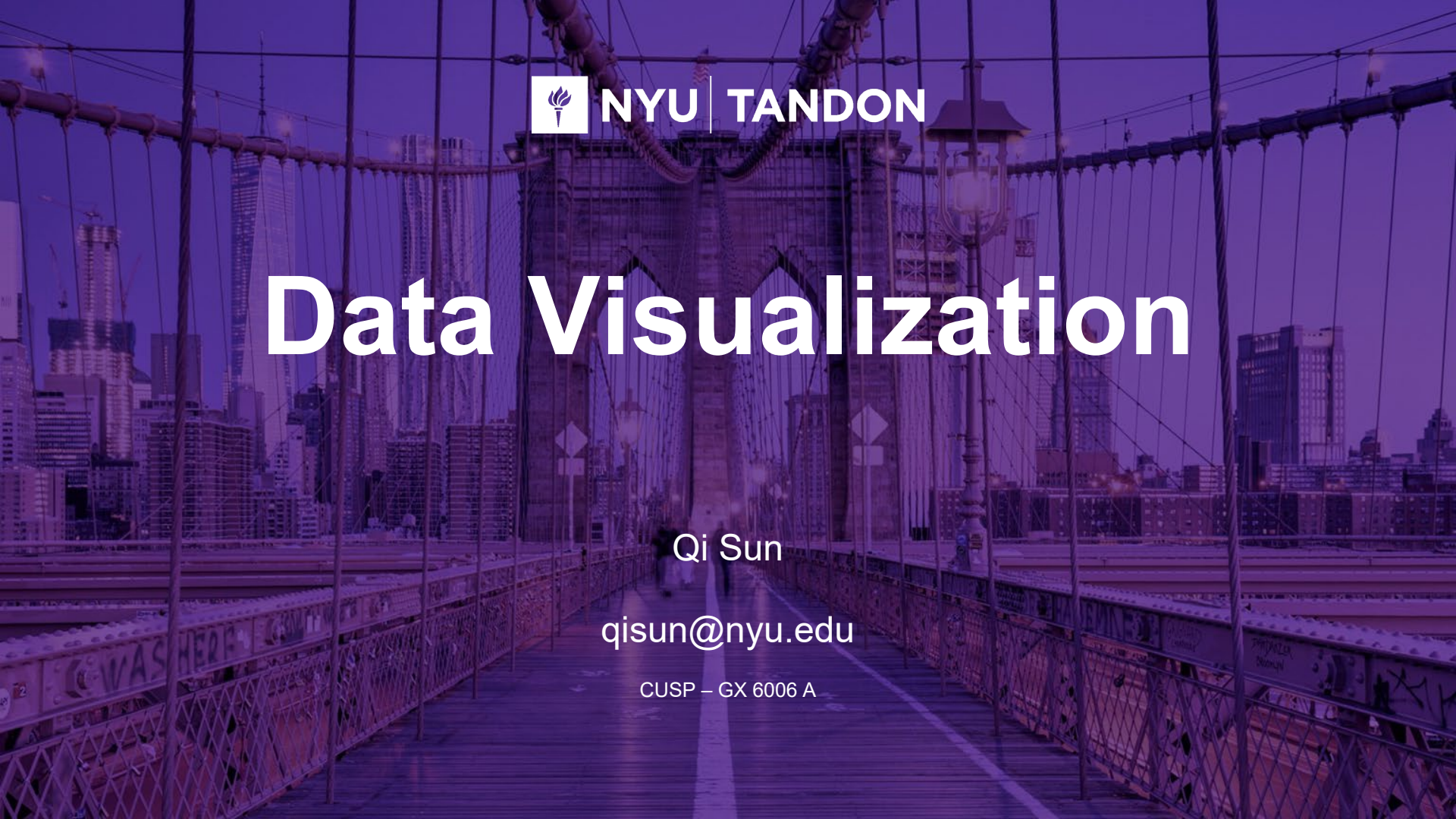


Data Visualization

Qi Sun

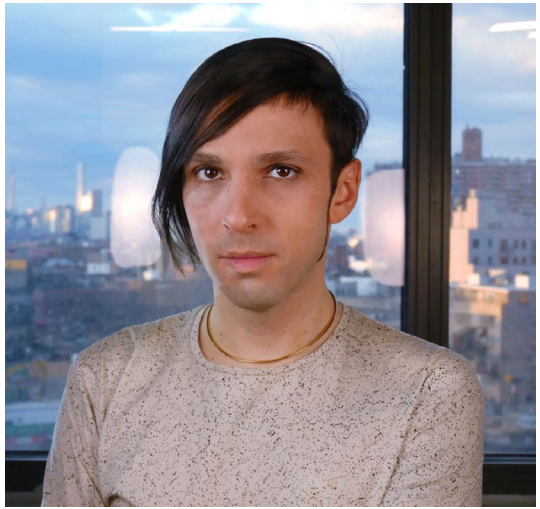
qisun@nyu.edu

CUSP – GX 6006 A

Logistics

- Guest lecture next week : in-person

Omer Shapira, NVIDIA Research, AI-Mediated Digital Human



Where We Are

- Basic mathematical tools

- Color: definition, space, map, and design
- 2D visualization: data -> spatial, temporal, network

- Human perception and geometries of 3D
- 3D visualization: projection and interaction

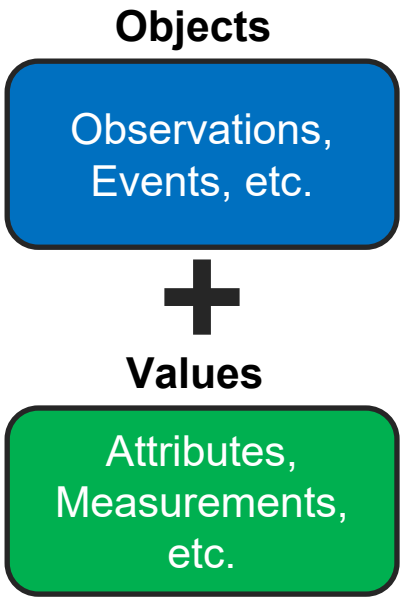
- Application case studies

Visualizing Network Data

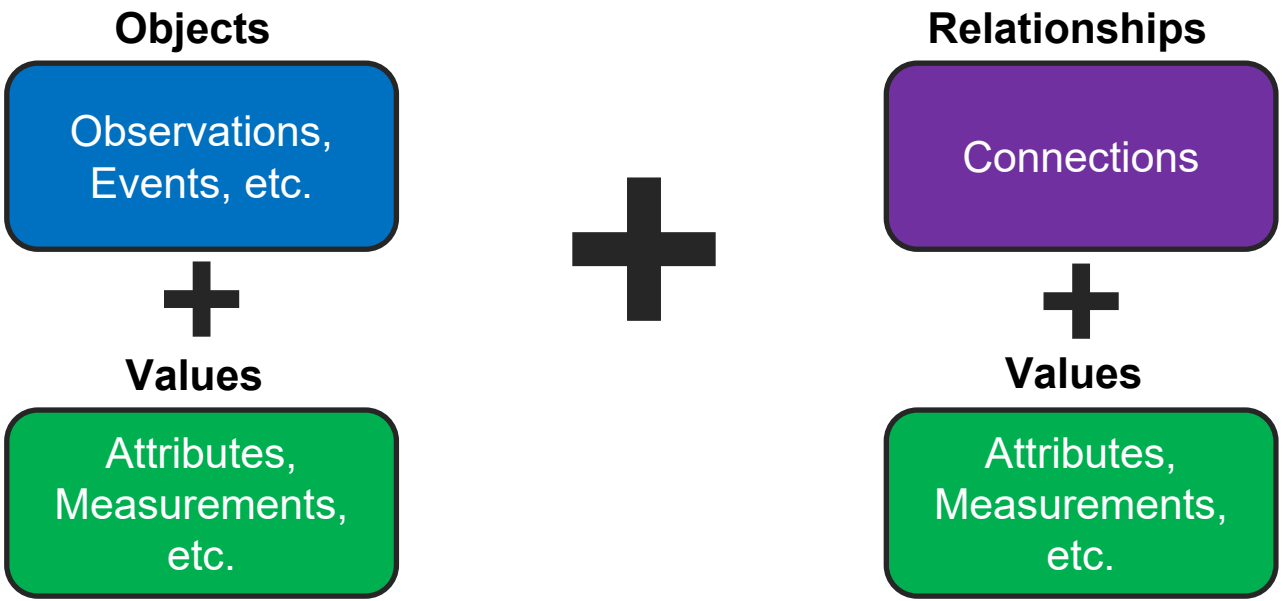
PART 01

What is **network** data?

Tabular Data



Network Data



Network Data

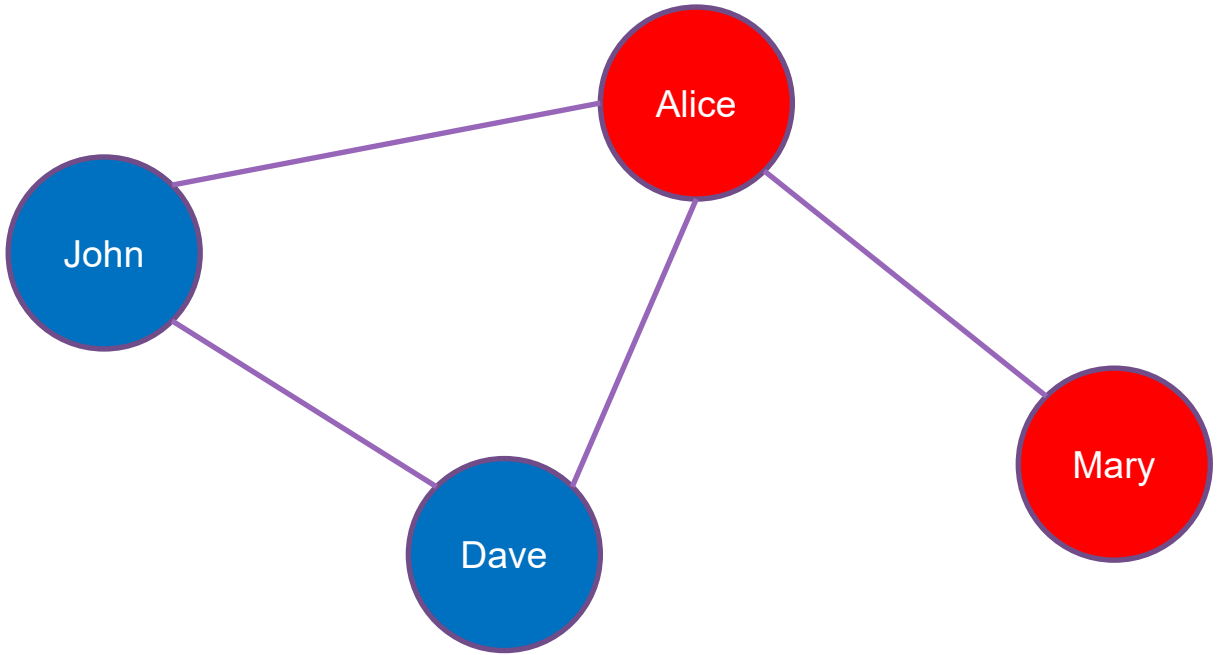
Nodes

Label	Value
A	5
B	10
C	15

Links

N1	N2	Value
A	B	9
A	C	27
B	C	18

Friendship Network

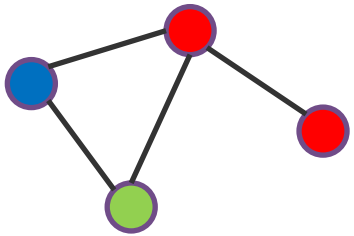


PART 01

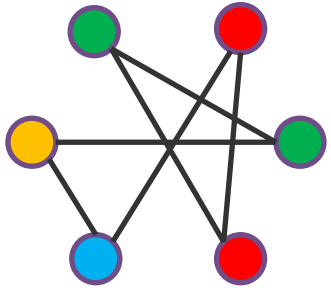
Network Visualization Techniques

Node-Link Diagram

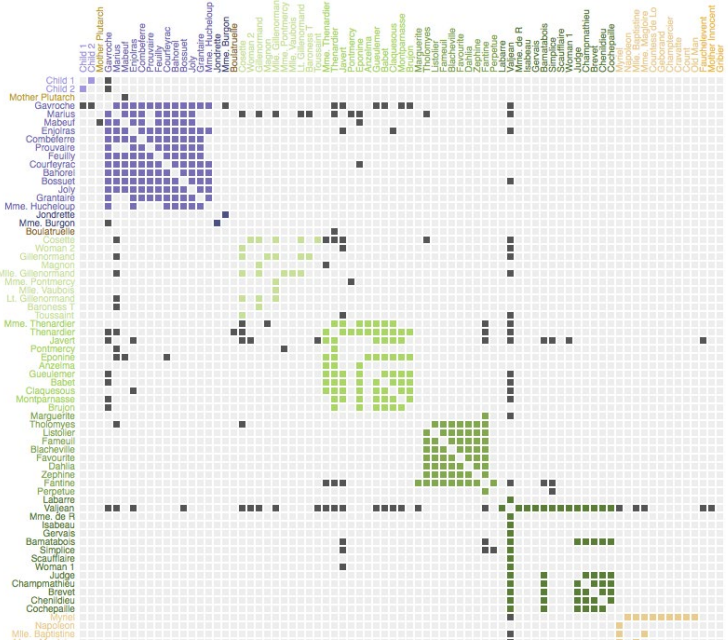
Force-Directed Layouts



Fixed Layouts



Matrices



PART 01

Node-Link Diagram

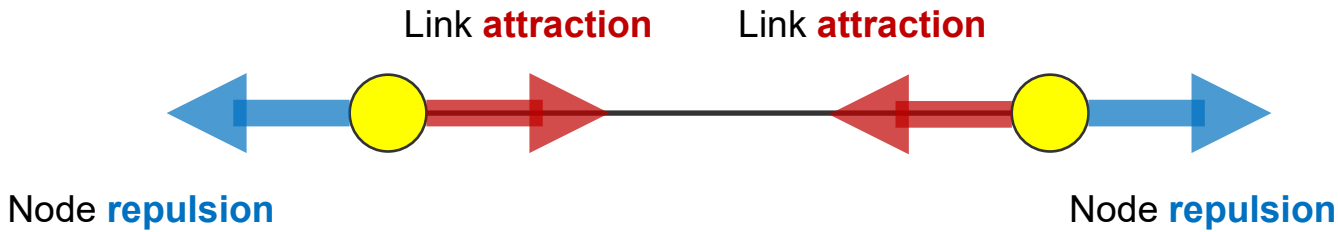
PART 01

Node-Link Diagram

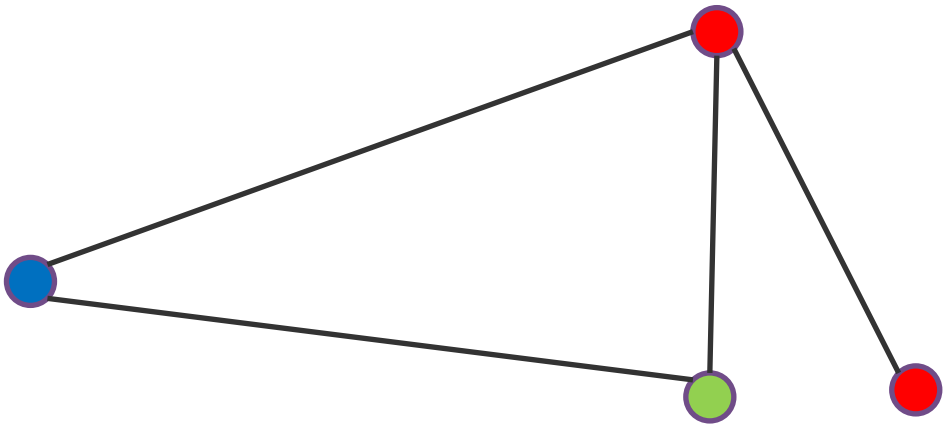
- **Nodes**: dots/markers.

- **Links**: lines connecting dots.

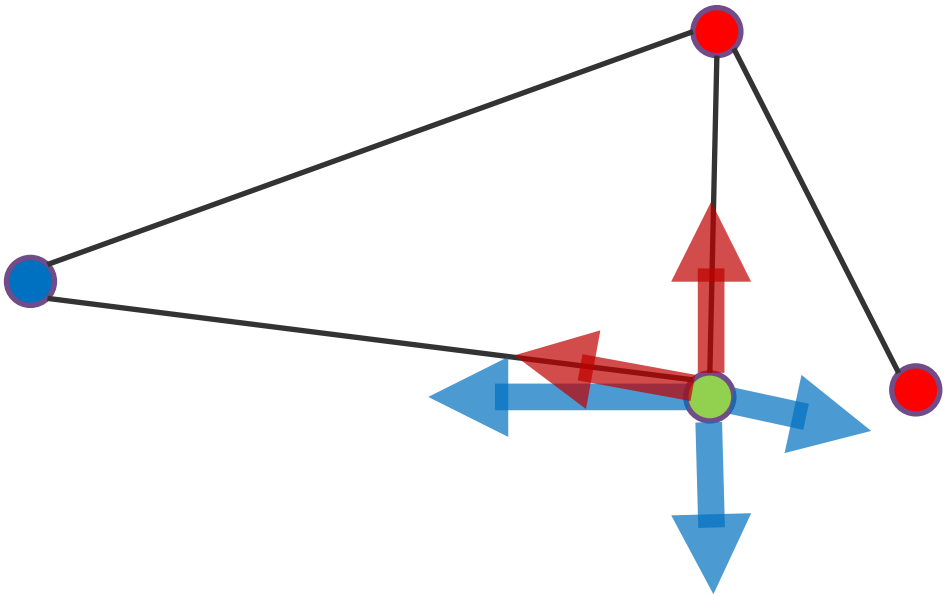
Force-Directed Layout



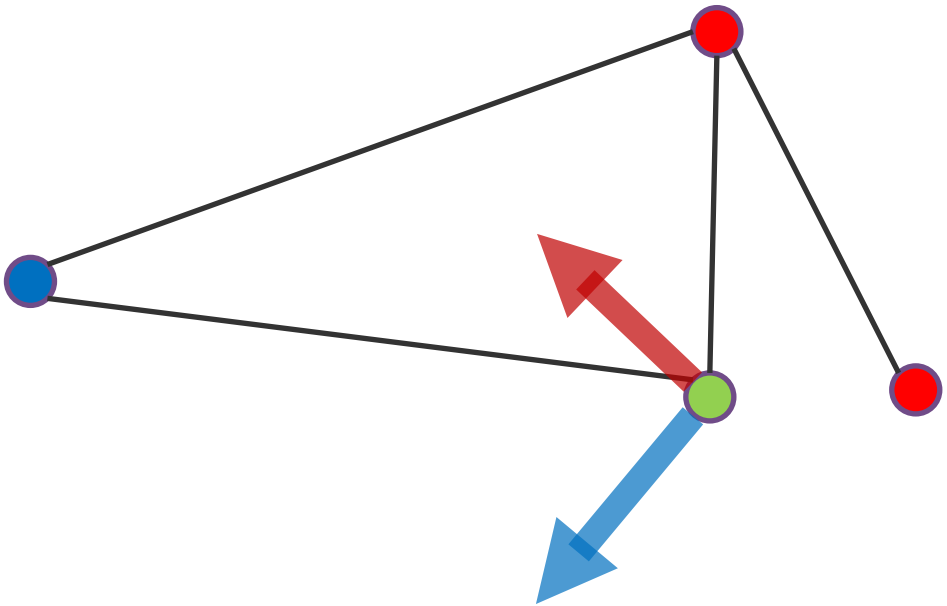
Initialize with random layout



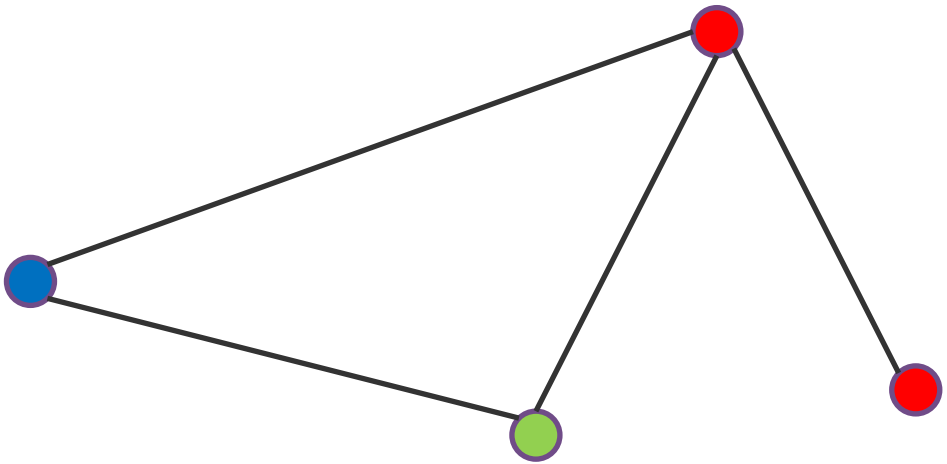
Compute attraction and repulson forces per node



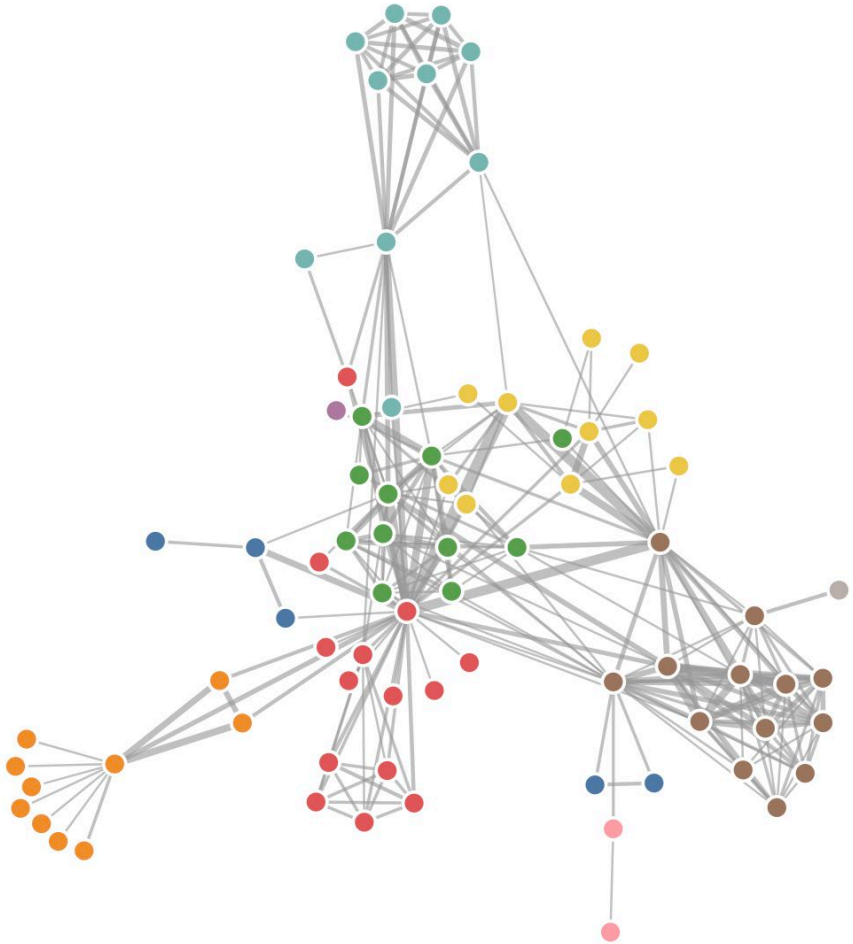
Compute the net force per node



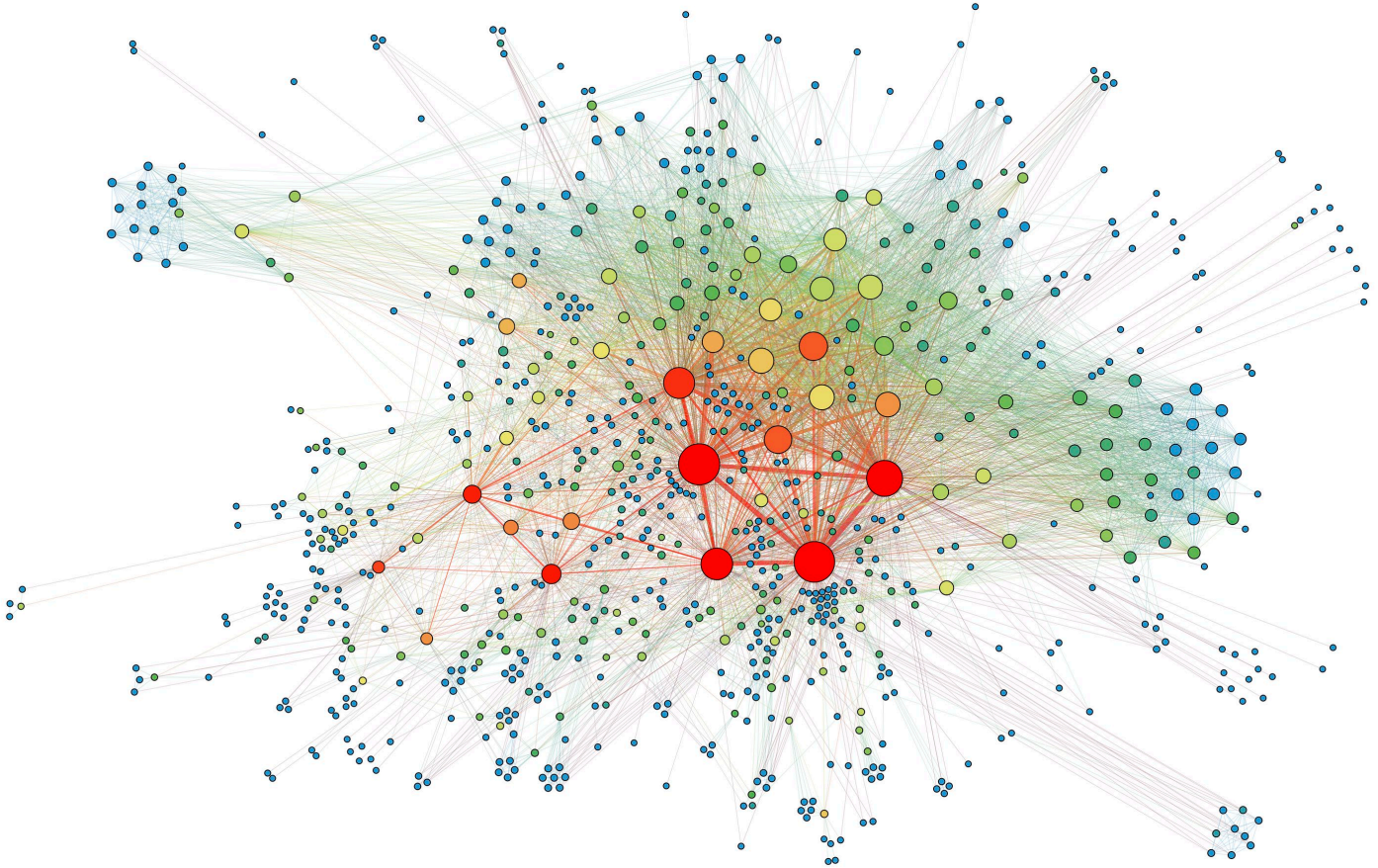
Move the node according to the net force



And repeat…

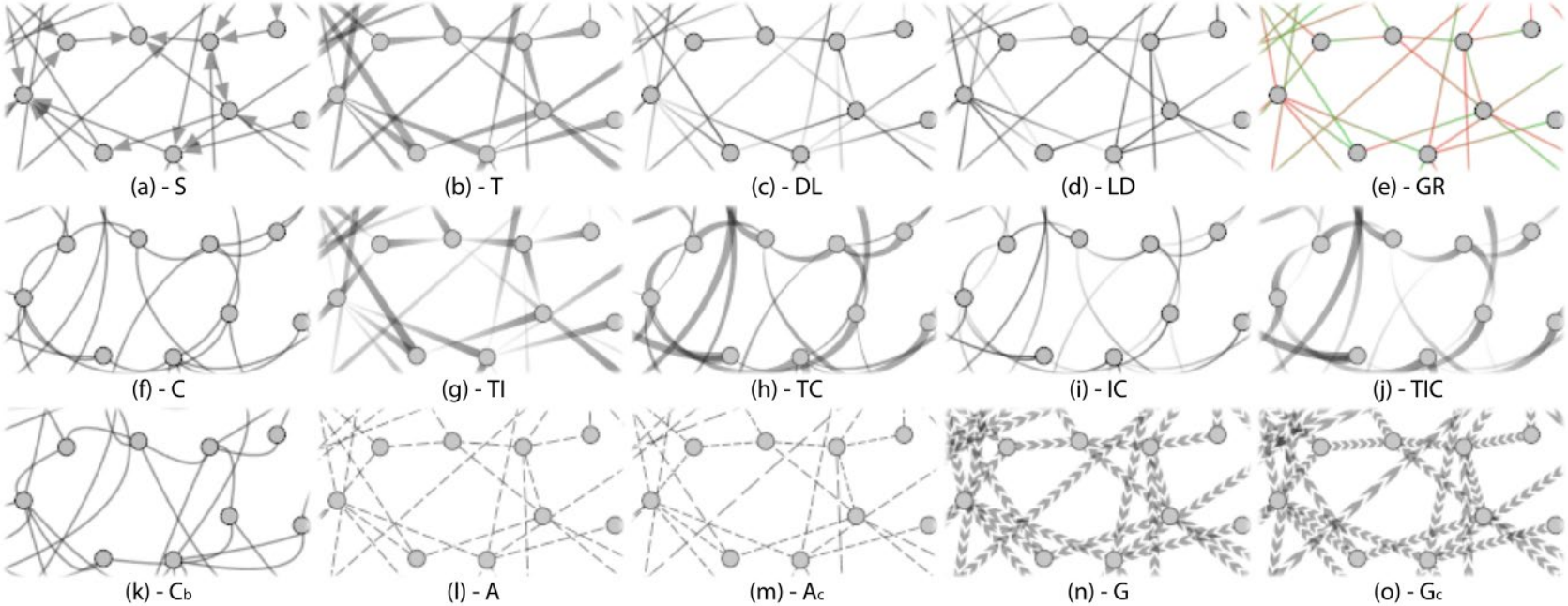


Source: https://observablehq.com/@d3/force-directed-graph



Source: https://en.wikipedia.org/wiki/Force-directed_graph_drawing#/media/File:SocialNetworkAnalysis.png

Directed Graphs

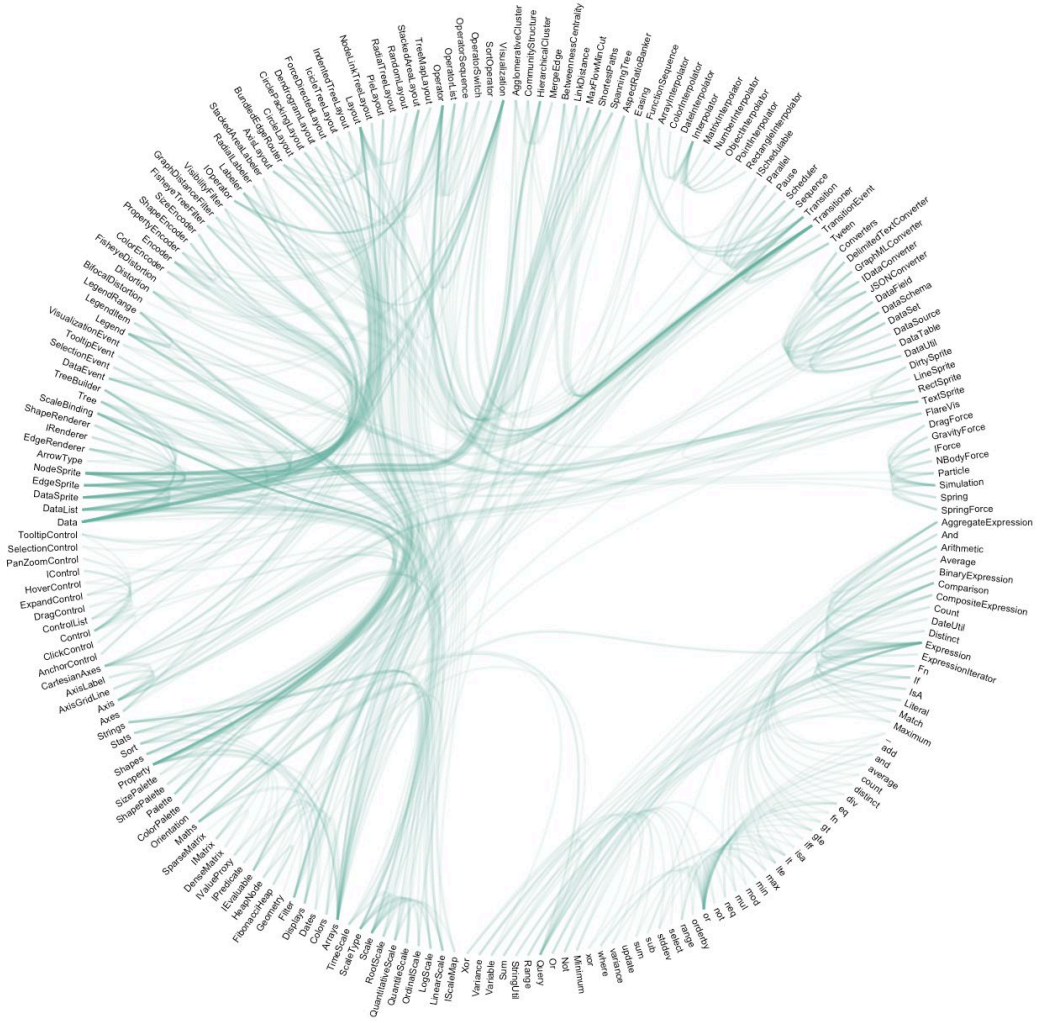


Holten, Danny, et al. "An extended evaluation of the readability of tapered, animated, and textured directed-edge representations in node-link graphs." *IEEE Pacific Visualization Symposium (PacificVis), 2011*.

PART 01

Fixed Layout

Positions of the nodes are fixed based on some criteria.



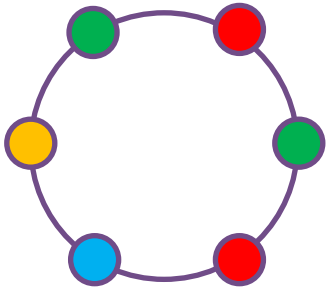
Source: https://www.data-to-viz.com/graph/edge_bundling.html

PART 01

Fixed Layout

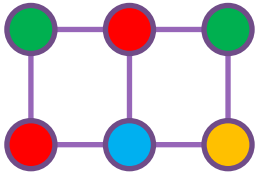
Linear

Circular



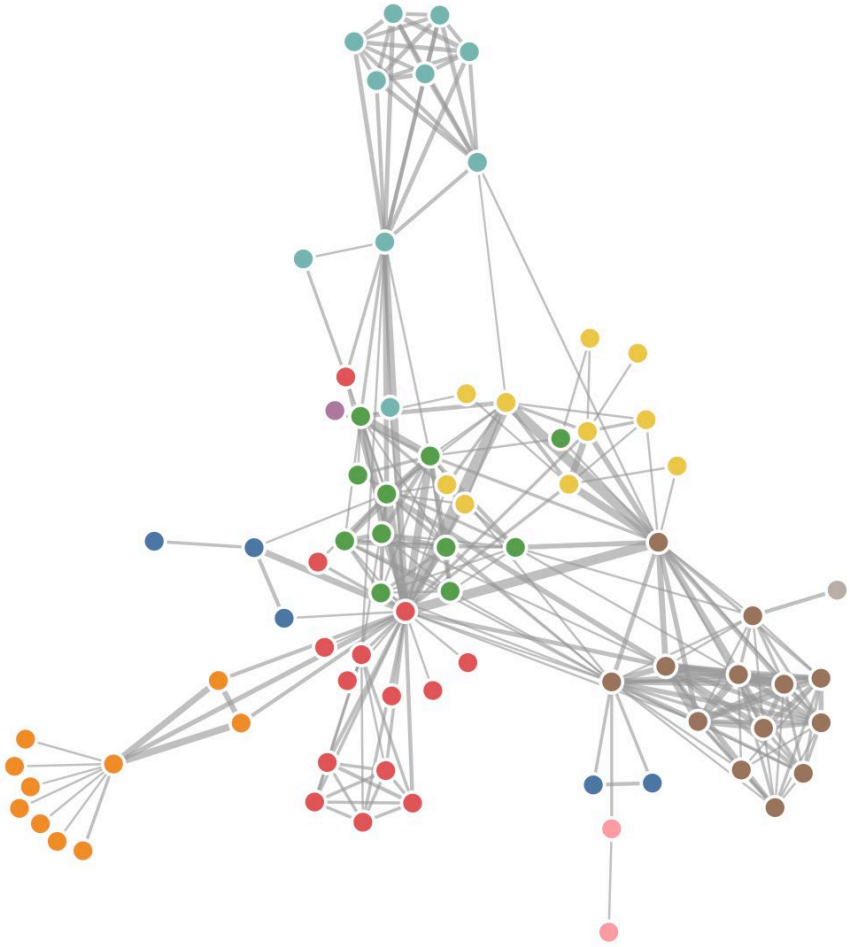
Most Common

Grid



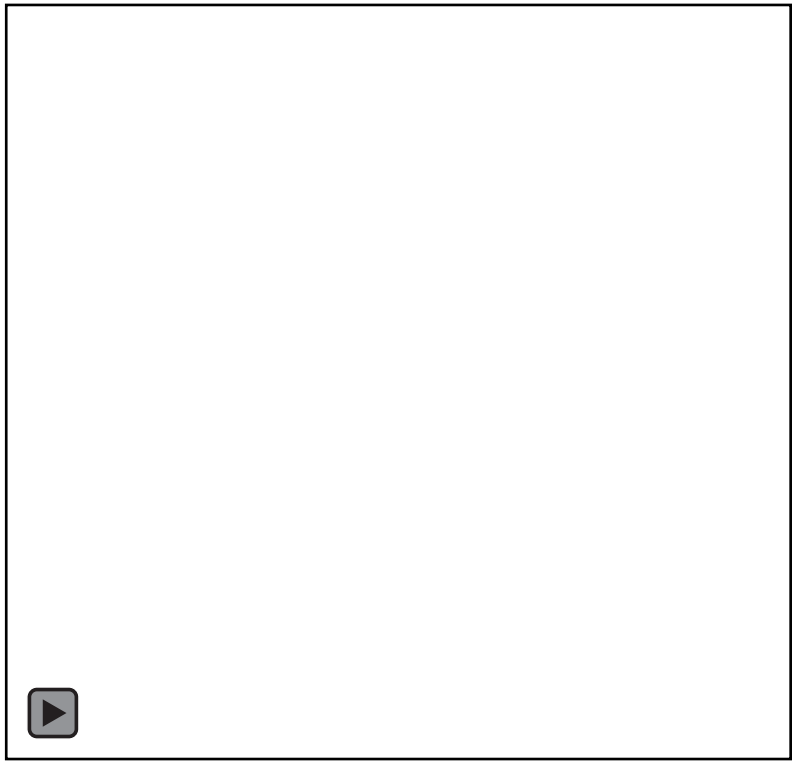
PART 01

When is fixed layout more appropriate than force-direct layout?



Force-directed layout reveals the **structure** of the network.

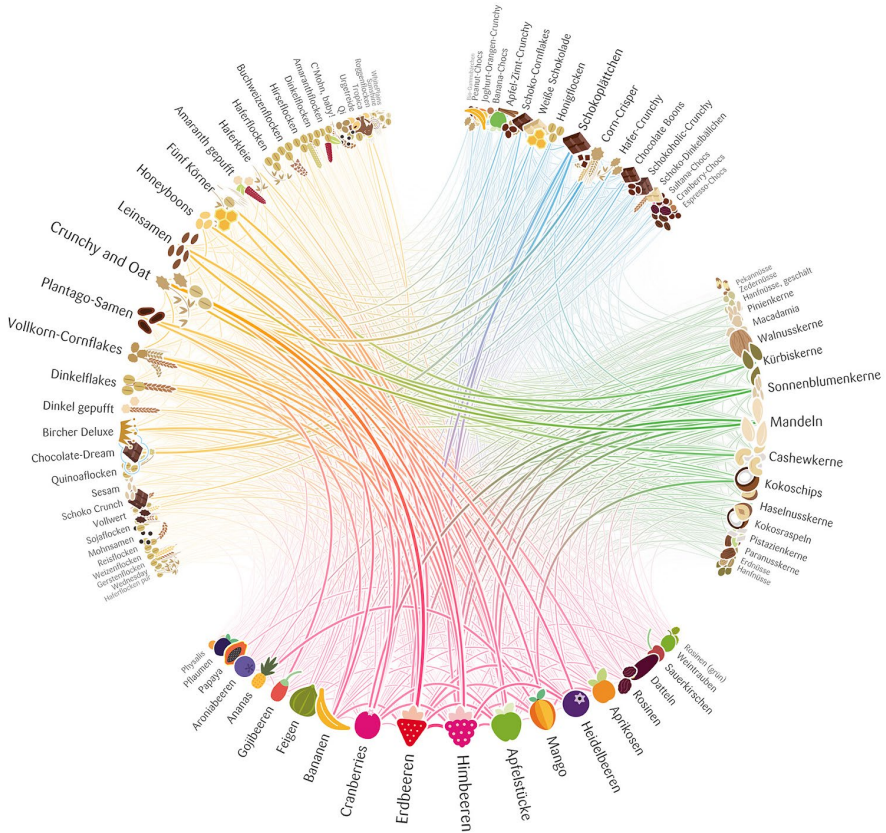
Source: https://observablehq.com/@d3/force-directed-graph



Advantages of fixed layout:

- Visibility of nodes.
- Visibility of edges.

Source: Enrico Bertini

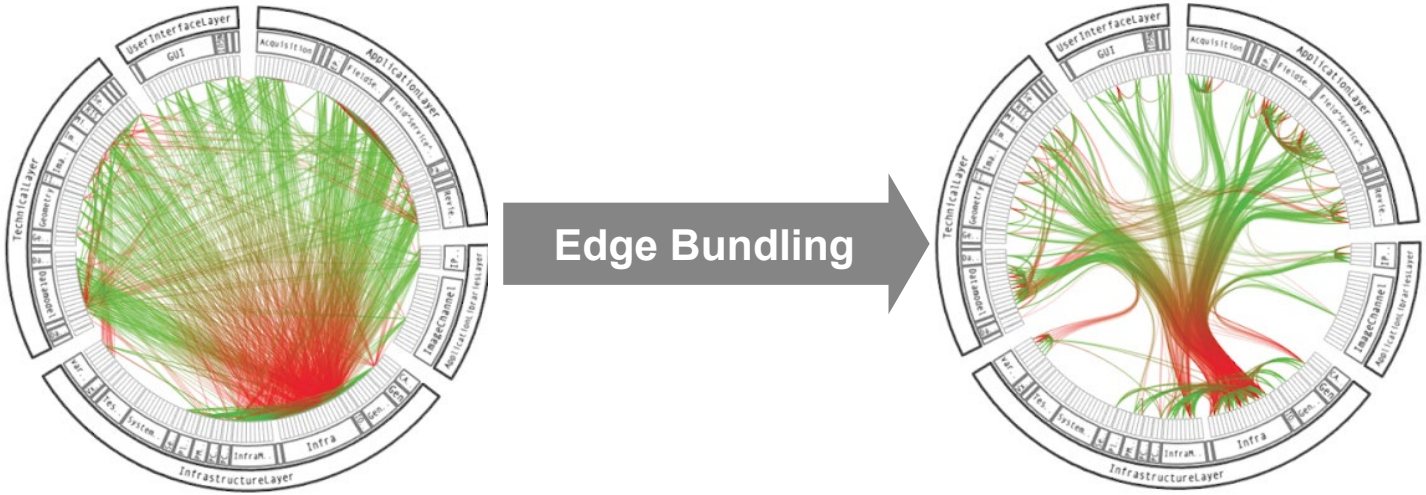


Advantages of fixed layout:

- Visibility of nodes.
- Visibility of edges.
- When nodes can be grouped into meaningful categories.

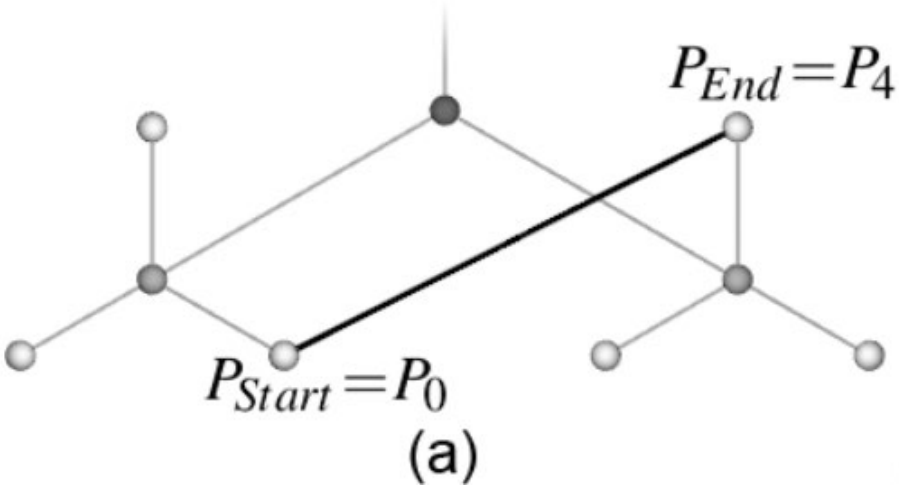
Source: https://truth-and-beauty.net/projects/muesli-ingredient-network

Avoid Overplotting



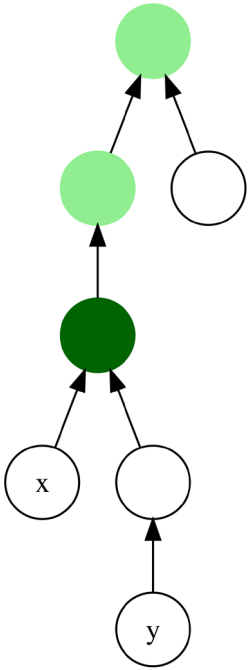
Holten, Danny. "Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data." IEEE Transactions on visualization and computer graphics 12.5 (2006): 741-748.

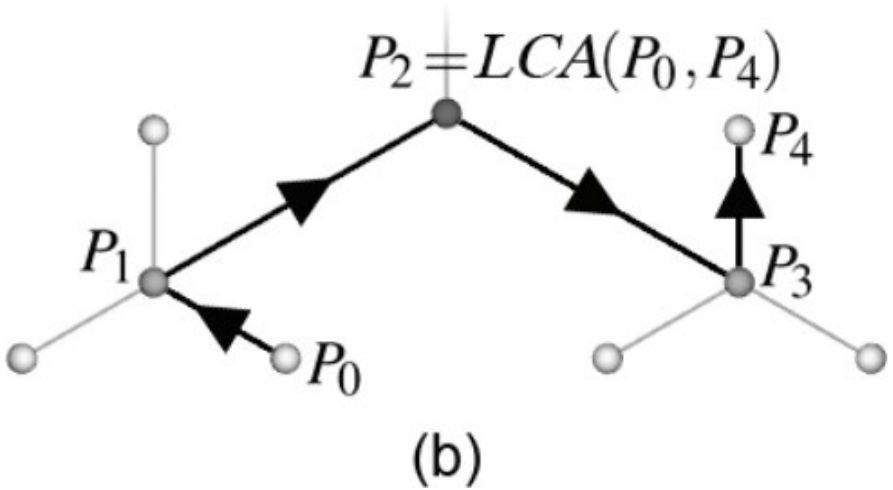
Edge Bundling



Holten, Danny. "Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data." IEEE Transactions on visualization and computer graphics 12.5 (2006): 741-748.

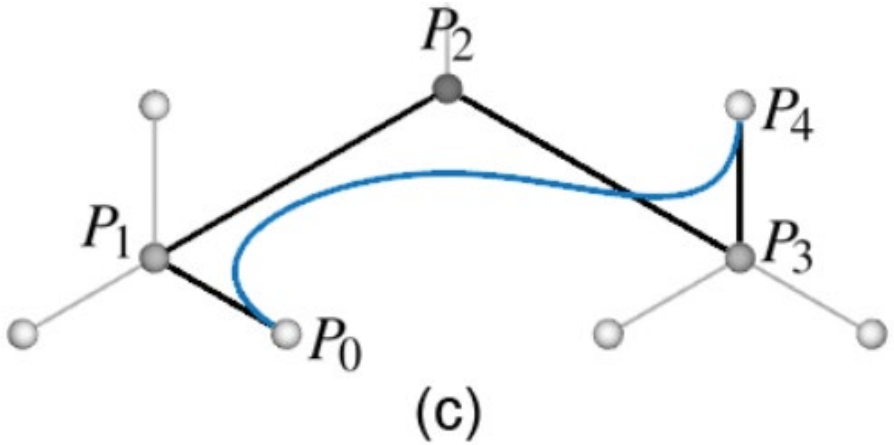
Edge Bundling - lowest common ancestor





Holten, Danny. "Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data." IEEE Transactions on visualization and computer graphics 12.5 (2006): 741-748.

Edge Bundling



Holten, Danny. "Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data." IEEE Transactions on visualization and computer graphics 12.5 (2006): 741-748.

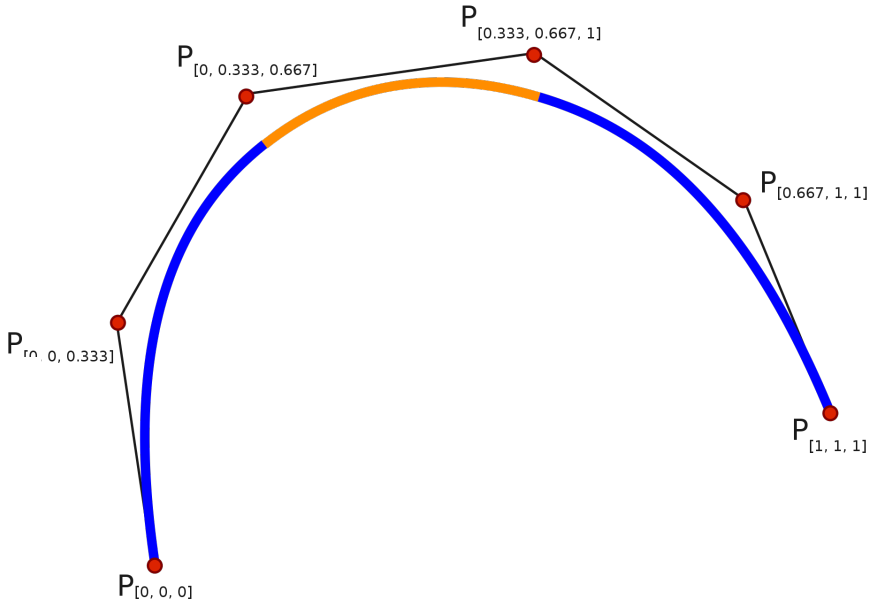
Spline

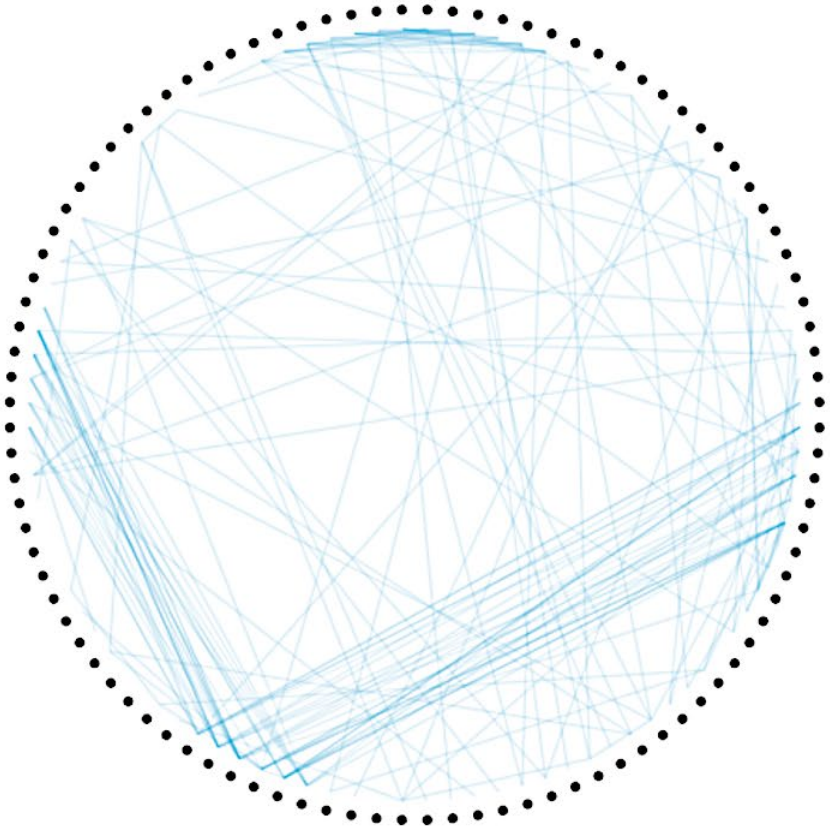
Example 1

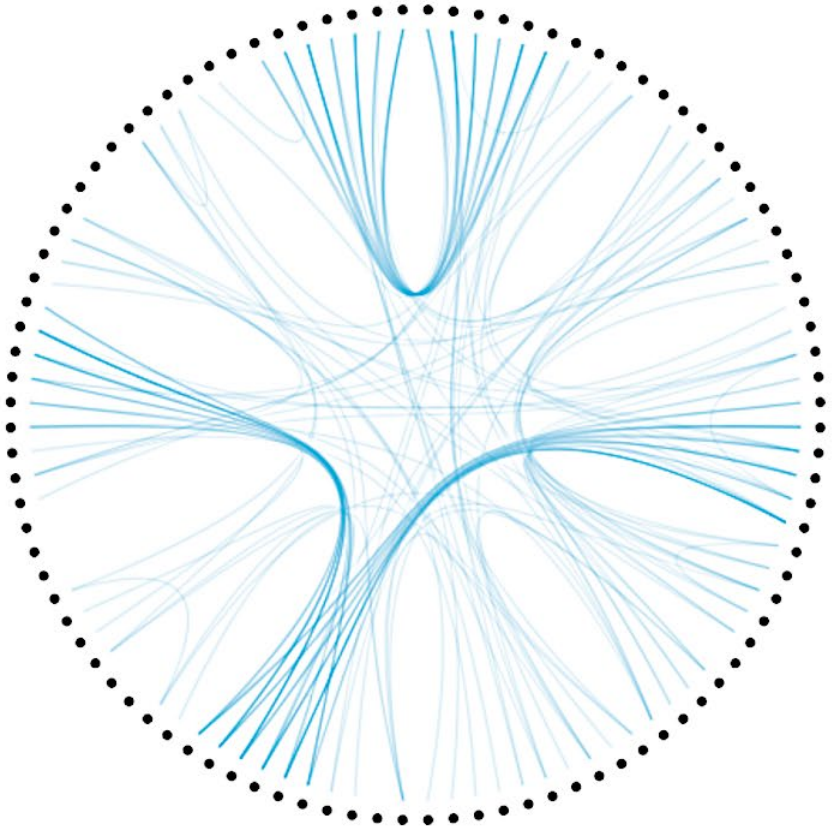
$$S(t) = P_0(t) = -1 + 4t - t^2 \ , 0 \leq t < 1$$
$$S(t) = P_1(t) = 2t \ , 1 \leq t < 2$$
$$S(t) = P_2(t) = 2 - t + t^2 \ , 2 \leq t \leq 3$$

Example 2

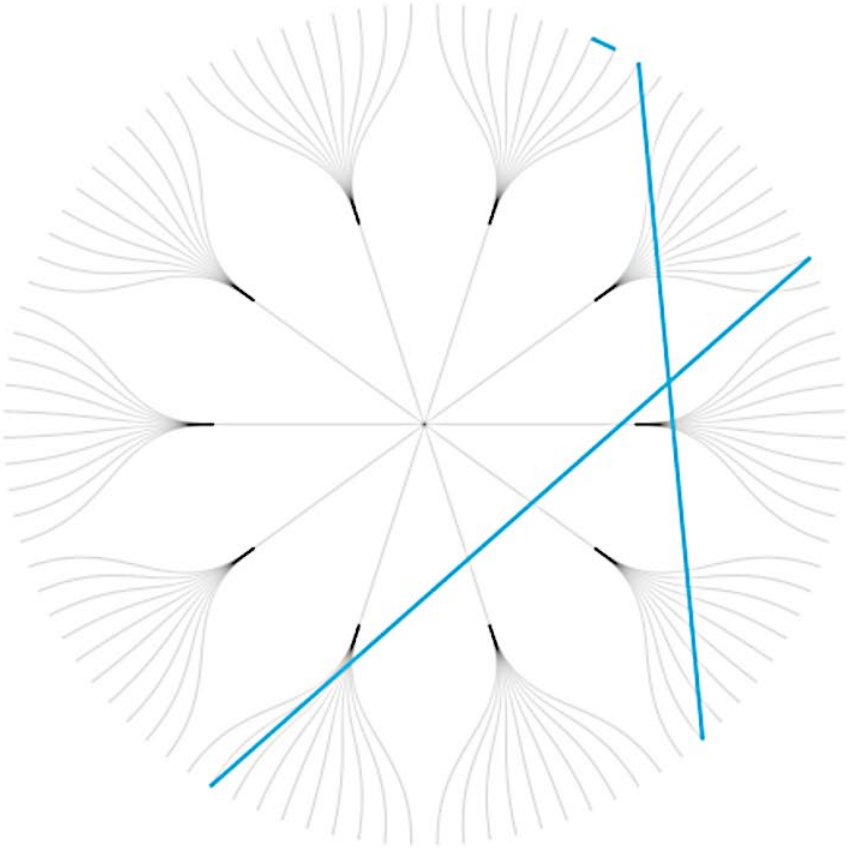
$$S(t) = P_0(t) = -2 - 2t^2 \ , 0 \leq t < 1$$
$$S(t) = P_1(t) = 1 - 6t + t^2 \ , 1 \leq t < 2$$
$$S(t) = P_2(t) = -1 + t - 2t^2 \ , 2 \leq t \leq 3$$

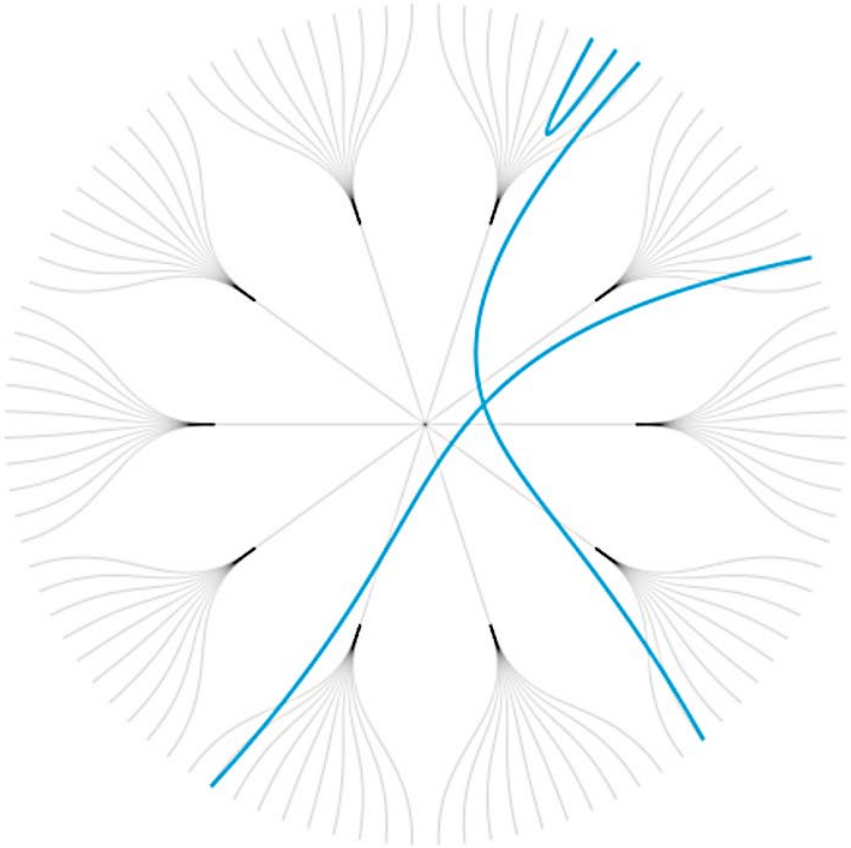




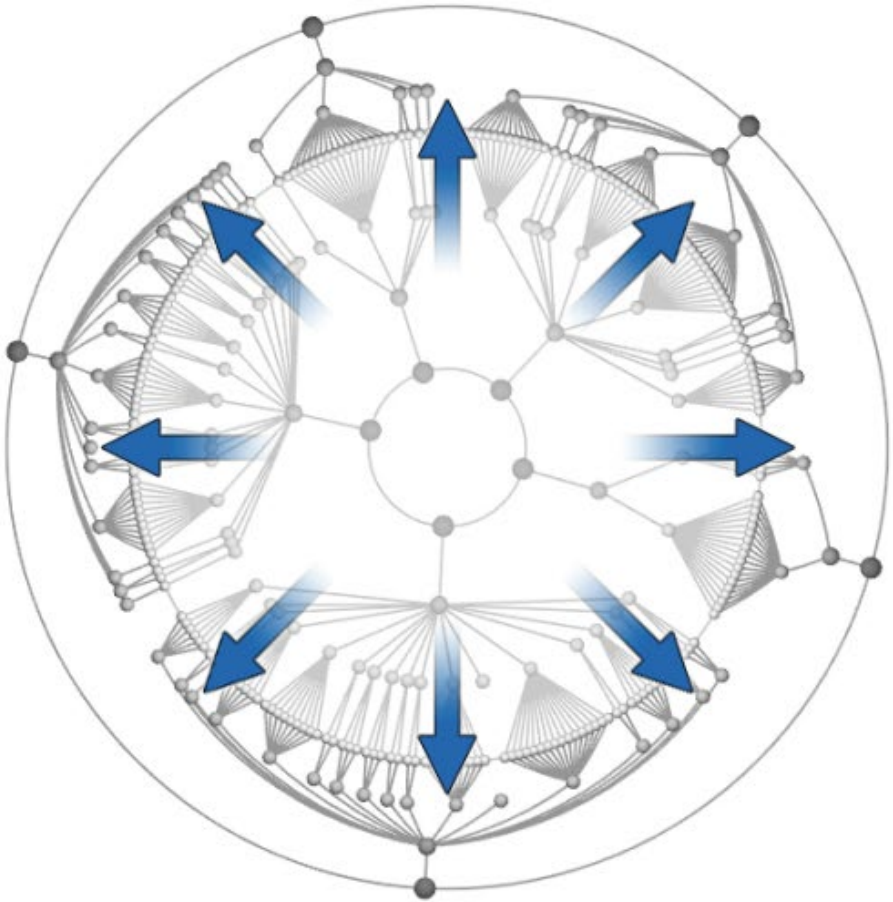


Source: https://www.r-graph-gallery.com/309-intro-to-hierarchical-edge-bundling/

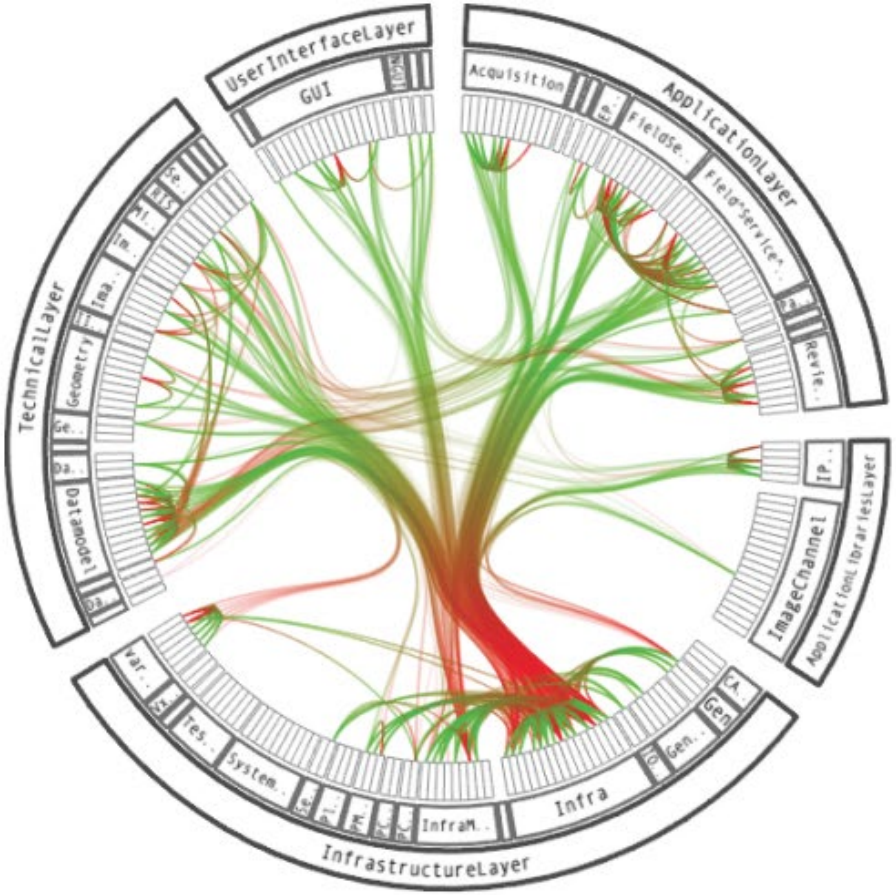




Source: https://www.r-graph-gallery.com/309-intro-to-hierarchical-edge-bundling/



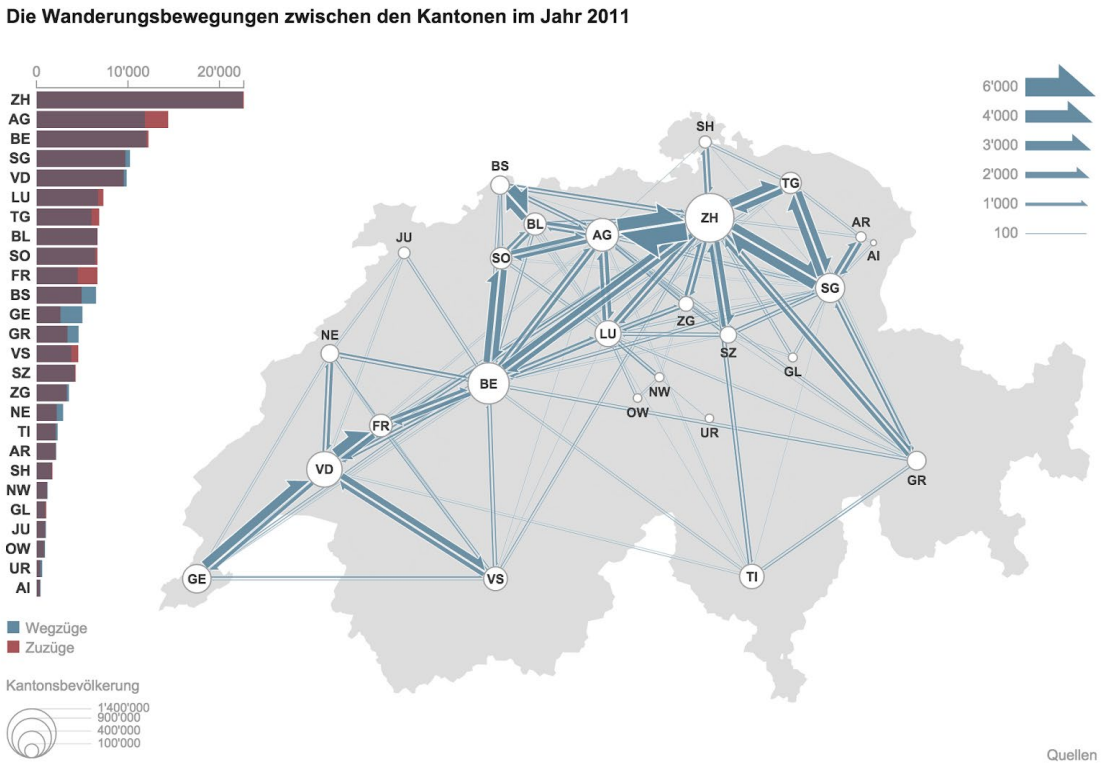
(a)


UserInterfaceLayer
GUI
ApplicationLayer
Acquisition
FieldService
TechnicalLayer
Geometry
DataModel
ImageChannel
InfrastructureLayer
Infra
System

(b)

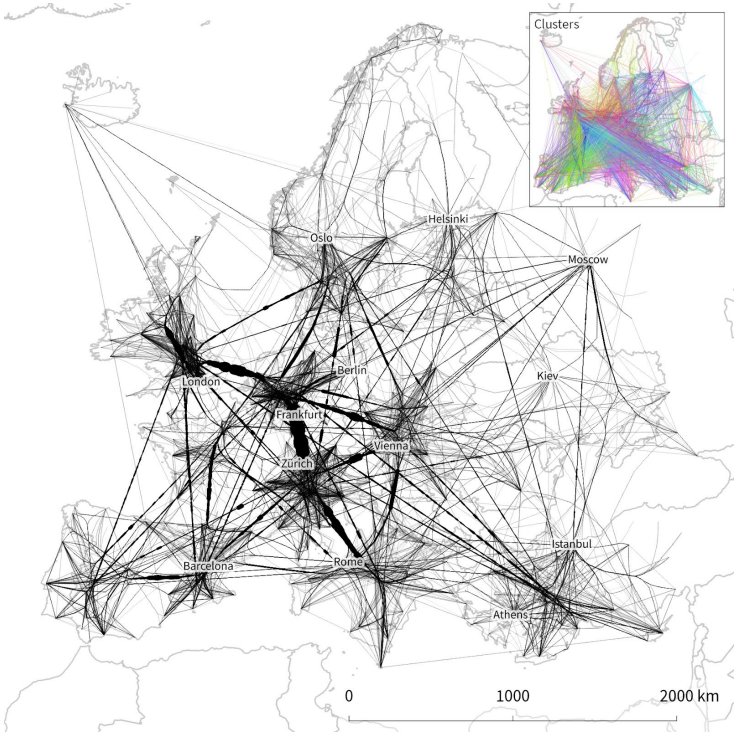
Fixed Layout: Spatial

- Nodes are placed based on geographical coordinates.

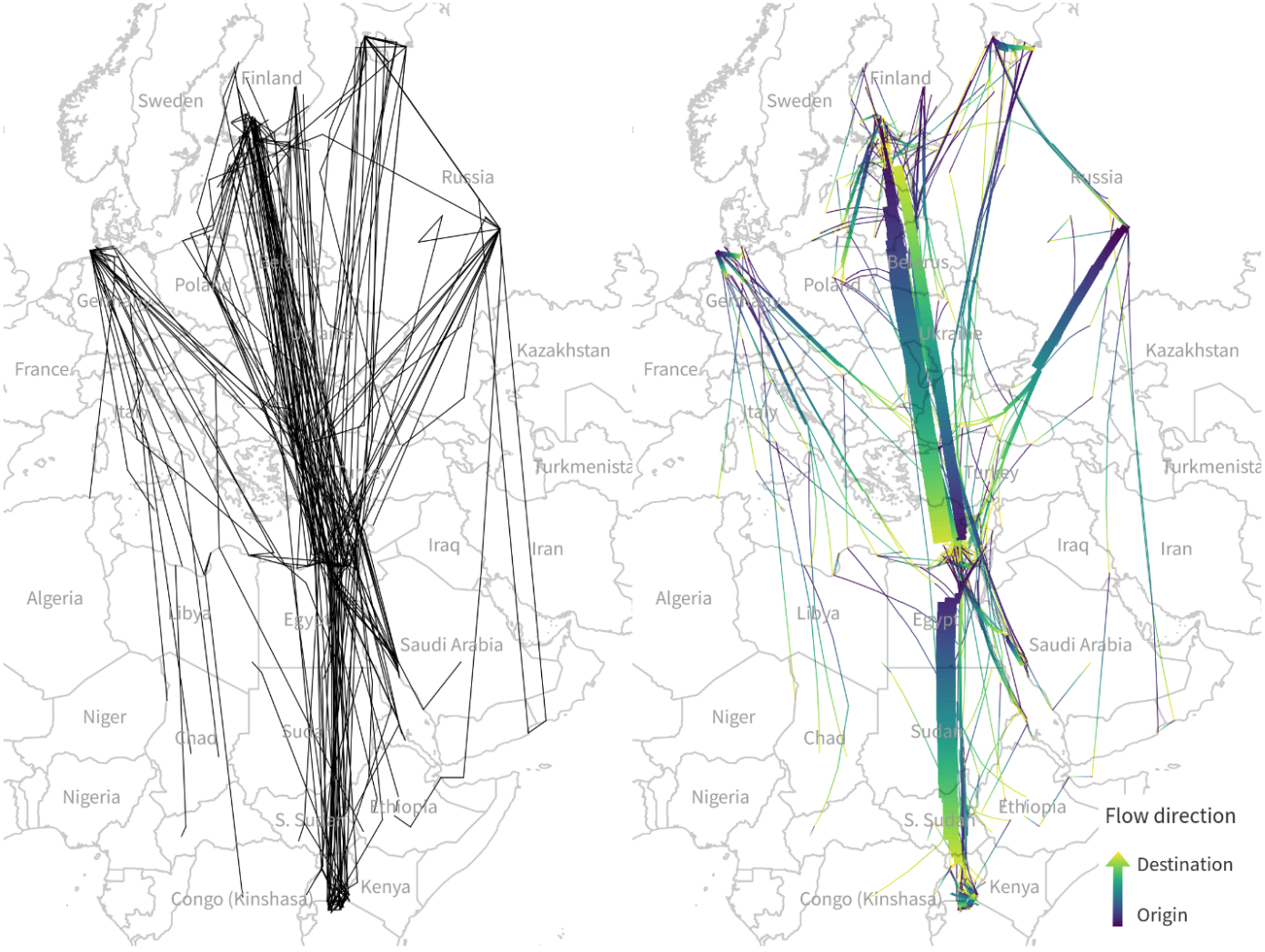


Source: https://ilya.boyandin.me/works/2013/08/30/summerseries

Problem: Clutter



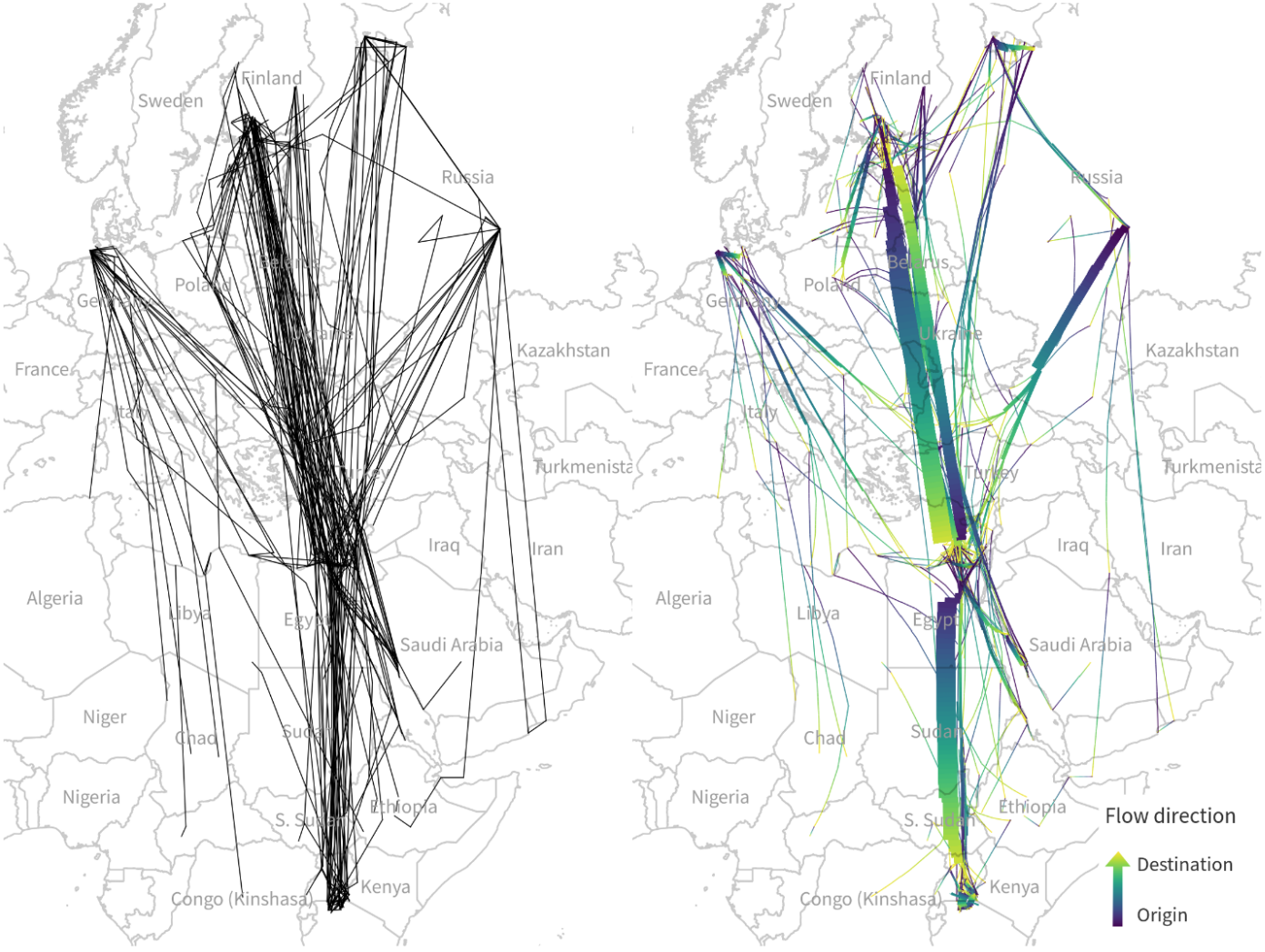
Source: https://github.com/dts-ait/qgis-edge-bundling


Sweden
Finland
Russia
Poland
France
Kazakhstan
Turkmenistan
Iraq
Iran
Algeria
Libya
Egypt
Saudi Arabia
Niger
Chad
Sudan
Nigeria
Ethiopia
S. Sudan
Kenya
Congo (Kinshasa)
Belarus
Ukraine
Italy
Turkey
Flow direction
Destination
Origin

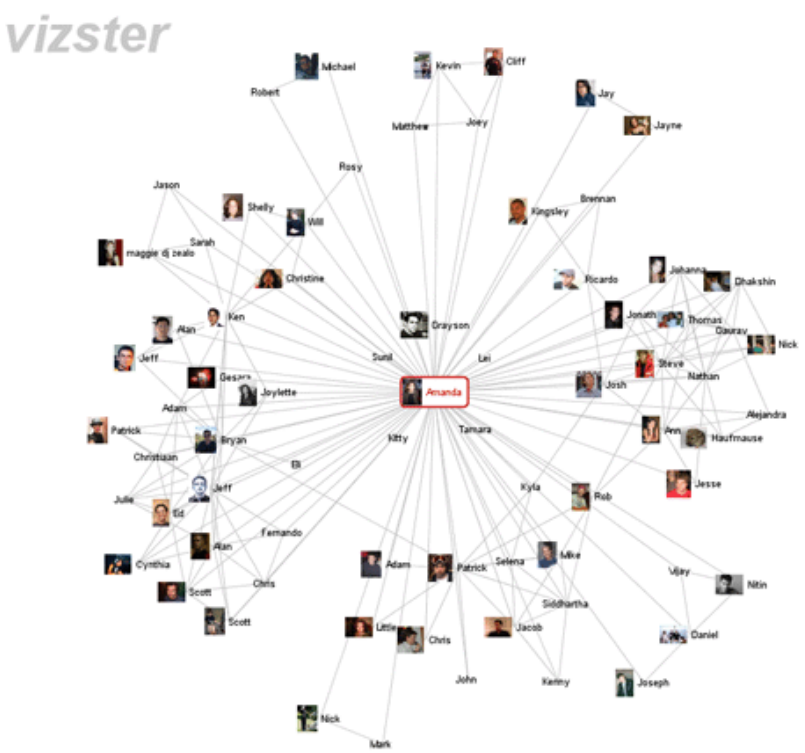
Clutter Reduction Methods

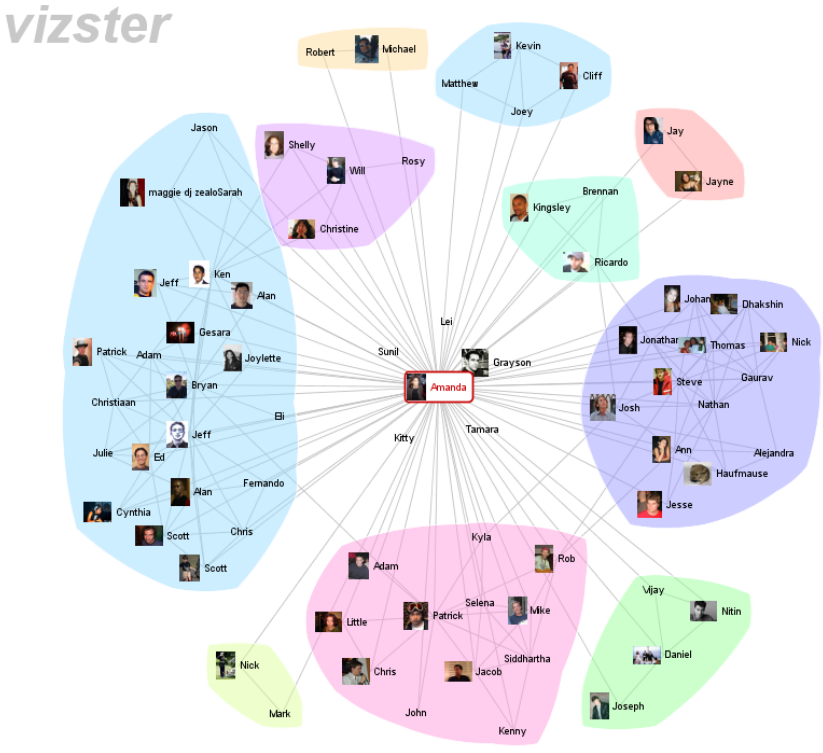
Clutter Reduction

- Edge Bundling.
- Clustering.
- Removing Edges.
- Draw edges on demand.
- Aggregation/Simplification.

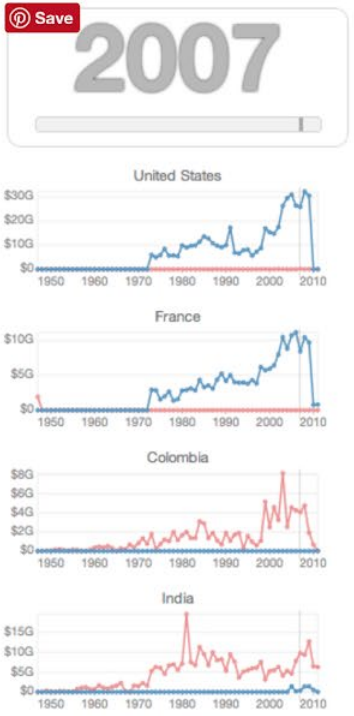

Sweden
Finland
Russia
Poland
France
Kazakhstan
Turkmenistan
Iraq
Iran
Algeria
Libya
Egypt
Saudi Arabia
Niger
Chad
Sudan
Nigeria
Ethiopia
S. Sudan
Kenya
Congo (Kinshasa)
Belarus
Ukraine
Italy
Turkey
Flow direction
Destination
Origin

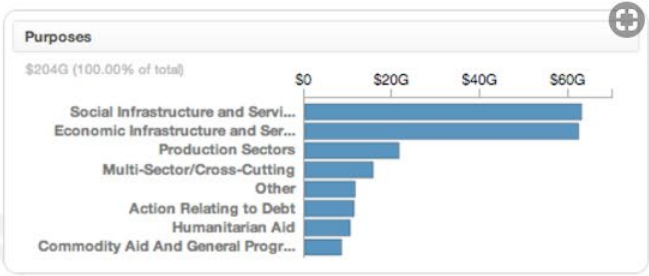
Clustering

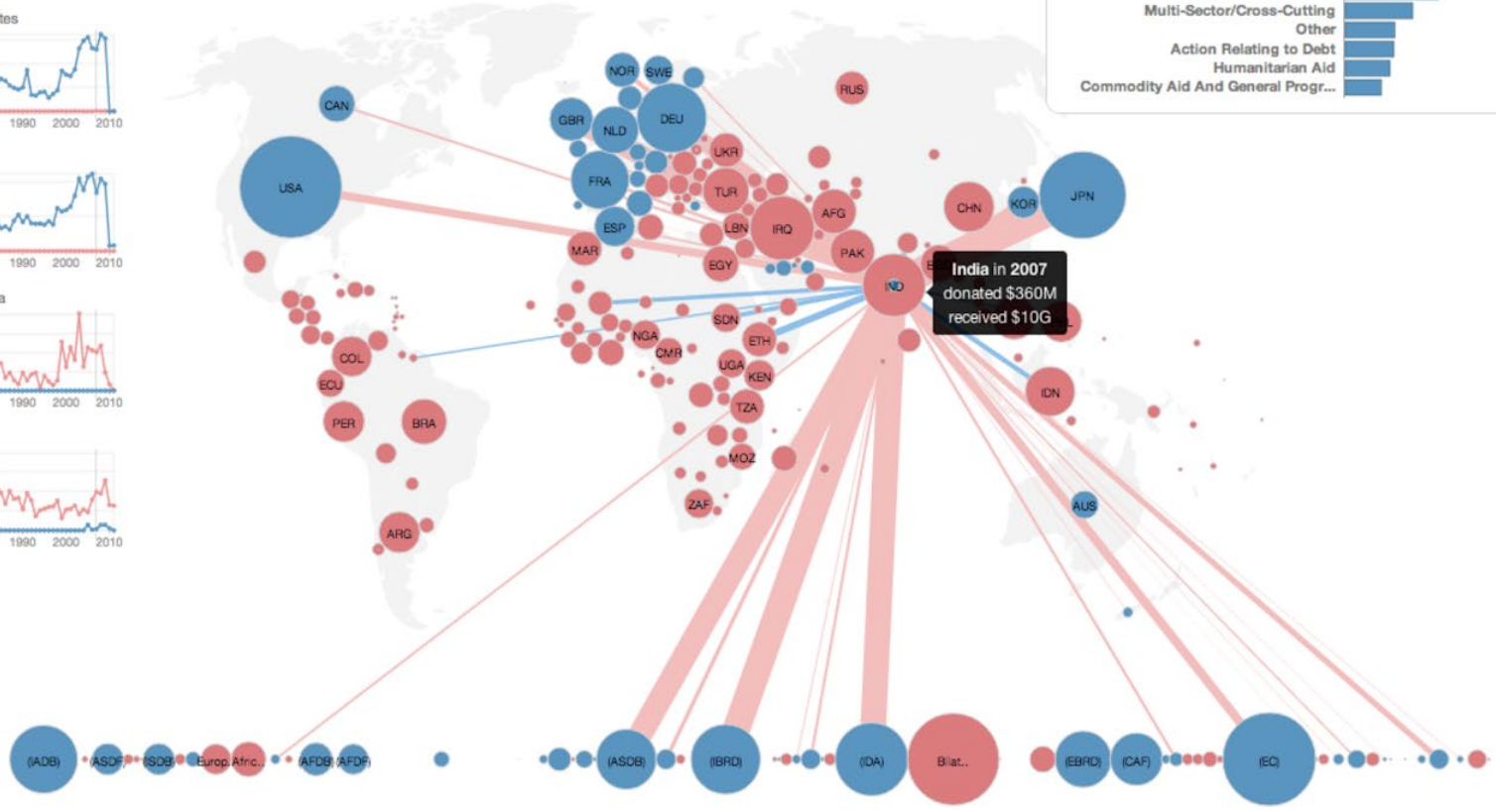

vizster
Michael
Kevin
Cliff
Robert
Jay
Matthew
Joey
Jayne
Rosy
Jason
Shelly
Will
Brennan
Kingsley
Sarah
maggie dj zealo
Christine
Ricardo
Johanna
Dhakshin
Ken
Jonath
Thomas
Gaurav
Grayson
Nick
Alan
Jeff
Sunil
Lei
Steve
Amanda
Nathan
Gesara
Joylette
Josh
Adam
Alejandra
Patrick
Bryan
Tamara
Ann
Haufmause
Kitty
Christiaan
Eli
Jesse
Julie
Jeff
Kyla
Rob
Ed
Fernando
Alan
Cynthia
Chris
Selena
Mike
Adam
Patrick
Vijay
Nitin
Scott
Siddhartha
Little
Chris
Jacob
Daniel
John
Kenny
Joseph
Nick
Mark


vizster
Robert
Michael
Kevin
Matthew
Cliff
Joey
Jason
Shelly
Rosy
Jay
Will
Jayne
maggie dj zealoSarah
Brennan
Kingsley
Christine
Ricardo
Jeff
Ken
Alan
Johan
Dhakshin
Lei
Gesara
Jonathar
Thomas
Nick
Patrick
Adam
Joylette
Sunil
Grayson
Steve
Gaurav
Bryan
Amanda
Christiaan
Nathan
Josh
Eli
Jeff
Ann
Alejandra
Julie
Ed
Tamara
Kitty
Haufmause
Alan
Fernando
Jesse
Cynthia
Chris
Kyla
Scott
Scott
Adam
Rob
Vijay
Nitin
Little
Patrick
Selena
Mike
Daniel
Nick
Chris
Siddhartha
Jacob
Joseph
Mark
John
Kenny

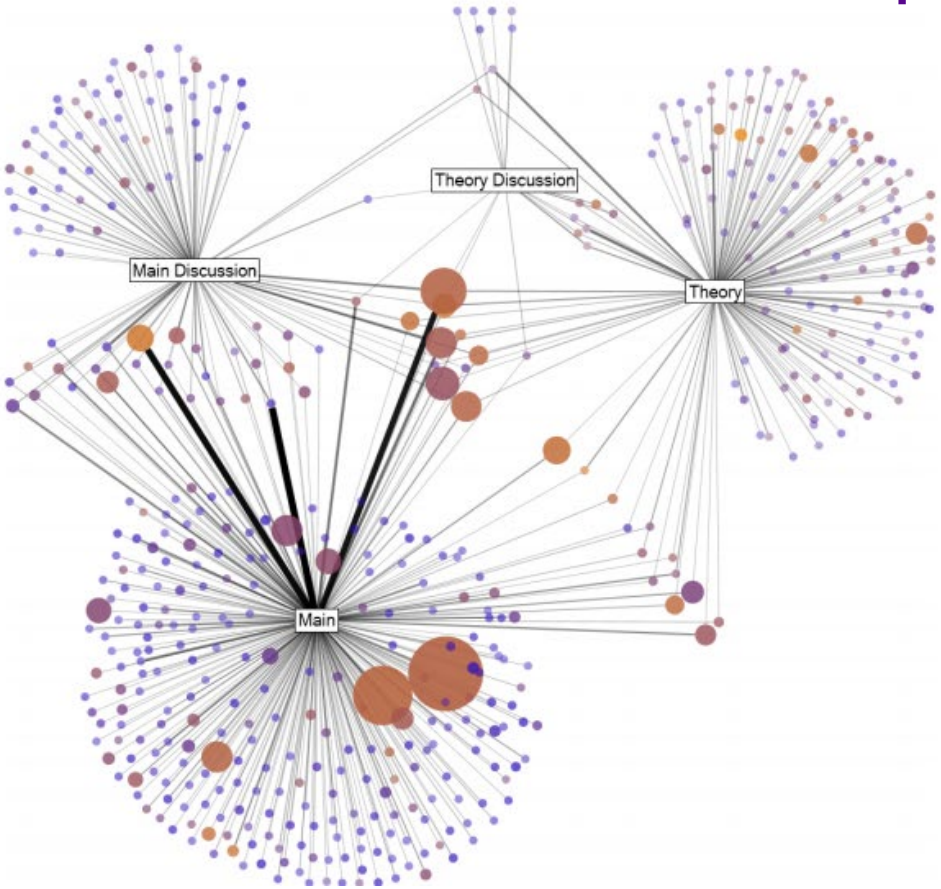
Drawing edge on demand

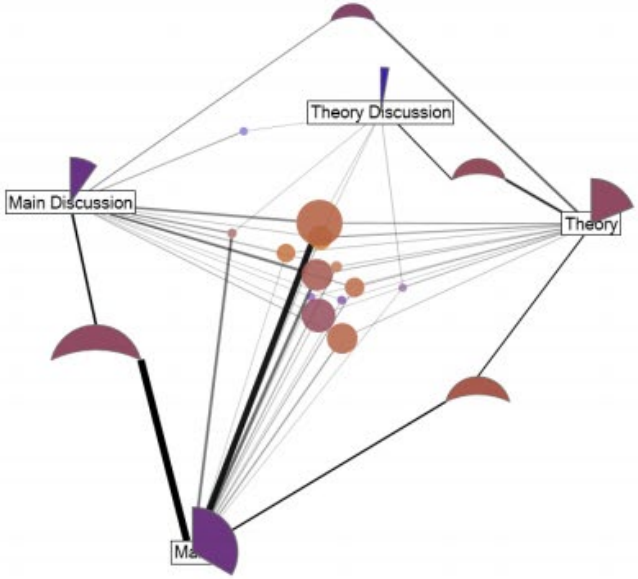






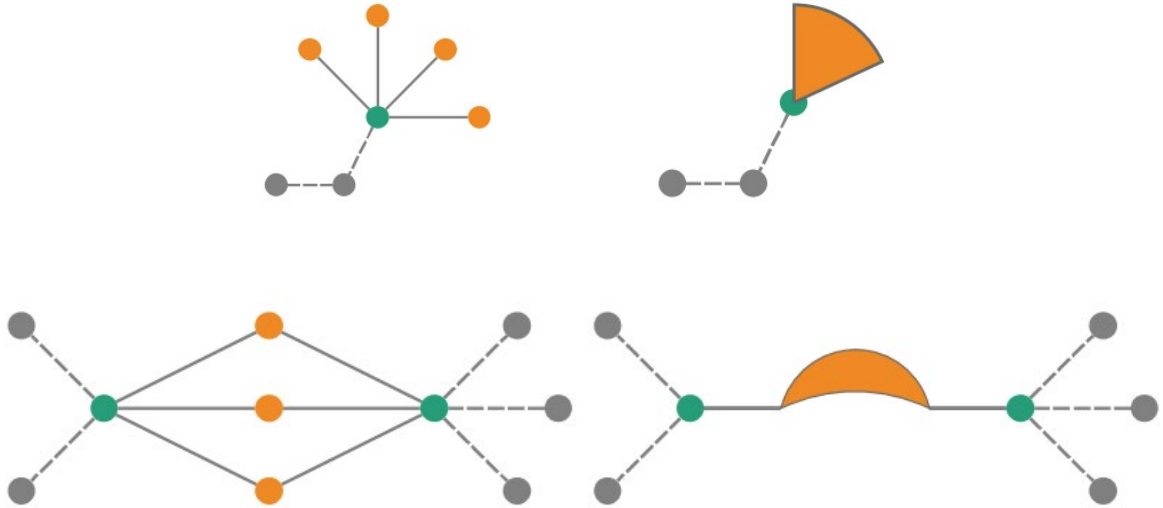
Motif Simplification



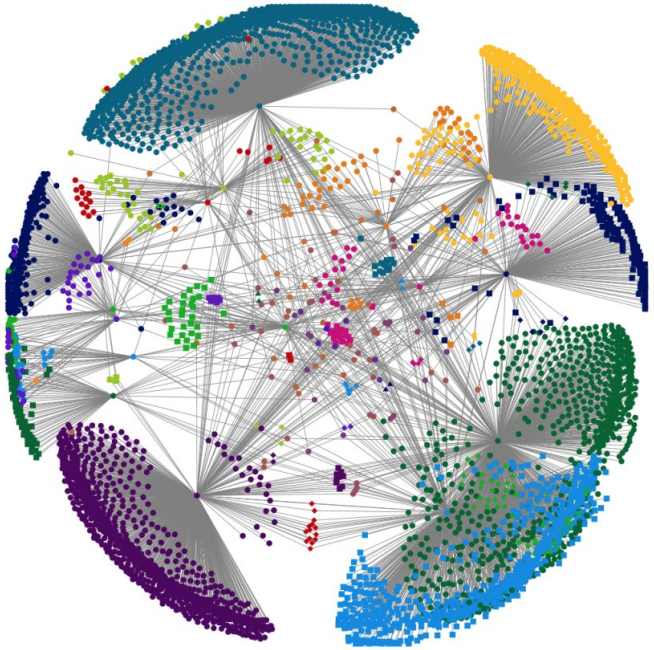


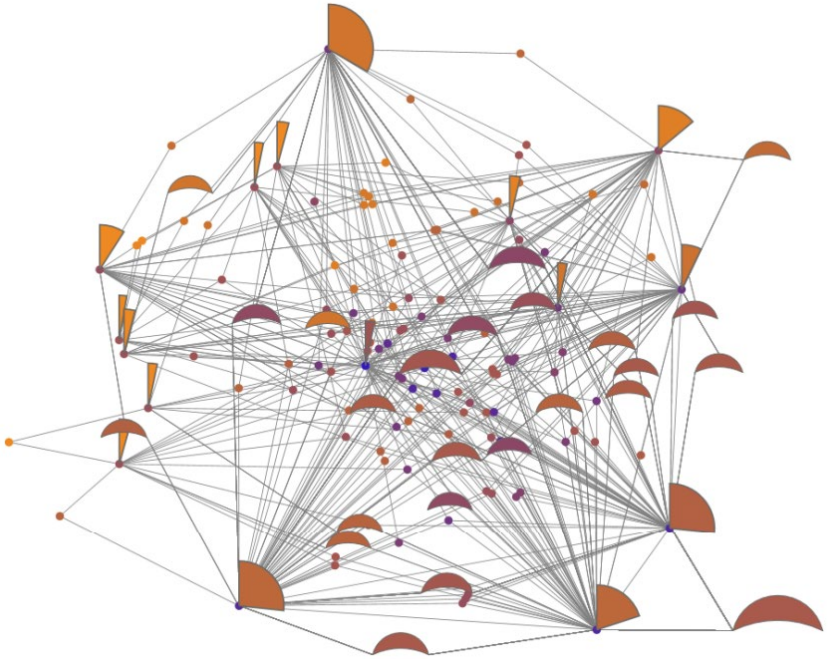
Dunne, Cody, and Ben Shneiderman. "Motif simplification: improving network visualization readability with fan, connector, and clique glyphs." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013.

Motif Simplification



Motif Simplification





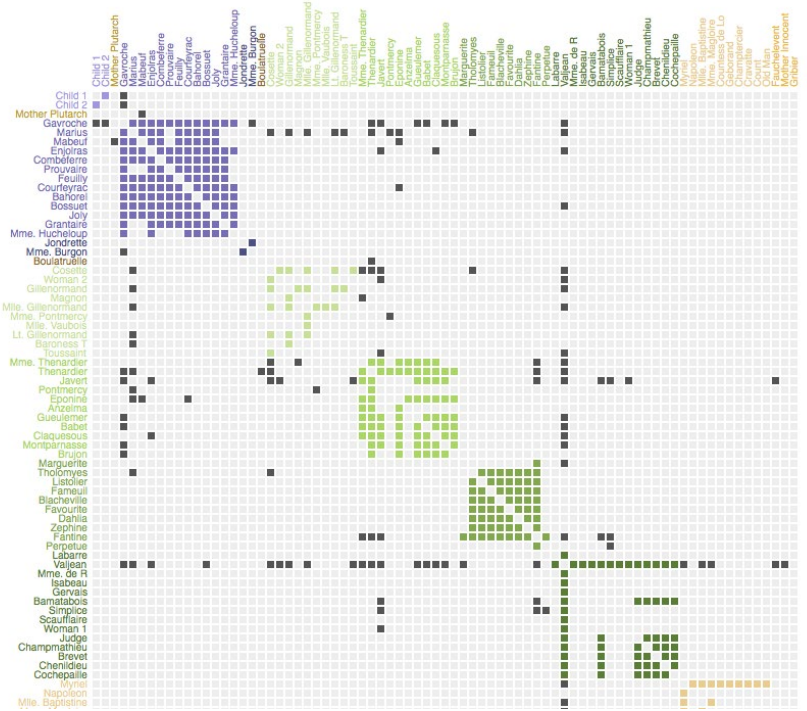
PART 01

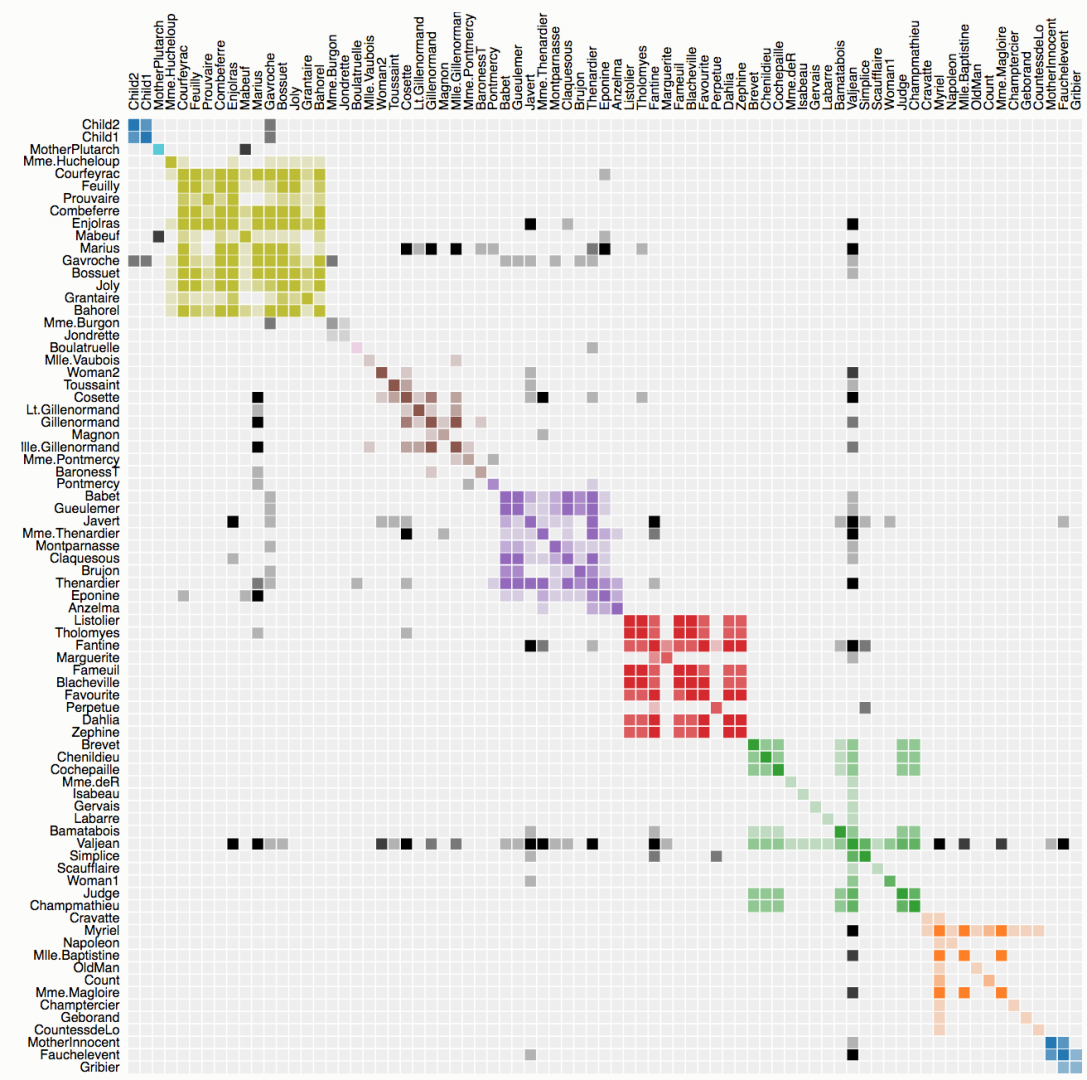
Matrices

PART 01

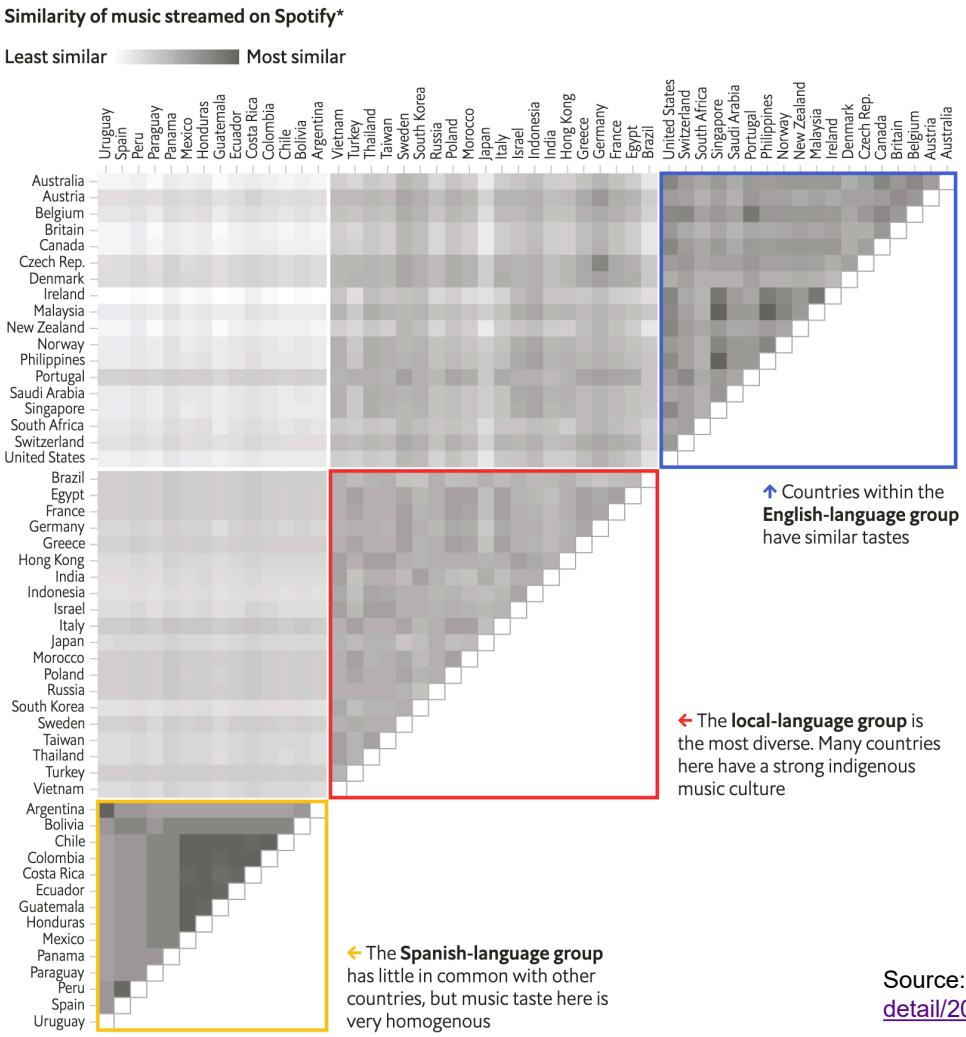
Adjacency Matrix

- Nodes become rows and cols.
- Edge become cells.





Character Co-occurrence in “Les Misérables”

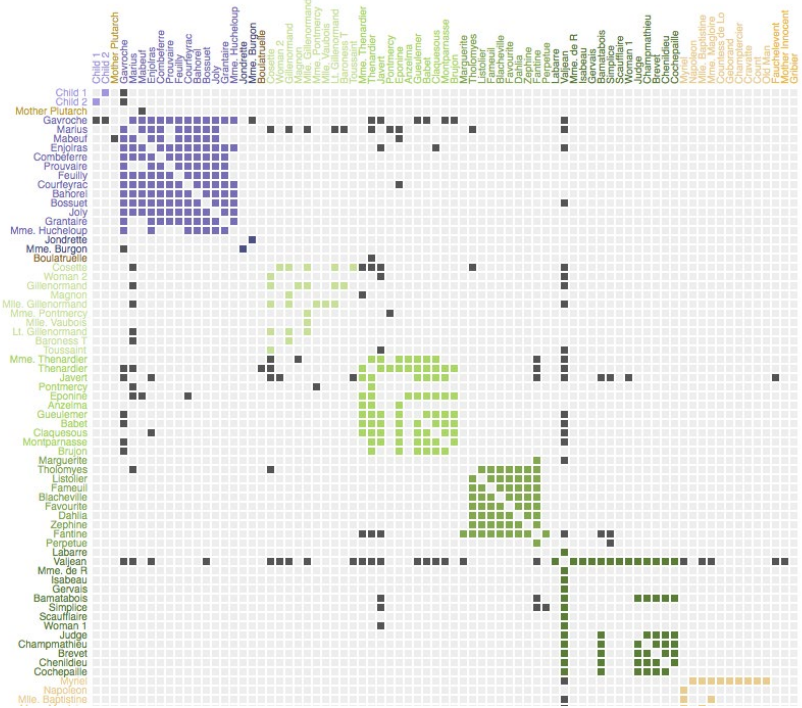


*From a model comparing top tracks, artists, genres and languages between January 1st-November 12th 2021

Source: https://www.economist.com/interactive/graphic-detail/2022/01/29/what-spotify-data-show-about-the-decline-of-english

PART 01

When to Use Matrix?

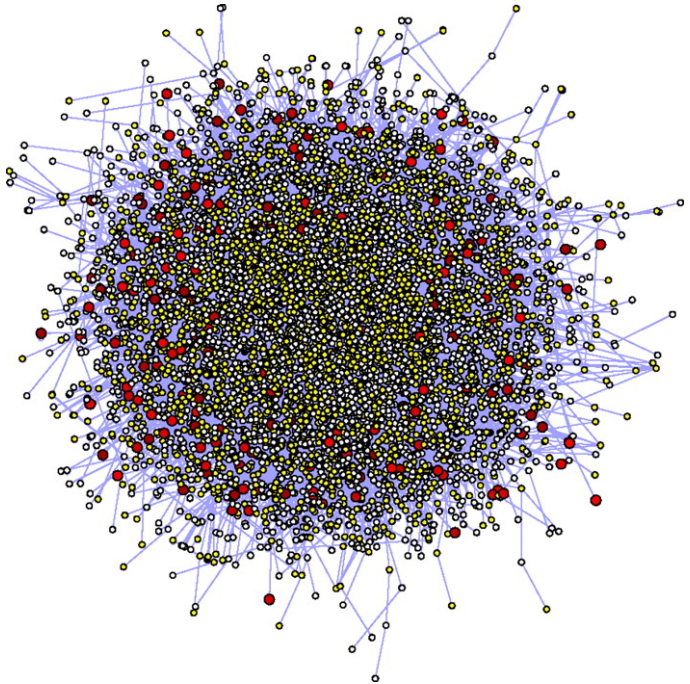


Advantages:

- Node visibility. No line crossing.

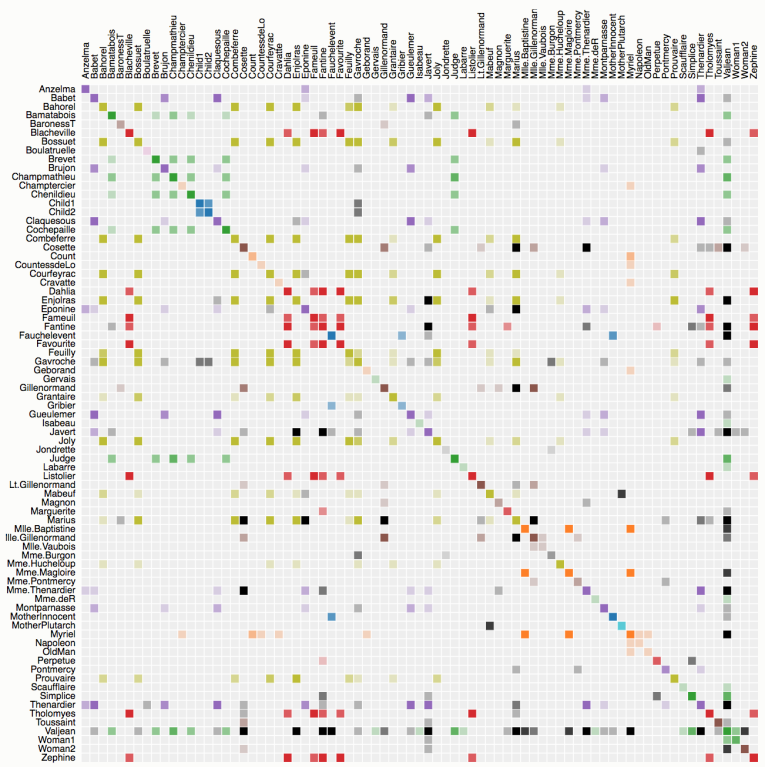
Disadvantages:

- Familiarity, scalability, needs reordering.



Biggest **advantage** of matrix:

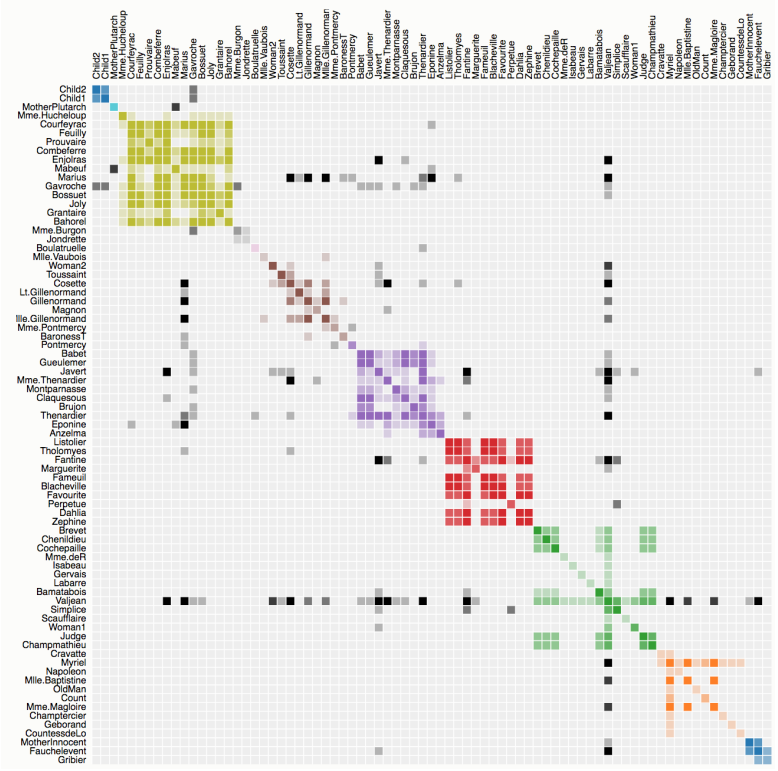
Avoids hairball effect!



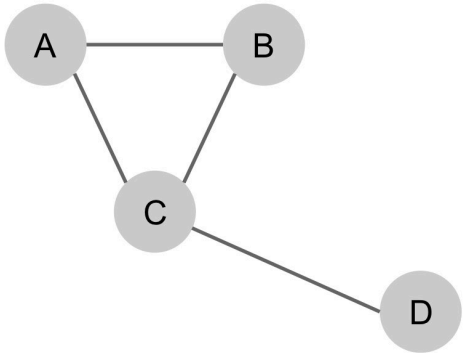
Biggest **disadvantage** of matrix:

Pattern visibility depends on how columns are ordered.

PART 01

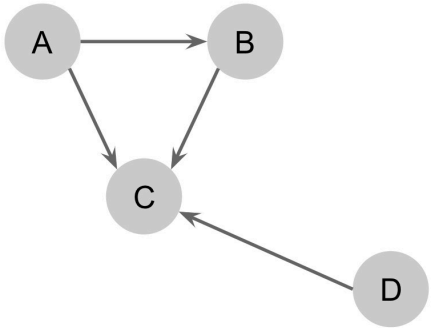


Undirected Network



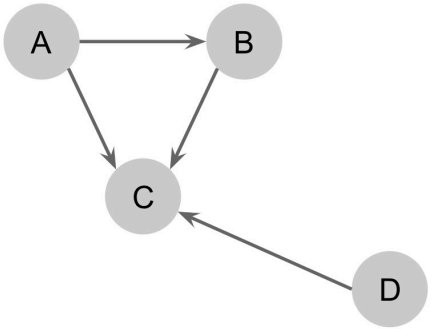
	A	B	C	D
A		■	■	
B	■		■	
C	■	■		■
D			■	

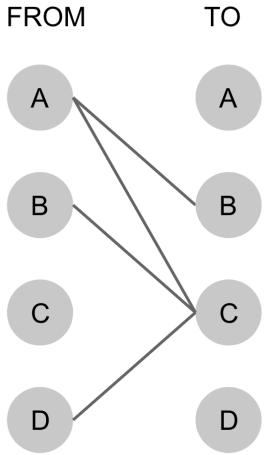
Directed Network



	A	B	C	D
A		■	■	
B			■	
C				
D			■	

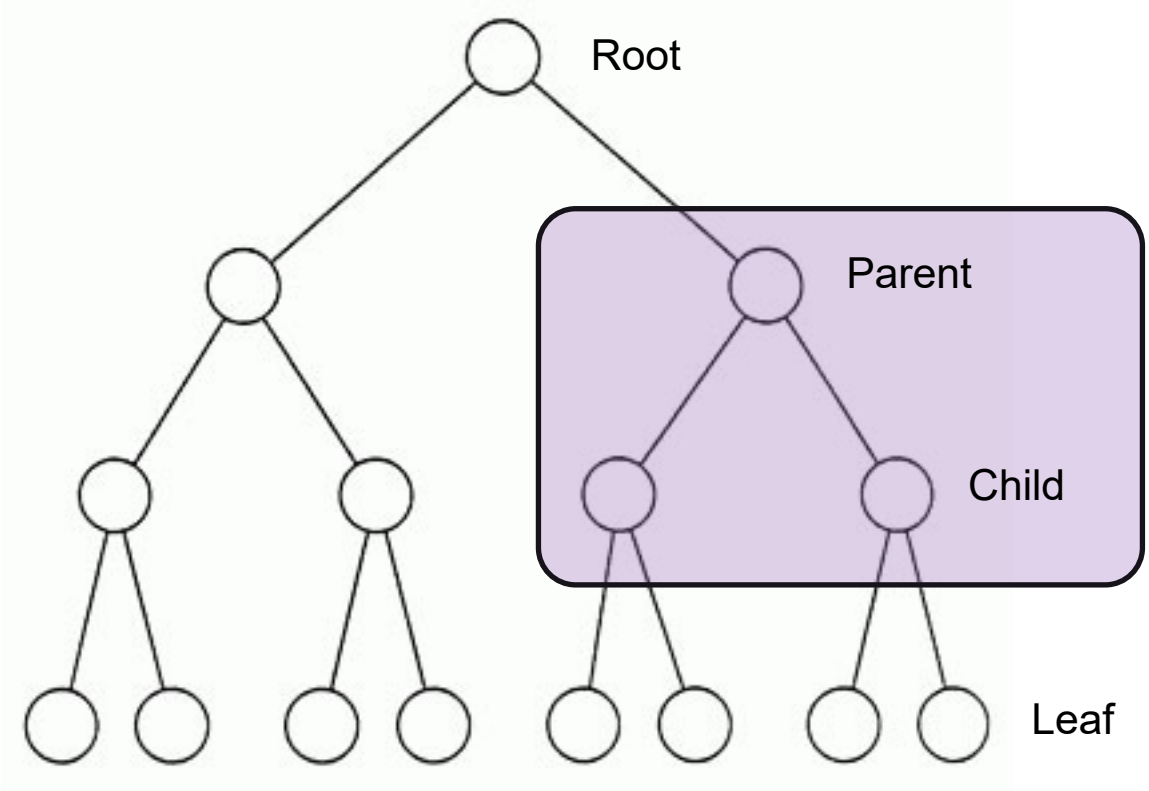
Directed Network





Parallel Axes

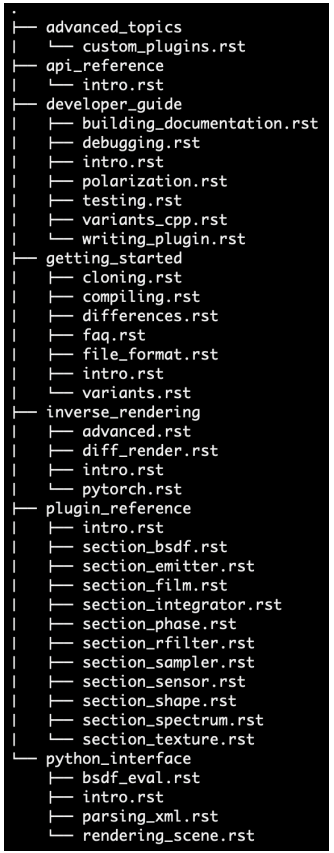
Trees & Hierarchies


Root
Parent
Child
Leaf

PART 02

Application of Trees

- File system tree.
- Evolutionary tree.
- Organization tree.
- Language structure tree.
- Etc.

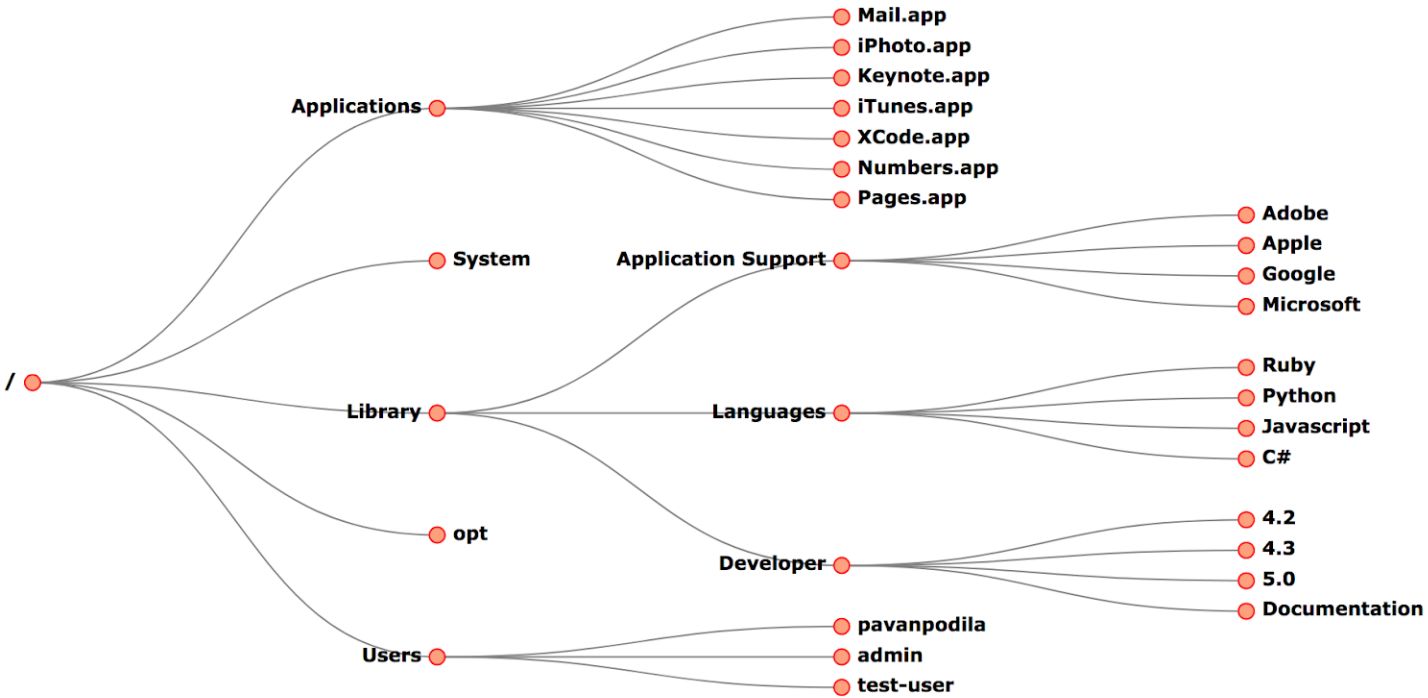


PART 02

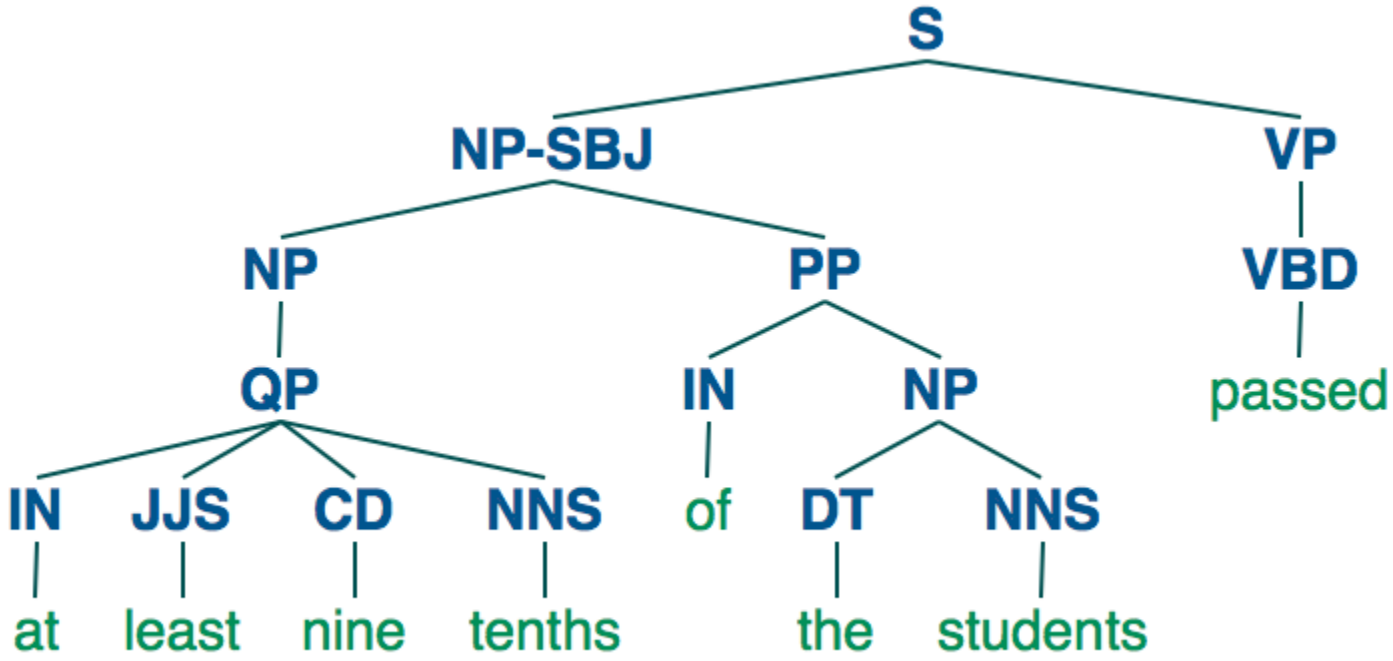
Visualizing Trees

Node-Link Diagram

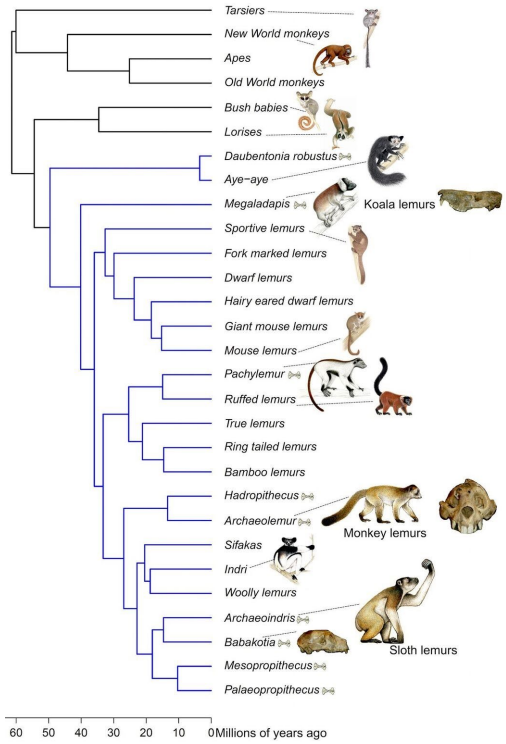
Containment



Source: https://blog.pixelingene.com/demos/d3_tree/



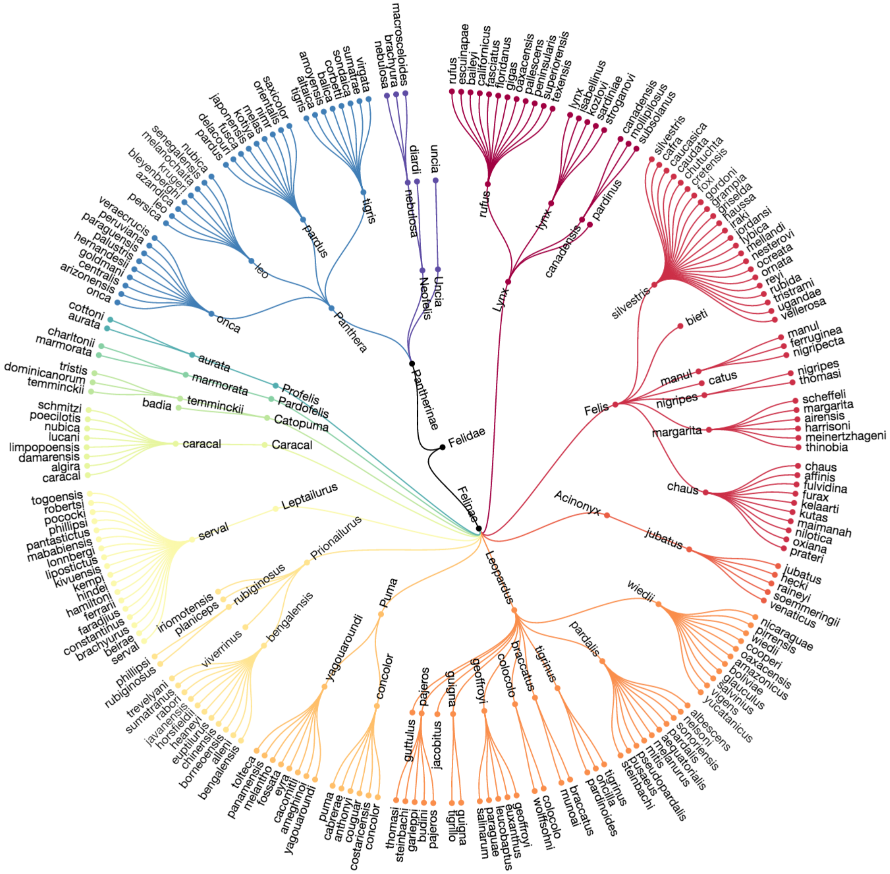
Source: https://stackoverflow.com/questions/23429117/saving-nltk-drawn-parse-tree-to-image-file



Source: http://novataxa.blogspot.com/2016/04/

- Animals
  - Birds
    - Parrot
    - Sparrow
  - Mammals
    - Humans
    - Cows
  - Fish
    - Salmon
  - Reptiles
    - Snake
    - Lizard
- Plants
  - Trees
    - Mango
    - Banyan
    - Redwood
  - Shrubs
    - Cactus
    - Congress

Source: https://stackoverflow.com/questions/14922247/how-to-get-a-tree-in-html-using-pure-css

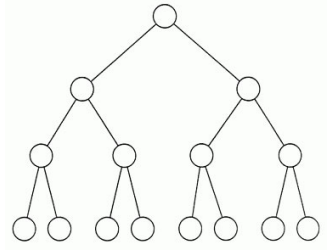


Source: https://www.safaribooksonline.com/library/view/expert-data-visualization/9781786463494/274b60c6-4b5f-42f0-8b02-393c91ae18dc.xhtml

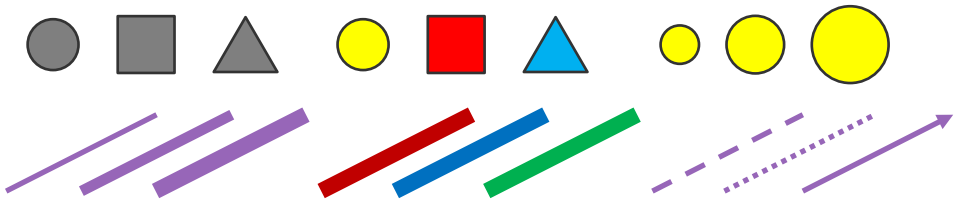
PART 02

Node-Link Diagram

- Scalability (1D growth).
- Labeling.
- Encoding information.



...

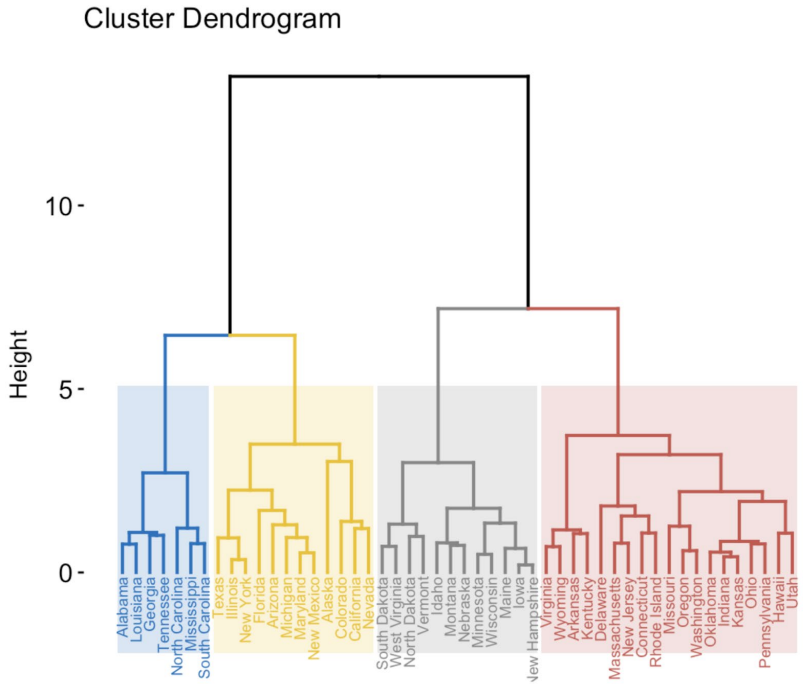


PART 02

Dendrogram

PART 02

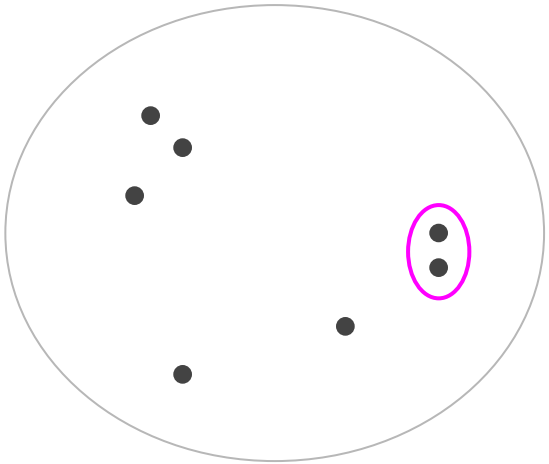
Dendrogram



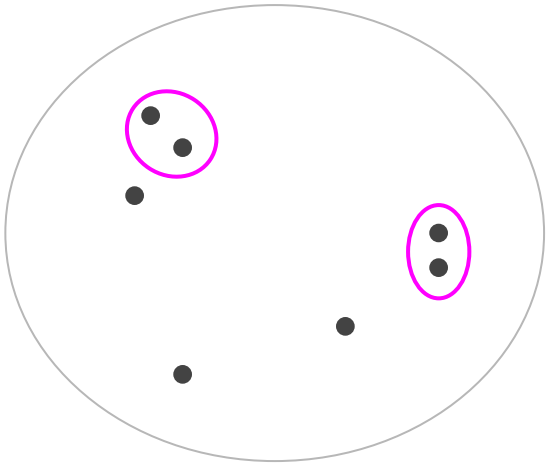
- Binary Tree
- Represent **hierarchical cluster**.

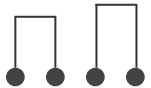
Hierarchical clustering:
Data mining algorithm to organize objects into a hierarchy according to a similarity metric defined between the objects.

Hierarchical Clustering

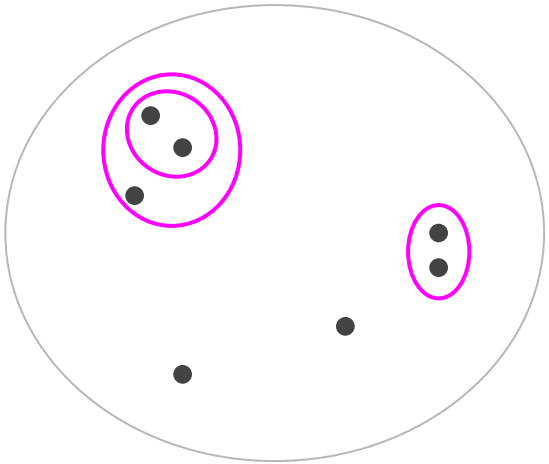


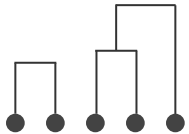
Hierarchical Clustering



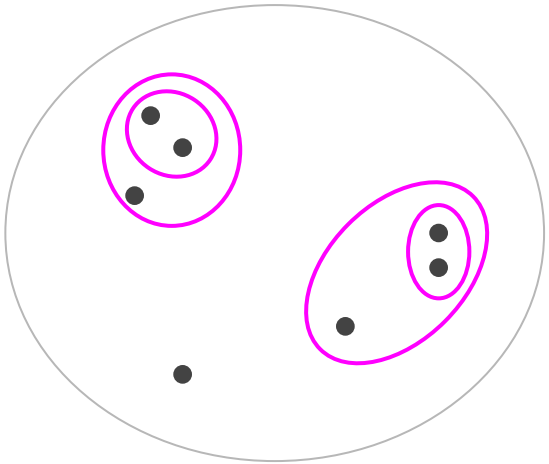


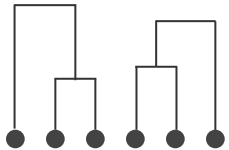
Hierarchical Clustering



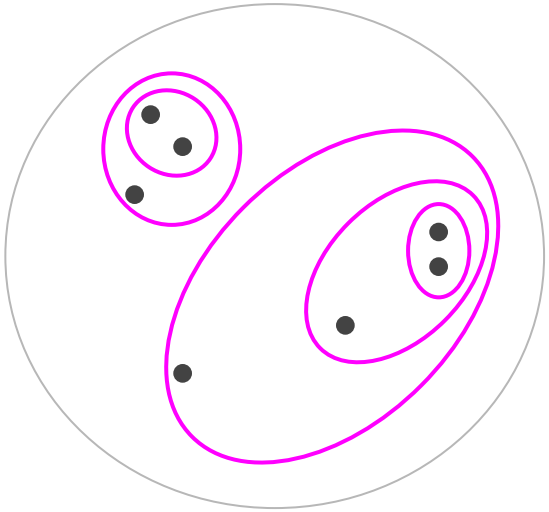


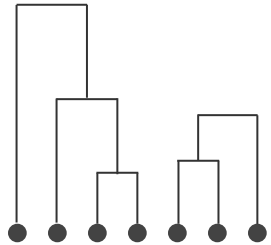
Hierarchical Clustering



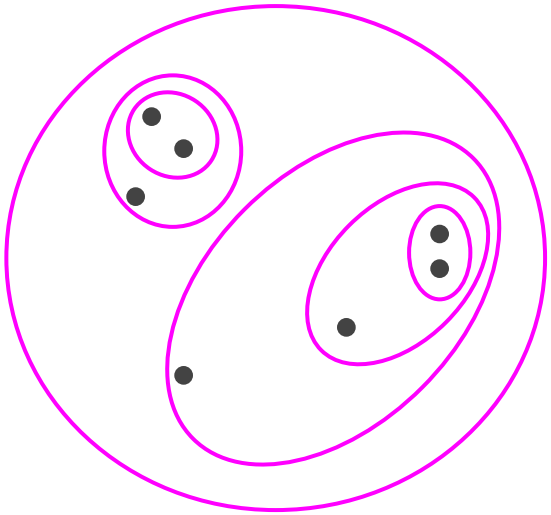


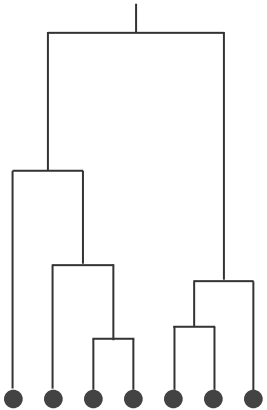
Hierarchical Clustering



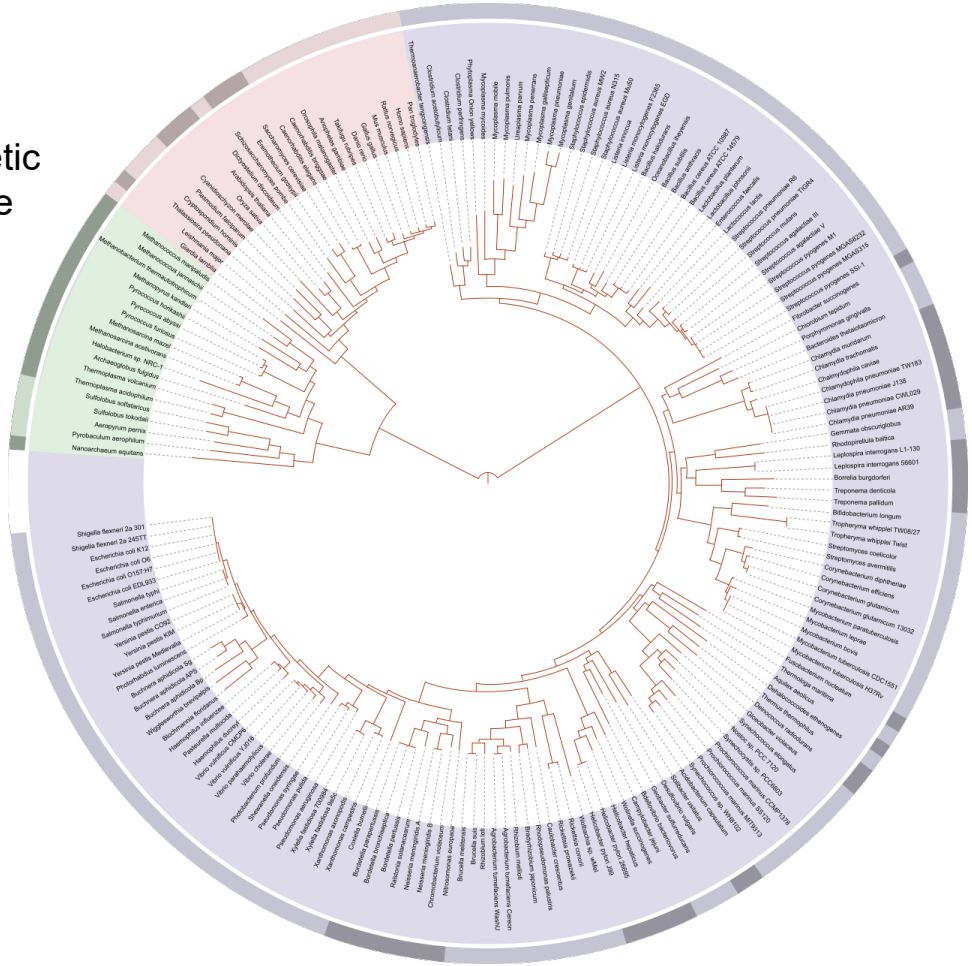


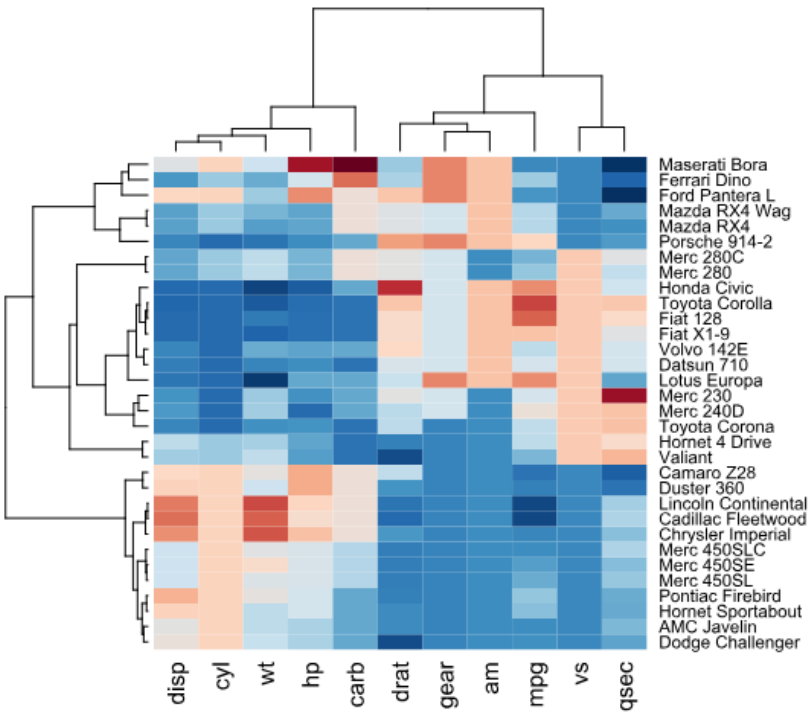
Hierarchical Clustering



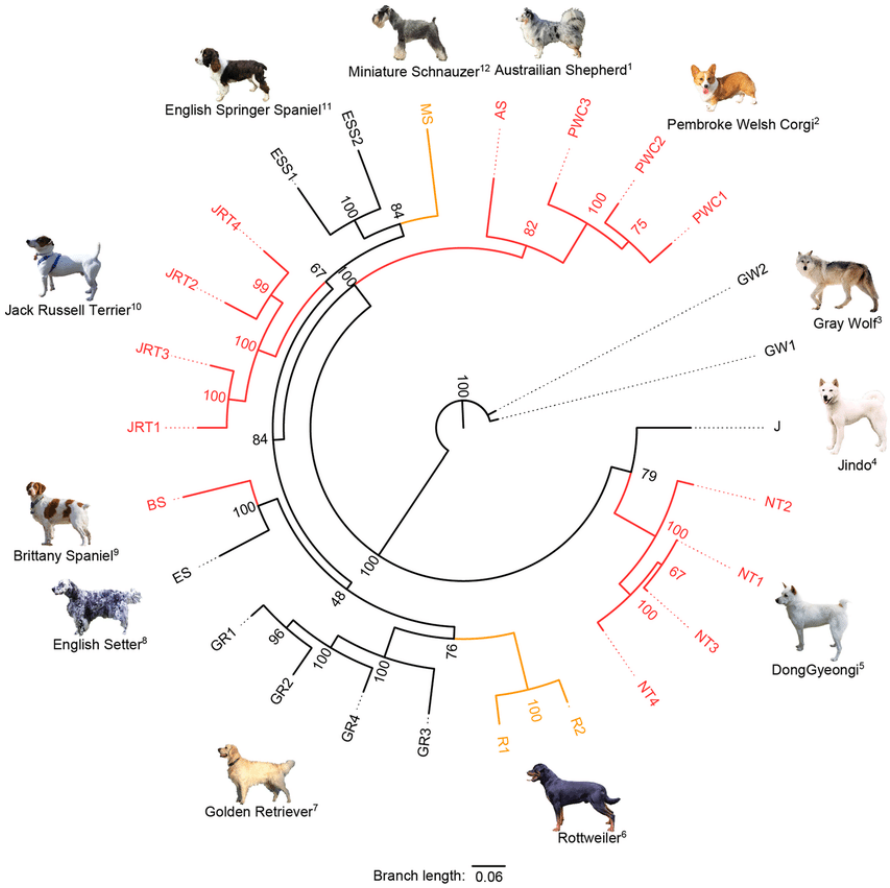


Phylogenetic Trees Computed from “Genetic Distance” from multiple sequence alignments.




Maserati Bora
Ferrari Dino
Ford Pantera L
Mazda RX4 Wag
Mazda RX4
Porsche 914-2
Merc 280C
Merc 280
Honda Civic
Toyota Corolla
Fiat 128
Fiat X1-9
Volvo 142E
Datsun 710
Lotus Europa
Merc 230
Merc 240D
Toyota Corona
Hornet 4 Drive
Valiant
Camaro Z28
Duster 360
Lincoln Continental
Cadillac Fleetwood
Chrysler Imperial
Merc 450SLC
Merc 450SE
Merc 450SL
Pontiac Firebird
Hornet Sportabout
AMC Javelin
Dodge Challenger
disp
cyl
wt
hp
carb
drat
gear
am
mpg
vs
qsec

Useful to group together and visualize any set of complex objects on top of which a distance function can be defined ...

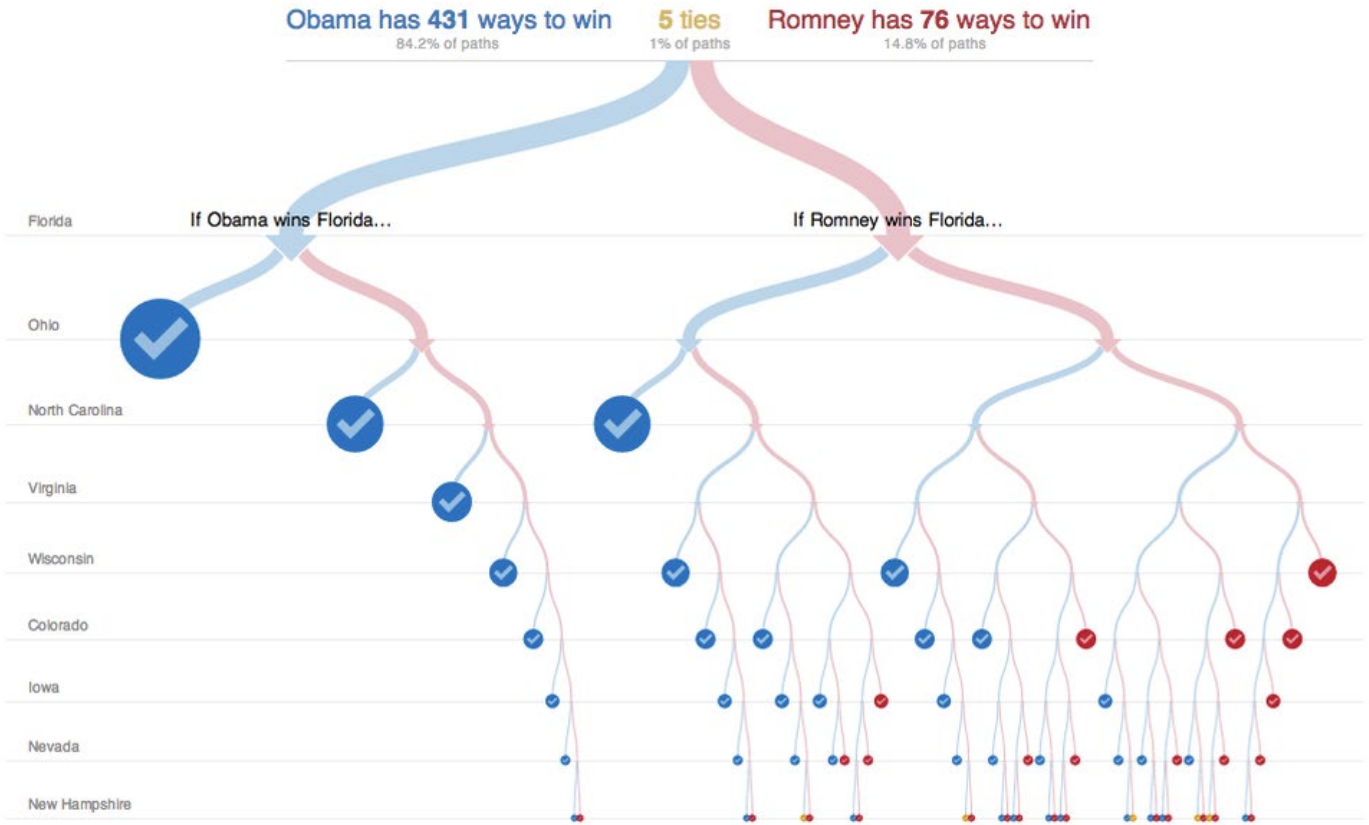


Yoo, D., Kim, K., Kim, H. *et al.* The Genetic Origin of Short Tail in Endangered Korean Dog, DongGyeongi. *Sci Rep* **7,** 10048 (2017). https://doi.org/10.1038/s41598-017-10106-6

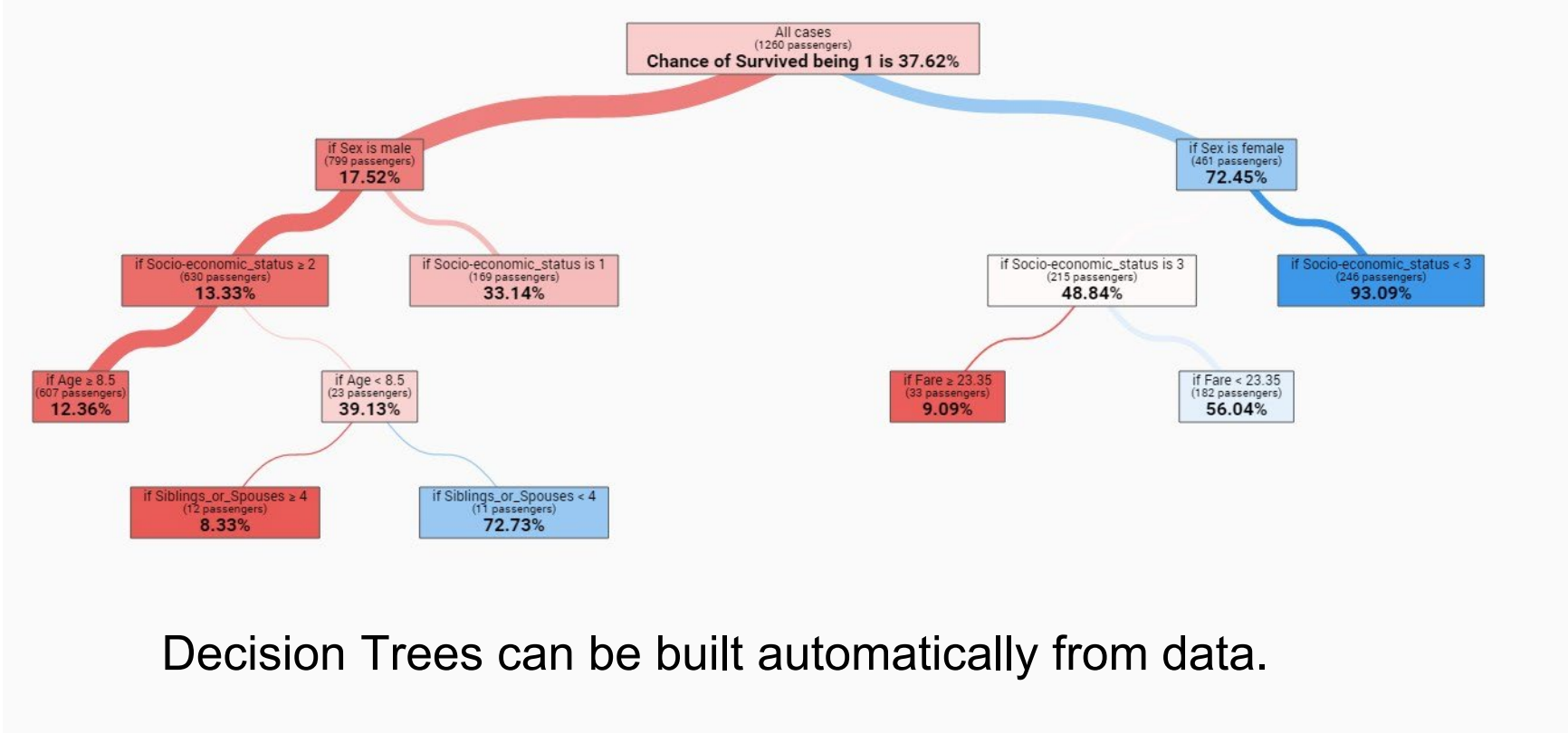
PART 02

Decision Trees

512 Paths to White House



Source: https://source.opennews.org/articles/nyts-512-paths-white-house/

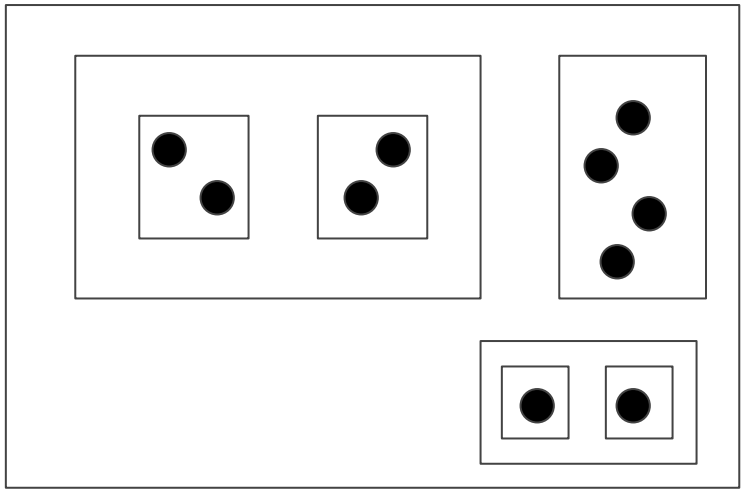


Decision Trees can be built automatically from data.

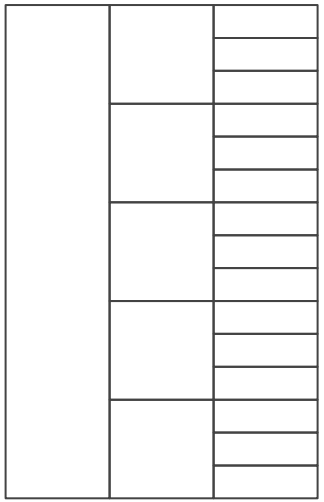
PART 02

Space Partitioning & Containment Methods

Containment

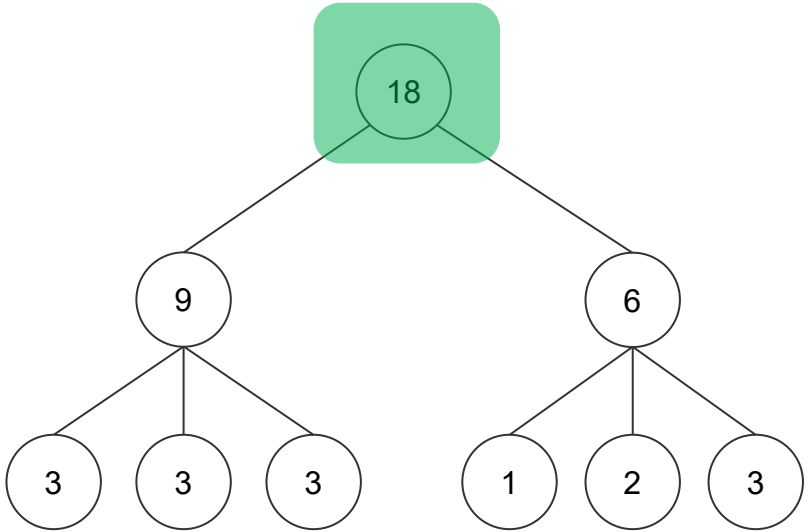


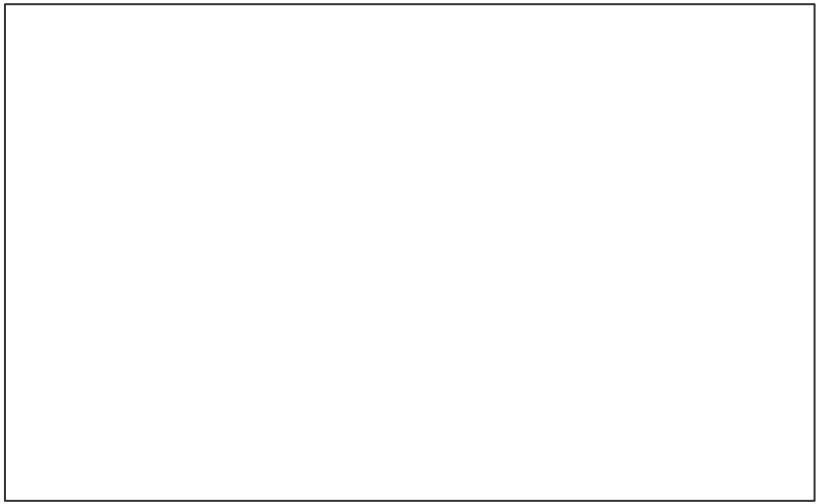
Partitioning



PART 02

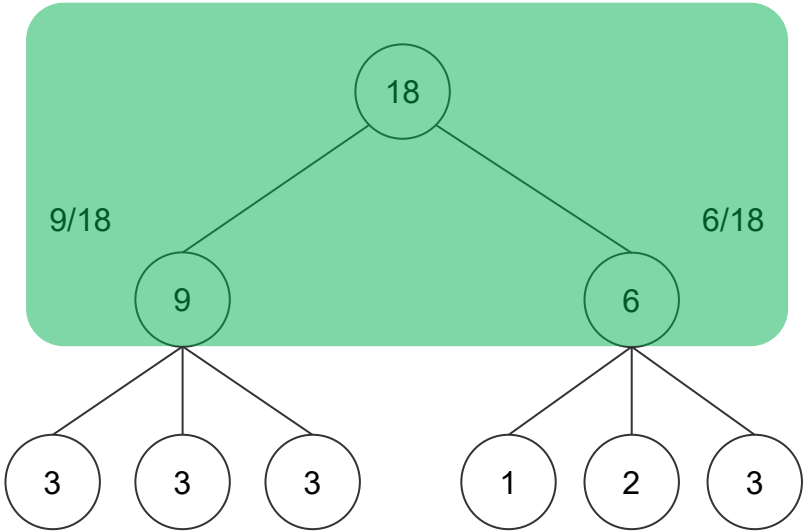
Tree Map

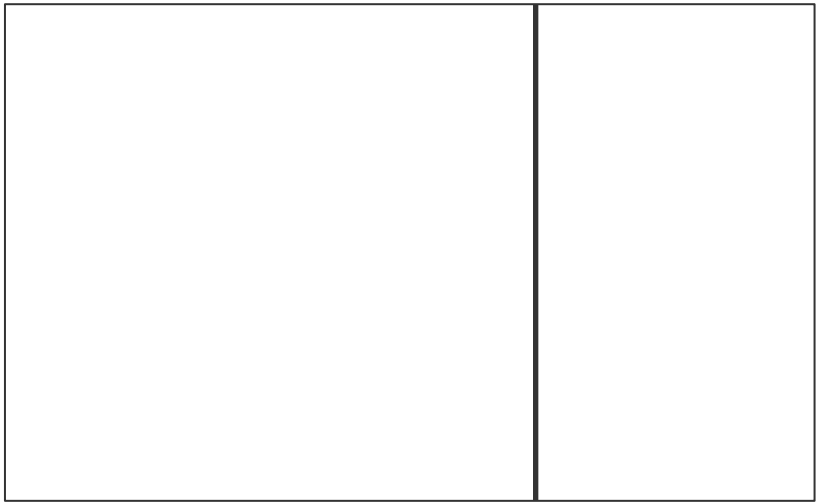




PART 02

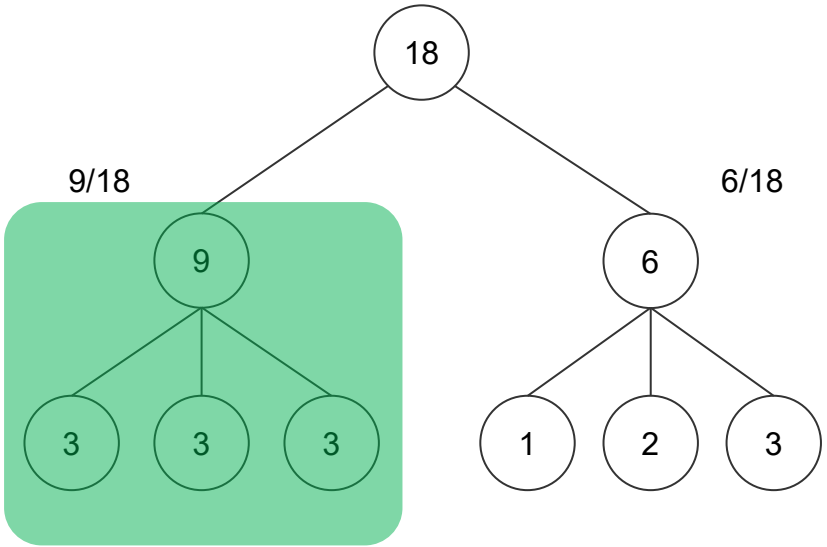
Tree Map

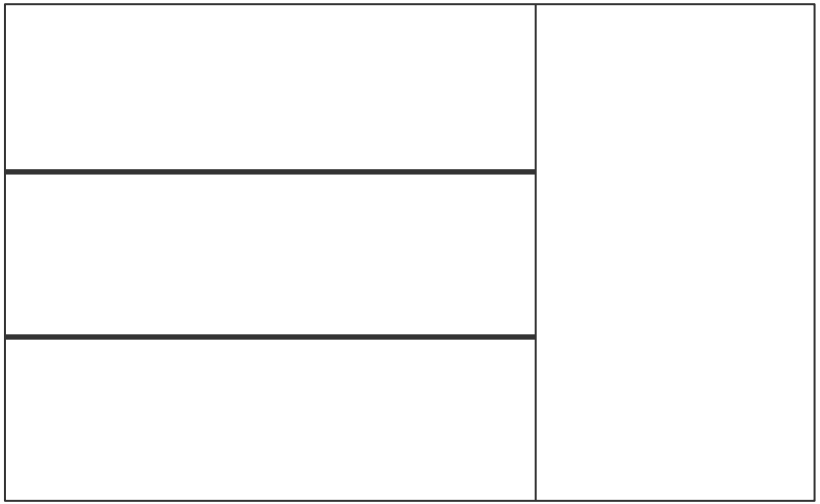




PART 02

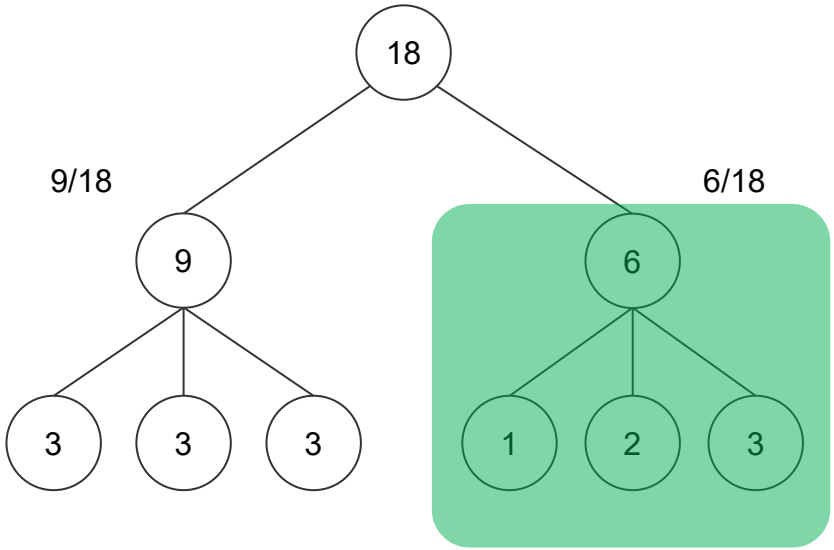
Tree Map

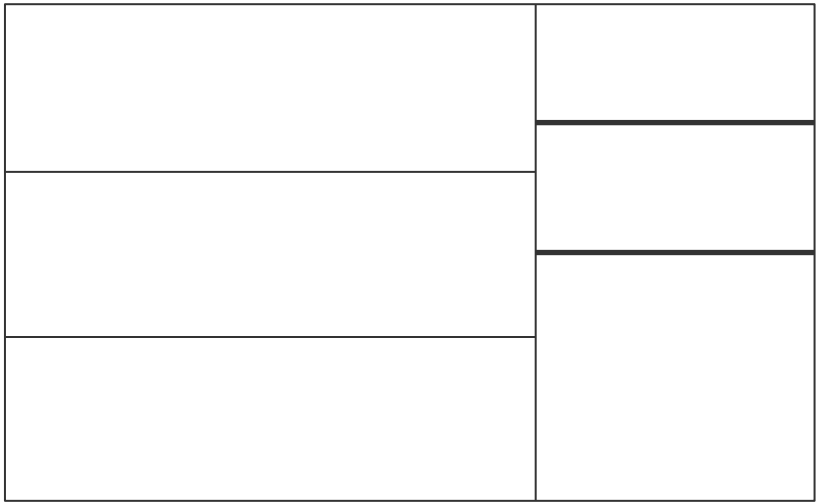




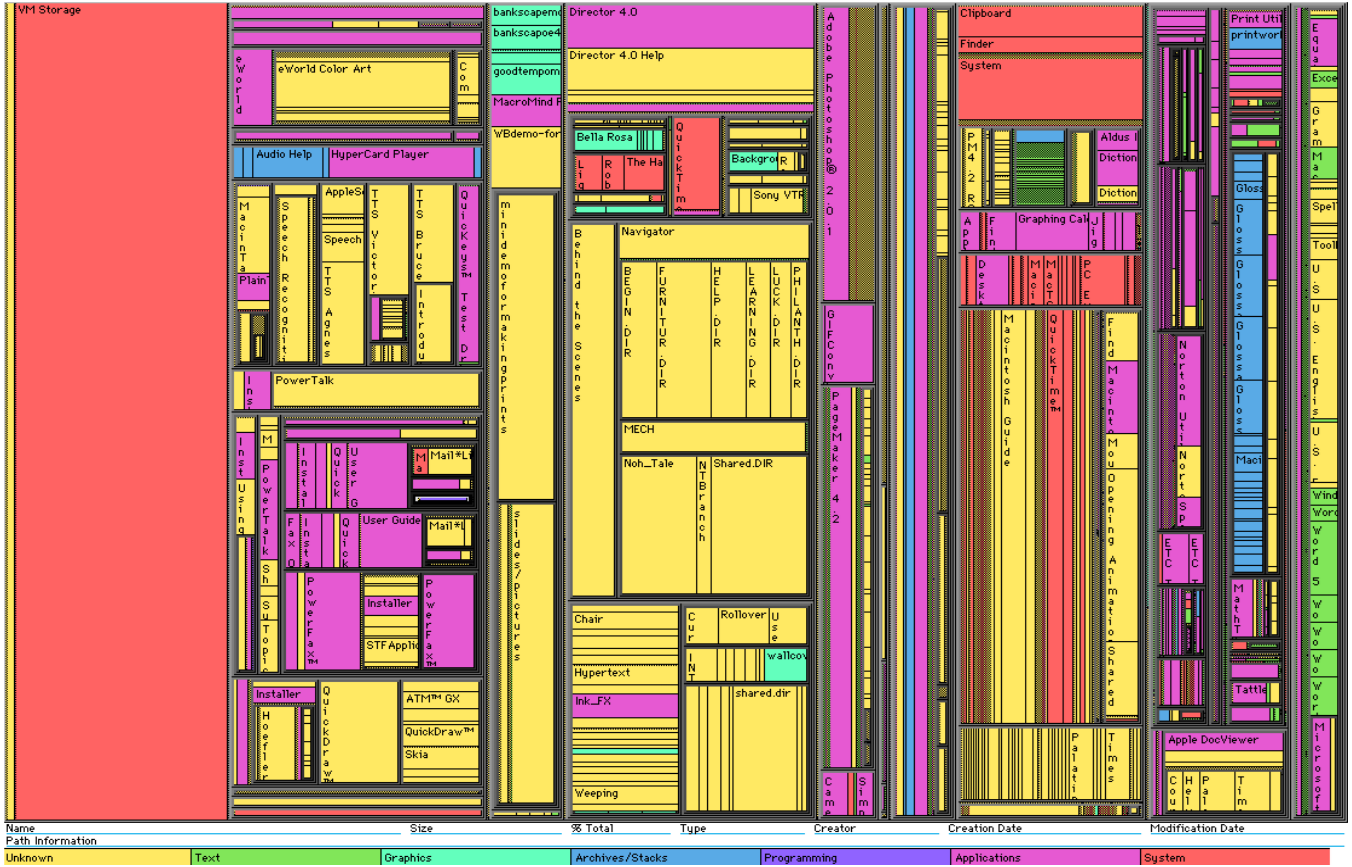
PART 02

Tree Map





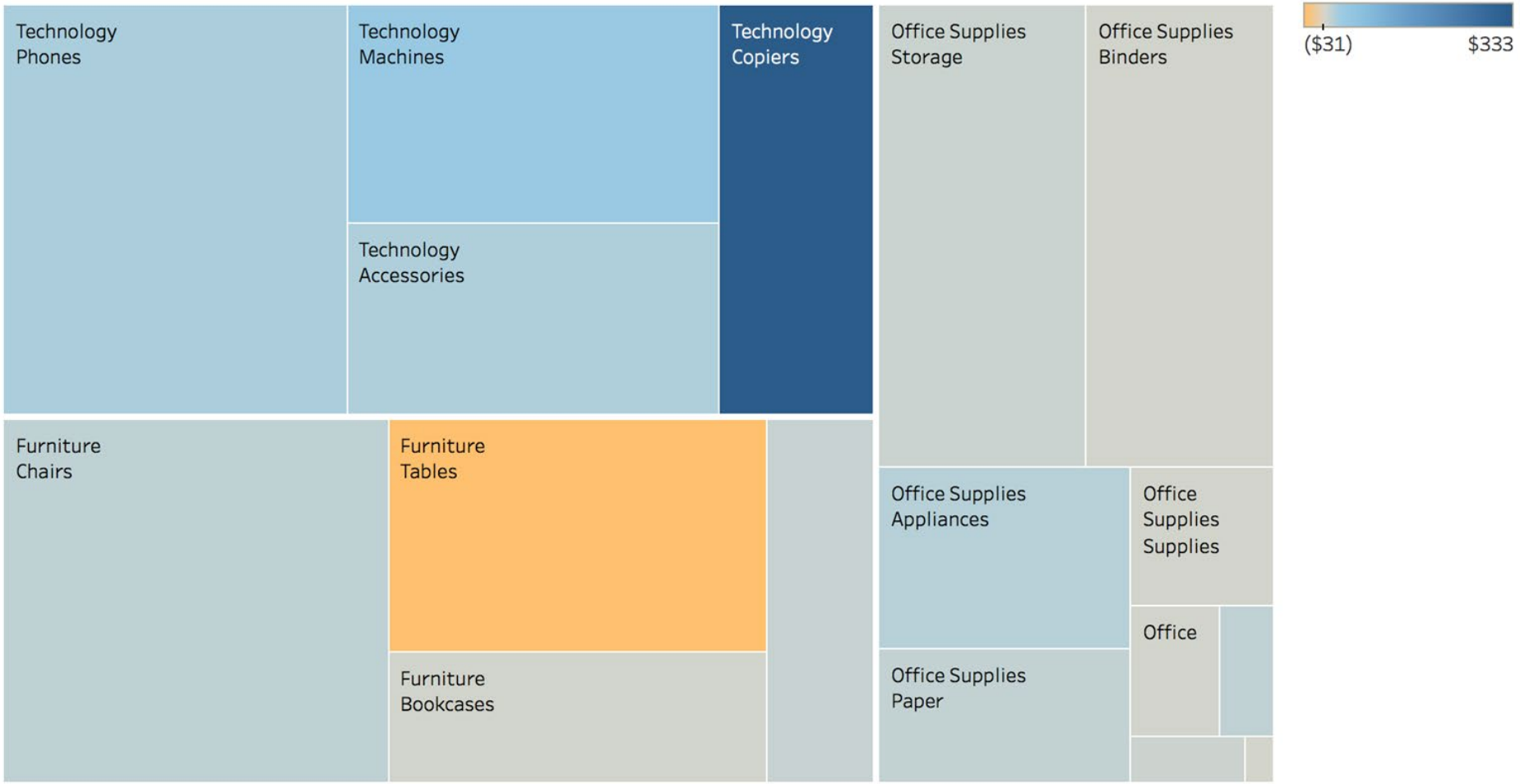
“During 1990, in response to the common problem of a filled hard disk, I became obsessed with the idea of producing a compact visualization of directory tree structures.” - Ben Shneiderman

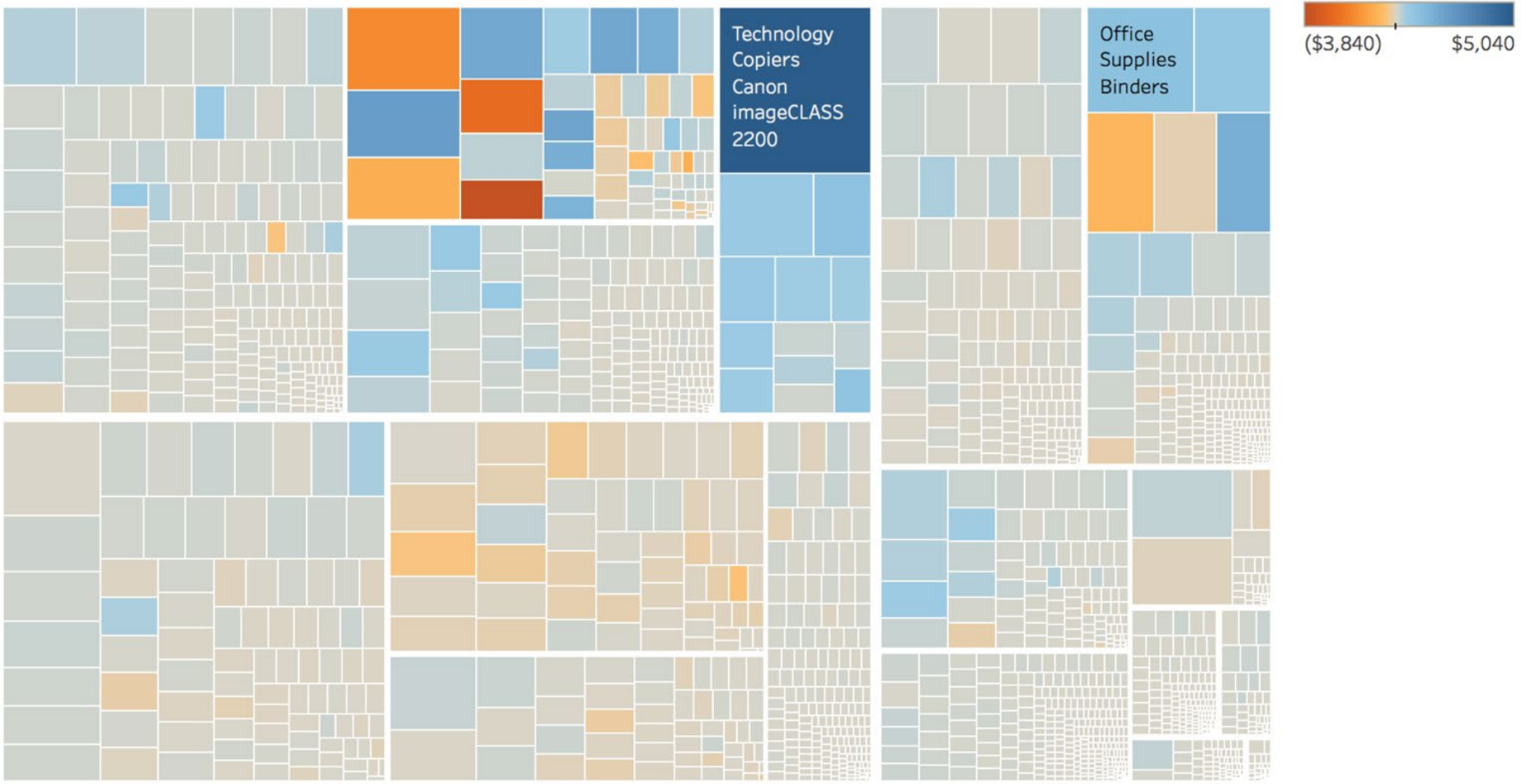


What information can be visualized with a tree map?

Area: Quantity
Color: Quantity/Category
Hierarchy: Nesting

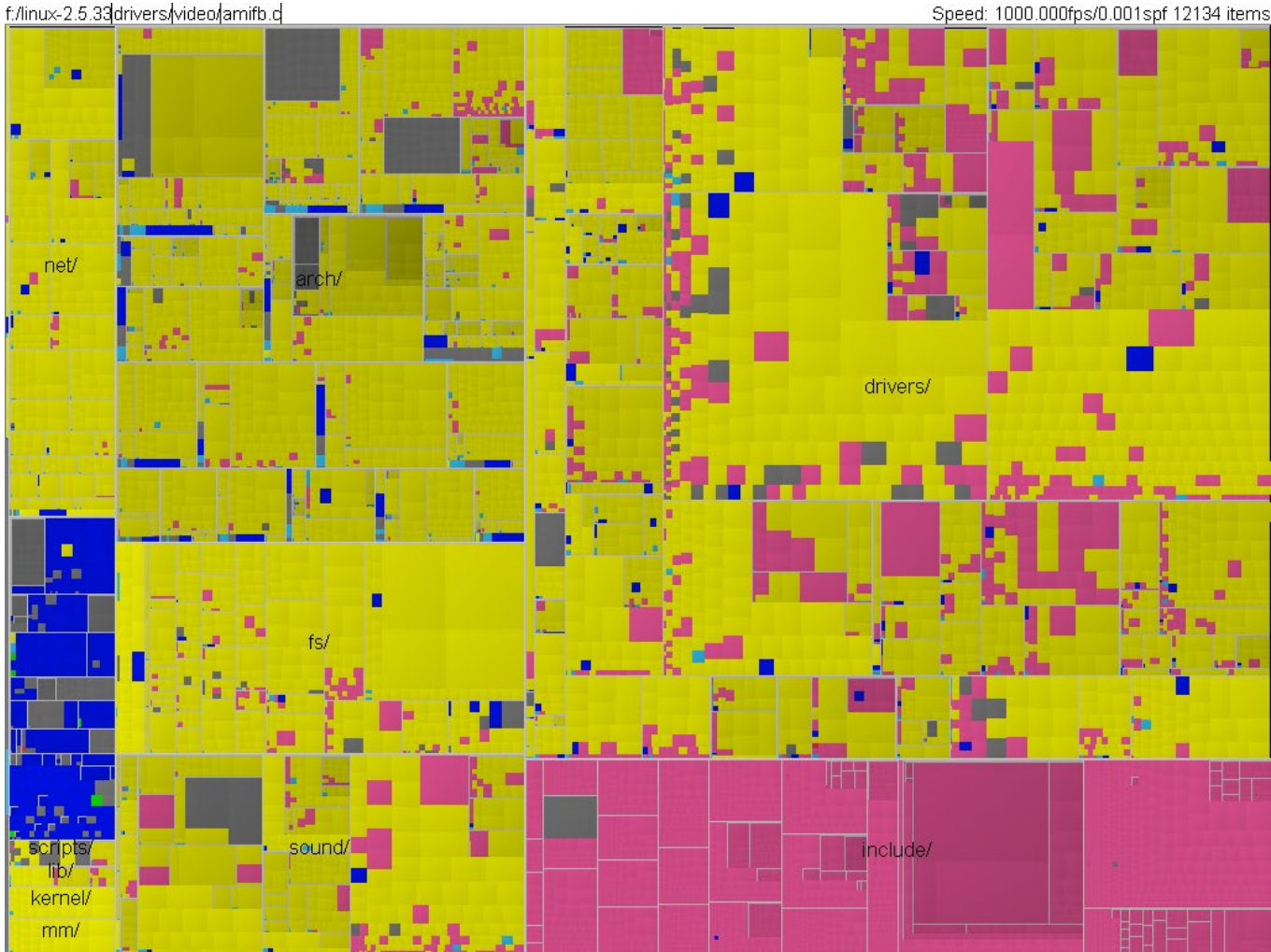
Source: http://www.cs.umd.edu/hcil/treemap-history/


Technology
Phones
Technology
Machines
Technology
Copiers
Technology
Accessories
Office Supplies
Storage
Office Supplies
Binders
($31)
$333
Furniture
Chairs
Furniture
Tables
Furniture
Bookcases
Office Supplies
Appliances
Office
Supplies
Supplies
Office
Office Supplies
Paper


Technology
Copiers
Canon
imageCLASS
2200
Office
Supplies
Binders
($3,840)
$5,040

Linux Kernel

Source files: yellow
Header files: pink
Text files: dark blue
Makefiles: light blue
Shell scripts: red
Images: green
Non-registered suffixes: grey

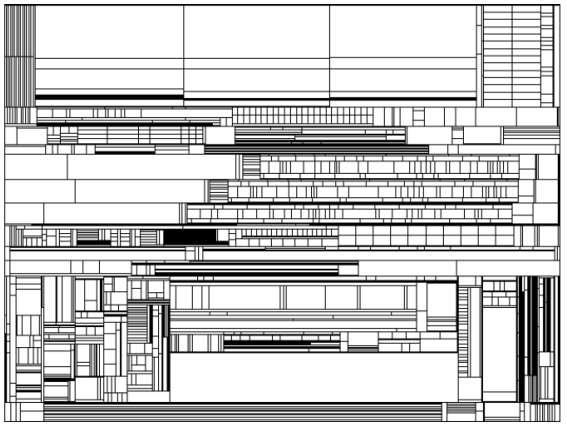


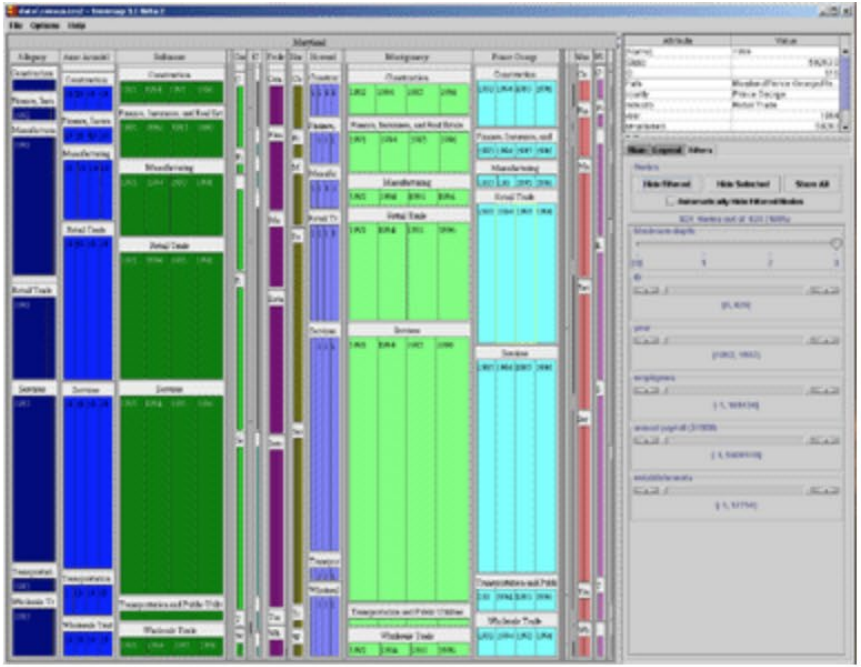
PART 02

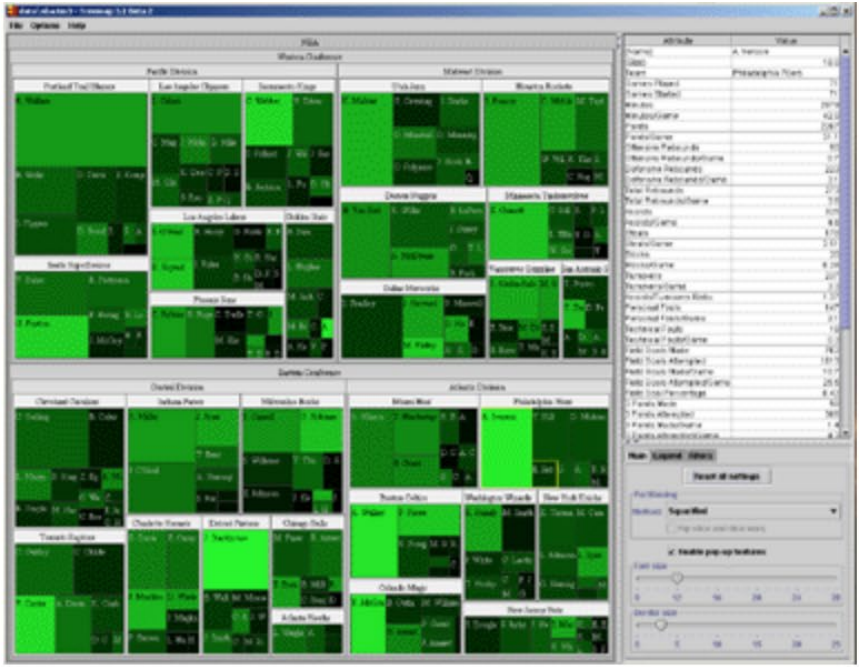
Tree Map: Drawbacks

When the rectangles have very different aspect ratios (proportion of height vs. width), comparing areas gets harder (especially with thin elongated rectangles).

Solution: Squarified Tree Maps.







Late-Breaking Results ACM ISBN: 1-58113-158-5 CHI 99 15-20 MAY 1999

Visualizing the Stock Market

Martin Wattenberg
Dow Jones & Co. (SmartMoney Magazine)
1755 Broadway
New York, NY 10019 USA
+1 212 830 9226
mwattenberg@smartmoney.com

ABSTRACT
We describe a new 2-dimensional visualization algorithm capable of presenting detailed information on hundreds of items while emphasizing overall patterns in the data. This display method, which builds on Shneiderman's treemap technique, makes use of both hierarchy and similarity information. We have implemented this display in the *SmartMoney Map of the Market*, a web page that reports current data on over 500 publicly traded companies.

Keywords
Visualization, interactive graphics, treemap, investing

INTRODUCTION
A key goal of financial journalism is to answer the question, "How is the market doing today?" What makes this question tough is that there are so many possible answers: on a given day, the market as a whole may be up, but technology stocks could be down--but Apple Computer's stock might be up. Summing up the market with one or two index values hides a lot of the action. But if you give all the details, as in the stock price pages of a newspaper, the result is hard to read and there is no way to spot overall trends. This paper describes a new interactive graphical display, related to Shneiderman's treemap diagram [3], that allows a user to track the performance of hundreds of stocks at once without losing sight of the bigger picture.

PROBLEM
Using a treemap to provide a snapshot of stock market performance makes sense for several reasons. Hierarchical classifications of companies (into broad sectors and, within sectors, specific industries) are common. Companies also have a natural "size," namely their market capitalization. Finally, price performance can be indicated by color coding. However, previous approaches to using treemaps to display stock market data[2] present two problems.

First, the "slice-and-dice" layout method often creates a partition containing rectangles with extremely large aspect ratios. Figure 1, a slice-and-dice layout with two levels of hierarchy, clearly shows this effect.

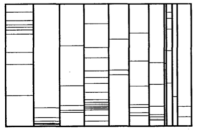


Figure 1: Typical slice-and-dice layout

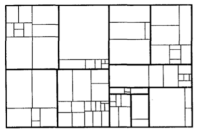


Figure 2: New layout scheme, described below

Rectangles with high aspect ratios are often hard to see, let alone compare in size. Extreme aspect ratios also make a zoom option problematic, since the zooming transformation must stretch one direction more than the other, which can confuse the user. Previous attempts to solve this problem, such as intentionally distorting rectangle sizes or eliminating the high-aspect-ratio rectangles[2,4], interfere with the accuracy of the diagram.

The second problem is that we wish to place similar companies and sectors next to each other. Doing so gives the user added cues to the relationships between companies and sectors, and increases the chance that when several similar companies experience related price movements, the

Squarified Treemaps

Mark Bruls, Kees Huizing, and Jarke J. van Wijk

Eindhoven University of Technology
Dept. of Mathematics and Computer Science,
P.O. Box 513,
5600 MB Eindhoven, The Netherlands
email:{keesh, vanwijk}@win.tue.nl

Abstract. An extension to the treemap method for the visualization of hierarchical information, such as directory structures and organization structures, is presented. The standard treemap method often gives thin, elongated rectangles. As a result, rectangles are difficult to compare and to select. A new method is presented to generate lay-outs in which the rectangles approximate squares. To strengthen the visualization of the structure, shaded frames are used around groups of related nodes.

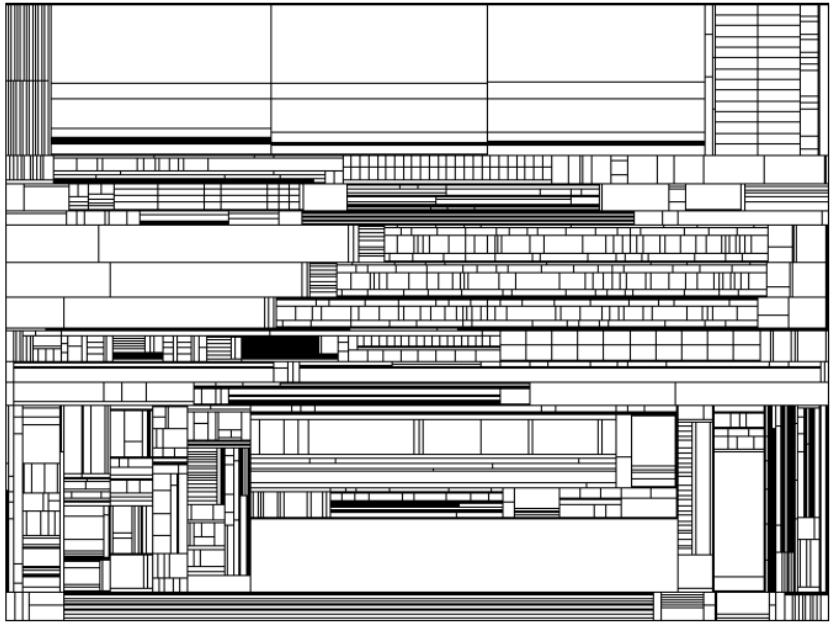
1 Introduction

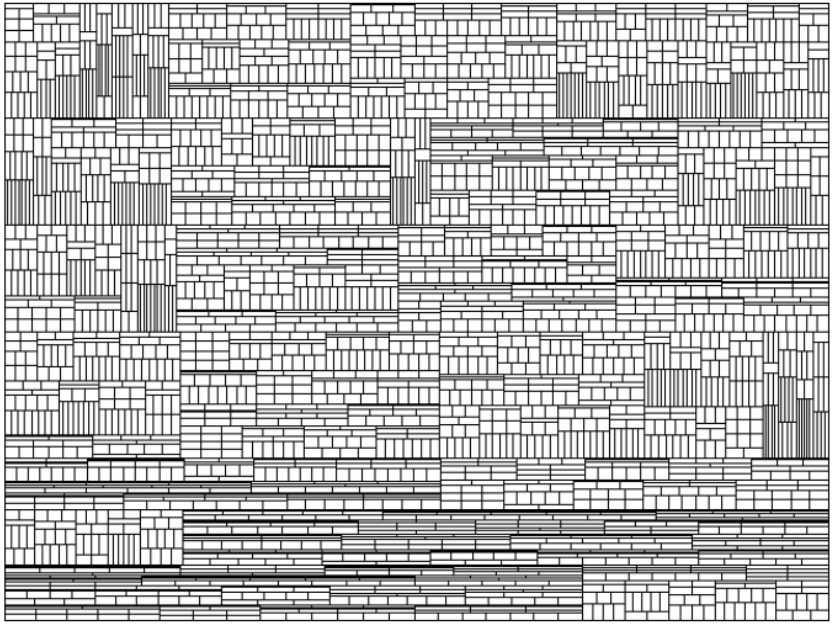
Hierarchical structures of information are everywhere: directory structures, organization structures, family trees, catalogues, computer programs, and so on. Small hierarchical structures are effective to locate information, but the content and organization of large structures is harder to grasp. We present a new visualization method for large hierarchical structures: Squarified Treemaps. The method is based on Treemaps, developed by Shneiderman and Johnson [9, 6]. Treemaps are efficient and compact displays, which are particularly effective to show the size of the final elements in the structure. In a previous paper [10] we introduced Cushion Treemaps, which provide shading as an extra cue to emphasize the hierarchical structure. In this paper we attack another problem of standard treemaps: the emergence of thin, elongated rectangles. We propose a new method to subdivide rectangular areas, such that the resulting subrectangles have a lower aspect ratio. These rectangles use space more efficiently, are easier to point at in interactive environments, and are easier to estimate with respect to size. Because the resulting structures are somewhat harder to grasp, we also introduce an improved method to visualize nested structures. A variant on nested treemaps is presented, where the rectangular enclosures have been replaced by shaded frames. The combination of these two methods leads to displays of hierarchical structures that are efficient and easy to understand.

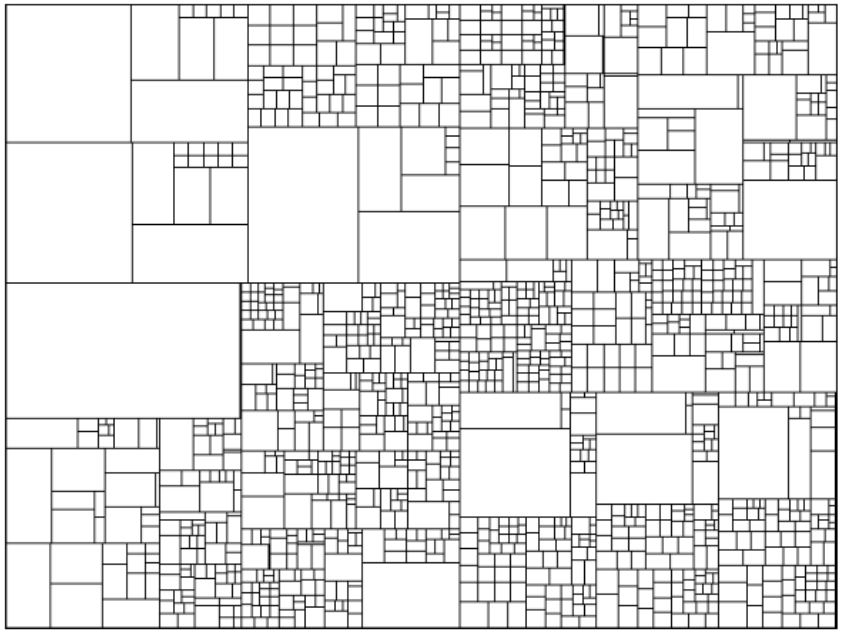
In section 2 we discuss existing methods to visualize hierarchical structures. The new method for improved subdivision is presented in section 3. The shaded frames are described in section 4. Finally, we discuss the results in section 5.

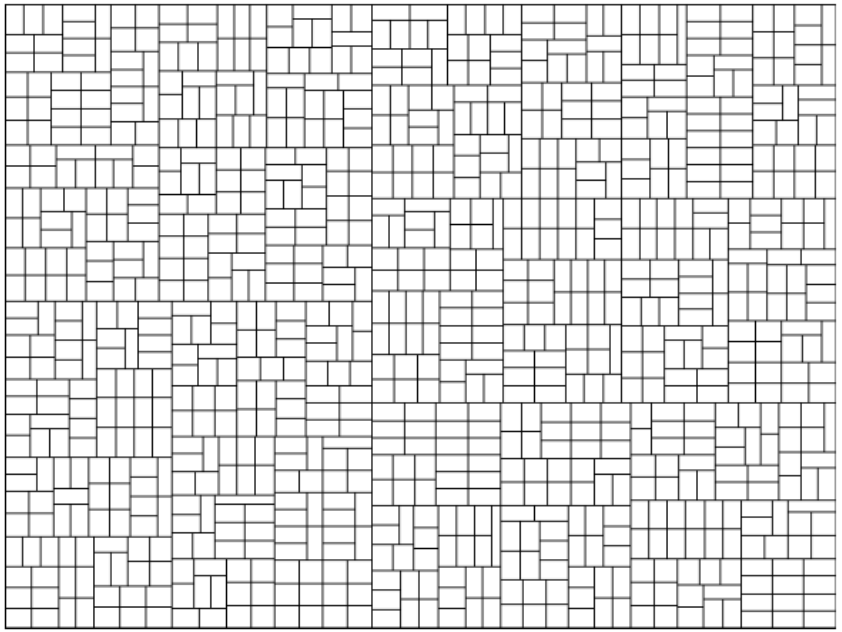
2 Background

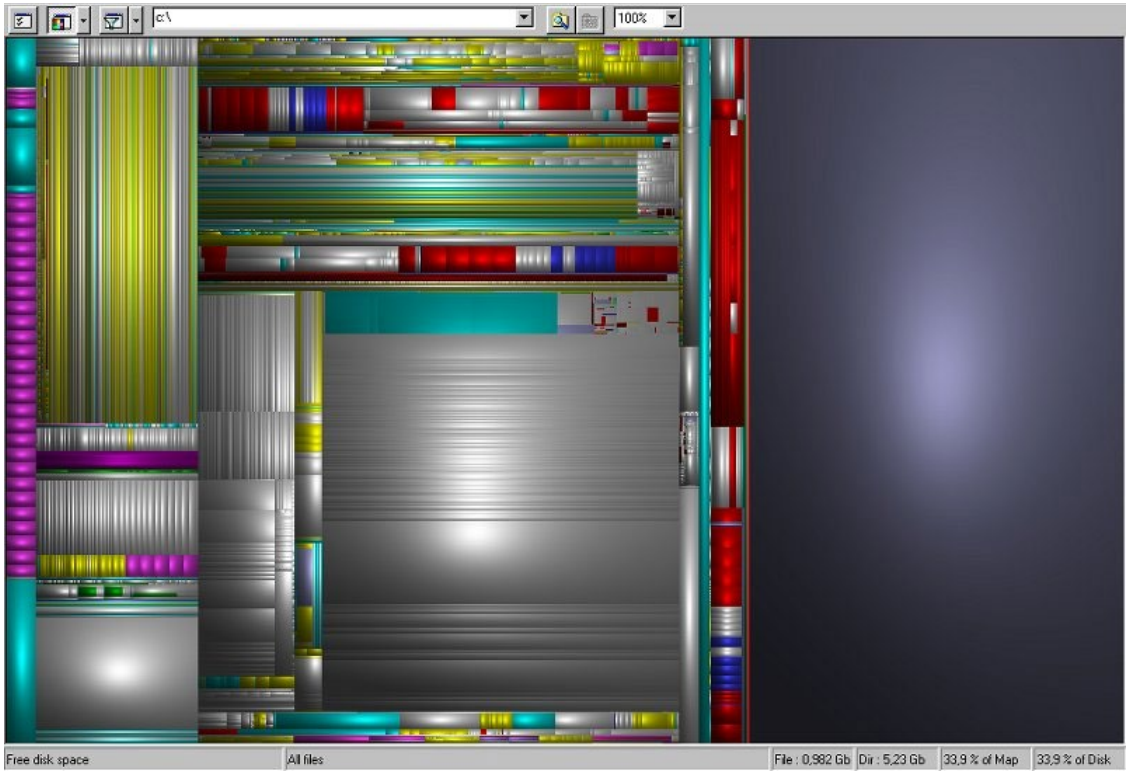
Many methods exist to browse through and to display hierarchical information structures, or, for short, trees. File browsers are the best known example. Usually a listing of









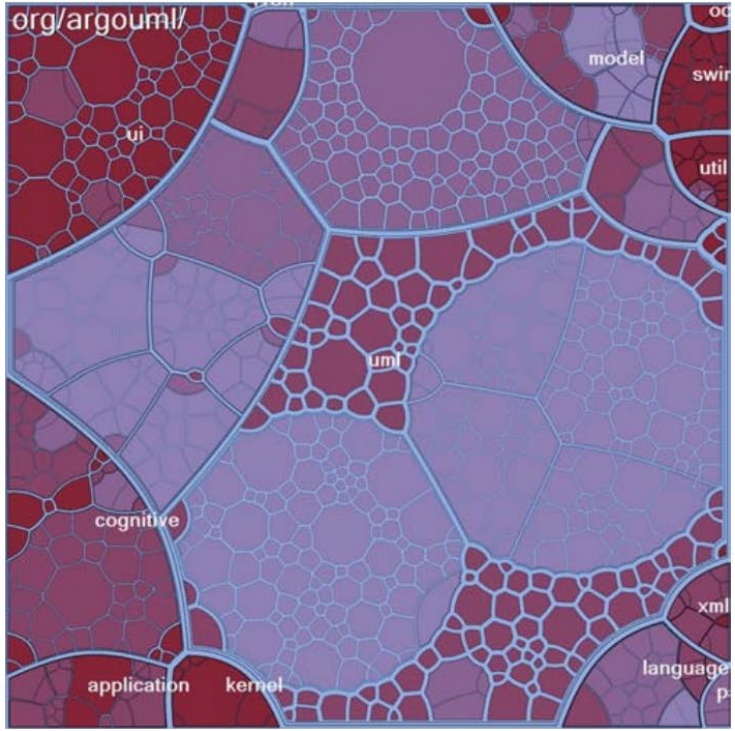


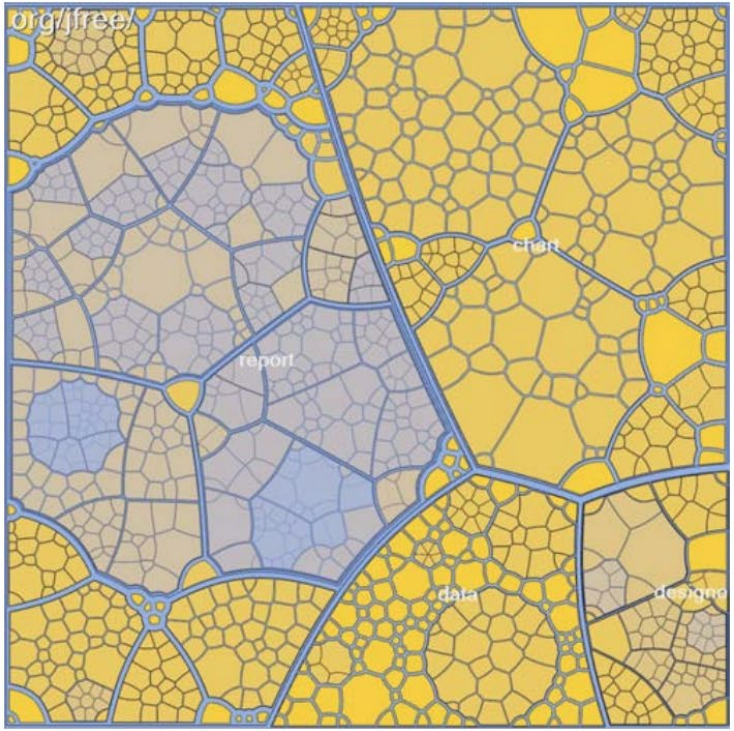
Cushion Tree Maps

http://www.win.tue.nl/~vanwijk/ctm.pdf

News Tree Maps

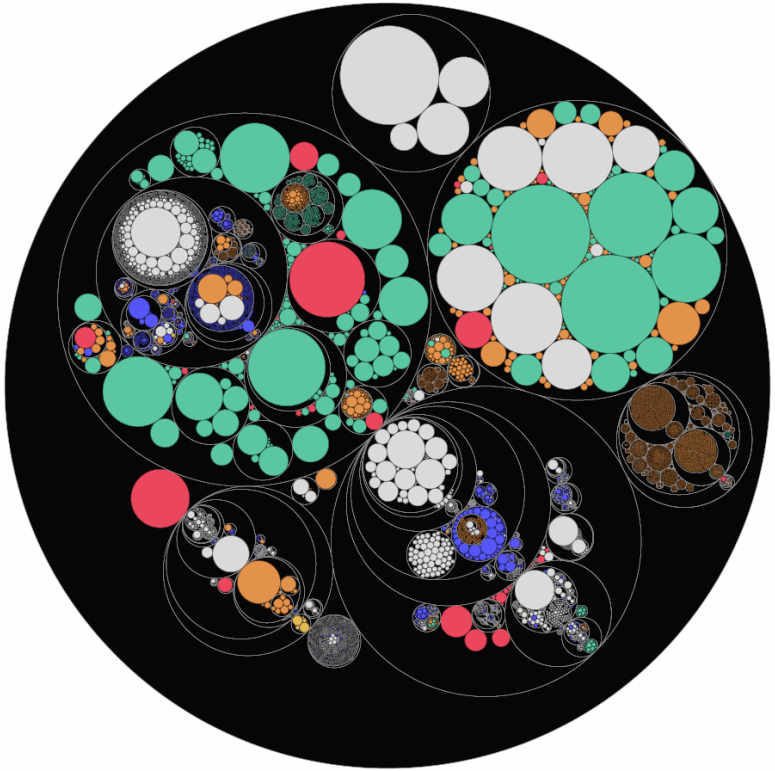
https://www.addictivetips.com/internet-tips/newsmap-google-news-with-a-color-coded-treemap-visualization/





Voronoi Tree Maps

http://delivery.acm.org/10.1145/1060000/1056041/p165-balzer.pdf



Circle-Packing Tree Maps

Treemaps for space-constrained visualization of hierarchies

Including the History of Treemap Research at the University of Maryland

Started Dec. 26th, 1998 by Ben Shneiderman
Later updates by Ben Shneiderman and Catherine Plaisant - Last update Sept 2014

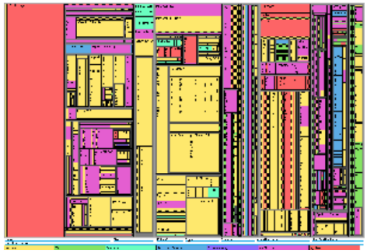
Our treemap products:

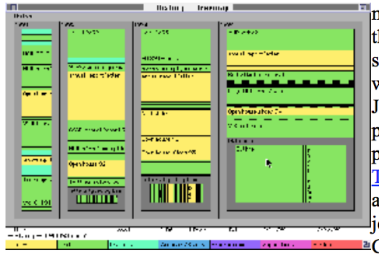
- Treemap 4.0: General treemap tool (Free demo version, plus licensing information for full package)
- PhotoMesa: Zoomable image library browser (Free demo version, plus licensing information for full package)
- Treemap Algorithms and Algorithm Animations (Open source Java code)

A History of Treemap Research at the University of Maryland

During 1990, in response to the common problem of a filled hard disk, I became obsessed with the idea of producing a compact visualization of directory tree structures. Since the 80 Megabyte hard disk in the HCIL was shared by 14 users it was difficult to determine how and where space was used. Finding large files that could be deleted, or even determining which users consumed the largest shares of disk space were difficult tasks.

Tree structured node-link diagrams grew too large to be useful, so I explored ways to show a tree in a space-constrained layout. I rejected strategies that left blank spaces or those that dealt with only fixed levels or fixed branching factors. Showing file size by area coding seemed appealing, but various rectangular, triangular, and circular strategies all had problems. Then while puzzling about this in the faculty lounge, I had the Aha! experience of splitting the screen into rectangles in alternating horizontal and vertical directions as you traverse down the levels. This recursive algorithm seemed attractive, but it took me a few days to convince myself that it would always work and to write a six line algorithm. This algorithm and the initial designs led to the first Technical Report (HCIL TR 91-03) in March 1991 which was published in the ACM Transactions on Graphics in January 1992 (http://www.acm.org/pubs/citations/journals/tog/1992-11-1/p92-shneiderman/). Choosing the right name took probably as long, but the term 'treemap' described the notion of turning a tree into a planar space-filling map.





My initial design simply nested the rectangles, but a more comprehensible design used a border to show the nesting. Finding an effective visualization strategy took only a few months but producing a working piece of software took over a year. Brian Johnson implemented the algorithms and refined the presentation strategies while preserving rapid performance even with 5,000 node hierarchies. The TreeViz application ran on color Macintosh models and led to the widely cited paper (HCIL TR 91-06) jointly authored paper in the October 1991 IEEE Conference on Visualization. This paper was reprinted in Readings in Information Visualization. I think treemaps are a convenient representation that has unmatched utility for certain tasks. The capacity to see tens of thousands of nodes in a fixed space and find large areas or duplicate directories is very powerful. I still use TreeViz for cleaning up my Macintosh. It does take some learning for novices to grasp the tree structure layout in treemaps, but the benefits are great.

Source: http://www.cs.umd.edu/hcil/treemap-history/

PART 02

Tree Map Summary

Advantages:

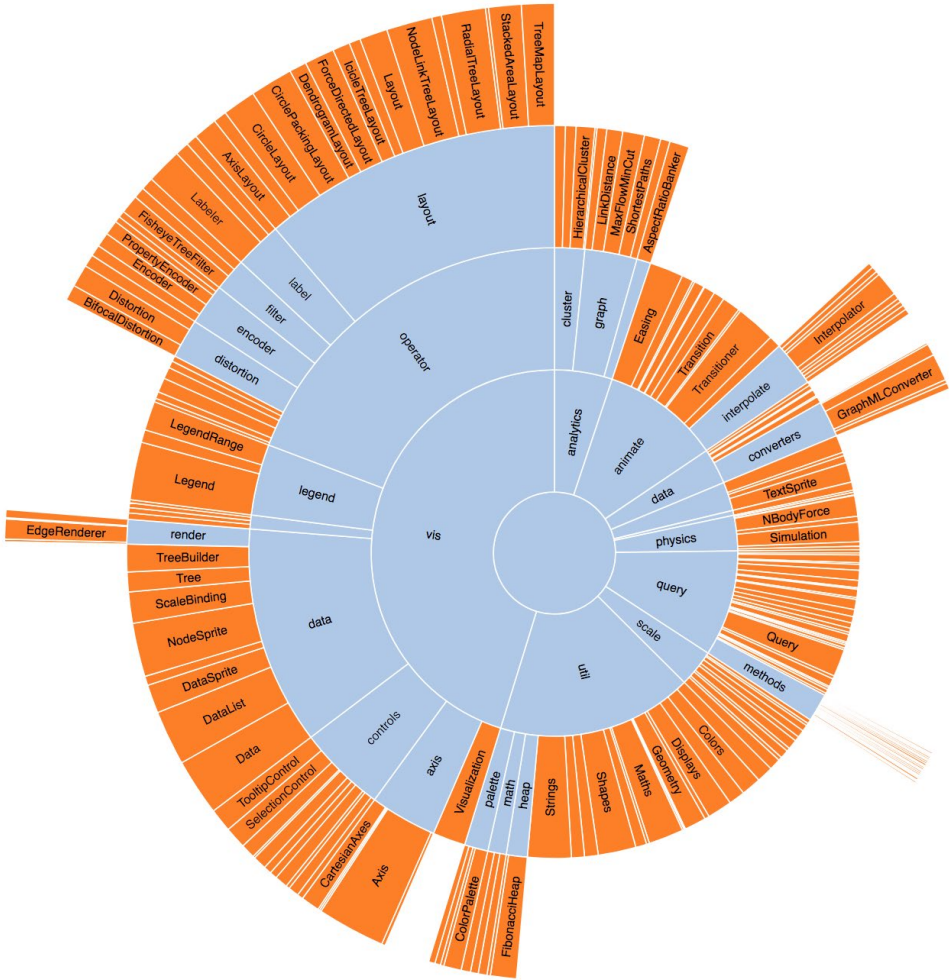
- Scalability.
  - Node visibility.
  - No Overlapping marks.
  - Supports size and color.

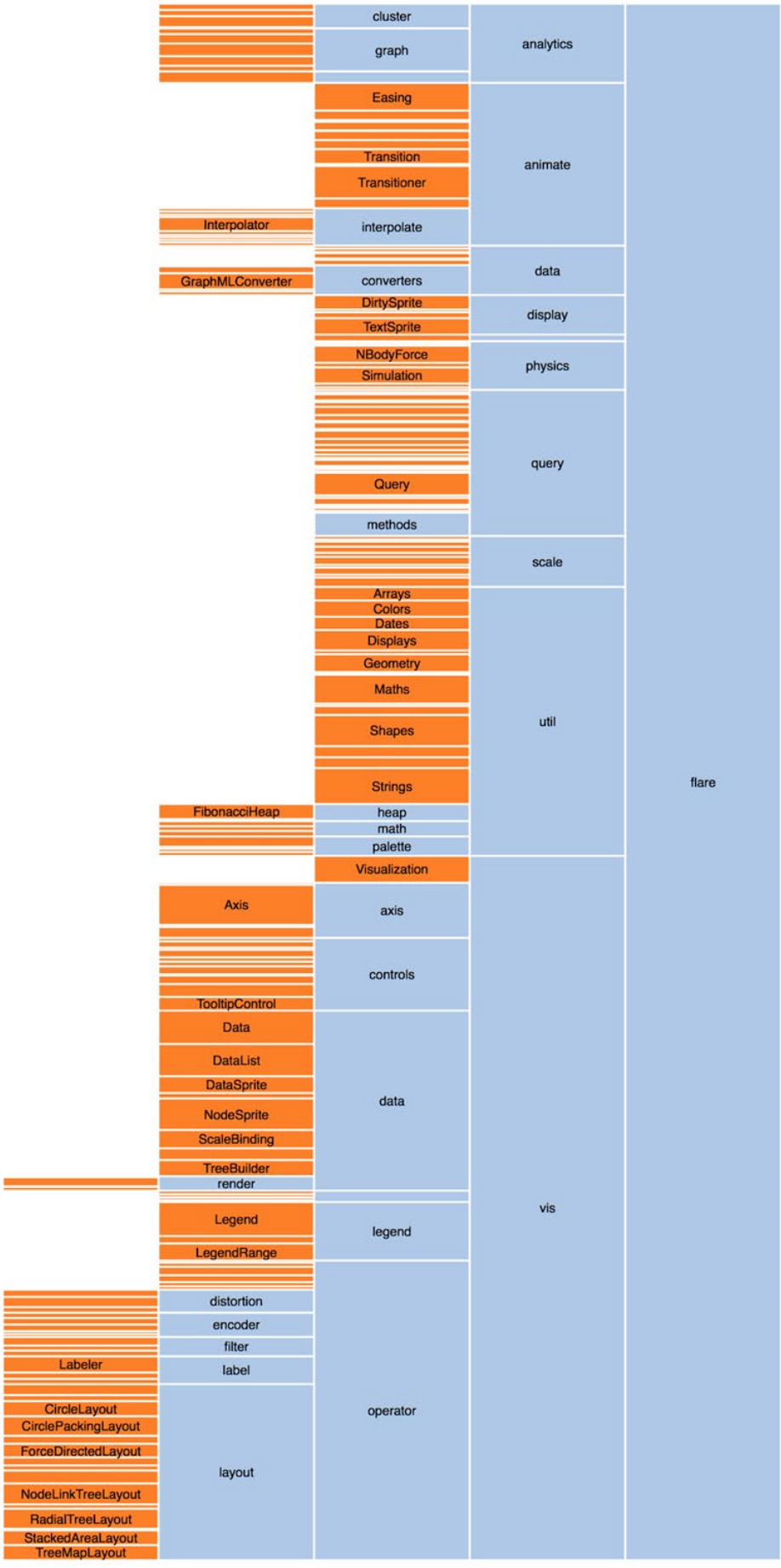
Disadvantages:

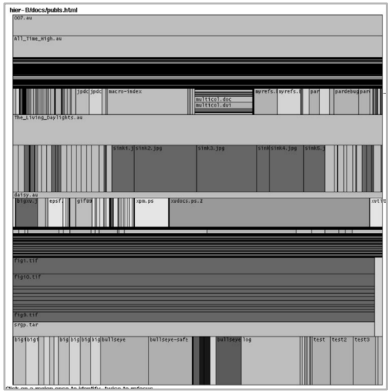
- Size is not the most accurate channel.
- Structures is hard to discern.

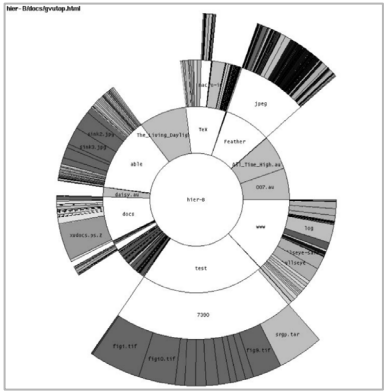
PART 02

Sunburst and Icicle Plots


vis
analytics
animate
data
physics
query
scale
util
cluster
graph
HierarchicalCluster
LinkDistance
MaxFlowMinCut
ShortestPaths
AspectRatioBanker
Easing
Transition
Transitioner
interpolate
converters
Interpolator
GraphMLConverter
TextSprite
NBodyForce
Simulation
Query
methods
Colors
Displays
Geometry
Maths
Shapes
Strings
Visualization
palette
math
heap
ColorPalette
FibonacciHeap
operator
layout
label
filter
encoder
distortion
TreeMapLayout
StackedAreaLayout
RadialTreeLayout
NodeLinkTreeLayout
Layout
IcicleTreeLayout
ForceDirectedLayout
DendrogramLayout
CirclePackingLayout
CircleLayout
AxisLayout
Labeler
FisheyeTreeFilter
PropertyEncoder
Encoder
Distortion
BifocalDistortion
legend
LegendRange
Legend
controls
axis
CartesianAxes
Axis
render
EdgeRenderer
data
TreeBuilder
Tree
ScaleBinding
NodeSprite
DataSprite
DataList
Data
TooltipControl
SelectionControl


cluster
graph
analytics
flare
Easing
Transition
Transitioner
animate
Interpolator
interpolate
GraphMLConverter
converters
data
DirtySprite
TextSprite
display
NBodyForce
Simulation
physics
Query
query
methods
scale
Arrays
Colors
Dates
Displays
Geometry
Maths
Shapes
Strings
util
FibonacciHeap
heap
math
palette
Visualization
Axis
axis
controls
TooltipControl
Data
DataList
DataSprite
NodeSprite
ScaleBinding
TreeBuilder
data
render
vis
Legend
LegendRange
legend
distortion
encoder
filter
Labeler
label
CircleLayout
CirclePackingLayout
ForceDirectedLayout
NodeLinkTreeLayout
RadialTreeLayout
StackedAreaLayout
TreeMapLayout
layout
operator





Int. J. Human-Computer Studies (2000) **53,** 663–694
doi:10.1006/ijhc.2000.0420
Available online at http://www.idealibrary.com on IDEAL®

An evaluation of space-filling information visualizations for depicting hierarchical structures

JOHN STASKO
Graphics, Visualization and Usability Center, College of Computing, Georgia Institute of Technology, Atlanta, GA 30332-0280, USA.
email: stasko@cc.gatech.edu

RICHARD CATRAMBONE
School of Psychology, Georgia Institute of Technology, Atlanta, GA 30332-0170, USA

MARK GUZDIAL AND KEVIN MCDONALD
Graphics, Visualization and Usability Center, College of Computing, Georgia Institute of Technology, Atlanta, GA 30332-0280, USA

(Received 31 January 2000, and accepted in revised form 31 May 2000)

A variety of information visualization tools have been developed recently, but relatively little effort has been made to evaluate the effectiveness and utility of the tools. This article describes results from two empirical studies of two visualization tools for depicting hierarchies, in particular, computer file and directory structures. The two tools examined implement space-filling methodologies, one rectangular, the Treemap method, and one circular, the Sunburst method. Participants performed typical file/directory search and analysis tasks using the two tools. In general, performance trends favored the Sunburst tool with respect to correct task performance, particularly on initial use. Performance with Treemap tended to improve over time and use, suggesting a greater learning cost that was partially recouped over time. Each tool afforded somewhat different search strategies, which also appeared to influence performance. Finally, participants strongly preferred the Sunburst tool, citing better ability to convey structure and hierarchy.

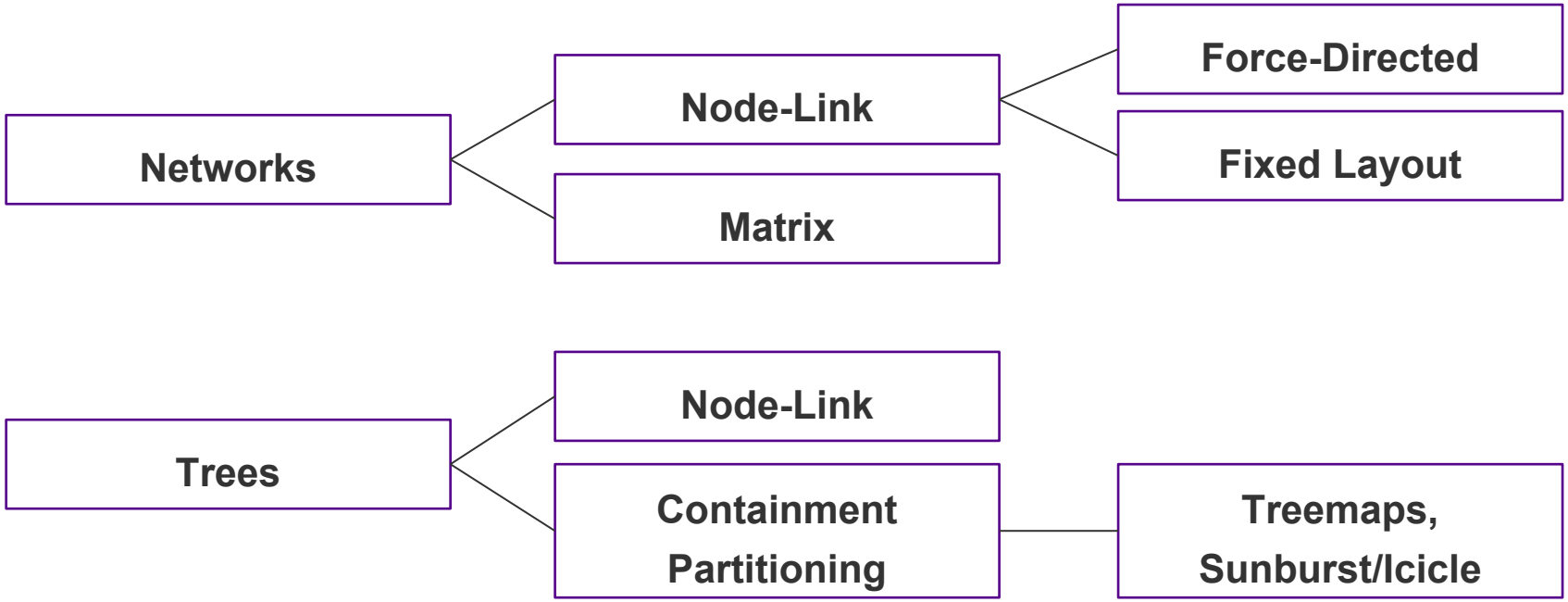
1. Introduction

In most areas of computer science, early research efforts focus on developing new, innovative techniques and algorithms. As the area matures, one can and should expect more critical, analytical studies to emerge. The area of information visualization is no different. Early research has largely focused on the development of innovative visualization techniques. Relatively little empirical study of the effectiveness of the visualizations has been conducted, however. In the opening plenary at the 1998 Information Visualization Symposium, George Robertson stressed the importance of empirical evaluation in this area. Careful empirical study of a visualization tool can help us to better understand its relative strengths and weaknesses, the tasks for which it is most appropriate, and can suggest improvements.

1071-5819/00/110663 + 32 $35.00/0

PART 02

Summary



PART 02

Summary

- Properties/Trade-offs
  - Clutter.
  - Scalability.
  - Structure.
  - Reordering/Aggregation.
  - Familiarity.
  - Intuitiveness.