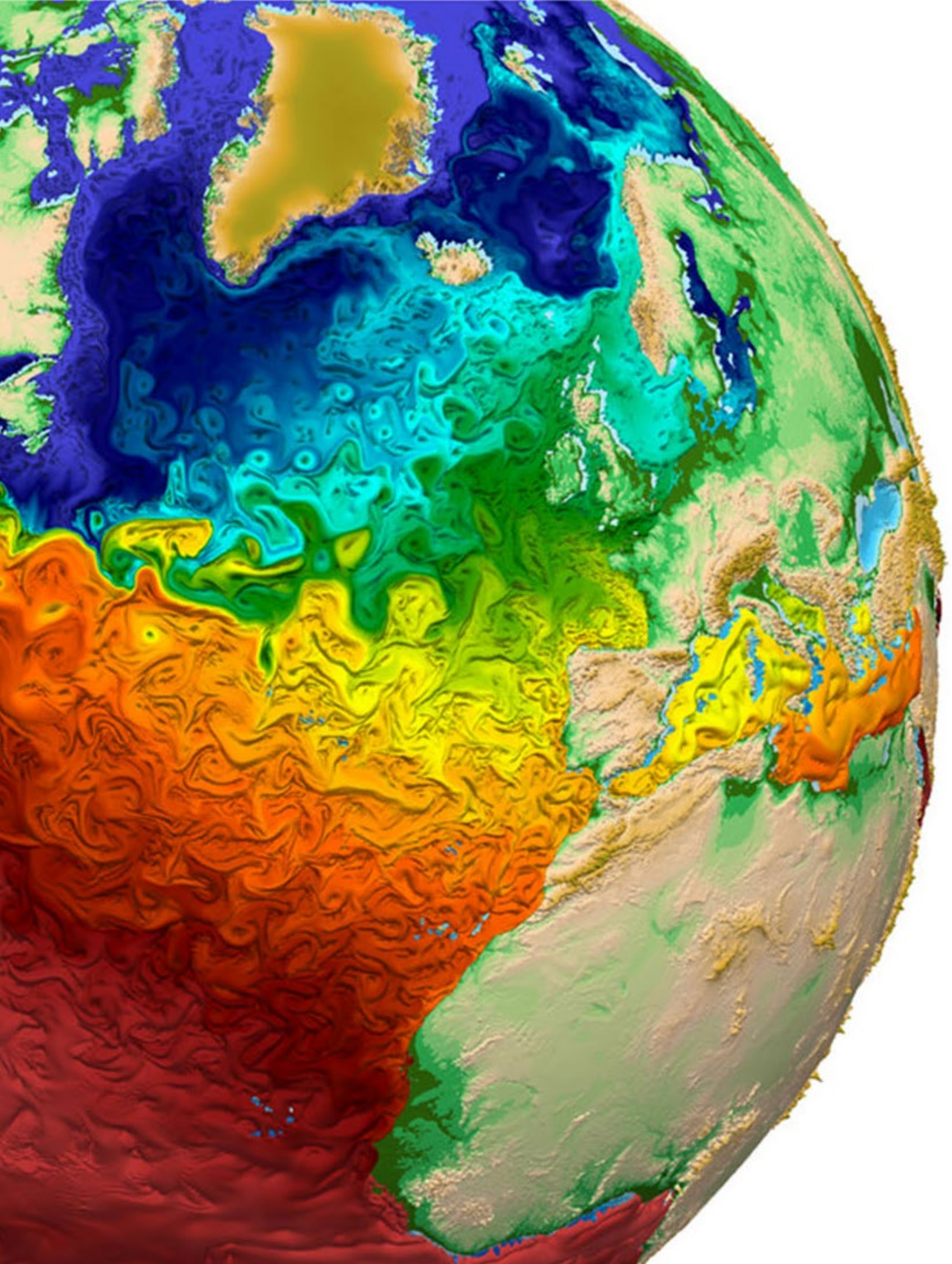


Visualizing Trees

Scientific Visualization

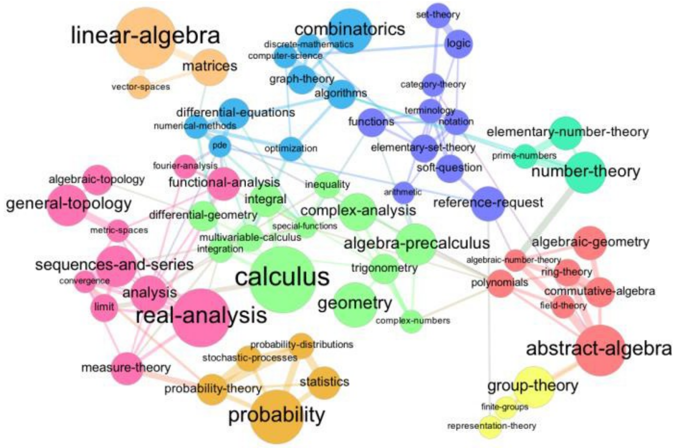
Professor Eric Shaffer

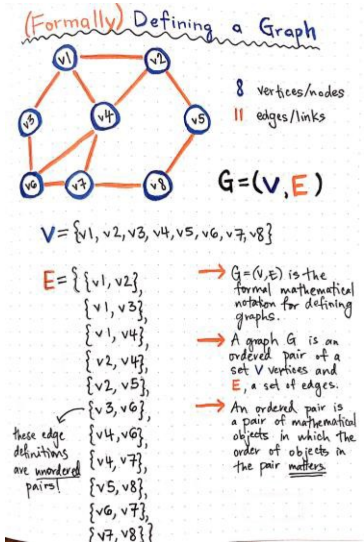
Network Visualization = Graph Visualization

Graph drawing is an area of mathematics and computer science combining methods from geometric graph theory and information visualization to derive two-dimensional depictions of graphs arising from applications such as social network analysis, cartography, linguistics, and bioinformatics.[1]

A drawing of a graph or **network diagram** is a pictorial representation of the vertices and edges of a graph.

- Wikipedia




(Formally) Defining a Graph

v1
v2
v3
v4
v5
v6
v7
v8

8 vertices/nodes
11 edges/links

G=(V, E)

V= {v1, v2, v3, v4, v5, v6, v7, v8}

E= { {v1, v2},
{v1, v3},
{v1, v4},
{v2, v4},
{v2, v5},
{v3, v6},
{v4, v6},
{v4, v7},
{v5, v8},
{v6, v7},
{v7, v8} }

these edge definitions are unordered pairs!

→ G=(V,E) is the formal mathematical notation for defining graphs.

→ A graph G is an ordered pair of a set V vertices and E, a set of edges.

→ An ordered pair is a pair of mathematical objects in which the order of objects in the pair matters.

node = vertex

link = edge

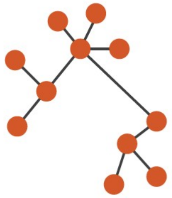
Network Visualization

Arrange networks and trees

Node–Link Diagrams

Connection Marks

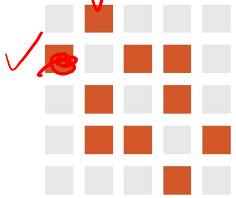
✓ NETWORKS ✓ TREES



Adjacency Matrix

Derived Table

✓ NETWORKS ✓ TREES

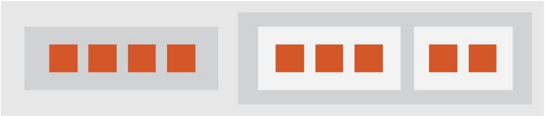


□	■	□	□	□
■	□	■	■	□
□	■	□	■	□
□	■	■	□	■
□	□	□	■	□

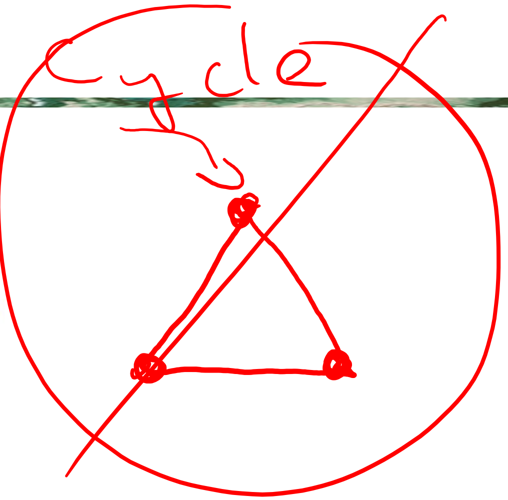
Enclosure

Containment Marks

✗ NETWORKS ✓ TREES



Trees

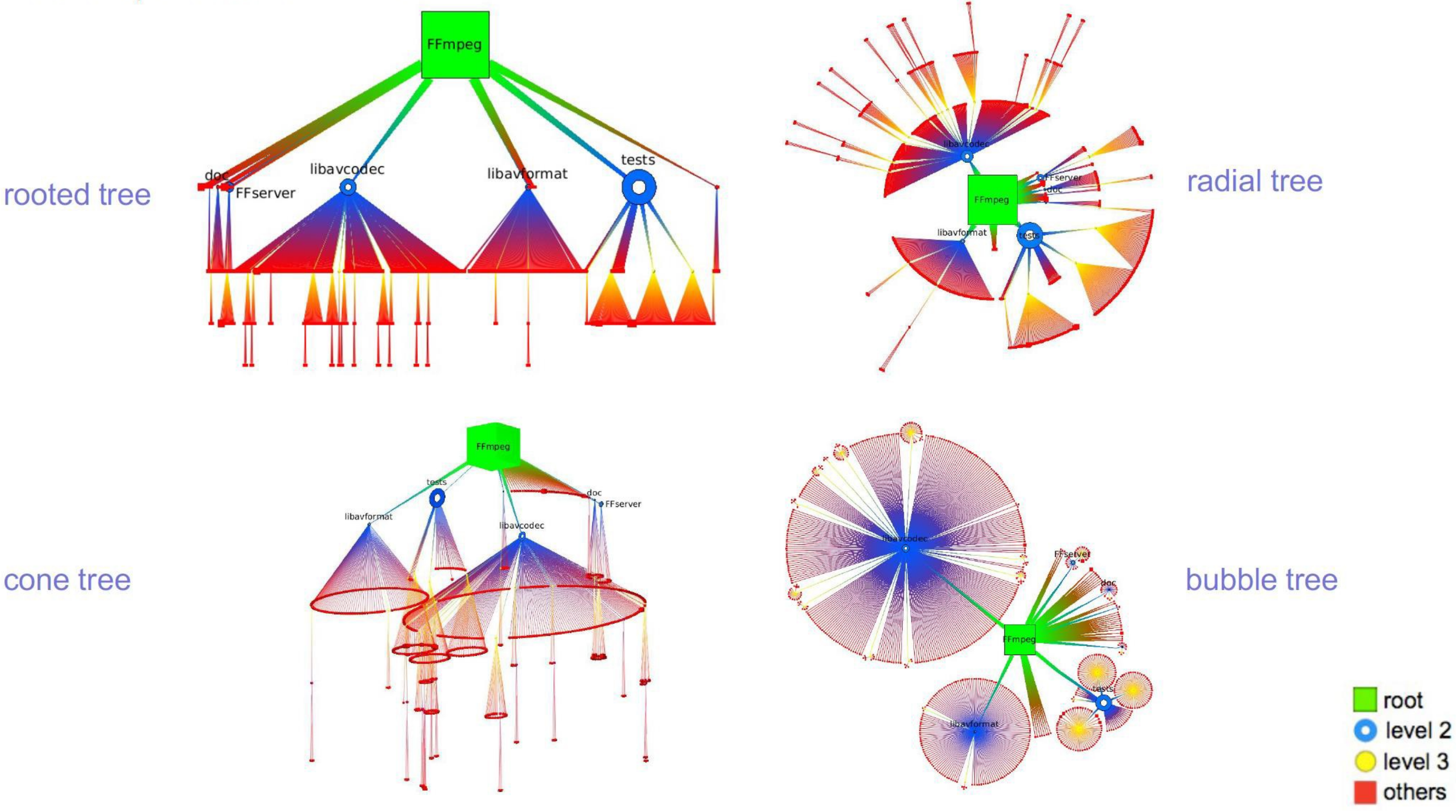


Trees are acyclic graphs

Examples

- icon size: folder size
- icon shape+color: level in tree

ffmpeg video code C library: 785 files, 42 folders



rooted tree

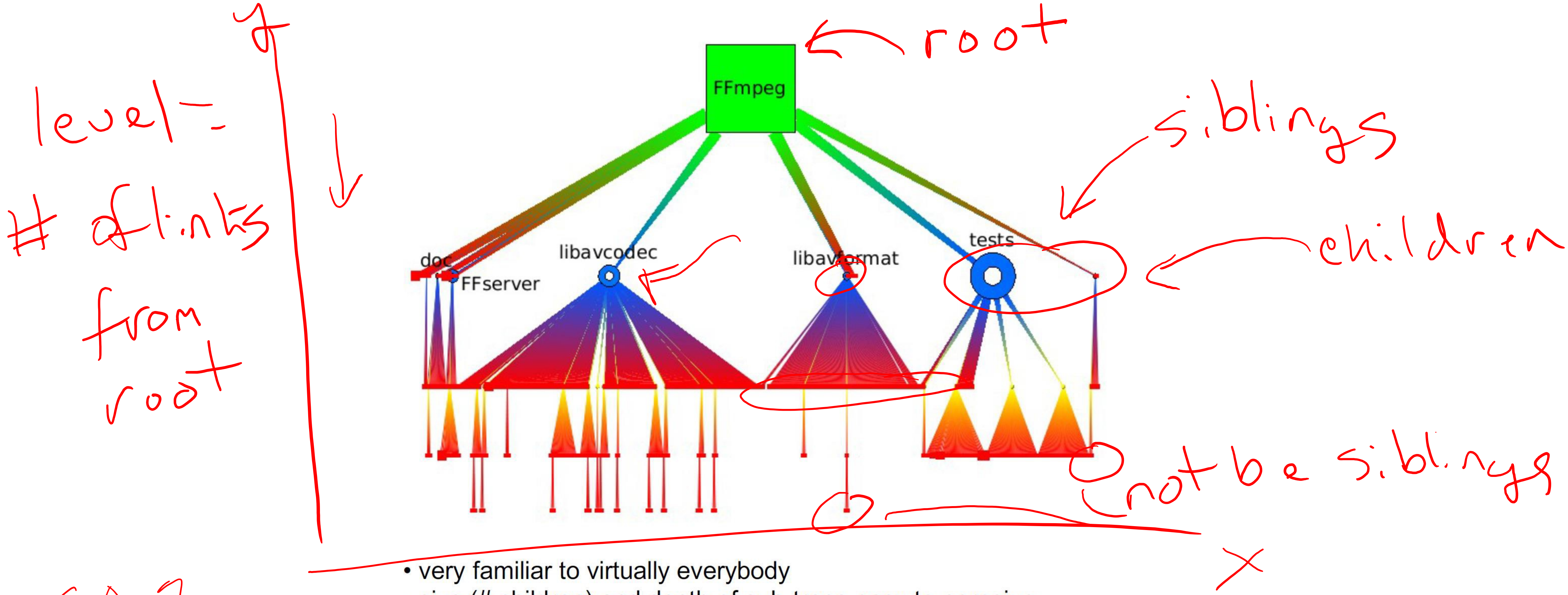
radial tree

cone tree

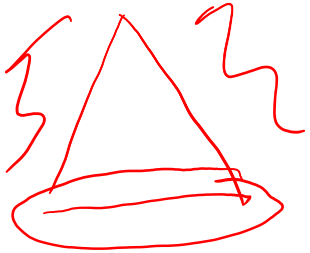
bubble tree

🟩 root
○ level 2
● level 3
🟥 others

Tree Layout: Rooted Layout

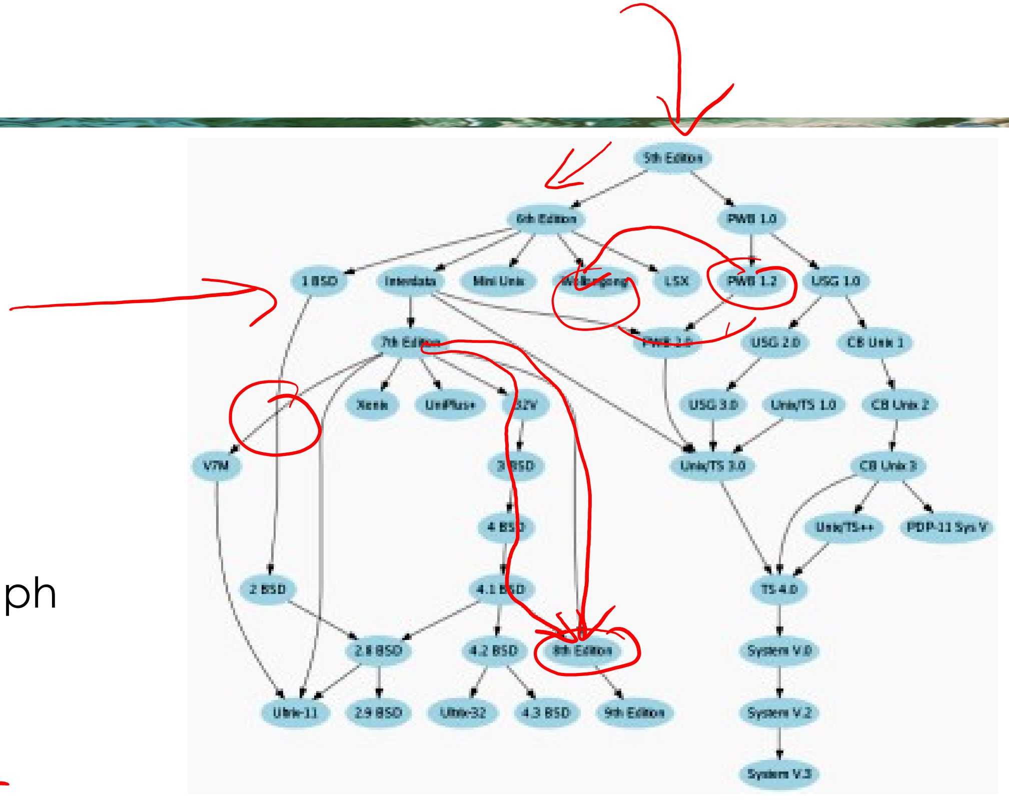


- very familiar to virtually everybody
- size (# children) and depth of sub-trees easy to perceive
- smooth edge shading → emphasize colors of small nodes
- unbalanced aspect ratios can occur → <span>limited scalability</span>



Directed Acyclic Graphs (DAGs)

- class hierarchies (multiple inheritance)
- organization structure (multiple bosses)
- also created from general graphs by removing cycles

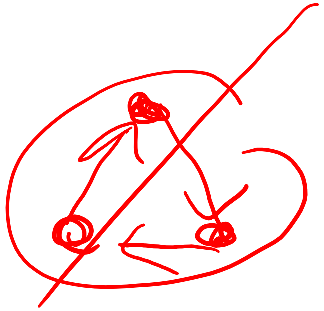


Hierarchical layout [Sugiyama et al ’81]

Algorithm:

- swap edges to eliminate cycles and get a directed (rooted) graph
- for every level, starting from root:
  - assign *y* coordinate as function of level
  - permute nodes on level to minimize edge lengths/crossings
- edges drawn as curves (splines) to minimize crossings

UNIX system history drawn with the GraphViz package (www.graphviz.org)

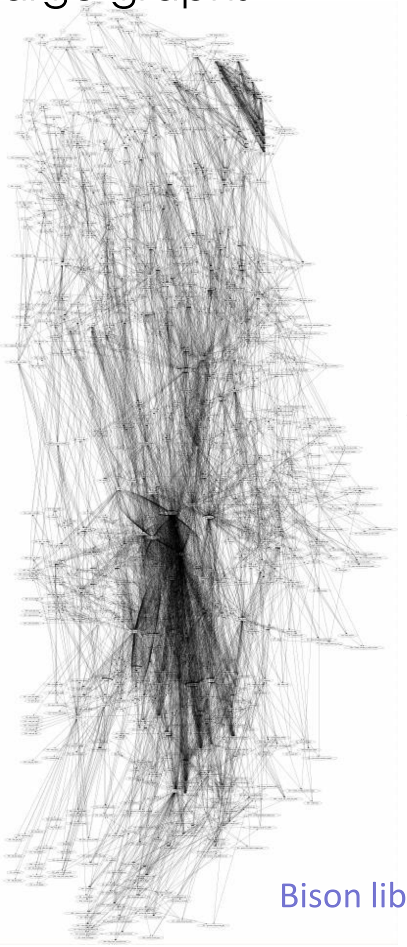


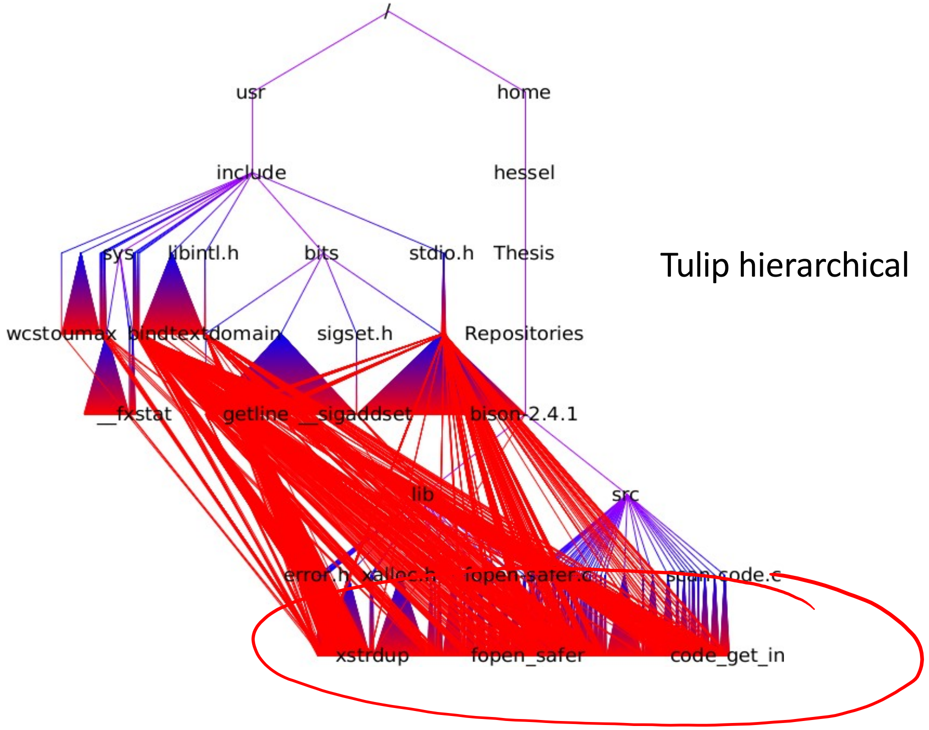
DAG Layout

Hierarchical layout scalability

- OK for < 1000 nodes or edges
- too many crossings and/or bad aspect ratios for large graphs
- we shall see later how to handle large graphs

Hierarchical DAG Layout





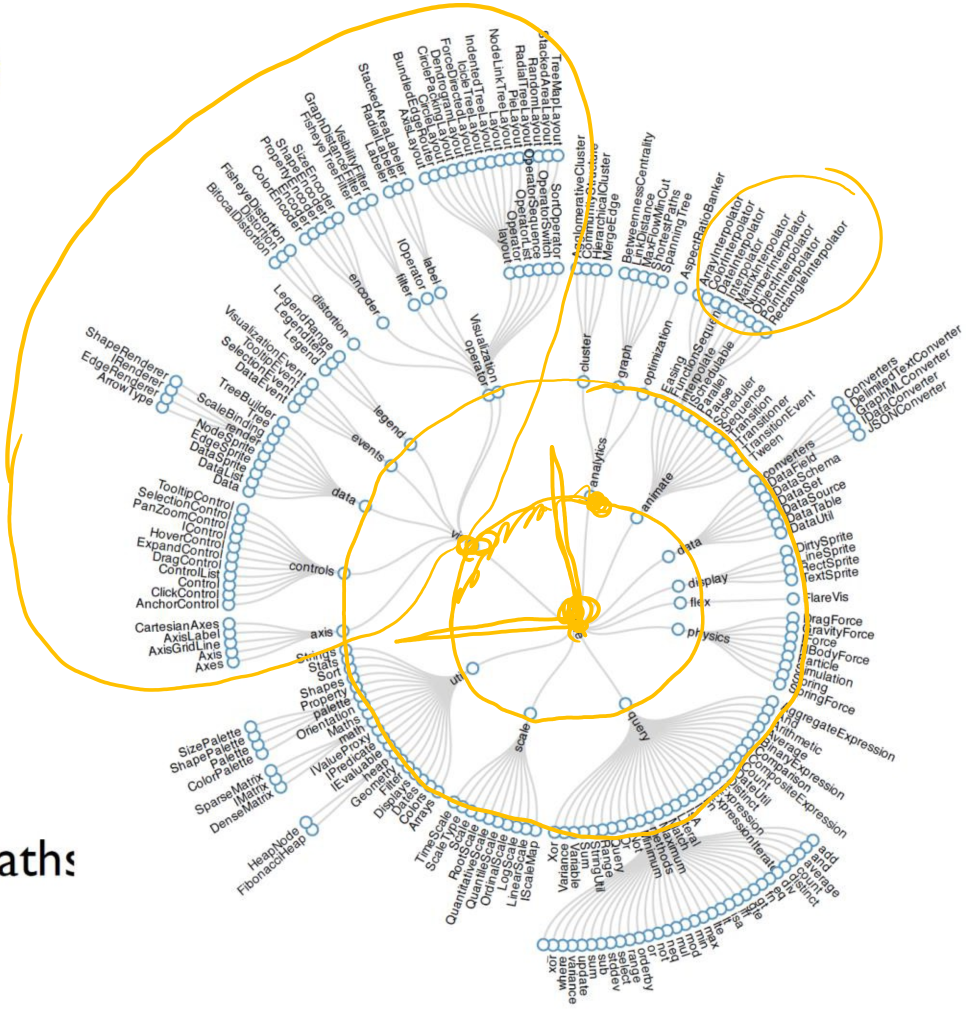
Tulip hierarchical DAG layout

Bison library call graph (3.214 nodes, 14.382 edges)

Radial Tree Layout

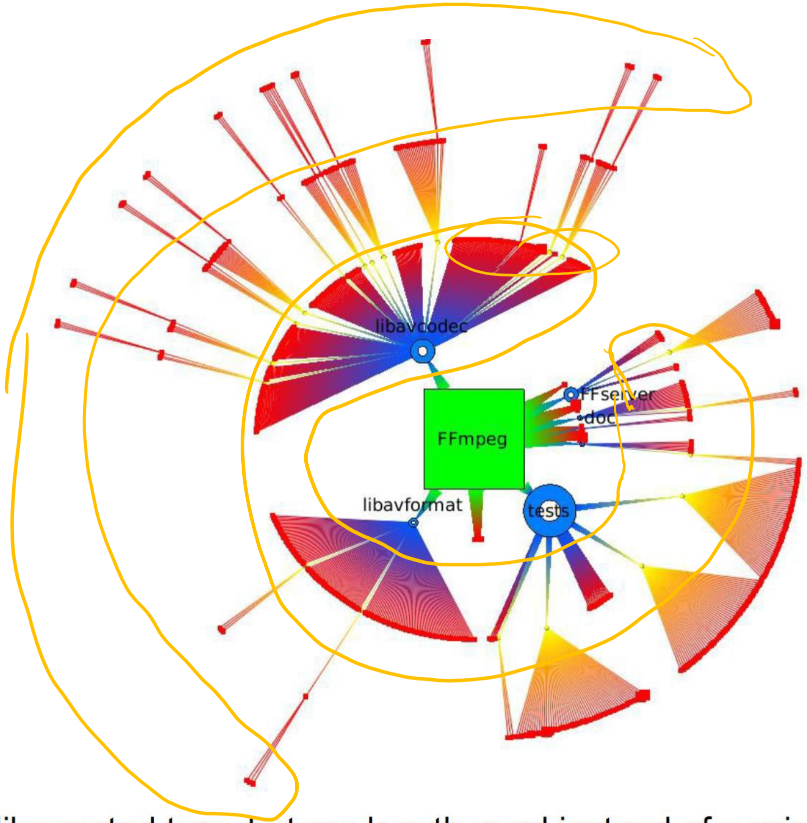
Idiom: **radial node-link tree**

- data
  - –tree
- encoding
  - –link connection marks
  - –point node marks
  - –radial axis orientation
    - angular proximity: siblings
    - distance from center: depth in tree
- tasks
  - –understanding topology, following paths
- scalability
  - –1K - 10K nodes



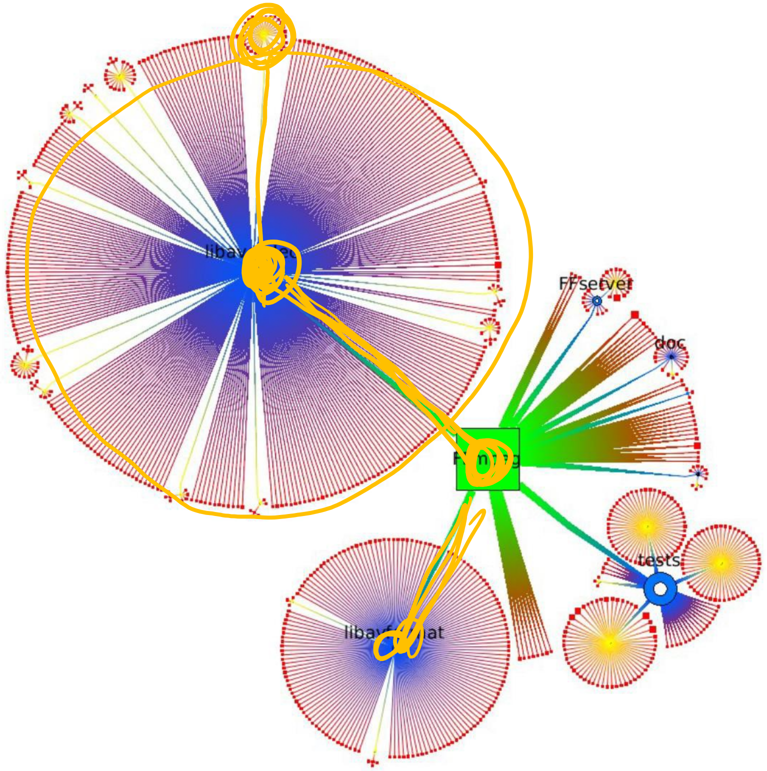
http://mbostock.github.com/d3/ex/tree.html

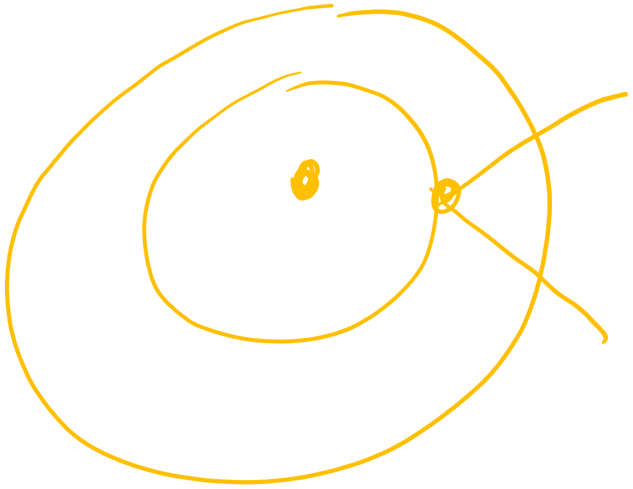
Tree Layout: Radial Layout



- like rooted tree, but arc-length used instead of x axis
- size (# children) and depth of sub-trees easy to perceive
- good aspect ratio guaranteed
- nodes close to root get less space

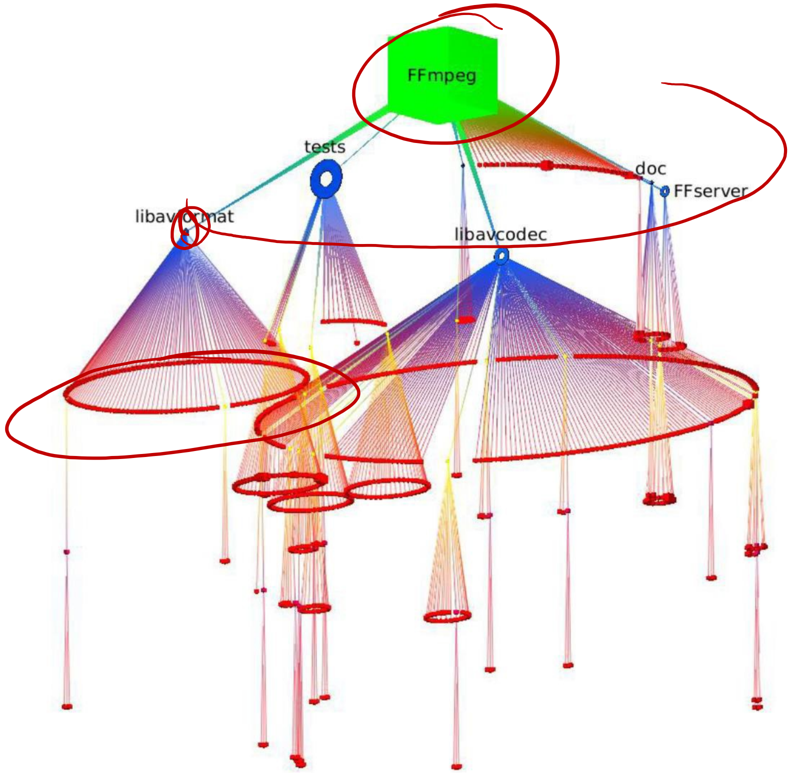
Tree Layout: Bubble Layout





- a subtree gets a full circle instead of a circle sector
- better spreading of the nodes for large trees
- variable edge lengths
- hard to distinguish node depth in the tree

Cone Tree



- a subtree gets a full cone instead of a sector / circle / line
- 3D effectively shows the tree depth
- combines bubble tree (seen from above) with rooted tree (seen from profile)
- 3D is tricky: occlusions, perspective shortening, navigation