CS 5450

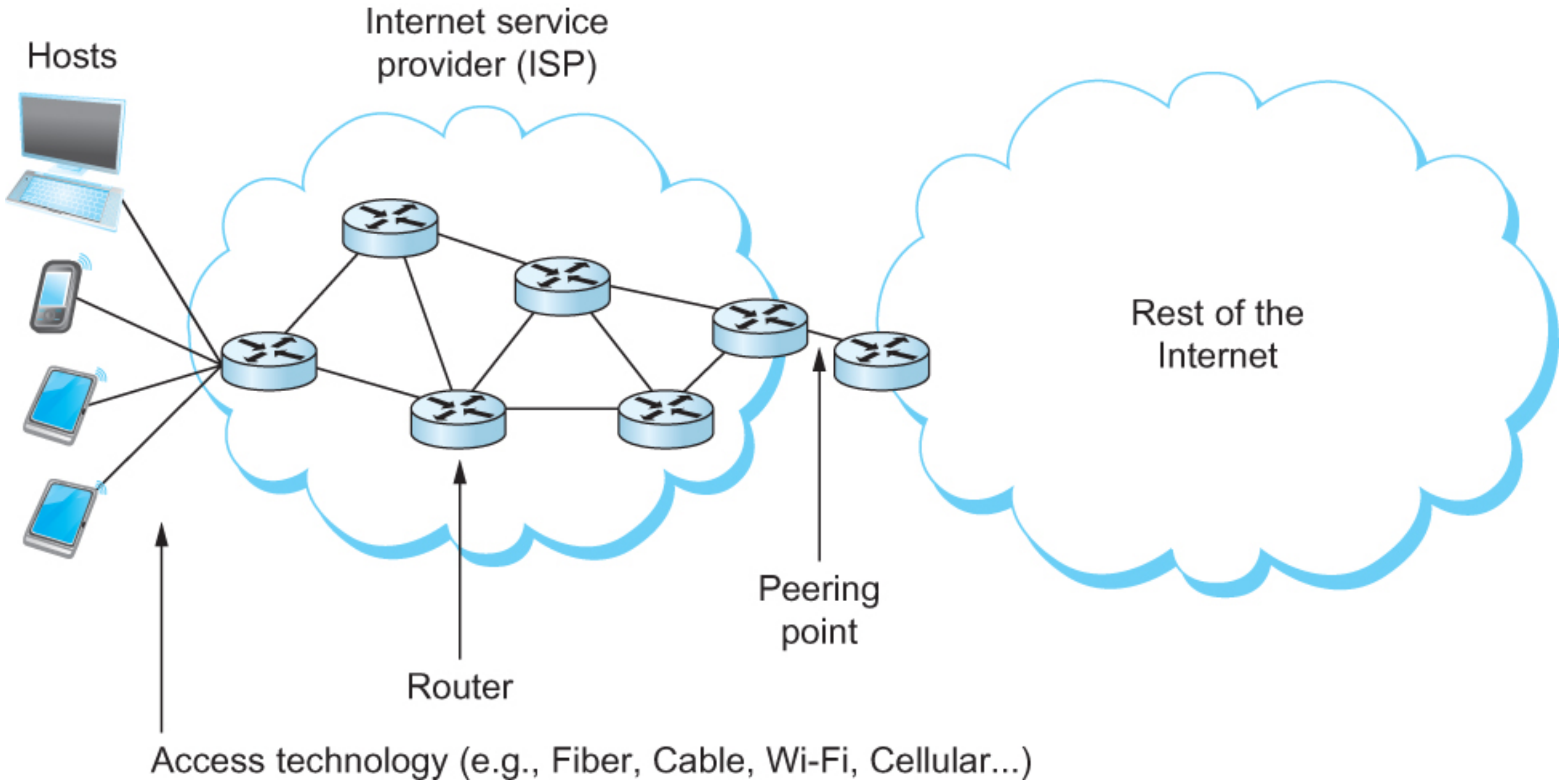
# Framing
# Reliable Transmission
# Ethernet and Wi-Fi

Vitaly Shmatikov

# Framing, Reliability, Sharing

- Delineate a 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
- Making the links reliable in spite of transmission problems
- Media access control problem
  - Collision detection vs collision avoidance

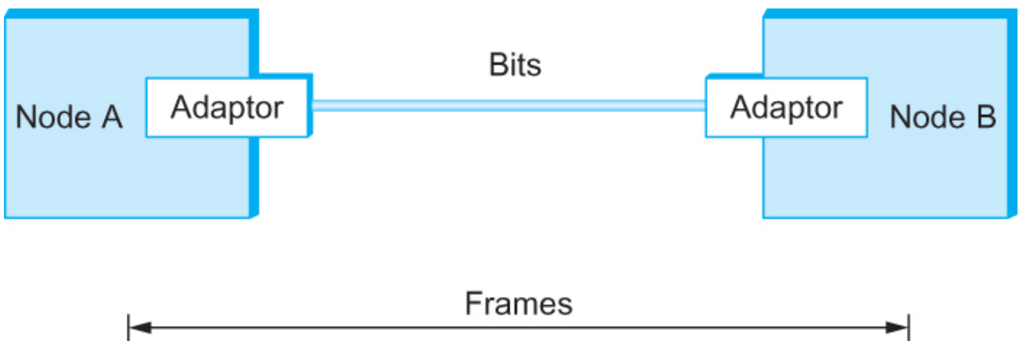
# Context



Hosts
Internet service
provider (ISP)
Rest of the
Internet
Peering
point
Router
Access technology (e.g., Fiber, Cable, Wi-Fi, Cellular...)

# Framing

- In packet-switched networks, blocks of data (called frames at this level), not bit streams, are exchanged between nodes
- Network adaptor 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

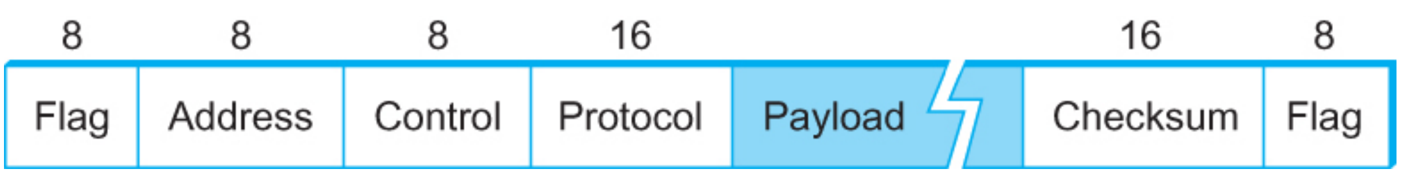
# Byte-Oriented Framing: BISYNC

- 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

# Byte-Oriented Framing: PPP

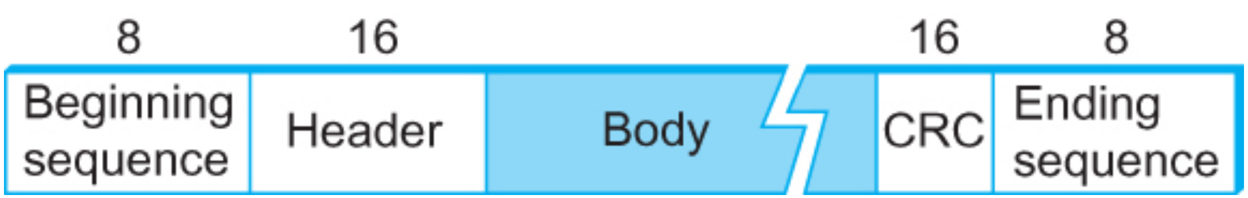
- PPP commonly run over Internet links uses sentinel approach
  - Special start of text character denoted as Flag
    - 0 1 1 1 1 1 1 0
  - Address, control : default numbers
  - Protocol for demux : IP / IPX
  - Payload : negotiated (1500 bytes)
  - Checksum : for error detection




PPP Frame Format

# Bit-Oriented Framing: HDLC

- 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
  - Reasons: electrical interference, thermal noises, …
- Common techniques for detecting transmission errors
  - CRC (Cyclic Redundancy Check)
    - Used in HDLC, DDCMP, CSMA/CD, Token Ring
  - Checksum
    - Used in IP

# Basic Idea of Error Detection

- Add redundant information to a frame that can be used to determine if errors have been introduced
- Extreme case: transmit two complete copies
  - Identical → No error
  - Differ → Error
  - Poor scheme ???
    - n bit message, n bit redundant information
    - Error can go undetected
- Strong error detection techniques
  - n-bit message, k redundant bits, k << n
  - In Ethernet, a frame carrying up to 12,000 bits of data requires only 32-bit CRC

# Redundant Bits for Error Detection

- Add no new information to the message
- Derived from the original message using some algorithm
- Both the sender and receiver know the algorithm
  - Link layer: CRC
  - IP: weak checksum (sum of 16-bit words)

Sender

| | |
|---|---|
| **m** | **r** |

Receiver

| | |
|---|---|
| **m** | **r** |

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

# If Error Is Detected…

Option 1: using error detection and correction algorithm, 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

Option 2: notify the sender that the message was corrupted, so the sender can send again

- If errors are rare, then the retransmitted message will be error-free

# Reliable Transmission

- A reliable link-level protocol must recover from discarded frames
- A combination of two fundamental mechanisms
  - Acknowledgements
  - Timeouts
- Acks + timeouts for reliable delivery = Automatic Repeat reQuest (ARQ)

# Acknowledgments

- An acknowledgement (ACK) is a small control frame that a protocol sends back to its peer saying that it has received the earlier frame
  - Control frame = 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

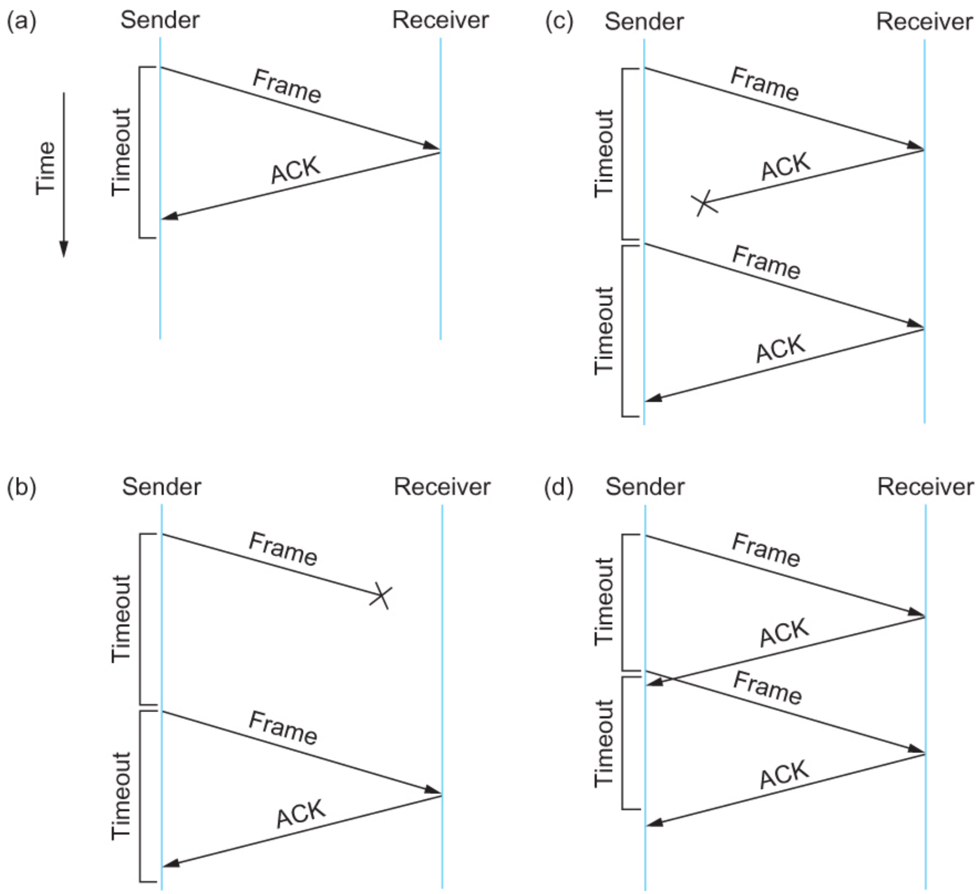
# Timeouts

- 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
  - What's "reasonable"?

# Stop and Wait Protocol

- 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

# Four Possible Scenarios



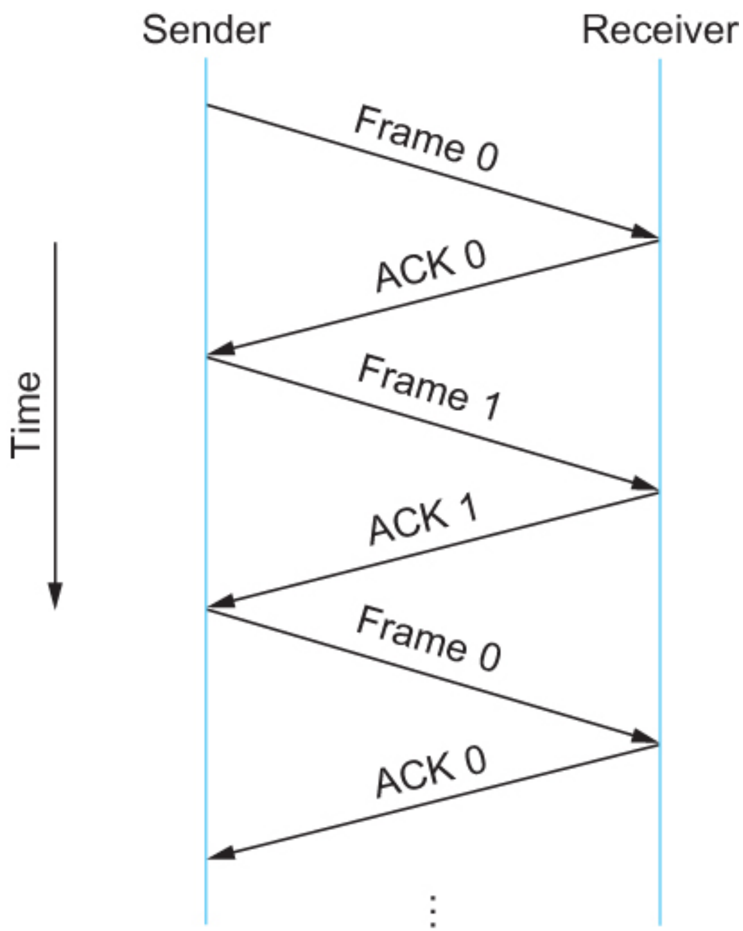

(a) The ACK is received before the timer expires; (b) the original frame is lost; (c) The ACK is lost; (d) the timeout fires too soon

# Duplicate Frames

- 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

# Stop And Wait w/ 1-Bit Seq No

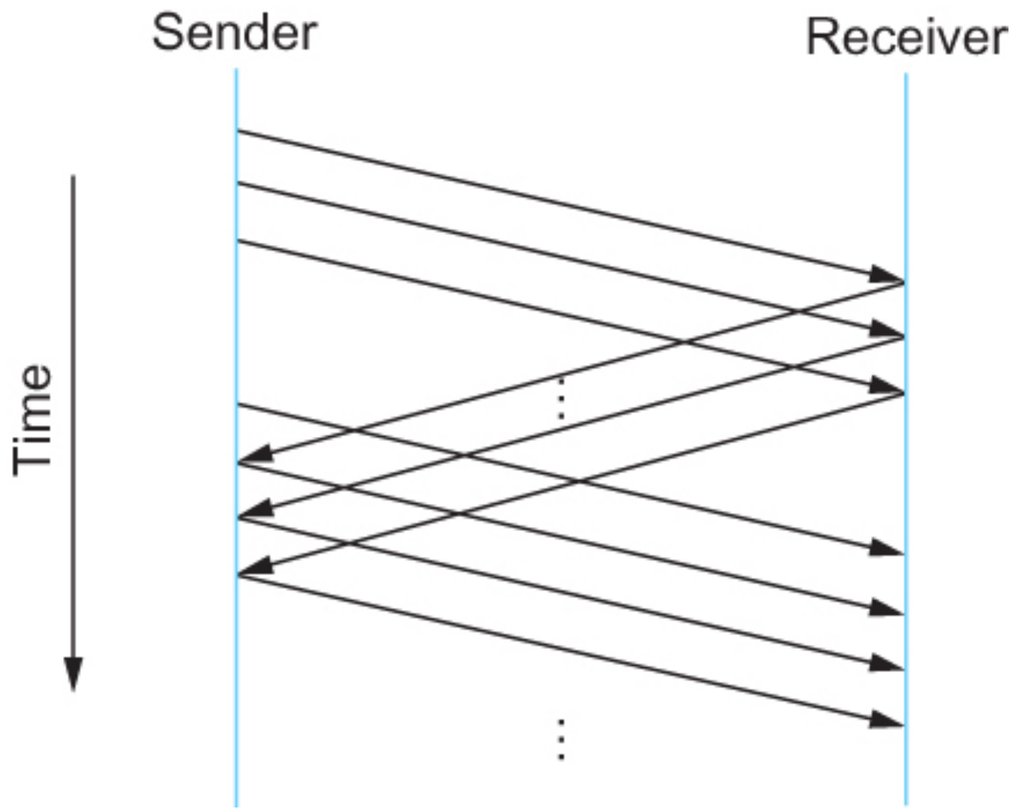


# Efficiency of Stop and Wait

- 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
    - 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

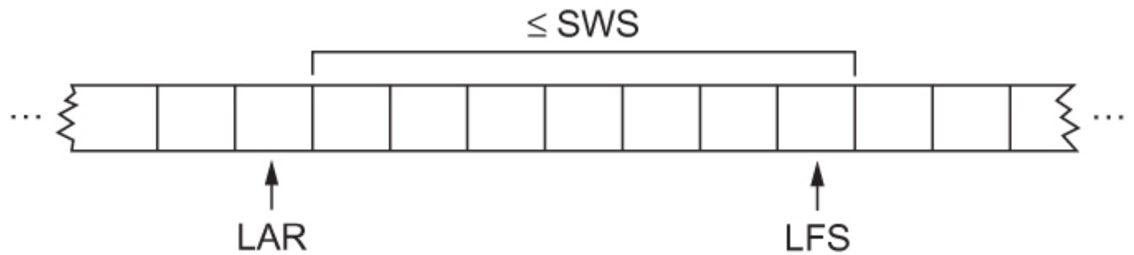
# Sliding Window Protocol




Idea: introduce pipelining

# Sliding Window: Sender Protocol

- Sender assigns a sequence number 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 maintains invariant: LFS – LAR ≤ SWS



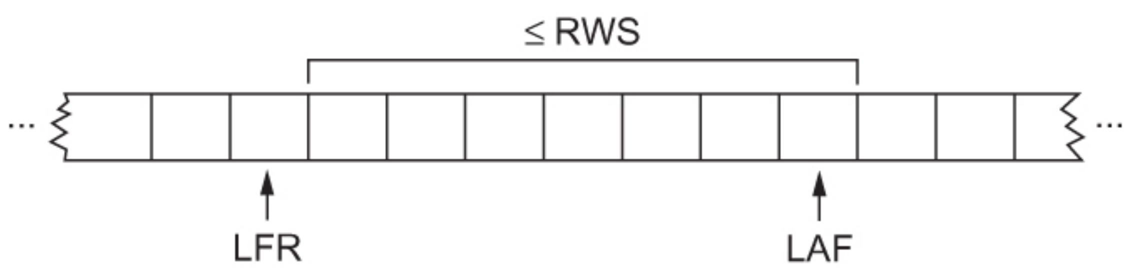
# Sliding Window: Sender Protocol

- When acknowledgement arrives, the sender moves LAR to right, thereby allowing the sender to transmit another frame
- The sender associates a timer with each frame it transmit. If the timer expires before the ACK is received, the sender retransmits the frame.
- The sender has to be willing to buffer up to …

  … SWS frames

# Sliding Window: Receiver 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



# Sliding Window: Receiver 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 frames have been received
    - This acknowledgement is said to be cumulative
  - The receiver then sets LFR = SeqNumToAck and adjusts LAF = LFR + RWS

# Sliding Window Inefficiencies

- When timeout occurs, the amount of data in transit decreases
  - Since the sender is unable to advance its window
- When frame loss occurs, this scheme is no longer keeping the pipe full
  - The longer it takes to notice that a frame loss has occurred, the more severe the problem becomes
  - Will discuss improvements and subtleties when discussing TCP
    - Negative Acknowledgement (NAK)
    - Additional Acknowledgement
    - Selective Acknowledgement

# NAKs, Duplicate ACKs, Selective ACKs

- Negative Acknowledgement (NAK)
  - Receiver sends NAK for frame 6 when frame 7 arrives
    - Unnecessary since sender's timeout mechanism will 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 numbers must be larger than the number of outstanding frames allowed
  - Stop and Wait: 1 outstanding frame
    - 2 distinct sequence numbers (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

# Link Layer

- Ethernet
- Wi-Fi

# Ethernet

- Most successful local area networking technology of last ~~20~~ ~~30~~ 40 years
- Developed in the mid-1970s by researchers at the Xerox Palo Alto Research Center (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

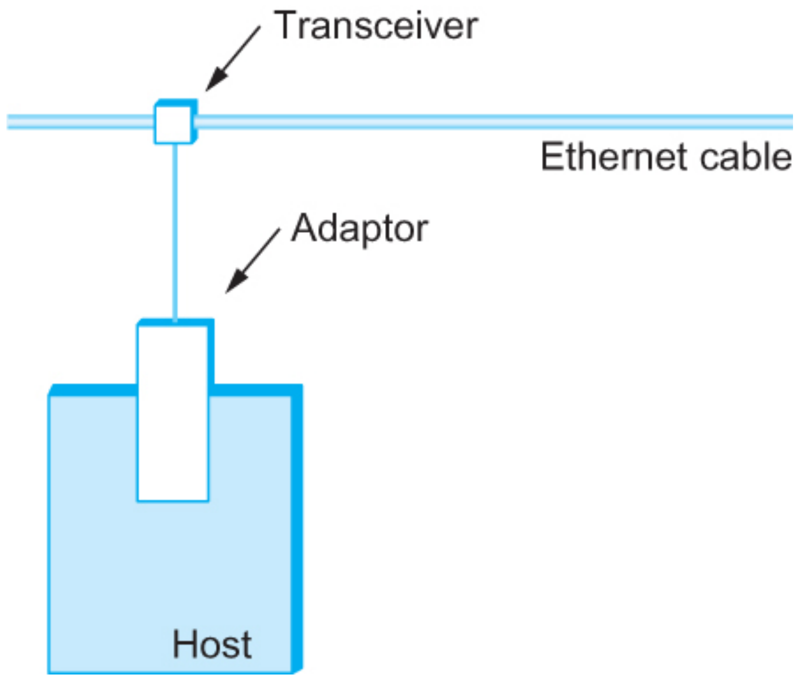
# Ethernet

- Uses ALOHA (packet radio network) as the root protocol
  - Developed at the University of Hawaii to support communication across the Hawaiian Islands
  - 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 has been extended to include a 100-Mbps version called Fast Ethernet and a 1000-Mbps version called Gigabit Ethernet

# Ethernet

- An Ethernet segment is implemented on a coaxial cable of up to 500 m.
  - Similar to the type used for cable TV except that it typically has an impedance of 50 ohms instead of cable TV's 75 ohms.
- 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
- The transceiver also receives incoming signal
- The transceiver is connected to an Ethernet adaptor which is plugged into the host
- The protocol is implemented on the adaptor

# Ethernet Transceiver and Adaptor

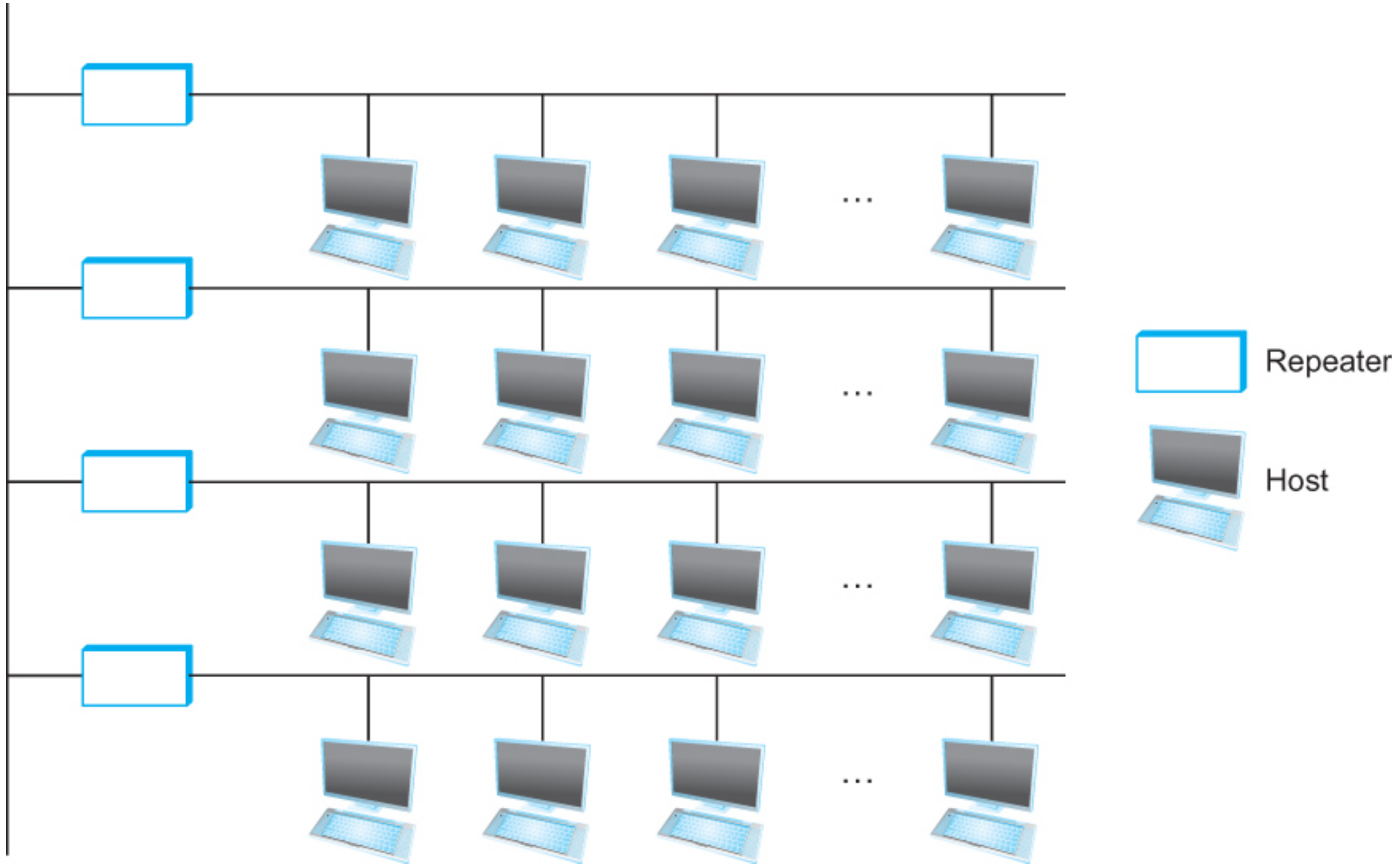



Ethernet transceiver and adaptor

# Ethernet Repeater

- Multiple Ethernet segments can be joined together by repeaters
- A repeater is a device that forwards digital signals.
- No more than four repeaters may be positioned between any pair of hosts
  - An Ethernet has a total reach of only 2500 m

# Ethernet




Ethernet repeater

# Ethernet Broadcast

- 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 uses Manchester encoding scheme

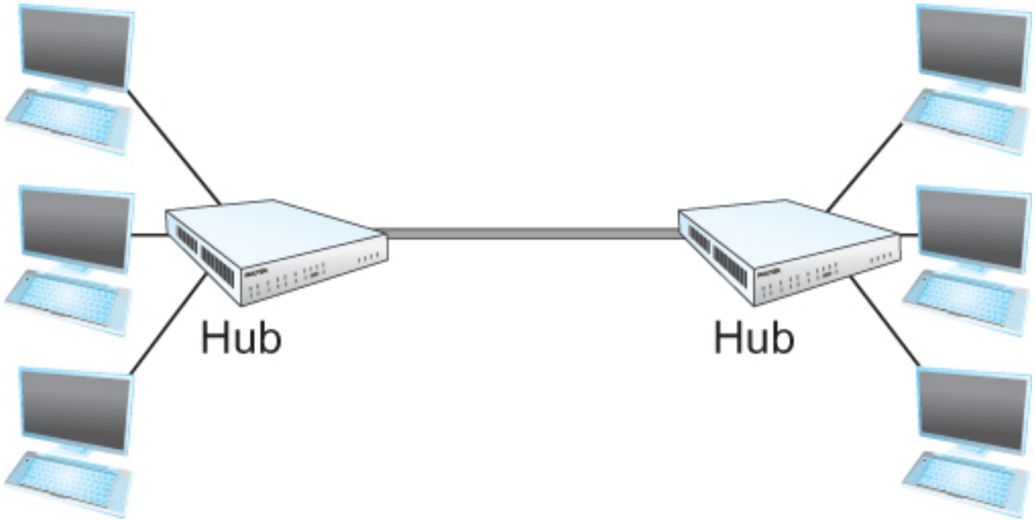
# Newer Ethernet Technologies

- Instead of using coax cable, an Ethernet can be constructed from a thinner cable known as 10Base2 (the original was 10Base5)
  - 10 means the network operates at 10 Mbps
  - Base means the cable is used in a baseband system
  - 2 means that a given segment can be no longer than 200 m

# Newer Ethernet Technologies

- Another cable technology is 10BaseT
  - T stands for twisted pair
  - Limited to 100 m in length
- With 10BaseT, the common configuration is to have several point to point segments coming out of a multiway repeater, called Hub

# Ethernet Hub

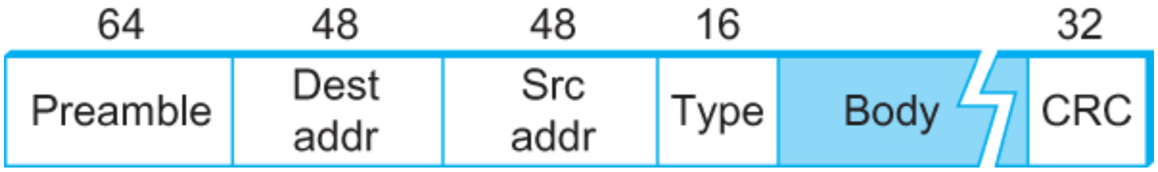



Ethernet Hub

# Ethernet's Media Access Control

- 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 Frame




Ethernet Frame Format

# Ethernet Address

- 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
  - Usually burnt into ROM
- Ethernet addresses are typically printed in a human readable format: 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.
  - For example, 8:0:2b:e4:b1:2 is

    00001000 00000000 00101011 11100100 10110001 00000010

# Ethernet Address Allocation

- To ensure that every adaptor gets a unique address, each manufacturer of Ethernet devices is allocated a different prefix that must be prepended to the address on every adaptor they build
  - For example, AMD has been assigned the 24-bit prefix 8:0:20

# Unicast, Broadcast, Multicast

- 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 Addresses

To summarize, an Ethernet adaptor receives all frames and accepts:

- Frames addressed to its own address
- Frames addressed to the broadcast address
- Frames addressed to a multicast addressed if it has been instructed

# Ethernet Transmitter Algorithm

- 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
- 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.

# Collisions

- 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.
- When this happens, the two (or more) frames are said to collide on the network.

# Collision Detection

- Since Ethernet supports collision detection, each sender is able to determine that a collision is in progress
- 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
  - Thus, a transmitter will minimally send 96 bits in the case of collision: 64-bit preamble + 32-bit jamming sequence

# Collision Detection

- One way that an adaptor will send only 96 bit (called a runt frame) is if the two hosts are close to each other
- Had they been farther apart, they would have had to transmit longer, and thus send more bits, before detecting the collision

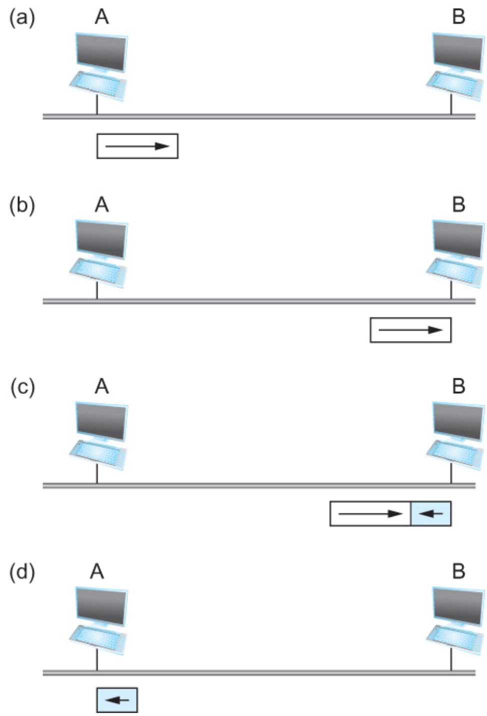
# Ethernet Worst-Case Scenario

- 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

# Ethernet Transmitter Algorithm

- A begins transmitting a frame at time *t*
- *d* denotes the 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 will happen at $t + 2 * d$
- Host A must continue to transmit until this time in order to detect the collision
  - Host A must transmit for $2 * d$ to be sure that it detects all possible collisions

# Ethernet Worst-Case Scenario




(a) A sends a frame at time t

(b) A's frame arrives at B at time t + d

(c) B begins transmitting at time t + d and collides with A's frame

(d) B's runt (32-bit) frame arrives at A at time t + 2d

# Why 512 Bits?

- 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

- Another interpretation: need to limit the Ethernet's maximum latency to a fairly small value (51.2 μs) for the access algorithm to work
  - Hence the maximum length for the Ethernet is around 2500 m

# Exponential Backoff

- 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

- 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

# Ethernet in Practice

- 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 vs. maximum of 1024
- Most Ethernets are far shorter than 2500m with a round-trip delay of closer to 5 $\mu$s than 51.2 $\mu$s
- 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
- Exclusive use of a particular frequency in a particular geographic area may be allocated to an individual entity such as a corporation

# Frequency Allocation

- Frequency allocations are determined by government agencies such as FCC (Federal Communications Commission) in USA
- Specific bands (frequency) ranges are allocated to certain uses.
  - Some bands are reserved for government use
  - Other bands are reserved for uses such as AM radio, FM radio, televisions, satellite communications, and cell phones
  - Specific frequencies within these bands are then allocated to individual organizations for use within certain geographical areas
  - Finally, there are several frequency bands set aside for “license exempt” usage
    - Bands in which a license is not needed

# Wireless Technologies

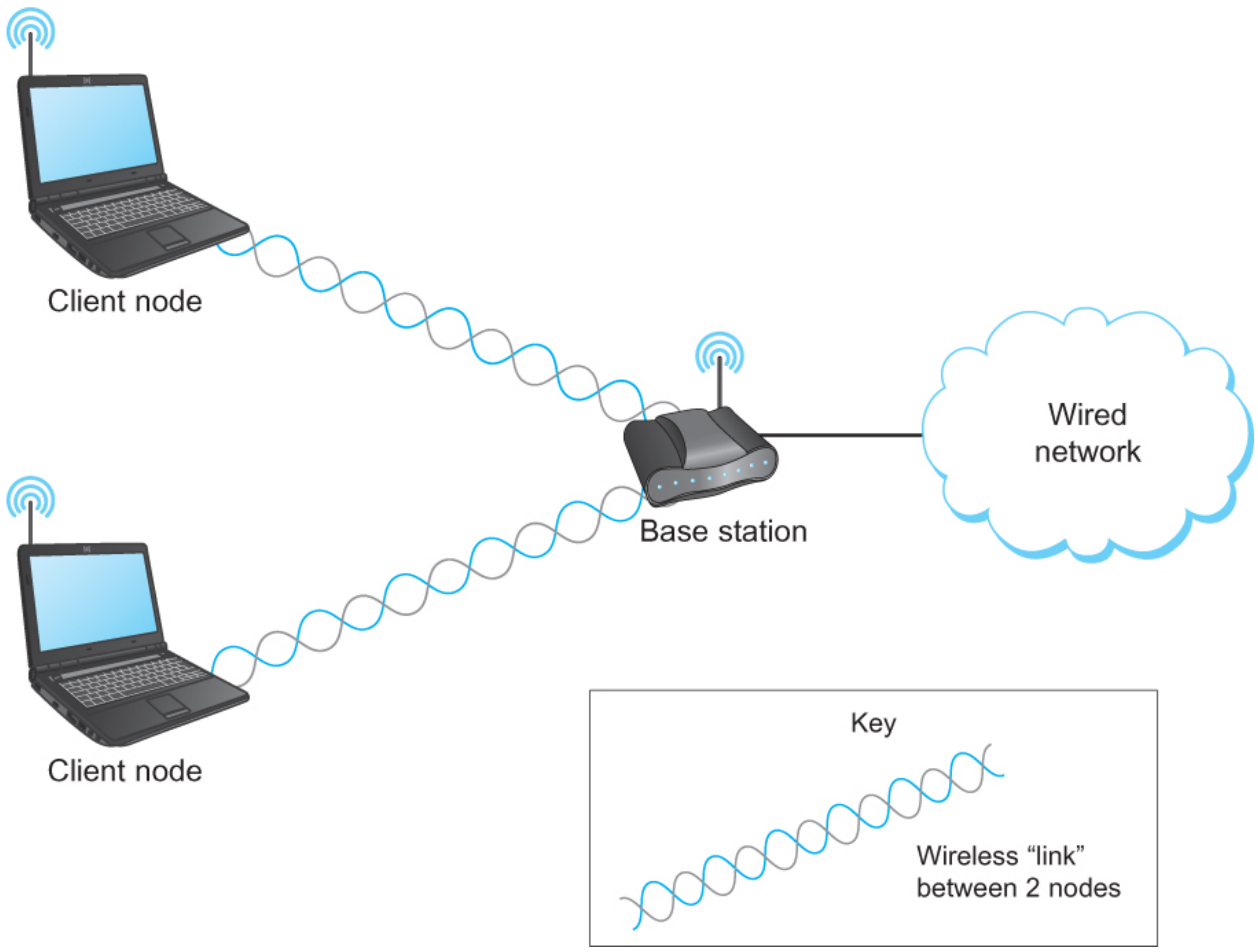
| | Bluetooth (802.15.1) | Wi-Fi (802.11) | 3G Cellular |
|---|---|---|---|
| Typical link length | 10 m | 100 m | Tens of kilometers |
| Typical data rate | 2 Mbps (shared) | 54 Mbps (shared) | Hundreds of kbps (per connection) |
| Typical use | Link a peripheral to a computer | Link a computer to a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

Overview of leading wireless technologies

# Asymmetry

- Mostly widely used wireless links today are usually asymmetric
- The 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)
  - The node at the other end of the link is often mobile

# Asymmetry

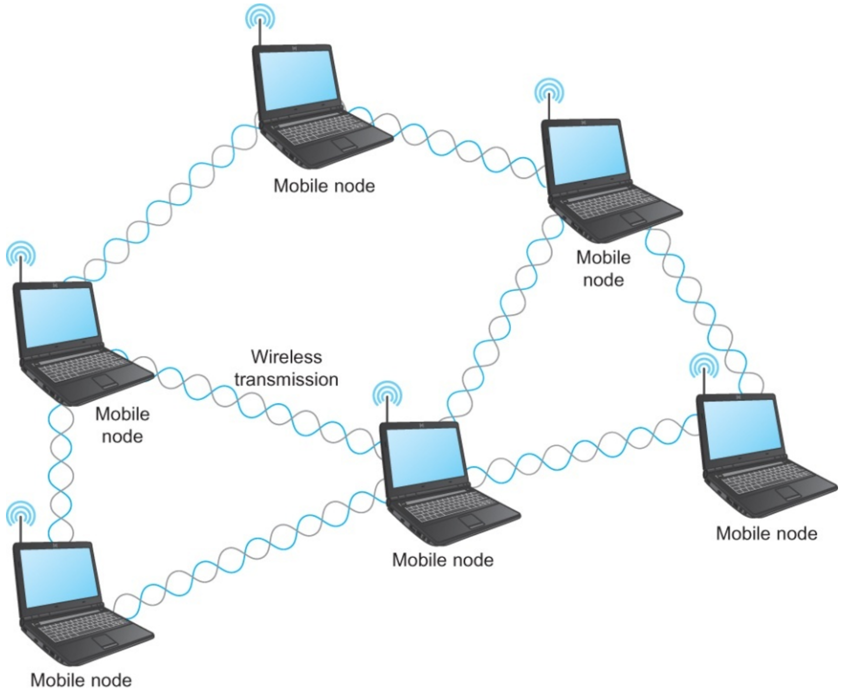

Client node
Client node
Base station
Wired
network
Key
Wireless "link"
between 2 nodes

# Wireless Links

- Wireless communication supports point-to-multipoint communication
- Communication between non-base (client) nodes is routed via the base station
- Three levels of mobility for clients
  - No mobility: the receiver must be in a fix location to receive a directional transmission from the base station (initial version of WiMAX)
  - Mobility is within the range of a base (Bluetooth)
  - Mobility between bases (Cell phones and Wi-Fi)

# Mesh / Ad-hoc Network

- Nodes are peers
- Messages may be forwarded via a chain of peer nodes

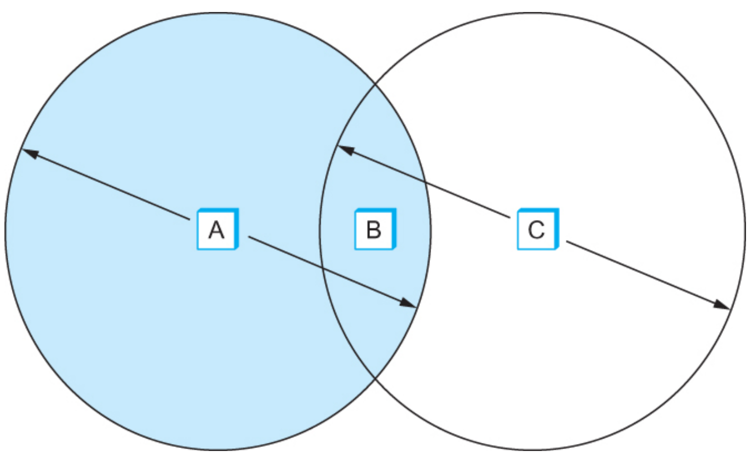


# IEEE 802.11, aka Wi-Fi

- 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 also supports power management and security

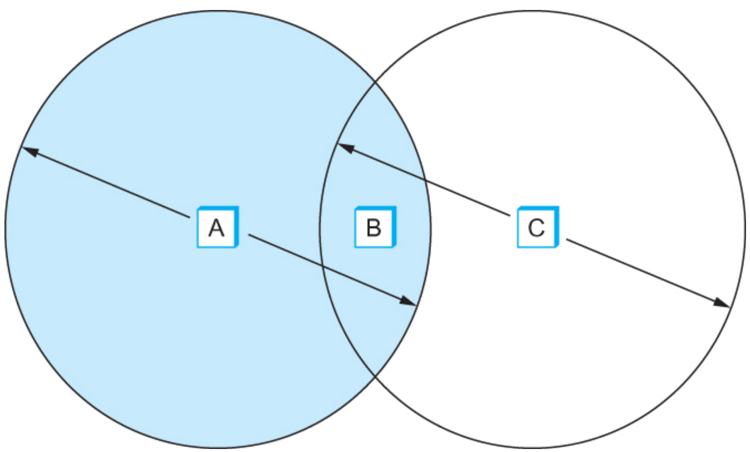
# Issues with Collision Avoidance

- Each of four nodes is able to send and receive signals that reach just the nodes to its immediate left and right
  - For example, B can exchange frames with A and C, but it cannot reach D
  - C can reach B and D but not A



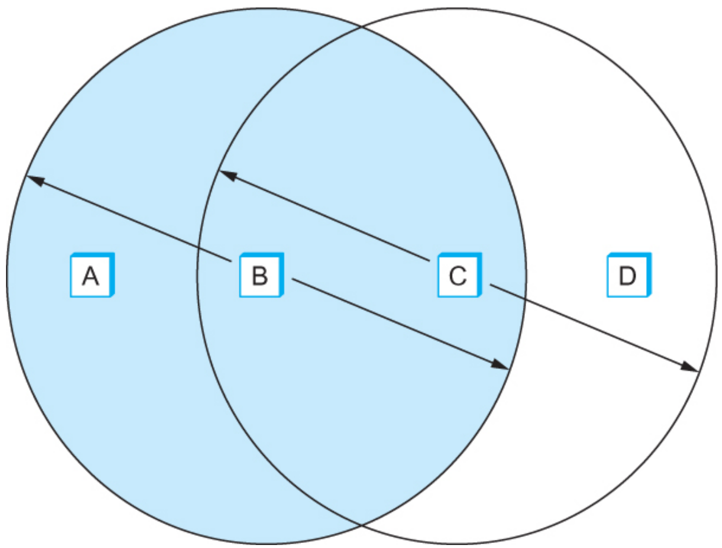
# Hidden Nodes

- 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
    - Unlike in an Ethernet, neither A nor C is aware of this collision
  - A and C are said to be hidden nodes with respect to each other



# Exposed Nodes

- Exposed node problem
  - Suppose B is sending to A. Node C is aware of this communication because it hears B's transmission.
  - It would be a mistake for C to conclude that it cannot transmit to anyone just because it can hear B's transmission.
  - Suppose C wants to transmit to node D.
  - This is not a problem since C's transmission to D will not interfere with A's ability to receive from B.



# IEEE 802.11: Collision Avoidance

- Multiple Access with Collision Avoidance (MACA)
- Key idea
  - 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 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