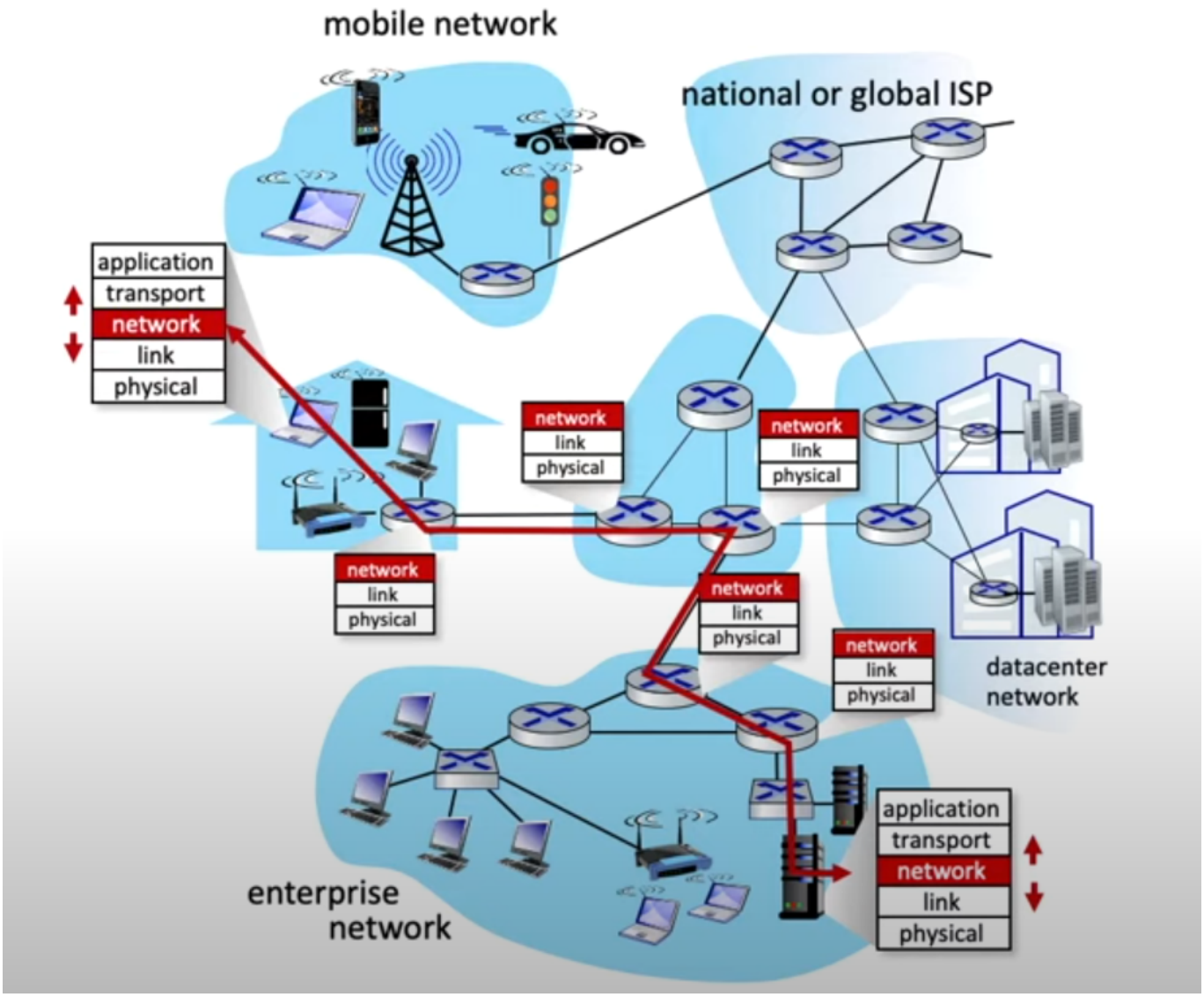
Review for Chapter 4 in the book "A Top-Down Approach to Networking"

Introduction of the Network Layer (4.1)

The network layers primary purpose is to move packets from one host to another across multiple networks.

This layer is designed to handle diverse and interconnected networks (E.G Ethernet, WiFi, Fiber Optics) while maintaining seamless data transfer.



Key Functions of the Network Layer

Routing: Determines the path packets will take from the source to the destination.

Routing Example: Imagine a user in Virginia accessing a website hosted in France. The network layer ensures that packets travel through multiple routers across different networks (such as local ISPs and international fiver links) based on optimal paths determined by routing protocols.

Forwarding: Moves pakcets to the next hop (intermediate router) on their path towards the final destination

Forwarding Example: A packet sent from a user's computer to a server is forwarded by routers along the way. Each router examines the packet's destination IP address, consults its routing table, and forwards the packet to the next router in line until it reaches the final destination.

Addressing: Each device (or host) on the network is assigned an IP address, which helps identify where packets should be sent.

IP Addressing Importance: IP Addresses are crucial for the network layer's operation. In IPv4, addresses are 32-bit numbers, allowing around 4.3 billion unique addresses. IPv6 addresses, with 128 bits, accommodate a vastly larger address space.

Internet Protocol (IP)

IP is the fundamental protocol for addressing and routing on the internet.

IP is a connectionless protocol, meaning it doesn't establish a dedicated end-to-end connection before sending data. Instead, packets may travel independently across different routes.

Best-Effort Delivery

IP is a best-effort service, meaning it doesn't guarantee packet delivery, ordering, or duplicate protection.

This lack of guarantees result from IP's design for simplicity and scalability, which prioritizes quick routing decisions.

Example: Unlike a phone call, where a circuit is established and maintained until the call ends, IP packets are sent independently, each potentially taking a different route. This model's benefit is resilience and efficiency, as the network dynamically adapts to conditions like traffic congestion or link failure.

Packets and Datagrams

In IP, data is divided into packets (or datagrams) that contain a header and data payload.

Each packet header includes important information, such as the source and destination IP addresses, which are used to guide the packet though the network.

End Systems and Routers

End Systems: (e.g Laptops, Smartphones, Servers) are the devices that send or receive data. They typically use applications that rely on transport layer protocols like TPC or UDP.

Routers: are intermediary devices that manage packet forwarding. They use routing tables to make forwarding decisions based on IP addresses.

Protocols Supporting Network Layer Functions

ARP (Address Resolution Protocol) maps IP addresses to physical hardware (MAC) addresses within a local network.

ICMP (Internet Control Message Protocol) is used by network devices to send error messages and operational information (e.g, destination unreachable, time exceeded).

Routing Protocols such as RIP, OSPF, and BGP help routers dynamically determine optimal paths for data transfer across networks.

what are "hops"

Hops refer to each point a data packet passes through on its journey from a source to a destination. Intermediate routers are the networking devices that make these hops possible, by forwarding data packets closer to their final destination. Each router that handles a packet represents one hop, and the more hops a packet requires, the more routers it has to traverse.

Key Concepts

Hop Count: The hop count is a measure of how many intermediate routers a packet crosses. This value is trancked in the packet's header as it moves through the netowrk. The hop count can affect latency, with more hops potentially increasing the time it takes for a packet to reach its destination.

TTL (Time-To-Live): Each packet has a field called the TTL, which sets a limit on the number of hops a packet can take before being discarded. Every time the packet hits an intermediate router, the TTL decreases by one. If the TTL reaches zero, the packet is dropped to prevent endless looping in the network.

Example: Consider a user in New York accessing a website hosted in Berlin. The packet carrying the request might pass through multiple hops, starting from the user's local router, moving to their ISP's gateway, and then traversing multiple backbone routers across continents. Each of these points is an intermediate router, working together to efficiently guide the packet to its final destination.

Intermediate Routers: Intermediate routers are essential in dynamic routing, where they use protocols like BGP (Border Gateway Protocol) to determine the best path for each packet based on real-time network conditions.

Routing Tables

A routing table is a data file stored in a router or networked device that helps it determine where to forward packets based on their destination IP addresses. The table contains rules and entries to guide the forwarding of data toward its destination across interconnected networks. Each entry in a routing table corresponds to a particular network path and typically includes several key peices of information.

Key Components of a Routing Table

```
Destination Address: the IP address of the destination network or host
Subnet Mask: used to determine which portion of an IP address refers to the
network
Next Hop: the IP address of the next router where the packet should go
Metric: Measure of route's desirability, such as hop count, cost, or delay. Lower
metrics usually indicate preferred paths
Interface: The network interface that should be used to forward the packet
```

Types of Routing Tables

Directly Connected Routes: Created automatically when an interface is configured with an IP address, representing networks directly connected to the router.

Static Routes: Manually entered by a network administrator to define specific paths for certain destinations.

Dynamic Routes: Automatically updated through routing protocols (e.g OSPF, BGP) to adapt to changing network conditions.

How Routing Tables Work

When a packet arrives, the router examines its destination address and compares it with the entries in its routing table. The router will select the route that best matches the destination and forward the packet to the specified next hop. If no match is found, the router may forward the packet to a default route it one exists, or it may discard the packet.

Example Scenario

Imagine a packet needs to go from a device in Network A to Network B. If router 1 connects these networks, its routing table might have an entry for Network B. The entry will specify the packets destined for Network B, which should be forwarded through a particular interface or to the next hop, router 2, which is closer to Network B.

RIP, OSPF, and BGP

RIP, OSPF, and BPG are three distinct routing protocols that routers use to determine the best path for fowarding packets in a network. Each protocol serves a specific purpose, operates on different scales, and uses unique methods for routing.

Routing Information Protocol (RIP)

Purpose: RIP is a simple, distance-vector protocol used mainly in smaller networks or networks with fewer requirements for rapid updates.

Operation: RIP calculates the best route based on the number of `hops` a packet must through to reach its destination.Each hop increases the 'distance' by one, and RIP has a maximum hop count of 15, limiting its use to small or medium-sized networks.

Updates: RIP periodically broadcasts the entire routing table to all neighboring routers, typically every 30 seconds.

Drawbacks: Due to its slow convergence (time to update routers) and limitation on hop count, RIP is not ideal for large networks, where fast and complex routing decisions are needed.

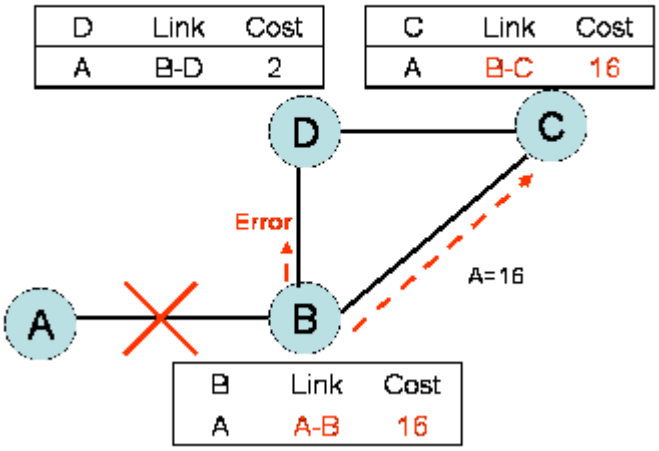
Poison Reverse

Poison Reverse is a technique where a router explicitly advertises a route back to the source with an infinite metric to break loops.
Instead of staying silent (like split horizon), Poison Reverse actively tells the neighbor, "The route to this

network through me is invalid."
The infinite metric, ensures that the route is not used.

Poison Reverse Example

Router A tells Router B it can reach Network X.
Router B learns about Network X via Router A.
Router B then advertises to Router A: "Network X is unreachable through me."

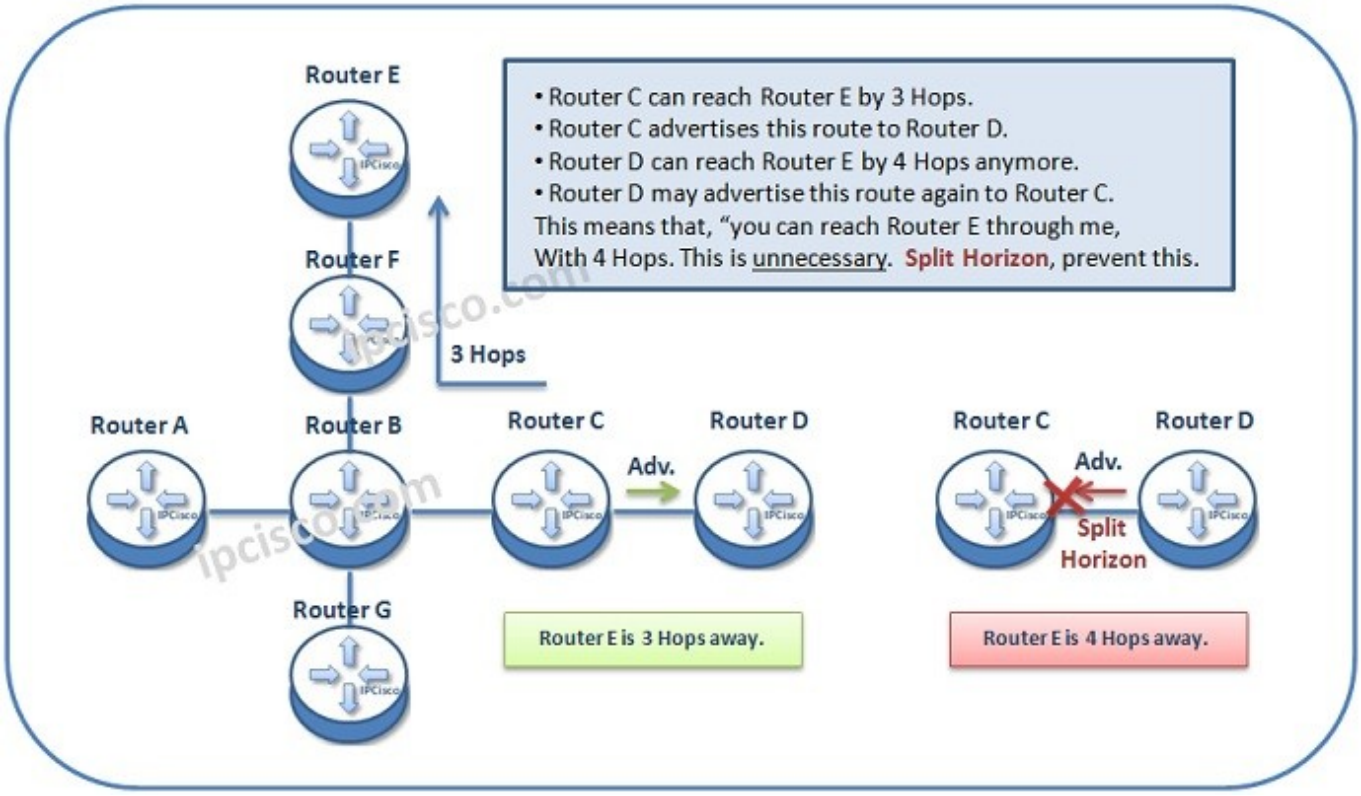


Split Horizon

Split Horizon is a routing protocol rule that prevents a router from advertising a route back to the interface from which it learned the route.
If a router advertises a route back to the source, it can create a routing loop where packets bounce endlessly between routers.
Split Horizon ensures that a router does not send redundant or misleading information about a route to its neighbor.

Split Horizon Example

Router A tells Router B it can reach Network X.
Router B learns about Network X via Router A.
Split Horizon prevents Router B from telling Router A, "I can reach Network X," because Router A already knows about Network X.



Comparison

Feature	Split Horizon	Poison Reverse
Mechanism	Prevents sending route updates back to the source	Advertises a route back as "unreachable"
Action Taken	Silently suppresses route advertisement	Actively advertises the route with an infinite metric
Effectiveness	Prevents simple routing loops	Handles more complex routing loops
Overhead	Lower, as updates are suppressed	Slightly higher, as additional updates are sent

Open Shortest Path First (OSPF)

Purpose: OSPF is a link-state protocol widely used in medium to large-sized enterprise networks. It provides fast convergence and is well-suited to complex network topologies.

Operation: OSPF calculates the best path based on the `Shortest Path First (SPF)` algorithm. Instead of counting hops, it considers a range of factors, such as bandwidth and delay, making OSPF routes more flexible and optimized for performance.

Area Division: OSPF supports `Hierarchical Routing` by dividing large networks into areas to optimize performance. Each router maintains a detailed map (link-state database) of its area but only summarizes information about other areas.

Updates: Rather than broadcasting its full routing table, OSPF only sends updates when there is a change in the network topology, improving efficienct and reducing network traffic.

Border Gateway Protocol (BGP)

Purpose: BGP is the primary protocol used on the internet to connect autonomous systems (AS), which are large networks or networks under a single administrative domain.

Operation: BGP is a path-vector protocol that considers a variety of factors (e.g, policy, path length, and route availability) to determine the best route between Autonomous Systems. BGP does not rely solely on shortest paths, but also incorporates policies that control data flow.

Updates: BGP only updates routes when there is a change, such as a link faliure or the discovery of a new path, reducing unnecessary traffic on the network.

Types of BGP

Internal BGP (iBGP): Used within an AS to handle routing among BGP routers within the same organization.

External BGP (eBGP): Used to route between different ASes on the internet.

Comparing the Three

Service	RIP	OSPF	BGP
Scale	Suitable for small networks	For larger enterprise networks	For internet-wide or multi-domain environments
Convergence	Slow convergence due to periodic updates and hop count limitation	Faster convergence since it uses link-state information and immediately floods updates across the network	Convergence time varies depending on the size of the network; it can take longer in global internet-scale networks
Routing Metrics	Uses hop count	Uses a cost based on link state	Uses a policy-driven path vector approach

Forwarding Tables, QoS, and ACLs

Forwarding Tables

A forwarding table is a data structure used in routers and switches to determine where to send incoming packets based on their destination. Similar to a routing table, however, it operates at the data plane level.

Example Structure:

Destination Prefix	Next Hop	Interface
192.168.1.0/24	192.168.1.1	eth0
10.0.0.0/8	1.0.0.2	eth1

QoS

A Quality of Service refers to mechanisms and policies designed to manage network resources and prioritize certain types of traffic to ensure performance and reliability for specific applications, users, or data flows.

Traffic Prioritization: High-priority traffic (such as VoIP) is given precedence over less critical traffic (such as email)

Metrics:
Bandwidth: Guarantees minimum data rates.
Latency: Ensures timely delivery of packets.
Jitter: control variation in packet delay.
Packet Loss: Minimizes dropped packets.

Techniques:
Traffic Shaping: Limit outgoing traffic to a predefined rate.
Policing: Drops packets that exceed certain thresholds.
Queue Managment: Organizes packets into priority-based queues.

ACLs

an Access Control List are a set of rules set by routers, switches or firewalls to control which packets are allowed or denied based on specific criteria such as IP addresses, protocols, or port numbers. They enhance network security and enforce policies.

Types of ACLs:
Filter traffic based on source IP addresses.
Simpler and used for basic access control.

Extended ACLs:
Filter traffic based on multiple criteria (source/destination IP, port numbers, porotocol types).
Provide more detailed control

Components of an ACL rule:
Permit/Deny: specifies whether to allow or block the traffic.
Source/Destination Address: Defines which addresses the rule applies to.
Protocol: Identifies the type of traffic (TCP, UDP, ICMP).
Port Numbers: Matches specific services (E.X HTTP on port 80).

Example
An ACL rule for a router might look like this:

```
permit tcp 192.168.1.0 0.0.0.255 any eq 80
deny ip any any
```

This allows HTTP traffic from 192.168.1.0/24 network while blocking all other traffic.

How they work together

Forwarding Tables: Handles basic packet forwarding decisions based on routing information.
QoS: Ensures that prioritized traffic receives the necessary resources for performance.
ACLs: Enforces access and security policies, deciding which packets can pass through the network.

Virtual Circuit and Datagram Networks

Virtual Circuit (VC) Networks

VC networks operate similarly to a telephone system, where a path (or circuit) is set up between the sender and receiver before any data is transmitted. This approach establishes a **logical connection** across multiple routers that remains consistent throughout the session.

Key Features

Connection Setup:
A connection establishment phase occurs before data transfer, involving signaling protocols to reserve resources and set up a unique path.
Eamples includel ATM (Asynchronous Transfer Mode) and Frame Delay.

State Information:

Routers maintain **state information** about active virtual circuits. For each active connection, routers store a mapping of incoming/outgoing interface identifiers and virtual circuit IDs.

Data Transmissions:

Data is transmitted in **packets**, but each packet includes a small virtual circuit identifier, instead of the destination IP.

Teardown:

After the session ends, the virtual circuit is torn down, releasing reserved resources.

Advantages:

Reliability: Path consistency ensures packets arrive in order.
Resource Reservation: Guaranteed bandwidth and QoS are easier to implement

Disadvantages:

Setup Overhead: Requires initial setup time, which adds latency for short-lived connections.
Scalability Issues: Maintaining state for each VC can strain resources in large networks.

Datagram Networks

Datagram Networks, such as the Internet, operate without establishing a pre-defined path. Each packet is treated independently, and routing decisions are made dynamically at each hop based on the packet's **Destination IP Address**.

Key-Features

No connection setup phase, data is sent immediately.
Each packet is independent and may follow a different path.

Routing Decisions:

Routers use routing tables to determine the next hop for each packet. These tables are updated dynamically with routing protocols like RIP, OSPF, or BGP.

No State Maintenance:

Routers do not store per-flow information, making the system highly scalable.

Advantages:

Allows for scalability, which allows networks to support a vast number of flows.
Allows for fault tolerance, packets can dynamically reroute and fail

Disadvantages:

Packets may arrive out of order, requiring reassembly at the receiver.
No built in Quality of Service, meaning it uses **Best-Effort Delivery** which means no guarantee on bandwidth or latency.

Examples of both

Virtual Circuit Example

ATM Networks: Before transmitting a video call, an ATM network establishes a virtual circuit. Each packet in the call carries a VC ID, ensuring it follows the same route, maintaining low latency and consistent delivery.

Datagram Example

The Internet: When you load a webpage, each HTTP request and response is broken into packets. These packets might traverse different paths through the network and are reassembled at the destination.

Hybrid Systems

MPLS (Multiprotocol Label Switching): it uses labels (Similar to VC IDs) for forwarding decisions while maintaining the scalability of datagram networks.

Routing Principles

Routing Algorithms

The purpose for routing algorithms is to determine the optimal path for packets to travel from source to a destination

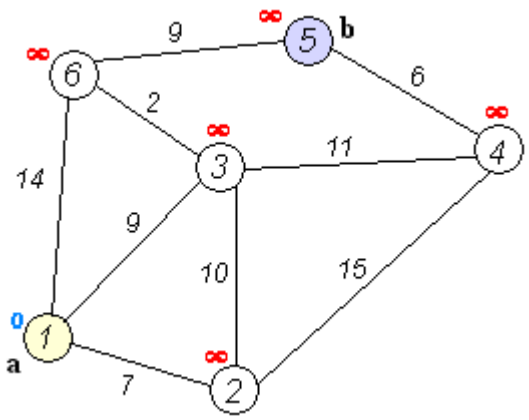
Types

Global Routing Algorithms:

Each router has a complete view of the network topology. An example would be Link-State routing which uses link-state routers to exchange messages and allow each router to learn the entire network topology. This allows algorithms such as Dijkstra's Algorithm to compute optimal routing routes. Although Link-State offers accuracy and complete topology awareness, it is quite expensive and slows down during changes / updates.

Global Routing Example

```
Path to router B: A → B (cost 2).
Path to router C: A → C (cost 3).
Path to router D: A → C → D (cost 5).
```

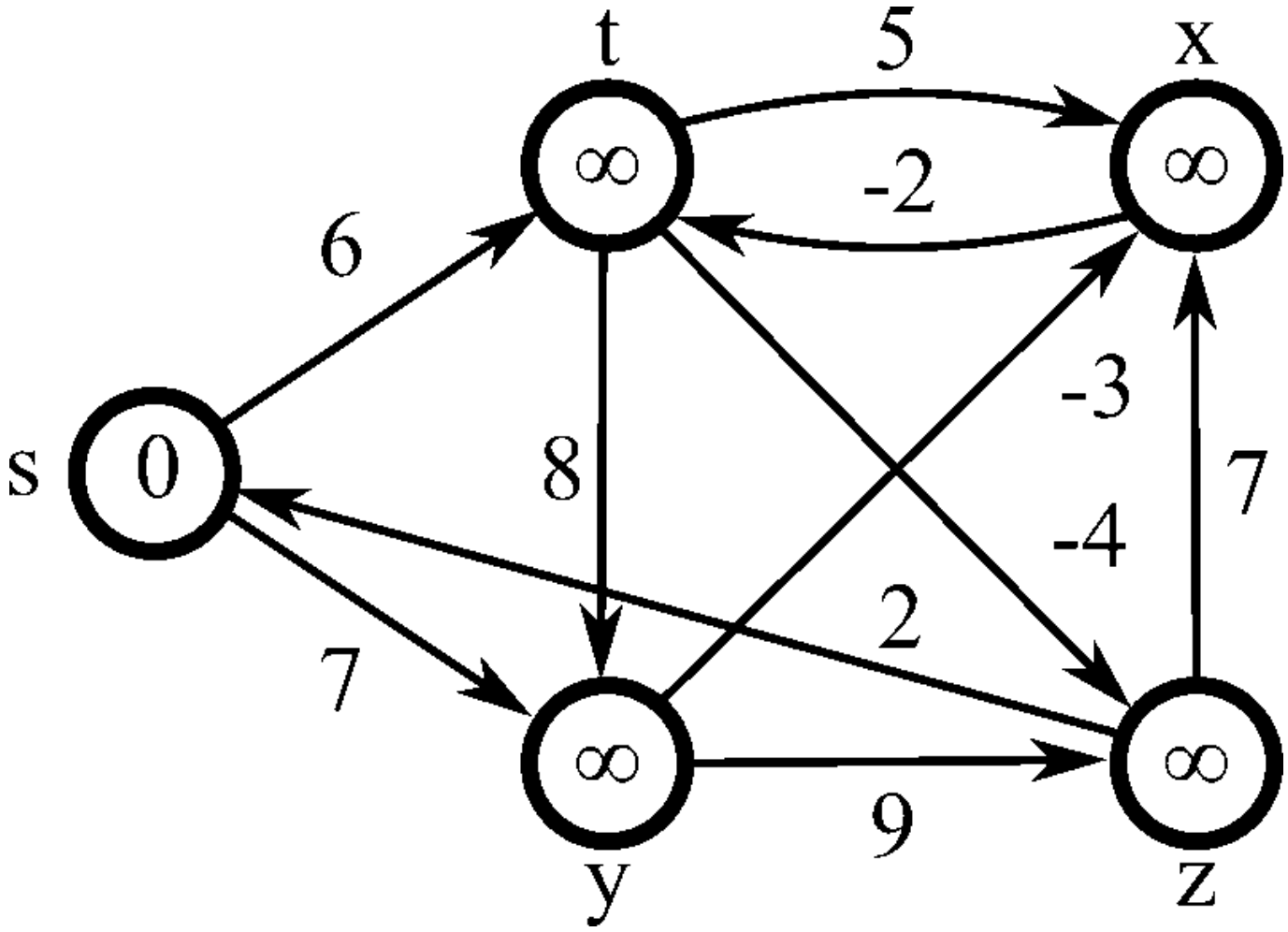


Decentralized Routing Alogrithms:

Compared to Global Routing Algorithms, Decentralized ones only allow routers to make decisions baded on information from their neighbors. Algorithms such as Distance-Vector (DV) Routing protocols use this. Each router maintains a table of know destinations and costs, and exchange their information with their neighbors periodically. Unlike Dijkstra's Algorithm, it must take account for **Negative Edges** which introduces the Bellman-Ford algorithm

Decentralized Example

Router A learns from its neighbor, router B, that the path to router D is 3 hops (or costs). Router A updates its routing table and advertises this to other routers.



Routing Metrics:

There is a specific criteria needed for selecting an optimal route. With programs such as RIP and BGP existing, these can cause delays in how the routing table picks an optimal route.

KEY CRITERIA

```
**Hop Count:** Number of intermediate devices (routers), the packet must go
through.
**Latency:** Time delay for data transmission.
**Bandwidth:** Capacity of the connection.
**Reliability:** Stability and fault tolerance of the route.
```

Shortest Path Routing

A primary goal for many routing algorithims is to minimize the total cost (time and distance) to deliver packets.

Routing Table Construction

Routing Table: Stores information about routes to different destinations. Populated dynamically through algorithms like link-state or distance-vendor.

Convergence

The state when all routers in the network have consisten and up-to-date routng information. Its importance is to ensue stability and accurate data delivery, however, slow convergence can lead to routing loops or packets lost in transit (black holes).

Discussed Routing Protocols

Link-State Protocols

Example: OSPF (Open Shortest Path First).
Routers distribute link-state advertisements (LSAs) containing link costs.
Each router computes shortest paths using the global view.

Distance-Vector Protocols

Example: RIP (Routing Information Protocol).
Routers periodically broadcast their routing tables to its neighbors.
Updates propogate until all router converge.

Hybrid Approaches

Cobines the elements of both link-state and distance-vector methods such as EIGRP (Enchanced Interior Gateway Routing Protocol)

Practical Applications

Intra-Domain Routing: Inside a single administrative domain (an organization), protocols like OSPF or RIP are used.
Inter-Domain Routing: Across multiple administrative domains, BGP facilitates routing decisions.

Bellman-Ford's Algorithm

The Bellman-Ford Algorithm computes the shortest path from a single source to all other 'edges' in a weighted graph. Unlike Dijkstra, Bellman-Ford includes negative weighted edges which ensures that the shortest path values are updated, allowing a source to find better and better paths throughout their journey.

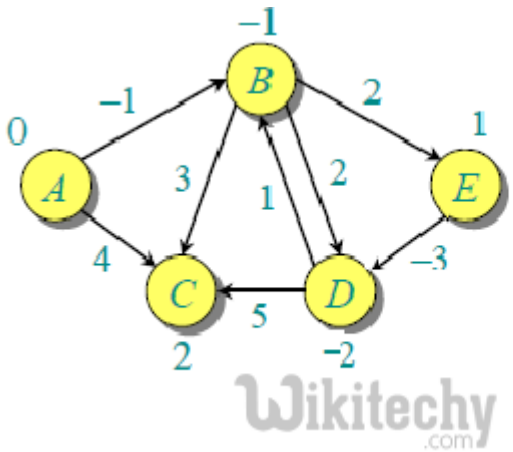
STEPS

```
Initialization: Set the distance to the source vertex as 0 and to all other
vertices as ∞.
Edge Relaxation: For (X) - 1. "Relaxing" an edge means trying to update a value
from its starting point to its end.
Negative Weight Cycle Check: After (X) is subtracted by 1, check to see if there
are any negative weights/negative loops. If there is a further reduce in distance,
a negative cycle exists.
```

Features

Handles Negative Weights: Unlike Dijkstra's algorithm, Bellman-Ford works with graphs containing negative weight edges.
Slower for Large Graphs: Bellman-Ford is one of the more slow SSSP's, meaning it has a higher time complexity than others such as Dijkstra.

Example Iteration



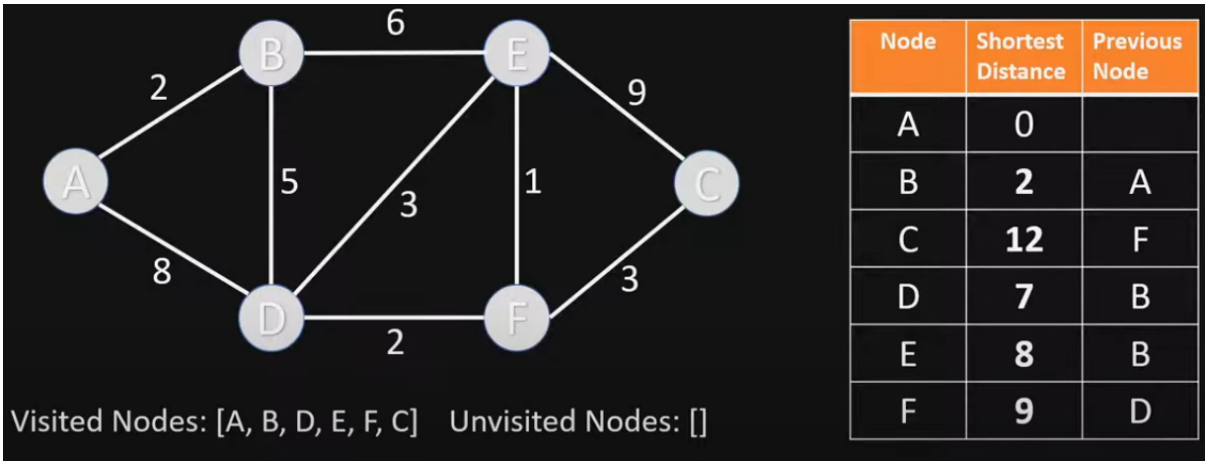
A	B	C	D	E
0	∞	∞	∞	∞
0	−1	∞	∞	∞
0	−1	4	∞	∞
0	−1	2	∞	∞
0	−1	2	∞	1
0	−1	2	1	1
0	−1	2	−2	1

As we can see from the picture above, Bellman-Ford introduces negative weights and multiple iterations. The iteration starts at letter A (weight 0) which moves to both letter B (weight -1) and C (weight 4), creating their weighted edges respectively. After letter A reaches all its edges, letter B then starts another iteration, with its starting weight of -1 moving towards letter C, giving it a weight of 2 (-1 + 3 = 2). Letter B then moves towards letter's E (weight 1) and D (weight 1), ending its iteration. After finding no changes, letter E finds a weight decrease for letter D and updates the table. After the final iteration, no more changes can be made.

Dijkstra's Algorithm

Dijkstra's Algorithm computes the shortest paths from a single source to all verticles in a non-negative weighted graph. It uses a greedy priority queue to select the closest unvisited vertex. Similar to Bellman-Ford, Dijkstra uses multiple iterations to ensure a short distance for the source. However, it does not include negative weights.

Efficient for Non-Negative Weights: Optimized for graphs with non-negative edge weights.
Fails with Negative Weights: Can produce incorrect results if the graph has negative weights.



Node	Shortest Distance	Previous Node
A	0	
B	**2**	A
C	**12**	F
D	**7**	B
E	**8**	B
F	**9**	D

Differences between the two

Feature	Bellman-Ford	Dijkstra
Purpose	Finds the shortest path from a source to all verticles, handles negative weights	Finds the shortest paths from a source to all verticles, no negative weights
Algorithm Type	Dynamic Programming and Edge Relaxation	Greedy Algorithm with Priority Queue (heap)
Negative Weights	Works with negative weights	Does not work with negative weights
Time Complexity	Slower and implements relaxation, no priority queue, and flexibility cost	Uses a "Greedy" approach, has a priority queue and is more efficient
Cycle Detection	Can detect negative weight cycles	cannot detect cycles
Priority Queue	Does not use a priority queue	Uses a priority queue for efficient vertex selection
Use Cases	Useful for graphs with negative weights and when cycle detection is needed	Best for graphs with non-negative weights and when speed is a priority
Real-World Use	Currency arbitrage, routing with negative costs	GPS Navigation, shortest paths in networks without penalties

Protocols inspired by Bellman-ford and Dijkstra

Thanks to the designs and efforts by Bellman-ford and Dijkstra, many routing tables and protocols, use these as a way to figure out the shortest and/or cheapest path for a source.

```
Routing Information Protocol (RIP)
Based on the Bellman-Ford algorithm
Uses a distance-vector approach
Simple and suitable for small networks
Coverges slower in larger or dynamically changing networks

Border Gateway Protocol (BGP)
Not directly related to either, BGP shares principles of path selection and
updates
It uses a path-vector routing, with path attributes rather than strictly shortest
path metrics

Enhanced Interior Gateway Routing Protocol (EIGRP)
Uses the Diffusing Update Algorithm (DUAL), an advanced variant of Bellman-Ford
Supports both Distance-Vector and Link-State characteristics
```

Enhanced Interior Gateway Routing Protocol (EIGRP)

Enhanced Interior Gateway Routing Protocol is an advanced hybrid protocol developed by Cisco. It combines the features of both link-state and distance-vector, this allows scalability, fast convergence and efficiency in

large networks.

Basic Terminology

Hello Messages: Used for neighbor discovery/recovery, if there is any device operating EIGRP.
NULL Update: Used to calculate SRTT (Smooth Round Trip Timer) and RTO (Retransmission Time Out).
SRTT: Time taken by a packet to reach the neighboring router and the ACK of the packet to reach the router.
RTO: Time for which the router waits for an ACK of a packet.
Full Update: Exchanging of messages after hello messages are exchanged or a relationship is formed.
Partial Update: Messages are exchanged when there is a topology change and new links are added, containing new routes which are multicasted to all neighbors.
Query Message: Multicast a message to neighbors when a device is declared "dead" and has no routes to it in its topology table.
Hello/Dead Timer: EIGRP sends a hello message which lasts for 5 seconds, a device will be declared dead if its neighbor does not send a hello message back within 15 seconds.

Key Features of EIGRP

Hybrid Protocl

EIGRP incorporates features of both distance-vector (best route based on distance) and link-state (graphing nodes into a routing table).
Like distance-vector, EIGRP shares its routing tables with its neighbors.
It calculates routes using a more complex algorithm (DUAL) for faster convergence.

DUAL Algorithm

Using the **Diffusing Update Algorithm (DUAL)** to ensure loop-free routing and rapid convergence.
DUAL allows routers to quickly adapt to network changes without requiring a full recalculation

Metric Calculation

EIGRP uses a composite metric based on five factors:

```
Bandwidth: Minimum bandwidth along the path.
Delay: Cumulative delay of links along the path.
Load: Network traffic on the path.
Reliability: Stability of the links.
Maximum Transmission Unit (MTU): not directly used in metric calculations, but
influences routing decisions.
```

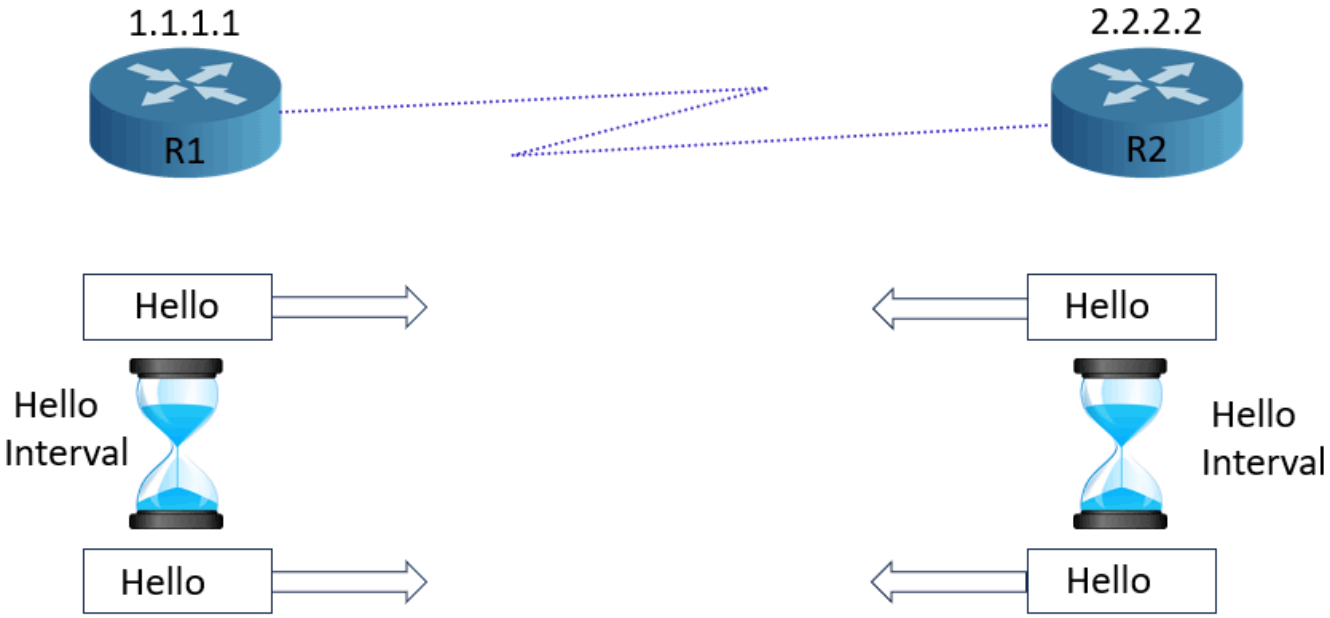
Example Metric

```
       Vector metric:
         Minimum bandwidth is 64 Kbit
         Total delay is 25000 microseconds
         Reliability is 255/255
```

```
          Load is 197/255
          Minimum MTU is 576
          Hop count is 2
```

Neighbor Discovery

Uses **Hello Packets** to discover and maintain neighbor relationships dynamically.
Neighbors are directly connected routers that speak EIGRP and are within the same autonomous systems



Partial Updates

Instead of sending full routing table updates, EIGRP sends **Incremental Updates** only when there are changes, reducing bandwidth usage

Protocol Independence

EIGRP supprots multiple network layer protocols including: IPv4, IPv6, AppleTalk, IPX (obsolete)

Scalability

Supports hierachical network designs using **Route Summarization (used to minimize the number of routing tables)** and **Stub Routing (network with no knowledge of other networks)** to reduce the size of routing tables and limit update propagation

Advantages of EIGRP

Fast Cibvergence: Thanks to DUAL, EIGRP can quickly react to network changes.
Efficient Resource Usage: Partial updates and neighbor discovery minimize bandwidth usage.
Loop Prevention: DUAL ensure loop-free routing paths.
Support for Variable-Length Subnet Masking (VLSM): Allows for efficient IP address allocation.
Easy Configuration: Relatively simple to configure compared to OSPF and BGP.

How EIGRP Works

Neighbor Establishment

Routers send Hello packets to discover neighbors.
If two routers agree on parameters (AS numbers, K values), they form a neighbor relationship.

Topology Exchange

Routers exchange routing information using **update** packets.
Updates are sent to all directly connected neighbors.

Route Selection

DUAL computes the best path based on the composite metric.
Two key paths are maintained:
Successor: Primary path to the destination.
Feasible Successor: Backup path, ensuring rapid failover.

Routing Table Updates

Only changes in topology trigger partial updates, minimizing resource usage.

Disadvantages of EIGRP

Vendor Specific: Initially, EIGRP was proprietary to Cisco, limiting interoperability. However, it was released as an open standard in 2013.
Complex Metrics: The composite metric can be hard to understand and troubleshoot compared to similar metrics like OSPF
Resource Intensity: Requires more processing and memory than simpler protocols like RIP.

Comparison to others

Feature	EIGRP	OSPF	RIP
Protocol Type	Hybrid (Distance-Vector and Link-State)	Link-State	Distance-Vector
Convergence Speed	Fast	Moderate	Slow
Scalability	High	High	Low
Complexity	Moderate	High	Low
Loop Prevention	Yes (DUAL)	Yes	No (Split Horizon/Poison Reverse)
Metric	Composite	Cost (Bandwidth)	Hop Count

What's inside a router?

As discussed in the previous sections, the router's primary function is to forward packets from an input interface to the correct output interface.
The router uses the Data Plane and Control Plane to manage the forwarding and network decisions

Data Plane

The data plane is responsible for **forwarding data packets** based on the decisions made by the control plane. It handles the actual movement of data through the network.

```
Functionality:

Operates at the router/switch's hardware or software level  .
Matches incoming packets to routing table entries and sends them to the
appropriate output interface.
Responsible for processing packets at line speed (the speed at which the physical
link operates).

Operations:

Forwarding: moving packets to the next hop.
Dropping: Discarding packets if there are no valid routes or if they violate
policies.
Modification: Adjusting packet headers (e.g decrementing the TTL field).

Examples:
A router forwarding an IP packet to the next hop using its forwarding table.
A switch forwarding Ethernet frames based on MAC address table lookups.
```

Control Plane

The control Plane is responsible for routing decisions and network topology management. It creates and updates the data used by the data plane.

```


Functionality:

Runs routing protocols like OSPF, BGP, RIP to determine the best paths for data.
Updates the router's routing table and populates the forwarding table used by the
data plane.
Handles signaling and network configuration.

Operations:

Topology Discovery: Learning about the network's structure through protocols.
Routing Decisions: Computing paths based on metrics like distance, cost, or
policy.
Policy Managment: Enforcing Quality of Service (QoS), Access Control Lists (ACLs),
```

```
etc.

Examples:
BGP calculating the best route to an external network and updating the routing
table.
OSPF discovering link failures and recalculating paths.
```

Key Differences

Aspect	Data Plane	Control Plane
Primary Role	Forwarding packets	Making routing decisions
Speed	Operates at line speed	Operates slower (decision-making takes time)
Scope	Per-packet operation	Network-wide or link-level computation
Protocols	No direct protocols; uses forwarding tables	Routing protocols (BGP, RIP, OSPF)
Implementation	Hardware-based (ASICs) or software	Usually software-based
State Maintenance	Stateless (no memory of previous packets)	Stateful (maintains routing tables, network state)

Real-World Example

In a router:

Control Plane: OSPF calculates the best path to a destination and updates the routing table.
Data Plane: When a packet arrives, the router looks up the forwarding table (populated by OSPF) and forwards the packet to the appropriate interface.

Emerging Technologies and Seperation

Software-Defined Networking (SDN): Seperates the control plane from the data plane entirely. The control plane operates on centralized controllers, while switches/routers handle only data forwarding, enabling more dynamic and programmable network management.

Key components of a Router

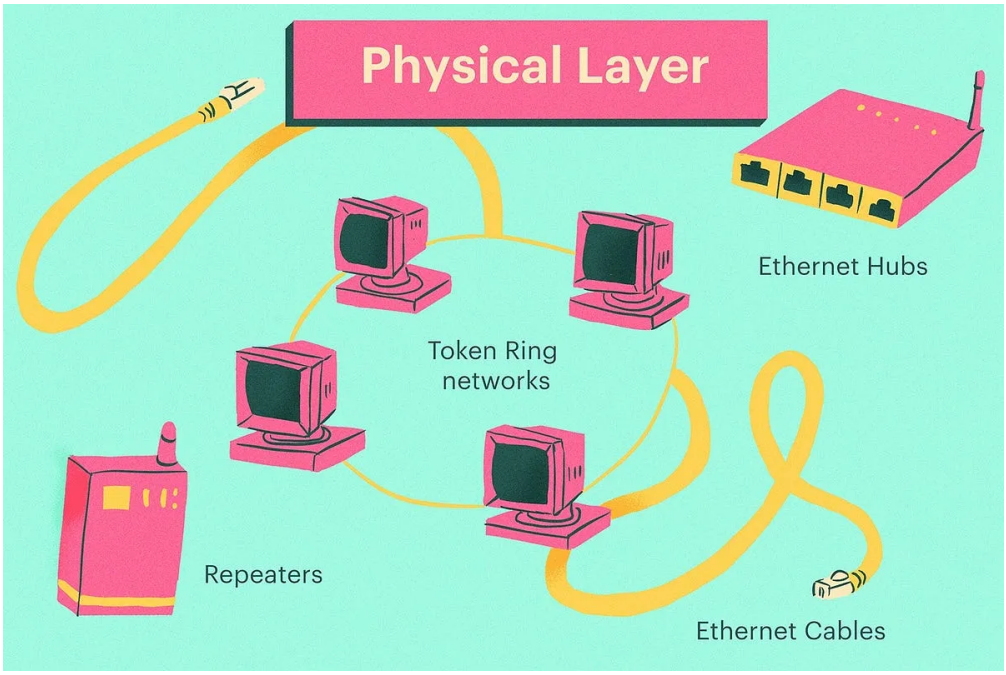
The router's primary goal is how inputs and outputs packets, while maintaining integrity and proficiency

Input Ports

Purpose: Receives incoming packets and performs preliminary processing.
Funtions: Physical Layer Functions (Signal Conversion), Data Link Layer Processing (Error checking and frame extraction), Forwarding table lookup to determine the correct output port.

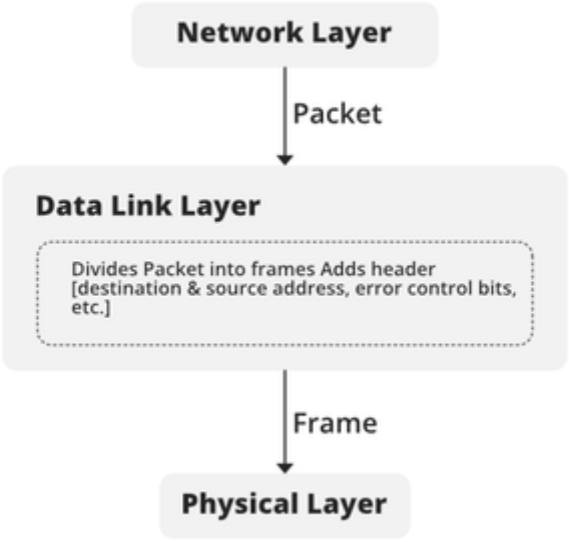
Physical Layer Functions

Purpose: Handles the physical reception of signals from the incoming communication link.
Tasks: Convert electrical (variations of voltages or currents used to represent data) or optical signals (light used to represent data) into digital data.
Recover clock signals and synchronize with the incoming bit stream.
Detect and correct physical layer errors.
Example: A router receives optical signals from a fiber-optic link and converts them into a digital bit stream for processing.



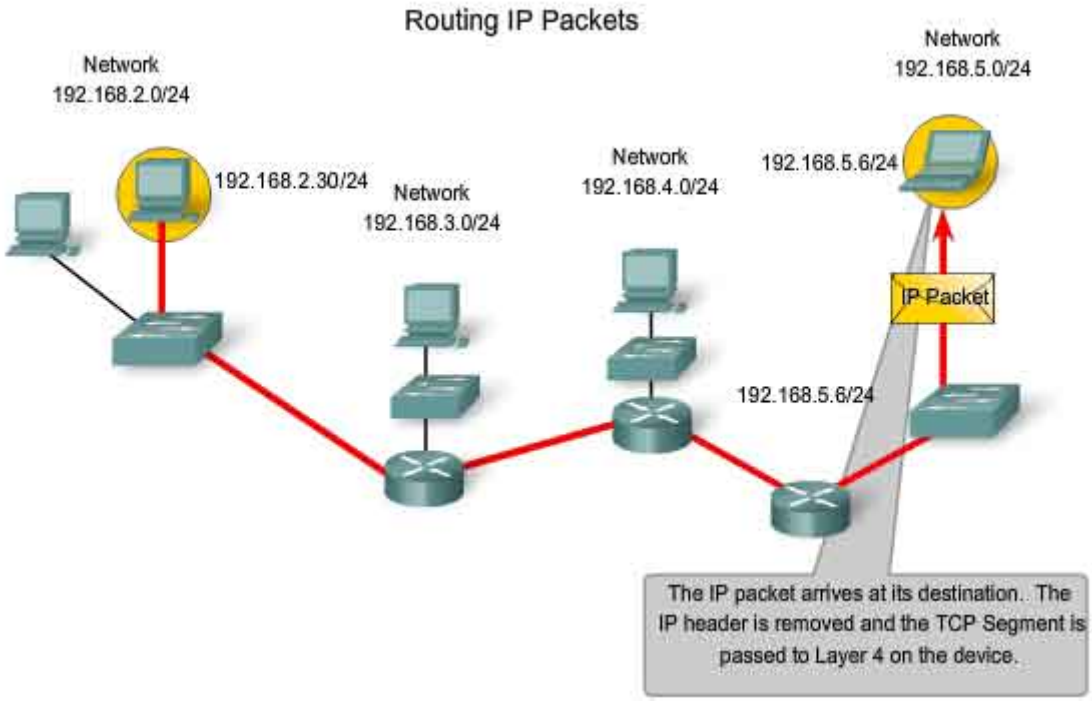
Data Link Layer Functions

Purpose: Processes incoming frames and prepares the payload (packet) for forwarding.
Tasks: Error detection and correction using frame checksums (CRC).
Frame demarcation: Identifying the start and end of a frame.
Removing the data link layer header and trailer.
Handling protocols like Ethernet, WiFi, or other link-layer standards.
Examples: When receiving Ethernet frames, the input port extracts the encapsulated IP packet by removing the Ethernet header and trailer.



Network Layer Processing

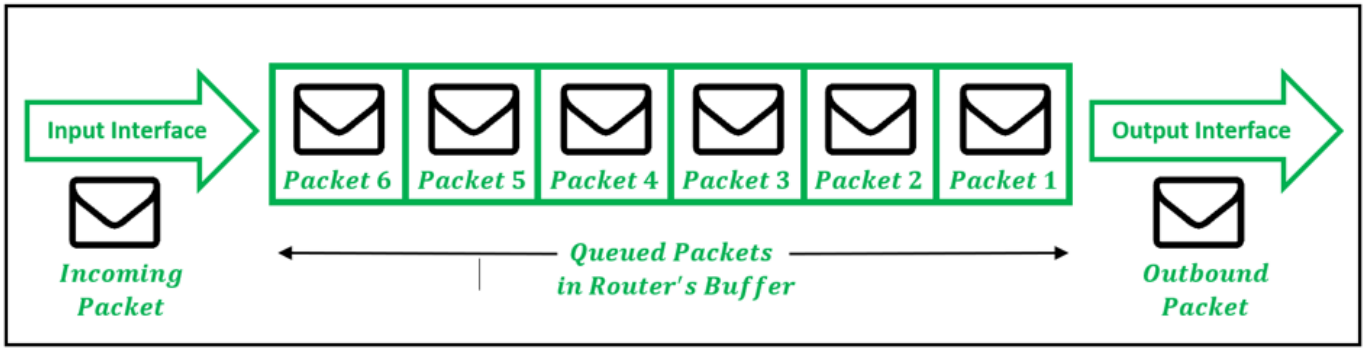
Purpose: Prepares packets for forwarding by determining their next hop and route.
Tasks: The Network Layer is reponsible for the Forwarding Table, which is used to determine the output port for a packet, lookup methods and convergence.
It also ensures packet's integrity and validity and discards corrupter or improperly formatted packets.
Example: For a packet with destination IP `192.168.1.10`, the input port checks the routing table and determines that the packet should be sent to output port 3.



Queue Management

Purpose: Temporarily stores packets in input queues if the switching fabric is busy or congested.
Tasks: Implementing FIFO (First In, First Out) or priority-based queuing for incoming packets.
Resolve contention when multiple packets need access to the switching fabric simultaneously.

Challenges: Head-Of-Line (HOL) Blocking: Occurs when a packet at the front of the input queue blocks others from accessing the switching fabric, even if their destinations are free.
Examples: If a burst of traffic arrives at multiple input ports destined for the same output port, some packets may be queued, while others are trasmitted.



Packet Classification and Processing

Purpose: Identify packets that require special handling (QoS, Firewall Rules).
Tasks: Apply Access Control Lists (ACLs) for security (blocking packets based on ip source/destination or port). Perform deep packet inspection if required.
Example: A packet with a flagged IP address is dropped by the input port.

Special Packet Handling

Purpose: Identify and handle control-plane traffic or exceptional packets.
Tasks: Process packets meant for the router itself (routing protocol messages like RIP or OSPF).
Discard malformed or unauthorized packets.
Example: A router processes OSPF 'Hello Messages' received on its input port to maintain its routing protocol neighbor relationships.

Workflow of Packet Processing in Input Ports

Signal Reception

The router receives a signal on the physical link, converts it to digital data, and extracts the frame.

Frame Decapsulation

The Ethernet or other data link header is removed, leaving the IP packet.

Forwarding Decision

The destination IP address in the packet is used to consult the forwarding table.
The appropriate output port is identified.

Switching Fabric Access

The packet is transferred to the switching fabric, queued if necessary, or flagged for special handling

Hardware and Software in Input Ports

Hardware Accelerations

Input ports use specialized hardware for high-speed processing, such as:
Network Interface Cards (NICs): Handle data link layer functions.
TCAMs: Enables fast forwarding table lookups.
Example: In high-speed routers, hardware performs packet classification and forwarding in microseconds.

Software Functions

For control-plane traffic or less time-critical tasks, the router's software processes packets.
Example: A software-based firewall rule examines incoming packets for malicious patterns.

Real World Examples of Input Ports

Enterprise Routers

Input ports handle traffic from multiple VLANs, requiring accurate packet classification and queuing.
An example would be VoIP packets might be prioritized over web browsing traffic.

ISPs

At an ISP edge router, input ports process large volumes of packets from customers. These packets undergo queuing, validation, and forwarding to handle high-speed demands.

Switching Fabric

Purpose: The switching fabric ensures high-speed data transfer between a router's input and output ports.
It ensures that packets are routed or switched to the correct destination efficiently and without loss.

Key Characteristics: Operates at very high speeds to keep up with modern networking demands.
Often implemented using specialized hardware for speed and scalability.
Supports simultaneous data transfers between multiple input-output port pairs.

Components of Switching Fabric

Input Ports: Process incoming packets and prepare them for transfer through the switching fabric.
Output Ports: Receive packets from the switching fabric and transmit them toward their next-hop destination.
Switching Mechanism: Determines how packets are transferred between input and output ports.
Considers the destination address, priority queuing, and congestion status.

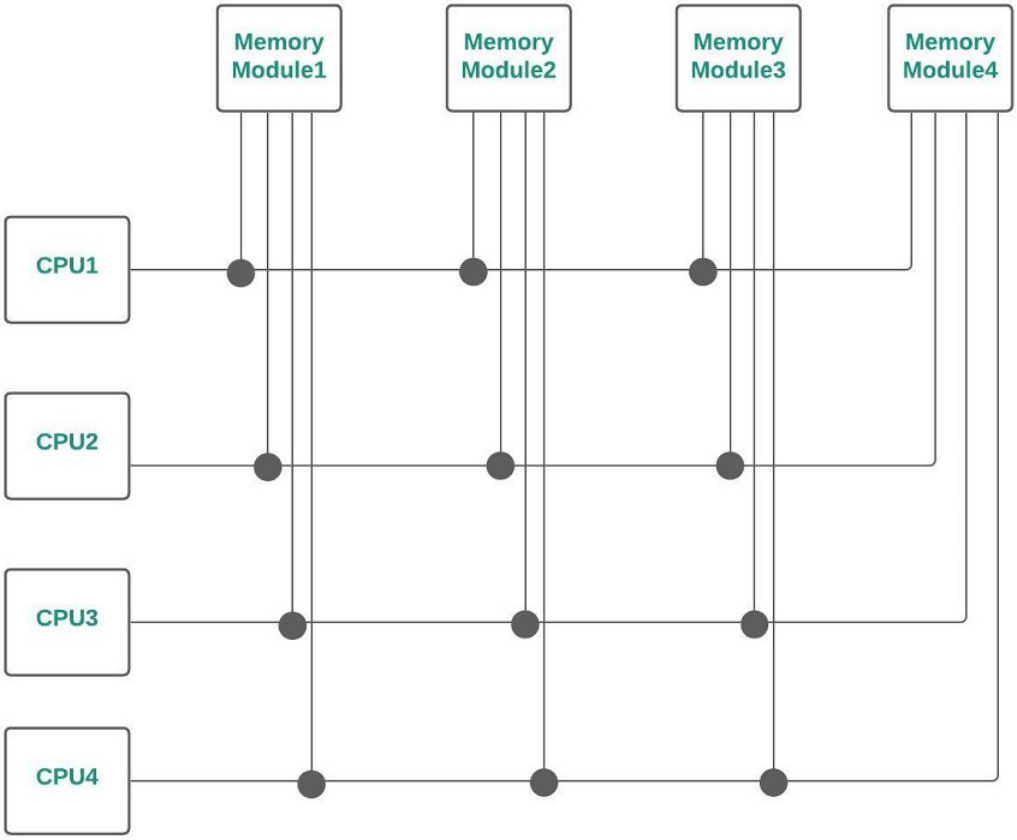
Types of Switching Fabrics

Memory-Based Switching

How It Works: The router's CPU is reponsible for transferring packets from input ports to output ports via system memory.
Incoming packets are stored in the memory temporarily before being forwarded.

Performance: Limited by the memory access speed and CPU processing power.
Suitable for low-speed routers.

Example: Early routers used memory-based switching, where the CPU processed each packet individually.

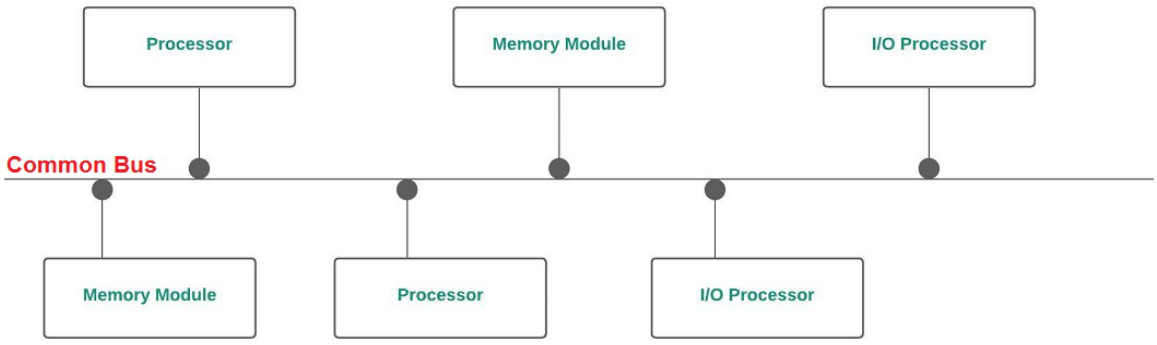


Bus-Based Switching

How It Works: All input and out ports are connected to a shared bus.
Packets are transferred from input ports to output ports using the bus.

Performance: Limited by the speed of the shared bus.
Only one packet transfer can occur at a time, causing potential bottlenecks.

Example: Small to medium-sized routers or switches may use a bus-based fabric.

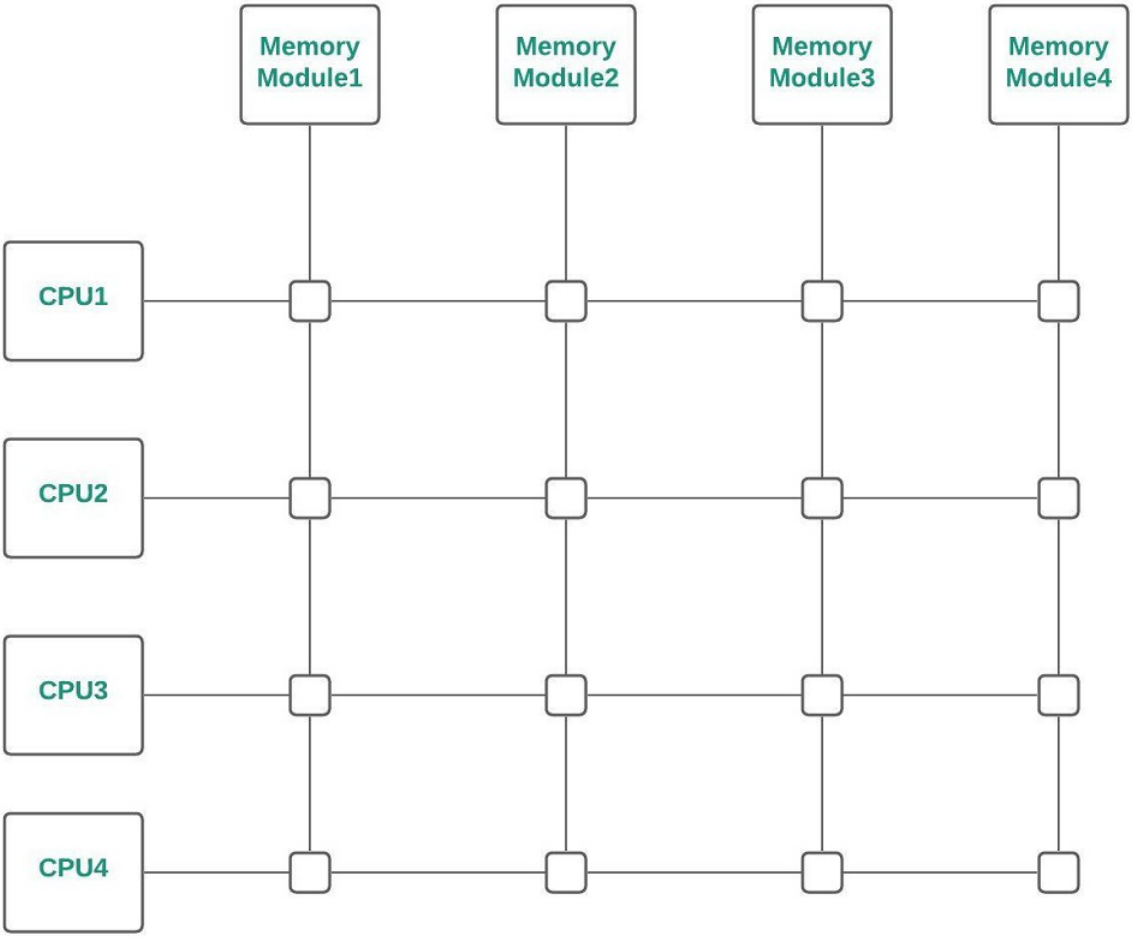


Crossbar Switching

How It Works: Uses a grid-like matrix where each input port can be connected to any output port via independent paths.
Simultaneous packet transfers are supported if no two packets are destined for the same output port.

Performance: Highly scalable and supports high-speed transfers.
Used in high-performance routers and switches.

Example: Data center switches with a large amount of ports often use crossbar switching for efficiency.



Advanced Switching Fabrics

Multistage Switches: Uses multiple interconnected crossbar switches to reduce hardware complexity and increase scalability.
Example: Clos Network, widely used in modern data center networks.

Asynchronous Transfer Mode (ATM) Switching: Designed for cell-based networks, this method handles fixed-size packets and is used in specialized environments.

Challenges of Switching Fabrics

Contention: When multiple packets at different input ports are destined to the same output port, the packets interfere with each other and may need to be queued or dropped.

Head-Of-Line Blocking: A packet at the front of an input queue blocks other packets in the queue, even if those packets could be forwarded to free output ports.

Scaling: As the number of ports increase, the compelxity and cost of the switching fabric also increases.

Latency: The switching fabric must operate with minimal delay to meet high-speed networking demands.

Real-World Examples

Enterprise Routers: High-performance routers in corporate networks often use crossbar or multistage switching gabrics to handle large volumes of traffic.

Data Center Switches: Top-of-rack switches use advanced fabrics to connect thousands of servers with minimal latency.

Carrier-Grade Equipment: ISPs rely on sophisticated switching fabrics to route traffic at speeds exceeding hundereds of gigabits per second.

Switching Fabric Example

Suppose a router has 4 input ports (A,B,C,D) and 4 output ports (1,2,3,4).
Using a crossbar fabric: Input Port A sends a packet to Output Port 3. At the same time, Input Port B sends a packet to Output Port 4. The crossbar fabric allows independent paths for both transfers, allowing them to occur at the same time.

Output Ports

The purpose of Output Ports in routers and switches is to forward packets from the switching fabric to their designated destination.

Packet Buffering (Queue Management)

Purpose: Stores packets temporarily before they are transmitted to the outbound link.

Why It's Needed: If the output link is busy or congested, incoming packets from the switching fabric must be queued to prevent packet loss.

Buffer Managment Techniques: FIFO (First In, First Out): packets are sent in the order they arrive in. **Priority Queuing:** Higher-priority packets are transmitted before lower-priority ones. **Weighted Fair Queuing (WFQ):** Packets are scheduled based on their weight or assigned priority levels.

Examples: In a VoIP call, voice packets might be prioritized over bulk file transfer packets to minimize latency and jitter (Priority Queuing).

Packet Scheduling

Purpose: Determine the order in which packets from the buffer are transmitted over the output link.

Scheduling Algorithms: Round-Robin: Cycles through each queue sequence. **Priority Scheduling:** Always sends packets from the highest-priority queue first. **Weighted Round Robin (WRR):** Similar to Round-Robin but allows different queues to have different amounts of bandwidth.

Example: A router serving both streaming video traffic and bulk file downloads might use WRR to ensure video packets are prioritized without completely starving file transfer traffic.

Link-Level Protocol Handling

Purpose: Ensure the packet is correctly formatted for transmission over the specific type of link (Ethernet, Fiber Optic, DSL).

Tasks: Adding or verifying **Frame Headers**. Ensuring **Link-Layer error detection and correction** mechanisms are applied (using Cyclic Redundancy Check (CRC)). Converting packets into appropriate **electrical or optical signals** for physical layer transmission.

Cyclic Redundancy Check: An error detection code used to detect accidental changes to raw data during transmission or storage.

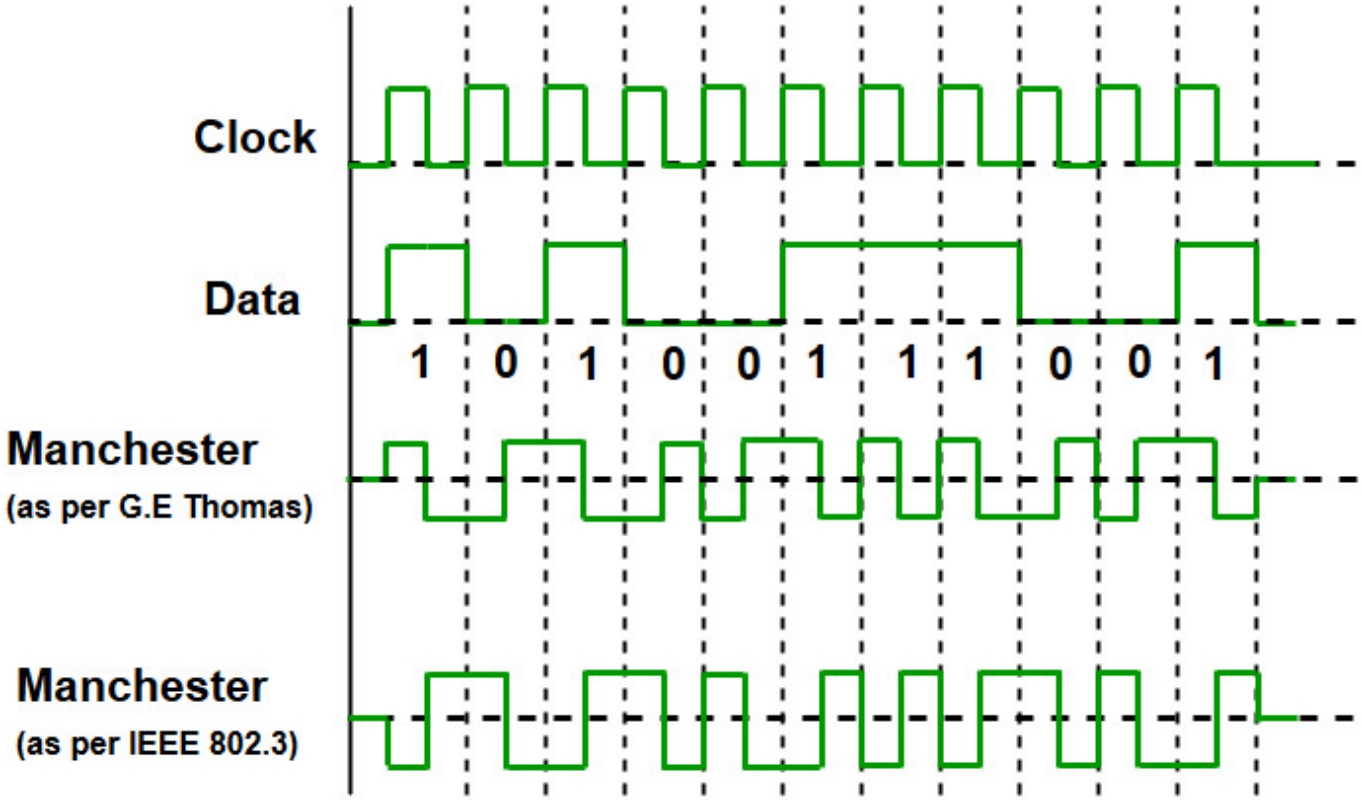
Example: On an Ethernet interface, the output adds Ethernet headers and trailers, including the MAC address of the next hop

Physical Layer Transmission

Purpose: Transmit the final signal (optical or electrical) over the physical medium

Tasks: Encoding the packet for physical transmission (Manchester Encoding for Ethernet). Controlling the signal strength and quality to ensure intergrity over distance. Managing **Error Detection** at the physical level to avoid signal corruption.

Manchester Encoding: used to ensure synchronization between the sender and receiver using clock signals while transmitting data over a network.



Example: For fiber optic links, the output port will convert data into optical signals using lasers or LEDs.

Congestion Management and Avoidance

Purpose: Prevent congestion on the outgoing link to reduce packet loss and latency.

Techniques: Random Early Detection (RED) is used to drop packets probabilistically to signal congestion before it becomes severe. **Explicit Congestion Notification (ENC)** is used to mark packets, informing the sender about congestion without dropping them. Lastly, **Traffic Shaping** is used to limit the rate at which packets are sent.

Example: If a video streaming application generates excessive traffic, traffic shaping might throttle its bandwidth to prevent the link from overwhelming

Packet Forwarding and Address Resolution

Purpose: Forward packets to the correct destination using the appropriate forwarding information.

Tasks: Consulting the **Forwarding Table** to determine the next-hop adress. Resolving addresses if necessary (ARP for Ethernet). Ensuring packets are sent to the correct outgoing interface.

Example: An IPv4 packet may need its destination IP address mapped to a MAC address before being forwarded over an Ethernet link.

Challenges of Output Port Design

Output Port Contention: Multiple packets destined for the same output port can overwhelm the buffer.
Head-Of-Line (HOL) Blocking: Packets at the front of the queue block subsequent packets from being transmitted.
Buffer Overflow: If the output port buffer fills up, packets may be dropped.
Latency: Packet queuing and scheduling can introduce delays, affecting real-time applications.

Real World Example

Imagine a router with 3 input ports, and only 1 output port.
Input Port 1: High-priority VoIP traffic.
Input Port 2: Streaming video traffic.
Input Port 3: Bulk file download traffic.

Scenario:

The output port queues the packets from all three inputs.
VoIP traffic is prioritized in the queue using a priority scheduling algorithm.
Video traffic follows, managed via Weighted Fair Queuing (WFQ).
The output port transmits the packets via an Ethernet interface, ensuring proper link-layer encapsulation and physical transmission.

Queuing Mechanisms in Depth

In networking, queuing mechanisms manage how packets are buffered, prioritized and transmitted through a router's output port when there is congestion.

Queuing Key Functions

Buffering Packets

Incoming packets are stored temporarily in the router's ourput queue buffer if the outbound link is busy.
The size of the buffer determines how many packets can be stored before overflow occurs.
Example: If a router is handling a burst of traffic, packets will be queued until the outbound link becomes free.

Packet Scheduling

Queuing mechanisms use **scheduling algorithms** to decide the order of packet transmission.
Different algorithms prioritize packets based on their importance, type, or application requirements.

Traffic Prioritization

Critical traffic (like voIP) is given priority over less time-sensitive traffic like file downloads.

Congestion Management

Queuing mechanisms manage packet flow to prevent or control congestion.
When queues are full, packets may be dropped or rerouted to alleviate congestion

Bandwidth Distribution

Certain queuing algorithms ensure **fair bandwidth distribution** to prevent a single application/user from controlling the link.

Common Queuing Mechanisms

First In, First Out (FIFO): Packets are transmitted as they arrive, simple and easy to implement.
Priority Queuing (PQ): Packets are assigned different priority levels from high to low. Real-time traffic get priority, lower priority packets may suffer.
Weighted Fair Queuing (WFQ): Allocates bandwidth proportionally to each flow based on weights assigned to packet queues.
Round Robin (RR): Packets from multiple queues are processed in a rotating order.
Random Early Detection (RED): Proactively drops packets randomly when queue thresholds are reached.

Real-World Example

Imagine a router managing traffic from a corporate network:

Priority Queuing: Real-time VoIP calls are sent immediately.
Weighted Fair Queuing: Different departments get a proportional share of the bandwidth.
FIFO: Backup traffic might follow a simple first-come-first-served rule.

Role of QoS

QoS Metrics: Delay, jitter, packet loss, and throughput.
Proper queuing ensures that crtical traffic meets specific QoS standards.
For example, in VoIP, queuing minimizes delay and jitter.

Challenges with queuing

Bufferbloat: Excessive buffering can increase latency.
Starvation: Lower-priority packets may never be transmitted.
Complexity: Advanced queuing algorithms require more processing power.

Internet Protocol (IP)

The IP protocol is the core protocol that enables communication across networks.

IP Datagram Structure

An IP datagram is the basic unit of data transfer in the IP layer. It consists of **header fields** which include the source, destination, TTL and checksums.

```
KEY FIELDS IN THE IP HEADER

Version: IPv4 or IPv6.
Header Length: Size of the header.
TTL (Time-To-Live): Limits the datagrams lifespan to prevent infinite looping.
Protocol: Indicates the protocol of the payload (TCP = 6, UDP = 17).
Source/Destination IP: Identifies the sender and receiver.
```

Datagram Forwarding

Datagram Forwarding focuses on where the next's IP hop should be according to the Forwarding Table

```
The router examines the Destination IP.
It looks up the Forwarding Table.
It determines the next-hop router or outbound interface.
It decrements the TTL.
If the TTL reaches 0, it is discarded with an ICMP message sent to the sender.
```

Example: A datagram destined for `192.168.1.10` might match an entry in the forwarding table:
`192.168.1.0/24 → Forward via Interface 1`
Routers use the **Longest Prefix Matching Rule** to choose the most specific route.

IP Addressing

Every device connected to the internet needs a **unique IP Address**, but there are some things to consider, which are IPv4, Subnetting and IPv6.

IPv4 Addressing

32-bit addresses (ex: `192.168.1.1`)
Divided into **Network ID** and **Host ID**
Address classes (A,B,C,D,E) are historical but still discussed

```
Class A: 1.0.0.0 to 127.255.255.255
Class B: 128.0.0.0 to 191.255.255.255
Class C: 192.0.0.0 to 223.255.255.255
Class D: 224.0.0.0 to 239.255.255.255
Class E: 240.0.0.0 to 255.255.255.255
```

Subnetting

Subnetting divides a large network into smaller sub-networks.
Example: `192.168.1.0/24`

IPv6 Adressing

128-bit addresses (ex: `2001:0db8:85a3:0000:0000:8a2e:0370:7334`)

IP Fragmentation and Reassembly

Different networks can handle different amounts of packet size or Maximum Transmission Units (MTUs).
The IP Protocol uses **fragmentation** to break apart packets too large for the network's MTU.
It also **reassebles** the packets at the final destination and not at intermediate routers.

```
EXAMPLE

Original Size: 4000 bytes
MTU: 1500 bytes
The datagram is fragmented into 3 pieces: 1500, 1500, and 1000 bytes

Each Fragment also carries

Identification Number
Fragment Offset
More Fragments (MF) Flag
```

Network Address Translation (NAT)

A NAT allows multiple devices in a private network to share a single public IP address.
Mostly common in small networks (such as home routers) to conserve the amount of public IPs.
Example: Devices on `192.168.1.x` share the public IP `203.0.133.5`.

Dynamic Host Configuration Protocol (DHCP)

The goal of DHCP is for the host to *dynamically* obtain IP addresses from the network server once it "joins" the network.
Much more efficient as there is not need to manually input IP addresses.

```
ADVANTAGES
```

```
Conserves IPv4 addresses.
Adds a layer of security (internal devices are hidden)

DISADVANTAGES

Breaks end-to-end connectivity.
Some applications (ex: voIP) may face issues.
```

Address Resolution Protocol (ARP)

ARP maps an IP address to a MAC address.
Example: A router knows an IP address (`192.168.1.10`) but needs the MAC address to forward a frame on the Ethernet link.

ARP Process: Host sends an **ARP Request Broadcast** (`Who has 192.168.1.10?`)
The host with that IP responds with its **MAC Address**

IPv4 vs IPv6

Feature	IPv4	IPv6
Address Size	32 Bits	128 Bits
Header Size	20 Bytes	40 Bytes
Address Format	Decimal	Hexadecimal
NAT	Common	Not Needed
Security	Optional	Built-in (IPSec)

```
IPv6 Benefits

Larger address space
Better support for QoS
Integrated security with IPSec
```

The Routing Problem

The main objective for routers is to create the best path for packets while minimizing cost, maximizing throughput and ensuring reliability.

For a more detailed example of what will be discussed click here

Link-State Routing

Link-State routing is a global routing algorithm, but what does this mean? It means every router had complete knowledge of the network topology.
They gain this information by sending and collection information from other routers (kind of like P2P).

Steps of Link-State

Each router discovers its neighbors.
Routers share information using Link-State Advertisements (LSAs).
Each router builts their own network map using the LSAs.
Using Dijkstra's Algorithm they compute the shortest path.
The results are then forwarded to the forwarding table.

Distance-Vector Routing

Distance-Vector Routing is a **decentralized routing algorithm** meaning each router know the cost (hops) of its **directly connected neighbors** and routers share their routing table with their neighbors periodically.
These types of routers use Bellman-Ford Algorithm to update their routing tables.

Steps for Distance-Vector

Each router maintains 3 things:
Destination Network
Cost (distance)
Next-hop router
Routers exchange their tables with their neighbors (from time to time)
which makes routers update their tables using a specific equation.

```
D(x,y)=min [c(x,v)+D(v,y)]
```

What does this mean?
D(x,y): Cost from router (x) to destination (y).
c(x,v): cost from router (x) to its neighbor (v).
D(v,y): Cost from neighbor (v) to destination (y).

According to this, router (x) learns the path to destination (y) through neighbor (v) at a cost of (x)

Although distance-vector may seem like a good protocol, it has a big disadvantage; **Count-to-Infinity Problem:** This problem occurs when a link fails, and router may loop many times trying to find a valid path.
Some solutions to this are Split Horizon and Poison Reverse

Hierarchical Routing

As networks grow, and have to maintain a complete routing table, it seems impractical to update it constantly just to link new routers into the table. The solution? **Autonomous Systmes!**

An Autonomous System (AS) operates under s **single administrative authority**. The Interior Gateway Protocol handles routing within an AS (such as OSPF, or RIP.) While Exterior Gateway Protocols handle routing between ASes (such as BGP).

The benefits of this include: reducing the routing table size, limits the scope of routing updates and improves scalability.

Routing Algorithm Compaison

Feature	Link-State (Dijkstra)	Distance-Vector (Bellman-Ford)
Routing Type	Global	Decentralized
Update Method	LSAs (flooding)	Periodic Table Exchanges
Algorithm	Dijkstra	Bellman-Ford
Convergence	Faster	Slower
Scalability	Poort in large nets	Better with hierarchical routing
Loop-Free	Yes	Potential Loops (Count-to-Infinity)