Xarxes de Computadors II

Tema 2. Direccionamiento IPv6

Davide Careglio

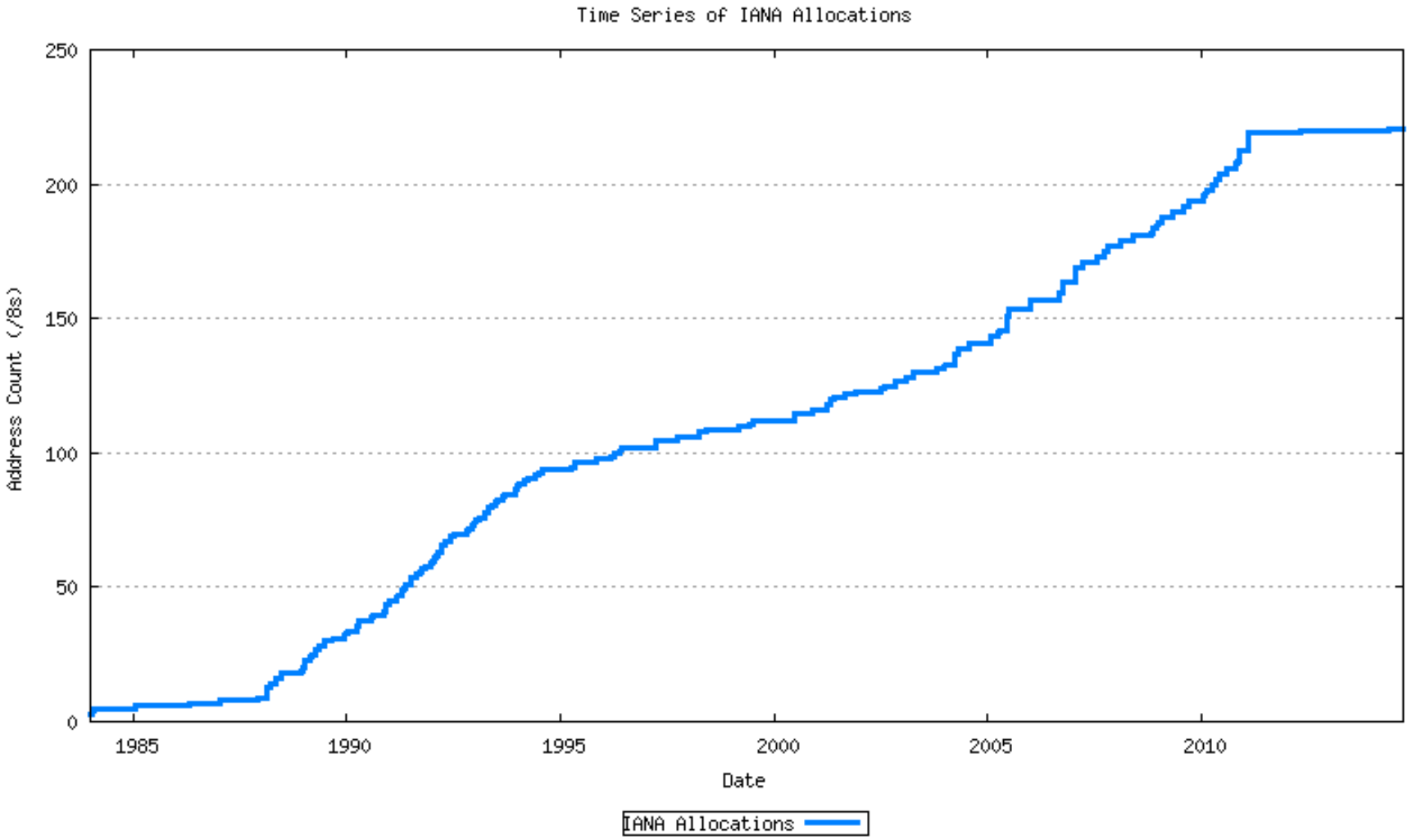
Temario

- ~~Tema 0. Repaso~~
- ~~Tema 1. Arquitectura y direccionamiento en Internet~~
- Tema 2. Direccionamiento IPv6
- Tema 3. Encaminamiento intra-dominio
- Tema 4. Multiprotocol Label Switching
- Tema 5. Encaminamiento inter-dominio
- Tema 6. Conceptos avanzados

Tema 2. Direccionamiento IPv6

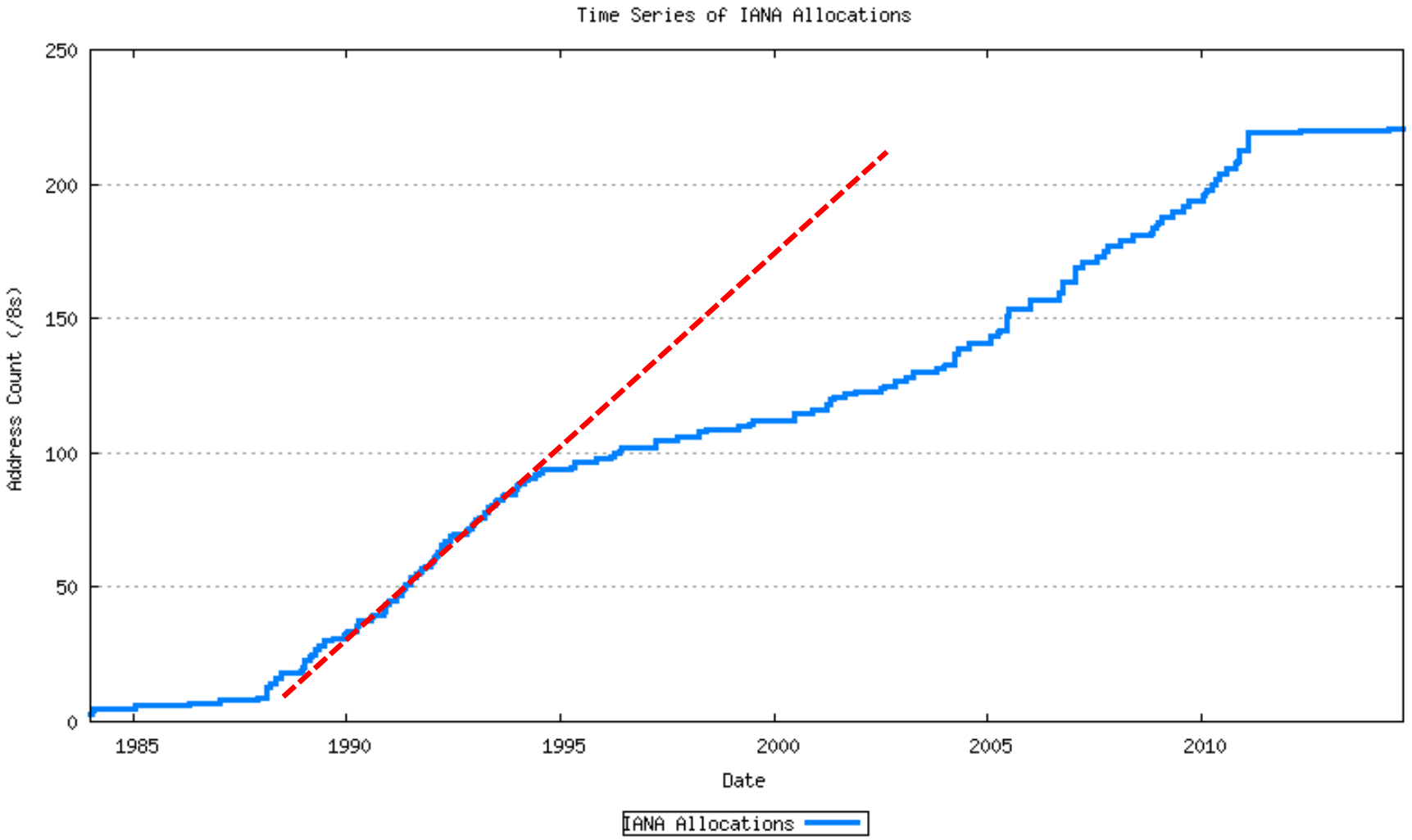
1. Evolución IPv4
2. Solución a corto plazo: ?
3. Solución a largo plazo: IPv6
4. Un poco de historia
5. Cabecera IPv6
6. Notación y formato IPv6
7. Tipos de datagramas
8. Direccionamiento: Stateful vs Stateless
9. InterfaceID
10. Compatibilidad IPv6 – IPv4

2.1 – Evolución IPv4



2.1 – Evolución IPv4

Situación en los ’90



2.2 – Solución a corto y largo plazo

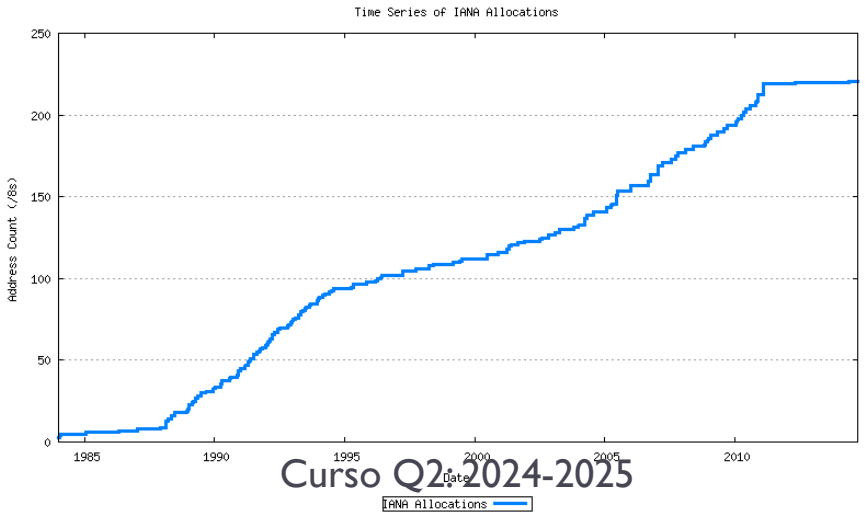
- Direcciones privadas y NAT
- Nuevo espacio de direccionamiento más amplio

2.2 – Solución a corto plazo
Direcciones privadas y NAT

- Idea
  - Si hay equipos en redes privadas → no necesitan una @IP publica única cada uno
- Solución
  - Se crean 3 grupos de @IP privadas (uno por clase)
  - Se pueden usar libremente en redes privadas
  - Se usa NAT para ir a Internet (red publica) para mantener la unicidad de las @IP
  - Se permite que varios equipos puedan compartir pocas (incluido una sola) @IP publica
  - Se reduce la necesidad de @IP publicas
- Problemas
  - Se necesitan tablas de traducción en los routers (la comunicación ya no es extremo-extremo ya que el router interviene los datagramas e incluido las cabeceras de transporte si implementa PAT)

2.2 – Solución a corto plazo
Direcciones privadas y NAT

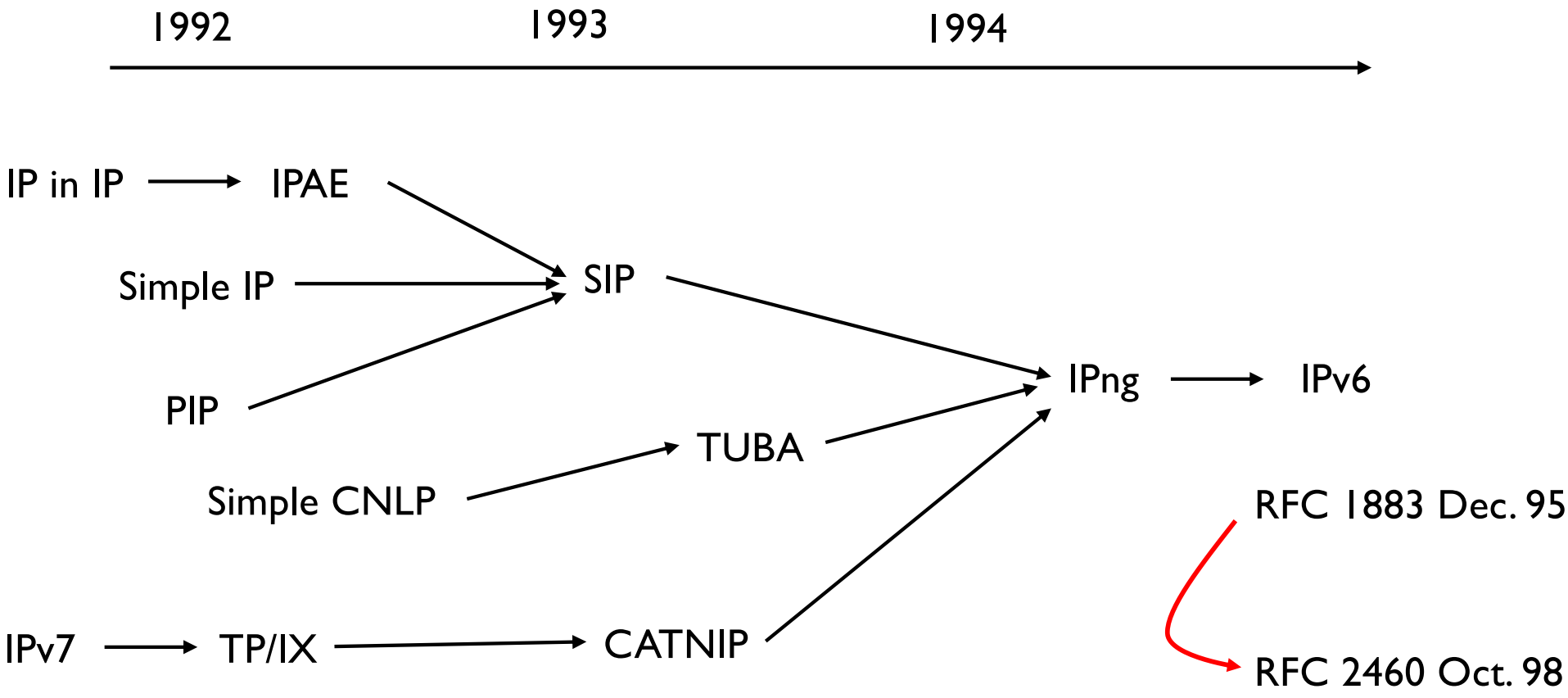
- Problemas
  - Se necesitan tablas de traducción en los routers
  - La comunicación ya no es extremo-extremo ya que el router interviene los datagramas, incluida las cabeceras de transporte TCP/UDP si implementa PAT (ya no es un equipo puramente de nivel 3)
  - Multicast con NAT difícil de configurar
  - VPN con IPsec (autentificación y encriptación)
    - Si el router debe modificar las @IP o los puertos, debe saber como desencriptar → posible punto vulnerable
  - Algunas aplicaciones no funcionan si hay NAT en el medio (como VoIP)
  - Finalmente, es una solución a corto plazo como se ve en la grafica



2.3 – Solución a largo plazo
Aumentar el espacio de direccionamiento

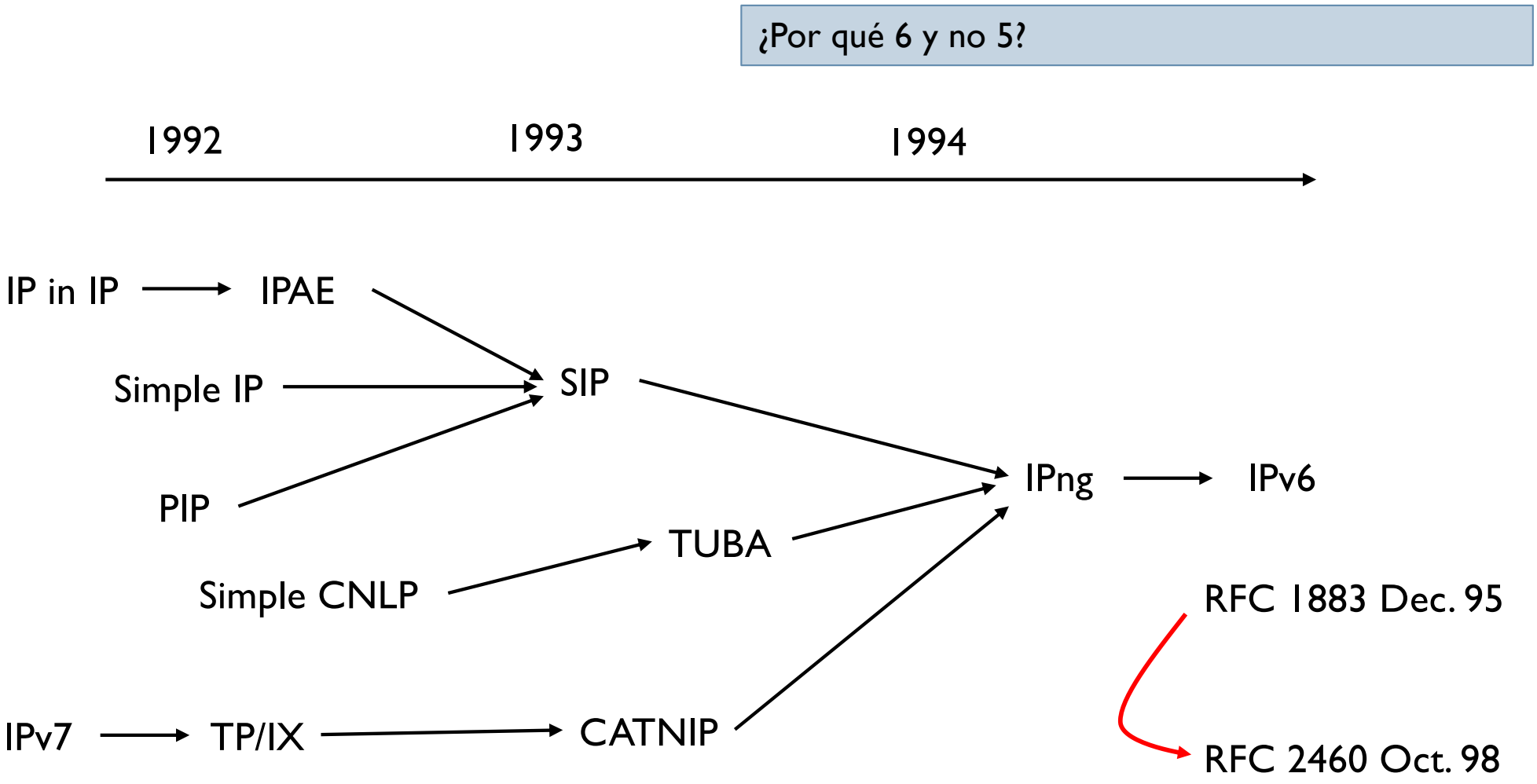
- Idea
  - Ya que el problema es que se agotan las @IP, crear un nuevo espacio de direccionamiento más amplio
- Solución
  - Crear una nueva versión de IP pensada para ser sostenible y aguantar Internet en los próximos 20/30 años

2.4 – Un poco de historia
Diferentes propuestas en IETF



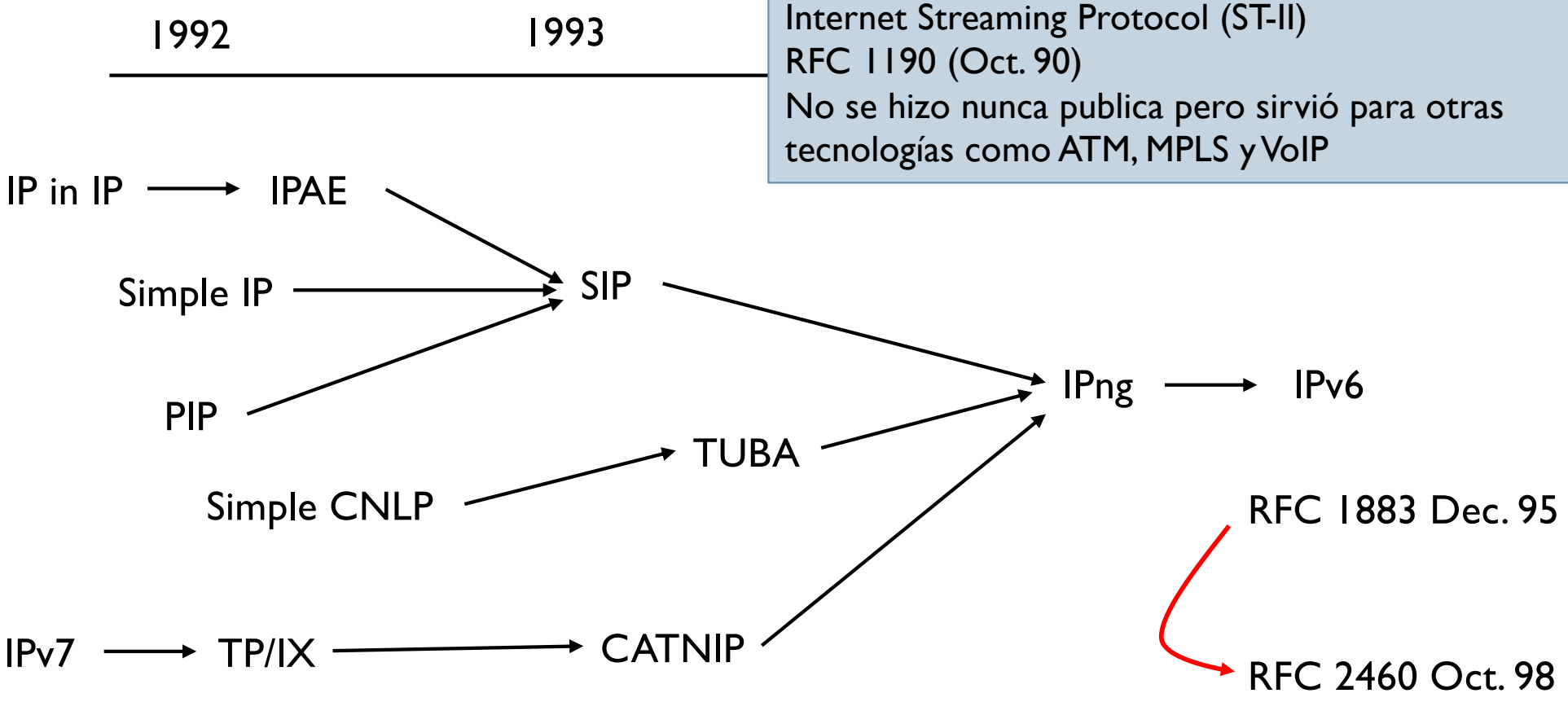
2.4 – Un poco de historia
Diferentes propuestas en IETF

¿Por qué 6 y no 5?



2.4 – Un poco de historia
Diferentes propuestas en IETF

¿Por qué 6 y no 5?
En 1990 se reservó el 5 para una red experimental Internet Streaming Protocol (ST-II)
RFC 1190 (Oct. 90)
No se hizo nunca publica pero sirvió para otras tecnologías como ATM, MPLS y VoIP



2.4 – Un poco de historia
Diferentes propuestas en IETF

- Diferentes propuestas
  - 64 bits para las direcciones
  - Longitud variable entre 64 y 160 bits
  - Finalmente se fija a 128 bits
    → $2^{128} \cong 10^{28}$ direcciones IPv6 por persona en el mundo
    → posibilidad de conectar a Internet cualquier dispositivo electronico
    → Internet of Things (IoT) e Internet of Everything (IoE)

2.4 – Un poco de historia
IPv6 la ganadora

- No compatible con IPv4
  - Por eso la transición es complicada IPv4 → IPv6
  - Durante este tiempo Internet funciona con ambos formato
    - La mayoría de los OS ya soportan IPv6
    - Los equipos hardware se van actualizando y/o substituyendo poco a poco
    - Los servicios se van actualizando

- Comprobar si tenemos IPv6 y estamos conectados a una red IPv6
  - http://test-ipv6.com

2.4 – Un poco de historia
IPv6 la ganadora

- El 8 de Junio de 2011, Internet Society (ISOC) celebró el World IPv6 day
  - Los sitios web mas importantes (~1000)
    google, youtube, facebook, yahoo, microsoft, cnn, bbc, etc.
    hicieron sus paginas accesibles vía IPv6 (e IPv4 claro) durante 24 horas
  - Muchos de ellos siguen funcionando con IPv6
  - http://www.internetsociety.org/ipv6/archive-2011-world-ipv6-day/

- El 6 de Junio de 2012, Internet Society (ISOC) abre el World IPv6 launch
  - Una plataforma para fomentar el despliegue de IPv6 en el mundo
  - Más operadoras, servicios y fabricantes se apuntan al IPv6
  - Mantiene medidas en tiempo real sobre el despliegue
  - http://www.worldipv6launch.org

2.4 – Un poco de historia

IPv6 situación actual

IPv6 Adoption By Country / Region

**Country data ranked by % of IPv6 connections from that country.*

RANK	IPV6%	COUNTRY / REGION
1	62.4%	India
2	62%	Malaysia
3	57.9%	Tokelau
4	56.4%	Uruguay
5	55.9%	Viet Nam
6	53.7%	Germany
7	53.2%	Montserrat
8	52.7%	France
9	51.4%	Saudi Arabia
10	49.1%	Japan
103	7.6%	Spain

IPv6 Adoption By Networks

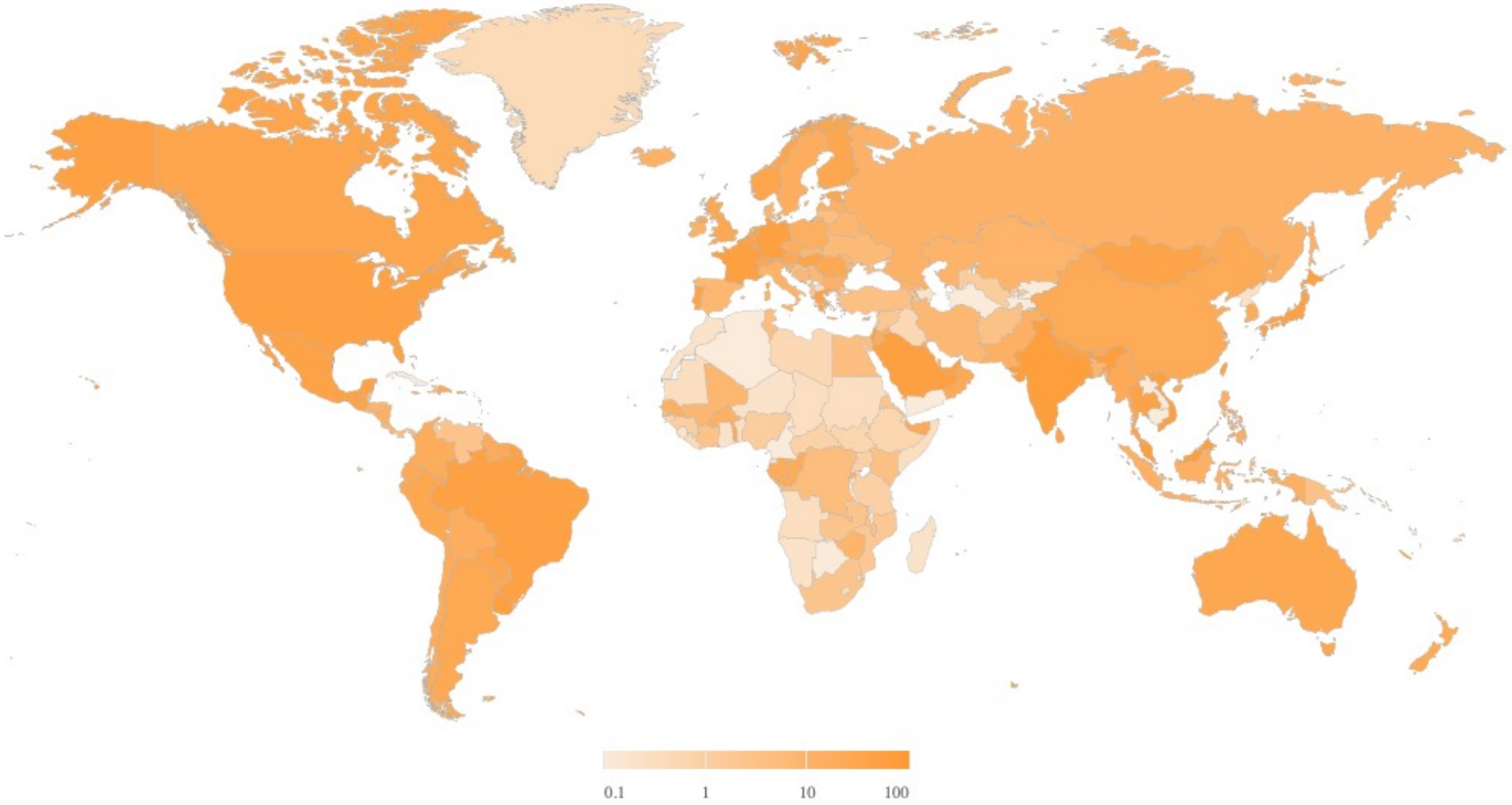
**Networks data is limited to the top 200 networks ranked by total IPv6 hits to platform.*

RANK	ipv6-adoption	NETWORK
56	100%	NIPPON TELEGRAPH AND TELEPHONE EAST CORPORATION
191	99.1%	ESTOXY-OU
139	98.2%	Facebook Inc.
37	97.9%	AKAMAI-AS
20	96.4%	Cloudflare
6	92.9%	Reliance Jio Infocomm Limited
4	92.4%	T-Mobile
184	90.1%	Google Inc..
57	90%	SK Telecom
101	89.9%	Rakuten Mobile

Fuente imagen: https://www.akamai.com/visualizations/state-of-the-internet-report/ipv6-adoption-visualization, acceso 2/2024

2.4 – Un poco de historia

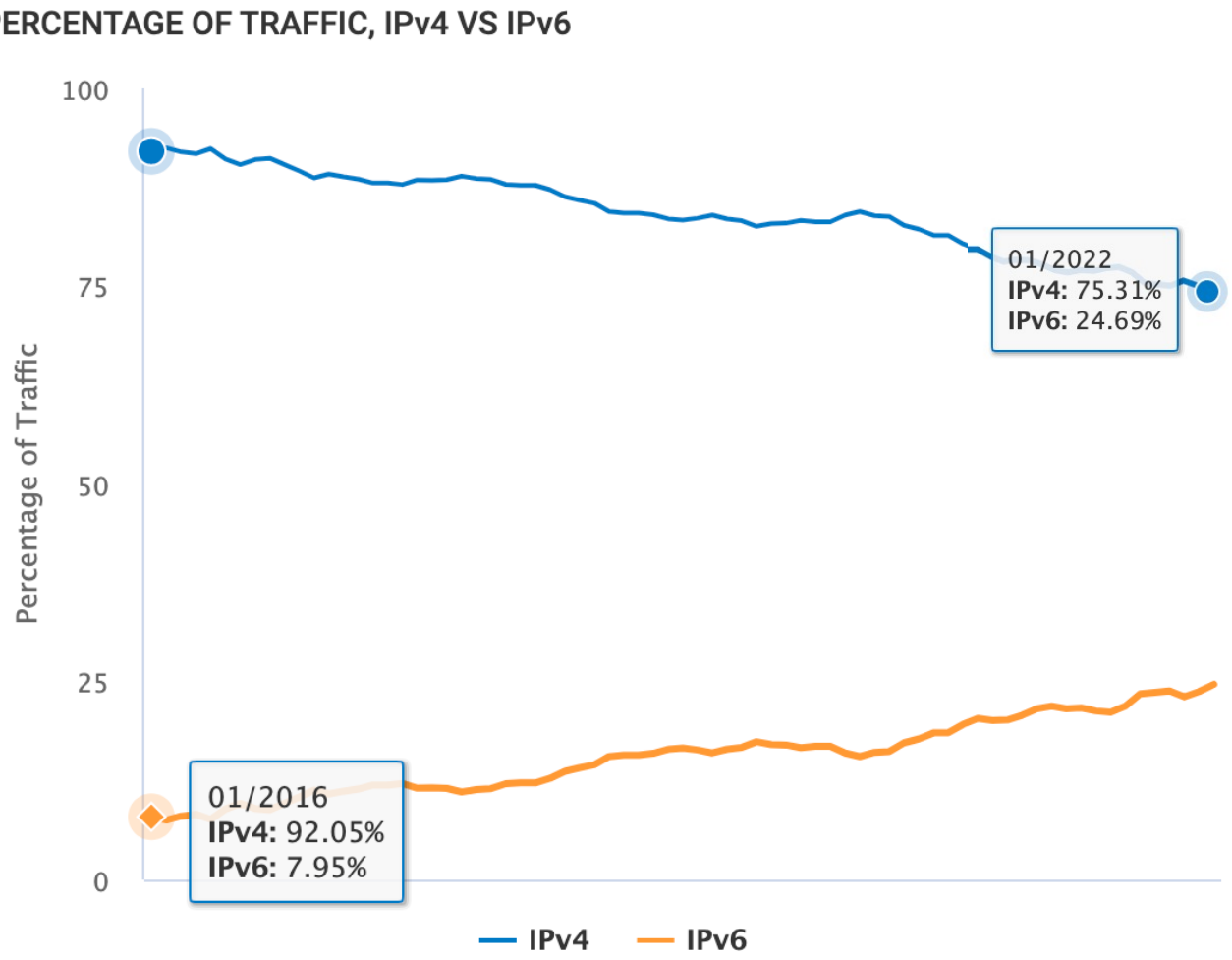
IPv6 situación actual



Fuente imagen: https://www.akamai.com/visualizations/state-of-the-internet-report/ipv6-adoption-visualization, acceso 2/2024

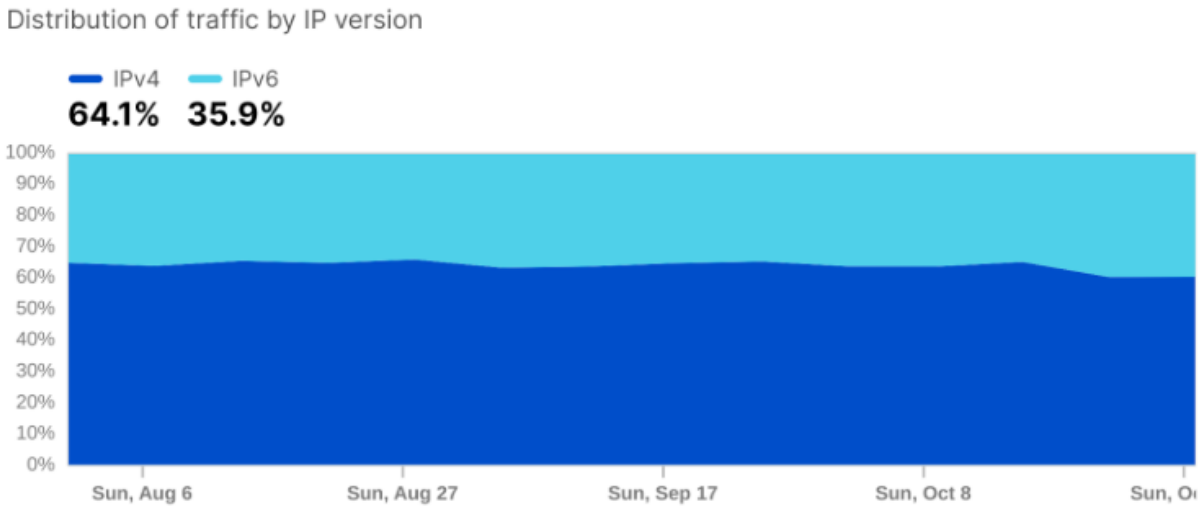
2.4 – Un poco de historia

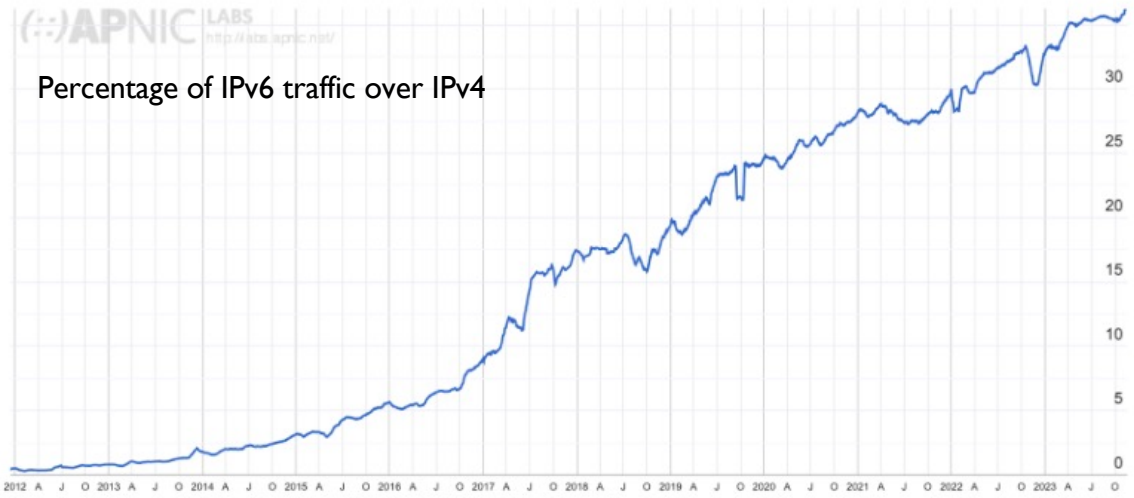
IPv6 situación actual



Fuente imagen: https://www.akamai.com/visualizations/dns-trends-and-traffic, acceso 1/2022

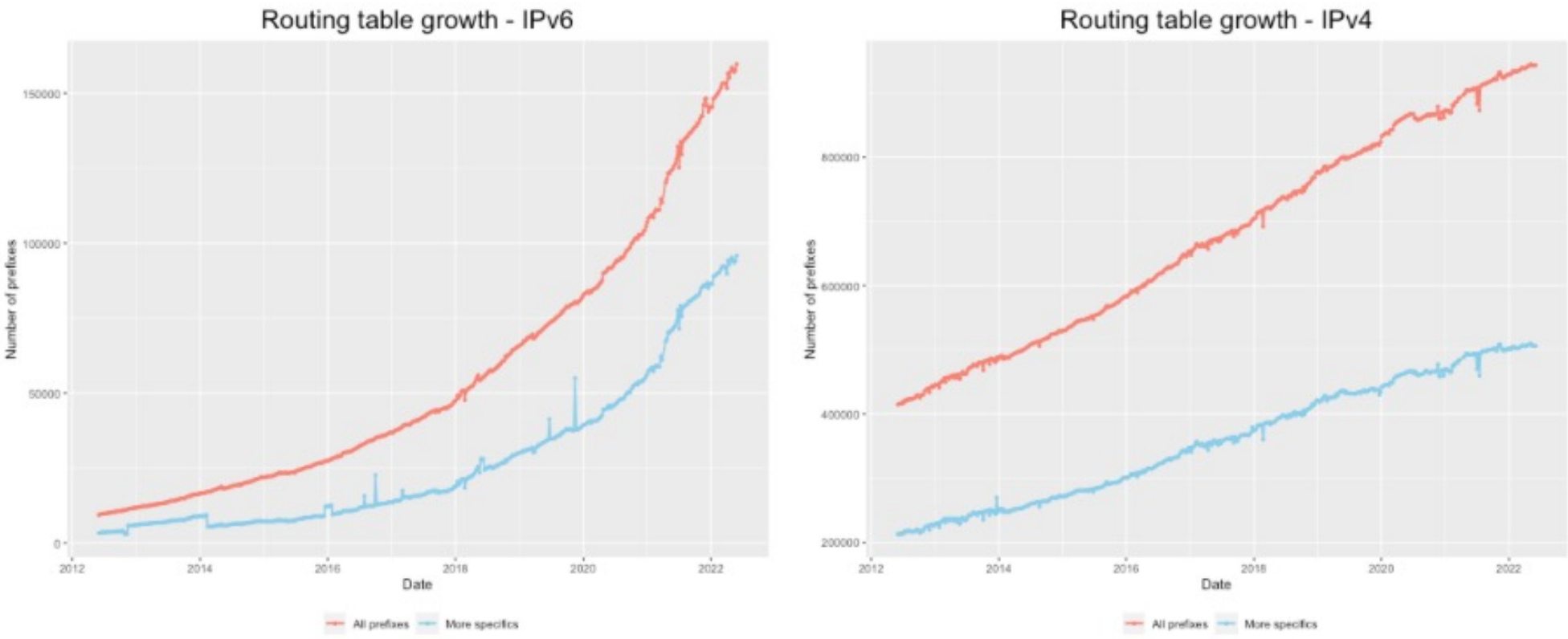
2.4 – Un poco de historia
IPv6 situación actual





Fuente imagen: https://blog.cloudflare.com/ipv6-from-dns-pov, acceso 2/2024

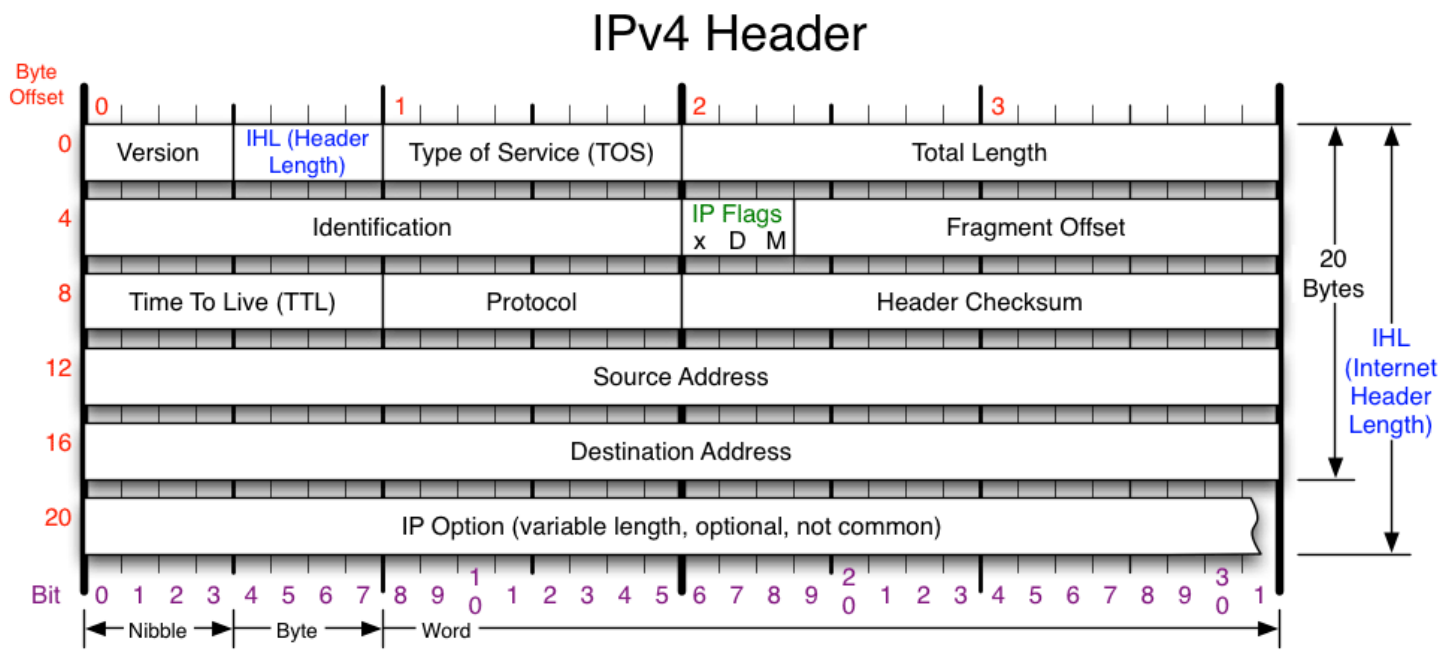
2.4 – Un poco de historia
IPv6 situación actual



Fuente imagen: https://labs.ripe.net/author/wilhelm/ipv6-10-years-out-an-analysis-in-users-tables-and-traffic/, acceso 2024

2.5 - Cabecera IPv6

Se analiza como le ha ido a IPv4



Version

Version of IP Protocol. 4 and 6 are valid. This diagram represents version 4 structure only.

Header Length

Number of 32-bit words in TCP header, minimum value of 5. Multiply by 4 to get byte count.

Protocol

IP Protocol ID. Including (but not limited to):

1 ICMP	17 UDP	57 SKIP
2 IGMP	47 GRE	88 EIGRP
6 TCP	50 ESP	89 OSPF
9 IGRP	51 AH	115 L2TP

Total Length

Total length of IP datagram, or IP fragment if fragmented. Measured in Bytes.

Fragment Offset

Fragment offset from start of IP datagram. Measured in 8 byte (2 words, 64 bits) increments. If IP datagram is fragmented, fragment size (Total Length) must be a multiple of 8 bytes.

Header Checksum

Checksum of entire IP header

IP Flags

x D M

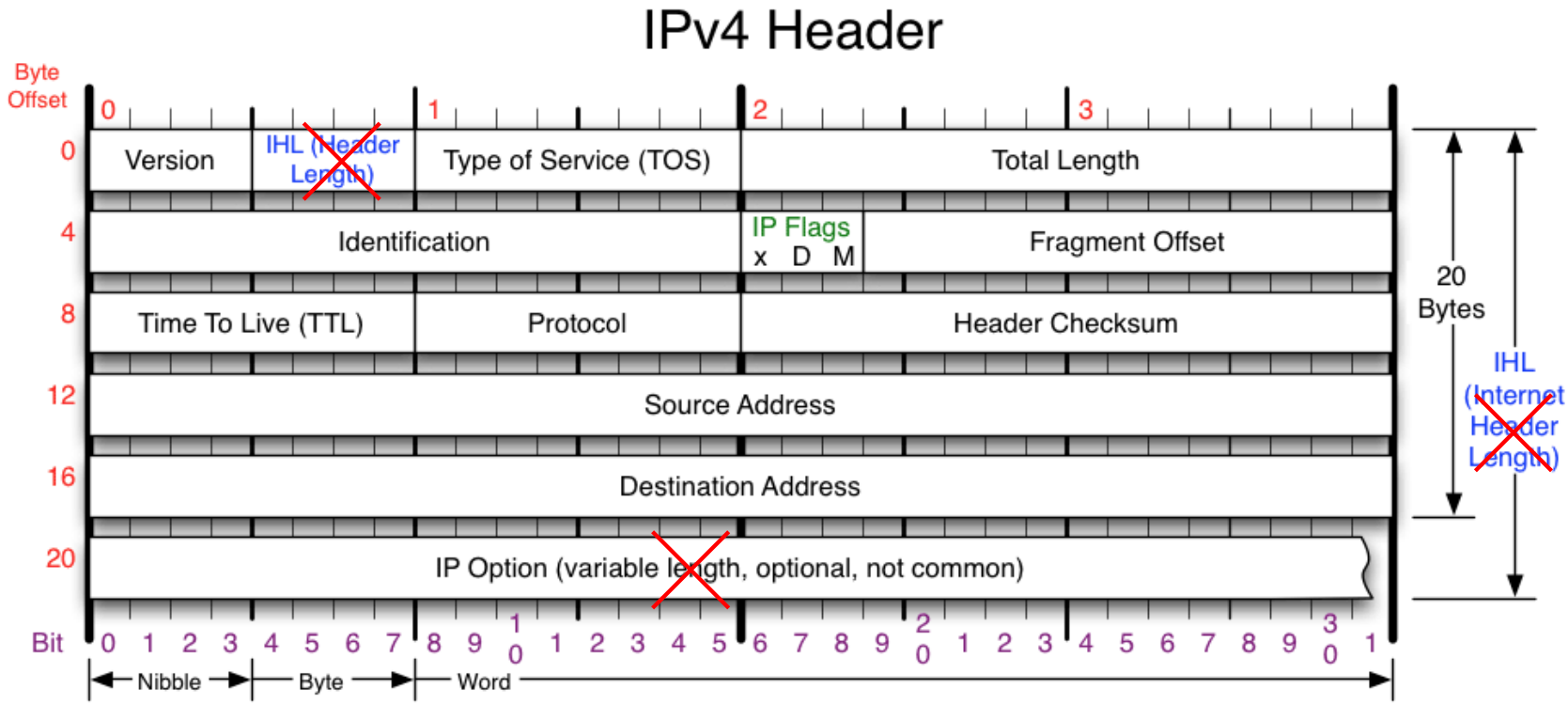
x 0x80 reserved (evil bit)
D 0x40 Do Not Fragment
M 0x20 More Fragments follow

RFC 791

Please refer to RFC 791 for the complete Internet Protocol (IP) Specification.

2.5 - Cabecera IPv6

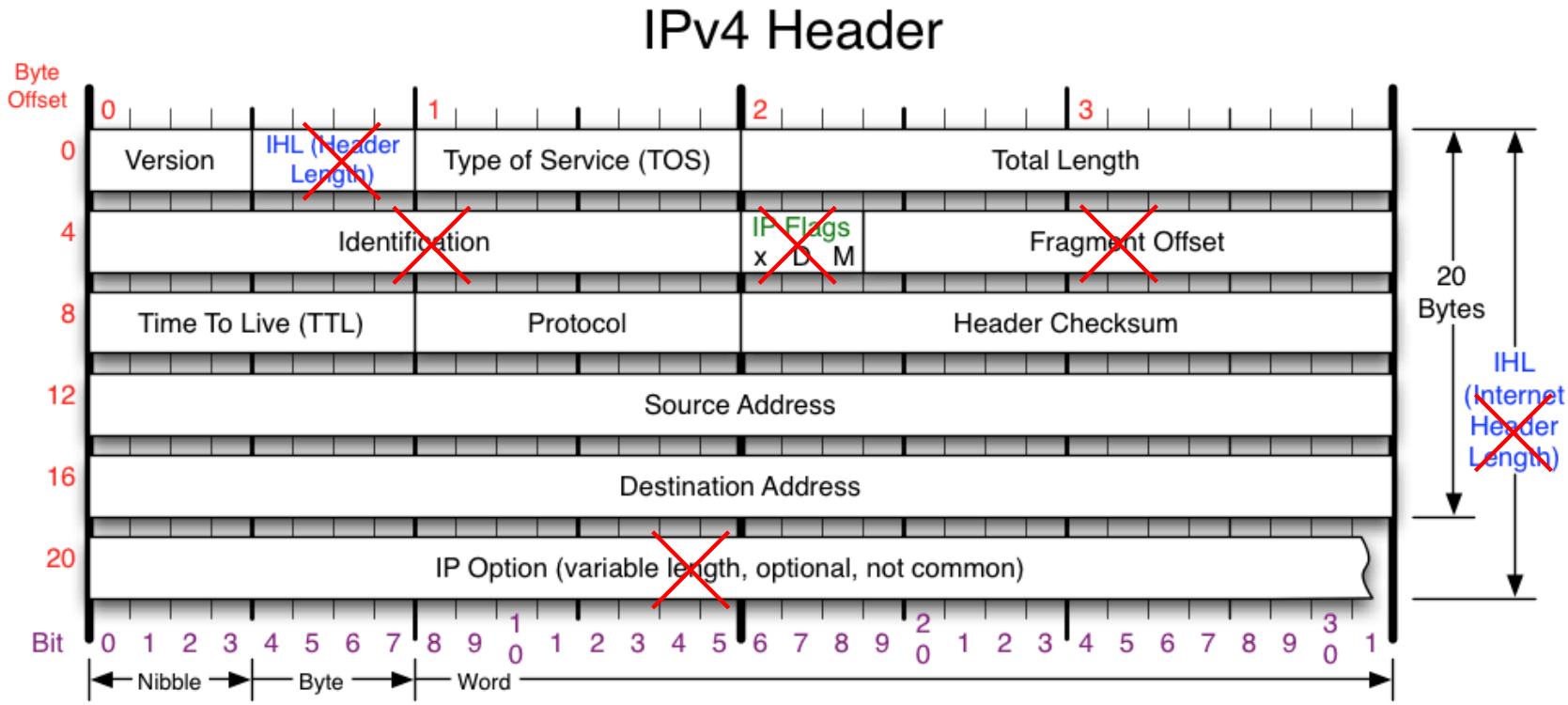
Se analiza como le ha ido a IPv4



- Longitud cabecera fija → fuera Header Length y Option

2.5 - Cabecera IPv6

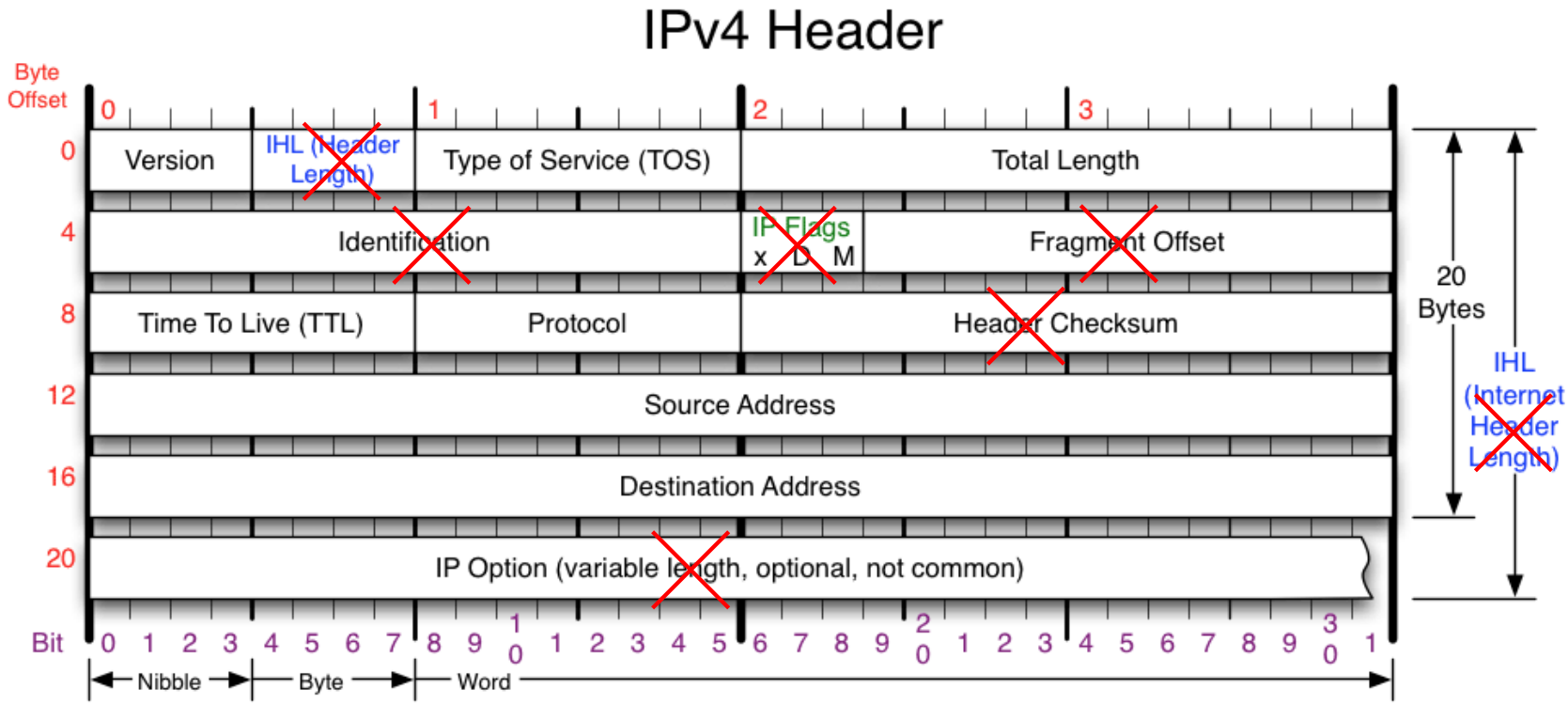
Se analiza como le ha ido a IPv4



- Fragmentación se evita siempre que se pueda
  → fuera Identification, Flags y Fragmen Offset

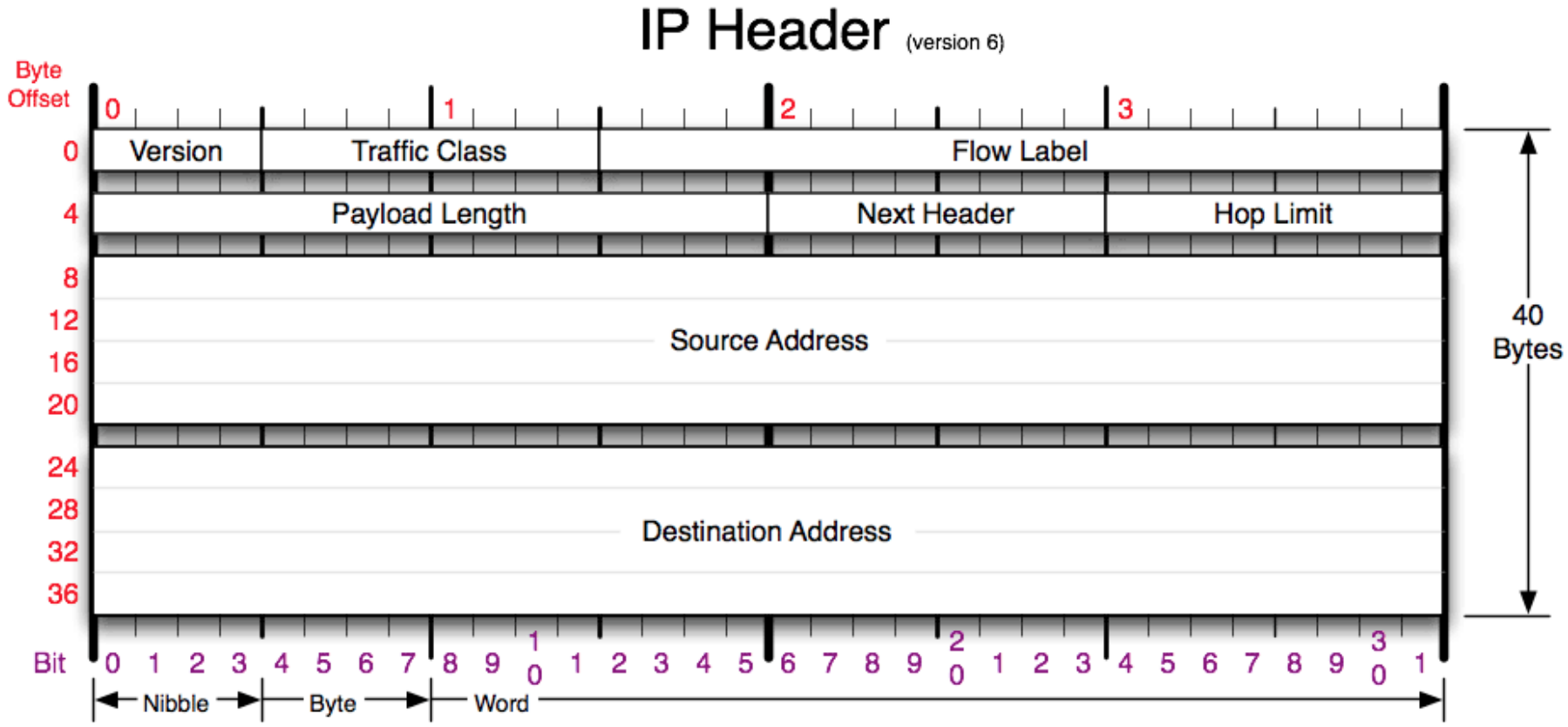
2.5 - Cabecera IPv6

Se analiza como le ha ido a IPv4



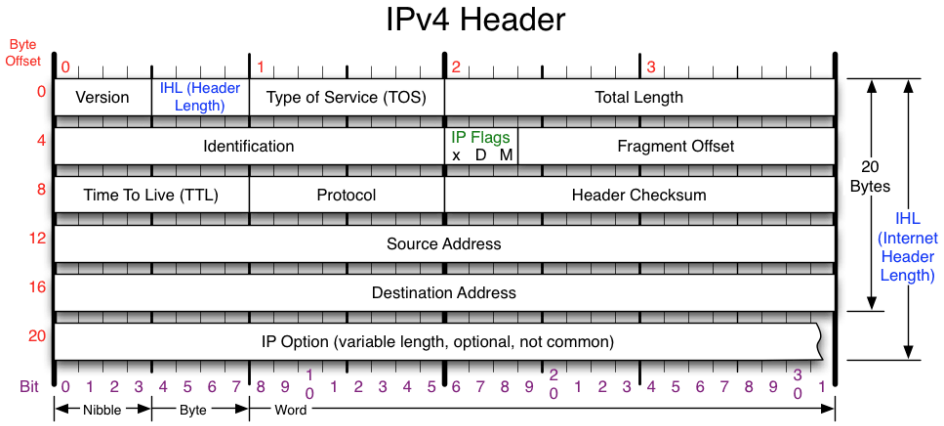
- Ya se hace un control de error en Transporte e Interfaz de Red
  → fuera Header Checksum

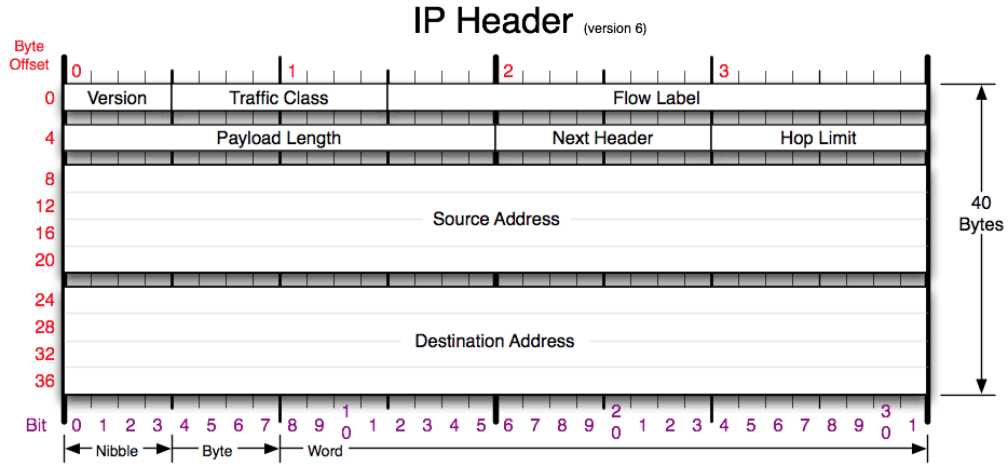
2.5 - Cabecera IPv6



2.5 - Cabecera IPv6

IPv4 vs IPv6

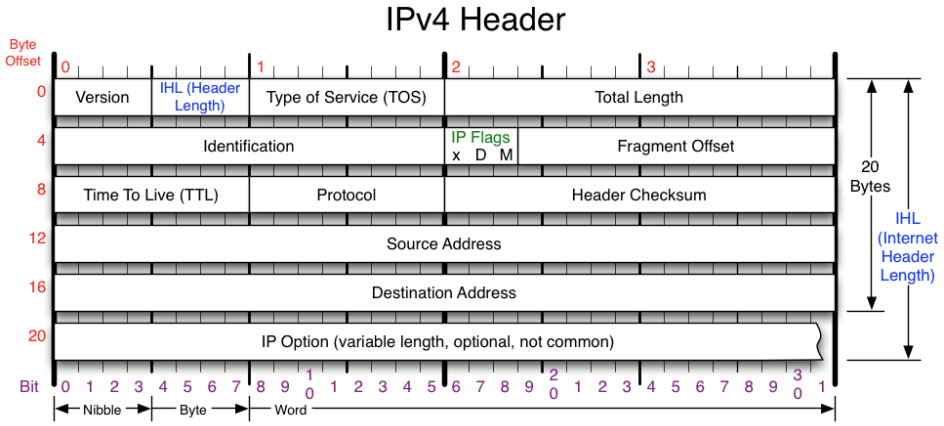


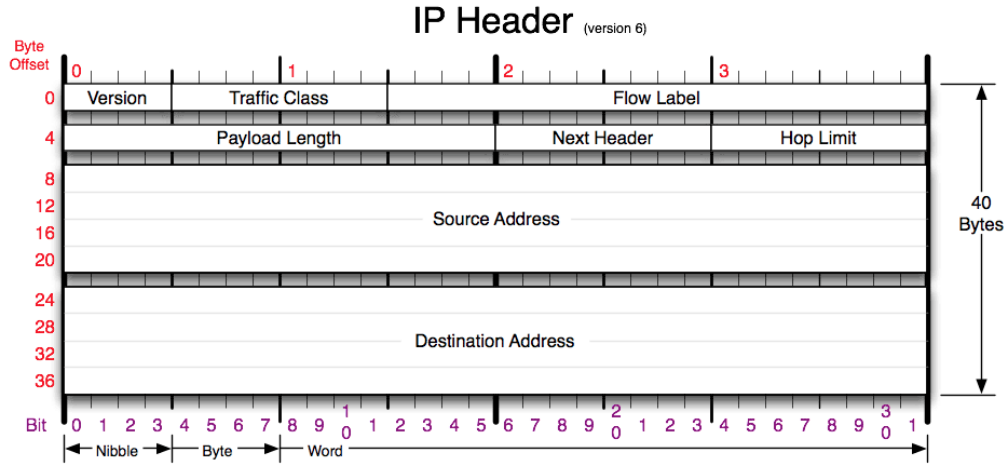


- Versión: ahora 6
- Longitud cabecera: en IPv6 la cabecera se fija a 40 bytes
- Tipo de servicio: ahora clase de servicio en IPv6 (8 bits)
  - Como en IPv4, este campo depende si los routers están configurados para soportar este servicio
  - 6 bits → Differentiated Service (DiffServ) RFC 2474 → paquetes con prioridades diferentes según unos Code Point (DSCP)
  - 2 bits → Explicit Congestion Notification (ECN) RFC 3168
    - un router puede marcar un paquete cuando está en congestión
    - cuando el destino recibe este paquete y tiene que contestar al origen, crea un eco de este nivel de congestión y lo incluye en su paquete para el origen
    - Al recibir este paquete, el origen baja su tasa de envío (funciona conjuntamente con TCP bajando el tamaño de la ventana de transmisión)

2.5 - Cabecera IPv6

IPv4 vs IPv6



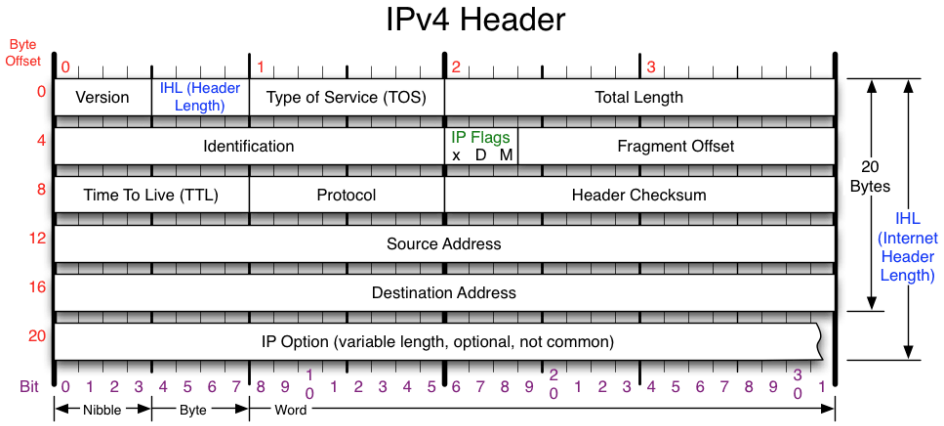


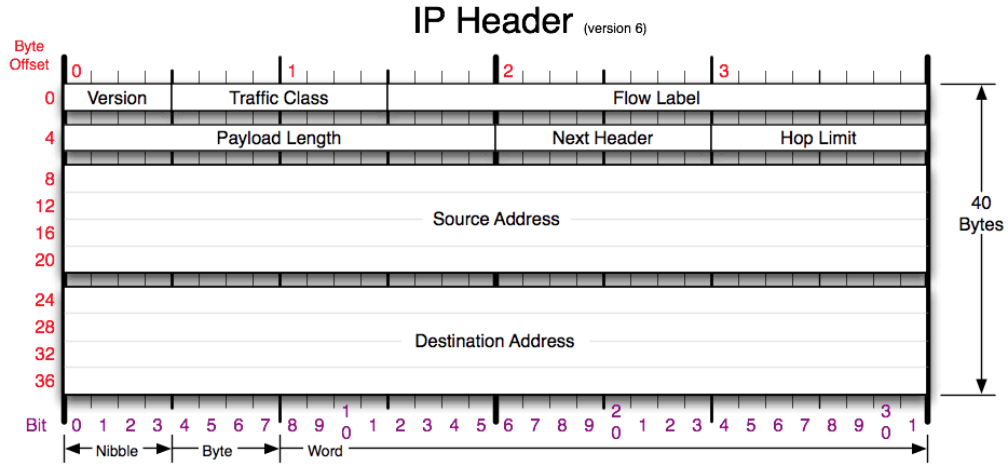
- Etiqueta de flujo: nuevo en IPv6
  - Para facilitar el reconocimiento de paquetes que pertenecen a un mismo flujo
    - Secuencia de paquetes relacionados entre sí (por ejemplo de una misma aplicación o de un mismo servicio)
    - Paquetes de una misma sesión TCP
  - Todos los paquetes de un mismo flujo se marcan con el mismo valor
  - Un valor 0 significa que no se está usando este campo

  → Permite aplicar un control/filtrado basado en flujo y no por paquete (en principio más rápido)
- Longitud total: en IPv6 no se consideran los 40 bytes de la cabecera ya que son fijos y solo cuentan los bytes del payload

2.5 - Cabecera IPv6

IPv4 vs IPv6

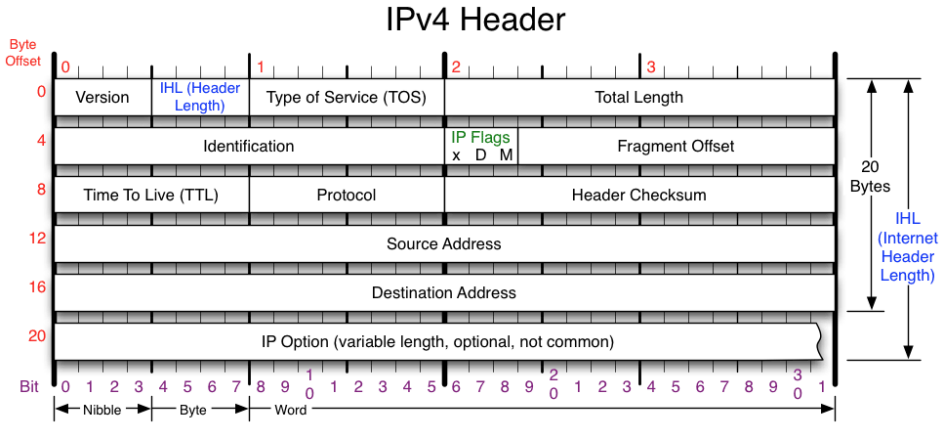


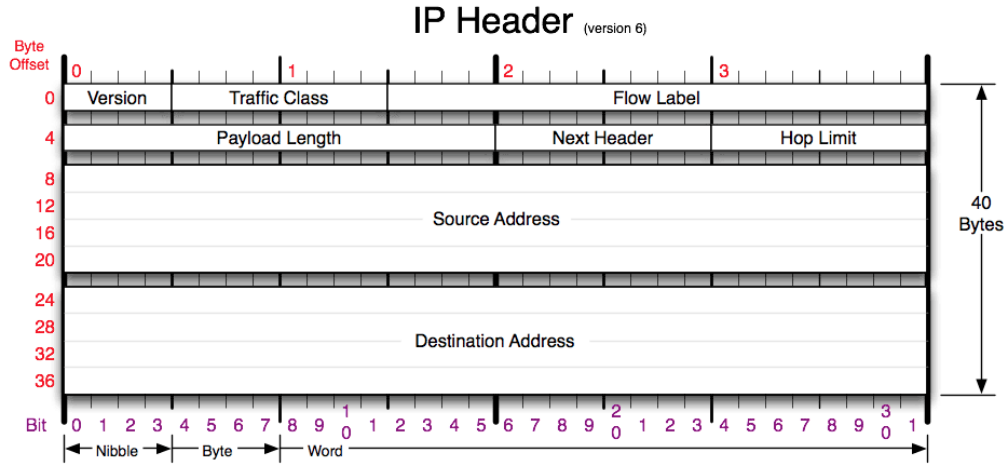


- Identificación, flags, fragmentos: se usan para fragmentar IPv4
  - En IPv6 se usa un método distinto y se eliminan de la cabecera
  - Ya que se intenta evitar al máximo la fragmentación y solo puntualmente se necesita fragmentar, es ineficiente tener siempre estos campos en la cabecera
- Tiempo de vida: ahora se llama Limite de Saltos
  - Mismo funcionamiento que en IPv4
  - El origen pone un valor que indica el número máximo de routers por donde puede pasar el datagrama
  - Cada router disminuye este valor de 1
  - Si al hacer esta operación este campo vale 0, el datagrama se descarta

2.5 - Cabecera IPv6

IPv4 vs IPv6



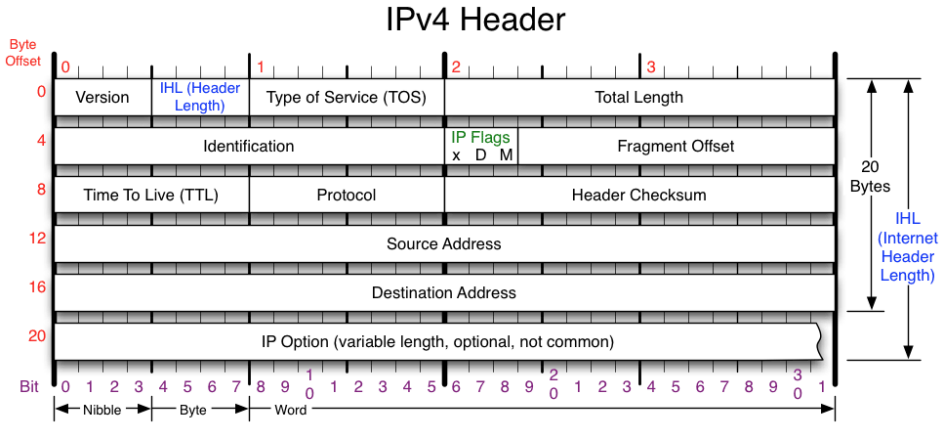


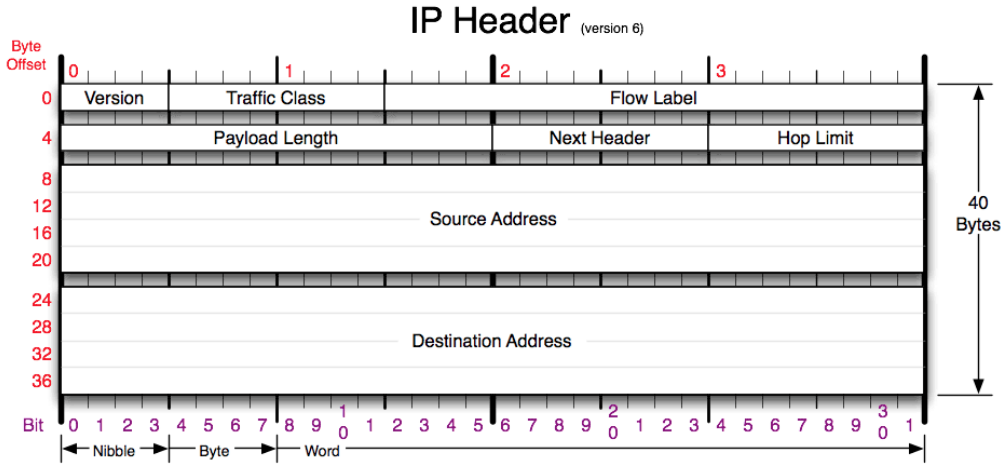
- Protocolo
  - En IPv4 indica el protocolo del payload (lo que viene de la capa superior y se encapsula en IP)

    Por ejemplo: 6 → TCP 17 → UDP 1 → ICMP
  - En IPv6 se substituye con el campo Siguiente Cabecera
  - Esta campo hace las funciones de Protocolo y Opciones de IPv4
  - Veremos luego
- Checksum
  - En IPv4, sirve como control de error de lectura de los bits de la cabecera
  - En IPv6 no se usa ya que hay controles similares en otros niveles (CRC en Ethernet y WiFi, checksum en TCP/UDP) → se considera entonces un control redundante y se elimina también para descargar el router de la tarea de comprobar el checksum

2.5 - Cabecera IPv6

IPv4 vs IPv6



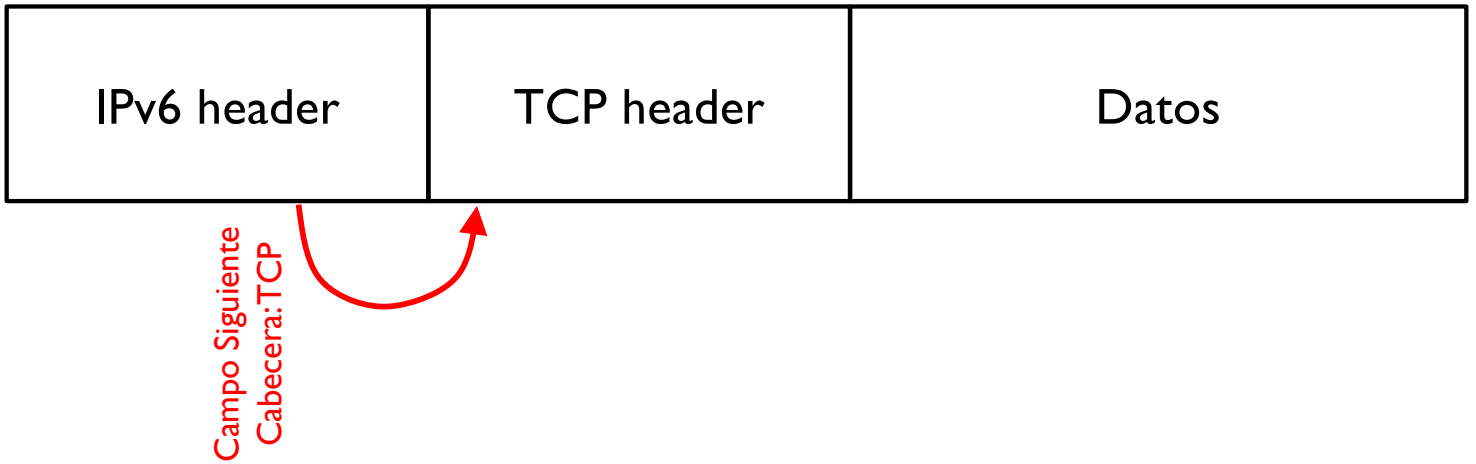


- IP origen y destino
  - Pasa de 32 a 128 bits
- Opciones
  - No se incluyen en la cabecera IPv6
  - Si se quieren añadir opciones, se usa un método diferente (luego veremos)

2.5 - Cabecera IPv6

Siguiente Cabecera IPv6

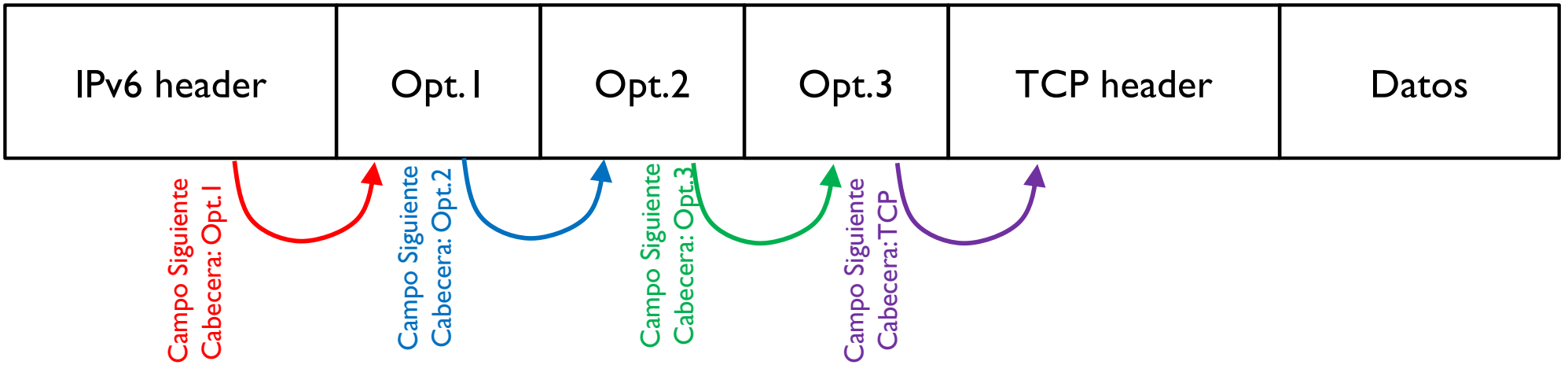
- Funcionamiento "normal"



2.5 - Cabecera IPv6

Siguiente Cabecera IPv6

- Añadir opciones en cascada



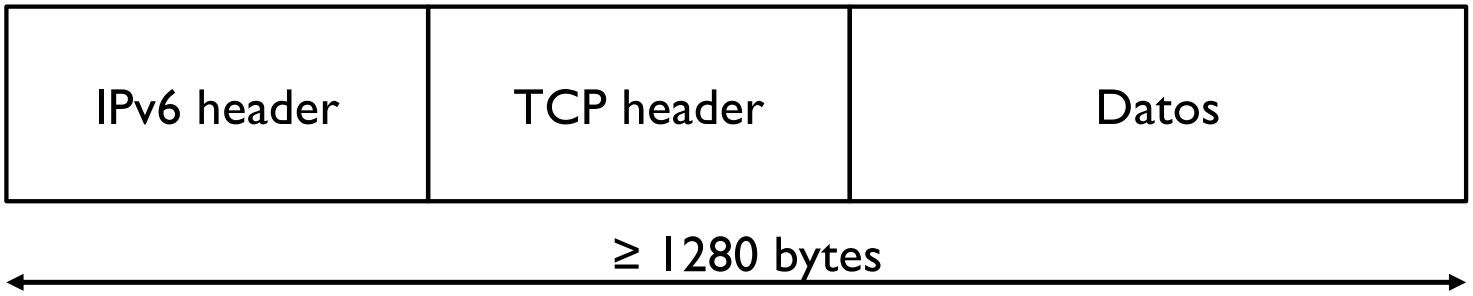
- Cada opción tendrá su propio formato, con sus campos, pero siempre hay un campo Siguiente Cabecera que indica que cabecera hay a continuación

2.5 - Cabecera IPv6
Siguiente Cabecera IPv6

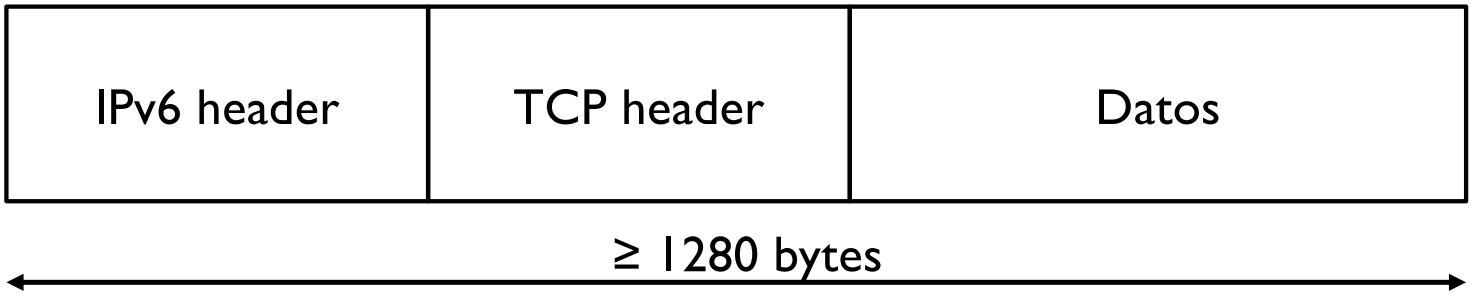
- Opciones estandarizadas y más usadas
  - Opciones hop-by-hop
  - Encaminamiento
  - Fragmentación
  - Autentificación
  - Encapsulamiento seguro
  - Opciones del destino

2.5 - Cabecera IPv6

Longitud mínima de un IPv6



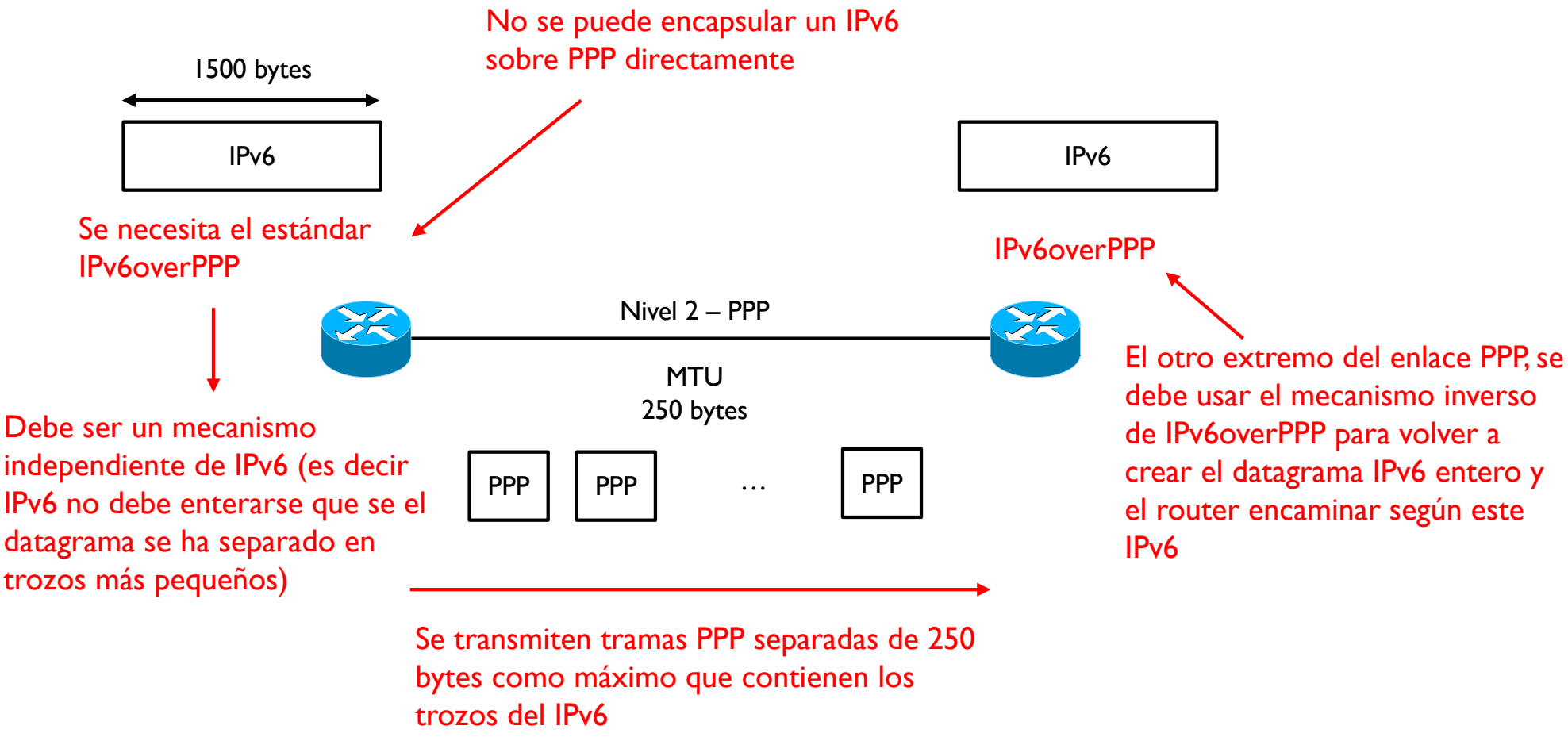
2.5 - Cabecera IPv6
Longitud mínima de un IPv6



- ¿Y su MTU de nivel enlace de una tecnología XYZ es menor de 1280 bytes?
- Hay que crear un estándar IPv6overXYZ que separe y junte los IPv6 a nivel enlace (nivel 2, menor que IPv6) para la transmisión con esta tecnología
- De manera que el nivel 3 (IPv6) no se entera y este nivel siempre trate datagramas superiores a 1280 bytes

2.5 - Cabecera IPv6

Longitud mínima de un IPv6



2.6.1 - Notación IPv6

- Se usan números hexadecimales separados por dos puntos

  2031 : 0000 : 130f : 0000 : 0000 : 09c0 : 876a : 130b

- Se simplifica quitando los 0 no significativos

  2031 : 0 : 130f : 0 : 0 : 9c0 : 876a : 130b

- Se simplifica quitando en un único lugar bloques de 0 seguidos y sustituyéndolo por ::

  2031 : 0 : 130f :: 9c0 : 876a : 130b

2.6.1 - Notación IPv6

- Solo se puede hacer esta sustitución en un único lugar ya que de lo contrario, la notación sería ambigua

- Esta @IPv6

  2031 :: 130f :: 09c0 : 876a : 130b

- Podría ser cualquiera de estas dos

  2031 : 0000 : 130f : 0000 : 0000 : 09c0 : 876a : 130b

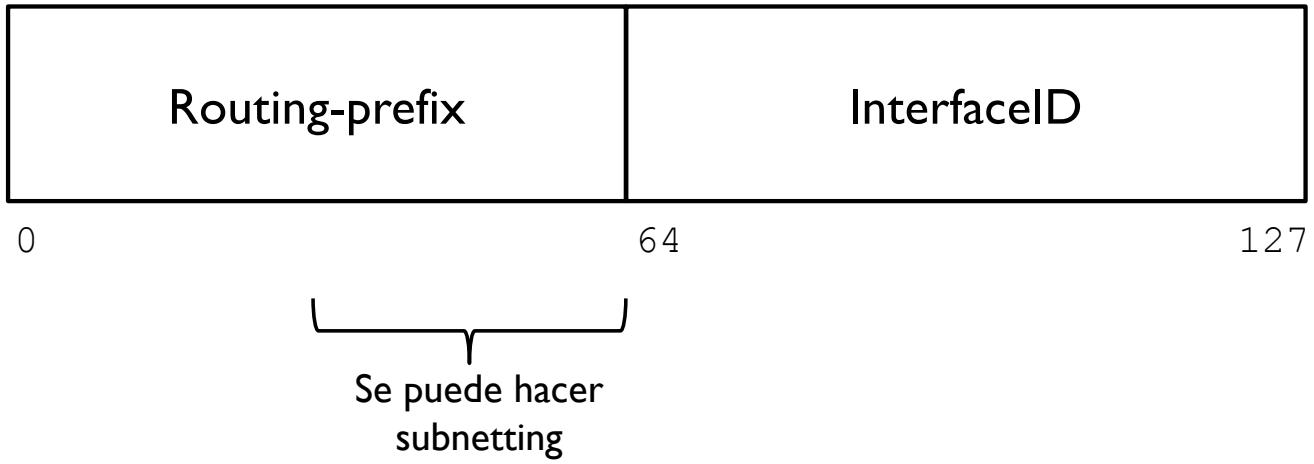
  2031 : 0000 : 0000 : 130f : 0000 : 09c0 : 876a : 130b

- Aunque simplificado, una @IPv6 es difícil de recordar

→ El DNS se hace aún mas fundamental

2.6.2 - Formato IPv6

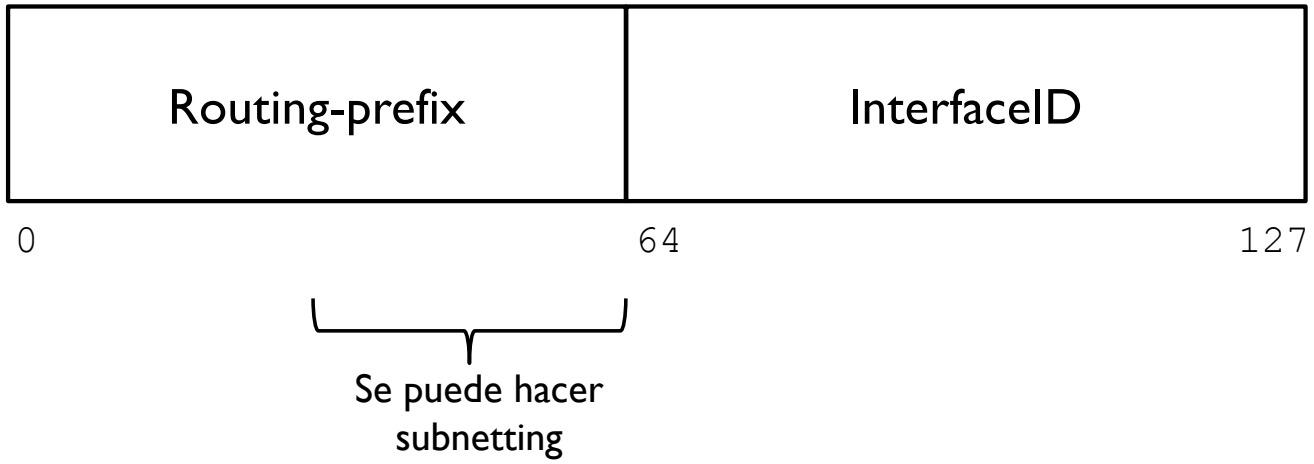
- No existen las direcciones de red y broadcast como en IPv4



- `2002:4c0::74:1:27`
  - `Routing prefix: ?`
  - `InterfaceID: ?`

2.6.2 - Formato IPv6

- No existen las direcciones de red y broadcast como en IPv4

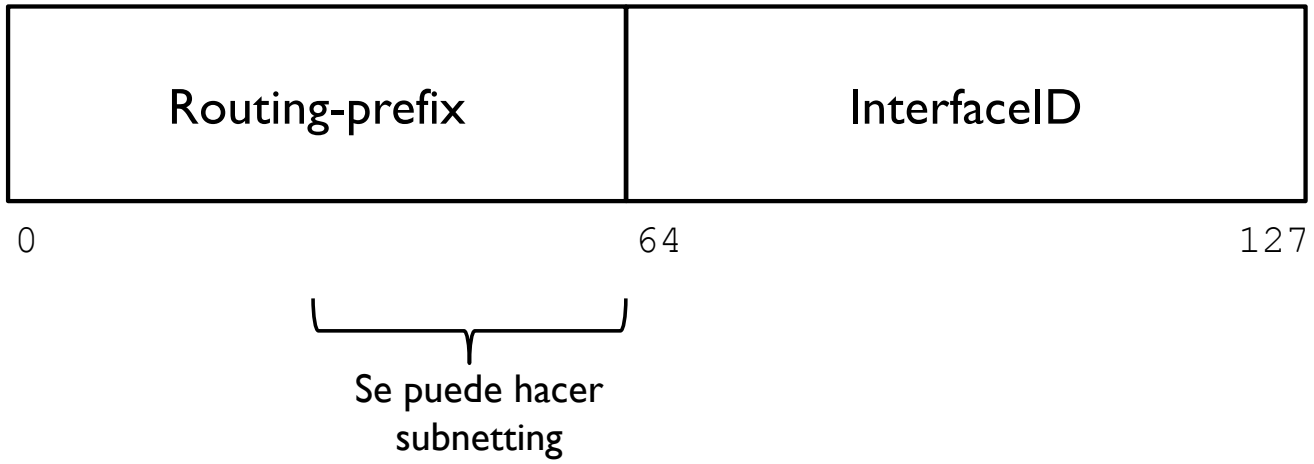


16 bits 16 ? 16 16 16

- `2002:4c0::74:1:27`
  - `Routing prefix: ?`
  - `InterfaceID: ?`

2.6.2 - Formato IPv6

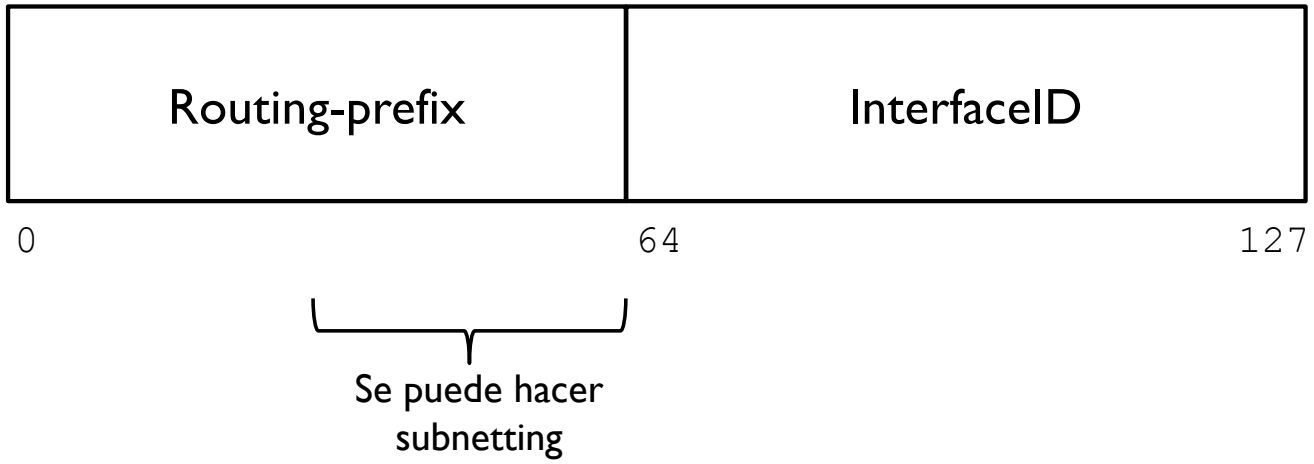
- No existen las direcciones de red y broadcast como en IPv4



- `2002:4c0::74:1:27`
  - `Routing prefix: 2002:04c0:0000:0000`
  - `InterfaceID: 0000:0074:0001:0027`

2.6.2 - Formato IPv6

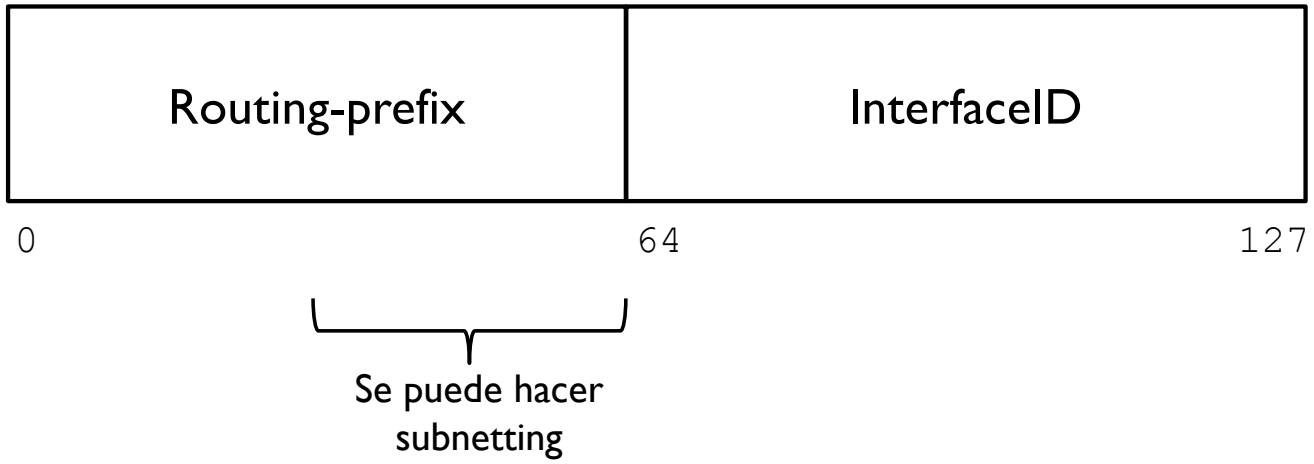
- No existen las direcciones de red y broadcast como en IPv4



- `2002:4c0::74:1:27`
- → Supongamos que el routing-prefix original recibido del ISP es de 52 bits

2.6.2 - Formato IPv6

- No existen las direcciones de red y broadcast como en IPv4

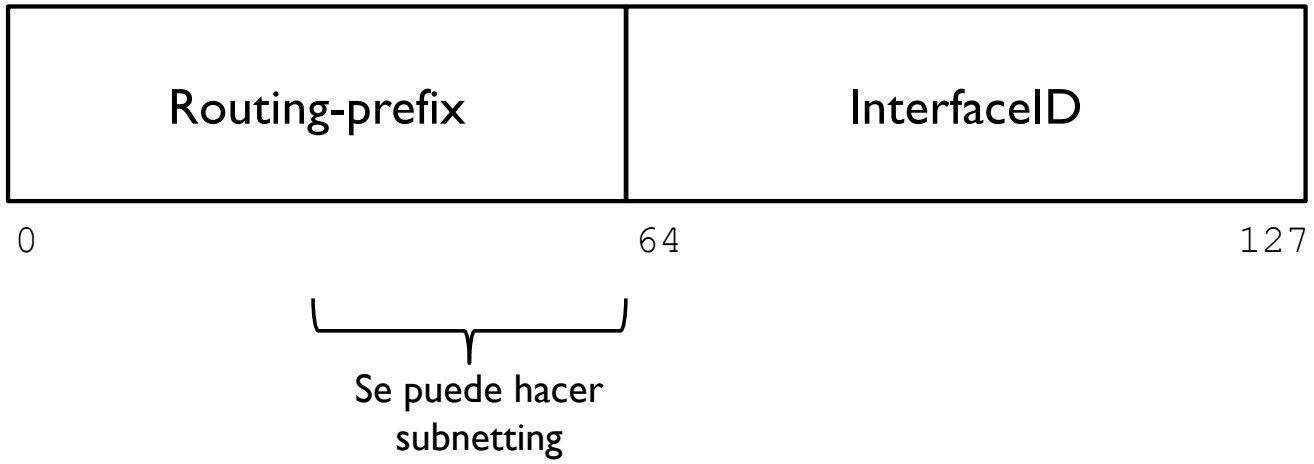


- `2002:4c0::74:1:27`
- → Supongamos que el routing-prefix original recibido del ISP es de 52 bits

  52 bits de routing-prefix → 64 – 52 = 12 bits de subnetting

2.6.2 - Formato IPv6

- No existen las direcciones de red y broadcast como en IPv4



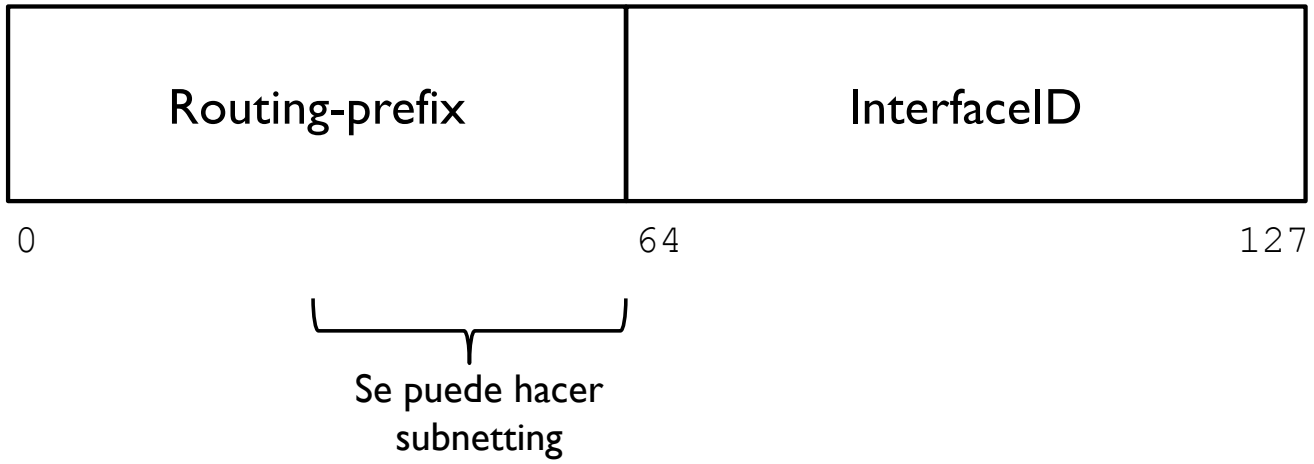
16 bits 16 ? 16 16 16

- `2002:4c0::74:1:27/52`

52 bits de routing-prefix → 64 – 52 = 12 bits de subnetting

2.6.2 - Formato IPv6

- No existen las direcciones de red y broadcast como en IPv4



16 bits 16 3x16 16 16 16

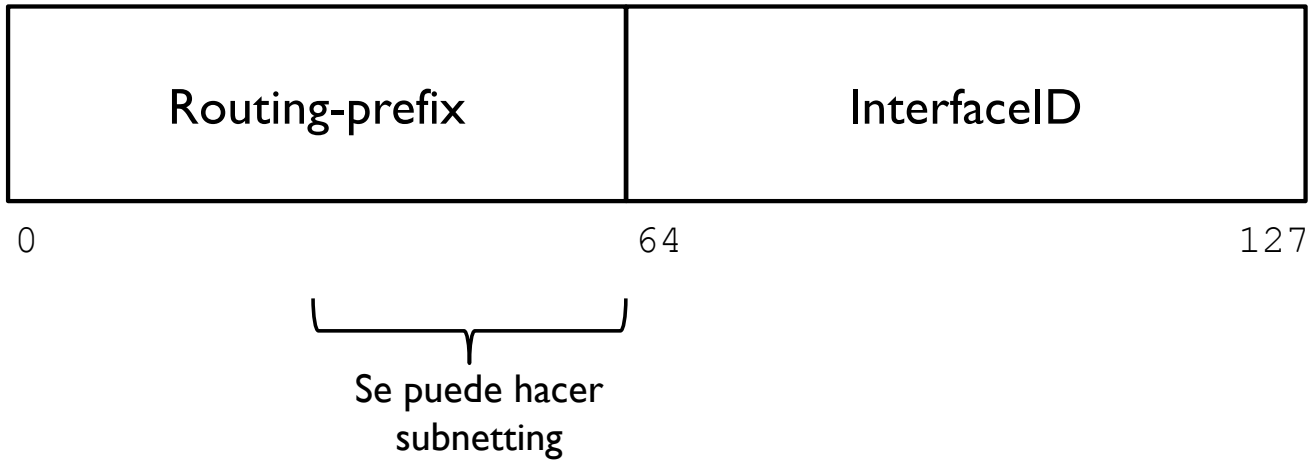
- `2002:4c0::74:1:27/52`

  52 bits de routing-prefix → 64 – 52 = 12 bits de subnetting

  64 bits de interfaceID → 4 grupos de 16 bits

2.6.2 - Formato IPv6

- No existen las direcciones de red y broadcast como en IPv4

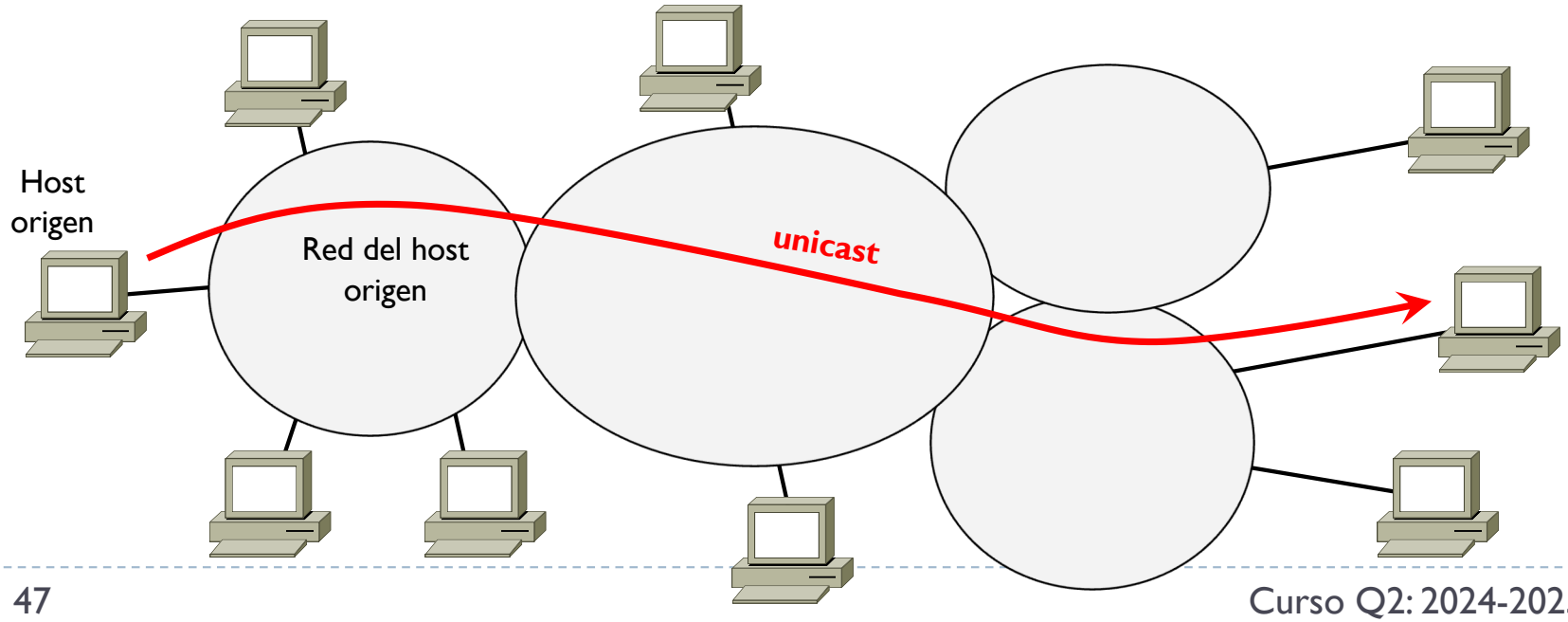


16 bits 16 3x16 16 16 16

- 2002:4c0::74:1:27/52
  - Routing prefix: 2002:04c0:0000:0
  - Subnet Routing prefix: 000
  - InterfaceID: :0000:0074:0001:0027

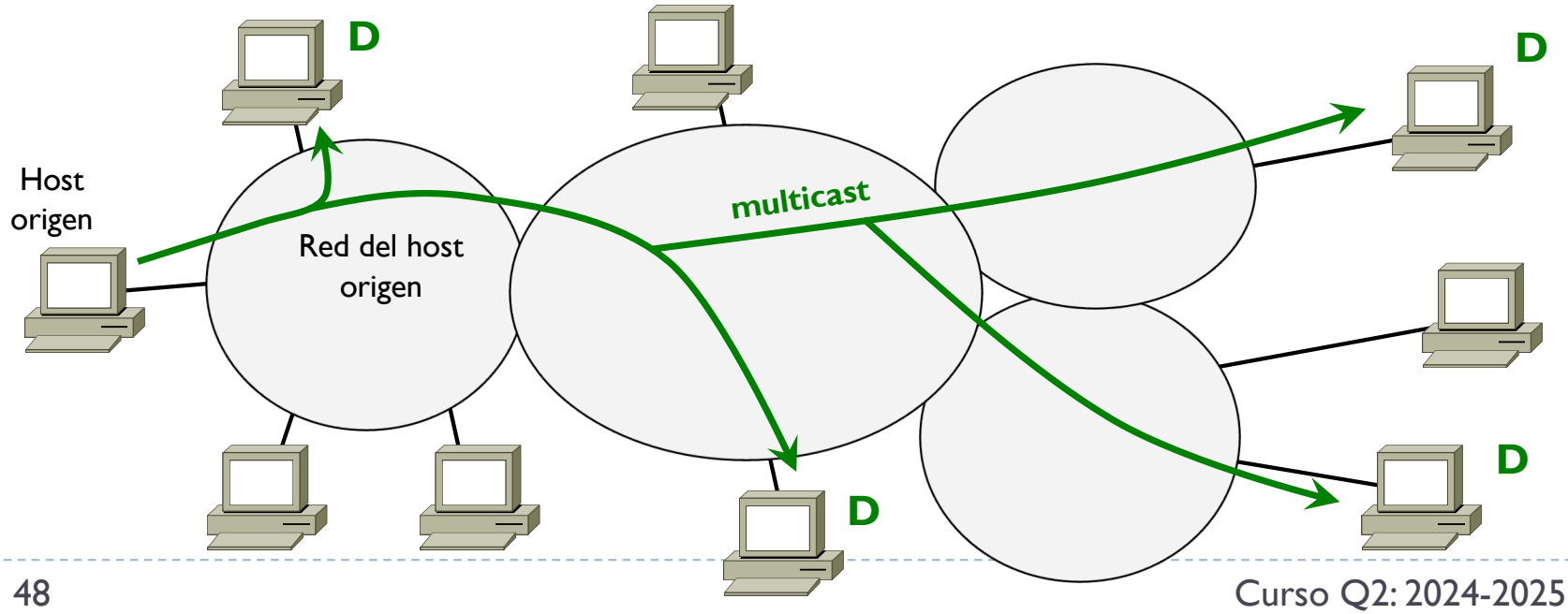
2.7 - Tipo de datagramas
Según el destino

- **Unicast**, un datagrama con un único destino
- **Multicast**, un datagrama que se replica en la red y alcanza un grupo bien definido de destinos
- **Anycast**, un datagrama que hay que entregar a un cualquier único destino de un grupo bien definido
  - Generalmente el que está más cerca



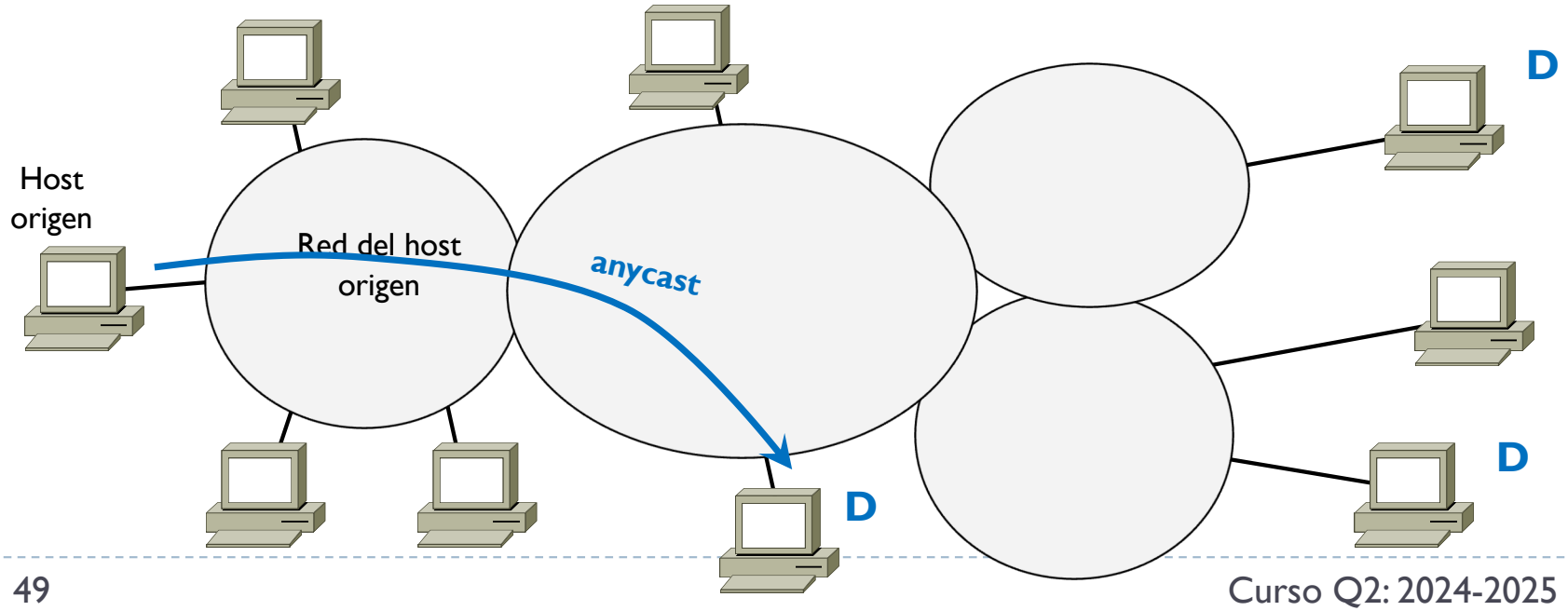
2.7 - Tipo de datagramas
Según el destino

- **Unicast**, un datagrama con un único destino
- **Multicast**, un datagrama que se replica en la red y alcanza un grupo bien definido de destinos
- **Anycast**, un datagrama que hay que entregar a un cualquier único destino de un grupo bien definido
  - Generalmente el que está más cerca



2.7 - Tipo de datagramas
Según el destino

- **Unicast**, un datagrama con un único destino
- **Multicast**, un datagrama que se replica en la red y alcanza un grupo bien definido de destinos
- **Anycast**, un datagrama que hay que entregar a un cualquier único destino de un grupo bien definido
  - Generalmente el que está más cerca



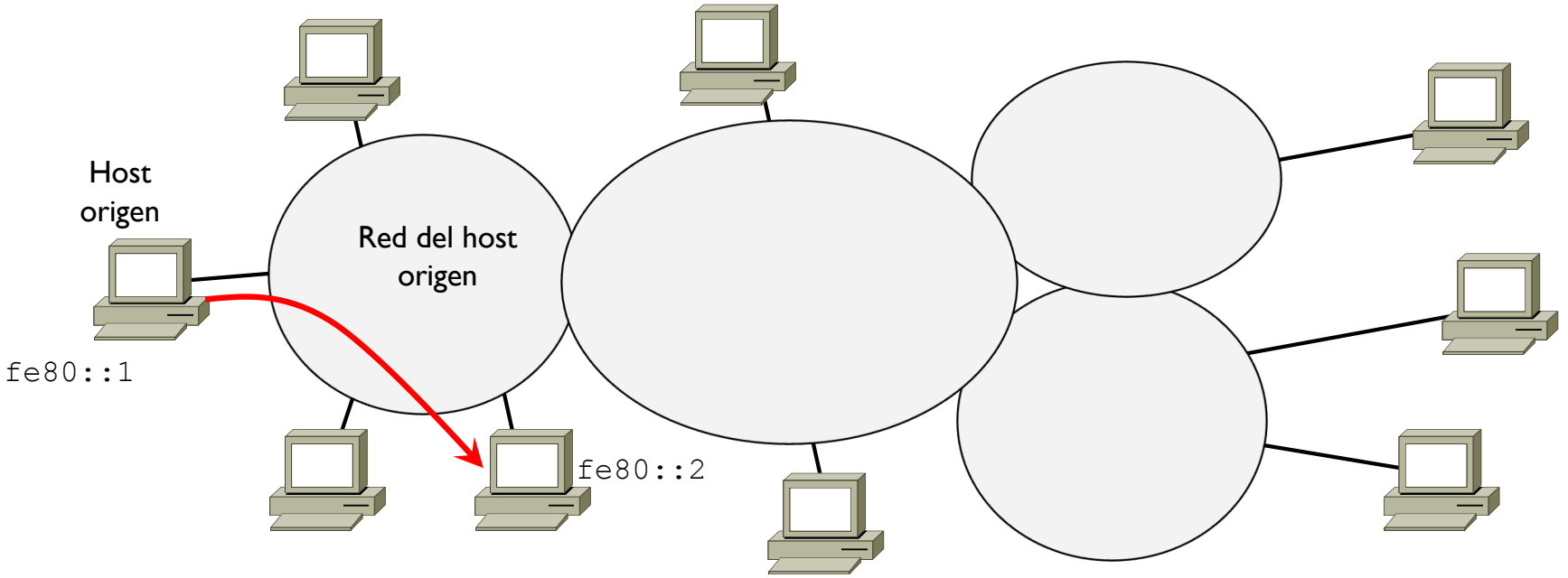
2.8 - Direccionamiento

- Inicialmente se crearon 3 tipos que se pueden asignar a cada interfaz

- **Link-local**
- **Site-local**
- **Global**

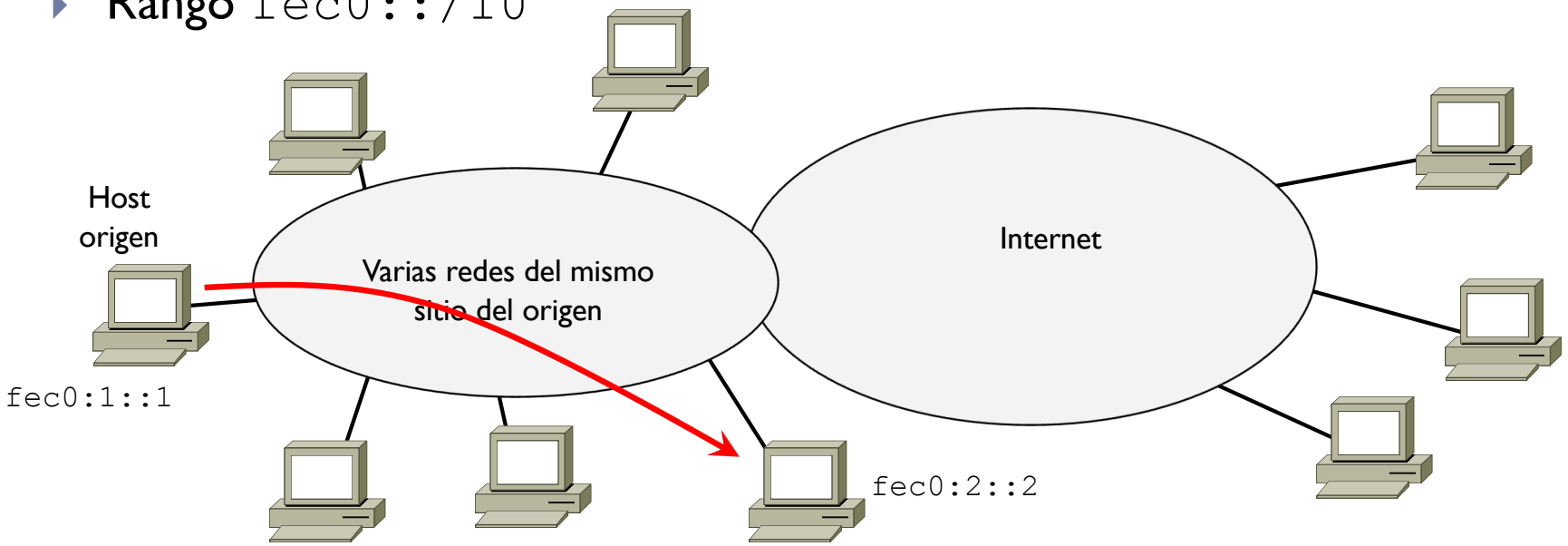
2.8 - Direccionamiento
Link-local

- Para transmisión a destinos de la misma red del origen
- Origen y destino usan un routing-prefix del rango
- `fe80::/10`
- completado con otros 54 bits y luego el InterfaceID de 64 bits



2.8 - Direccionamiento
Site-local

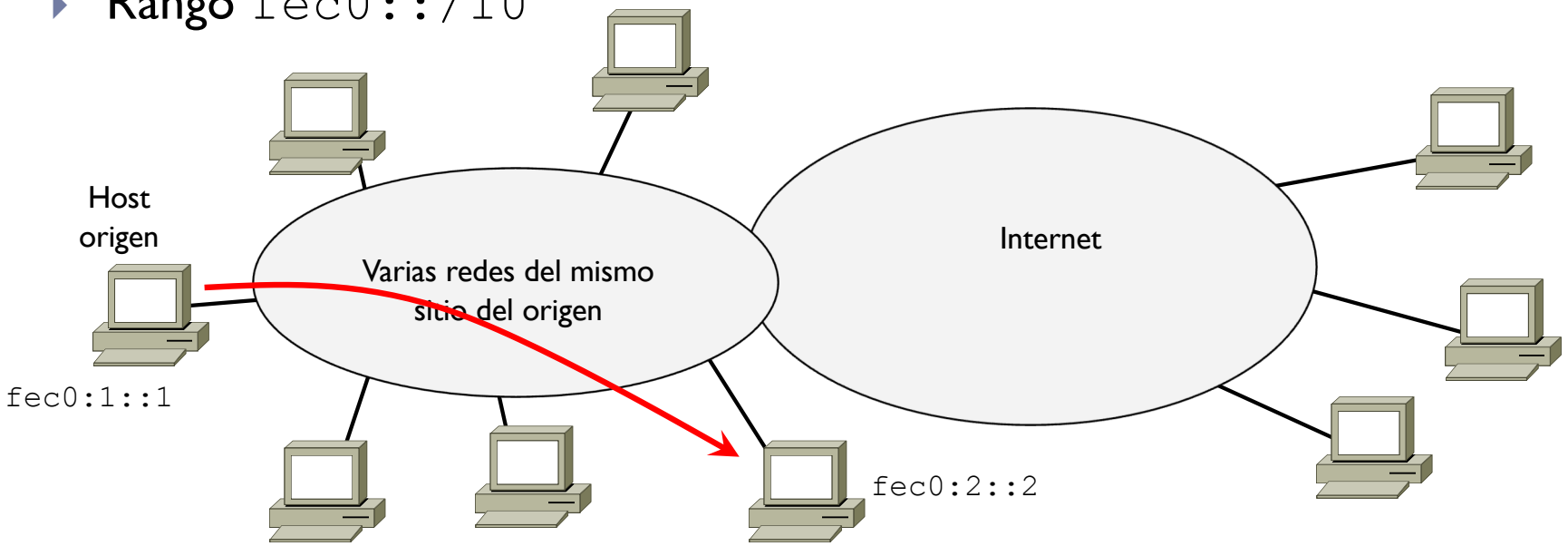
- Para transmisión entre un origen y un destino dentro de un mismo "sitio"
- Rango `fec0::/10`



- Son direcciones privadas no enrutables en Internet

2.8 - Direccionamiento
Site-local

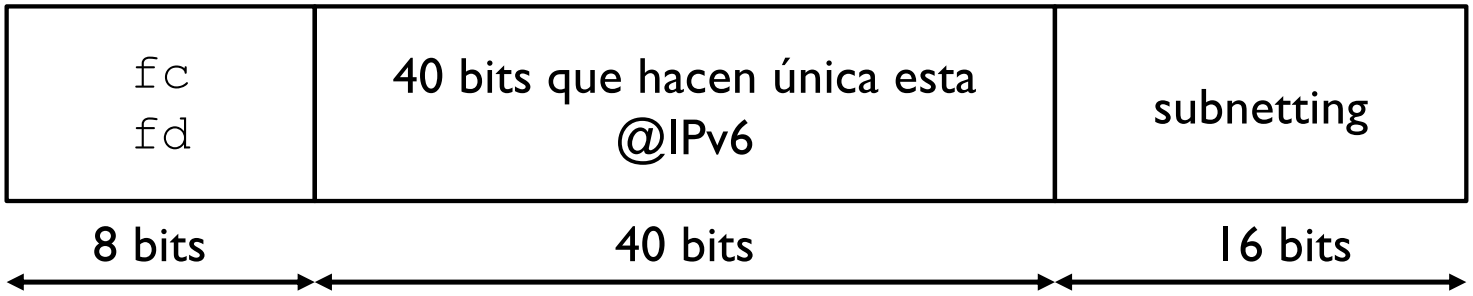
- Para transmisión entre un origen y un destino dentro de un mismo "sitio"
- Rango `fec0::/10`



- En septiembre 2004 (RFC 3879), esta @IPv6 ya no se soporta porque ambigua
  - ¿Qué es un site y cual es su limite?
- Y porque no facilita una administración de red simple con poca intervención en caso de cambios en la infraestructura
  - por ejemplo, al juntar dos sistemas, hay que reconfigurar todo el site otra vez

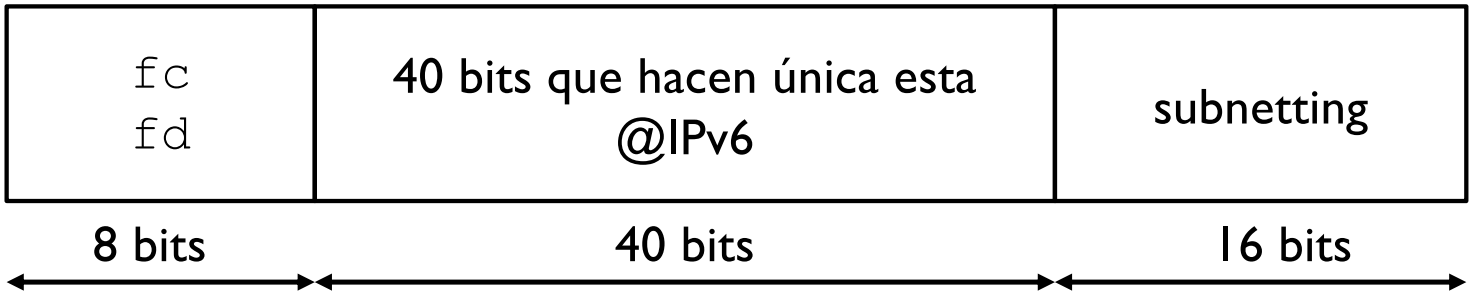
2.8 - Direccionamiento
Unique Local Address (ULA)

- Unique Local Address (ULA)
  - En octubre 2005 (RFC 4193), se crea la ULA como sustituida de site-local
  - Pensada como dirección privada (equivalente a las de IPv4) pero que además sean únicas
  - Facilita la administración de red ya que eventuales cambios no afectan todo el sistema porque se garantiza que no haya redes duplicadas
- Rango `fc00::/7`
- Se crearon 2 grupos con 2 soluciones diferentes
  - Rango `fc00::/8`
  - Rango `fd00::/8`



2.8 - Direccionamiento
Unique Local Address (ULA)

- Rango `fc00::/8`
  - Los 40 bits únicos los proporciona una entidad centralizada que controla esta unicidad
  - Esta entidad actualmente aún no está disponible
- Rango `fd00::/8`
  - Los 40 bits únicos se generan con un algoritmo definido en el RFC 4193
  - Estos 40 bits son los mismos para todo el sistema, luego se hace subnetting para cada red sobre los 16 bits que quedan del routing-prefix
  - Método usado actualmente



2.8 – Direccionamiento

Otras direcciones particulares

- Otras direcciones reservadas (principales)
- `::/128` → Dirección no especificada, valor inicial de las tarjetas
- `::1/128` → Loopback
- `ff00::/8` → Multicast

- Y muchísimas más
  - ::ffff:0:0/96 reserved for IPv4-mapped Address
  - 64:ff9b::/96 is used in an algorithmic mapping between IPv4 to IPv6 addresses
  - 2001:0000::/32 reserved for TEREDO
  - 2001:0002::/48 reserved for Benchmarking
  - 2001:5::/32 reserved for EID Space for LISP
  - 2001:db8::/32 reserved for Documentation
  - 2002::/16 reserved for 6to4
  - etc.

https://www.iana.org/assignments/ipv6-address-space/ipv6-address-space.xhtml

2.8 – Direccionamiento
Multicast IPv6

- Bloque asignado a Multicast: `ff00::/8`
  - Es decir, los primeros 8 bits son todos 1 en binario

- Existen grupos multicast ya definidos
  - Al enviar un datagrama con destino estas @IPv6, el datagrama alcanza todos estos destinos

  - `ff02::1` → todos los nodos de una LAN
  - `ff02::2` → todos los routers de una LAN
  - `ff02::9` → todos los routers RIP de una LAN
  - `ff02::1:2` → todos los servidores DHCP de una LAN
  - Y muchos más

https://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xhtml

2.8 – Direccionamiento

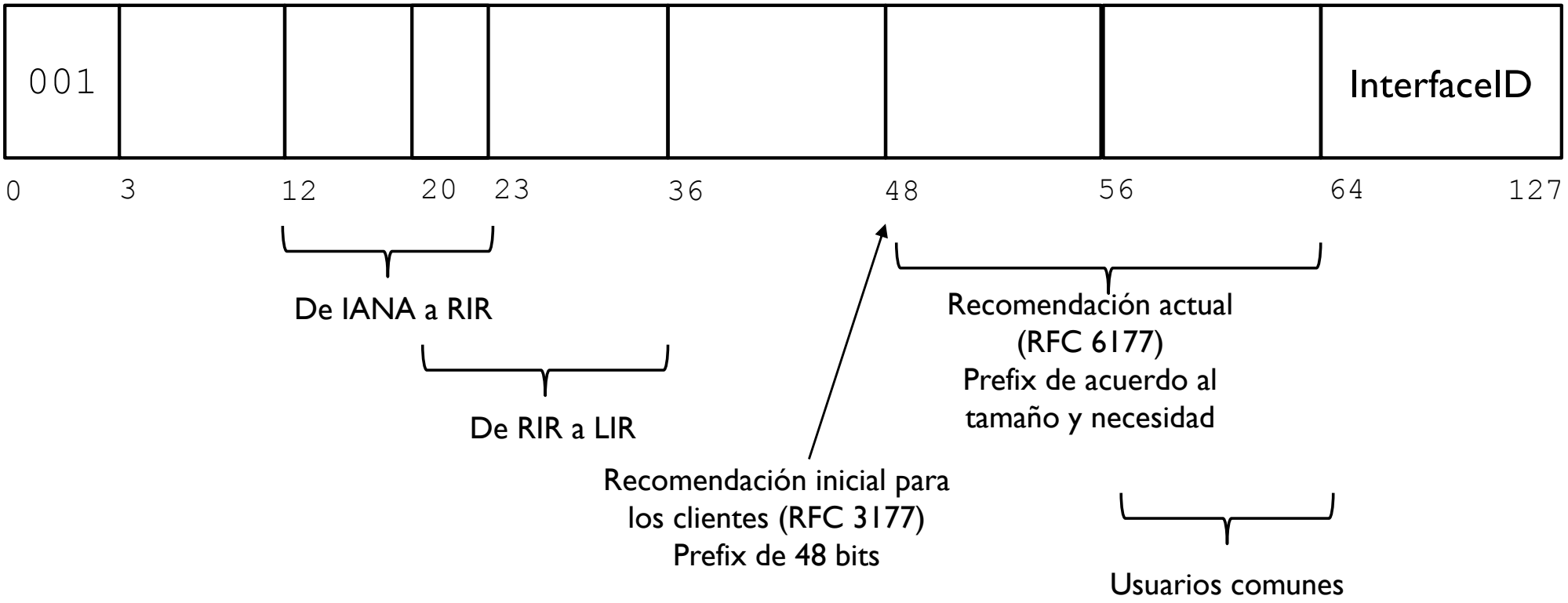
Anycast IPv6

- Se usan @IPv6 globales
- Las direcciones anycast son sintácticamente indistinguibles de las direcciones unicast.
- Una dirección anycast se asigna multiples interfaces que pertenecen a diferentes hosts
- Cuando una dirección global se asigna a más de una interfaz, se convierte en una dirección anycast y los nodos a los que se asigna la dirección deben configurarse explícitamente para saber que es una dirección anycast

2.8 – Direccionamiento
IPv6 global unicast

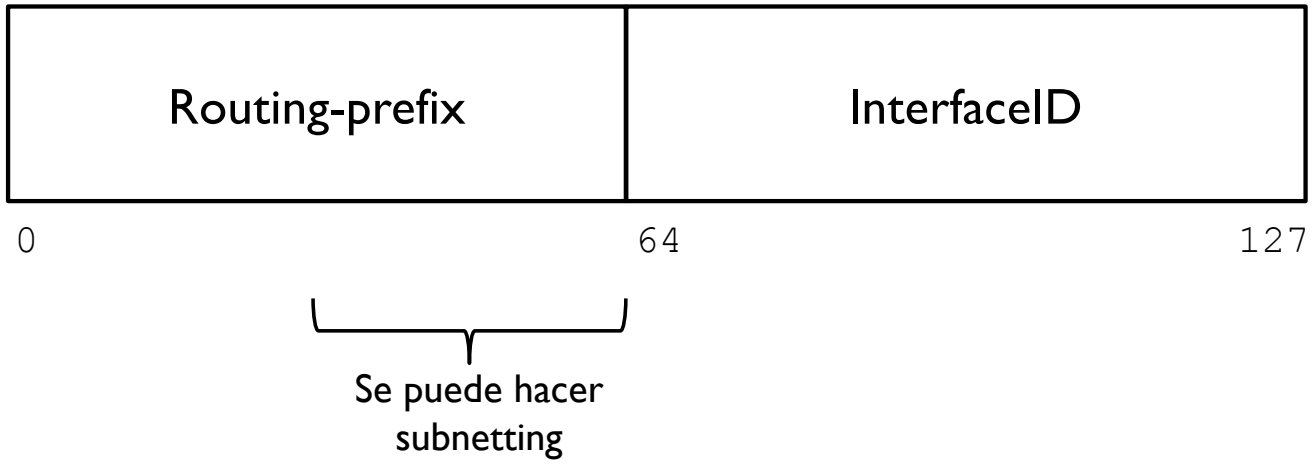
- Todo lo que queda disponible
- Actualmente IANA solo está asignado @IPv6 de este rango

`2000::/3`



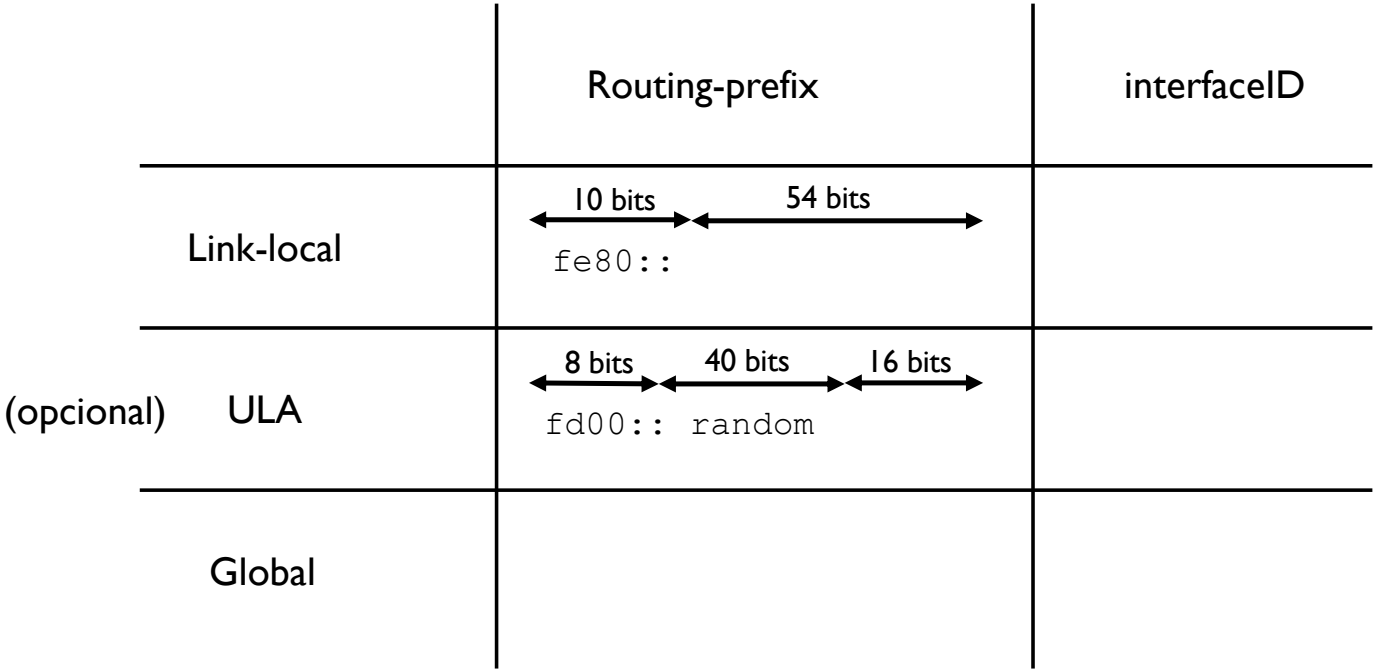
2.8 – Direccionamiento
Como se asignan @IPv6

- No existen las direcciones de red y broadcast como en IPv4



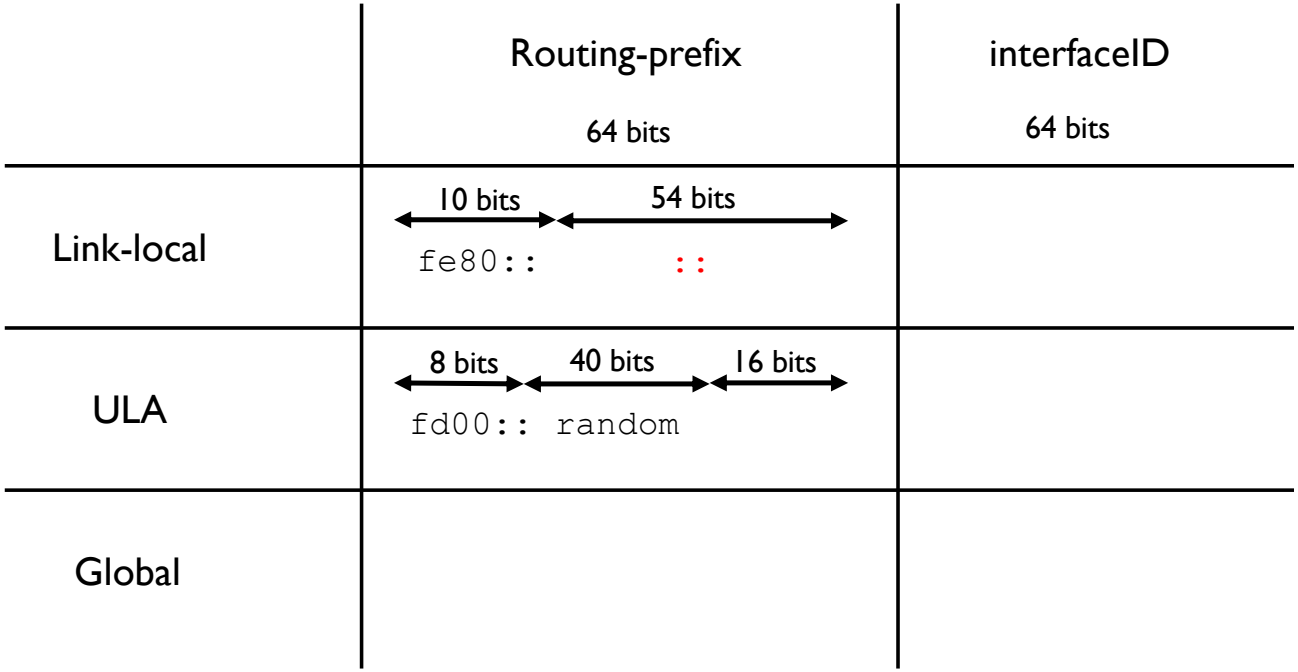
2.8 – Direccionamiento

¿Qué es lo que hay que asignar?



2.8 – Direccionamiento

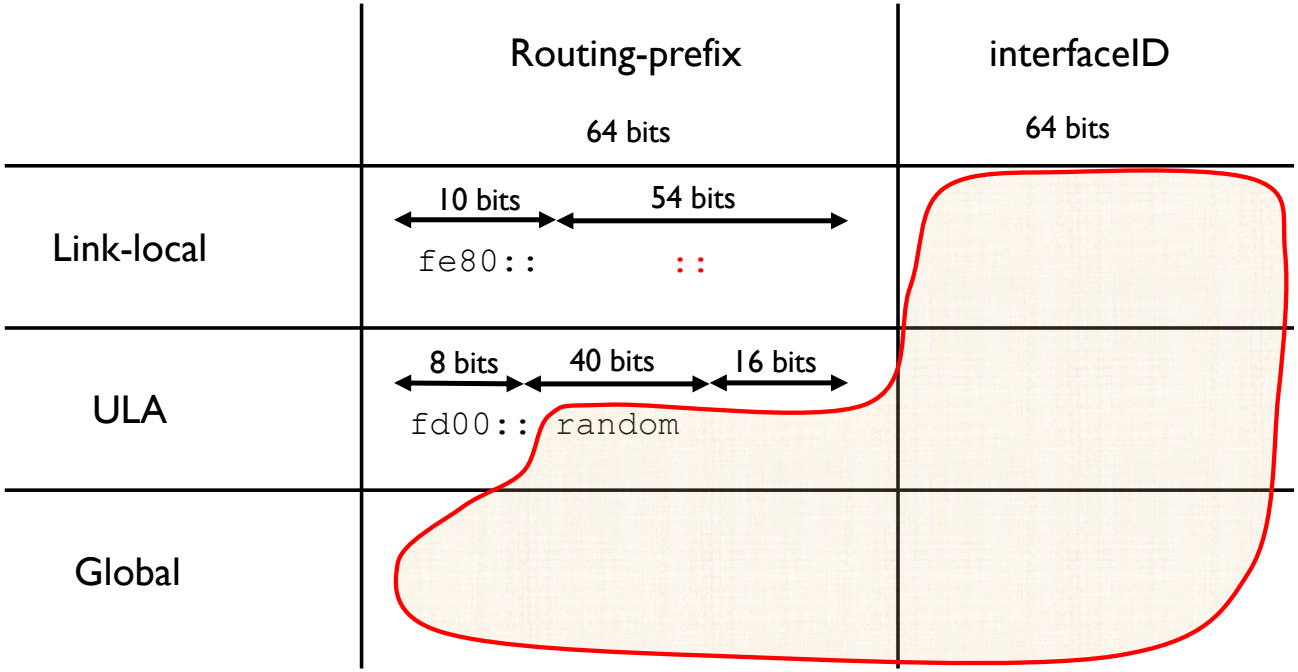
¿Qué es lo que hay que asignar?



Como el alcance está limitado a la misma red y si no hay alguna razón especifica para que sea diferente, estos 54 bits suelen ser todos 0

2.8 – Direccionamiento

¿Qué es lo que hay que asignar?



¿Y el resto como se asigna?

2.8 – Direccionamiento
Como se asignan @IPv6

- Stateful
  - Conocimiento completo de los "estados"
  - Una entidad mantiene todas las IP de los hosts y evita que se dupliquen
  - Por ejemplo, servidores DHCP o manualmente
  - Como en IPv4

- Stateless
  - Stateless Address Auto-Configuration (SLAAC)
  - Nuevo método
  - Sin conocer todos los "estados"
  - Cada host se autoconfigura correctamente sin duplicar @IP

2.8 – Direccionamiento
Configuración stateful

- Manual
  - Se asigna de forma manual y se usa DAD para reconocer @IPv6 duplicadas

- DHCPv6
  - Puerto origen 546 y puerto destino 547
  - El host sin @IP envía un DHCPDISCOVER con @IP origen su link-local y @IP destino la dirección multicast ff02::1:2 (todos los servidores DHCPv6)
  - El o los servidores contestan con los parámetros de configuración: routing-prefix, ULA, mascaras, Gateway, hostname, domain, etc.

2.8 – Direccionamiento
Configuración SLAAC

- Hay que introducir primero el ICMPv6

2.8 - ICMPv6

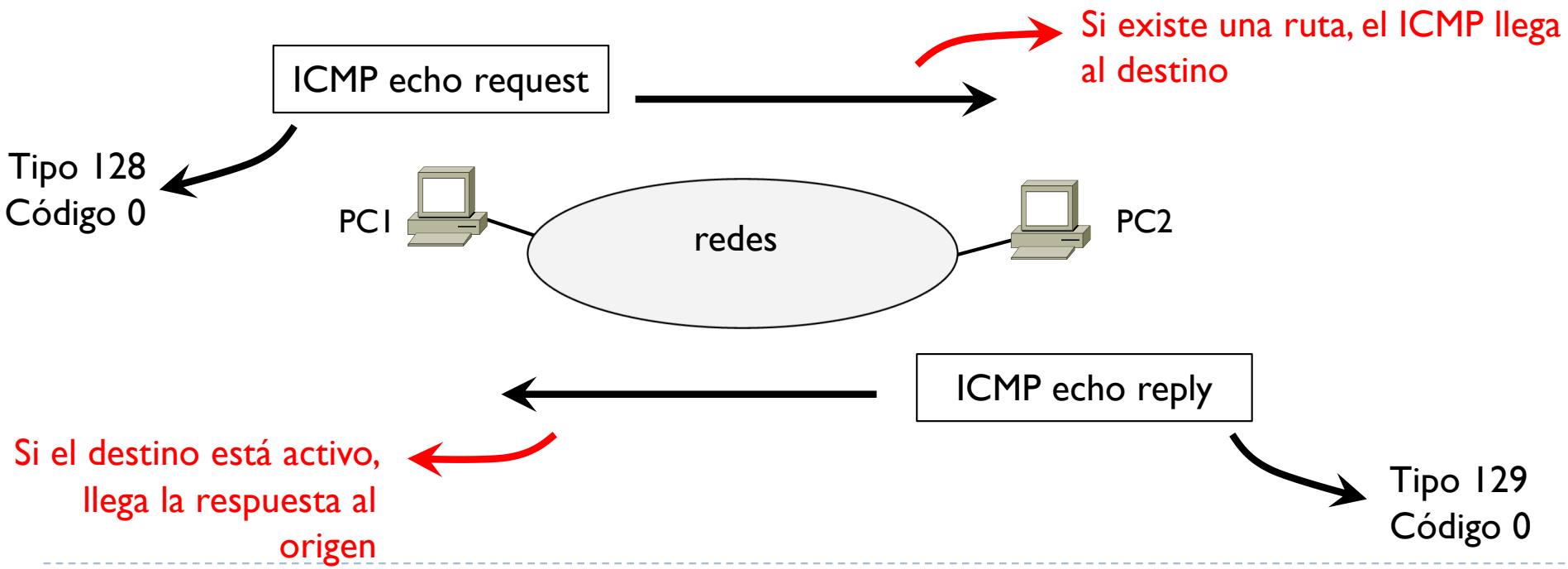
- ICMPv4 + ARP + nuevas funciones

2.8 - ICMPv6

Como ICMPv4

- ICMPv4: protocolo de control de IPv4
  - Envío de mensajes de supervisión (echo request/reply) y de error

- ICMPv6: supervisión

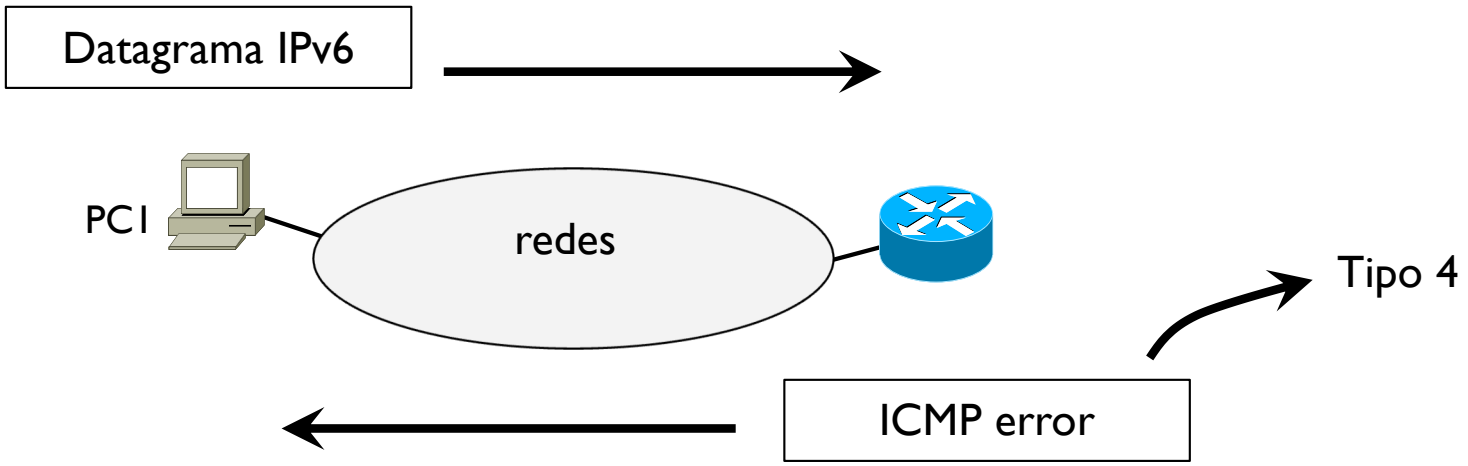


2.8 - ICMPv6

Como ICMPv4 + nuevos errores propio de IPv6

- ICMPv4: protocolo de control de IPv4
  - Envío de mensajes de supervisión (echo request/reply) y de error

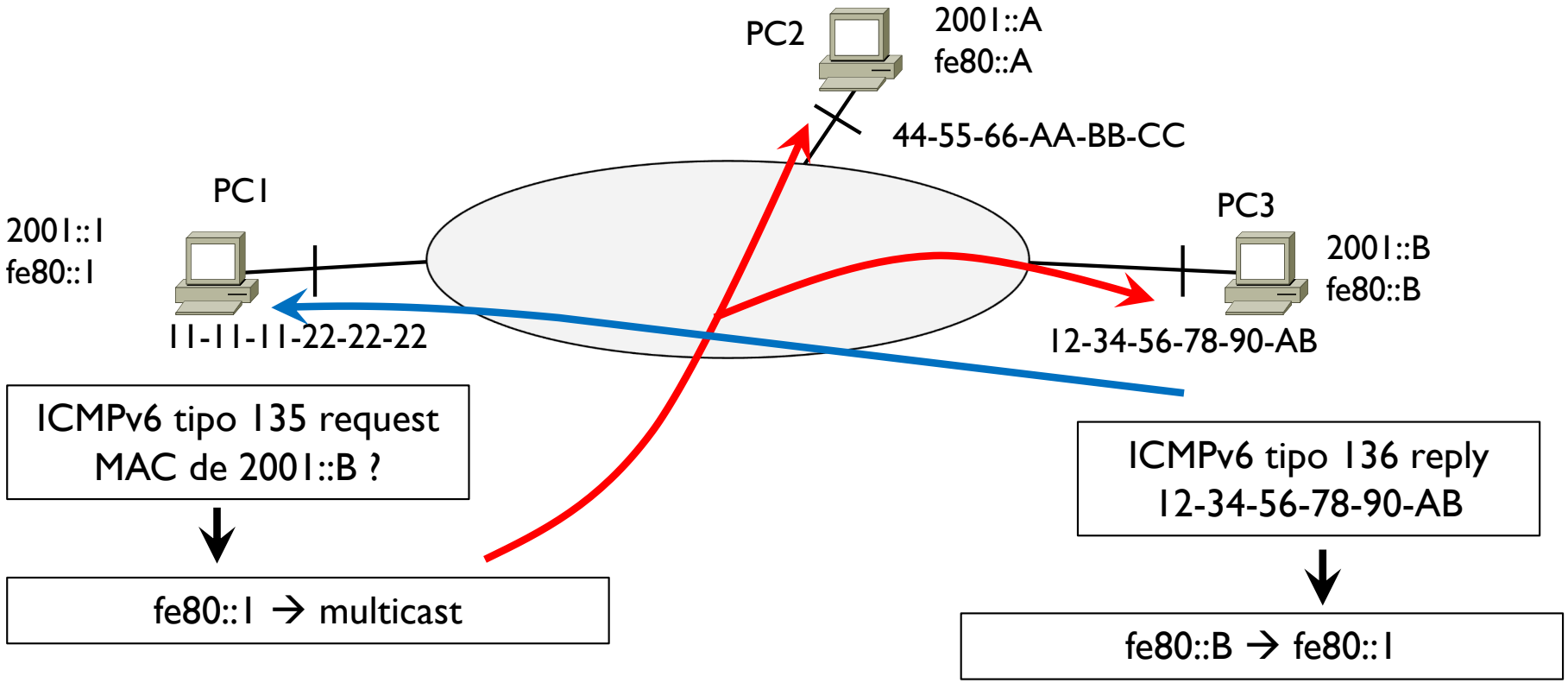
- ICMPv6: nuevos errores
  - Tipo 4: Problema con los parámetros
    - Código 1: campo Siguiente Cabecera no reconocido
    - Código 2: opción IPv6 no reconocida



2.8 - ICMPv6

Incluye el protocolo ARP de IPv4

- ARP: protocolo de resolución de dirección MAC
  - Conocida una @IP, se quiere descubrir su dirección MAC

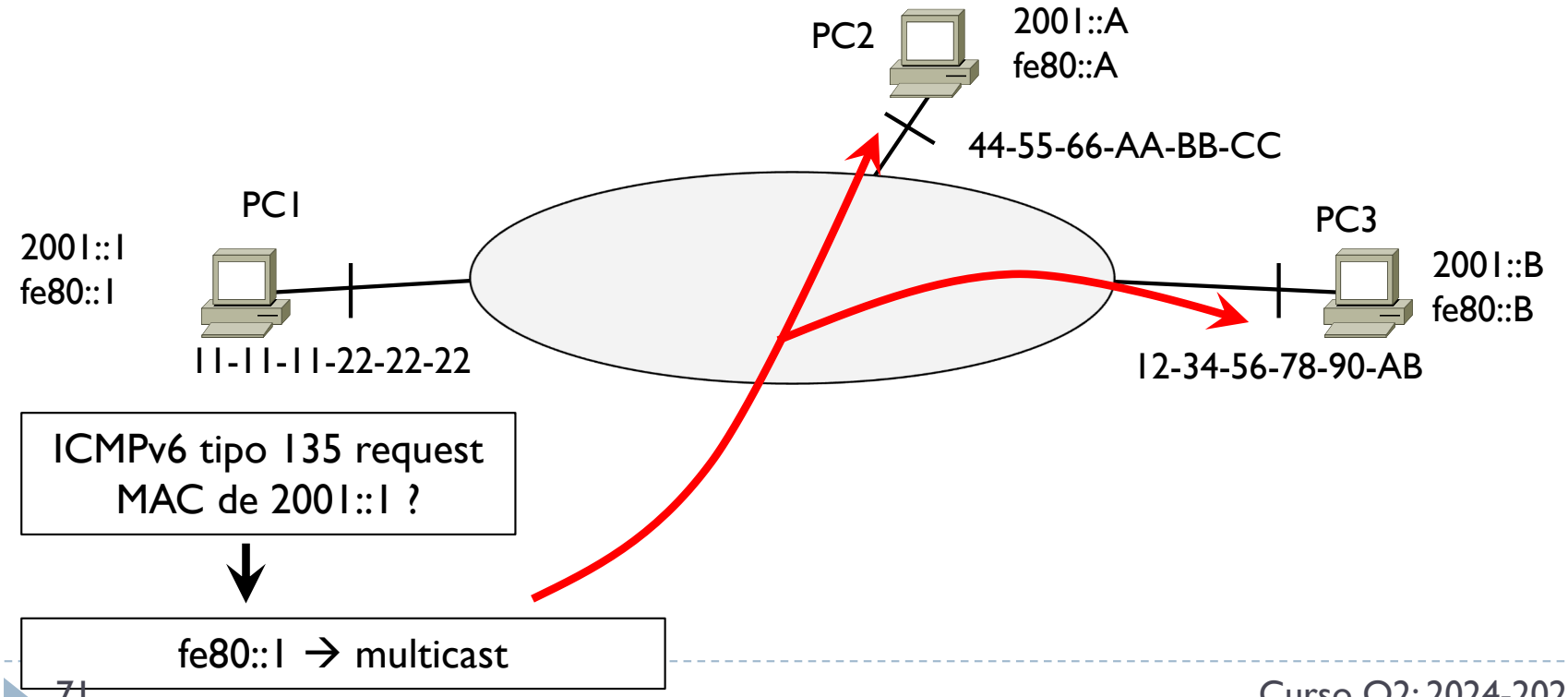


2.8 - ICMPv6

Incluye el protocolo ARP de IPv4

- Incluidos los ARP gratuitos ahora llamado DAD

- En un ARP gratuito
  - Se envía un ARP request indicando que se busca la MAC de la @IP del que envía el mensaje



2.8 - ICMPv6

Incluye el protocolo ARP de IPv4

- Incluidos los ARP gratuitos ahora llamado DAD

- En un ARP gratuito
  - Se envía un ARP request indicando que se busca la MAC de la @IP del que envía el mensaje
  - Si alguien contesta, es que este tiene la misma @IP del que ha enviado el request → @IP duplicada

- DAD: Duplicate Address Discovery

2.8 – ICMPv6

Nueva función: asignación de routing-prefix global + ULA

- Al arrancar un host tiene la @IPv6 ::/128
- Envía un ICMPv6 (nueva función) de tipo Router Solicitation (RS, tipo 133) en multicast en su red pidiendo la configuración global y ULA

- Un router envía periódicamente un ICMP de tipo Router Advertisement (RA, tipo 134) o como respuesta a un ICMP RS en multicast en la red informando de la configuración del routing-prefix Global y si se usa y cual es el routing-prefix ULA

2.8 – Direccionamiento
Como se asignan @IPv6

- Stateful
  - Conocimiento completo de los "estados"
  - Una entidad mantiene todas las IP de los hosts y evita que se dupliquen
  - Por ejemplo, servidores DHCP o manualmente
  - Como en IPv4

- Stateless
  - Stateless Address Auto-Configuration (SLAAC)
  - Nuevo método
  - Sin conocer todos los "estados"
  - Cada host se autoconfigura correctamente sin duplicar @IP

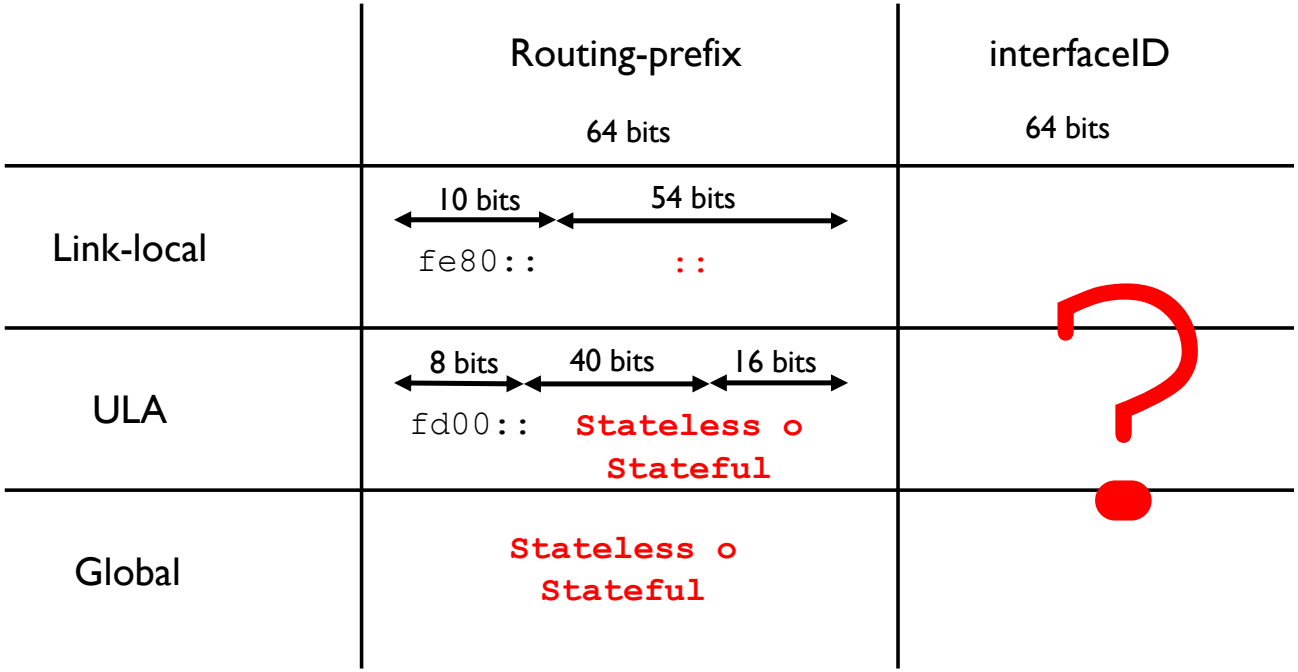
2.8 – Direccionamiento
Configuración SLAAC

- Proceso

1) Un host pide el routing-prefix y ULA (opcional) con ICMPv6 RS/RA
2) Genera la parte interfaceID y completa las @IPv6
3) Se verifica que la @IPv6 es única en la red (se envía un DAD)
4) Si es única se asigna a la interfaz
   Si no es única se vuelve al punto 2)

2.8 – Direccionamiento

¿Qué es lo que hay que asignar?



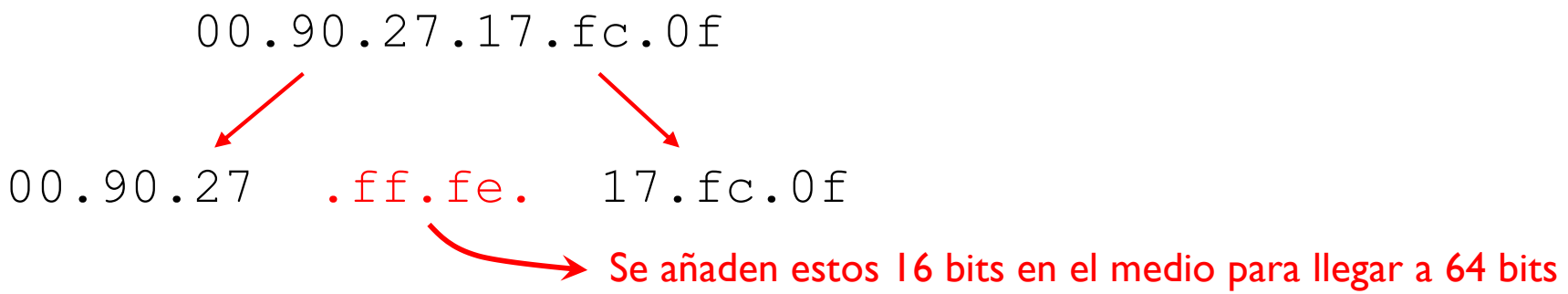
Como el alcance está limitado a la misma red y si no hay alguna razón especifica para que sea diferente, estos 54 bits suelen ser todos 0

2.9 – Direccionamiento
Dos métodos para InterfaceID

- Usando la MAC
- Aleatorio

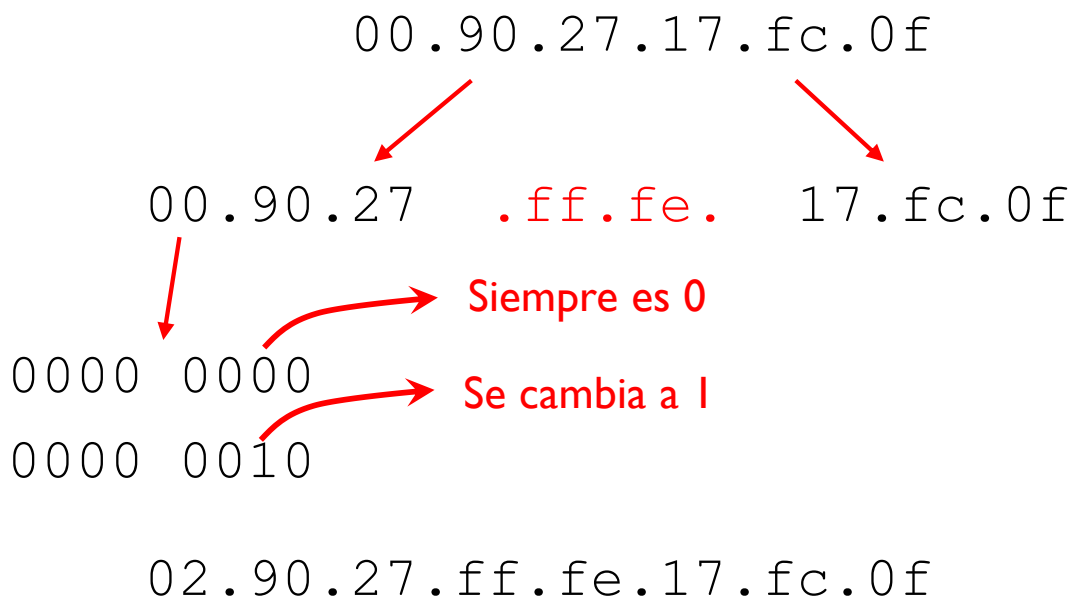
2.9 – Direccionamiento
Usando la MAC

- La dirección MAC en principio es un número de 48 bits único asignado a cada interfaz de hosts y routers
- Se transforma este número en otro de 64 bits
- Proceso



2.9 – Direccionamiento
Usando la MAC

- La dirección MAC en principio es un número de 48 bits único asignado a cada interfaz de hosts y routers
- Se transforma este número en otro de 64 bits
- Proceso



→ En notación IPv6

290:27ff:fe17:fc0f

2.9 – Direccionamiento
Usando la MAC

- Problemas

1) Los fabricantes han admitido vender tarjetas con la misma MAC

   De manera que con este proceso, dos tarjetas lleguen a tener la misma @IPv6

   Se ejecuta DAD para descubrir si está duplicada

2) ?

2.9 – Direccionamiento
Usando la MAC

- Problemas

1) Los fabricantes han admitido vender tarjetas con la misma MAC

   De manera que con este proceso, dos tarjetas lleguen a tener la misma @IPv6

   Se ejecuta DAD para descubrir si está duplicada

2) Privacidad?

2.9 – Direccionamiento
Usando la MAC

- Problemas

1) Los fabricantes han admitido vender tarjetas con la misma MAC

   De manera que con este proceso, dos tarjetas lleguen a tener la misma @IPv6

   Se ejecuta DAD para descubrir si está duplicada

2) Como la InterfaceID ahora sería siempre la misma independiente del lugar donde se conecta un host móvil (solo cambiaría el routing-prefix), ahora sería posible conocer su localización y sus movimientos

2.9 – Direccionamiento
InterfaceID aleatoria

- Se puede activar la opción de generar un interfaceID aleatorio
- Luego DAD descubre si es única

2.9 – Direccionamiento

Método usado generalmente

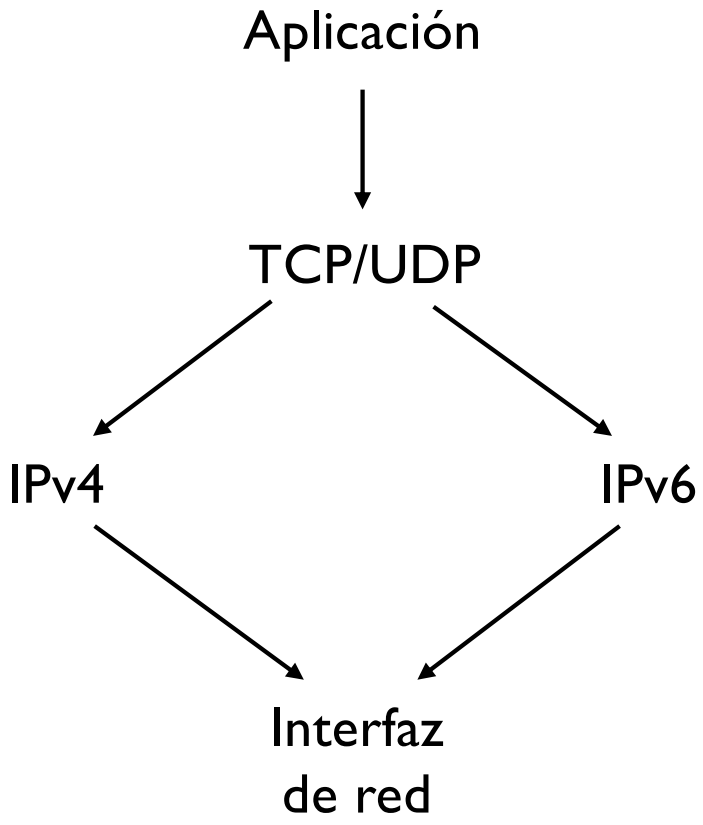
- MAC como InterfaceID para conexiones a redes internas (por razones de seguridad y permisos a sistemas y servicios internos)
- InterfaceID aleatoria para conexiones a redes externas

2.10 - Compatibilidad IPv4 – IPv6

- Diferentes métodos según el escenario
  - Dual stack
  - IPv4-mapped
  - Tunneling
  - NAT64
  - etc.

2.10 - Compatibilidad IPv4 – IPv6
Dual stack

- Generalmente un host funciona con dual stack y se adapta según la red y el destino



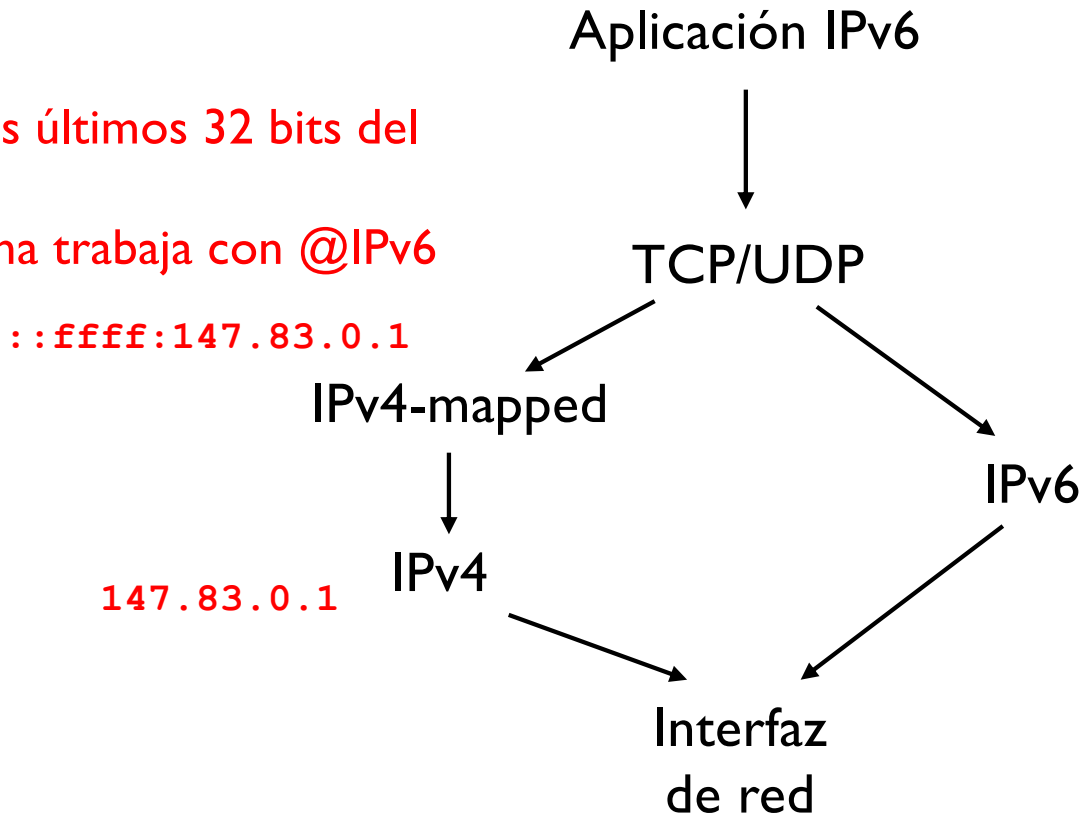
2.10 - Compatibilidad IPv4 – IPv6

IPv4-mapped

- Si una aplicación solo funciona con @IPv6, se usa IPv4-mapped
- Rango reservado `::ffff:0:0/96`

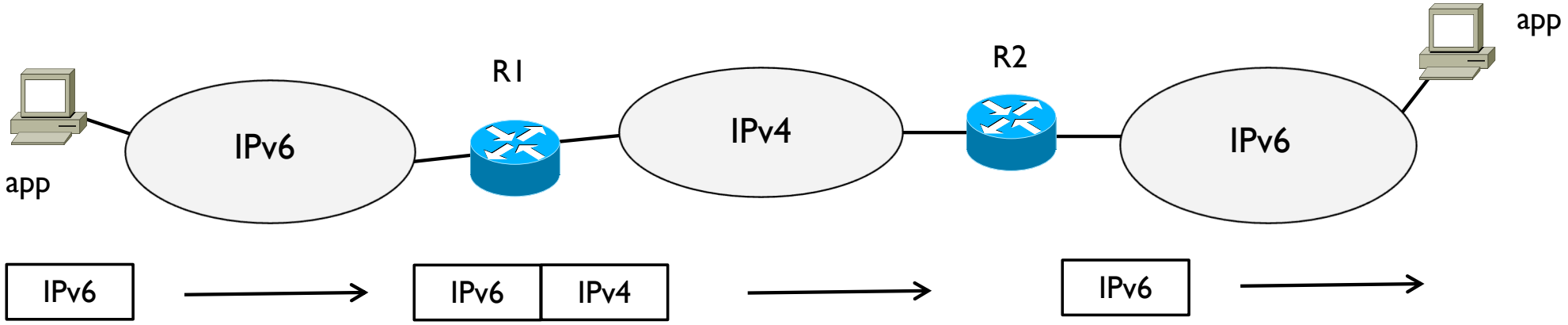
La @IPv4 se mapea en los últimos 32 bits del `::ffff:0:0:/96`

La aplicación de esta forma trabaja con @IPv6



2.10 - Compatibilidad IPv4 – IPv6
Tunneling

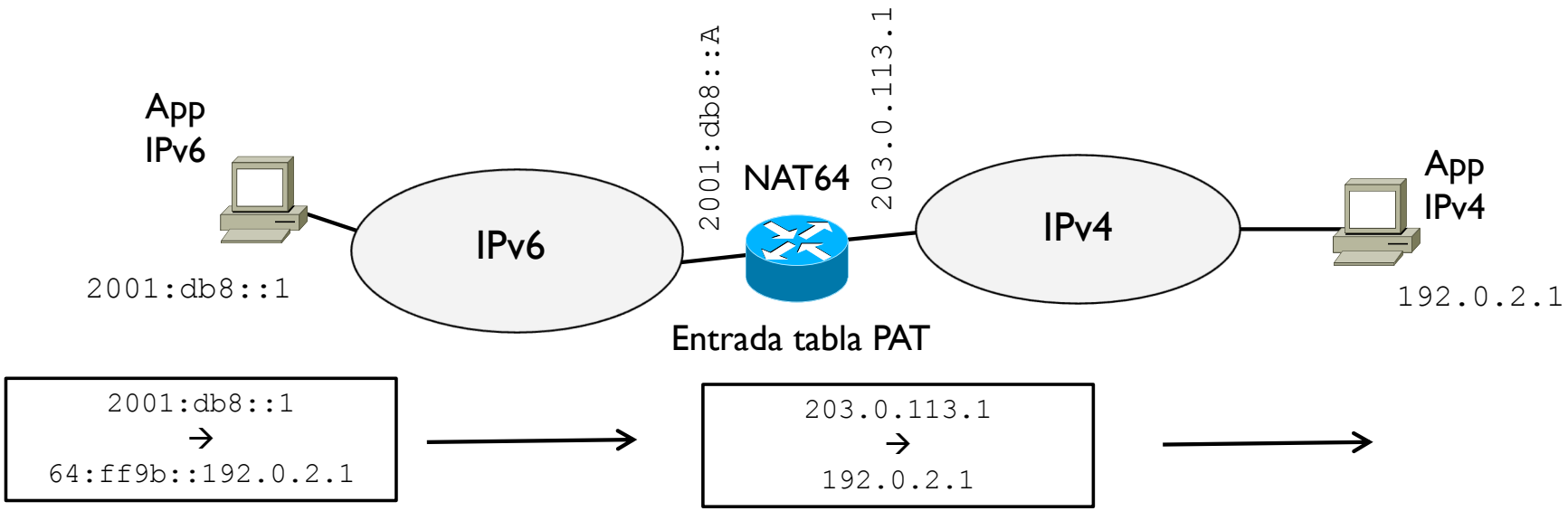
- Cuando dos aplicaciones pueden usar IPv6 pero en el camino hay IPv4



- 6to4 tunnel → creación del túnel automático con 2002::/16 RFC 3056
- Teredo tunnel → si hay un NAT IPv4 RFC 4380
- Tunnel bróker → configuración a través de un server (bróker) RFC 3053

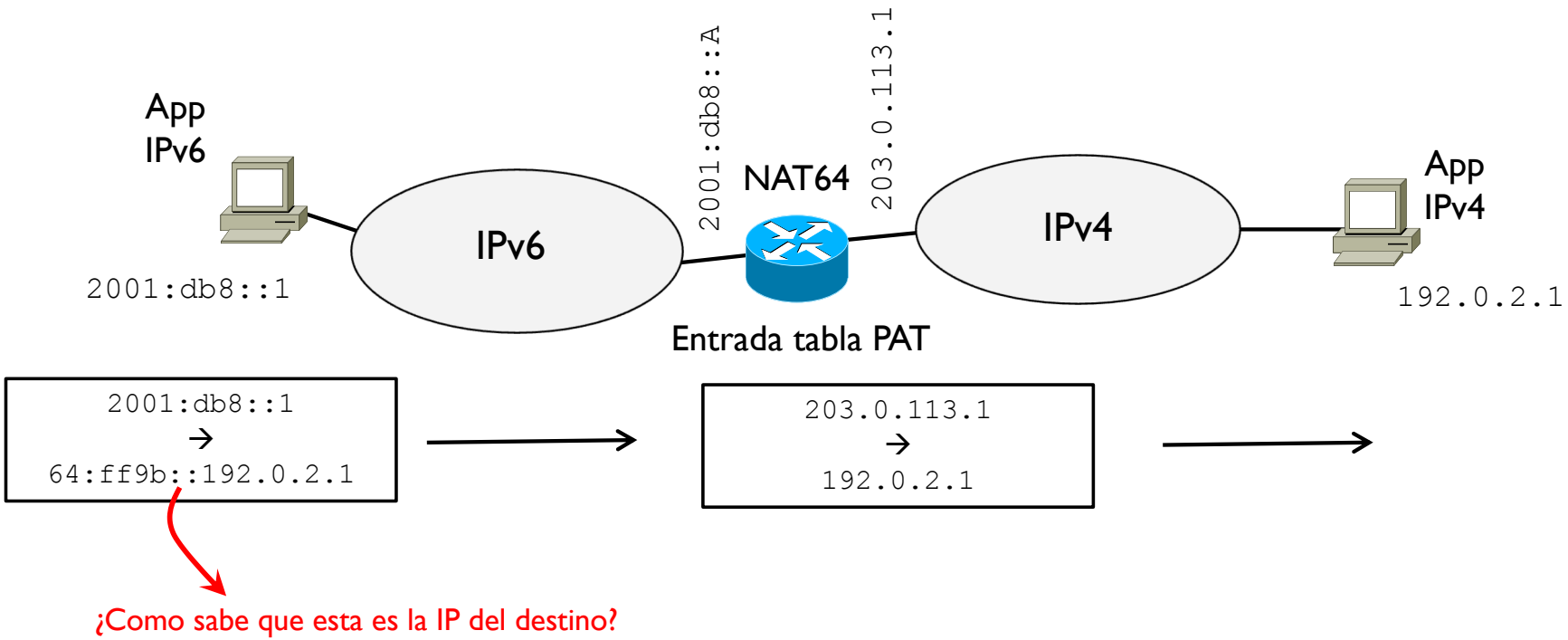
2.10 - Compatibilidad IPv4 – IPv6
NAT64

- Cuando una aplicación IPv6 comunica con una IPv4
- Rango reservado: `64:ff9b::/96`



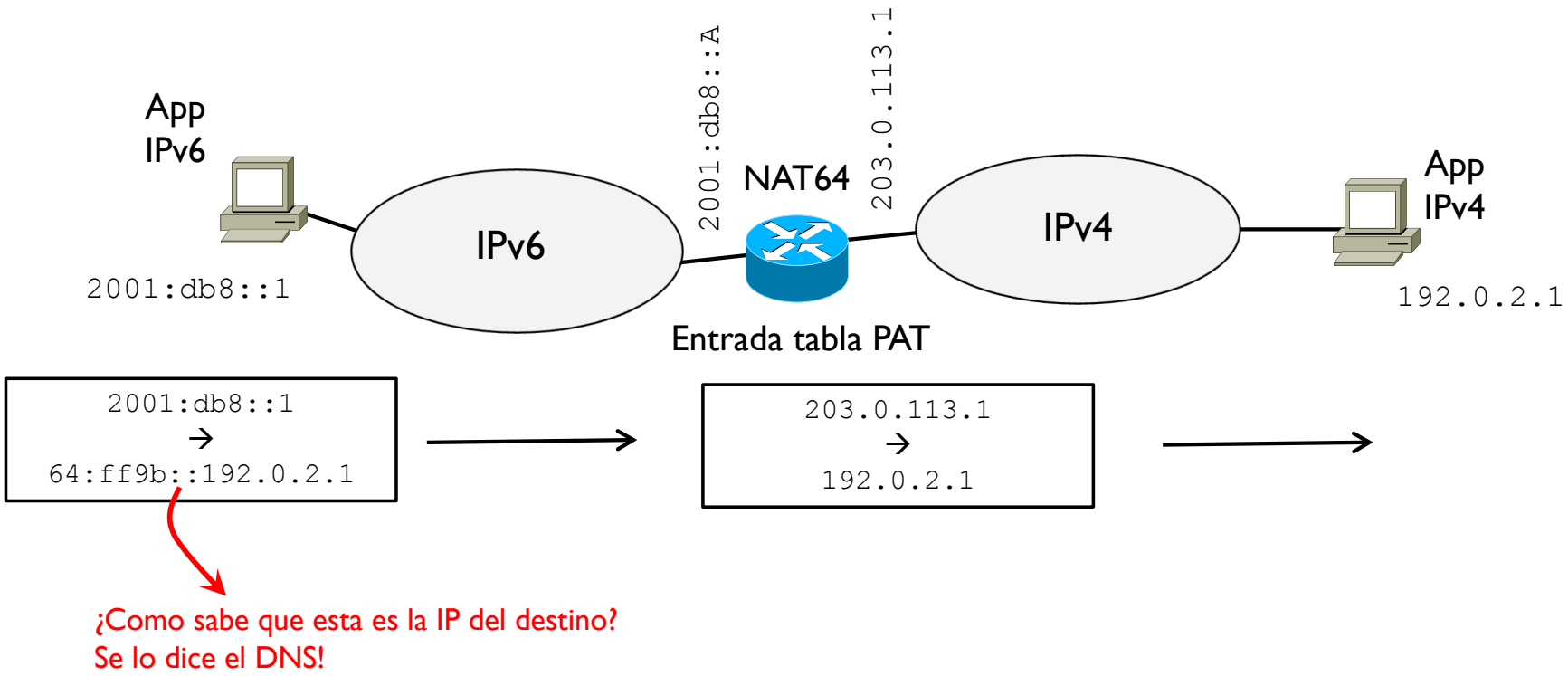
2.10 - Compatibilidad IPv4 – IPv6
NAT64

- Cuando una aplicación IPv6 comunica con una IPv4
- Rango reservado: `64:ff9b::/96`



2.10 - Compatibilidad IPv4 – IPv6
NAT64

- Cuando una aplicación IPv6 comunica con una IPv4
- Rango reservado: `64:ff9b::/96`



2.10 - Compatibilidad IPv4 – IPv6

Otros cambios

- RIP → RIPng RFC 2080
- OSPF → OSPFv6 RFC 5340
- BGP → BGP4+ RFC 2545

- DNS → nuevos RR para asociar IPv6 y nombres

Xarxes de Computadors II

Tema 2. Direccionamiento IPv6

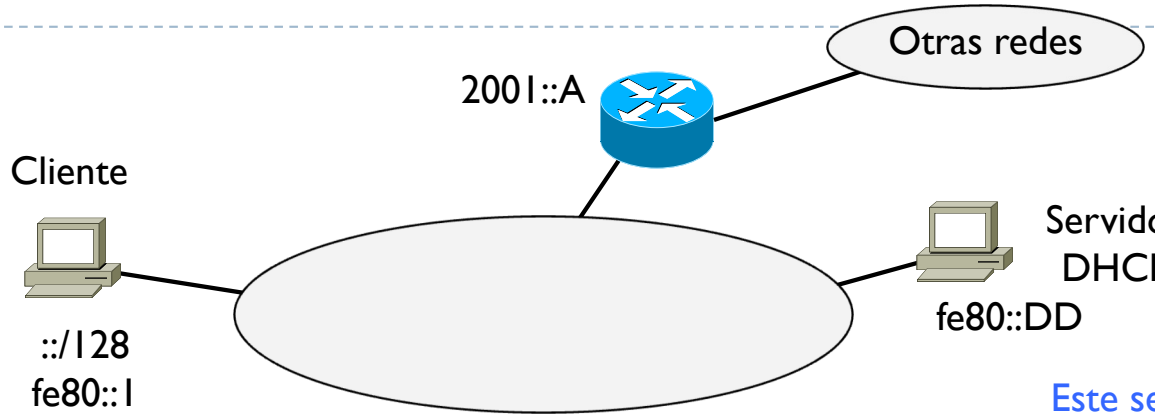
Davide Careglio

Xarxes de Computadors II

Tema 2. Direccionamiento IPv6 - BACKUP

Davide Careglio

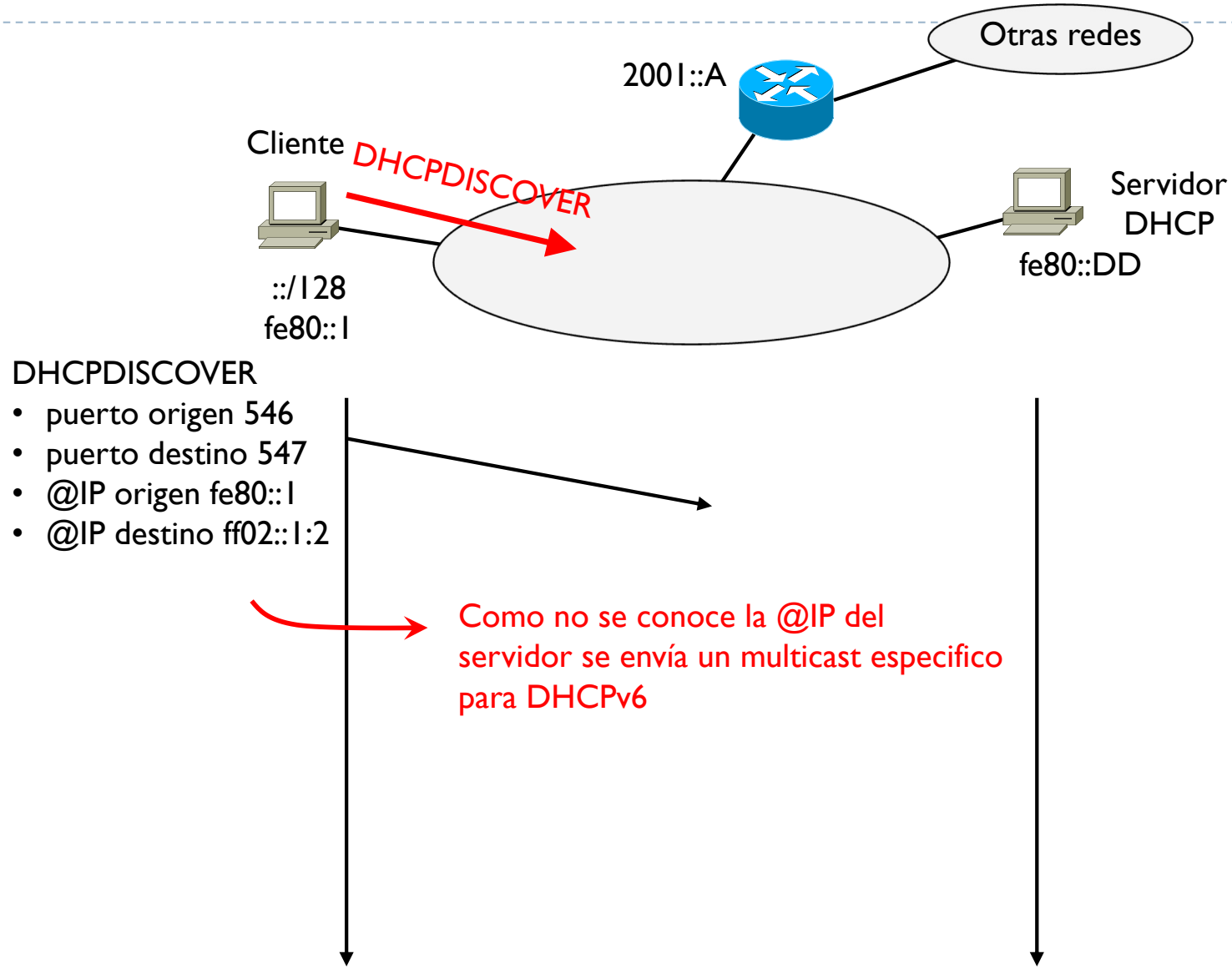
2.9 - DHCPv6



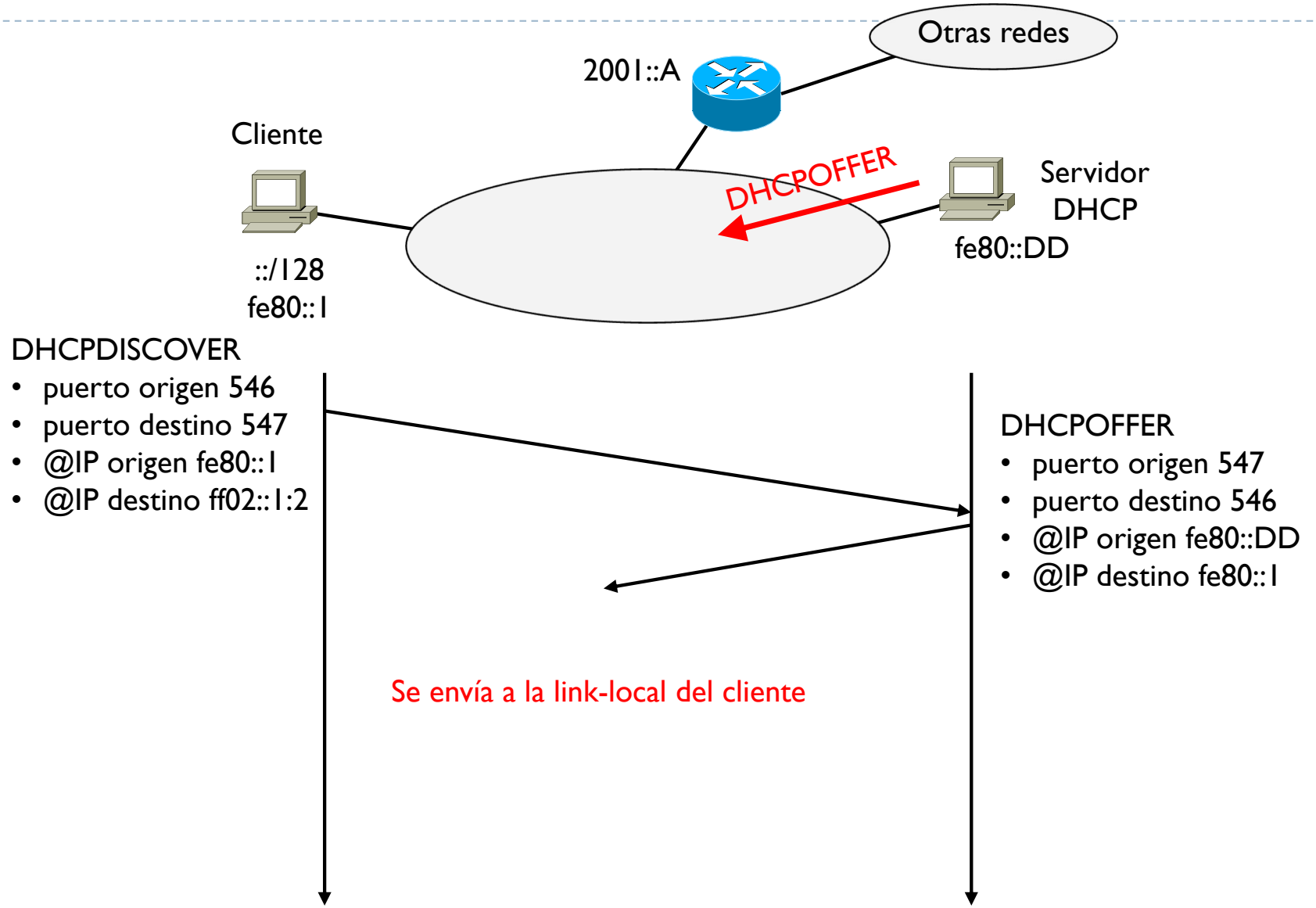
Este host al principio no tienen ninguna @IP global
Pero puede tener una link-local

Este servidor tiene el socket abierto en escuche para proporcionar servicio DHCP, puerto 547

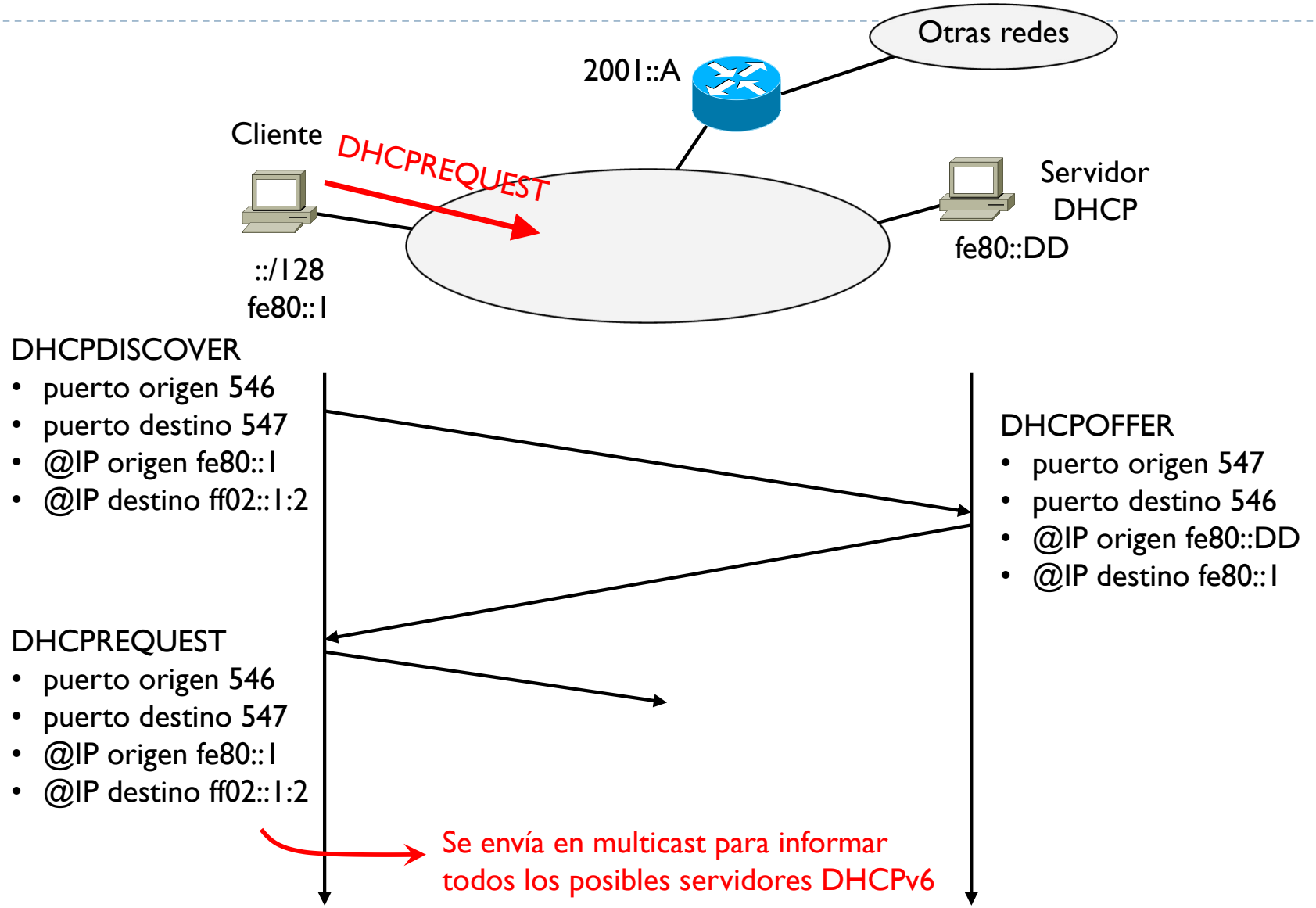
2.9 - DHCPv6


Otras redes
2001::A
Cliente
DHCPDISCOVER
Servidor
DHCP
fe80::DD
::/128
fe80::1
DHCPDISCOVER
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
Como no se conoce la @IP del servidor se envía un multicast especifico para DHCPv6

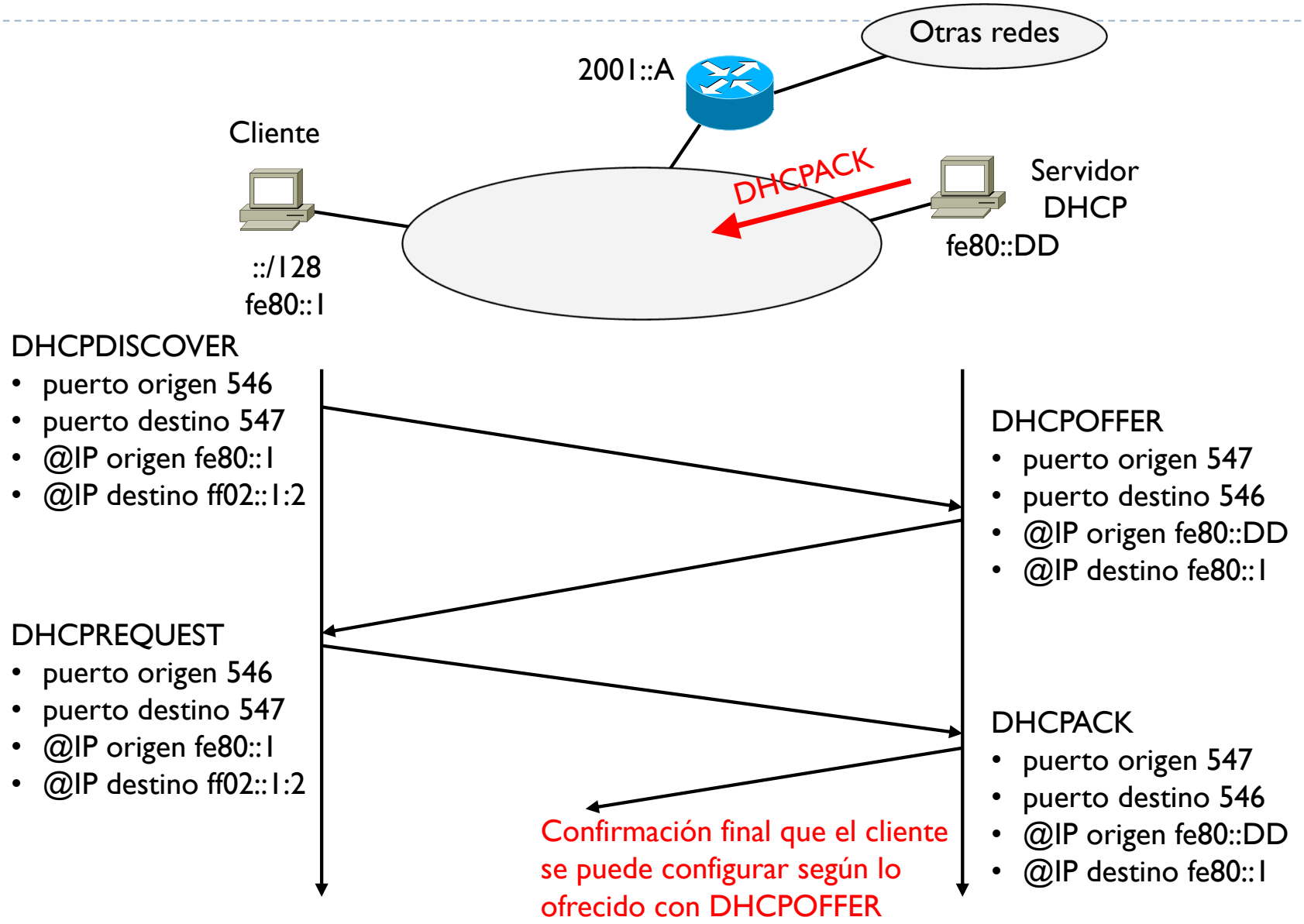
2.9 - DHCPv6


Otras redes
2001::A
Cliente
DHCPOFFER
Servidor
DHCP
fe80::DD
::/128
fe80::1
DHCPDISCOVER
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
DHCPOFFER
• puerto origen 547
• puerto destino 546
• @IP origen fe80::DD
• @IP destino fe80::1
Se envía a la link-local del cliente

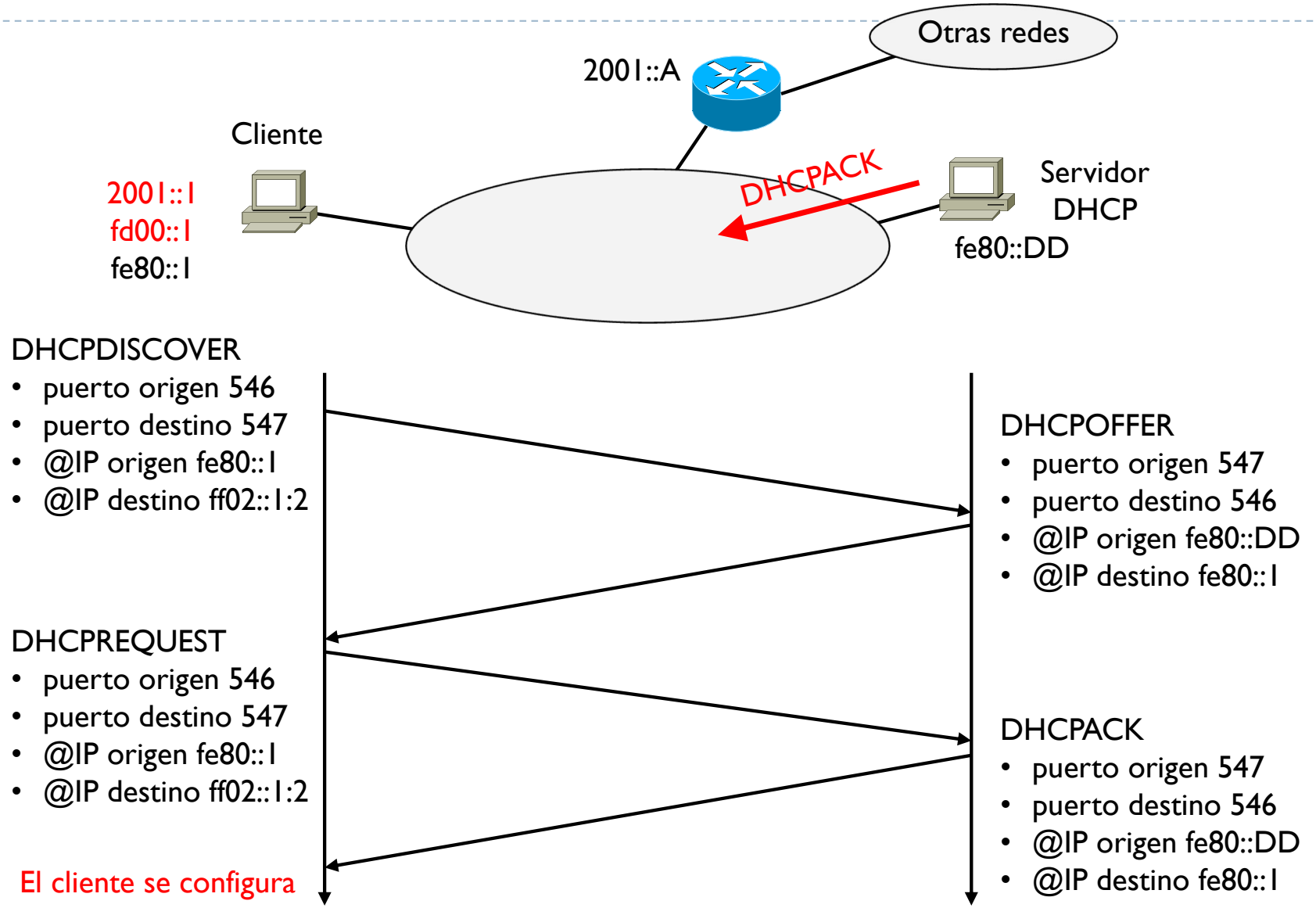
2.9 - DHCPv6


Otras redes
2001::A
Cliente
DHCPREQUEST
Servidor
DHCP
fe80::DD
::/128
fe80::1
DHCPDISCOVER
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
DHCPOFFER
• puerto origen 547
• puerto destino 546
• @IP origen fe80::DD
• @IP destino fe80::1
DHCPREQUEST
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
Se envía en multicast para informar
todos los posibles servidores DHCPv6

2.9 - DHCPv6


Otras redes
2001::A
Cliente
DHCPACK
Servidor
DHCP
fe80::DD
::/128
fe80::1
DHCPDISCOVER
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
DHCPOFFER
• puerto origen 547
• puerto destino 546
• @IP origen fe80::DD
• @IP destino fe80::1
DHCPREQUEST
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
DHCPACK
• puerto origen 547
• puerto destino 546
• @IP origen fe80::DD
• @IP destino fe80::1
Confirmación final que el cliente se puede configurar según lo ofrecido con DHCPOFFER

2.9 - DHCPv6


Otras redes
2001::A
Cliente
2001::1
fd00::1
fe80::1
DHCPACK
Servidor
DHCP
fe80::DD
DHCPDISCOVER
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
DHCPOFFER
• puerto origen 547
• puerto destino 546
• @IP origen fe80::DD
• @IP destino fe80::1
DHCPREQUEST
• puerto origen 546
• puerto destino 547
• @IP origen fe80::1
• @IP destino ff02::1:2
DHCPACK
• puerto origen 547
• puerto destino 546
• @IP origen fe80::DD
• @IP destino fe80::1
El cliente se configura