

Software to Download:

**Cytoscape**

<https://cytoscape.org>



# BGGN 213

## Biological Network Analysis

Lecture 17

Barry Grant  
UC San Diego

<http://thegrantlab.org/bggn213>

# R 'Knowledge Check' Quiz Results

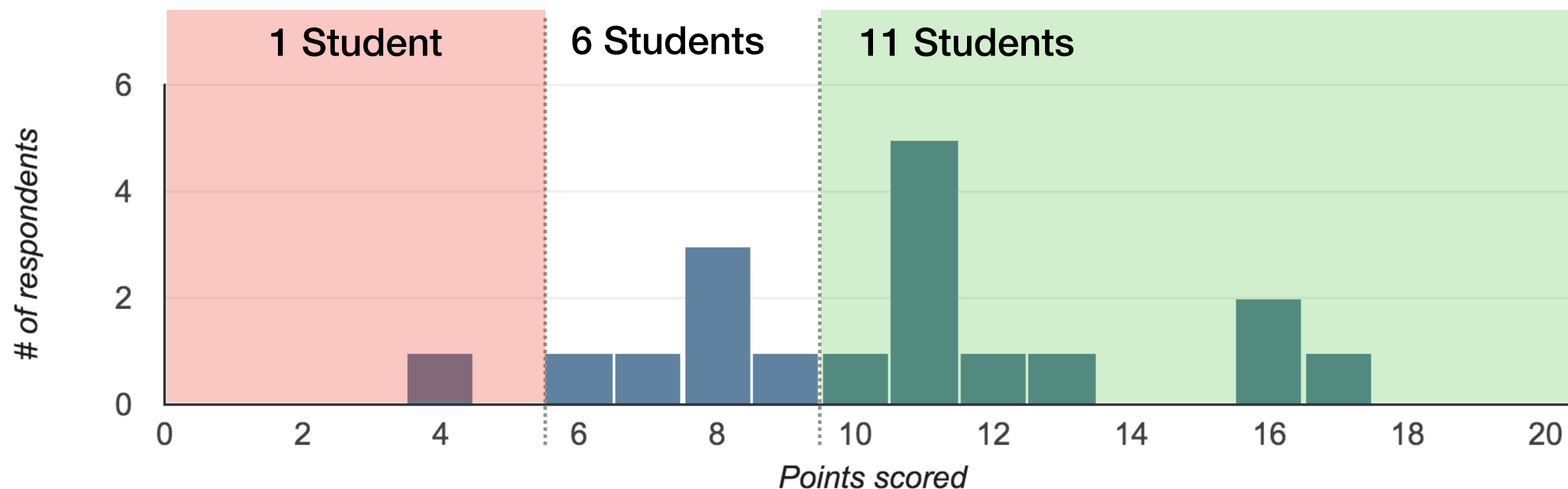
Winter 2020

**Average**  
10.5 / 20 points

**Median**  
11 / 20 points

**Range**  
4 - 17 points

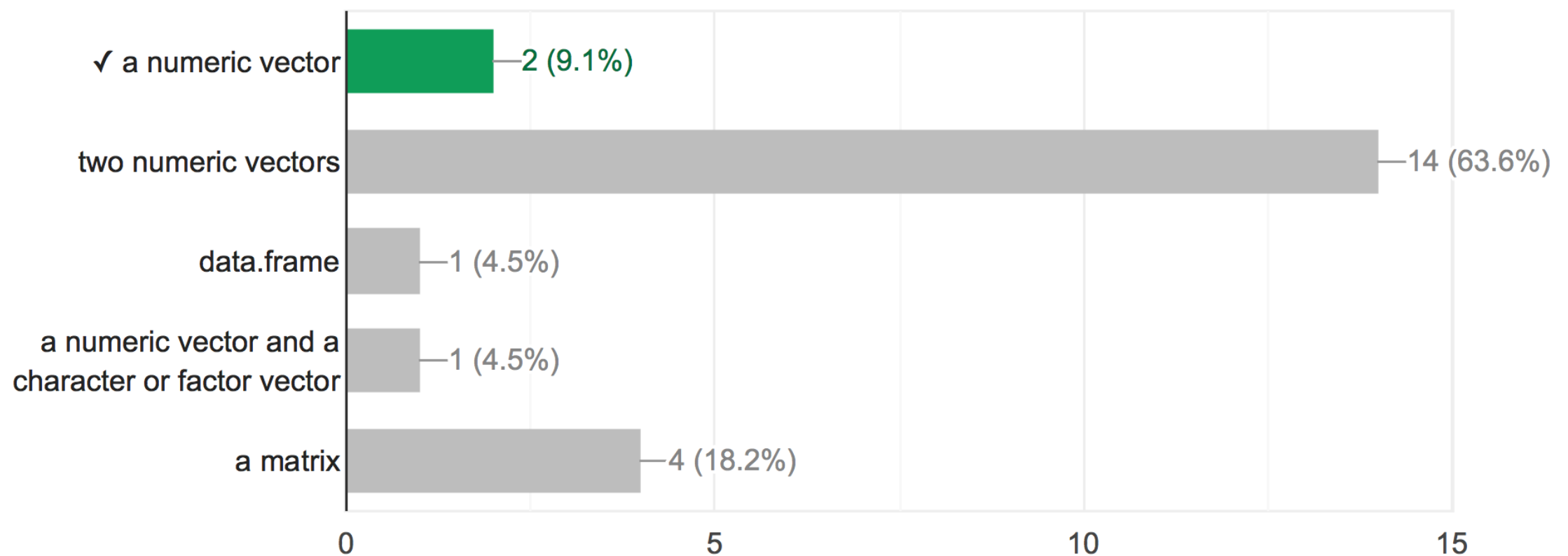
**18 Total Students**  
Total points distribution



# Frequently Missed Questions

What is the minimum input required to produce a single-layer scatter plot

2 / 22 correct responses

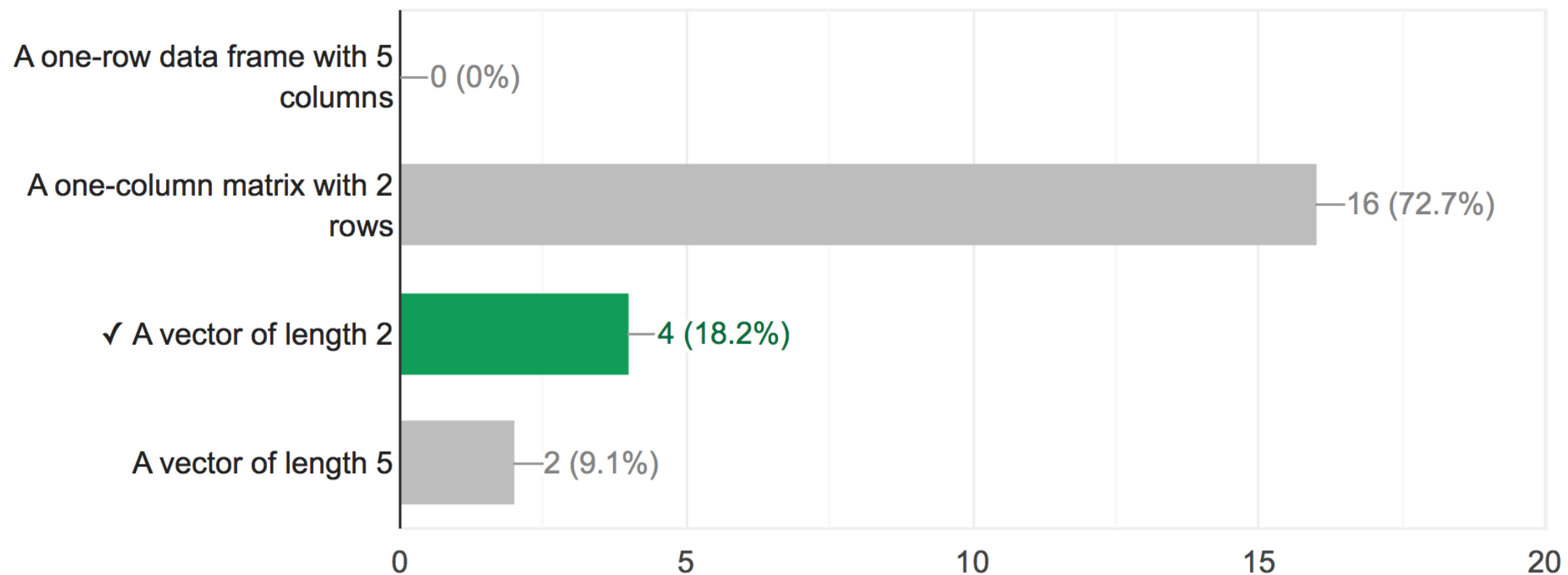




# Frequently Missed Questions

If I have a data frame "df" with 2 rows and 5 columns, what will `df[,1]` return?

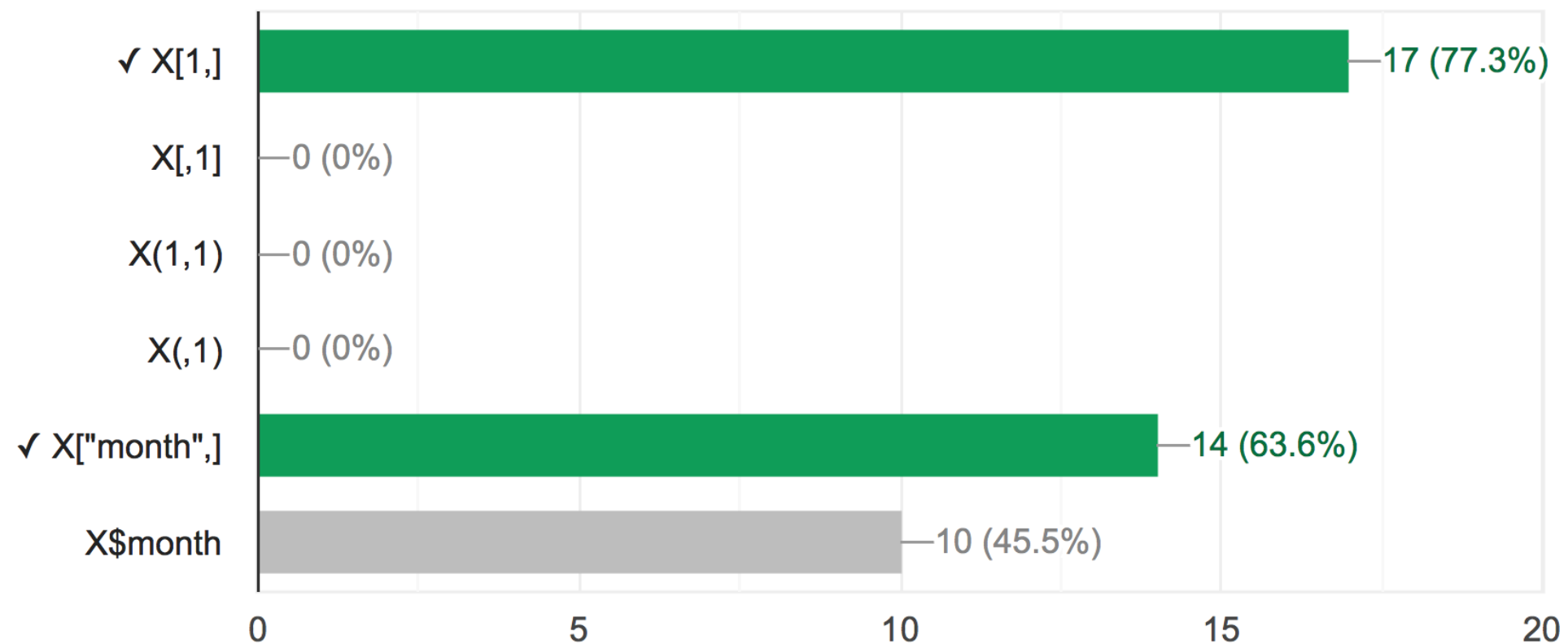
4 / 22 correct responses



# Frequently Missed Questions

Select the correct way(s) to extract a vector from a row in the data frame X.  
The name of the first row is month.

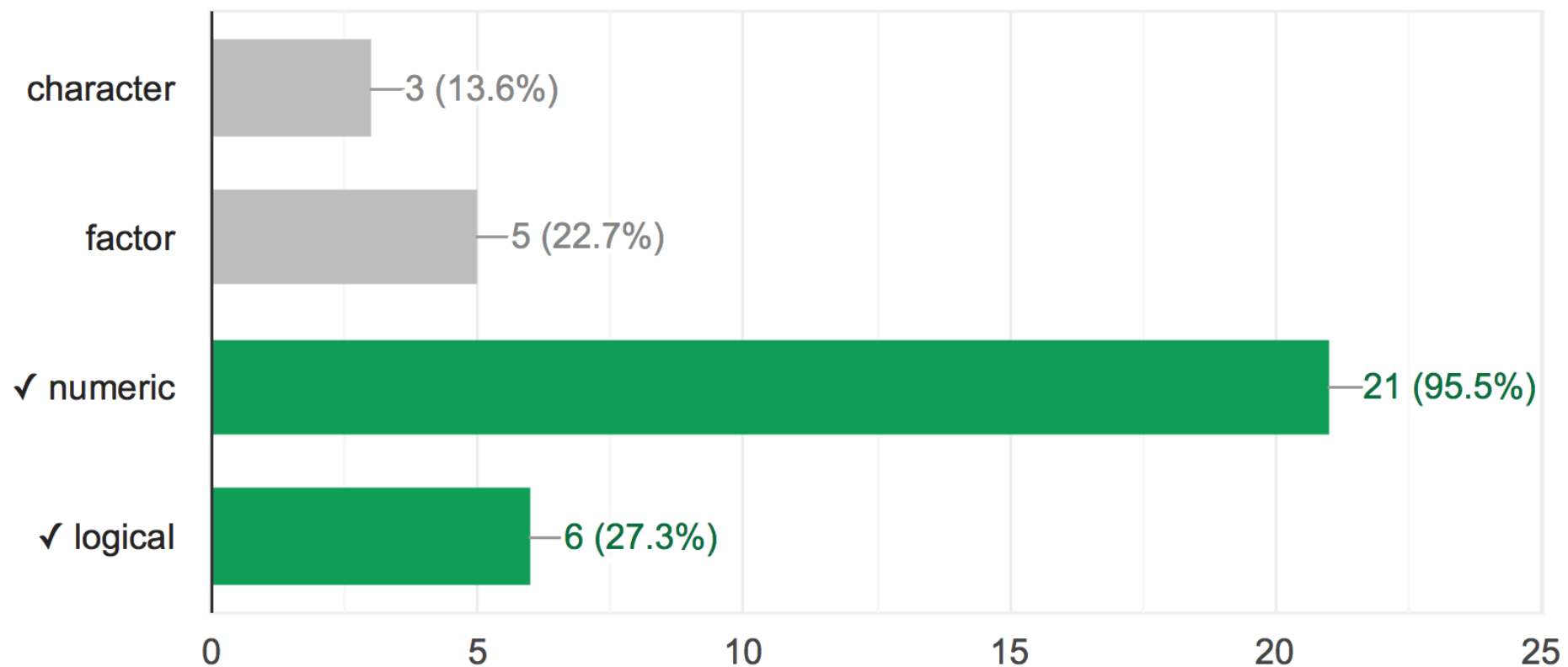
9 / 22 correct responses



# Frequently Missed Questions

`jal <- function(x) {y=x^2; return(y)}` #If I want to execute `jal(hmk)`, what kind of data type can `hmk` be without `r...n` error message? Check all that apply

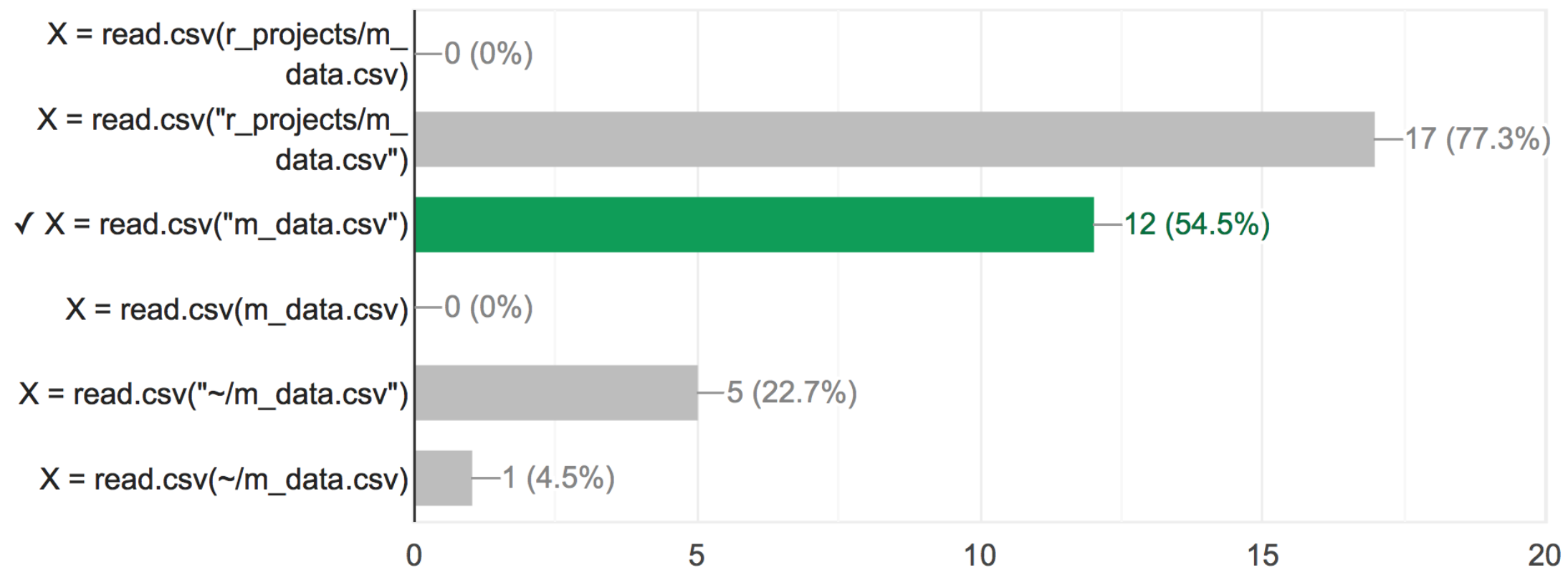
5 / 22 correct responses



# Frequently Missed Questions

The working directory is in the "r\_projects" folder, and it contains the file m\_data.csv. Select the correct way(s) to read m\_data into X.

4 / 22 correct responses

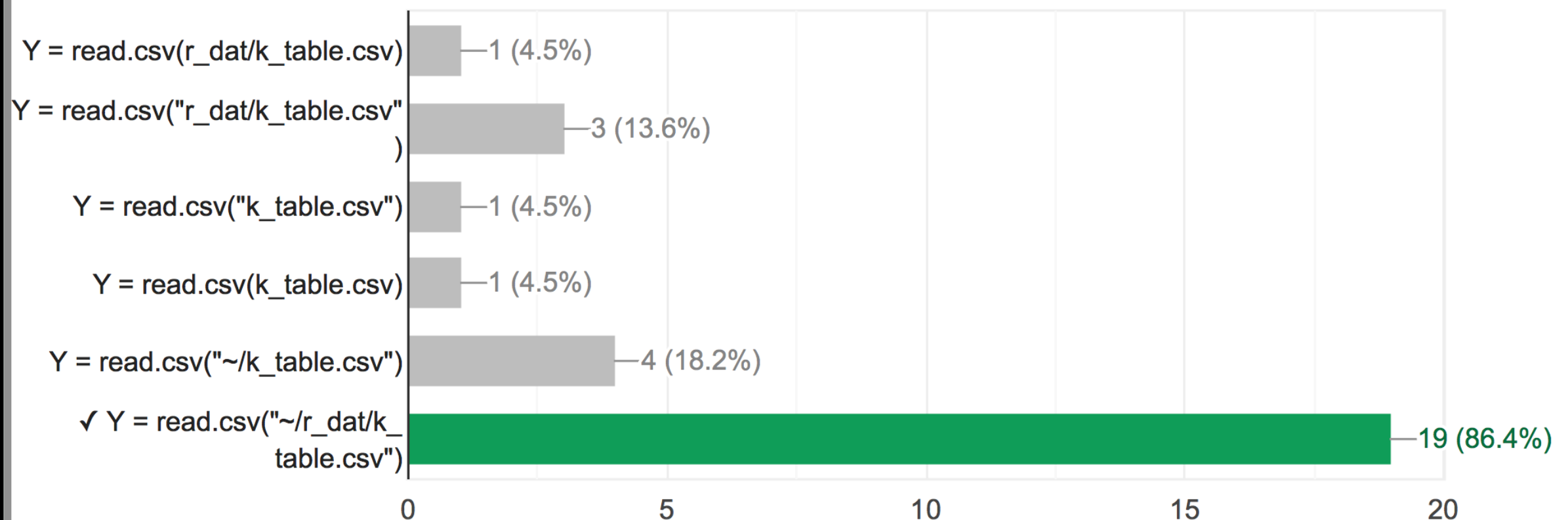




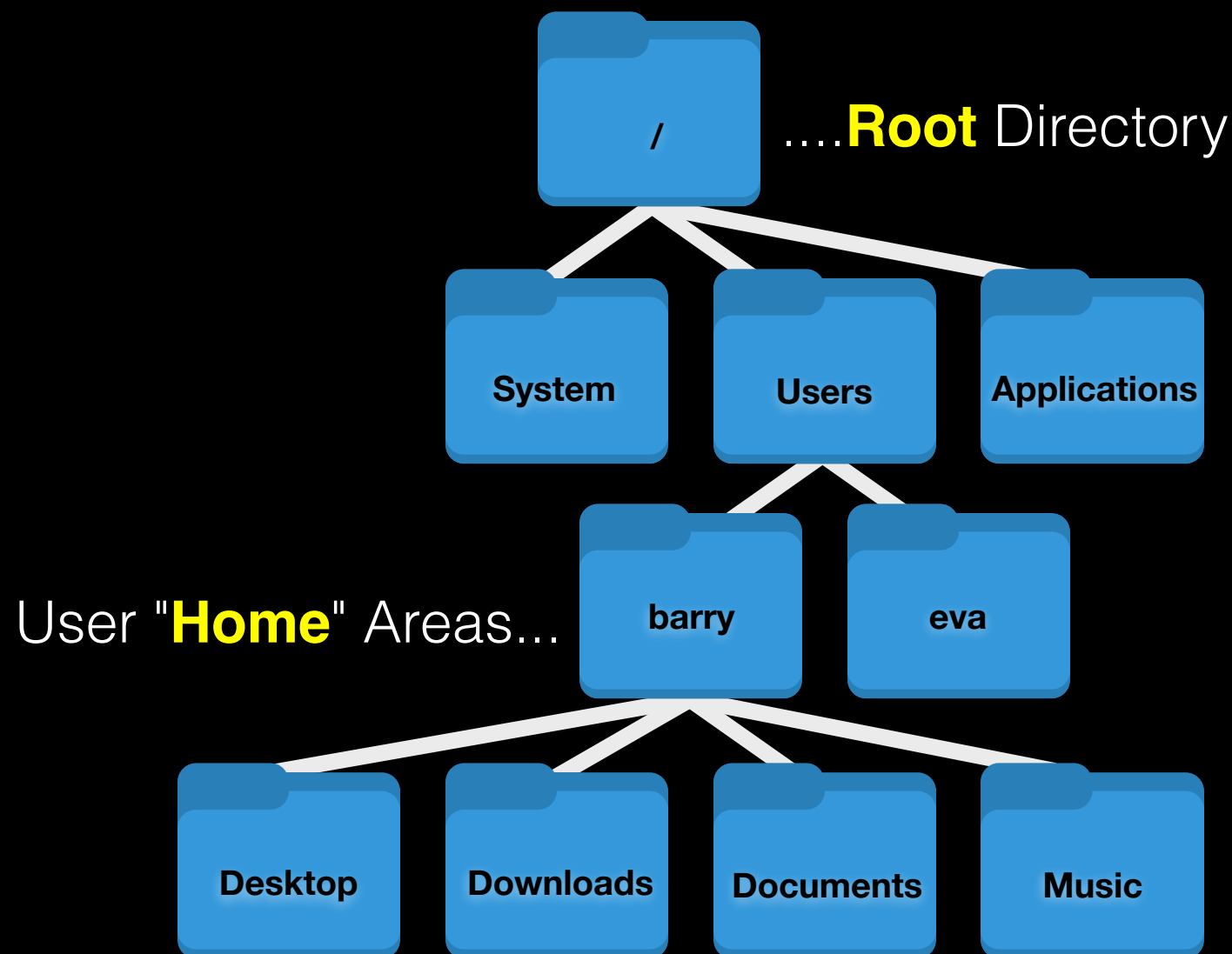
# Frequently Missed Questions

The "r\_dat" folder is not in your working directory but it is in your home directory. It contains the file k\_table.csv...rect way(s) to read k\_table.csv into Y.

15 / 22 correct responses



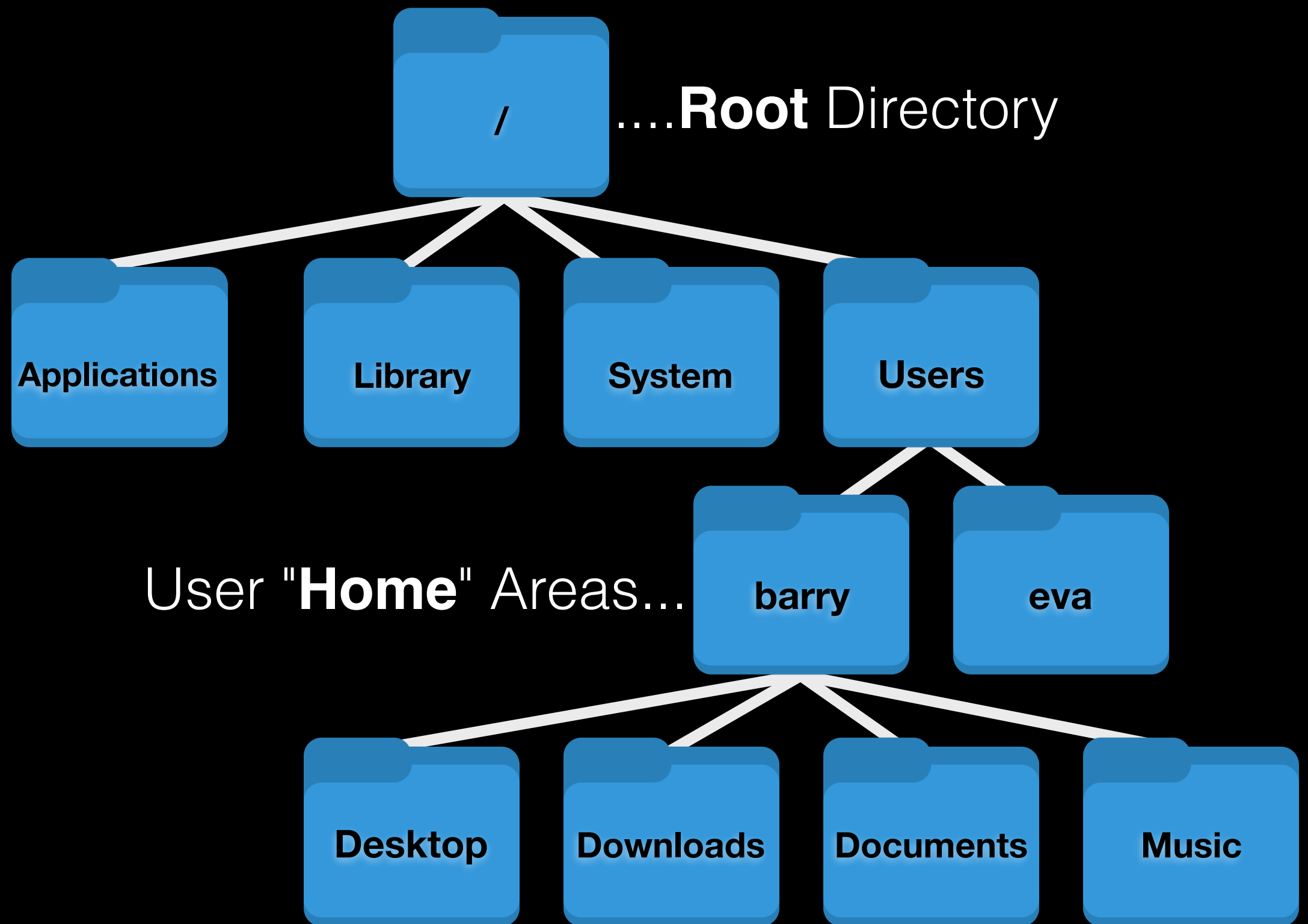
Information in the **file system** is stored in files, which are stored in **directories** (a.k.a. folders). Directories can also store other directories, which forms a **directory tree**.



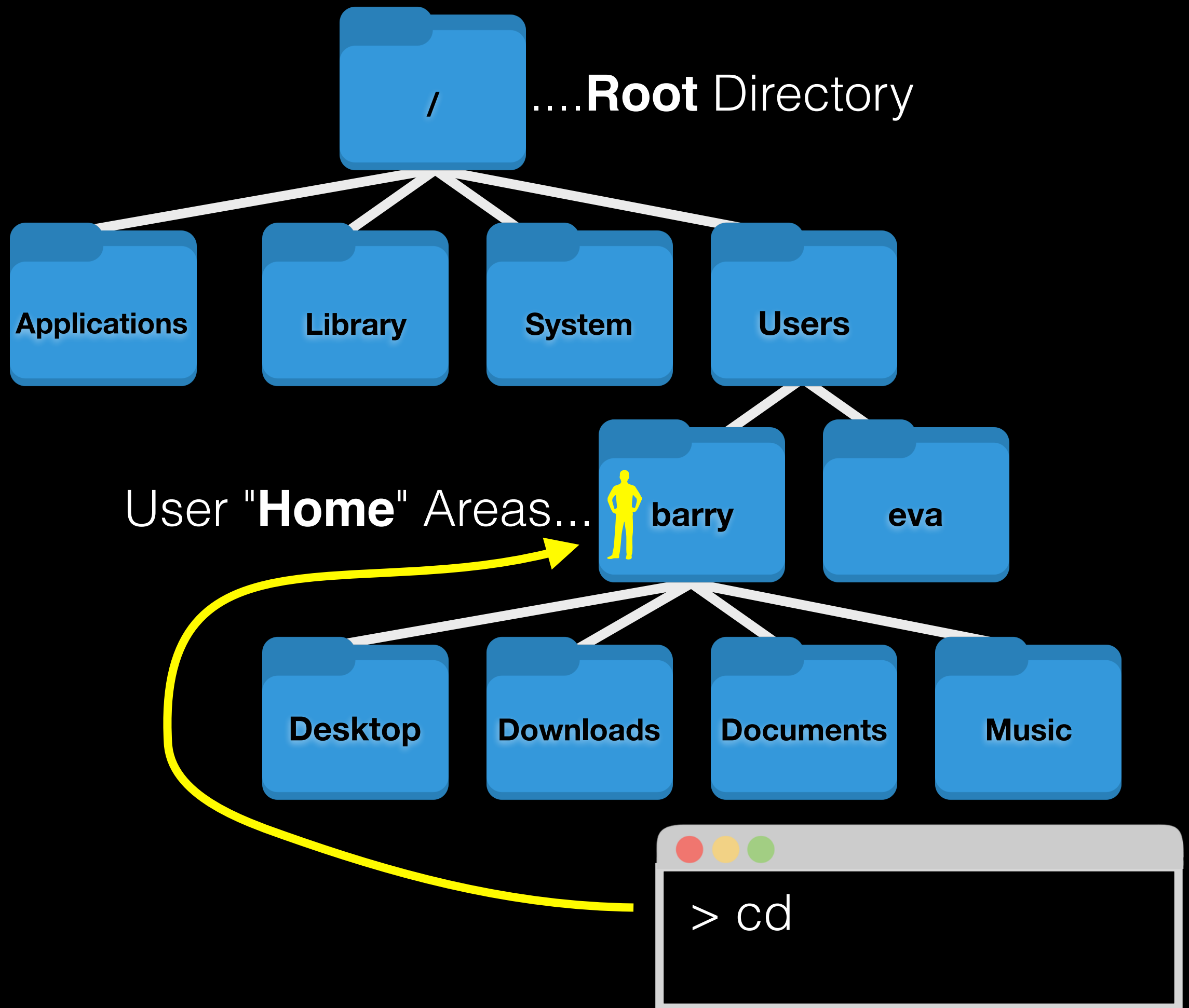
The forward slash character **/** is used to represent the **root** directory of the whole file system, and is also used to separate directory names. E.g. **/Users/barry/Desktop/myfile.csv**

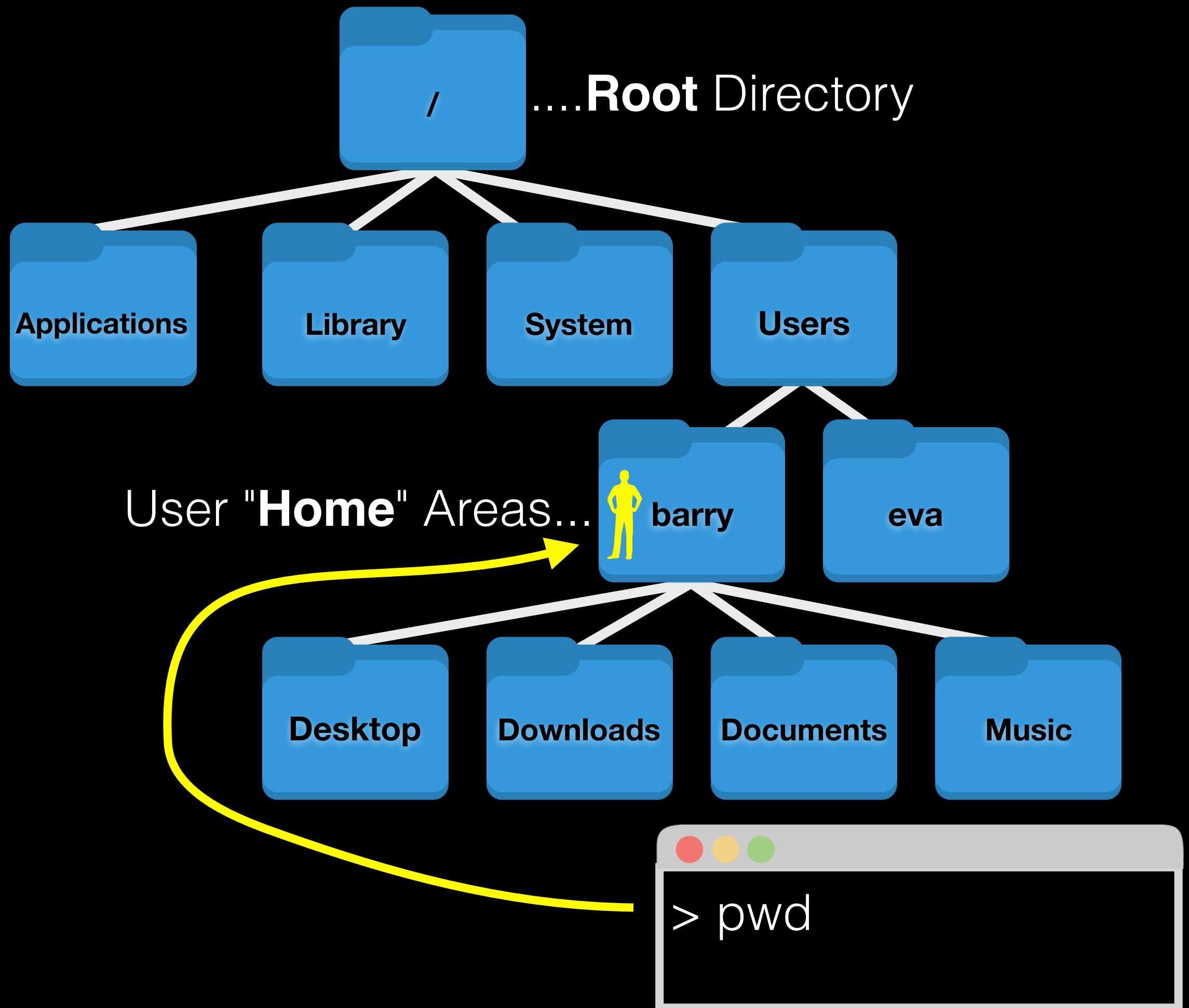
# UNIX Basics: Using the filesystem

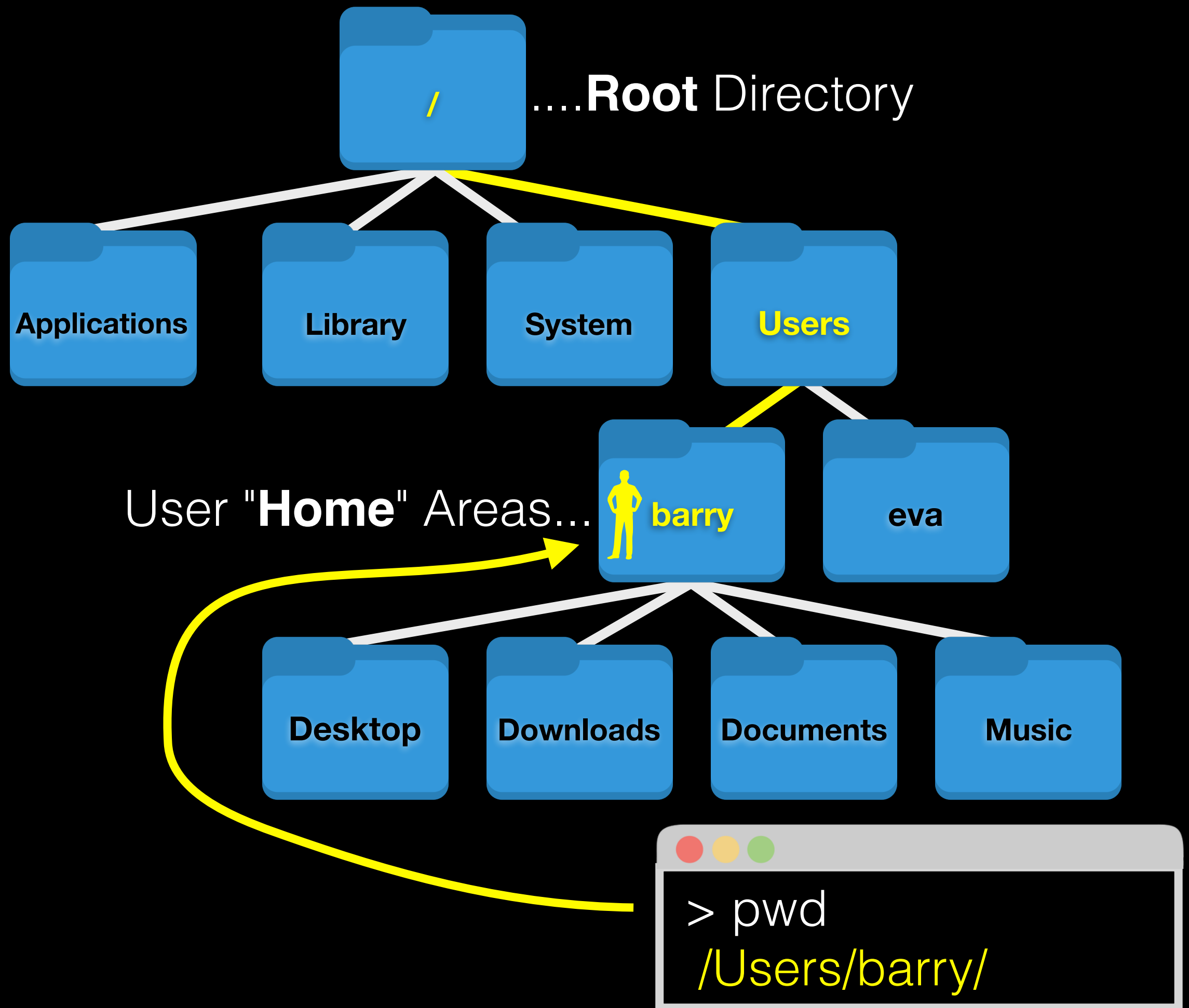
<b>ls</b>	List files and directories
<b>cd</b>	Change directory (i.e. move to a different 'folder')
<b>pwd</b>	Print working directory (which folder are you in)
<b>mkdir</b>	<u>Ma</u> <u>Ke</u> a new <u>DI</u> <u>Re</u> ctories
<b>cp</b>	<u>Co</u> <u>P</u> y a file or directory to somewhere else
<b>mv</b>	<u>Mo</u> <u>V</u> e a file or directory (basically rename)
<b>rm</b>	<u>Re</u> <u>M</u> ove a file or directory

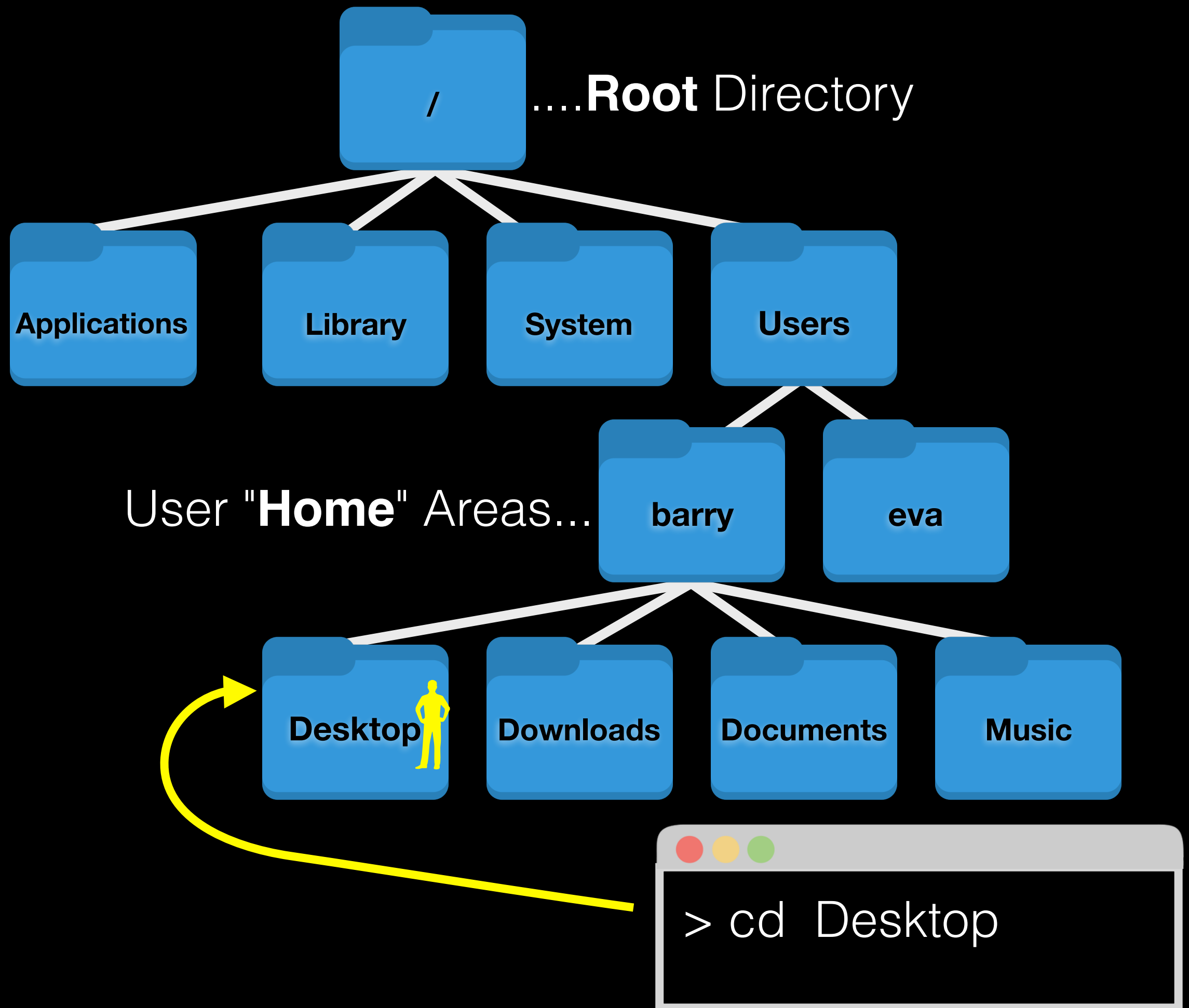




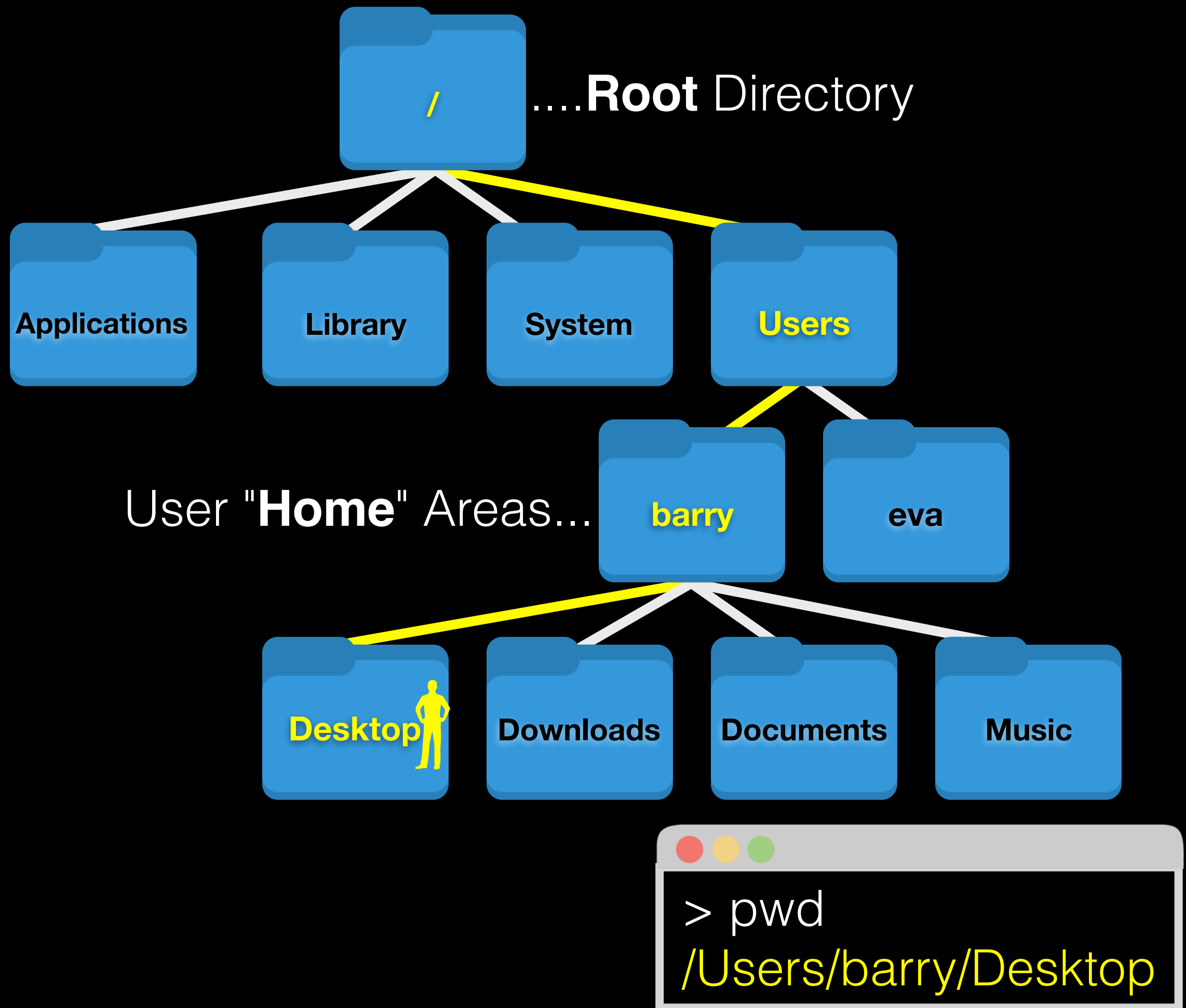


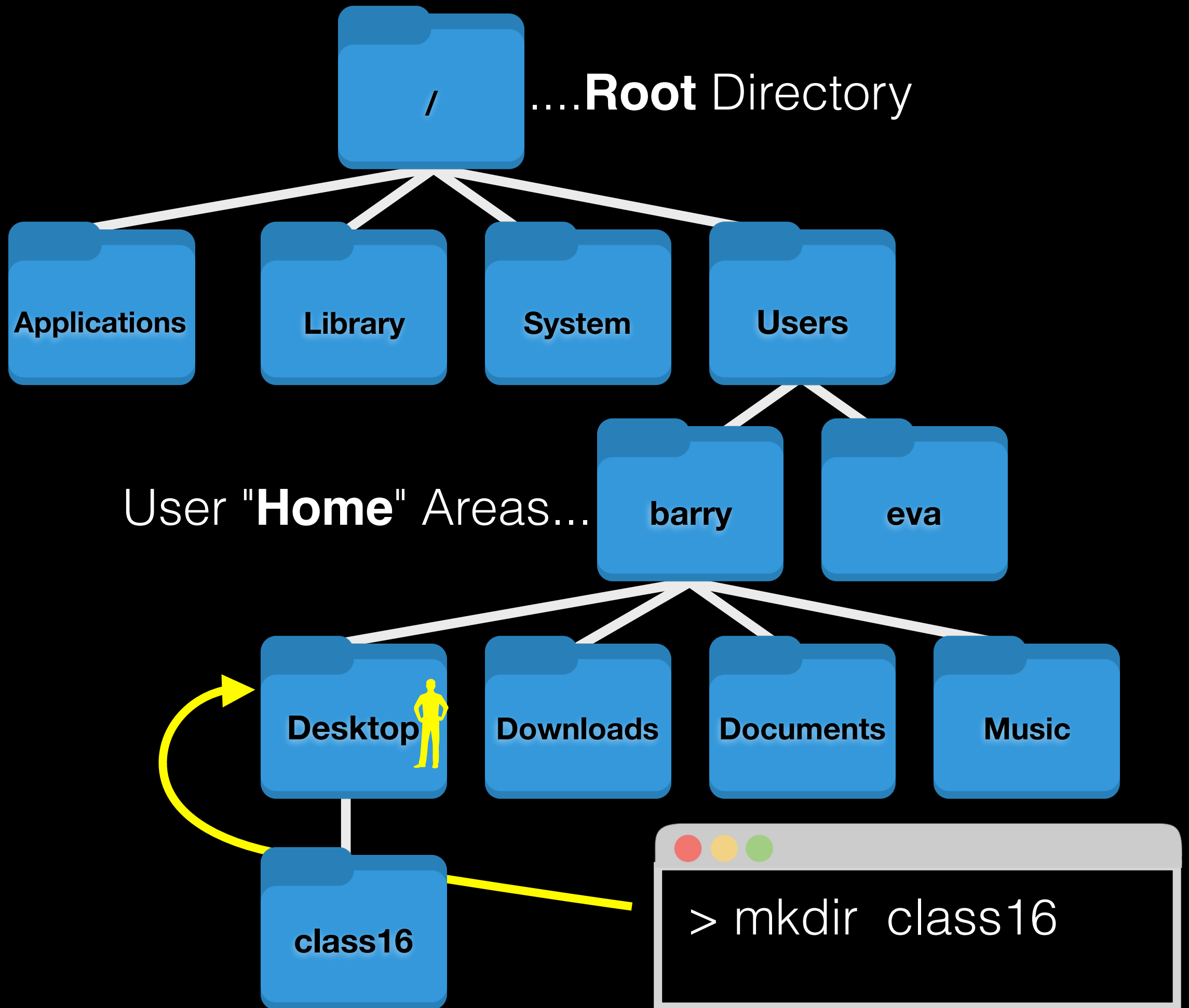


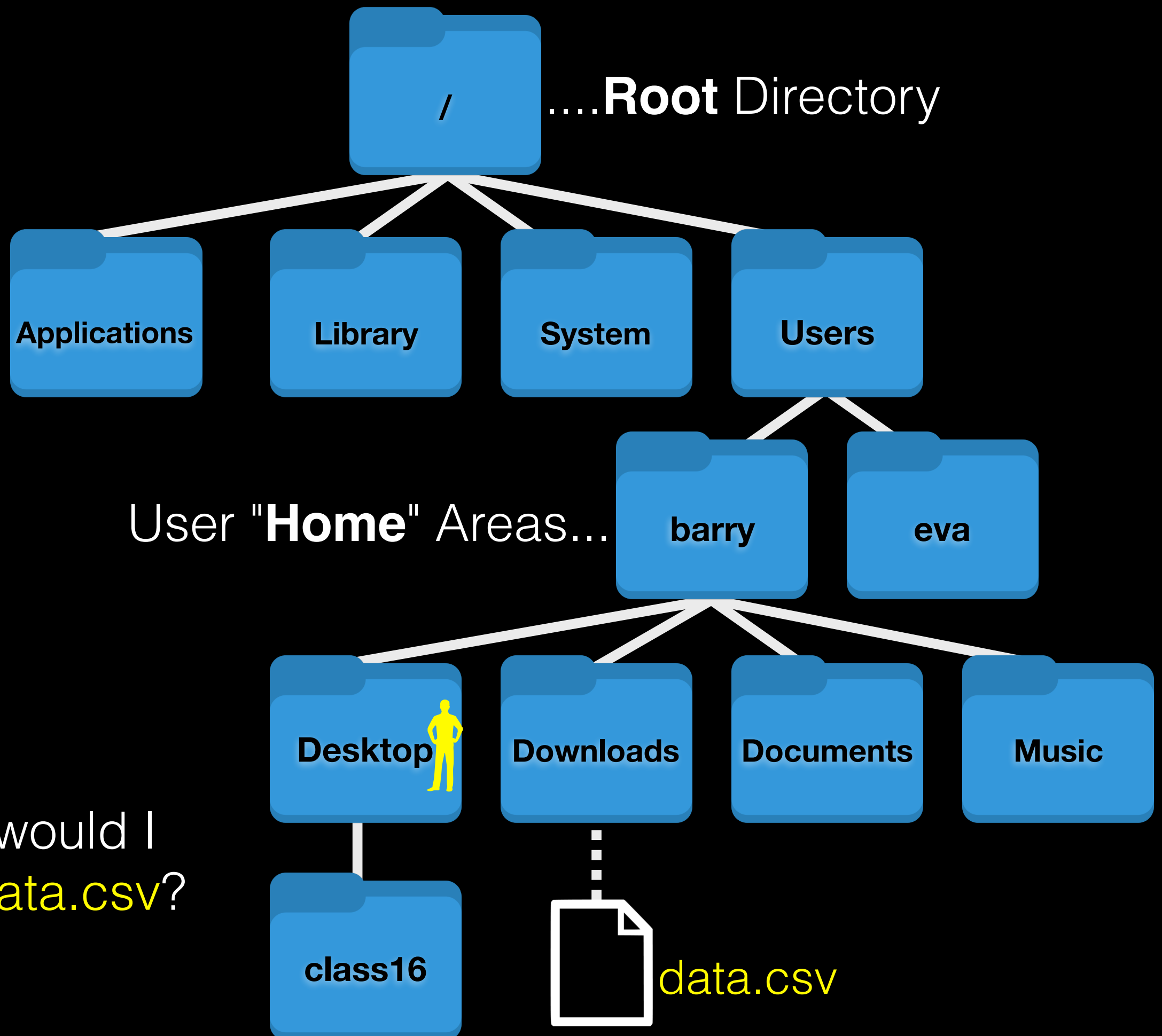




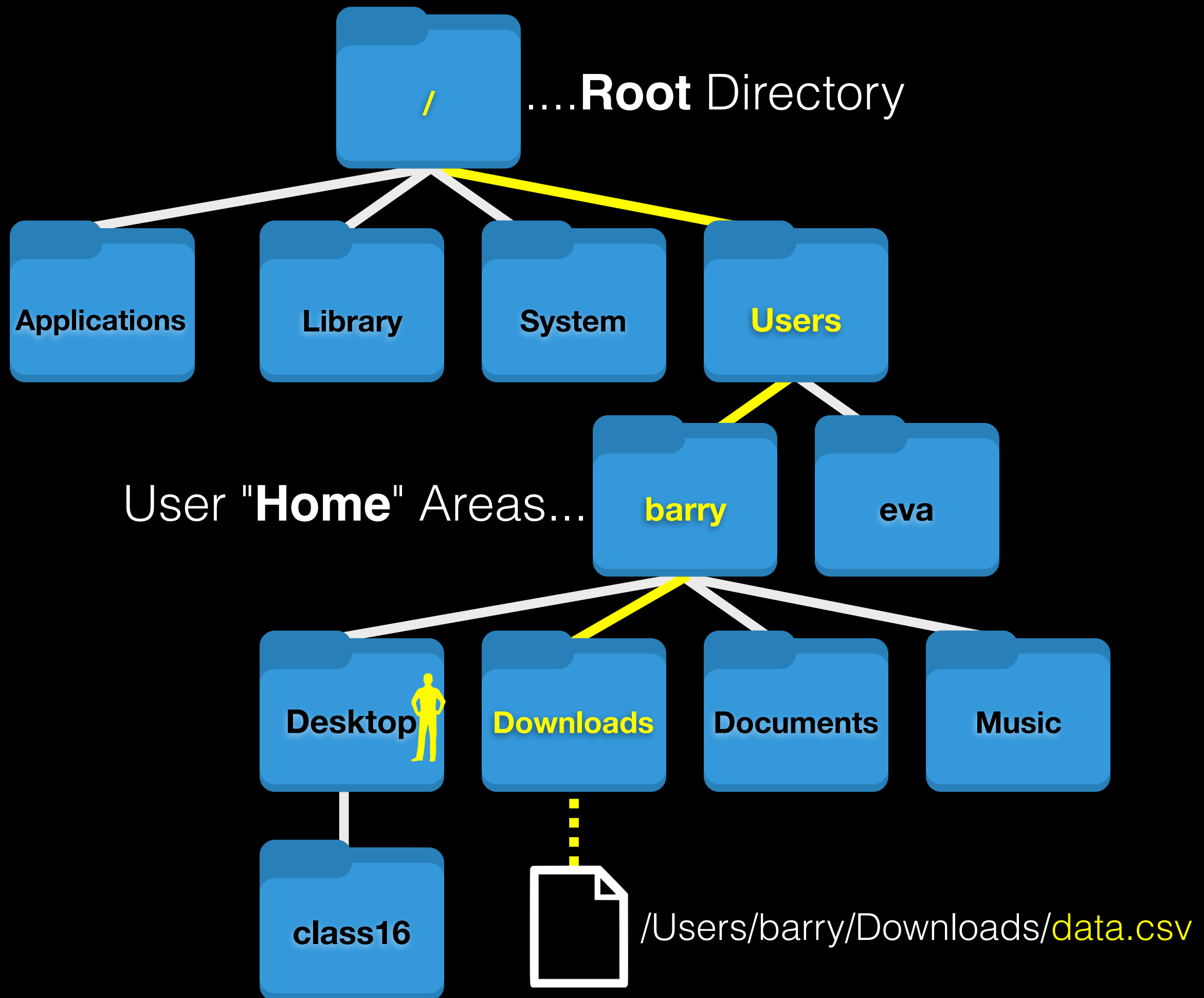




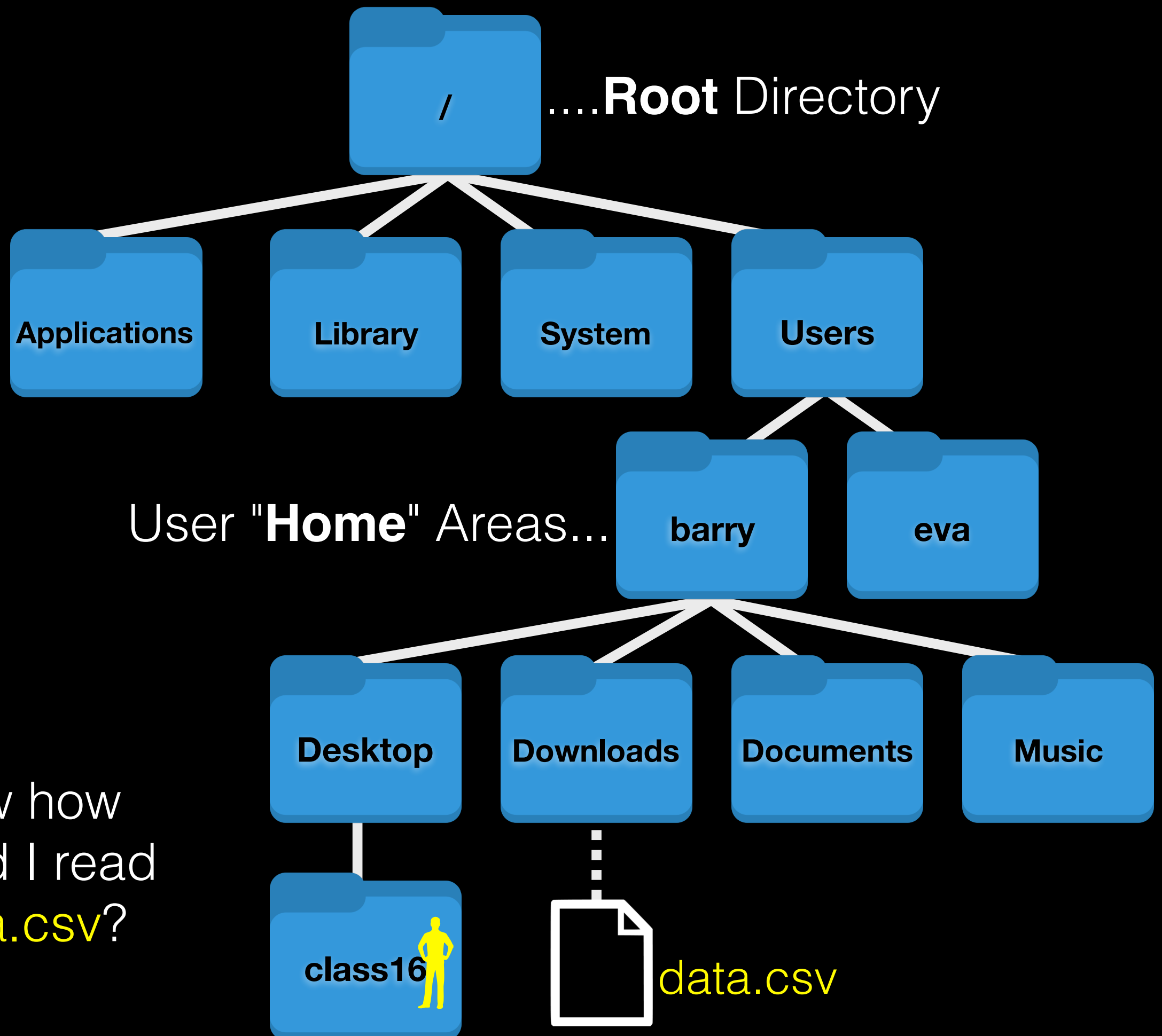




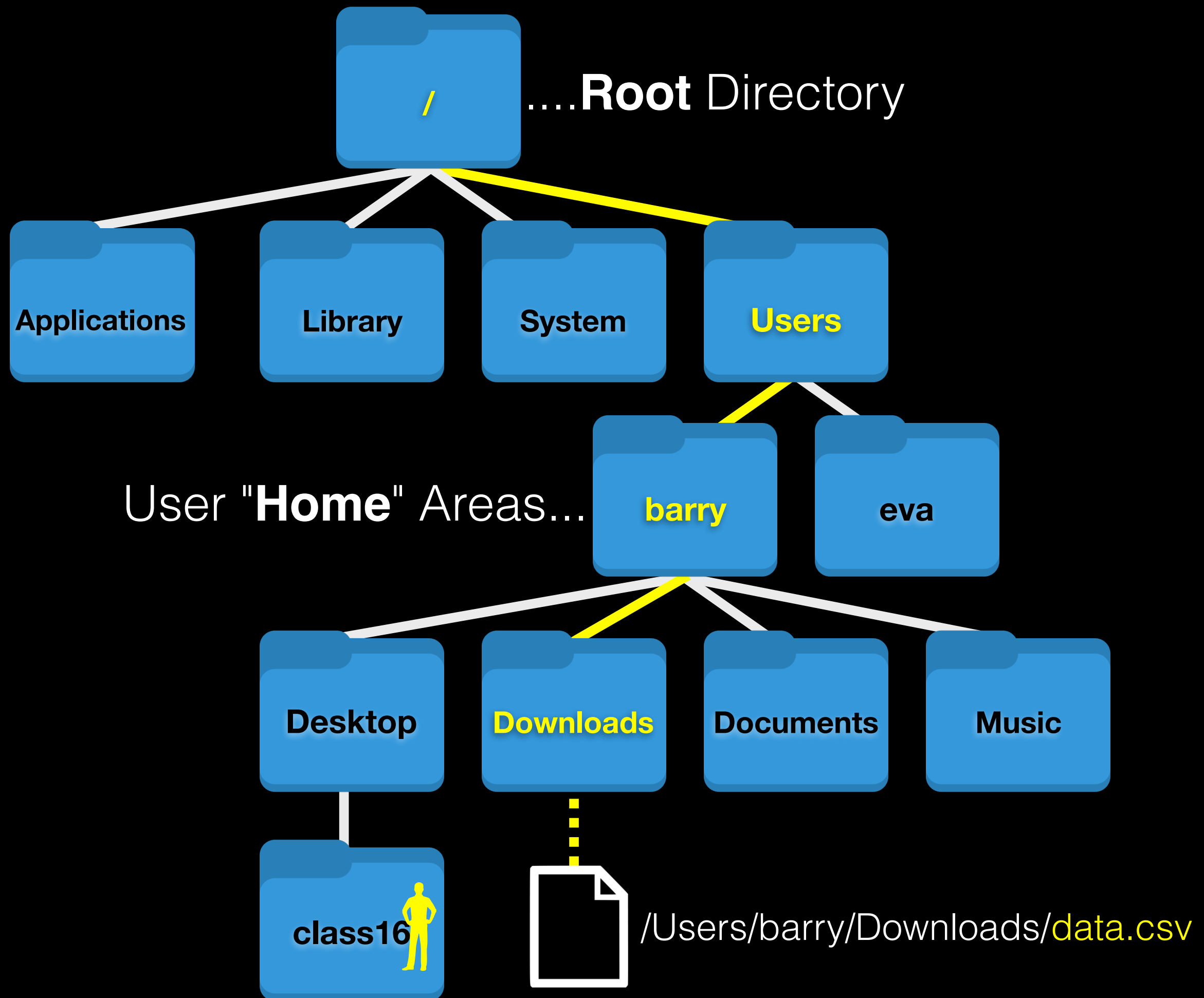
How would I  
read **data.csv**?







Now how  
would I read  
**data.csv**?



# Side Note: File Paths

- An **absolute path** specifies a location from the root of the file system. E.g. **/Users/barry/Downloads/data.csv**  
**~/Downloads/data.csv**
- A **relative path** specifies a location starting from the current location. E.g. **../data.csv**

<b>.</b>	Single dot ' <b>.</b> ' (for current directory)
<b>..</b>	Double dot ' <b>..</b> ' (for parent directory, back one dir in our tree)
<b>~</b>	Tilda ' <b>~</b> ' (shortcut for your home directory)
<b>[Tab]</b>	Pressing the <b>tab</b> key can autocomplete names

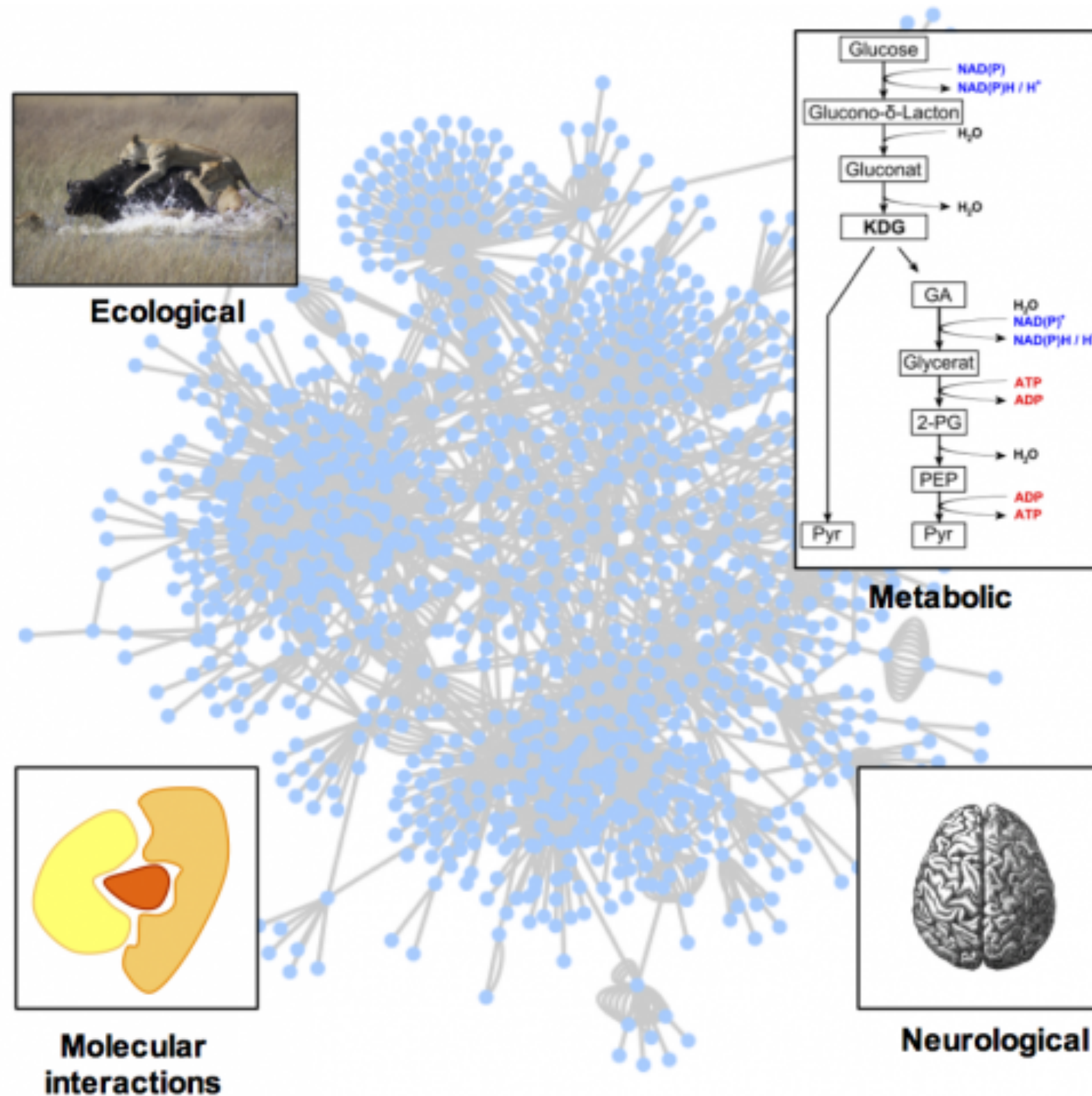
# TODAYS MENU:

- ▶ **Network introduction**
- ▶ **Network visualization**
- ▶ **Network analysis**
- ▶ **Hands-on:**
  - Cytoscape and R (igraph) software tools for network visualization and analysis

# TODAYS MENU:

- ▶ **Network introduction**
- ▶ **Network visualization**
- ▶ **Network analysis**
- ▶ **Hands-on:**  
Cytoscape and R (igraph) software tools  
for network visualization and analysis

# Networks can be used to model many types of biological data



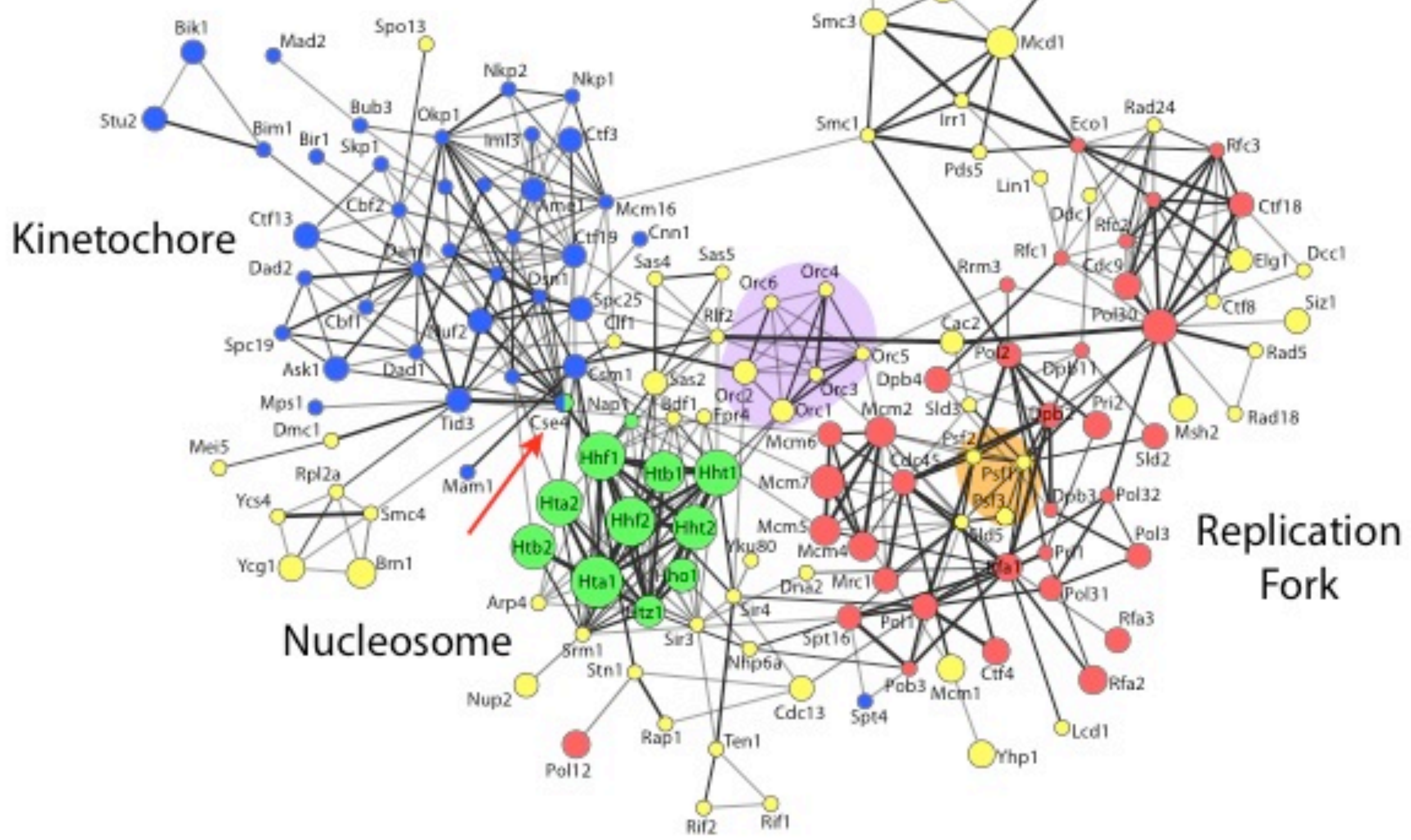
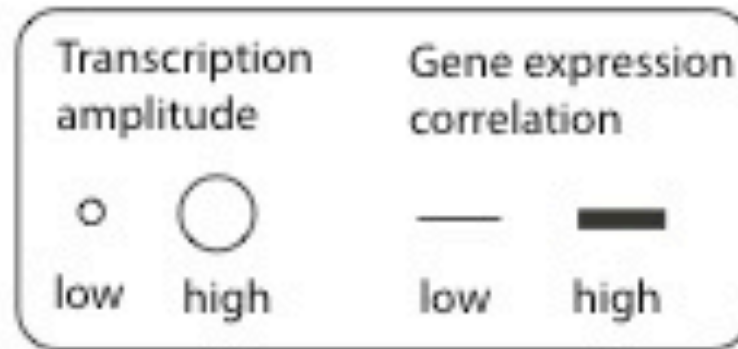


# Biological Networks



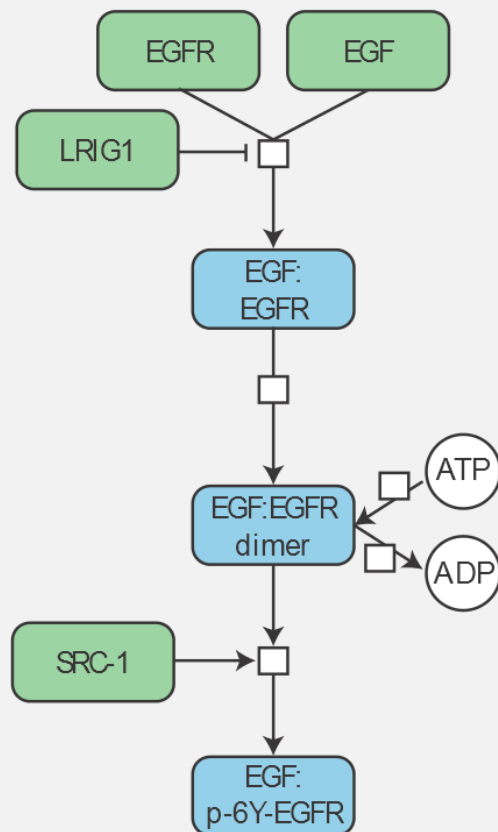
- **Represent biological interactions**
  - ➔ Physical, regulatory, genetic, functional, etc.
- **Useful for discovering relationships in big data**
  - ➔ Better than tables in Excel
- **Visualize multiple heterogeneous data types together**
  - ➔ Help highlight and see interesting patterns
- **Network analysis**
  - ➔ Well established quantitative metrics from graph theory





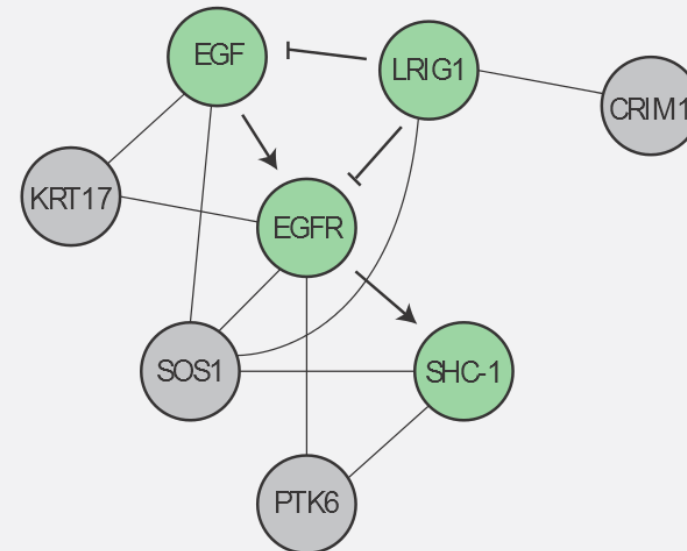
# Pathways vs Networks

EGFR-centered  
Pathway

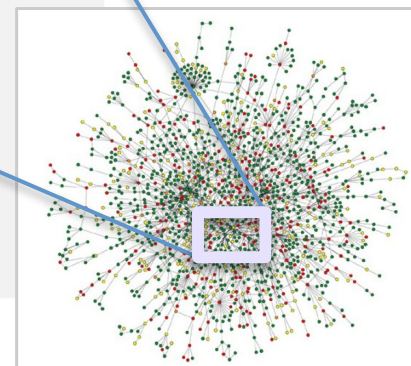


- Detailed, high-confidence consensus
- Biochemical reactions
- Small-scale, fewer genes
- Concentrated from decades of literature

EGFR-centered  
Network



- Simplified cellular logic, noisy
- Abstractions: directed, undirected
- Large-scale, genome-wide
- Constructed from *omics* data integration



## Goal

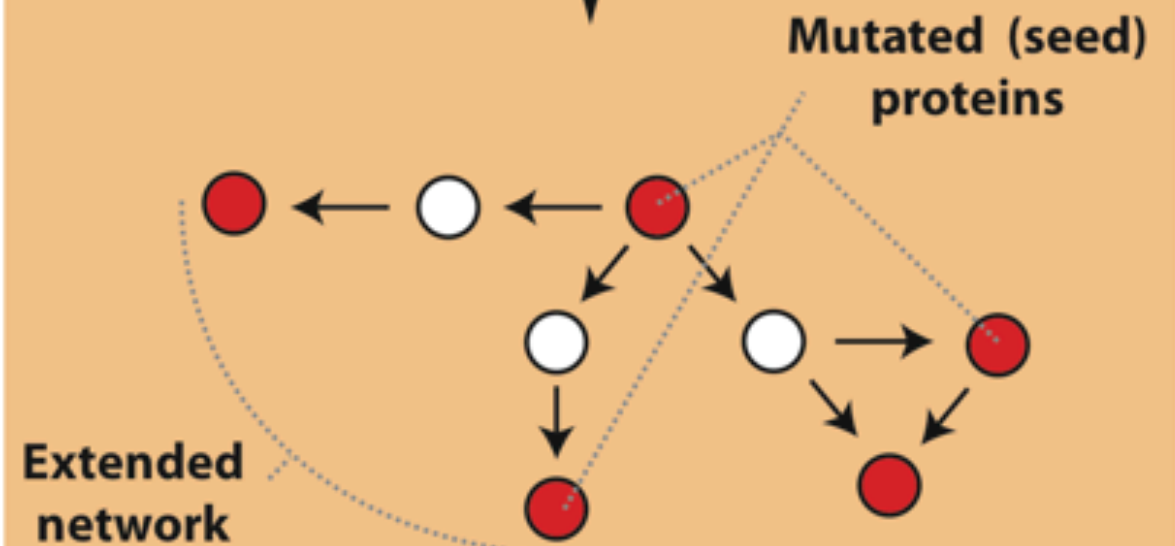
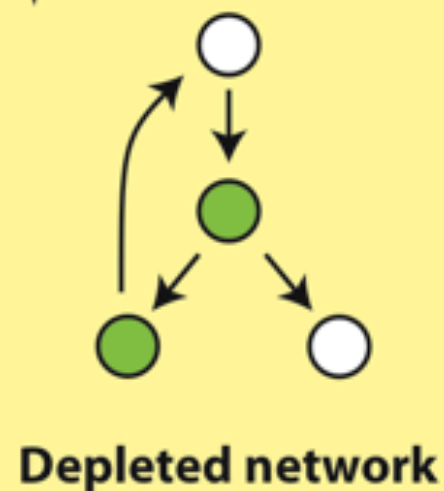
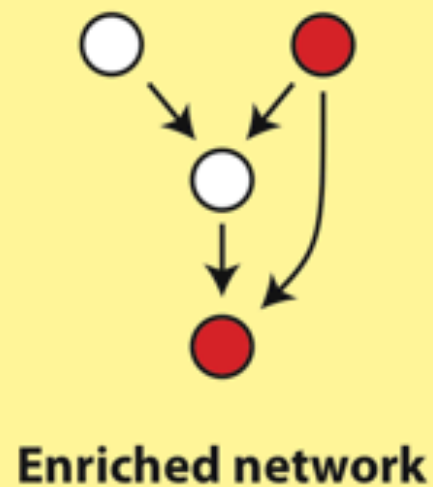
### 1 Enrichment of fixed gene sets

Identification of pre-built pathways or networks that are enriched in a set of mutated or differentially expressed genes

### 2 De novo sub-network construction and clustering

Construction of specific sub-networks from the set of mutated or differentially expressed genes to identify an extended list of putative cancer genes

## Output





## Goal

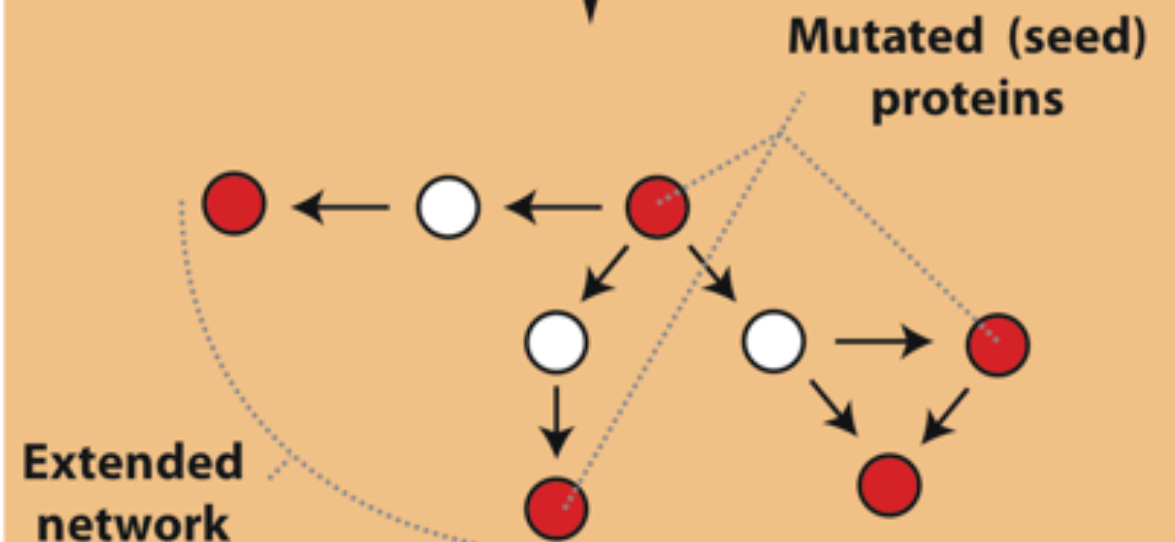
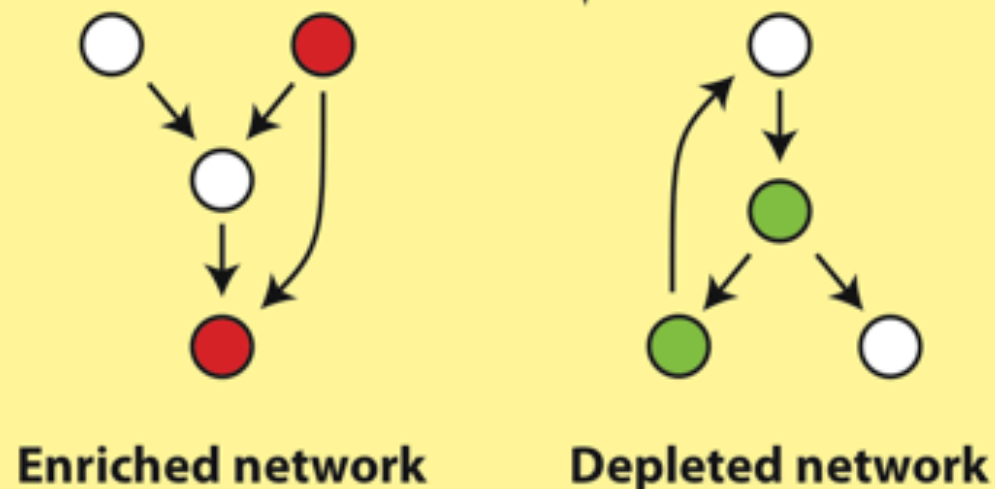
### 1 Enrichment of fixed gene sets

Identification of pre-built pathways or networks that are enriched in a set of mutated or differentially expressed genes

### 2 De novo sub-network construction and clustering

Construction of specific sub-networks from the set of mutated or differentially expressed genes to identify an extended list of putative cancer genes

## Output



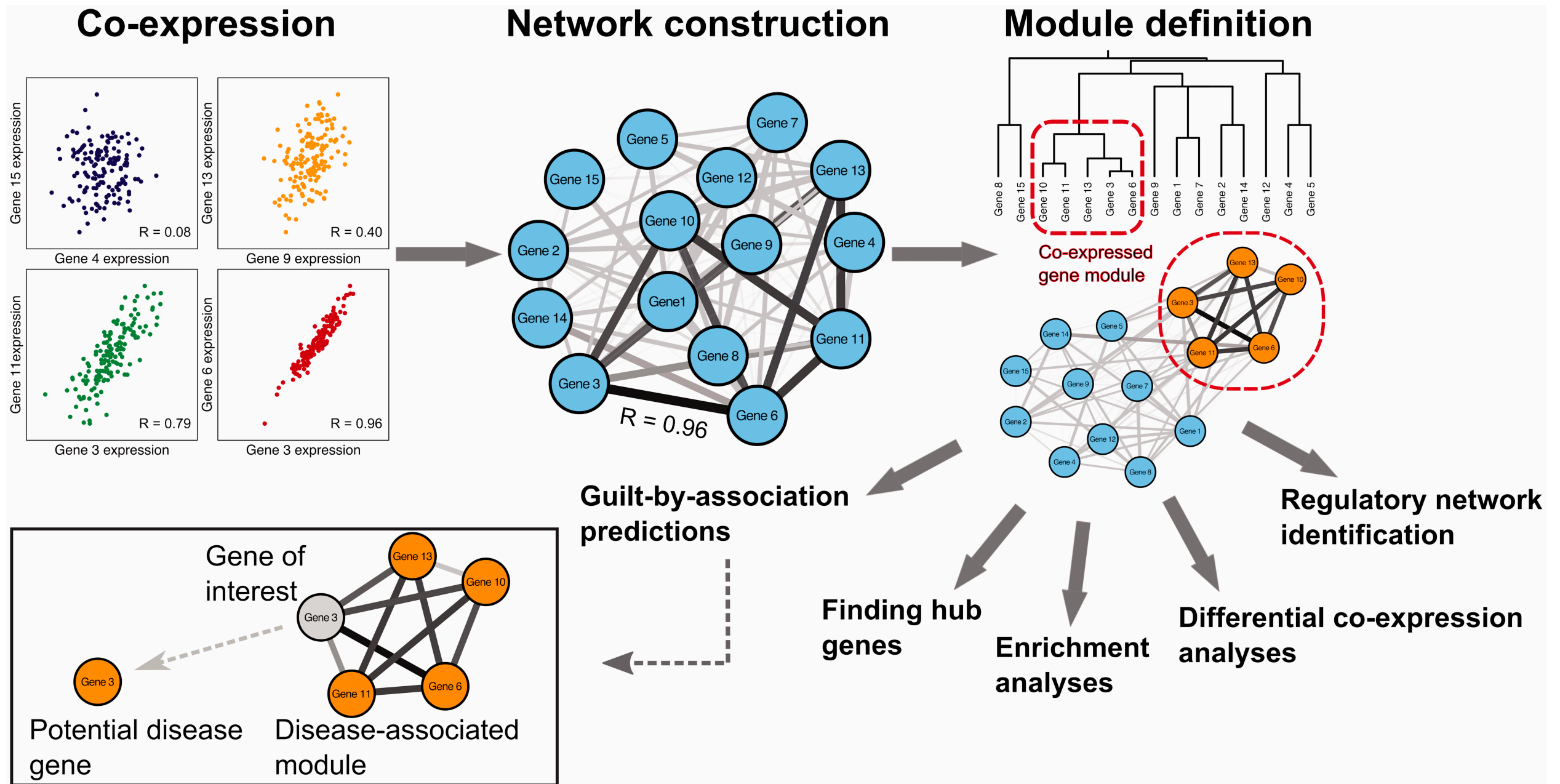
**What biological process is altered in this cancer?**

**Are NEW pathways altered in this cancer? Are there clinically relevant tumor subtypes?**

**Network analysis is complementary to pathway analysis and can be used to show how key components of different pathways interact.**

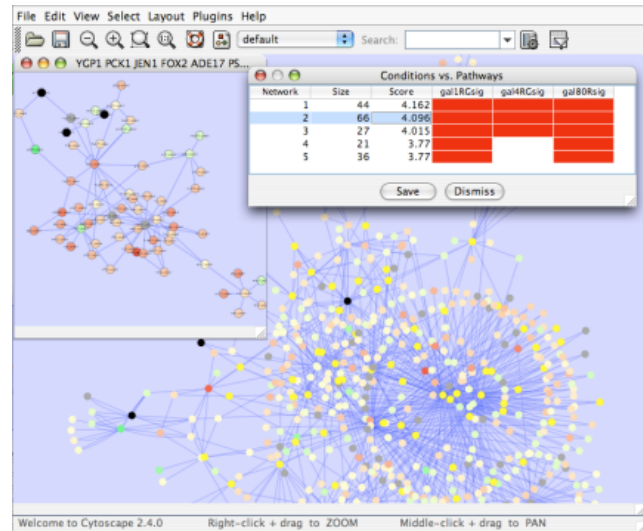
**This can be useful for identifying regulatory events that influence multiple biological processes and pathways**

# Network analysis approaches



# Applications of Network Biology

- **Gene Function Prediction** – shows connections to sets of genes/proteins involved in same biological process



jActiveModules, UCSD

- **Detection of protein complexes/other modular structures** – discover modularity & higher order organization (motifs, feedback loops)

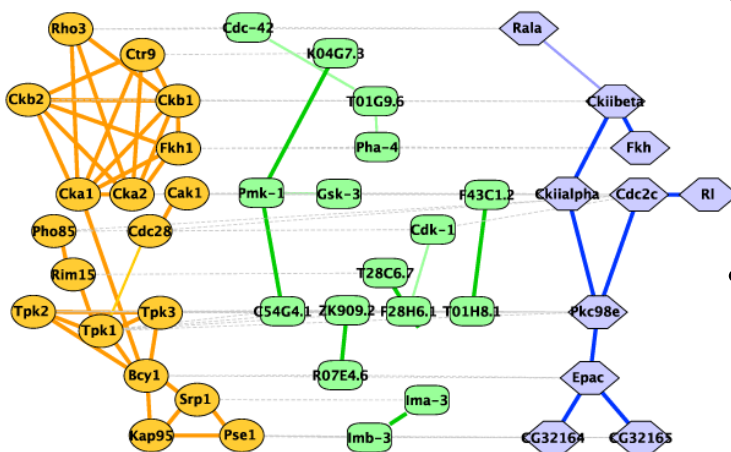


MCODE, University of Toronto

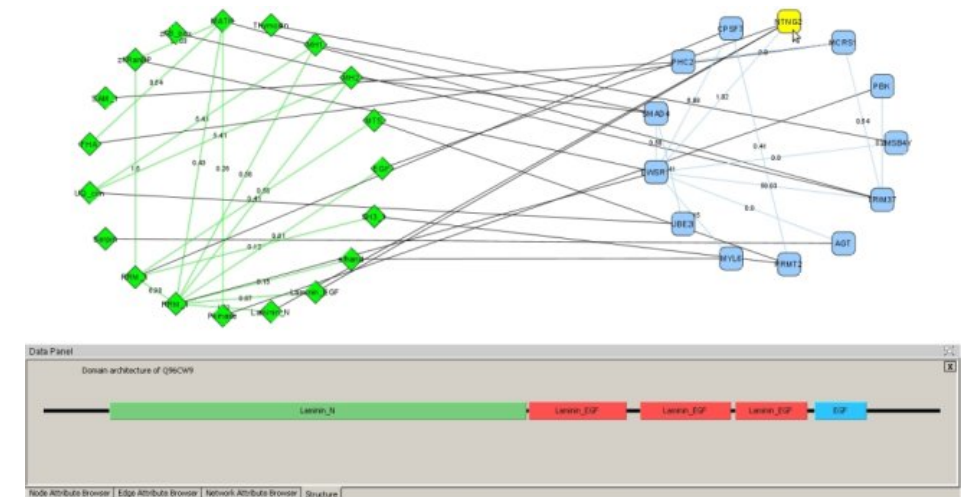
- **Network evolution** – biological process(es) conservation across species

- **Prediction of new interactions and functional associations** – Statistically significant domain-domain correlations in protein interaction network to predict protein-protein or genetic interaction; allostery in molecular networks

[b] Phosphorus metabolism  
Complexes 32, 296, 728, 822, 894, 927



PathBlast, UCSD



DomainGraph, Max Planck Institute

# What's missing

- **Dynamics**
  - ➔ Pathways/networks represented as static processes
  - ➔ Difficult to represent a calcium wave or a feedback loop
  - ➔ More detailed mathematical representations exist that handle these e.g. Stoichiometric modeling, Kinetic modeling (VirtualCell, E-cell, ...)
- **Detail** – atomic structures & exclusivity of interactions.
- **Context** – cell type, developmental stage



# What have we learned so far...

- **Networks are useful for seeing relationships in large data sets**
  - ➔ Important to understand what the nodes and edges mean
  - ➔ Important to define the biological question - know what you want to do with your gene list or network
- **Many methods available for network analysis**
  - ➔ Good to determine your question and search for a solution
  - ➔ Or get to know many methods and see how they can be applied to your data

# TODAYS MENU:

▶ Network introduction

▶ **Network visualization**

▶ **Network analysis**

▶ **Hands-on:**

Cytoscape and R (igraph) software tools  
for network visualization and analysis

# Network Visualization Outline

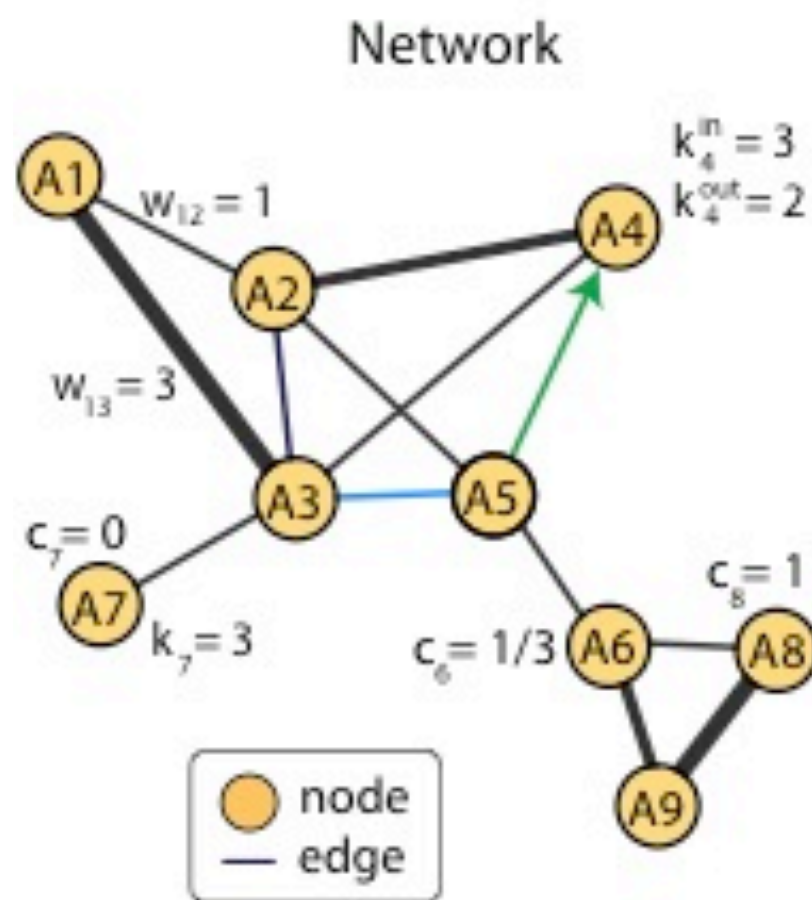
- Network representations
- Automatic network layout
- Visual features
- Visually interpreting a network

# Network representations

Relationships	Optional weight
A1 ↔ A2	1
A1 ↔ A3	3
A2 ↔ A3	1
A2 ↔ A4	2
A2 ↔ A5	1
A3 ↔ A4	1
A3 ↔ A5	1
A3 ↔ A7	1
A5 → A4	1
A5 ↔ A6	1
A6 ↔ A8	1
A6 ↔ A9	2
A8 ↔ A9	3

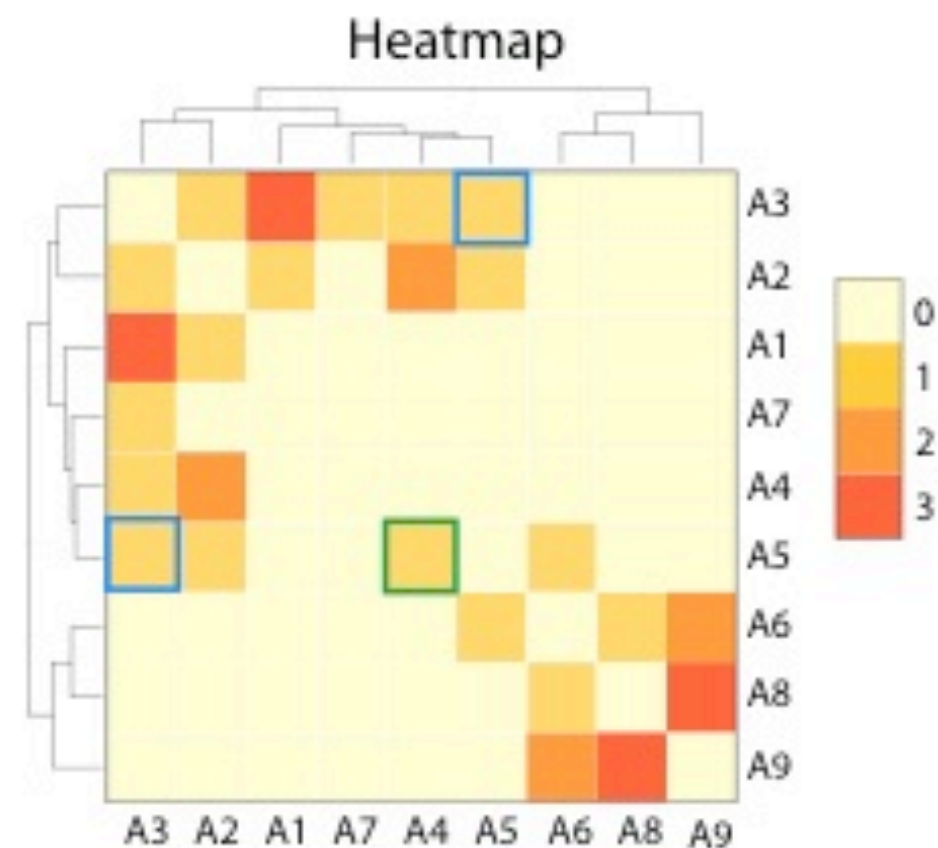
1

List of relationships



2

Network view



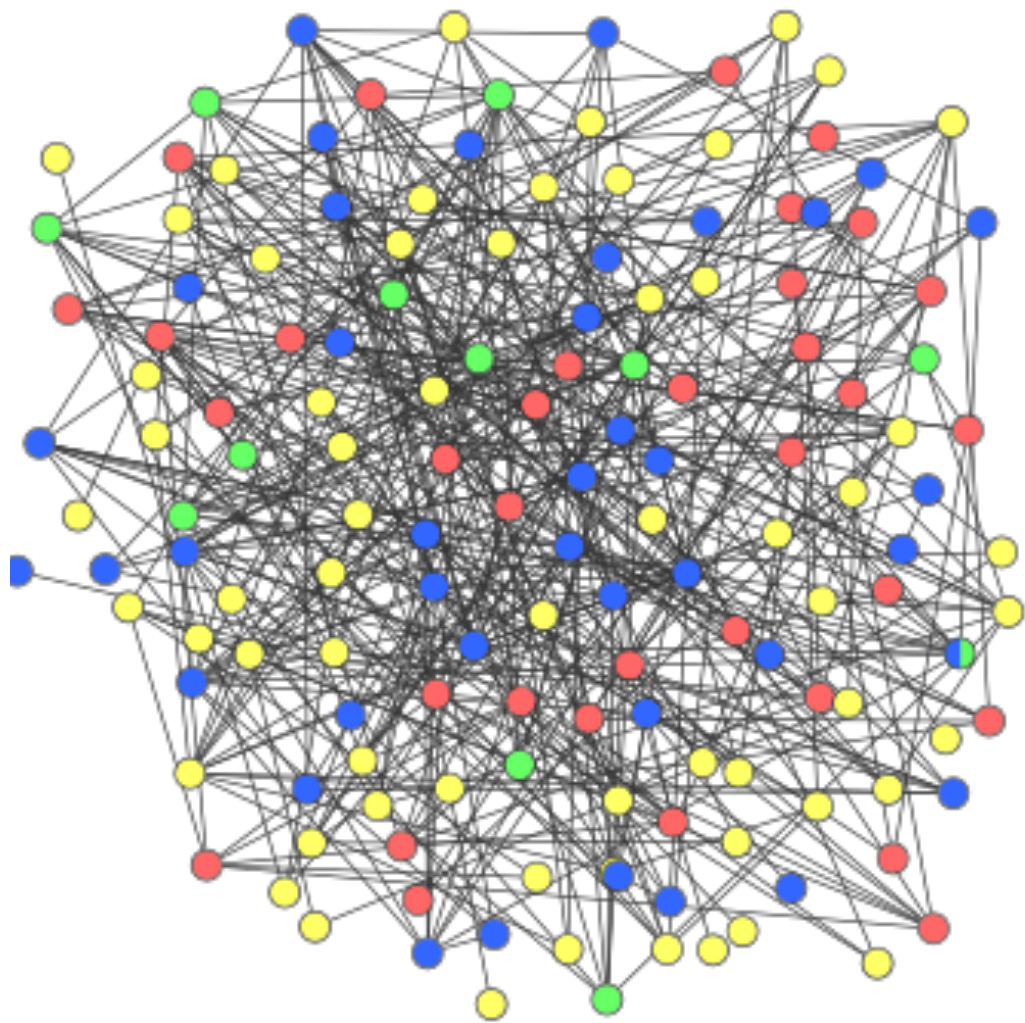
3

Adjacency matrix view

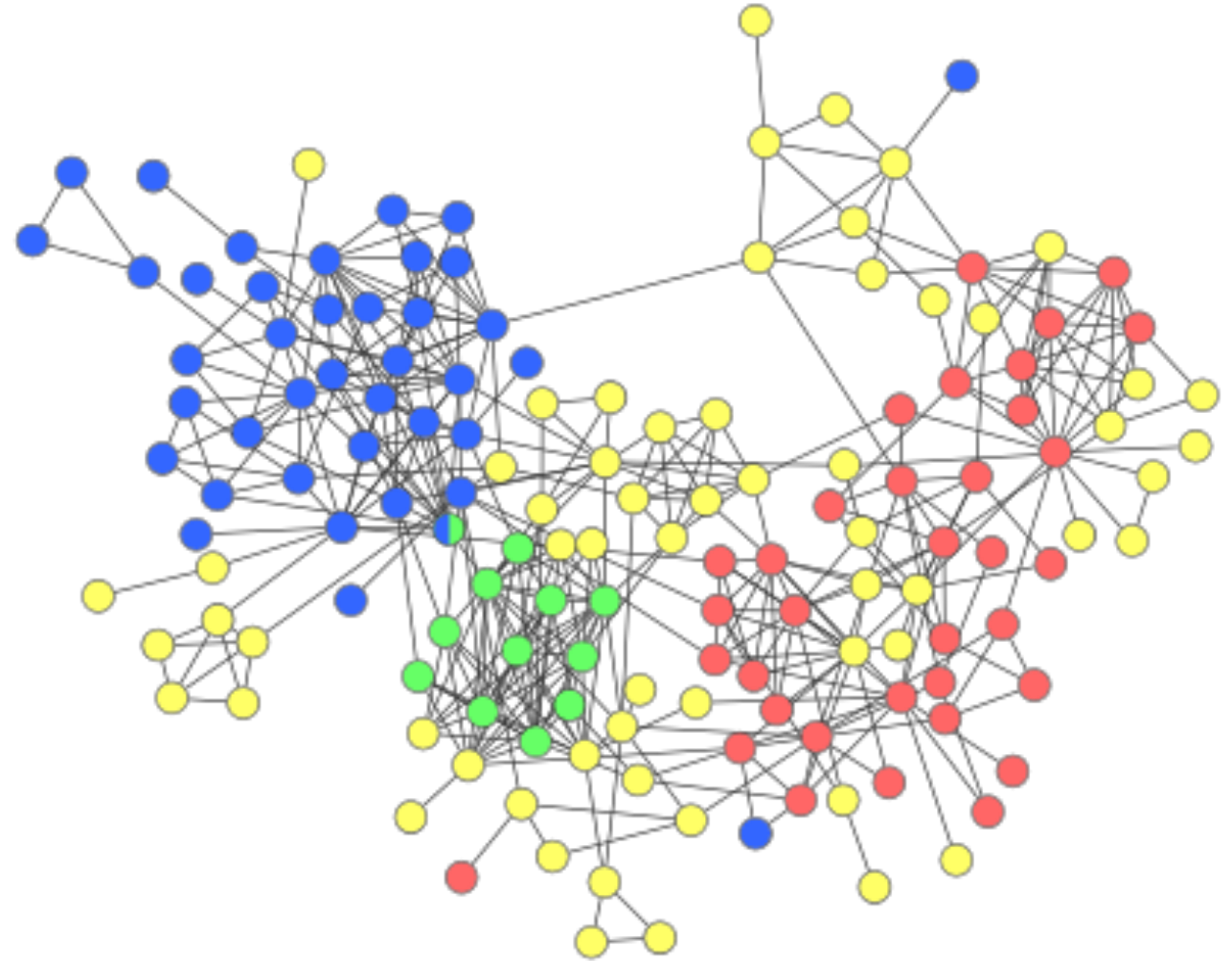
**Network view is most useful when network is sparse!**

# Automatic network layout

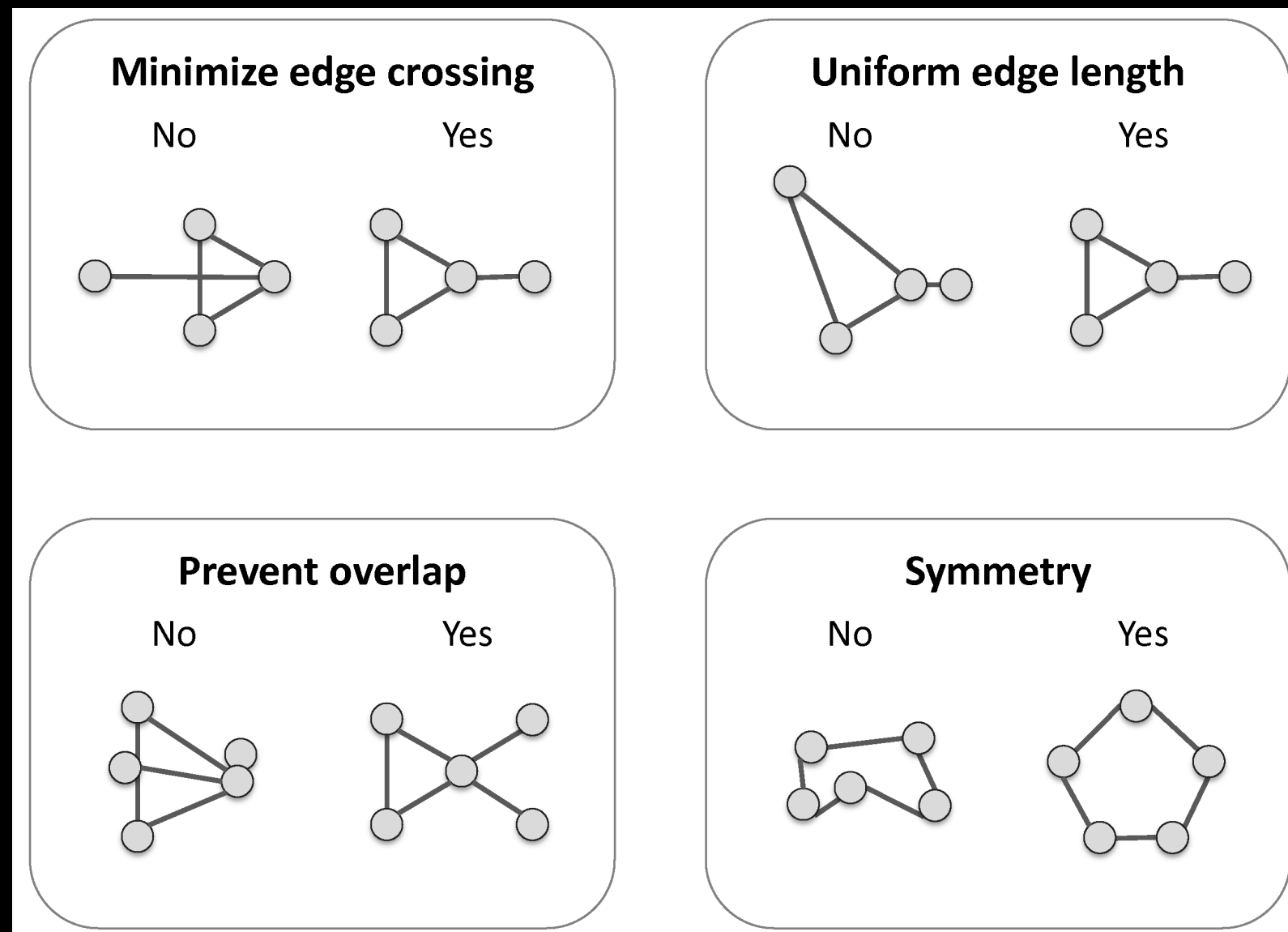
Before layout



After layout



- Modern **graph layouts** are optimized for speed and aesthetics. In particular, they seek to minimize overlaps and edge crossing, and ensure similar edge length across the graph.



# Force-directed layout:

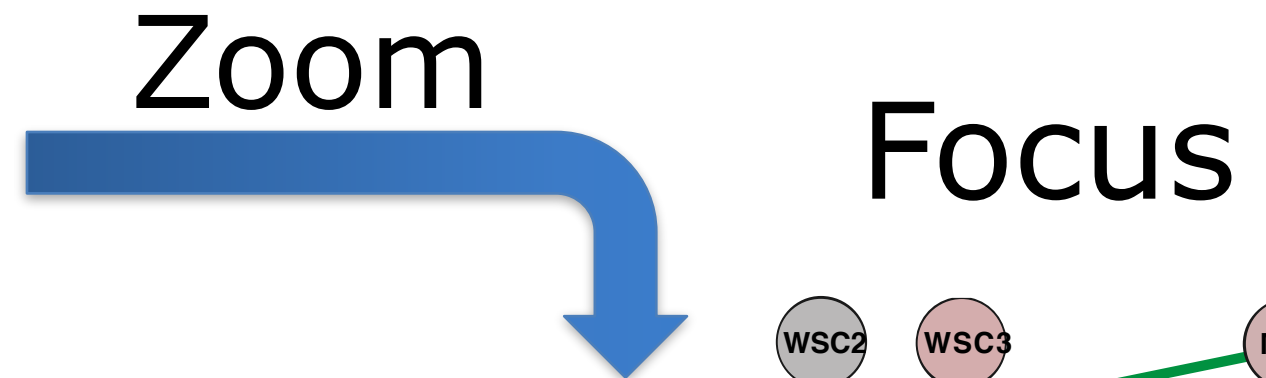
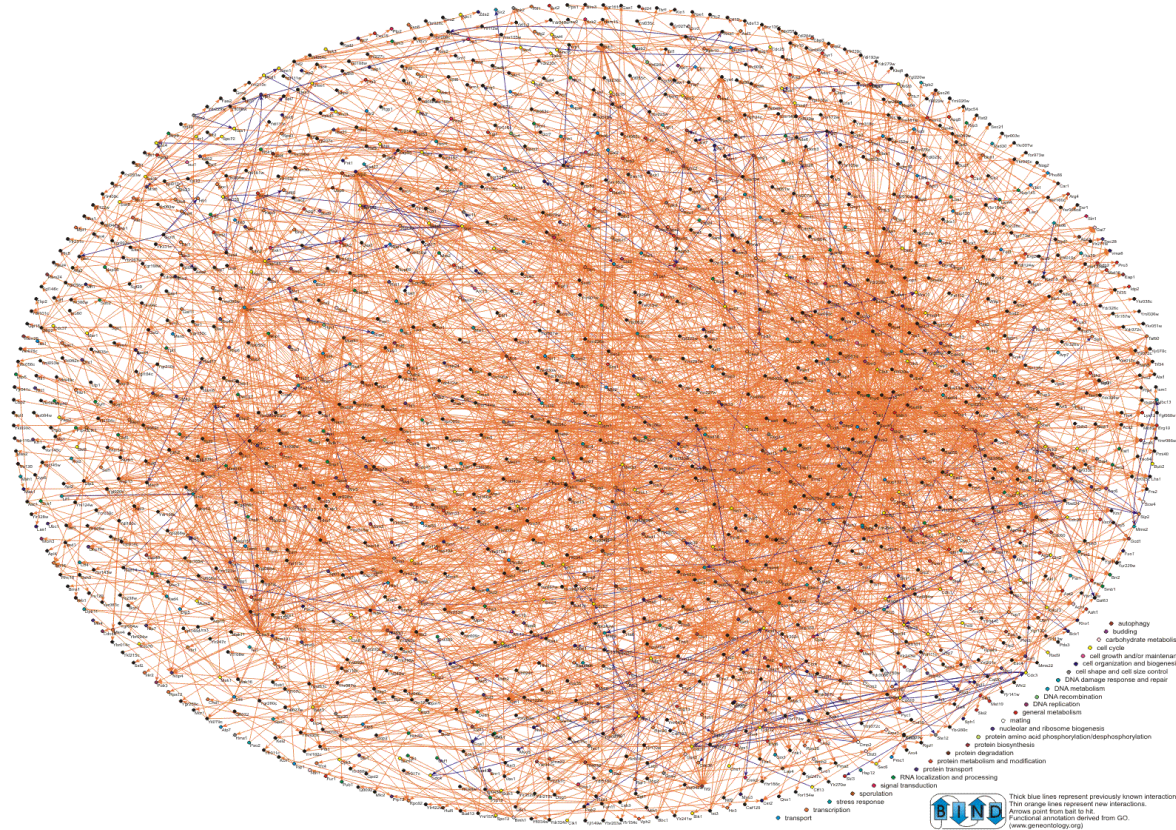
Nodes repel and edges pull

- Good for up to 500 nodes
  - ➔ Bigger networks give hairballs
  - ➔ Reduce number of edges
  - ➔ Or just use a heatmap for dense networks
- Advice: try force directed first, or hierarchical for tree-like networks
- Tips for better looking networks
  - ➔ Manually adjust layout
  - ➔ Load network into a drawing program (e.g. Illustrator) and adjust labels







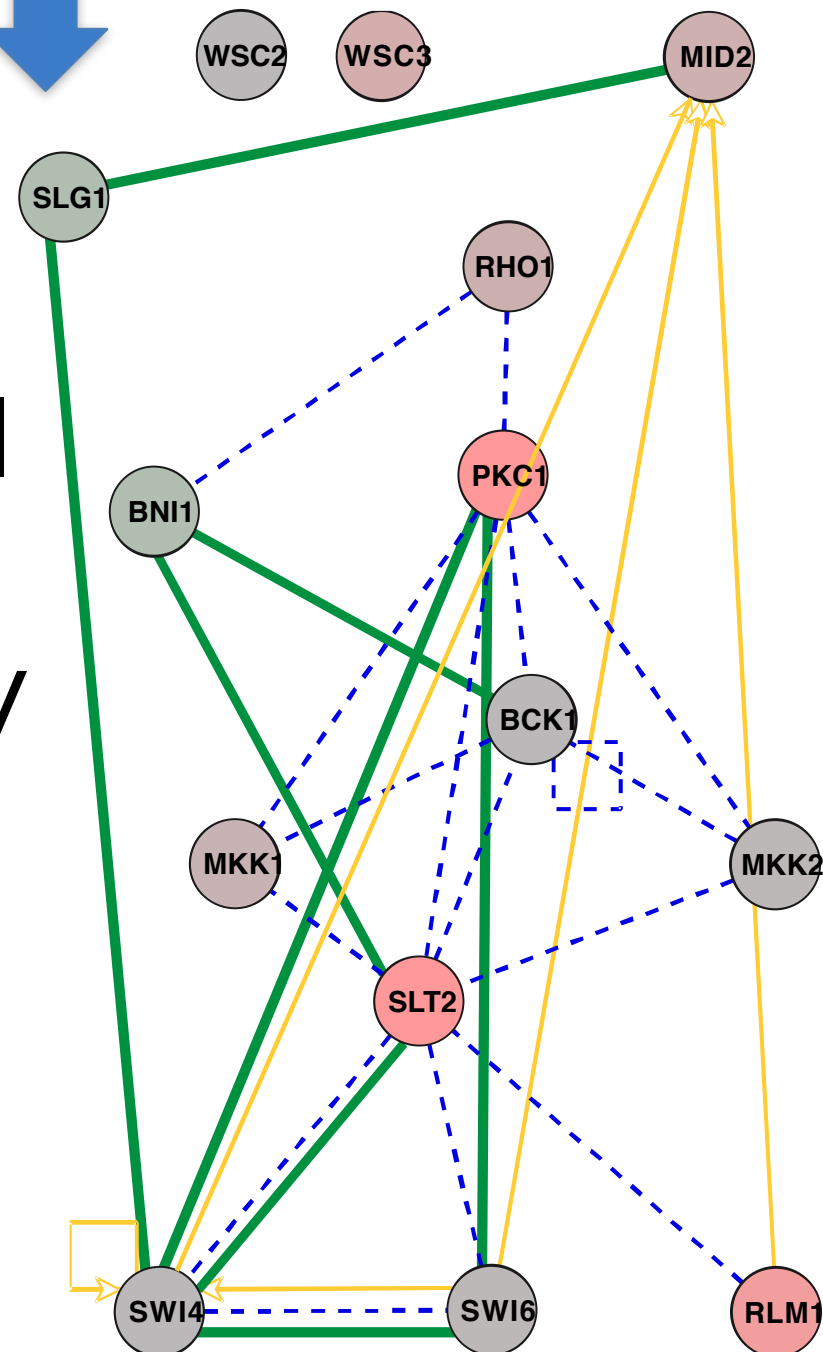
# Dealing with 'hairballs': zoom or filter

Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry



PKC Cell  
Wall  
Integrity

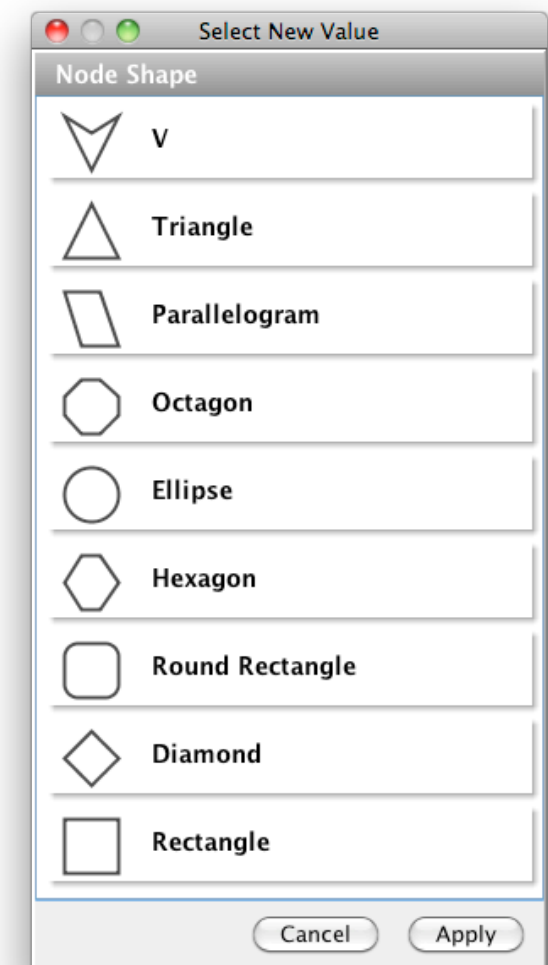
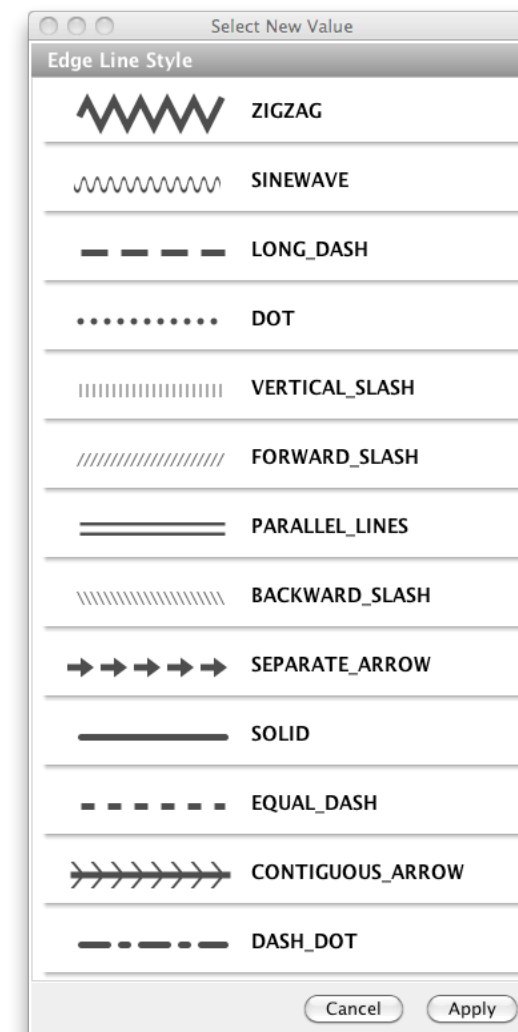
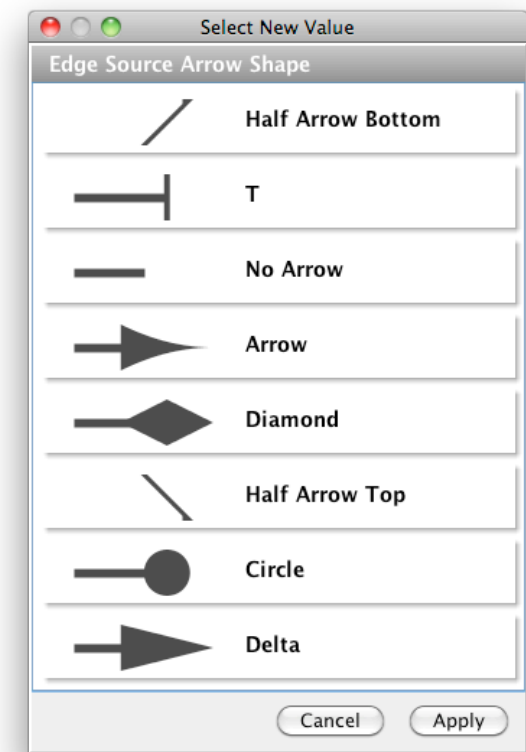
-  Synthetic Lethal
-  Transcription Factor Regulation
-  Protein-Protein Interaction
-  Up Regulated Gene Expression
-  Down Regulated Gene Expression



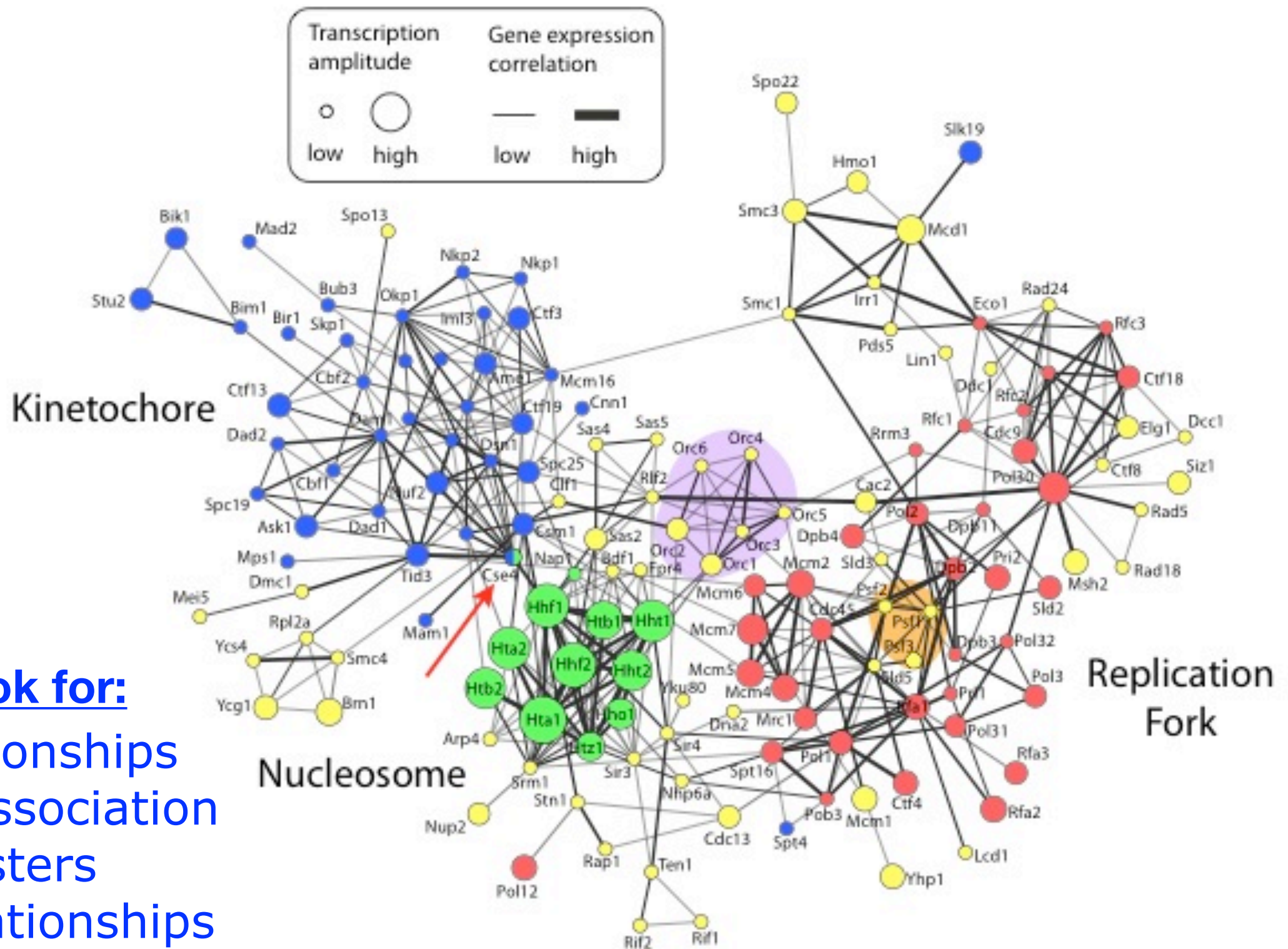


# Visual Features

- Node and edge attributes
  - Text (string), integer, float, Boolean, list
  - E.g. represent gene, interaction attributes
- Visual attributes
  - Node, edge visual properties
  - Color, shape, size, borders, opacity...



# Visually Interpreting a Network



## What to look for:

Data relationships  
Guilt-by-association  
Dense clusters  
Global relationships

# What have we learned so far...

- Automatic layout is required to visualize networks
- Networks help you visualize interesting relationships in your data
- Avoid hairballs by focusing analysis
- Visual attributes enable multiple types of data to be shown at once – useful to see their relationships

# TODAYS MENU:

- ▶ Network introduction
- ▶ Network visualization
- ▶ **Network analysis**
- ▶ **Hands-on:**
  - Cytoscape and R (igraph) software tools for network visualization and analysis

# Hands-on: Part 1

[https://bioboot.github.io/bggn213\\_W20/lectures/#17](https://bioboot.github.io/bggn213_W20/lectures/#17)

- The data used in **part 1** is from yeast, and the genes **Gal1**, **Gal4**, and **Gal80** are all yeast transcription factors. The experiments all involve some perturbation of these transcription factor genes.

# Practical issues

## Lab1

- Major tools for the **creation, manipulation** and **visualization** of biological networks include:
  - ➔ Cytoscape,
  - ➔ Gephi
  - ➔ R packages (igraph, graph, tidygraph, ggraph)
- Tools for network analysis and modeling include:
  - ➔ Cytoscape apps/plugins
  - ➔ R packages (igraph and many others)
  - ➔ NetworkX (for Python)
  - ➔ ByoDyn, COPASI



Session: /Applications/Cytoscape\_v3.6.0/Cytoscape.app/./sampleData/sessions/Yeast Perturbation.cys



Control Panel

Network Style Select

galFiltered Style

Properties

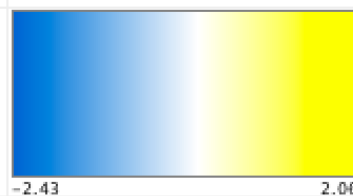
Def. Map. Byp.

☐ Border Paint Border Width☐ Fill Color

Column gal1RGexp

Mapping Type Continuous Mapping

Current Mapping

☐ Height☐ Image/Chart 1☐ Label☐ Label Color Label Font Size☐ Shape Size Transparency☐ Width

Node Edge Network

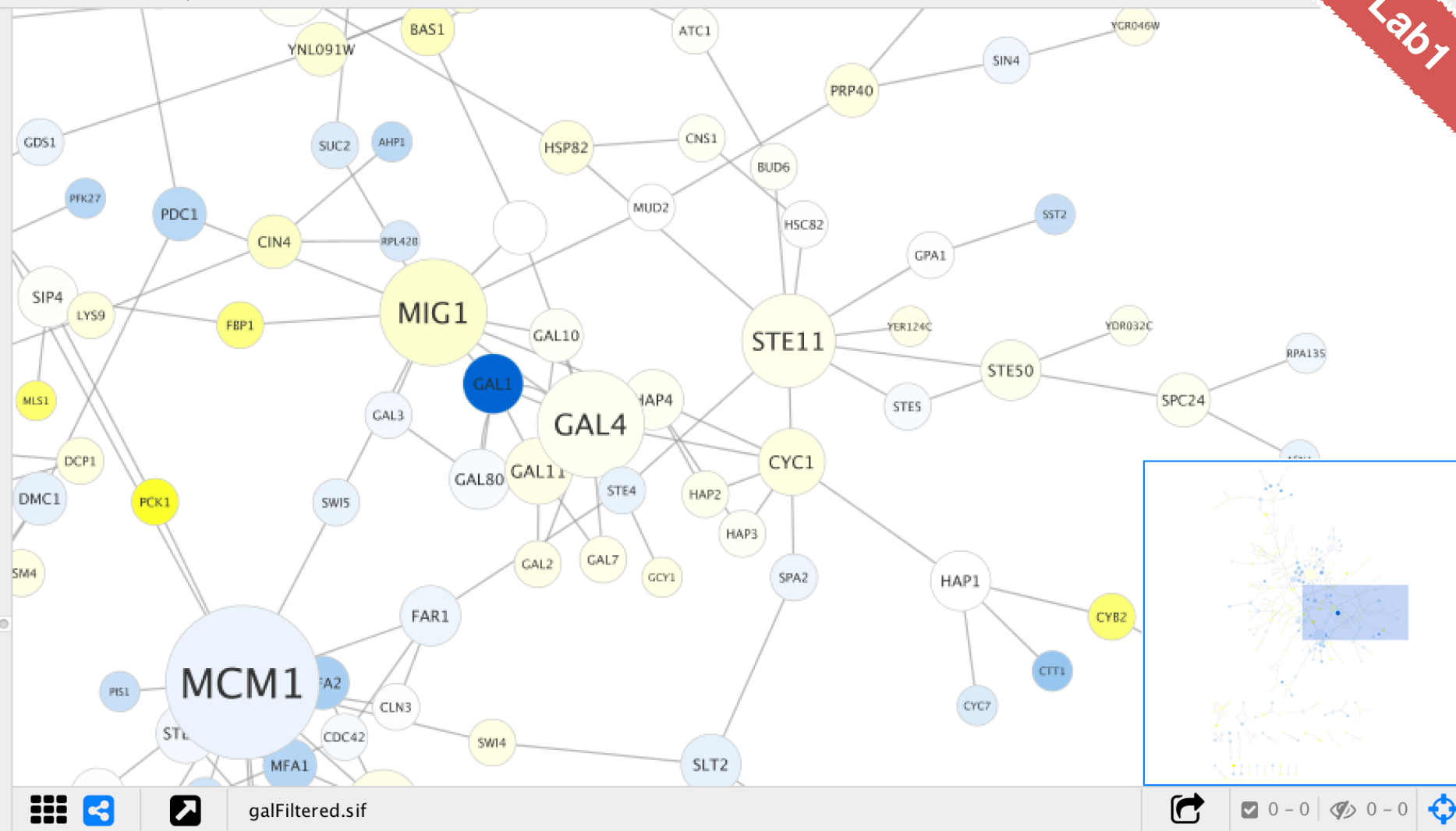


Table Panel



COMMON	gal1RGexp	gal4RGexp	gal80RGexp	gal1RGsig	gal4RGsig	gal80Rsig
GCN3	-0.154	-0.501	0.292	9.1177E-4	3.5692E-6	0.011229
NAB2	0.174	0.02	0.187	8.7295E-4	0.61707	0.0059966
CRM1	-0.018	-0.001	-0.018	0.61381	0.9794	0.80969
SRM1	0.16	-0.23	0.008	0.0021913	0.0022461	0.93826
DED1	-0.033	-0.056	-0.91	0.39944	0.31268	8.349E-16
YEF3	-0.39	-0.394	-0.769	2.713E-8	0.04747	0.035939

Node Table

Edge Table

Network Table

Memory

# Cytoscape Memory Issues

- Cytoscape uses lots of memory and doesn't like to let go of it
  - ➔ An occasional restart when working with large networks is a good thing
  - ➔ Destroy views when you don't need them
- Since version 2.7, Cytoscape does a much better job at “guessing” good default memory sizes than previous versions but it still not great!
  - ➔ Java doesn't give us a good way to get the memory right at start time



# Cytoscape Sessions

- Sessions save pretty much everything:
  - ➔ Networks
  - ➔ Properties
  - ➔ Visual styles
  - ➔ Screen sizes
- Saving a session on a large screen may require some resizing when opened on your laptop

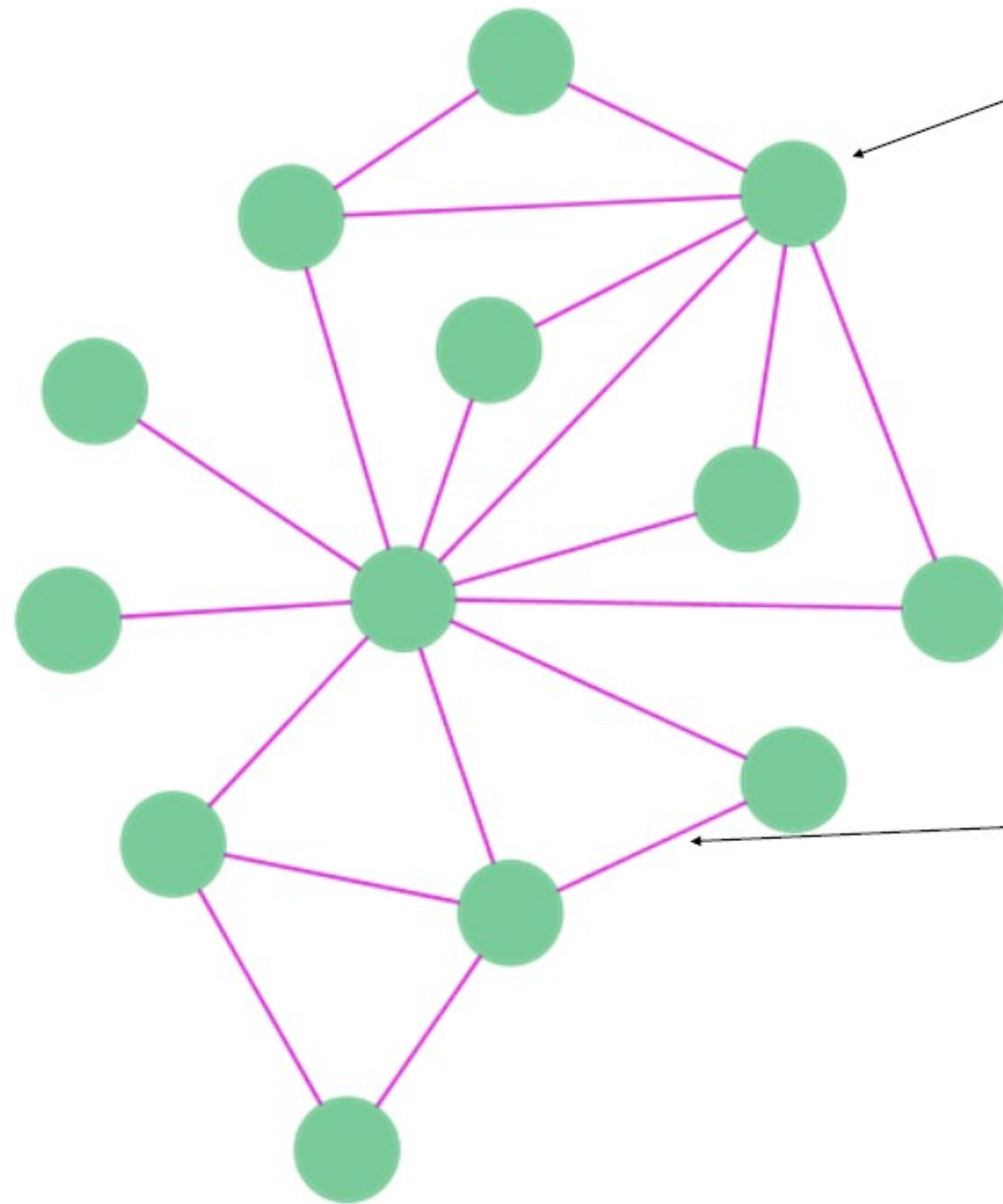
# TODAYS MENU:

- ▶ Network introduction
- ▶ Network visualization
- ▶ **Network analysis**
- ▶ **Hands-on:**
  - Cytoscape and R (igraph) software tools for network visualization and analysis

# Introduction to graph theory

- Biological network analysis historically originated from the tools and concepts of **social network analysis** and the application of **graph theory** to the social sciences.
- Wikipedia defines graph theory as:
  - ➔ “[...] the study of graphs used to model pairwise relations between objects. A graph in this context is made up of **vertices** connected by **edges**”.
- In practical terms, it is the set of concepts and methods that can be used to visualize and analyze networks

## Network or graph



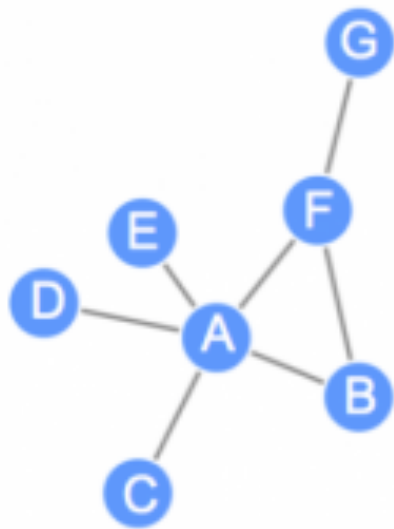
Node or vertex: protein,  
gene, drug, disease

Edge or link: relation between  
nodes

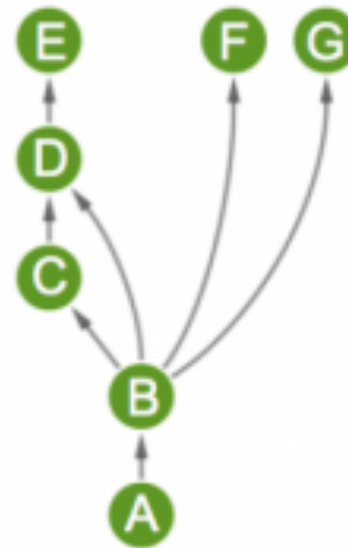
- Binary or continuous
- Directed or undirected
- Edge types

# Types of network edges

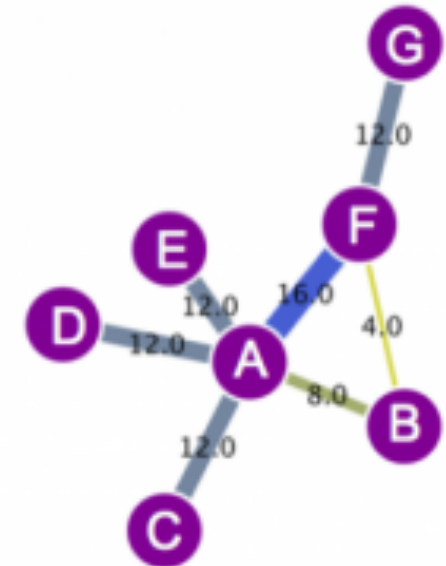
Undirected



Directed



Weighted



Connection,  
without a given  
'flow' implied

(e.g. protein A  
binds protein B)

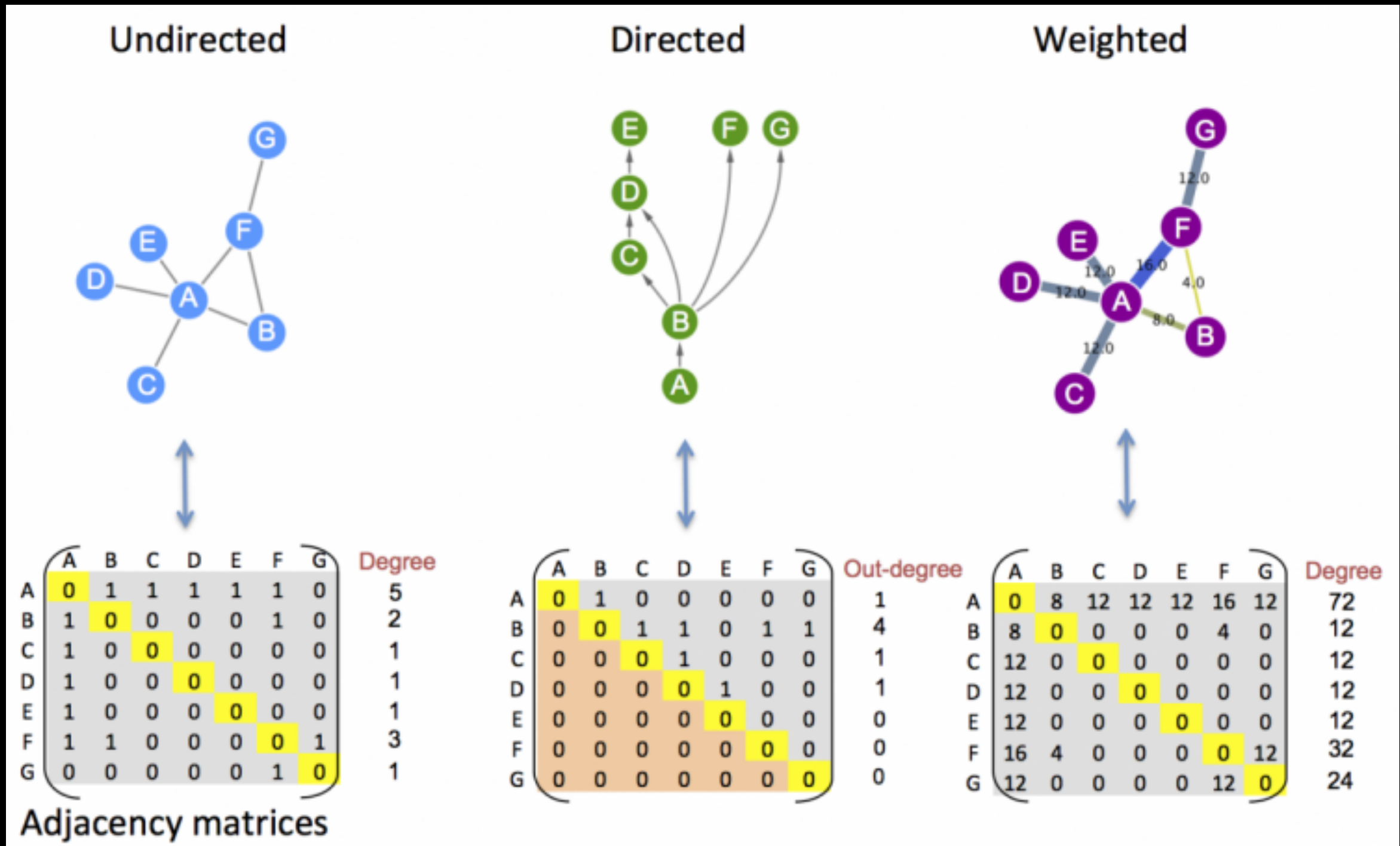
There is directional  
flow/signal implied

(e.g. metabolic or  
gene networks)

Edges can also  
have weight

(i.e. a 'strength' of  
interaction).

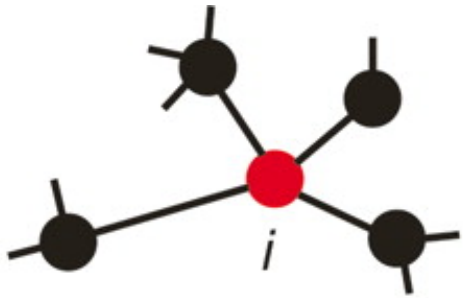
- Every network can be expressed mathematically in the form of an adjacency matrix



# Network topology

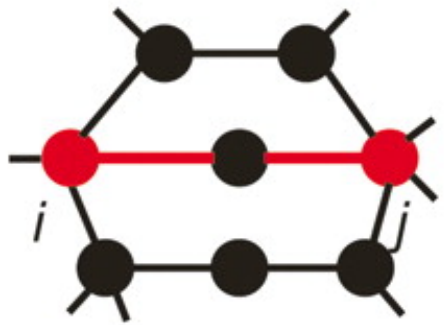
- Topology is the way in which the nodes and edges are arranged within a network.
- The most used topological properties and concepts include:
  - ➔ **Degree** (i.e. how many node neighbors)
  - ➔ **Communities** (i.e. clusters of well connected nodes)
  - ➔ **Shortest Paths** (i.e. shortest distance between 2 nodes)
  - ➔ **Centralities** (i.e. how 'central' is a given node?)
  - ➔ **Betweenness** (a measure of centrality based on shortest paths)





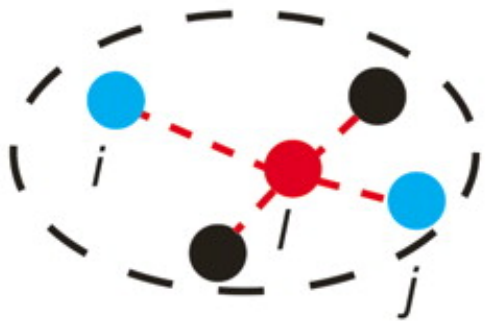
Degree

$k_i$  = number of links connected to node  $i$



Distance

$d_{ij}$  = shortest path length between node  $i$  and  $j$



Betweenness

$b_l = \sum_{ij} p_{ij}(l) / p_{ij}$

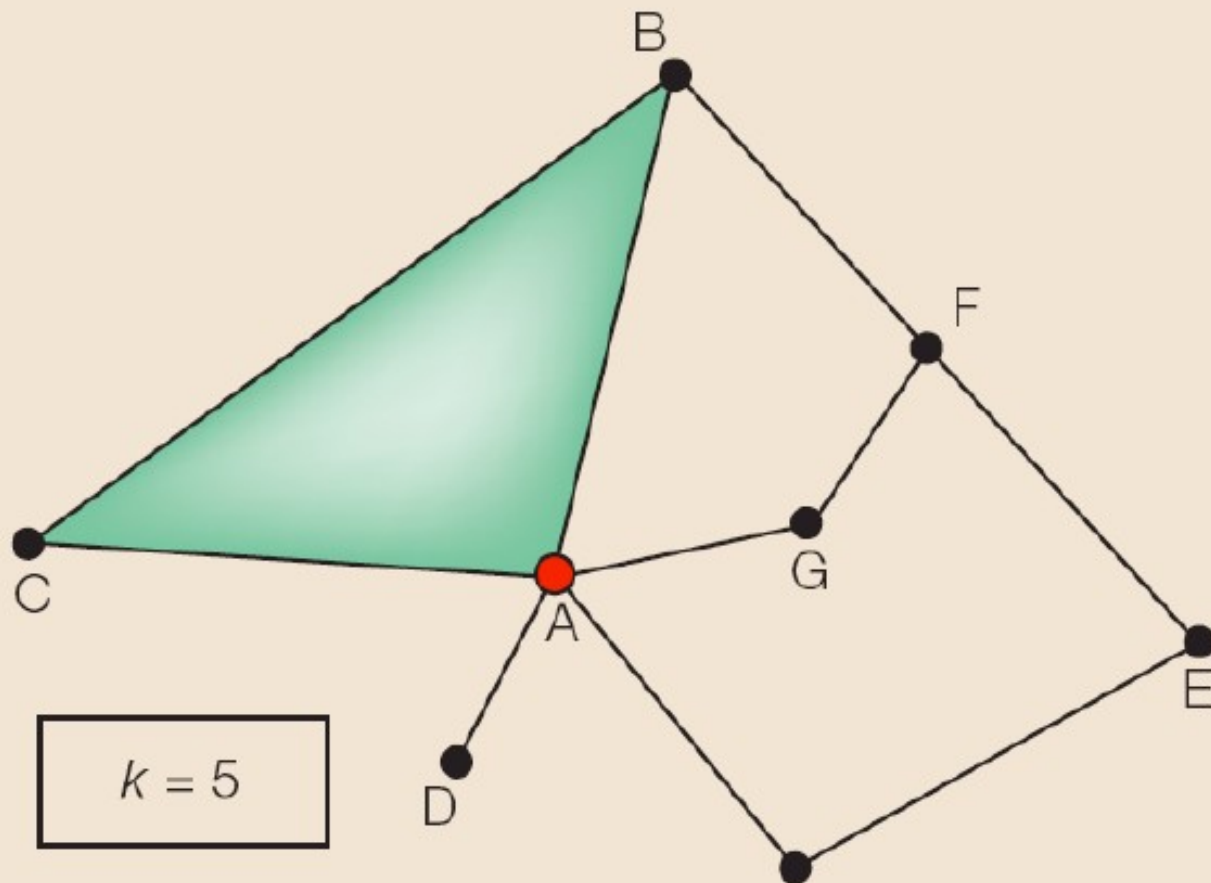
$p_{ij}$  : number of shortest paths between  $i$  and  $j$

$p_{ij}(l)$  : number of shortest paths between  $i$  and  $j$  going through node  $l$

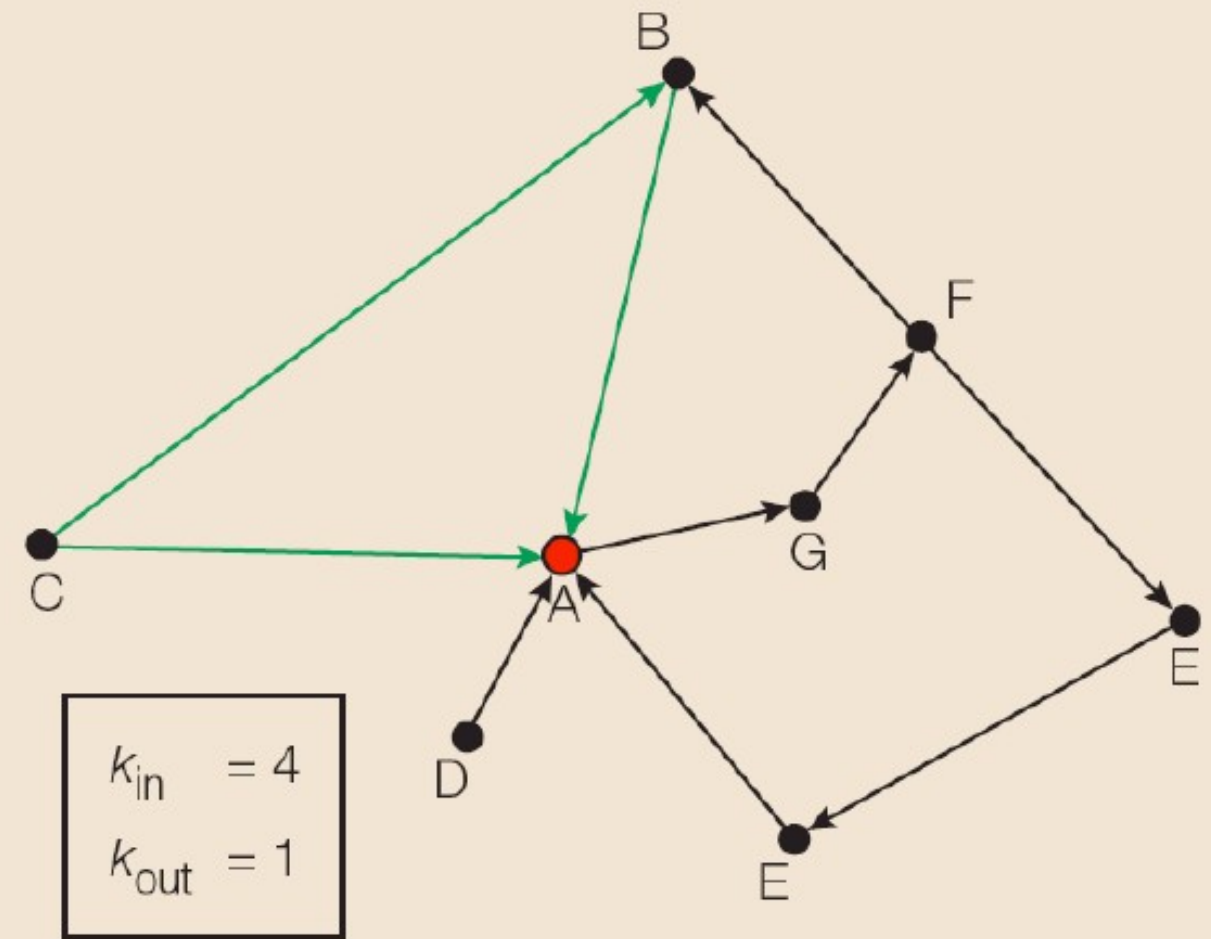


# Network Measures: Degree

**a** Undirected network

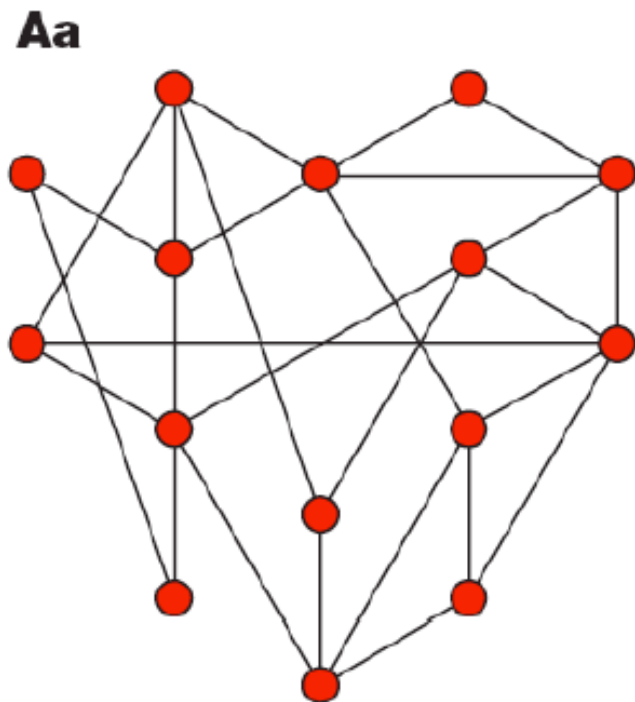


**b** Directed network

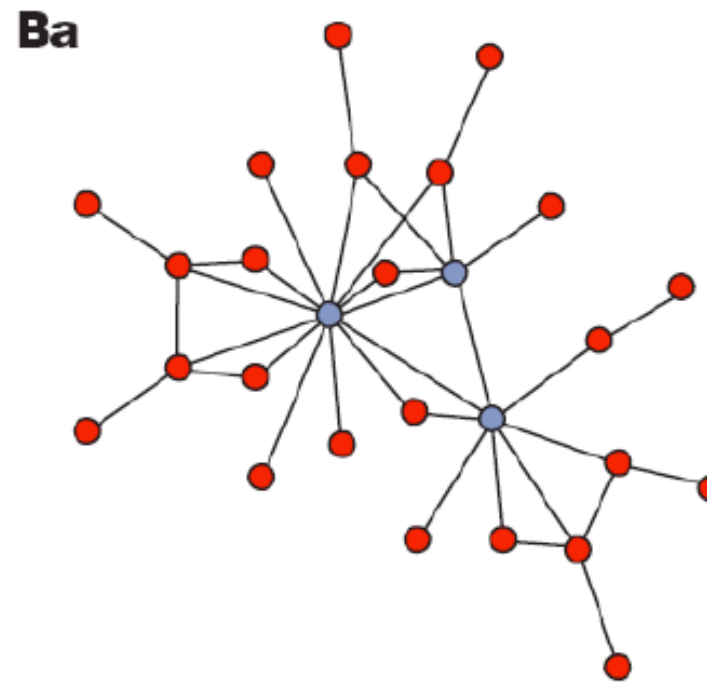


# Degree Distribution

**A** Random network



**B** Scale-free network



$P(k)$  is probability of each degree  $k$ , i.e fraction of nodes having that degree.

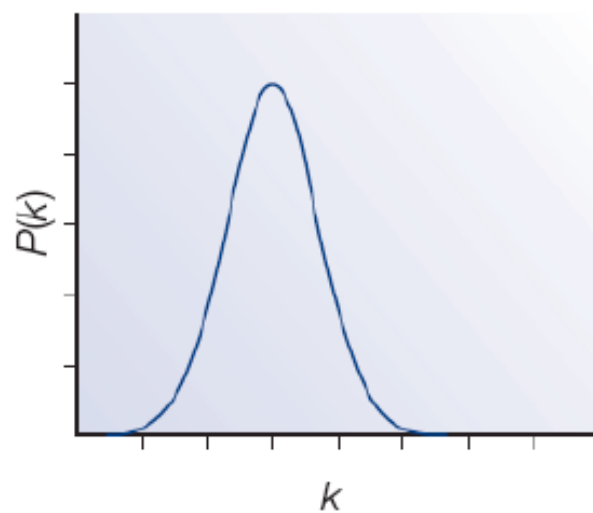
For random networks,  $P(k)$  is normally distributed.

For real networks the distribution is often a power-law:

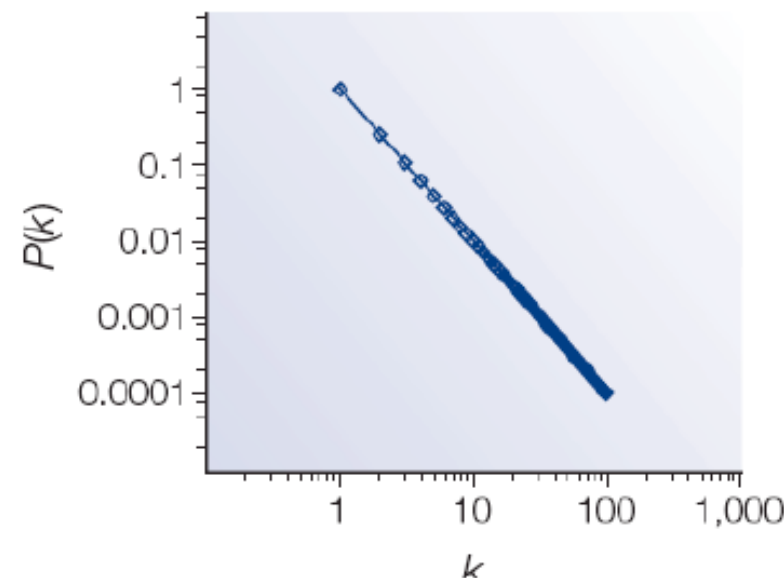
$$P(k) \sim k^{-\gamma}$$

Such networks are said to be **scale-free**

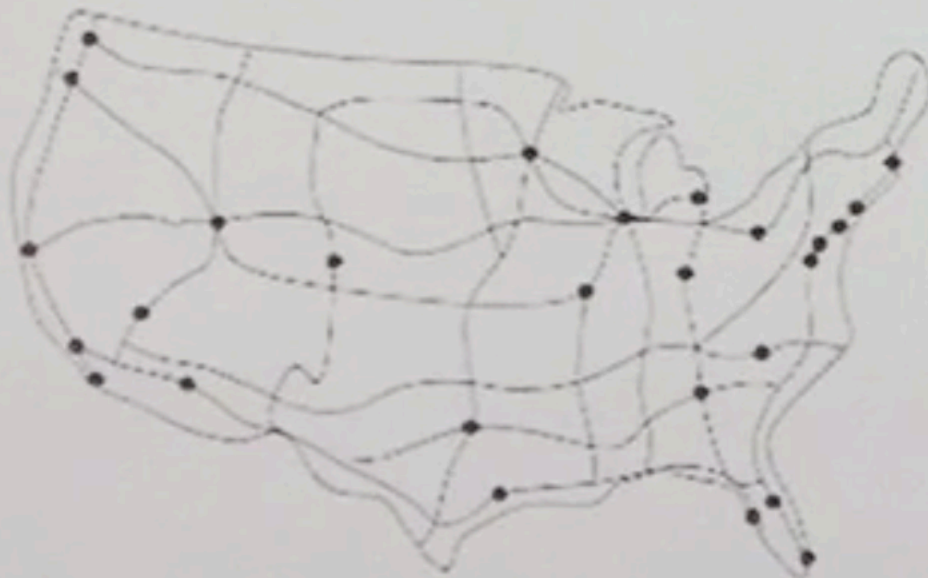
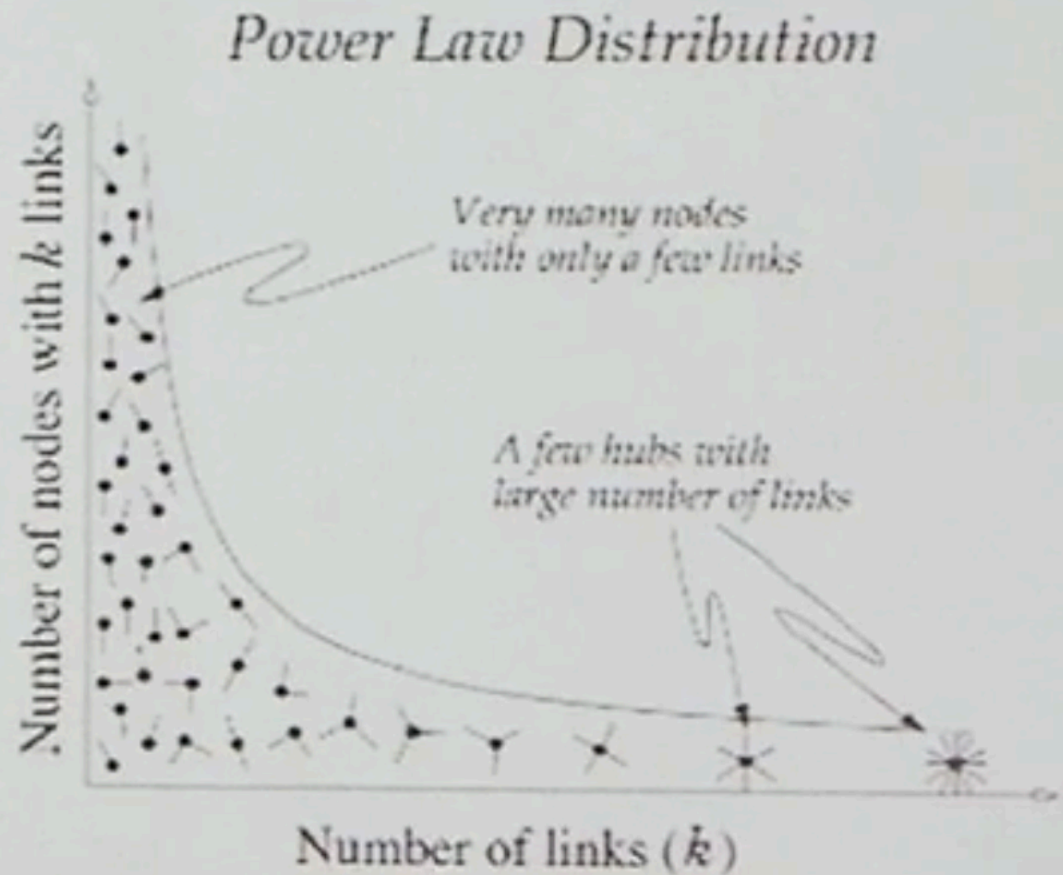
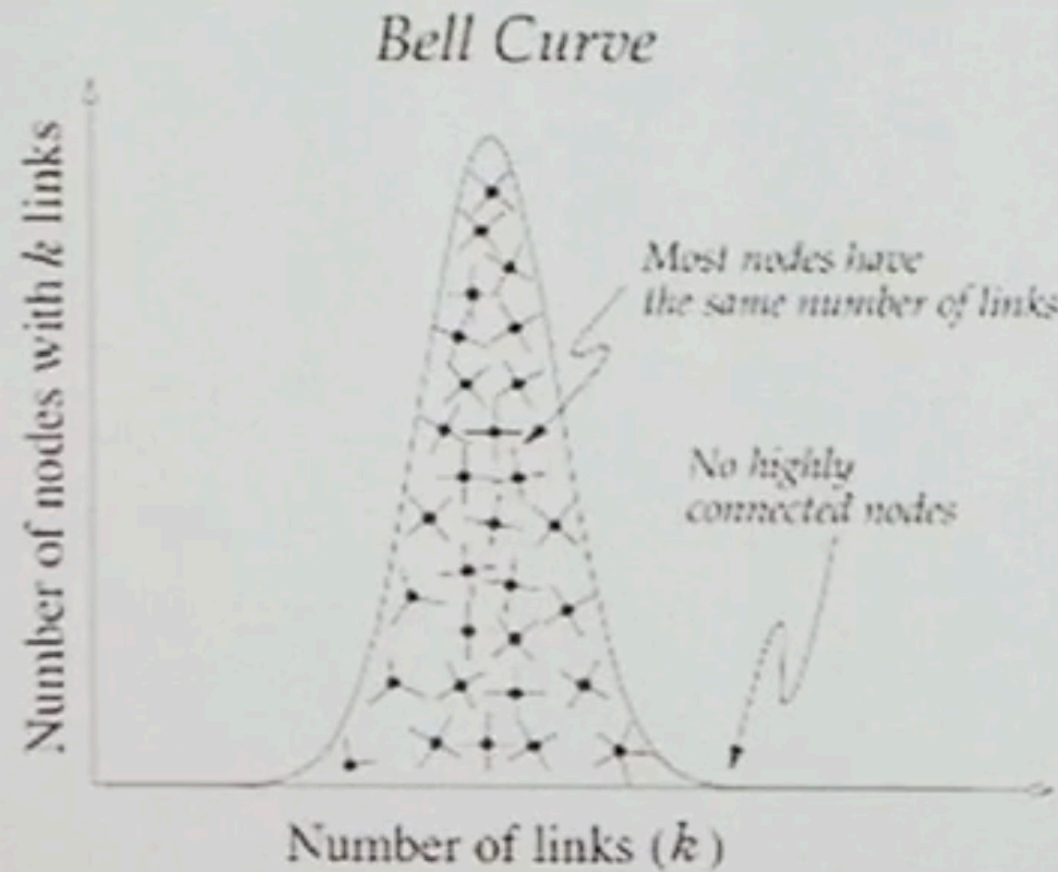
**Ab**



**Bb**



# Random graphs vs scale free





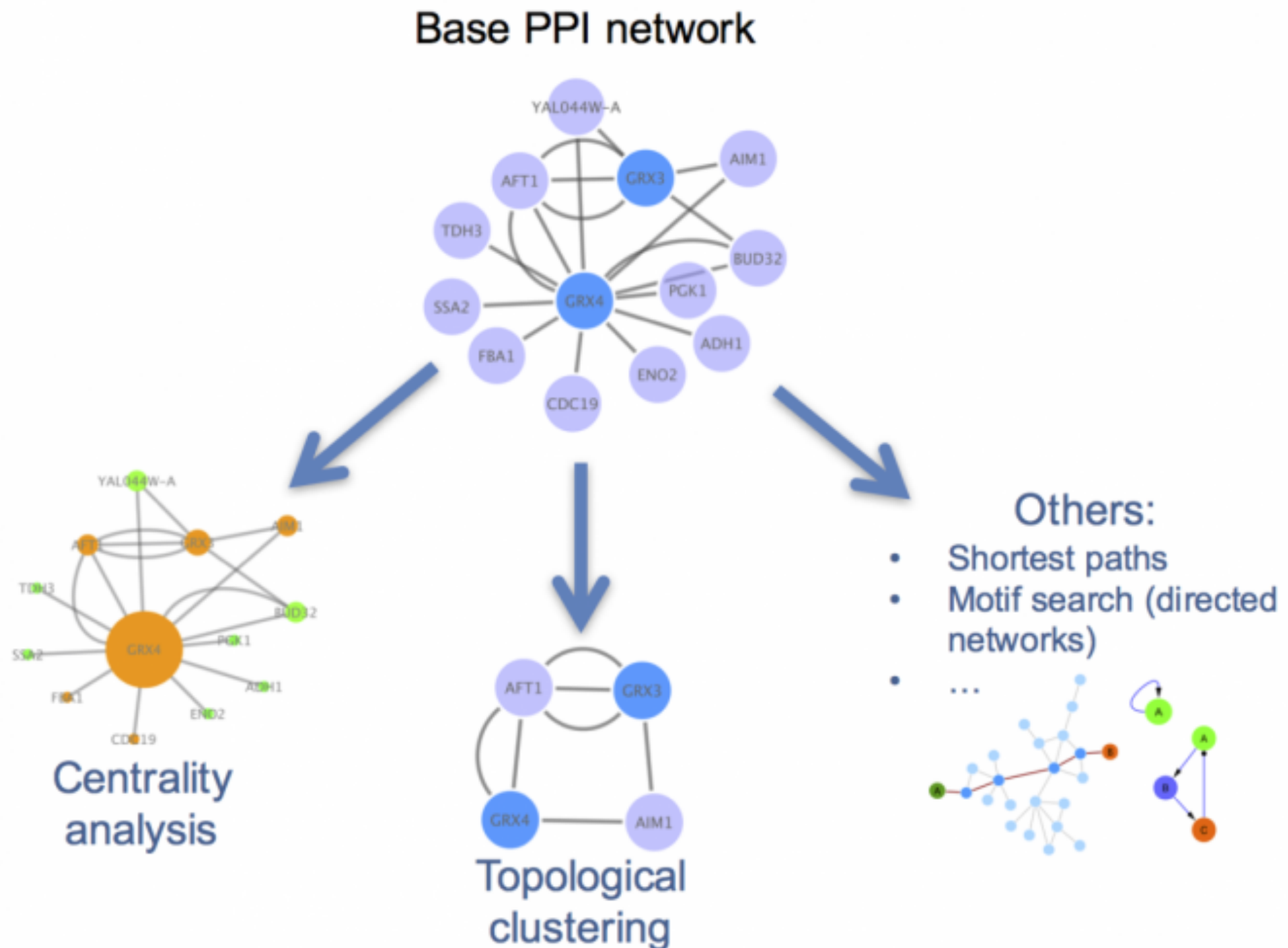
# Scale-Free Networks are Robust

- Complex systems (cell, internet, social networks), are resilient to component failure
- Network topology plays an important role in this robustness
  - Even if ~80% of nodes fail, the remaining ~20% still maintain network connectivity
- *Attack vulnerability* if hubs are selectively targeted
- In yeast, only ~20% of proteins are lethal when deleted, and are 5 times more likely to have degree  $k > 15$  than  $k < 5$ .

# Implications

- Many biological networks (protein-protein interaction networks regulatory networks, etc...) are thought to have hubs, or nodes with high degree.
- For protein-protein interaction networks (PPIs) these hubs have been shown to be older [1] and more essential than random proteins [2]
  - ➔ [1] Fraser et al. *Science* (2002) 296:750
  - ➔ [2] Jeoung et al. *Nature* (2001) 411:41

Analyzing the topological features of a network is a useful way of identifying relevant participants and substructures that may be of biological significance.

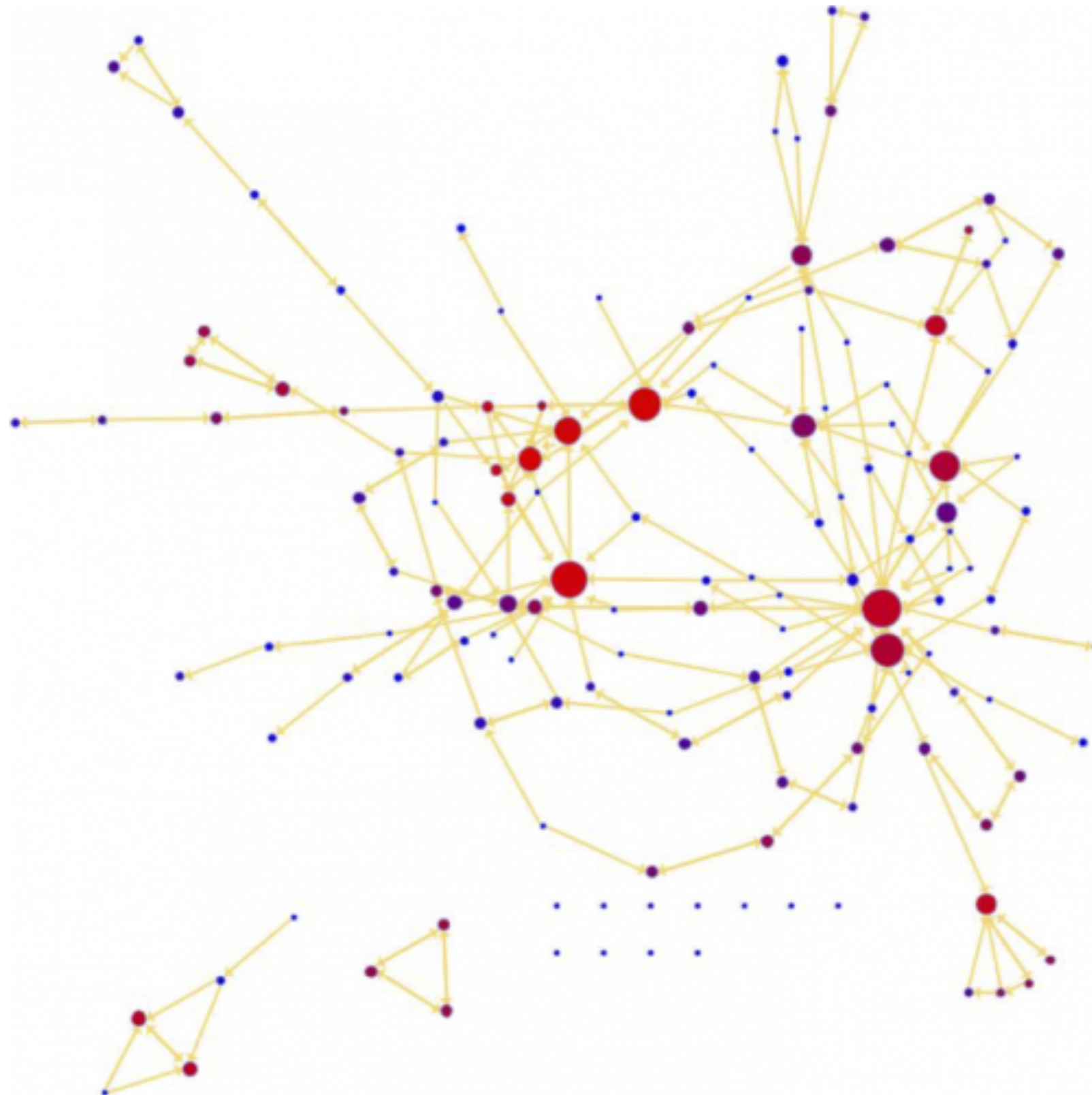


# Centrality analysis

- Centrality gives an estimation on how important a node or edge is for the connectivity or the information flow of the network
- It is a useful parameter in signalling networks and it is often used when trying to find drug targets.
- Centrality analysis in PPIs usually aims to answer the following question:
  - ➔ Which protein is the most important and why?

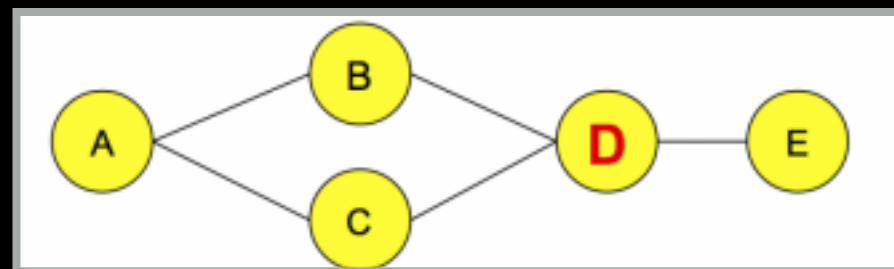


Bigger, redder nodes have higher **centrality values** in this representation.



# Betweenness centrality

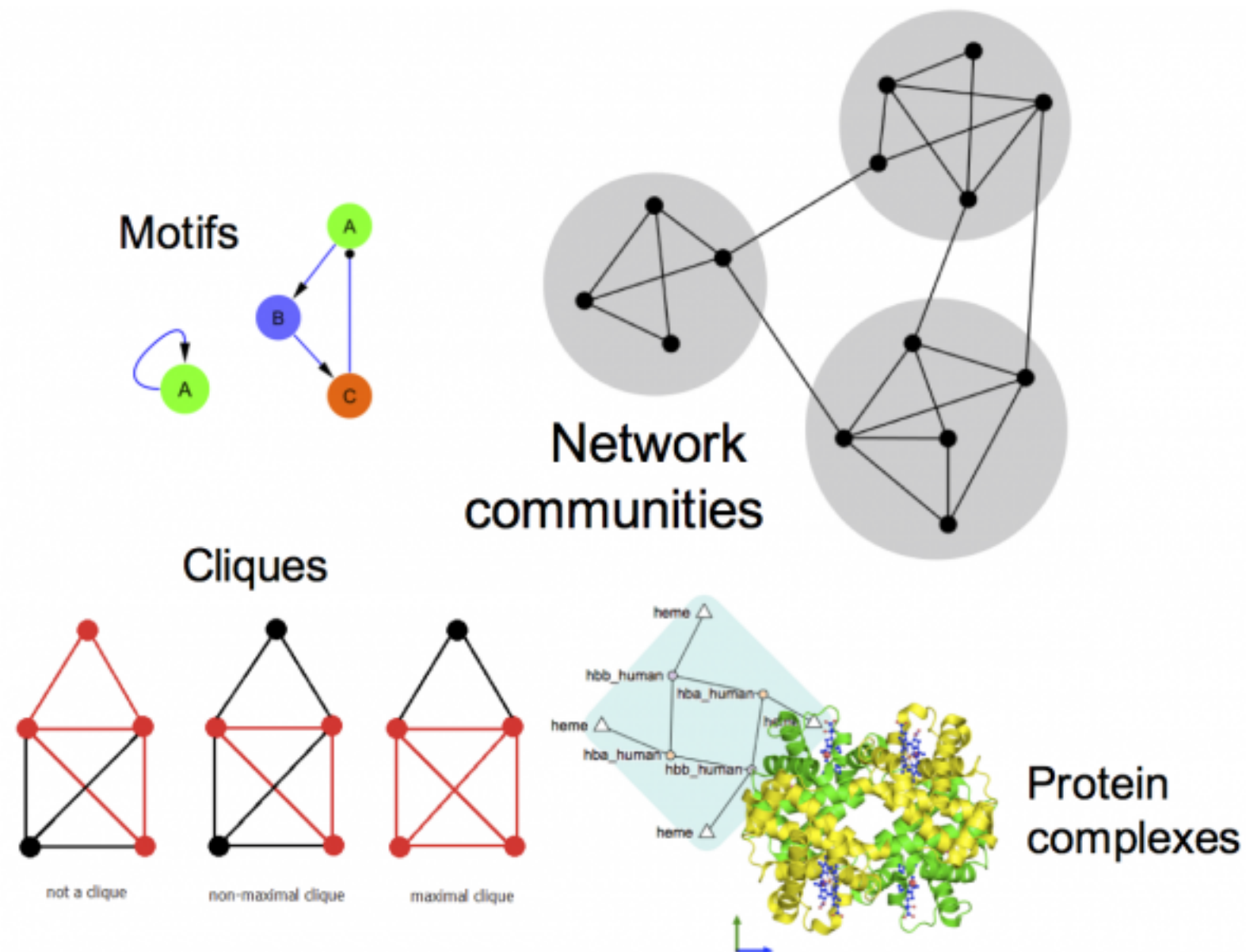
- Nodes with a high betweenness centrality are interesting because they lie on communication paths and can control information flow.
- The number of shortest paths in the graph that pass through the node divided by the total number of shortest paths.
- Betweenness centrality measures how often a node occurs on all shortest paths between two nodes.



# Community analysis

- **Community:** A general, catch-all term that can be defined as a group (i.e. *cluster*) of nodes that are more connected within themselves than with the rest of the network. The precise definition for a community will depend on the method or algorithm used to define it.

Looking for communities in a network is a nice strategy for reducing network complexity and extracting functional modules (e.g. protein complexes) that reflect the biology of the network.



# TODAYS MENU:

- ▶ Network introduction
- ▶ Network visualization
- ▶ Network analysis
- ▶ **Hands-on:**
  - Cytoscape and R (igraph) software tools for network visualization and analysis

# Hands-on: Part 2

[https://bioboot.github.io/bggn213\\_W20/lectures/#17](https://bioboot.github.io/bggn213_W20/lectures/#17)

- The data used in **part 2** is from an ocean metagenomic sequencing project - where all the genetic material in a sample of ocean water is sequenced.
- We will use the R package **igraph** and the bioconductor package **RCy3** together with Cytoscape.
- Many of these microbial species in these types of studies have not yet been characterized in the lab.
  - ➔ Thus, to know more about the organisms and their interactions, we can observe which ones occur at the same sites.
  - ➔ One way to do that is by using **co-occurrence networks** where you examine which organisms occur together at which sites.

# Practical issues

- Major tools for the **creation, manipulation** and **visualization** of biological networks include:
  - Cytoscape,
  - Gephi
  - R packages (igraph, graph, tidygraph, ggraph)

**Lab2**

- Tools for network analysis and modeling include:
  - Cytoscape apps/plugins
  - R packages (igraph and many others)
  - NetworkX (for Python)
  - ByoDyn, COPASI



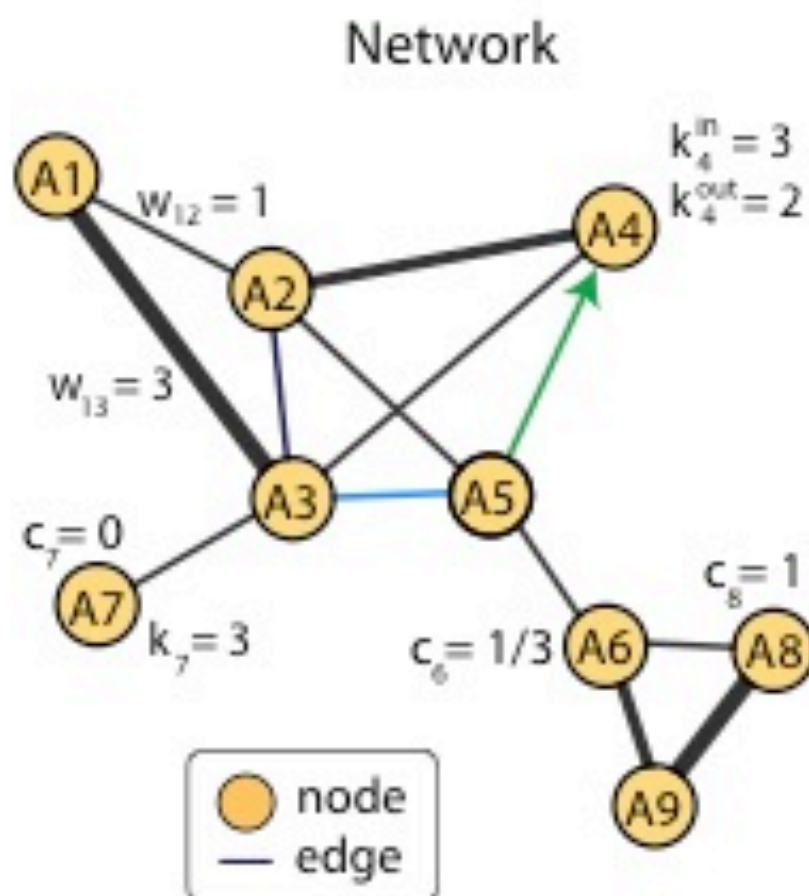
## Our Input

Relationships	Optional weight
A1 ↔ A2	1
A1 ↔ A3	3
A2 ↔ A3	1
A2 ↔ A4	2
A2 ↔ A5	1
A3 ↔ A4	1
A3 ↔ A5	1
A3 ↔ A7	1
A5 → A4	1
A5 ↔ A6	1
A6 ↔ A8	1
A6 ↔ A9	2
A8 ↔ A9	3

1

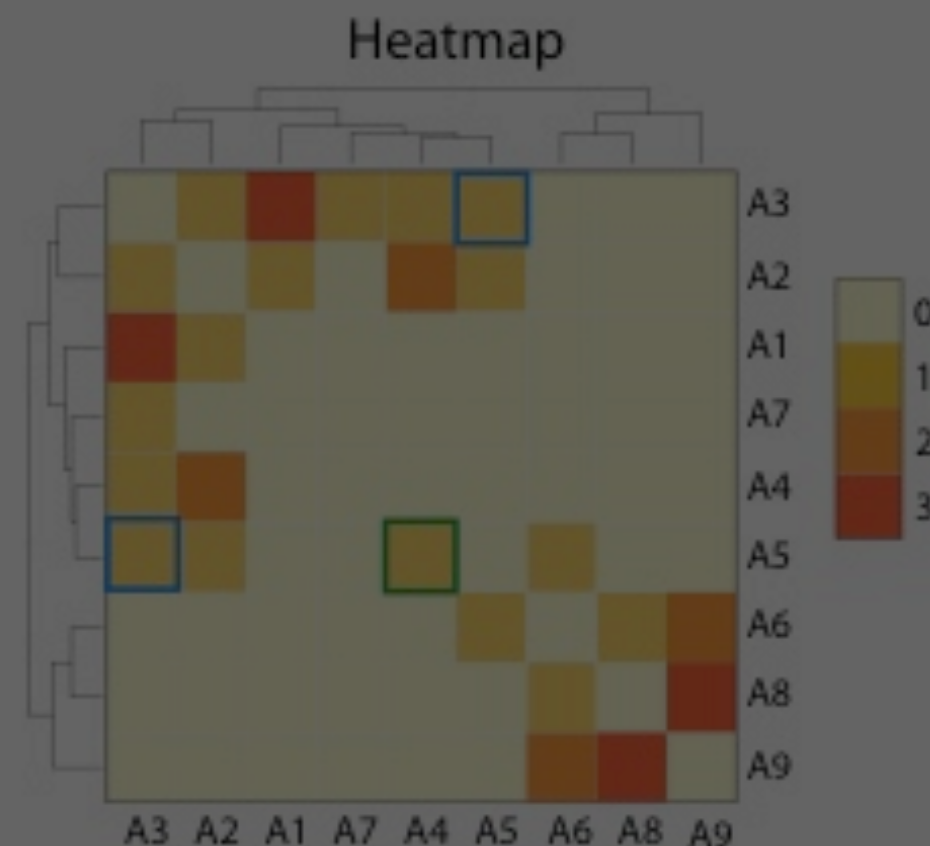
List of relationships

## Our Output



2

Network view



3

Adjacency matrix view

**Network view is most useful when network is sparse!**

# Summary

- Network biology makes use of **graph theory** to represent and analyze complex biological systems as a set of **nodes** and **edges**.
- Major types of biological networks include: genetic, metabolic, cell signaling etc.
- Biological networks have a number of characteristics, mainly:
  - ➔ **Scale-free**: A small number of nodes (**hubs**) are a lot more connected than the average node.
  - ➔ **Transitivity**: The networks contain **communities** of nodes that are more connected internally than they are to the rest of the network.
- Two of the most used topological methods are:
  - ➔ **Centrality analysis**: Which identifies the most important nodes in a network, using different ways to calculate centrality.
  - ➔ **Community detection**: Which aims to find heavily inter-connected components that may represent e.g. protein complexes and machineries

# Summary cont...

- **Cytoscape** is very useful for network visualization and:
  - ➔ Provides basic network manipulation features
  - ➔ Plugins/Apps are available to extend analysis functionality
- The R **igraph** package has extensive network analysis functionality beyond that in Cytoscape
- The R bioconductor **RCy3** package allows us to bring networks and associated data from R to Cytoscape so we can have the best of both worlds.
- Other popular tools include Gephi and NetworkX

# Network Analysis Overview

