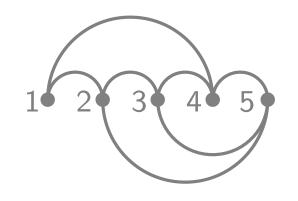


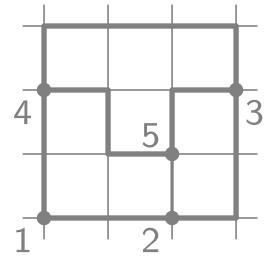
Visualization of Graphs



Lecture 0:

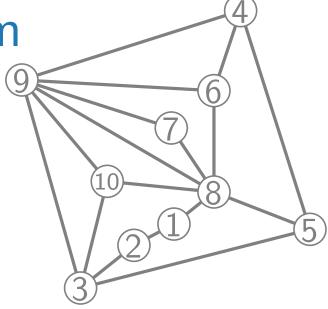
Introduction

The Graph Visualization Problem



Alexander Wolff

Summer term 2025



Lectures: Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

Tutorials:

■ Samuel Wolf (M4, room 01.005, samuel.wolf@uni-wuerzburg.de)

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

- Samuel Wolf (M4, room 01.005, samuel.wolf@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 30)
- one exercise sheet each week (Friday to Friday; first sheet appears today)

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

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

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

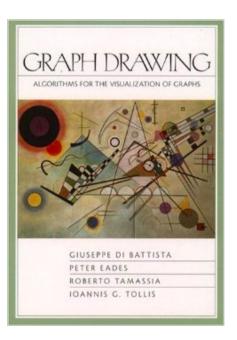
- Samuel Wolf (M4, room 01.005, samuel.wolf@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 30)
- one exercise sheet each week (Friday to Friday; first sheet appears today)
- 20 points per sheet
- \blacksquare average score 50% or more \Rightarrow bonus of 0.3 grade points
-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

Lectures:

- Alexander Wolff (M4, room 01.001, johannes.zink@uni-wuerzburg.de)
- Friday, 10:15–11:45, SE II
- videos (in German) from 2021 by Jonathan Klawitter available on WueCampus

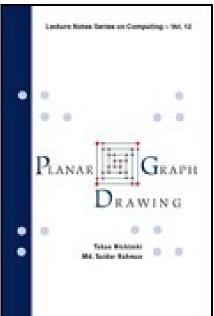
- Samuel Wolf (M4, room 01.005, samuel.wolf@uni-wuerzburg.de)
- Wednesday, 16:00–17:30, SE II (first tutorial: April 30)
- one exercise sheet each week (Friday to Friday; first sheet appears today)
- 20 points per sheet
- \blacksquare average score 50% or more \Rightarrow bonus of 0.3 grade points
-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

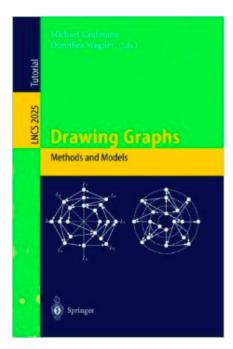


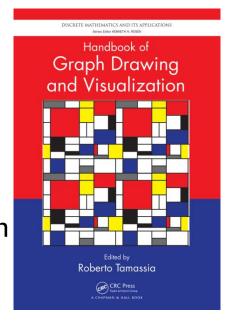
T. Nishizeki, Md. S. Rahman: Planar Graph Drawing World Scientific, 2004

R. Tamassia:

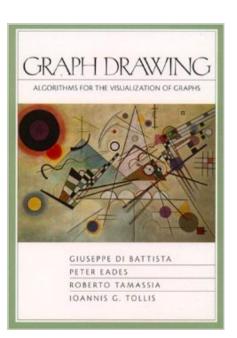
Handbook of Graph Drawing and Visualization CRC Press, 2013

http://cs.brown.edu/people/rtamassi/gdhandbook/





Books



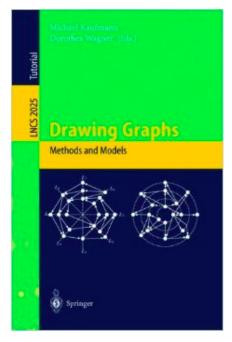
G. Di Battista, P. Eades, R. Tamassia, I. Tollis: Graph Drawing: Algorithms for the Visualization of Graphs Prentice Hall, 1998

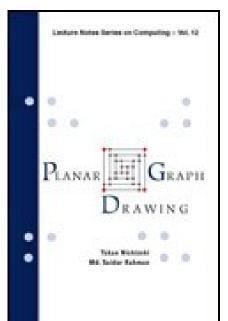
[GD]

[DG

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

Springer, 2001





T. Nishizeki, Md. S. Rahman:

Planar Graph Drawing World Scientific, 2004

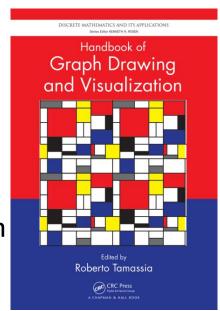
[PGD]

HGDV]

R. Tamassia:

Handbook of Graph Drawing and Visualization CRC Press, 2013

http://cs.brown.edu/people/rtamassi/gdhandbook/



Learning objectives

Learning objectives

Overview of graph visualization

Learning objectives

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

Learning objectives

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

Visualization problem:

 \blacksquare Given a graph G, visualize it with a drawing Γ

Learning objectives

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

Visualization problem:

 \blacksquare Given a graph G, visualize it with a drawing Γ

Here:

Reducing the visualization problem to its algorithmic core

Learning objectives

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

Visualization problem:

 \blacksquare 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

Learning objectives

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

Visualization problem:

 \blacksquare 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

Learning objectives

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

Visualization problem:

 \blacksquare 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

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

Learning objectives

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

Visualization problem:

 \blacksquare 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

divide & conquer, incremental

proofs

- data structures
- combinatorial optimization (flows, ILPs)
- force-based algorithm

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

What is a graph?

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

What is a graph?

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

Representation?

What is a graph?

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

Representation?

Set notation

```
\begin{split} 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}\}\} \end{split}
```

What is a graph?

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

Representation?

Set notation

```
\begin{split} 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}\}\} \end{split}
```

Adjacency list

What is a graph?

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

Representation?

Set notation

```
\begin{split} 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}\}\} \end{split}
```

Adjacency list

Adjacency matrix

What is a graph?

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

Representation?

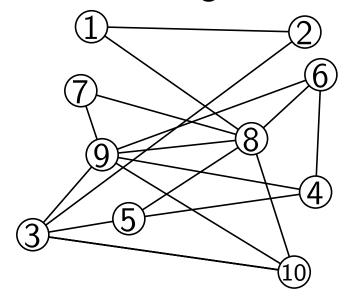
Set notation

```
\begin{split} 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}\}\} \end{split}
```

Adjacency list

Adjacency matrix

Drawing



What is a graph?

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

Representation?

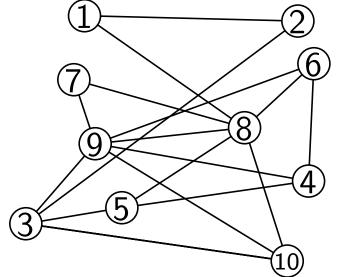
Set notation

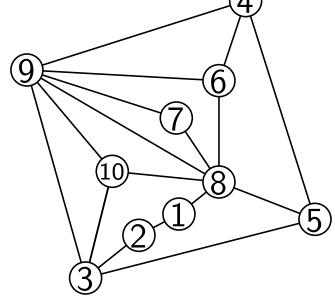
```
V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}
E(G) = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \{v_9, v_{10}\}\}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

Adjacency list

Adjacency matrix

Drawing





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

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

Physical networks

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

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)
- ...

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

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

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.

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.

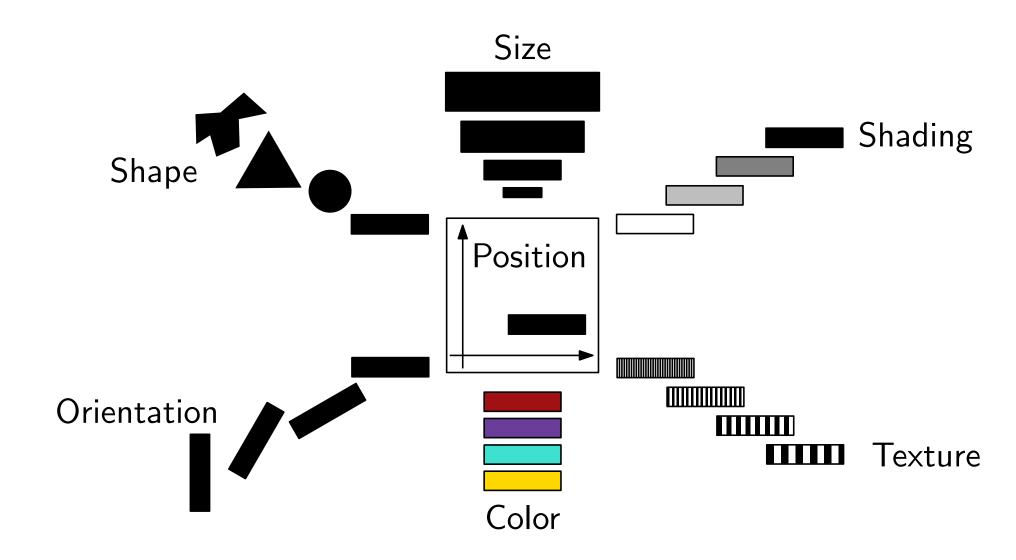
Graphs are a mathematical representation of real physical and abstract networks.

- 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.

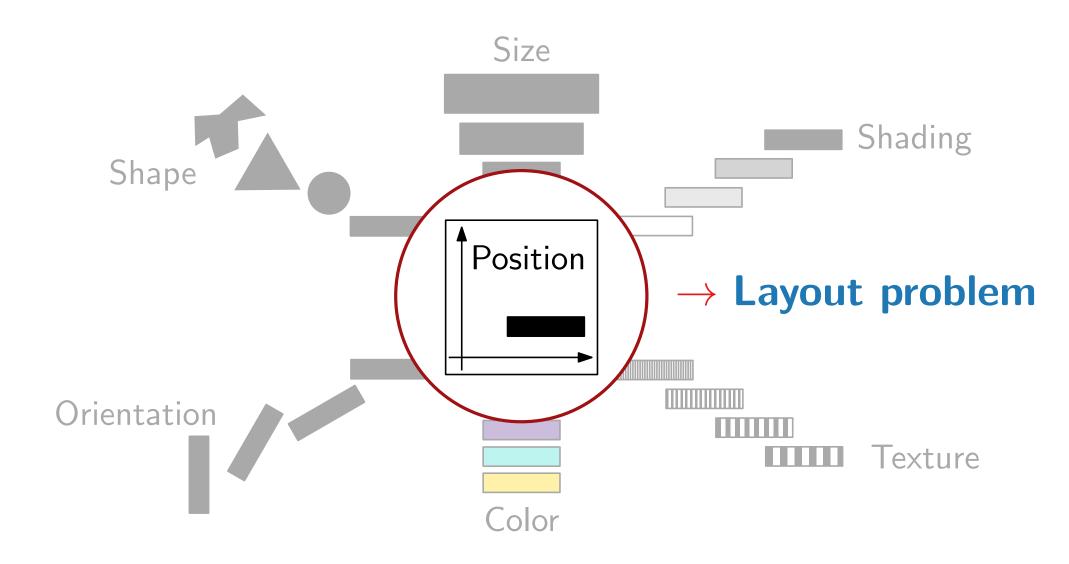
We need algorithms that draw graphs automatically to make networks more accessible to humans.

Jacques Bertin defined *visualization variables* (1967)

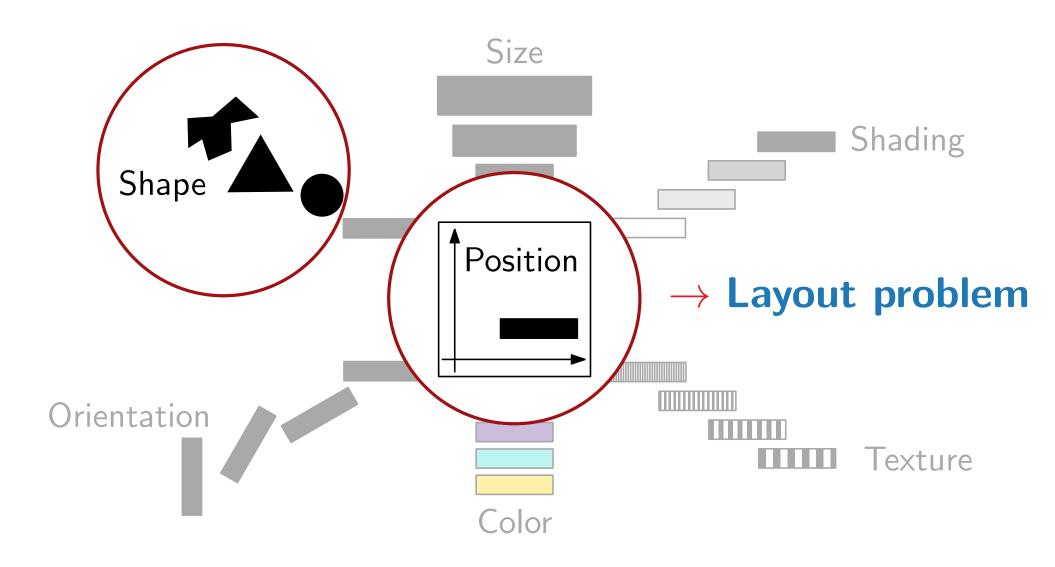
■ Jacques Bertin defined *visualization variables* (1967)



Jacques Bertin defined *visualization variables* (1967)

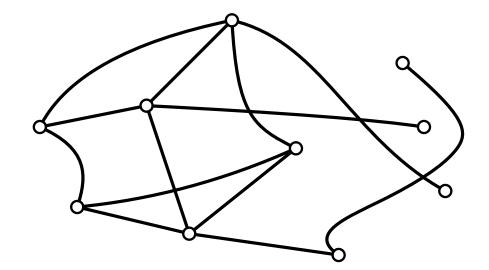


■ Jacques Bertin defined *visualization variables* (1967)



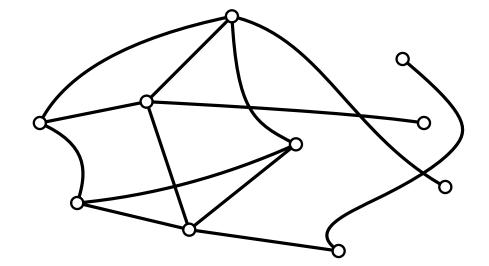
The Layout Problem

■ *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.



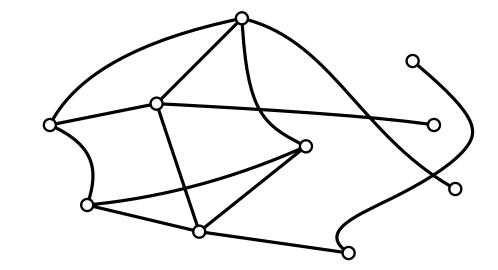
Graph Visualization Problem

in: graph G

out:

The Layout Problem

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



Graph Visualization Problem

in: graph G

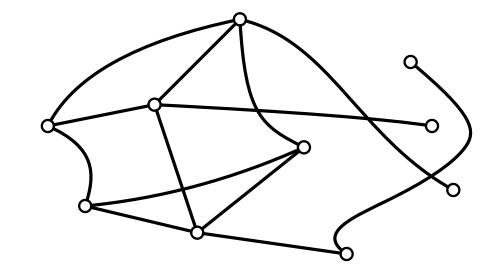
out: nice drawing Γ of G

 $\Gamma \colon V(G) \to \mathbb{R}^2$, vertex $v \mapsto \text{point } \Gamma(v)$

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

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

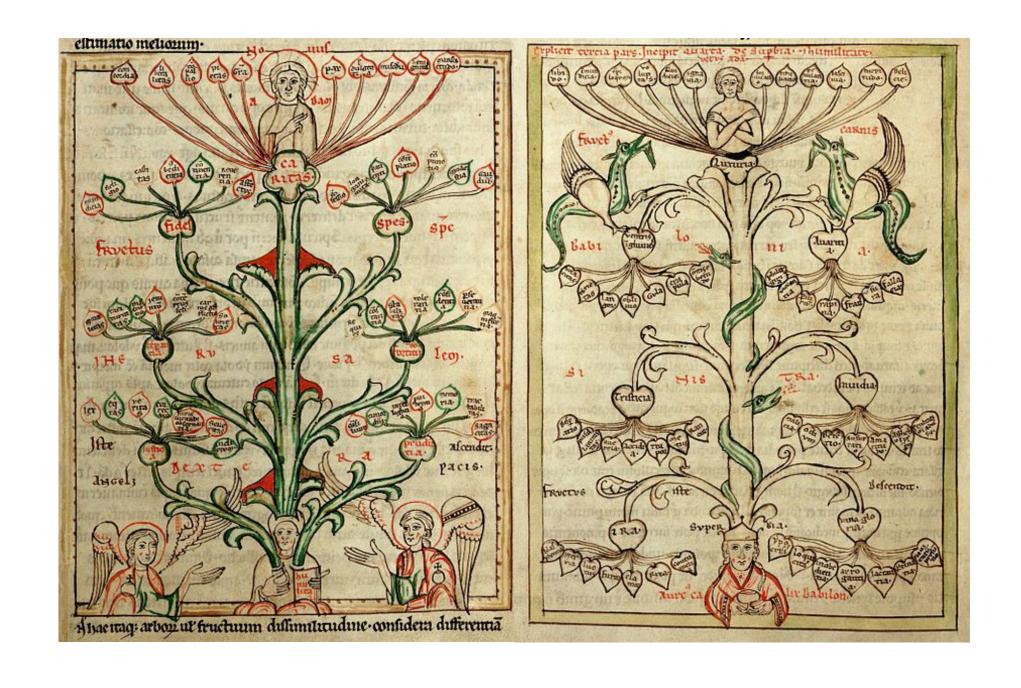
 $\Gamma \colon V(G) \to \mathbb{R}^2$, vertex $v \mapsto \text{point } \Gamma(v)$

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

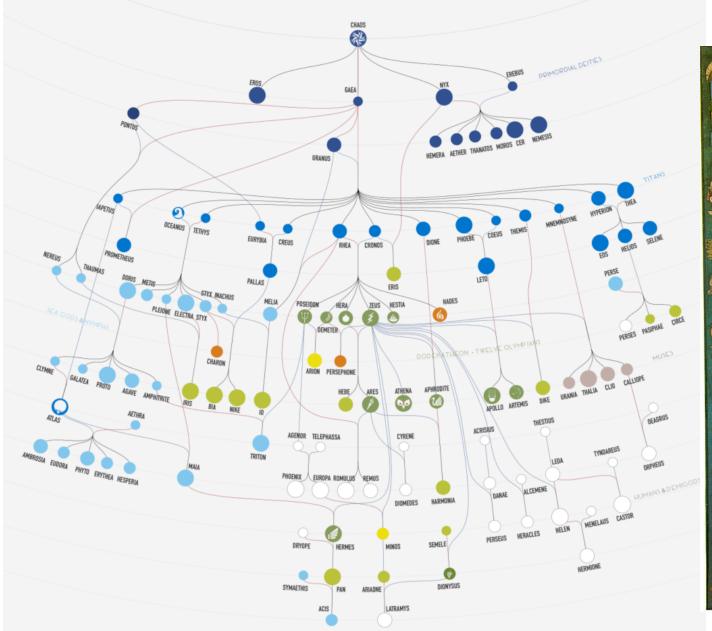
But what is a **nice** drawing?

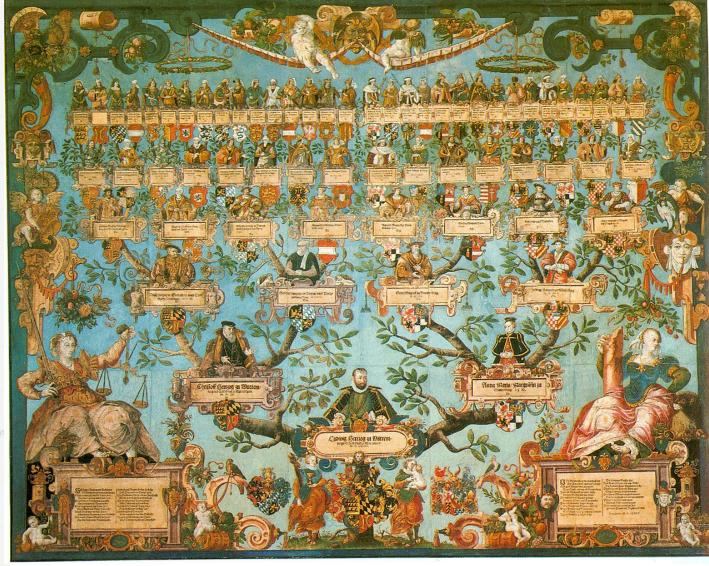
Tree of Virtues and Tree of Vice

ca. 1200



Social Networks – Family Trees

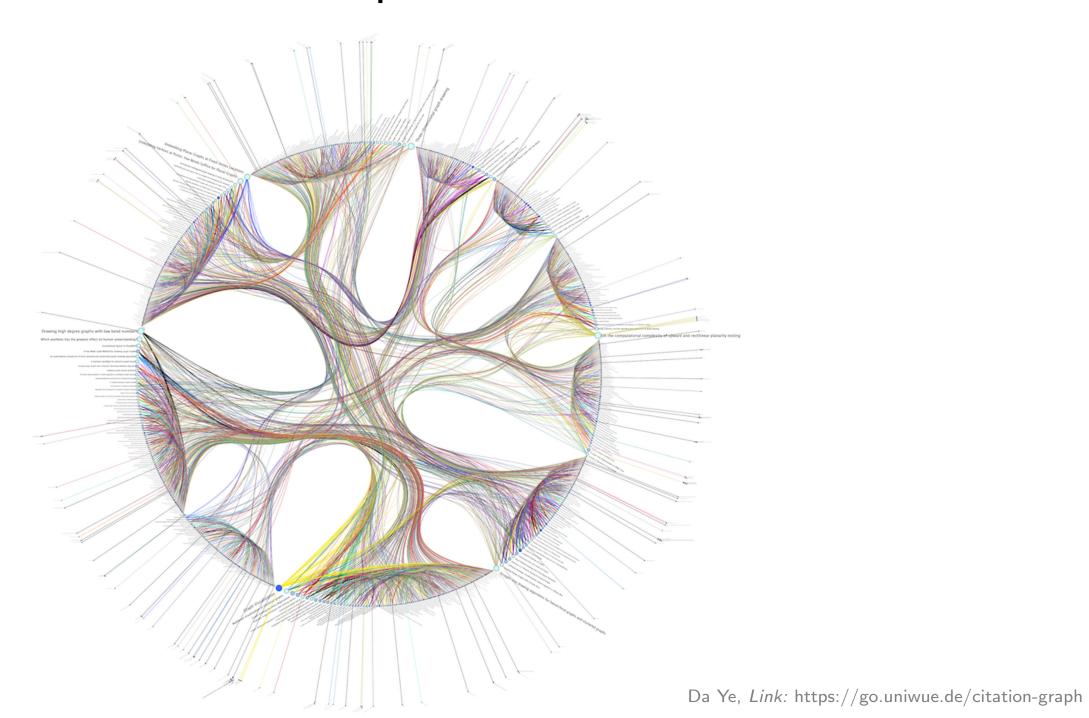




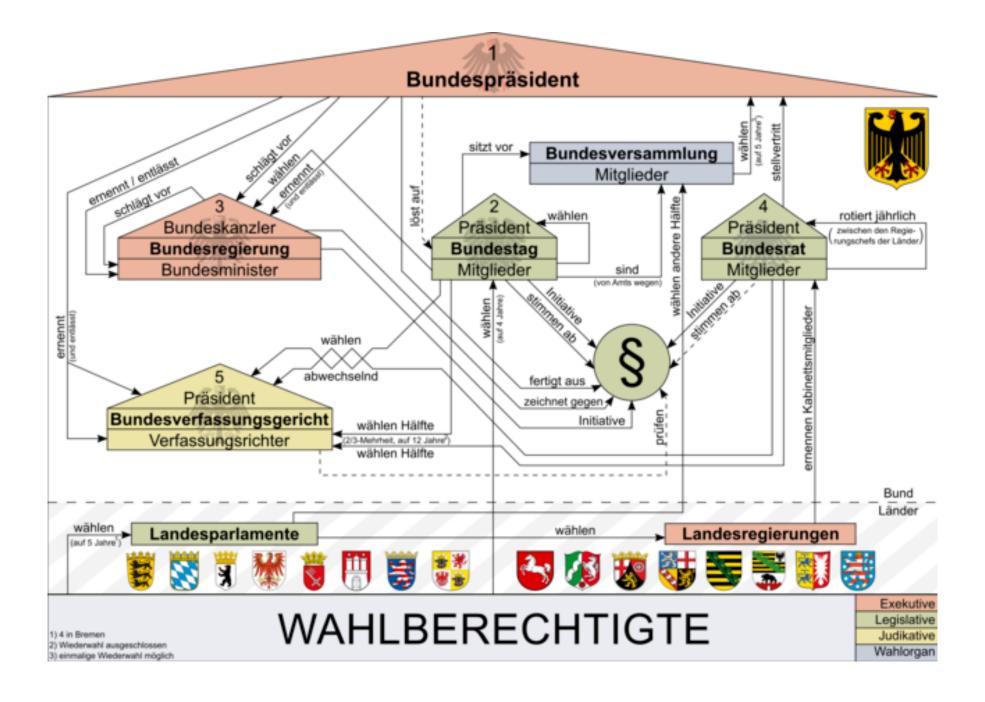
J. Klawitter, T. Mchedlidze, *Link:* go.uniwue.de/myth-poster

Ahnentafel Herzog Ludwig von Württemberg, 1585

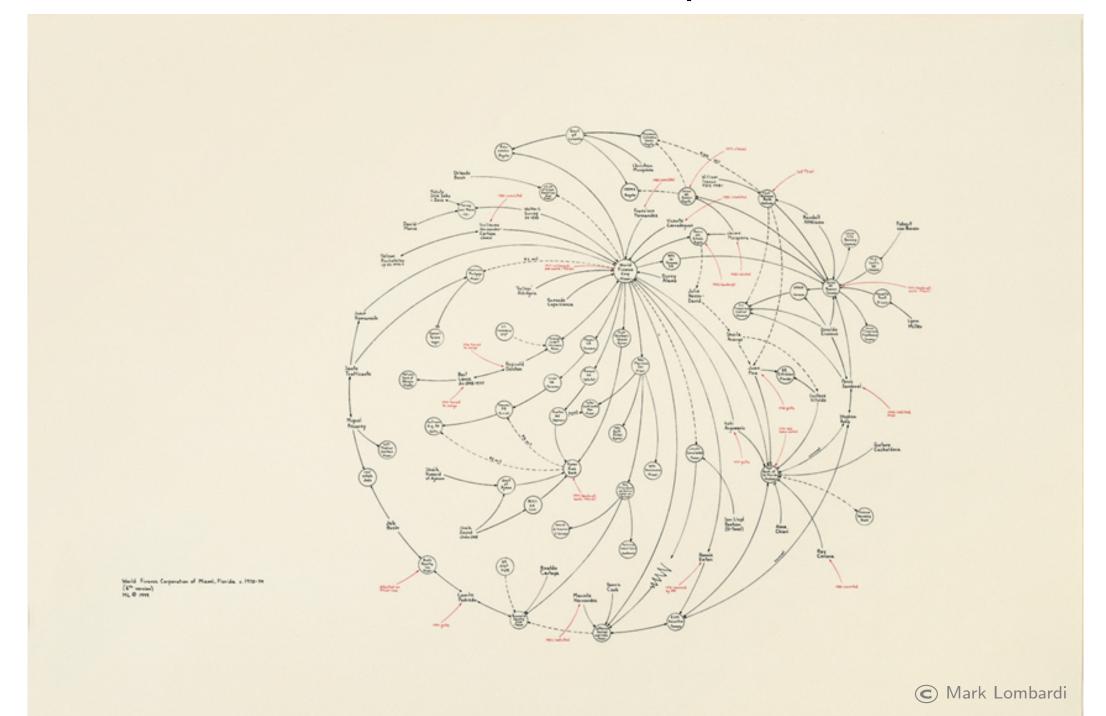
Social Network – Citation Graph



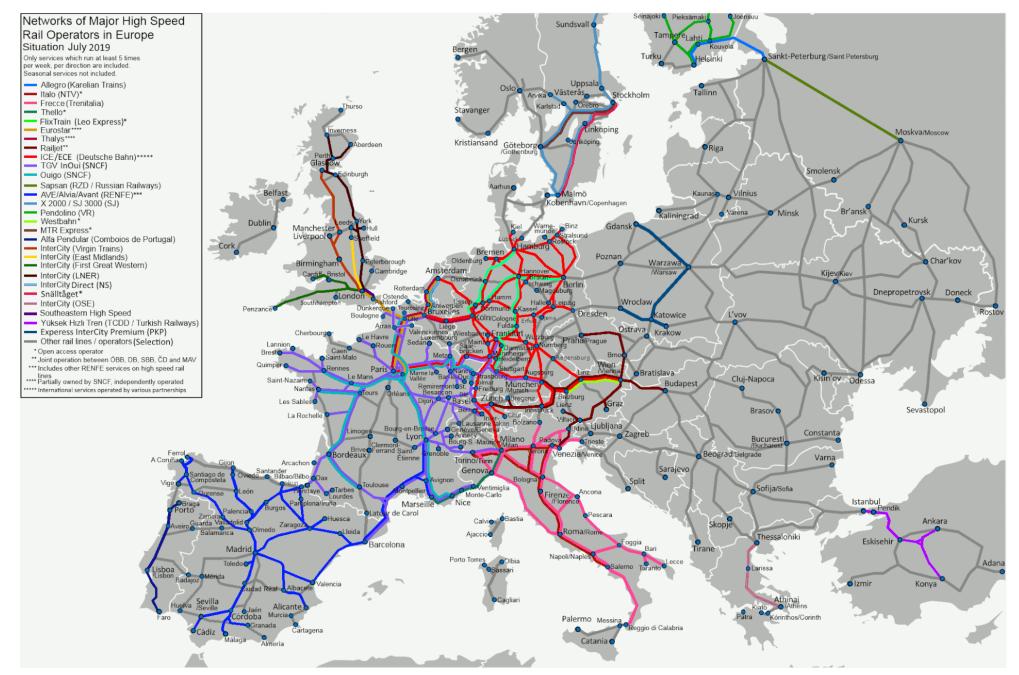
Social Network – Organizational Chart



Social Network – World Finance Corporation



Transportation Network – European High-Speed Railroads



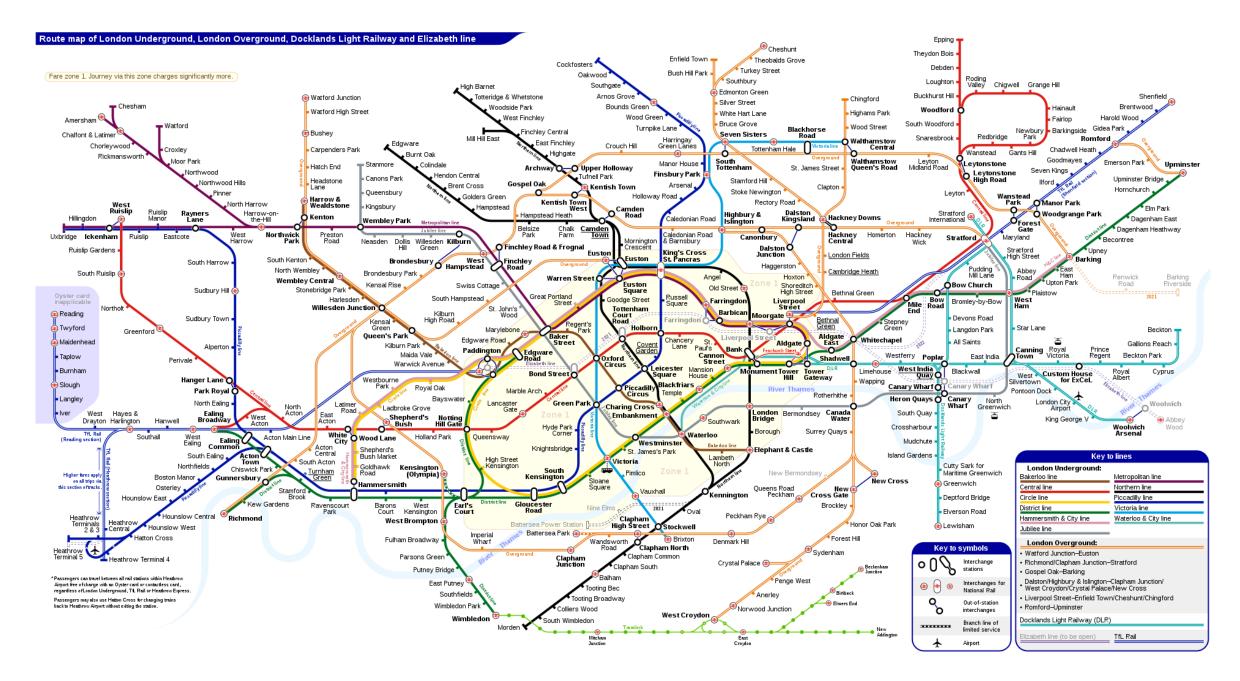
Source: Wiki Commons: Networks of Major High Speed Rail Operators in Europe - CC BY-SA 3.0

Transportation Network – London Underground

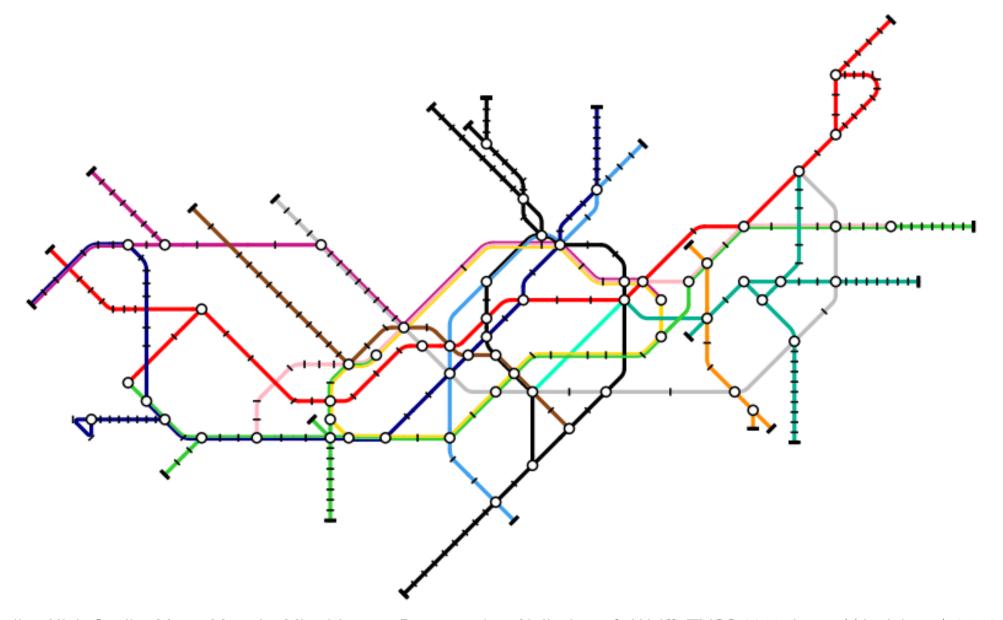


Source: Wiki Commons: London Underground full map - CC BY-SA 3.0

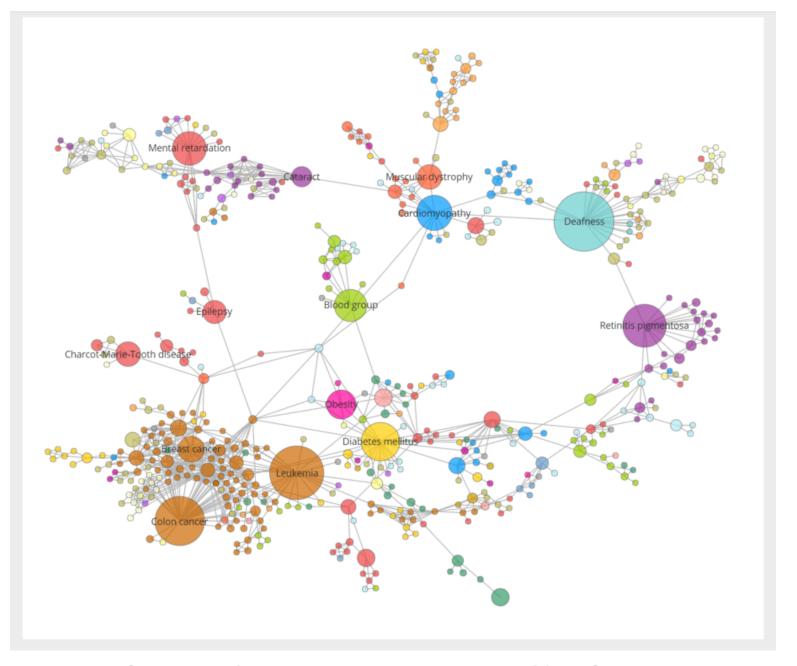
Transportation Network – London Underground



Transportation Network – London Underground

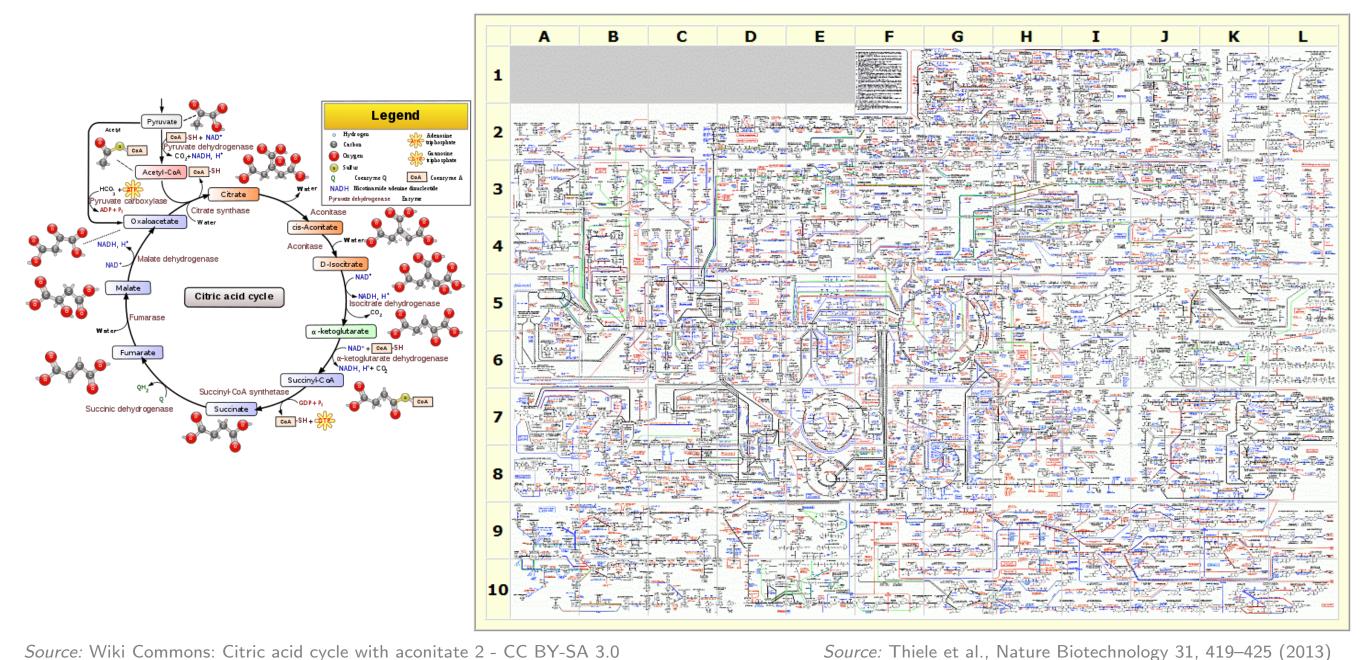


Bioinformatics – Disease Interaction

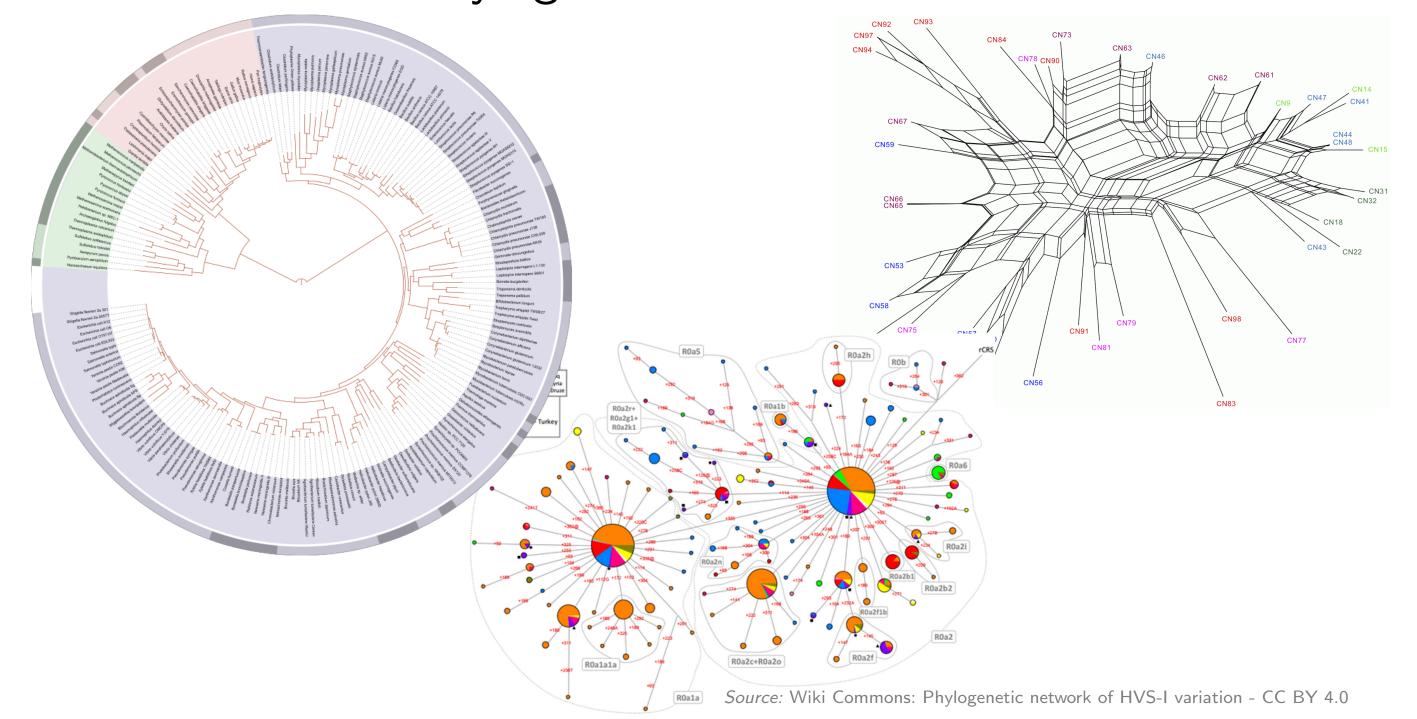


Source: Wiki Commons: Human disease network - CC BY-SA 4.0

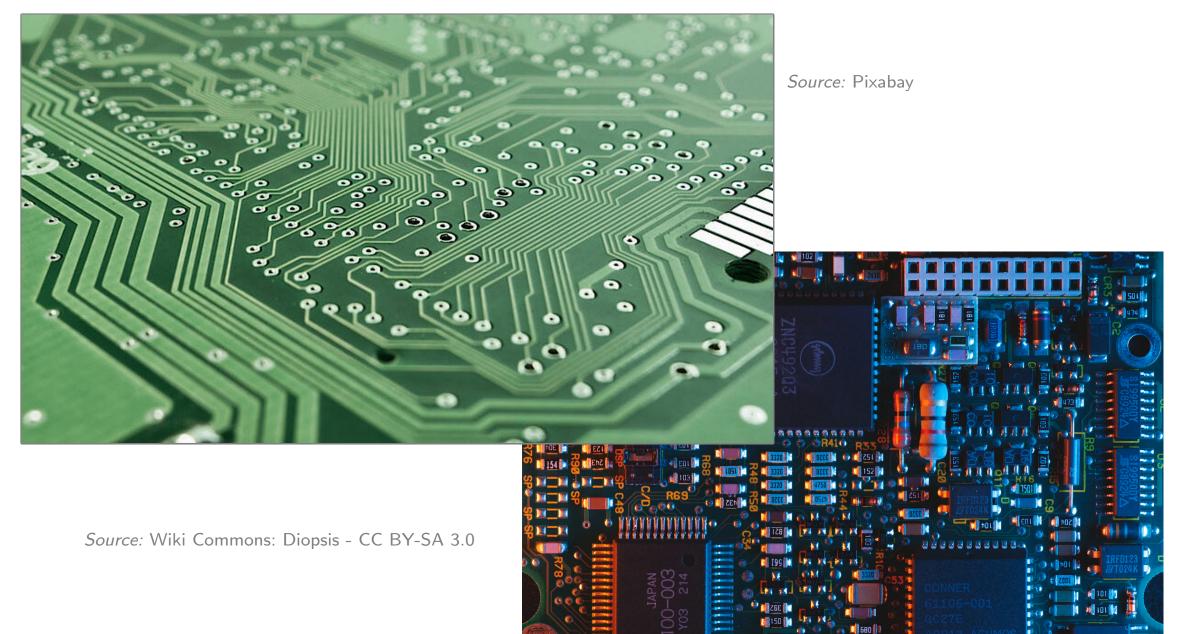
Bioinformatics – Molecular Metabolic Network



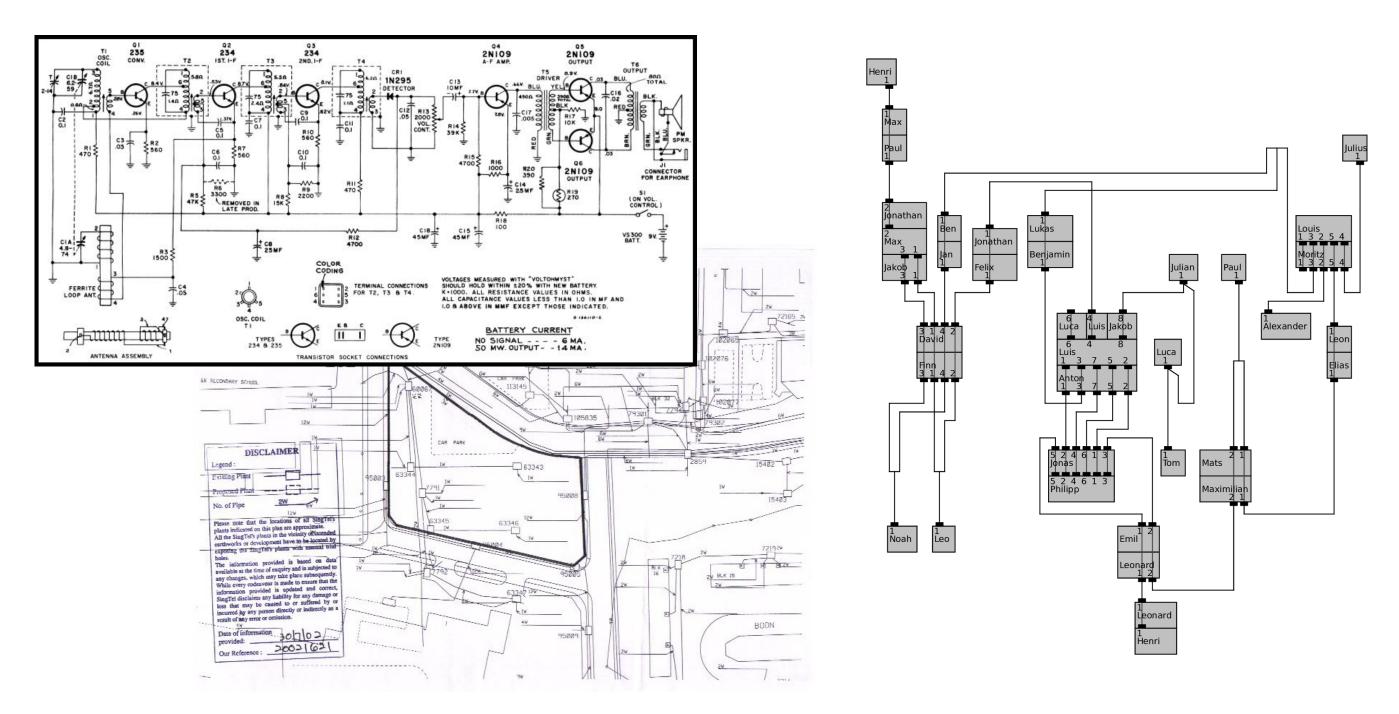
Bioinformatics – Phylogenetic Trees & Networks



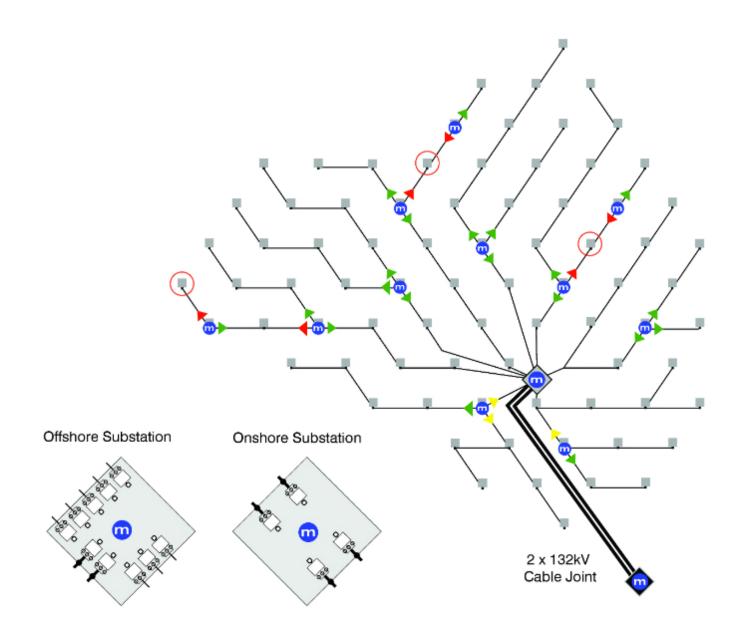
Technical Network – Very Large-Scale Integration (VLSI)



Technical Network – Transistor Diagram, Wiring Diagram



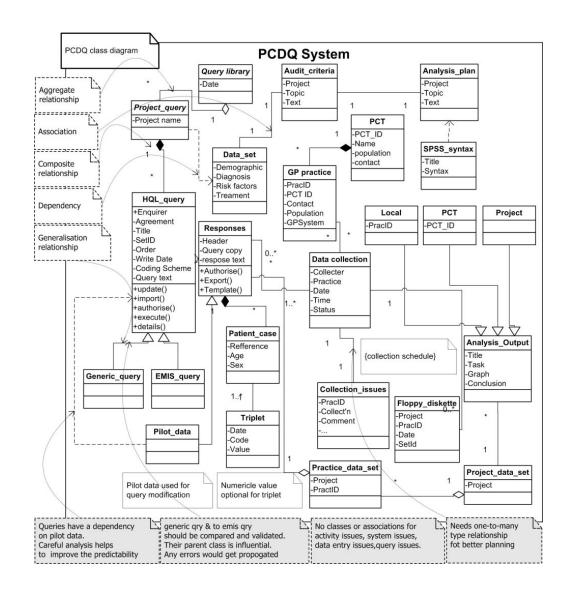
Technical Networks – Offshore Wind Farms

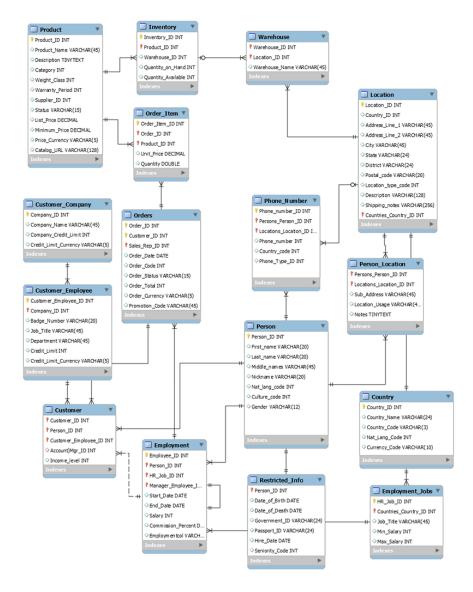




Source: Wiki Commons: Alpha Ventus Windmills - CC BY-SA 3.0

Technical Network – UML Diagram

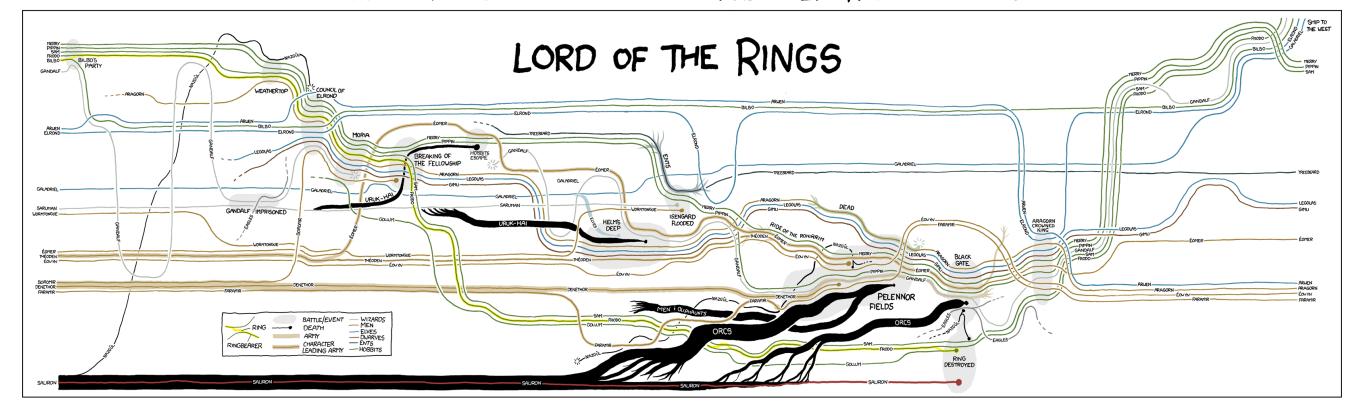




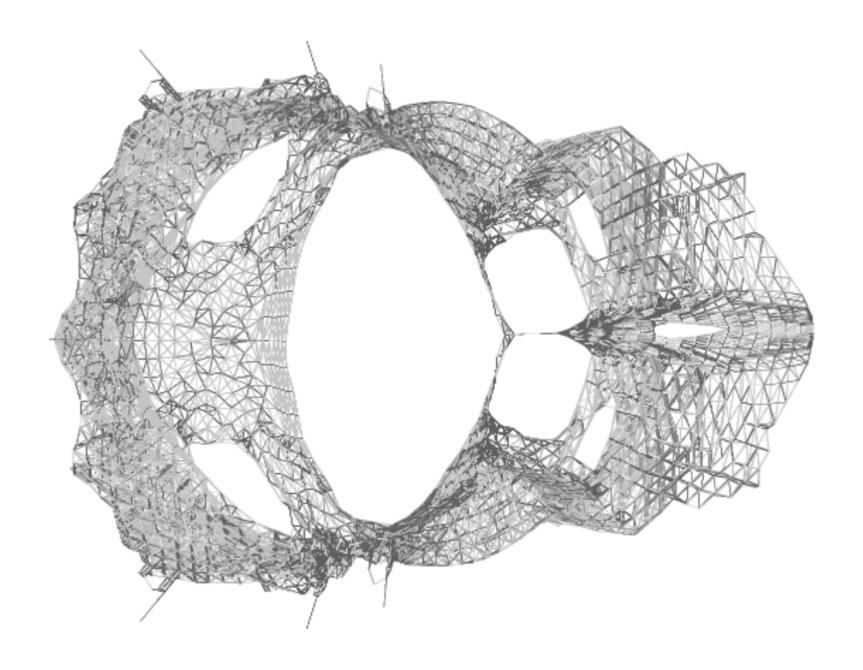


Temporal Graph Layout – Storylines

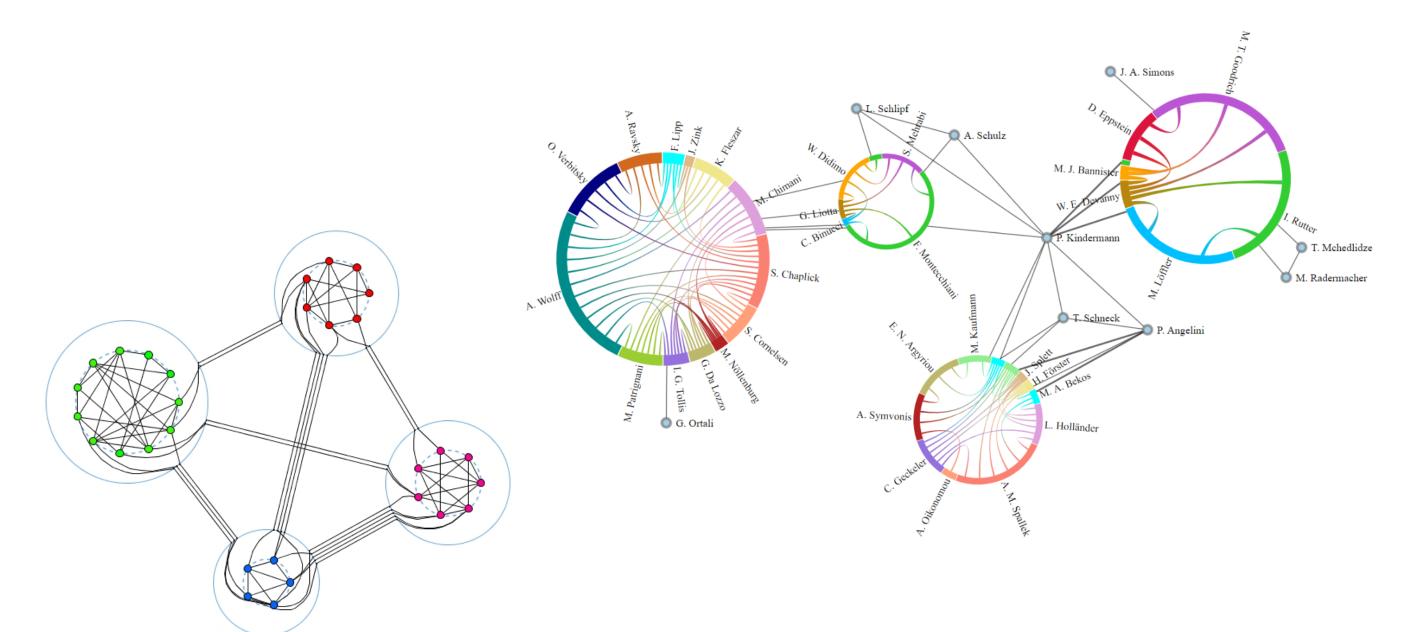
THESE CHARTS SHOW MOVIE CHARACTER INTERACTIONS.
THE HORIZONTAL AXIS IS TIME. THE VERTICAL GROUPING OF THE
LINES INDICATES WHICH CHARACTERS ARE TOGETHER AT A GIVEN TIME.



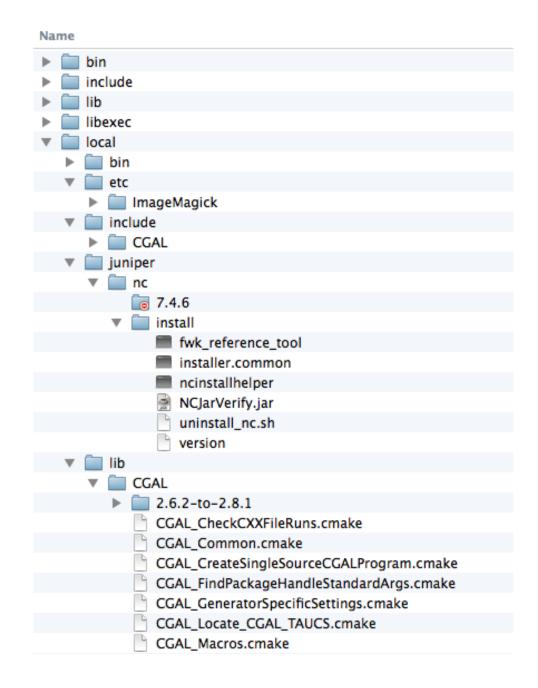
Large Graphs – Object Mesh

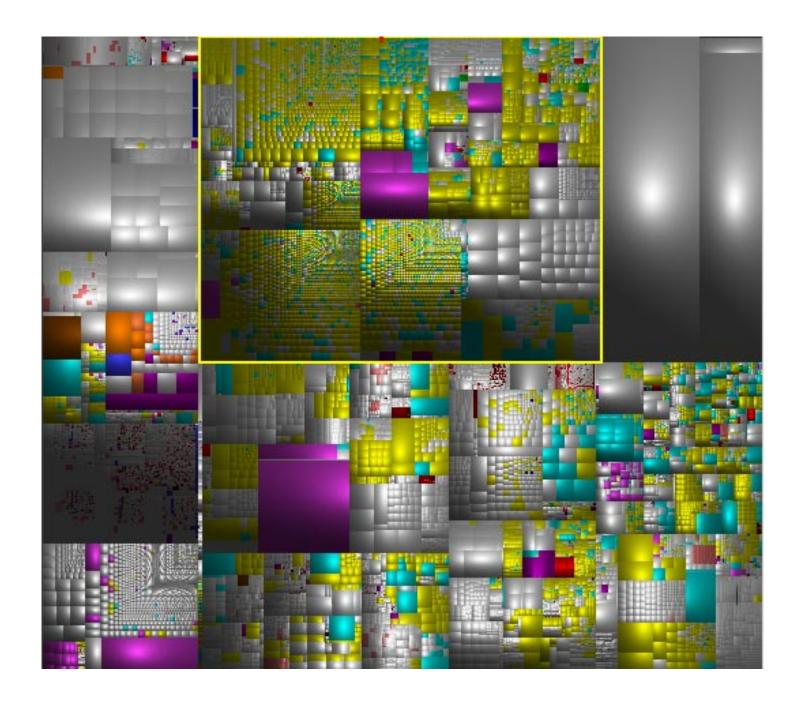


General Graphs – Micro–Macro Layout

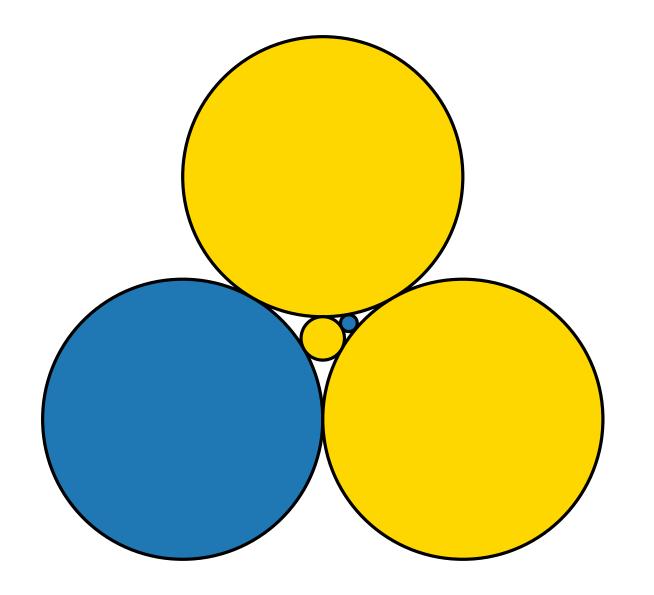


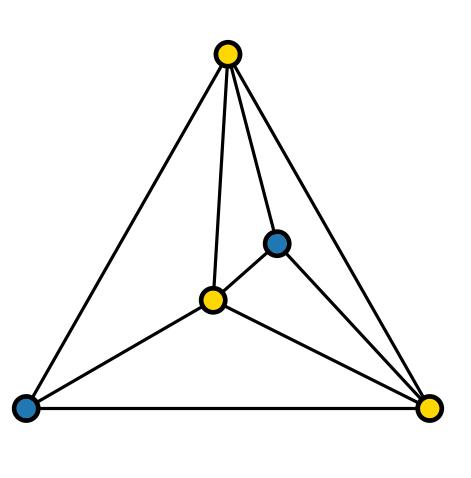
Alternative Representations – Treemap



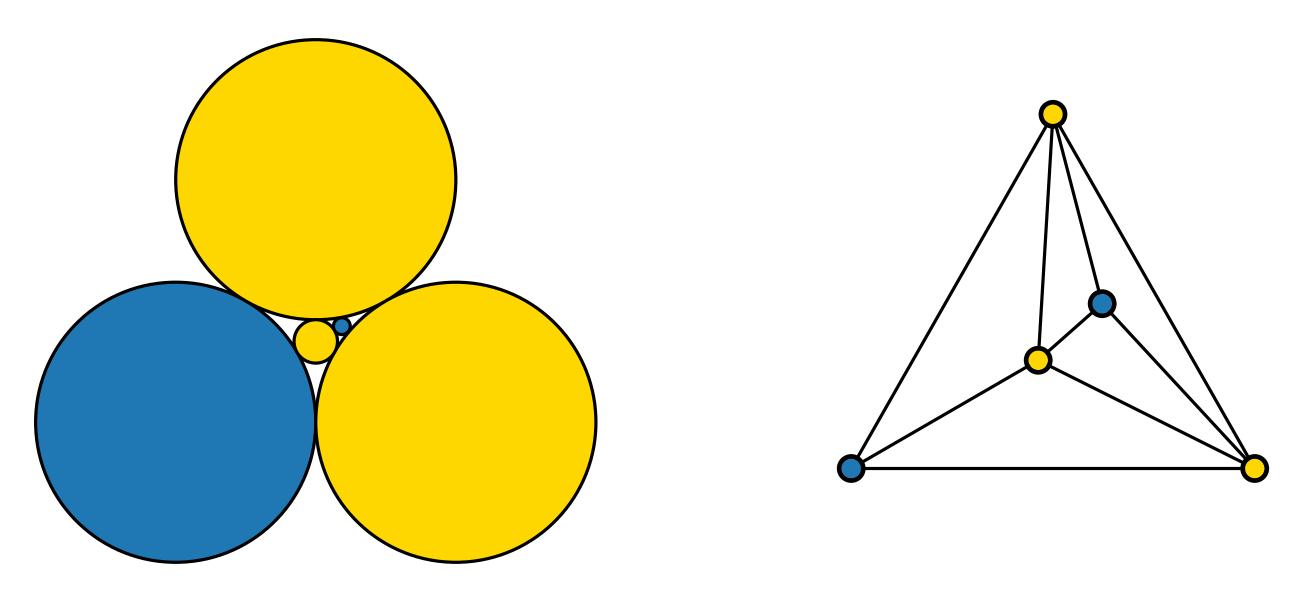


Alternative Representations – Contact Graphs





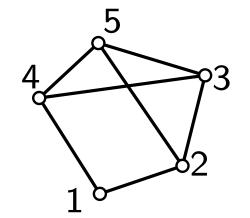
Alternative Representations – Contact Graphs



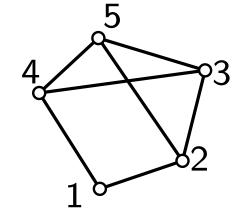
■ For more examples see visualcomplexity.com

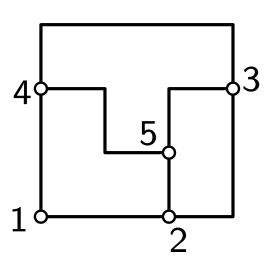
1. Drawing conventions and requirements, e.g.,

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

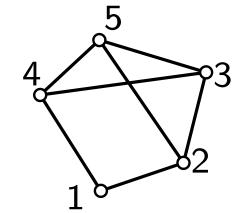


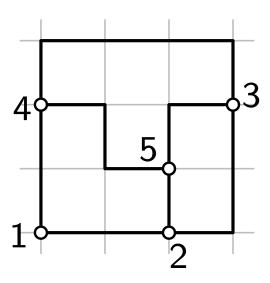
- 1. Drawing conventions and requirements, e.g.,
- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)





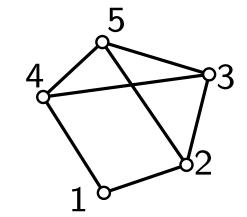
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
 - orthogonal edges (with bends)
 - grid drawings

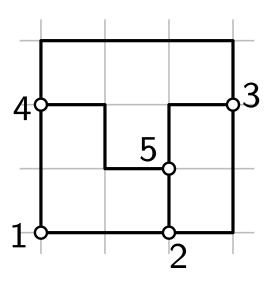




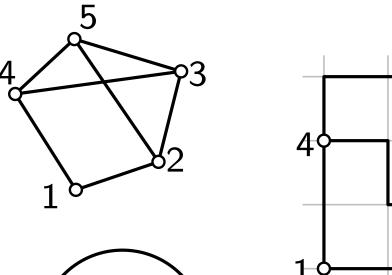
1. Drawing conventions and requirements, e.g.,

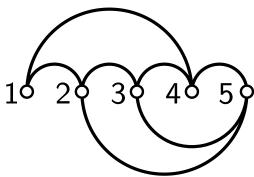
- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)
- grid drawings
- without crossing

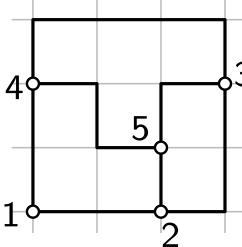




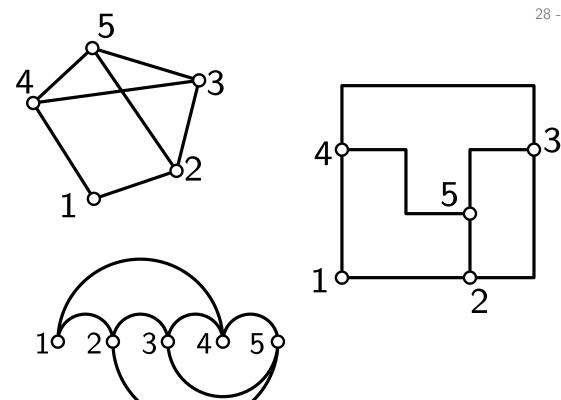
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing



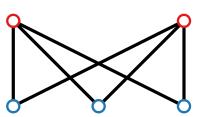


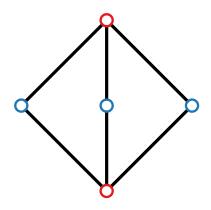


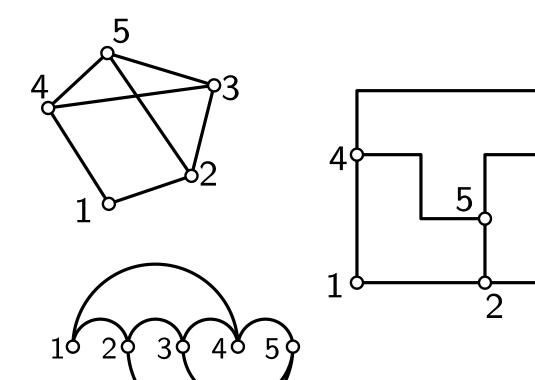
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing
- 2. Aesthetics to be optimized, e.g.



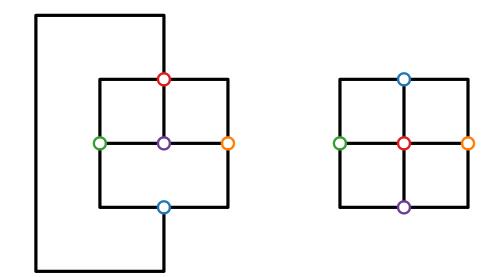
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing
- 2. Aesthetics to be optimized, e.g.
 - crossing/bend minimization

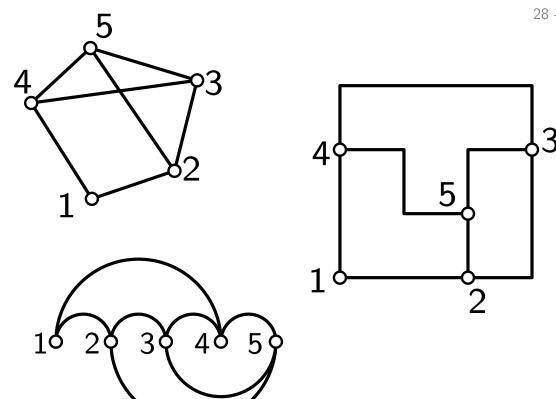




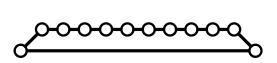


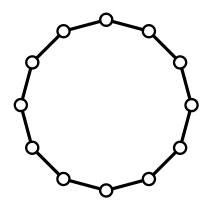
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing
- 2. Aesthetics to be optimized, e.g.
 - crossing/bend minimization

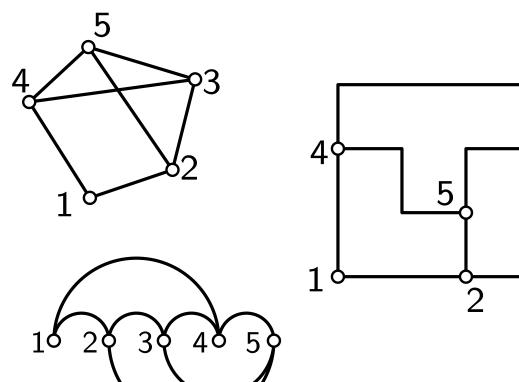




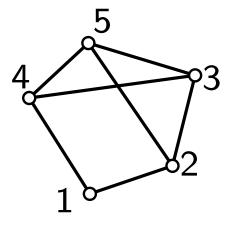
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing
- 2. Aesthetics to be optimized, e.g.
- crossing/bend minimization
- edge length uniformity

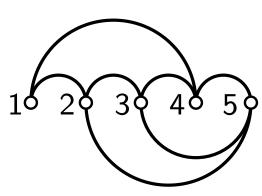


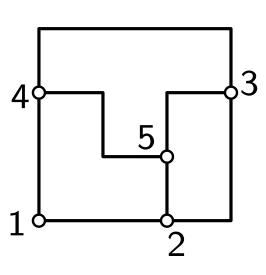


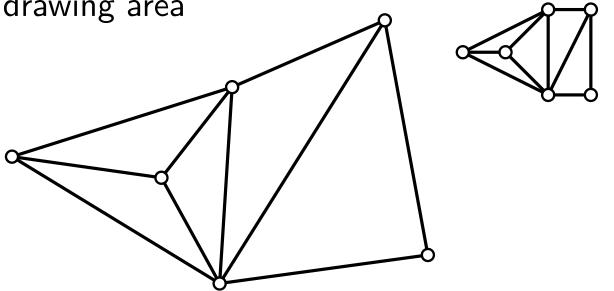


- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing
- 2. Aesthetics to be optimized, e.g.
 - crossing/bend minimization
 - edge length uniformity
 - minimizing total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









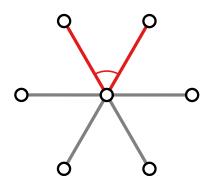
1. Drawing conventions and requirements, e.g.,

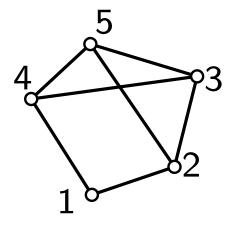
- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)
- grid drawings
- without crossing

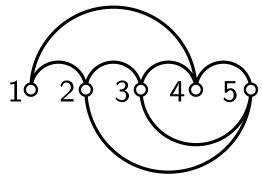
2. Aesthetics to be optimized, e.g.

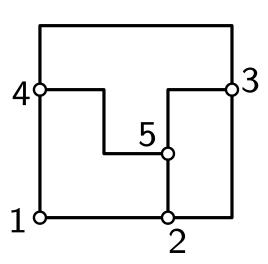
- crossing/bend minimization
- edge length uniformity
- minimizing total edge length/drawing area
- angular resolution









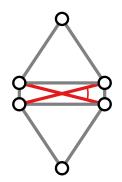


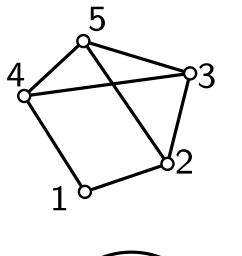
1. Drawing conventions and requirements, e.g.,

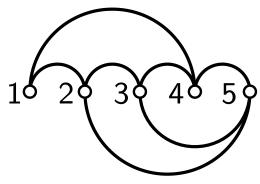
- straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
- orthogonal edges (with bends)
- grid drawings
- without crossing

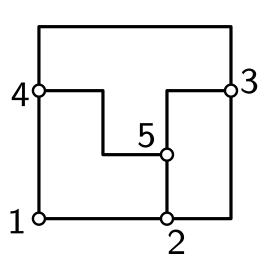
2. Aesthetics to be optimized, e.g.

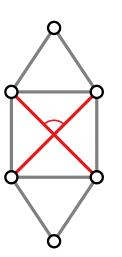
- crossing/bend minimization
- edge length uniformity
- minimizing total edge length/drawing area
- angular resolution



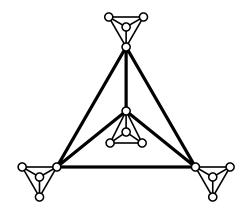


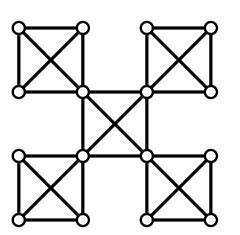


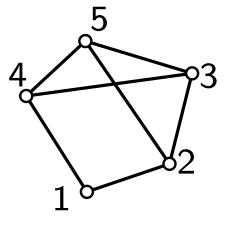


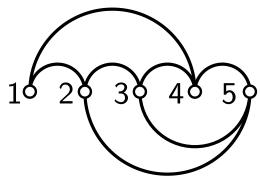


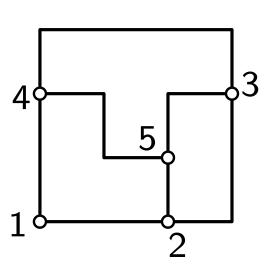
- 1. Drawing conventions and requirements, e.g.,
 - straight edges with $\Gamma(uv) = \overline{\Gamma(u)\Gamma(v)}$
 - orthogonal edges (with bends)
 - grid drawings
 - without crossing
- 2. Aesthetics to be optimized, e.g.
 - crossing/bend minimization
 - edge length uniformity
 - minimizing total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









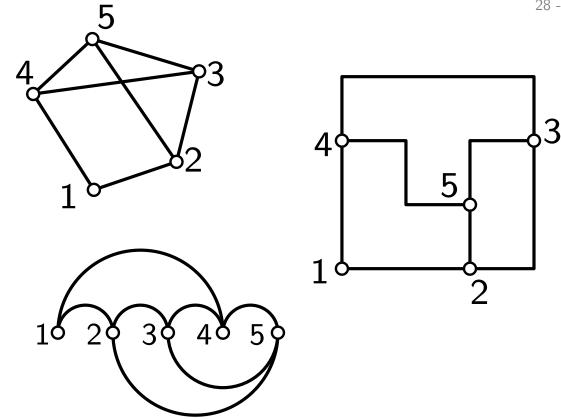


1. Drawing conventions and requirements, e.g.,

- straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
- 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



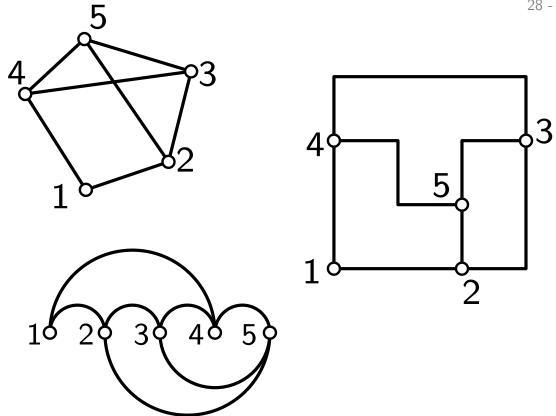
 \rightarrow such criteria are often inversely related

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

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

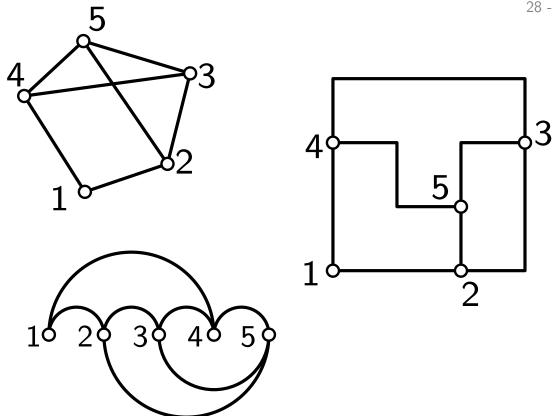
2. Aesthetics to be optimized, e.g.

- 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

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



- \rightarrow such criteria are often inversely related
- \rightarrow lead to NP-hard optimization problems

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

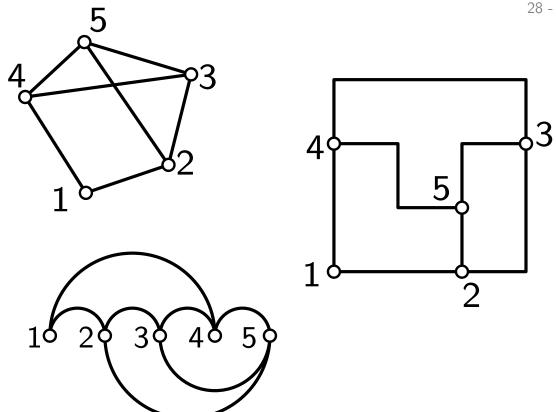
- straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
- orthogonal edges (with bends)
- grid drawings
- without crossing

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").



- → such criteria are often inversely related
- \rightarrow lead to NP-hard optimization problems

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

drawing conventions are met,

Graph Visualization Problem (more general)

in: Graph G

out: Drawing Γ of G such that

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

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.

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

- 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).

- 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).
- GD 2025: Sep. 24–26, 2025, Norrköping, Sweden https://mozart.diei.unipg.it/gdcontest/2025/

- 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).
- GD 2025: Sep. 24–26, 2025, Norrköping, Sweden https://mozart.diei.unipg.it/gdcontest/2025/
- \blacksquare Creative topic: 360° visualization of the relational data about the netflix series *Dark*.

- 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).
- GD 2025: Sep. 24–26, 2025, Norrköping, Sweden https://mozart.diei.unipg.it/gdcontest/2025/
- \blacksquare Creative topic: 360° visualization of the relational data about the netflix series *Dark*.
- Live Challenge: *Minimizing the local crossing number*:
 - \blacksquare Given: a graph G,
 - \blacksquare task: assign the vertices of G to grid points of a square grid of restricted size,
 - objective: minimize (over all straight-line grid drawings) the maximum number of crossings over all edges of G.

- 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).
- GD 2025: Sep. 24–26, 2025, Norrköping, Sweden https://mozart.diei.unipg.it/gdcontest/2025/
- \blacksquare Creative topic: 360° visualization of the relational data about the netflix series *Dark*.
- Live Challenge: *Minimizing the local crossing number*:
 - \blacksquare Given: a graph G,
 - \blacksquare task: assign the vertices of G to grid points of a square grid of restricted size,
 - objective: minimize (over all straight-line grid drawings) the maximum number of crossings over all edges of G.
- Interested in implementing a program for the live challenge? May be done as a Praktikum!