

# LA CAPA DE ENLACE DE DATOS

## LLC (control de enlace lógico)

Introducción.

Los protocolos utilizan ALGORITMOS para lograr una comunicación confiable y eficiente entre dos máquinas adyacentes en la capa de enlace de datos.

Por desgracia, los circuitos de comunicación cometen errores ocasionales.

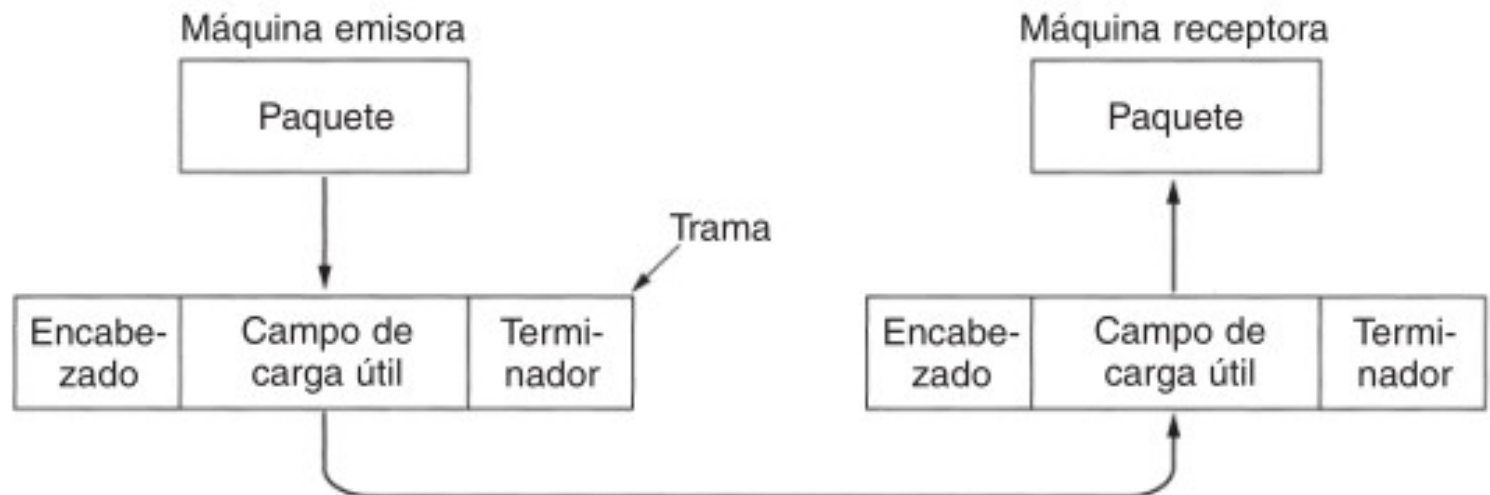
- Tasa de datos finita

- Retardo de propagación diferente de cero en la comunicación

# LA CAPA DE ENLACE DE DATOS

## CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

1. Proporcionar una **interfaz de servicio bien definida** con la capa de red.
2. Manejar los **errores de transmisión**.
3. **Regular el flujo de datos** para que receptores lentos no sean saturados por emisores rápidos.

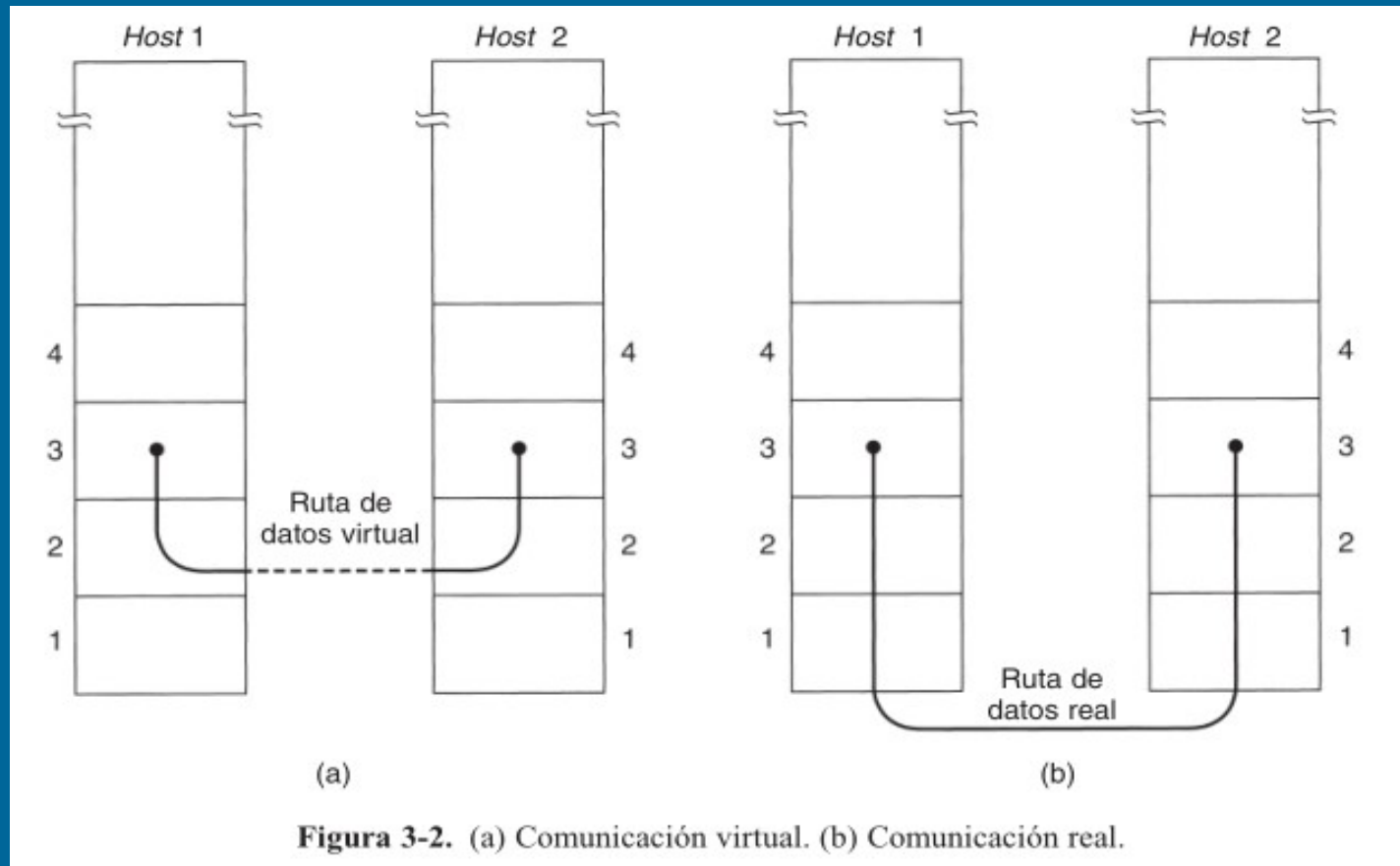


**Figura 3-1.** Relación entre los paquetes y las tramas.

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## SERVICIOS PROPORCIONADOS A LA CAPA DE RED

El servicio principal es transferir datos de la capa de red en la máquina de origen a la capa de red en la máquina de destino.



# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## SERVICIOS PROPORCIONADOS A LA CAPA DE RED

**Servicio no orientado a la conexión sin confirmación de recepción** (apropiada cuando la tasa de errores es muy baja).

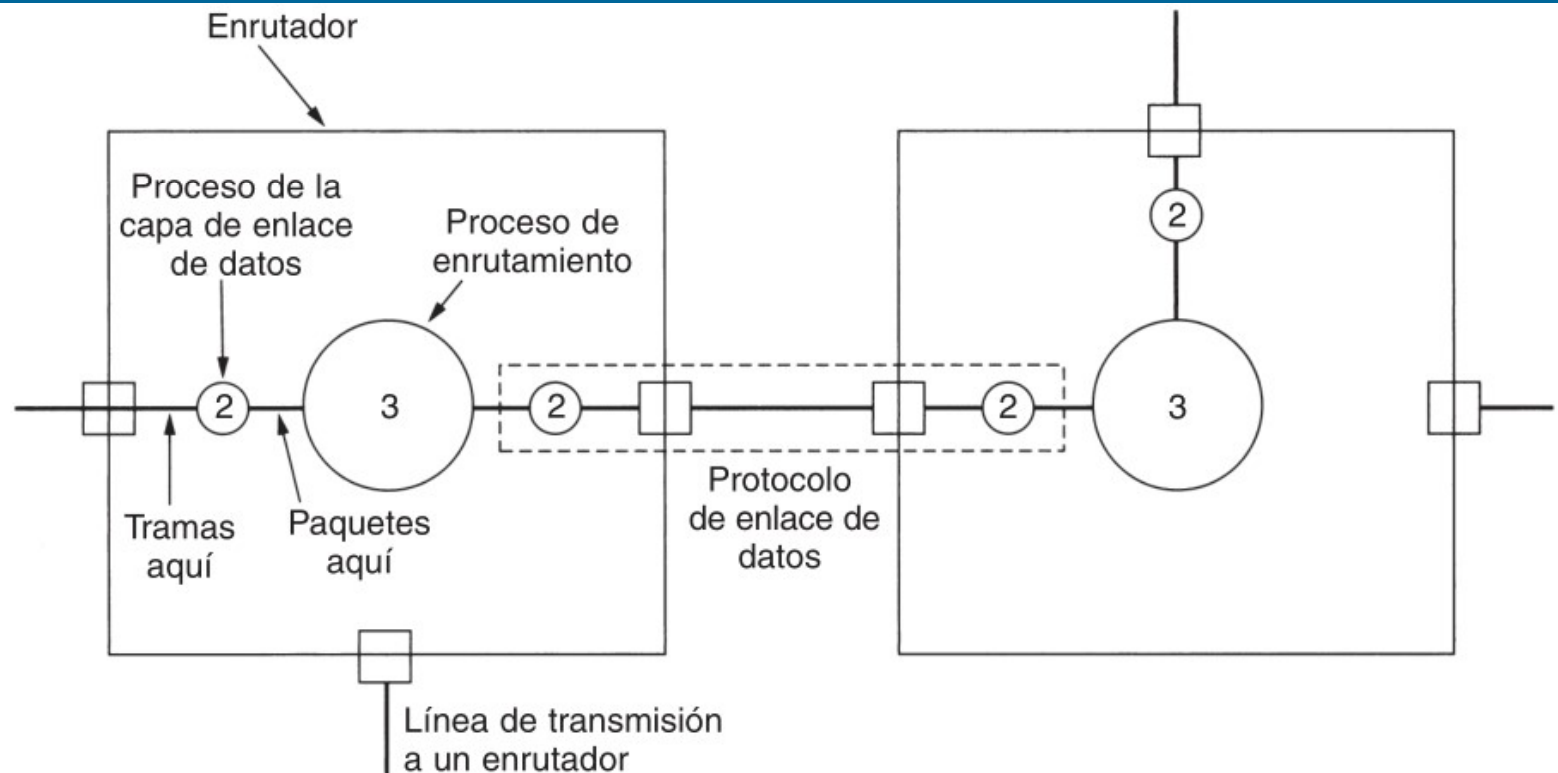
**Servicio no orientado a la conexión con confirmación de recepción** (se confirma de manera individual la recepción de cada trama enviada).

**Servicio orientado a la conexión con confirmación de recepción** (Establecen una conexión antes de transferir datos; Flujo de bits confiable).

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## SERVICIOS PROPORCIONADOS A LA CAPA DE RED

Flujo de datos a través de dos enrutadores (ejm. Sub red WAN.).



**Figura 3-3.** Ubicación del protocolo de enlace de datos.

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## ENTRAMADO

Es responsabilidad de la capa de enlace de datos detectar y, de ser necesario corregir los errores. Cuando estos se reciben desde la capa física.

Suma de Verificación de cada Trama (**CRC**)

División de tramas del flujo de bits (**Temporización** - introducción de intervalos de tiempo entre tramas-)

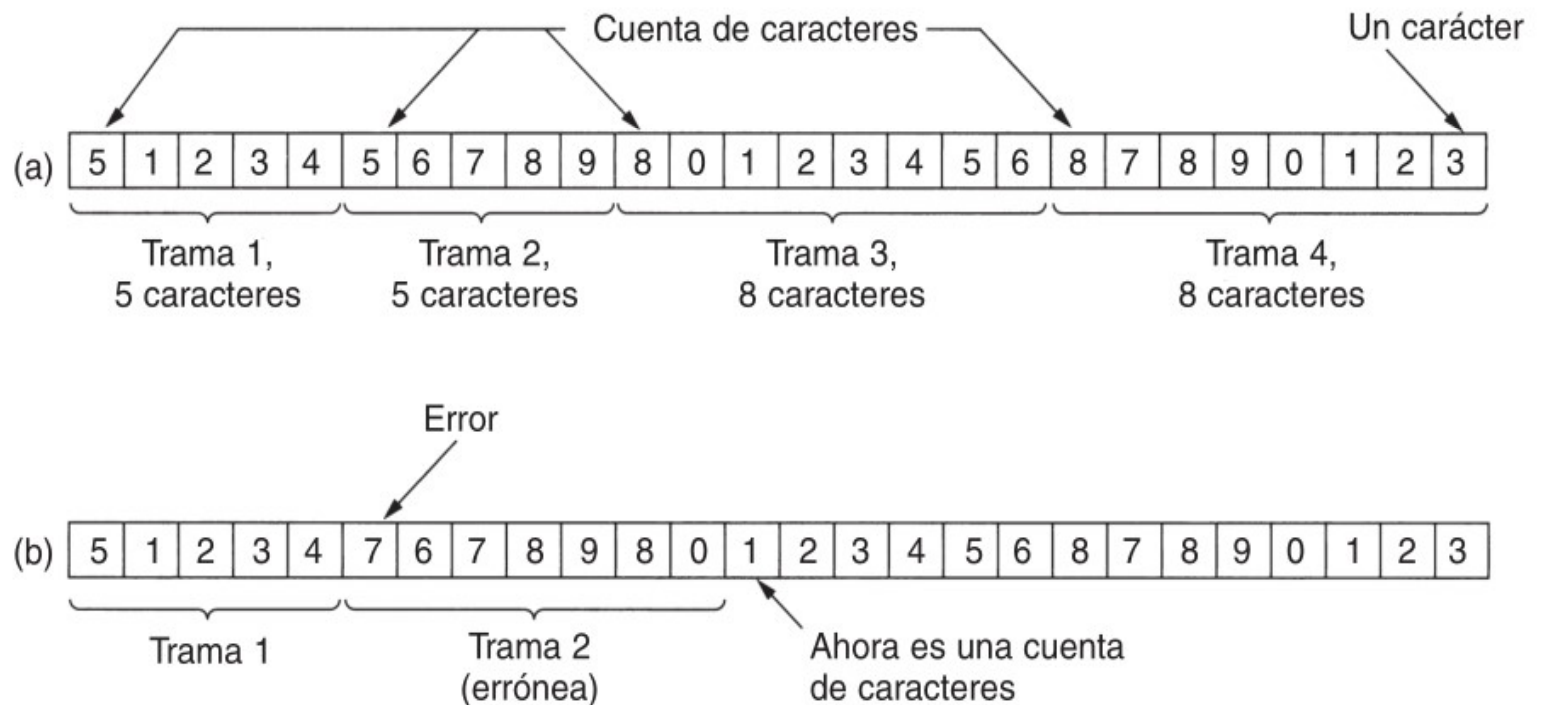
Otros métodos de entramado:

- ☐ Conteo de caracteres.
- ☐ Banderas, con relleno de caracteres.
- ☐ Banderas de inicio y fin, con relleno de bits.

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## ENTRAMADO - Primer Método Conteo de Caracteres

Se vale de un campo en el encabezado para especificar el número de caracteres en la trama.

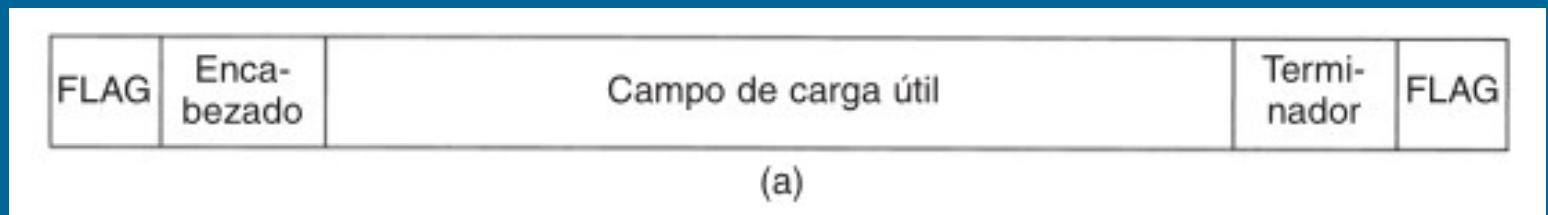


**Figura 3-4.** Un flujo de caracteres. (a) Sin errores. (b) Con un error.

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## ENTRAMADO - Segundo Método Banderas, con Relleno de Caracteres

Cada trama inicia y termina con bytes especiales (Bandera o Indicador).



Se puede dar el caso con mucha facilidad de que el **patrón de bits de la bandera aparezca en los datos** (payload).

**Relleno de Caracteres:** Inserte un byte de escape especial (**ESC**) justo antes de cada bandera “accidental” en el campo de datos (payload)

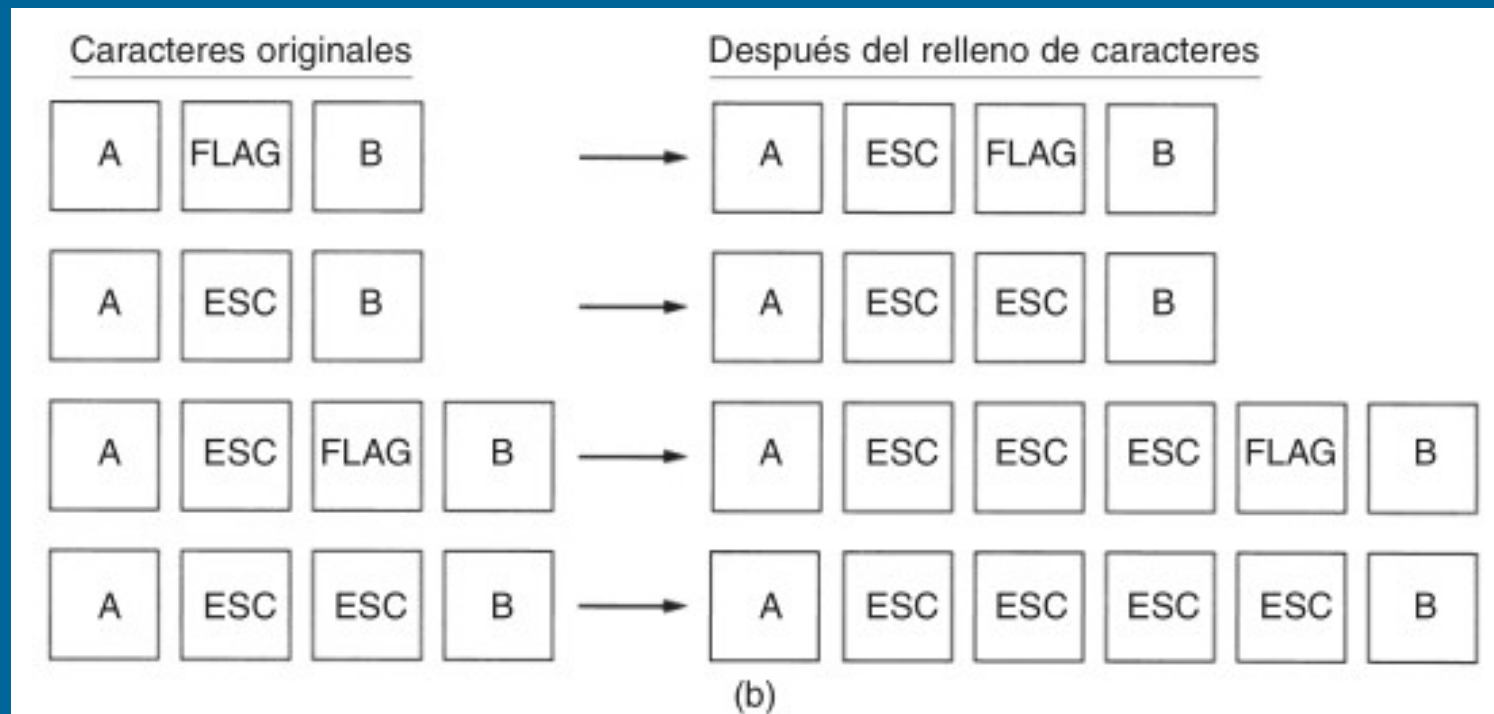


# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## ENTRAMADO - Segundo Método Banderas, con Relleno de Caracteres

El esquema de relleno de caracteres, es una ligera simplificación del esquema empleado en el protocolo PPP

-A y B datos Flag (bandera accidental) se rellena con un carácter ESC-



# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## ENTRAMADO - Tercer Método Banderas de inicio y fin, con relleno de bits

**RELLENO DE BITS:** Cada vez que la capa de enlace de datos del emisor encuentra **cinco unos consecutivos en los datos (payload)**, automáticamente **inserta un bit 0** en el flujo de bits saliente.

(a) 011011111111111111110010

(b) 01101111011111011111010010

Bits de relleno

(c) 0110111111111111111111110010

**Figura 3-6.** Relleno de bits. (a) Los datos originales. (b) Los datos, según aparecen en la línea. (c) Los datos, como se guardan en la memoria del receptor tras eliminar el relleno.

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## Control de Errores

Por lo general, el protocolo exige que el **receptor regrese tramas de control especiales** que contengan confirmaciones de recepción positivas o negativas de las tramas que llegan

El asunto de la administración de **temporizadores** y **números de secuencia** es para asegurar que cada trama llegue finalmente a la capa de red en el destino una sola vez, ni más ni menos, es una parte importante de las tareas de la capa de enlace de datos

# CUESTIONES DE DISEÑO DE LA CAPA DE ENLACE DE DATOS

## Control de Flujo

Qué hacer con un emisor que quiere transmitir tramas de manera sistemática y a mayor velocidad que aquella con que puede aceptarlos el receptor

**Control de flujo basado en retroalimentación**, el receptor regresa información al emisor autorizándolo para enviar más datos o indicándole su estado

**Control de flujo basado en tasa**, el protocolo tiene un mecanismo integrado que limita la tasa a la que el emisor puede transmitir los datos, sin recurrir a retroalimentación por parte del receptor.

# DETECCIÓN y CORRECCIÓN DE ERRORES

## Código de Corrección de Errores

*La conclusión es: los errores de transmisión van a ser inevitables durante muchos años más. Tendremos que aprender a lidiar con ellos.*

### *Código de Corrección de Errores*

**Incluir suficiente información redundante** en cada bloque de datos transmitido para que el receptor pueda deducir lo que debió ser el carácter transmitido.

# DETECCIÓN y CORRECCIÓN DE ERRORES

## Código de Corrección de Errores

En los canales que son altamente confiables es más económico utilizar un código de **detección de errores** y simplemente retransmitir los bloques defectuosos que surgen ocasionalmente

**-Corrección de errores-** En canales que causan muchos errores, como los **enlaces inalámbricos**, es mejor agregar la redundancia suficiente para que el receptor pueda descubrir cuál era el bloque original

Los códigos utilizados para la corrección de errores son los siguientes

- FEC (Forward Error Correction)
- Código Binario de Golay
- Reed-Solomon.
- Hamming

# DETECCIÓN y CORRECCIÓN DE ERRORES

## Código de Detección de Errores

Los códigos de **corrección de errores** se utilizan de manera amplia en los **enlaces inalámbricos**, que son mas ruidosos y propensos a errores que el alambre de cobre o la fibra óptica

### *Detección de Errores* - **Código Polinomial**

Este tipo de códigos se basan en el tratamiento de cadenas de bits como representaciones de polinomios con coeficiente de cero y uno solamente.

Cuando se emplea este método, el receptor y el emisor deben acordar por adelantado un **polinomio generador**.

# PROTOCOLOS ELEMENTALES DE ENLACE DE DATOS

## PROTOCOLES ELEMENTALES

**Independencia de las capas**, en las capa física y enlace de datos (procesador dentro de un chip especial de E/S)

**Procedimiento** que realiza la capa de enlace de datos para enviar una trama desde un hosts origen a uno de destino (*to\_physical\_layer* -enviar una trama-; *from\_physical\_layer* -recibir una trama-) – Primitivas de servicio

El grupo de **eventos posibles difiere** para cada uno de los **diferentes protocolos**

La capa de red nunca debe recibir ninguna parte del encabezado de una trama: para ***mantener completamente separados el protocolo de red y el de enlace de datos.***

Es importante entender la relación entre ***Paquete*** y ***Trama***, la encapsulación de estos y su independencia



# PROTOCOLOS ELEMENTALES DE ENLACE DE DATOS

## PROTOCOLES ELEMENTALES

En la mayoría de los protocolos suponemos un **canal inestable que pierde tramas completas ocasionalmente**. Para poder recuperarse de tales problemas, la capa de enlace de datos emisora debe arrancar un **temporizador o reloj interno cada vez que envía una trama**

# PROTOCOLOS ELEMENTALES DE ENLACE DE DATOS

## Un protocolo símplex sin restricciones – Protocolo 1

Como ejemplo inicial consideraremos un protocolo que es lo más sencillo posible. Los datos se transmiten sólo en una dirección; las capas de red tanto del emisor como del receptor siempre están listas; el tiempo de procesamiento puede ignorarse; hay un espacio infinito de búfer y, lo mejor de todo, el canal de comunicación entre las capas de enlace de datos nunca tiene problemas ni pierde tramas. Este protocolo completamente irreal, al que apodaremos “**utopía**”

# PROTOCOLOS ELEMENTALES DE ENLACE DE DATOS

## Protocolo símplex de parada y espera – Protocolo 2

El problema principal que debemos resolver aquí es cómo evitar que el emisor sature al receptor enviando datos a mayor velocidad de la que este último puede procesarlos.

Si los diseñadores de la red pueden calcular el comportamiento del peor de los caso del receptor, podrán programar al emisor para que transmita con tanta lentitud que, aun si cada trama sufre el retardo máximo, no haya desbordamientos

Una solución más general para este dilema es hacer que el receptor proporcione retroalimentación al emisor

**Protocolo de parada y espera** el **emisor envía una trama** y luego **espera una confirmación** de recepción antes de continuar el proceso.

Aquí sería suficiente un canal físico semidúplex.

# PROTOCOLOS ELEMENTALES DE ENLACE DE DATOS

## Protocolo símplex para un canal con ruido – Protocolo 3

Un canal de comunicación que comete errores.  
(Ruido - Interferencia)

La capa de red de A entrega el paquete 1 a su capa de enlace de datos. El paquete se recibe correctamente en B y se pasa a la capa de red de B. B regresa a A una trama de confirmación de recepción.

La trama de confirmación de recepción se pierde por completo.

El temporizador de la capa de enlace de datos de A expira en algún momento. Al no haber recibido una confirmación de recepción, supone (incorrectamente) que su trama de datos se ha perdido o dañado, y envía otra vez la trama que contiene el paquete 1.

# PROTOCOLOS ELEMENTALES DE ENLACE DE DATOS

## Protocolo símplex para un canal con ruido – Protocolo 3

La trama duplicada también llega bien a la capa de enlace de datos de  $B$  y de ahí se pasa de manera inadvertida a la capa de red. Si  $A$  está enviando un archivo a  $B$ , parte del archivo se duplicará (es decir, la copia del archivo reconstruida por  $B$  será incorrecta y el error no se habrá detectado). En otras palabras, el protocolo fallará.

La forma de evitar esto es hacer que el emisor ponga un **número de secuencia en el encabezado de cada trama que envía**. A continuación, el receptor puede examinar el número de secuencia de cada trama que llega para ver si es una trama nueva o un duplicado que debe descartarse.

# PROTOCOLO DE VENTANA CORREDIZA

## Protocolo Ventana Corrediza

La técnica superposición.

Cuando llega una trama de datos, en lugar de enviar inmediatamente una trama de control independiente, **el receptor se aguanta y espera hasta que la capa de red le pasa el siguiente paquete**. La confirmación de recepción se anexa a la trama de datos de salida (usando el campo *ack del encabezado de la trama*).

¿Cuánto tiempo debe esperar la capa de enlace de datos un paquete al cual superponer la confirmación de recepción?

La esencia de todos los protocolos de **ventana corrediza** es que, en cualquier instante, el emisor mantiene un grupo de números de secuencia que corresponde a las tramas que tiene permitido enviar. Se dice que estas tramas caen dentro de la ventana emisora. De manera semejante, el receptor mantiene una ventana receptora correspondiente al grupo de tramas que tiene permitido aceptar.

# PROTOCOLO DE VENTANA CORREDIZA

## Protocolo Ventana Corrediza

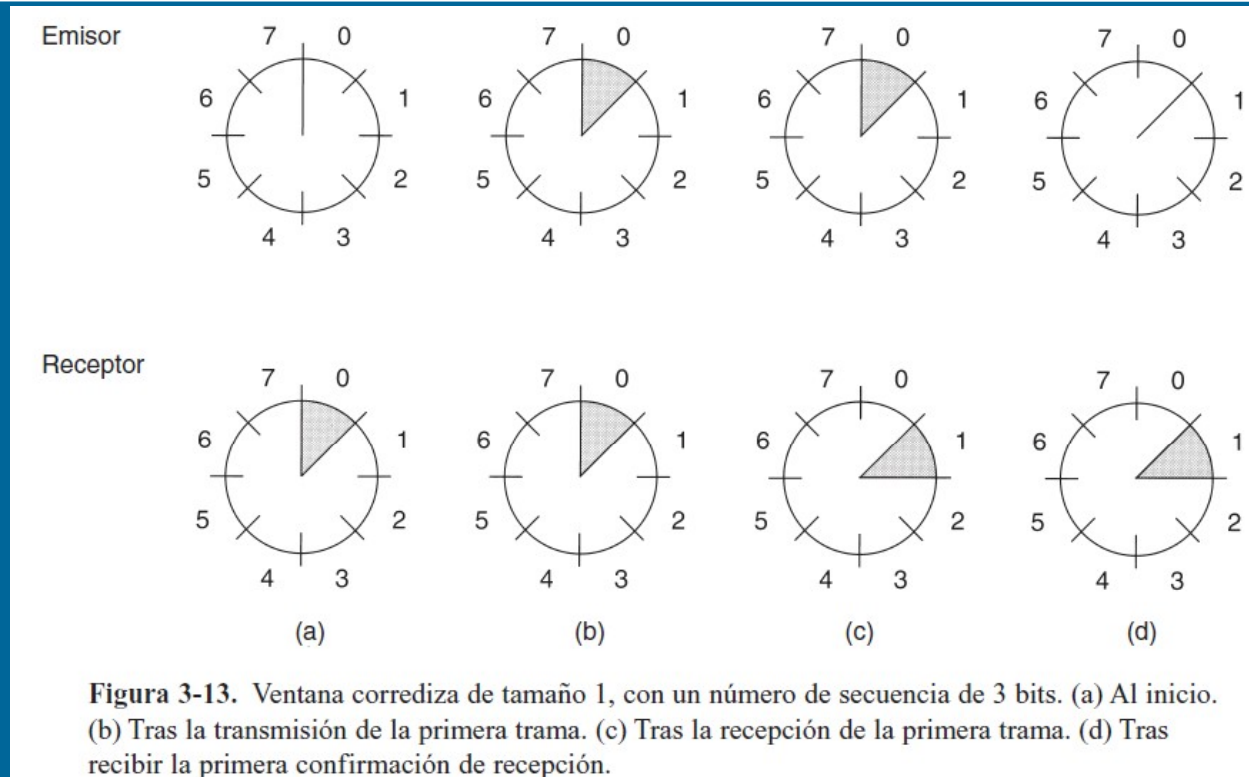


Figura 3-13. Ventana corrediza de tamaño 1, con un número de secuencia de 3 bits. (a) Al inicio. (b) Tras la transmisión de la primera trama. (c) Tras la recepción de la primera trama. (d) Tras recibir la primera confirmación de recepción.

Si el **tamaño máximo de la ventana es  $n$** , el emisor necesita  **$n$  búferes para contener las tramas sin confirmación de recepción.**

Si la ventana llega a crecer a su tamaño máximo, la capa de enlace de datos emisora deberá hacer que la capa de red se detenga hasta que se libere otro búfer.

# PROTOCOLO DE VENTANA CORREDIZA

## Protocolo de ventana corrediza de un bits

Protocolo de ventana corrediza con un tamaño máximo de ventana de 1. Tal protocolo utiliza **parada y espera**, ya que **el emisor envía una trama y espera su confirmación de recepción** antes de transmitir la siguiente.

El campo de confirmación de recepción contiene el número de la última trama recibida sin error. Si este **número concuerda con el de secuencia** de la trama que está tratando de enviar el emisor, éste sabe que ha terminado con la trama almacenada en el búfer y que puede obtener el siguiente paquete de su capa de red.

Si el número de **secuencia no concuerda**, debe continuar intentando enviar la misma trama. Por cada trama que se recibe, se regresa una.



# PROTOCOLO DE VENTANA CORREDIZA

## Protocolo de ventana que usa retroceso N

Hasta ahora hemos supuesto que el tiempo de transmisión requerido para que una trama llegue al receptor más el necesario para que la confirmación de recepción regrese es insignificante. A veces esta suposición es totalmente falsa. En estas situaciones el **tiempo de viaje de ida y vuelta prolongado puede tener implicaciones importantes para la eficiencia del aprovechamiento del ancho de banda**

### Técnica se conoce como canalización

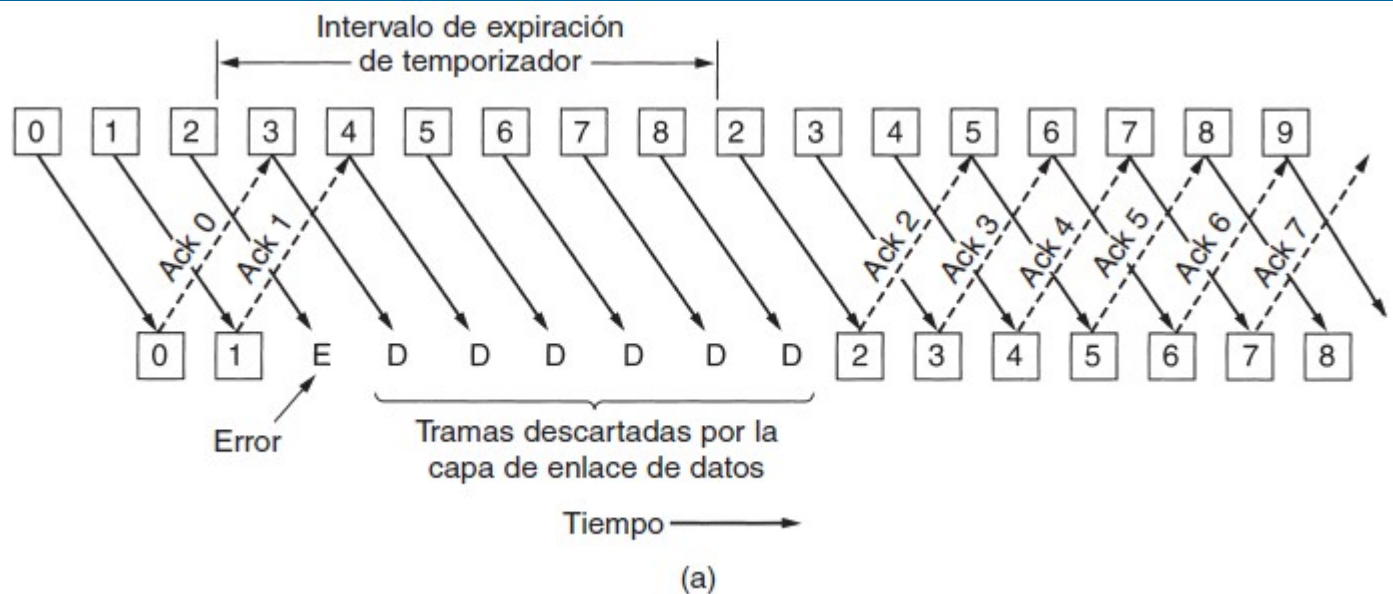
La necesidad de una ventana grande en el lado emisor se presenta cuando el producto del ancho de banda por el retardo del viaje de ida y vuelta es grande.

# PROTOCOLO DE VENTANA CORREDIZA

## Protocolo de ventana que usa retroceso N

Una manera, llamada **retroceso n**, es que **el receptor simplemente descarte todas las tramas subsecuentes**, sin enviar confirmaciones de recepción para las tramas descartadas.

Esta estrategia corresponde a una ventana de recepción de tamaño 1. En otras palabras, **la capa de enlace de datos se niega a aceptar cualquier trama excepto la siguiente que debe entregar a la capa de red.**



# PROTOCOLO DE VENTANA CORREDIZA

## Protocolo de ventana que usa retroceso N

La otra estrategia general para el manejo de errores se conoce como **repetición selectiva**. Cuando se utiliza, se descarta una trama dañada recibida, pero **las tramas en buen estado recibidas después de ésta se almacenan en el búfer**. Cuando el emisor termina, sólo la última trama sin confirmación se retransmite.

La repetición selectiva con frecuencia se combina con el hecho de que el receptor envíe una **confirmación de recepción negativa (NAK)** cuando detecta un error, por ejemplo, cuando recibe un error de suma de verificación o una trama en desorden

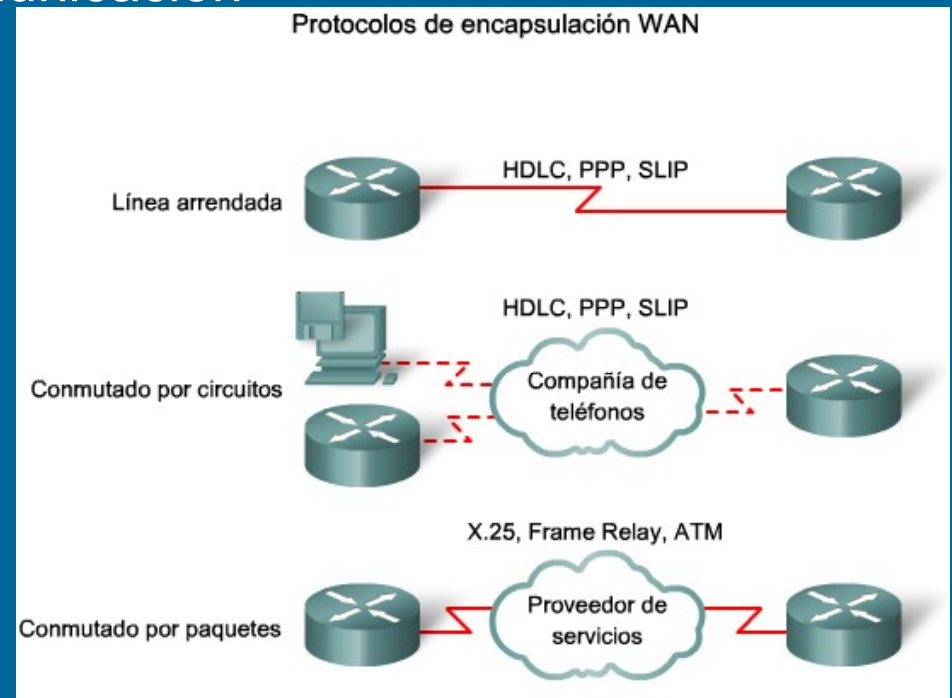


(b)

# PROTOCOLO DE ENLACE DE DATOS

## Encapsulación HDLC - Control de Enlace de Datos de Alto Nivel

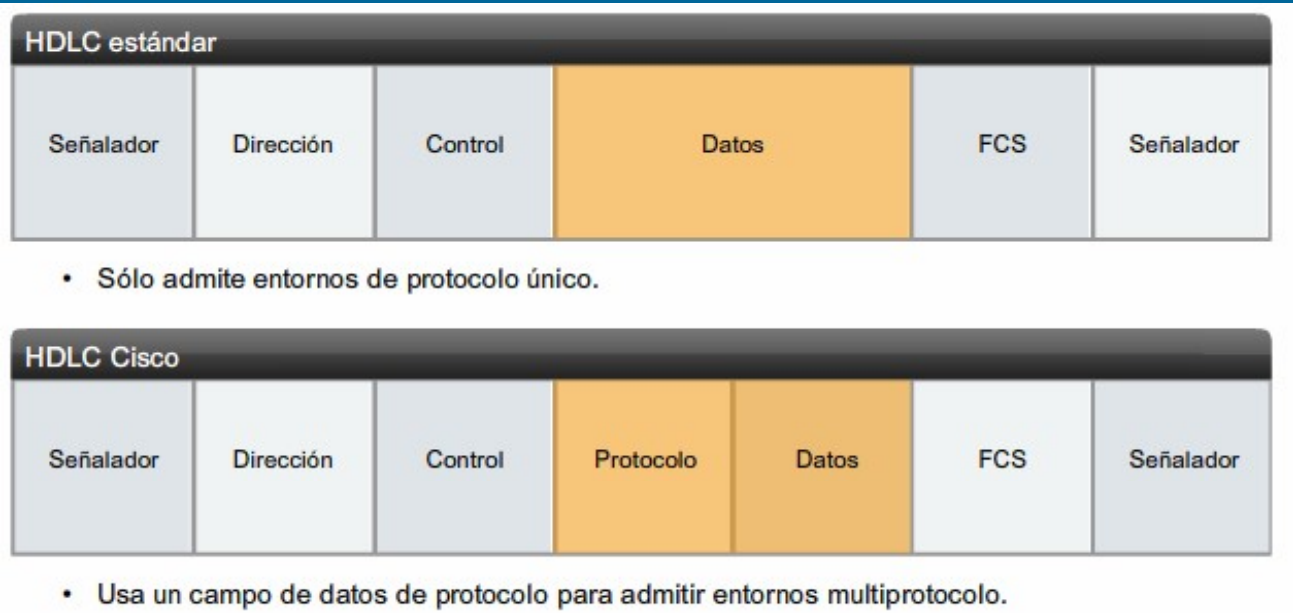
La elección del protocolo depende de la tecnología WAN y del equipo de comunicación



El HDLC utiliza **transmisión serial síncrona** para brindar comunicación libre de errores entre dos puntos. El HDLC define una estructura del entramado de Capa 2 que permite el control del flujo y el control de errores mediante el uso de acuses de recibo. Cada trama presenta el mismo formato, ya sea una trama de datos o una trama de control.

# PROTOCOLO DE ENLACE DE DATOS

## Encapsulación HDLC - Control de Enlace de Datos de Alto Nivel



**Señalador:** La trama siempre comienza y finaliza con un campo señalador de 8 bits (**01111110**)

**Dirección:** puede contener una dirección específica, un grupo de direcciones o una dirección de broadcast.

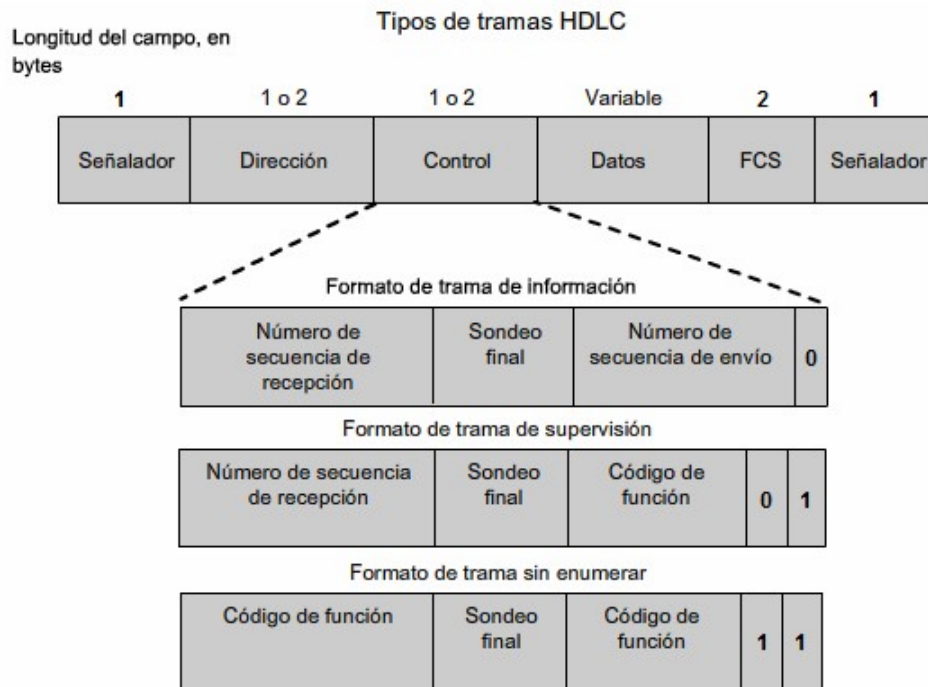
**Control:** el campo de control utiliza tres formatos diferentes, según el tipo de trama HDLC usada.

# PROTOCOLO DE ENLACE DE DATOS

## Encapsulación HDLC - Control de Enlace de Datos de Alto Nivel

**Trama de información (I):** las tramas I contienen información de la capa superior y alguna información de control. Esta trama envía y recibe números de secuencia y el bit de sondeo final (P/F) realiza el control de flujo y error.

**Trama de supervisión (S):** Una trama S puede solicitar y suspender la transmisión, solicitar siguiente trama, ack negativo, detener envío solicitar una trama especificada



**Trama sin enumerar (U):** se usa para propósitos de control, aunque también puede servir para llevar datos cuando se solicita un **servicio no confiable sin conexión**

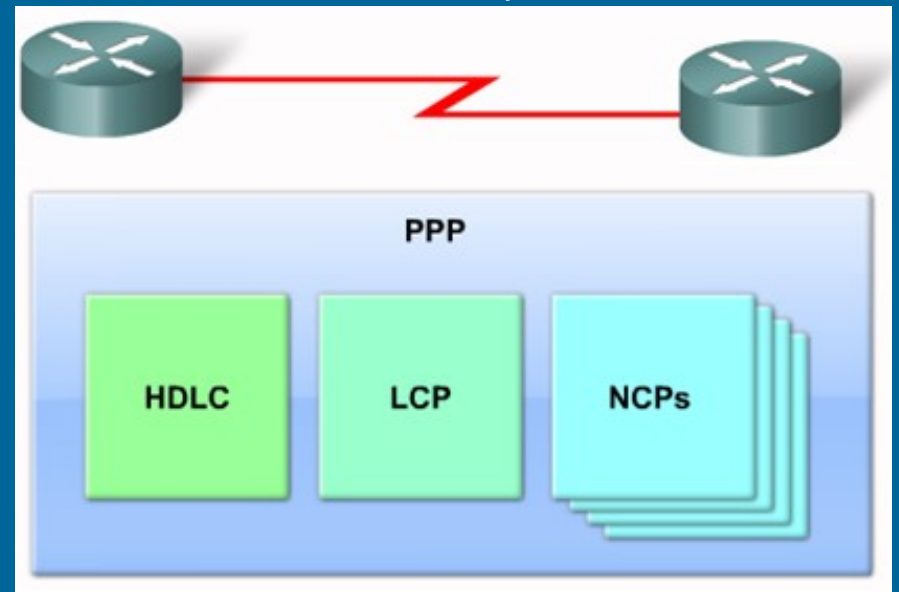
# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto

El **PPP** establece una conexión directa mediante cables seriales, líneas telefónicas, líneas troncales, teléfonos celulares, enlaces de radio especializados o enlaces de fibra óptica.

PPP contiene tres componentes principales

1. El **protocolo HDLC** para la encapsulación de datagramas a través de enlaces punto a punto.
2. Un protocolo de control de enlace (**LCP, Link Control Protocol**)
3. Una familia de protocolos de control de red (**NCP, Network Control Protocols**)

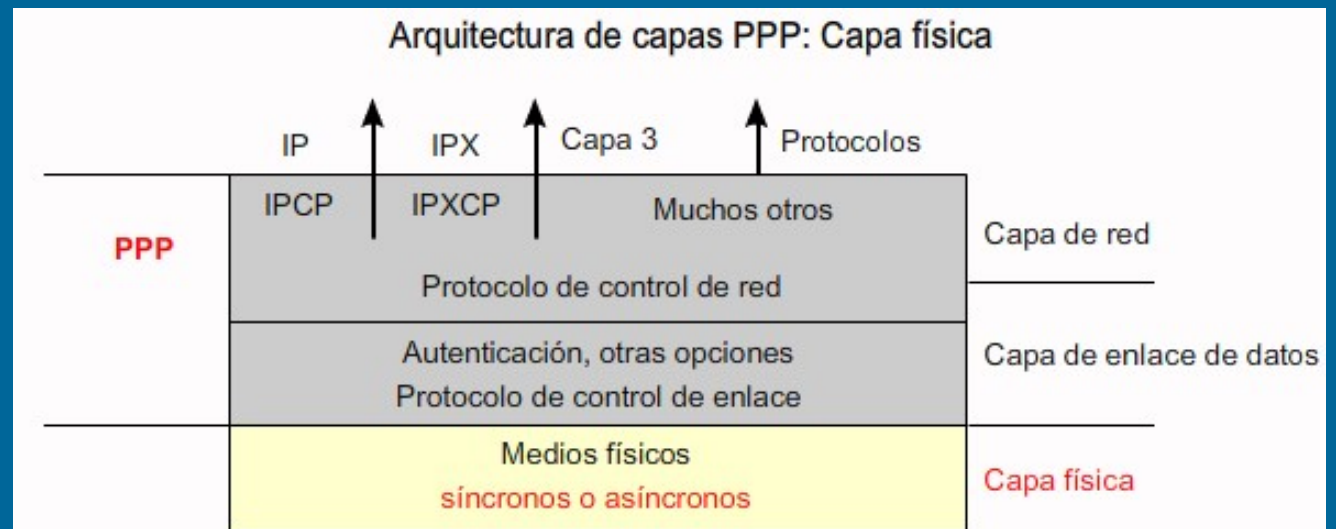




# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto

La mayoría del trabajo realizado por el **PPP se produce en el enlace de datos** y las capas de red mediante el **LCP** y los **NCP**. EL **LCP establece la conexión PPP** y sus parámetros, los **NCP manejan configuraciones de protocolo** de capa superior y el LCP finaliza la conexión PPP.



Mediante las funciones de nivel más bajo, PPP puede usar:

- Medios físicos síncronos
- Medios físicos asíncronos, como los que utiliza el servicio telefónico básico para las conexiones dial-up del módem



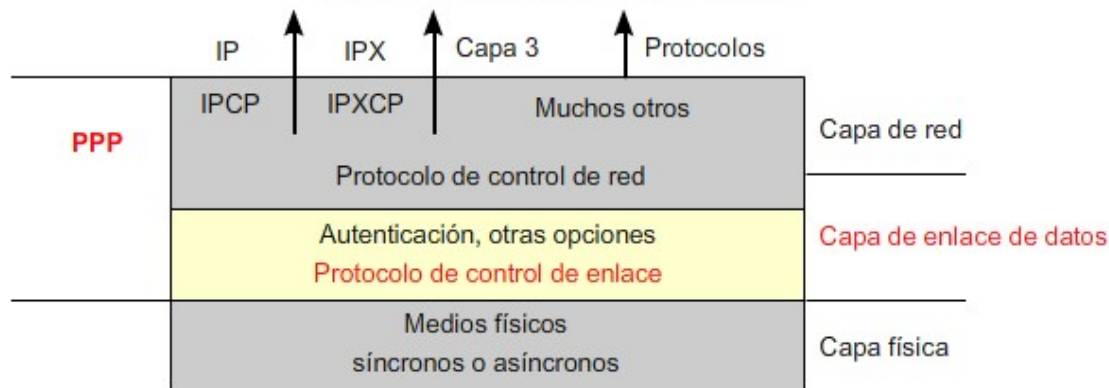
# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto

El **LCP** brinda configuración automática de las interfaces en cada extremo, lo que incluye:

- El manejo de límites variables en el **tamaño del paquete**
- La **detección de errores** comunes de configuración
- La **finalización del enlace**
- La determinación de **cuándo** un enlace **funciona correctamente o cuándo falla**

Arquitectura de capas PPP: Capa LCP



Formatos de encapsulación (**autenticación, compresión, detección de errores**)

PPP ofrece opciones de servicio en LCP y es utilizado principalmente para negociar y para verificar las tramas cuando se implementan los controles punto a punto especificados por un administrador.

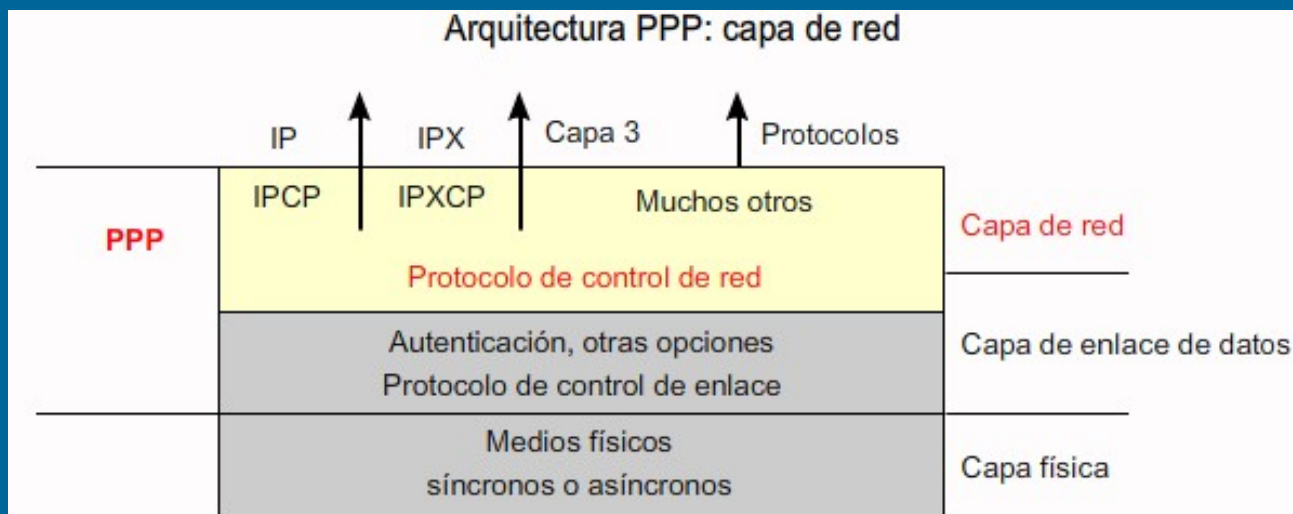
# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto

El PPP permite que varios protocolos de capa de red operen en el mismo enlace de comunicación.

PPP utiliza un **NCP** distinto

- ❑ **IP** (**IPCP**, IP Control Protocol)
- ❑ **IPX** utiliza el protocolo de control IPX (**IPXCP**, IPX Control Protocol).



Mediante las funciones de nivel superior, PPP transporta paquetes de varios protocolos de capa de red en NCP. Estos son campos funcionales que contienen códigos estandarizados para indicar el tipo de protocolo de capa de red que PPP encapsula.

# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto

### Estructura de la trama PPP

Campos de la trama PPP					
Longitud del campo, en bytes					
1	1	1	2	Variable	2 o 4
Señalador	Dirección	Control	Protocolo	Datos	FCS

### Establecimiento de una sesión PPP

#### Establecimiento de una sesión PPP



Fase 1. Establecer el enlace: "Negociemos".



Fase 2. Determinar la calidad del enlace: "Quizás deberíamos analizar algunos detalles sobre la calidad. O quizás no ..."



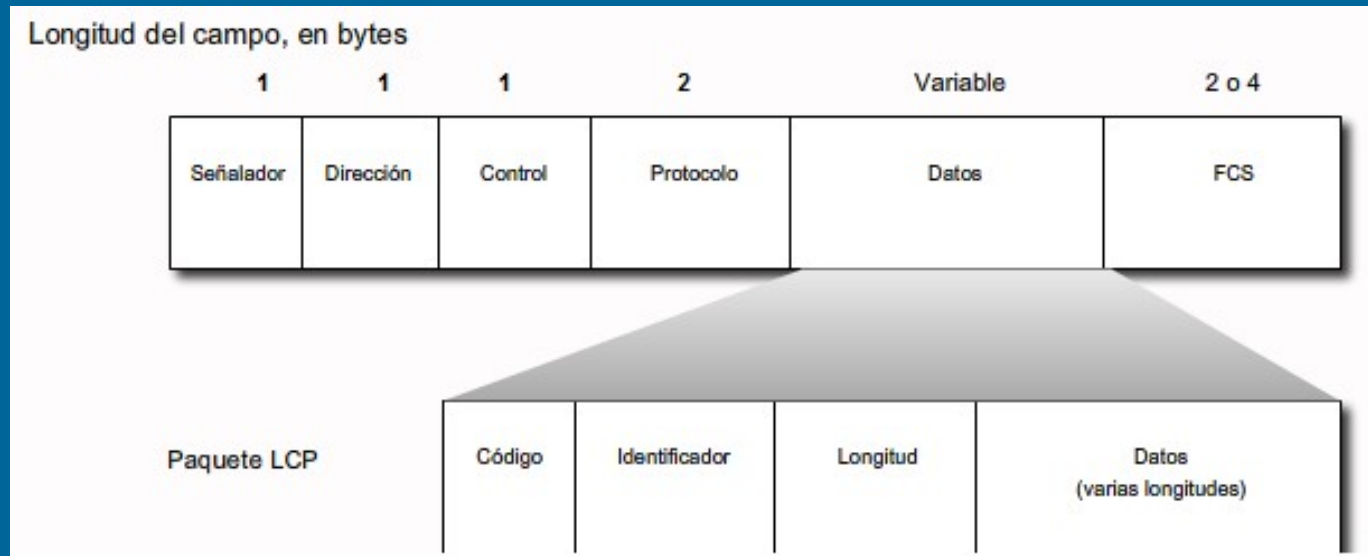
Fase 3. Negociación del protocolo de red: "Está bien, dejaré a los NCP que analicen los detalles de nivel superior".

El LCP realiza toda la conversación.

# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto

### Estructura Paquete LCP



El campo de código tiene una longitud de un byte e identifica el tipo de paquete LCP.

### Longitud:

El campo de longitud tiene dos bytes de extensión e indica la longitud total (incluidos todos los campos) del paquete LCP.

### Identificador:

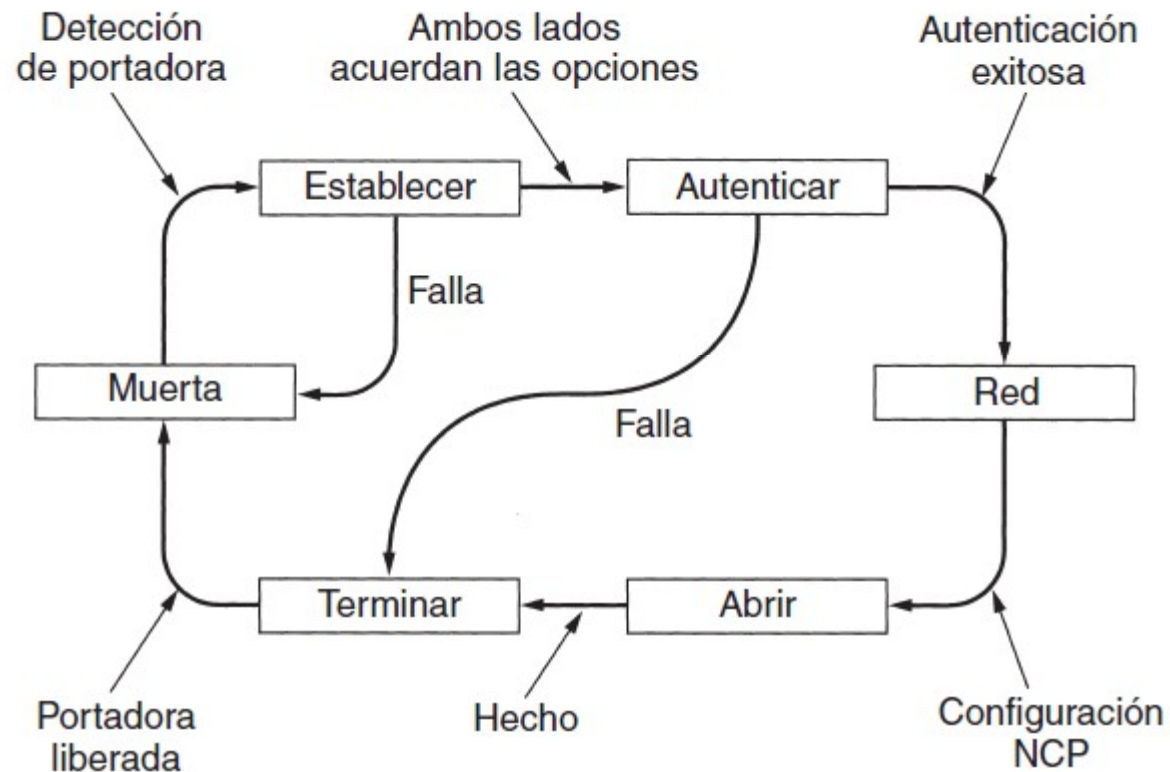
El campo de identificador tiene una longitud de un byte y se utiliza para hacer coincidir solicitudes y respuestas de paquetes.

### Datos (varias longitudes)

El campo de datos tiene cero o más bytes, según lo indica el campo de longitud. El formato de este campo es determinado por el código.

# PROTOCOLO DE ENLACE DE DATOS

## PPP - Protocolo Punto a Punto



**Figura 3-28.** Diagrama de fases simplificado para activar y desactivar una línea.