

Módulo 05

La Capa de Enlace

(Pt. 1)



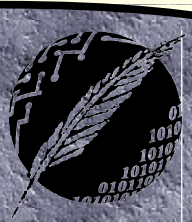
Redes de Computadoras
Depto. de Cs. e Ing. de la Comp.
Universidad Nacional del Sur



Copyright

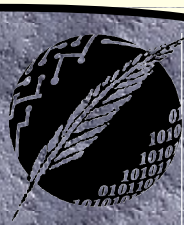
- Copyright © **2010-2021** A. G. Stankevicius
- Se asegura la libertad para copiar, distribuir y modificar este documento de acuerdo a los términos de la **GNU** Free Documentation License, versión 1.2 o cualquiera posterior publicada por la Free Software Foundation, sin secciones invariantes ni textos de cubierta delantera o trasera
- Una copia de esta licencia está siempre disponible en la página <http://www.gnu.org/copyleft/fdl.html>
- La versión transparente de este documento puede ser obtenida de la siguiente dirección:

<http://cs.uns.edu.ar/~ags/teaching>

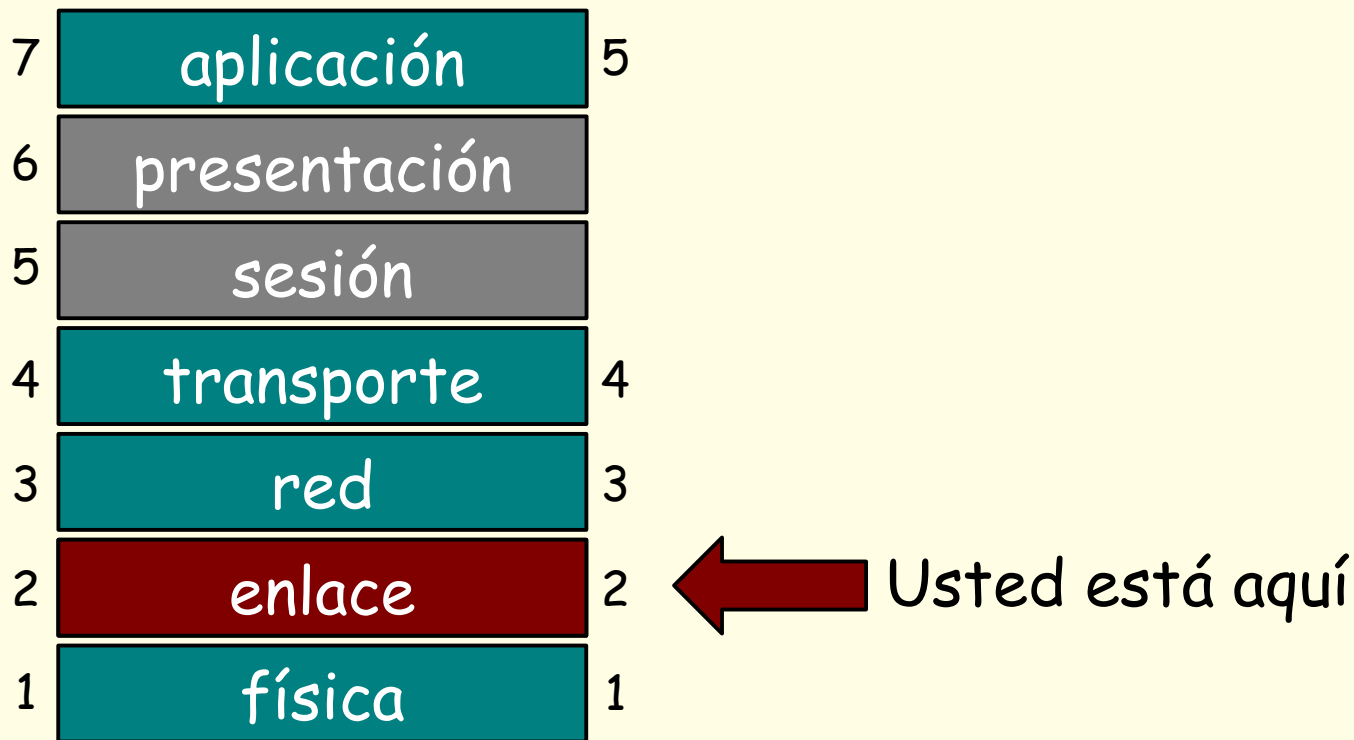


Contenidos

- Servicios provistos por la capa de enlace
- Protocolos de acceso múltiple
- Direcciones de red local y protocolo **ARP**
- Ethernet
- Hubs, bridges y switches
- Enlaces inalámbricos
- Virtualización de enlaces
- Datacenters

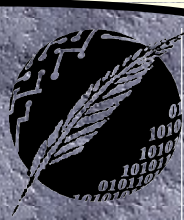


ISO/OSI - TCP/IP



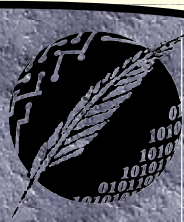
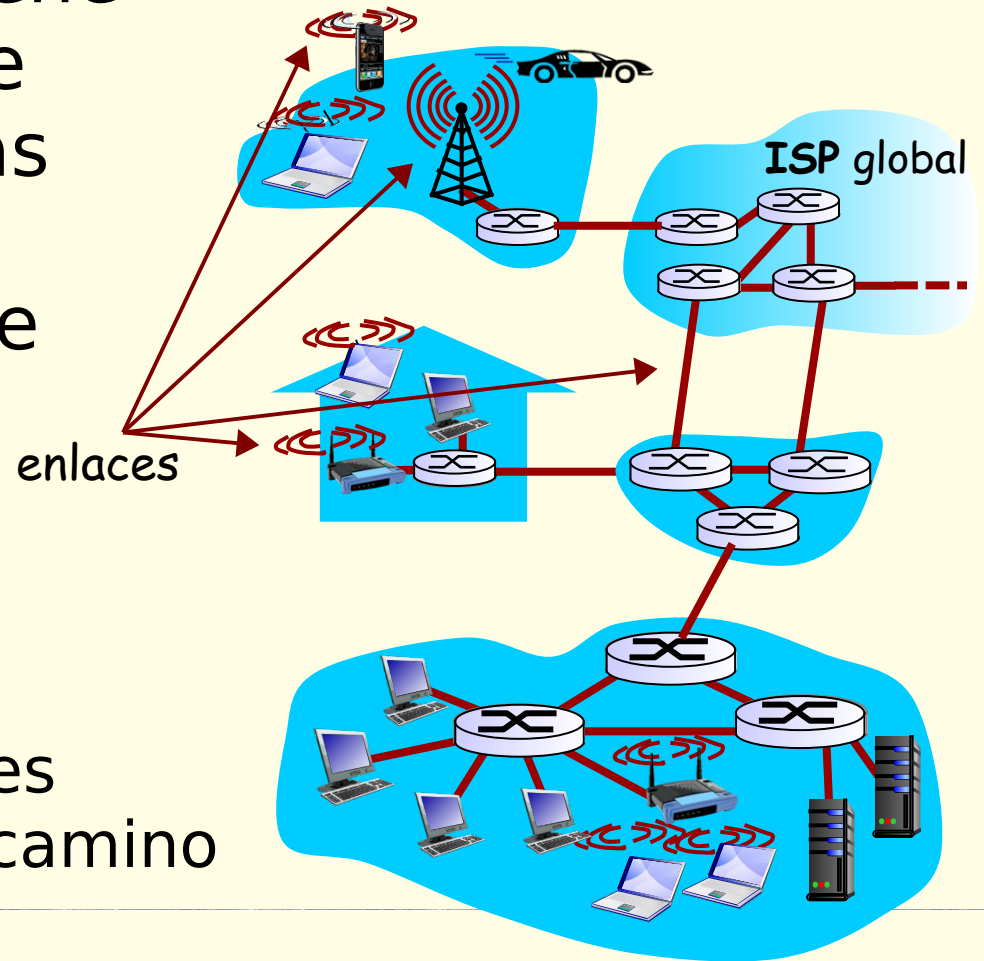
Un poco de terminología

- A nivel de capa de enlace no es necesario distinguir entre computadoras y routers
 - ➔ Denominaremos genéricamente **nodo** a cualquier dispositivo de la red (incluso bridges y switches)
- El canal de comunicación conectando un conjunto de nodos se denomina **enlace**
 - ➔ Enlaces cableados
 - ➔ Enlaces inalámbricos
- El **PDU** (Protocol Data Unit) de esta capa se denomina usualmente **trama**



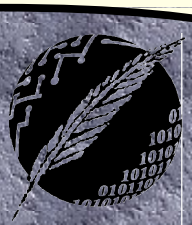
Capa de enlace

- La capa de enlace tiene la responsabilidad de transferir datagramas de un nodo a otro a través de un enlace
- ➔ Los datagramas son encapsulados dentro de las tramas
- ➔ Una multiplicidad heterogénea de enlaces conforman un mismo camino



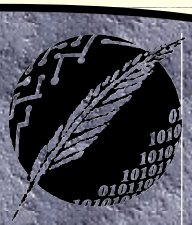
Protocolo de enlace

- Los datagramas son transferidos usando los distintos **protocolos de enlace** que figuran a lo largo de un camino de origen a destino:
 - ➔ Por caso, Ethernet en el primer enlace, frame relay en el enlace intermedio y **IEEE 802.11** en el último enlace
- Cada protocolo de enlace brinda un conjunto particular de funcionalidades
 - ➔ Por ejemplo, algunos (no todos) aseguran la **transferencia segura de datos**



Otra car analogy más...

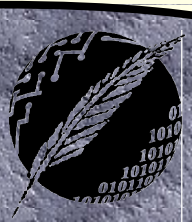
- Los autores consideran que en este punto corresponde apelar a otra car analogy
- Se desea viajar a la ciudad de Ushuaia:
 - Primero viajamos a la ciudad de buenos aires, por caso usando algún ómnibus o quizás en auto
 - Luego, tomamos el avión desde CABA a Ushuaia
- Evidentemente los servicios provistos en los distintos tramos (enlaces) dependerán del medio de locomoción (protocolo de enlace)



Servicios de la capa

● Entramado y acceso físico al enlace:

- Los datagramas se encapsulan dentro de tramas, las que se delimitan mediante **un encabezado y un terminador**
- Si el medio físico del enlace es compartido por varios nodos, también **se arbitra el acceso al mismo**
- Se usan **direcciones físicas en el encabezado** de las tramas para identificar el origen y el destino
- ¡No confundir la dirección física con la dirección **IP**!



Servicios de la capa

● Transmisión confiable de datos:

- ➔ Los mecanismos utilizados en la capa de enlace son exactamente los mismo que repasamos al abordar la capa de transporte
- ➔ Este servicio **no es requerido en los enlaces de alta calidad** que presentan muy baja probabilidad de error (por caso, fibra óptica, algunos tipos de par trenzado)
- ➔ En cambio, **en los enlaces inalámbricos se registran altas tasas de error** en la transmisión de datos
- ➔ ¿Tendrá sentido replicar esta funcionalidad en capa 2?



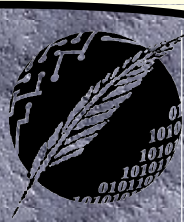
Servicios de la capa

● Control de flujo:

- Gestión del **ritmo de envío** entre nodos adyacentes al enviar y recibir información

● Detección de errores:

- Los errores se producen por la atenuación de las señales, el ruido de fondo, etc.
- El receptor es el encargado de **detectar que se produjo un error**
- Informa al emisor que debe **retransmitir o descartar** la trama en cuestión



Servicios de la capa

● Corrección de errores:

- ➔ En ocasiones el receptor puede no sólo detectar que se produjo un error sino que además puede contar con la suficiente información como para **corregir los bits en error** sin depender de una retransmisión de la trama en error

● Comunicación unidireccional y bidireccional:

- ➔ En los enlaces unidireccionales ambos extremos pueden emitir datos siempre y cuando no lo hagan al mismo tiempo



¿Dónde se implementa?

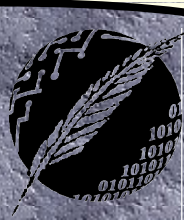
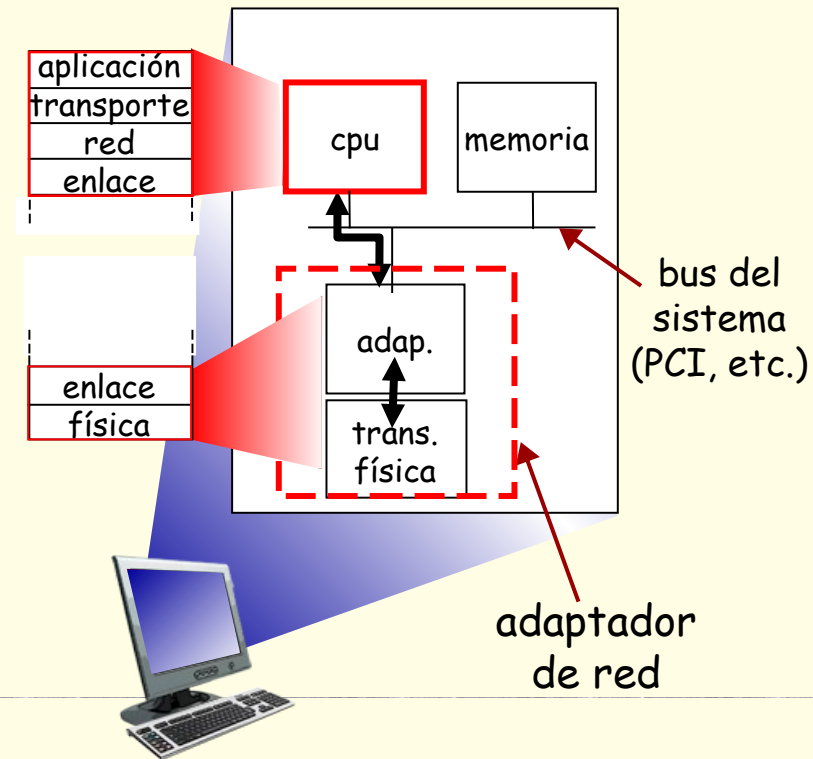
- La capa de enlace se implementa en absolutamente todos los dispositivos

- ➔ Cada dispositivo cuenta con uno o más adaptadores de red

- ➔ El adaptador implementa tanto la capa de enlace como la física

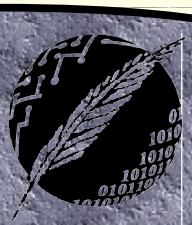
- ➔ También se conectan al bus del sistema

- ➔ Combina hardware, software e incluso firmware



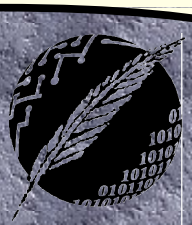
Emisor y receptor

- Emisor y receptor interactúan en sincronía a fin de proveer los servicios de la capa de enlace
- El emisor debe:
 - ➔ Encapsular los datagramas en las tramas
 - ➔ Agregar los bits necesarios para el control de errores, control de flujo, transmisión confiable de datos, etc.
- El receptor debe:
 - ➔ Verificar los bits de error, de control de flujo, etc.
 - ➔ Extraer el datagrama y entregarlo a la capa de arriba



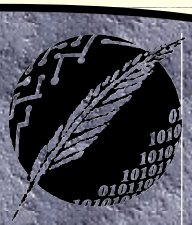
Detección de errores

- La **detección de errores** consiste esencialmente en reconocer si un patrón de bits guarda su forma original o bien ha sido alterado
 - ➔ A tal efecto se incorporan al dato un conjunto de bits redundantes denominados **bits de código**
 - ➔ Si se disponen de suficiente cantidad de bits de código es posible intentar no sólo detectar el error sino también conocer dónde se produjo a fin de poder corregirlo, evitando la retransmisión
- La detección **no es infalible**



Detección y corrección

- Se han ensayada un conjunto bastante diverso de códigos detectores y correctores de errores:
 - ➔ Paridad: sólo detecta errores simples
 - ➔ Hamming básico: detecta y corrige errores simples, o bien sólo detecta errores simples y dobles
 - ➔ Hamming extendido: detecta hasta errores dobles y corrige simples, o bien detecta hasta errores triples
 - ➔ VRC: detecta errores en burst hasta un límite
 - ➔ CRC: detecta errores en burst de una longitud menor o igual al grado del polinomio generador



Distancia mínima

- Se denomina **distancia mínima** a la cantidad de bits que se deben modificar en un patrón denotando un código válido para obtener otro código válido
- La distancia mínima de un código se relaciona directamente con su **capacidad para detectar y para corregir errores**
 - ➔ Si **D** es la distancia mínima de un código, **d** su capacidad de detectar errores y **c** la de corregir, se verifica que **$D > d + c$** y que **$d \geq c$**



Tipos de enlaces

- Enlaces tipo **punto-a-punto**:

- El protocolo **PPP** de acceso a internet por teléfono convencional
- Los enlaces punto-a-punto de la mayoría de las configuraciones ethernet de alta velocidad

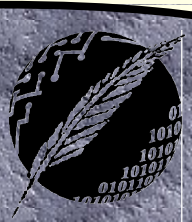
- Enlaces tipo **broadcast**:

- Ethernet histórico
- El protocolo **HFC** de acceso a internet por cable (**TV**)
- El protocolo **IEEE 802.11** de lo que llamamos **WiFi**



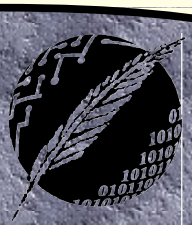
Protocolo de acceso múltiple

- El protocolo de acceso múltiple se utiliza para poder hacer uso de los canales compartidos propios de los enlaces de tipo broadcast
 - En este tipo de canal, si dos o más nodos transmiten en simultáneo se produce una interferencia
 - Ante una interferencia, todo lo que se está transmitiendo se vuelve ininteligible
 - En otras palabras, sólo un nodo por vez puede hacer uso del canal de forma libre de interferencias



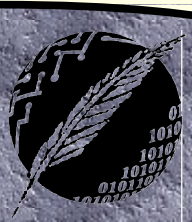
Protocolo de acceso múltiple

- Se basa en un algoritmo distribuido el cual determina **cómo compartir el canal**
 - ➔ La clave está en determinar cómo hacer una distribución de manera efectiva
 - ➔ Esto es, el protocolo debe definir cuándo puede transmitir cada nodo
 - ➔ Debemos tener en cuenta que toda comunicación relativa a cómo compartir el canal **también tiene que hacer uso de propio canal**



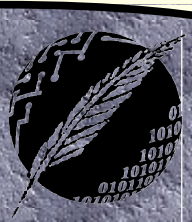
Protocolo ideal

- Para un canal compartido el cual cuenta con un ancho de banda **R**:
 - ➔ Cuando un nodo solo desee transmitir, debe poder hacerlo a la tasa **R**
 - ➔ Cuando **M** nodos deseen transmitir, cada uno debe poder hacerlo a una tasa promedio de **R / M**
 - ➔ Debe ser totalmente descentralizado
 - ➔ Es decir, no debe depender de un nodo especial para coordinar las transmisiones, o de un reloj compartido
 - ➔ Por último, debe ser simple



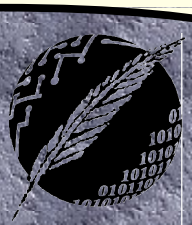
Taxonomía de protocolos MAC

- Los protocolos **MAC** (Medium Access Control) se clasifican en tres grandes grupos
- En primer lugar están los protocolos que **particionan el canal compartido**
 - ➔ La idea central es **dividir el canal compartido** en partes más pequeñas (por caso, frecuencias, ranuras o slots de tiempo, códigos, etc.)
 - ➔ Luego, **se asigna una parte del canal a cada nodo** para que haga un uso exclusivo de la misma



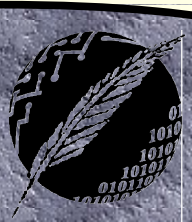
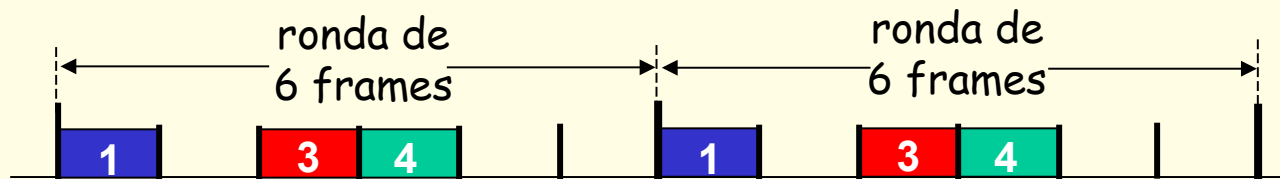
Taxonomía de protocolos MAC

- Otra alternativa es **organizar los accesos de manera aleatoria**
 - El canal no se divide en partes, pero se anticipa que **se van a producir colisiones**
 - Existe un mecanismo puntual para **recuperarse de esta situación**
- Finalmente, otra opción es “**tomar turnos**” a la hora de acceder al medio compartido
 - La idea es **coordinar de manera precisa el acceso al medio** a fin de evitar colisiones



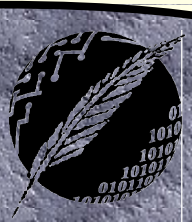
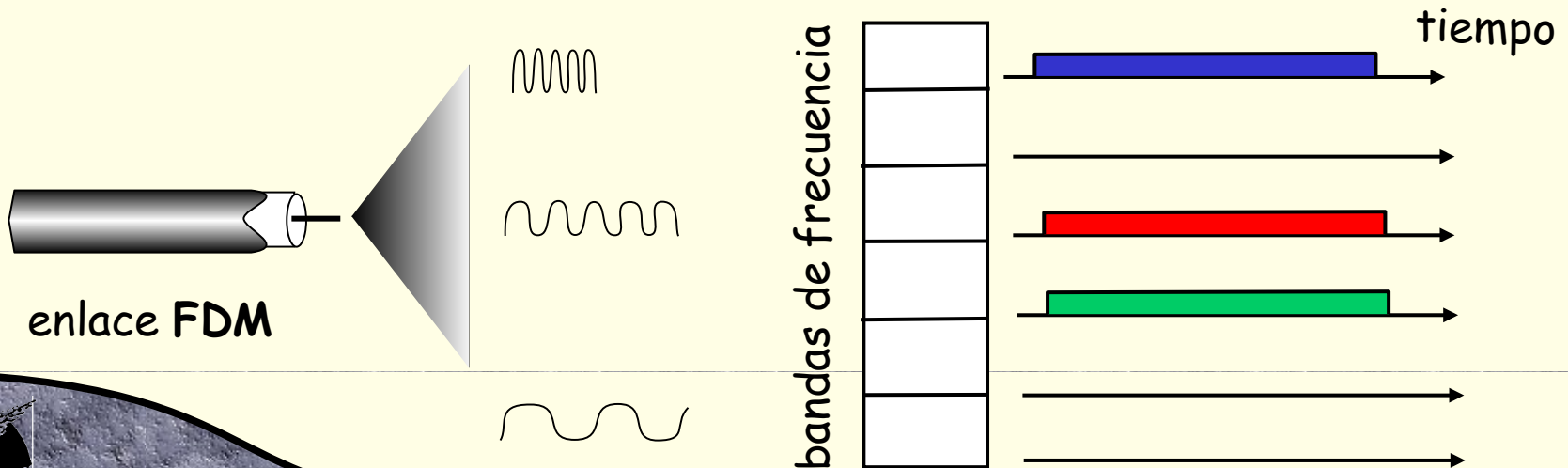
MAC vía TDMA

- La organización **TDMA** (Time Division Multiple Access) particiona el canal compartido entre los distintos nodos
 - El canal **es accedido periódicamente**, en rondas
 - Los nodos reciben una porción fija y predeterminada de tiempo en cada ronda
 - Las porciones que no sean usadas en una cierta ronda terminan siendo desperdiciadas



MAC vía FDMA

- La organización **FDMA** (Frequency Division Multiple Access)
- El espectro disponible **es particionado en canales**
- Cada estación recibe un determinado canal fijo
- Los canales asignados a nodos sin datos para transmitir serán desperdiciados



Protocolos de acceso aleatorio

- Otra posibilidad es organizar los accesos al medio compartido **de manera aleatoria**:
 - ➔ Cuando un nodo tenga datos para enviar **hará uso de la totalidad del ancho de banda R** del canal
 - ➔ Los nodos **no necesitan coordinarse** entre sí de manera anticipada
 - ➔ Si dos o más nodos transmiten a la vez, **se produce una colisión**



Protocolos de acceso aleatorio

- El protocolo de acceso al medio aleatorio debe resolver dos cuestiones:
 - ➔ ¿Cómo detectar que se produjo una colisión?
 - ➔ ¿Cómo se recupera de una colisión?
- Actualmente existen diversos protocolos **MAC** que siguen esta política:
 - ➔ **ALOHA** con slots
 - ➔ **ALOHA** puro
 - ➔ La familia **CSMA**, **CSMA/CD** y **CSMA/CA**



¿Preguntas?

