Chapter 1: Advanced Networking Protocols

Lesson 1: Overview of advanced networking protocols

Network protocols are a set of rules outlining how connected devices communicate across a network to exchange information easily and safely. Protocols serve as a common language for devices to enable communication irrespective of differences in software, hardware, or internal processes.

Basic Recall
Types of network protocols
Different protocols serve different functions to ensure efficient, quick, and secure network communication. Various types of network protocols can be categorized into the following three broad categories to help organizations operate seamlessly across different business scenarios:

1. Network Communication Protocols
These protocols determine the rules and formats to transfer data across networks. Communication protocols govern various aspects of analog and digital communications, such as syntax, authentication, semantics, and error detection, among others. Some key network communication protocols include:

- **Hyper-Text Transfer Protocol (HTTP)**: Commonly referred to as the protocol of the internet that allows communication between a server and browser.
- **Transmission Control Protocol (TCP)**: A reliable, connection-oriented protocol that helps in the sequential transmission of data packets to ensure data reaches the destination on time without duplication.
- **Internet Protocol (IP)**: Facilitates routing the data packets across networks. IP contains addressing and control information to deliver packets across a network. It works along with TCP. While it ensures delivering the packets to the right address, TCP aligns them in the right order.
- **User Datagram Protocol (UDP)**: Unlike TCP, UDP is a connectionless protocol that doesn't ensure a connection between the application and server before transmitting a message. It's effective for use cases such as broadcasts or multicast connections.
- **File Transfer Protocol (FTP)**: Allows file sharing between servers by establishing two TCP connections, one for data transfer and the other for control. The data transfer connection transfers the actual files while the control connection transfers control information such as passwords to ensure data retrieval in case of data loss.

Helps diagnose network connectivity issues. Network devices employ ICMP for sending error messages, highlighting congestion and timeouts, and transmitting other operational information to assist in network troubleshooting.

2. Network Security Protocols
These protocols ensure safe data transmission over the network connections. Network security protocols define the procedures to secure data from any unauthorized access. These protocols leverage encryption and cryptography to safeguard. Here are the most widely used network security protocols:

- **Secure File Transfer Protocol (SFTP):** Helps securely transfer files across a network by using public-key encryption and authenticating the client and server.
- **Hyper-Text Transfer Protocol Secure (HTTPS):** Overcomes the limitation of HTTP by ensuring the security of data transmitted between the browser and server through data encryption. HTTPS is a secure version of HTTP.
- **Secure Socket Layer (SSL):** Primarily helps secure internet connections and safeguard sensitive data using encryption. SSL protocol enables both server-client communication and server-server communication.

3. Network Management Protocols
Network managers require standard policies and procedures to manage and monitor the network for maintaining smooth communication. Network management protocols ensure quick troubleshooting and optimal performance across the network. The following are essential network protocols management:

- **Simple Network Management Protocol (SNMP):** Helps administrators manage network devices by monitoring endpoint information to proactively track network performance and pinpoint network glitches for quick troubleshooting.
- **Internet Control Message Protocol (ICMP):** Helps diagnose network connectivity issues. Network devices employ ICMP for sending error messages, highlighting congestion and timeouts, and transmitting other operational information to assist in network troubleshooting.

How do network protocols function in each OSI model layer?
Single or multiple protocols operate at each layer of the OSI model to enable communication. Here's a quick snapshot of how network protocols function in each OSI model layer.

- **Layer 1: Physical Layer Protocols:** These protocols provide an interface between devices and network medium.
- **Layer 2: Data Link Layer Protocols:** The protocols operating at this level ensure the framing of packets while proactively identifying and rectifying packet transmission errors.
- **Layer 3: Network Layer Protocols:** By leveraging the right set of network layer protocols, administrators route packets efficiently while managing the network flow and congestion to prevent network resource depletion.
- **Layer 4: Transport Layer Protocols:** These protocols ensure reliable end-to-end packet delivery across networks in the right sequence at the receiving end.
- **Layer 5: Session Layer Protocols:** Protocols at the session layer help manage dialogues and user sessions by seamlessly establishing and terminating sessions for communication exchange.
- **Layer 6: Presentation Layer Protocols:** These protocols are necessary to encode and decode data to smoothly mask the variations in data formats across different systems.
- **Layer 7: Application Layer Protocols:** These protocols help transform user requests to network-friendly formats.

Lesson 2: Internet Protocol version 6 (IPv6)

What is IPv6?
IPv6 represents the newer generation of technology and development. The older version of IP – IPv4 is still very popular, but its shortage is a common issue. So, at some time in the future, we should let it go. Therefore, it is essential to understand the newer version of IP – IPv6.

IPv6 – What does it mean, and what is it used for?
IPv6 stands for Internet Protocol version 6, and it is the newer version of the Internet Protocol (IP). Yet, can you imagine it was around for more than 20 years? It was introduced back in December 1995! The main goal for its creation is to take over and eventually replace the previous protocol – IPv4. The reason is simple. The number of devices that want to connect to the Internet is growing tremendously, and IPv4 is not able to satisfy such needs.

The IPv6 is a network layer protocol that allows communication and data transfer between two different hosts. It sets specific rules that help identify the separate hosts and track their location. That way, they could exchange information successfully. Only when the two corresponding IP addresses are identified, the route could be established, and the hosts are able to communicate.

IPv4 protocol, the previous standard, allows 4.2 billion unique IP addresses. However, with the newer tech developments and the various new wireless and network-attached devices, such as the IoT devices, it was predicted that by 2010, the Internet would have exhausted all unique IPv4 addresses.

On the other hand, thanks to the standardization of the new IPv6, it allows 3.4 x 1038 unique IP addresses. This is equal to 340 trillion IP addresses.

IPv6 operates with 128-bit addresses. Each address includes eight different groups of strings, and every group has four characters (alphanumeric), divided by a colon. Thanks to these characteristics, it is able to provide an incredible amount of unique IP addresses. That guarantees that we should have available unique IP addresses to assign to all of the new devices for a very long time.

How does the Internet work?
The Internet is a pretty extensive cable network. It connects numerous data centers placed all over the world and the users that desire to reach and connect with their services. All of the network points are connected with massive cables.

Additionally, such a large network of interconnected machines and devices requires proper order and the ability to identify all of the different devices with their associated addresses. Therefore, both users and servers should have an IP address for that purpose. Moreover, the servers hold hostnames, too.

When a user wants to view a particular website, it has to type its domain name (hostname) and connect to the web server that holds the information for it. Every website on the Internet is hosted on web servers in different data centers. That way, you can access websites, applications, and services.

IP address – definition
The IP address serves as an ID and identifies all of the various hosts on the network – both servers and users.

There are two main types of IP addresses:

Private: This type of IP address is used when users connect on a closed private network. Thanks to it, the user gains access to the specific network, and it is able to communicate with the other devices, which it includes.

Public: This type of IP address is used when you want to connect to the Internet. Usually, an Internet service provider (ISP) provides you with a router that you need and a public IP address. Servers need such an address too, and it should not change, meaning they should be static.

You are probably wondering why we are talking about IP addresses. In reality, to access a website, we just type domain names.

Domain Name System explained
The Domain Name System (DNS) is a global database that contains all of the existing domain names and their IP addresses. It answers the DNS queries of the users for the domain names and their IP addresses daily.

The Domain Name System is decentralized and built in a hierarchical order. Therefore, each level knows the answer for the one below. On the top level are the Root servers, which provide information about the TLD (Top-Level Domain) servers. In addition, they hold data about where the different extensions are, such as .com, .info, .net, etc.

Thanks to this arrangement, it is easy for users to type the domain name and reach the website. The user requests the needed IP address (IPv4 or IPv6), and it first checks the DNS cache of the device. If it's not available there, the recursive DNS server performs the next step. It searches for the answer until it reaches the authoritative DNS server that holds the needed information (A record or AAAA record). This whole process is also known as DNS resolution.

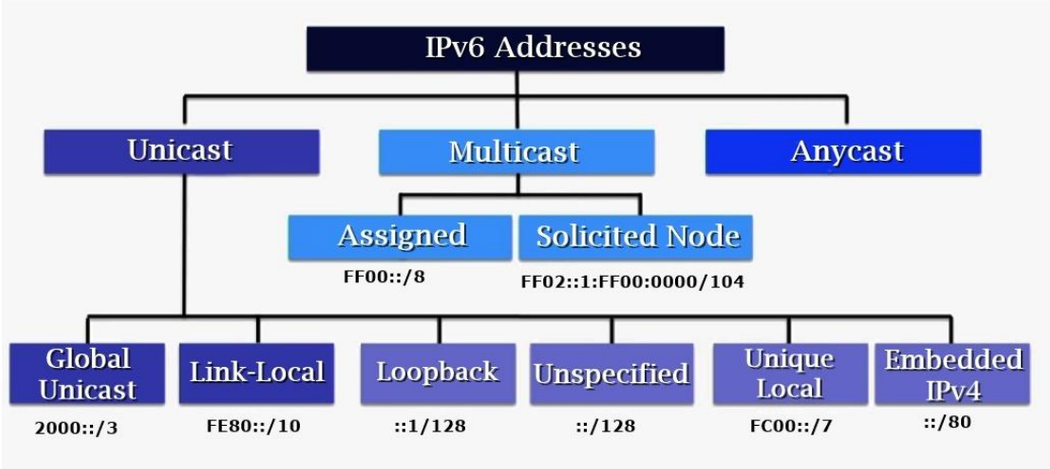
Types of Internet Protocol version 6 addresses
Now we know what an IPv6 address is. So, let's take a look at its three different types: unicast, anycast, and multicast, which are defined by RFC 4291: IP Version 6 Addressing Architecture.

1. **Unicast (a single interface)** – It represents a particular node on a network and frequently alludes to a specific transmitter or receiver. Accordingly, it is one-on-one communication.
2. **Anycast (a set of interfaces)** – It is linked to a group of interfaces, most of which are connected to various nodes. Accordingly, it is one-to-closest communication.
3. **Multicast (a group of interfaces)** – We only implement it as a datagram's destination and represents a collection of IP devices. Accordingly, it is one-to-many communication.

It all depends upon how the source wants to communicate with the destination. e.g.

- If the source wants to communicate with single destination, then it will use the unicast address as a destination.
- If the source wants to communicate with multiple devices at a time, then it will use the multicast address as a destination.
- We can use Anycast addresses as identifiers for a set of interfaces that may belong to the different nodes and IP address will be same on these nodes.

These three main categories are further divided into sub-categories, you can see in the below diagram.



Note: IPv6 'Source' always have a unicast address, and 'Destination' address can be unicast, multicast or anycast.

IPv6 Unicast Addresses
Unicast addresses represent a single interface. Packets addressed to a unicast address will be delivered to a specific network interface.

There are three types of IPv6 unicast addresses:

- **global unicast** – similar to IPv4 public IP addresses. These addresses are assigned by the IANA and used on public networks. They have a prefix of 2000::/3, (all the addresses that begin with binary 001).
- **unique local** – similar to IPv4 private addresses. They are used in private networks and aren't routable on the Internet. These addresses have a prefix of FD00::/8.
- **link local** – these addresses are used for sending packets over the local subnet. Routers do not forward packets with this address to other subnets. IPv6 requires a link-local address to be assigned to every network interface on which the IPv6 protocol is enabled. These addresses have a prefix of FE80::/10.

Global Unicast Address (GUA)
When an IPv6 enabled device wants to access Internet, then it needs an address which must be unique over the Internet. The address used for this purpose is the global unicast address. Its range is 2000::/3 (First hextext: 2000::/3 to 3FFF::3). These addresses are globally unique and routable, which are similar to public IPv4 addresses.

This address is used by:

Hosts to communicate to the IPv6 network before it has a GUA.
Router's link-local address is used by hosts as the default gateway address.
Adjacent routers to exchange routing updates.
Next-hop addresses in IPv6 routing tables.

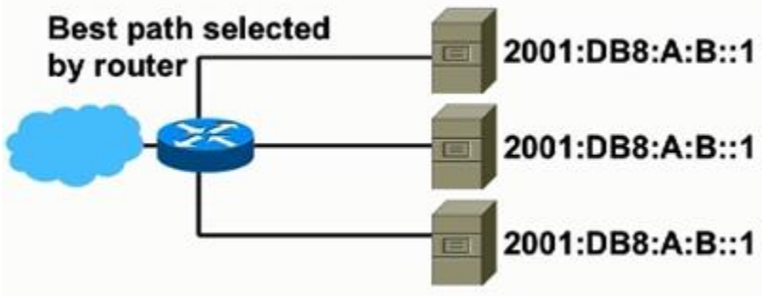
Link-Local Unicast
A link-local address is an IPv6 unicast address that can be automatically configured on any interface using the link-local prefix FE80::/10 (First hextet: FE80::10 to FEBF::/10). Link is a network or subnet, this type of address is not routable off the link and it is unique only on the link. An IPv6 device must have at least a link-local address.

This address is used by:

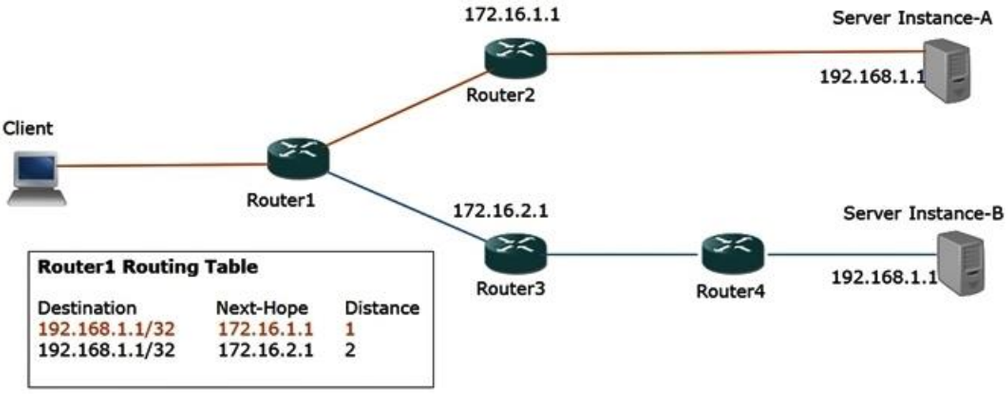
- Hosts to communicate to the IPv6 network before it has a GUA.
- Router's link-local address is used by hosts as the default gateway address.
- Adjacent routers to exchange routing updates.
- Next-hop addresses in IPv6 routing tables.

Anycast Address

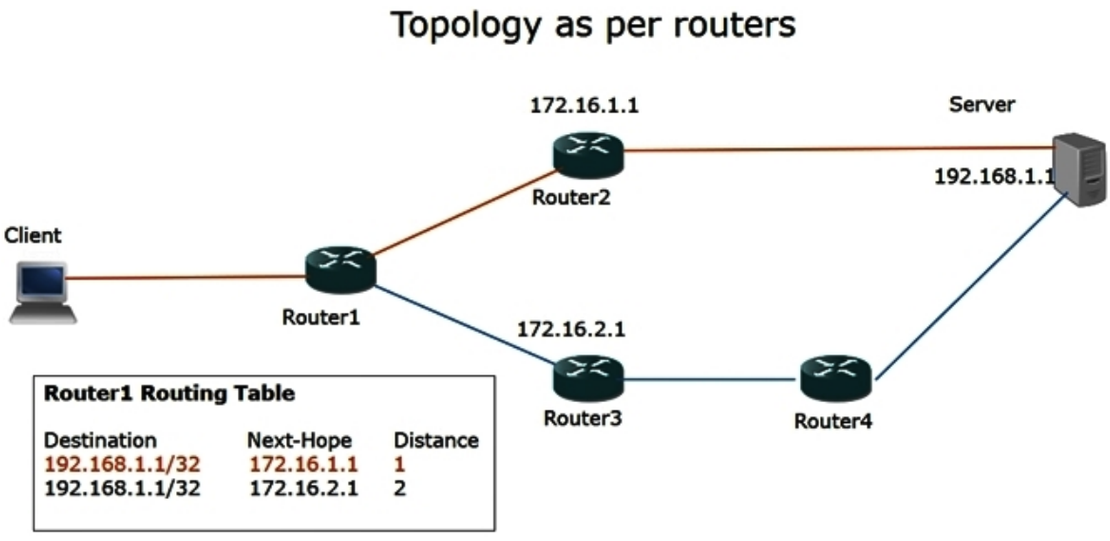
A unicast address that is assigned to more than one interface (typically different devices) and these devices are performing almost the same functionality. The request routed towards the device which is nearest to the user who sent the request. The router takes that decision, based on the best route in the routing table of it.



In the following example of IPv4 anycast address type, there are two servers in different locations, and having the same addresses. When a client sends a request to the server, the Router1 will check its routing table and identify which server is near to it as per the calculation of routing protocol. Server Instance-A is having the lower distance as per the Server Instance-B.



The Router1 will select the route --> Router1-->Router2-->Server Instance-A and will forward the request on that route. From the prospective of the Route1 the topology will be as per the below diagram:



Anycast provides high availability, and distributes the traffic load in different locations.

If a client is connected to the Router3 and it sends a request to the server, for the Router3 Server Instance-B is near as compared to the Server Instance-A. In this way when the traffic will come behind the Router1 and Router2, that will be forwarded to the Server Instance-A. And the traffic that will come from the Router3 and Router4 will be transferred towards the Server Instance-B. In this example, if Sever Instance-A will be down, the traffic coming from any of the router will be a route towards Server Instance-B.

Why Did We Need a New Version of IP?
At this point, you might be wondering why IPv6 even exists.

Well, while the 4.3 billion potential IP addresses in IPv4 might seem like a lot, we need a lot more IP addresses!

There are a lot of people in the world with a lot of devices. This is an even larger issue with the rise of IoT devices (Internet of Things) and sensors, as these greatly expand the pool of connected devices.

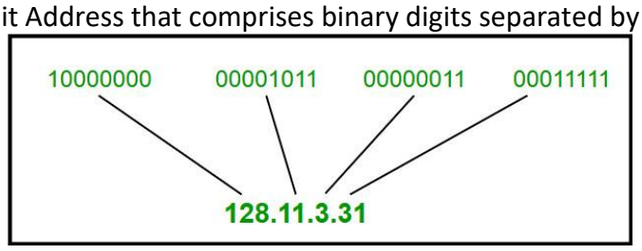
Put simply, the world was running out of unique IPv4 addresses, which is the biggest reason why we needed IPv6.

Differences between IPv4 and IPv6

IPv4
IPv4 address consists of two things that are the network address and the host address. It stands for Internet Protocol version four. It was introduced in 1981 by DARPA and was the first deployed version in 1982 for production on SATNET and on the ARPANET in January 1983.

IPv4 addresses are 32-bit integers that have to be expressed in Decimal Notation. It is represented by 4 numbers separated by dots in the range of 0-255, which have to be converted to 0 and 1, to be understood by Computers. For Example, An IPv4 Address can be written as 189.123.123.90.

IPv4 Address Format
IPv4 Address Format is a 32-bit Address that comprises binary digits separated by a dot (.).



IPv4 Address Format

IPv6
IPv6 is based on IPv4 and stands for Internet Protocol version 6. It was first introduced in December 1995 by Internet Engineering Task Force. IP version 6 is the new version of Internet Protocol, which is way better than IP version 4 in terms of complexity and efficiency. IPv6 is written as a group of 8 hexadecimal numbers separated by colon (:). It can be written as 128 bits of 0s and 1s.

IPv6 Address Format
IPv6 Address Format is a 128-bit IP Address, which is written in a group of 8 hexadecimal numbers separated by colon (:).

Benefits of IPv6
The recent Version of IP IPv6 has a greater advantage over IPv4. Here are some of the mentioned benefits:

- **Larger Address Space:** IPv6 has a greater address space than IPv4, which is required for expanding the IP Connected Devices. IPv6 has 128-bit IP Address rather and IPv4 has a 32-bit Address.
- **Improved Security:** IPv6 has some improved security which is built in with it. IPv6 offers security like Data Authentication, Data Encryption, etc. Here, an Internet Connection is more Secure.
- **Simplified Header Format:** As compared to IPv4, IPv6 has a simpler and more effective header Structure, which is more cost-effective and also increases the speed of Internet Connection.
- **Prioritize:** IPv6 contains stronger and more reliable support for QoS features, which helps in increasing traffic over websites and increases audio and video quality on pages.
- **Improved Support for Mobile Devices:** IPv6 has increased and better support for Mobile Devices. It helps in making quick connections over other Mobile Devices and in a safer way than IPv4.

Sample:
The IPv6 address **2607:f8b0:4004:0c0b:0000:0000:0000:001b** can be broken down into the following steps:

The first 4 bits (2607) represent the global routing prefix. This prefix is assigned by the Internet Assigned Numbers Authority (IANA) and identifies the AS that the address belongs to.

The next 16 bits (f8b0) represent the subnet ID. This ID identifies the specific subnet within the AS that the address belongs to.

The next 32 bits (4004:0c0b) represent the interface ID. This ID uniquely identifies the interface on the router that the address is assigned to.

The remaining 64 bits (0000:0000:0000:001b) are reserved for future use.

Here is a step- the IPv6 address 2607:f8b0:4004:0c0b:0000:0000:0000:001b:

Global routing prefix: The first 4 bits (2607) represent the global routing prefix. This prefix is assigned by the IANA and identifies the AS that the address belongs to. In this case, the global routing prefix is 2607, which is assigned to Google.

Subnet ID: The next 16 bits (f8b0) represent the subnet ID. This ID identifies the specific subnet within the AS that the address belongs to. In this case, the subnet ID is f8b0, which is a private subnet used by Google.

Interface ID: The next 32 bits (4004:0c0b) represent the interface ID. This ID uniquely identifies the interface on the router that the address is assigned to. In this case, the interface ID is 4004:0c0b, which is an interface on a router in Google's data center.

Reserved bits: The remaining 64 bits (0000:0000:0000:001b) are reserved for future use.
IPv6 addresses are written in hexadecimal notation, which means that each group of 4 bits is represented by a hexadecimal digit. The hexadecimal digits 0-9 and A-F are used to represent the values 0-15.

To convert an IPv6 address from hexadecimal notation to decimal notation, you can use the following conversion table:

Hexadecimal digit	Decimal value
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10

B | 11
C | 12
D | 13
E | 14
F | 15

For example, to convert the hexadecimal digit f to decimal, you would look up f in the conversion table and find that it has a value of 15.

To convert the IPv6 address 2607:f8b0:4004:0c0b:0000:0000:0000:001b to decimal notation, you would first convert the global routing prefix, subnet ID, and interface ID to decimal notation using the conversion table above.

Global routing prefix: 2607 = 39891
Subnet ID: f8b0 = 63744
Interface ID: 4004:0c0b = 163844049
Once you have converted the global routing prefix, subnet ID, and interface ID to decimal notation, you can simply concatenate them together to get the IPv6 address in decimal notation.

IPv6 address in decimal notation: 39891:63744:163844049:0:0:0:0:1b

Lesson 3: Border Gateway Protocol (BGP)

What Is BGP?
Border Gateway Protocol (BGP) is the routing method that enables the Internet to function. Without it, it would not be possible to search on Google or send an email.

BGP is a routing protocol. Here is a simple definition of network routing from John F. Shoch, an American computer scientist who developed the predecessor to TCP/IP:

"The name of a resource indicates what we seek, an address indicates where it is, and a route tells us how to get there".

The BGP protocol helps find the best route for network traffic seeking to reach an autonomous system (AS). An AS can be an Internet Service Provider or a large organization that controls a network prefix, representing a range of IP addresses. An autonomous system has a unique number called an ASN. BGP determines the best path to the ASN, depending on the topology of network nodes and current network conditions.

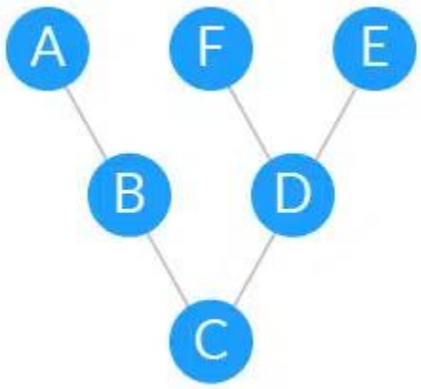
To use an analogy, an AS is like a city with many streets. A network prefix is one street, and an IP address is one particular house. Network packets are like cars traveling from one house to another, and BGP is like a navigation app that helps them take the best possible route.

A Brief History of Internet Traffic Routing
In the early days of the Internet, there were only a few networks connected to each other. As a result, routing between network nodes was quite static. All that needed to be done to set up routing was to define network nodes and make connections between them as needed.

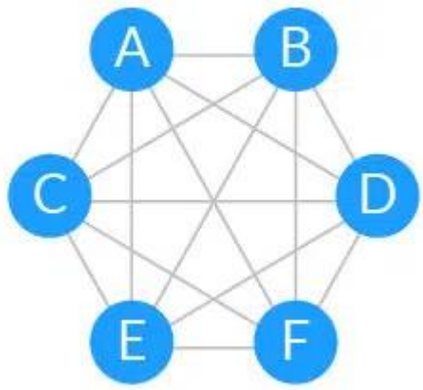
However, the Internet grew quickly, adding more and more networks, which necessitated a more dynamic routing system. EGP (External Gateway Protocol) was invented to do the job.

EGP is a simple routing protocol based on tree-like hierarchical topologies. In modern networks, tree topologies have become fully connected mesh topologies to allow for maximum scalability

Tree-like vs. full mesh topologies



In a tree-like topology, to reach E or F, A will have to go through B, C and D.



In a full mesh topology, nodes have many paths to reach each other.

First let's explain what an Autonomous System (AS) really is.

The Internet is a network of networks; it involves hundreds of thousands of smaller networks known as AS. Each of these networks consists in a large pool or routers run and administered by a single organization. Autonomous systems typically belong to Internet Service Providers (ISPs) or other large organizations (technology agencies, universities, government agencies, scientific institutions, …). Each AS is represented by a unique number called ASN (Autonomous System Number). Internet Assigned Numbers Authority (IANA) assigns ASNs to Regional Internet Registries (RIRs). These in turn assign them to ASNs owners.

In 2020, the number of ASNs nearly reaches 100.000. Already in the late 80's, the number of AS grew in such a way that the limitations of EGP became more pronounced.

In June 1989, the first version of a new routing protocol was formalized. Its name is BGP, which stands for Border Gateway Protocol. The current version of BGP is version 4, published in 2006 under RFC 4271.

Compared to its predecessor EGP, BGP supports fully meshed topologies, making multi-paths routing possible. BGP is used to route traffic from AS to AS. In this case, we are talking about eBGP (external BGP). It makes intelligent routing decisions based on different parameters like reliability, speed and cost. The routing is said to be "policy-based".

Within an AS, other routing protocols like OSPF, EIGRP and IS-IS can be freely used. More generally, we are talking about IGP (Interior Gateway Protocol) when it comes to routing within AS's. iBGP (interior BGP) is also one protocol that you may use for this.

How BGP routing works

When two ASs communicate with each other, they exchange information about their respective networks. This includes details such as IP address ranges, subnet masks, and other network-related data. This information is then used to build a routing table that contains all the possible routes between two ASs.

Route updates

Once a routing table has been built, it needs to be updated regularly so that any changes in the network can be reflected in the routing table. This process is known as route updating, and it involves sending messages back and forth between two ASs to keep their respective routing tables up-to-date.

BGP path attributes

In addition to exchanging information about networks, BGP also uses path attributes to determine which route should be taken when sending packets from one AS to another. These attributes include things like hop count, latency, and cost of transmission. By considering these factors, BGP ensures that packets are sent along the most efficient route possible.

Why is Border Gateway Protocol important?

BGP is an integral part of how the internet works today. Without it, there would be no way for different networks to communicate or share information about routes, making it impossible for data to be sent from one place to another efficiently and securely.

BGP enables the exchange of information between different networks and allows them to determine the best path for data to travel. It's used by ISPs, large organizations, and cloud providers to connect their networks with each other and

with the rest of the internet. BGP is also used to ensure traffic flows through the most efficient route possible, which helps reduce latency and improve performance.

8 main functions of BGP

BGP provides critical functions to the operation of the internet, including everything from maintaining route information, selecting the shortest route, and providing redundancy in case of routing errors, to providing security through authentication and facilitating communication between different network types.

1. Maintaining route information

BGP maintains an up-to-date routing table that regularly updates it with information about all available routes on the internet. BGP routers use this table to determine the best paths for sending packets from one network to another.

2. Selecting the best route for sending packets

BGP uses a variety of parameters, such as distance and latency, to calculate the best route for sending packets. BGP routers typically have multiple paths to choose from and will select the one that offers the best performance.

3. Providing redundancy in case of route failure

BGP will automatically reroute traffic over an alternative path if it detects that a primary path is not functioning.

4. Detecting loops in routing paths

BGP can detect and eliminate loops in routing paths using a set of algorithms known as the BGP Decision Process. This helps ensure packets are sent along the most efficient route possible without wasting bandwidth or taking unnecessary detours.

5. Preventing malicious attacks

BGP can filter out malicious traffic by verifying that BGP messages come from legitimate autonomous systems.

6. Providing security

BGP authenticates messages between routers using a preconfigured password or key. This helps ensure that only authorized entities can exchange information and keep malicious actors from disrupting traffic.

7. Controlling traffic flow

BGP enables ISPs to control how traffic flows through their networks by specifying the route taken when sending packets from one network to another.

8. Facilitating network communication

BGP allows communication between networks, such as IPv4 and IPv6. This helps ensure that all devices can communicate with one another, regardless of which type of network they're on.

What are common issues of BGP routing?

Despite its many benefits, there are some issues associated with using BGP for routing traffic across the internet—both in terms of general applicability (e.g., stability and configuration) and security (e.g., route manipulation and hijacking).

General BGP applicability issues

BGP routing has some important issues to be aware of, including propagation delay and potential instabilities caused by manual configuration.

Propagation delay

One of the main issues with using BGP is that changes made in one network can take a long time to propagate throughout all other connected networks. This can be a concern if you need to make changes quickly or if your network relies on up-to-date information.

Instability

Another issue is that BGP can cause instability if not configured correctly. If routes are not set up properly, packets may be routed inefficiently, leading to slow performance and potential outages.

Manual configuration

BGP requires manual configuration, which can be time-consuming and error-prone. This means that any mistakes in setting up the routes could lead to problems down the line. In addition, BGP does not scale well when dealing with large numbers of routers or large amounts of data being routed simultaneously.

BGP security issues
Some security concerns around BGP routing arise out of the general issues above and as a result of criminals actively trying to exploit BGP by manipulating or hijacking routes for malicious purposes.

BGP route manipulation
BGP route manipulation is a serious threat to the integrity of a network, as it involves malicious actors deliberately altering BGP tables to prevent traffic from reaching its intended destination. This can not only lead to data loss but can cause considerable disruption to service continuity and potentially become used in a range of cyberattack scenarios.

In addition, BGP route manipulation can damage route credibility and require users to manually vet or deploy additional security products to detect route manipulation attempts.

BGP route hijacking
This is a method of exploitation that allows attackers to announce a victim's IP address prefixes to reroute traffic through itself, leading to instability and increased load from the sudden influx of traffic. In some cases, BGP route hijacking could enable attackers to access unencrypted data streams or be used for bypassing IP blocklist mitigation for launching unsolicited campaigns like spam.

BGP denial-of-service (DoS)
This malicious attack primarily targets BGP routing protocols. In this attack, a cybercriminal sends unexpected or undesirable BGP traffic to the victim system, which exhausts all available resources, making it impossible to process valid BGP traffic.

Lesson 4. Multiprotocol Label Switching (MPLS)

What is Multiprotocol Label Switching (MPLS)?
Multiprotocol Label Switching (MPLS) is a switching mechanism used in wide area networks (WANs).

MPLS uses labels instead of network addresses to route traffic optimally via shorter pathways. MPLS is protocol-agnostic and can speed up and shape traffic flows across WANs and service provider networks. By optimizing traffic, MPLS reduces downtime and improves speed and quality of service (QoS).

History of MPLS
As the internet grew in popularity, organizations looked for an efficient way to perform packet forwarding. Bandwidth demands increased, but label-switching mechanisms struggled to handle the load. Traditional methods, such as IP switching and tag switching, require each router to independently determine a packet's next hop by inspecting its destination IP address before consulting its routing table. This slow process involves hardware resources and introduces the potential of degraded performance for real-time applications, such as voice and video. Traditional routers needed to scale more effectively to meet the bandwidth needs of the modern internet and avoid slow speeds, jitter and packet loss.

In 1997, the Internet Engineering Task Force (IETF) Multiprotocol Label Switching working group formed to create standards to help fix the issues around internet traffic routing. MPLS was developed as an alternative to multilayer switching and IP over asynchronous transfer mode (ATM). MPLS routers don't look up routes in routing tables, which helps boost the speed of network traffic. As MPLS techniques were developed and adopted throughout the early 2000s, the protocol became widely adopted.

MPLS can work in a multiprotocol environment, such as ATM, frame relay, Synchronous Optical Network and Ethernet. MPLS continues to evolve as backbone network technologies evolve, and the IETF working group still works on MPLS protocols and mechanisms. MPLS also played a significant role in the support of legacy network technologies, as well as newer technology based on IP networks.

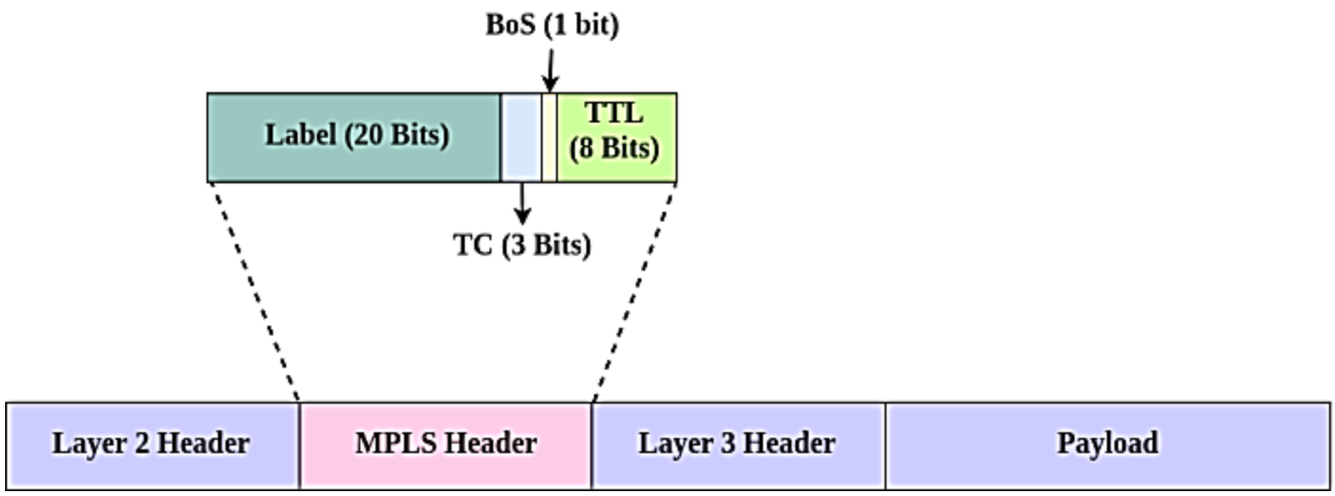
Components of MPLS
MPLS is defined by its use of labels instead of network addresses. This factor drives the flexibility and efficiency of MPLS.

A label is a four-byte -- 32-bit -- identifier that conveys the packet's predetermined forwarding path in an MPLS network. While a network address specifies an endpoint, a label specifies paths between endpoints. This

latter capability enables MPLS to decide the optimal pathway route of a given packet. Labels can also contain information about QoS and a packet's priority level.
MPLS labels consist of the following four parts:

Label value: 20 bits.
Experimental: 3 bits.
Bottom of stack: 1 bit.
Time to live: 8 bits.

MPLS Labels in an IP Packet are used to make forwarding decisions, but also to perform QoS, and TTL (Time To Live) values as well as demonstrated in this illustration:



As seen in the break down, an MPLS label is 32 bits in total size, comprised of:

- **20 Label bits** – The MPLS Label value from a range of 0-1048575
- **3 Traffic Class / EXP bits** – Used to perform QoS / CoS within the MPLS Network
- **1 Bottom of Stack bit** – Multiple labels can be “stacked” on an IP Packet, this label indicates which is on the bottom (more on that in a moment)
- **8 TTL Bits** – This gives the same 254 TTL value as an IP Packet so MPLS packets are not bouncing around the network endlessly, so nothing real tricky here

As seen in the break down, an MPLS label is 32 bits in total size, comprised of:

1. **(Label Bits)** The range of label values is a bit misleading as labels 0-15 are a reserved range of label values for built in features for MPLS, which all of them can be reviewed in depth on the IANA’s official website here.
2. **(TC / EXP Bits)** These are called “EXP” or “Experimental” bits in the CBT Nuggets Fundamentals series I am watching, however I have also seen these described as “TC” bits for Traffic Class which is built in QoS / CoS into the MPLS network which may be the updated use for them, so I decided to include both here in case you see it described as one or the other to avoid confusion on these bits in the label
3. **(BoS bit)** The “Bottom of Stack” bit is interesting in how it explains how MPLS works, as multiple MPLS Labels can be “stacked” onto an IP Packet, the Bottom of Stack label bit being on (1) indicates it is the closest label to the Layer 3 IP Header of the packet while all other labels not on the Bottom of Stack will have BoS bits set to off (0) and the Top most label (closest to Layer 2 header) is the Label preferred / used to forward the IP Packet.
4. **(TTL Bits)** The TTL 8-bits is pretty self-explanatory, each hop the TTL value is decremented by one starting from 254 in the MPLS label, and once the value is zero the packet is discarded.

MPLS is multiprotocol, which means it can handle multiple network protocols. MPLS is highly versatile and unifying, as it provides mechanisms to carry a multitude of traffic, including Ethernet traffic. One of the key differentiators between MPLS and traditional routers is it doesn't need specialized or additional hardware.

Below is an overview of MPLS:

- It forwards using labels, as opposed to network addresses.
- The label contains the service class, as well as the destination, of the packet.
- It operates between Layers 2 and 3 of the Open Systems Interconnection (OSI) model.
- It guarantees the bandwidth of paths.
- ATM (Asynchronous Transfer Mode) switches can act as routers, so no additional hardware is needed.

How an MPLS network works

In an MPLS network, packets are labeled by an ingress router -- a label edge router (LER) -- as they enter a service provider's network. The first router to receive a packet calculates the packet's entire path upfront. It also conveys a unique identifier to subsequent routers using a label in the packet header.

Every prefix in a routing table receives a unique identifier, and the MPLS service tells routers exactly where to look in the routing table for a specific prefix. This mechanism speeds up communication and traffic hopping.

MPLS works between the following OSI model layers:

Layer 2. The data-link layer, or switching level, which uses protocols such as Ethernet.
Layer 3. The routing layer, which covers traffic routing.

MPLS label traffic is sent via a label-switched path (LSP) inserted between the Layer 2 and Layer 3 headers. Label switch routers (LSRs) interpret the MPLS labels -- not the full IP address of any traffic. MPLS forwards data packets to Layer 2 of the OSI model, rather than passing to Layer 3. For this reason, MPLS is informally described as operating at Layer 2.5.

MPLS routing terminology

Label edge routers. LERs are the ingress or egress routers or nodes when an LSR is the first or last router in the path, respectively. LSRs label incoming data -- the ingress node -- or pop the label off the packet.

Label-switched paths. LSPs are the pathways through which packets are routed. An LSP enables service providers to decide the best way to flow certain types of traffic within a private or public network.

Label switch routers. LSRs read the labels and send labeled data on identified pathways. Intermediate LSRs are available if a packet data link needs to be corrected.

Pop. This mechanism removes a label and is usually performed by the egress router.

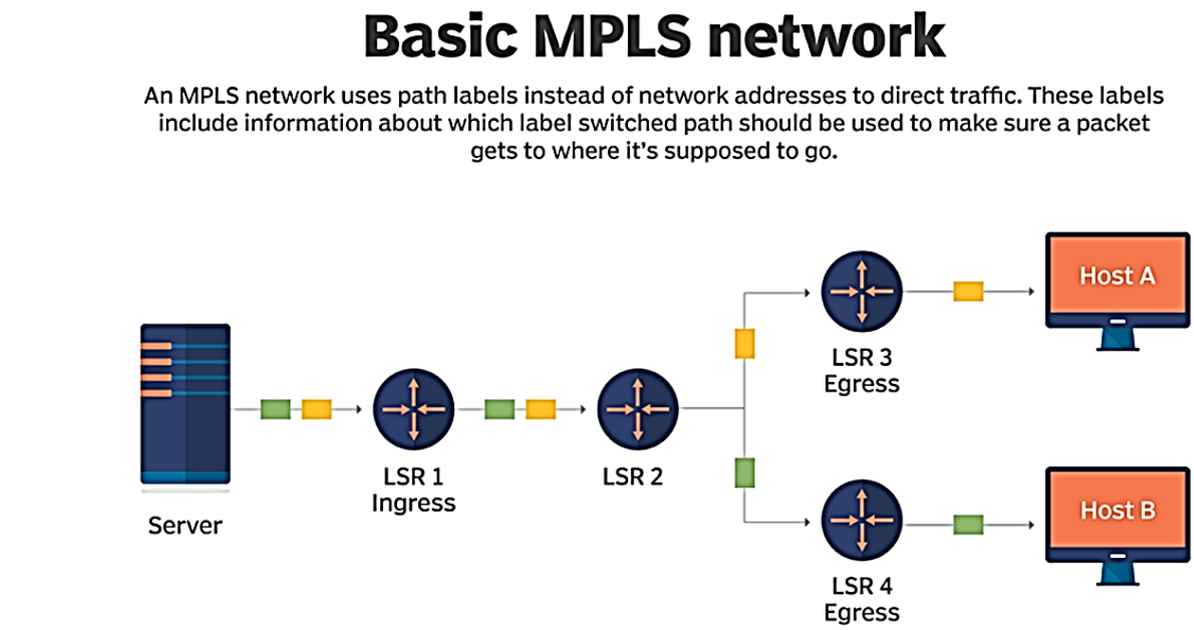
Push. This mechanism adds a label and is typically performed by the ingress router.

Swap. This mechanism replaces a label and is usually performed by LSRs between the ingress and egress routers.

MPLS routing terminology

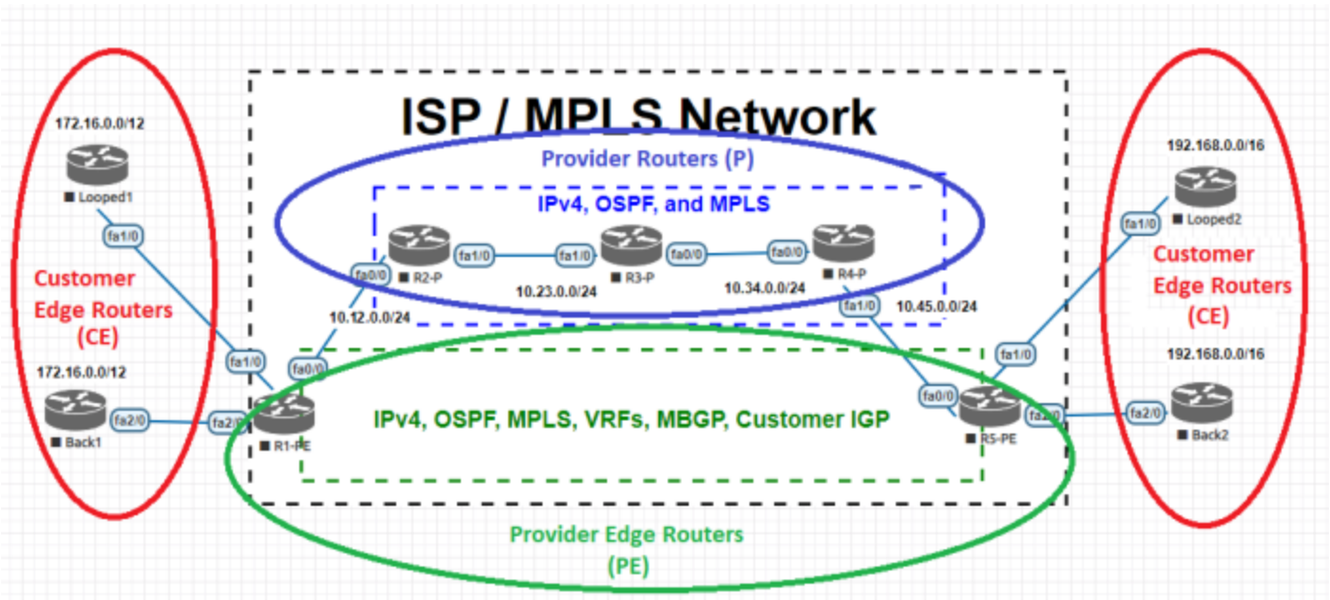
An example of how a packet travels through an MPLS network:

1. A packet enters the network through an LER.
2. The packet is assigned to a forwarding equivalence class (FEC). The FEC assignment depends on the type of data and the destination. FECs are used to identify packets with similar or identical characteristics.
3. The LER -- or ingress node -- applies a label to the packet and pushes it inside an LSP. The LER decides on which LSP the packet takes until it reaches its destination address.
4. The packet moves through the network across LSRs.
5. When an LSR receives a packet, it carries out the Push, Swap and Pop actions.
6. In the final step, the LSR -- or egress router -- removes the labels and then forwards the original IP packet toward its destination.



This diagram shows how an MPLS network uses path labels to direct traffic.

These different router types simply refer to where they are / who they talk to in the MPLS Network, as shown here:



Lesson 5. Quality of Service (QoS) in advanced networks

What is QoS in Networking?

Quality of service (QoS) is the use of mechanisms or technologies that work on a network to control traffic and ensure the performance of critical applications with limited network capacity. It enables organizations to adjust their overall network traffic by prioritizing specific high-performance applications.

QoS is typically applied to networks that carry traffic for resource-intensive systems. Common services for which it is required include internet protocol television (IPTV), online gaming, streaming media, videoconferencing, video on demand (VOD), and Voice over IP (VoIP).

Using QoS in networking, organizations have the ability to optimize the performance of multiple applications on their network and gain visibility into the bit rate, delay, jitter, and packet rate of their network. This ensures they can engineer the traffic on their network and change the way that packets are routed to the internet or other networks to avoid transmission delay. This also ensures that the organization achieves the expected service quality for applications and delivers expected user experiences.

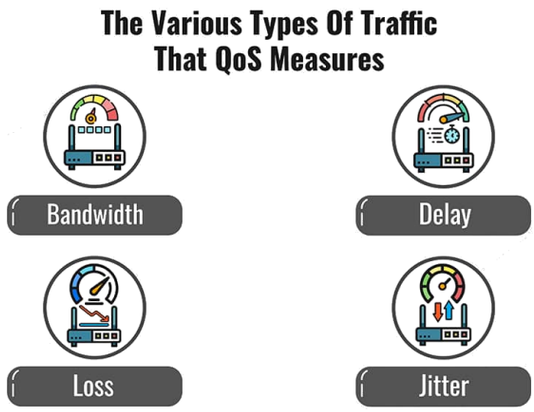
As per the QoS meaning, the key goal is to enable networks and organizations to prioritize traffic, which includes offering dedicated bandwidth, controlled jitter, and lower latency. The technologies used to ensure this are vital to enhancing the performance of business applications, wide-area networks (WANs), and service provider networks.

How Does QoS Work?
QoS networking technology works by marking packets to identify service types, then configuring routers to create separate virtual queues for each application, based on their priority. As a result, bandwidth is reserved for critical applications or websites that have been assigned priority access.

QoS technologies provide capacity and handling allocation to specific flows in network traffic. This enables the network administrator to assign the order in which packets are handled and provide the appropriate amount of bandwidth to each application or traffic flow.

Types of Network Traffic
Understanding how QoS network software works is reliant on defining the various types of traffic that it measures. These are:

1. **Bandwidth:** The speed of a link. QoS can tell a router how to use bandwidth. For example, assigning a certain amount of bandwidth to different queues for different traffic types.
2. **Delay:** The time it takes for a packet to go from its source to its end destination. This can often be affected by queuing delay, which occurs during times of congestion and a packet waits in a queue before being transmitted. QoS enables organizations to avoid this by creating a priority queue for certain types of traffic.
3. **Loss:** The amount of data lost as a result of packet loss, which typically occurs due to network congestion. QoS enables organizations to decide which packets to drop in this event.
4. **Jitter:** The irregular speed of packets on a network as a result of congestion, which can result in packets arriving late and out of sequence. This can cause distortion or gaps in audio and video being delivered.



How Does QoS in Computer Networks Work?

QoS facilitates the manipulation of packet loss, postponement, and jitter in your community infrastructure. Since we are operating with a finite quantity of bandwidth, our first order of enterprise is to become aware of what packages could benefit from handling those three things.

Once community and alertness directors become aware of the packages that want to have precedence over bandwidth in a community, the following step is to become aware of those visitors. There are numerous approaches to become aware of or mark the visitors. Class of Service (CoS) and Differentiated Services Code Point (DSCP) are examples.

We may utilize this information to set policies on those groups in order to give some data streams preferential treatment over others now that we can group data streams into different groups. Queuing is the term for this. The routing or switching device will advance these packets/frames to the front of the queue and transmit them right away, for instance, if voice traffic is tagged and a policy is developed to grant it access to the bulk of network bandwidth on a channel.

However, if a typical TCP data transfer stream is given a lower priority designation, it will wait (be queued) until enough bandwidth is available to send. These lower-priority packets/frames are the first to be dropped if the queues get overcrowded.

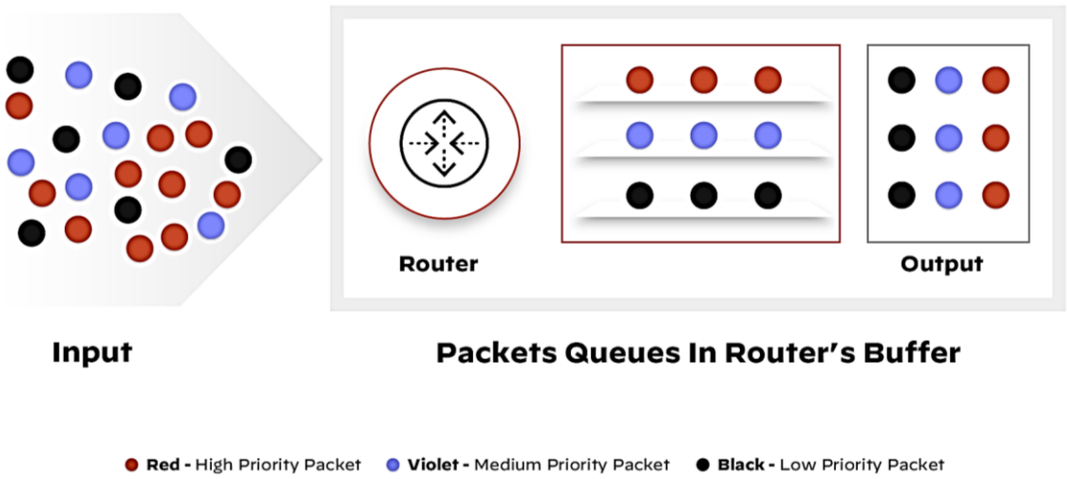
QoS networking generation works through marking packets to become aware of provider types, then configuring routers to create separate digital queues for every utility, primarily based totally on their precedence.

As a result, bandwidth is reserved for essential packages or websites that have been assigned precedence to get entry.

QoS technology offers the ability and manage allocation to particular flows of community visitors. This allows the community administrator to assign the order wherein packets are treated and offer an appropriate quantity of bandwidth to every utility or visitor going with the drift.

The queuing mechanism allows for packets within traffic flows to be stored until the network is ready to process it. Priority queuing (PQ) ensures necessary availability and minimal latency of network performance. The most important applications and traffic are assigned priority and bandwidth based on their classification.

This ensures the most important activities on a network are not starved of bandwidth by activities of lower priority. Applications, users and traffic can be batched in up to eight differentiated queues.



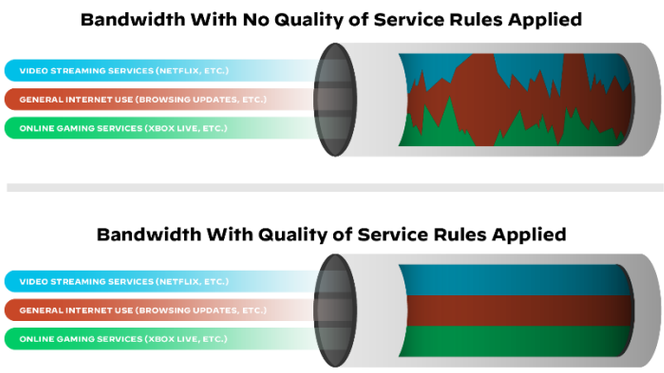
The process of classifying and prioritizing network data

Bandwidth management mechanisms measure and control traffic flows on the network. Preventing exceeding its capacity allows for network congestion avoidance that occurs.

Mechanisms for bandwidth management include:

1. **Traffic shaping** — a rate limiting technique used to optimize or guarantee performance and increase usable bandwidth where necessary.

2. **Scheduling algorithms** — algorithms that offer varied methods for providing bandwidth to specific traffic flows.



Visualizing bandwidth with and without quality of service rules

Why is QoS Important?
QoS enables an organization to prioritize traffic and resources to guarantee the promised performance of a specific application or service. It also enables enterprises to prioritize different applications, data flows, and users in order to guarantee the optimum level of performance across their networks.

Network QoS plays a vital role in helping network administrators manage limited bandwidth within the network. QoS ensures the availability of real-time applications such as online video meetings, voice calls, and video conferencing. For instance, packet loss during voice or video communication directly impacts the quality of the video/call for end users. Outlined below are some of the advantages of QoS:

- **Improved User Experience:** QoS works by identifying the traffic flow within the network and prioritizing it accordingly. It ensures critical applications run at their best and are available with fast response times for users.
- **Traffic Management:** With QoS, network admins can better manage traffic flow by setting different bandwidths for different types of packets. This prioritization helps better drive traffic and avoid potential network congestion.
- **Cost Reduction:** QoS enables better management of network resources. It reduces the need for organizations to upgrade network bandwidth and purchase additional network infrastructure.
- **Improved Security:** QoS can detect abnormalities in the network. Network admins can block unwanted traffic and ensure application reliability by setting specific QoS security policies.
- **Reduced Packet Loss:** Network congestion can lead to packet loss and hamper the performance of critical applications. QoS prioritization policies ensure packets get queued accordingly to avoid traffic jams within the network.

Advantages of QoS
The deployment of QoS is crucial for businesses that want to ensure the availability of their business-critical applications. It is vital for delivering differentiated bandwidth and ensuring data transmission takes place without interrupting traffic flow or causing packet losses. Major advantages of deploying QoS include:

- **Unlimited application prioritization:** QoS guarantees that businesses' most mission-critical applications will always have priority and the necessary resources to achieve high performance.
- **Better resource management:** QoS enables administrators to better manage the organization's internet resources. This also reduces costs and the need for investments in link expansions.
- **Enhanced user experience:** The end goal of QoS is to guarantee the high performance of critical applications, which boils down to delivering optimal user experience. Employees enjoy high performance on their high-bandwidth applications, which enables them to be more effective and get their job done more quickly.
- **Point-to-point traffic management:** Managing a network is vital however traffic is delivered, be it end to end, node to node, or point to point. The latter enables organizations to deliver customer packets in order from one point to the next over the internet without suffering any packet loss.
- **Packet loss prevention:** Packet loss can occur when packets of data are dropped in transit between networks. This can often be caused by a failure or inefficiency, network congestion, a faulty router, loose connection, or poor signal. QoS avoids the potential of packet loss by prioritizing bandwidth of high-performance applications.
- **Latency reduction:** Latency is the time it takes for a network request to go from the sender to the receiver and for the receiver to process it. This is typically affected by routers taking longer to analyze information and storage delays caused by intermediate switches and bridges. QoS enables organizations to reduce latency, or speed up the process of a network request, by prioritizing their critical application.

Chapter 2: Network Design and Optimization

In an era defined by the rapid exchange of information and the proliferation of digital technologies, the architecture and efficiency of computer networks have become paramount. The course on Network Design and Optimization offers a comprehensive exploration of the fundamental principles and advanced strategies essential for constructing robust, scalable, and high-performing networks. This course is designed to equip students with the knowledge and skills needed to design, implement, and manage network infrastructures that meet the evolving demands of modern businesses, institutions, and industries.

Network Design encompasses the strategic planning and configuration of interconnected systems, focusing on elements such as topology, hardware selection, security protocols, and traffic management. It is the cornerstone upon which a reliable and efficient network is built. Optimization, on the other hand, delves into the techniques and methodologies for enhancing network performance, minimizing latency, and maximizing resource utilization. Through a combination of theoretical instruction and hands-on exercises, this course aims to empower students to analyze real-world network scenarios, identify bottlenecks, and implement solutions that optimize network efficiency, ensuring seamless communication and data transfer across diverse environments. By the end of this course, students will have the proficiency to architect networks that not only meet current requirements but are also adaptable to future technological advancements and organizational growth.

Lesson 1: Designing scalable and high-performance networks

Network architecture is the backbone of any IT system, but it can also be a source of frustration and downtime if not designed properly. How can you create a network architecture that can handle increasing demands, avoid bottlenecks, and recover from failures? In this article, we will explore some best practices and principles for network engineering that can help you achieve these goals.

Network scalability refers to how well a network can handle sudden changes in workload brought about by sudden spikes or drops in the volume of data it processes. If, for instance, you went on a massive hiring spree and upped your staff count from 100 to 300 people in one day, your network can be described as very scalable if its performance can be tripled while you only double your network spend.

Understanding Network Infrastructure

What is Network Infrastructure?
Network infrastructure refers to the foundation of interconnected devices, hardware, software, and communication protocols that facilitate the flow of data within an organization and beyond. It encompasses everything from routers, switches, and access points to cables, servers, and firewalls. A well-structured network infrastructure enables smooth data transmission, resource sharing, and internet connectivity for all users.

Importance of a Scalable Network Infrastructure
A scalable network infrastructure is designed to accommodate the growing needs of a business without compromising performance, security, or reliability. As a company expands, its network demands also increase. A scalable infrastructure ensures that these demands can be met seamlessly, allowing businesses to adapt to higher workloads and user traffic.

Components of a Scalable Network Infrastructure
Network Devices
Building a scalable network infrastructure starts with selecting the right network devices. These include routers to direct data traffic, switches to connect devices within a local area network (LAN), and access points for wireless connectivity. Choosing reliable and high-performance devices is essential for long-term scalability.

Network Topology
The network topology defines the layout of devices and connections within the infrastructure. For scalability, a flexible and expandable topology such as a mesh or hybrid network is preferred over a rigid one. Mesh networks provide multiple pathways for data to travel, reducing the risk of network bottlenecks.

Network Security
Robust network security is vital for protecting sensitive data and preventing unauthorized access. A scalable network infrastructure incorporates firewalls, intrusion detection systems (IDS), and encryption protocols to safeguard information from potential threats as the network grows.

Bandwidth Management
As more devices and users join the network, effective bandwidth management becomes crucial. Scalable network infrastructures employ Quality of Service (QoS) techniques to prioritize critical traffic and ensure consistent performance.

Redundancy and Failover
To maintain uninterrupted connectivity, redundancy and failover mechanisms are integrated into a scalable network infrastructure. Redundant devices and connections act as backups in case of primary component failures.

Scalable IP Addressing
With the depletion of IPv4 addresses, adopting IPv6 becomes necessary for a scalable network infrastructure. IPv6 offers a vast pool of unique addresses, accommodating the growing number of devices connected to the internet.

Why network scalability matters
Network scalability empowers business scalability. If your business relies on a dependable Internet connection and IT infrastructure to thrive, then it needs those to be scalable in order to grow.

Without that, your business risks falling behind, without a connection that can maintain your day-to-day needs. You face the potential of wasting time and money when your employees are unable to do their jobs right. And there may even be service interruptions, putting your relationship with customers at risk.

What to consider when it comes to scalability

- **Your bandwidth requirements**: How much bandwidth does your current business require? This will depend on the number and types of devices you're relying on, as well as the overall number of employees relying on the Internet to do their jobs.
- **Your space constraints**: Do you have the physical space available to accommodate the networking hardware you'll require to run your business? And on a similar note, do you have the power available to stay up to date with your networking needs – including backup power so that you don't lose your connection during outages? The hardware requirements will vary depending on the size of your business and your specific requirements, but may take dedicated onsite space to accommodate.
- **How your business is growing**: If you're planning on expanding your business, there's a good chance that your bandwidth requirements will change – after all, you'll have more people using more devices. So look ahead. Do you have the flexibility available to easily adjust your network capabilities and extend your hardware as you need to?

High-Performance network (HPN)

We define a high-performance network (HPN) as a communication network that supports a large variety of user applications and that is scalable. In order to support many applications, the network must be able to transfer user traffic at high speed and with low delay. It must be able to allocate resources in ways that match the application requirements. Network organization and management must be flexible so that new applications can be supported as the need arises.

To implement a high-performance network, a number of critical bottlenecks must be addressed. These bottlenecks arise at all layers of the network operations.

A scalable network can accommodate growing numbers of users without degradation in performance. Growth is usually accommodated by interconnecting distinct networks. The network must be able to provide connectivity over a growing span. The span may be expressed in distance, number of links, or number of subnetworks.

An HPN that supports a wide range of current and future applications and that can accommodate growth is built and managed differently from networks that are designed for a specific application or user population.

At one extreme, phone networks have many nodes and operate over very large distances, but users can transfer data only at low speeds. (Thus phone networks are scalable, but support limited applications.) At the other extreme, a computer backplane bus operates at high speeds but connects only a small number of devices very close to each other. Local area networks or LANs (e.g., Ethernet) can transfer data between tens of nodes at moderate speeds (10 Mbps) over moderate distances (1 km). More recent LANs support speeds of 1 Gbps. Wide area networks or WANs, such as X.25 networks and the Internet, connect hundreds of nodes over hundreds of kilometers, but they operate at limited speeds of a few Mbps or less. Metropolitan area networks (MANs–e.g., DQDB, FDDI, SMDS) run at a higher speed and connect users separated by about 100 km. Frame Relay networks are streamlined versions of X.25 networks that can operate at high speed. The "backbone" telephone network is an HPN: it comprises the switches and links or trunks connecting them, but excludes the low-speed links connecting user telephones to the switches.

The precise values of the user transfer rate, the acceptable delay, the network span, and the number of users that characterize an HPN are somewhat arbitrary. We have in mind a user rate that exceeds 100 Mbps, delays on the order of 100 ms, a span of at least 100 m, and a number of users that can exceed 100. What is essential for a network to qualify as an HPN is for it to be able to support demanding services such as interactive MPEG video and LAN interconnections among many users.

Lesson 2: Network virtualization and software-defined networking (SDN)

What Is Network Virtualization?

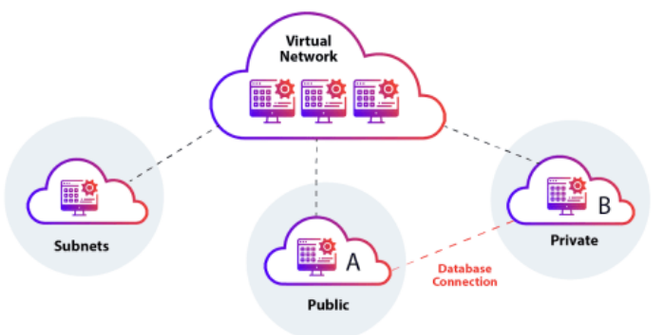
Network virtualization represents the administration and monitoring of an entire computer network as a single administrative entity from a single software-based administrator's console.

Network virtualization can include storage virtualization, which contains managing all storage as an individual resource. Network virtualization is created to enable network optimization of data transfer rates, flexibility, scalability, reliability, and security. It automates many network management functions, which disguise a network's true complexity. All network servers and services are considered as one pool of resources, which can be used independently of the physical elements.

Virtualization can be defined as making a computer that runs within another computer. The virtual computer, or guest device, is a fully functional computer that can manage the same processes your physical device can. The processes performed by the guest device are separated from the basic processes of your host device. You can run several guest devices on your host device and each one will identify the others as an independent computer.

Network services are decoupled from the physical hardware they run on. They can be used independently, making them perfect for any network device. With this shift to programmable networks, we can more flexibly provision networks, more securely manage them, and programmatically and dynamically manage them.

Network virtualization simplifies life for network administrators by making it easier to move workloads, modify policies and applications, and avoid complex and time-consuming reconfigurations when performing these tasks. In addition, customers and business people need instant access to various content, services, and information.



How does network virtualization work?
Network virtualization results from network virtualization software, which simulates the presence of physical hardware, like routers, switches, load balancers, and firewalls. In layman's terms, a network virtualization implementation may virtualize components spanning multiple layers of the Open Systems Interconnection Model. These include ones at Layer 2 (switches) and Layer 4 and beyond (load balancers, firewalls, etc. So, for example, in an SD-WAN solution, you can manage your virtual appliances using a management tool.

Network virtualization software creates virtual representations of a network's underlying hardware and software. This enables you to combine virtualized representations of underlying hardware and software into a single administrative unit. A virtualized environment allows the resources to be hosted inside virtual machines (VMs) or containers and run on top of off-the-shelf commercial x86 hardware to reduce costs. Network virtualization is a technology that allows for workloads to be deployed over a virtual network. Current network policies ensure that the correct network services are coupled with each VM- or container-based workload.

What are the different types of network virtualization?
There are two broad categories of network virtualization: External and internal network virtualization.

External network virtualization

The goal of the external network virtualization is to allow for seamless interoperation of physical networks and thus allow for better administration and management. Network switching hardware and virtual local area network (VLAN) solutions are used to create a VLAN.

In this VLAN, hosts attached to different physical LANs can communicate as if they were all in the same broadcast domain. This type of network virtualization is prevalent in data centers and large corporate networks. A VLAN may separate the systems on the same physical network into smaller virtual networks.

Internal network virtualization

Network virtualization entails creating an emulated network inside an operating system partition. The guest VMs inside an OS partition may communicate with each other via a network-like architecture, via a virtual network interface, a shared interface between guest and host paired with Network Address Translation, or some other means. Internal network virtualization can help prevent attacks on your internal network by isolating applications that might be vulnerable to malicious threats. Networking solutions that implement it are sometimes marketed as "network-in-a-box" offerings by their vendors.
Many organizations are also taking advantage of cloud technologies to further their network virtualization objectives. Network virtualization in cloud computing follows the same basic idea, but instead relies on cloud-based resources to create a working virtual network.

Simply put, the question "what is network virtualization?" can be answered as the ability to run networks uncoupled from your hardware. This allows for certain advantages.

Advantages of Network Virtualization

- **Lower hardware costs** − With network virtualization, entire hardware costs are reduced, while providing a bandwidth that is more efficient.

- **Dynamic network control** − Network virtualization provides centralized control over network resources, and allows for dynamic provisions and reconfiguration. Also, computer resources and applications can connect with virtual network resources precisely. This also enables for optimization of application support and resource utilization.

- **Rapid scalability** − Network virtualization generated an ability to scale the network rapidly either up or down to handle and make new networks on-demand. This is a valuable device as enterprises transform their IT resources to the cloud and shift their model to an 'as a service'.

Types of Network Virtualization

- **Network Virtualization** − Network virtualization is a technique of combining the available resources in a network by splitting up the available bandwidth into different channels, each being separate and distinguished.

- **Server Virtualization** − This technique is the masking of server resources. It simulates physical servers by transforming their identity, numbers, processors, and operating frameworks. This spares the user from continuously managing complex server resources. It also makes a lot of resources available for sharing and utilizing, while maintaining the capacity to expand them when needed.

- **Data Virtualization** − This type of cloud computing virtualization technique is abstracting the technical details generally used in data management, including location, performance, or format, in favor of broader access and more resiliency that are directly related to business required.

- **Application Virtualization** − Software virtualization in cloud computing abstracts the application layer, separating it from the operating framework.

Virtual vs. physical networks

While virtual networks seem to be the best choice for upgrading existing networking infrastructure, we must not forget about the hardware needed along with other aspects of network virtualization. Let's discuss the advantages and disadvantages of a virtual network and its physical counterpart.

Advantages of virtual networks:

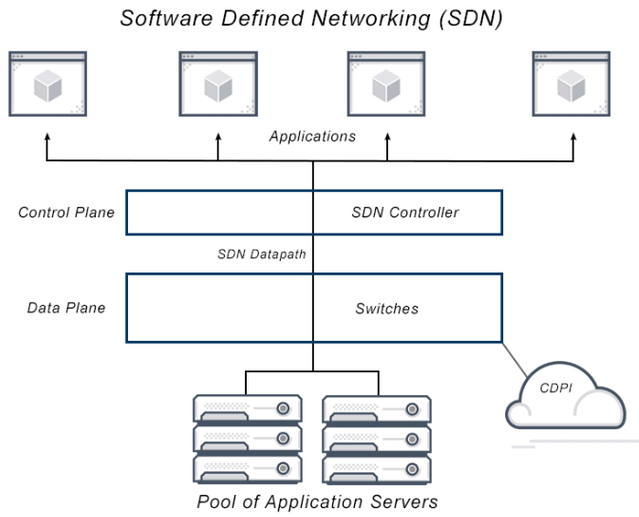
- **Scalability:** Virtual networks can be easily scaled up or down to meet changing needs without purchasing or installing new hardware.
- **Flexibility:** Virtual networks are modified rapidly and efficiently to meet the needs of various applications and users.
- **Cost-effectiveness:** Virtual networks are often less expensive to set up and maintain than physical networks, as they do not require the purchase of physical hardware.
- Improved security: Virtual networks can be isolated from one another and can have specific security policies and configurations applied to them.

Disadvantages of virtual networks:

- **Performance:** Virtual networks can have performance limitations compared to physical networks, particularly in network latency and bandwidth.
- **Dependence on underlying infrastructure:** Virtual networks depend on the underlying physical infrastructure and hardware, so if there is a problem with this infrastructure, it can affect the virtual network.
- **Complexity:** Virtual networks can be more complex to manage and maintain than physical networks, as they require expertise in both virtualization and networking.

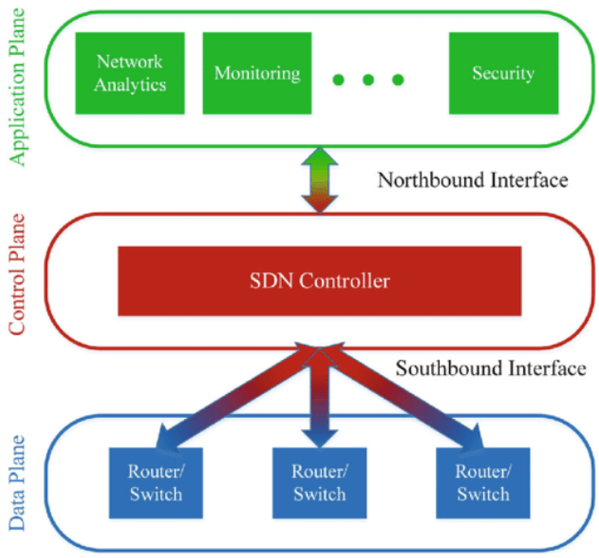
What Is Software-Defined Networking

Software Defined Networking (SDN) is an architecture that gives networks more programmability and flexibility by separating the control plane from the data plane. The role of software defined networks in cloud computing lets users respond quickly to changes. SDN management makes network configuration more efficient and improves network performance and monitoring.



Architecture of SDN

An SDN architecture typically includes three main layers: application plane, control plane, and data plane.



1. **Application Layer:** This is the top layer of the SDN architecture, and it is responsible for defining the desired behavior of the network. Applications at this layer might include traffic engineering tools, security policies, or virtual network overlays.
2. **Control layer:** The control layer is responsible for implementing the policies and rules defined at the application layer. It is typically implemented as a central controller that communicates with the network devices in the data plane.
3. **Data Plane or Infrastructure Layer:** This layer consists of the physical network devices, such as switches and routers, that make up the data plane. These devices are responsible for forwarding network traffic through the network.

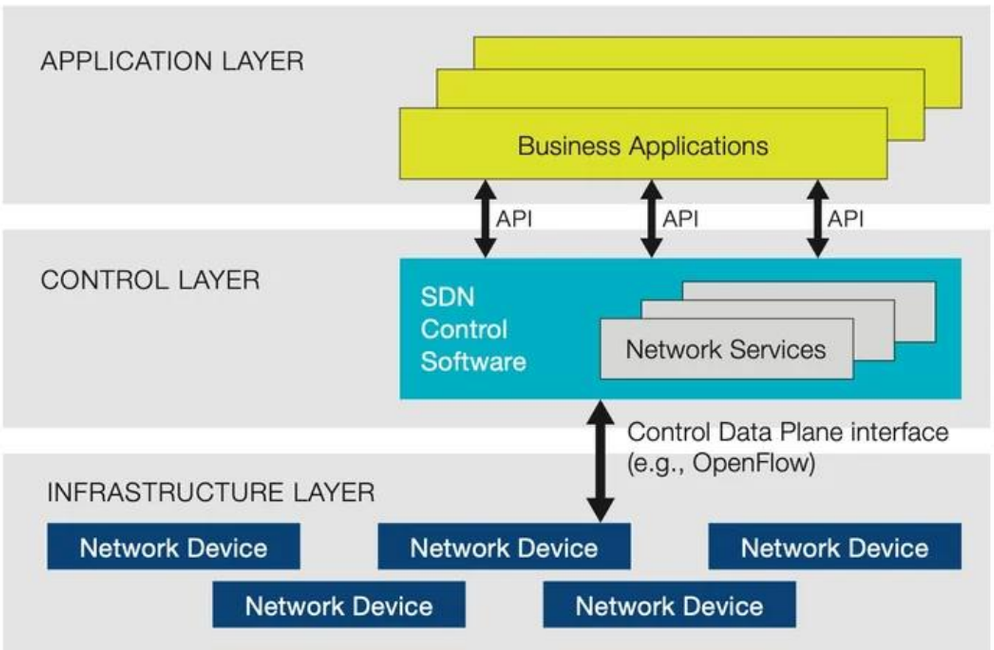
The Northbound and Southbound interfaces are used to facilitate communication between the different layers of the architecture. Integrating these three layers allows the network to operate in a coordinated and efficient manner.

Northbound APIs: Applications using an SDN rely on the controller to tell them what the status of the network infrastructure is so that they can know what resources are available. Additionally, the SDN controller can automatically ensure application traffic is routed according to policies established by network administrators.

Southbound APIs: The SDN controller communicates with the network infrastructure, such as routers and switches, through southbound APIs. The network infrastructure is told what path the application data must take as decided by the controller. In real time, the controller can change how the routers and switches are moving data. The data no longer relies on the devices and routing tables to determine where the data goes. Instead, the controller's intelligence makes informed decisions that optimize the data's path.

SDN Controllers

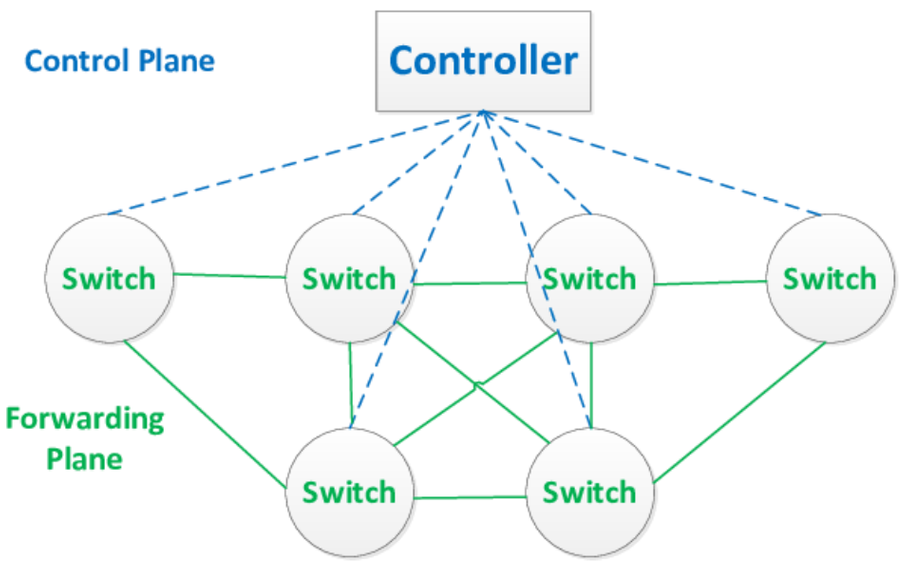
An SDN controller is the software that provides a centralized view of and control over the entire network. Network administrators use the controller to govern how the underlying infrastructure's forwarding plane should handle the traffic. The controller is also used to enforce policies that dictate network behavior. Network administrators establish policies that are uniformly applied to multiple nodes in the network. Network policies are rules that are applied to traffic that determines what level of access it has to the network, how much resources it is allowed, or what priority it is assigned.



The application, control, and infrastructure layers are kept separate in SDN and communicate through

How SDN works?

In an SDN network, the control plane and the data plane are separated. The control plane makes decisions about how traffic is forwarded through the network, while the data plane is responsible for forwarding traffic according to those decisions.



The control plane is implemented using a central controller, a software application that runs on a single server or a set of servers. The controller maintains a global view of the network and uses this view to make decisions about how traffic should be forwarded. It does this by communicating with the data plane elements in the network, which are known as "forwarding elements" or "switches."

These switches in an SDN network are typically "open," meaning they can be controlled and programmed by external software rather than being hard-coded with a fixed set of rules for forwarding traffic. As a result, the controller can configure the switches to transmit traffic in the desired manner.

To control the switches, the controller communicates with them using a southbound API, a set of protocols and interfaces that the controller can use to send instructions to the switches and receive status information from them. And the controller uses northbound APIs to communicate with higher-level applications and systems that need to use the network, such as applications running in the cloud.

In this way, the controller acts as the “brain” of the network by making decisions about how traffic should be forwarded and communicating those decisions to the switches, which act as a “muscle” of the network, carrying out the instructions received from the controller and forwarding traffic accordingly.

Features of SDN

There are several key features of SDN that distinguish it from traditional networking architectures:

- **Flexibility:** Changes to the network can be made without physically reconfiguring devices which allows network managers to react swiftly to evolving requirements and circumstances.
- **Programmability:** It is possible to programmatically control the behavior of the network using APIs or other software development tools. This makes it easier to automate network tasks and integrate the network with other systems.
- **Abstraction:** In an SDN architecture, the control plane is separated from the data plane, which forwards the traffic. This helps engineers easily change how the network operates without affecting the forwarding traffic devices.
- **Virtualization:** It also allows for the virtualization of networking resources, allowing administrators to create virtual networks on demand. This can be particularly useful in cloud computing environments where the demand for networking resources can be highly dynamic.

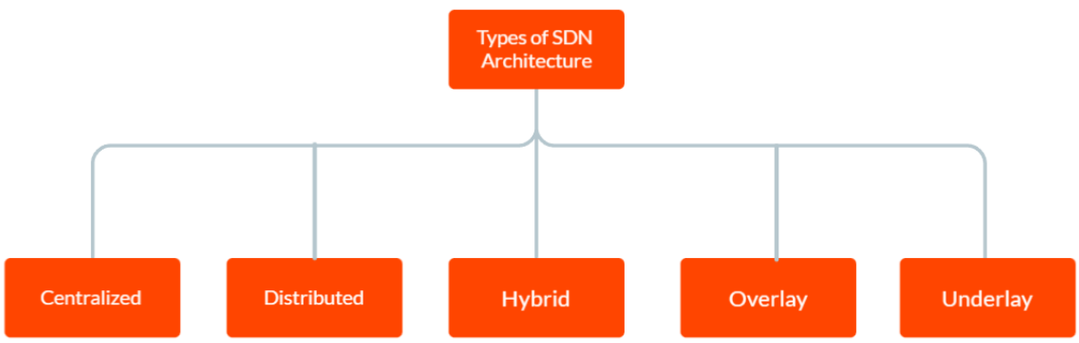
In addition to the features listed above, the primary advantage of using SDN is that it enables businesses to simulate their physical networking infrastructure in software, thereby lowering overall capital expenses (CAPEX) and operating expenses (OPEX).

Types of SDN Architectures

In general, different types of networks may require different approaches to SDN.

For example, a large enterprise network with many different types of devices and a complex topology may benefit from a hybrid SDN architecture, which combines elements of both centralized and distributed SDN. Conversely, a centralized SDN design might work well for a smaller network with fewer devices and a simpler topology.

It is important to carefully evaluate the different options and choose the architecture that best meets the needs of the organization. SDN primarily uses five different architecture models.



1. Centralized SDN

In a centralized SDN architecture, all control and management functions are consolidated into a single central controller, which allows administrators to define and control the behavior of the network easily. Still, it can also create a single point of failure.

2. Distributed SDN

In this architecture type, the control functions are distributed among multiple controllers, improving reliability but making it more complex to manage the network.

3. Hybrid SDN
Hybrid SDN architecture model combines centralized and distributed SDN elements. It may use a centralized controller for some functions and distributed controllers for others, depending on the needs of the network.

4. Overlay SDN
Overlay architectures use virtual networking technologies, such as VXLAN or NVGRE, to create a logical network on top of an existing physical network. This allows administrators to create virtual networks that can be easily created, modified, and deleted.

5. Underlay SDN
Underlay architecture utilizes the existing network infrastructure to support the creation of virtual networks that may use technologies such as MPLS or segment routing to create virtual links between devices in the network.