

# Índice

<u>Paquete IP(v4).....</u>	2
<u>ICMP(v4).....</u>	3
<u>ARP.....</u>	3
<u>DHCP.....</u>	3
<u>Paquete IP(v6): Cabeceras de extensión.....</u>	4
<u>ICMPv6: Errores y mensajes básicos.....</u>	5
<u>ICMPv6: Neighbor Discovery.....</u>	6
<u>Ethernet y ATM sobre ADSL (AAL5).....</u>	7
<u>Tramas MAC.....</u>	8
<u>Esquemas con/sin RTS y CTS.....</u>	9
<u>Erlang (no es encapsulado, pero para acceso rápido).....</u>	10
<u>Encapsulados.....</u>	11

# Funcionalidad: PDU

## PDU: DATAGRAMA IP

### - NO ORIENTADO A CONEXIÓN: QoS Best effort

Diferenciación CIRCUITO VIRTUAL  
(recordatorio: Redes Computadores)

- No hay conexión extremo a extremo (sin establecimiento, sin información de estado...)
- Cada paquete tratado de forma individual (encaminamiento según dirección destino)
- No hay garantías de entrega
- Tamaño máximo / mínimo =  $65535 / 28$  (20 de cabecera completa + 8 de fragmento mínimo)
- Tamaño mínimo MTU recomendable = 576 bytes
- Tamaño variable de cabecera: Opciones

32 bits



### Tipo de Servicio

Precedencia | D | T | R | O | O

Bits 0-2 Prioridad  
Bits 3 Retardo 0 – normal, 1 – bajo  
Bits 4 Caudal 0 – normal, 1 – alto  
Bits 5 Fiabilidad 0 – normal, 1 – alta  
Bits 6-7 Reservado

### Formato de Opciones

Byte Tipo | Byte Longitud | Byte Datos

CF | Clase | Número

### Campos en octeto tipo de opción

Bits 0 Flag de copia 0 – No copiar, 1 – copiar  
Bits 1-2 Clase de opción 00 – control  
01 – reservado  
10 – debug y medida  
11 – reservado  
Bits 3-7 Número de opción (en tabla siguiente)

### Opciones en IP

Clase	Número	Longitud	Descripción
0	0	1	Fin de lista de opciones
0	1	1	No operación
0	2	11	Seguridad
0	3	variable	Encaminamiento fuente ( <i>loose</i> )
0	9	variable	Encaminamiento fuente ( <i>estricto</i> )
0	7	variable	Registro de ruta
2	4	variable	Internet Timestamp

Campo	bits	Descripción
Versión	4	Número de versión, incluido para permitir la evolución del protocolo. Actualmente es la versión 4.
Longitud de cabecera	4	Longitud de la cabecera del datagrama en palabras de 32 bits. La longitud mínima es 5. Si cabecera contiene opciones, estas deben ser múltiplo de 32 bits, rellenándose con ceros los bytes no usados ( <i>padding</i> ). ⇒ LONGITUD VARIABLE
Tipo de servicio	8	Tipo de servicio deseado.
Longitud total	16	Longitud total del datagrama en octetos.
Número de identificación	16	Permite al destinatario identificar los datagramas. Cada datagrama lleva un identificador diferente, salvo si son fragmentos de un único datagrama
Flags		
<i>Don't Fragment</i>	1	si tiene el valor 1, no puede ser fragmentado.
<i>More Fragment</i>	1	indica que es un datagrama fragmentado. Todos los fragmentos tienen este bit a 1, excepto el último fragmento.
Offset de fragmento	13	Indica la posición del fragmento medida en unidades de 8 bytes (64 bits): En un datagrama, todos los fragmentos, a excepción del último, deben contener un campo de datos múltiplo de 64 bits
Tiempo de vida (TTL)	8	Evita la circulación indefinida de un datagrama. Es decrementado por los router y cuando llega a cero es descartado.
Protocolo	8	Indica el protocolo de nivel inmediatamente superior transportado.
Checksum de cabecera	16	Posibilita la detección de errores en la cabecera del datagrama (no en los datos)
Direcciones (emisor/receptor)	32	Indican las direcciones de la fuente y el destino.
Opciones + Padding	var.	Usado para codificar opciones pedidas por el emisor. Caso de que la longitud de las opciones no fuera múltiplo de 32 bits, se llenan con ceros los bits no usados ( <i>padding</i> ).

### Números de Protocolo Asignados

- 1 Internet Control Message Protocol (ICMP)
- 3 Gateway-to-Gateway Protocol
- 6 Transmission Control Protocol (TCP)
- 8 Exterior Gateway Protocol (EGP)
- 11 Network Voice Protocol
- 17 User Datagram Protocol (UDP)

# Protocolos de CONTROL - ICMP

0	8	16	31
Type	Code	Checksum	
Unused			

IP Header + 64 bits of original datagram

#### (a) Destination Unreachable; Time Exceeded; Source Quench

### **Paquete que ocasionó el error**

0	8	16	31
Type	Code	Checksum	
Pointer	Unused		
IP Header + 64 bits of original datagram			

**(b) Parameter Problem**

Type	Code	Checksum
Gateway Internet Address		
IP Header + 64 bits of original datagram		

**(c) Redirect**

Type	Code	Checksum
Identifier		Sequence Number
Optional data		

**(d) Echo, Echo Reply**

0	8	16	31
Type	Code	Checksum	
Identifier		Sequence Number	
Originate Timestamp			

**(e) Timestamp**

0	8	16	31
Type	Code	Checksum	
Identifier		Sequence Number	
		Originate Timestamp	
		Receive Timestamp	
		Transmit Timestamp	

**(f) Timestamp Reply**

0	8	16	31
Type	Code	Checksum	
Identifier		Sequence	Number

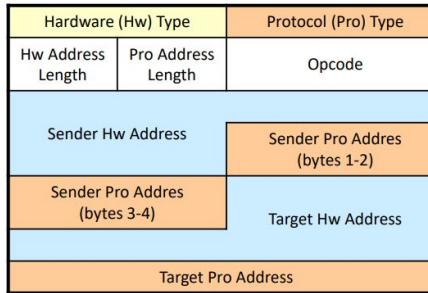
**(g) Address Mask Request**

0	8	16	31
Type	Code	Checksum	
Identifier		Sequence Number	
Address Mask			

**(h) Address Mask Reply**

# Protocolos de CONTROL - ARP

- ARP: *Address Resolution Protocol*



Hw type	Hardware Type
1	Ethernet (10 Mbps)
6	IEEE 802 Networks
7	ARCNET
15	Frame Relay
16	Asynchronous Transfer Mode (ATM)
17	HDLC
18	Fibre Channel
19	Asynchronous Transfer Mode (ATM)
20	Serial Line

## Protocolos de CONTROL – DHCP

- **DHCP (Dynamic Host Configuration Protocol)**

- BROADCAST**

  1. Cliente DHCP → Servidor DHCP: **DhcpDiscover**.
    - Cliente puede especificar preferencias (*lease time*, @IP...)

**BROADCAST**

  2. Un servidor DHCP → cliente **DhcpOffer**
    - Indica @IP disponible: campo 'yiaddr'
    - Todavia no hay acuerdo !!!
  3. Client recibe 1 o + DhcpOffer:
    - Selección de uno: toma @IP del servidor

**BROADCAST**

  4. Cliente → Servidor DHCP elegido: **DhcpRequest**
    - Opción DHCP *identifier*: @IP antes guardada (del servidor)

**BROADCAST**

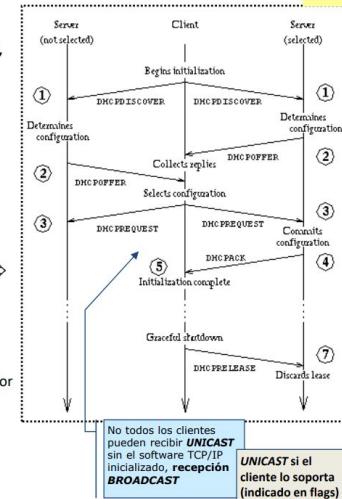
  5. Servidor verifica ¿DCHP *Identifier* = su @IP?: sí ⇒
    - **DhcpAck** (@IP disponible)
    - **DhcpNack** (@IP en uso)

**UNICAST**

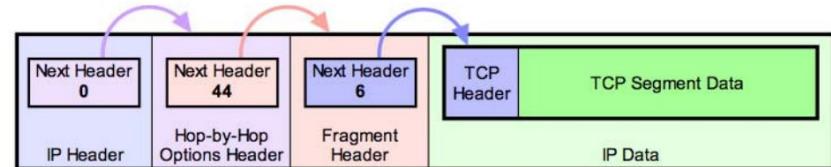
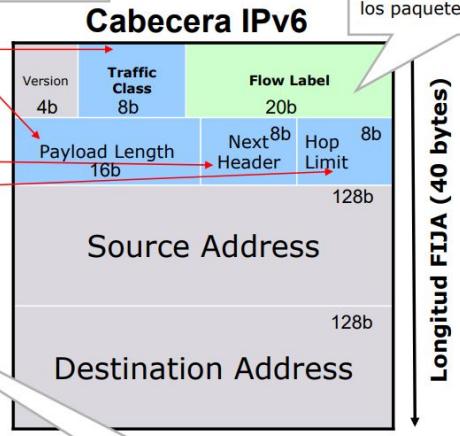
  6. Cliente recibe
    - DhcpAck: puede usar @IP
      - Si hay algún problema con esa IP: Cliente → Servidor DHCP **DhcpDecline** (volver a 1)
    - DhcpNak: volver al paso 1

**UNICAST**

  7. Cliente → Servidor: **DhcpRelease**
    1. Liberar la @IP antes del fin del "lease time"

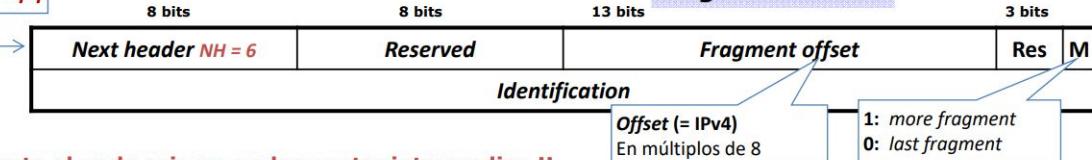


# PDU. Cabeceras de extensión



- Cabeceras de extensión - ejemplo

NH = 44



Fragmenta el nodo origen, no los router intermedios !!

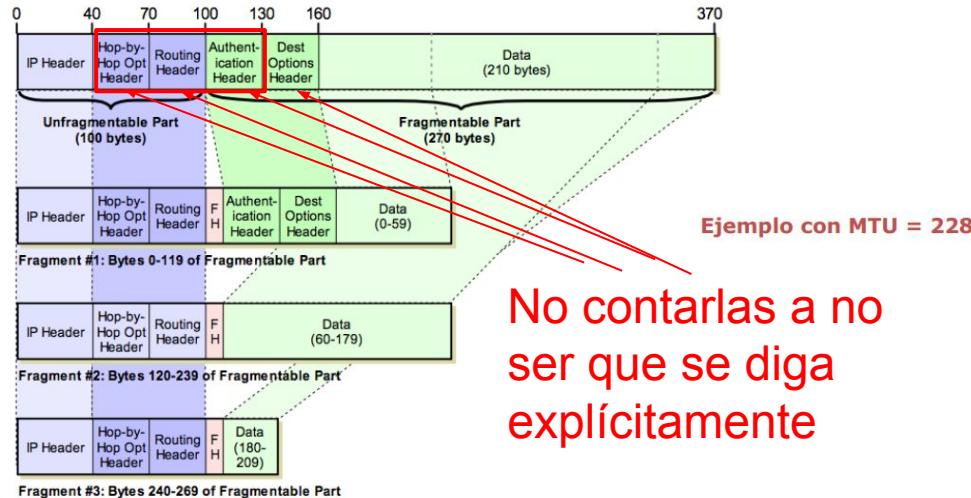
## Cabeceras de extensión [RFC 2460](#)

- Sólo se añaden las cabeceras adicionales **necesarias**
  - menor carga
- Ofrecen varios servicios y mejoras.
- Cabeceras **Hop-by-Hop, Routing** y **Destination Options [1]** analizadas salto a salto.
- Resto de cabeceras analizadas **solo por el extremo final**
- Campo 'Next Header' (NH) indica la cabecera siguiente

Siempre la primera

- **Hop-by-Hop Options (NH=0)**
  - para todos los router intermedios
- **Destination Options [1] (NH=60)**
  - para el destinatario (IP dest) y nodos que aparecen en routing header
- **Routing (NH=43)**
  - encaminamiento fuente
- **Fragment (NH=44)**
  - solo en origen
- **Authentication (NH=51)**
  - asegura la identidad del router
- **Encapsulating Security Payload (NH=50)**
  - cifrado de los datos
- **Destination Options [2] (NH=60)**
  - sólo para el destinatario final
- **Upper-layer (NH=6:TCP, 17:UDP, 58:ICMPv6)**
  - Cabeceras superiores (como IPv4 Protocol)
- **Mobility (NH=135)**
  - Opciones de movilidad

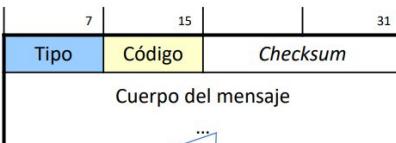
Orden recomendado



# Funciones de control

## • ICMPv6, como ICMPv4

### Formato de mensaje ICMPv6



### Mensajes INFORMATIVOS ICMPv6

Tipo	Código	Descripción
128	Echo Request (Solicitud de eco)	
129	Echo Reply (Respuesta de eco)	

Mensajes de ERROR ICMPv6		
Tipo	Código	Descripción
1	Destination Unreachable (Destino inalcanzable)	
	0	Sin ruta hacia el destino
	1	Comunicación prohibida administrativamente
	2	Sin asignar
	3	Dirección no alcanzable
2	Packet too Big (Paquete demasiado grande)	
	3	Time Exceeded (Tiempo excedido)
	0	Límite de saltos excedido
	1	Tiempo de desfragmentación excedido
	4	Parameter Problems (Problema de parámetros)
4	0	Campo erróneo de cabecera
	1	Tipo de cabecera siguiente desconocida
	2	Opción IPv6 desconocida

### Packet too big

Type = 2	Code	Checksum
MTU		
Parte del paquete que provocó el error (lo máximo posible dado el mínimo MTU de IPv6)		

### Parameter problem

Type = 4	Code	Checksum
Pointer		
Parte del paquete que provocó el error (lo máximo posible dado el mínimo MTU de IPv6)		

Pointer: offset del paquete donde se encontró el error

### Destination Unreachable

Type = 1	Code	Checksum
Unused		
Parte del paquete que provocó el error (lo máximo posible dado el mínimo MTU de IPv6)		

### Time exceeded

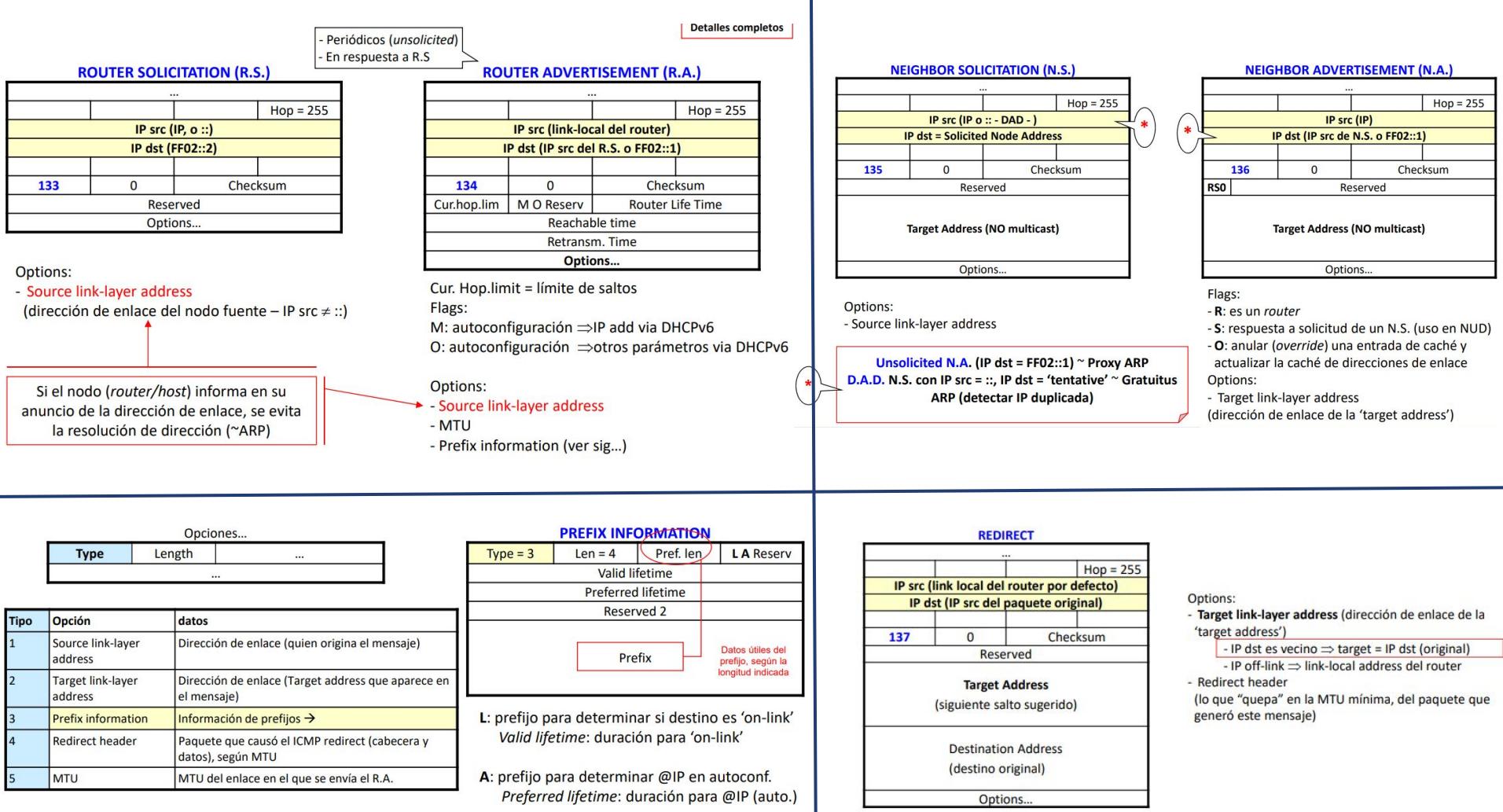
Type = 3	Code	Checksum
Unused		
Parte del paquete que provocó el error (lo máximo posible dado el mínimo MTU de IPv6)		

### Echo request (E.RQ)

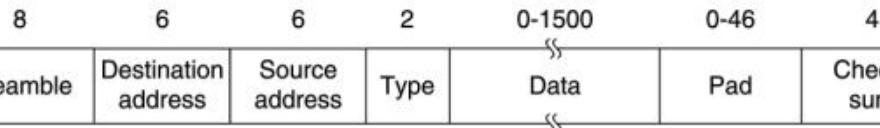
Type = 128	Code	Checksum
Identifier		Sequence number
Data...		

### Echo reply (E.RP)

Type = 129	Code	Checksum
Identifier (= E.RQ)		Sequence num. (= E.RQ)
Data...		

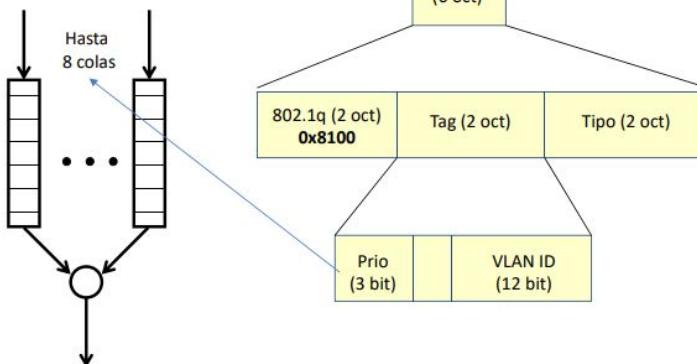


# Ethernet



Trama Ethernet con prioridad

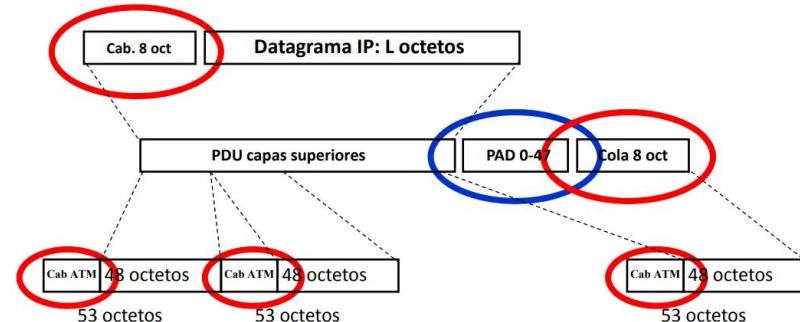
Preámbulo (8 oct)	Dirección MAC destino (6 oct)	Dirección MAC fuente (6 oct)	Tipo (2 oct)	Datos + PAD (46 a 1500 oct)	CRC (4 oct)
----------------------	----------------------------------	---------------------------------	-----------------	--------------------------------	----------------



Dentro del túnel GRE NO lleva preámbulo ni CRC

# ATM sobre ADSL

Possible encapsulado de datagramas IP sobre AAL5 (existen otras posibilidades):



$$\text{Número de celdas} = C = \lceil (L+8+8)/48 \rceil$$

Da igual cual elijas, pero le tienes que indicar si asumes o no cabecera



Se calcula el número de celdas ATM necesarias:  $C = \lceil (L+8)/48 \rceil$

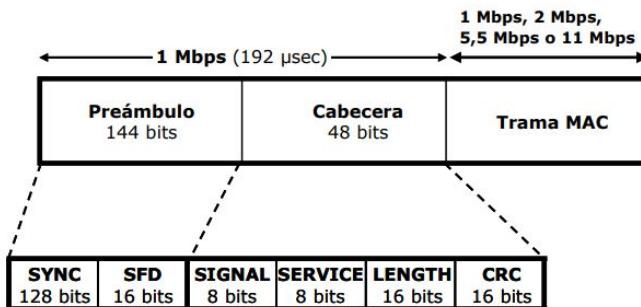
# Tecnologías de Acceso Inalámbricas. WLAN

## IEEE 802.11. NIVEL FÍSICO. Protocolos de nivel físico.

ARP, etc).

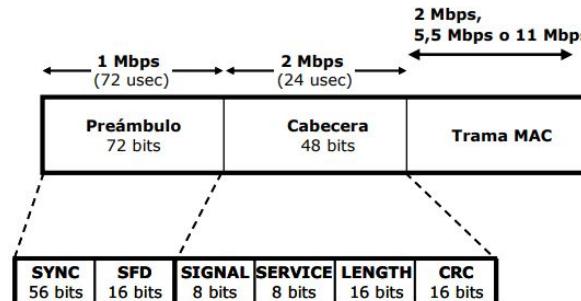
### IEEE 802.11b

Trama PLCP: PPDU (PLCP Protocol Data Unit)

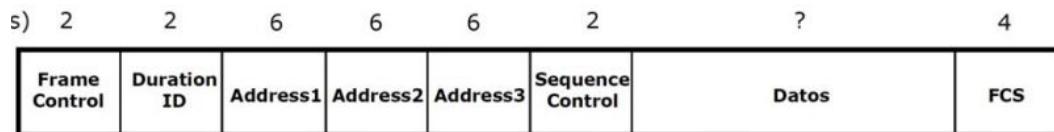


Formato largo (obligatorio)

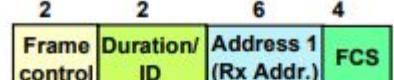
DATOS (28B + datos)



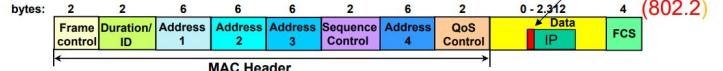
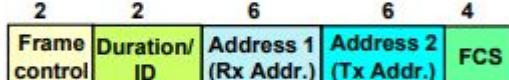
Formato corto (opcional)



ACK = CTS (14B)



RTS (14B)



general  
( no para lo que  
usamos nosotros )

**QoS SOLO si se  
especifica  
EXPLÍCITAMENTE  
(suma 2B a las de datos)**

**ASUMIR ESTE FORMATO  
SIEMPRE**  
(a no ser que digan  
explicitamente de usar otro)

## Fragmentación

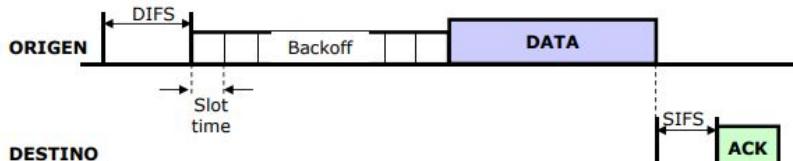
- En el nivel MAC de 802.11 se prevé la posibilidad de que el transmisor fragmente una trama para enviarla en trozos más pequeños.
- Si el emisor ve que las tramas no están llegando bien, puede decidir fragmentar las tramas grandes para que tengan más probabilidad de llegar al receptor.
- La fragmentación permite enviar datos en canales más hostiles, aun a costa de aumentar la sobrecarga (se reduce la eficiencia).
- Por cada fragmento se devuelve un ACK, por lo que en caso necesario es retransmitido por separado.

Intervalo = [a,b]

$$\text{Backoff medio} = \text{Tslot} \times \frac{(a + b)}{2}$$

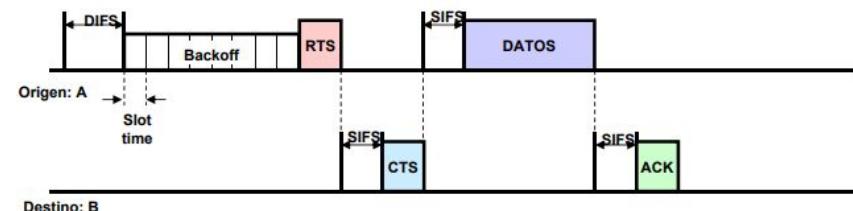


## SIN envío de RTS/CTS



$$T_{TOTAL} = DIFS + \overline{\text{Backoff}} + T_{\text{preámbulo+H\_PLCP}}^{\text{DATA}} + T_{\text{MAC}}^{\text{DATA}} + SIFS + T_{\text{preámbulo+H\_PLCP}}^{\text{ACK}} + T_{\text{MAC}}^{\text{ACK}}$$

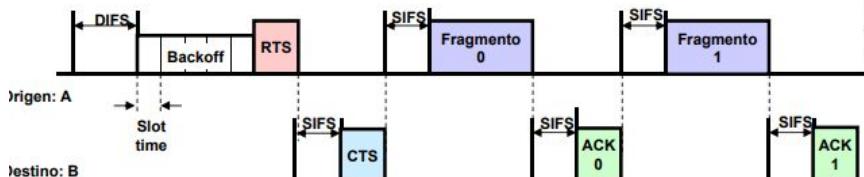
## CON envío de RTS/CTS



Hay que sumar a los tiempos anteriores  $T_{RTS} + SIFS + T_{CTS} + SIFS$

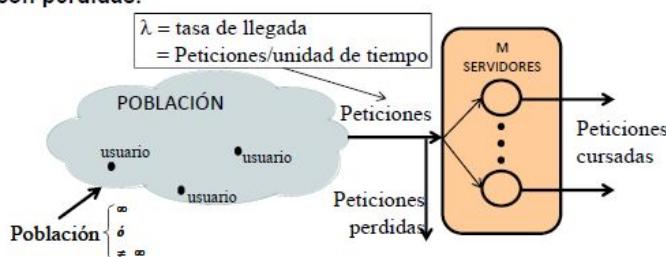
- $aSlotTime = 20 \mu s$
- $SIFS = 10 \mu s$
- $DIFS = SIFS + 2 \times aSlotTime = 50 \mu s$
- $\text{Backoff} = \text{Aleatorio entre } 0 \text{ y } \text{CW} (\text{CW}_{\min} = 31 \text{ y } \text{CW}_{\max} = 1023 \text{ aSlotTime})$ 
  - Cogemos el valor mínimo  $\text{CW}_{\min} = 31 \Rightarrow \text{Backoff}_{\text{medio}} = 15,5 \times 20 \mu s = 310 \mu s$

## Con FRAGMENTACIÓN (cambia el envío de DATA)



## Dimensionado de sistemas de voz

### 1.- Sistemas con pérdidas.



El parámetro de calidad es la probabilidad de pérdidas

Modelo Erlang-B

Prob\_Pérdidas=B=ERLANG-B (A,N)

A=Tráfico Ofrecido- Número conexiones por unidad de tiempo \*  
\* Duración de la conexión =  $\lambda \cdot \mu$   
N=número de servidores (circuitos)

TABULADAS  $\rightarrow B = E_{1,N}(A) = \frac{A^N / N!}{\sum_{k=0}^{N-1} A^k / k!}$

seño y Administración de Redes  
sos Prácticos. Análisis de infraestructuras para TelP y datos.

en Erlangs (s/s, h/h...)

N servidores que transmiten

erlang-B(A,N)	1	2
0,05	047619	001189
0,10	090909	004525

prob. de pérdida (de 0 a 1)

### 2.- Sistemas con demora.

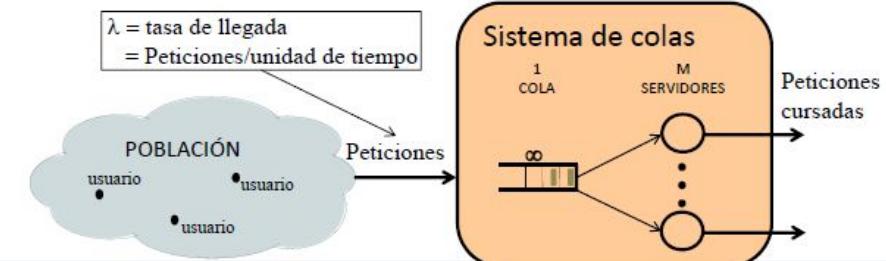
Capacidad del buffer ilimitada. El sistema no tiene pérdidas.  
El parámetro de calidad es la probabilidad de demora.

El modelo a aplicar es Erlang-C  $\rightarrow$  Prob\_Demora=P\_D=ERLANG-C (A,N)

$$P_D = E_{2,N}(A) = \frac{A^N}{N!} \frac{N}{N-A} P(0)$$

$$\bar{T} = \frac{h}{N-A} \quad P(T_D \geq t) = e^{-\frac{t}{\bar{T}}} \quad \bar{x} = \frac{A}{N-A} + N$$

$$\bar{T} = \frac{h}{N-A} P_D \quad P(T_D \geq t) = e^{-\frac{t}{\bar{T}} P_D} \quad \bar{x} = \frac{A}{N-A} P_D + A$$



N →

erlang-C(A,N)	1	2
0,1	100000	004762
0,2	200000	018182

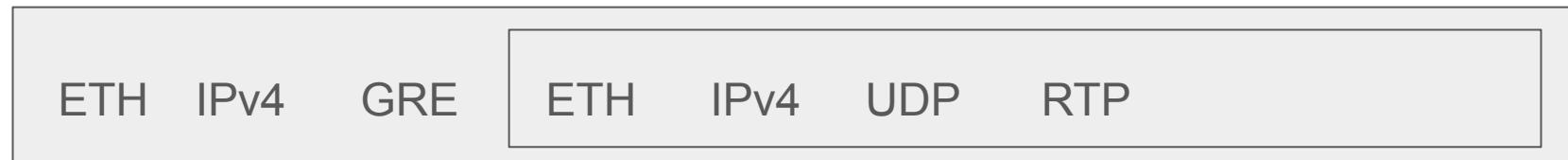
A ↓

004762 → 0.004762  
018182 → 0.018182

prob. de demora (de 0 a 1)

(ya tiene en cuenta  
preámbulo y crc)

## Ethernet sobre GRE



(sin preámbulo  
ni CRC)

## IPv6 sobre IPv4



MTU si no se dice nada



	Tamaño mínimo de trama	MTU habitual
Ethernet	46	1500
WiFi	0	2312