



# Conmutación

Karima Velásquez

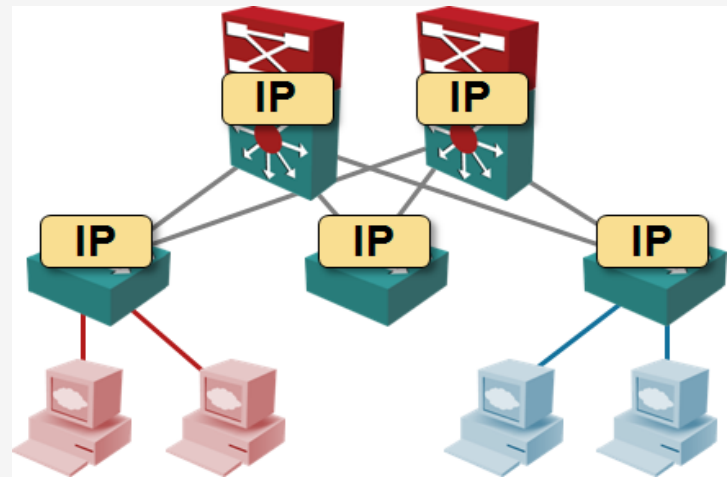
[karima.velasquez@ciens.ucv.ve](mailto:karima.velasquez@ciens.ucv.ve)



# Agenda

- Switching vs. Forwarding
- ATM
- Frame Relay
- MPLS

# Switching vs. Forwarding



# Switching vs Forwarding

- Un *switch* es un dispositivo que permite interconectar enlaces para formar redes más grandes
- Es un dispositivo multi entrada, multi salida que transfiere paquetes de una entrada a una o más salidas

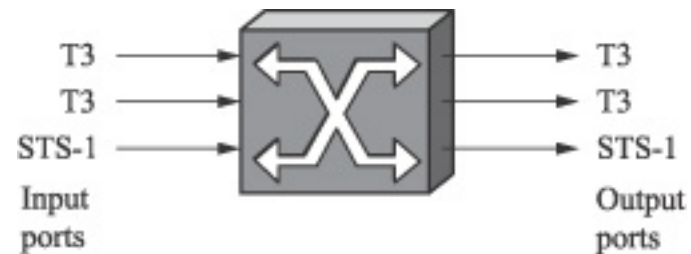
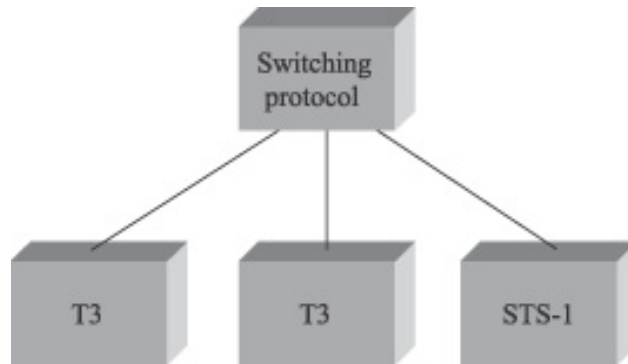


# Switching vs Forwarding

- Un gran número de hosts pueden ser conectados, interconectando varios switches
- Los switches se pueden interconectar usando enlaces punto a punto
- Conectar un host agregando un switch no reduce el rendimiento de la red

# Switching vs Forwarding

- Switching (conmutación) consiste en recibir paquetes entrantes por un enlace y transmitirlo por otro



# Switching vs Forwarding

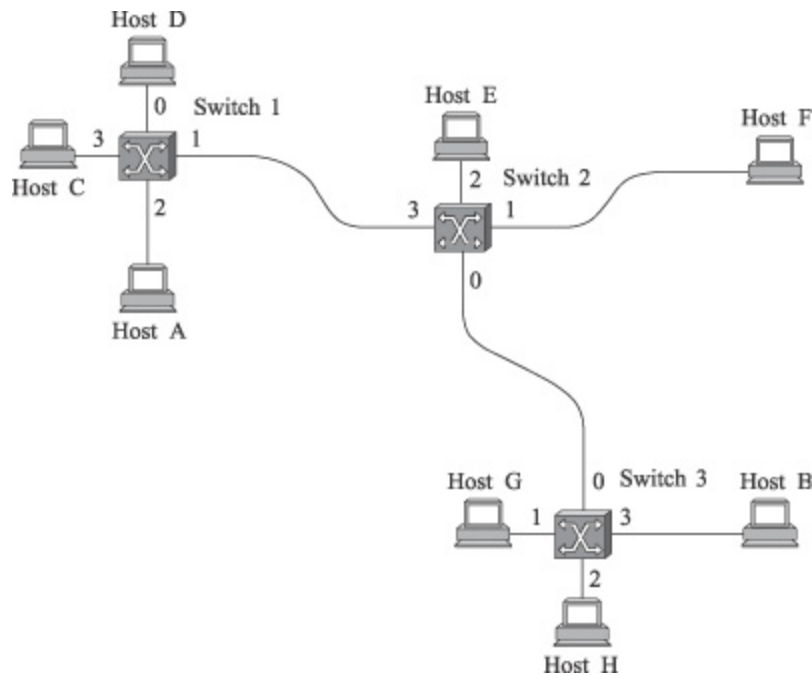
- ◉ ¿Cómo decide el switch el puerto de salida para el paquete?
  - ◉ Observa un identificador en el encabezado del paquete
  - ◉ Hay tres aproximaciones para ello:
    - ◉ Datagrama
    - ◉ Circuito virtual
    - ◉ Enrutamiento fuente

# Switching vs Forwarding: Datagrama

- Datagrama
  - Cada paquete necesita tener suficiente información para que el switch pueda enviarlo a su destino



# Switching vs Forwarding: Datagrama



Destination	Port
A	3
B	0
C	3
D	3
E	2
F	1
G	0
H	0

**Table 3.1 Forwarding table for switch 2.**

# Switching vs Forwarding: Datagrama

- Se requiere enrutamiento
- Cuando un paquete llega, se tiene la información correcta en la tabla de enrutamiento para enviarlo

# Switching vs Forwarding: Datagrama

- Un paquete puede ser enviado inmediatamente (asumiendo una tabla correctamente llena)
- Cuando un paquete llega, no se sabe si la red puede enviarlo o si el destino está operativo (*up & running*)
- Cada paquete es enviado de forma independiente
- Una falla del switch o del enlace puede no tener un efecto grave sobre la comunicación

# Switching vs Forwarding: Circuito Virtual

- Requiere que una conexión sea establecida antes de que se envíe la data
- Requiere de:
  - Establecimiento de la conexión
  - Transferencia de la data
  - Terminación de la conexión

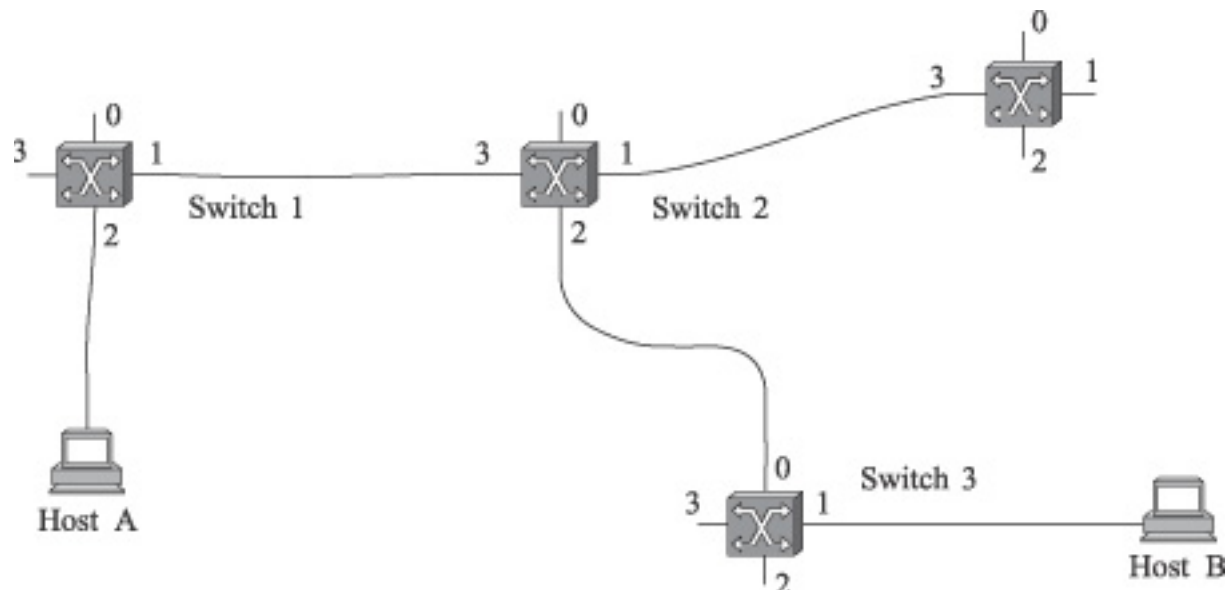
# Switching vs Forwarding: Circuito Virtual

- Establecimiento de la conexión:
  - Establece una entrada en la tabla VC (circuito virtual) de cada switch a través del cual la conexión pasa

# Switching vs Forwarding: Circuito Virtual

- Cada tabla de VC contiene:
  - Un identificador de circuito virtual (VCI) que identifica la conexión en este switch
  - Una interfaz de entrada
  - Una interfaz de salida
  - Un VCI que será usado para los paquetes de salida
- Note que el VCI tiene **alcance local** para ese enlace

# Switching vs Forwarding: Circuito Virtual



Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
2	5	1	11

**Table 3.2 Virtual circuit table entry for switch 1.**

# Switching vs Forwarding: Circuito Virtual - Ejemplo

Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
2	5	1	11

**Table 3.2 Virtual circuit table entry for switch 1.**

Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
3	11	2	7

VC table entry at switch 2

Incoming Interface	Incoming VCI	Outgoing Interface	Outgoing VCI
0	7	1	4

VC table entry at switch 3



# Switching vs Forwarding: Circuito Virtual - Ejemplo

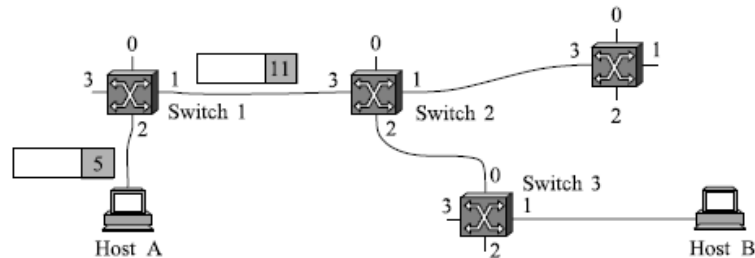


Figure 3.6 A packet is sent into a virtual circuit network.

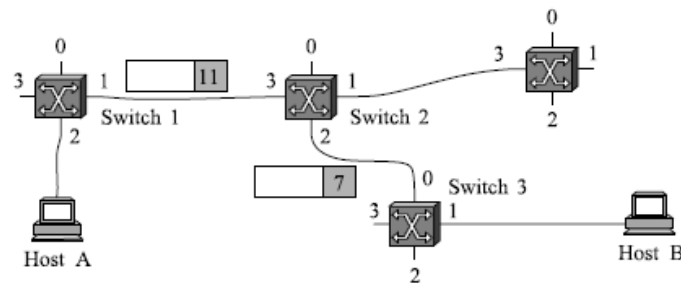


Figure 3.7 A packet makes its way through a virtual circuit network.

# Switching vs Forwarding:

## Circuito Virtual - Tipos

- PVC (Circuitos Virtuales Permanentes)
  - Circuitos de larga vida
  - Administrativamente configurados
- SVC (Circuitos Virtuales Conmutados)
  - La señalización consiste en mensajes de control enviados a la red para establecer un VC
  - Los SVCs puede ser establecidos y borrados sin intervención del administrador
  - Se usa señalización para ello

# Switching vs Forwarding: Circuito Virtual

- Características de los VCs:
  - Ya que el nodo fuente debe esperar a que el requerimiento de conexión alcance el otro lado de la red y se devuelva antes de enviar un paquete de datos, se debe esperar un retardo de al menos de un RTT antes de que se envíe la data
  - Solo el requerimiento de conexión tiene la dirección completa al host destino, pero cada paquete de datos solo contiene un pequeño identificador
    - Se reduce el *overhead* causado por el encabezado

# Switching vs Forwarding: Circuito Virtual

- Características de los VCs:
  - Si un *switch* o enlace se rompe, una nueva conexión debe ser establecida
  - Se requiere una especie de **algoritmo de enrutamiento** para decidir sobre qué enlace enviar el requerimiento de conexión

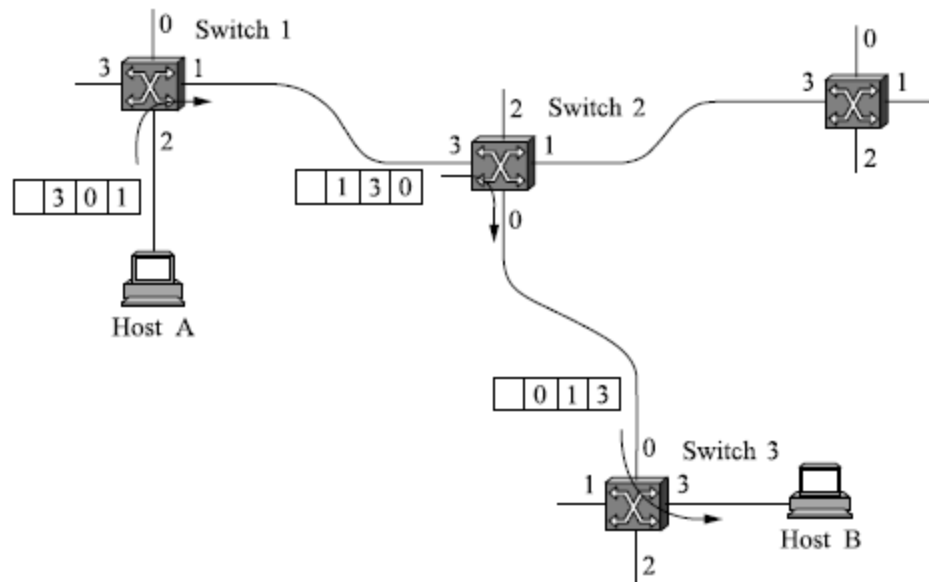
# Switching vs Forwarding: Enrutamiento Fuente

- La información para conmutar un paquete es provista por el nodo fuente
- Hay varias implementaciones:
  - Asignar un número de puerto a cada puerto de salida de un switch
  - El número de cada puerto por el cual debe ser enviado el paquete en su viaje al destino es colocado en la cabecera del paquete antes de salir del nodo fuente

# Switching vs Forwarding: Enrutamiento Fuente

- Hay tres maneras de manejar esta lista de puertos:
- Ordenar la lista
  - Rotarla en cada switch intermedio
  - Eliminar el primer elemento de la lista después de usarlo
    - El destino debe mantener una copia de la lista original, para enviar paquetes en dirección contraria
- Usar un apuntador al próximo puerto a ser usado

# Switching vs Forwarding: Enrutamiento Fuente – Ejemplo Rotación de Puertos

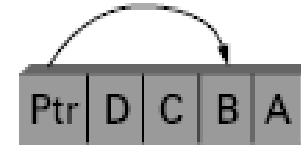


# Switching vs Forwarding: Enrutamiento Fuente – Comparación de Estrategias de Manejo de Encabezados

Header entering switch



Header leaving switch



(a)

(b)

(c)



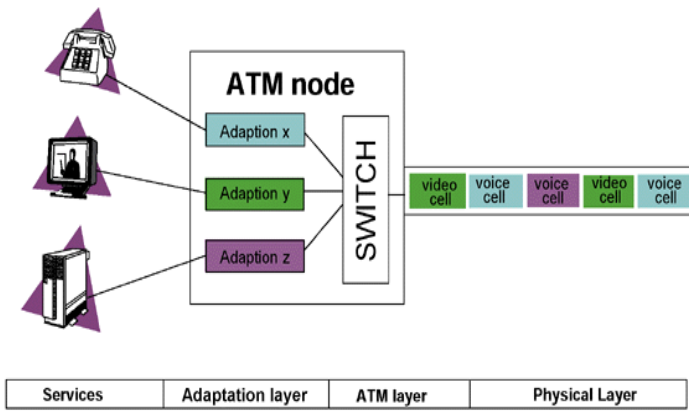
## Switching vs Forwarding: Enrutamiento Fuente - Consideraciones

- ¿Cómo se construye la tabla de enrutamiento en el nodo fuente?
  - El encabezado debe ser lo suficientemente grande para mantener la cantidad máxima puertos para enrutar un paquete entre origen y destino
  - Podría ser de longitud variable

# Switching vs Forwarding: Enrutamiento Fuente

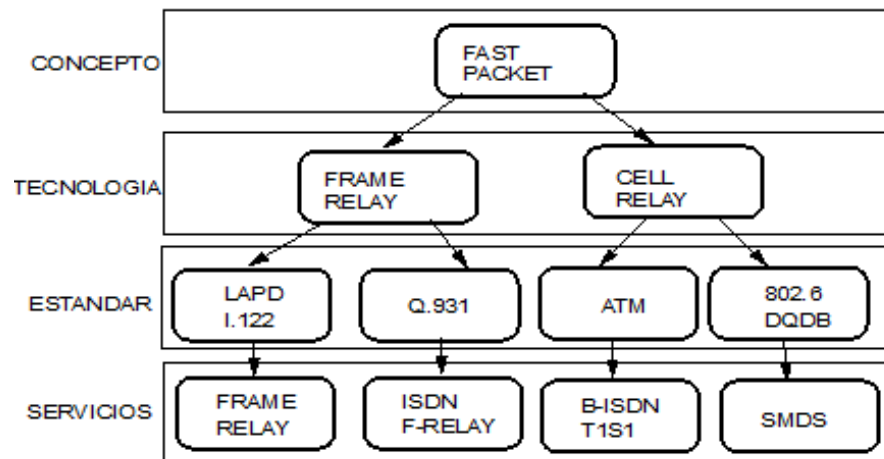
- Clasificación:
  - Estricto (*strict*)
    - Cada nodo a lo largo de la ruta debe ser especificado
  - No estricto (*loose*)
    - Especifica un conjunto de nodos que deben ser visitados

# ATM: Asynchronous Transfer Mode



# Conmutación por Celdas: ATM (Asynchronous Transfer Mode)

- Las redes de conmutación por paquete rápido (fast packet):
  - Se diferencian en el tamaño de las unidades de información transferidas y dónde están colocados los protocolos que se emplean en la red
  - Se dividen en:



## Conmutación por Celdas: ATM (Asynchronous Transfer Mode)

- En *frame relay* las unidades son de longitud variable y se denominan tramas (*frames*)
- En *cell relay* las unidades son de tamaño fijo y se denominan celdas (*cell*)

## Conmutación por Celdas: ATM

- *La Red Digital de Servicios Integrados de Banda Ancha, BISDN( BroadBand Integrated Services Digital Network )* es la propuesta para las redes de área amplia de alta velocidad
- Aunque el concepto de BISDN se venía madurando desde 1985, no fue sino en 1988 que se dan a conocer las primeras recomendaciones definidas por el CCITT (hoy ITU-T)

## Conmutación por Celdas: ATM

- Diseñada para manejar una amplia variedad de servicios de voz, datos, audio, video e imagen sobre facilidades digitales



## Conmutación por Celdas: ATM

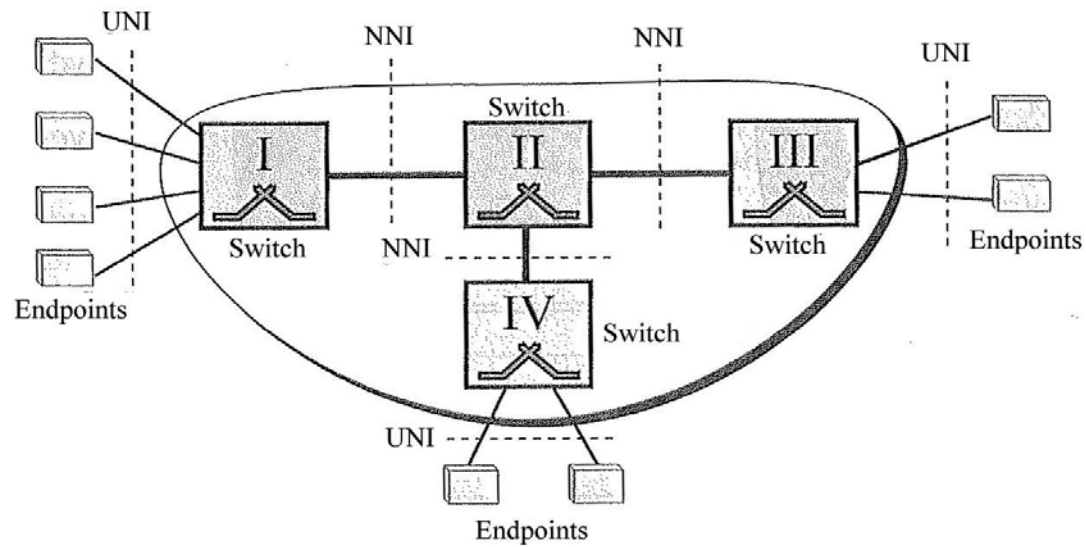
- En las recomendaciones iniciales se adopta el *Modo de Transferencia Asíncrono, ATM (Asynchronous Transfer Mode)*, como base de BISDN
- BISDN no evolucionó
- ATM quedó como una tecnología independiente



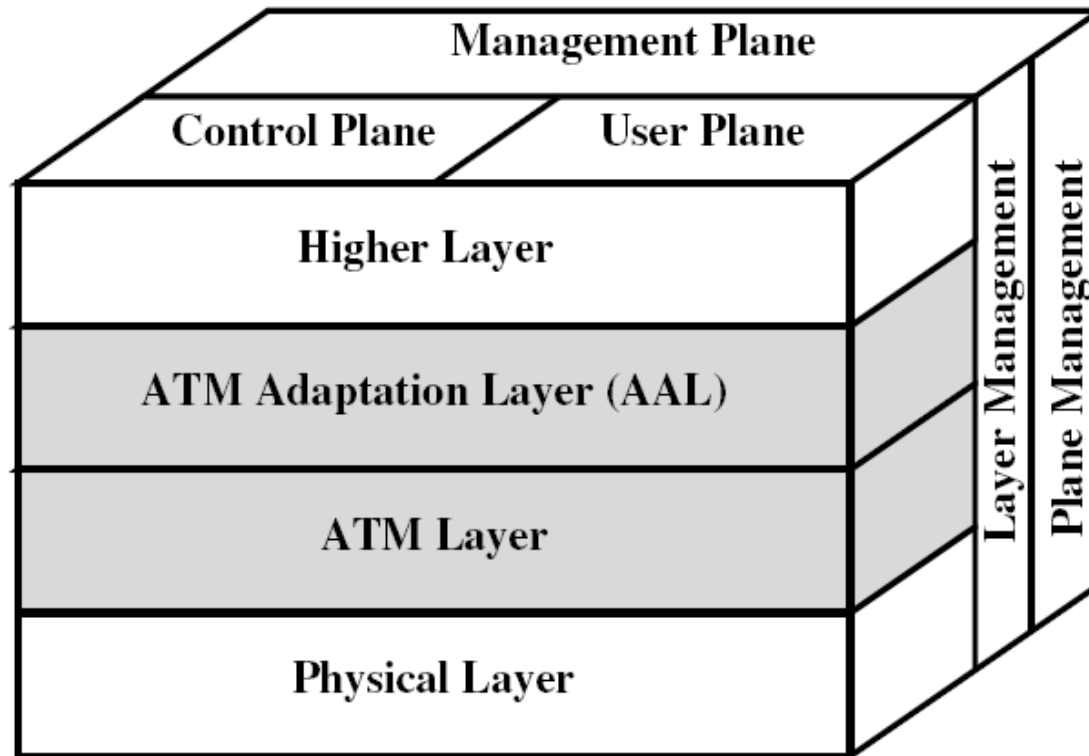
## ATM: Interfaces

- UNI (User Network Interface)
  - Define los límites regulares entre el usuario y la red
  - Ejemplo: interfaz compañía de teléfono y usuarios
- NNI (Network Node Interface)
  - Es el límite entre la facilidad de transmisión y el nodo de la red
  - Ejemplo: interfaz entre un par de compañías telefónicas

## ATM: Interfaces



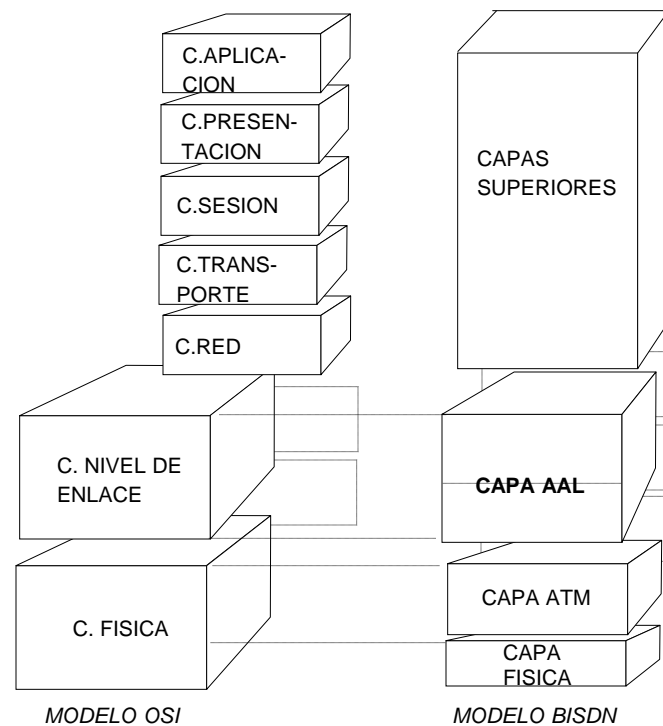
# ATM: Modelo de Referencia



# ATM: Modelo de Referencia

- Se definen los siguientes planos:
  - **Plano de Usuario:** proporciona la transferencia de información del usuario
  - **Plano de Control:** proporciona transferencia de información de señalización de control
  - **Plano de Gestión:** incluye funciones de gestión de la capa y gestión del plano
    - La primera ejecuta las funciones relacionadas a los recursos y parámetros que residen en las entidades de protocolo
    - La segunda ejecuta funciones de gestión relacionadas al sistema como un todo, y proporciona coordinación entre los planos

# ATM: Modelo de Referencia



# ATM: Modelo de Referencia

- ATM emplea paquetes de tamaño pequeño y fijo, denominados *celdas* (*cells*).
- La longitud de la celda es de 53 bytes:
  - 5 bytes de encabezado
  - 48 bytes de payload

# ATM

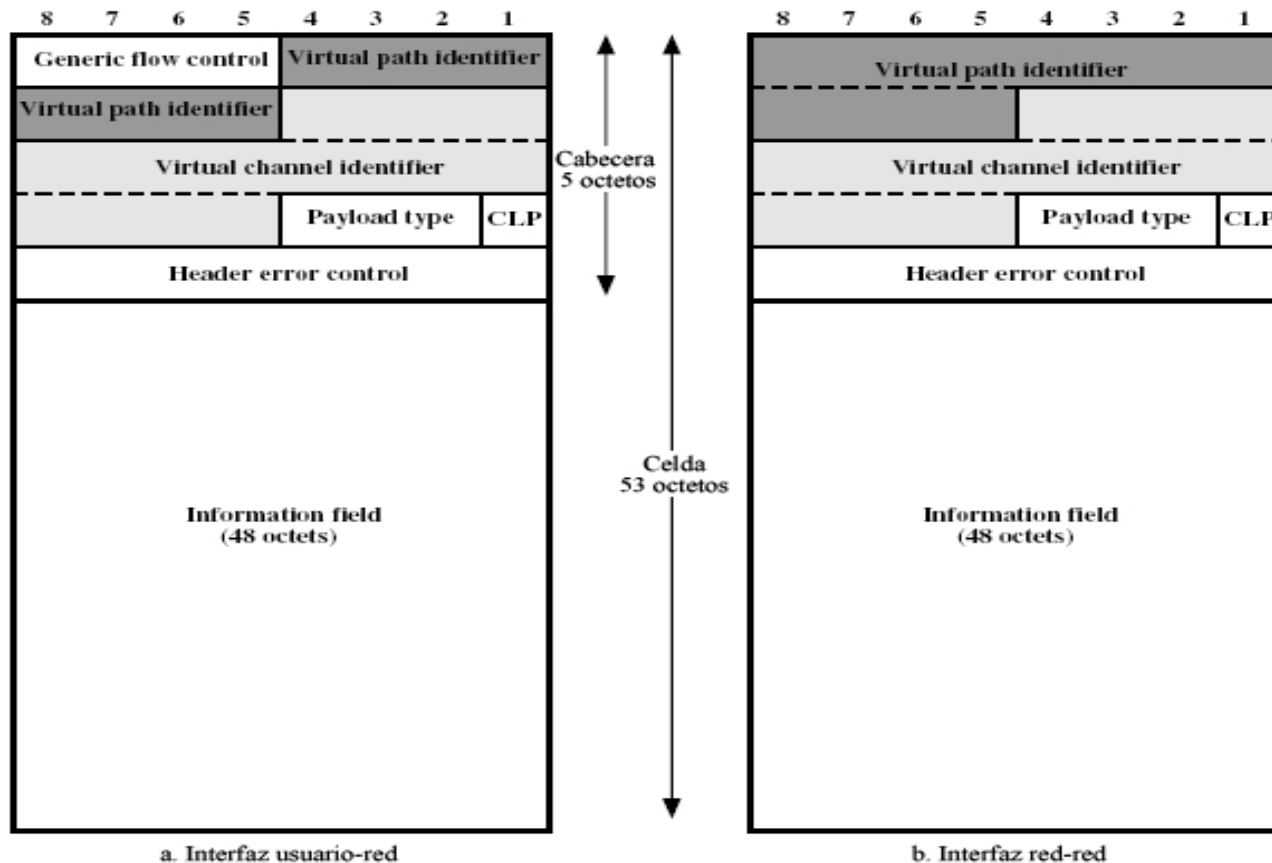
- ◉ ¿Por qué usar paquetes de longitud fija?
- ◉ Fácil de implementar en hardware switches:
  - ◉ Se pueden construir switches rápidos y escalables
  - ◉ Es fácil construir *switches* que hacen trabajos simples
  - ◉ El elemento que maneja el próximo paquete de la cola a ser transmitido no está ocupado por más del tiempo que se tarda en transmitir una celda

# ATM

- ◉ Colas más cortas significa menos retardo para todos los tráficos
- ◉ ¿Cuál es el tamaño correcto para una celda?
  - ◉ Tamaño encabezado vs tamaño payload
  - ◉ Si la celda es muy grande, necesidad de relleno cuando se envía poca data
  - ◉ Número de celdas por segundo cae proporcionalmente al tamaño de la celda



# ATM: Formato de la Celda



# ATM: Formato de la Celda

- GFC (Generic Flow Control ): usado para asistir al cliente en el control del tráfico para diferentes calidades de servicio
- Campos de enrutamiento: el VPI (*Virtual Path Identifier*) y el VCI (*Virtual Channel Identifier*) constituyen los campos para el control de enrutamiento de las celdas
  - El VPI provee una identificación explícita del camino (o ruta) para la celda
  - El VCI provee una identificación explícita del canal

# ATM: Formato de la Celda

- **PT (Payload Type):** indica el tipo de información en el campo de payload
  - El primer bit indica si la celda transporta información del usuario (0) o información de mantenimiento y gestión (1)
  - Si el primer bit es 0, el segundo bit indica si se ha presentado congestión o no, y el tercero es el bit de señalización del usuario
  - El tercer bit es usado con la AAL 5 para delimitar las tramas

# ATM: Formato de la Celda

- CLP (Cell Loss Priority):
  - Un valor de 0 indica una celda de más alta prioridad, la cual podría no ser descartada
  - Un valor de 1 indica que la celda puede ser descartada
- HEC (Header Error Control): es un código de error de 8 bits que puede ser usado para corregir errores de un bit en el encabezado y detectar un número mayor de errores
  - CRC
  - $X^8 + X^2 + X + 1$

# ATM: Formato de la Celda

- **Campo de Información:** contiene información de diversos tipos, tales como datos, imágenes, mensajes de señalización y de mantenimiento

# ATM: Funciones

- ◉ Multiplexación y demultiplexación de las celdas
- ◉ Traducción de VPI y de VCI
- ◉ Generación de los encabezados de las celdas
- ◉ Control de flujo

# ATM: Conexiones

- Tres conceptos importantes en ATM son los relacionados con las conexiones:
  - TP (Transmission Path)
  - VP (Virtual Path)
  - VC (Virtual Channel)

# ATM: Conexiones

- Las conexiones entre dos extremos son logradas a través de un TP, que está constituido por la ruta física de transmisión entre el endpoint y el switch o entre dos switches



# ATM: Conexiones

- Un VC se define como un flujo unidireccional de celdas ATM con el mismo identificador entre un switch y un endpoint, o entre dos switches
- Una VCC se define como una lista de VCs concatenados
- Una VCC define un flujo unidireccional de celdas ATM de un usuario a uno o más usuarios

# ATM: Conexiones

- Un identificador de VC (VCI) es un identificador de un enlace VC, este último existe entre el punto donde se asigna el VCI y el punto donde se traduce este valor o en el punto final de la conexión

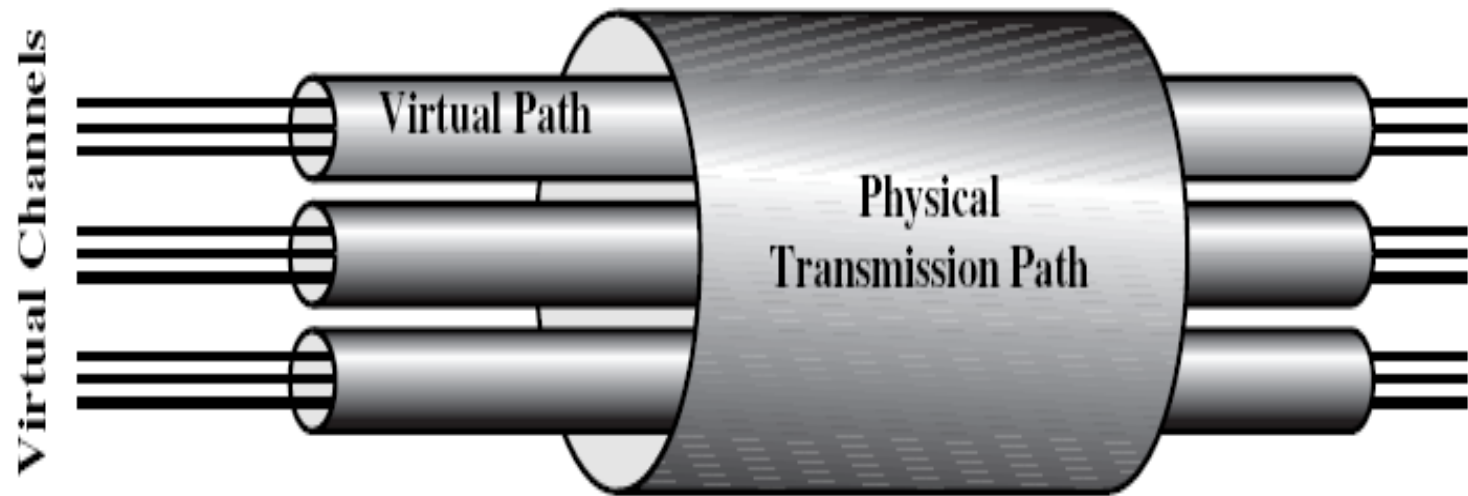
# ATM: Conexiones

- Un VP se define como la agrupación de VCs, y provee la transferencia unidireccional de celdas con un mismo identificador entre un switch y un endpoint, o entre dos switches
- Una VPC se define como una lista de VPs concatenados
- Una VPC define un flujo unidireccional de celdas ATM de un usuario a uno o más usuarios

# ATM: Conexiones

- Un identificador de VP (VPI) identifica un enlace de VP
- La comunicación es identificada por un número lógico formado por un el VPI y VCI

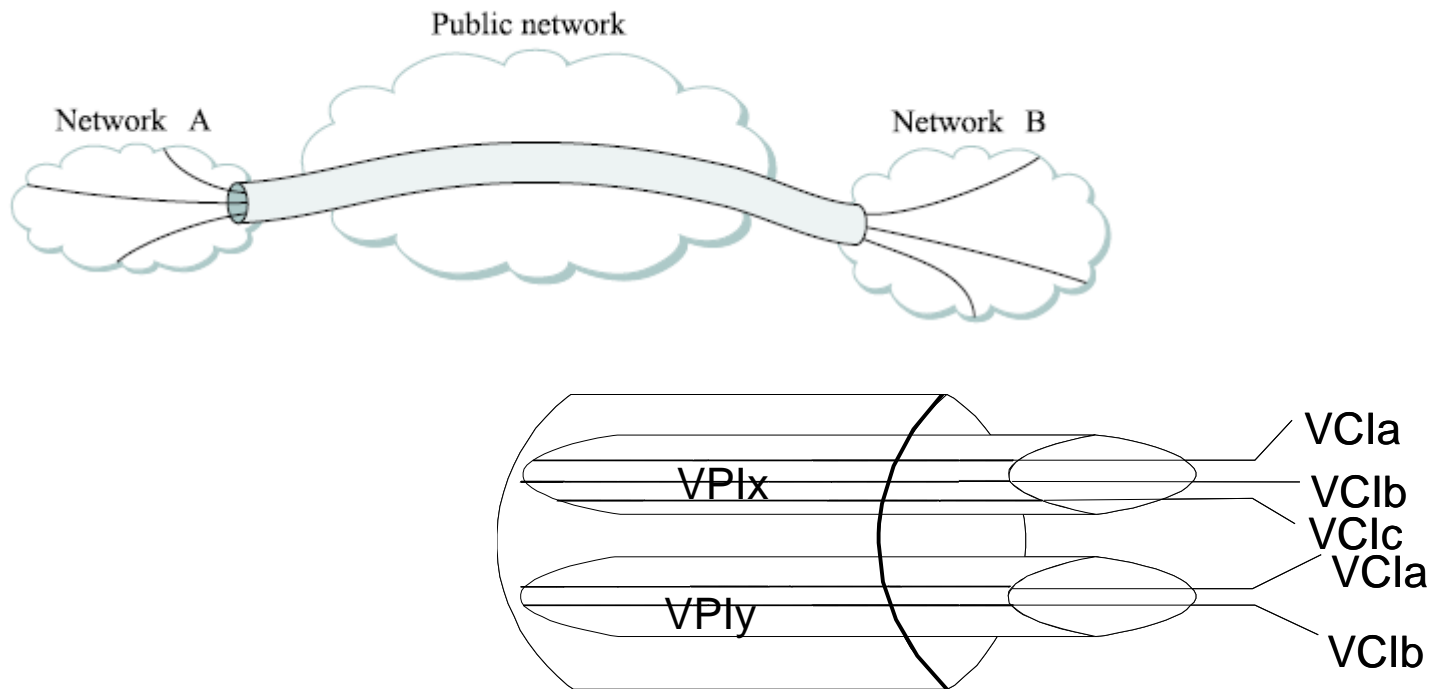
# ATM: Conexiones



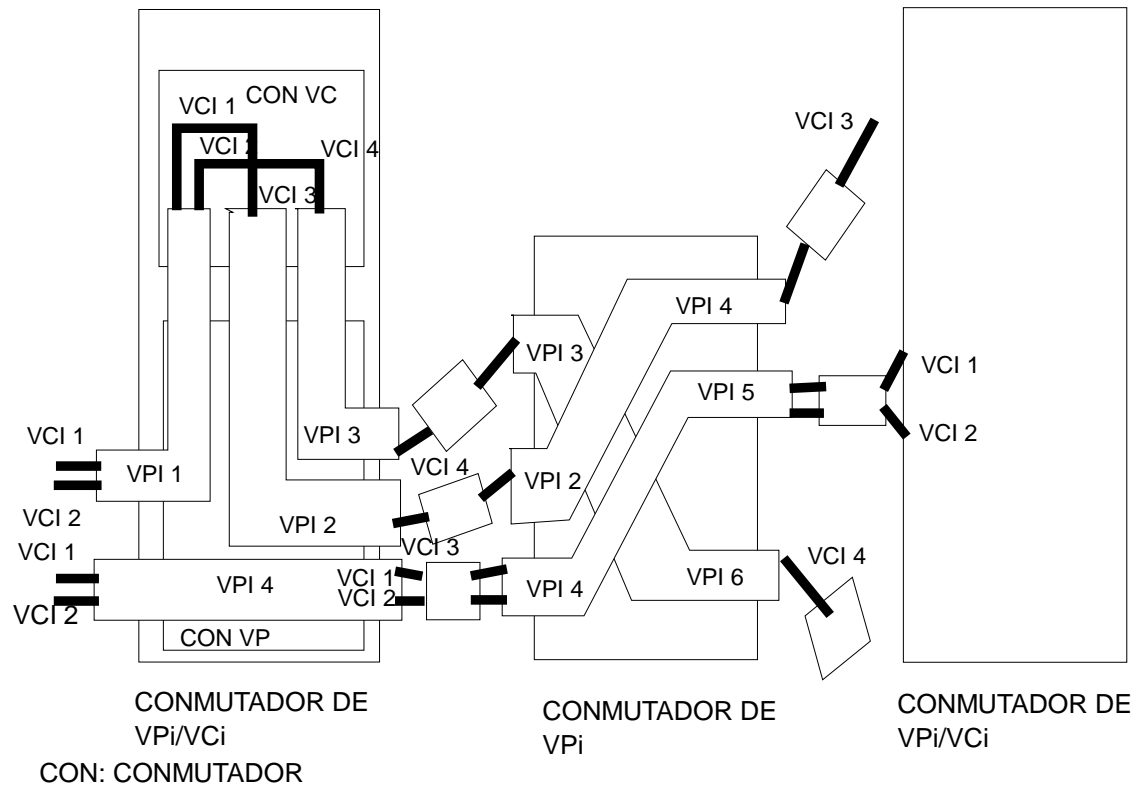
## ATM: Conexiones

- Un TP está dividido en varios VPs
- Un VP provee una conexión o un conjunto de conexiones entre dos switches
- Se puede pensar en el VP como una autopista que conecta dos ciudades
- Cada autopista es un VP; el conjunto de las autopistas es el TP, y los canales de la autopista son los VCs

# ATM: Conexiones

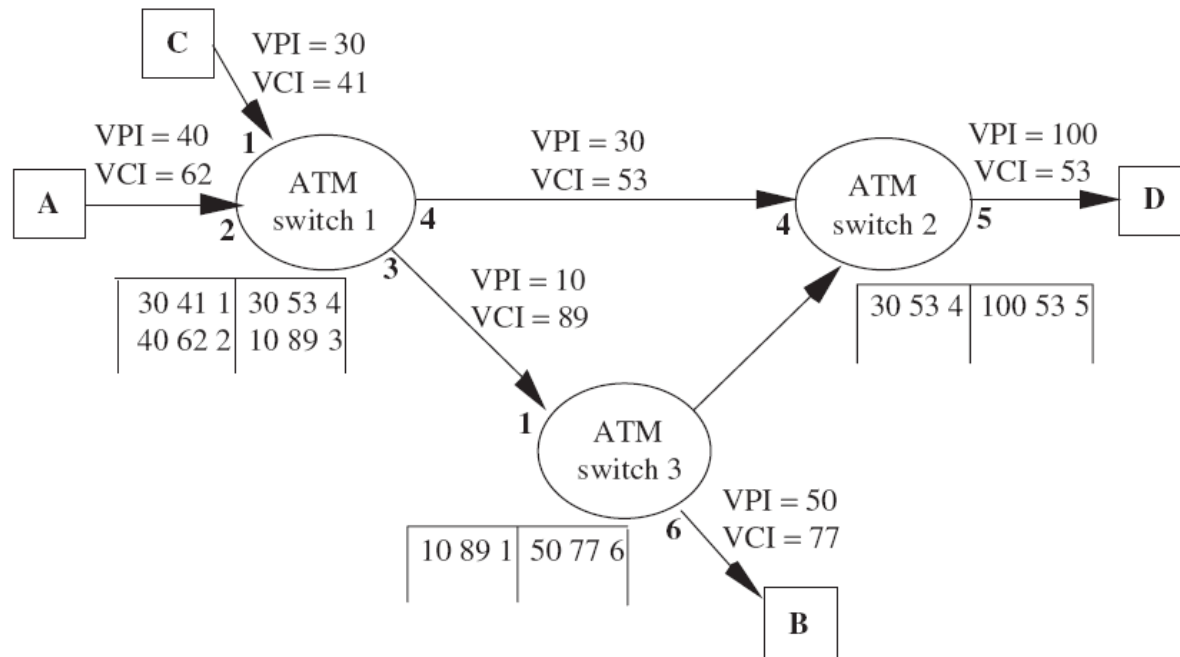


# ATM: Conexiones





# ATM: Conexiones



# ATM: Capa Física

- ATM sobre SONET/SDH:
- Se usa la trama STS-3 con una velocidad de transmisión de 155,520 MbPS
- Las celdas ATM viajan en una trama STS-3, transportadas en la porción de carga de la trama SDH

## ATM: Capa Física

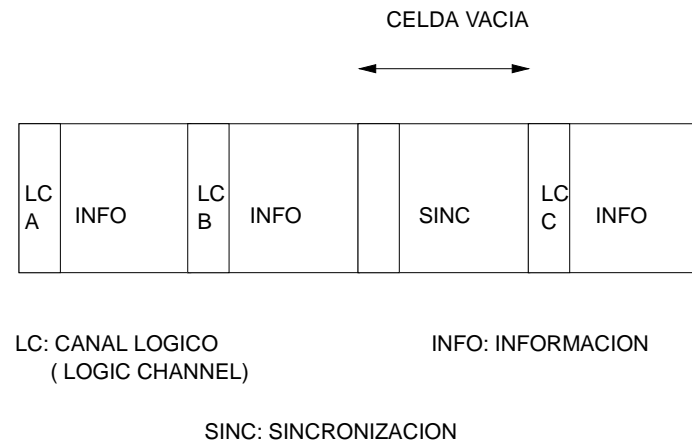
- La carga puede estar desplazada de la trama, ya que la capacidad de carga de la trama SDH es de 2340 bytes, que no es múltiplo de 53 (longitud de la celda ATM)

# ATM: Capa Física

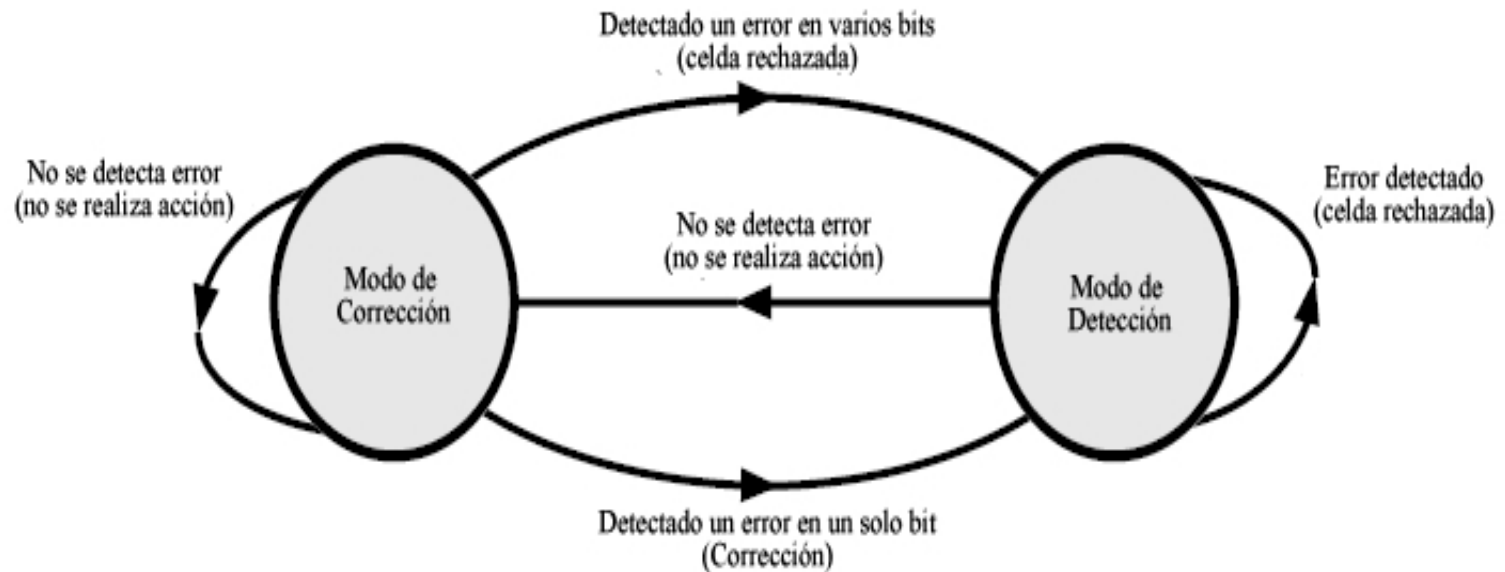
	DESIGNACIÓN SDH	DESIGNACIÓN CCITT	TASA DE BITS ( Mbps )
OC-1	STS-1		51,84
OC-3	STS-3	STM-1	155,52
OC-9	STS-4	STM-3	466,56
OC-12	STS-12	STM4	622,08
OC-18	STS-18	STM-6	933,12
OC-24	STS-24	STM-8	1244,16
OC-36	STS-36	STM-12	1866,24
OC-48	STS-16	STM-16	2488,32

# ATM: Capa Física

- La *División de Tiempo Asíncrono, ATD (Asynchronous Time División)* es la especificación de la capa física que consiste en la transmisión de celdas ATM, es decir, de un flujo continuo de 53 bytes



# ATM: Capa Física



# ATM: Capa Física

- El proceso de detección y corrección de errores usado es el *CRC (Cyclic Redundancy Check)*, con el polinomio  $x^8+x^2+x+1$
- Este código es usado solamente para detectar errores en los 32 bits del encabezado, a través del campo HEC del encabezado de la celda (8 bits)
- En vista de que se está usando un polinomio de grado 8 para detectar errores en 32 bits (correspondientes a los 4 bytes restantes del encabezado), existen suficientes bits de redundancia para corregir algunos errores

# ATM: Capa Física

- Algoritmo para corregir y/o detectar errores:
  - En un inicio el receptor está en *modo de corrección*
  - Cada vez que llega una celda se calcula el HEC y se realiza la comparación con lo recibido
  - En la medida que no existan errores el receptor permanece en el modo de corrección
  - Si se consigue un error, el receptor corrige el error y pasa al *modo de detección*
  - Si se detecta más de un error se pasa al *modo de detección*
  - El receptor permanece en este estado mientras se encuentren celdas con error en su encabezado, con la idea de detectar ráfagas de error que hayan afectados varias celdas



## ATM: AAL

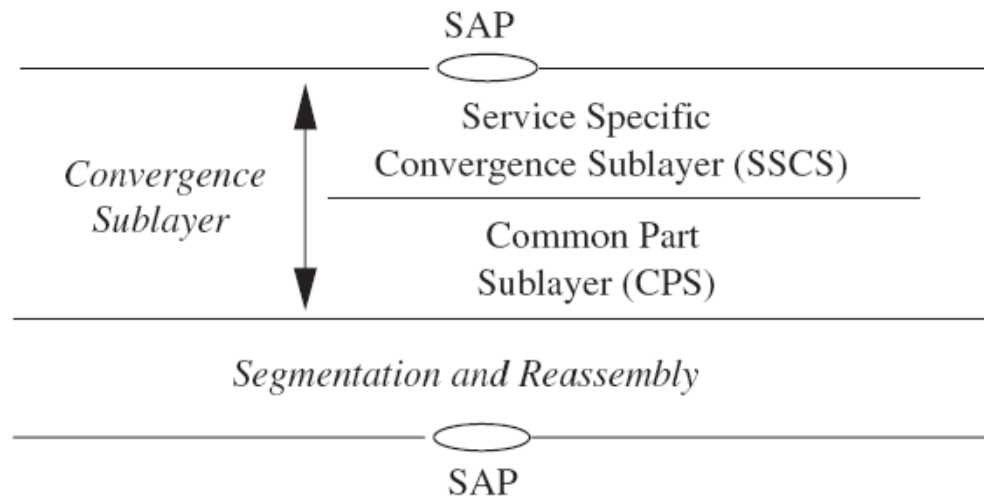
- ◉ Diseñada para permitir:
  - ◉ ATM debe aceptar cualquier tipo de tráfico (frames y stream)
  - ◉ ATM debe transportar datos multimedia

## ATM: AAL

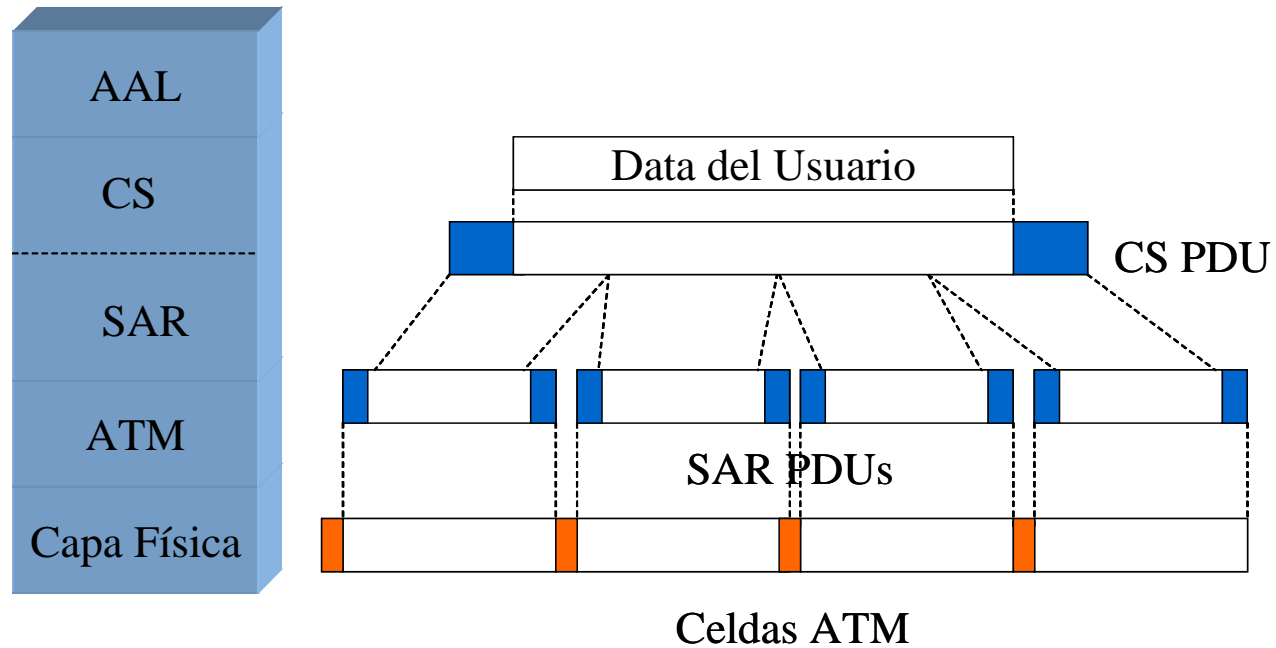
- Las funciones principales de la AAL son:
  - Manejo de los errores de transmisión
  - Segmentación y re-ensamblado
  - Manejo de celdas perdidas o mal insertadas
  - Control de flujo y de temporización

# ATM: AAL

- La AAL está dividida en dos subcapas:
  - SAR (Segmentation And Reassembly sublayer)
  - CS (Convergence Sublayer)



# ATM: AAL



## ATM: AAL

- Hay 4 tipos de AAL: AAL 1, AAL 2, AAL  $\frac{3}{4}$ , AAL 5
- Estas AALs soportan distintas clases de servicio los cuales se agrupan en función de las siguientes características:
  - Relación de temporización entre fuente y destino
  - Tasa de bit
  - Modo de conexión
- Comunes hoy en día: AAL 1 y AAL 5

# ATM: AAL

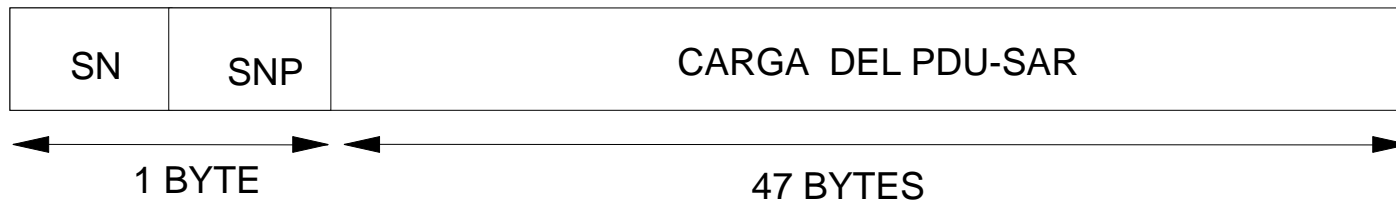
Modo de Conexión	Orientado a conexión			No Orientado a conexión
Velocidad de bit	Constante	Variable		
Relación de temporización	Requerido		No requerido	
AAL	1	2	3/4, 5	3/4, 5

# ATM: AAL1

- Está orientada a fuentes de tasa de bits constante (CBR), tales como emulación de circuitos y voz sobre ATM
- Los siguientes campos se encuentran en el AAL 1 SAR PDU:
  - SN (Sequence Number): 4 bits, usado para detectar celdas perdidas o erróneamente insertadas en la secuencia
  - SNP (Sequence Number Protection): 4 bits, puede ser usado para proveer capacidades de corrección y detección de errores para el campo de número de secuencia

# ATM: AAL1

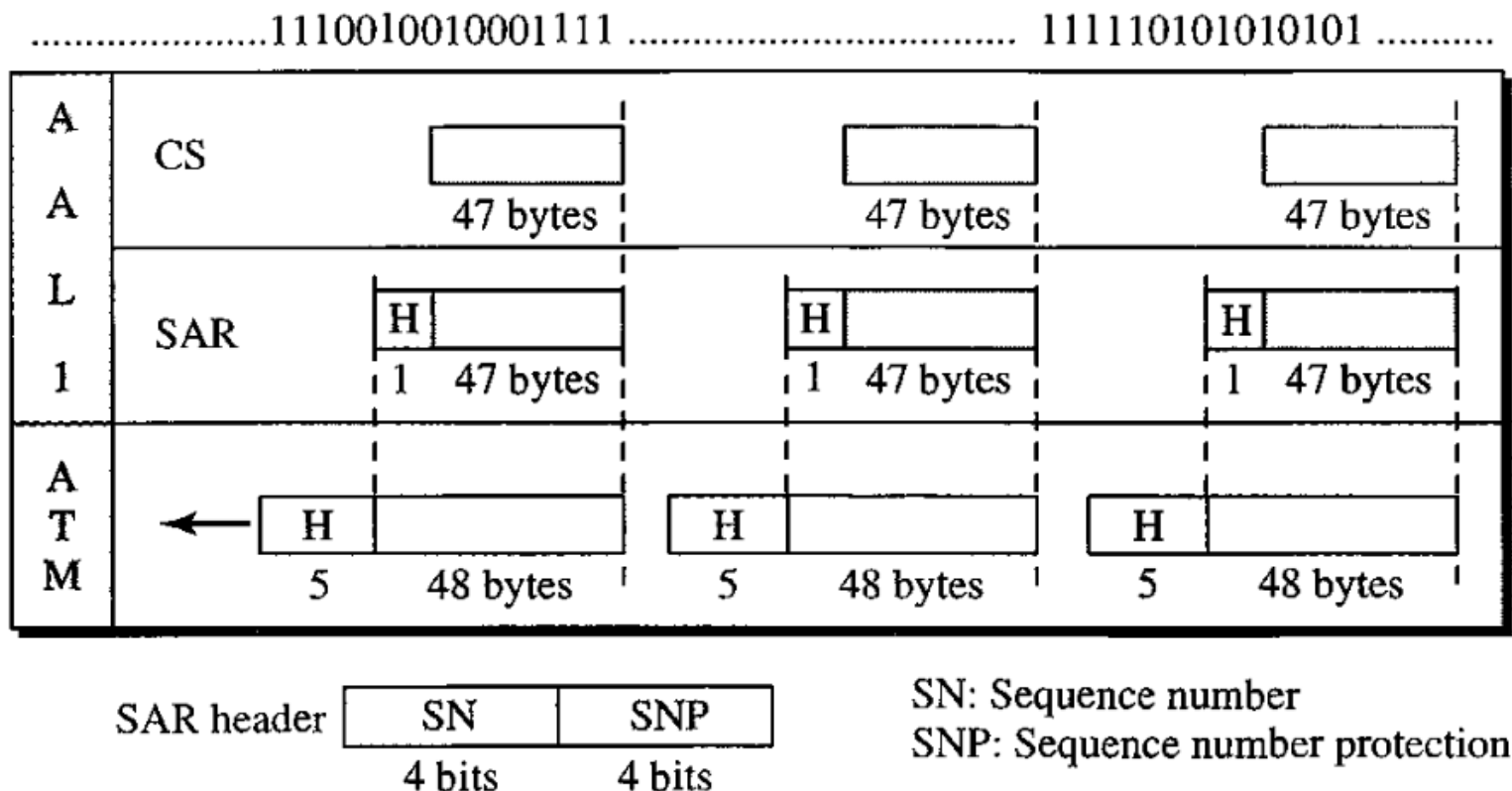
*AAL TIPO 1*





# ATM: AAL1

Constant-bit-rate data from upper layer



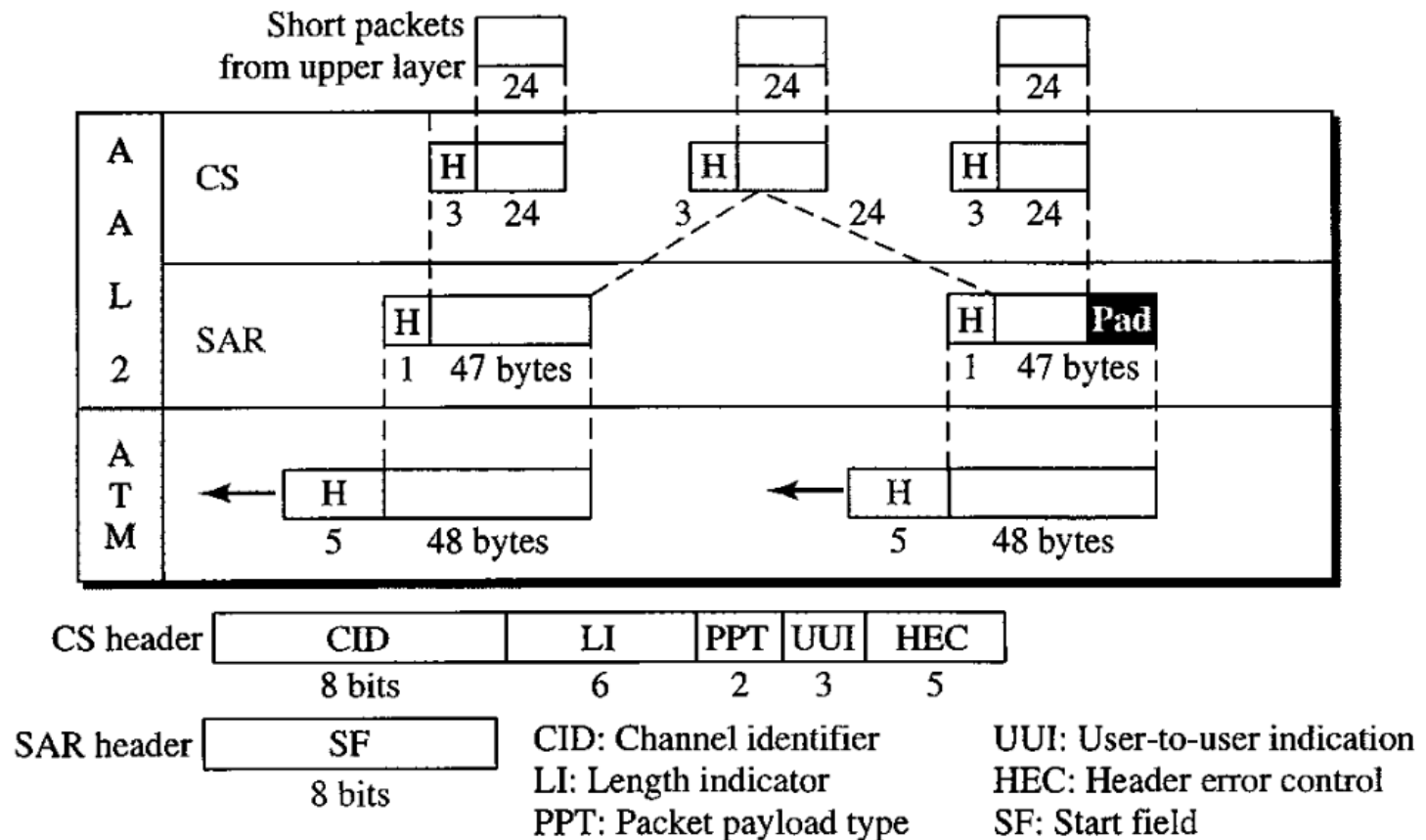
# ATM: AAL1

- ◉ AAL 1 CS
  - ◉ Manejo de la variación del retardo de la celda
  - ◉ Procesamiento del contador de secuencia
  - ◉ Manejo del monitoreo del rendimiento
  - ◉ Transferencia de data estructurada y no estructurada
  - ◉ Transferencia de información de temporización

## ATM: AAL2

- Transporta aplicaciones sensitivas al retardo y con baja tasa de bit variable (VBR), por ejemplo, voz y fax
- Actualmente usada para tráfico *low-bit rate*, por ejemplo audio (comprimido o no) o telefonía móvil
- Permite la multiplexación de varias tramas pequeñas en una celda

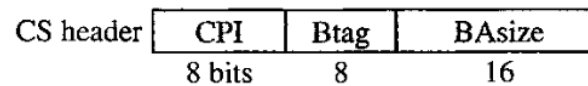
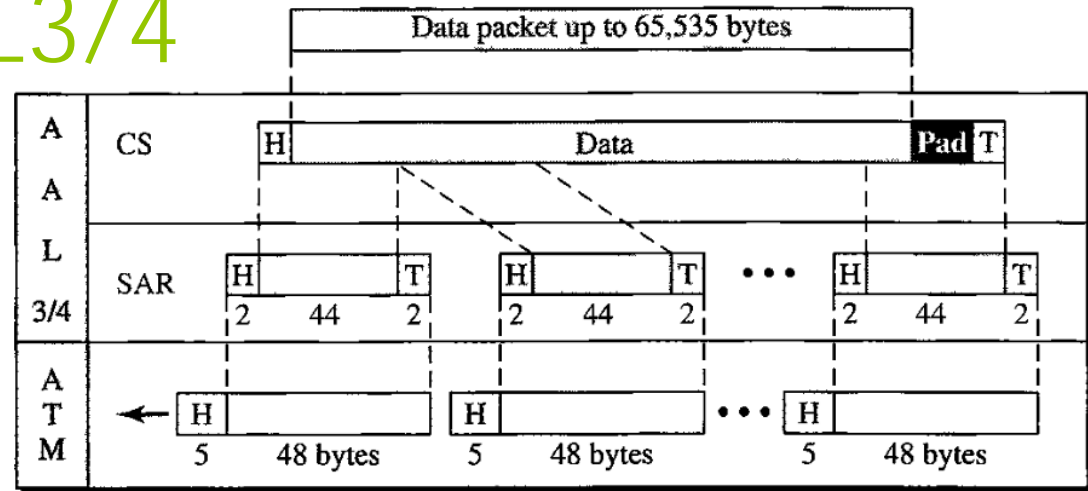
# ATM: AAL2



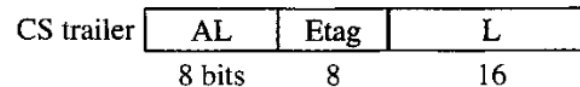
## ATM: AAL3/4

- AAL 3 inicialmente pensada para servicios de datos orientado a conexión
- AAL 4 inicialmente pensada para servicios no orientados a conexión
- Se hizo evidente que las tareas fundamentales eran las mismas

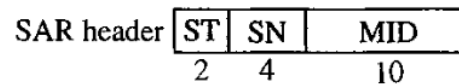
## ATM: AAL3/4



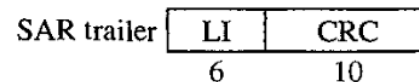
CPI: Common part identifier  
 Btag: Beginning tag  
 BAsize: Buffer allocation size



AL: Alignment  
 Etag: Ending tag  
 L: Length



ST: Segment type  
 SN: Sequence number  
 MID: Multiplexing identifier



LI: Length identifier  
 CRC: Error detector

# ATM: AAL3/4

- Está orientada a la información con tasa de bit variable (VBR)
- Los tipos de servicio que proporciona esta AAL son
  - Servicio de modo-mensaje: usado para dividir en tramas los datos. Un bloque de datos simple proveniente de niveles superiores al AAL es transferido en una o más celdas. Es usado para servicios como Frame Relay y LAPD
  - Servicio de modo-streaming: los datos son presentados a la AAL en bloques de tamaño fijo, y ellos son transferidos por celdas. Esta AAL soporta transferencia de información a baja velocidad

# Conmutación por Celdas: ATM AAL3/4



CS-PDU

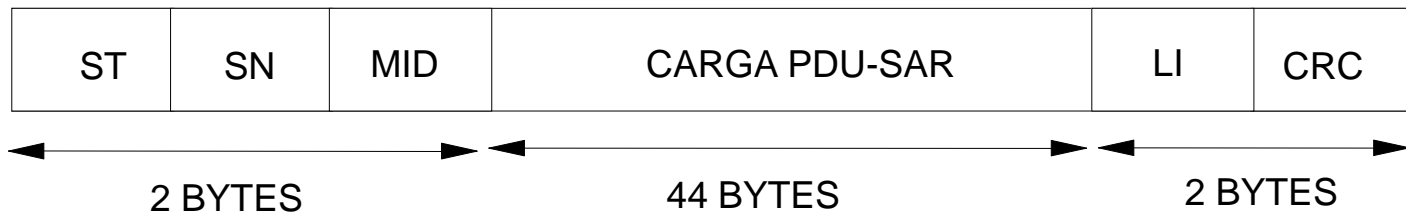


# Conmutación por Celdas: ATM AAL3/4

- Common part indicator (CPI)
- Beginning tag (Btag)
- End tag (Etag)
- Buffer allocation size (BASize)
- Pad
- Length (Len)

# Conmutación por Celdas: ATM AAL3/4

*AAL TIPO 3/4*



SAR-PDU

# ATM: AAL3/4

- Los siguientes campos forman parte de la AAL 3/4 SAR PDU (ver lámina de SAR PDUs):
  - CRC (Cyclic Redundancy Check): 10 bits, usados para detectar errores y corregir hasta dos bits correlacionados errados en el SAR-PDU. El polinomio es:  $X^{10} + X^9 + X^5 + X^4 + X + 1$
  - LI (Length Indicator): 6 bits, indica el número de bytes del CS-PDU que son incluidos en el SAR-PDU
  - MID (Multiplexing Identification): 9 bits, es usado para la multiplexación y demultiplexación de múltiples CS-PDUs sobre una conexión ATM simple. Todos los SAR-PDUs asociados a los CS-PDUs tienen el mismo MID
-

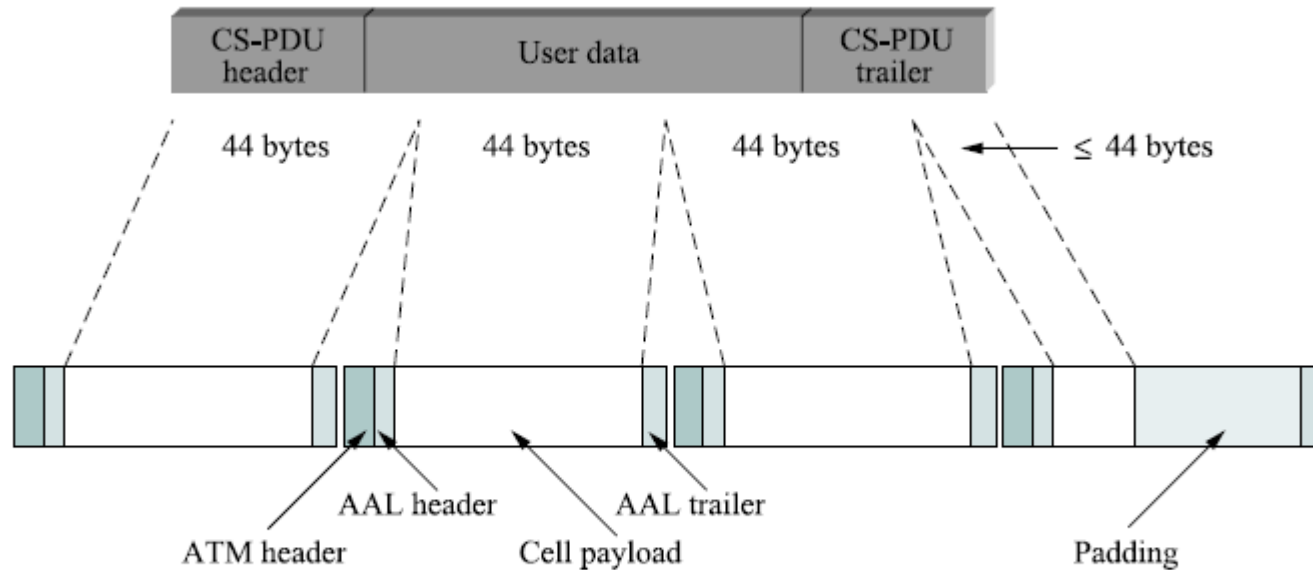
# ATM: AAL3/4

- ST (Segment Type): 2 bits, es usado para indicar el comienzo, continuación o fin de un mensaje, o si se trata de un mensaje simple.
- SN (Sequence Number): 4 bits, usado para detectar celdas perdidas o erróneamente insertadas en la secuencia.

Value	Name	Meaning
10	BOM	Beginning of message
00	COM	Continuation of message
01	EOM	End of message
11	SSM	Single-segment message

**Table 3.5 AAL3/4 Type field.**

# ATM: AAL3/4

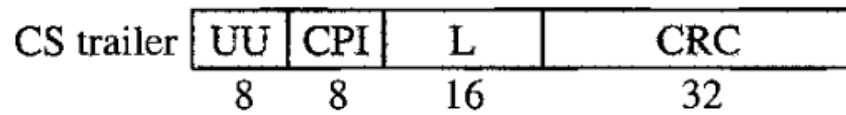
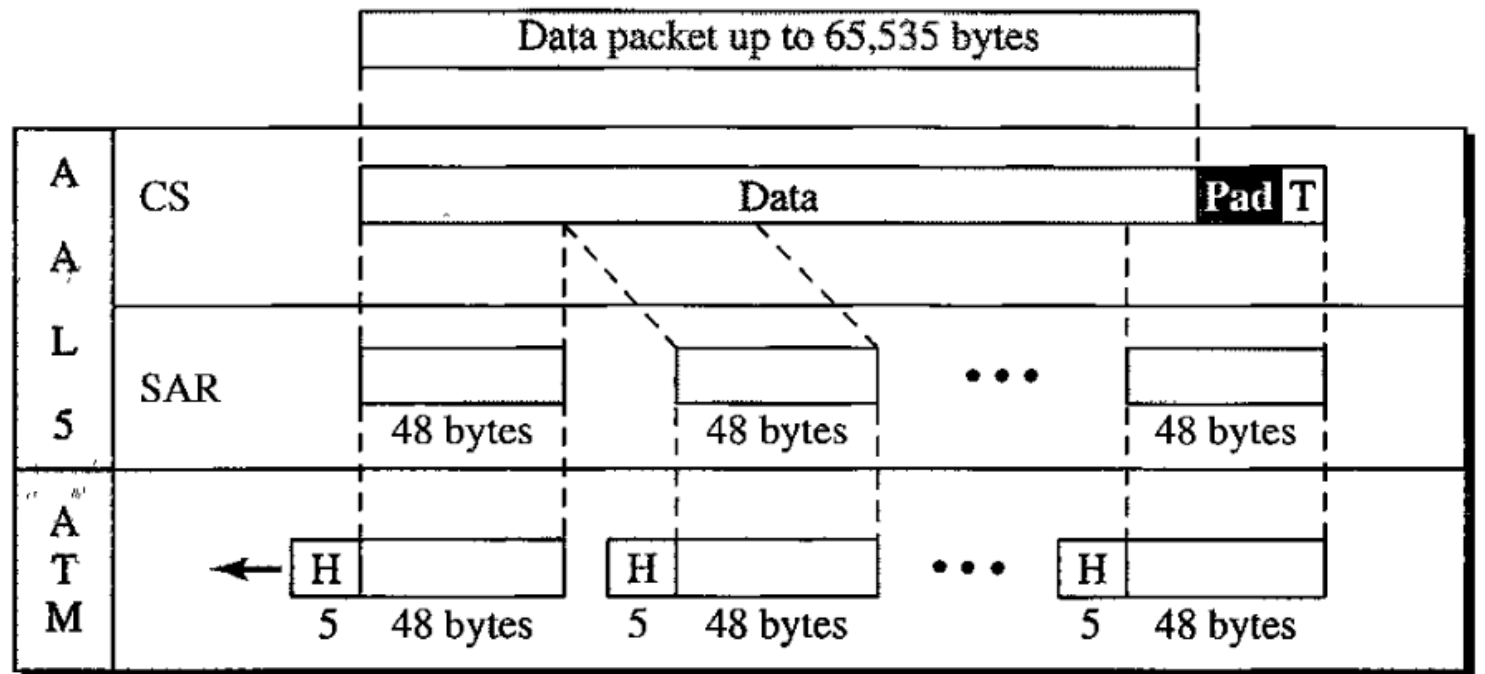


**Figure 3.20 Encapsulation and segmentation for AAL3/4.**

## ATM: AAL5

- La AAL 3/4 provee mecanismos exhaustivos para manejo de errores y secuencia no necesarios para todas las aplicaciones
- SEAL (Simple and Efficient Adaptation Layer)
  - Posteriormente AAL 5
- Asume que las celdas que pertenecen a un mismo mensaje viajan secuencialmente
  - Funciones de control son incluidas en las capas superiores

# ATM: AAL5



UU: Channel identifier  
 CPI: Common part identifier  
 L: Length  
 CRC: Error detector

## ATM: AAL5

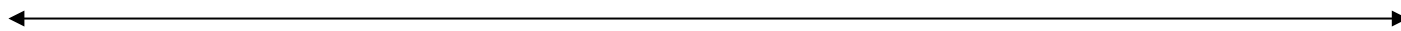
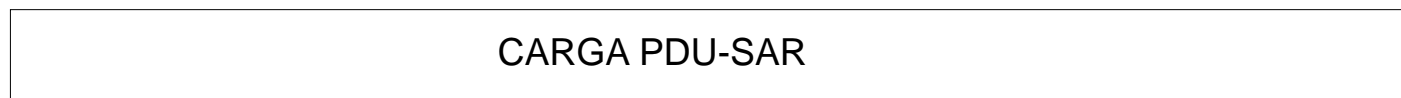
- UU (User-to-User): Usado por usuarios finales
- CPI (Common Part Identifier): Define cómo se interpretan los campos subsiguientes
- L (Length): Longitud del mensaje original
- CRC (Cyclic Redundancy Check): Control de errores, 4 bytes



# ATM: AAL5

- Tiene como propósito:
  - Reducir el overhead de procesamiento envuelto en el protocolo
  - Reduce el overhead de transmisión
- Asegurar adaptabilidad para los protocolos de transporte existentes
- La SAR PDU solo contiene información

## AAL TIPO 5



48 BYTES

# Control de Tráfico y Congestión en ATM

- Control de Tráfico y Congestión es requerido para mantener los niveles de servicio (QoS) en la red
- La *congestión* se define como un estado de los elementos de la red (conmutadores, multiplexores, otros) en el cual estos no son capaces de cumplir con los objetivos de rendimiento previstos para las conexiones establecidas o por establecerse
- El *control de tráfico* se refiere al conjunto de acciones tomadas por la red para impedir condiciones de congestión. El *control de congestión* se refiere al conjunto de acciones tomadas por la red para minimizar la intensidad, difusión y duración de la congestión

# Control de Tráfico y Congestión en ATM

- Las Clases de Servicio definidas por el ATM Forum:
  - Tasa de Bit Constante (CBR)
  - Tasa de Bit Variable-Tiempo Real (rtVBR)
  - Tasa de Bit Variable-Tiempo No-Real (nrtVBR)
  - Tasa de Bit Disponible (ABR)
  - Tasa de Bit No Especificada (UBR)

# Control de Tráfico y Congestión en ATM

- Los **parámetros de tráfico** describen las características de tráfico de una conexión ATM, tales como: tiempo de sostenimiento de la conexión, tasa pico de las celdas, duración promedio de una ráfaga, etc.
- Un parámetro de tráfico describe un aspecto cuantitativo o cualitativo del tráfico

# Control de Tráfico y Congestión en ATM

- Por otra parte, un **descriptor de tráfico** es una lista de parámetros de tráfico usados para capturar las características intrínsecas del tráfico de una conexión ATM
- El descriptor de tráfico fuente, en particular, es el conjunto de parámetros pertenecientes al descriptor de tráfico ATM usado durante la fase de establecimiento de la conexión, para especificar las características de tráfico requeridas para la conexión por la fuente

# Control de Tráfico y Congestión en ATM

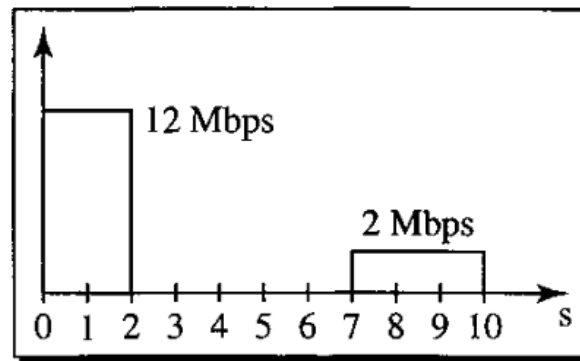
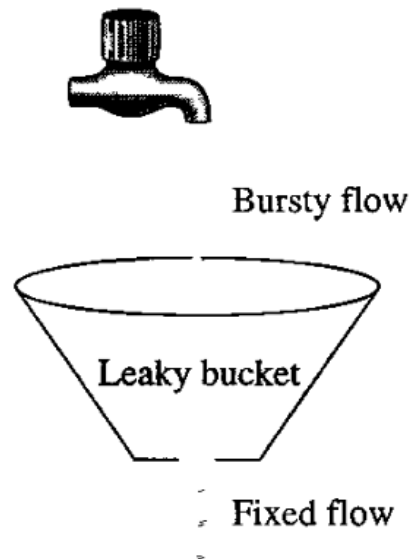
- Lo siguientes parámetros se han definidos:
  - Tasa de Celda Sostenible (SCR: Sustainable Cell Rate)
  - Tasa de la Celda Pico (PCR: Peak Cell Rate)
  - Tolerancia de la Ráfaga (BT: Burst Tolerance):
  - Variación del retardo de la celda (CDV: Cell Delay Variation)
  - Retardo máximo de transferencia de la celda (Max CTD: Maximum Cell Transfer Delay) y Retardo de transferencia de la celda medio (Mean CTD: Mean Cell Transfer Delay)
  - Tasa de error de la celda (CER: Cell Error Ratio)
  - Tasa de pérdida de la celda (CLR: Cell Loss Ratio)
  - Tasa de bloque de celda errada severamente (SECBR: Severely Errored Cell Block Ratio)
  - Tasa de celdas mal insertadas (CMR: Cell Misinsertion Rate)

# Control de Tráfico y Congestión en ATM

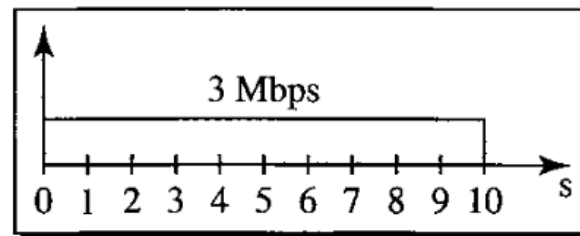
- La Recomendación I.371 de la ITU (ITU-I.371, 1994) define las siguientes funciones que pueden ser usadas independientemente o en forma combinada para manejar y controlar el tráfico y la congestión:
  - Manejo de los Recursos de la Red (NRM: Network Resource Management)
  - Control de Admisión de la Conexión (CAC: Conexion Admission Control)
  - Control de los Parámetros de Usuario (UPC: Usage Parameter Control)
  - Descarte Selectivo de la Celda
  - *Shaping* del Tráfico
  - Control de Prioridad

# Control de Tráfico y Congestión en ATM

Shaping del Tráfico: Funcionamiento de Leaky Bucket



Bursty data

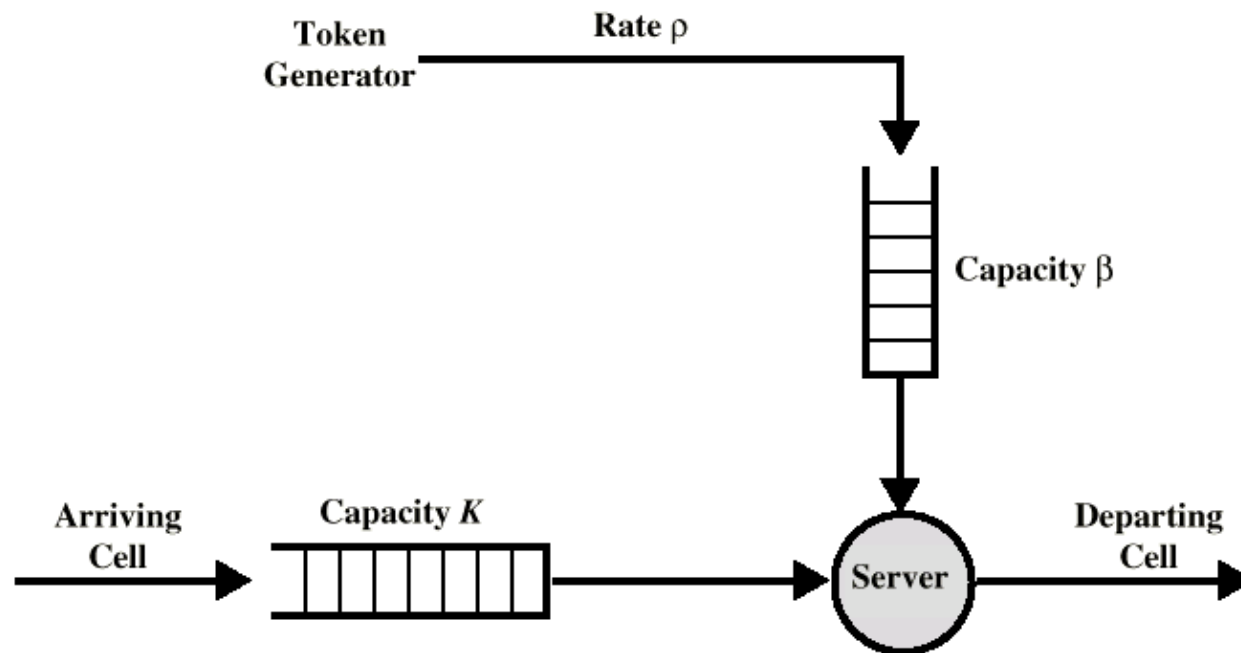


Fixed-rate data



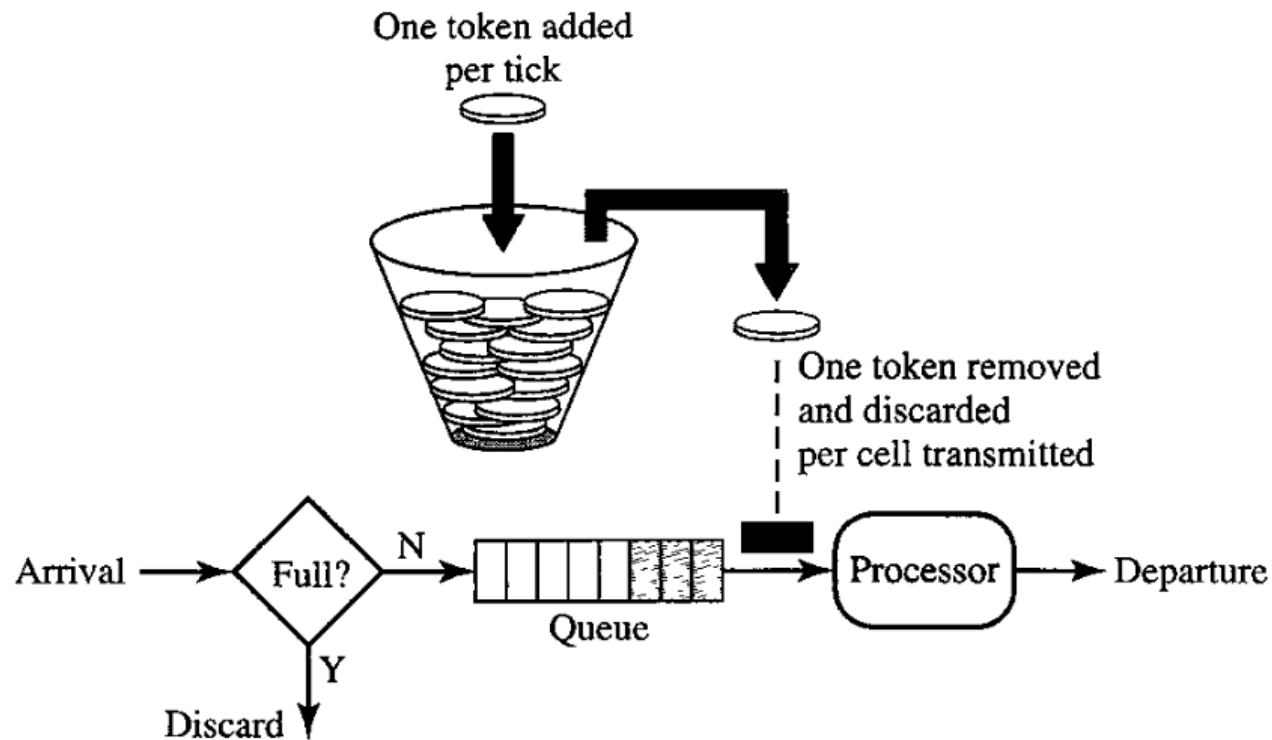
# Control de Tráfico y Congestión en ATM

Shaping del Tráfico: Funcionamiento de Token Bucket



# Control de Tráfico y Congestión en ATM

## Token Bucket



# Control de Tráfico y Congestión en ATM

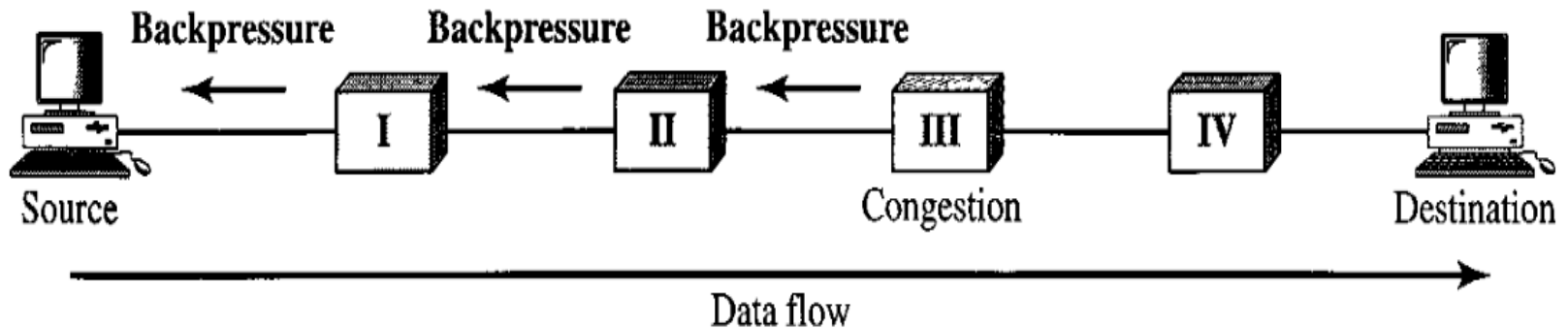
- El tráfico ABR usa técnica de *closed-loop control*
- Las técnicas de *closed loop control* usan mecanismos de *feedback* de la red y se orientan a las aplicaciones que no tienen sus características de tráfico bien definidas tales como acceso Web, FTP, etc.
- Los parámetros usados por las técnica de feedback para el tráfico ABR son:
  - Allowed Cell Rate (ACR): la tasa actual a la que transmite el emisor
  - Minimum Cell Rate (MCR): el mínimo valor que puede tener el ACR
  - Peak Cell Rate (PCR): el máximo valor que puede tener el ACR
  - Initial Cell Rate (ICR): el valor inicial del ACR

# Control de Tráfico y Congestión en ATM

- Algunas técnicas *closed-loop*:
  - Backpressure
  - Choke Packet
  - Implicit Signaling
  - Explicit Signaling

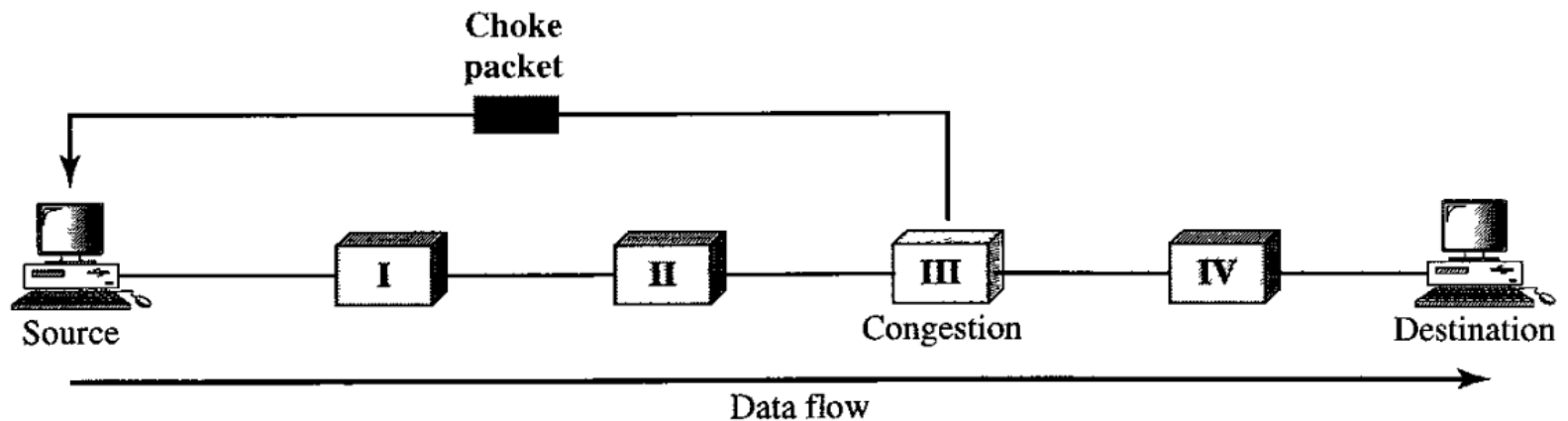
# Control de Tráfico y Congestión en ATM

- Backpressure

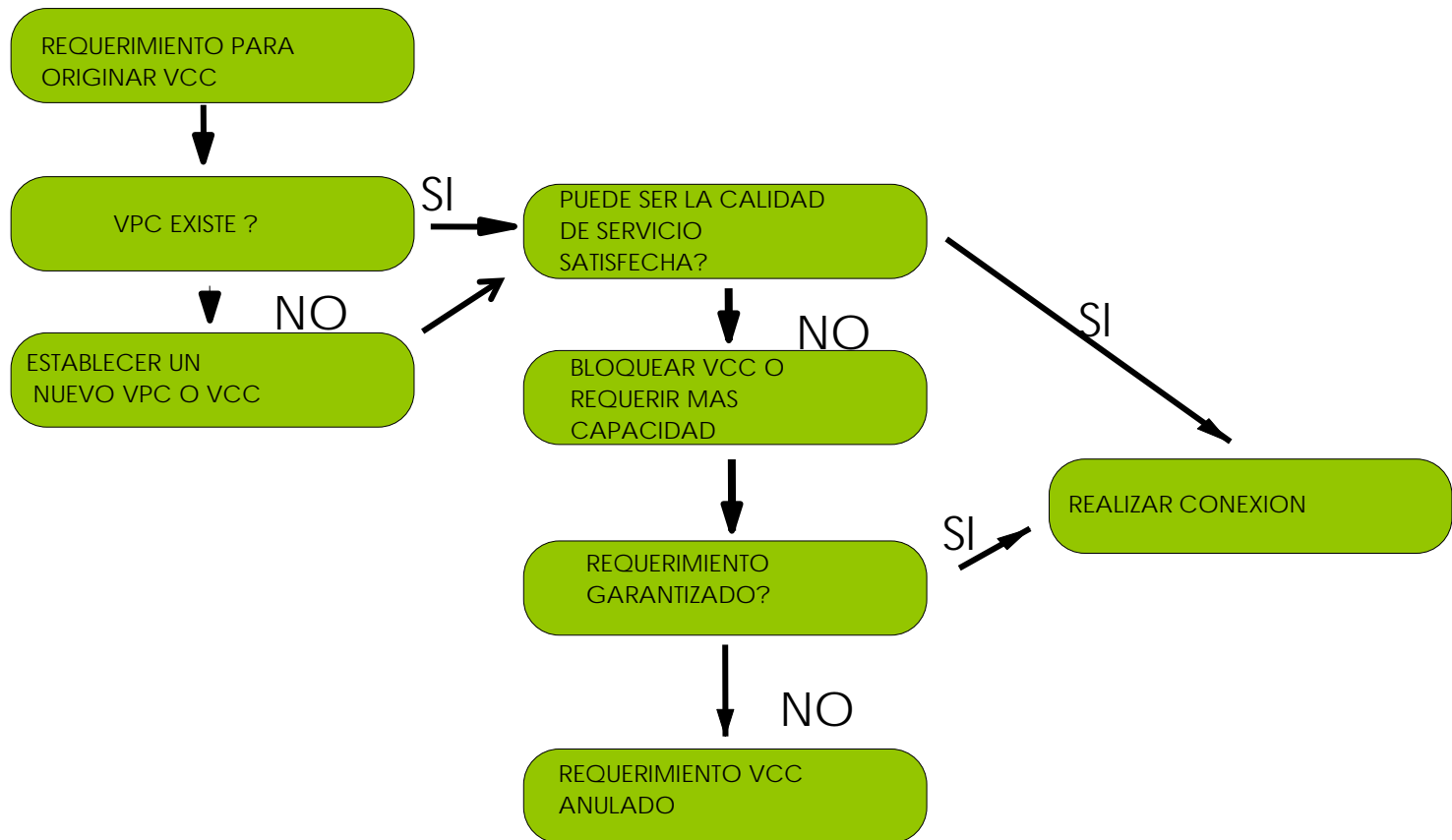


# Control de Tráfico y Congestión en ATM

- Choke Packet



# ATM: Establecimiento de una llamada



## ATM: Señalización de Control

- En ATM es necesario un mecanismo para el establecimiento/liberación de VPC y VCC
- El intercambio de información involucrado en este proceso se denomina *señalización de control* y tiene lugar a través de conexiones distintas de las que están siendo gestionadas



# ATM: Señalización de Control

- Son especificados 4 métodos para llevar a cabo el establecimiento/liberación de VCCs. En todas las redes se usa uno de estos métodos o una combinación de ellos:
  1. Las SVC se pueden usar para el intercambio usuario-usuario, en cuyo caso no se necesita señalización de control
  2. Si no existe canal de señalización de control de llamada preestablecido, se debe establecer uno. Para este fin tiene lugar un intercambio de señales de control entre el usuario y la red a través de algún canal. Por tanto, es necesaria una PVC que pueda ser utilizada para establecer las VCC en el control de llamadas. Un canal de este tipo se denomina *canal de metaseñalización*, dado que se emplea para establecer canales de señalización

# ATM: Señalización de Control

3. El canal de metaseñalización se puede utilizar para establecer una VCC entre el usuario y la red para la señalización de control de llamadas. Este *canal virtual de señalización del usuario a la red* se puede utilizar para establecer VCC para la transmisión de datos de usuario
4. El canal de metaseñalización se puede utilizar también para establecer un *canal virtual de señalización usuario-usuario*, que debe configurarse en una VPC preestablecida. Este canal se puede utilizar para permitir a los dos usuarios finales (sin que la red intervenga) el establecimiento y liberación de VCC usuario-usuario para el transporte de datos

# ATM: Claves del Éxito

- El uso de celdas de tamaño fijo favorece el reenvío rápido de las mismas debido a las características de la implementación de hardware
- Las capacidades mínimas de control de errores y flujo reduce el tamaño de la cabecera de la celda destinado a estas labores
- Tener pocos campos en la cabecera reduce el costo operativo de procesar la celda

# ATM: Claves del Éxito

- El uso de canales virtuales y caminos virtuales permite mantener una arquitectura de red sencilla con funciones bien definidas para los caminos y los canales
- La agrupación de entidades permite un mayor rendimiento de la red ya que se puede mantener pequeñas las tablas de conmutación

## ATM: Claves del Éxito

- El tiempo de establecimiento de una llamada es corto debido a que una vez creado el camino virtual, la creación de un canal virtual toma muy poco tiempo

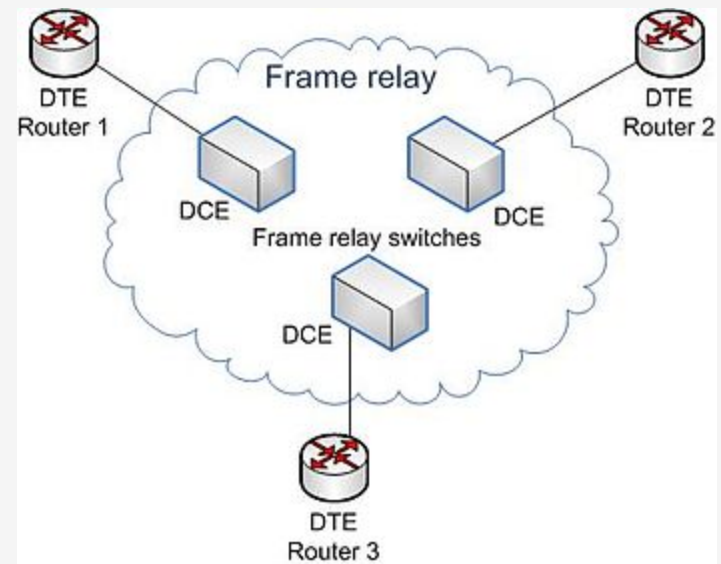
# ATM: Resumen

- ◉ Conmutación de celdas
  - ◉ Tamaño fijo (53 bytes)
- ◉ Orientado a conexión con circuitos virtuales
- ◉ Transmisión asíncrona de celdas
- ◉ Multiplexación de la conexión

# ATM: Resumen

- Secuenciamiento de celdas
  - Tubería FIFO
- Control de errores mínimo a nivel de enlace
  - No contempla retransmisión
  - Sólo la cabecera

# Frame Relay





# Frame Relay

- Las investigaciones sobre conmutación de paquetes comienzan alrededor de 1970
  - Comunicaciones digitales de datos a larga distancia
- La tecnología básica de conmutación de paquetes es fundamentalmente la misma
- La conmutación de paquetes sigue siendo una de las pocas tecnologías eficientes para comunicaciones a largas distancias

# Frame Relay

- ◉ Dos tecnologías WAN importantes:
  - ◉ ATM
  - ◉ Frame Relay
- ◉ Variaciones del enfoque básico de conmutación de paquetes
- ◉ Las ventajas de la conmutación de paquetes acarrear un costo
  - ◉ Los nodos deben conocer el estado de la red → retardo
  - ◉ Sobrecarga en la comunicación de estado de la red

# Frame Relay

- Propuesta presentada por el CCITT (ahora ITU-T) en 1984.
- Mayor desarrollo ocurre en 1990.
- Un consorcio formado por Cisco, Digital Equipment Corporation (DEC), Northern Telecom, y StrataCom desarrolló una especificación alineada con las bases de Frame Relay.
- Propusieron unas extensiones conocidas como Local Management Interface (LMI).

# Frame Relay

- Está diseñada para eliminar la sobrecarga de X.25
  - La señalización de control viaja en una conexión lógica separada de los datos
  - Multiplexación y conmutación de conexiones lógicas en capa 2 en lugar de capa 3
  - No hay control de flujo y errores en cada salto

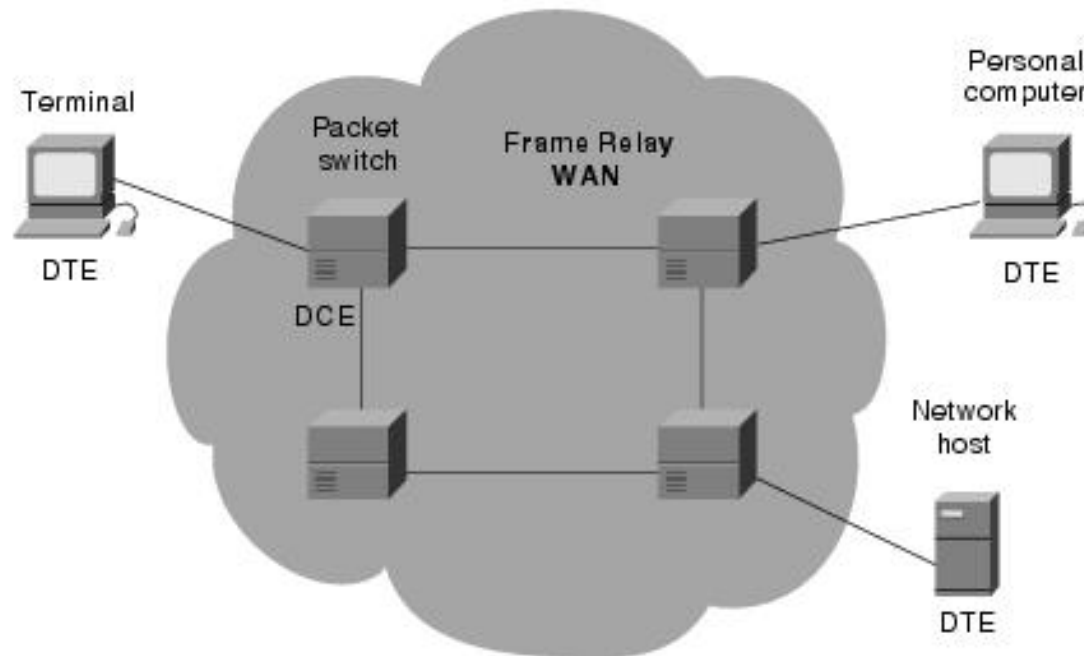
# Frame Relay

- Orientado a conexión
- Paquetes de longitud variable
- Multiplexación estadística

# Frame Relay

- Dispositivos de Frame Relay:
  - Data Terminal Equipment (DTE)
  - Data Circuit-terminating Equipment (DCE)

# Frame Relay



# Frame Relay

- Circuitos Virtuales
  - Conexión lógica establecida entre dos DTEs a través de un red Frame Relay (PSN)
  - Existen dos tipos de VCs:
    - Switched Virtual Circuits (SVCs)
    - Permanent Virtual Circuits (PVCs)



# Frame Relay

- SVCs:
  - Call setup – establecimiento de la llamada.
  - Data transfer – transferencia de datos.
  - Idle – no data es transferida.
  - Call termination – el VC entre DTEs es terminado.

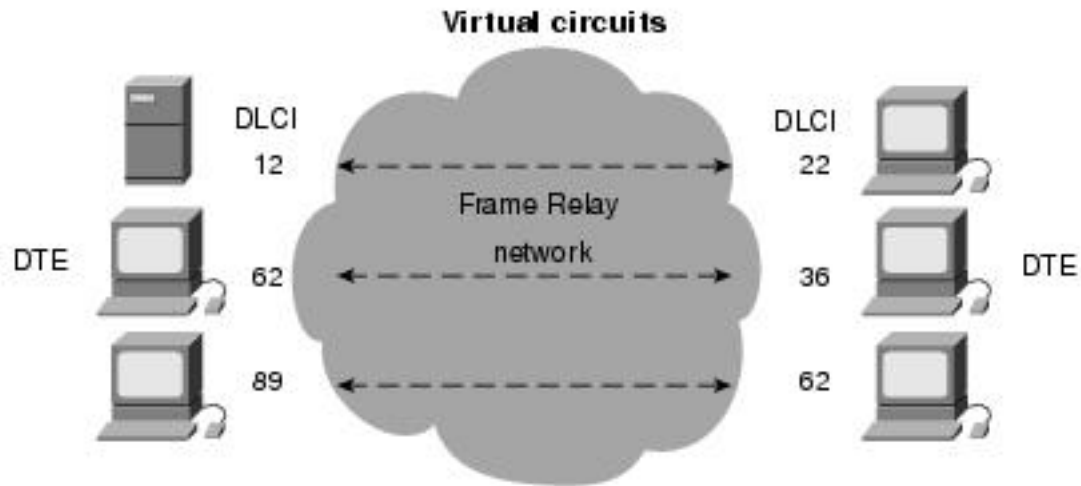
# Frame Relay

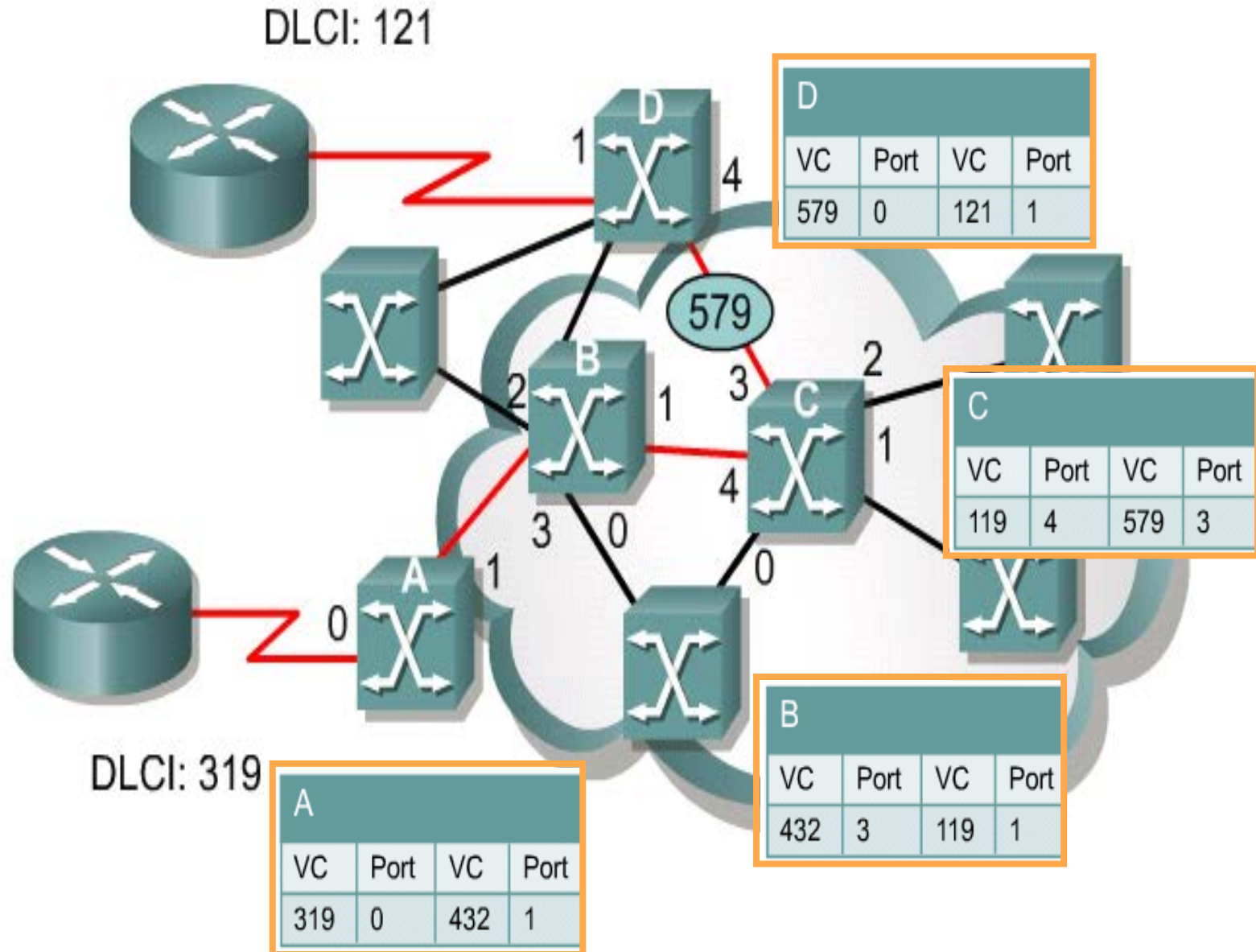
- PVCs:
  - Data transfer – transferencia de datos.
  - Idle – no data es transferida.

# Frame Relay

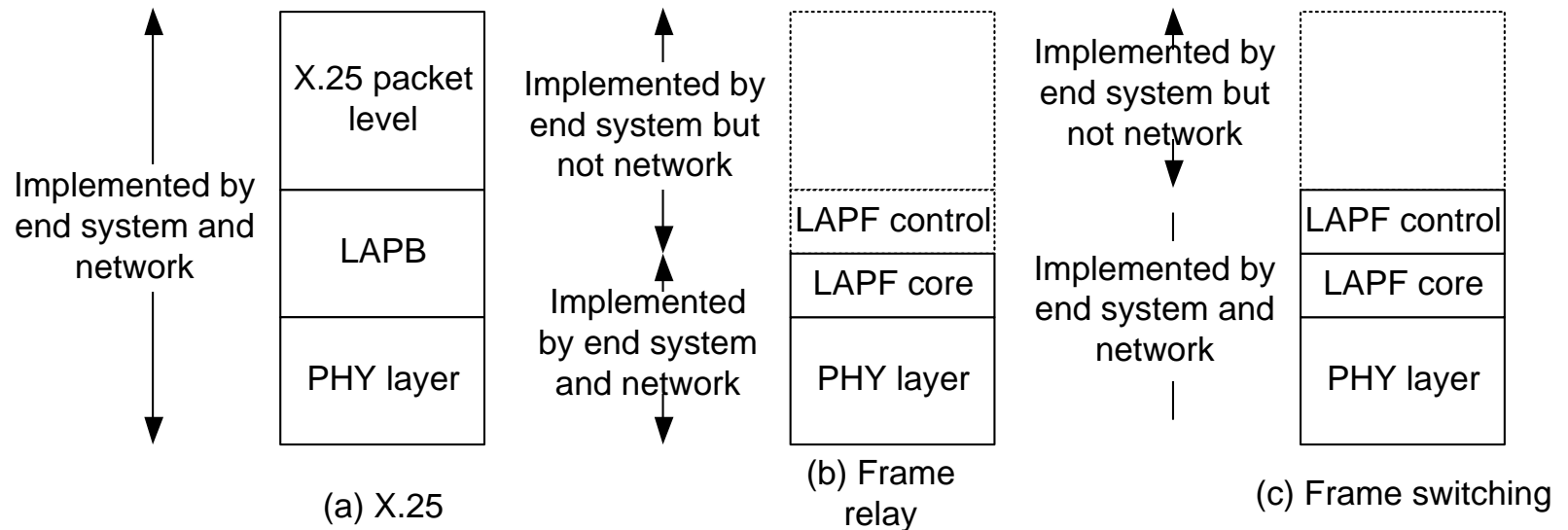
- ◉ Data-Link Connection Identifier (DLCI)
- ◉ Los VCs son identificados por DLCIs
- ◉ Los valores de los DLCIs son asignados por el proveedor del servicio de Frame Relay
- ◉ Los DLCIs tienen significado local
  - ◉ En la LAN, pero no en la WAN

# Frame Relay





# Arquitectura Frame Relay



# Frame Relay

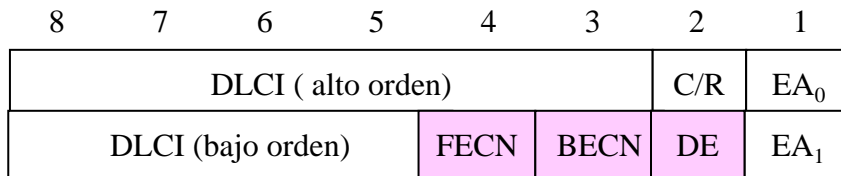
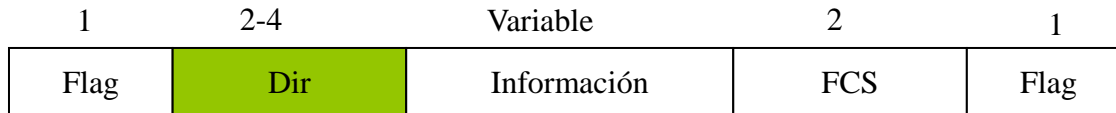
- Funciones core LAPF:
  - Delimitación y alineamiento de trama
  - Multiplexación y demultiplexación de VC
  - Alienación de octetos
  - Seguridad de que las tramas no son ni muy largas ni muy cortas
  - Detección de errores de transmisión
  - Funciones de control de congestión

# Frame Relay

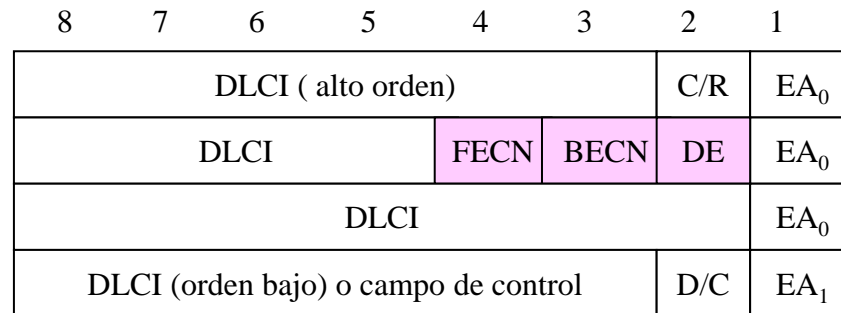
- Funciones adicionales LAPF:
  - Sólo en sistemas finales
  - No forma parte de los *servicios Frame Relay*
  - Control de flujo y errores
- El servicio FR posee las siguientes propiedades:
  - Mantenimiento del orden de las tramas *end-to-end*
  - Baja probabilidad de pérdida de tramas



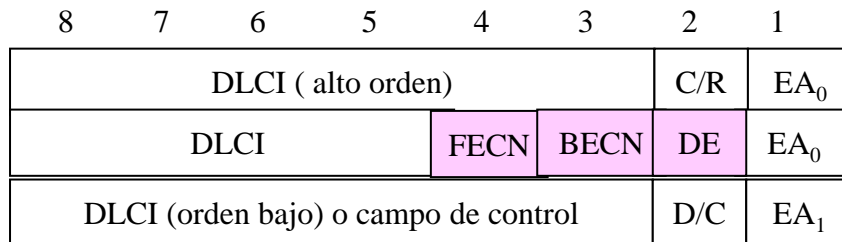
# Frame Relay: LAPF



**Formato del campo de direccionamiento de 2 bytes**



**Formato del campo de direccionamiento de 4 bytes**



**Formato del campo de direccionamiento de 3 bytes**

# Frame Relay

- El formato del campo de direccionamiento es diferente, está formado por los siguientes campos:
  - EA: extensión del campo de dirección
  - C/R: comando/respuesta
  - FECN: notificación de congestión hacia delante
  - BECN: notificación de congestión hacia atrás
  - DLCI: identificador de la conexión de enlace de datos
  - D/C: indicador de control de DLCI o DL-CORE
  - DE: elegible para descarte

# Frame Relay

- Frame Relay usa un mecanismos de verificación de errores basado en CRC
- Sólo se usa para determinar la existencia de errores de transmisión en las tramas
- La corrección de errores se deja a las capas superiores
  - Descarte silencioso

# Control de Congestión

- La norma I.370 define los siguientes objetivos para el control de la congestión en Frame Relay:
  - Minimizar el número de tramas descartadas
  - Mantener la QoS acordada
  - Minimizar la posibilidad de monopolizar los recursos
  - Sencillo de implementar
  - Poca sobrecarga para el usuario y la red
  - Mínimo tráfico de red adicional
  - Distribuir equitativamente los recursos
  - Limitar la propagación de la congestión

# Control de Congestión

- La norma I.370 define los siguientes objetivos para el control de la congestión en Frame Relay:
  - Operar con efectividad, con independencia del flujo de tráfico en cualquier dirección
  - Mínima interacción o impacto sobre otros sistemas de la red
  - Minimizar la varianza de QoS entregada a conexiones individuales durante la congestión

# Control de Congestión

- Difícil por el limitado conjunto de herramientas
  - Racionalización para maximizar el rendimiento y la eficiencia
- No se puede usar una técnica tradicional
  - Ventanas deslizantes en HDLC
- El control de congestión es responsabilidad conjunta de la red y los usuarios finales
  - La red está en mejor posición para monitorizar el grado de congestión
  - Los usuarios están en mejor posición para controlar la congestión limitando el flujo de tráfico
- Algunas técnicas de control de congestión definidas por la ITU-T...

# Control de Congestión

Técnica	Tipo	Elemento clave
Control de Descarte	Estrategia de Descarte	Bit DE
BECN	Prevención de la Congestión	Bit BECN
FECN	Prevención de la Congestión	Bit FECN
Notificación de congestión implícita	Recuperación de Congestión	Número de secuencias en capas superiores

# Control de Congestión

- Descarte
  - Respuesta más importante frente a la congestión
  - Deseable hacerlo de forma equitativa
- Evitar congestión
  - Cuando comienza a producirse la congestión, para minimizar su efecto
  - Mecanismos de señalización explícita
- Recuperación frente a la congestión
  - Prevenir el colapso de la red
  - La red ha empezado a descartar tramas
  - LAPF informa de tramas descartadas
  - Señalización implícita



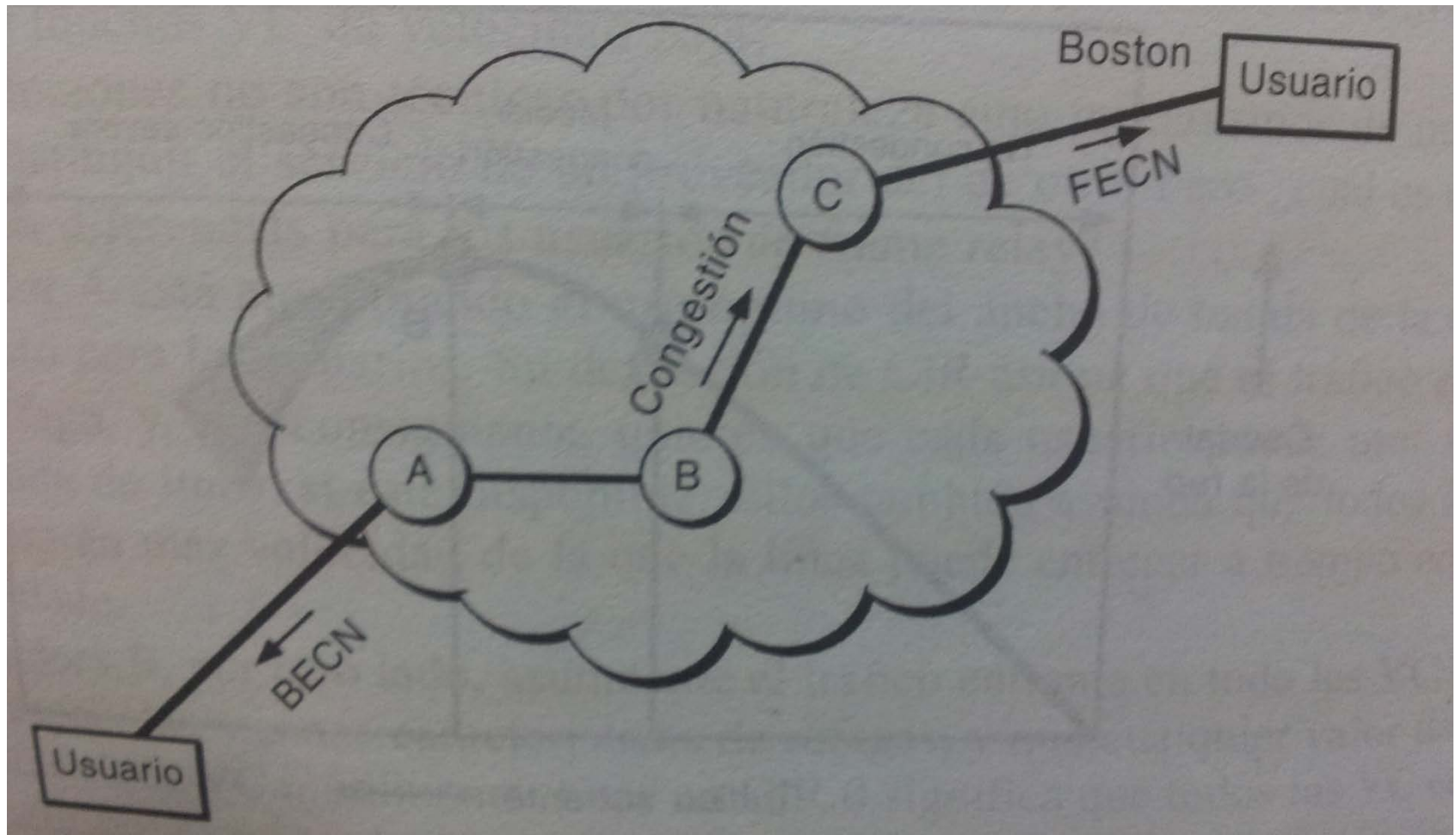
# Control de Congestión

- **Control de descarte:** se le informa a la red que las tramas deben ser descartadas cuando la congestión es severa
- **Notificación explícita de congestión hacia atrás:** el BECN notifica al usuario que los procedimientos de prevención de la congestión deberían ser iniciados *donde sea aplicable* en la dirección opuesta de la trama recibida

# Control de Congestión

- **Notificación explícita de congestión hacia adelante:** el FECN notifica al usuario que los procedimientos de prevención de la congestión deberían ser iniciados *donde sea aplicable* en la misma dirección de la trama recibida
- **Notificación de congestión Implícita:** el sistema final infiere que hay congestión en base a las tramas perdidas

# Control de Congestión



# Control de Congestión

- Frame Relay proporciona un mecanismo de control de tráfico para prevenir la congestión
- Esto lo hace a través de la **gestión de la tasa de tráfico**
- Se define la *Committed Information Rate (CIR)*
  - Tasa en bps que la red se compromete para soportar una conexión particular
- La data transmitida por encima de su CIR se puede descartar en caso de congestión

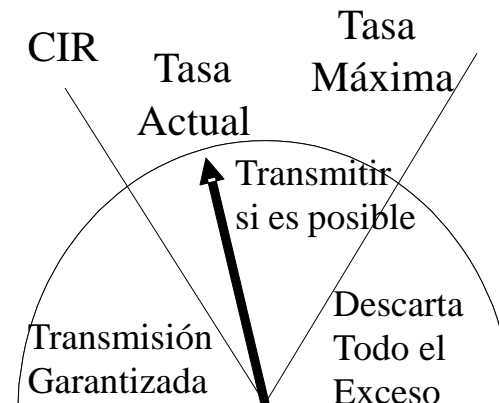
# Control de Congestión

- A la hora de reservar recursos existen algunas limitaciones por la tasa de acceso:

$$\sum_i CIR_{ij} \leq \textit{Tasa de Acceso}_j$$

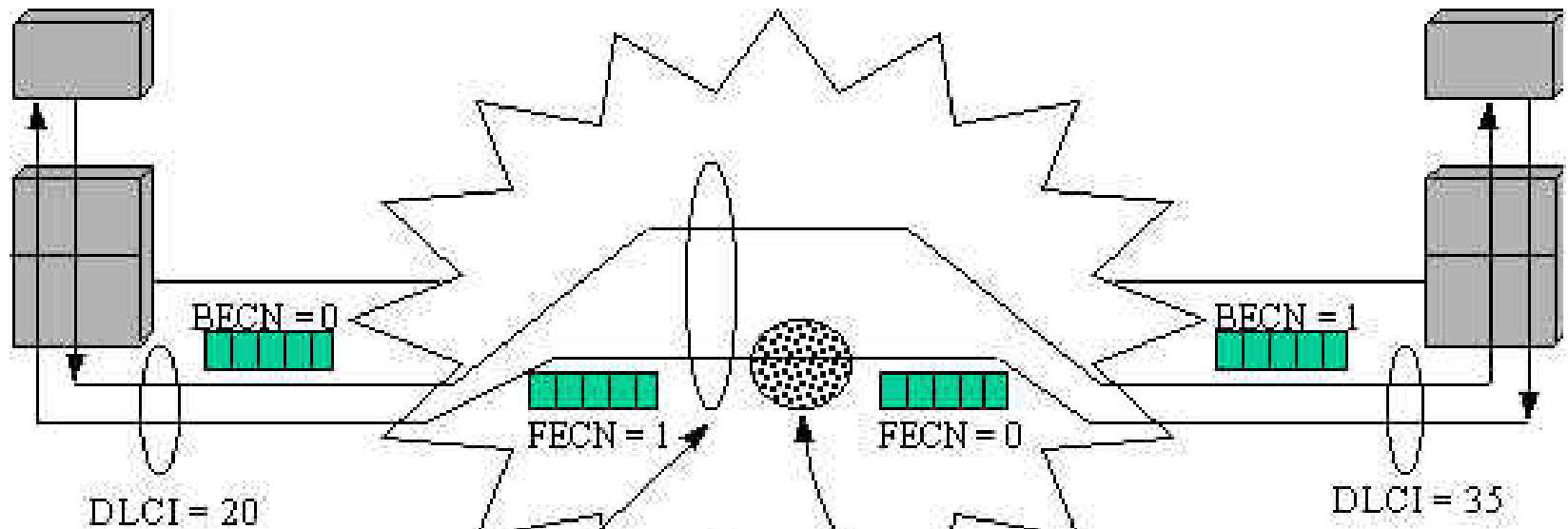
$CIR_{ij}$  : CIR para la conexión  $i$  sobre el canal  $j$

$\textit{Tasa de Acceso}_j$ : Velocidad de transmisión de datos del canal  $j$  de acceso al usuario



# Control de Congestión

(representación de una sola conexión)



Los sentidos de una misma conexión no tienen  
porqué atravesar los mismos órganos de la red.

Puede congestionarse sólo esta  
parte de la red.

# Control de Congestión

- Existen otros dos parámetros que se usan para gestionar el tráfico:
  - Tamaño de Ráfaga comprometido ( $B_c$ ):** es la máxima cantidad de datos que la red se compromete a transmitir bajo condiciones normales en un intervalo  $T$
  - Tamaño de Ráfaga en exceso ( $B_e$ ):** la máxima cantidad en exceso sobre el  $B_c$  que la red puede intentar transmitir bajo condiciones normales en un tiempo  $T$ . La data que representa  $B_e$  es transmitida con una probabilidad menor que la data dentro de su  $B_c$

# Control de Congestión

- La relación entre Bc y CIR es:

$$T = Bc/CIR$$



# Control de Congestión

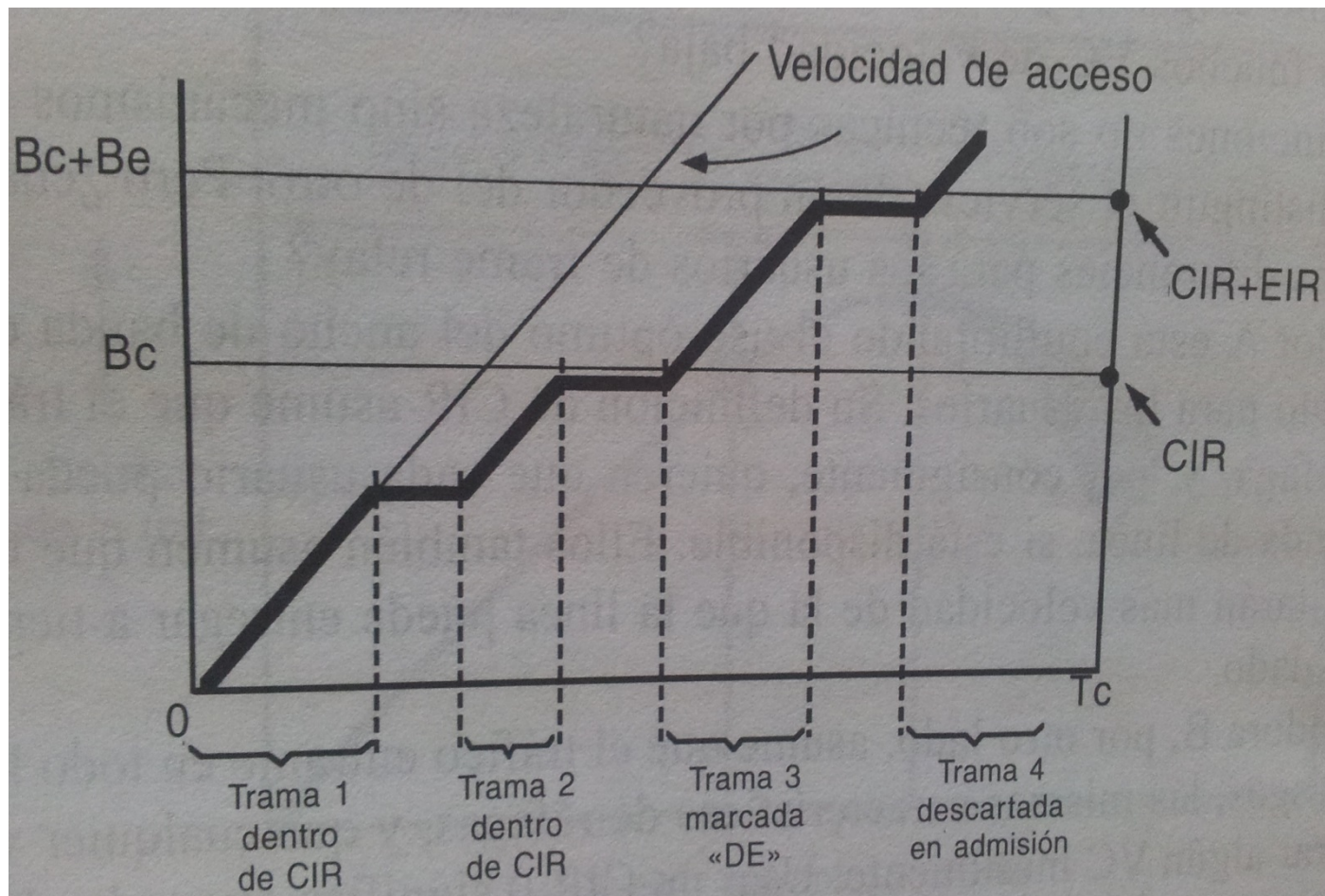
- El tamaño del exceso de la ráfaga, en bps, puede calcularse como:

$$EIR = Be/T$$

# Control de Congestión

- Hay un límite superior para la cantidad de información que el usuario puede ceder en un intervalo de tiempo dado
- Si un usuario transmite más de  $(B_c + B_e)$  bits en un tiempo  $T$ , las tramas en exceso se descartan inmediatamente

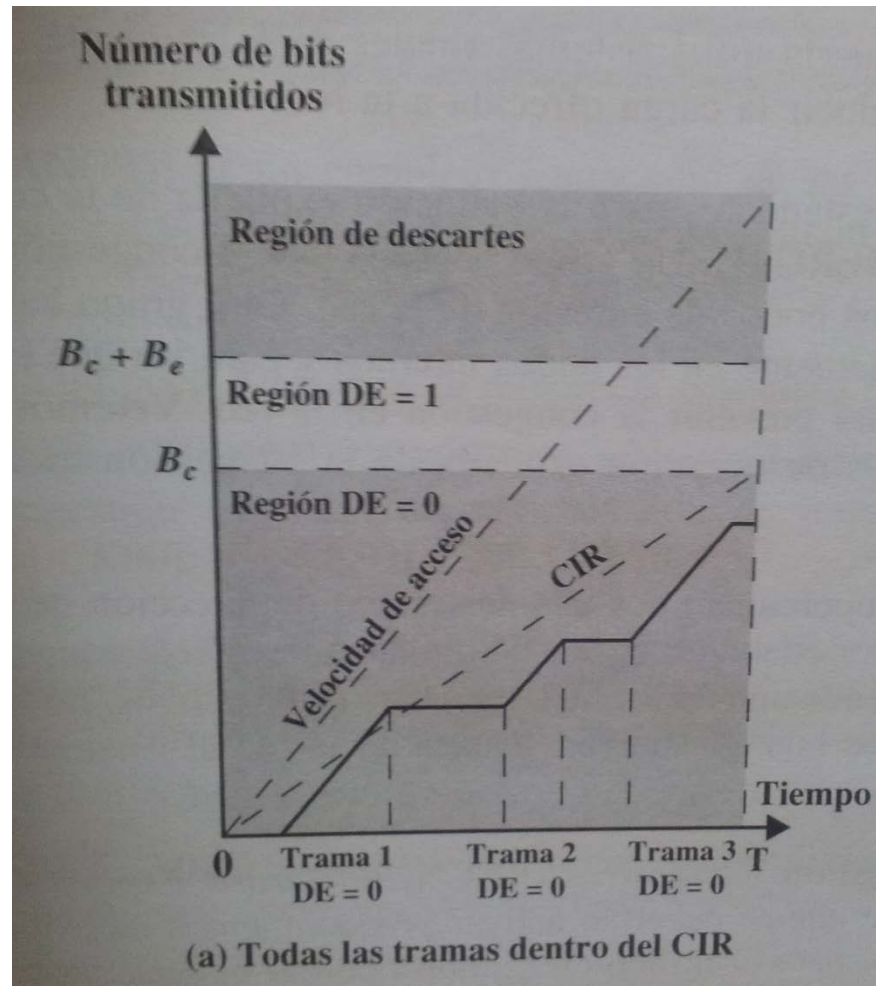
# Control de Congestión



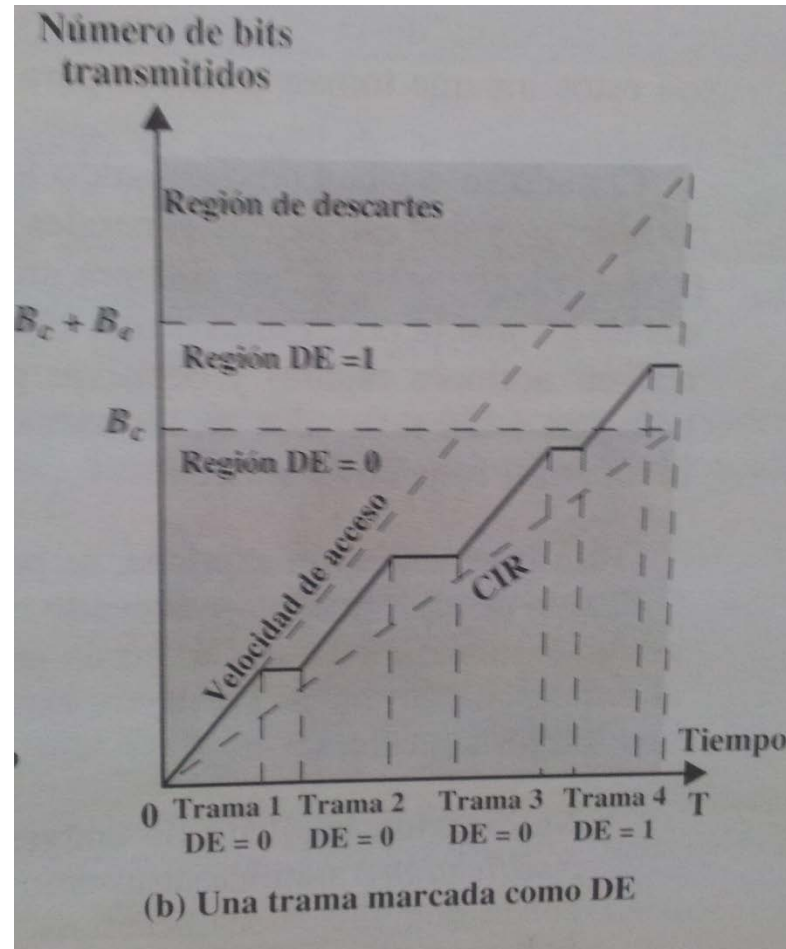
# Control de Congestión

- Para evitar descartar tramas en el nodo de entrada a la red,  $B_e$  normalmente se establece como la diferencia entre  $B_c$  y la velocidad de acceso a la interfaz
- Para que  $CIR + EIR$  iguale a la velocidad de acceso

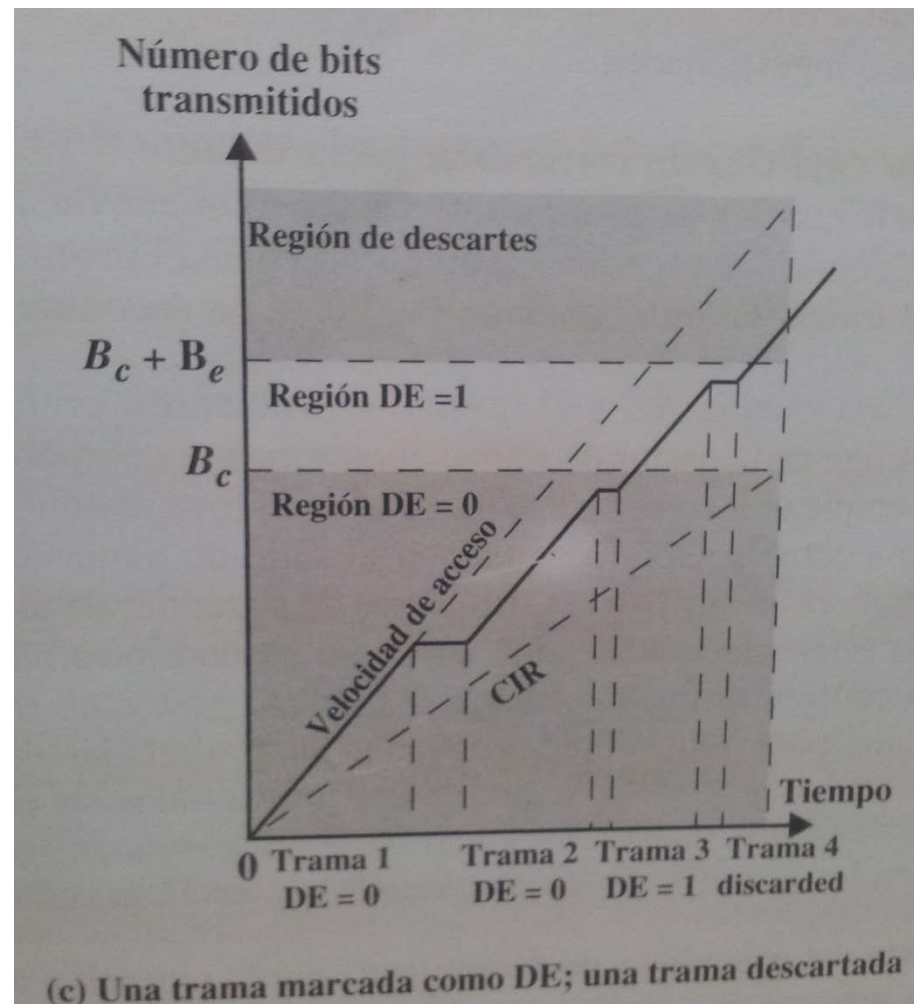
# Control de Congestión



# Control de Congestión

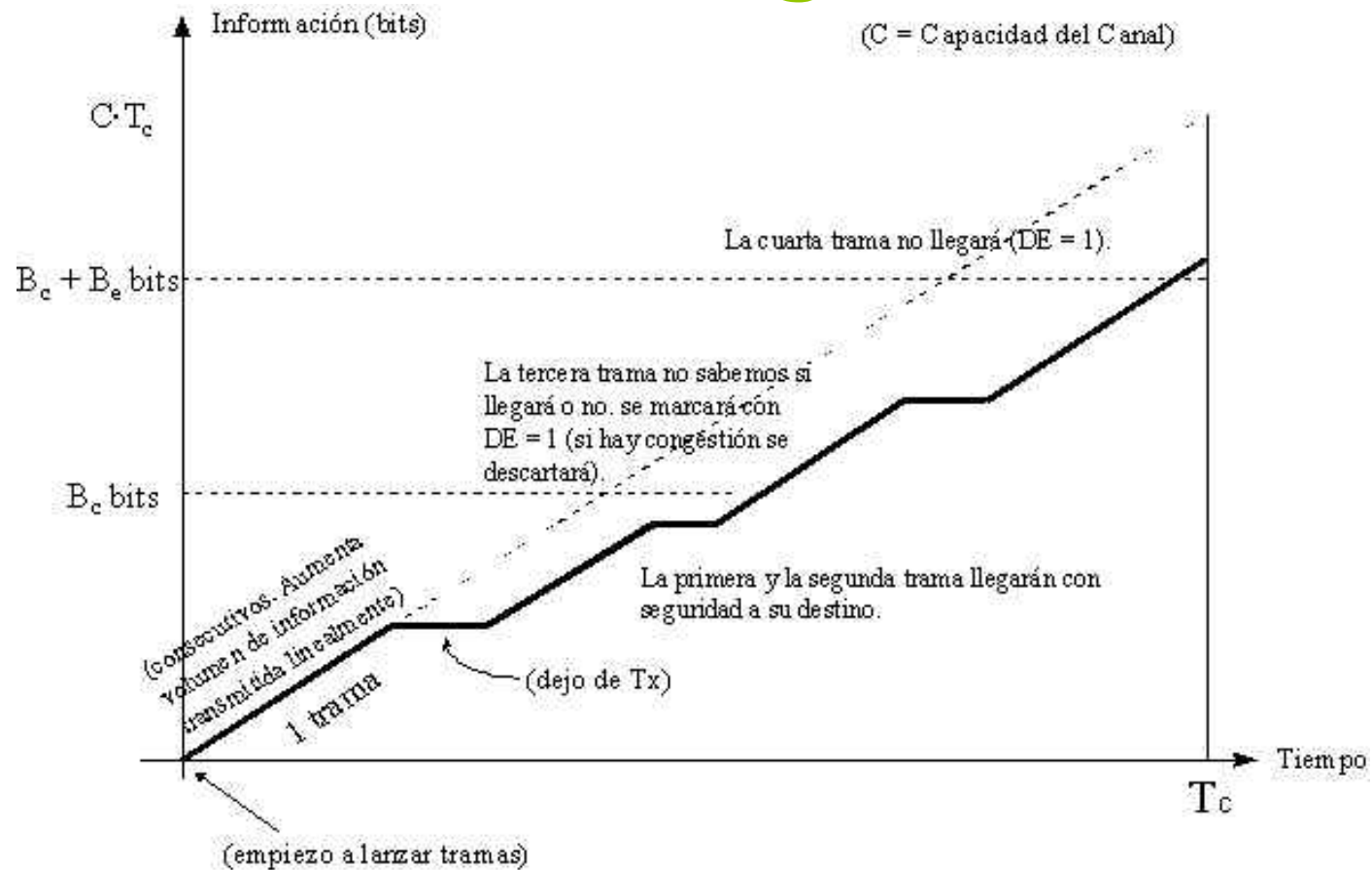


# Control de Congestión





# Control de Congestión





# Control de Congestión

- El switch mantiene un contador de bit para cada VC
- Una trama entrante es marcada como DE si pone el contador de bit sobre la tasa Committed Burst ( $B_c$ )
- Una trama entrante es descartada si excede el contador sobre la suma de Committed Burst + Excess Burst ( $B_c + B_e$ )
- Al final de cada Committed Time interval ( $T$ ) el contador es reducido por el valor de la tasa Committed Burst ( $\text{Counter} - B_c$ )

# Control de Congestión

- Es deseable usar tanto como sea posible la capacidad de la red
  - Capacidad de reaccionar a la congestión de forma controlada y equitativa
- Alertar a los sistemas finales de congestión creciente
  - Tomar medidas para reducir la carga de la red

# Control de Congestión

- BECN
  - Backward Explicit Congestion Notification
  - Notifica al usuario que se deberían activar procedimientos para evitar la congestión para el tráfico en sentido opuesto al de la trama recibida
  - Las tramas que transmita el usuario en esta conexión lógica pueden encontrarse con recursos congestionados

# Control de Congestión

- FECN
  - Forward Explicit Congestion Notification
  - Notifica al usuario que se deberían activar procedimientos para evitar la congestión para el tráfico en mismo sentido al de la trama recibida
  - Esta trama, en esta conexión lógica, se ha encontrado con recursos congestionados

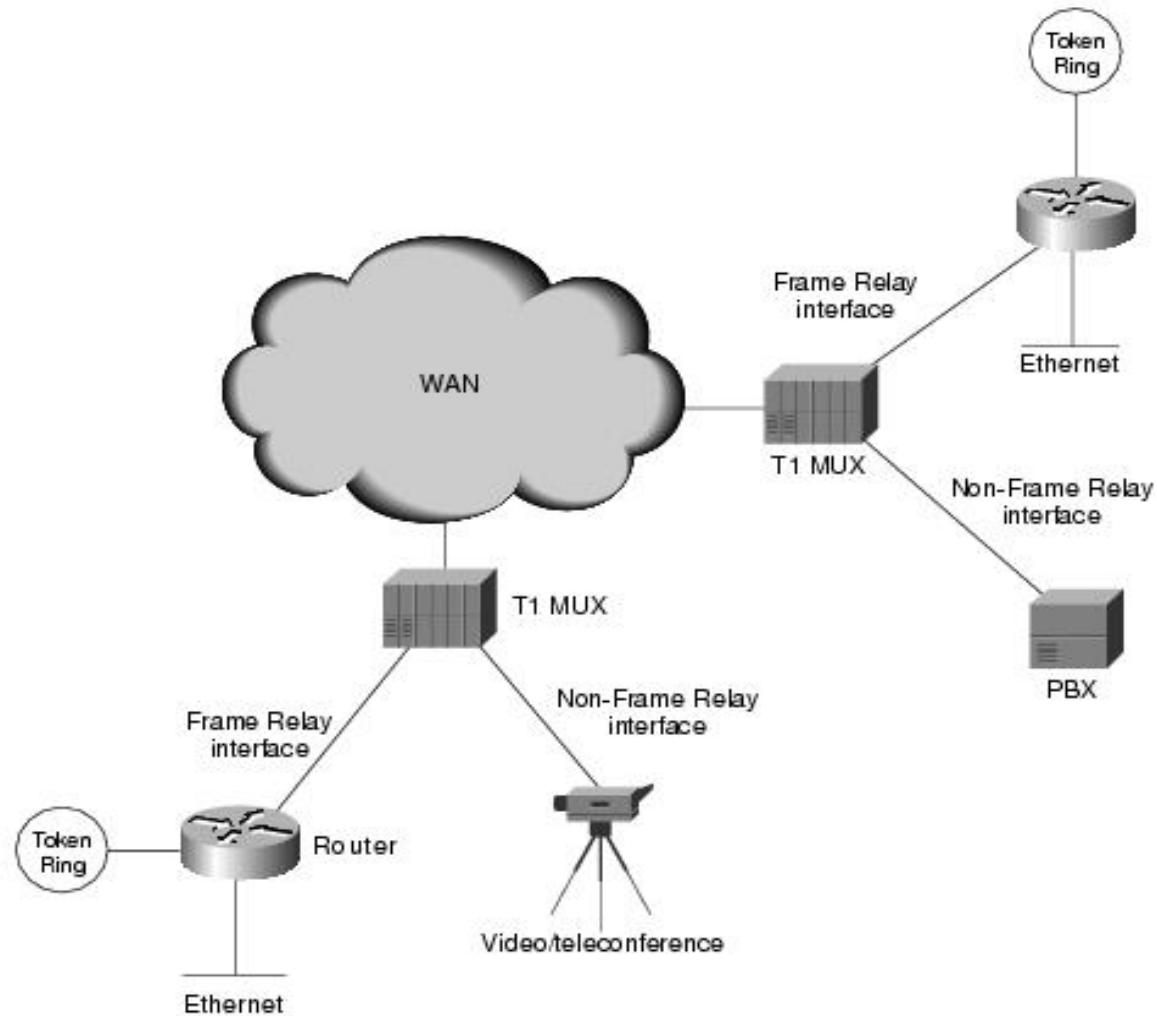
# Control de Congestión

- ¿Cómo los usan los bits FECN y BECN la red y el usuario?
- Respuesta de la red
  - Monitoreo del comportamiento de las colas
  - Si la longitud de las colas comienza a crecer hasta un umbral, se activa FECN o BECN, o ambos
  - Capacidad de elección sobre qué conexiones deben ser alertadas de la congestión
    - Si es muy seria, todas las conexiones
    - Etapas tempranas, a quienes generan mayor tráfico

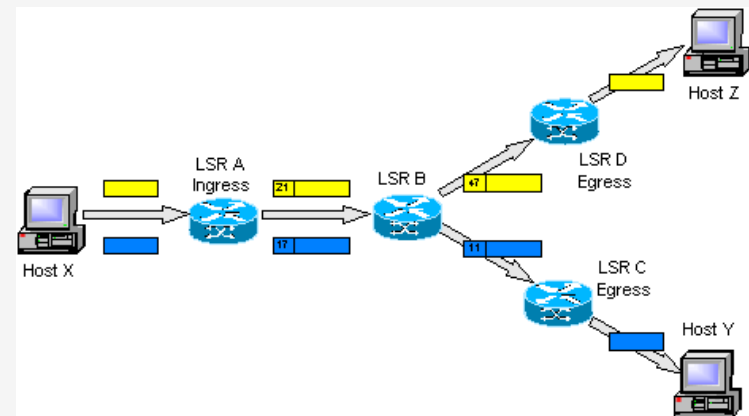
# Control de Congestión

- ¿Cómo los usan los bits FECN y BECN la red y el usuario?
- Respuesta del usuario
  - Recepción de señales BECN y/o FECN
  - Respuesta a BECN, reducir velocidad de transmisión
  - Respuesta a FECN, notificar a interlocutor
    - Niveles superiores (por ejemplo, transporte)

# Ejemplo de Implementación



# MPLS: MULTIPROTOCOL LABEL SWITCHING





# MPLS

- Diversos mecanismos han sido diseñados para mejorar el rendimiento de las redes IP y para proporcionar diferentes niveles de QoS a los usuarios del servicio
- Aunque los protocolos de enrutamiento tienen como propósito fundamental encontrar dinámicamente una ruta entre cualquier par fuente y destino dentro de una red, también proporcionan soporte para objetivos relacionados con el rendimiento, de dos maneras:
  - Reacción ante la congestión
  - Las rutas pueden basarse en distintas métricas

# MPLS

- Mecanismos muy primitivos
  - No aportan soluciones a la mejora del rendimiento global y al manejo del retardo en una red
- MPLS supone un esfuerzo prometedor en la provisión del tipo de soporte de QoS, orientado a conexión y a la gestión de tráfico
- Acelerar los procesos de envío de paquetes conservando la flexibilidad de un modelo de red IP
- Las raíces de MPLS se remontan a una serie de esfuerzos realizados a mediados de los '90s encaminados a "casar" dos tecnologías: IP y ATM.

# MPLS

- ◉ El primero de esos esfuerzos que alcanzó el mercado real fue la conmutación IP, desarrollada por Ipsilon
- ◉ Para competir con esta oferta, numerosas compañías anunciaron sus propios productos, fundamentalmente Cisco Systems, IBM y Cascade
- ◉ Sus productos tenían como objetivo mejorar el rendimiento y las características de retardo de IP, y todos tomaban el mismo enfoque: utilizar un protocolo de enrutamiento estándar, como OSPF, para definir las rutas entre los diversos puntos, asignar paquetes a estas rutas según entraban en la red y utilizar conmutadores ATM para mover los paquetes a lo largo de las rutas

# MPLS

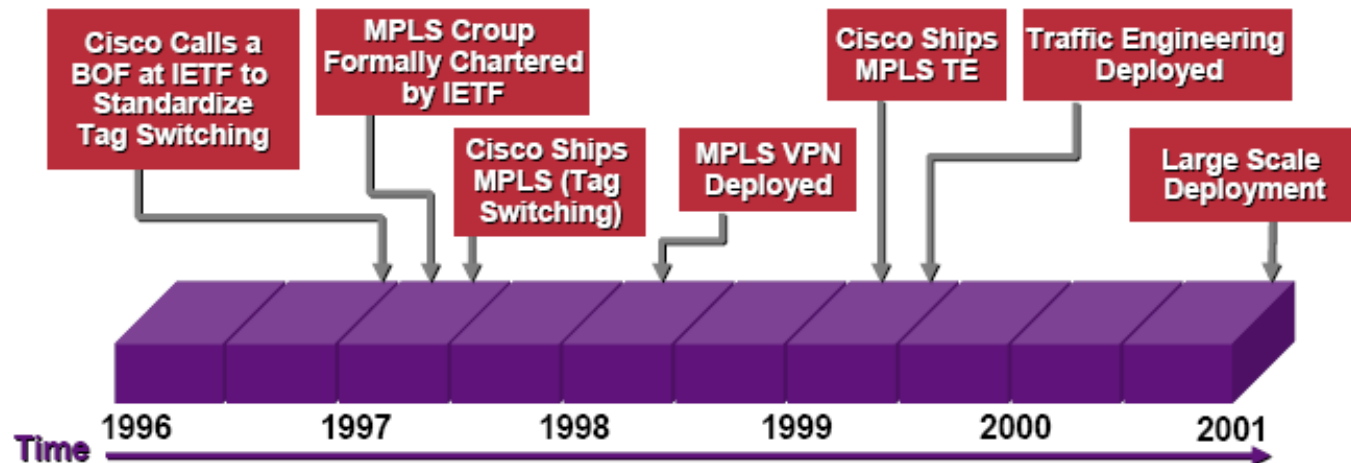
- Cuando estos productos salieron al mercado, los conmutadores ATM eran mucho más rápidos que los routers IP y lo que se pretendía era mejorar el rendimiento empujando el mayor número posible de tráfico hacia abajo, hacia el nivel ATM y empleando hardware de conmutación ATM

# MPLS

- En respuesta a estas iniciativas, la IETF constituyó el grupo de trabajo MPLS en 1997 para desarrollar un modelo común estandarizado
- Este grupo hizo público su primer conjunto de estándares propuestos en el año 2001
- Mientras tanto, el mercado no se quedó de brazos cruzados
  - El final de los '90s fue testigo de la introducción de varios routers igual de rápidos que los conmutadores ATM, eliminando la necesidad de tener ambas tecnologías (ATM e IP) en la misma red
  - MPLS juega un papel esencial, ya que reduce la cantidad de procesamiento requerida por paquete en cada router

# MPLS

Evoluciona a partir del tag switching (Cisco) y otras propuestas IBM (ARIS) y Toshiba (CSR)



# MPLS: Conceptos Básicos

- Publicado por la IETF en el RFC 3031 en el año 2001
- Tecnología de mapeo de etiquetas y reenvío de paquetes
  - Integra el intercambio de etiquetas con el enrutamiento a nivel de red
- El intercambio de etiquetas implica el cambio del valor de la etiqueta en la cabecera del paquete a medida que este se desplaza de nodo en nodo dentro de la red
- La idea de MPLS es mejorar el rendimiento del enrutamiento de capa de red y su escalabilidad
- MPLS da soporte a diversas tecnologías de red, ubicándose entre las capas 2 y 3 del modelo OSI

# Arquitectura: Nodos

- ◉ Dispositivo de interconexión que soporta MPLS
- ◉ Tiene conocimiento de los protocolos de control MPLS
- ◉ Opera en uno o más protocolos de enrutamiento de capa 3
- ◉ Capaz de reenviar paquetes en base a etiquetas
- ◉ Opcionalmente puede reenviar paquetes capa 3 nativos
- ◉ Los nodos MPLS son llamados LSR (Label Switching Router)



# Arquitectura: Nodos

- Nodo de Tránsito (Transit Node)
  - Recibe el PDU y usa la cabecera MPLS para tomar las decisiones de reenvío
  - Realiza intercambio de etiquetas
  - También es llamado LSR interior, o LSR de *core*

# Arquitectura: Nodos

- ◉ Nodo de Borde (Edge Node)
  - ◉ Conecta un dominio MPLS con un nodo fuera del dominio, ya sea porque no soporta MPLS, y/o porque está en un dominio diferente
  - ◉ Dos tipos:
    - ◉ Nodo de Egreso (Egress Node)
    - ◉ Nodo de Ingreso (Ingress Node)

# Forwarding Equivalence Class

- MPLS no toma decisiones de reenvío por cada paquete de capa 3, sino que usa un concepto denominado *Forwarding Equivalence Class* (FEC)
- La FEC es un grupo de paquetes IP que son reenviados del mismo modo
- Una FEC es asociada con una clase de datagramas
  - La clase depende de varios factores
- Basándose en la FEC, una etiqueta es negociada entre LSRs vecinos, desde el ingreso hasta el egreso de un dominio

## Forwarding Equivalence Class

- La asignación de un particular FEC es realizado una vez que el paquete entra a la red
  - Así se le asigna una etiqueta
- Cuando el paquete es reenviado al próximo salto, la etiqueta viaja en él, es decir el paquete es etiquetado antes de ser reenviado

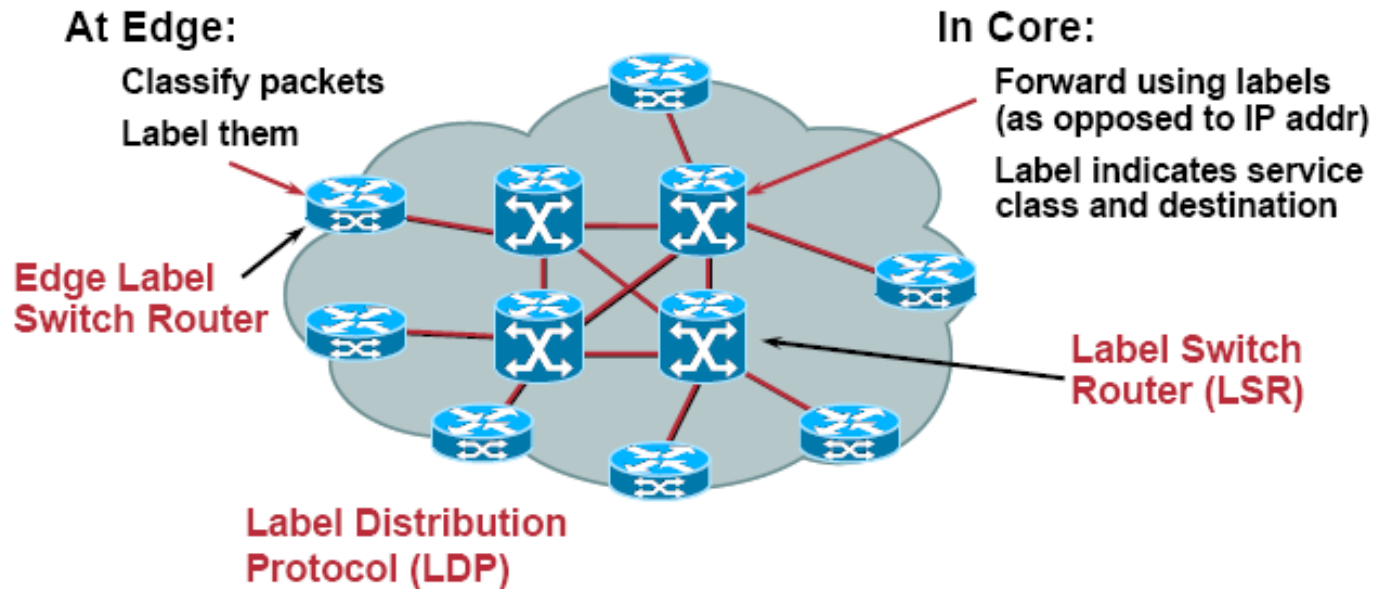
# Forwarding Equivalence Class

- Los posibles parámetros para determinar la FEC de un paquete:
  - Dirección IP origen/destino
  - Puerto origen/destino
  - ID del protocolo
  - *Codepoint* de servicios diferenciados
  - Etiqueta de flujo IPv6

# Etiquetas

- Una etiqueta (Label) es un identificador corto, de longitud fija y de alcance local que identifica una FEC
- La etiqueta que se coloca en un paquete en particular representa a la FEC a la que el paquete está asignado
- Comúnmente, esta asignación se hace basándose en la dirección de red del destino, sin llegar a ser una codificación de la misma

# MPLS: Conceptos



# Planos

- MPLS opera con planos de datos y de control
- La principal tarea del *plano de control* es anunciar las etiquetas y las direcciones, y relacionarlas
  - Asociar (mapear) etiquetas y direcciones
- Más de un protocolo puede operar en el plano de control de MPLS, entre ellos LDP (Label Distribution Protocol)



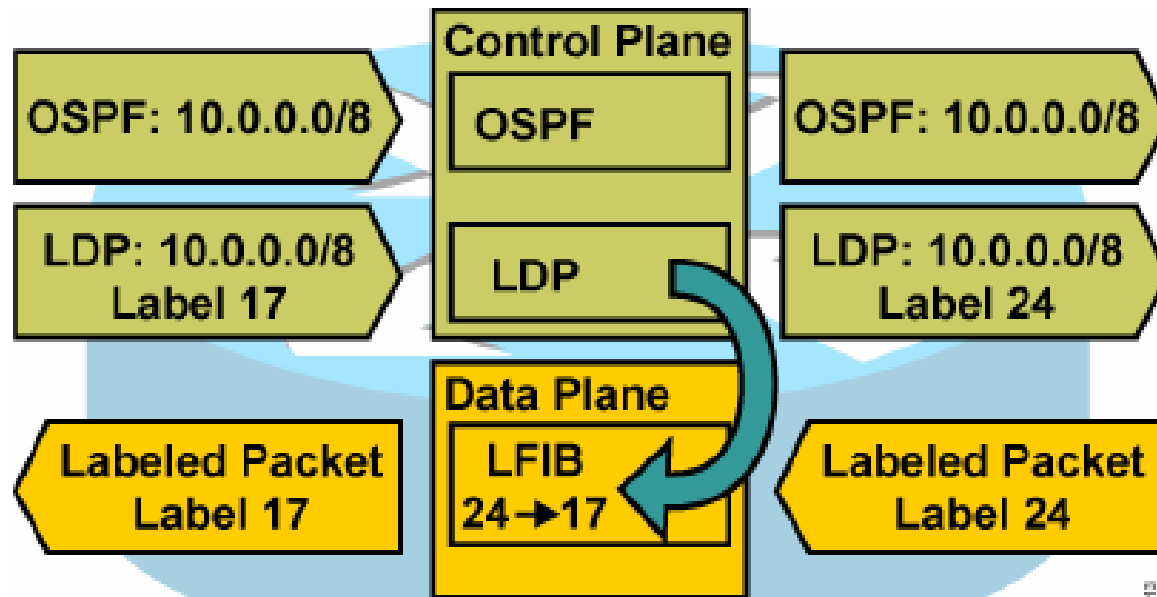
# Planos

- Los mensajes de control se intercambian entre LSRs para realizar varias operaciones:
  - Intercambio de mensajes periódicos de *hello*
  - Intercambio de mensajes de etiquetas y direcciones para ser asociadas, y usar posteriormente esta asociación en el plano de datos para el reenvío de tráfico
- Luego del intercambio de etiquetas y direcciones de red, estas son *asociadas* (binding)

# Planos

- El *plano de datos* de MPLS reenvía el tráfico examinando la etiqueta en el encabezado del paquete MPLS
- La dirección de red no es examinada hasta que el tráfico es entregado a través de la red al nodo del usuario receptor
- La cabecera de etiquetas es entonces eliminada y se usa la dirección de red para entregar el paquete

# Planos



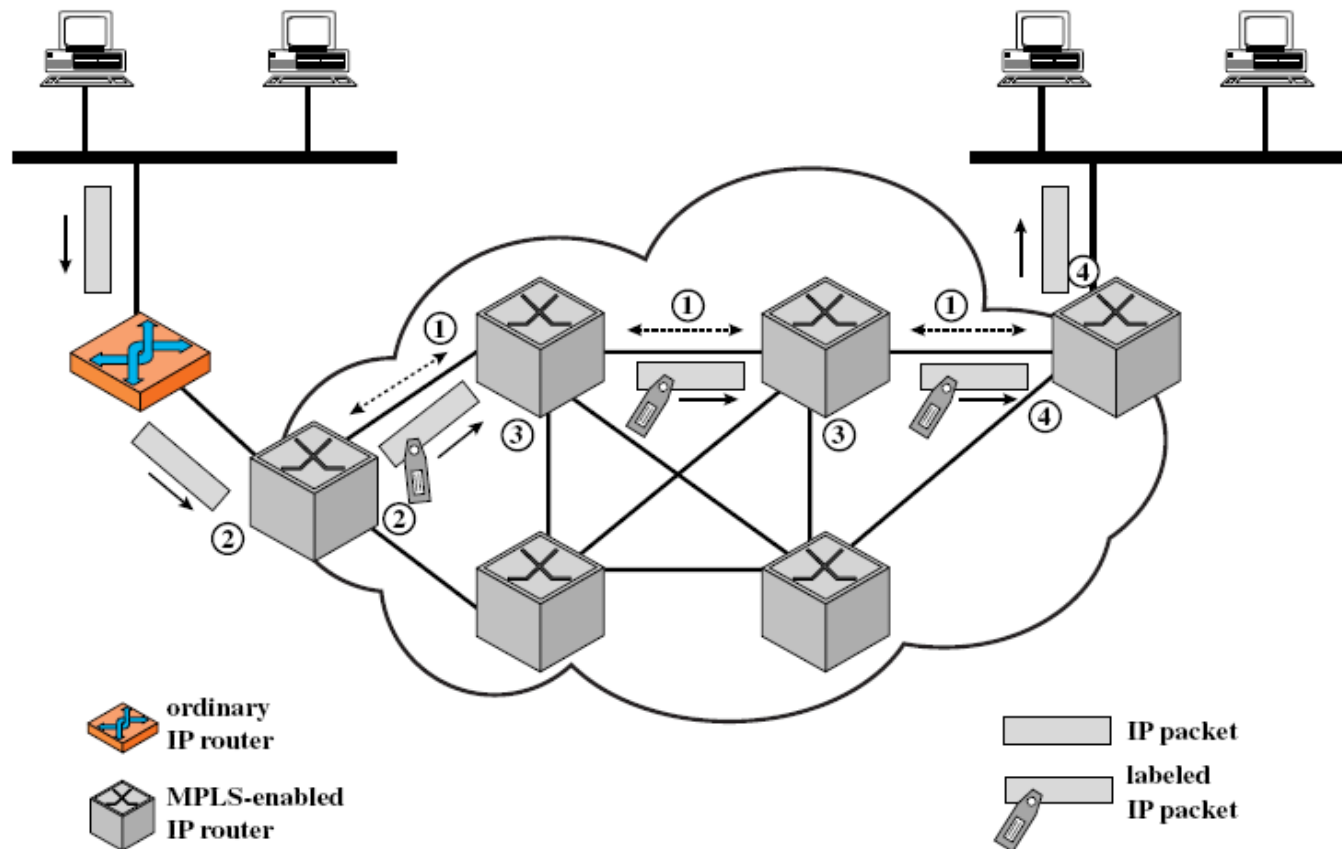
# Paquete Etiquetado

- Paquete en el cual se ha codificado una etiqueta
- La etiqueta se coloca en una cabecera de encapsulación que se ubica entre las cabeceras de capa 2 y capa 3
- La etiqueta puede colocarse en la cabecera de capa 2 o capa 3

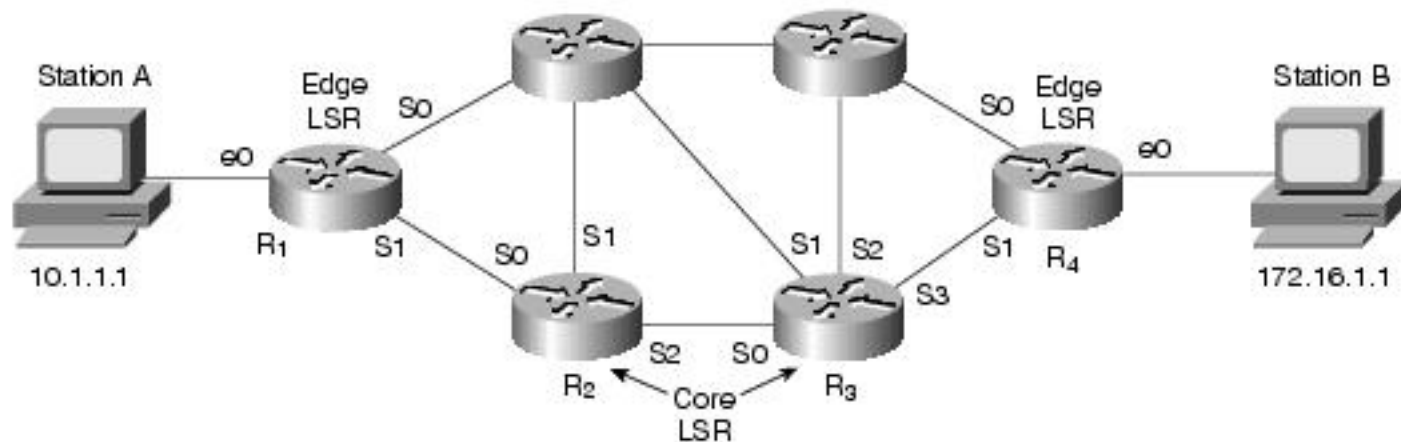
# Funcionamiento de MPLS

- Una red MPLS se compone de un conjunto de nodos (LSRs) capaces de conmutar y enrutar paquetes según una etiqueta asignada
- Las etiquetas definen un flujo de paquetes entre dos extremos
  - En el caso de multicast, entre una fuente y un grupo de destinatarios
- Para cada flujo distinto se define un camino específico a través de la red de LSRs
  - MPLS es una tecnología orientada a la conexión
- Asociada a cada FEC se tiene una caracterización del tráfico que define los requisitos de QoS de ese flujo
- Los LSRs no necesitan examinar o procesar la cabecera IP, sino enviar simplemente cada paquete basándose en el valor de su etiqueta

# Funcionamiento de MPLS



# Funcionamiento de MPLS



Router	Incoming label	Incoming interface	Destination network	Outgoing interface	Outgoing label
R <sub>1</sub>	—	e0	172.16.1	S1	6
R <sub>2</sub>	6	S0	172.16.1	S2	11
R <sub>3</sub>	11	S0	172.16.1	S3	7
R <sub>4</sub>	7	S1	172.16.1	e0	—

# MPLS: Operación

- En los siguientes saltos, no se analiza el encabezado de la capa de red
- La etiqueta es usada como un índice en la tabla
- Las etiquetas viejas son reemplazadas por unas nuevas y el paquete es reenviado al próximo salto



# MPLS: Operación

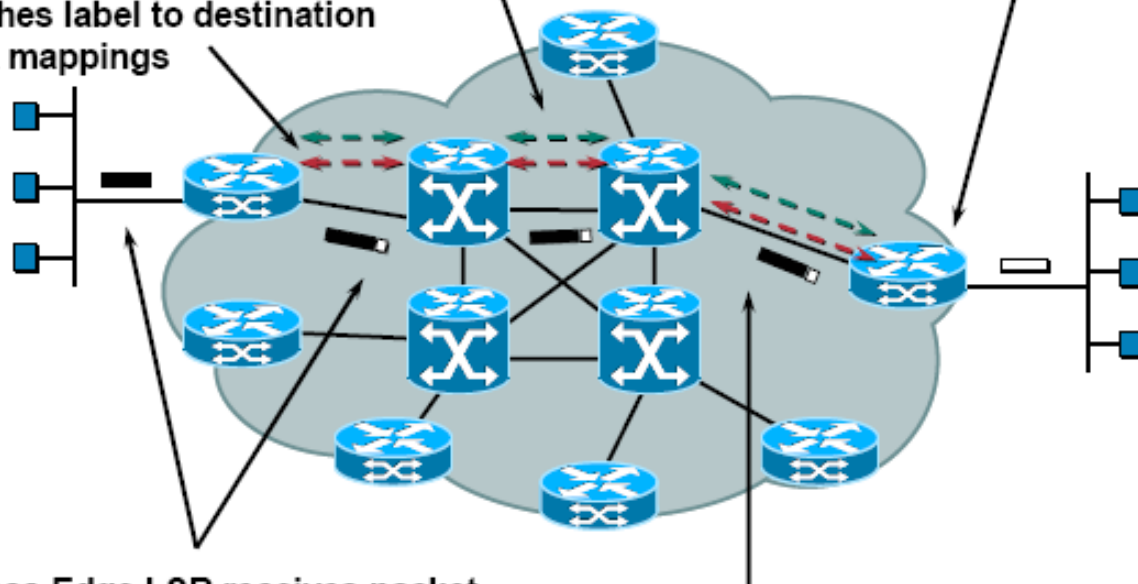
1a. Existing routing protocols (e.g. OSPF, IS-IS) establish reachability to destination networks

1b. Label Distribution Protocol (LDP) establishes label to destination network mappings

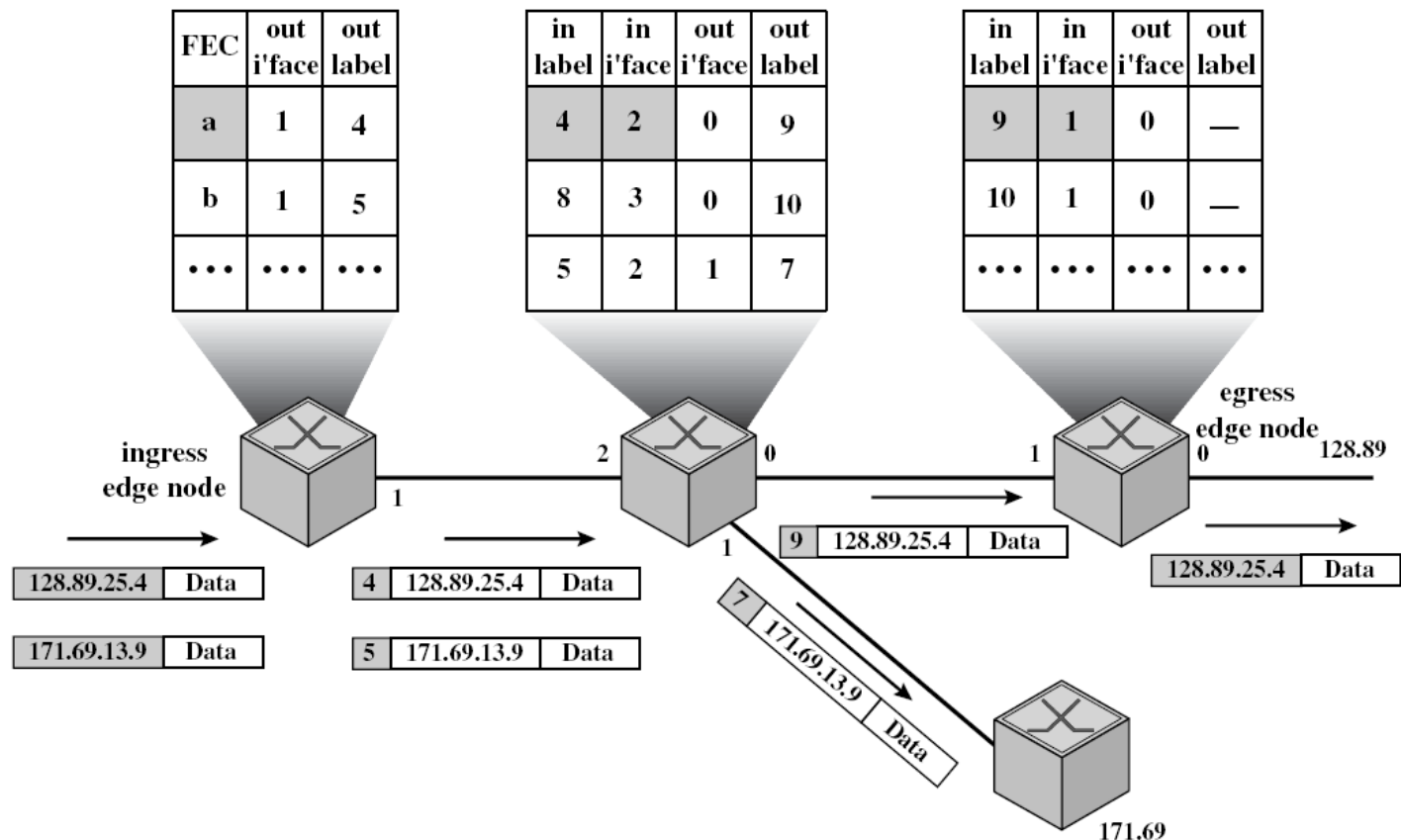
4. Edge LSR at egress removes label and delivers packet

2. Ingress Edge LSR receives packet, performs Layer 3 value-added services, and "labels" packets

3. LSR switches packets using label swapping



# Operación Reenvío de Paquetes



# Operación de MPLS: En Detalle

- Los siguientes pasos deben ser seguidos para que un paquete de datos viaje a través de una red MPLS
  - Creación y distribución de *label*
  - Creación de tablas en cada router
  - Creación de *Label Switched Path* (LSP)
  - Inserción de *labels* y su acceso en tablas
  - Reenvío de paquetes

# MPLS: Consideración

- MPLS es aplicable a cualquier protocolo de la capa de red
- Nos enfocamos en IP

# Etiquetas y Apilado de Etiquetas

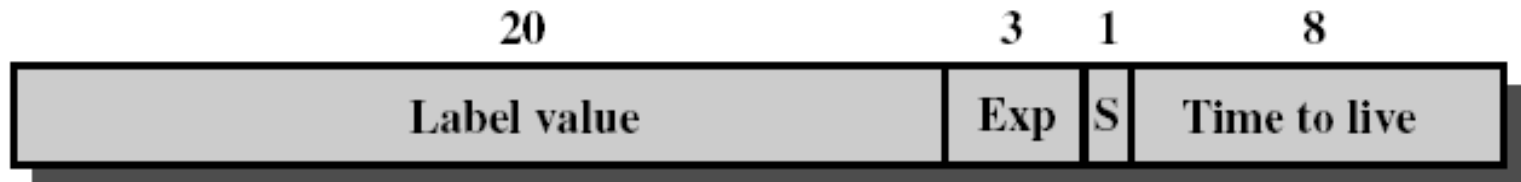
- Una de las características más potentes de MPLS es la capacidad de apilar etiquetas
- Un paquete etiquetado puede tener varias etiquetas, organizadas en una pila LIFO
- El procesamiento siempre toma como referencia la etiqueta superior
- En cualquier LSR, una etiqueta puede ser apilada o retirada
- El apilado de etiquetas hace posible enmascarar múltiples LSPs (Label Switched Path) como uno solo, creando así un túnel

# Etiquetas y Apilado de Etiquetas

- Al comienzo del túnel, un LSR asigna la misma etiqueta a los paquetes de varios LSP introduciendo la etiqueta en cada pila
  - Permite agregar varios LSPs en un LSP (similar al concepto de VP y VCs en ATM)
- Al final del túnel, otro LSR extrae el elemento superior de la pila, revelando la etiqueta interna
- MPLS admite un apilado ilimitado, dentro de las restricciones de tamaño impuestas por la red
- El proceso de etiquetado es completamente independiente del nivel de jerarquía
  - El nivel de la etiqueta no es relevante para el LSR

# Formato de las Etiquetas

bits:



**Exp** = experimental

**S** = bottom of stack bit

# Formato de las Etiquetas

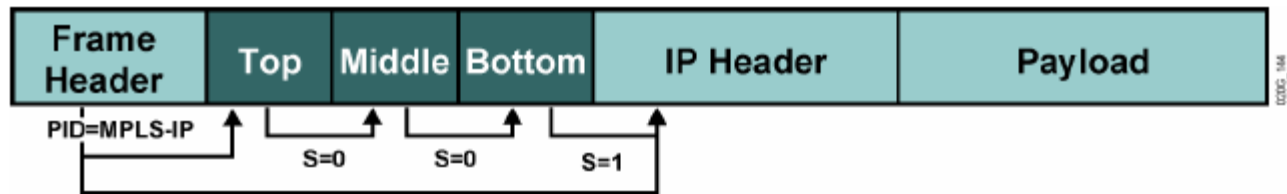
- Label value: etiqueta de 20 bits, de alcance local
- Exp: 3 bits reservados para uso experimental
- S: establecer un 1 para la entrada más antigua de la pila y cero para las demás
- Time to live (TTL): 8 bits que se utilizan para codificar un contador de saltos



# MPLS: Encapsulamiento

- MPLS usa un campo de etiqueta insertado entre los encabezados de la capa 2 y 3
- MPLS en modo ATM usa el encabezado ATM como etiqueta

# MPLS: Formato de la Etiqueta - Uso Campo S



- El identificador del protocolo en la capa 2 especifica que la carga útil inicia con una etiqueta o etiquetas y es seguido de un encabezado IP

# Encabezado MPLS



(a) Data link frame

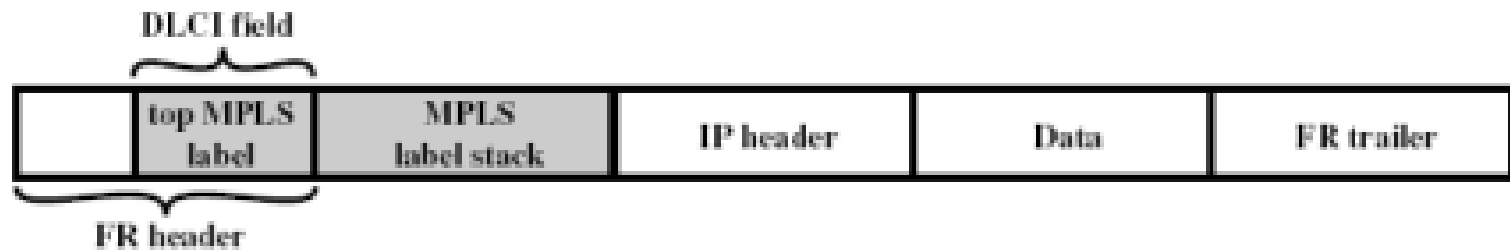


(b) IEEE 802 MAC frame

# Encabezado MPLS



(c) ATM cell



(d) Frame relay frame

# LSP: Label Switched Path

- El camino completo a través de una red de conmutación de etiquetas es denominado LSP
- El LSP es determinado de varias formas:
  - A través de un protocolo de enrutamiento convencional (RIP, OSPF, etc.)
    - La dirección del próximo nodo es correlacionada con la etiqueta, generando el LSP
  - Manualmente

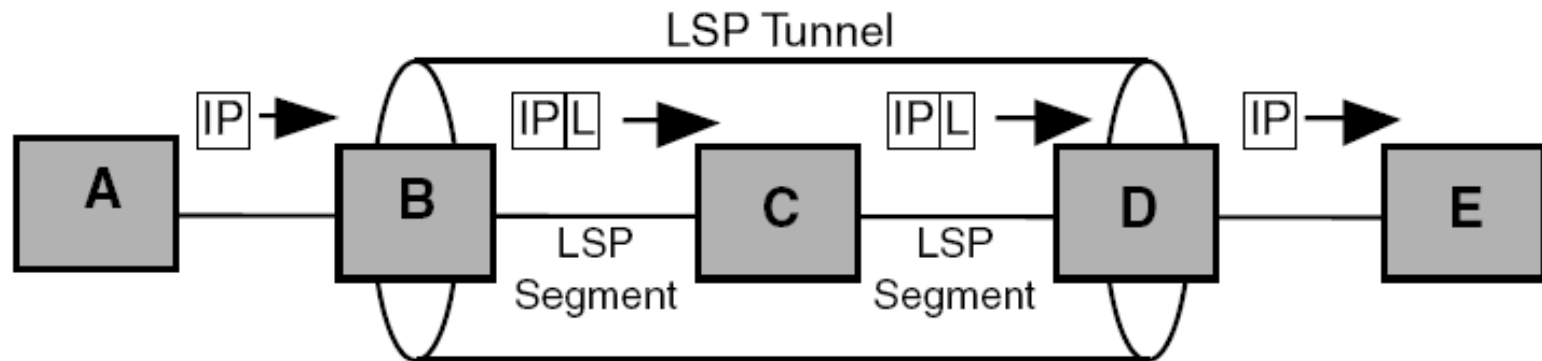
# LSP: Label Switched Path

- Un LSP de nivel  $m$  para un paquete particular  $P$  es una secuencia de routers,  $\langle R_1, \dots, R_n \rangle$  con las siguientes propiedades:
  - $R_1$ , el LSR de ingreso, es el LSR que apila una etiqueta en la pila de  $P$ , resultando en una pila de profundidad  $m$
  - Para todo  $i$ ,  $1 < i < n$ ,  $P$  tiene una etiqueta de profundidad  $m$  cuando es recibido por  $R_i$
  - En ningún momento durante el tránsito de  $P$  de  $R_1$  a  $R_{n-1}$  su pila de etiquetas tiene una profundidad menor a  $m$
  - Para todo  $i$ ,  $1 < i < n$ :  $R_i$  transmite  $P$  a  $R_{i+1}$  a través de MPLS, es decir, usando la etiqueta al tope de la pila

# LSP: Label Switched Path

- El LSP de extremo a extremo es llamado *túnel LSP*, que es una concatenación de cada segmento de LSP entre cada nodo
- Las características del túnel LSP son determinadas mediante una negociación entre los nodos
- Luego de la negociación, el nodo de ingreso define el flujo de tráfico por su elección de la etiqueta
- A medida que el tráfico es enviado por el túnel, la idea no es examinar el contenido de otras cabeceras, sino sólo la cabecera de etiquetas
- El resto del tráfico es “enviado por el túnel” a través del LSP sin ser examinado ni alterado
- Al final del túnel LSP, el nodo de egreso elimina la etiqueta y entrega el tráfico de capa 3 al siguiente nodo

# LSP: Label Switched Path



**IPL** = Label (L) and IP, including other headers and user traffic



# Métodos de Asignación de Etiquetas

- Existen varios enfoques para definir los modos de asignación de etiquetas
  - De acuerdo al punto donde se asigna la etiqueta en el LSR
  - De acuerdo al LSR que la asigna

# Métodos de Asignación de Etiquetas

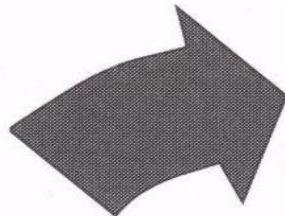
- De acuerdo al punto donde se asigna la etiqueta en el LSR
  - Las etiquetas pueden ser asignadas entre LSRs por dos métodos
  - Para explicar esta idea se introduce el término *espacio de etiquetas* (label space) para referirse a la forma en la que una etiqueta se asocia con un LSR

# Métodos de Asignación de Etiquetas

- De acuerdo al punto donde se asigna la etiqueta en el LSR
- Espacio de Etiquetas por Interfaz*

## Per Interface Label Space

Label space  
1 - 5000



Label space  
1 - 5000

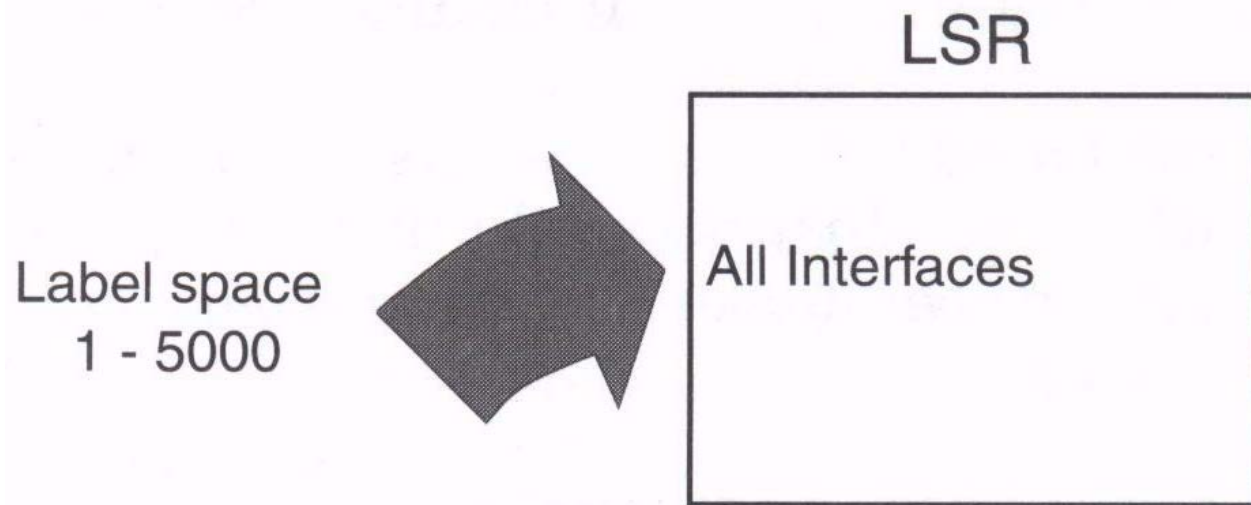
LSR

a interface

b interface

# Métodos de Asignación de Etiquetas

- De acuerdo al punto donde se asigna la etiqueta en el LSR
  - Espacio de Etiquetas por Plataforma*  
**Per Platform Label Space**



# Métodos de Asignación de Etiquetas

- De acuerdo al LSR que asigna la etiqueta
  - Este enfoque se refiere a cuál LSR de un par dado será el encargado de asignar las etiquetas a usar en el tráfico que se intercambiará entre uno y otro
  - Para un flujo dado de datos, el LSR que debe interpretar las etiquetas en los paquetes del flujo recibido del otro es el LSR de *downstream*
  - El LSR que coloca las etiquetas en los paquetes del flujo que se envía a otro es el LSR de *upstream*

# Métodos de Asignación de Etiquetas

- De acuerdo al LSR que asigna la etiqueta
  - *Downstream*: se genera una mínima cantidad de negociación de etiquetas, ya que el LSR que debe interpretarlas es responsable de su asignación
  - *Upstream*: el hardware de conmutación puede obtener ganancias significativas al poder utilizar la misma etiqueta en distintas interfaces para tráfico multicast

## LIB y LFIB

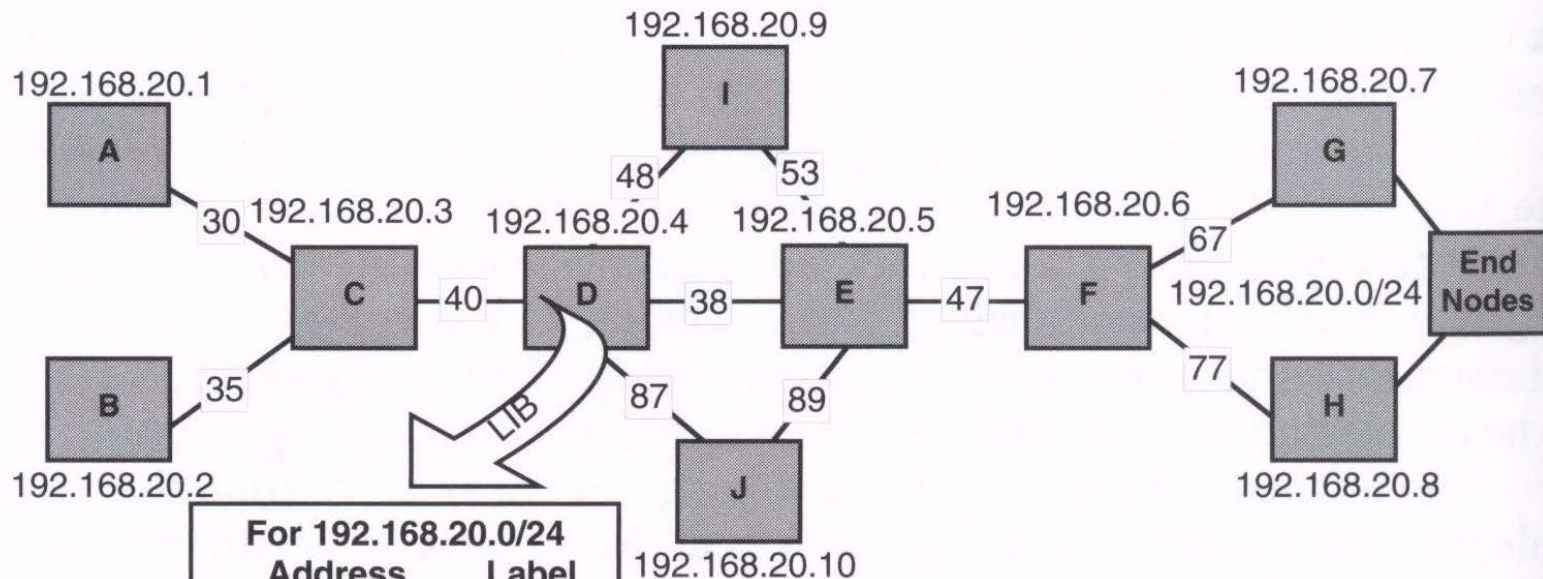
- La LIB (*Label Information Base*) es una base de datos que almacena información acerca de los posibles destinos y la forma de alcanzarlos
  - Se almacena la información sobre las direcciones en esta especie de “tabla de rutas” y se asigna una etiqueta a cada una
- La LFIB (*Label Forwarding Information Base*) es usada para determinar cómo procesar los paquetes entrantes etiquetados, tal como determinar el próximo nodo que debe recibir el paquete
  - La LFIB es para MPLS lo que la tabla de rutas es para IP

## LIB y LFIB

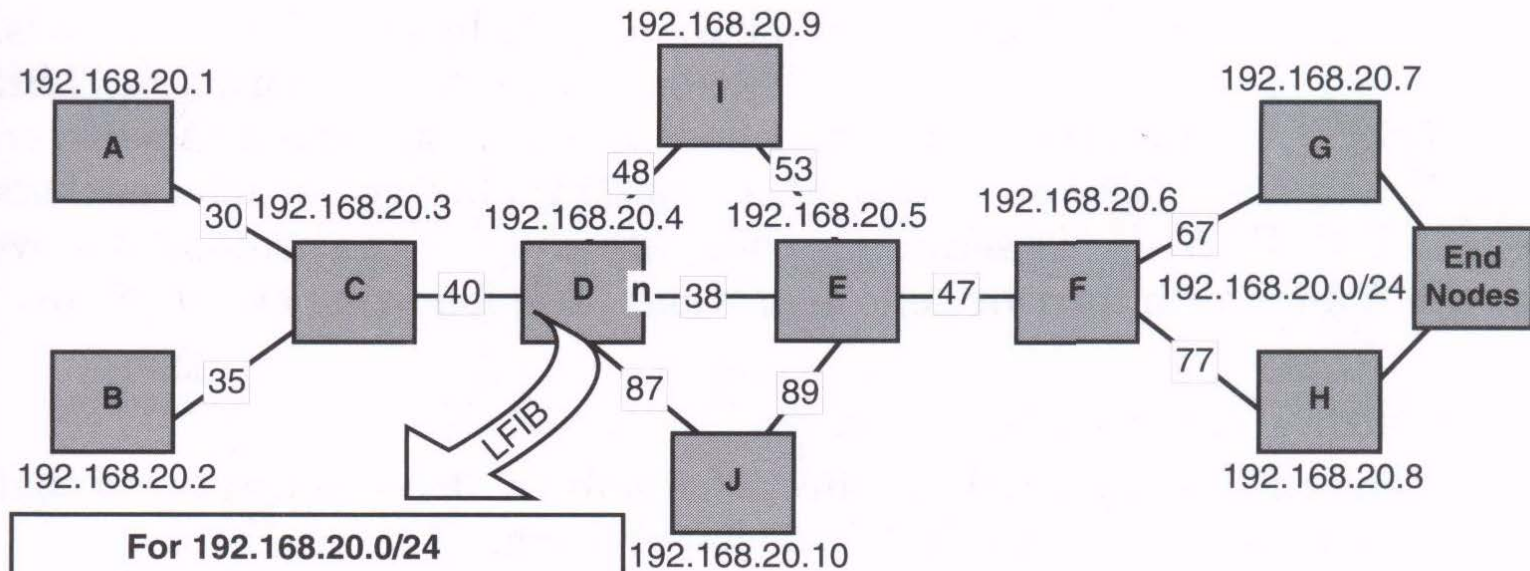
- *Los términos LIB y LFIB son usados por Cisco, sin embargo, existen otros términos empleados para definir estos elementos de información*



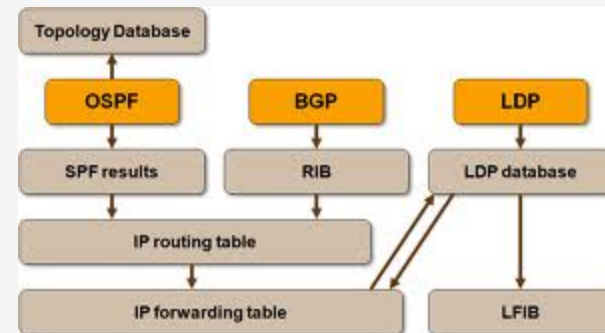
# Ejemplo LIB



# Ejemplo LFIB



# Label Distribution Protocol (LDP)



# LDP

- MPLS debe proveer un método estándar para la distribución de etiquetas de enrutamiento entre LSRs vecinos
- Descrito en el RFC 3036

# LDP

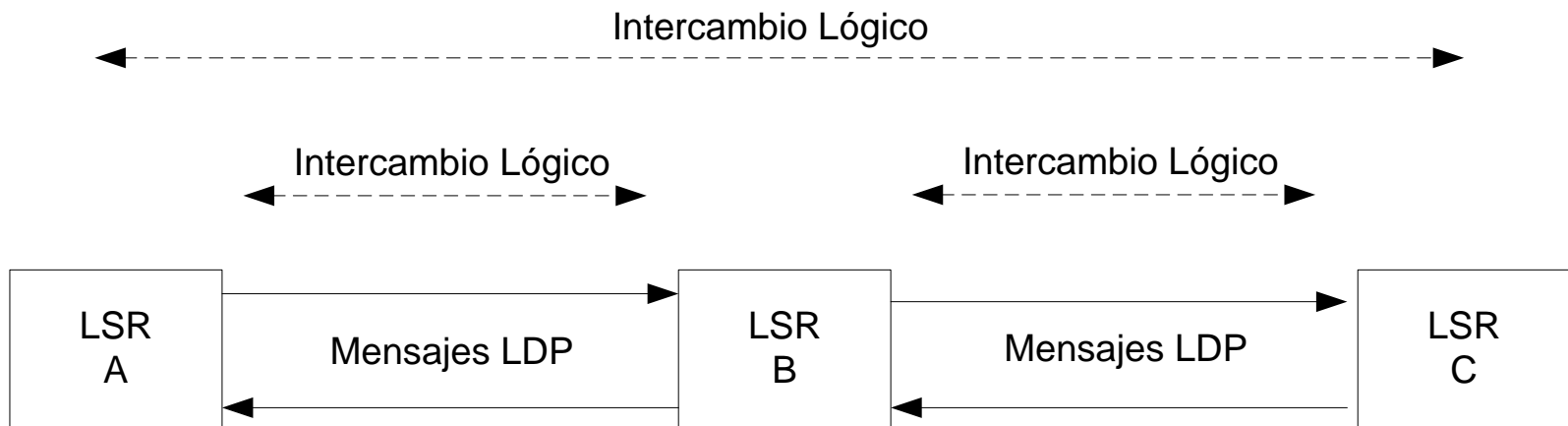
- Creado específicamente para distribuir etiquetas
- No intenta ejecutar funciones de enrutamiento
  - Esto lo deja a IGP
- LDP incluye un conjunto de mensajes y procedimientos por el cual LSRs establecen LSPs a través de la red:
  - Mapeando directamente información de la capa de red a caminos conmutados de la capa de enlace de datos

# LDP

- MPLS no toma decisiones de reenvío por cada datagrama de capa 3 recibido
  - Se determina una FEC de datagramas capa 3, y se negocia una etiqueta de longitud fija entre LSRs vecinos en un LSP de ingreso a egreso
- Los LSRs deben ser capaces de determinar cuáles de sus vecinos tienen soporte de operaciones MPLS, y deben acordar los valores de las etiquetas que se usarán para el tráfico de datos de usuario
- LDP es usado para dar soporte a este requerimiento

# LDP

- Intercambio de mensajes



# LDP: Características

- Provee un mecanismo de *descubrimiento* de LSRs vecinos que les permite ubicarse y establecer comunicación
- Define cuatro clases de mensajes:
  - DISCOVERY
  - ADJACENCY
  - ADVERTISEMENT
  - NOTIFICATION
- Se ejecuta sobre TCP para brindar confiabilidad en la entrega de los mensajes
- Fue diseñado para ser fácilmente extensible, usando mensajes especificados como una colección de objetos codificados en formato TLV (Type-Length-Value)

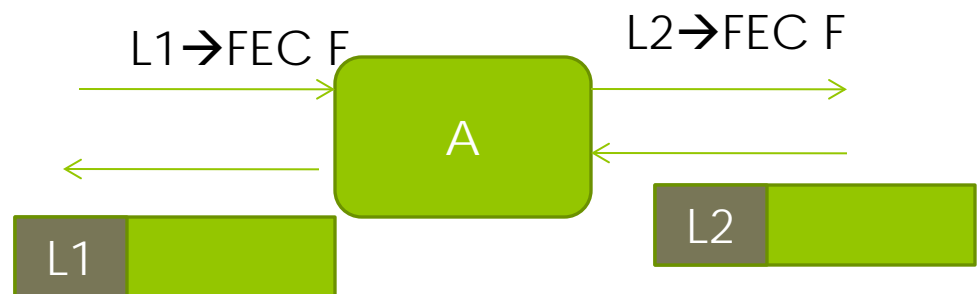


# LDP

- ◉ Distribución de etiquetas:
  - ◉ Crea asociaciones entre etiquetas y FECs
  - ◉ Construir tablas de *forwarding* con etiquetas de entrada y salida
- ◉ Asignación de etiquetas *downstream*:
  - ◉ Cada router espera recibir tráfico con la etiqueta escogida localmente
  - ◉ El tráfico fluye en dirección contraria a la distribución de la etiqueta

# LDP

- Por ejemplo,
  - A recibe  $L1 \rightarrow FEC\ F$
  - A anuncia  $L2 \rightarrow FEC\ F$
  - Entonces, A espera recibir el tráfico destinado a FEC F con etiqueta L2
  - A envía tráfico para FEC F con etiqueta L1



# LDP: Mensajes

- ◉ **DISCOVERY:** anunciar la presencia de un LSR adyacente
  - ◉ hello
- ◉ **ADJACENCY:** inicialización, mantenimiento y cierre de las sesiones entre LSRs
  - ◉ INITIALIZATION, KEEPALIVE
- ◉ **ADVERTISEMENT:** anuncio de asociación de etiquetas, solicitudes y liberaciones
  - ◉ LABEL MAPPING, LABEL REQUEST
- ◉ **NOTIFICATION:** proveer información de notificación de errores

# LDP: Mensajes

- Discovery messages:
  - Se envían mensajes de *Hello* periódicos
  - Se encapsula en UDP al puerto LDP
  - Se envía al grupo multicast de la subred (todos los routers)

## LDP: Mensajes

- Se puede ahora establecer una sesión con LSRs descubiertos anteriormente
  - Se usa una conexión TCP
- Una vez establecida la sesión las entidades LDP pueden intercambiar anuncios de etiquetas y asociaciones

# LDP: Mensajes

- La asignación es local al LSR:
  - Un LSR pide una asignación de etiqueta de un LSR vecino cuando lo necesita
  - Un LSR anuncia una asignación de etiqueta a un LSR vecinos cuando se desea que el vecino use una etiqueta
- LDP utiliza TCP para todo, excepto para el mecanismo de descubrimiento

# LDP: Mensajes

- Mensajes comunmente usados:
  - INITIALIZATION
  - KEEPALIVE
  - LABEL MAPPING
  - LABEL RELEASE

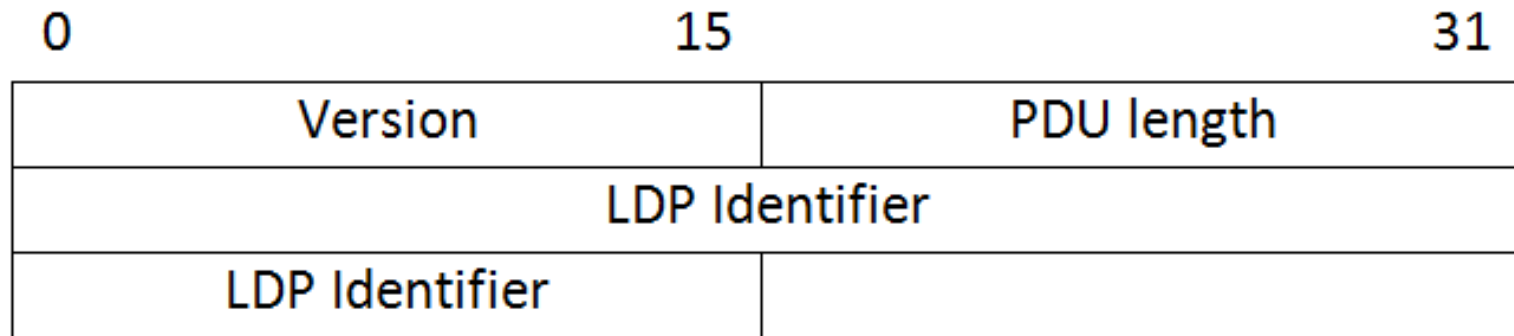
# Espacios de Etiquetas e Identificadores

- Los espacios de etiquetas en LDP son los mismos que en MPLS
  - Por interfaz
  - Por plataforma
- Un espacio de etiquetas es descrito por un identificador de 6 octetos
  - Los primeros 4 octetos identifican un LSR y deben tener un valor único global
  - Los últimos dos octetos identifican el espacio de etiquetas dentro del LSR



# Encabezado LDP

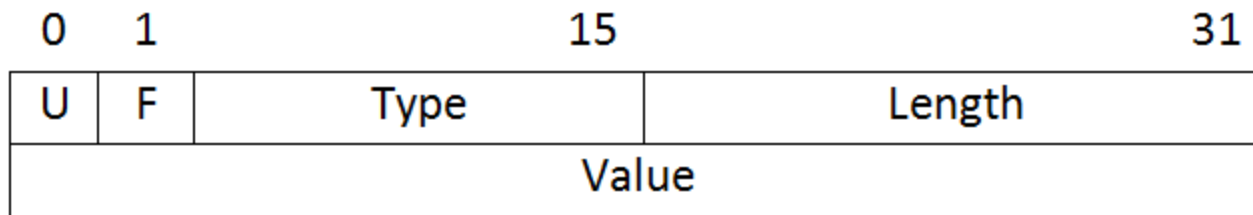
- Cada PDU de LDP comienza con una cabecera, seguida de uno o más mensajes LDP



# Encabezado LDP

- Version: el número de la versión del protocolo
- PDU length: longitud total del PDU en octetos, sin incluir los campos de versión y longitud
- LDP ID: identificador del espacio de etiquetas del LSR emisor de este mensaje

# Codificación TLV



# Codificación TLV

- En la recepción de un TLV desconocido:
  - Si el bit U (unknown) está fijado en 0, se debe devolver una notificación al origen y el mensaje debe ser descartado
  - Si el bit U está fijado en 1, el TLV desconocido es ignorado de forma silenciosa y el resto del mensaje es procesado como si el TLV desconocido no existiese
  - El bit F (forward unknown) sólo aplica cuando el bit U está fijado en 1 y el mensaje LDP que contiene el TLV desconocido debe ser reenviado.
    - Si F=0, el TLV desconocido no se reenvía con el mensaje
    - Si F=1 el TLV desconocido es reenviado con el mensaje

# Formato del Mensaje LDP

0	1	15	31
U	Message type	Message Length	
Message ID			
Mandatory parameters			
Optional parameters			

# Formato del Mensaje LDP

- Bit U(unknown): si está fijado en 1 y es un mensaje desconocido por el receptor, el mensaje es descartado de forma silenciosa
- Message type: tipo del mensaje
- Message length: longitud de los campos *Message ID*, *Mandatory parameters* y *Optional parameters*
- Message ID: identificador unívoco del mensaje
- Mandatory parameters: Parámetros obligatorios. Dependen del tipo de mensaje usado
- Optional parameters: Parámetros opcionales. Dependen del tipo de mensaje usado