

Red "switchheada"

❑ 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 sólo VLAN posible, salvo en los *trunks*

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

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 señalar 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**

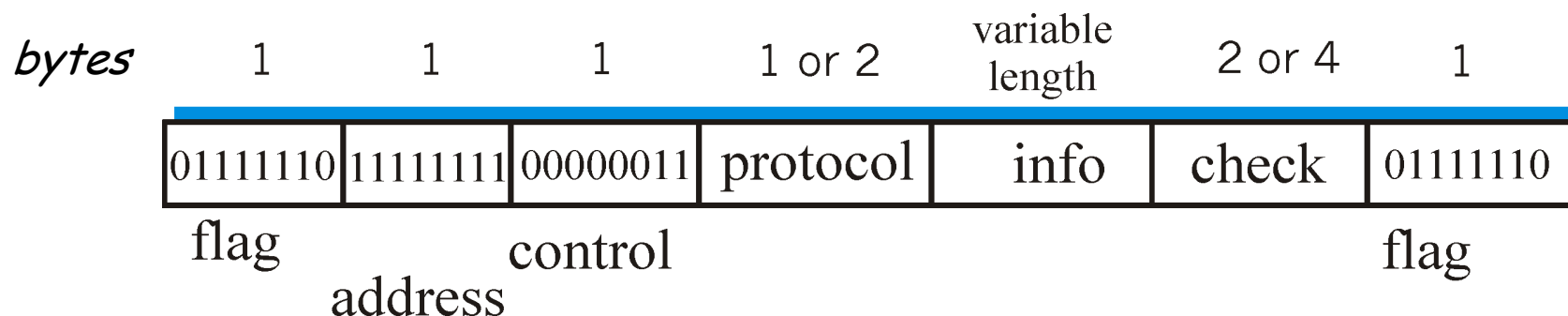
No requerimientos de PPP

- ❑ 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

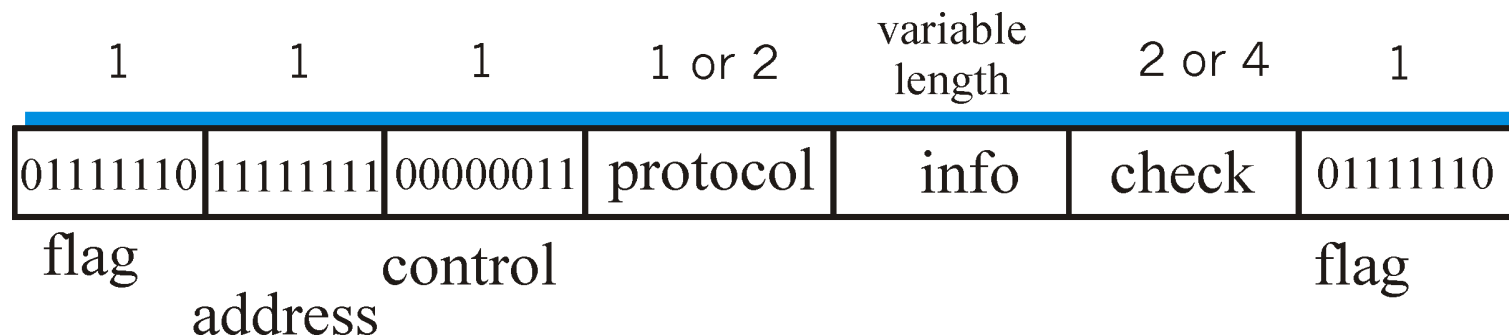
Trama de Datos de PPP (RFC 1662)

- ❑ **Flag:** delimitador (*framing*); sincronización de trama
- ❑ Se pueden no enviar lo siguientes campos
 - **Address:** no hace nada; uso futuro
 - **Control:** no hace nada; uso futuro
- ❑ **Protocol:** protocolo de capa superior a quien debe ser entregada la trama (ej., LCP, NCP, IP, etc)



Tramas de Datos de PPP

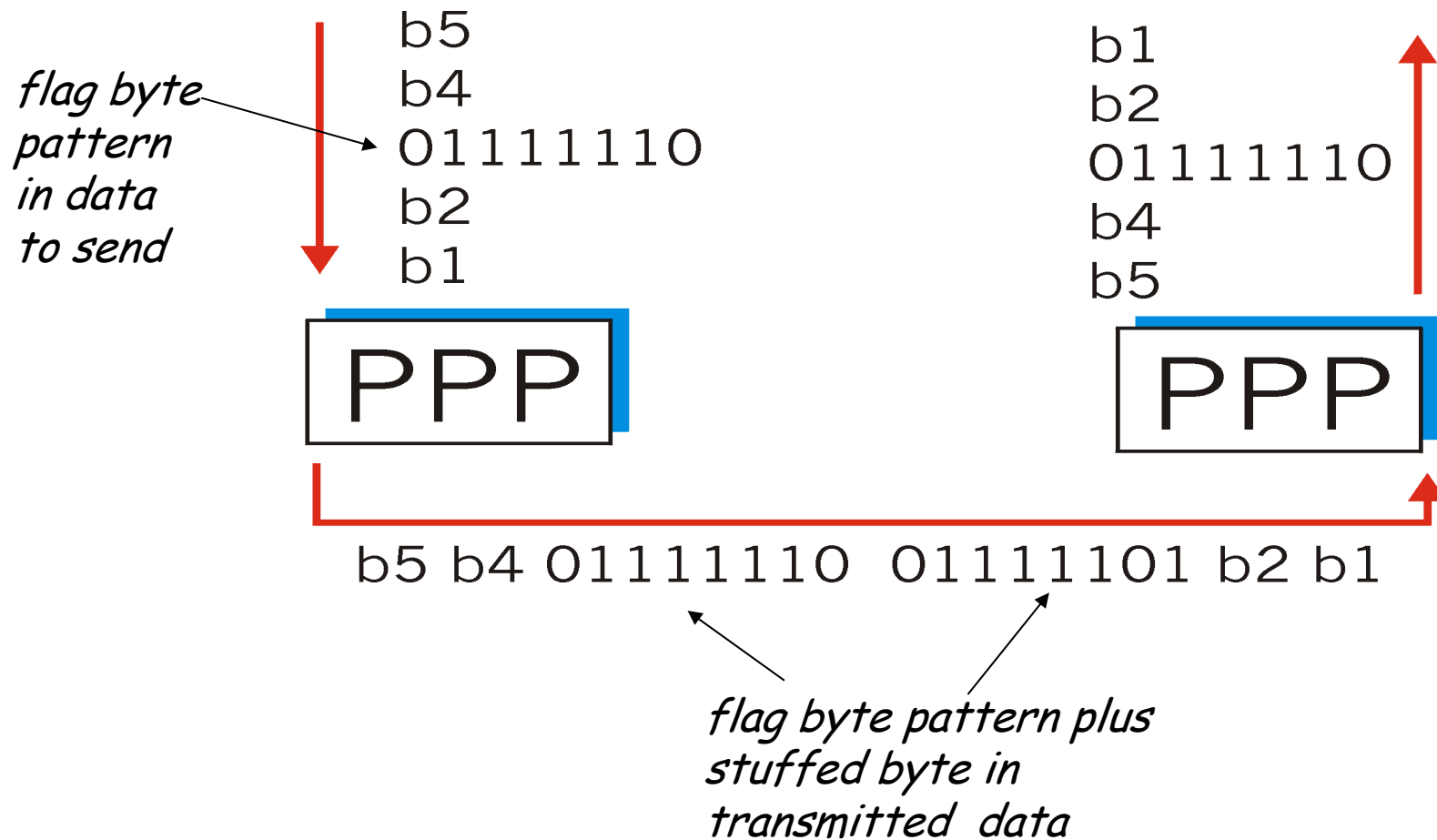
- *Info:* datos de la capa superior que son llevados; 1500 bytes máximo o negociable
- *Check:* detección de errores con *Cyclic Redundancy Check*



Byte Stuffing ("relleno")

- No es algo exclusivo de PPP
- requerimiento de "transparencia de los datos": el campo de datos debe permitir incluir el patrón de la *flag* <01111110>
 - P: se recibe <01111110>, ¿es datos o *flag*?
- **Transmisor**: agrega ("*stuffs*") un byte de relleno <01111101> (byte de escape) extra antes de cada byte <01111110> de *datos*. Ídem si aparece el byte de escape.
- **Receptor**:
 - Reacciona a la aparición de <01111110> o <01111101> respetando lo anterior
 - un único <01111110>: byte *flag*

Byte Stuffing



Protocolo de Control de PPP Link Control Protocol

- antes de intercambiar datos de la capa de red, los *peers* de la capa de enlace deben configurar el enlace y algunas cosas relacionadas con la capa de red

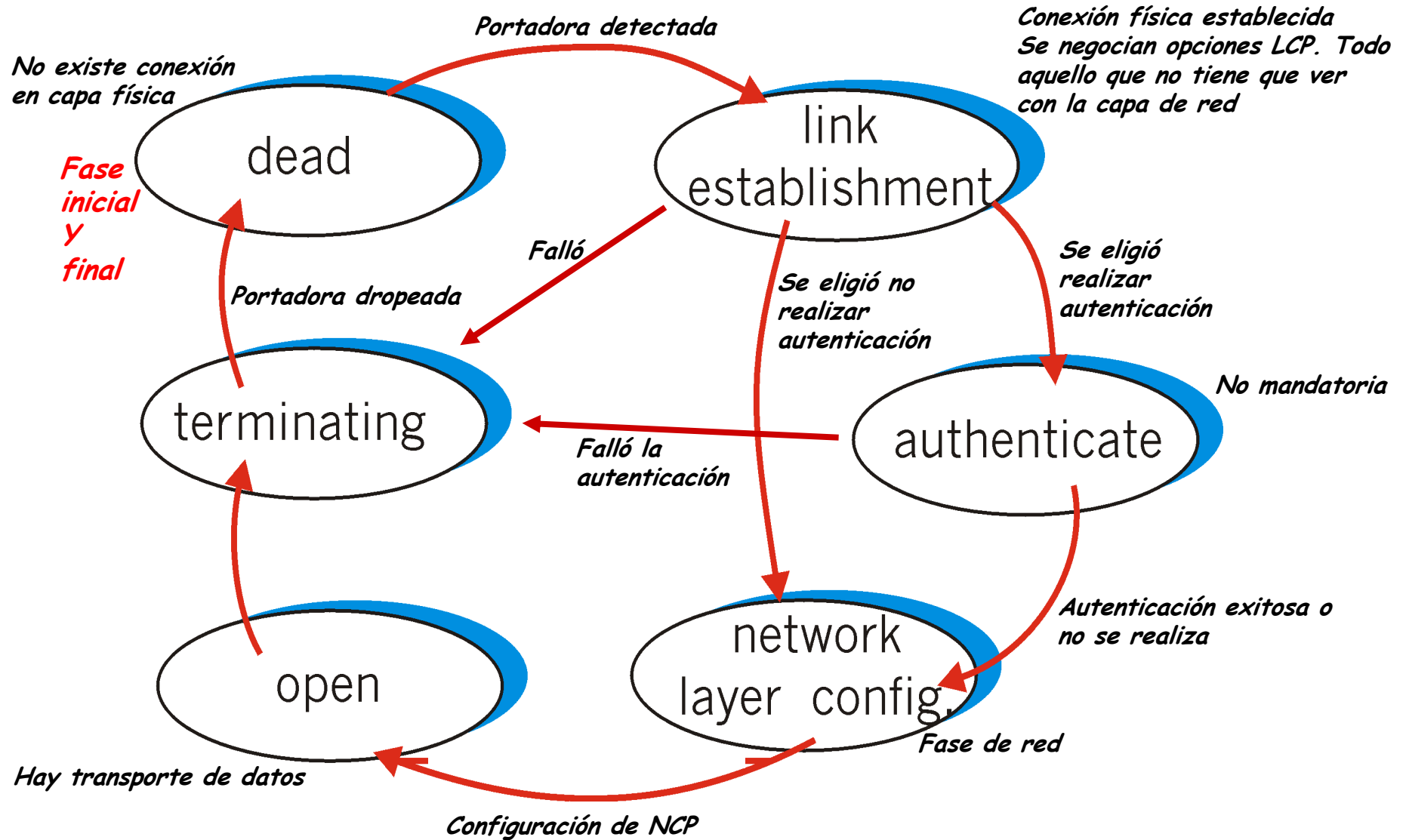
- **configurar enlace PPP**
 - máx. long. de trama
 - uso o no de autenticación
 - protocolo
 - *testing* de la calidad de la línea
 - compresión de *header*

Protocolos de Control de PPP *Network Control Protocol*

- configuración de aspectos de capa red

- NCP para IP: IPCP
 - asignación de dirección IP
 - asignación de direcciones de servidores DNS

Diagrama simplificado de fases



Virtualización en Internet

- ❑ En la evolución del estudio de capa de enlace:
- ❑ Cable que conecta dos computadoras -> múltiples computadoras conectadas por un cable compartido y ese "cable" puede ser el aire -> infraestructura de switches
- ❑ Por lo tanto, hemos ido agregando "complejidad" en el canal de interconexión, pero para los hosts sigue siendo "la capa de enlace entre nodos adyacentes"
- ❑ En el caso de PPP utilizado en el acceso directo a Internet, se virtualiza completamente la red telefónica ("es un cable")

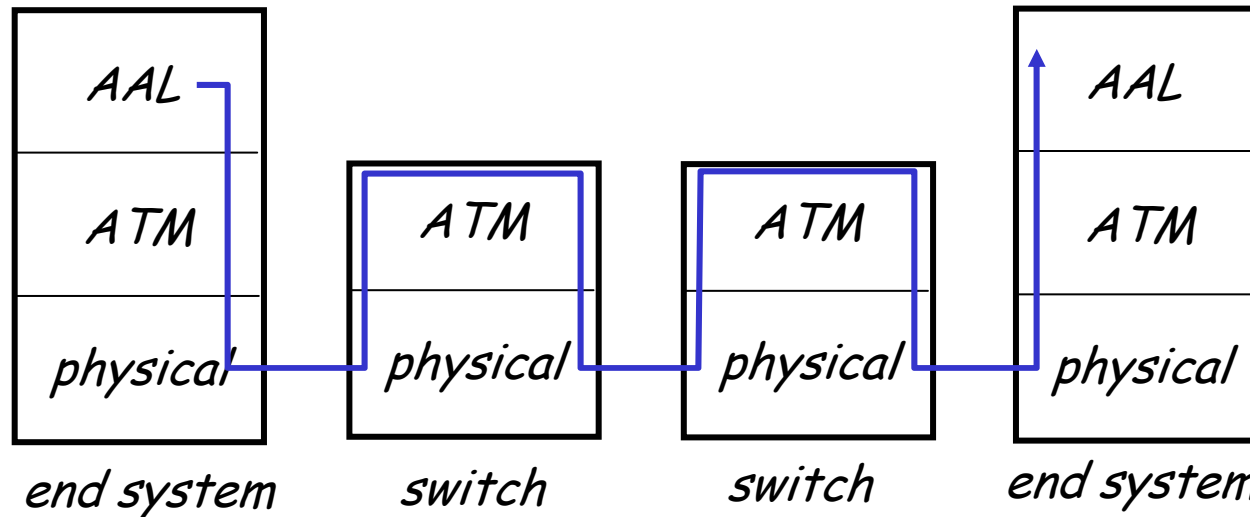
ATM y MPLS

- Las redes ATM y MPLS (circuitos virtuales, conmutación de paquetes, con sus formas propias de paquetes y técnicas de *forwarding*) se pueden ver como tecnologías de capa de enlace que interconectan dispositivos IP.
 - ATM: *Asynchronous Transfer Mode*
 - MPLS: *Multiprotocol Label Switching*

Asynchronous Transfer Mode: ATM

- Estándar de los 80's para alta velocidad (desde 155Mbps hasta 622 Mbps y más); arquitectura *Broadband Integrated Service Digital Network*
 - ex-ATM Forum, ex-MFA Forum, ahora IP/MPLS Forum, e ITU
- Objetivo: transporte integrado y *end-to-end* de voz, video y datos
- Principales características
 - soporta diversos modelos de servicio (CBR, VBR, ...)
 - reúne requerimientos de QoS de voz y video (versus el modelo *best effort* de Internet)
 - *packet-switching* (paquetes de longitud fija, llamados "celdas") utilizando una arquitectura de red de circuitos virtuales
 - Servicio orientada a conexión (*Virtual Channel o Virtual Circuit*)
 - 3 capas: AAL, ATM y PHY

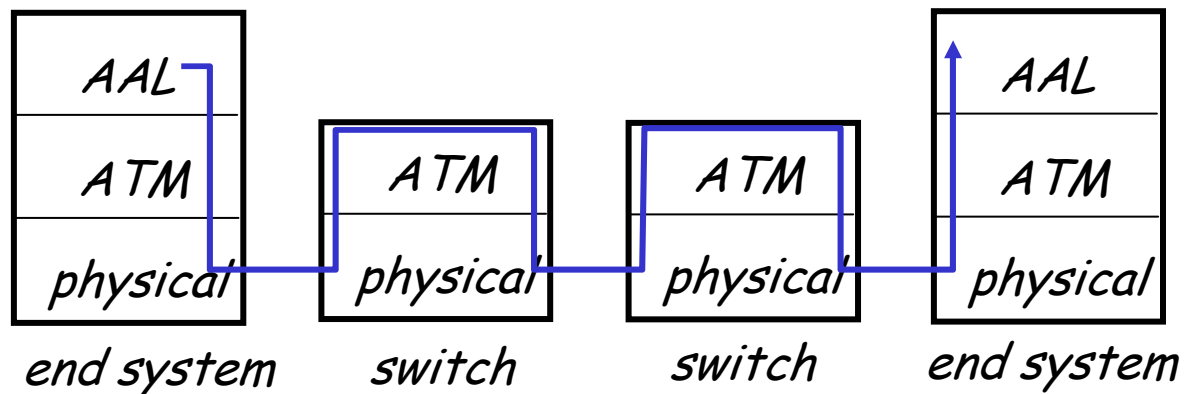
Arquitectura ATM



- **ATM Adaptation Layer:** sólo en el borde de la red ATM
 - Servicios: detección de errores, segmentación y re-ensamblado (SAR en inglés) de los datos
 - Sin mucha rigurosidad, análogo a la capa de transporte de Internet
 - Sobre ella, por ejemplo, IP
- **ATM layer:** "network" layer
 - Switching de celdas
- **physical layer**
 - Según el medio físico

ATM Adaptation Layer (AAL)

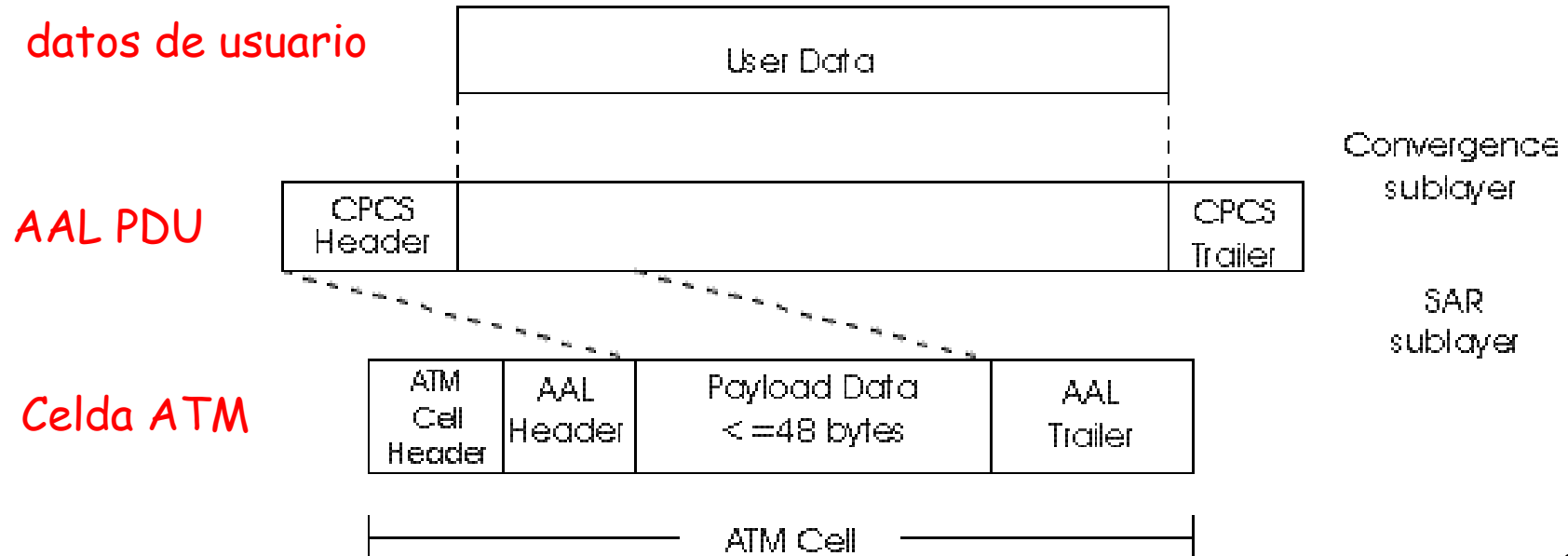
- *ATM Adaptation Layer (AAL)*: "adapta" las capas superiores (IP o aplicaciones ATM nativas) a la capa ATM de abajo
- AAL está presente **solamente** en los *end systems*, no en los *switches*
- Los PDUs de la capa AAL son fragmentados en múltiples celdas ATM



ATM Adaptation Layer (AAL), más...

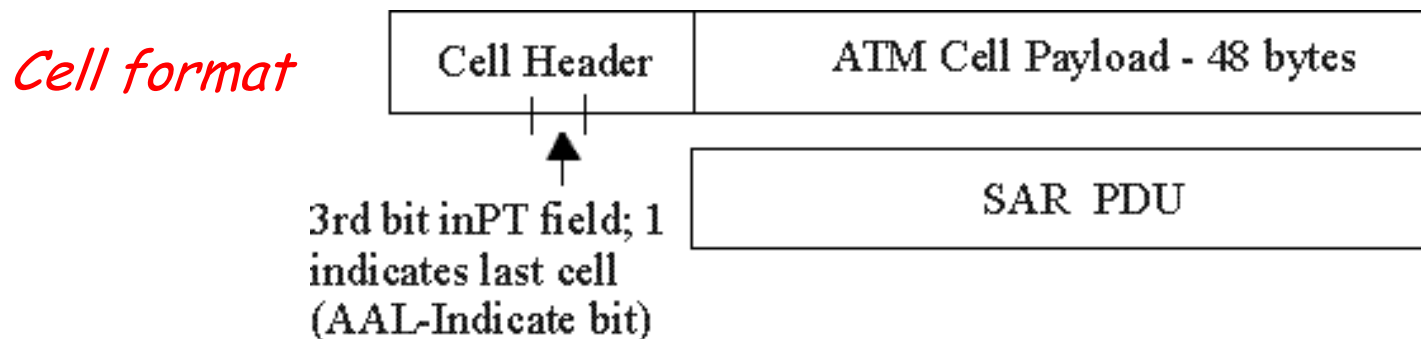
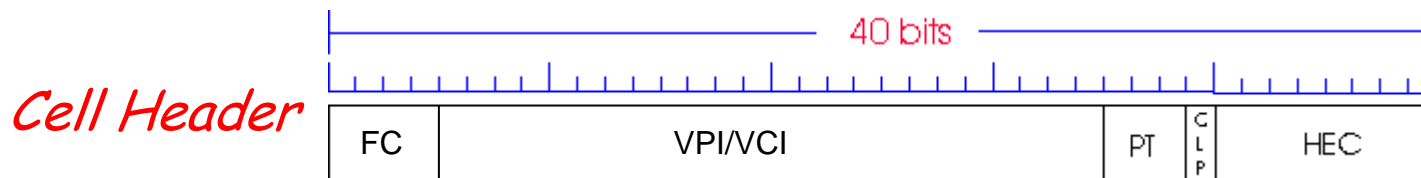
Diferentes "sabores" de capas AAL, dependiendo de la clase de servicio ATM:

- ❑ **AAL1:** para servicios CBR (*Constant Bit Rate*), p.e. E1, T1
- ❑ **AAL2:** para servicios VBR (*Variable Bit Rate*), p.e., MPEG video
- ❑ **AAL5:** para datos (p.e., datagramas IP)



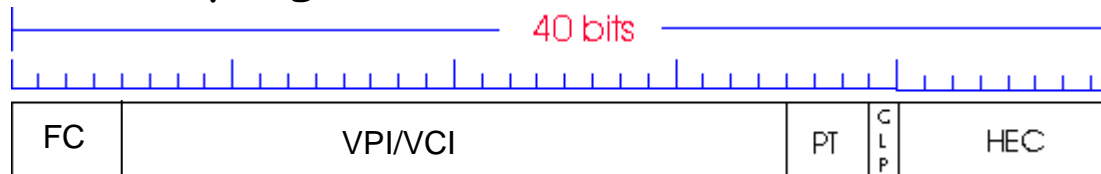
Capa ATM: celda ATM

- ❑ Encabezado de 5 bytes
- ❑ Carga útil de 48 bytes
 - ¿Por qué?: payload pequeño -> corto retardo de creación de la celda; para voz digitalizada
 - semisuma de 32 y 64



Encabezado de la celda ATM

- **FC:** Control de flujo
 - Sólo se utiliza en el borde de la red
- **VPI/VCI:** *Virtual Path ID / Virtual Channel ID*
 - Uno por cada VC
 - 24 bits
 - VPI: camino; 8 bits
 - VCI: un VC; 16 bits
 - cambia de *link* a *link* a través de la red (*label switching*)
- **PT:** *Payload Type* (p.e. celda de gestión, celda de datos)
- **CLP:** *Cell Loss Priority bit*
 - CLP = 1 implica celda de baja prioridad, puede ser descartada si hay congestión
- **HEC:** *Header Error Checksum*
 - Cyclic Redundancy Check
 - Detección y algunas correcciones



Capa ATM: Circuitos Virtuales

- **transporte VC:** celdas llevadas en VC desde el origen al destino
 - establecimiento de llamada
 - cada paquete lleva identificador de VC (no ID de destino)
 - *cada switch* en el camino entre el origen y el destino mantiene el "estado" para cada conexión que pasa por él
 - enlace, recursos de conmutación (ancho de banda, *buffers*) pueden ser *reservados* al VC: para tener una comportamiento *circuit-like*
- ***Permanent VCs (PVCs)***
 - conexiones larga vida; estáticas
 - Establecimiento: *provisioning*
 - típicamente: encaminado "permanente" entre *routers IP*
- ***Switched VCs (SVC):***
 - establecidos dinámicamente
 - *signaling* entre el *host* y el switch ATM
 - Protocolo de señalización PNNI (*Private Network to Network Interface*)
 - VPI/VCI para control

ATM VCs

□ Ventajas de ATM VC:

- *Performance* de QoS garantizada para la conexión mapeada al VC (ancho de banda, retardo, *jitter*)

□ Desventajas de ATM VC:

- un PVC entre cada pareja origen/destino: no escala
- SVC introduce la latencia del establecimiento de la llamada, *overhead* de procesamiento para conexiones de corta vida

Capa ATM

Servicio: transportar celdas a través de la red ATM

- ❑ análogo a la capa de red IP
- ❑ servicios muy diferentes a los de la capa de red IP

- ❑ **CBR: *Constant Bit Rate***
 - Velocidad constante fija y retardo máximo
 - Reserva de recursos
- ❑ **VBR: *Variable Bit Rate***
 - Aplicaciones sensibles al tiempo (retardo y *jitter*)
 - Velocidad pico y promedio
 - Pueden haber ráfagas
 - Reserva de recursos
- ❑ **ABR: *Available Bit Rate***
 - Velocidad pico y mínima de celdas
 - Reserva de recursos
- ❑ **UBR: *Unspecified Bit Rate***
 - Retardos variables y ciertas tasas de pérdidas
 - No hay reserva de recursos ni toma en cuenta la congestión

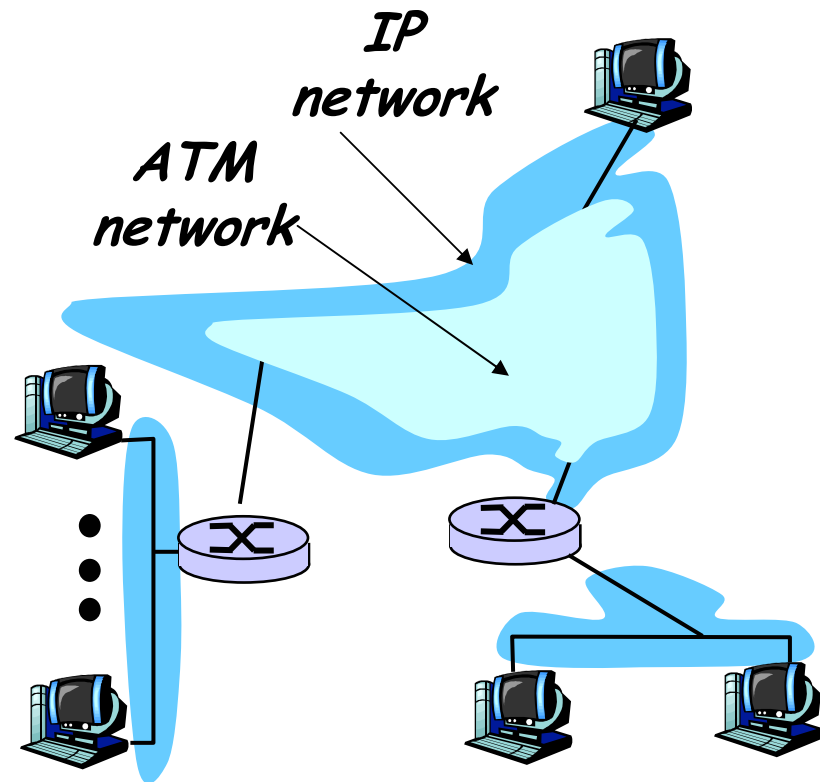
ATM: ¿capa de red o capa de enlace?

Visión: transporte *end-to-end*: "ATM desde el *desktop* al *desktop*"

- ATM es una tecnología de red

Realidad: utilizado para conectar los routers del *backbone* IP

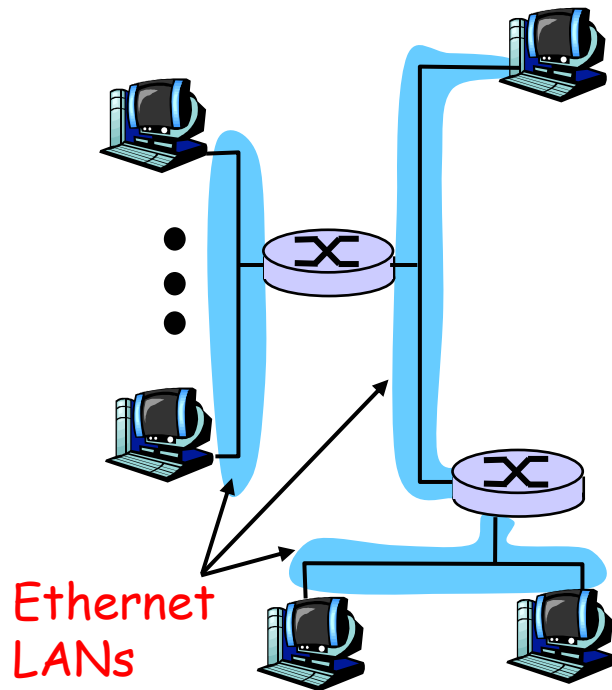
- "IP over ATM"
- ATM como una capa de enlace *switched*, conectando routers IP



IP sobre ATM

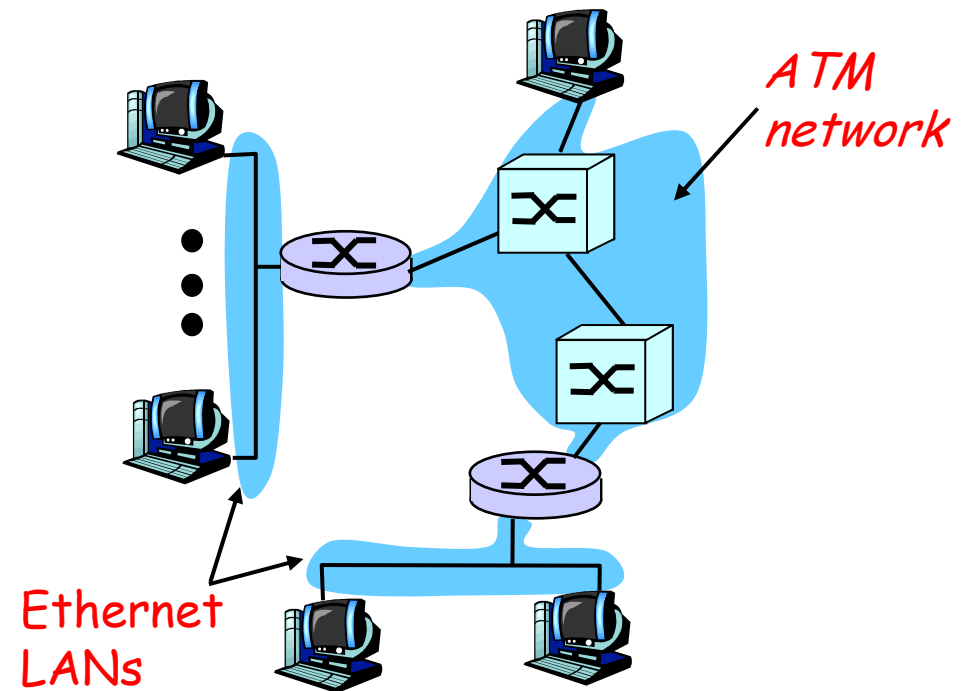
IP Clásico

- ❑ 3 "redes" (p.e., segmentos LAN)
- ❑ direcciones MAC (802.3) e IP

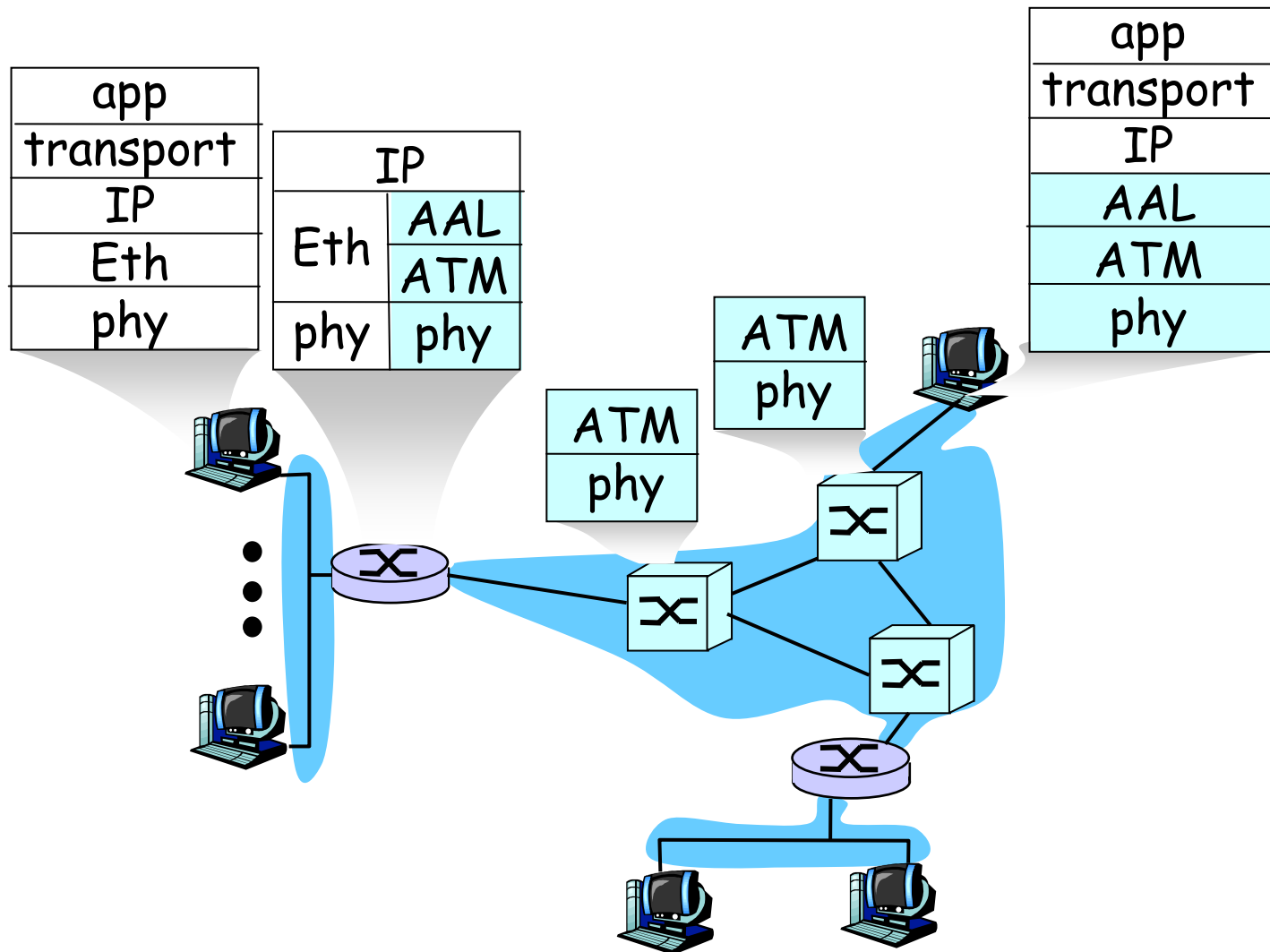


IP sobre ATM

- ❑ reemplaza "red" (p.e., segmento LAN) con red ATM
- ❑ direcciones ATM e IP



IP sobre ATM



Viaje de datagrama en red IP sobre ATM

- *en el host origen:*
 - La capa IP mapea entre las direcciones destino IP y ATM (utilizando ARP)
 - Pasa los datagramas a AAL5
 - AAL5 encapsula los datos, segmenta en celdas y las pasa a la capa ATM
- *red ATM:* mueve celdas a través de VC hacia el destino
- *en el host destino:*
 - AAL5 re-ensambla las celdas en el datagrama original
 - si CRC OK, el datagrama es pasado a IP

IP sobre ATM

- ❑ datagramas IP dentro de PDUs de ATM AAL5
- ❑ de direcciones IP a direcciones ATM
 - igual que de direcciones IP a direcciones MAC 802.3
 - protocolo ATMARP (RFC 2225)

