Data Link Layer: Part 2

- Data Link Layer Functions: Recap
- Point-to-Point Data Link Protocols
- Broadcast LAN and Media Access Control
  - Taxonomy of MAC Protocols
  - Static Partitions: TDMA, FDMA, CDMA, etc.
  - (Demand Adaptive) Controlled Access: (master-slave based) polling (e.g., Bluetooth/802.15); token-passing (e.g., Token Bus/802.4, Token Ring/802.5, FDDI); …
  - Random Access: e.g., Aloha and slotted Aloha; CSMA and CSMA/CD (Ethernet/802.3); CSMA/CA (WiFi/802.11); …
- Ethernet and Its Evolution
- Token Ring; DOCSIS
- Ethernet vs. Token Ring: “battle of technology”
Data Link Layer: Basic Functions

Recap

Some terminology:

- hosts and routers are **nodes** (bridges and switches too)
- communication channels that connect adjacent nodes along communication path are **links**
  - wired links
  - wireless links
  - **LANs** (local area networks)
- layer 2 PDU ("packet") referred to as **frame**, which encapsulates a layer-3 packet, e.g., an IP datagram
What Does Data Link Layer Do?

**Data link layer** has responsibility of transferring frames from one node to adjacent node over a single link

- An IP packet from host A to host B may traverses different links using different data link protocols
  - e.g., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- Each link protocol provides different services
  - e.g., may or may not provide reliable data delivery
- Different link protocols are not inter-operable!
  - IP packets are encapsulated/decapsulated with appropriate data link protocol header over each link
  - IP protocol and IP routers glue the links (“physical networks”) together and provide end-to-end data delivery!
Data Link Layer Functions

• **Framing**
  - sender (transmitter): encapsulate datagram into frame, adding header, trailer, transmit frame
  - receiver: detect beginning of frames, receive frame, decapsulate frame, stripping off header, trailer

• **Link Access (Media Access Control)**
  - determine whether it’s Okay to transmit over the link
    • particularly important when link shared by many nodes
      - also an issue over “half-duplex” point-to-point link (why?)
    • need media access control (MAC)
  - “physical addresses” identify sender/receiver on a link!
    • particularly important when link shared by many nodes, while over point-to-point link, not necessary
    • “physical addresses” often referred to as “MAC” addresses
      - different from IP addresses (which are logical & global)!
Other Data Link Layer Functions

- **Error Detection (commonly implemented)**
  - errors caused by signal attenuation, noise, etc.
  - sender computes “checksum”, attaches to frame
  - receiver detects presence of errors by verifying “checksum”
    - drops corrupted frame, may ask sender for retransmission
  - Commonly used “checksum”: cyclic redundancy code (CRC)

- **Reliable Delivery between adjacent nodes (optional)**
  - using, e.g., go-back-N or selective repeat protocol
    - seldom used on low bit error link (fiber, some twisted pair)
    - wireless links: high error rates
    - Q: why both link-level and end-end reliability?

- **Error Correction (optional)**
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission, using forward error correction (FEC) codes

- **Flow Control (optional)**
  - negotiating transmission rates between two nodes
Where is the Link Layer Implemented?

- in each and every host
- link layer implemented in “adaptor” (aka network interface card NIC) or on a chip
  - Ethernet card, 802.11 card; Ethernet chipset
  - implements link, physical layer
- attaches into host’s system buses
- combination of hardware, software, firmware
Adaptors Communicating

- **sending side:**
  - encapsulates datagram in frame
  - adds error checking bits, rdt, flow control, etc.

- **receiving side**
  - looks for errors, rdt, flow control, etc.
  - extracts datagram, passes to upper layer at receiving side
Point to Point Data Link Control

• one sender, one receiver, one link: easier than broadcast link:
  - no Media Access Control
  - no need for explicit MAC addressing
  - e.g., dialup link, ISDN line

• popular point-to-point DLC protocols:
  - PPP (point-to-point protocol)
  - HDLC: High level data link control
    • data link layer used to be considered “high layer” in protocol stack!
PPP Design Requirements [RFC 1557]

- **packet framing**: encapsulation of network-layer datagram in data link frame
  - carry network layer data of any network layer protocol (not just IP) at same time
  - ability to demultiplex upwards

- **bit transparency**: must carry any bit pattern in the data field

- **error detection** (no correction)

- **connection liveness**: detect, signal link failure to network layer

- **network layer address negotiation**: endpoint can learn/configure each other’s network address
PPP Non-Requirements

- no error correction/recovery
- no flow control
- out of order delivery OK
- no need to support multipoint links (e.g., polling)

Error recovery, flow control, data re-ordering all relegated to higher layers!
PPP Data Frame

- **Flag:** delimiter (framing)
- **Address:** does nothing (only one option)
- **Control:** does nothing; in the future possible multiple control fields
- **Protocol:** upper layer protocol to which frame delivered (e.g., PPP-LCP, IP, IPCP, etc)
PPP Data Frame

- **info**: upper layer data being carried
- **check**: cyclic redundancy check for error detection

```
flag 1 1 1 1 or 2 variable length 2 or 4 1
01111110 11111111 00000011 protocol info check 01111110
```

flag  control  address
Byte Stuffing

• “data transparency” requirement: data field must be allowed to include flag pattern <01111110>
  - Q: is received <01111110> data or flag?

• Sender: adds (“stuffs”) extra < 01111110> byte after each < 01111110> data byte

• Receiver:
  - two 01111110 bytes in a row: discard first byte, continue data reception
  - single 01111110: flag byte
Byte Stuffing

flag byte pattern in data to send

flag byte pattern plus stuffed byte in transmitted data
PPP Link/Network Control Protocols

Before exchanging network-layer data, data link peers must:

- **configure PPP link** (max. frame length, authentication)
- **learn/configure network layer information**
  - for IP: carry IP Control Protocol (IPCP) msgs (protocol field: 8021) to configure/learn IP address
Multiple Access Links: MAC Protocols

two types of “links”:

- **point-to-point**
  - PPP for dial-up access
  - point-to-point link between Ethernet switch, host (PPPoE)

- **broadcast (shared wire or medium)**
  - old-fashioned Ethernet
  - upstream HFC
  - 802.11 wireless LAN

shared wire (e.g., cabled Ethernet)  shared RF (e.g., 802.11 WiFi)  shared RF (satellite)  humans at a cocktail party (shared air, acoustical)
Broadcast LAN: Media Access Control

- Broadcast LAN: single shared broadcast channel
  - two or more simultaneous transmissions by nodes: interference!
    - collision if node receives two or more signals at the same time
  - only one node can send successfully at a time!
- How to share a broadcast channel?
  - Humans use multi-access protocols all the time

Multiple Access Protocol
- distributed algorithm that determines how nodes share channel, i.e., determine when node can transmit
- communication about channel sharing must use channel itself!
- what to look for in multiple access protocols:
  - synchronous or asynchronous
  - information needed about other stations
  - robustness
  - performance: access delay and throughput
MAC Protocols: a Taxonomy

Three broad classes:

- **Channel Partitioning** (static controlled access)
  - divide channel into smaller “pieces” (e.g., time slots → TDMA, frequency→FDMA, code→CDMA)
  - allocate piece to node for exclusive use

- “Demand Adaptive” Controlled Access: e.g., Polling or Taking Turns
  - tightly coordinate shared access to avoid collisions

- **Random Access**
  - channel not divided, allow collisions
  - “recover” from collisions
Taxonomy of MAC Protocols

Multiple Access Protocols

random access

CSMA/CA
(WiFi/802.11)

controlled access

ALOHA

CSMA/CD
(IEEE 802.3)

demand adaptive

static channel

polling

token passing

reservation

allocation

token passing

bus

IEEE 802.4

ring

IEEE 802.5

FDDI

TDM

FDM
Channel Partitioning MAC protocols: TDMA

TDMA: time division multiple access

- access to channel in "rounds"
- each station gets fixed length slot (length = packet transmission time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have packets to send, slots 2,5,6 idle
FDMA: frequency division multiple access

- channel spectrum divided into frequency bands
- each station assigned fixed frequency band
- unused transmission time in frequency bands go idle
- example: 6-station LAN, 1,3,4 have packet to send, frequency bands 2,5,6 idle
“Taking Turns” MAC protocols

channel partitioning MAC protocols:
- share channel efficiently and fairly at high load
- inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node!

“Demand-Adaptive” Controlled Protocols

- Human analogy:
  - traffic control with green/red light
    - fixed time vs. adaptive time vs. no lights at all
  - (Master-Slave based) Polling:
    - e.g., in a classroom: I am the “master” ;-
  - “Taking Turns” via token-passing:
    - e.g., a round-table panel with a single microphone
“Taking Turns” MAC Protocols

Polling:
- centralized
- master node “invites” slave nodes to transmit in turn
- concerns:
  - polling overhead
  - latency
  - single point of failure (master)

Token passing:
- distributed
- control token passed from one node to next sequentially.
- what is a token? a special control message
- concerns:
  - token overhead
  - latency
  - single point of failure (token)
Token Ring Topology

Using token-passing, nodes do not have to form a physical ring! E.g., token bus: all nodes connected via a bus, forming a logical ring!

CSci4211: Data Link Layer: Part 2
Token Release

- Release after Reception (used by Token Ring)
- Release after Transmission (used by FDDI)
Token Ring Performance

• Efficiency with “release after reception”

\[ \approx \frac{1}{1 + a} \]

where \( a = \frac{PROP}{TRANS} \)

• What is the efficiency with “release after transmission”? 

CSci4211: Data Link Layer: Part 2
Random Access Protocols

• When node has packet to send
  - transmit at full channel data rate $R$.
  - no \textit{a priori} coordination among nodes

• two or more transmitting nodes $\rightarrow$ “collision”,

• random access MAC protocol specifies:
  - how to detect or avoid collisions
  - how to recover from collisions (e.g., via delayed retransmissions)

• Examples of random access MAC protocols:
  - ALOHA
  - slotted ALOHA
  - CSMA, CSMA/CD, CSMA/CA
Pure (unslotted) ALOHA

- unslotted Aloha: simple, no synchronization
- when frame first arrives
  - transmit immediately
- collision can happen!
  - frame sent at $t_0$ collides with other frames sent in $[t_0-1, t_0+1]$
Slotted ALOHA

Assumptions
• all frames same size
• time is divided into equal size slots, time to transmit 1 frame
• nodes start to transmit frames only at beginning of slots
• nodes are synchronized
• if 2 or more nodes transmit in slot, all nodes detect collision

Operation
• when node obtains fresh frame, it transmits in next slot
• no collision, node can send new frame in next slot
• if collision, node retransmits frame in each subsequent slot with prob. p until success
Slotted ALOHA

Pros

• Single active node can continuously transmit at full rate of channel
• Highly decentralized: only slots in nodes need to be in sync
• Simple

Cons

• Collisions, wasting slots
• Idle slots
• Nodes may be able to detect collision in less than time to transmit packet
Slotted Aloha efficiency

**Efficiency** is the long-run fraction of successful slots when there’s many nodes, each with many frames to send

- Suppose N nodes with many frames to send, each transmits in slot with probability $p$
- prob that 1st node has success in a slot = $p(1-p)^{N-1}$
- prob that any node has a success = $Np(1-p)^{N-1}$

- For max efficiency with N nodes, find $p^*$ that maximizes $Np(1-p)^{N-1}$
- For many nodes, take limit of $Np^*(1-p^*)^{N-1}$ as N goes to infinity, gives $1/e = .37$

**At best:** channel used for useful transmissions 37% of time!
Pure Aloha Efficiency

\[ P(\text{success by given node}) = P(\text{node transmits}) \cdot \]
\[ P(\text{no other node transmits in } [p_0 - 1, p_0]) \cdot P(\text{no other node transmits in } [p_0, p_0 + 1]) = p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1} = p \cdot (1-p)^{2(N-1)} \]

... choosing optimum \( p \) and then letting \( n \to \infty \) ...

\[ = 1/(2e) = .18 \]

Efficiency is even worse!
Performance of Aloha Protocols

$S = \text{throughput} = \text{"goodput"}$

$G = \text{offered load} = Np$

Can we do better with random access?
Carrier Sense Multiple Access

- Aloha is inefficient (and rude):
  - doesn’t listen before talking
- CSMA: Listen before transmit
  - Human analogy: don’t interrupt others!
  - If channel idle, transmit entire packet
  - If busy, defer transmission
    - How long should we wait?
- Persistent vs. Nonpersistent CSMA
  - Nonpersistent:
    - if idle, transmit
    - if busy, wait random amount of time
  - p-persistent
    - If idle, transmit with probability p
    - If busy, wait till it becomes idle
    - If collision, wait random amount of time
- Can carrier sense avoid collisions completely?
CSMA Collisions

Collisions can still occur: propagation delay means two nodes may not hear each other’s transmission.

Collision: entire packet transmission time wasted.

Note: role of distance & propagation delay in determining collision probability.
CSMA/CD (Collision Detection)

**CSMA/CD:** carrier sensing, deferral as in CSMA
- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage

- **human analogy:** the polite conversationalist
  - talking while keep listening, stop if collision detected

- **How to detect collision?**
  - easy in wired LANs: measure signal strengths, compare transmitted, received signals
  - difficult in wireless LANs: receiver shut off while transmitting
CSMA/CD: Illustration

t₀

collision
detect/abort
t₁

time

space
Ethernet

“Dominant” LAN technology today:

- cheap $20 or less for 100 Mbps or even 1Gbps!
- first widely used LAN technology
- Simpler, cheaper than alternative technologies such as token ring LANs
- Kept up with speed race: 10, 100, 1 Gbps, 10 Gbps, 40 Gbps, and now 100 Gbps

Metcalfe’s Ethernet sketch
Ethernet Frame Format

Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

DIX frame format

<table>
<thead>
<tr>
<th>8 bytes</th>
<th>6</th>
<th>6</th>
<th>2</th>
<th>0-1500</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preamble</td>
<td>Dest addr</td>
<td>Src addr</td>
<td>Type</td>
<td>Data</td>
<td>CRC</td>
</tr>
</tbody>
</table>

IEEE 802.3 format

<table>
<thead>
<tr>
<th>8 bytes</th>
<th>6</th>
<th>6</th>
<th>2</th>
<th>0-1500</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preamble</td>
<td>Dest addr</td>
<td>Src addr</td>
<td>Length</td>
<td>Data</td>
<td>CRC</td>
</tr>
</tbody>
</table>

• Ethernet has a maximum frame size: data portion <=1500 bytes
• It has imposed a minimum frame size: 64 bytes (excluding preamble)
  If data portion <46 bytes, pad with “junk” to make it 46 bytes

Q: Why minimum frame size in Ethernet?
Fields in Ethernet Frame Format

• **Preamble:**
  - 7 bytes with pattern 10101010 followed by one byte with pattern 10101011 (SoF: start-of-frame)
  - used to synchronize receiver, sender clock rates, and identify beginning of a frame

• **Addresses:** 6 bytes
  - if adapter receives frame with matching destination address, or with broadcast address (e.g., ARP packet), it passes data in frame to net-layer protocol
  - otherwise, adapter discards frame

• **Type:** indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk)
  - 802.3: Length gives data size; “protocol type” included in data

• **CRC:** checked at receiver, if error is detected, the frame is simply dropped
Ethernet and IEEE 802.3

1-persistent CSMA/CD

- **Carrier sense**: station listens to channel first
  - Listen before talking
- **If idle, station may initiate transmission**
  - Talk if quiet
- **Collision detection**: continuously monitor channel
  - Listen while talking
- **If collision, stop transmission**
  - One talker at a time
Ethernet CSMA/CD Algorithm

1. Adaptor gets datagram from and creates frame
2. If adapter senses channel idle, it starts to transmit frame. If it senses channel busy, waits until channel idle and then transmits
3. If adapter transmits entire frame without detecting another transmission, the adapter is done with frame! Signal to network layer “transmit OK”
4. If adapter detects another transmission while transmitting, aborts and sends jam signal
5. After aborting, adapter enters exponential backoff: after the mth collision, adapter chooses a K at random from \( \{0,1,2,\ldots,2^m-1\} \). Adapter waits \( K \times 512 \) bit times and returns to Step 2
6. Quit after 16 attempts, signal to network layer “transmit error”
Jam Signal: make sure all other transmitters are aware of collision; 48 bits;

Bit time: .1 microsec for 10 Mbps Ethernet; for K=1023, wait time is about 50 msec

Exponential Backoff:
- **Goal**: adapt retransmission attempts to estimated current load
  - heavy load: random wait will be longer
- first collision: choose K from \{0,1\}; delay is \(K \times 512\) bit transmission times
- after second collision: choose K from \{0,1,2,3\}...
- after ten collisions, choose K from \{0,1,2,3,4,...,1023\}

See/interact with Java applet on AWL Web site: highly recommended !
IEEE 802.3 Parameters

• 1 bit time = time to transmit one bit
  - 10 Mbps \(\Rightarrow\) 1 bit time = 0.1 microseconds \((s)\)

• Maximum network diameter \(\leq 2.5\) km
  - Maximum 4 repeaters

• “Collision Domain”
  - Distance within which collision can be detected
  - IEEE 802.3 specifies:
    worst case collision detection time: 51.2 \(s\)

• Why minimum frame size?
  - 51.2 \(s\) \(\Rightarrow\) minimum # of bits can be transited at 10Mpbs is 512 bits \(\Rightarrow\) 64 bytes is required for collision detection
Worst Case Collision Detection Time

(a) Packet starts at time 0

(b) Packet almost at B at time t

(c) Collision at time t

(d) Noise burst gets back to A at 1.2t
CSMA/CD Efficiency

Relevant parameters
- cable length, signal speed, frame size, bandwidth

- $T_{\text{prop}} = \text{max prop between 2 nodes in LAN}$
- $t_{\text{trans}} = \text{time to transmit max-size frame}$

\[
\text{efficiency} = \frac{1}{1 + 5\frac{t_{\text{prop}}}{t_{\text{trans}}}}
\]

- Efficiency goes to 1 as $t_{\text{prop}}$ goes to 0
- Goes to 1 as $t_{\text{trans}}$ goes to infinity
- Much better than ALOHA, but still decentralized, simple, and cheap
Ethernet Technologies: 10Base2

- **10**: 10Mbps; **2**: under 200 meters max cable length
- thin coaxial cable in a bus topology
- repeaters used to connect up to multiple segments
- repeater repeats bits it hears on one interface to its other interfaces: physical layer device only!
- has become a legacy technology
10BaseT and 100BaseT

• 10/100 Mbps rate; latter called “fast ethernet”
• T stands for Twisted Pair
• Nodes connect to a hub: “star topology”; 100 m max distance between nodes and hub

Hubs are essentially physical-layer repeaters:
  - bits coming in one link go out all other links
  - no frame buffering
  - no CSMA/CD at hub: adapters detect collisions
  - provides net management functionality
100Base T (Fast) Ethernet: Issues

• 1 bit time = time to transmit one bit
  - 100 Mbps $\Rightarrow$ 1 bit time $= 0.01$ s (microseconds)

• If we keep the same “collision domain”, i.e., worst case collision detection time kept at 51.2 (microseconds)
  Q: What will be the minimum frame size?
  - 51.2 s $\Rightarrow$ minimum # of bits can be transited at 100Mpbs is 5120 bits $\Rightarrow$ 640 bytes is required for collision detection
  - This requires change of frame format and protocol!

• Or we can keep the same minimum frame size, but reduce “collision domain” or network diameter!
  • from 51.2 s to 5.12 s!
  • maximum network diameter 100 m!
Gigabit Ethernet:
• use standard Ethernet frame format
• allows for point-to-point links and shared broadcast channels
• in shared mode, CSMA/CD is used; short distances between nodes to be efficient
  - also uses hubs, called “Buffered Distributors”
• Full-Duplex at 1 Gbps for point-to-point links

- Now: 10 & 40 Gbps are widely available
- And 100 Gbps is also here!
- All are used in “point-to-point” settings with Ethernet switches
Ethernet Summary

• 1-persistent CSMA/CD
• 10Base Ethernet
  - 51.2 $s$ to seize the channel
  - Collision not possible after 51.2 $s$
  - Minimum frame size of 64 bytes
  - Binary exponential backoff
  - Works better under light load
  - Delivery time non-deterministic

• Evolution of Ethernet: Fast (100BaseT) and Gigabit Ethernet, and beyond
Token Ring (IEEE 802.5)

• Station
  - Wait for token to arrive
  - Hold the token and start data transmission
    - Maximum token holding time $\Rightarrow$ max packet size
  - Strip the data frame off the ring
    - After it has gone around the ring
  - When done, release the token to next station

• When no station has data to send
  - Token circulates continuously
  - Ring must have sufficient delay to contain the token
Ring Topology
In token passing protocols, sender is always responsible for removing the frame it has transmitted! (Why?)
## Tokens and Data Frames

A frame in the Data Link Layer consists of the following fields:

- **Start delimiter**: 8 bits
- **Access control**: 8 bits
- **Frame control**: 8 bits
- **Dest addr**: 48 bits
- **Src addr**: 48 bits
- **Body**: Variable length
- **Checksum**: 32 bits
- **End delimiter**: 8 bits
- **Frame status**: 8 bits

In the diagram, the **body** field is highlighted, indicating its variable length.
Token Ring Frame Fields

• **Access Control**
  - Token bit: 0 → token, 1 → data
  - Monitor bit: used for monitoring ring
  - Priority and reservation bits: multiple priorities

• **Frame Status**
  - Set by destination, read by sender

• **Frame control**
  - Various control frames for ring maintenance
Priority and Reservation

- Token carries priority bits
  - Only stations with frames of equal or higher priority can grab the token
- A station can make reservation
  - When a data frame goes by
  - If a higher priority has not been reserved
- A station raising the priority is responsible for lowering it again
Ring Maintenance

• Each ring has a monitor station

• How to select a monitor?
  – Election/self-promotion: CLAIM_TOKEN

• Responsibilities
  – Insert additional delay
    • To accommodate the token
  – Check for lost token
    • Regenerate token
  – Watch for orphan frames
    • Drain them off the ring
  – Watch for garbled frames
    • Clean up the ring and regenerate token
Fault Scenarios

• What to do if ring breaks?
  - Everyone participates in detecting ring breaks
  - Send beacon frames
  - Figure out which stations are down
  - By-pass them if possible

• What happens if monitor dies?
  - Everyone gets a chance to become the new king

• What if monitor goes berserk?
HERE'S YOUR PROBLEM. THE CONNECTION TO THE NETWORK IS BROKEN.

UH-OH. IT'S A "TOKEN RING" LAN. THAT MEANS THE TOKEN FELL OUT AND IT'S IN THIS ROOM SOMEPLACE.

YOU ARE THE WIND BENEATH MY WINGS.

I'LL WAIT A WEEK THEN TELL HIM THE TOKEN MUST BE IN THE "ETHERNET."
Token Ring Summary

- Stations take turns to transmit
- Only the station with the token can transmit
- Sender receives its own transmission
  - Drains its frame off the ring
- Releases token after transmission/reception
- Deterministic delivery possible
- High throughput under heavy load
Ethernet vs Token Ring

- Non-deterministic
- No delays at low loads
- Low throughput under heavy load
- No priorities
- No management overhead
- Large minimum size

- Deterministic
- Substantial delays at low loads
- High throughput under heavy load
- Multiple priorities
- Complex management
- Small frames possible
Cable Access Network

CMTS:
cable modem termination system

Internet frames, TV channels, control transmitted
downstream at different frequencies

Cable headend

CMTS

cable modem termination system

ISP

multiple 40Mbps downstream (broadcast) channels (each: 6MHz)
  - single CMTS transmits into channels
multiple 30 Mbps upstream channels (each: 6.4MHz)
  - multiple access: all users contend for certain upstream channel time slots (others assigned)
DOCSIS: data over cable service interface spec

- FDM over upstream, downstream frequency channels
- TDM upstream: some slots assigned, some have contention
  - downstream MAP frame: assigns upstream slots
  - request for upstream slots (and data) transmitted random access (binary backoff) in selected slots (“content slots”)

CSci4211: Data Link Layer: Part 2
Summary of MAC Protocols

- Why media access control?
  - Shared media: only one user can send at a time
  - Media access control: determine who has access

- MAC issues:
  - distributed, using the same channel for regulating access

- What do you do with a shared media?
  - Channel Partitioning, by time, frequency or code
    - Time Division, Code Division, Frequency Division
  - Random Access (dynamic)
    - ALOHA, S-ALOHA, CSMA, CSMA/CD
    - carrier sensing easy in some technologies (wire), hard in others (wireless)
    - CSMA/CD used in Ethernet; CSMA/CA used in WiFi/802.11
  - Taking Turns
    - polling from a central site, token passing (Bluetooth, Token Ring, FDDI)