

# Tecnología Digital IV: Redes de Computadoras

## Clase 13: Nivel de Red - Parte 1

Lucio Santi & Emmanuel Iarussi

Licenciatura en Tecnología Digital  
Universidad Torcuato Di Tella

8 de mayo 2025

# Nivel de Red: agenda

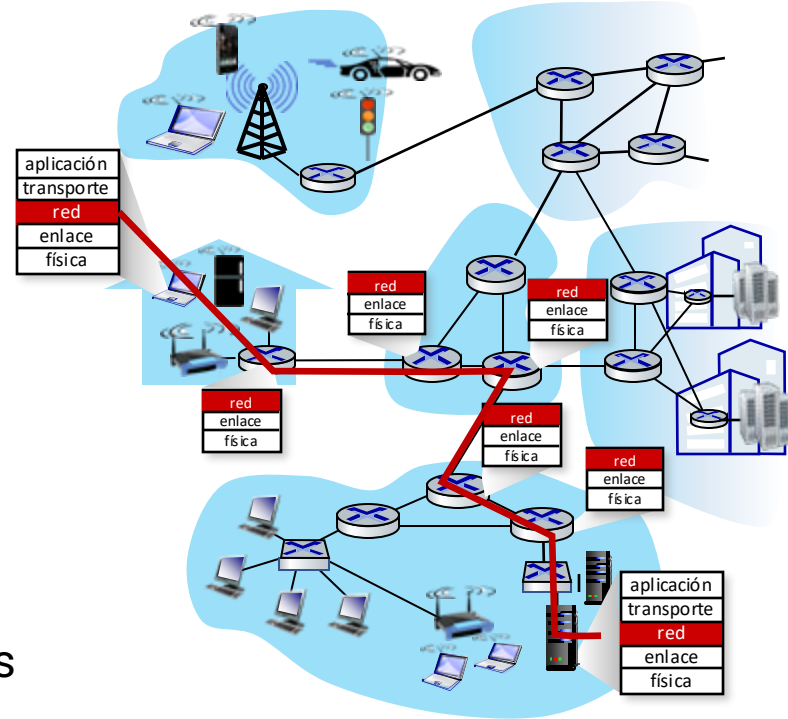
- Vista general del nivel de red
  - Plano de datos
  - Plano de control
- Routers
  - Puertos de *input* y *output*
  - Switching
- El Protocolo de Internet (IP)
  - Datagrama
  - Direcccionamiento
  - NAT
  - IPv6
- *Software Defined Networking (SDN)*

# Objetivos

- Entender los conceptos detrás de los servicios de la capa de red, focalizando en el **plano de datos**:
  - Modelos de servicio del nivel de red
  - Reenvío (*forwarding*) vs. ruteo (*routing*)
  - Cómo funciona un router
  - Direccionamiento
  - Arquitectura de Internet
- Instanciación e implementación en Internet
  - El protocolo IP
  - NAT
  - *Middleboxes*

# Servicios del nivel de red

- Transporte de segmentos entre un host emisor y un host receptor
  - **emisor**: encapsula segmentos dentro de datagramas, luego pasados al nivel de enlace
  - **receptor**: entrega segmentos al protocolo del nivel de transporte
- Los protocolos del nivel de red están en **todos los dispositivos** de Internet (hosts y routers)
- Los **routers**:
  - Inspeccionan el *header* de todos los datagramas IP en circulación en sus interfaces
  - Trasladan datagramas desde **puertos de entrada** hacia **puertos de salida**



# Funciones esenciales del nivel de red

## Funciones del nivel de red

- Reenvío (**forwarding**): trasladar paquetes desde un enlace de entrada hacia el enlace de salida adecuado
- Ruteo (**routing**): determinar la ruta que toman los paquetes desde el origen hasta el destino
  - Computada por **algoritmos de ruteo**

## Analogía: viajes en auto

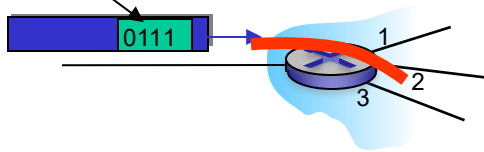
- **Forwarding**: proceso de atravesar una intersección vial
- **Routing**: proceso de planificar el viaje desde el origen hasta el destino

# Plano de datos y plano de control

## Plano de datos

- Función **local** de cada router
- Determina cómo se redirecciona un datagrama desde un puerto de entrada hacia uno de salida

valores en el *header*  
del paquete

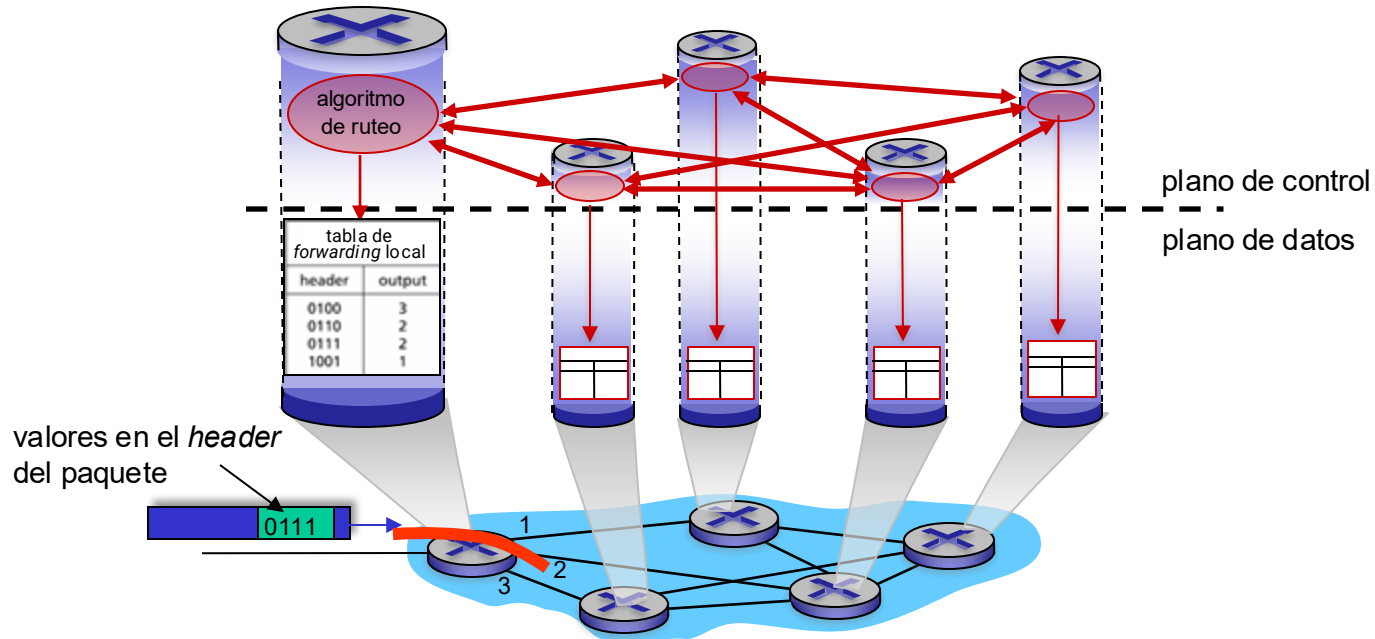


## Plano de control

- Función **global** de la red
- Determina cómo se encamina un datagrama entre distintos routers a lo largo de la red
- Dos enfoques:
  - Algoritmos de ruteo tradicionales: implementados en los routers
  - *Software-defined networking (SDN)*: implementado en servidores remotos

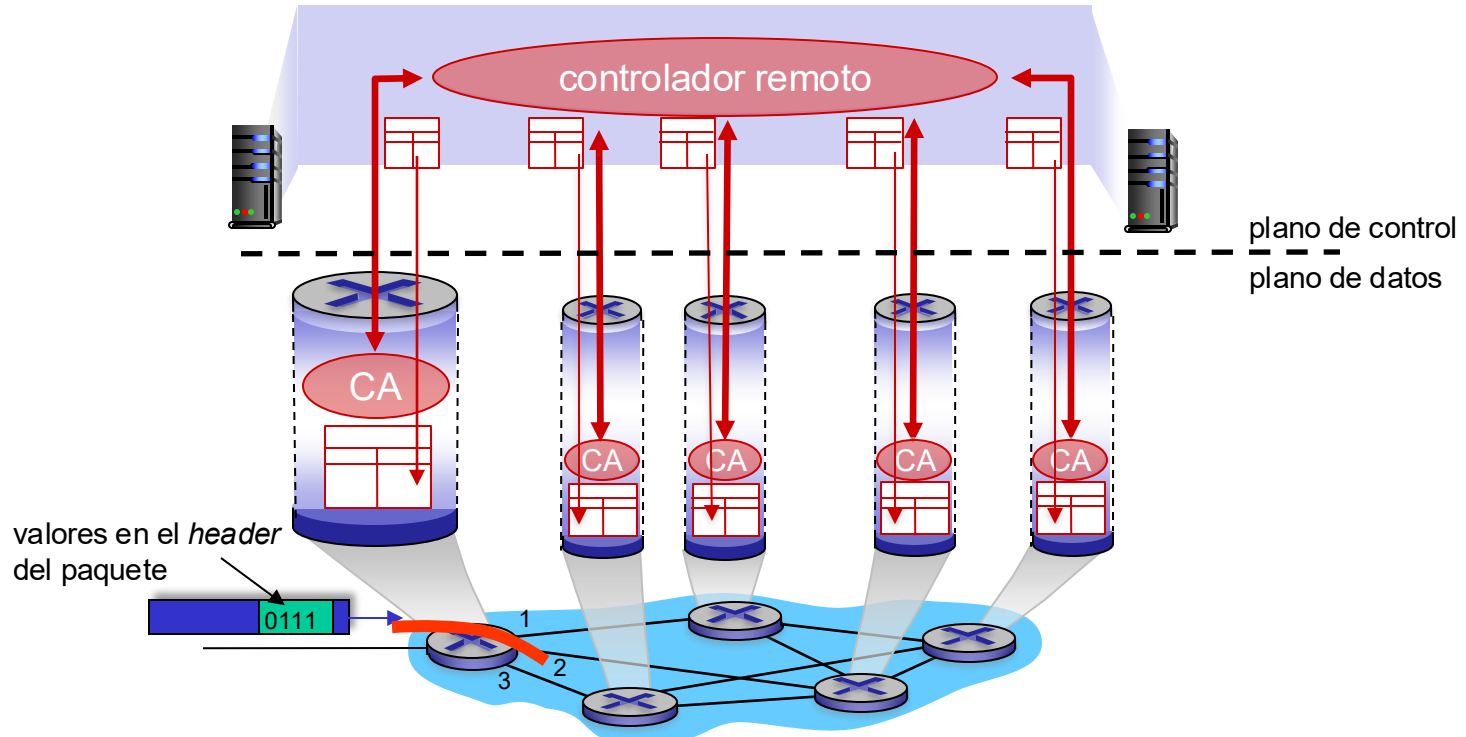
# Plano de control en cada router

Algoritmos de ruteo distribuidos, implementados en todos los routers e interactuando entre sí



# Plano de control vía SDN

Un controlador remoto computa e instala las tablas de *forwarding* en los routers





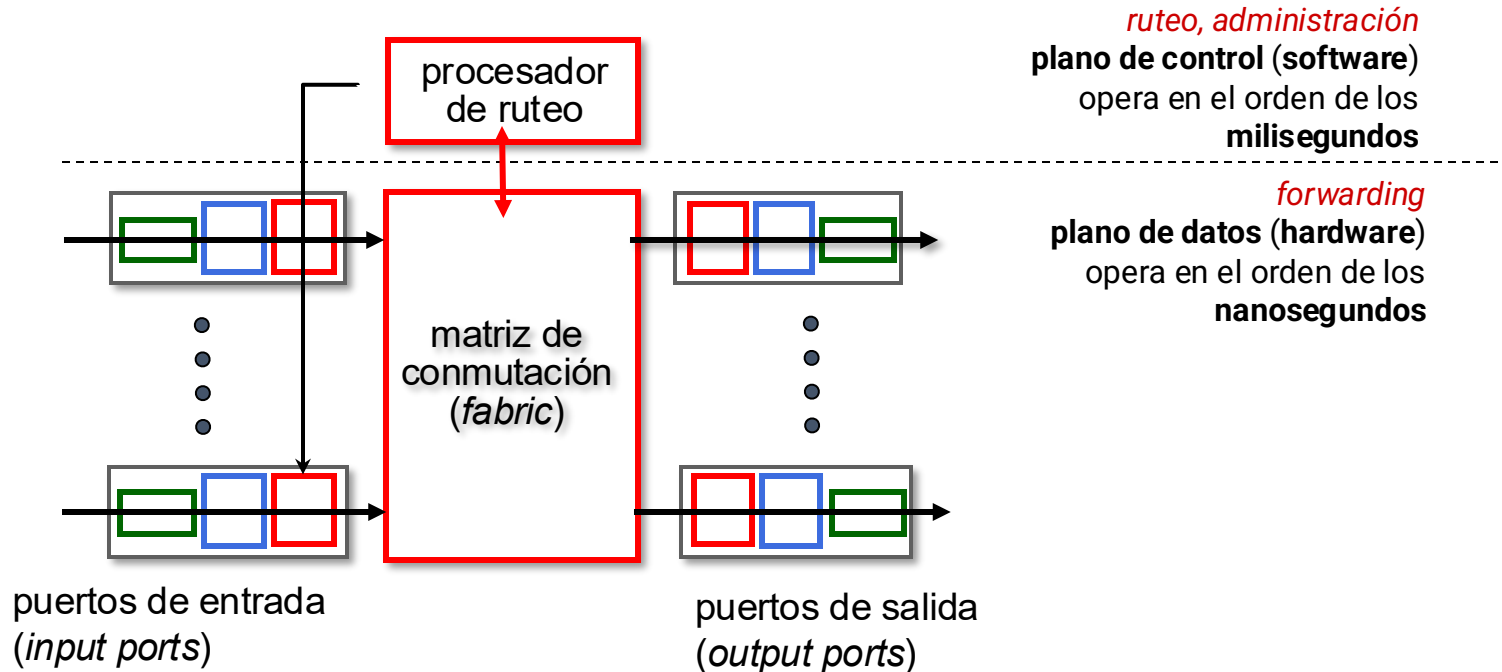
# Modelo de servicio del nivel de red

- En Internet, el nivel de red ofrece un servicio **best-effort**
- Sin garantías de:
  - Entrega de datagramas
  - Ordenamiento y/o tiempos de entrega
  - Capacidad disponible para flujos punto a punto
- Otras arquitecturas de redes implementan modelos de servicio más sofisticados (e.g. ATM)
- Entonces, ¿por qué un servicio best-effort?
  - Simplicidad; facilidad de adopción e instalación
  - La combinación con las funcionalidades de los protocolos de nivel superior demostró buenos resultados (e.g. performance adecuada para apps de streaming, videoconferencias, etc.)

# Arquitectura de routers

# Arquitectura de un router: vista general

Vista de alto nivel de la arquitectura de un router



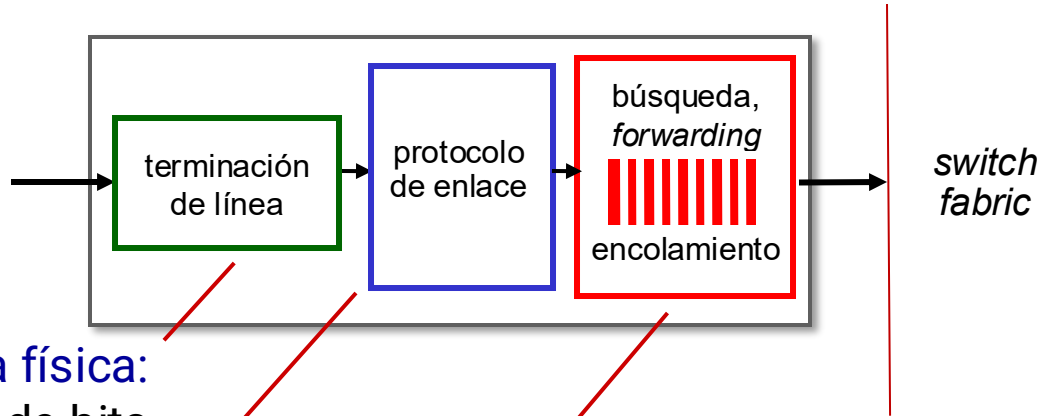
# Arquitectura de un router: vista general

En un **router**, la **switching fabric** (estructura de conmutación) es el **subsistema interno que conecta los puertos de entrada con los puertos de salida**, permitiendo que los paquetes/datos fluyan desde el puerto donde ingresaron hasta el puerto por el cual deben salir.

## **Función principal:**

Su función es **transferir un datagrama** desde la memoria del **puerto de entrada** hacia el **puerto de salida apropiado**, en base a las decisiones de reenvío.

# Funciones de los puertos de entrada



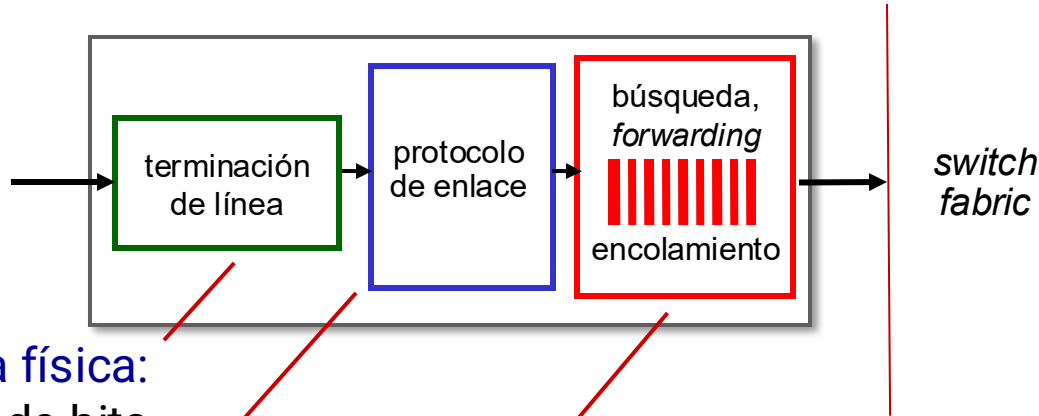
capa física:  
recepción de bits

capa de enlace:  
e.g. **Ethernet**

*switching* descentralizado:

- Utilizando los valores del *header*, buscar el puerto de salida en la tabla de *forwarding* copiada en la memoria del puerto de entrada
- Se produce **encolamiento** si los datagramas llegan más rápido que la tasa de *forwarding* hacia la matriz de conmutación

# Funciones de los puertos de entrada



capa física:  
recepción de bits

capa de enlace:  
e.g. **Ethernet**

*switching* descentralizado:

- Utilizando los valores del *header*, buscar el puerto de salida en la tabla de *forwarding* copiada en la memoria del puerto de entrada
- *Destination-based forwarding*: tradicional; sólo utiliza la dirección IP destino
- *Forwarding generalizado*: utiliza otros valores del *header*

# *Destination-based forwarding*

Rango de direcciones destino	Interfaz
11001000 00010111 00010000 00000000 A	0
11001000 00010111 00010000 11111111	
11001000 00010111 00011000 00000000 A	1
11001000 00010111 00011000 11111111	
11001000 00010111 00011010 00000000 A	2
11001000 00010111 00011110 00000000	
en otro caso	3

# Coincidencia de prefijo más largo

## coincidencia de prefijo más largo

al buscar la entrada en la tabla de *forwarding* para una dirección destino dada, utilizar la del prefijo más largo que coincida con dicha dirección

Rango de direcciones destino	Interfaz
11001000 00010111 00010 *** *****	0
11001000 00010111 00011000 *****	1
11001000 00010111 00011 *** *****	2
en otro caso	3

ejemplos:

11001000	00010111	00010110	10100001	<b>0</b>
11001000	00010111	00011000	10101010	<b>1</b>



# Coincidencia de prefijo más largo

- La razón detrás del uso de esta estrategia radica en cómo funciona el direccionamiento en Internet (lo veremos más adelante)
- Típicamente, esto se implementa con un hardware especializado—**TCAMs** (*Ternary Content Addressable Memories*)
  - Al recibir una dirección IP de 32 bits, la TCAM obtiene la respuesta en un ciclo de clock (independientemente del tamaño de la tabla)
  - Por ejemplo, los routers Catalyst de Cisco pueden alojar alrededor de 1 millón de entradas de la tabla en una TCAM

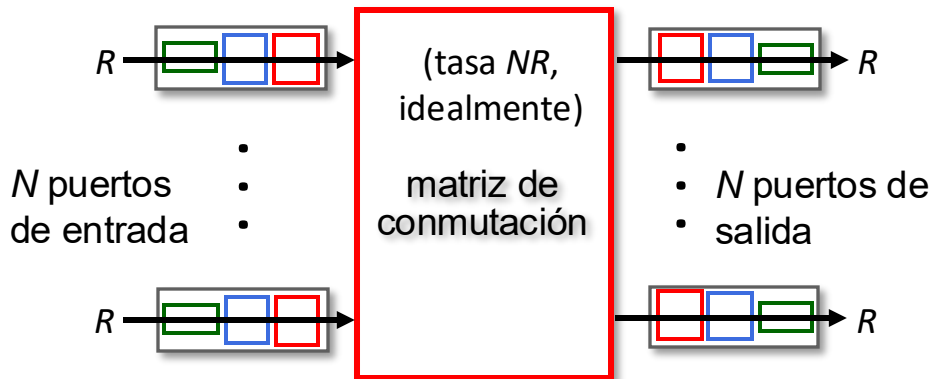
# Coincidencia de prefijo más largo

- La razón detrás del uso de esta estrategia radica en cómo funciona el direccionamiento en Internet (lo veremos más adelante)
- Típicamente, esto se implementa con un hardware especializado—**TCAMs** (*Ternary Content Addressable Memories*)
  - Al recibir una dirección IP de 32 bits, la TCAM obtiene la respuesta en un ciclo de clock (independientemente del

Prefijo IP	Máscara	Salida
11000000.10101000.X.X	/16 (192.168.0.0)	eth0
11000000.10101000.00000001.X	/24 (192.168.1.0)	eth1

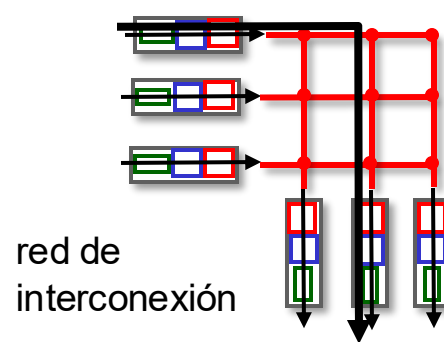
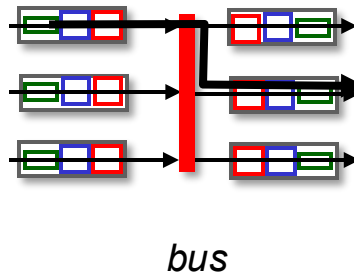
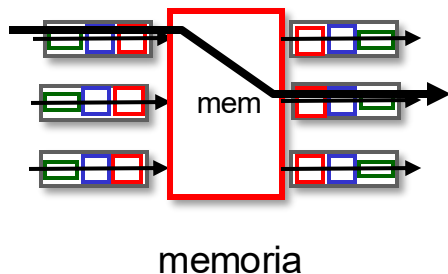
# Matrices de conmutación

- **Objetivo:** trasladar paquetes desde los enlaces de entrada a los enlaces de salida que les corresponden
- *tasa de conmutación:* tasa a la cual es posible conmutar paquetes desde las entradas hacia las salidas
  - Se mide como múltiplo de la tasa de transmisión de los enlaces de entrada
  - Con  $N$  entradas, es deseable una tasa de conmutación de  $N$  veces la tasa de transmisión



# Matrices de conmutación

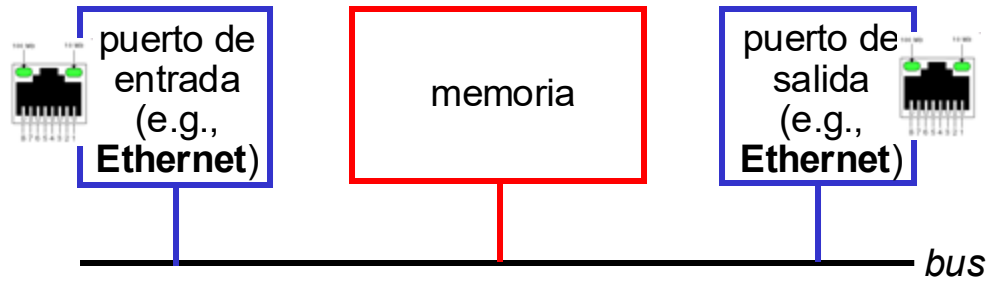
- **Objetivo:** trasladar paquetes desde los enlaces de entrada a los enlaces de salida que les corresponden
- *tasa de conmutación:* tasa a la cual es posible conmutar paquetes desde las entradas hacia las salidas
  - Se mide como múltiplo de la tasa de transmisión de los enlaces de entrada
  - Con  $N$  entradas, es deseable una tasa de conmutación de  $N$  veces la tasa de transmisión
- Tres clases principales:



# Conmutación en memoria

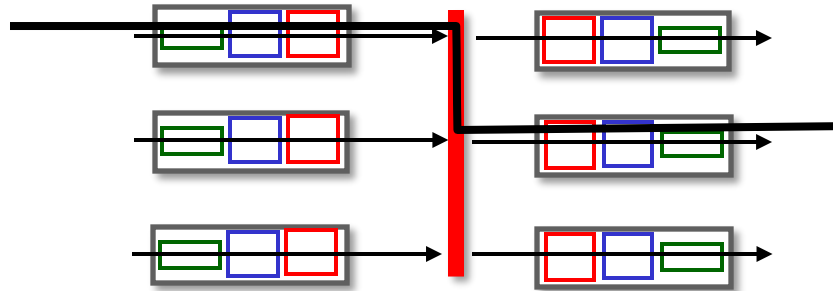
Routers de primera generación (1980s - e.g. Cisco):

- Eran computadoras tradicionales; el *switching* se hacía por CPU
- El paquete se copiaba a la memoria RAM (interrupciones al CPU ante la llegada de paquetes)
- La velocidad queda limitada por el ancho de banda de la memoria



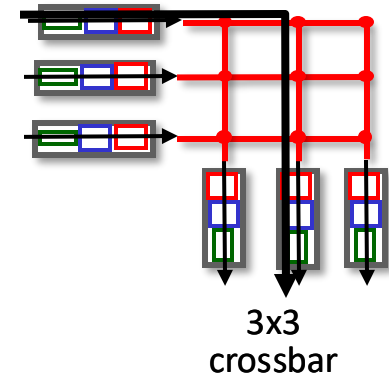
# Conmutación a través de un *bus*

- Los paquetes se transmiten a los puertos de salida sin intervención del procesador
- Etiqueta interna con el puerto de salida para viajar por el *bus*; todos los puertos reciben el paquete
- Velocidad limitada por el ancho de banda del bus
- Suele operar bien para routers de redes locales



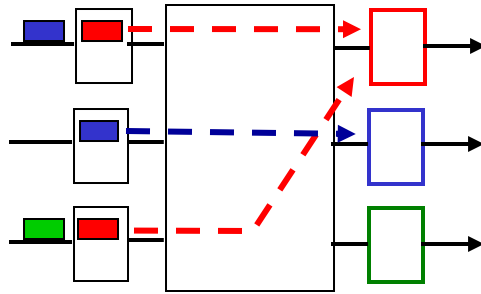
# Conmutación a través de red de interconexión

- *Crossbar*:  $2N$  buses que conectan  $N$  puertos de entrada con  $N$  puertos de salida (figura:  $N = 3$ )
- Los puntos de cruce pueden abrirse o cerrarse por un controlador
- Al llegar paquetes, el controlador cierra el punto de cruce apropiado y coloca el paquete en el *bus*
- Admiten **paralelismo**: transmisión de múltiples paquetes en simultáneo

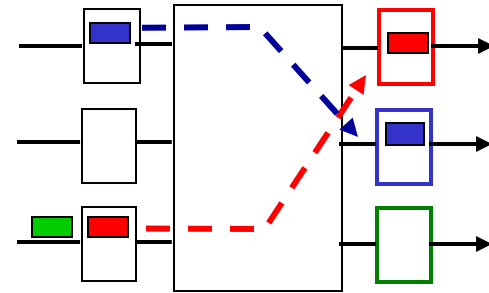


# Encolamiento en puertos de entrada

- Se produce **encolamiento** en los puertos de entrada si la matriz de conmutación es más lenta que los puertos de entrada combinados
  - Esto genera **demoras** y posibles **pérdidas** por *overflow* de buffers
- Head-of-Line (HOL) blocking**: el paquete encolado al frente no permite que los demás avancen



**contención**: sólo puede transferirse un paquete rojo (mismo puerto)  
el paquete rojo inferior queda **bloqueado**



el paquete verde experimenta  
*HOL blocking*



# Encolamiento en puertos de salida

- Se produce **encolamiento** cuando los datagramas llegan más rápido que la tasa de transmisión del enlace
- **Política de drop**: decidir cuáles paquetes descartar si no hay lugar disponible
- **Política de scheduling**: determinar cuáles paquetes en la cola se van a transmitir primero

