International Journal of Academic Engineering Research (IJAER)
ISSN: 2643-9085
Vol. 8 Issue 6 June - 2024, Pages: 1-5

Comparison – Centralized, Decentralized and Distributed Systems

Phd Shamiljon Rustamov1 and Norboev Bekhzod2

1Dean of the Faculty of Computer Engineering,
Karshi branch of the Tashkent University of Information Technologies
shamiljon1988@gmail.com
2Master's 1st stage student,
Karshi branch of the Tashkent University of Information Technologies
norboyevbexzod98@gmail.com

Abstract*: A distributed system is a system of multiple nodes that are physically separated but linked together using the network. Each of these nodes includes a small amount of the distributed operating system software. Every node in this system communicates and shares resources with each other and handles processes in a team. Distributed systems are a growing trend as more and more applications migrate to the cloud. They allow improved performance, scalability, and fault tolerance compared to a traditional centralized system. This Distributed System tutorial will cover all the basic to advanced topics of distributed systems like communication, remote procedure calls, Distributed File Systems, Distributed shared memory, etc. to provide you with a thorough overview of distributed systems.*

Keywords: DFS,Server,Bitcoin,Computer,Blockchain,coins,Ethereum

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 organizations where a client sends a request to a company server and receives the response.

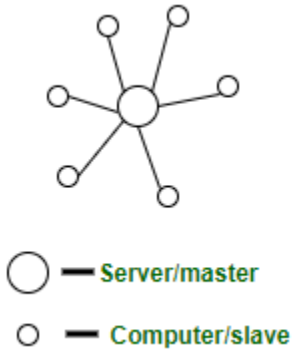

Figure – Centralized system visualization

Characteristics of Centralized Systems

- Run on a single computer system and do not interact with another computer system.
- A modern, general-purpose computer system consists of one to a few processors and a number of device controllers that are connected through a common bus that provides access to shared memory.
- The processors have local cache memories that store local copies of parts of the memory, to speed up access to data.
- Centralized systems have specialized device controllers responsible for managing various hardware components, such as disk drives and audio devices
- A typical single-user system is a desktop unit used by a single person, usually with only one processor and one or two hard disks, and usually only one person using the machine at a time.
- A typical multiuser system, has more disks and more memory and may have multiple processors. It serves a large number of users who are connected to the system remotely.

- Database systems designed for use by single users usually do not provide many of the facilities that a multiuser database provides.
- Centralized systems can have multiple processors, and fine-grained parallelism is a characteristic that may be present in some centralized systems to improve performance

Characteristics of Centralized System –

- 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).
- One single central unit: One single central unit which serves/coordinates all the other nodes in the system.
- Dependent failure of components**:** Central node failure causes the entire system to fail. This makes sense because when the server is down, no other entity is there to send/receive responses/requests.

Scaling –
Only vertical scaling on the central server is possible. Horizontal scaling will contradict the single central unit characteristic of this system of a single central entity.

Components of Centralized System –
Components of Centralized System are,

- Node (Computer, Mobile, etc.).
- Server
- Communication link (Cables, Wi-Fi, etc.).

Architecture of Centralized System –
Client-Server architecture. The central node that serves the other nodes in the system is the server node and all the other nodes are the client nodes.

Limitations of Centralized System –

- Can’t scale up vertically after a certain limit – After a limit, even if you increase the hardware and software capabilities of the server node, the performance will not increase appreciably leading to a cost/benefit ratio < 1.
- Bottlenecks can appear when the traffic spikes – as the server can only have a finite number of open ports to which can listen to connections from client nodes. So, when high traffic occurs like a shopping sale, the server can essentially suffer a Denial-of-Service attack or Distributed Denial-of-Service attack.

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 up to a certain limit – As the central systems take fewer 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.

- Centralized control: In a centralized system, the central authority has complete control over the system, which can lead to better coordination and decision-making.
- Easier to manage: As there is only one central node to manage, it is easier to maintain and manage the system.
- Lower latency: Centralized systems can provide lower latency compared to distributed systems as there is no delay in communication between different nodes.
- Higher performance: Centralized systems can achieve higher performance as the resources can be optimized for specific tasks.
- Simpler implementation: Centralized systems are easier to implement as they require less complex algorithms and protocols.

Disadvantages of Centralized System –

- Highly dependent on the network connectivity – The system can fail if the nodes lose connectivity as there is only one central node.
- No graceful degradation of the 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.
- Single point of failure: Centralized systems have a single point of failure, which can cause the entire system to fail if the central node goes down.
- Lack of transparency: Centralized systems lack transparency as the central authority has complete control over the system, which can lead to issues like censorship and bias.
- Security risks: Centralized systems are more vulnerable to security risks as the central authority has complete access to all the data.
- Limited scalability: Centralized systems have limited scalability as the central node can only handle a limited number of clients at a time.
- Limited innovation: Centralized systems can stifle innovation as the central authority has complete control over the system, which can limit the scope for experimentation and creativity.

Applications of Centralized System –

- Application development – Very easy to set up a central server and send client requests. Modern technology these days do come with default test servers which can be launched with a couple of commands. For example, Express server, Django server.
- Data analysis – Easy to do data analysis when all the data is in one place and available for analysis
- Personal computing

Use Cases –

- Centralized databases – all the data in one server for use.
- Single-player games like Need For Speed, GTA Vice City – an entire game in one system(commonly, a Personal Computer)
- Application development by deploying test servers leading to easy debugging, easy deployment, easy simulation
- Personal Computers

Organizations Using –
National Informatics Center (India), IBM

These are other types of systems that have been gaining a lot of popularity, primarily because of the massive hype of Bitcoin. Now many organizations 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.

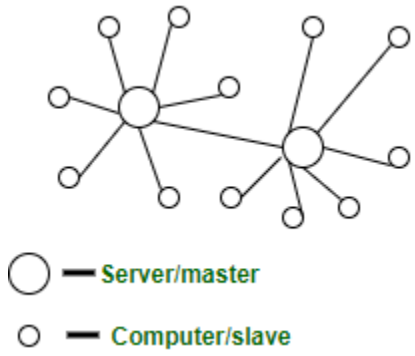

Figure – Decentralized system visualization

Example –
Bitcoin. Let's take Bitcoin for example because it is the most popular use case of decentralized systems. No single entity/organization owns the bitcoin network. The network is a sum of all the nodes who talk to each other for maintaining the amount of bitcoin every account holder has.

Characteristics of Decentralized System –

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

Scaling –
Vertical scaling is possible. Each node can add resources(hardware, software) to itself to increase the performance leading to an increase in the performance of the entire system.

Components –
Components of Decentralized System are,

- Node (Computer, Mobile, etc.)
- Communication link (Cables, Wi-Fi, etc.)

Architecture of Decentralized System –

- peer-to-peer architecture – all nodes are peers of each other. No one node has supremacy over other nodes
- master-slave architecture – One node can become a master by voting and help in coordinating of a part of the system but this does not mean the node has supremacy over the other node which it is coordinating

Limitations of Decentralized System –

- May lead to the problem of coordination at the enterprise level – When every node is the owner of its own behavior, its difficult to achieve collective tasks

- Not suitable for small systems – Not beneficial to build and operate small decentralized systems because of the low cost/benefit ratio
- No way to regulate a node on the system – no superior node overseeing the behavior of subordinate nodes

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.
- Improved fault tolerance: Decentralized systems are designed to be fault tolerant, meaning that if one or more nodes fail, the system can still continue to function. This is because the workload is distributed across multiple nodes, rather than relying on a single point of failure.
- Increased transparency: Decentralized systems often have a transparent and open structure, which allows for greater accountability and trust. Each node has access to the same information, making it more difficult to manipulate or corrupt the data.
- Greater security: Decentralized systems can be more secure than centralized systems because there is no single point of failure or vulnerability that can be exploited by attackers. Data is distributed across multiple nodes, making it more difficult to hack or compromise.

Conclusion:

Each system architecture—centralized, decentralized, and distributed—has its own set of advantages and trade-offs. The choice of architecture depends on the specific needs and constraints of the application or organization:

- Centralized systems offer simplicity and ease of management but come with risks of single points of failure and scalability issues.
- Decentralized systems provide better fault tolerance and security but require complex coordination mechanisms.
- Distributed systems balance load and offer scalability and resilience, but their complexity can pose challenges in terms of design and maintenance.

Understanding the differences and use cases for each system type is crucial for designing robust, efficient, and secure applications and infrastructures.

Literature

1. Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). Operating System Concepts. This book provides an in-depth discussion on centralized system architectures, particularly in operating systems.
2.Tanenbaum, A. S., & Van Steen, M. (2017). Distributed Systems: Principles and Paradigms. Although focusing on distributed systems, this text contrasts centralized systems, highlighting the ease of management.
3.Kurose, J. F., & Ross, K. W. (2021). Computer Networking: A Top-Down Approach**.** This textbook discusses the vulnerabilities and challenges associated with centralized network architectures.