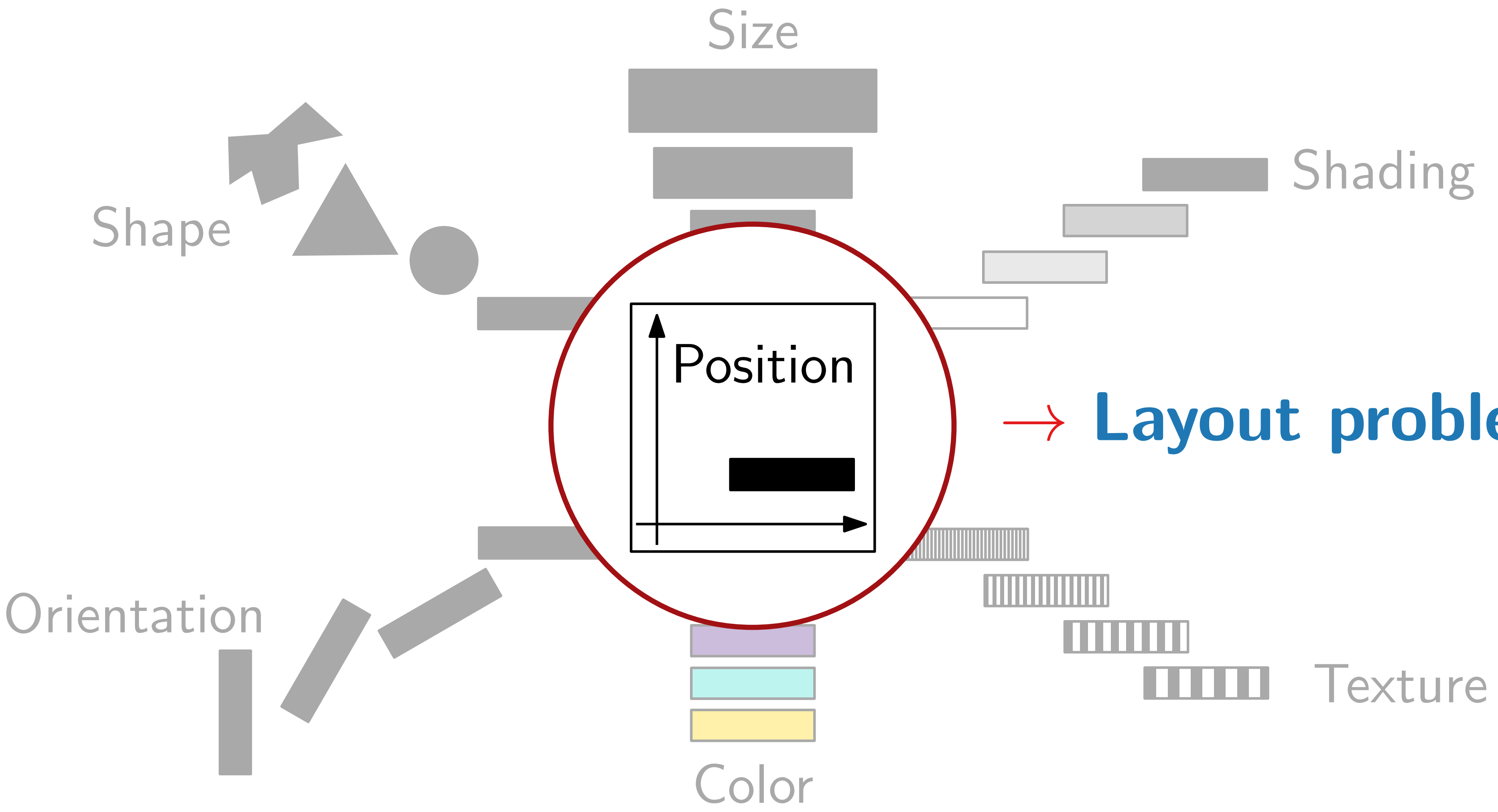
What Are We Interested In?

- Jacques Bertin defined *visualization variables* (1967)



What Are We Interested In?

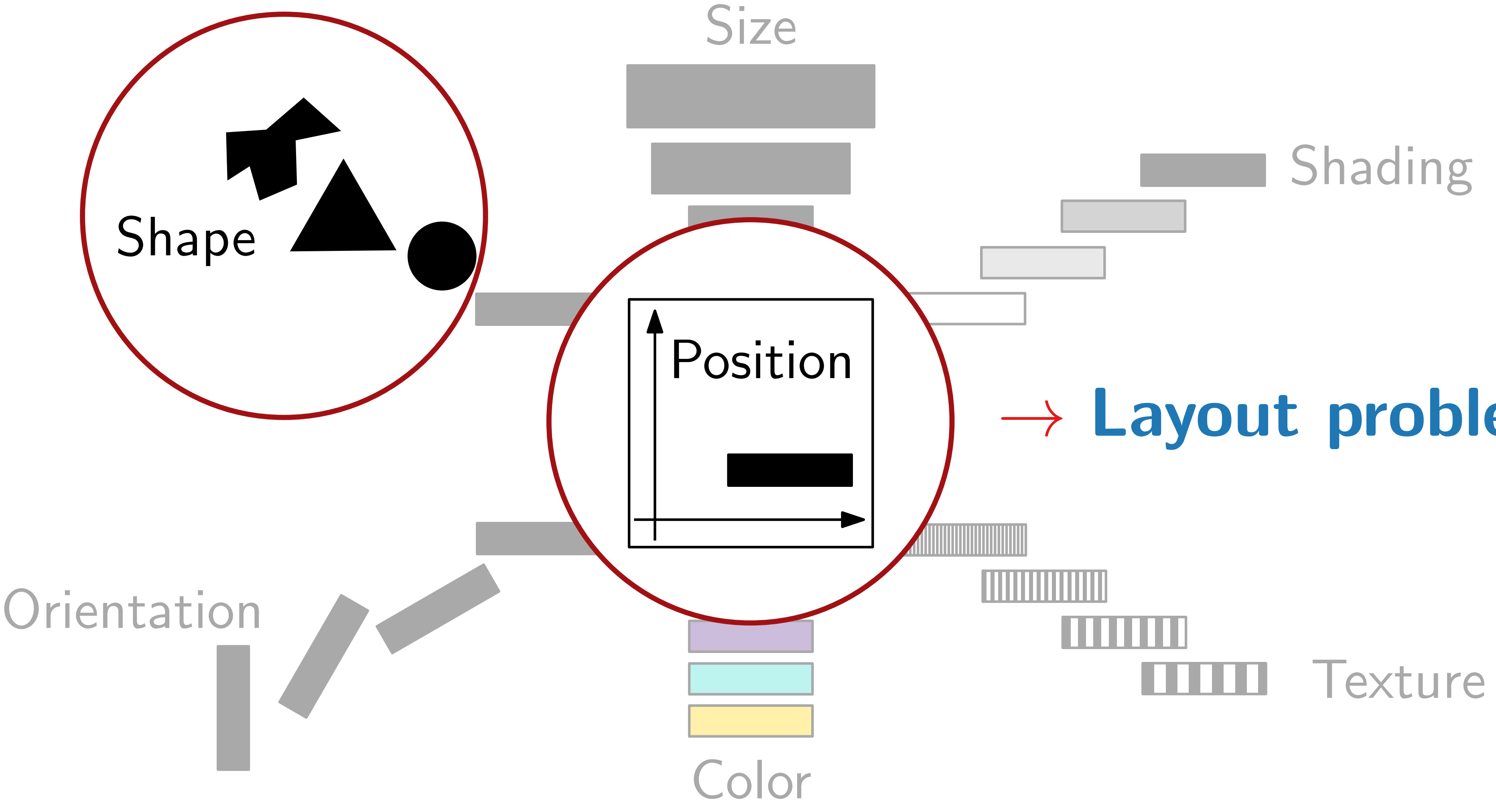
- Jacques Bertin defined *visualization variables* (1967)



→ **Layout problem**

What Are We Interested In?

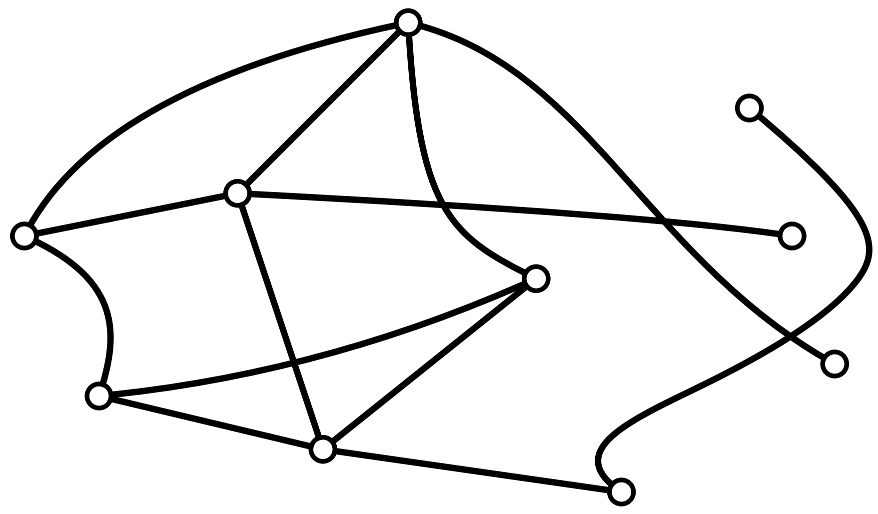
- Jacques Bertin defined *visualization variables* (1967)



→ **Layout problem**

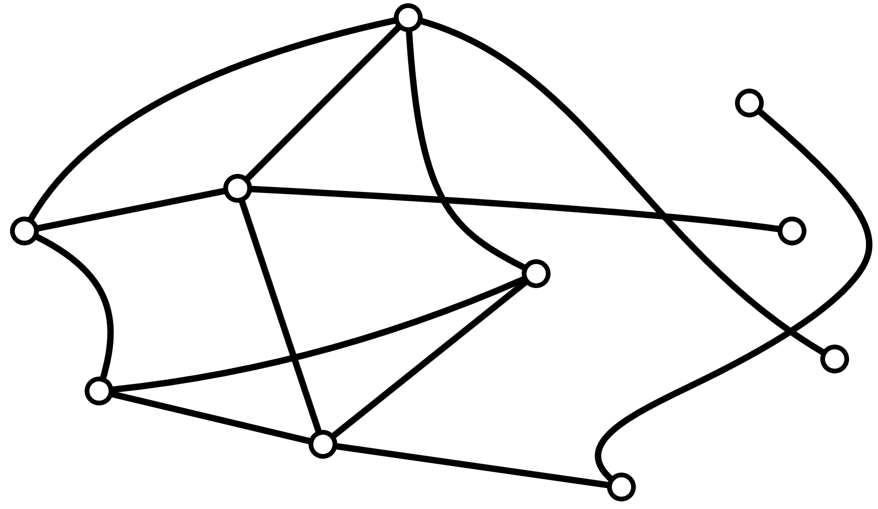
The Layout Problem

- *Here* restricted to the **standard representation**, so-called node–link diagrams.



The Layout Problem

- *Here* restricted to the **standard representation**, so-called node–link diagrams.

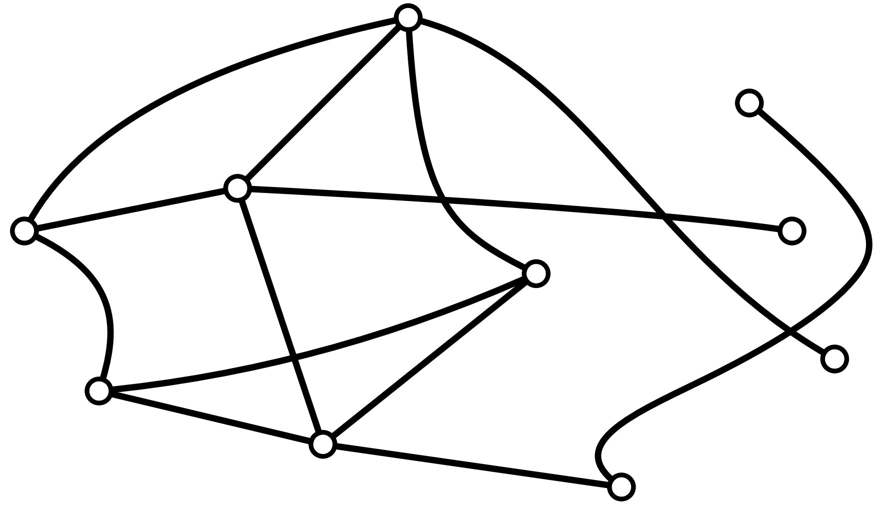


Graph Visualization Problem

in: graph G

out:

The Layout Problem



- *Here* restricted to the **standard representation**, so-called node–link diagrams.

Graph Visualization Problem

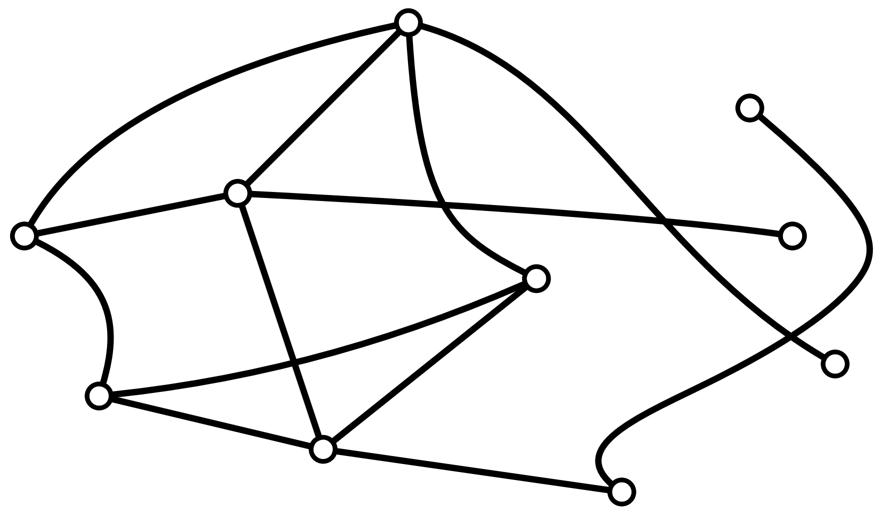
in: graph G

out: nice drawing Γ of G

- $\Gamma \colon V(G) \to \mathbb{R}^2$, vertex $v \mapsto$ point $\Gamma(v)$
- $\Gamma \colon E(G) \to$ simple, open curves in $\mathbb{R}^2$
  $\{u, v\} \mapsto \Gamma(\{u, v\})$ with endpoints $\Gamma(u)$ and $\Gamma(v)$

The Layout Problem?

- *Here* restricted to the **standard representation**, so-called node–link diagrams.



Graph Visualization Problem

in: graph G

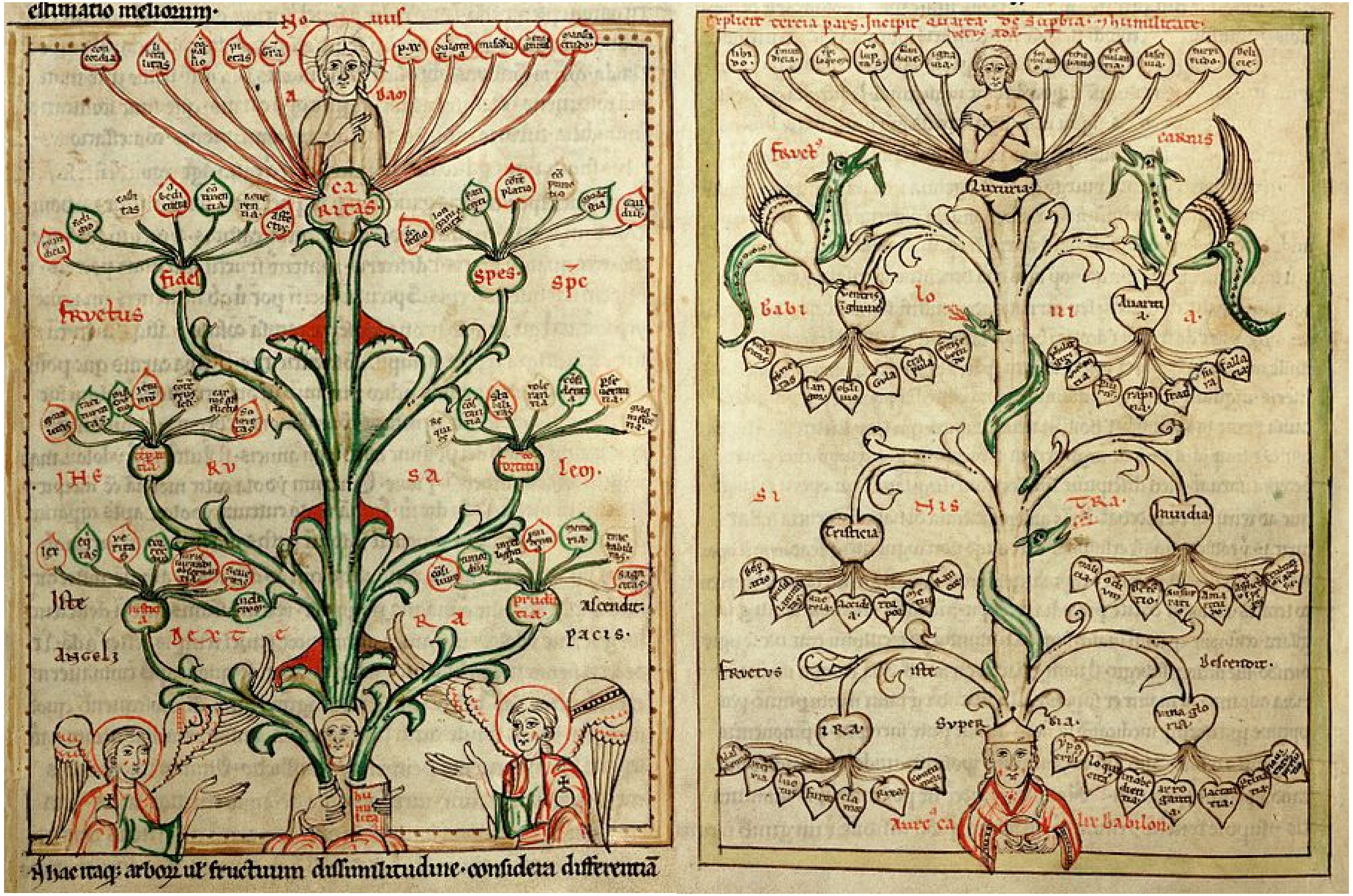
out: nice drawing Γ of G

- $\Gamma \colon V(G) \to \mathbb{R}^2$, vertex $v \mapsto$ point $\Gamma(v)$
- $\Gamma \colon E(G) \to$ simple, open curves in $\mathbb{R}^2$
  $\{u, v\} \mapsto \Gamma(\{u, v\})$ with endpoints $\Gamma(u)$ and $\Gamma(v)$

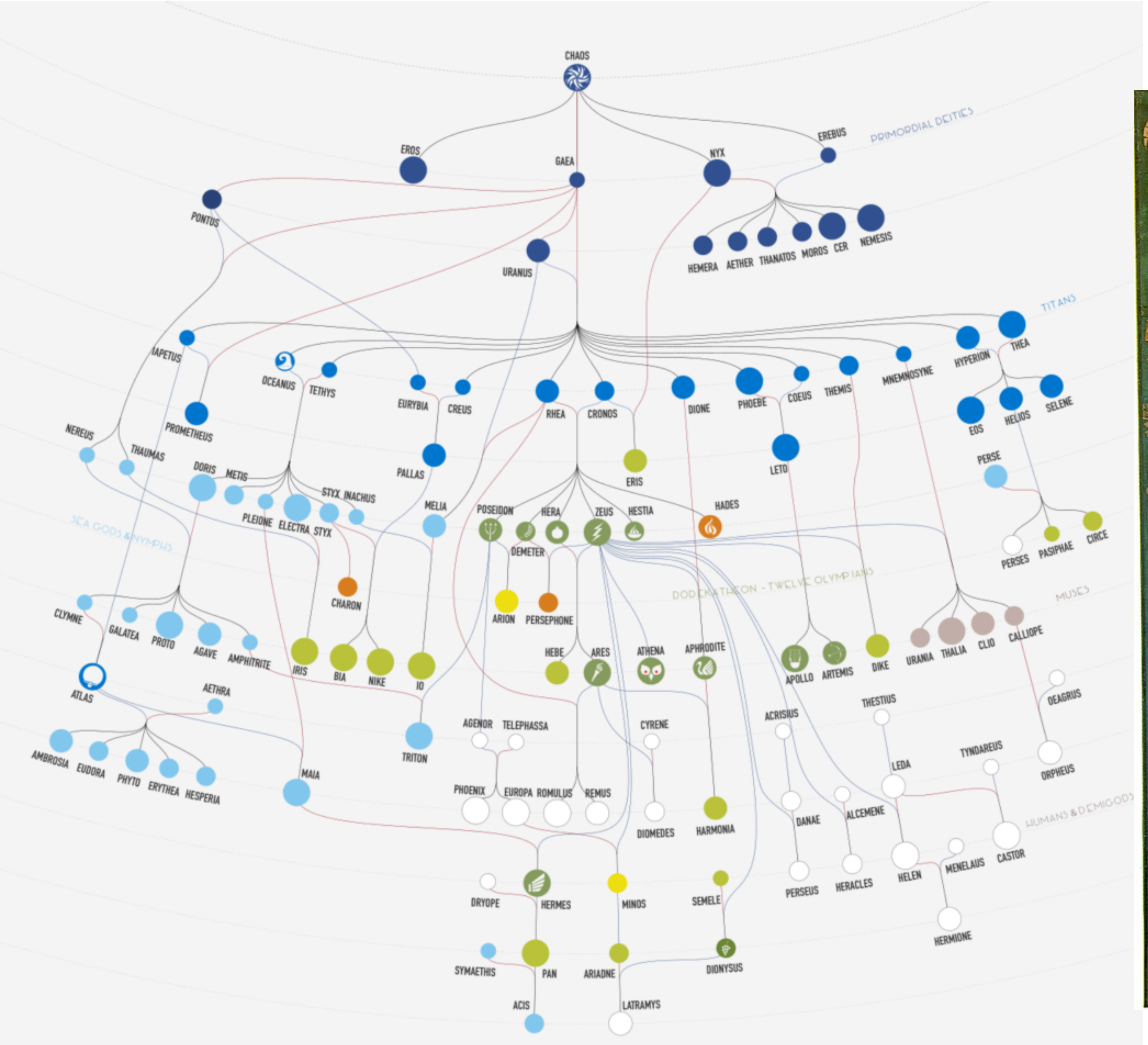
But what is a **nice** drawing?

Tree of Virtues and Tree of Vice

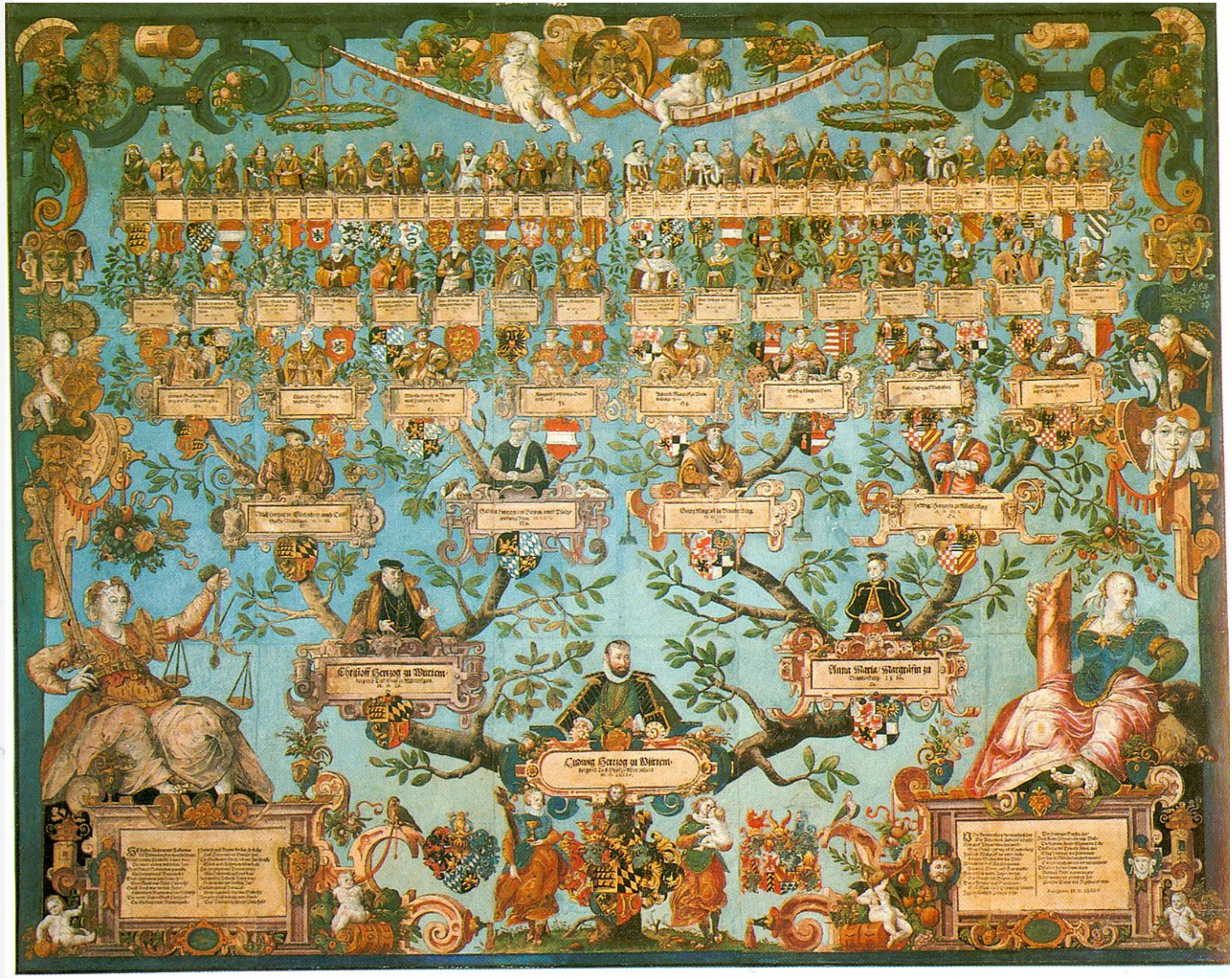
ca. 1200



Social Networks – Family Trees

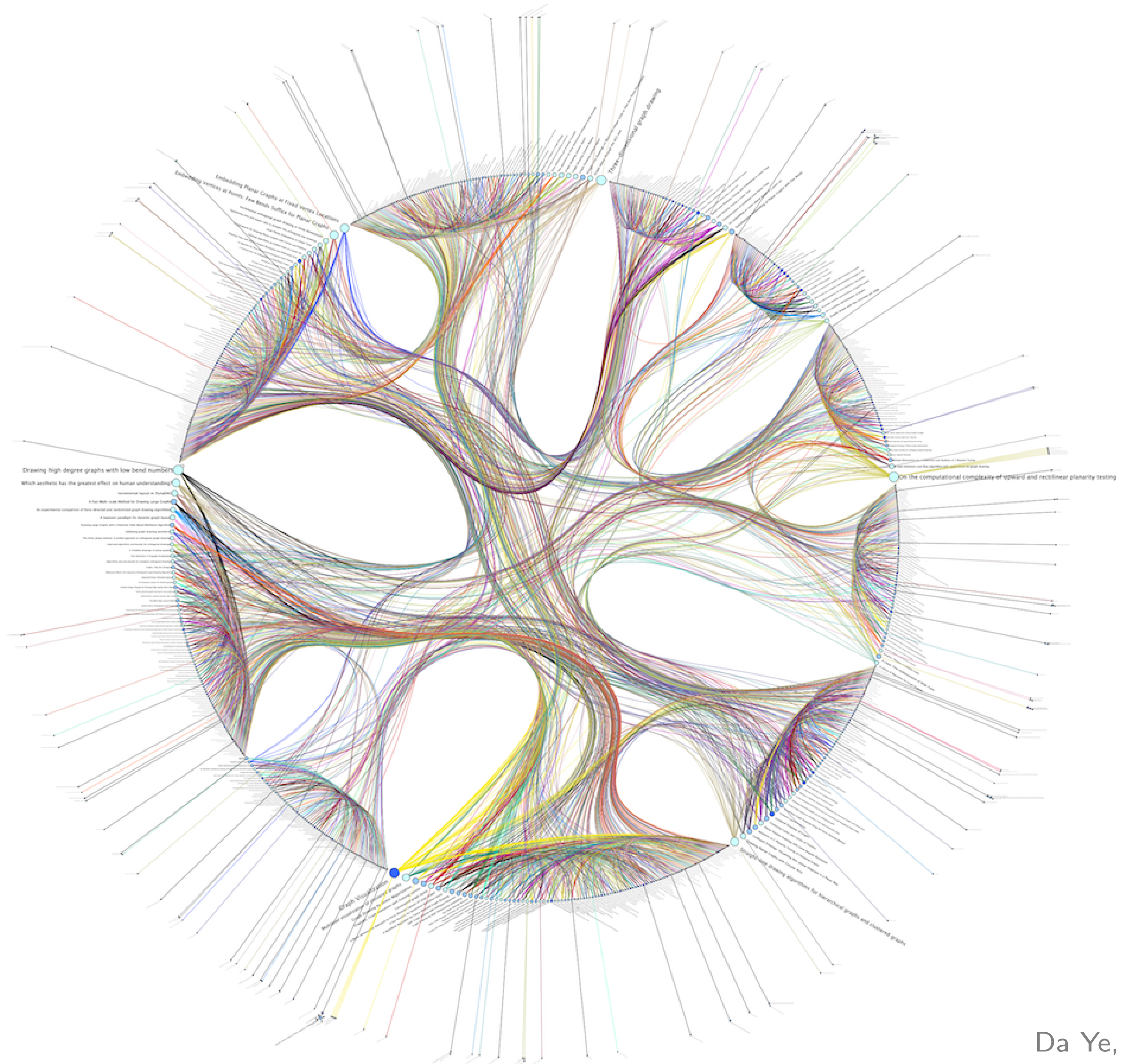


J. Klawitter, T. Mchedlidze, *Link:* go.uniwue.de/myth-poster



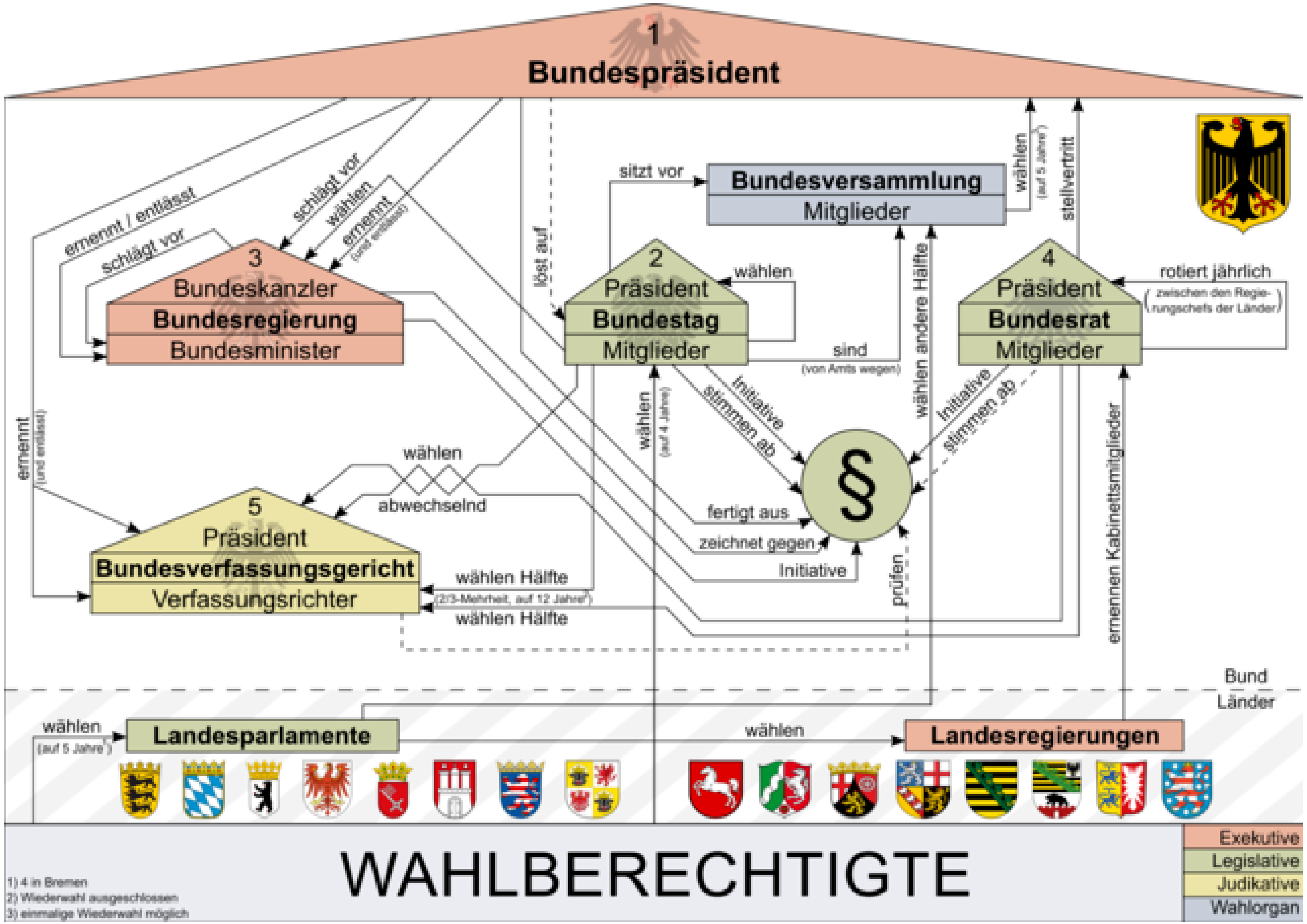
Ahnentafel Herzog Ludwig von Württemberg, 1585

Social Network – Citation Graph

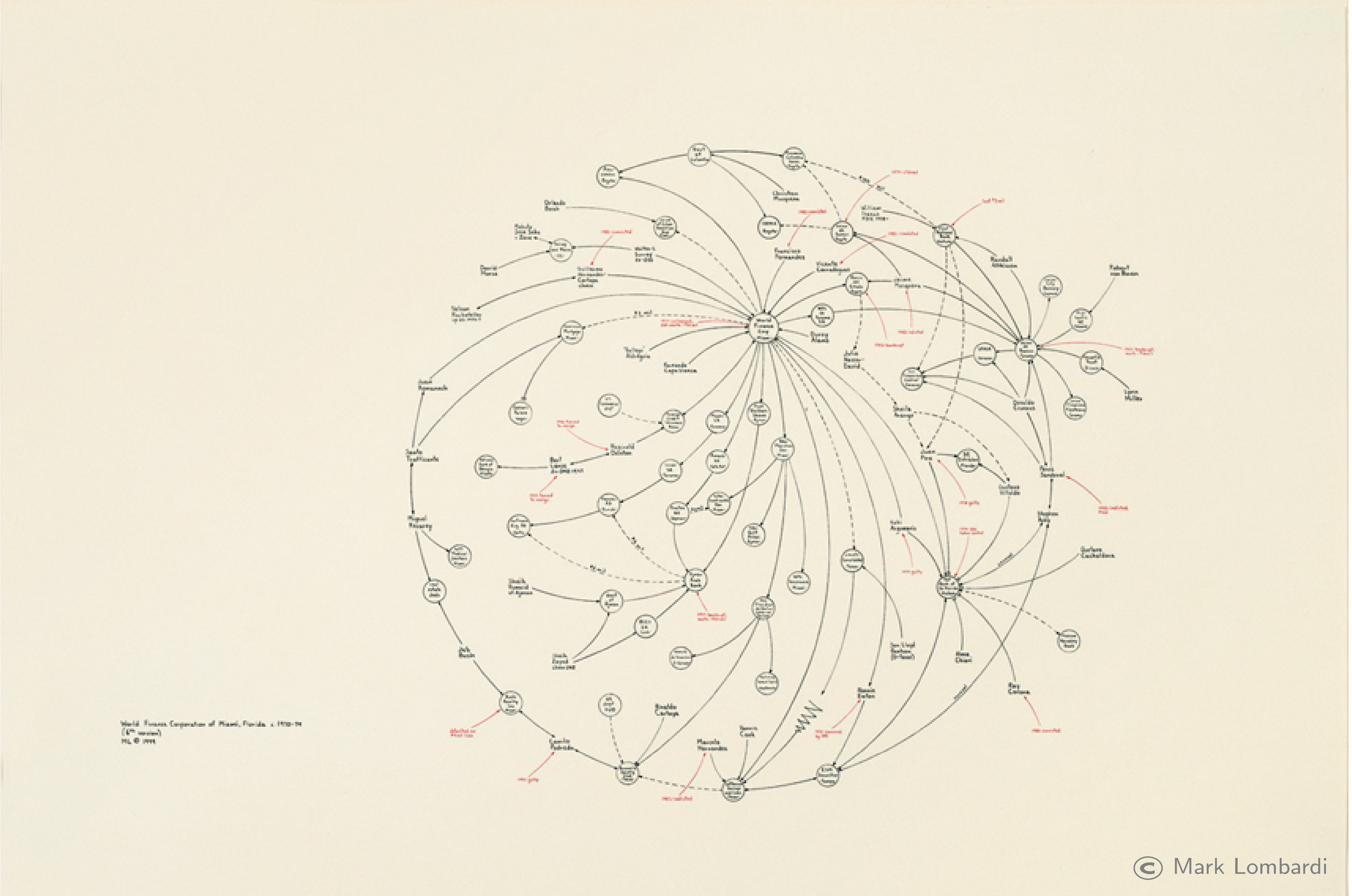


Da Ye, *Link:* https://go.uniwue.de/citation-graph

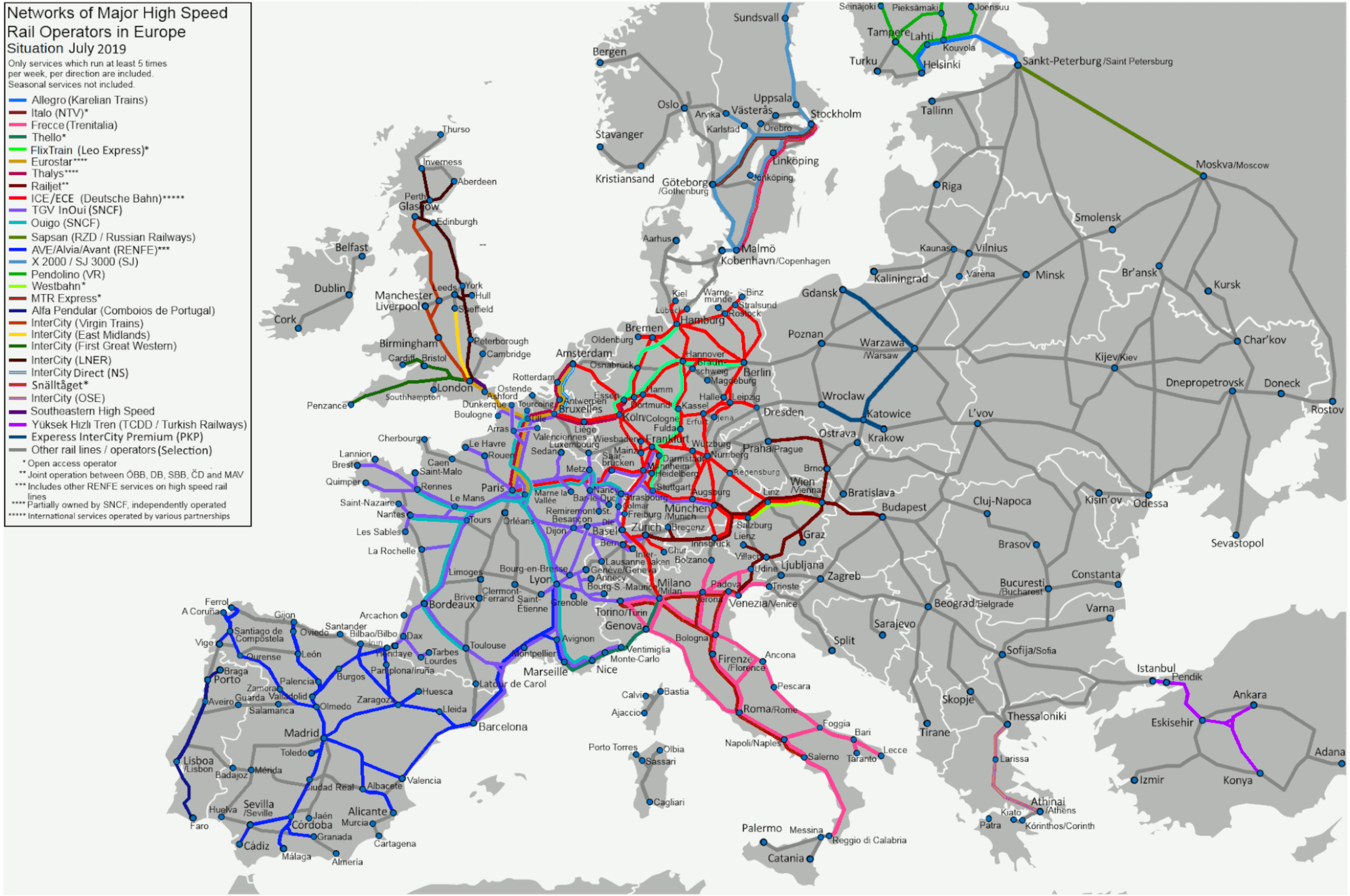
Social Network – Organizational Chart



Social Network – World Finance Corporation



Transportation Network – European High-Speed Railroads

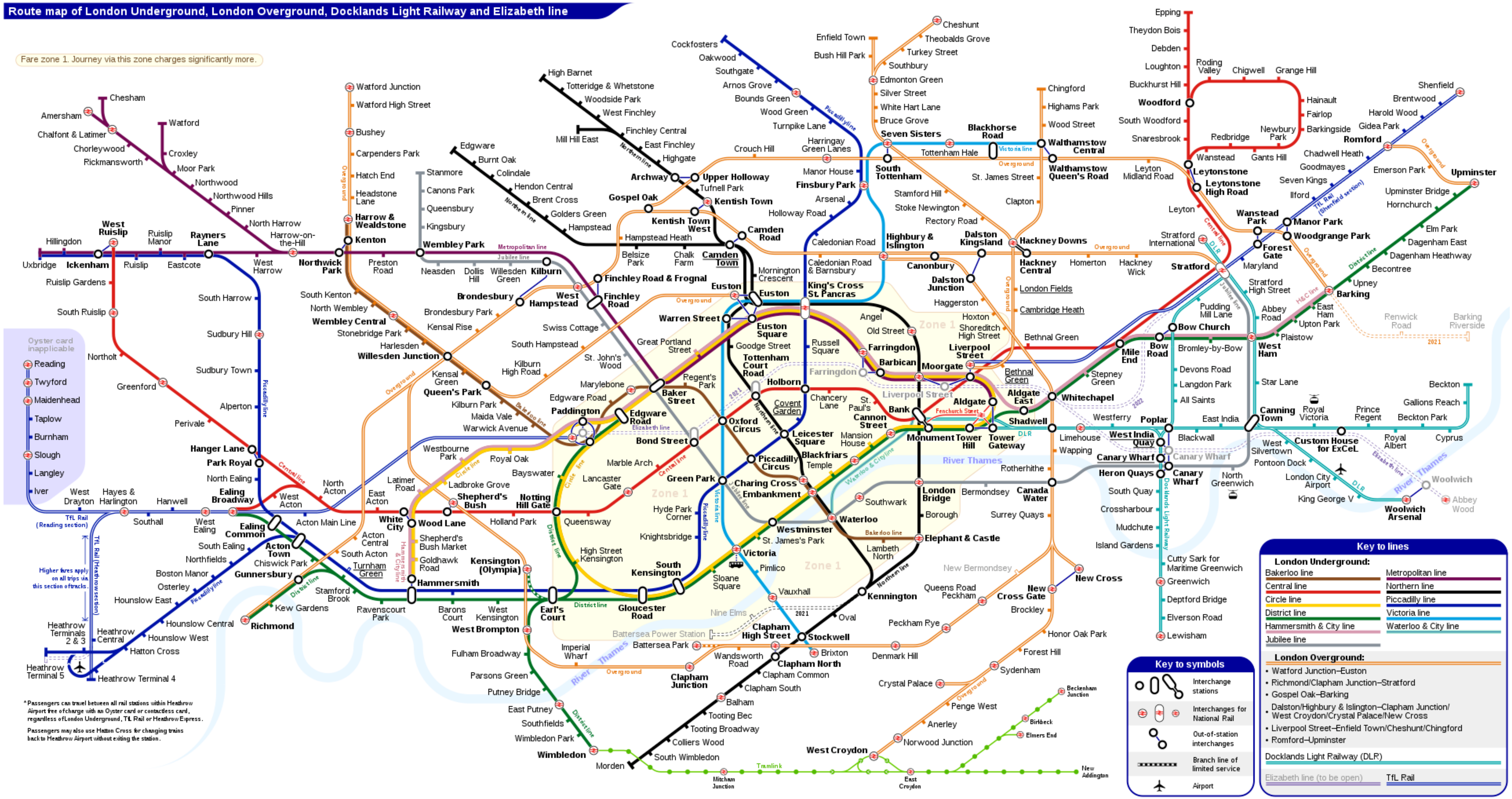


Source: Wiki Commons: Networks of Major High Speed Rail Operators in Europe - CC BY-SA 3.0

Transportation Network – London Underground

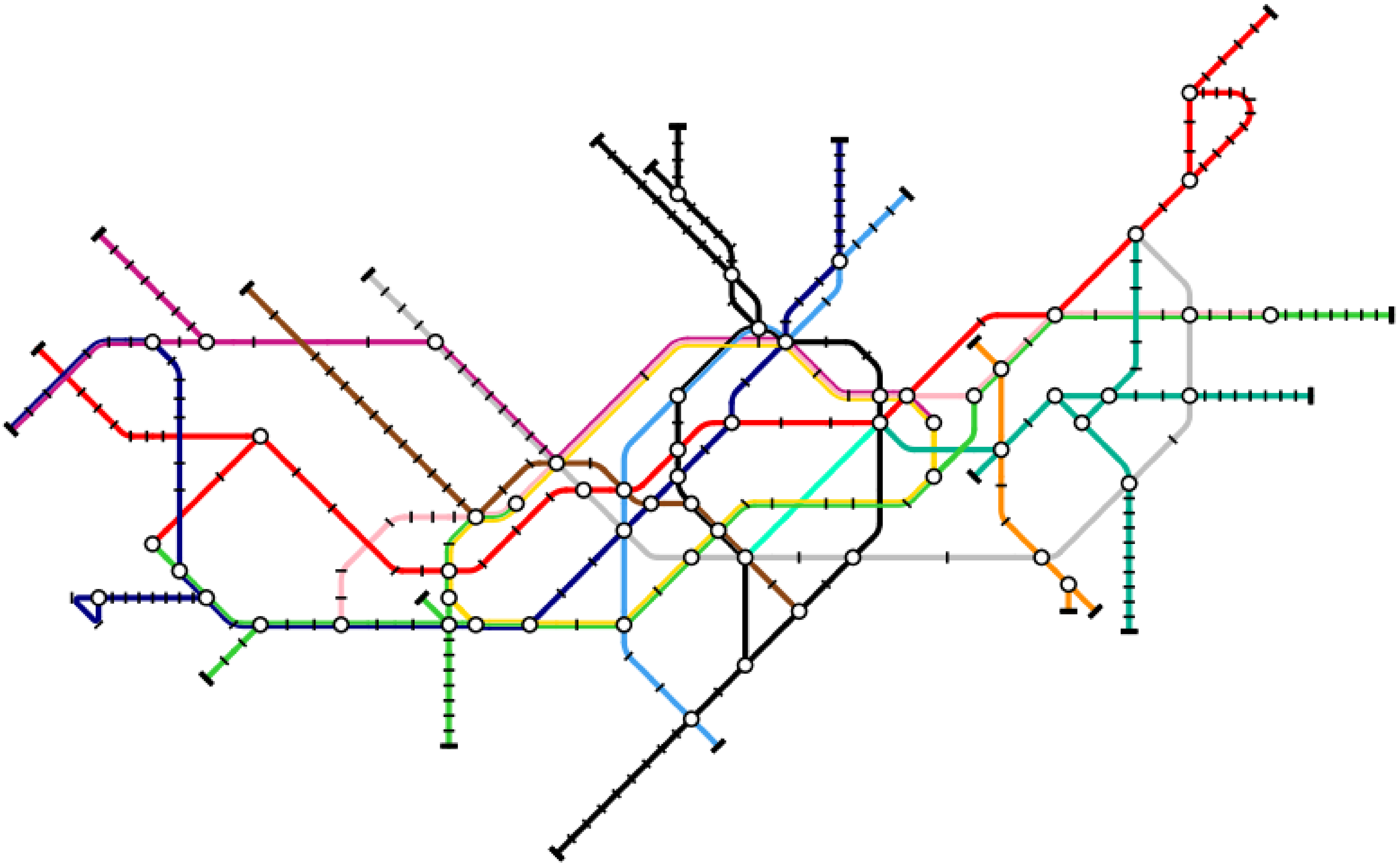
Source: Wiki Commons: London Underground full map - CC BY-SA 3.0

Transportation Network – London Underground



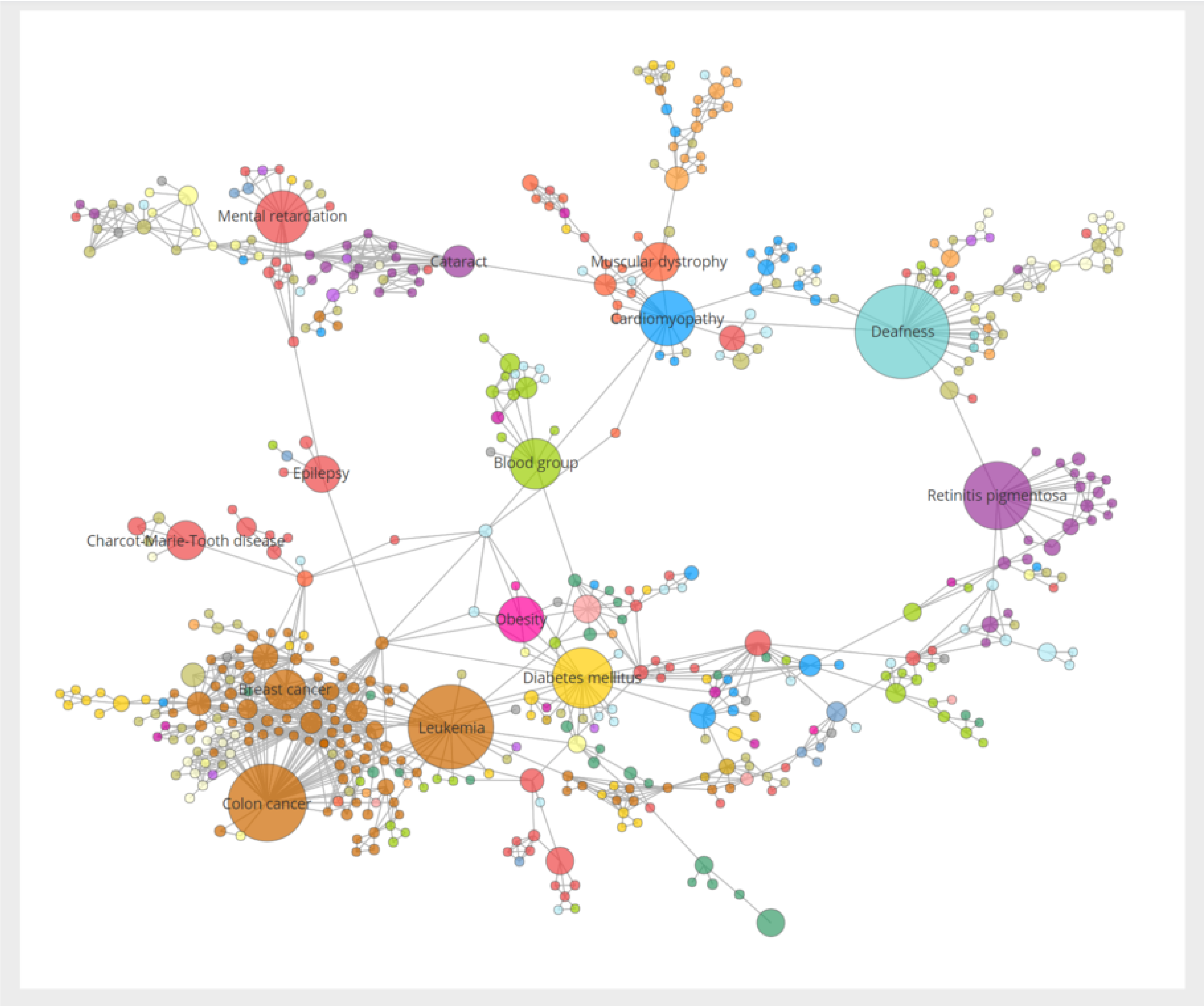
Source: Wiki Commons: London Underground Overground DLR Crossrail map - CC BY-SA 4.0

Transportation Network – London Underground



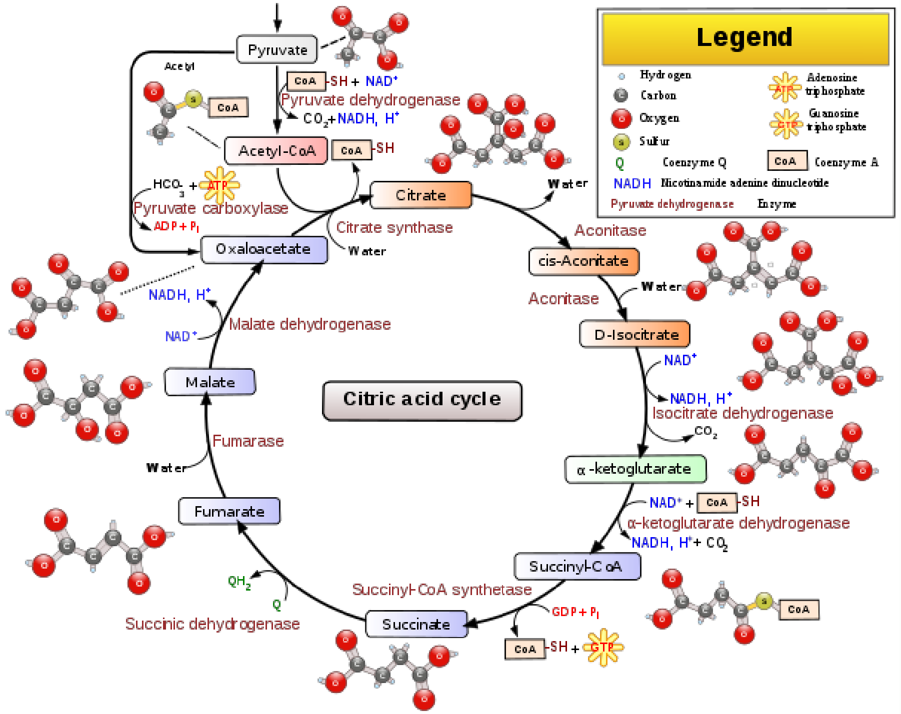
Drawing and Labeling High-Quality Metro Maps by Mixed-Integer Programming. Nöllenburg & Wolff. TVCG 2011, https://dx.doi.org/10.1109/TVCG.2010.81

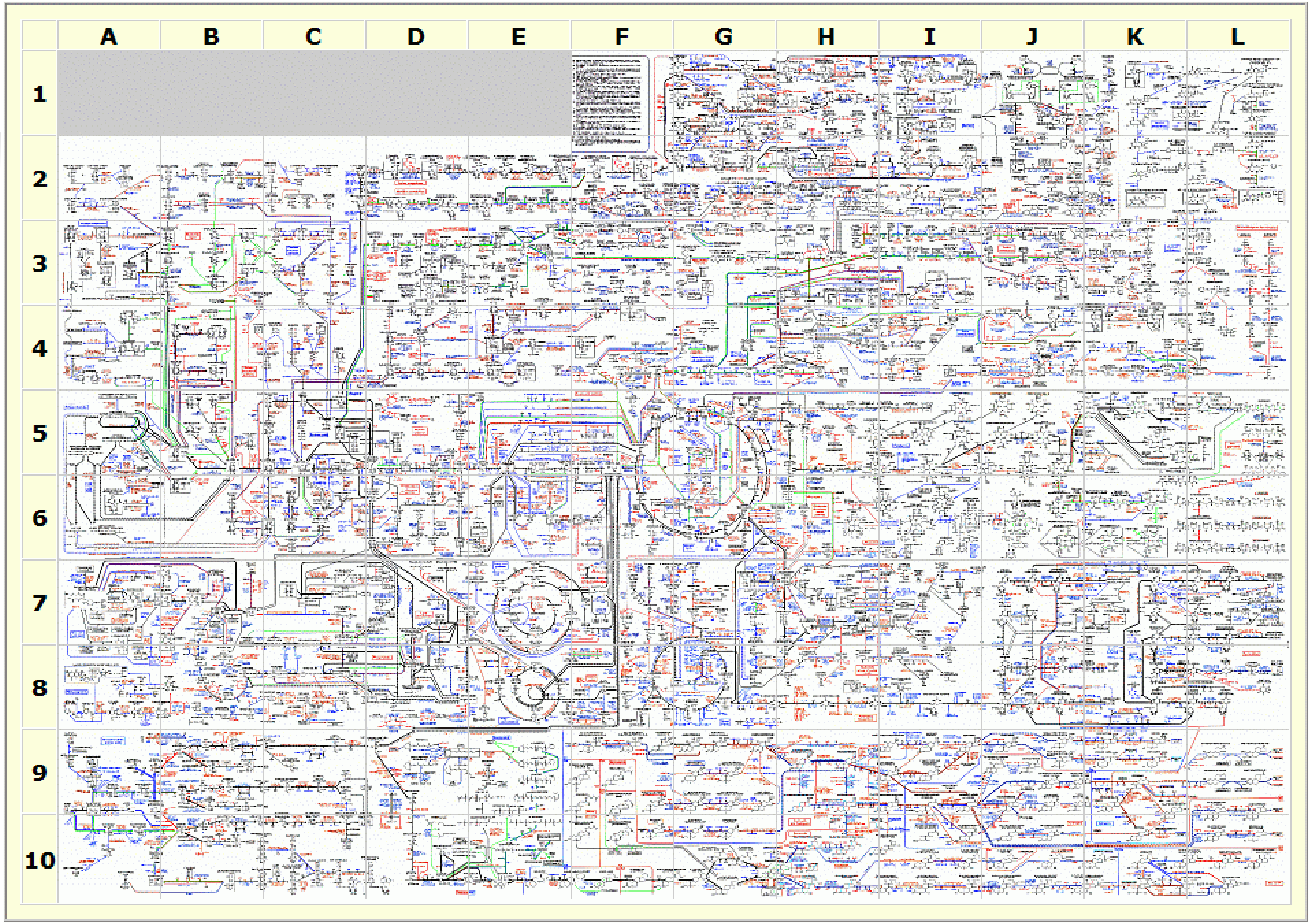
Bioinformatics – Disease Interaction



Source: Wiki Commons: Human disease network - CC BY-SA 4.0

Bioinformatics – Molecular Metabolic Network

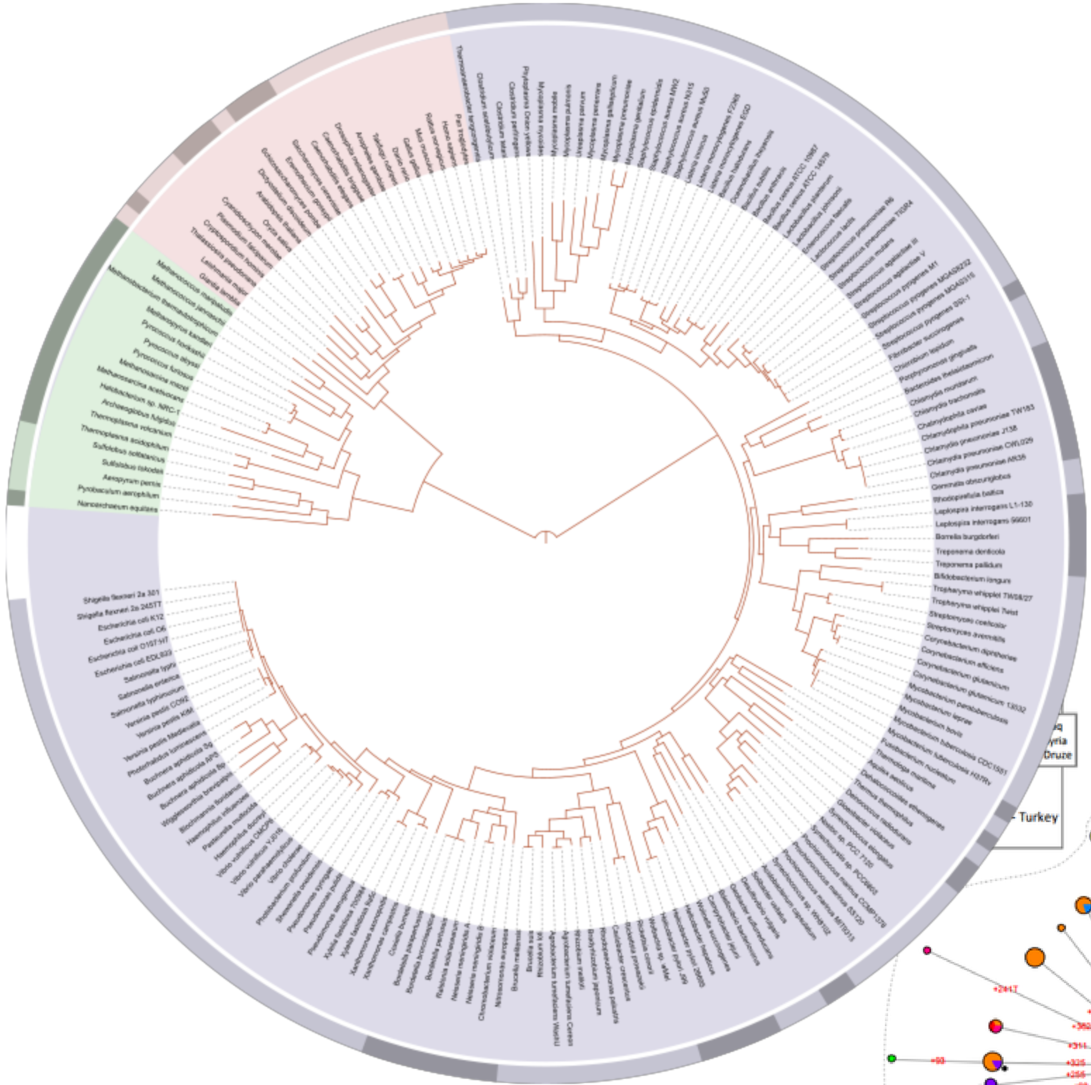


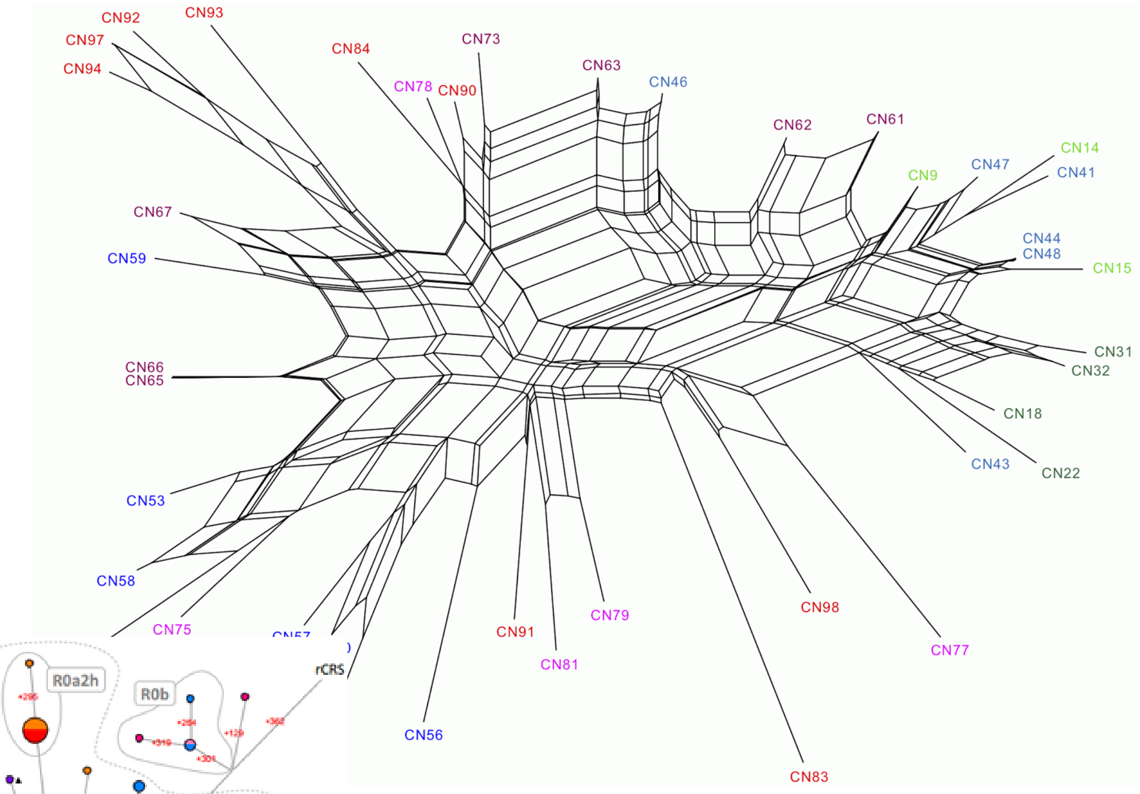


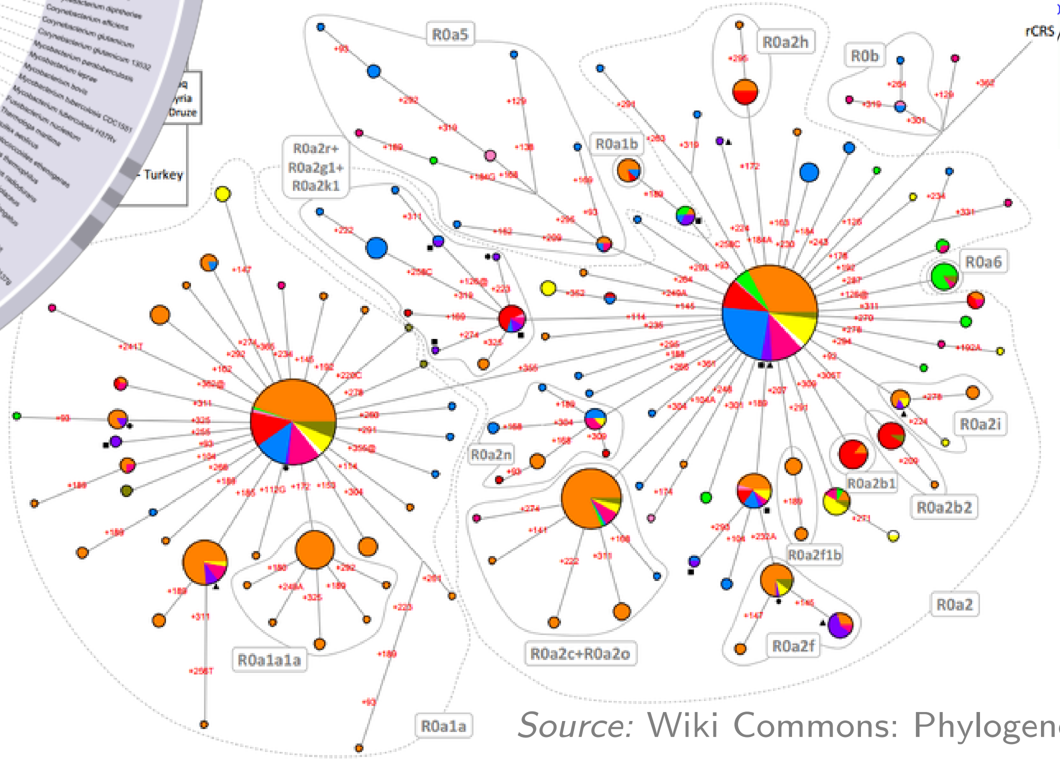
Source: Wiki Commons: Citric acid cycle with aconitate 2 - CC BY-SA 3.0

Source: Thiele et al., Nature Biotechnology 31, 419–425 (2013)

Bioinformatics – Phylogenetic Trees & Networks

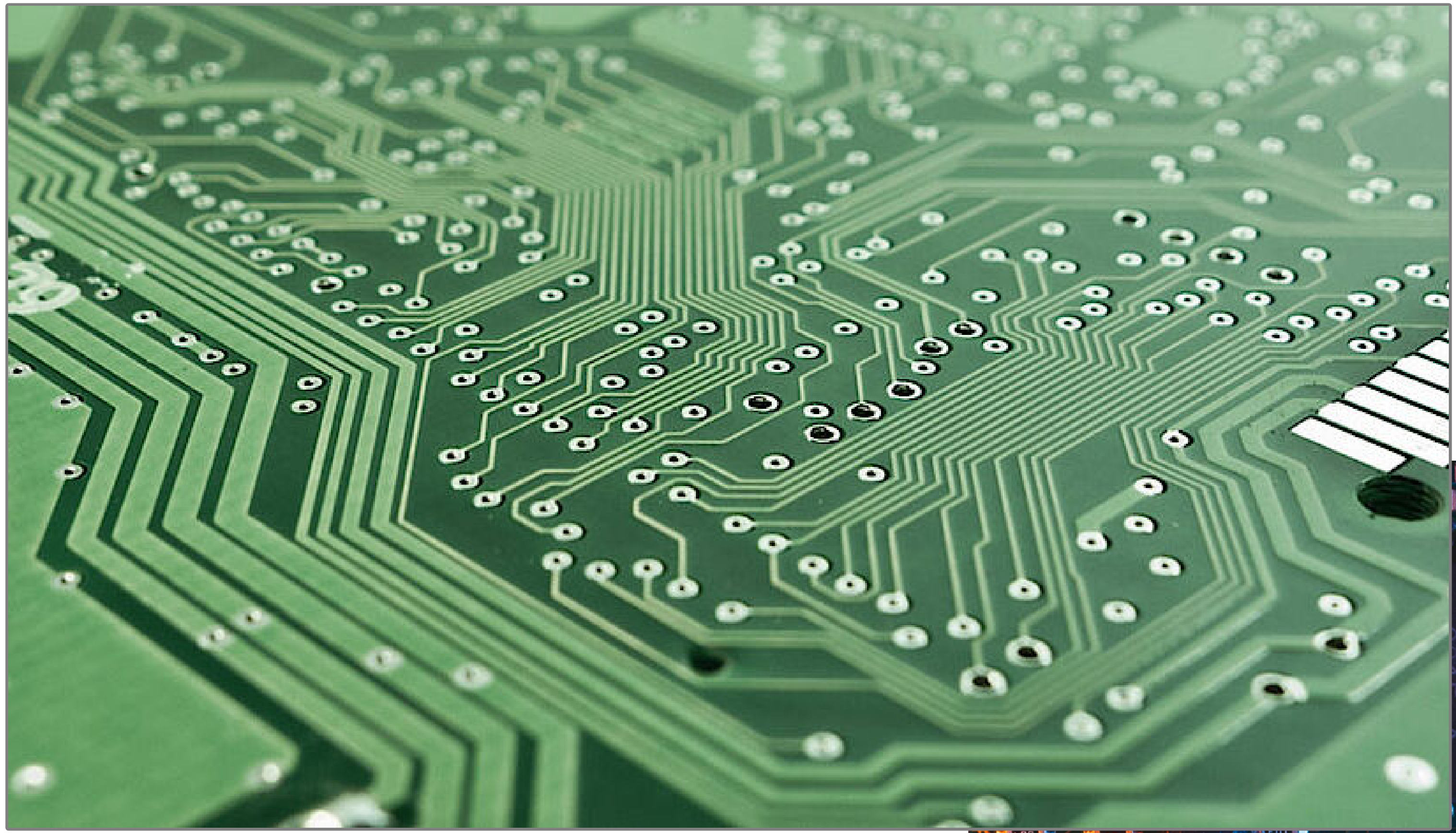




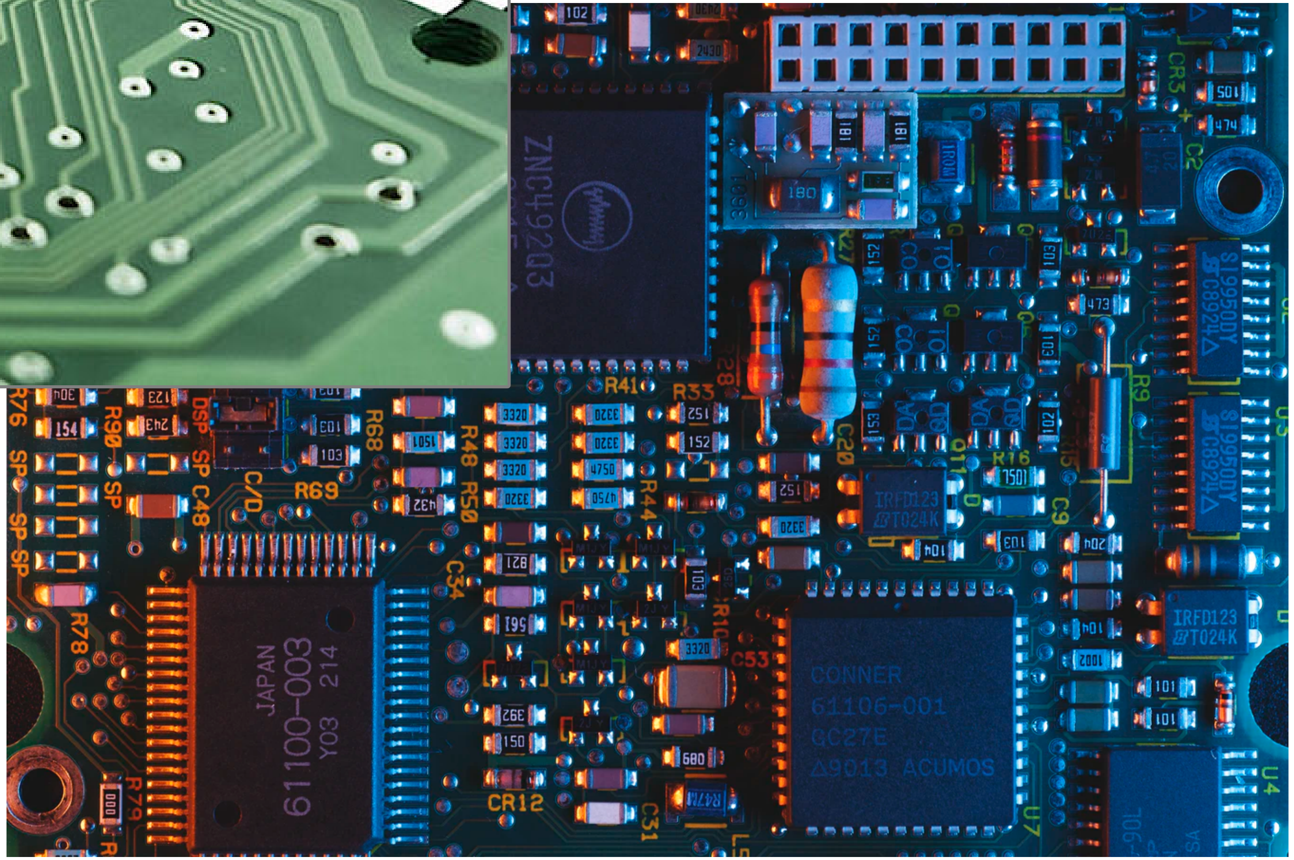


Source: Wiki Commons: Phylogenetic network of HVS-I variation - CC BY 4.0

Technical Network – Very Large-Scale Integration (VLSI)

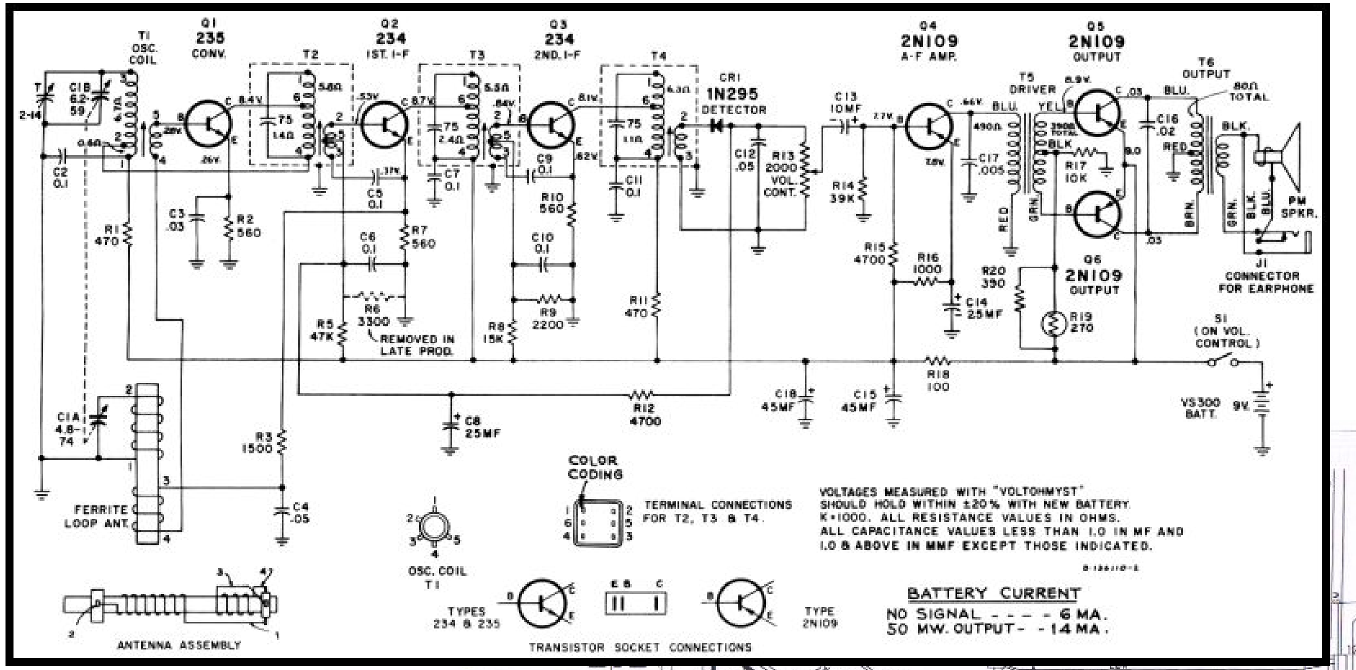


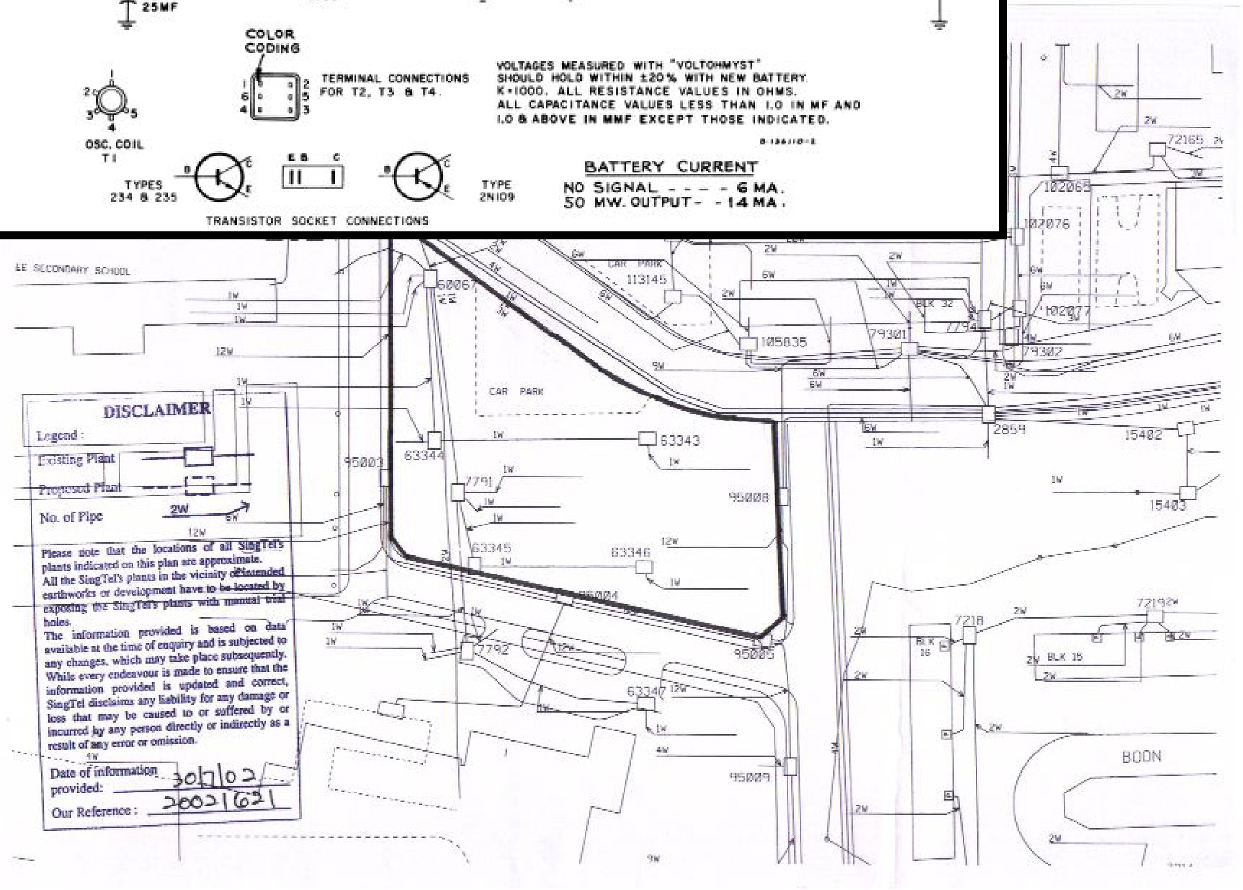
Source: Pixabay

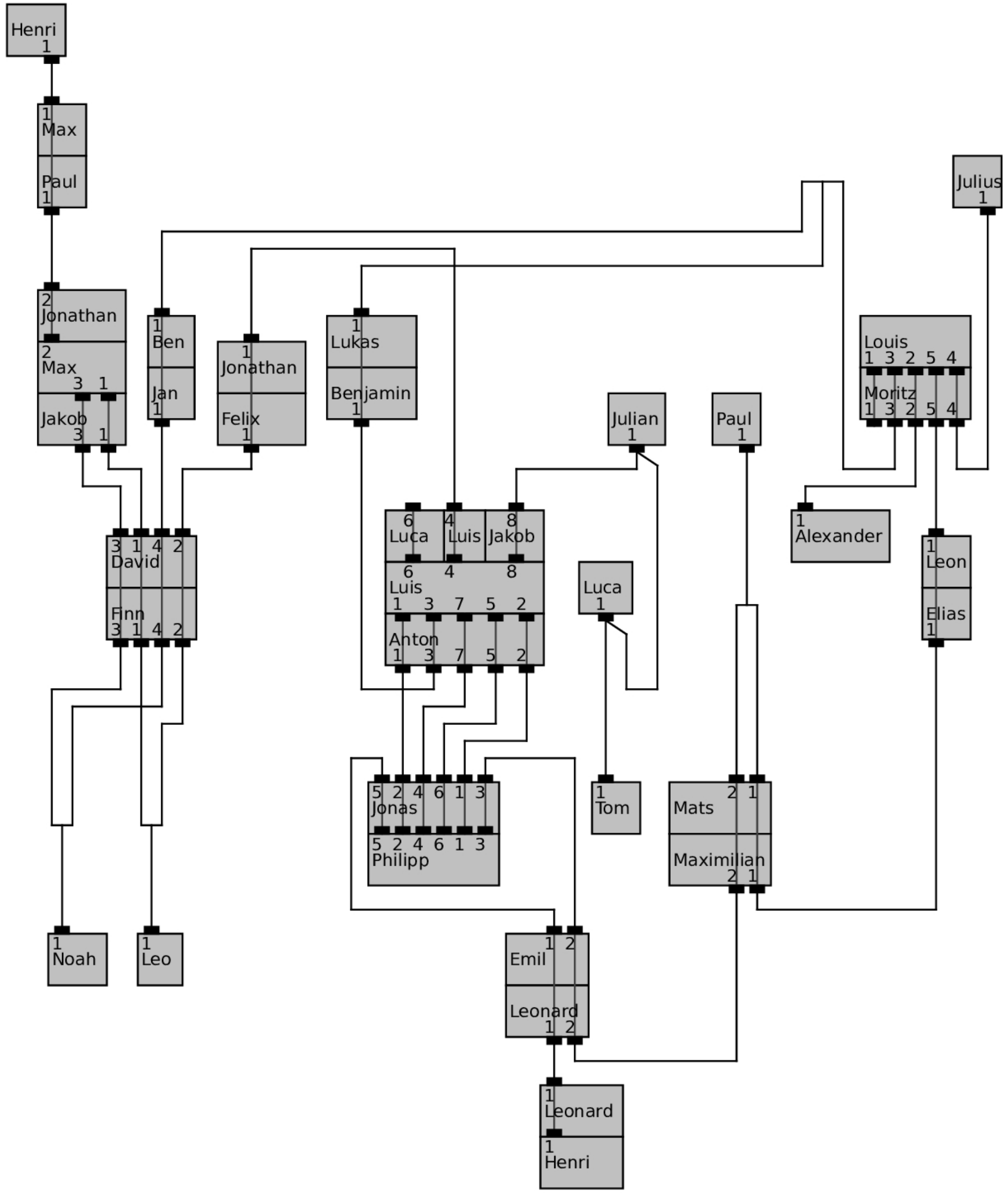


Source: Wiki Commons: Diopsis - CC BY-SA 3.0

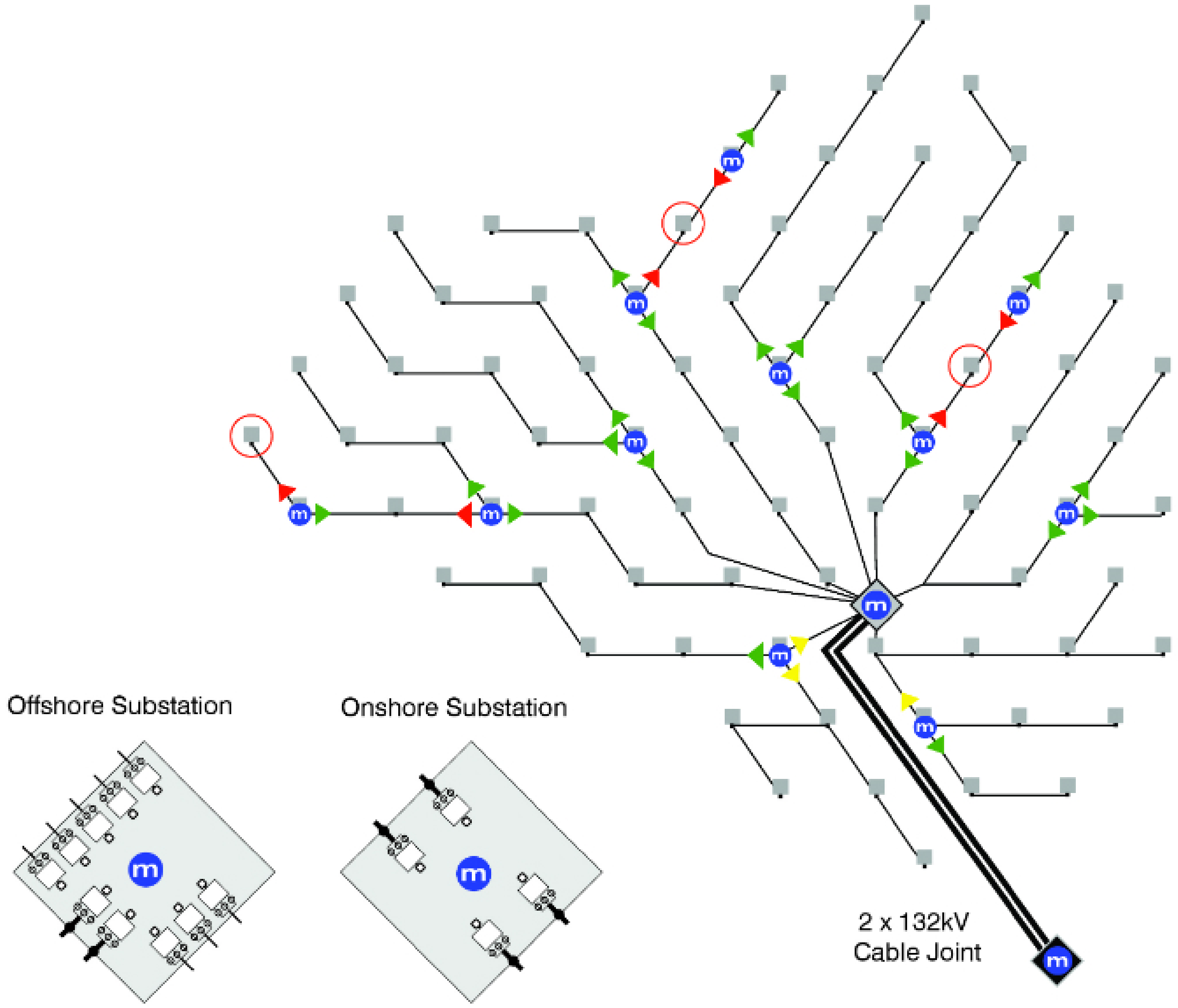
Technical Network – Transistor Diagram, Wiring Diagram





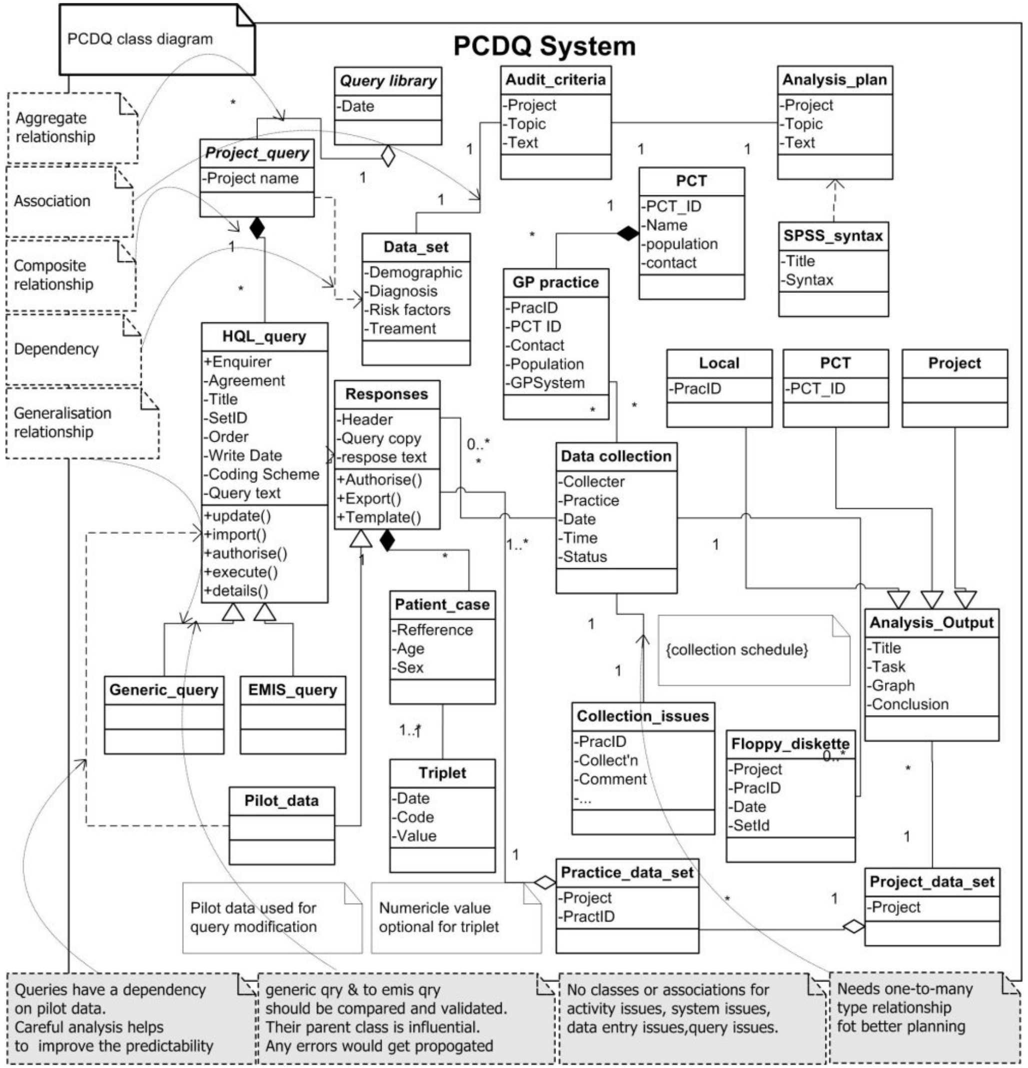


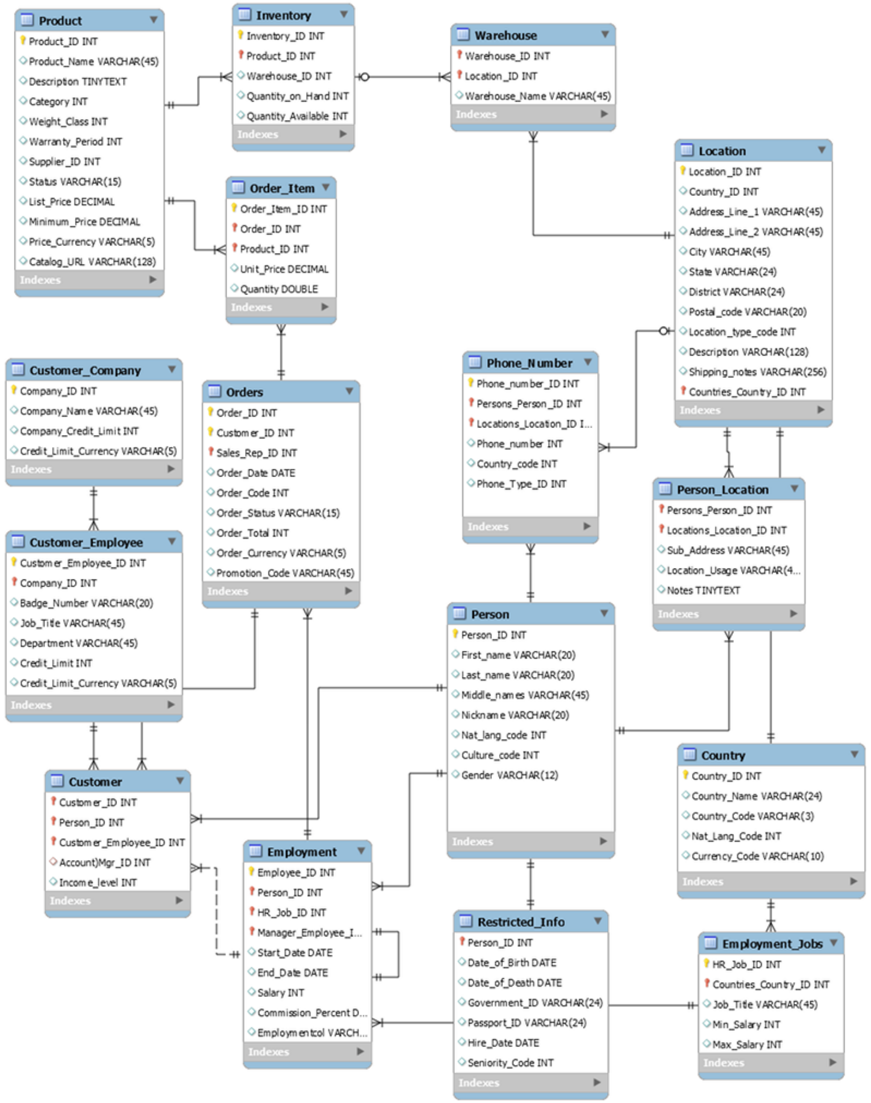
Technical Networks – Offshore Wind Farms



Source: Wiki Commons: Alpha Ventus Windmills -

Technical Network – UML Diagram

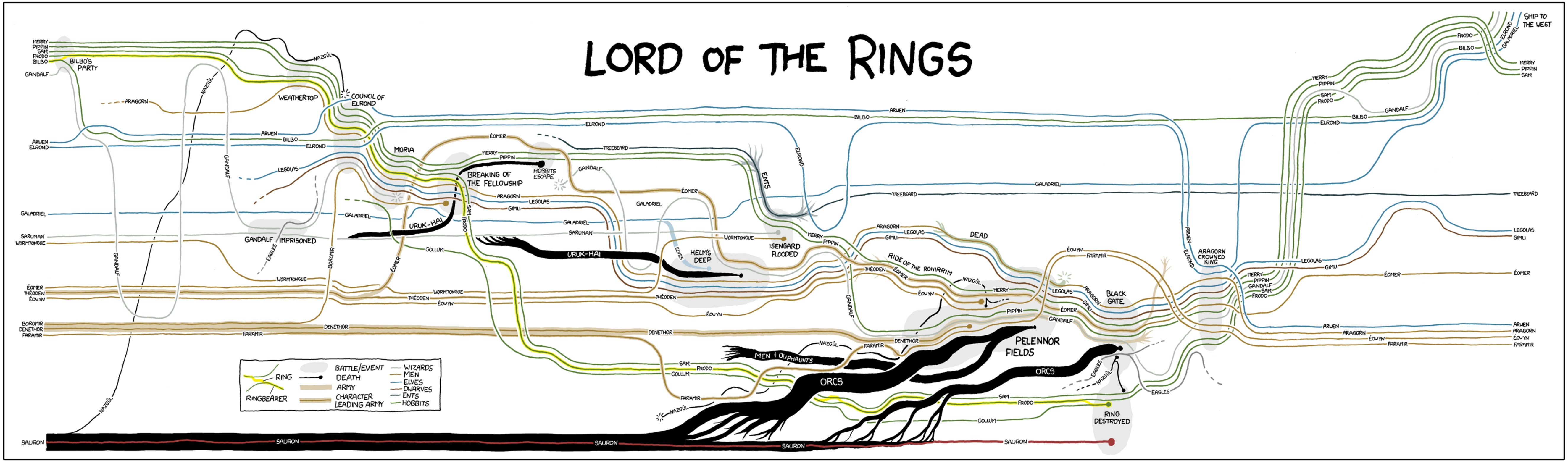




© AWS

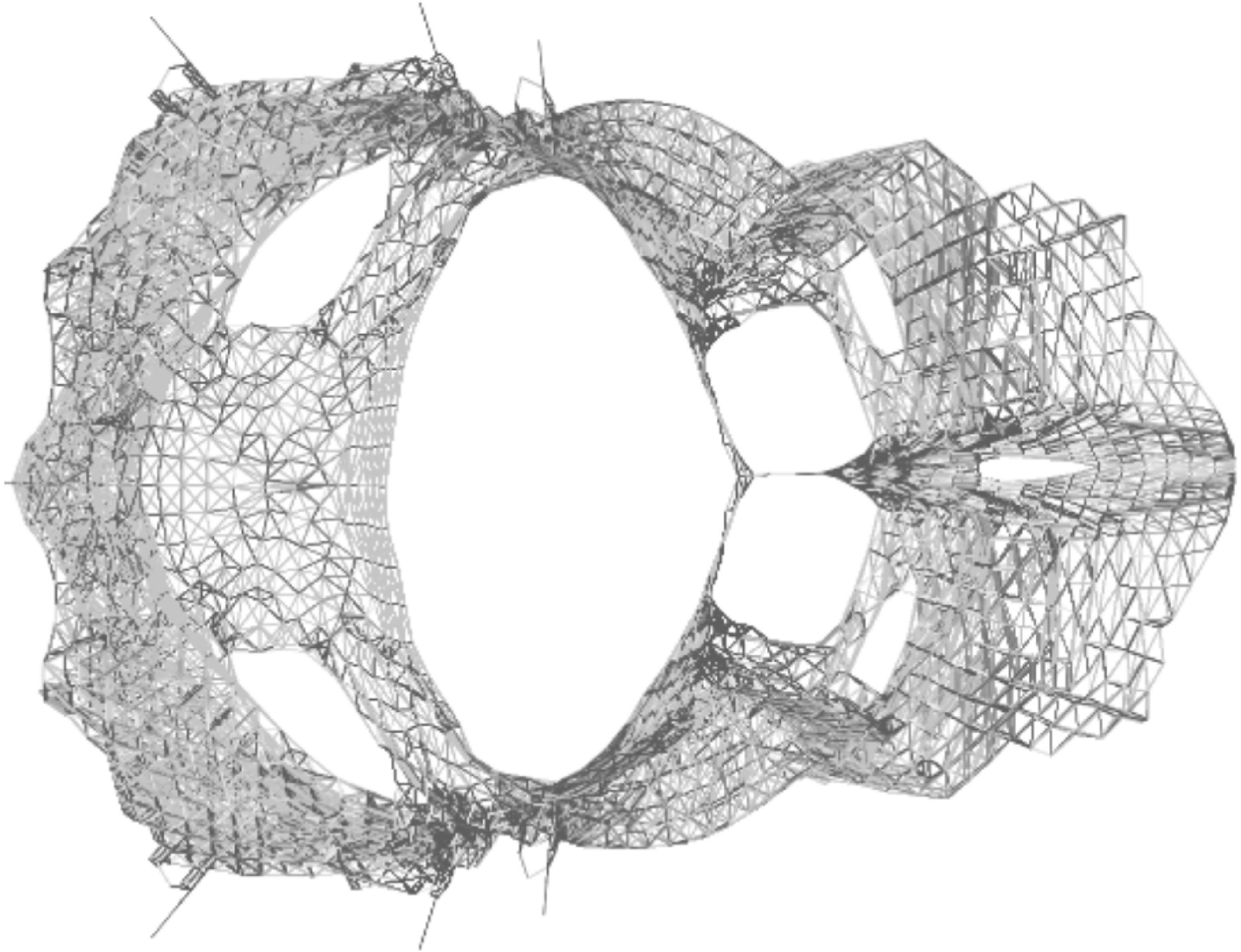
Temporal Graph Layout – Storylines

THESE CHARTS SHOW MOVIE CHARACTER INTERACTIONS.
THE HORIZONTAL AXIS IS TIME. THE VERTICAL GROUPING OF THE
LINES INDICATES WHICH CHARACTERS ARE TOGETHER AT A GIVEN TIME.

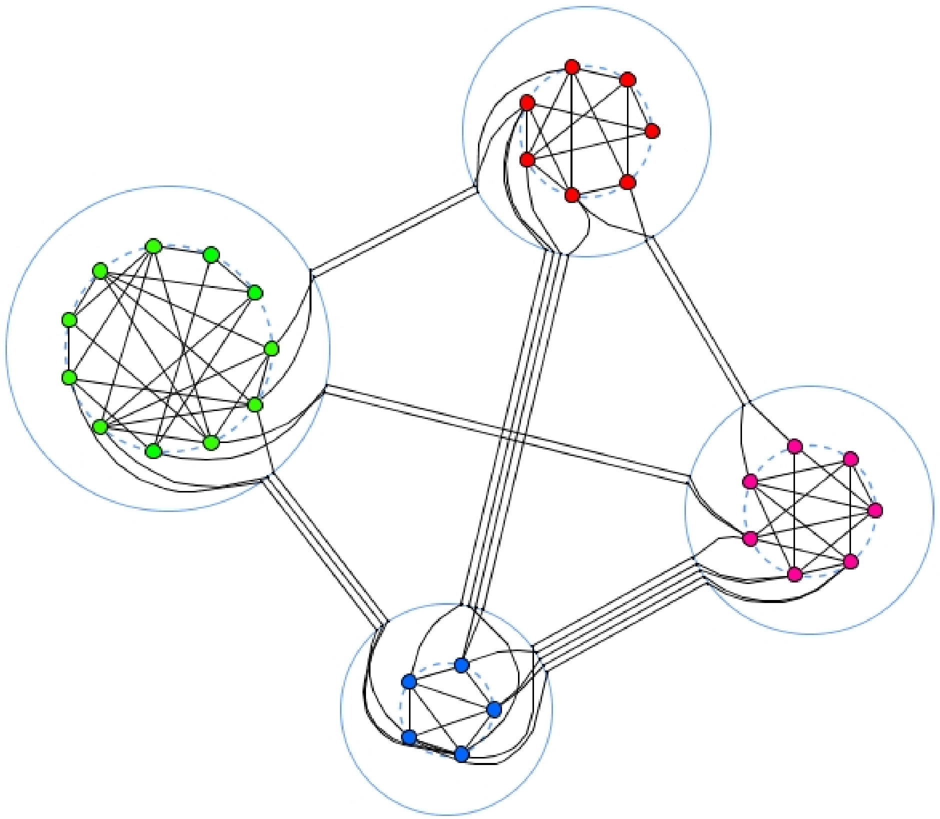


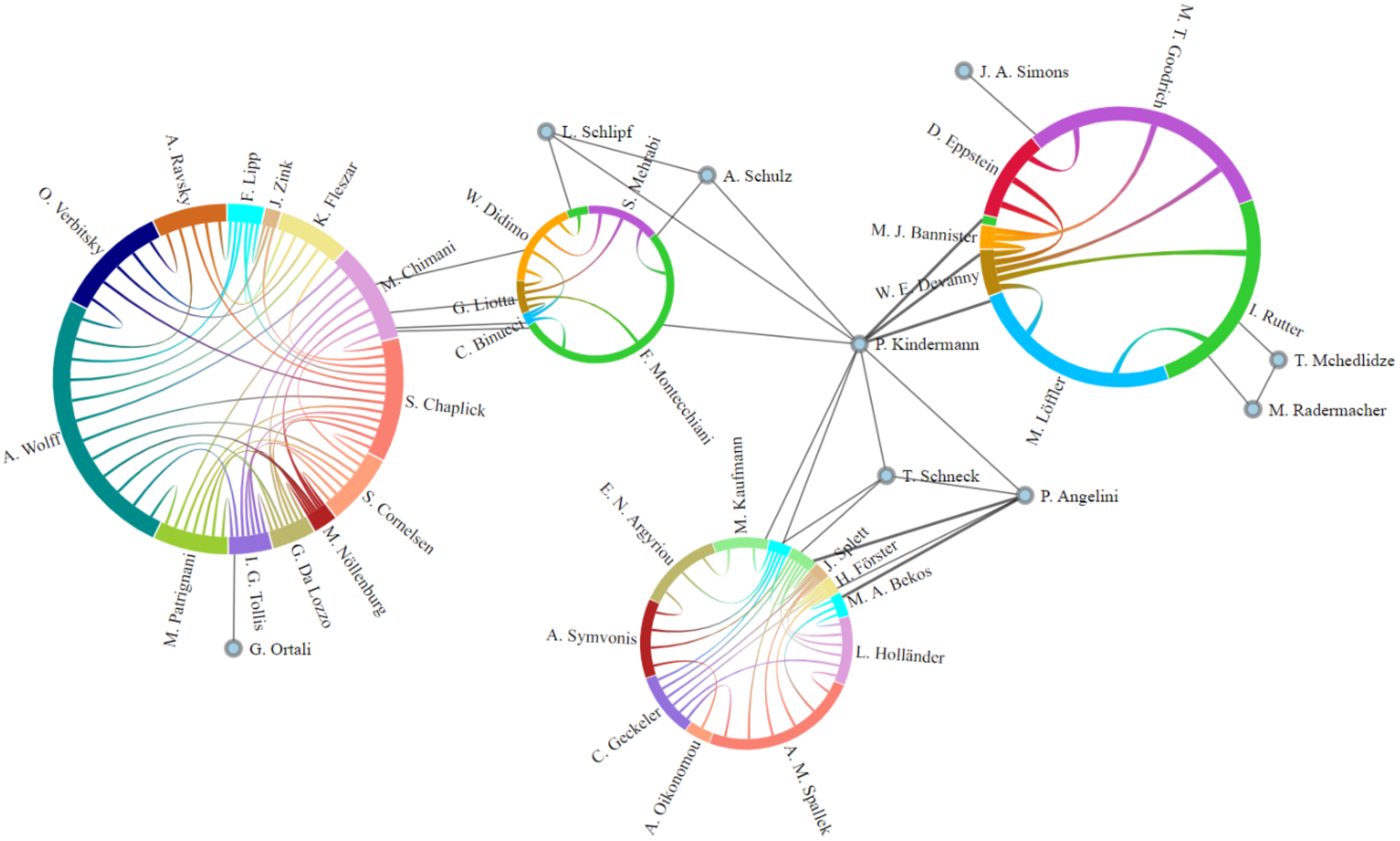
Source and more: xkcd Comic 657 – xkcd.com/657/

Large Graphs – Object Mesh



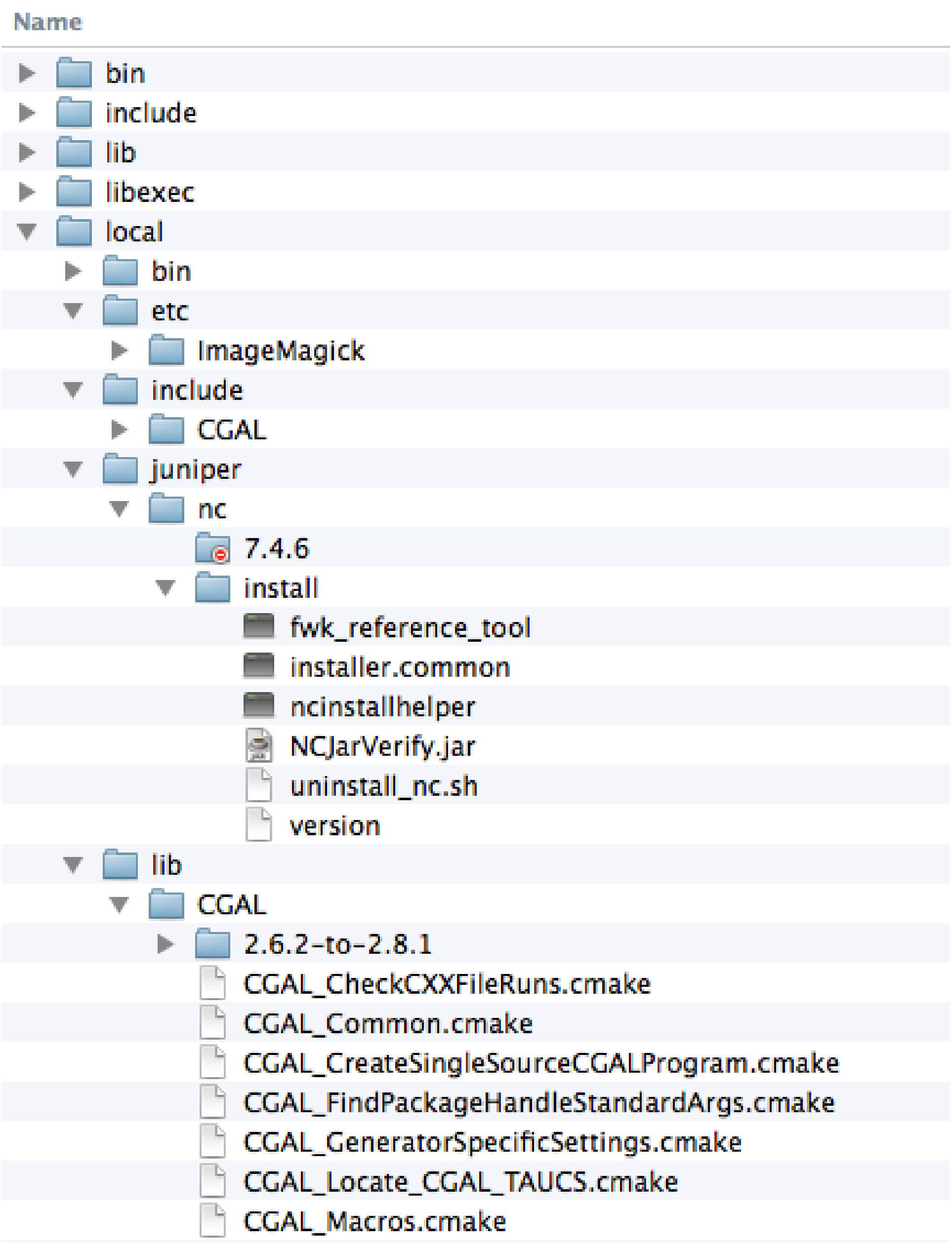
General Graphs – Micro–Macro Layout

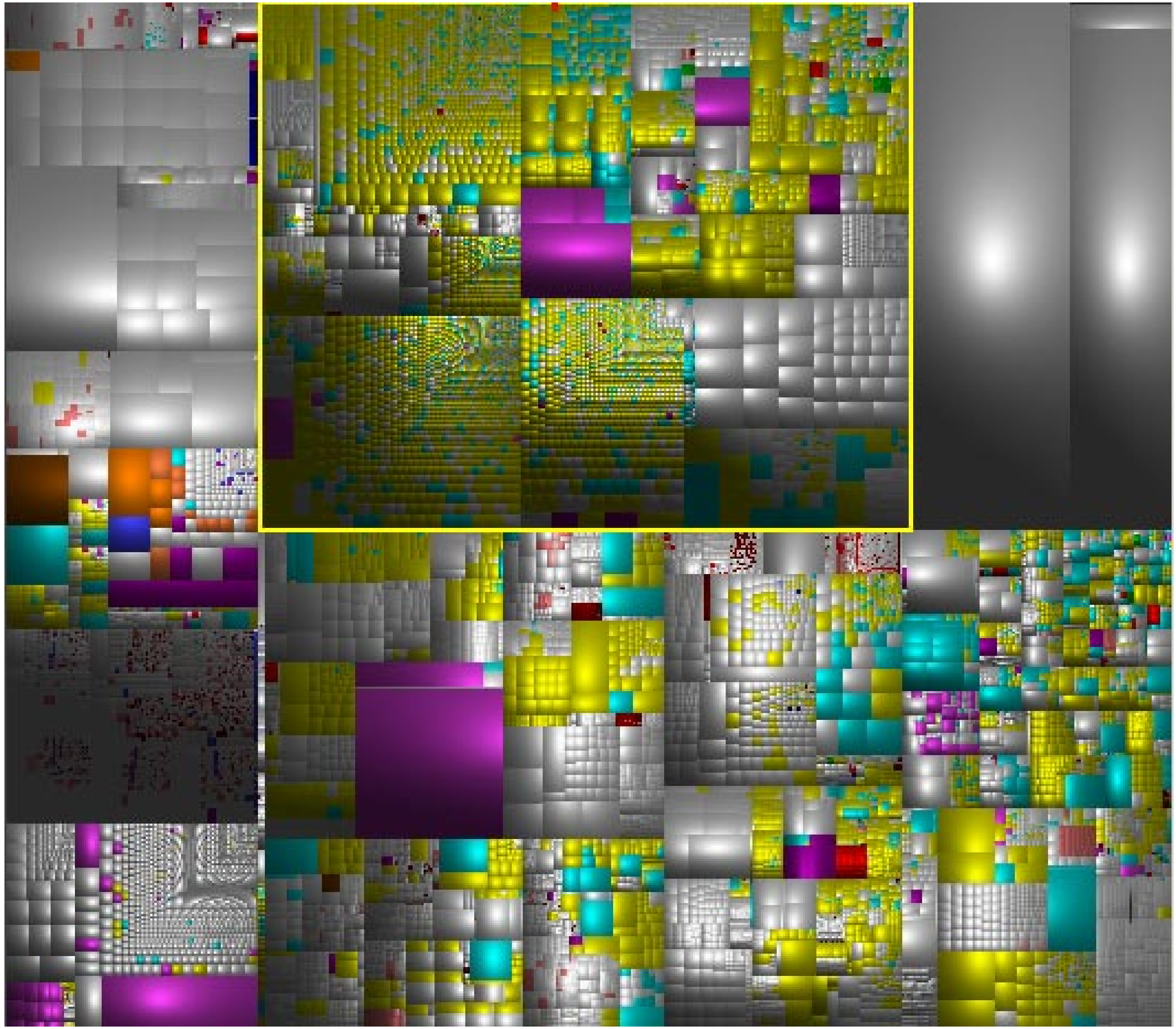




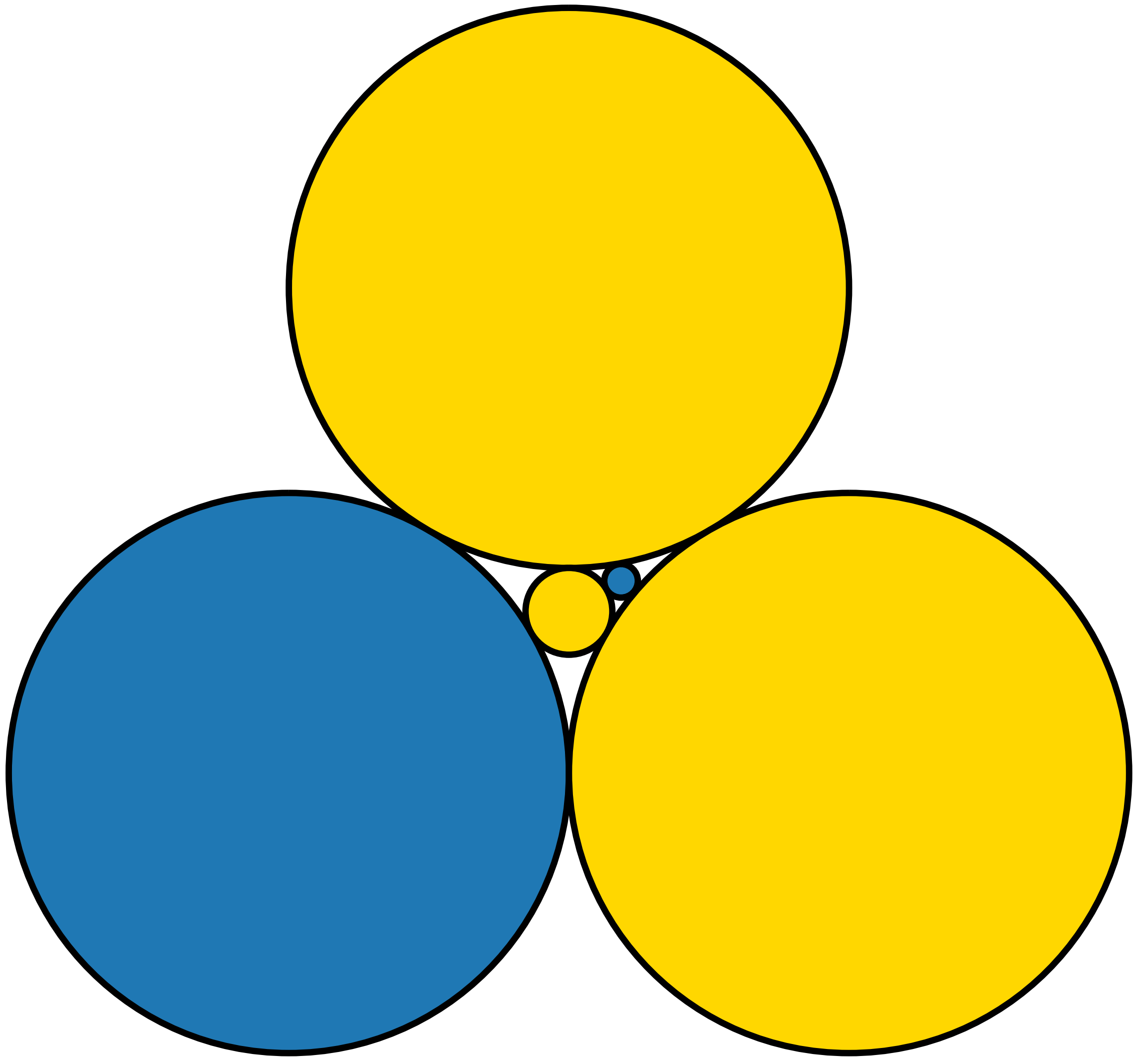
Source: Angori et al., ChordLink: A New Hybrid Visualization Model, GD'19 (2019)

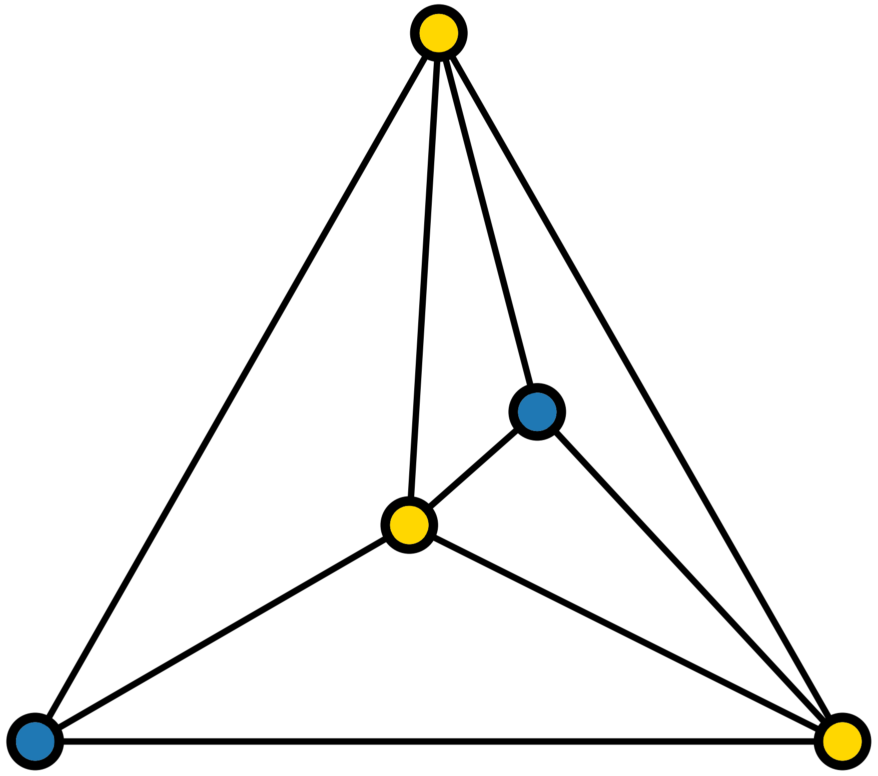
Alternative Representations – Treemap



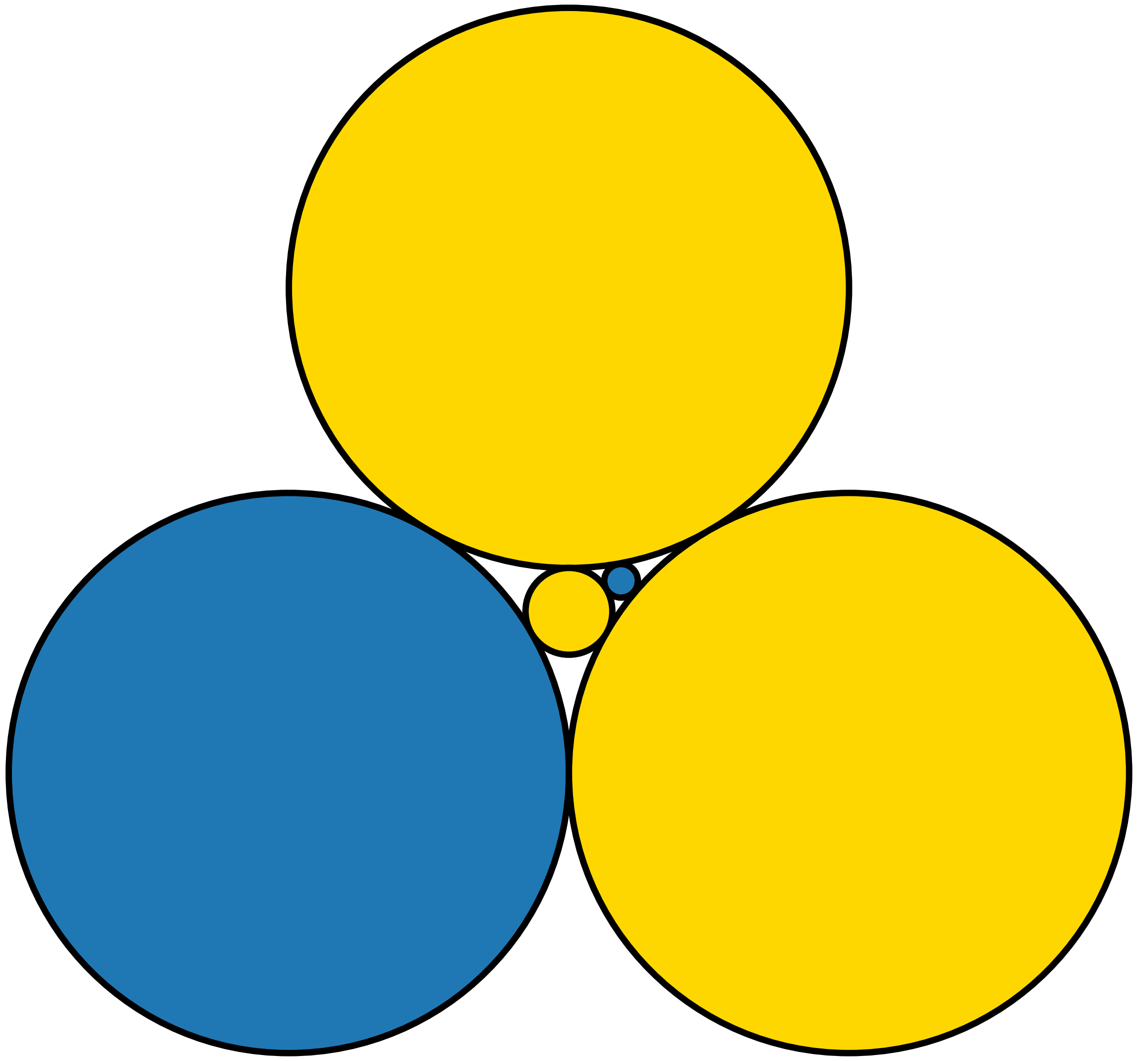


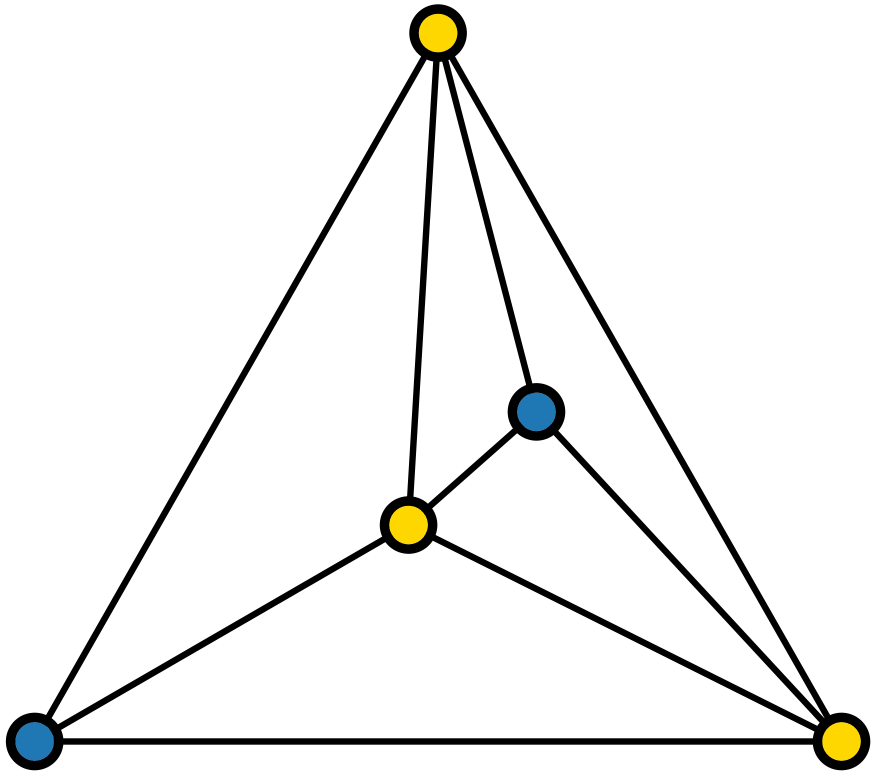
Alternative Representations – Contact Graphs





Alternative Representations – Contact Graphs



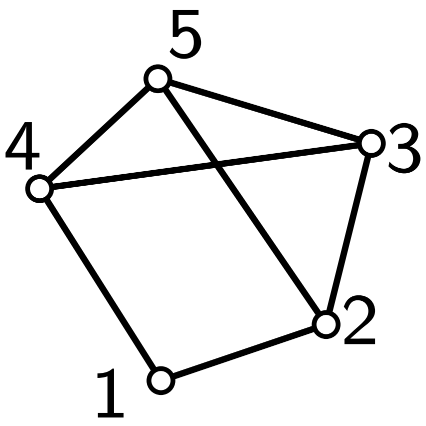


- For more examples see `visualcomplexity.com`

Requirements of a Graph Layout

1. Drawing conventions and requirements, e.g.,

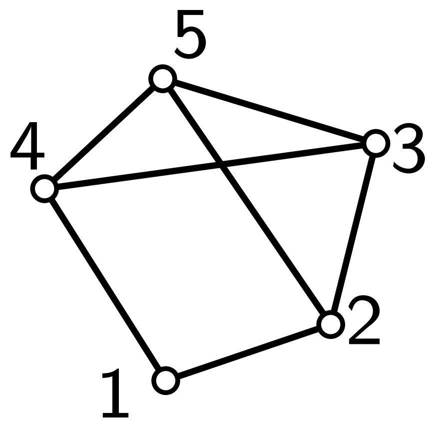
Requirements of a Graph Layout

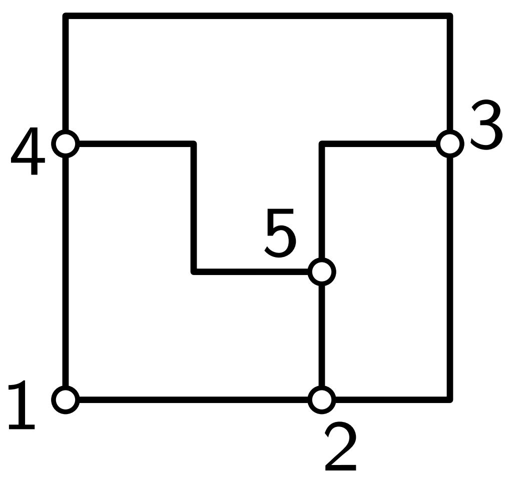


1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$

Requirements of a Graph Layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)

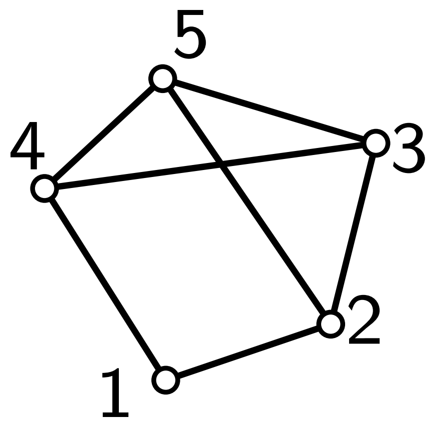


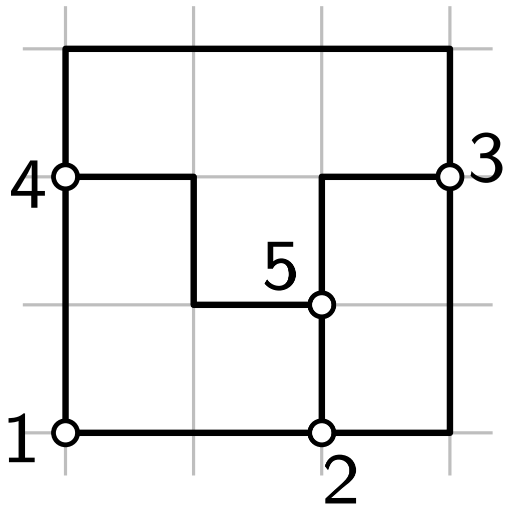


Requirements of a Graph Layout

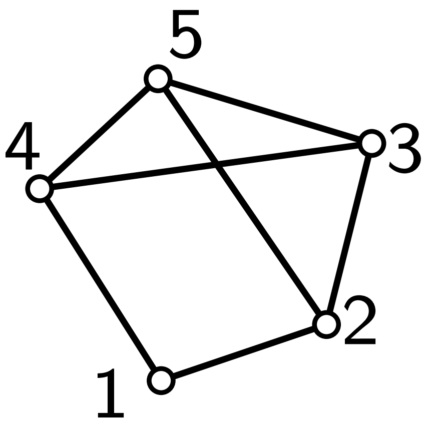
1. Drawing conventions and requirements, e.g.,

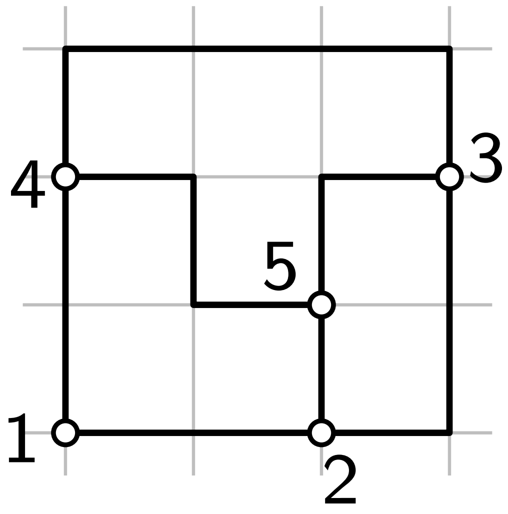
- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)
- grid drawings





Requirements of a Graph Layout





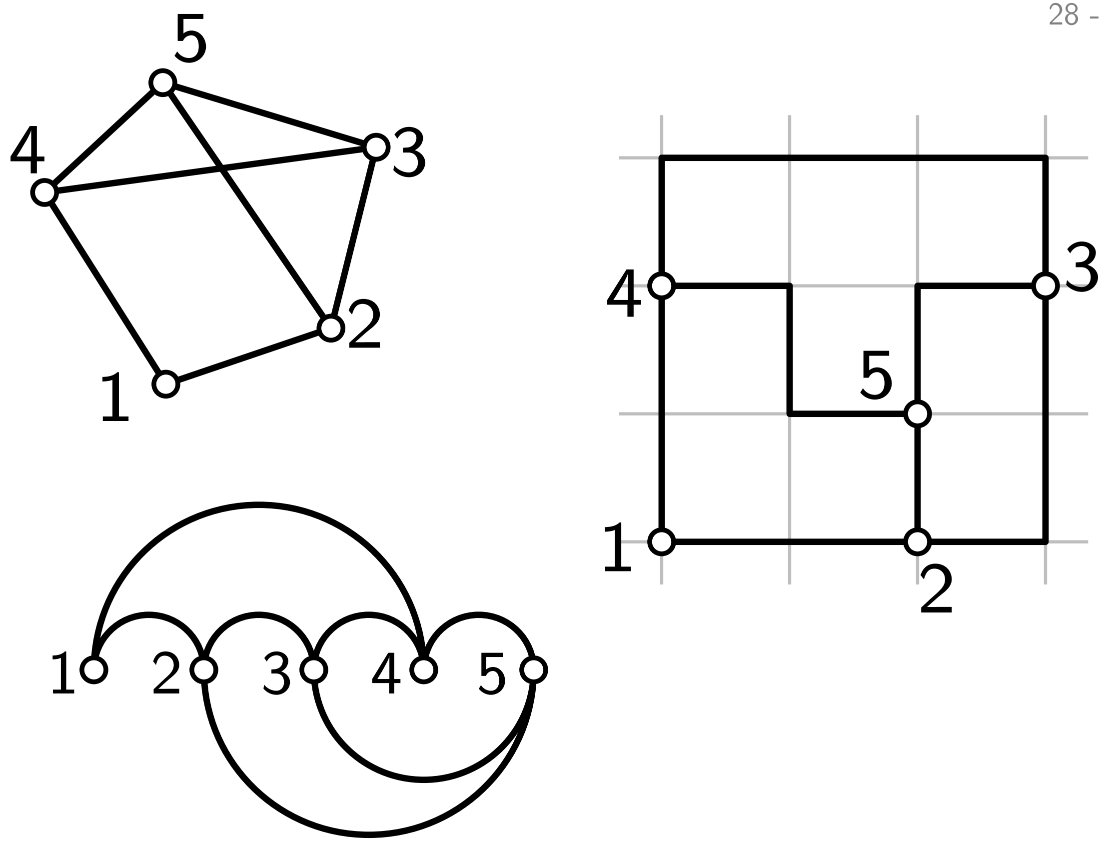
1. Drawing conventions and requirements, e.g.,

- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)
- grid drawings
- without crossing

Requirements of a Graph Layout

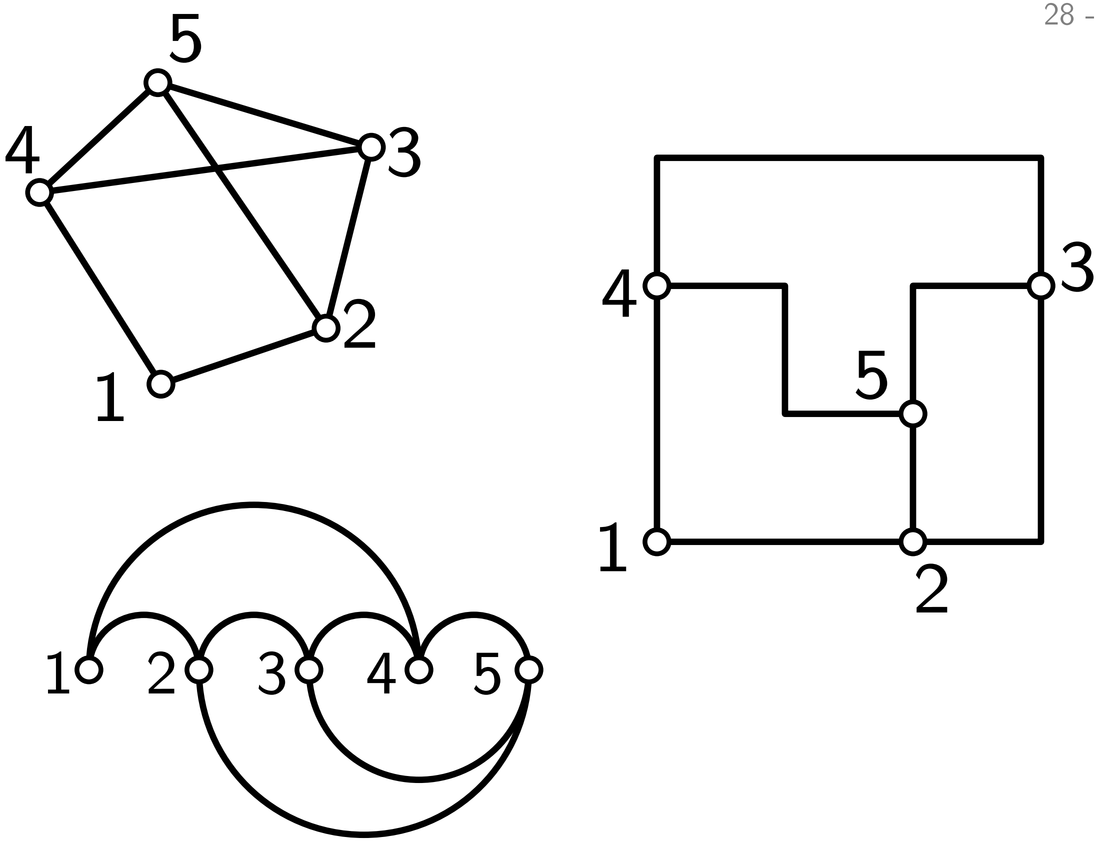
1. Drawing conventions and requirements, e.g.,

- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)
- grid drawings
- without crossing



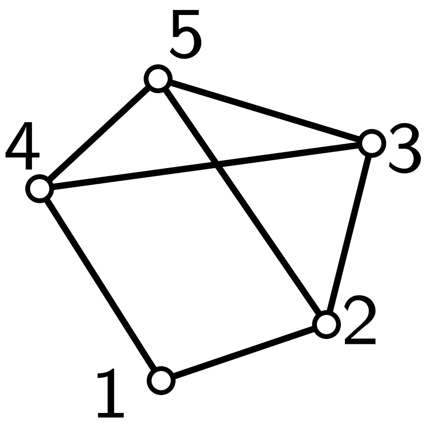
Requirements of a Graph Layout

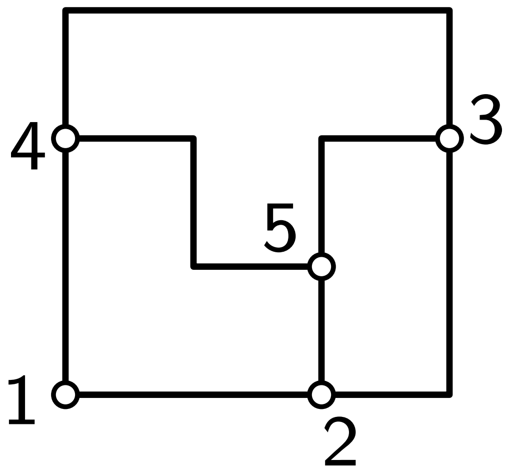
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.

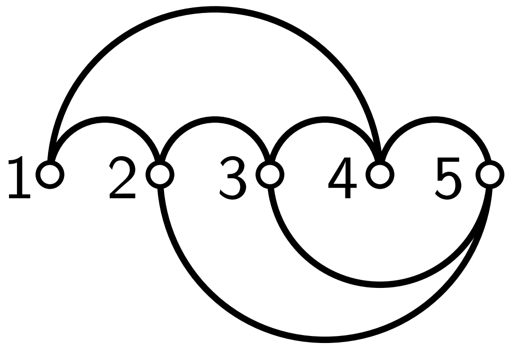


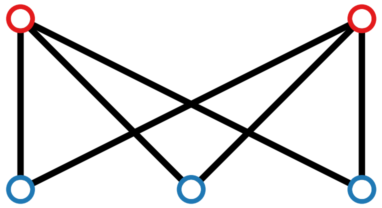
Requirements of a Graph Layout

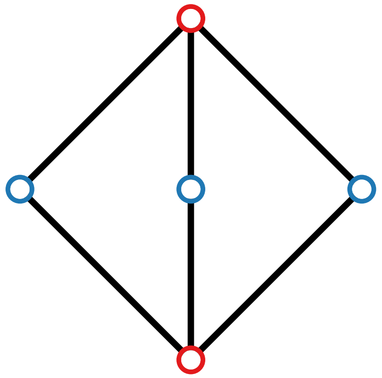
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization





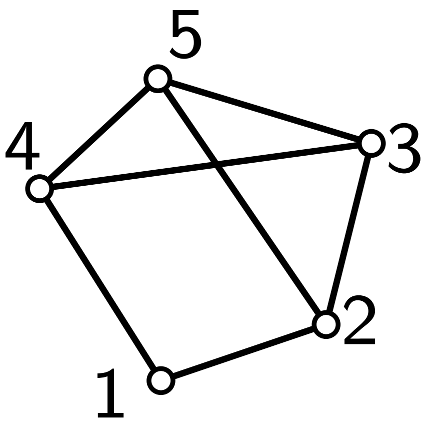


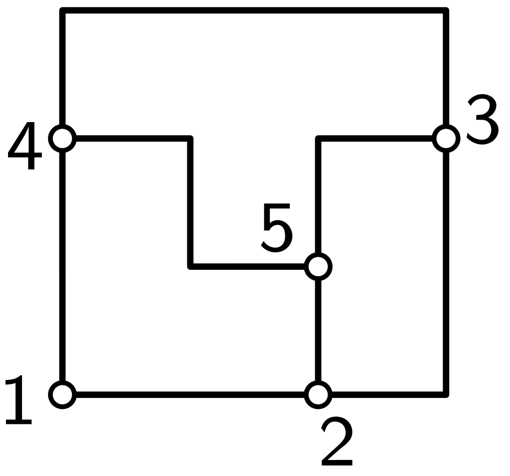


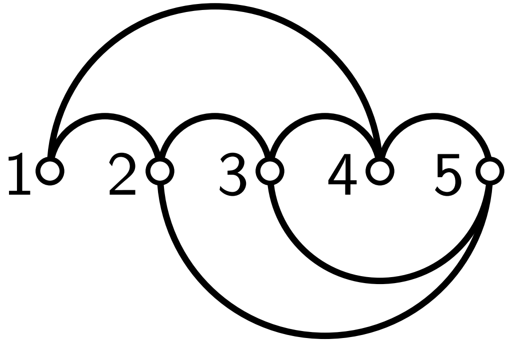


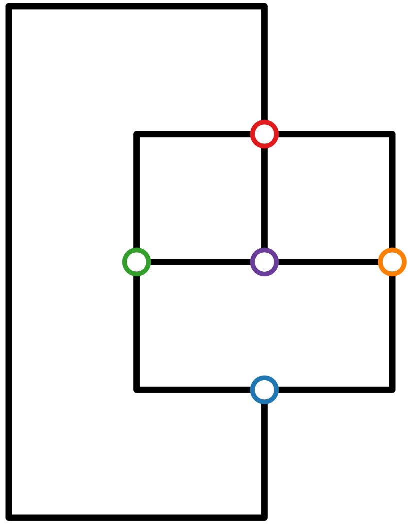
Requirements of a Graph Layout

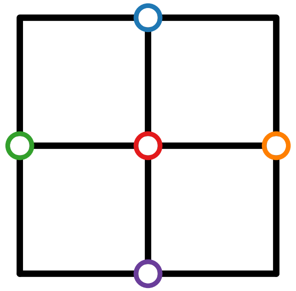
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization





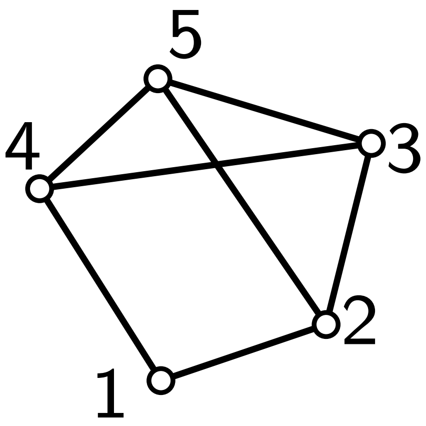


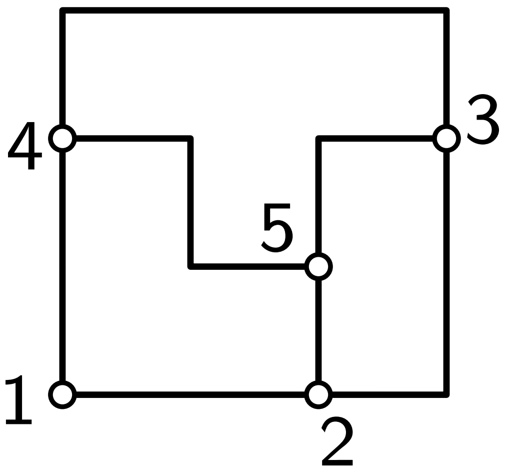


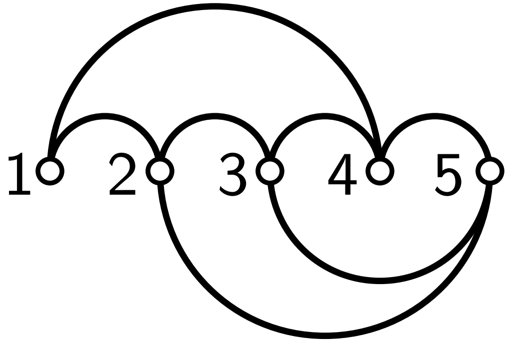


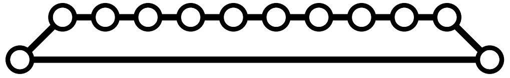
Requirements of a Graph Layout

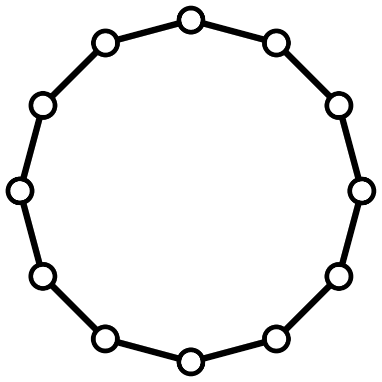
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity





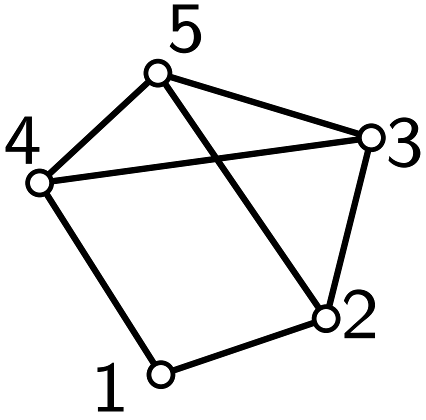


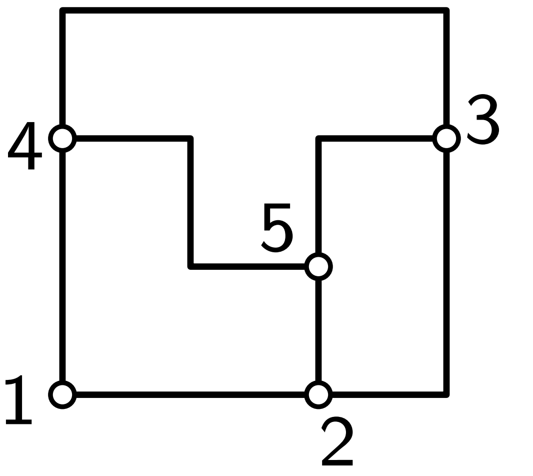


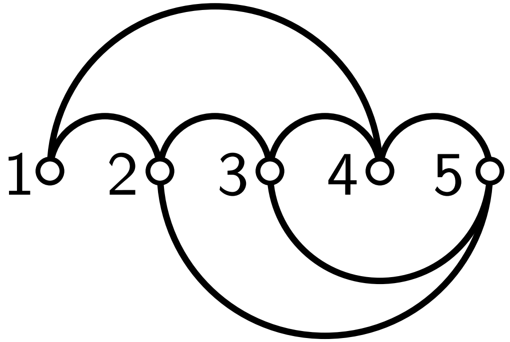


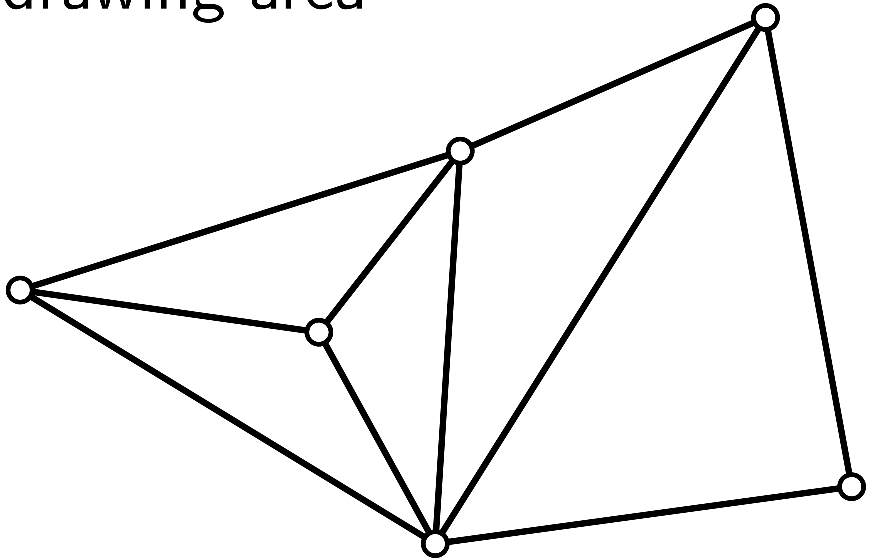
Requirements of a Graph Layout

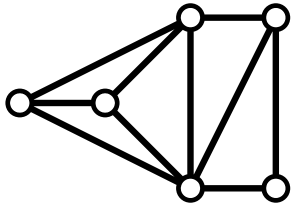
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area



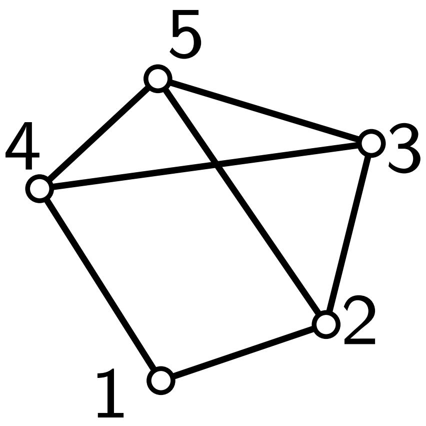


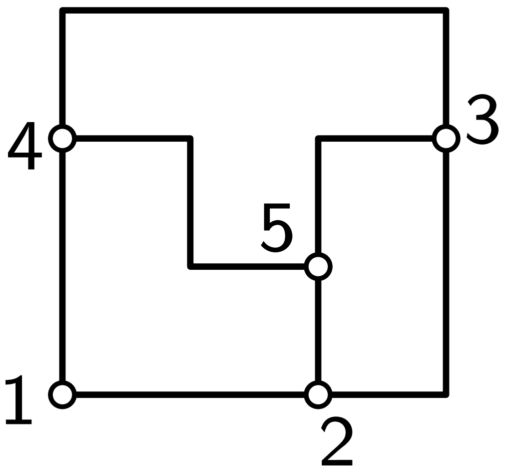


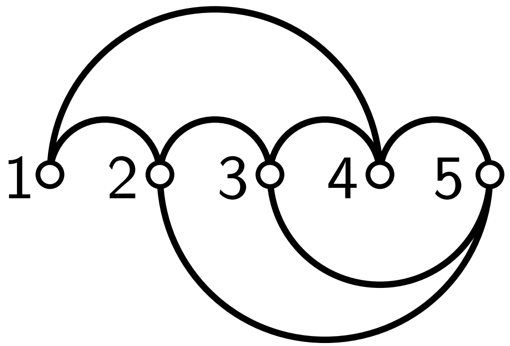




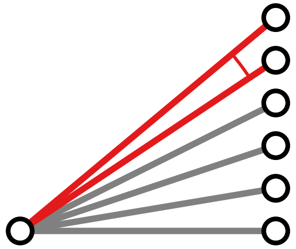
Requirements of a Graph Layout

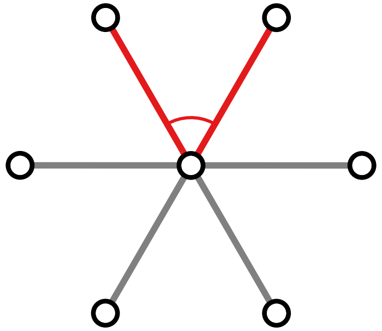






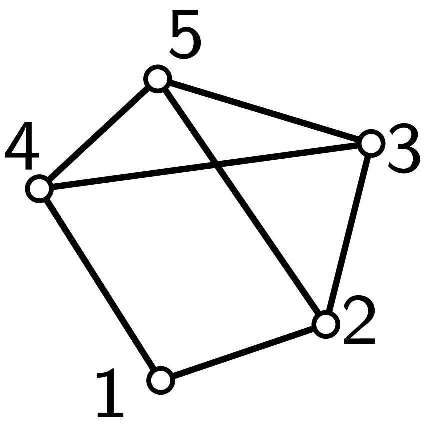
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution

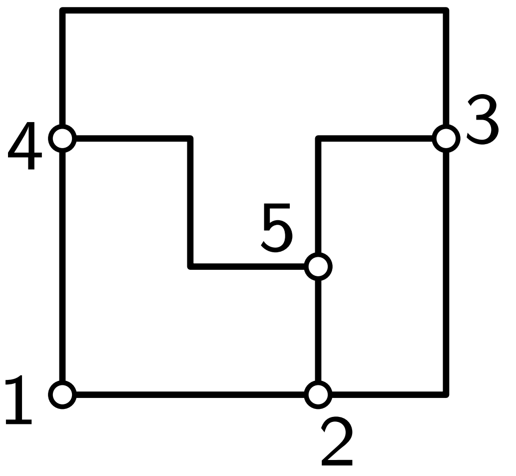


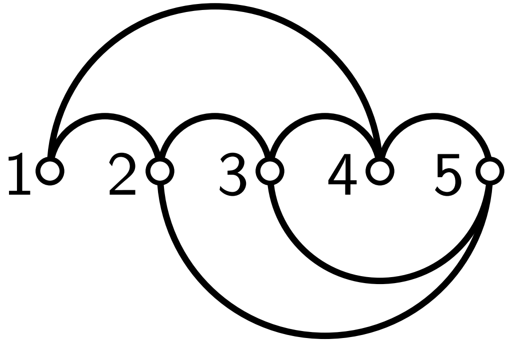


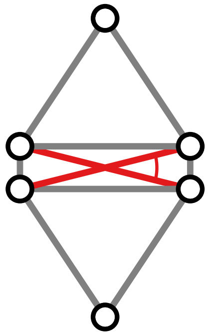
Requirements of a Graph Layout

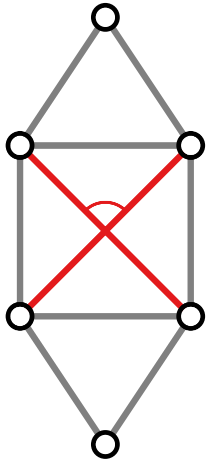
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution





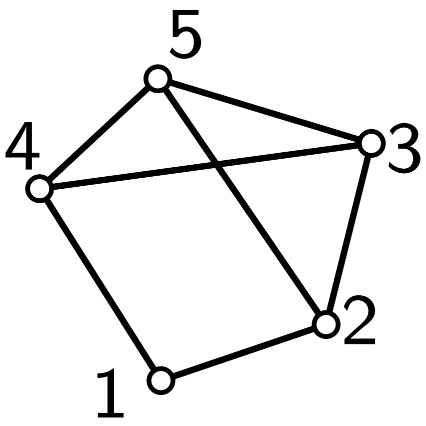


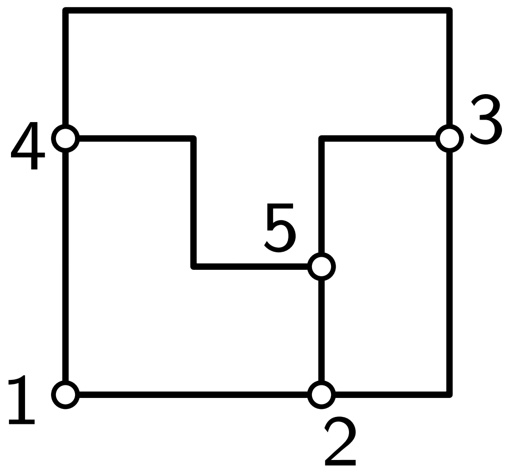


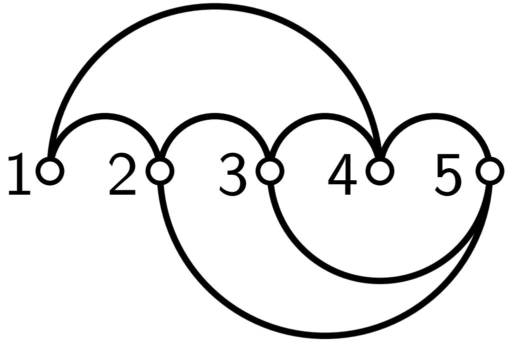


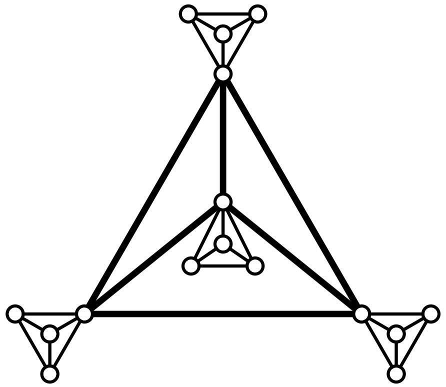
Requirements of a Graph Layout

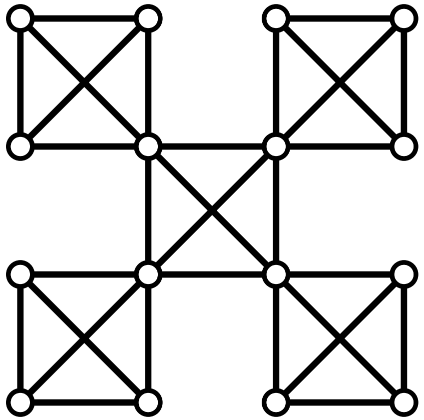
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure





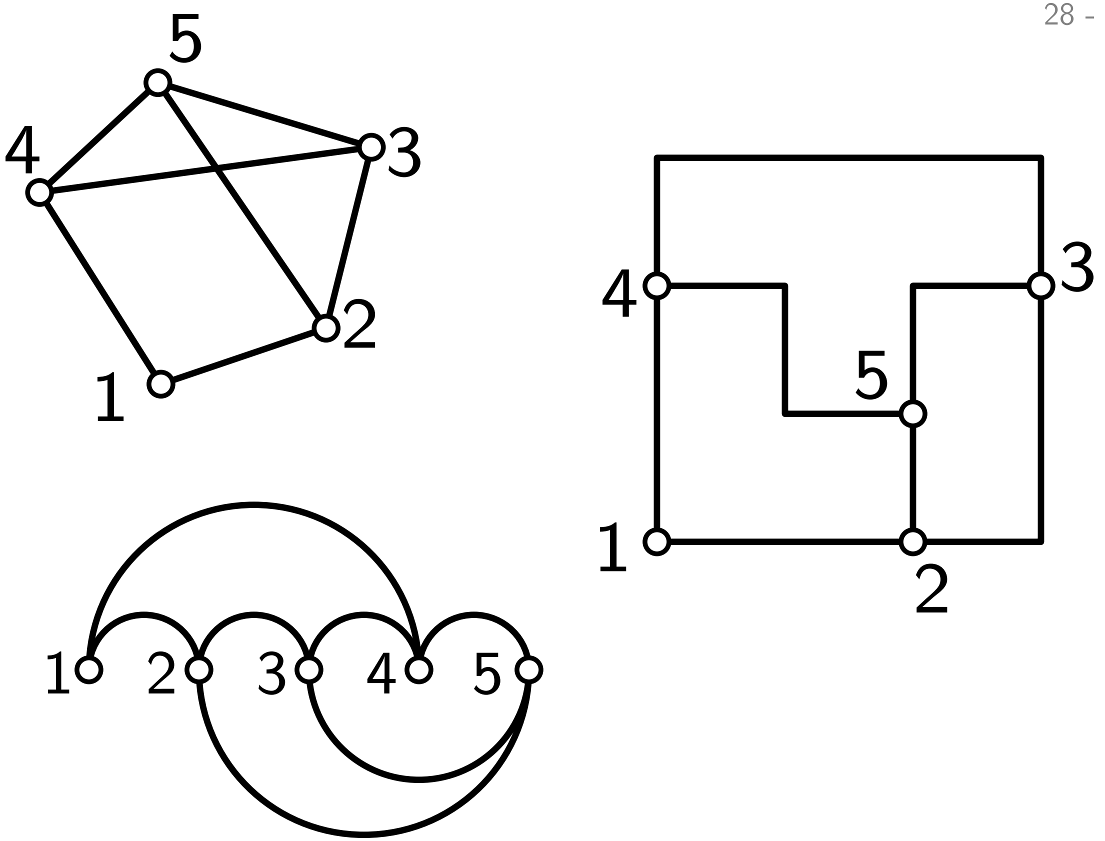






Requirements of a Graph Layout

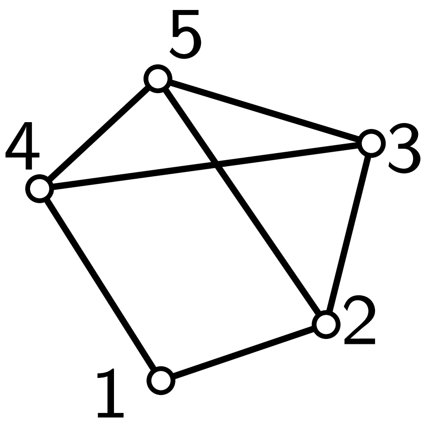
1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure

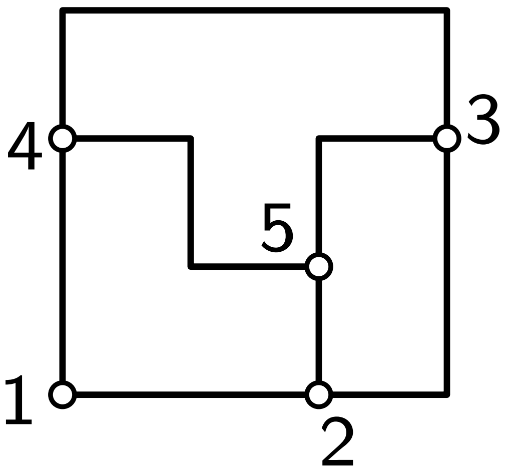


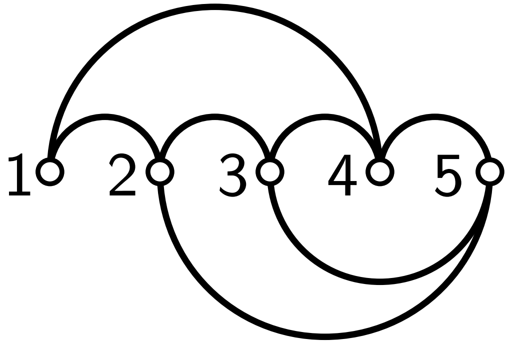
$\rightarrow$ such criteria are often inversely related

Requirements of a Graph Layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure





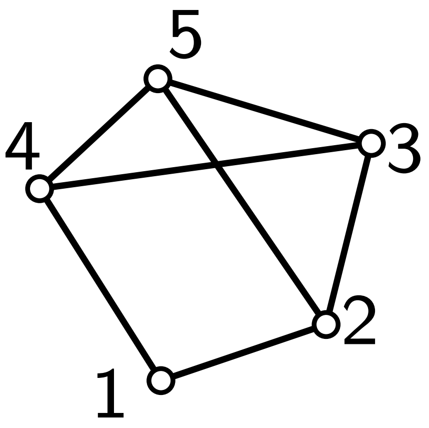


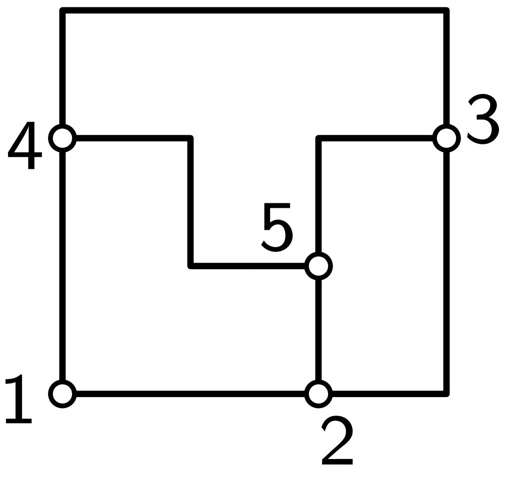
$\rightarrow$ such criteria are often inversely related

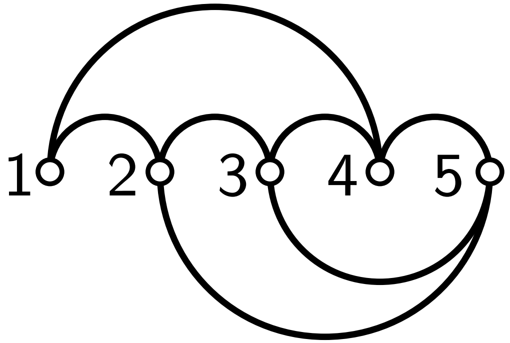
$\rightarrow$ lead to NP-hard optimization problems

Requirements of a Graph Layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure
3. Local Constraints, e.g.



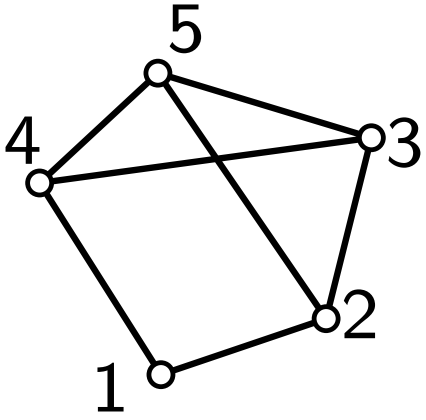


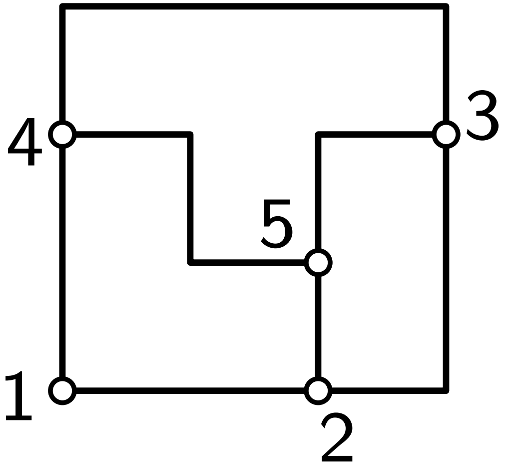


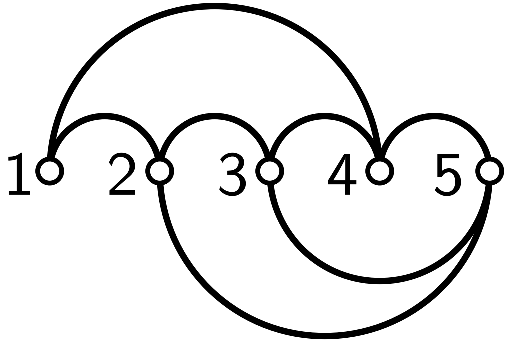
→ such criteria are often inversely related

→ lead to NP-hard optimization problems

Requirements of a Graph Layout







1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
   - orthogonal edges (with bends)
   - grid drawings
   - without crossing
2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure
3. Local Constraints, e.g.
   - restrictions on neighboring vertices (e.g., “upward”).
   - restrictions on groups of vertices/edges (e.g., “clustered”).

$\rightarrow$ such criteria are often inversely related

$\rightarrow$ lead to NP-hard optimization problems

The Layout Problem

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

The Layout Problem

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

- **drawing conventions** are met,

The Layout Problem

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

- **drawing conventions** are met,
- **aesthetic criteria** are optimized, while

The Layout Problem

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

- **drawing conventions** are met,
- **aesthetic criteria** are optimized, while
- some **additional constraints** are satisfied.

Graph Drawing Contest 2024

- We have seen that it is not always clear how a *nice* graph visualizations looks like.

Graph Drawing Contest 2024

- We have seen that it is not always clear how a *nice* graph visualizations looks like.
- Therefore, there is a graph drawing contest at the Annual International Symposium on Graph Drawing and Network Visualization (GD).

Graph Drawing Contest 2024

- We have seen that it is not always clear how a *nice* graph visualizations looks like.
- Therefore, there is a graph drawing contest at the Annual International Symposium on Graph Drawing and Network Visualization (GD).
- September 18–20, 2024, TU Wien, Vienna
  https://mozart.diei.unipg.it/gdcontest/2024/

Graph Drawing Contest 2024

- We have seen that it is not always clear how a *nice* graph visualizations looks like.
- Therefore, there is a graph drawing contest at the Annual International Symposium on Graph Drawing and Network Visualization (GD).
- September 18–20, 2024, TU Wien, Vienna
  https://mozart.diei.unipg.it/gdcontest/2024/
- Creative topic: Free-form visualization of *Olympic Games*

Graph Drawing Contest 2024

- We have seen that it is not always clear how a *nice* graph visualizations looks like.
- Therefore, there is a graph drawing contest at the Annual International Symposium on Graph Drawing and Network Visualization (GD).
- September 18–20, 2024, TU Wien, Vienna
  https://mozart.diei.unipg.it/gdcontest/2024/
- Creative topic: Free-form visualization of *Olympic Games*
- Live Challenge: *Crossing-Minimal Point-Set Embedding*
  - given: a set of points on the grid and a graph
  - task: assign the vertices to the points
  - objective function: minimize the number of crossings

Graph Drawing Contest 2024

- We have seen that it is not always clear how a *nice* graph visualizations looks like.
- Therefore, there is a graph drawing contest at the Annual International Symposium on Graph Drawing and Network Visualization (GD).
- September 18–20, 2024, TU Wien, Vienna
  https://mozart.diei.unipg.it/gdcontest/2024/
- Creative topic: Free-form visualization of *Olympic Games*
- Live Challenge: *Crossing-Minimal Point-Set Embedding*
  - given: a set of points on the grid and a graph
  - task: assign the vertices to the points
  - objective function: minimize the number of crossings
- Interested in implementing a program for the live challenge? May be done as a Praktikum!