

---

# Link Layer

EECS 489 Computer Networks

<http://www.eecs.umich.edu/courses/eecs489/w07>

Z. Morley Mao

Wednesday Feb 14, 2007

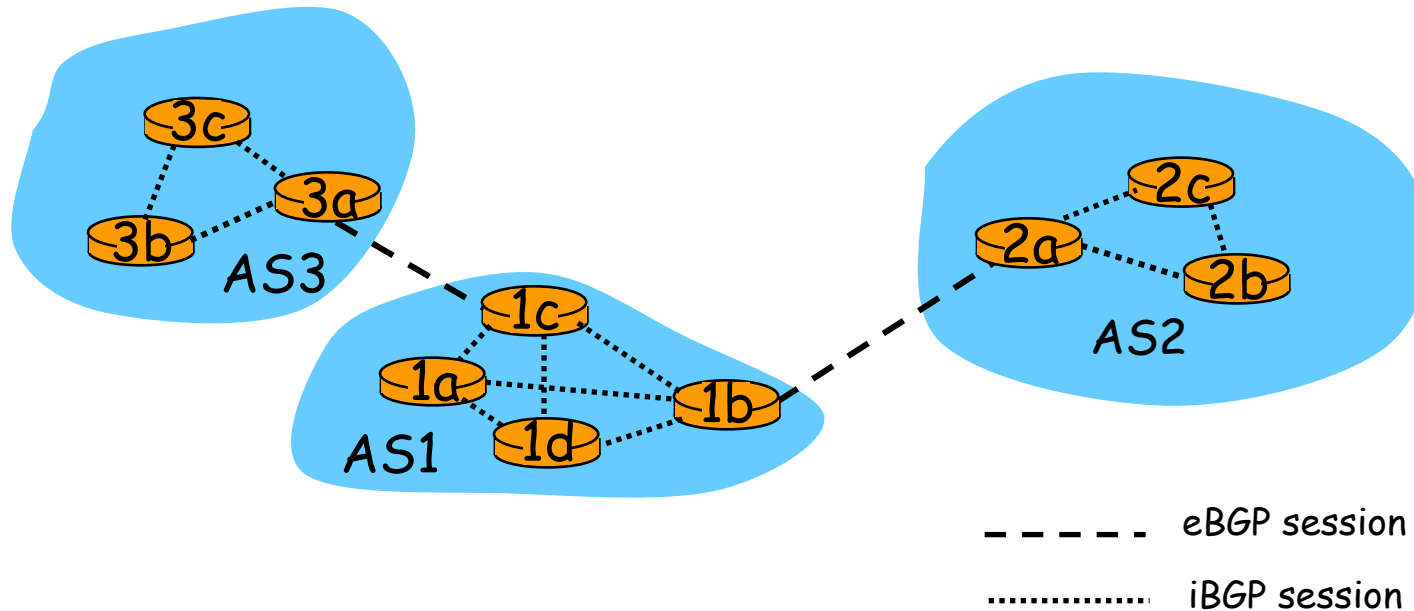
# Adminstrivia

---

- Homework 2 is posted
  - Problems from the book
  - You can either use Turnin program or turn in the homework on paper to my office.
  - Due date: next Tuesday 2/20
- Midterm 1 is in class on Wednesday March 7<sup>th</sup>
  - Please let us know if you prefer to take it early
  - Material: Chapter 1-4
  - Including half of today's lecture
  - You can have one sheet of notes for the midterm.

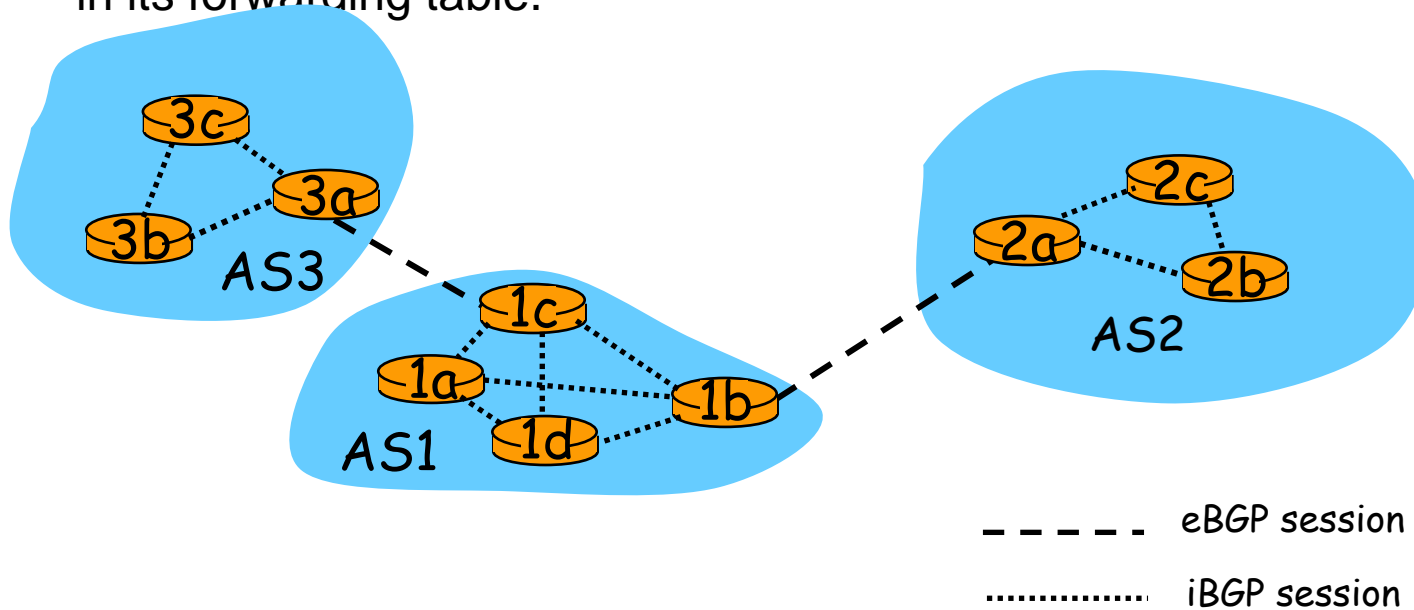
# BGP basics

- Pairs of routers (BGP peers) exchange routing info over semi-permanent TCP conctns: **BGP sessions**
- Note that BGP sessions do not correspond to physical links.
- When AS2 advertises a prefix to AS1, AS2 is *promising* it will forward any datagrams destined to that prefix towards the prefix.
  - AS2 can aggregate prefixes in its advertisement



# Distributing reachability info

- With eBGP session between 3a and 1c, AS3 sends prefix reachability info to AS1.
- 1c can then use iBGP to distribute this new prefix reach info to all routers in AS1
- 1b can then re-advertise the new reach info to AS2 over the 1b-to-2a eBGP session
- When router learns about a new prefix, it creates an entry for the prefix in its forwarding table.



# Path attributes & BGP routes

---

- When advertising a prefix, advert includes BGP attributes.
  - prefix + attributes = “route”
- Two important attributes:
  - **AS-PATH**: contains the ASs through which the advert for the prefix passed: AS 67 AS 17
  - **NEXT-HOP**: Indicates the specific internal-AS router to next-hop AS. (There may be multiple links from current AS to next-hop-AS.)
- When gateway router receives route advert, uses **import policy** to accept/decline.

# BGP route selection

---

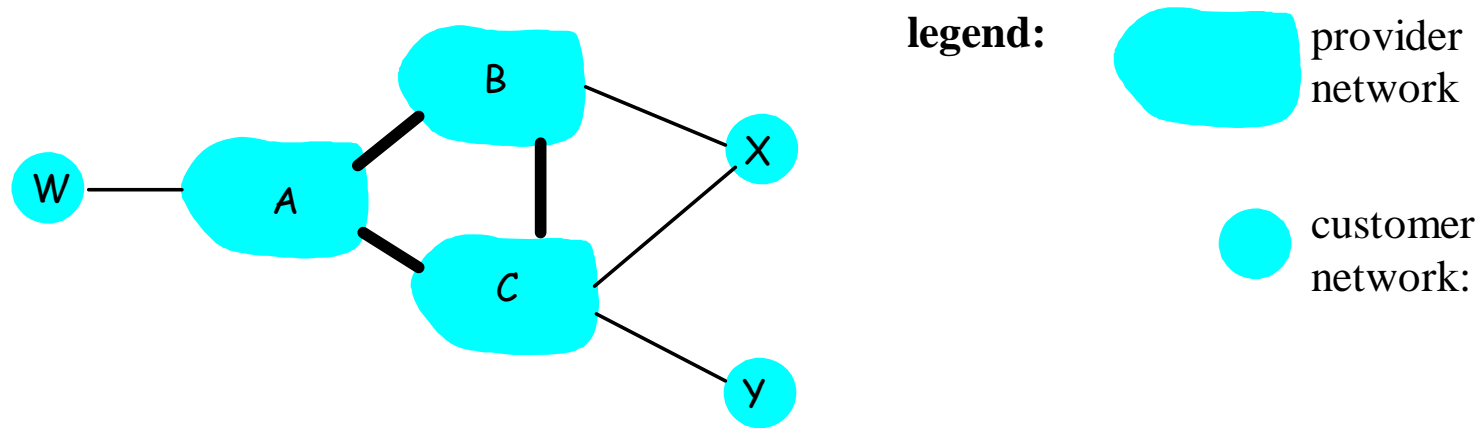
- Router may learn about more than 1 route to some prefix. Router must select route.
- Elimination rules:
  1. Local preference value attribute: policy decision
  2. Shortest AS-PATH
  3. Closest NEXT-HOP router: hot potato routing
  4. Additional criteria

# BGP messages

---

- BGP messages exchanged using TCP.
- BGP messages:
  - **OPEN**: opens TCP connection to peer and authenticates sender
  - **UPDATE**: advertises new path (or withdraws old)
  - **KEEPALIVE** keeps connection alive in absence of UPDATES; also ACKs OPEN request
  - **NOTIFICATION**: reports errors in previous msg; also used to close connection

## BGP routing policy



A,B,C are **provider networks**

X,W,Y are customer (of provider networks)

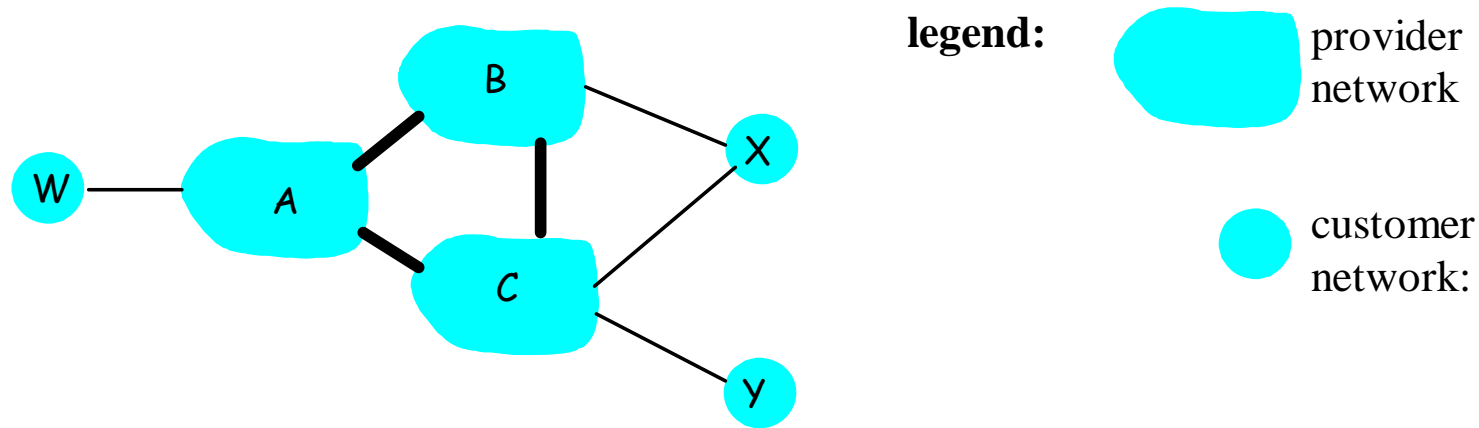
X is **dual-homed**: attached to two networks

X does not want to route from B via X to C

.. so X will not advertise to B a route to C



## BGP routing policy (2)



A advertises to B the path AW

B advertises to X the path BAW

Should B advertise to C the path BAW?

No way! B gets no “revenue” for routing CBAW since neither W nor C are B’s customers

B wants to force C to route to w via A

B wants to route *only* to/from its customers!

# Why different Intra- and Inter-AS routing ?

---

## Policy:

- Inter-AS: admin wants control over how its traffic routed, who routes through its net.
- Intra-AS: single admin, so no policy decisions needed, exception: VPN networks.

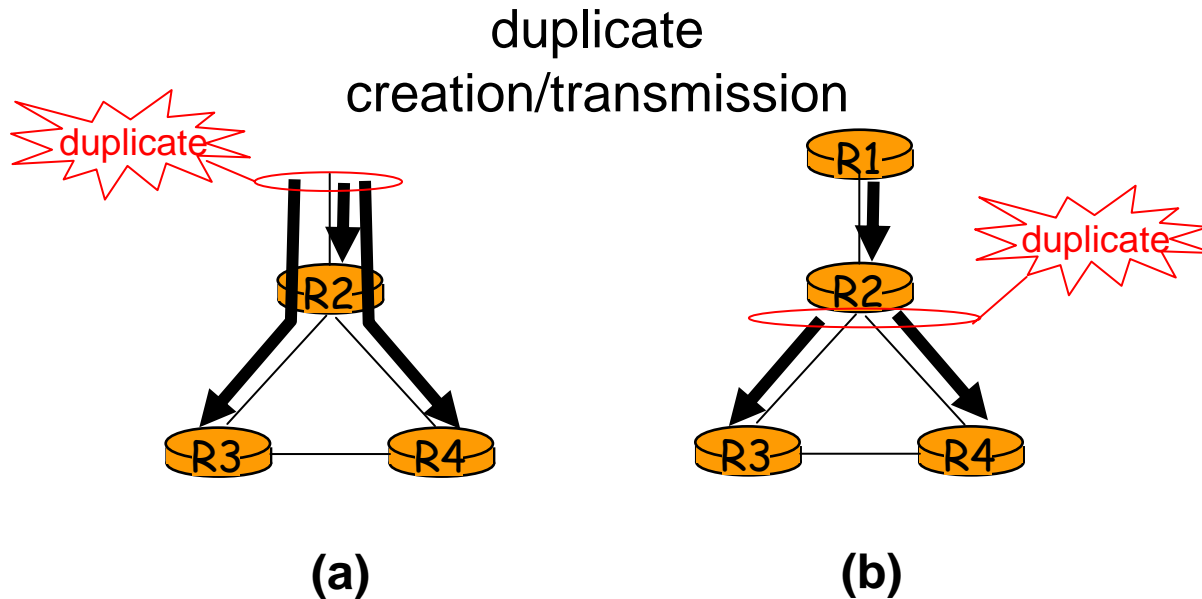
## Scale:

- hierarchical routing saves table size, reduced update traffic

## Performance:

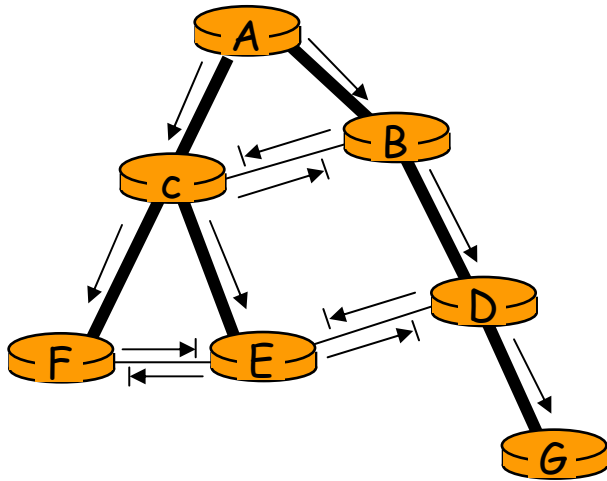
- Intra-AS: can focus on performance
- Inter-AS: policy may dominate over performance

# Broadcast routing



Source-duplication versus in-network duplication.  
(a) source duplication, (b) in-network duplication

# How to get rid of duplicates?

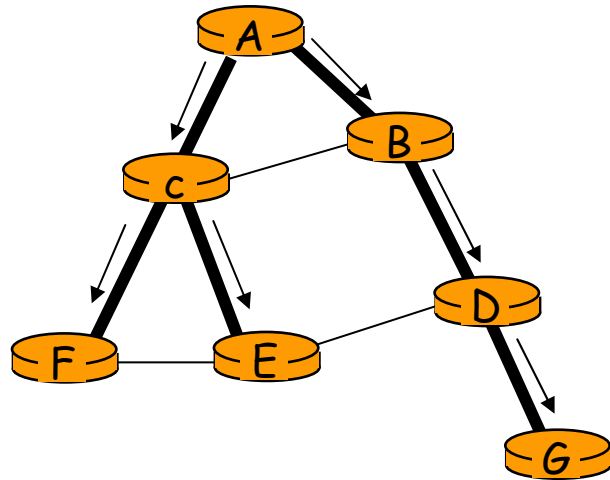


Reverse path forwarding

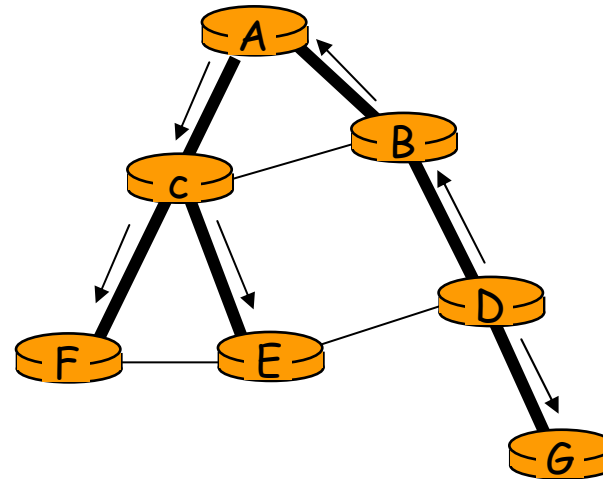
- Sequence-number-controlled flooding
  - Broadcast sequence number
  - Source node address
- Only forward if packet arrived on the link on its own shortest unicast path back to source

# Spanning tree to the rescue

- Spanning-tree broadcast
  - A tree containing every node, no cycles



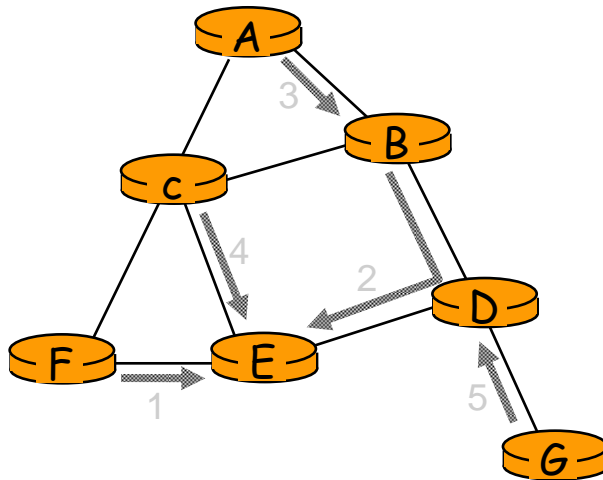
(a) Broadcast initiated at A



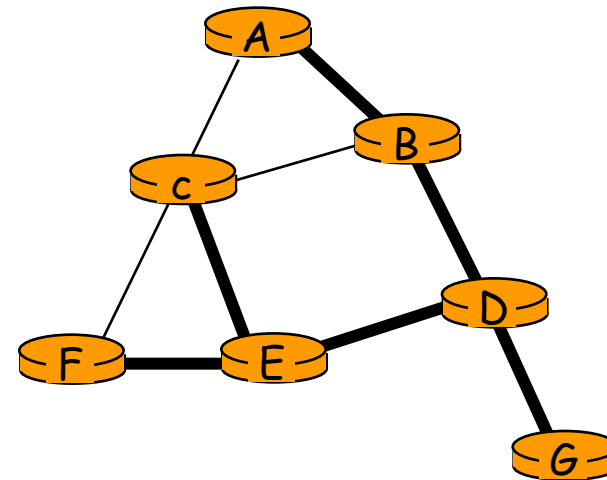
(b) Broadcast initiated at D

Broadcast along a spanning tree

# How to construct a spanning tree?



(a) Stepwise construction of spanning tree



(b) Constructed spanning tree


Center-based construction of a spanning tree

- E is the center of the tree
- Is this a minimum spanning tree?

---

**How is BGP relevant to the us?**

# Level 3 depeers with Cogent!



NEWS OF CHANGE

Tell us what you think

CNET tech sites: Product reviews | Shop | Tech news | Downloads

Welcome. Please [log in](#) or [register](#)

My News Readers' Choice Extra Blogs

Front Door Business Tech Cutting Edge Access Threats Media 2.0 Market

Search

## Network feud leads to Net blackout


By [John Borland](#)  
Staff Writer, CNET News.com  
Published: October 5, 2005, 5:00 PM PDT

[TalkBack](#) [E-mail](#) [Print](#) [TrackBack](#)

**Two major Internet backbone companies are feuding, potentially cutting off significant swaths of the Internet for some of each other's customers.**

On Wednesday, network company Level 3 Communications cut off its direct "peering" connections to another big network company called Cogent Communications. That technical action means that some customers on each company's network now will find it impossible, or slower, to get to Web sites on the other

advertisement



The HP ProLiant BL20p with Intel® Xeon™ Processors and Performance Management Pack.

[Download white paper](#) [Storage info](#)

**DID YOU KNOW?**  
Select a tab below to set your default view.

THE BIG RELATED WHAT'S HOT L



# Botnet of 100,000 PCs crushed!

Part of the **TechWeb** Business Technology Network

**InformationWeek**  
BUSINESS INNOVATION POWERED BY TECHNOLOGY

CMP  
United Business Media

HOME  
EVENTS

WINDOWS SOFTWARE HARDWARE SECURITY

Security Pipeline | Viruses and Patches | Privacy | Spam | Windows Security | Se

## SECURITY

### Dutch Police Crush Big 'Botnet,' Arrest Trio

Oct. 10, 2005

**A huge network of 100,000 PCs was used to conduct a denial-of-service attack against an unidentified U.S. company in an extortion attempt, and for many other nefarious deeds, according to Dutch police.**

By Gregg Keizer  
[TechWeb News](#)

Dutch police arrested three men for creating a botnet of more than 100,000 compromised PCs, authorities in the Netherlands said Friday. They allege the botnet was used in an attempt to extort a U.S. company, to steal PayPal and eBay accounts, and to install adware and spyware.

The pinch is among the biggest [botnet](#) scores ever for law enforcement, Dutch authorities said. "With 100,000 infected computers, the dismantled botnet is one of the largest ever seen" the Public Prosecution Service (Openbaar

E-Mail This Article  
Print This Article  
Discuss This Article  
Write To An Editor  
Subscribe To InformationWeek

# Up Until Now.....

---

- Short-term contention is loss-less
  - main resource (link bandwidth) is controlled by router
  - router deals with short-term contention by queuing packets
  - switch algorithms and router buffers ensure no packets are dropped due to short-term contention
  
- We have focused on long-term contention
  - queuing schemes (FQ, FIFO, RED, etc.)
  - end-to-end congestion control (TCP)

# What's New in This Lecture?

---

- Short-term contention leads to loss!
- Lecture deals with networking over shared media
  - long-range radio
  - ethernet
  - short-range radio
- Also known as “multiple-access”
  - don't go through central router to get access to link
  - instead, multiple users can access shared medium

# Medium Access Protocols

---

- Channel partitioning
  - Divide channel into smaller “pieces” (e.g., time slots, frequency)
  - Allocate a piece to node for exclusive use
- Random access
  - Allow collisions
  - “recover” from collisions
- “Taking-turns”
  - Tightly coordinate shared access to avoid collisions

# Problem in a Nutshell

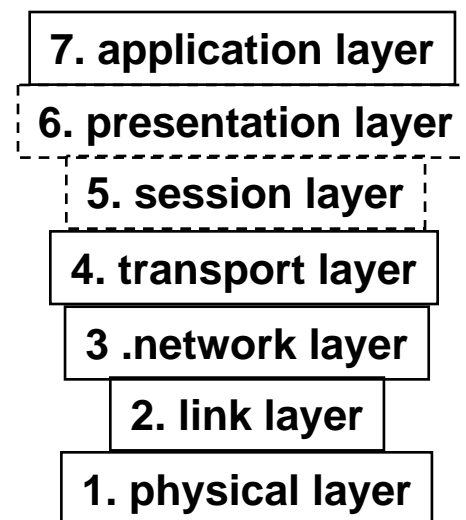
---

- Shared medium
  - If two users send at the same time, collision results in no packet being received (interference)
  - If no users send, channel goes idle
  - Thus, want to have only one user send at a time
- Want high network utilization
  - TDMA doesn't give high utilization
- Want simple distributed algorithm
  - no fancy token-passing schemes that avoid collisions

# What Layer?

---

- Where should short-term contention be handled?
- Network layer?
- Application layer?
- Link layer?



# The Data Link Layer

---

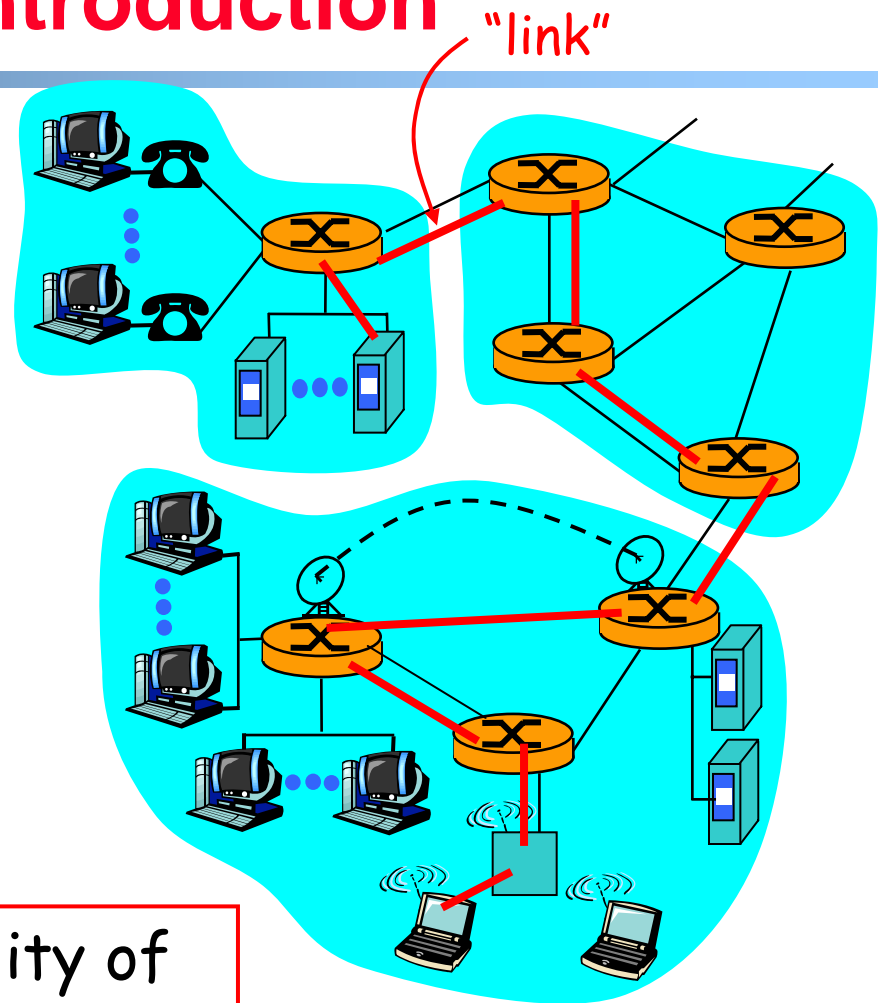
## Our goals:

- understand principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
  - reliable data transfer, flow control: *done!*
- instantiation and implementation of various link layer technologies

# Link Layer: Introduction

## Some terminology:

- hosts and routers are **nodes**
- communication channels that connect adjacent nodes along communication path are **links**
  - wired links
  - wireless links
  - LANs
- layer-2 packet is a **frame**, encapsulates datagram



**data-link layer** has responsibility of transferring datagram from one node to adjacent node over a link



# Link layer: context

- Datagram transferred by different link protocols over different links:
  - 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 rdt over link

## transportation analogy

- trip from Princeton to Lausanne
  - limo: Princeton to JFK
  - plane: JFK to Geneva
  - train: Geneva to Lausanne
- tourist = **datagram**
- transport segment = **communication link**
- transportation mode = **link layer protocol**
- travel agent = **routing algorithm**

# Link Layer Services

---

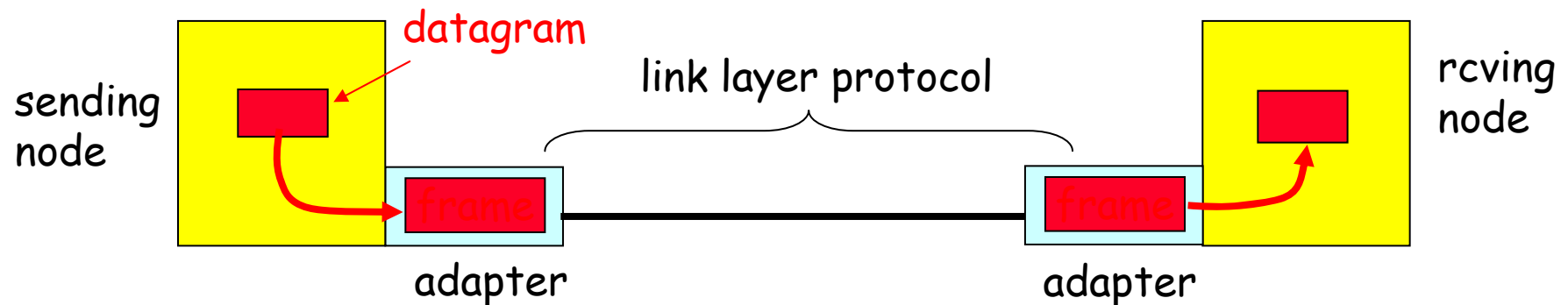
- **Framing, link access:**
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - “MAC” addresses used in frame headers to identify source, dest
    - different from IP address!
- **Reliable delivery between adjacent nodes**
  - we learned how to do this already (chapter 3)!
  - 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?

# Link Layer Services (more)

---

- *Flow Control:*
  - pacing between adjacent sending and receiving nodes
- *Error Detection:*
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame
- *Error Correction:*
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission
- *Half-duplex and full-duplex*
  - with half duplex, nodes at both ends of link can transmit, but not at same time

# Adaptors Communicating



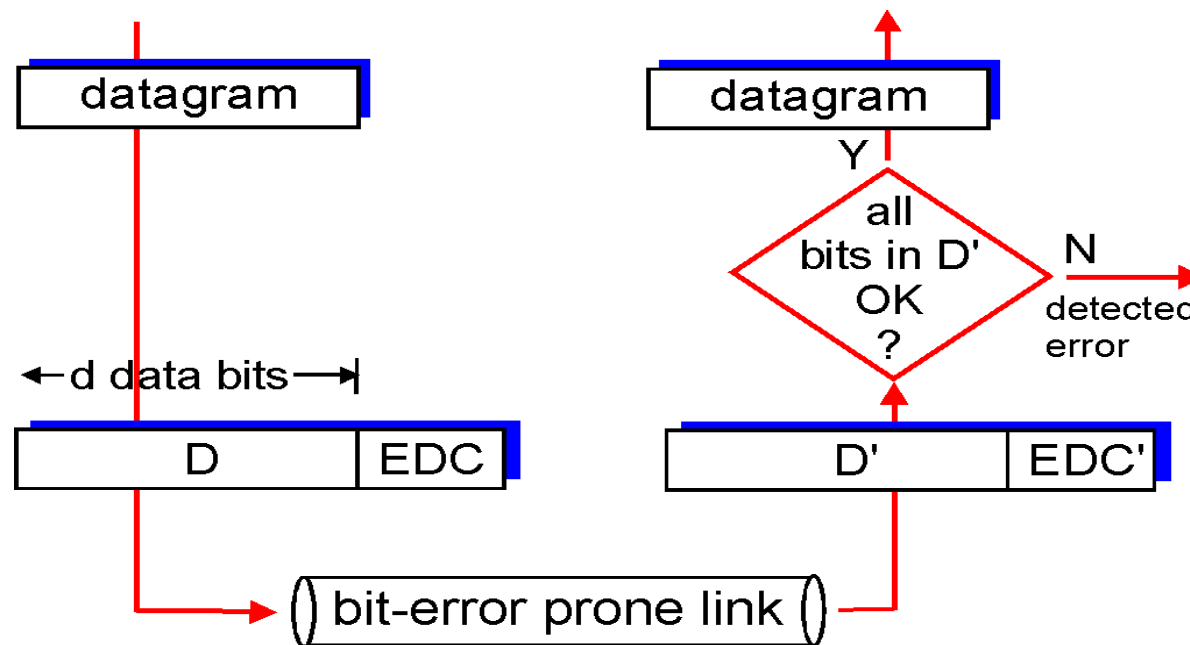
- link layer implemented in “adaptor” (aka NIC)
  - Ethernet card, PCMCIA card, 802.11 card
- sending side:
  - encapsulates datagram in a frame
  - adds error checking bits, rdt, flow control, etc.
- receiving side
  - looks for errors, rdt, flow control, etc
  - extracts datagram, passes to rcving node
- adaptor is semi-autonomous
- link & physical layers

# Error Detection

EDC= Error Detection and Correction bits (redundancy)

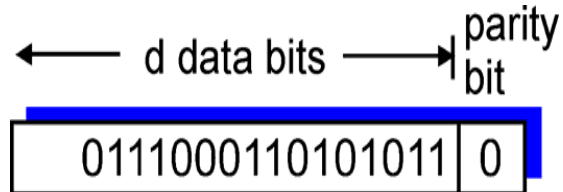
D = Data protected by error checking, may include header fields

- Error detection not 100% reliable!
  - protocol may miss some errors, but rarely
  - larger EDC field yields better detection and correction

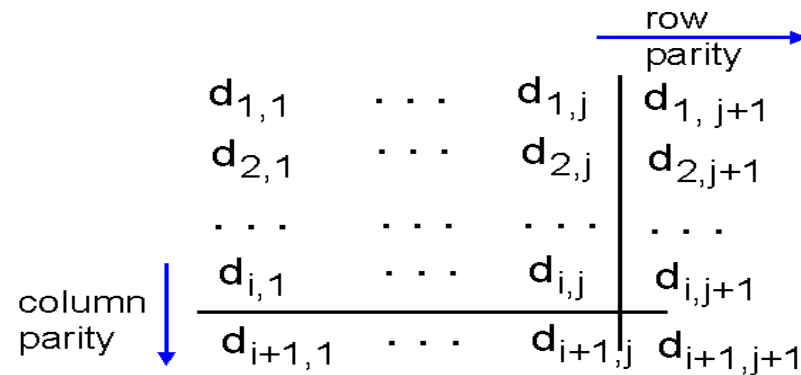


# Parity Checking

## Single Bit Parity: Detect single bit errors



## Two Dimensional Bit Parity: Detect *and correct* single bit errors



1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

*no errors*

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

↑ parity error  
↓ parity error  
*correctable single bit error*

# Internet checksum

---

Goal: detect “errors” (e.g., flipped bits) in transmitted segment (note: used at transport layer *only*)

## Sender:

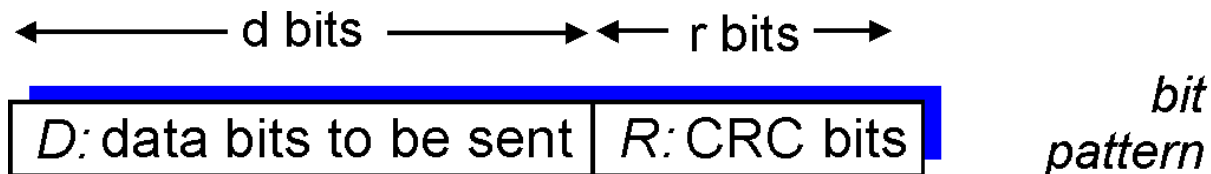
- treat segment contents as sequence of 16-bit integers
- checksum: addition (1’s complement sum) of segment contents
- sender puts checksum value into UDP checksum field

## Receiver:

- compute checksum of received segment
- check if computed checksum equals checksum field value:
  - NO - error detected
  - YES - no error detected. *But maybe errors nonetheless?*  
More later ....

# Checksumming: Cyclic Redundancy Check

- view data bits, **D**, as a binary number
- choose  $r+1$  bit pattern (generator), **G**
- goal: choose  $r$  CRC bits, **R**, such that
  - $\langle D, R \rangle$  exactly divisible by  $G$  (modulo 2)
  - receiver knows  $G$ , divides  $\langle D, R \rangle$  by  $G$ .  
If non-zero remainder: error detected!
  - can detect all burst errors less than  $r+1$  bits
- widely used in practice (ATM)



$$D * 2^r \text{ XOR } R$$

*mathematical formula*



## CRC Example

Want:

$$D \cdot 2^r \text{ XOR } R = nG$$

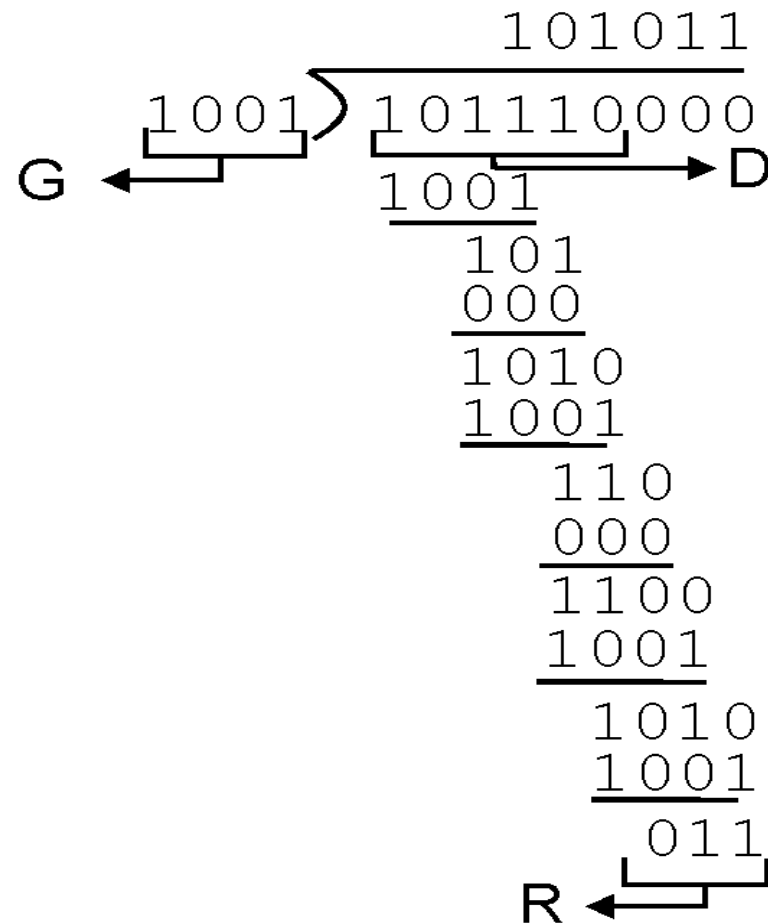
*equivalently:*

$$D \cdot 2^r = nG \text{ XOR } R$$

*equivalently:*

if we divide  $D \cdot 2^r$  by  $G$ ,  
want remainder  $R$

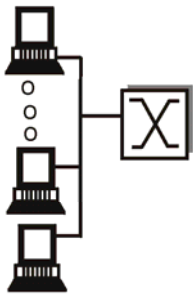
$$R = \text{remainder} \left[ \frac{D \cdot 2^r}{G} \right]$$



# Multiple Access Links and Protocols

Two types of “links”:

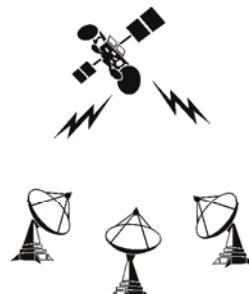
- point-to-point
  - PPP for dial-up access
  - point-to-point link between Ethernet switch and host
- **broadcast** (shared wire or medium)
  - traditional Ethernet
  - upstream HFC
  - 802.11 wireless LAN



shared wire  
(e.g. Ethernet)



shared wireless  
(e.g. Wavelan)



satellite



ZZZZZZZZZZZZZZZZ



cocktail party

# Multiple Access protocols

---

- single shared broadcast channel
- two or more simultaneous transmissions by nodes: interference
  - **collision** if node receives two or more signals at the same 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!
  - no out-of-band channel for coordination

# Ideal Multiple Access Protocol

---

## Broadcast channel of rate $R$ bps

1. When one node wants to transmit, it can send at rate  $R$ .
2. When  $M$  nodes want to transmit, each can send at average rate  $R/M$
3. Fully decentralized:
  - no special node to coordinate transmissions
  - no synchronization of clocks, slots
4. Simple

# MAC Protocols: a taxonomy

---

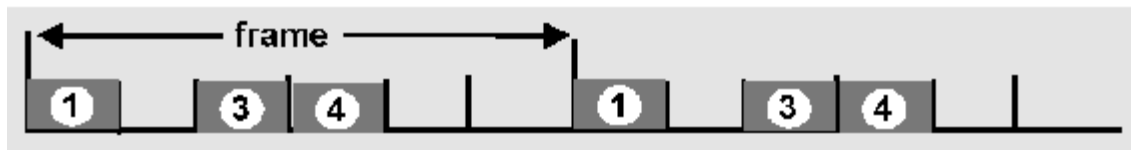
Three broad classes:

- **Channel Partitioning**
  - divide channel into smaller “pieces” (time slots, frequency, code)
  - allocate piece to node for exclusive use
- **Random Access**
  - channel not divided, allow collisions
  - “recover” from collisions
- **“Taking turns”**
  - Nodes take turns, but nodes with more to send can take longer turns

# Channel Partitioning MAC protocols: TDMA

## TDMA: time division multiple access

- access to channel in "rounds"
- each station gets fixed length slot (length = pkt trans time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle

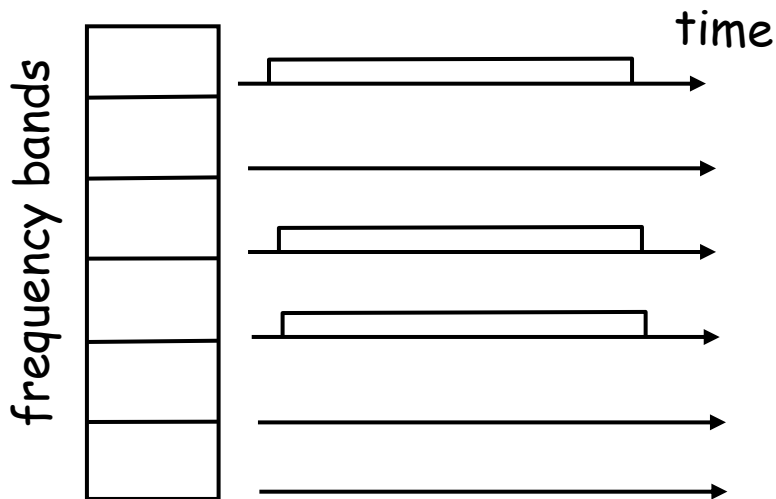


- TDM (Time Division Multiplexing): channel divided into N time slots, one per user; inefficient with low duty cycle users and at light load.
- FDM (Frequency Division Multiplexing): frequency subdivided.

# Channel Partitioning MAC protocols: FDMA

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 pkt, frequency bands 2,5,6 idle



- TDM (Time Division Multiplexing): channel divided into N time slots, one per user; inefficient with low duty cycle users and at light load.
- FDM (Frequency Division Multiplexing): frequency subdivided.

# Random Access Protocols

---

- When node has packet to send
  - transmit at full channel data rate  $R$ .
  - no *a priori* coordination among nodes
- two or more transmitting nodes → “collision”,
- **random access MAC protocol** specifies:
  - how to detect collisions
  - how to recover from collisions (e.g., via delayed retransmissions)
- Examples of random access MAC protocols:
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA



# Slotted ALOHA

---

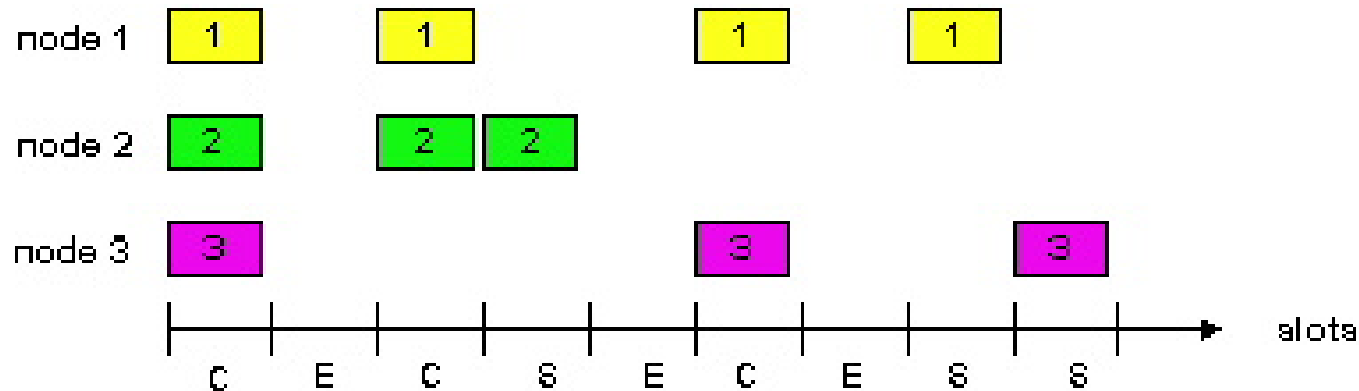
## Assumptions

- 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
- clock synchronization

# Slotted Aloha efficiency

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

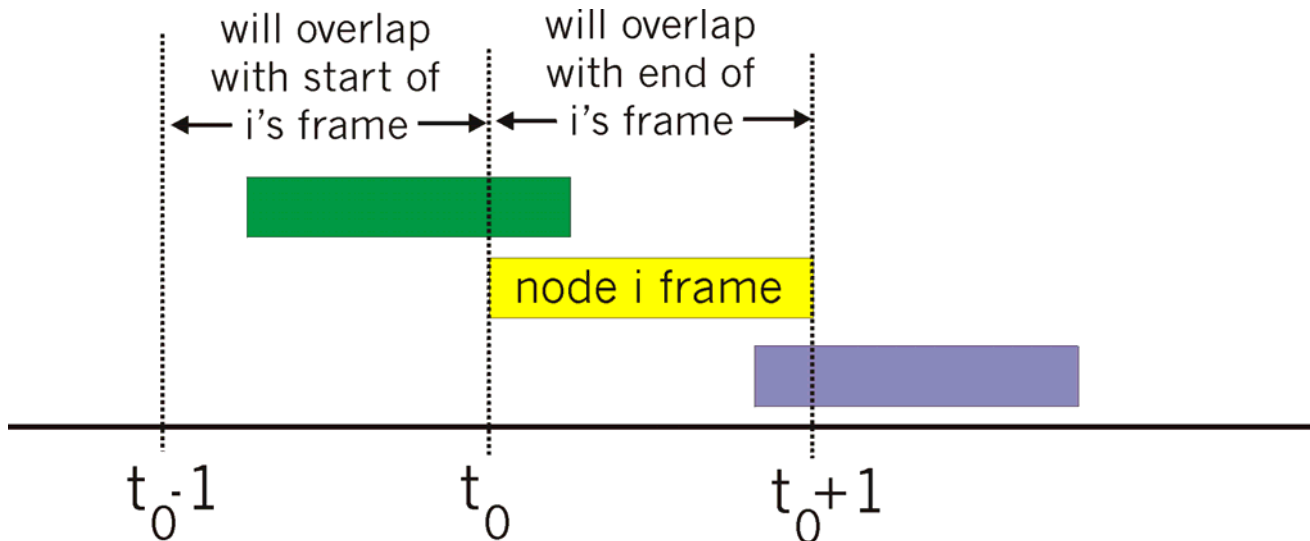
- Suppose  $N$  nodes with many frames to send, each transmits in slot with probability  $p$
- prob that node 1 has success in a slot  
 $= p(1-p)^{N-1}$
- prob that any node has a success  $= Np(1-p)^{N-1}$

- 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 (unslotted) ALOHA

- unslotted Aloha: simpler, no synchronization
- when frame first arrives
  - transmit immediately
- collision probability increases:
  - frame sent at  $t_0$  collides with other frames sent in  $[t_0-1, t_0+1]$



## Pure Aloha efficiency

---

$P(\text{success by given node}) = P(\text{node transmits}) \cdot$

$P(\text{no other node transmits in } [p_0-1, p_0]) \cdot$

$P(\text{no other node transmits in } [p_0-1, p_0])$

$$= p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1}$$

$$= p \cdot (1-p)^{2(N-1)}$$

... choosing optimum  $p$  and then letting  $n \rightarrow \infty$  ...

Even worse !  $= 1/(2e) = .18$

# Why is this better than TDMA?

---

- In TDMA, you always have to wait your turn
  - delay proportional to number of sites
- In Aloha, can send immediately
- Aloha gives much lower delays, at the price of lower utilization (as we will see)

# Slotted Aloha

---

- Divide time into slots
- Only start transmission at beginning of slots
- Decreases chance of “partial collisions”

# CSMA (Carrier Sense Multiple Access)

---

**CSMA**: listen before transmit:

If channel sensed idle: transmit entire frame

- If channel sensed busy, defer transmission
  
- Human analogy: don't interrupt others!



# CSMA collisions

collisions *can still occur*:

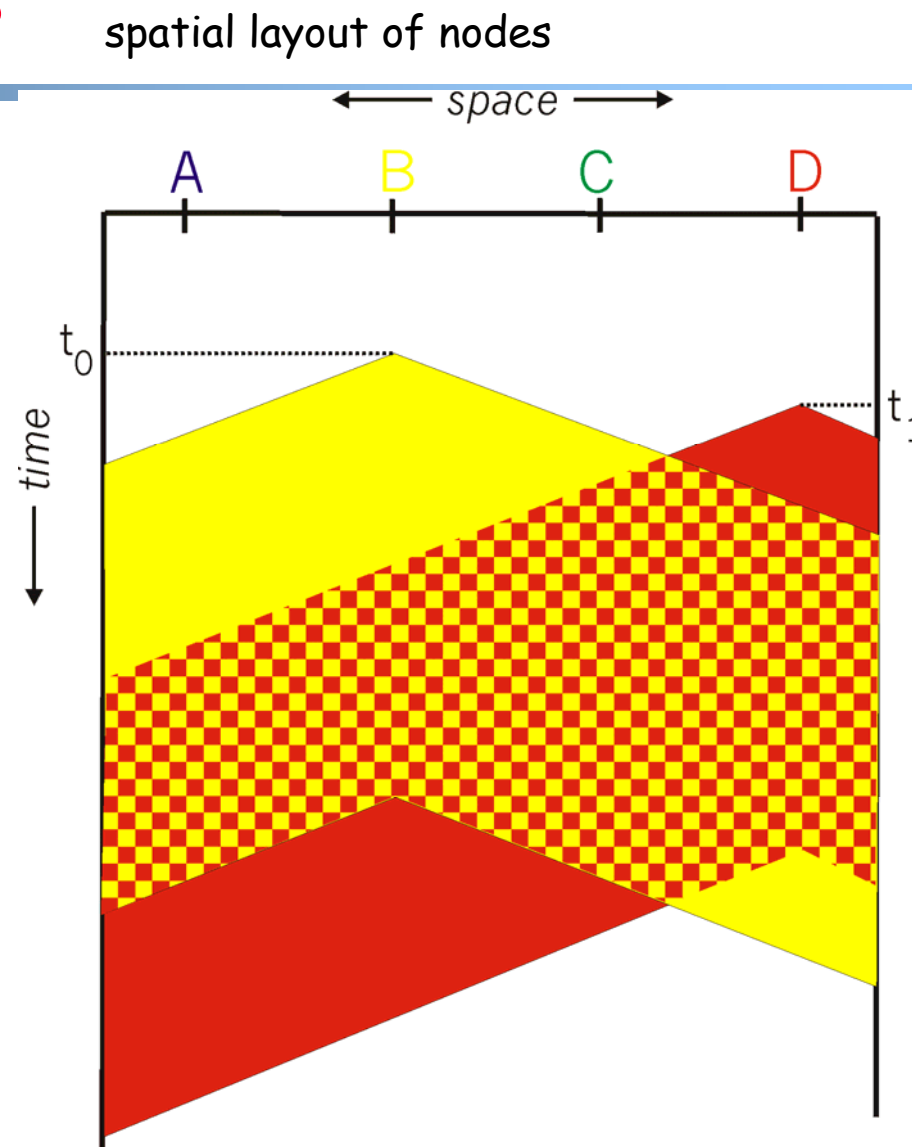
propagation delay means  
two nodes may not hear  
each other's transmission

collision:

entire packet transmission  
time wasted

note:

role of distance & propagation  
delay in determining collision  
probability



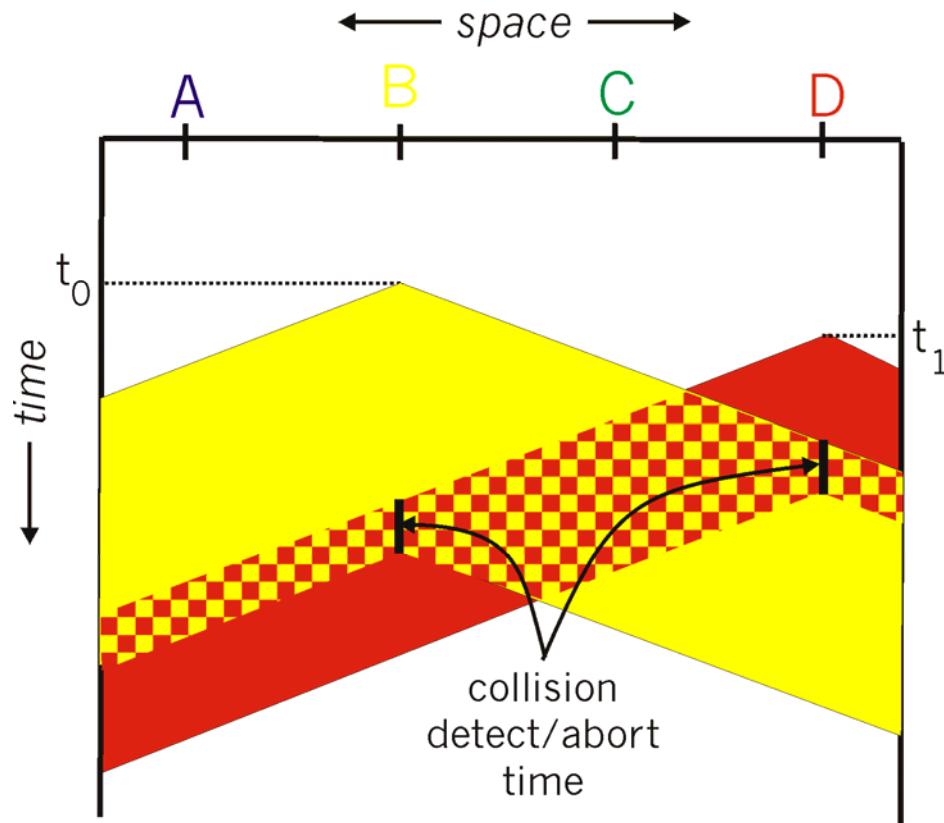
# CSMA/CD (Collision Detection)

---

**CSMA/CD:** carrier sensing, deferral as in CSMA

- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage
- collision detection:
  - easy in wired LANs: measure signal strengths, compare transmitted, received signals
  - difficult in wireless LANs: receiver shut off while transmitting
- human analogy: the polite conversationalist

# CSMA/CD collision detection



# “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!

## Random access MAC protocols

- efficient at low load: single node can fully utilize channel
- high load: collision overhead

## “taking turns” protocols

look for best of both worlds!

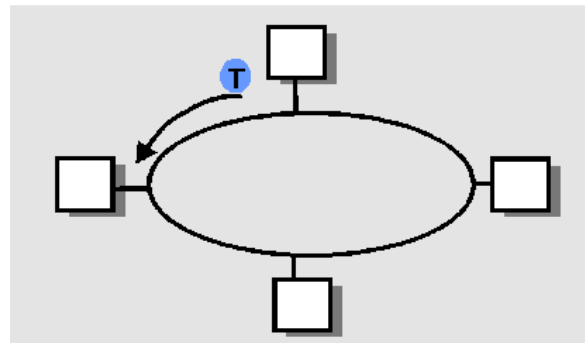
# “Taking Turns” MAC protocols

## Polling:

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

## Token passing:

- control **token** passed from one node to next sequentially.
- token message
- concerns:
  - token overhead
  - latency
  - single point of failure (token)



# Summary of MAC protocols

---

- What do you do with a shared media?
  - Channel Partitioning, by time, frequency or code
    - Time Division, Frequency Division
  - Random partitioning (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 802.11
  - Taking Turns
    - polling from a central site, token passing