Redes de sensores inalámbricos (RSI)

IEEE802.15.4 & 6lowpan (capa de adaptación a IPv6)

Leonardo Steinfeld

Inst. de Ingeniería Eléctrica, Fac. de Ingeniería
Universidad de la República (Uruguay)

NEON
Network of Competence on Internet of Things

Co-funded by the Erasmus+ Programme of the European Union

FACULTAD DE INGENIERÍA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Objetivos

- Describir características particulares de las tramas IEEE 802.15.4.
- Comprender las limitaciones de IEEE 802.15.4 para contener paquetes IPv6.
- Describir los conceptos aplicados por 6LoWPAN para fragmentación de paquetes y compresión de encabezados.

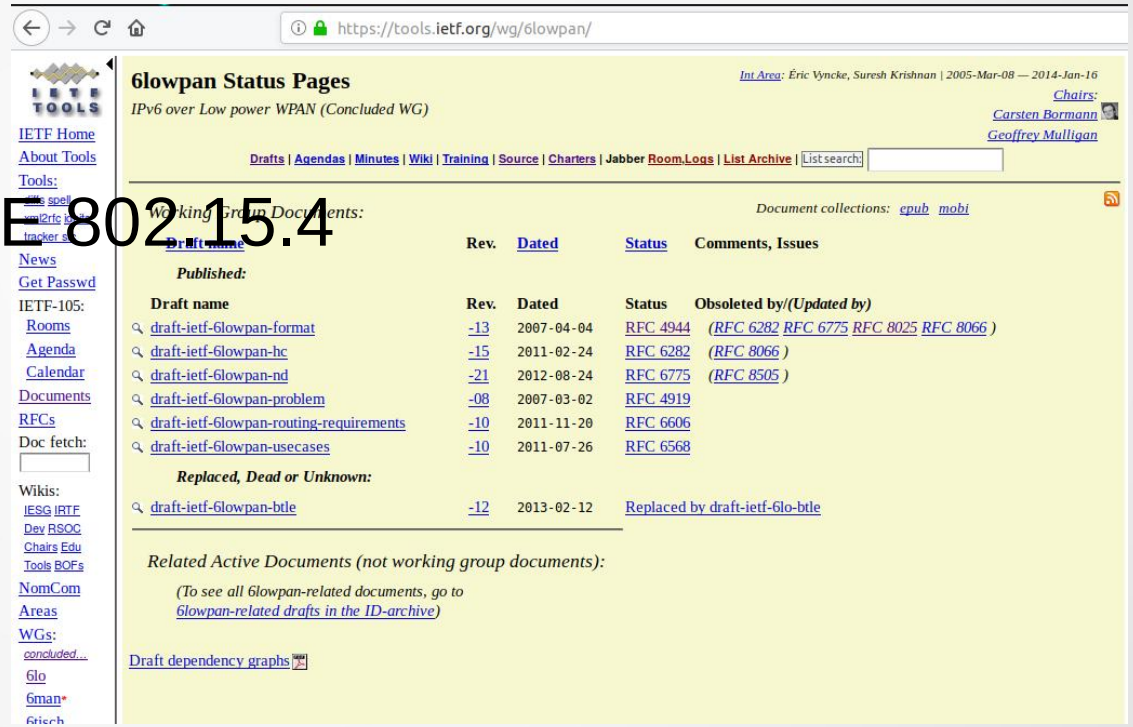
Agenda

- IEEE 802.15.4 PHY/MAC
  - funciones y tramas (*data units*)
  - tipos y direccionamiento
- 6LoWPAN
  - motivación & desafíos
  - funciones

IPv6 over Low power WPAN

- **6lowpan**: IETF Working Group (finalizado)

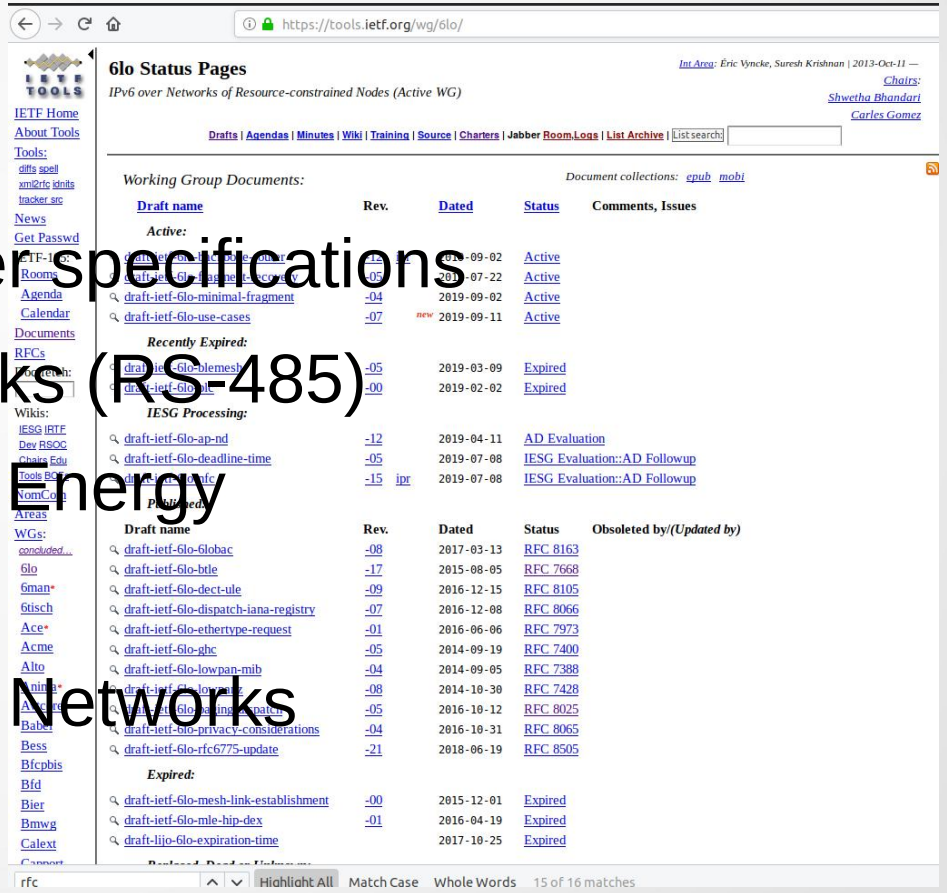
- IPv6 Packets over IEEE 802.15.4
  - RFC 4944: bases
  - RFC 6282: NHC
  - RFC 6775: ND



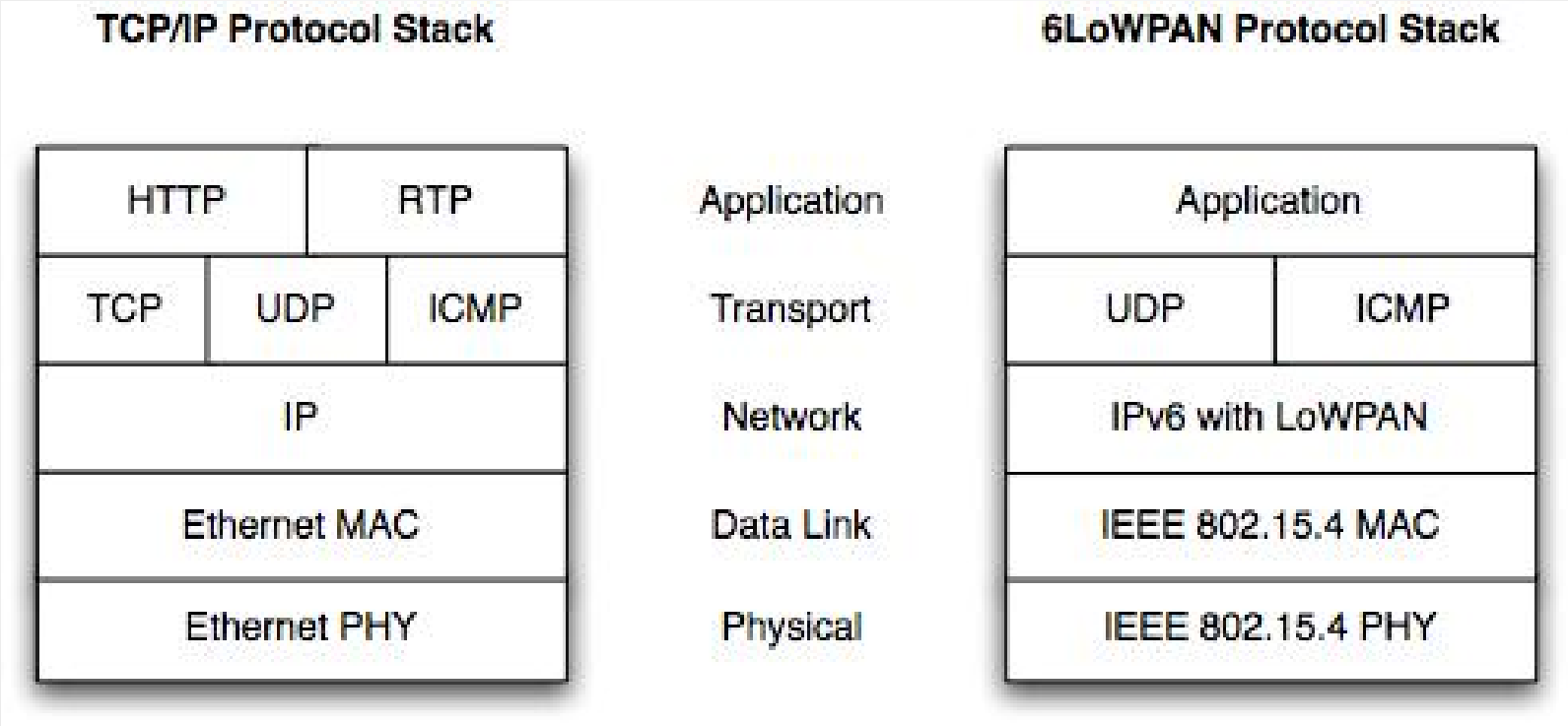
IPv6 over Netw. of Resource-constrained Nodes

- **6lo**: IETF Working Group (activo)
  - generaliza **6lowpan**

- IPv6-over-foo adaptation layer specifications
  - RFC 8163: MS/TP Networks (RS-485)
  - RFC 7668: Bluetooth Low Energy
  - RFC 8105: DECT – ULE
  - RFC 7428: ITU-T G.9959 Networks

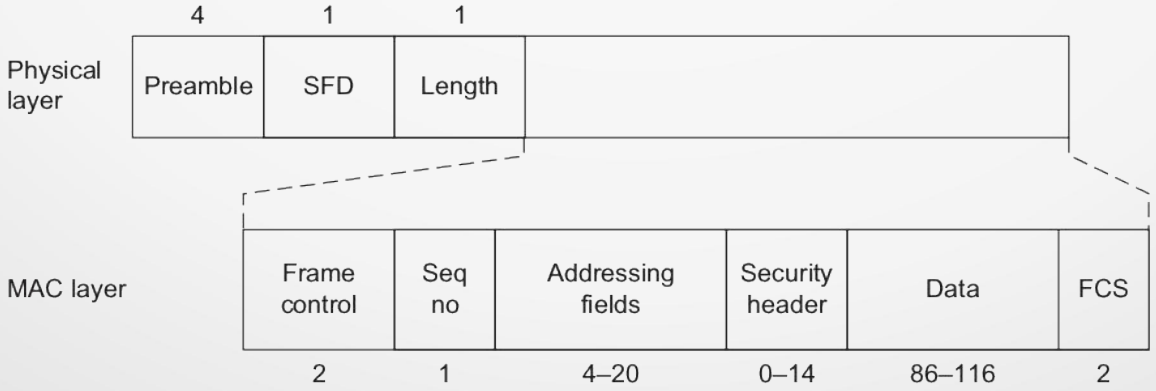


6lowpan: capa de adaptación



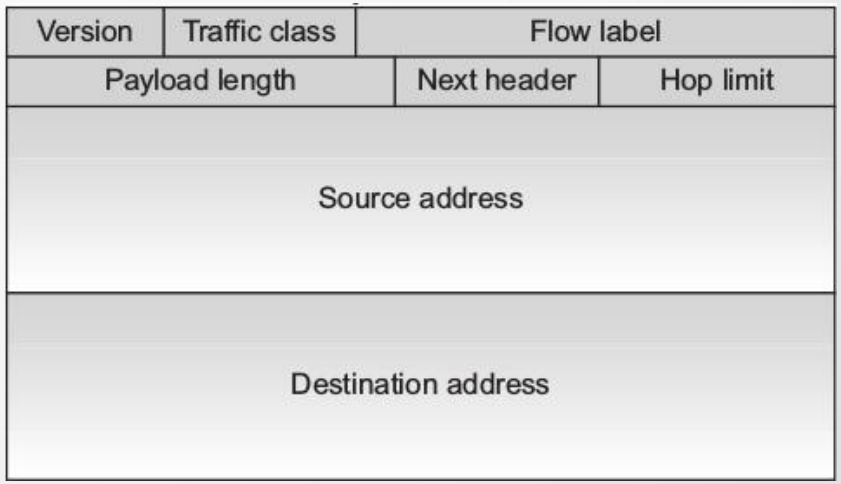
IEEE 802.15.4: algunas características

- Frame pequeño: 127 bytes
  - PER razonablemente bajos para BER no despreciables
- Direcciones
  - 16-bit short / IEEE 64-bit extended MAC.
- Low data rates
  - De 20 kbps (868 MHz) a 250 kbps (2.45 GHz).



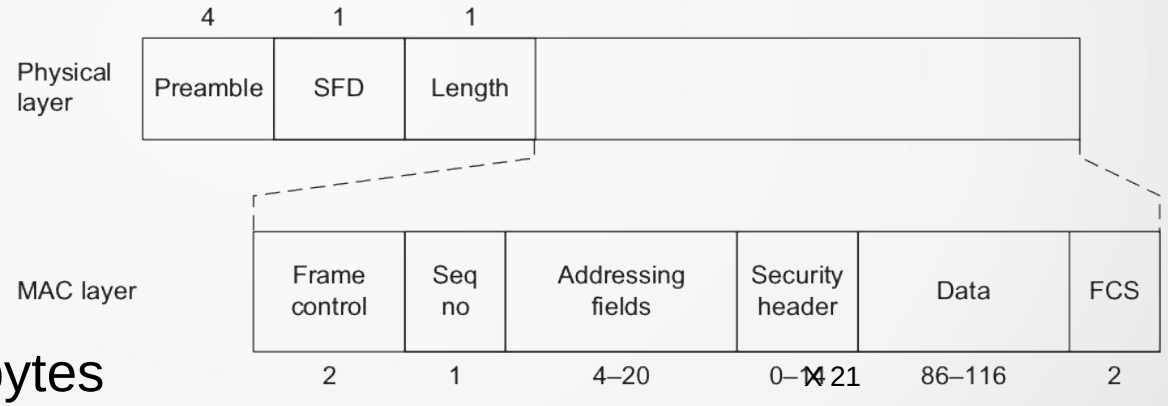
IPv6: algunas características

- Paquete relativamente grande: 1280 bytes.
- Direcciones: 128 bits.
- Stateless address autoconfiguration (SAA)
  - Simplifica configuración y gestión
- IPv6 incluye multicast como parte integral de su arquitectura,
  - Neighbor Discovery (ND) usa link-local multicast para
    - address resolution
    - duplicate address detection
    - router discovery.



IPv6 over LoWPAN: desafíos

- Paquetes IPv6 en LoWPANs.
  - Bajo throughput
  - Buffers limitados
  - **Frames ~10 veces menores que MTU mínimo requerido por IPv6**
- Necesidad
  - Fragmentación
  - Compresión
- Ejemplo:
  - Payload efectivo 81 bytes
  - IPv6 header: 40 bytes
  - UDP/TCP header: 8 / 20 bytes,
  - Queda: ....



127−(2+1+20+2) = 102 bytes
AES-CCM-128: 21 bytes extra

6LoWPAN: capa de adaptación

- RFC 4944 (September 2007):
  - Transmission of IPv6 Packets over IEEE 802.15.4 Networks
    - técnicas de compresión de encabezados
- RFC 6282 (September 2011):
  - Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks
    - mejoras a RFC 4944
- RFC 6775 (November 2012):
  - Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)

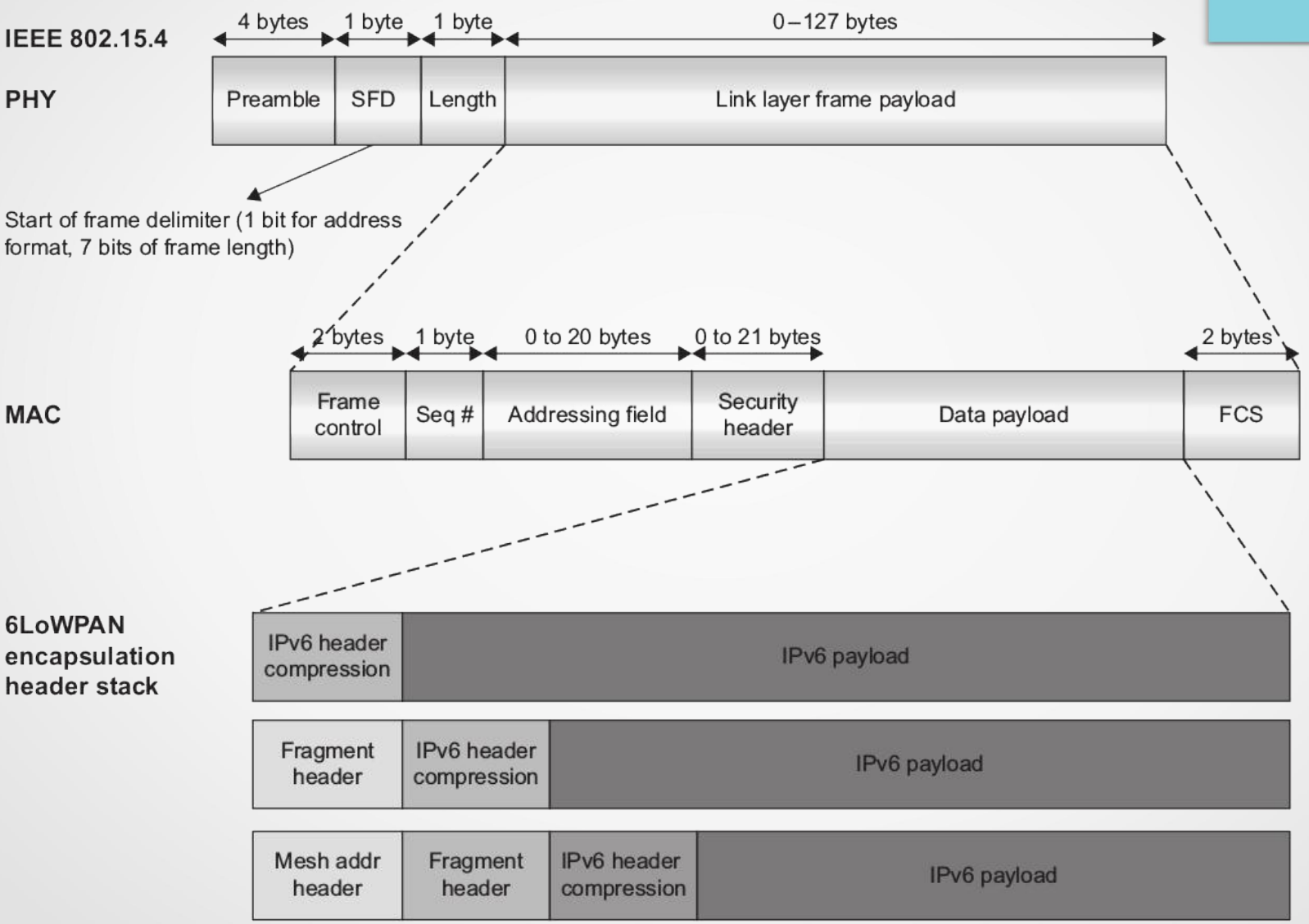
6LoWPAN: funciones

- Provee tres servicios:
  - Fragmentado y reensamblado de paquetes
  - Compresión de encabezados
  - Enrutamiento en capa 2 “mesh-under”

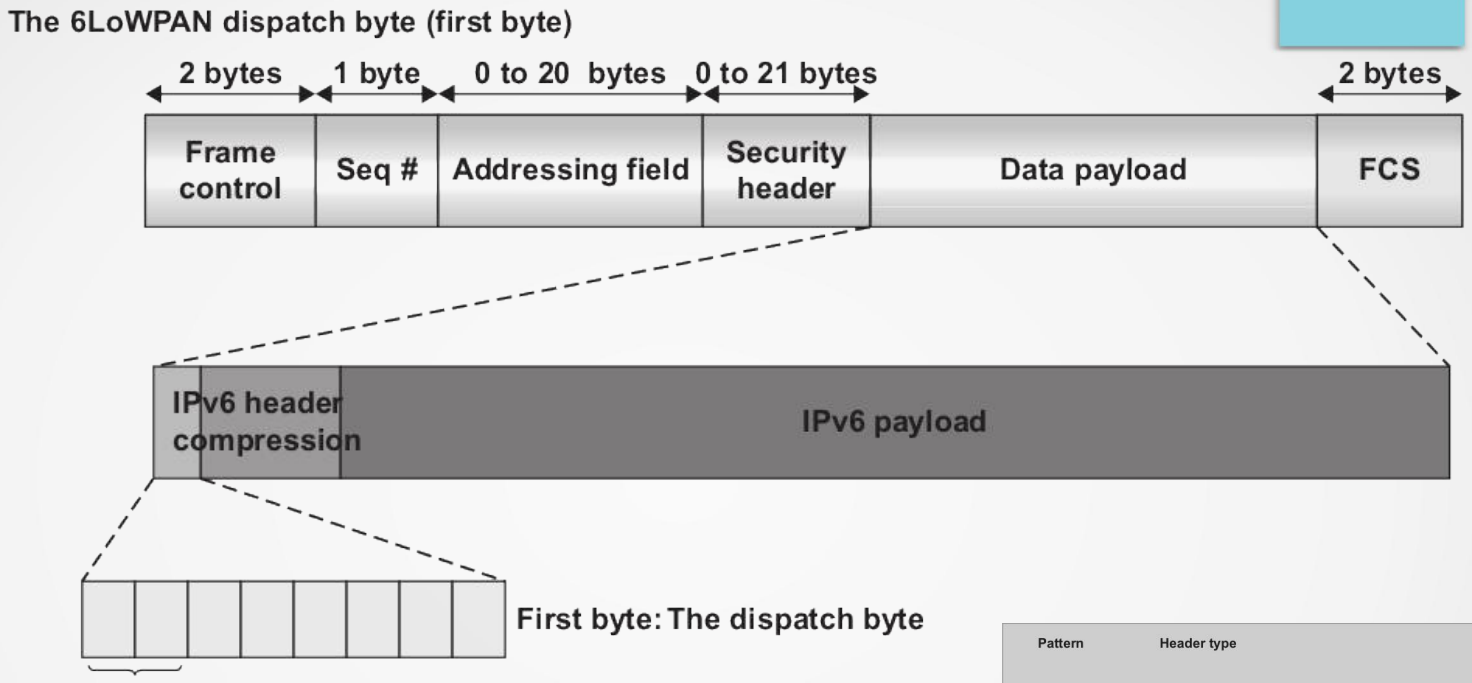
6loWPAN: pila de encabezados

- IEEE 802.15.4 encapsula paquete IPv6
  - “encapsulation header stack” antes de cada paquete IPv6
  - cada encabezado se agrega si se necesita
  - dispach byte (primer byte): identifica el *next header* de la pila.
- Tres encabezados:
  - mesh addressing header
  - fragment header
  - IPv6 header compression header
- Nota: deben aparecer en ese orden si están presentes

6loWPAN: encapsulado de IPv6



6loWPAN: dispach byte (header)



00	**Not a 6LoWPAN frame**
01	IPv6 addressing header
10	Mesh header
11	Fragmentation header (6 lower bits are 100xxx)

Pattern	Header type
00 xxxxxx	NALP - not a LoWPAN frame
01 000001	IPv6 - uncompressed IPv6 addresses
01 000010	LOWPAN_HC1-LOWPAN_HC1 compressed IPv6
01 000011	reserved - reserved for future use
...	reserved - reserved for future use
01 001111	reserved - reserved for future use
01 010000	LOWPAN_BCO - LOWPAN_BCO broadcast
01 010001	reserved - reserved for future use
...	reserved - reserved for future use
01 111110	reserved - reserved for future use
01 111111	ESC - additional dispatch byte follows
10 xxxxxx	MESH - Mesh header
11 000xxx	FRAG1 - fragmentation header (first)
11 001000	reserved - reserved for future use
...	reserved - reserved for future use
11 011111	reserved - reserved for future use
11 100xxx	FRAGN - fragmentation header (subsequent)
11 101000	reserved - reserved for future use
...	reserved - reserved for future use
11 111111	reserved - reserved for future use

6loWPAN: dispach byte (detalle)

00	**Not a 6LoWPAN frame**
01	IPv6 addressing header
10	Mesh header
11	Fragmentation header (6 lower bits are 100xxx)

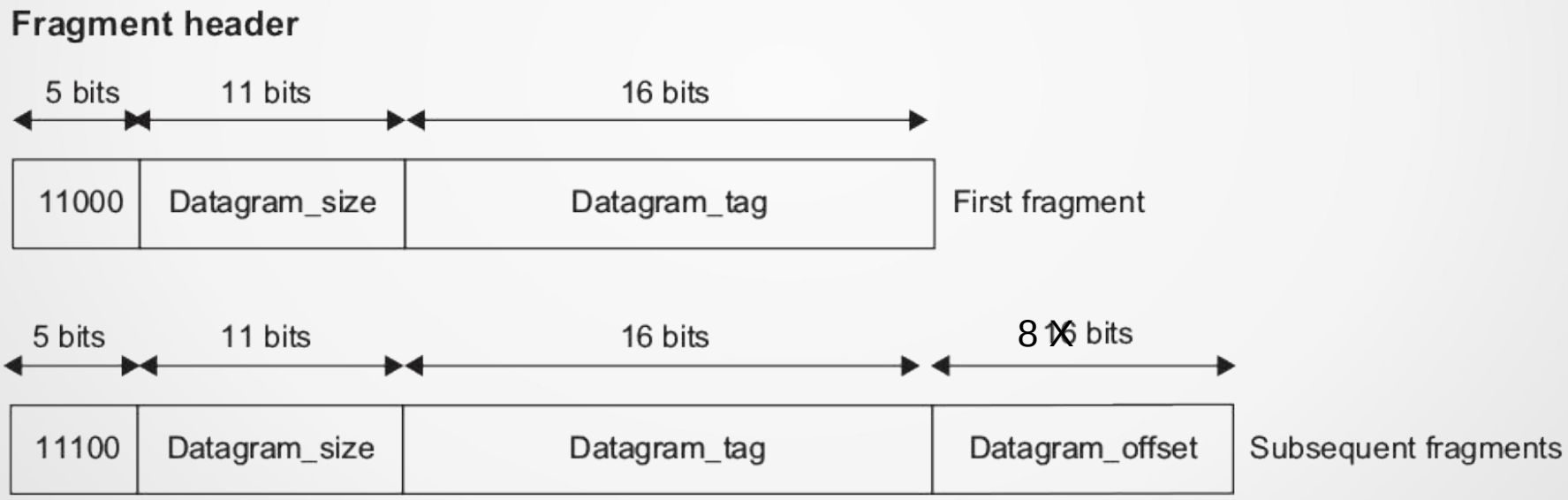
Pattern	Header type
00 xxxxxx	NALP - not a LoWPAN frame
01 000001	IPv6 - uncompressed IPv6 addresses
01 000010	LOWPAN_HC1-LOWPAN_HC1 compressed IPv6
01 000011	reserved - reserved for future use
...	reserved - reserved for future use
01 001111	reserved - reserved for future use
01 010000	LOWPAN_BCO - LOWPAN_BCO broadcast
01 010001	reserved - reserved for future use
...	reserved - reserved for future use
01 111110	reserved - reserved for future use
01 111111	ESC - additional dispatch byte follows
10 xxxxxx	MESH - Mesh header
11 000xxx	FRAG1 - fragmentation header (first)
11 001000	reserved - reserved for future use
...	reserved - reserved for future use
11 011111	reserved - reserved for future use
11 100xxx	FRAGN - fragmentation header (subsequent)
11 101000	reserved - reserved for future use
...	reserved - reserved for future use
11 111111	reserved - reserved for future use

Mesh addressing header (dispach: 10)

- Estrategia mesh-under “routing”
  - nodos rutean en capa 2 (link layer) usando direcciones de capa 2
  - solo FFD (full function devices, según IEEE 802.15.4), reduced function devices (RFDs) mandan a FFD

Fragmentation header (dispach 11)

- FRAG1: primero 11 000 XXX
- FRAGN: siguientes 11 100 XXX



Fragmentation

- FRAG1
  - datagram_size (11 bits)
    - Suficiente para 1280 bytes?
  - datagram_tag (16 bits)
    - identificador único, igual en todos los fragmentos
    - se recomienda incrementar con cada nuevo *frame* fragmentado
- FRAGN
  - datagram_offset (8 bit)
  - indica offset (en unidades de 8 bytes)
- RFC4944 especifica usar un timer de 60 s para recibir todos los fragmentos

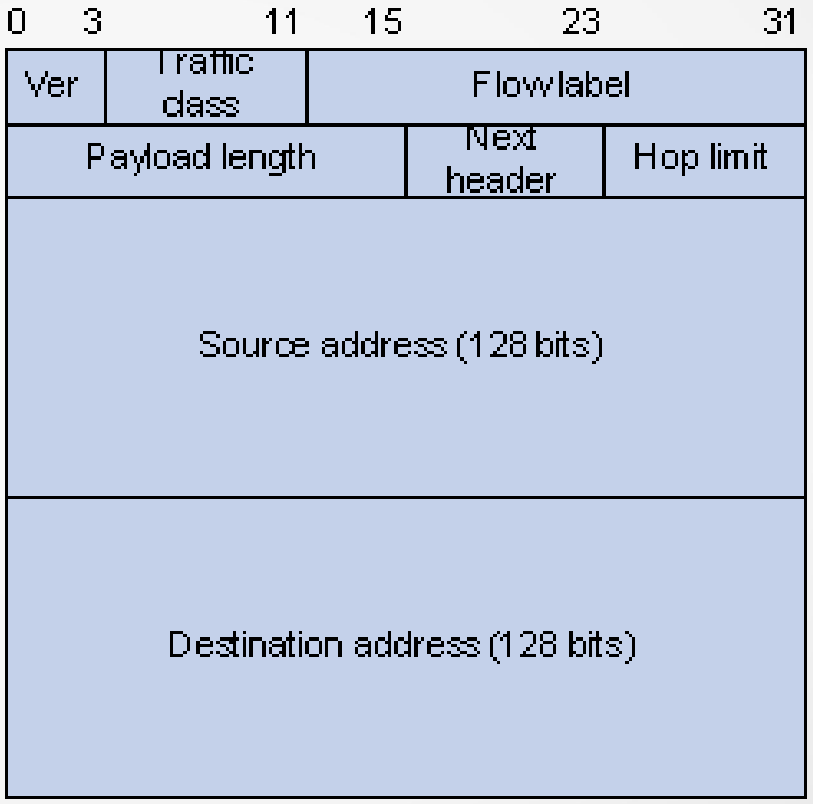
6LoWPAN Header Compression

- RFC 4944:
  - LOWPAN_HC1 / LOWPAN_HC2
- RFC 6282:
  - LOWPAN_IPHC (IPHC)
  - LOWPAN_NHC (NHC).
- Nota: IPHC seguramente sea la técnica usada, HC1 y HC2 serán declaradas obsoletas (deprecated).

HC1 Compression Technique

- ¿Qué podemos hacer?

  Ideas....



Basic IPv6 header

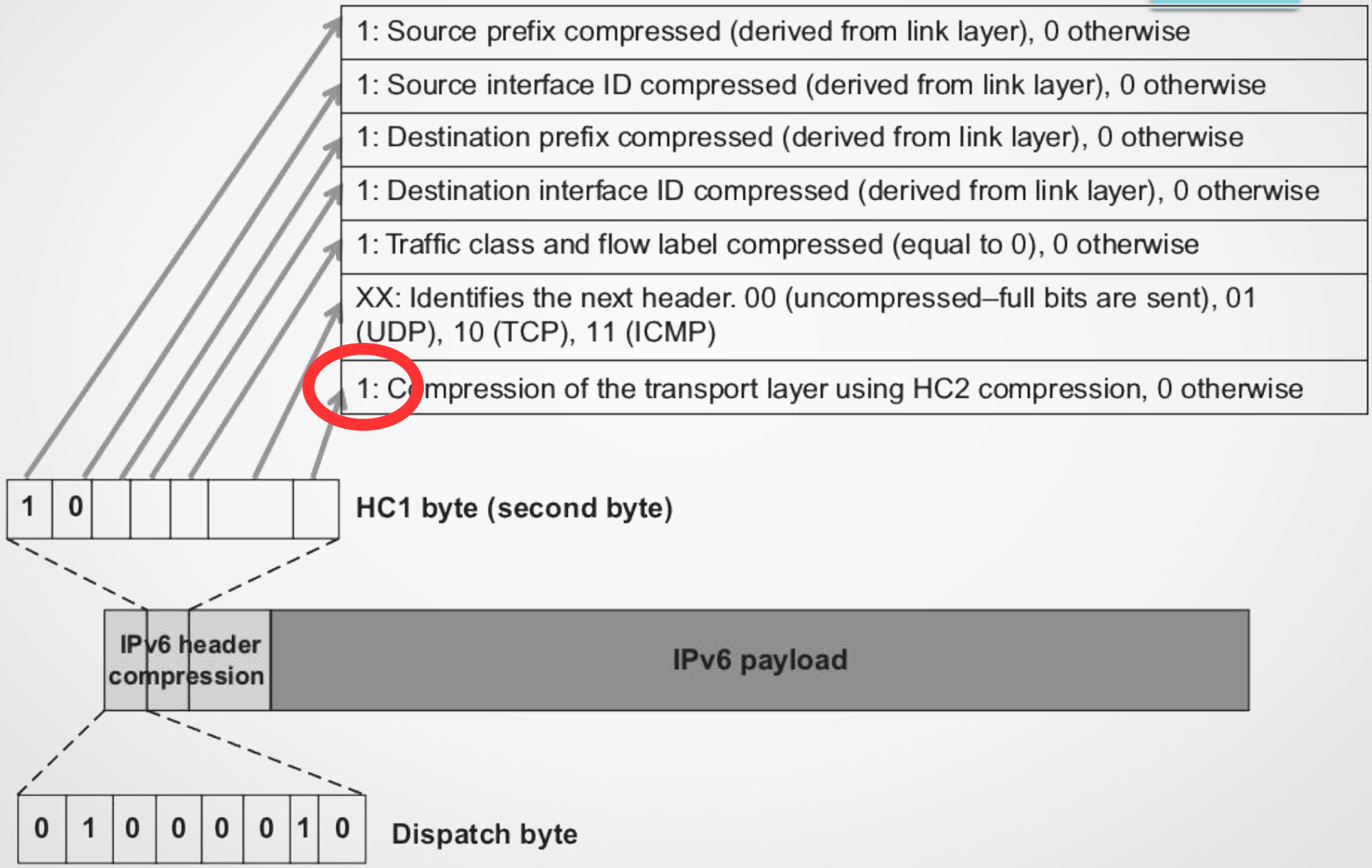
HC1 Compression Technique

- Se basa en:
  - IP version es siempre 6.
  - Packet length puede inferirse del *frame length* (IEEE 802.15.4 frame).
  - Traffic Class y Flow label comúnmente vale 0
  - Next header es: UDP, TCP, or ICMP.
  - IPv6 interface ID (últimos 64 bits de la dirección IPv6) puede inferirse de la dirección link layer MAC (si se usó para asignarla)

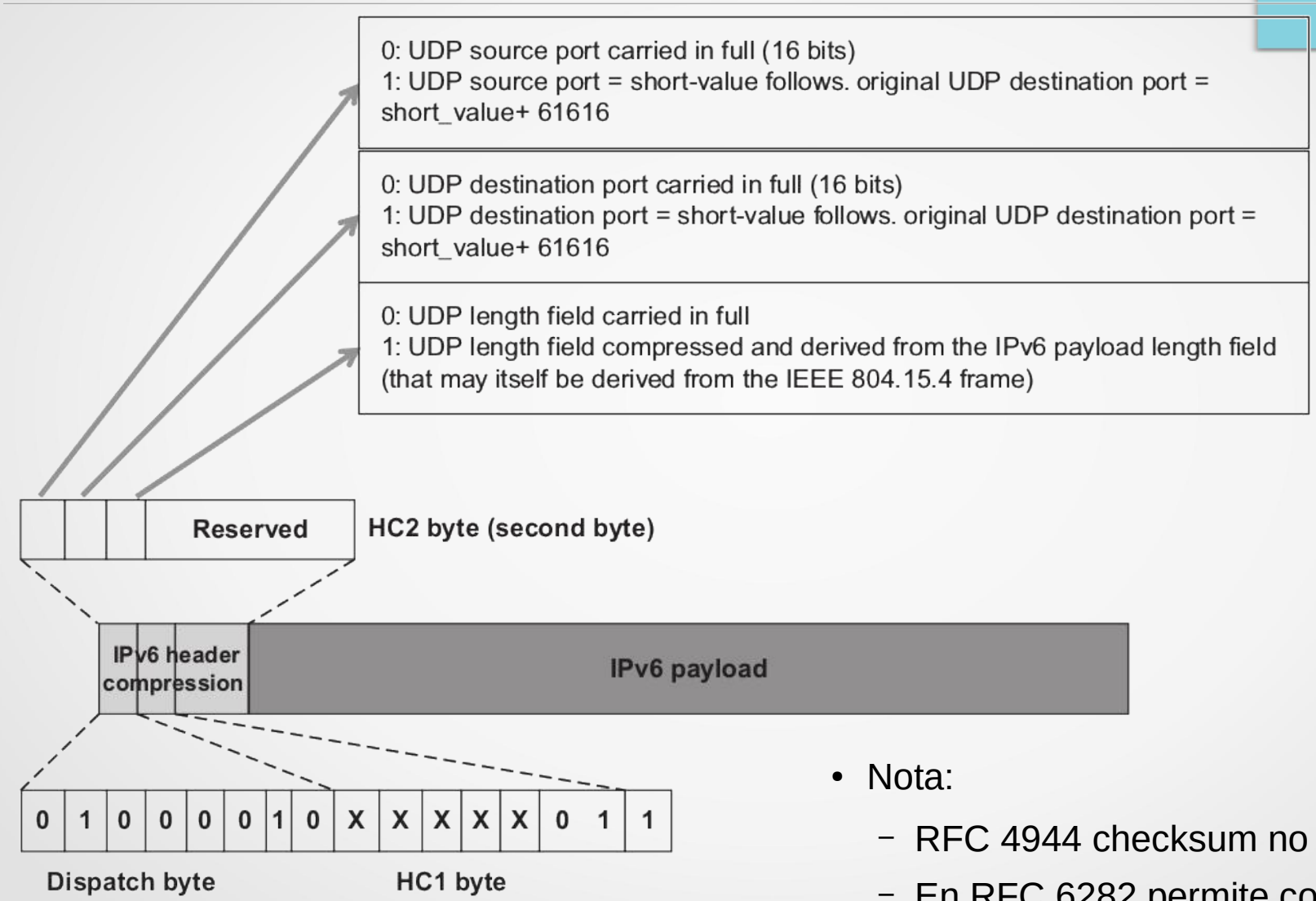
HC1 Compression Technique (dispach: 01)

- IPv6 header: 40 bytes => 3 bytes (mejor caso)
  - dispatch byte (equal to 01000010)
  - HC1 byte
  - Hop limit field (1 byte)
- Si existen campos sin comprimir, van en el sig. orden:
  - source address prefix (64 bits) y/o interface ID (64 bits),
  - destination address prefix (64 bits) y/o interface ID (64 bits)
  - TC (8 bits), flow label (20 bits)
  - next header (8 bits)

HC1 byte

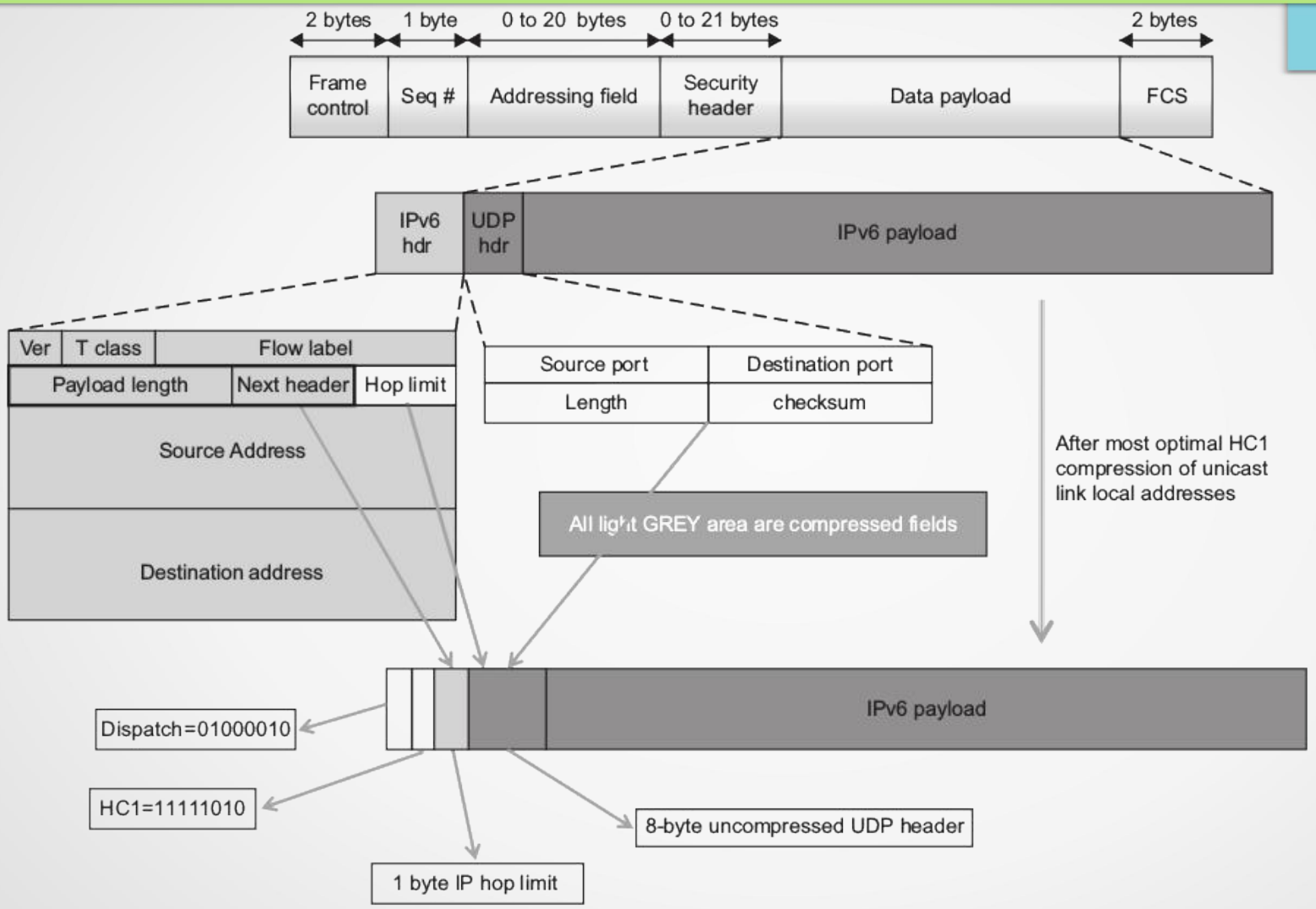

1: Source prefix compressed (derived from link layer), 0 otherwise
1: Source interface ID compressed (derived from link layer), 0 otherwise
1: Destination prefix compressed (derived from link layer), 0 otherwise
1: Destination interface ID compressed (derived from link layer), 0 otherwise
1: Traffic class and flow label compressed (equal to 0), 0 otherwise
XX: Identifies the next header. 00 (uncompressed–full bits are sent), 01 (UDP), 10 (TCP), 11 (ICMP)
1: Compression of the transport layer using HC2 compression, 0 otherwise
1 0
HC1 byte (second byte)
IPv6 header compression
IPv6 payload
0 1 0 0 0 0 1 0
Dispatch byte

HC2 byte: HC UDP



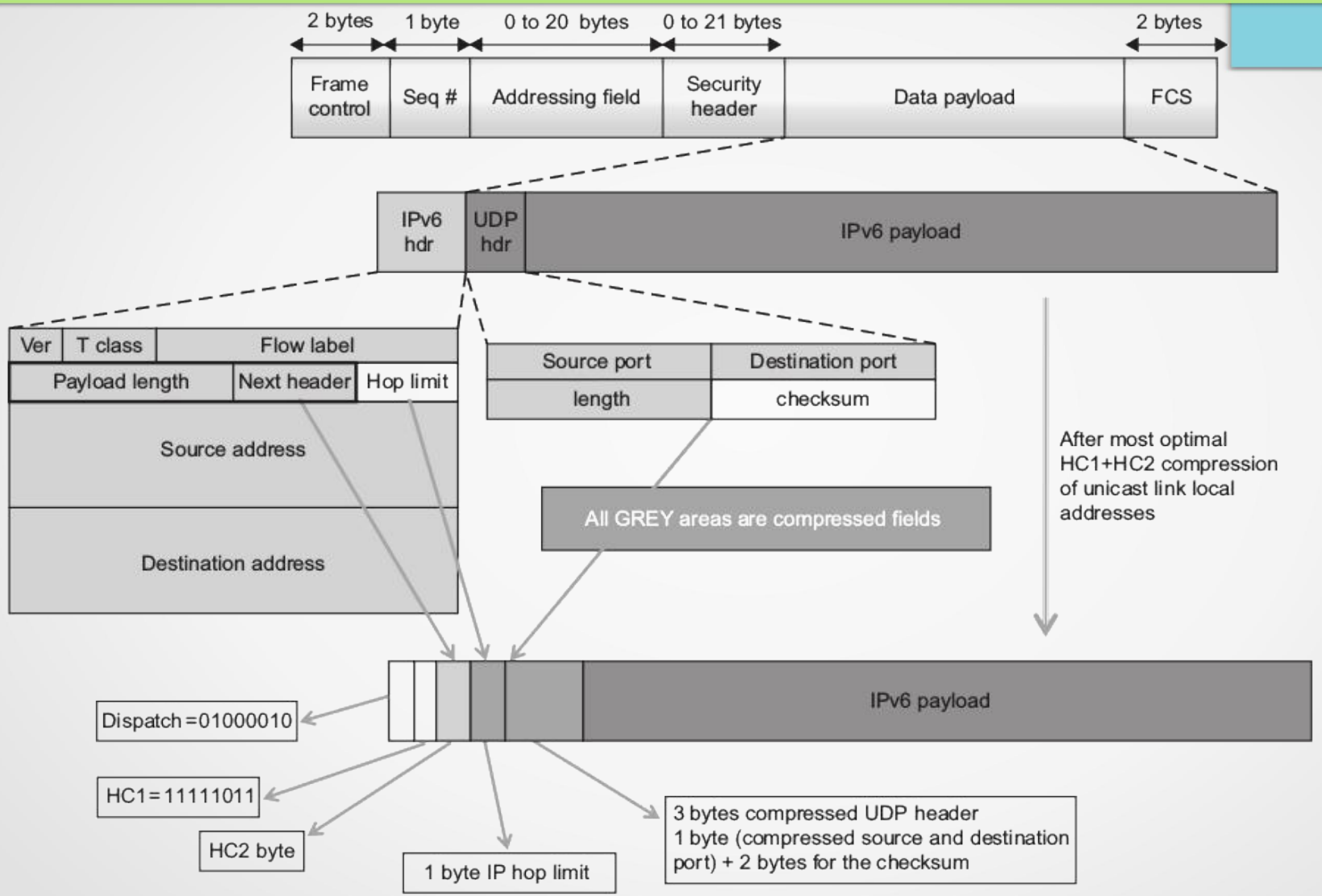
- Nota:
  - RFC 4944 checksum no comprimido
  - En RFC 6282 permite compresión.

Resultados I



HC1 link-local IPv6 addresses sin compresión UDP header

Resultados II

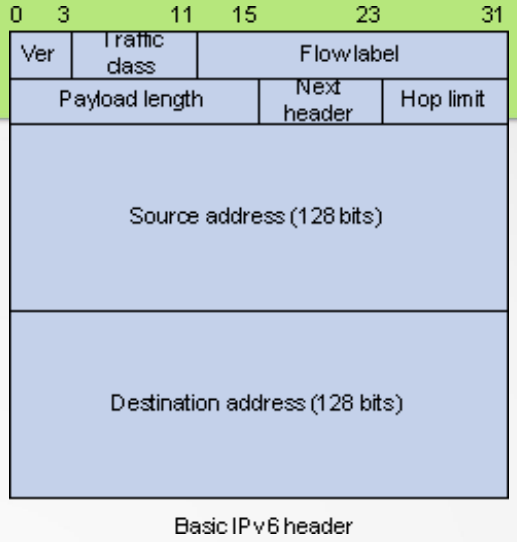


IPv6 y UDP headers usando HC1 y HC2 con direcciones link-local IPv6

Evaluación

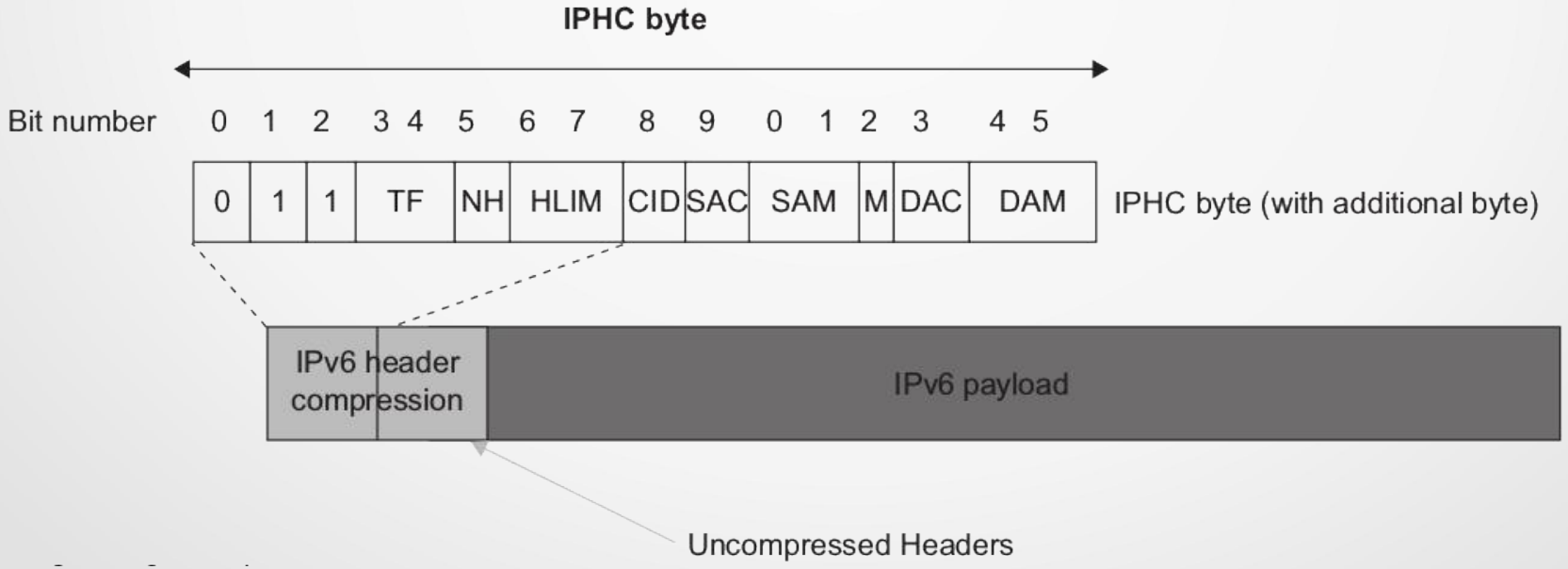
- RFC 4944
  - Define esquema general de encapsulado
  - HC1 / HC2
    - bastante eficiente para direcciones unicast link-local (usadas por protocolos ND, DHCP, y otros)
    - efecto limitado en direcciones globales y multicast
    - Estrategia: todo o nada.
- RFC 6282: nuevas técnicas de compresión
  - LOWPAN_IPHC (IPHC)
  - LOWPAN_NHC (NHC)

IPHC (dispach 011)

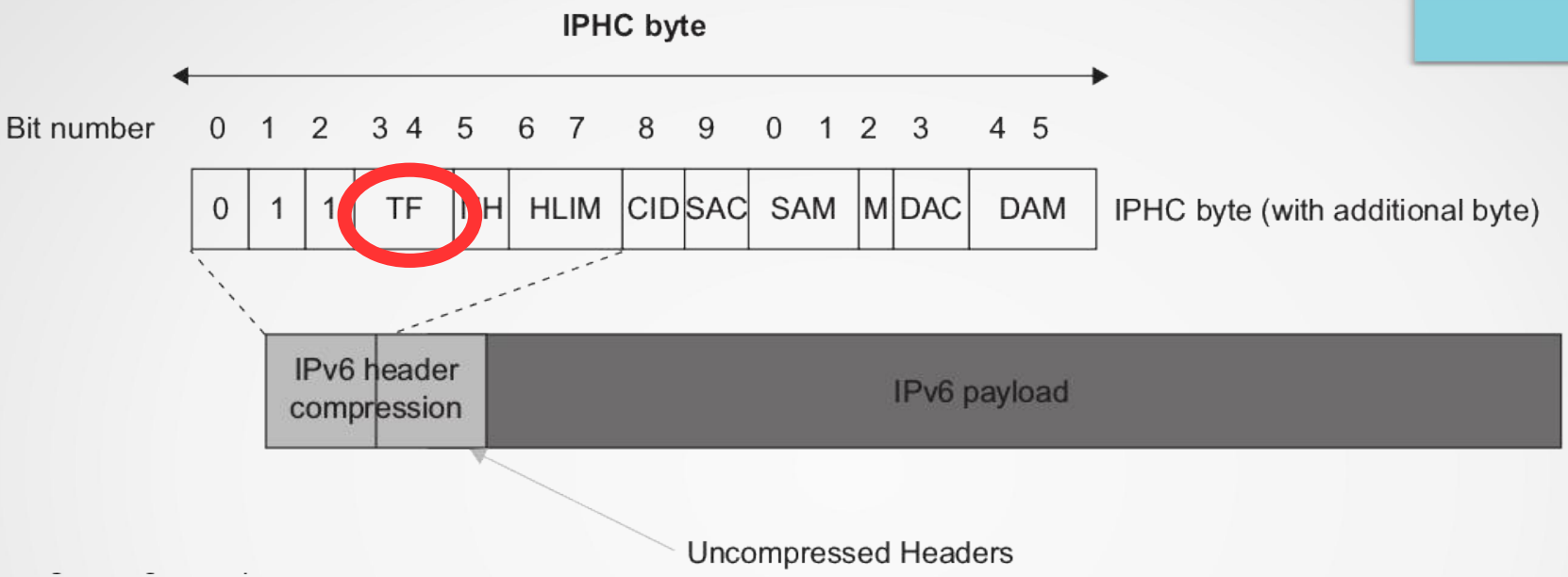


Basic IPv6 header

- IPHC: 13 bits
  - 5 bits del dispatch byte
  - 1 byte adicional opcional
- Campos de encabezado sin comprimir en orden

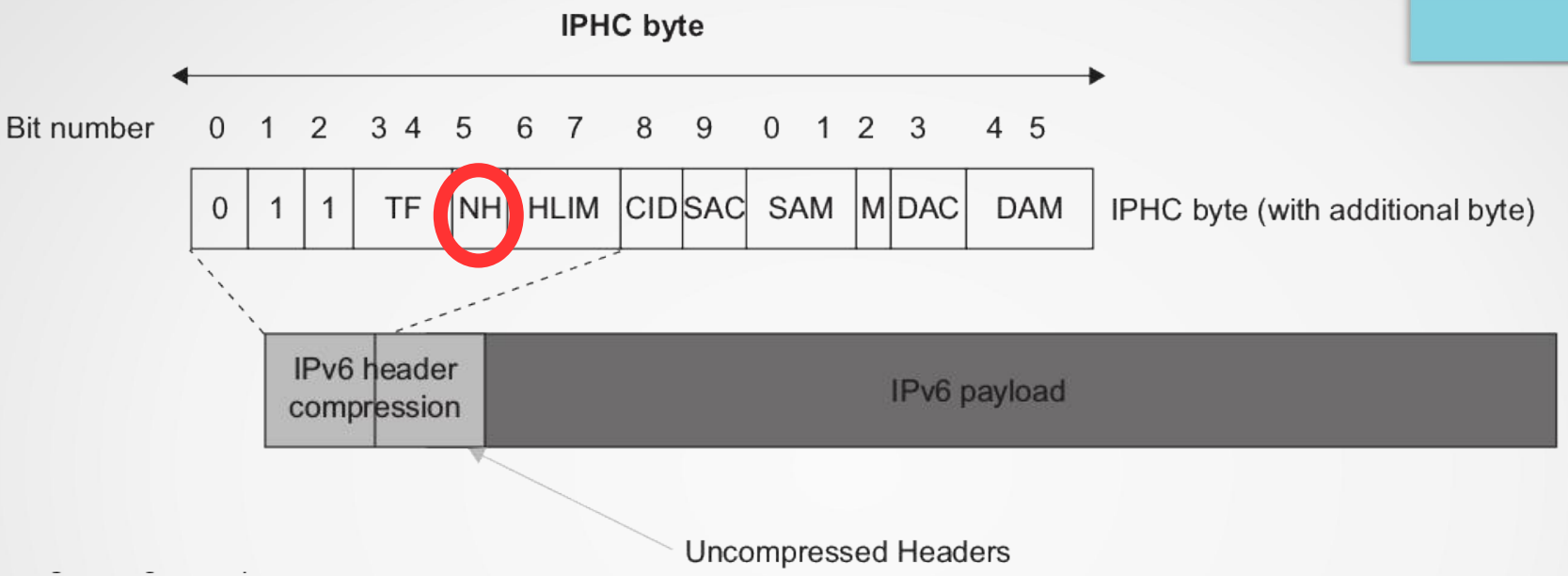


IPHC



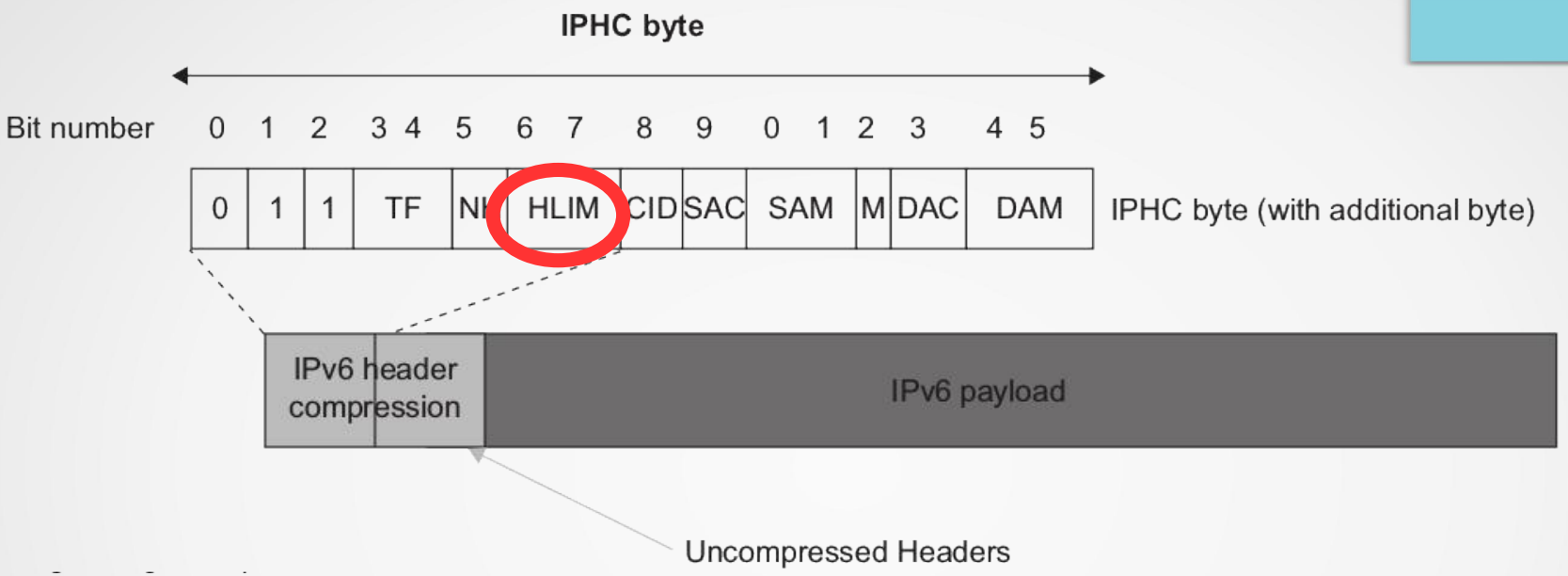
- TF: Traffic class (8 bits), Flow label (20 bits)
  - 00: in-line (Nota: + 4 bits para alineado)
  - 01: TC comprimido a 2 bits (ECN), y flow label sin comprimir
  - 10: TC in-line, y flow label field comprimido
  - 11: TC y flow label comprimidos.

IPHC



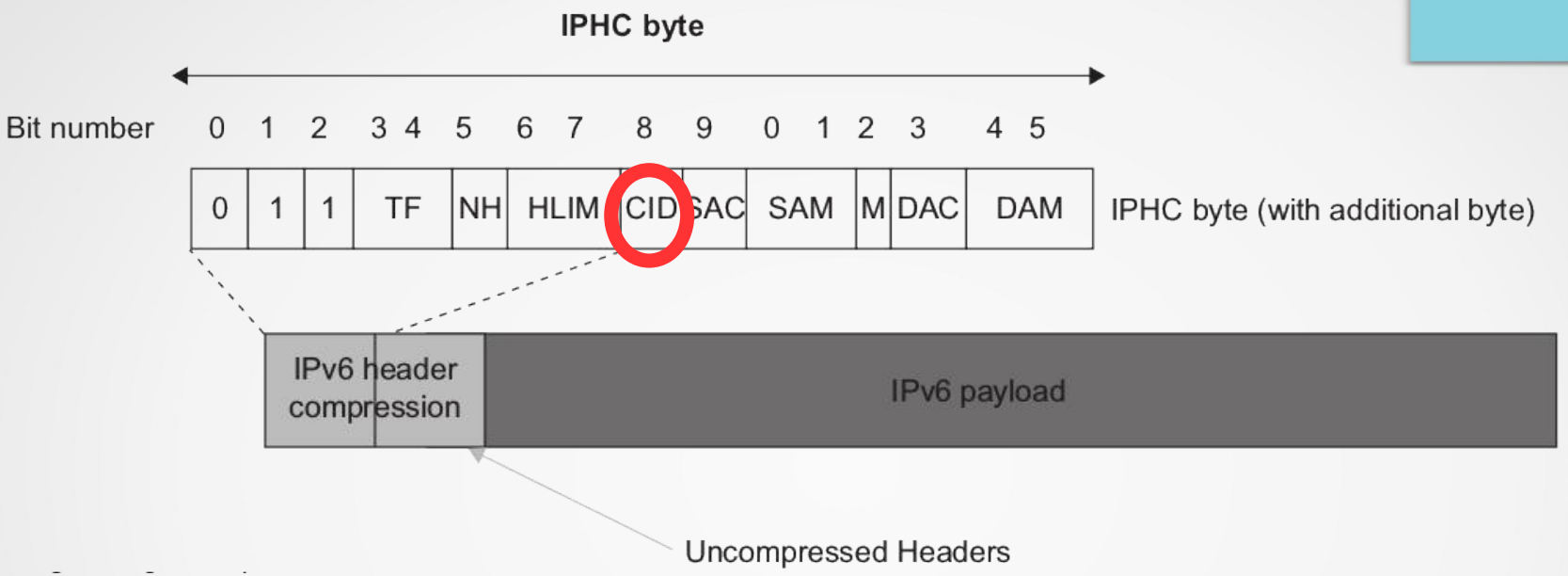
- NH (next header)
  - 0: in-line.
  - 1: next header codificado usando NHC (next header coding)

IPHC



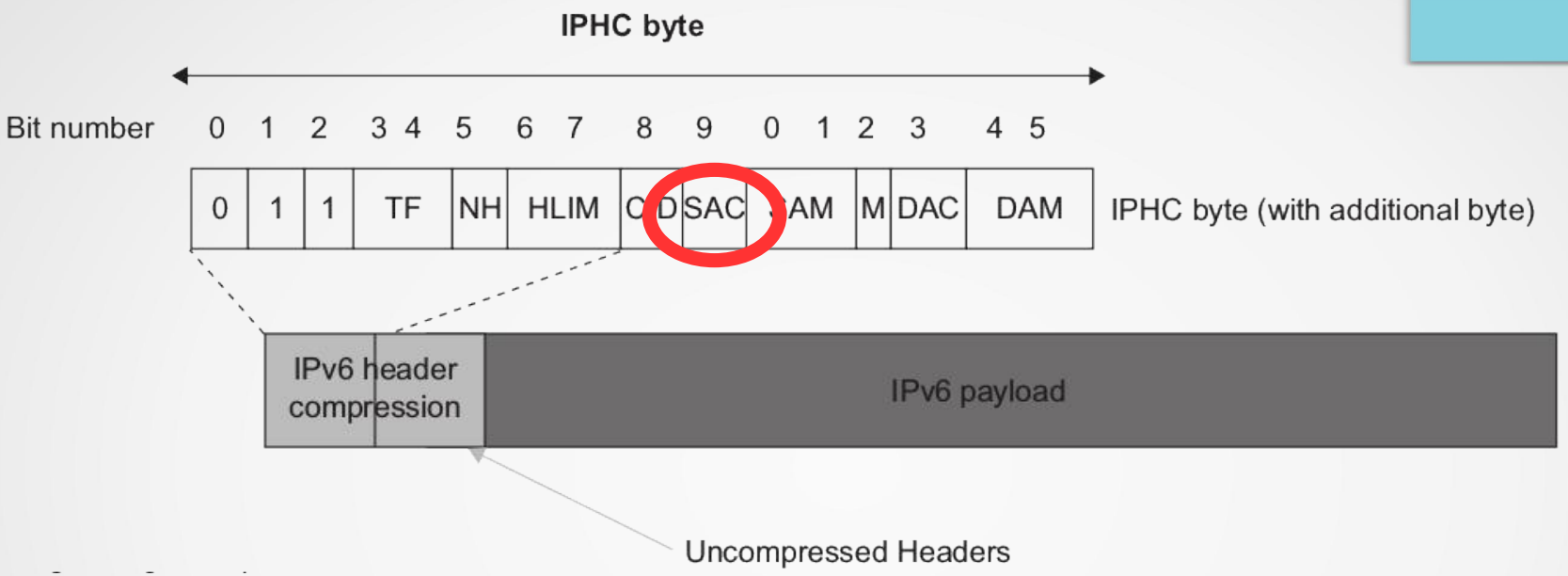
- HLIM (hop limit)
  - 00: in-line.
  - 01: hop limit = 1
  - 10: hop limit = 64
  - 11: hop limit = 255
- Nota: HC1 no comprimía.

IPHC



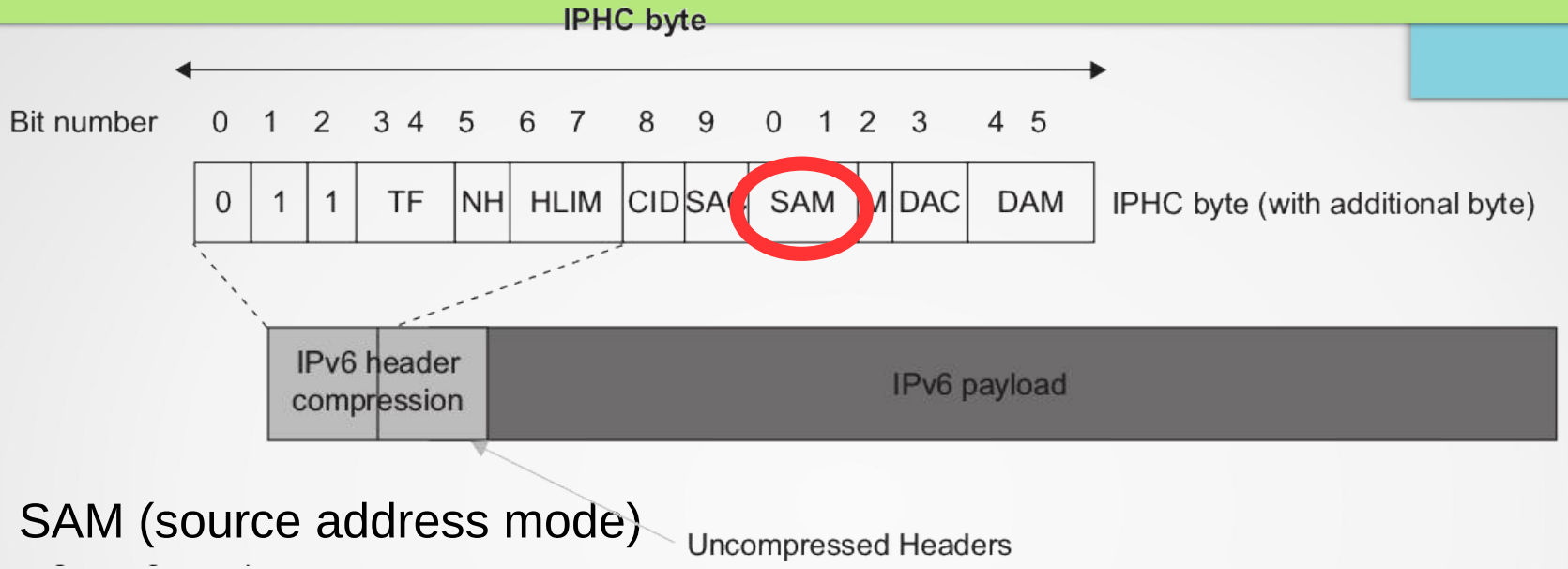
- CID (context identifier extension)
  - 0: no se usa info de contexto adicional
  - 1: se agrega 1 byte de CID luego de DAM.

IPHC



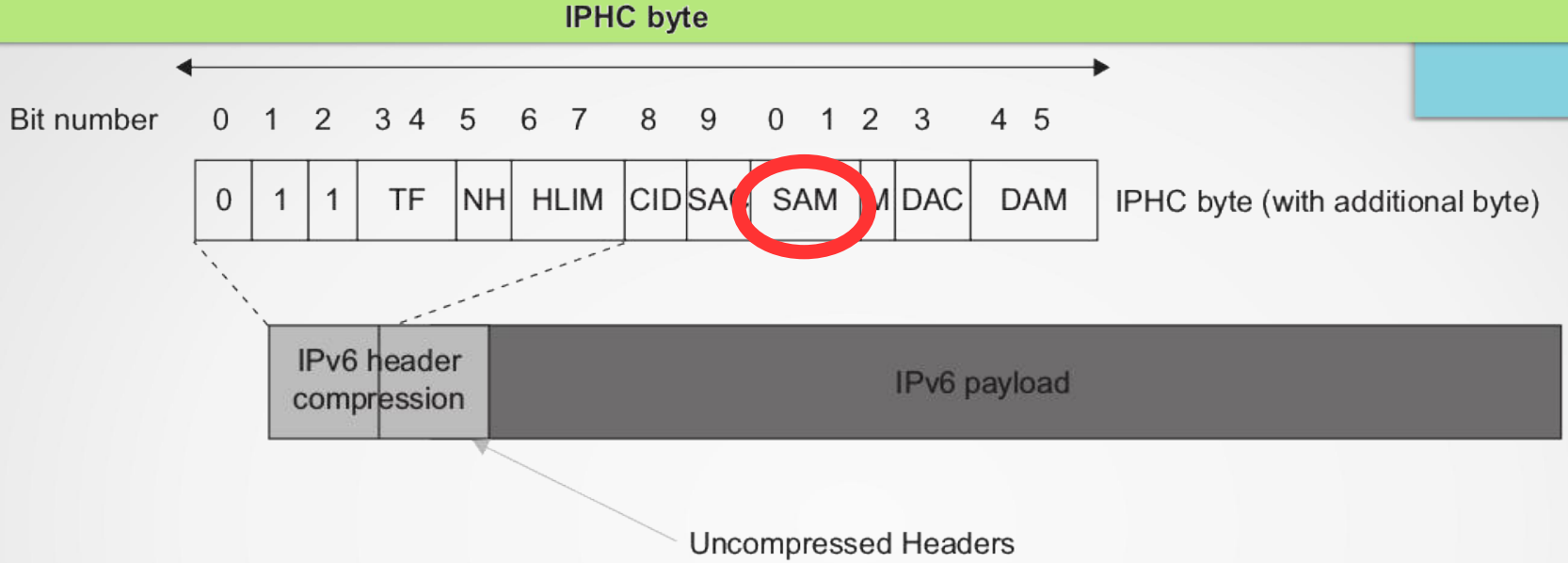
- SAC (source address compression)
  - 0: stateless.
  - 1: stateful basada en contexto.

IPHC



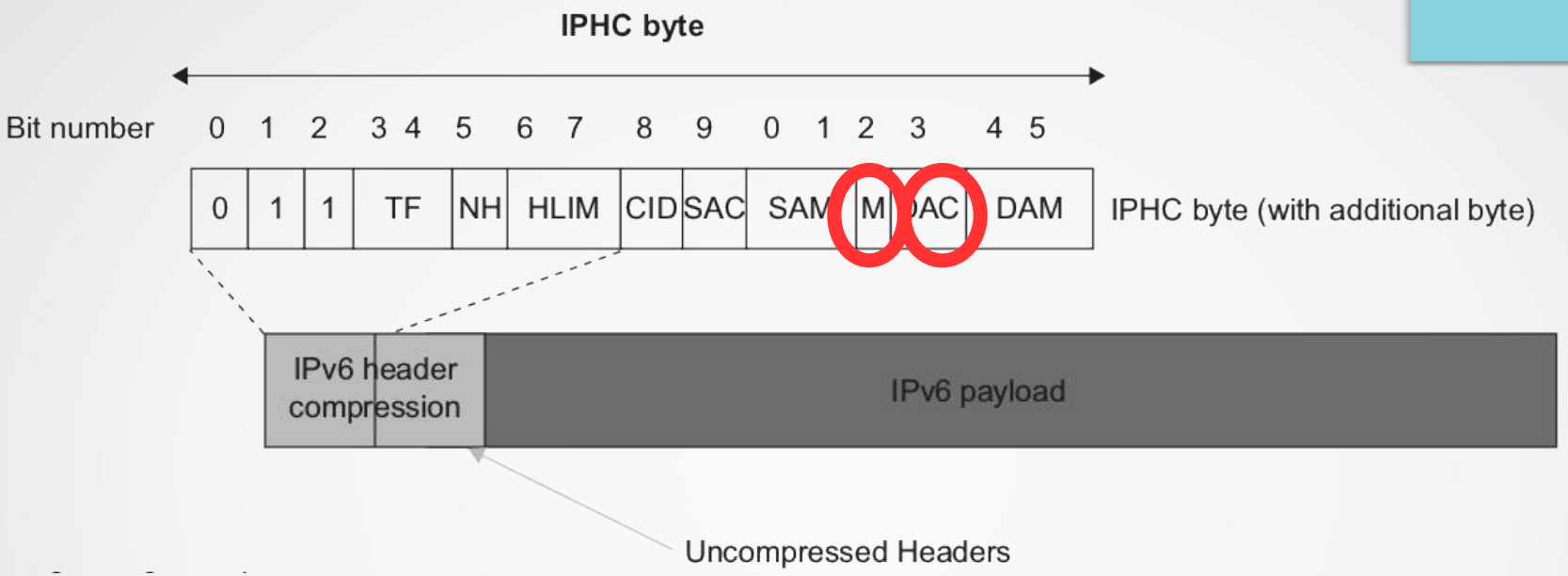
- SAM (source address mode)
- If SAC = 0 (stateless)
  - 00: 128-bit, in-line.
  - 01: 64 bits, primeros 64 bits link-local prefix rellenado con ceros, 64 bits restantes in-line
  - 10: 16 bits, primeros 64 bits link-local prefix rellenado con ceros, 64 bits restantes son 0000:00ff:fe00:XXXX, donde XXXX son los 16 bits in-line.
  - 11: 0 bits, primeros 64 bits link-local prefix rellenado con ceros, 64 bits restantes inferidos de IEEE 802.15.4 frame.

IPHC



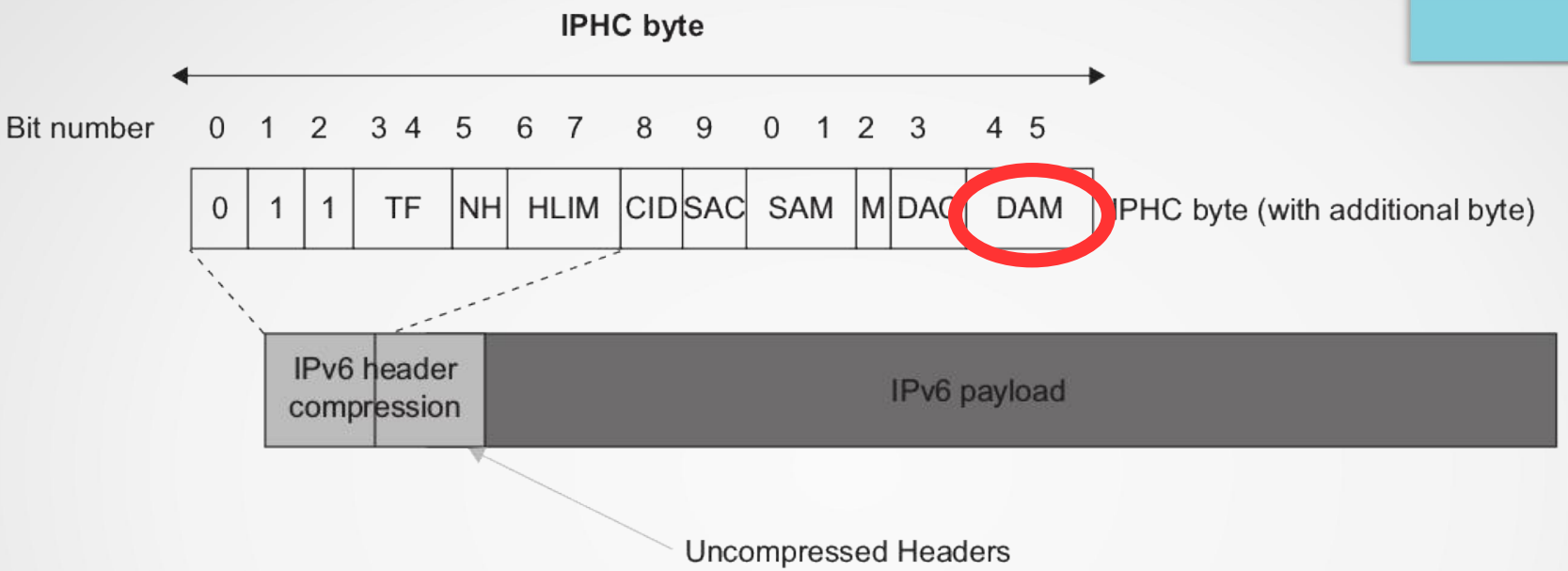
- SAM (source address mode)
- If SAC = 1 (stateful basada en contexto)
  - 01: 64 bits. 64-bit prefix inferida de la info de contexto, 64 bits restantes in-line.
  - 10: 16-bits. 64-bit prefix inferida de info de contexto, 16 bits restantes in-line.
  - 11: 0 bits. Dirección inferida de contexto y posiblemente de link layer.

IPHC



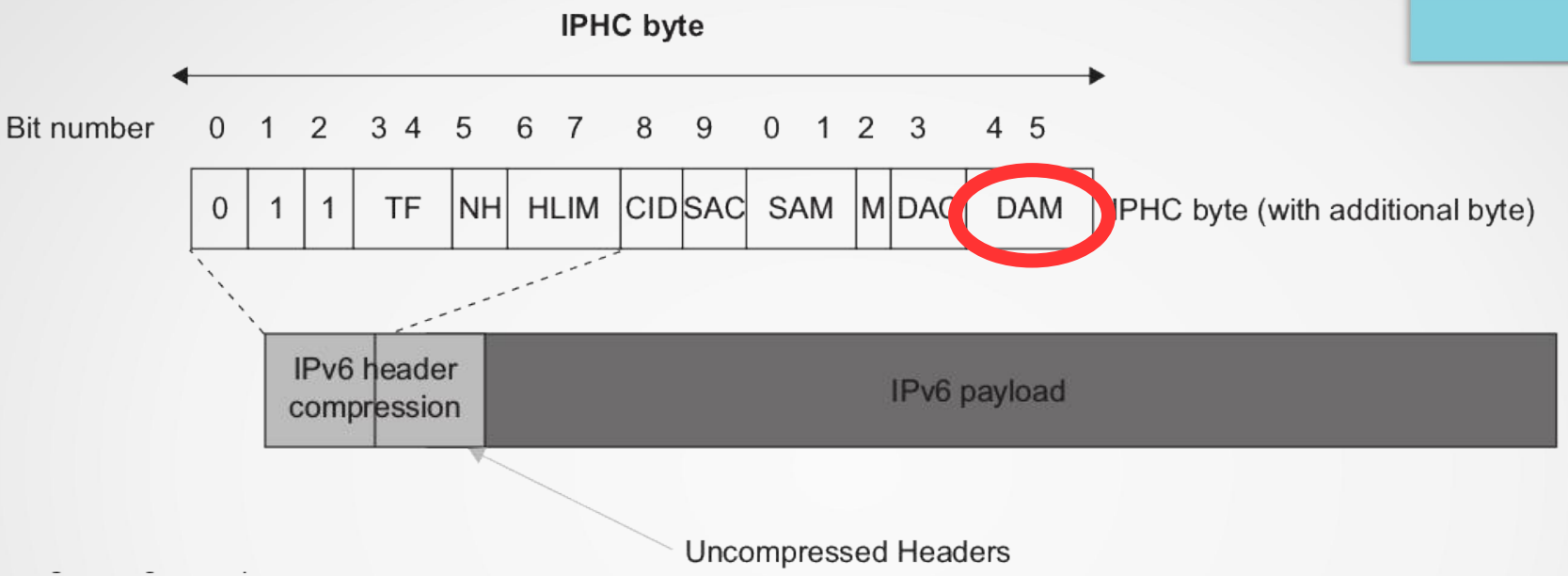
- M (multicast compression)
  - 0: destino no es multicast address.
  - 1: destino es multicast address.
- DAC (destination address compression)
  - 0: stateless.
  - 1: stateful basada en contexto.

IPHC



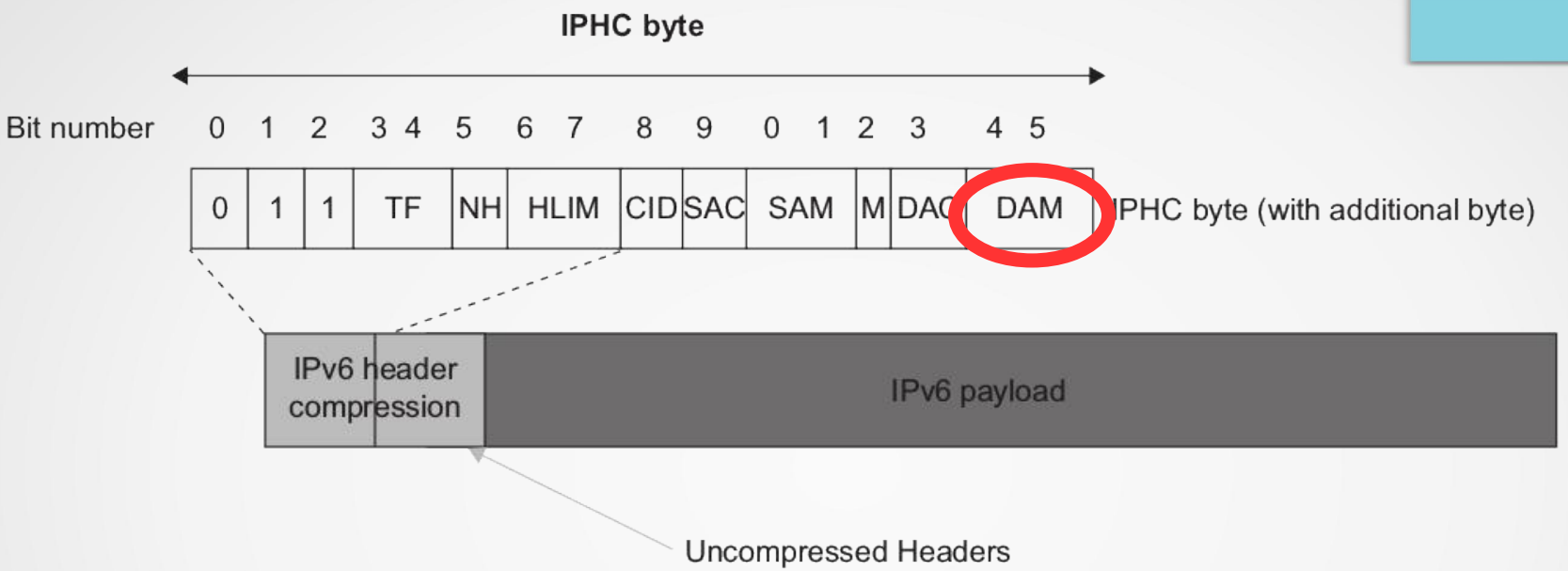
- DAM (destination address mode)
- Si M=0 & DAC=0 (no multicast y stateless) => igual a SAC = 0
  - 00: 128 bits.
  - 01: 64 bits.
  - 10: 16 bits.
  - 11: 0 bits.

IPHC



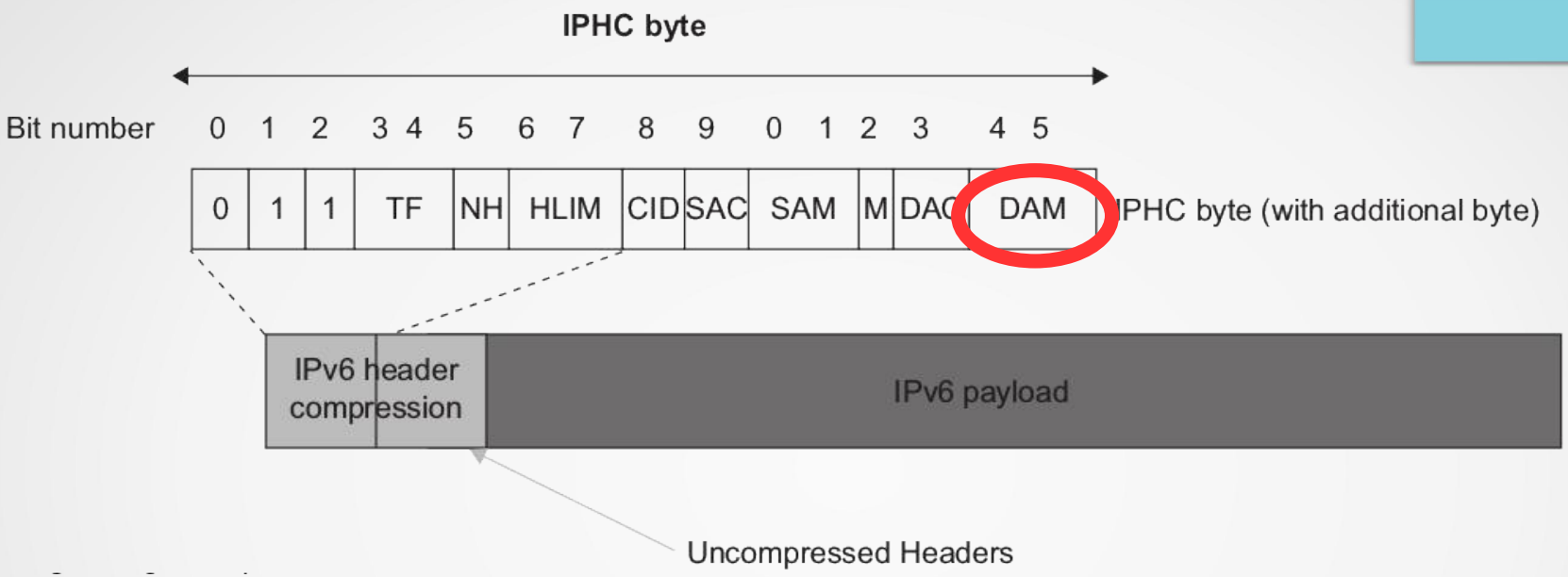
- DAM (destination address mode)
- Si M=0 & DAC=1 (no multicast y stateful basada en contexto)
  - 00: Reserved.
  - 01: 64 bits. Dirección inferida de contexto y de los 64 bits in-line.
  - 10: 16 bits. Similar a anterior.
  - 11: 0 bits.

IPHC



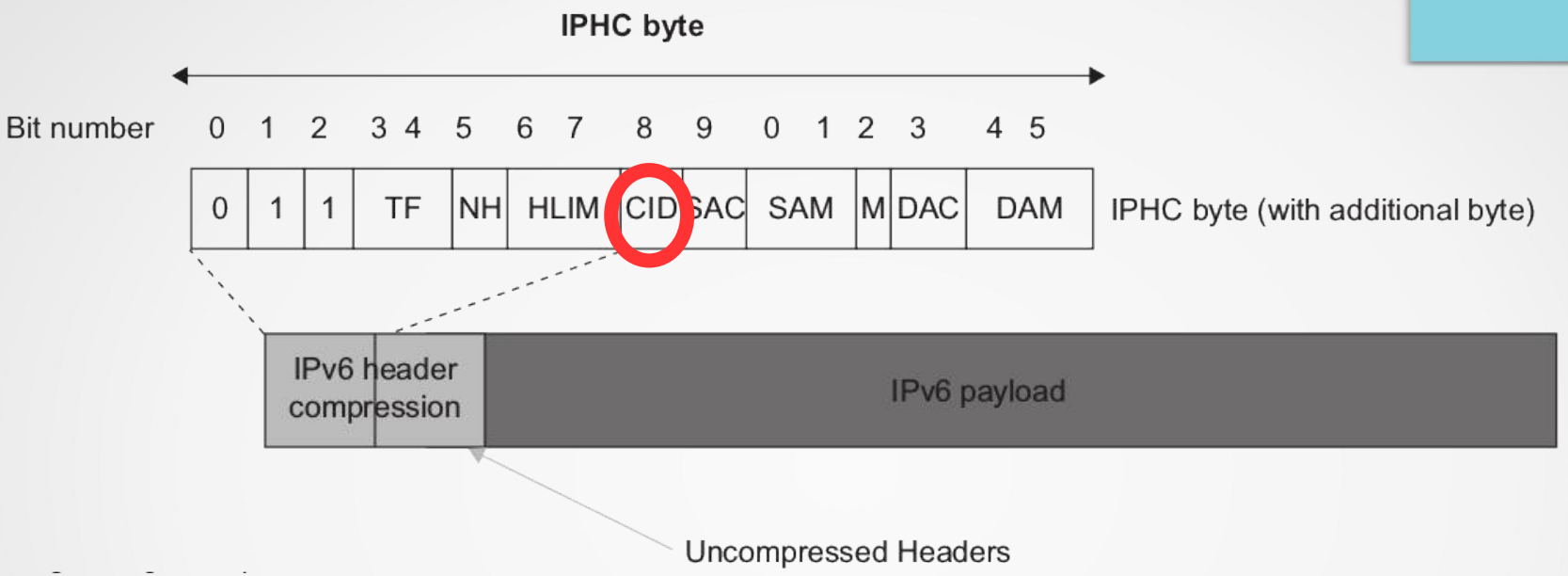
- DAM (destination address mode)
- Si M=1 & DAC=0 (multicast y stateless)
  - 00: 128 bits, in-line
  - 01: 48 bits, ffXX::00XX:XXXX:XXXX
  - 10: 32 bits. ffXX::00XX:XXXX.
  - 11: 8 bits. ff02::00XX

IPHC



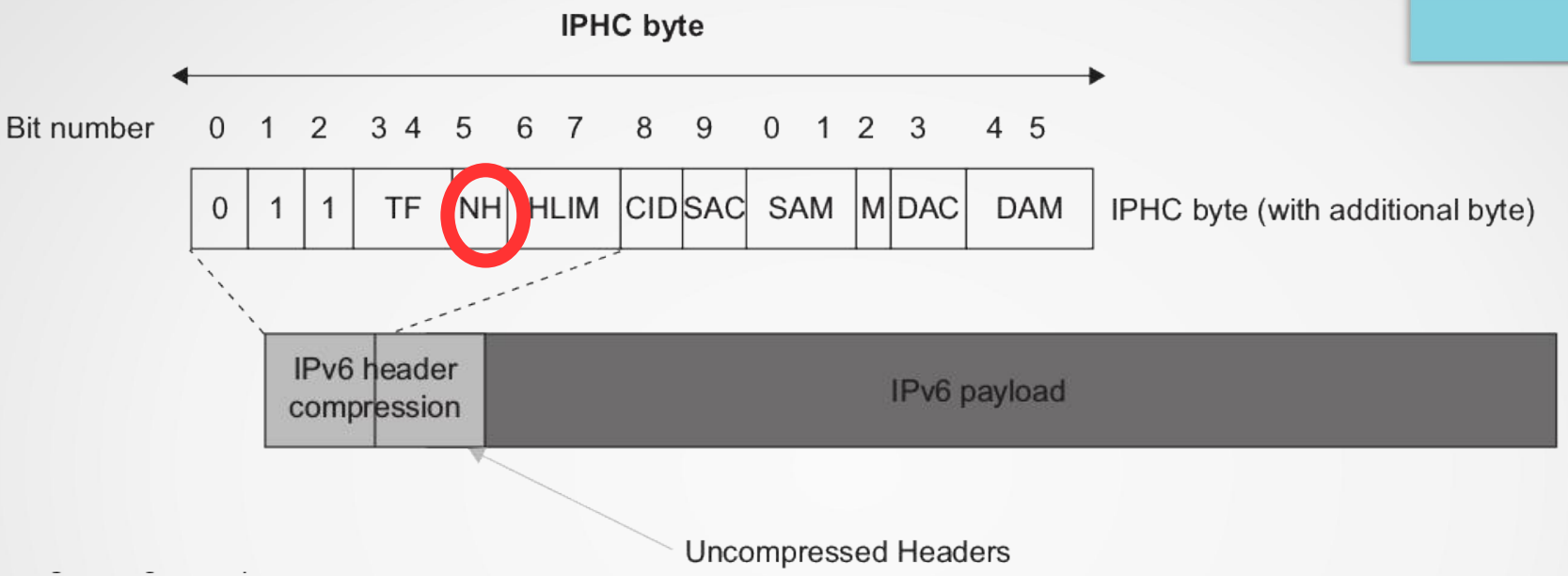
- DAM (destination address mode)
- Si M=1 & DAC=1 (multicast y stateful basada en contexto)
  - 00: 48 bits. ffXX:XXLL:PPPP:PPPP:PPPP:PPPP:XXXX:XXXX donde: X nibbles in-line, P and L se obtiene de contexto
  - 01: reserved
  - 10: reserved
  - 11: reserved

IPHC



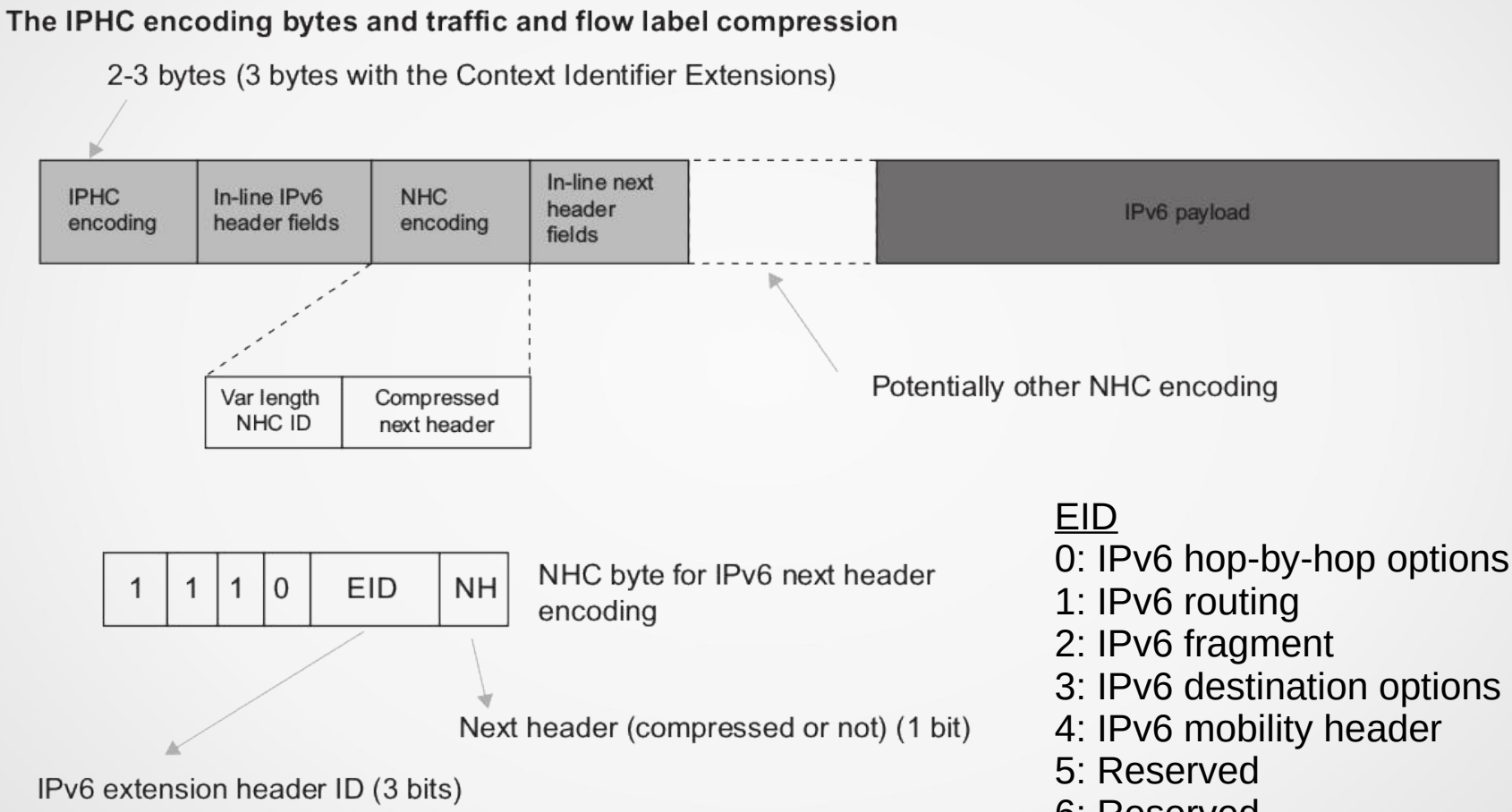
- Si CID = 1 => 1 byte adicional
  - SCI (source context identifier)
  - DCI (destination context identifier)
- 16 contextos

IPHC



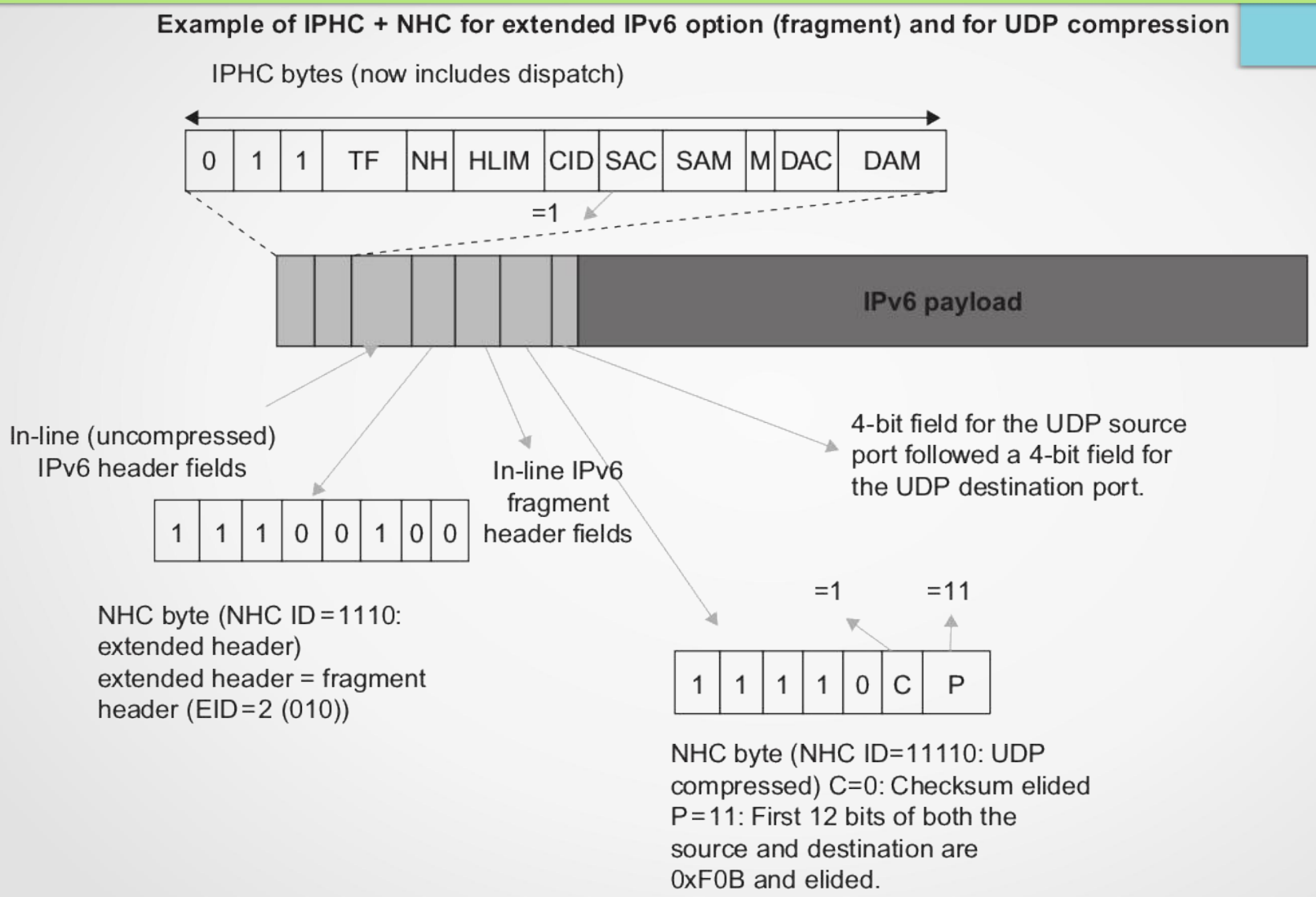
- Si NH = 1 => 1 byte adicional
  - LOWPAN_NHC (Next Header Compression)

NHC: next header compression



EID
0: IPv6 hop-by-hop options
1: IPv6 routing
2: IPv6 fragment
3: IPv6 destination options
4: IPv6 mobility header
5: Reserved
6: Reserved
7: IPv6 header

Resultado


Example of IPHC + NHC for extended IPv6 option (fragment) and for UDP compression
IPHC bytes (now includes dispatch)
0 1 1 TF NH HLIM CID SAC SAM M DAC DAM
=1
IPv6 payload
In-line (uncompressed) IPv6 header fields
In-line IPv6 fragment header fields
4-bit field for the UDP source port followed a 4-bit field for the UDP destination port.
1 1 1 0 0 1 0 0
NHC byte (NHC ID=1110: extended header) extended header = fragment header (EID=2 (010))
=1
=11
1 1 1 1 0 C P
NHC byte (NHC ID=11110: UDP compressed) C=0: Checksum elided P=11: First 12 bits of both the source and destination are 0xF0B and elided.

6lowpan en ContikiOS

- contiki/core/net/ipv6
  - sicslowpan.h
  - sicslowpan.c

Conclusiones

- 6LowPAN permite mandar paquetes IPv6:
  - Fragmentado y reensamblado de paquetes
  - Compresión de encabezados

Normalización

- IETF groups:
  - 6lowpan: IPv6 over Low power WPAN (concluded)
  - 6lo: IPv6 over Networks of Resource-constrained Nodes
  - 6tisch: IPv6 over the TSCH mode of IEEE 802.15.4e
  - RoLL: Routing over Low-power and Lossy networks (ROLL)

Referencias

- G. Montenegro, J. Hui, D. Culler, and N. Kushalnagar, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks," **RFC 4944**, Sept. 2007.
- P. Thubert and J. Hui, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks," **RFC 6282**, Sept. 2011.
- J. P. Vasseur and A. Dunkels, Interconnecting Smart Objects with IP: The Next Internet. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2010.
  - **Nota**: las figuras casi en su totalidad fueron tomadas de este libro.

Planificación clases

1) Introducción RSI
2) Plataformas de hardware
3) Arquitectura 6LoWPAN (IPv6)
4) Plataforma de software: Contiki-NG (parte 1)
5) Plataforma de software: Contiki-NG (parte 2)
6) Capa de aplicación: CoAP / MQTT
7) Capa de red: RPL
8) MAC / IEEE 802.15.4
9) **Capa adaptación 6LoWPAN**
10) Capa Fisica & antenas
11) IoT y las RSI

FIN... ¿preguntas?