

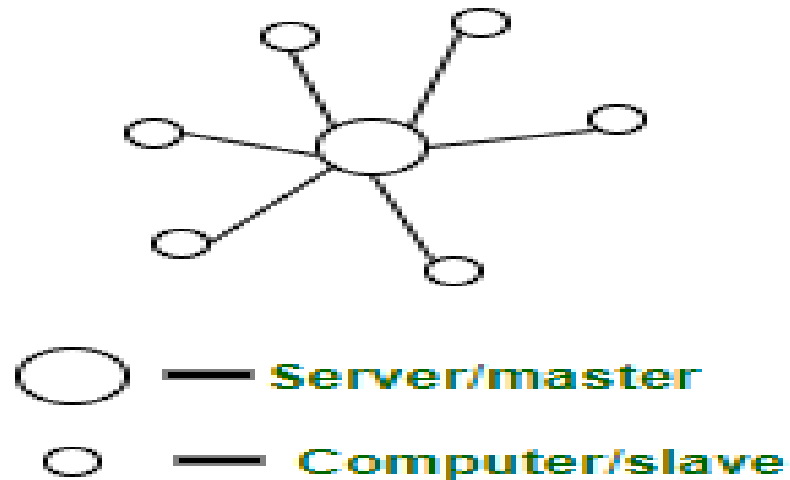
Centralized, Decentralized and Distributed Systems

Prepared by: mahamad karim

Group: A

- **CENTRALIZED SYSTEMS:**

Centralized systems are systems that use client/server architecture where one or more client nodes are directly connected to a central server. This is the most commonly used type of system in many organisations where client sends a request to a company server and receives the response.



Characteristics of Centralized System –

- 1. Presence of a global clock:** As the entire system consists of a central node(a server/ a master) and many client nodes(a computer/ a slave), all client nodes sync up with the global clock(the clock of the central node).
- 2. One single central unit:** One single central unit which serves/coordinates all the other nodes in the system.
- 3. Dependent failure of components:** Central node failure causes entire system to fail. This makes sense because when the server is down, no other entity is there to send/receive response/requests.

Advantages of Centralized System –

- Easy to physically secure. It is easy to secure and service the server and client nodes by virtue of their location
- Smooth and elegant personal experience – A client has a dedicated system which he uses (for example, a personal computer) and the company has a similar system which can be modified to suit custom needs
- Dedicated resources (memory, CPU cores, etc)
- More cost efficient for small systems upto a certain limit – As the central systems take less funds to set up, they have an edge when small systems have to be built
- Quick updates are possible – Only one machine to update.
- Easy detachment of a node from the system. Just remove the connection of the client node from the server and voila! Node detached.

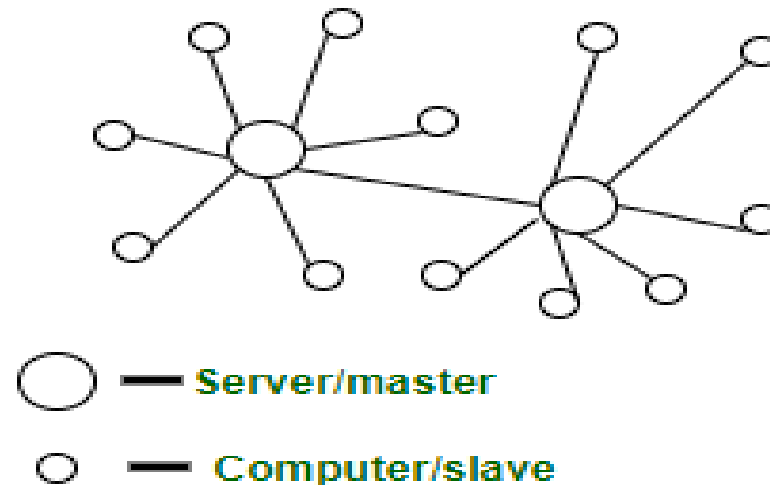
Disadvantages of Centralized System –

- Highly dependent on the network connectivity – System can fail if the nodes lose connectivity as there is only one central node.
- No graceful degradation of system – abrupt failure of the entire system
- Less possibility of data backup. If the server node fails and there is no backup, you lose the data straight away
- Difficult server maintenance – There is only one server node and due to availability reasons, it is inefficient and unprofessional to take the server down for maintenance. So, updates have to be done on-the-fly(hot updates) which is difficult and the system could break.

- **DECENTRALIZED SYSTEMS:**

These are another type of systems which have been gaining a lot of popularity, primarily because of the massive hype of Bitcoin. Now many organisations are trying to find the application of such systems.

In decentralized systems, every node makes its own decision. The final behavior of the system is the aggregate of the decisions of the individual nodes. Note that there is no single entity that receives and responds to the request.



Characteristics of Decentralized System –

1. **Lack of a global clock:** Every node is independent of each other and hence, have different clocks that they run and follow.
2. **Multiple central units (Computers/Nodes/Servers):** More than one central unit which can listen for connections from other nodes
3. **Dependent failure of components:** one central node failure causes a part of system to fail; not the whole system

Advantages of Decentralized System –

- Minimal problem of performance bottlenecks occurring – The entire load gets balanced on all the nodes; leading to minimal to no bottleneck situations
- High availability – Some nodes(computers, mobiles, servers) are always available/online for work, leading to high availability
- More autonomy and control over resources – As each node controls its own behavior, it has better autonomy leading to more control over resources

Disadvantages of Decentralized System –

- Difficult to achieve global big tasks – No chain of command to command others to perform certain tasks
- No regulatory oversight
- Difficult to know which node failed – Each node must be pinged for availability checking and partitioning of work has to be done to actually find out which node failed by checking the expected output with what the node generated
- Difficult to know which node responded – When a request is served by a decentralised system, the request is actually served by one of the nodes in the system but it is actually difficult to find out which node indeed served the request.