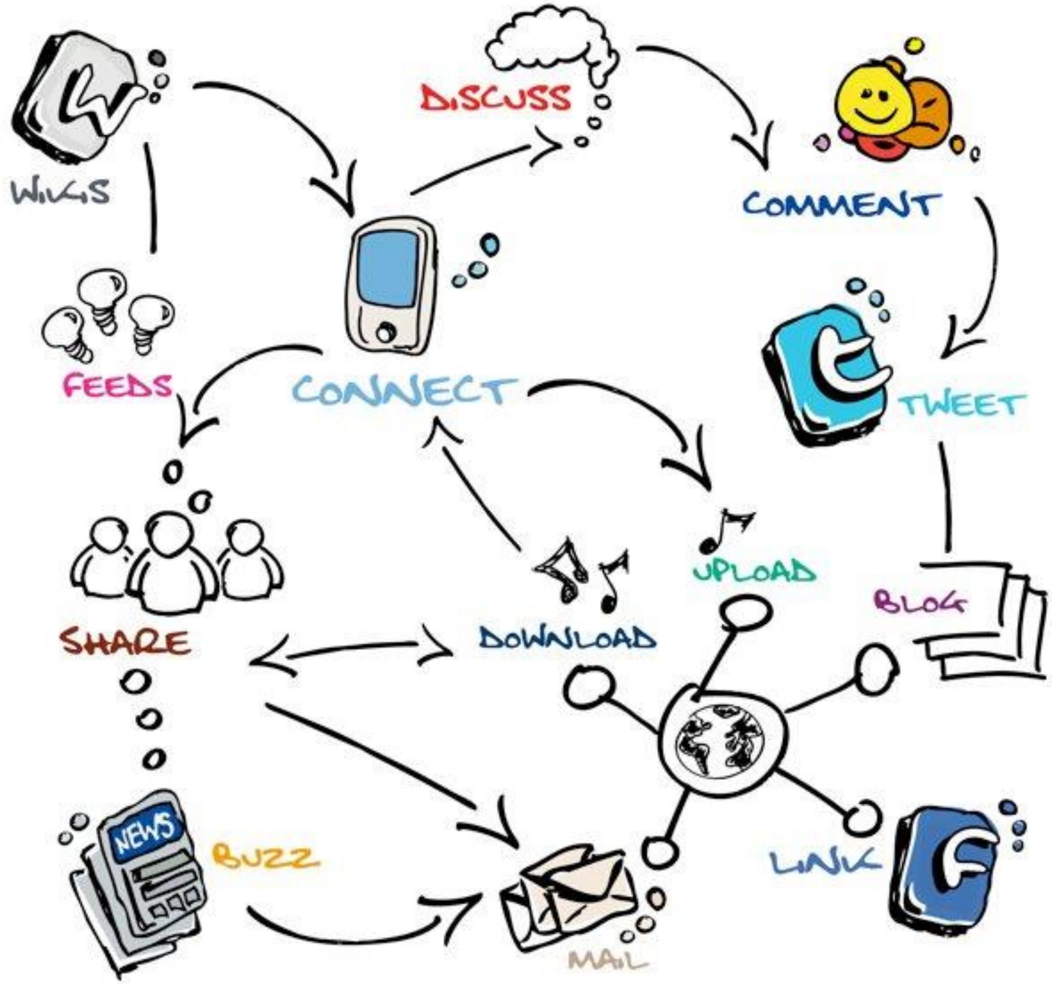
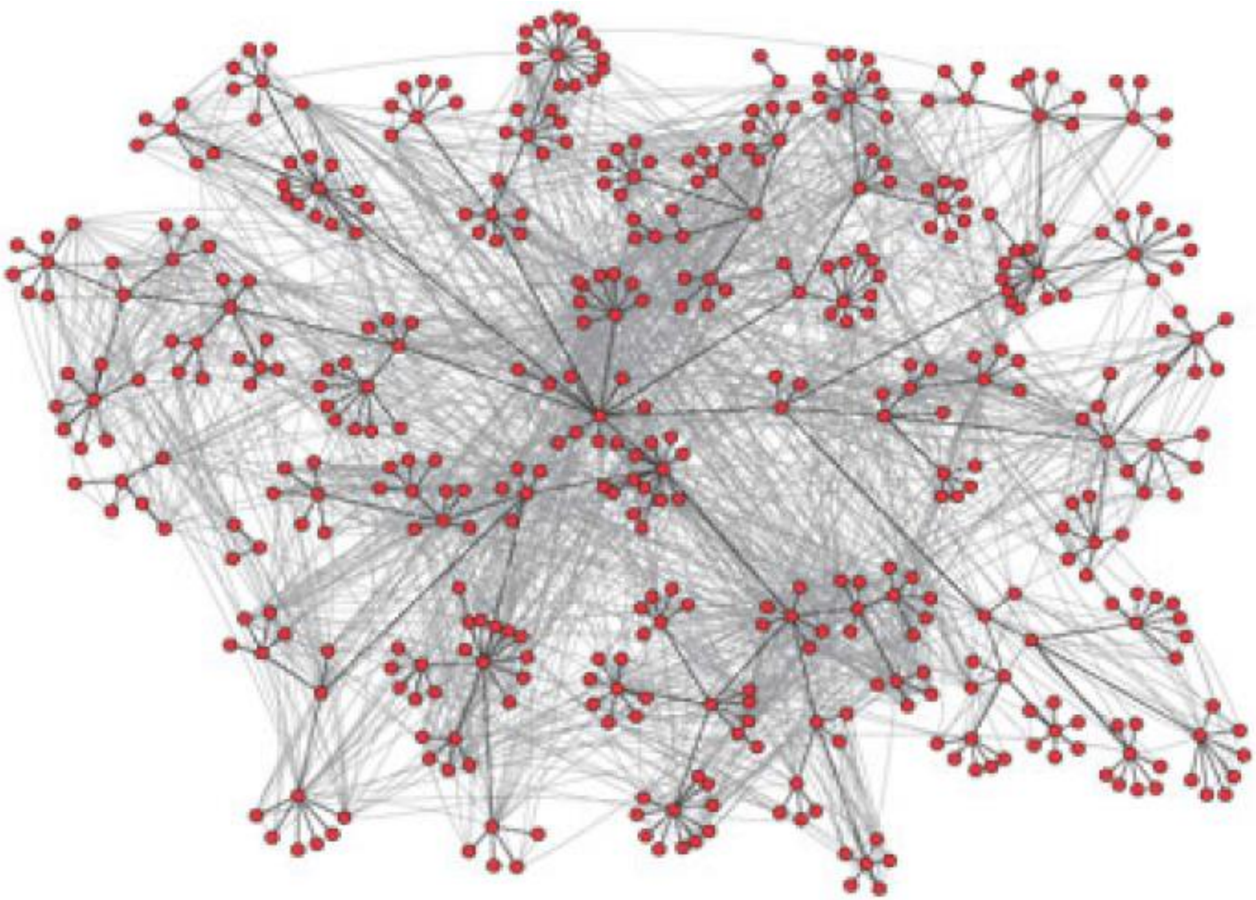


Structure of the Web Graph

Ahmet Onur Durahim

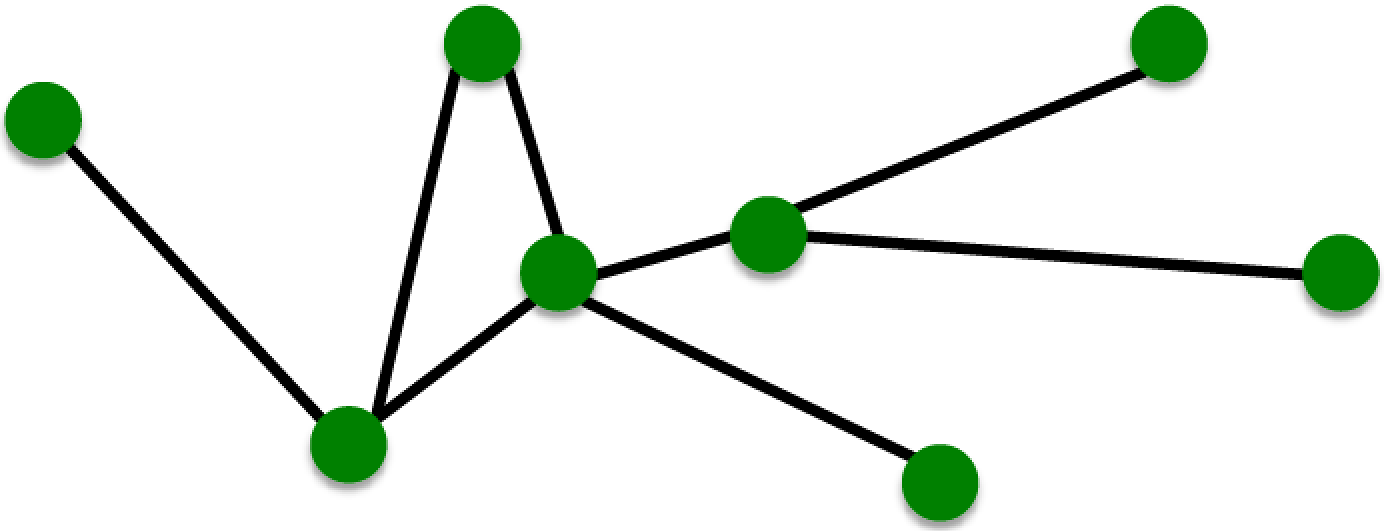
Structure of Networks?



Network is a collection of objects where some pairs of objects are connected by links

What is the structure of the network?

Components of a Network



- **Objects:** nodes, vertices — ***N***
- **Interactions:** links, edges — ***E***
- **System:** network, graph — ***G(N,E)***

Networks or Graph?

- **Network** often refers to real systems
  - Web, Social network, Metabolic network

  Language: Network, node, link
- **Graph** is mathematical representation of a network
  - Web graph, Social graph (a Facebook term)

  Language: Graph, vertex, edge

We will try to make this distinction whenever it is appropriate, but in most cases we will use the two terms interchangeably

Networks or Graph?

- **Network** often refers to real systems
  - Web, Social network, Metabolic network

  Language: Network, node, link

- **Gra** … of

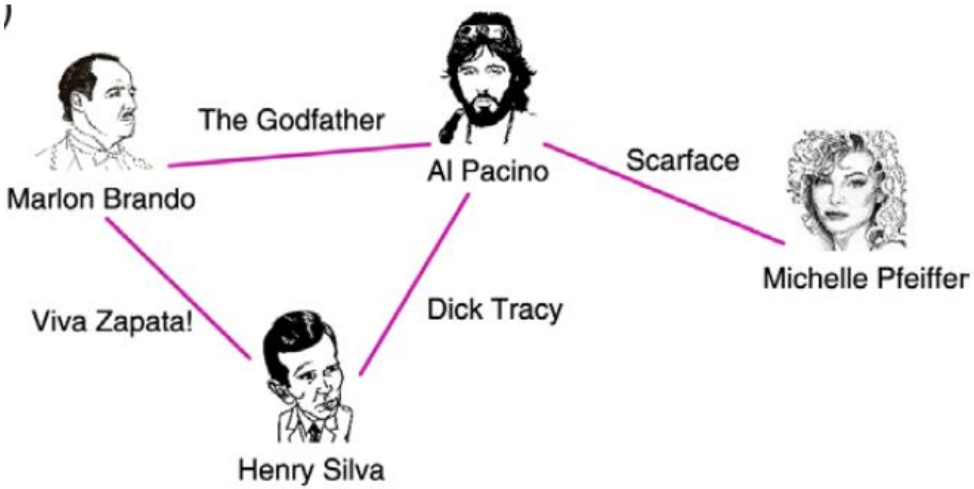
  a ne …
  - W … m)

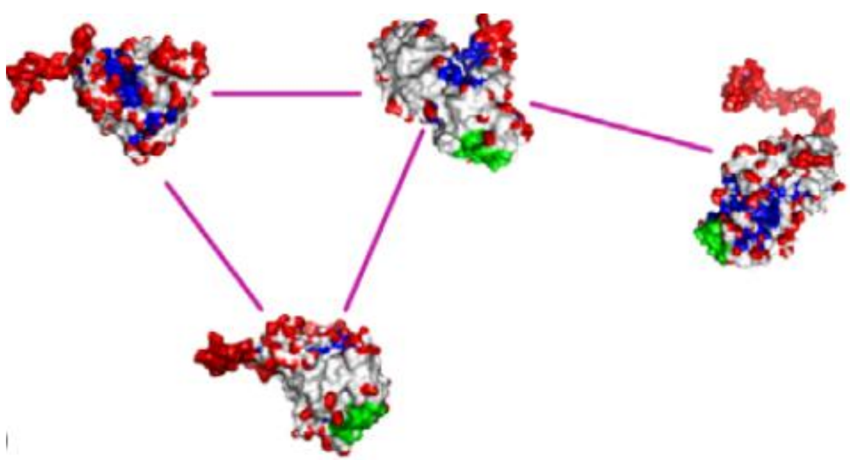
  Language: Graph, vertex, edge

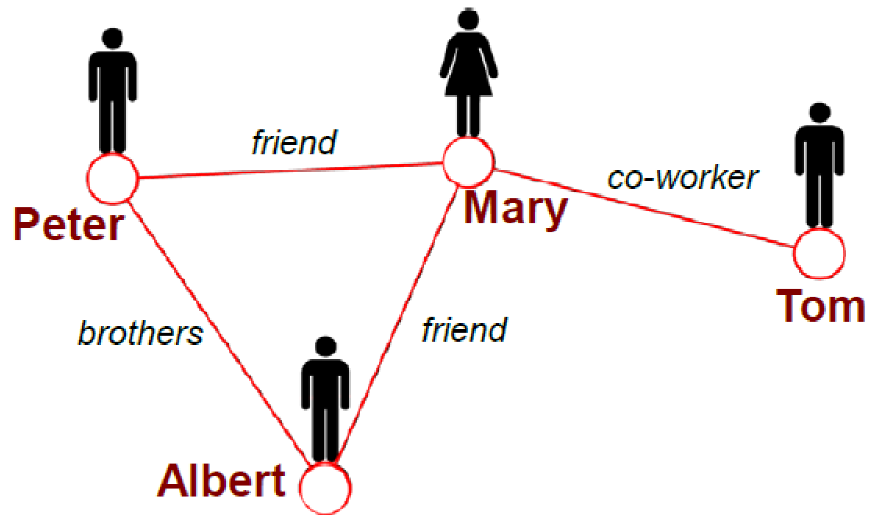
Network Science	Graph Theory
network	graph
node	vertex
link	edge

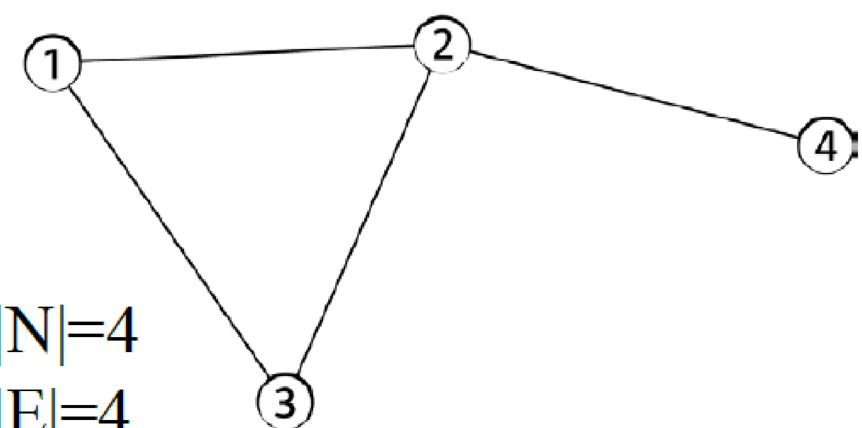
We will try to make this distinction whenever it is appropriate, but in most cases we will use the two terms interchangeably

Networks: Common Language









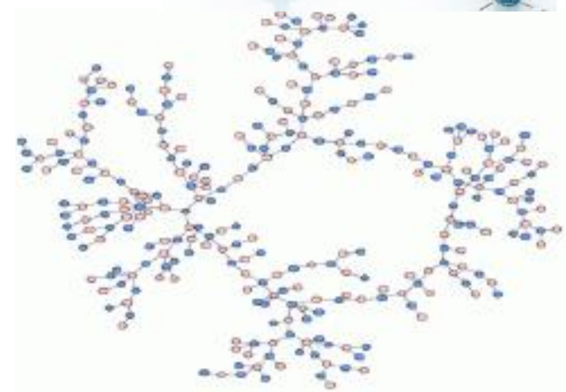
$|N|=4$

$|E|=4$

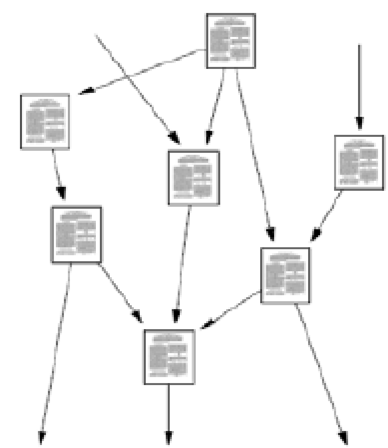
Choosing Proper Representation

- If you connect individuals that work with each other, you will explore a **professional network**

- If you connect those that have a friend relationship, you will be exploring **friendship networks**



- If you connect scientific papers that cite each other, you will be studying the **citation network**



- If you connect all papers with the same word in the title, you will be exploring what?
  - It is a network, nevertheless

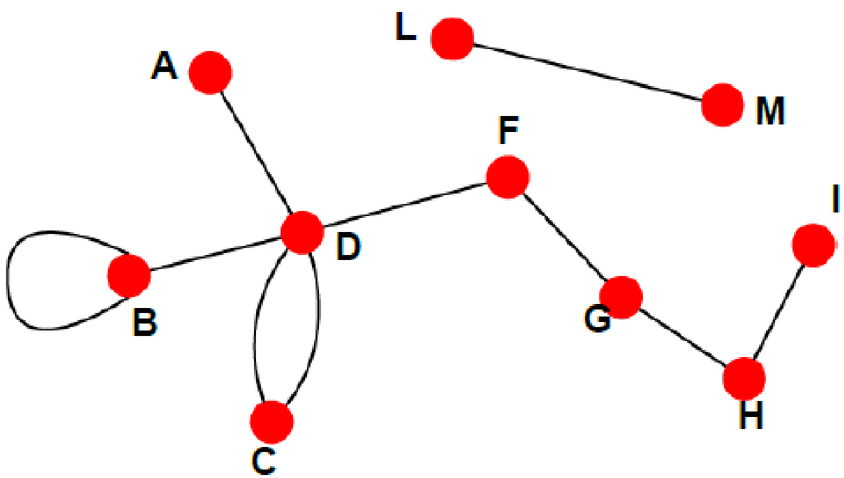
Choosing Proper Representation

- **How to build a graph:**
  - What are nodes?
  - What are edges?
- ***Choice of the proper network representation*** of a given domain/problem determines our ability to use networks successfully:
  - In some cases there is a unique, unambiguous representation
  - In other cases, the representation is by no means unique
  - *The way you assign links* will determine the nature of the question you can study

Undirected vs. Directed Networks

Undirected

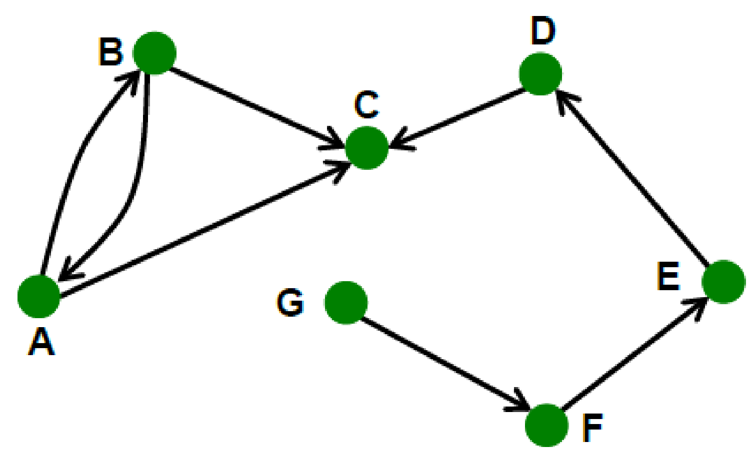
- Links: undirected (symmetrical, reciprocal)



- **Examples:**
  - Collaborations
  - Friendship on Facebook

Directed

- Links: directed (arcs)

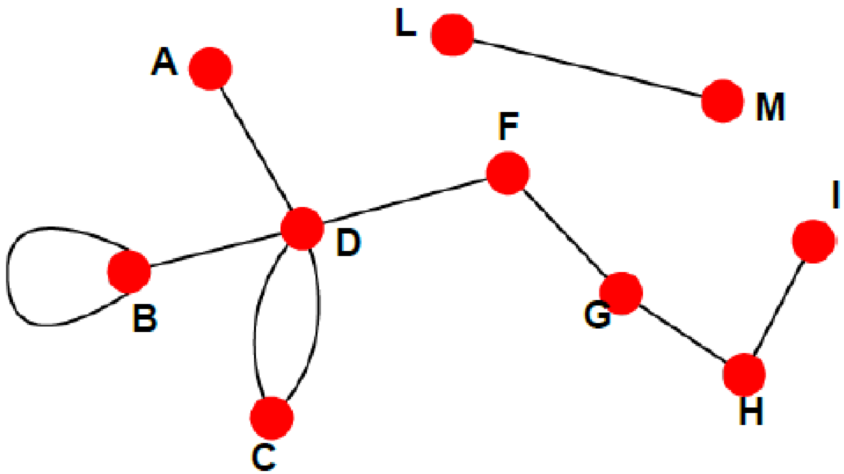


- **Examples:**
  - Phone calls
  - Following on Twitter

Undirected vs. Directed Networks

Undirected

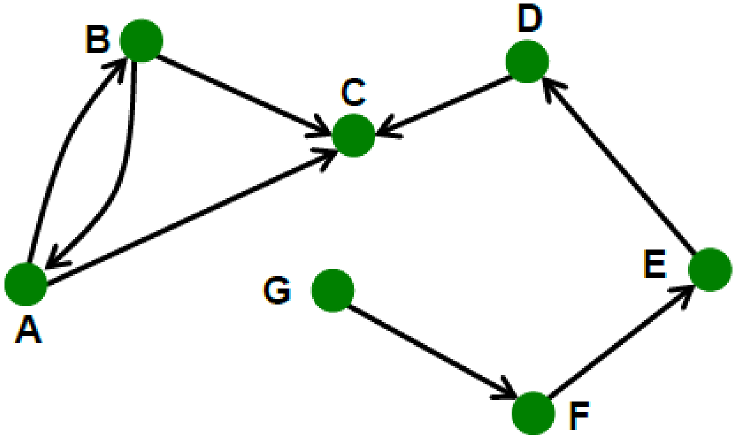
- **Links:** undirected (symmetrical, reciprocal)



- **Examples:**
  - A and D like each other
  - D and F are siblings/co-authors

Directed

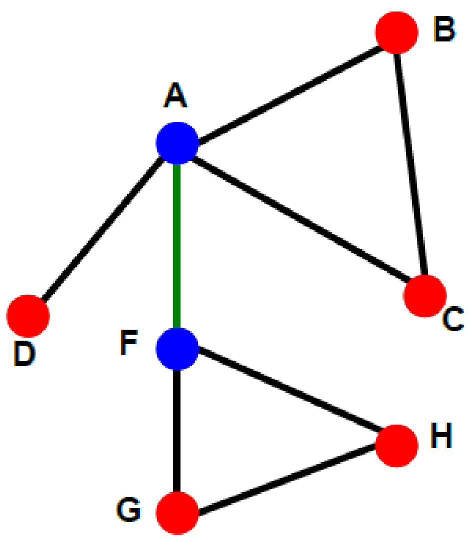
- **Links:** directed (arcs)

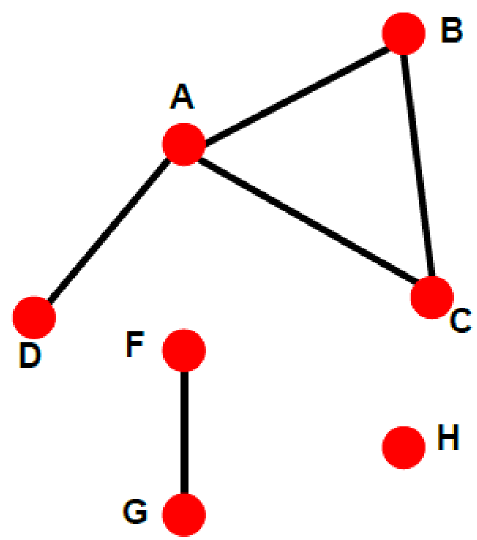


- **Examples:**
  - A likes B
  - C is B's child

Connectivity of Graphs

- **Connected (undirected) graph:**
  - Any two vertices can be joined by a path
- A ***disconnected graph*** is made up by two or more ***connected components***





Largest Component:
Giant Component

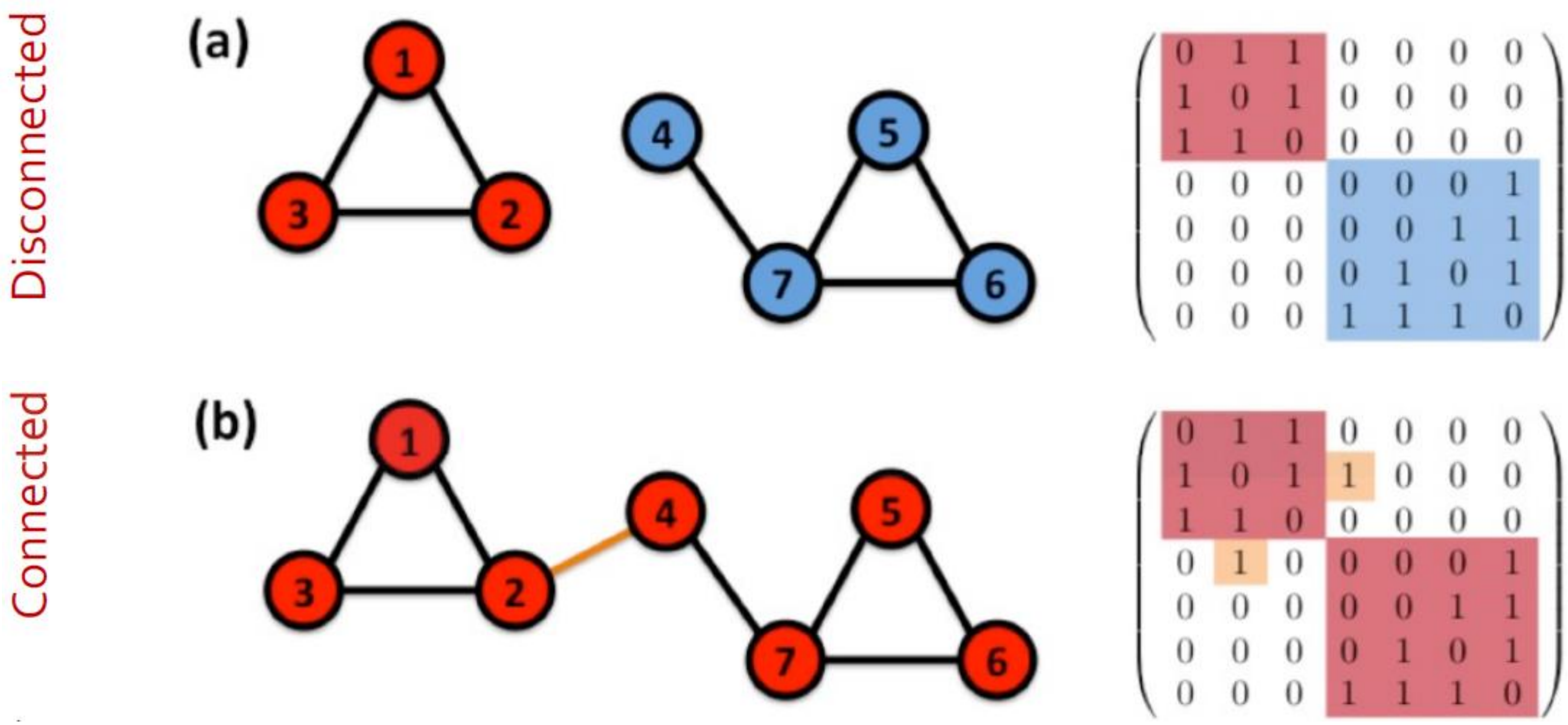
Isolated node (node H)

Bridge edge: If we erase it, the graph becomes disconnected

Articulation point: If we erase it, the graph becomes disconnected

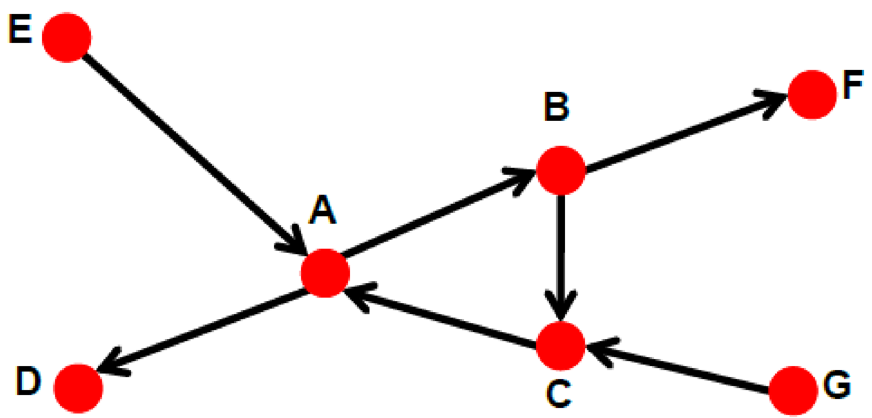
Connectivity of Graphs

- The adjacency matrix of a network with ***several components*** can be written in a block-diagonal form
  - so that nonzero elements are confined to ***squares***
  - with all other elements being zero



Connectivity of Directed Graphs

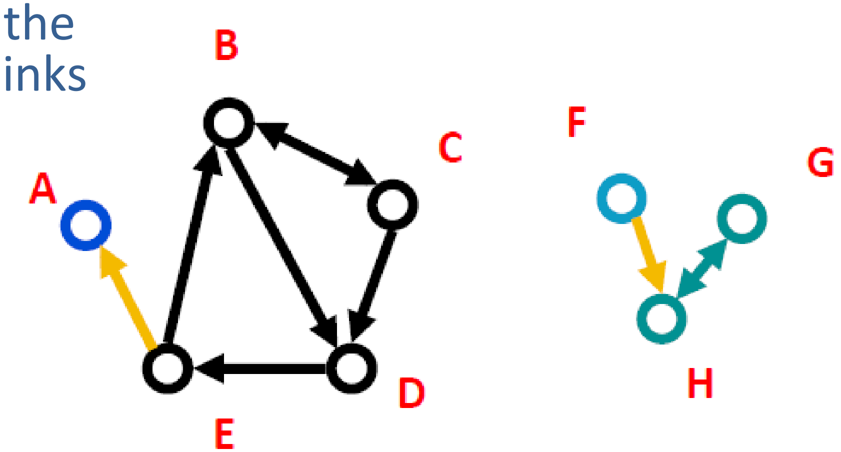
- **Strongly connected directed graph**
  - has a path from each node to every other node and vice versa (e.g., A-B path and B-A path)
- **Weakly connected directed graph**
  - is connected if we disregard the edge directions



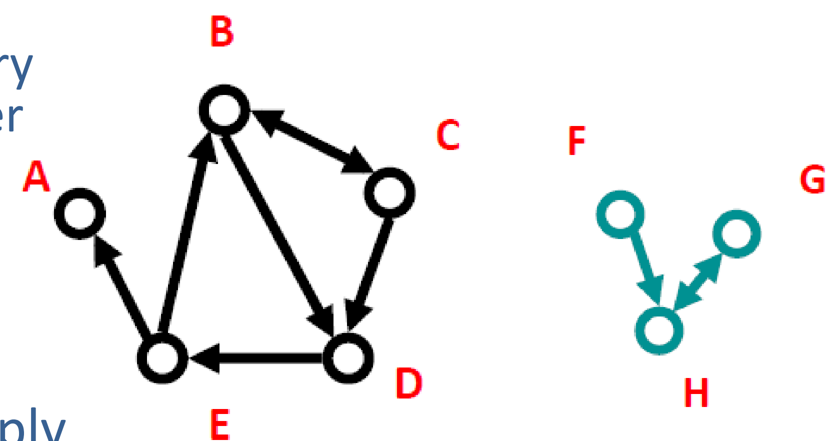
Graph on the left is connected but not strongly connected (e.g., there is no way to get from F to G by following the edge directions).

Connected Components

- **Strongly connected components**
  - Each node within the component can be reached from every other node in the component by following directed links
    - ***SCCs***
      - B C D E
      - A
      - G H
      - F

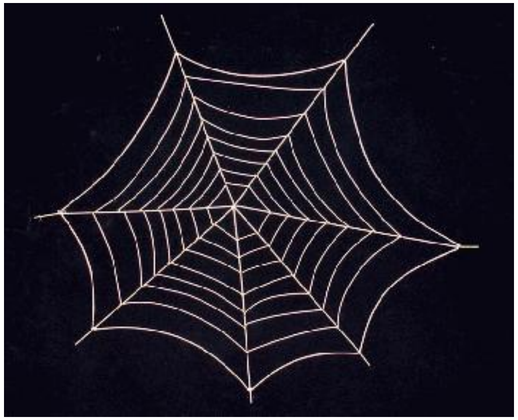


- **Weakly connected components**
  - Every node can be reached from every other node by following links in either direction
    - ***WCCs***
      - A B C D E
      - G H F



- In undirected networks one talks simply about "***connected components***"

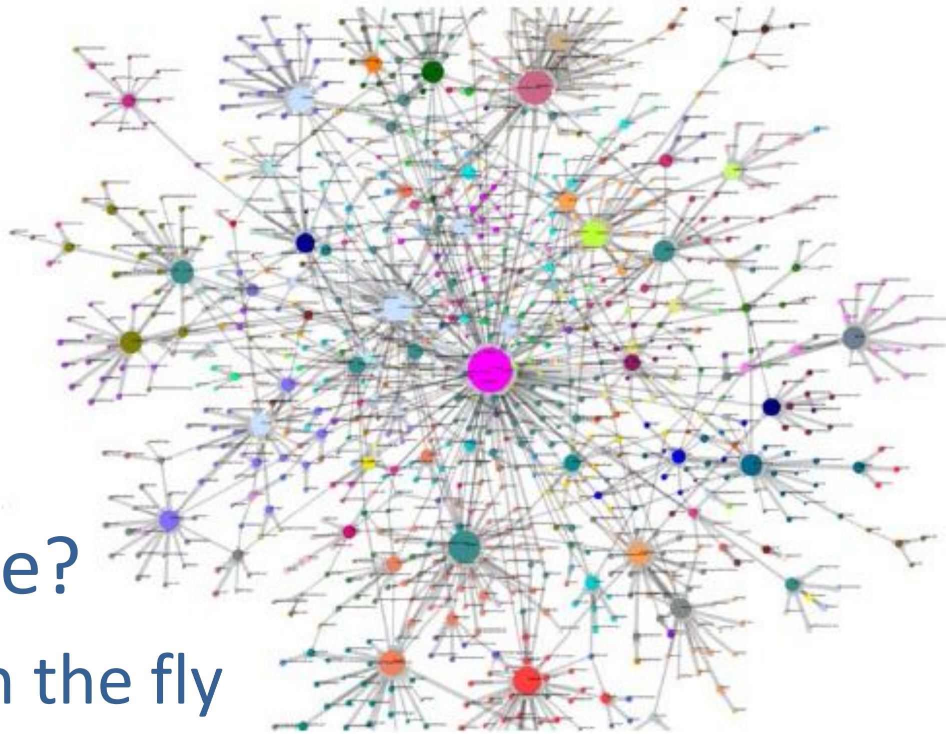
Web as a Graph



- **Q: What does the Web "look like"?**
- **Here is what we will do next:**
  - We will take a real system (i.e., the Web)
  - We will represent the Web as a graph
  - We will use language of graph theory to reason about the structure of the graph
  - Do a computational experiment on the Web graph
  - **Learn something about the structure of the Web!**

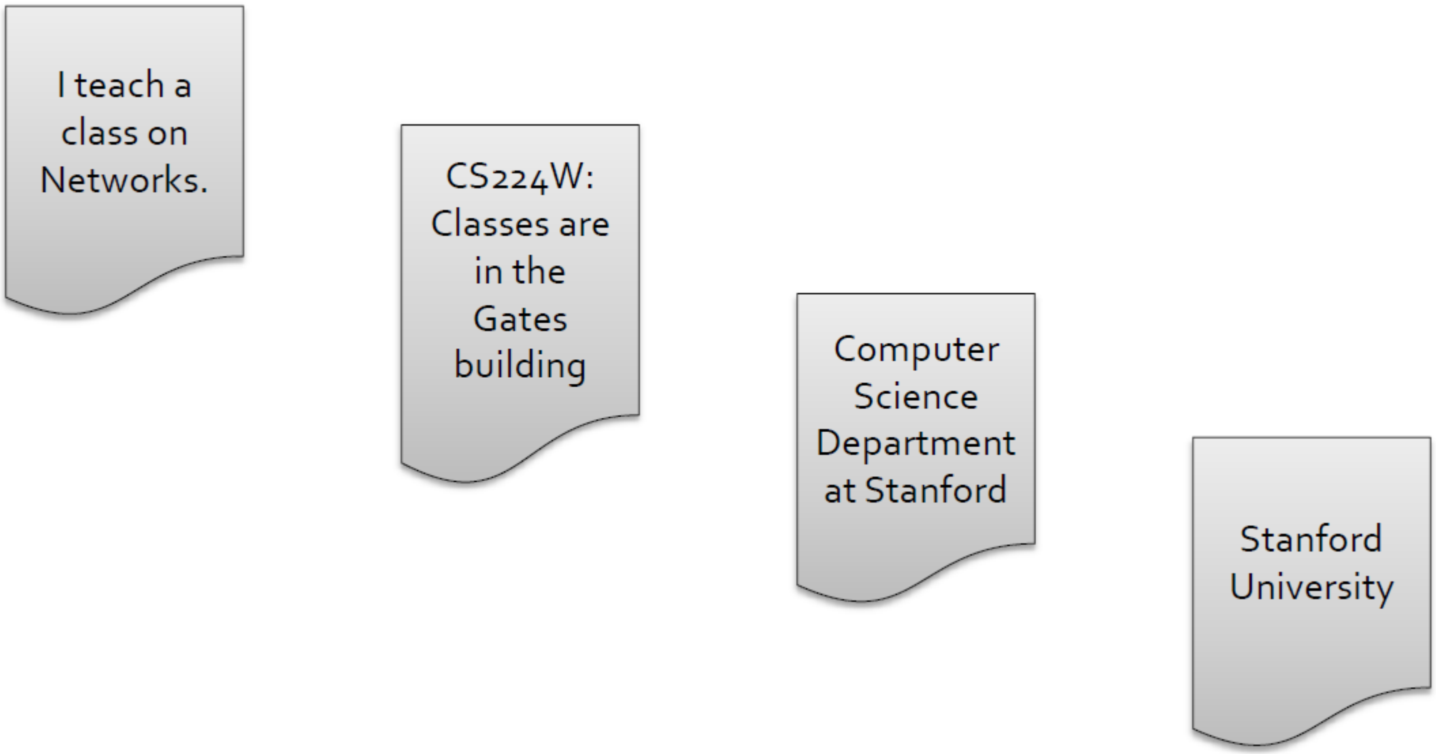
Web as a Graph

- **Q: What does the Web "look like" at a global level?**
- **Web as a graph:**
  - Nodes = web pages
  - Edges = hyperlinks

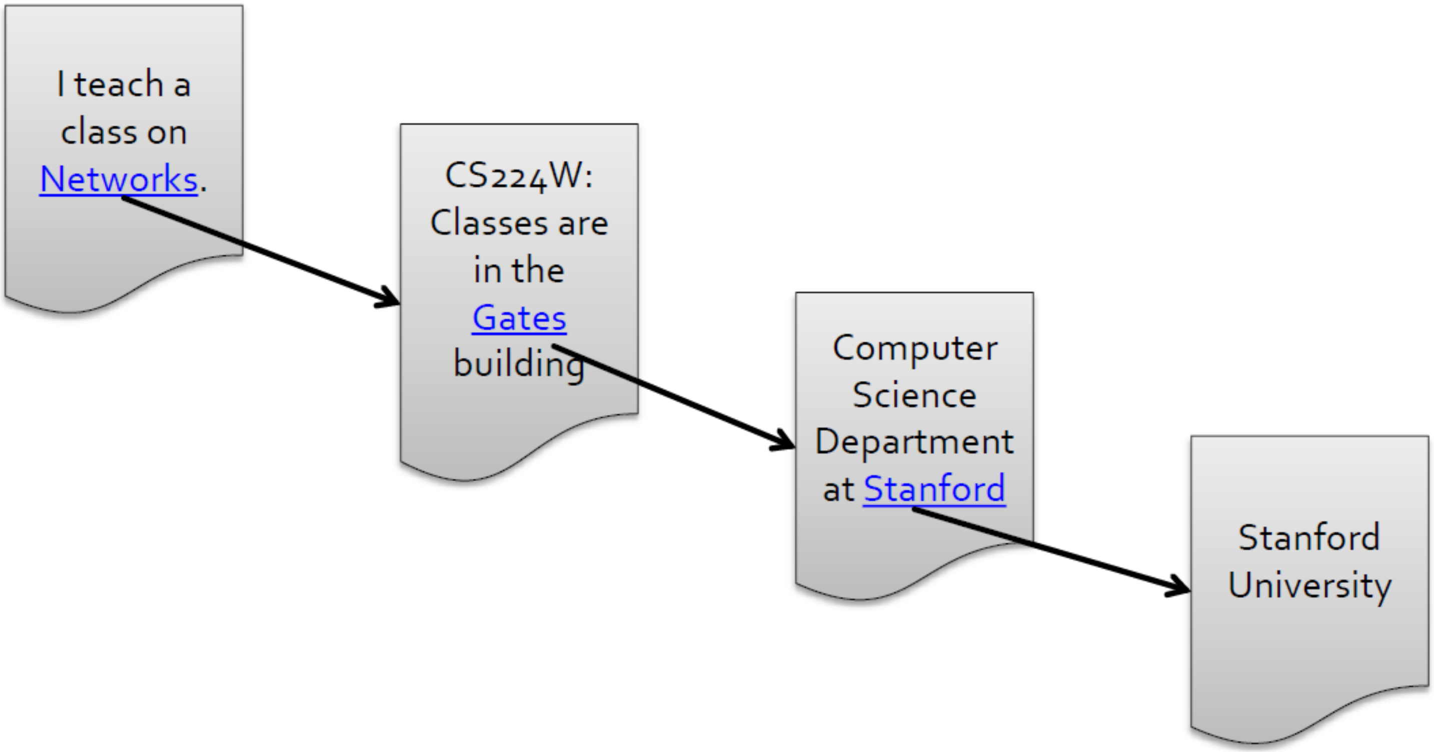


- **Side issue:** What is a node?
  - Dynamic pages created on the fly
  - "dark web" – inaccessible database generated pages

The Web as a Graph

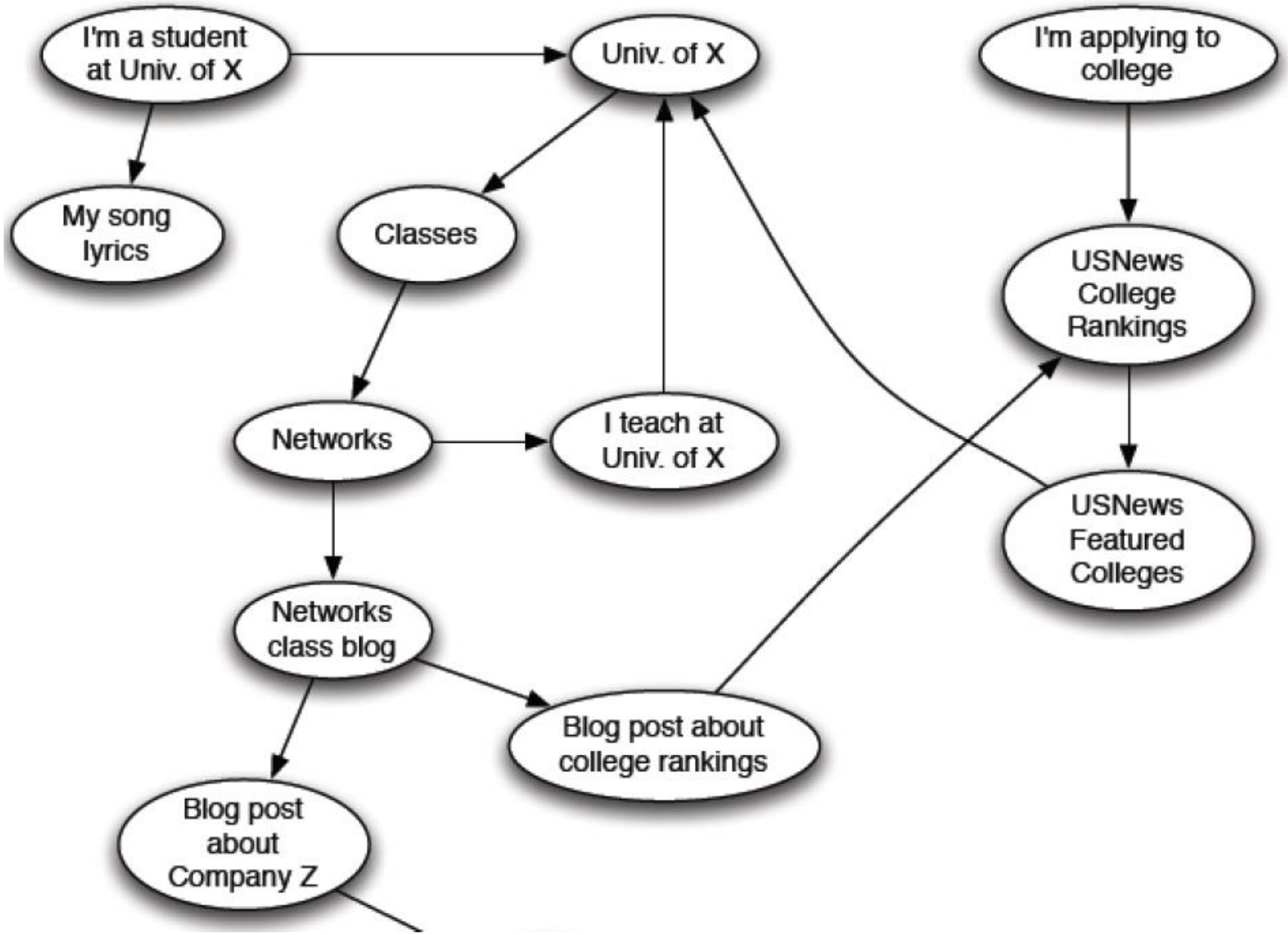


The Web as a Graph

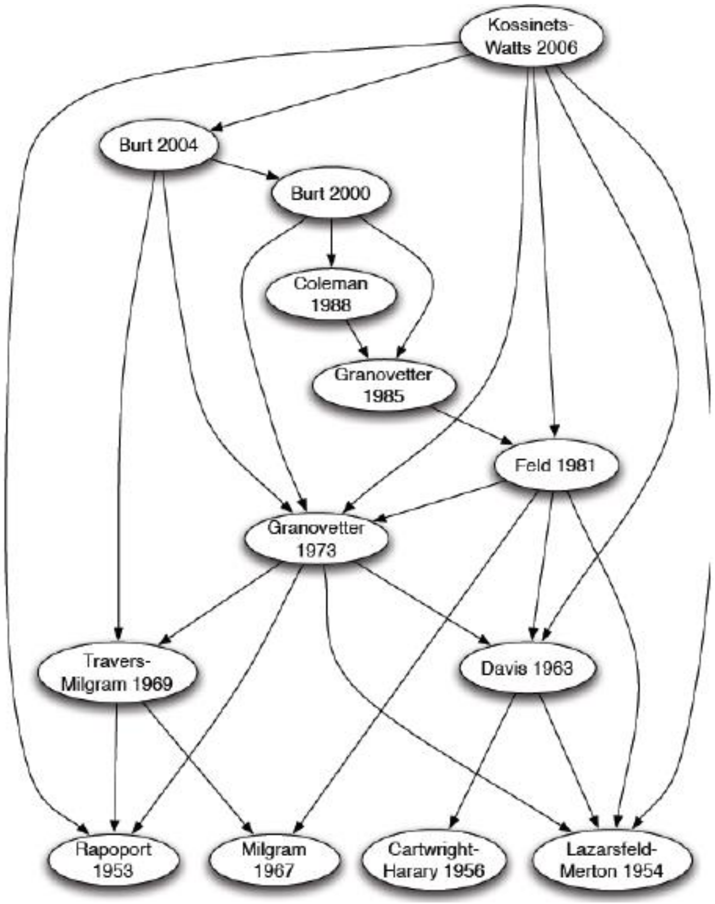


- In early days of the Web links were **navigational**
- Today many links are **transactional**

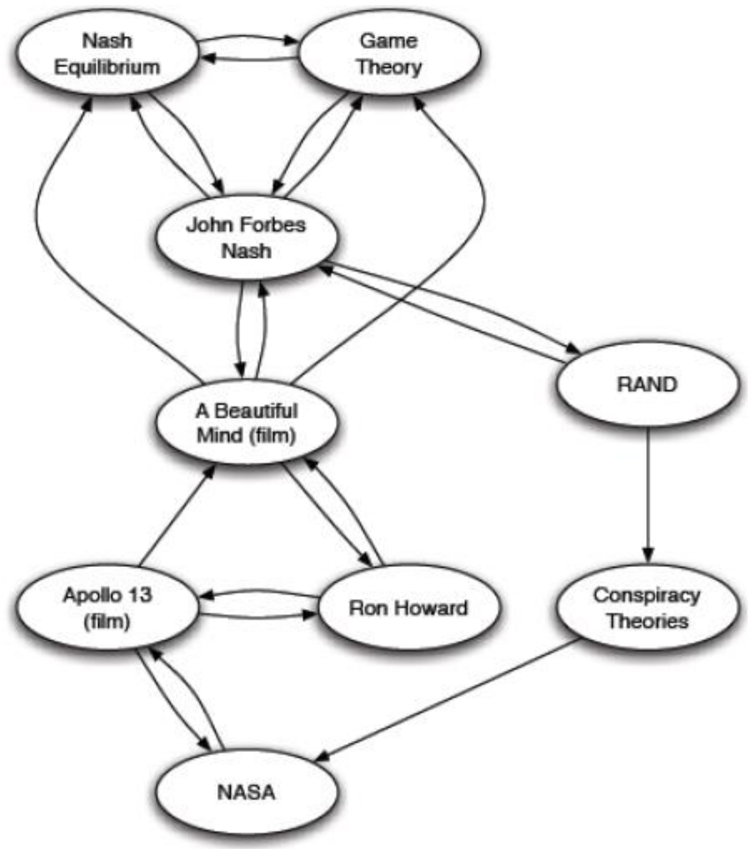
The Web as a Directed Graph



Other Information Networks



Citations



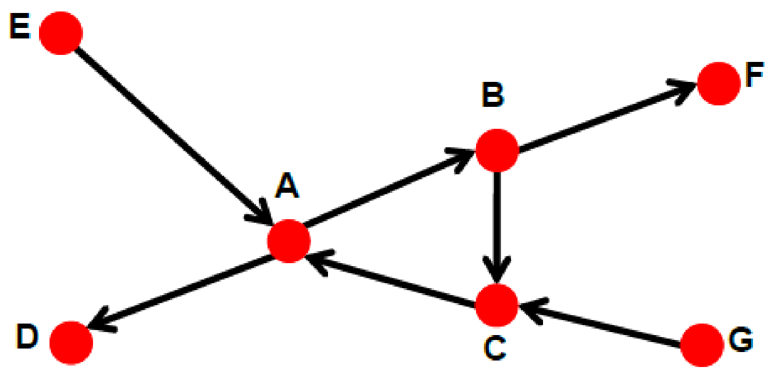
References in an Encyclopedia

What Does the Web Look Like?

- How is the Web linked?
- What is the "map" of the Web?

Web as a ***directed graph*** [Broder et al. 2000] [Revisited.2014] :

- Given node ***v***, what can ***v*** reach?
- What other nodes can reach ***v***?

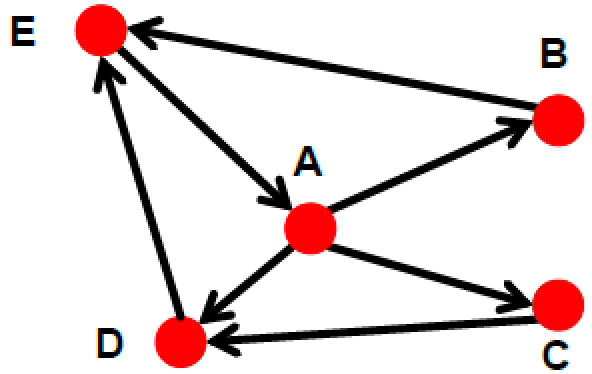


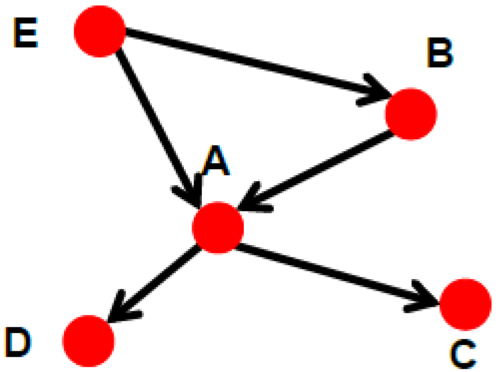
$$In(v) = \{w \mid w \text{ can reach } v\}$$
$$Out(v) = \{w \mid v \text{ can reach } w\}$$

For example:
In(A) = {A,B,C,E,G}
Out(A)={A,B,C,D,F}

Directed Graphs

- **Two types of directed graphs:**
  - **Strongly connected:**
    - Any node can reach any node via a directed path
    - In(A)=Out(A)={A,B,C,D,E}
  - **DAG – Directed Acyclic Graph:**
    - Has no cycles: if ***u*** can reach ***v***, then ***v*** can not reach ***u***
- *Any directed graph can be expressed in terms of these two types!*

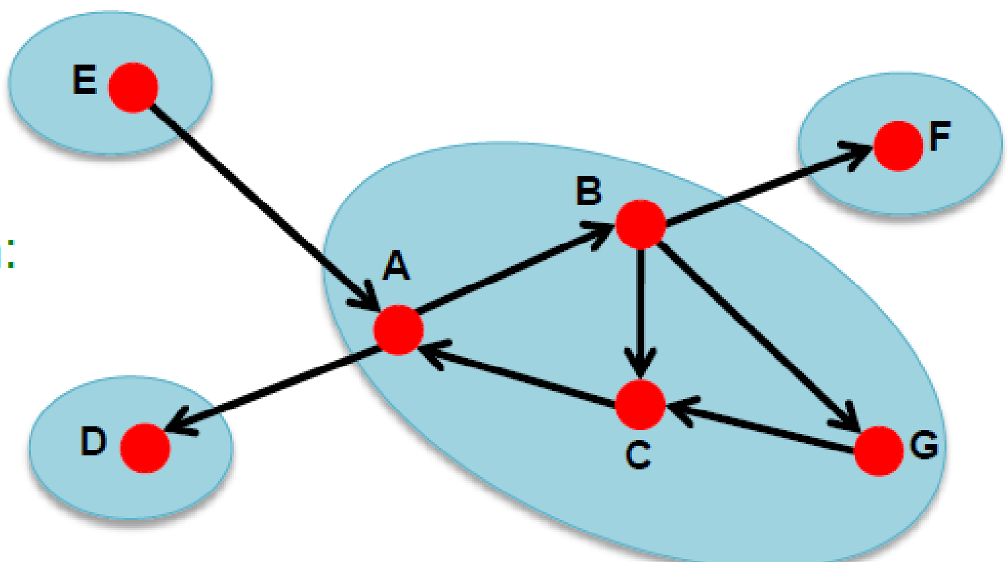




Strongly Connected Component

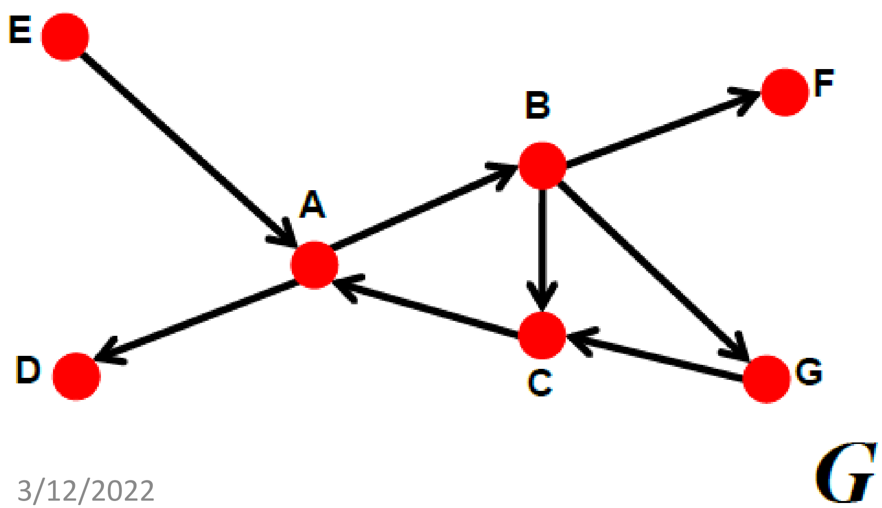
- **Strongly connected component (SCC)** is a set of nodes ***S*** so that:
  - Every pair of nodes in ***S*** can reach each other
  - There is no larger set containing ***S*** with this property

Strongly connected components of the graph: {A,B,C,G}, {D}, {E}, {F}



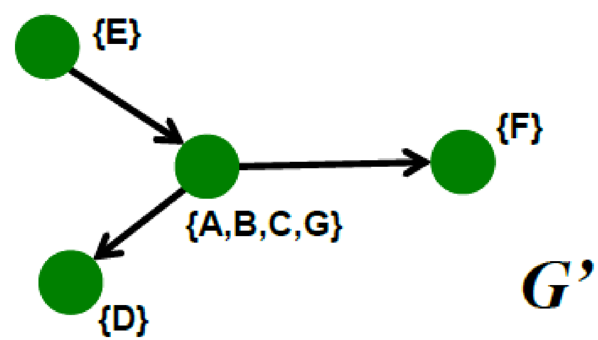
Strongly Connected Component

- **Fact: Every directed graph is a DAG on its SCCs**
  - **(1)** SCCs partitions the nodes of ***G***
    - each node is in exactly one SCC
  - **(2)** If we build a graph ***G'***
    - nodes are SCCs
    - edge between nodes of ***G'*** – if there is an edge between corresponding SCCs in ***G***
    - then ***=> G'*** is a ***DAG***



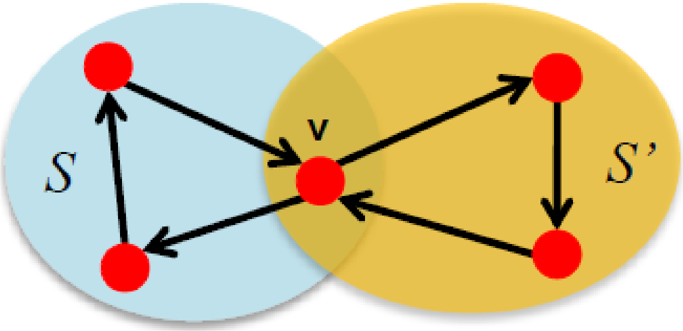
(1) Strongly connected components of graph G: {A,B,C,G}, {D}, {E}, {F}

(2) G' is a DAG:



Proof of Claim (1)

- **Claim: SCCs partitions nodes of G**
  - each node is member of exactly 1 SCC
- **Proof by contradiction:**
  - Suppose there exists a node v which is a member of two SCCs ***S*** and ***S'***

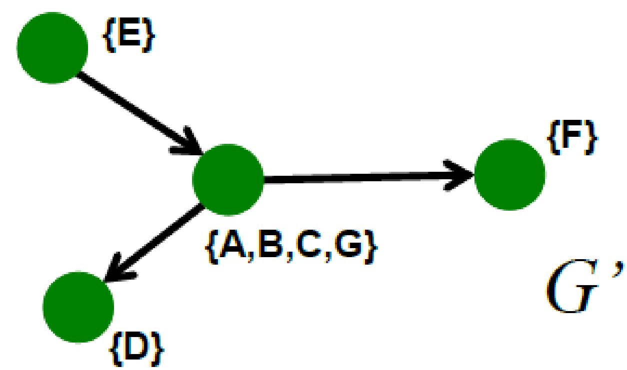


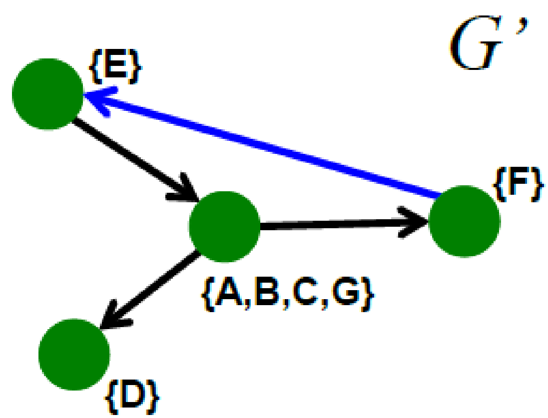
- But then ***S* U *S'*** is one large ***SCC***!
  - Contradiction!

Proof of Claim (2)

- **Claim:** **G' (graph of SCCs) is a DAG.**
  - this means => ***G'*** has ***no cycles***
- **Proof by contradiction:**
  - Assume ***G'*** is not a DAG
  - Then ***G'*** has a directed cycle
  - Now all nodes on the cycle are mutually reachable, and all are part of the same SCC
  - But then ***G'*** is not a graph of connections between SCCs

    (SCCs are defined as maximal sets)
    - Contradiction!



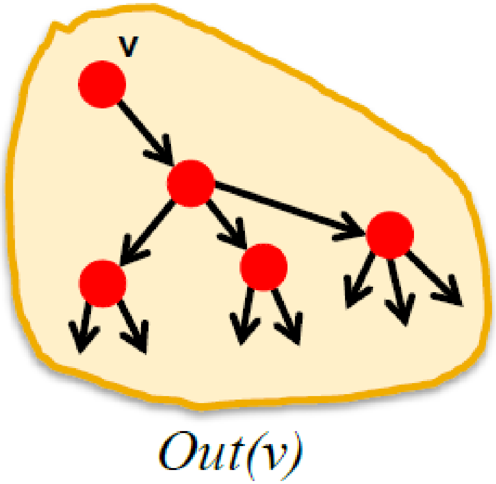


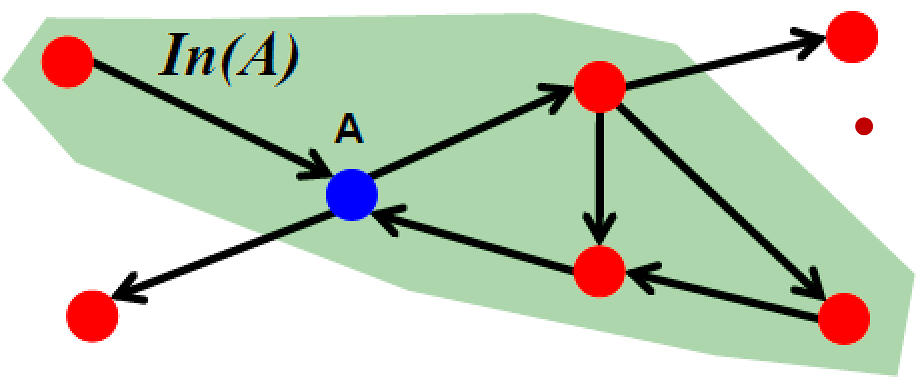
Now {A,B,C,G,E,F} is a SCC!

Graph Structure of the Web

- **Goal:** Take a large snapshot of the Web and try to understand how its SCCs "fit together" as a DAG
- **Computational issue:**
  - Want to find a SCC containing node ***v***?
  - **Observation:**
    - ***Out(v)*** … nodes that can be reached from ***v***
    - ***SCC containing v is***: Out(v) ∩ In(v)

    = Out(v,***G***) ∩ Out(v,***G'***), where ***G'*** is ***G*** with all edge directions flipped

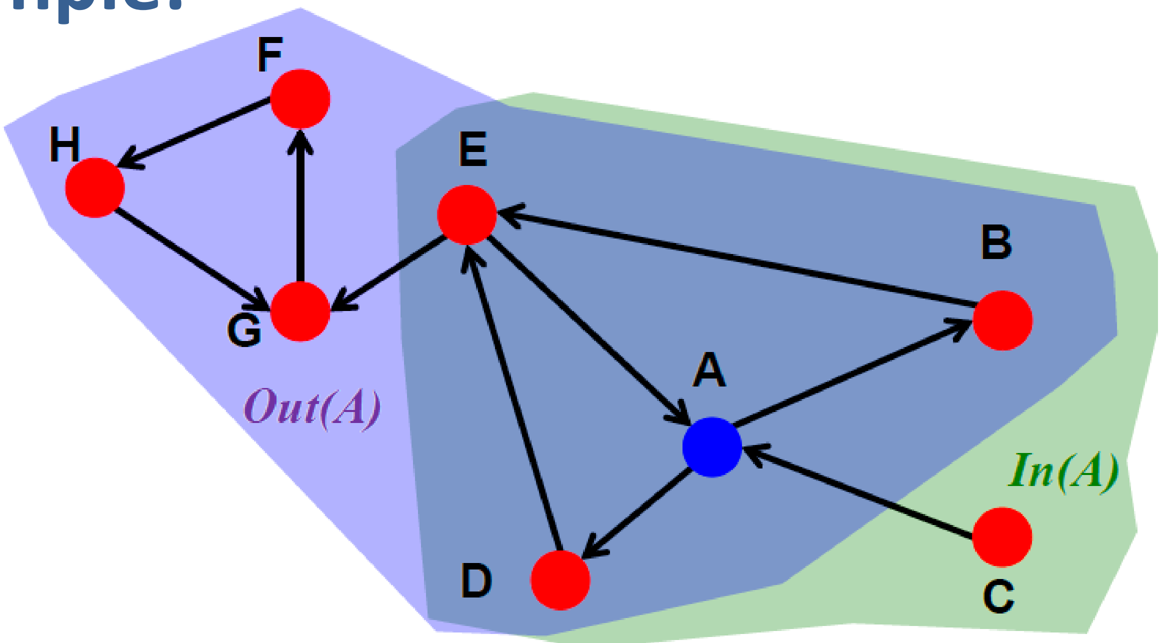




- In(v,***G***) = Out(v,***G'***)
  - ***G'*** is ***G*** with all edge directions flipped

Out(A) ∩ In(A) = SCC

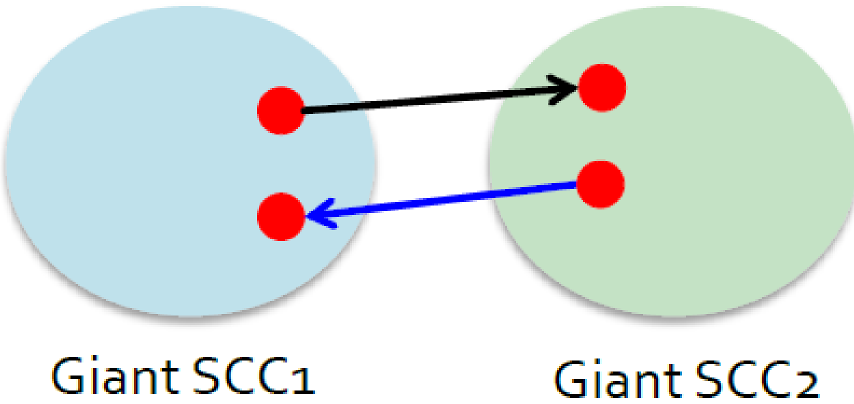
- **Example:**



  - Out(A) = {A, B, D, E, F, G, H}
  - In(A) = {A, B, C, D, E}
  - So => ***SCC(A)*** = Out(A) ∩ In(A) = {A, B, D, E}

Graph Structure of the Web

- **There is a single giant SCC**
  - that is => *there won't be two SCCs*
- **Heuristic argument:**
  - It just takes 1 page from one SCC to link to the other SCC
  - If the 2 SCCs have millions of pages the likelihood of this not happening is very very small



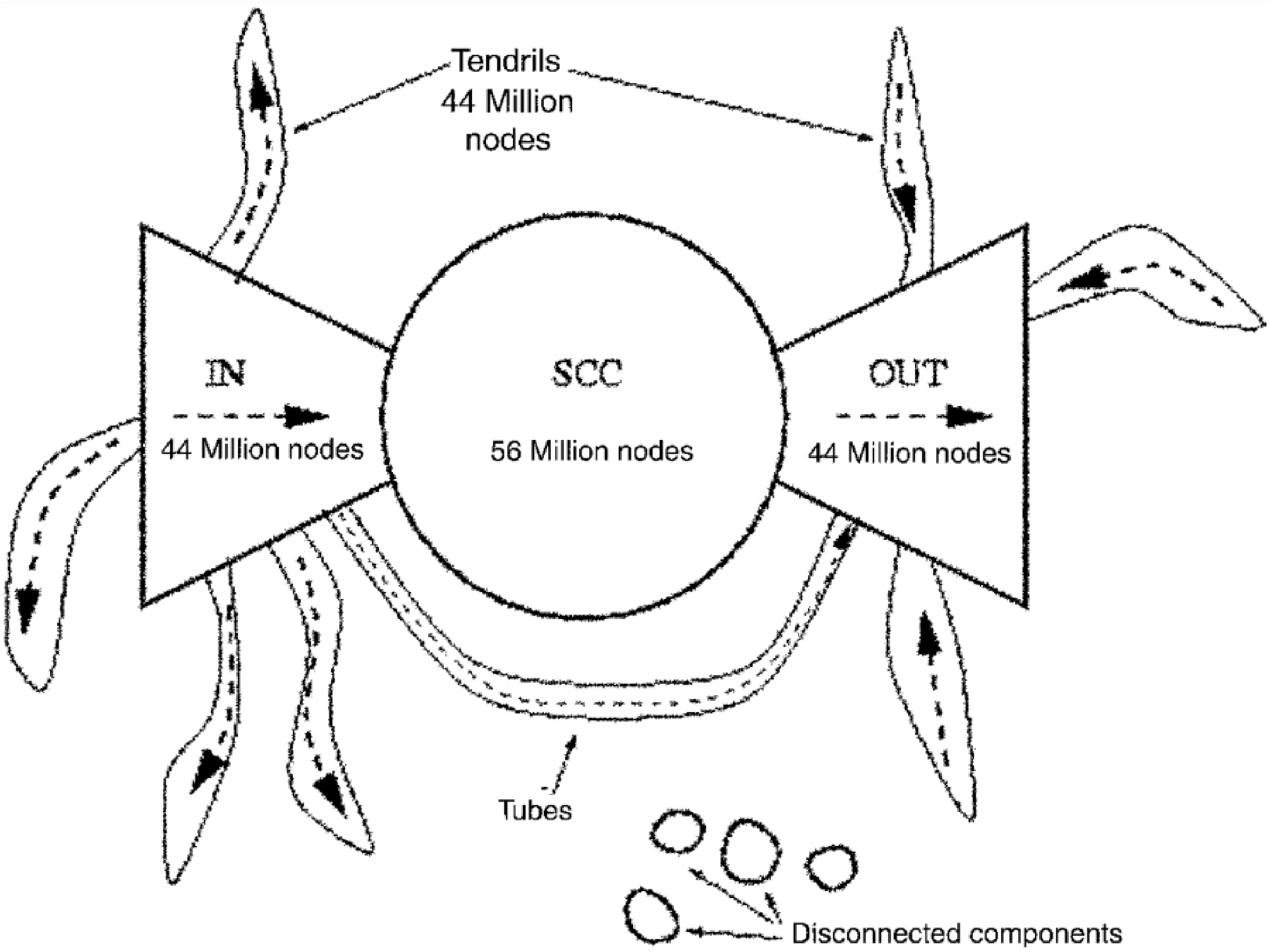
Structure of the Web

- **Broder et al., 2000:**
  - Altavista crawl from October 1999
    - 203 million URLS
    - 1.5 billion links
  - Computer: Server with 12GB of memory
- **Undirected version of the Web graph:**
  - 91% nodes in the largest weakly conn. component
  - *Are hubs making the web graph connected?*
    - Even if they deleted links to pages with in-degree > 10 WCC was still ≈50% of the graph

Structure of the Web

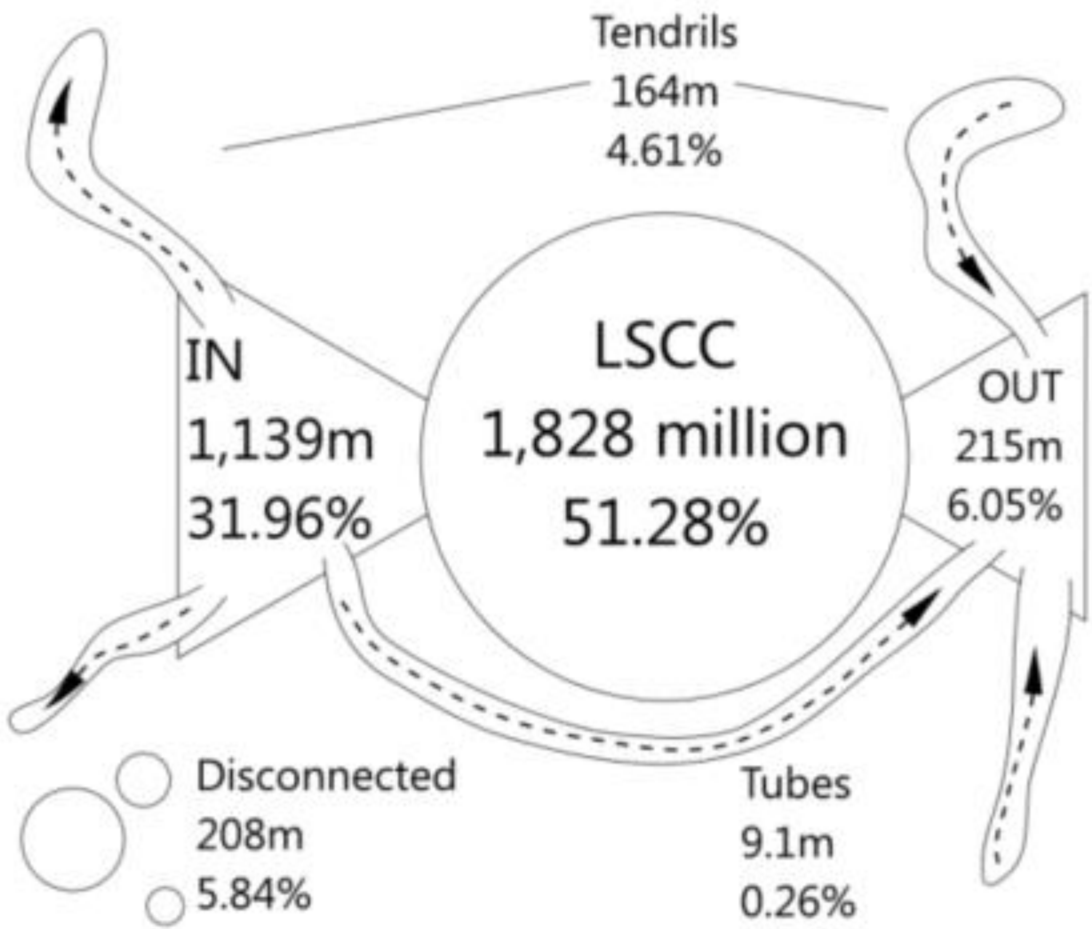
- **Directed version of the Web graph:**
  - **Largest SCC:** 28% of the nodes (56 million)
  - Taking a random node ***v***
    - ***Out(v)*** ≈ 50% (100 million)
    - ***In(v)*** ≈ 50% (100 million)
- ***What does this tell us about the conceptual picture of the Web graph?***

Bow-tie Structure of the Web



203 million pages, 1.5 billion links [Broder et al. 2000]

Bow-tie Structure of the Web - 2012



3.5 billion pages, 128.7 billion links **[Meusel et al. 2014]**

What did We Learn/Not Learn ?

- **What did we learn:**
  - Some conceptual organization of the Web (i.e., the bowtie)
- **What did we not learn:**
  - **Treats all pages as equal**
    - Google's homepage ≈≈ my homepage
  - **What are the most important pages**
    - How many pages have k ***in-links*** as a function of k?
      - the ***degree distribution***: ~ k^{-2}
    - ***Link analysis ranking*** -- as done by search engines (PageRank)
  - **Internal structure inside giant SCC**
    - Clusters, implicit communities?
  - **How far apart are nodes in the giant SCC**
    - Distance = # of edges in shortest path
    - Avg Distance=**16.12** **[Broder et al.]**, Avg Distance=**12.84** **[Meusel et al.]**