Capítulo 3

La Capa de Enlace

Autor: Santiago Felici

Fundamentos de Telemática

(Ingeniería Telemática)

Sumario

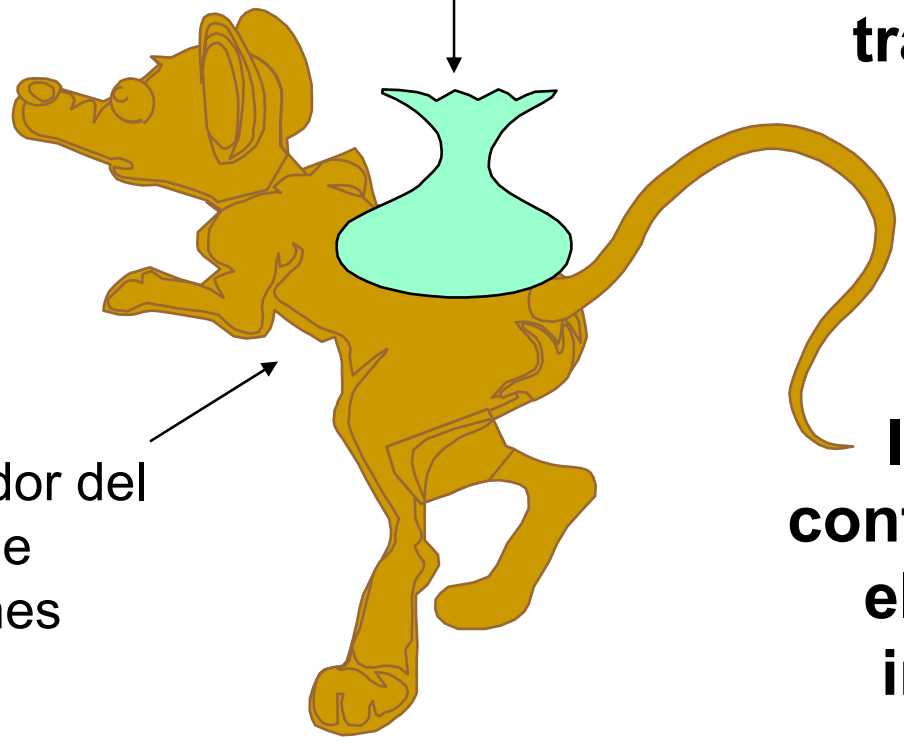
- **Funciones de la capa de enlace**
- Control de errores
- Control de flujo
- Protocolos de la capa de enlace
- Protocolos de nivel de enlace: SDLC/HDLC y PPP
- Ejemplos de tecnologías de capa 1 y 2 (protocolos WAN): X.25, Frame-Relay, ATM

Capa de Enlace

Provee el control de la capa física

Datos de la capa superior

Detecta y/o corrige Errores de transmisión

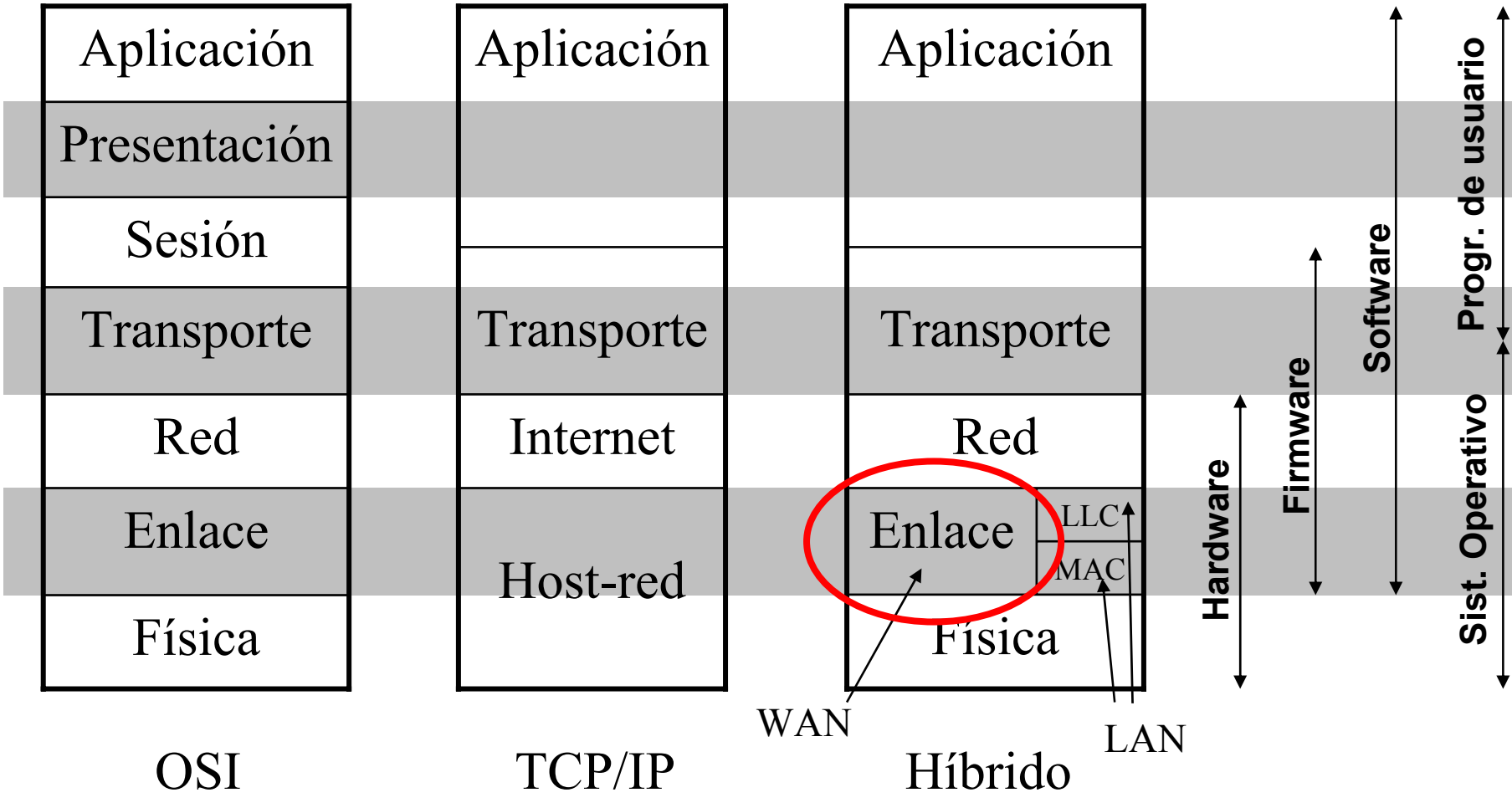


Driver o controlador del dispositivo de comunicaciones

Implementa control de flujo en el envío de la información

Capa 2

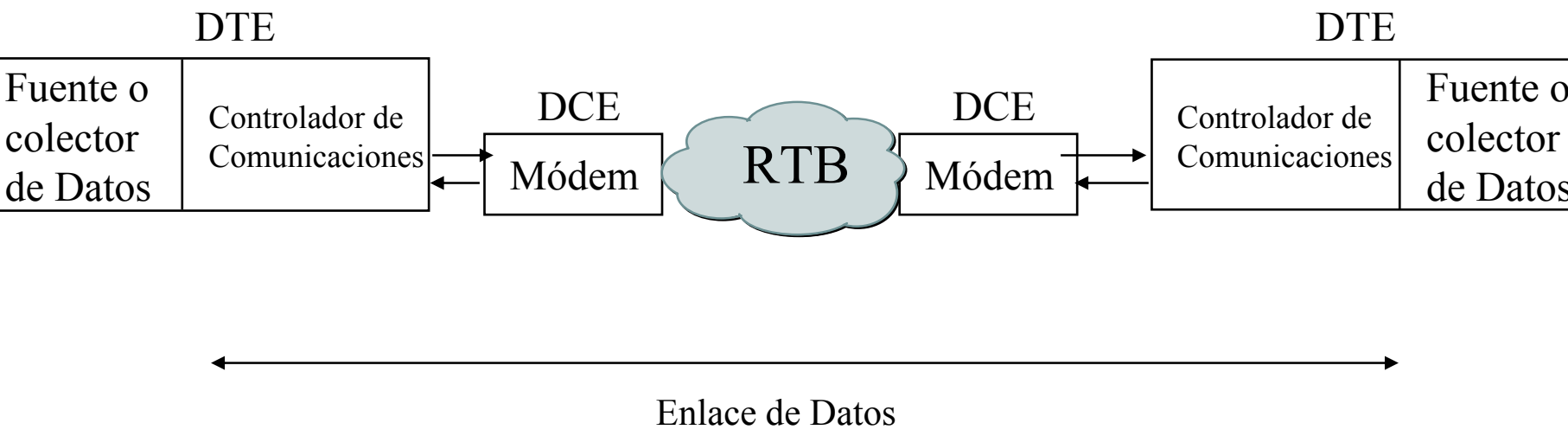
Comparación de modelos OSI, TCP/IP e híbrido



Host-Red o también conocida como de “Acceso a la Red”

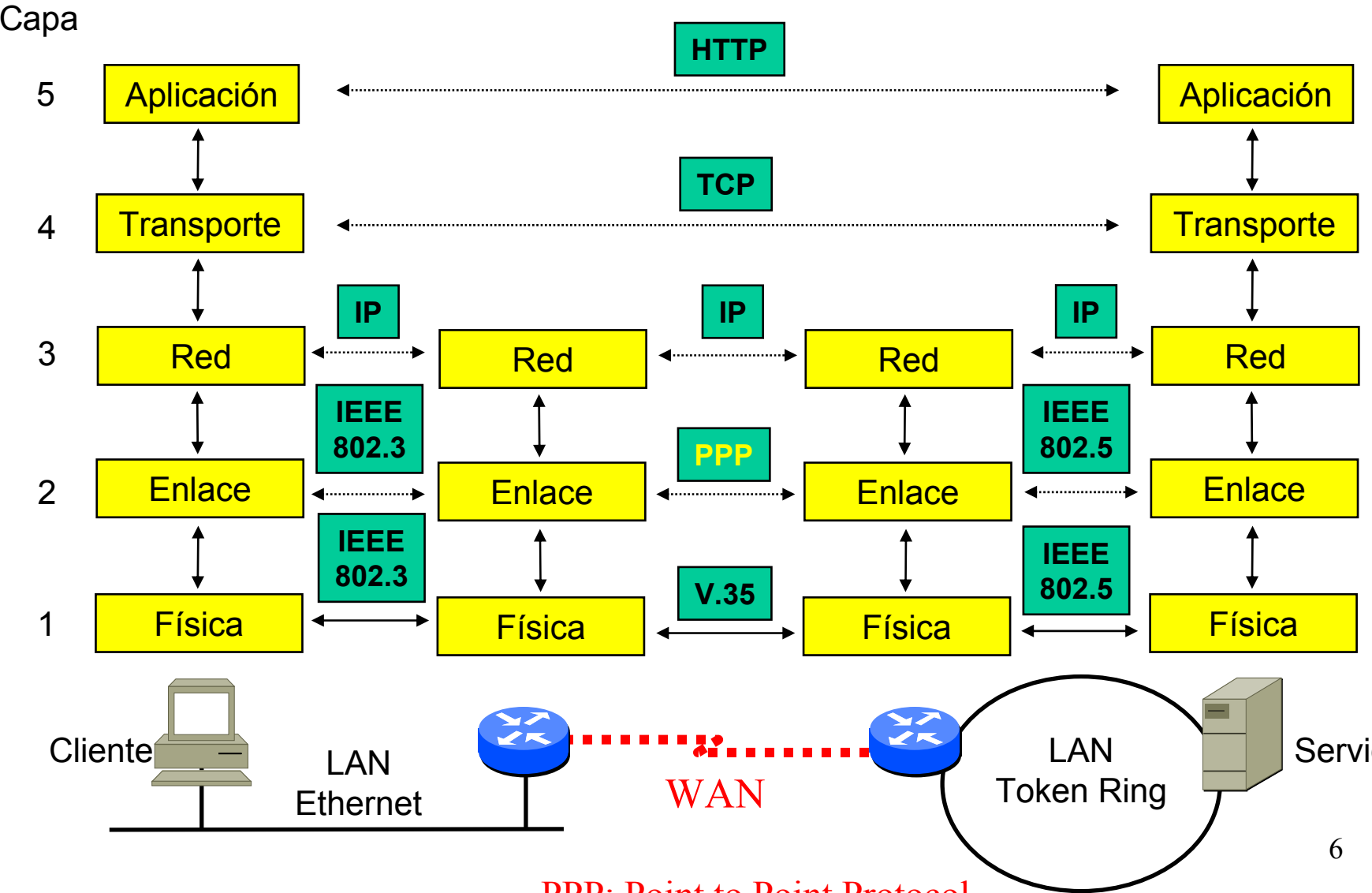
Esquema de conexión para Enlace de Datos

Objetivo: transmisión fiable de tramas entre equipos directamente conectados.



RTB: red telefónica básica o analógica (tradicional)

Acceso a un servidor Web a través de una conexión remota



PPP: Point to Point Protocol

Funciones de la capa de enlace

- Obligatorias:
  - Identificar **tramas** (*agrupación de bits que se intercambia a nivel de enlace*)
  - Detección de errores
- Opcionales (servicio orientado a conexión):
  - Control de flujo
  - Corrección de errores

Técnicas de identificación de tramas

Las tramas se delimitan por diferentes métodos:

- **Contador de caracteres**; posibles problemas por pérdida de sincronismo.
- **Caracteres de inicio y final** con caracteres de relleno. Normalmente ASCII “DLE” “STX” para inicio y “DLE” “ETX” para final, con DLE de relleno.
- **Secuencia de bits** indicadora de inicio y final, con bits de relleno; normalmente 01111110 (0x7E); si en los datos aparecen 5 bits seguidos a 1 se intercala automáticamente un 0.
- **Violaciones de código** a nivel físico.

Códigos ASCII: DLE (0x1C) data link escape, STX (0x02) start text, ETX(0x03) end text.

Ejemplo de bit de relleno para evitar el código 0x7E

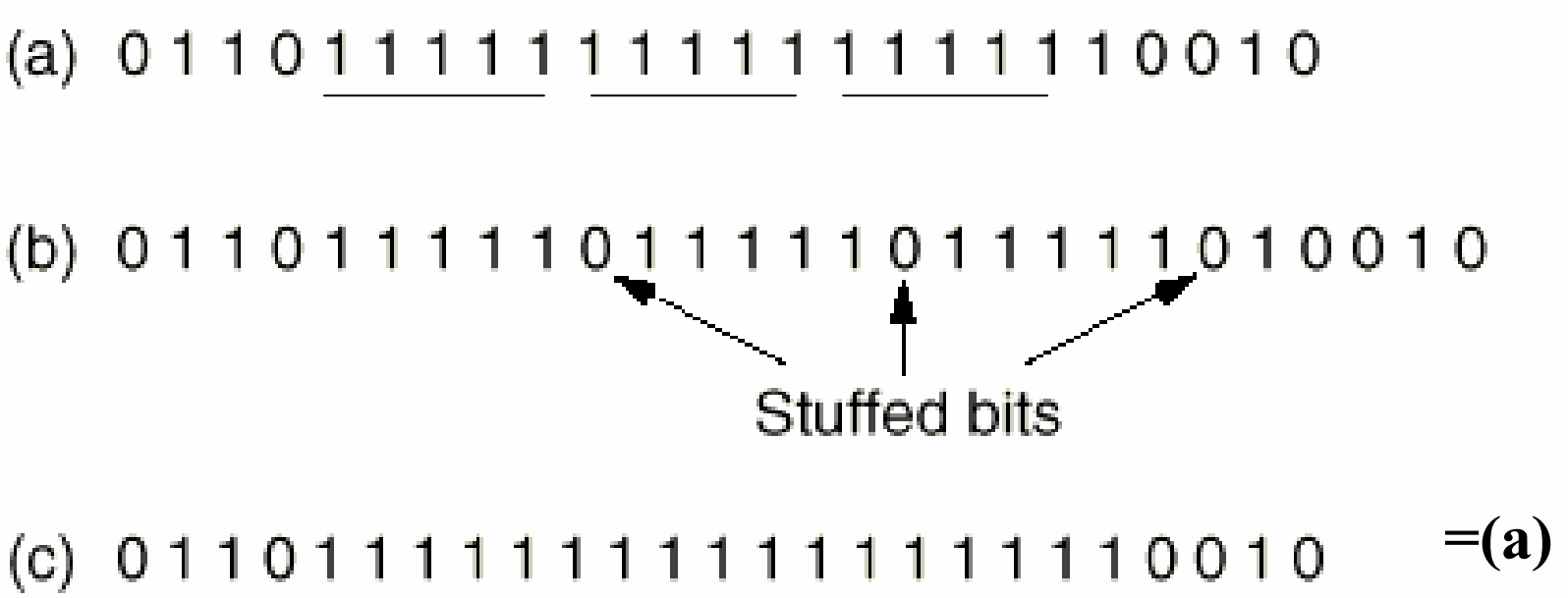


Fig. 3-5. Bit stuffing. (a) The original data. (b) The data as they appear on the line. (c) The data as they are stored in the receiver's memory after destuffing.

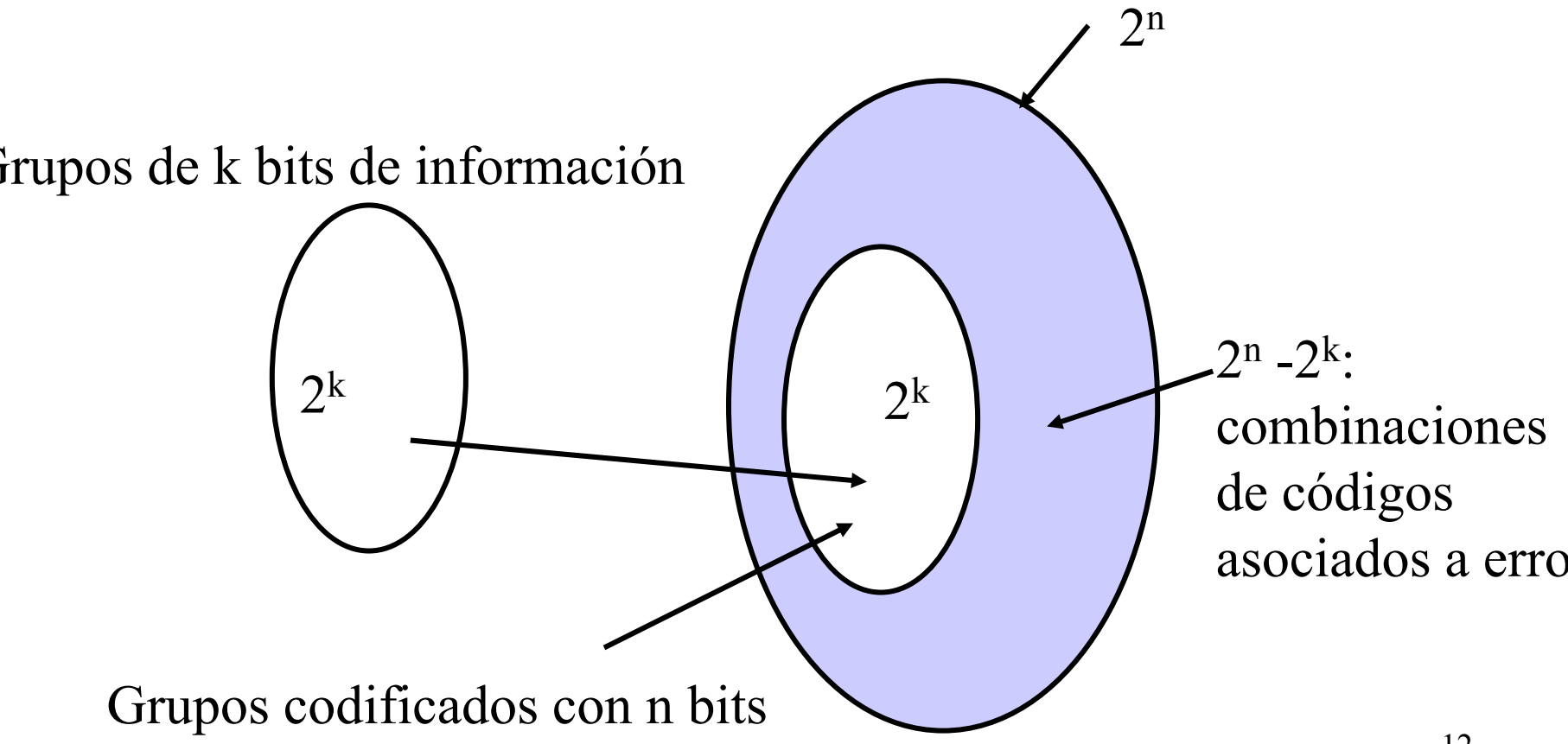
Sumario

- Funciones de la capa de enlace
- **Control de errores**
- Control de flujo
- Protocolos de la capa de enlace
- Protocolos de nivel de enlace: SDLC/HDLC y PPP
- Ejemplos de tecnologías de capa 1 y 2 (protocolos WAN): X.25, Frame-Relay, ATM

Códigos y distancia Hamming

- Los datos a enviar se codifican en un formato especial (códigos), que consiste normalmente en añadir información adicional (*overhead*) al final. Es decir, que si los datos son "k" bit y se añaden "n-k" bit adicionales, enviando "n" bits, n=k+(n-k). *Este tipo de código se conoce como códigos bloque (n,k) con n>k, que operan bloque de bits a bloque de bits*
- Con "k" bit las combinaciones en la fuente de posibles códigos son 2^k y con "n" bit, son 2^n. El ***objetivo*** de la codificación es hacer corresponder unívocamente a cada uno de los 2^k un único valor de los 2^n.
- En una transmisión, pueda ser que alguno de los "n" bit se alteren, por tanto, si los errores introducidos (bit erróneos) hacen que el código sea uno de los 2^n- 2^k, podremos saber que existe error.
- Se llama **distancia Hamming (***llamada* **"d")**, el mínimo número de bits diferentes que pueden tener dos códigos.

Códigos (n,k)

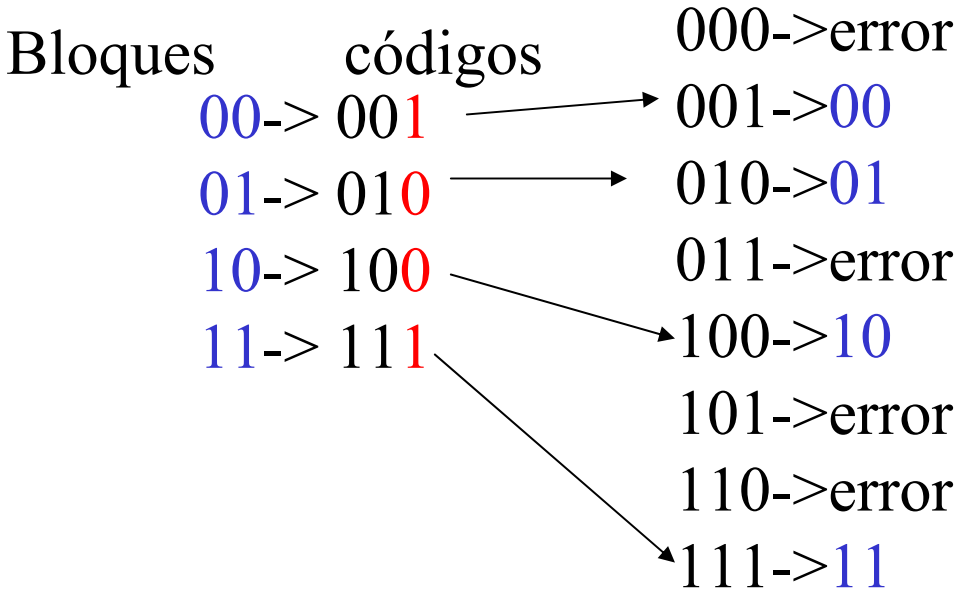

2^n
Grupos de k bits de información
2^k
2^k
2^n -2^k:
combinaciones
de códigos
asociados a erro
Grupos codificados con n bits

Códigos de control de errores

- Los códigos en función de la distancia de Hamming (d) pueden ser:
  - **Detectores**: sólo permiten detectar "d-1" errores , p. ej. *CRC (Cyclic Redundancy Check).* Este tipo de códigos se llaman *BEC Backward Error Correction*, de forma que tienen que solicitar al emisor, hacia atrás (*back*) el reenvío.
  - **Correctores:** permite corregir "⌊(d-1)/2⌋" errores . Este tipo de códigos también se llaman *FEC* del inglés *Forward Error Correction.* p. ej. *RS (Reed-Solomon).*
- **Ejemplo**: si la distancia de Hamming es 5, podremos detectar 4 errores per corregir sólo 2.
- *Los códigos detectores tienen menos overhead, pues necesitan incorporar menos redundancia.*
- *Los códigos correctores se utilizan bien en conexiones simplex, bien en multicast o bien en tiempo real.*
- La redundancia de un código se define como=(n-k)/n

Ejemplo de código detector basado en paridad impar

Si a nuestro código (bloques de 2 bits) introducimos paridad impar, es decir añadimos un bit para obtener un número impar de 1s, la distancia Hamming obtenida es de 2, y por tanto sólo podemos detectar un bit erróneo.



Combinaciones con 3 bits, tamaño del código, de forma son palabras bloque aquellas que estén codificadas correctamente

Ejemplo de código corrector

Supongamos la codificación basada en 4 códigos de 10 bits: 00000 00000, 00000 11111, 11111 00000, 11111 11111.

Este código tiene una distancia Hamming de 5, por lo cual puede corregir errores dobles.

Si recibimos 00000 00111, con un doble error, por tanto la palabra código válida más próxima sería 00000 11111.

Si recibimos 00000 00111, con un triple error, es decir proveniente de 00000 00000, no se podría corregir.

Tasa de errores (BER)

- La tasa de errores de un medio de transmisión se mide por la BER (Bit Error Rate) que se define como:

BER = bits erróneos / bits transmitidos

Medio físico	**BER típico**
Fibras ópticas	$< 10^{-12}$
LANs de cobre, Radioenlaces fijos (microondas)	$< 10^{-8}$
Enlaces telefónicos, satélite, ADSL, CATV	$<10^{-5}$
GSM	$>10^{-5}$

Pregunta: si en GSM transmitidos a 9600 bps y BER= 10^{-5}, ¿cuánto tiempo tiene que pasar para que falle un bit?

Estrategias de control de errores en la capa de Enlace

El propio BER del canal me determina el tipo de código a utilizar, si corrige/detecta 1, 2 o 3 bits.

Las estrategias utilizadas son:

- Tasa de error baja/muy baja: código detector sin reenvío de tramas erróneas (si acaso se hará a nivel de transporte): BEC (backward error correction)
- Tasa de error alta/muy alta: código detector con reenvío de tramas erróneas. *El reenvio ralentiza el proceso de comunicación, no aconsejable para tiempo real:* : BEC (backward error correction)
- Tasa de error alta/muy alta con canal simplex o envío broadcast (p. ej. TV digital): código corrector: : FEC (forward error correction)

Ejemplo, un RS (ReedSolomon) con 10% de overhead puede mejorar el BER en 10^{-4} (p. ej. de 10^{-5} a 10^{-9})

Control/detección de errores

- Los medios de Transmisión son fuentes de ruido que degeneran la señal procedente del emisor
- La subsanación de dichos errores se puede realizar en diferentes niveles:
  - NIVEL DE ENLACE:
    - El emisor envía información adicional en la trama enviada al nivel inferior junto con los datos recibidos de los niveles superiores.
      - Bits de **PARIDAD** (Horizontal, Vertical)
      - Bits de **CRC** (Código de Redundancia Cíclica)
  - NIVELES SUPERIORES: **Checksum**

Paridad horizontal

- Esta técnica es muy simple de implementar, pero poco robusta, se llama paridad PAR ó paridad IMPAR. Con ello se consigue distancia Hamming de 2.

  Ejemplo: cada carácter ASCII se compone de 8 bits

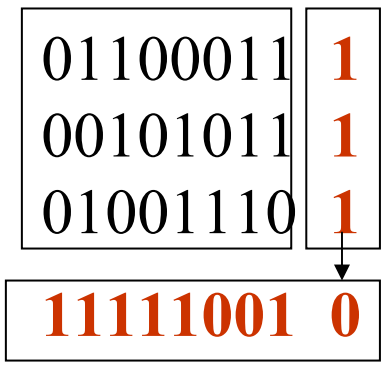
  0x61 --> 0110 0001

  - SI paridad par ----> deben ir número pares de 1's
    - 01100001 1
  - SI paridad impar ----> deben ir número impares de 1's
    - 01100001 0

Paridad vertical

- Esta técnica es para evitar ráfagas de error en la transmisión de grandes bloques

  Ejemplo: a cada cadena de caracteres ASCII de una trama se le asocia una paridad (par/impar) (paridad horizontal) y tras ello, se le asocia paridad a todos los bits de mismo peso de la trama (paridad vertical)



Ej: paridad impar en horizontal y vertical

***Nota**: tiene la posibilidad de corregir un bit erróneo, detectado por fila y columna*

CRC: Cyclic Redundancy Check (1/2)

- Se utiliza un algoritmo matemático basado en **álgebra modular** y su implementación se realiza a través de circuitos integrados, lo cual permite gran velocidad para comprobar la integridad de los datos recibidos.
- Son códigos detectores.
- Los bits de una trama se representan como coeficientes de un polinomio, de forma que los k-bit de mensaje generan un polinomio de grado *k-1* = $x^k + \ldots + x^2 + x^1 + x^0$

  Ej: si la trama es 100011, el polinomio es: $x^5+0x^4+0x^3+0x^2+x^1+x^0$
- Se agregan *n-k* bits de redundancia a los *k*-bit del mensaje (interesa que *n>>k).*
- Se define un polinomio divisor C(x), también conocido como generador, de grado *n-k* con unas propiedades especiales para la detección. El estudio de estas propiedades queda fuera de los objetivos de esta asignatura.

  Ej. C(x) = $x^3 + x^2 + 1$ *o en binario 1101*

CRC: Cyclic Redundancy Check (2/2)

- **Pasos a seguir**: Se genera el polinomio $P(x)$ a partir del mensaje a transmitir en forma de polinomio $m(x)$ tal que sea divisible en forma exacta por $C(x)$, de la forma siguiente ***utilizando siempre operaciones XOR para la resta*** (0 XOR 0= 0, 1 XOR 1= 0 , 1 XOR 0 =1, es decir bits iguales, 0 y diferentes 1)
  1) se corre a la izquierda n-k bits=> $m(x)x^{n-k}$
  2) restar el resto polinomial de ($m(x)x^{n-k}$ / $C(x)$) (que también es conocido como CRC) a $m(x)x^{n-k}$, que es equivalente a cambiar los n-k primeros 0s de $m(x)x^{n-k}$ por dicho resto (o CRC), resultando $P(x)$. **Este procesamiento se realiza con XOR y por tanto el resultado no guarda ninguna relación con la aritmética tradicional**.
- El polinomio $P(x)$ contiene el mensaje y es el valor transmitido.
- En general se recibe el polinomio $P(x) + e(x)$, siendo $e(x)$ el polinomio de error. $e(x) = 0$ implica ausencia de errores
- Se divide ($P(x) + e(x)$) por $C(x)$. Si el resto es 0, será si:
  - $e(x)$ fue cero (ningún error), o
  - $e(x)$ es exactamente divisible por $C(x)$ (las propiedades de C(x) deberían evitar dicha divisibilidad)
- *Esto supone escoger un polinomio divisor C(x) adecuado*

CRC: Seleccionando $C(x)$

Propiedades para detectar:

- Todo error simple, x^{n-k} y x^0 deben tener coeficiente no cero.
- Todo error doble, $C(x)$ debe contener un factor con al menos tres términos
- Cualquier número impar de errores, $C(x)$ debe contener el factor $(x + 1)$
- Detecta cualquier "ráfaga" de errores ... esto requiere de un análisis más detallado que se detalla en otras asignaturas, *concretamente "Teoría de la Información y la Codificación" de 3º de Ing. Telemática*

Polinomios más usados:

- CRC-16: $x^{16}+x^{15}+x^2+1$
- CRC-32: $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

Ejemplo de calculo de CRC (1/2)

Sea m(x) = 1100001 y c(x) = x^3+x+1, con n-k = 3, que en binario c(x)=1011

Procedimiento

1. m(x) x^3 = 1100001**000**
2. **Resto**= (m(x) x^3)/c(x) = 101
3. (m(x) x^3)-Resto= 1100001**101**
4. Comprobación: calcular **resto'** de ((m(x) x^3)-Resto) **¿= 0?**

Operaciones

1100001000:1011
1011
01110
1011
01010
1011
0001100
1011
01110
1011
0**101 = Resto**

1100001101 :1011
1011
01110
1011
01010
1011
0001110
1011
01011
1011
0**000 = Resto'**

¿Probar si es divisible 1000001? *Ojo, que pasar a decimal y operar, no es válido con esta álgebra modular.*

Ejemplo de calculo de CRC (2/2)

- Supongamos: Código generador C(x)=x^3+x^2+1 ó *1101*
- Mensaje *m(x):* 10011010

Procedimiento: *k*=3->añadir 3 ceros a la derecha de *m(x)*

```
10011010000:1101=1111001
1101
 1001
 1101
  1000
  1101
   1011
   1101
    1100
    1101
       1000
       1101
        101  Resto
```

Mensaje a transmitir: 10011010**101**

Checksum

- La idea es sumar todas las palabras que se transmiten, añadiendo al final de la trama el resultado de esta suma --> o **checksum**.
- Esta suma se realiza en complemento a UNO

Sumario

- Funciones de la capa de enlace
- Control de errores
- **Control de flujo**
- Protocolos de la capa de enlace
- Protocolos de nivel de enlace: SDLC/HDLC y PPP
- Ejemplos de tecnologías de capa 1 y 2 (protocolos WAN): X.25, Frame-Relay, ATM

Control de flujo

- Necesario para no 'agobiar' al receptor y se realiza principalmente en la capa de enlace y de transporte.
- Utiliza diferentes mecanismos de retroalimentación para mandar señales de control de flujo, y por tanto requiere un canal semi-duplex o full-duplex. Estas señales pueden ser activación de líneas hardware (RTS, CTS), caracteres especiales (Xon, Xoff) o tramas especiales (*tramas que se llaman de reconocimiento o **ack**nowledgment (ack)*) de reconocimiento, para notificar la recepción correcta.
- El envio de ***acks*** permite controlar al transmisor, de forma que si no se le reconocen las tramas enviadas, éste espera hasta que se le reconozcan.
- Los ***acks*** se envían en ocasiones aprovechando la transmisión de datos en sentido contrario. Esta técnica es llamada *piggybacked* o llevar a espaldas.
- El control de flujo no debe limitar la eficiencia del canal.

Control de flujo

- Básico entre **dos equipos directamente conectados** **(PC <-->MODEM)**
  - Control de flujo software (**XON/XOFF**)
  - Control de flujo hardware (**RTS/CTS**)
- Bajo **protocolo "en redes de datos", o llamados protocolos de enlace de datos**
  - Procedimiento de repetición automática y con acuse de recibo, utilizando un canal de comunicaciones
    - Control de flujo con **parada y espera**
    - Control de flujo por **VENTANA DESLIZANTE**

Protocolo XON/XOFF

- Este control de flujo consiste en mandar los caracteres ASCII XON y XOFF por el mismo canal de comunicaciones. **Por ejemplo**, es utilizado en conexiones serie RS-232 con 3 hilos (*Tx, Rx y GND*)
- **El protocolo consiste en**: cuando el receptor está a punto de congestionarse, manda un carácter XOFF al emisor y éste se detiene. En el momento que el receptor se descongestiona, indica al emisor que puede reanudar, mandado el carácter XON.

Códigos ASCII: XON (0x11), XOFF (0x13)

Sumario

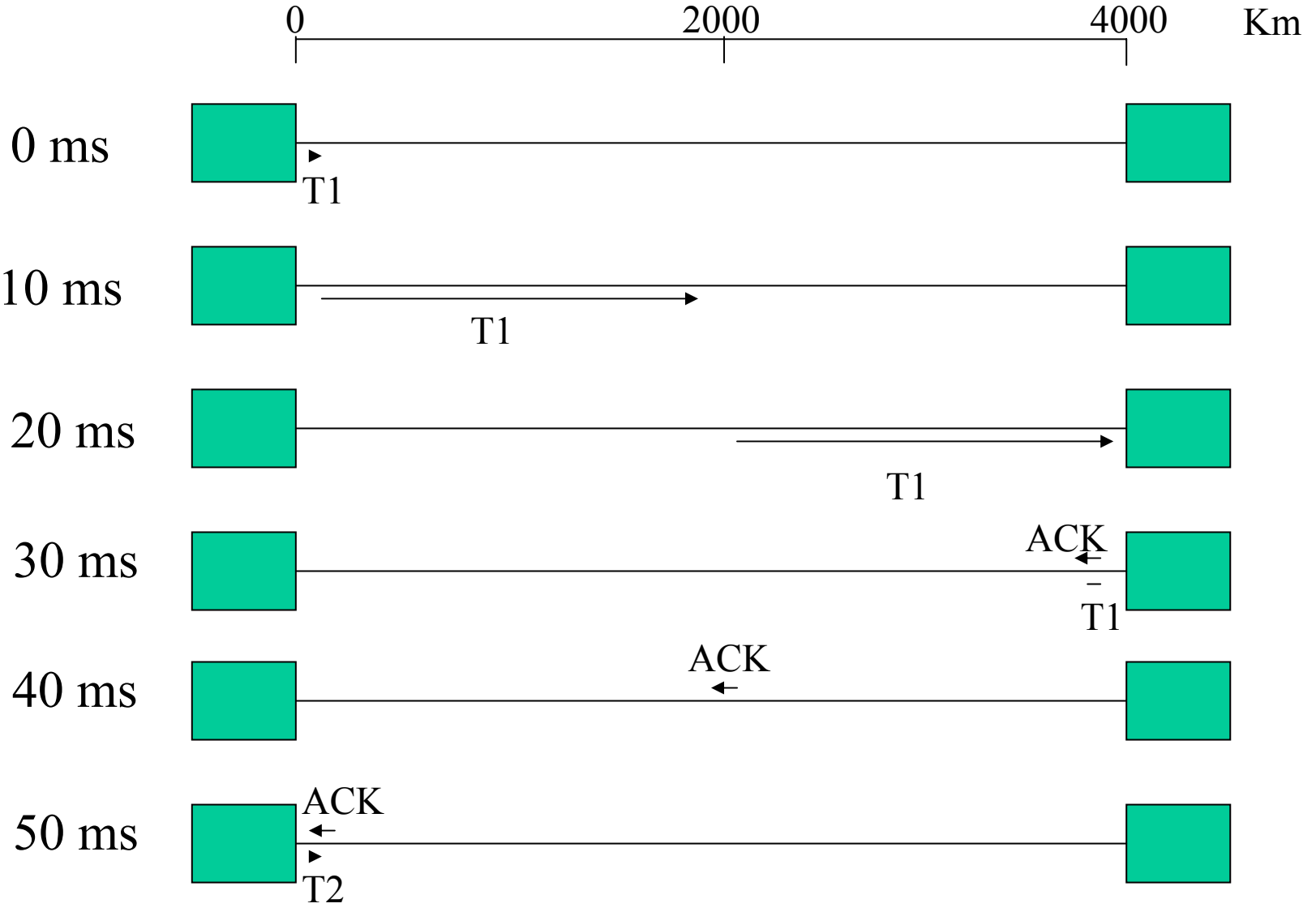
- Funciones de la capa de enlace
- Control de errores
- Control de flujo
- **Protocolos de la capa de enlace**
- Protocolos de nivel de enlace: SDLC/HDLC y PPP
- Ejemplos de tecnologías de capa 1 y 2 (protocolos WAN): X.25, Frame-Relay, ATM

Protocolo de parada y espera (1/3)

- Es el protocolo fiable orientado a conexión más sencillo, también conocido en inglés como “Stop&wait”
- Impide un uso eficiente de los enlaces, p. ej. línea punto a punto de A a B de 64 Kb/s de 4000 Km (***lo que supone un retraso de 20ms, pues por la propagación que cada km introduce 5us***), tramas de 640 bits(***640bits/64Kbps=10ms***):
  - 0 ms: A empieza el envío de trama T1
  - 10 ms: A termina envío de T1 y espera
  - 20 ms: B empieza recepción de T1
  - 30 ms: B termina recepción de T1; envía ACK de T1 por canal duplex
  - 50 ms: A recibe ACK de T1; empieza envío de T2

  Eficiencia: tiempo útil/ tiempo total=10 ms/50 ms = 0,2 = 20%

Protocolo de parada y espera (2/3)



Protocolo de parada y espera (3/3)

Time-out: en el caso de pérdida de una trama o de su ACK, el emisor no sabe si la trama ha llegado correctamente o no. Por tanto, por cada trama enviada lanza un contador (*time-out*), de forma que si vence antes de recibir alguna notificación, vuelve a retransmitir la trama correspondiente. Este tiempo se suele aproximar al tiempo de "*ida y vuelta*" de una trama. Además, en ocasiones según los retrasos (o la congestión de la red) este tiempo se puede ajustar dinámicamente.

Numeración de las tramas: es imporntante una numeración de tramas para evitar recibir tramas duplicadas en el caso de realizarse retransmisiones

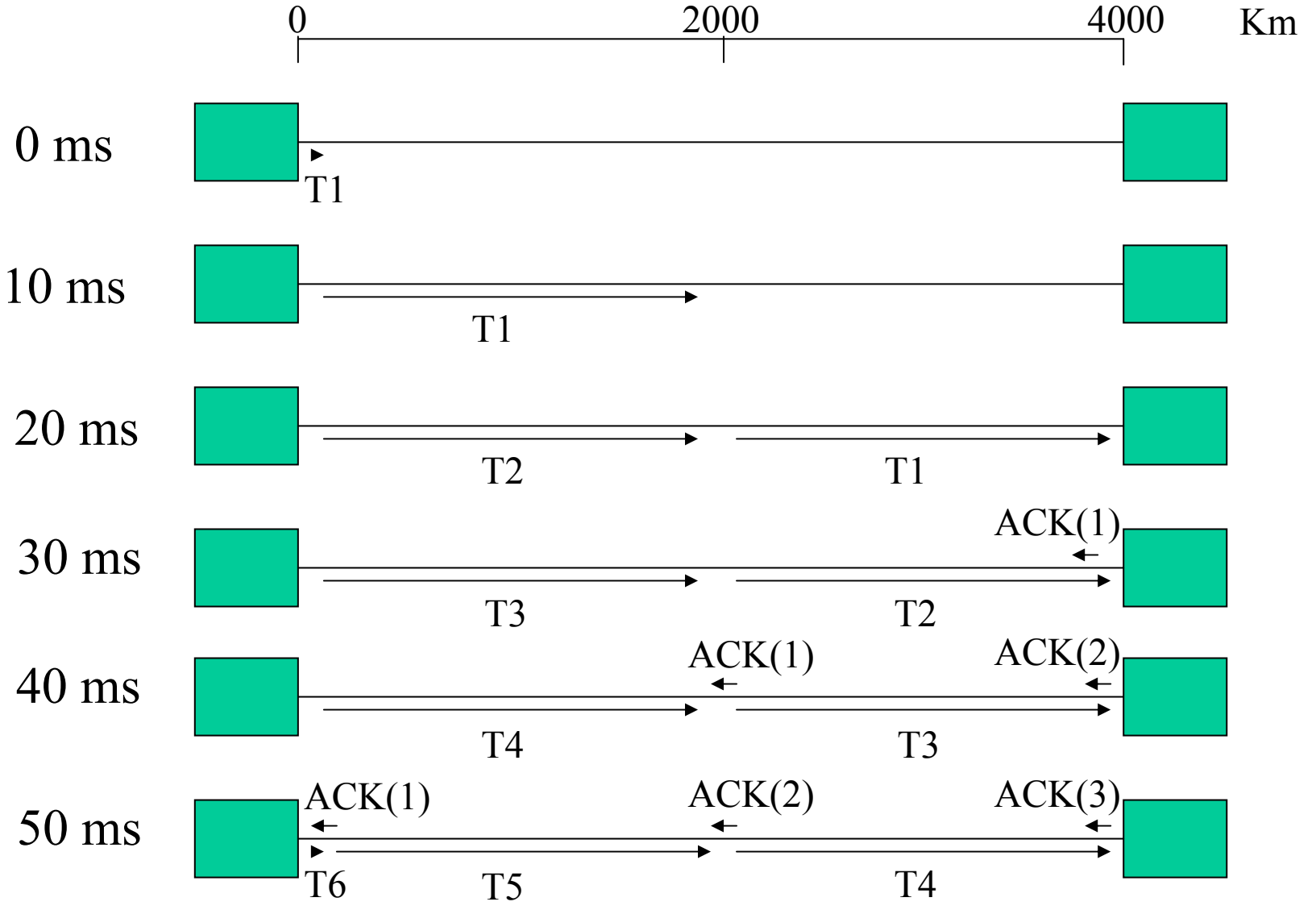
Protocolo de ventana deslizante (1/3)

- Implementa un pipeline (segmentación o tubería) para evitar los tiempos muertos en la línea:
  - 0 ms: A envía T1
  - 10 ms: A envía T2;
  - 20 ms: A envía T3; B empieza a recibir T1
  - 30 ms: A envía T4; B envía ACK(T1)
  - 40 ms: A envía T5
  - 50 ms: A recibe ACK(T1) y envía T6

Ventana mínima para 100% de ocupación: 5, es decir, el máximo número de tramas que pueden estar viajando simultáneamente en el canal.

- Resuelve problema de eficiencia a cambio de mayor complejidad y espacio en buffers

Protocolo de ventana deslizante (2/3)



Protocolo de ventana deslizante (3/3)

- La ventana mínima para 100% de ocupación es la que 'llena el hilo' de datos en ambos sentidos mas uno, evitando que hayan 2 tramas válidas simultáneamente con la misma numeración:
- **$W = 2\tau * v/t + 1$**
  - W: tamaño de ventana
  - τ: tiempo de propagación (segundos)
  - v: velocidad de la línea (bps)
  - t: tamaño de trama (bits/trama)

  Por tanto v/t, serán tramas por segundo y por 2τ serán las tramas en linea en ese tiempo
  - (+1 para redondear al entero superior)

  Nota: 2τ es el tiempo de "ida y vuelta"

  Del ejemplo anterior: τ=20ms, v = 64 Kbps, t = 640 bits $\rightarrow$W = 5

Protocolos de ventana deslizante

- El protocolo puede ser:
  - **Retroceso n:** no se acepta una trama hasta haber recibido las anteriores. Tamaño de ventana receptor 1, obligando la recepción de trama en trama de forma consecutiva, y emisor >1.
  - **Repetición selectiva**: se admite cualquier trama en el rango esperado y se pide solo la que falta. Tamaños >1
- *Repetición selectiva es más complejo pero más eficiente, y requiere mas espacio en buffers en el receptor.*
- **Tamaño de ventana:**
  - Retroceso n: *Número de secuencia – 1*
  - Repetición selectiva: *Número de secuencia/2*

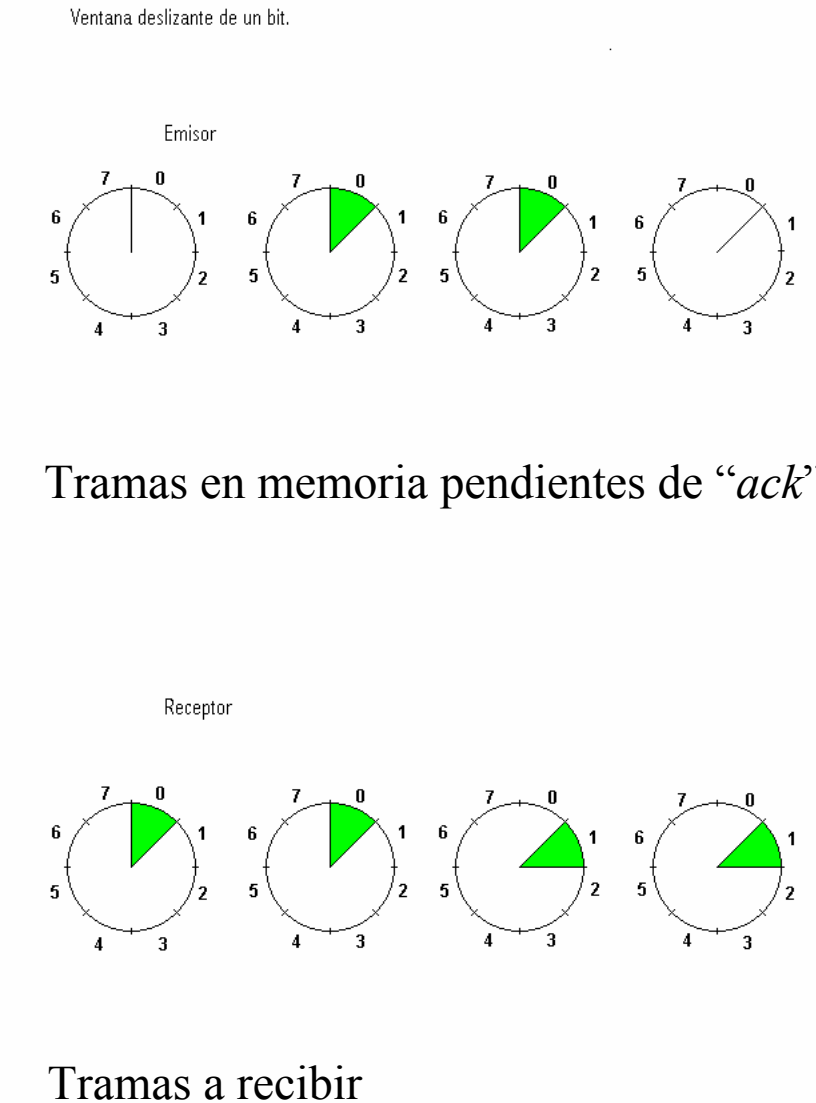
Nota: en los reconocimientos o "ack" se indica la última trama que ha llegado bien, reconociendo todas las anteriores como correctas.

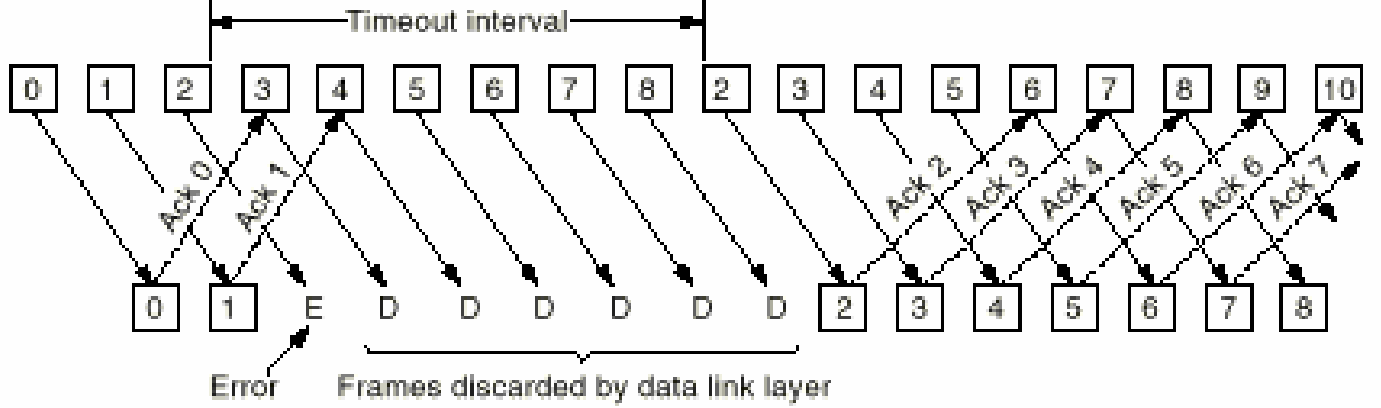
Protocolo de ventana deslizante de un bit

La esencia de los protocolos de ventana deslizante es que en cualquier instante de tiempo, el emisor mantiene una lista de números consecutivos de secuencia, correspondientes a las tramas que puede enviar. En el ejemplo de la derecha, dicha lista es sólo de un número de secuencia, mientras el receptor mantiene la ventana para un número de secuencia para recibir.

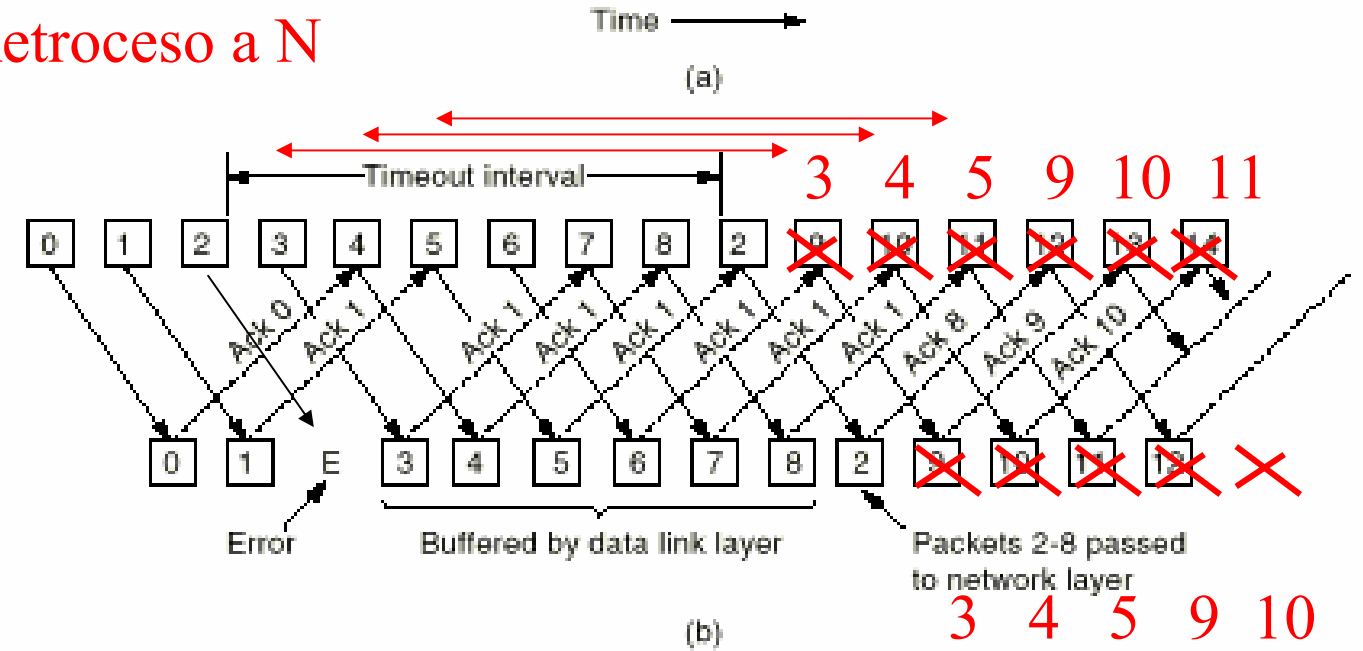
En este caso, las ventanas son de tamaño 1 tanto en emisión como recepción.

Este proceso es equivalente al protocolo de parada y espera.





Retroceso a N



Repetición selectiva

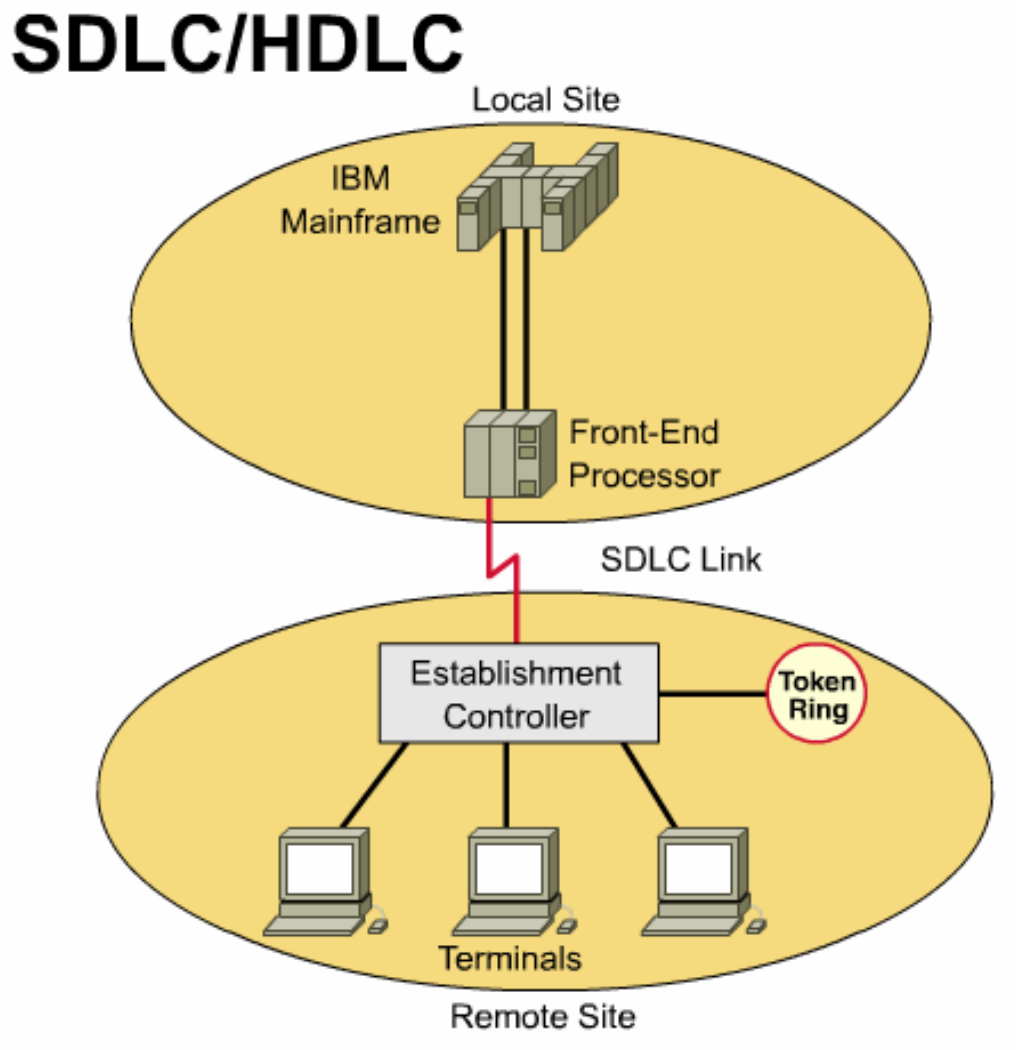
Fig. 3-15. (a) Effect of an error when the receiver window size is 1. (b) Effect of an error when the receiver window size is large.

Comentario sobre figura anterior

- “Nack”: También se podría contemplar la inclusión de un *reconocimiento negativo*, non-ack de la trama 2 y por tanto, en la trama recibida 3 se indicaría “nack-2”, reduciendo el tiempo de espera del “timeout”.
- Las implementaciones que pueden dar pie estos protocolos son muy variadas y contemplan todas las situaciones posibles, como veremos en los siguientes ejemplos.

Sumario

- Funciones de la capa de enlace
- Control de errores
- Control de flujo
- Protocolos de la capa de enlace
- **Protocolos de nivel de enlace: SDLC/HDLC y PPP**
- Ejemplos de tecnologías de capa 1 y 2 (protocolos WAN): X.25, Frame-Relay, ATM

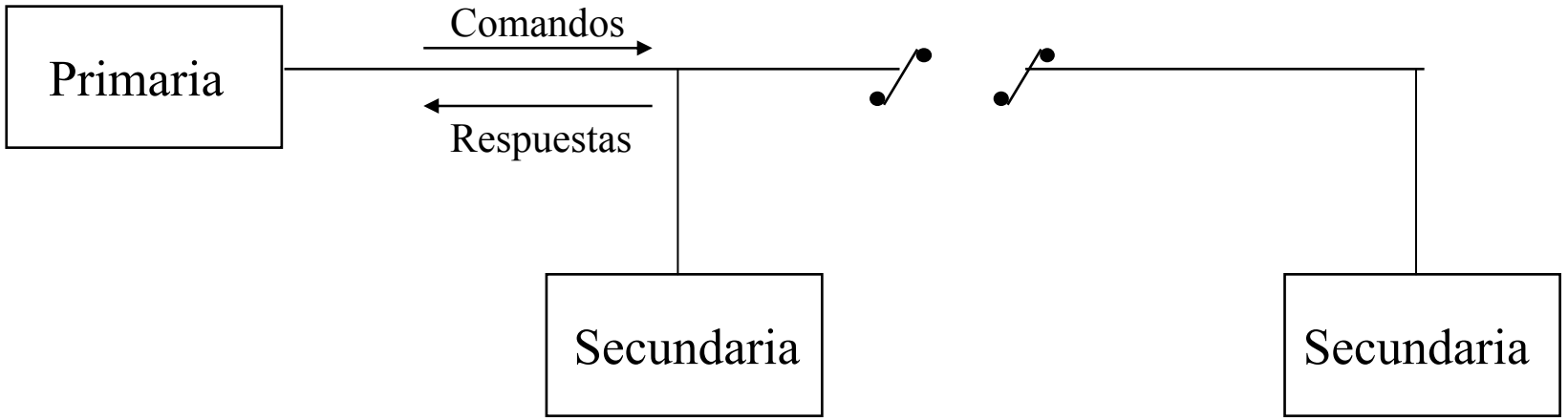


Ej: Arquitectura SNA de IBM con conexiones por protocolo SDLC

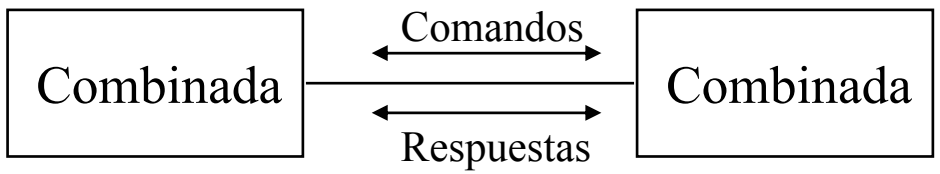
SDLC/HDLC

- HDLC (High level Data Link Control) es un "**estándar" ISO** y deriva del SDLC (Synchronous Data Link Control) desarrollado por IBM en 1972 para su arquitectura **SNA**.
- HDLC como evolución de SDLC, se considera el protocolo que ha incluido los aspectos recogidos por SDLC y otras funcionalidades. Tanto SDLC como HDLC son **protocolos de ventana deslizante** muy completos.
- SDLC/HDLC es un protocolo inicialmente pensado para conexiones remotas a un supercomputador en modo **bien punto a punto o bien multipunto**.
- En las conexiones punto a punto, son llamadas "**balanceadas**", una comunicación de igual a igual.
- En las conexiones multipunto, son llamadas "**no balanceadas**", los elementos que participan en SDLC/HDLC son **un nodo llamado primario y varios secundarios**. El nodo primario controla a los secundarios por "polling" o monitorización. Los secundarios, sólo responden a los primarios bajo petición.

SDLC/HDLC: modos



a) Modo no balanceado

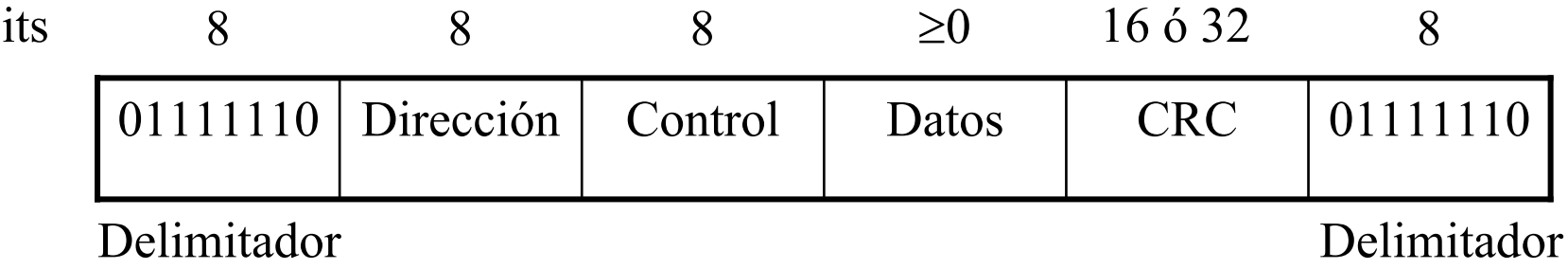


b) Modo balanceado

Familia de protocolos SDLC/HDLC

- Prácticamente todos los protocolos de enlace actuales son subconjuntos de HDLC:
  - PPP (Point-to-Point Protocol): Internet
  - LAP-B (Link Access Procedure Balanced): X.25
  - LAP-F: Frame Relay
  - LLC (IEEE 802.2): redes locales
  - LAPM: módems RTC

Formato de trama HDLC



- Se utiliza relleno de bits (*bit stuffing*)
- El campo dirección siempre vale 11111111 (dirección broadcast) en las conexiones punto a punto (modo balanceado)
- El campo control es el que realiza todas las tareas propias del protocolo de ventana deslizante.
- El CRC es normalmente de 16 bits, pero puede ser de 32

Nota: En las implementaciones de los diferentes fabricantes, el campo de control ha sufrido modificaciones y actualmente el protocolo HDLC de diferentes fabricantes, puede no ser compatible.

Tipos de tramas HDLC

Las tramas HDLC pueden ser de tres tipos según el valor de los primeros bits del campo control:

De información (tramas tipo **I**) *Information*

1	3	1	3
0	Tx. SEQ	P/F	NEXT

De supervisión o control (tramas tipo **S**) *Supervisory*

2		2	1	3
1	0	ORDEN	P/F	NEXT

No numeradas (tramas tipo **U**), *Unnumbered: para inicializar secundarios*

2		2	1	3
1	1	ORDEN 1/2	P/F	ORDEN 2/2

Descripción: **Tx. SEQ**: número de secuencia en transmisión, **NEXT**: indica la trama pendiente de recibir, reconociendo todas las anteriores, **ORDEN:** para codificar órdenes, **P/F** (Polling/Final, sólo utilizado en líneas multipunto): es activado por el modo primario para invocar al secundario a dar respuesta. El secundario lo activa para indicar el final.

Comandos en tramas de supervisión HDLC

Orden	Comando	Significado
00	RECEIVE READY	ACK cuando no hay tráfico de vuelta para piggybacking
10	RECEIVE NOT READY	Recepción correcta pero pide suspender transmisión (control de flujo)
01	REJECT	Acuse de recibo negativo (NAK). Pide reenvío cuando se usa retroceso n
11	SELECTIVE REJECT	Petición de reenvío cuando se usa repetición selectiva

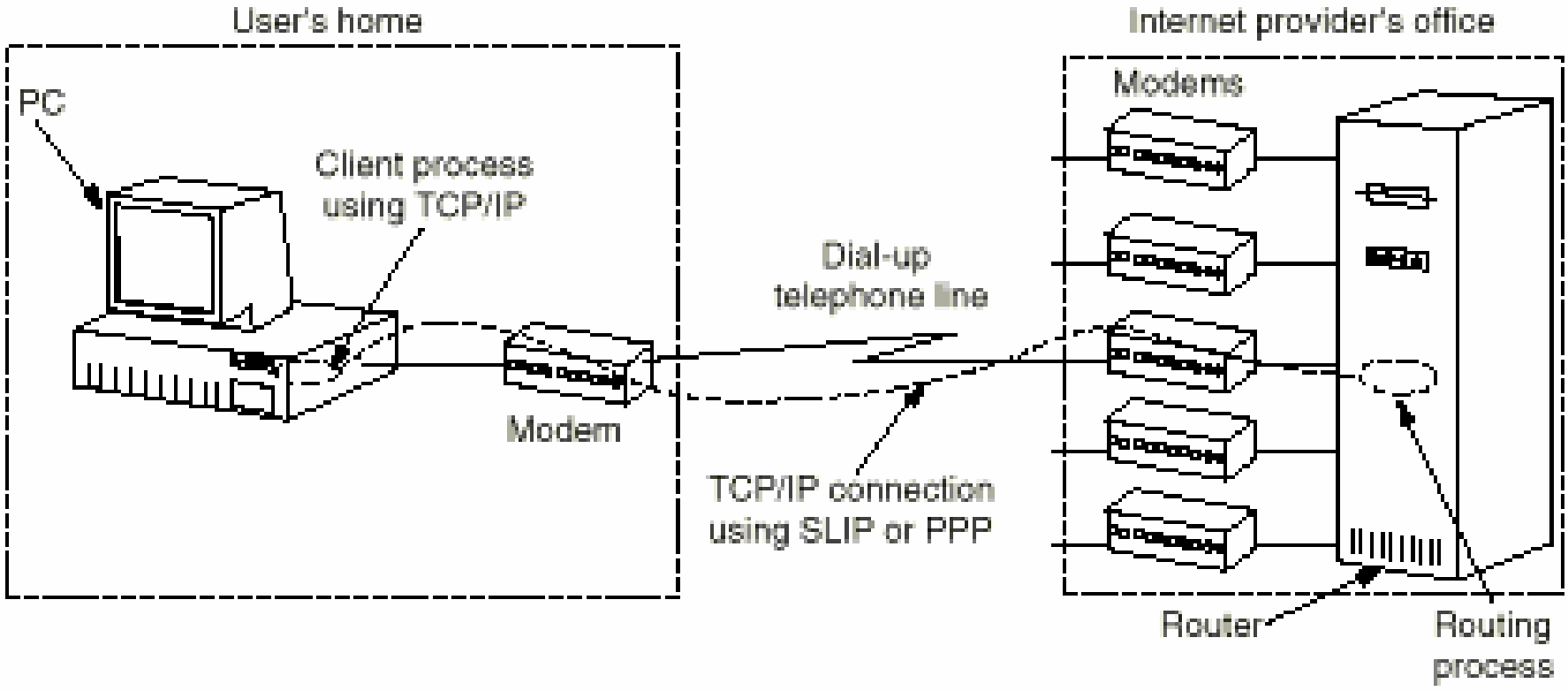
Elaboración de tramas HDLC

- En el emisor:
  1. Concatenar campos dirección, control y datos
  2. Calcular el CRC de la cadena resultante
  3. Realizar el relleno de bits poniendo un bit a cero siempre que en la cadena a enviar aparezcan cinco 1s seguidos
  4. Añadir a la trama los delimitadores de inicio y final (01111110). Si se envían dos tramas seguidas el delimitador de final de una sirve como inicio de la siguiente
- El receptor procede de manera inversa (4,3,2,1)

Nivel de enlace en Internet y PPP

- El protocolo IP está diseñado para funcionar sobre casi cualquier medio físico ('IP over everything'):

Medio	**RFC**	**Año**
X.25	877, 1356	1983
Ethernet	894	1984
802.x	1042	1988
FDDI	1188, 1390	1990
PPP	**1171, 1663**	**1990**
Frame Relay	1490	1993
ATM	1483, 1577	1994



NAS

Fig. 3-26. A home personal computer acting as an Internet host.

SLIP (serial line internet protocol):

Protocolo antiguo utilizado para conectar dos estaciones de trabajo a través de Internet vía modem

Sus *características principales*:

- sólo envía paquetes IP
- usa delimitación de trama
- usa relleno de bits (bit-stuffing)
- actualmente efectúa compresión de cabeceras TCP e IP

Problemas que plantea:

- no detecta ni corrige errores
- sólo trabaja con IP
- no soporta direccionamiento IP dinámico
- no tiene mecanismos de autenticidad (no sabemos con quién hablamos)
- no está normalizado

PPP (Point to Point Protocol)

- El protocolo de enlace'característico' de Internet es el PPP, que se utiliza para transportar datos en la capa de enlace sobre:
  - Líneas dedicadas punto a punto
  - Conexiones RTC analógicas o digitales (RDSI o en inglés ISDN)
  - Conexiones de alta velocidad sobre enlaces SONET/SDH
- Es *multiprotocolo*, una comunicación soporta simultáneamente varios protocolos del nivel de red.
- PPP consta de varios protocolos, definiendo una arquitectura

Funcionamiento de PPP

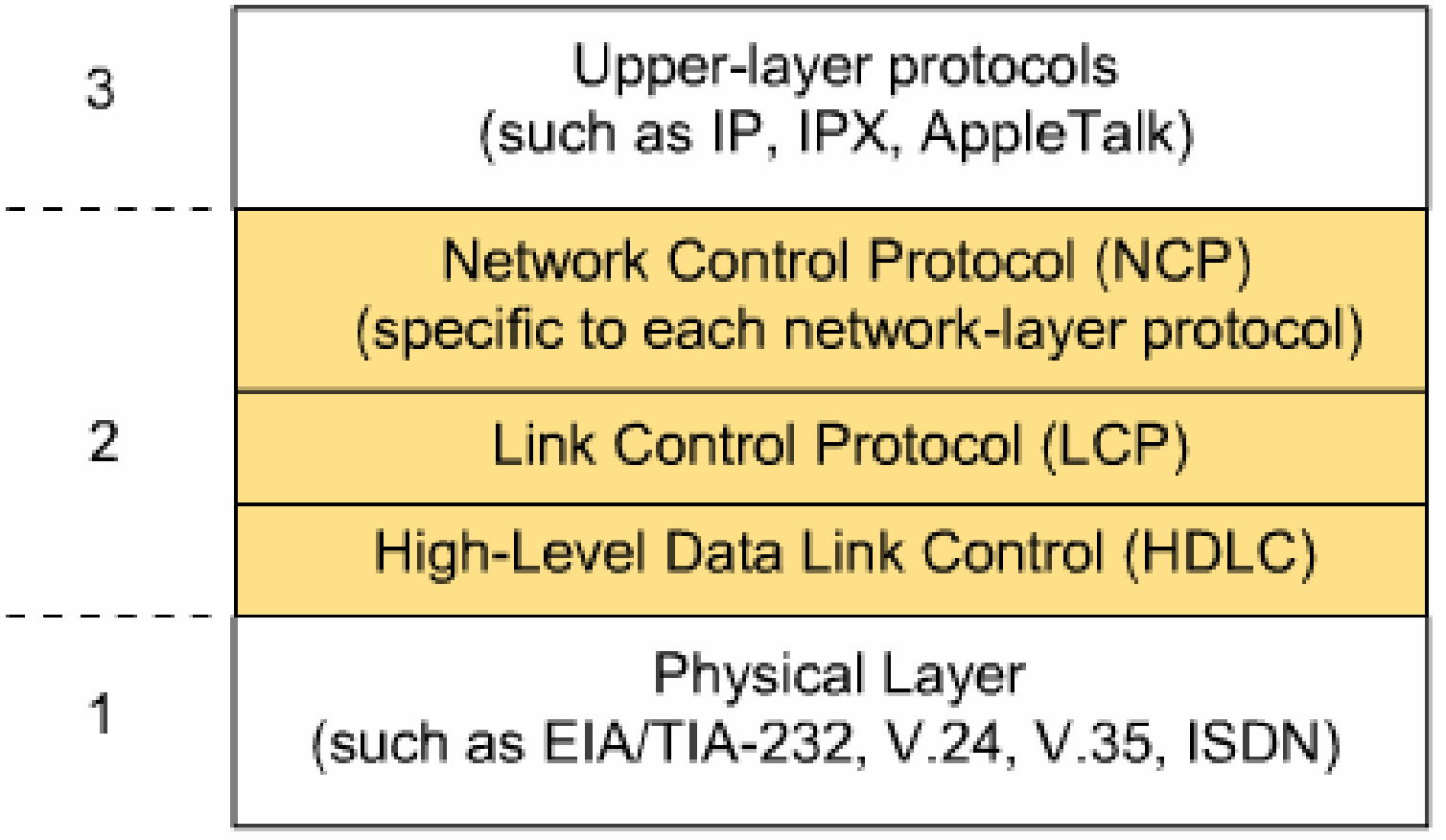
- Utiliza estructura de tramas tipo HDLC:

its

8	8	8	8 o 16	Variable	16 o 32	8
Delimitad. 01111110	Dirección 11111111	Control	Protocolo	Datos p.ej IP	CRC	Delimitad. 01111110

- La trama siempre tiene un número entero de bytes
- El campo dirección no se utiliza, siempre vale 11111111
- Generalmente en el inicio se negocia omitir los campos dirección y control (compresión de cabeceras)

Arquitectura PPP



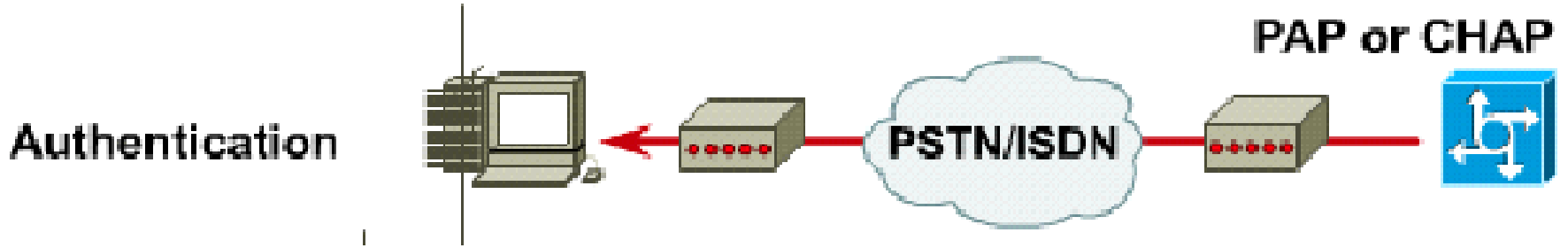
Componentes de PPP

- *LCP (Link Control Protocol):* negocia parámetros del nivel de enlace en el inicio de la conexión, por ejemplo.:
  - Establece y configura el enlace
  - Controla la calidad de la línea
  - Supresión de campos dirección y control, se ponen de acuerdo con el formato de trama HDLC
  - Uso de protocolos fiables (con ACK)
  - Negocia tamaño máximo de trama
  - Opciones configurables: métodos de autentificación de la conexión entrante por temas de seguridad, compresión de cabeceras, gestión de múltiples enlaces, llamadas revertidas
- *NCP (Network Control Protocol):* negocia parámetros del nivel de red:
  - Protocolos soportados: IP, IPX y AppleTalk
  - Asignación dinámica de dirección IP en el caso de IP

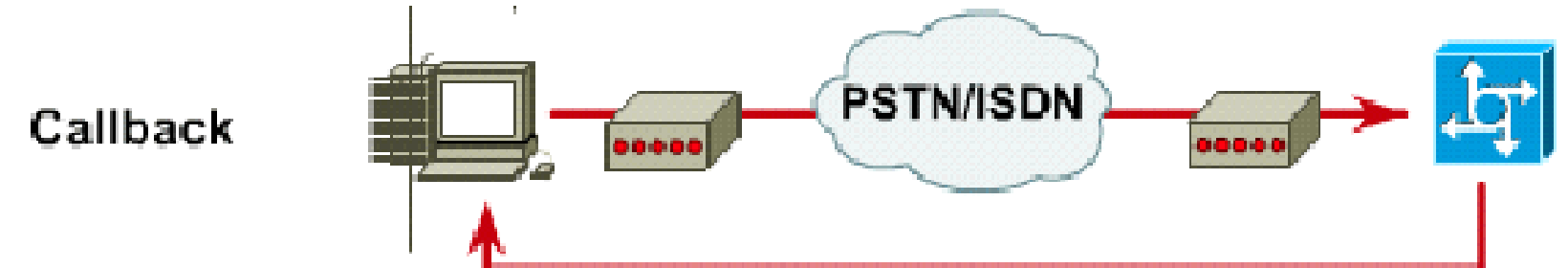
PPP: opciones en LCP (1/2)

LCP negocia y establece el enlace antes de invocar al apropiado NCP, permitiendo configurar las siguientes opciones

A) Authentication, con PAP o CHAP



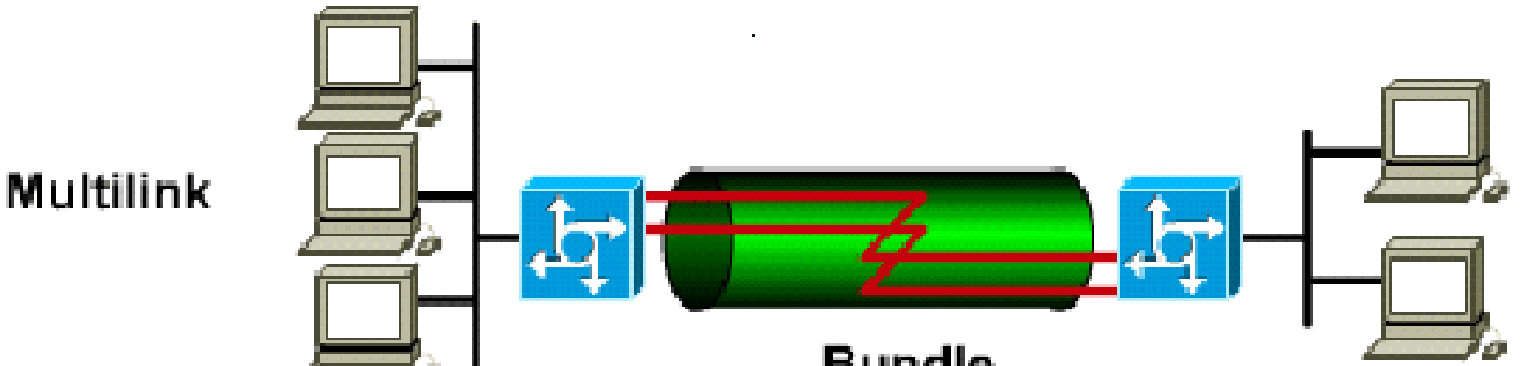
B) Callback o llamada revertida utilizada por temas de facturación y/o costes



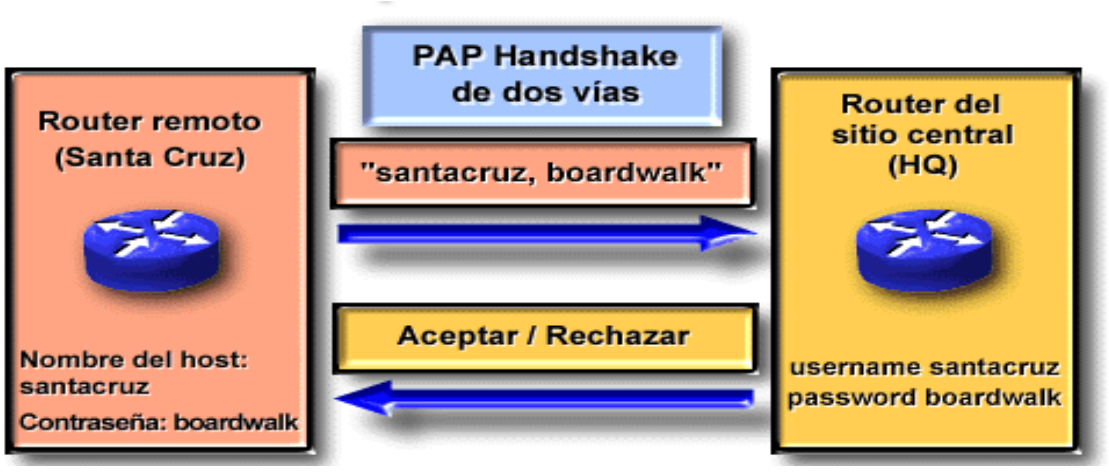
PPP: opciones en LCP (2/2)

C) Compression utilizado para mejorar el throughput o capacidad de un enlace utilizando diferentes técnicas de compresión en los datos.

D) Multilink PPP (MLP) para gestionar simultáneamente diferentes canales o circuitos, por ejemplo en el caso de RDSI con 2 canales B de un BRI



PAP: Password Authentication Protocol



PAP consta de las siguientes fases:

1º Se envía "*Login/Password*" repetidamente hasta reconocerse en texto claro (No existe protección contra ataques repetidos de prueba y error)

2º Si "*Login/Password*" coincide con los registrados localmente en el Router del Sitio Central, se acepta la conexión.

➔ **No constituye un método fuerte o rígido.**

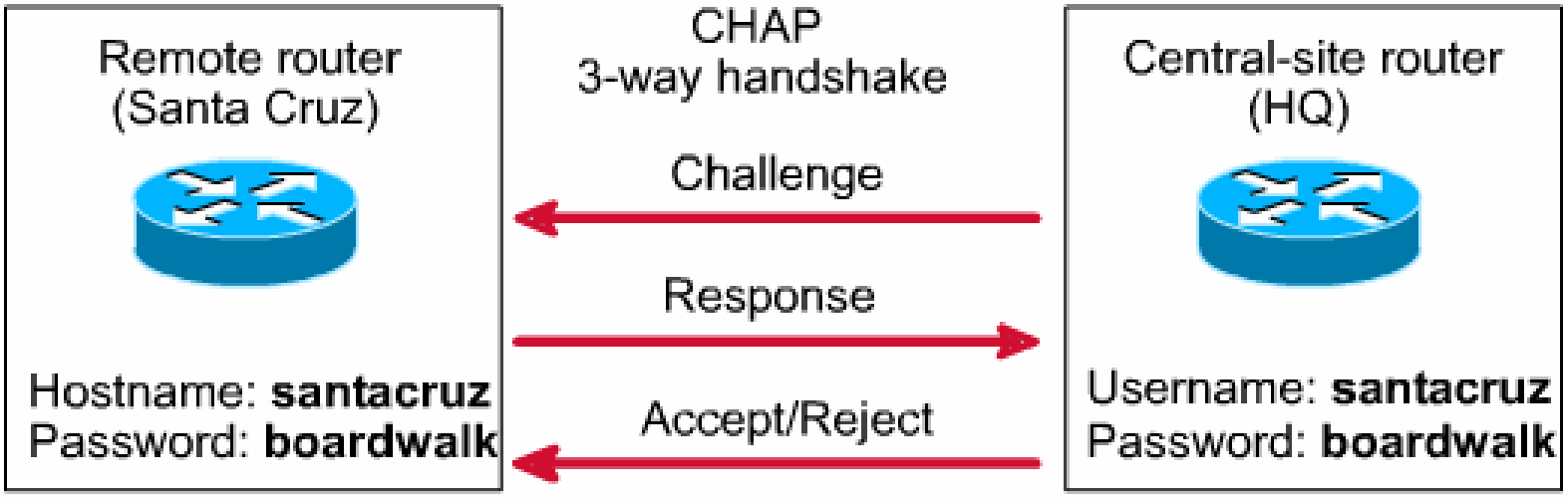
Ejemplo de configuración de PAP en PPP

```
hostname RTA
username RTB password juliet
interface async 0
   encapsulation ppp
   ppp authentication pap
   ip address 10.0.0.1 255.255.255.0
   dialer map ip 10.0.0.2 name RTB 5551234
   ppp pap sent-username RTA password romeo
```


```
hostname RTB
username RTA password romeo
interface async 0
   encapsulation ppp
   ppp authentication pap
   ip address 10.0.0.2 255.255.255.0
   dialer map ip 10.0.0.1 name RTA 5554321
   ppp pap sent-username RTB password juliet
```

Nota: configuración sobre equipos de Cisco Systems

CHAP: Challenge Handshake Authentication Protoco



CHAP consta de las siguientes fases:

1º Central Site Router:envía mensaje "challenge" o reto a Remote Router, basado en un número aleatorio.

2º Remote Rotuer: responde con un valor calculado con su password utilizando una función matemática hash sobre el número aleatorio.

3º Central Site Router:verifica la respuesta y compara con valor esperado.

Comentarios del CHAP

- CHAP no permite al que realiza la llamada, intentar la autenticación sin un desafío o reto (challenge) previo.
- Los login y contraseña no viajan por la red.
- El valor de “challenge” es único e impredecible, por lo que se proporciona protección frente a ataques.
- Ofrece características como la verificación periódica para mejorar la seguridad (en PAP sólo se verifica una vez)

Ejemplo de configuración de CHAP en PPP


```
hostname left
username right password
      someone
int async 0
encapsulation ppp
ppp authentication CHAP
```

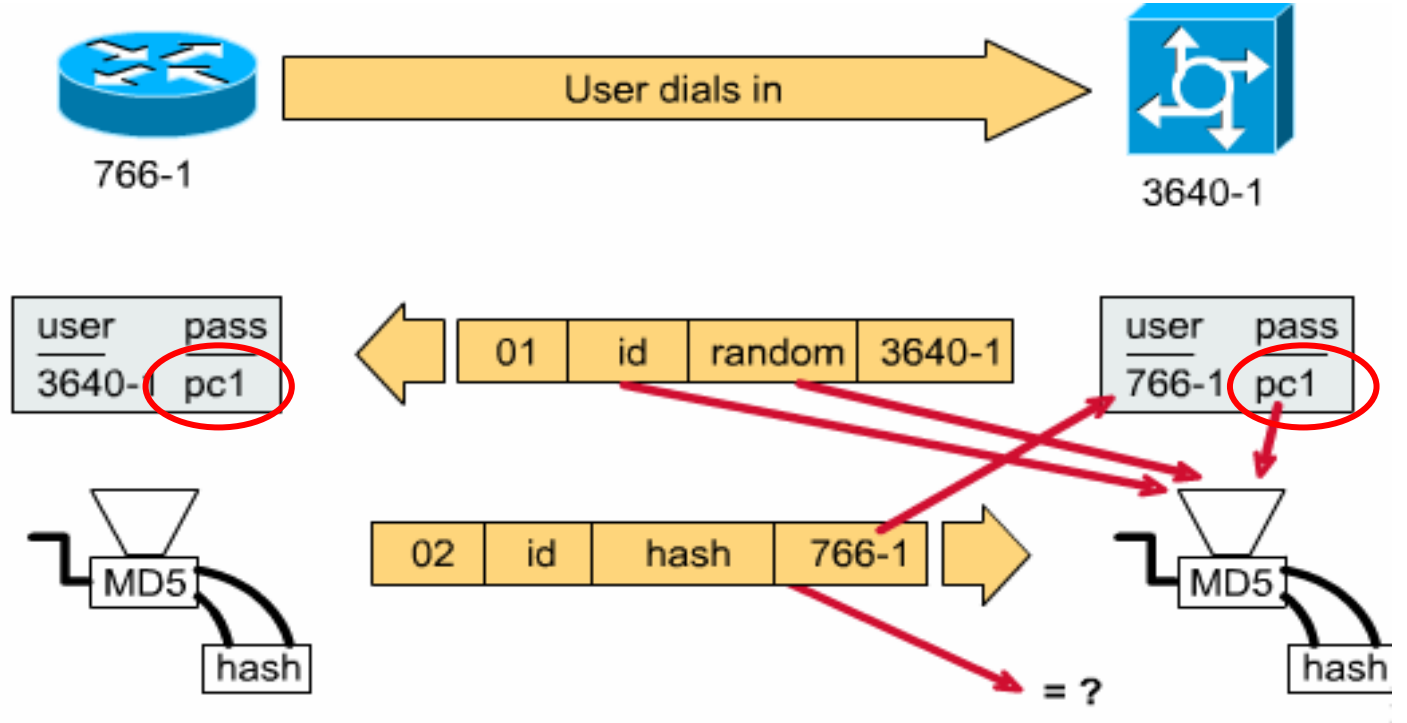
```
hostname right
username left password
      someone
int async 0
encapsulation ppp
ppp authentication CHAP
```

Los passwords son sensibles a minúsculas/mayúsculas y deben ser identicos

Nota: configuración sobre equipos de Cisco Systems

Password y CHAP

Los *passwords* o contraseñas no viajan por la red en CHAP.



La función *hash* se realiza con el método MD5 (Message Digest –5). Son funciones que generan un valor unívoco dado un texto determinado pasado como argumento "*id; random; password*".

Sumario

- Funciones de la capa de enlace
- Control de errores
- Control de flujo
- Protocolos de la capa de enlace
- Protocolos de nivel de enlace: SDLC/HDLC y PPP
- **Ejemplos de tecnologías de capa 1 y 2 (protcolos WAN): X.25, Frame-Relay, ATM**

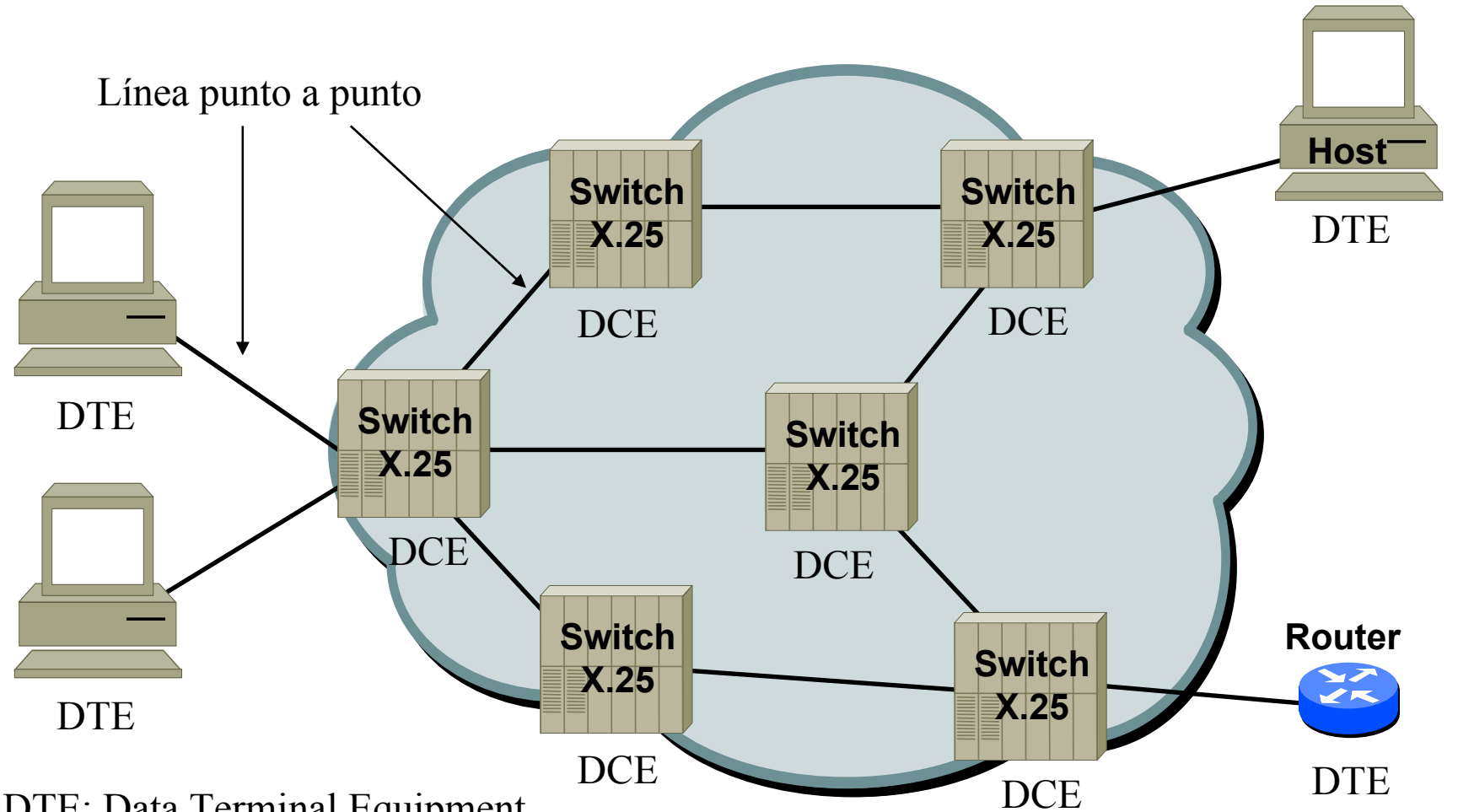
Conmutación de paquetes con circuitos virtuales

- Redes de conmutación de paquetes orientadas a conexión:
  - **X.25**: primer estándar de red pública de conmutación de circuitos. En España desde 1984 (red *Iberpac* de Telefónica). Hoy en día poco interesante.
  - **Frame Relay** (conmutación de tramas): versión aligerada de X.25. En España desde 1992 (red *Uno* de Telefónica)
  - **ATM** (conmutación de celdas): en España desde 1997(red *Cinco* y servicio *Gigacom* de Telefónica)
- Posibilidad de crear circuitos virtuales de dos tipos:
  - Temporales: SVCs (Switched Virtual Circuits). Se crean y destruyen dinámicamente cuando se necesitan.
  - Permanentes: PVCs (Permanent Virtual Circuits). Se configuran manualmente en los equipos para que estén siempre activos
- Las redes públicas X.25 permiten SVCs y PVCs. Las redes públicas Frame Relay y ATM solo permiten PVCs

X.25

- Primer servicio estándar de red pública de datos. Especificado en 1976.
- Especifica los tres niveles inferiores (físico, enlace y red)
- Sistema jerárquico de direccionamiento X.121. Interconexión a nivel mundial.
- Diseñado para medios físicos poco fiables. Comprobación de datos a nivel de enlace (protocolo de ventana deslizante).
- No apto para tráfico en tiempo real
- Paquetes de hasta 128 bytes normalmente.
- Servicio orientado a conexión. Orden garantizado.
- Costo proporcional al tiempo (normalmente SVC) y al tráfico (número de paquetes).
- Velocidades típicas de 9,6 a 64 Kbps.
- Servicio poco interesante en la actualidad, ofrecido desde el 1984 como IberPac por Telefónica de España.

Red de conmutación de paquetes X.25

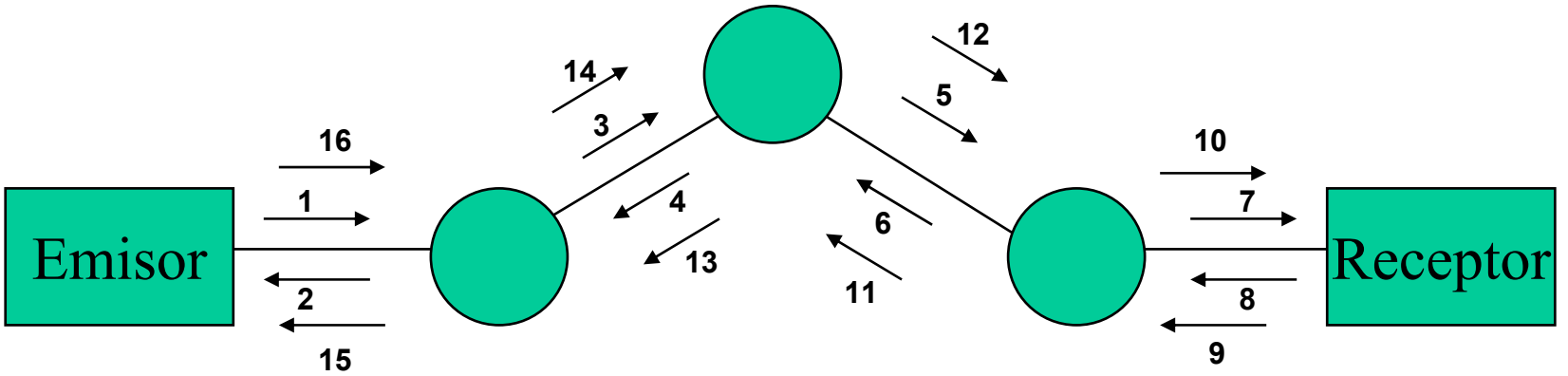


DTE: Data Terminal Equipment
DCE: Data Communications Equipment

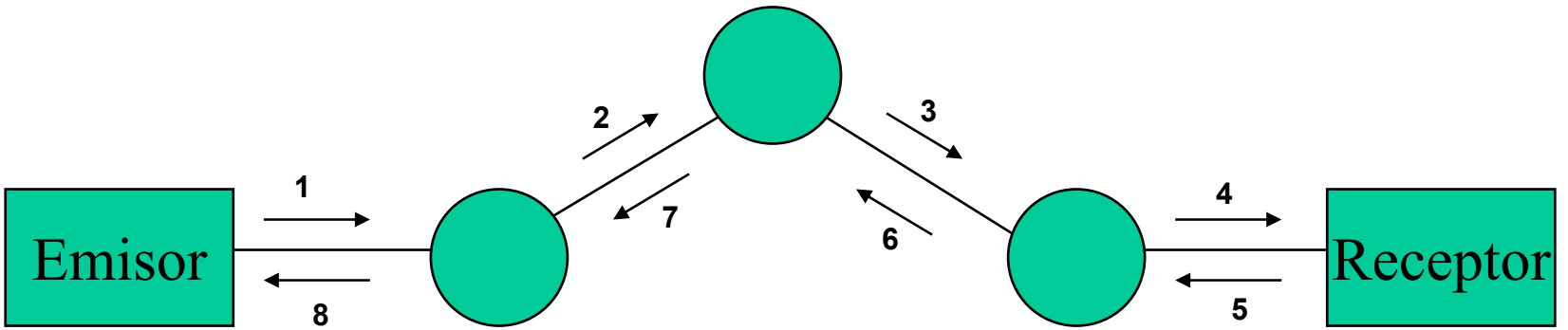
Frame Relay o Retransmisión de trama

- Versión aligerada del X.25 pensada sólo para transmitir datos.
- Pensada para combinar con otros protocolos como TCP/IP, y para interconexión multiprotocolo de LANs
- Servicio no fiable; si llega una trama errónea se descarta y el nivel superior (normalmente transporte) ya se enterará y pedirá retransmisión
- Tamaño máximo de paquete (trama) de 1 a 8 KB
- Velocidades de acceso hasta 44.736 Mb/s, típicas de 64 a 1.984 Kb/s
- QoS definida por CIR (Committed Information Rate) y por EIR (Excess Information Rate). Puede ser asimétrico.
- Eficiencia mucho mejor que X.25, especialmente a altas velocidades
- Habitualmente utiliza PVCs. SVCs no soportados por muchos operadores.
- Costo proporcional a capacidad de línea física y al CIR , no al EIR
- El servicio es ofrecido por Telefónica como RedUNO.

Comunicación sobre X.25 y Frame Relay



X.25



Frame Relay

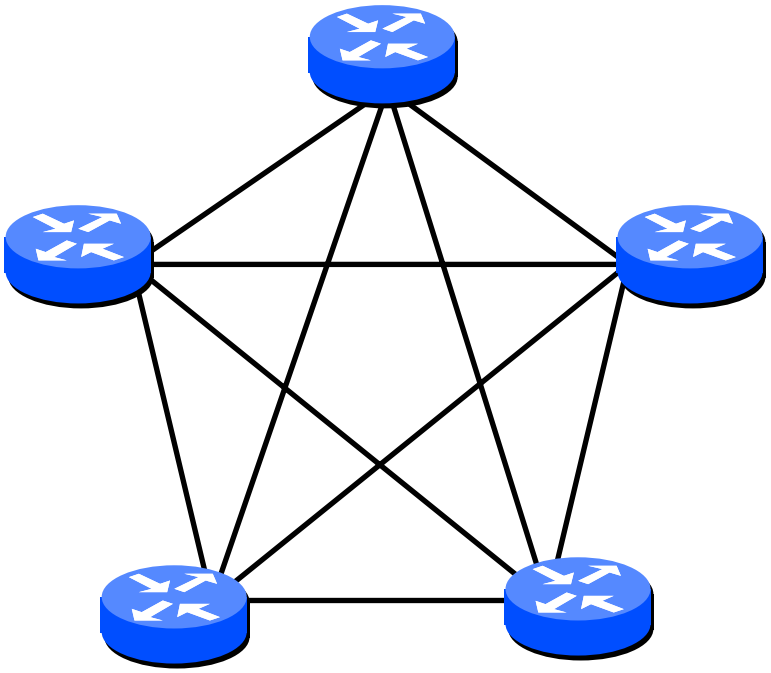
Nivel de enlace en Frame Relay

Estructura de trama, procedente de HDLC:

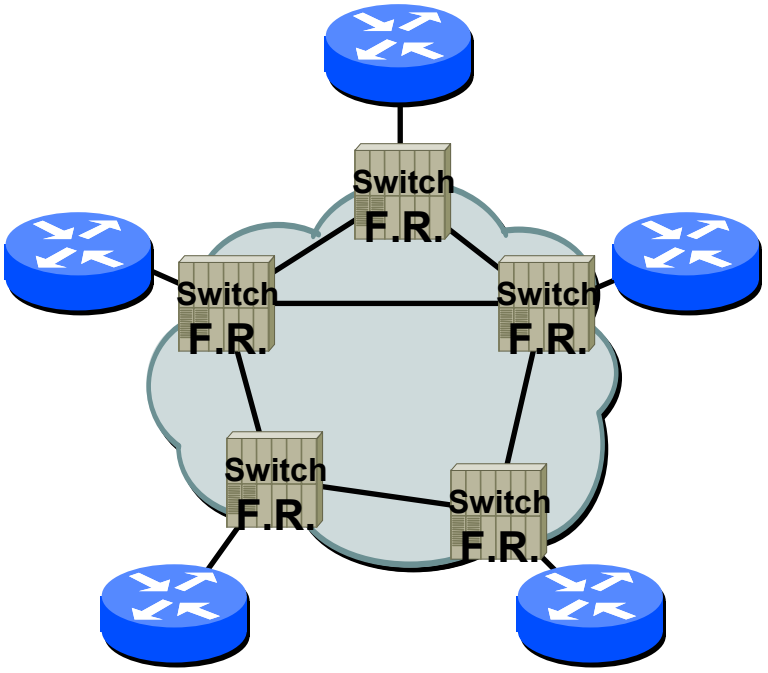
Bytes →	1	2	0-8188	2	1
	01111110	Dirección	Datos	CRC	01111110

- No se realiza reenvío en caso de error
- El campo dirección contiene la información del circuito virtual y los parámetros propios de las funciones de Frame Relay; su estudio corresponde al nivel de red.

Líneas dedicadas *vs* conmutación de paquetes (Frame Relay)

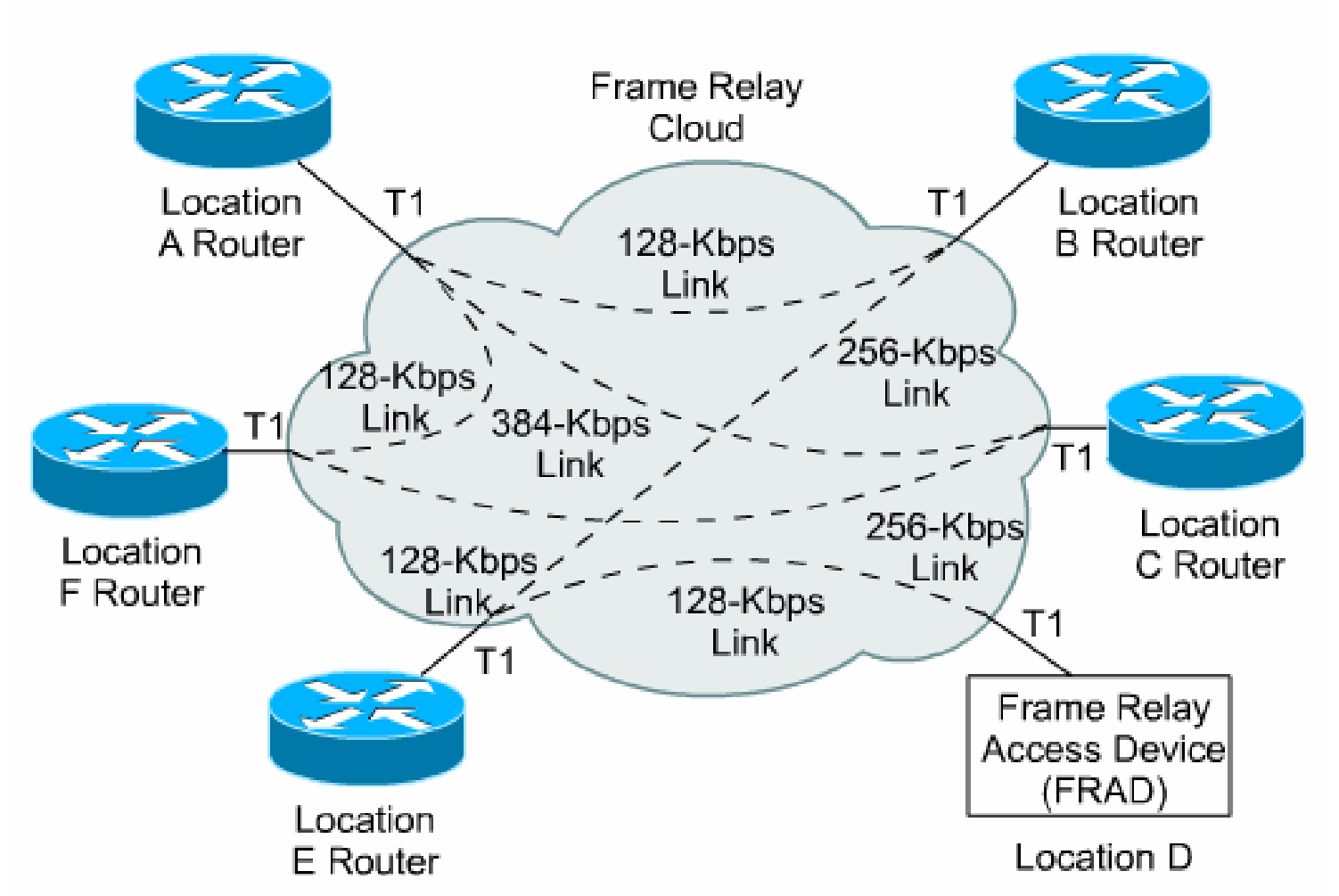


Mallado completo de una red con cinco nodos mediante enlaces punto a punto. Se establecen 10 enlaces.



Mallado completo de una red con cinco nodos mediante accesos Frame Relay. Se establecen cinco enlaces y 10 circuitos virtuales

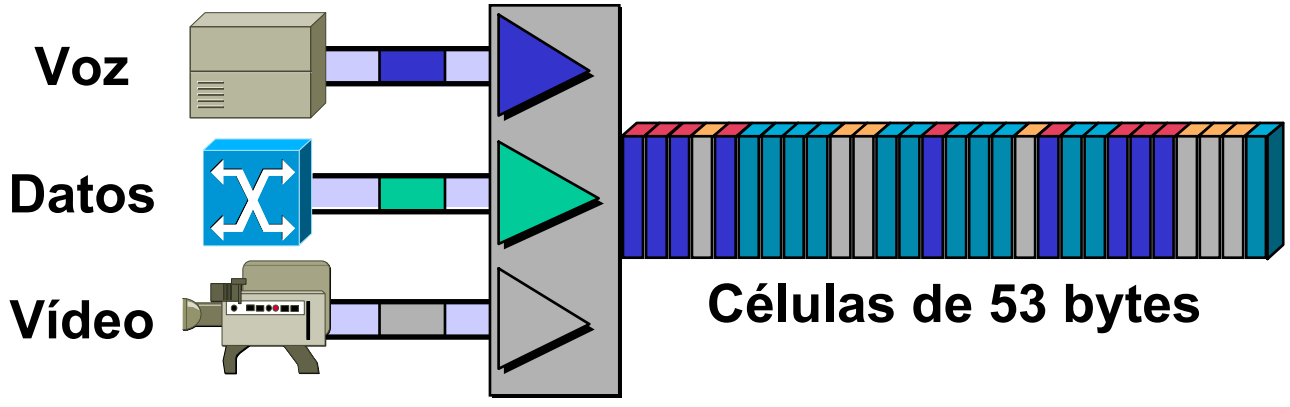
Ejemplo de mapa de circuitos establecidos con Frame-Relay



Broadband-ISDN y ATM (Asynchronous Transfer Mode)

- RDSI (o ISDN, Integrated Services Digital Network) es una red que integra voz y datos, que ha tenido dos propuestas, una para banda estrecha BE (narrow band) y otra más ambiciones para banda ancha BA (broad band).
- B-ISDN (o RDSI-BA) es un concepto: red de alta capacidad con posibilidad de cursar tráfico multimedia (voz, datos, video, etc.)
- En 1986 la CCITT eligió la tecnología ATM para implementar las redes B-ISDN
- ATM es un servicio de conmutación de *celdas* (paquetes pequeños y todos del mismo tamaño). Especialmente adaptado para tráfico a ráfagas (‘bursty traffic’)
- Una celda 53 bytes (5 de cabecera y 48 de datos).
- A nivel físico utiliza preferentemente SONET/SDH (155,52 Mb/s)
- Gran control sobre tipos de tráfico, posibilidad de negociar prácticamente todos los parámetros de QoS, prioridades, etc.
- La creación del ATM Forum en 1991 implicó a los fabricantes de equipos, lo cual dio un gran impulso a la tecnología ATM.

Características de ATM



Utiliza células de tamaño fijo, 53 bytes, lo que permite conmutar por hardware y disponer de memoria optimizada para trabajar con células ATM

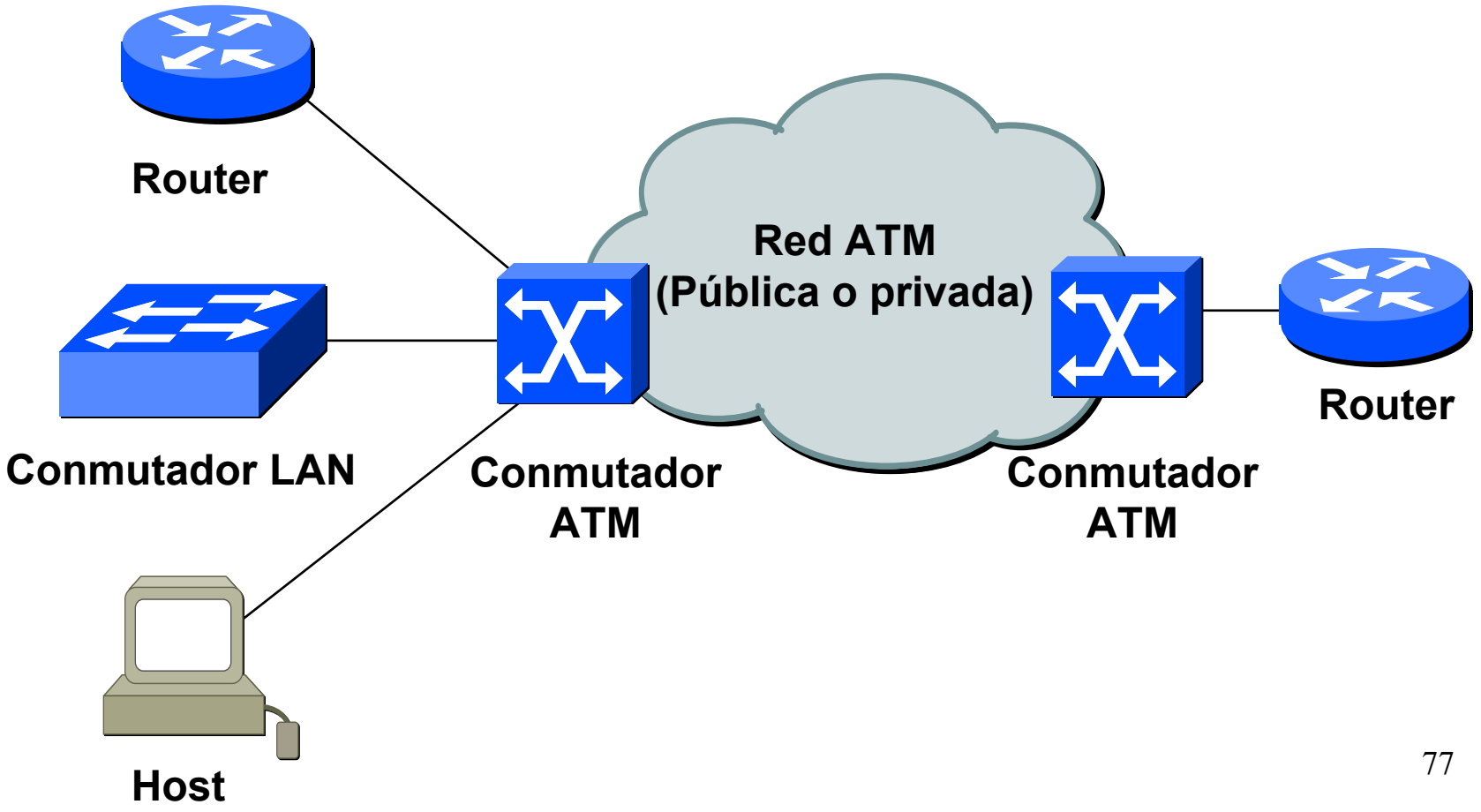
Actualmente su utilización queda relegada sólo a las operadoras, ya que la gran mayoría de los administradores de redes, utilizan tecnologías GigabitEthernet en ve de ATM para sus conexiones WAN.

ATM sufrió un lento proceso de estandarización y todo lo que conlleva.

Telefónica ofrece este servicio como Gigacom

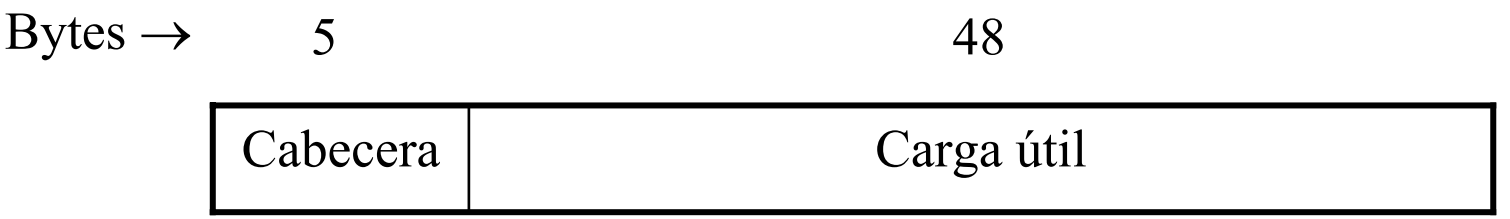
Orientado a conexión, con diferentes tipos de servicios: *CBR (Constant Bit Rate, para voz), VBR (Variable Bit Rate, para vídeo digital), y servicios sin calidad garantizada como son: ABR (Available Bit Rate) y UBR (Unspecified Bit Rate)*

Ejemplo de uso de una red ATM para transmisión de datos



Nivel de enlace en ATM

- Estructura de una celda ATM:



El tamaño (48 bytes) fue elegido por la ITU como compromiso entre la postura de las PTT europeas (16-32 bytes) y los fabricantes de ordenadores (128-64 Bytes).

Los 5 bytes de la cabecera llevan información del circuito virtual y CheckSum de dicha cabecera, también llamado *(Header Error Control)*.

Identificación de celdas ATM

Las celdas no llevan un delimitador. Para averiguar donde empiezan se usan dos técnicas:

1. Características del medio físico. Por ejemplo en SONET/SDH la información de control de línea contiene un puntero que indica el principio de una celda ATM en la trama
2. Tanteo del HEC: se busca en el flujo de bits recibido una secuencia de 40 bits en la que los ocho últimos sean el HEC de los 32 primeros. Cuando se encuentra uno válido se confirma en las cuatro celdas siguientes

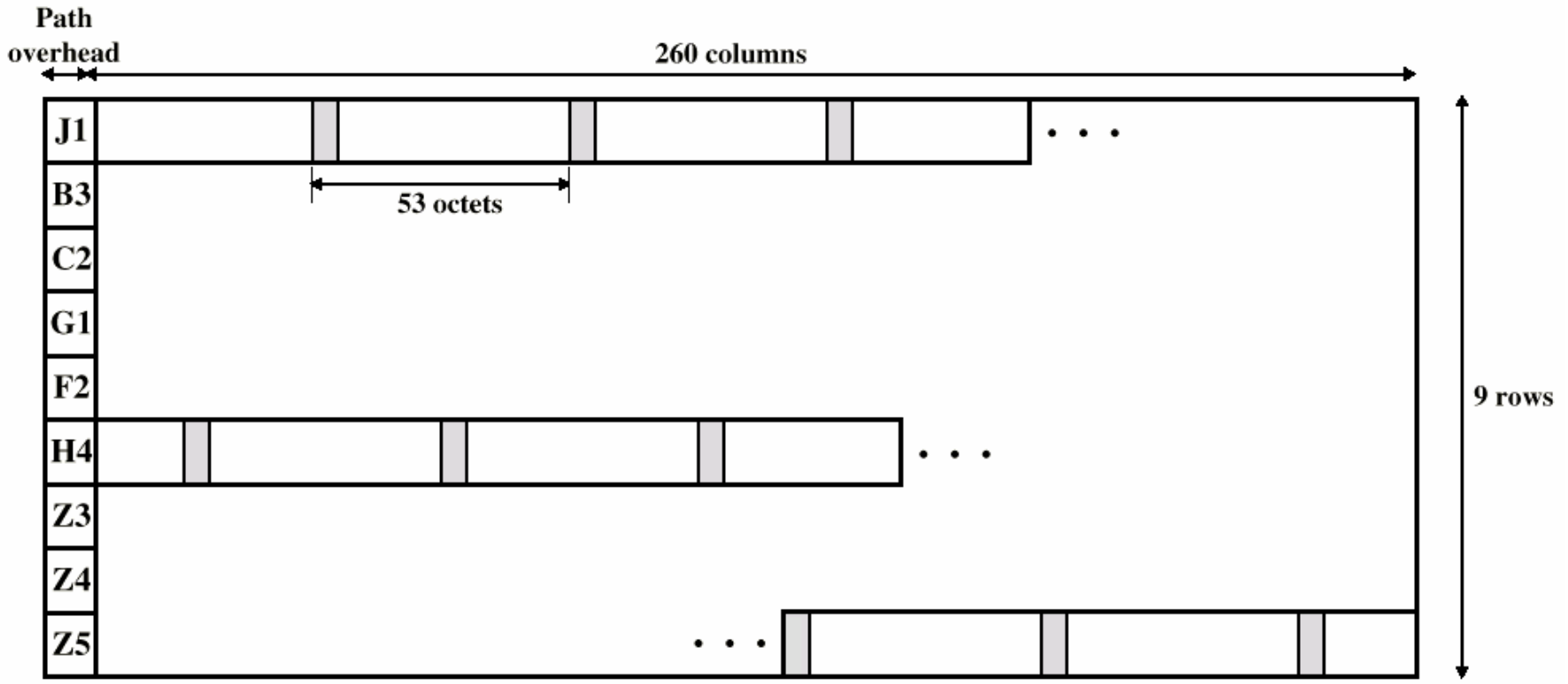


Figure 11.11 STM-1 Payload for SDH-Based ATM Cell Transmission