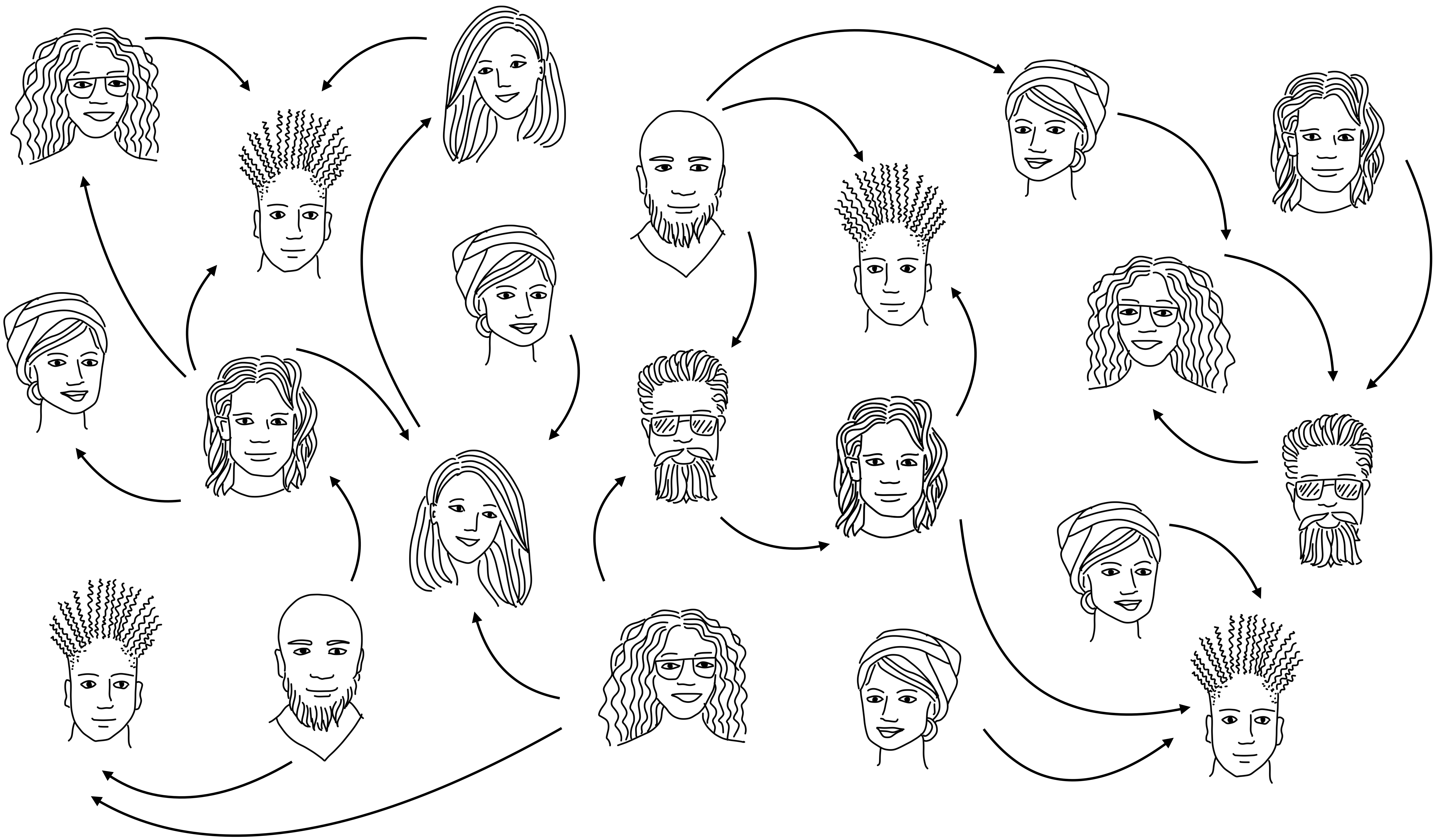
VISUALIZING
MULTIVARIATE NETWORKS

Carolina Nobre


foursquare
LiVEJOURNAL
vimeo
YouTube
Insta
facebook
Blogger
my
twitter
flickr
tumblr.
Like

Glyoxy-
late
cycle
S-CoA
LASE
H₂O
CoA-SH
OXALATE
DECARBOXY-
LASE
CO₂
OXALYL-CoA
DECARBOXY-
LASE
OXALATE-CoA-TRANS
FERRE-
DIOXIN
(red)
FERRE-
DIOXIN
(ox.)
2-OXO-GLUTARATE SYNTHASE
Deficient
in hyper-
oxaluria I
PYRUVATE
ThPP
ALANINE-
GLYOXYLATE
TRANS-
AMINASE
OXALYL-CoA
OXALATE
CoA-SH
PyrP
L-ALA-
NINE
TOSE,
EROL
NADPH
+H⁺
NADP⁺
CoA-SH
GLYOXY-
LATE
DEHYDRO-
GENASE
(ACY-
LATING)
FMNH₂ · E
(S)-2-HY-
DROXY ACID
DEHYDRO-
GENASE
O₂
FMN · E
H₂O₂
HOOC-C-OH
COOH
NAD⁺
NADH+H⁺
YCOLATE
GLYOXYLATE
REDUCTASE
GLYOXYLATE
NH₃
H₂O₂
D-AMINO
ACID
OXIDASE
O₂
H₂O
FAD
ThPP
Mg⁺⁺
(S)-2-HYDROXY-
ACID OXIDASE
GLYCINE
TRANSAMINASE
PyrP
2-OXO-
GLUTARATE
LUORO-
ITRATE
FMN · E
FMNH₂ · E
L-GLUT-
COOH
DEHY
NADH+



A Multivariate Network is Network Topology + Node and Edge Attributes

Tradeoff between Topology and Attributes

Choosing efficient encodings for one aspect often interferes with the ability to effectively visualize the other.

SURVEYED 205 PAPERS FROM 1991 – 2018

Technique Papers, Evaluation Papers, Application Papers

EUROVIS 2019
R. S. Laramee, S. Oeltze, and M. Sedlmair
(Guest Editors)

Volume 38 (2019), Number 3
STAR – State of The Art Report

The State of the Art in Visualizing Multivariate Networks

C. Nobre[1], M. Meyer[1], M. Streit[2], and A. Lex[1]

[1]University of Utah, Utah, USA
[2]Johannes Kepler University Linz, Austria

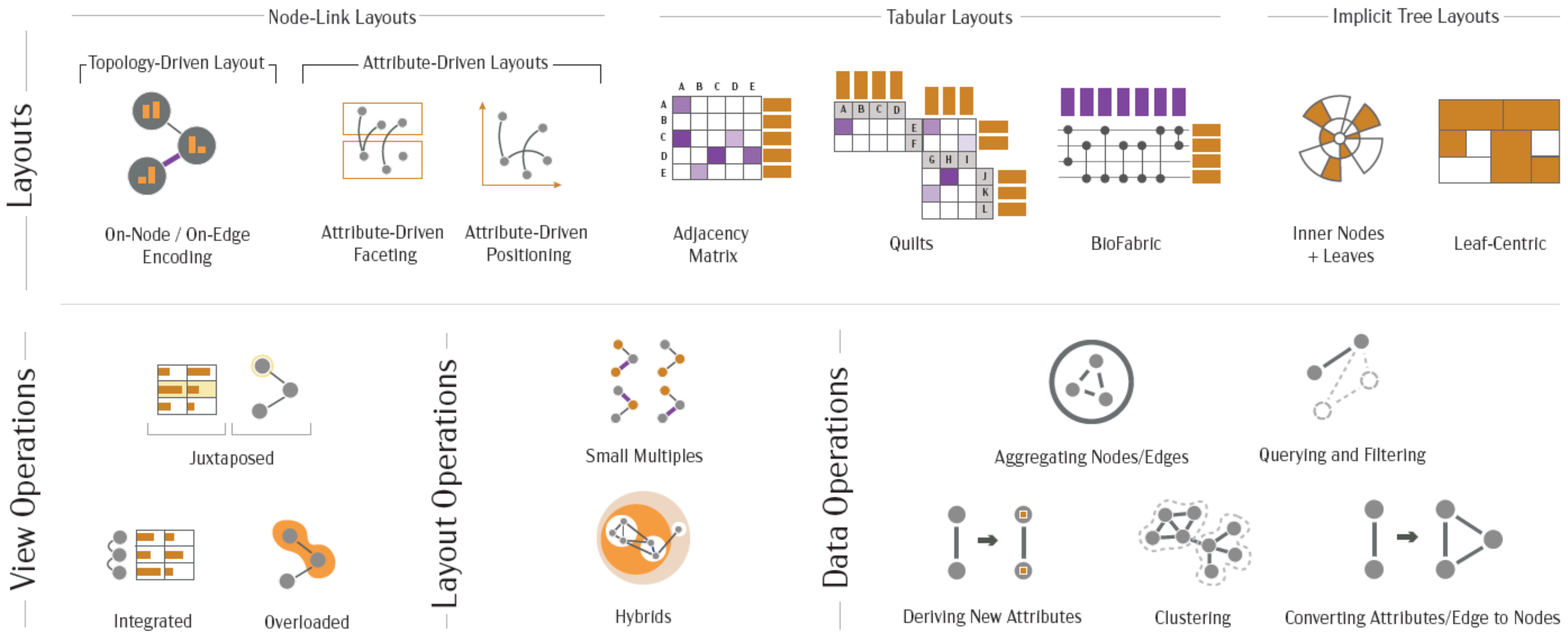
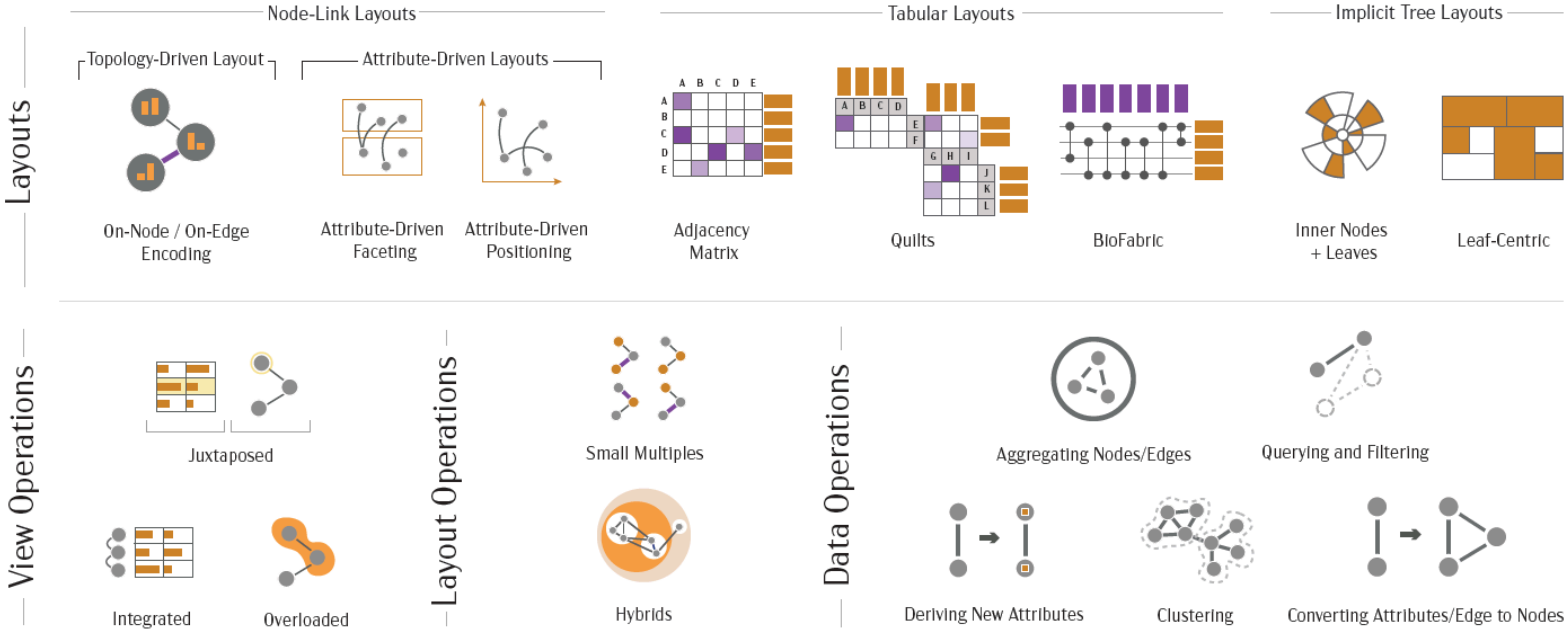


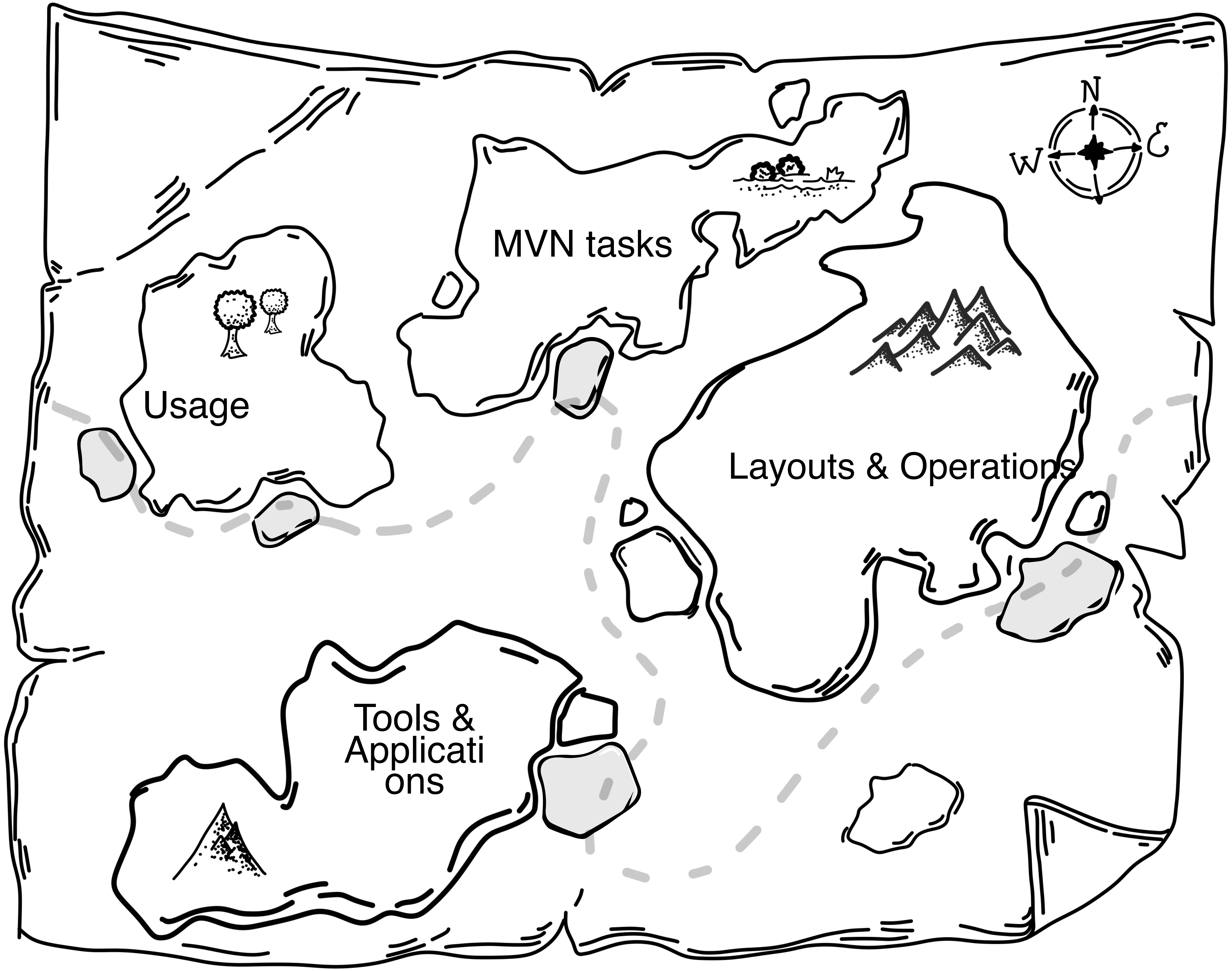
Figure 1: *A typology of operations and layouts used in multivariate network visualization.* ***Layouts*** *describe the fundamental choices for encoding multivariate networks.* ***View Operations*** *capture how topology and attribute focused visualizations can be combined.* ***Layout Operations*** *are applied to basic layouts to create specific visualization techniques.* ***Data Operations*** *are used to transform a network or derive attributes before visualizations. The colors reflect node attributes (orange), edge attributes (purple), and topology (grey).*

Abstract

Multivariate networks are made up of nodes and their relationships (links), but also data about those nodes and links as attributes. Most real-world networks are associated with several attributes, and many analysis tasks depend on analyzing both, relationships and attributes. Visualization of multivariate networks, however, is challenging, especially when both the topology of the network and the attributes need to be considered concurrently. In this state-of-the-art report, we analyze current practices and classify techniques along four axes: layouts, view operations, layout operations, and data operations. We also provide an analysis of tasks specific to multivariate networks and give recommendations for which technique to use in which scenario. Finally, we survey application areas and evaluation methodologies.


Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

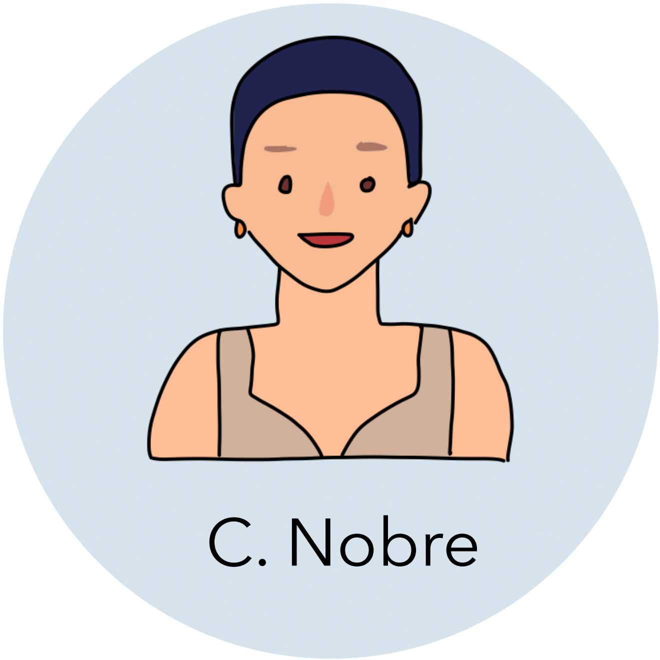
Land of Multivariate

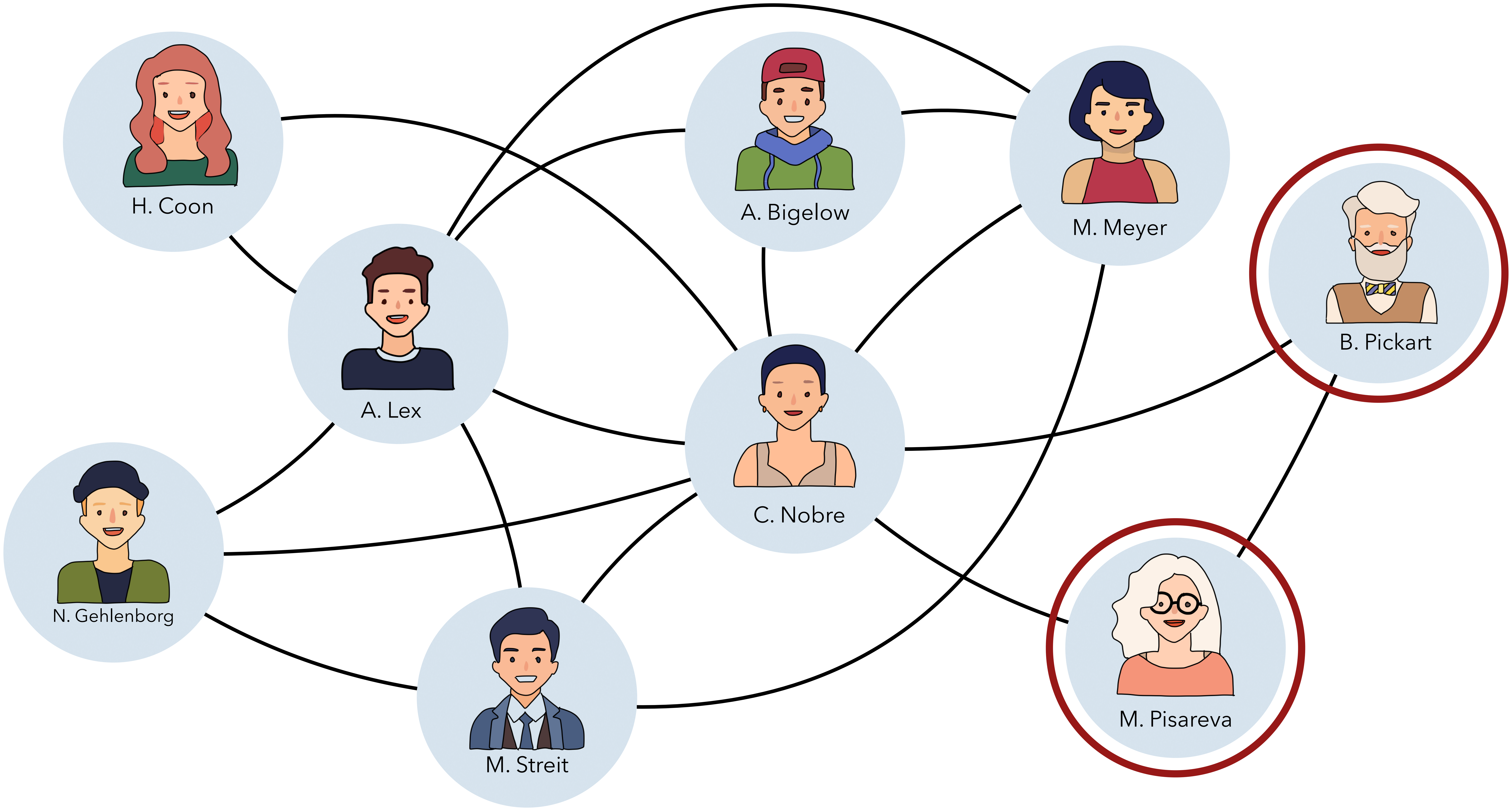


MVNV Tasks

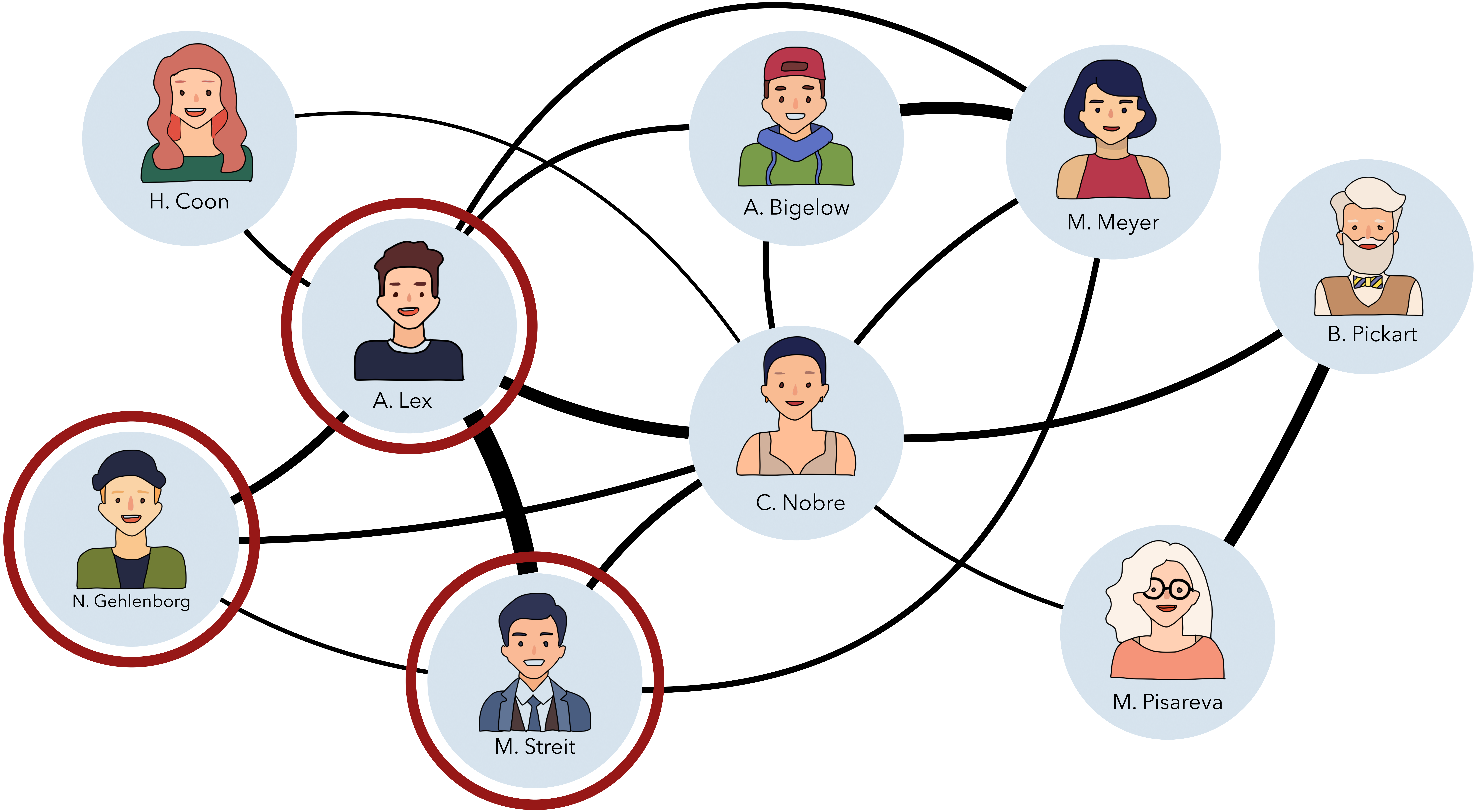
How is an MVN task different than a regular graph task?

MVN Tasks rely on both the topology of the network and the attributes of the nodes and edges

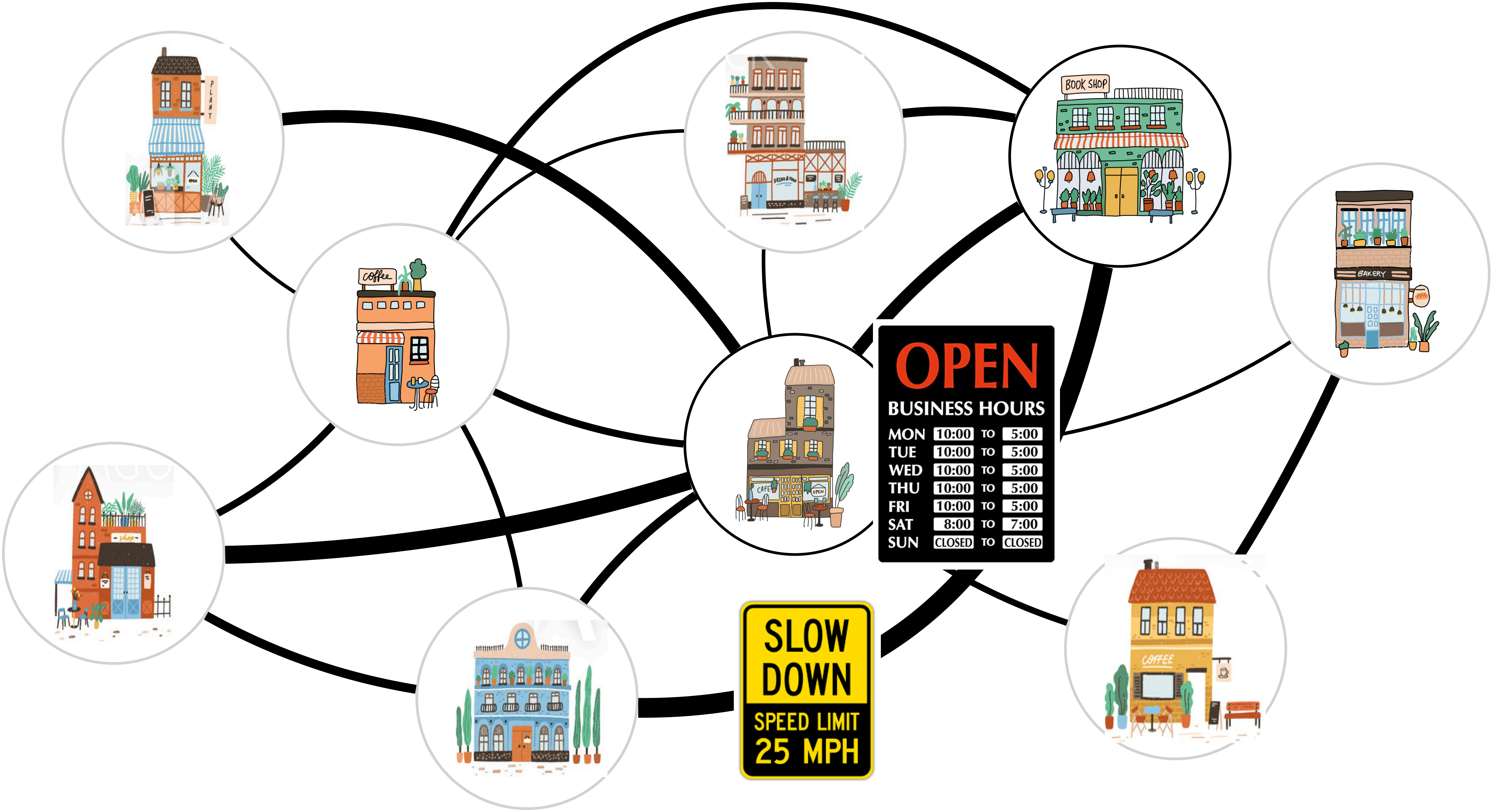

C. Nobre



How many of my collaborators are from the oceanography field?



Which cluster of authors has the highest number of combined collaborations?



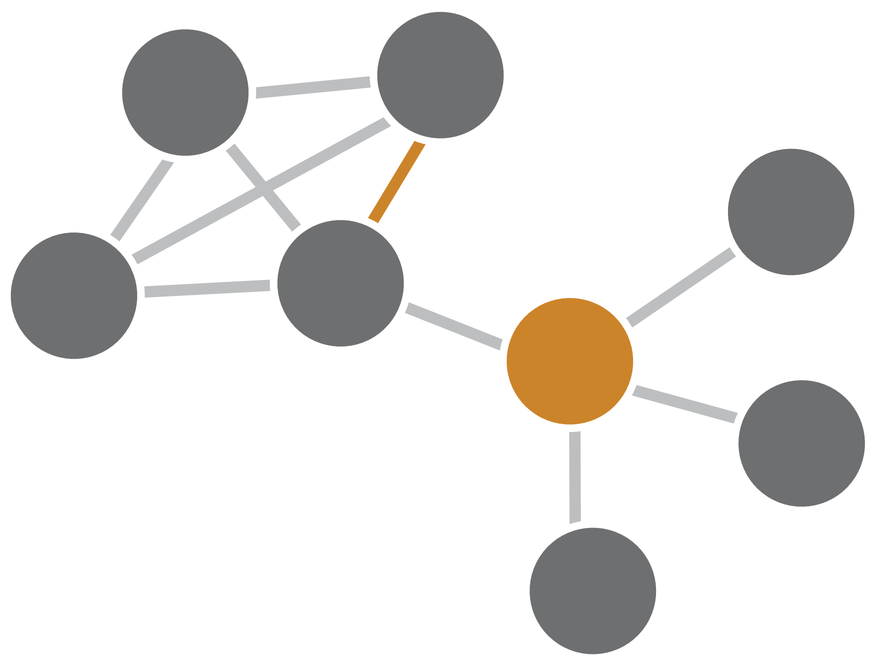
What is an efficient way I can complete all my errands?

- How many of my collaborators are in the oceanography field?
- Which cluster has the highest number of collaborations?
- What is the fastest route to get all my errands done?

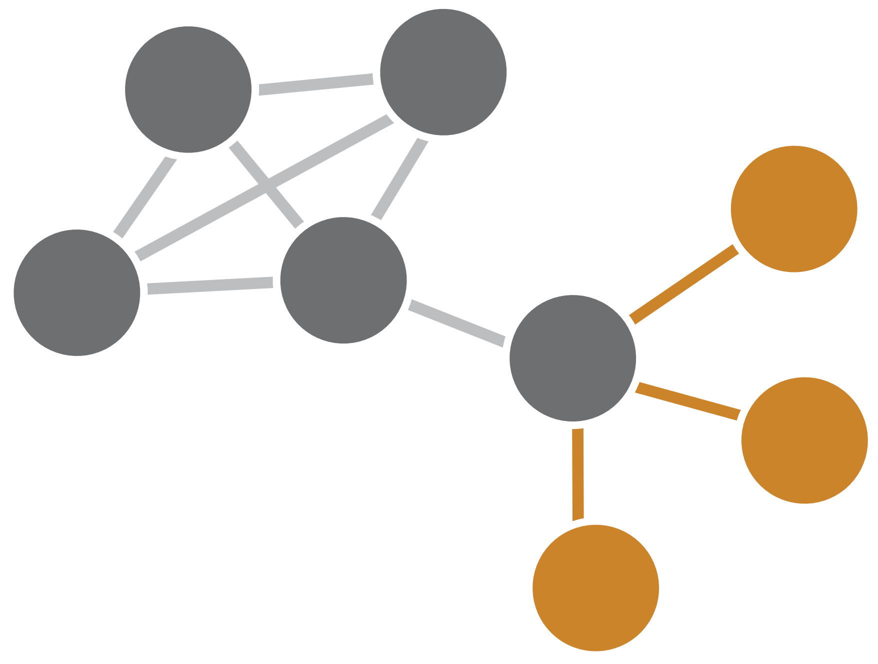
Tasks that rely on the **topology** of the network and the **attributes** of the nodes and edges

MVNV tasks are applied to topological structures

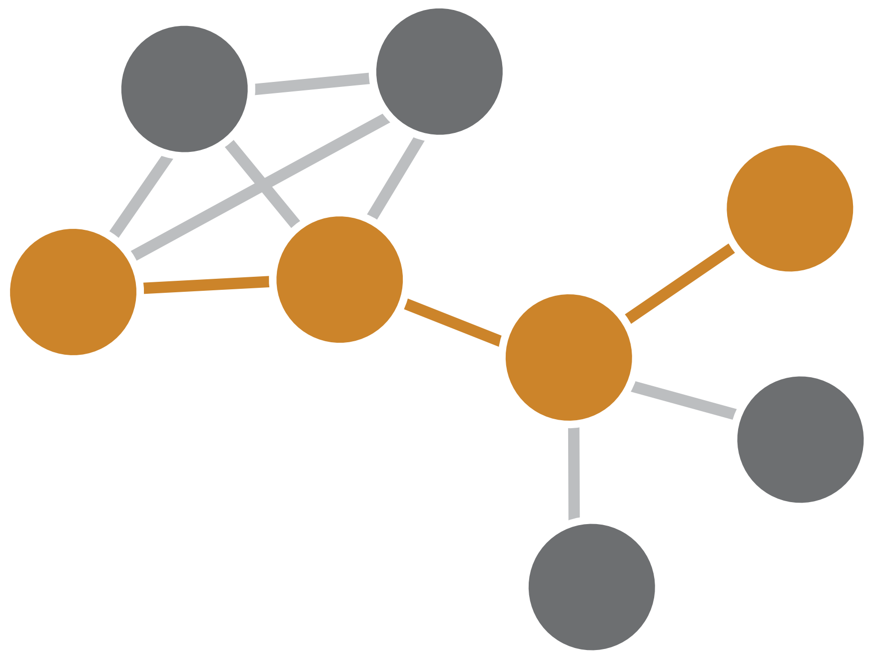
Single Node/Edge



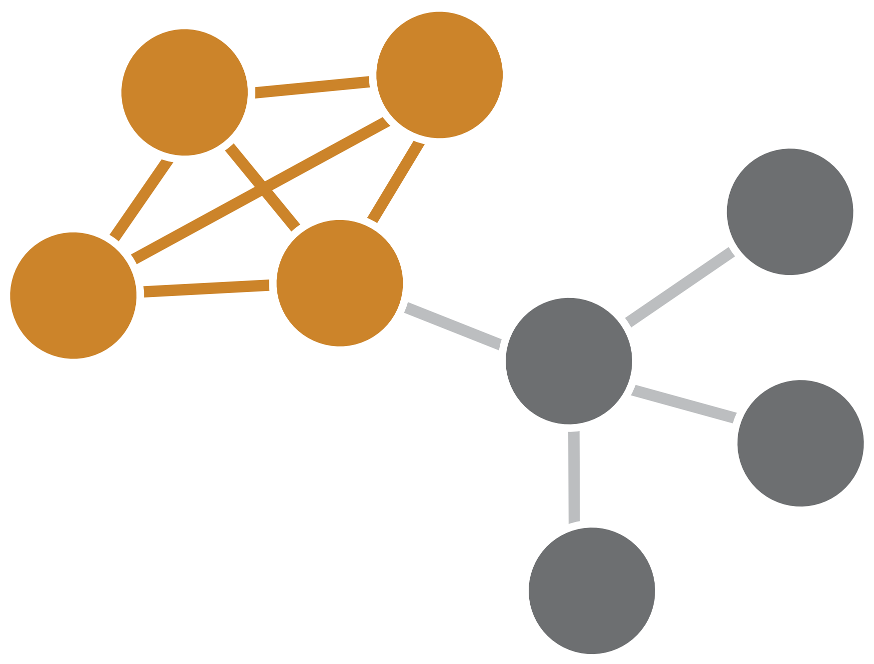
Node Neighbors



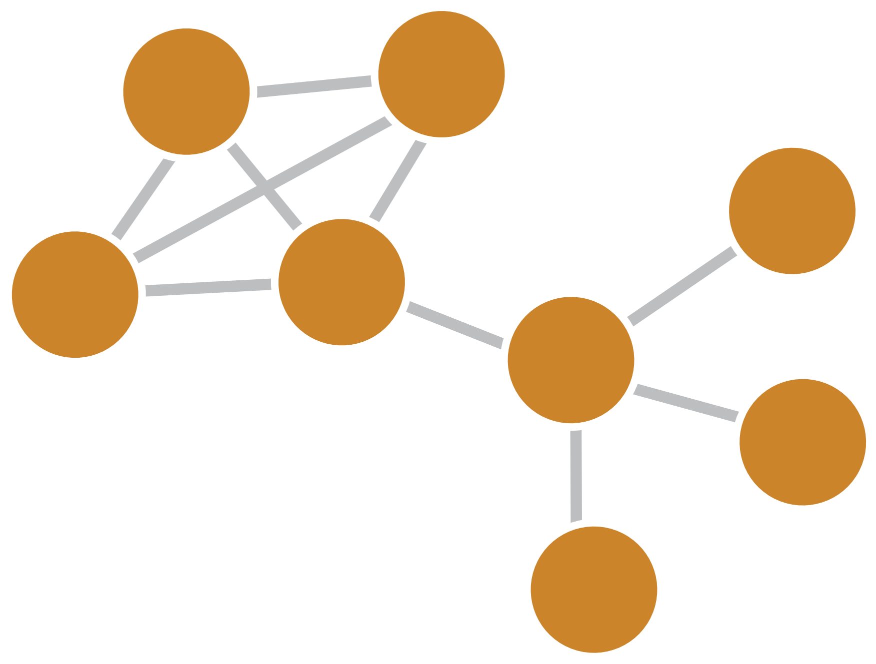
Path



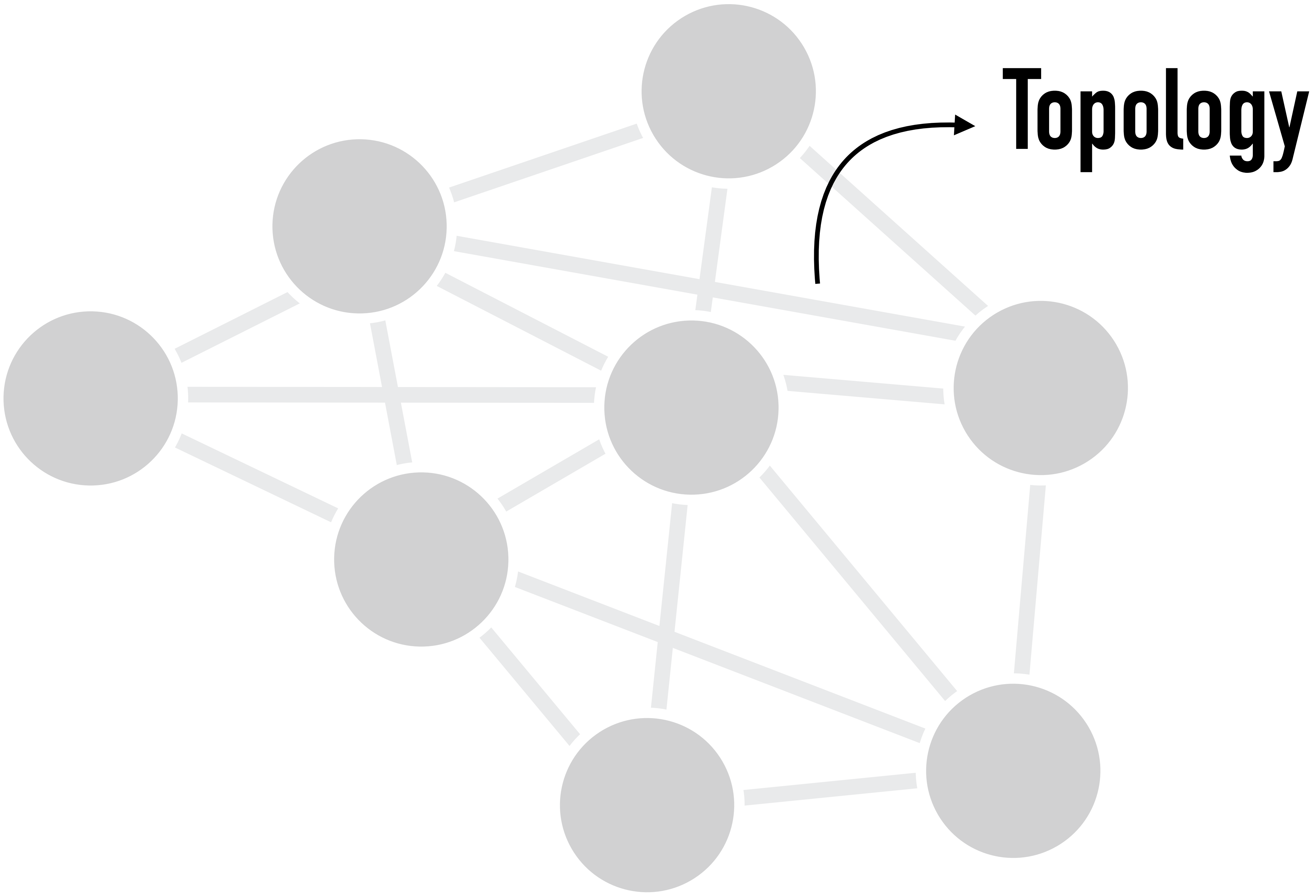
Cluster

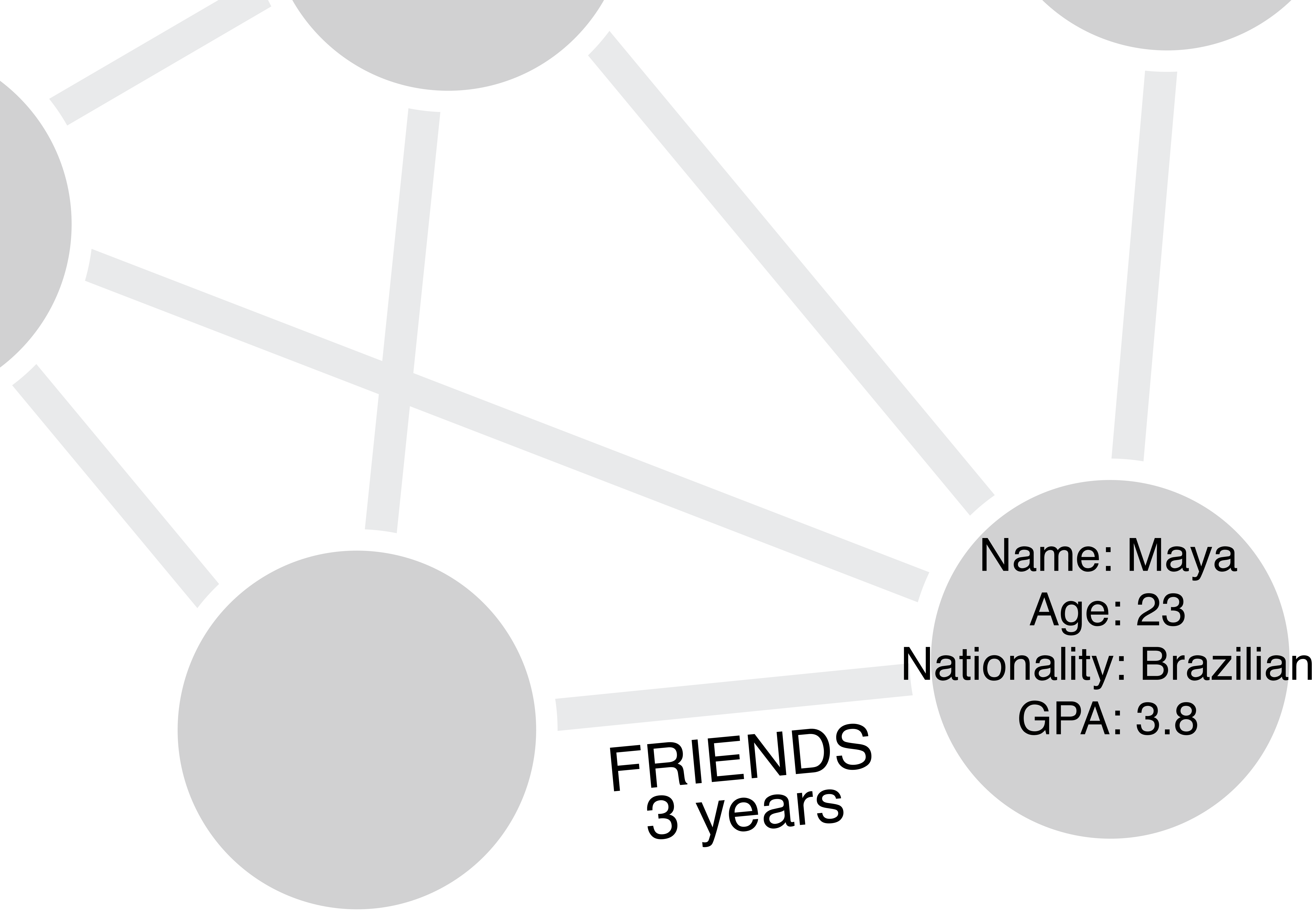


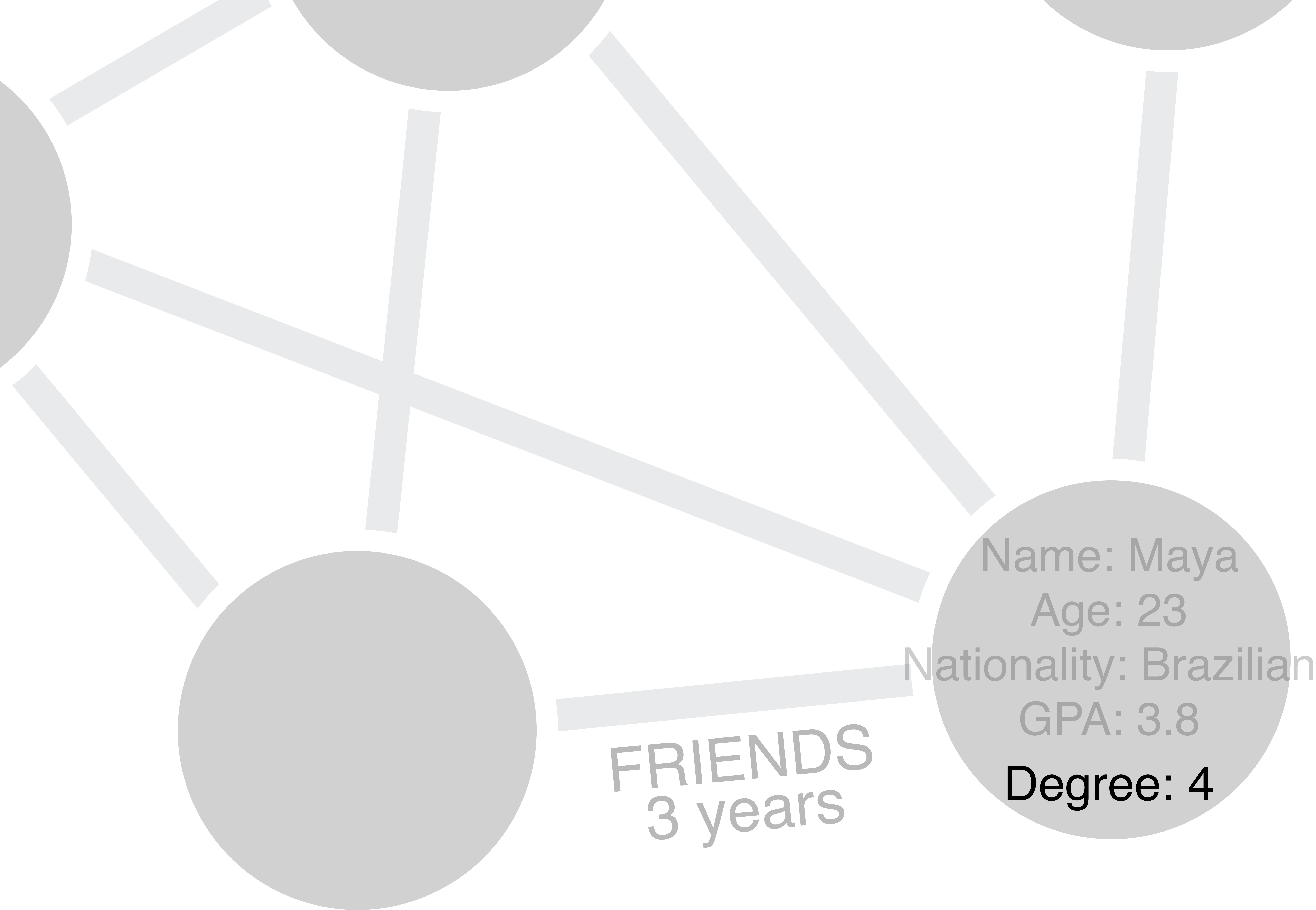
Network/Subnetwork

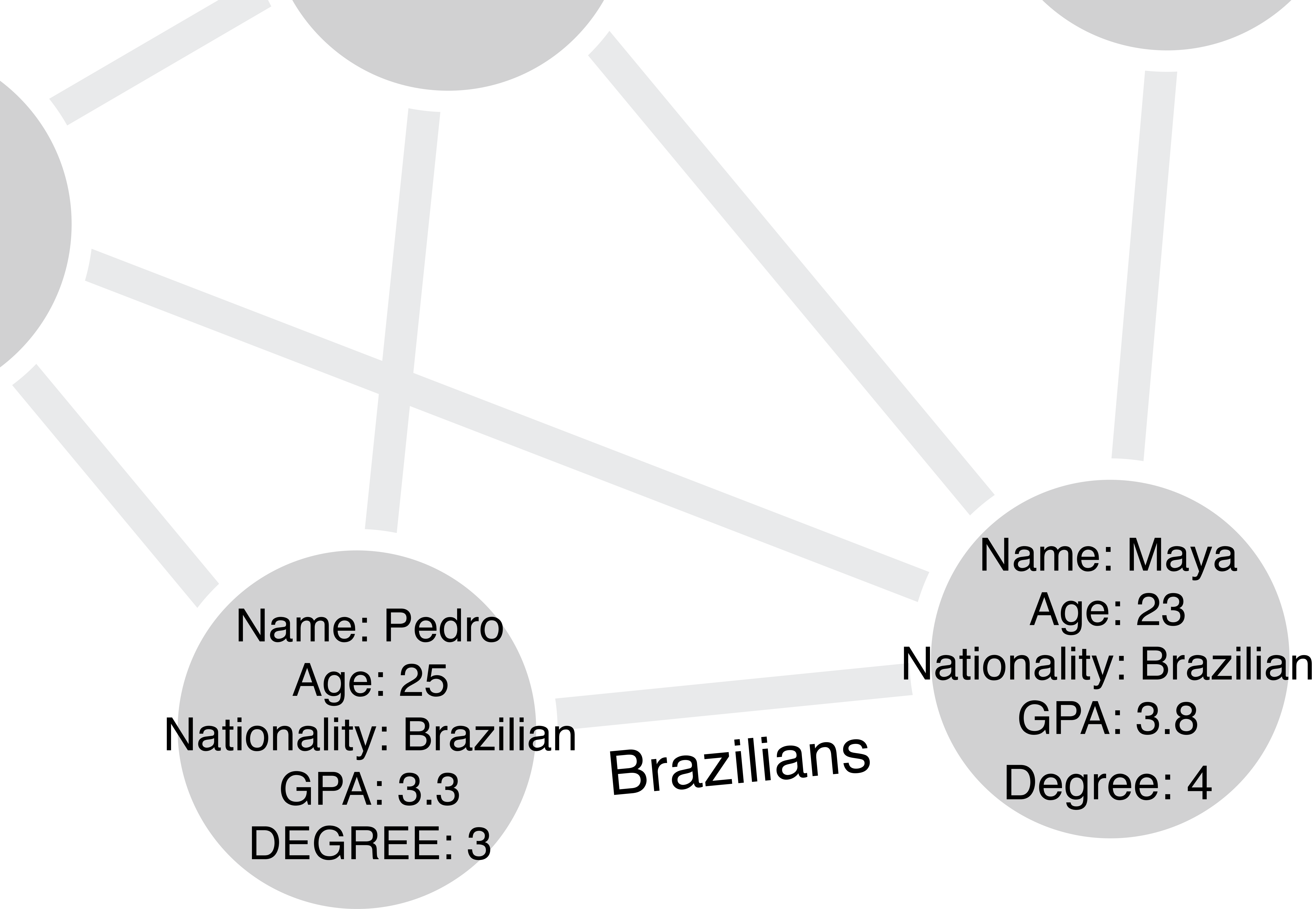


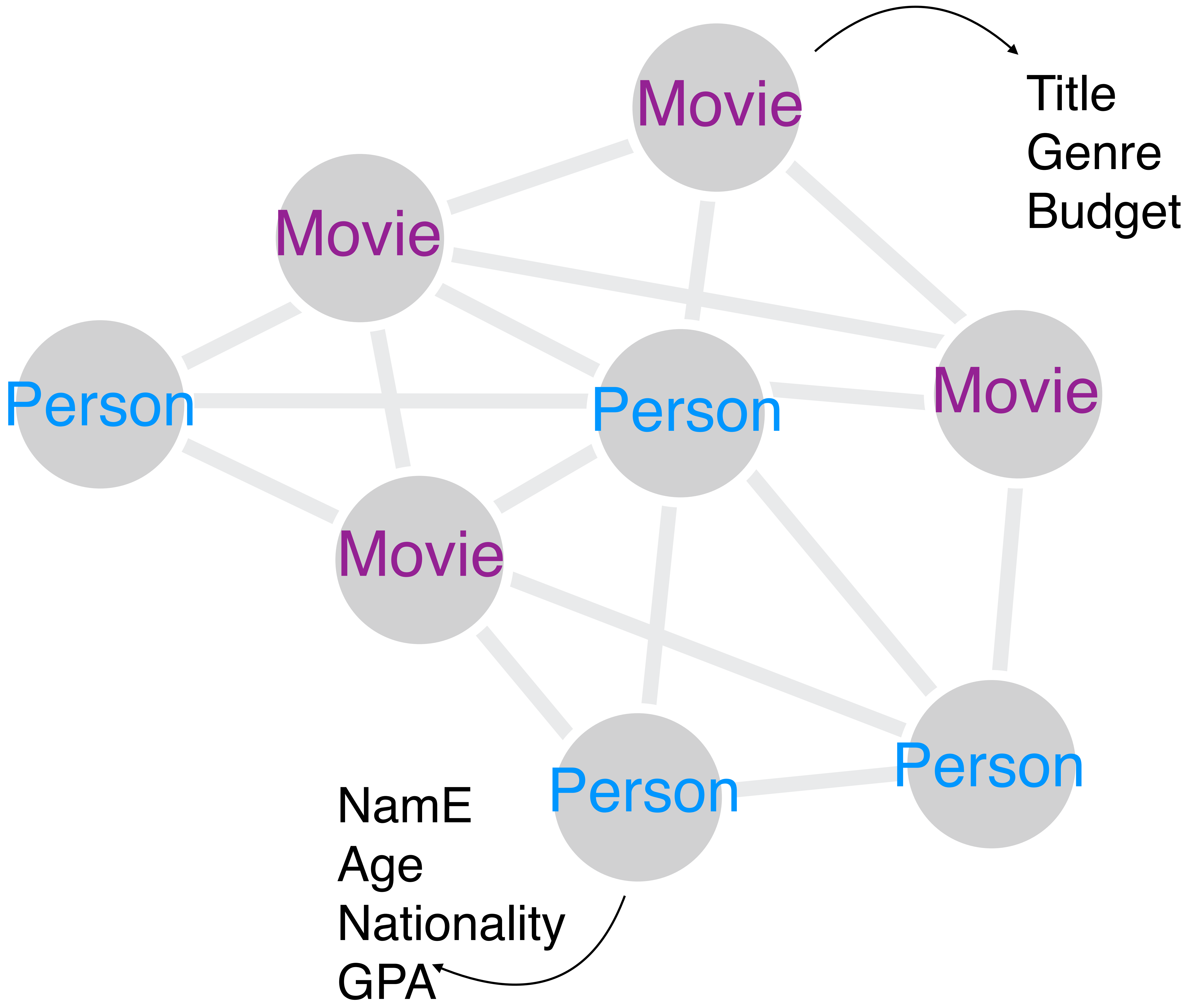
Network and Attribute Characteristics


Topology

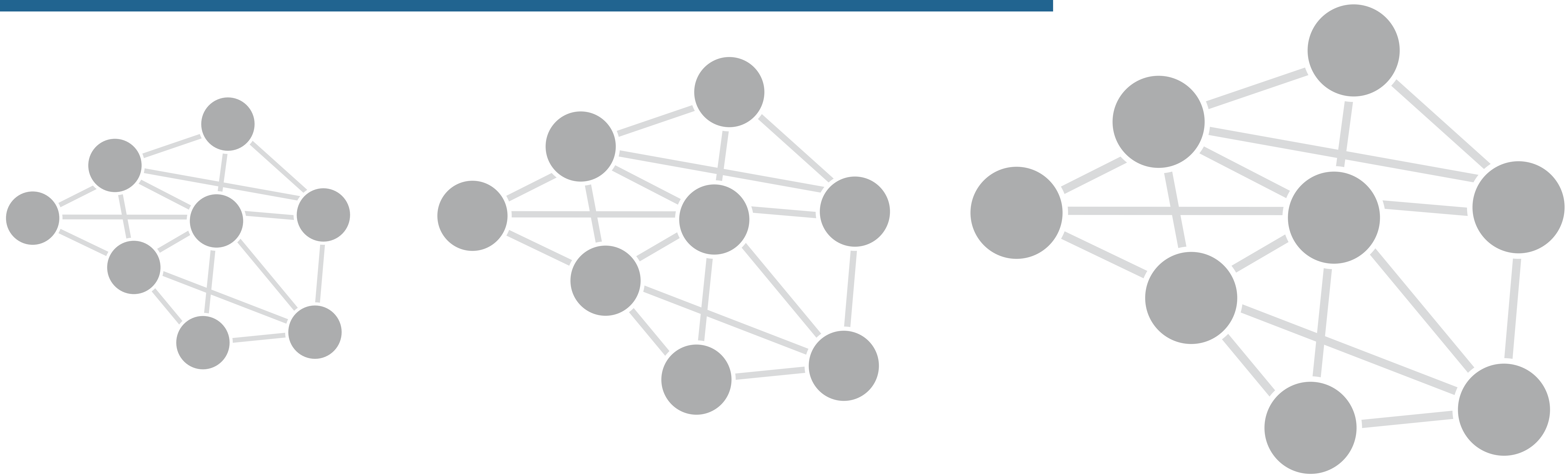

Name: Maya
Age: 23
Nationality: Brazilian
GPA: 3.8
FRIENDS
3 years


Name: Maya
Age: 23
Nationality: Brazilian
GPA: 3.8
Degree: 4
FRIENDS
3 years


Name: Pedro
Age: 25
Nationality: Brazilian
GPA: 3.3
DEGREE: 3
Brazilians
Name: Maya
Age: 23
Nationality: Brazilian
GPA: 3.8
Degree: 4


Movie
Title
Genre
Budget
Movie
Person
Person
Movie
Movie
Person
Person
NamE
Age
Nationality
GPA

Network Size

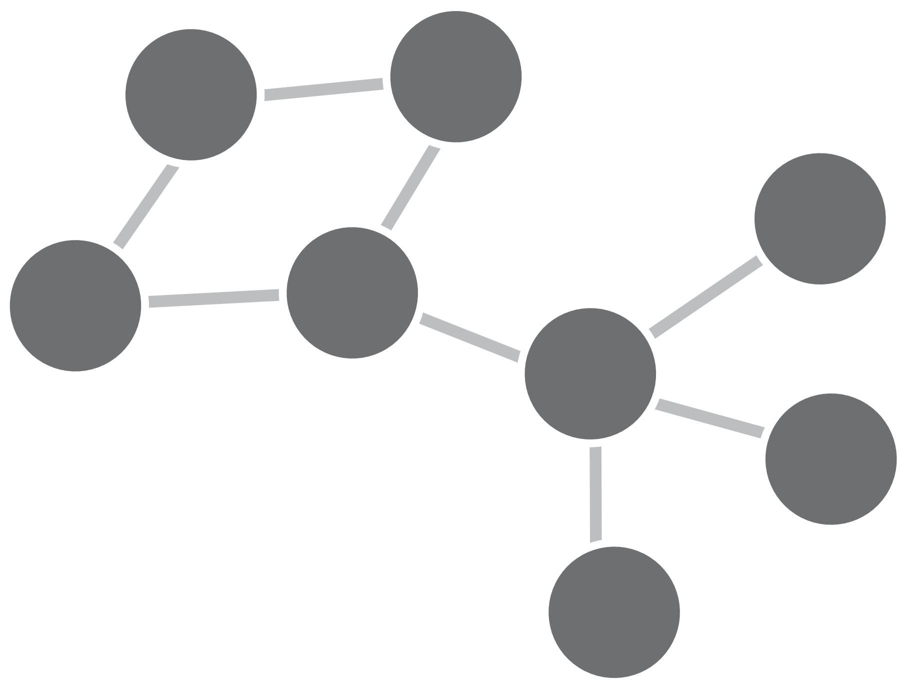


Small
<100

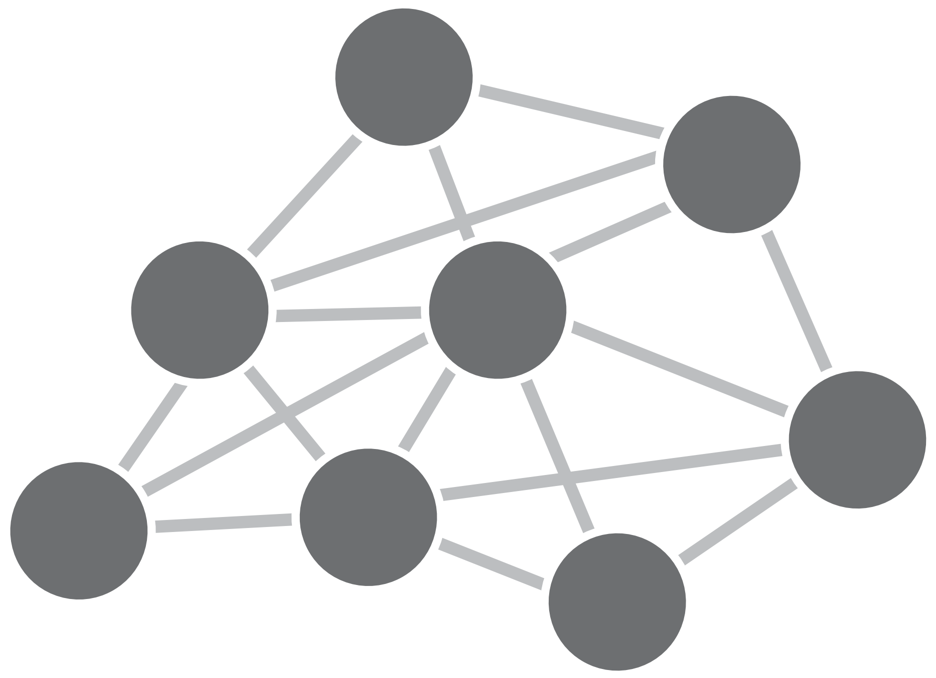
Medium
100–1000

Large
>1000

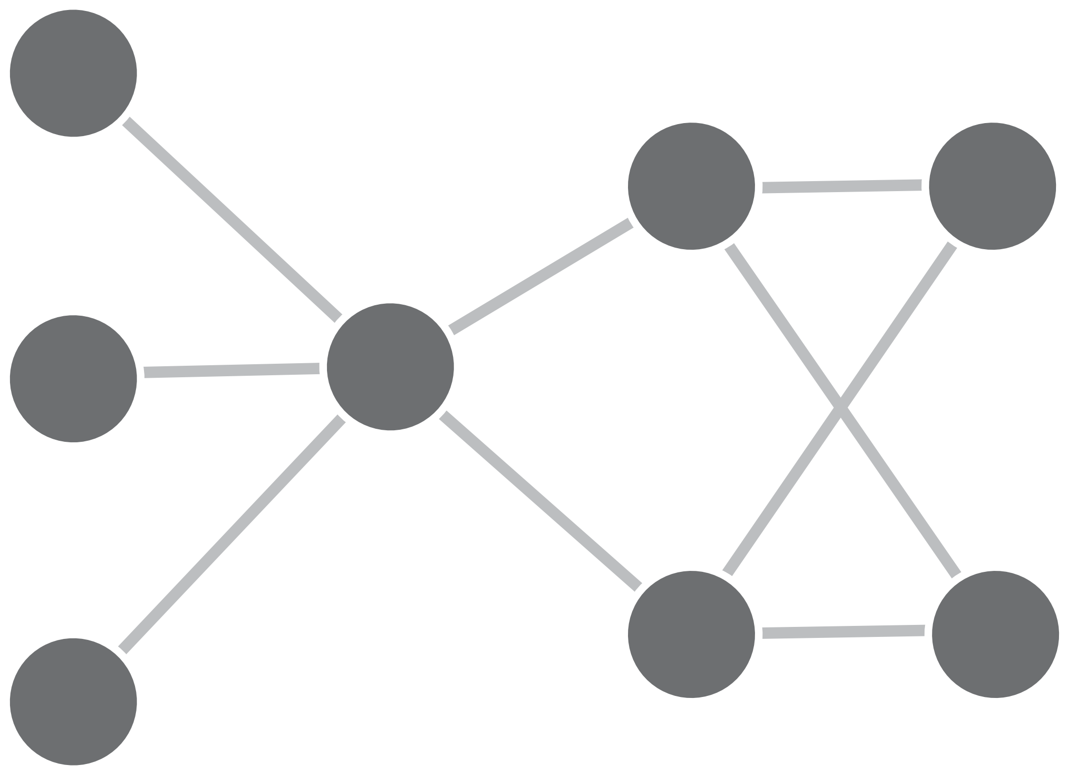
Network Types



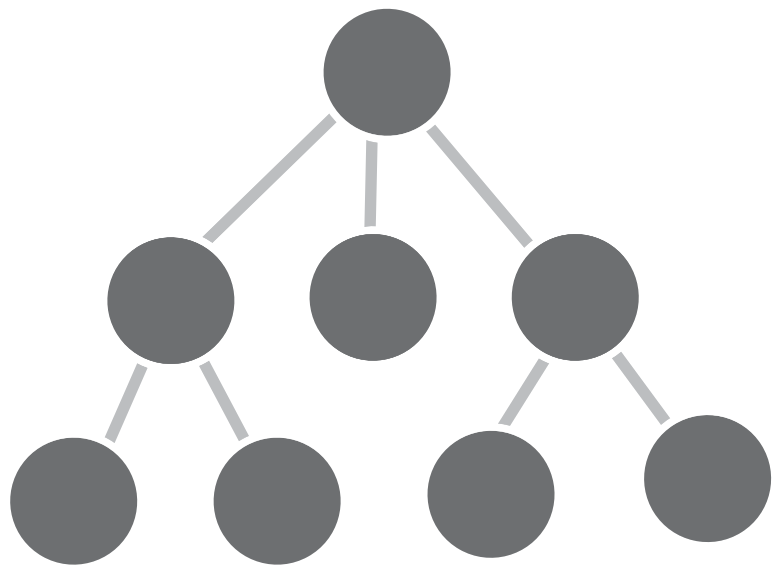
Sparse



Dense

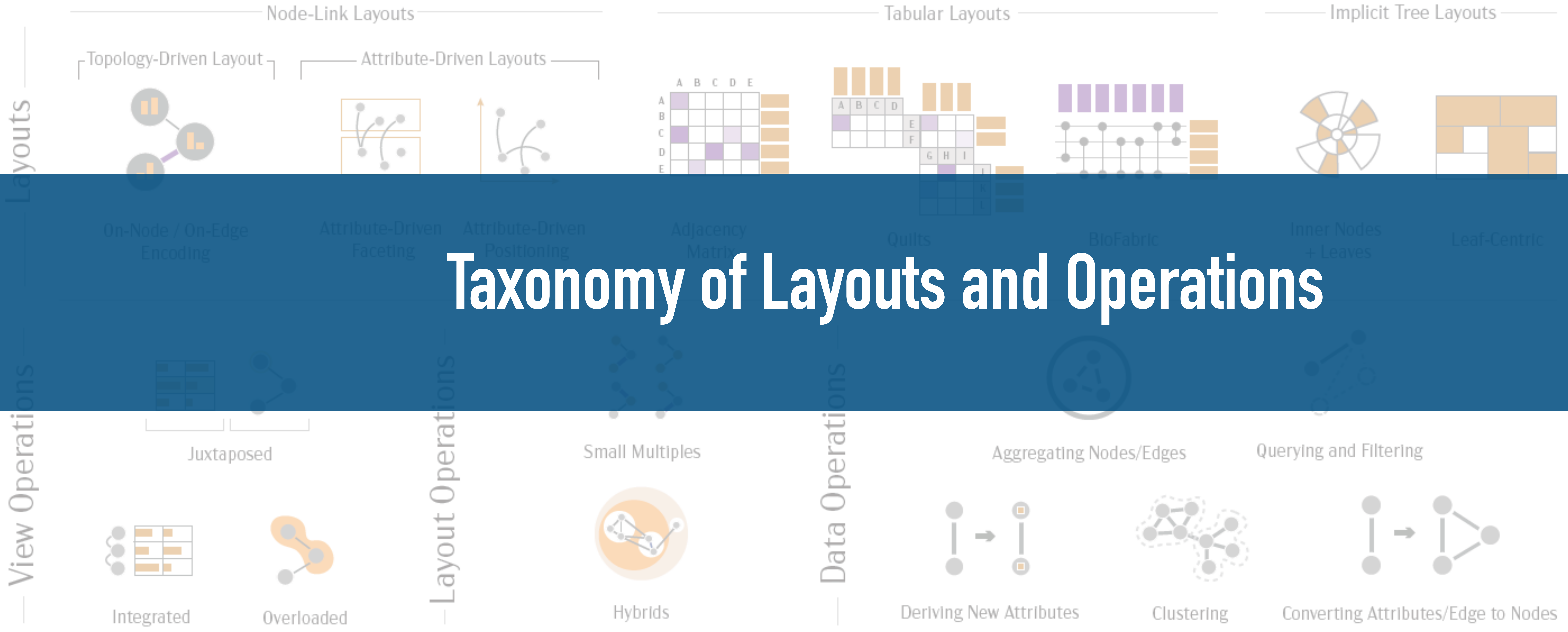


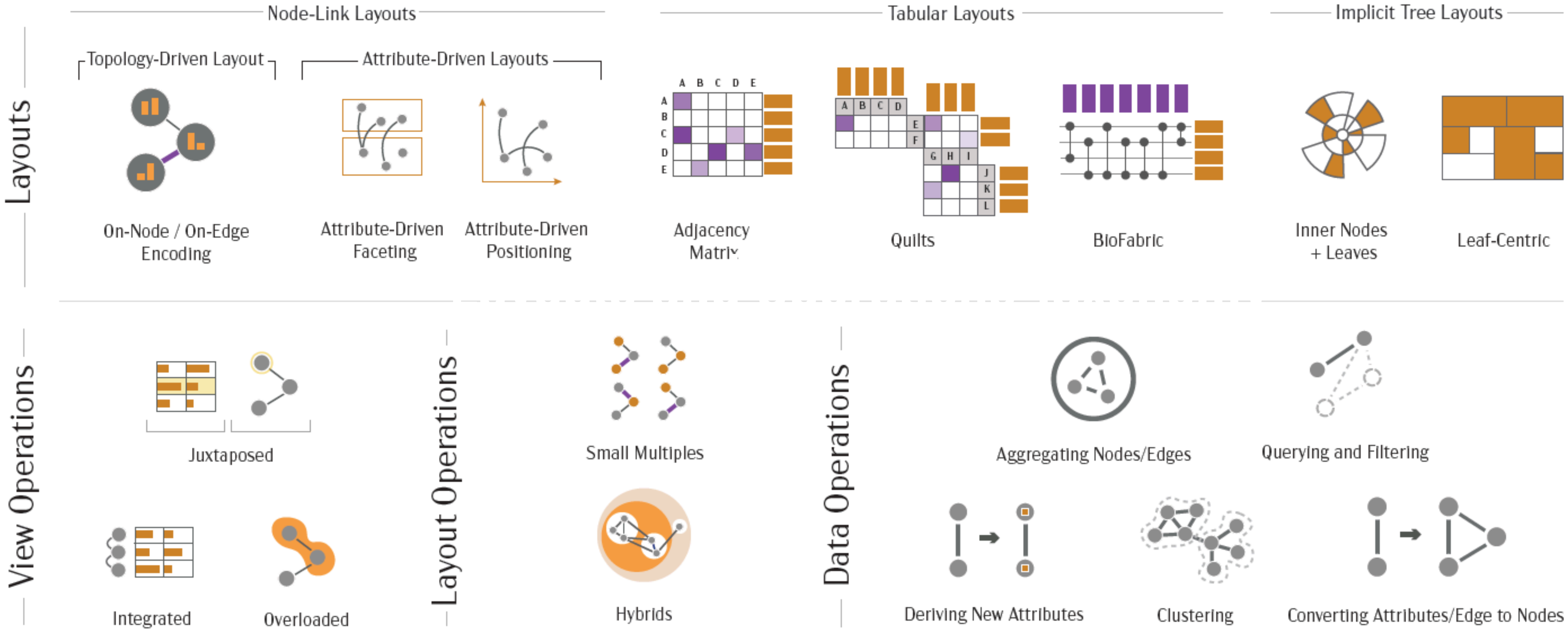
Layered

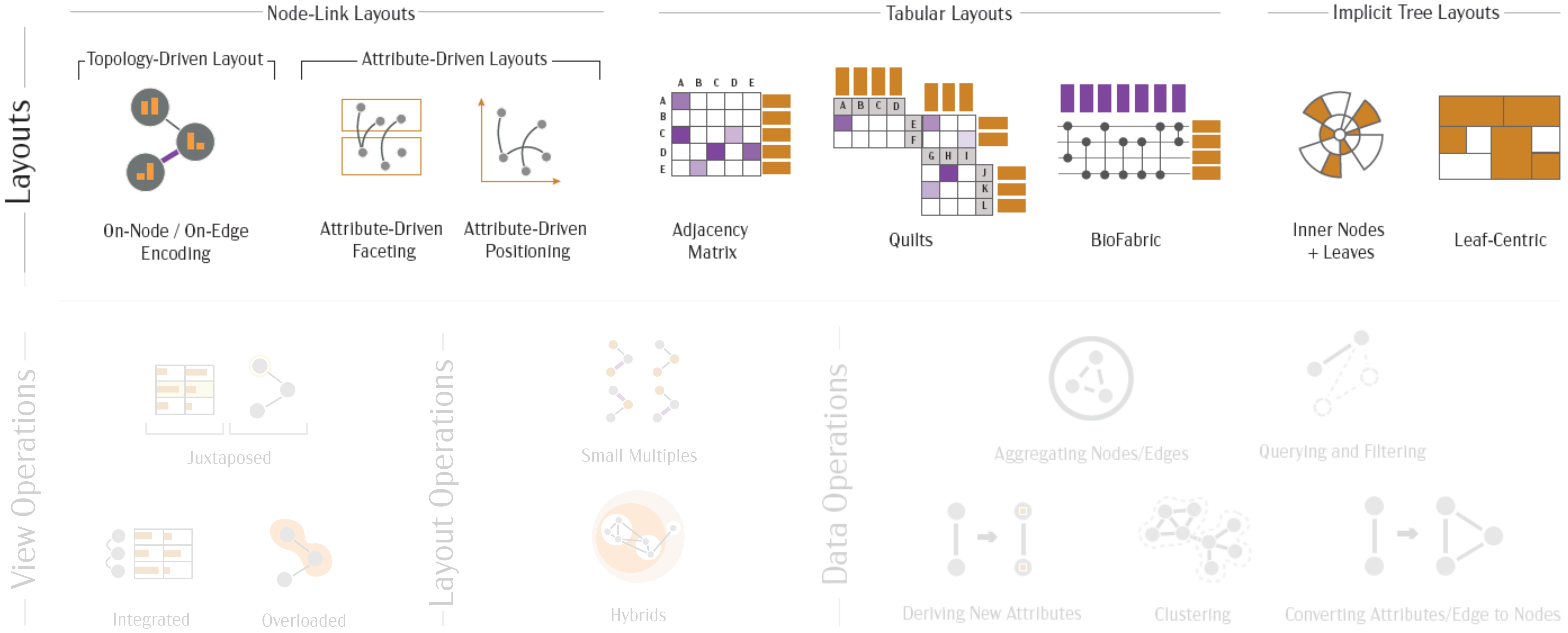


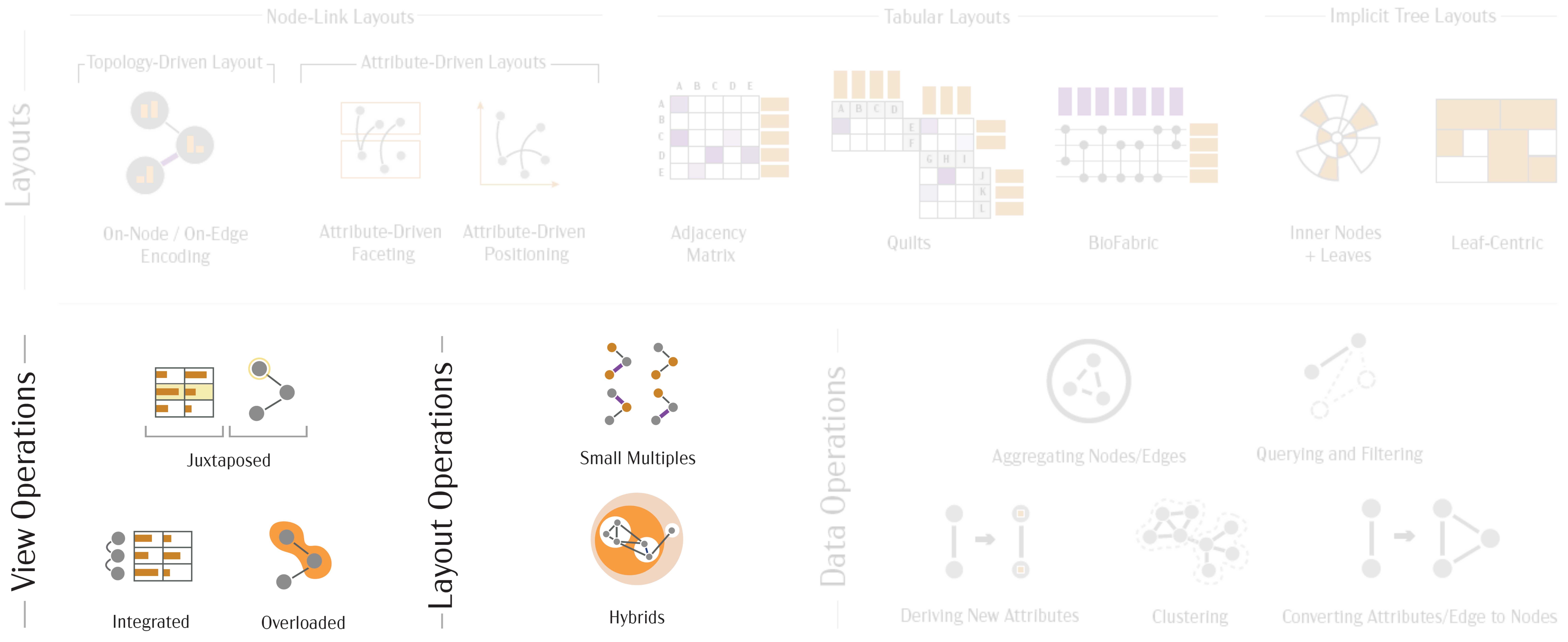
Trees

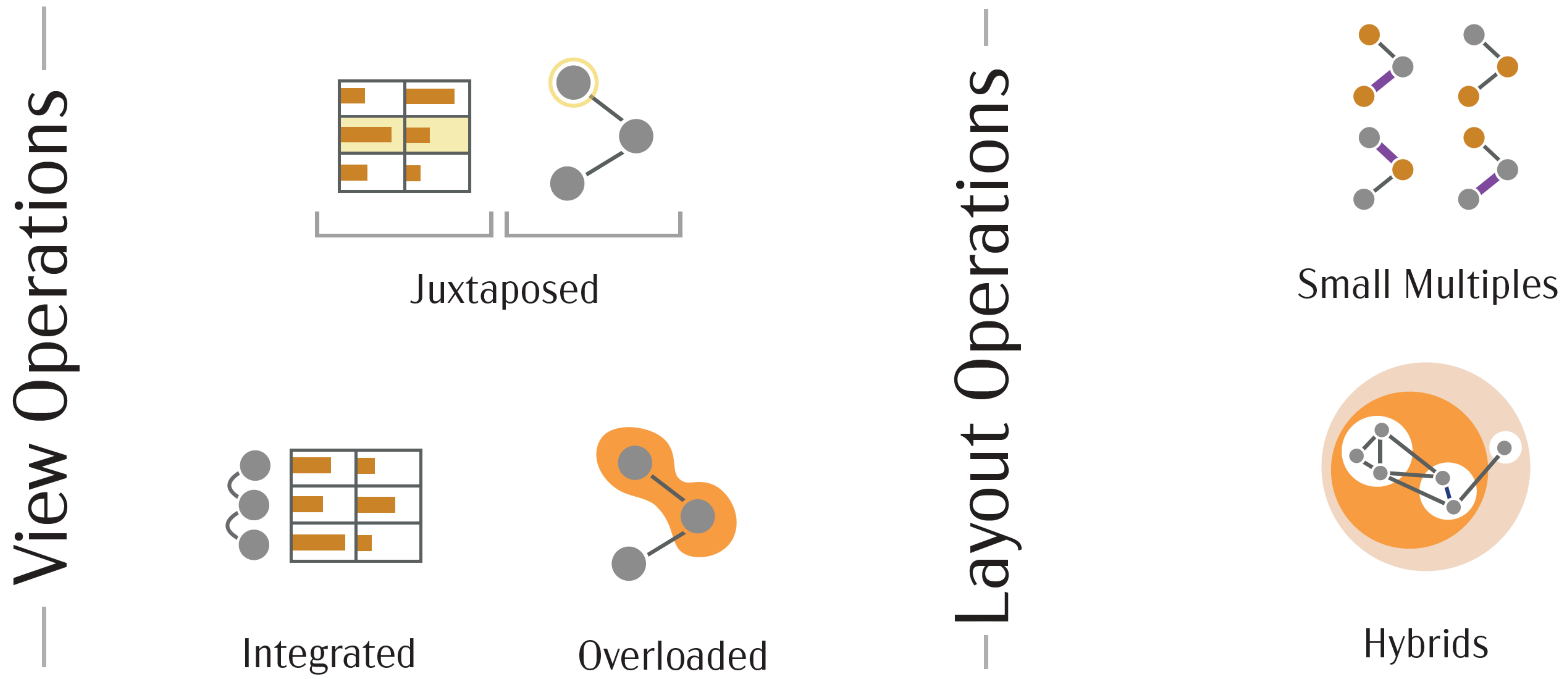
Taxonomy of Layouts and Operations




Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

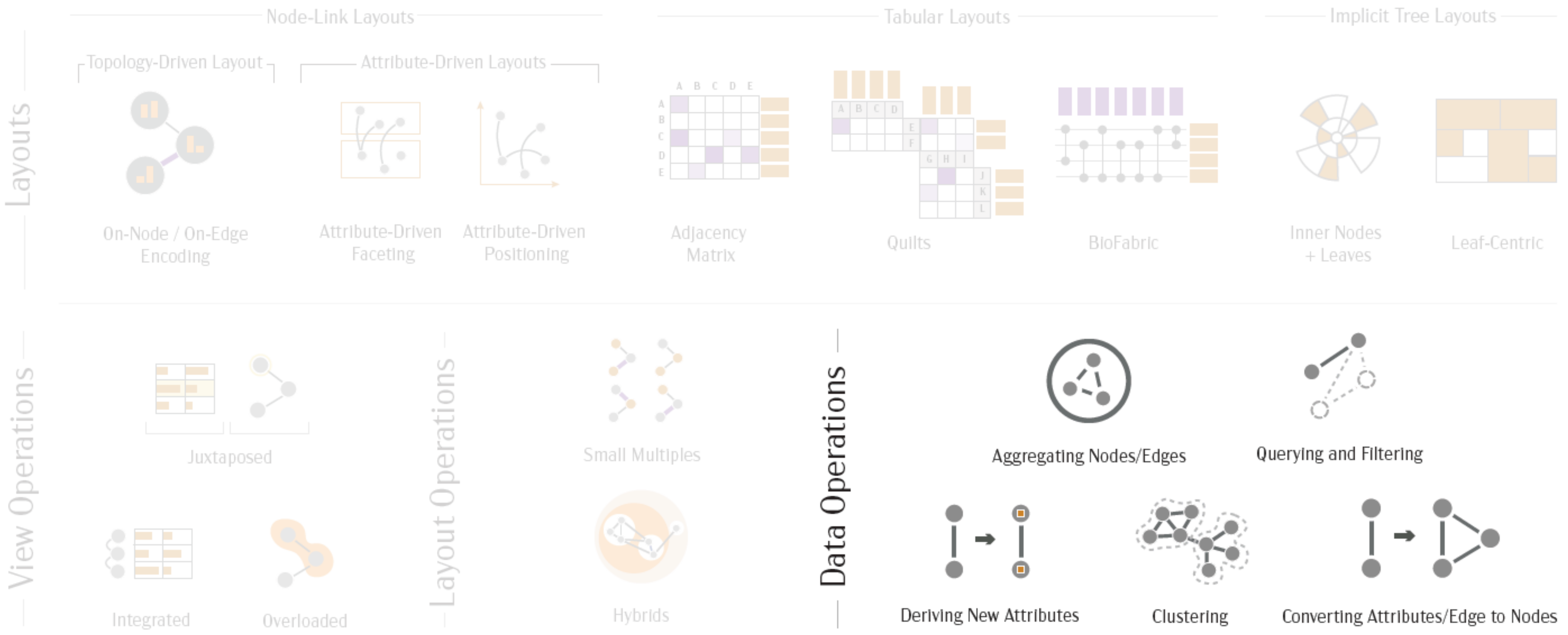

Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes


Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes



Separate views for Topology and Attributes

Multiple layouts for Topology or Attributes



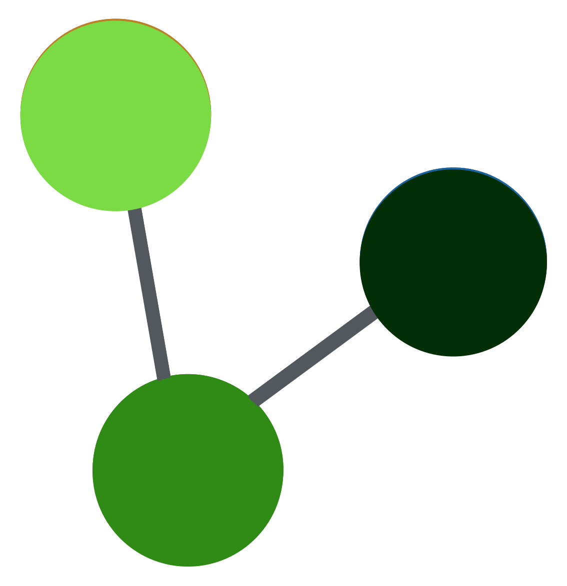
Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes


Layouts
Node-Link Layouts
Tabular Layouts
Implicit Tree Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Adjacency Matrix
Quilts
BioFabric
Inner Nodes + Leaves
Leaf-Centric
LAYOUTS
View Operations
VIEW OPERATIONS
Integrated
Overloaded
Layout Operations
LAYOUT OPERATIONS
Hybrids
Data Operations
Querying and Filtering
DATA OPERATIONS
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

LAYOUTS

VIEW OPERATIONS	LAYOUT OPERATIONS	DATA OPERATIONS

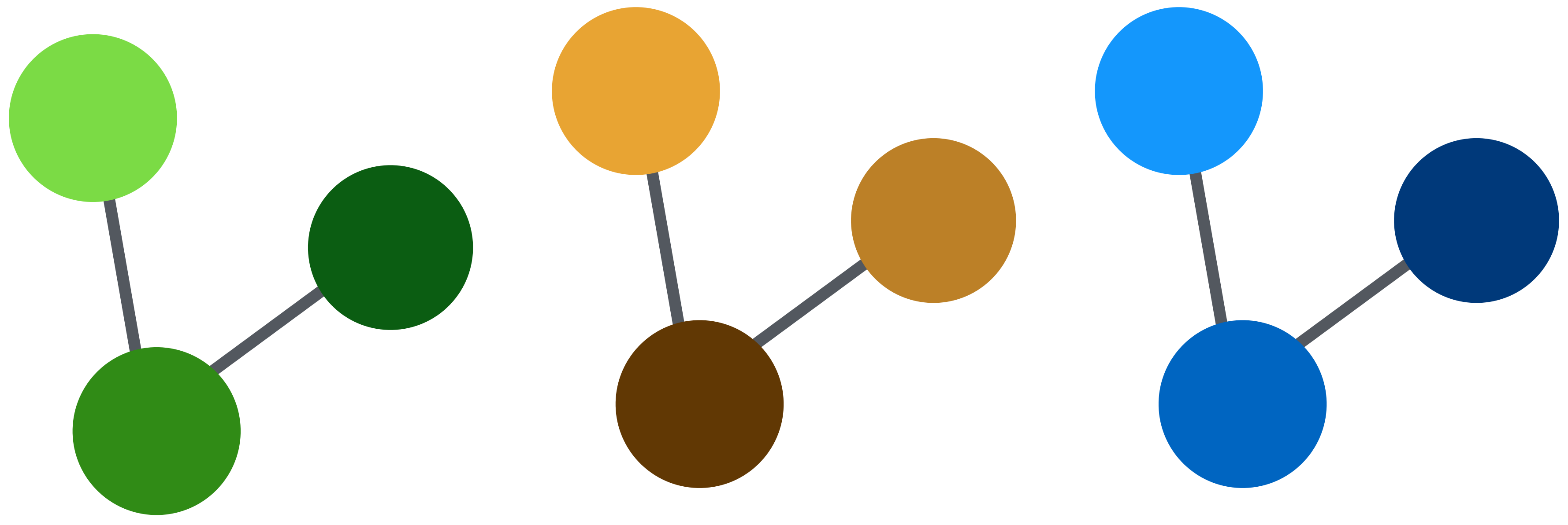
Node-Link Diagram with on-node encoding



LAYOUTS

VIEW OPERATIONS | LAYOUT OPERATIONS | DATA OPERATIONS

Small Multiples



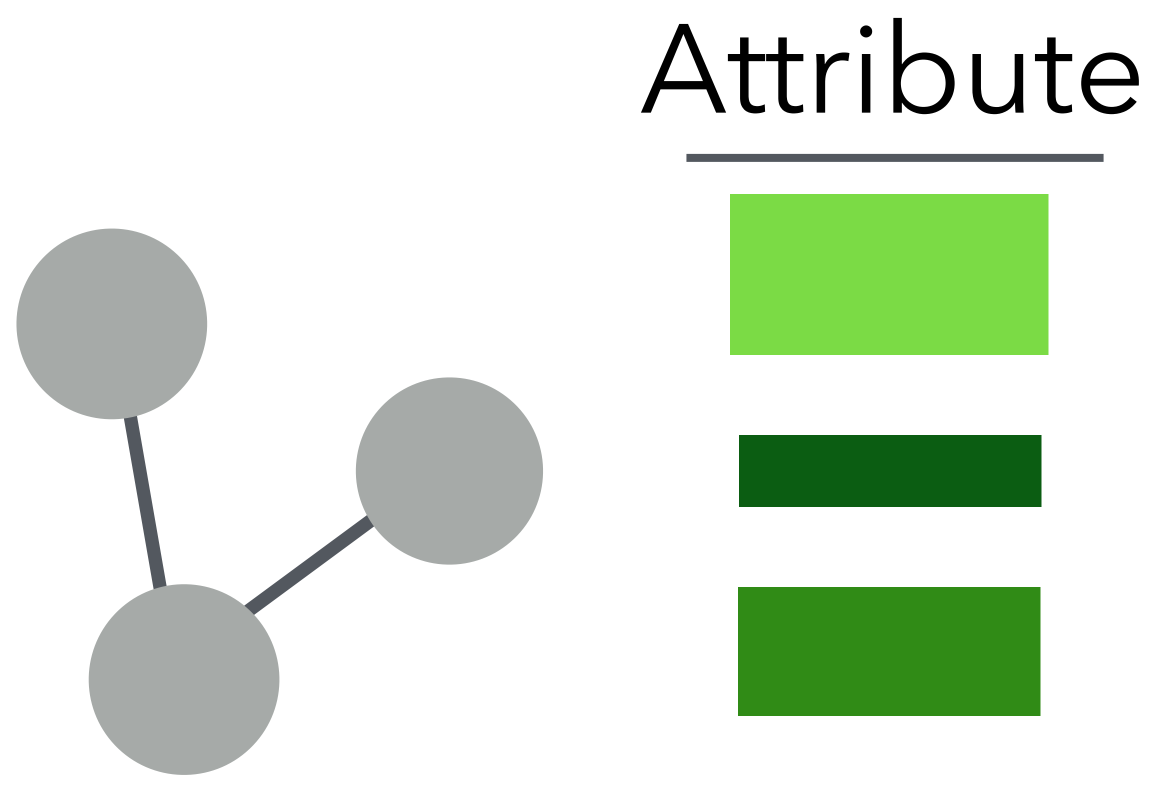
LAYOUTS

VIEW OPERATIONS

LAYOUT OPERATIONS

DATA OPERATIONS

Juxtaposed Views



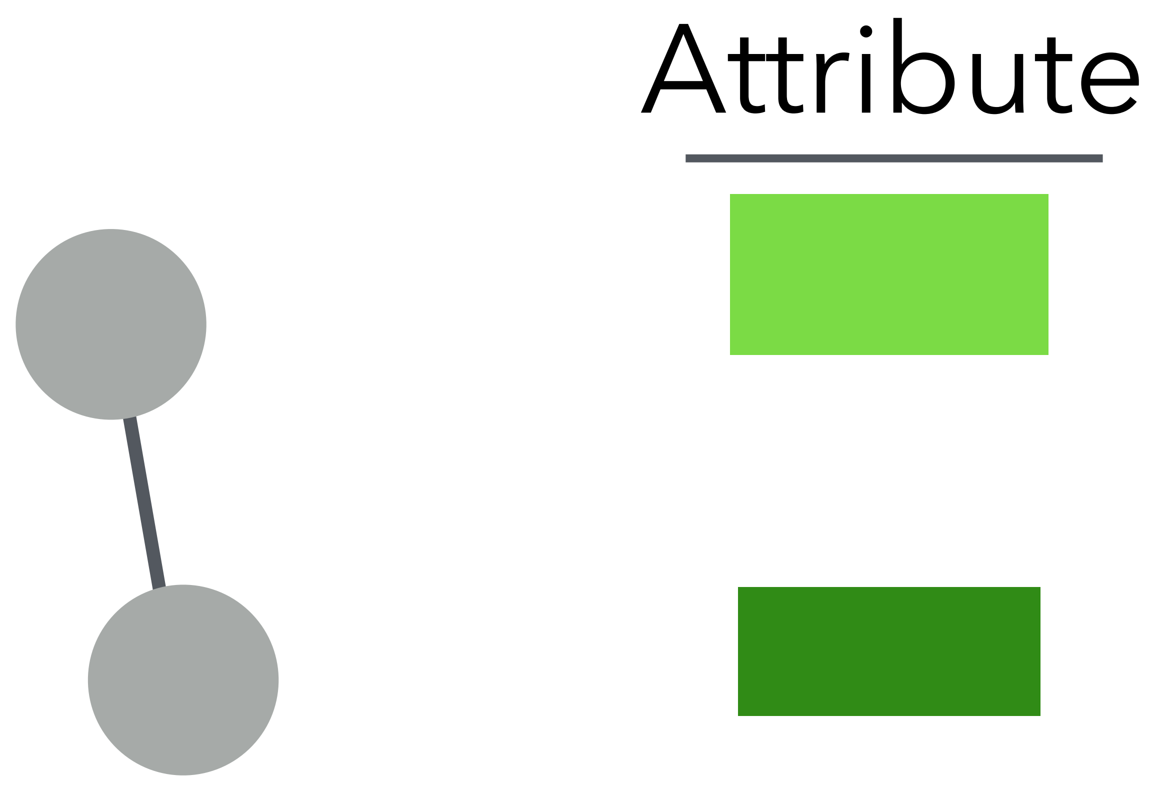
LAYOUTS

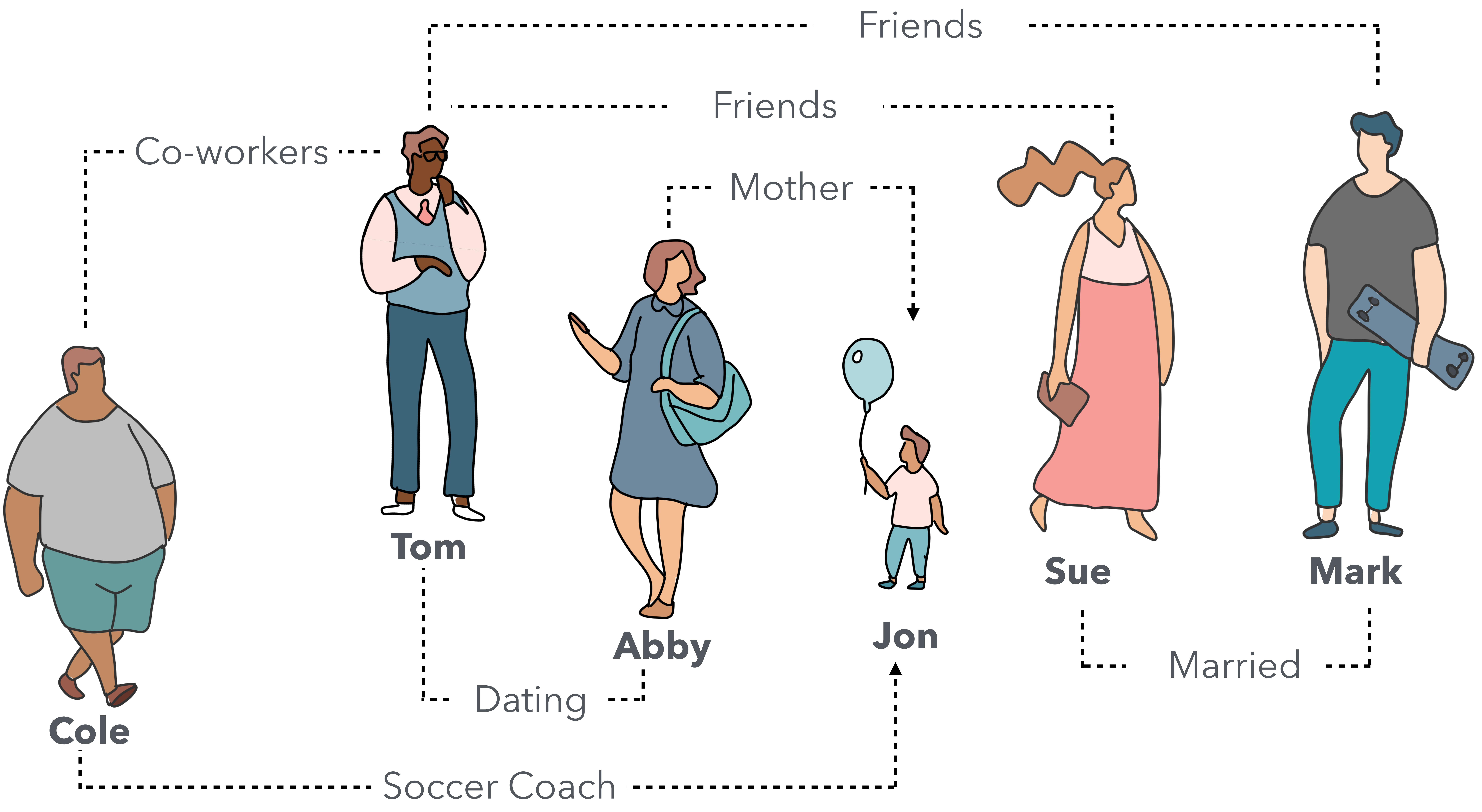
VIEW OPERATIONS

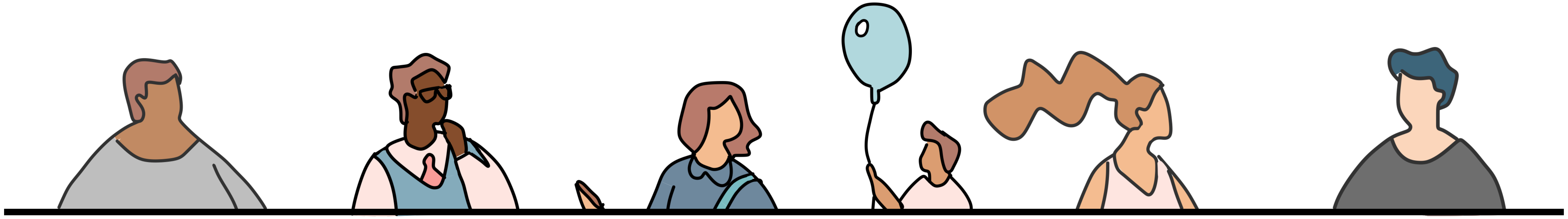
LAYOUT OPERATIONS

DATA OPERATIONS

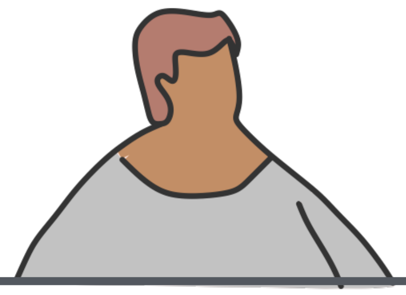
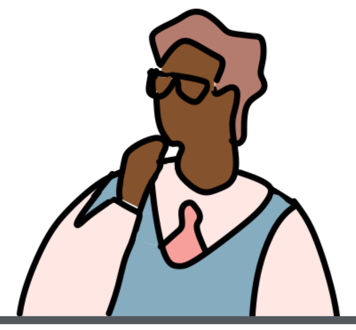
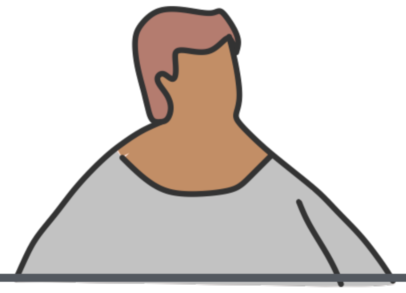
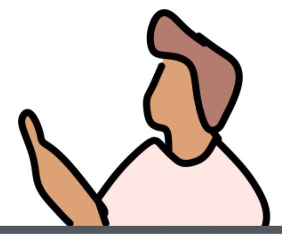
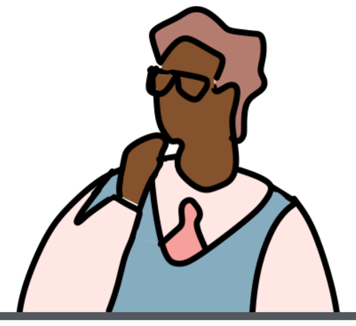
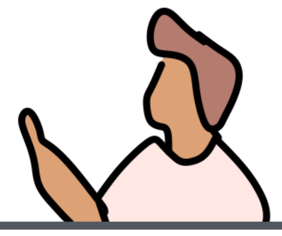
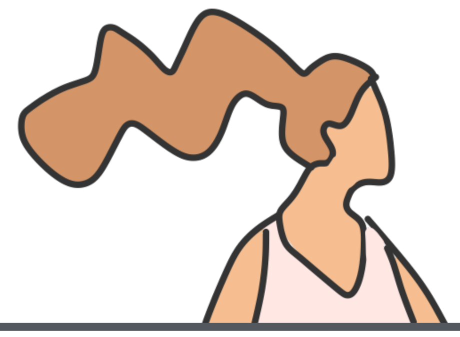
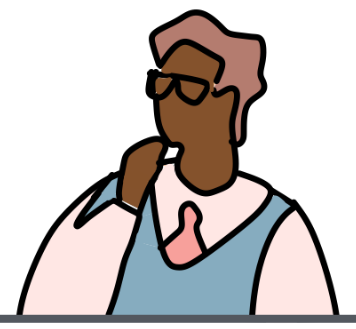
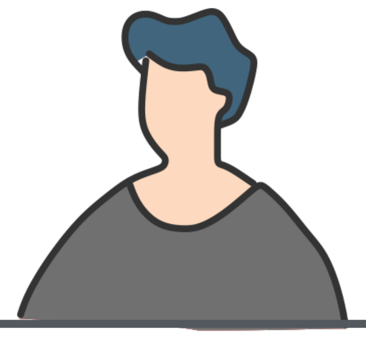
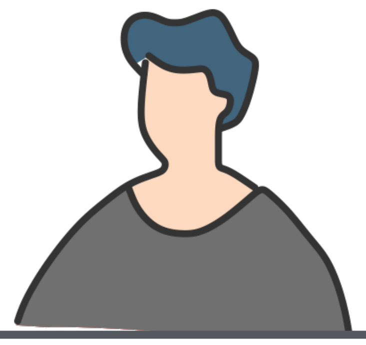
Filter Data



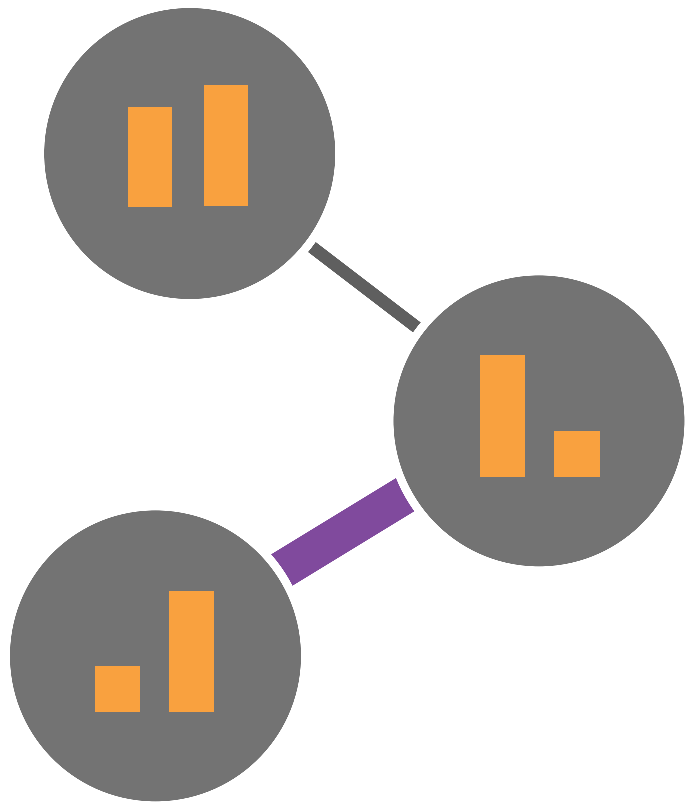

Friends
Friends
Co-workers
Mother
Tom
Abby
Jon
Sue
Mark
Cole
Dating
Married
Soccer Coach

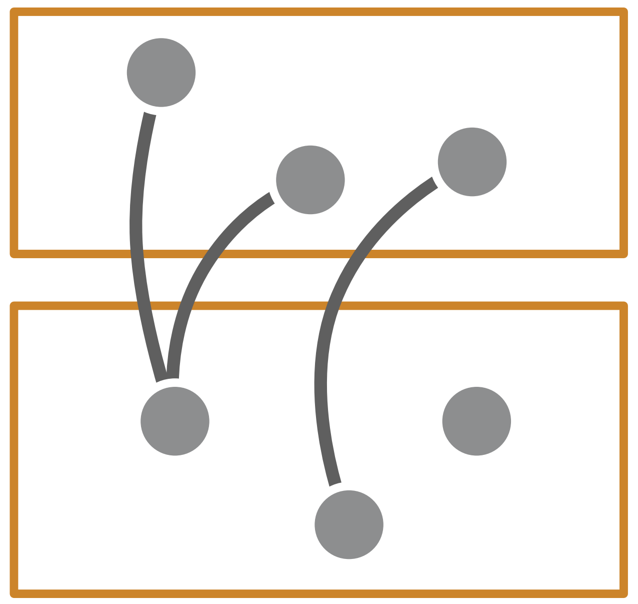


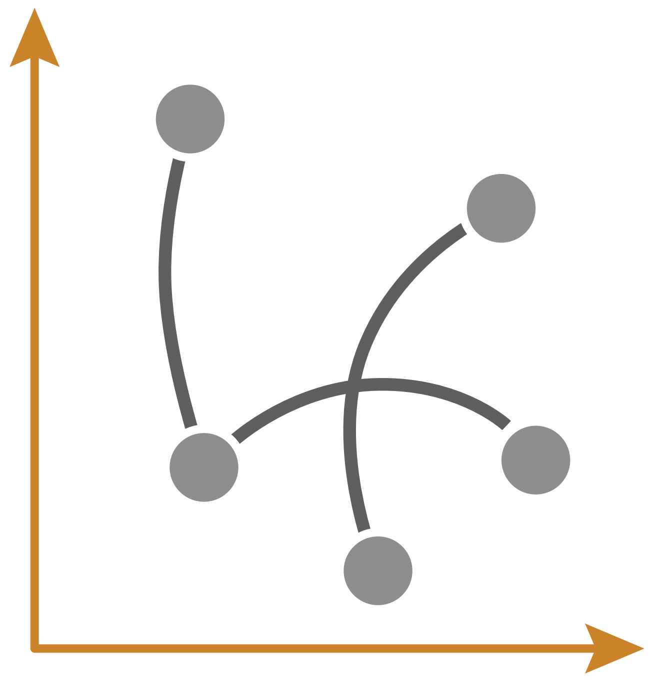
Name	Cole	Tom	Abby	Jon	Sue	Mark
Beverage	Port	Beer	Port	Coke	Coke	Beer
Day 1	1	0	4	3	3	5
Day 2	0	2	5	3	5	5
Day 3	4	1	2	2	4	3

Source	Target	Type	Duration
		Co-workers	3 years
		Soccer Coach	2 years
		Dating	1 year
		Mother / Son	7 years
		Friends	12 years
		Friends	3 years
		Married	6 years

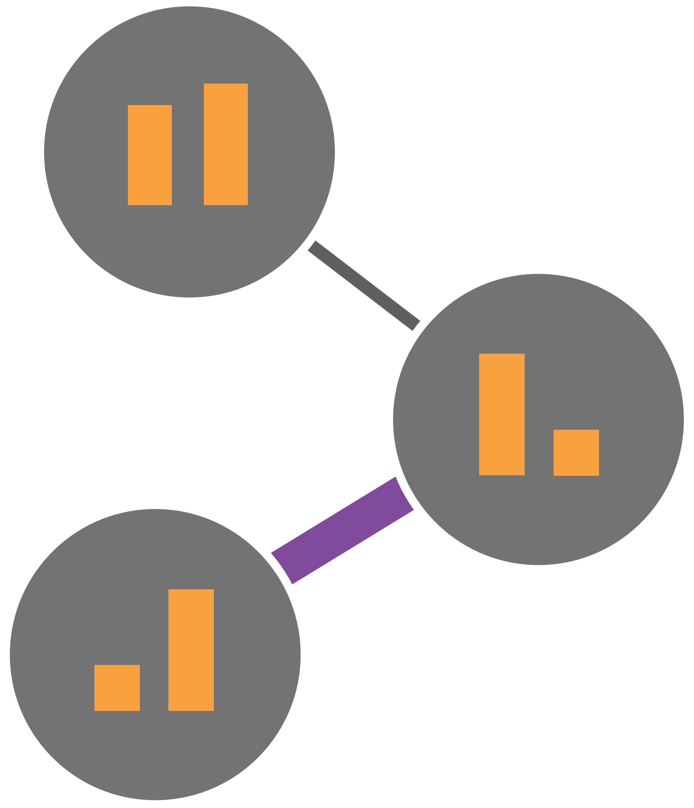
Node-Link Layouts



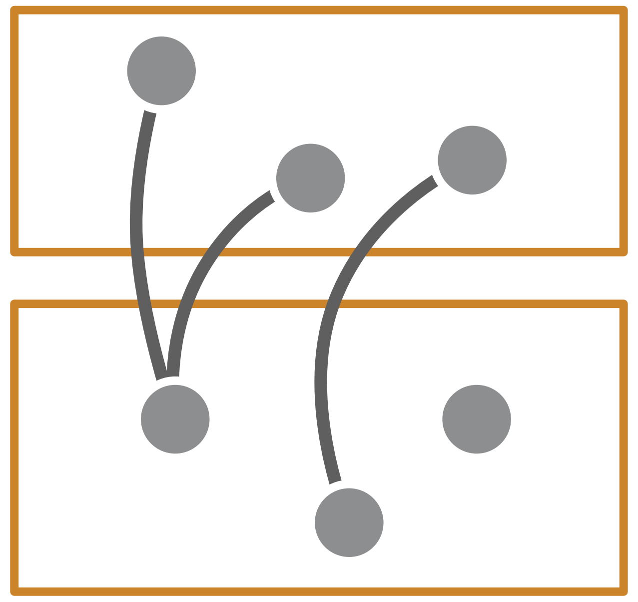


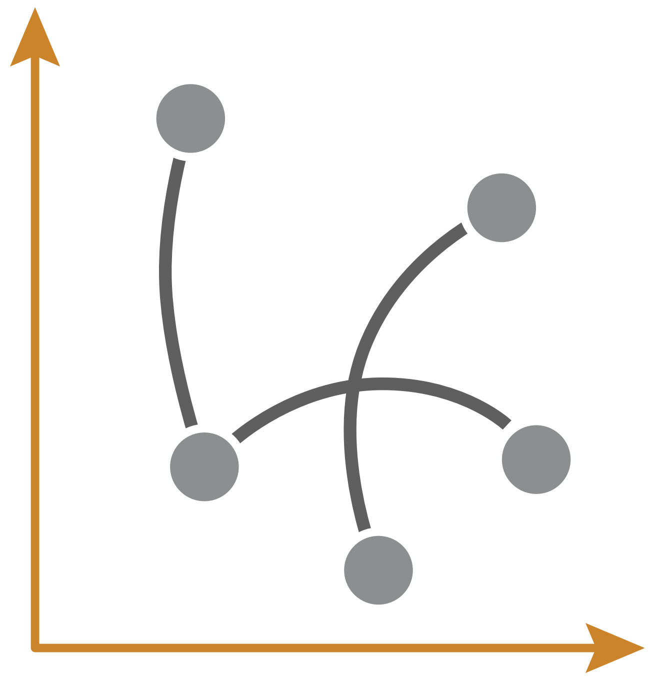


Topology Driven Layout

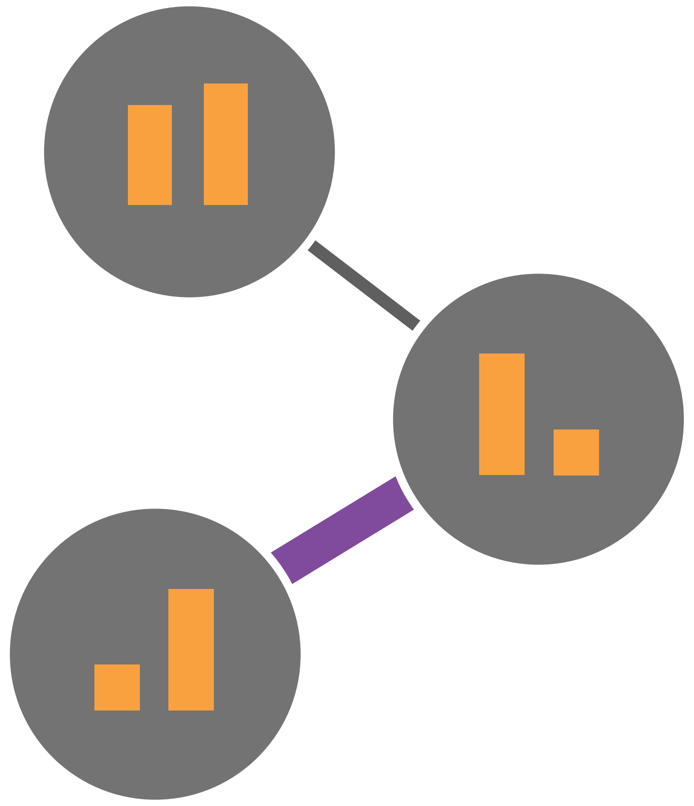


Attribute Driven Layouts



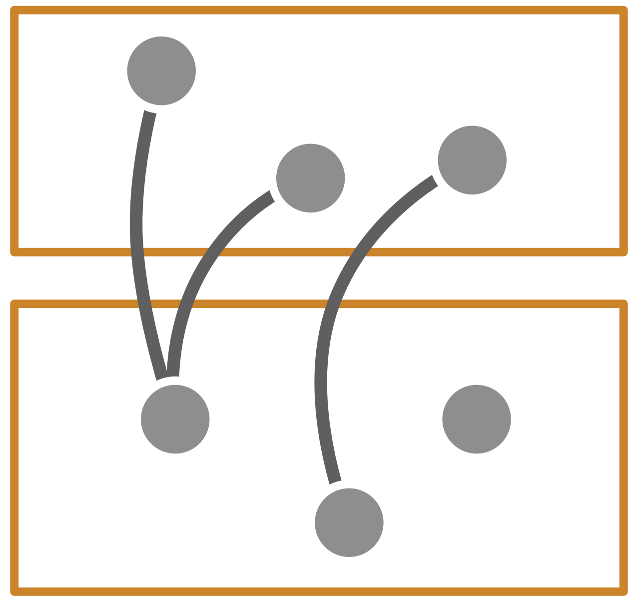


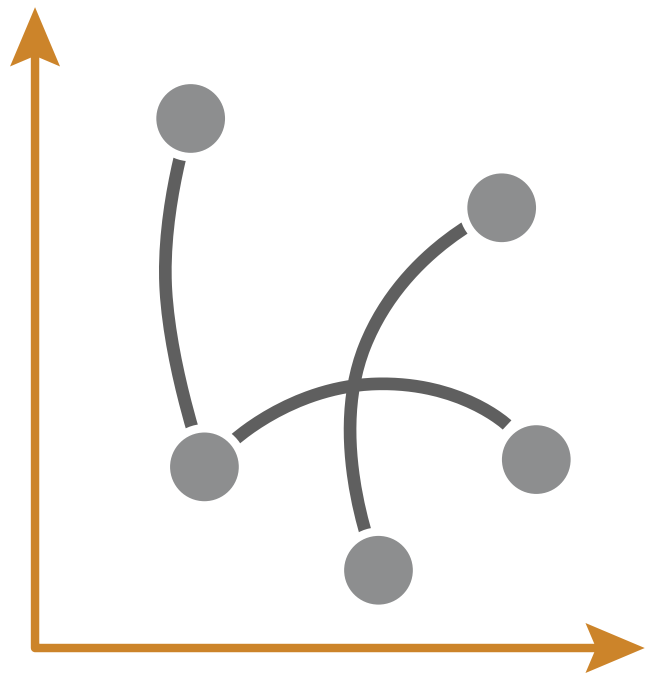
Topology Driven Layout



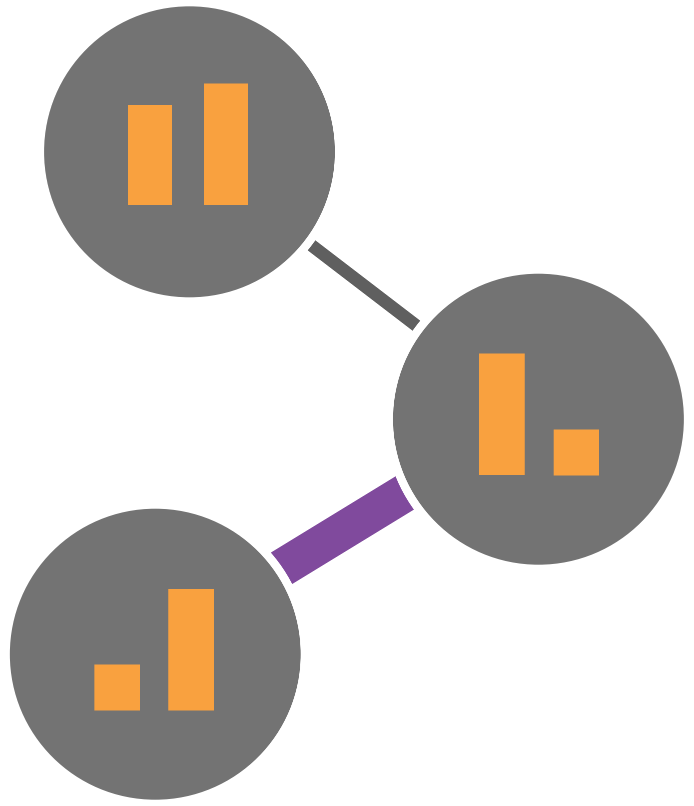
On-Node / On-Edge
Encoding

Attribute Driven Layouts



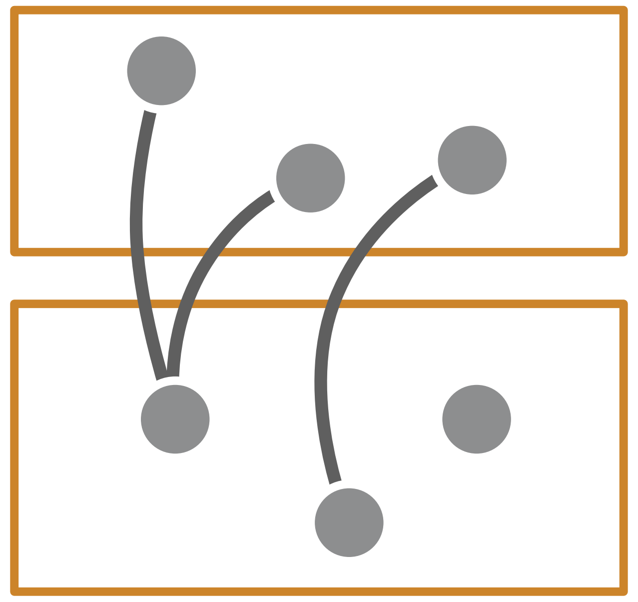


Topology Driven Layout

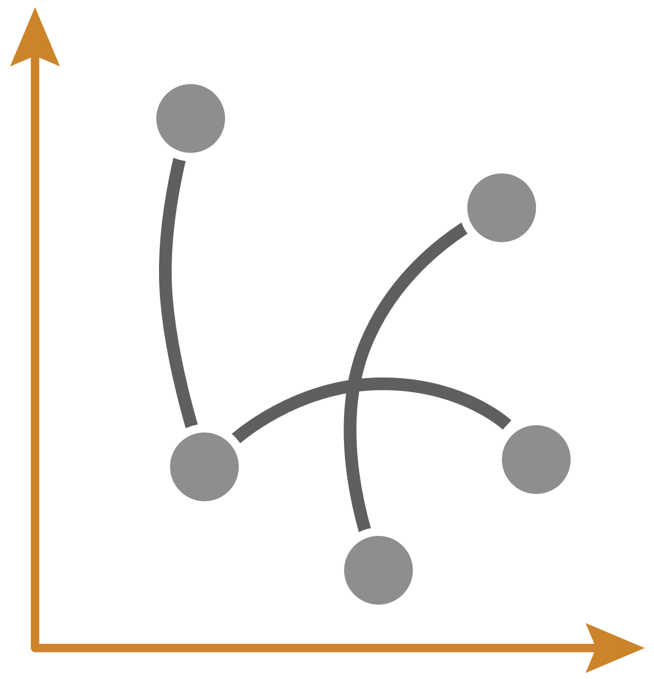


On-Node / On-Edge
Encoding

Attribute Driven Layouts

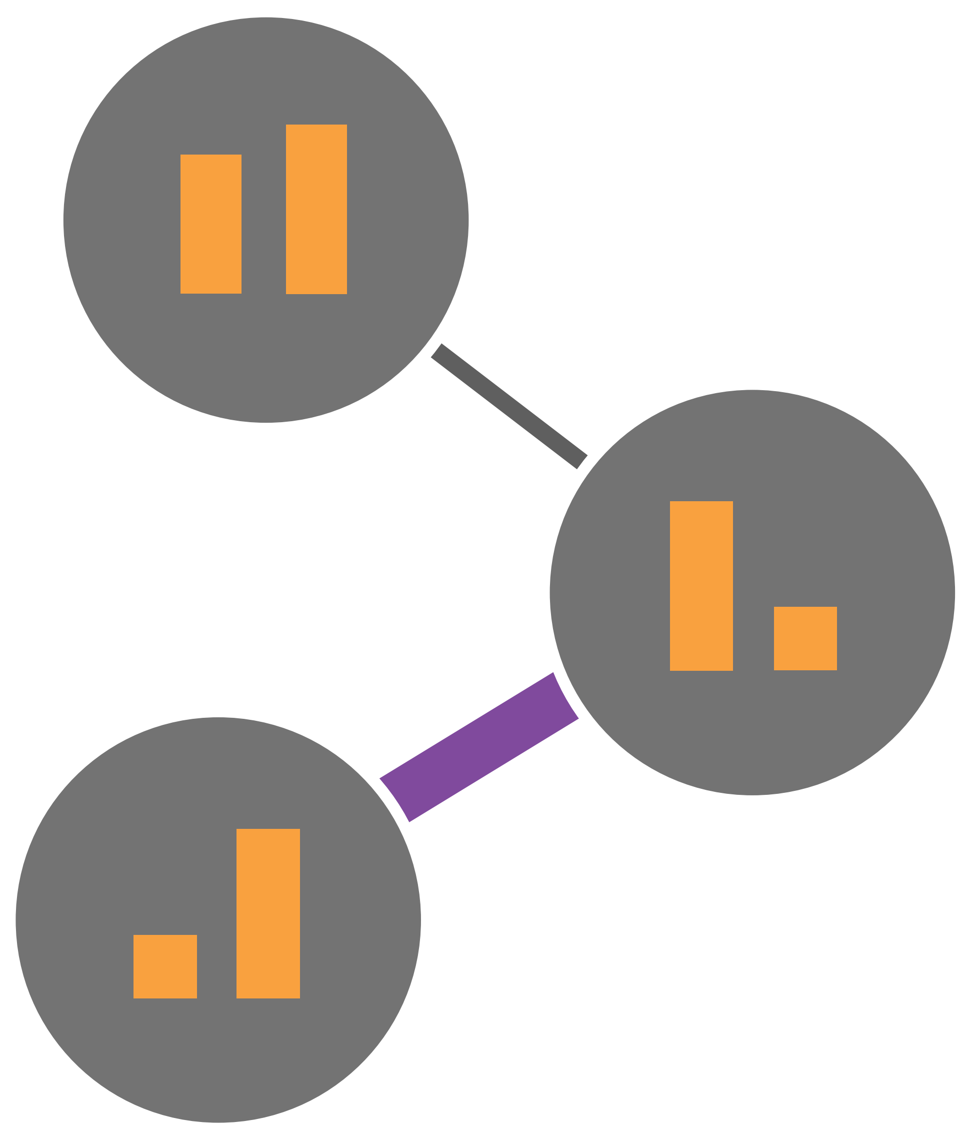


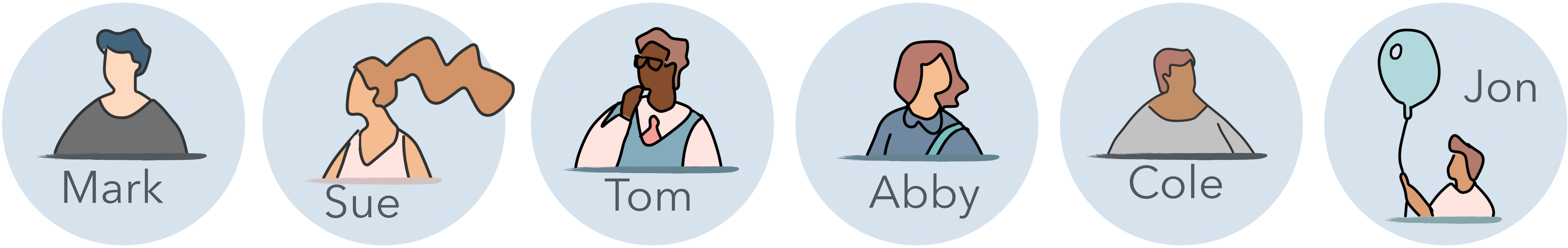
Attribute-Driven
Faceting

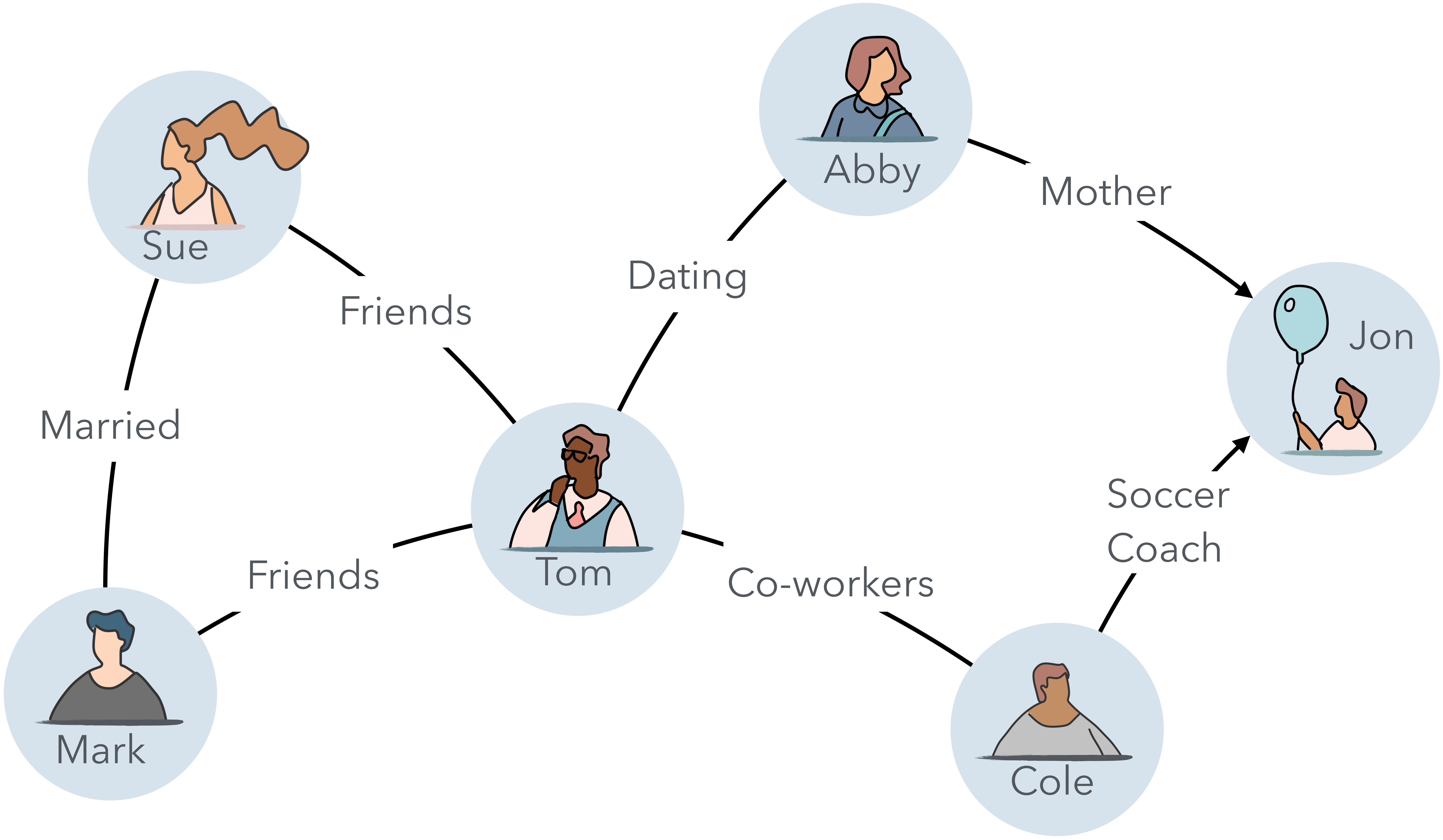


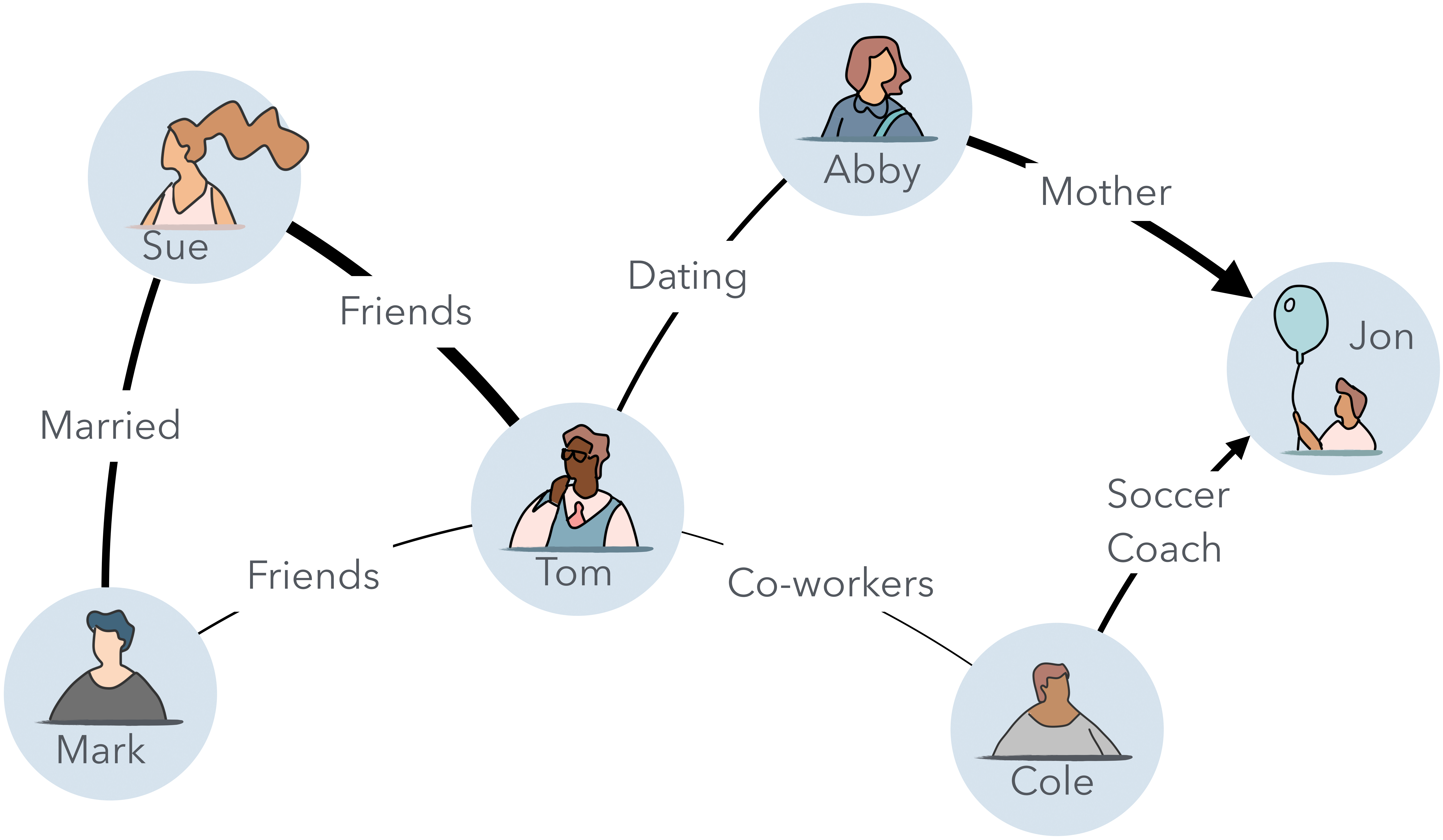
Attribute-Driven
Positioning

On-Node / On-Edge Encoding

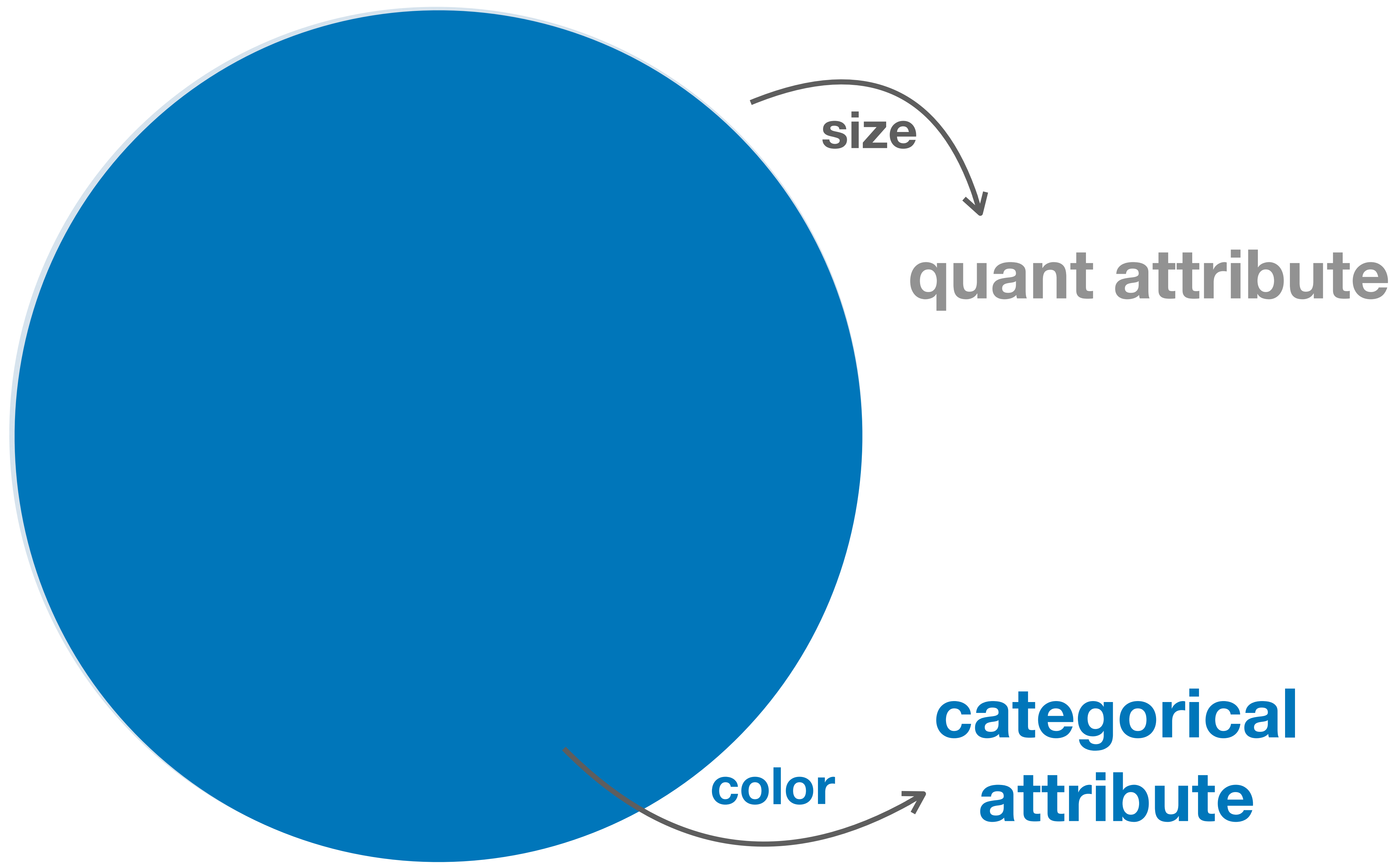


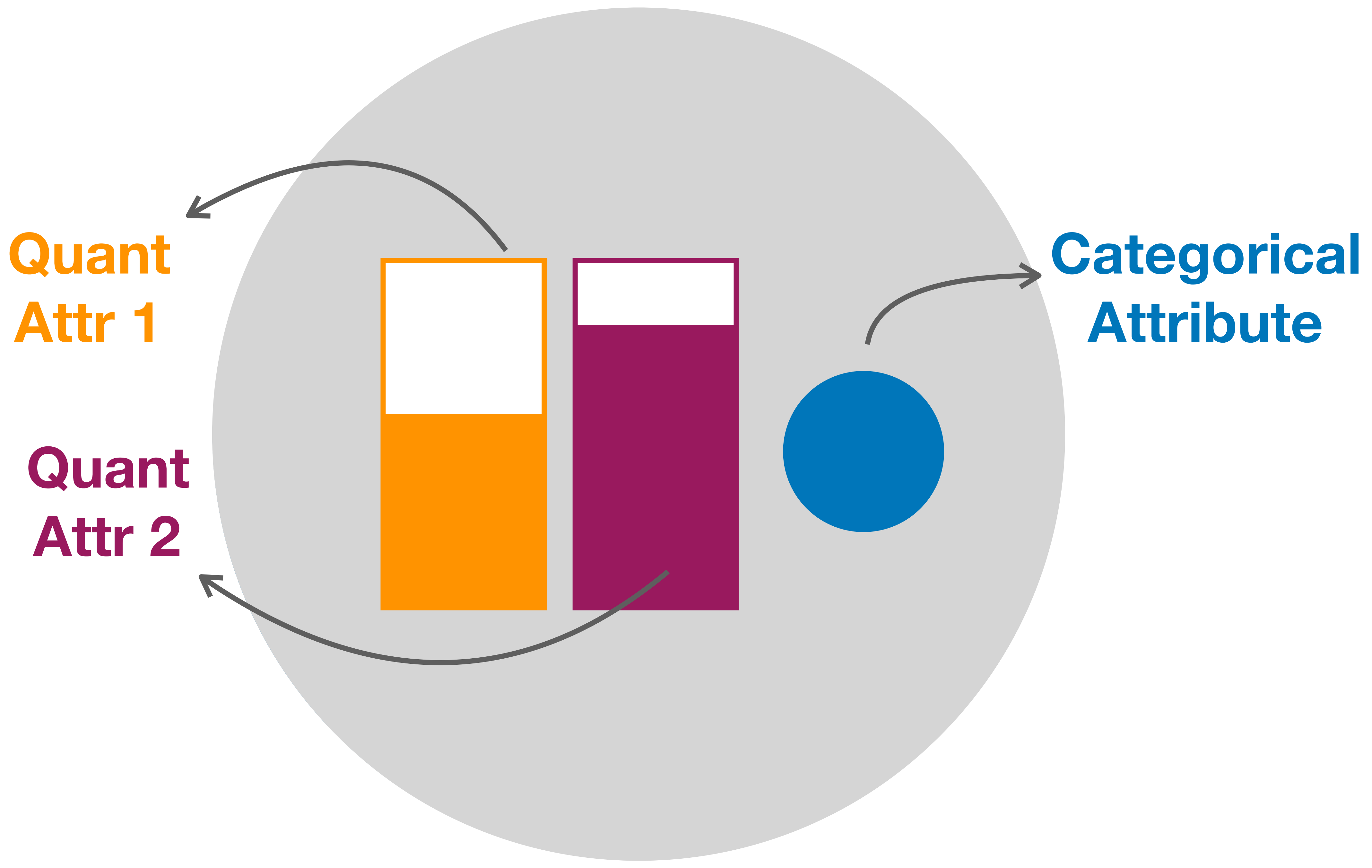

Mark
Sue
Tom
Abby
Cole
Jon

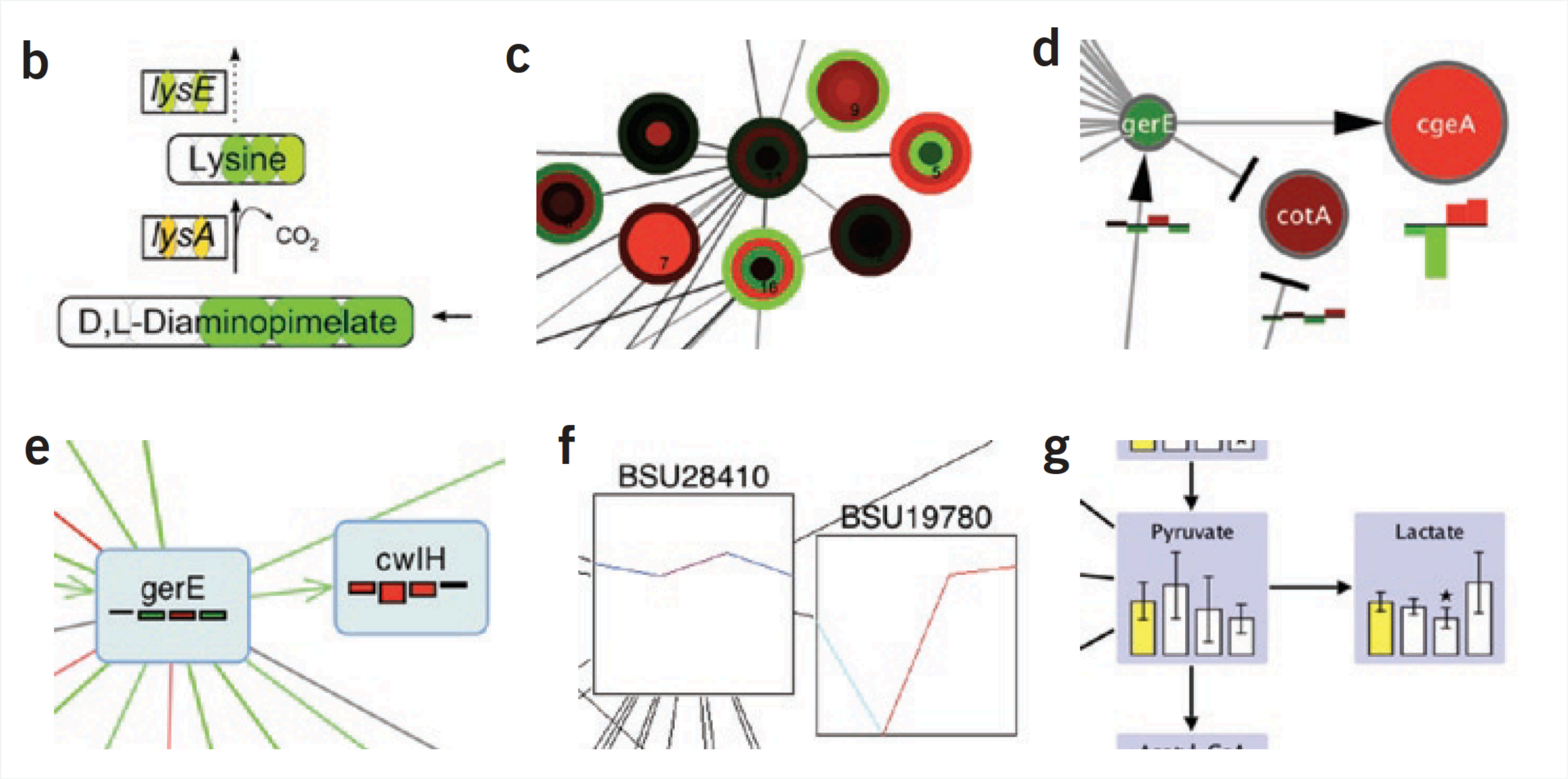

Sue
Abby
Friends
Dating
Mother
Jon
Married
Tom
Soccer
Coach
Friends
Co-workers
Mark
Cole



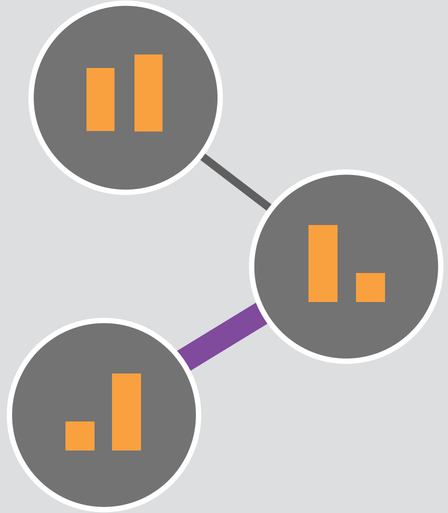
Sue
Friends
Abby
Mother
Dating
Jon
Married
Tom
Soccer
Coach
Friends
Co-workers
Mark
Cole


size
quant attribute
color
categorical attribute

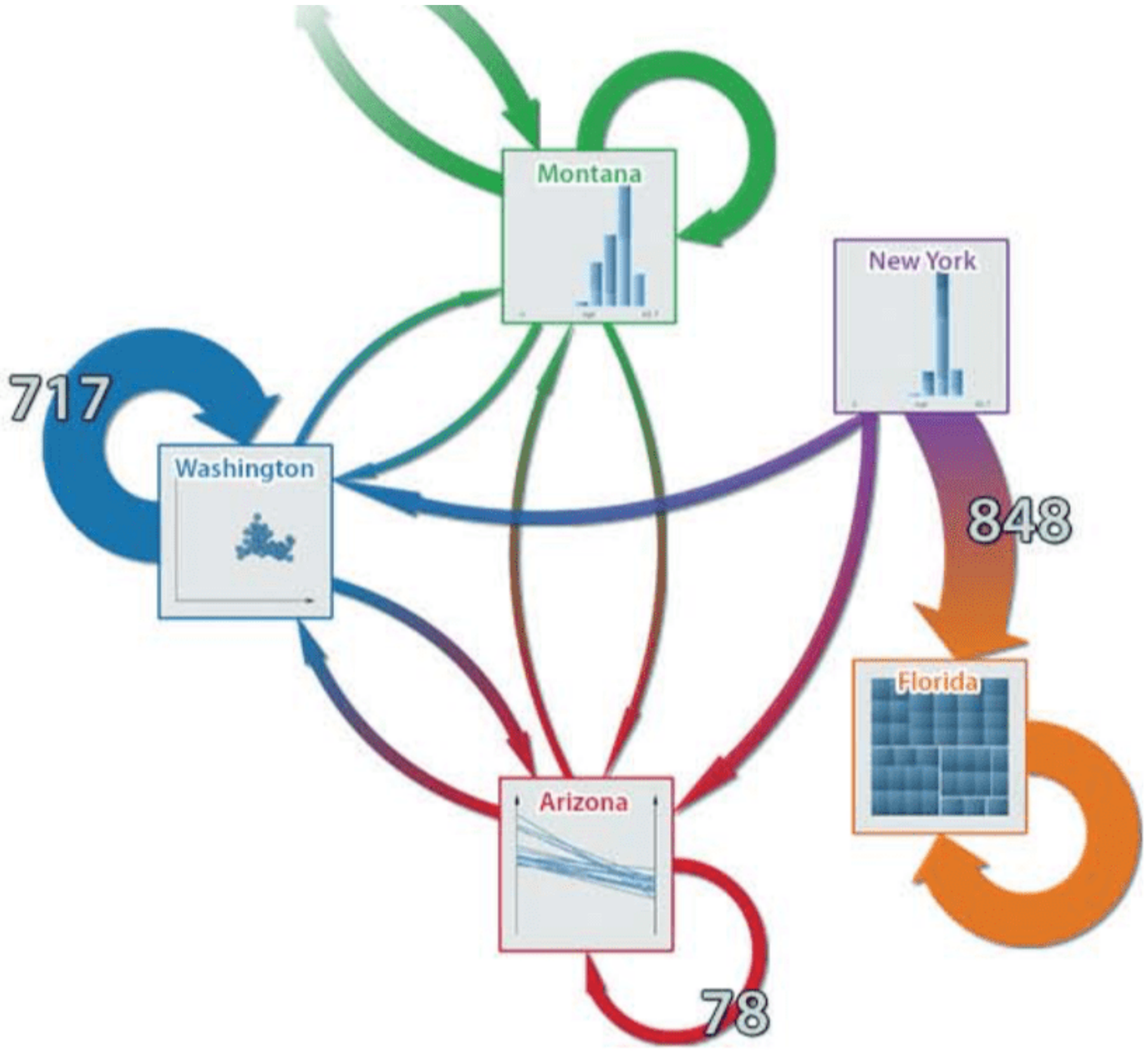

Quant
Attr 1
Quant
Attr 2
Categorical
Attribute



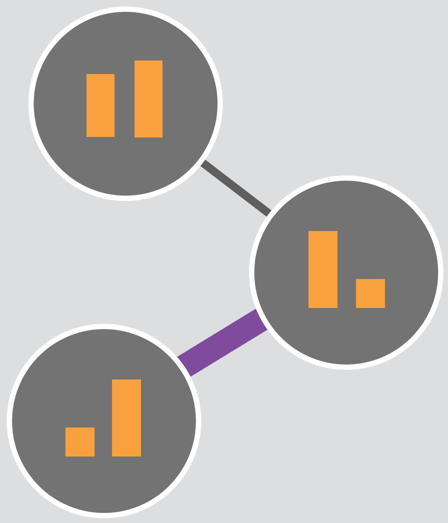
Gehlenborg et al. 2010



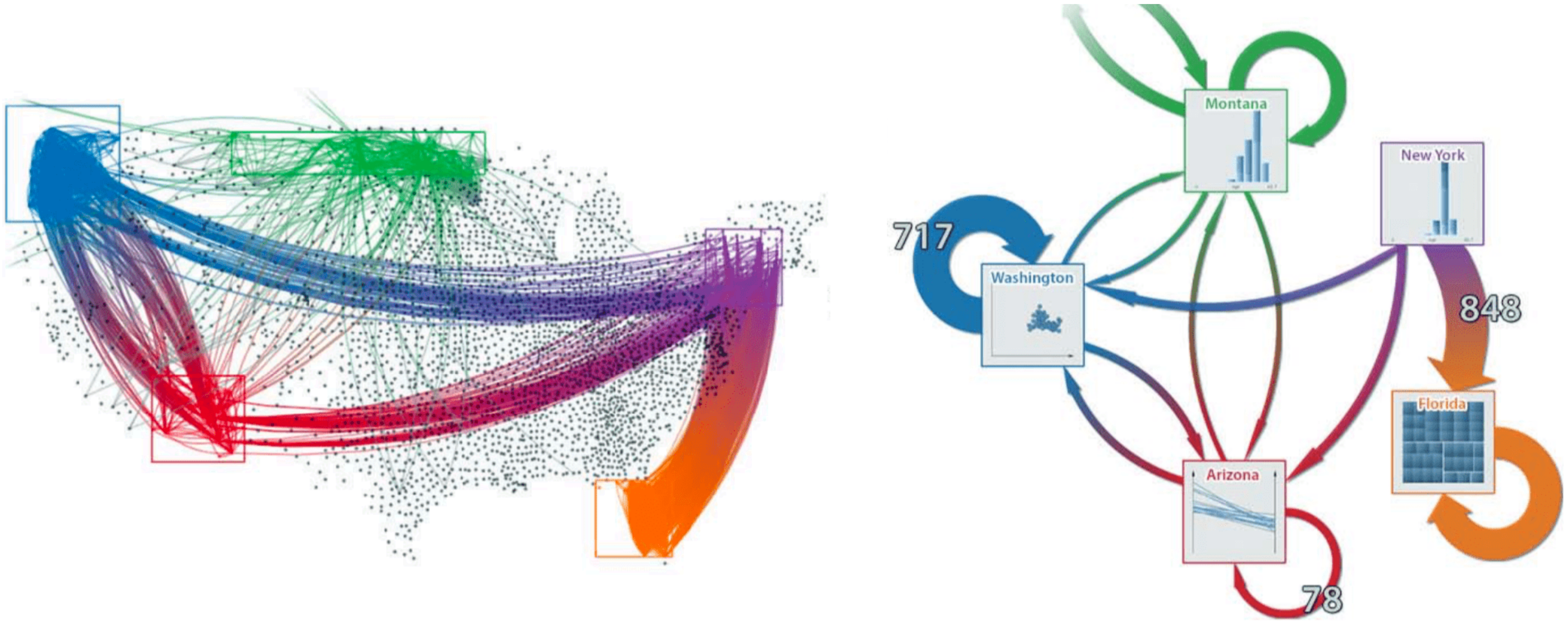
On-Node / On-Edge Encoding



Elzen and Wijk, 2014

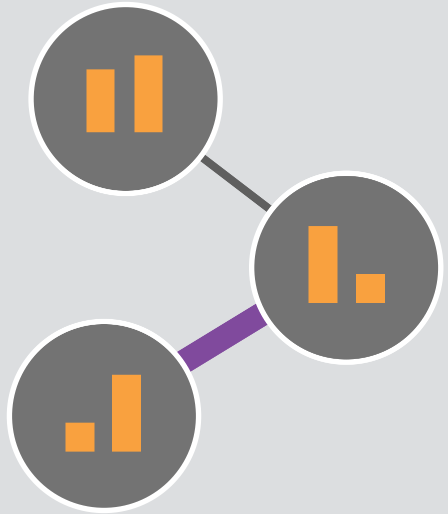


On-Node / On-Edge Encoding

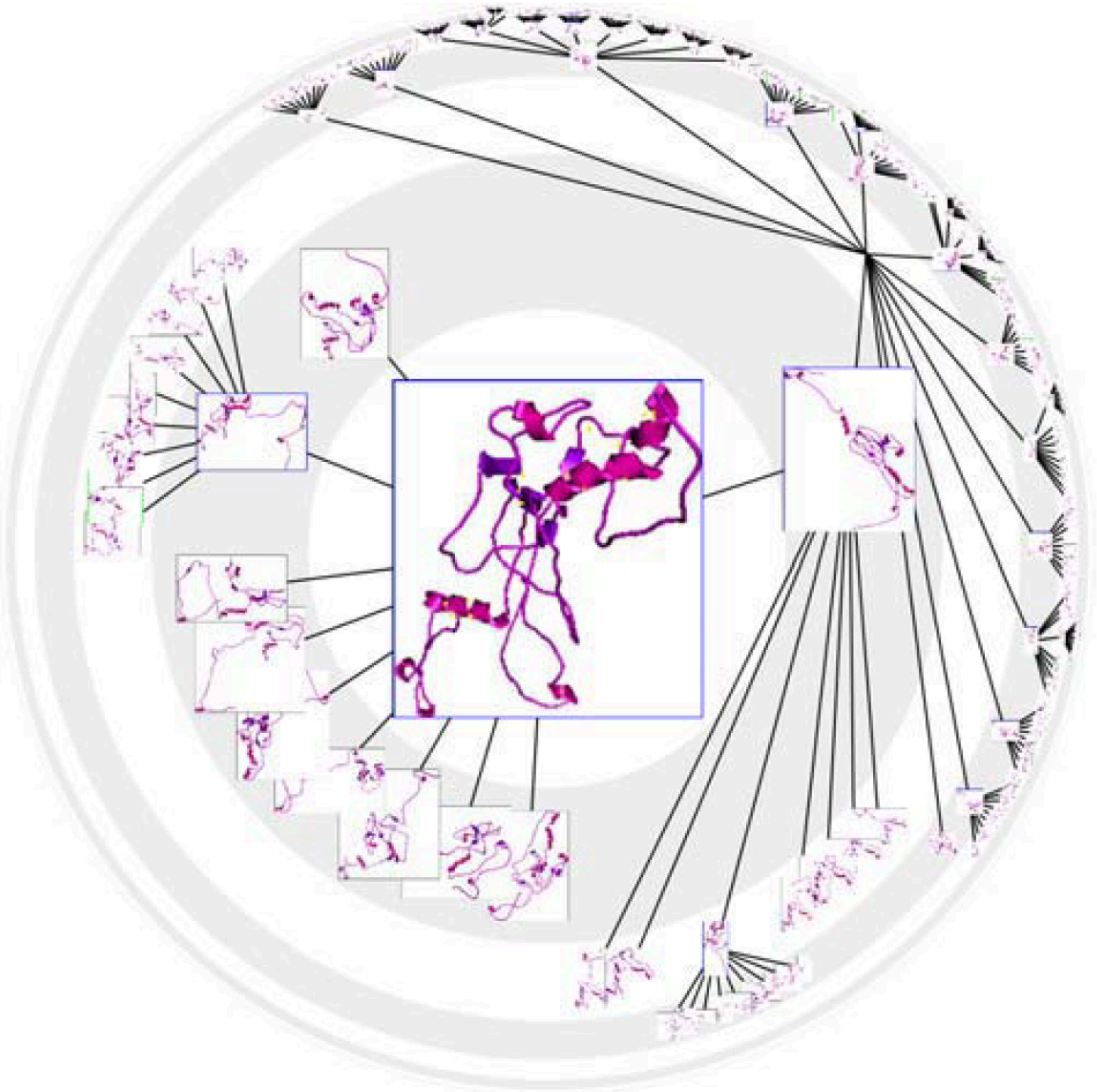


Elzen and Wijk, 2014

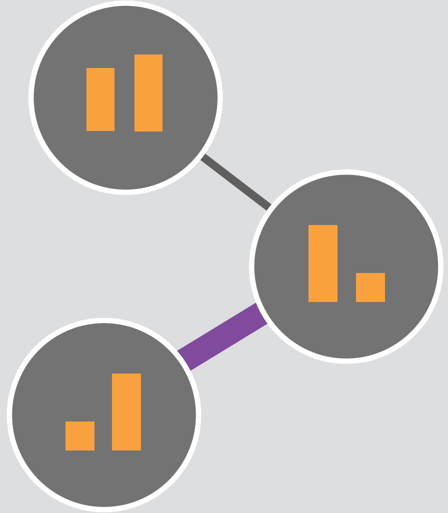
Aggregating Nodes/Edges



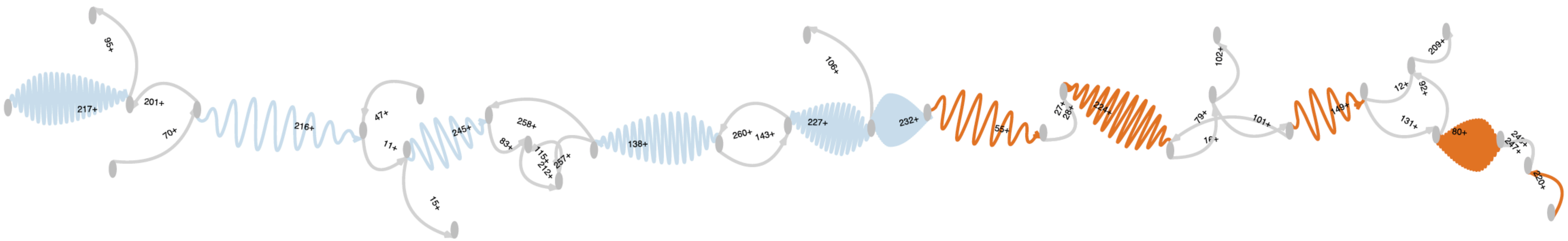
On-Node / On-Edge Encoding



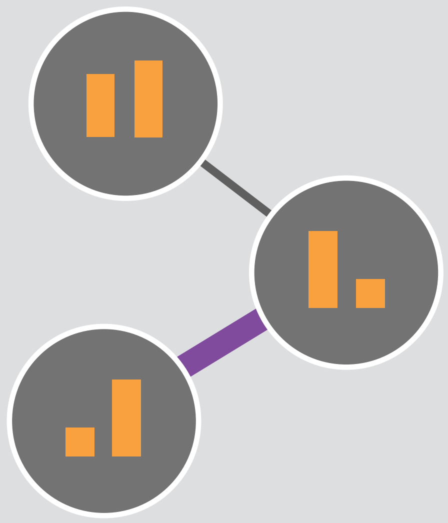
Jankun-Kelly and Ma, 2003



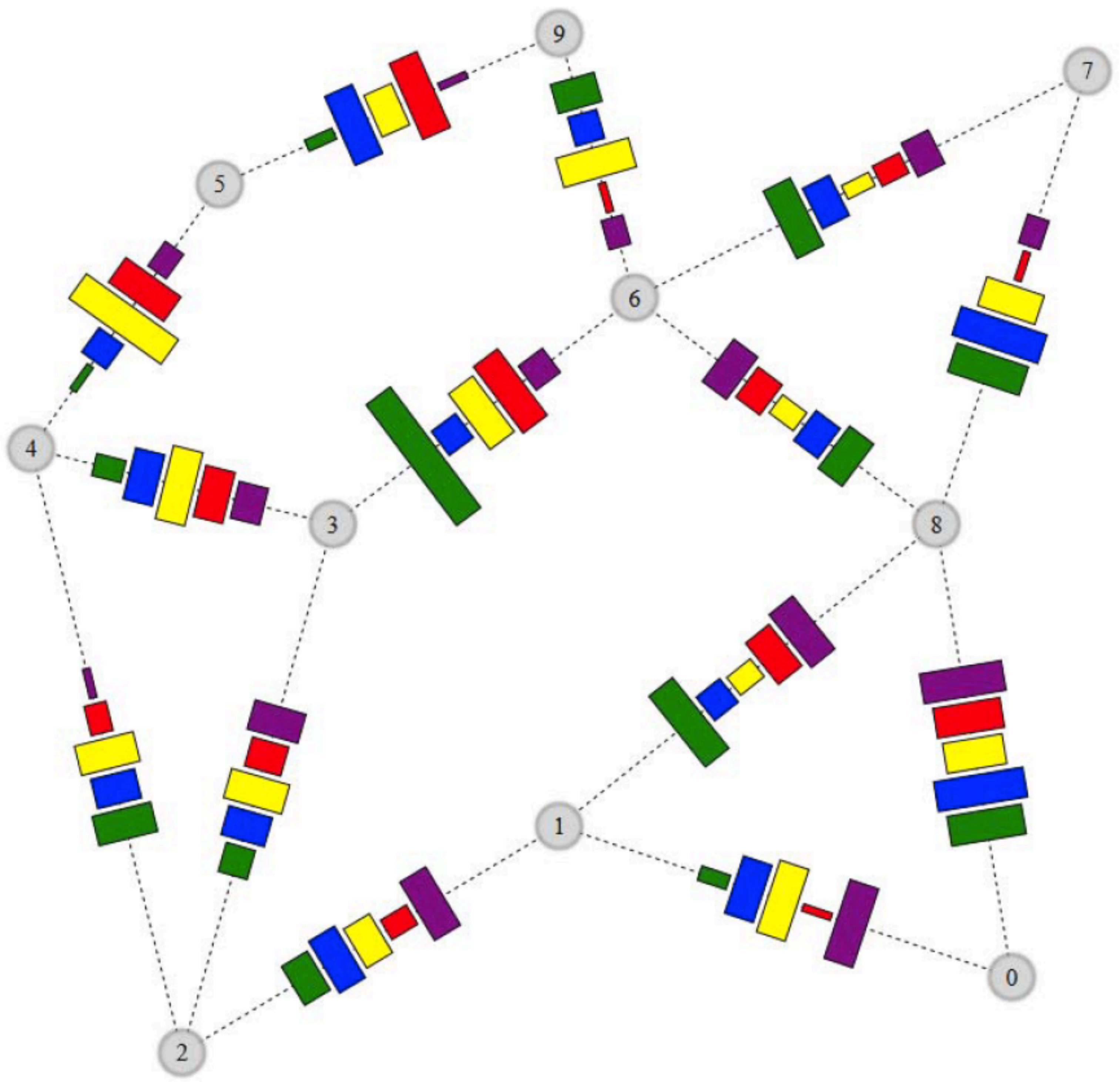
On-Node / On-Edge
Encoding

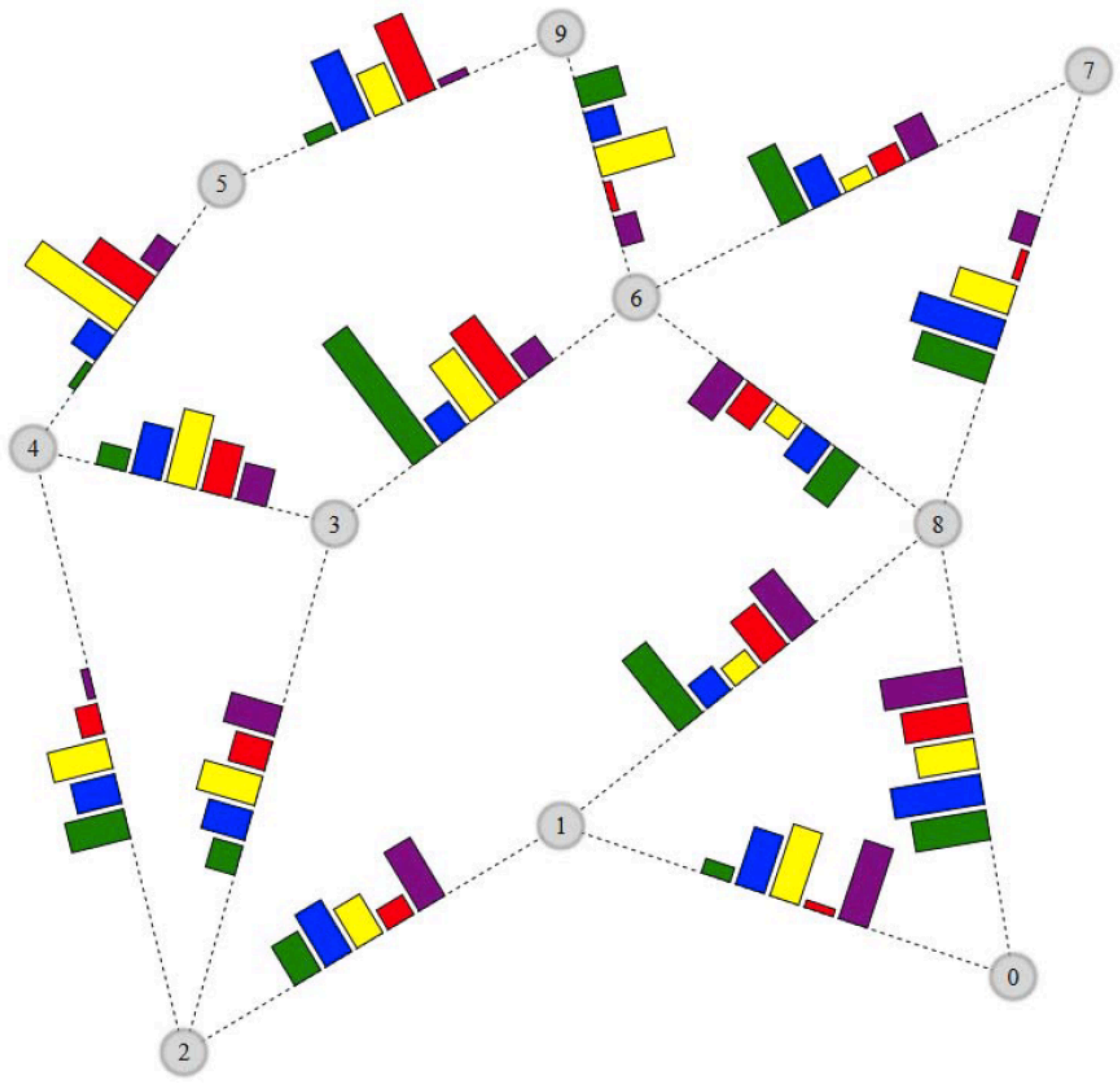


Nielsen, 2009

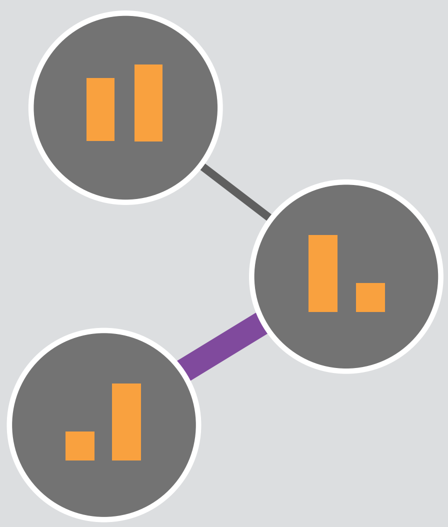


On-Node / On-Edge Encoding

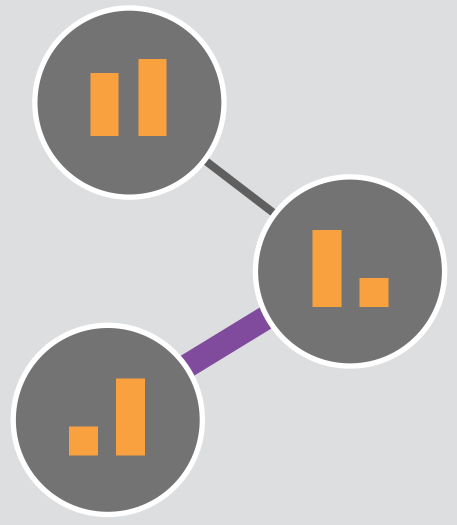




Schöffel et al, 2016



On-Node / On-Edge Encoding



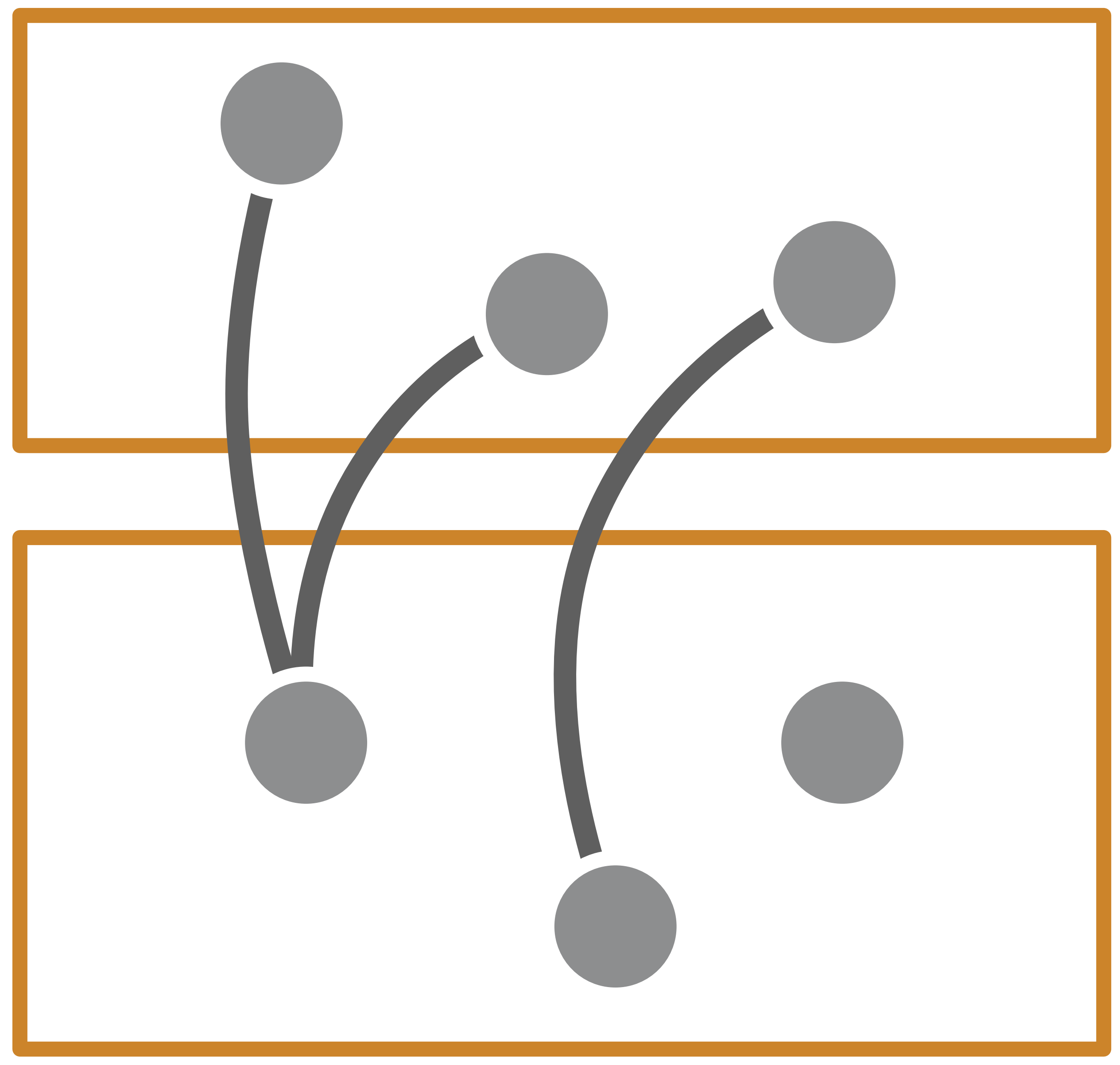
On-Node / On-Edge Encoding

Is easily understood by most users
Works well for all types of networks

Scalability.
Node size leaves little space to encode attributes.

Recommended for small networks when only a few (usually under five) attributes on the nodes are shown, or in combination with a zooming/filtering strategy

Attribute-Driven Faceting

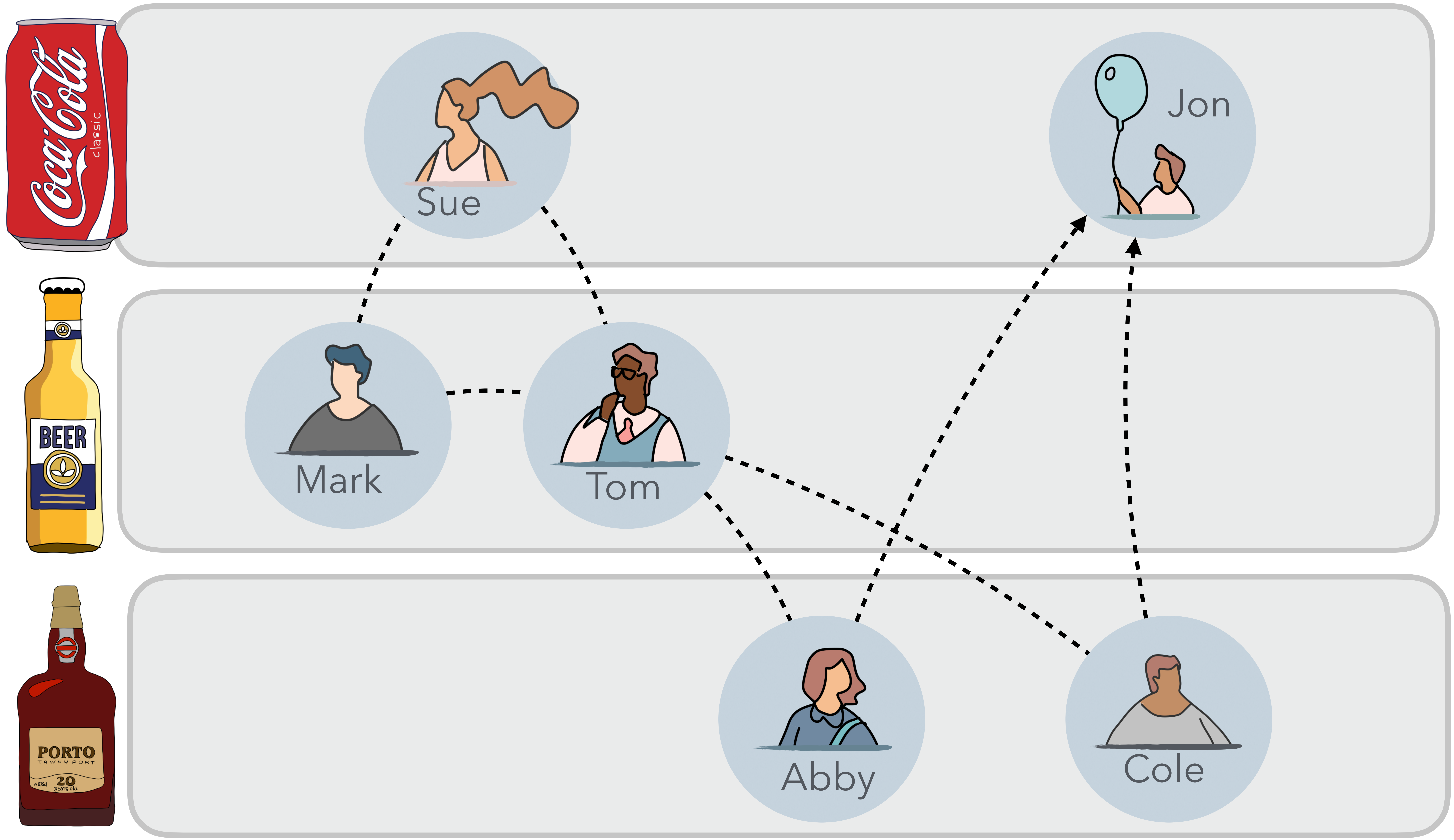


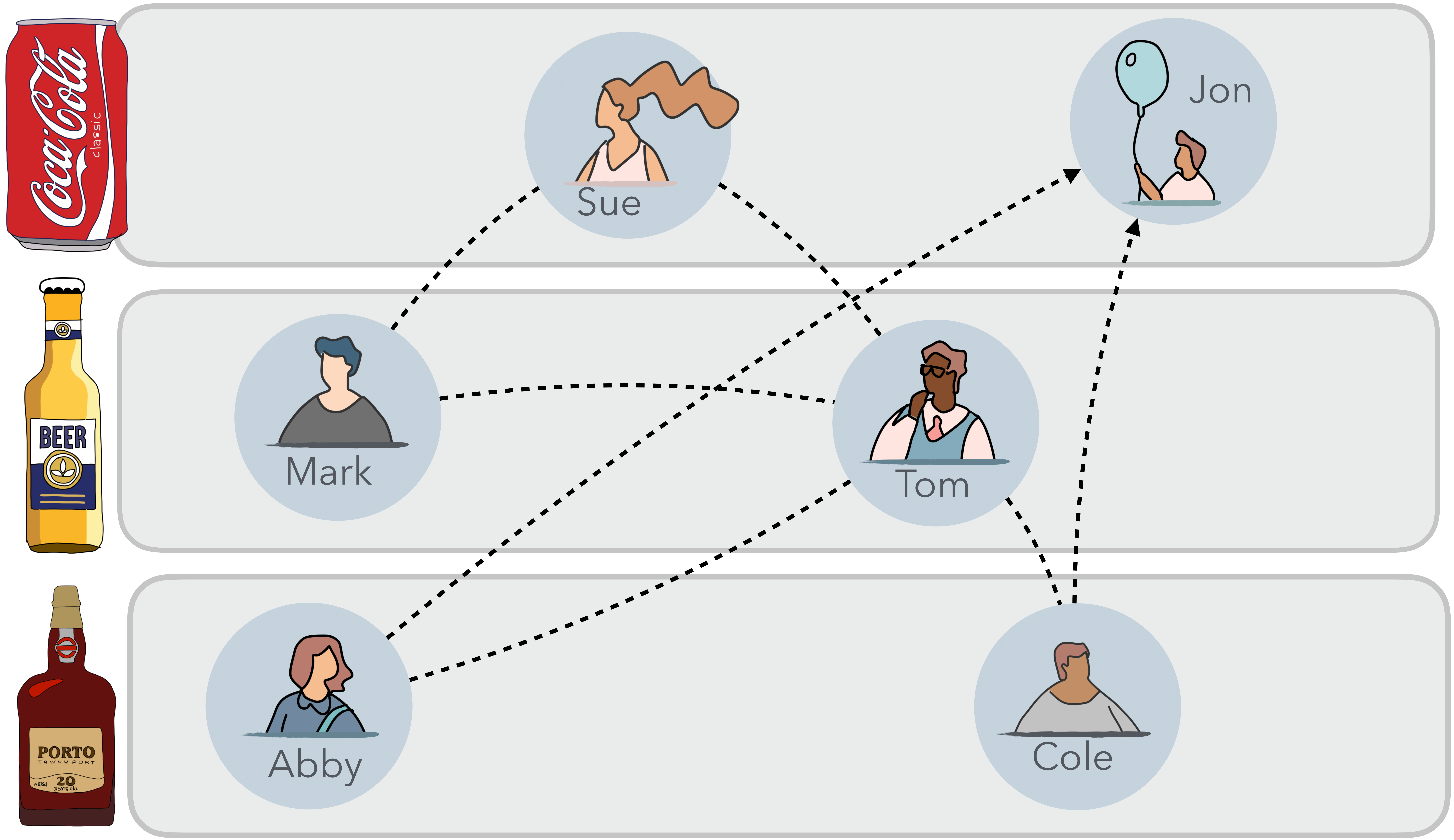

Coca-Cola
classic
BEER
PORTO
TAWNY PORT
e 0,75cl
20
years old


Coca-Cola
classic


BEER

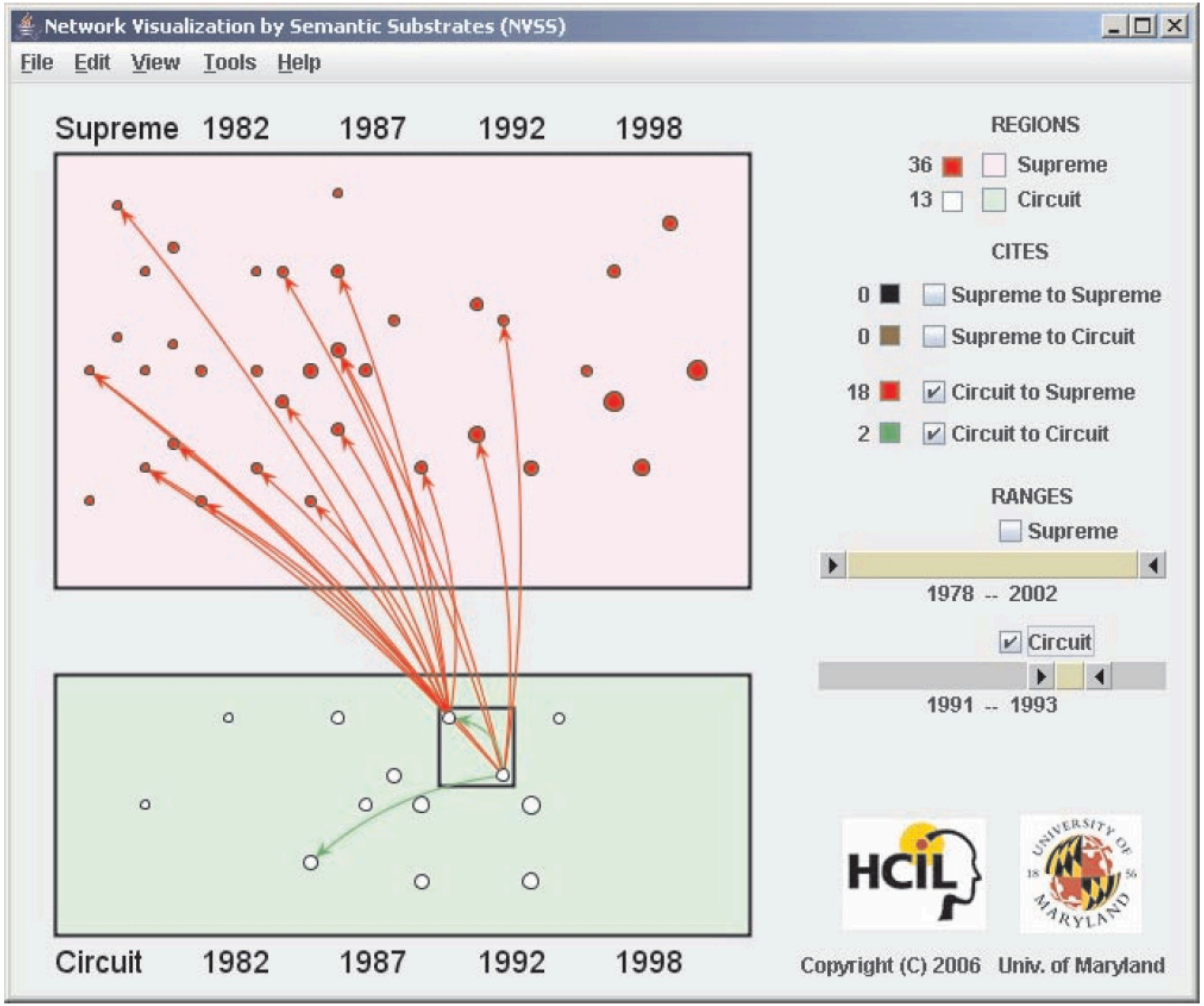

PORTO
TAWNY PORT
20
years old

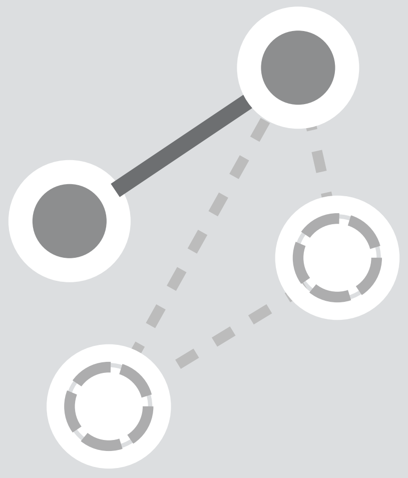

Coca-Cola classic
BEER
PORTO
TAWNY PORT
20
years old
Sue
Jon
Mark
Tom
Abby
Cole



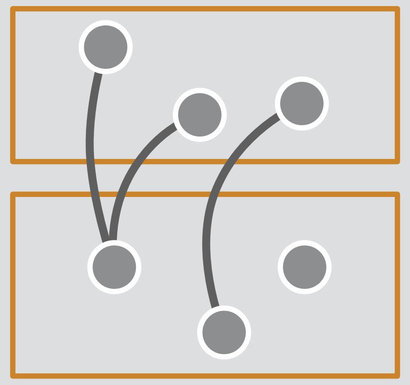
Coca-Cola classic
Sue
Jon
BEER
Mark
Tom
PORTO
TAWNY PORT
20
years old
Abby
Cole

Semantic Substrates *Shneiderman and Aris, 2006*

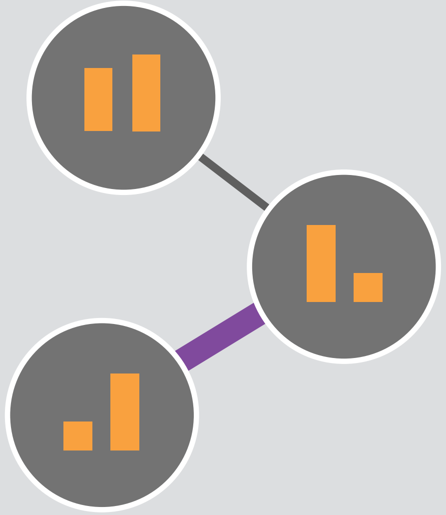




Querying and Filtering

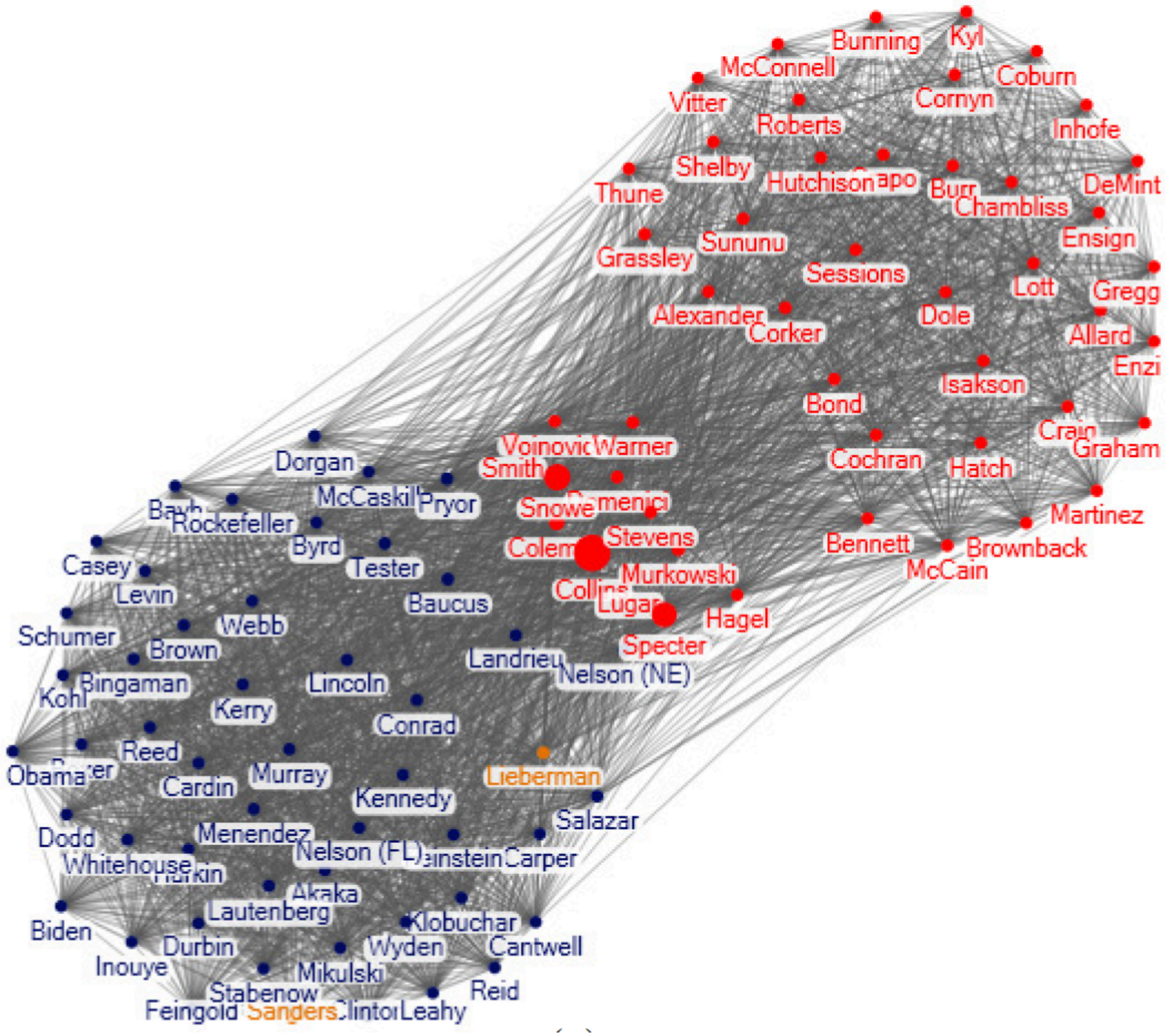


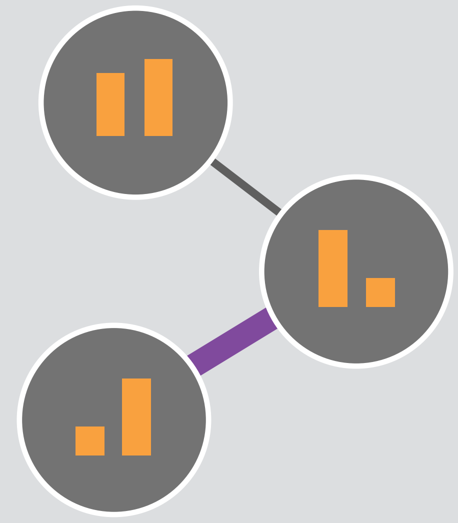
Attribute-Driven Faceting



On-Node / On-Edge Encoding

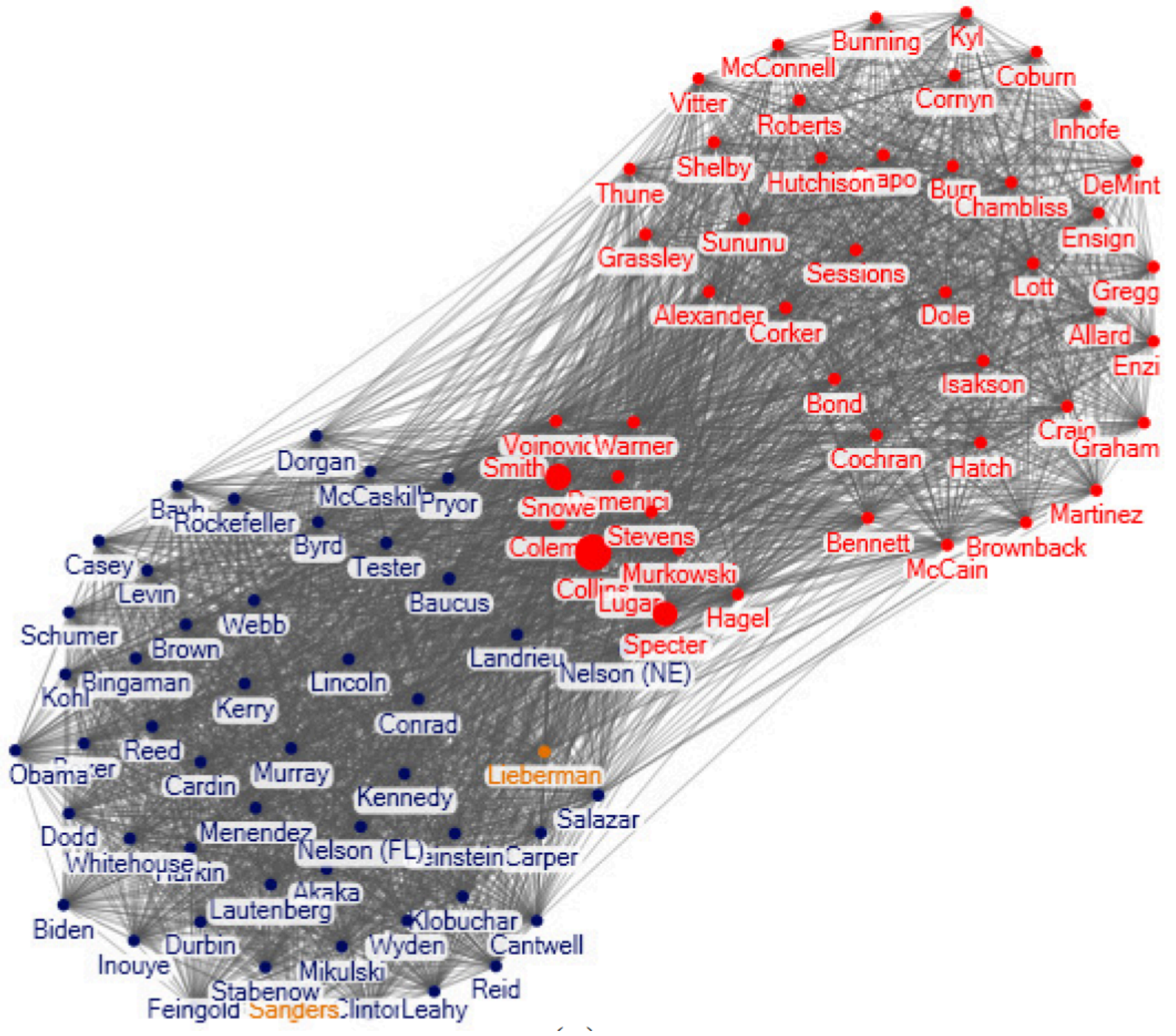
Group-in-a-box *Rodrigues et al. 2011*

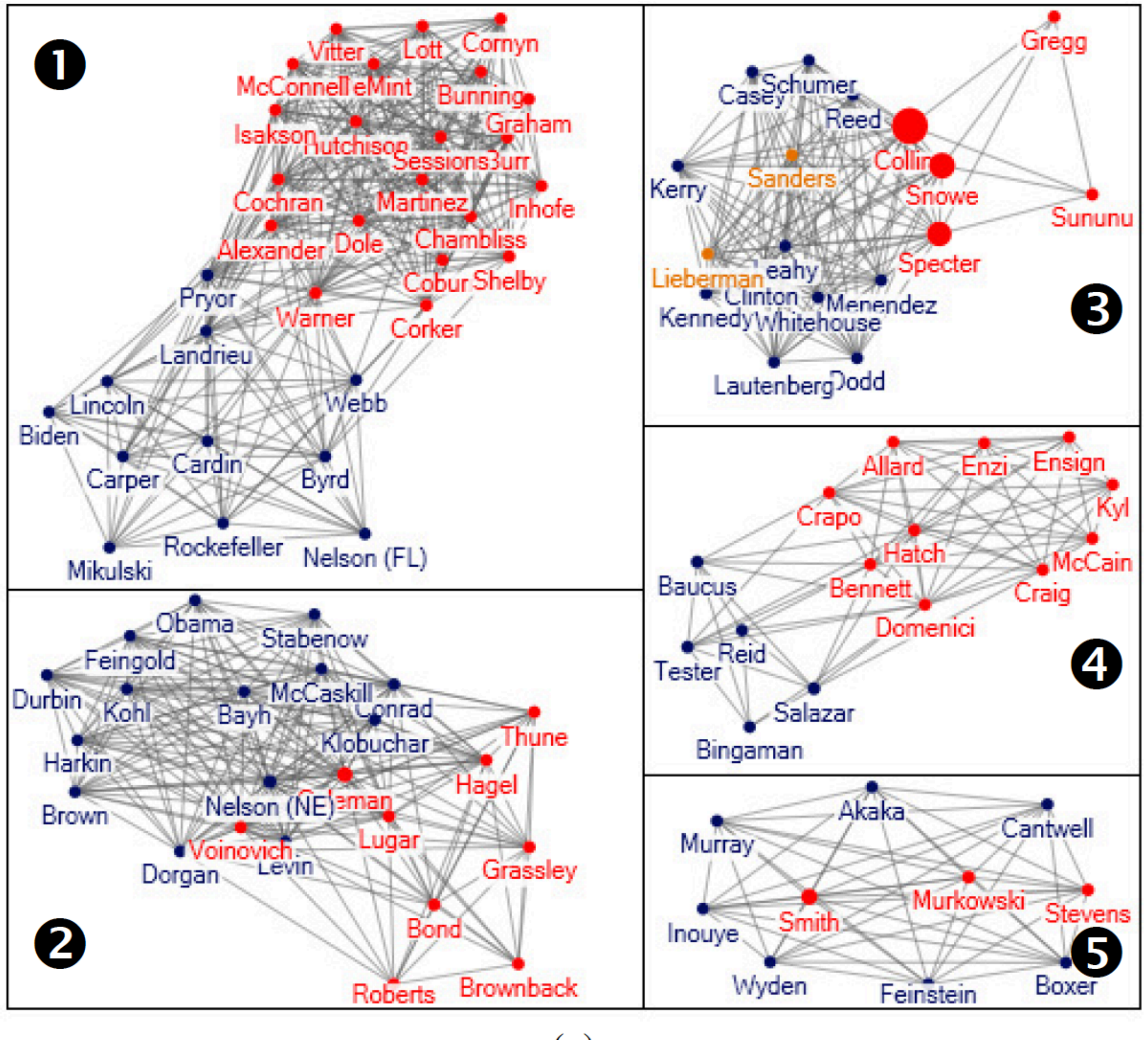


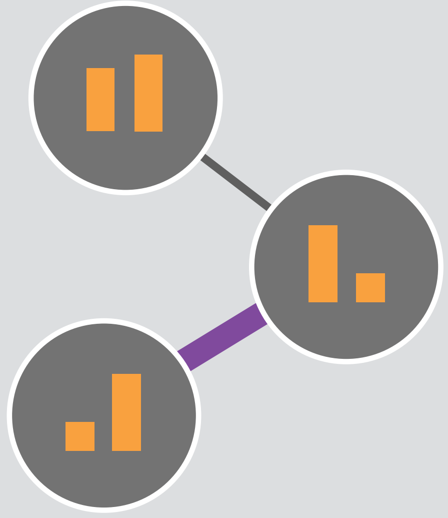


On-Node / On-Edge Encoding

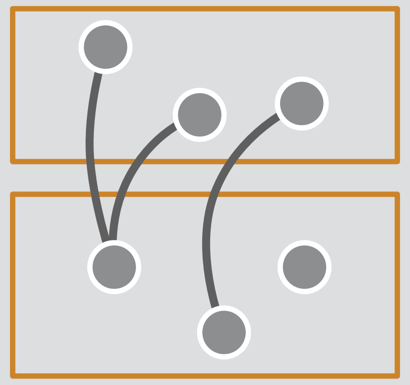
Group-in-a-box *Rodrigues et al. 2011*





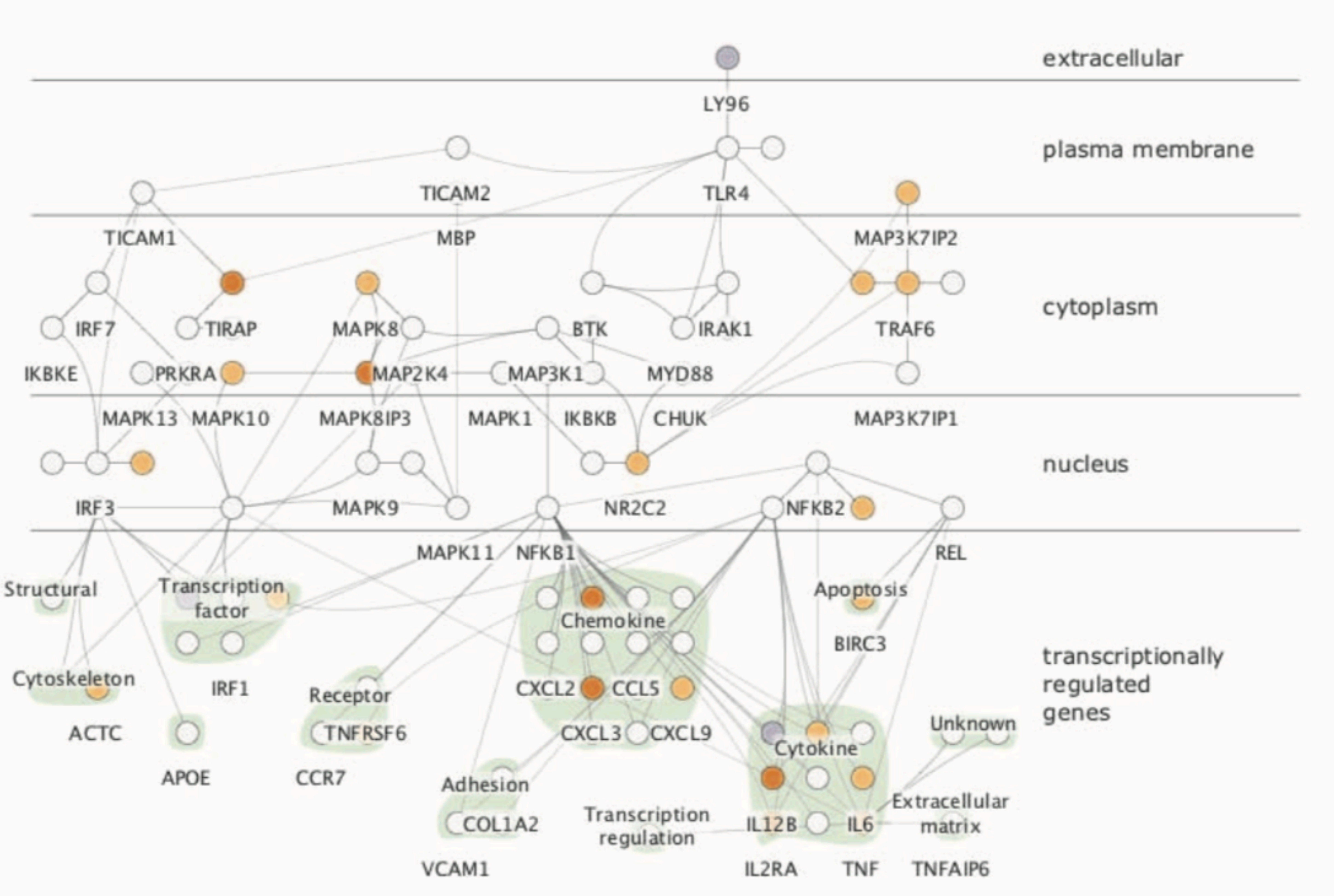


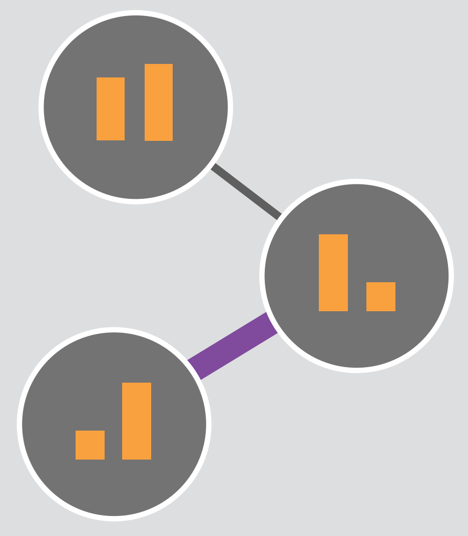
On-Node / On-Edge Encoding



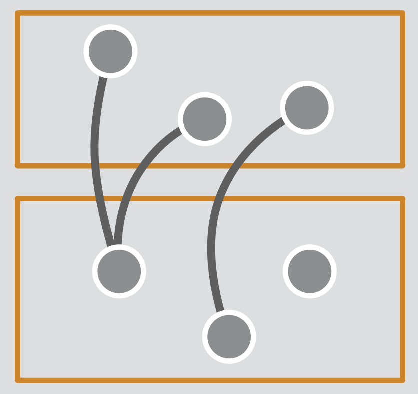
Attribute-Driven Faceting

Cerebral *Barskey et al. 2008*

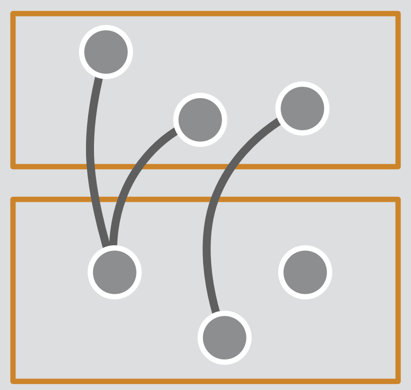




On-Node / On-Edge Encoding



Attribute-Driven Faceting



Attribute-Driven Faceting

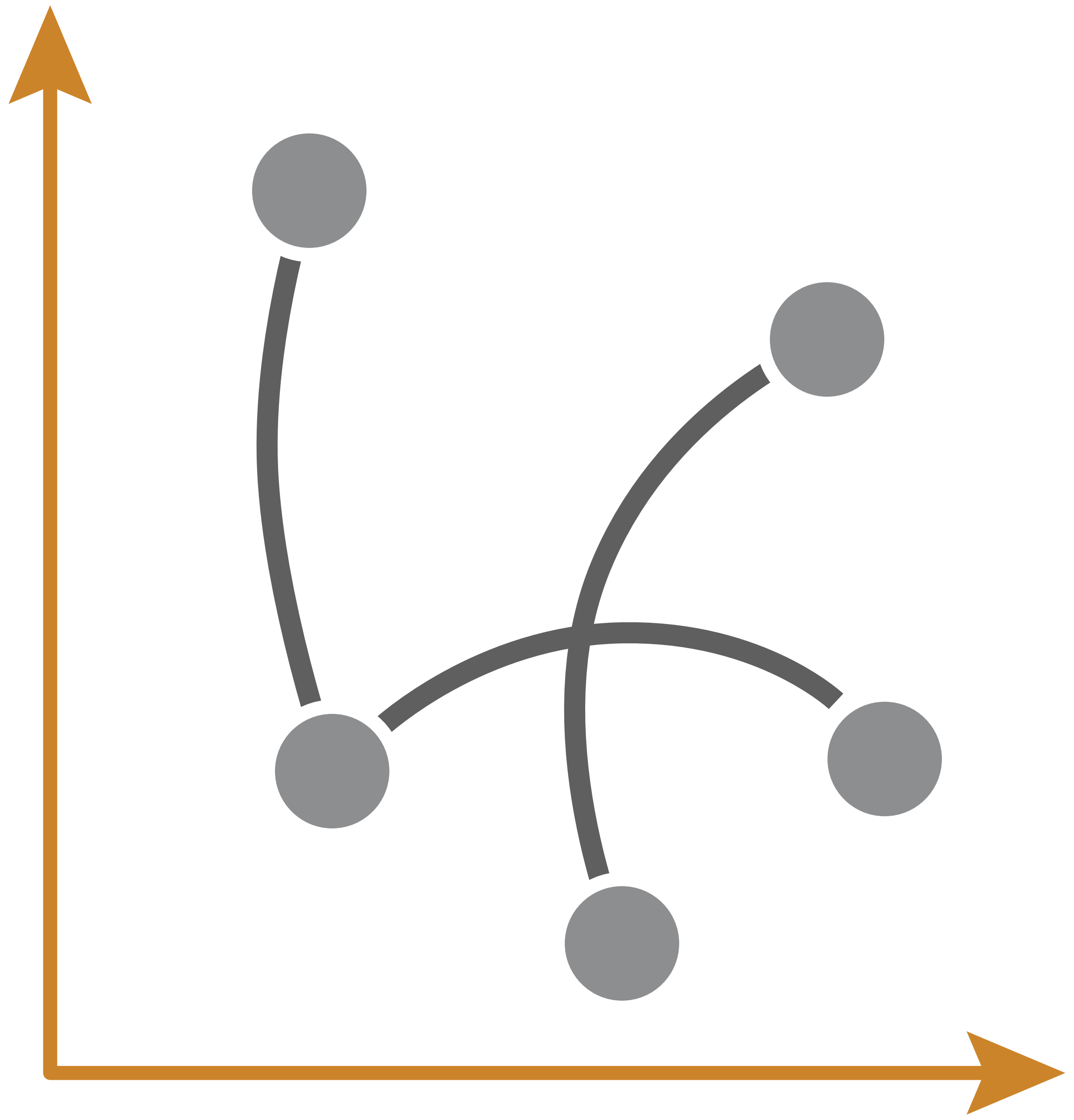
Well suited for networks with different node types or with an important categorical or set-like attribute.

Less scalable with respect to the number of nodes and network density than node-link layouts.

Neighborhoods, paths, and clusters are not easily visible if they span different facets.

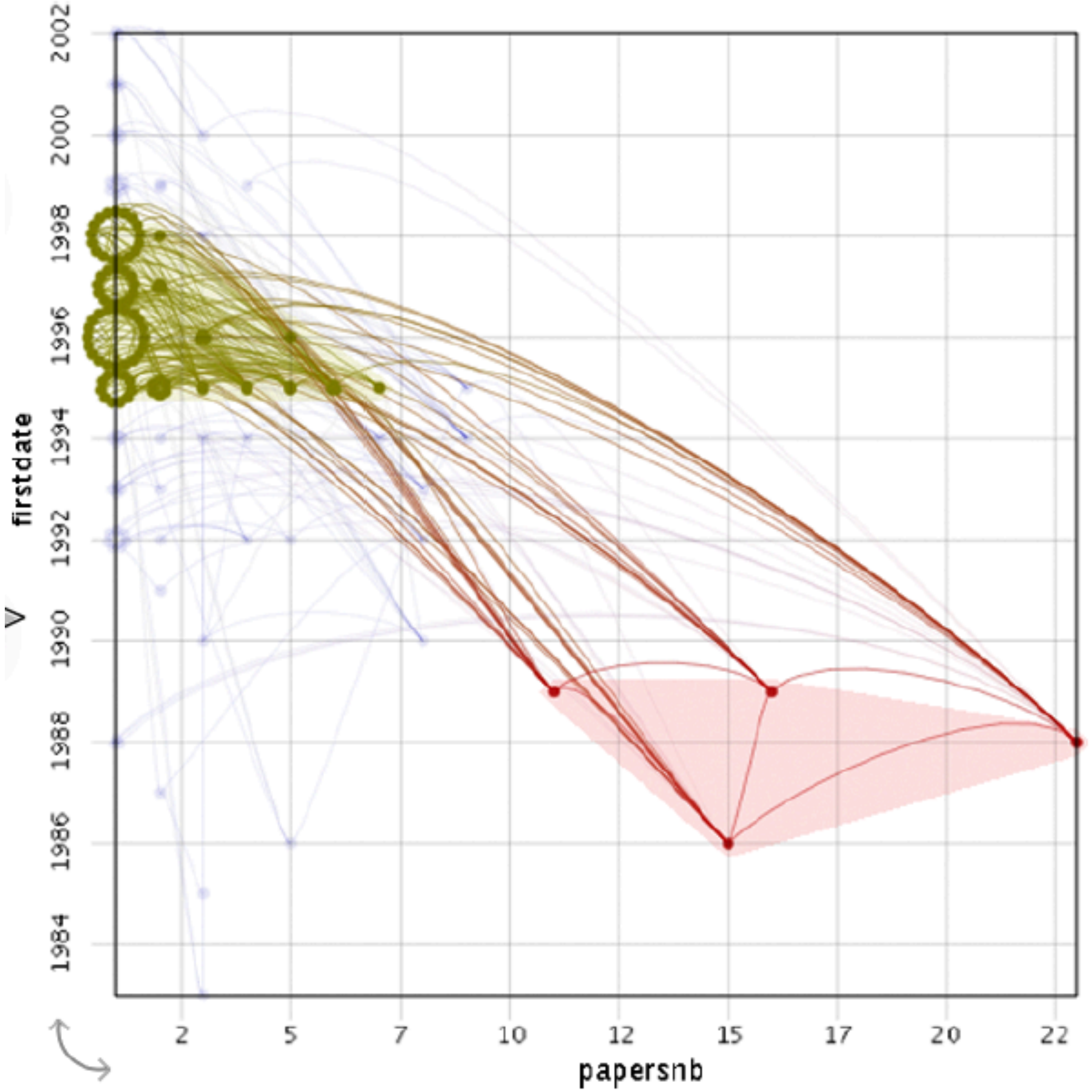
Recommended for networks where nodes can be separated into groups easily and where these groups are central to the analysis

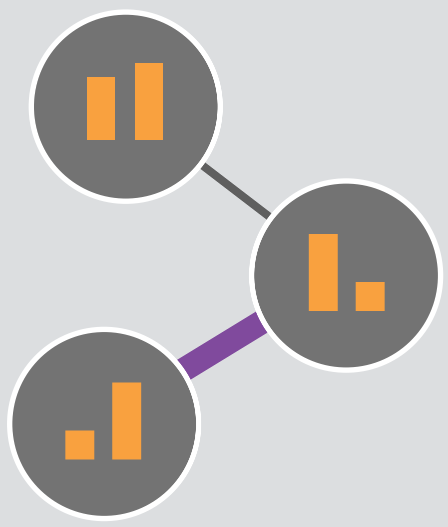
Attribute-Driven Positioning



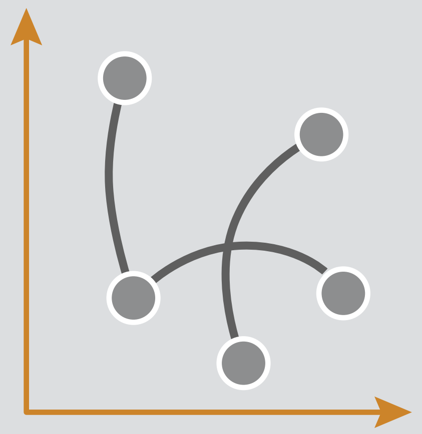

ANCHORAGE
VANCOUVER
EDMONTON
SEATTLE
PORTLAND
SAN FRANCISCO
DENVER
MINNEAPOLIS / ST. PAUL
CHICAGO
KANSAS CITY
DALLAS
TORONTO
CLEVELAND
MONTREAL
BALTIMORE
WASHINGTON D.C.
PHILADELPHIA
NEW YORK
JFK & NEWARK
BOSTON
HALIFAX
TAMPA BAY
ORLANDO
ICELAND
HELSINKI
STOCKHOLM
OSLO
GOTHENBURG
BERGEN
COPENHAGEN
BILLUND
BERLIN
HAMBURG
FRANKFURT
MUNICH
AMSTERDAM
GLASGOW
BRUSSELS
ZURICH
MANCHESTER
MILAN
DUBLIN
LONDON
HEATHROW
& GATWICK
PARIS
ORLY & CDG
GENEVA
MADRID

Graph Dice *Bezerianos et al. 2010*



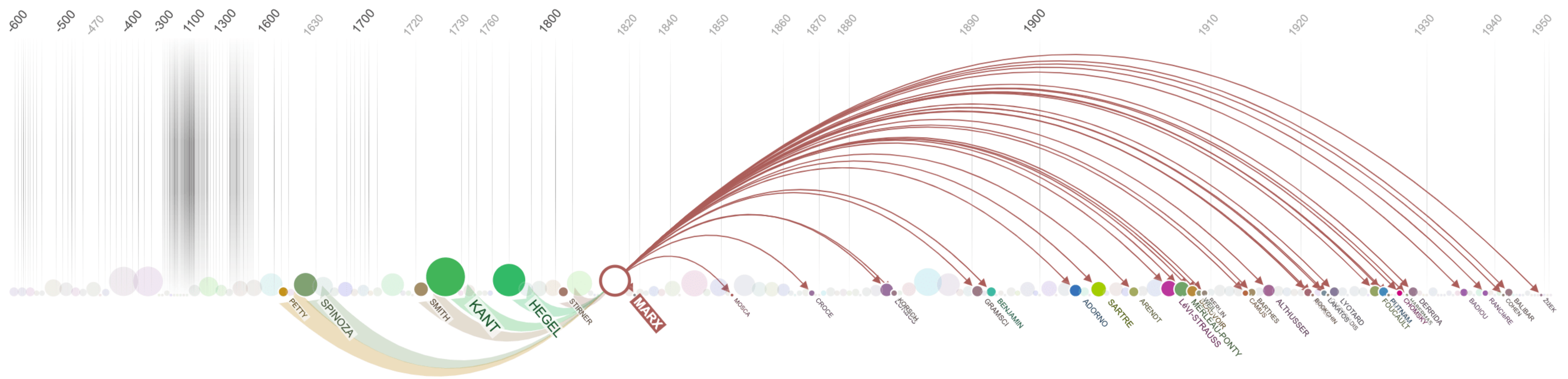


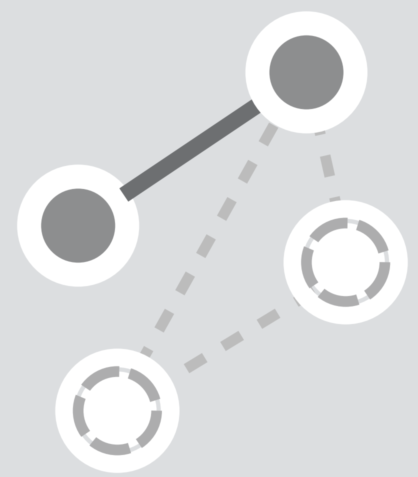
On-Node / On-Edge Encoding



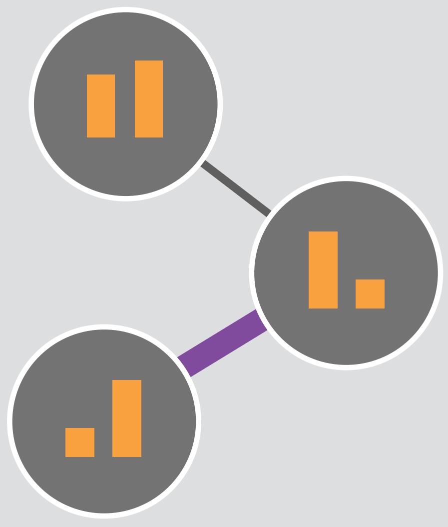
Attribute-Driven Positioning

Edge Map *Dork et al. 2011*

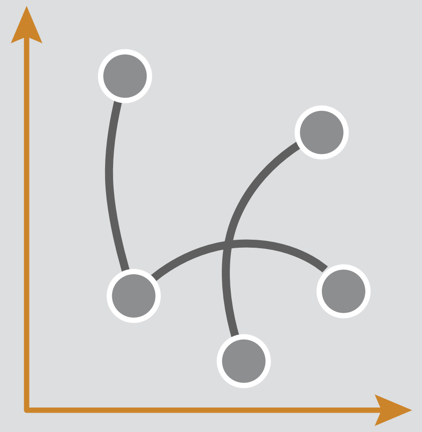




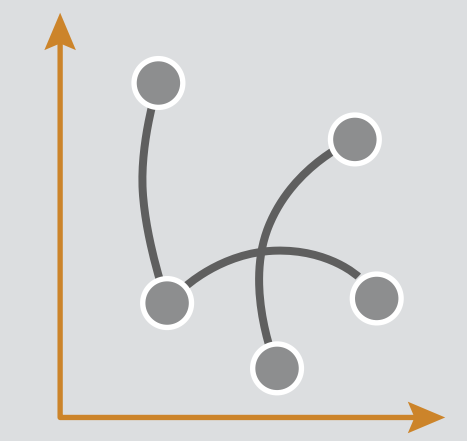
Querying and Filtering



On-Node / On-Edge Encoding



Attribute-Driven Positioning



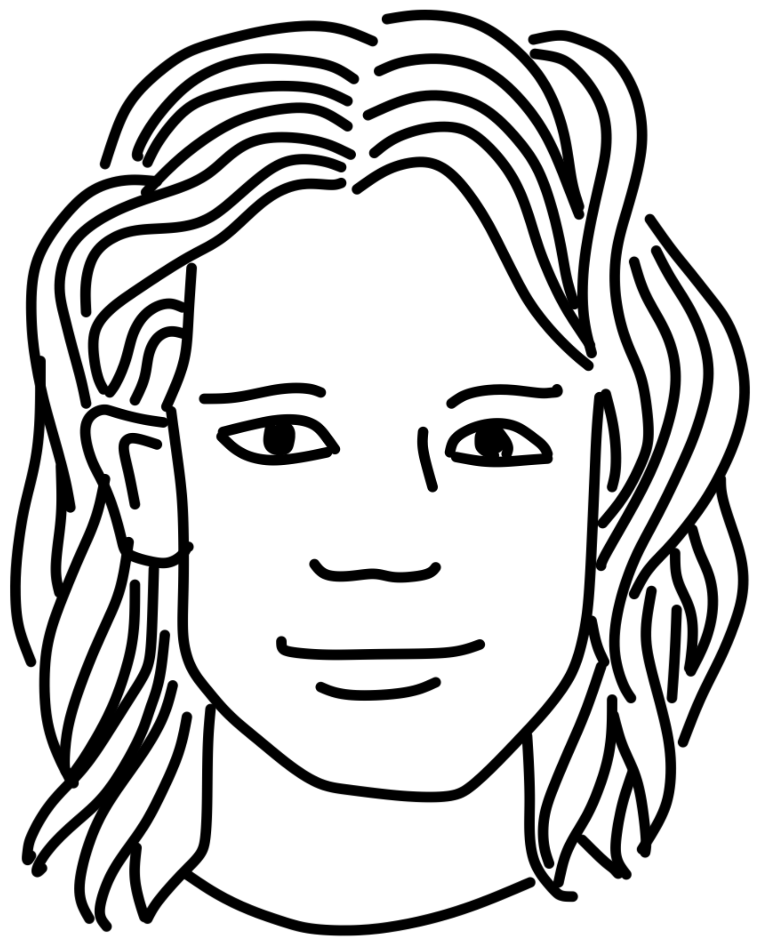
Attribute-Driven Positioning

Well suited for quantitative attributes

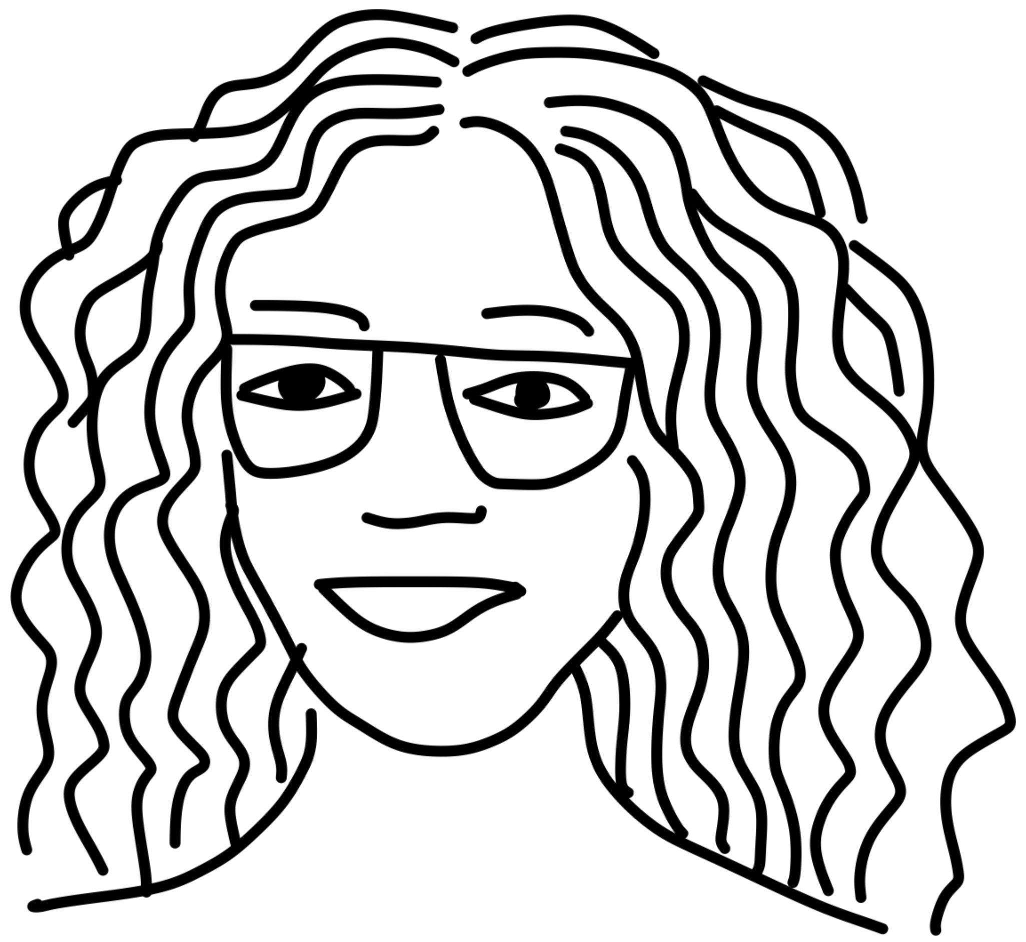
Does not lend itself well to visualizing the topology of the network.

Recommended for smaller, sparse networks where relationships between node attributes are paramount to the analysis task, and topological features only provide context

Tools and Applications

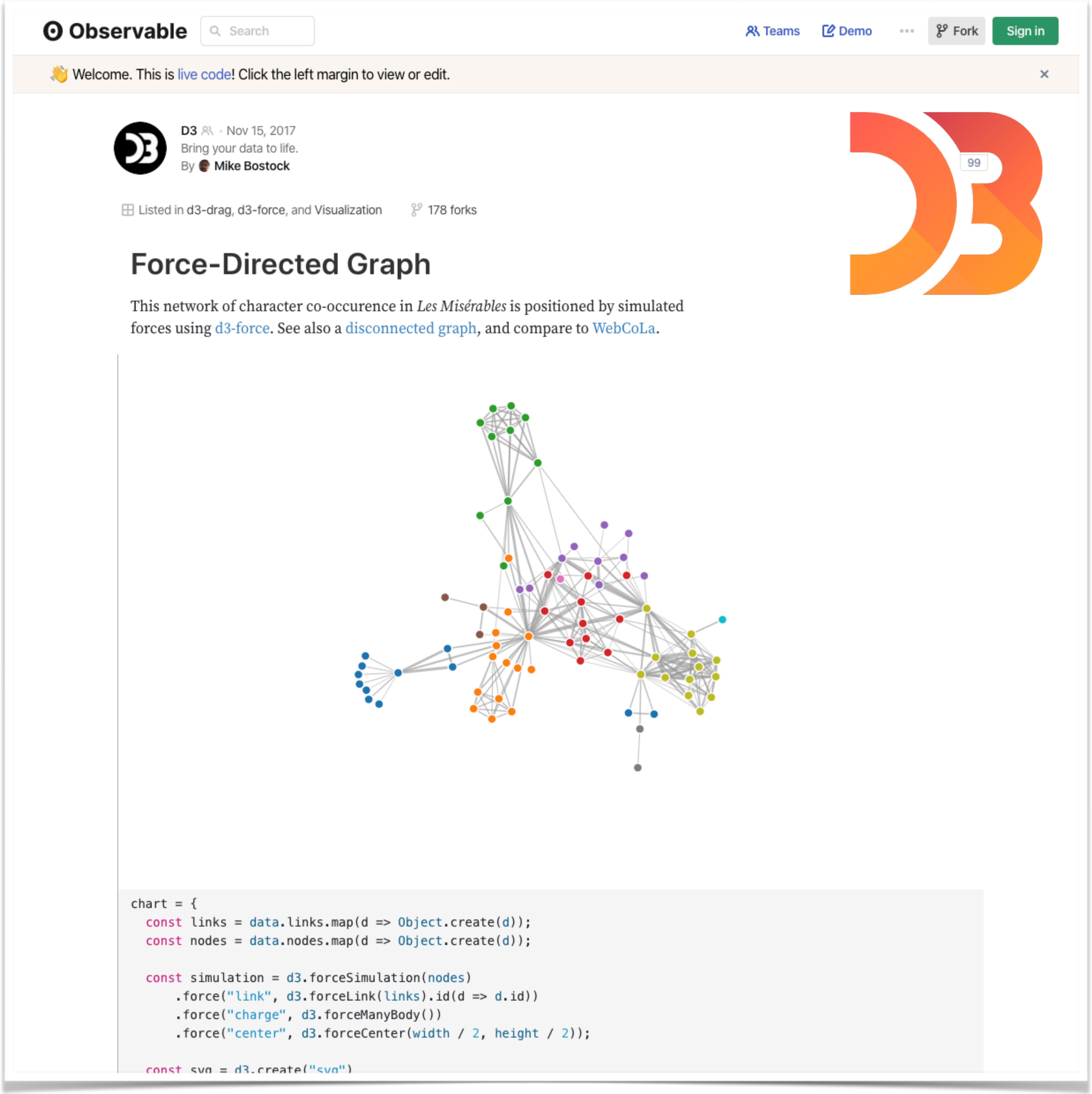


Brad
graphic designer

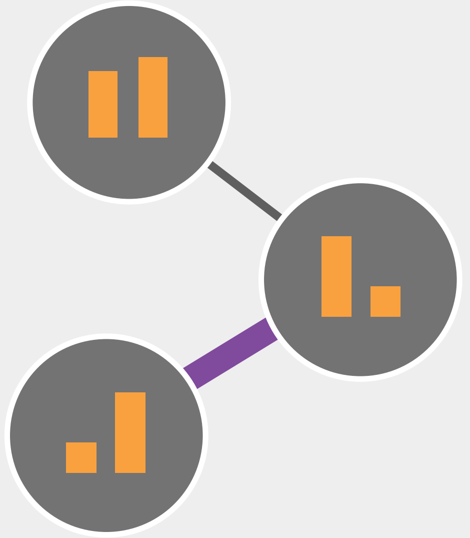


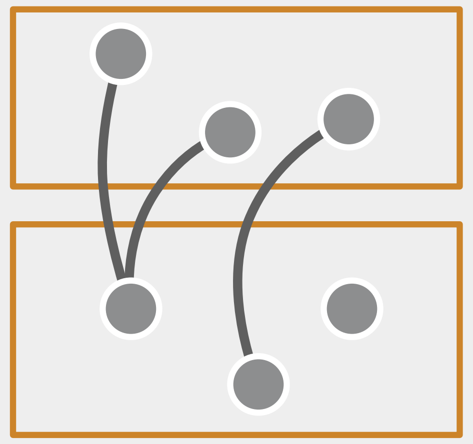
Maya
developer

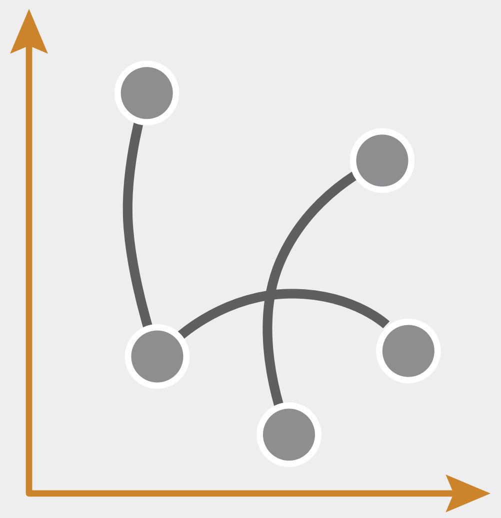
developer



Observable
Search
Teams
Demo
Fork
Sign in
Welcome. This is live code! Click the left margin to view or edit.
D3 · Nov 15, 2017
Bring your data to life.
By Mike Bostock
99
Listed in d3-drag, d3-force, and Visualization
178 forks
Force-Directed Graph
This network of character co-occurence in Les Misérables is positioned by simulated forces using d3-force. See also a disconnected graph, and compare to WebCoLa.
chart = {
const links = data.links.map(d => Object.create(d));
const nodes = data.nodes.map(d => Object.create(d));
const simulation = d3.forceSimulation(nodes)
.force("link", d3.forceLink(links).id(d => d.id))
.force("charge", d3.forceManyBody())
.force("center", d3.forceCenter(width / 2, height / 2));







developer

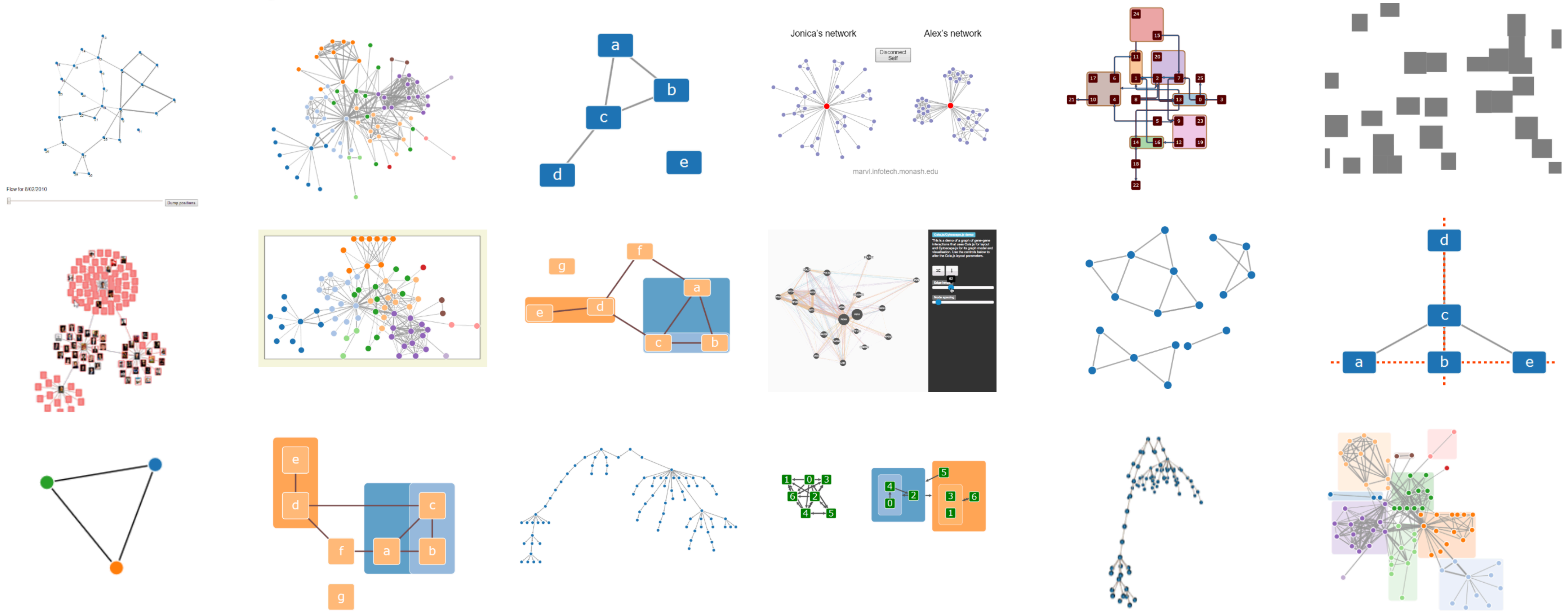
JS

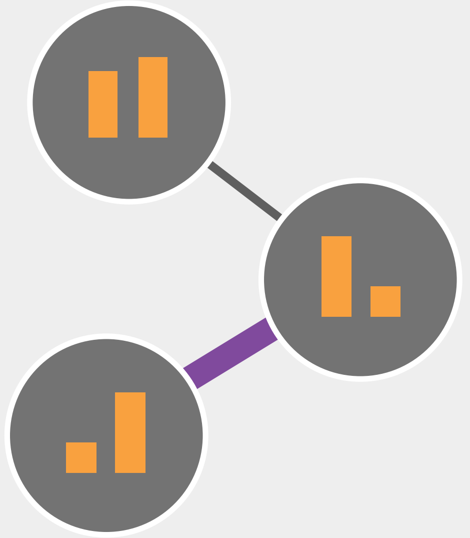
Cola.js (A.K.A. "WebCoLa") is an open-source JavaScript library for arranging your HTML5 documents and diagrams using constraint-based optimization techniques.

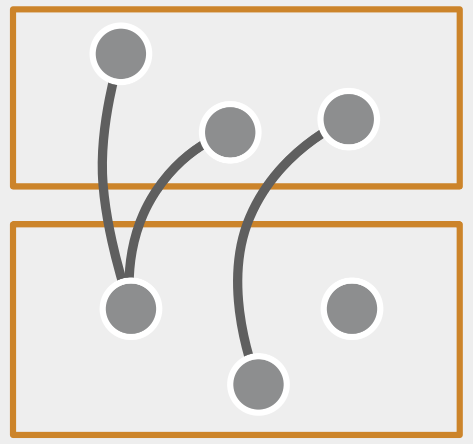
Overview Wiki API Source

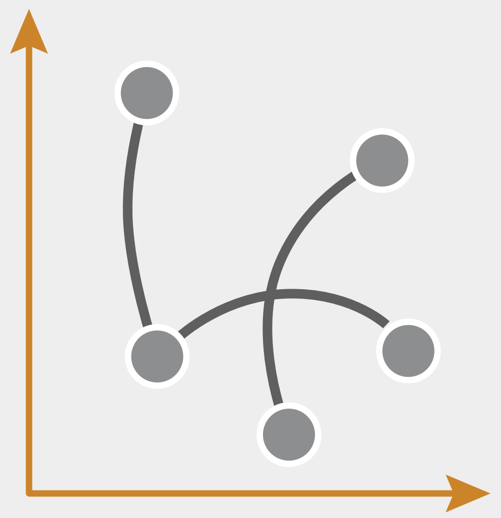
cola.js

Constraint-Based Layout in the Browser









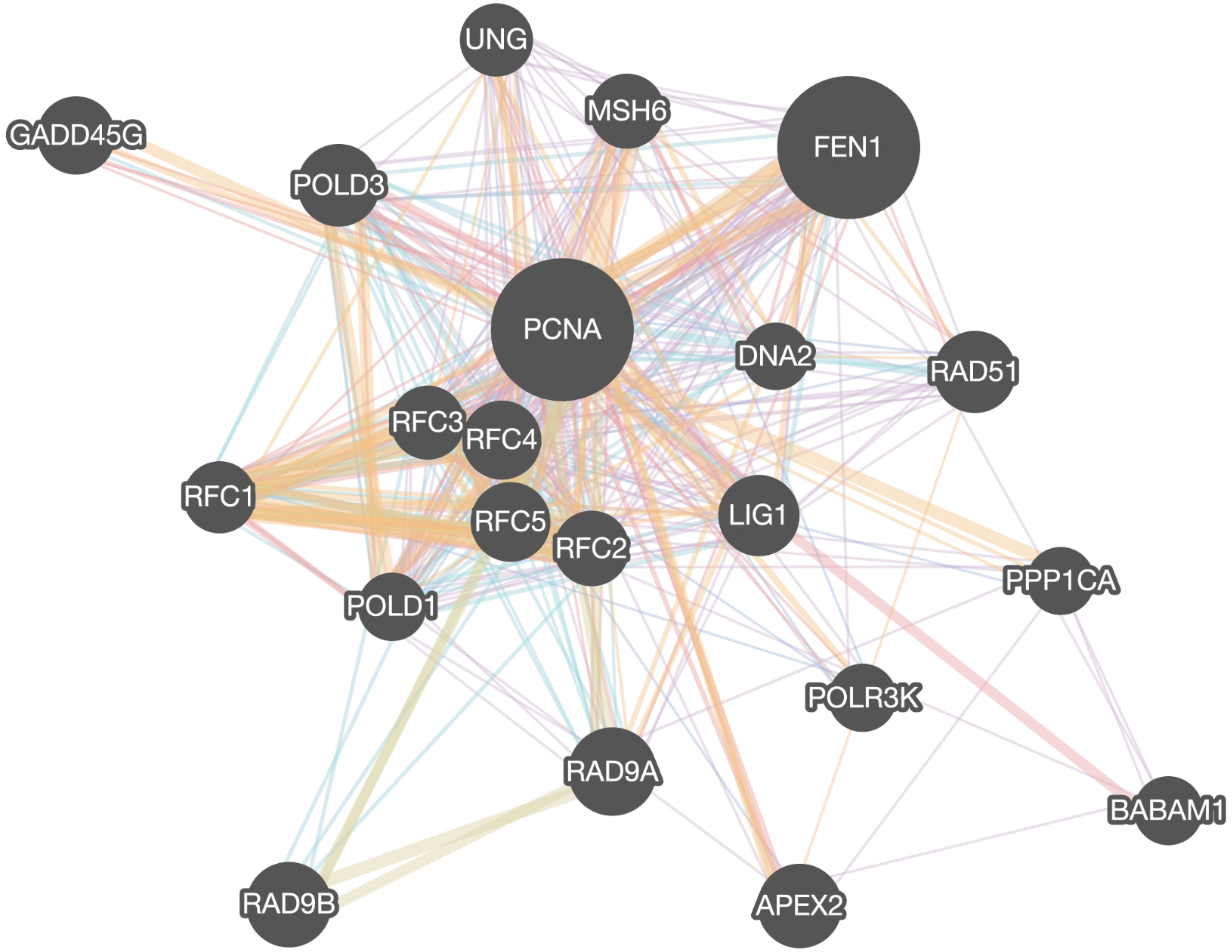
developer

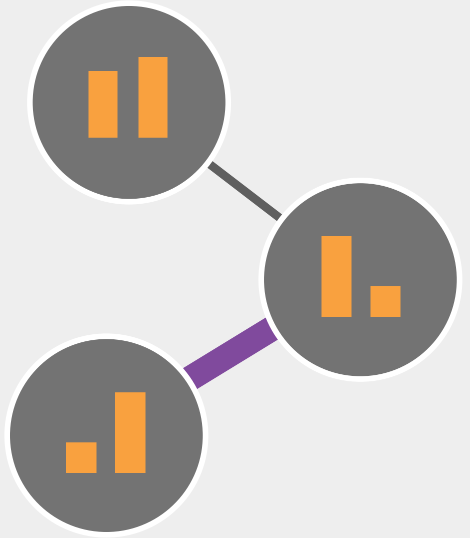
Cytoscape.js

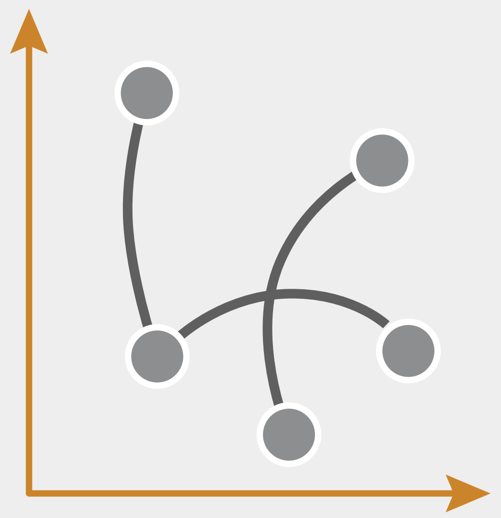
Graph theory (network) library for visualisation and analysis

Repo GitHub | Updates Twitter | News and tutorials Blog | Questions StackOverflow | Ask a question StackOverflow

npm installs 100k/month | master branch passing | unstable branch passing | Greenkeeper enabled







developer

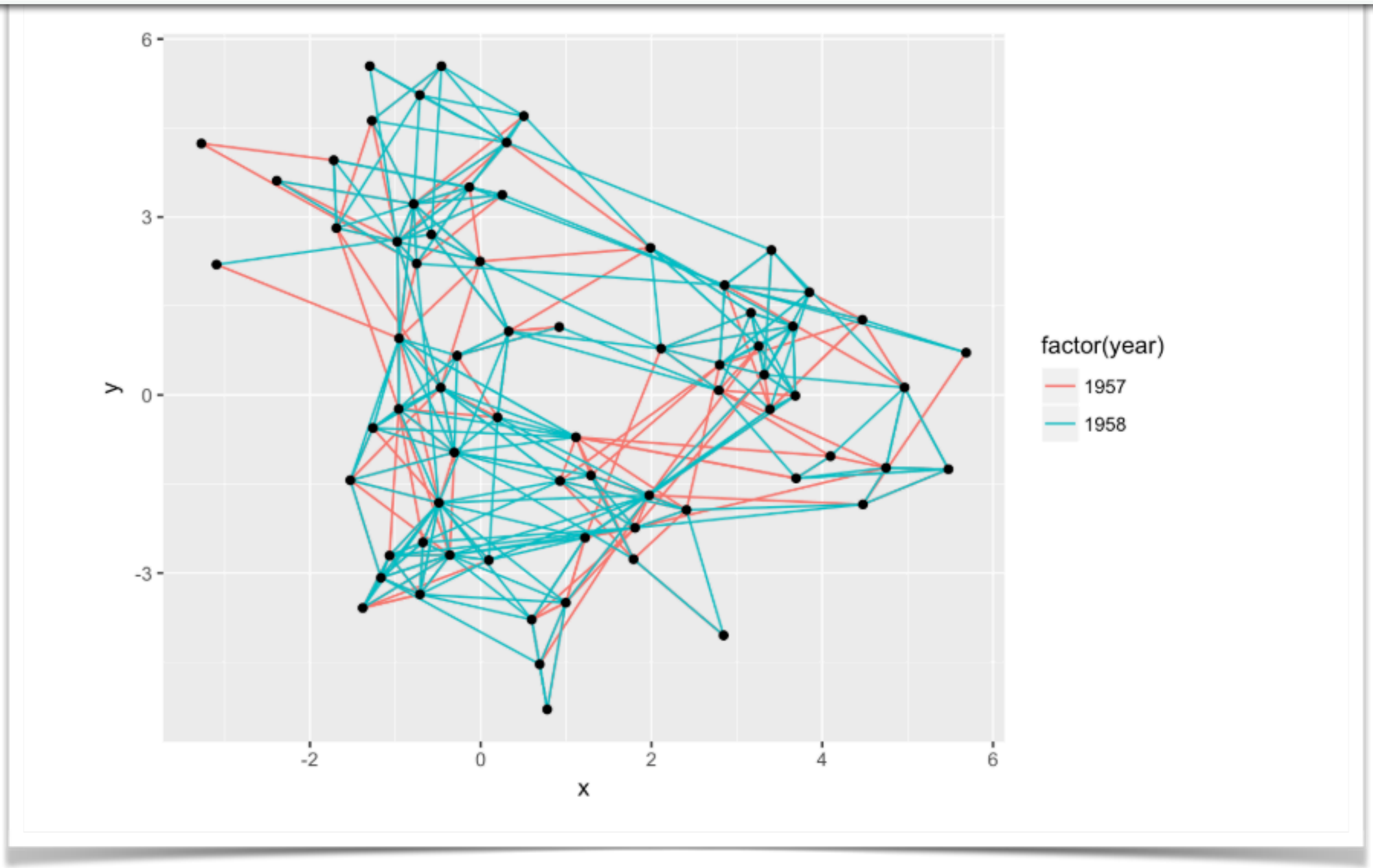
GGRAPH 1.0.2.9999 | Reference | Getting Started | Articles | News

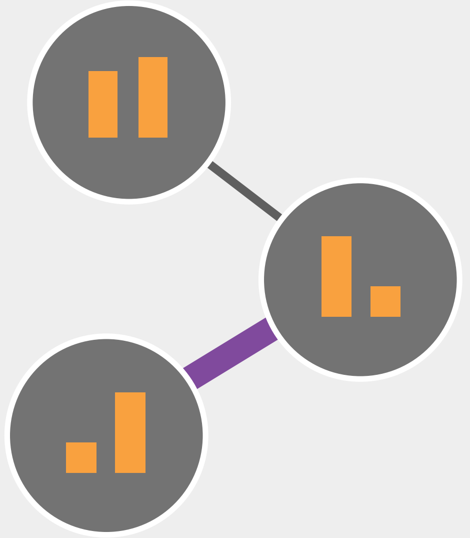
ggraph

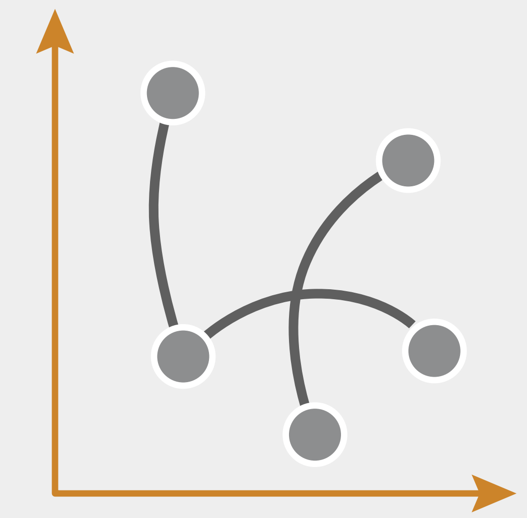
/dʒiː.dʒɪˈraːf/ (or g-giraffe)

A grammar of graphics for relational data

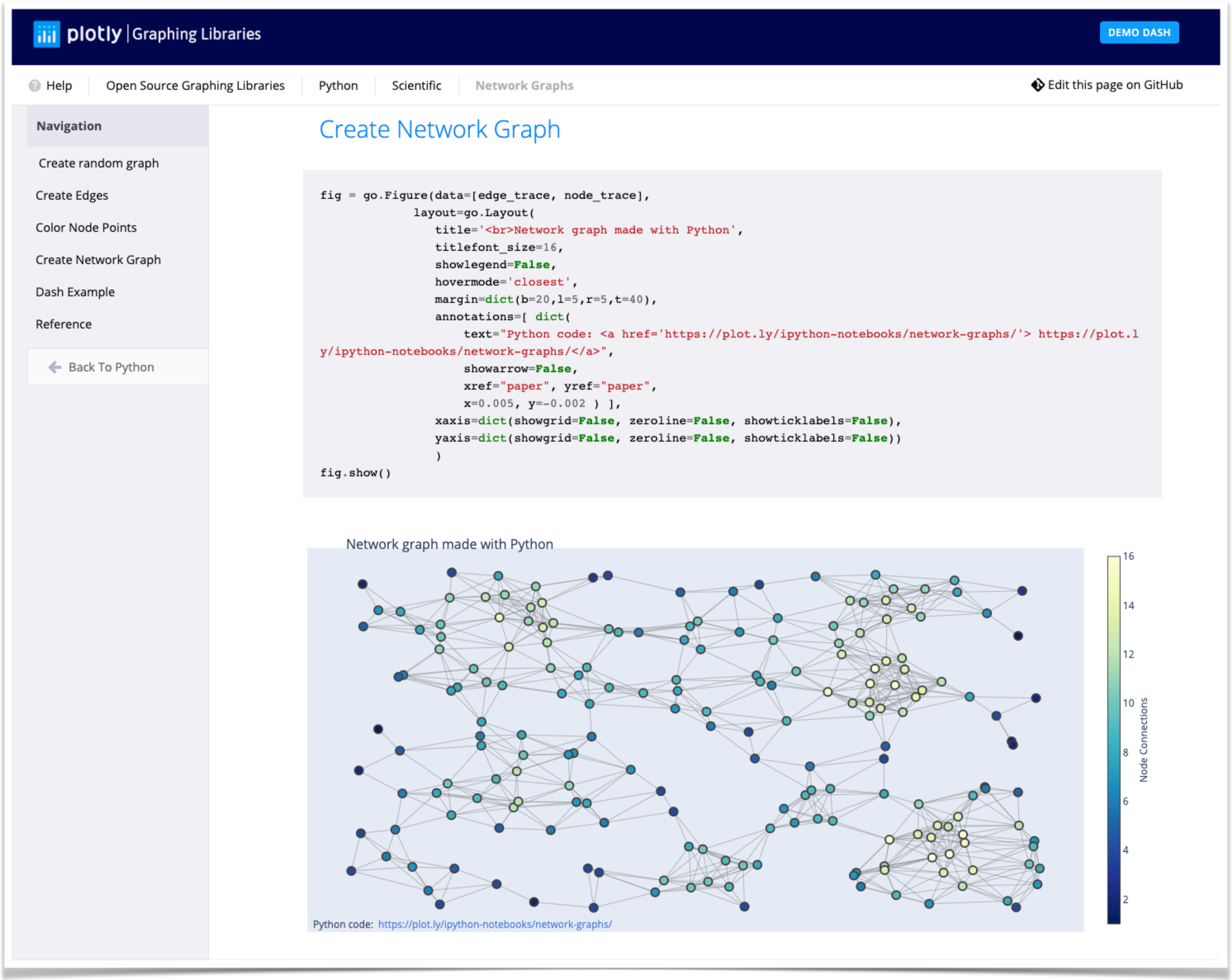
ggraph is an extension of `ggplot2` aimed at supporting relational data structures such as networks, graphs, and trees. While it builds upon the foundation of `ggplot2` and its API it comes with its own self-contained set of geoms, facets, etc., as well as adding the concept of *layouts* to the grammar.



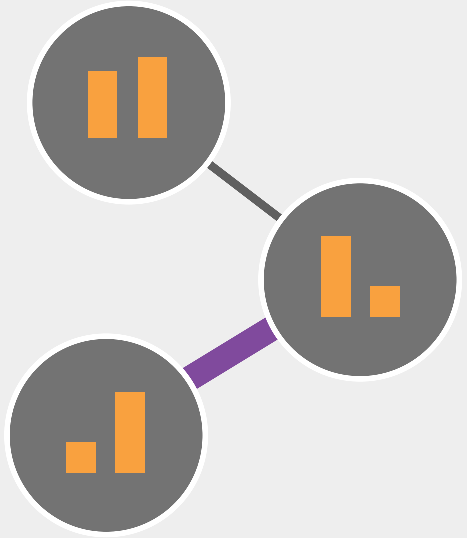


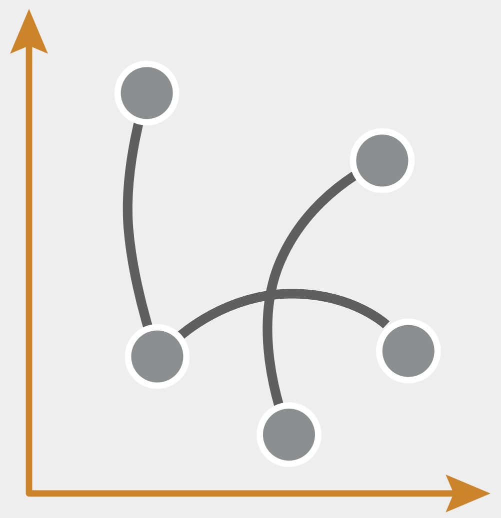


developer



plotly | Graphing Libraries
DEMO DASH
Help
Open Source Graphing Libraries
Python
Scientific
Network Graphs
Edit this page on GitHub
Navigation
Create random graph
Create Edges
Color Node Points
Create Network Graph
Dash Example
Reference
Back To Python
Create Network Graph
fig = go.Figure(data=[edge_trace, node_trace],
layout=go.Layout(
title='
Network graph made with Python',
titlefont_size=16,
showlegend=False,
hovermode='closest',
margin=dict(b=20,l=5,r=5,t=40),
annotations=[dict(
text="Python code: https://plot.ly/ipython-notebooks/network-graphs/",
showarrow=False,
xref="paper", yref="paper",
x=0.005, y=-0.002)],
xaxis=dict(showgrid=False, zeroline=False, showticklabels=False),
yaxis=dict(showgrid=False, zeroline=False, showticklabels=False))
)
fig.show()
Network graph made with Python
16
14
12
10
8
6
4
2
Node Connections
Python code: https://plot.ly/ipython-notebooks/network-graphs/





developer

NetworkX

Stable (notes)

2.3 — April 2019
download | doc | pdf

Latest (notes)

2.4 development
github | doc | pdf

Archive

Contact

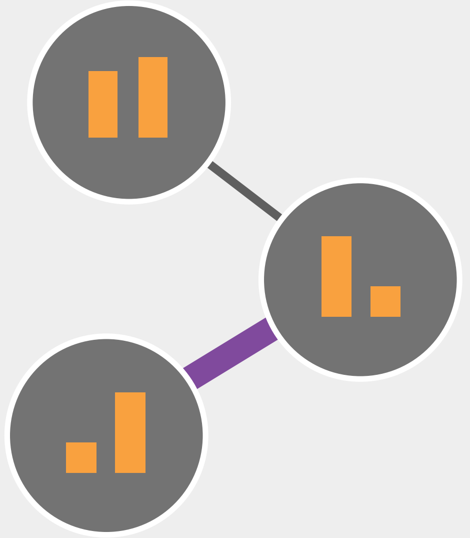
Mailing list
Issue tracker

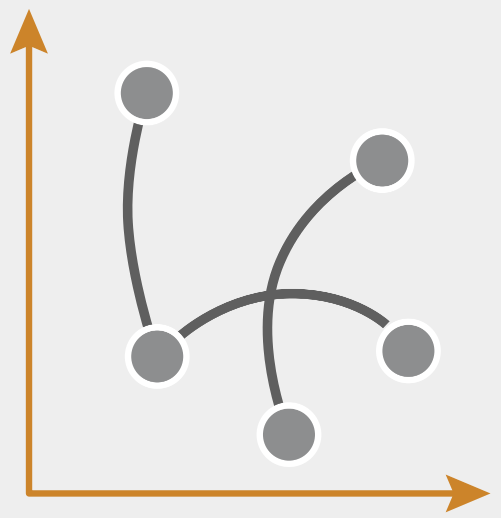
Software for complex networks

NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.

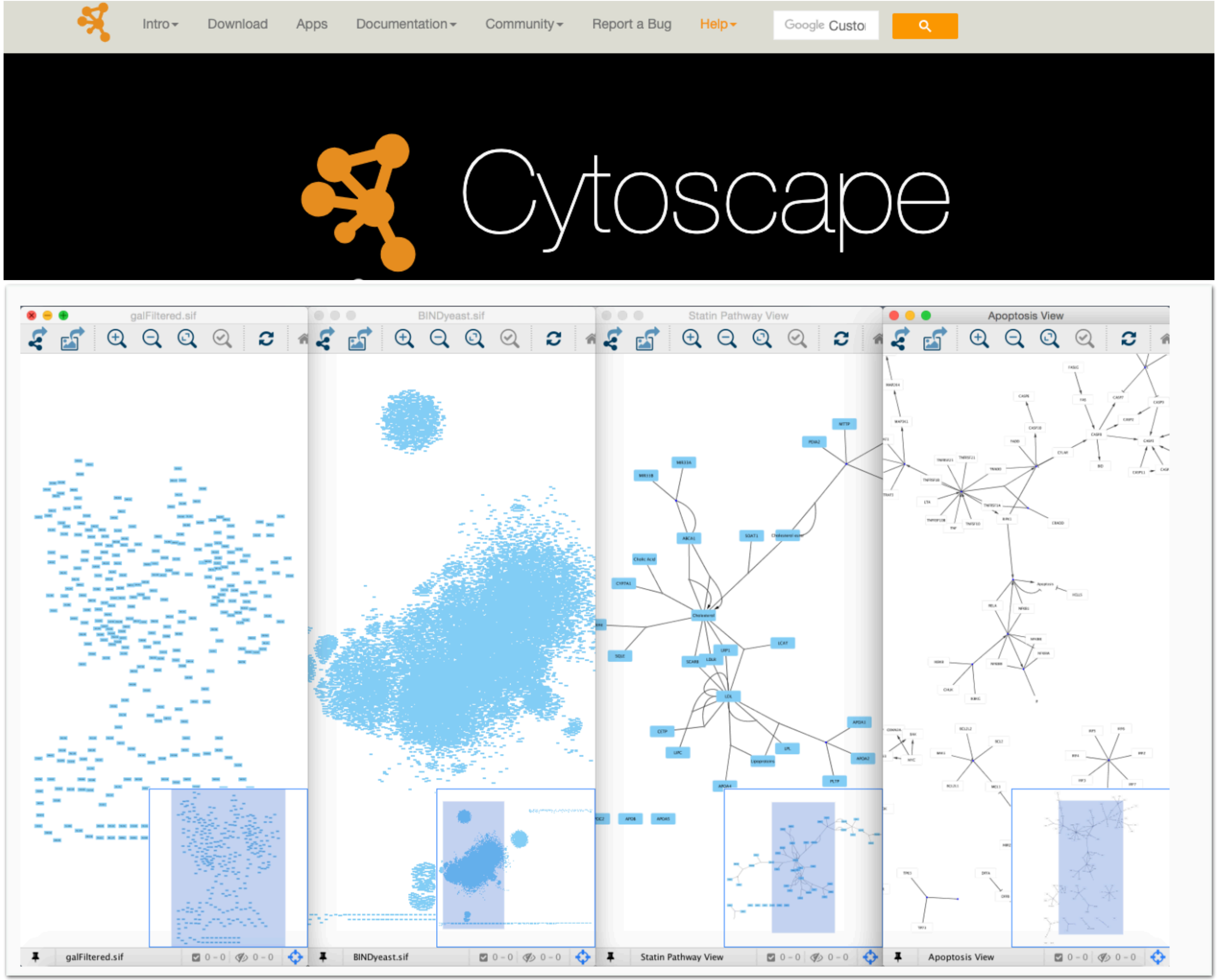
Features

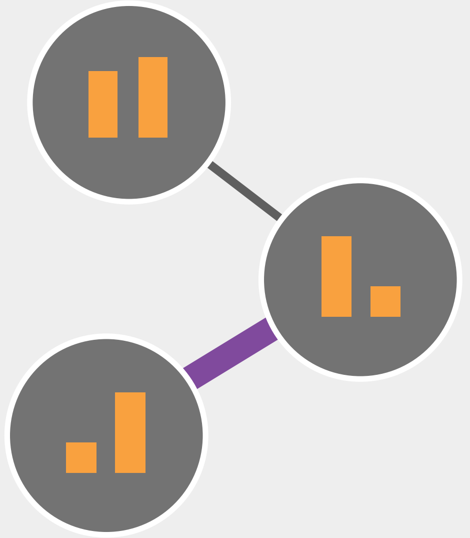
- Data structures for graphs, digraphs, and multigraphs
- Many standard graph algorithms
- Network structure and analysis measures
- Generators for classic graphs, random graphs, and synthetic networks
- Nodes can be "anything" (e.g., text, images, XML records)
- Edges can hold arbitrary data (e.g., weights, time-series)
- Open source 3-clause BSD license
- Well tested with over 90% code coverage
- Additional benefits from Python include fast prototyping, easy to teach, and multi-platform

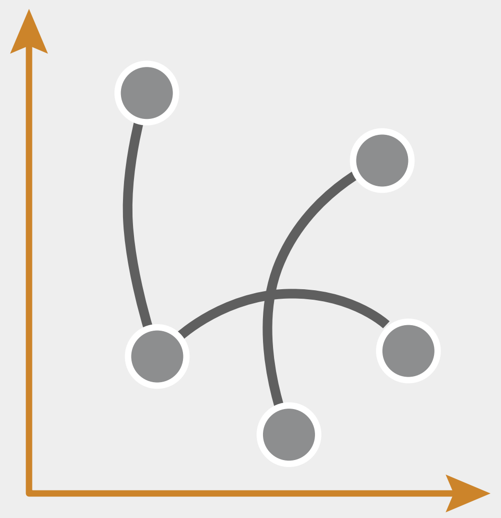




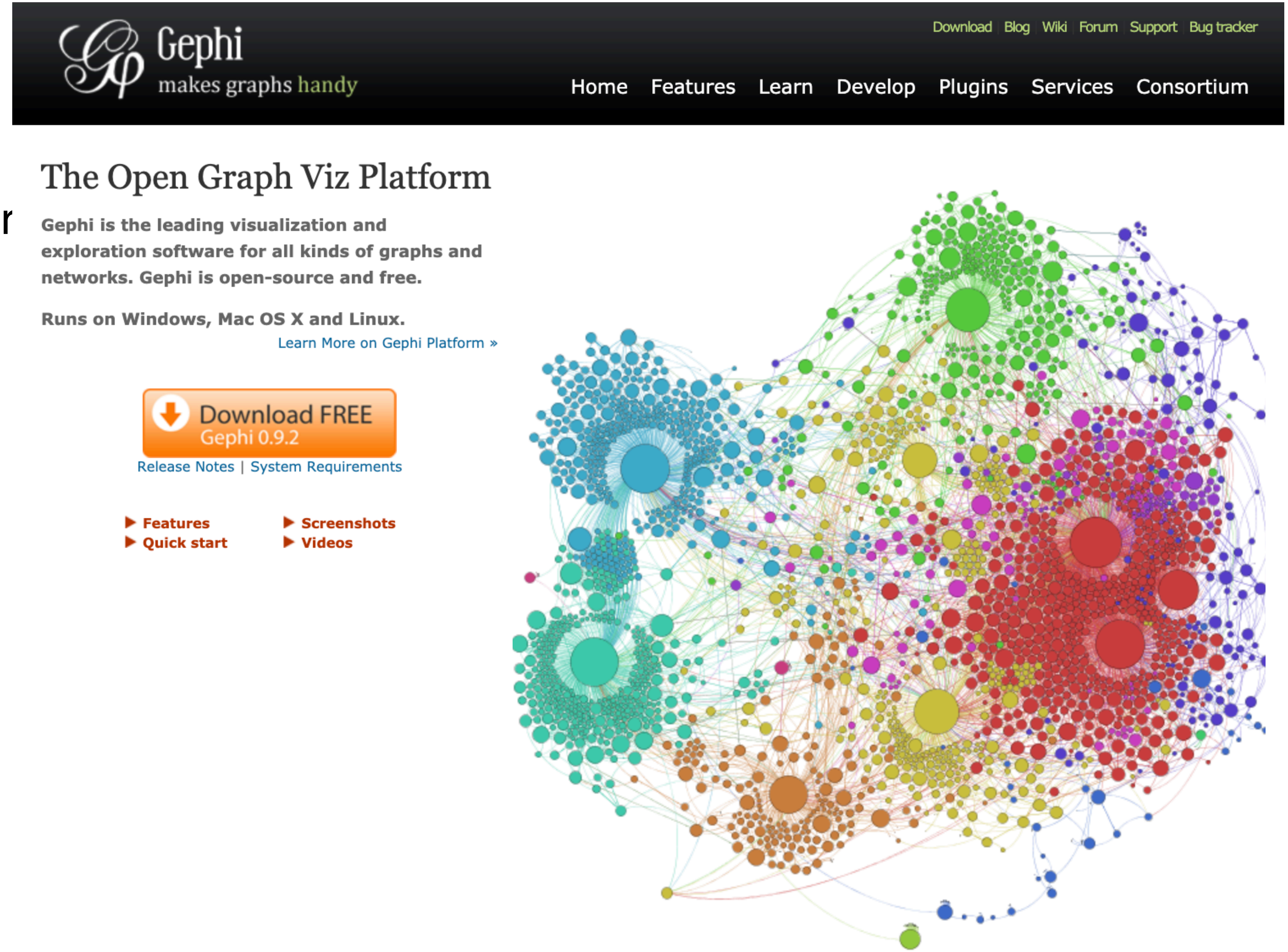
graphic designer

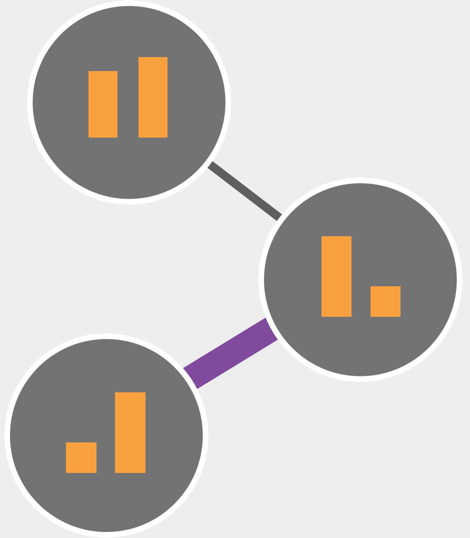


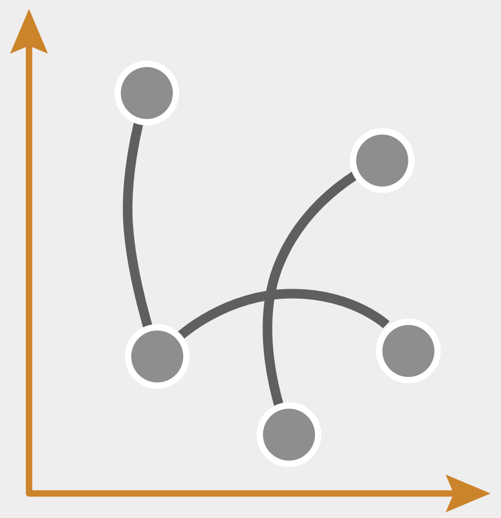




graphic designer

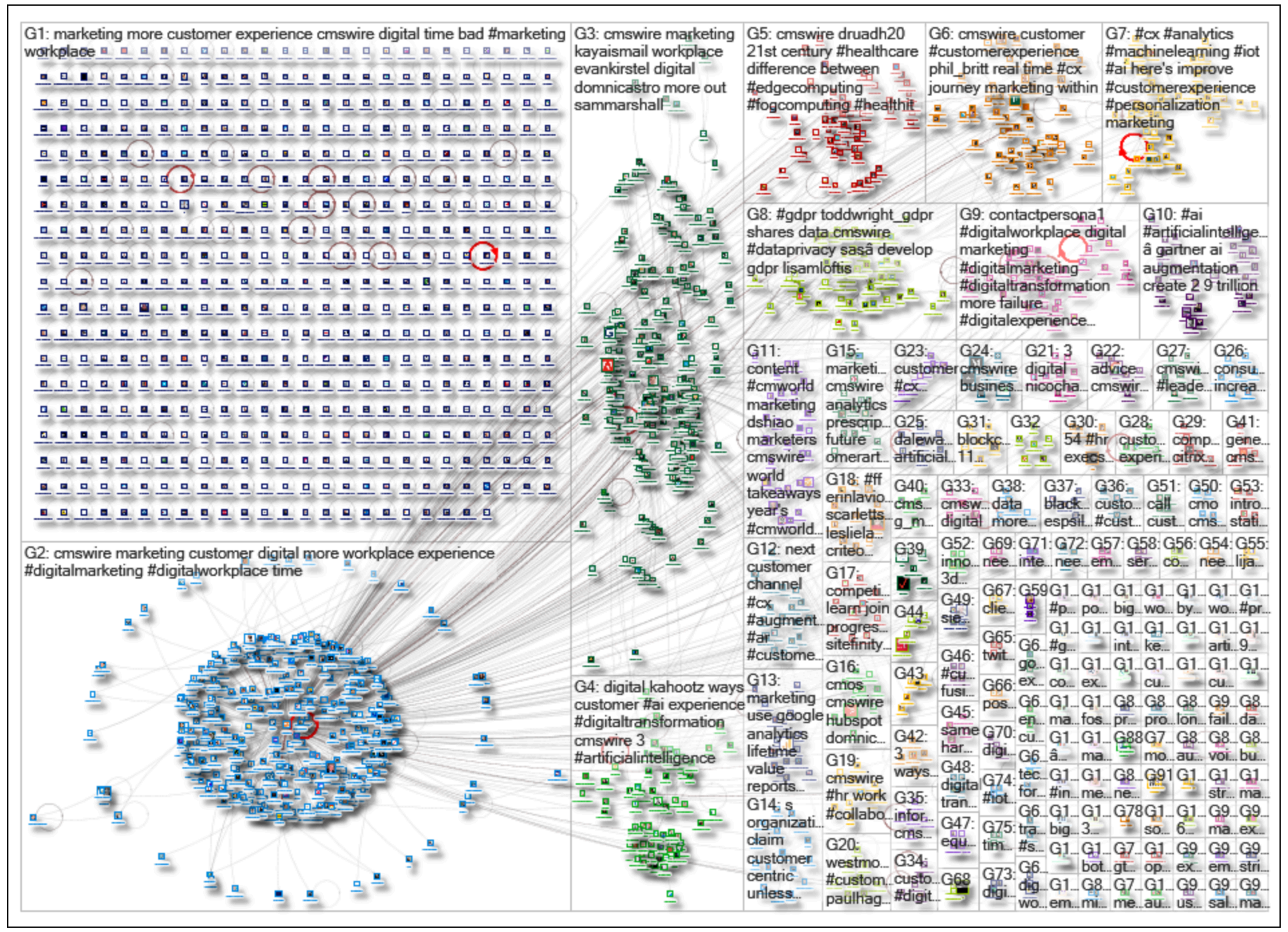


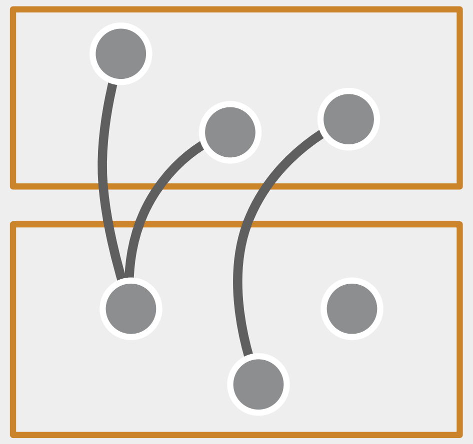




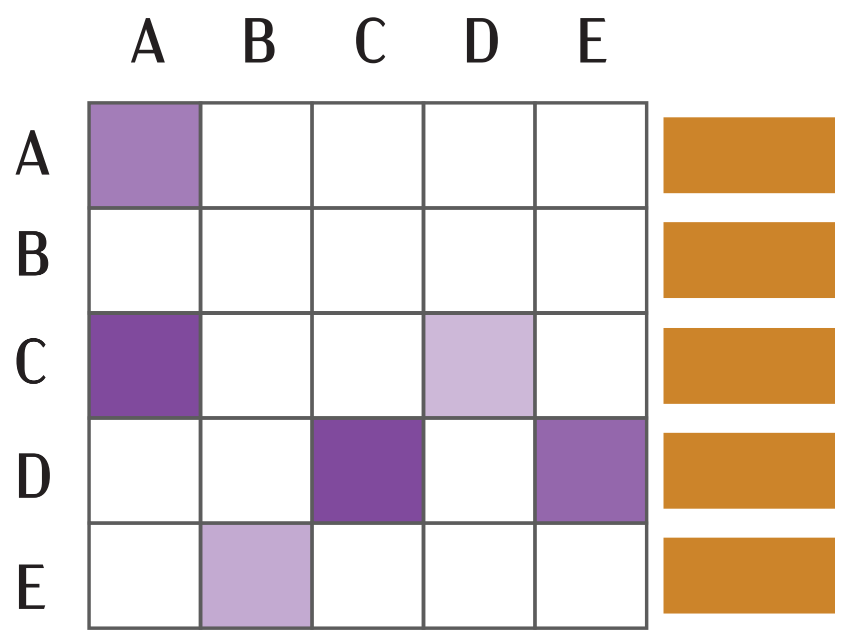
graphic designer

NODE

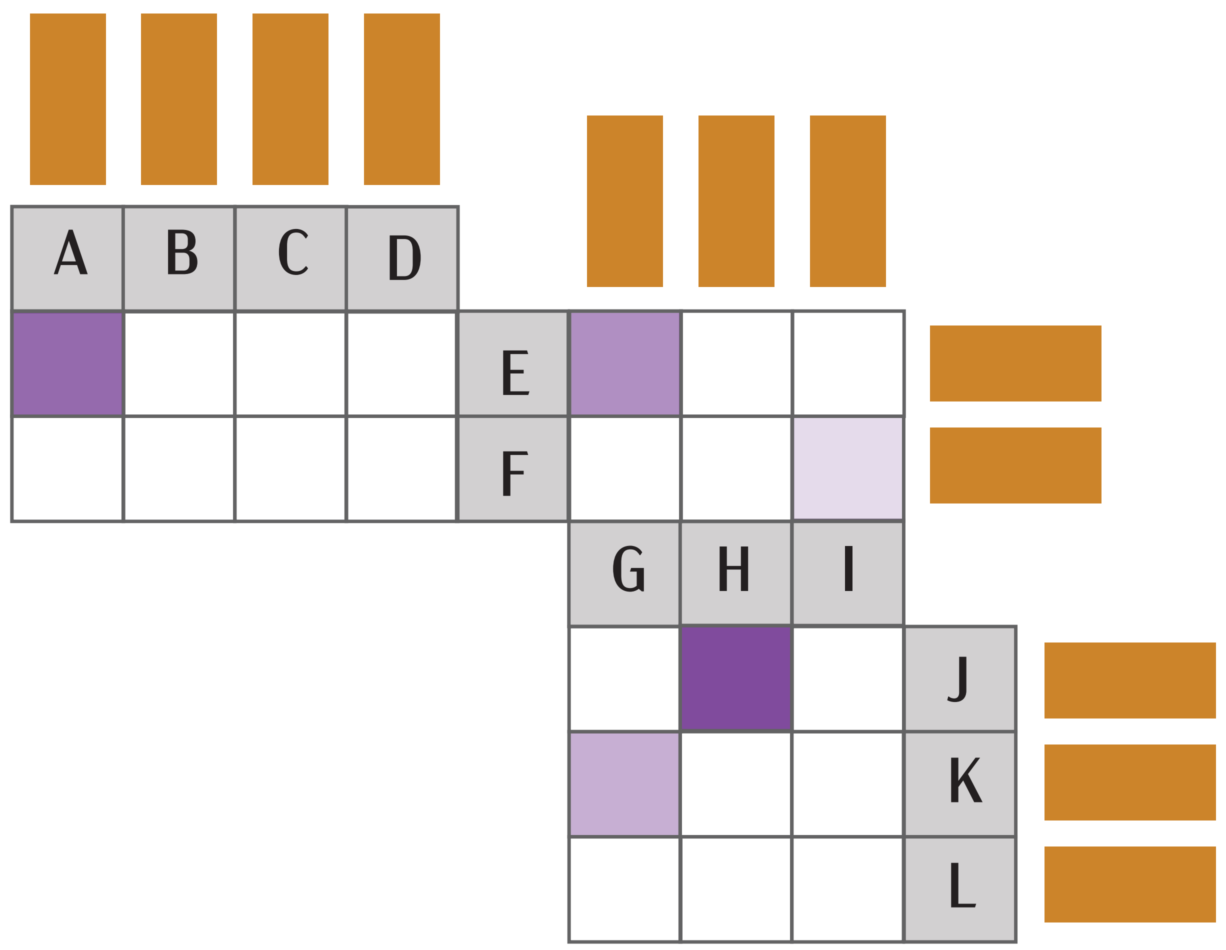




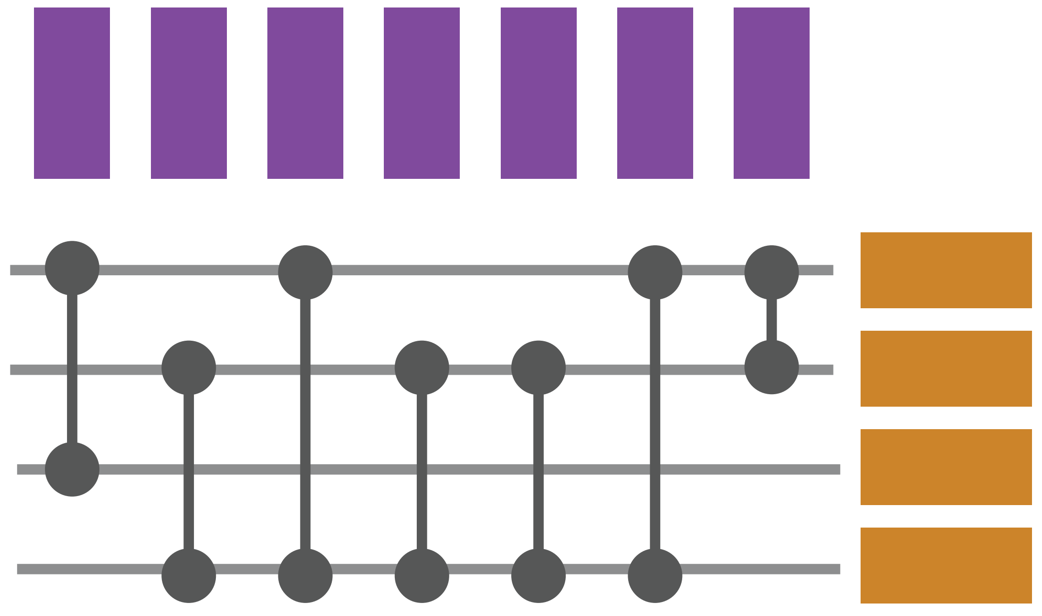
Tabular Layouts



Adjacency Matrix

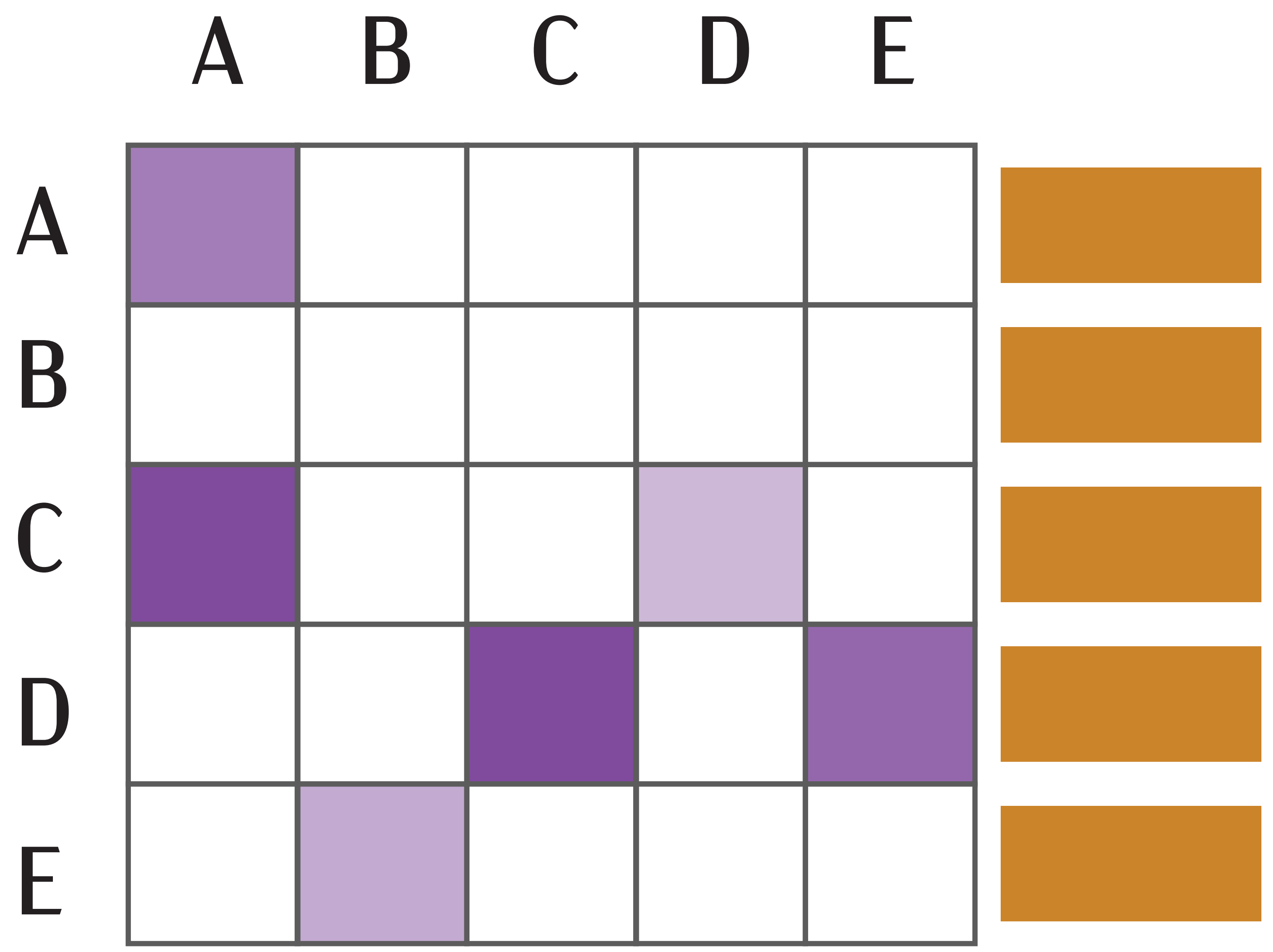


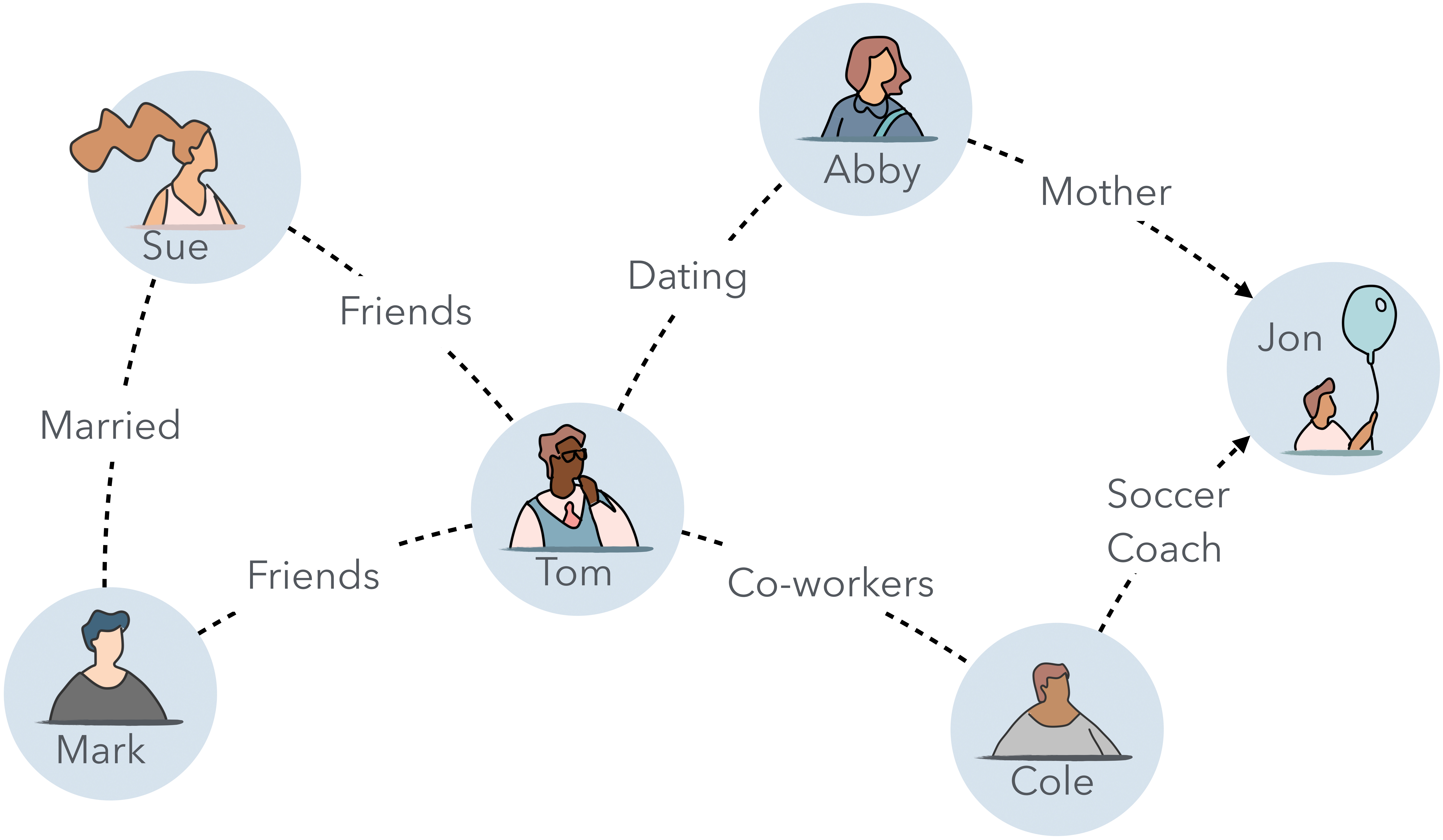
Quilts

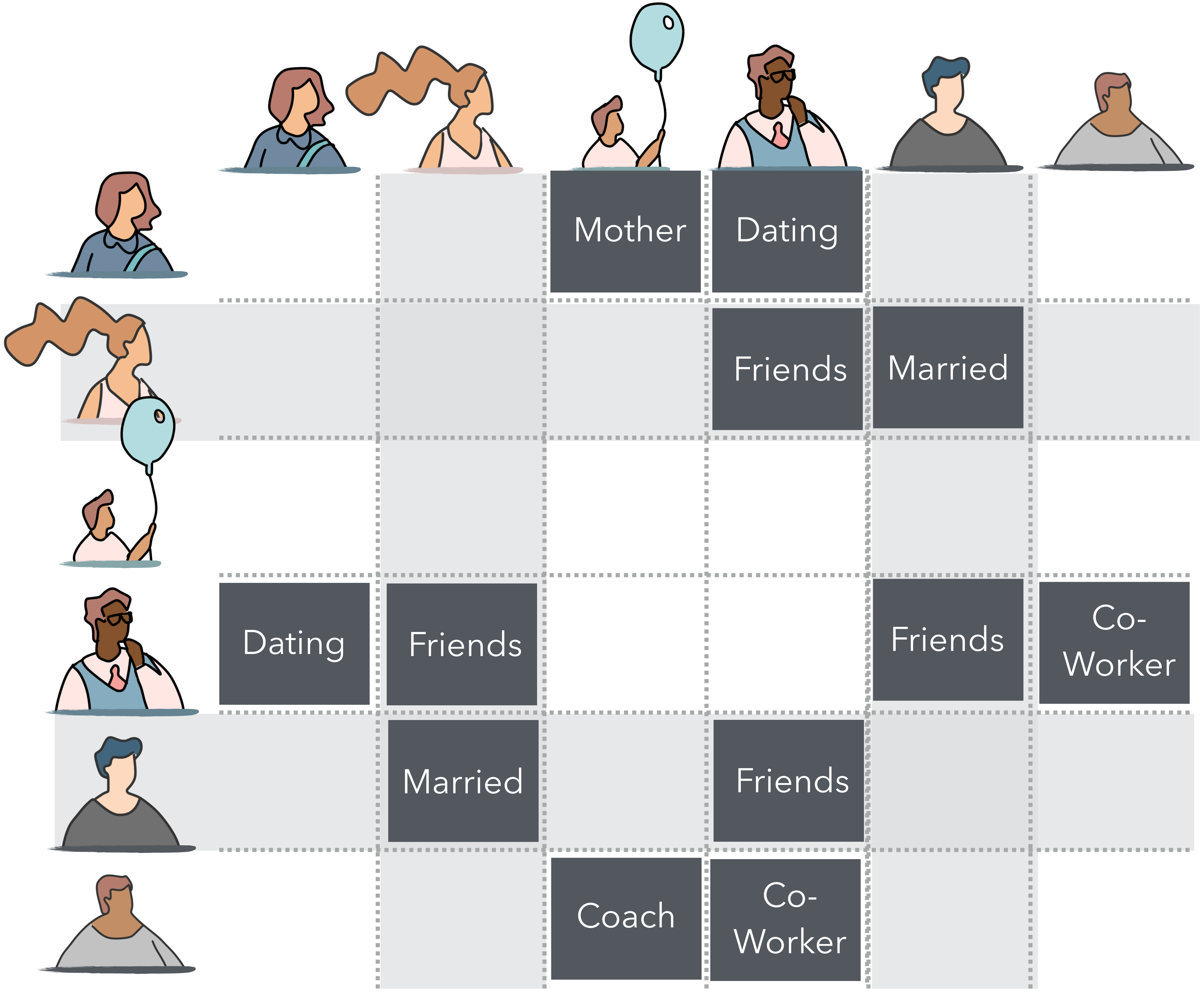


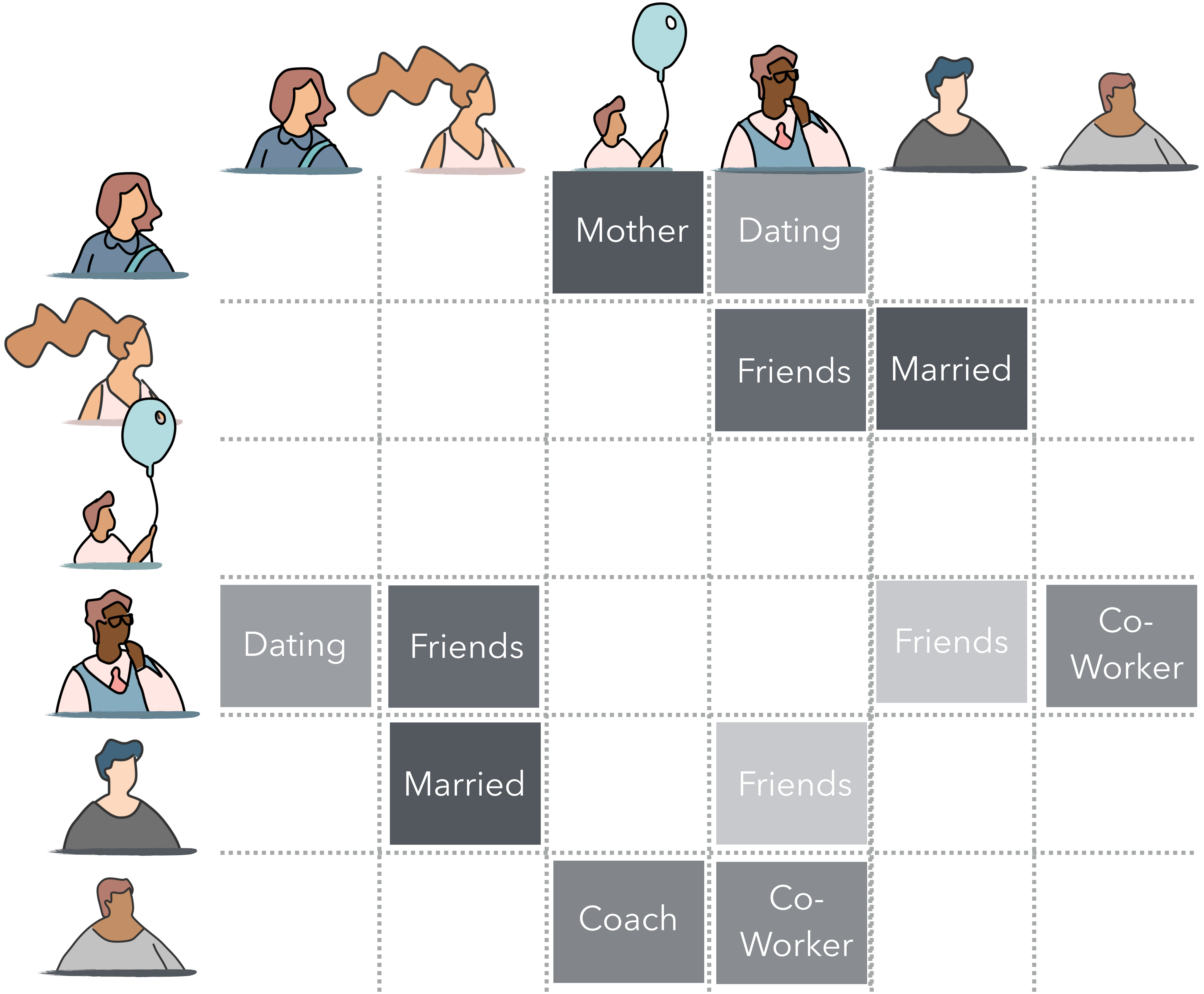
BioFabric

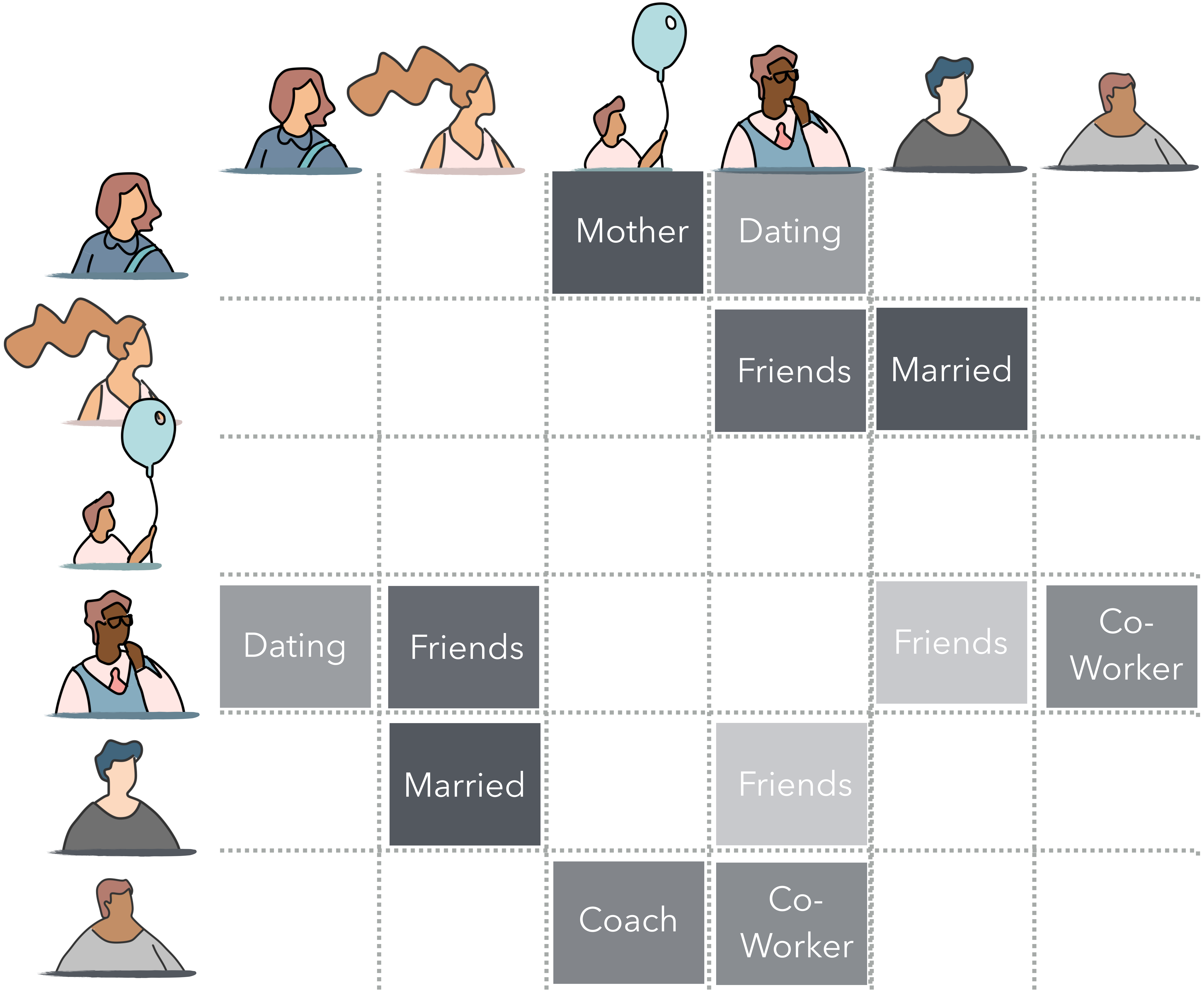
Adjacency Matrix



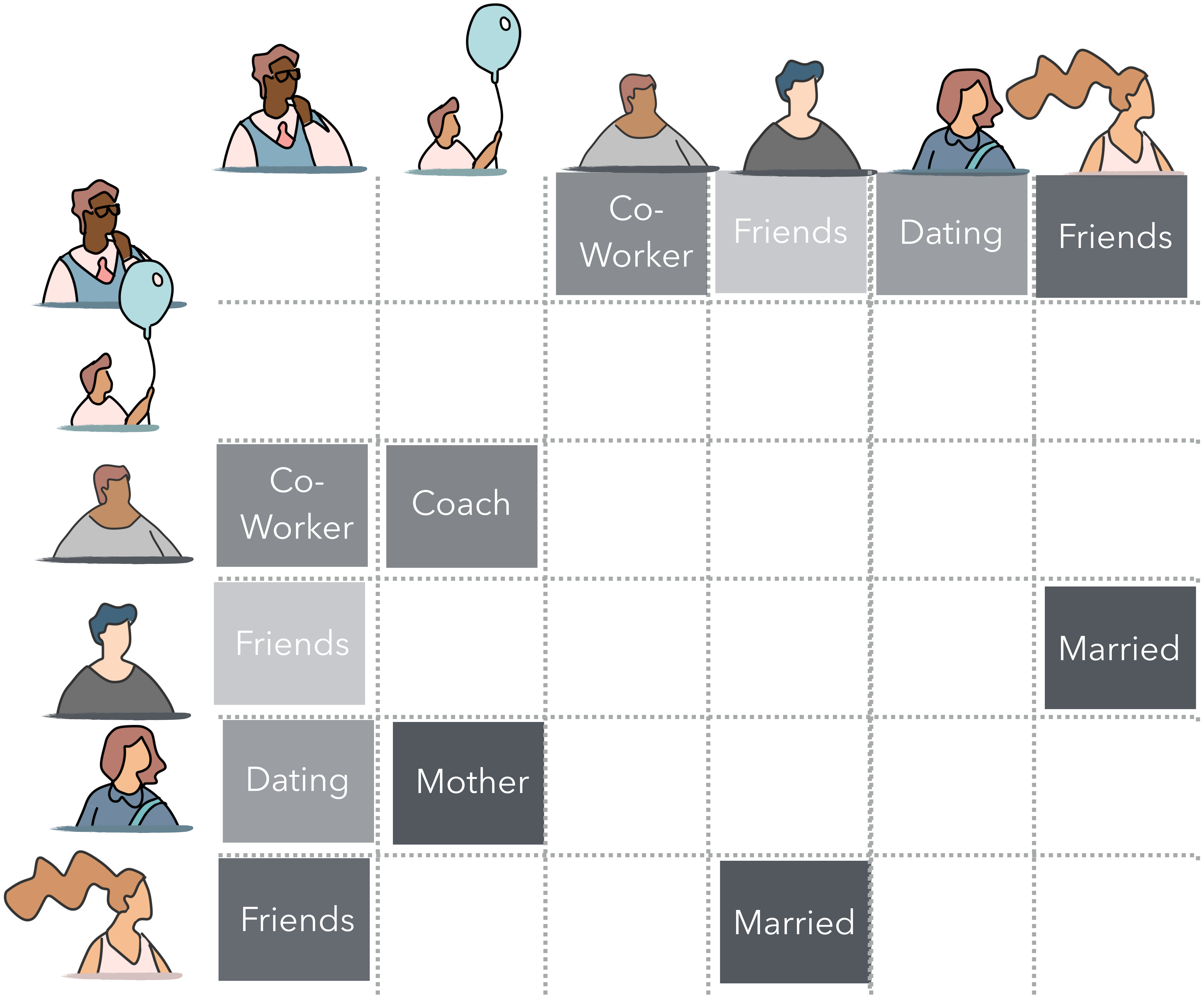

Sue
Abby
Mother
Friends
Dating
Jon
Married
Soccer
Coach
Tom
Friends
Co-workers
Mark
Cole


Mother
Dating
Friends
Married
Dating
Friends
Friends
Co-
Worker
Married
Friends
Coach
Co-
Worker


Mother
Dating
Friends
Married
Dating
Friends
Friends
Co-Worker
Married
Friends
Coach
Co-Worker

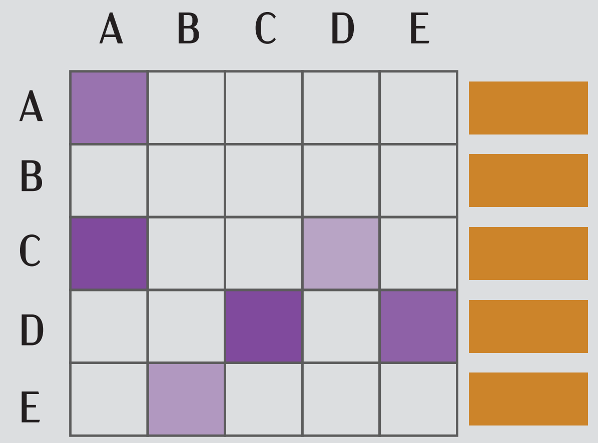


Name	Beverage	Day 1
Abby	Port	1
Sue	Coke	0
Jon	Coke	4
Tom	Beer	5
Mark	Beer	2
Cole	Port	3

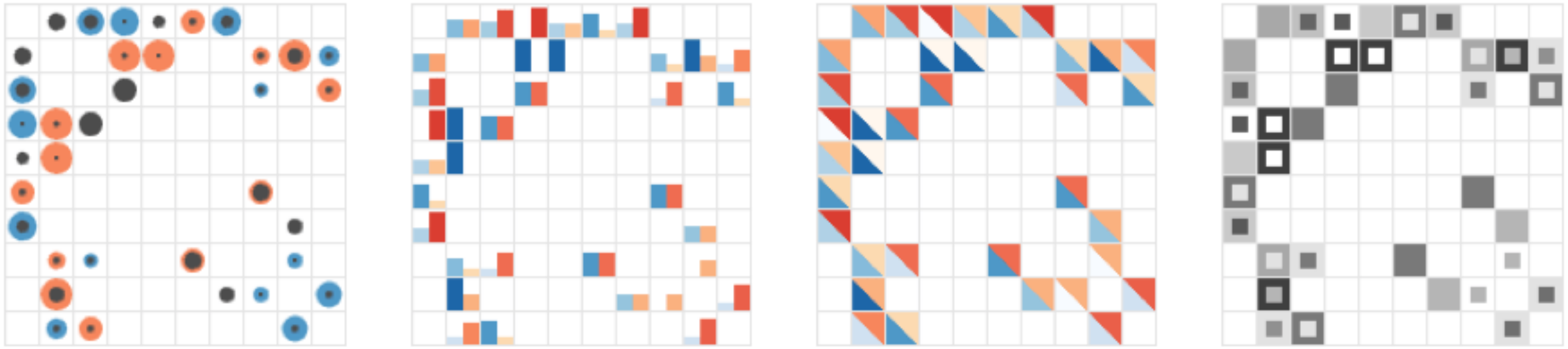


Name	Beverage	Day 1
Tom	Beer	5
Jon	Coke	4
Cole	Port	3
Mark	Beer	2
Abby	Port	1
Sue	Coke	0

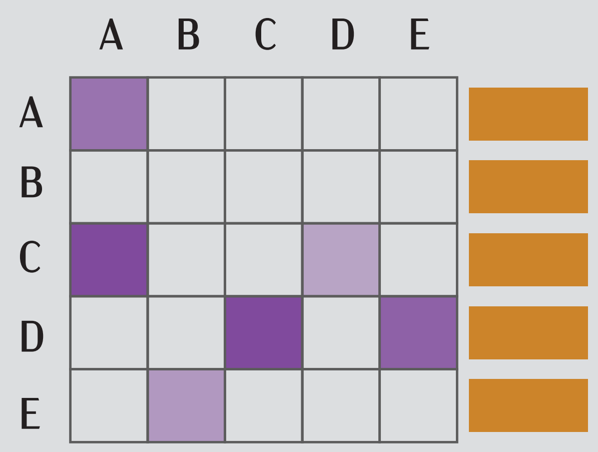
Moritz Stefaner, Musli Ingredient Network. https://truth-and-beauty.net/projects/muesli-ingredient-network



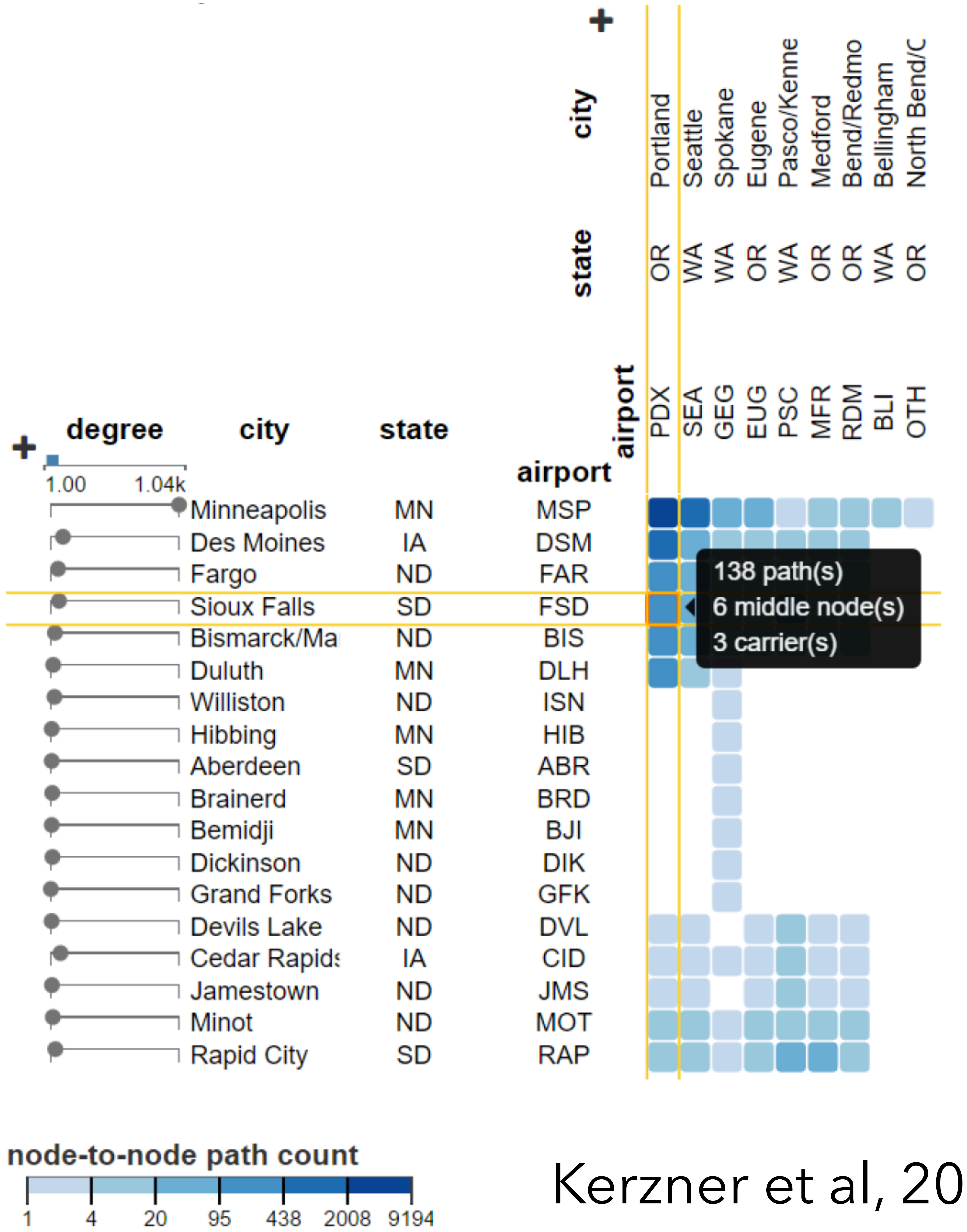
Adjacency Matrix



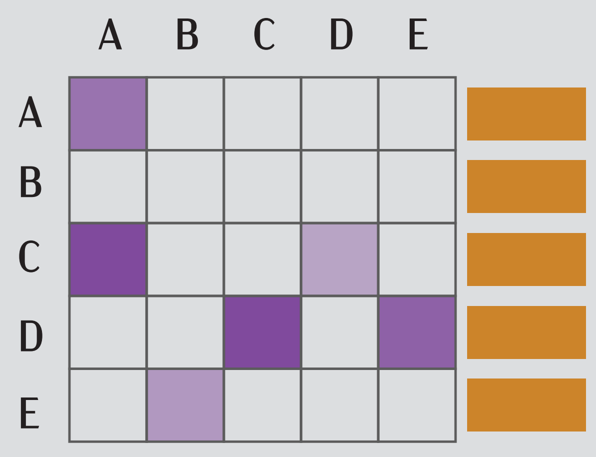
Alper et al, 2013



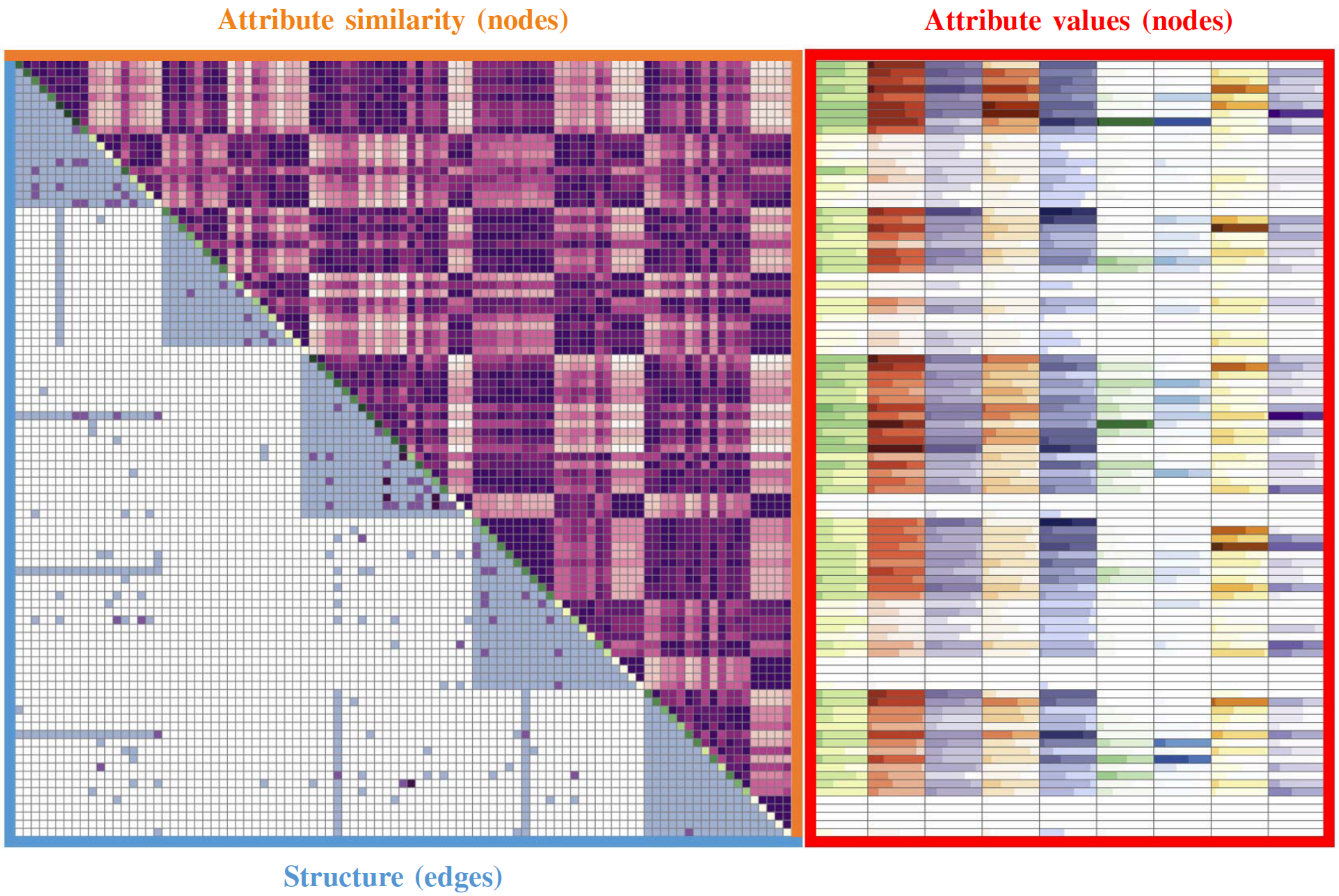
Adjacency Matrix



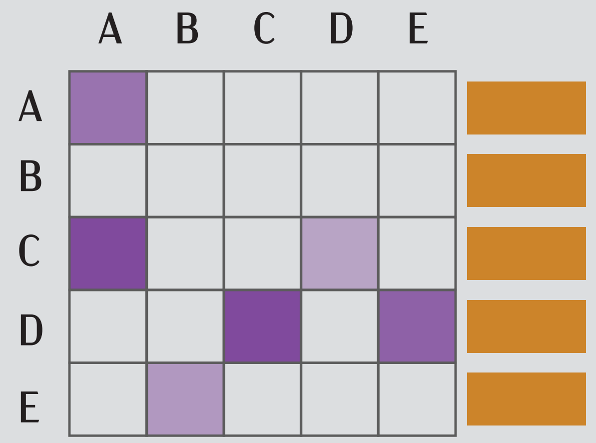
Kerzner et al, 2017



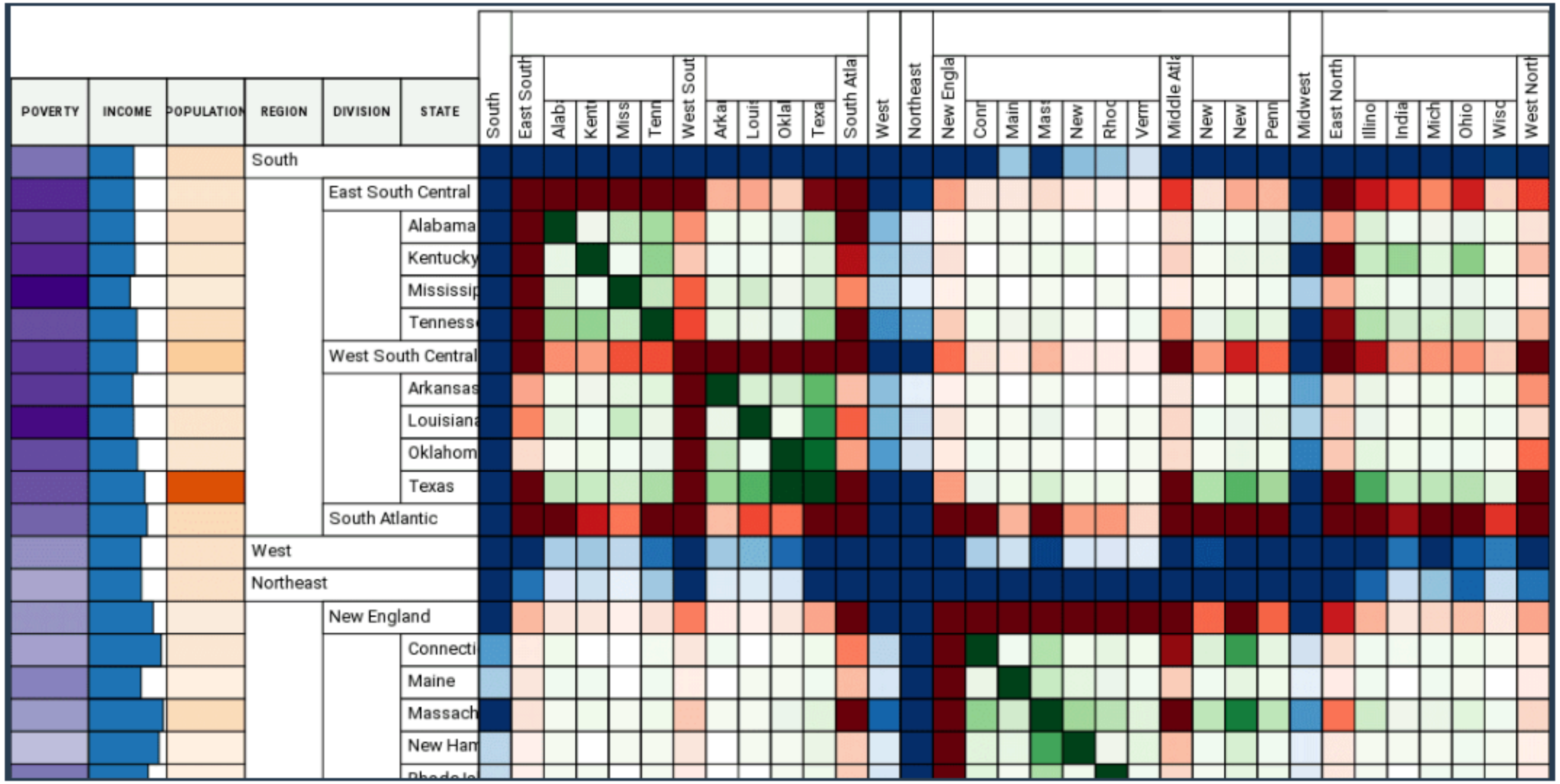
Adjacency Matrix

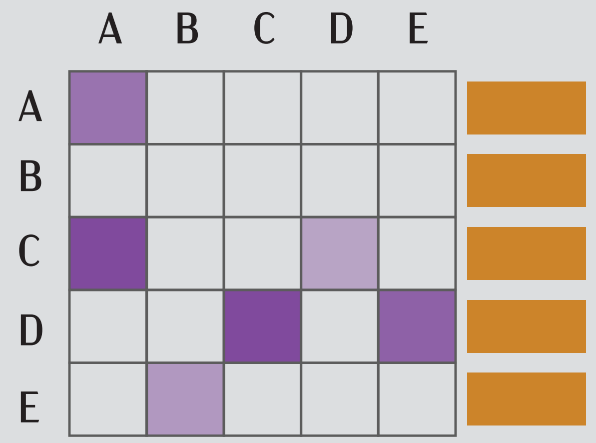


Berger et al, 2019



Adjacency Matrix





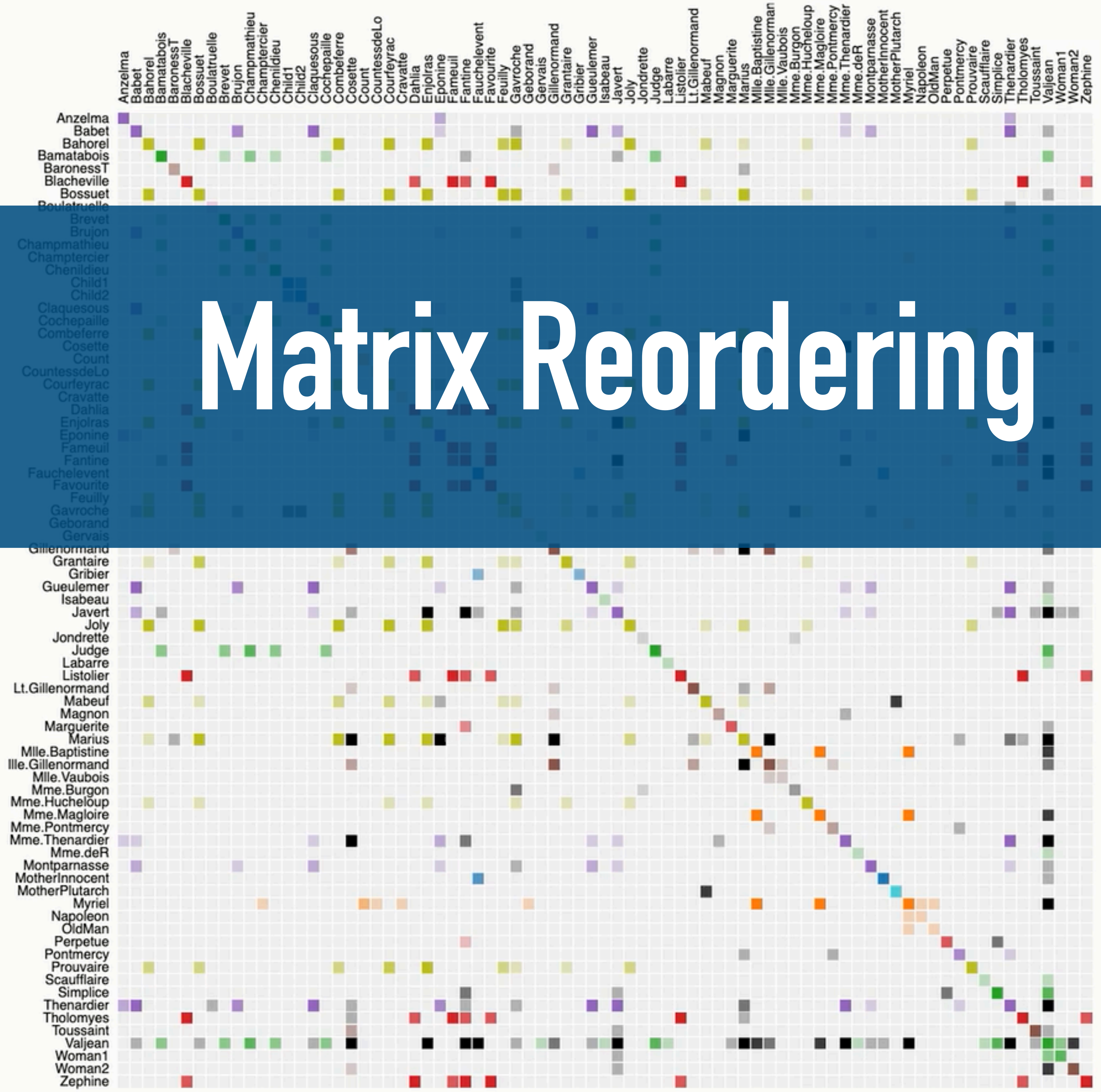
Adjacency Matrix

Safarli and Lex, 2019

April 10, 2012 / Mike Bostock

Les Misérables Co-occurrence

Matrix Reordering



Order: by Name

This matrix diagram visualizes character co-occurrences in Victor Hugo's *Les Misérables*.

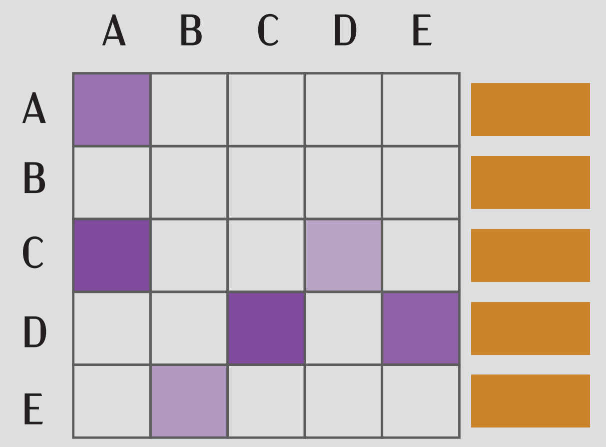
Each colored cell represents two characters that appeared in the same chapter; darker cells indicate characters that co-occurred more frequently.

Use the drop-down menu to reorder the matrix and explore the data.

Built with d3.js.

Source: The Stanford GraphBase.


Code
Issues 2
Pull requests 1
Projects 0
Wiki
Security
Insights
Home
Edit
New Page
Jean-Daniel Fekete edited this page on Apr 23, 2015 · 2 revisions
Reorder.js is a library to reorder tables and graph/networks.
Resources
Introduction
API Reference
Browser / Platform Support
Reorder.js is mainly developed on Chrome and Node.js. Use npm install reorder.js to install, and require("reorder") to load.
Installing
Download the latest version here:
https://github.com/jdfekete/reorder.js/releases
Add a custom footer
Pages 12
Find a Page...
Home
API Reference
Conversion
Core
Gallery
Graph
Introduction
LinearAlgebra
Matrix
Measure
Permutation
Reordering
Add a custom sidebar
Reorder.js

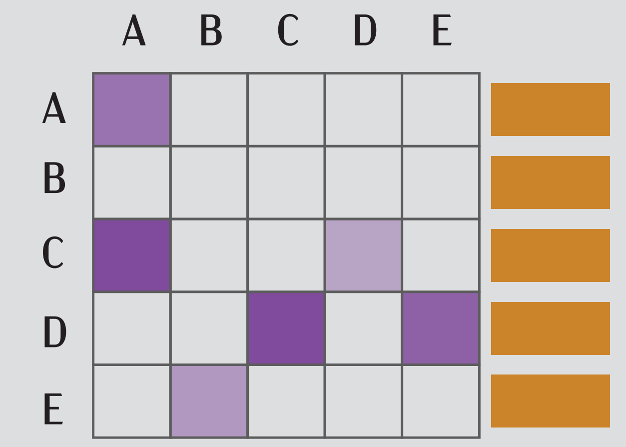


Adjacency Matrix

Ideal for dense and completely connected networks

Requires quadratic space with respect to the number of nodes.

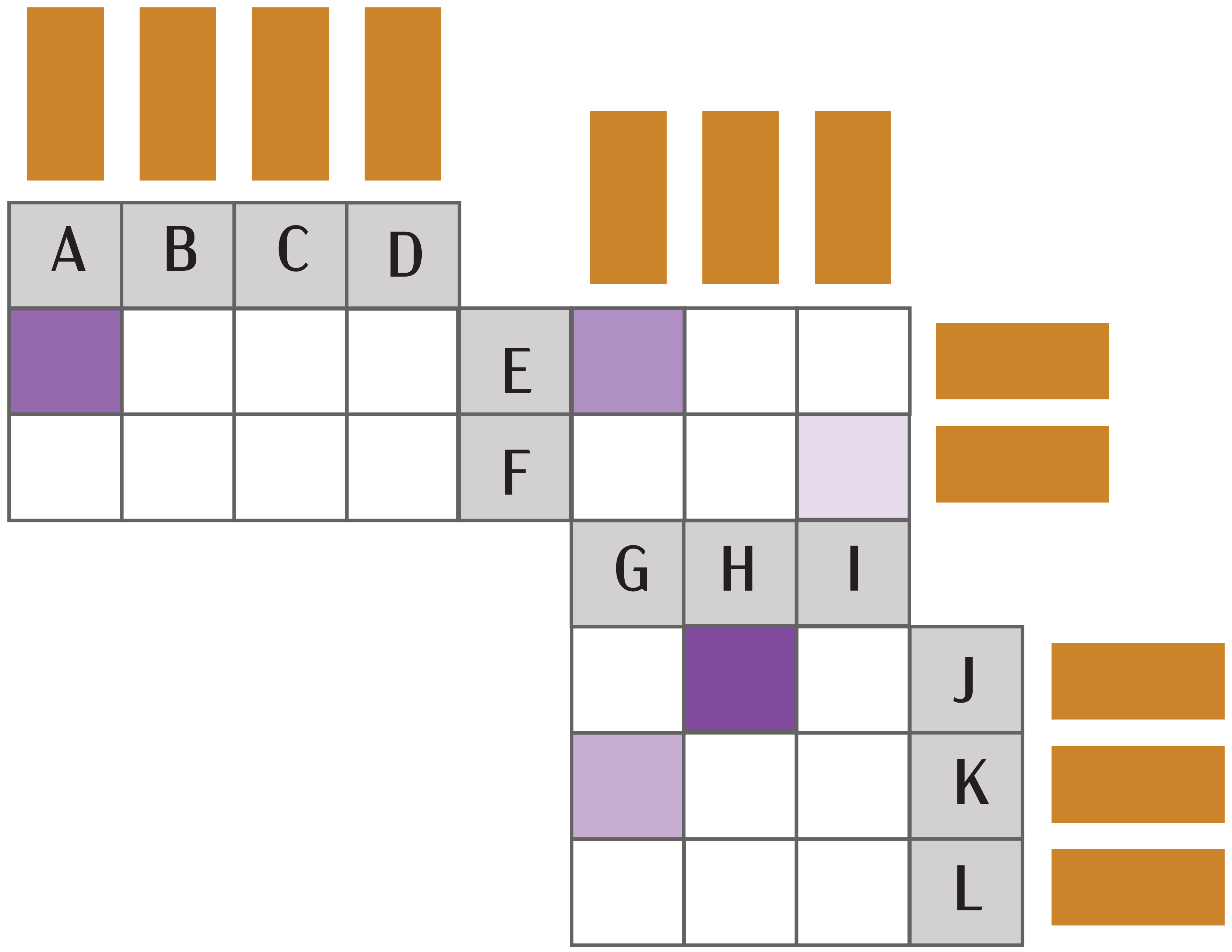
Complexity of choosing the right reordering algorithm

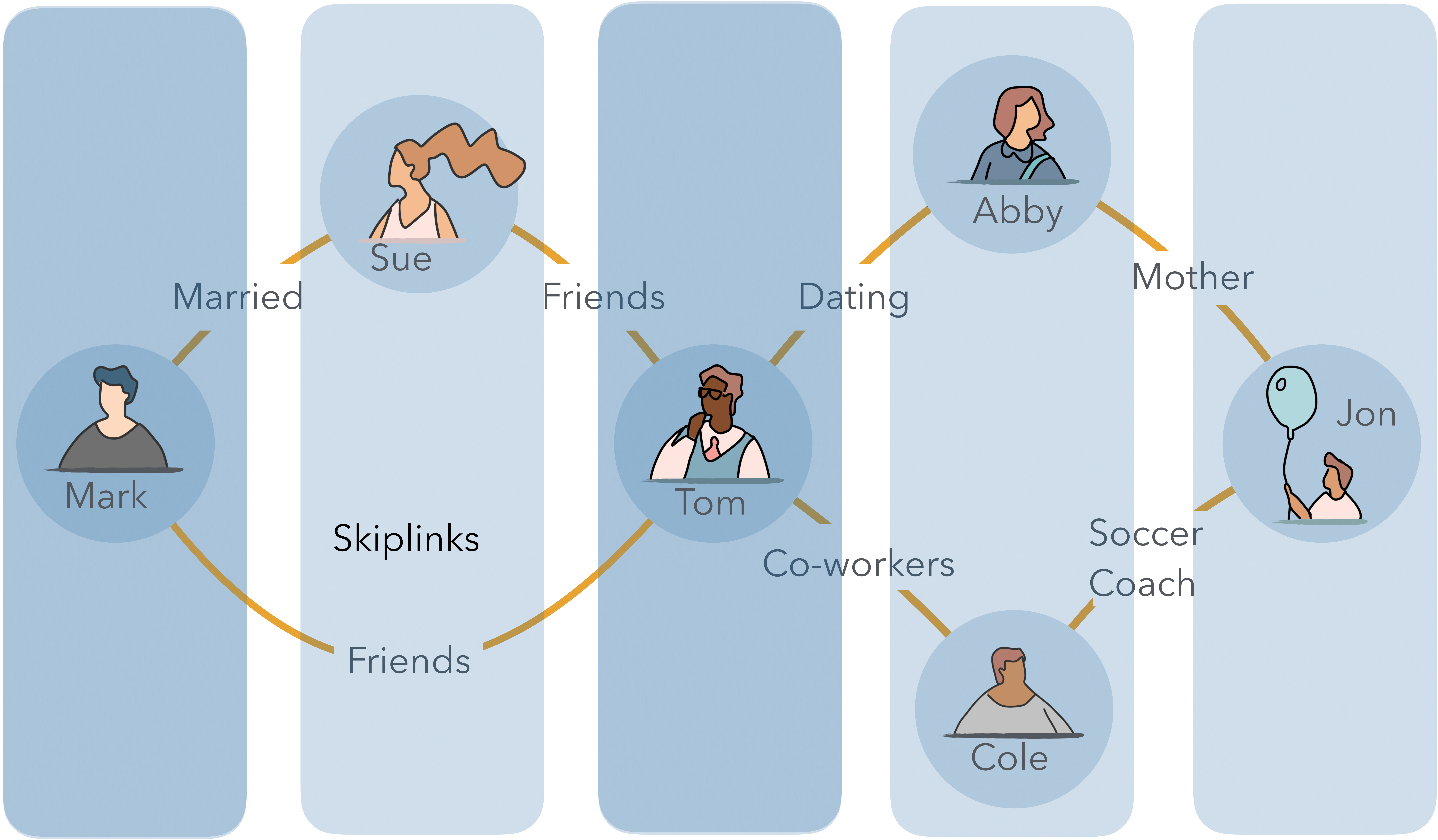


Adjacency Matrix

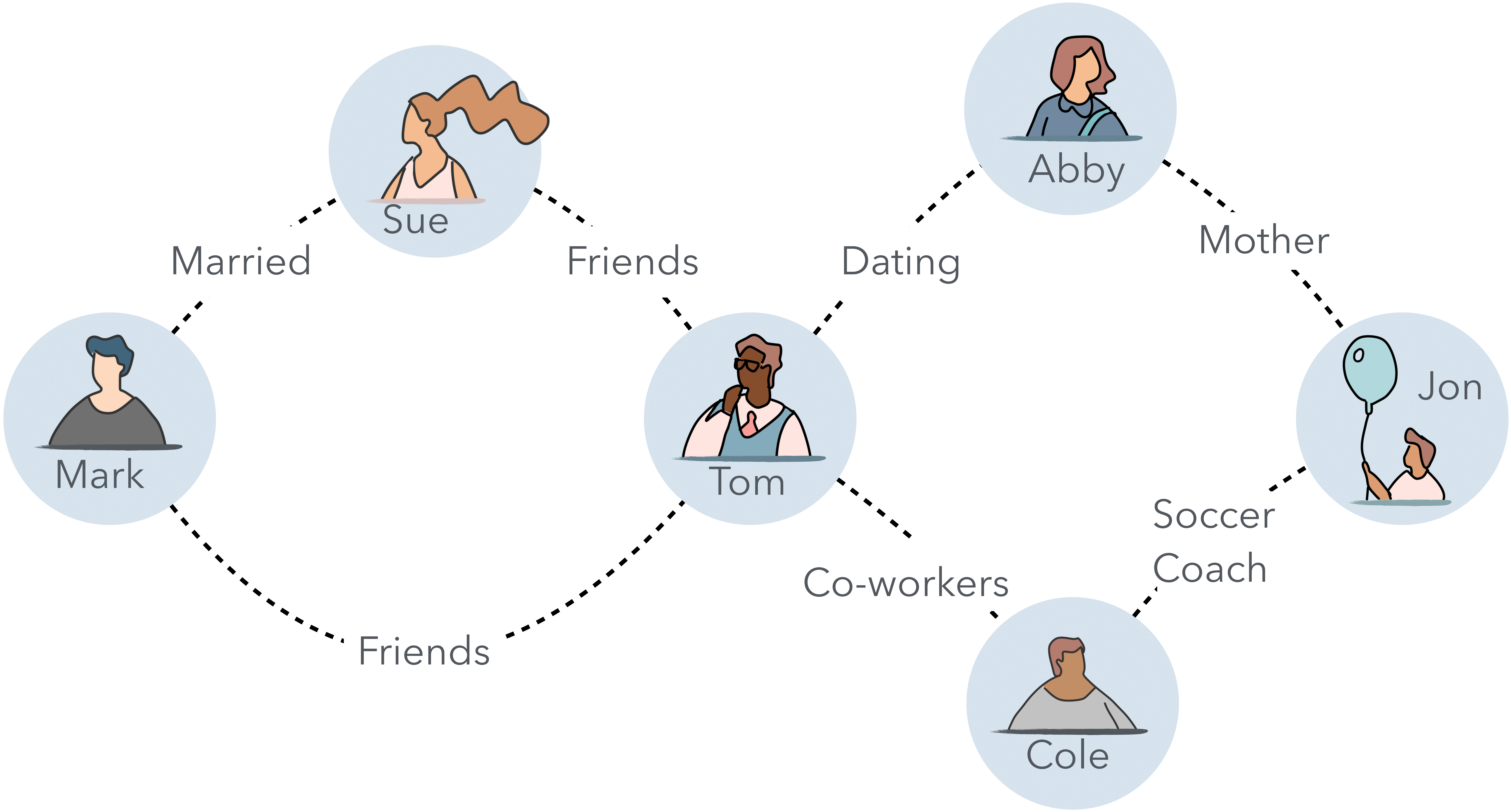
Recommended for smaller, complex and dense networks with rich node and/or edge attributes, for all tasks except for those involving paths

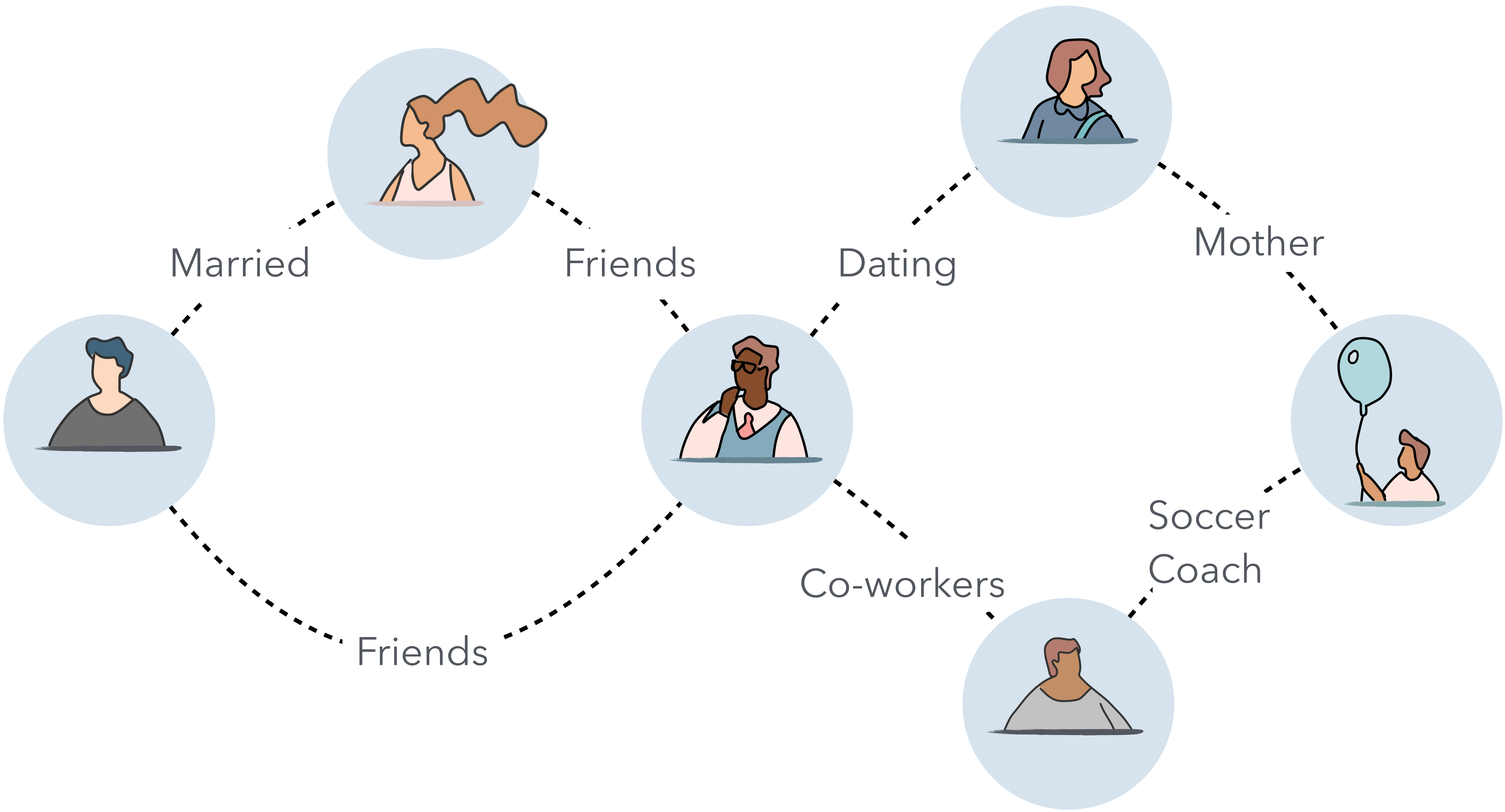
Quilts

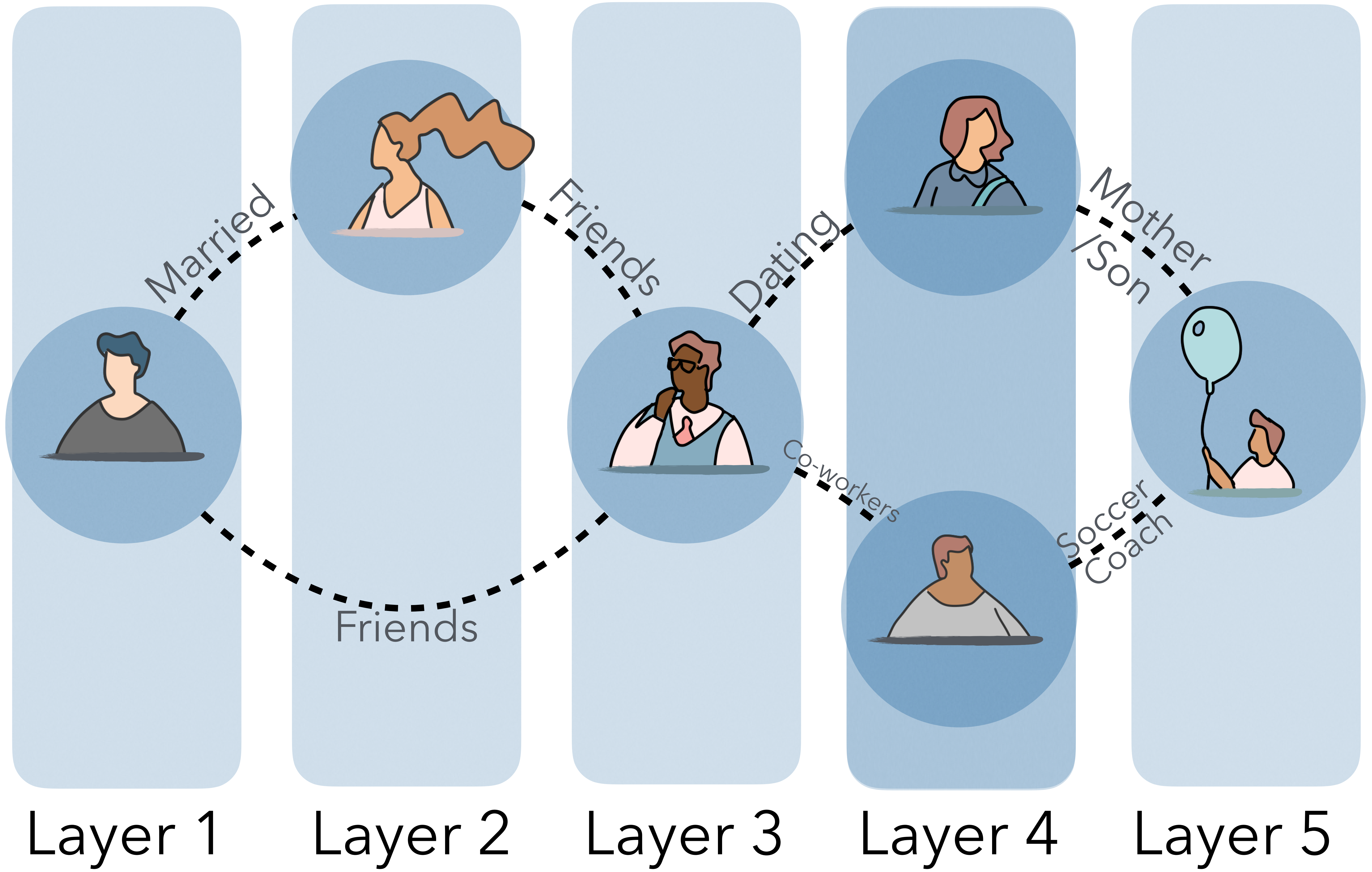




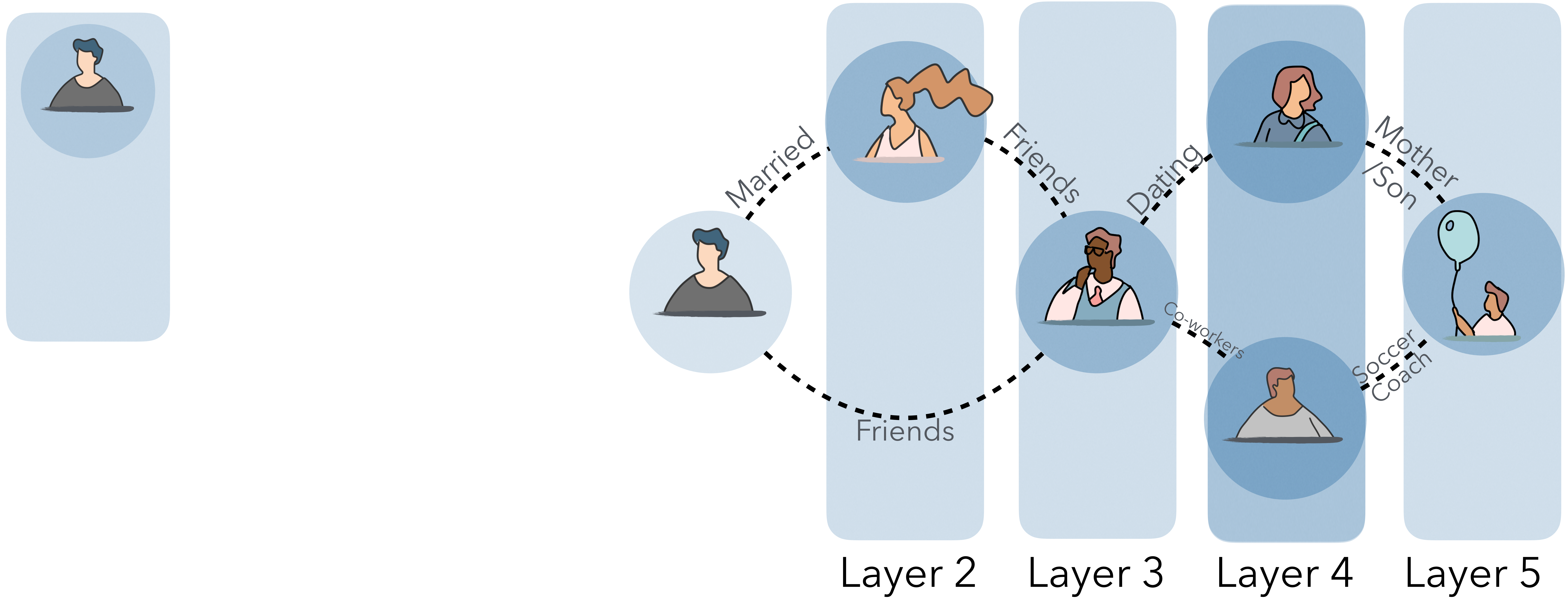
Sue
Abby
Married
Friends
Dating
Mother
Jon
Mark
Tom
Skiplinks
Co-workers
Soccer
Coach
Friends
Cole


Sue
Married
Friends
Abby
Dating
Mother
Mark
Tom
Jon
Soccer Coach
Co-workers
Friends
Cole

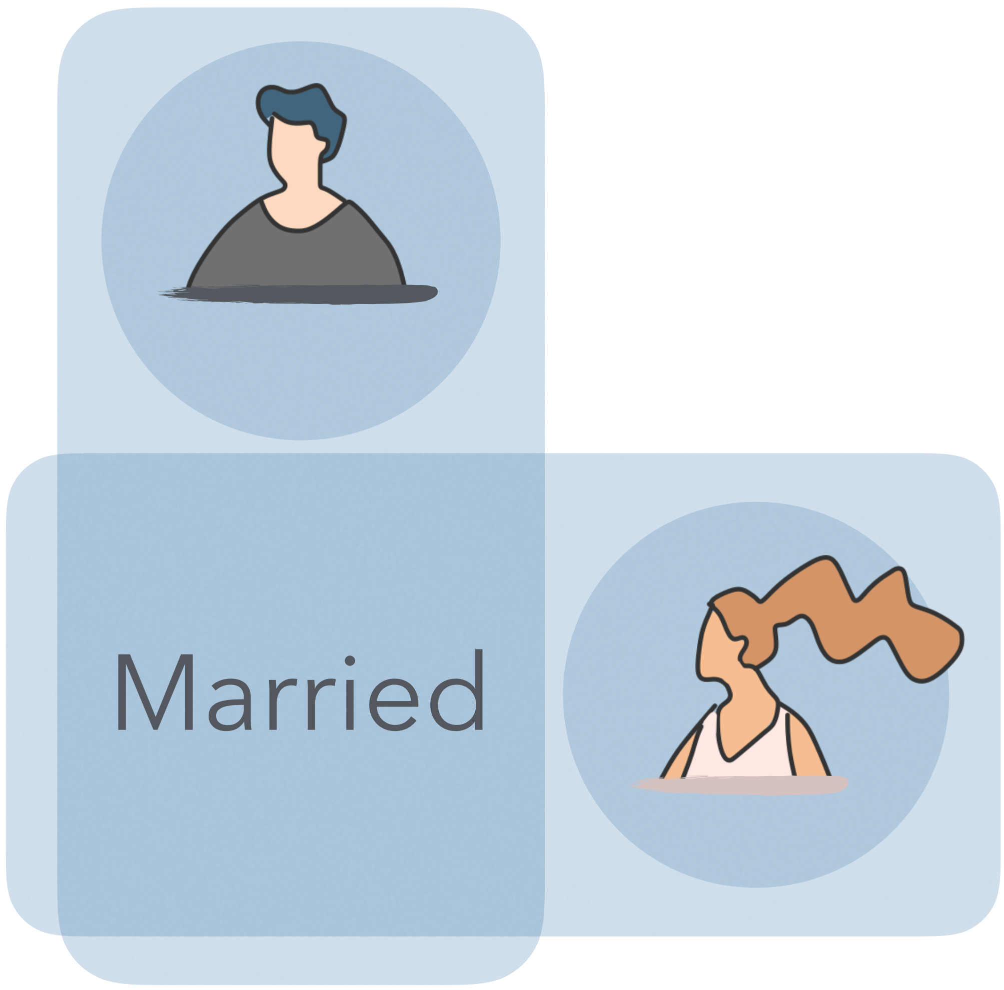

Married
Friends
Dating
Mother
Friends
Co-workers
Soccer Coach

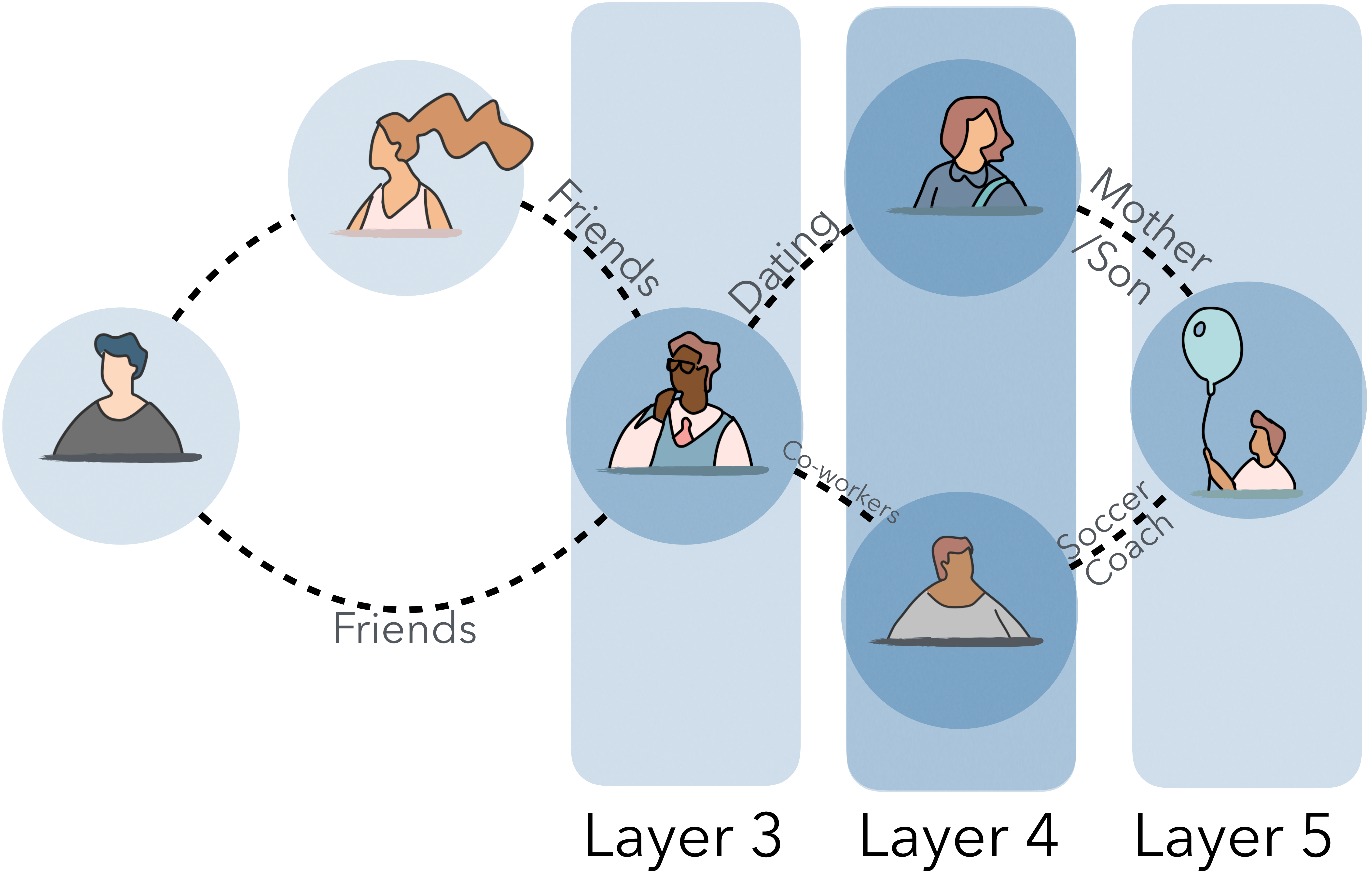


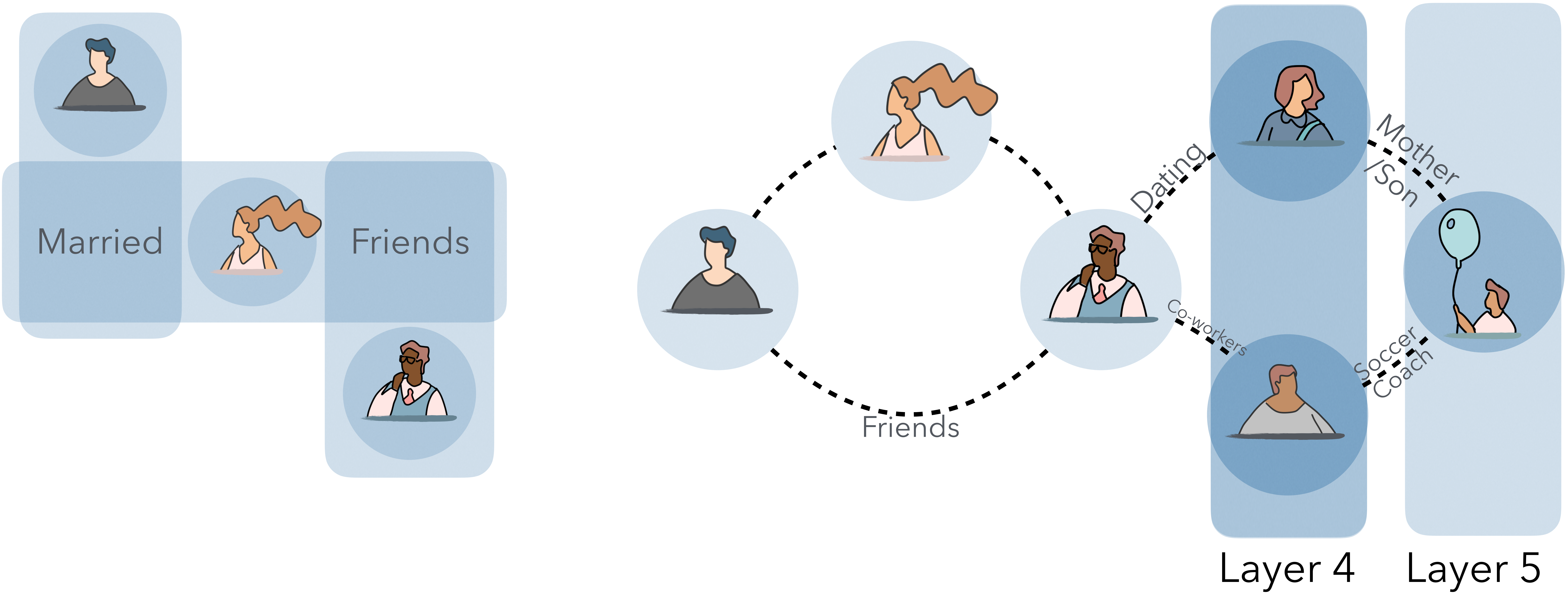
Married
Friends
Dating
Mother
/Son
Co-workers
Soccer
Coach
Friends
Layer 1
Layer 2
Layer 3
Layer 4
Layer 5



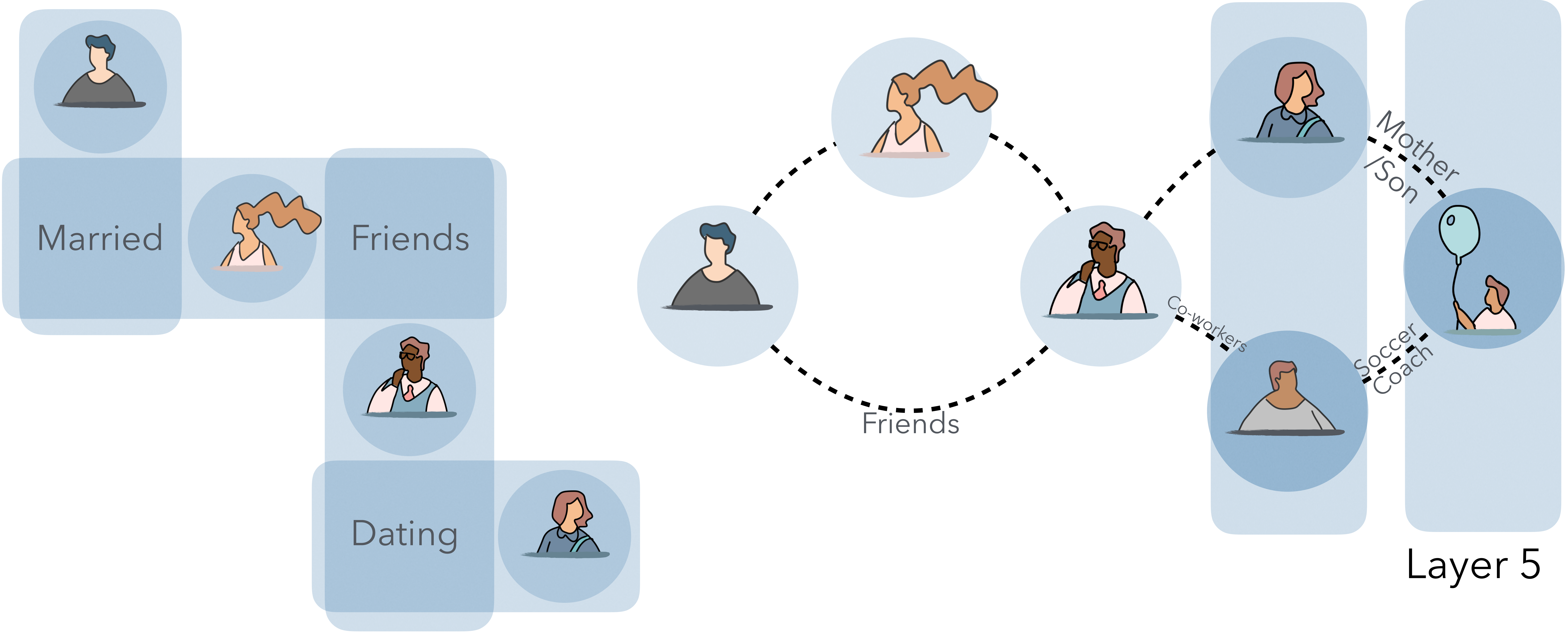
Married
Friends
Dating
Mother
/Son
Co-workers
Soccer
Coach
Friends
Layer 2
Layer 3
Layer 4
Layer 5

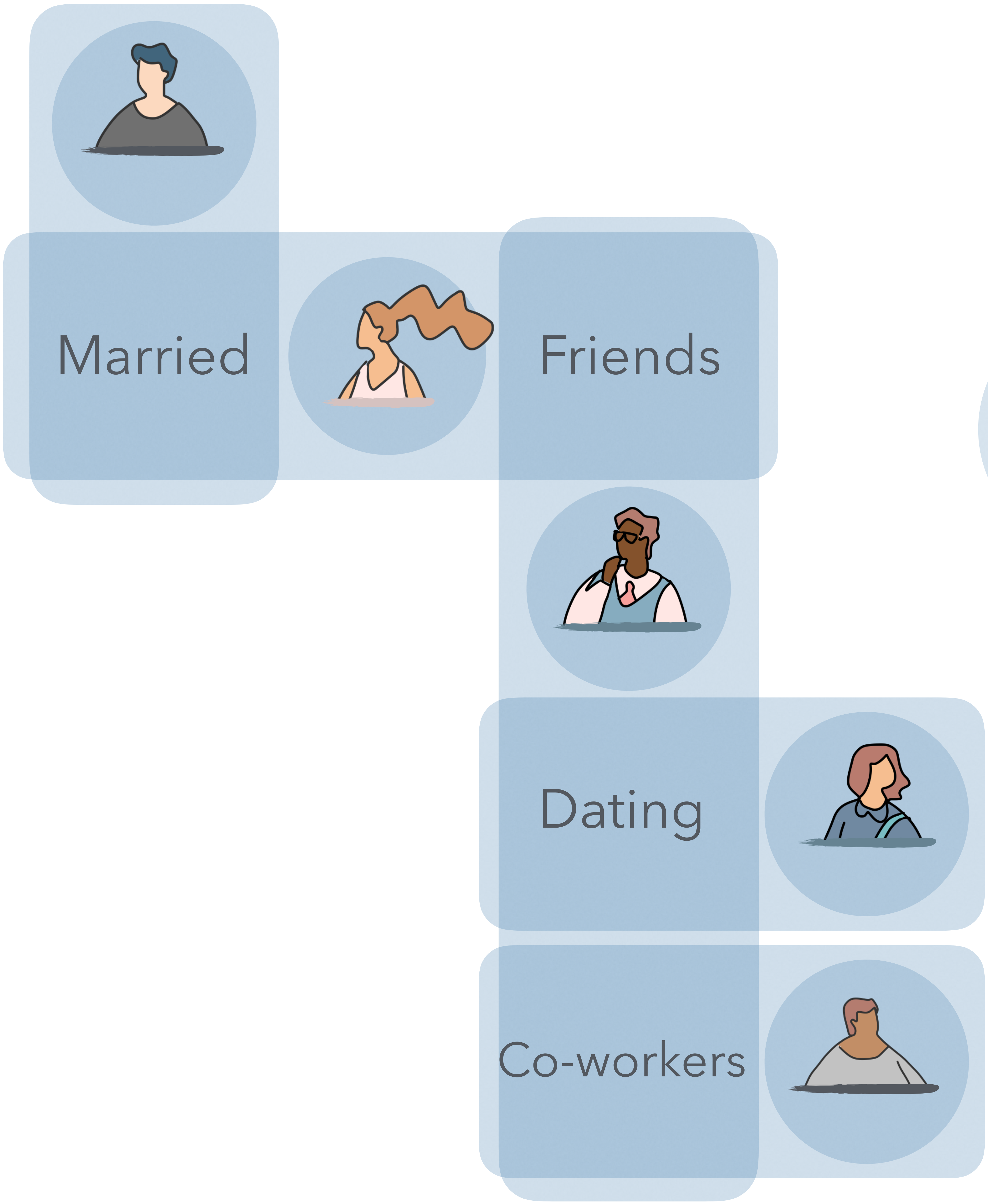

Married

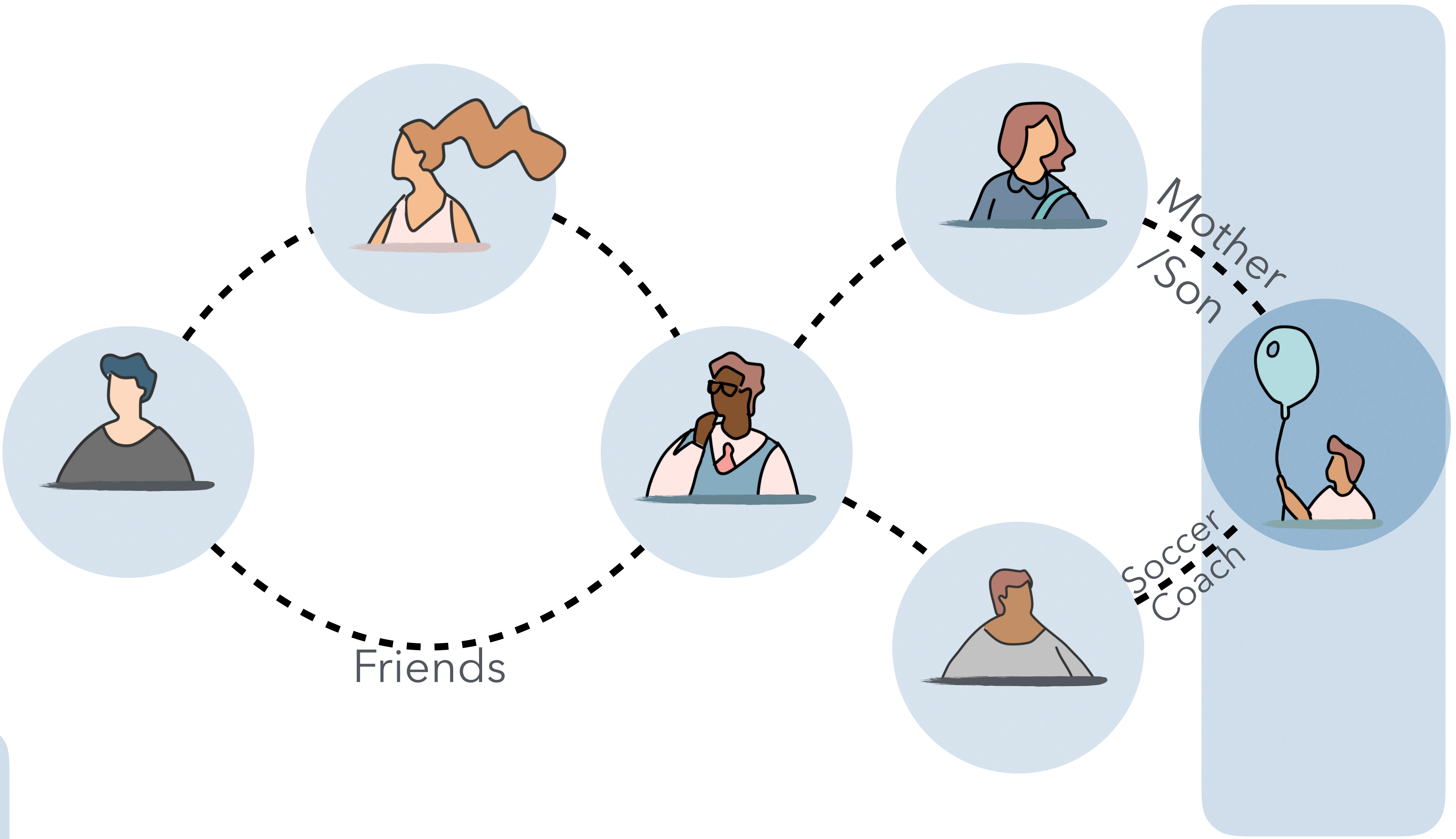

Friends
Friends
Dating
Mother
/Son
Co-workers
Soccer
Coach
Layer 3
Layer 4
Layer 5



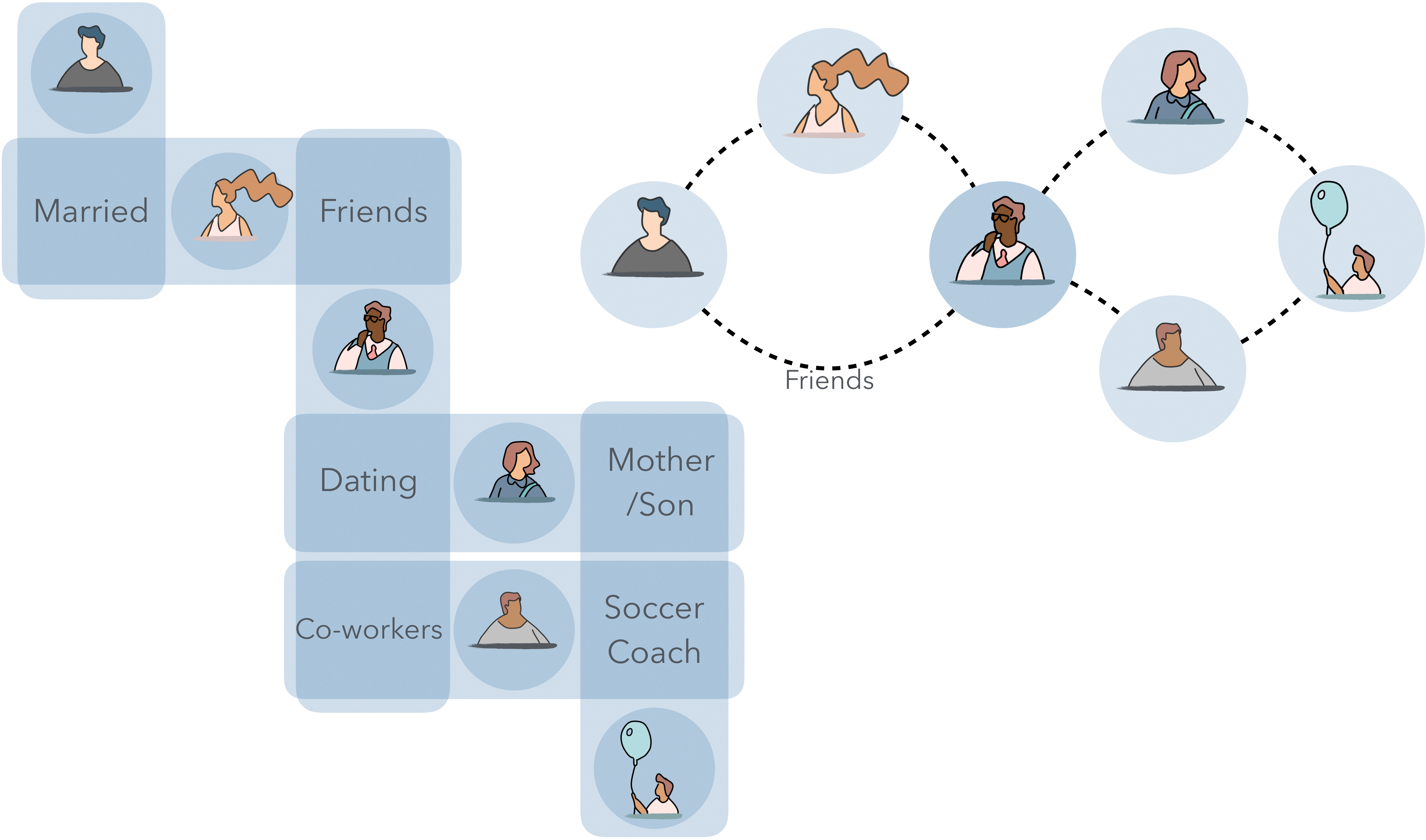
Married
Friends
Dating
Mother /Son
Co-workers
Soccer Coach
Friends
Layer 4
Layer 5

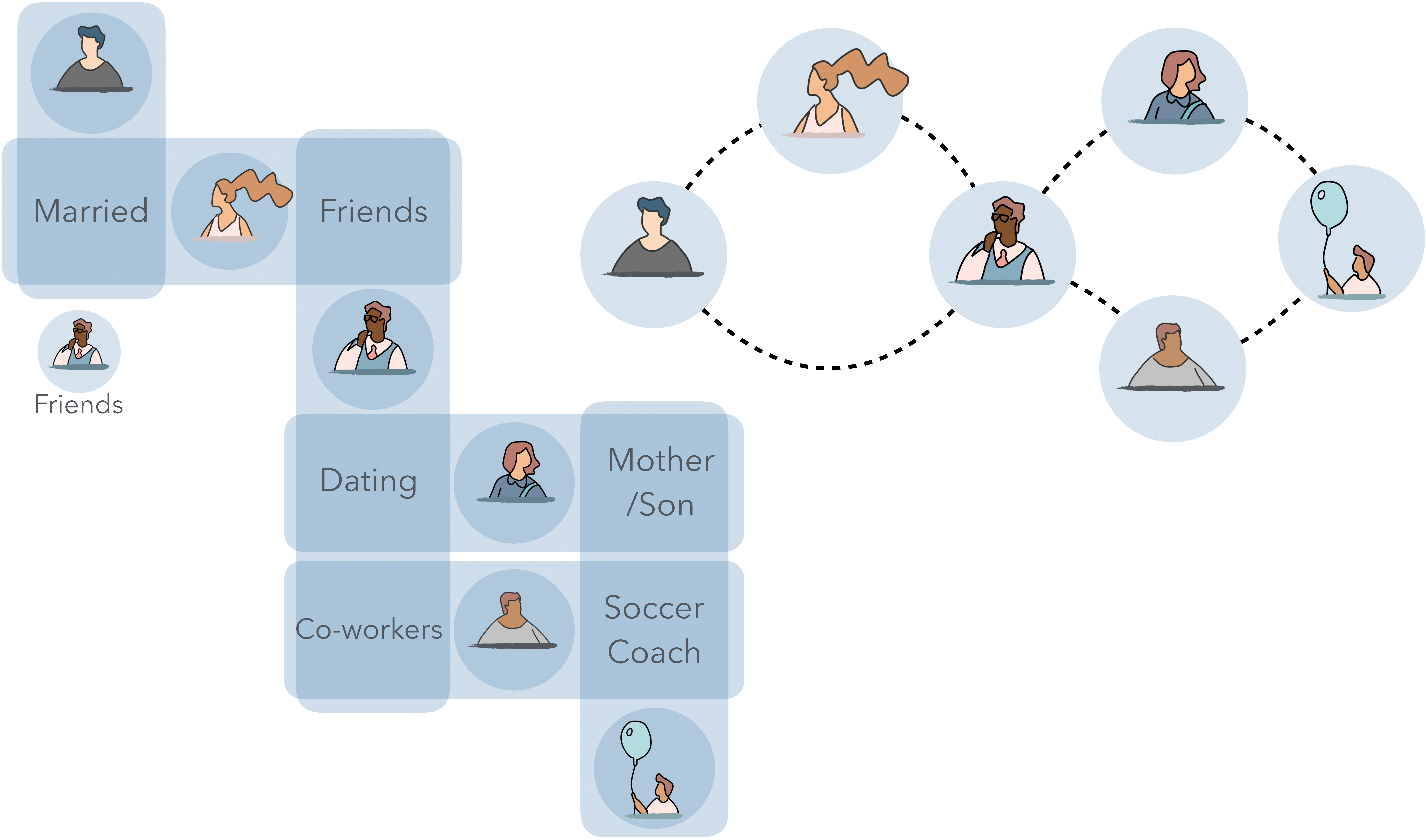

Married
Friends
Dating
Friends
Co-workers
Mother
/Son
Soccer
Coach
Layer 5


Married
Friends
Dating
Co-workers

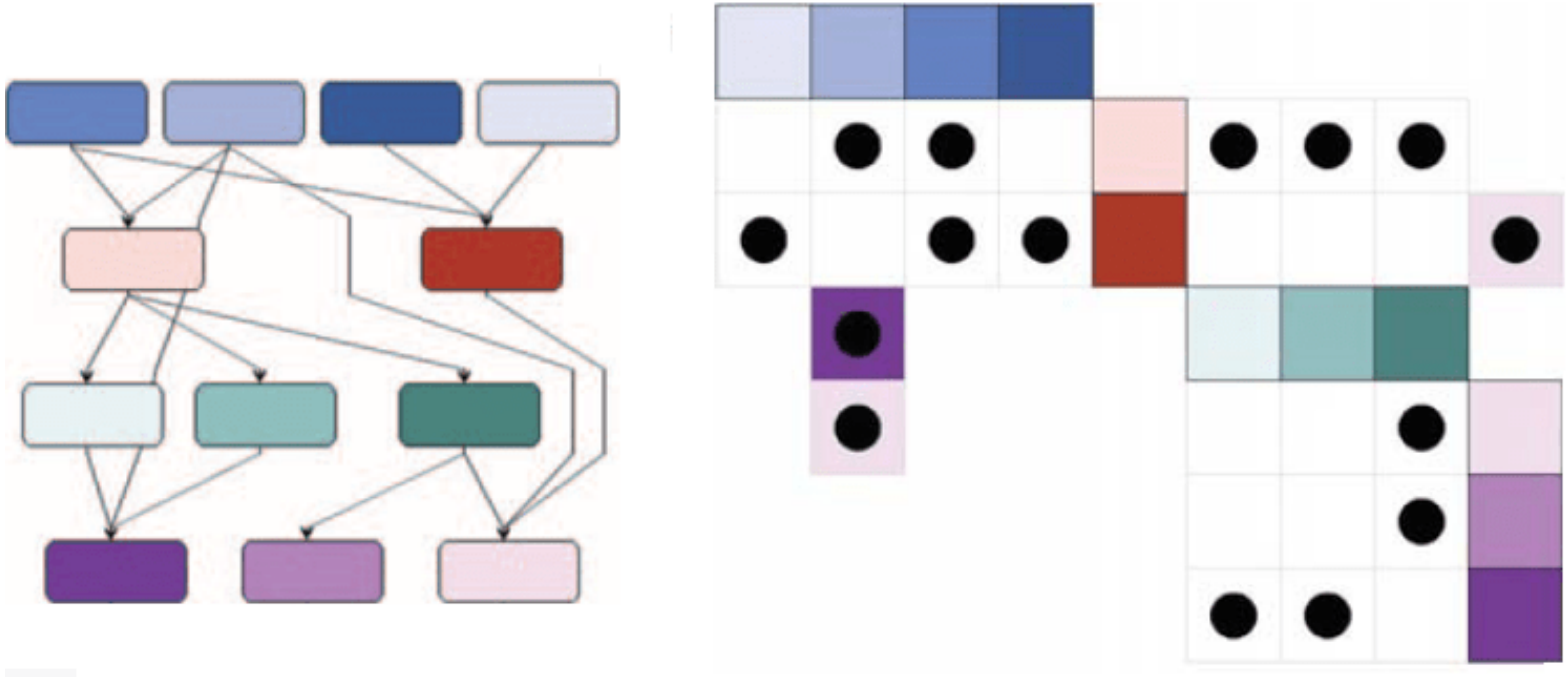

Friends
Mother
/Son
Soccer
Coach

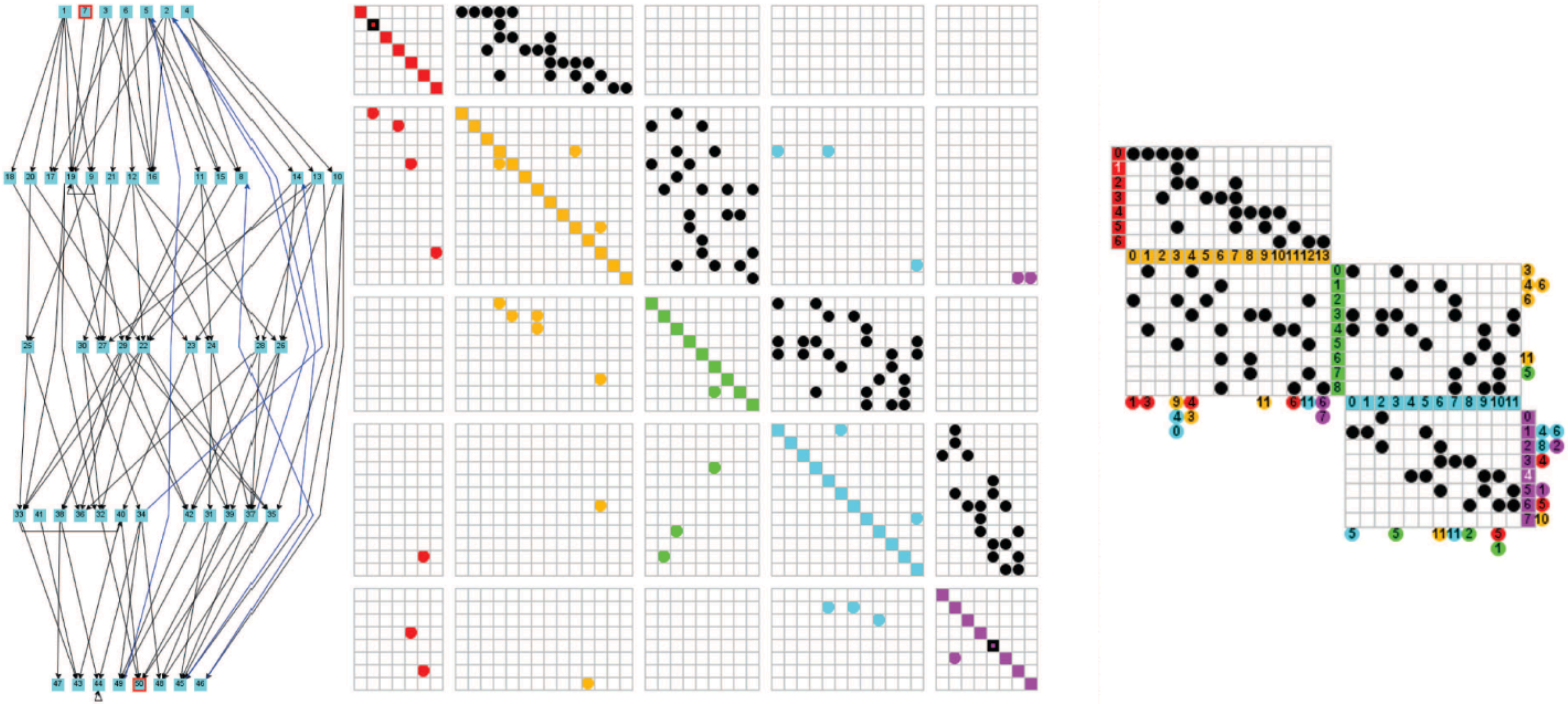
Layer 5

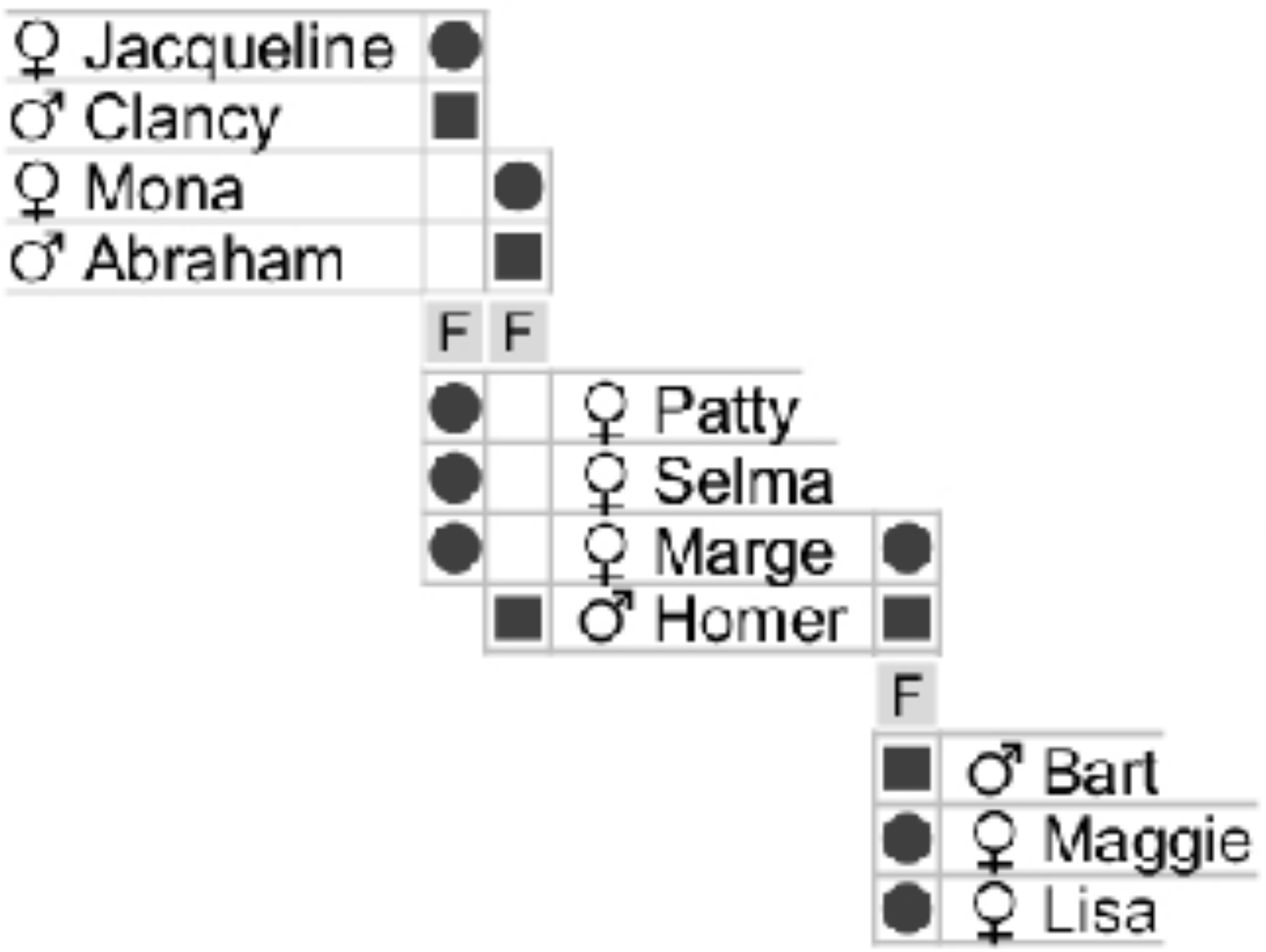

Married
Friends
Dating
Mother
/Son
Co-workers
Soccer
Coach
Friends

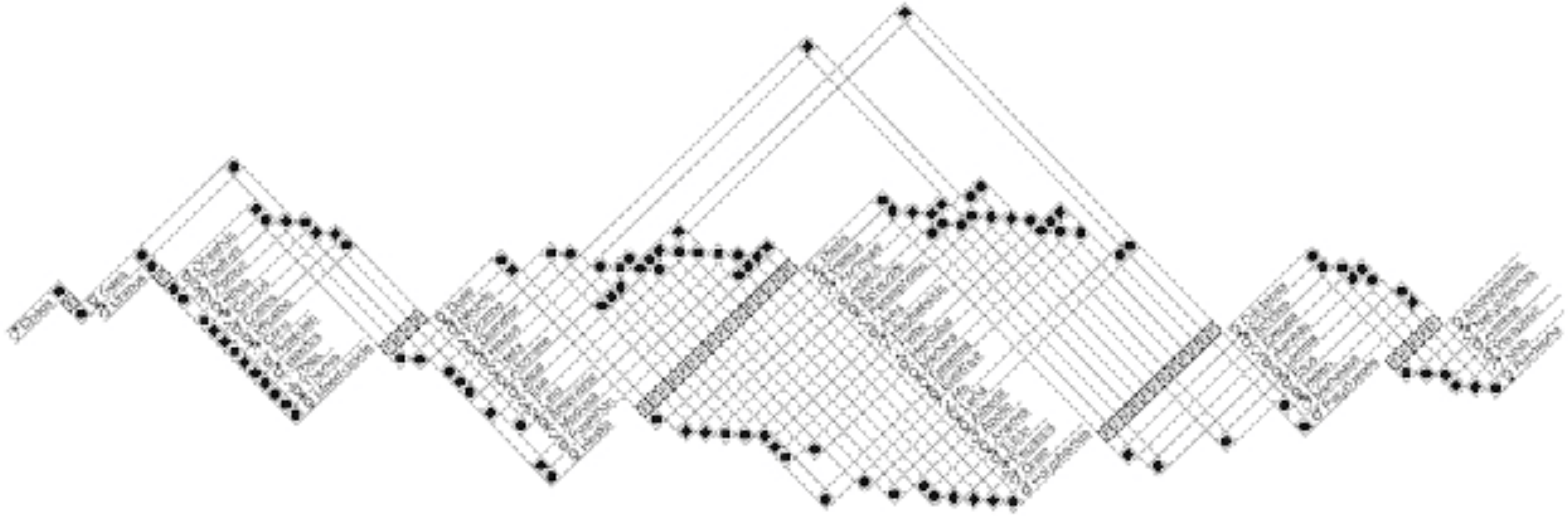


Married
Friends
Friends
Dating
Mother /Son
Co-workers
Soccer Coach






♀ Jacqueline
♂ Clancy
♀ Mona
♂ Abraham
F F
♀ Patty
♀ Selma
♀ Marge
♂ Homer
F
♂ Bart
♀ Maggie
♀ Lisa

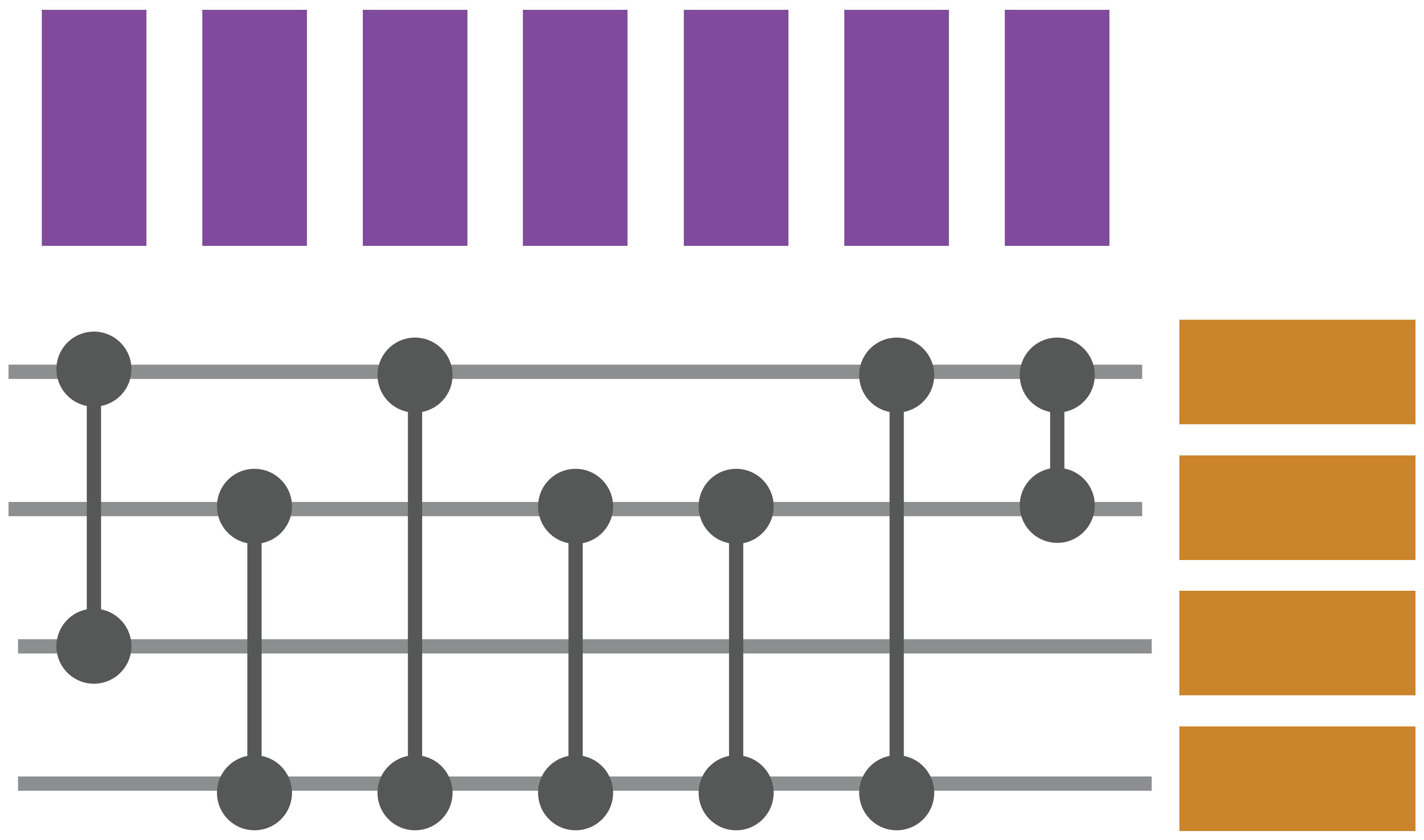


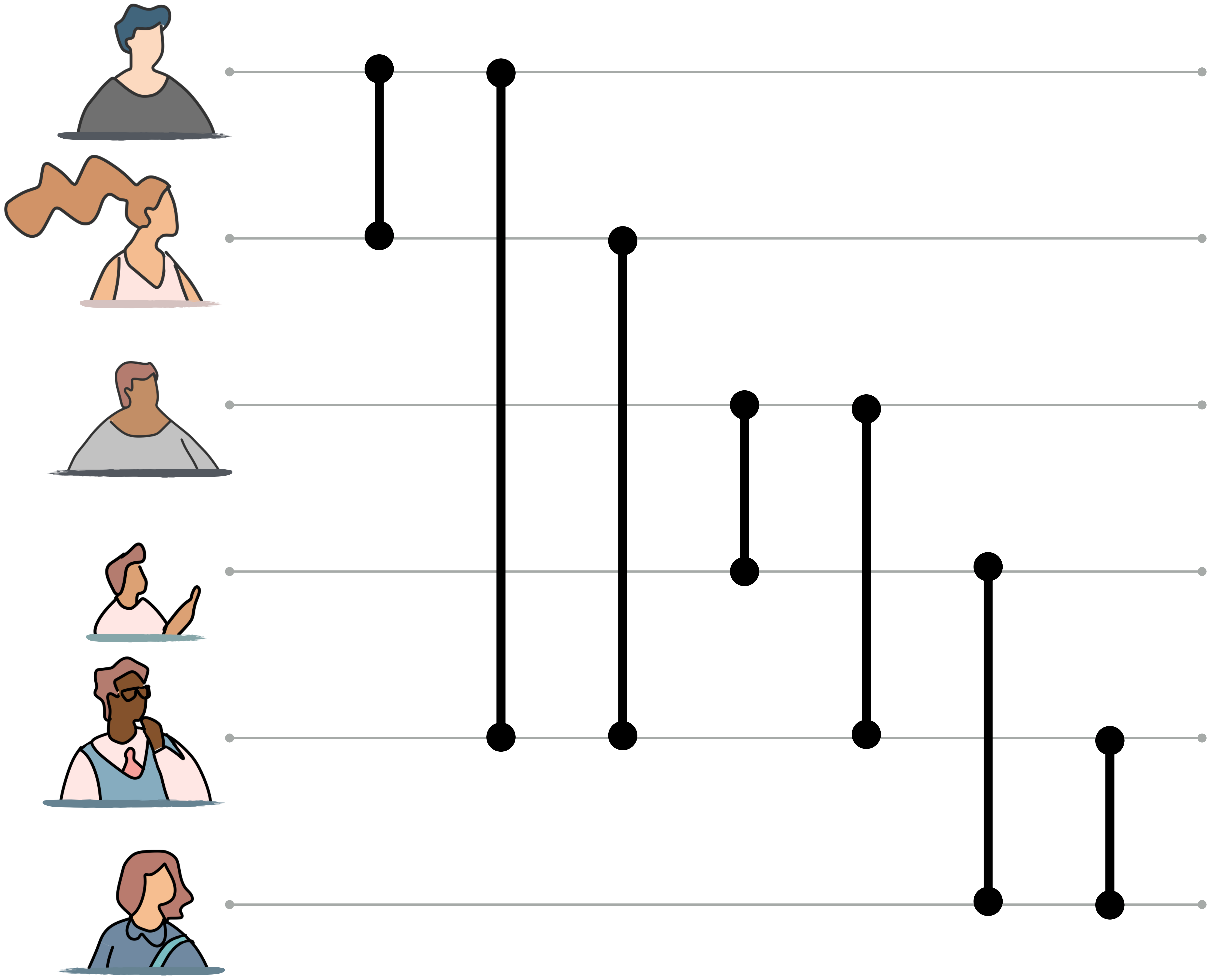
Well suited for layered networks

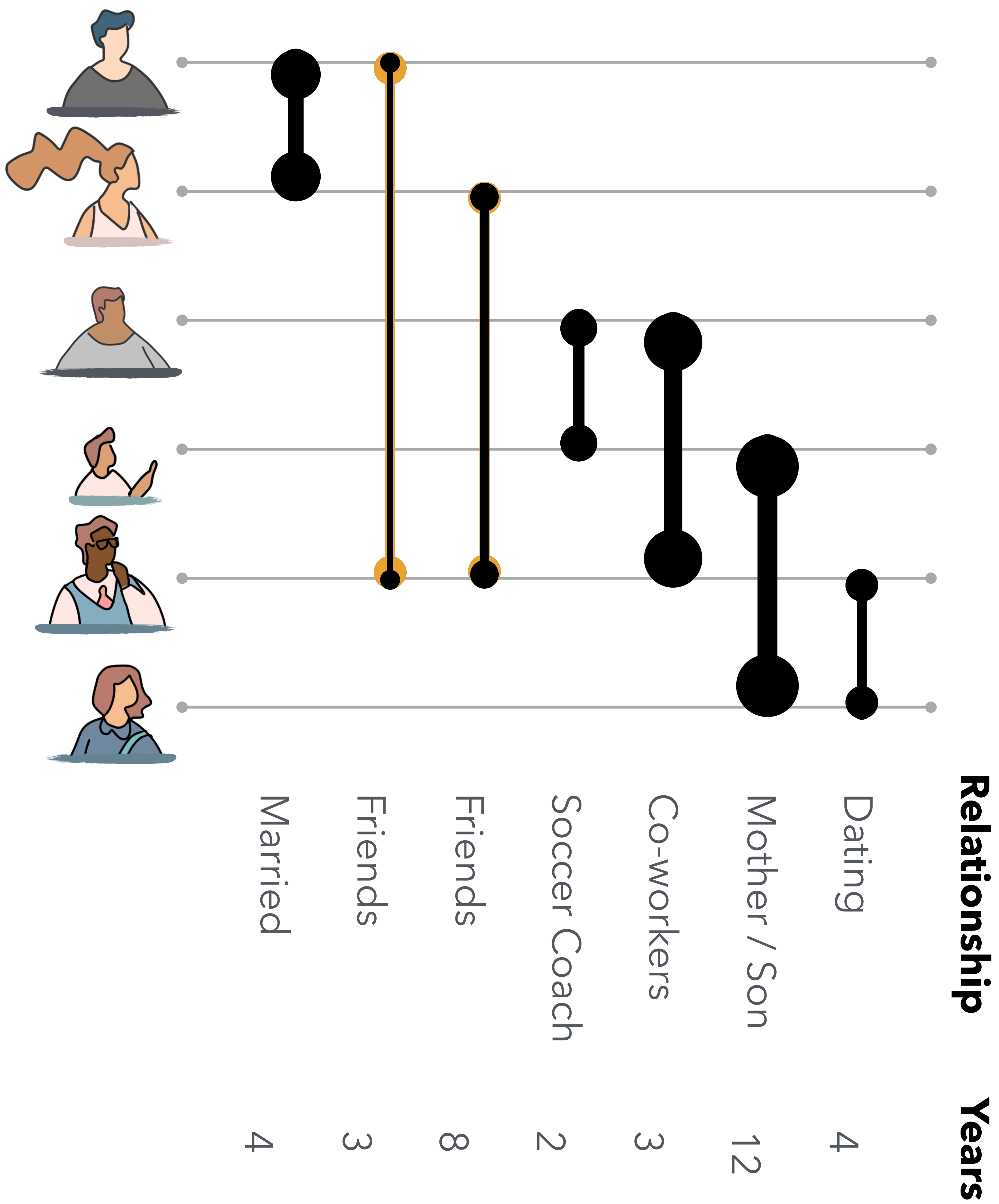
Links between nonconsecutive layers can be problematic to integrate and non-intuitive

Recommended for layered or k-partite networks with limited skiplinks.

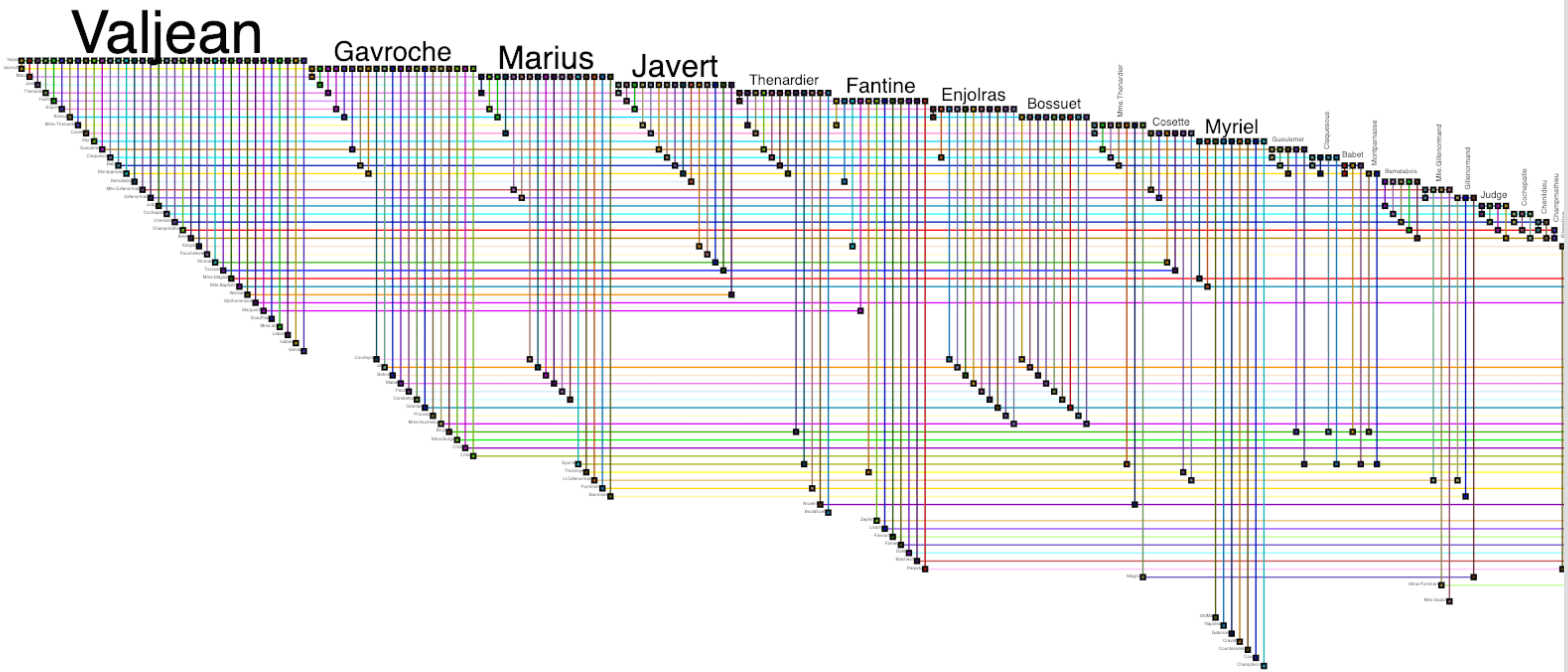
BioFabric

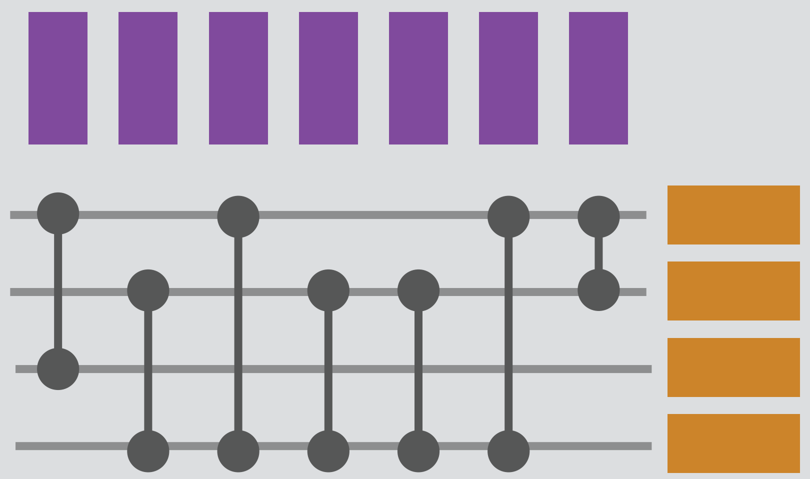






Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3

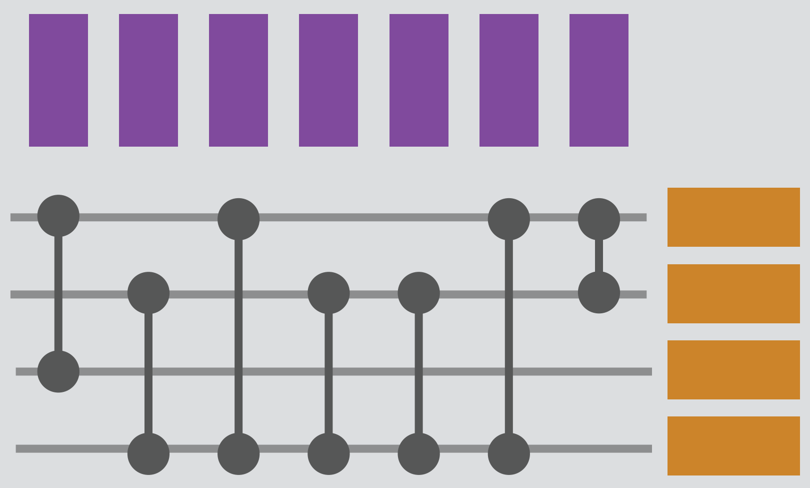




BioFabric

Longabaugh, 2012

BioFabric

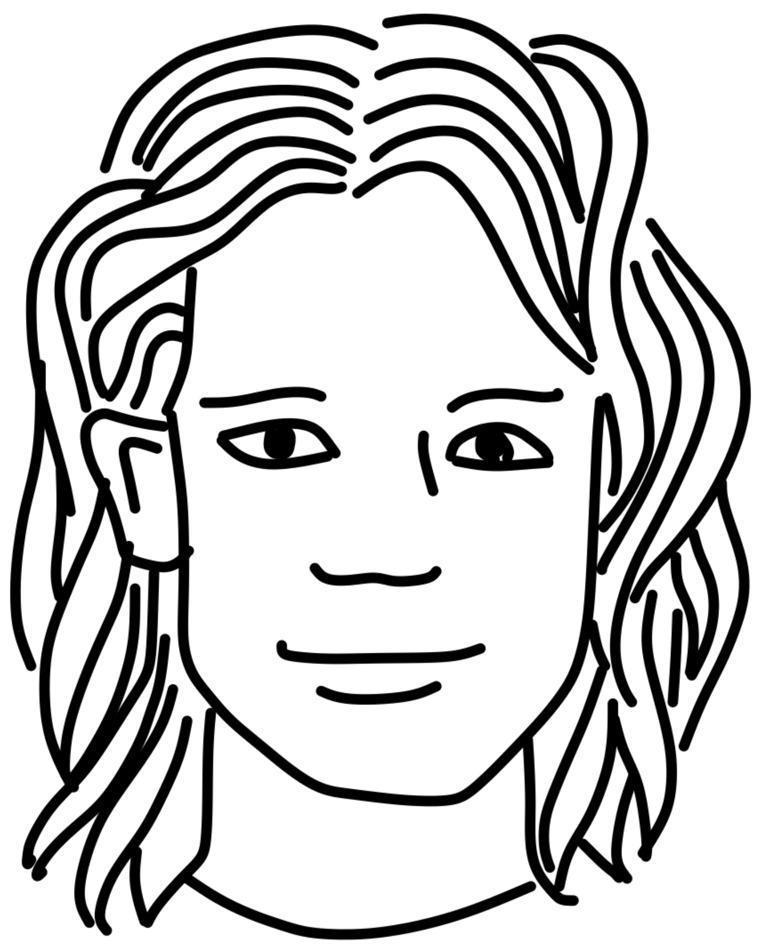

BioFabric

Can be used to visualize rich edge attributes and node attributes at the same time

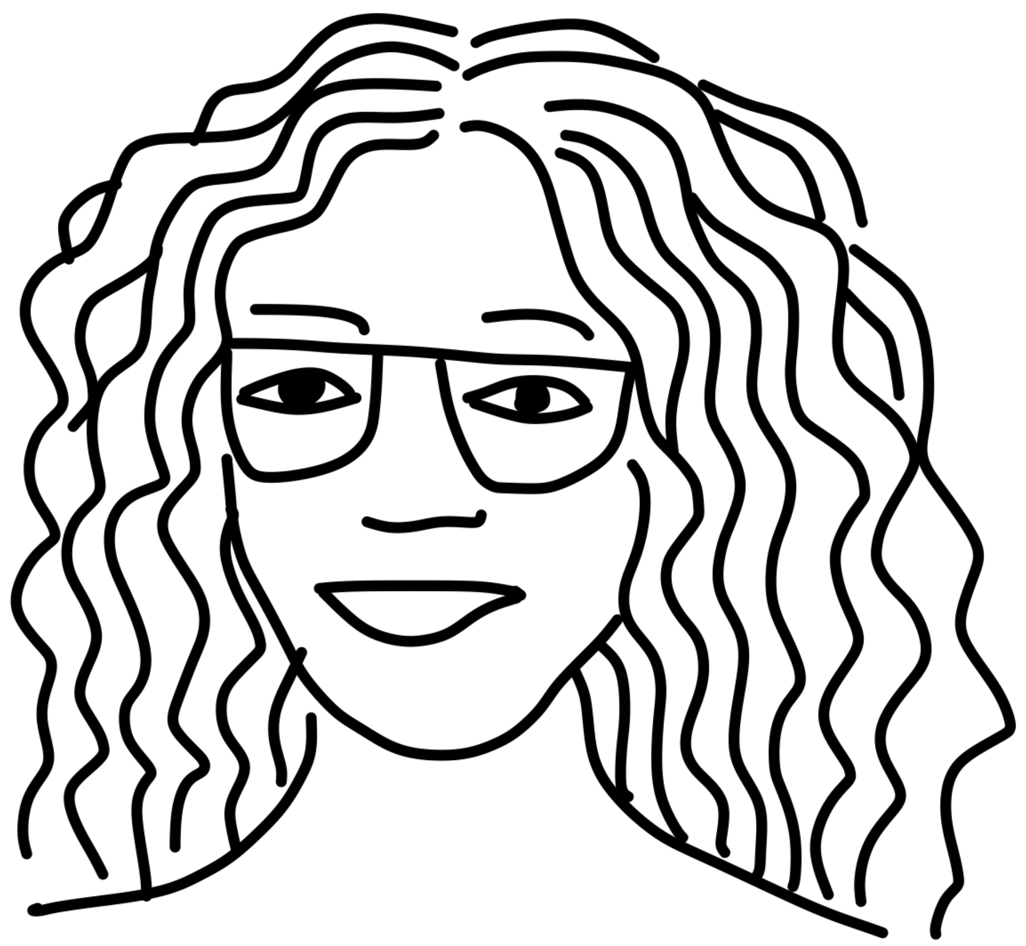
More difficult to discover neighbors and clusters in Biofabric compared to matrices.

Recommended for small, sparse networks with many nodes and rich edge attributes

Tools and Applications

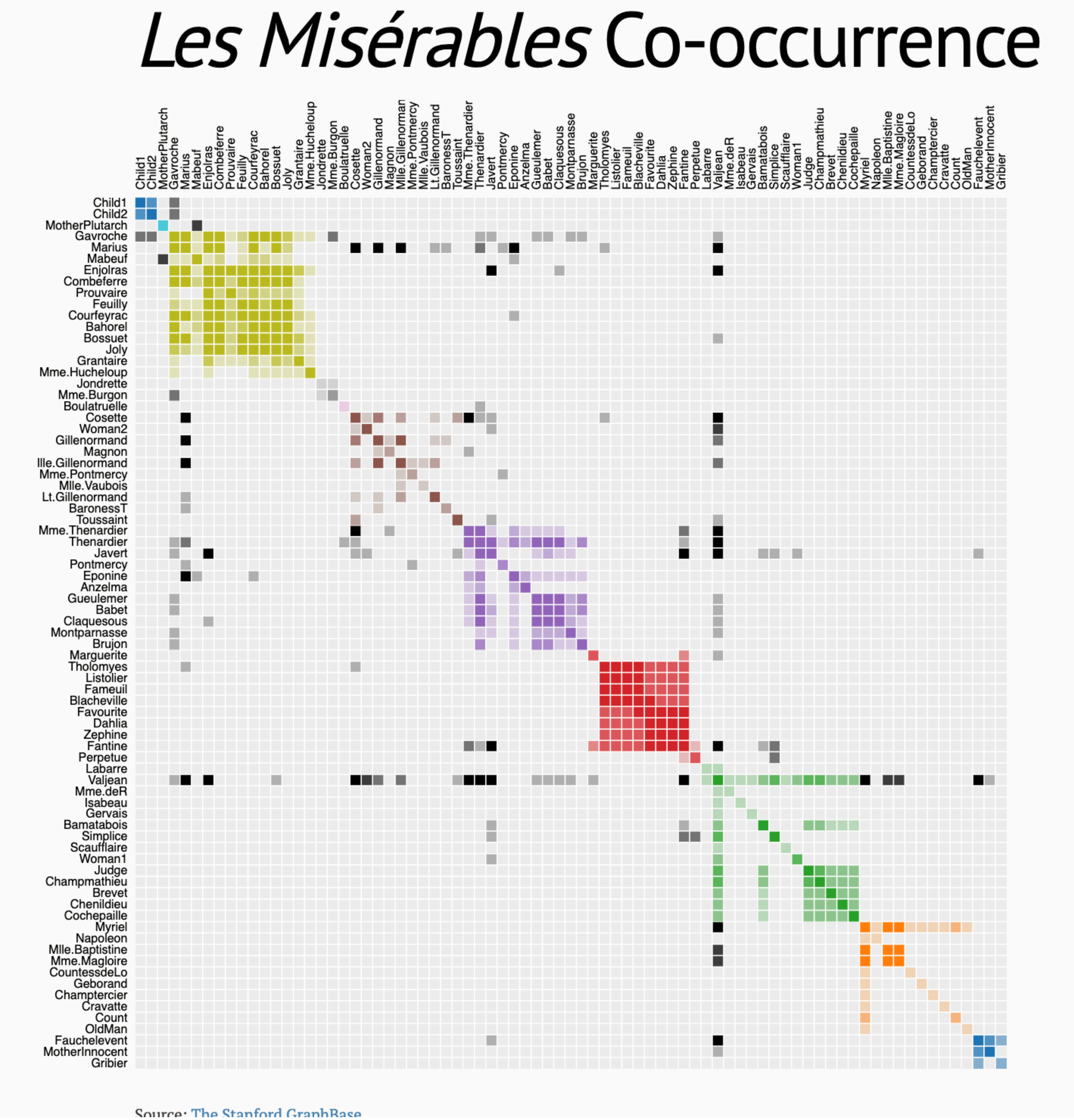


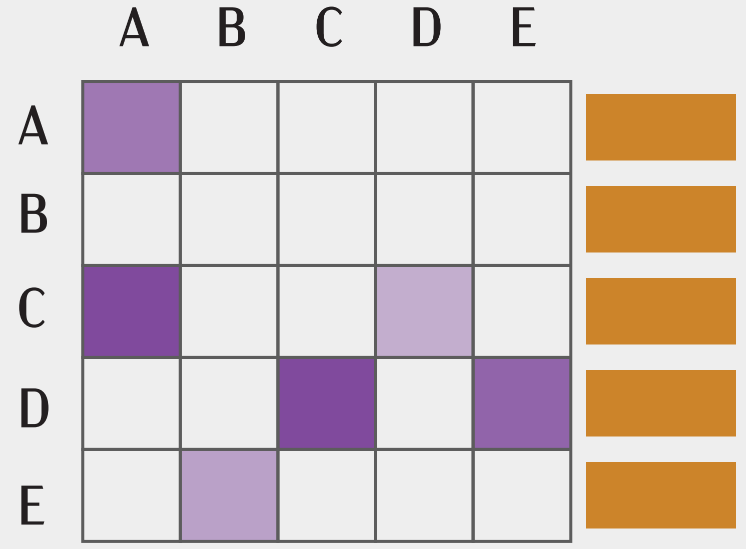
Brad
graphic designer

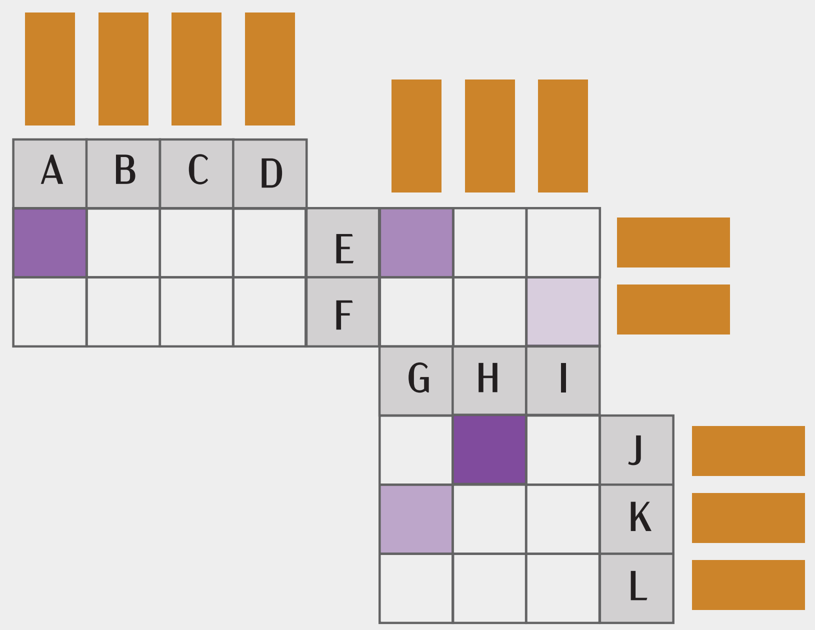


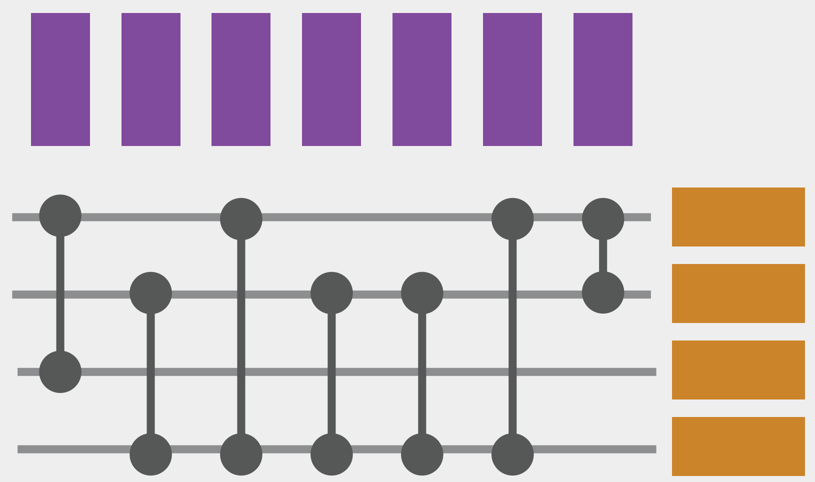
Maya
developer

developer

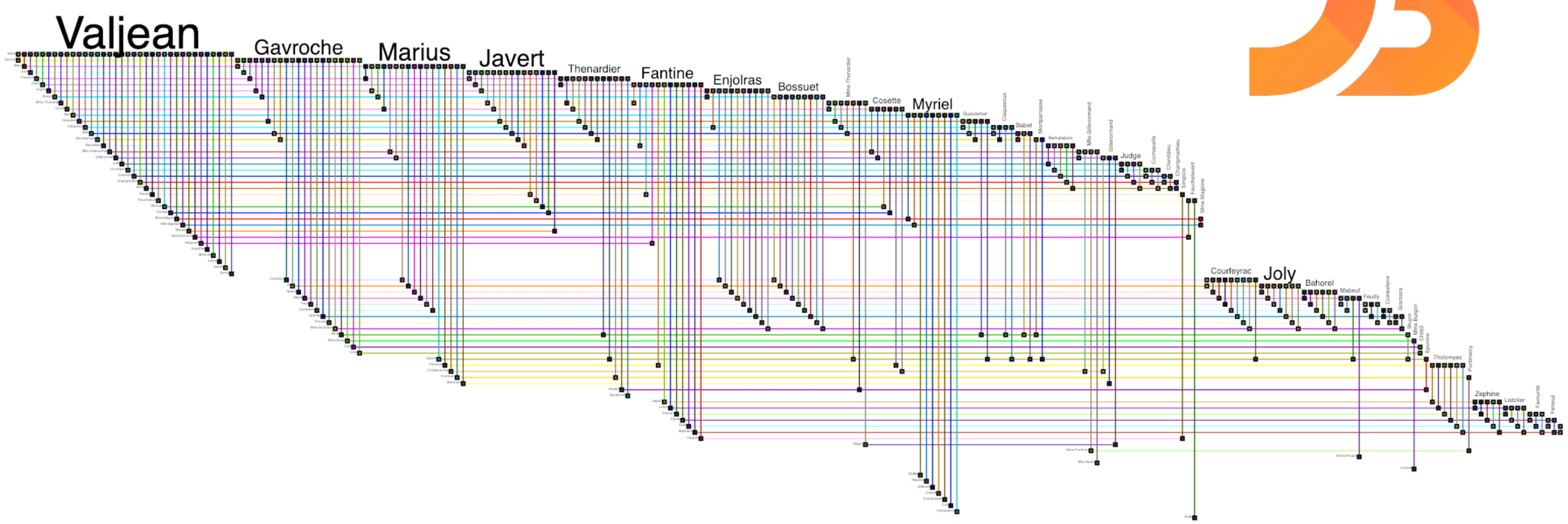

Les Misérables Co-occurrence
Source: The Stanford GraphBase


A
B
C
D
E

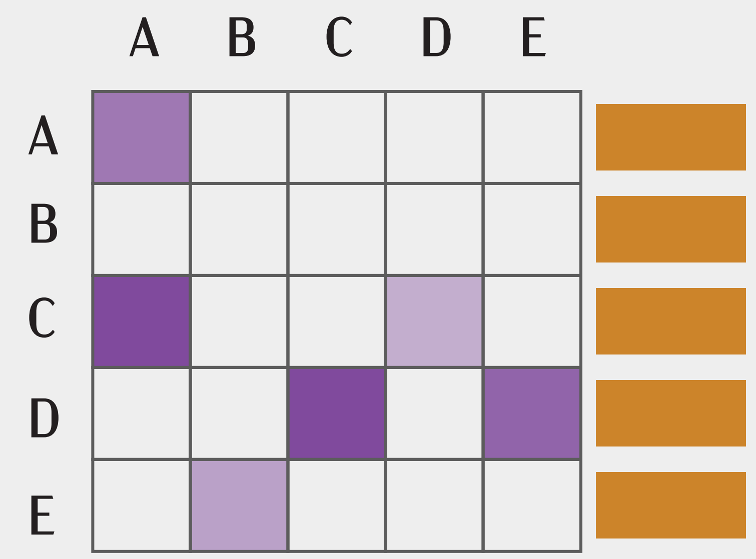

A
B
C
D
E
F
G
H
I
J
K
L

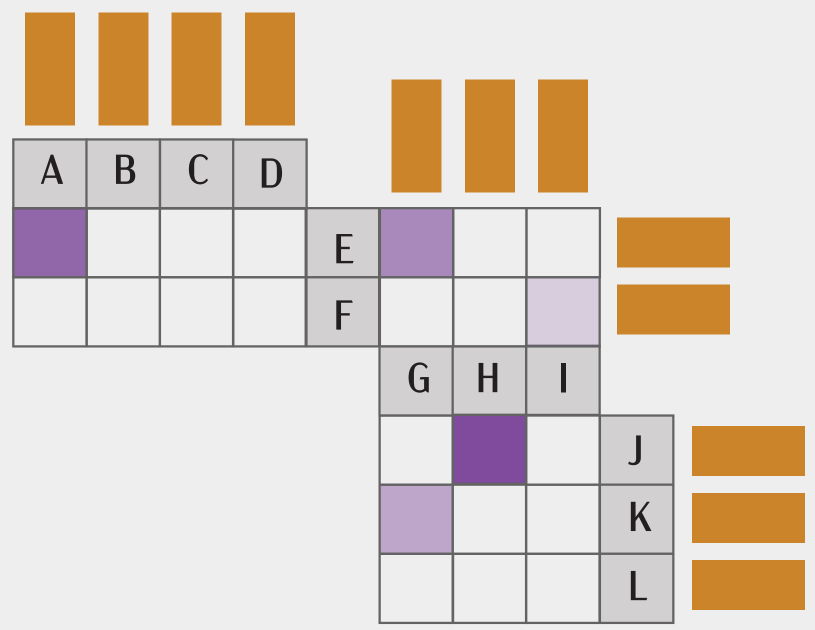


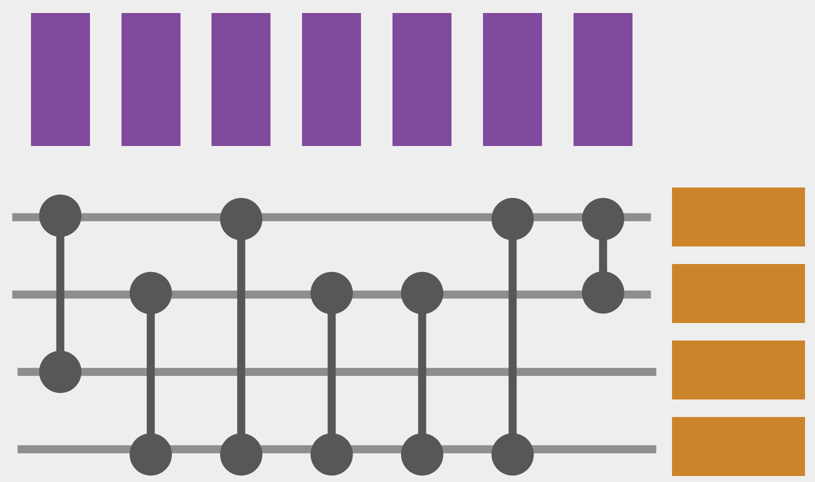
developer



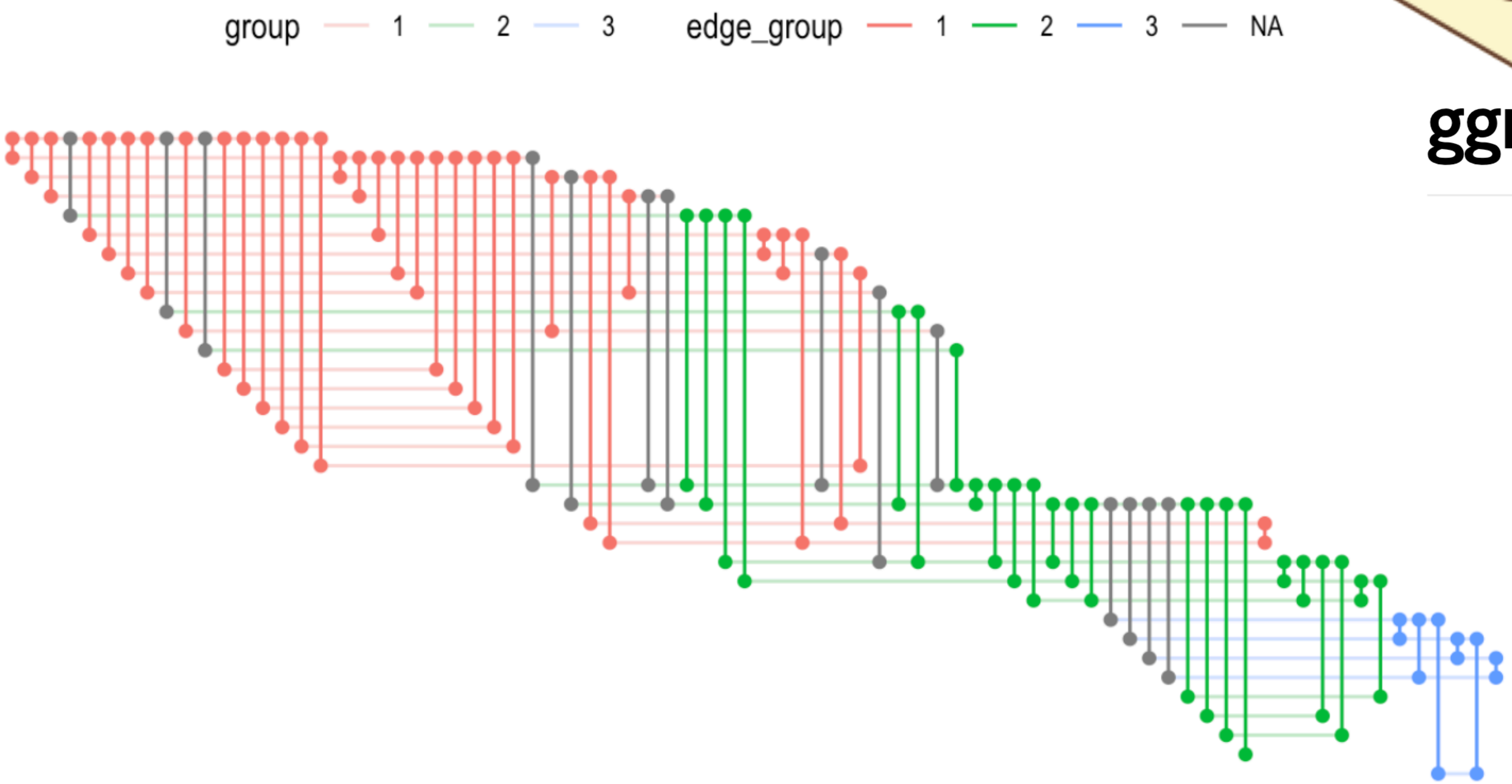
http://www.biofabric.org



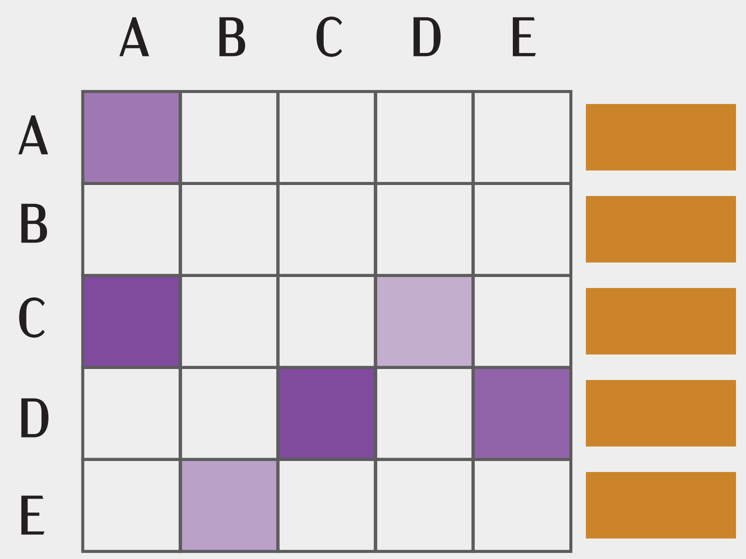


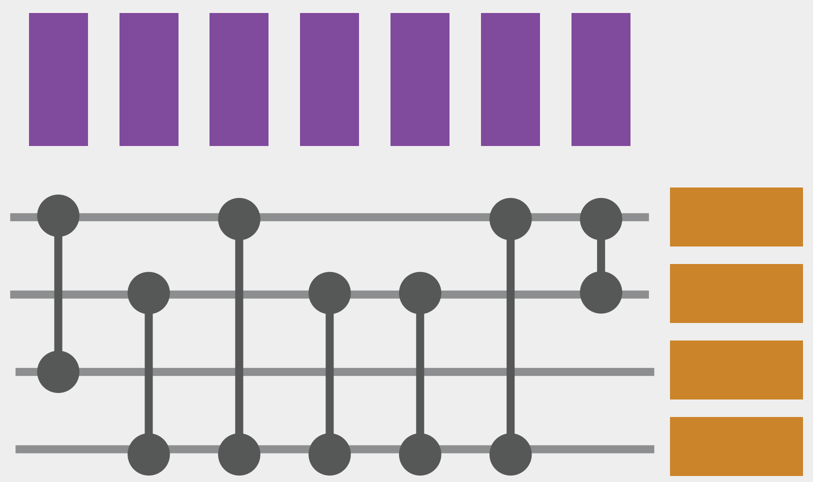


developer



ggraph

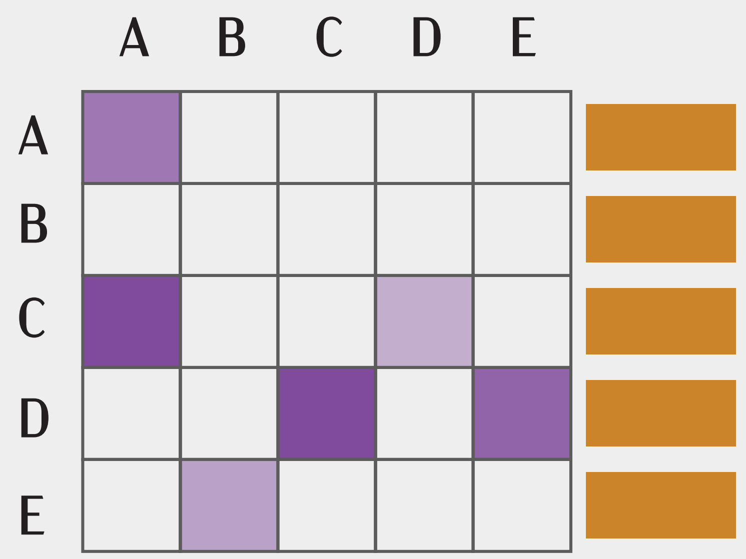


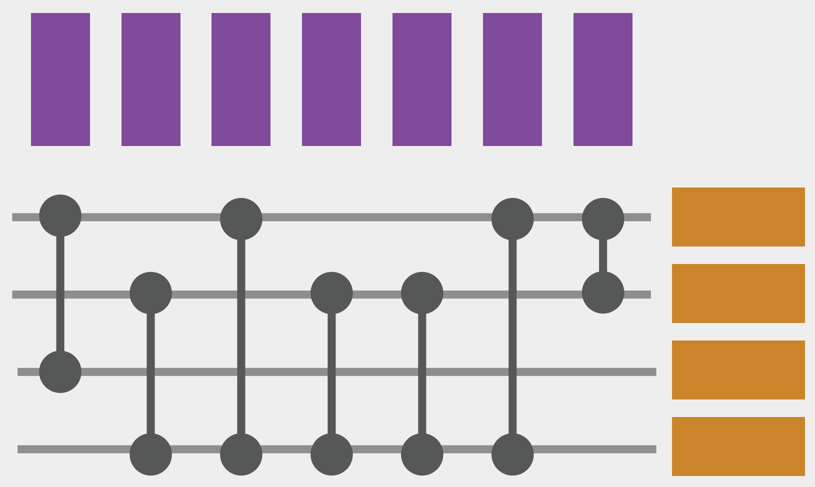


http://www.biofabric.org

develope

ggraph

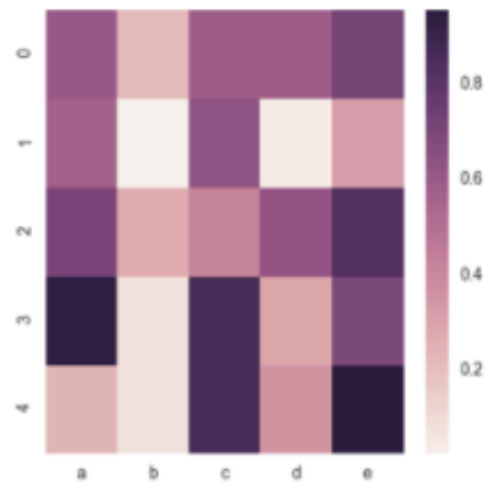




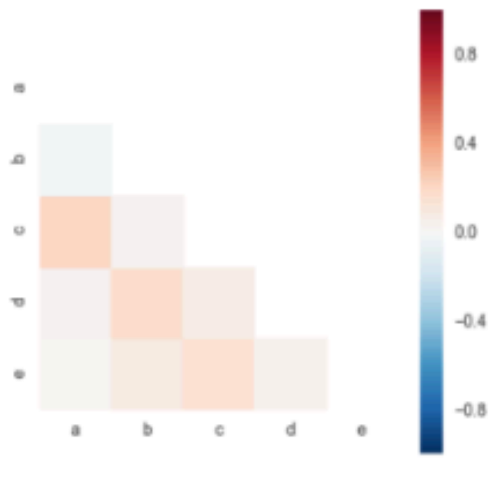
developer

python

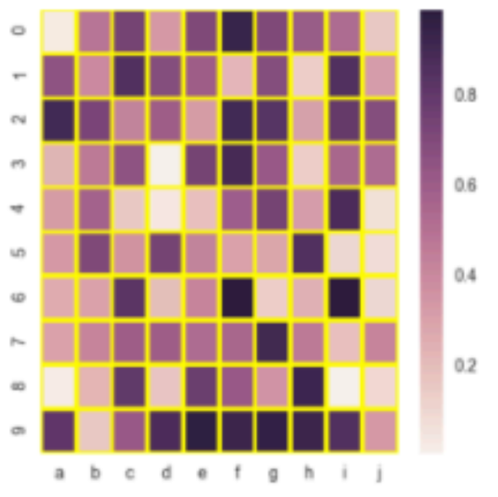
matplotlib



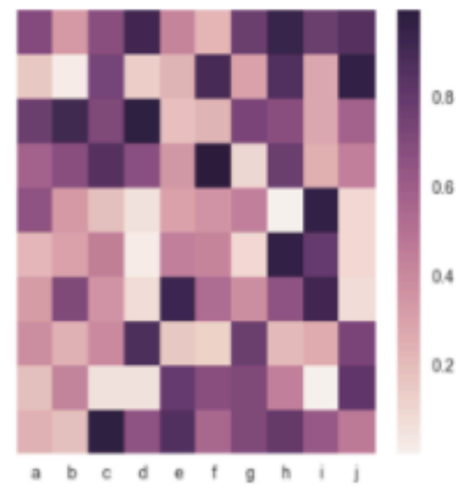
#90 Input format for heatmap



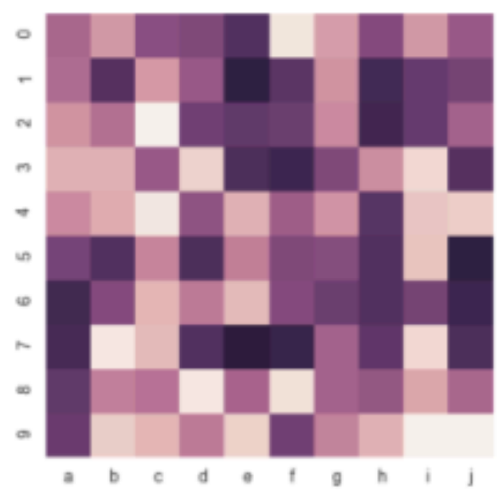
#90 Half heatmap



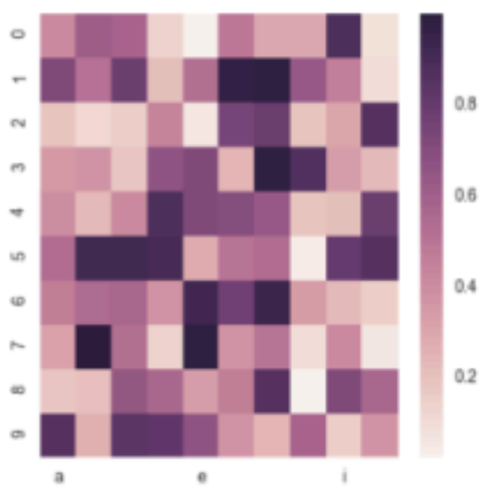
#91 Custom seaborn heatmap



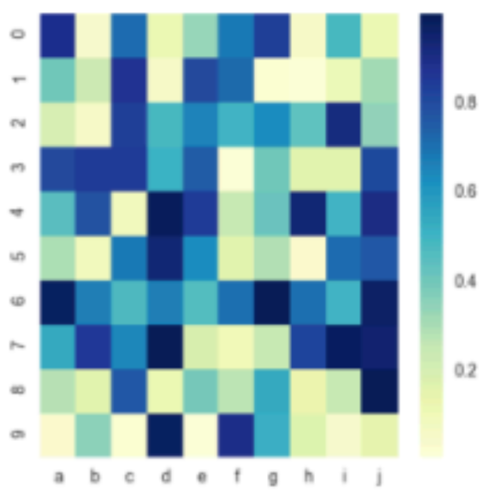
#91 Custom seaborn heatmap



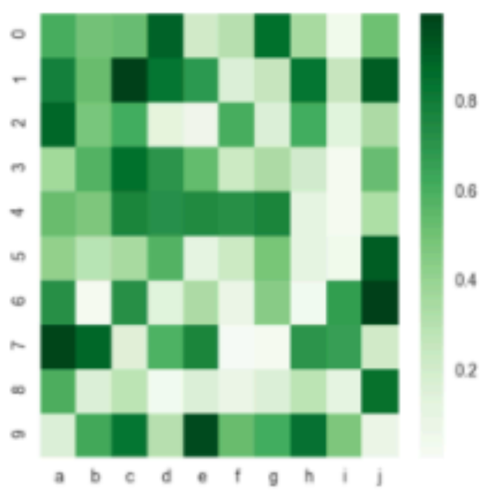
#91 Custom seaborn heatmap



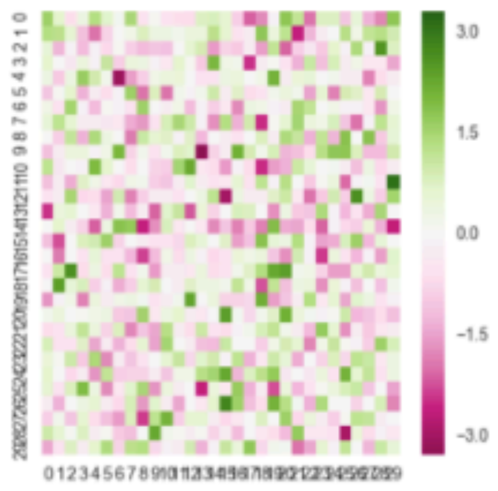
#91 Custom seaborn heatmap



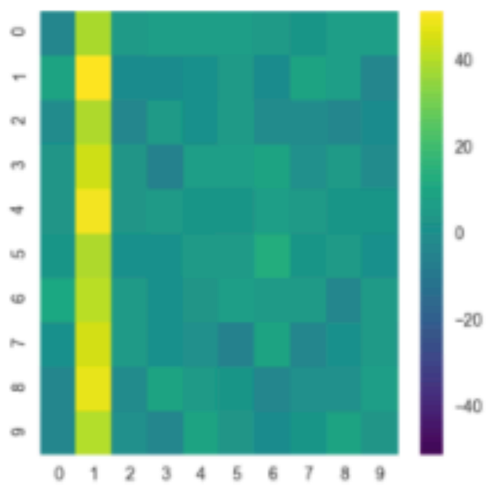
#92 Control heatmap color



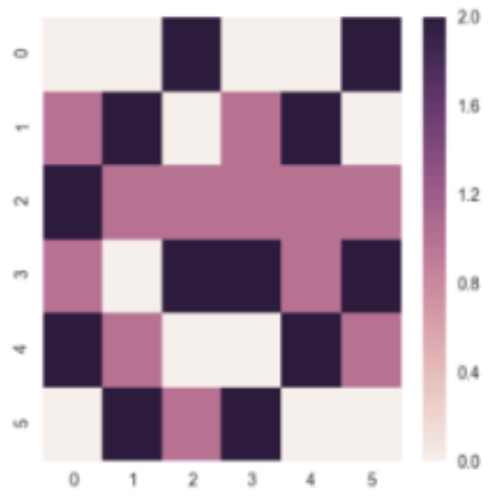
#92 Control heatmap color



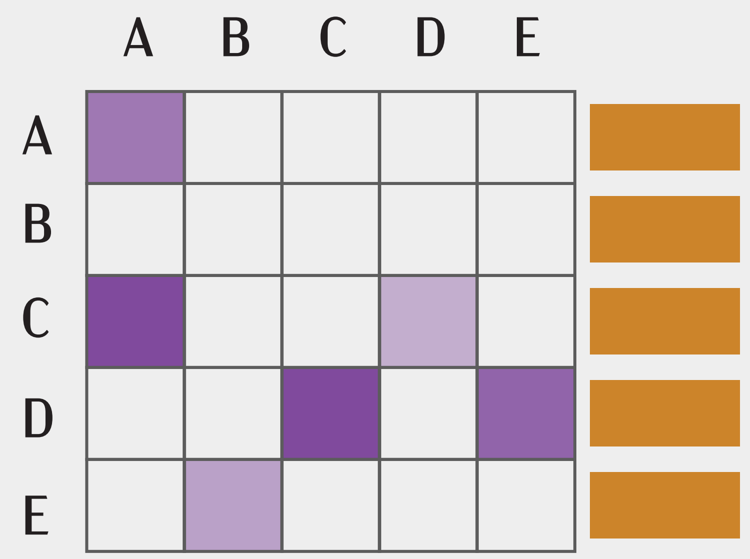
#92 Control heatmap color

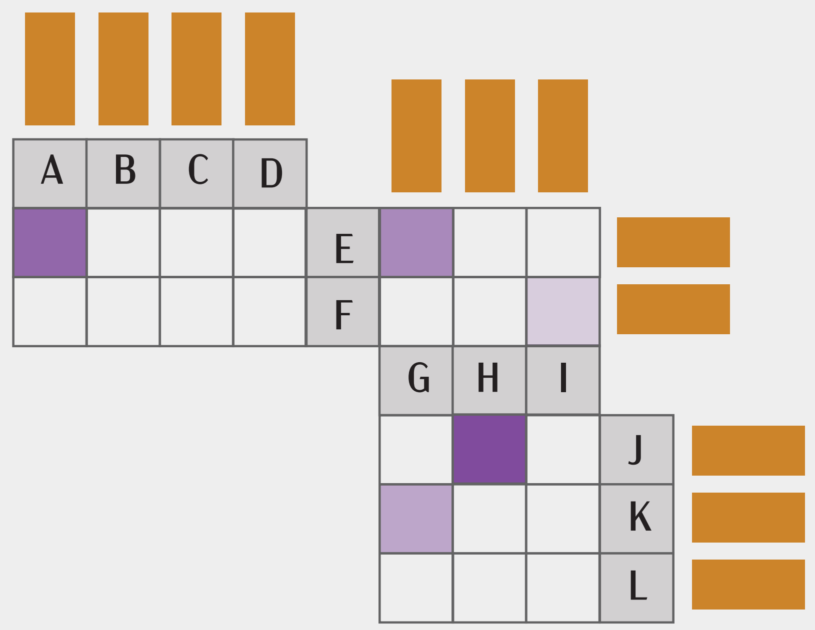


#94 Column normalization on heatmap



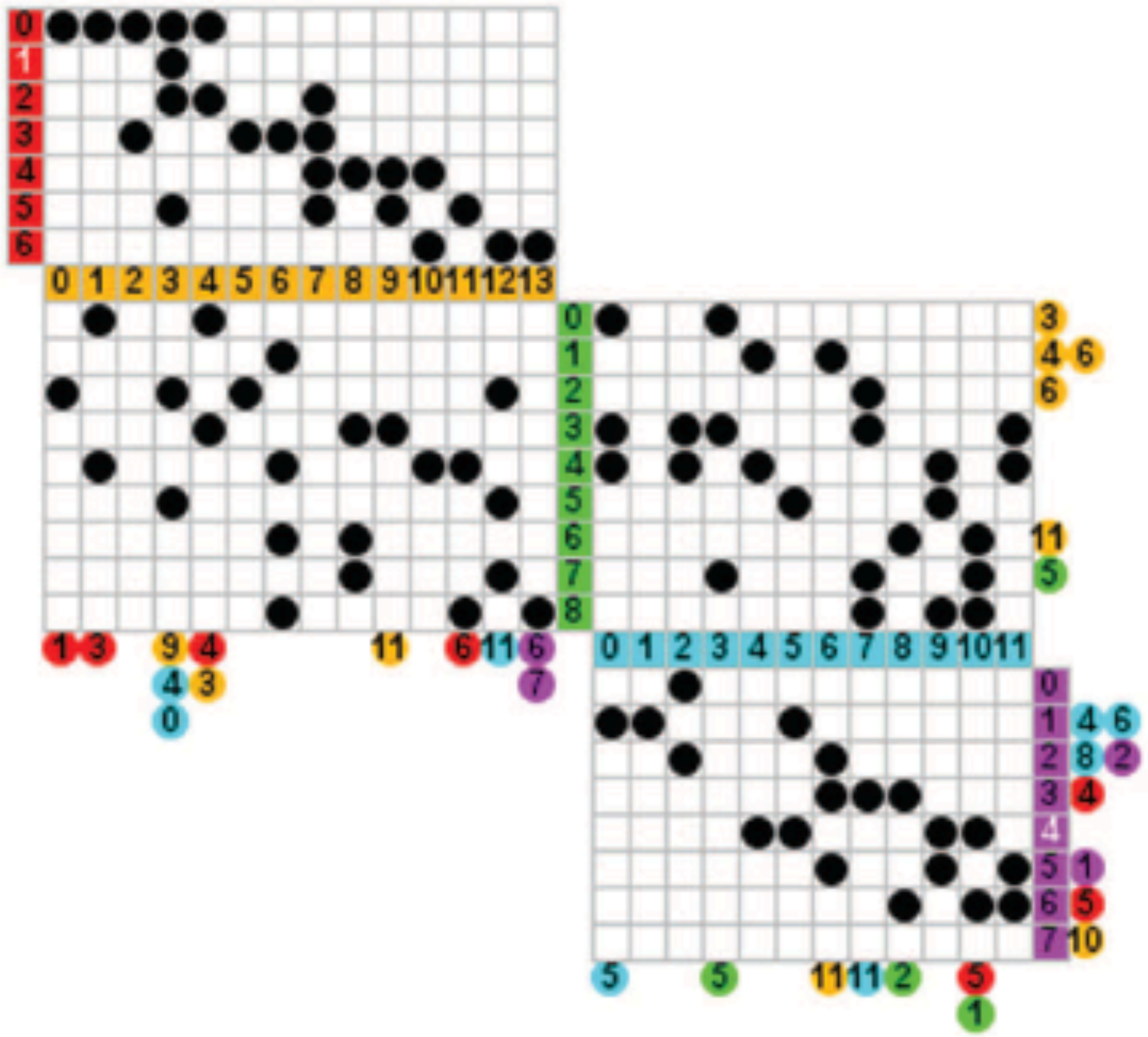
#92 Turn your data categorical for heatmap

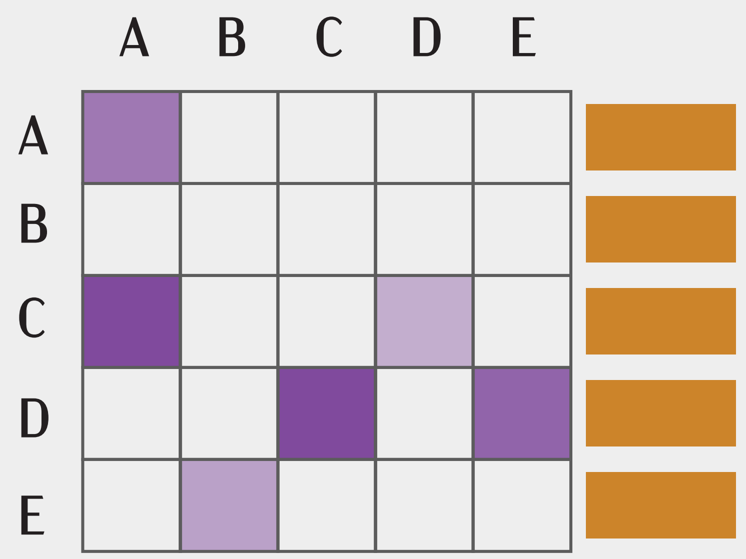


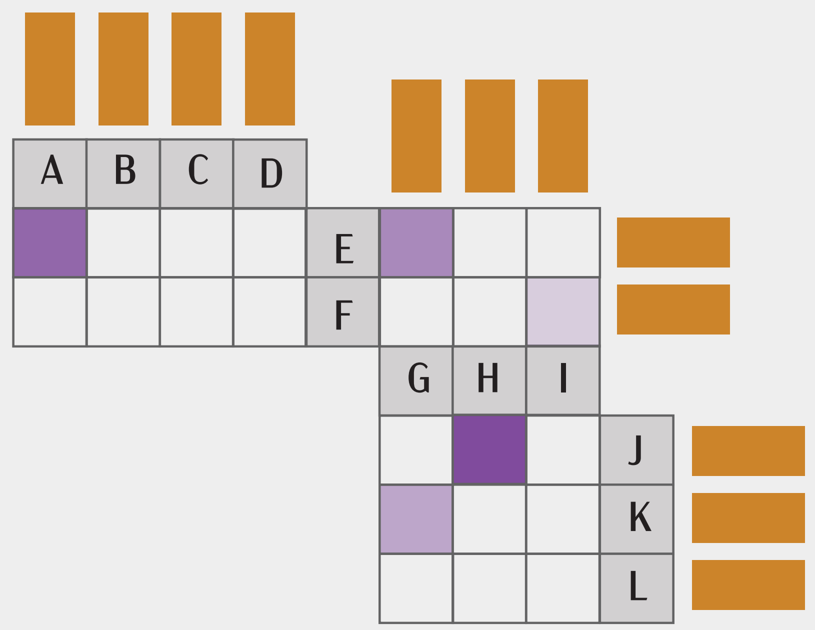


developer

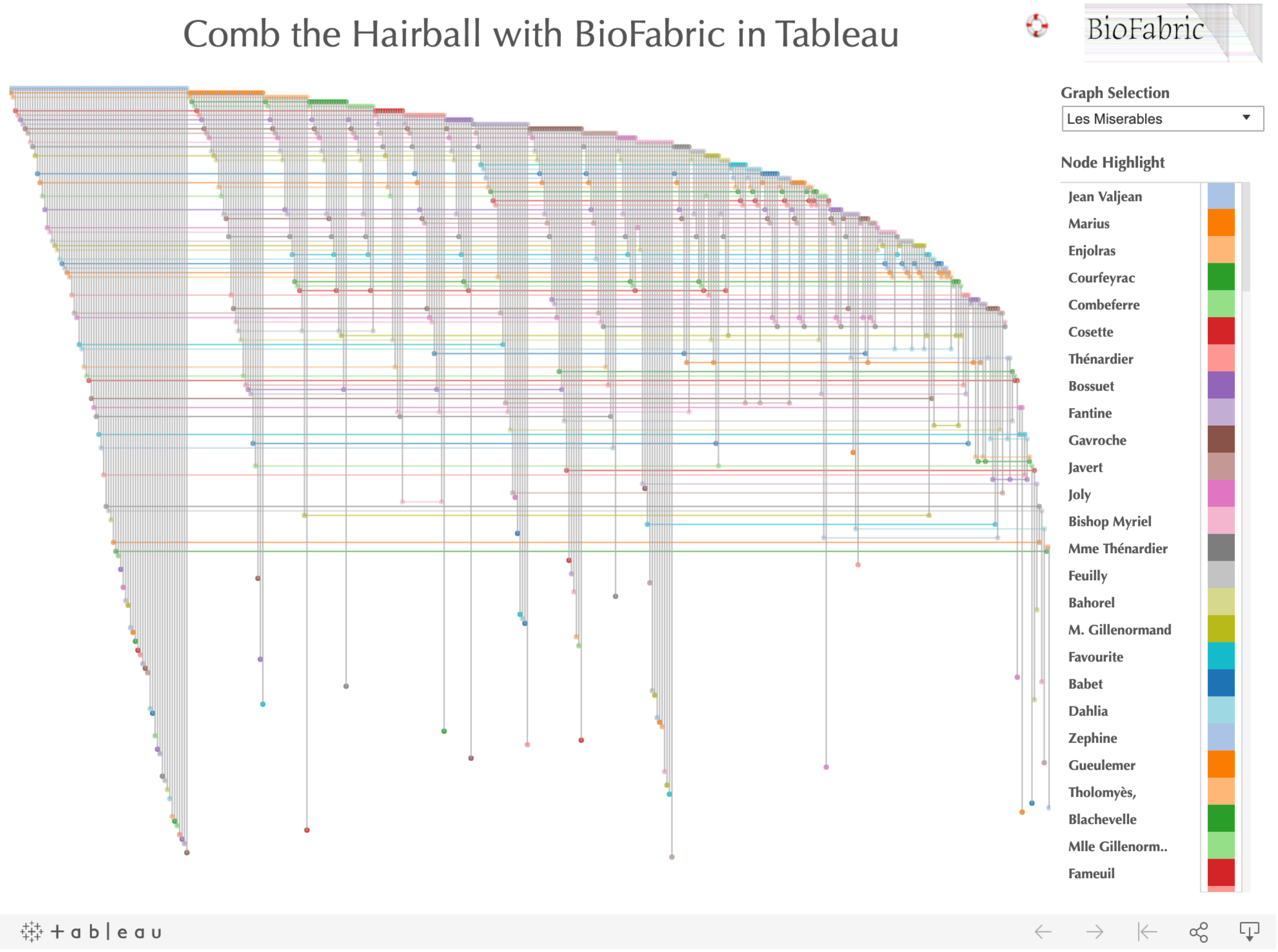
python

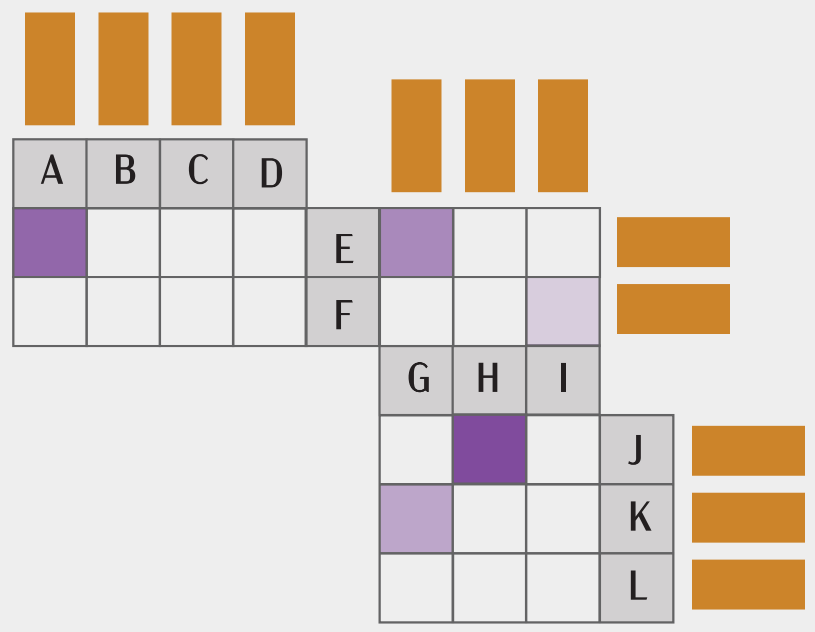


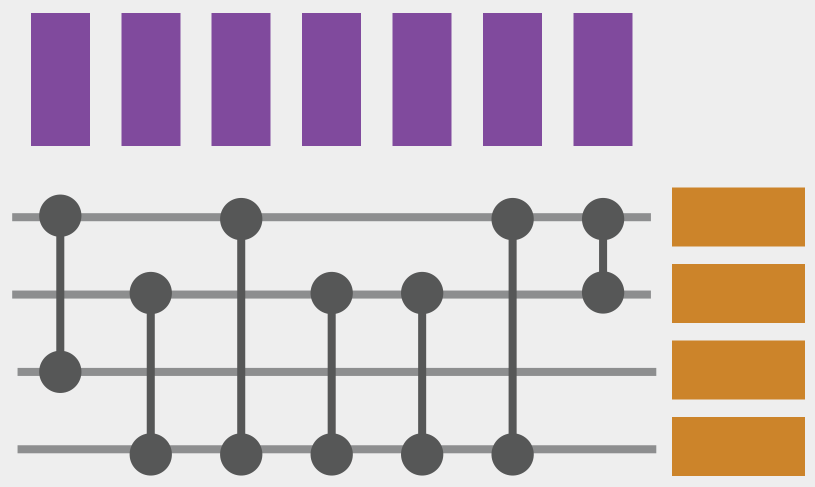




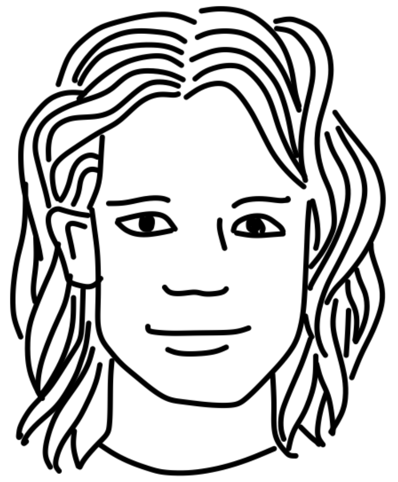
graphic
designer



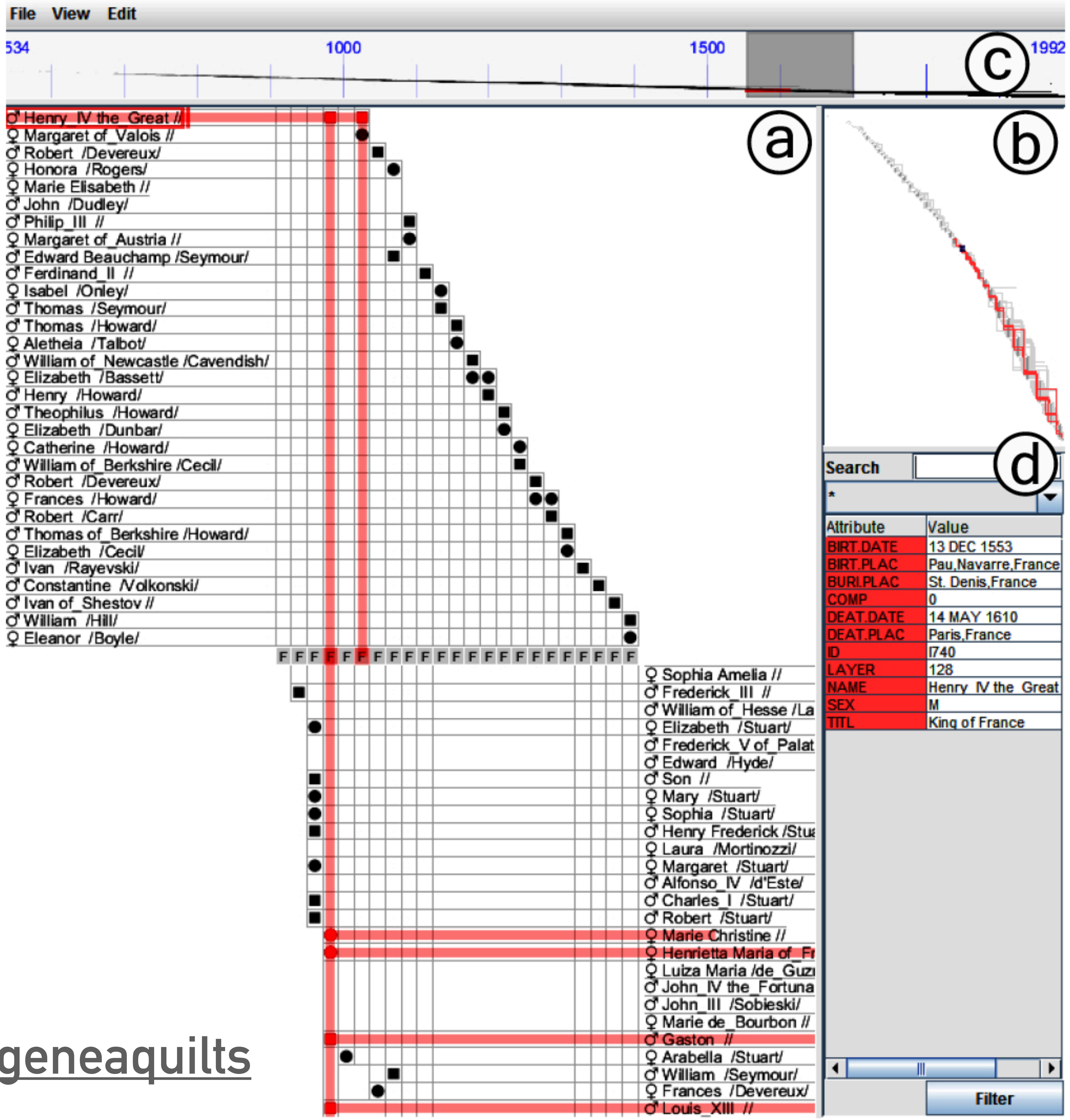


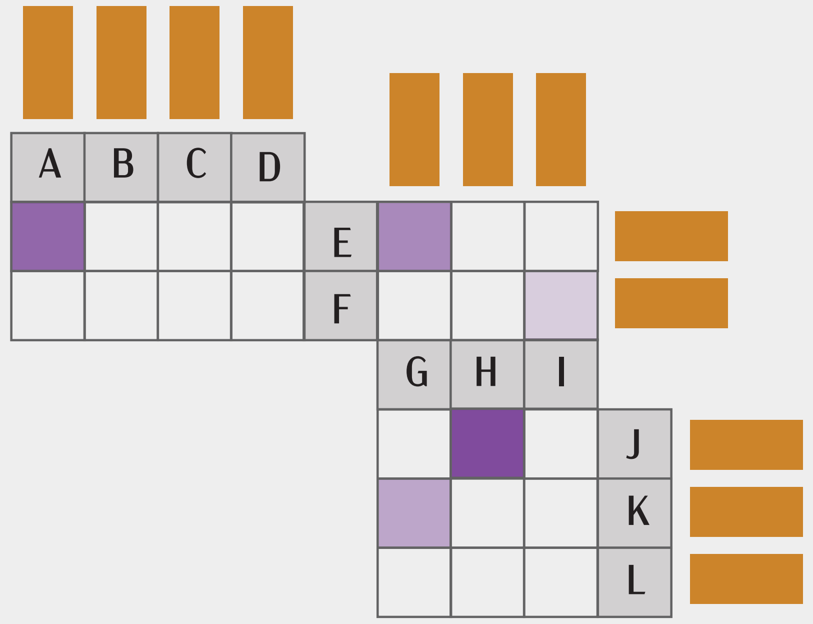


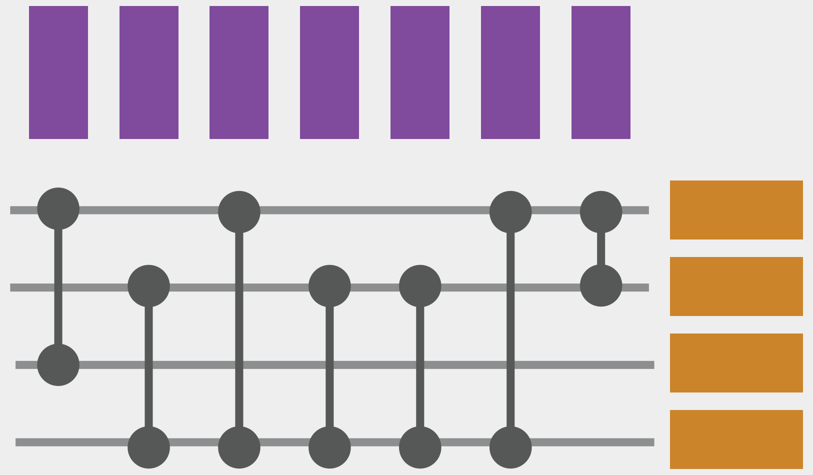
Genea Quilts



graphic designer

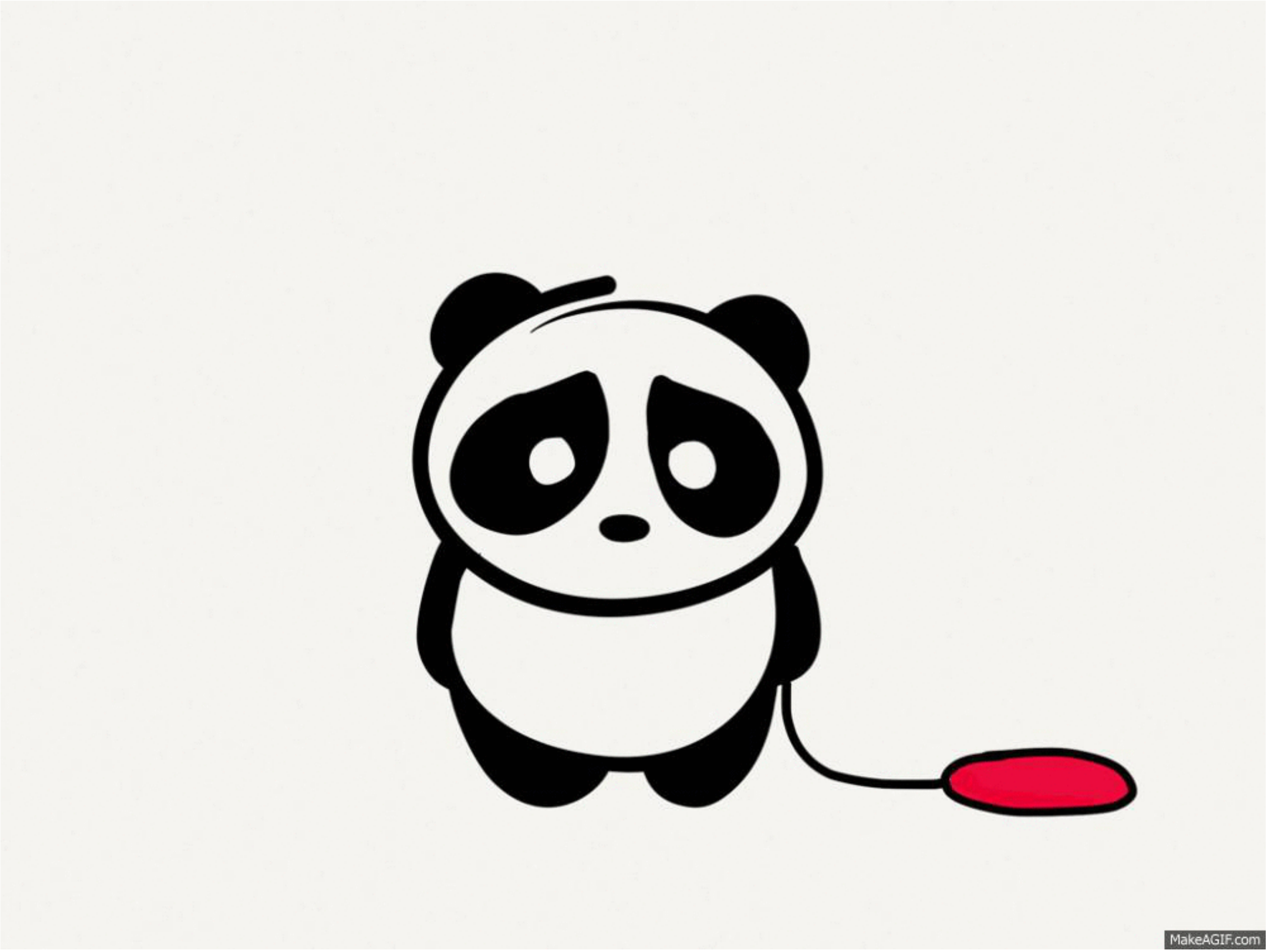


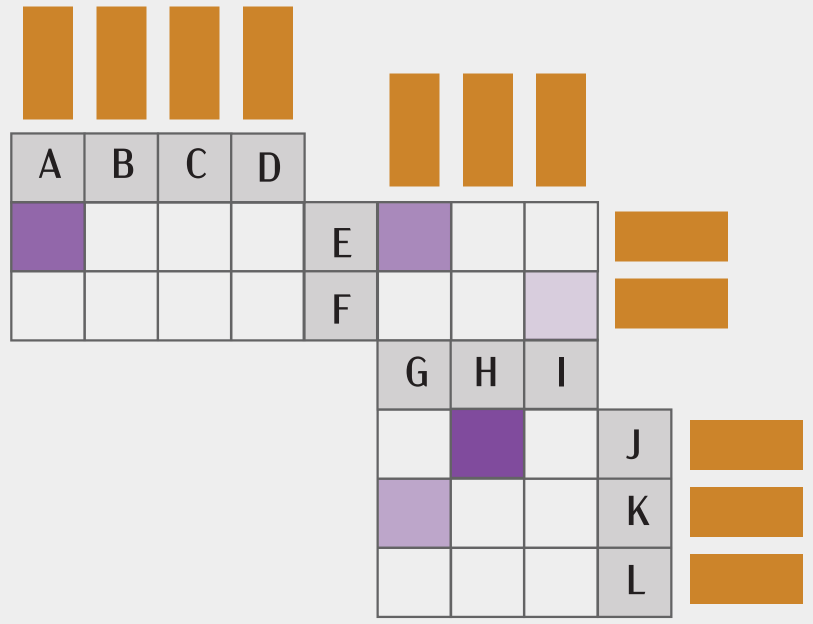


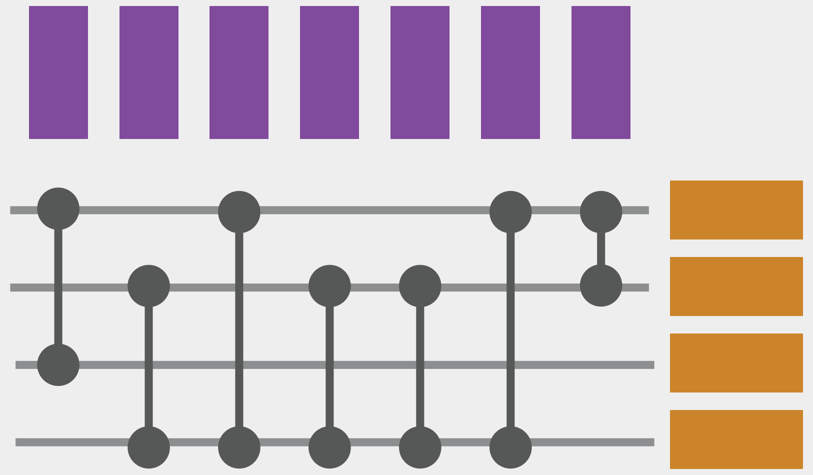


https://aviz.fr/geneaquilts

graphic
designer



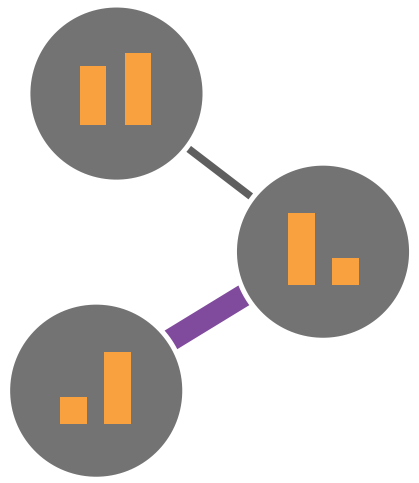




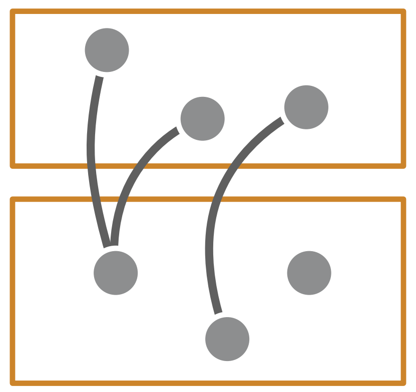
Activity

get your own twitter network @
bit.ly/twitter-network

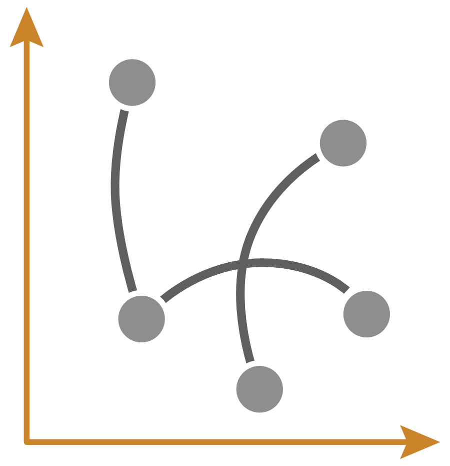
Choose a representation



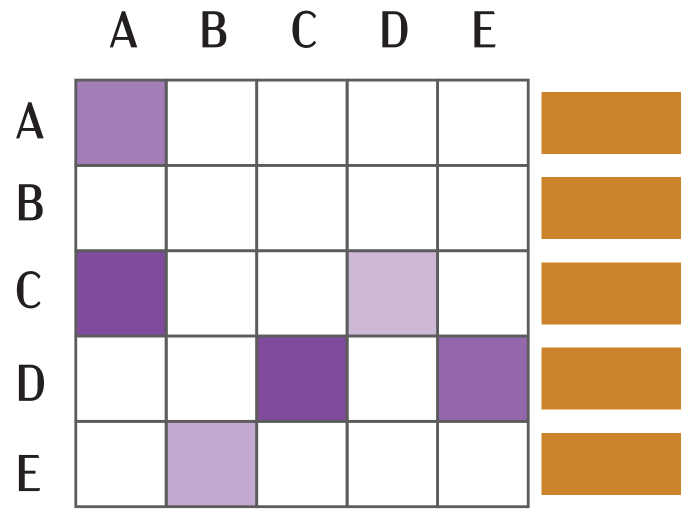
On-Node / On-Edge Encoding



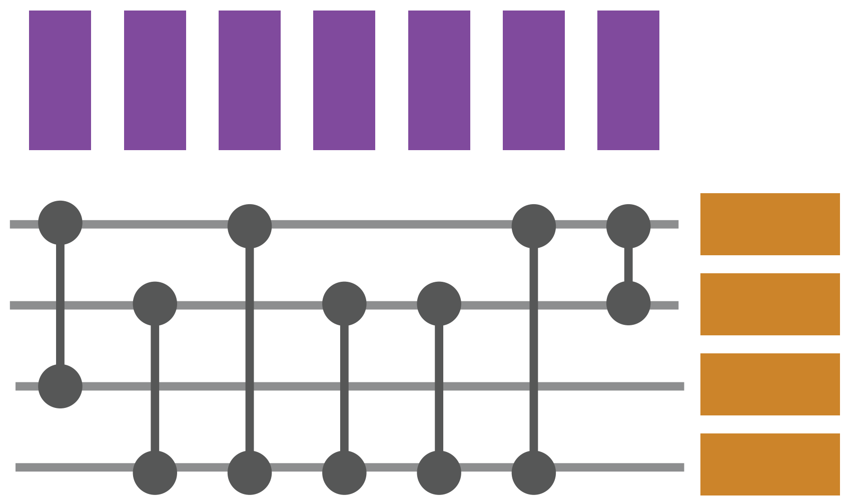
Attribute-Driven Faceting



Attribute-Driven Positioning



Adjacency Matrix



BioFabric

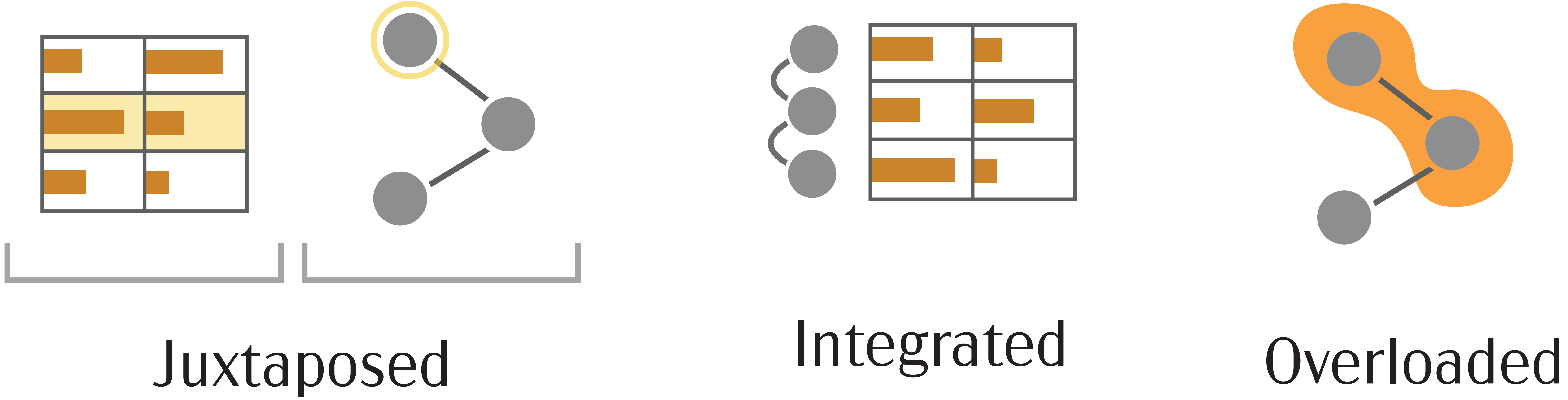
15 minutes

Exchange visualizations with your neighbor and explain your encodings.

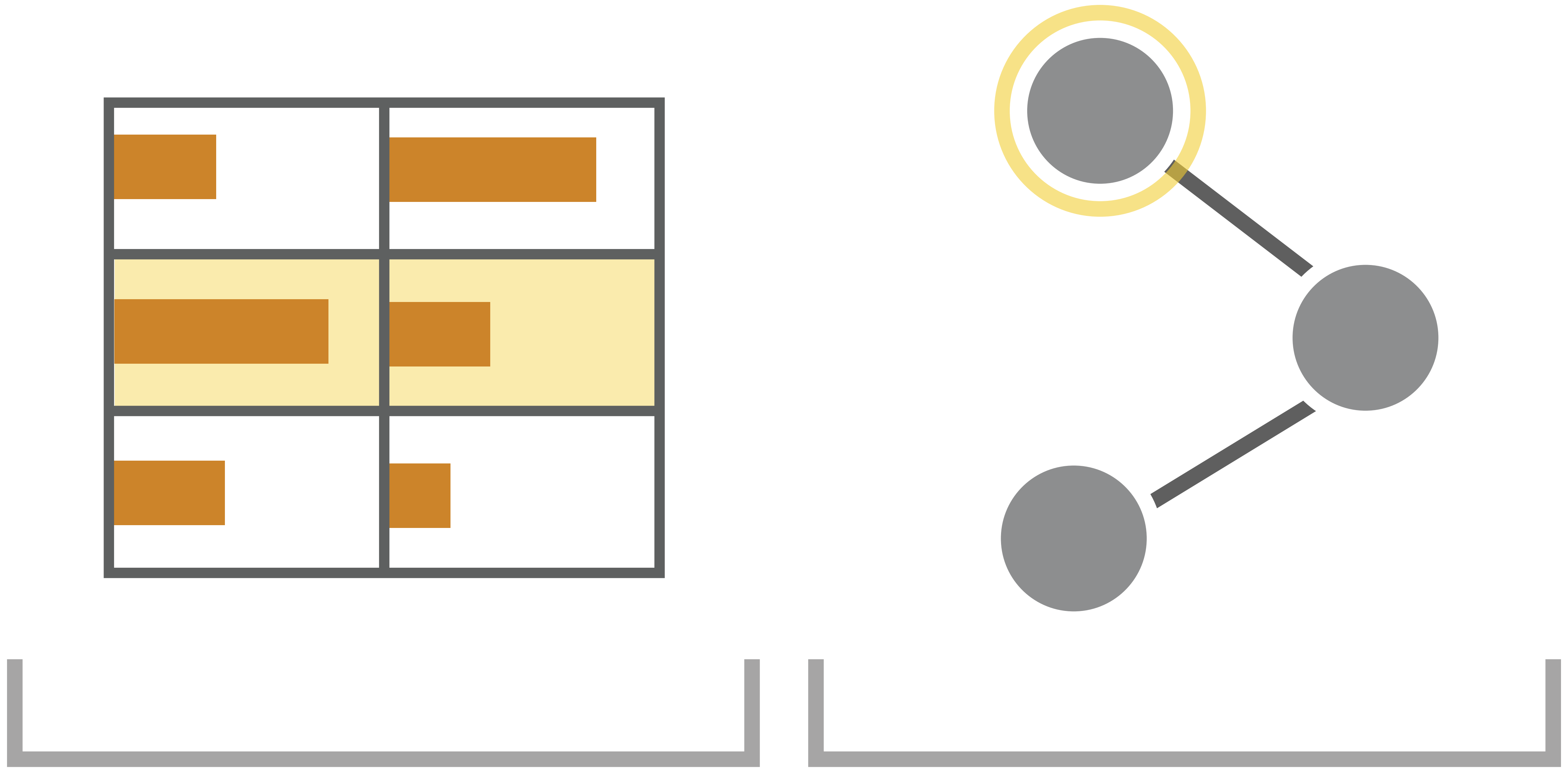
How many tweets does the person who has the most connections in this graph have?

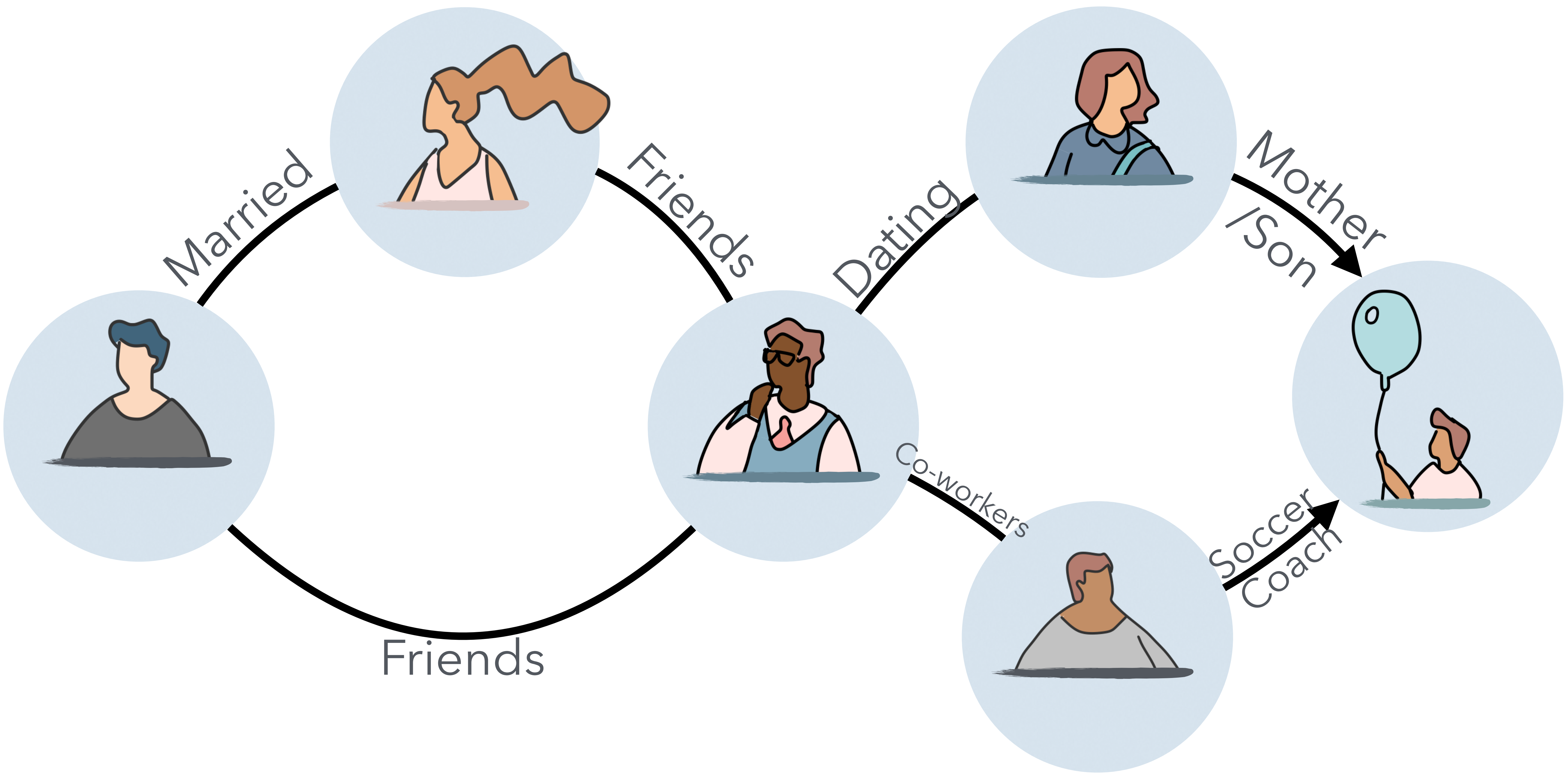
Does the person with the least tweets have more interactions of type retweet or mention?

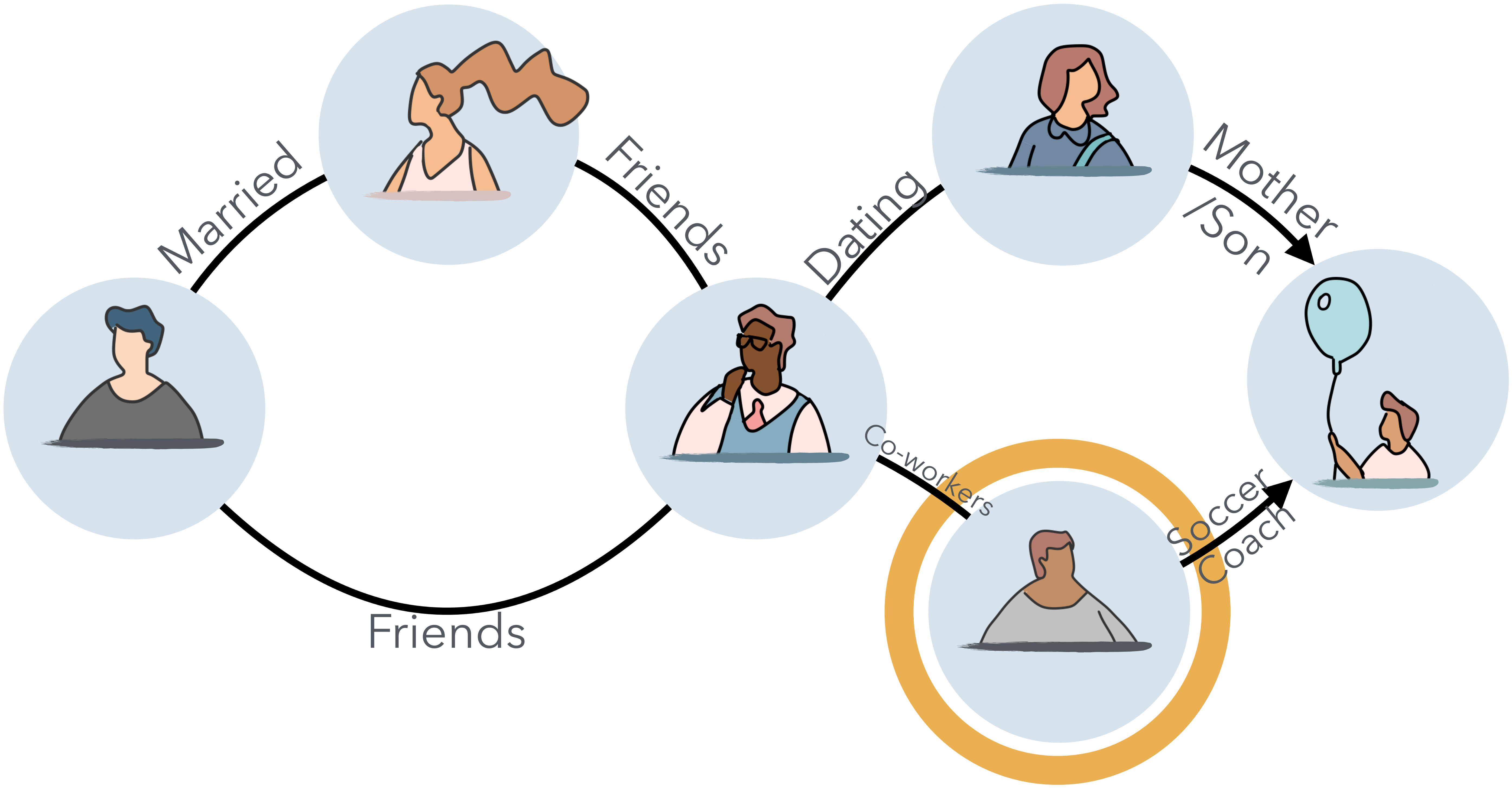
View Operations



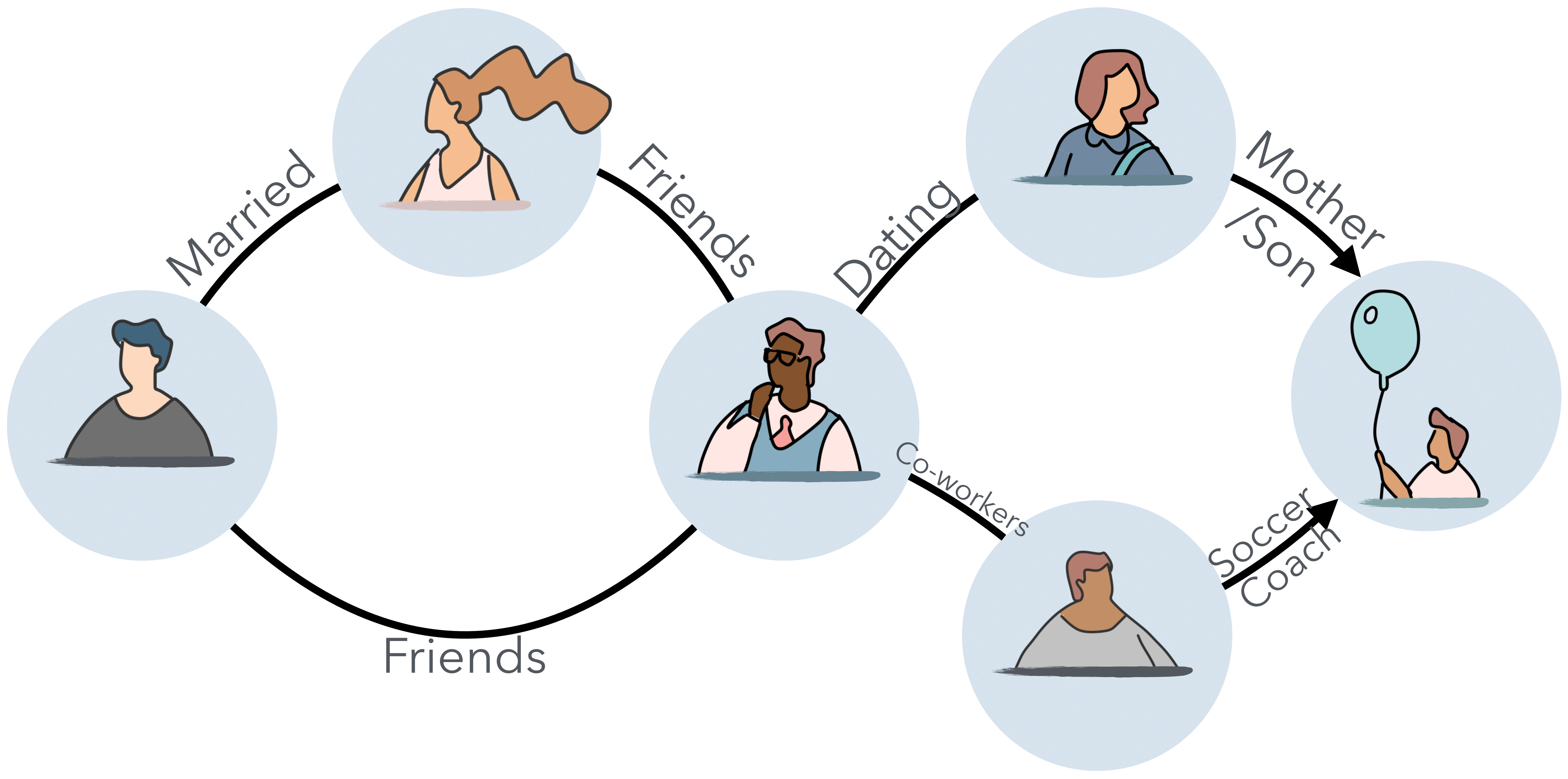
Juxtaposed




Married
Friends
Friends
Dating
Mother
/Son
Co-workers
Soccer
Coach

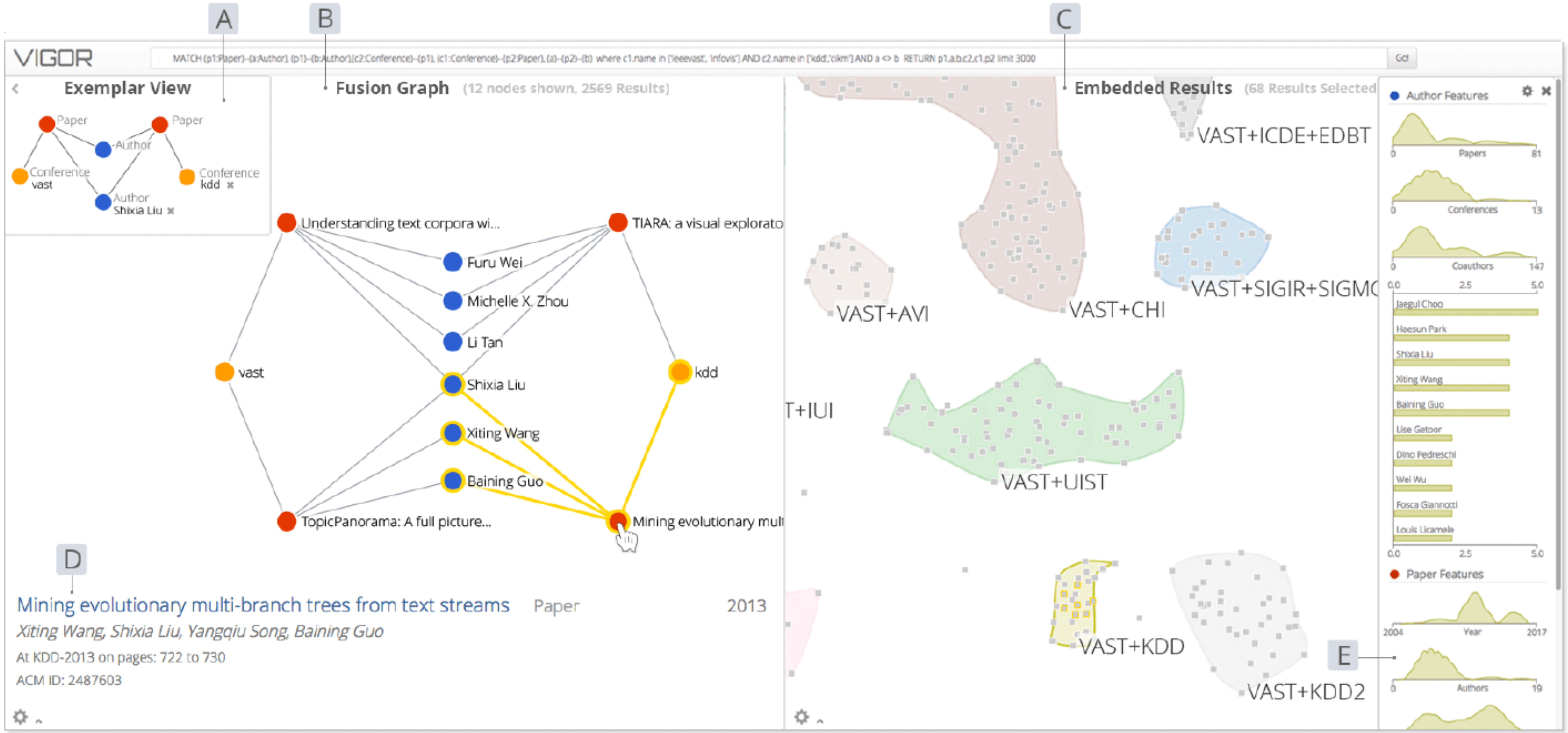


Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3

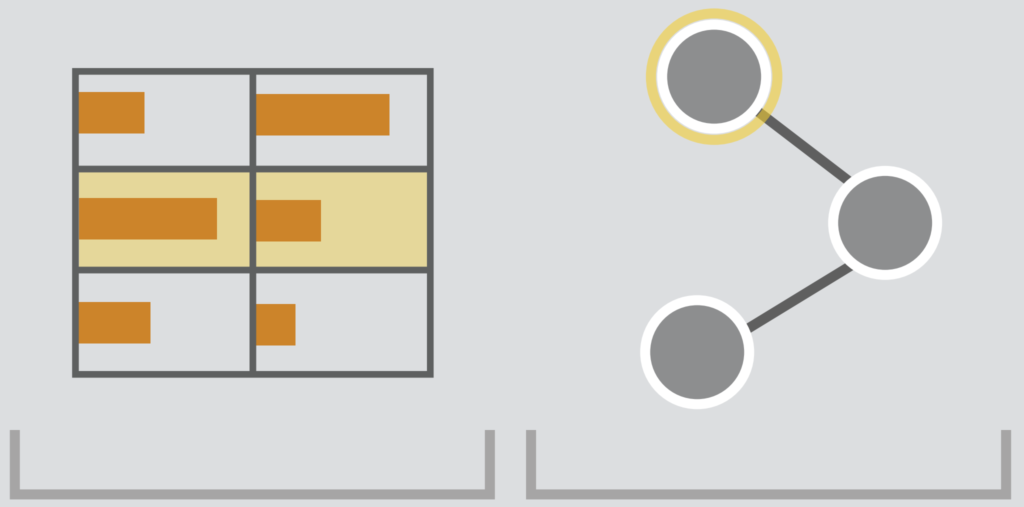


Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3

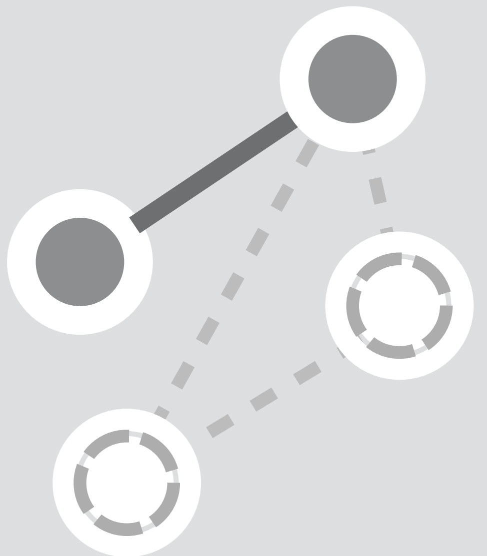
Relationship	Years
Dating	4
Mother / Son	12
Co-workers	3
Soccer Coach	2
Friends	8
Friends	3
Married	4



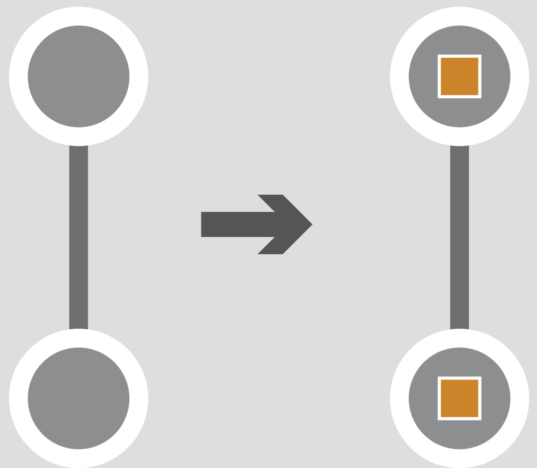
VIGOR *Pienta et al. 2018*



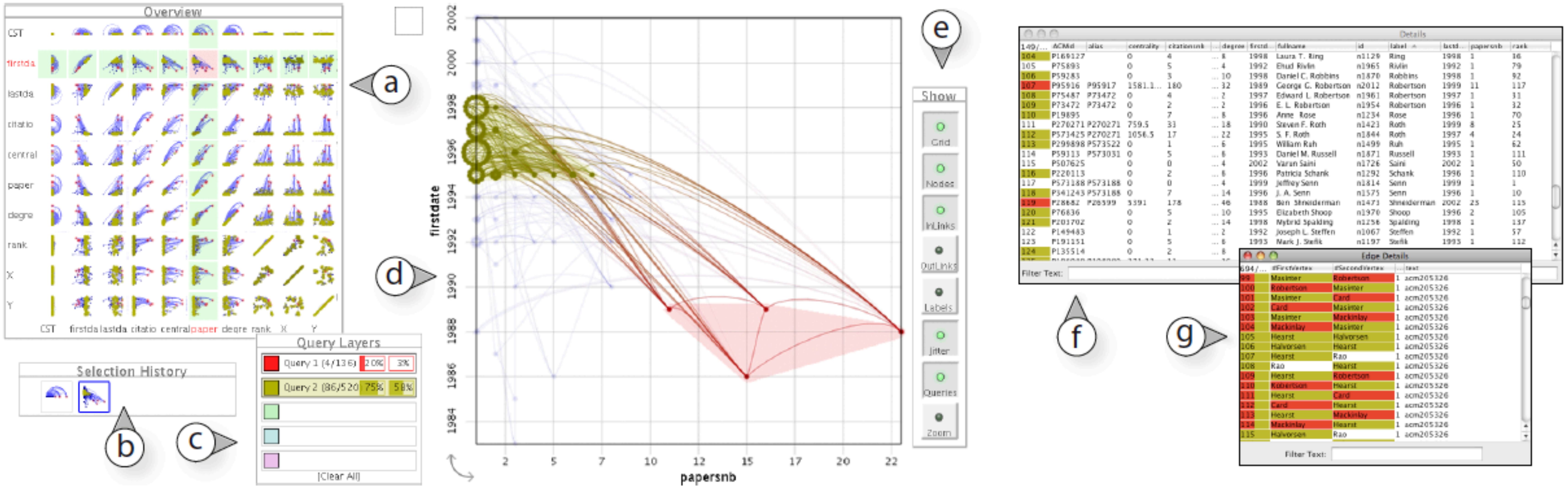
Juxtaposed

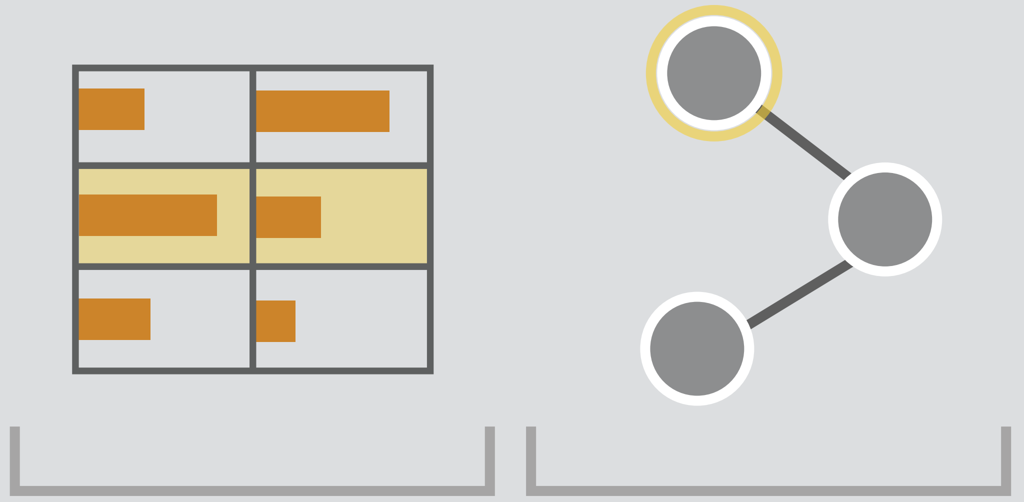


Querying and Filtering



Deriving New Attributes

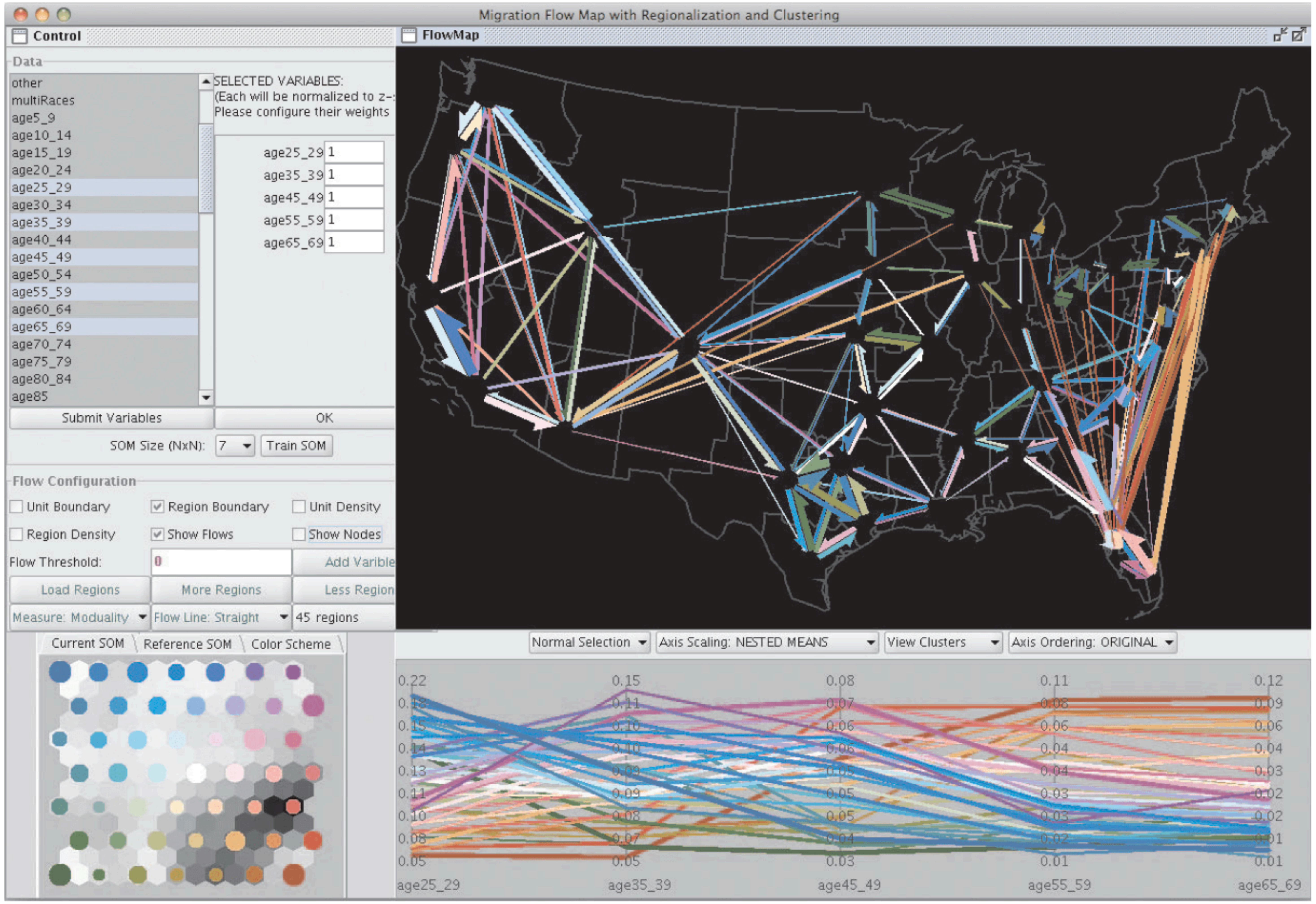


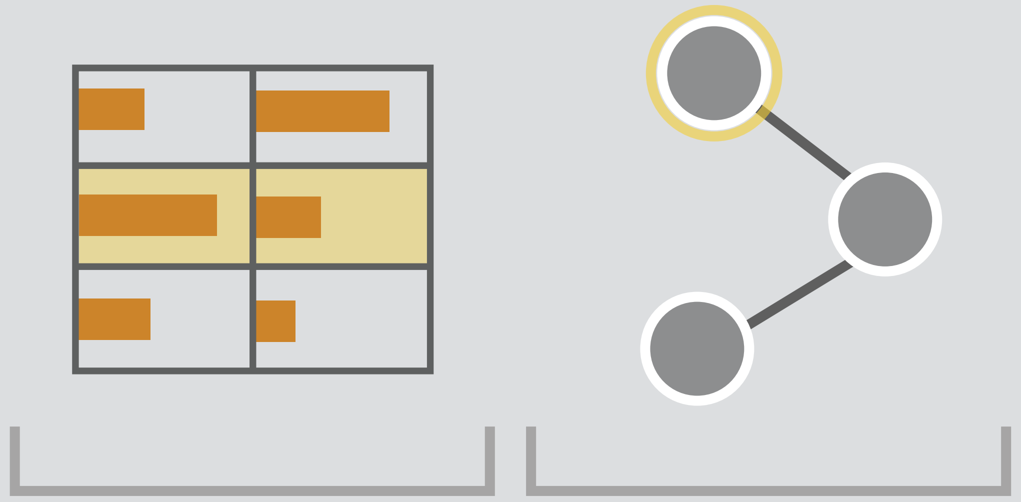


Juxtaposed

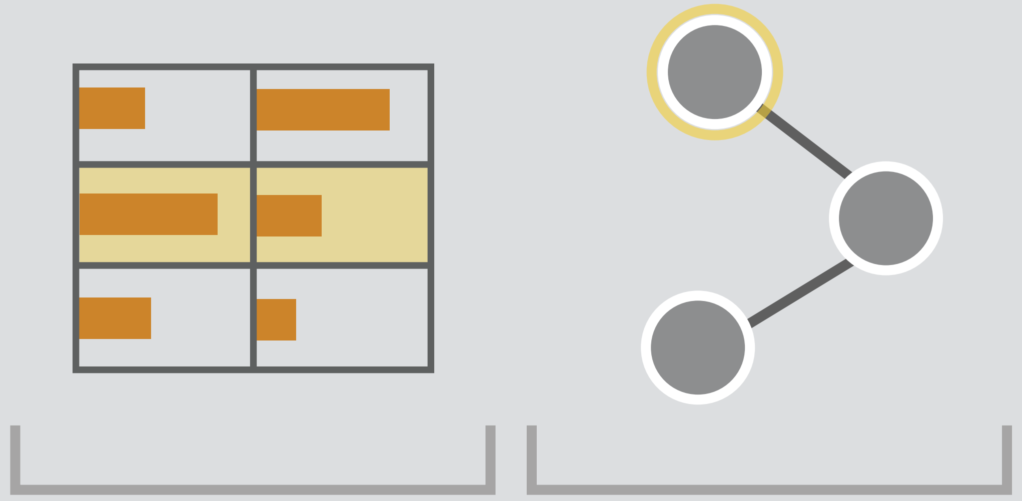
Graph Dice *Bezerianos et al. 2010*

Guo, 2009

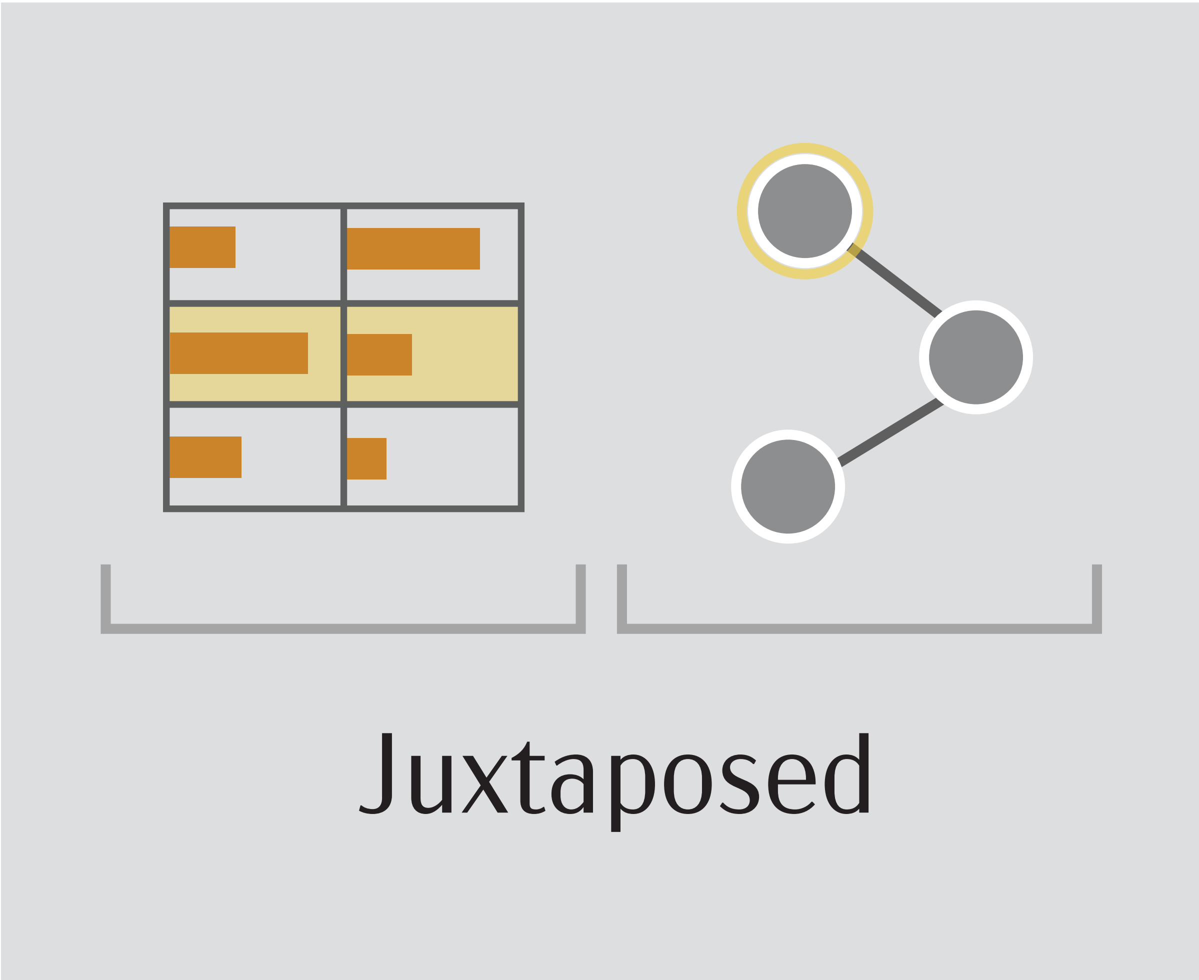




Juxtaposed



Juxtaposed

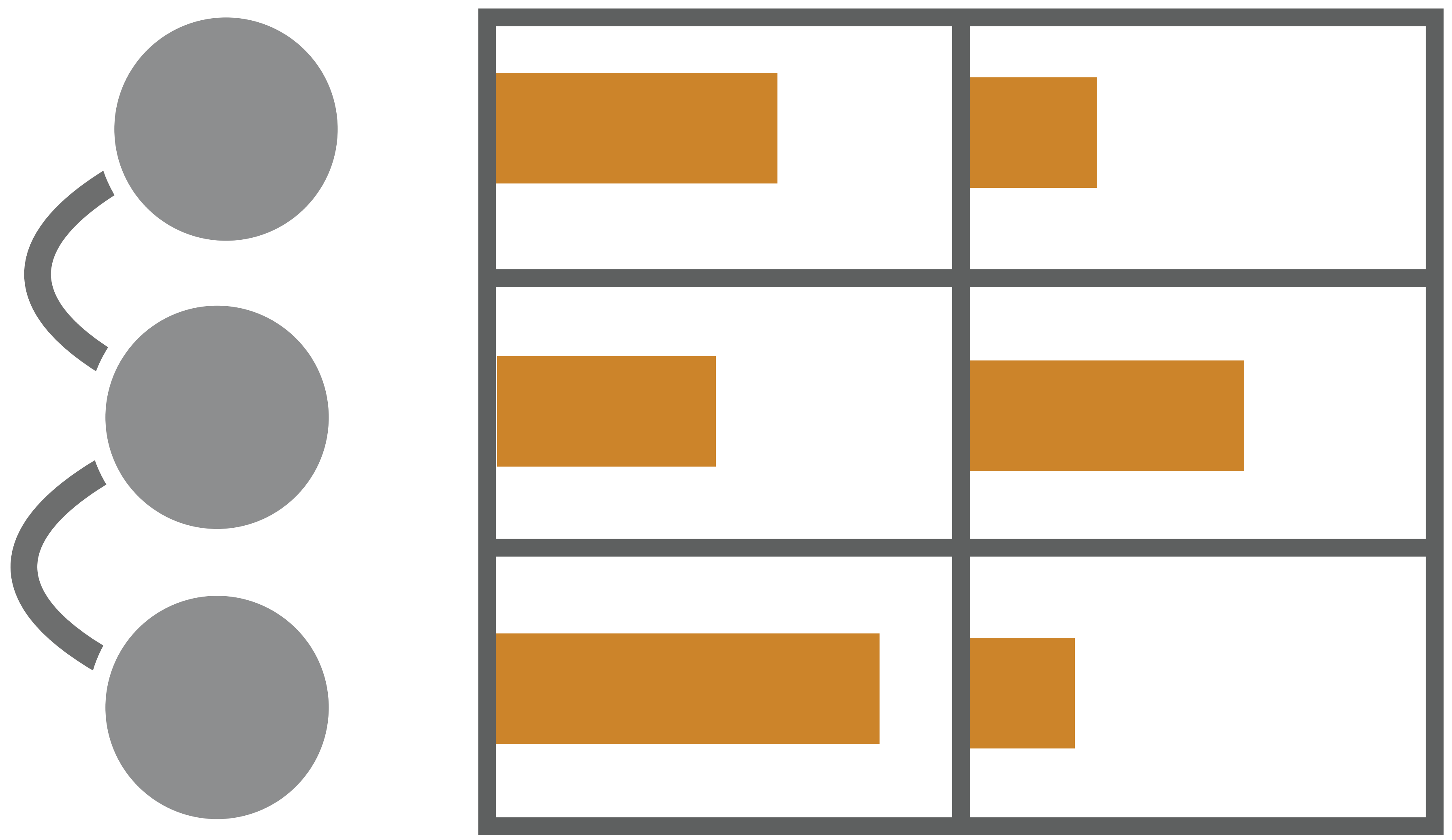


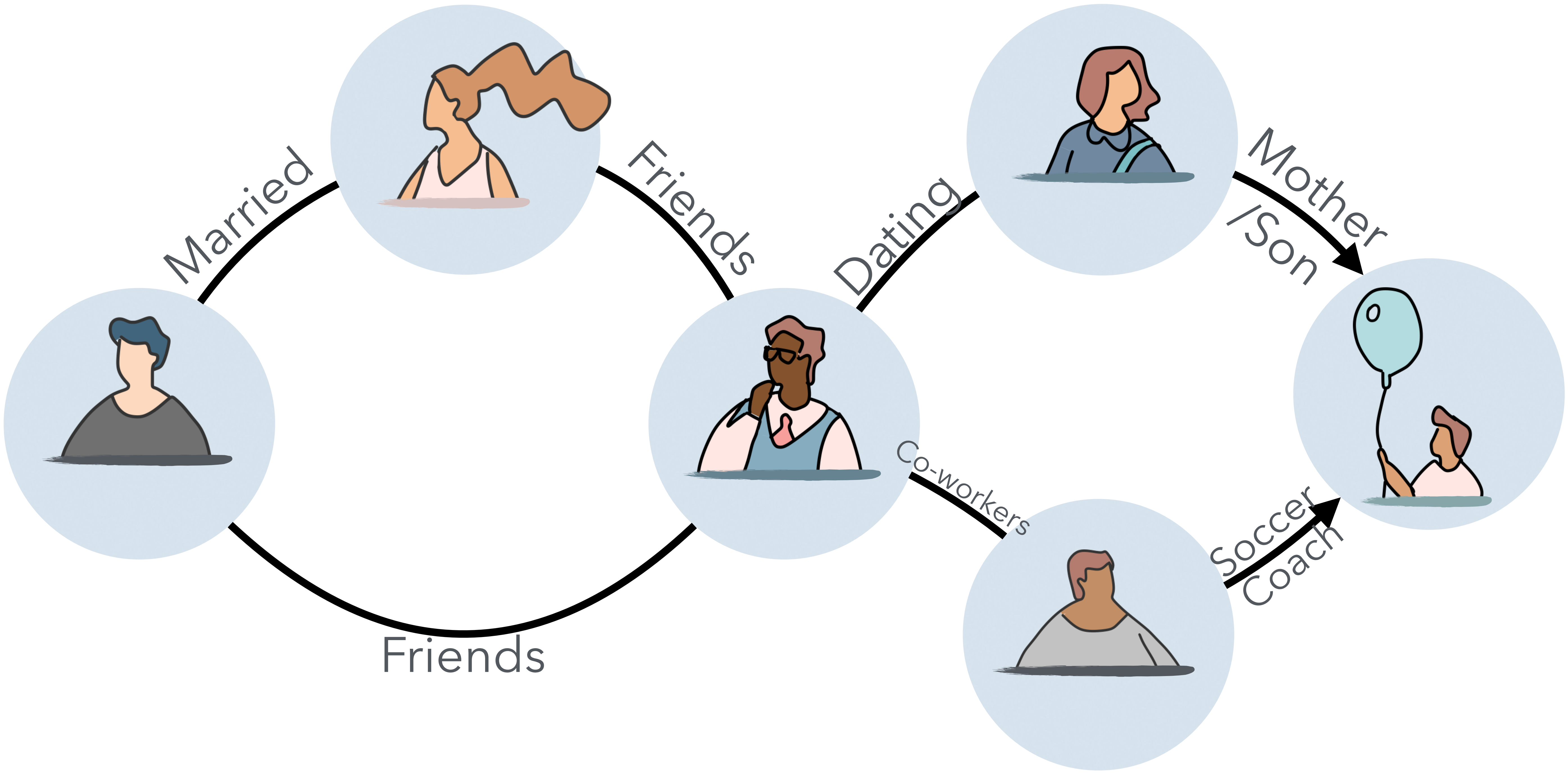
Independent views can optimize for topology and attribute independently.

Not great for tasks on topological structures beyond a single node or edge.

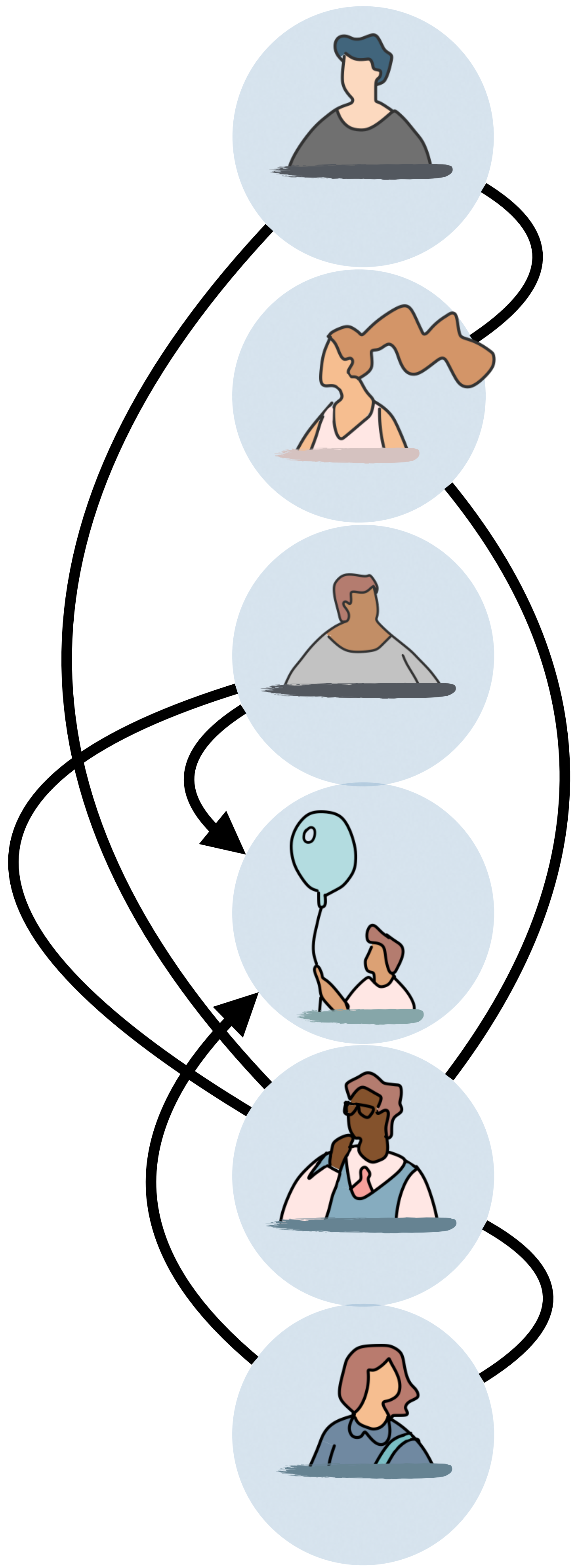
Recommended for large networks and/or very large numbers or heterogeneous types of node and link attributes

Integrated

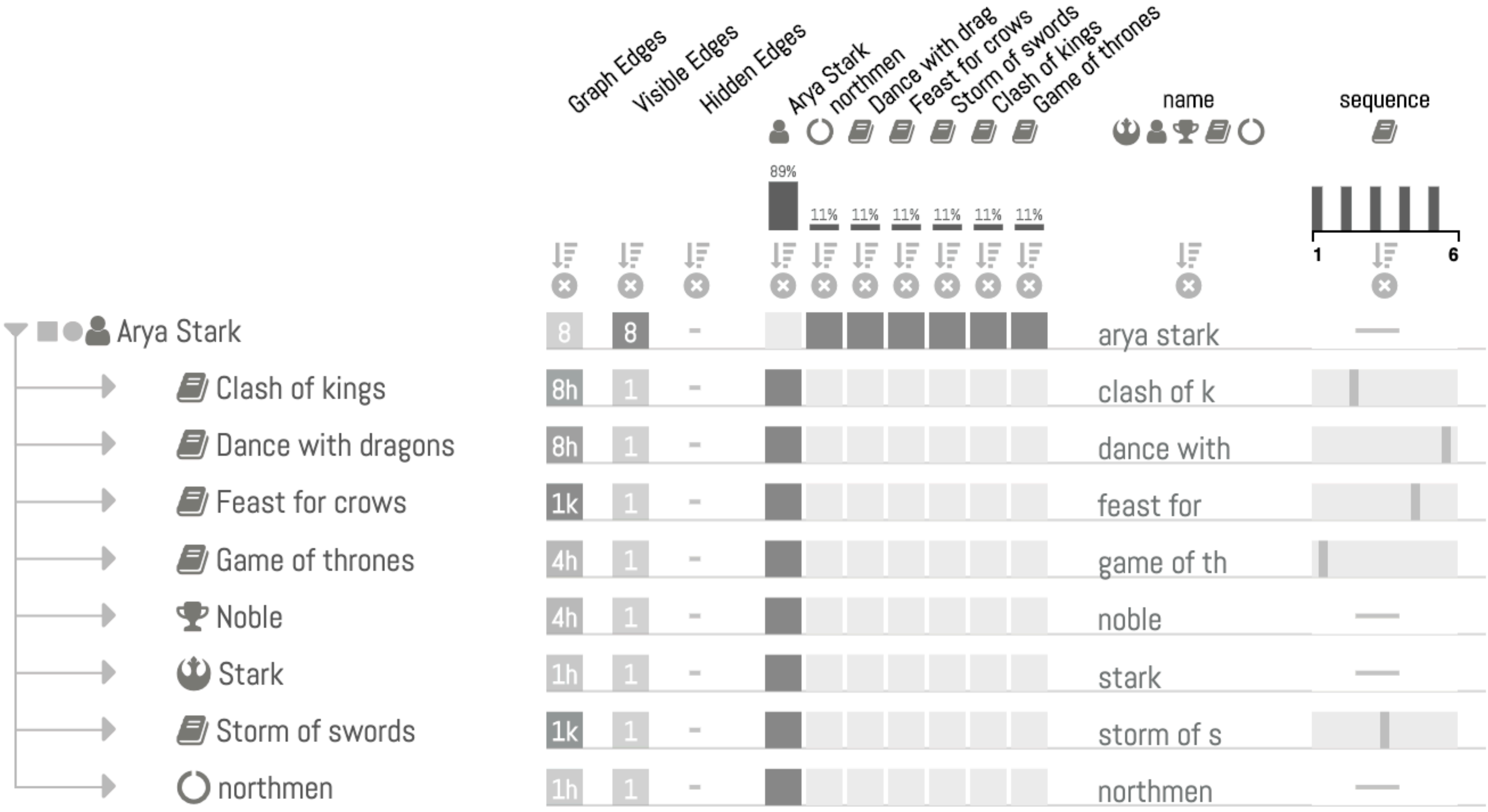




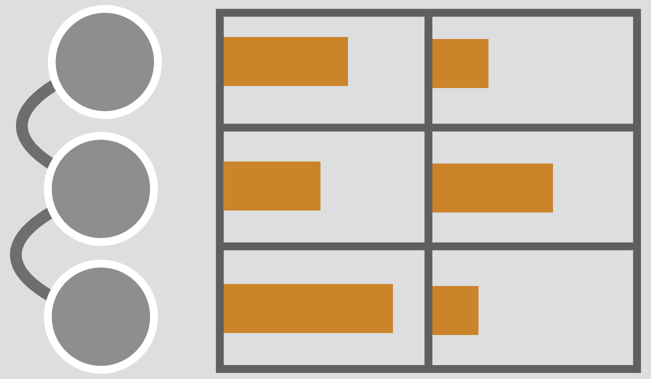
Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3



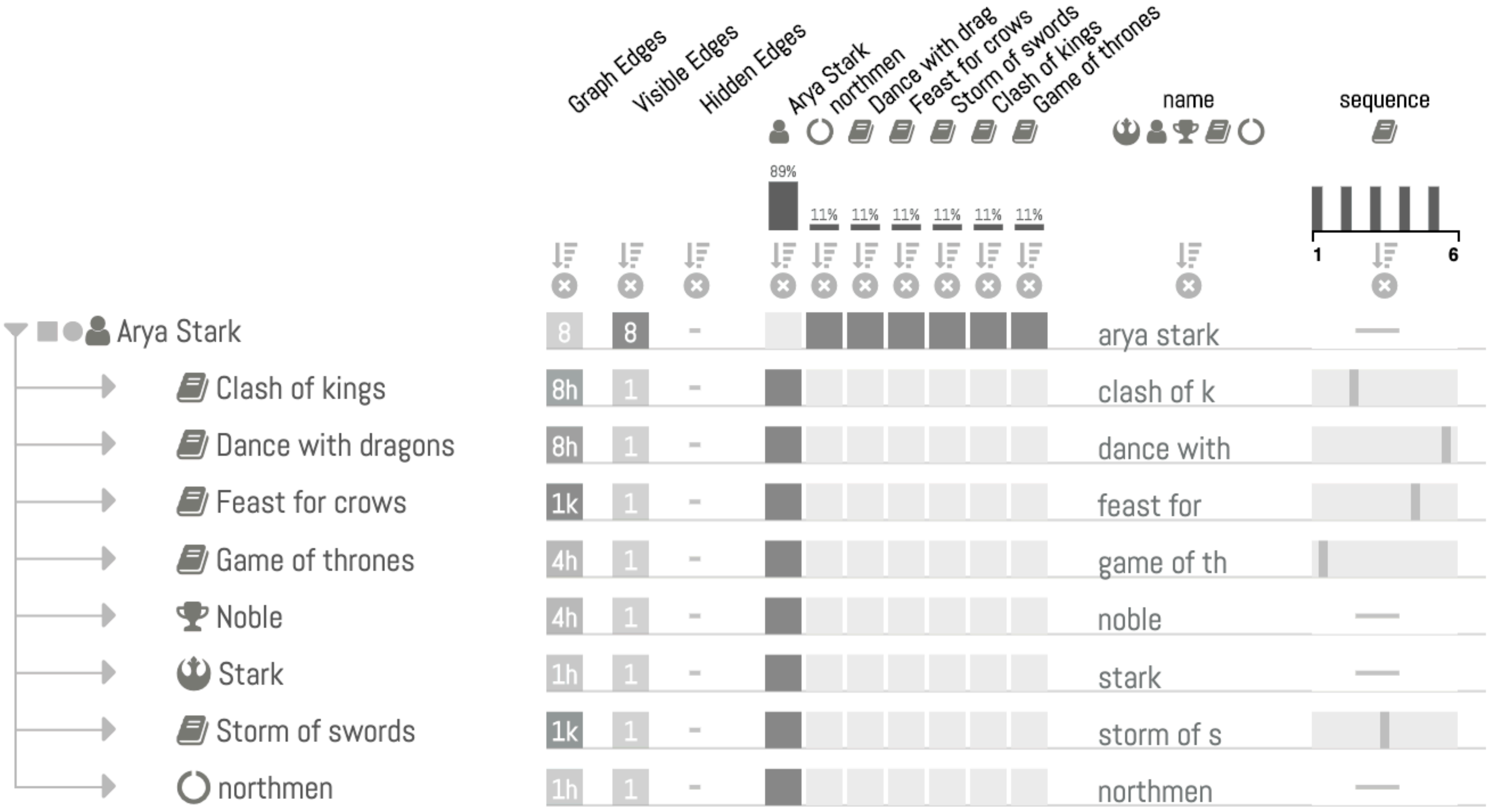
Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3



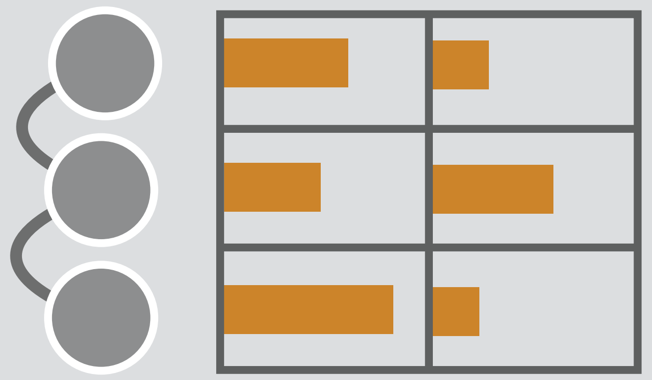
Juniper *Nobre et al. 2018*



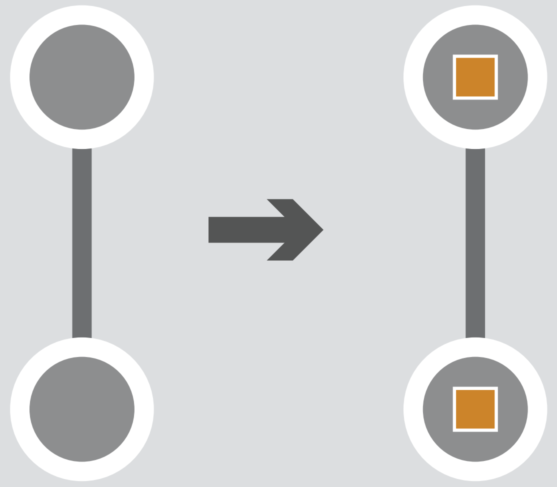
Integrated



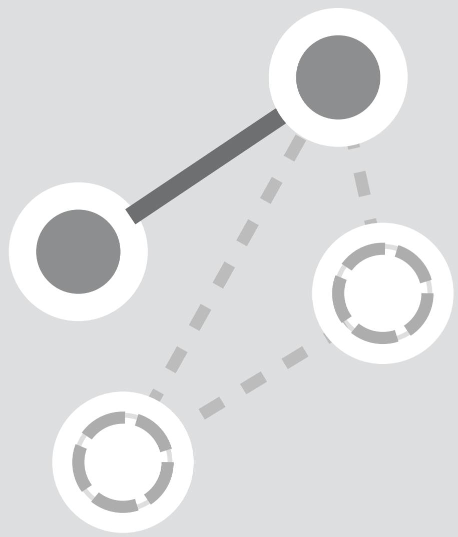
Juniper *Nobre et al. 2018*



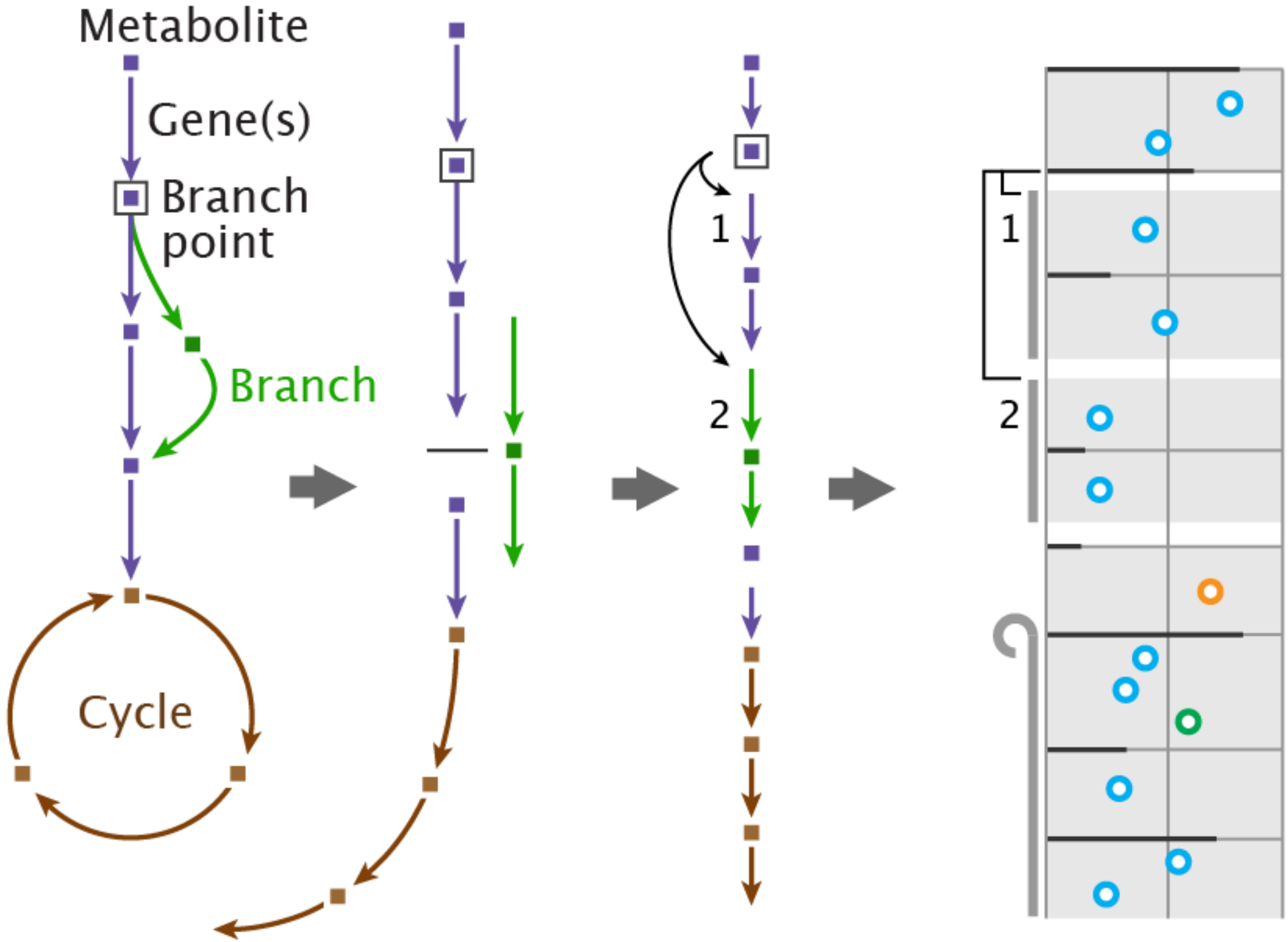
Integrated



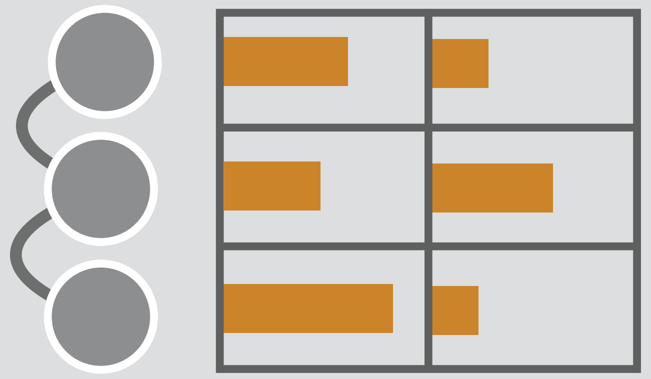
Deriving New Attributes

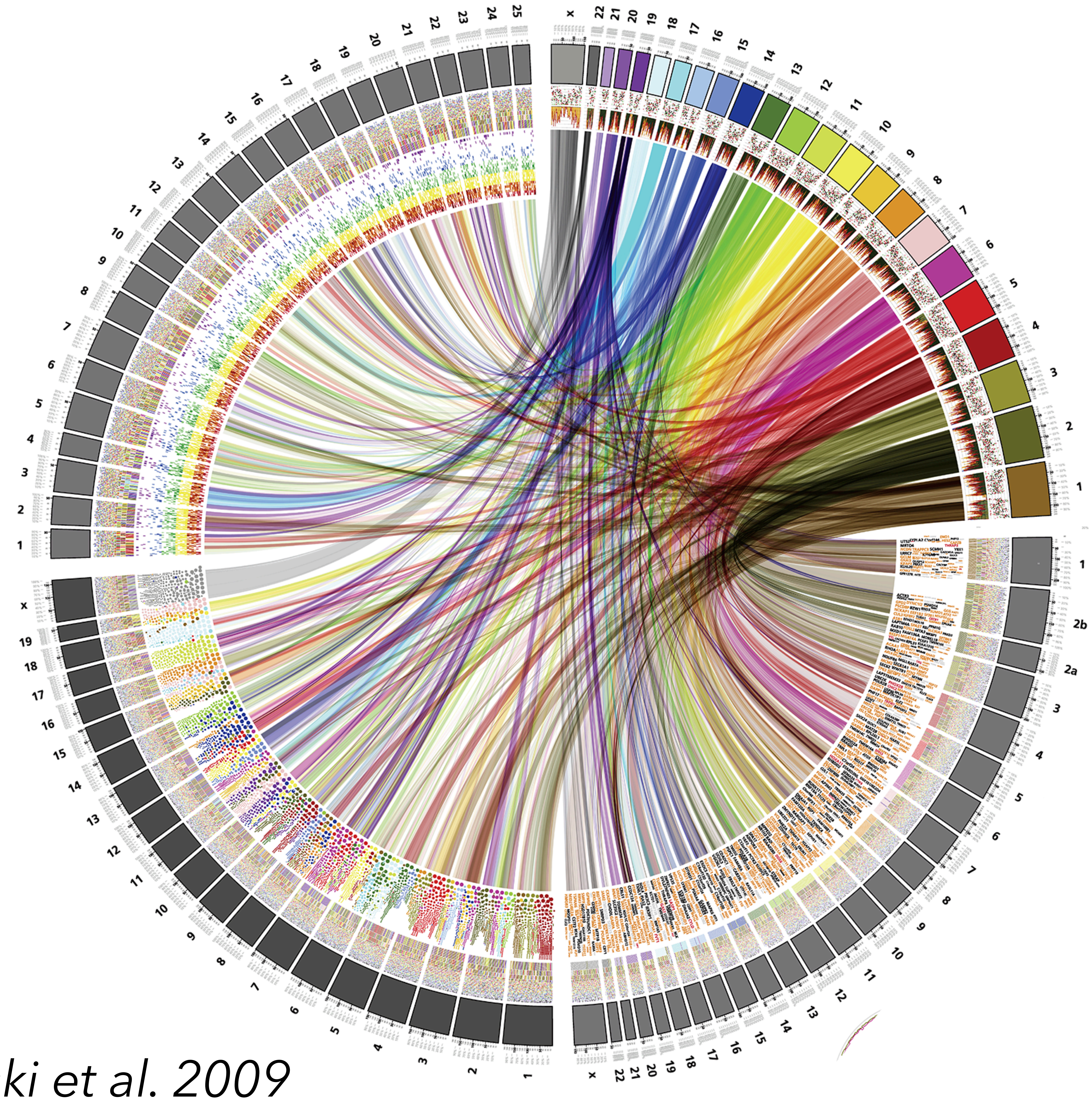


Querying and Filtering



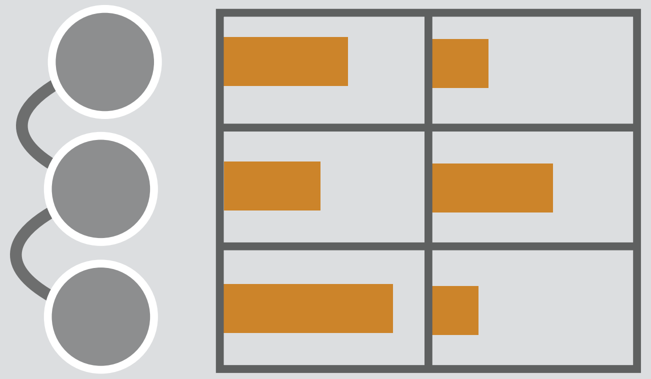
Pathline Meyer *et al. 2010*





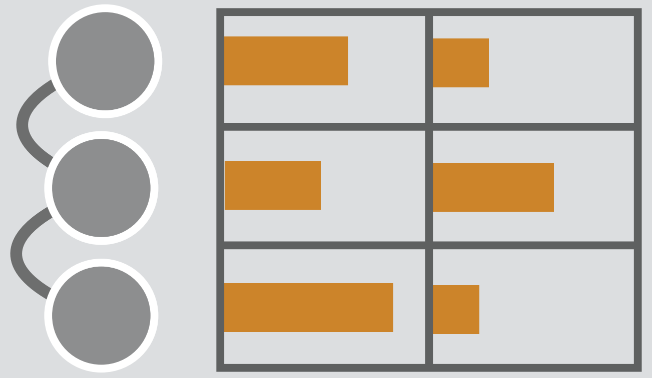
Circos

Krzywinski et al. 2009



Integrated

Integrated



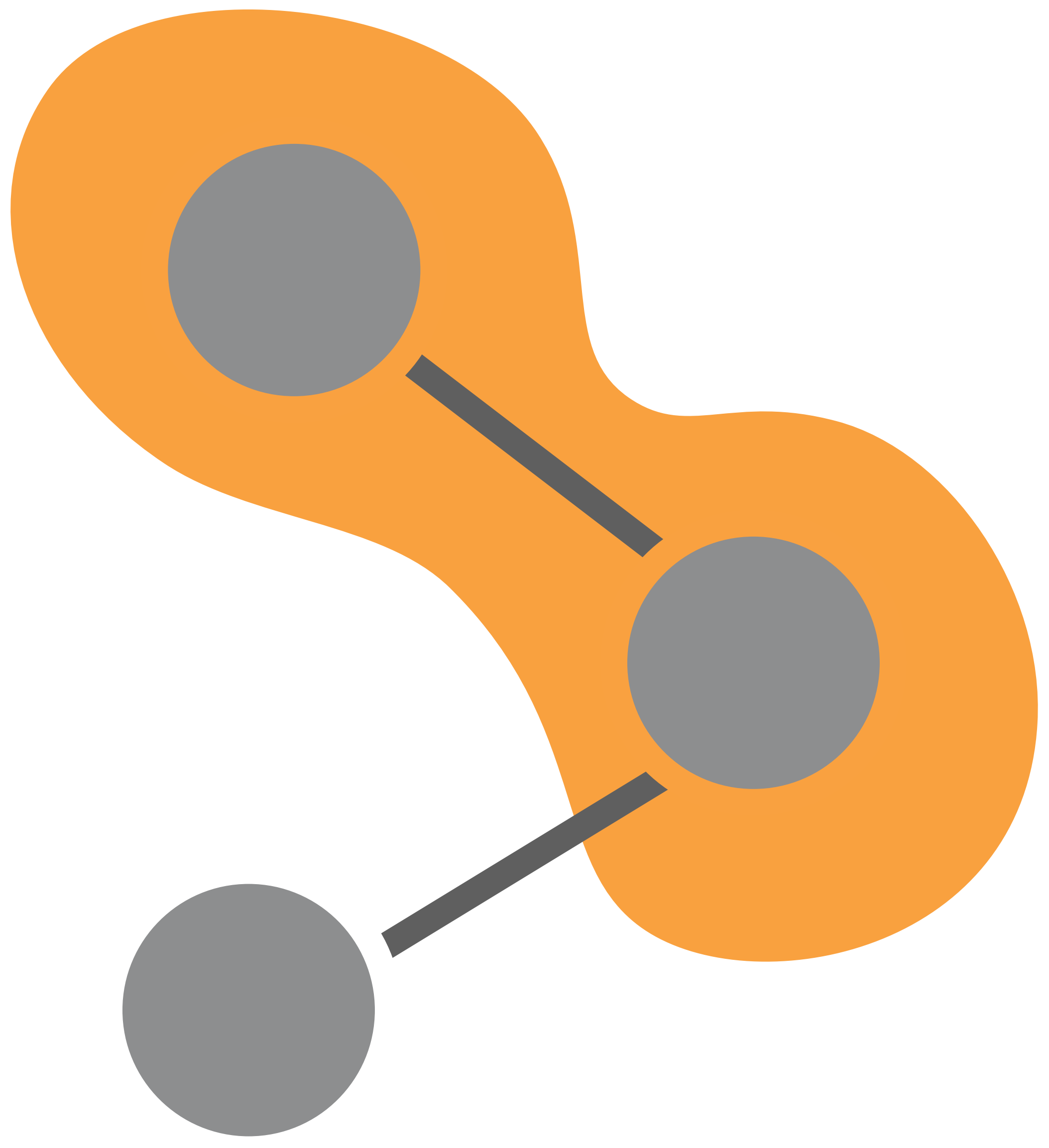
Integrated

good at integrating attributes with topology, if the topology can be represented in a linear layout.

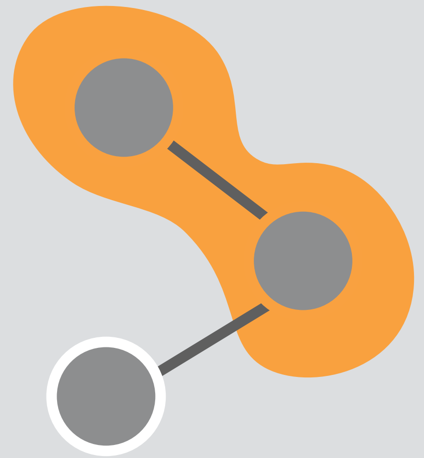
Not suitable for networks that can not be sensibly linearized.

Recommended for networks with several, heterogenous, node attributes and well suited for tasks on single nodes, neighbors, and paths

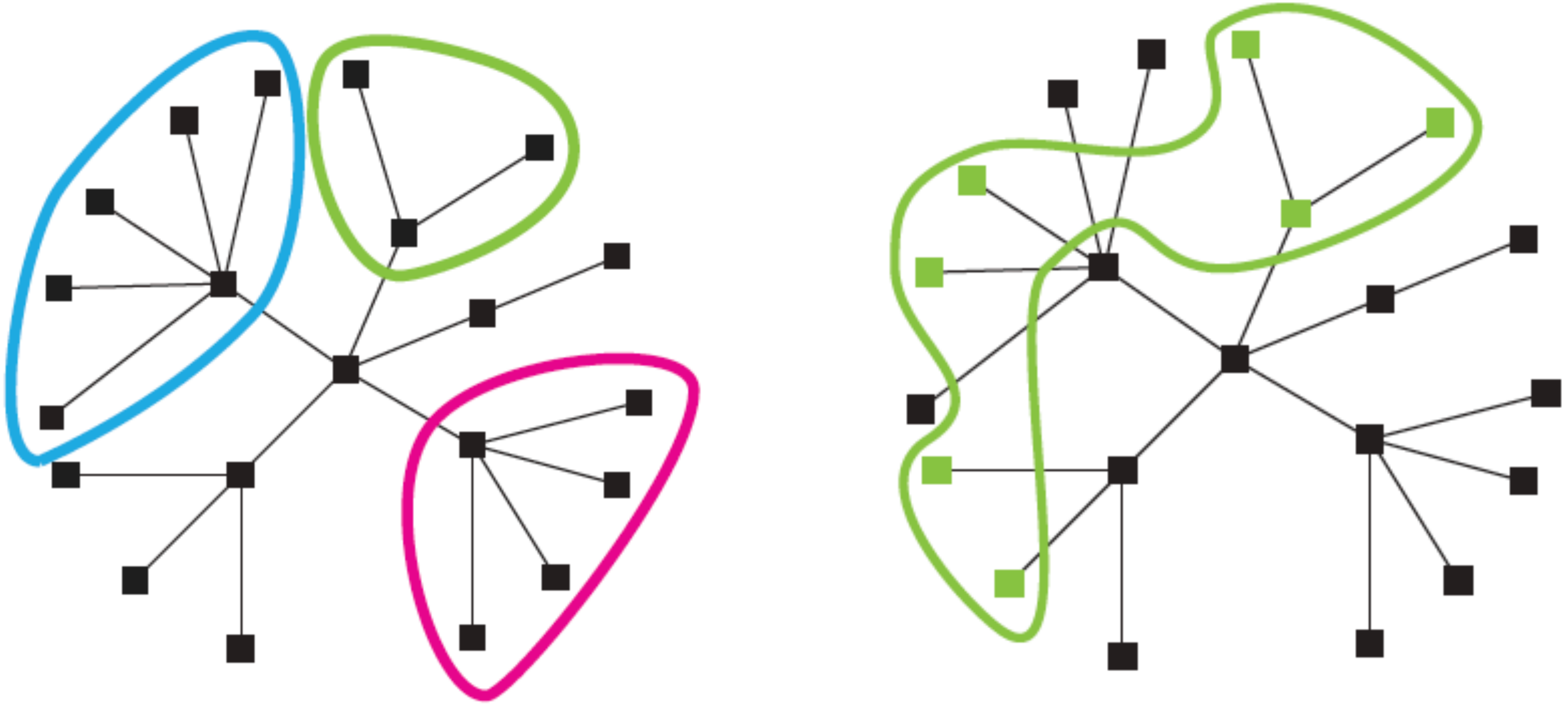
Overloaded

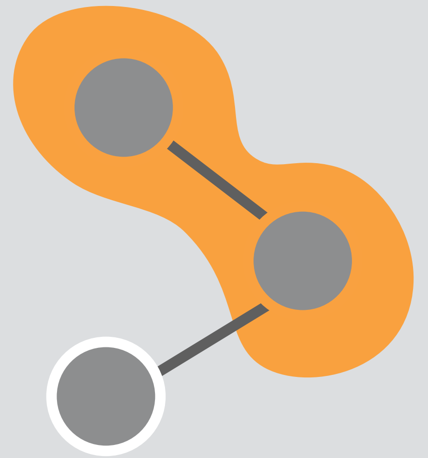


GMaps *Gansner et al. 2010*



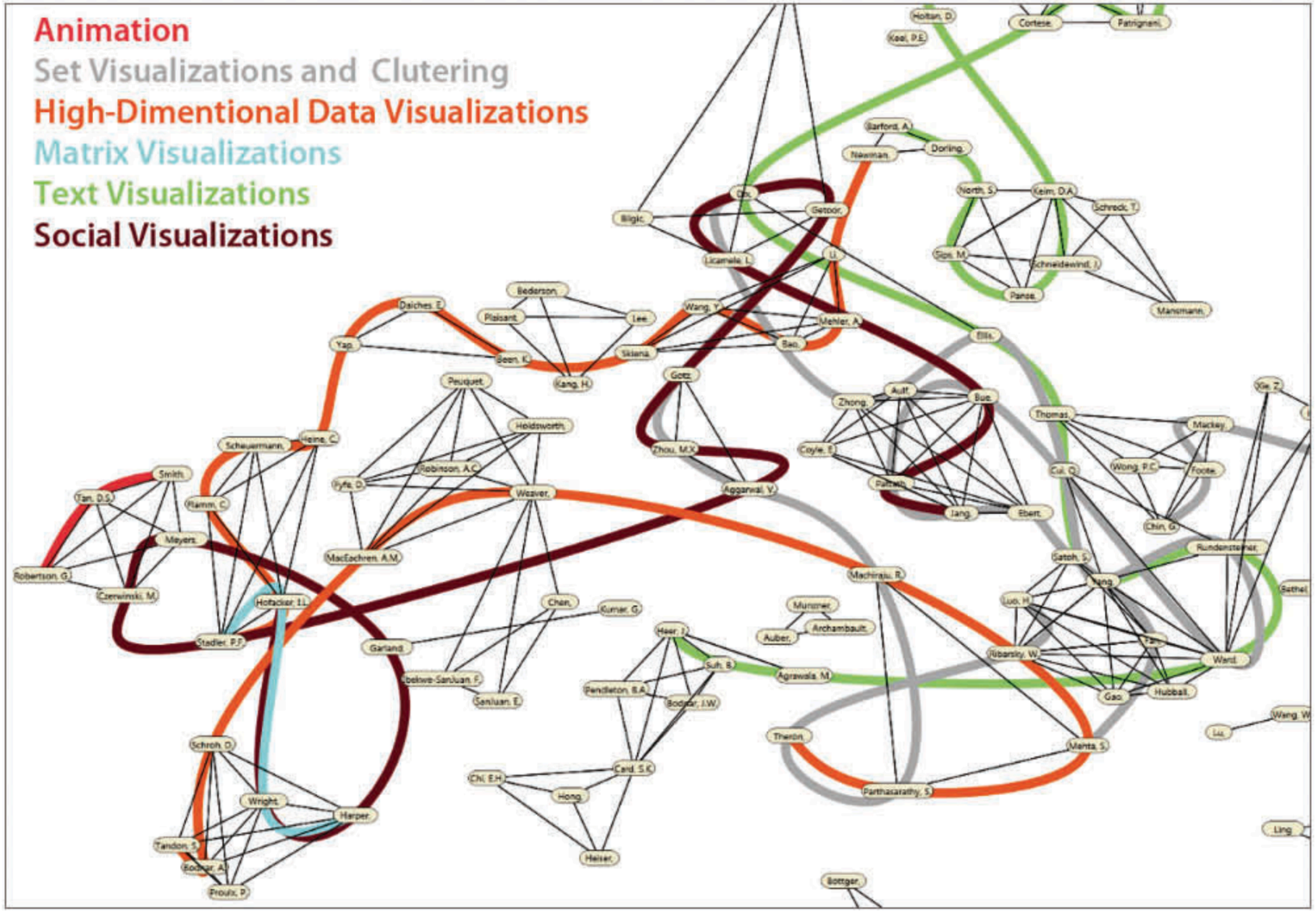
Overloaded



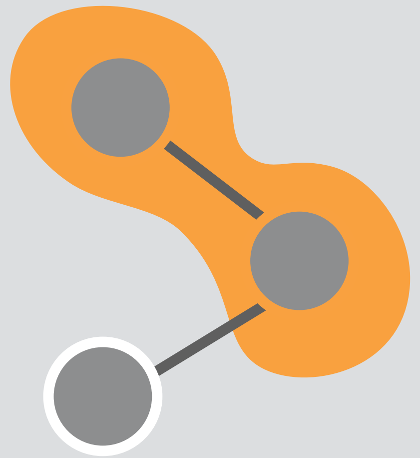


Overloaded

Bubble Sets *Collins et al. 2009*

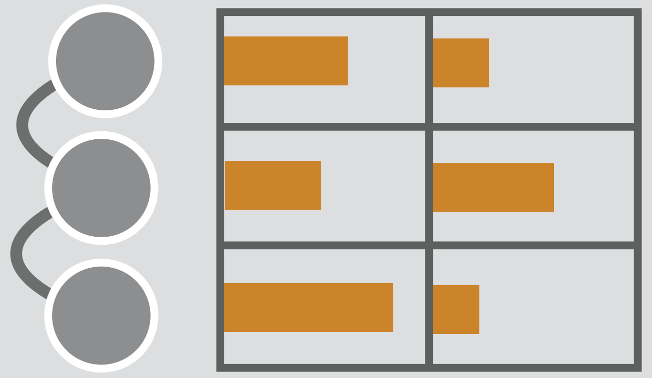


LineSets *Alper et al. 2011*



Overloaded

Overloaded



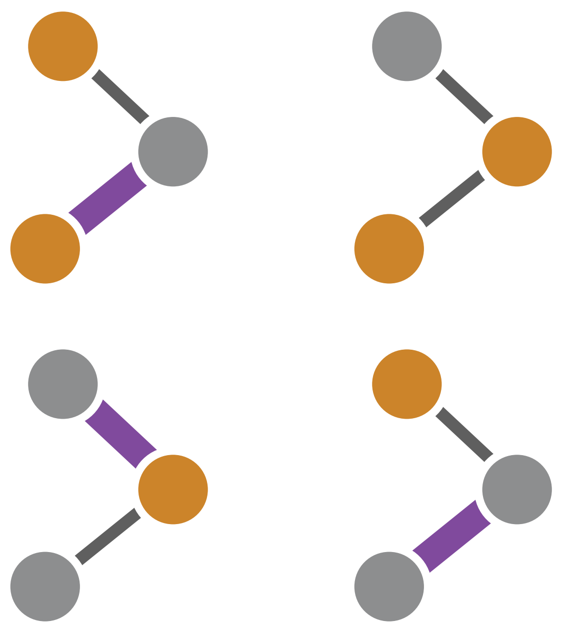
Integrated

good at displaying sets and clusters

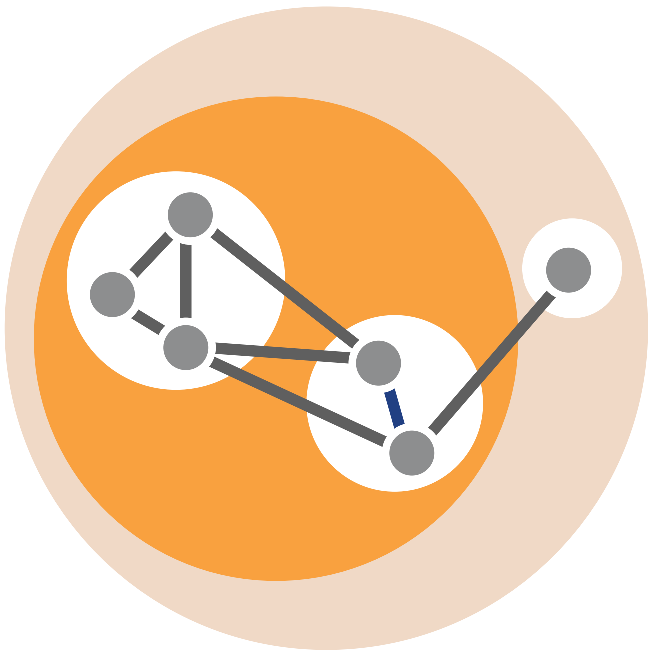
Not suitable for displaying more than one or two attributes at a time.

Recommended for recommend overloading for the particular use case of visualizing set-memberships or clusters on top of node-link diagrams

Layout Operations

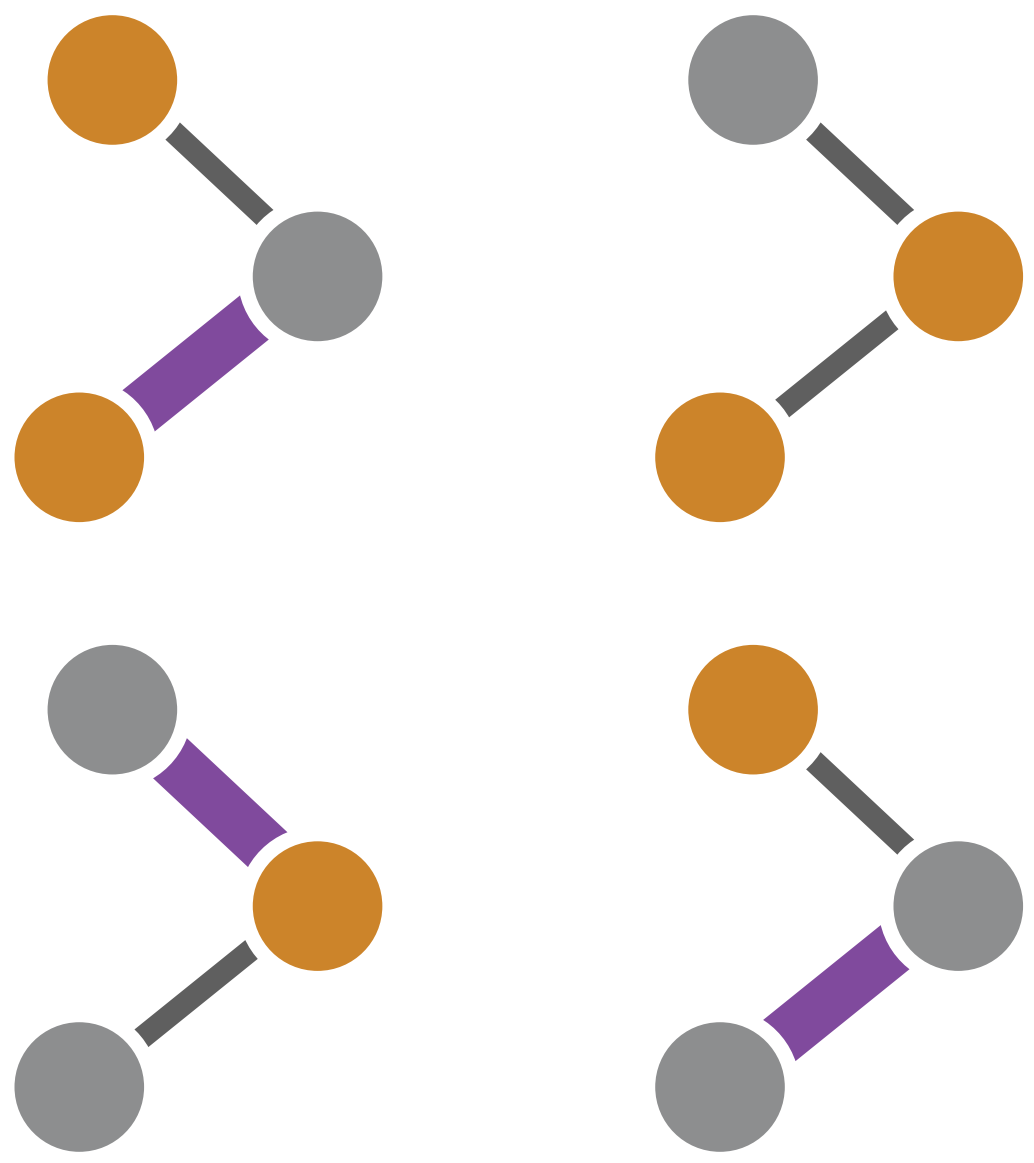


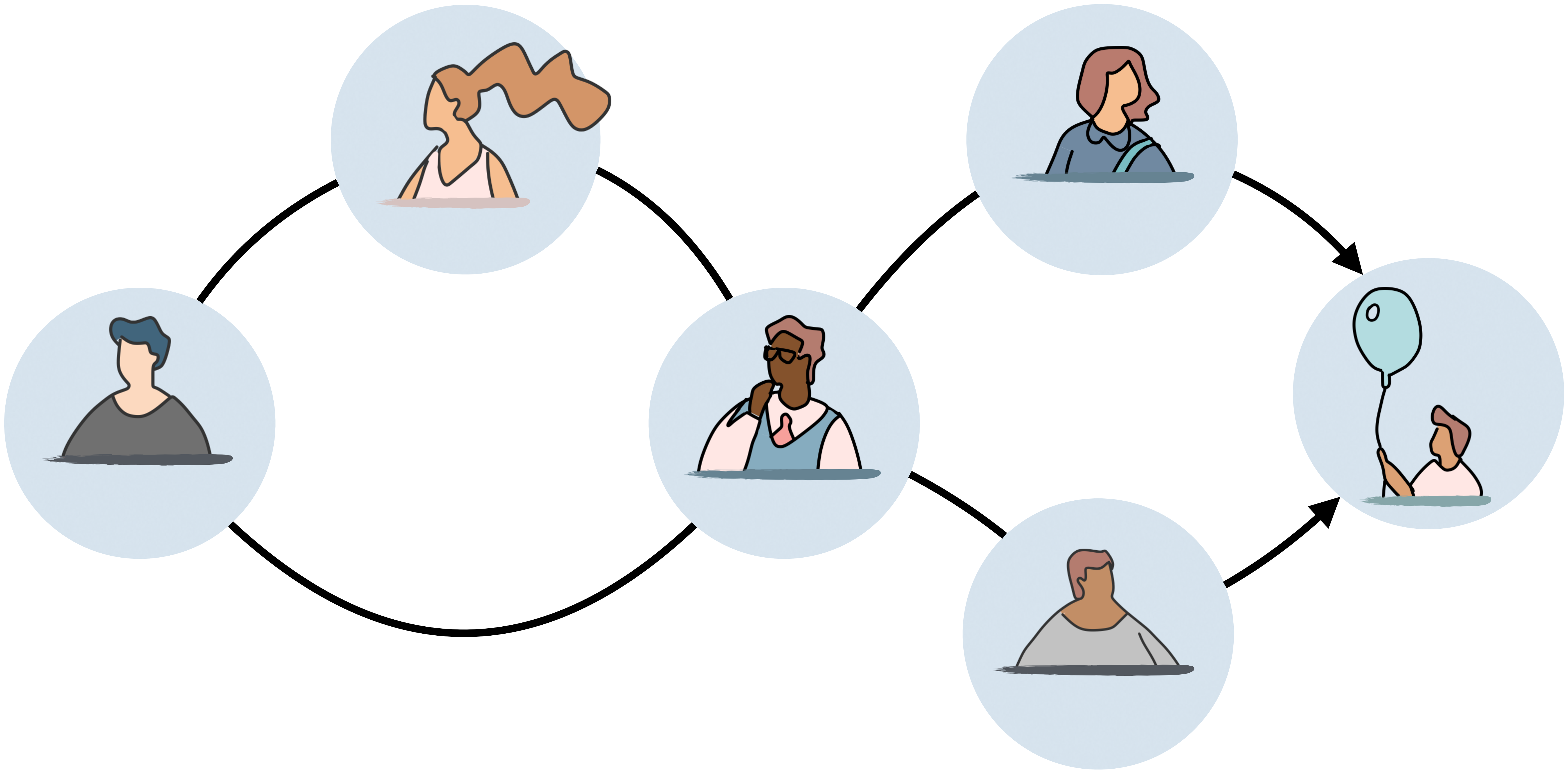
Small Multiples



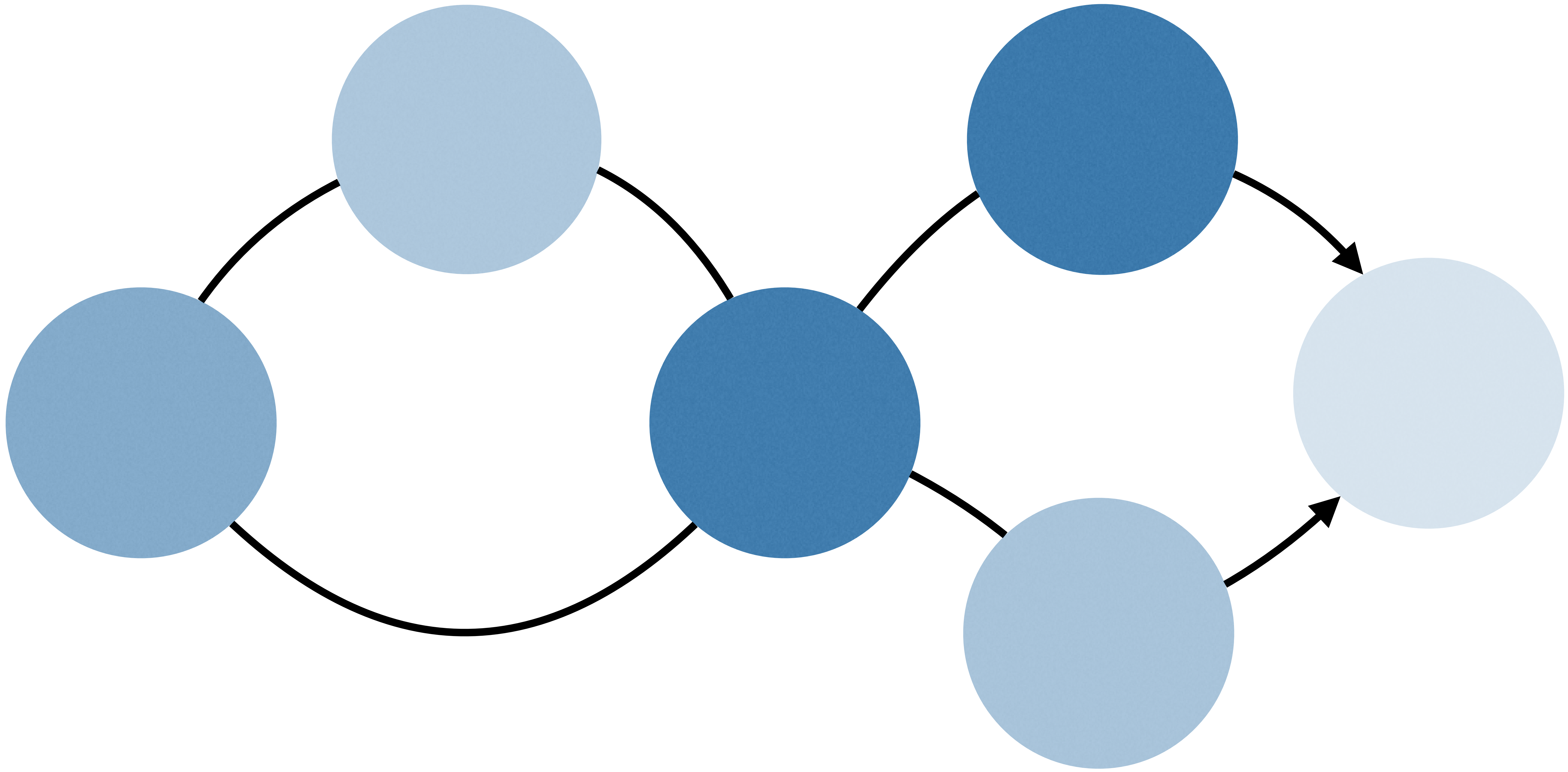
Hybrids

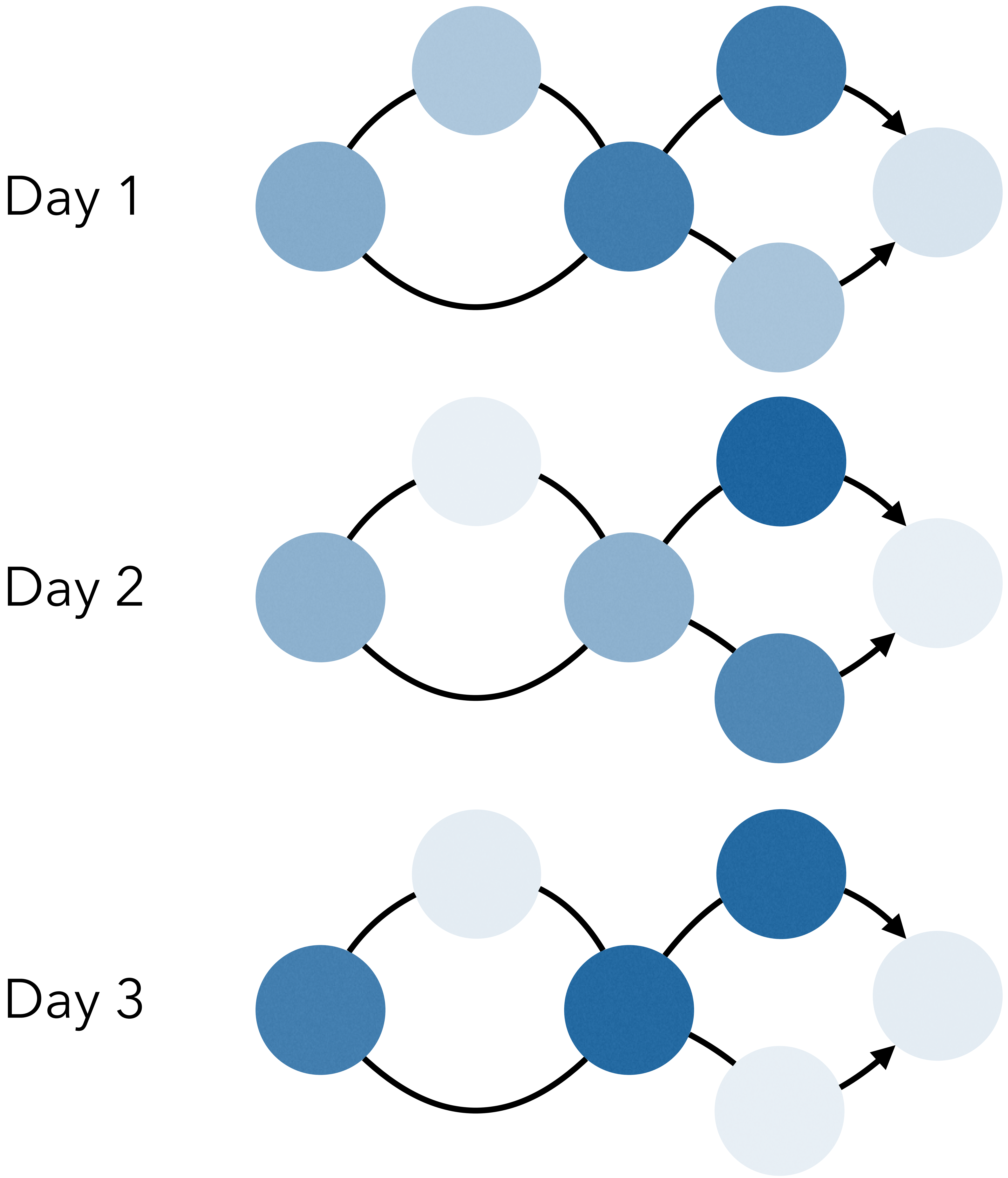
Small Multiples

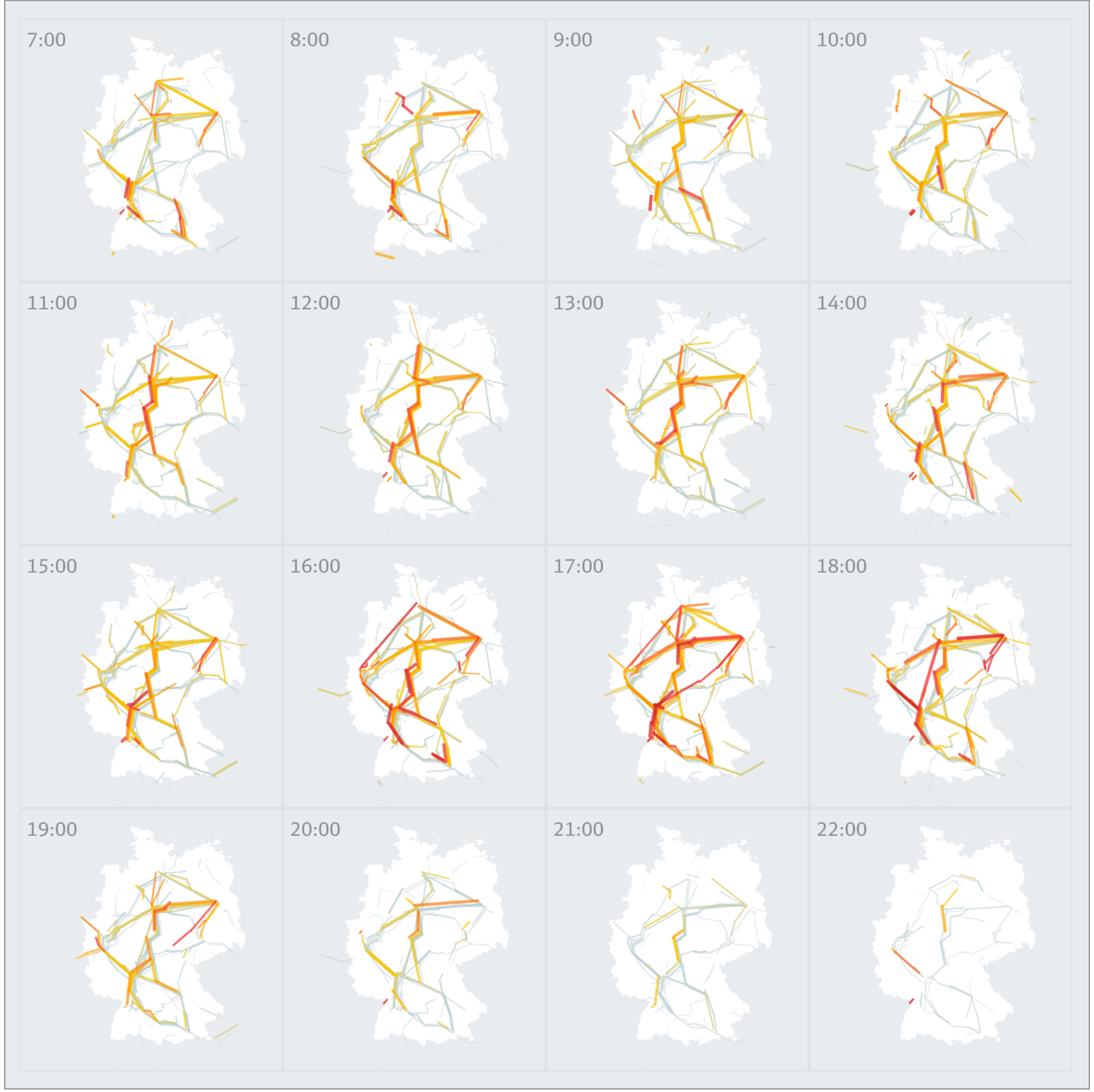


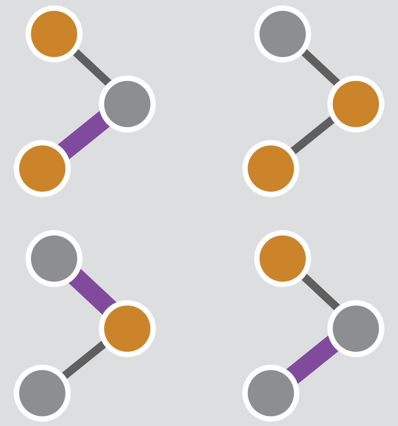


Day 1

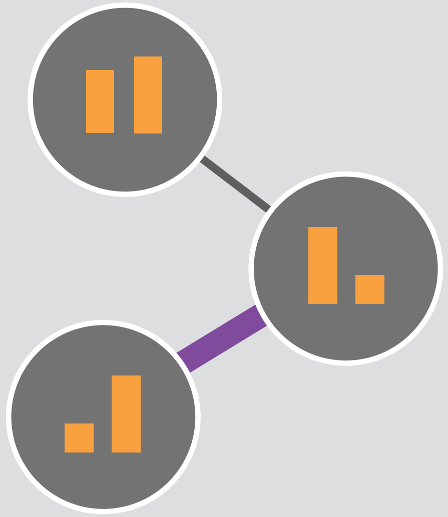



Day 1
Day 2
Day 3



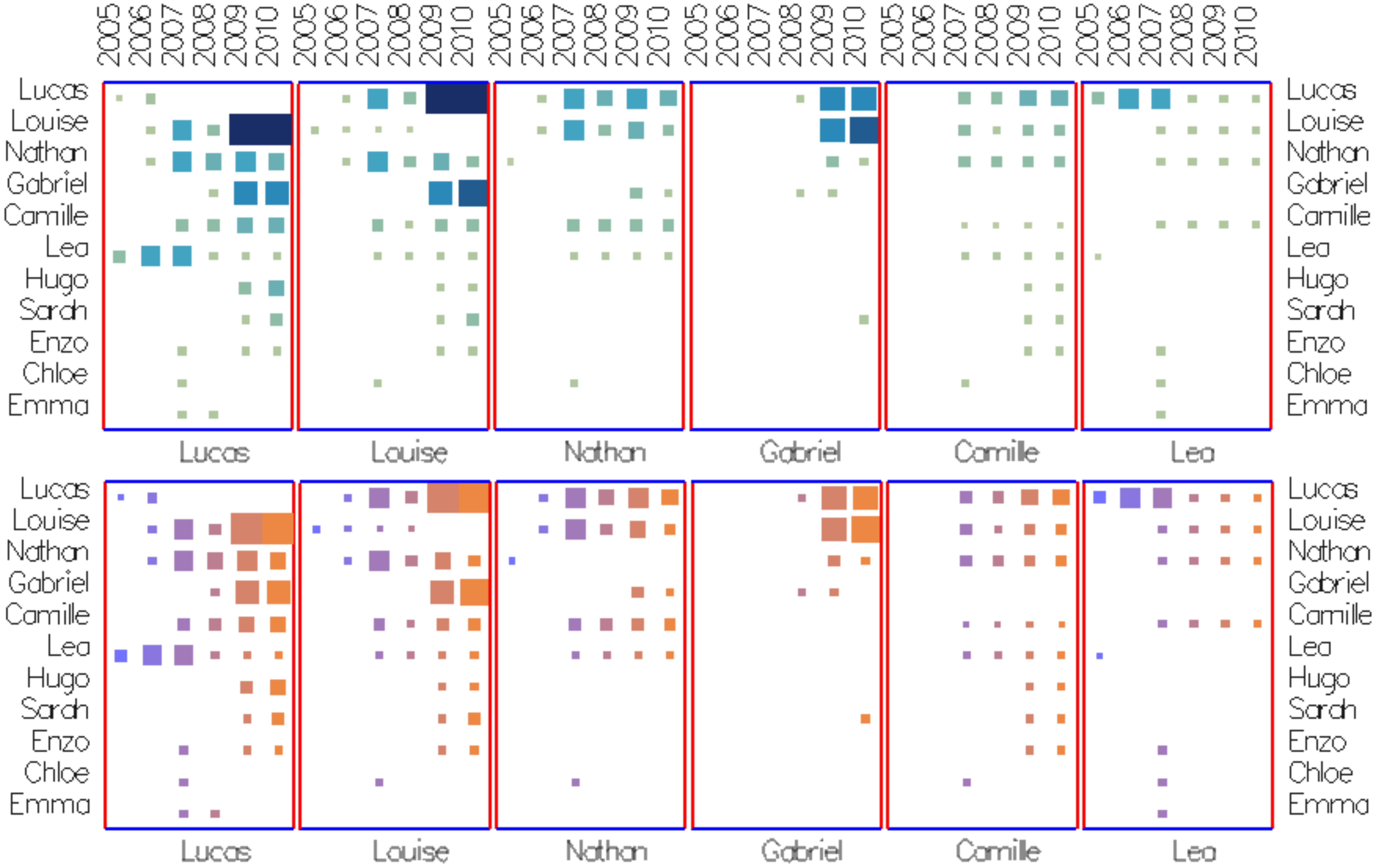


Small Multiples

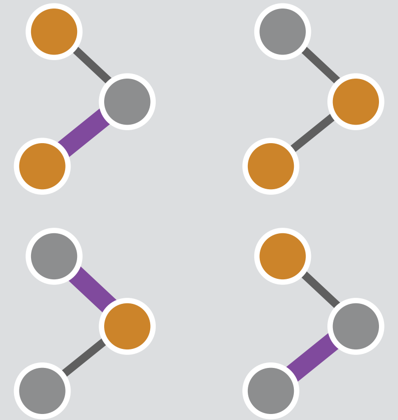


On-Node / On-Edge
Encoding

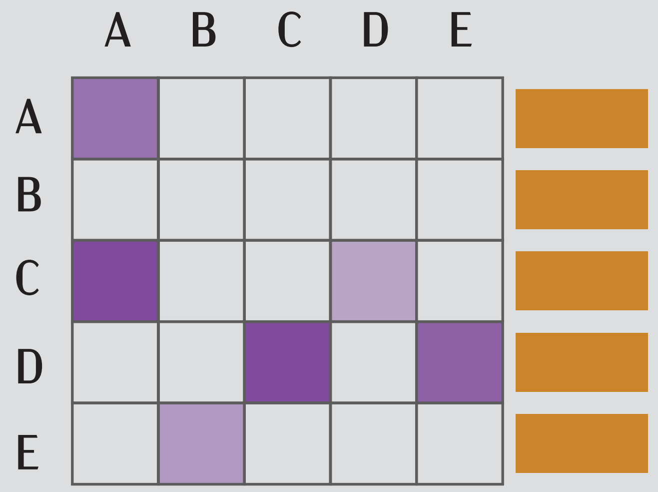
Peakspotting - https://truth-and-beauty.net/projects/peakspotting



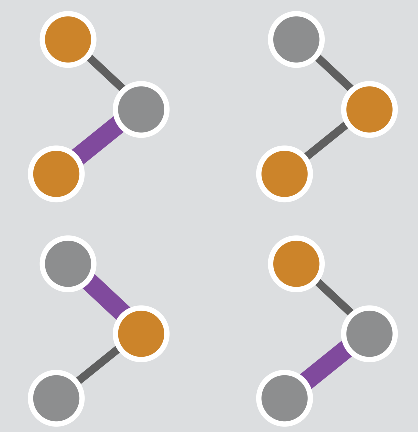
Bach et al. 2014



Small Multiples



Adjacency Matrix



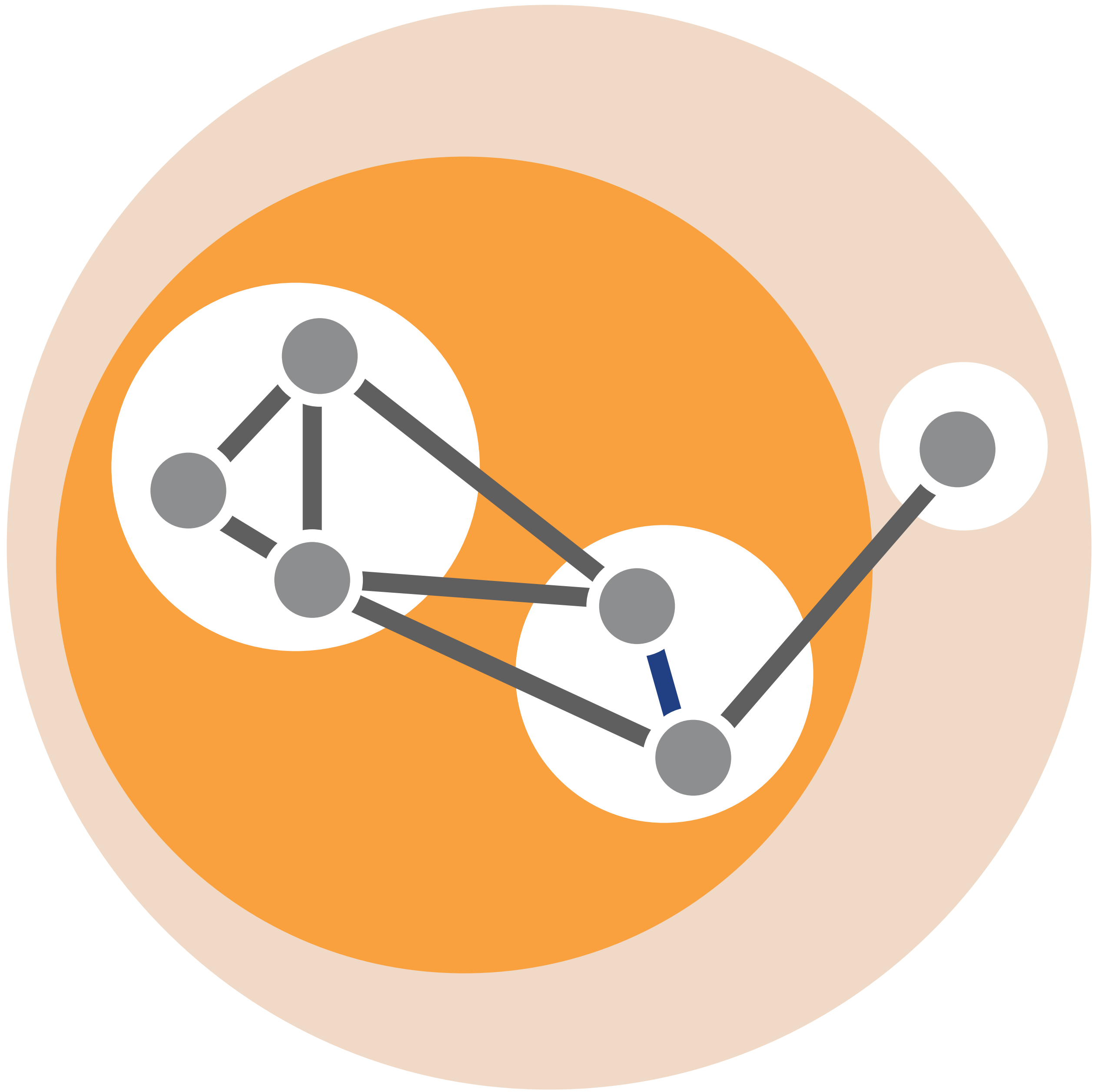
Small Multiples

Common layout facilitates attribute comparisons in specific topological features

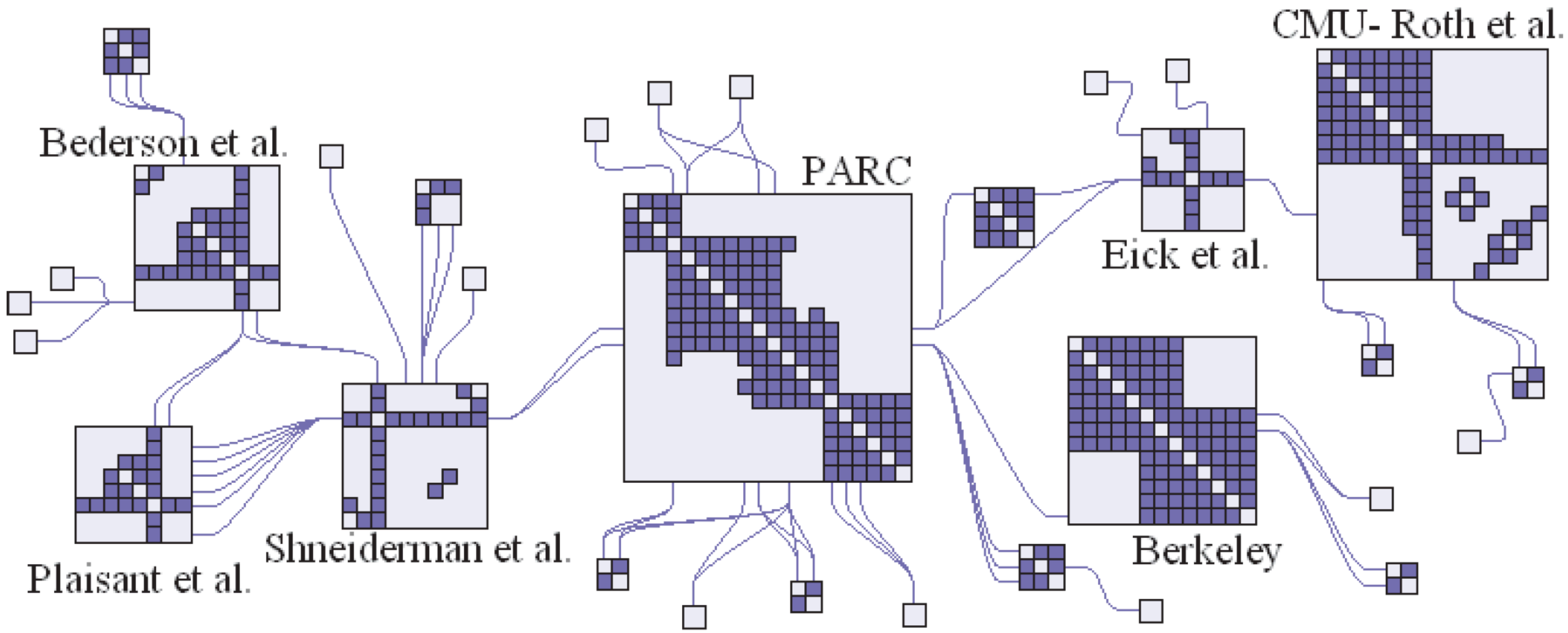
Not ideal for large networks, or tasks on clusters

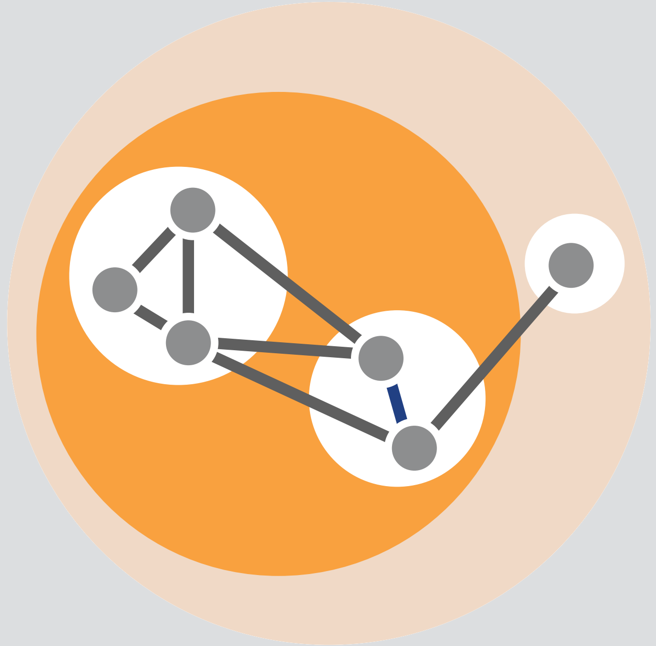
Recommended for small networks where the tasks are focused on attribute comparison

Hybrids



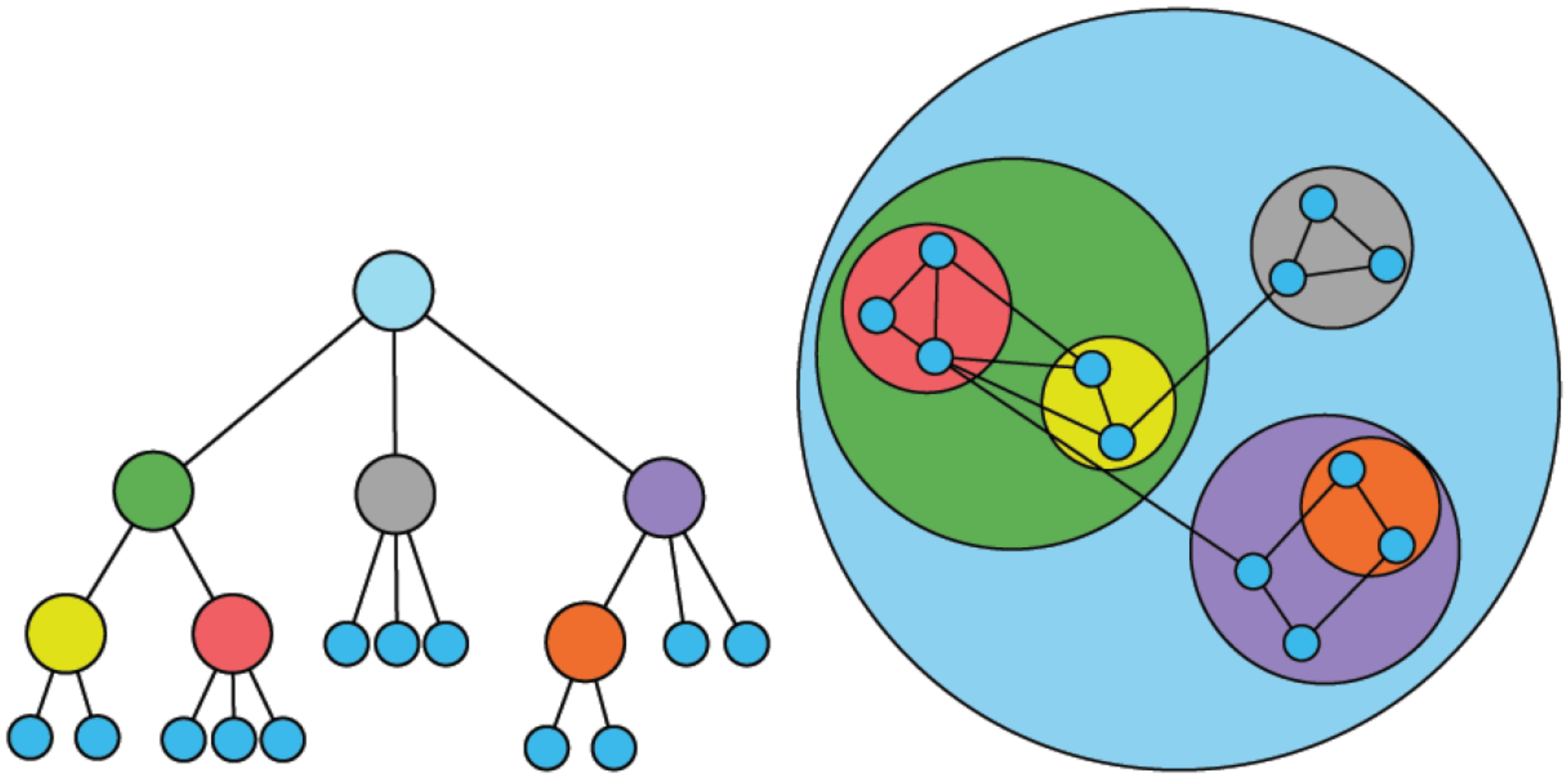
NodeTrix *Henry et al. 2007*

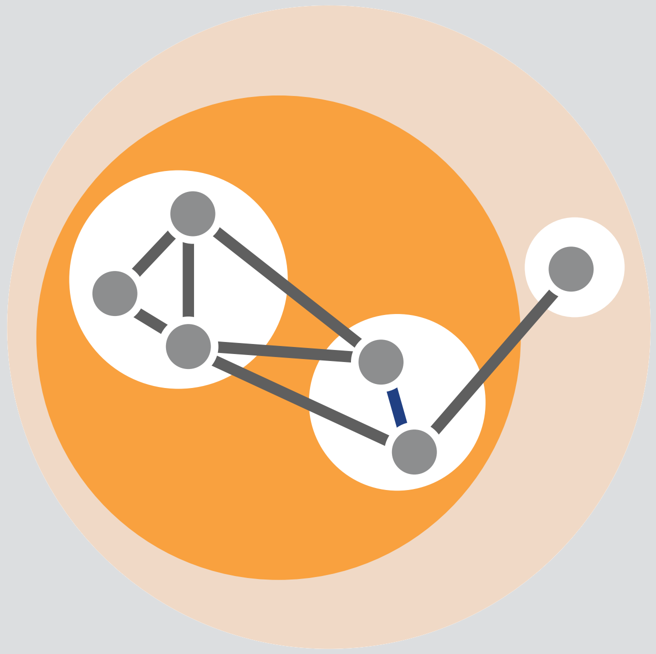




Hybrids

GrouseFlocks *Archambault et al. 2008*





Hybrids

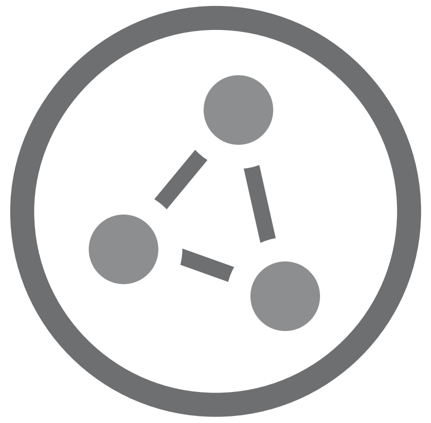
Hybrids

Can be useful for networks with irregular degree distribution

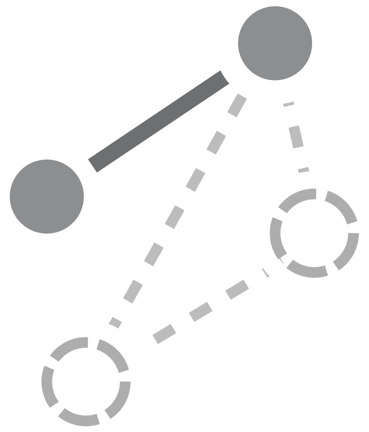
Adds complexity since users must parse different techniques simultaneously.

Recommended for networks with irregular degree distribution and few attributes

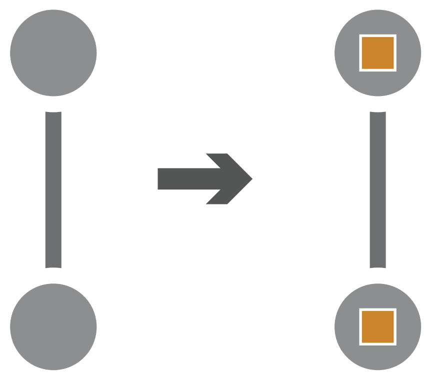
Data Operations



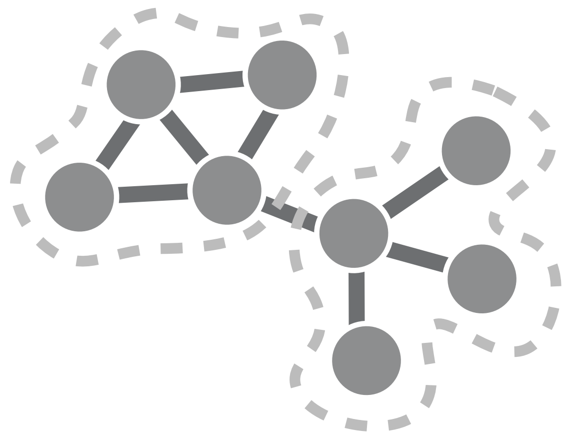
Aggregating Nodes/Edges



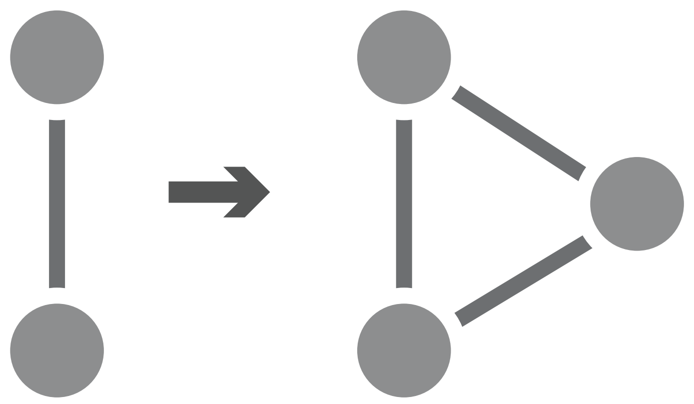
Querying and Filtering



Deriving New Attributes



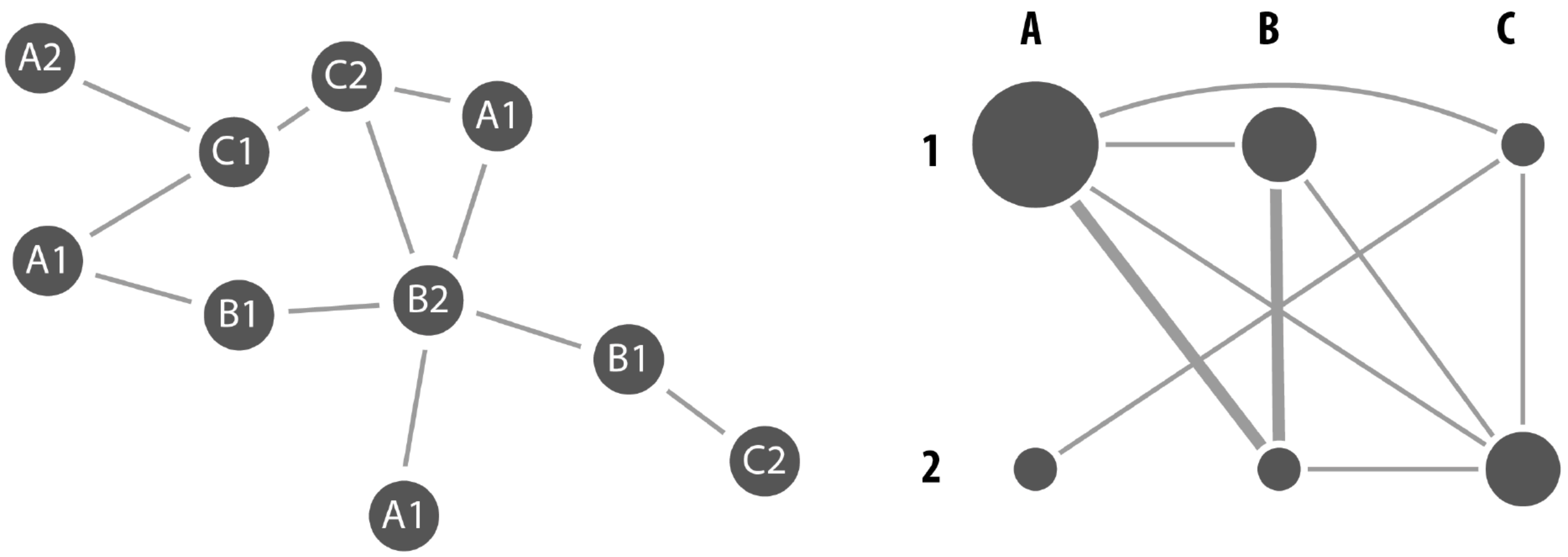
Clustering



Converting Attributes/Edge to Nodes

Elzen and Wijk, 2014

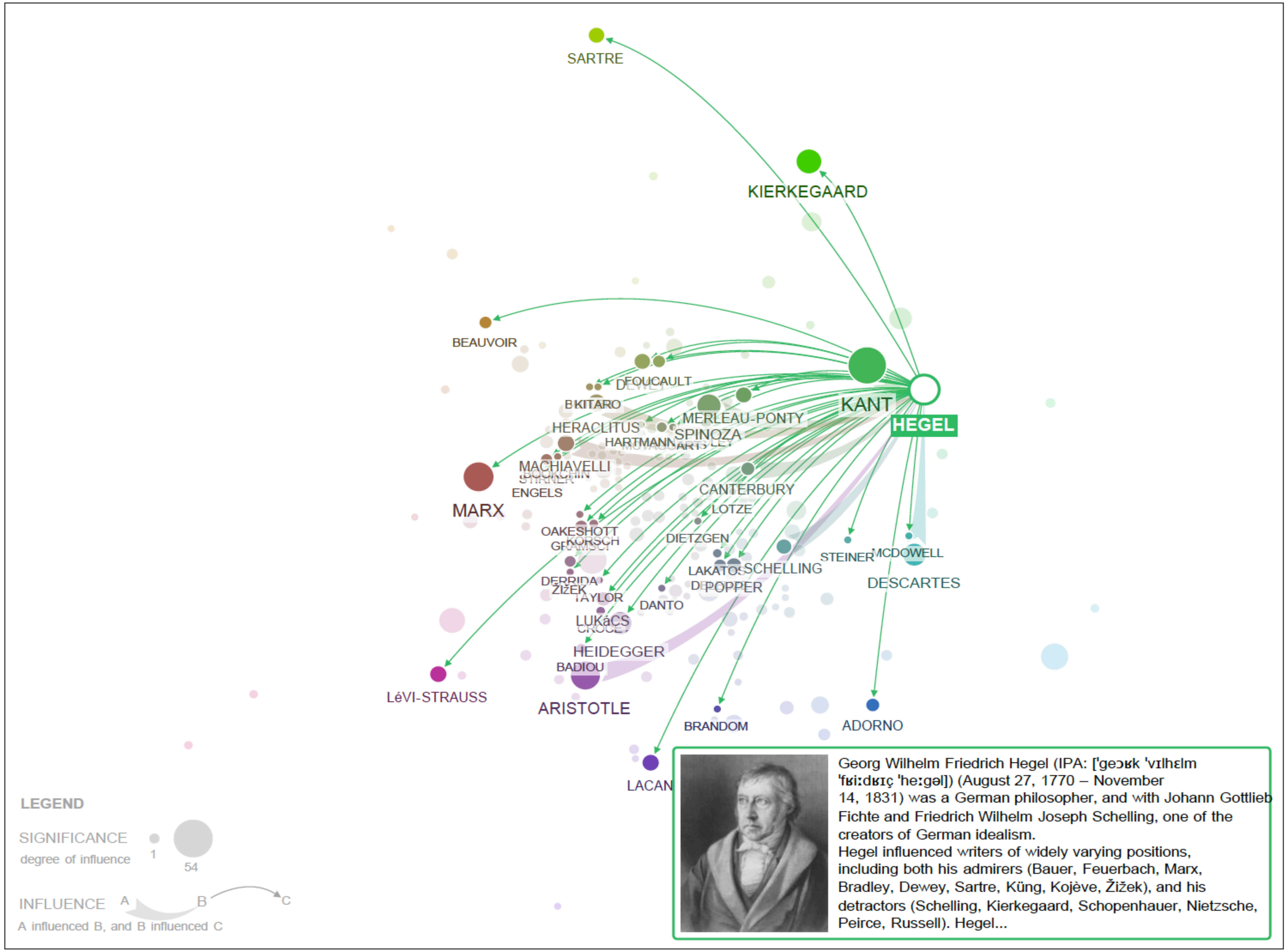
Aggregating Nodes/Edges

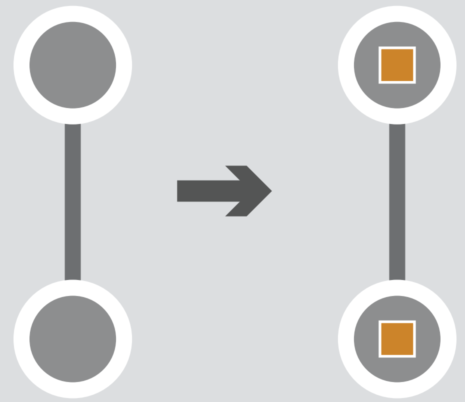


Node-Link Diagram **PivotGraph Roll-up**

Wattenberg, 2006

Aggregating Nodes/Edges





Deriving New Attributes

Edge Map *Dork et al. 2011*

Multivariate Network Visualization Techniques

A companion website for the STAR Report on Multivariate Network Visualization Techniques.

HOME   TECHNIQUES   WIZARD

About

This is a companion website for a review article on multivariate network visualization techniques.

Multivariate networks are networks where both the structure of the network and the attributes of the nodes and edges matter. It turns out, these are very common. Every person in a social network, for example, has both, relationships and lots of other characteristics, such as their age, the school they went to, or the city they live in. Multivariate network visualization techniques are designed to be able to show both, these attributes and the structure. Using these visualization techniques, we can analyze, for example, if a network of friends predominantly went to the same high school.

The visualization research community has developed many techniques to visualize these kinds of networks, and our review article – and this website – are designed to help you sort through these options.

Browse through the techniques illustrated below, or use our wizard to find the right multivariate network visualization technique for your datasets and tasks!

Get in touch if you have questions or comments.

Use the Wizard

Technique recommendations to fit your needs!

Navigate to the wizard tab and select your specific network characteristics, such as the size of the network and its type, and what tasks are relevant for your analysis and receive technique recommendations that are best suited to your selection.

Read the Review Article

The State of the Art in Visualizing Multivariate Networks
Carolina Nobre, Miriah Meyer, Marc Streit, and Alexander Lex
To appear in Computer Graphics Forum (EuroVis 2019)

vdl.sci.utah.edu/mvnv/