Computer Networks

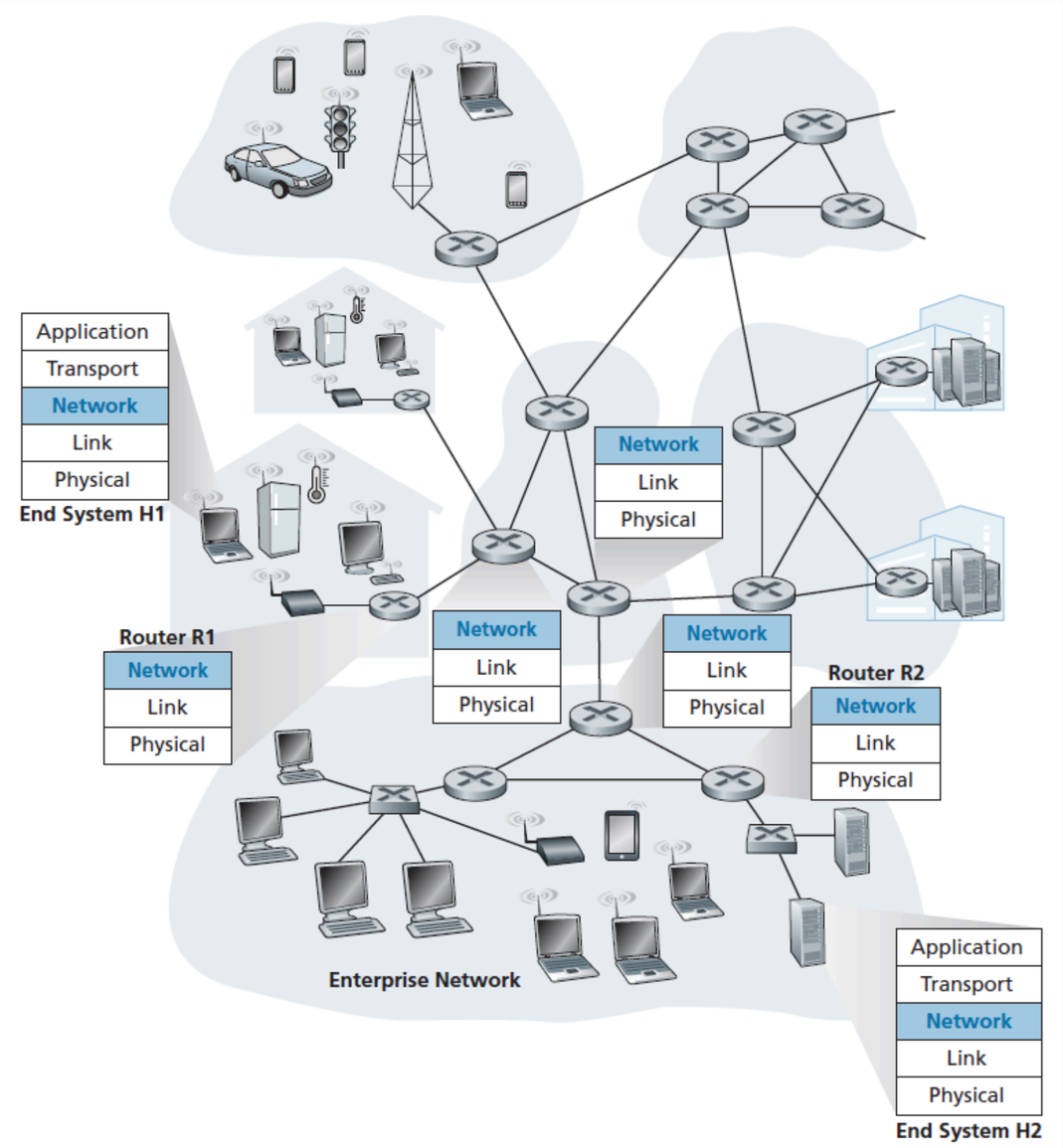
Malik Algazaeery

Chapter4-Network Layer

Agenda:

- **Forwarding and Routing The Data and Control Planes.**
  - **Control Plane: The Traditional Approach.**
  - **Control Plane: The SDN Approach**
- **What's Inside a Router?**
- **Packet Scheduling:**
  - **First-in-First-Out (FIFO).**
  - **Priority Queuing.**
  - **Round Robin and Weighted Fair Queuing (WFQ).**
- **The Internet Protocol (IP): IPv4.**
- **IPv4 Addressing**
- **The Dynamic Host Configuration Protocol.**
- **Network Address Translation (NAT).**
- **Routing Algorithms**
  - **Link-State Algorithm**
- **Intra-AS Routing in the Internet: OSPF**
- **Routing Among the ISPs: BGP**

- In this chapter, we'll learn exactly how the network layer can provide its host-to-host communication service. We'll see that unlike the transport and application layers, there is a piece of the network layer in each and every host and router in the network. Because of this, network-layer protocols are among the most challenging (and therefore among the most interesting!) in the protocol stack.
- Let's suppose that H1 is sending information to H2. The network layer in H1 takes segments from the transport layer in H1, encapsulates each segment into a datagram, and then sends the datagrams to its nearby router, R1. At the receiving host, H2, the network layer receives the datagrams from its nearby router R2, extracts the transport-layer segments, and delivers the segments up to the transport layer at H2.

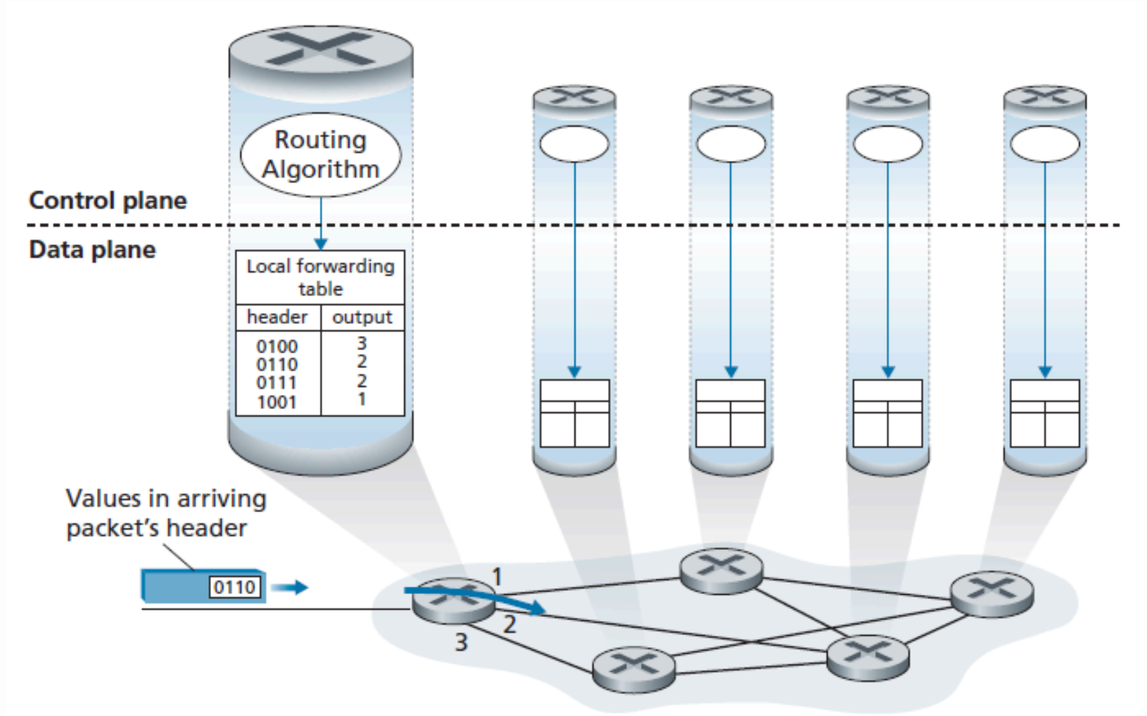


Forwarding and Routing The Data and Control Planes:

- The primary role of the network layer is deceptively simple—to move packets from a sending host to a receiving host. To do so, two important network-layer functions can be identified:
  - Forwarding: When a packet arrives at a router's input link, the router must move the packet to the appropriate output link. Forwarding is implemented in the data plane.
  - Routing: The network layer must determine the route or path taken by packets as they flow from a sender to a receiver. The algorithms that calculate these paths are referred to as routing algorithms. A routing algorithm would determine, for example, the path along which packets flow from H1 to H2. Routing is implemented in the control plane of the network layer.
- Forwarding refers to the router-local action of transferring a packet from an input link interface to the appropriate output link interface. Forwarding takes place at very short timescales (typically a few nanoseconds), and thus is typically implemented in hardware.
- Routing refers to the network-wide process that determines the end-to-end paths that packets take from source to destination. Routing takes place on much longer timescales (typically seconds), and is often implemented in software.
- A key element in every network router is its forwarding table. A router forwards a packet by examining the value of one or more fields in the arriving packet's header, and then using these header values to index into its forwarding table. The value stored in the forwarding table entry for those values indicates the outgoing link interface at that router to which that packet is to be forwarded.

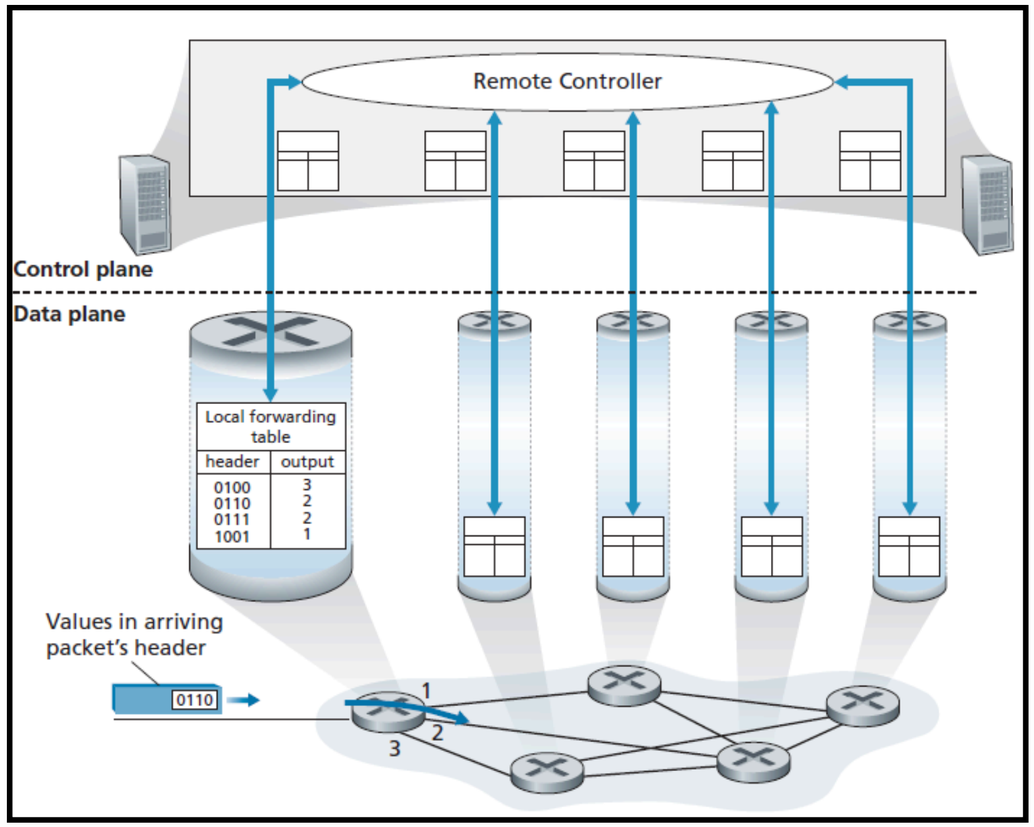
Control Plane: The Traditional Approach

- The routing algorithm determines the contents of the routers' forwarding tables. In this example, a routing algorithm runs in each and every router and both forwarding and routing functions are contained within a router. The routing algorithm function in one router communicates with the routing algorithm function in other routers to compute the values for its forwarding table.
- How is this communication performed?
    - By exchanging routing messages containing routing information according to a routing protocol!



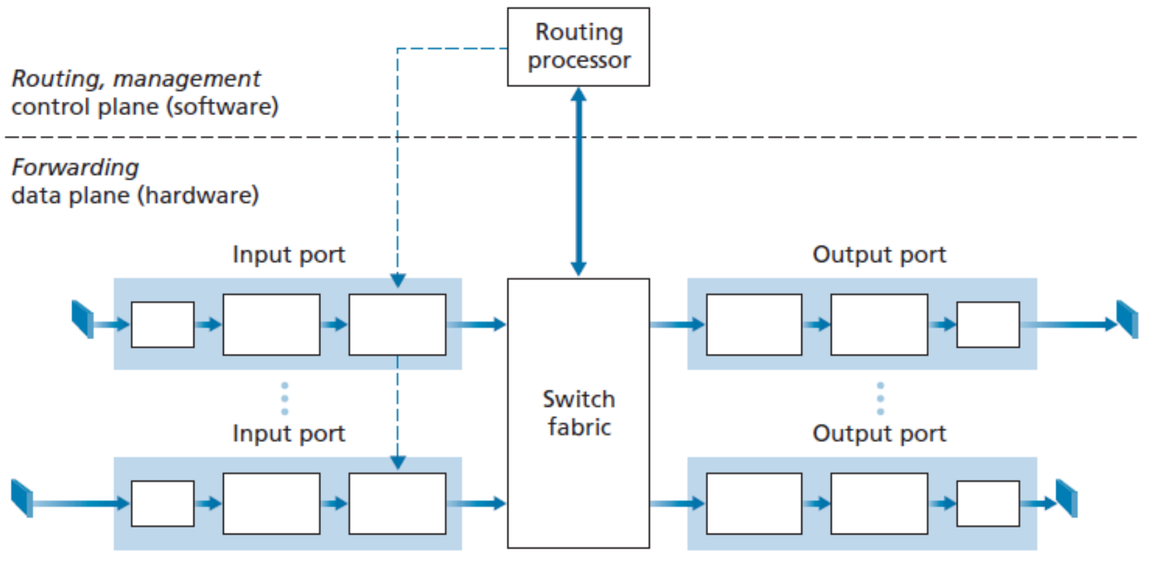
Control Plane: The SDN Approach

- An alternative approach in which a physically separate, remote controller computes and distributes the forwarding tables to be used by each and every router.
- The routing device performs forwarding only, while the remote controller computes and distributes forwarding tables.
- The remote controller might be implemented in a remote data center with high reliability and redundancy, and might be managed by the ISP or some third party.
- The routers and the remote controller communicate by exchanging messages containing forwarding tables and other pieces of routing information.
- The control-plane approach shown in the Figure is at the heart of software-defined networking (SDN), where the network is “software-defined” because the controller that computes forwarding tables and interacts with routers is implemented in software.



What's Inside a Router?

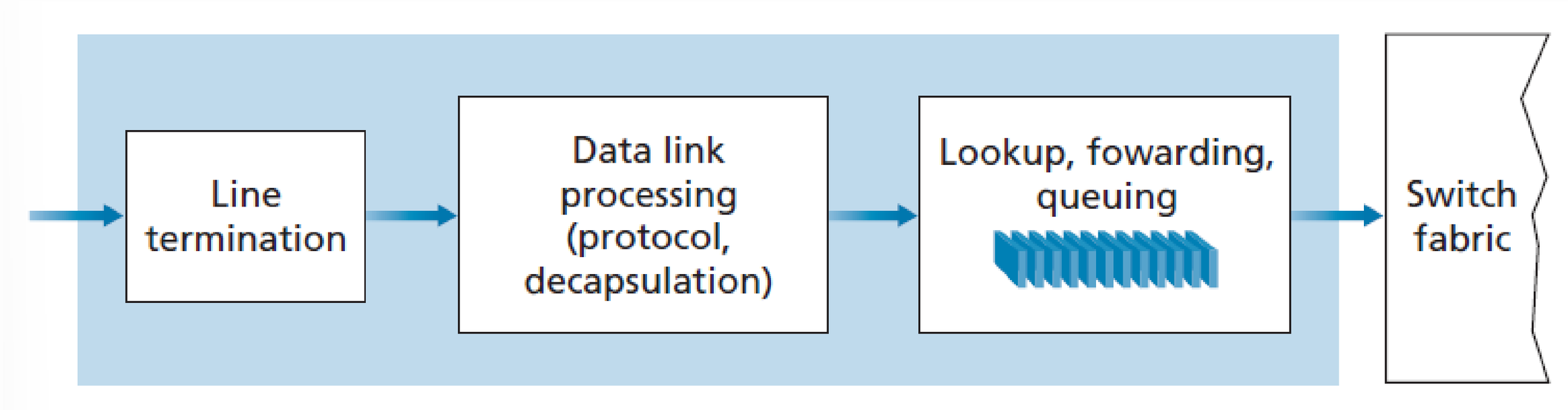
- Input ports: performs several key functions.
  - It performs the physical layer function of terminating an incoming physical link at a router; this is shown in the leftmost box of an input port and the rightmost box of an output port in the Figure
  - An input port also performs link-layer functions needed to interoperate with the link layer at the other side of the incoming link; this is represented by the middle boxes in the input and output ports.
  - Perhaps most crucially, a lookup function is also performed at the input port; this will occur in the rightmost box of the input port. It is here that the forwarding table is consulted to determine the router output port to which an arriving packet will be forwarded via the switching fabric.



    - Control packets (for example, packets carrying routing protocol information) are forwarded from an input port to the routing processor.
- Switching fabric: Connects the router’s input ports to its output ports. This switching fabric is completely contained within the router.
- Output ports: Stores packets received from the switching fabric and transmits these packets on the outgoing link by performing the necessary link-layer and physical-layer functions.
- Routing processor: Performs control-plane functions. In traditional routers, it executes the routing protocols, maintains routing tables, and computes the forwarding table for the router. In SDN routers, the routing processor is responsible for communicating with the remote controller in order to (among other activities) receive forwarding table entries computed by the remote controller, and install these entries in the router’s input ports.

Input Port Processing and Destination-Based Forwarding

- the input port's line-termination function and link-layer processing implement the physical and link layers for that individual input link. The lookup performed in the input port is central to the router's operation—it is here that the router uses the forwarding table to look up the output port to which an arriving packet will be forwarded via the switching fabric.
- The forwarding table is either computed and updated by the routing processor or is received from a remote SDN controller.



- Let’s suppose that our router has four links, numbered 0 through 3, and that packets are to be forwarded to the link interfaces as in the first table:

- We could, for example, have the forwarding table in the second figure below with just four entries:

- You may have noticed that it is possible for a destination address to match more than one entry. For example, the first 24 bits of the address 11001000 00010111 00011000 10101010 match the second entry in the table, and the first 21 bits of the address match the third entry in the table. When there are multiple matches, the router uses the longest prefix matching rule

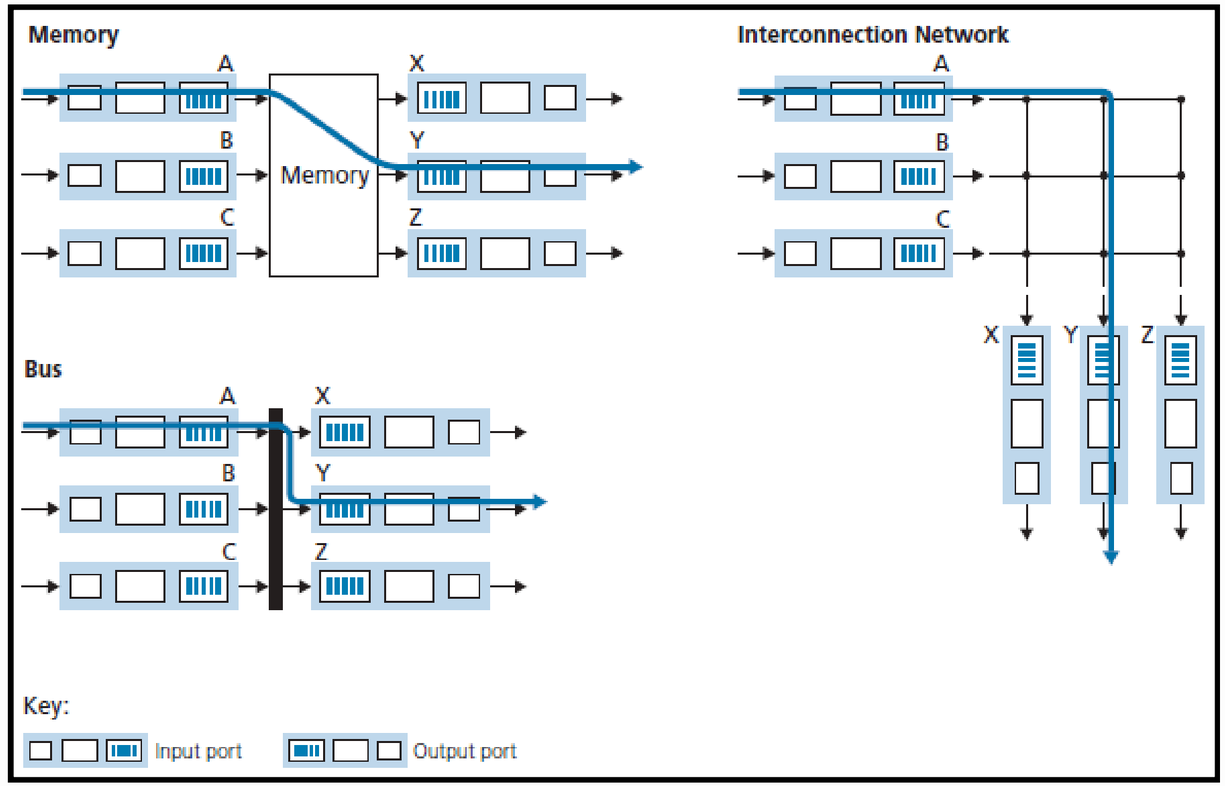
Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
Otherwise	3

Prefix	Link Interface
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
Otherwise	3

Switching

The switching fabric is at the very heart of a router, as it is through this fabric that the packets are actually switched (that is, forwarded) from an input port to an output port. Switching can be accomplished in a number of ways, as shown in the Figure:

- Switching via memory: An input port with an arriving packet first signal the routing processor via an interrupt. The packet is then copied from the input port into processor memory. The routing processor then extracts the destination address from the header, look up the appropriate output port in the forwarding table, and copies the packet to the output port's buffers. In this scenario, if the memory bandwidth is such that a maximum of B packets per second can be written into, or read from, memory, then the overall forwarding throughput must be less than B/2.



- Switching via a bus: In this approach, an input port transfers a packet directly to the output port over a shared bus, without intervention by the routing processor. This is typically done by having the input port pre-pend a switch-internal label (header) to the packet indicating the local output port to which this packet is being transferred and transmitting the packet onto the bus. All output ports receive the packet, but only the port that matches the label will keep the packet. The label is then removed at the output port, as this label is only used within the switch to cross the bus. If multiple packets arrive to the router at the same time, each at a different input port, all but one must wait since only one packet can cross the bus at a time.
- Switching via an interconnection network: One way to overcome the bandwidth limitation of a single, shared bus is to use a more sophisticated interconnection network. A crossbar switch is an interconnection network consisting of 2N buses that connect N input ports to N output ports. Each vertical bus intersects each horizontal bus at a crosspoint, which can be opened or closed at any time by the switch fabric controller. When a packet arrives from port A and needs to be forwarded to port Y, the switch controller closes the crosspoint at the intersection of busses A and Y, and port A then sends the packet onto its bus, which is picked up (only) by bus Y.

Output Port Processing

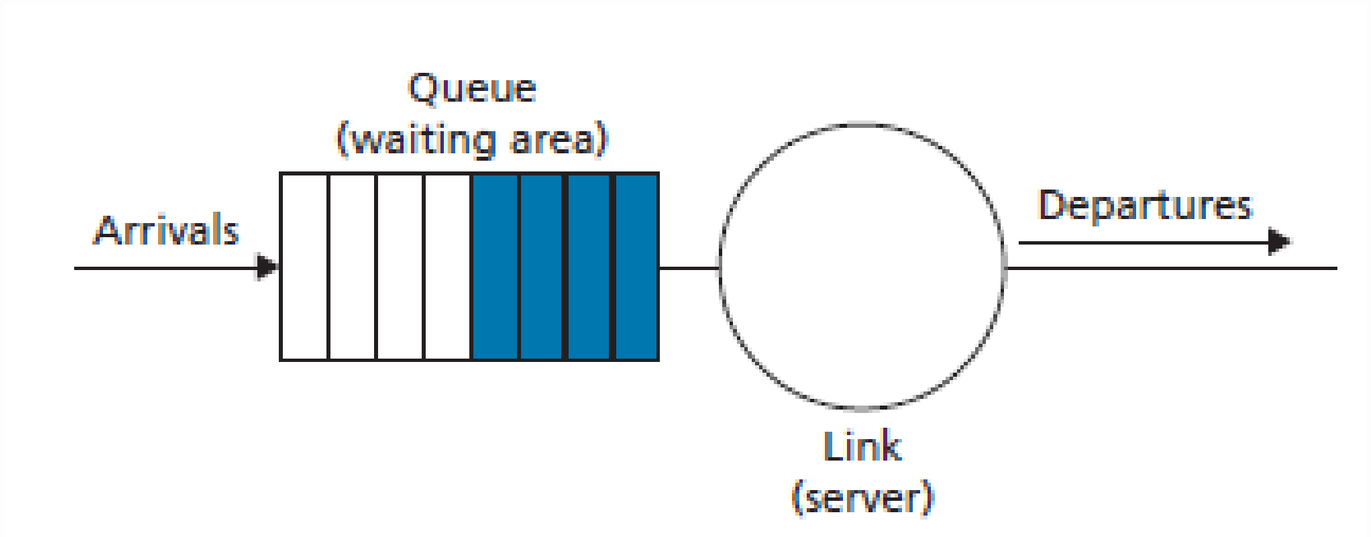
Output port processing takes packets that have been stored in the output port's memory and transmits them over the output link. This includes selecting (i.e., scheduling) and de-queuing packets for transmission, and performing the needed link-layer and physical-layer transmission functions.

Packet Scheduling

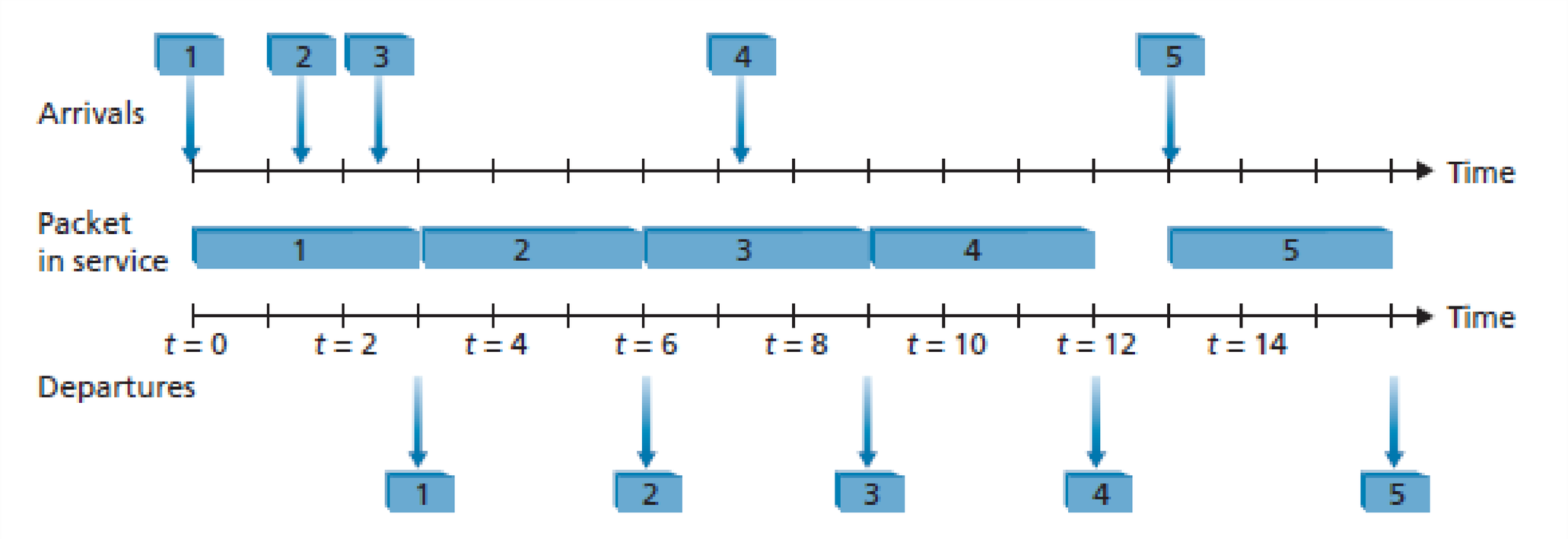
- The order in which queued packets are transmitted over an outgoing link.
- Since you yourself have undoubtedly had to wait in long lines on many occasions and observed how waiting customers are served, you're no doubt familiar with many of the queuing disciplines commonly used in routers. For Example:
  - Ex: There is first-come-first-served (FCFS, also known as first-in-first-out, FIFO)
  - There is also round-robin queuing.

First-in-First-Out (FIFO)

- Packets arriving at the link output queue wait for transmission if the link is currently busy transmitting another packet.
- If there is not sufficient buffering space to hold the arriving packet, the queue's packet-discarding policy then determines whether the packet will be dropped (lost) or whether other packets will be removed from the queue to make space for the arriving packet.
- When a packet is completely transmitted over the outgoing link (that is, receives service) it is removed from the queue.
- The FIFO (also known as first-come-first-served, or FCFS) scheduling discipline selects packets for link transmission in the same order in which they arrived at the output link queue.

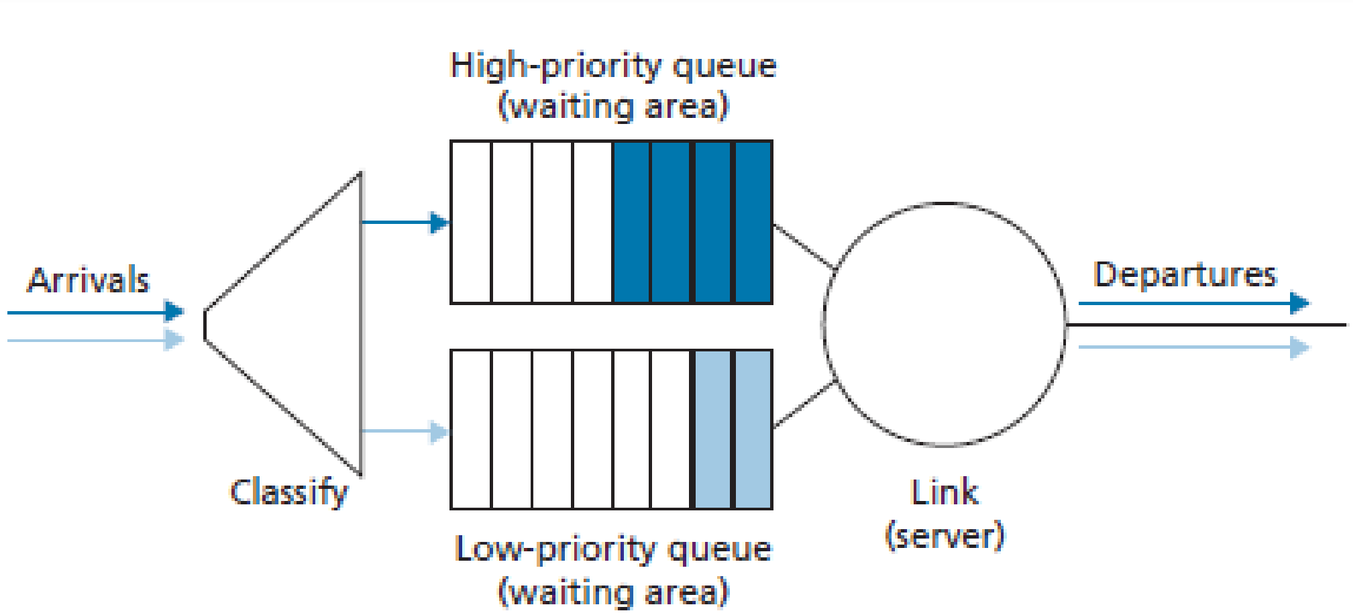


- In our examples here, let’s assume that each packet takes three units of time to be transmitted.
- Under the FIFO discipline, packets leave in the same order in which they arrived.
- Note that after the departure of packet 4, the link remains idle (since packets 1 through 4 have been transmitted and removed from the queue) until the arrival of packet 5.

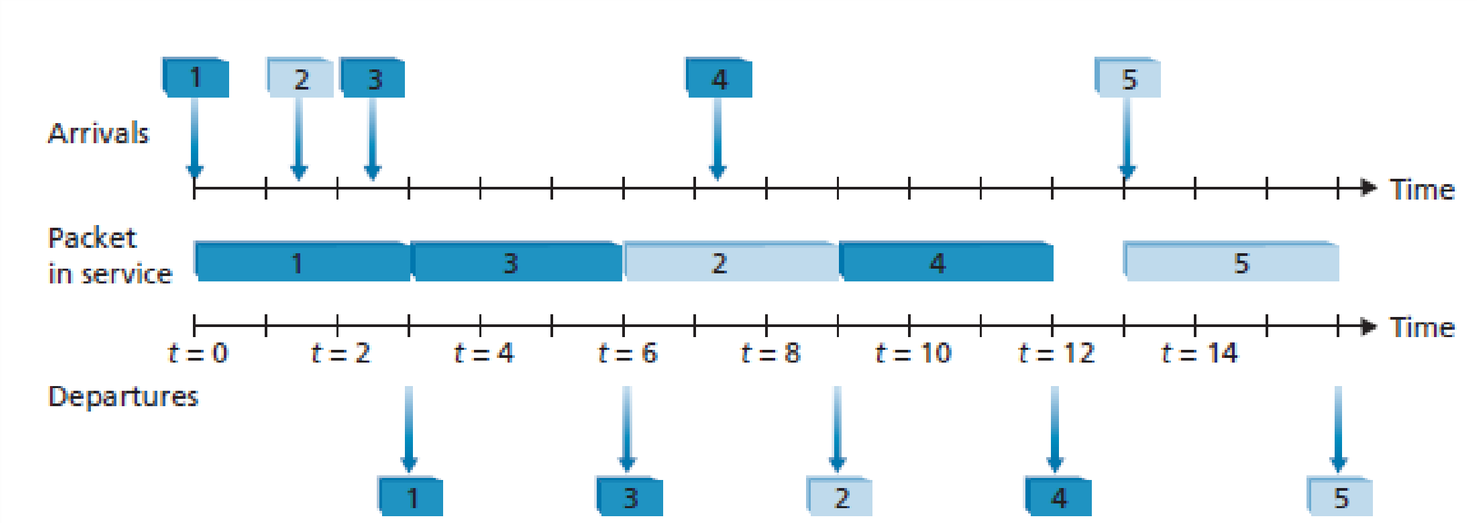


Priority Queuing

- packets arriving at the output link are classified into priority classes upon arrival at the queue.
- In practice, a network operator may configure a queue so that packets carrying network management information (for example, as indicated by the source or destination TCP/UDP port number) receive priority over user traffic; additionally, real-time voice-over-IP packets might receive priority over non-real-time traffic such e-mail packets.
- Each priority class typically has its own queue.
- When choosing a packet to transmit, the priority queuing discipline will transmit a packet from the highest priority class that has a nonempty queue (that is, has packets waiting for transmission). The choice among packets in the same priority class is typically done in a FIFO manner.

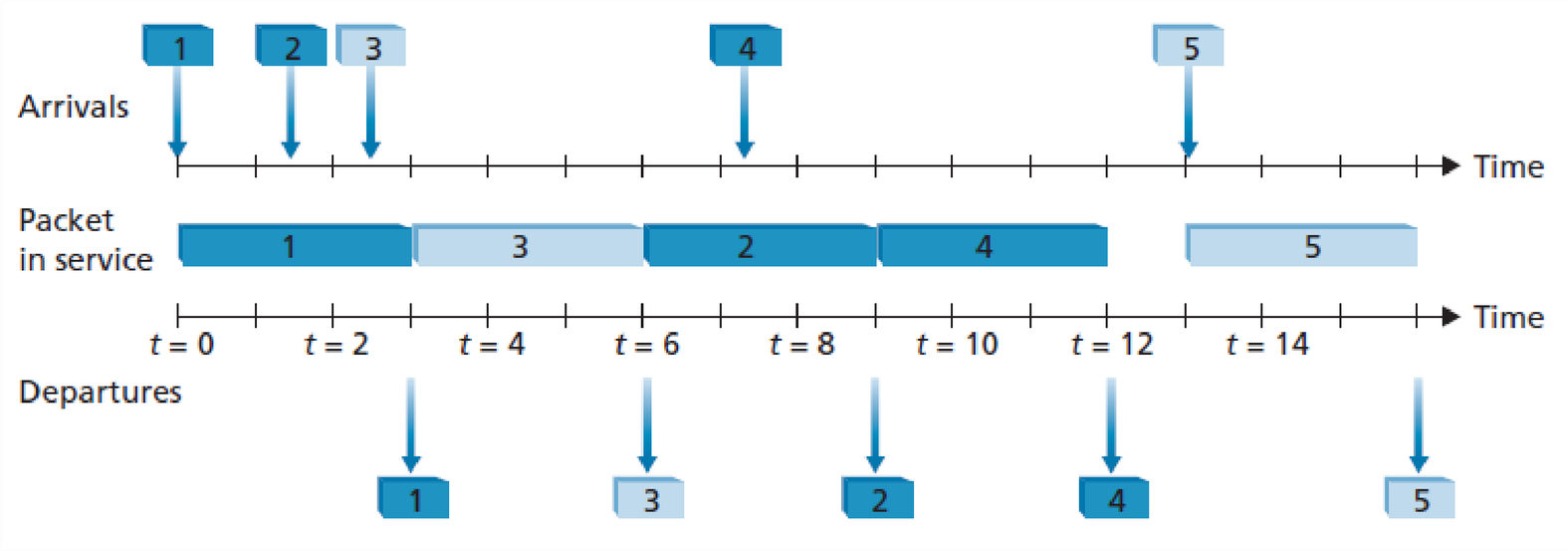


The Figure illustrates the operation of a priority queue with two priority classes. Packets 1, 3, and 4 belong to the high-priority class, and packets 2 and 5 belong to the low-priority class. Packet 1 arrives and, finding the link idle, begins transmission. During the transmission of packet 1, packets 2 and 3 arrive and are queued in the low- and high-priority queues, respectively. After the transmission of packet 1, packet 3 (a high-priority packet) is selected for transmission over packet 2 (which, even though it arrived earlier, is a low-priority packet). At the end of the transmission of packet 3, packet 2 then begins transmission. Packet 4 (a high-priority packet) arrives during the transmission of packet 2 (a low-priority packet). Under a non-preemptive priority queuing discipline, the transmission of a packet is not interrupted once it has begun. In this case, packet 4 queues for transmission and begins being transmitted after the transmission of packet 2 is completed.

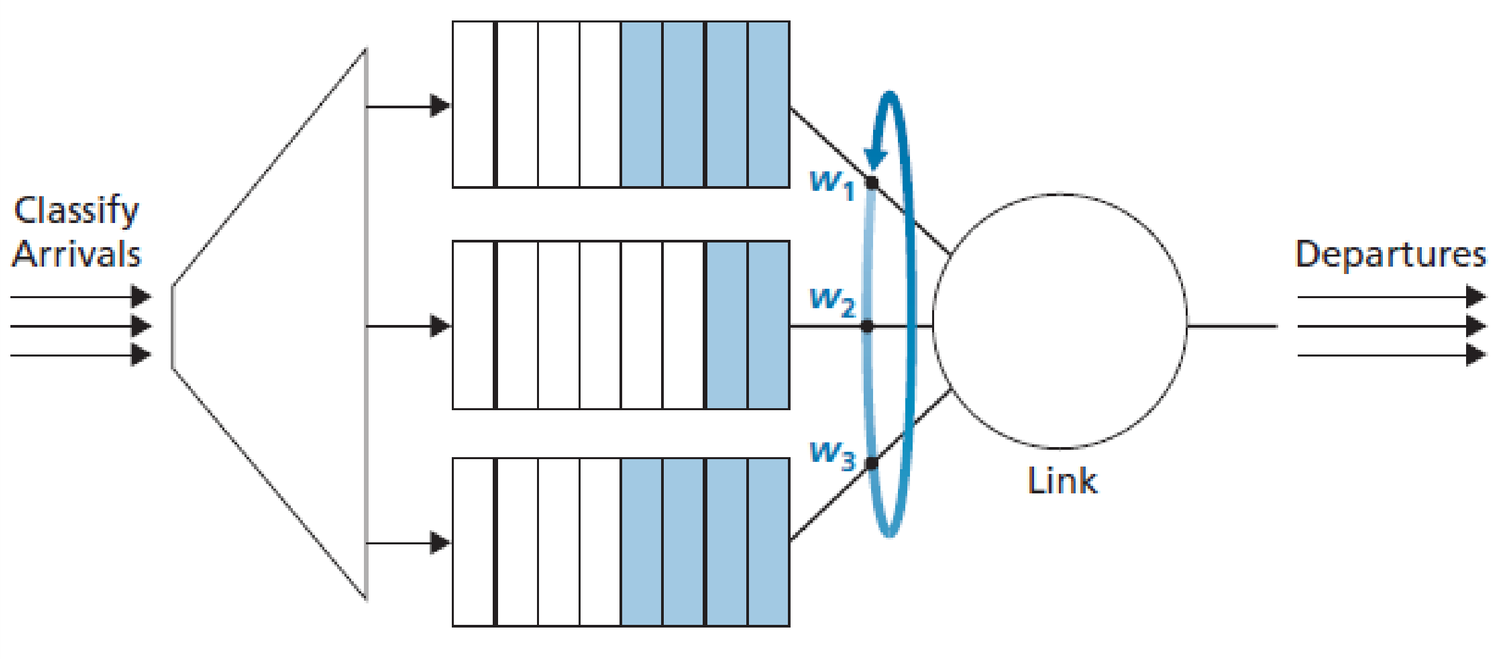


Round Robin and Weighted Fair Queuing (WFQ)

- packets are sorted into classes as with priority queuing. However, rather than there being a strict service priority among classes, a round robin scheduler alternates service among the classes.
- In the simplest form of round robin scheduling, a class 1 packet is transmitted, followed by a class 2 packet, followed by a class 1 packet, followed by a class 2 packet, and so on.
- A so-called work-conserving queuing discipline will never allow the link to remain idle whenever there are packets (of any class) queued for transmission. When looking for a packet of a given class but finds none, immediately checks the next class in the round robin sequence.
- The figure illustrates the operation of a two-class round robin queue. In this example, packets 1, 2, and 4 belong to class 1, and packets 3 and 5 belong to the second class.



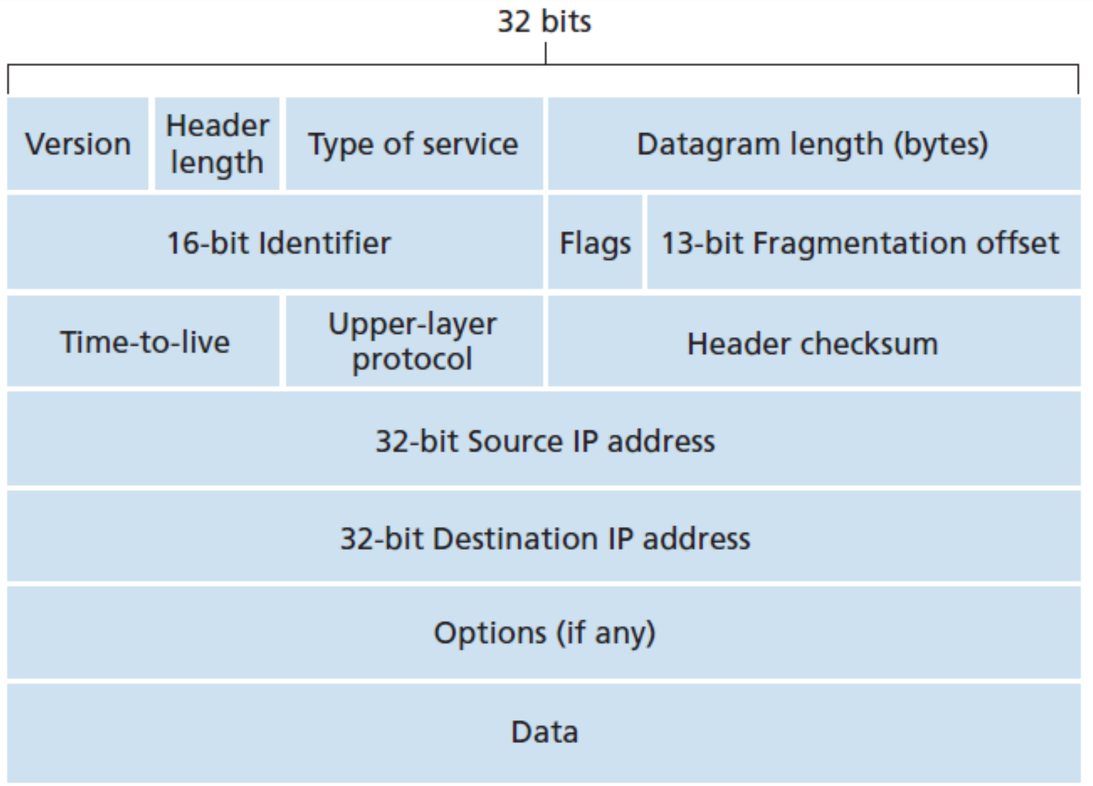
- As in round robin scheduling, a WFQ scheduler will serve classes in a circular manner—first serving class 1, then serving class 2, then serving class 3, and then (assuming there are three classes)repeating the service pattern.
- WFQ differs from round robin in that each class may receive a differential amount of service in any interval of time.



The Internet Protocol (IP): IPv4

Internet's network-layer packet is referred to as a datagram. We begin our study of IP with an overview of the syntax and semantics of the IPv4 datagram.

- Version number: 4 bits specify the IP protocol version of the datagram. By looking at the version number, the router can determine how to interpret the remainder of the IP datagram. Different versions of IP use different datagram formats. The datagram format for IPv4 is shown in the fgure.
- Header length: Because an IPv4 datagram can contain a variable number of options, these 4 bits are needed to determine where in the IP datagram the payload (for example, the transportlayer segment being encapsulated in this datagram) actually begins. Most IP datagrams do not contain options, so the typical IP datagram has a 20-byte header.



- Type of service: (TOS) bits were included in the IPv4 header to allow different types of IP datagrams to be distinguished from each other. For example, it might be useful to distinguish real-time datagrams (such as those used by an IP telephony application) from non-real-time traffic (e.g., FTP).
- Datagram length: This is the total length of the IP datagram (header plus data), measured in bytes. Since this field is 16 bits long, the theoretical maximum size of the IP datagram is 65,535 bytes. However, datagrams are rarely larger than 1,500 bytes, which allows an IP datagram to fit in the payload field of a maximally sized Ethernet frame.
- Identifier, flags, fragmentation offset: These three fields have to do with so-called IP fragmentation, when a large IP datagram is broken into several smaller IP datagrams which are then forwarded independently to the destination, where they are reassembled before their payload data is passed up to the transport layer at the destination host. We'll not cover fragmentation here.
- Time-to-live: (TTL) field is included to ensure that datagrams do not circulate forever in the network. This field is decremented by one each time the datagram is processed by a router. If the TTL field reaches 0, a router must drop that datagram.
- Protocol: This field is typically used only when an IP datagram reaches its final destination. The value of this field indicates the specific transport-layer protocol to which the data portion of this IP datagram should be passed. For example, a value of 6 indicates that the data portion is passed to TCP, while a value of 17 indicates that the data is passed to UDP. Note that the protocol number in the IP datagram has a role that is analogous to the role of the port number field in the transport-layer segment. The protocol number is the glue that binds the network and transport layers together, whereas the port number is the glue that binds the transport and application layers together.

- Header checksum: Aids a router in detecting bit errors in a received IP datagram. The header checksum is computed by treating each 2 bytes in the header as a number and summing these numbers using 1s complement arithmetic. Routers typically discard datagrams for which an error has been detected. Note that only the IP header is checksummed at the IP layer, while the TCP/UDP checksum is computed over the entire TCP/UDP segment.
- Source and destination IP addresses: When a source creates a datagram, it inserts its IP address into the source IP address field and inserts the address of the ultimate destination into the destination IP address field.
- Options: Allow an IP header to be extended. Header options were meant to be used rarely.
- Data (payload): In most circumstances, the data field of the IP datagram contains the transport-layer segment (TCP or UDP) to be delivered to the destination.

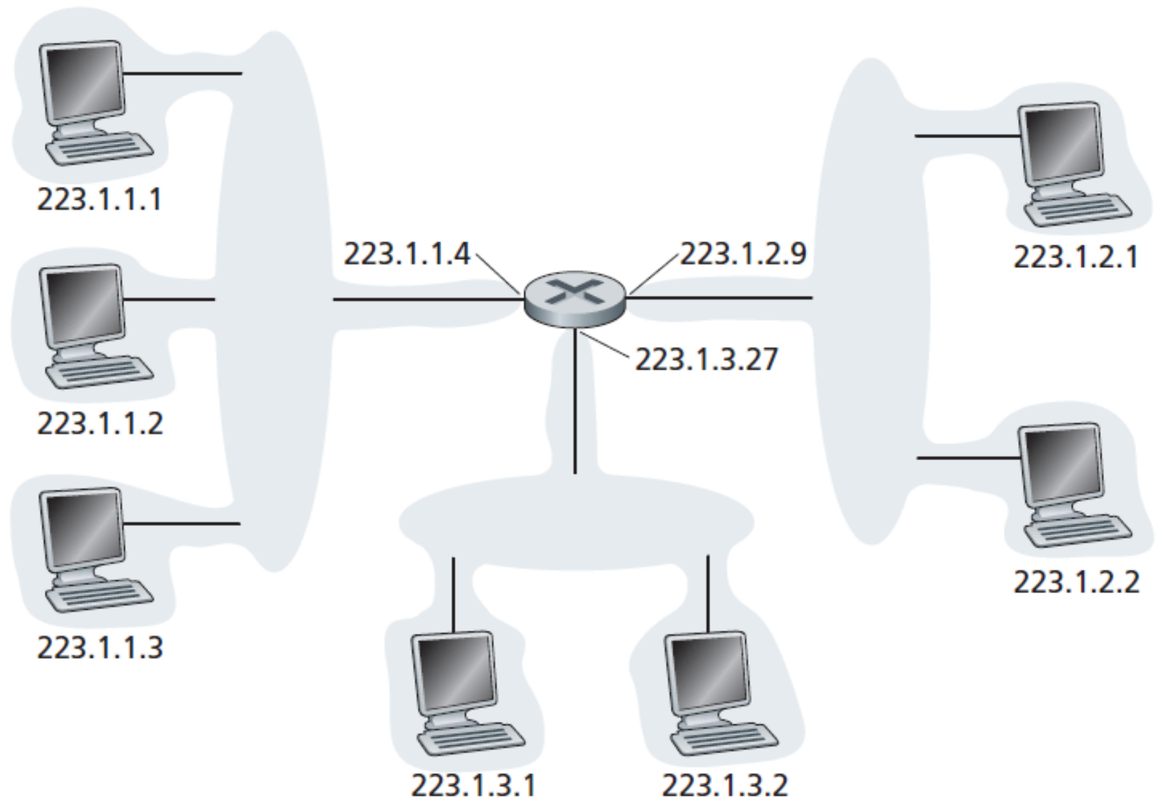
Note that an IP datagram has a total of 20 bytes of header (assuming no options). If the datagram carries a TCP segment, then each datagram carries a total of 40 bytes of header (20 bytes of IP header plus 20 bytes of TCP header) along with the application-layer message.

IPv4 Addressing

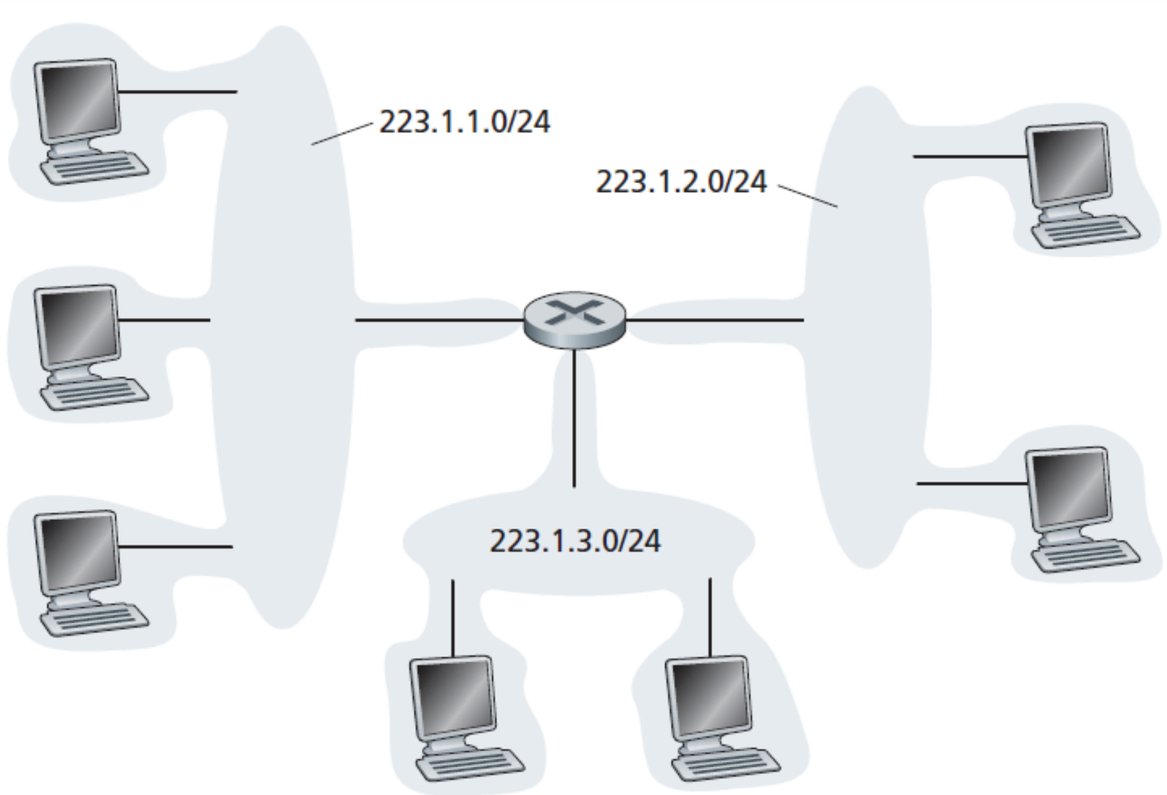
- A host typically has only a single link into the network; when IP in the host wants to send a datagram, it does so over this link.
- The boundary between the host and the physical link is called an interface.
- A router necessarily has two or more links to which it is connected.
- The boundary between the router and any one of its links is also called an interface.
- A router thus has multiple interfaces, one for each of its links.
- IP requires each host and router interface to have its own IP address.
- An IP address is technically associated with an interface, rather than with the host or router containing that interface.
- Each IP address is 32 bits long (equivalently, 4 bytes), and there are thus a total of 2^32 (or approximately 4 billion) possible IP addresses.
- These addresses are typically written in so-called dotted-decimal notation, in which each byte of the address is written in its decimal form and is separated by a period (dot) from other bytes in the address.
- For example, consider the IP address 193.32.216.9. The 193 is the decimal equivalent of the first 8 bits of the address; the 32 is the decimal equivalent of the second 8 bits of the address, and so on. Thus, the address 193.32.216.9 in binary notation is

11000001 00100000 11011000 00001001

- In this figure, one router (with three interfaces) is used to interconnect seven hosts.
- The three hosts in the upper-left portion and the router interface to which they are connected, all have an IP address of the form 223.1.1.xxx. That is, they all have the same leftmost 24 bits in their IP address.
- These four interfaces are also interconnected to each other by a network that contains no routers. This network could be interconnected by an Ethernet LAN, in which case the interfaces would be interconnected by an Ethernet switch, or by a wireless access point.
- We'll represent this routerless network connecting these hosts as a cloud for now.
- In IP terms, this network interconnecting three host interfaces and one router interface forms a **subnet**.

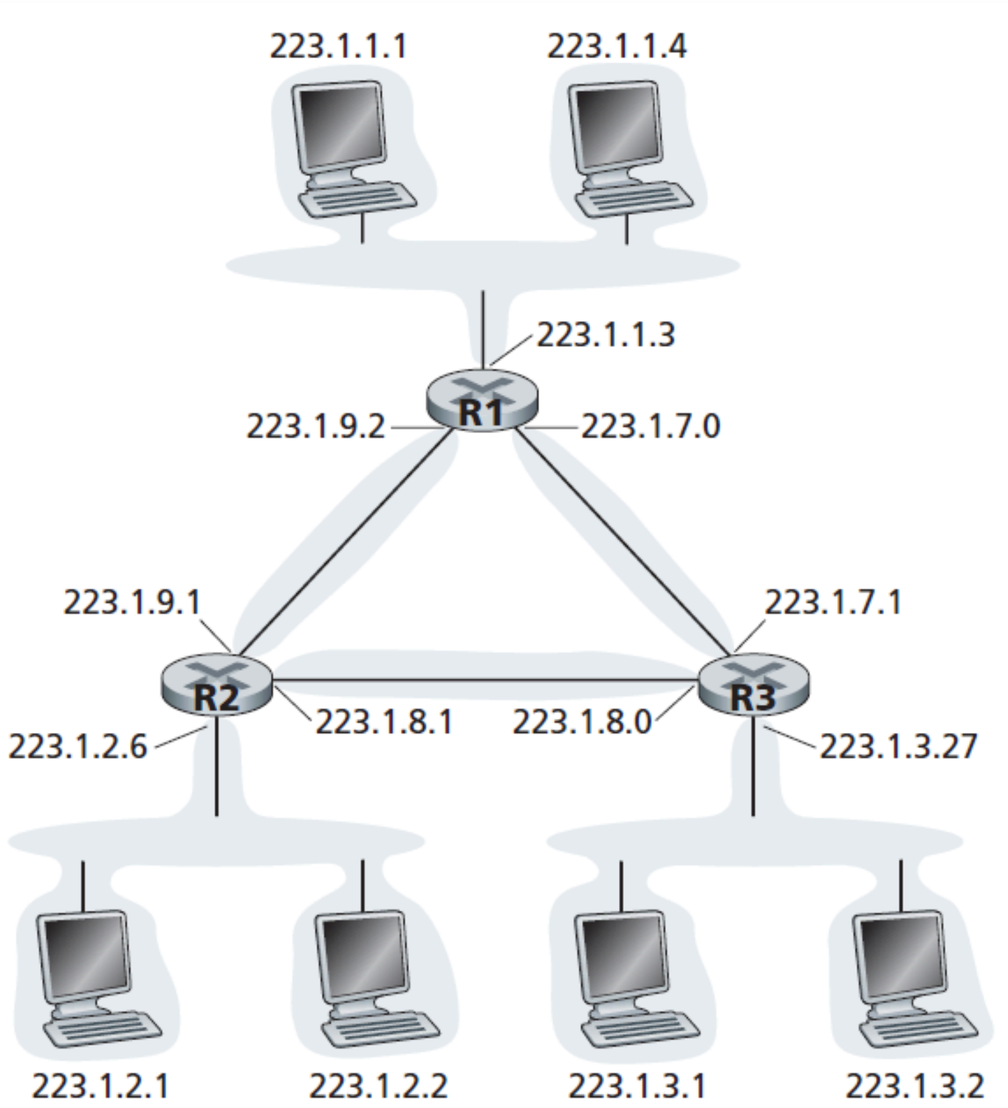


- IP addressing assigns an address to this subnet: 223.1.1.0/24, where the /24 (“slash-24”) notation, sometimes known as a subnet mask, indicates that the leftmost 24 bits of the 32-bit quantity define the subnet address.
- The 223.1.1.0/24 subnet thus consists of the three host interfaces (223.1.1.1, 223.1.1.2, and 223.1.1.3) and one router interface (223.1.1.4).
- Any additional hosts attached to the 223.1.1.0/24 subnet would be required to have an address of the form 223.1.1.xxx.
- There are two additional subnets shown in the figure on the slide, the 223.1.2.0/24 network and the 223.1.3.0/24 subnet. The figure on the right side illustrates the three IP subnets present.



Example: three routers that are interconnected with each other by point-to-point links.

- Each router has three interfaces, one for each point-to-point link and one for the broadcast link that directly connects the router to a pair of hosts.
- Three subnets, 223.1.1.0/24, 223.1.2.0/24, and 223.1.3.0/24, are similar to the subnets we encountered in last example.
- Note that there are three additional subnets in this example as well: one subnet, 223.1.9.0/24, for the interfaces that connect routers R1 and R2; another subnet, 223.1.8.0/24, for the interfaces that connect routers R2 and R3; and a third subnet, 223.1.7.0/24, for the interfaces that connect routers R3 and R1.



- An organization with multiple Ethernet segments and point-to-point links will have multiple subnets, with all of the devices on a given subnet having the same subnet address.
- The Internet's address assignment strategy is known as Classless Interdomain Routing (CIDR—pronounced cider)
- As with subnet addressing, the 32-bit IP address is divided into two parts and again has the dotted-decimal form a.b.c.d/x, where x indicates the number of bits in the first part of the address.
- The x most significant bits of an address of the form a.b.c.d/x constitute the network portion of the IP address, and are often referred to as the prefix (or network prefix) of the address.
- An organization is typically assigned a block of contiguous addresses, that is, a range of addresses with a common prefix. In this case, the IP addresses of devices within the organization will share the common prefix.
- Only these x leading prefix bits are considered by routers outside the organization's network. That is, when a router outside the organization forwards a datagram whose destination address is inside the organization, only the leading x bits of the address need be considered.
- The remaining 32-x bits of an address can be thought of as distinguishing among the devices within the organization, all of which have the same network prefix.
- The IP broadcast address 255.255.255.255: When a host sends a datagram with destination address 255.255.255.255, the message is delivered to all hosts on the same subnet.

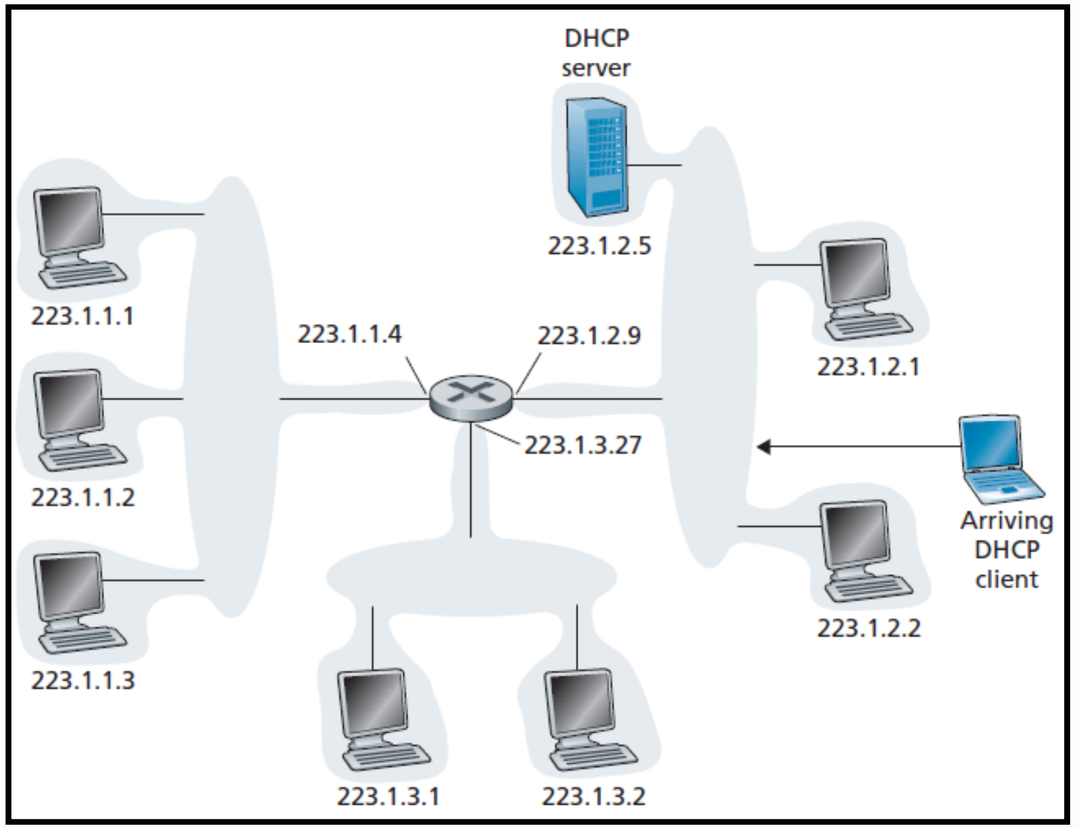
Obtaining a Block of Addresses

- In order to obtain a block of IP addresses for use within an organization's subnet, a network administrator might first contact its ISP, which would provide addresses from a larger block of addresses that had already been allocated to the ISP.
- For example, the ISP may itself have been allocated the address block 200.23.16.0/20. The ISP, in turn, could divide its address block into eight equal-sized contiguous address blocks and give one of these address blocks out to each of up to eight organizations that are supported by this ISP, as shown below.
- There is a global authority that has ultimate responsibility for managing the IP address space and allocating address blocks to ISPs and other organizations (ICANN).

ISP's block:	200.23.16.0/20	11001000 00010111 00010000 00000000
Organization 0	200.23.16.0/23	11001000 00010111 00010000 00000000
Organization 1	200.23.18.0/23	11001000 00010111 00010010 00000000
Organization 2	200.23.20.0/23	11001000 00010111 00010100 00000000
… …		…
Organization 7	200.23.30.0/23	11001000 00010111 00011110 00000000

Obtaining a Host Address: The Dynamic Host Configuration Protocol

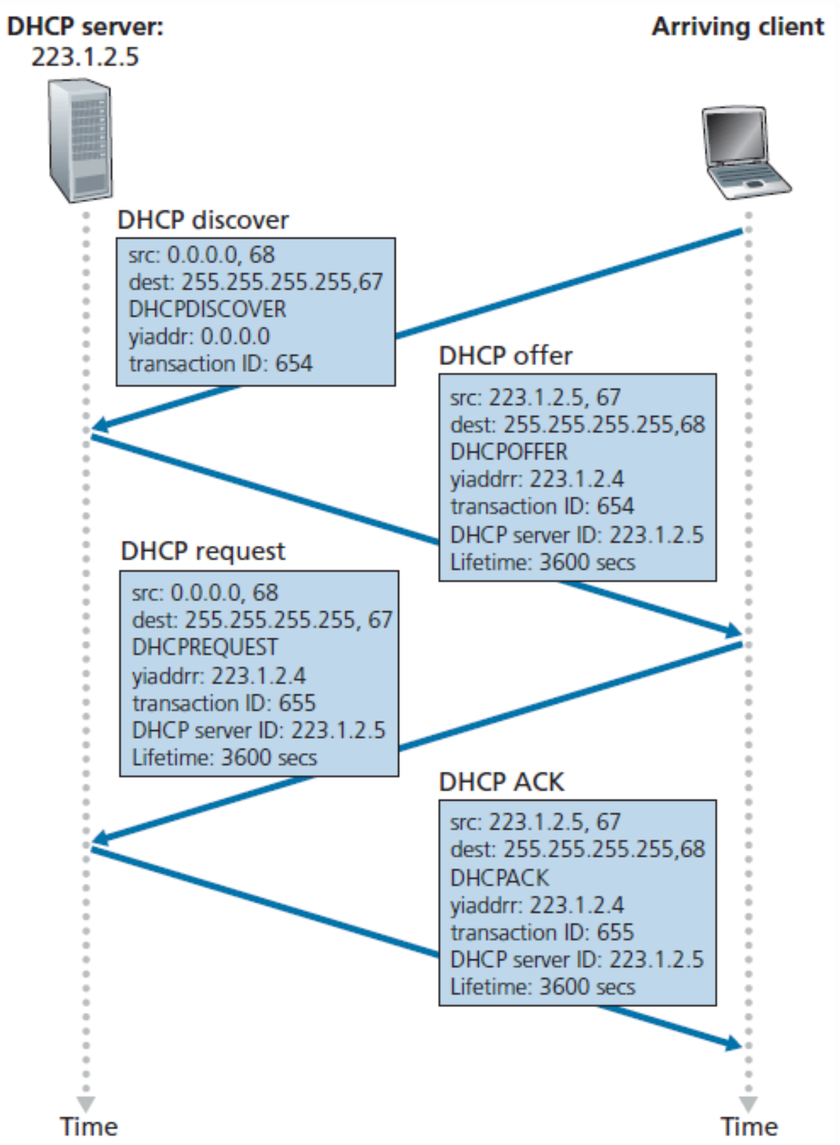
- Once an organization has obtained a block of addresses, it can assign individual IP addresses to the host and router interfaces in its organization. A system administrator will typically manually configure the IP addresses into the router.
- Host addresses can be configured manually, but typically this is done using the Dynamic Host Configuration Protocol (DHCP)
- DHCP allows a host to obtain (be allocated) an IP address automatically.
- A network administrator can configure DHCP so that a given host receives the same IP address each time it connects to the network, or a host may be assigned a temporary IP address that will be different each time the host connects to the network.



- DHCP also allows a host to learn additional information, such as its subnet mask, the address of its first-hop router (often called the default gateway), and the address of its local DNS server.
- DHCP is a client-server protocol. A client is typically a newly arriving host wanting to obtain network configuration information, including an IP address for itself.

For a newly arriving host, the DHCP protocol is a four-step process, as shown. In this figure, yiaddr (as in “your Internet address”) indicates the address being allocated to the newly arriving client. The four steps are:

- DHCP server discovery: The client sends a DHCP discover messag within a UDP packet to port 67. The UDP packet is encapsulated in an IP datagram. The host doesn’t know the IP address of the network to which it is attaching, much less the address of a DHCP server for this network. Given this, the DHCP client creates an IP datagram containing its DHCP discover message along with the broadcast destination IP address of 255.255.255.255 and a “this host” source IP address of 0.0.0.0. The DHCP client passes the IP datagram to the link layer, which then broadcasts this frame to all nodes attached to the subnet.
- DHCP server offer(s): A DHCP server receiving a DHCP discover message responds to the client with a DHCP offer message that is broadcast to all nodes on the subnet, again using the IP broadcast address of 255.255.255.255. Since several DHCP servers can be present on the subnet, the client may find itself in the enviable position of being able to choose from among several offers.



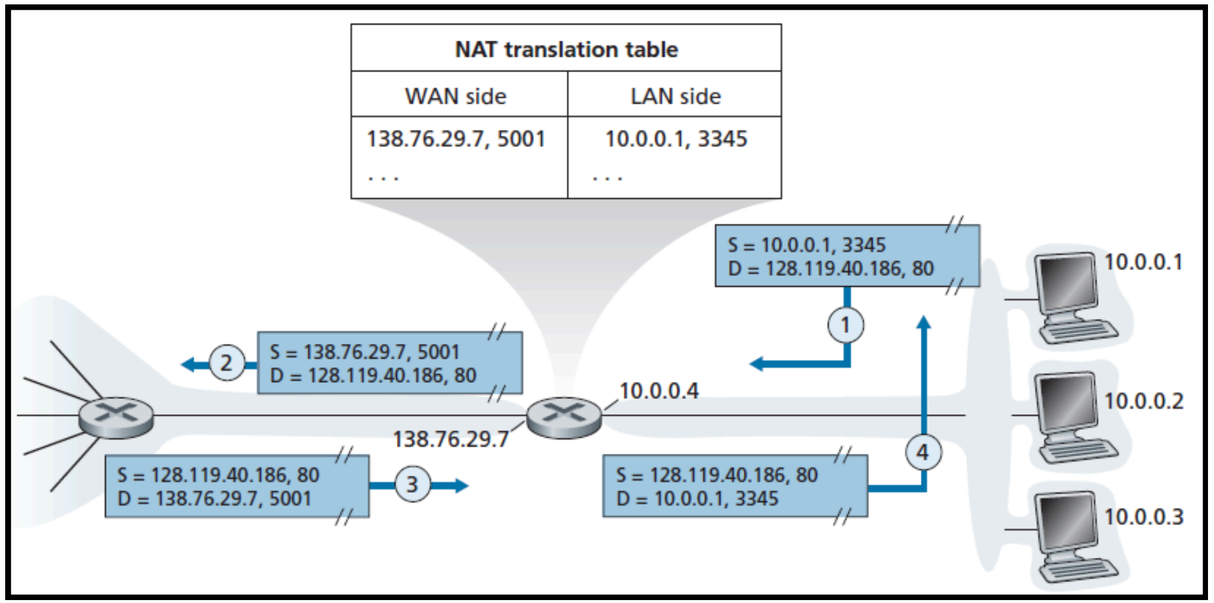
- DHCP request: The newly arriving client will choose from among one or more server offers and respond to its selected offer with a DHCP request message, echoing back the configuration parameters.
- DHCP ACK: The server responds to the DHCP request message with a DHCP ACK message, confirming the requested parameters.

Once the client receives the DHCP ACK, the interaction is complete and the client can use the DHCP-allocated IP address for the lease duration. Since a client may want to use its address beyond the lease's expiration, DHCP also provides a mechanism that allows a client to renew its lease on an IP address.

Network Address Translation (NAT)

- The NAT-enabled router, residing in the home, has an interface that is part of the home network.
- Addressing within the home network is exactly as we have seen —all four interfaces in the home network have the same subnet address of 10.0.0.0/24.
- The address space 10.0.0.0/8 is one of three portions of the IP address space that is reserved in [RFC 1918] for a private network, such as the home network.
- A realm with private addresses refers to a network whose addresses only have meaning to devices within that network.
- Consider the fact that there are hundreds of thousands of home networks, many using the same address space, 10.0.0.0/24. Devices within a given home network can send packets to each other using 10.0.0.0/24 addressing. However, packets forwarded beyond the home network into the larger global Internet clearly cannot use these addresses (as either a source or a destination address) because there are hundreds of thousands of networks using this block of addresses. That is, the 10.0.0.0/24 addresses can only have meaning within the given home network.
- But if private addresses only have meaning within a given network, how is addressing handled when packets are sent to or received from the global Internet, where addresses are necessarily unique? The answer lies in understanding NAT.

- the NAT router behaves to the outside world as a single device with a single IP address. In the Figure, all traffic leaving the home router for the larger Internet has a source IP address of 138.76.29.7, and all traffic entering the home router must have a destination address of 138.76.29.7.
- In essence, the NAT-enabled router is hiding the details of the home network from the outside world.

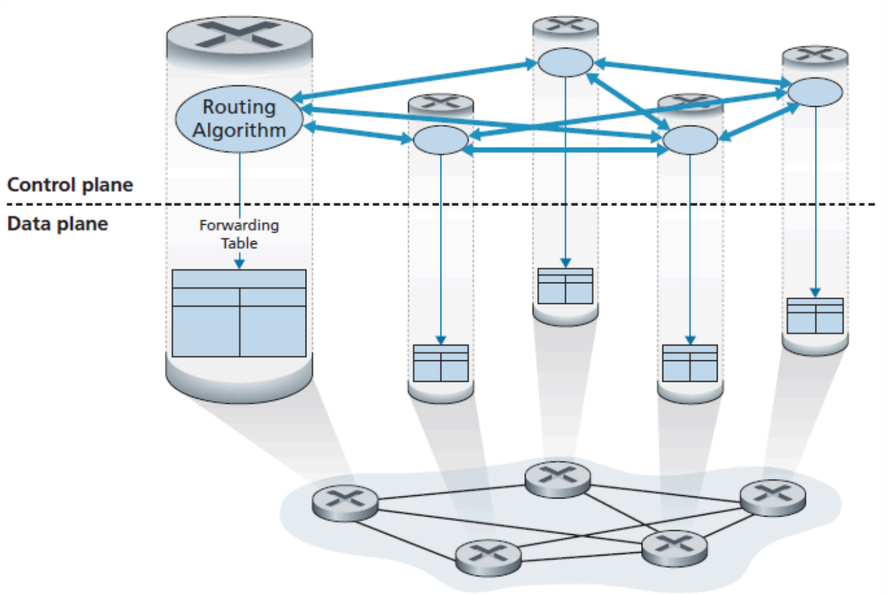


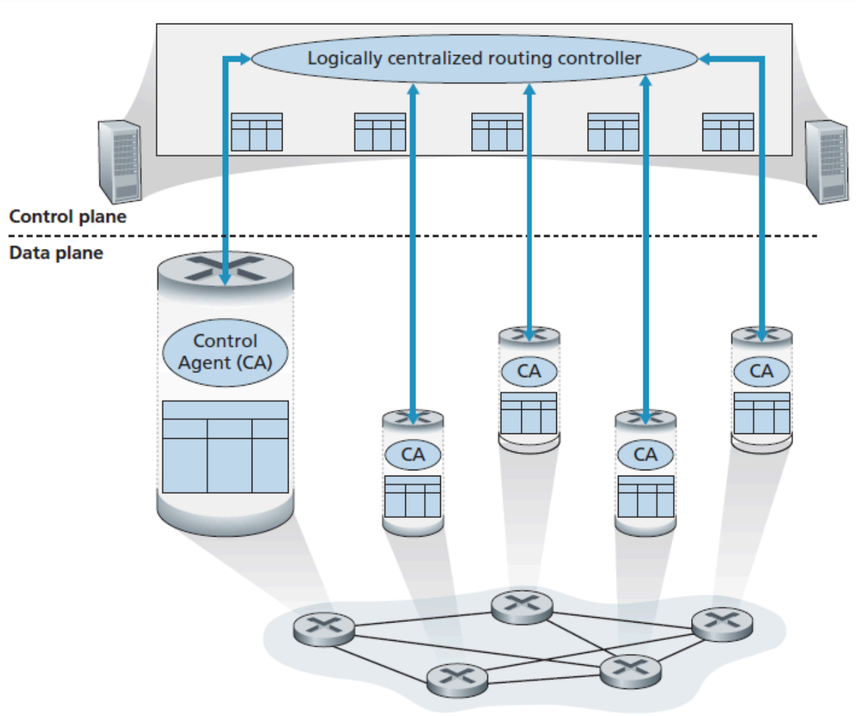
- If all datagrams arriving at the NAT router from the WAN have the same destination IP address, then how does the router know the internal host to which it should forward a given datagram? The trick is to use a NAT translation table at the NAT router, and to include port numbers as well as IP addresses in the table entries.
- Suppose a user sitting in a home network behind host 10.0.0.1 requests a Web page on some Web server (port 80) with IP address 128.119.40.186. The host 10.0.0.1 assigns the (arbitrary) source port number 3345 and sends the datagram into the LAN. The NAT router receives the datagram, generates a new source port number 5001 for the datagram, replaces the source IP address with its WAN-side IP address 138.76.29.7, and replaces the original source port number 3345 with the new source port number 5001. When generating a new source port number, the NAT router can select any source port number that is not currently in the NAT translation table.

- NAT in the router also adds an entry to its NAT translation table.
- The Web server, blissfully unaware that the arriving datagram containing the HTTP request has been manipulated by the NAT router, responds with a datagram whose destination address is the IP address of the NAT router, and whose destination port number is 5001.
- When this datagram arrives at the NAT router, the router indexes the NAT translation table using the destination IP address and destination port number to obtain the appropriate IP address (10.0.0.1) and destination port number (3345) for the browser in the home network. The router then rewrites the datagram's destination address and destination port number, and forwards the datagram into the home network.

The Network Layer: Control Plane

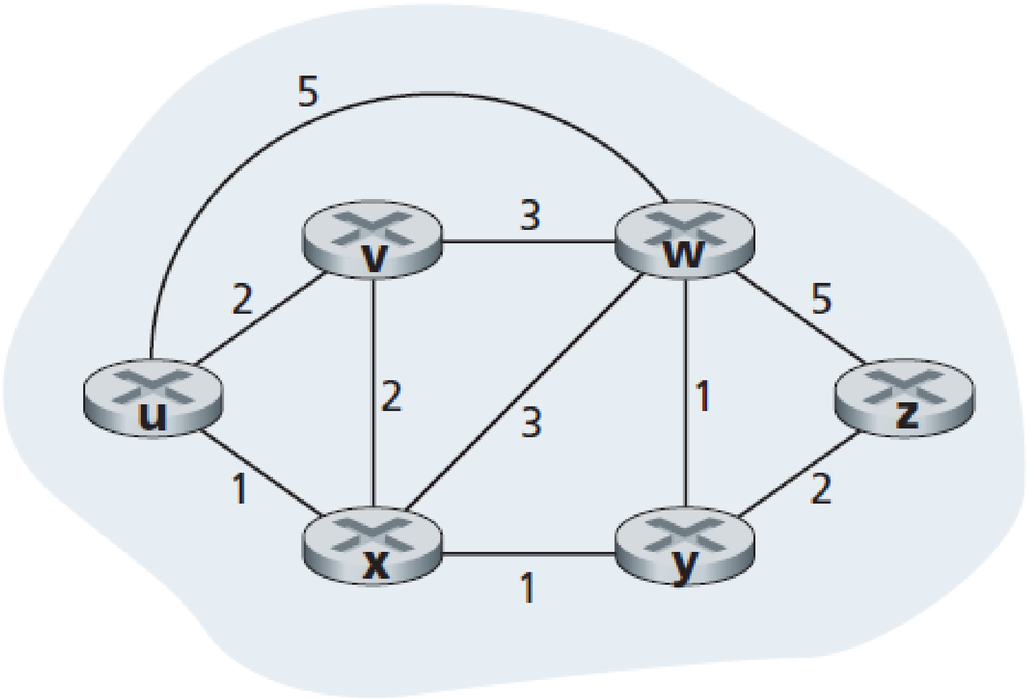
- Per-router control. First Figure illustrates the case where a routing algorithm runs in each and every router; both a forwarding and a routing function are contained within each router. Each router has a routing component that communicates with the routing components in other routers to compute the values for its forwarding table. This per-router control approach has been used in the Internet for decades.
- Logically centralized control. The secons Figure illustrates the case in which a logically centralized controller computes and distributes the forwarding tables to be used by each and every router.
- The controller interacts with a control agent (CA) in each of the routers via a well-defined protocol to configure and manage that router's flow table. Typically, the CA has minimum functionality; its job is to communicate with the controller, and to do as the controller commands. Unlike the routing algorithms in the first Figure, the CAs do not directly interact with each other nor do they actively take part in computing the forwarding table. This is a key distinction between per-router control and logically centralized control.





Routing Algorithms

- goal is to determine good paths (equivalently, routes), from senders to receivers, through the network of routers. Typically, a “good” path is one that has the least cost.
- A graph is used to formulate routing problems. Recall that a graph G = (N, E) is a set N of nodes and a collection E of edges, where each edge is a pair of nodes from N. In the context of network-layer routing, the nodes in the graph represent routers and the edges connecting these nodes represent the physical links between these routers.



- As shown in the figure , an edge also has a value representing its cost. Typically, an edge’s cost may reflect the physical length of the corresponding link, the link speed, or the monetary cost associated with a link. For our purposes, we’ll simply take the edge costs as a given and won’t worry about how they are determined.
- For any edge (x, y) in E, we denote c(x, y) as the cost of the edge between nodes x and y.
- we’ll only consider undirected graphs in our discussion here, so that edge (x, y) is the same as edge (y, x) and that c(x, y) = c(y, x).

- a node y is said to be a neighbor of node x if (x, y) belongs to E.
- a natural goal of a routing algorithm is to identify the least costly paths between sources and destinations.
- a path in a graph G = (N, E) is a sequence of nodes (X1, X2, ..., Xp) such that each of the pairs (X1, X2), (X2, X3), ..., (Xp-1, Xp) are edges in E. The cost of a path (X1, X2, ..., Xp) is simply the sum of all the edge costs along the path, that is, c(X1, X2) + c(X2, X3) + ...+ c(Xp-1, Xp).
- Given any two nodes x and y, there are typically many paths between the two nodes, with each path having a cost. One or more of these paths is a least-cost path.
- The least-cost problem is therefore clear: Find a path between the source and destination that has least cost. In last Figure, for example, the least-cost path between source node u and destination node w is (u, x, y, w) with a path cost of 3.
- Broadly, one way in which we can classify routing algorithms is according to whether they are centralized or decentralized.
    - A centralized routing algorithm computes the least-cost path between a source and destination using complete, global knowledge about the network. The key distinguishing feature here, is that the algorithm has complete information about connectivity and link costs. Algorithms with global state information are often referred to as link-state (LS) algorithms
    - In a decentralized routing algorithm, the calculation of the least-cost path is carried out in an iterative, distributed manner by the routers. No node has complete information about the costs of all network links.

The Link-State (LS) Routing Algorithm

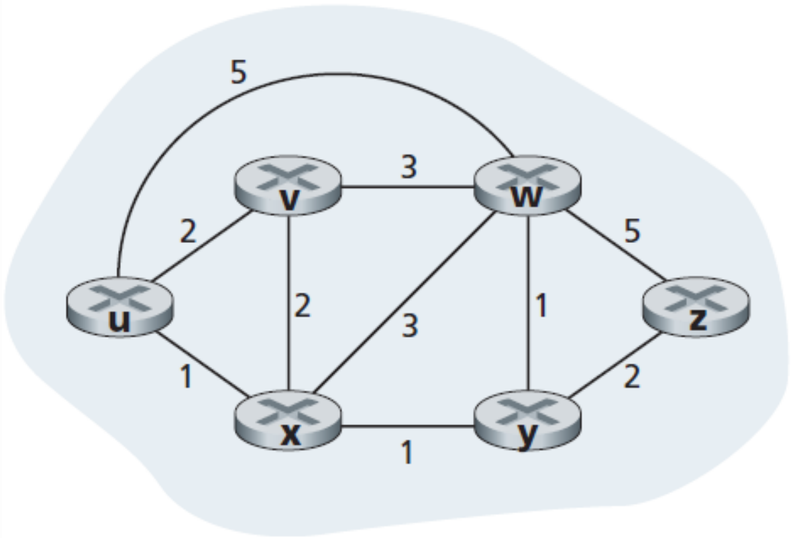
- network topology and all link costs are known, that is, available as input to the LS algorithm.
- This is accomplished by having each node broadcast link-state packets to all other nodes in the network, with each link-state packet containing the identities and costs of its attached links.
- The result of the nodes' broadcast is that all nodes have an identical and complete view of the network. Each node can then run the LS algorithm and compute the same set of least-cost paths as every other node.
- The link-state routing algorithm we present is known as Dijkstra's algorithm, named after its inventor.
- Dijkstra's algorithm computes the least-cost path from one node (the source, which we will refer to as u) to all other nodes in the network. Dijkstra's algorithm is iterative and has the property that after the kth iteration of the algorithm, the least-cost paths are known to k destination nodes, and among the least-cost paths to all destination nodes, these k paths will have the k smallest costs.
- D(v): cost of the least-cost path from the source node to destination v as of this iteration of the algorithm.
- p(v): previous node (neighbor of v) along the current least-cost path from the source to v.
- N´: subset of nodes; v is in N´ if the least-cost path from the source to v is definitively known.
- The centralized routing algorithm consists of an initialization step followed by a loop. The number of times the loop is executed is equal to the number of nodes in the network. Upon termination, the algorithm will have calculated the shortest paths from the source node u to every other node in the network.

Link-State (LS) Algorithm for Source Node *u*

```
1   Initialization:
2    N′ = {u}
3    for all nodes v
4      if v is a neighbor of u
5        then D(v) = c(u,v)
6      else D(v) = ∞
7
8   Loop
9    find w not in N′ such that D(w) is a minimum
10   add w to N′
11   update D(v) for each neighbor v of w and not in N′:
12           D(v) = min(D(v), D(w)+ c(w,v) )
13    /* new cost to v is either old cost to v or known
14     least path cost to w plus cost from w to v */
15  until N′= N
```

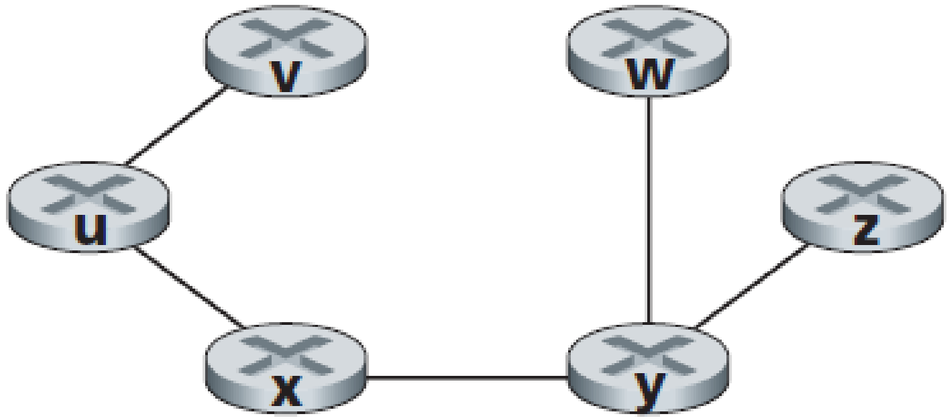
Example:

- In the initialization step, the currently known least-cost paths from u to its directly attached neighbors, v, x, and w, are initialized to 2, 1, and 5, respectively.
- In the first iteration, we look among those nodes not yet added to the set N´and find that node with the least cost as of the end of the previous iteration.
- In the second iteration, nodes v and y are found to have the least-cost paths (2), and we break the tie arbitrarily and add y to the set N´so that N´ now contains u, x, and y.
- And so on . . .



step	N′	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2, u	5, u	1,u	∞	∞
1	ux	2, u	4, x		2, x	∞
2	uxy	2, u	3, y			4, y
3	uxyv		3, y			4, y
4	uxyvw					4, y
5	uxyvwz					

When the LS algorithm terminates, we have, for each node, its predecessor along the least-cost path from the source node. For each predecessor, we also have its predecessor, and so in this manner we can construct the entire path from the source to all destinations. The forwarding table in a node, say node u, can then be constructed from this information by storing, for each destination, the next-hop node on the leastcost path from u to the destination. The figure shows the resulting least-cost paths and forwarding table in u for the network.



Destination	Link
v	(u, v)
w	(u, x)
x	(u, x)
y	(u, x)
z	(u, x)

Intra-AS Routing in the Internet: OSPF

- autonomous systems (ASs): each AS consists of a group of routers that are under the same administrative control. Often the routers in an ISP, and the links that interconnect them, constitute a single AS. Some ISPs, however, partition their network into multiple ASs.
- An autonomous system is identified by its globally unique autonomous system number (ASN). AS numbers, like IP addresses, are assigned by ICANN regional registries.
- Routers within the same AS all run the same routing algorithm and have information about each other. The routing algorithm running within an autonomous system is called an intra-autonomous system routing protocol.
- Open Shortest Path First (OSPF) routing is widely used for intra-AS routing in the Internet. The Open in OSPF indicates that the routing protocol specification is publicly available.
- OSPF is a link-state protocol that uses flooding of link-state information and a Dijkstra's least-cost path algorithm. With OSPF, each router constructs a complete topological map (that is, a graph) of the entire autonomous system. Each router then locally runs Dijkstra's shortest-path algorithm to determine a shortest-path tree to all subnets, with itself as the root node. Individual link costs are configured by the network administrator.
- The administrator might choose to set all link costs to 1, thus achieving minimum-hop routing, or might choose to set the link weights to be inversely proportional to link capacity in order to discourage traffic from using low-bandwidth links. OSPF does not mandate a policy for how link weights are set

- With OSPF, a router broadcasts routing information to all other routers in the autonomous system, not just to its neighboring routers.
- A router broadcasts link-state information whenever there is a change in a link's state (for example, a change in cost or a change in up/down status). It also broadcasts a link's state periodically (at least once every 30 minutes), even if the link's state has not changed.
- OSPF advertisements are contained in OSPF messages that are carried directly by IP, with an upper-layer protocol of 89 for OSPF. Thus, the OSPF protocol must itself implement functionality such as reliable message transfer and link-state broadcast.
- The OSPF protocol also checks that links are operational (via a HELLO message that is sent to an attached neighbor) and allows an OSPF router to obtain a neighboring router's database of network-wide link state.

Routing Among the ISPs: BGP

- To route a packet across multiple ASs, say from a smartphone in Timbuktu to a server in a datacenter in Silicon Valley, we need an inter-autonomous system routing protocol.
- Since an inter-AS routing protocol involves coordination among multiple ASs, communicating ASs must run the same inter-AS routing protocol.
- In fact, in the Internet, all ASs run the same inter-AS routing protocol, called the Border Gateway Protocol, more commonly known as BGP.
- BGP is arguably the most important of all the Internet protocols, as it is the protocol that glues the thousands of ISPs in the Internet together.
- In BGP, packets are not routed to a specific destination address, but instead to CIDRized prefixes, with each prefix representing a subnet or a collection of subnets.
- In the world of BGP, a destination may take the form 138.16.68/22, which for this example includes 1,024 IP addresses. Thus, a router's forwarding table will have entries of the form (x, I), where x is a prefix (such as 138.16.68/22) and I is an interface number for one of the router's interfaces.
- As an inter-AS routing protocol, BGP provides each router a means to:
    - Obtain prefix reachability information from neighboring ASs. In particular, BGP allows each subnet to advertise its existence to the rest of the Internet.
    - Determine the "best" routes to the prefixes. A router may learn about two or more different routes to a specific prefix. To determine the best route, the router will locally run a BGP route-selection procedure (using the prefix reachability information it obtained via neighboring routers). The best route will be determined based on policy as well as the reachability information.

Summary

- we’ve covered the data plane functions of the network layer—the perrouter functions that determine how packets arriving on one of a router’s input links are forwarded to one of that router’s output links.
- We covered IP forwarding.
- We also studied the IPv4.
- We learned that the control plane is the network-wide logic that controls how a datagram is forwarded among routers along an end-to-end path from the source host to the destination host.
- We learned that there are two broad approaches towards building a control plane: traditional per-router control (where a routing algorithm runs in each and every router and the routing component in the router communicates with the routing components in other routers) and software-defined networking (SDN) control (where a logically centralized controller computes and distributes the forwarding tables to be used by each and every router).
- We studied a fundamental routing algorithm for computing least cost paths in a graph (link-state routing)
- We took a look at OSPF and BGP Protocols.