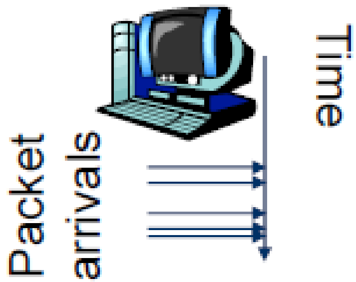
Computer Networks

Lecture 8: Data Link layer

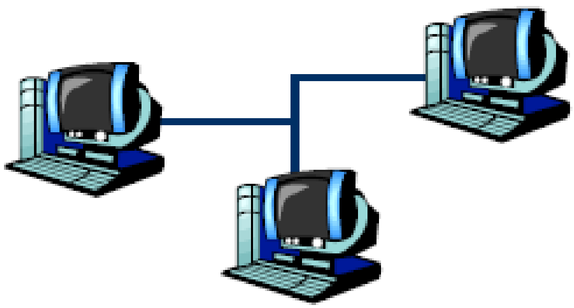
>> MAC sublayer

Based on slides from D. Choffnes Northeastern U. and P. Gill from StonyBrook University
Revised Autumn 2015 by S. Laki

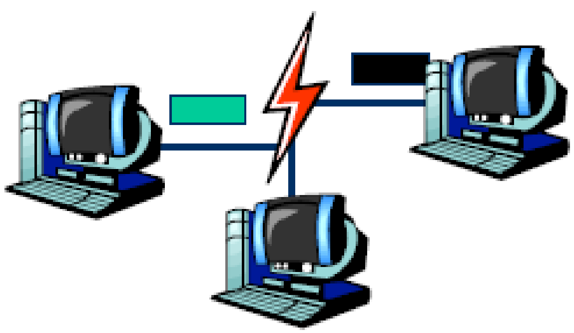
Dynamic Channel Allocation in LANs and MANs



1. Station Model.
   - N terminals/hosts
   - The prob. of a frame being generated in Δt is $\lambda \Delta t$, where the arrival rate is λ.



2. Single Channel Assumption.
   - All stations are equivalent
   - A single channel is available for all communications

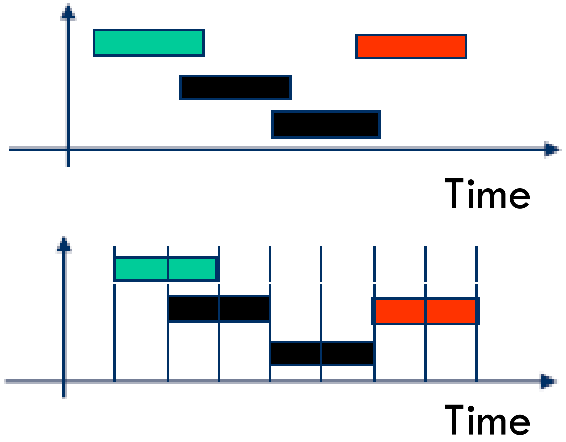


3. Collision Assumption.
   - If two frames are transmitted simultaneously, they overlap in time which results a garbled signal
   - This event is called collision

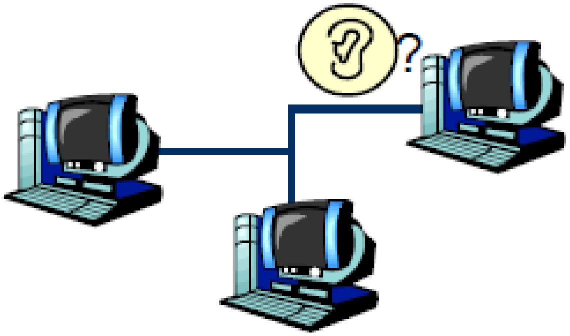
4. Continuous Time VS Slotted Time.

5. Carrier Sense VS No Carrier Sense.

Dynamic Channel Allocation in LANs and MANs



4. Continuous Time VS Slotted Time.



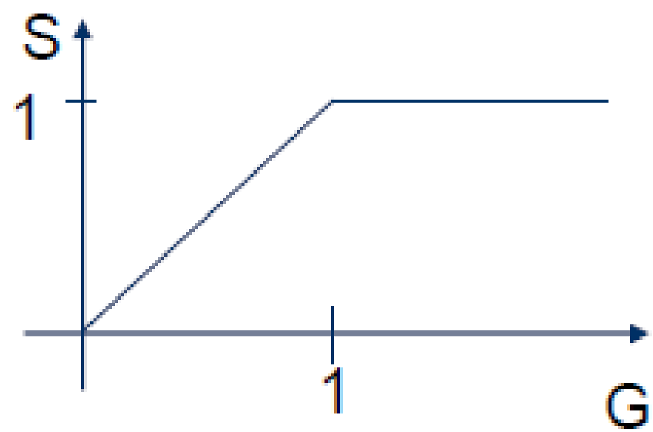
5. Carrier Sense VS No Carrier Sense.

How can the efficiency be measured?

- **Throughput (S)**
  - Number of packets/frames transmitted in a time unit (successfully)
- **Delay**
  - The time needs for transmitting a packet
- **Fairness**
  - All the terminals are treated as equals

Throughput and offered load

- **Offered load (G)**
  - The number of packets in a time unit that the protocol must handle
  - G>1: overloading



- **An ideal protocol**
  - If G<1, S=G
  - If G≥1, S=1
  - where sending out a packet takes 1 time unit.

Strategies for Media Access

- Channel partitioning
  - Divide the resource into small pieces
  - Allocate each piece to one host
  - Example: Time Division Multi-Access (TDMA) cellular
  - Example: Frequency Division Multi-Access (FDMA) cellular
- Taking turns
  - Tightly coordinate shared access to avoid collisions
  - Example: Token ring networks
- Contention
  - Allow collisions, but use strategies to recover
  - Examples: Ethernet, Wifi

Contention MAC Goals

- Share the medium
  - Two hosts sending at the same time collide, thus causing interference
  - If no host sends, channel is idle
  - Thus, want one user sending at any given time
- High utilization
  - TDMA is low utilization
  - Just like a circuit switched network
- Simple, distributed algorithm
  - Multiple hosts that cannot directly coordinate
  - No fancy (complicated) token-passing schemes

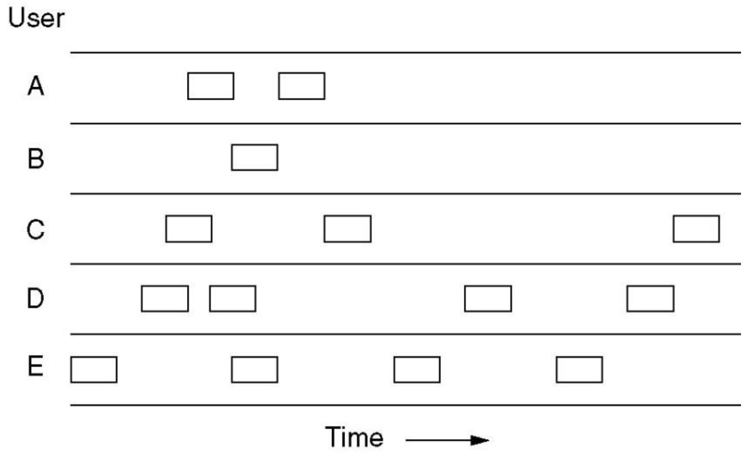
Contention Protocol Evolution

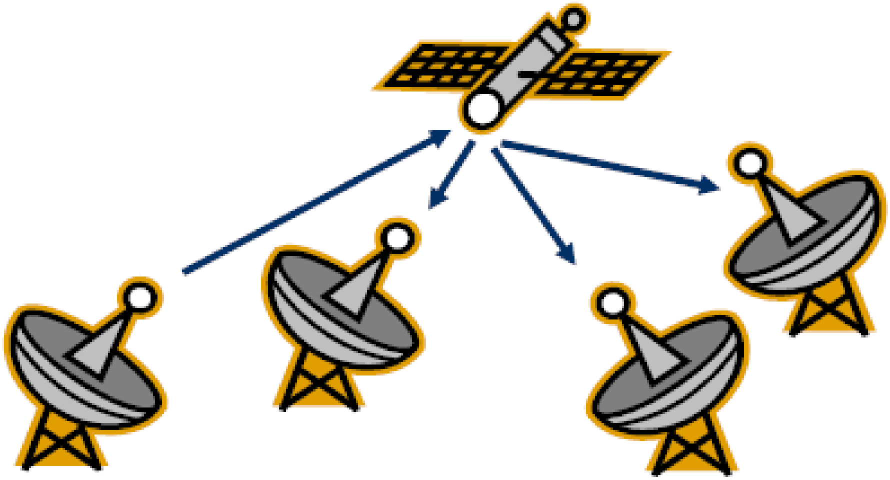
- ALOHA
  - Developed in the 70's for packet radio networks
  - Stations transmit data immedately
    - If there is a collision, it retransmits the packet later.
- Slotted ALOHA
  - Start transmissions only at fixed time slots
  - Significantly fewer collisions than ALOHA
- Carrier Sense Multiple Access (CSMA)
  - Start transmission only if the channel is idle
- CSMA / Collision Detection (CSMA/CD)
  - Stop ongoing transmission if collision is detected

Pure ALOHA

- The goal was to use low-cost commercial radio equipment to connect users on Oahu and the other Hawaiian islands with a central time-sharing computer on the main Oahu campus.

- Algorithm was developed by Uni. of Hawaii
  - **If you have data to send, send the data**
  - Low-cost and very simple





ALOHA

- Topology: radio broadcast with multiple stations
- Protocol:
  - Stations transmit data immediately
  - Receivers ACK all packets
  - No ACK = collision, wait a random time then retransmit

- Simple, but radical concept
- Previous attempts all divided the channel
  - TDMA, FDMA, etc.
- Optimized for the common case: few senders

Performance analysis -Poisson Process

- The Poisson Process is a celebrated model used in Queuing Theory for "random arrivals". Assumptions leading to this model include:
  - The probability of an arrival during a short time interval Δt is proportional to the length of the interval, and does not depend on the origin of the time interval (memory-less property)
  - The probability of having multiple arrivals during a short time interval Δt approaches zero.

Performance analysis - Poisson Distribution

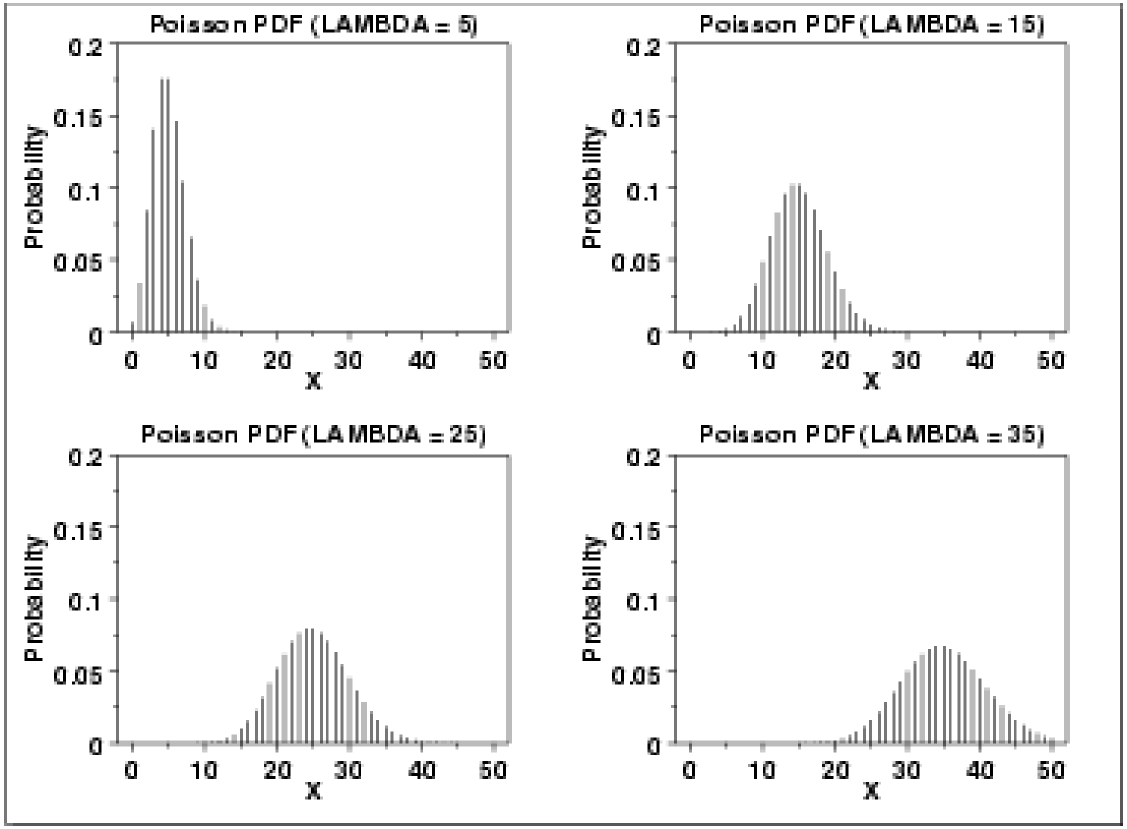
The probability of having *k* arrivals during a time interval of length *t* is given by:

$$P_k(t) = \frac{(\lambda t)^k e^{-\lambda t}}{k!}$$

where λ is the arrival rate. Note that this is a single-parameter model; all we have to know is λ.

FYI: Poisson Distribution

- The following is the plot of the Poisson probability density function for four values of λ.



Analysis of Pure ALOHA

- Notation:
  - T_f = frame time (processing, transmission, propagation)
  - *S*: Average number of successful transmissions per T_f ; that is, the *throughput*
  - *G*: Average number of total frames transmitted per T_f
  - *D*: Average delay between the time a packet is ready for transmission and the completion of successful transmission.
- We will make the following assumptions
  - All frames are of constant length
  - The channel is noise-free; the errors are only due to collisions.
  - Frames do not queue at individual stations
  - The channel acts as a Poisson process.

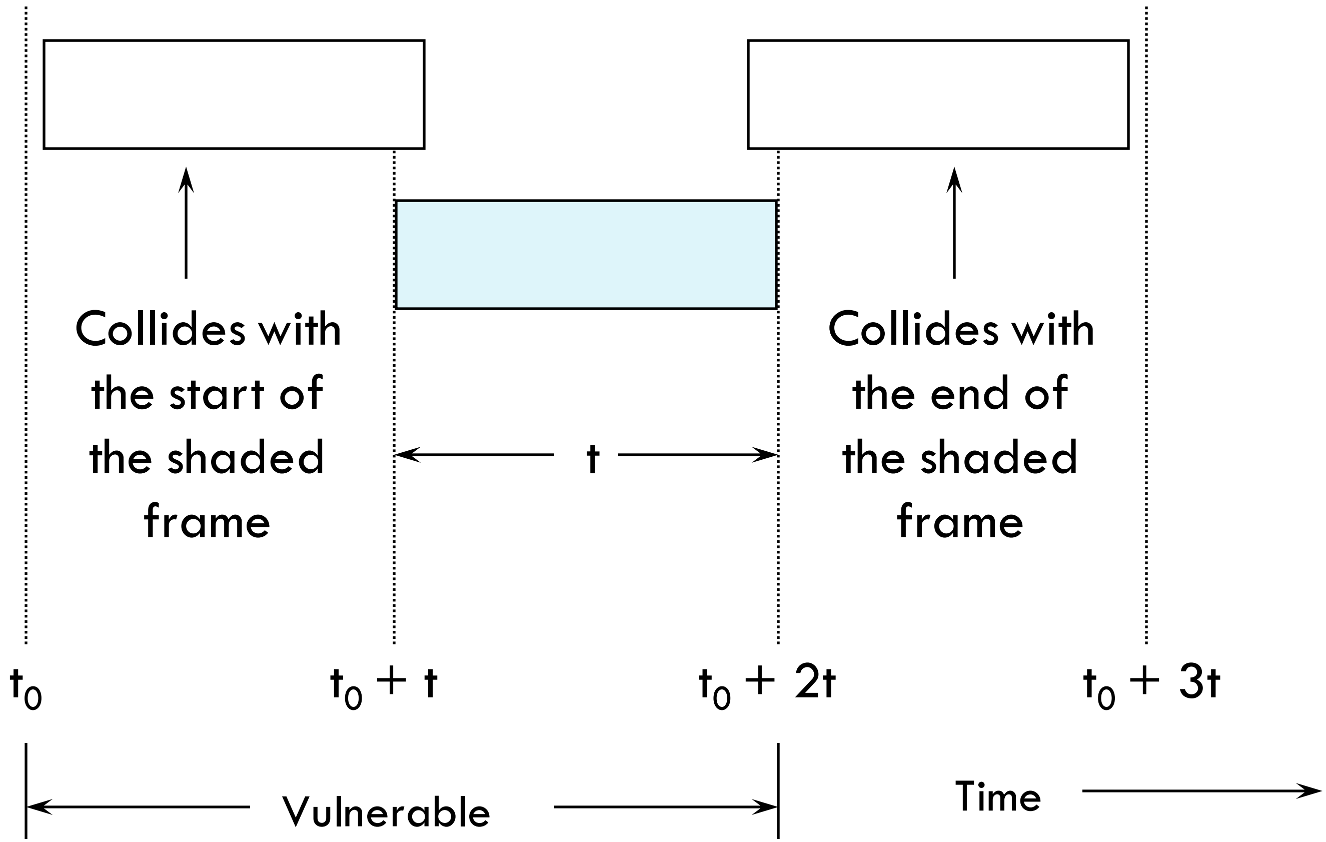
Analysis of Pure ALOHA...

- Since S represents the number of "good" transmissions per *frame time*, and G represents the total number of attempted transmissions per *frame time*, then we have:

$$S = G \times (\text{Probability of good transmission})$$

- The vulnerable time for a successful transmission is $2T_f$
- So, the probability of good transmission is not to have an "arrival" during the vulnerable time .

Analysis of Pure ALOHA…



Vulnerable period for the shaded frame

Analysis of Pure ALOHA...

Using:

$$P_k(t) = \frac{(\lambda t)^k e^{-\lambda t}}{k!}$$

And setting $t = 2T_f$ and $k = 0$, we get

$$P_0(2T_f) = \frac{(\lambda \cdot 2T_f)^0 e^{-\lambda 2T_f}}{0!} = e^{-2G}$$

becasue $\lambda = \dfrac{G}{T_f}$. Thus, $\boxed{S = G \cdot e^{-2G}}$

Analysis of Pure ALOHA…

- If we differentiate $S = Ge^{-2G}$ with respect to G and set the result to 0 and solve for G, we find that the maximum occurs when

$$G = 0.5,$$

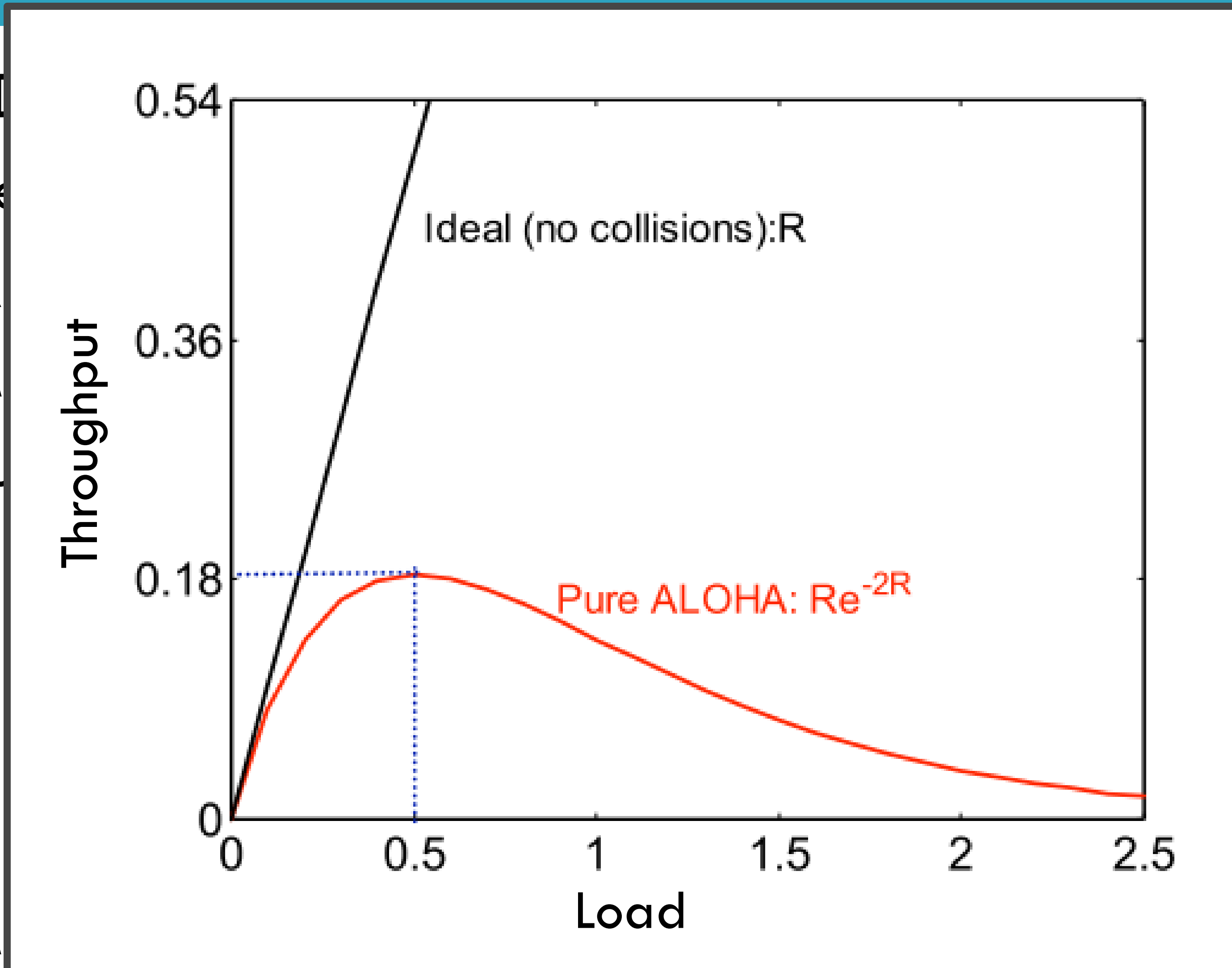
and for that $S = 1/2e = 0.18$. So, the maximum throughput is only 18% of capacity.

Tradeoffs vs. TDMA

- In TD
  - De
- In A
  - M
  - Bu

Send

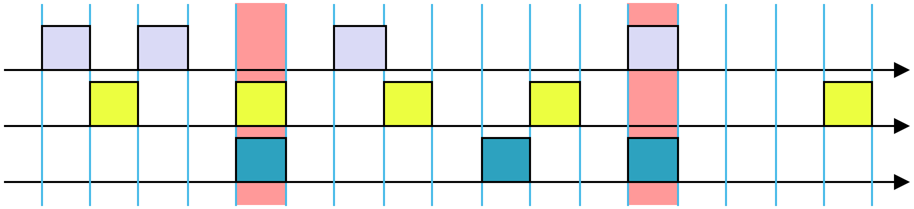
Send



  - M

Slotted ALOHA

- Channel is organized into uniform slots whose size equals the frame transmission time.
- Transmission is permitted only to begin at a slot boundary.



- Here is the procedure:
  - While there is a new frame A to send do

    Send frame A at (the next) slot boundary

Analysis of Slotted ALOHA

- Note that the vulnerable period is now reduced in half. Using:

$$P_k(t) = \frac{(\lambda t)^k e^{-\lambda t}}{k!}$$

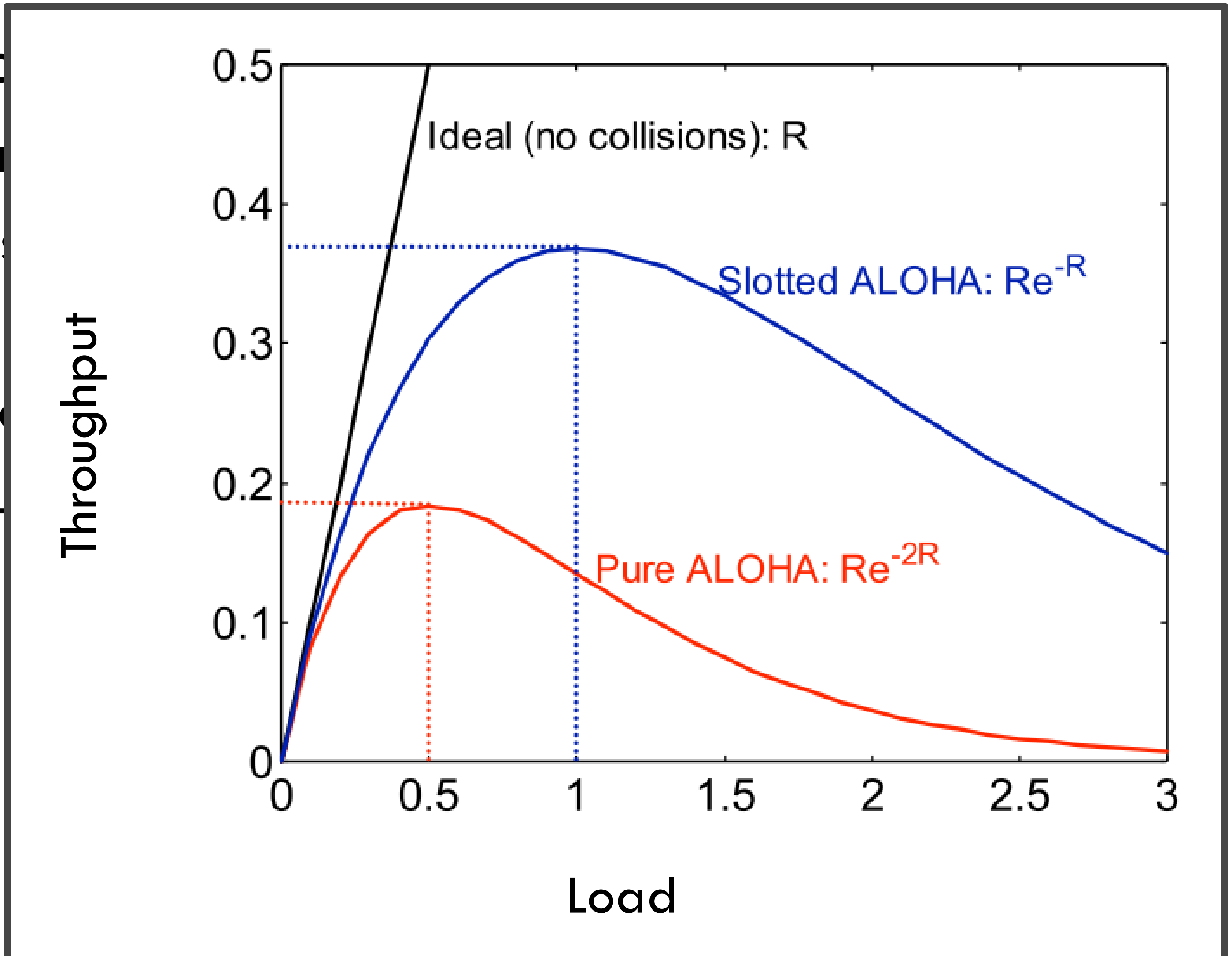
And setting $t = T_f$ and $k = 0$, we get

$$P_0(T_f) = \frac{(\lambda \cdot T_f)^0 e^{-\lambda T_f}}{0!} = e^{-G}$$

$$\text{because} \quad \lambda = \frac{G}{T_f}. \qquad \text{Thus, } \boxed{S = G \cdot e^{-G}}$$

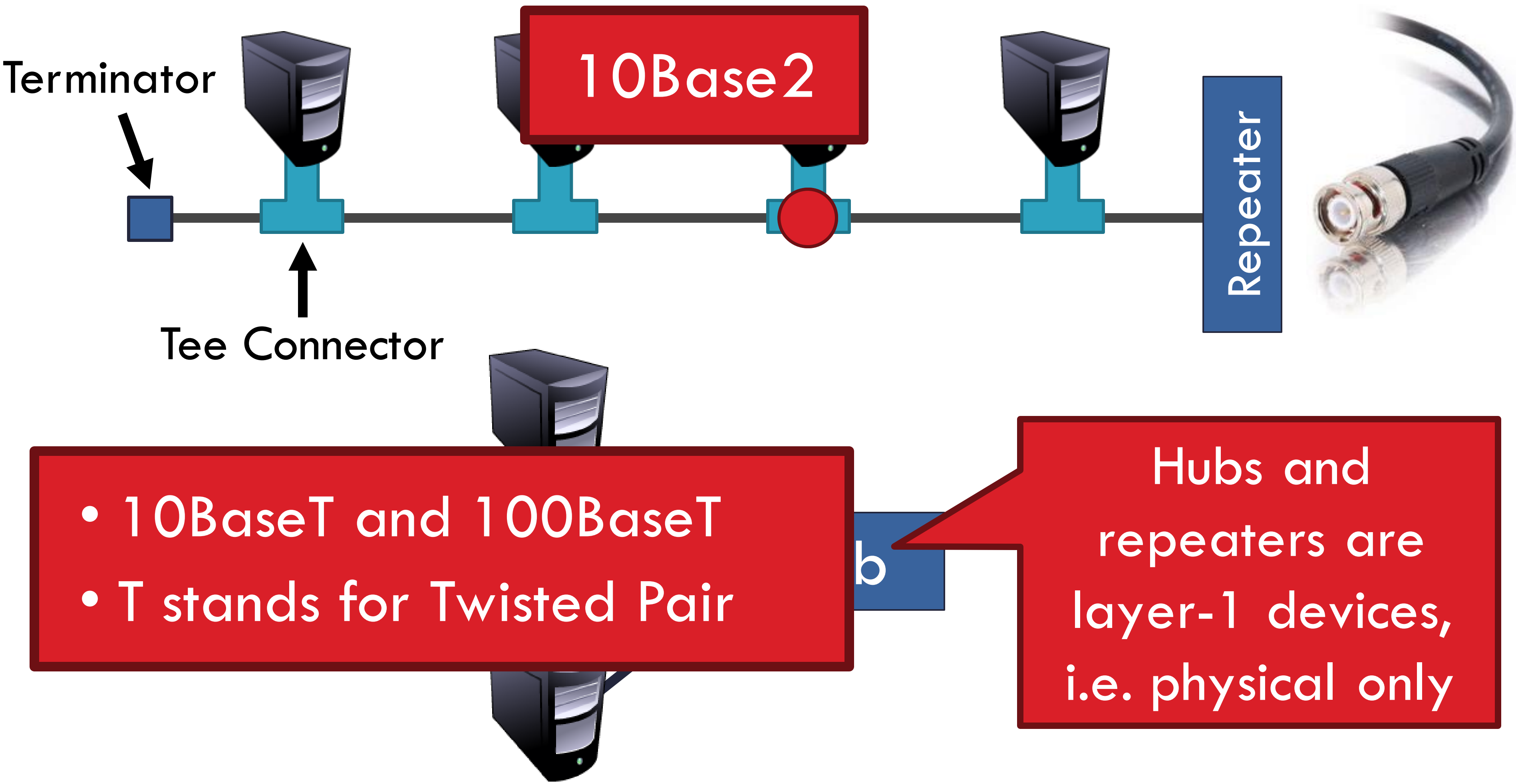
Slotted ALOHA

- Proto
  - Sa
  - Ho
- Thus,
  - 37
  - But



Broadcast Ethernet

- Originally, Ethernet was a broadcast technology



Carrier Sense Multiple Access (CSMA)

- Additional assumption:
  - Each station is capable of sensing the medium to determine if another transmission is underway

Non-persistent CSMA

While there is a new frame A to send

1. Check the medium
2. If the medium is busy, wait some time, and go to 1.
3. (medium idle) Send frame A

1-persistent CSMA

While there is a new frame A to send

1. Check the medium
2. If the medium is busy, go to 1.
3. (medium idle) Send frame A

p-persistent CSMA

While there is a new frame A to send

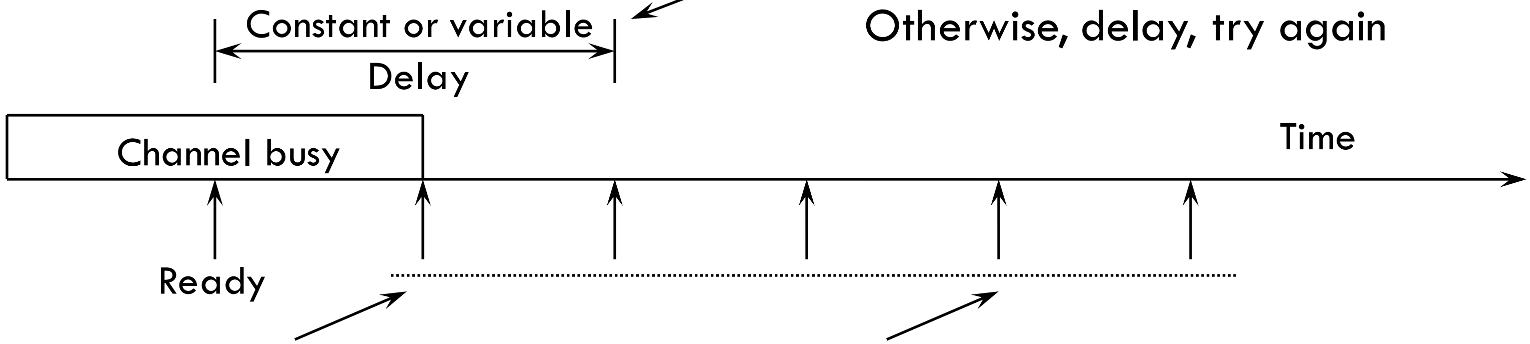
1. Check the medium
2. If the medium is busy, go to 1.
3. (medium idle) With probability p send frame A, and probability (1- p) delay one time slot and go to 1.

CSMA Summary

- Nonpersistent
- 1-persistent
- *p*-persistent

<u>CSMA persistence and backoff</u>

***Non-persistent*:**
Transmit if idle
Otherwise, delay, try again

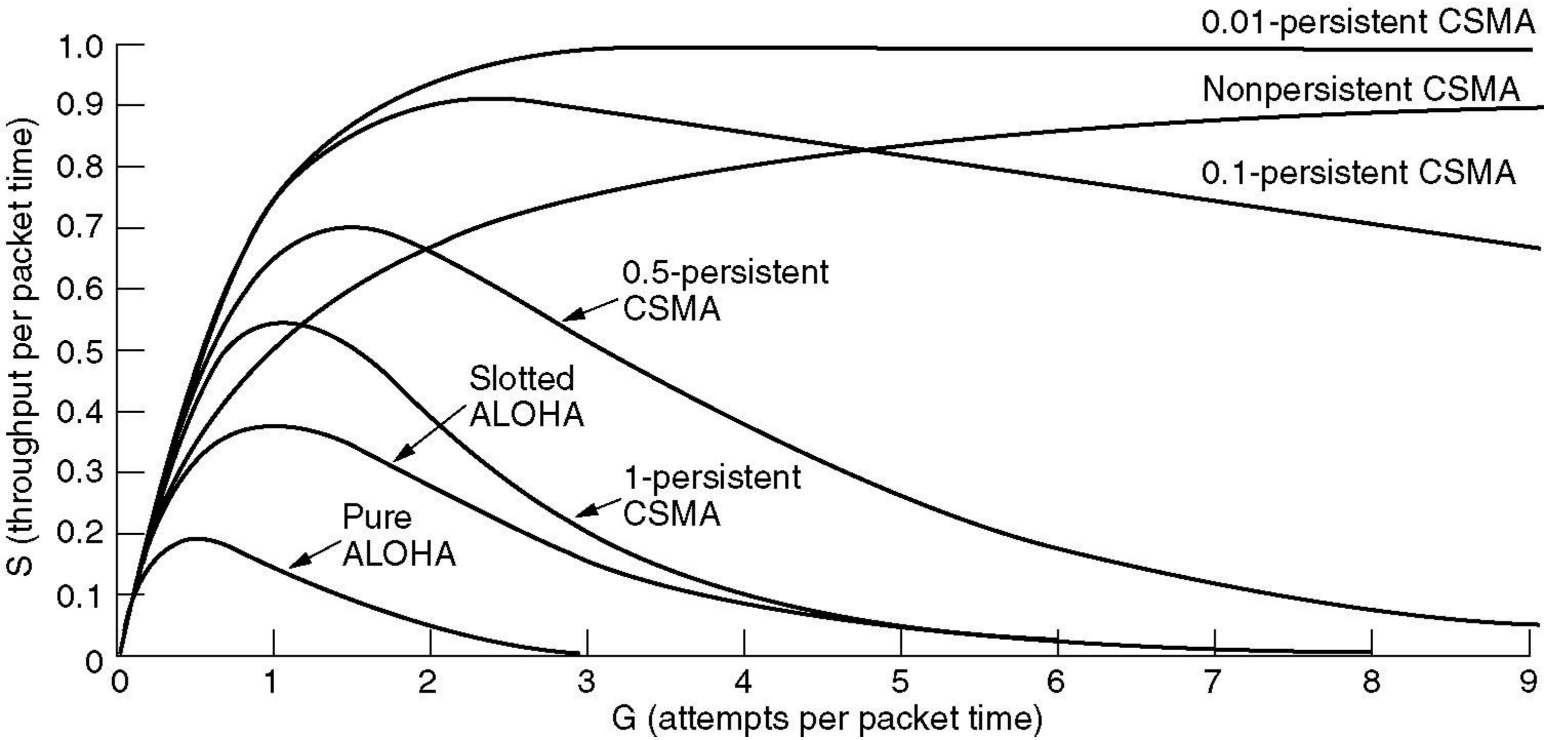


***1-persistent*:**
Transmit as soon as channel goes idle
If collision, back off and try again

p-persistent:
Transmit as soon as channel goes idle with probability *p*
Otherwise, delay one slot, repeat process

Persistent and Non-persistent CSMA

Comparison of throughput versus load for various random access protocols.



CSMA with Collision Detection

- Stations can sense the medium while transmitting
- A station aborts its transmission if it senses another transmission is also happening (that is, it detects collision)
- Question: When can a station be sure that it has *seized* the channel?
  - Minimum time to detect collision is the time it takes for a signal to traverse between two farthest apart stations.

CSMA/CD

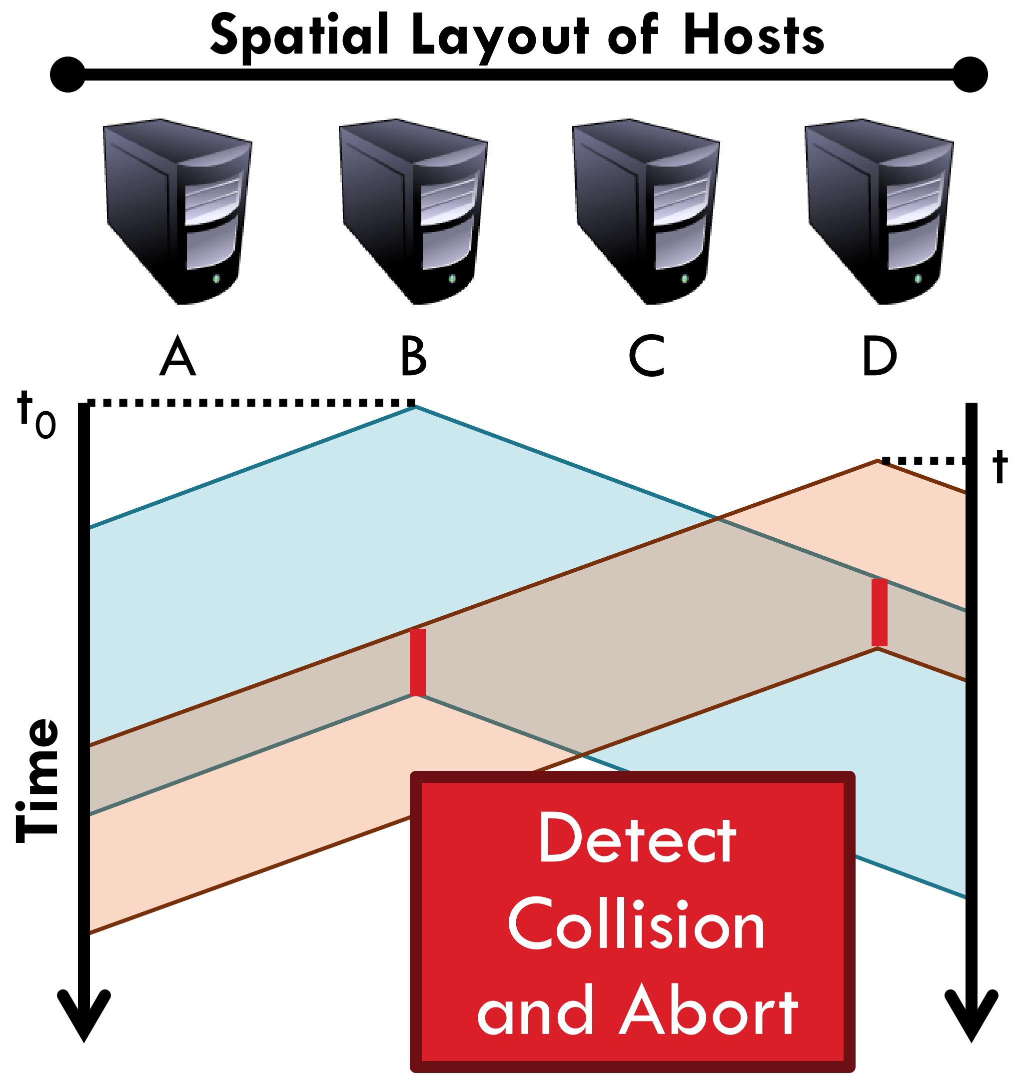
- A station is said to *seize* the channel if all the other stations become aware of its transmission.
- There has to be a lower bound on the length of each frame for the *collision detection* feature to work out.
- Ethernet uses CSMA/CD

CSMA/CD

- Carrier sense multiple access with collision detection
- Key insight: wired protocol allows us to sense the medium
- Algorithm
  1. Sense for carrier
  2. If carrier is present, wait for it to end
     - Sending would cause a collision and waste time
  3. Send a frame and sense for collision
  4. If no collision, then frame has been delivered
  5. If collision, abort immediately
     - Why keep sending if the frame is already corrupted?
  6. Perform exponential backoff then retransmit

CSMA/CD Collisions

- Collisions can occur
- Collisions are quickly detected and aborted
- Note the role of distance, propagation delay, and frame length



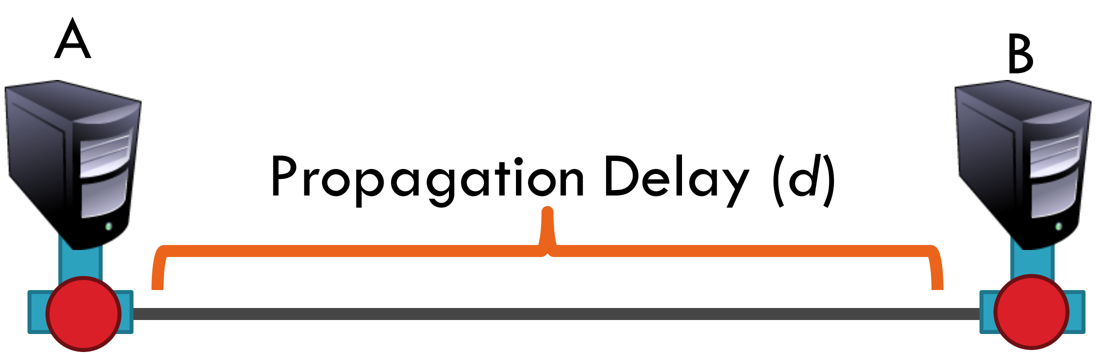
Exponential Backoff

- When a sender detects a collision, send "jam signal"
  - Make sure all hosts are aware of collision
  - Jam signal is 32 bits long (plus header overhead)
- Exponential backoff operates:
  - Select $k \in [0, 2^n - 1]$ unif. rnd., where n = number of collisions
  - Wait k time units (packet times) before retransmission
  - n is capped at 10, frame dropped after 16 collisions
- Backoff time is divided into contention slots

Minimum Packet Sizes

- Why is the minimum packet size 64 bytes?
  - To give hosts enough time to detect collisions
- What is the relationship between packet size and cable length?

1. Time *t*: Host A starts transmitting
2. Time *t* + *d*: Host B starts transmitting
3. Time *t* + 2**d*: collision detected

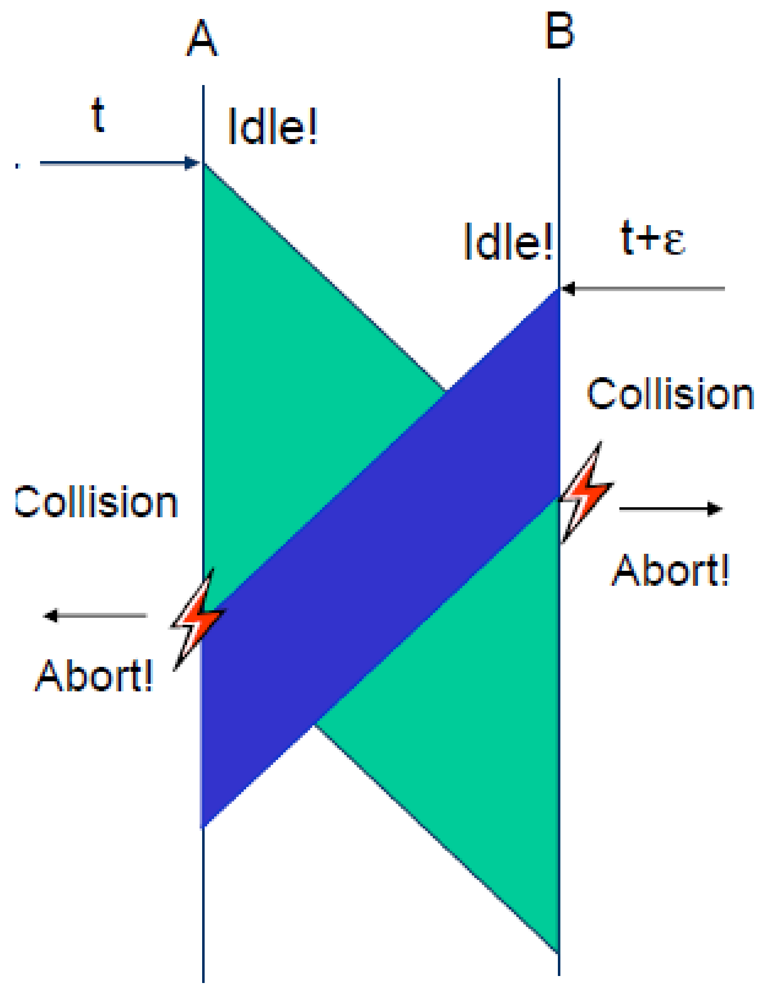


<u>Basic idea: Host A must be transmitting at time 2*d!</u>

CSMA/CD

- CSMA/CD can be in one of three states: contention, transmission, or idle.

- To detect all the collisions we need
  - $T_f \geq 2T_{pg}$
  - where T_f is the time needed to send the frame
  - And T_{pg} is the propagation delay between A and B



Minimum Packet Size

- Host A must be transmitting after 2*d time units
  - Min_pkt = rate (b/s) * 2 * d (s)
    - … but wh
    - Propagati
  - This gives:
  - Min_pkt =
- So cable length is equ
  - Dist = min_pkt * light sp /(2 * rate)

(64B*8)*(2.5*10^8mps)/(2*10^7bps) = 6400 meters

- 10 Mbps Ethernet
- Packet and cable lengths change for faster Ethernet standards

Cable Length Examples

min_frame_size*light_speed/(2*bandwidth) = max_cable_length

$(64B*8)*(2.5*10^8 mps)/(2*10Mbps) = 6400$ meters

- What is the max cable length if min packet size were changed to 1024 bytes?
  - 102.4 kilometers
- What is max cable length if bandwidth were changed to 1 Gbps ?
  - 64 meters
- What if you changed min packet size to 1024 bytes and bandwidth to 1 Gbps?
  - 1024 meters

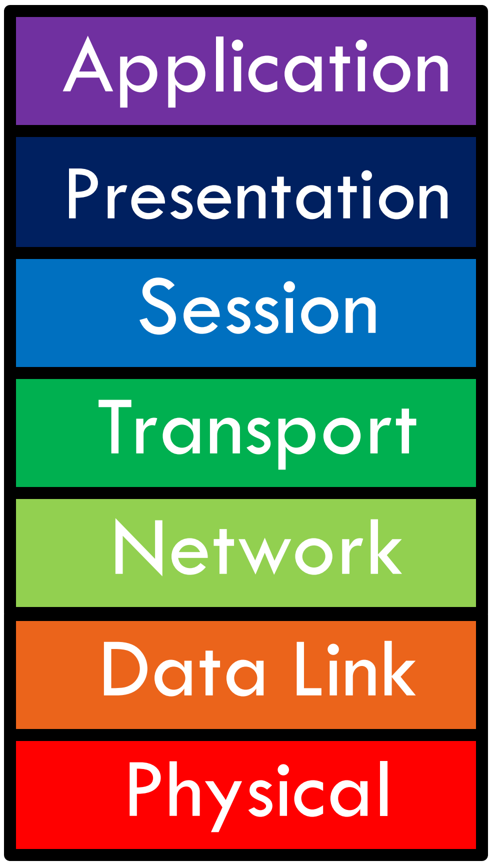
Maximum Packet Size

- Maximum Transmission Unit (MTU): 1500 bytes
- Pros:
  - Bit errors in long packets incur significant recovery penalty
- Cons:
  - More bytes wasted on header information
  - Higher per packet processing overhead
- Datacenters shifting towards Jumbo Frames
  - 9000 bytes per packet

Long Live Ethernet

- Today's Ethernet is switched
  - More on this later
- 1Gbit and 10Gbit Ethernet now common
  - 100Gbit on the way
  - Uses same old packet header
  - Full duplex (send and receive at the same time)
  - Auto negotiating (backwards compatibility)
  - Can also carry power

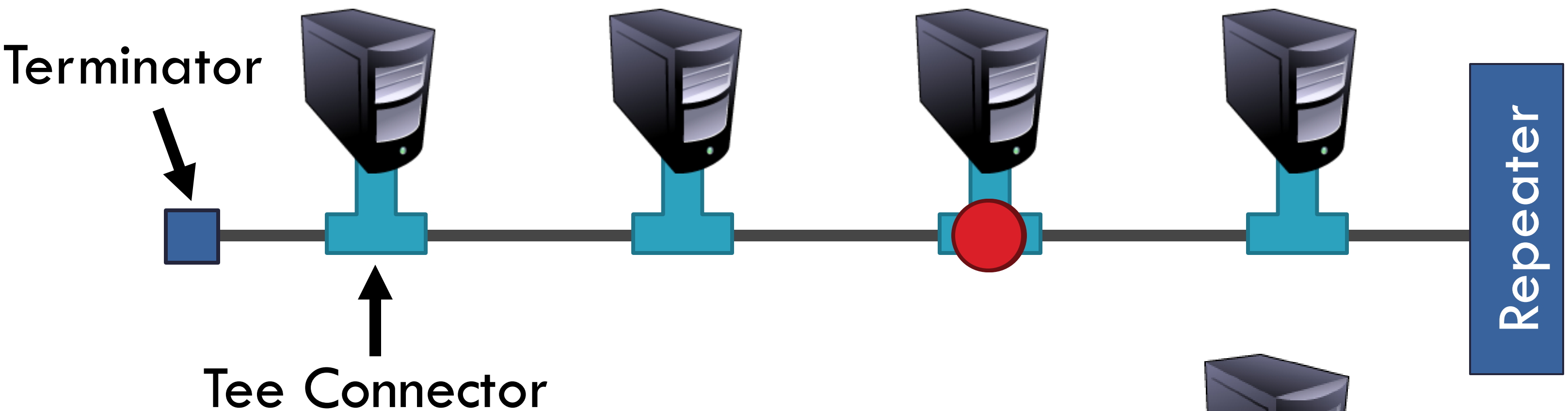
Just Above the Data Link Layer

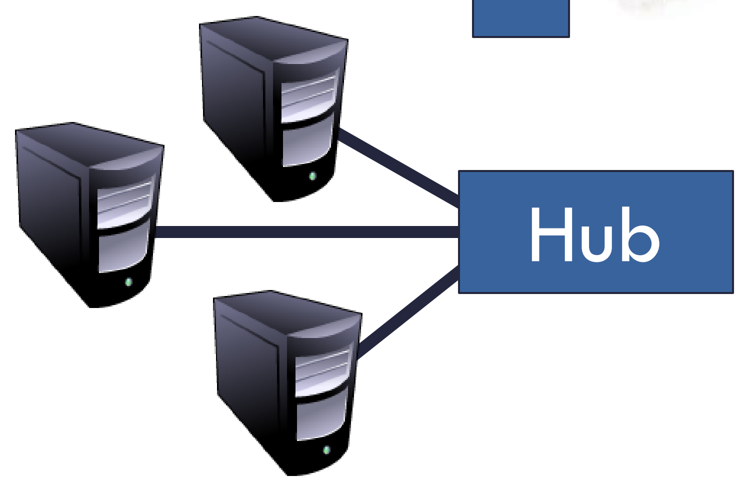


- Bridging
  - How do we connect LANs?
- Function:
  - Route packets between LANs
- Key challenges:
  - Plug-and-play, self configuration
  - How to resolve loops

Recap

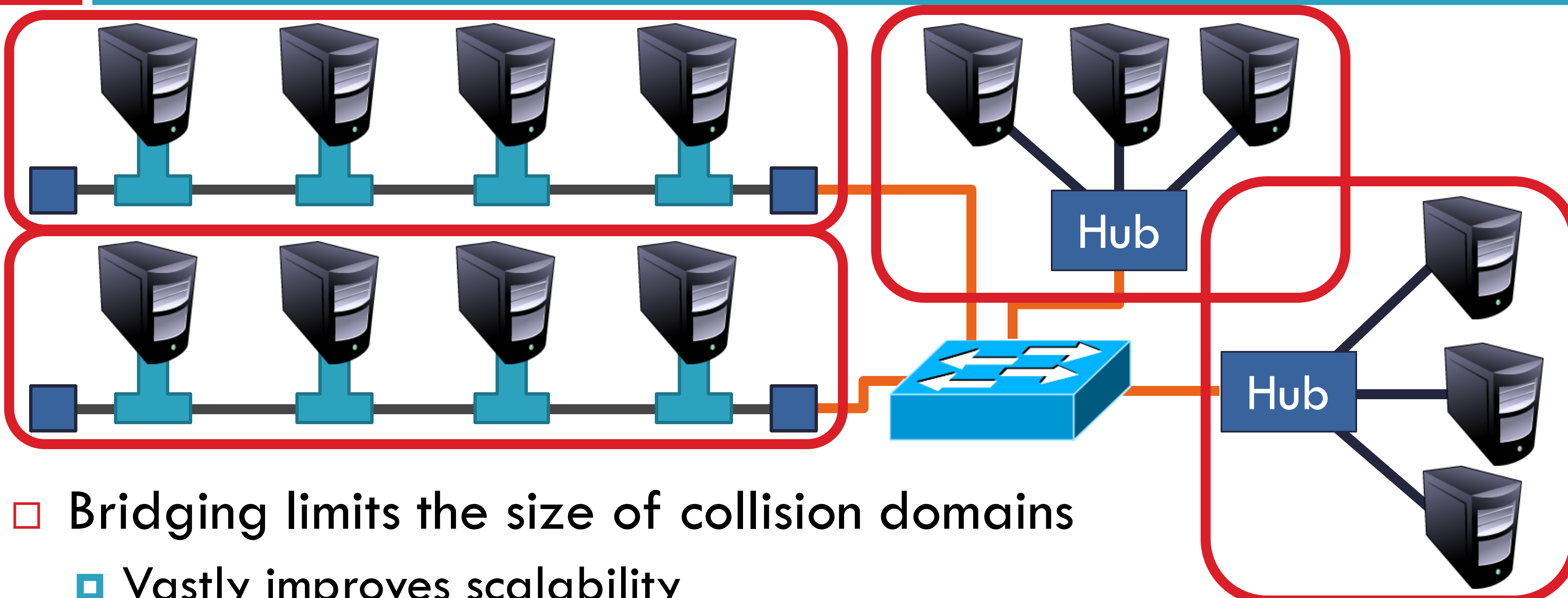
- Originally, Ethernet was a broadcast technology





- Pros: Simplicity
  - Hardware is stupid and cheap
- Cons: No scalability
  - More hosts = more collisions = pandemonium

Bridging the LANs



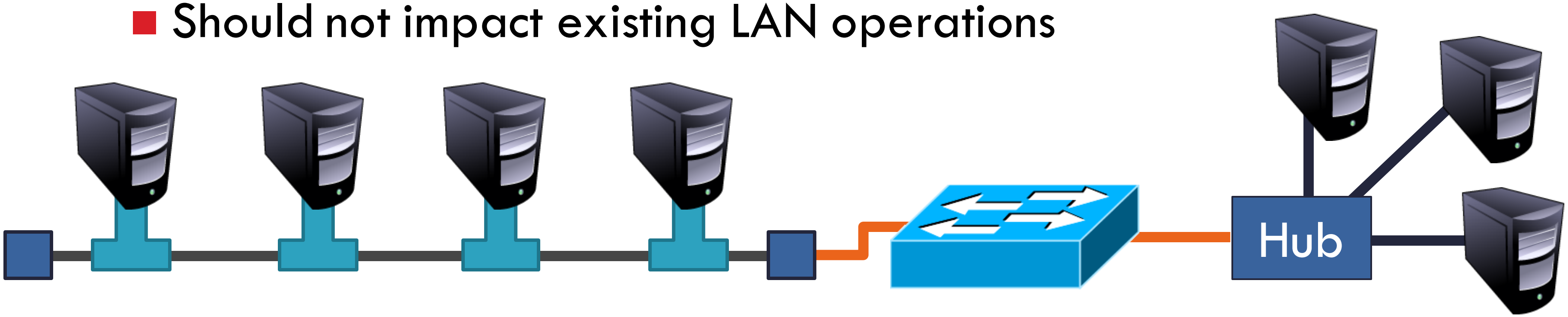
- Bridging limits the size of collision domains
  - Vastly improves scalability
  - Question: could the whole Internet be one bridging domain?
- Tradeoff: bridges are more complex than hubs
  - Physical layer device vs. data link layer device
  - Need memory buffers, packet processing hardware, routing tables

Bridges

- Original form of Ethernet switch

1. Forwarding of frames
2. Learning of (MAC) Addresses
3. Spanning Tree Algorithm (to handle loops)

    - No hardware of software changes on hosts/hubs
    - Should not impact existing LAN operations



Frame Forwarding Tables

- Each bridge maintains a forwarding table

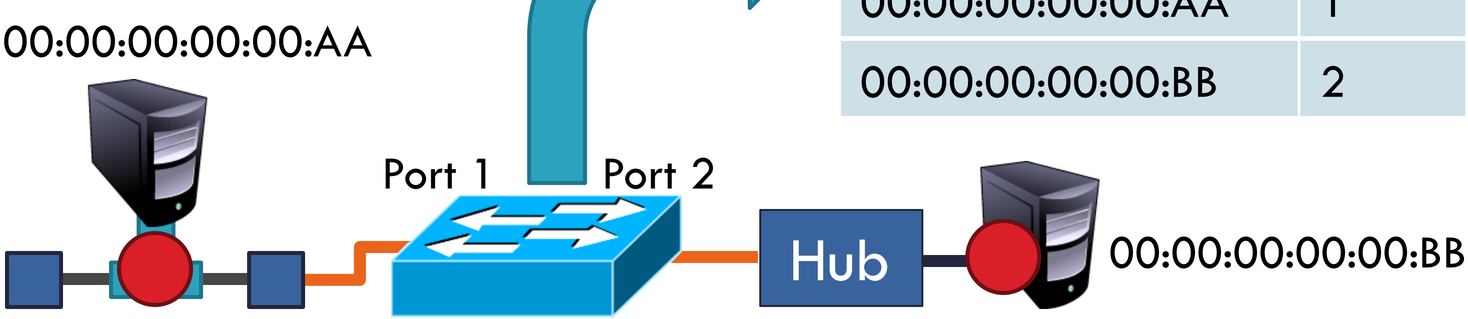
MAC Address	Port	Age
00:00:00:00:00:AA	1	1 minute
00:00:00:00:00:BB	2	7 minutes
00:00:00:00:00:CC	3	2 seconds
00:00:00:00:00:DD	1	3 minutes

Learning Addresses

- Manual configuration is possible, but…
  - Time consuming
  - Error Prone
  - Not adaptable (hosts may get added or removed)
- Instead, learn addresses using a simp
  - Look at the source of frames that arrive on each po

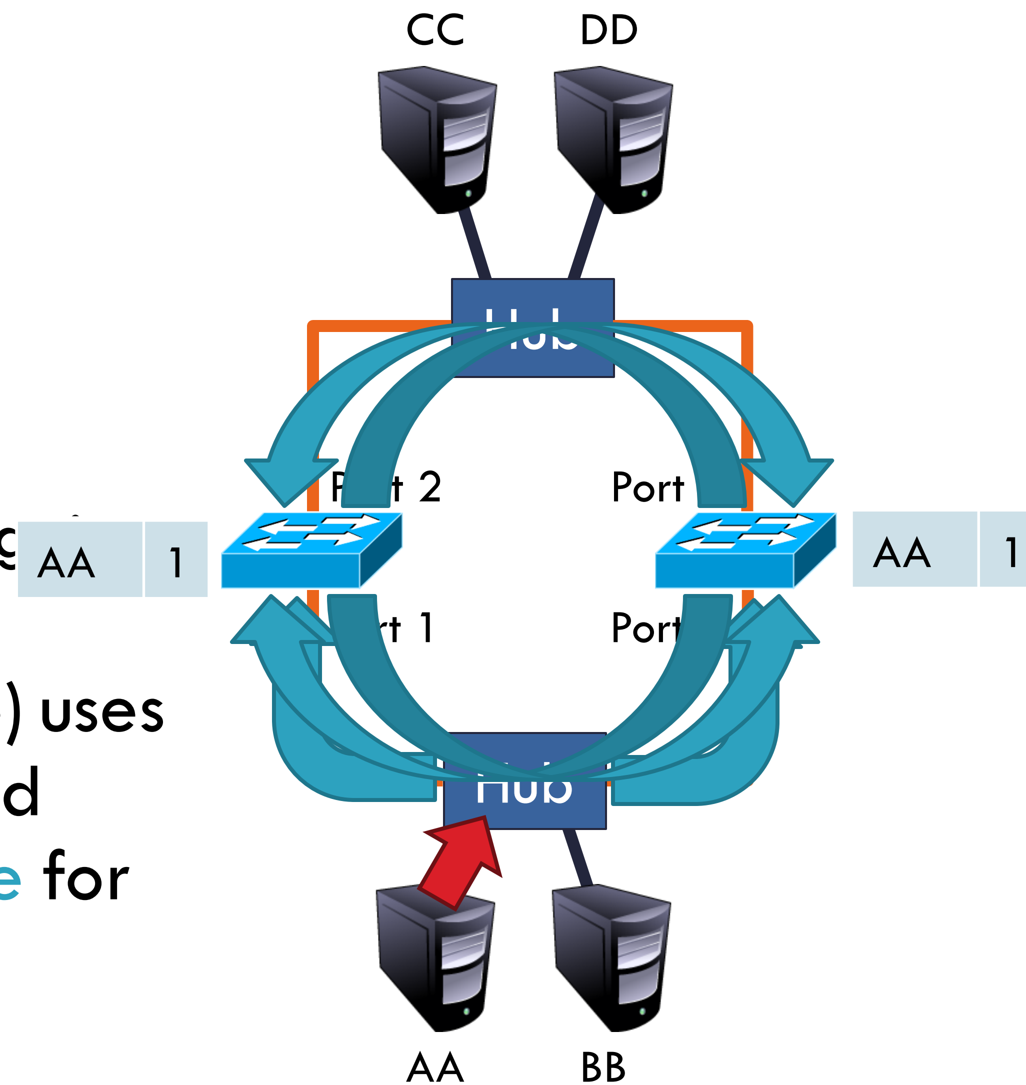
Delete old entries after a timeout

MAC Address	Port	Age
00:00:00:00:00:AA	1	0 minutes
00:00:00:00:00:BB	2	0 minutes



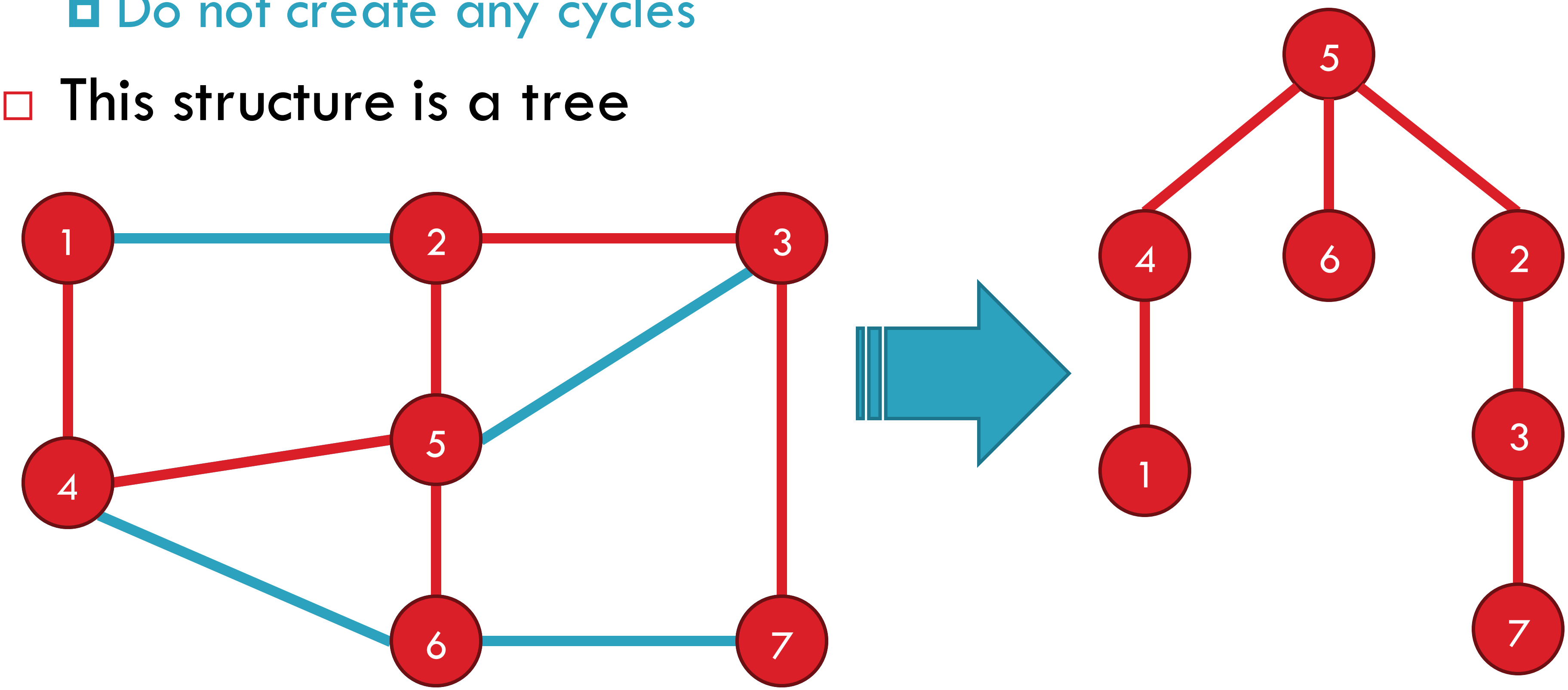
The Danger of Loops

- <Src=AA, Dest=DD>
- This continues to infinity
  - How do we stop this?
- Remove loops from the topology
  - Without physically unplug cables
- 802.1 (LAN architecture) uses an algorithm to build and maintain a spanning tree for routing



Spanning Tree Definition

- A subset of edges in a graph that:
  - Span all nodes
  - Do not create any cycles
- This structure is a tree



802.1 Spanning Tree Approach

1. Elect a bridge to be the root of the tree
2. Every bridge finds shortest path to the root
3. Union of these paths becomes the spanning tree

- Bridges exchange Configuration Bridge Protocol Data Units (BPDUs) to build the tree
  - Used to elect the root bridge
  - Calculate shortest paths
  - Locate the next hop closest to the root, and its port
  - Select ports to be included in the spanning trees

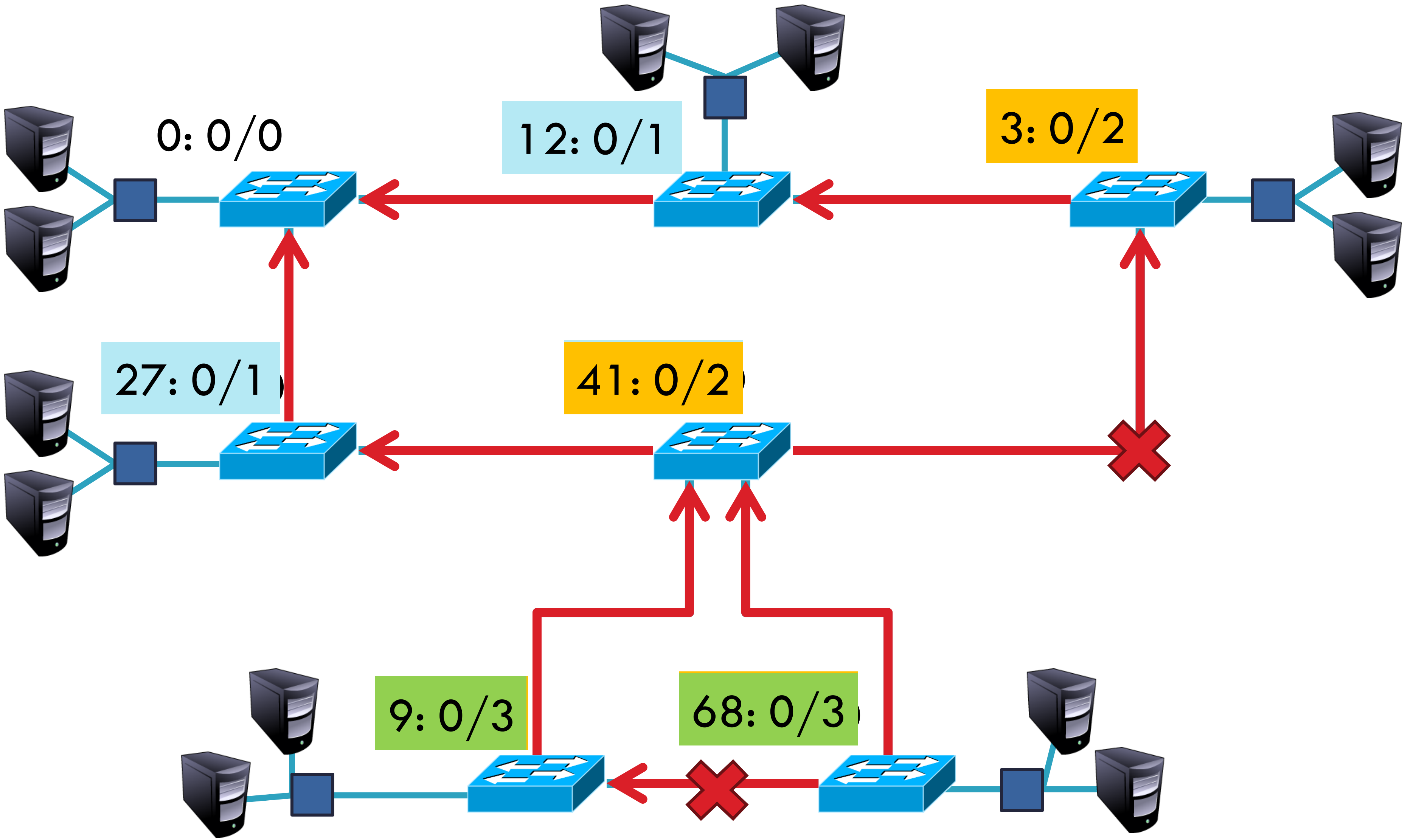
Determining the Root

- Initially, all hosts assume they are the root
- Bridges broadcast BPDUs:

Bridge ID	Root ID	Path Cost to Root

- Based on received BPDUs, each switch chooses:
  - A new root (smallest known Root ID)
  - A new root port (what interface goes towards the root)
  - A new designated bridge (who is the next hop to root)

Spanning Tree Construction



Bridges vs. Switches

- Bridges make it possible to increase LAN capacity
  - Reduces the amount of broadcast packets
  - No loops
- Switch is a special case of a bridge
  - Each port is connected to a single host
    - Either a client machine
    - Or another switch
  - Links are full duplex
  - Simplified hardware: no need for CSMA/CD!
  - Can have different speeds on each port

Switching the Internet

- Capabilities of switches:
  - Network-wide routing based on MAC addresses
  - Learn routes to new hosts automatically
  - Resolve loops
- Could the whole Internet be one switching domain?

NO

Limitations of MAC Routing

- Inefficient
  - Flooding packets to locate unknown hosts
- Poor Performance
  - Spanning tree does not balance load
  - Hot spots
- Extremely Poor Scalability
  - Every switch needs every MAC address on the Internet in its routing table!
- IP addresses these problems (next …)