



# Capa de enlace

Comunicaciones T-412

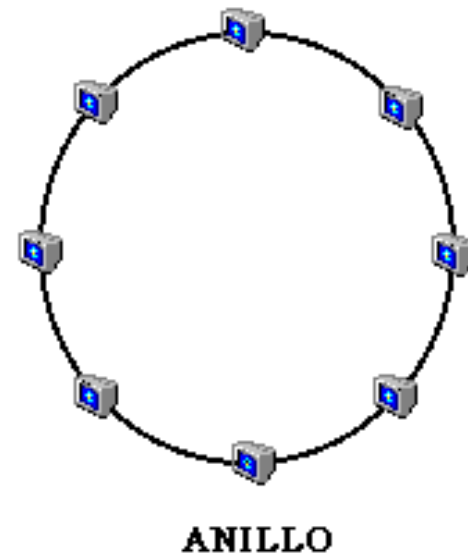
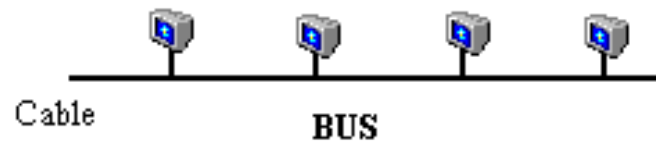
2012

Contacto: [bulacio@cifasis-conicet.gov.ar](mailto:bulacio@cifasis-conicet.gov.ar)



# Capa de Enlace: Objetivo

- Transferencia fiable de información de 2 máquinas adyacentes a través de un circuito de transmisión de datos.
- Recibe peticiones de la capa de red y utiliza los servicios





# Capa de Enlace: Motivación

El canal físico

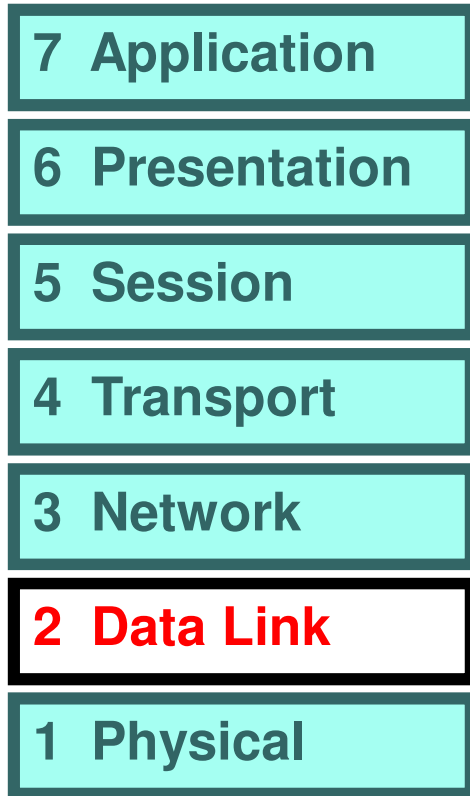
- Tiene errores
- Tiene retardo
- Puede ser compartido

Tx, Rx pueden tener

- Velocidades distintas
- Flujo de Ix distinto

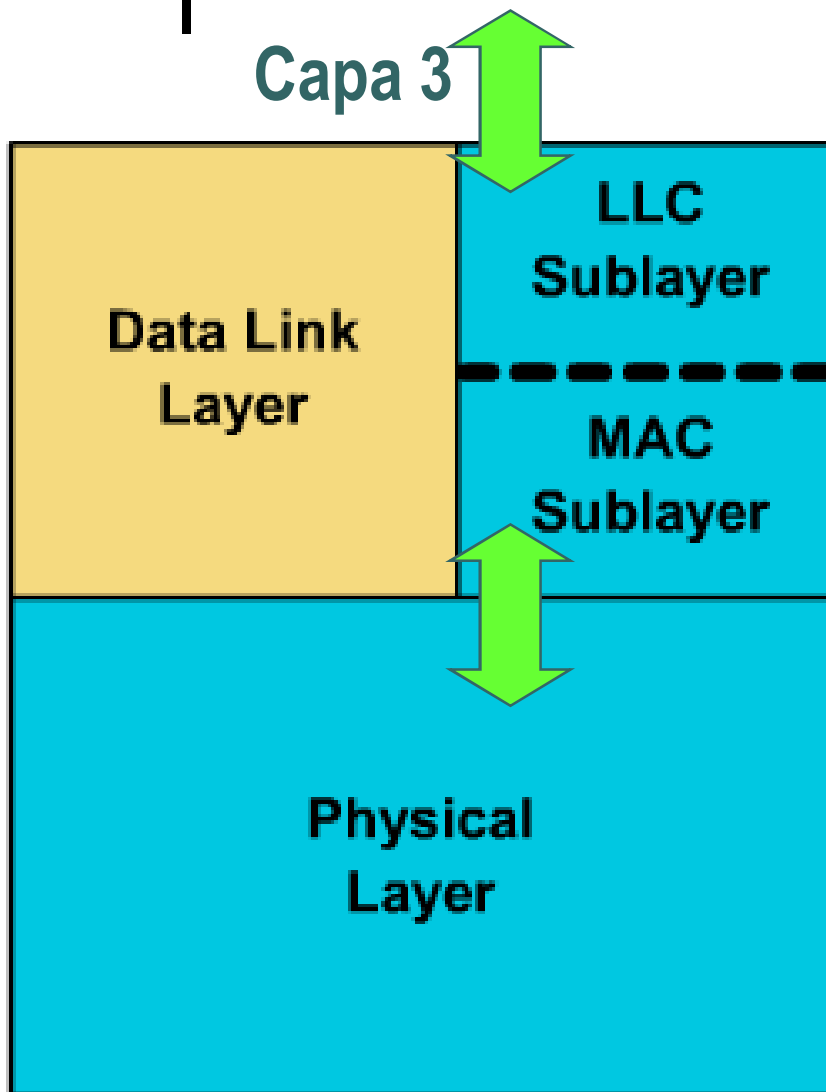


# Capa 2: Funciones



- Implementar la **interfaz de servicio** para Capa de Red,
- Agrupar los bits a transmitir en **tramas (enmarcar)**,
- Realizar la **detección de errores**,
- Regular el **flujo de las tramas**,
- **Traducción** de tramas de redes heterogéneas,
- Especificar quién va a recibir los bits: **direccionamiento**,
- Definir métodos de **acceso al medio común**.

# ● ● ● | Capa 2: Funciones-Subdivisión



➤ **Logical Link Control (LLC):**  
Entramado, Control de flujo, Control de Errores; independiente de la tecnología LAN

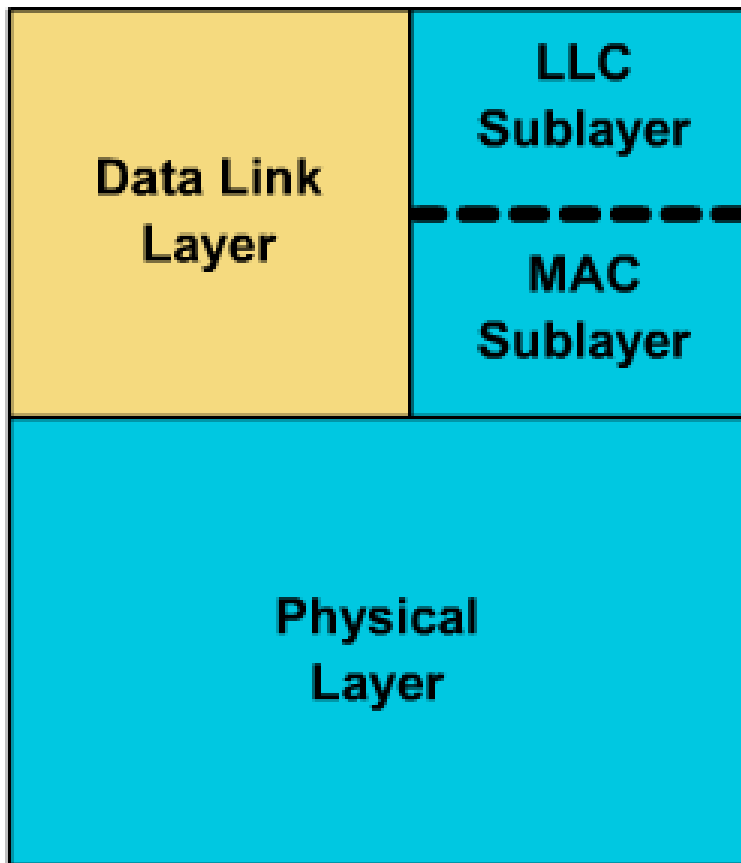
➤ **Media Access Control (MAC):**  
Acceso a la Capa Física, Protocolos MAC asociado a la topología, y considerando la tecnología



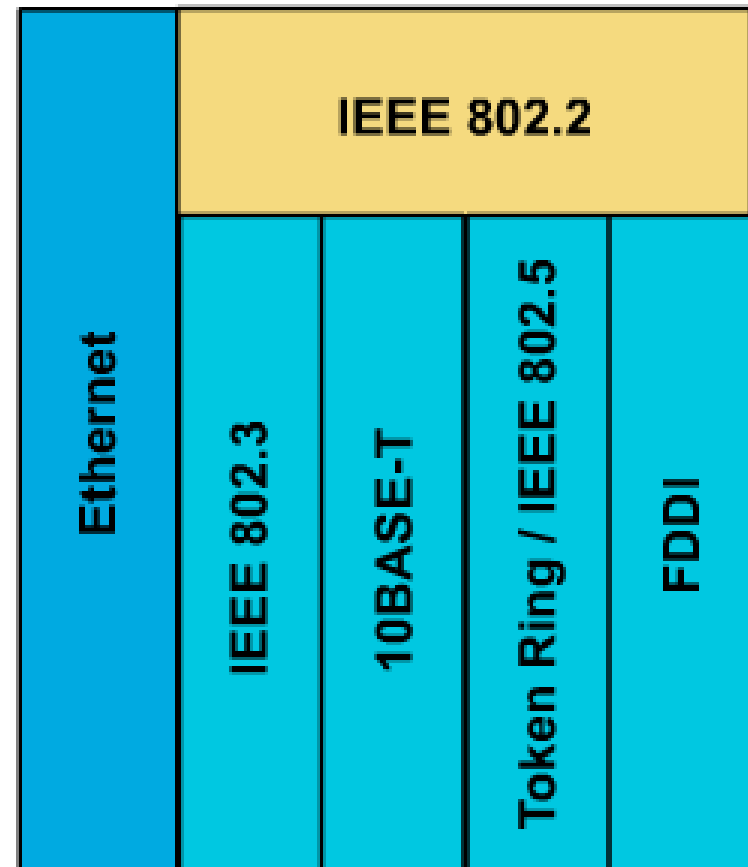
# Capa 2: LAN std

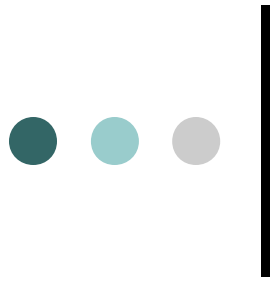


## OSI Layers



## LAN Specification





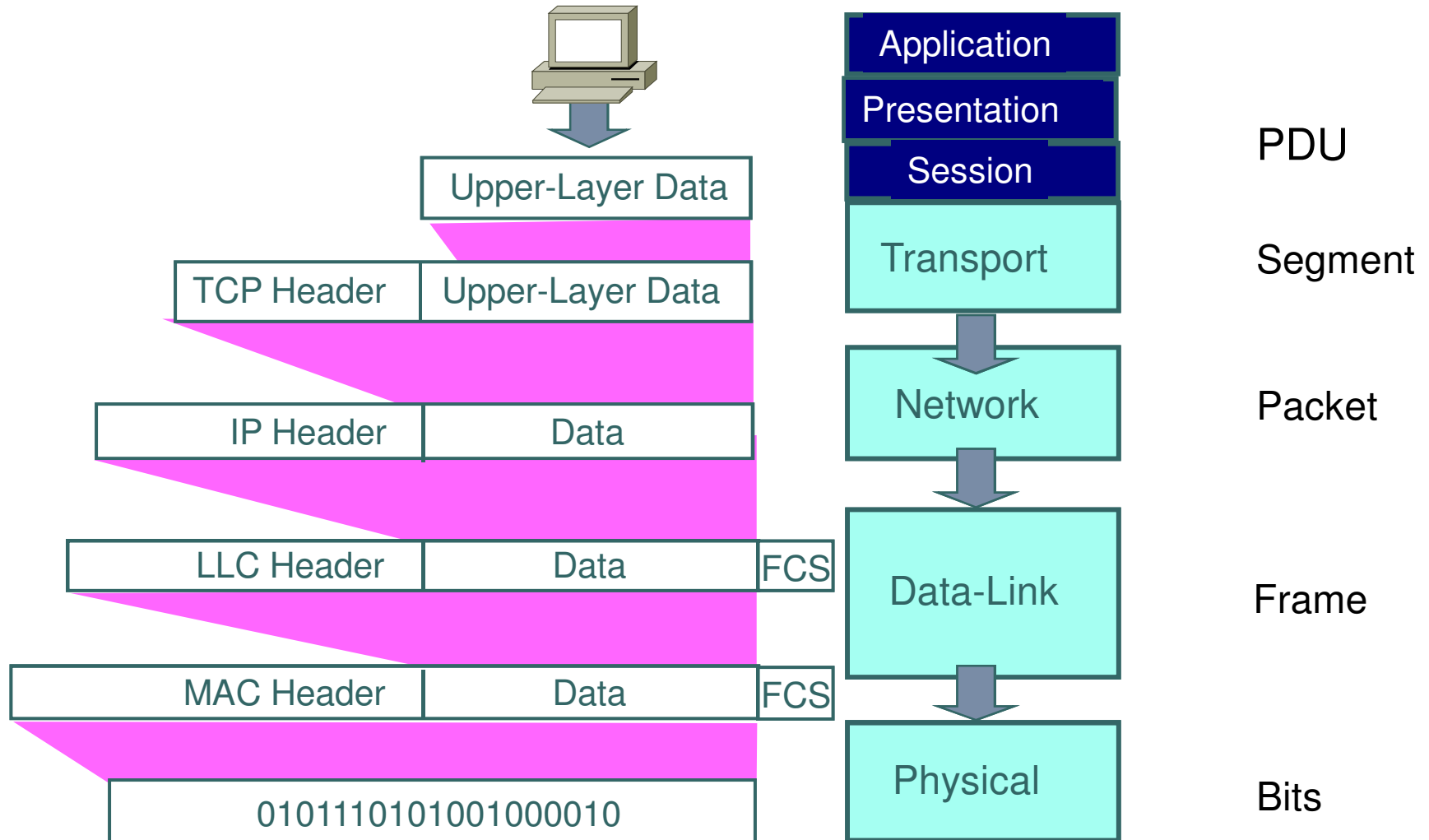
IEEE



- The **I**nstitute of **E**lectrical and **E**lectronic **E**ngineers.
- LAN standards:
  - **802.2: LLC.**
  - **802.3: MAC ~ Ethernet.**
  - **802.5: MAC ~ Token ring.**
  - **802.11: Wireless LAN.**



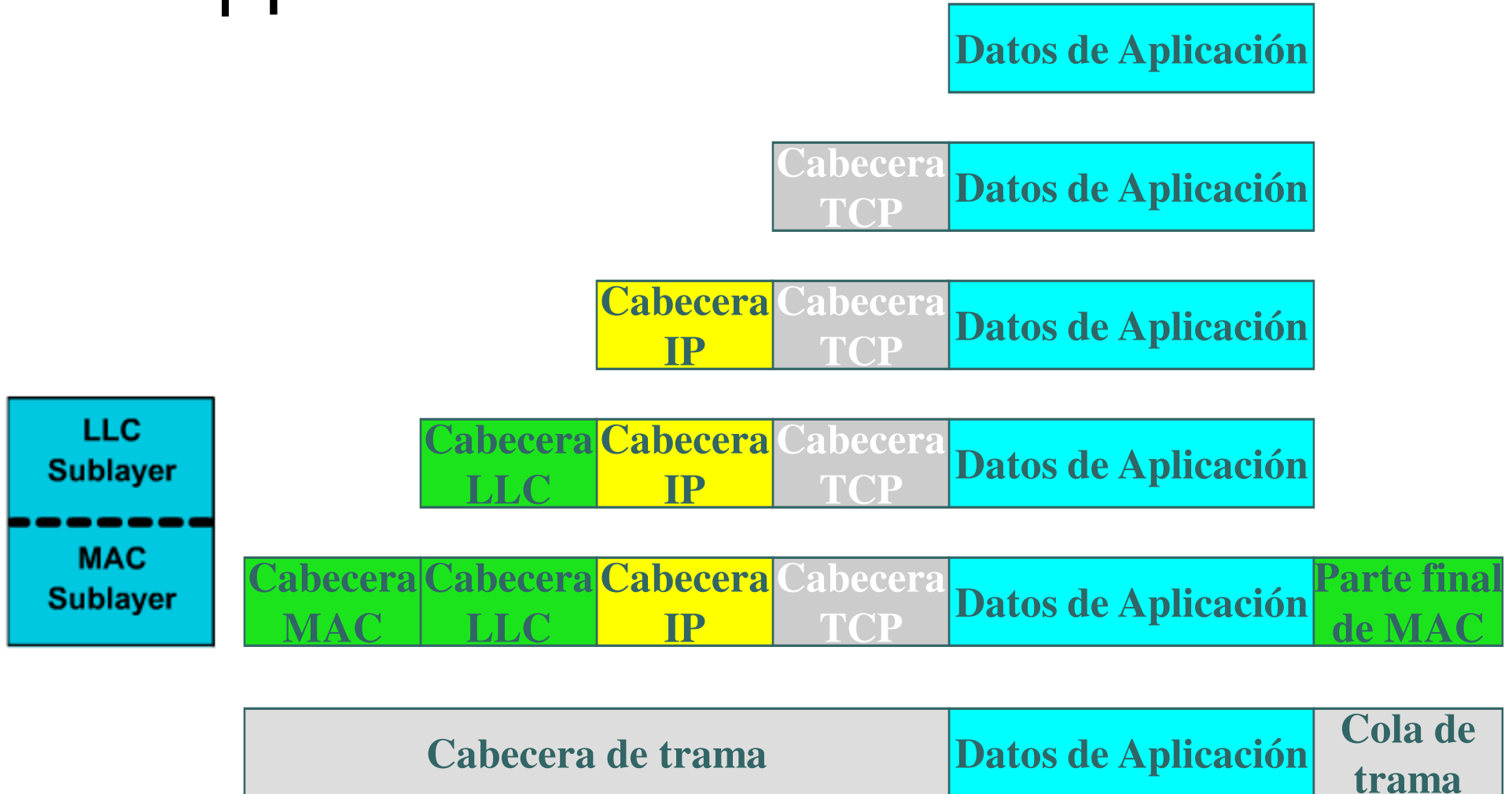
# Encapsulamiento de datos







# Encapsulamiento del protocolo LAN





# LLC: Gestión de Enlace



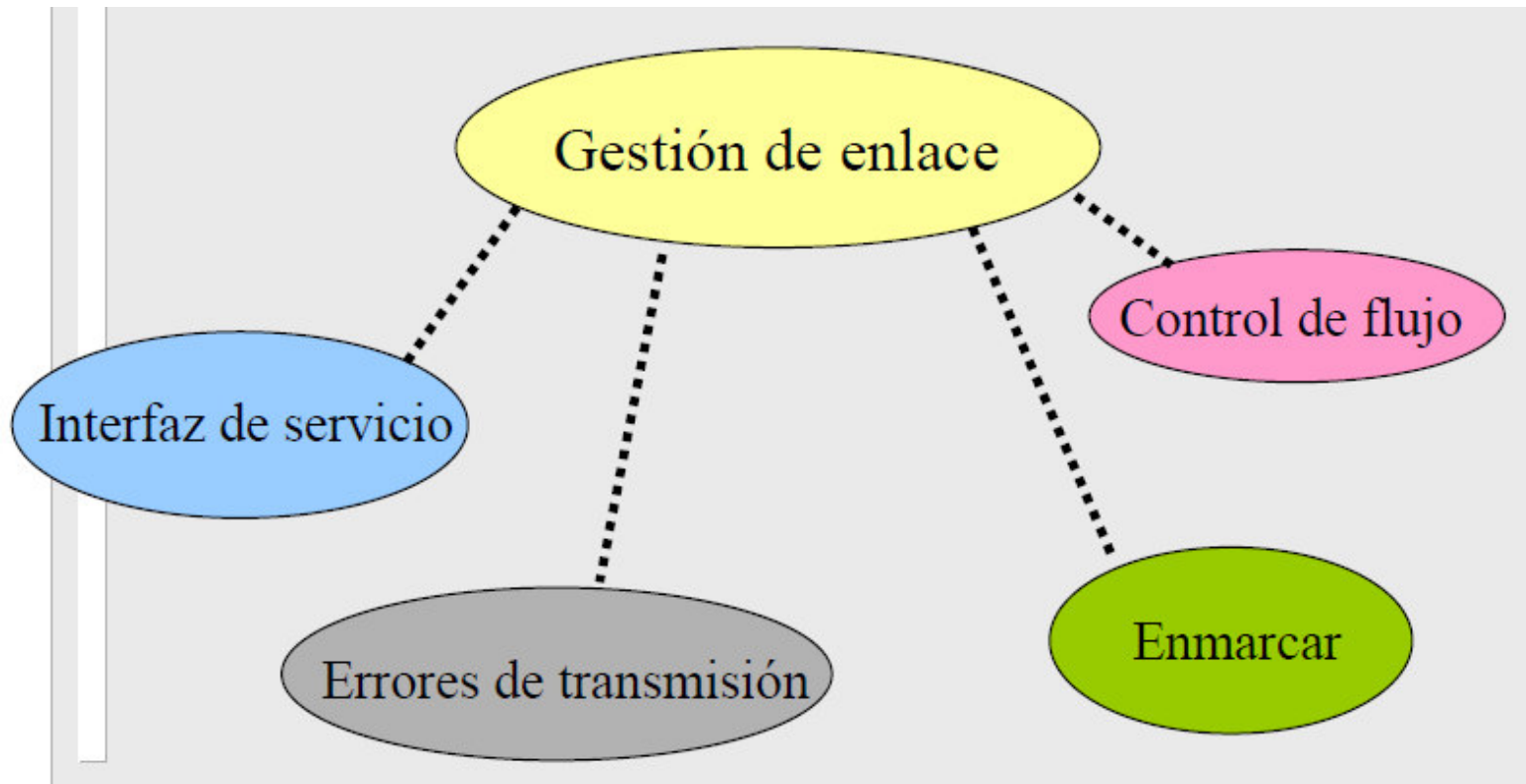
# Funcionalidades LLC

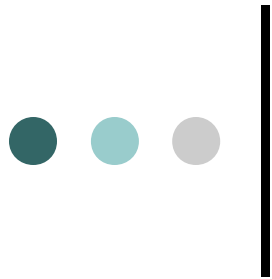
- Gestión de enlace **independiente** del medio físico, de la topología y del método de acceso empleados en la LAN
  - Entramado,
  - Control de flujo,
  - Control de Errores; independiente de la tecnología LAN
- Transporta paquetes de C3 agregando Ix de control (dir. LLC) para la entrega en destino: Punto de Acceso al Servicio Destino (DSAP) y el Punto de Acceso al Servicio Fuente (SSAP).



# LLC: Funciones

Gestión de enlace **independiente** del medio físico, de la topología y del método de acceso empleados en la LAN





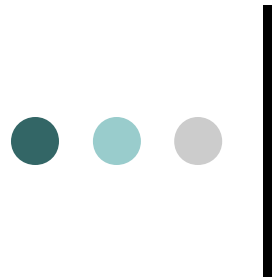
# Trama

Las entidades de enlace se comunican entre sí mediante las unidades de datos del **nivel de enlace** denominadas **tramas**.

Las tramas están compuestas por campos, por ejemplo:

<b>Cabecera</b>	Permite identificar el <i>comienzo</i> de la trama
<b>Control</b>	Información de <i>control</i> .
<b>Datos</b>	Contiene los <i>datos</i> de usuario
<b>Final</b>	Permite identificar el <i>fin</i> de la trama
<b>Redundancia</b>	Contiene información para control de <i>errores</i>





# Métodos de Enmarcado

## **División de Ix de bits en tramas**

### *Orientados a Caracteres:*

1. Cuenta de caracteres
2. Caracteres de inicio fin con inserción de carácter

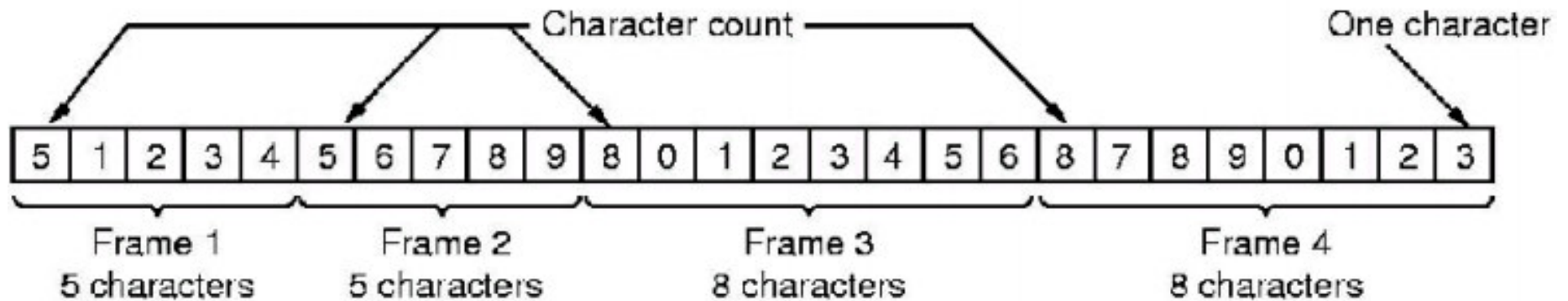
### *Orientados a bits:*

3. Cadena de bits de inicio-fin con inserción de bit
4. Violaciones de código de Capa Física

- 1. Cuenta de caracteres
- 2. Caracteres de inicio fin con inserción de carácter
- 3. Cadena de bits de inicio-fin con inserción de bit
- 4. Violaciones de código de Capa Física

# ● ● ● | Enmarcado

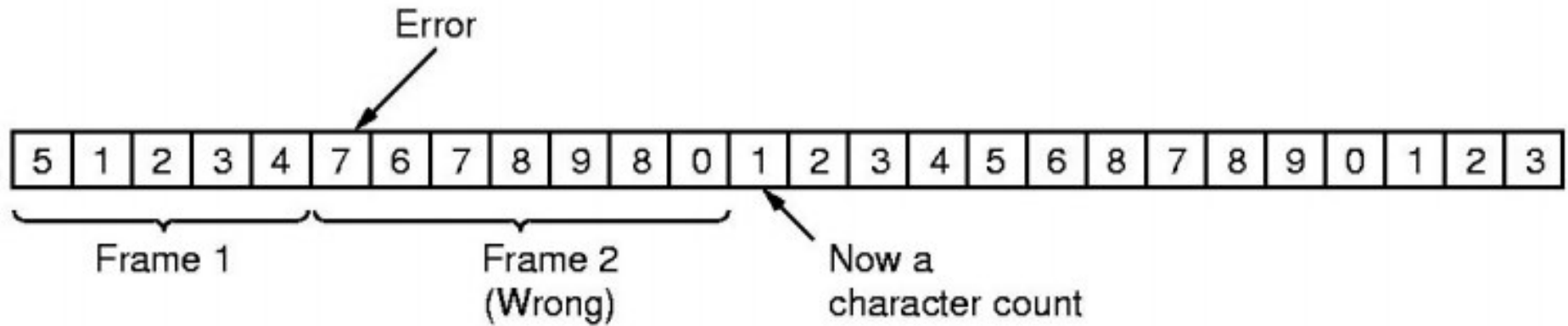
**1. Cuenta de caracteres:** Un carácter indica la longitud de caracteres de datos de la trama y por tanto, sabe cuando termina



- 1. Cuenta de caracteres
- 2. Caracteres de inicio fin con inserción de carácter
- 3. Cadena de bits de inicio-fin con inserción de bit
- 4. Violaciones de código de Capa Física

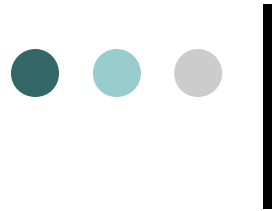
# Enmarcado

**Problema:** puede alterarse el carácter longitud...





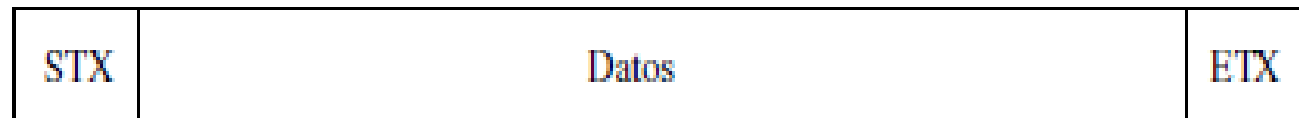
1. Cuenta de caracteres
- 2. Caracteres de inicio fin con inserción de carácter**
3. Cadena de bits de inicio-fin con inserción de bit
4. Violaciones de código de Capa Física



# Enmarcado

## 2. Caracteres de inicio y fin.

Identifica el comienzo y fin de la trama mediante dos caracteres especiales STX y ETX



**STX** (Start of TeXt), **ETX** (End of TeXt)

- 1. Cuenta de caracteres
- 2. **Caracteres de inicio fin con inserción de carácter**
- 3. Cadena de bits de inicio-fin con inserción de bit
- 4. Violaciones de código de Capa Física

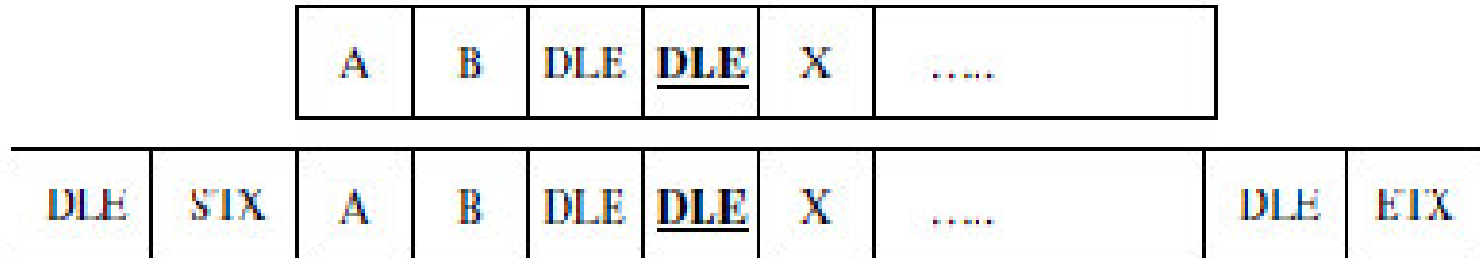
# ● ● ● | Enmarcado

Como estos caracteres de inicio-fin pueden estar dentro de los datos, se preceden por un **DLE** (Data Link Escape)

Las tramas se encontrarán entre los delimitadores: **DLE STX y DLE ETX**



Si aparece un DLE dentro de los datos, **antes de transmitir se duplica**  
 ejemplo: para Tx los datos 'A', 'B', 'DLE', 'X', .....



1. Cuenta de caracteres
2. Caracteres de inicio fin con inserción de carácter
3. **Cadena de bits de inicio-fin con inserción de bit**
4. Violaciones de código de Capa Física

# ● ● ● | Enmarcado

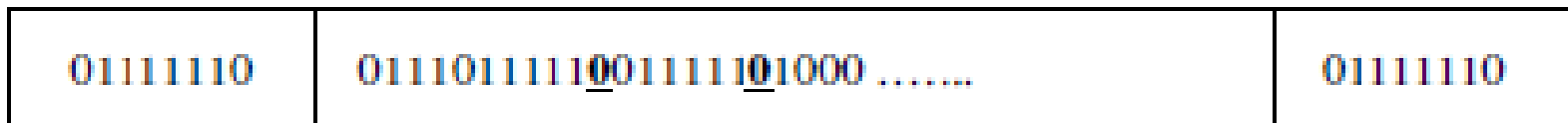
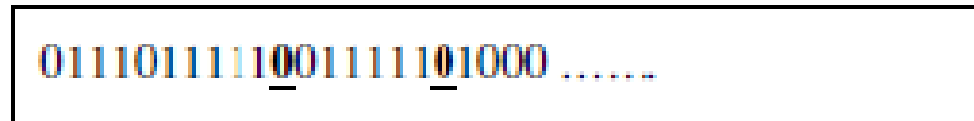
## 3. Banderas de inicio y fin. Inserción de bit.

Similar al anterior, pero a nivel bits: usa un patrón de bits como comienzo y fin de la trama 0111 1110



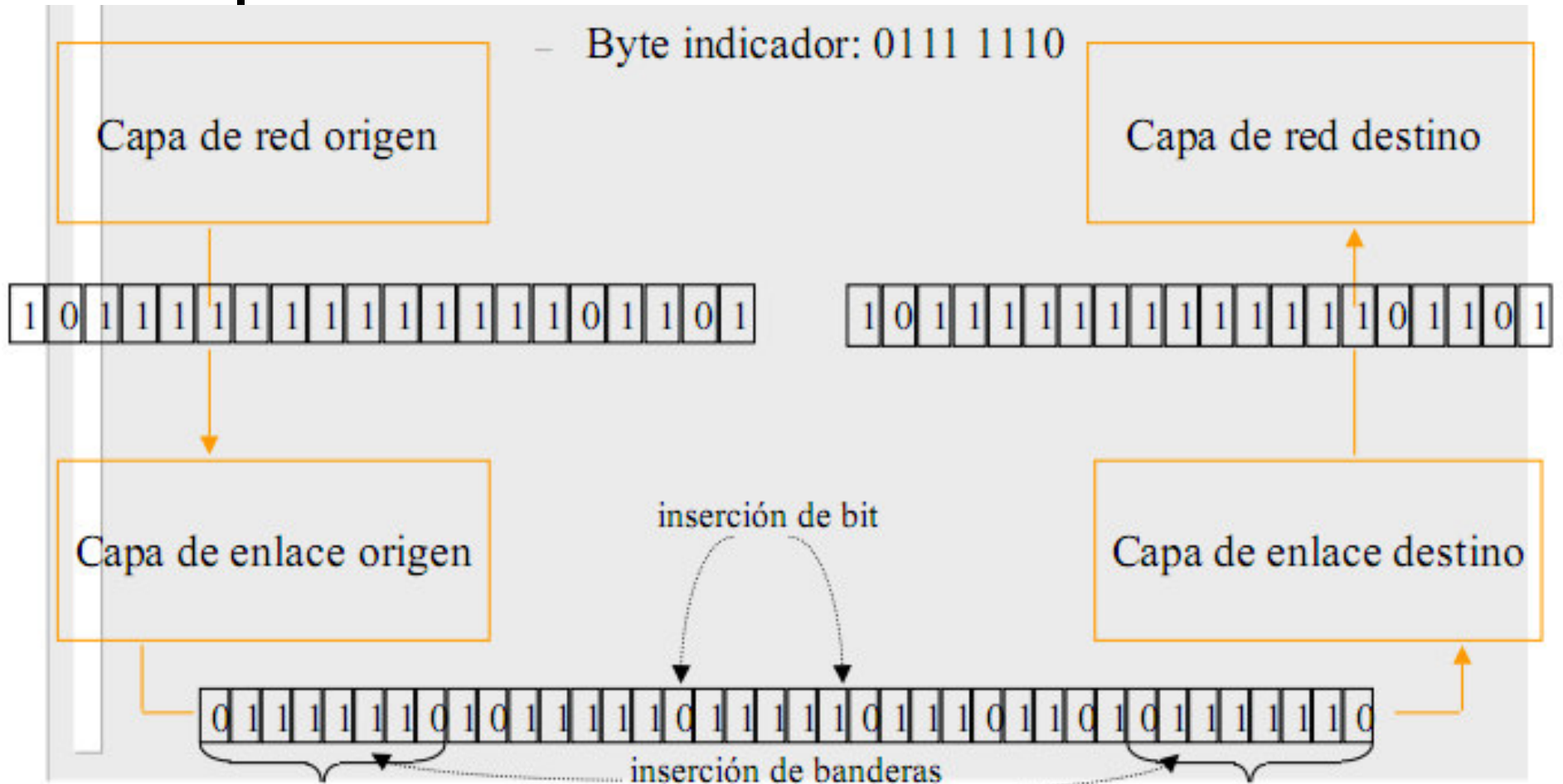
Si dentro de los datos aparece una secuencia de 5 bit con valor 1 consecutivos se **inserta un bit** con valor 0

Por ejemplo para transmitir 01110111110111111000 .....



- 1. Cuenta de caracteres
- 2. **Caracteres de inicio fin con inserción de carácter**
- 3. Cadena de bits de inicio-fin con inserción de bit
- 4. Violaciones de código de Capa Física

# ● ● ● | Enmarcado: Cadena de bits



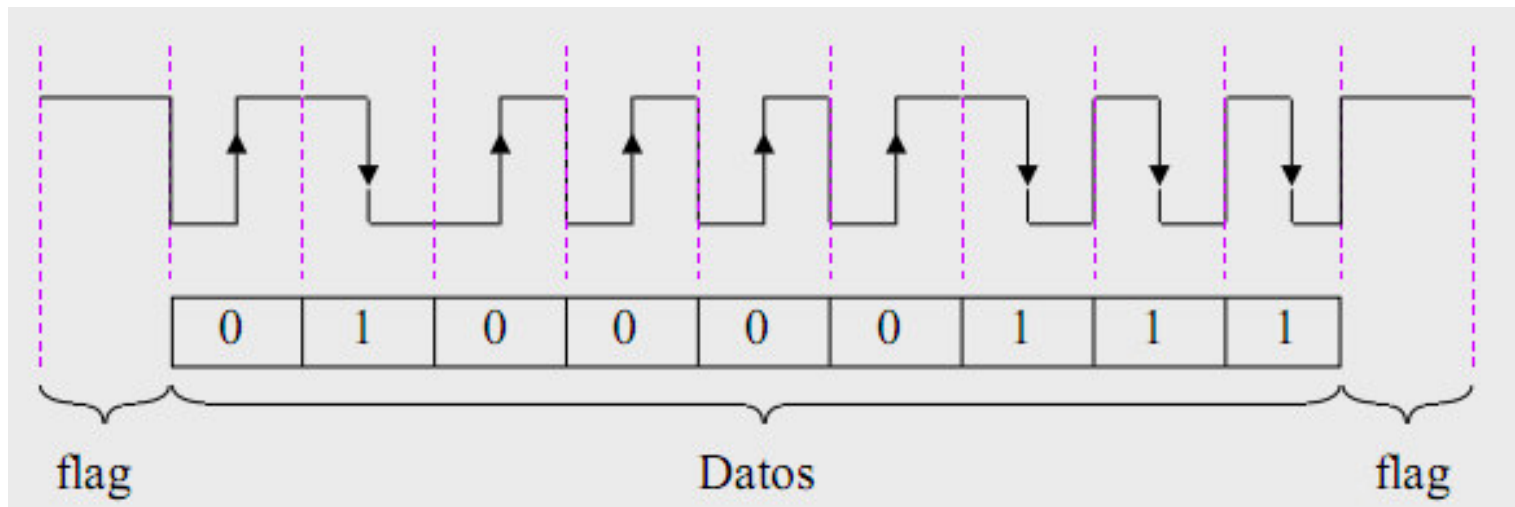
1. Cuenta de caracteres
2. Caracteres de inicio fin con inserción de carácter
3. Cadena de bits de inicio-fin con inserción de bit
4. **Violaciones de código de Capa Física**

# Enmarcado

## 4. Violaciones del nivel físico

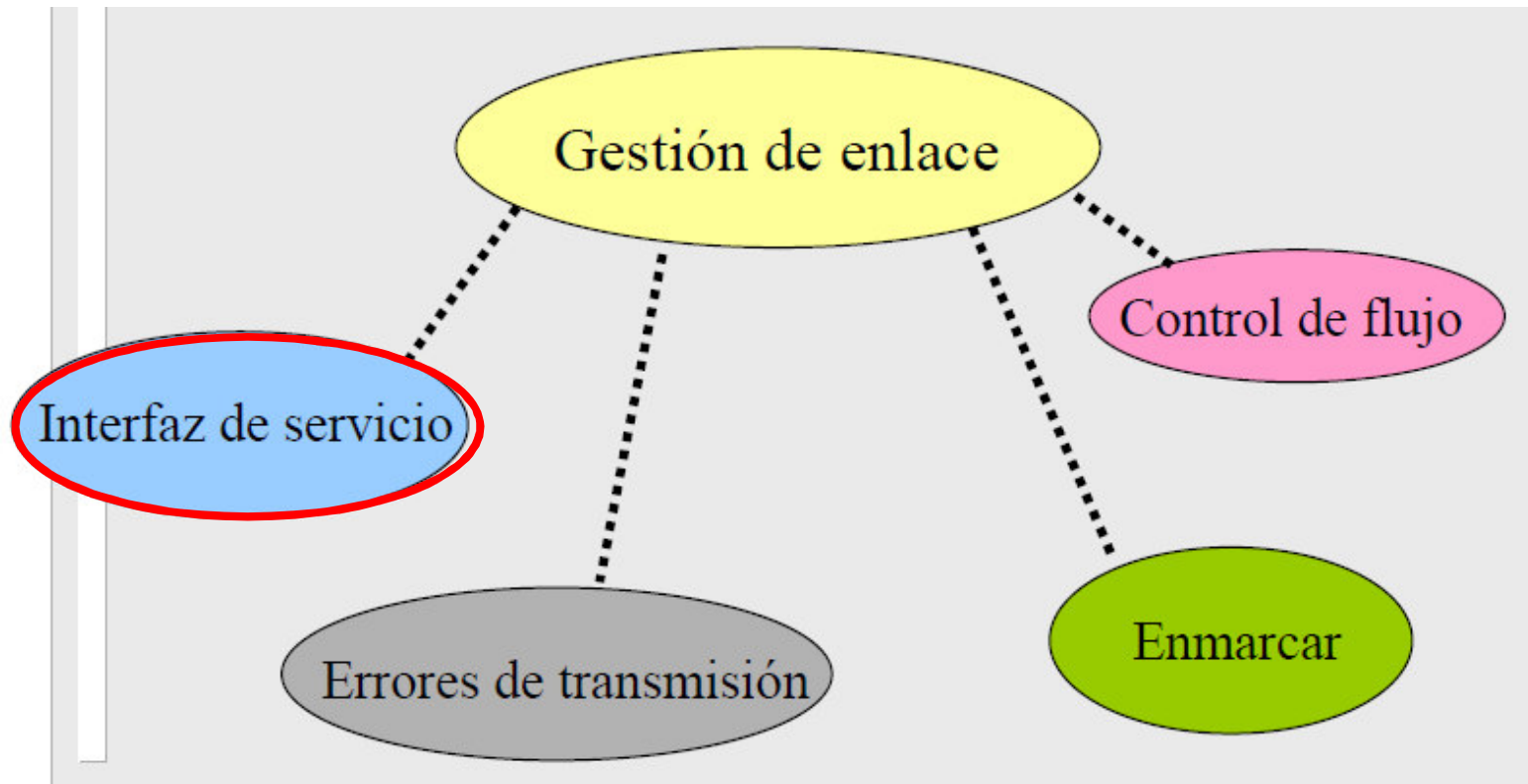
Por ej., el código Manchester (Token Ring) codifica el **1** como **alto-bajo** y el **0** como **bajo-alto**.

La combinación **alto-alto** no usada, representa el inicio y fin de trama.





# LLC: Funciones





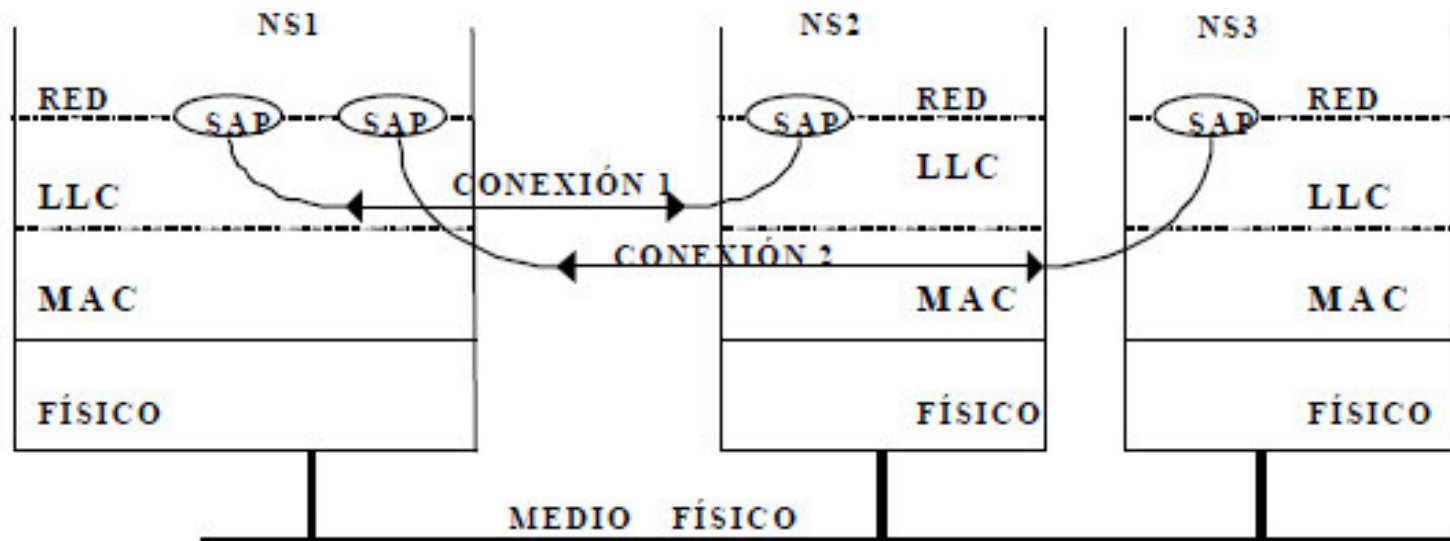
# LLC: Servicios de capa 2 – 3+

## IEEE 802.2

- **Servicios sin conexión y sin reconocimiento (Tipo I)**
  - Transmite tramas hacia el destino sin esperar reconocimiento
  - No tiene control de flujo ni detección de errores: entrega de datos no garantizada
- **Servicios con conexión (Tipo II)**
  - Conexión lógica entre extremos
  - Tiene control de flujo y detección de errores: garantiza recepción de tramas en orden correcto
- **Servicio sin conexión y con reconocimiento (Tipo III)**
  - No se establece conexión lógica, pero los datagramas son confirmados!!!
  - P. Ej. Medios de Txión inestables

# Servicios: Service Access Point (SAP)

SAP genérico:  
Pej. una  
dirección de  
puerto



- ✓ Una estación puede cursar varias comunicaciones lógicas => para diferenciarlas se usan los **SAPs**: Punto de Acceso al Servicio Destino (DSAP) y el Punto de Acceso al Servicio Fuente (SSAP).
- ✓ La dirección SSAP es individual y única en la estación a la que pertenece; la DSAP puede identificar a un individuo o a un grupo





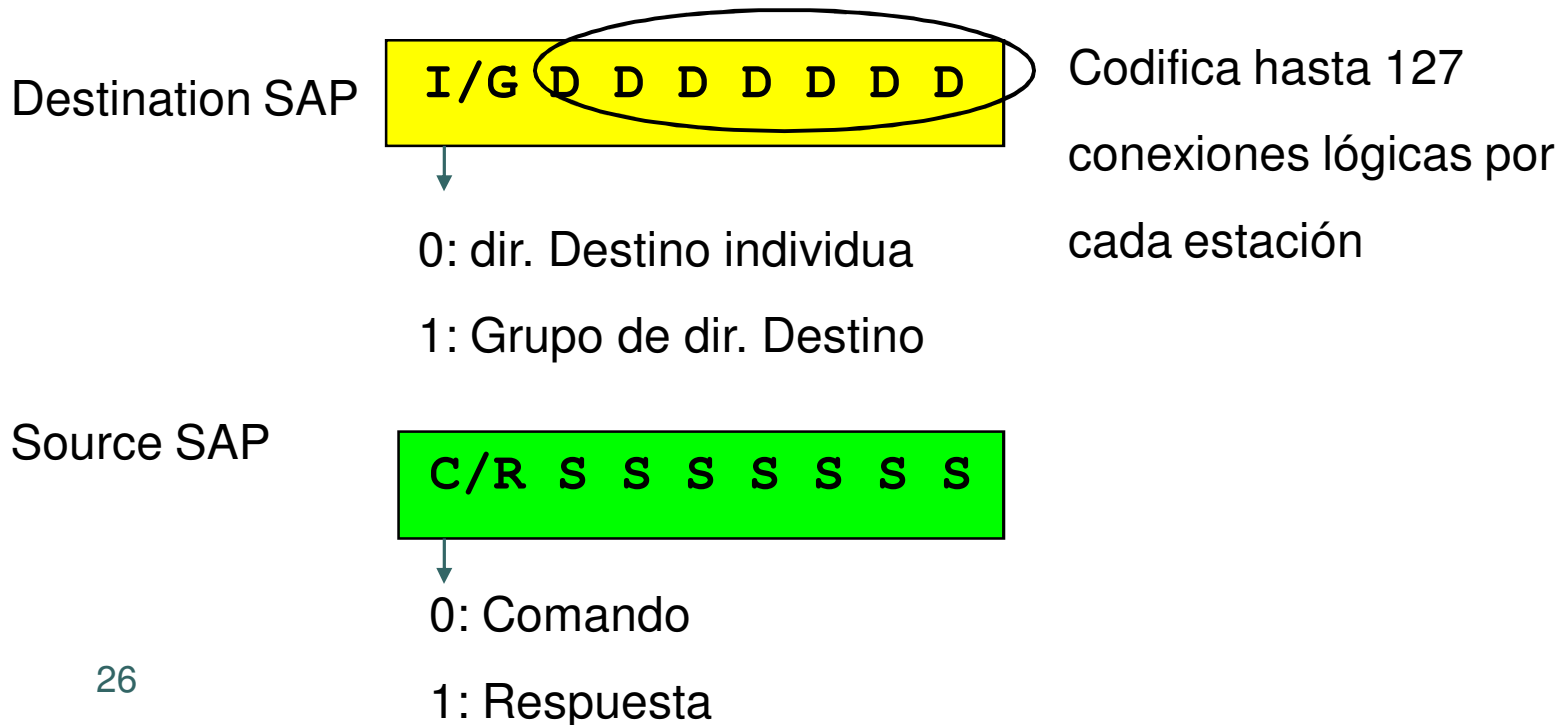
# Protocolo LLC

Reglas para proveer un servicio:

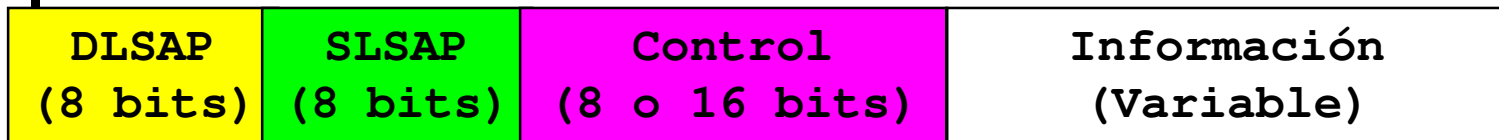
- 1) ¿Cuál es el intercambio de información de Capa 2 entre dos entidades para **establecer/brindar** un servicio?
- 2) ¿Cómo es el diálogo?,
- 3) ¿Cómo son los mensajes a intercambiar?



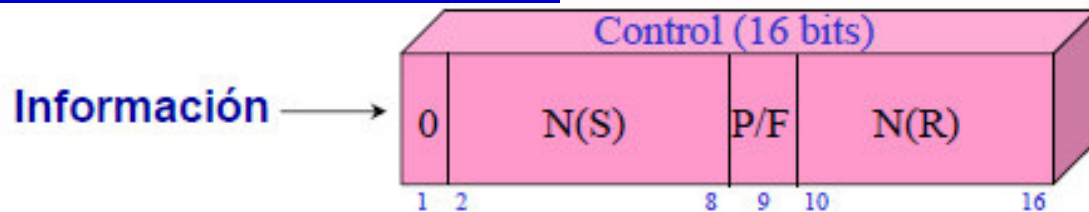
# Formato de trama LLC PDU



# Formato de trama LLC PDU

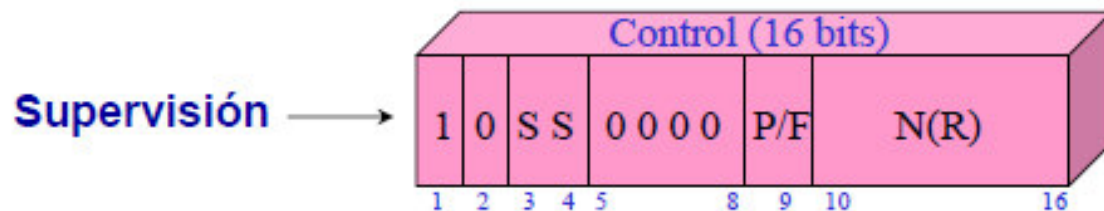


## Tipos de Tramas LLC



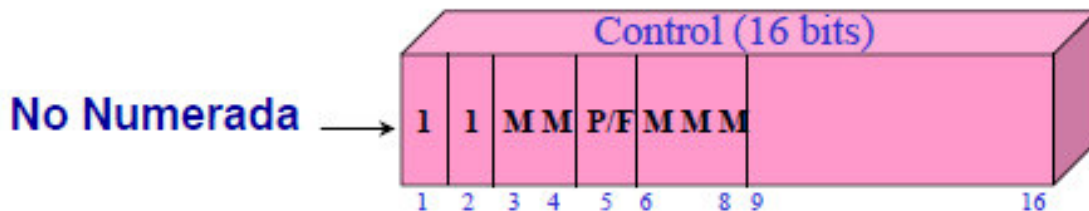
N(S) ≡ Número de secuencia de envío

N(R) ≡ Número de secuencia recibido



P/F ≡ Bit de Poll / Final

S ≡ Bits de tipo de trama de supervisión



M ≡ Bits de tipo de trama no numerada

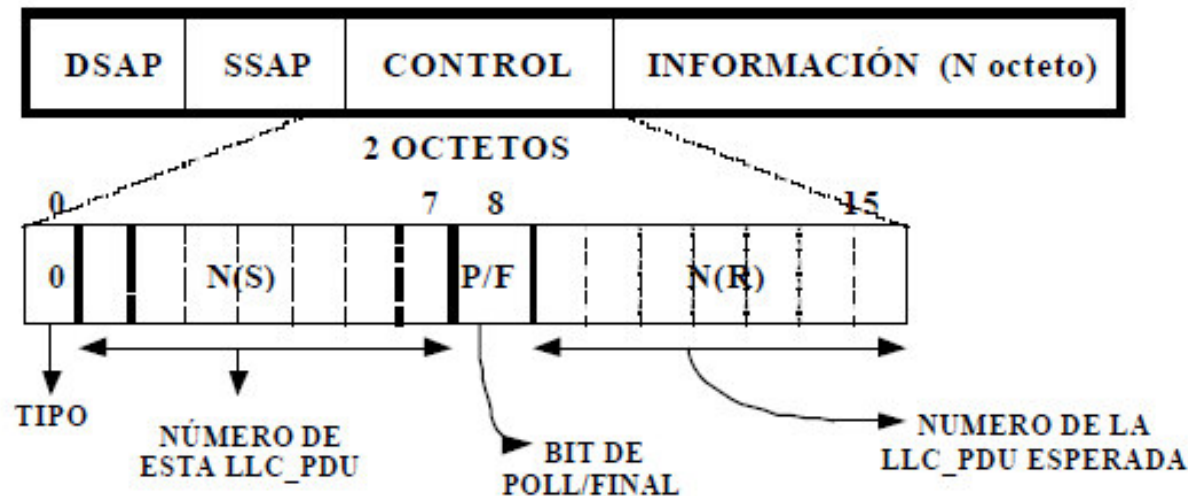


# Trama LLC de Información



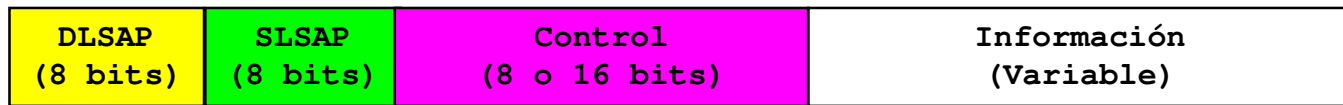
Uso en Tipo II: Son *comandos o respuestas numerados* en fase de transferencia **con conexión**

- Validan *Ix* recibida
- Realizan control de flujo
- Recuperación de errores



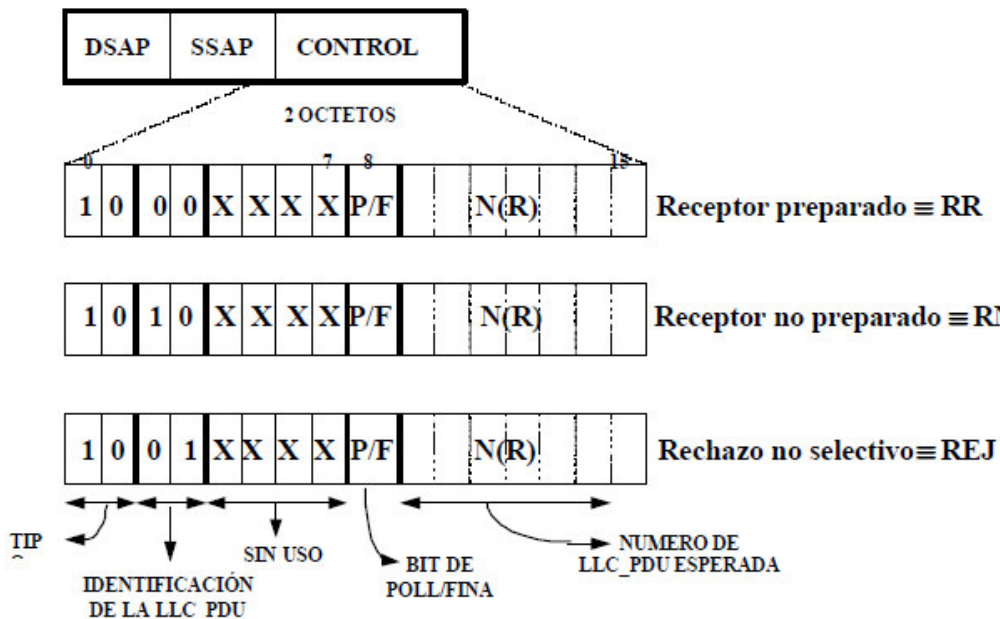


# Trama LLC de Supervisión



**Uso en Tipo II:** Son *comandos o respuestas* **NO** son numerados en fase de transferencia **con conexión**

- Validan Ix recibida
- Realizan control de flujo
- **Recuperación de errores cuando no hay datos para enviar**

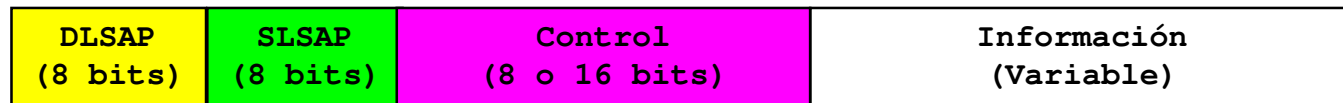


Valida unidades de Dx e indica que el Rx **está** preparado para recibir Dxs

Valida unidades de Dx e indica que el Rx **NO** está preparado para recibir Dxs

Pide retransmisión de tramas con  $N(R)^* \geq N(R)$ , validando a las inferiores

# Trama LLC NO Numerada



Son comandos o respuestas que no tienen nro de orden

- En tipo I o III: Envío de Información y Control (servicios **sin conexión**)
- En tipo II: Fases **previas-posteriores** a la transferencia (servicios **con conexión**)

LLC\_PDU's no numeradas usuales, se codifican en **MM MMM**:

- UI (unnumbered Ix): información no numerada entre entidades (**Tipo I**)
- AC0 y AC1 (Acknowledged Connectionless Ix) o Ix para **no conexión con reconocimiento (Tipo III)**;
- XID (eXchange ID) intercambia **Ix gestión** en **Tipo I, II, III**: tipo de servicio soportado (1, 2, 3), cantidad de Ix en cada LLC\_PDU, tamaño de ventana

Ver más en *El estándar IEEE/802.2 e Control de Enlace Lógico: LLC*

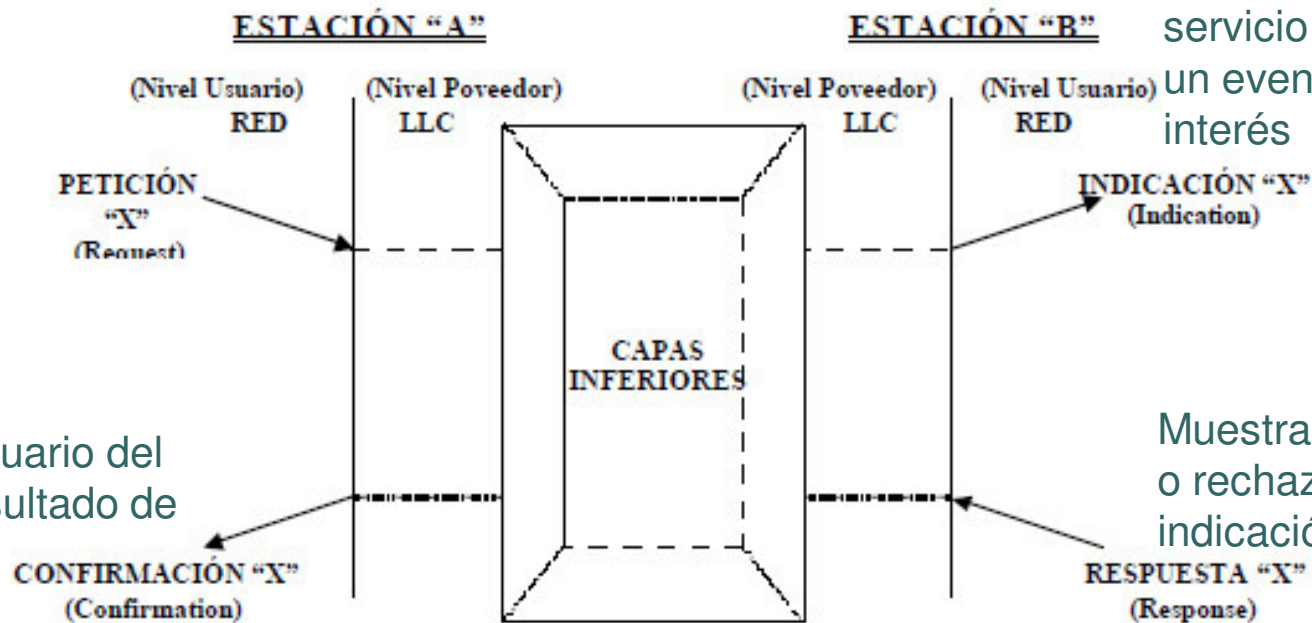
# Servicios

Los **servicios** de una capa (subcapa) son las **capacidades** que ésta ofrece a un usuario de la capa superior

## Primitivas:

Pide  
inicializar un  
servicio

Señalan al usuario del  
servicio el resultado de  
la petición



Señala al usuario del  
servicio que ocurrió  
un evento de su  
interés

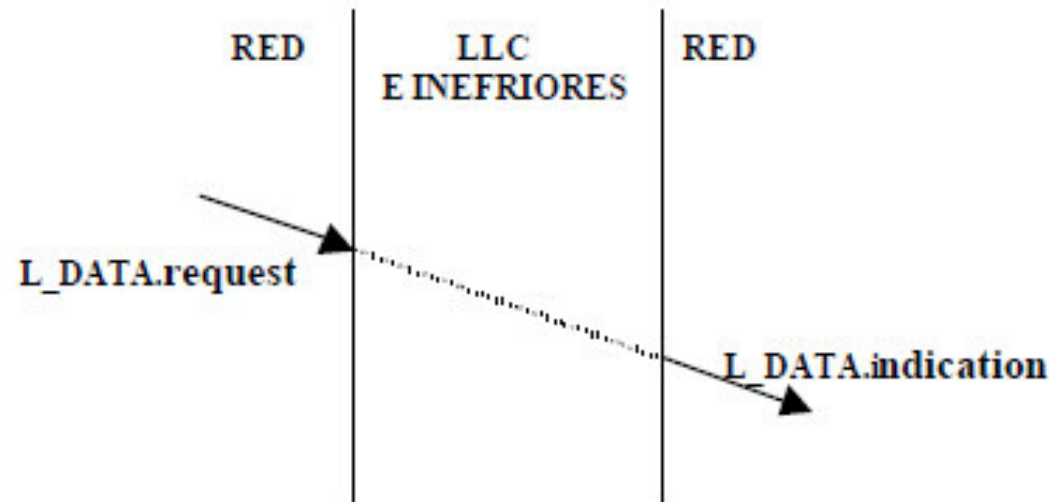
Muestran aceptación  
o rechazo a la  
indicación realizada



# Tipo I: Interacciones

## a) Connectionless-oriented **data transfer**

- ◆ **L\_DATA.request**: Permite pasar datos desde el nivel de red al LLC para su envío.
- ◆ **L\_DATA.indication**: Avisa al nivel de red que el LLC ha recibido una unidad de datos y se los pasa.







# Tipo I: Interacciones

```
DL_UNITDATA request(  
  source_address, //LSAP  
  destination_address, //LSAP  
  data, //LSDU  
  priority  
) //Pasada de C3 a C2-LLC para  
  pedir el envío de un LSDU  
  a uno o varios LSAPs
```

```
DL_UNITDATA indication(  
  source_address, //LSAP  
  destination_address, //LSAP  
  data, //LSDU  
  priority  
) //De C2-LLC a C3 para  
  indicar el arrivo de un  
  LSDU desde una entidad  
  específica
```



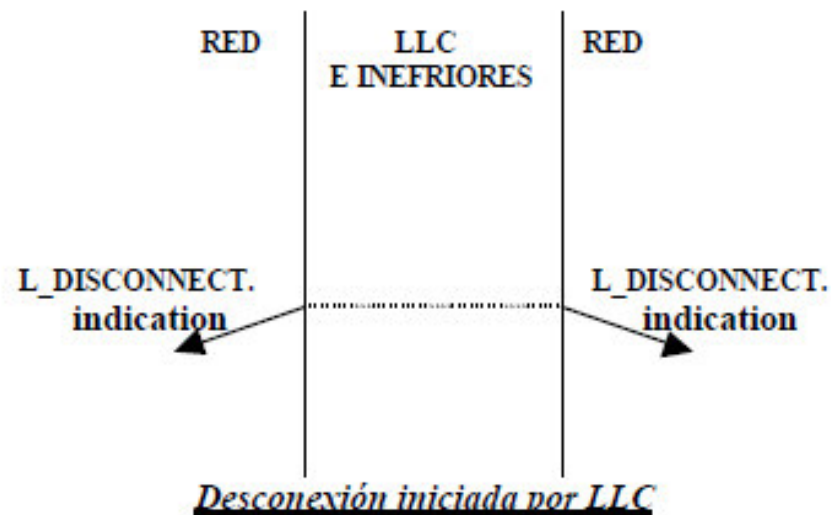
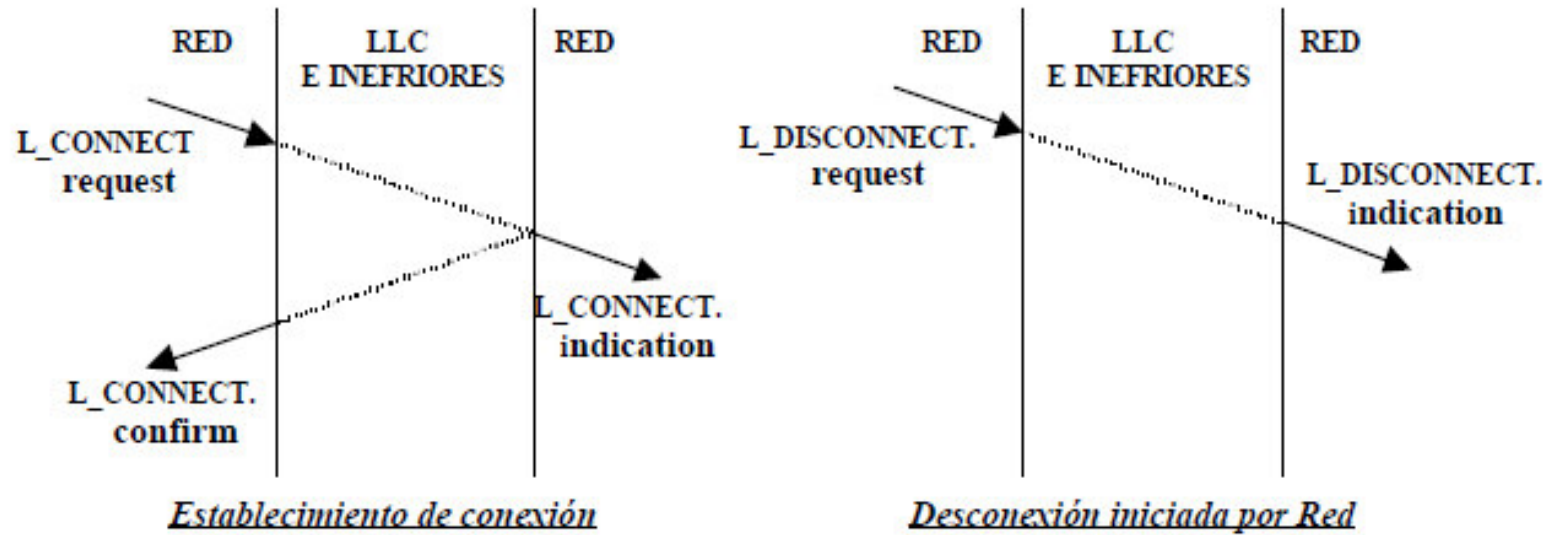
# Tipo II: Interacciones

## Orientado a la conexión

- a) Connection **establishment**
- b) Connection-oriented **data transfer**
- c) Connection **reset**
- d) Connection **termination**
- e) Connection **flow control**



# Tipo II: Interacciones





# Servicios LLC sublayer– MAC

Servicios requeridos a la subcapa MAC para que entidades LLC puedan intercambiar LLC data units (MSDU)

## INTERACCIONES:

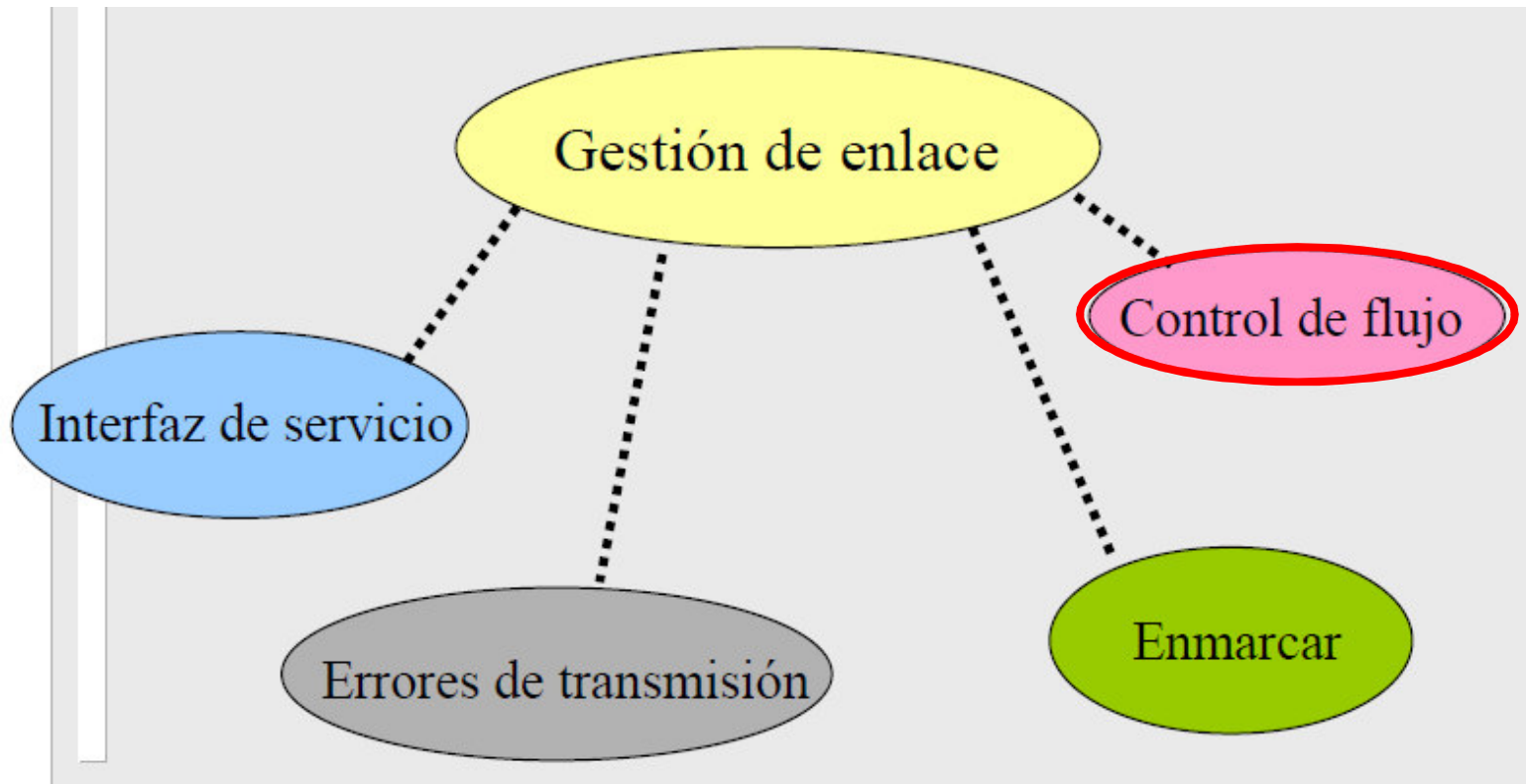
- **MA\_UNITDATA request:** permite transferir MSDU de una entidad LLC a otra(s)

```
MA_UNITDATA request(  
    source_address, //MAC  
    destination_address, //MAC(s)  
    Routing_information, // ruta deseada  
    data, //MSDU que incluye SDAP, SSAP, C e Ix  
    Priority,  
    Service_class  
)// from LLC to MAC sublayer
```

- **MA\_UNITDATA indication:** de subcapa MAC a LLC para indicar el arribo de una trama
- **MA\_UNITDATA\_STATUS indication:** de la subcapa MAC a la LLC para indicar el estado del servicio asociado al previo `MA_UNITDATA request`



# LLC: Funciones





# Control de Flujo

OBJ: regular la velocidad de transmisión de datos para no saturar al Rx.

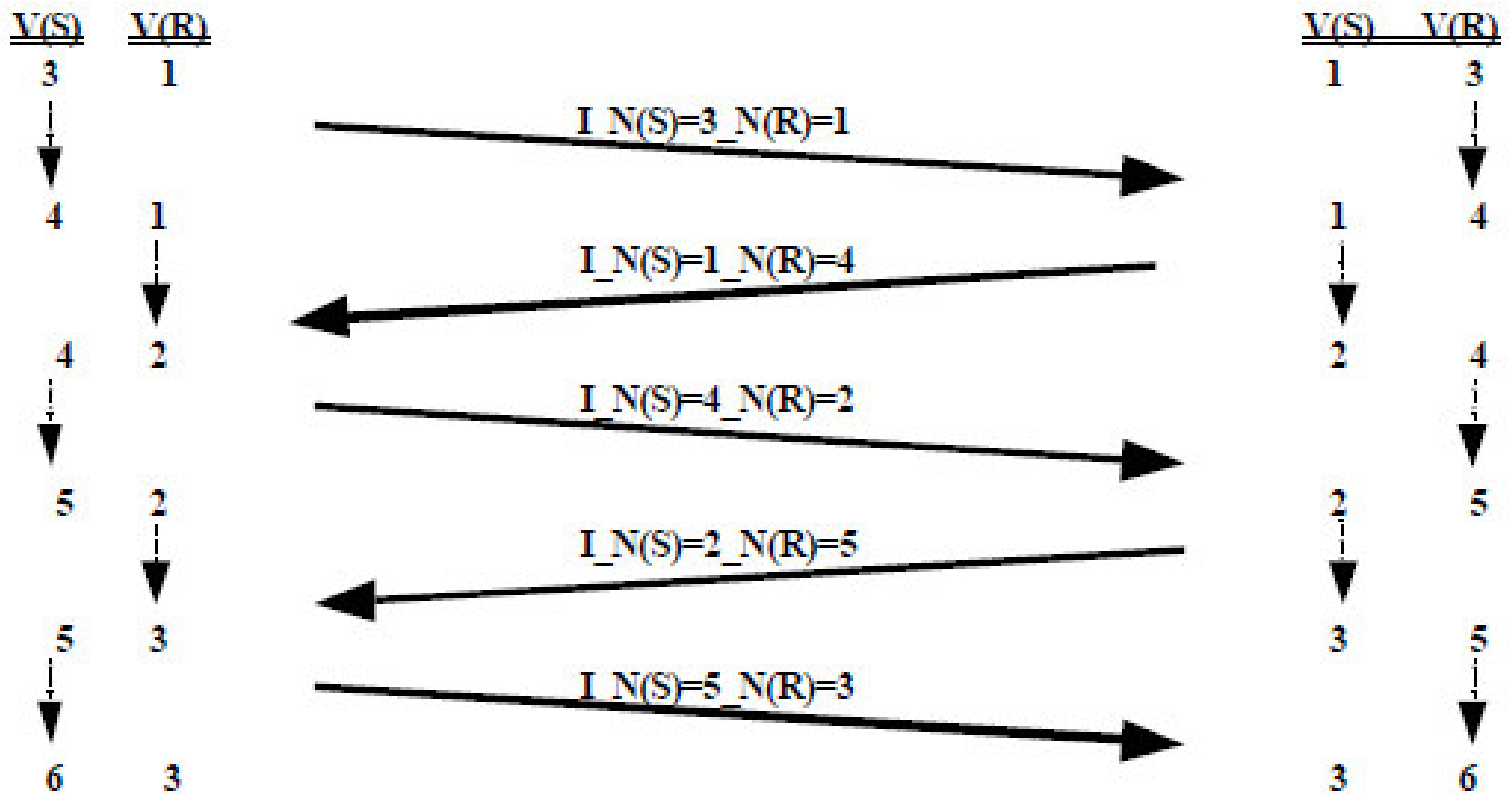
USO: Servicios **con conexión** (Tipo II)

## **Técnicas:**

- Parada y espera
- Ventana deslizante

# Control de Flujo: Tipo II

V(S) contador de emisión; : a quién mandó; V(R) contador de recepción: a quién espera





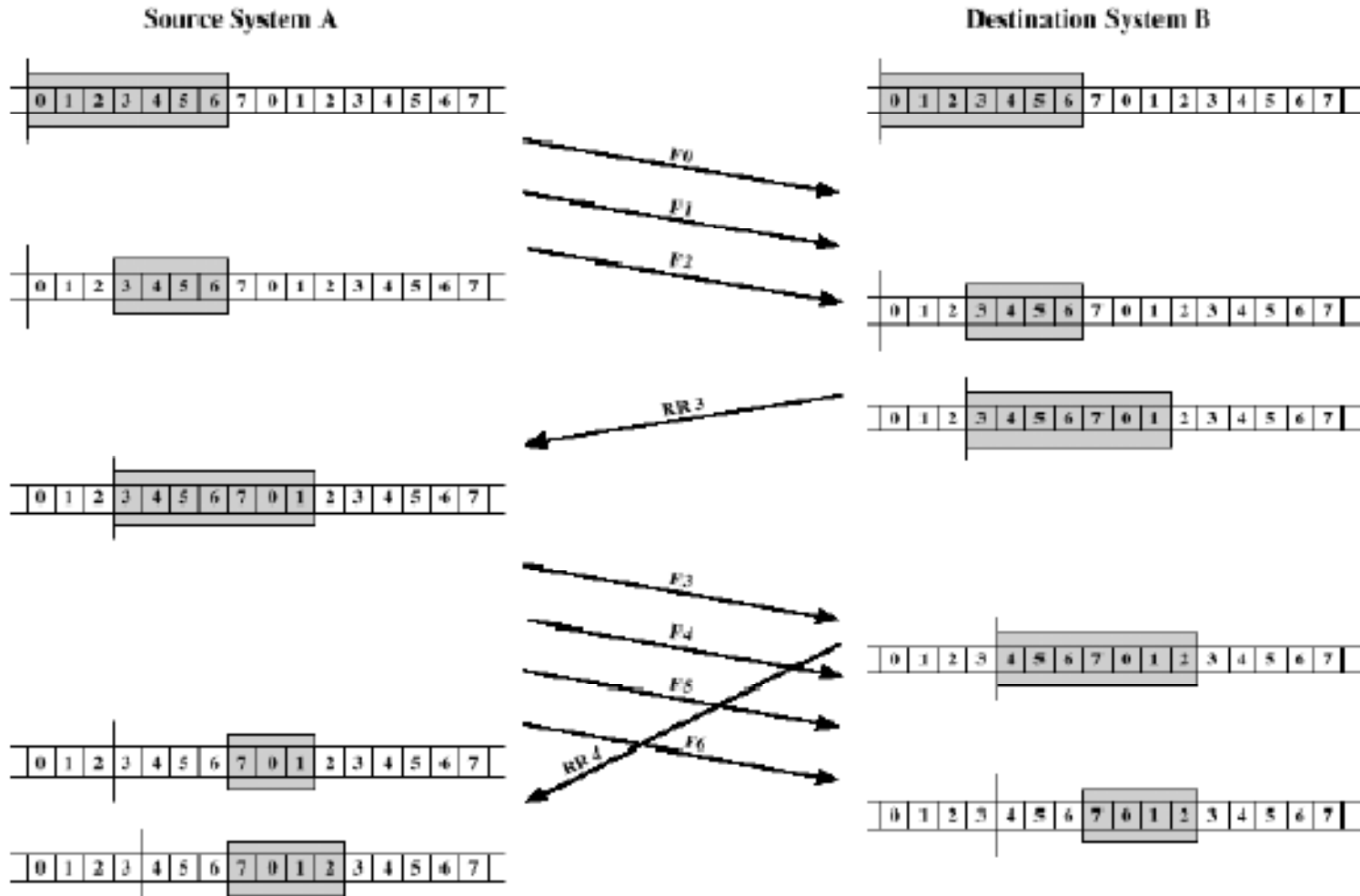
# Control de Flujo: Ventana Deslizante

- *Permite transitar varias tramas simultáneamente a través del enlace.*
- *La estación receptora es capaz de almacenar varias tramas.*
- *Las tramas son nombradas con un número de secuencia.*



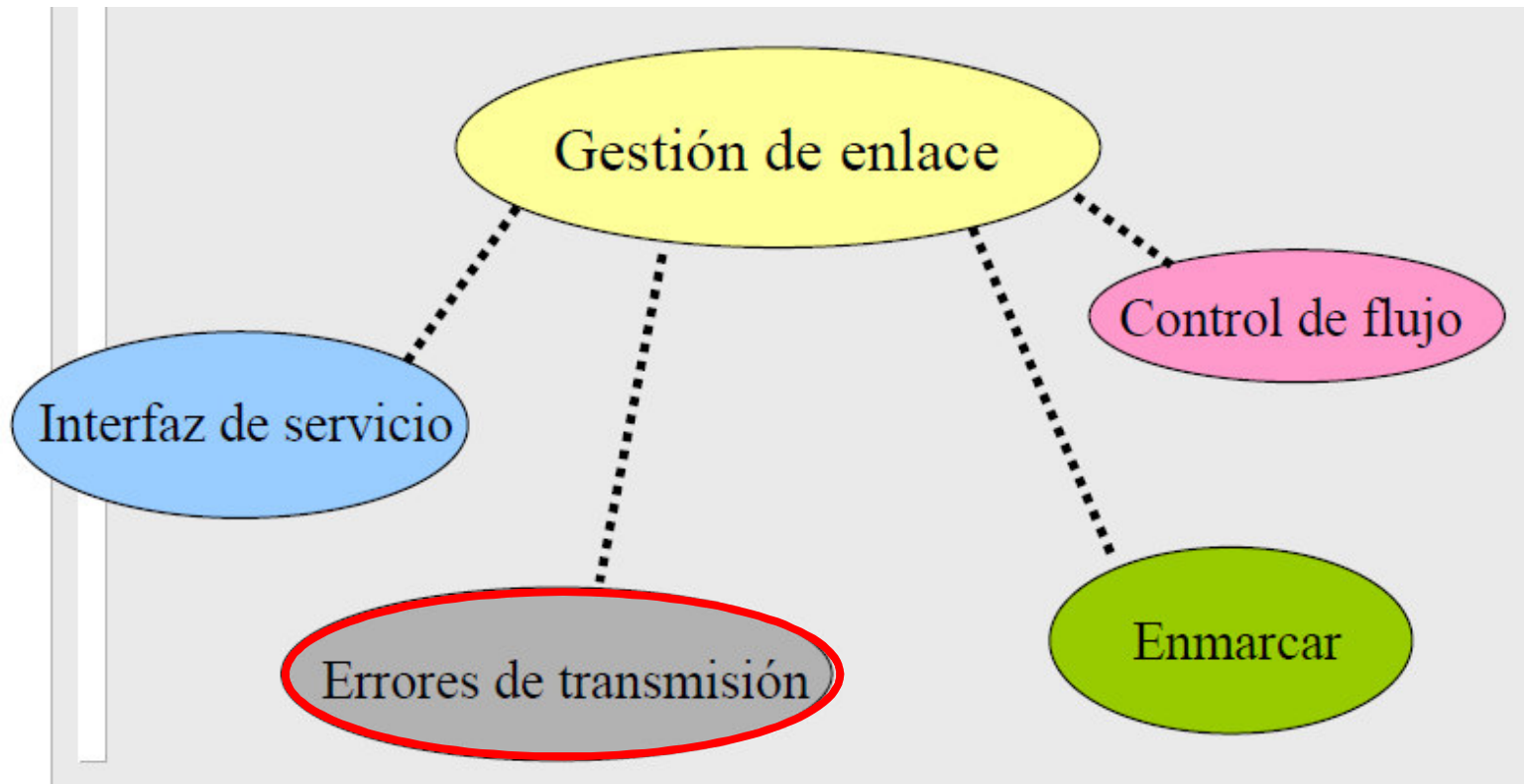


# Control de Flujo: Ventana Deslizante





# LLC: Funciones



# ● ● ● | Control de errores

Técnicas par lograr “entrega confiable”:

- Detección de errores
- Corrección de errores





# Control de errores: detección

- El Rx detecta que hubo  $e$
- Idealmente, pide retransmisión
- Técnicas usuales:
  - Bit de paridad
  - Checksum
  - CRC (Cyclic Redundancy Check)



# 1. Detección por bit de paridad

Ejemplo de generación de un bit de paridad simple:

Queremos enviar la cadena “1110100”:

1° Contamos la cantidad de 1s que hay: 4 1s

2° El número de 1s es par, por tanto, añadimos un bit = 0

3° La cadena enviada es 11101000

○ **Detecta un nro. de errores impares**



# 1. Detección por Checksum

- La trama es tratada como una secuencia de caracteres
- Se suman los caracteres y se envía la suma

H	e	l	l	o		w	o	r	l	d	.
48	65	6C	6C	6F	20	77	6F	72	6C	64	2E
$4865 + 6C6C + 6F20 + 776F + 726C + 642E + \text{carry} = 71FC$											

- Envía menos bits que el “bit de paridad”, pero tampoco detecta errores múltiples



# 1. Suma de Verificación (CRC)

- La idea es hacer que la trama a transmitir sea divisible por un polinomio Generador  $G(x)$ , en caso de que no lo sea, la trama contendrá un error.
- Para esto, el Tx agrega los bits de redundancia al final con la información, de forma tal que la nueva trama sea efectivamente divisible por  $G(x)$ .
- Algunos polinomios  $G(x)$  estándares son:
  - CRC-12 =  $x^{12}+x^{11}+x^3+x^2+x+1$
  - CRC-16 =  $x^{16}+x^{15}+x^2+1$
  - CRC-CCITT =  $x^{16}+x^{12}+x^5+1$



# Suma de Verificación (CRC)

- CRC-12 se usa con caracteres de 6 bits de longitud, CRC-16 y CRC-CCITT con caracteres de 8 bits.
- CRC-16 y CRC-CCITT detectan los siguientes errores:
  - Todos los errores de 1 o dos bits.
  - Todos los errores con un número impar de bits.
  - Todos los errores de “ráfaga” de 16 bits o menos.
  - 99.997% de la ráfagas de errores de 17 bits.
  - 99.998% de las ráfagas de 18 bits o más.





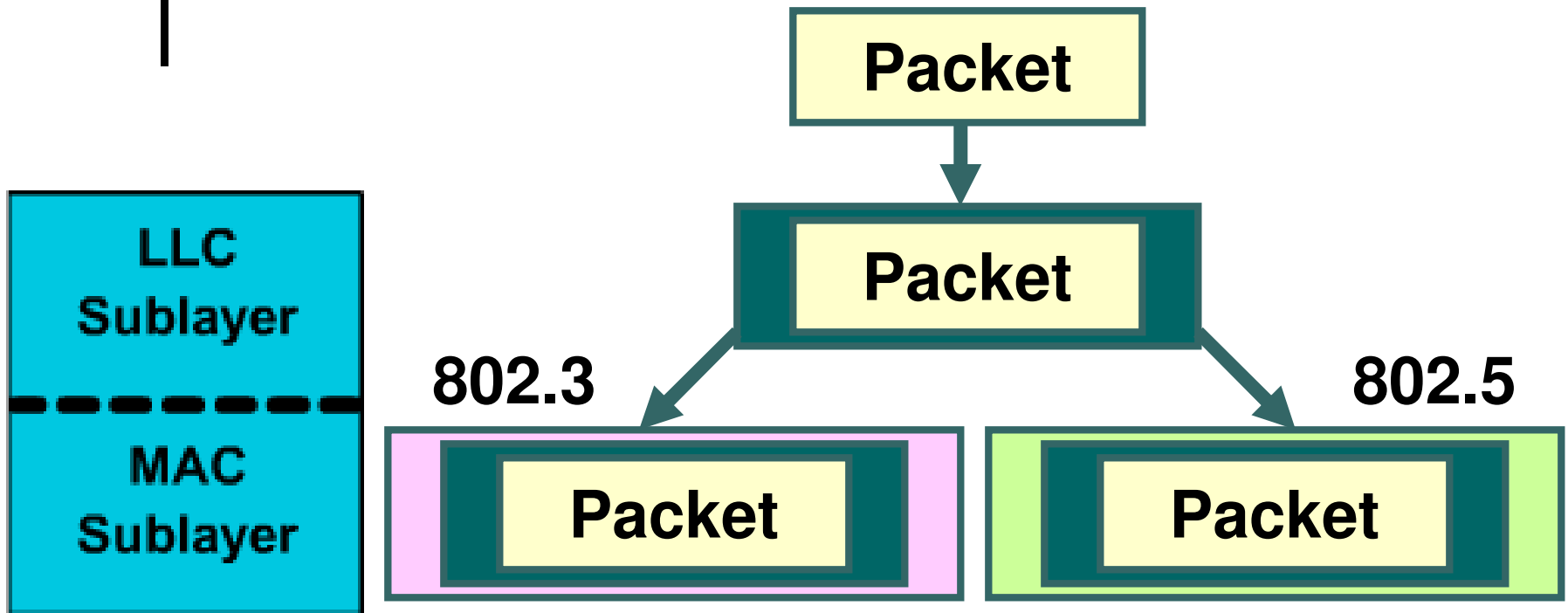
# Control de errores: corrección

- Al detectar errores, el Rx trata de corregirlos sin pedir retransmisión
- Se agrega redundancia
- Técnicas usuales
  - BCH
  - Hamming



Función principal: optimizar el uso del canal

# ● ● ● | Funcionalidades MAC



El paquete (IP) es reempaquetado en la subcapa MAC con  $I_x$  adicional según la tecnología específica.

- Reglas de control de acceso al medio
- Direcciónamiento físico



# Tipo de canales

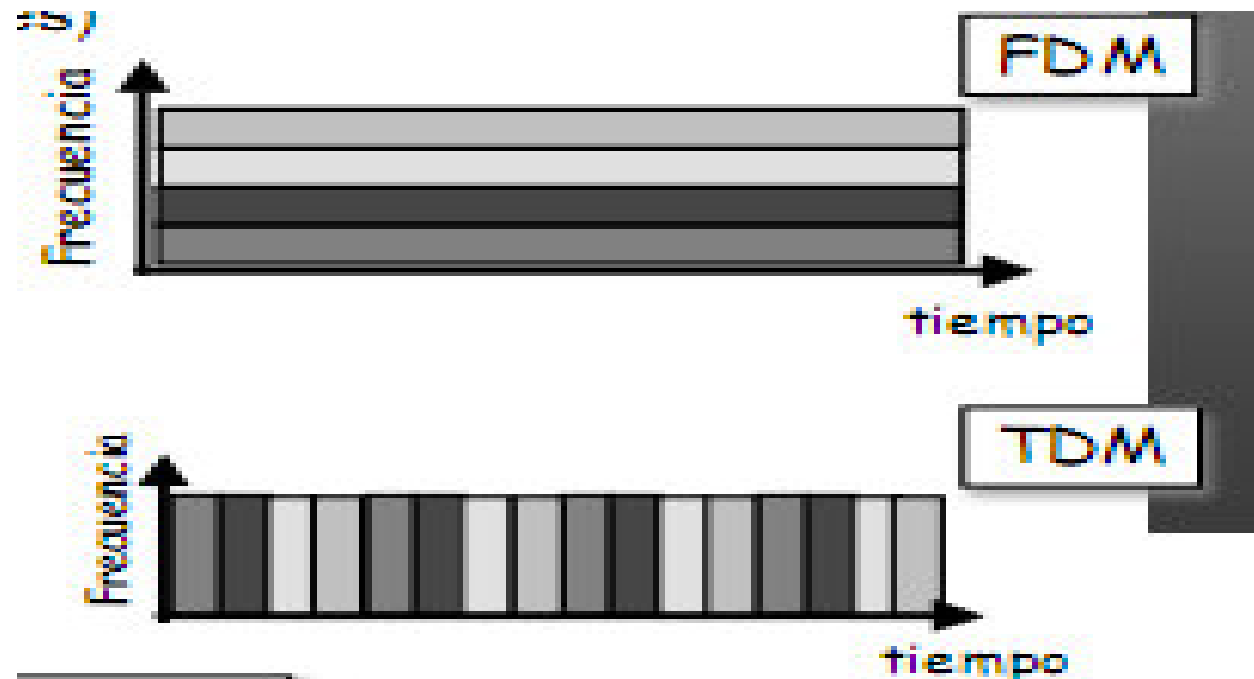
- Canales punto a punto
- Canales de difusión: compartido

# ● ● ● | MAC: Control de Acceso

- Dependiente de la tecnología usada, determina cuando y quién puede transmitir.
- Dos tipos:
  - 1) **Determinístico: “por turnos”.**
    - a) Reparto estático del canal: FDM-TDM
    - b) Paso de testigo: *Token-Ring, FDDI*.
  - 2) **Aleatorio o de Reparto Dinámico: “First come, first serve”.**
    - a) Contienda: *ALOHA*,
    - b) Contienda con escucha: *Ethernet (CSMA/CD)*.

# Determinístico: Particionamiento del canal

Por división de frecuencias (FDMA) Por ranuras de tiempo (TDMA) y



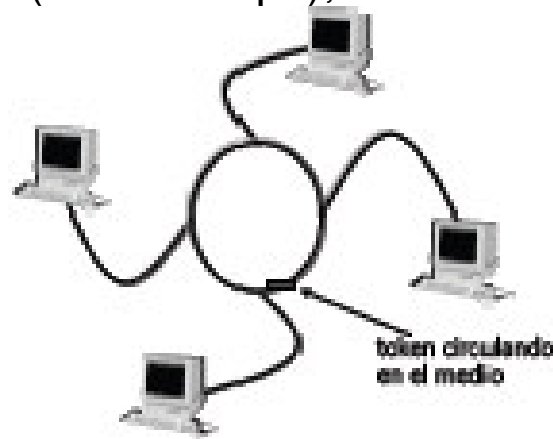
# ● ● ● | Determinístico: Paso de testigo

Token Ring-IEEE 802.5, método de acceso llamado token passing

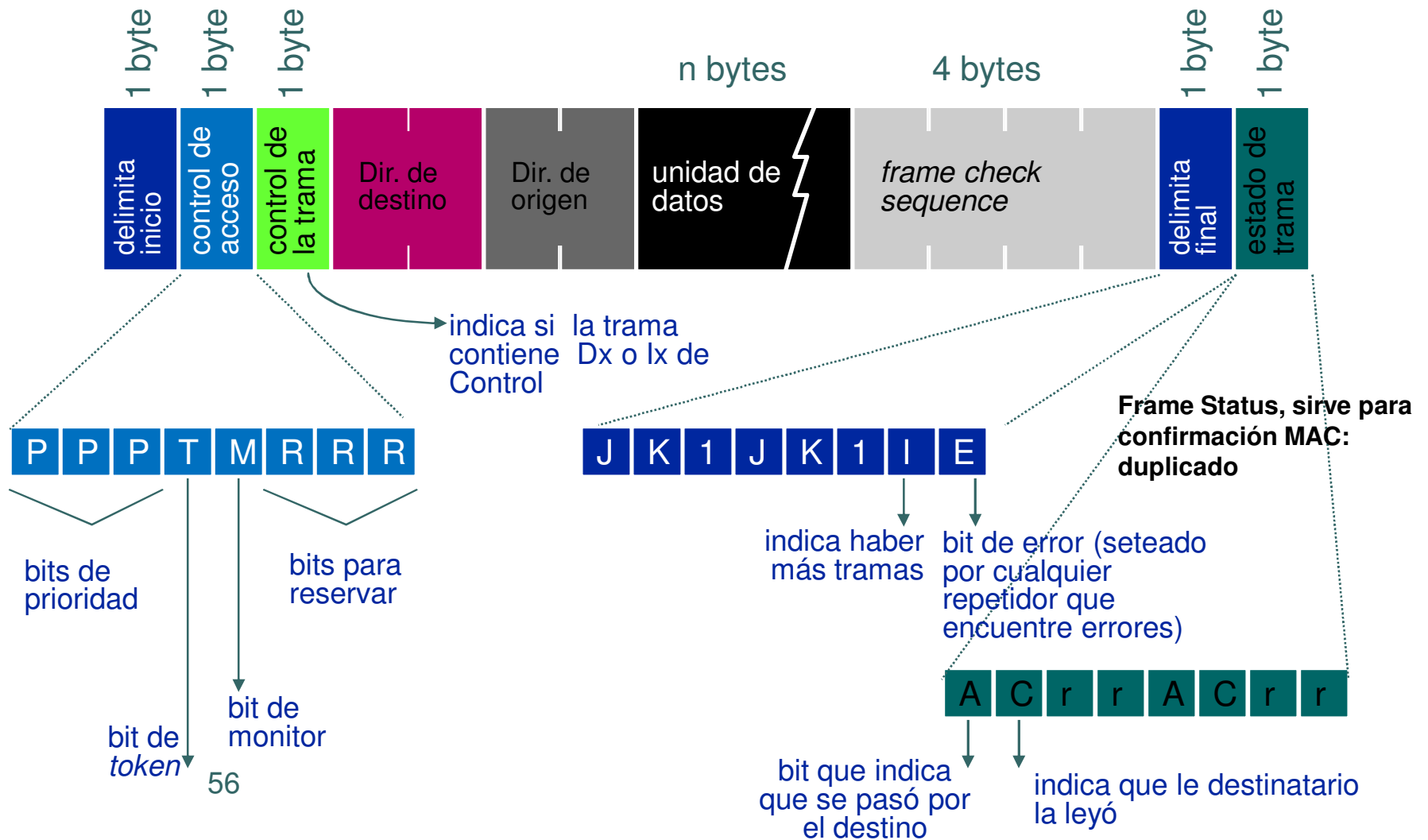
- Una trama especial (token) es transmitida periódicamente
- El token circula por todas las estaciones
- Si una desea transmitir, entonces anexa sus datos al token, si no lo deja seguir
- Si un token con datos pasa por la estación receptora, los datos son retirados y el token sigue circulando

**Problema: overhead de token, una ruptura de anillo deja afuera la red, secuencial**

C. Física: UTP cat. 5 (4 a 16 Mbps), Manchester, perímetro < 360m



# 1.b) Trama *Token Ring*







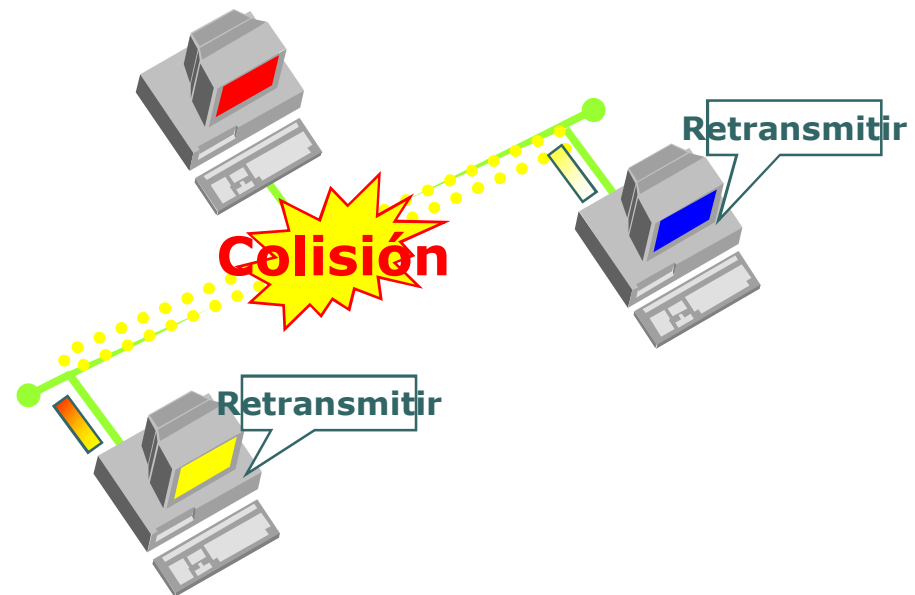
# Protocolos de acceso aleatorio

- Cuando el nodo tiene algo que enviar,
  - Transmitir a tasa máxima  $R$
  - Sin coordinación a priori entre nodos
- Si 2 o más nodos transmiten simultáneamente, puede haber colisiones
  - El protocolo MAC debe especificar cómo detectar las colisiones y como recuperarse
- Ejemplos:
  - ALOHA, slotted ALOHA
  - CSMA, CSMA/CD, CSMA/CA

# Protocolo de acceso aleatorio: Alohanet

U.de Hawai, Norman Abramson, 1970: los terminales compiten por el uso de un sólo canal compartido.

- Conexión entre islas a través de un nodo central,
- Comunicación por radio. Todos los terminales usan la misma frecuencia para comunicarse con el nodo central,
- El nodo central usa otra frecuencia distinta para comunicarse con los terminales.

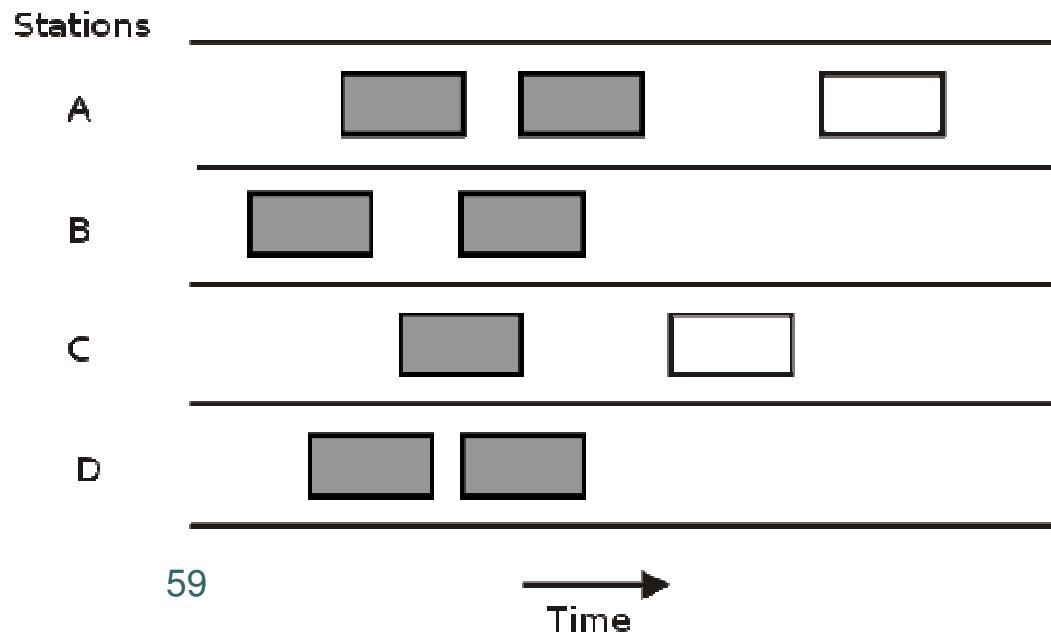




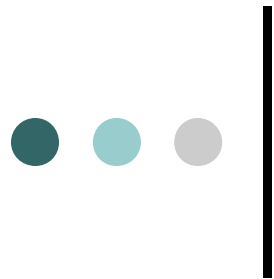
# Alohanet ranurado

1. IF (tengo datos)  
Transmito  $t_{ran}$
2. IF ( $t > t_{ack}$ )  
Retransmito  
THEN (GOTO 1)

Si se produce una colisión, el nodo no recibe información y no envía **señal de confirmación (ACK)**, al vencer el **tiempo de espera de ACK**, retransmite.



Rendimiento del Cx es de 18%,  
37% en caso de ser ranurado



# Protocolo de acceso aleatorio: CSMA (Carrier Sense Multiple Access)

Las estaciones escuchan al canal antes de transmitir para ver si está libre...

- **CSMA persistente-1:**

- Si está **desocupado**, transmite
- Si el canal está **ocupado**, espera hasta que se desocupe (hasta el 50%)

- **CSMA persistente- $p$ :**

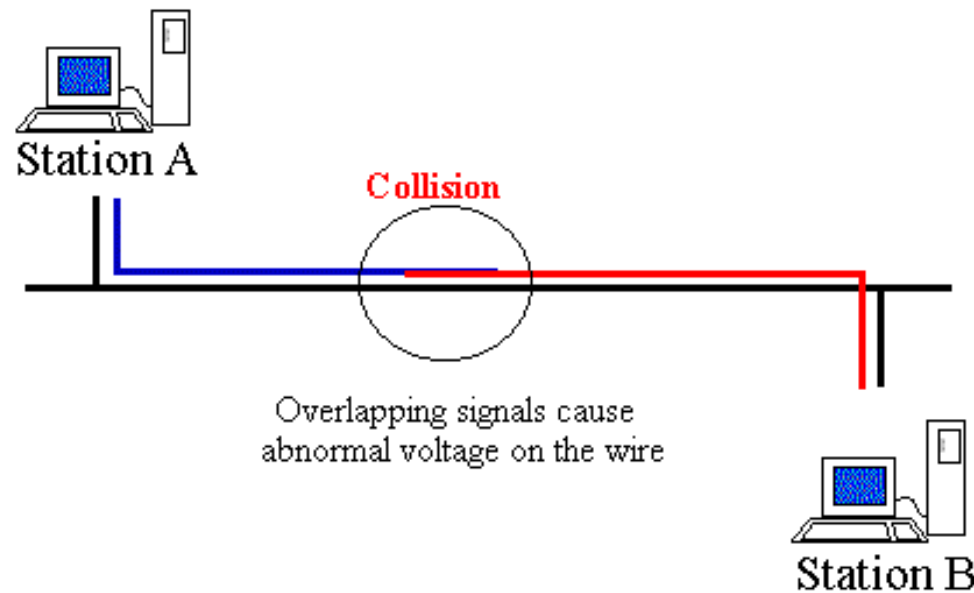
- Si el canal esta **desocupado**, transmite con probabilidad  $p$ .

- **CSMA no persistente:**

- Si el canal está **ocupado**, no espera hasta que se desocupe sino que espera un tiempo aleatorio para volver a escucharlo (hasta 90%).

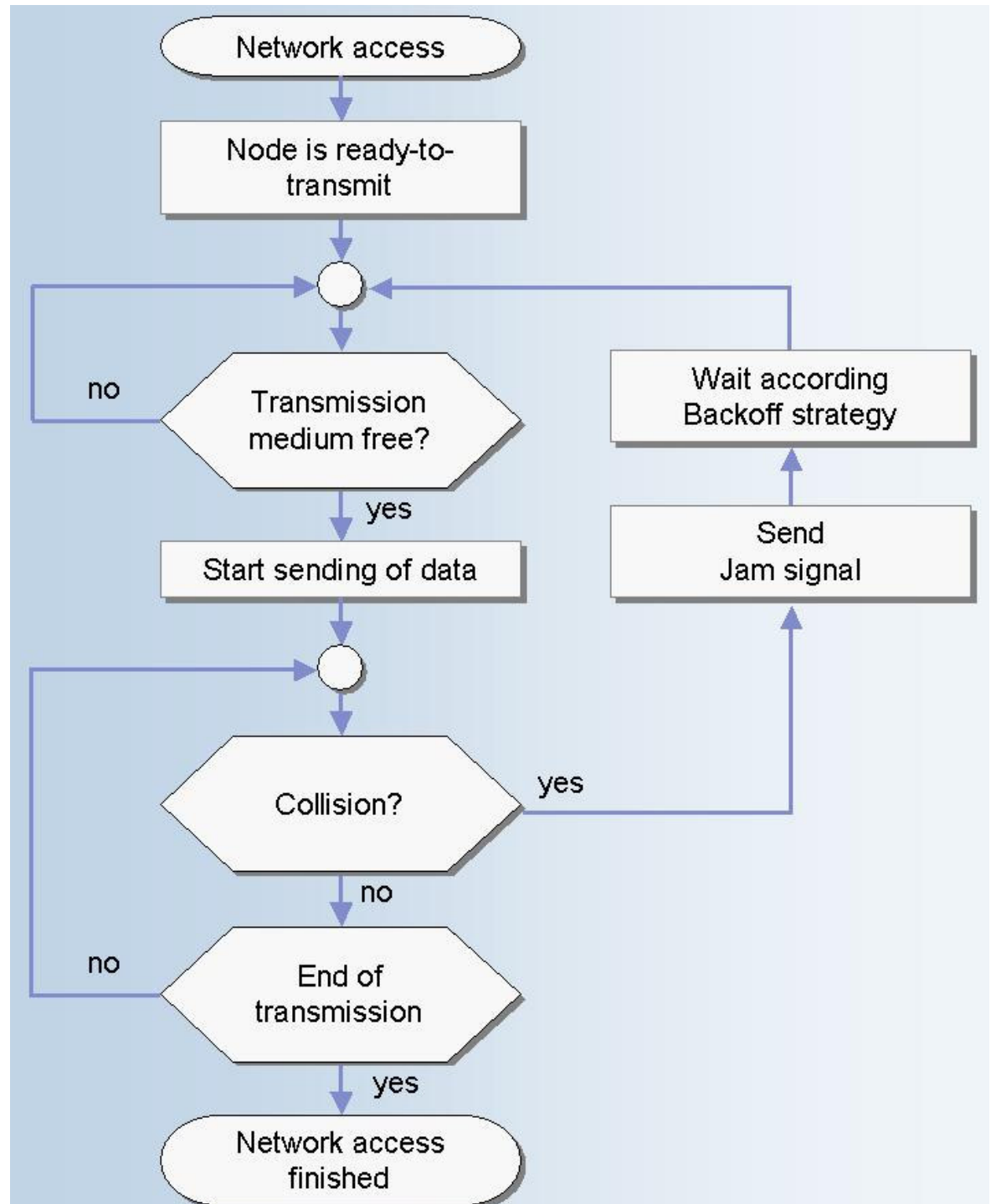
# Protocolo de acceso aleatorio: CSMA/CD collision Detection

- Análogo al CSMA, pero si hay colisión, implementa la generación de una señal de error y acciones a realizar.



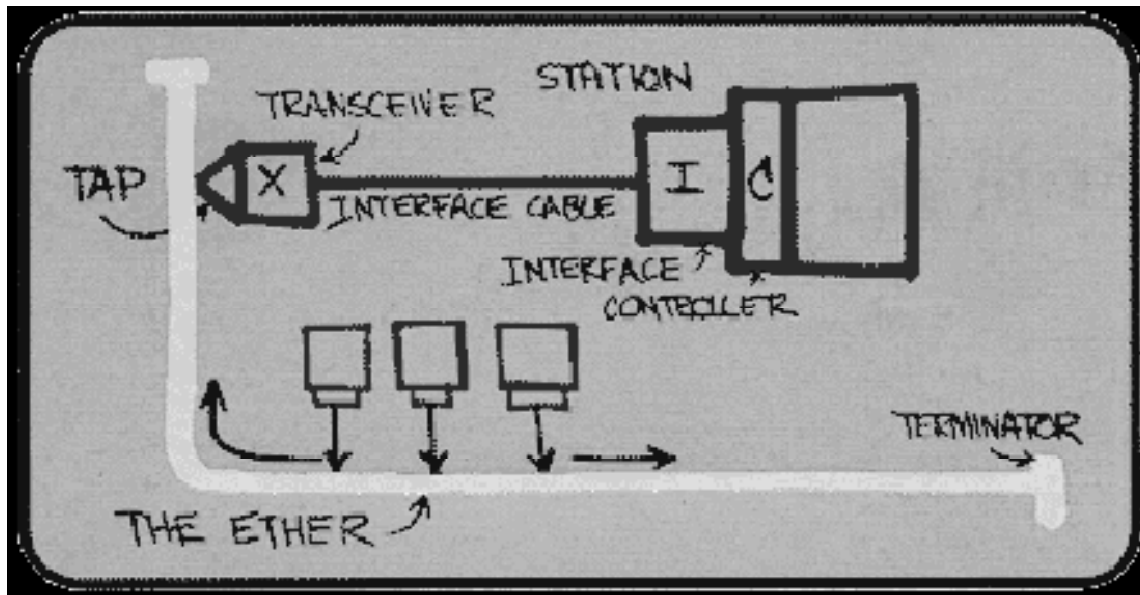


# CSMA/CD



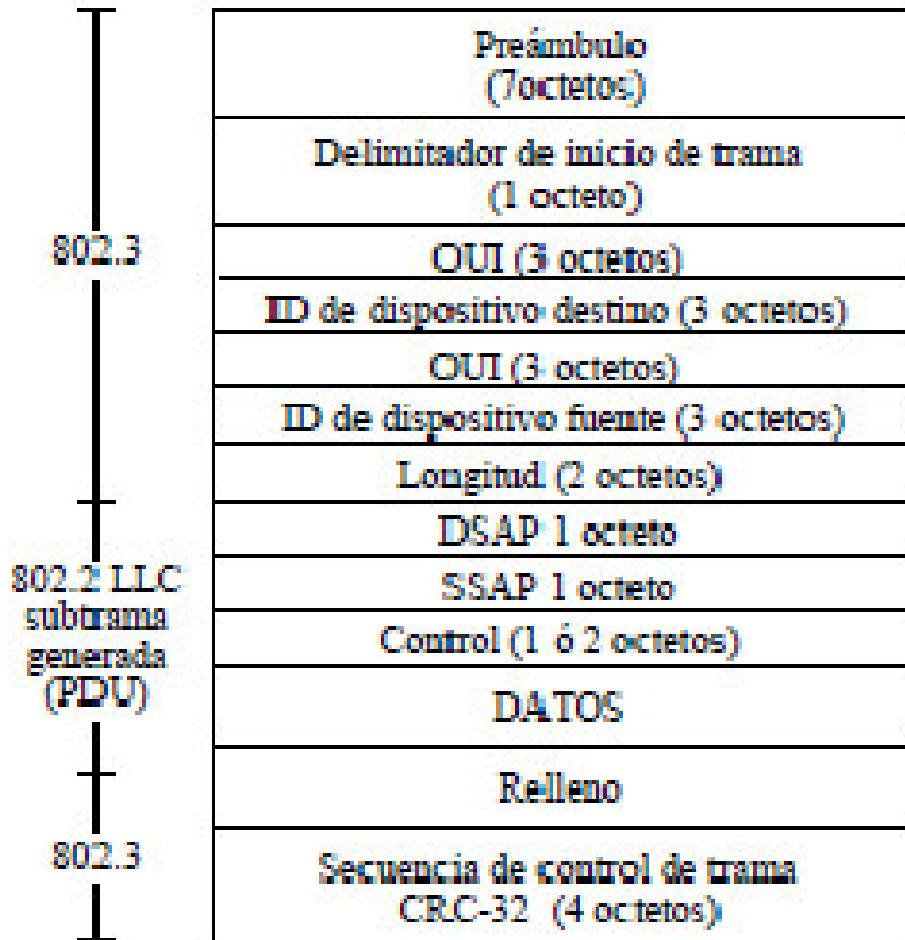
# Ethernet IEEE 802.3

- Bob Metcalfe, 1976
- Método de acceso: CSMA/CD



# Ethernet

## Trama 802.2



**Preámbulo:** Corresponde al patrón de 8 *bytes*, utilizado por el Rx se sincronice (010101.....)

**Delimitador de inicio de trama:** Indica el inicio de la trama. Su valor es 10101011.

**Dirección de destino:** contiene la dirección física (MAC) del equipo destinatario de la trama.

**Dirección de origen:** contiene la dirección MAC de la estación emisora de la trama.

**Longitud:** necesaria para determinar la longitud del campo de datos en los casos que se utiliza un campo de relleno.

**Datos:** Lleva tramas de información del nivel LLC. Su tamaño mínimo es 46 octetos y el tamaño máximo es 1500 octetos, incluyendo el campo de relleno





# Capa física

velocidad	Base	Tecnología
-----------	------	------------

Se distinguen diferentes variantes de tecnología Ethernet según el tipo y el diámetro de los cables utilizados:

10Base2: el cable que se usa es un cable coaxial delgado, llamado *thin Ethernet*.

10Base5: el cable que se usa es un cable coaxial grueso, llamado *thick Ethernet*.

10Base-T: se utilizan dos cables trenzados (la T significa *twisted pair*) y alcanza una velocidad de 10 Mbps.

100Base-FX: permite alcanzar una velocidad de 100 Mbps al usar una fibra óptica multimodo (la F es por *Fiber*).

100Base-TX: es similar al 10Base-T pero con una velocidad 10 veces mayor (100 Mbps).

1000Base-T: utiliza dos pares de cables trenzados de categoría 5 y permite una velocidad de 1 gigabite por segundo.

1000Base-SX: se basa en fibra óptica multimodo y utiliza una longitud de onda corta (la S es por *short*) de 850 nanómetros (770 a 860 nm).

1000Base-LX: se basa en fibra óptica multimodo y utiliza una longitud de onda larga (la L es por *long*) de 1350 nanómetros (1270 a 1355 nm).



## Resumiendo: 802.3/.4/.5

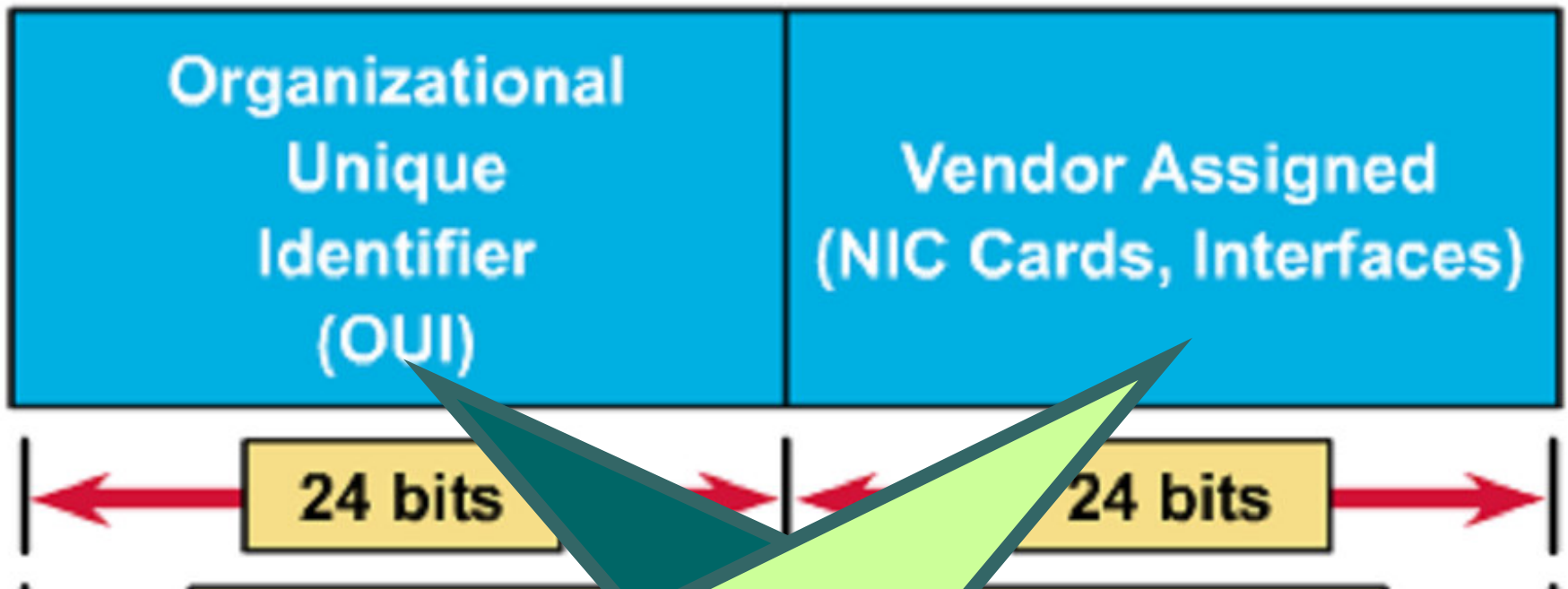
- Ethernet es simple y eficiente con Dx largos, pero si aumenta el tráfico, aumentan las colisiones.
  - Tamaño máximo de paquete de 1518 bytes
  - Acceso Probabilístico.
  - No se debe conectar más de 60 estaciones por segmento
- Token Bus es más elaborada y da la posibilidad de un tráfico sin colisiones y con prioridades
- Token Ring es simple, pero depende del estado y la velocidad de las estaciones. Si una sale de servicio, se cae la comunicación
  - Tiempo de transmisión dividido (prioridades).
  - Cada nodo actúa como repetidor

# ● ● ● | MAC Address



- Every computer has a unique way of identifying itself : MAC address or physical address.
- The physical address is located on the Network Interface Card (NIC).
- MAC addresses have no structure, and are considered flat address spaces.

- ● ● | MAC address format



The remaining six hexadecimal digits comprise the interface serial number.

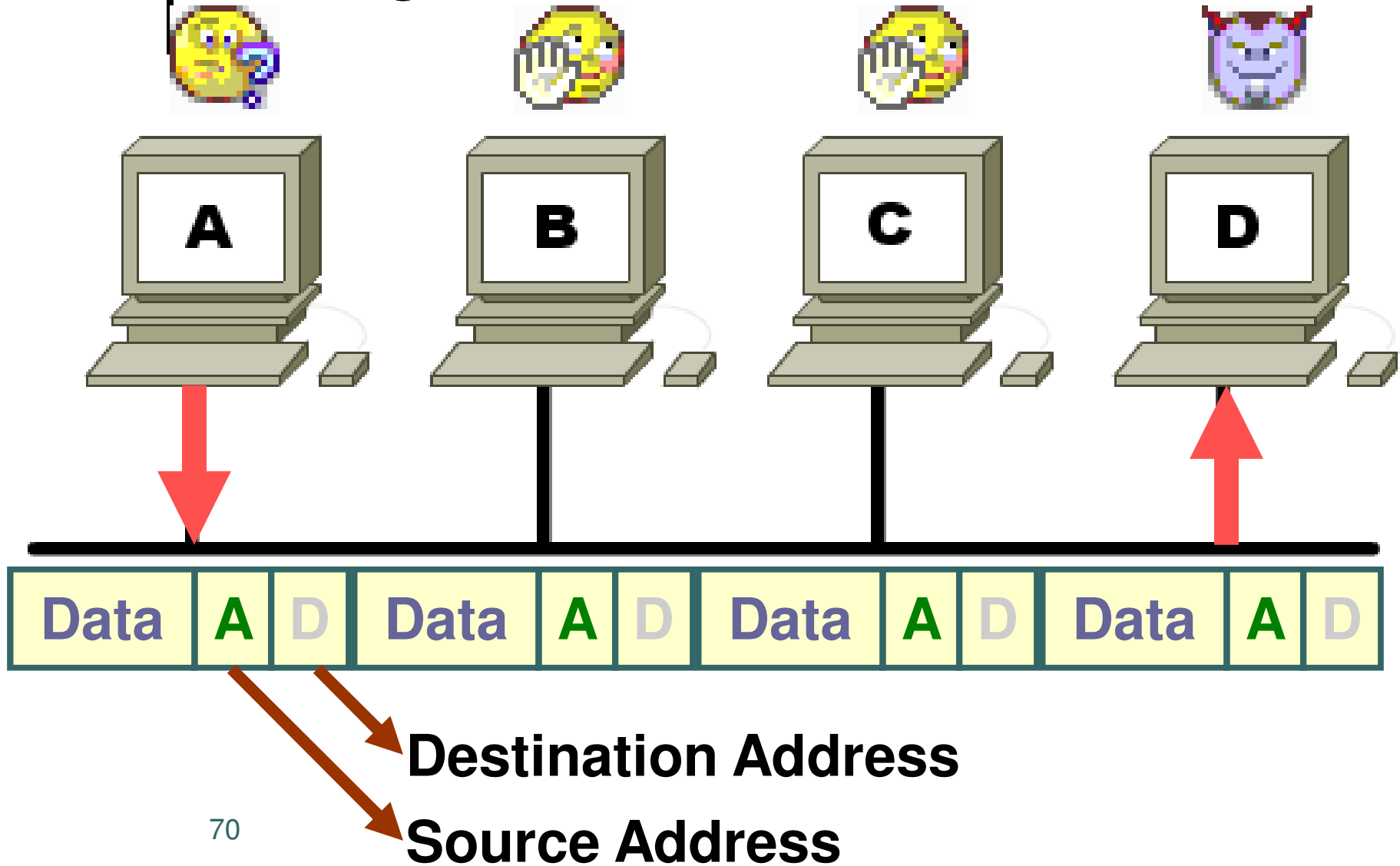
hexadecimal  
ministered  
ntify the  
vendor.



# MAC address

- MAC addresses are sometimes referred to as *burned-in addresses (BIAs)* because they are burned into read-only memory (ROM) and are copied into random-access memory (RAM) when the NIC initializes.
- 0000.0c12.3456 or 00-00-0c-12-34-56.

# Using MAC addresses



# Generic frame format

Field Names					
A	B	C	D	E	F
<b>Start Frame Field</b>	<b>Address Field</b>	<b>Type/ Length Field</b>	<b>Data Field</b>	<b>FCS Field</b>	<b>Stop Frame Field</b>

- There are many different types of frames described by various standards.

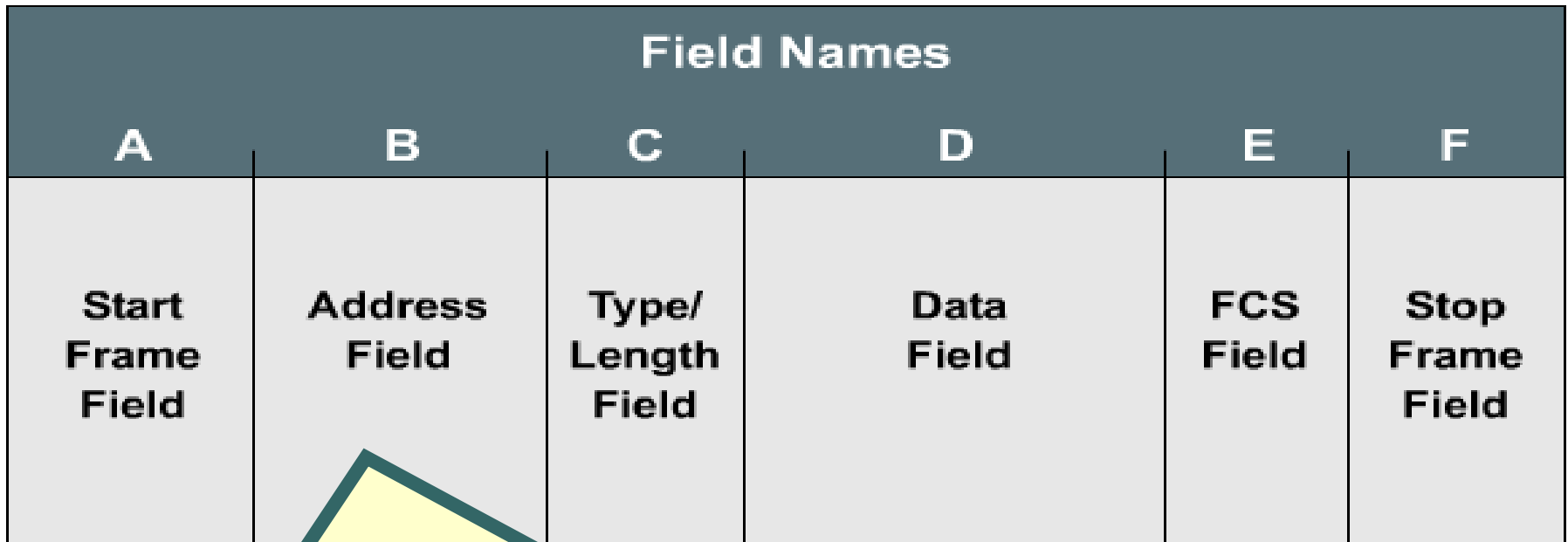
- ● ● | Start frame field

Field Names					
A	B	C	D	E	F
Start Frame Field	Address Field	Type/Length Field	Data Field	FCS Field	Stop Frame Field

- **Start Frame field** tells other devices on the network that a frame is coming down the wire.



# Address field



- **Address field** tiene la dirección MAC del origen y destino

- ● ● | Length/Type field

Field Names					
A	B	C	D	E	F
Start Frame Field	Address Field	Type/Length Field	Data Field	FCS Field	Stop Frame Field

- **Type/Length field** campo opcional usado por algunos protocolos para decir el tipo de datos transportado o la long. de trama

● ● ● | Data field

Field Names					
A	B	C	D	E	F
Start Frame Field	Address Field	Type/Length Field	Data Field	FCS Field	Stop Frame Field

- **Data field** es la información de las capas superiores encapsulada

- ● ● | Stop frame field

Field Names					
A	B	C	D	E	F
Start Frame Field	Address Field	Type/Length Field	Data Field	FCS Field	Stop Frame Field

- **Stop Frame**, campo opcional usado cuando la long. de trama no está especificada en Type/Length

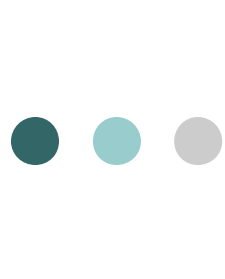
# Rec. LAN IEEE 802

<b>L L C</b>	<b>IEEE 802.2</b>									
	<b>Servicio no orientado a conexión no confirmado</b>									
	<b>Servicio en modo de conexión</b>									
<b>M A C</b>	<b>CSMA/CD</b>		<b>Bus con paso de testigo</b>	<b>Rotación circular con prioridad</b>	<b>Anillo con paso de testigo</b>	<b>CSMA: sondeo</b>				
	<b>IEEE 802.3 ETHERNET</b>	Cable coaxial de banda base: 10 Mbps.  Par trenzado no apantallado 10, 100 Mbps.  Par trenzado apantallado 100 Mbps.  Fibra óptica: 10 Mbps.	<b>IEEE 802.4 TOKEN BUS</b>	Cable coaxial de banda ancha: 1.5, 10 Mbps.  Fibra óptica: 5, 10, 20 Mbps.	<b>IEEE 802.12 100VG-AnyLAN</b>	Par trenzado no apantallado 100 Mbps.	<b>IEEE 802.5 TOKEN RING</b>	Par trenzado apantallado 4, 16 Mbps.  Par trenzado no apantallado 4 Mbps.	<b>IEEE 802.11 WLAN</b>	Infrarrojos: 1, 2 Mbps  Espectro expandido 1, 2 Mbps.
<b>F I S I C A</b>										

**Topología en Bus/árbol estrella**

**Topología en anillo**

**Inalámbricos**



# Ix Adicional: Funciones LLC

T1: unacknowledge connectionless

T2: connection-mode

```
DL_UNITDATA request (  
  source_address, //LSAP  
  destination_address, //LSAP  
  data, //LSDU  
  priority  
) //Pasada de C3 a C2-LLC  
  para pedir el envío de un  
  LSDU a uno o varios LSAPs
```

```
DL_UNITDATA indication(  
  source_address, //LSAP  
  destination_address, //LSAP  
  data, //LSDU  
  priority  
) //De C2-LLC a C3 para  
  indicar el arribo de un  
  LSDU desde una entidad  
  específica
```

```
DL_CONNECT request (  
  source_address, //LSAP  
  destination_address, //LSAP  
  priority  
) //Pasada de C3 a C2-LLC  
  cuando una entidad quiere  
  conexión lógica de  
  enlace a un LSAP
```

```
DL_CONNECT indication(  
  source_address, //LSAP  
  destination_address, //LSAP  
  priority  
) //De C2-LLC a C3 indicando  
  que se solicitó una  
  conexión
```



# Ix Adicional: Funciones LLC

T1: unacknowledge connectionless

T2: connection-mode

...

```
DL_CONNECT response(  
    source_address, //LSAP  
    destination_address, //LSAP  
    priority  
)//De C3 a C2-LLC indicando  
    aceptación de la conexión  
    solicitada
```

```
DL_CONNECT confirm(  
    source_address, //LSAP  
    destination_address, //LSAP  
    priority  
)//De C2-LLC a C3 p/llevar  
    resultados del proceso
```



# Funciones

T1: unacknowledge connectionless

T2: connection-mode

...

```
DL_DATA request(  
    source_address, //LSAP  
    destination_address, //LSAP  
    data  
)//De C3 a C2-LLC con una LSDU a  
    ser TX sobre una conexión. No  
    hay prioridad porq es = en  
    toda la conexión  
DL_DATA indication(  
    source_address, //LSAP  
    destination_address, //LSAP  
    data  
)//De LLC a red para indicar  
    arrivo de LSDU de una entidad  
    de red espec. En una conexión  
    espec.
```





# Funciones

T1: unacknowledge connectionless

T2: connection-mode

...

```
DL_DISCONNECT request(  
  source_address, //LSAP  
  destination_address, //LSAP  
) //De C3 a C2-LLC cuando se  
  quiere terminar la conexión
```

```
DL_DISCONNECT indication(  
  source_address, //LSAP  
  destination_address, //LSAP  
  data  
) //De LLC a red para indicar  
  final de conexión.
```