

Capa de Enlace

- ❑ 5.1 Introducción y servicios
- ❑ 5.2 Detección y corrección de errores
- ❑ 5.3 Protocolos de acceso múltiple
- ❑ 5.4 Direccionamiento de Capa de Enlace
- ❑ 5.5 Ethernet
- ❑ 5.6 Switches de Capa de Enlace
- ❑ 5.7 PPP

LAN

- ❑ Recordar que LAN (*Local Area Network*) es una red concentrada en un área geográfica concreta que podemos asimilarla a una oficina, un piso, un edificio, un campus.
- ❑ Recordar además:
 - PAN
 - MAN, WAN
- ❑ Velocidades típicas actuales: 10 Mbps, 100 Mbps, 1 Gbps.
- ❑ Ya es una realidad: 10 Gbps en cobre.

Direcciones MAC

- Direcciones IP de 32 bits:
 - direcciones de la *capa de red*
 - utilizada para llevar el datagrama a la subred IP destino

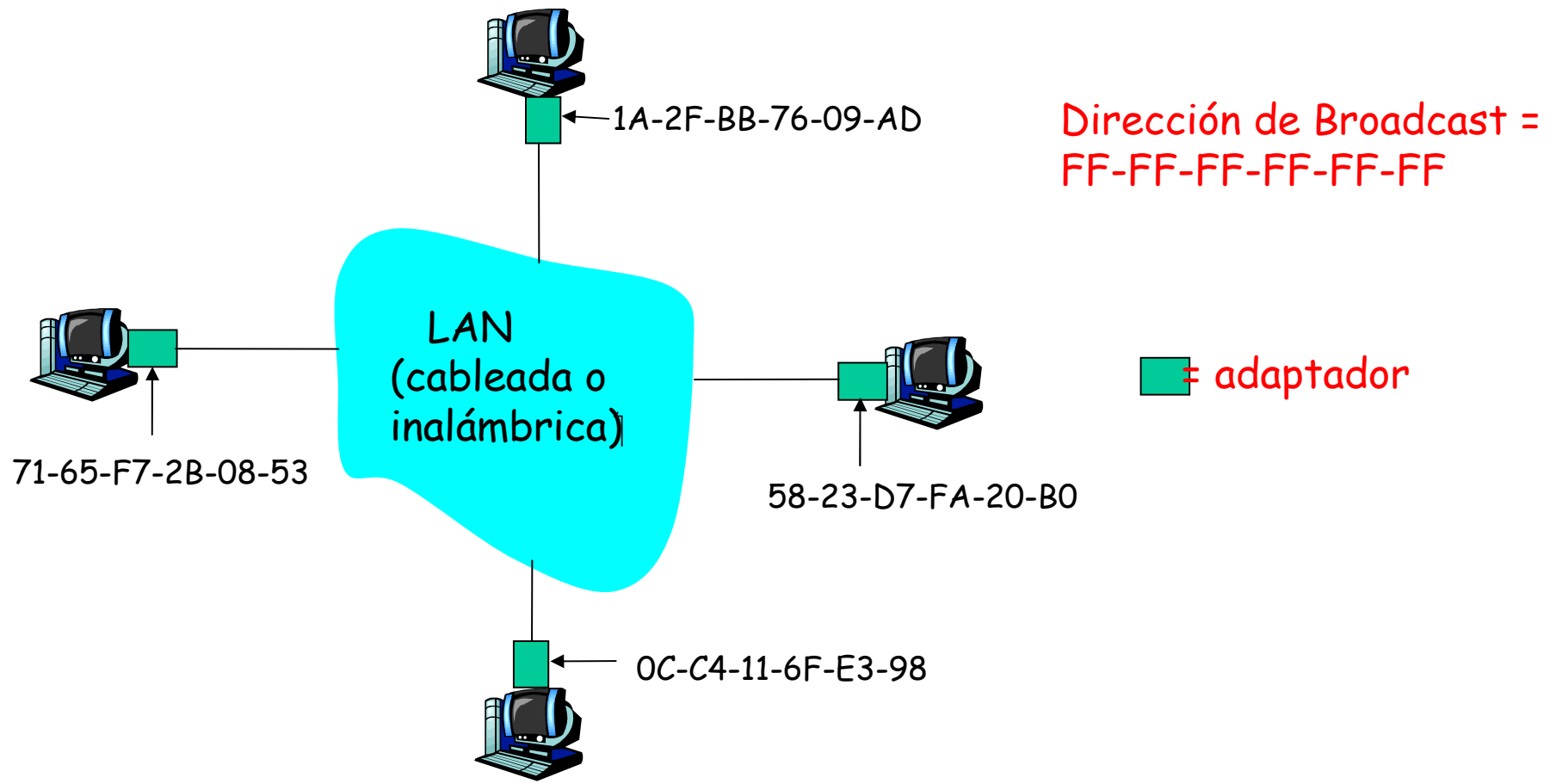
- Dirección MAC (o LAN o física o hardware o del adaptador o "Ethernet"):
 - función: *llevar la trama de una interfaz a otra interfaz físicamente conectada (misma red)*
 - Direcciones MAC de 48 bits (en la mayoría de las redes LAN)
 - grabada en la ROM de la NIC; en algunos casos (cada vez más) configurable por software

Direcciones MAC

- ❑ asignación de direcciones MAC administrada por IEEE
- ❑ los fabricantes compran porciones del espacio de direcciones MAC (para asegurar unicidad)
 - OUI (*Organizationally Unique Identifier*): 3 primeros octetos, asignados a las compañías (*company_id*)
 - <http://standards.ieee.org/regauth/oui/index.shtml>
 - Restantes 3 octetos (*NIC Specific*): administrados por cada compañía
- ❑ Dirección MAC plana → portable
 - puedo mover la tarjeta de una LAN a otra
- ❑ Dirección IP jerárquica → no portable
 - la dirección depende de la subred IP a la que el nodo está conectado

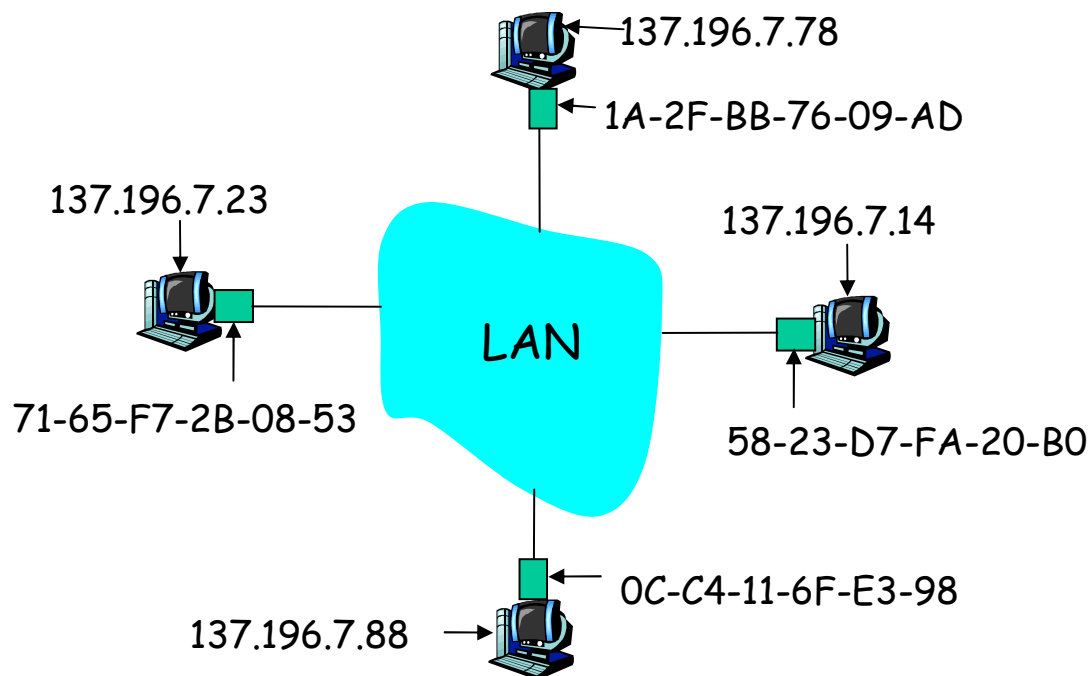
Direcciones MAC

Cada adaptador en la LAN tiene una dirección LAN única



ARP: Address Resolution Protocol

Pregunta: ¿Cómo determinamos la dirección MAC de B, conociendo la dirección IP de B?



- ❑ Cada nodo IP (host, router) en la LAN tiene una tabla **ARP**
- ❑ Tabla ARP: mapeo de direcciones IP/MAC para algunos nodos de la LAN

< dirección IP; dirección MAC; TTL >

- TTL (*Time To Live*): tiempo después del cual el mapeo de direcciones debe ser olvidado (por ejemplo, 20 min)

Protocolo ARP: en la misma red

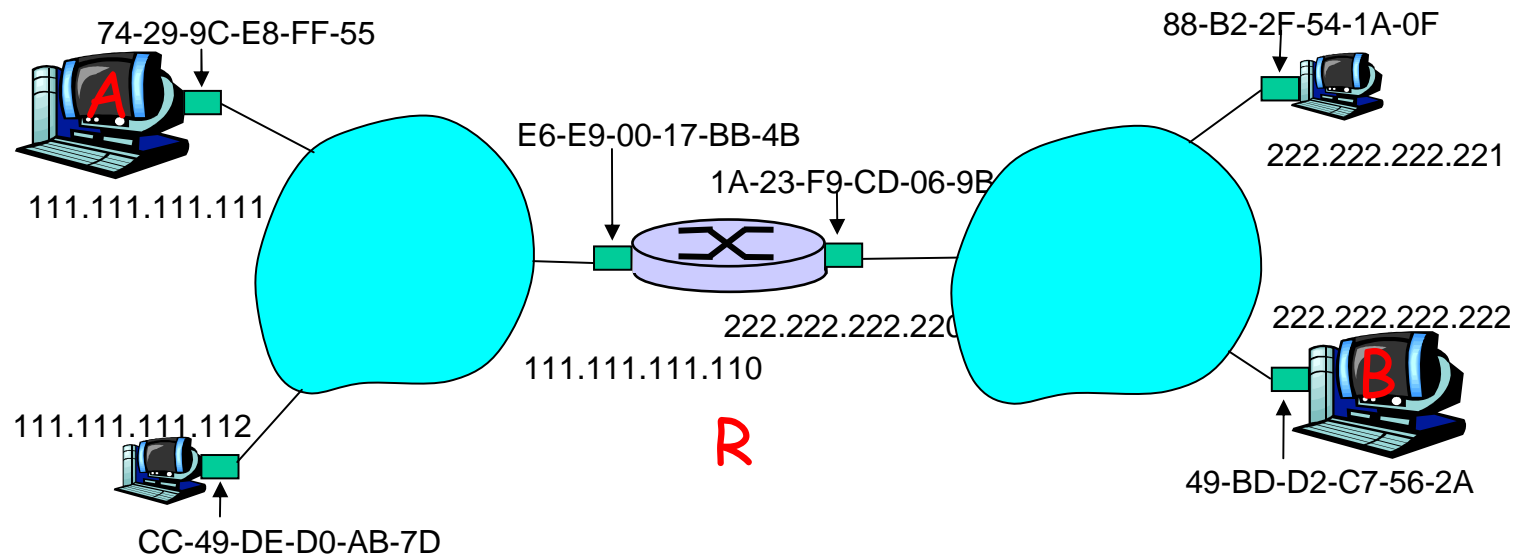
LAN

- ❑ A quiere enviar un datagrama a B, y la dirección MAC de B no está en la tabla ARP de A.
- ❑ A realiza un **broadcast** de un paquete **ARP query**, conteniendo la dirección IP de B
 - Dirección MAC destino = FF-FF-FF-FF-FF-FF
 - Todas las máquinas en la LAN reciben el **ARP query**
- ❑ B recibe el paquete ARP y responde a A con su dirección MAC (**ARP reply**)
 - La trama es enviada a la dirección MAC de A (**unicast**)
- ❑ A salva (*cache*) el par direcciones IP-MAC en su tabla ARP hasta que la información se considere vieja (*timeout*)
- ❑ ARP es "plug-and-play":
 - Los nodos crean (bajo demanda) sus tablas ARP sin intervención del administrador de la red
- ❑ RFC 826
- ❑ ¿ARP es análogo a DNS?

Direccionamiento: *routing* hacia otra LAN

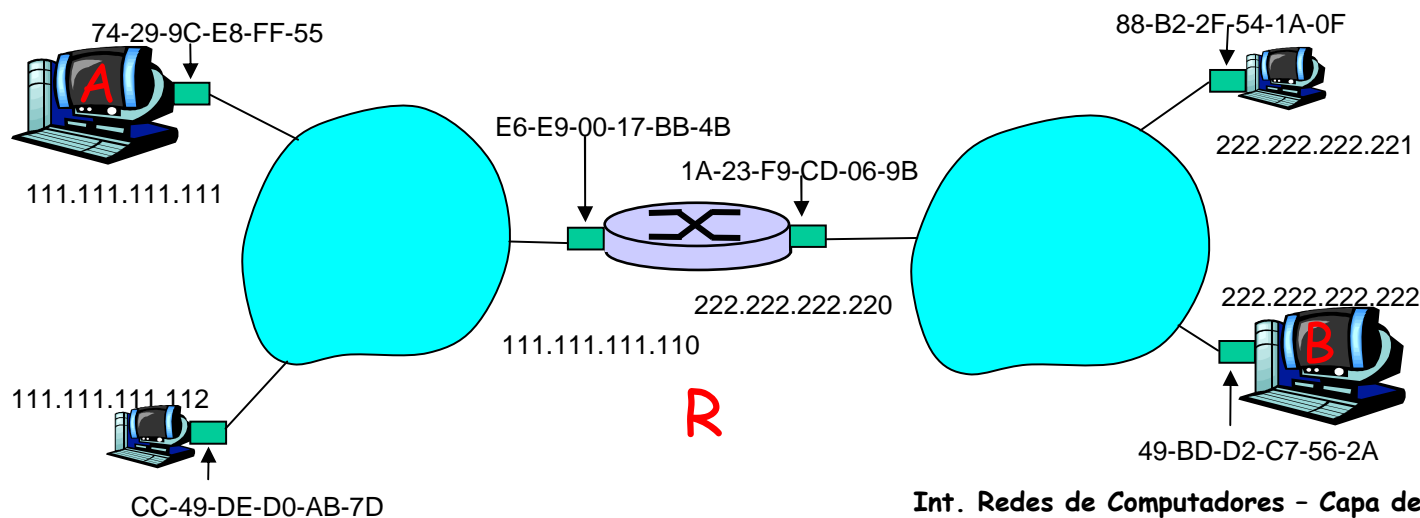
datagrama desde A hasta B, vía R

asumimos que A conoce la dirección IP de B



- dos tablas ARP en el router R, una para cada red IP (LAN)

- ❑ A crea el datagrama IP con origen A, destino B
- ❑ A utiliza ARP para obtener la dirección MAC de R para 111.111.111.110
- ❑ A crea una trama con la dirección MAC de R como destino; la trama contiene el datagrama IP "A-to-B"
- ❑ La NIC de A envía la trama
- ❑ La NIC de R recibe la trama
- ❑ R quita el datagrama IP de la trama Ethernet, observando que está destinado a B
- ❑ R consulta la tabla de *forwarding* para identificar la interfaz de salida
- ❑ R utiliza ARP para obtener la dirección MAC de B
- ❑ R crea una trama que contiene el datagrama IP "A-to-B" y la envía a B



Capa de Enlace

- ❑ 5.1 Introducción y servicios
- ❑ 5.2 Detección y corrección de errores
- ❑ 5.3 Protocolos de acceso múltiple
- ❑ 5.4 Direccionamiento de Capa de Enlace
- ❑ 5.5 Ethernet
- ❑ 5.6 Switches de Capa de Enlace
- ❑ 5.7 PPP

Ethernet

Tecnología LAN cableada dominante:

- ❑ Creada "en los 70" (Metcalfe & Boggs)
- ❑ NICs baratas (USD 5) y switches baratos
- ❑ Primera tecnología LAN ampliamente utilizada
- ❑ Más simple y barata que *token* LANs y ATM
- ❑ Velocidades: 10 Mbps - 10 Gbps

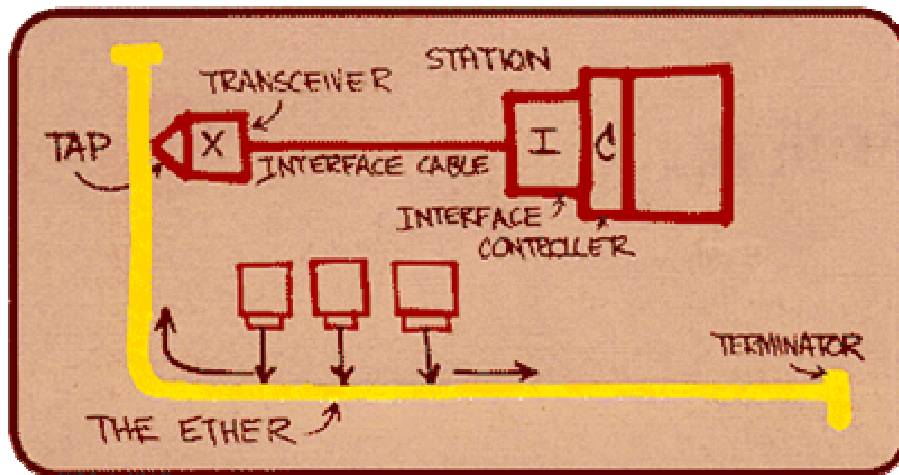
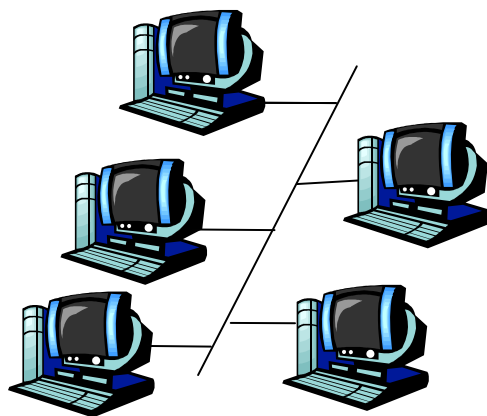


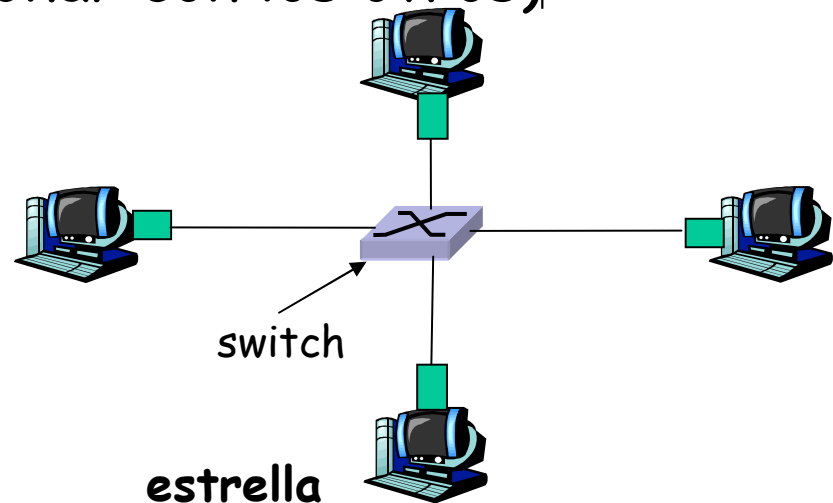
Diagrama de Ethernet de Robert Metcalfe

Topología en estrella

- ❑ la topología en bus fue popular hasta mediados de los 90
 - todos los nodos en el mismo dominio de colisión (pueden colisionar con cualquiera de los otros)
- ❑ hoy: prevalece la topología *estrella*
 - **switch** activo en el centro (desde "fines de los 90")
 - cada "spoke" corre el protocolo Ethernet (los nodos no pueden colisionar con los otros)

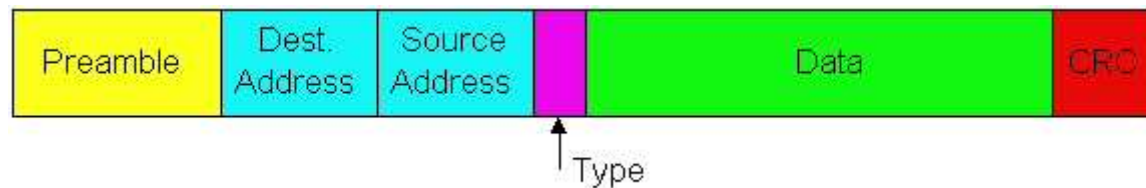


bus: cable coaxial



Estructura de la trama Ethernet

- El adaptador del emisor encapsula el datagrama IP (u otro paquete de protocolo de capa de red) en una **trama Ethernet**

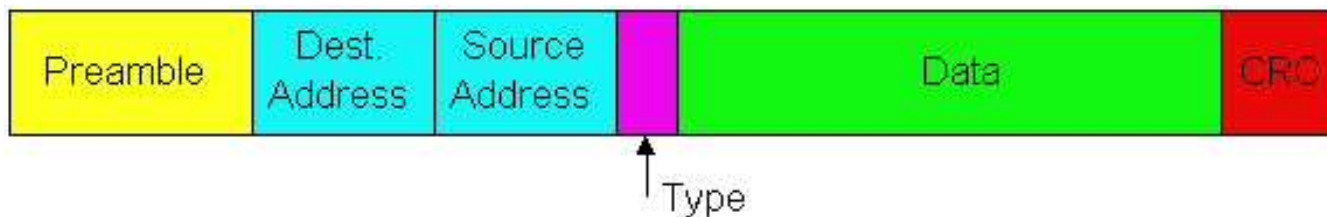


Preamble:

- siete bytes con el patrón 10101010 seguido por un byte con el patrón 10101011
- utilizado para despertar al receptor y sincronizar los relojes de emisor y receptor

Estructura de la trama Ethernet (más)

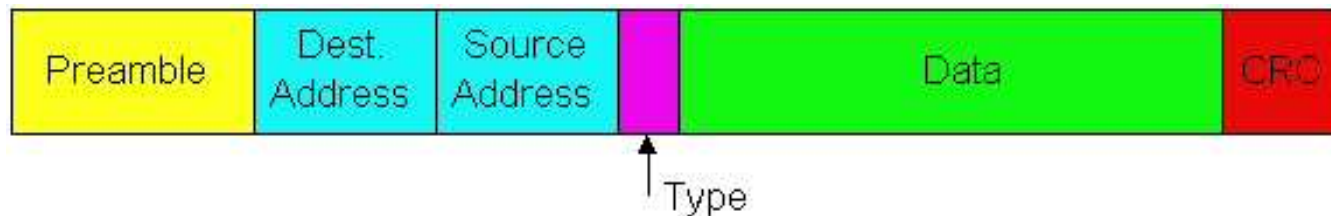
- **Direcciones:** 6 bytes cada una
 - si el adaptador recibe una trama con dirección destino la suya o la dirección de *broadcast*, (ej. paquete ARP), pasa los datos en la trama al protocolo de capa de red
 - en otro caso, el adaptador descarta la trama
- **Type:** 2 bytes
 - multiplexación
 - indica el protocolo de la capa superior (casi siempre IP pero otros es posible, p.e., IPX, AppleTalk)



Estructura de la trama Ethernet (más)

- **Data:** de 46 a 1500 bytes

- **CRC:** 4 bytes
 - CRC-32
 - chequeado en el receptor, si un error es detectado, la trama es descartada
 - Para calcularlo se utiliza todo menos el "*Preamble*"



Ethernet: servicio no confiable, no orientado a conexión

- ❑ **No orientado a conexión:** No hay *handshaking* entre las NICs de emisor y receptor
- ❑ **No confiable:** la NIC que recibe no envía ACKs o NAKs a la NIC emisora
 - el flujo de datagramas pasados a la capa de red puede tener huecos (datagramas perdidos)
 - los huecos serán llenados si la aplicación utiliza TCP
 - en otro caso, la aplicación verá los huecos
- ❑ Protocolo MAC de Ethernet: **CSMA/CD**
- ❑ La detección de colisiones es un servicio de Capa Física

Algoritmo CSMA/CD de Ethernet

1. La NIC recibe un datagrama de la capa de red; crea una trama, y la pone en el buffer del adaptador
2. Si la NIC determina que el canal está libre por *96 bit times*, comienza la transmisión de la trama (no hay *slots*) Si está ocupado, espera hasta que el canal esté libre y luego transmite
3. Si la NIC transmite la trama entera sin detectar otra transmisión, ¡éxito!
4. Si la NIC detecta otra transmisión cuando está transmitiendo, aborta y envía una señal *jam* de 48 bits
5. Después de abortar (o sea, después de transmitir la *jam signal*, la NIC entra en la fase ***exponential backoff***: después de la n^{th} colisión para la misma trama, la NIC elige K aleatorio de $\{0,1,2,\dots,2^m-1\}$ donde $m = \min(n,10)$. La NIC espera $K \cdot 512 \text{ bit times}$, y vuelve al Paso 2

CSMA/CD de Ethernet (más)

- No existe ningún tipo de coordinación entre los adaptadores de una red Ethernet

Señal Jam: asegura que todos los otros transmisores estén interados de la colisión; 48 bits

Bit time: tiempo necesario para transmitir un bit. $0.1 \mu s$ para 10 Mbps Ethernet; for $K=1023$, el tiempo de espera es de aprox. 50 msec

Exponential Backoff:

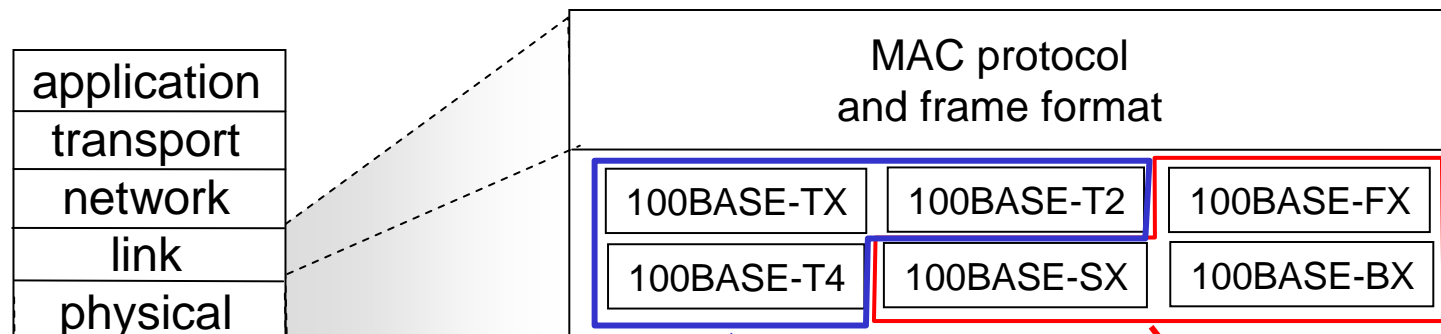
- **Objetivo:** adaptar los intentos de retransmisión a la carga actual estimada
 - alta carga: la espera randómica debe ser mayor
- primera colisión: selecciona K de $\{0,1\}$; retardo es $K \cdot 512 \text{ bit time} = K \cdot \text{slot Time}$
- después de la segunda colisión: selecciona K de $\{0,1,2,3\}$...
- después de 10 colisiones, selecciona K de $\{0,1,2,3,4,\dots,1023\}$
- después de 16 colisiones consecutivas para la misma trama, se descarta

Más sobre Ethernet

- ❑ En una red LAN Ethernet, el tiempo se alternan momentos "libre", "transmisión" y "colisión"
- ❑ Notación: *RRRBase-TTT*
 - *RRR*: Velocidad
 - 10: 10 Mbps
 - 100: 100 Mbps
 - ...
 - **Base**: Se transmite en bandabase (un solo canal de comunicación en el medio físico)
 - *TTT*: especificación relativa al medio físico en uso
 - 5, 2: coaxial, 500 o 200 metros
 - T: par trenzado (UTP)
 - FX: Fibra óptica
 - ...

802.3 Ethernet Standards: Capas de Enlace y Física

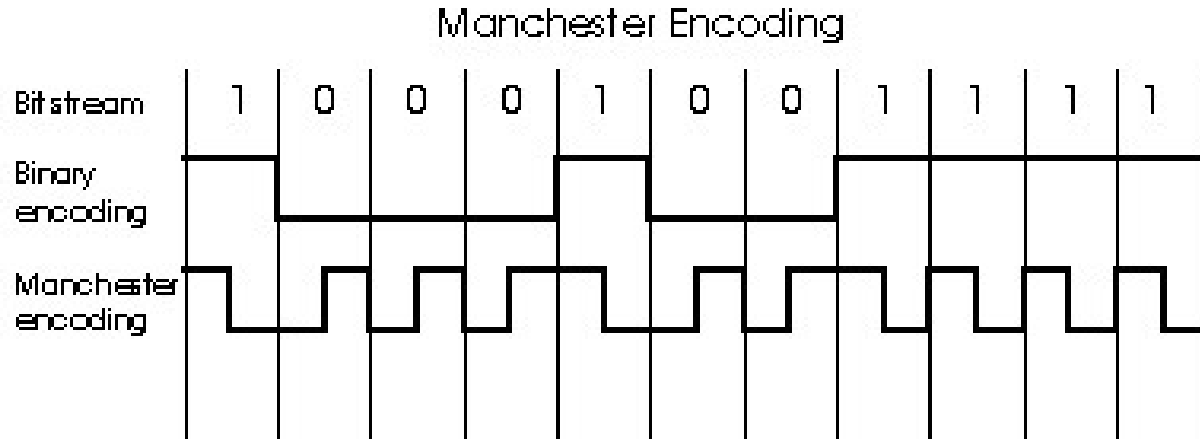
- *varios* diferentes estándares Ethernet
 - protocolo MAC y formato de trama único
 - diferentes velocidades: 2 Mbps, 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps
 - diferentes medios físicos: fibra óptica, cable



cobre (par trenzado)
Capa física

Fibra óptica
Capa física

Codificación Manchester



- ❑ Utilizado en 10BaseT
- ❑ Cada bit tiene una transición
- ❑ Permite que los relojes de los nodos emisores y receptores siempre estén sincronizados entre sí
 - No se requiere un reloj centralizado, global

Capa de Enlace

- ❑ 5.1 Introducción y servicios
- ❑ 5.2 Detección y corrección de errores
- ❑ 5.3 Protocolos de acceso múltiple
- ❑ 5.4 Direccionamiento de Capa de Enlace
- ❑ 5.5 Ethernet
- ❑ 5.6 Switches de Capa de Enlace
- ❑ 5.7 PPP

Dominios

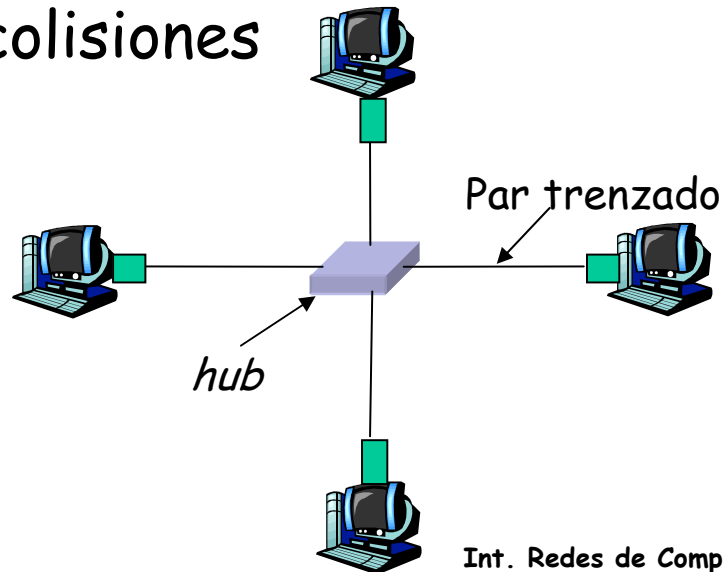
- ❑ **Dominio de colisión**
 - Porción de la red hasta donde se propaga una colisión

- ❑ **Dominio de broadcast**
 - Porción de la red hasta donde se propaga un broadcast

Hubs

... repetidores de Capa Física ("tonto"):

- los bits que llegan en un *link* salen por *todos* los otros *links* a la misma velocidad
- todos los nodos conectados al *hub* pueden colisionar con los otros
- no existe *buffering* de tramas
- no hay CSMA/CD en el hub: la NIC del *host* detecta las colisiones

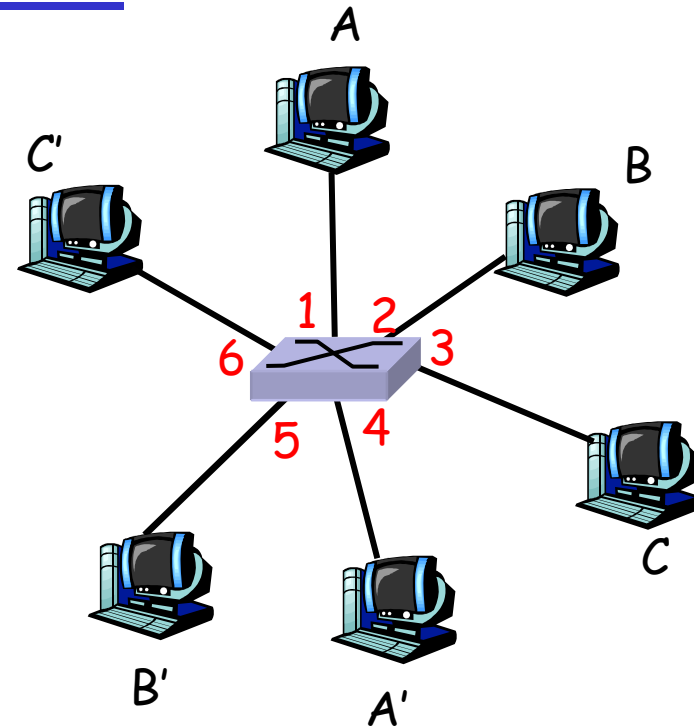


Switch

- ❑ dispositivo de Capa de Enlace: más "inteligente" que los *hubs*, tienen un rol *activo*
 - almacenamiento, envío de tramas Ethernet
 - examina la dirección MAC destino de la trama entrante, realiza un envío **selectivo** de la trama a uno o más *links* de salida; cuando la trama será enviada en un segmento, utiliza CSMA/CD para acceder al segmento
- ❑ *transparente*
 - los *hosts* no se "enteran" de la presencia de los *switches*
- ❑ *plug-and-play, self-learning*
 - los *switches* no necesitan ser configurados (para su operación básica)

Switch: permite múltiples transmisiones simultáneas

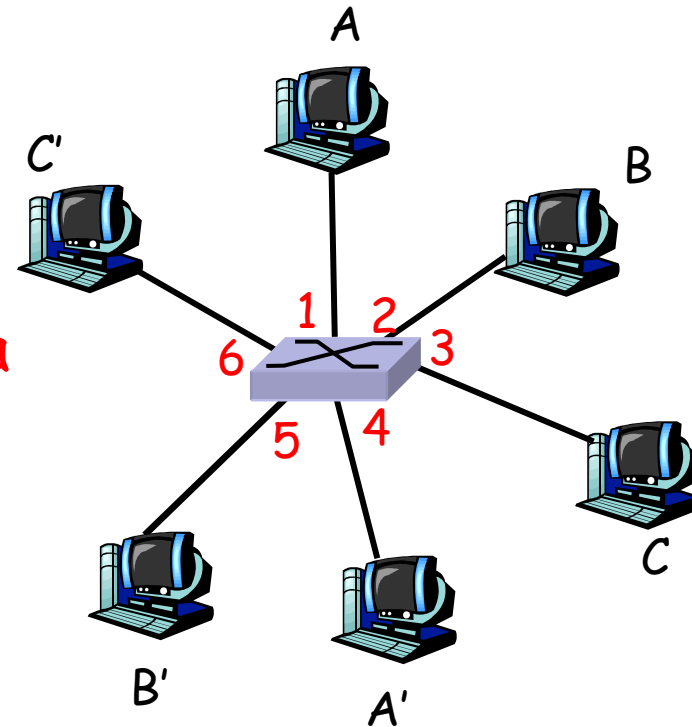
- ❑ Los *hosts* tienen conexiones dedicadas, directas al *switch*
- ❑ Los *switches* hacen *buffer* de las tramas
- ❑ El protocolo Ethernet es utilizado en *cada* link entrante, pero no hay colisiones; *full duplex*
 - cada *link* es su propio dominio de colisión
- ❑ **switching**: A-to-A' and B-to-B' simultáneamente, sin colisiones
 - no posible con *hub*



*switch con seis interfaces
(1,2,3,4,5,6)*

Tabla del Switch

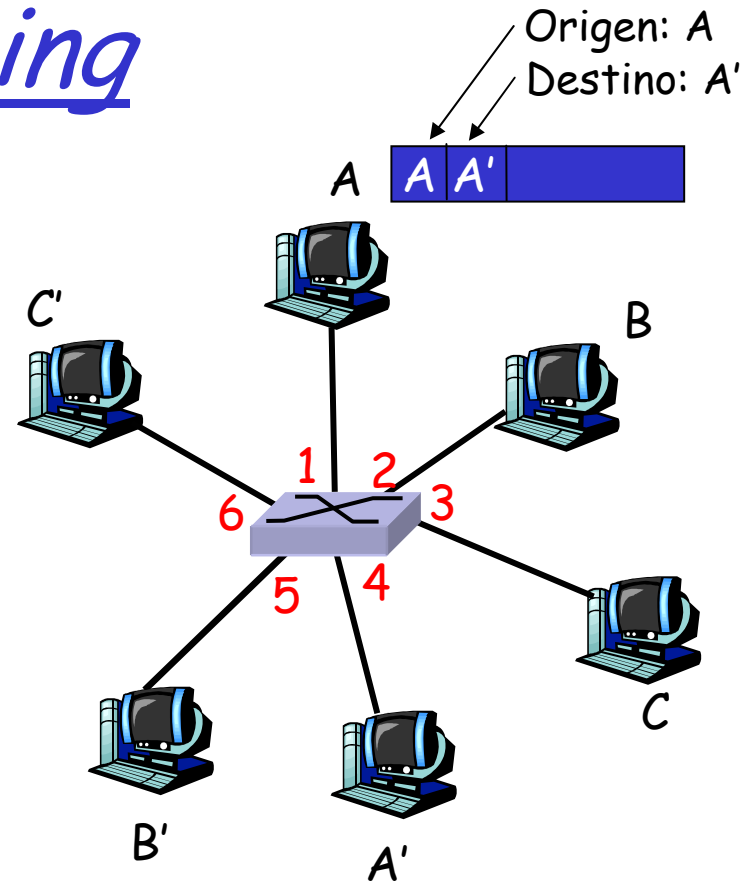
- **P:** ¿Cómo sabe un switch que A' es alcanzable a través de la interfaz 4 y B' a través de la interfaz 5?
- **R:** cada switch tiene una **tabla de switch**, con cada entrada:
 - (dirección MAC del host, interfaz por donde alcanzar al host, time stamp)
 - algo similar a una tabla de *routing*
- **P:** ¿Cómo las entradas son creadas y mantenidas en la tabla del switch?
 - ¿Parecido en algo a un protocolo de *routing*?



*switch con seis interfaces
(1,2,3,4,5,6)*

Switch: *self-learning*

- el switch *aprende* qué *hosts* puede ser alcanzado a través de qué interfaces
 - cuando una trama es recibida, el switch "aprende" la ubicación del emisor: el segmento LAN de entrada
 - registra el par emisor/ubicación en la tabla del switch



Dir. MAC	interfaz	TTL
A	1	60

*Tabla del switch
(inicialmente vacía)*

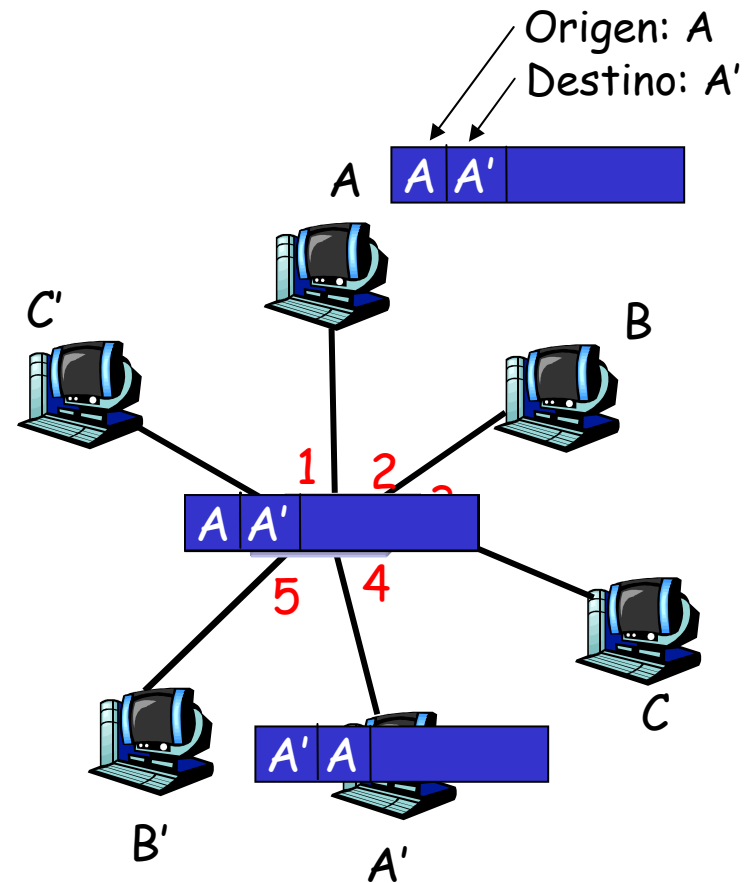
Switch: filtering/forwarding de tramas

Cuando una trama es recibida:

1. registra el link asociado con el host que envía
2. busca en la *switch table* utilizando la dirección MAC destino
3. **if** encuentra una entrada para el destino
then {
 if destino en segmento de donde arribó la trama
 then descartar la trama
 else forward de la trama en la interfaz indicada
}
else flood ← *forward en todas las interfaces menos en la que arribó*

Ejemplo: Self-learning, forwarding

- Destino de la trama desconocido: *flood*
- Ubicación del destino conocida: *envío selectivo*



Dir. MAC	interfaz	TTL
A	1	60
A'	4	60

*Tabla del switch
(inicialmente vacía)*

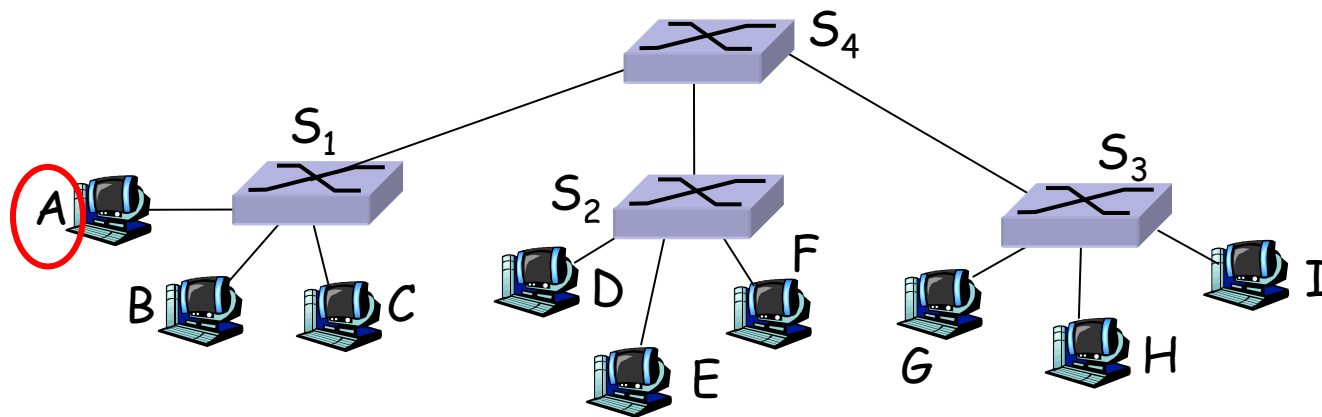
Evolución de las redes LAN

Ethernet

- Ha pasado de *half-duplex* a *full-duplex*
- "Han dejado de existir las colisiones"

Interconectando switches

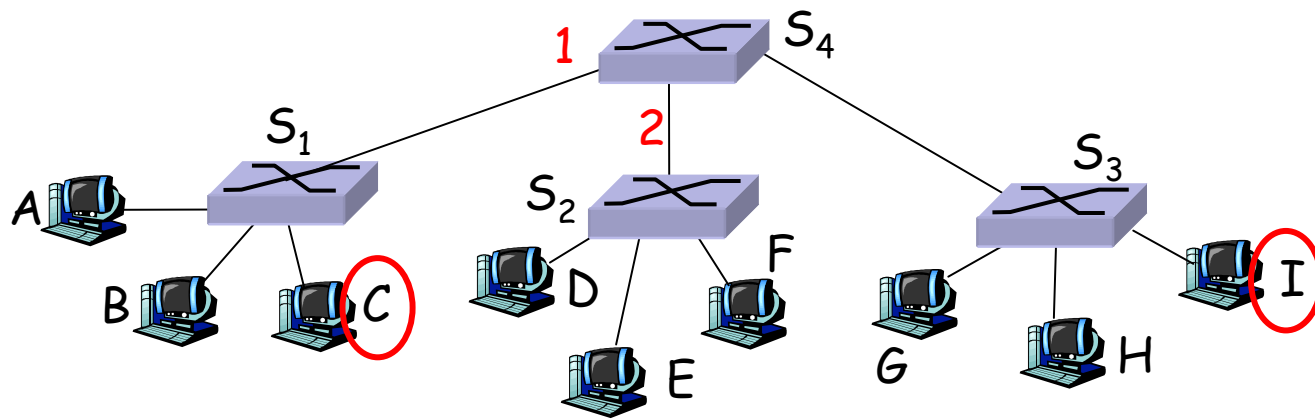
- Los switches se pueden conectar entre sí



- P: enviando de A a G - ¿de qué forma S₁ conoce cómo enviar la trama destinada a G a través de S₄ y S₃?
- R: *self-learning* (trabaja de la misma forma que en el caso de un solo *switch*)

Ejemplo: *self-learning multi-switch*

Suponga que C envía una trama a I; I responde a C



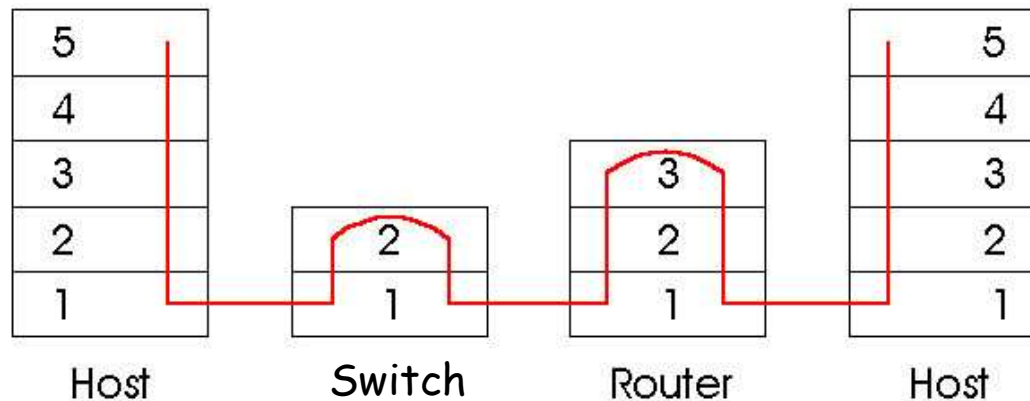
- P: muestre las tablas de los switches y el *forwarding* de las tramas en S₁, S₂, S₃, S₄

Técnicas de conmutación de tramas

- ❑ Técnicas utilizadas por los *switches* para pasar la trama desde el puerto de entrada hasta el puerto de salida
- ❑ Se decide en función de la DA
- ❑ Dos grandes familias
 - *Cut-through*
 - Sólo espera la *Destination Address*
 - No realiza FCS (Frame-Check-Sequence)
 - *Store & Forward*
 - Espera toda la trama
 - Realiza FCS

Switches vs. Routers

- ❑ ambos son dispositivos *store-and-forward*
 - routers: dispositivos de capa de red (examina encabezados de capa de red)
 - switches: dispositivos de capa de enlace
- ❑ los routers mantienen tablas de *routing*; implementan algoritmos de *routing*
- ❑ los switches mantienen tablas de switch, implementan filtrado, algoritmos de aprendizaje



Segmentando redes LAN...

- “Teoría de Darwin de las redes LAN” ☺ :
 - la evolución del **hub** al **switch**
 - existió un dispositivo intermedio que vivió poco: el **bridge**
- Hub
 - Capa Física
 - 1 dominio de colisión y 1 dominio de broadcast
- Bridge
 - Capa de Enlace de Datos
 - 1 dominio de colisión en cada puerta y 1 dominio de broadcast
- Switch
 - Capa de Enlace de Datos
 - 1 dominio de colisión en cada puerta y 1 dominio de broadcast
 - Pero además, mayor
 - cantidad de puertas que un bridge
 - capacidad de conmutación de tramas **que un bridge**

Eficiencia de CSMA/CD

- t_{prop} = retardo propagación entre 2 nodos en la LAN
- t_{trans} = tiempo para transmitir una trama

$$efficiency = \frac{1}{1 \pm 5t_{prop}/t_{trans}}$$

- Ejemplo
 - *veloc.: 10 Mbps; dist. entre nodos: 2,5 Km; retardo de prop.: 2×10^8 m/s; tamaño de trama: 72 bytes*
 - entonces, $t_{prop} = 12,5 \mu s$, $t_{trans} = 57,6 \mu s$
 - finalmente, *eficiencia = 0,48*
- mejor *performance* que ALOHA y además... simple, barato, descentralizado