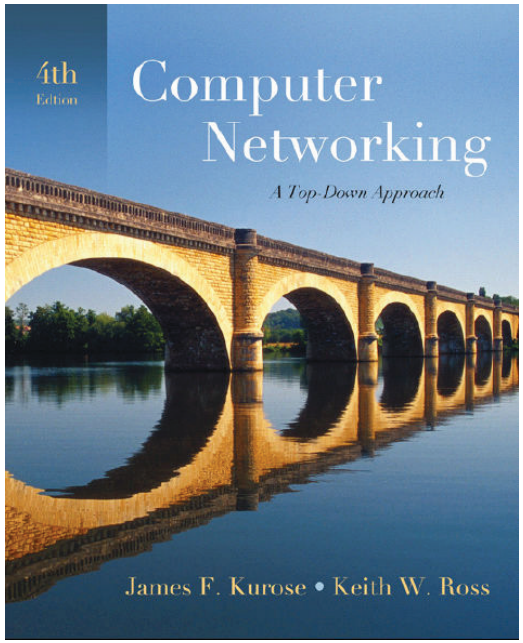
Introducción a las Redes de Computadores

Capítulo 5
Capa de Enlace y LANs

Nota acerca de las transparencias del curso:

Estas transparencias están basadas en el sitio web que acompaña el libro y han sido modificadas por los docentes del curso.



Computer Networking: A Top Down Approach 4th edition. Jim Kurose, Keith Ross Addison-Wesley, July 2007.

Capítulo 5: La Capa de Enlace de Datos

Objetivos:

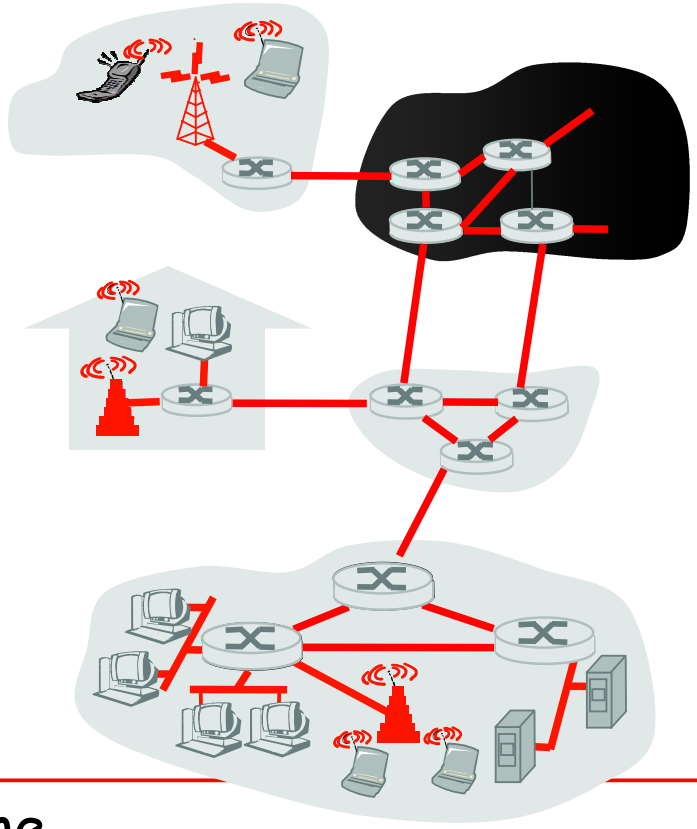
- Entender los principios detrás de los servicios de la capa de enlace de datos:
  - detección de errores; corrección
  - compartir un canal de *broadcast*: acceso múltiple
  - direccionamiento de capa de enlace
  - transferencia de datos confiable, control de flujo

- Algunas tecnologías de Capa de Enlace

Capa de Enlace: Introducción

Algo de terminología:

- hosts y routers son ***nodes***
- los canales de comunicación que conectan nodos adyacentes a través de caminos de comunicación son ***links***
  - enlaces cableados
  - enlaces inalámbricos
  - LANs
- la PDU de capa 2 es el ***frame***, que encapsula un datagrama



la capa de enlace de datos tiene la responsabilidad de transferir datagramas desde un nodo a otro nodo adyacente, a través de un *link*

Capa de enlace: contexto

- los datagramas son transferidos por diferentes protocolos de enlace sobre diferentes enlaces:
  - p.e., Ethernet en el primer enlace, Frame Relay en los enlaces intermedios, 802.11 en el último enlace
- cada protocolo de enlace brinda diferentes tipos de servicios
  - p.e., puede o no proveer **rdt** (*reliable data transfer*) sobre el enlace

Analogía transporte

- Viaje desde Montevideo a Mar del Plata
  - remise: Montevideo a Carrasco
  - avión: Carrasco a Aeroparque
  - ómnibus: Aeroparque a Mar del Plata
- turista = datagrama
- segmento de transporte = enlace de comunicación
- modo de transporte = protocolo de capa de enlace
- agencia de viaje = algoritmo de enrutamiento

Servicios de Capa de Enlace

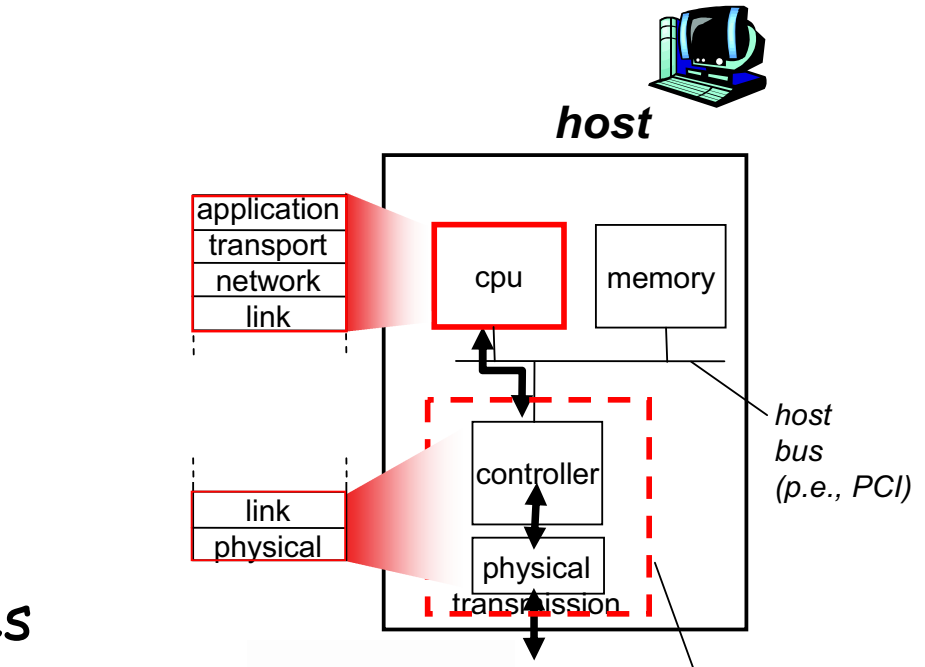
- *entramado (framing):*
  - encapsulado del datagrama en la trama, agregando encabezado (*header*) y cola (*trailer*)
- *acceso al enlace:*
  - acceso al canal si es un medio compartido (*Medium Access Control*)
  - direcciones "MAC" *addresses* utilizadas en los encabezados de las tramas para identificar el origen y el destino
    - distintas de las direcciones IP
- *entrega confiable:*
  - entre nodos adyacentes
  - ¡ya aprendimos cómo hacer ésto (teo Capa de Transp.)!
  - rara vez utilizados en enlaces de pocos errores (fibra óptica, algunos pares trenzados)
  - enlaces inalámbricos: alta tasa de error
    - **P**: ¿Por qué confiabilidad a nivel de enlace y *end-end*?

Servicios de Capa de Enlace (más)

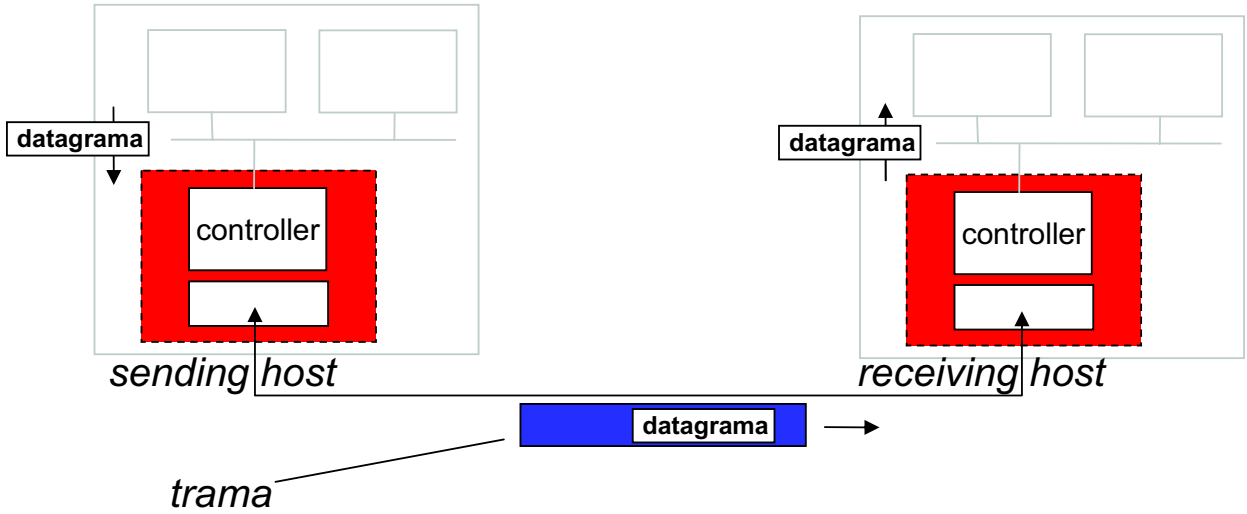
- *control de flujo:*
  - acuerdo entre los nodos emisor y receptor (aquí, adyacentes)
  - Recordar: *buffers* y capacidad de procesamiento
- *detección de errores:*
  - errores causados por atenuación de la señal, por ruido.
  - el receptor detecta presencia de errores:
    - señaliza al emisor para una retransmisión o descarta la trama
- *corrección de errores (FEC: Forward Error Correction):*
  - el receptor identifica *y corrige* el/los error/es en bit/s sin necesidad de retransmisión
- *half-duplex and full-duplex:*
  - con *half-duplex*, los nodos en los extremos del enlace pueden transmitir, pero no al mismo tiempo

¿Dónde está implementada la Capa de Enlace?

- En todos los *hosts*
- En el adaptador de red (*Network Interface Card*: NIC)
  - Tarjetas Ethernet, PCMCIA, 802.11
  - Implementa las capas de Enlace y Física (como mínimo)
- Incorporadas a los buses del sistema de los *hosts*
- combinación de *hardware, software, firmware*



Comunicación de adaptadores

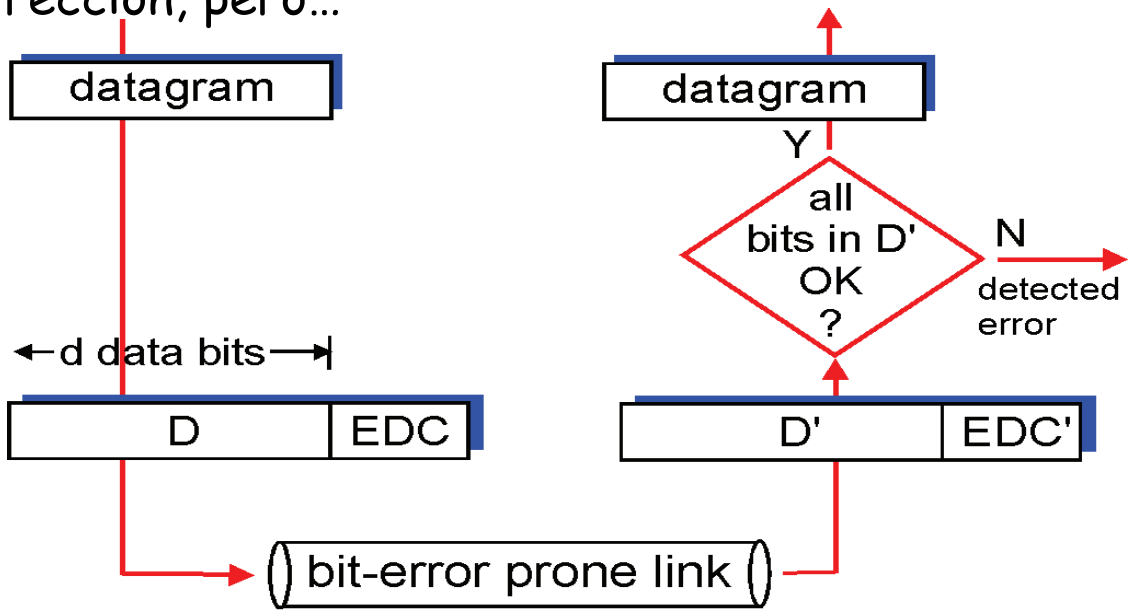


- lado emisor:
  - encapsula el datagrama en una trama
  - agrega bits de chequeo de error, rdt, control de flujo, etc.
- lado receptor:
  - busca por errores, rdt, control de flujo, etc
  - extrae el datagrama y lo pasa a las capas superiores

Detección de errores

EDC= Error Detection and Correction bits (**redundancia**)
D = Datos protegidos por chequeo de errores; puede incluir campos del encabezado
¡La detección de errores no es 100% confiable!

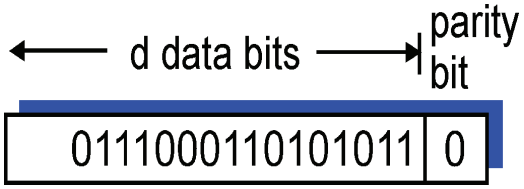
- el protocolo puede perder algunos errores
- un campo de EDC mayor proporciona mejor detección y corrección, pero…



Chequeo de paridad

Paridad de un bit:

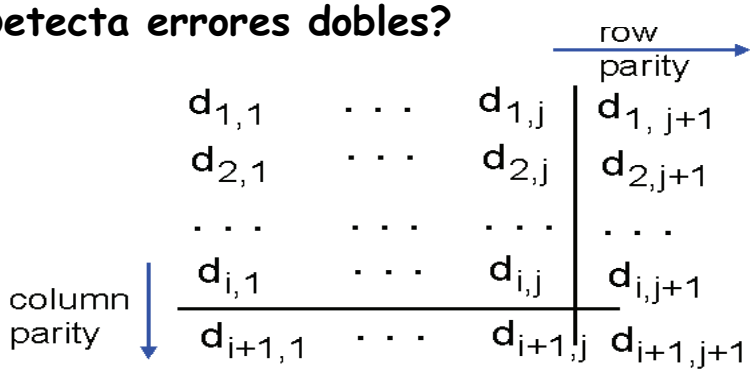
Detecta errores en 1 bit

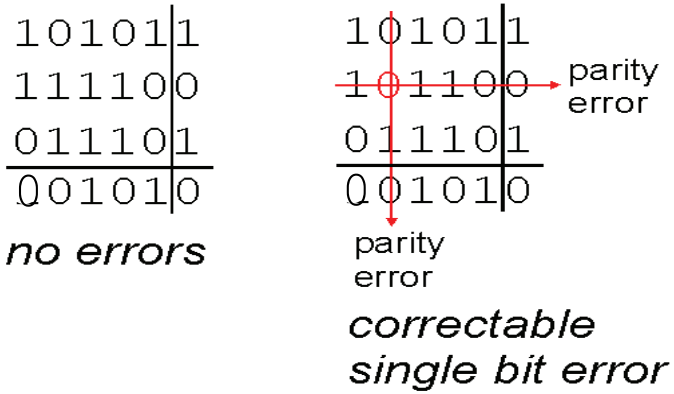


Paridad en dos dimensiones:

Detecta *y corrige* errores en 1 bit

¿Detecta errores dobles?





Internet checksum (suma de comprobación)

- Objetivo: detectar "errores" (bits cambiados) en el paquete transmitido (nota: generalmente utilizado en la capa de transporte)

- Recordar lo visto en Capa de Transporte

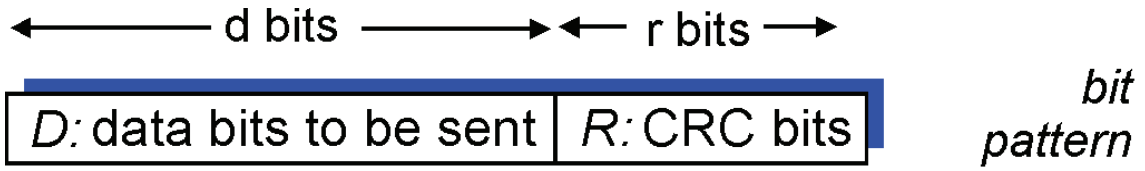
- En general es un método menos potente que el próximo que veremos

Cyclic Redundancy Check

- códigos CRC o códigos polinómicos
- ampliamente utilizado en la práctica (Ethernet, 802.11 WiFi, ATM)
- ver a los bits de datos, D, como los coeficientes de un polinomio
  - por ejemplo: 110001 es x^5+x^4+1
- Toda la aritmética que se utiliza es módulo 2 sin *carry* en las operaciones (sumas y restas equivalentes a XOR)
- elegimos un patrón de **r+1 bits** (polinomio **generador**), G, de grado **r**, que conocen el transmisor y el receptor

Cyclic Redundancy Check

- objetivo: **determinar r CRC bits, R**, tal que
  - <D,R> (concatenado) es divisible exactamente por G
    - D x 2^r es desplazar hacia la izquierda r bits y agregando 0s
    - D x 2^r + R es concatenarlos
  - el receptor divide <D,R> entre G. Si el resto es distinto de cero: ¡error detectado!



bit pattern

$D * 2^r$ XOR R

mathematical formula

Ejemplo CRC

- El emisor busca R, tal que exista Q que cumpla:

$D \cdot 2^r$ XOR R = Q . G

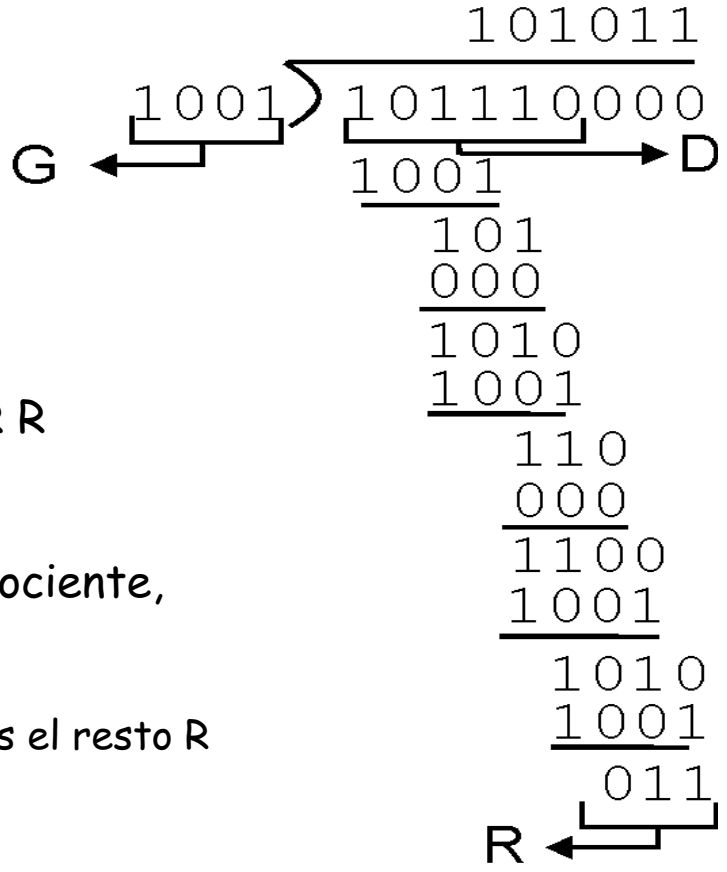
Que G divida a $D \cdot 2^r$ - R sin resto

$D \cdot 2^r$ XOR R= Q . G
$D \cdot 2^r$ XOR R XOR R = Q . G XOR R
$D \cdot 2^r = nG + R$

$D \cdot 2^r$: dividendo, G: divisor, Q: cociente, R: resto

- si dividimos $D \cdot 2^r$ por G, buscamos el resto R

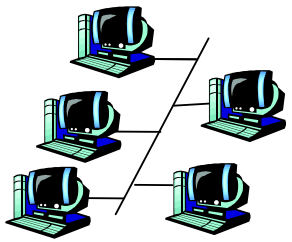
$$R = \text{resto} \left[\frac{D \cdot 2^r}{G}\right]$$



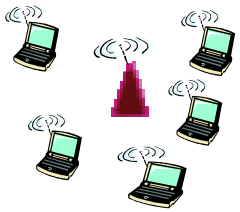
Protocolos y enlaces de acceso múltiple

Dos tipos de enlaces:

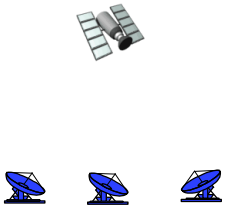
- punto a punto
  - PPP para acceso discado
  - Enlace punto a punto entre switch Ethernet y *host*
- *broadcast* (cable o medio compartido)
  - Ethernet "legacy"
  - HFC: *Hybrid Fiber Cable*
  - 802.11: LAN inalámbrica



cable compartido (p.e., cable Ethernet)



RF compartida (p.e., 802.11 WiFi)



RF compartido (satélite)

personas en una fiesta (aire compartido)

Protocolos de acceso múltiple

- Único canal *broadcast* compartido
- Dos o más transmisiones simultáneas: interferencia
  - Colisión
    - si un nodo recibe dos o más señales al mismo tiempo
    - simultaneidad en el tiempo y en la frecuencia de dos o más tramas en el mismo medio físico

Protocolo de Acceso Múltiple

- Algoritmo distribuido que determina cómo los nodos comparten el canal, y determina cuándo el nodo puede transmitir
- La comunicación acerca de compartir el canal debe utilizar el mismo canal
  - no canal *out-of-band* para coordinación

Protocolo de acceso múltiple **ideal**

Canal *Broadcast* con velocidad R bps

1. cuando un nodo quiere transmitir, lo hará a una velocidad R.
2. cuando M nodos quieren transmitir, cada uno enviará a una velocidad promedio de R/M
3. completamente descentralizado:
   - no hay un nodo especial para coordinar las transmisiones
   - no hay sincronización de relojes, *slots*
4. simple

Protocolos MAC: taxonomía

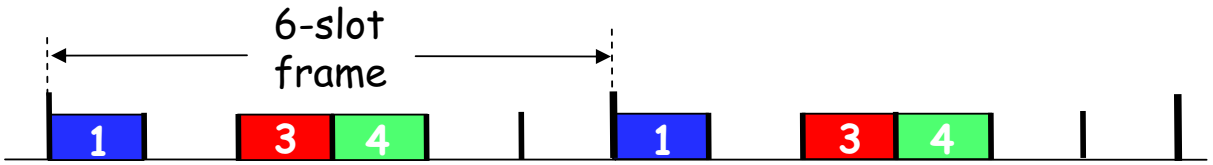
Tres grandes clases:

- Particionado del canal
  - Protocolos de arbitraje
  - divide el canal en pequeñas "piezas" (ranuras de tiempo, frequencia, código)
  - asigna una pieza a un nodo para su uso exclusivo
  - estrategia estática
  - equitativo
- Acceso Randómico
  - el canal no se divide, permite colisiones
  - "recuperación" de colisiones
  - estrategia dinámica
- "Toma de turnos"
  - Los nodos toman turnos, pero los nodos con más tramas para enviar podrían tomar turnos más largos
  - estrategia dinámica
  - estrategias de reserva o centralizada

Protocolos MAC de **particionado del canal**: TDMA

TDMA: *Time Division Multiple Access*

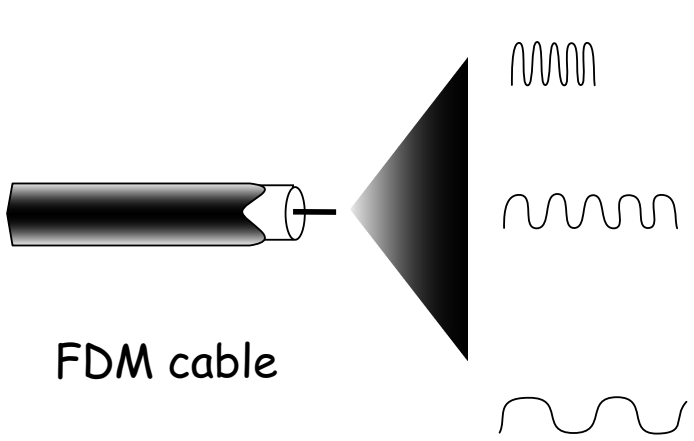
- acceso al canal rotativo
- cada estación tiene un *slot* de longitud fija (longitud = tiempo de transm. de la trama) en cada vuelta
- los *slots* sin usar quedan libres
- ejemplo: LAN con 6 estaciones, 1,3 y 4 tiene trama; los *slots* 2,5 y 6 quedan libres

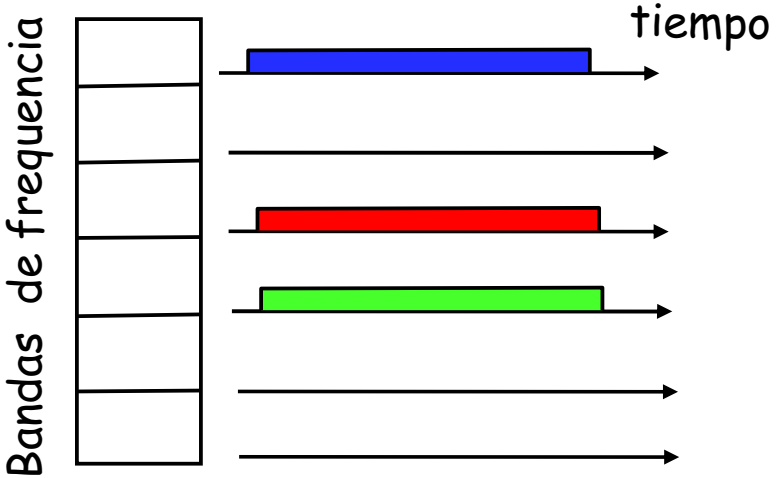


Protocolos MAC de **particionado del canal**: FDMA

FDMA: *Frequency Division Multiple Access*

- el espectro del canal se divide en bandas de frecuencia
- a cada estación se le asigna una banda de frecuencia fija
- el tiempo de transmisión no utilizado en las bandas de frecuencia queda libre
- ejemplo: LAN con 6 estaciones, 1,3 y 4 tienen trama; las bandas de frequencia 2,5 y 6 están libres





Protocolos de **acceso randómico**

- cuando un nodo tiene un paquete para enviar
  - transmite a la velocidad total del canal, R
  - no existe *a priori* coordinación entre nodos
- dos o más nodos transmitiendo "colisión"
- protocolos MAC de acceso randómico especifican:
  - cómo detectar colisiones (directa o indirecta)
  - cómo recuperarse de las colisiones (p.e., a través de re-transmisiones retrasadas)
- ejemplos de protocolos MAC de acceso randómico:
  - ALOHA ranurado, ALOHA
  - CSMA, CSMA/CD, CSMA/CA
  - También se les conoce como sistemas de contención o sistemas de contienda

CSMA (*Carrier Sense Multiple Access*)

CSMA: escuchar antes de transmitir

- Si el canal está libre: transmitir la trama entera
- Si el canal está ocupado: diferir la transmisión
  - volver a escuchar después de un tiempo
  - seguir escuchando hasta que quede libre y transmitir
  - seguir escuchando hasta que quede libre y transmitir con probabilidad **p**
- Analogía humana: ¡no interrumpir a los otros!

CSMA/CD (*Collision Detection*)

- CSMA/CD: si hay presencia de portadora, se difiere la transmisión, como en CSMA
  - las transmisiones que colisionan son abortadas, reduciendo el desperdicio de canal
  - colisión = desperdicio del canal
- detección de colisión:
  - relativamente fácil en LANs cableadas
  - dificil en LANs inalámbricas

Protocolos MAC "Toma de turnos"

protocolos MAC de particionado del canal:

- compartir el canal *justa* y *eficiente* a alta carga
- ineficiente a baja carga: retardo en el acceso al canal, ancho de banda 1/N asignado aún si hay un sólo nodo activo

protocolos MAC de acceso randómico

- eficiente a baja carga: un único nodo puede utilizar completamente el canal
- alta carga: *overhead* por colisión

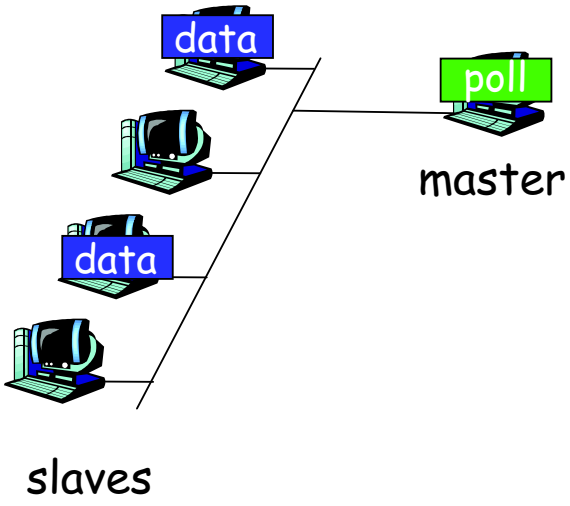
protocolos de "toma de turnos"

busca lo mejor de los dos mundos

Protocolos MAC "Tomando turnos"

Polling:

- el nodo *master* "invita" a los nodos *slaves* a transmitir en turnos
- típicamente utilizado con dispositivos *slaves* "tontos"
- sin colisiones
- determinístico
- involucra:
  - *overhead* por *polling*
  - latencia
  - único punto de falla (master)
- ejemplo
  - Bluetooth
    - IEEE 802.15
  - Un modo de operación de 802.11 (Wi Fi)



Resumen de protocolos MAC

- *particionado de canal,* en tiempo, frequencia
  - división en el tiempo, división en la frecuencia
- *acceso randómico* (dinámico),
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - Escucha de portadora: fácil en algunas tecnologías (cableadas), dificil en otras (inalámbricas)
  - CSMA/CD utilizado en Ethernet
  - CSMA/CA (*Colission Avoidance*) utilizado en 802.11
- *toma de turnos*
  - *polling* desde un sitio central, pasaje de *token*
  - Bluetooth, Token Ring

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

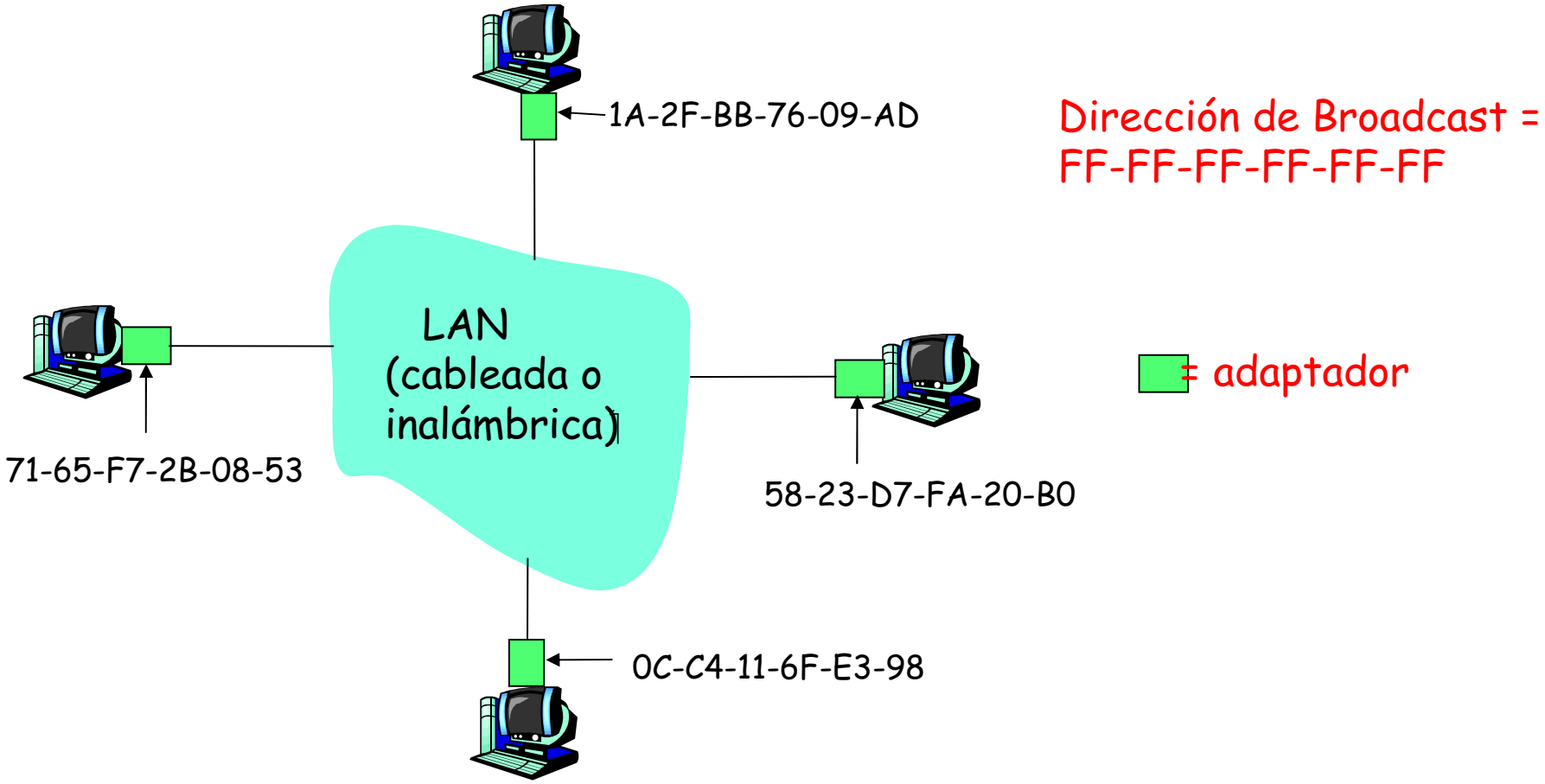
- Dirección MAC (o LAN o física o hardware o del adaptador o "Ethernet"):
  - función: *llevar la trama de una interfaz a otra interfaz físicamente conectada (misma red)*
  - Direcciones MAC de 48 bits (en la mayoría de las redes LAN)
    - grabada en la ROM de la NIC; en algunos casos (cada vez más)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

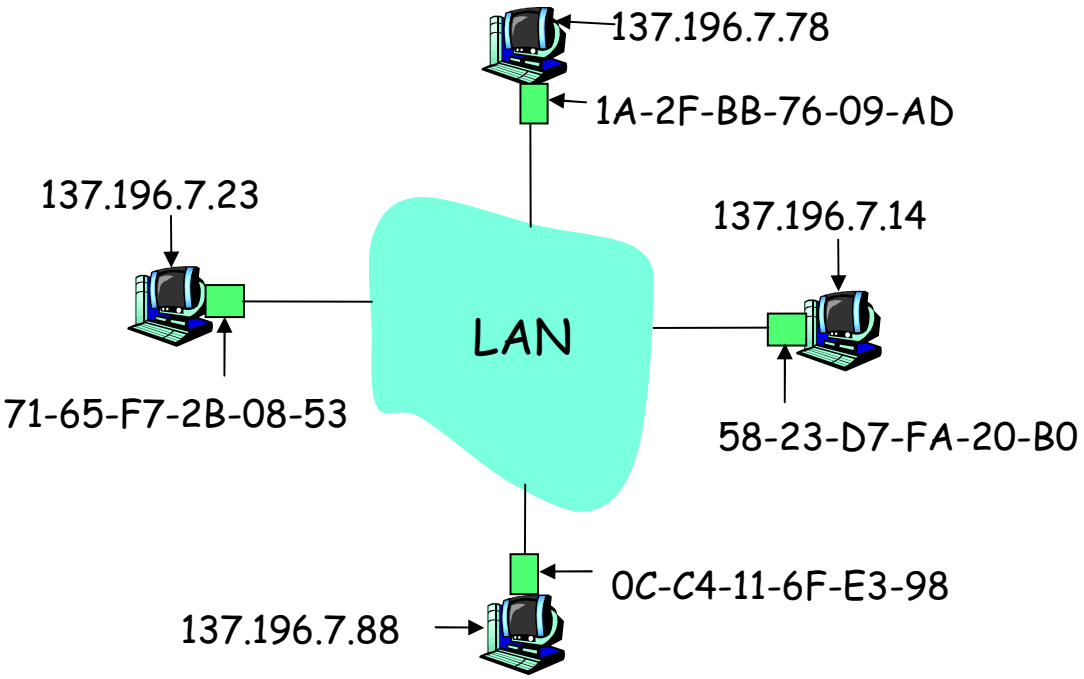
Direcciones MAC

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

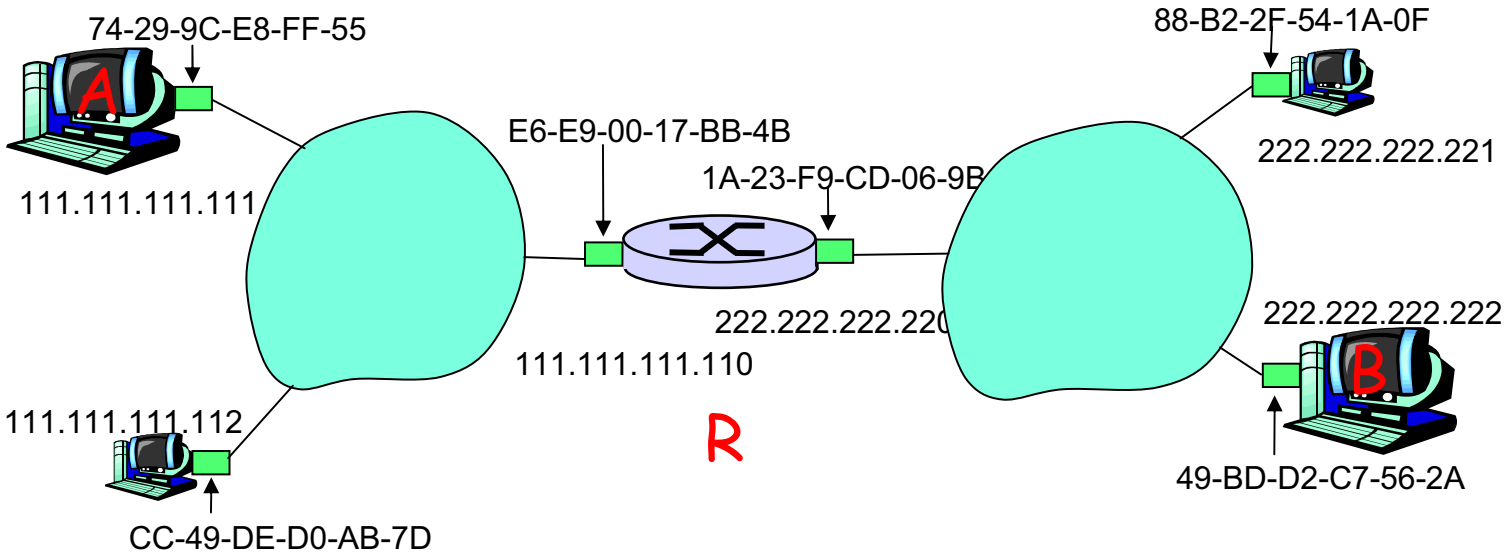
  < dirección IP; dirección MAC; TTL>

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

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)

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

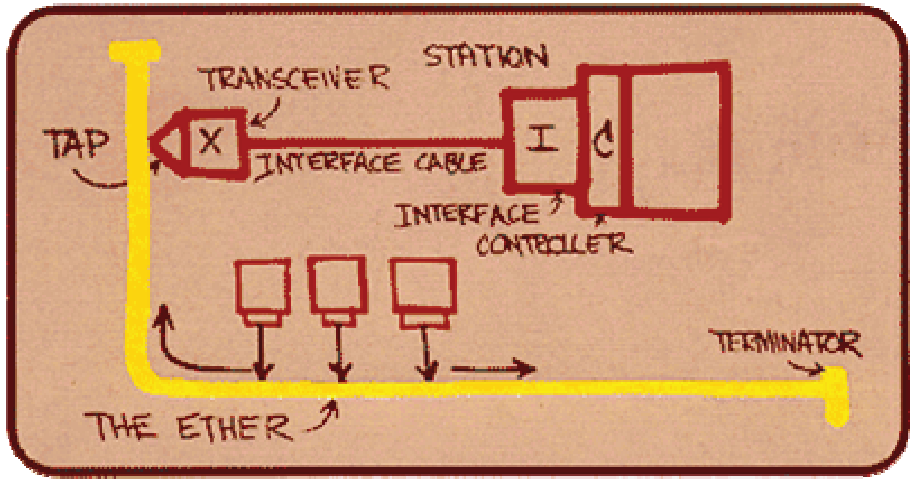
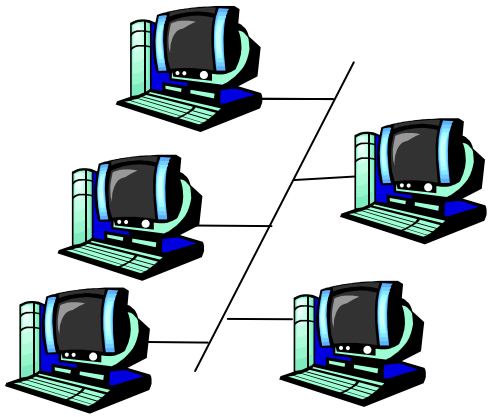


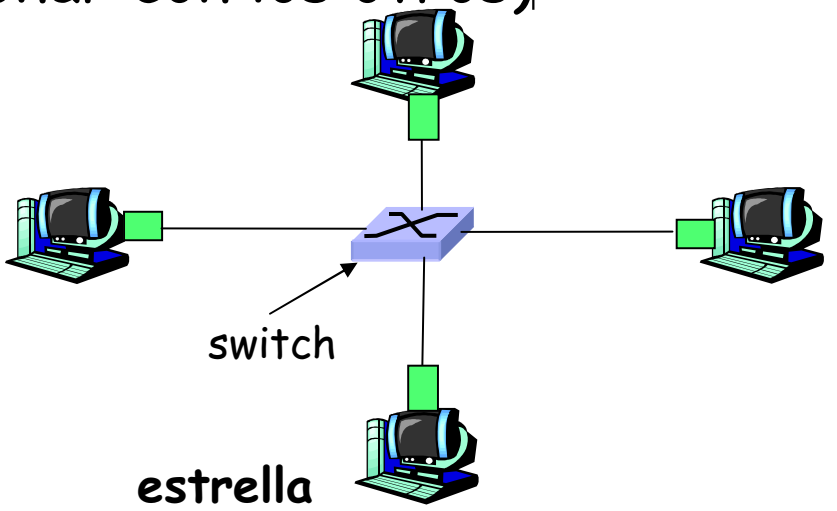
Diagrama de Ethernet
de Robert Metcalfe

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)



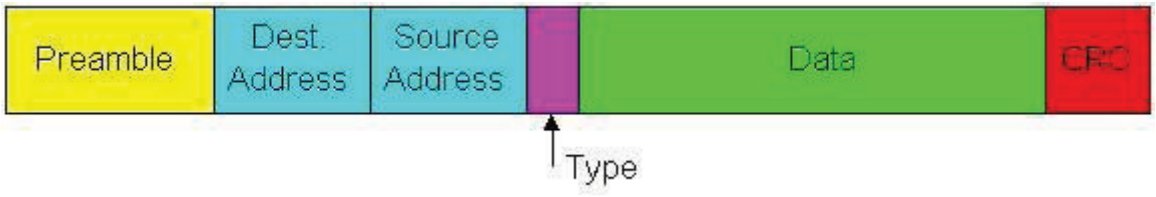
bus: cable coaxial



estrella

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

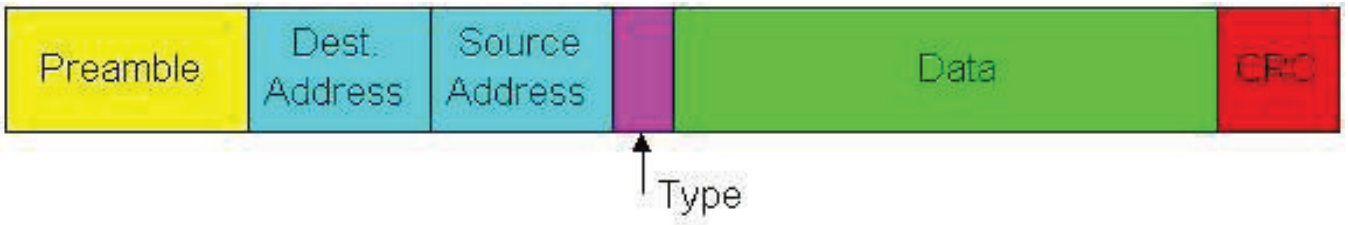


Preamble:

- 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)

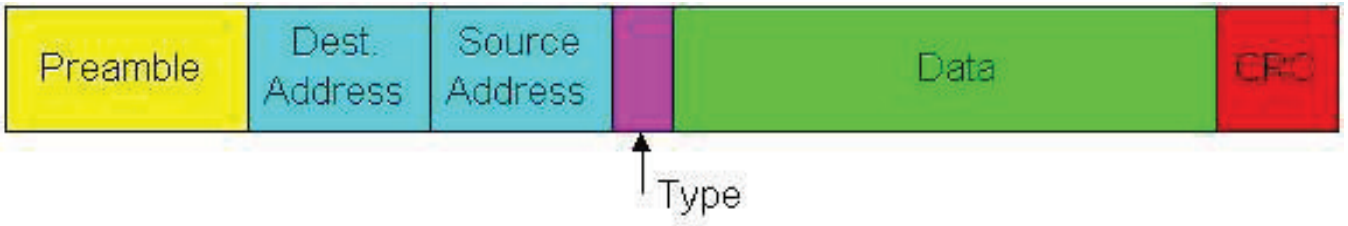
- Direcciones: 6 bytes cada una
  - si el adaptador recibe una trama con dirección destino la suya o la dirección de *broadcast*, (ej. paquete ARP), pasa los datos en la trama al protocolo de capa de red
  - en otro caso, el adaptador descarta la trama
- Type: 2 bytes
  - multiplexación
  - indica el 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*"



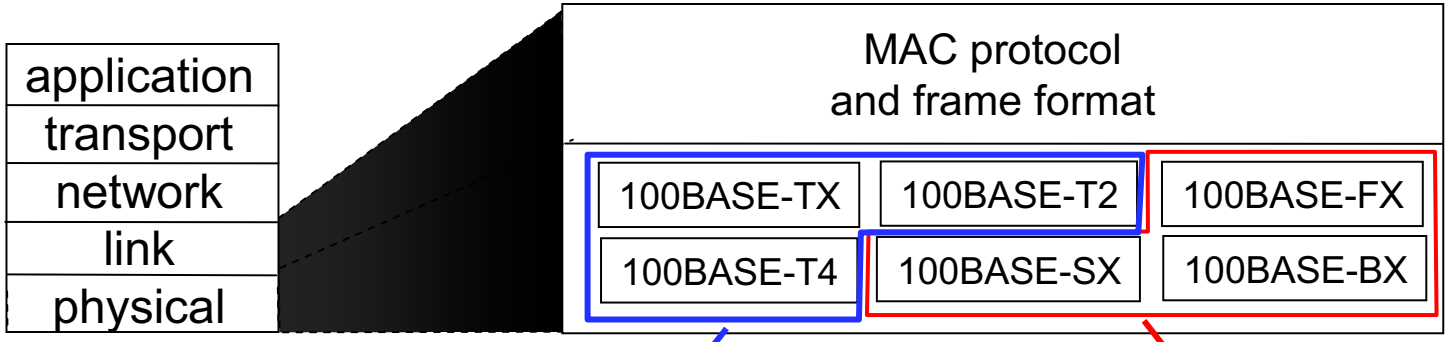
Ethernet: servicio no confiable, no orientado a conexión

- 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

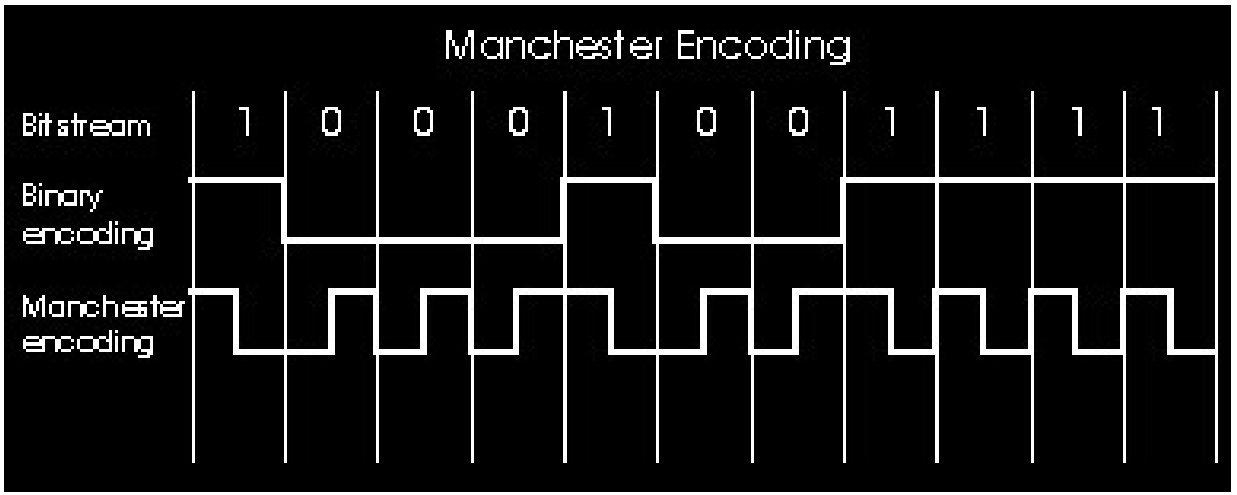
- Protocolo MAC de Ethernet: CSMA/CD
- La detección de colisiones es un servicio de Capa Física

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



Codificación Manchester

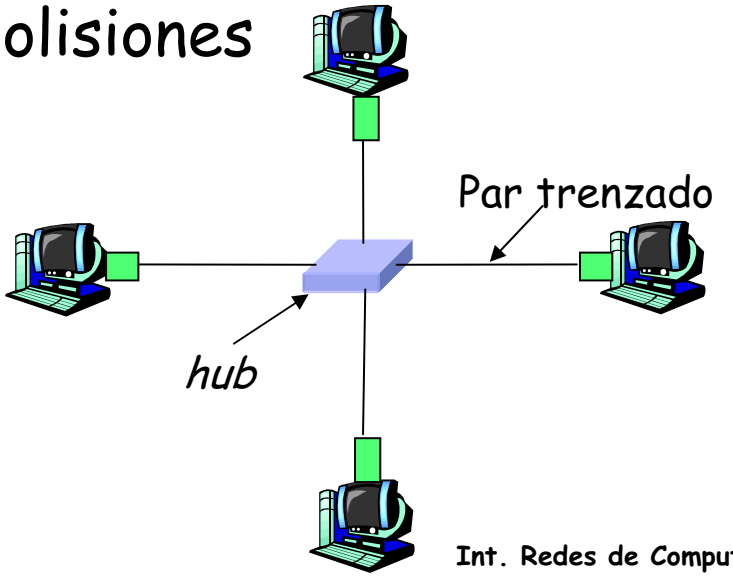


- Utilizado en 10BaseT
- Cada bit tiene una transición
- Permite que los relojes de los nodos emisores y receptores siempre estén sincronizados entre sí
  - No se requiere un reloj centralizado, global

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

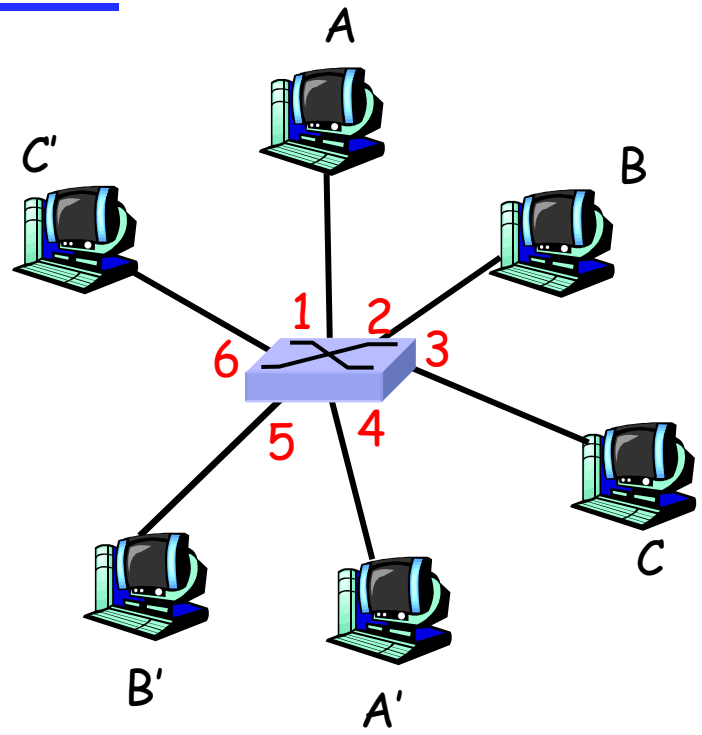


Switch

- 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

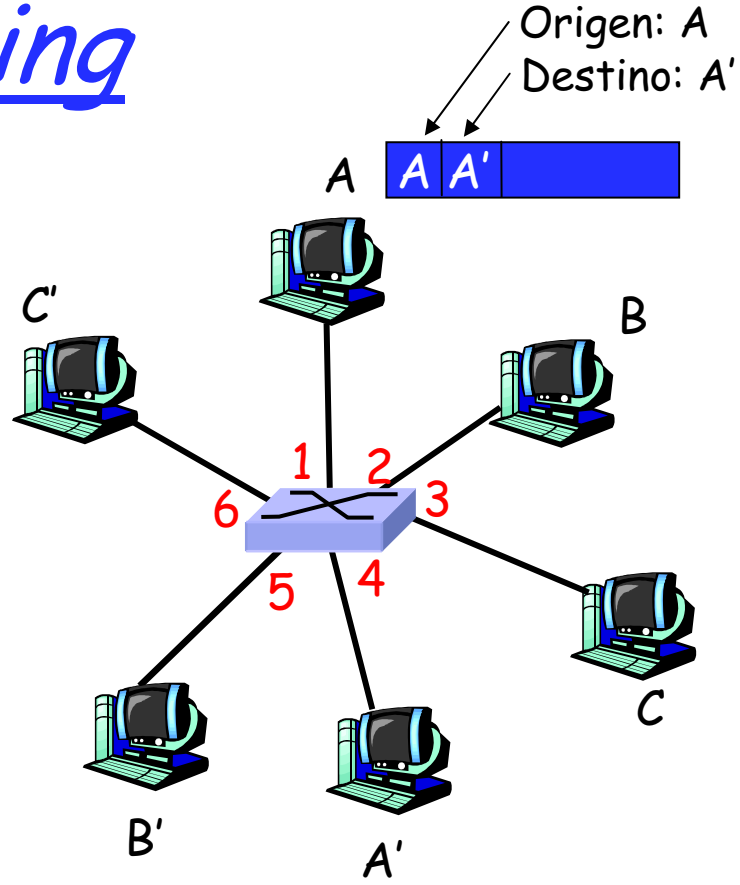
- Los *hosts* tienen conexiones dedicadas, directas al *switch*
- Los *switches* hacen *buffer* de las tramas
- El protocolo Ethernet es utilizado en *cada* link entrante, pero no hay colisiones; *full duplex*
  - cada *link* es su propio dominio de colisión
- *switching:* A-to-A' and B-to-B' simultáneamente,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



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

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:

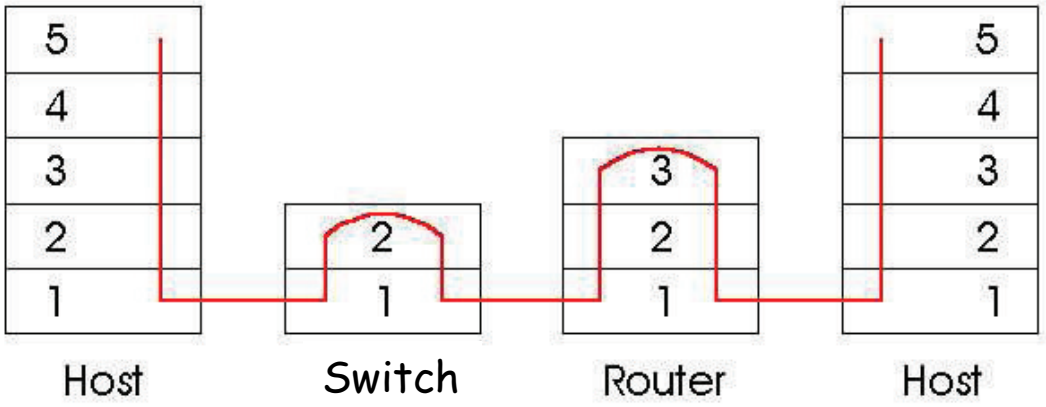
1. registra el link asociado con el host que envía
2. busca en la *switch table* utilizando la dirección MAC destino
3. **if** encuentra una entrada para el destino
   then {
   if destino en segmento de donde arribó la trama
   then descartar la trama
   else forward de la trama en la interfaz indicada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

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

Red "switcheada"

- Redundancia
  - Confiabilidad, disponibilidad
  - Costos
  - Pero quizás también, inestabilidad
    - Por ejemplo, un simple ***ARP request*** puede generar una tormenta de *broadcast* y afectar la *performance* de los switches de toda la red
    - Algo similar puede ocasionar un *unicast*
    - Precisamos una solución que evite los *loops* pero sin perder las bondades de la redundancia
  - En capa de enlace no existe el concepto de TTL
- *Spanning-Tree Protocol* (STP): Protocolo de gestión de capa de enlace que pone a disposición la redundancia de caminos pero previene de posibles *loops* en la red de *switches* (posible origen de duplicación de mensajes)

Protocolo *Spanning-Tree* (STP)

- El objetivo es que en cada instante exista un solo camino activo entre dos *switches*
  - Que existan *loops* físicos pero no lógicos
- Se define un árbol a través del cual se alcanza a todos los *switches* pero el árbol se "poda" de tal forma que algunos puertos quedan bloqueados a la espera de algún cambio topológico y los restantes puertos están en estado *forwarding*
- Algunos comentarios
  - Protocolo transparente a los usuarios
  - Radia Perlman -> IEEE 802.1D
  - "Protocolo de árbol de expansión"
  - Referencias en la bibliografía
    - Secciones 4.4 o 4.7 "del Tanenbaum"
    - Sección 5.6 "del Kurose & Ross"

VLAN: *Virtual* LAN

- Empresa con ***k*** departamentos
  - 1 red LAN por departamento
    - Agrupar lógicamente usuarios de la red y recursos conectados a puertos definidos administrativamente
    - Broadcast
    - Seguridad
    - Carga
- En los 90's: ***k*** redes LAN independientes significaba instalar ***k*** hubs (como mínimo)
- Luego, se incorporaron los *switches*
- Ahora: ***k*** redes LAN, técnicamente puede significar simplemente instalar **1** *switch*

VLAN: *Virtual LAN* (más)

- IEEE 802.1Q
- Permite crear "*switches* virtuales" en uno o más *switches* y de esa forma separar dominios de *broadcast* (más pequeños)
- Se debe definir:
  - Cantidad
  - Nombre de cada una ("color")
  - Miembros de cada una
- En cada puerto del *switch*, una sóla VLAN posible, salvo en los *trunks*

Enlace de Datos Punto a Punto

- un emisor, un receptor, un enlace: más fácil que un enlace *broadcast*:
  - no se requiere *Medium Access Control*
  - no se necesita direccionamiento MAC explícito
  - p.e., enlace discado
- protocolos *point-to-point* más populares:
  - PPP: *Point-to-Point Protocol*
  - HDLC: *High level Data Link Control*

PPP (RFC 1547, 1661, 1962, 2153)

- Requerimientos de diseño de PPP: RFC 1547
  - simple
  - entramado de paquete: encapsulado del datagrama de capa de red en una trama de capa de enlace
  - transparencia: debe poder llevar cualquier patrón de bit en el campo de datos (incluso los vinculados al *framing*)
  - multiplexación: porta datos de capa de red de cualquier protocolo (no solamente IP) al mismo tiempo
    - posibilidad de demultiplexar
  - detección de error (no corrección)
  - estado de la conexión: detectar y señalizar a la capa de red sobre falla en el *link*
  - negociación de la dirección de la capa de red: un *endpoint* puede configurar la dirección de red del otro
  - posibilidad de negociación de opciones
  - posibilidad de compresión de datos

No requerimientos de PPP

- corrección/recuperación de errores
- control de flujo
- entrega de tramas en orden (secuenciamiento)
- no hay necesidad de soporte de enlaces multipunto (p.e., *polling*)

Recuperación de errores, control de flujo, re-ordenamiento de datos
son relegados a las capas superiores