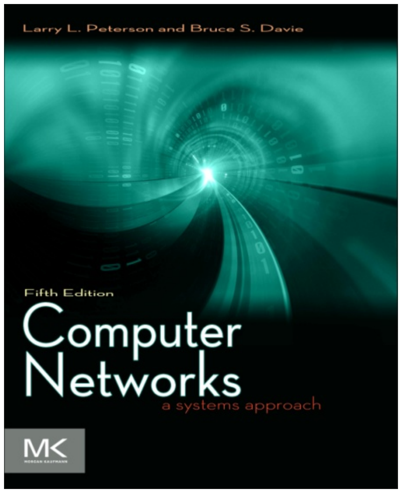


Chapter 2 (Subset of topics)

Getting Connected

- Framing
- Error Detection. Reliable Transmission
- Sharing: Ethernet and Multiple Access Networks, Wireless Networks

Framing, Reliability, Sharing

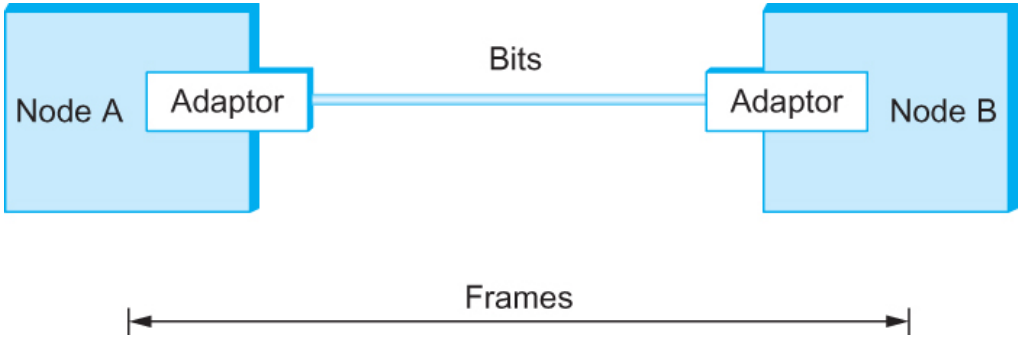
- Delineating the sequence of bits transmitted over the link into complete messages that can be delivered to the end node

- Techniques to detect transmission errors and take the appropriate action (conceptually same as RPC)
- Making the links reliable in spite of transmission problems

- Media Access Control Problem
- Carrier Sense Multiple Access (CSMA) networks
- Wireless Networks with different available technologies and protocol

Framing

- We are focusing on packet-switched networks, which means that blocks of data (called *frames* at this level), not bit streams, are exchanged between nodes.
- It is the network adaptor that enables the nodes to exchange frames.



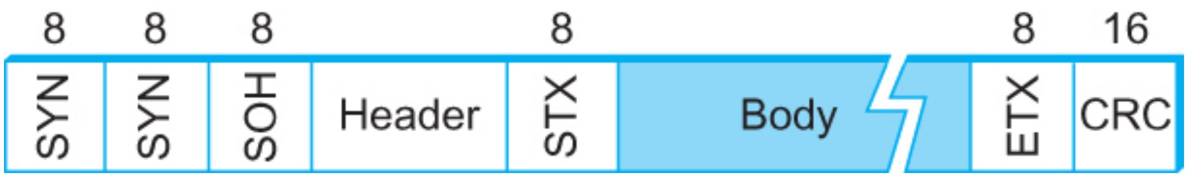
Bits flow between adaptors, frames between hosts

Framing

- When node A wishes to transmit a frame to node B, it tells its adaptor to transmit a frame from the node's memory. This results in a sequence of bits being sent over the link.
- The adaptor on node B then collects together the sequence of bits arriving on the link and deposits the corresponding frame in B's memory.
- Recognizing exactly what set of bits constitute a frame—that is, determining where the frame begins and ends—is the central function of the adaptor

Framing: Byte oriented

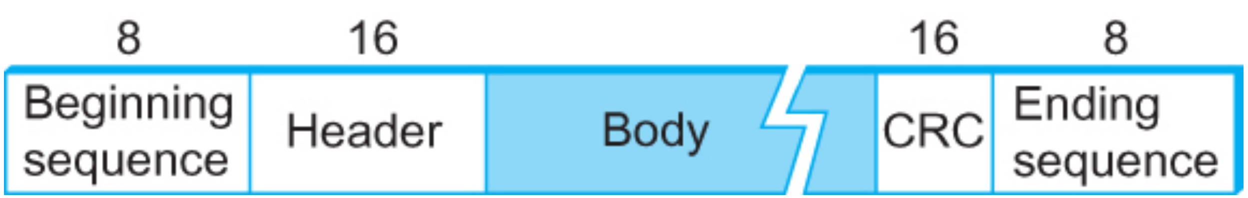
- BISYNC – original Sentinel Approach (late 60s, IBM)
  - Frames transmitted beginning with leftmost field
  - Beginning of a frame is denoted by sending a special SYN (synchronize) character
  - Data portion of the frame is contained between special sentinel character STX (start of text) and ETX (end of text)
  - SOH : Start of Header
  - DLE : Data Link Escape
  - CRC: Cyclic Redundancy Check



BISYNC Frame Format

Framing: Bit-oriented Protocol

- HDLC : High Level Data Link Control Beginning and Ending Sequences: 0 1 1 1 1 1 1 0
- Sender: any time five consecutive 1's have been transmitted from the body of the message sender inserts 0 before transmitting the next bit (bit stuffing)
- Receiver: 5 consecutive 1's received:
  - Next bit 0 : Stuffed, so discard it
  - Next bit 1 : Either End of the frame marker, Or Error has been introduced in bitstream
  - If 0 (01111110) → End of the frame marker
  - If 1 (01111111) → Error, discard the whole frame



HDLC Frame Format

Error Detection

- Bit errors are introduced into frames
  - Because of electrical interference and thermal noises
- Common technique for detecting transmission error
  - CRC (Cyclic Redundancy Check)
    - Used in HDLC, CSMA/CD
  - Other approaches
    - Checksum (IP)

Error Detection

- Basic Idea of Error Detection
  - To add redundant information to a frame that can be used to determine if errors have been introduced
  - Imagine (Extreme Case)
    - Transmitting two complete copies of data
      - Identical → No error
      - Differ → Error
      - Poor Scheme ???
        - n bit message, n bit redundant information
        - Error can go undetected
    - In general, we can provide strong error detection technique efficiently
      - k redundant bits, n bits message, k << n
      - In Ethernet, a frame carrying up to 12,000 bits of data requires only 32-bit CRC

Error Detection

- Extra bits are redundant
  - They add no new information to the message
  - Derived from the original message using some algorithm
  - Both the sender and receiver know the algorithm
  - IP weak checksum (sum of ‘16 bit’ words)
  - Link Level: CRC (see book/wikipedia for details as interested)

m	r

Sender

m	r

Receiver

Receiver computes *r* using *m,* If they match, no error

Error Correction

- Need to: Detect Error, Correct/Recover from Error
- Two approaches when the recipient detects an error
  - Using error correction algorithm, the receiver reconstructs the message
    - The overhead is often considered too high for wired links…more relevant to wireless…some errors go uncorrected
    - Corrupt frames must be discarded.
  - “Notify” the sender that the message was corrupted, so the sender can send again.
    - If the error is rare, then the retransmitted message will be error-free

Reliable Transmission

- A link-level protocol that wants to deliver frames reliably must recover from discarded frames.
- Combination of two fundamental mechanisms
- 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.

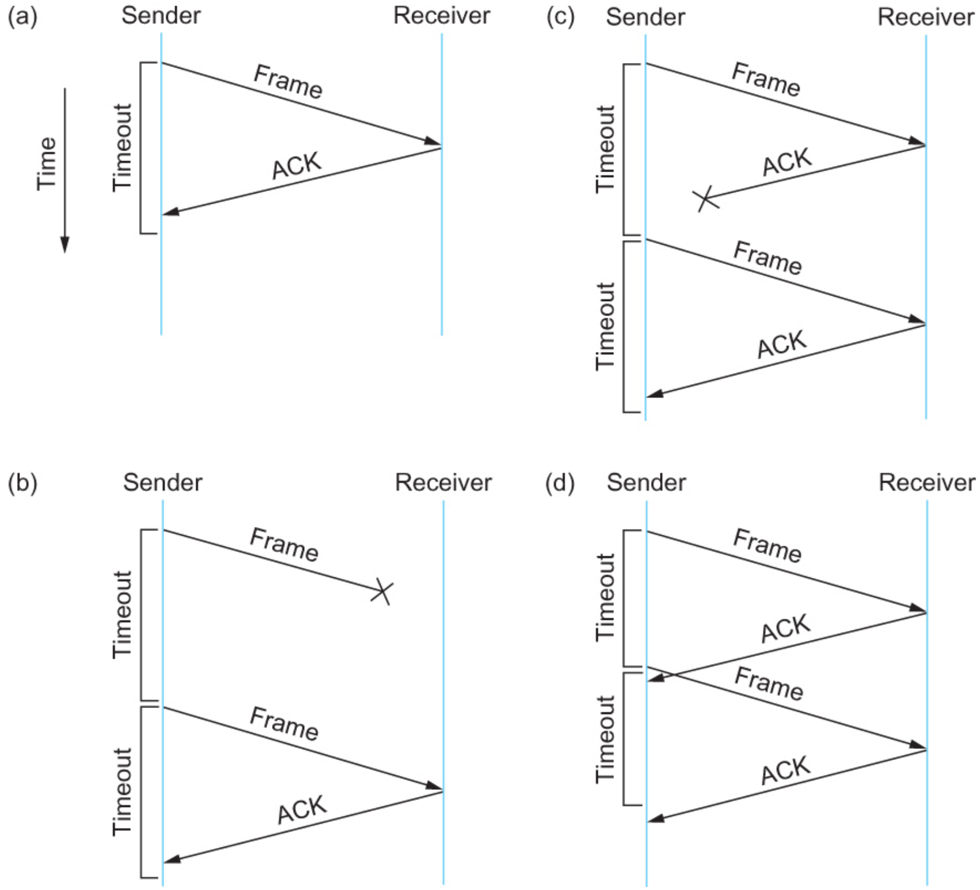
Reliable Transmission

- 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).
- Build up the concept so you understand why we don't use the simplest form…

Stop and Wait Protocol

- Idea of stop-and-wait protocol 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

Stop and Wait Protocol

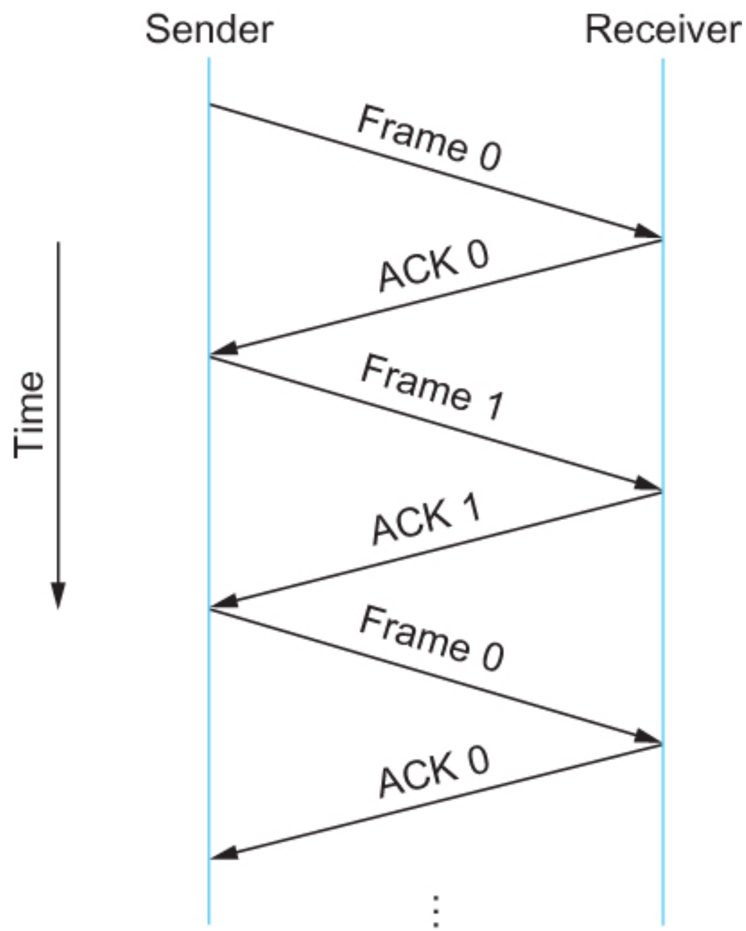


Timeline showing four different scenarios for the stop-and-wait algorithm.
(a) The ACK is received before the timer expires; (b) the original frame is lost; (c) the ACK is lost; (d) the timeout fires too soon

Stop and Wait Protocol

- 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
  Same conceptual issue as with TCP, RPC
  - 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)

Stop and Wait Protocol



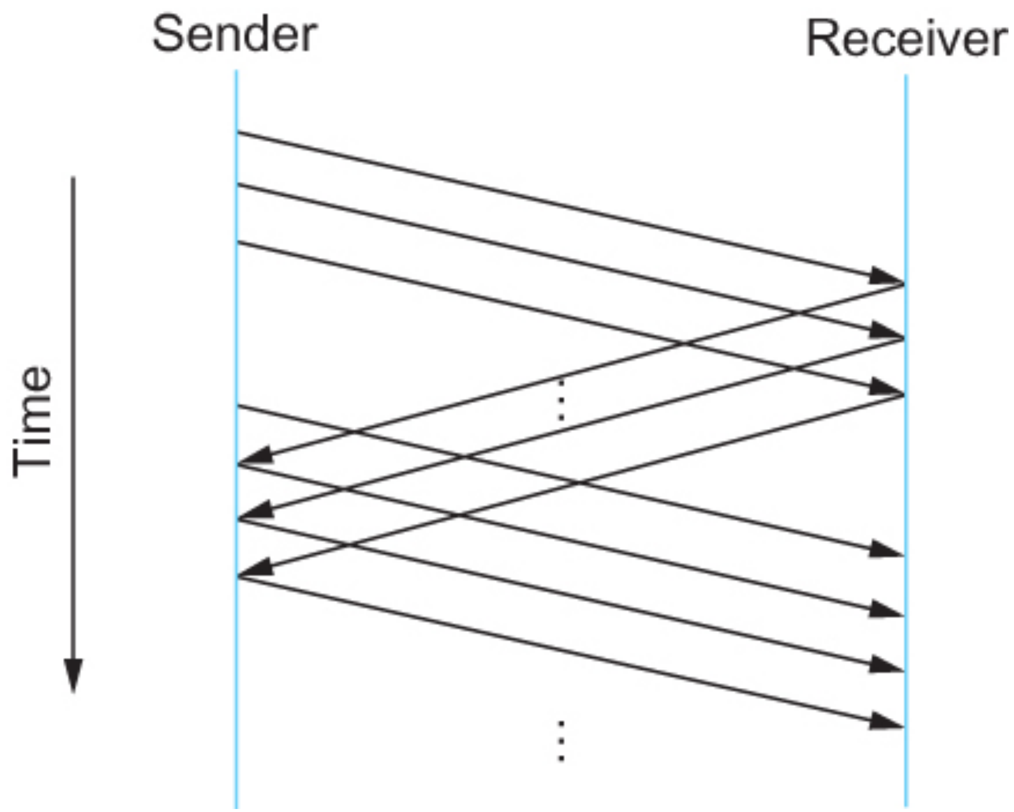
Timeline for stop-and-wait with 1-bit sequence number

Sender Stop and Wait Protocol -- Limitations

- The sender has only one outstanding frame on the link at a time
  - This may be far below the link's capacity
- Consider a slow 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
    - 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
  - QUIZ QUESTION

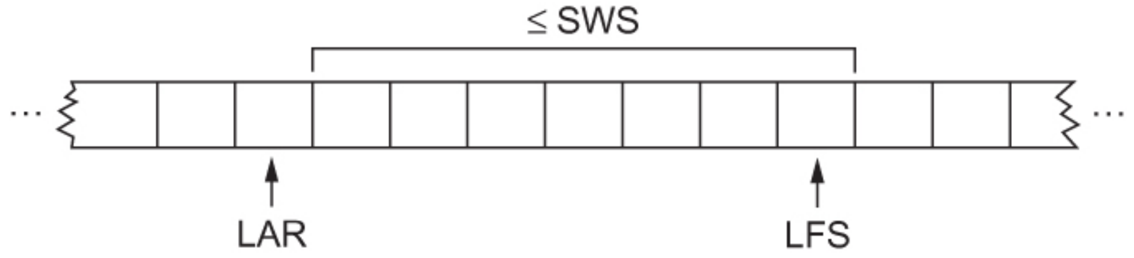
Sliding Window Protocol: Parallelism



Timeline for Sliding Window Protocol

Sender Sliding Window Protocol

- Sender assigns a sequence number denoted as SeqNum to each frame
- 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 maintains invariant: LFS – LAR ≤ SWS

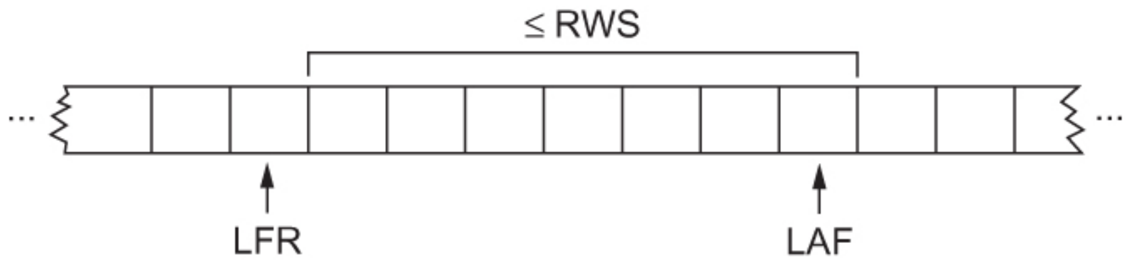


Sender Sliding Window Protocol

- 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

Receiver Sliding Window Protocol

- 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
- Receiver maintains invariant: LAF – LFR ≤ RWS



Receiver Sliding Window Protocol

- 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
  - Decide whether or not to send an ACK
    - Let SeqNumToAck Denote 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 higher-numbered packets have been received
    - This acknowledgement is said to be cumulative.
  - The receiver then sets
    - LFR = SeqNumToAck and adjusts
    - LAF = LFR + RWS

Sliding Window Protocol Inefficiencies

- 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 -- Will discuss nuances in context of TCP in future lecture
    - Negative Acknowledgement (NAK)
    - Additional Acknowledgement
    - Selective Acknowledgement

NAKs, Duplicate Acks, Selective Acks…

- 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)

Sequence numbers

- How to distinguish between different incarnations of the same sequence number?
  - Number of possible sequence number must be larger than the number of outstanding frames allowed
    - Stop and Wait: One outstanding frame
      - 2 distinct sequence number (0 and 1)
    - Let MaxSeqNum be the number of available sequence numbers
    - SWS + 1 ≤ MaxSeqNum
      - Is this sufficient?

Sequence number space

SWS + 1 ≤ MaxSeqNum

- Is this sufficient?

- Depends on RWS
- If RWS = 1, then sufficient
- If RWS = SWS, then not good enough

- For example, we have eight sequence numbers
  0, 1, 2, 3, 4, 5, 6, 7
  RWS = SWS = 7
  Sender sends 0, 1, …, 6
  Receiver receives 0, 1, … ,6
  Receiver acknowledges 0, 1, …, 6
  ACK (0, 1, …, 6) are lost
  Sender retransmits 0, 1, …, 6
  Receiver is expecting 7, 0, …., 5

- To avoid this, If RWS = SWS, SWS < (MaxSeqNum + 1)/2

Sharing Links

- Ethernet
- Wifi

Ethernet

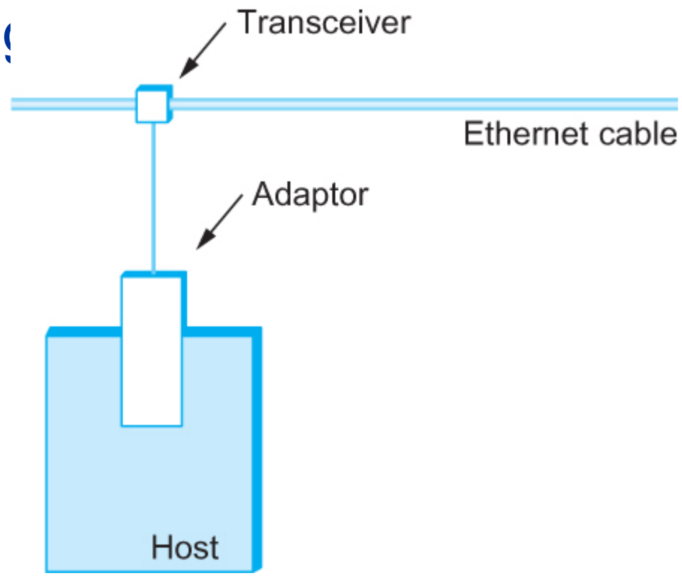
- Most successful local area networking technology of last 30 years.
- Developed in the mid-1970s by researchers at the Xerox Palo Alto Research Centers (PARC).
- Uses CSMA/CD technology
  - Carrier Sense Multiple Access with Collision Detection.
  - A set of nodes send and receive frames over a shared link.
  - Carrier sense means that all nodes can distinguish between an idle and a busy link.
  - Collision detection means that a node listens as it transmits and can therefore detect when a frame it is transmitting has collided with a frame transmitted by another node.
- Interesting design lessons for higher level distributed systems (pub/sub bus)

Ethernet -- Origins

- Uses ALOHA (packet radio network) as the root protocol
  - Developed at the University of Hawaii to support communication across the Hawaiian Islands (early ‘70s).
  - For ALOHA the medium was atmosphere, for Ethernet the medium is a coax cable.
- DEC and Intel joined Xerox to define a 10-Mbps Ethernet standard in 1978.
- This standard formed the basis for IEEE standard 802.3
- 802.3 was later extended to include a 100-Mbps Fast Ethernet and a 1000-Mbps Gigabit Ethernet.

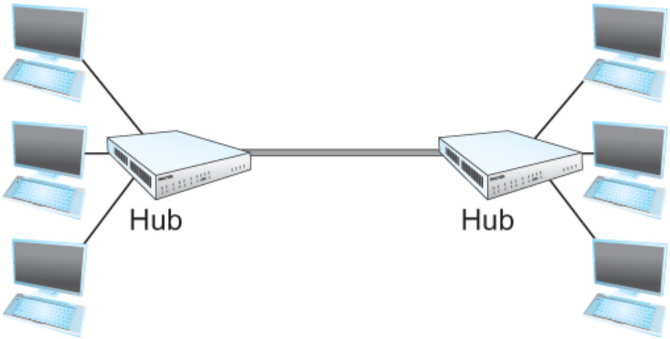
Ethernet -- hardware

- Ethernet segment – original was coaxial cable of up to 500 m.
  - Up to 5 segments can be joined by *repeaters* -- total reach 2500 m.
  - Later thinner cable known as 10Base2 up to 200 m; Twisted pair 10Base1 up to 100 m (used with Ethernet Hubs that xmit onto all links)
- Hosts connect to an Ethernet segment by tapping into it.
- A transceiver (a small device directly attached to the tap) detects when the line is idle and drives signal when the host is transmitting, and receives incoming si
- Ethernet adaptor plugs into host and implements protocol



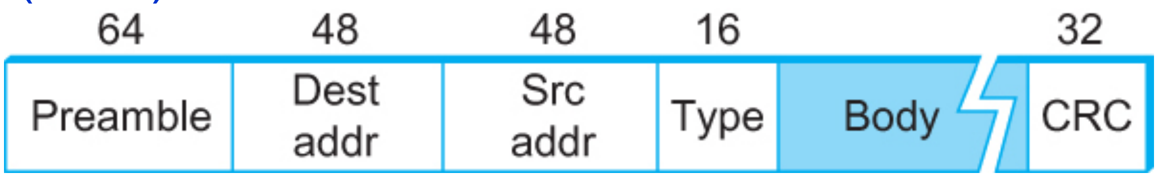
Ethernet is a "Bus"

- Any signal placed on the Ethernet by a host is broadcast over the entire network
  - Signal is propagated in both directions.
  - Repeaters forward the signal on all outgoing segments.
  - Terminators attached to the end of each segment absorb the signal.
- Ethernet hubs rebroadcast onto all attached links so still one broadcast network



Access Protocol for Ethernet -- Framing

- Ethernet's Media Access Control (MAC).
  - implemented in Hardware on the network adaptor.
- Frame format
  - Preamble (64bit): allows the receiver to synchronize with the signal (sequence of alternating 0s and 1s).
  - Host and Destination Address (48bit each).
  - Packet type (16bit): acts as demux key to identify the higher level protocol.
  - Data (up to 1500 bytes)
    - Minimally a frame must contain at least 46 bytes of data.
    - Frame must be long enough to detect collision.
  - CRC (32bit)



Ethernet Addresses

- Each host on an Ethernet (in fact, every Ethernet host in the world) has a unique Ethernet Address.
- The address belongs to the adaptor, not the host.
  - It is usually burnt into ROM.
- Ethernet addresses printed in a human readable format
  - sequence of six numbers separated by colons.
  - Each number corresponds to 1 byte of the 6 byte address and is given by a pair of hexadecimal digits, one for each of the 4-bit nibbles in the byte; Leading 0s are dropped.
  - 8:0:2b:e4:b1:2 is
    00001000 00000000 00101011 11100100 10110001 00000010
- Each manufacturer of devices is allocated a different prefix prepended to address on every adaptor they build (AMD assigned 24bit prefix 8:0:20)

Ethernet Receiver

- Each frame transmitted on an Ethernet is **received** by every adaptor connected to that Ethernet.
- **Each adaptor recognizes those frames addressed to its address and passes only those frames on to the host.**
- In addition, to *unicast* address, an Ethernet address consisting of all 1s is treated as a *broadcast* address.
  - All adaptors pass frames addressed to the *broadcast* address up to the host.
- Similarly, an address that has the first bit set to 1 but is not the *broadcast* address is called a *multicast* address.
  - A given host can program its adaptor to accept some set of *multicast* addresses.

Ethernet Transmitter – Listen before Talk

- When the adaptor has a frame to send and the line is idle, it transmits the frame immediately.
  - **The upper bound of 1500 bytes in the message means that the adaptor can occupy the line for a fixed length of time**.
- When the adaptor has a frame to send and the line is busy, it waits for the line to go idle and then transmits immediately. ("Carrier Sense"=Listen before talk)
- The Ethernet is said to be 1-persistent protocol because an adaptor with a frame to send transmits with probability 1 whenever a busy line goes idle.

Ethernet Transmitter – Collision Detection

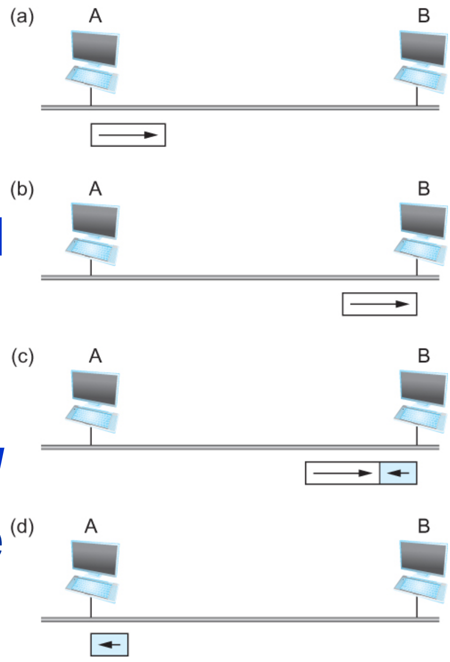
- Since there is no centralized control it is possible for two (or more) adaptors to begin transmitting at the same time,
  - Either because both found the line to be idle,
  - Or, both had been waiting for a busy line to become idle.
- The two (or more) frames are said to be *collide* on the network.
  - Each sender is able to determine that a collision is in progress (signal is abnormal) by “Listening while transmitting".
- At the moment an adaptor detects that its frame is colliding with another, it first makes sure to transmit a 32-bit jamming sequence and then stops transmission.
  - Transmitter minimally sends 96 bits in the case of collision with nearby host: 64-bit preamble + 32-bit jamming sequence

Collision Detection not instantaneous

- The worst case scenario happens when the two hosts are at opposite ends of the Ethernet.
- To know for sure that the frame its just sent did not collide with another frame, the transmitter may need to send as many as 512 bits.
  - Every Ethernet frame must be at least 512 bits (64 bytes) long.
    - 14 bytes of header + 46 bytes of data + 4 bytes of CRC
  - The farther apart two nodes are, the longer it takes for a frame sent by one to reach the other, and the network is vulnerable to collision during this time

Collision detection delay

- A begins frame at time *t; d* is one link latency
- The first bit of A's frame arrives at B at time *t* + *d*
- Suppose an instant before host A's frame arrives, host B begins to transmit its own frame
  - B's frame will immediately collide with A's frame and this collision will be detected by host B
  - Host B will send the 32-bit jamming sequence
  - Host A will not know that the collision occurred until B's frame reaches it, which could happen at *t* + 2 * *d*
- Host A must continue to transmit until this time in order to detect the collision (listen while talking)
  - Host A must transmit for 2 * *d* to be sure that it detects all collisions (d= 51.2 μs if 2500 m→transmit 512 bits on 10 Mbps Ethernet)



Ethernet Transmitter Algorithm

- Consider that a maximally configured Ethernet is 2500 m long, and there may be up to four repeaters between any two hosts, the round trip delay has been determined to be 51.2 μs
  - Which on 10 Mbps Ethernet corresponds to 512 bits
- The other way to look at this situation,
  - We need to limit the Ethernet's maximum latency to a fairly small value (51.2 μs) for the access algorithm to work efficiently
    - Hence the maximum length for the Ethernet is on the order of 2500 m.

Exponential backoff

- What if the network is very busy – lots of host want to transmit a lot – Collision rate goes up
- Once an adaptor has detected a collision, and stopped its transmission, it waits a certain amount of time and tries again.
- **Each time the adaptor tries to transmit but fails, it doubles the amount of time it waits before trying again.**
- This strategy of doubling the delay interval between each retransmission attempt is known as *Exponential Backoff*.

Exponential backoff w/ randomization

- The adaptor first delays either 0 or 51.2 μs, selected at random.
- If this effort fails, it then waits 0, 51.2, 102.4, 153.6 μs (selected randomly) before trying again;
  - This is k * 51.2 for k = 0, 1, 2, 3
- After the third collision, it waits k * 51.2 for $k = 0 \ldots 2^3 - 1$ (again selected at random).
- In general, the algorithm randomly selects a k between 0 and $2^n - 1$ and waits for k * 51.2 μs, where n is the number of collisions experienced so far.

Experience with Ethernet

- Ethernets work best under lightly loaded conditions.
  - Under heavy loads, too much of the network's capacity is wasted by collisions.
- Most Ethernets are used in a conservative way.
  - Have fewer than 200 hosts connected to them which is far fewer than the maximum of 1024.
- Most Ethernets are far shorter than 2500m with a round-trip delay of closer to 5 μs than 51.2 μs.
- Ethernets are easy to administer and maintain.
  - There are no switches that can fail and no routing and configuration tables that have to be kept up-to-date.
  - It is easy to add a new host to the network.
  - It is inexpensive.
    - Cable is cheap, and only other cost is the network adaptor on each host.

Wireless Links

- Wireless links transmit electromagnetic signals
  - Radio, microwave, infrared
- Wireless links all share the same “wire” (so to speak)
  - The challenge is to share it efficiently without unduly interfering with each other
  - Most of this sharing is accomplished by dividing the “wire” along the dimensions of frequency and space
- Control Range through transmit signal power
- Control Frequency through frequency hopping, spread spectrum…

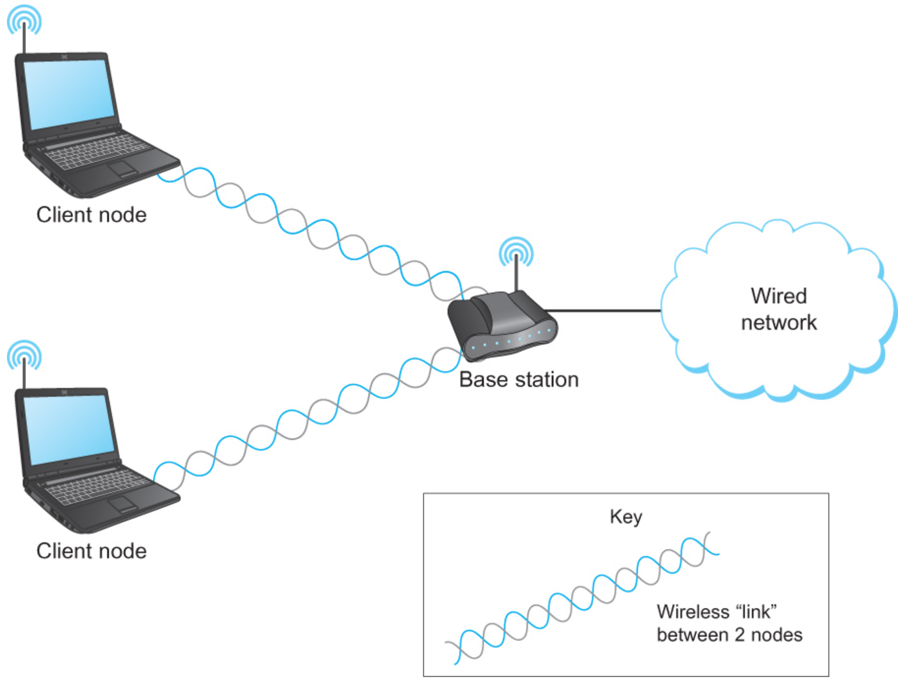
Wireless across various scales

- Wireless technologies differ in a variety of dimensions
  - How much bandwidth they provide
  - How far apart the communication nodes can be
- Four prominent wireless technologies
  - Bluetooth; Wi-Fi (more formally known as 802.11); WiMAX (802.16); 3G-4G cellular wireless

	Bluetooth (802.15.1)	Wi-Fi (802.11)	3G Cellular
Typical link length	10 m	100 m	Tens of kilometers
Typical data rate	2 Mbps (shared)	54 Mbps (shared)	Hundreds of kbps (per connection)
Typical use	Link a peripheral to a computer	Link a computer to a wired base	Link a mobile phone to a wired tower
Wired technology analogy	USB	Ethernet	DSL

Most wireless networks use asymmetry

- Mostly widely used wireless links today are usually asymmetric-- Two end-points are different kinds of nodes
  - One end-point usually has no mobility, but has wired connection to the Internet (known as base station or Access Point (AP))
  - The node at the other end of the link is often mobile



IEEE 802.11

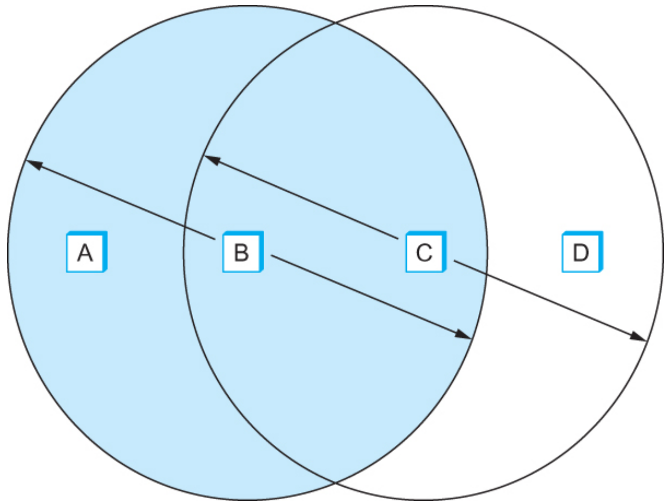
- Also known as Wi-Fi
- Like its Ethernet and token ring siblings, 802.11 is designed for use in a limited geographical area (homes, office buildings, campuses)
  - Primary challenge is to mediate access to a shared communication medium – in this case, signals propagating through space
- 802.11 supports additional features
  - power management and
  - security mechanisms
- Many variants over time, many details, will focus on shared concepts

IEEE 802.11

- Original 802.11 standard defined two radio-based physical layer standard
  - One using the frequency hopping
    - Over 79 1-MHz-wide frequency bandwidths
  - Second using direct sequence
    - Using 11-bit chipping sequence
  - Both standards run in the 2.4-GHz and provide up to 2 Mbps
- Then physical layer standard 802.11b was added
  - Using a variant of direct sequence 802.11b provides up to 11 Mbps
  - Uses license-exempt 2.4-GHz band
- Then came 802.11a which delivers up to 54 Mbps using OFDM
  - 802.11a runs on license-exempt 5-GHz band
- Most recent standard is 802.11g which is backward compatible with 802.11b
  - Uses 2.4 GHz band, OFDM and delivers up to 54 Mbps

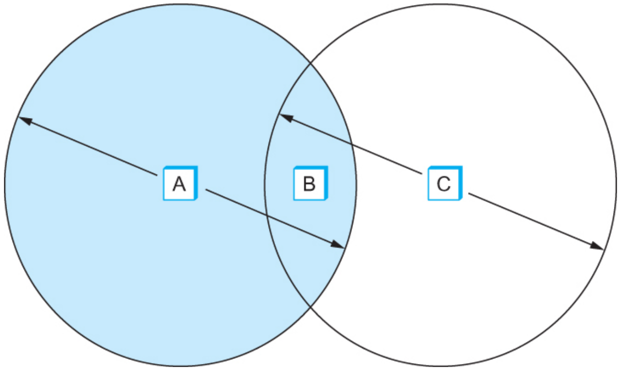
Wireless channels more challenging

- Example: each of four nodes able to send and receive signals with select other nodes – not all
  - B can exchange frames with A and C, but it cannot reach D
  - C can reach B and D but not A
- If they were all connected to an Ethernet they would all reach each other



Collision management challenge -- Hidden nodes

- Suppose both A and C want to communicate with B and so they each send it a frame.
  - A and C are unaware of each other since their signals do not carry that far
  - These two frames collide with each other at B
    - But unlike an Ethernet, neither A nor C is aware of this collision
  - A and C are said to *hidden nodes* with respect to each other



The “Hidden Node” Problem. Although A and C are hidden from each other, their signals can collide at B. (B’s reach is not shown.)

IEEE 802.11 – Collision Avoidance

- 802.11 addresses these two problems with an algorithm called Multiple Access with Collision Avoidance (MACA).
- Key Idea – "less random access, more coordination"
  - Sender and receiver exchange control frames with each other before the sender actually transmits any data.
  - This exchange informs all nearby nodes that a transmission is about to begin
  - Sender transmits a *Request to Send* (RTS) frame to the receiver.
    - The RTS frame includes a field that indicates how long the sender wants to hold the medium
      - Length of the data frame to be transmitted
  - Receiver replies with a *Clear to Send* (CTS) frame
    - This frame echoes this length field back to the sender

IEEE 802.11 – Collision Avoidance

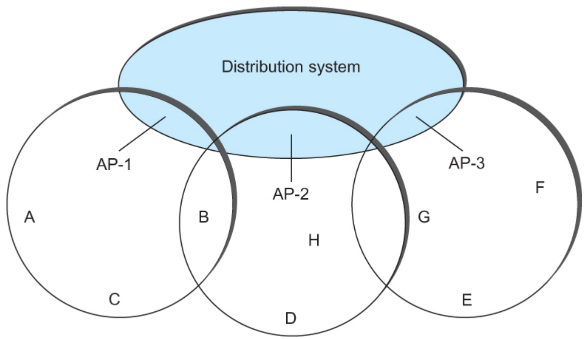
- Any node that sees the CTS frame knows that
  - it is close to the receiver, therefore
  - cannot transmit for the period of time it takes to send a frame of the specified length
- Any node that sees the RTS frame but not the CTS frame
  - is not close enough to the receiver to interfere with it, and
  - so is free to transmit

IEEE 802.11 – Collision Avoidance

- Using ACK in MACA
  - Proposed in MACAW: MACA for Wireless LANs
- Receiver sends an ACK to the sender after successfully receiving a frame
- All nodes must wait for this ACK before trying to transmit
- If two or more nodes detect an idle link and try to transmit an RTS frame at the same time
  - Their RTS frame will collide with each other
- 802.11 does not support collision detection
  - So the senders realize the collision has happened when they do not receive the CTS frame after a period of time
  - In this case, they each wait a random amount of time before trying again.
  - The amount of time a given node delays is defined by the same *exponential backoff* algorithm used on the Ethernet.

IEEE 802.11 – Access Point Distribution System

- Following figure illustrates a distribution system that connects three access points, each of which services the nodes in the same region
- Each of these regions is analogous to a cell in a cellular phone system with the APs playing the same role as a base station
  - Each nodes associates itself with one access point
  - For node A to communicate with node E, A first sends a frame to its AP-1 which forwards the frame across the distribution system to AP-3, which finally transmits the frame to E



Frame Format for Distribution across APs

- Source and Destinations addresses: each 48 bits
- Data: up to 2312 bytes; CRC: 32 bit
- Control field: 16 bits
  6 bit **Type** field: indicates whether the frame is an RTS or CTS frame or being used by the scanning algorithm; A pair of 1 bit fields : called **ToDS** and **FromDS**
- Simplest case: Both DS bits are 0 -- one node sends directly to another, Addr1 identifies the target node, and Addr2 identifies the source node
- Complex case: Both DS bits are 1 -- message went from wireless node onto distribution system, and from distribution system to another wireless node
  - Addr1 identifies the ultimate destination,
  - Addr2 identifies the immediate sender (the one that forwarded the frame from the distribution system to the ultimate destination)
  - Addr3 identifies the intermediate destination (the one that accepted the frame from a wireless node and forwarded across the distribution system)
  - Addr4 identifies the original source
  - Addr1: E, Addr2: AP-3, Addr3: AP-1, Addr4: A

16	16	48	48	48	16	48	0–18,496	32
Control	Duration	Addr1	Addr2	Addr3	SeqCtrl	Addr4	Payload	CRC

IEEE 802.11 – AP association

- How do the nodes select their access points
- How does it work when nodes move from one cell to another

- The technique for selecting an AP is called *scanning*
  - The node sends a *Probe* frame
  - All APs within reach reply with a *Probe Response* frame
  - The node selects one of the access points and sends that AP an *Association Request* frame
  - The AP replies with an *Association Response* frame

- A node engages this protocol whenever
  - it joins the network, as well as
  - when it becomes unhappy with its current AP
    - This might happen, for example, because the signal from its current AP has weakened due to the node moving away from it
    - Whenever a node acquires a new AP, the new AP notifies the old AP of the change via the distribution system

Bluetooth

- Used for very short range communication between mobile phones, PDAs, notebook computers and other personal or peripheral devices
- Operates in the license-exempt band at 2.45 GHz
- Has a range of only 10 m
- Communication devices typically belong to one individual or group
  - Sometimes categorized as Personal Area Network (PAN)
- Version 2.0 provides speeds up to 2.1 Mbps
- Power consumption is low

ZigBee

- ZigBee competes with Bluetooth
- Devised by the ZigBee alliance and standardized as IEEE 802.15.4
- It is designed for situations where the bandwidth requirements are low and power consumption must be very low to give very long battery life
- It is also intended to be simpler and cheaper than Bluetooth, making it financially feasible to incorporate in cheaper devices such as a wall switch that wirelessly communicates with a ceiling-mounted fan