

# Tema 5. Protocolo X.25

*Índice*

Funcionamiento

Paquetes

Consideraciones Finales

## Tema 5. El Protocolo X.25

*Enrique Alba Torres*



Lenguajes y Ciencias  
de la Computación

*Universidad de Málaga (UMA)*

# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

## Introducción

- Para evitar evoluciones hacia estándares distintos en cada país la CCITT propuso en 1974 protocolos para las capas 1, 2 y 3, y se revisaron desde 1976 cada cuatro años.
- La capa 3 se llama **PLP X.25** (*Packet Layer Protocol*). Es un protocolo orientado a la conexión muy usado (ajustado a OSI).
- Terminología:
  - ✓ **DTE** (*Data Terminal Equipment*): Es un host.
  - ✓ **DCE** (*Data Circuit-Terminating Equipment*): Es el nodo interfaz con el DTE.
  - ✓ **DSE** (*Data Switching Exchange*): Es un nodo IMP intermedio.
- X.25 define el formato y significado de la información intercambiada entre DTE y DCE en las tres primeras capas (llamados **interfaz DTE-DCE**).
- El resto de los protocolos para las capas superiores **no están definidos** en X.25 en absoluto.

# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

## Introducción

- **Capa 1**  
Especifica la interfaz eléctrica, mecánica, procedimental y funcional entre DTE y DCE. Realmente X.25 no los define, sino que referencia a los estándares **X.21** y **X.21 bis**, que definen interfaces digitales y analógicas, respectivamente. El X.21 bis es el usado temporalmente hasta tener redes digitales disponibles a gran escala. Básicamente se trata de **RS-232-C**.
- **Capa 2**  
Se encarga de asegurar una comunicación confiable entre DTE y DCE, aunque estén conectados por una línea telefónica con ruido. Utiliza los protocolos **LAP** y **LAPB**.
- **Capa 3**  
Maneja conexiones entre pares de DTEs. Existen dos formas de conexión: **Llamadas Virtuales** y **Circuitos Virtuales Permanentes**. Una llamada virtual consiste en establecer-usar-liberar una conexión. Un CV permanente es como una línea alquilada: siempre disponible y cualquier DTE puede comenzar a transmitir sin establecer conexión (son aconsejables para grandes volúmenes de datos, envíos inconstantes, ...).

# Tema 5. Protocolo X.25

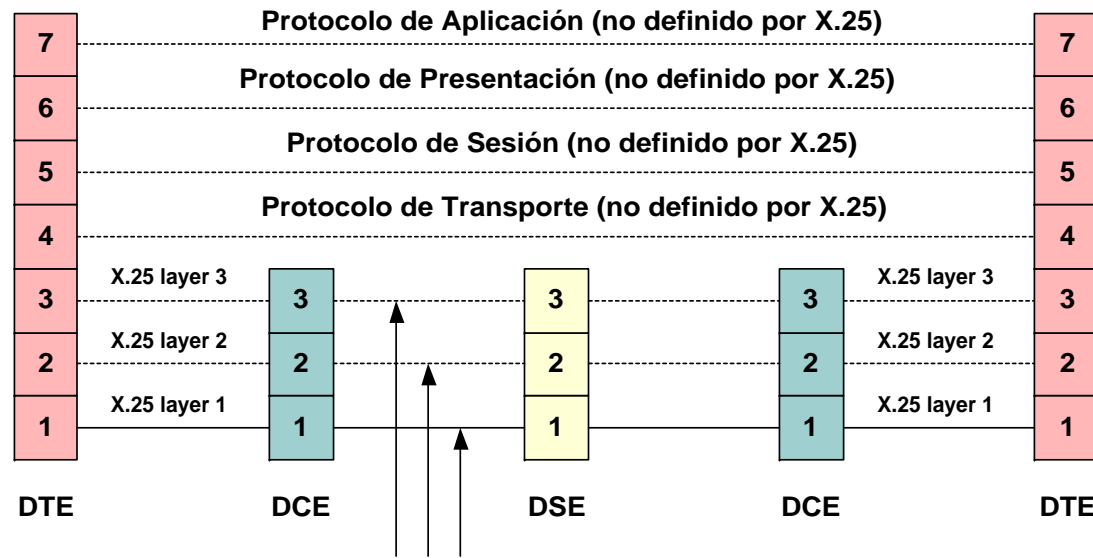
## Índice

### Funcionamiento

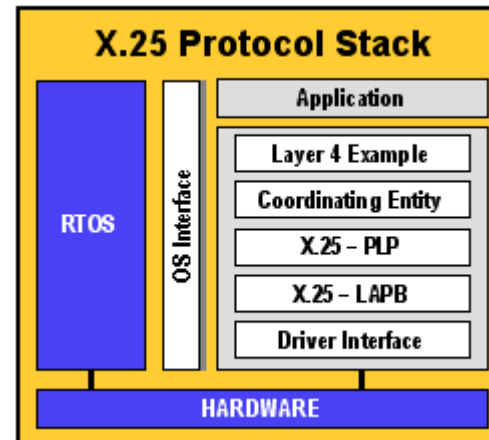
### Paquetes

### Consideraciones Finales

# Introducción



Los protocolos internos no están definidos por X.25



# Tema 5. Protocolo X.25

## Índice

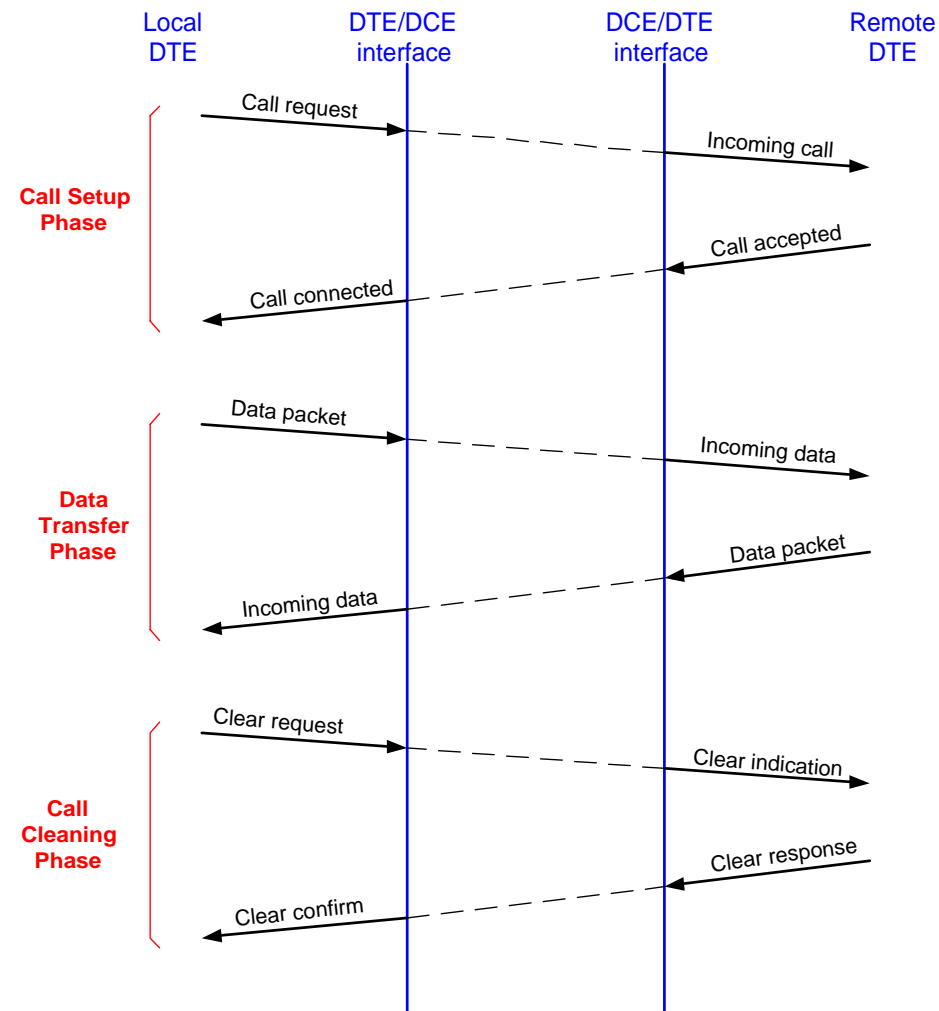
### Funcionamiento

### Paquetes

### Consideraciones Finales

## Establecimiento de la Conexión

- El establecimiento de la conexión en X.25 consta de 3 fases:



## Tema 5. Protocolo X.25

### Índice

#### Funcionamiento

#### Paquetes

#### Consideraciones Finales

## Establecimiento de la Conexión

- El establecimiento de la conexión en X.25 consta de 3 fases:
  - ✓ El DTE que inicia la apertura de la conexión construye un paquete *CALL REQUEST* y se lo pasa a su DCE. La subred se lo pasa al DCE destino quien lo entrega al DTE correspondiente. Si este último acepta la llamada devolverá un paquete *CALL ACCEPTED* que llegará al DTE origen como *CALL CONNECTED*.
  - ✓ La siguiente fase consiste en el uso full-dúplex del CV para intercambiar datos.
  - ✓ Cuando uno de ellos quiere terminar envía un paquete *CLEAR REQUEST* al otro extremo quien se lo confirmará con un paquete *CLEAR CONFIRMATION*.

## Tema 5. Protocolo X.25

### *Índice*

#### Funcionamiento

#### Paquetes

#### Consideraciones Finales

## Establecimiento de la Conexión

- La elección del número de circuito virtual la hace el DTE origen (cualquier número no usado) y cuando la llamada llega al DCE destino si este número de CV ya está siendo usado en su DTE lo cambia.
- Luego el **número de CV** lo elige el DTE en su salida y el DCE destino en la entrada. Si un DTE y su DCE origen eligen el mismo número de CV a la vez se produce una **Colisión de Llamada**. En X.25 la llamada de salida progresa y la de entrada se descarta (su origen ya lo intentará más tarde).
- Para minimizar las colisiones el **DTE** elige el **mayor** identificador para las llamadas de salida mientras que el **DCE** elige el **menor** disponible para las de entrada.

# Tema 5. Protocolo X.25

## Índice

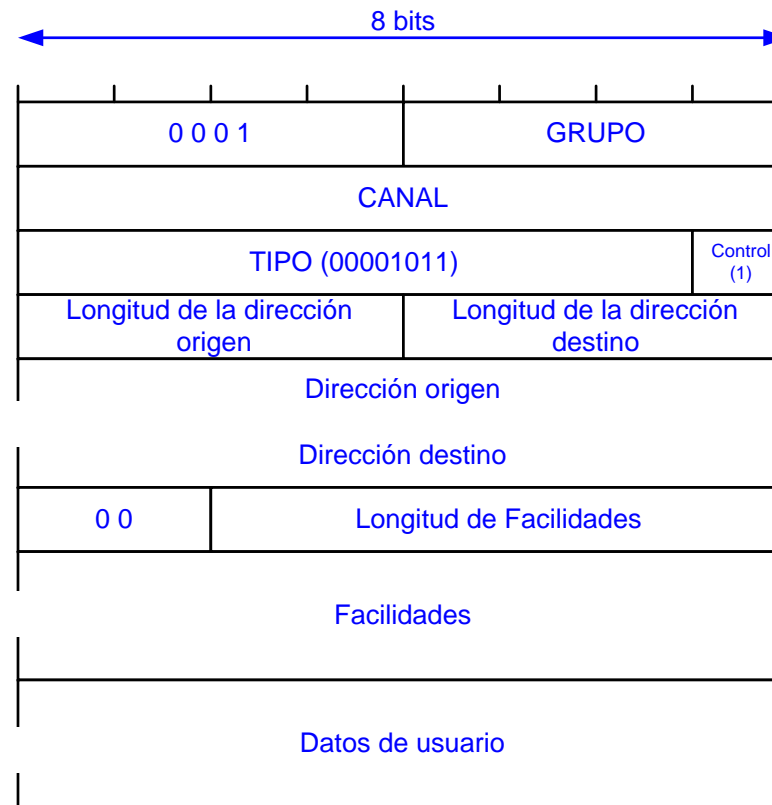
### Funcionamiento

### Paquetes

### Consideraciones Finales

## Paquete Call Request

- La estructura de un paquete *CALL REQUEST* es la siguiente:





# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

# Paquete Call Request

- La **cabecera** ocupa 3 octetos (terminología CCITT). El primer campo es 0001.
- Los campos **GRUPO** y **CANAL** juntos forman un número de CV de 12 bits. Por separado no tienen sentido. El número 0 está reservado para el futuro y así el DTE puede tener hasta 4095 CVs abiertos simultáneamente.
- El campo de **TIPO** para *CALL REQUEST* vale 00001011.
- El campo **CONTROL** está a 1 en los paquetes de control (como este paquete) y a 0 en los paquetes de datos.
- Las **direcciones** del que llama y del llamado (origen y destino) están codificadas como dígitos decimales usando **4 bits por dígito (X.121)**. El sistema es similar al de teléfonos: código del país, código de la red y dirección dentro de la red. La dirección completa puede ser de **hasta 14 dígitos decimales** (los 3 primeros del país). Algunos países con muchas redes tienen más de un código de país asignados.

# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

## Paquete Call Request

- El campo **FACILIDADES** (y su **longitud**) permiten usar características especiales para la conexión. Dependen de la subred usada. Algunos ejemplos son cobro revertido, llamadas colectivas, entrega de alta prioridad o usar CV simplex.
- El llamante puede especificar la longitud máxima de un paquete y el tamaño de ventana (por defecto 128 bytes y 2 paquetes). Si al receptor no le gustan puede hacer una contraoferta en su paquete *CALL ACCEPTED*, únicamente acercando estos valores a los valores por defecto, nunca alejándolos. Algunas facilidades pueden elegirse al suscribirse a la red: grupos de usuarios cerrados, máxima ventana a 7, velocidad de la línea o prohibición de ciertas llamadas.
- El campo de **datos de usuario** permite al DTE enviar hasta 16 bytes de datos en el paquete *CALL REQUEST*. Los DTEs deciden cómo manejarlos: a qué proceso se desea la conexión, una palabra de paso para hacer login, etc...

# Tema 5. Protocolo X.25

## Índice

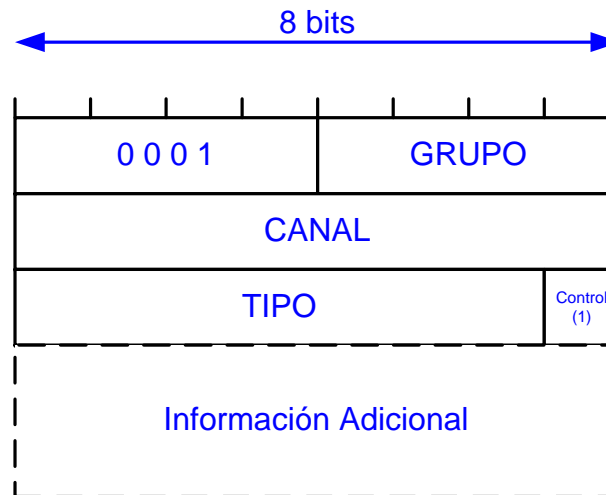
### Funcionamiento

### Paquetes

### Consideraciones Finales

## Otros Paquetes de Control

- El resto de **paquetes de control** sólo pueden tener **cabecera**.



Tipo de paquete	Campo TIPO
DATA	PPPMSSS0
CALL REQUEST	00001011
CALL ACCEPTED	00001111
CLEAR REQUEST	00010011
CLEAR CONFIRMATION	00010111
INTERRUPT	00100011
INTERRUPT CONFIRMATION	00100011
RECEIVE READY	PPP00001
RECEIVE NOT READY	PPP00101
REJECT	PPP01001
RESET REQUEST	00011011
RESET CONFIRMATION	00011111
RESTART REQUEST	11111011
RESTART CONFIRMATION	11111111
DIAGNOSTIC	11110001

- Aunque algunos pueden tener 1 o 2 bytes adicionales. Por ejemplo, el cuarto byte de un *CLEAR REQUEST* informa de por qué se está liberando la conexión. Este paquete podría generarlo la subred cuando el *CALL REQUEST* no puede progresar.

# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

## Otros Paquetes de Control

- Además del paquete *CALL REQUEST* existen los siguientes paquetes de control:
  - ***INTERRUPT***: permite enviar fuera de secuencia señales cortas. Un uso típico es pulsar QUIT o BREAK en un terminal. Se confirma con el paquete ***INTERRUPT CONFIRMATION***.
  - ***RECEIVE READY***: se usa para enviar confirmaciones separadas cuando no hay tráfico inverso y no puede realizarse piggyback.
  - ***RECEIVE NOT READY***: permite al DTE parar al otro extremo por un momento. Después puede usarse el paquete ***RECEIVE READY*** para indicar al DCE que siga.
  - ***REJECT***: permite al DTE pedir la transmisión de una serie de paquetes.
  - ***RESET*** y ***RESTART***: se usan para la recuperación de problemas a varios niveles.
  - ***DIAGNOSTIC***: permite a la red informar de problemas al usuario.

# Tema 5. Protocolo X.25

## Índice

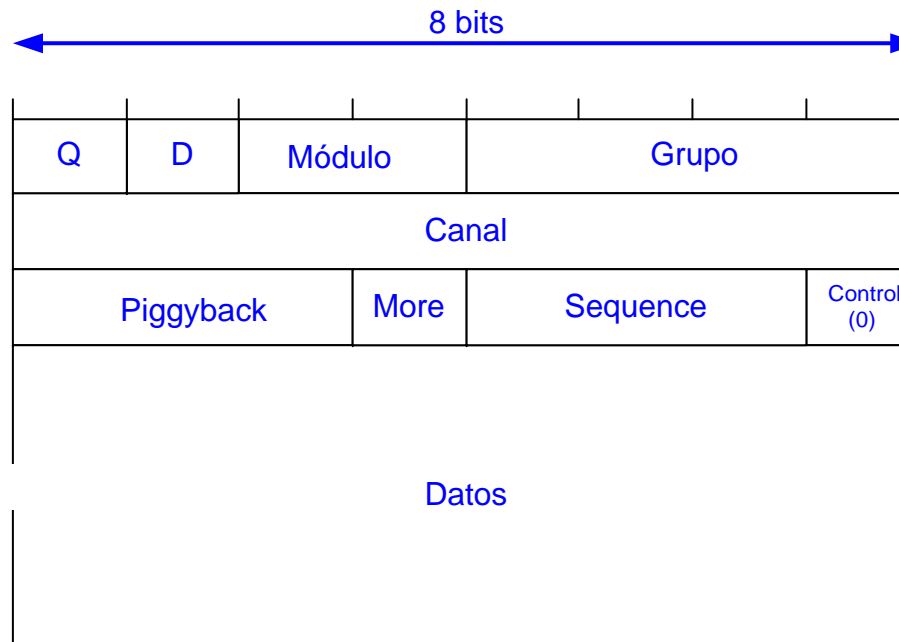
### Funcionamiento

### Paquetes

### Consideraciones Finales

## Paquetes de Datos

- La estructura de un paquete de datos es la siguiente:



# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

## Paquetes de Datos

- El **bit Q** indica datos cualificados. Esto no está definido en X.25 pero la idea es que el protocolo de transporte lo use para separar sus paquetes de control y datos.
- El campo **CONTROL** está siempre a 0.
- Los campos **SECUENCIA** y **PIGGYBACK** se usan para control del flujo por un mecanismo de ventana deslizante. Los números de secuencia son Mod-8 si el campo **MODULO** es 01 y Mod-128 si es 10 (00 y 11 son ilegales). Si se usan números de secuencia Mod-128 la cabecera se extiende con un byte extra para acomodar los campos más largos de secuencia y piggyback (4 bits adicionales para cada uno).
- El significado del campo de piggyback lo determina el **bit D**:
  - ✓ **D=0** → la siguiente confirmación indica que el DCE local recibió el paquete, y no que el DTE remoto lo recibió.
  - ✓ **D=1** → la confirmación es realmente indicio de una recepción correcta del paquete en el DTE remoto al otro extremo.
- Aunque D valga 0 el campo de piggyback es útil pues le indica al DTE que su DCE está listo para el próximo paquete (controlando así el flujo DTE→DCE).

# Tema 5. Protocolo X.25

## Índice

### Funcionamiento

### Paquetes

### Consideraciones Finales

## Paquetes de Datos

- En una confirmación X.25 el DTE remoto devuelve el número de paquete que espera a continuación (no el último correctamente recibido). Es arbitrario pero obligatorio para ser compatible con X.25.
- El campo **MORE** es un bit a 1 en todos los paquetes menos en el último. Un paquete completo puede tener ese bit activo. La subred puede re-empaquetar datos del paquete en otros de distinta longitud si lo necesita, pero nunca combinará datos de mensajes distintos en el mismo paquete.
- El estándar especifica que el transporte de señales debe permitir una longitud máxima de **128 bytes de datos** por paquete (este valor se usa por eficiencia), así como establecer como máximos optativos 16, 32, 64, 256, 512, 1024, 2048 y 4096 bytes. Esto es negociable. Usar menos de 128 bytes sirve para proteger a los terminales con poca memoria (pocos buffers) de paquetes de entrada largos.

# Tema 5. Protocolo X.25

## Índice

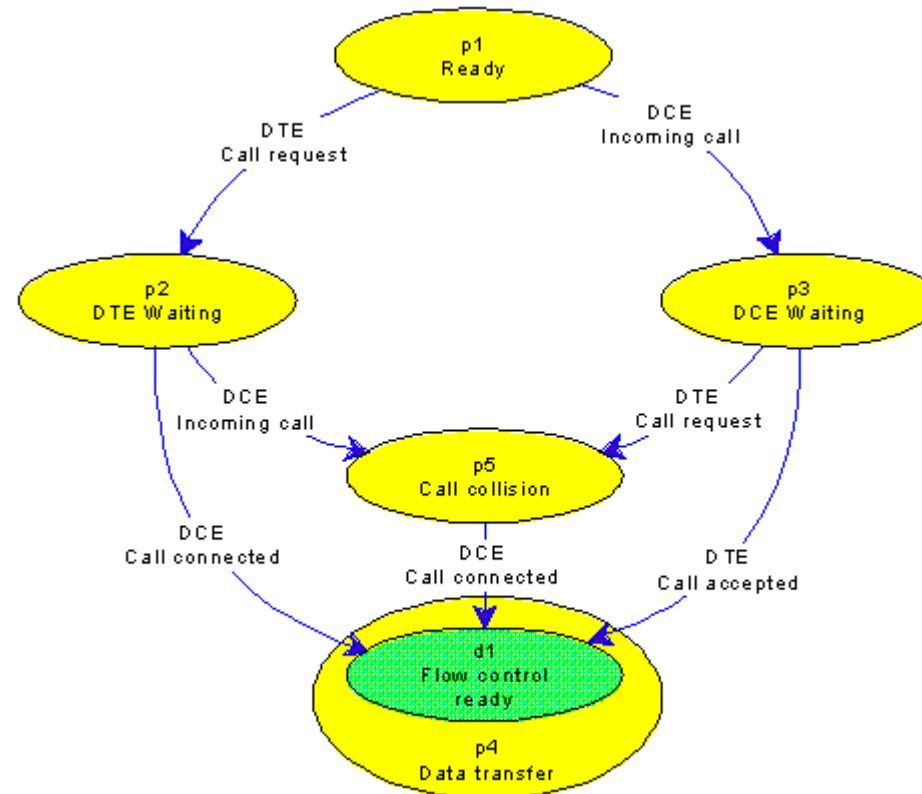
Funcionamiento

Paquetes

Consideraciones Finales

## Consideraciones Finales

- El estándar X.25 contiene varios **diagramas de estado** para describir secuencias de eventos tales como establecer o cerrar una conexión, reseteo o recomienzo.





# Tema 5. Protocolo X.25

## Índice

Funcionamiento

Paquetes

Consideraciones Finales

## Consideraciones Finales

- El estándar original X.25 (1976) es como el descrito aquí menos el bit D, los paquetes de DIAGNÓSTICO, la negociación sobre longitud de paquete y tamaño de ventana (y otros detalles).
- Inicialmente existía una **demanda de facilidades datagrama**, además de CVs. Tanto USA como Japón las especificaron pero eran incompatibles entre ellas, y en 1980 la CCITT las admitió a ambas (!) complicando aún más el protocolo. Sin embargo las PTTs (proveedores) con 100 años en telefonía no implementaron los datagramas y varios años después se comprobó que nadie los usaba ni quería, así que se eliminaron en el estándar de 1984.

### Facilidades

Números de secuencia extendidos (7 bits)  
Tamaño de ventana no estándar  
Tamaño de paquete no estándar  
Establecer prestaciones (75 bps a 48 kbps)  
Pedir cobro revertido  
Seleccionar proveedor (telenet, tymnet, ...)  
Sólo datos de salida  
Sólo datos de entrada  
Usar go-back-n frente a repetición selectiva  
Usar selección rápida

# Tema 5. Protocolo X.25

## Índice

Funcionamiento

Paquetes

Consideraciones Finales

## Selección Rápida

- Por otro lado la demanda de un **servicio sin conexión** de algún tipo persiste para aplicaciones como TPVs, verificación de tarjetas de crédito o transferencia de fondos bancarios. Estos **servicios transaccionales** no necesitan realmente abrir y cerrar conexión. Para satisfacerlos la CCITT añadió al estándar la llamada por **SELECCIÓN RÁPIDA** en 1984.
- En esta llamada el paquete *CALL REQUEST* se expande para incluir **128 bytes** de datos de usuario. Esto se pide como **facilidad**. Por lo que concierne a la red el paquete es un intento de establecer un circuito virtual. El DTE destino puede negarse a la llamada con un paquete *CLEAR REQUEST*, también expandido para incluir 128 bytes de datos de respuesta.
- También se podría aceptar la llamada y establecer así un circuito virtual normal, depende de si la aplicación lo necesita o no, ya que para muchas aplicaciones 128 bytes les permite comunicar datos útiles sin mayores requisitos futuros.

# Tema 5. Protocolo X.25

## Índice

Funcionamiento

Paquetes

**Consideraciones Finales**

## Primitivas OSI y X.25

Network Service Primitive	X.25 Action
N-CONNECT.request N-CONNECT.indication N-CONNECT.response N-CONNECT.confirm	Send CALL REQUEST CALL REQUEST arrives Send CALL ACCEPTED CALL ACCEPTED arrives
N-DISCONNECT.request N-DISCONNECT.indication	Send CLEAR REQUEST CLEAR REQUEST arrives
N-DATA.request N-DATA.indication	Send DATA PACKET DATA PACKET arrives
N-DATA-ACKNOWLEDGE.request N-DATA-ACKNOWLEDGE.indication	- -
N-EXPEDITED-DATA.request N-EXPEDITED-DATA.indication	Send INTERRUPT INTERRUPT arrives
N-RESET.request N-RESET.indication N-RESET.response N-RESET.confirm	Send RESET REQUEST RESET REQUEST arrives - -

# Tema 5. Protocolo X.25

## Índice

Funcionamiento

Paquetes

Consideraciones Finales

## Tendencias

- La tendencia es a la migración de las empresas a *FRAME RELAY* (retransmisión de tramas): **DIGITAL**.
- La retransmisión de tramas está pensada para transmitir datos y el control de los retrasos no es importante.
- Ventajas de *FRAME RELAY*:
  - Rapidez.
  - Hardware multiplexado en tiempo/estadístico (STDM).
  - VPN (Virtual Private Networks).
  - Menor control de errores (este control lo realiza el nivel de enlace).