Distributed Systems

Design Models, Network Issues

DS Design and Implementation

Before investing in any distributed system, there exist many design and implementation models that have to be considered to ensure that the system will have the appropriate return on the investment

Increased return on the investment can be achieved by:

- Reduced latency, faster data processing, and optimized resource utilization contribute to overall performance gains.
- accommodate growth without significant cost increases and adapt to changing needs.
- Efficient resource utilization, load balancing, and reduced hardware requirements can result in cost savings.
- Cloud-based distributed systems often offer pay-as-you-go models, reducing upfront capital expenditure.

DS Design and Implementation

- Redundancy and fault-tolerant design in distributed systems contribute to increased reliability and availability.
- A well-designed distributed system can provide a better user experience through faster response times, consistent performance, and high availability of services.
- A DS that enables quicker decision-making, innovative services, or faster product delivery can provide a competitive edge, attracting more customers or clients.
- Implementing robust security measures and meeting compliance standards can prevent costly data breaches, ensuring trust and protecting the system from legal issues.
- A well-designed distributed system allows for the integration of new technologies, making it adaptable to changing business needs and fostering innovation.

DS Design and Implementation

Physical Model:

- The actual hardware, network infrastructure, and the way physical components are organized and connected in the system.
- It includes servers, storage devices, network connections, data centers, and their physical layout.
- deals with tangible, physical components and their arrangement to support the distributed system's functioning.
- Helps in Requirement analysis and best technology selection in the physical environment in which the distributed system operates.

DS Design and Implementation

Architectural model:

- Deals with the high-level structure and design of the system, including the components, their interactions, and the overall system organization.
- It focuses on the system's Processes, **<u>services,</u>** components, and their relationships.
- is concerned with the conceptual layout, design patterns, communication protocols, and software architecture used to create the distributed system.

DS Design and Implementation

Security and Disaster Recovery Models

- Approach to protect against unauthorized access and save sensitive information.
- It employs robust authentication, access controls, encryption techniques, and continuous monitoring to prevent security breaches.
- Adherence to compliance standards for maintaining data confidentiality, integrity and Availability.
- Disaster recovery strategies in distributed systems focus on preparedness and resilience against system failures or catastrophic events.

DS Design and Implementation

Amazon AWS Design Models

Physical Model in AWS:

- Data Centers: AWS operates numerous data centers worldwide.
- The physical infrastructure includes processing servers, storage systems, networking equipment, and cooling systems within these data centers. The layout, geographical locations, and physical security measures are part of AWS' physical model.
- Global Edge Locations: AWS has a network of edge locations strategically placed worldwide for content delivery and reduced latency. These edge locations house caching and content delivery systems to improve service performance.

DS Design and Implementation

Architectural Model in AWS:

- Service Offerings and Architecture: AWS provides a wide array of cloud services, each with its architectural model.
  - Amazon S3 (Simple Storage Service): It follows an object storage architectural model for scalable and durable storage.
  - AWS Lambda: It operates on a serverless architecture, enabling event-driven computing without provisioning or managing servers.
  - Amazon EC2 (Elastic Compute Cloud): EC2 services adhere to a virtualized server architecture allowing users to run virtual machines in the cloud.

DS Architectural Patterns:

Layered Architecture Model:

- Divides the system into distinct layers, each responsible for specific functionalities.
- Data passes through these layers, with each layer providing services to the layer above and using services from the layer below.
- Modularity, abstraction, and ease of maintenance.

DS Architectural Patterns:

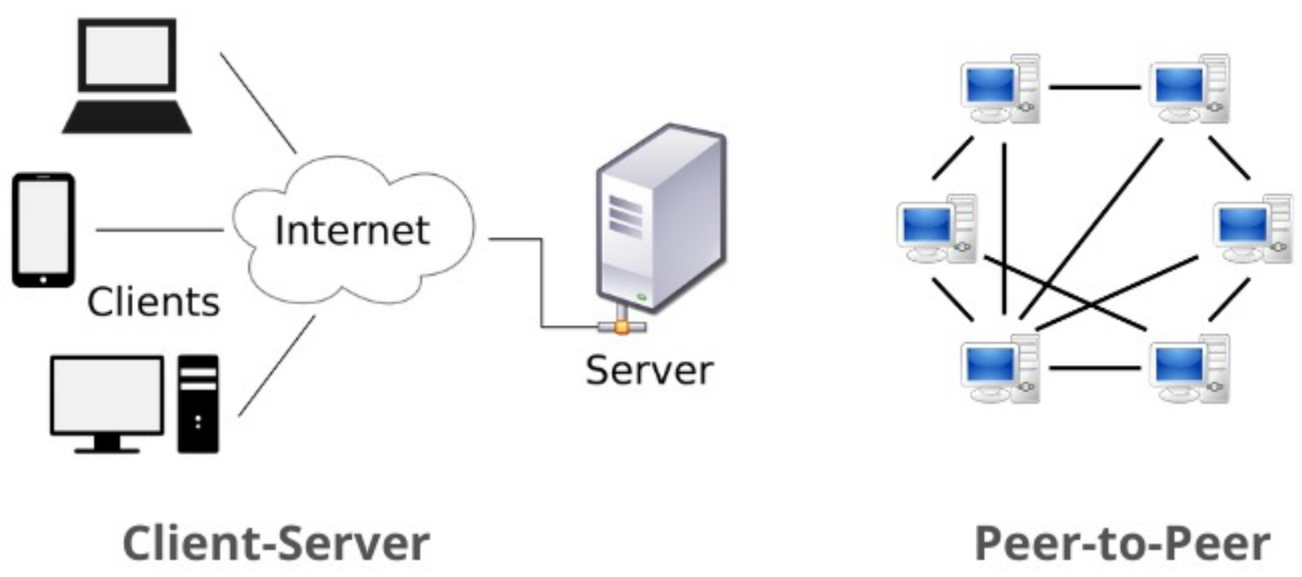
Client-Server Model:

- In this model, clients make requests to servers, which provide resources or services. It's a fundamental model in which the server hosts and delivers services, while clients consume these services.
- It Offers scalability, flexibility, and centralization of resources and services.

Peer-to-Peer (P2P) Model:

- P2P systems enable nodes (peers) to act both as clients and servers. Peers can directly communicate and share resources without the need for a centralized server.
- Decentralized structure, self-organization, and efficient resource sharing among peers.

DS Architectural Patterns:



https://www.networkstraining.com/peer-to-peer-vs-client-server-network/

Distributed Systems Networks

- Networking is a fundamental aspect of distributed systems as a network key important resource.
- It encompassing several crucial resources that ensure communication, connectivity, and interaction between various components in a distributed environment.
- Design of Networking hardware and protocols is a key important issue in distributed systems.
- Networking performance metrics is crucial for improving distributed systems services

Distributed Systems Networks

TCP/IP

- TCP/IP, which stands for Transmission Control Protocol/Internet Protocol, is a suite of communication protocols that form the foundation of the internet.
- It provides the rules and conventions for devices to communicate over a network. TCP/IP is a stack of protocols, with each layer handling specific aspects of the communication process.
- The TCP/IP protocol suite consists of several layers, each serving a specific purpose:

Distributed Systems Networks

TCP/IP

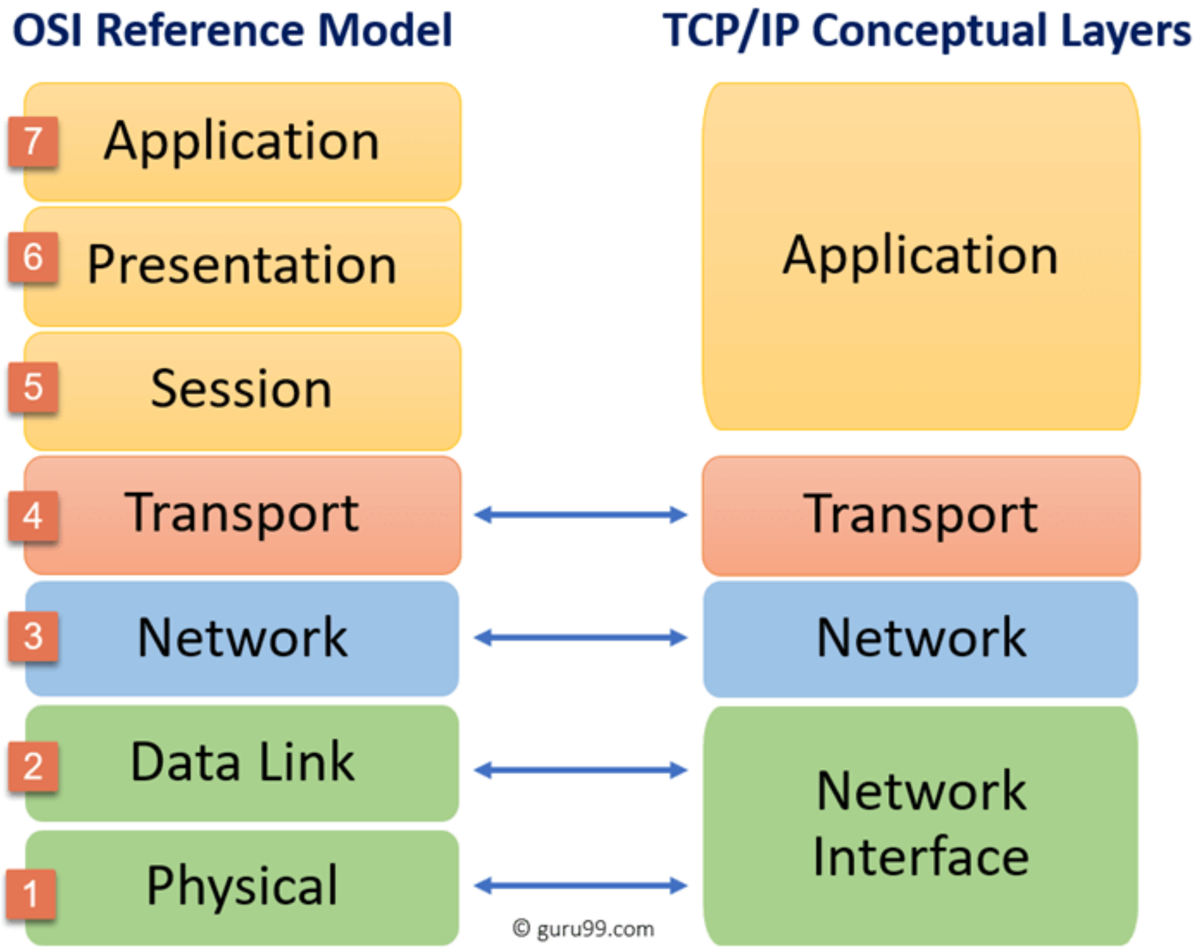
- Link Layer (or Network Interface Layer): This layer deals with the physical connection between devices on the same network. It includes protocols that define how data is framed for transmission and how devices on the network are addressed using MAC (Media Access Control) addresses.
- Internet Layer: The Internet Protocol (IP) operates at this layer and is responsible for the addressing and routing of packets between devices across different networks. IP assigns a unique IP address to each device on a network, allowing them to be identified and located.

Distributed Systems Networks

TCP/IP

- Transport Layer: This layer is responsible for end-to-end communication and data flow control between devices. The most common protocols at this layer are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). TCP provides reliable, connection-oriented communication, while UDP offers a connectionless, lightweight option.
- Application Layer: This is the topmost layer and is where application-specific protocols operate. Protocols like Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), and Simple Mail Transfer Protocol (SMTP) operate at this layer, enabling various applications to communicate over the network.

Distributed Systems Networks



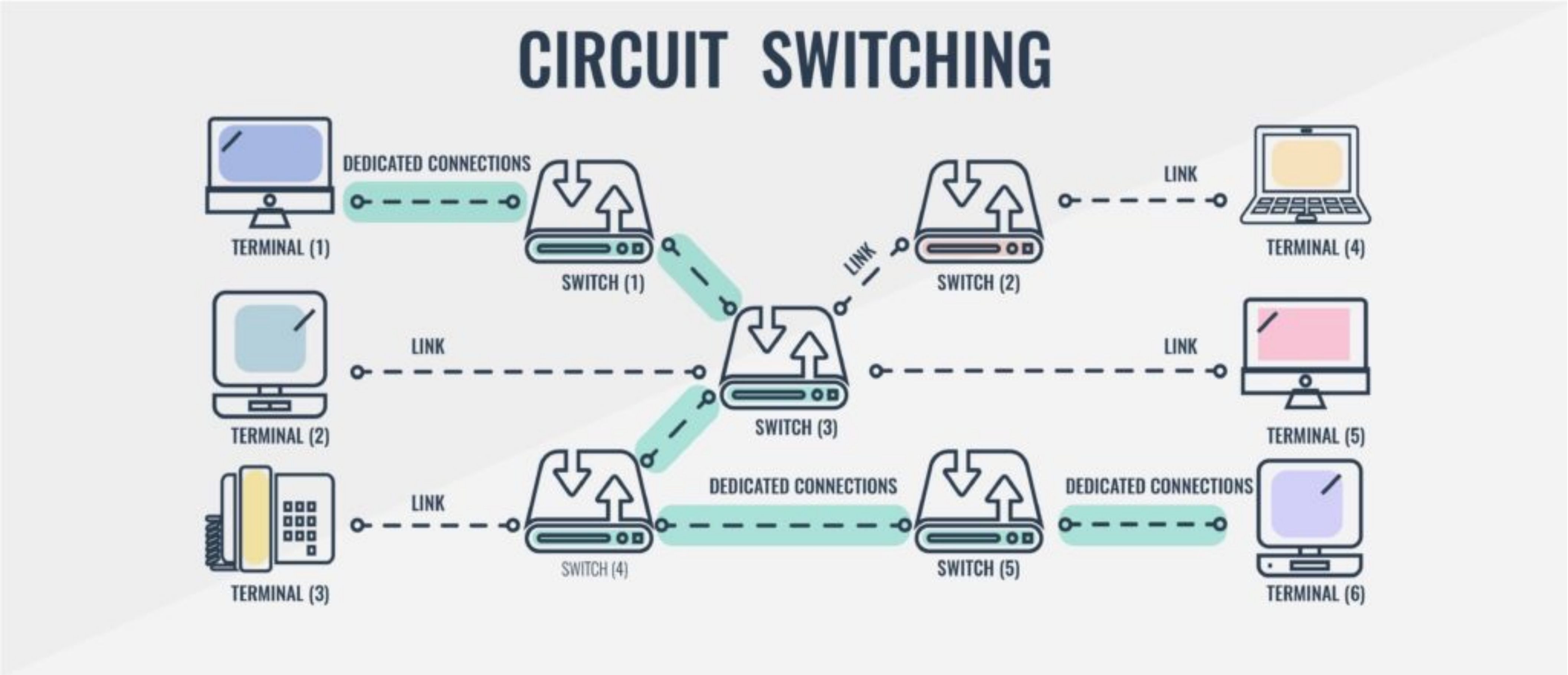
https://www.guru99.com/tcp-ip-model.html

Distributed Systems Networks

Packet Switching

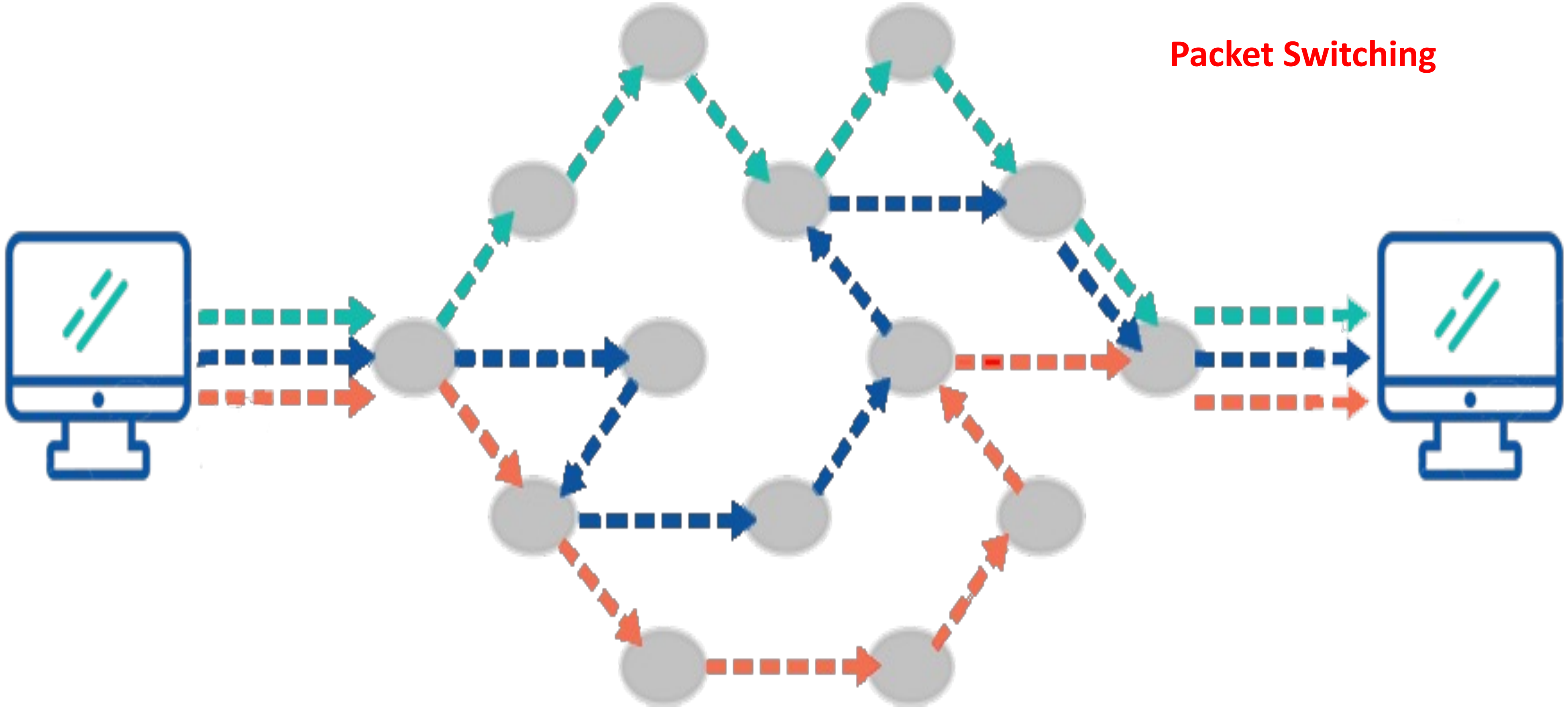
- Data is divided into small, independent packets, each containing part of the original information along with source and destination addresses.
- These packets are then transmitted individually across the network, taking different routes to reach their destination. Once there, the packets are reassembled to reconstruct the original data.
- Offers advantages such as efficient use of network resources, resilience to failures, scalability, and flexibility, making it the foundation of modern communication networks, notably exemplified by the internet

Distributed Systems Networks



https://www.bzfar.org/publ/computer_networks/types_of_networks/packet_switching_vs_circuit_switching/29-1-0-129

Distributed Systems Networks



Packet Switching

https://www.apposite-tech.com/packet-switching-vs-circuit-switching/

Distributed Systems Networks

Routing, and load balancing

- Load balancing in routing is a technique employed in computer networks to optimize the distribution of data traffic across multiple paths or network resources.
- It aims to prevent congestion and uneven utilization by dynamically distributing the workload among available routes, thereby improving network performance, response time, and resource utilization.
- Load balancing can be achieved through various algorithms that consider factors like link utilization, network latency, or available bandwidth.
- By efficiently spreading the network load, load balancing enhances overall reliability and ensures that no single path is overburdened, contributing to a more robust and responsive communication infrastructure.

Distributed Systems Networks

MAC / IP address/ Port Number

- MAC addresses are used for local network communication and are hardware-specific, (e.g., 00:1A:2B:3C:4D:5E). It used when communicating inside the same network
- IP addresses are essential for global network communication and represent a device's network location. It is used in end-to-end communication.
- Port numbers facilitate communication between specific processes or services on a device, allowing multiple services to coexist on the same device.

Distributed Systems Networks

TCP (Transmission Control Protocol):

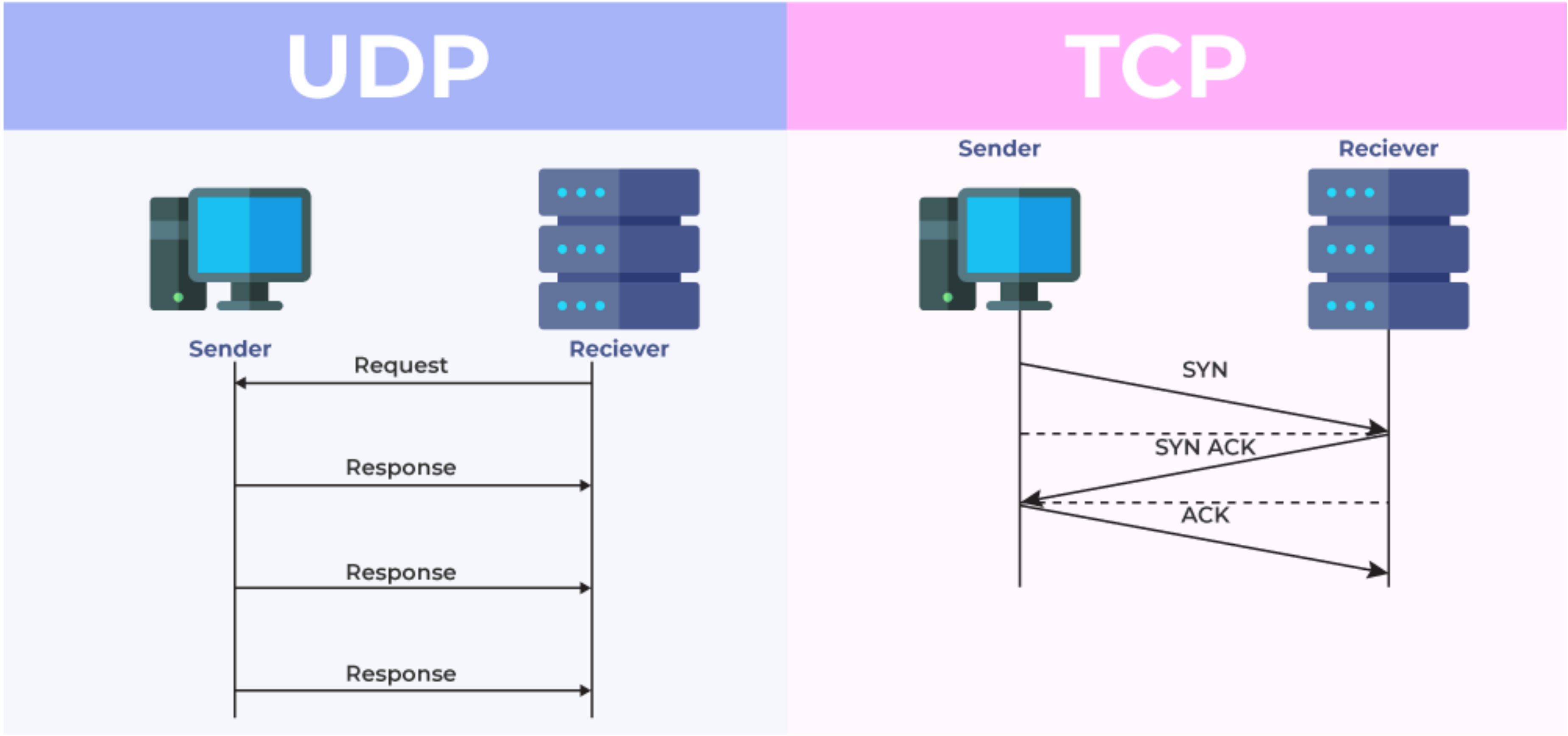
- TCP is a connection-oriented protocol. Before any data transfer occurs, a reliable connection is established between the sender and the receiver. This ensures that data is delivered in the correct order and without errors.
- TCP guarantees the delivery of data. It uses mechanisms such as acknowledgment, retransmission, and flow control to ensure that data reaches its destination intact and in the correct order.
- Due to the connection setup and the reliability mechanisms, TCP is generally considered slower compared to UDP.
- TCP is used for applications where data integrity and order are crucial, such as file transfers, email, and web browsing.

Distributed Systems Networks

UDP (User Datagram Protocol):

- UDP is a connectionless protocol. It does not establish a connection before sending data and does not guarantee that the data will reach its destination.
- Unlike TCP, UDP does not ensure the delivery of data. It is a best-effort protocol, which means it may not be suitable for applications that require data reliability.
- UDP is faster than TCP due to its connectionless nature and lack of reliability mechanisms. It is often used in real-time applications where low latency is more critical than guaranteed delivery, such as streaming and online gaming.
- UDP is used for applications where low latency and real-time communication are more important, such as voice over IP (VoIP), video conferencing, and online gaming.

Distributed Systems Networks



https://www.geeksforgeeks.org/differences-between-tcp-and-udp/

Distributed Systems Networks

Networks Heterogeneity

- The presence of diverse hardware and software components within the network environment.
- Device and Topology Heterogeneity , Operating System Heterogeneity are the most important issues
- Internet Protocol solves the issue of Heterogeneity in computer networks

Distributed Systems Networks

Network performance

- Refers to the efficiency and speed with which data is transmitted and received across a network.
- Bandwidth: The capacity of the network to transmit data.
- Latency: The delay between the initiation and completion of a data transfer.
- Jitter: Variability in latency, affecting the consistency of data delivery.
- Packet Loss: Occurs when data packets do not reach their destination. Retransmitted in TCP and usually not Retransmitted in UDP

Distributed Systems Networks

Improving Network Performance:

- Optimizing network design.
- Upgrading hardware and infrastructure.
- Implementing Quality of Service (QoS) policies which gives priorities for different communicating processes in the available using network bandwidth.
- Utilizing content delivery networks (CDNs) for distributed content (Edge Servers).

Distributed Systems Networks

Reliability in a network

- The ability of the network to consistently and accurately deliver data without failures or errors.
- Use of reliable communication protocols like TCP.
- Regular monitoring and maintenance to identify and address potential issues.
- Regularly testing and updating network equipment and software.
- Utilizing error-checking mechanisms to detect and correct transmission errors.
- Employing resilient routing protocols.

Distributed Systems Networks

Fault Tolerance In Networks

- Network's ability to continue operating even in the presence of failures or faults.
- Duplicating critical components to ensure that if one fails, another takes over.
- Distributing network traffic across multiple paths or devices to prevent overload or failure in a single component.
- Automatically redirecting traffic to backup systems in case of a failure.

References

- https://www.networkstraining.com/peer-to-peer-vs-client-server-network/
- https://www.geeksforgeeks.org/differences-between-tcp-and-udp/
- https://www.imperva.com/learn/availability/fault-tolerance/
- https://www.bzfar.org/publ/computer_networks/types_of_networks/packet_switching_vs_circuit_switching/29-1-0-129
- https://www.apposite-tech.com/packet-switching-vs-circuit-switching/