Computer Networks

Part 4_II

Data Link Layer

Summary

- Overview
- Protocols: PPP, ATM, Ethernet
- Ethernet in more details

Data Link Layer

Overview

- **The first point to address is why the data link layer is needed at all?**
  - We have addressing in Layer 3, so why is this not sufficient to move a packet from one system to another
- **The answer** to this lies in the **separation of duties** performed by each layer, which provides for a wide variety of **disparate technologies**
  - It could be the case that a Layer 3 protocol such as IP could talk directly to the hardware layer
  - Imagine how many different IP protocol stacks would have to be written
  - Or the IP stack code would be enormously large to account for all the separate devices it would need to interact with
  - Any changes to the IP protocol would necessitate massive code changes for every conceivable type of hardware

Data Link Layer

Overview

- Benefits for separation of duties:
  - Layer 3 protocols usually have no knowledge of the underlying physical infrastructure
  - The data link layer hides the details of the interaction with the physical medium entirely from upper protocols such as IP
  - In theory, this would allow IP to run over any possible physical medium
  - This has led to the mantra, “**IP Over Everything**”

- In the real world, most data link layer protocols support only a very limited number of physical media.
  - Ethernet can be run on only a few carefully specified physical media

Data Link Layer

Overview

- Layer 2/Data Link networks can be classified broadly into three types:
  - Point-to-Point Networks
    - Point-to-point network protocols do not usually require source and destination addresses since they are established between two networking devices only
  - Circuit-Based Networks
    - Circuit-based networks create virtual circuits between different devices over a shared infrastructure
  - Shared Networks
    - Shared networks provide each device with a share of the underlying network medium such as a physical cable or a switch
- These Layer 2 frames (packets) usually consist of:
  - **A circuit identifier** in the case of circuit-based networks
  - **An address** that directs the packet to the required destination in the case of shared media
  - **A maximum transmission unit (MTU)** established between the source and receiving component. Data from higher layers is broken into fixed-length frames
  - **An error check** that is inserted by the source component and verified by the receiving component to verify data integrity on each data link segment

Data Link Layer

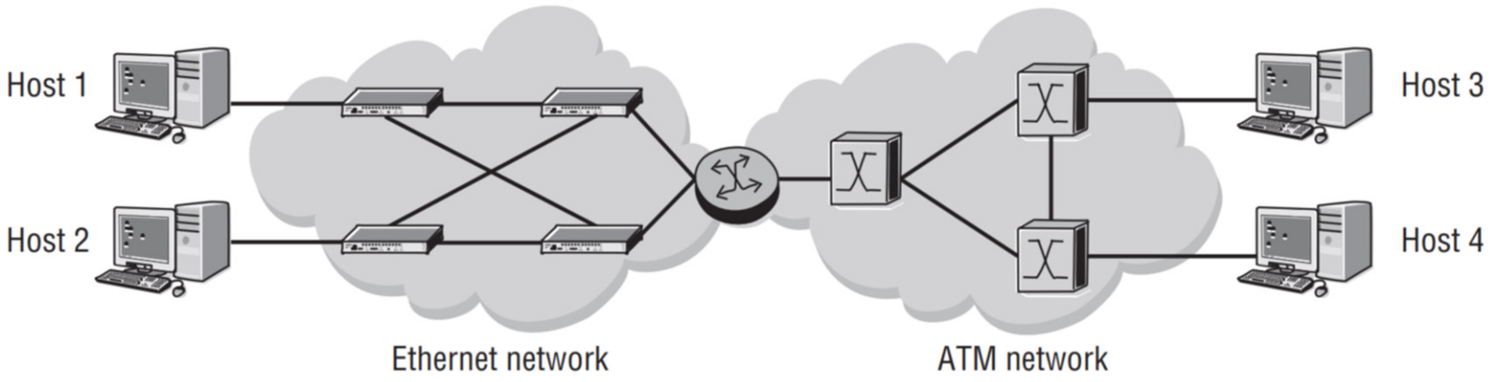
Overview

- Scope of the Data Link Layer
  - The Data Link frame remains intact while it traverses the Layer 2 devices in a particular IP subnet
  - If the IP packet needs to be routed to another subnet via an IP router, the original Data Link frame will be removed after it ingresses the IP router.
  - When forwarding the IP packet out from the appropriate port, the IP router constructs a new Data Link frame with a new header
  - This new Data Link header is used as the frame traverses the next subnet
  - This process continues until the destination host is reached
- The application data sent between two host stations can traverse several physically different networks
  - Each network has a different Data Link header and may even use different Data Link protocols that depend on the physical wire itself, for example, Ethernet, PPP, ATM, or Frame-Relay

Data Link Layer

Overview

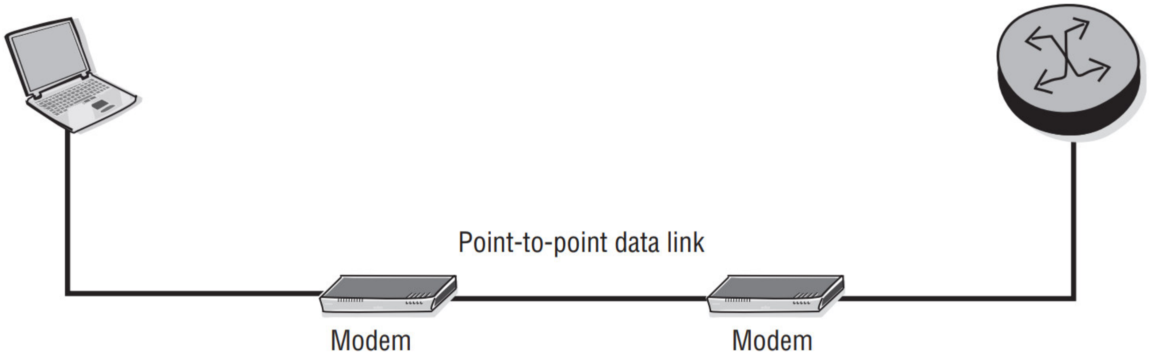
- The PCs on the left side of the Ethernet network do not require anything other than Ethernet Layer 2 framing to talk to each other
- The PCs on the right side of the network similarly require only ATM Layer 2 framing to talk to each other
- The Layer 2 networks are separated by routers, which are Layer 3 OSI devices. The PCs on the Ethernet network can only communicate with the PCs on the ATM network through Layer 3 addresses
- Note that the devices in the ATM cloud represent ATM switches and the devices in the Ethernet cloud represent Ethernet switches
- The device connecting the two clouds is a router



Data Link Layer

PPP: The Point-to-Point Protocol

- In the early days of the Internet, **point-to-point data links** allowed hosts to communicate with each other through the telephone network
- Older protocols such as **Serial Line IP (SLIP)** provided a simple mechanism for framing higher-layer applications for **transmission along serial lines**
- **Serial lines** allow for data to be sent in a **single-byte stream one after another** in “serial”
- **SLIP** was simple enough but could not control the characteristics of the connection
- **Point-to-Point Protocol (PPP)** provides advantages such as link control to negotiate the link characteristics



Typical configuration: a PC using a modem to connect to the Internet or any other dial-up network would use the PPP protocol

Data Link Layer

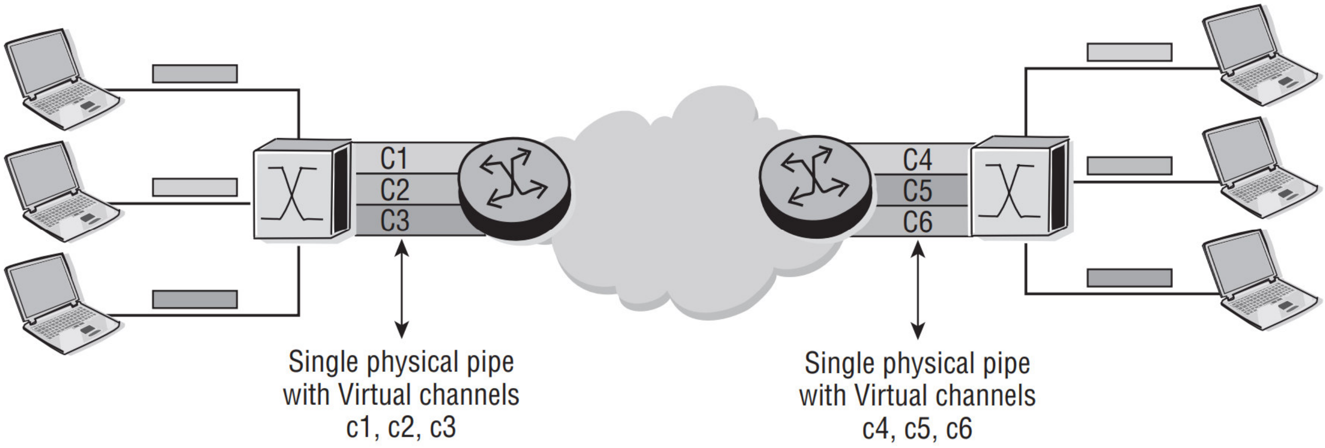
Circuit-Switching

- **Circuit-switched protocols** allow the transfer of user information as a unique set of packets identified by **Virtual Circuits (VC)**
  - A virtual connection may exist only for the duration of a particular network conversation
  - In some cases, the connection can exist for much longer than this
- The reason the **circuit is virtual** is that a VC can be configured over an infrastructure that can support multiple connections
  - This is beneficial because if a given path fails, it is very easy to reconfigure the VC to take another path through the network
  - It also allows for multiple VCs to share the same physical infrastructure

Data Link Layer

Circuit-Switching

- In the figure, the switch on the left accepts traffic from each host PC into a virtual circuit and switches to another virtual circuit when going to the router.
- The **virtual circuit number** is the same between the host PC and the switch, and between the switch and the router.
- Traffic from each PC is uniquely identified by a **virtual circuit number** at every hop. This allows for **many logical connections to be configured over a single physical connection** and is the predominant way that WAN connections are handled in modern networks.
- The two most predominant circuit-switching technologies are Frame-Relay and Asynchronous Transfer Mode (ATM).


C1
C2
C3
C4
C5
C6
Single physical pipe with Virtual channels c1, c2, c3
Single physical pipe with Virtual channels c4, c5, c6

Data Link Layer

Broadcast and Shared Access Data Links

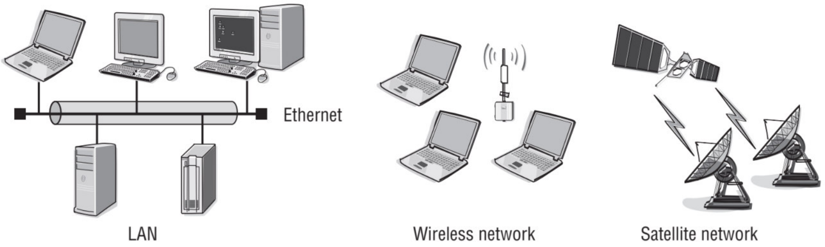
- Unlike point-to-point and circuit-switching networks, **broadcast networks** typically **use a shared media** to communicate to all the devices that are attached to that shared

- For data to be reliably delivered from the source to the destination, each of the devices on the shared media is identified by a **unique address**

- To transmit data reliably, the device on the shared media
  - Must compose the frame
  - Obtain control of the media
  - Then transmit the information

- Since the media is shared, it is possible for multiple stations to transmit their information simultaneously, resulting in a collision

- This collision causes data corruption
  - An algorithm needs to be followed to ensure a minimum number of collisions and also to ensure proper recovery from collisions



Examples of shared media technologies where every station receives the same information simultaneously.

Data Link Layer

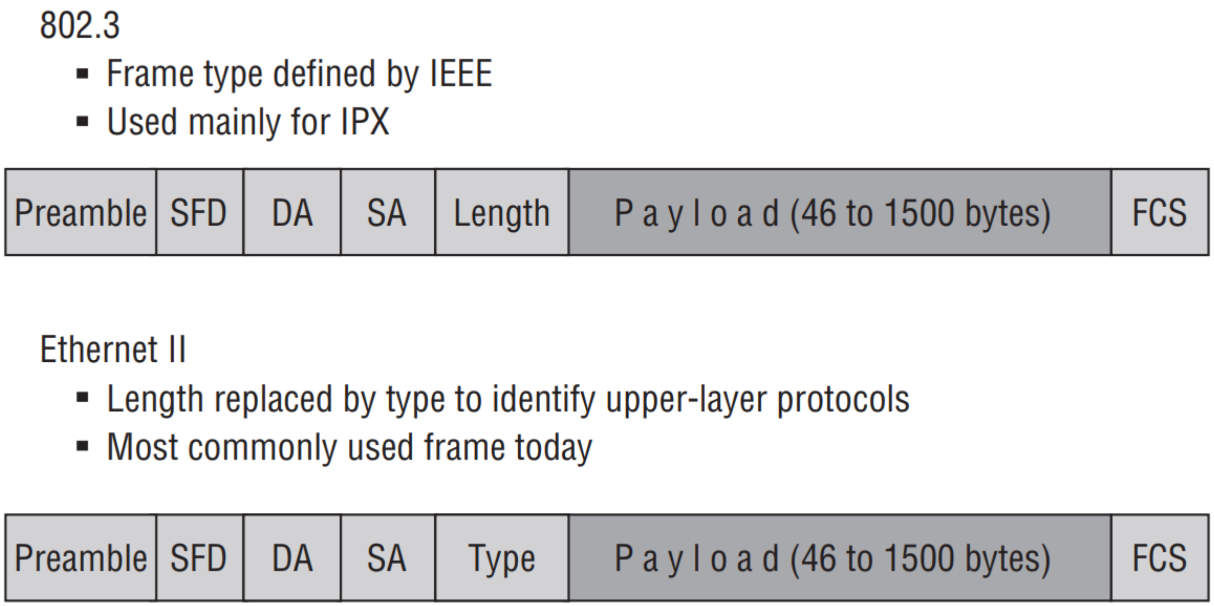
Ethernet Overview

- Ethernet is a broadcast technology that relies on a shared media for communication
  - It uses a "passive," wait-and-listen protocol called Carrier Sense Multiple Access with Collision Detection (CSMA/CD)
  - It uses data link layer addressing known as Media Access Control (MAC) addresses
  - it provides the ability to send a data frame to all devices on the network simultaneously (broadcasting)
- Two very similar but different standards were developed
  - Both of the standards are still in use, but the Ethernet II standard is by far the more widely accepted

Data Link Layer

Ethernet Overview

- The frame of each type is shown in figure below



802.3

- Frame type defined by IEEE
- Used mainly for IPX

Preamble	SFD	DA	SA	Length	Payload (46 to 1500 bytes)	FCS

Ethernet II

- Length replaced by type to identify upper-layer protocols
- Most commonly used frame today

Preamble	SFD	DA	SA	Type	Payload (46 to 1500 bytes)	FCS

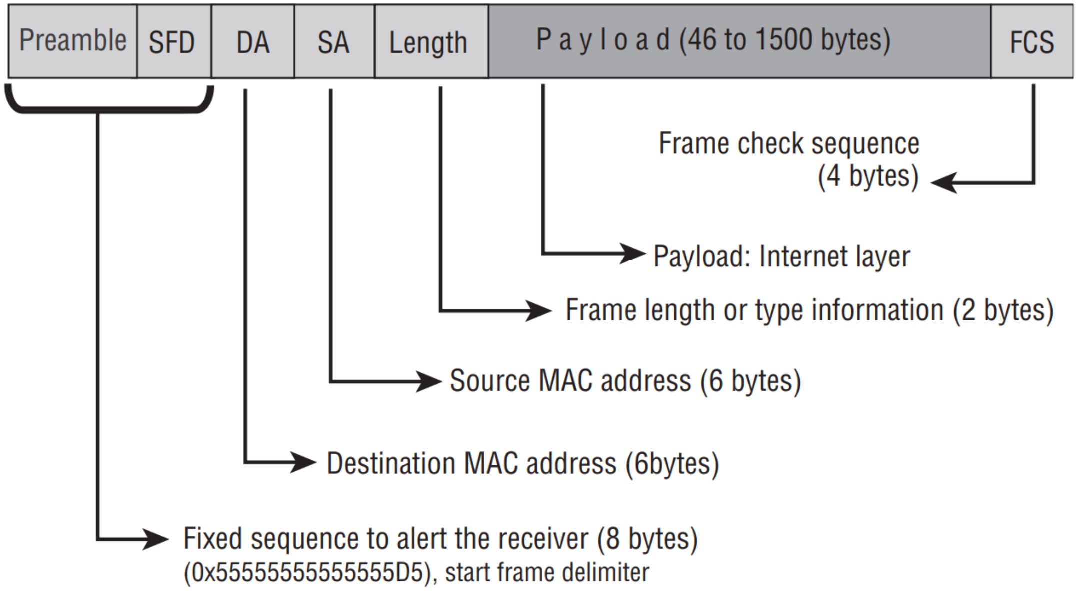
- The 16-bit field that follows the source address (SA) indicates whether the frame is Ethernet II or 802.3
  - If the value is 1,500 or less, the frame is treated as 802.3. If the value is greater than 1,500, the frame is treated as Ethernet II.

Data Link Layer

Ethernet Overview

- The original Ethernet standards defined the minimum frame size as 64 bytes and the maximum as 1,518 bytes
  - These numbers include all bytes from the destination MAC address field to the Frame Check Sequence (FCS) field
  - The preamble and the SFD fields are not included when quoting the size of a frame
  - The IEEE 802.3ac standard released in 1998 extended the maximum allowable frame size to 1,522 bytes to allow for a virtual LAN (VLAN) tag to be inserted into the Ethernet frame format
  - Gigabit Ethernet and 10 gigabit Ethernet ports may support jumbo frames that can be 9,000 bytes

General Ethernet Frame Format


Preamble
SFD
DA
SA
Length
Payload (46 to 1500 bytes)
FCS
Frame check sequence
(4 bytes)
Payload: Internet layer
Frame length or type information (2 bytes)
Source MAC address (6 bytes)
Destination MAC address (6bytes)
Fixed sequence to alert the receiver (8 bytes)
(0x55555555555555D5), start frame delimiter

Data Link Layer

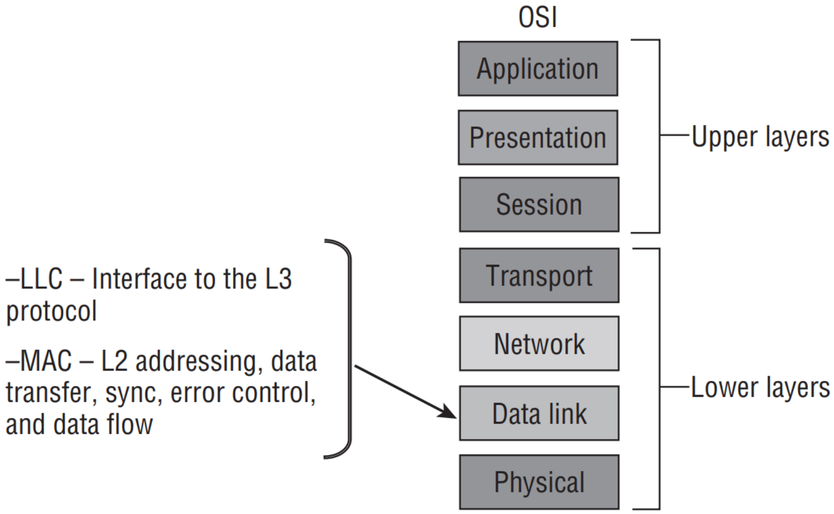
Ethernet Overview

- **Preamble**: a stream of bits used to allow the transmitter and receiver to synchronize their communication
  - The preamble is an alternating pattern of binary 56 ones and zeros
- **Start of Frame Delimiter:** this is always 10101011 and is used to indicate the beginning of the frame information
- **Length/Type:** The payload length or type field, also known as Ethertype
  - If the Ethernet frame is in the 802.3 format, this field is interpreted as length
  - If the Ethernet frame is in the Ethernet II or original DIX format, this field is interpreted as type, or Ethertype
  - The numeric value in this field determines whether the frame is an 802.3 frame or Ethernet II frame. If the value is less than 1,536 (hex value 0x600), it is an 802.3 frame. If the value is equal to or greater than 1,536, it is an Ethernet II frame
- **Data (a.k.a. Payload):** The data is inserted here
  - This is where the IP header and data are placed if you are running IP over Ethernet
- **Frame Check Sequence (FCS):** This is a part of the frame put in place to verify that the information each frame contains is not damaged during transmission
  - If a frame is corrupted during transmission, the FCS carried in the frame will not match with the recipient’s calculated FCS. Any frames that do not match the calculated FCS will be discarded

Data Link Layer

Ethernet Overview

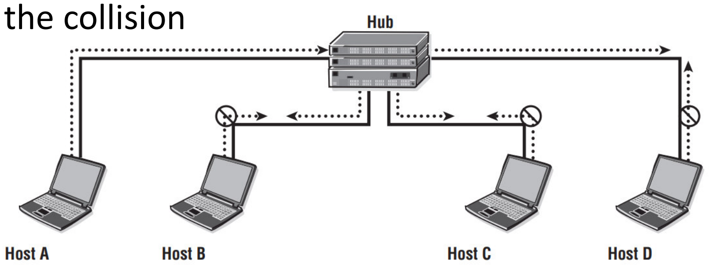
- Ethernet resides at the data link layer, and this layer can be subdivided further into two sublayers:
  - the Logical Link Control (LLC) and
  - the Media Access Control (MAC)
- The LLC interfaces between the network interface layer and the higher L3 protocol
  - may provide additional functions such as flow control or retransmission
- The MAC layer is responsible:
  - for determining the physical source and destination addresses for a particular frame and
  - for the synchronization of data transmission and
  - for error checking.
- The Ethernet II type of frame used for IP does not contain an LLC header and therefore does not provide any LLC functions
- IP uses Ethernet II simply for the transmission of frames on a shared media.


OSI
Application
Presentation
Session
Transport
Network
Data link
Physical
Upper layers
Lower layers
–LLC – Interface to the L3 protocol
–MAC – L2 addressing, data transfer, sync, error control, and data flow

Data Link Layer

Ethernet Transmission—CSMA/CD

- *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD)
  - Make sure that only one host can access the shared media at a time, and provide for a corrective mechanism in the event that two or more hosts try to talk simultaneously
  - **Carrier Sense (CS)**
    - Each Ethernet LAN-attached host continuously listens for traffic on the medium to determine when gaps between frame transmissions occur. In other words, “sense” (listen to) the “carrier” (the physical media)
  - **Multiple Access (MA)**
    - When the host senses that the carrier has no other host accessing the physical media, then it can begin transmitting
  - **Collision Detect (CD)**
    - If two or more hosts in the same CSMA/CD network or collision domain begin transmitting at approximately the same time, the bit streams from the transmitting hosts will interfere (collide) with each other
    - Each host must stop transmitting as soon as it has detected the collision and then must wait a random length of time as determined by a back-off algorithm before attempting to retransmit the frame
    - In this event, each transmitting host will transmit a 32-bit jam signal alerting all LAN-attached hosts of a collision before running the back-off algorithm. The purpose of the jam signal is to ensure that all hosts have received notice of the collision

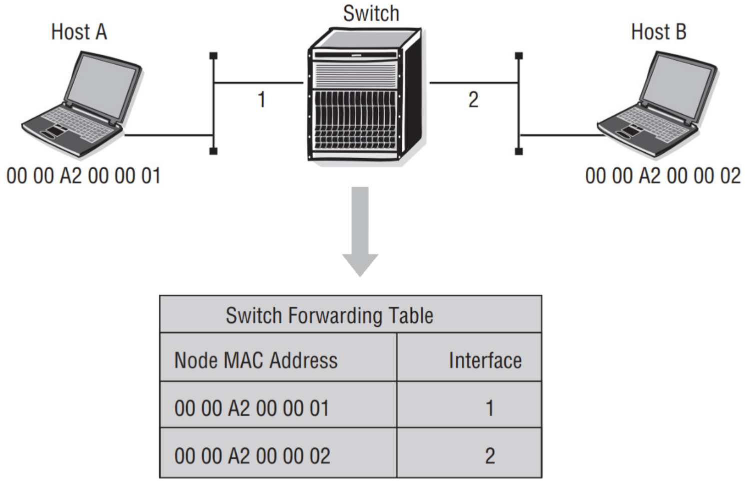


- All hosts constantly listen to the line.
- Host A transmits.
- Host B, C,and D listen to Host A and do not transmit.
- All hosts receive Host A’s message.

Data Link Layer

Ethernet Switching Operations

- The Ethernet switch will forward a frame only to the port that needs to receive it
- It performs this function by building a dynamic MAC address table (FDB - forwarding database) that matches MAC addresses to ports so that it “knows” which ports correspond to which MAC addresses
- When the switch receives an Ethernet frame:
  - It records the source MAC address and the interface on which it arrived
  - It looks at the destination MAC address of the frame, compares it to the entries in its MAC FDB, and then transmits the frame out of the appropriate interface
  - If no entry is found, the switch floods the frame out of all its interfaces except the interface on which the frame arrived



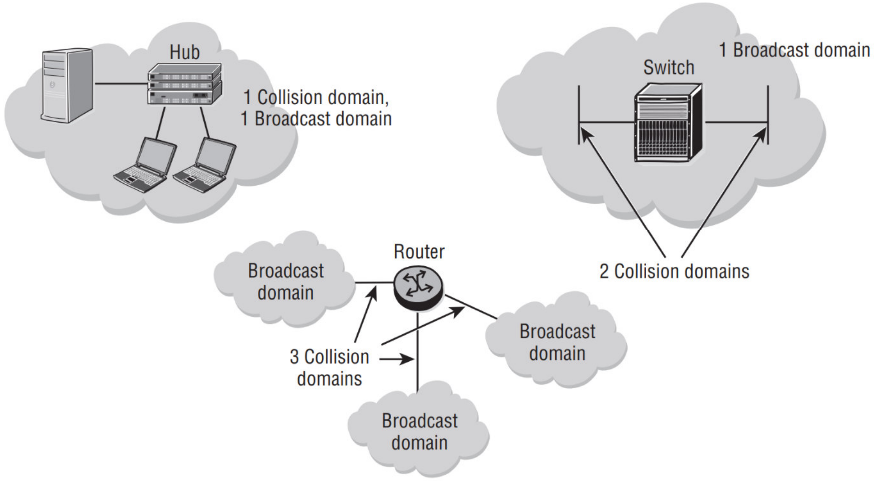
Switch Forwarding Table	
Node MAC Address	Interface
00 00 A2 00 00 01	1
00 00 A2 00 00 02	2

Switches build up their FDB table by recording the source address of frames as they enter each port on the switch

Data Link Layer

Ethernet Switching Operations

- Routers operate at Layer 3 of the OSI model, and so an Ethernet frame would not be forwarded across a router
- This boundary or domain that includes all the Ethernet switches contained by a router boundary is known as a **broadcast domain**
  - Within a broadcast domain, every Ethernet device will receive and process all broadcast packets
- In contrast to a broadcast domain, a **collision domain** exists between devices only within a single wire or hub
  - A collision domain is a group of Ethernet or Fast Ethernet devices in a CSMA/CD LAN that are connected by repeaters or hubs and that compete for access in the network
  - Only one device in the collision domain may transmit at any one time
- Devices on a hub are in a single collision domain, whereas each device on a switch has its own collision domain between the device and its individual port

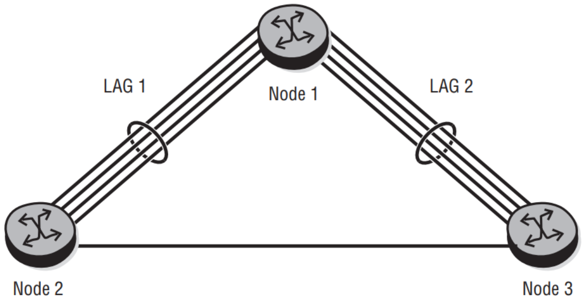


- Hubs provide no separation for collision or broadcast domains
- Switches provide collision domain separation
- Routers provide both collision and broadcast domain separation

Data Link Layer

Ethernet Link Redundancy: LAG

- There are two basic types of redundancy available with Ethernet networks: **link redundancy** and **path redundancy**
  - Link redundancy is provided via the **Link Aggregation Group (LAG)** protocol
  - Path redundancy is provided by the **Spanning Tree Protocol** (STP)
  - The primary difference between link redundancy and path redundancy is that the former does not provide redundancy in the event of a switch failure
  - A failure of a single or multiple links between LAG-connected switches would be survivable



- A LAG is based on the IEEE 802.3ad standard
- LAG allows you to aggregate multiple physical links between Ethernet devices so that they are functionally equivalent to a single logical link
  - All frames transmitted between the same source/destination MAC address pair (referred to as a **conversation**) will be transmitted across the same physical link in the bundle
- The primary benefits of LAG are that it increases the bandwidth available between two Ethernet devices by grouping several ports into one logical link
- The aggregation of multiple physical links allows for statistical load sharing and offers seamless redundancy

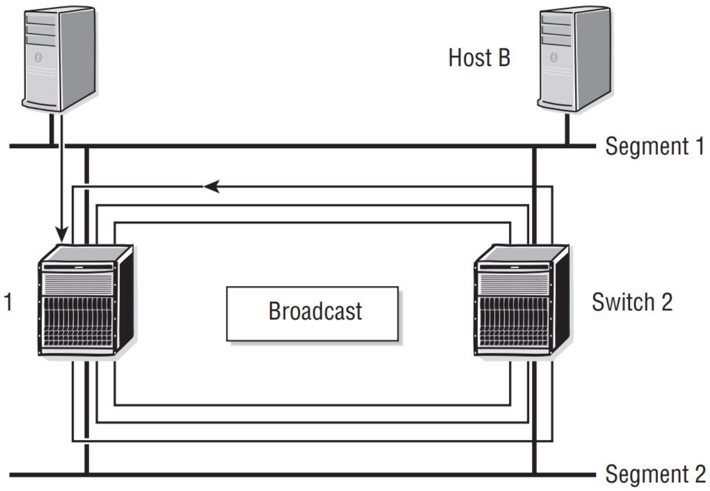
Data Link Layer

Ethernet Path Redundancy: STP

- LAG is a good solution for providing link redundancy between neighboring Ethernet devices
- If you require end-to-end path redundancy, LAG cannot provide this functionality
- Path redundancy is provided by the Spanning Tree Protocol (STP)
- There are some potential problems associated with providing path redundancy because of the nature of Ethernet switches
    - Providing redundant Ethernet switch paths can result in broadcast storms due to constant “looping” of Ethernet frames
    - A loop exists in a network when a frame or packet exits one interface on a device and then re-enters the device on a second interface.
    - It may also lead to FDB table instability as switches might see source addresses coming in on different interfaces
- In looping scenarios, it is often the case that the FDB table is unstable, resulting in excessive flooding.
    - Without STP, broadcast traffic may increase exponentially because, as the switch receives multiple copies of a frame, it further replicates each frame and transmits them out one or more ports on the switch

Data Link Layer

Ethernet Path Redundancy: STP

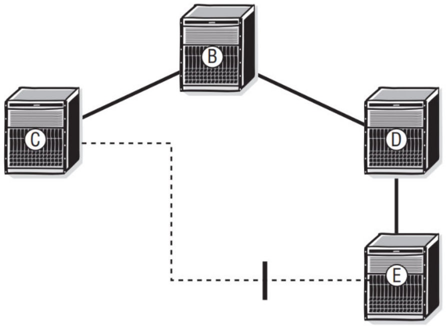


- Host A sends a frame with the destination MAC address of Host B
  - One copy of the frame is received by Host B and processed
- The original frame from Host A is also received by Switch 1
  - Switch 1 records the source MAC of Host A to be on Segment 1
  - Since Switch 1 does not know where Host B is, it replicates the frame and sends it out the port connected to Segment 2
- The original frame is also received by Switch 2 on Segment 1
  - Switch 2 also records the source MAC of Host A to be on Segment 1
  - Since Switch 2 does not know where Host B is, like Switch 1 it replicates the frame and sends it out the port connected to Segment 2
- Switch 2 receives the replicated frame from Switch 1 in Step 2 above via Segment 2
  - Switch 2 removes the existing entry for Host A in the MAC FDB and records that Host A belongs to the port attached to Segment 2
  - Switch 2 then replicates the frame and transmits it out the port attached to Segment 1, where it will be received by Switch 1 on Segment 1
- This process continues indefinitely as both Switch 1 and Switch 2 replicate the original frame from Host A onto Segments 1 and 2, causing excessive flooding and MAC FDB instability

Data Link Layer

Ethernet Path Redundancy: STP

- The Spanning Tree Protocol (STP) was developed to solve these instability and broadcast-storm issues
- STP is intended to prevent loops in an Ethernet switched network
  - It does this by selectively blocking ports to achieve a loop-free topology
  - It determines what ports it can put into a nonfunctioning state to prevent loops from occurring, while still allowing frames to reach every destination in the Ethernet network
- STP uses a root/branch/leaf model, which determines a single path to each leaf spanning the entire switched network
- The sole purpose of STP is to build an active loop-free topology (active in the sense that the ports that are blocked can change in response to changed network conditions)
- Spanning Tree topology can be thought of as a tree that includes the following components:
  - A root (a root bridge/switch)
  - Branches (LANs and designated bridges/switches)
  - Leaves (end nodes)

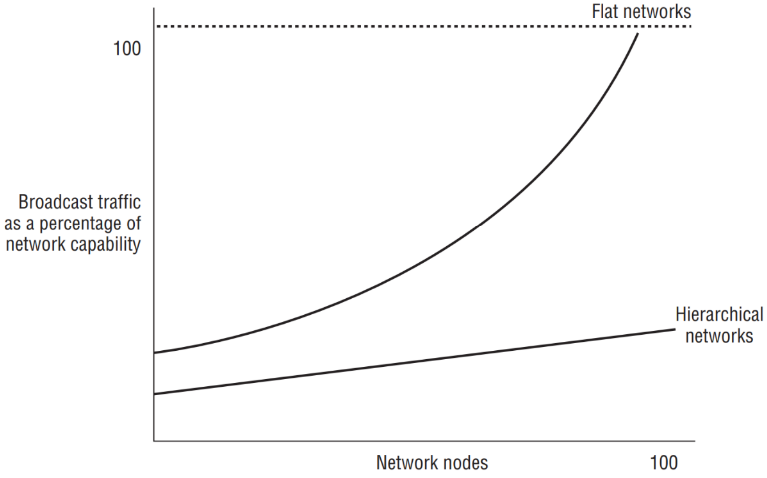


- STP will block the ports between Switches C and E, ensuring a loop-free topology in the switched network

Data Link Layer

Virtual LANs

- A virtual LAN (VLAN) is a mechanism that allows you to segregate devices, and their associated traffic from other devices and traffic
- There are two main reasons to use VLANs:
  - To decrease the amount of broadcast traffic
  - To increase the security of your network



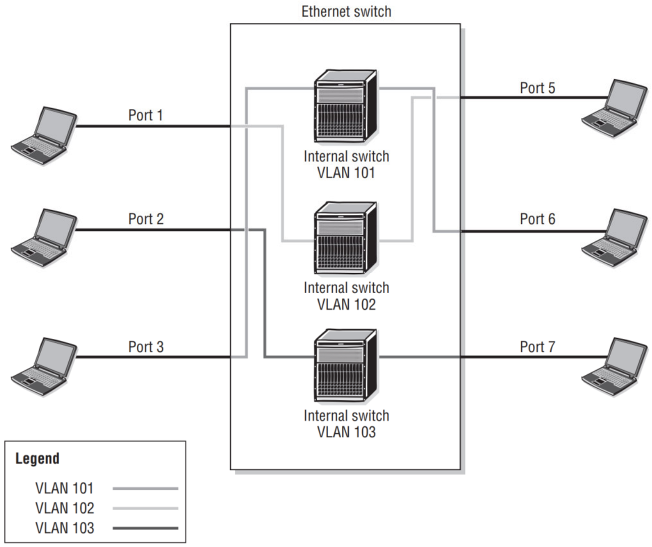
- As broadcasts increase on a flat network, they quickly consume all available network resources
- A flat network means a network without routers or without VLANs or both
- By segregating a group of devices to a particular VLAN, a switch will block broadcasts from devices in that VLAN to devices that are not in that VLAN

- VLANs also have the benefit of added security by separating the network into distinct logical networks.
  - Traffic in one VLAN is separated from another VLAN as if they were physically separate networks

Data Link Layer

Virtual LANs

- While in theory there are many ways to create VLANs, such as by MAC address, IP address, workstation names, and so on, in practice, these methods are very cumbersome
- The primary way that VLANs are created in modern networks is by physical port
- Each VLAN is identified by a VLAN ID (VID),
  - This is usually a number such as 100, 101, and the like
  - They can reside on only a single switch, or they can be distributed throughout the entire network on each switch
- You can think of a VLAN as a broadcast domain, and, in fact, that is exactly what it is



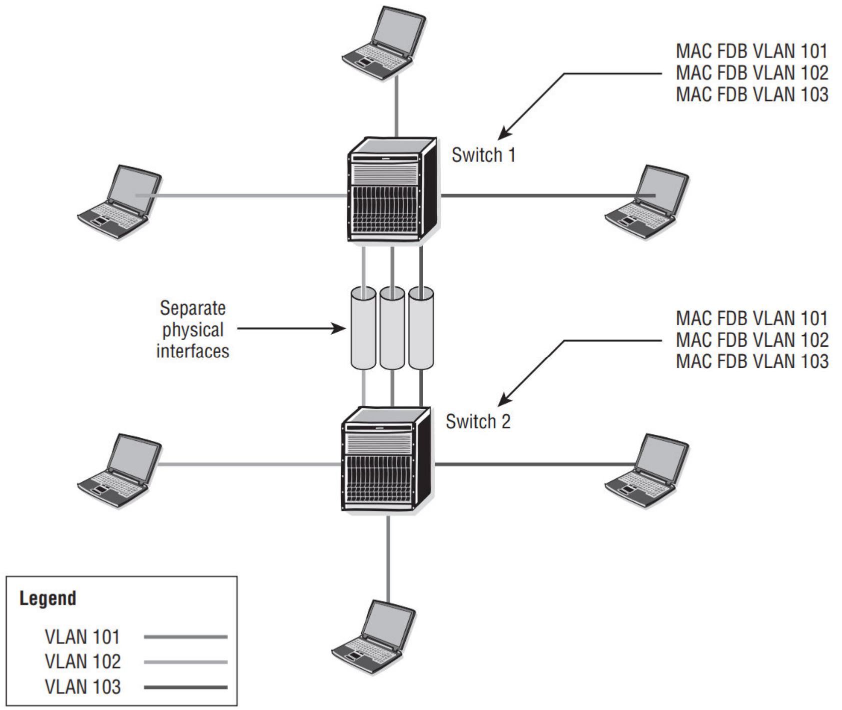
VLANs provide for logical separation of devices on the same physical switch

Data Link Layer

Virtual LANs

VLAN Trunking

- Considering VLANs that are shared across multiple switches
  - Frames ingressing a port in a particular VLAN will only be allowed to egress a port on the same VLAN, regardless of the switch it exits

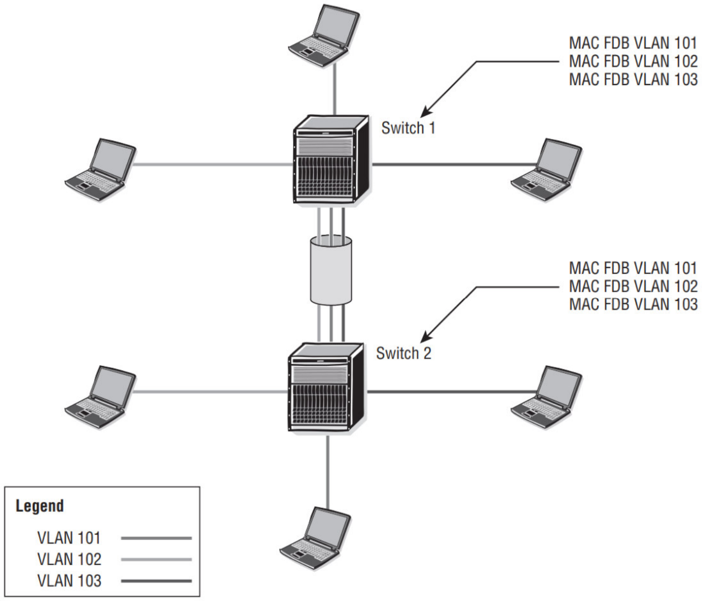


- In this case, there is a separate physical interswitch link for each VLAN
- That might be acceptable for two or three VLANs, but it is not a very scalable or practical solution
- This is where VLAN “trunking” comes into play

Data Link Layer

Virtual LANs - VLAN trunking

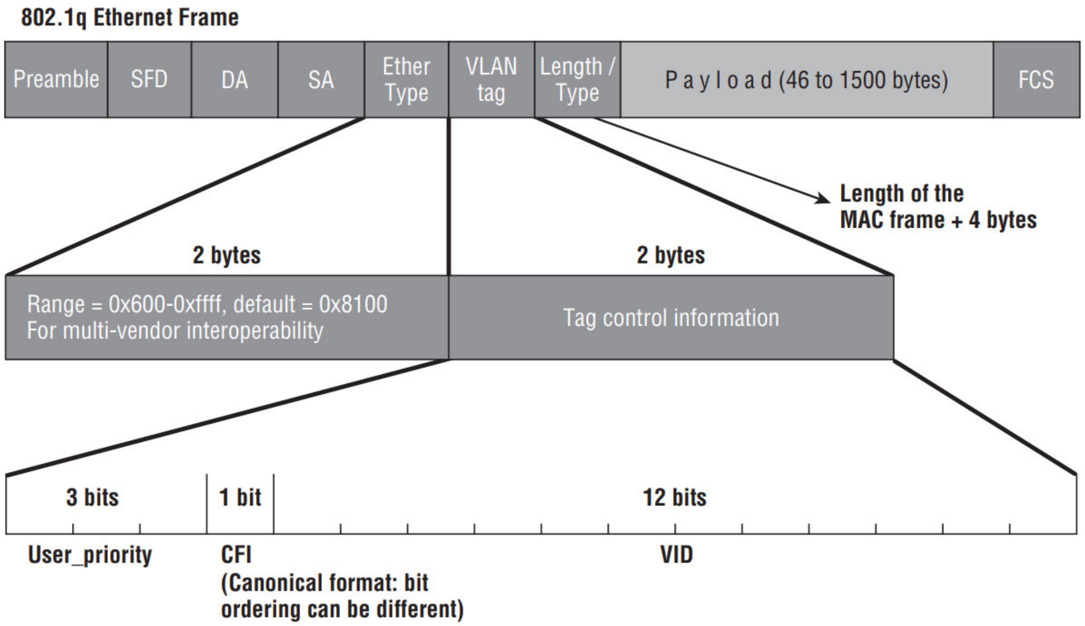
- When a frame is leaving a switch with another switch as its destination, the egress switch will tag the frames with a VID so that the ingress switch knows which VLAN the frame belongs to
  - The IEEE 802.1q standard governs the format of the assigned tag
  - The procedure works by inserting a 32-bit VLAN header into the Ethernet frame of all network traffic for a VLAN as it exits the egress switch
  - The VID uses 12 bits of the 32-bit VLAN header
  - The ingress switch then uses the VID to determine which FDB it will use to find the destination
  - After a frame reaches the destination switch port and before the frame is forwarded to the end destination, the VLAN header is removed



- There is a single VLAN trunk port between the switches that carries traffic for all VLANs by tagging the frames with the correct VID on egress to the other switch

Data Link Layer

Virtual LANs - VLAN trunking


802.1q Ethernet Frame
Preamble
SFD
DA
SA
Ether Type
VLAN tag
Length / Type
P a y l o a d (46 to 1500 bytes)
FCS
Length of the MAC frame + 4 bytes
2 bytes
2 bytes
Range = 0x600-0xffff, default = 0x8100
For multi-vendor interoperability
Tag control information
3 bits
1 bit
12 bits
User_priority
CFI
(Canonical format: bit ordering can be different)
VID

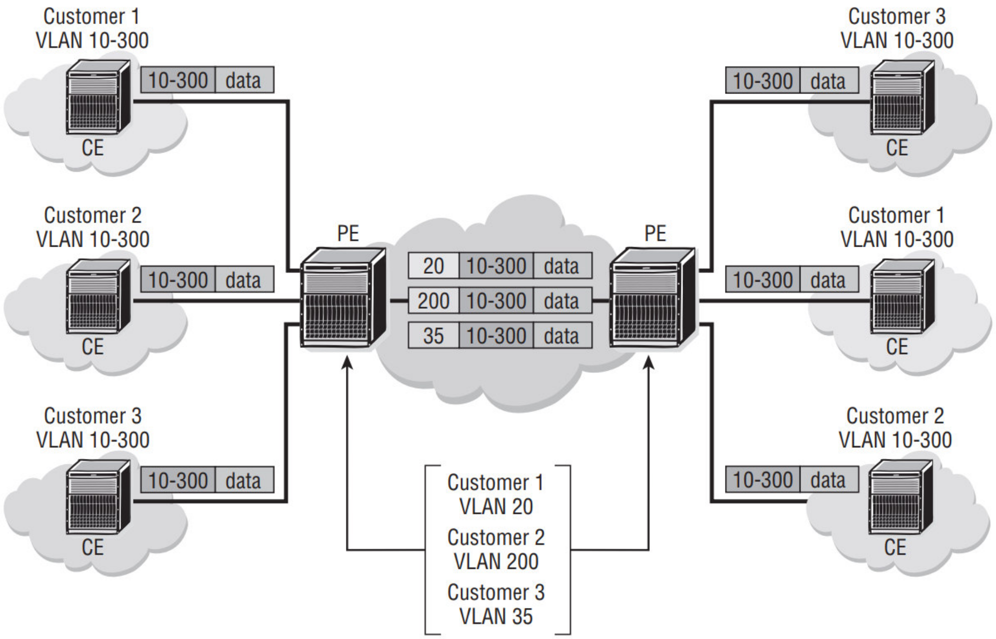
- **Priority Value (User Priority):** A 3-bit value that specifies a frame’s priority
- **CFI:** A single bit. A setting of 0 means that the MAC address information is in its simplest form.
- **VID**: A 12-bit value that identifies the VLAN that the frame belongs to. If the VID is 0, the tag header contains only priority information

- https://en.wikipedia.org/wiki/IEEE_802.1Q

Data Link Layer

Virtual LANs - VLAN trunking

- A restriction of Ethernet VLANs is the limited number of VIDs
  - Because VLANs 0 and 4095 are reserved, a Provider Edge (PE) router (connection to the customer) is really only capable of supporting 4094 VLANs - not a significant number if it is compared with the expanding rates of networks
  - While 4094 might seem sufficient, a single PE router might support hundreds or even thousands of customers, and the number of available VIDs can quickly evaporate
- One of the solutions to this restriction is VLAN stacking, also known as Q-in-Q



Data Link Layer

Virtual LANs

VLAN Trunking

- VLAN trunking provides efficient interswitch forwarding of VLAN frames
  - It allows a single Ethernet port to carry frames from multiple VLANs instead of the “one link per VLAN” approach
- The sharing of VLANs between switches is achieved by the insertion of a header or “tag” with a 12-bit VID
- A VID must be assigned for each VLAN
  - Assigning the same VID to VLANs on different connected switches can extend the VLAN (broadcast domain) across a network
- When a frame is leaving a switch with another switch as its destination, the egress switch will tag the frames with a VID so that the ingress switch knows which VLAN the frame belongs to
- The IEEE 802.1q standard governs the format of the assigned tag
  - The procedure works by inserting a 32-bit VLAN header into the Ethernet frame of all network traffic for a VLAN as it exits the egress switch
  - The VID uses 12 bits of the 32-bit VLAN header
  - The ingress switch then uses the VID to determine which FDB it will use to find the destination

Data Link Layer

Reliable Transmission

- **Link: https://book.systemsapproach.org/direct/reliable.html**

- CRC is used to detect errors
- Some error codes are strong enough to correct errors
- The overhead is typically too high
- Corrupted frames must be discarded
- A link-level protocol that wants to deliver frames reliably must recover from these discarded frames
- This is accomplished using a combination of two fundamental mechanisms:
  - Acknowledgements and
  - Timeouts

Data Link Layer

Reliable Transmission

- An **acknowledgement** (ACK for short) is a small control frame that a protocol sends back to its peer saying that it has received the earlier frame
  - A control frame is a frame with header only (no data)
- The receipt of an acknowledgement indicates to the sender of the original frame that its frame was successfully delivered
- If the sender does not receive an acknowledgment after a reasonable amount of time, then it retransmits the original frame
  - The action of waiting a reasonable amount of time is called a **timeout**
- The general strategy of using acknowledgements and timeouts to implement reliable delivery is sometimes called **Automatic Repeat reQuest (ARQ)**

Data Link Layer

Stop-and-Wait

- Idea of stop-and-wait algorithm is straightforward
  - After transmitting one frame, the sender waits for an acknowledgement before transmitting the next frame
  - If the acknowledgement does not arrive after a certain period of time, the sender times out and retransmits the original frame

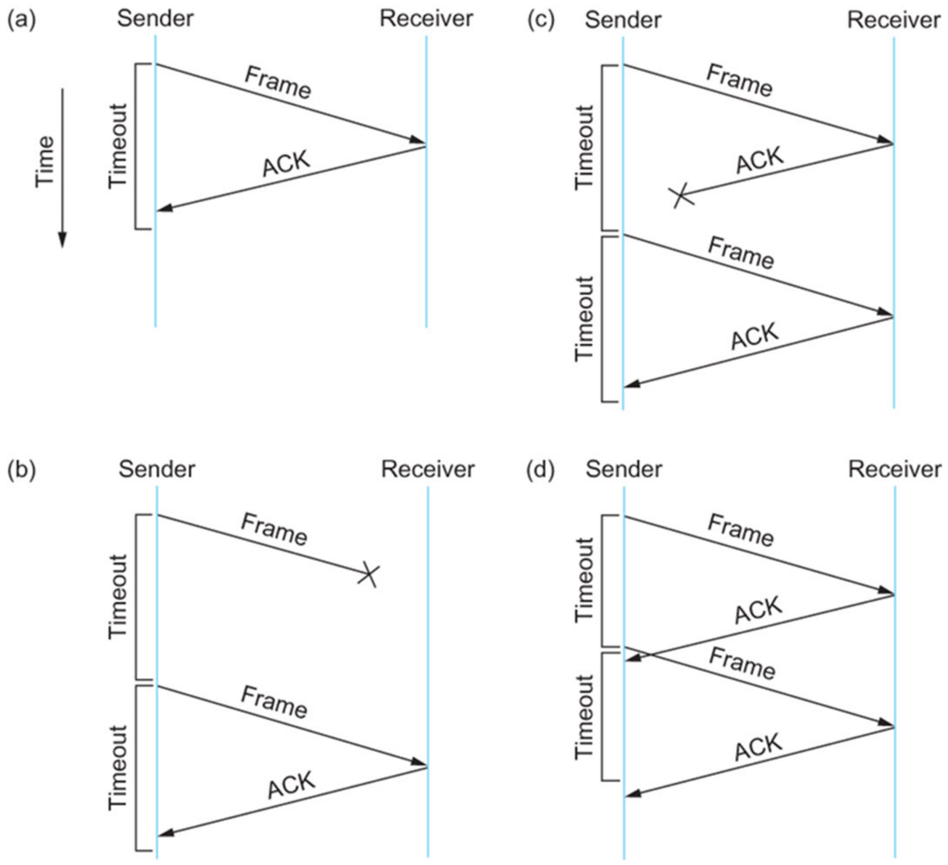
- Timeline showing four different scenarios for the stop-and-wait algorithm

(a) The ACK is received before the timer expires

(b) The original frame is lost

(c) The ACK is lost

(d) The timeout fires too soon

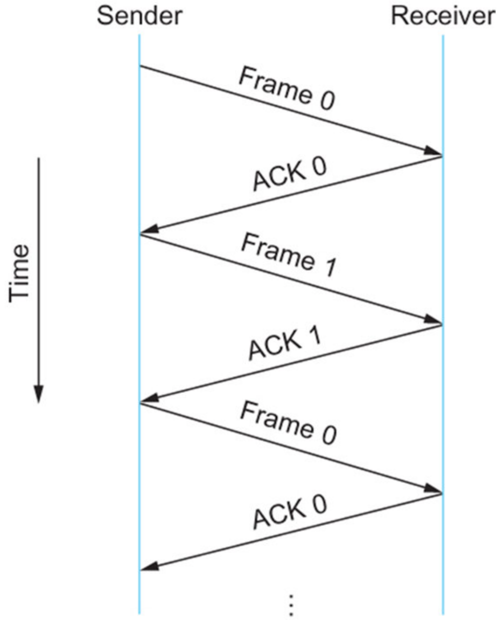


Data Link Layer

Stop-and-Wait

- If the acknowledgment is lost or delayed in arriving:
  - The sender times out and retransmits the original frame, but the receiver will think that it is the next frame since it has correctly received and acknowledged the first frame
  - As a result, duplicate copies of frames will be delivered

- How to solve this:
  - Use 1 bit sequence number (0 or 1)
  - When the sender retransmits frame 0, the receiver can determine that it is seeing a second copy of frame 0 rather than the first copy of frame 1 and therefore can ignore it (the receiver still acknowledges it, in case the first acknowledgement was lost)



Data Link Layer

Stop-and-Wait

- The sender has only one outstanding frame on the link at a time
  - This may be far below the link's capacity
- Consider a 1.5 Mbps link with a 45 ms RTT
  - The link has a **delay × bandwidth** product of 67.5 Kb or approximately 8 KB
  - Since the sender can send only one frame per RTT and assuming a frame size of 1 KB
  - Maximum Sending rate is

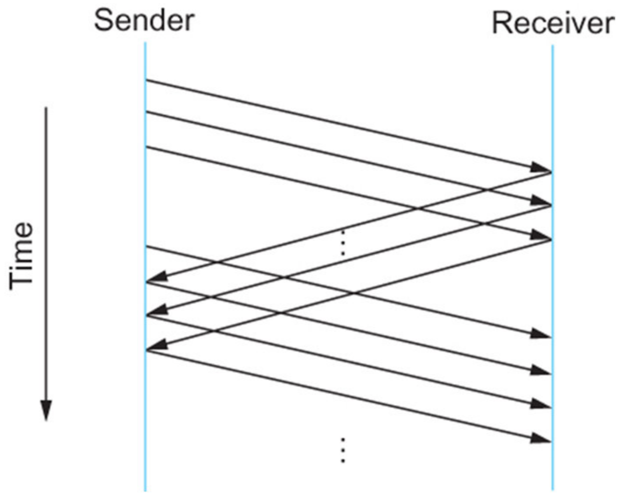
    Bits per frame / Time per frame = 1024 × 8 / 0.045 = 182 Kbps

    or about one-eighth of the link's capacity
  - To use the link fully, then sender should transmit up to eight frames before having to wait for an acknowledgement

Data Link Layer

Sliding Window

- The significance of the **delay × bandwidth** product is that:
  - It represents the amount of data that could be in transit
  - We would like to be able to send this much data without waiting for the first acknowledgment
  - The principle at work here is often referred to as keeping the pipe full
  - The algorithms presented in the following section do exactly this

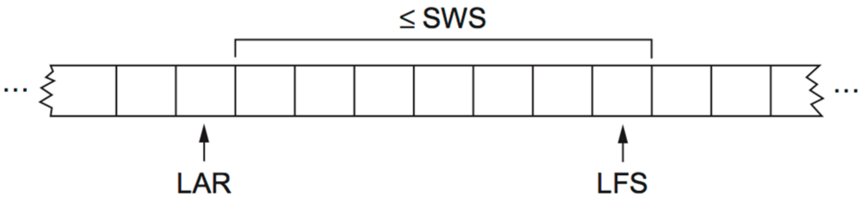


Data Link Layer

Sliding Window

- The sliding window algorithm works as follows
- Sender assigns a sequence number denoted as **SeqNum** to each frame
  - Assume it can grow infinitely large
- Sender maintains three variables:
  - **Sending Window Size (SWS)**
    - Upper bound on the number of outstanding (unacknowledged) frames that the sender can transmit
  - **Last Acknowledgement Received (LAR)**
    - Sequence number of the last acknowledgement received
  - **Last Frame Sent (LFS)**
    - Sequence number of the last frame sent
- Sender also maintains the following invariant

$$LFS – LAR \leq SWS$$



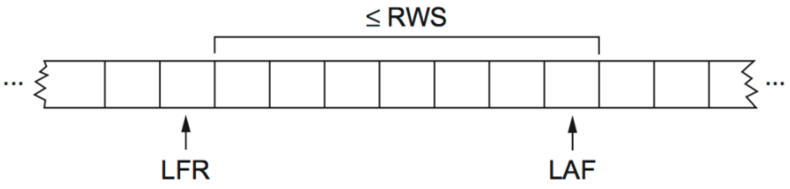
Sliding Window on Sender

- When an acknowledgement arrives
  - the sender moves LAR to right, thereby allowing the sender to transmit another frame
- Also, the sender associates a timer with each frame it transmits
  - It retransmits the frame if the timer expires before the ACK is received
- Note that the sender has to be willing to buffer up to SWS frames (WHY?)

Data Link Layer

Sliding Window

- Receiver maintains three variables
  - **Receiving Window Size (RWS)**
    - Upper bound on the number of out-of-order frames that the receiver is willing to accept
  - **Largest Acceptable Frame (LAF)**
    - Sequence number of the largest acceptable frame
  - **Last Frame Received (LFR)**
    - Sequence number of the last frame received



Sliding Window on Receiver

- Receiver also maintains the following invariant

  LAF – LFR ≤ RWS

- When a frame with sequence number SeqNum arrives:
  - If SeqNum ≤ LFR or SeqNum > LAF
    - Discard it (the frame is outside the receiver window)
  - If LFR < SeqNum ≤ LAF
    - Accept it
    - Now the receiver needs to decide whether or not to send an ACK
      - Let SeqNumToAck
        - The largest sequence number not yet acknowledged, such that all frames with sequence number less than or equal to SeqNumToAck have been received
      - The receiver acknowledges the receipt of SeqNumToAck even if high-numbered packets have been received
        - This acknowledgement is said to be cumulative.
      - The receiver then sets

        LFR = SeqNumToAck and adjusts

        LAF = LFR + RWS

Data Link Layer

Sliding Window

- For example, suppose LFR = 5 and RWS = 4
  - (i.e. the last ACK that the receiver sent was for seq. no. 5)
  - LAF = 9
- If frames 7 and 8 arrive, they will be buffered because they are within the receiver window
  - But no ACK will be sent since frame 6 is yet to arrive
  - Frames 7 and 8 are out of order
- Frame 6 arrives (it is late because it was lost first time and had to be retransmitted)
  - Now Receiver Acknowledges Frame 8
  - and bumps LFR to 8
  - and LAF to 12

Data Link Layer

Sliding Window

- When timeout occurs, the amount of data in transit decreases
    - Since the sender is unable to advance its window
- When the packet loss occurs, this scheme is no longer keeping the pipe full
    - The longer it takes to notice that a packet loss has occurred, the more severe the problem becomes
- How to improve this
    - *Negative Acknowledgement (NAK)*
    - *Additional Acknowledgement*
    - *Selective Acknowledgement*

Data Link Layer

Sliding Window

- *Negative Acknowledgement (NAK)*
  - Receiver sends NAK for frame 6 when frame 7 arrive (in the previous example)
  - However, this is unnecessary since sender's timeout mechanism will be sufficient to catch the situation
- *Additional Acknowledgement*
  - Receiver sends additional ACK for frame 5 when frame 7 arrives
  - Sender uses duplicate ACK as a clue for frame loss
- *Selective Acknowledgement*
  - Receiver will acknowledge exactly those frames it has received, rather than the highest number frames
  - Receiver will acknowledge frames 7 and 8
  - Sender knows frame 6 is lost
  - Sender can keep the pipe full (additional complexity)

Data Link Layer

Sliding Window

- Sequence Number serves three different roles
    - Reliable
    - Preserve the order
        - Each frame has a sequence number
        - The receiver makes sure that it does not pass a frame up to the next higher-level protocol until it has already passed up all frames with a smaller sequence number
    - Frame control
        - Receiver is able to throttle the sender
        - Keeps the sender from overrunning the receiver
        - From transmitting more data than the receiver is able to process