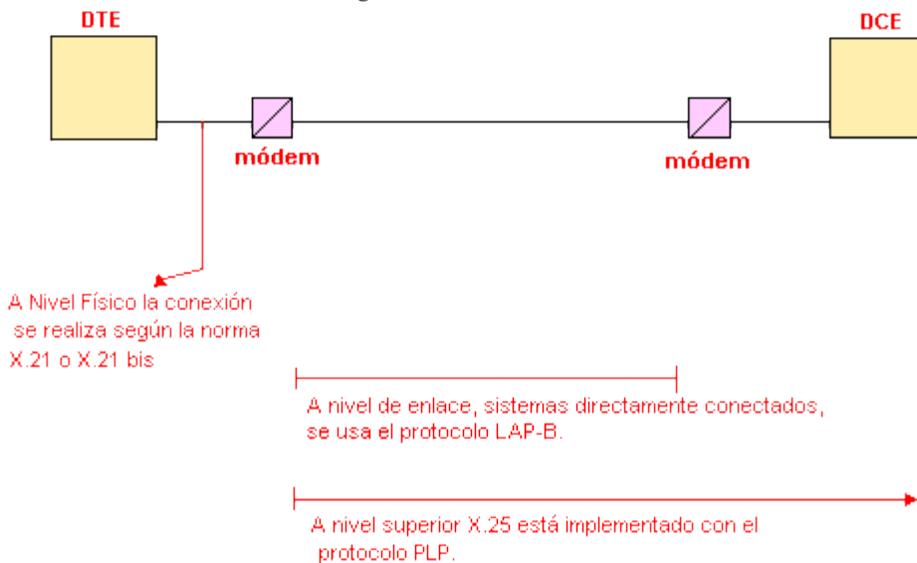


## Redes X.25

X.25 es un estándar para el acceso a redes públicas de conmutación de paquetes. No especifica cómo está la red implementada interiormente aunque el protocolo interno suele ser parecido a X.25. El servicio que ofrece es orientado a conexión (previamente a usar el servicio es necesario realizar una conexión y liberar la conexión cuando se deja de usar el servicio), fiable, en el sentido de que no duplica, ni pierde ni desordena (por ser orientado a conexión), y ofrece multiplexación, esto es, a través de un único interfaz se mantienen abiertas distintas comunicaciones. El servicio X.25 es un diálogo entre dos entidades DTE Y DCE. Estudiaremos en X.25 desde el nivel Físico al nivel de Red. En primer lugar veamos cuál es la forma más común de conexión y lo que abarca cada nivel:

FIGURA 1 Conexión a X.25. Imagen enlace.



### Nomenclatura:

- **DTE** (Data Terminal Equipment): Es lo que utiliza el usuario final (PC con placa X.25 por ejemplo). Es el equipo terminal de datos. Incorpora los niveles 2 y 3.
- **DCE** (Data Circuit Terminating Equipment): Podemos interpretarlo como un nodo local. A nivel de enlace (LAPB) las conexiones se establecen DTE-DCE. Ahora con el nivel de red, ampliamos las comunicaciones más allá del DCE, que hace de interconexión. Sólo incluye el nivel 1.

Con X.25 no hay conexiones multipunto. Es un servicio punto a punto, por lo que sólo puedo conectar un DTE con otro DTE.

En X.25 se regulan:

- INTERFAZ A NIVEL FÍSICO.
- PROTOCOLO DE ENLACE.
- PROTOCOLO DE RED.

### Nivel Físico.

La interfaz de nivel físico regula el diálogo entre el DCE y el DTE.

Se describe desde 3 puntos de vista distintos:

1. Mecánico
2. Eléctrico.
3. Funcional.

Existen dos posibilidades para la interfaz a nivel físico:

- X.21: Se utiliza para el acceso a redes de conmutación digital. (Similares a las de telefonía digital.)
- X.21bis: Se emplea para el acceso a través de un enlace punto a punto. (Similar a RS-232 en modo síncrono.)

En cuanto la interfaz mecánica, se usan conectores *Canon* DB15 (de 15 pines, para X.21) o DB25 (de 25 pines para X.21 bis).

En cuanto a la interfaz eléctrica X.21 utiliza X.26, que es una interfaz no balanceada, por lo que se suele usar mejor X.27 que es balanceada y, por tanto, permite tasas de transmisión superiores. La interfaz eléctrica de X.21 bis está recogida en la norma V.28. Las velocidades se mueven entre los 64kbps y los 2Mbps, velocidades que pueden parecer bajas y, de hecho, así son. X.25 presenta un problema de baja eficiencia por la exagerada protección contra errores que implementa y que con las redes de hoy en día no tienen sentido.

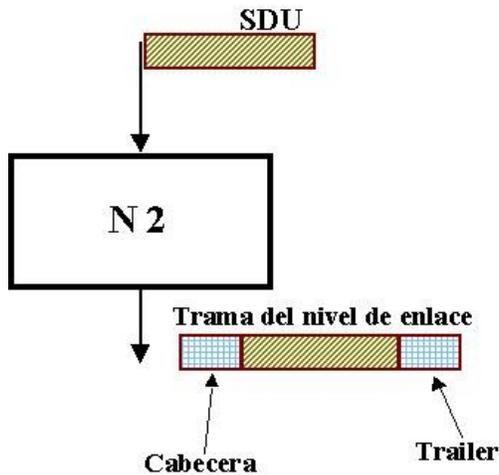
La interfaz funcional de X.21 se define mediante las distintas señales que intercambia el DB15. La interfaz funcional de X.21 bis está recogida en la norma V.24.

X.21 no ha tenido mucho éxito. X.21 bis es más popular.

## **Nivel de Enlace (LAP-B)**

Ya sabemos que el objeto del Nivel de Enlace es garantizar la comunicación entre dos equipos directamente conectados. En X.25, este nivel queda implementado con el protocolo LAP-B (*Link Access Procedure - B*) que es un protocolo HDLC 2,8, es decir, con rechazo simple, indicado por el 2, y en el cual las tramas de información pueden ser utilizadas como tramas de control, indicado esto último por el 8.

El servicio que ofrece el nivel 2 al nivel superior es orientado a conexión, fiable y en modo paquete. El nivel 2 sólo ofrece una conexión al nivel superior. El nodo local es el que presta servicio al DTE conectado a él. El nivel de enlace no resuelve el servicio extremo a extremo. La comunicación extremo a extremo la resuelve el nivel de red. El diálogo entre entidades de enlace es salto a salto. Por tanto, existen tantas conexiones de enlace como unidades tengamos entre el DTE local y DTE remoto. El nivel de enlace recibe peticiones del nivel de red, mediante primitivas, para que transmita un bloque de información, que le es opaco al nivel de enlace. Se pasa una SDU del nivel de red al nivel de enlace. El nivel de enlace le añade una cabecera y un trailer con objeto de detectar y evitar errores. Todo junto es una trama del nivel de enlace.



Hemos de distinguir aquí entre lo que es una SDU y lo que es una PDU:

- SDU: bloque de información que se intercambian dos niveles consecutivos.
- PDU: estructura de datos que se intercambian dos entidades gemelas.

La trama del nivel de enlace pasa al nivel físico, quien la transforma en binario y la pasa como señal al medio físico, por donde se transmite.

Si la trama no sufre errores llega a destino perfectamente. La entidad gemela puede reconstruir la trama porque sabe donde empieza y termina gracias a la información de control que metimos en la cabecera y el trailer. La parte central de la trama (campo de información) es extraída por la entidad de protocolo del nivel de enlace y se entrega al nivel superior. Esto ocurre en ausencia de errores. Por tanto el nivel de enlace permite que los niveles de red de las entidades gemelas puedan intercambiar PDUs.

### ***Nivel 3 o nivel de red en la recomendación X.25.***

#### **4.1 Introducción.**

Este nivel está especificado por el **PLP (Packet Layer Protocol)** que es un protocolo de acceso a nivel de red y que proporciona un servicio al nivel superior:

- *de subred* ([SNACP](#)).
- *modo paquete*
- [orientado a conexión](#).
- *fiable*.
- *multiplexión*: uso de una conexión para varias comunicaciones simultáneas. El DTE origen dialoga con su nodo, pero *virtualmente* lo hace con todos los DTEs multiplexados.

#### **4.2 Circuitos virtuales (CV).**

Podríamos definirlos como la asociación lógica entre usuarios para comunicarse entre ellos. En X.25 hay 2 tipos de CV:

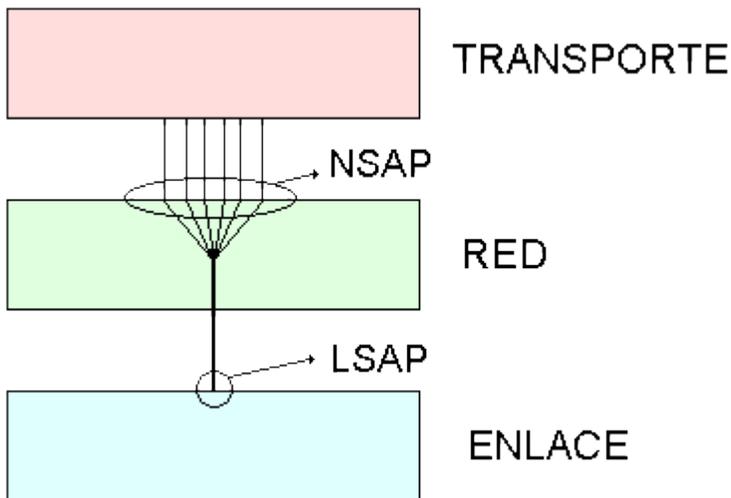
- **Conmutados (CVC)** : Hay que realizar un diálogo previo a la transmisión con el nodo local para establecerlos y para liberarlos.

- **Permanentes (CVP):** Están establecidos de antemano (por contrato), así que no hace falta *fase de establecimiento ni de liberación*. Se preconfiguran los nodos de tal forma que, por contratación, el circuito está permanentemente establecido. Son muy útiles si se transmite mucho y con mucha frecuencia hacia un mismo destino.

Se identifican dentro de cada DTE por el número de canal lógico (NCL), que se negocia en la fase de establecimiento (sólo CVCs).

Podría además tener, por ejemplo, varios CVs establecidos con la misma máquina (cada uno con distinto NCL evidentemente).

**FIGURA 4.19** *Los CVs y la multiplexión en terminología OSI.*



En la figura se muestra como la multiplexión que se ofrece al nivel de transporte, no es tal a nivel de enlace: en LAPB sólo hay una conexión.

La multiplexión se resuelve a nivel de red, aunando las diferentes conexiones (asimilables a CVs) que aparecen en el *NSAP* (Punto de Acceso al Servicio a Nivel de Red), en la que se ve desde el nivel de enlace en el *LSAP*.

### 4.3 Protocolo.

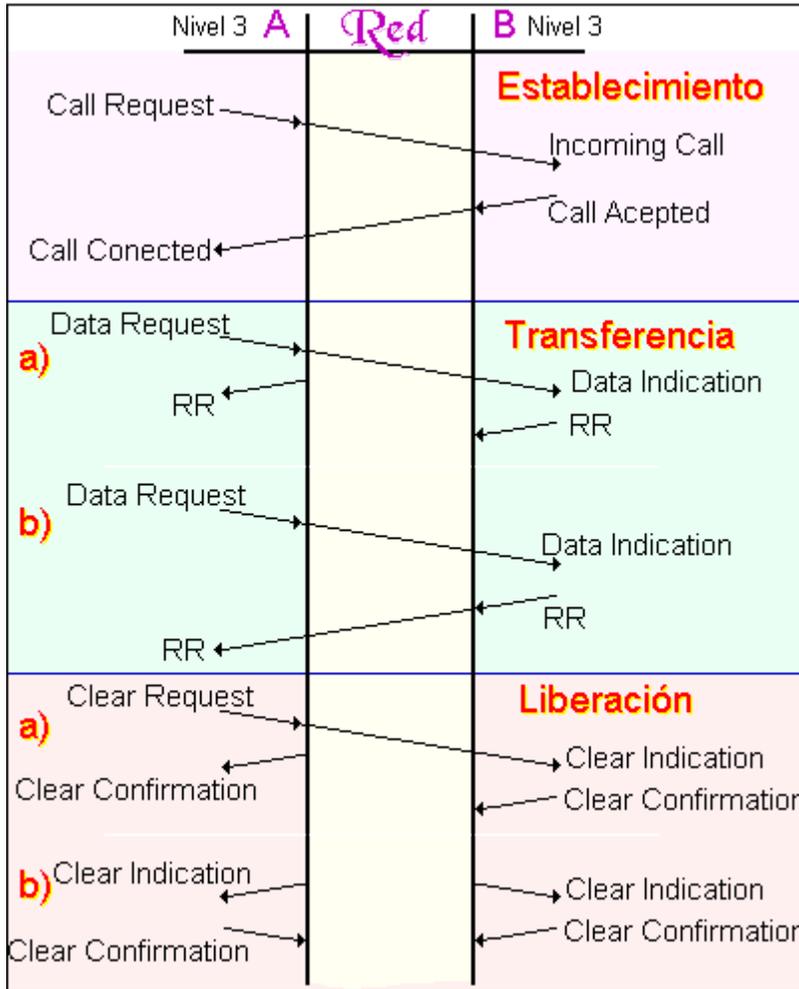


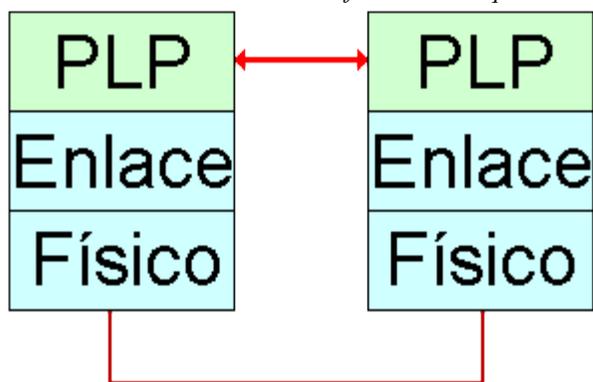
FIGURA 4.20 Fases de la Transmisión.

- **Fase de Establecimiento:** En la figura hemos supuesto que la llamada es aceptada, pero como veremos más adelante, podría ser rechazada. Esta fase sólo tiene lugar para CVCs.

Llegados a este punto ambos lados estarán seguros de que la conexión se estableció bien.

- **Fase de Transferecia:** Como veremos, los datos pueden ser asentidos en el nodo local (caso 'a'), o en destino ('b').
- **Fase de Liberación:** La liberación a su vez puede ser solicitada por uno de los dos lados ('a') o por la propia red ('b').

FIGURA 4.21 Gráfica esquemática con los niveles OSI involucrados.



- Los que dialogan son los dos PLPs. El nivel de enlace sólo sirve de mensajero
- Las direcciones a nivel de enlace son distintas de las de nivel de red.
- Con la dirección de enlace (que ya vimos que no se necesitaba realmente) llego al primer nodo. Allí se desencapsula y se usa la de red para llegar a los demás.

NOTA: A nivel de paquete no tenemos retransmisiones. Sí hay control (detección) de errores, pero no corrección.

#### 4.4 Número de canal lógico (NCL).

Es un número que permite identificar al CV involucrado en una determinada transferencia y que es distinto a cada lado de la comunicación, aunque el CV sea el mismo. El rango de NCL que pueden usarse, es algo a negociar con la empresa que ofrece el servicio (Telefónica, etc. ). Más NCL, mayor número de CVs establecibles. Un NCL se especifica con 12bits, lo cual da lugar a que puedan usarse como máximo 4095 NCLs (el 0 tiene un significado especial).

##### Utilización:

Los NCLs se escogen por el DTE o por el DCE (la red en el fondo) cuando se necesitan, liberándolos cuando los acaban de usar. Ambos tienen una lista donde marcan los NCLs libres y ocupados (lo que se marca en una lista se refleja inmediatamente en la otra).

- El DTE empieza a escoger por los NCLs de mayor numeración.
- El DCE (la red) empieza por los de menor numeración.
- Podría ocurrir que se juntasen *en el centro* (los DTE vienen de arriba y los DCE de abajo) y esto desemboca en varias posibilidades:
  1. Que cuando DTE o DCE vayan a escoger un número, en sus listas figuren todos como ocupados. En este caso, no se aceptarían sus paquetes.
  2. Que sólo quede un NCL por elegir y los dos lo cojan al mismo tiempo. En este caso la red (DCE) tendría prioridad. La conexión del DTE se contesta con un *clear* desde la red y se rechaza. (ver figura 4.23 a la derecha).

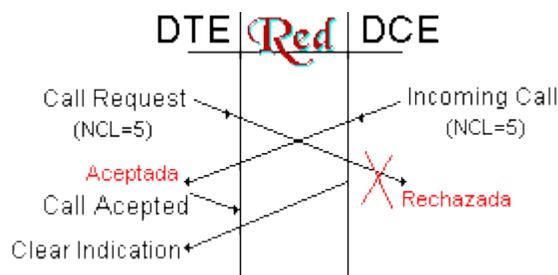


FIGURA 4.23 Ejemplos del uso de NCLs.

## NCL=10

Lo coge el DTE      NCL=10  
 →  
 Call Request

Lo coge la red      NCL=1  
 ←  
 Incoming Call

Comentario a la **Figura 4.23**:

En respuesta a un *Call Request* anterior (que el DTE asignó sin problemas al NCL 6 por ejemplo), el DCE trata de asignar el NCL 5 pues lo ve libre. Así el CV de esa conexión tendría asociado el NCL 6 en el DTE y el 5 en el DCE.

Al mismo tiempo el DTE ha visto libre el NCL 5 y trata de establecer un nuevo CV asignándosele

Como consecuencia de esto es la operación del DCE (de la red) la que se acepta, rechazándose el *Call Request* del DTE.

Una posible solución para evitar colisiones de este tipo, es dividir por rangos la oferta de NCLs. Por ejemplo asignar una cierta cantidad de números para CVs entrantes, otra para salientes y otros que fuesen bivalentes. Así solo habría colisión en los bivalentes, pues los entrantes y salientes sólo podrían ser elegidos por DTE y DCE respectivamente.

Se verá su aplicación más directa cuando se estudien los paquetes de datos ([Ver](#))

## 4.5. PDUs DEL NIVEL 3 EN X.25.

Los niveles de red de las entidades que quieren establecer la comunicación intercambian PDUs (o paquetes) mediante los servicios prestados por el nivel de enlace. Estos paquetes se estructuran en:

- [Paquetes para establecimientos de conexiones.](#)
- [Paquetes para el intercambio de datos normales.](#)
- [Paquetes para el intercambio de datos acelerados.](#)
- [Paquetes para el reinicio y rearranque de conexiones.](#)
- [Paquetes para la liberación de conexiones.](#)

### 4.5.1. ESTABLECIMIENTO DE CONEXIONES.

Se definen 4 tipos de paquetes para el establecimiento:

1. [Petición de llamada.](#)
2. [Llamada entrante.](#)
3. [Llamada aceptada.](#)
4. [Comunicación establecida.](#)

Estos paquetes se usan para las distintas fases de la conexión.

La conexión se establece por iniciativa de la entidad de nivel superior. El nivel 4 de la entidad local va a dialogar con el nivel 4 de la entidad remota (diálogo extremo a extremo). Para ello hay que tener una conexión a nivel 3 (que es el que proporciona el servicio extremo a extremo).

El nivel 4 de la entidad local manda una primitiva de comunicaciones al nivel 3, éste construye un paquete que se usa para solicitar conexión a la res. Este paquete es el de petición de llamada. Esta PDU se envía a la red. Para ello se solicita al nivel de enlace que sea transmitida. Puesto que el nivel de enlace es orientado a conexión, la conexión a nivel de enlace se establece cuando el sistema arranca. El nivel de enlace consigue que la SDU que le había mandado el nivel de red llegue al nivel de enlace del nodo remoto. Allí se construye otro paquete, el paquete de llamada entrante. Este paquete se entrega al nivel 3 del DCE remoto. El nivel 3 del nodo remoto detecta que se le pide una conexión y pregunta la nivel 4 mediante una primitiva, y es el nivel 4 (el nivel superior) quien rechaza o acepta la conexión. Si el nivel 4 decide aceptar la conexión se lo comunica al nivel 3 mediante otra primitiva. El nivel 3 genera entonces otro paquete, el de llamada aceptada. El nivel 3 se lo pasa al nivel 2, éste lo convierte en bits y se lo pasa al nivel físico, quien lo convierte en señal se transmite y llega al nivel 3 del nodo remoto donde se destruye. Por procedimientos internos que desconocemos la red hace llegar al nivel 3 del nodo local la información, donde se genera el paquete de comunicación establecida y este paquete llega al nivel 3 del DTE local y mediante una primitiva llega al nivel superior informándole de que la comunicación está establecida.

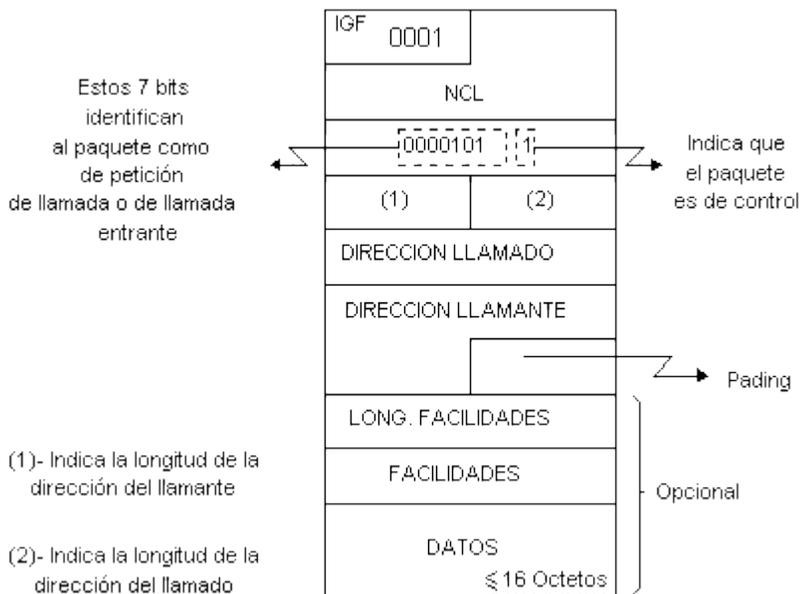
El nivel 4 de la entidad remota, a partir de que acepta la conexión pasa a la fase de transmisión de datos. El nivel 4 de la entidad local no considera que la conexión se ha establecido con éxito hasta que le llega la última primitiva de conexión establecida.

Veamos ahora los formatos de estos paquetes.

#### **4.5.1.1. Paquetes de petición de llamada, llamada entrante .**

Los paquetes de **petición de llamada** y **llamada entrante** tienen el mismo formato, diferenciándose únicamente en el sentido en que se transmite el paquete: DTE-DCE el primer caso y DCE-DTE en el segundo. Así pues tendremos:

**FIGURA 4.25** *Formato general de Petición de Llamada o Llamada Entrante.*



- El **IGF** es el índice general de formato. Su longitud es de 4 bits.
- El **NCL** es el [Número de Canal Lógico](#), que es un identificador de multiplexión (posibilidad de cursar varias conexiones de nivel superior sobre una sólo conexión de nivel inferior).
- Los campos (1) y (2) indican la longitud de la dirección del llamante y del llamado respectivamente. Las **longitudes** se codifican con cuatro bits que indican el número total de dígitos de las [direcciones en X.25](#)
- Los puntos de acceso al servicio se identifican mediante el plano de numeración que asigna direcciones únicas a los usuarios de la red. Las direcciones se codifican en los dos campos de direcciones. Se obliga a que el campo de direcciones esté alineado a octeto. Si las direcciones de llamante y llamado no tienen ambas un número impar de dígitos o un número par (de España a España por ejemplo, el semiocteto que sobra con una se cubre con el de la otra (9+9=18 par), puede sobrar algún semiocteto. Los campos de direcciones son de longitud variable.
- El **padding** se usa precisamente para solventar este problema de semi-octetos sobrantes. Es un relleno de 4 ceros que se pone para que no sobre nada.
- **Campo de longitud de facilidades:** indica la longitud del campo de facilidades. Es obligatorio ya que indica si el siguiente campo está presente o no
- **Campo de facilidades:** Las [facilidades](#) son servicios suplementarios al servicio básico (Ej. especificación de uso de un tamaño de paquetes de datos superiores al tamaño por defecto (128 oct.), o de un tamaño de ventana diferente a 2, etc.).
- Estos paquetes contienen **datos de usuario** del nivel superior, los cuales no llevan delante un campo de longitud ya que acaban donde acaba la trama. Este campo transporta una SDU que ha sido transmitida por el nivel superior al inmediatamente inferior. Esto contradice que el servicio sea orientado a conexión, porque permite que antes de que se establezca la conexión las entidades de nivel superior intercambien datos.

La longitud de este campo, como se ve en la figura 4.25, es menor o igual que 16 octetos. Sin embargo, si se usa la facilidad de selección rápida ([fast select](#)) pasa a ser menor o igual que 128 octetos.

Estos datos, si el tamaño es de 16 octetos, se usan para:

1. Proporcionar un mecanismo de control de acceso, ya que con la dirección del llamante sólo se identifica al *ptr*, no a los usuarios (autenticación). Es decir, se usa para intercambiar una palabra de paso, para que así el usuario tenga información adicional para

aceptar o rechazar la conexión.

2. Enviar la PDU de establecimiento de conexión del nivel superior.

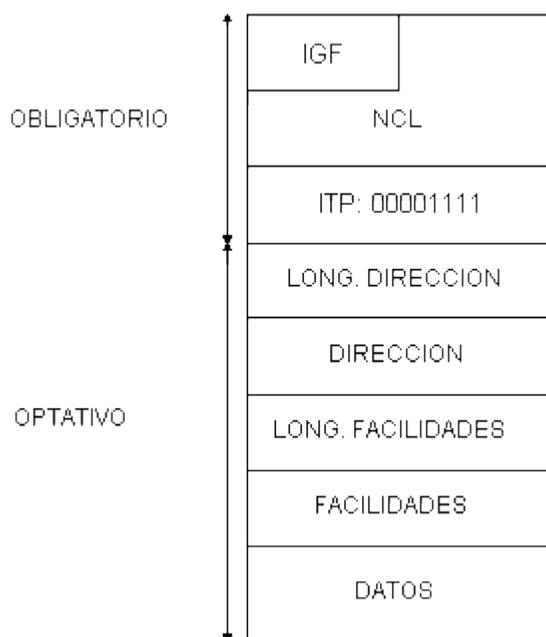
En el caso de que el tamaño del campo de datos sea de 128 octetos se usa, además de para las dos anteriores para enviar datos en modo no orientado a conexión.

#### 4.5.1.2 Paquete de llamada aceptada y de comunicación establecida.

El paquete de **llamada aceptada** va del DTE al DCE y el de **comunicación establecida** del DCE al DTE.

El formato es el mismo para los dos:

FIGURA 4.26 *Formato general de Llamada Aceptada o Comunicación Establecida.*



El campo de control (ITP) nos permite saber si el paquete es de llamada aceptada o de comunicación establecida.

Sólo son imprescindibles los tres primeros octetos. La parte opcional está presente cuando hay alguna facilidad que justifique su presencia (si ponemos los campos opcionales de **facilidades** o **datos**, tengo que poner también los de **longitud** y **direcciones**, aunque puedo dejarlos a cero). Las facilidades que justifican esta parte son:

- Selección rápida (fast select): permite un campo de datos de hasta 128 octetos. Esto permite la presencia de datos en los paquetes que se usan para aceptar o rechazar la llamada. Por tanto, si una estación nada más recibir una Petición de Llamada manda una PDU de desconexión, consigue una transferencia de datos sin haberse establecido la conexión.

Tanto estos paquetes como los de *petición de llamada* y los de *llamada entrante*, así como los de *liberación de conexión* y de *indicación de liberación* admiten la facilidad de **selección rápida**. X.25 facilita esta opción ya que hay aplicaciones que funcionan mejor en modo datagrama (servicio no orientado a conexión) (que es lo que permite esta utilidad).

#### 4.5.2. INTERCAMBIO DE DATOS.

Una vez que hemos establecido la conexión ya estamos preparados para el intercambio de datos. Los paquetes que se usan en este caso son:

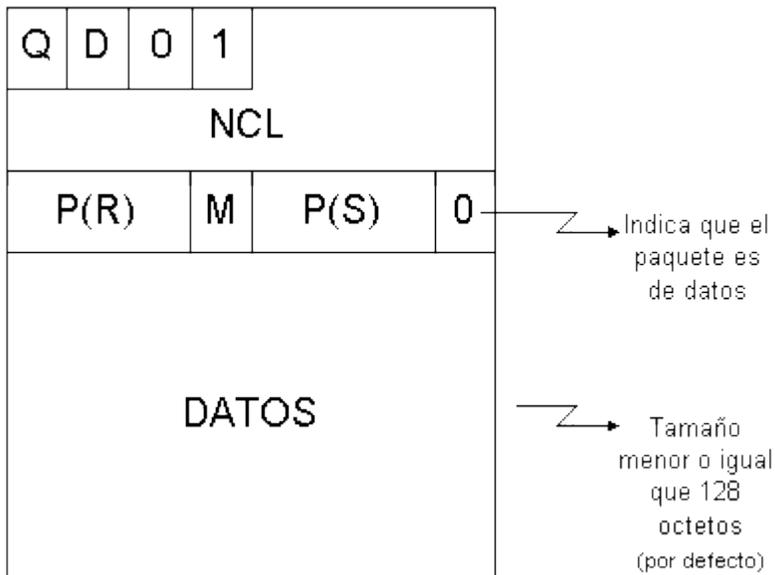
- [Paquetes de datos.](#)
- [Paquetes RR.](#)
- [Paquetes RNR.](#)

Estos dos últimos paquetes son paquetes de confirmación o también llamados paquetes de supervisión (asentimiento) que se usan para control de flujo.

##### 4.5.2.1 Paquete de datos.

El paquete de **datos** en la recomendación X.25 presenta el siguiente formato:

**FIGURA 4.31** *Formato general de un Paquete de Datos.*



Esto es el formato normal. Existe además un formato extendido, cuyo aspecto es el de la **figura 4.32**. Este formato usa dos octetos para llevar la información de control. Se sabe en que formato estamos por los bits 6 y 5 del primer octeto del paquete. Si estos bits son 10 el formato es extendido y si son 01 el formato es normal. En una conexión todos los paquetes de datos tienen el mismo formato. El formato extendido en general se usa poco.

- **Campo de datos:** En este campo van los datos del usuario. Consiste en una secuencia de octetos (al menos uno, el paquete de datos no puede ir vacío). Hay un número máximo de octetos por paquete: 128 octetos. Esta longitud máxima es la longitud por defecto. Al igual que la ventana, se puede solicitar un aumento del tamaño del paquete, pero esto también implica mayor ocupación de la red y mayor precio.
- El **campo de control** está dividido en subcampos. Estos subcampos no pueden tener valores arbitrarios. De los 8 bits que componen el campo de control se usan algunos para distinguir

unos paquetes de otros y otros para llevar información de control. Se usa un sólo bit para identificar al paquete de datos, que es un 0 en el primer bit (el de más a la derecha). El resto de los paquetes tienen un 1 en esta posición. Los otros subcampos son:

- **P(R)**: Número de secuencia de recepción. Es también un asentimiento ([piggybacking](#))
- **P(S)**: Número de secuencia de transmisión. Los números de secuencia ( $P(R)$  y  $P(S)$  o también  $N(R)$  y  $N(S)$ ) y la ventana se utilizan exclusivamente para control de flujo y detección de errores. Un número de secuencia es un identificador secuencial cíclico que se asigna a las PDUs transmitidas ( $P(S)$ ) en una conexión dada. Este número de secuencia se va incrementando con cada PDU que se envía. El número de secuencia se utiliza generalmente para:
  - Realizar control de errores.
  - Realizar control de flujo.

En X.25 esto se realiza a nivel de enlace. A nivel de red los números de secuencia sólo se utilizan para control de flujo. Esto se realiza en combinación con el mecanismo de ventana.

El paquete de datos en formato extendido tiene números de secuencia más grandes, que permiten ventanas más grandes. El tamaño de la ventana por defecto en X.25 es 2. Se puede solicitar un aumento de ventana, pero será más caro, ya que se utilizan más recursos de la red. Este incremento viene limitado por el número de secuencia.

- **bit M**: se utiliza para implementar la función de segmentación y reensamblado, que consiste en cursar datos de una SDU utilizando varias PDUs (entregándose la SDU íntegramente en destino). En origen se segmenta la SDU y en destino se reensambla. La segmentación y el reensamblado se hacen a nivel de red no de enlace.
  - Si  $M=1$  faltan más paquetes por llegar de la SDU que se está transmitiendo.
  - Si  $M=0$  no faltan más paquetes por llegar de la SDU que se está transmitiendo.
- **Campo NCL**: Es el número de canal lógico. Nos va a permitir distinguir los datos de las distintas conexiones que podemos establecer en X.25. El paquete de datos no necesita campo para la dirección de destino, ya que el servicio es orientado a conexión y cuando se establece una conexión hay una asociación lógica entre los niveles de red de los dos extremos. El campo NCL aparece para poder usar multiplexión (cursar varias conexiones de nivel superior sobre una única conexión de nivel inferior). En X.25 se permite que sobre la única conexión que nos ofrece el nivel de enlace se puedan establecer tantas conexiones como desee el nivel de red. Estas conexiones pueden que tengan el mismo destino o un destino distinto (podemos tener tantas conexiones en paralelo como queramos). El NCL es un identificador de multiplexión, que es un número que va en el campo de control y que en principio no tiene ninguna estructura y que nos permite saber qué datos pertenecen a cada conexión. Por tanto es obligatorio que aparezca. La longitud de este campo es de 12 bits, por lo que el número máximo de conexiones que podemos establecer es  $2^{12}$ . El NCL se asigna dinámicamente (a cada conexión se le asigna dinámicamente un valor para el NCL). La asignación se realiza en la fase de establecimiento de conexión. Las entidades que se conectan usan un procedimiento para negociar un número para el identificador. La entidad que pide la llamada selecciona un NCL que esté libre (hay tablas de

conexión en los sistemas que tienen registradas las conexiones establecidas). El paquete de petición de llamada se envía a través del nivel de enlace al nodo local, que lo recibe a nivel de red. El NCL que le propone la entidad que pide la llamada lo incluye en sus tablas como una conexión ocupada. Esta información llega hasta el nodo local del abonado remoto (por procedimientos internos de la red que desconocemos). El nivel de red del nodo remoto de la red genera el paquete de llamada entrante. El NCL que genera es distinto al usado en el origen. Para asignar el NCL se usa el mismo procedimiento que en origen, por lo que el NCL que esté libre para ese usuario lo añade a la tabla, lo codifica en el paquete de llamada entrante y lo entrega al abonado, que apunta en sus tablas que tenemos un nuevo NCL correspondiente a una conexión remota. Si este procedimiento tiene éxito cuando se mande el paquete de llamada aceptada, éste irá con el mismo NCL que el de llamada entrante. Por eso en el paquete de llamada aceptada la dirección de destino es opcional, puede ser suficiente con el NCL. El usar la dirección de destino puede resultar ambiguo, pero el uso del NCL nunca es ambiguo.

Los NCLs están limitados por contratación. Aunque desde el punto de vista técnico pueden haber hasta  $2^{12}$ , por motivos comerciales se limita los NCLs que pueden estar útiles. Se limitan para que el operador dimensione la red. Se paga por cada NCL que contratemos. El límite a la cantidad de recursos de red que puede usar el usuario es la capacidad del canal. Se establecen rangos en todos los posibles valores de NCL.

El NCL número 0 está reservado para procedimientos de control. Los números 1 a X están reservados para los circuitos virtuales permanentes. Luego hay otros reservados para las llamadas entrantes salientes o ambas, que se determinan por contratación.

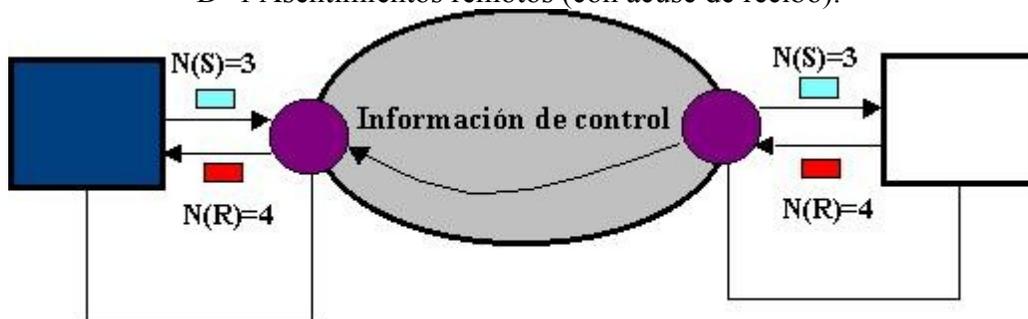
Hay dos modalidades de conexión en X.25:

- CVP: circuitos virtuales permanentes.
- CVC: circuitos virtuales conmutados.

Los CVP no se establecen ni se liberan usando los paquetes de petición y liberación de llamada, sino por contratación. Se preconfiguran los nodos de tal forma que, por contratación, el circuito está permanentemente establecido.

Los CVC son los que para establecerse y liberarse necesitan del intercambio de paquetes de establecimiento y liberación.

- **bit Q**: no afecta al comportamiento de X.25. La entidad de nivel de red simplemente informa al usuario del nivel superior de su estado a 0 o a 1. El bit Q, a nivel X.25, se envía de forma transparente. Se puede utilizar para que los protocolos de nivel superior marquen a sus paquetes de control (Q=1) o datos (Q=0). En destino se procesarán de distinta forma y tendrán un tratamiento preferente a los paquetes de datos.
- **bit D**: se utiliza para controlar el tipo de asentimiento (también llamado *acuse de recibo*)
  - D=0 Asentimientos locales (sin acuse de recibo).
  - D=1 Asentimientos remotos (con acuse de recibo).



El paquete que contiene  $N(R) = 4$  (asentimiento) se puede mandar cuando llega el paquete al nodo local o cuando llega el paquete al abonado remoto y se produce información de control.

Si se manda cuando el paquete llega al nodo local tiene la ventaja de la rapidez.

Si se manda cuando el paquete ha sido asentido en destino tenemos la ventaja de estar seguros de que el receptor ha recibido los datos. La manera en que el usuario solicita confirmación de entrega es mediante:

- Parámetro de la primitiva de Datos para solicitar confirmación de entrega.
- Primitiva específica: la que se da al usuario para confirmar.

La configuración se realiza mediante una primitiva específica en X.25. Este servicio se implementa usando el bit D:

- $D=1$ : solicita confirmación de entrega, por lo que la red asiente los paquetes sólo después de que se hallan asentido en destino.
- $D=0$ : no solicita confirmación de entrega. Los asentimientos se hacen desde el nodo local.

El inconveniente de tener  $D=1$  es que la ventana sirve para el control de flujo. Podemos transmitir hasta que se acabe la ventana sin que nos llegue asentimiento. Si el tiempo de asentimiento es pequeño hay envío continuo, por lo que la ventana no limita la transmisión. Si el tiempo de asentimiento es elevado puede que se termine la ventana, por lo que el circuito virtual del emisor no puede transmitir. Al usar el bit D el tiempo de asentimiento es más grande. Esto es indeseable. Queremos envío continuo. Si queremos acuse de recibo este problema debe tolerarse.

**FIGURA 4.32** *Formato general extendido de un Paquete de Datos.*

Q	D	1	0	
NCL				
P(S)				0
P(R)				M
DATOS				

#### 4.5.2.2 Paquetes de supervisión.

Existen dos paquetes de supervisión:

- RR
- RNR

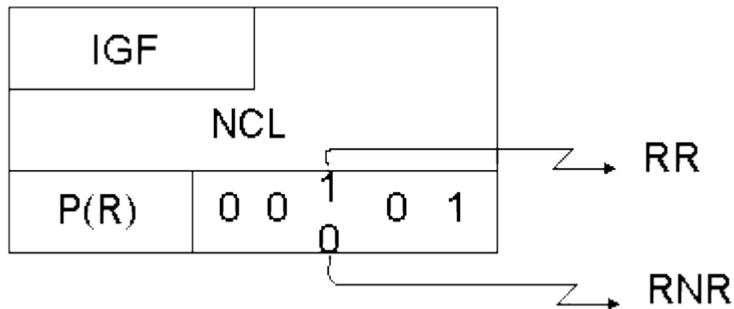
Hay un tercero que es opcional (REJ)

La función de estos paquetes es:

- Realizar control de flujo XON/XOFF.
- Realizar asentimientos explícitos.

El formato de los paquetes RR y RNR es el mismo, diferenciándose entre ellos solamente en un bit:

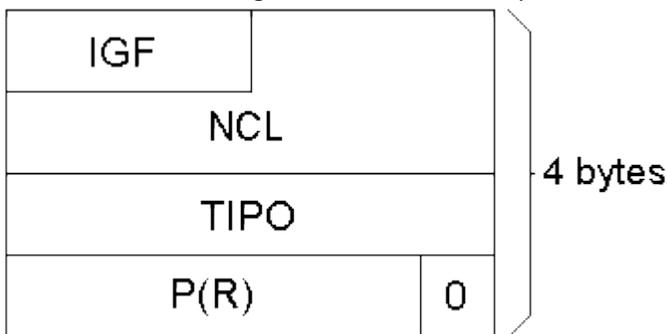
**FIGURA 4.29** Formato general de Paquetes de asentimiento



**P(R)** son 3 bits que indican el número de secuencia que se asiente (indica cuál es el siguiente que espera recibir)

Los paquetes RR y RNR varían ligeramente en el formato extendido:

**FIGURA 4.30** Formato general extendido de RR y RNR.



Los asentimientos explícitos se usarán cuando no disponemos de datos para enviar al receptor, puesto que no es posible enviar paquetes de datos vacíos para representar asentimiento.

La diferencia entre RR y RNR para su uso en control de flujo XON/XOFF (para al emisor o le deja transmitir) es:

- Si envío RR el emisor puede recibir.
- Si envío RNR la entidad que lo emite informa de que no está preparada.

Desde el punto de vista del receptor del paquete:

- Si se recibe RNR para de transmitir (sólo en el canal lógico por el que se recibe el paquete).
- Si se recibe RR puede seguir transmitiendo

NOTA: Existe una opción adicional en X.25, que prácticamente no se usa y que incluye rechazos con retransmisiones (**REJ** se consigue con el campo de tipo a 01001)

### 4.5.3. INTERCAMBIO DE DATOS ACELERADOS.

El servicio de datos acelerados o datos fuera de banda consiste en datos asociados a una conexión pero que no guardan la secuencia ni el control de flujo de los datos normales. El objetivo es que lleguen

lo más rápidamente posible (son enviabiles incluso cuando la transmisión está parada).

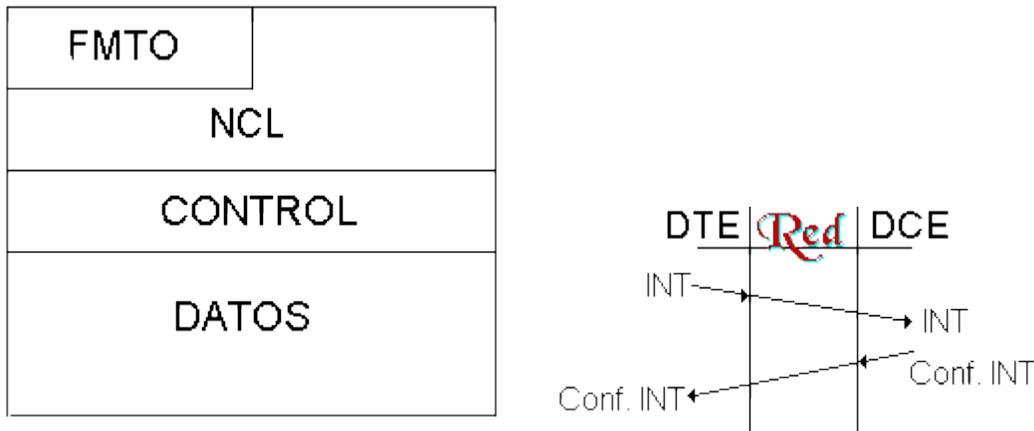
En X.25 este servicio se implementa mediante el paquete de interrupción. Hay dos paquetes de interrupción:

- uno enviado por parte del ETD.
- otro enviado por parte del ETCD.

En cuanto a formato estos paquetes son iguales. Sólo se diferencian en el sentido de la transmisión.

Tanto la **interrupción** por ETCD como por ETD presentan el siguiente formato:

**FIGURA 4.39** *Formato general del Paquete de Interrupción y protocolo de transmisión.*



Funciona en *parada y espera*: hasta que no llega un *Conf.INT*, no puedo enviar otro *INT*.

El campo **DATOS** es variable. Históricamente el tamaño de los datos era de un octeto. Las versiones más modernas permiten hasta 32 octetos (en cualquier caso, es una cantidad pequeña, pues ya estaría enviando más de lo que se debe).

El servicio de datos acelerados se usa para control de flujo de la entidad de nivel superior, por lo que no afecta al flujo de datos normales. Es como si tuviéramos 2 flujos de transmisión perfectamente distinguibles.

X.25 es un protocolo para teleproceso, por lo que tiene conectado un terminal remoto. En el caso de que el terminal se cuelgue, existe un carácter de interrupción para interrumpir el proceso. Este caracter se envía en el paquete de interrupción.

Hay dos paquetes más asociados a éste:

- Confirmación de interrupción por parte del ETD.
- Confirmación de interrupción por parte del ETCD.

Estos paquetes se utilizan porque el servicio de interrupción es un servicio confirmado.



El formato de estos paquetes es:

X	
NCL	
00011011	

Existe una restricción, hay que esperar confirmación antes de enviar otra interrupción. Es decir, sólo puede haber un paquete de interrupción pendiente de confirmación en cada canal lógico. El efecto práctico de esta restricción es que el volumen de datos que podemos enviar con formato de interrupción es pequeño.

#### 4.5.4. REINICIO Y REARRANQUE DE CONEXIONES.

##### 4.5.4.1 Reinicio.

El servicio de reinicio puede realizarse a iniciativa del proveedor o a iniciativa del usuario.

Este servicio consiste en situar la conexión en el estado inicial, por lo que los temporizadores asociados a la conexión se paran, los números de secuencia asociados a la conexión se ponen al valor inicial y los buffers asociados a la conexión se vacían.

El resultado final es que la conexión vuelve al estado inicial.

El servicio de reinicio potencialmente destruye datos, por lo que se contradice que X.25 sea fiable. Por ello este procedimiento es indeseable.

Pero este servicio es necesario cuando:

- Se producen errores no recuperables por otro procedimiento. Estos errores son:
  - Caída de un nodo de tránsito, por lo que se pierde la información de estado asociada a esa conexión. En este caso la propia red ejecuta el procedimiento de reinicio.
  - Fallo detectado por el nivel de red (errores de protocolo), por ejemplo recibir un paquete de datos vacío, de mayor longitud que la permitida...

Este servicio se implementa mediante el intercambio de paquetes de reinicio. Es un servicio confirmado. Se usan 4 paquetes:

- [Petición de reinicio \(del DTE al DCE\).](#)
- [Indicación de reinicio \(del DCE al DTE\).](#)
- [Confirmación de reinicio por ETD \(del DTE al DCE\).](#)
- [Confirmación de reinicio por ETCD \(del DCE al DTE\).](#)

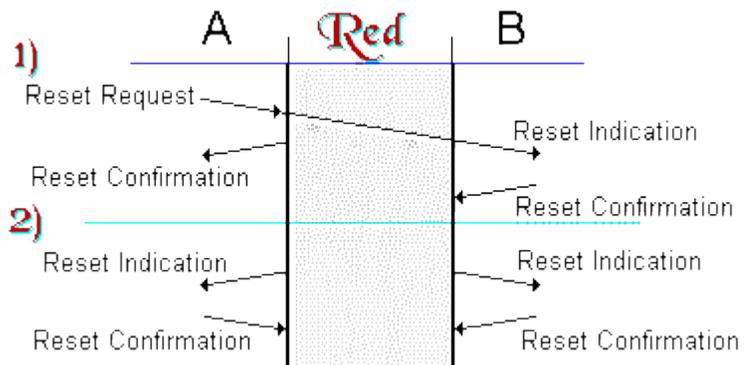
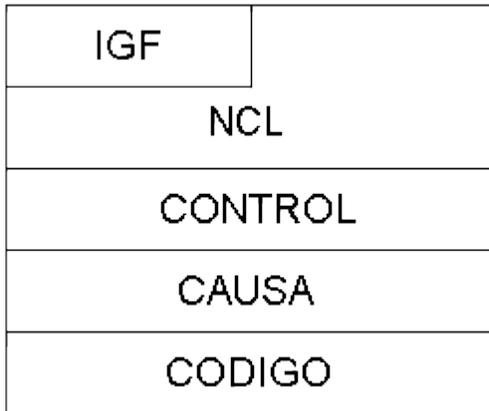
Cuando un DTE envía petición de reinicio espera a recibir confirmación de reinicio por parte del DCE, y entonces considera la conexión reestablecida.

Cuando es la red o el abonado remoto se envía un paquete de indicación de reinicio y se espera recibir confirmación de reinicio por parte de DTE y entonces considera la conexión reestablecida.

En realidad es el nivel de red el que se reinicia. El reinicio se notifica también al nivel superior. El nivel de red manda la primitiva reset indication al nivel superior, ya que el reinicio destruye los datos y el nivel superior considera que el servicio es fiable y no recupera errores. La red debe avisar al nivel superior. Aún así el nivel superior debe estar preparado para tomar una medida correctiva.

Los paquetes de **petición e indicación de reinicio** tienen el mismo formato:

**FIGURA 4.34** *Formato general de un Paquete de Petición de Reinicio y esquema del protocolo(RESET).*

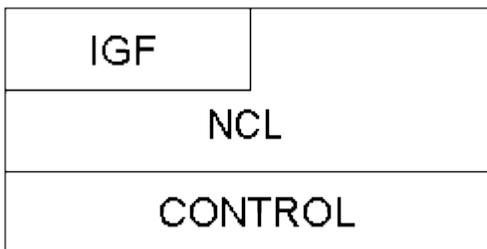


El contenido del campo de control es 00011011.

El contenido de los campos **causa** y **código** viene especificado en la recomendación (están tabulados). Estos campos son opcionales.

Los paquetes de **confirmación de reinicio por ETD y ETCD** tienen el siguiente formato:

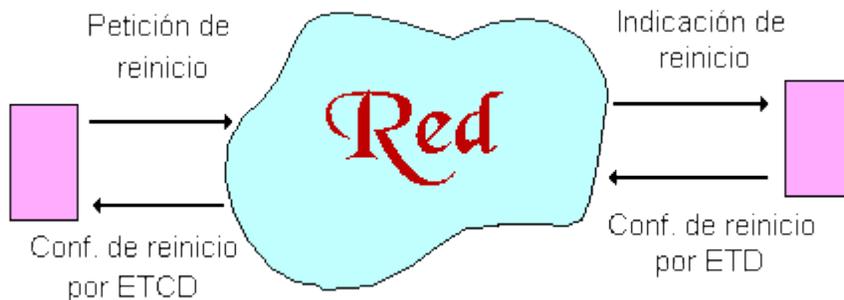
**FIGURA 4.35** *Formato general de un Paquete de Confirmación de Reinicio.*



El campo de control tiene el siguiente contenido: 00011111.

La única diferencia entre ellos es el lugar donde se generan:

**FIGURA 4.36** *Gráfico de la generación de paquetes de reinicio.*



Una situación excepcional es que los dos interlocutores soliciten reinicio a la vez. En este caso el servicio se reestablece cuando confirma ETD.

#### 4.5.4.2 Rearranque.

Es un procedimiento de recuperación frente a errores particularmente graves que afectan a toda la interfaz entre el usuario y la red (ej. se cae el equipo de usuario o el nodo local).

Usa para su implementación 4 paquetes:

- [Petición de rearranque.](#)
- [Indicación de rearranque.](#)

- [Confirmación de re arranque por parte de ETD.](#)
- [Confirmación de re arranque por parte de ETCD.](#)

Estos paquetes tienen la particularidad de que se cursan para el NCL =0. La razón es que el procedimiento de re arranque no sólo afecta a una conexión sino a todas las de la interfaz usuario red. El procedimiento de re arranque por tanto es local a una interfaz.

Se usa cuando se inicializa el sistema o cuando se detecta un fallo irre recuperable por otros medios. Ej: datos que se reciben por un NCL que no está definido.

Al inicializarse se ejecuta el procedimiento de re arranque para sincronizar el sistema con la red y que estén en el mismo estado inicial. Este estado inicial consiste en que no halla ningún circuito conmutado y que los circuitos virtuales estén en el estado inicial permanentes.

Si el procedimiento de re arranque es iniciado por el DTE, éste genera una PDU de petición d re arranque que es cursada por el NCL 0 hasta el nodo local y éste le contesta mediante un paquete de confirmación de re arranque.

Al producirse el re arranque hay que re configurar todos los elementos de la conexión. Las acciones que se toman son:

1. Reinicio de la conexión a nivel de enlace (previa). Esto se hace antes de enviar el paquete de petición de re arranque mediante una primitiva que manda el nivel de red al de enlace. Por tanto, llevamos la conexión a nivel de enlace al estado inicial.
2. Una vez hecho el reinicio se envía el paquete (o se recibe si es el otro extremo). Se recibe el paquete de re arranque, se notifica al nivel superior el reinicio de todos y cada uno de los circuitos virtuales permanentes de la red y hay indicación de liberació a todos y cada uno de los circuitos virtuales conmutados. Se entiende que en los CVC la conexión ha sido liberada y en los CVP la conexión ha sido reiniciada (se encuentra en el estado inicial). Esto se hace mediante primitivas

El formato de los paquetes de **petición e indicación de re arranque** (o reinicialización) es:

**FIGURA 4.37** *Formato general del Paquete de Reinicialización.*

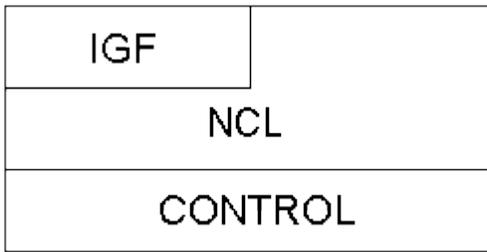
IGF	
NCL	
CONTROL	
CAUSA	
CODIGO	

Se envía por el canal lógico 0, que es el que se reserva para señalización. Los **efectos** de RESTART son:

- Todos los CV permanentes se reinician.
- Todos los CV conmutados se desconectan.

Los paquetes de **confirmación de re arranque por parte del ETD y ETCD** son también iguales a los de confirmación de RESET:

**FIGURA 4.38** *Formato general de Confirmación de Re arranque.*



#### 4.5.5. LIBERACIÓN DE CONEXIONES.

Se usan 4 paquetes:

- [Petición de liberación \(de ETD a ETCD\).](#)
- [Indicación de liberación \(de ETCD a ETD\).](#)
- [Confirmación de liberación por parte del ETD \(de ETD a ETCD\).](#)
- [Confirmación de liberación por parte del ETCD \(de ETCD a ETD\).](#)

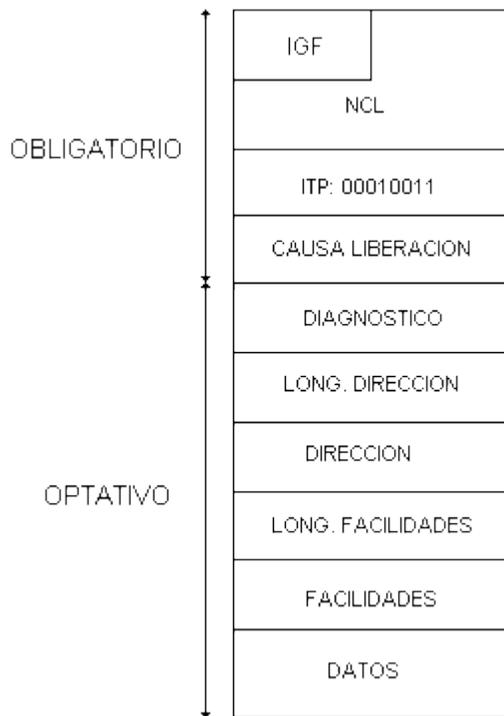
##### 4.5.5.1 Petición/Indicación de liberación.

El paquete de **petición de liberación de conexión** va del DTE al DCE y el de **indicación de liberación** del DCE al DTE.

Por lo demás ambos paquetes tienen el mismo formato:

- **Causa:** Es un byte que indica por qué se ha liberado la comunicación (si ha sido el DTE, la red, etc.).
- **Diagnóstico:** Es un byte que no hace sino refinar la causa. Es opcional en el de petición de liberación si los siguientes campos no existen. En el paquete de indicación es obligatorio si la parte opcional está presente, de lo contrario la decodificación del paquete sería ambigua..
- La **parte optativa** se envía en base a las reglas del protocolo: no cuando el usuario lo desee, sino cuando lo exige el protocolo. Es decir, si hay una facilidad que lo solicite, como por ejemplo selección rápida, que permite que los paquetes de petición de liberación puedan llevar datos y por tanto se puedan intercambiar datos.
- El campo de datos puede tener hasta 128 octetos.

**FIGURA 4.27** *Formato general de Liberación de Conexión e Indicación de Liberación.*



Si la respuesta a una paquete de llamada entrante es uno de petición de liberación se rechaza la conexión.

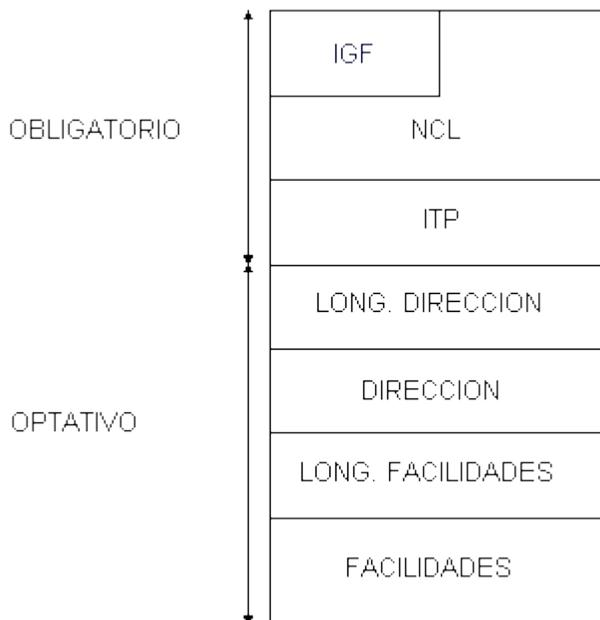
#### 4.5.5.2. Confirmación de liberación por parte de ETD/ETCD.

No hay campo de datos en este caso, ya que cuando se envía este paquete se cierra la conexión.

Su formato es el siguiente:

La parte optativa sólo está presente si una facilidad requiere su presencia.

**FIGURA 4.28** Formato general de Confirmación de Liberación



Hay dos formas de realizar la liberación de las conexiones:

- **Liberación abrupta:** liberación unilateral por parte de una de las entidades que participan en la conexión, por lo que hay una potencial pérdida de datos entrantes.
- **Liberación ordenada:** ambas entidades, y las entidades de red deben estar de acuerdo para realizar la liberación, por lo que no hay pérdida de datos entrantes.

En X.25 se usa liberación abrupta. Cualquiera de las dos entidades puede decidir el final de la conexión en cualquier momento unilateralmente.

### ***Procedimiento de liberación.***

El nivel 4 de la entidad que desea la liberación pasa al nivel 3 la primitiva disconnection request. El nivel 3 genera la PDU de petición de liberación. El nivel 3 se la pasa al nivel 2 mediante la primitiva data request. El nivel 2 construye la trama en la que mete la SDU que le ha pasado el nivel 3. La trama se pasa al nivel 1 convertida en binario y del nivel 1 se pasa al medio físico convertida en señal. De aquí llega al DCE. Se pasa del medio físico al nivel 1 de éste al nivel 2 y aquí se decodifica y se pasa al nivel 3 mediante la primitiva data indication. Mediante señalización interna de red se indica que se quiere liberar la conexión y mediante primitivas se realiza el proceso inverso.

En el momento en el que el nodo local recibe la petición de liberación puede generar la PDU de confirmación de liberación, que es entregada al nivel 2 mediante Data Request. No se necesita confirmación del abonado remoto.

La liberación es abrupta. Aún así la red notifica al abonado remoto que se libera la conexión. El nivel 3 del nodo remoto genera una PDU de indicación de liberación que llega al nivel 3 del abonado remoto, que notifica al nivel superior una primitiva de disconnection indication. El nivel 3 puede enviar espontáneamente una confirmación de liberación, aunque el nivel superior manda una primitiva de disconnection response.

Después de pedir la petición de liberación no se pueden mandar datos al nivel superior, ya que éste considera liberada la conexión y sólo espera la confirmación de liberación.

La primitiva de disconnection response no es obligatorio que sea respondida por el nivel 3 del DCE ya que se entiende que la conexión ha sido liberada.

La confirmación a nivel de protocolo es necesaria puesto que la entidad local tiene que estar segura que el nodo local ha recibido la petición de liberación, para poder liberar el NCL de esa conexión y poder usarlo para otra.

Las primitivas del DTE remoto son fúerfluas y en algunos servicios no se envían.

La desconexión puede suceder espontáneamente por parte de la red. La red manda una PDU a cada usuario de indicación de liberación. La red lo realiza en casos de errores irreuperables. Causa habitual de liberación espontánea es un error en un nodo local de abonado no recuperable por otros medios.

Aunque hay pérdida de datos, el servicio es fiable puesto que avisa que pueden haberse perdido datos. Un mecanismo para confirmar que no se han perdido datos es que las entidades de nivel superior, de mutuo acuerdo, soliciten acuse de recibo y hagan uso de paquetes de confirmación de entrega.

La diferencia entre liberación y rearranque es que la liberación sólo afecta a un canal lógico y el rearranque afecta a todos los de una misma interfaz usuario-red.

## **FORMATO DE DIRECCIONES EN X.25.**

La recomendación X.121 (se pueden usar otras opcionalmente) especifica el formato de las direcciones:

FIGURA 4.22 Formato de las direcciones.



Según X.121 las direcciones se estructuran en 2 campos:

- **DNIC** (*Data Network Identifier Code*): Identifica a cada red X.25 y distingue al operador público (Iberpac tiene uno, Transpac (Francia) otro, etc.). Es único a nivel mundial. Tiene 4 dígitos decimales.
- **NTN** (*Network Terminal Number*): Número de abonado (hasta 10 dígitos). En España está limitado a 9 dígitos.

Esta estructura obedece a una convención administrativa de la red.

Este tipo de dirección posibilita el encaminamiento jerárquico.

- **Codificación:**
  - La codificación se hace en BCD; concretamente se usa un octeto para cada 2 dígitos.
  - En algunas ocasiones, el número de dígitos es impar (España p.ej.) lo cual da lugar a medio octeto sobrante y luego veremos que probablemente habrá que usar un relleno (*padding*).
- **Utilización:**
  - Usar siempre el DNIC, incluso si llamo a mi propia red. (es como si telefonoo Madrid-Madrid y siempre pongo el prefijo 91 delante).
  - No usar el DNIC internamente (sólo NTN) y usar para llamadas externas **0+DNIC+NTN** (esto es lo que se utiliza en Iberpac).

## Apéndice I: Protocolo LAP-B.

El protocolo LAP-B (*Link Access Procedure - B*) es un protocolo HDLC 2,8, es decir, con rechazo simple, indicado por el 2, y en el cual las tramas de información pueden ser utilizadas como tramas de control, indicado esto último por el 8. Veamos, en detalle, cuál es el formato de las tramas:

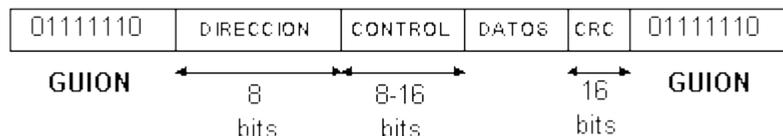
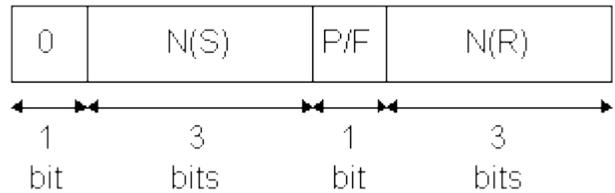


FIGURA 4.13 Trama de LAP-B.

Analizamos más a fondo los tres tipos de tramas que se manejan:

- a) **Tramas de información.**: El campo de control es:



**FIGURA 4.14** Campo de control de una trama de información.

- El primer bit es obligatoriamente un '0'.
- En segundo lugar se coloca el número de secuencia de la trama de información que se envía.
- A continuación existe un bit denominado P/F. Es el bit Poll/Final de los protocolos HDLC que no entraremos a estudiar en profundidad.
- Por último, aparece un número de secuencia de asentimiento. Se utiliza **piggybacking**, esto significa que se aprovechan las tramas de información para mandar asentimientos. Si un terminal recibe correctamente una trama y él quiere enviar otra, no genera un ACK y después manda su trama sino que incorpora el asentimiento en la propia trama. Por esto, representaremos las tramas de información con una 'I' seguida de dos números. Con I23, por ejemplo, quien lo manda envía el equivalente a lo que antes representábamos con I2 y ACK3, es decir, envía la trama 2 y advierte de que está esperando la trama 3 del otro interlocutor.

En principio por defecto se utiliza numeración modulo 7 (3 bits), así, las tramas irán con números desde el 0 hasta el 7 ambos incluidos. Si el retardo de asentimiento, tiempo que transcurre desde que se envía el último bit de una trama hasta que se recibe su asentimiento, es muy alto, puede interesar aumentar la numeración para poder mandar más tramas en dicho tiempo de asentimiento. Este es el motivo por el que se permite utilizar numeración extendida a módulo 127 (7 bits).

- b) **Tramas de supervisión.** El campo de control es:

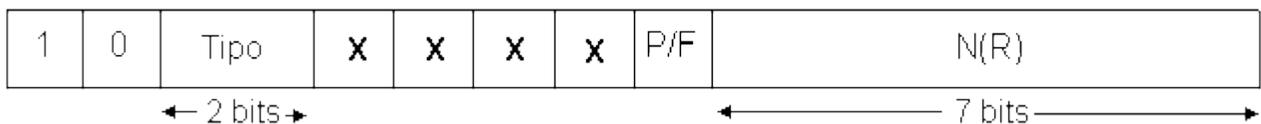
**FIGURA 4.15** Campo de control de una trama de supervisión.



supervisión.

y en el caso de numeración extendida:

**FIGURA 4.16** Campo de control de una trama de supervisión con numeración extendida.



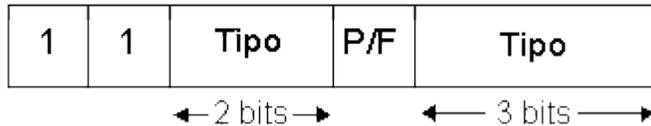
Los tipos son:

BITS	TIPO	SIGNIFICADO
00	RR (Receiver Ready)	ACK
01	REJ (Reject)	Informa de que una trama llegó mal
10	RNR (Receiver Not Ready)	Se avisa al terminal origen que el receptor se desborda. Aún con esto se confirma la última trama recibida. El origen se queda parado hasta

		recibir un RR.
11	SREJ	Se utiliza en rechazo selectivo. Por tanto, no se usa en X.25.

- b) **Tramas no numeradas.** El campo de control es:

**FIGURA 4.17** Campo de control de una trama no numerada.



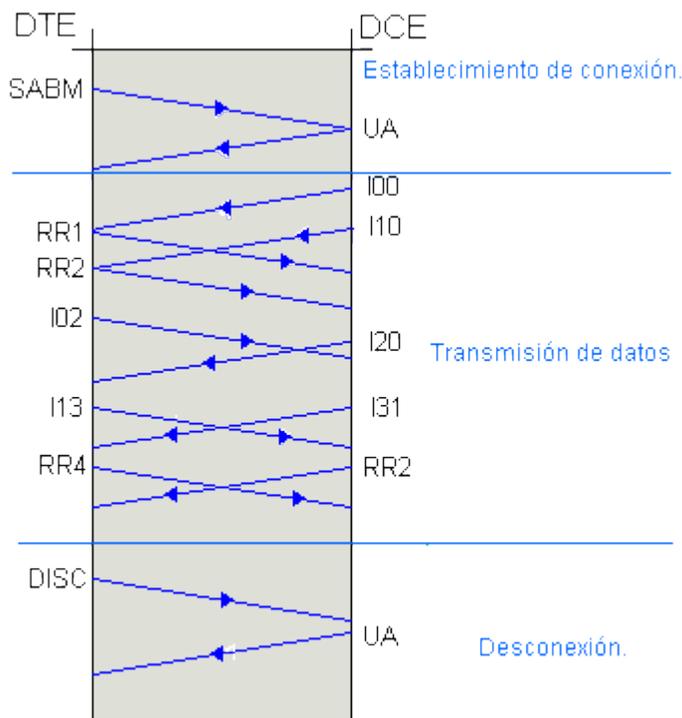
numerada.

Algunos tipos utilizados son:

- **SABM** (Set Asynchronous Balance Mode): Sirve para configurar el receptor y el emisor.
- **UA** (Unnumbered ACK): Confirma tramas no numeradas que funcionan en modo parada y espera.
- **DISC**: Se utiliza para desconectar.
- **SABME**: Se configuran emisor y receptor acordando utilizar numeración extendida.
- **RESET**: Ante situaciones irrecuperables se pone todo a cero y se informa al nivel superior de que ha habido un fallo grave.

Analicemos un ejemplo completo para fijar todo lo explicado hasta ahora:

**FIGURA 4.18** Ejemplo de comunicación de LAP-B.



Detengámonos en esta figura para entenderla a fondo. Diferenciamos en este tipo de comunicaciones tres fases:

#### Establecimiento de conexión:

En esta fase, un sistema final o DTE pide que se abra una comunicación con la trama SABM. En primer lugar, es importante señalar que el receptor será siempre un DCE puesto que trabajamos

en el Nivel de Enlace, es decir, con comunicaciones entre entidades directamente conectadas. Con la trama citada, el DTE consigue informar al DCE de qué características tendrá la comunicación que quiere establecer, en este caso por ejemplo, la numeración será la que exista por defecto y no será numeración extendida.

Una vez recibida la trama correctamente en el DCE, éste contesta con UA para confirmar que la comunicación queda abierta.

Hasta aquí, como podemos comprobar en la figura, se trabaja en modo parada y espera.

#### **Fase de transmisión de datos:**

Tras establecer la conexión y algunos de sus parámetros ya se puede pasar a mandar información. En el caso de la figura, es el DCE quien envía una trama, la trama I00. Como ya sabemos, esto quiere decir que la trama que se envía es la trama 0 y que el DCE está esperando recibir del DTE la trama 0. Tanto esta trama como la siguiente que manda el DCE, la I01, son confirmadas por el DTE con tramas RR. Como recordamos, no se asiente una trama con su número sino con el número de la trama que a partir de ese momento se espera, es decir, la siguiente a la que se confirma. Ésta es la razón por la que I00 se confirma con RR1.

La primera trama que envía el DTE es I02, es decir, en este punto él manda la trama 0 y está esperando la 2. Una vez llega ésta al DCE, éste la confirma con I31, esto es, mandando su cuarta trama e indicando que queda a la espera de la trama 1 del DTE.

En el proceso ilustrado no figura ningún error pero, de haberlo, todo funcionaría como quedó descrito en ARQ con rechazo simple. Bien porque saltase un TIMER o por la recepción de una trama REJ se obligaría a la retransmisión a partir de la trama errónea.

El proceso así descrito continuará, si no surge ningún problema irrecuperable, hasta que uno de los interlocutores pida la desconexión.

#### **Desconexión:**

Una de las entidades envía la trama DISC que es confirmada con UA. Queda así la comunicación cerrada.

Por último, estudiemos algunos parámetros que intervienen en la comunicación y que son modificables y configurables en función de las condiciones de la red. Son:

#### **T1 o Plazo de Retransmisión:**

Es el tiempo que se espera desde la transmisión de una trama hasta su retransmisión por falta de ACK. Es el objeto del TIMER del que hemos venido hablando hasta ahora.

#### **T2 o Retardo Máximo antes de Asentimiento:**

Pueden no asentirse las tramas inmediatamente según llegan. Puede esperarse un tiempo menor que este T2 por si llegan más tramas que puedan ser asentidas todas juntas.

#### **T3 o Plazo de Inactividad:**

Si transcurre un tiempo sin que se transmita o reciba nada se emite un RR asintiendo la última trama que hubiese llegado. Es necesario testear el enlace para comprobar un posible fallo grave como la caída de un nodo...

**N1 o Longitud Máxima de la Trama.**

**N2 o Número Máximo de Retransmisiones de una Trama:**

Si después de N2 retransmisiones de una trama, ésta no es asentida se resetea el enlace o se desconecta informando al nivel superior.

**K o Tamaño de Ventana.**