



Chapter 5: Data Link Layer

Silvia Giordano

ICA, EPFL

5: DataLink Layer 5-1

The **data-link layer** is responsible for transferring a datagram across an individual link. A link is the communication channels that connect two adjacent hosts or routers.

Examples of link-layer protocols include Ethernet, token ring, FDDI, and PPP.

The Data Link Layer

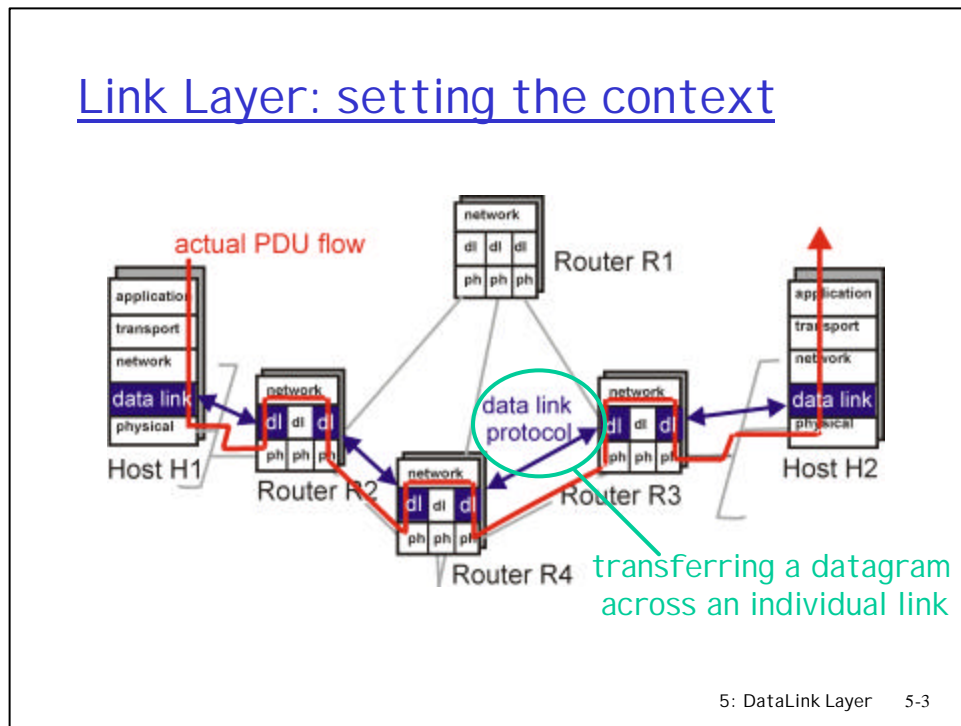
Our goals:

- ❑ 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

Overview:

- ❑ link layer services
- ❑ error detection, correction
- ❑ multiple access protocols and LANs
- ❑ link layer addressing, ARP
- ❑ specific link layer technologies:
 - Ethernet
 - hubs, bridges, switches
 - PPP

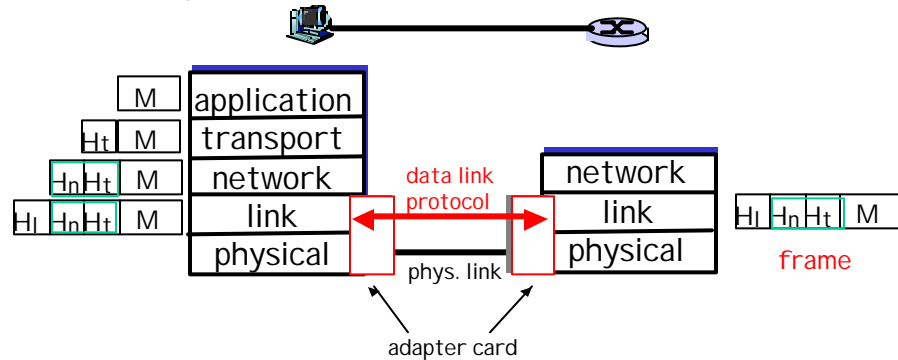
Link Layer: setting the context



Communication networks provide a communication service between two hosts. This communication path starts at the source host, passes through a series of routers, and ends at the destination host. At that layer, where we are not particularly concerned whether a device is a router or a host, hosts and the routers are referred to simply as **nodes**, and to the communication channels that connect adjacent nodes along the communication path as **links**. In order to move a datagram from source host to destination host, the datagram must be moved over each of the *individual links* in the path. The **data-link layer** is responsible for transferring a datagram that comes from the network layer across an individual link.

Link Layer: setting the context

- ❑ two *physically connected* devices:
 - host-router, router-router, host-host
- ❑ unit of data: *frame*
- ❑ a datagram may be handled by different link-layer protocols, offering different services, on the different links in the path.



5: DataLink Layer 5-4

A link is the physical communication channels that connect either two host, two routers or a host-router pair. The **link-layer protocol** defines the format of the units of data (**frames**) exchanged between the nodes at the ends of the link, as well as the actions taken by these nodes when sending and receiving these data units. Each link-layer frame typically encapsulates one network-layer datagram.

A link-layer protocol has the node-to-node job of moving a network-layer datagram over a *single link* in the path. An important characteristic of the link layer is that a datagram may be handled by different link-layer protocols, offering different services, on the different links in the path.

Link Layer Services

- Framing, link access:
 - encapsulate datagram into frame, adding header, trailer
 - implement channel access if shared medium,
 - 'physical addresses' used in frame headers to identify source, dest
 - different from IP address!
- Reliable delivery between two physically connected devices:
 - we learned how to do this already (see chapter3)
 - seldom used on low bit error link (fiber, some twisted pair)
 - wireless links: high error rates
 - link-level reliability to avoid end-end retransmission

5: DataLink Layer 5-5

Possible services that can be offered by a link-layer protocol include:

• *Framing and link access.* Almost all link-layer protocols encapsulate each network-layer datagram within a network-layer datagram is inserted, and a number of header fields. A data-link protocol specifies the structure of the frame, as well as a channel access protocol that specifies the rules by which a frame is transmitted onto the link. For point-to-point links that have a single sender on one end of the link and a single receiver at the other end of the link, the link-access protocol is simple (or non-existent)--the sender can send a frame whenever the link is idle. The more interesting case is when multiple nodes share a single broadcast link--the so-called multiple access problem. Here, the channel access protocol serves to coordinate the frame transmissions of the many nodes link-layer frame before transmission onto the link. A frame consists of a data field, in which the. The frame headers also often include fields for a node's so-called **physical address**, which is completely *distinct* from the node's network layer (for example, IP) address.

• *Reliable delivery.* When a link-layer protocol provides reliable-delivery service, it guarantees to move each network-layer datagram across the link without error. This is achieved with acknowledgments and retransmissions. A link-layer reliable-delivery service is often used for links that are prone to high error rates, such as a wireless link, with the goal of correcting an error locally, on the link where the error occurs, rather than forcing an end-to-end retransmission of the data by a transport- or application-layer protocol. However, link-layer reliable delivery can be considered an unnecessary overhead for low bit-error links, including fiber, coax, and many twisted-pair copper links. For this reason, many of the most popular link-layer protocols do not provide a reliable delivery service.

Link Layer Services (more)

- ❑ **Flow Control:**
 - pacing between sender and receivers
- ❑ **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**

5: DataLink Layer 5-6

• *Flow control.* A link-layer protocol can provide flow control in order to prevent the sending node on one side of a link from overwhelming the receiving node on the other side of the link.

• *Error detection.* Many link-layer protocols provide a mechanism to detect the presence of one or more errors. This is done by having the transmitting node set error-detection bits in the frame, and having the receiving node perform an error check. Error detection is a very common service among link-layer protocols. Error detection in the link layer is usually more sophisticated than the one at the transport layer and network layers and implemented in hardware.

• *Error correction.* Error correction is similar to error detection, except that a receiver cannot only detect whether errors have been introduced in the frame but can also determine exactly where in the frame the errors have occurred (and hence correct these errors).

• *Half-duplex and full-duplex.* With full-duplex transmission, the nodes at both ends of a link may transmit packets at the same time. With half-duplex transmission, a node cannot both transmit and receive at the same time.

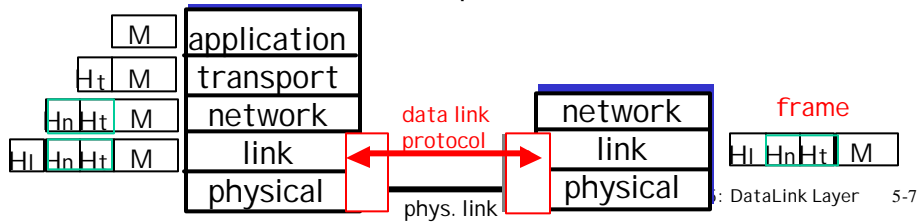
Last week

□ Link Layer

- transferring a datagram across an individual link
- two *physically connected* devices:
 - host-router, router-router, host-host
- unit of data: *frame*
- a pkt may be handled by different link-layer protocols, offering different services, on the different links in the path.


□ Link Layer Services

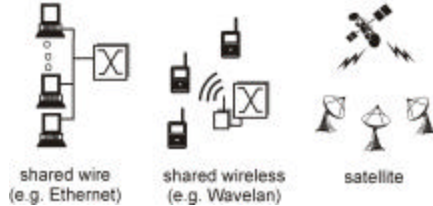
- Framing, link access
- Reliable delivery between two physically connected devices
- Flow Control
- Error Detection
- Error Correction
- Half-duplex and full-duplex



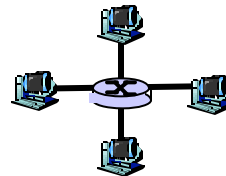
Multiple Access Links and Protocols

Three types of "links":

- point-to-point (single wire, e.g. PPP, SLIP) 
- **broadcast** (shared wire or medium; e.g. Ethernet, Wavelan, etc.)



- switched (e.g., switched Ethernet, ATM etc)



5: DataLink Layer 5-8

A **point-to-point link** consists of a single sender on one end of the link, and a single receiver at the other end of the link. Many link-layer protocols have been designed for point-to-point links; PPP and HDLC for example. The second type of link, a **broadcast link**, can have multiple sending and receiving nodes all connected to the same, single, shared broadcast channel, where each node on the channel receives a copy of any sent frame, e.g. Ethernet. In shared broadcast channel there is the problem of how to coordinate the access of multiple sending and receiving nodes to a --the so-called **multiple access problem**. Broadcast channels are often used in **local area networks (LANs)**, networks that are geographically concentrated in a single building (or on a corporate or university campus). A **switched** link is a direct link that allows multiple and simultaneous dedicated access to the LAN, thus it allows for simultaneous transmission.

Multiple Access protocols

- ❑ single shared communication channel
- ❑ two or more simultaneous transmissions by nodes: interference
 - only one node can send **successfully** at a time
- ❑ **multiple access protocol:**
 - distributed algorithm that determines how stations share channel, i.e., determine when station 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 (e.g., to channel errors)
 - performance

5: DataLink Layer 5-9

In presence of a shared medium, it can happen that some nodes transmit at the same time and that frames collide or interfere. It is therefore necessary to find a protocol for sharing a broadcast medium. **Multiple access protocols** regulate nodes transmission onto the shared broadcast channel. Moreover, also the communication due to the coordination of the transmission must use the channel itself.

Multiple Access Protocols

Three broad classes:

- ❑ **Channel Partitioning**
 - divide channel into smaller “pieces” (time slots, frequency)
 - allocate piece to node for exclusive use
- ❑ **Random Access**
 - allow collisions
 - “recover” from collisions
- ❑ **“Taking turns”**
 - tightly coordinate shared access to avoid collisions

Goal: efficient, fair, simple, decentralized

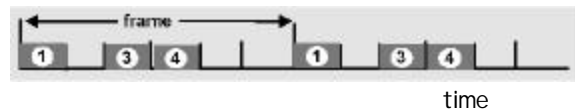
5: DataLink Layer 5-10

Multiple access protocols can be classified as belonging to one of three categories: **channel partitioning protocols, random access protocols, and taking-turns protocols.**

Channel Partitioning 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



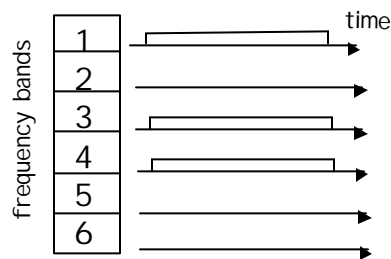
5: DataLink Layer 5-11

For a TDM link, time is divided into **time frames** of fixed duration, and each frame is divided into a fixed number of time **slots**. When the network establishes a connection across a link, the network dedicates one time slot in every frame to the connection. These slots are dedicated for the sole use of that connection, with a time slot available for use (in every frame) to transmit the connection's data. Whenever a node has a frame to send, it transmits the frame's bits during its assigned time slot in the revolving TDM frame. Typically, frame sizes are chosen so that a single frame can be transmitting during a slot time. The figure shows a simple six-node TDM example where 3 nodes have some traffic to transmit (stations 1,3,4) and use their slots, while the others (stations 2,5,6) have no traffic and their slots remain unused. Once everyone has had its chance to talk, the pattern repeats. TDM is appealing as it eliminates collisions and is perfectly fair: each node gets a dedicated transmission rate of R/N bps during each frame time. However, it has two major drawbacks. First, a node is limited to an average rate of R/N bps even when it is the only node with frames to send. A second drawback is that a node must always wait for its turn in the transmission sequence--again, even when it is the only node with a frame to send.

Channel Partitioning 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



5: DataLink Layer 5-12

FDM divides the R bps channel into different frequencies (each with a bandwidth of R/N) and assigns each frequency to one of the N nodes. FDM thus creates N smaller channels of R/N bps out of the single, larger R bps channel. FDM shares both the advantages and drawbacks of TDM. It avoids collisions and divides the bandwidth fairly among the N nodes. The figure shows the same six-node example where 3 nodes have some traffic to transmit (stations 1,3,4) and use their frequency, while the others (stations 2,5,6) have no traffic and their frequency remain unused. However, even with FDM a node is limited to a bandwidth of R/N , even when it is the only node with frames to send.

Channel Partitioning Protocols: CDMA

CDMA = Code Division Multiple Access

- ❑ used mostly in wireless broadcast channels (cellular, satellite, etc)
- ❑ unique "code" assigned to each user; ie, code set partitioning
- ❑ all users share same frequency, but each user has own "chipping" sequence (ie, code) to encode data
- ❑ *encoded signal* = (original data) X (chipping sequence)
- ❑ *decoding*: inner-product of encoded signal and chipping sequence
- ❑ allows multiple users to "coexist" and transmit simultaneously with minimal interference (if codes are "orthogonal")

5: DataLink Layer 5-13

Code division multiple access (CDMA) assigns a different *code* to each node. Each node then uses its unique code to encode the data bits it sends. CDMA allows different nodes to transmit *simultaneously* and yet have their respective receivers correctly receive a sender's encoded data bits (assuming the receiver knows the sender's code) in spite of interfering transmissions by other nodes. In a CDMA protocol, each bit being sent by the sender is encoded by multiplying the bit by a signal (the code) that changes at a much faster rate (known as the **chipping rate**) than the original sequence of data bits. CDMA works under the assumption that the interfering transmitted bit signals are additive, i.e. coded with orthogonal code. Therefore, each receiver can recover the data sent by a given sender out of the aggregate signal simply by using the sender's code.

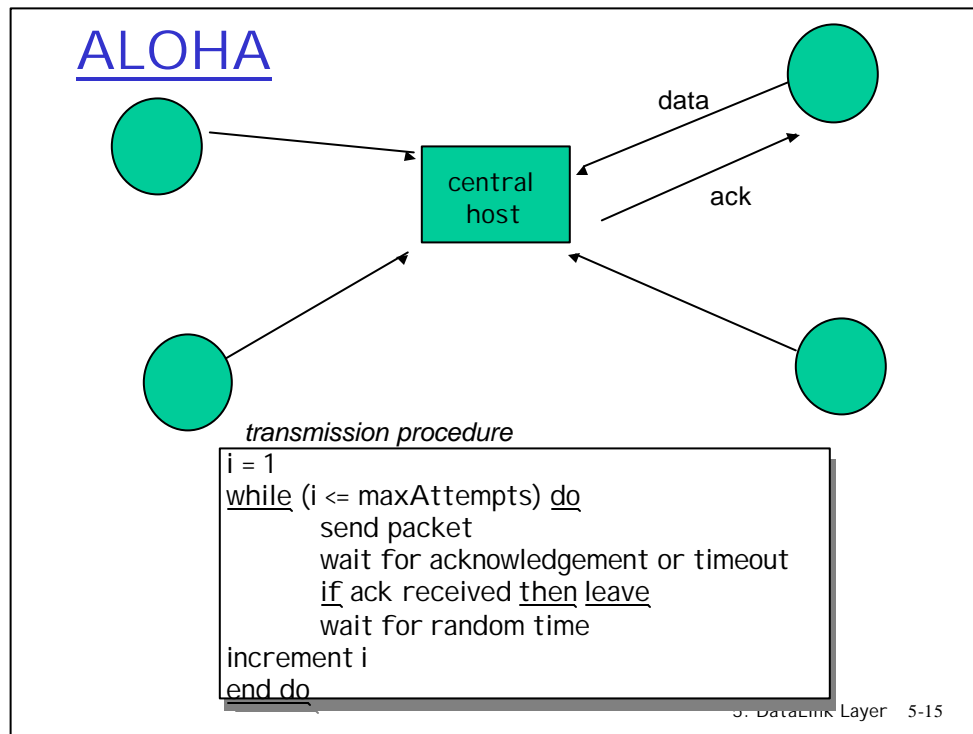
However, in order for the CDMA receivers to be able to extract out a particular sender's signal, the CDMA codes must be carefully chosen and assigned. Secondly, our discussion has assumed that the received signal strengths from various senders at a receiver are the same

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 protocol** specifies:
 - how to detect collisions
 - how to recover from collisions (e.g., via delayed retransmissions)
- ❑ Examples of random access protocols:
 - ALOHA
 - slotted ALOHA
 - CSMA and CSMA/CD

5: DataLink Layer 5-14

In a random access protocol, a transmitting node always transmits at the full rate of the channel, namely, R bps. When there is a collision, each node involved in the collision repeatedly retransmits its frame until the frame gets through without a collision. But when a node experiences a collision, it doesn't necessarily retransmit the frame right away. *Instead it waits a random delay before retransmitting the frame.* Each node involved in a collision chooses independent random delays. Because after a collision the random delays are independently chosen, it is possible that one of the nodes will pick a delay that is sufficiently less than the delays of the other colliding nodes and will therefore be able to sneak its frame into the channel without a collision.



ALOHA is the basis of all non-deterministic access methods. The ALOHA protocol was originally developed for communications between islands (University of Hawaiï) that use radio channels at low bit rates. The ALOHA protocol requires acknowledgements and timers.

In this scheme a station wishing to transmit, does so at will. As a result, two or more frames may overlap in time, causing a collision. Collisions occur, and if a packet is lost, then sources have to retransmit; but they must stagger their attempts randomly, following some collision resolution algorithm, to avoid colliding again.

There is no feedback to the source in case of collision (was too complex to implement at that time). The picture shows a radio transmission scenario; Aloha can also be used on a cable (bus). It is used today in cases where simplicity is more important than performance (for example: ATM metasignalling).

The maximum utilization can be proven to be 18% (see below). This is assuming an ideal retransmission policy that avoids unnecessary repetitions of collisions.

Maximum Utilization of Aloha

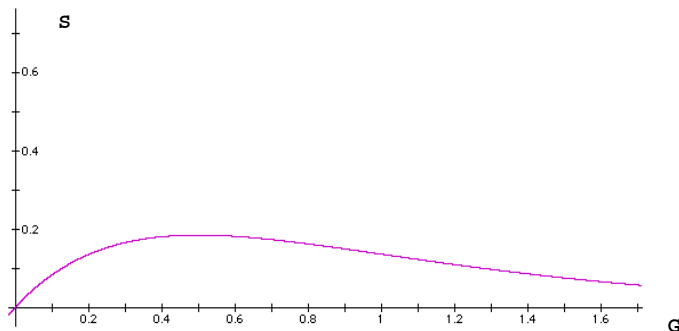
- arrival rate (poisson) μ
- transmission time T
- packet arrive at t
- packet is transmitted without collision iff no other packet arrives during time interval $[t-T, t+T]$ and this happen with probability $\exp(-2\mu T)$.
- over long time s the arrival is μs
- the max utilization is
$$S = \text{max transmission} / \text{time (normalized)} =$$
$$= \mu s \exp(-2\mu T) T / s = \mu T \exp(-2\mu T)$$

5: DataLink Layer 5-16

Maximum Utilization of Aloha

- Max utilization is

$$S = \mu s \exp(-2\mu T) T / s = \mu T \exp(-2\mu T)$$
- $\mu T = G$ is the normalized total transmission attempt rate
- S is maximum equal to $1/2e$ for $G=0.5$



5: DataLink Layer 5-17

The maximum utilization is difficult to obtain and depends on a large number of parameters. We provide an upper bound.

We observe packet arrivals at one point on the medium. We assume that packet arrivals (fresh + retransmissions) are Poisson, and call μ the parameter. This assumption is not obvious. It has been shown to be valid if fresh traffic is Poisson, and if the retransmission policy is optimal. See also in the exercises for an experimental verification. Other retransmission policies lead to worse utilizations, or even to unstable systems (see below)

We assume that packet transmission time is constant, equal to T .

Consider a packet arriving at time t . The packet will be transmitted without collision iff no other packet arrives during time interval $[t-T, t+T]$. The probability of this to happen is $\exp(-2\mu T)$.

Over a long time interval s , the total number of packet arrivals is close to μs , the fraction of packets transmitted without collision is close to $\mu s \exp(-2\mu T)$, therefore the maximum utilization is :

$$\mu s \exp(-2\mu T) T / s = \mu T \exp(-2\mu T)$$

μ is unknown and depends on the retransmission policy. However we can compute the maximum value of the utilization over all possible values of μ . The function is maximum for $2\mu T = 1$, and the value of the maximum is $1/2e =$

ca.

0.18.

Slotted Aloha efficiency

Suppose N stations have packets to send

- each transmits in slot with probability p
- prob. successful transmission S is:

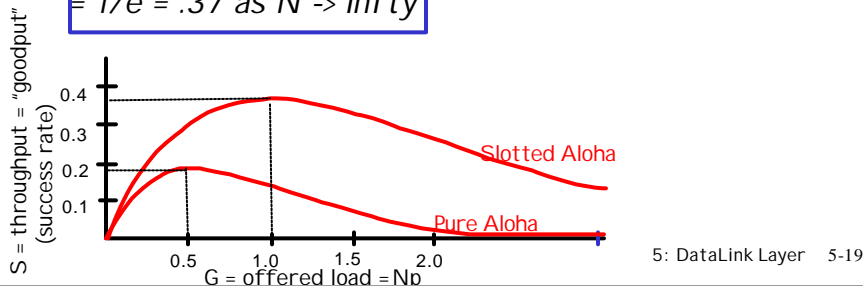
by single node: $S = p(1-p)^{(N-1)}$

by any of N nodes

$S = \text{Prob (only one transmits)} = N p (1-p)^{(N-1)}$

... choosing optimum p as $N \rightarrow \infty$...

$= 1/e = .37$ as $N \rightarrow \infty$



The **efficiency** of a slotted multiple access protocol is defined to be the long-run fraction of successful slots in the case when there are a large number of active nodes, each always having a large number of frames to send.

The probability that a given node transmits with success is $p(1-p)^{N-1}$. Because there are N nodes, the probability that an arbitrary node has a success is $Np(1-p)^{N-1}$, that is the efficiency of slotted ALOHA. To obtain the *maximum* efficiency for N active nodes, we have to find the p^* that maximizes this expression. With p^* the maximum efficiency of the protocol is given by $1/e = 0.37$.

In both slotted and pure ALOHA, a node's decision to transmit is made independently of the activity of the other nodes attached to the broadcast channel. In particular, a node neither pays attention to whether another node happens to be transmitting when it begins to transmit, nor stops transmitting if another node begins to interfere with its transmission.

CSMA: Carrier Sense Multiple Access)

CSMA: listen before transmit:

- If channel sensed idle: transmit entire pkt
- If channel sensed busy, defer transmission
 - **Persistent CSMA:** retry immediately with probability p when channel becomes idle (may cause instability)
 - **Non-persistent CSMA:** retry after random interval

```
i = 1
while (i <= maxAttempts) do
    listen until channel idle
    transmit immediately
    wait for acknowledgement or timeout
    if ack received then leave
    wait random time /* collision*/
    increment i
end do
```

5: DataLink Layer 5-20

CSMA improves on Aloha by requiring that stations listen before transmitting (compare to CB radio). Some collisions can be avoided, but not completely. This is because of propagation delays. Two or more stations may sense that the medium (= the channel) is free and start transmitting at time instants that are close enough for a collision to occur. Assume propagation time between A and B is 2 ms and that all stations are silent until time 0. At time 0, station A starts transmitting for 10 ms, at time 1 ms, station B has not received any signal from A yet, so it can start transmitting. At time 2ms, station B senses the collision but it is too late according to the protocol.

The CSMA protocol requires that stations be able to monitor whether the channel is idle or busy (no requirements to detect collisions). It is a simple improvement to Aloha, at the expense of implementing the monitoring hardware.

The effect of the CSMA protocol can be expressed in the following way. Call T the maximum propagation time from station A to any other stations; if no collision occurs during a time interval of duration T after A started transmitting, then A has seized the channel (no other station can send).

CSMA works well only if the transmission time is much larger than propagation, namely bandwidth-delay product \ll frame size. It has the same stability problems as Aloha. In order to avoid repeated collisions, it is required to wait for a random delay before re-transmitting. If all stations choose the random delays independently, and if the value of the delay has good chances of being larger than T , then there is a high probability that only one of the retransmitting stations seizes the channel.

CSMA/CD (Collision Detection)

CSMA/CD: carrier sensing, deferral as in CSMA

- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage
- persistent or non-persistent retransmission

□ collision detection:

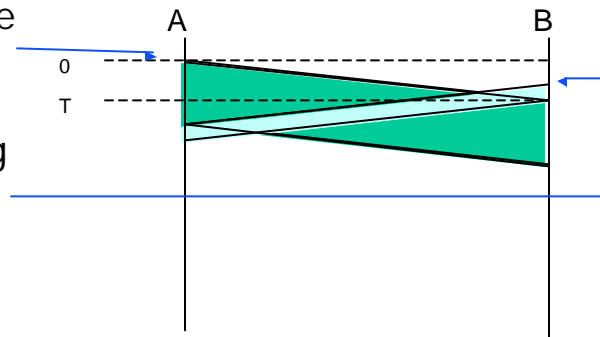
- easy in wired LANs: measure signal strengths, compare transmitted, received signals
- difficult in wireless LANs: receiver shut off while transmitting

```
i = 1
while (i <= maxAttempts) do
    listen until channel is idle
    transmit and listen
    wait until (end of transmission) or (collision detected)
    if collision detected then
        stop transmitting and instead sends a jam signal
    else
        wait for interframe delay
        leave
        wait random time
        increment i
end do
```

CSMA/CD is the protocol used by Ethernet. In addition to CSMA, it requires that a sending station monitors the channel and detects a collision. The benefit is that a collision is detected within a propagation round trip time. These mechanisms give CSMA/CD much better performance than slotted ALOHA in a LAN environment. In fact, if the maximum propagation delay between stations is very small, the efficiency of CSMA/CD can approach 100%. Collisions may still occur.

CSMA / CD Collision

- ❑ A senses idle channel, starts transmitting
- ❑ shortly before T , B senses idle channel, starts transmitting



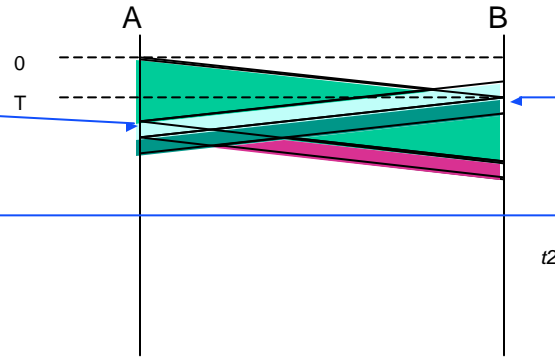
5: DataLink Layer 5-22

If the adapter in A senses that the channel is idle (that is, there is no signal energy from the channel entering the adapter), it starts to transmit the frame. However, due to the transmission time T , the adapter in B can sense that the channel is idle as well, even if A has started the transmission.

In this case there is a collision.

CSMA / CD Jam Signal

- ❑ B senses collision, continues to transmit the jam signal (48-bit)
- ❑ A senses collision, continues to transmit the jam signal



5: DataLink Layer 5-23

If the adapter detects signal energy from other adapters while transmitting, it stops transmitting its frame and instead transmits a jam signal. Jam signals are simply there to make sure the collision is long enough to be detected by the hardware.

Exponential Backoff

- random time before re-transmission is given by:

```
k = min (10, AttemptNb)
r = random (0, 2k - 1) * slotTime
```

"AttemptNb" is the number of the re-transmission attempt that will be attempted after the random time (k=1 for the first *retransmission*);

"random" returns an integer, uniformly distributed between the two bounds given in argument;

- examples:

first retransmission attempt:

k = 1; r = 0 or r = slotTime

second retransmission attempt (if preceding one failed):

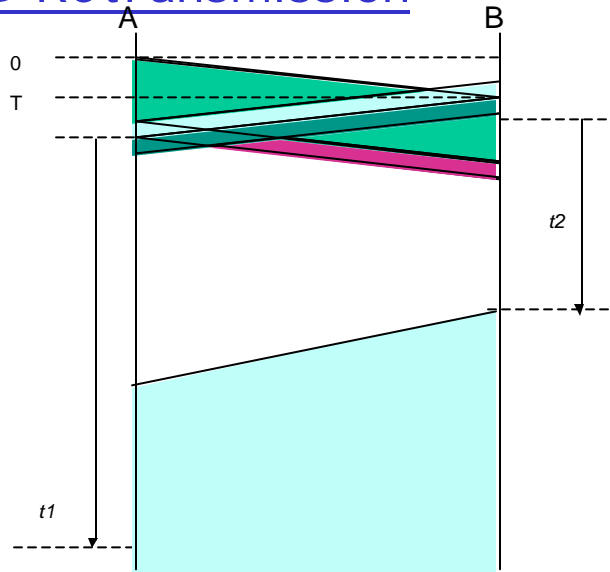
k = 2; r = 0, 1, 2 or 3 * slotTime

5: DataLink Layer 5-24

After aborting (that is, transmitting the jam signal), the adapter enters an **exponential backoff** phase. Specifically, when transmitting a given frame, after experiencing the n th collision in a row for this frame, the adapter chooses a value for K at random from $\{0, 1, 2, \dots, 2^m - 1\}$ where $m := \min(n, 10)$. The adapter then waits $K \cdot 512$ bit times and then returns to sense the channel.

CSMA / CD Retransmission

- ❑ A waits random time t_1
- ❑ B waits random time $t_2 = \text{slottime} < t_1 = 2 * \text{slottime}$
- ❑ B senses channel idle and transmits
- ❑ A senses channel busy and *defers* to B
- ❑ A now waits until channel is idle



5: DataLink Layer 5-25

If both stations would restart retransmission after a deterministic (fixed) time, there will occur a new collision. Therefore, after a collision is detected, stations will re-attempt to transmit after a random time. The random time before retransmission is chosen in such a way that if repeated collisions occur, then the time increases exponentially. The effect is that in case of congestion (too many collisions) the access to the channel is slowed down.

Acknowledgements are not necessary because absence (detection and recovery) of collision means that the frame could be transmitted. The inter-frame delay ("gap") is $9.6 \mu\text{s}$. It is used to avoid blind times, during which adapters are filtering typical noise at transmission ends.

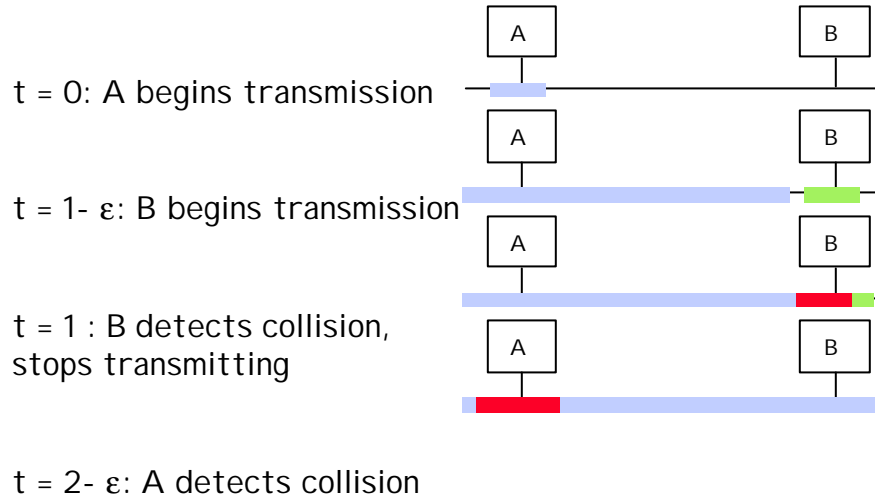
Minimum Frame Size

- a minimum frame size equal to number of bits transmitted during one round trip is required to detect all collisions
- beta = number of bits transmitted by a source during the maximum round trip time for any ethernet network
 - beta = bandwidth - delay product + jam time + safety margin
 - = 512 bits (corresponding to 51.2 μ s at 10 Mb/s)
 - + 8 Bytes as preamble for synchronization = 576 bits
- rule: in Ethernet, all frames must be as large as beta
- properties:
 - P1: all collisions are detected by sources while transmitting
 - P2: collided frames are shorter than beta

5: DataLink Layer 5-26

Beta is called “slotTime” in the IEEE standards. We prefer to use some other name, because it is not a time, but a number of bits.

Minimum Frame Size



5: DataLink Layer 5-27

In this case, B is able to detect the collision (P1).

Also the property P2 is easily proved by noting that if A transmits a frame larger than beta, all the other stations will sense the channel busy.

CSMA / CD performance

- ❑ Maximum utilization of Ethernet is difficult to determine analytically.
 - Approximation :

$$\theta \approx \frac{1}{1 + C \alpha}$$

where $\alpha = \frac{2 \times \text{propagation delay}}{\text{transmission time}}$

L = frame size, b = bandwidth-delay product

- C is a constant : $C = 3.1$ is a pessimistic value; $C = 2.5$ is an approximate value based on simulations
- ❑ for a large network, b is close to 60 Bytes; for traffic with small frames ($L = 64$ bytes), the utilization is less than 30 %. For large frames (1500 Bytes), it is around 90%.
- ❑ Key for high utilization is: bandwidth delay-product \ll frame size

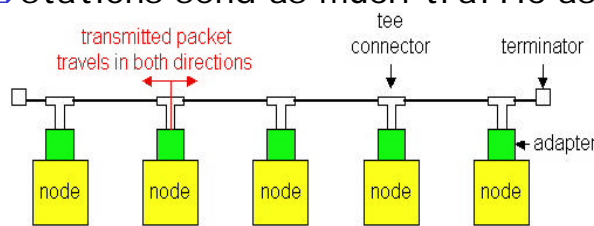
5: DataLink Layer 5-28

The formula with $C= 3.1$ is proven in the next slide. It is a pessimistic estimate.

Proof

We prove the formula with $C = 3.1$

- all frames have a constant length T
- arrival is a transmission or re-transmission submitted when the channel is sensed idle.
- arrival rate μ
- R is the maximum propagation delay.
- stations send as much traffic as possible.



5: DataLink Layer 5-29

The bound is derived as follows. We assume that all frames have a constant length T . We call “arrival” a transmission or re-transmission submitted when the channel is sensed idle. Also call R the maximum propagation delay. Lastly, we assume that stations are trying to saturate the network with as much traffic as can be transmitted. We obtain a pessimistic bound by doing the following worst case assumption: arrivals are always at alternate ends of the network, namely, separated by the maximum propagation delay.

We consider cycles starting with the end of a successful or aborted transmission. Call:

- x_1 : time until all stations know the channel is idle
- x_2 : time from then until next arrival
- x_3 : time until transmission completes or is aborted due to collision.

We have:

$$x_1 = R$$

$$x_2 = 1/\mu \quad \text{in average}$$

$$E(x_3 | \text{collision occurred}) = 2R;$$

$$\text{Prob}(\text{collision occurred}) = 1 - \exp(-R\mu)$$

$$E(x_3 | \text{successful transmission}) = T;$$

$$\text{Prob}(\text{successful transmission}) = \exp(-R\mu)$$

The last formula is because collisions can occur only if an arrival occurs during the propagation time R , because of collision avoidance.

The average cycle time is thus, for this worst case scenario:

$$\tau = R + 1/\mu + 2R(1 - \exp(-R\mu)) + T \exp(-R\mu)$$

and the corresponding utilization:

$$\theta_{\max} = \text{average useful time per cycle} / \text{average cycle duration}$$

$$= T \exp(-R\mu) / \tau$$

Computing the maximum of θ_{\max} with respect to $x = R\mu$ gives the formula (maximum obtained for $x = 0.43$). Note that $\alpha = 2R/T$.

Proof

pessimistic bound

- **worst case assumption:** arrivals are always at alternate ends of the network:
 - separated by the maximum propagation delay.
- consider cycles starting with the end of a successful or aborted transmission.

Proof

- x_1 : time until all stations know the channel is idle
- x_2 time from then until next arrival
- x_3 : time until transmission completes or is aborted due to collision.

We have:

- $x_1 = R$ $x_2 = 1/\mu$ in average
- $x_3 = E(x_3 \mid \text{collision occurred}) * \text{Proba}(\text{collision occurred}) + E(x_3 \mid \text{successful transmission}) * \text{Proba}(\text{successful transmission})$
 - $E(x_3 \mid \text{collision occurred}) = 2R$;
 - $E(x_3 \mid \text{successful transmission}) = T$;
 - $\text{Proba}(\text{collision occurred}) = 1 - \exp(-R\mu)$ (Poisson \rightarrow negative exp.)
 - $\text{Proba}(\text{successful transmission}) = \exp(-R\mu)$
- $x_3 = 2R(1 - \exp(-R\mu)) + T \exp(-R\mu)$

Proof

- The **average cycle time** is thus, for this worst case scenario:
$$\tau = x_1 + x_2 + x_3$$
- $\tau = R + 1/\mu + 2R(1 - \exp(-R\mu)) + T \exp(-R\mu)$
- and the corresponding **utilization**:
$$\theta_{\max} = \text{average useful time per cycle} / \text{average cycle duration}$$
$$= T \exp(-R\mu) / \tau$$
- Computing the maximum of θ_{\max} with respect to $x = R/\mu$ gives
$$\theta \approx \text{?????} 1/(1 + C \alpha) \text{ (maximum obtained for } x = 0.43).$$
- Note that $\alpha = 2R / T$.

“Taking Turns” protocols

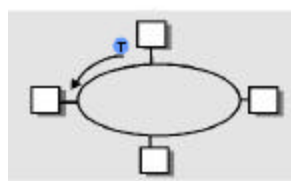
Polling:

- ❑ master node “invites” slave nodes to transmit in turn
- ❑ transmit a max num of frames
- ❑ 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)



5: DataLink Layer 5-33

The **polling protocol** requires one of the nodes to be designated as a master node. The master node **polls** each of the nodes in a round-robin fashion (i.e. with an alternate fair scheme). In particular, the master node first sends a message to node 1, saying that it can transmit up to some maximum number of frames. After node 1 transmits some frames, the master node tells node 2 it can transmit up to the maximum number of frames. (The master node can determine when a node has finished sending its frames by observing the lack of a signal on the channel.) The procedure continues in this manner, with the master node polling each of the nodes in a cyclic manner.

The polling protocol eliminates the collisions and the empty slots that plague the random access protocols. This allows it to have a much higher efficiency. But it also has a few drawbacks: (1) the protocol introduces a polling delay--the amount of time required to notify a node that it can transmit. (If, for example, only one node is active, then the node will transmit at a rate less than R bps, as the master node must poll each of the inactive nodes in turn, each time the active node has sent its maximum number of frames.); (2) is that if the master node fails, the entire channel becomes inoperative.

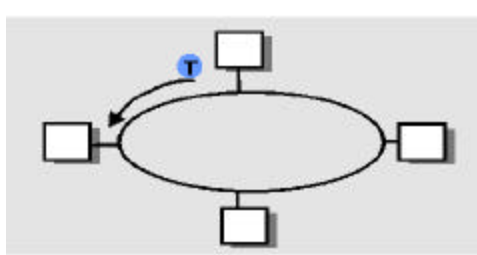
In **token-passing protocol** there is no master node. A small, special-purpose frame known as a **token** is exchanged among the nodes in some fixed order. For example, node 1 might always send the token to node 2, node 2 might always send the token to node 3, node N might always send the token to node 1. When a node receives a token, it holds onto the token only if it has some frames to transmit; otherwise, it immediately forwards the token to the next node. If a node does have frames to transmit when it receives the token, it sends up to a maximum number of frames and then forwards the token to the next node. Token passing is decentralized and has a high efficiency. But it has its problems as well. For example, the failure of one node can crash the entire channel. Or if a node accidentally neglects to release the token, then some recovery procedure must be invoked to get the token back in circulation. Examples of token passing technologies are fiber distributed data interface (FDDI) and Token-Ring (IEEE 802.5).

Access Method Topology

□ Logical Topology:

○ ring:

- all bits are passed from one station to next station, then to next's neighbour, etc
- bits eventually return to originating station which has to remove them
- all stations see all frames
- used by Token Ring and FDDI



5: DataLink Layer 5-34

CSMA/CD uses a bus logical topology, whereas token passing schemes such as the Token Ring and FDDI use ring topologies.

The cabling topology is in general different from the logical topology. A simple network today uses a star topology: all cables go from a central point (the hub) to all end-systems. A more complex network uses a tree of stars.

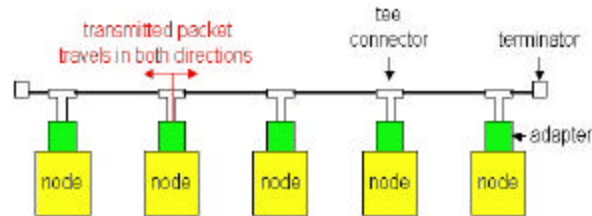
It is the Token Ring network which first introduced a star based cabling topology; this because the designers of the Token Ring took requirement (2b) seriously. With the first Token Rings, a hub contained electro-magnetic relays which would automatically bypass a station which does not correctly function (or is powered off).

Access Method Topology

□ Logical Topology:

○ bus:

- all bits sent by one station are propagated to all stations
- data die at end of bus
- all stations see all frames
- used by Ethernet, Token Bus, LocalTalk, Wireless systems



- ### □ cabling topology = layout of cables = star in most cases (hubs)

5: DataLink Layer 5-35

CSMA/CD uses a bus logical topology, whereas token passing schemes such as the Token Ring and FDDI use ring topologies.

The cabling topology is in general different from the logical topology. A simple network today uses a star topology: all cables go from a central point (the hub) to all end-systems. A more complex network uses a tree of stars.

It is the Token Ring network which first introduced a star based cabling topology; this because the designers of the Token Ring took requirement (2b) seriously. With the first Token Rings, a hub contained electro-magnetic relays which would automatically bypass a station which does not correctly function (or is powered off).

Summary of Multiple Access protocols

- ❑ Channel Partitioning, by time, frequency or code
 - Time Division, Code 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
- ❑ Taking Turns
 - polling from a central site, token passing
 - FDDI, Token Ring

Summary of Multiple Access protocols

- ❑ channel partitioning protocols:
 - share channel efficiently at high load
 - inefficient at low load: delay in channel access, $1/N$ bandwidth allocated even if only 1 active node!
- ❑ Random access protocols
 - efficient at low load: single node can fully utilize channel
 - high load: collision overhead
- ❑ "taking turns" protocols
 - overhead for taking turn
 - not robust to master or node failure or error

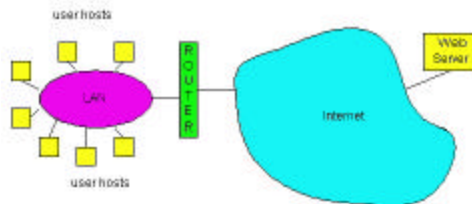
5: DataLink Layer 5-37

The two most desirable properties of a multiple access protocol are (1) when only one node is active, the active node has a throughput of R bps, and (2) when M nodes are active, then each active node has a throughput of nearly R/M bps. The ALOHA and CSMA protocols have this first property but not the second. Taking-turns protocols allows it to have a much higher efficiency by eliminating the collisions and the empty slots present in the random access protocols. This. However these protocol introduces a delay due to the notification to a node that it can transmit. In centralized schemes, if the master node fails, the entire channel becomes inoperative.

Also decentralized tacking turns protocols have similar problems as well. For example, the failure of one node can crash the entire channel. Or if a node accidentally neglects to release the token, then some recovery procedure must be invoked to get the token back in circulation.

LAN technologies

- Data link layer:
 - services, multiple access
- LAN technologies
 - addressing
 - Ethernet
 - repeaters, hubs, bridges, switches
 - virtual LANs



5: DataLink Layer 5-38

Multiple access protocols are extensively used in **local area networks (LANs)**. A LAN is a broadcast channel, which provides to its host access to the Internet through a router. The LAN is a single "link" between each user host and the router, where each node sends frames to each other over a broadcast channel; it therefore uses a link-layer protocol, part of which is a multiple access protocol. The transmission rate, R , of most LANs is very high (up to 1 Gbps).

However, despite the broadcast capability, in general a node in the LAN doesn't want to send a frame to *all* of the other LAN nodes but instead wants to send to some *particular* LAN node. Therefore, the nodes need LAN addresses (in reality this adapters has a LAN address) and the link-layer frame needs a field to contain such a destination address. In this manner, when a node receives a frame, it can determine whether the frame was intended for it or for some other node in the LAN. Note that, with the introduction of layer 2 addresses, broadcast must be explicitly addressed. Additionally, some LANs needs to be interconnected together, and this can be obtained with different type of devices: repeaters, hubs, bridges, switches. This interconnection takes place at layer 2. Finally, several geographically distant LANs can be interconnected only at physical layer and "virtually" interconnected at layer 2 in a so called virtual LAN.

LAN Addresses and ARP

32-bit IP address:

- *network-layer* address
- used to get datagram to destination network (recall IP network definition)

LAN (or MAC or physical) address:

- used to get datagram from one interface to another physically-connected interface (same network)
- 48 bit MAC address (for most LANs) burned in the adapter ROM

Why?

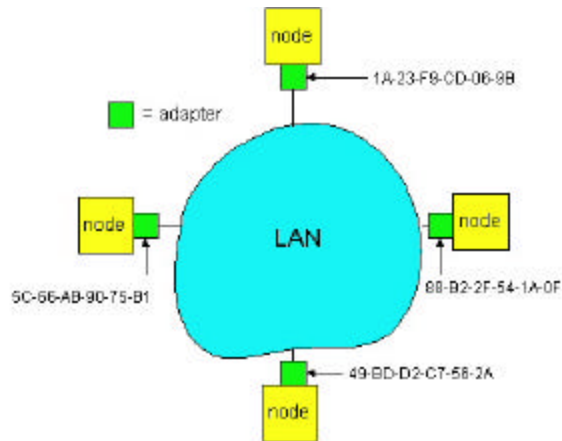
- LANs not only for IP (LAN addresses are neutral)
- if IP used, stored in RAM and reconfigured every movement
- independency of layers

5: DataLink Layer 5-39

A **LAN address** is also variously called a **physical address**, an **Ethernet address**, or a **MAC** (media access control) **address**. For most LANs (including Ethernet and token-passing LANs), the LAN address is six-bytes long, giving 2^{48} possible LAN addresses. These six-byte addresses are typically expressed in hexadecimal notation, with each byte of the address expressed as a pair of hexadecimal numbers.

LAN Addresses and ARP

- each adapter on LAN has unique LAN address
- broadcast address: FF-FF-FF-FF-FF-FF



5: DataLink Layer 5-40

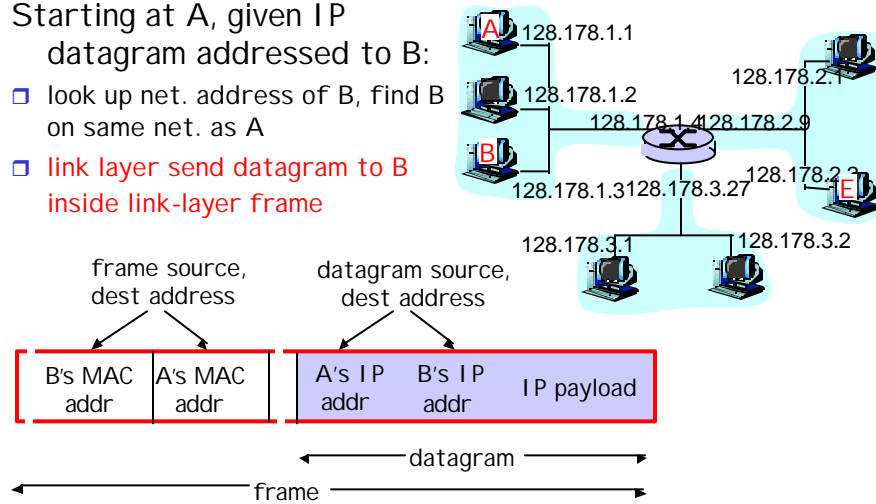
No two adapters have the same address, as the IEEE manages the physical address space; and an address is permanent and flat.

To send to all the hosts in the LAN a sender must put the **LAN broadcast address** into the destination address field of the frame. For LANs that use six-byte addresses (such as Ethernet and token-passing LANs), the broadcast address is a string of 48 consecutive 1s (that is, FF-FF-FF-FF-FF-FF in hexadecimal notation).

Address translation

Starting at A, given IP datagram addressed to B:

- look up net. address of B, find B on same net. as A
- link layer send datagram to B inside link-layer frame

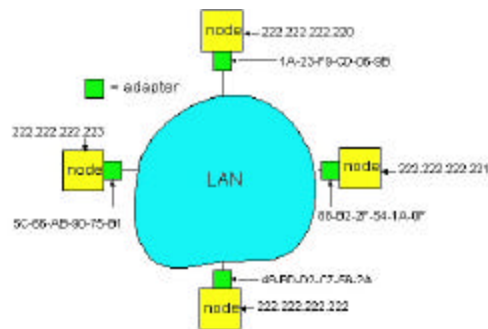


5: DataLink Layer 5-41

Because there are both network-layer addresses (for example, Internet IP addresses) and link-layer addresses (that is, LAN addresses), there is a need to translate between them. The IP datagram contains the IP addresses of source and destination. Once the datagram is passed to the link-layer, it is necessary to specify the MAC address of the destination, in order to transmit the link-layer frame.

ARP: Address Resolution Protocol

ARP is used to determine
The MAC address of B
given B's IP address



- Each IP node (Host, Router) on LAN has **ARP** module, table
- ARP Table: IP/MAC address mappings for some LAN nodes
< IP address; MAC address; TTL >
< >
 - TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

5: DataLink Layer 5-42

For the Internet, this is the job of the address resolution protocol (ARP) [[RFC 826](#)]. Every Internet host and router on a LAN has an **ARP module**. ARP resolves an IP address to a LAN address ***only* for nodes on the same LAN**. The ARP module in each node has a table in its RAM called an **ARP table**. This table contains the mappings of IP addresses to LAN addresses.

For each address mapping the table also contains a time-to-live (TTL) entry, which indicates when the entry will be deleted from the table (typically 20 minutes).

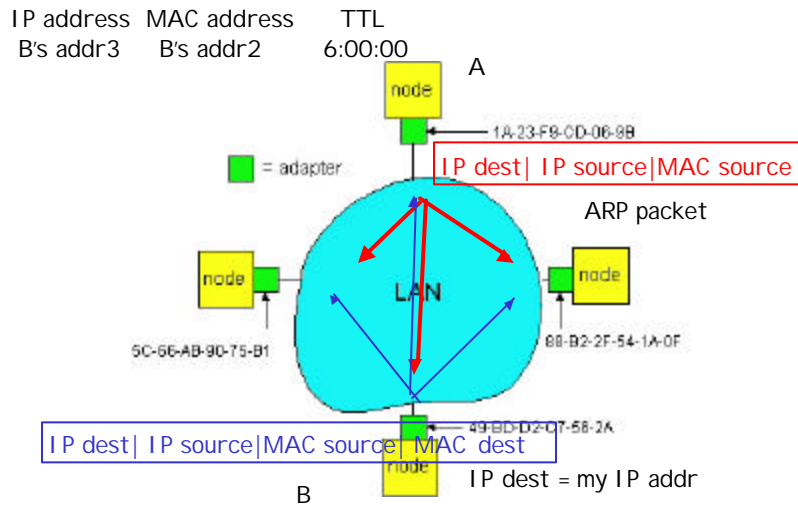
ARP protocol

- ❑ A knows B's IP address, wants to learn physical address of B
- ❑ A **broadcasts** ARP query pkt, containing B's IP address
 - all machines on LAN receive ARP query
- ❑ B receives ARP packet, replies to A with its (B's) physical layer address
- ❑ A caches (saves) IP-to-physical address pairs until information becomes old (times out)
 - soft state: information that times out (goes away) unless refreshed

5: DataLink Layer 5-43

If the table does not contain the MAC address of the destination, the source constructs a special packet called an **ARP packet**. An ARP packet has several fields, including the sending and receiving IP and LAN addresses. Both ARP query and response packets have the same format. The purpose of the ARP query packet is to query all the other nodes on the LAN to determine the LAN address corresponding to the IP address that is being resolved. If the MAC address is returned, the querying node can then update its ARP table and send its IP datagram.

ARP protocol

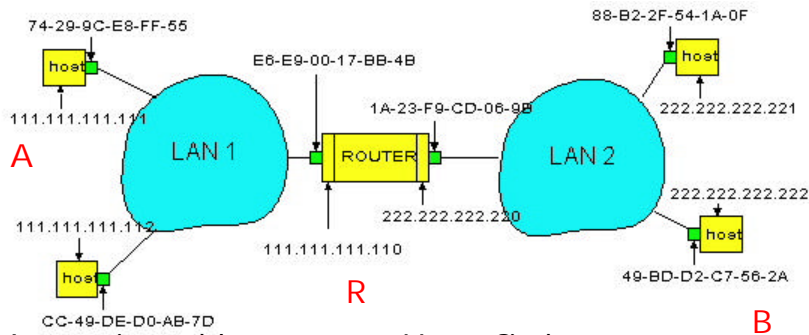


5: DataLink Layer 5-44

If the table does not contain the MAC address of the destination, the source constructs a special packet called an **ARP packet**. An ARP packet has several fields, including the sending and receiving IP and LAN addresses. Both ARP query and response packets have the same format. The purpose of the ARP query packet is to query all the other nodes on the LAN to determine the LAN address corresponding to the IP address that is being resolved. If the MAC address is returned, the querying node can then update its ARP table and send its IP datagram.

Routing to another LAN

walkthrough: routing from A to B via R



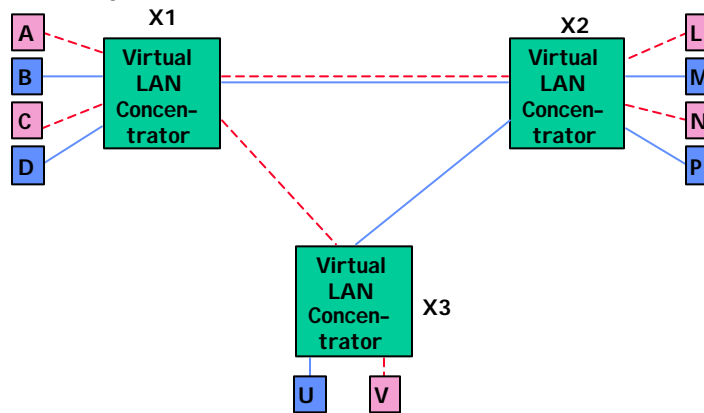
- ❑ In routing table at source Host, find router 111.111.111.110
- ❑ In ARP table at source, find MAC address E6-E9-00-17-BB-4B, etc

5: DataLink Layer 5-45

ARP operates when a node wants to send a datagram to another node *on the same LAN*. The situation is more complex when a node on a LAN wants to send a network-layer datagram to a node *off the LAN*. All of the interfaces connected to LAN 1 have addresses of the form 111.111.111.xxx and all of the interfaces connected to LAN 2 have the form 222.222.222.xxx. Now suppose that host 111.111.111.111 wants to send an IP datagram to host 222.222.222.222. The sending host passes the datagram to its adapter, as usual. However, it is not able to indicate an appropriate destination LAN address. Even if known, the MAC address of the destination cannot be used in this case: none of the adapters on LAN 1 would bother to pass the IP datagram up to its network layer, since the frame's destination address would not match the LAN address of any adapter on LAN 1. And the datagram would die. Indeed, the route of the datagram is decided at network layer. It has to pass through the router R, that will forward it to the LAN2. Therefore, the MAC address that has to be used is the one of the next step, that is the one of the interface on LAN1 of R. In R the packet is passed up to the network layer, where the next routing step is considered. When in the LAN2 (e.g. at the interface of R on LAN2) R uses ARP to get the destination physical layer address. Finally, R creates the frame containing source-to-destination IP datagram and sends it to destination.

Virtual LANs

- several bridged LANs consolidated on one physical layer
- uses Emulated-LAN (ATM), VLAN (proprietary methods)
- works at layer 2



5: DataLink Layer 5-46

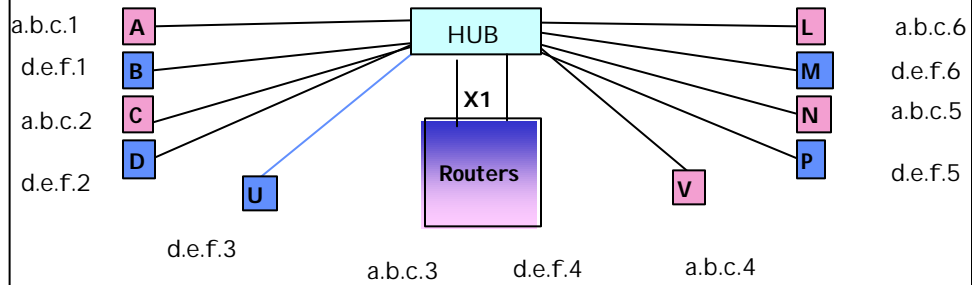
A **Virtual LAN** is a subset of stations physically connected in a LAN that are logically connected. The procedure of logically connecting a group of stations can be seen as a colouring procedure that is managed by a manager generally implemented in a switch.

The picture shows two virtual LANs: (ACLNV) and (BDMPU). For each of the virtual LANs, there exists one or more collision domains per concentrator, plus one per inter-concentrator link. The concentrators perform bridging between the different collision domains of the *same* virtual LAN.

Between X1 and X2, the two virtual LANs use the same physical link. The advantage is that physical location becomes independent of LANs. For example, all servers and routers can be concentrated in the same rooms (ex: U and V). There is no communication between the different virtual LANs at layer 2.

Virtual LANs via subnetting

- ❑ several subnet inside the same physical LAN
- ❑ one physical layer considered as more
- ❑ NOT REALLY a Virtual LAN: it works at layer 3

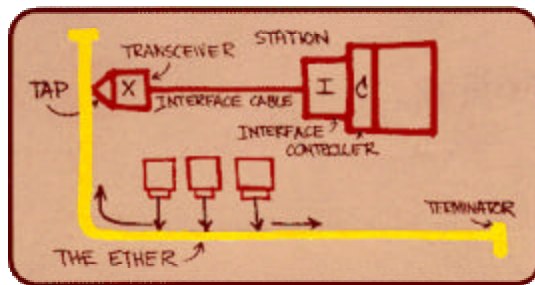


5: DataLink Layer 5-47

Ethernet

"dominant" LAN technology:

- ❑ cheap \$20 for 100Mbps!
- ❑ first widely used LAN technology
- ❑ Simpler, cheaper than token LANs and ATM
- ❑ Kept up with speed race: 10, 100, 1000 Mbps



Metcalfe's Ethernet sketch

5: DataLink Layer 5-48

Today, Ethernet is by far the most prevalent LAN technology, and is likely to remain so for the foreseeable future. There are many reasons for Ethernet's success. First, Ethernet hardware (in particular, network interface cards) has become a commodity and is remarkably cheap. This low cost is also due to the fact that Ethernet's multiple access protocol, CSMA/CD, is completely decentralized, which has also contributed to a simple design. Ethernet is easy to install and manage than token LANs or ATM. Moreover, Ethernet was the first widely deployed high-speed LAN, therefore familiar to many network administrators reluctant to switch to new technologies. Finally, Ethernet is an evolving technology. In the past only 10 Mbps Ethernet was available, but currently so called fast Ethernet allows a nominal bandwidth of 100 Mbps and even 1000 Mbits (1 Gbps).

Ethernet Frame Structure

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



Preamble:

- ❑ 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
- ❑ used to synchronize receiver, sender clock rates

5: DataLink Layer 5-49

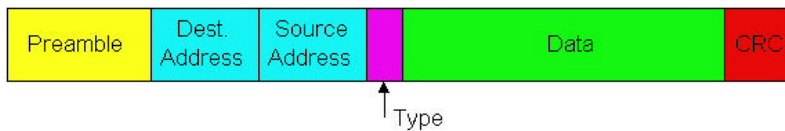
An Ethernet LAN can have a bus topology or a star topology. An Ethernet LAN can run over coaxial cable, twisted-pair copper wire, or fiber optics. Furthermore, Ethernet can transmit data at different rates, specifically, at 10 Mbps, 100 Mbps, and 1 Gbps.

The structure of an Ethernet frame is as follows:

• *Preamble (8 bytes)*. The Ethernet frame begins with an eight-byte preamble field. Each of the first seven bytes of the preamble has a value of 10101010; the last byte is 10101011. The first seven bytes of the preamble serve to "wake up" the receiving adapters and to synchronize their clocks to that of the sender's clock. Why should the clocks be out of synchronization? Keep in mind that adapter A aims to transmit the frame at 10 Mbps, 100 Mbps, or 1 Gbps, depending on the type of Ethernet LAN. However, because nothing is absolutely perfect, adapter A will not transmit the frame at exactly the target rate; there will always be some *drift* from the target rate, a drift which is not known *a priori* by the other adapters on the LAN. A receiving adapter can lock onto adapter A's clock by simply locking onto the bits in the first seven bytes of the preamble. The last two bits of the eighth byte of the preamble (the first two consecutive 1s) alert adapter B that the "important stuff" is about to come. When host B sees the two consecutive 1s, it knows that the next six bytes are the destination address. An adapter can tell when a frame ends by simply detecting absence of current.

Ethernet Frame Structure (more)

- ❑ **Addresses:** 6 bytes, frame is received by all adapters on a LAN and dropped if address does not match
- ❑ **Type:** indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk)
- ❑ **CRC:** checked at receiver, if error is detected, the frame is simply dropped

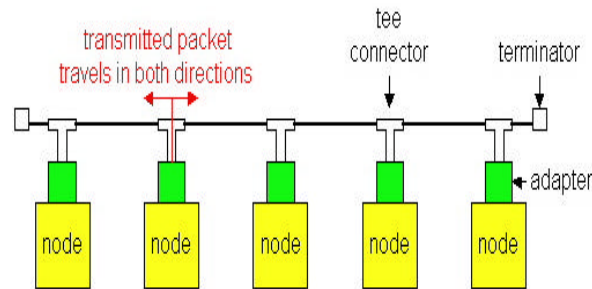


5: DataLink Layer 5-50

- *Destination Address (6 bytes)*. This field contains the destination address. If a node receives a frame with an address *other* than its own MAC address, or the LAN broadcast address, it discards the frame. Otherwise, it passes the contents of the data field to the network layer.
- *Source Address (6 bytes)*. This field contains the LAN address of the source.
- *Data Field (46 to 1500 bytes)*. This field carries the IP datagram. The maximum transfer unit (MTU) of Ethernet is 1500 bytes. The minimum size of the data field is 46 bytes. This means that if the IP datagram is less than 46 bytes, the data field has to be "stuffed" to fill it out to 46 bytes. Data on Ethernet is transmitted least significant bit of first octet first (a bug dictated by Intel processors). Canonical representation thus inverts the order of bits inside a byte (the first bit of the address is the least significant bit of the first byte).
- *Type Field (2 bytes)*. The type field permits Ethernet to distinguish the network-layer protocols.
- *Cyclic Redundancy Check (CRC) (4 bytes)*. To detect whether any errors have been introduced into the frame.

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!

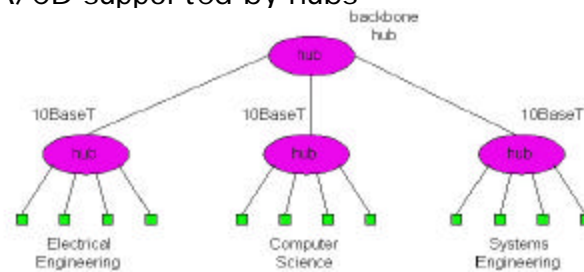
5: DataLink Layer 5-51

10Base2 is a very popular Ethernet technology. The "10" in 10Base2 stands for "10 Mbps"; the "2" stands for "200 meters," which is the approximate maximum distance between any two nodes without repeaters between them. A repeater is a physical-layer device that acts on individual bits rather than on frames. It has two or more interfaces. When a bit, representing a zero or a one, arrives from one interface, the repeater simply recreates the bit, boosts its energy strength, and transmits the bit onto all the other interfaces. Repeaters are commonly used in LANs in order to extend their geographical range. When used with Ethernet, it is important to keep in mind that repeaters do not implement carrier sensing or any other part of CSMA/CD; a repeater repeats an incoming bit on all outgoing interfaces even if there is signal energy on some of the interfaces.

The physical medium used to connect the nodes is **thin coaxial cable** connected in a bus topology.

10BaseT and 100BaseT

- ❑ 10/100 Mbps rate; latter called "fast ethernet"
- ❑ T stands for Twisted Pair
- ❑ Hub to which nodes are connected by twisted pair, thus "star topology"
- ❑ CSMA/CD supported by hubs



5: DataLink Layer 5-52

10BaseT and 100BaseT Ethernet are similar technologies. The first transmits at 10 Mbps and 100BaseT Ethernet transmits at 100 Mbps. 100BaseT is also commonly called "fast Ethernet". Both 10BaseT and 100BaseT Ethernet use a star based topology cabling. There is a central device called a **hub** (also sometimes called a concentrator.) Each adapter on each node has a direct, point-to-point connection to the hub. This connection consists of two pairs of twisted-pair copper wire, one for transmitting and the other for receiving. At each end of the connection there is a connector that resembles the RJ-45 connector used for ordinary telephones. The "T" in 10BaseT and 100BaseT stands for "twisted pair." For both 10BaseT and 100BaseT, the maximum length of the connection between an adapter and the hub is 100 meters; the maximum length between any two nodes is thus 200 meters. A hub is a repeater: when it receives a bit from an adapter, it sends the bit to all the other adapters. In this manner, each adapter can (1) sense the channel to determine if it is idle, and (2) detect a collision while it is transmitting. But hubs are popular because they also provide network management features. When a node has a problem the hub will detect the problem and internally disconnect the malfunctioning adapter.

Gbit 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
- ❑ Full-Duplex at 1 Gbps for point-to-point links

5: DataLink Layer 5-53

Gigabit Ethernet is an extension to a raw data rate of 1,000 Mbps. Gigabit Ethernet is backward compatible with 10BaseT and 100BaseT technologies. It allows for point-to-point links as well as shared broadcast channels. Point-to-point links use switches whereas broadcast channels use hubs. Gbit Ethernet uses CSMA/CD for shared broadcast channels. In order to have acceptable efficiency, the maximum distance between nodes must be severely restricted. It allows for full-duplex operation at 1,000 Mbps in both directions for point-to-point channels.

Interconnecting LANs

Why not just one big LAN?

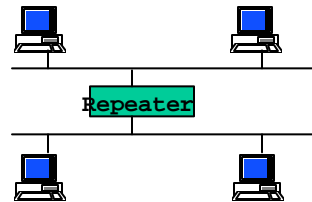
- ❑ Limited amount of supportable traffic: on single LAN, all stations must share bandwidth
- ❑ limited length
- ❑ large "collision domain" (can collide with many stations)
- ❑ broadcast

5: DataLink Layer 5-54

In principle, Internet could be implemented as one big LAN. However, there are several limitations to this solution: (1) the cables used for LANs are usually limited in length, therefore intercontinental distance could not be covered; (2) LANs use shared technologies, therefore the bandwidth is shared among all the station participating to the LAN; (3) statistically, if the number of stations increases, the number of collisions augments.

Repeaters - layer 1

- ❑ Extend network beyond cable length limit
- ❑ Function of a simple (2 port-) repeater:
repeat bits received on one port to other port
if collision sensed on one port, repeat random bits on other port
- ❑ One network with repeaters = **one** collision domain
- ❑ Even with repeaters, network is limited
 - propagation time
 - 51.2µs slotTime includes repeaters
 - at most 4 repeaters in one path
- ❑ Repeaters perform physical layer functions only (bit repeaters)



5: DataLink Layer 5-55

There are limitations on the number of repeaters and cable segments allowed between any two stations on the network. There are two different ways of looking at the same rules:

1. The Ethernet way: A remote repeater pair (with an intermediate point-to-point link) is counted as a single repeater (IEEE calls it two repeaters). You cannot put any stations on the point to point link (by definition!), and there can be two repeaters in the path between any pair of stations. This seems simpler to me than the IEEE terminology, and is equivalent.

2. The IEEE way: There may be no more than five (5) repeated segments, nor more than four (4) repeaters between any two Ethernet stations; and of the five cable segments, only three (3) may be populated. This is referred to as the "5-4-3" rule (5 segments, 4 repeaters, 3 populated segments).

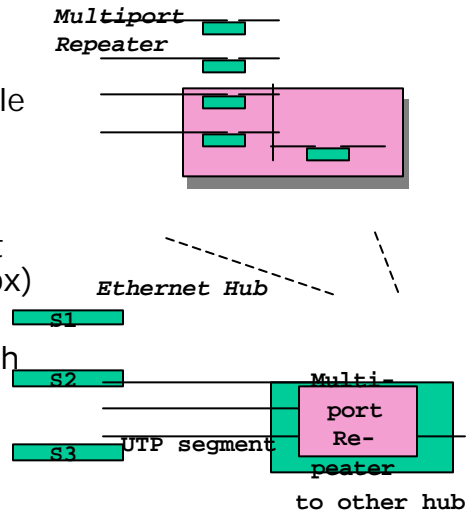
From 3Com, for 110 Mb/s Ethernet:

The 100BASE-T standard defines two classes of repeaters, called Class I and Class II repeaters. A collision domain can include at most one Class I or two Class II repeaters. Key topology rules are as follows:

- Using two Class II repeaters, the maximum diameter of the collision domain is 205 meters (typically 100m + 5m + 100m). With just a single Class II repeater in the collision domain, the diameter can be extended to 309 meters using fiber (typically 100m UTP + 209m fiber downlink). With a single Class I repeater in the collision domain, the diameter can be extended to 261 meters using fiber (typically 100m UTP + 161m fiber downlink).
- Connecting from MAC to MAC (switch to switch, or end-station to switch) using half-duplex 100BASE-FX, a 412-meter fiber run is allowed.
- For very long distance runs, a nonstandard, full-duplex version of 100BASE-FX can be used to connect two devices over a 2-kilometer distance. The IEEE is currently working on a standard for full duplex, but at this time all full-duplex solutions are proprietary.

From Repeaters to Hubs - layer1

- ❑ Multiport repeater (n ports) logically equivalent to: n simple repeaters connected to one internal Ethernet segment
- ❑ Multi-port repeaters make it possible to use point-to-point segments (Ethernet in the box)
- ❑ Hubs **do not isolate** collision domains: node may collide with any node residing at any segment in LAN
- ❑ simple, inexpensive device



5: DataLink Layer 5-56

The simplest way to interconnect LANs is to use a hub. A **hub** is a simple device that takes an input (that is, a frame's bits) and retransmits the input on the hub's outgoing ports. Hubs are essentially repeaters, operating on bits. They are thus physical-layer devices. When a bit comes into a hub interface, the hub simply broadcasts the bit on all the other interfaces. All nodes belong to the same **collision domain**, that is, whenever two or more nodes on the LAN segments transmit at the same time, there will be a collision and all of the transmitting nodes will enter exponential backoff.

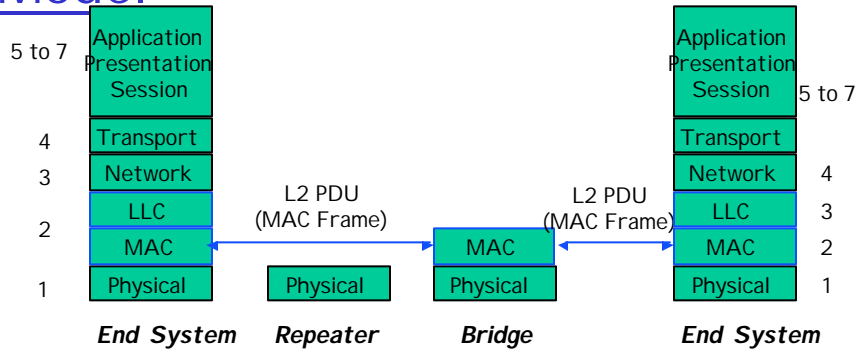
Bridges – layer 2

- ❑ **Link Layer devices:** operate on Ethernet frames, examining frame header and selectively forwarding frame based on its destination
- ❑ Bridge **isolates collision** domains since it buffers frames
- ❑ When frame is to be forwarded on segment, bridge uses CSMA/CD to access segment and transmit
- ❑ Can connect different type Ethernet since it is a buffering device
- ❑ two bridges protocols: transparent bridge and spanning tree protocol

5: DataLink Layer 5-57

Bridges operate on Ethernet frames and thus are layer-2 devices. In fact, **bridges** are full-fledged packet switches that forward and filter frames using the LAN destination addresses. When a frame comes into a bridge interface, the bridge does not just copy the frame onto all of the other interfaces. Instead, the bridge examines the layer-2 destination address of the frame and attempts to forward the frame on the interface that leads to the destination. First, bridges permit isolates collision. Second, bridges can interconnect different LAN technologies, including 10 Mbps and 100 Mbps Ethernets. Third, there is no limit to how large a LAN can be when bridges are used to interconnect LAN segments; in theory, using bridges, it is possible to build a LAN that spans the entire globe.

Repeaters and Bridges in OSI Model

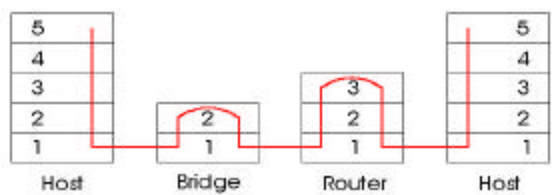


- Bridges are layer 2 intermediate systems
- Repeaters are in layer 1 intermediate systems
- There also exist layer 3 intermediate systems (IP routers) -> module M3

5: DataLink Layer 5-58

Bridges vs. Routers

- ❑ both store-and-forward devices
 - routers: network layer devices (examine network layer headers)
 - bridges are Link Layer devices
- ❑ routers elaboration ~ 10 times bridges elaboration
- ❑ bridges are plug-and-play

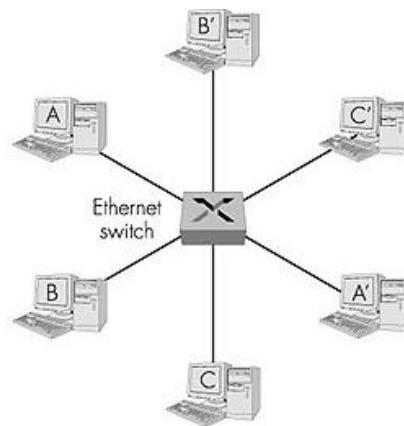


5: DataLink Layer 5-59

Routers are store-and-forward packet switches that forward packets using network-layer addresses. Although a bridge is also a store-and-forward packet switch, it is fundamentally different from a router in that it forwards packets using LAN addresses. Whereas a router is a layer 3 packet switch, a bridge is a layer-2 packet switch.

Ethernet Switches - layer 2

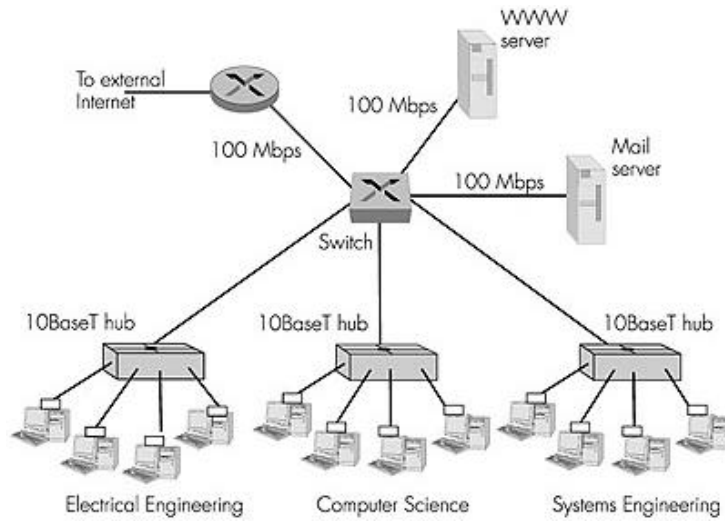
- ❑ layer 2 (frame) forwarding, filtering using LAN addresses
- ❑ **Switching**: A-to-B and A'-to-B' simultaneously, no collisions
- ❑ large number of interfaces
- ❑ often: individual hosts, star-connected into switch
 - Ethernet, but no collisions!



5: DataLink Layer 5-60

Ethernet **switches** are in essence high-performance multi-interface bridges. As do bridges, they forward and filter frames using LAN destination addresses, and they automatically build forwarding tables using the source addresses in the traversing frames. The most important difference between a bridge and switch is that bridges usually have a small number of interfaces (that is, 2-4), whereas switches may have dozens of interfaces. A large number of interfaces generates a high aggregate forwarding rate through the switch fabric, therefore necessitating a high-performance design (especially for 100 Mbps and 1 Gbps interfaces). When a host has a direct connection to a switch (rather than a shared LAN connection), the host is said to have **dedicated access**.

Ethernet Switches (more)



5: DataLink Layer 5-61

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 used to be considered “high layer” in protocol stack!)

5: DataLink Layer 5-62

The **point-to-point protocol (PPP)** [[RFC 1661](#); [RFC 2153](#)] is a data-link layer protocol that operates over a **point-to-point link**--a link directly connecting two nodes, one on each end of the link. The point-to-point link over which PPP operates might be a serial dialup telephone line (for example, a 56K modem connection), a SONET/SDH link, an X.25 connection, or an ISDN circuit. As noted above, PPP has become the protocol of choice for connecting home users to their ISPs over a dialup connection.

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 aliveness:** detect, signal link failure to network layer
- ❑ **network layer address negotiation:** endpoint can learn/configure each other's network address

5: DataLink Layer 5-63

The **point-to-point protocol (PPP)** [RFC 1661; RFC 2153] is a data-link layer protocol that operates over a **point-to-point link**--a link directly connecting two nodes, one on each end of the link. The point-to-point link over which PPP operates might be a serial dialup telephone line (for example, a 56K modem connection), a SONET/SDH link, an X.25 connection, or an ISDN circuit. As noted above, PPP has become the protocol of choice for connecting home users to their ISPs over a dialup connection.

The PPP link remains configured for communication until an LCP terminate-request packet is sent. If a terminate-request LCP frame is sent by one end of the PPP link and replied to with a terminate-ack LCP frame, the link then enters the dead state.

In summary, PPP is a data-link layer protocol by which two communicating link-level peers, one on each end of a point-to-point link, exchange PPP frames containing network layer datagrams. The principal components of PPP are:

- *Framing.* A method for encapsulating data in a PPP frame, identifying the beginning and end of the frame, and detecting errors in the frame.
- *Link-control protocol.* A protocol for initializing, maintaining, and taking down the PPP link.
- *Network-control protocols.* A family of protocols, one for each upper layer network protocol, that allows the network-layer modules to configure themselves before network-level datagrams begin flowing across the PPP link.

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!

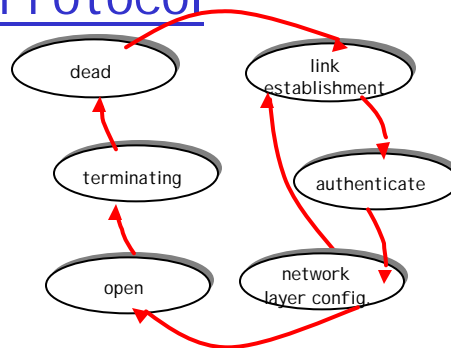
5: DataLink Layer 5-64

- Error correction.* PPP is required to detect bit errors but is *not* required to correct them.
- Flow control.* A PPP receiver is expected to be able to receive frames at the full rate of the underlying physical layer. If a higher layer cannot receive packets at this full rate, it is then up to the higher layer to drop packets or throttle the sender at the higher layer.
- Sequencing.* PPP is *not* required to deliver frames to the link receiver in the same order in which they were sent by the link sender.
- Multipoint links.* PPP need only operate over links that have a single sender and a single receiver.

PPP Data Control Protocol

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



5: DataLink Layer 5-65

Before any data is exchanged over a PPP link, the two peers (one at each end of the PPP link) must first specify the desired link configuration options using an LCP configure-request frame. Once the link has been established, link options negotiated, and the authentication (if any) performed, the two sides of the PPP link then exchange network-layer-specific network control packets with each other. In the case of IP protocol, IP control protocol (IPCP) is used.

Chapter 5: Summary

- ▣ principles behind data link layer services:
 - sharing a broadcast channel: multiple access
 - link layer addressing, ARP
- ▣ various link layer technologies
 - Ethernet
 - repeaters, hubs, bridges, switches
 - PPP