

Classless Interdomain Routing

Area de Ingeniería Telemática
<http://www.tlm.unavarra.es>

Arquitectura de Redes, Sistemas y Servicios
3º Ingeniería de Telecomunicación

Temario

1. Introducción
2. Arquitecturas, protocolos y estándares
3. **Conmutación de paquetes**
 - Arquitectura de protocolos para LANs
 - Ethernet
 - Protocolos de Internet
 - Internetworking
 - **Direccionamiento**
 - Fragmentación e ICMP. IP en LAN
4. Conmutación de circuitos
5. Tecnologías
6. Control de acceso al medio en redes de área local
7. Servicios de Internet

Objetivo

- Cómo asignar direcciones a redes y hosts
- Esquema de direccionamiento actual

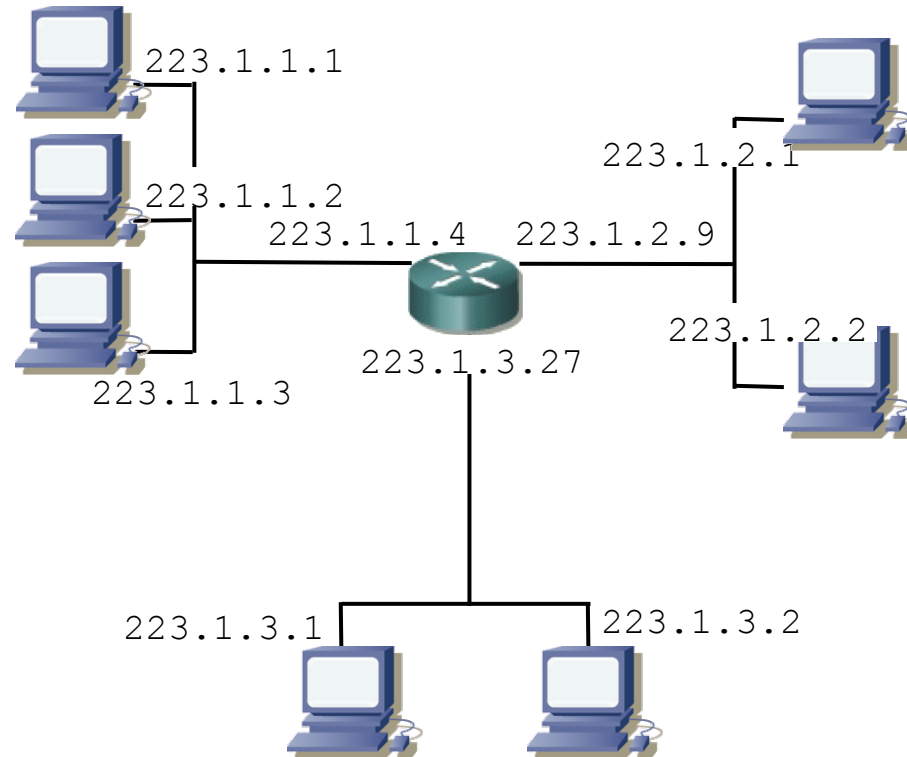


Hemos visto



Direccionamiento IP

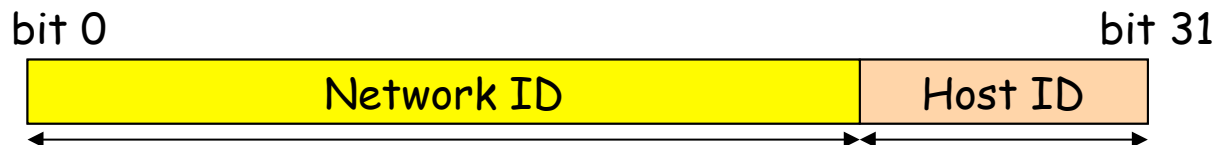
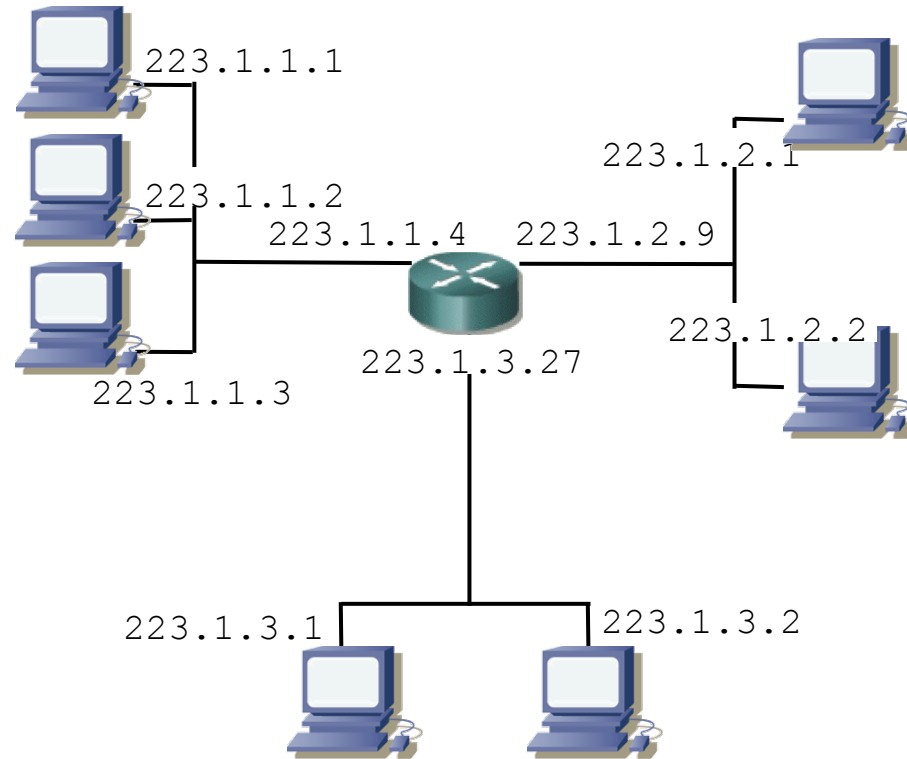
- **Interfaz:** Conexión entre un host/router y una red (subred)
 - Los routers típicamente tienen varios interfaces
 - Los hosts podrían tener varios interfaces
- **Dirección IP:** identificador de 32bits para un interfaz de un host o router



$$223.1.1.1 = \underbrace{11011111}_{223} \underbrace{00000001}_1 \underbrace{00000001}_1 \underbrace{00000001}_1$$

Subredes

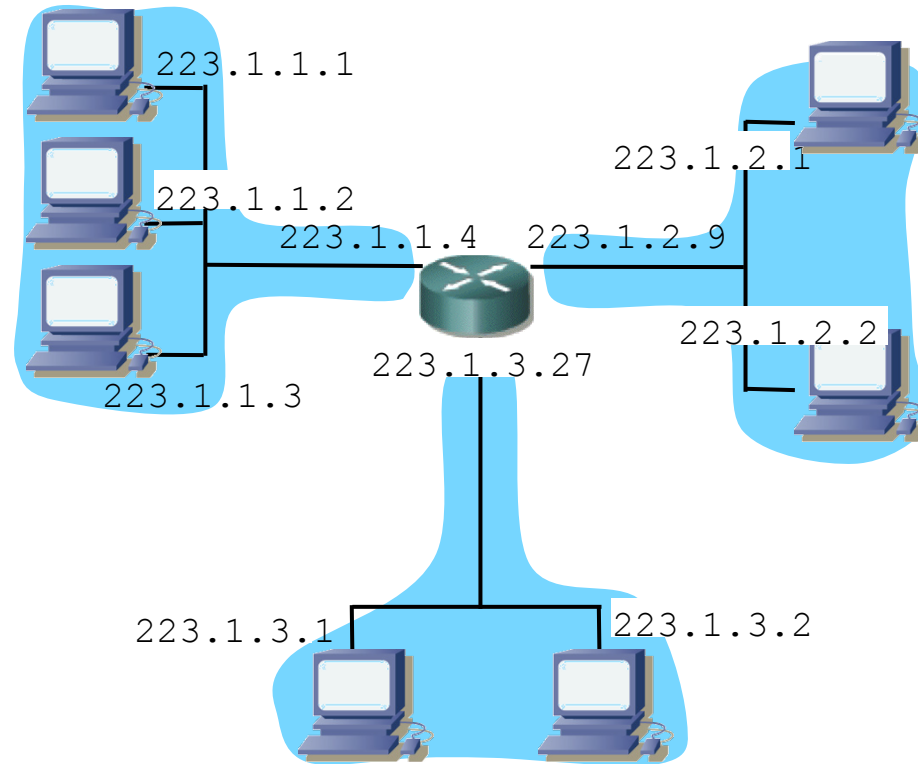
- Dos partes en la IP:
 - Identificador de la red (bits más significativos)
 - Identificador del host (bits menos significativos)
- *¿Qué es una subred?*
 - Interfaces de red cuyas direcciones tienen el mismo identificador de red
 - Cada uno puede comunicarse con otro en su misma subred **sin emplear un router**
 - Para reconocerlas (...)



Subredes

Para reconocer las subredes presentes:

- Desconecte los interfaces de los routers
- Se crean zonas aisladas: las subredes (...)
- Redes sin hosts (... ..)

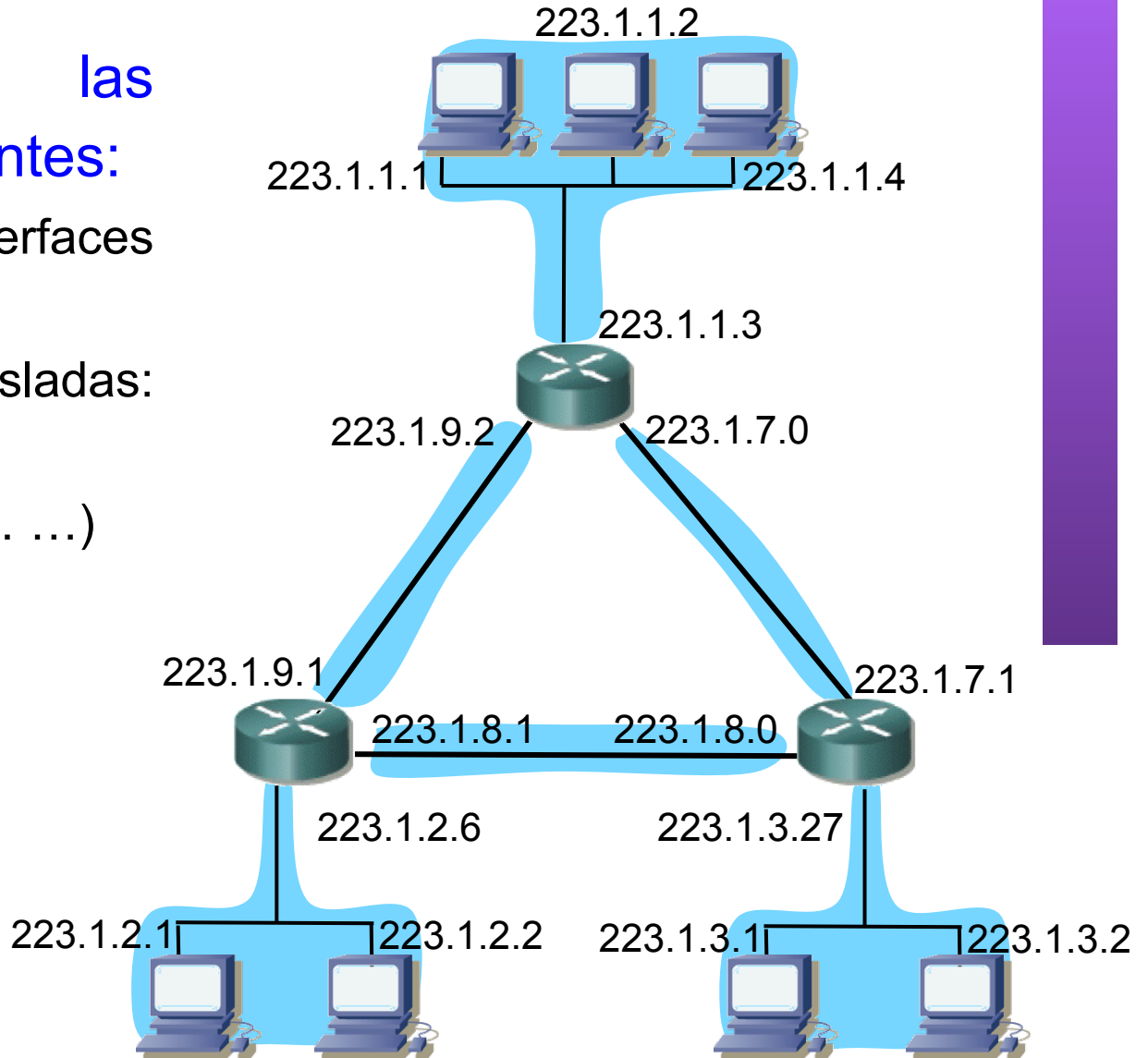


Red formada por 3 subredes

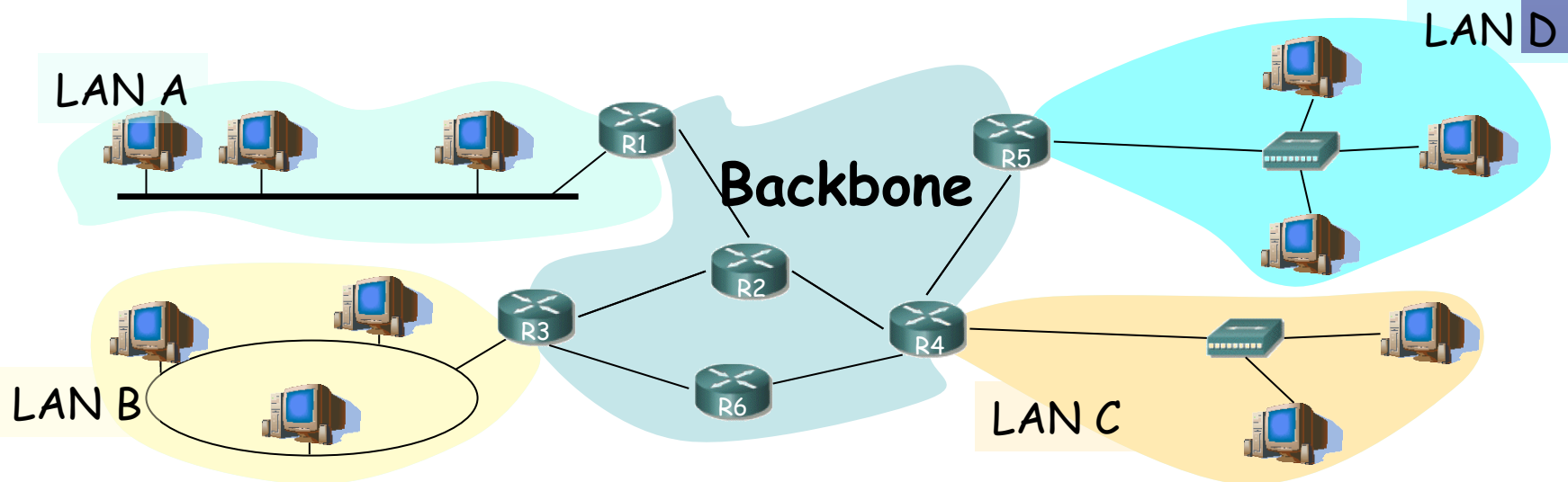
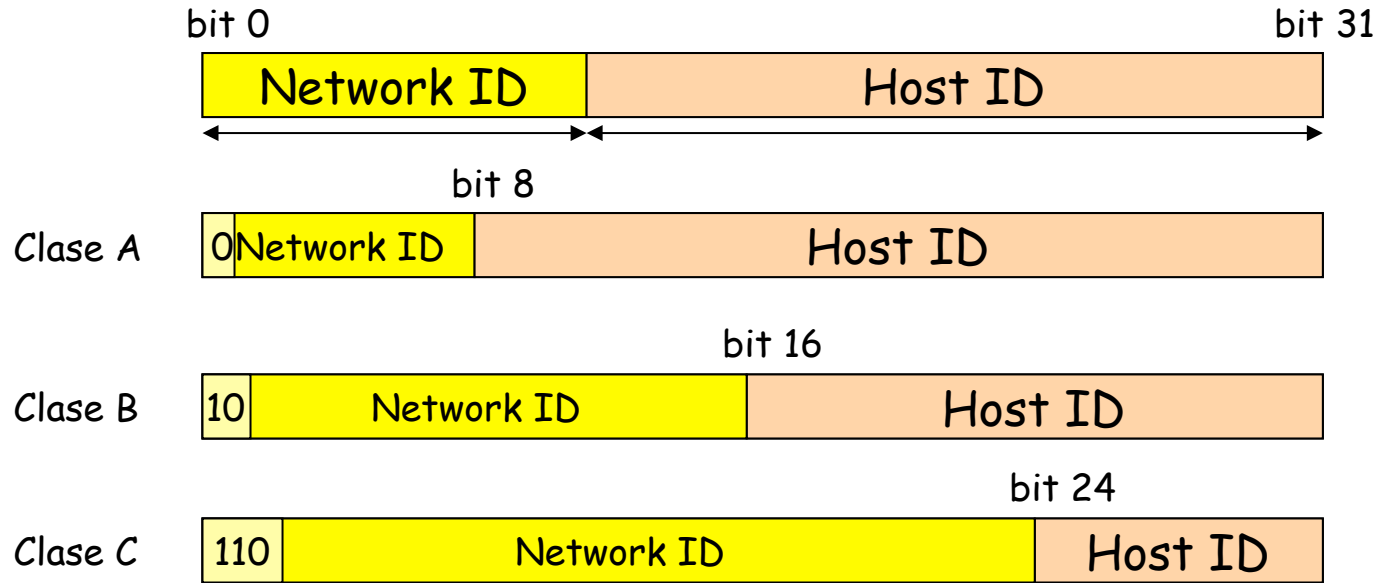
Subredes

Para reconocer las subredes presentes:

- Desconecte los interfaces de los routers
- Se crean zonas aisladas: las subredes (...)
- Redes sin hosts (... ..)



Direccionamiento Classful

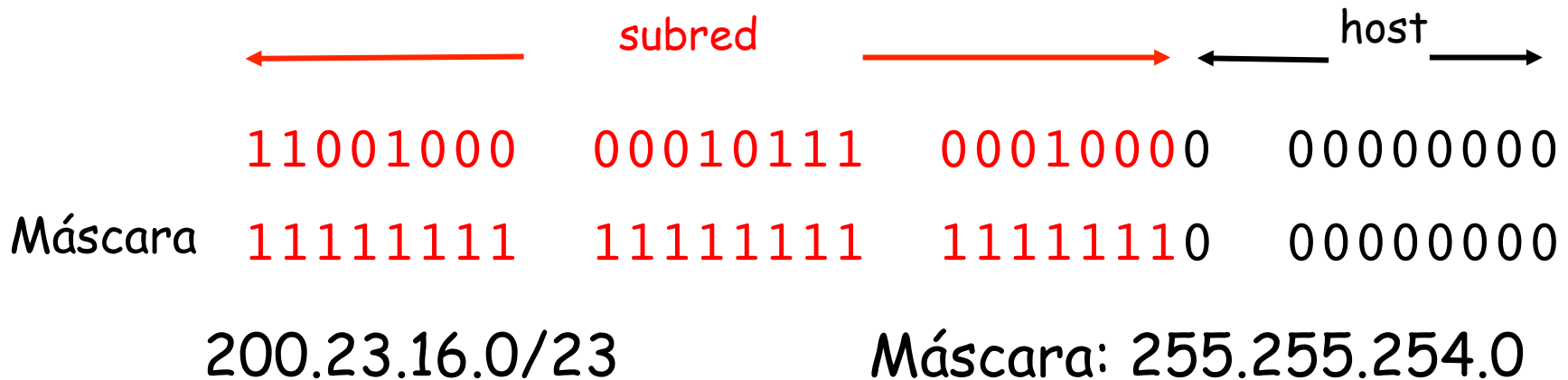


CIDR

Direccionamiento IP: CIDR

CIDR: Classless InterDomain Routing

- Identificador de subred de cualquier longitud
- **a.b.c.d/x**, donde x es el número de bits en el identificador de subred
- Otra forma de marcar la separación es mediante la **máscara de subred**



¿Una IP en una Red?

¿Cómo se puede saber con facilidad si una dirección IP pertenece a una Red?

- “Aplicar” la máscara:
- Operación AND entre dirección IP y máscara
- Si el resultado es la dirección de red, es parte de ella

¿ 200.23.17.42 pertenece a la red 200.23.16.0/23 ?

	11001000	00010111	00010001	00101010
AND	11111111	11111111	11111110	00000000
<hr/>				
	11001000	00010111	00010000	00000000

Debe salir la dirección de la red: 200.23.16.0

Valores reservados

- Host-ID todo 0's: dirección reservada para hacer referencia a la red (**dirección de red**)
200.23.16.0/23 → 200.23.16.0 11001000 00010111 00010000 00000000
- Host-ID todo 1's: hace referencia a todos los hosts de la subred : **Dirección de Broadcast de subred**
200.23.16.0/23 → 200.23.17.255 11001000 00010111 00010001 11111111
- Otra dirección de broadcast es la dirección de **broadcast limitado**:
 - Todo 1's = 255.255.255.255
 - Es independiente de la red
 - Paquetes dirigidos a esa IP nunca son reenviados por los routers
- Direcciones reservadas para **redes privadas**:
 - 10.0.0.0/8
 - 172.16.0.0/12
 - 192.168.0.0/16
 - Paquetes con ese origen o destino nunca deben llegar a Internet

CIDR

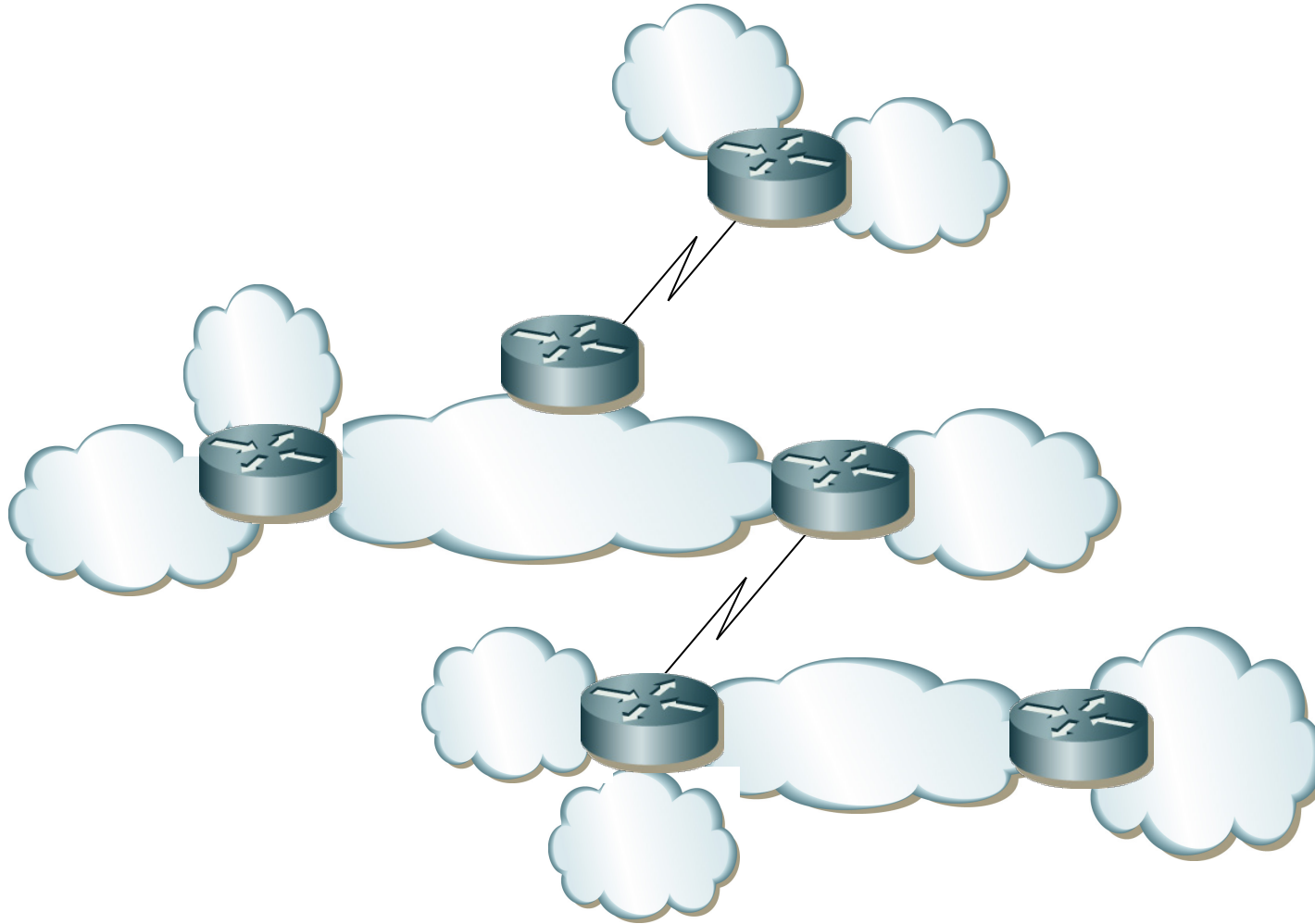
¿Cómo actúan los hosts y los routers?

- Tienen configurado:
 - Dirección IP en cada interfaz
 - Máscara en cada uno
 - Tabla de rutas

Destino	Máscara	Next-hop	Interfaz
Dir.Red	Máscara	IP_next	If X
...

- Ojo: la máscara en una ruta no tiene por qué ser la de una red final
- IP_D que no es ninguna de sus direcciones IP
 - Comprueba con cada ruta si lleva hacia IP_D :
 - $(IP_D \text{ AND } \text{Máscara}) == \text{Dir.Red}$? válida : no válida
 - ¿ Ninguna ruta es válida ? \Rightarrow descarta paquete
- Ruta por defecto es simplemente 0.0.0.0/0
- Escoge la ruta válida con **prefijo más largo** (máscara con más 1's)
- **Longest Prefix Match**

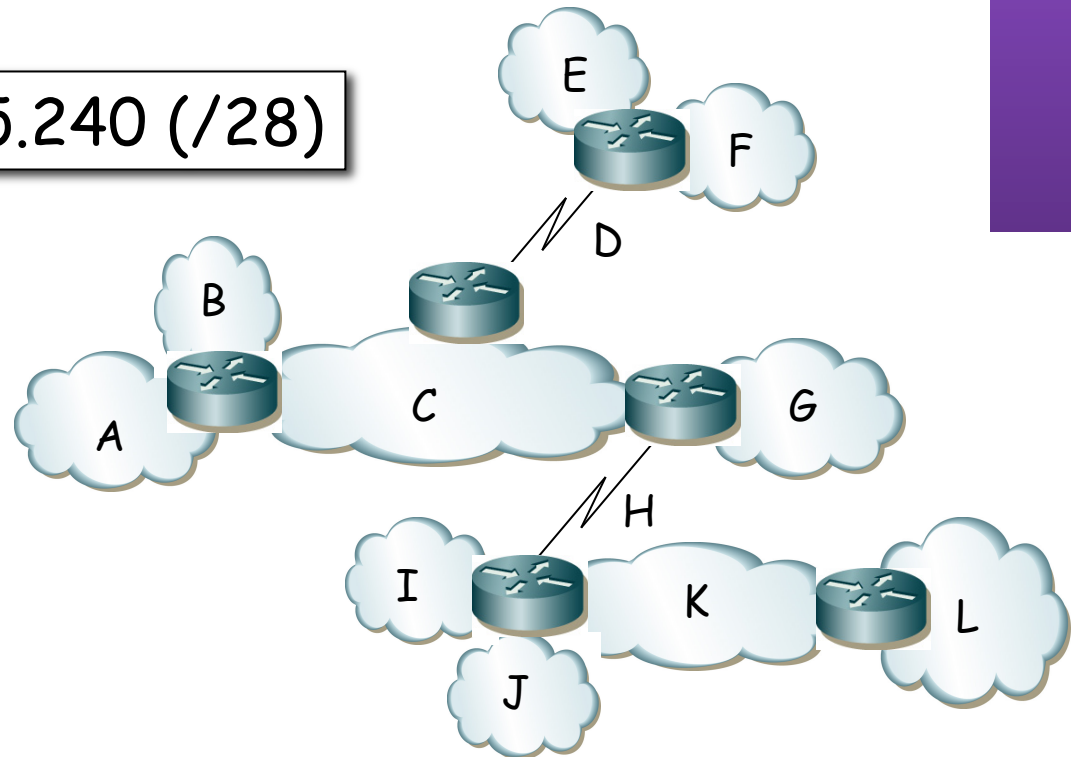
Ejemplo



Ejemplo

- Misma máscara para todas las subredes
- FLSM (Fixed Length Subnet Masks)
- 12 Subredes $\rightarrow 2^4=16$, 4 bits subnetwork-id
- Máximo 10 hosts por red (+2) $\rightarrow 2^4=16$, 4 bits host-id
- Red 192.168.3.0/24 $\rightarrow 192.168.3. [0000] [0000]$

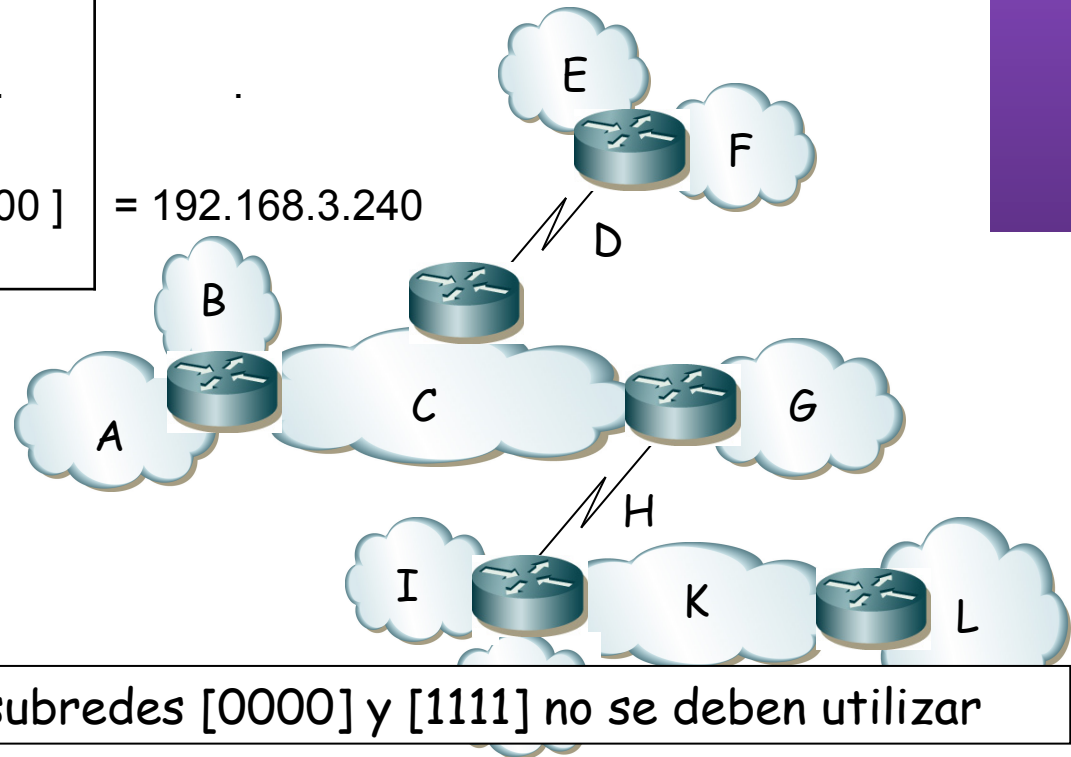
Máscara 255.255.255.240 (/28)



Ejemplo

A	192.168.3	.	[0000]	[0000]	= 192.168.3.0
B	192.168.3	.	[0001]	[0000]	= 192.168.3.16
C	192.168.3	.	[0010]	[0000]	= 192.168.3.32
.
.
O	192.168.3	.	[1111]	[0000]	= 192.168.3.240

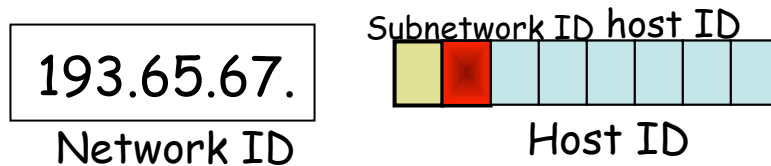
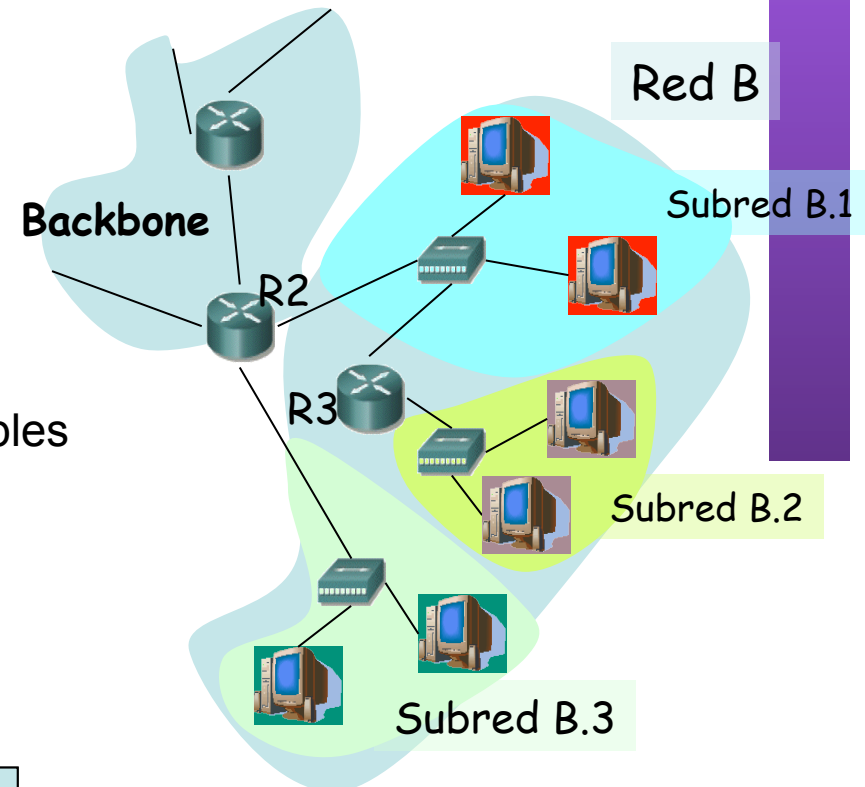
Subredes



Ojo, en cierto escenarios las subredes [0000] y [1111] no se deben utilizar

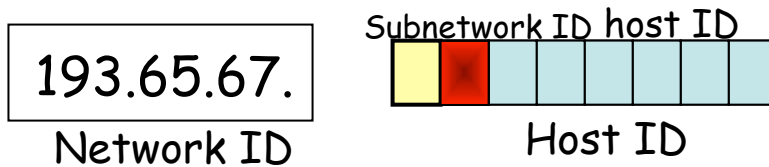
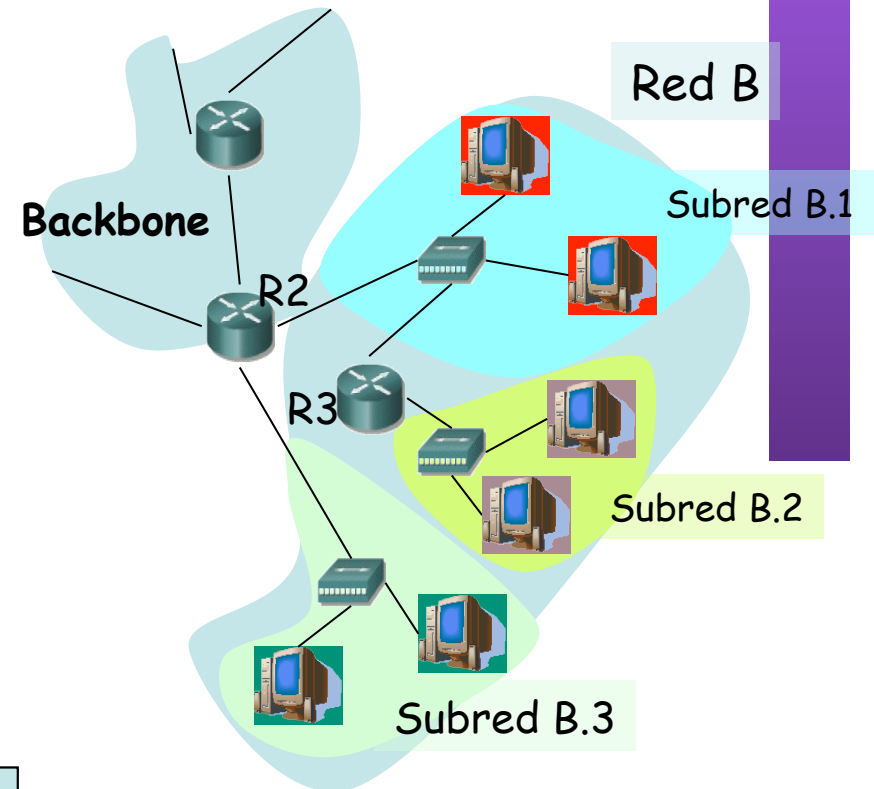
Problemas con FLISM

- Todas las subredes deben emplear la misma máscara
- Subredes de tamaño heterogéneo desaprovechan direcciones
- Ejemplo:
 - Red 193.65.67.0
 - Se crean 3 subredes
 - B.1: Al menos 80 hosts
 - B.2: Al menos 20 hosts
 - B.3: Al menos 20 hosts
 - Total: 120 hosts
 - Clase C \Rightarrow 256 direcciones disponibles
 - 3 subredes \Rightarrow SubNetID 2 bits (...)
 - B.1 80 hosts \Rightarrow HostID > 6 bits (...)



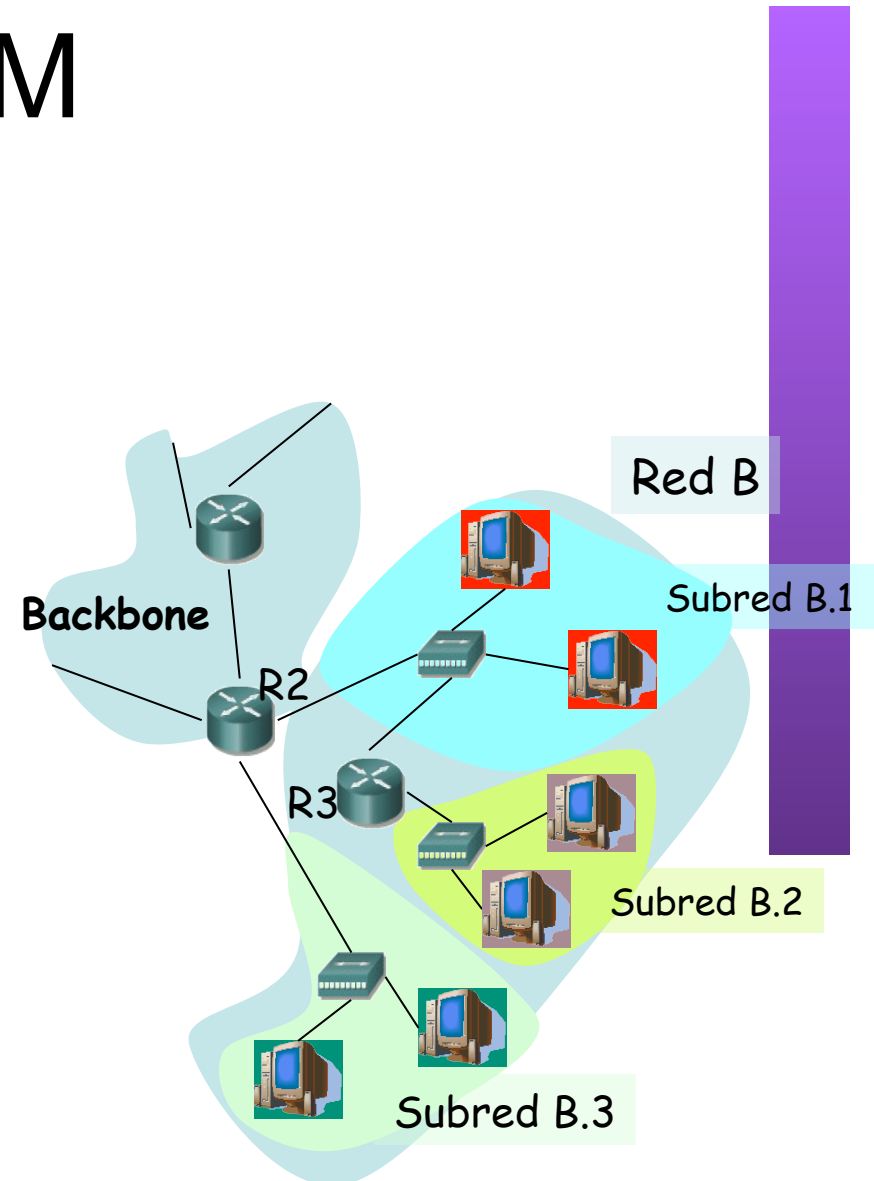
Problemas con FLISM

- ¿Dónde se han perdido las direcciones?
- Las 3 subredes dimensionadas con el tamaño de la mayor (máscara fija)
- Una subred no se usa
- ¡Todas son del mismo tamaño que la mayor!



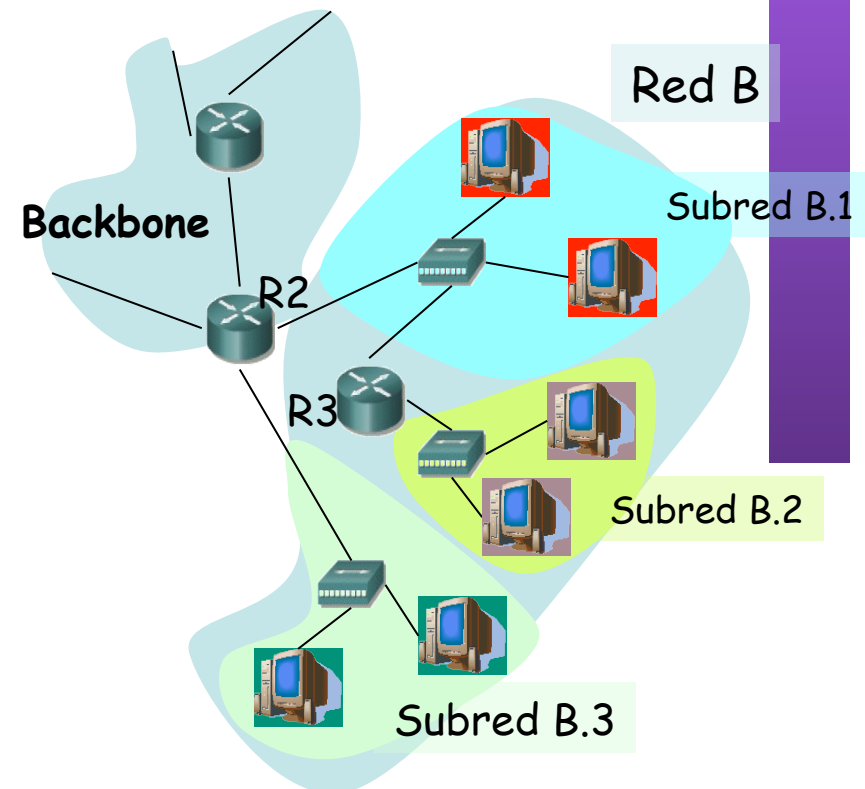
VLSM

- Subnetting = FLSM
- En Subnetting clásico no se podrían usar subredes de Subnetwork ID todo 0s o todo 1s
- FLSM es “one-size-fits-all”
- ¿Cómo ajustar mejor el tamaño de cada subred?
 - VLSM = Variable Length Subnet Masks
- Ejemplo:
 - B.1 80 hosts \Rightarrow HostID = 7 bits
 193.65.67. [0 XXXXXXX]
 - B.2 20 hosts \Rightarrow HostID = 5 bits
 193.65.67. [10 0 XXXXX]
 - B.3 20 hosts \Rightarrow HostID = 5 bits
 193.65.67. [10 1 XXXXX]
 - Quedan sin asignar:
 193.65.67. [11 XXXXXX]



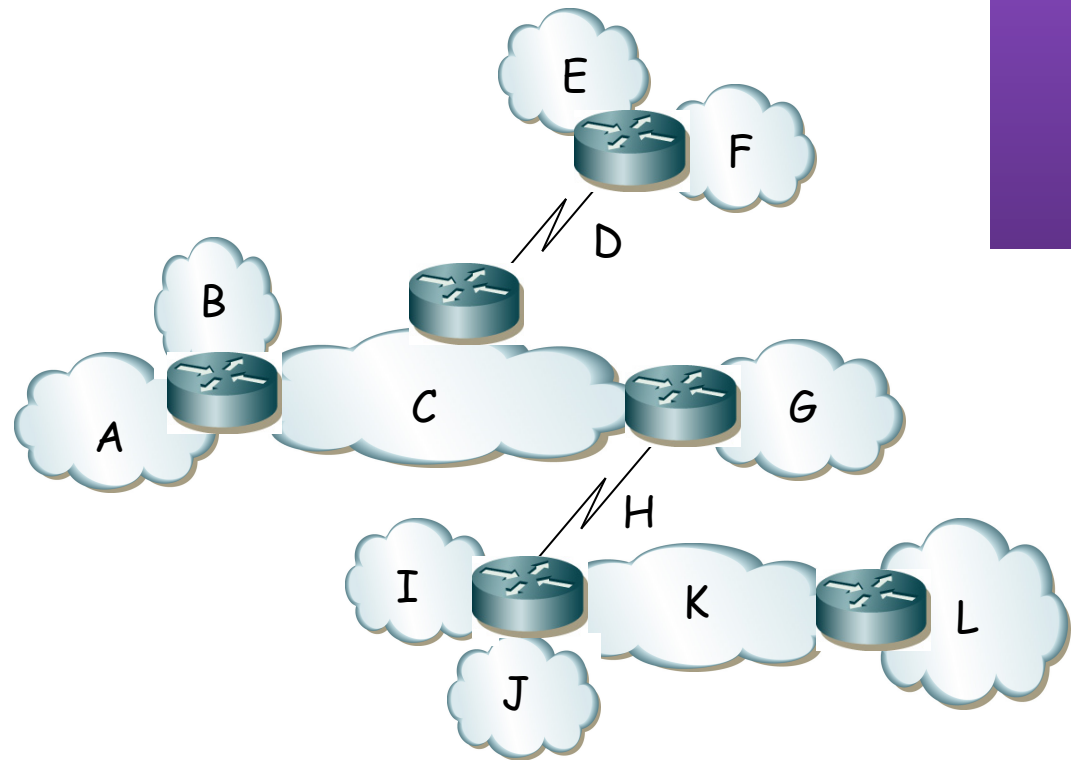
VLSM (Ejemplo)

- B.1 50 hosts \Rightarrow HostID = 6 bits
 193.65.67. [0 XXXXXXX]
 Dir. Red = 193.65.67.0
 Máscara = 255.255.255.128
- B.2 20 hosts \Rightarrow HostID = 5 bits
 193.65.67. [10 0 XXXXX]
 Dir. Red = 193.65.67.128
 Máscara = 255.255.255.224
- B.3 20 hosts \Rightarrow HostID = 5 bits
 193.65.67. [10 1 XXXXX]
 Dir. Red = 193.65.67.160
 Máscara = 255.255.255.224
- Quedan sin asignar:
 193.65.67. [11 XXXXXX]
 Dir. Red = 193.65.67.192
 Máscara = 255.255.255.192



Ejemplo (2)

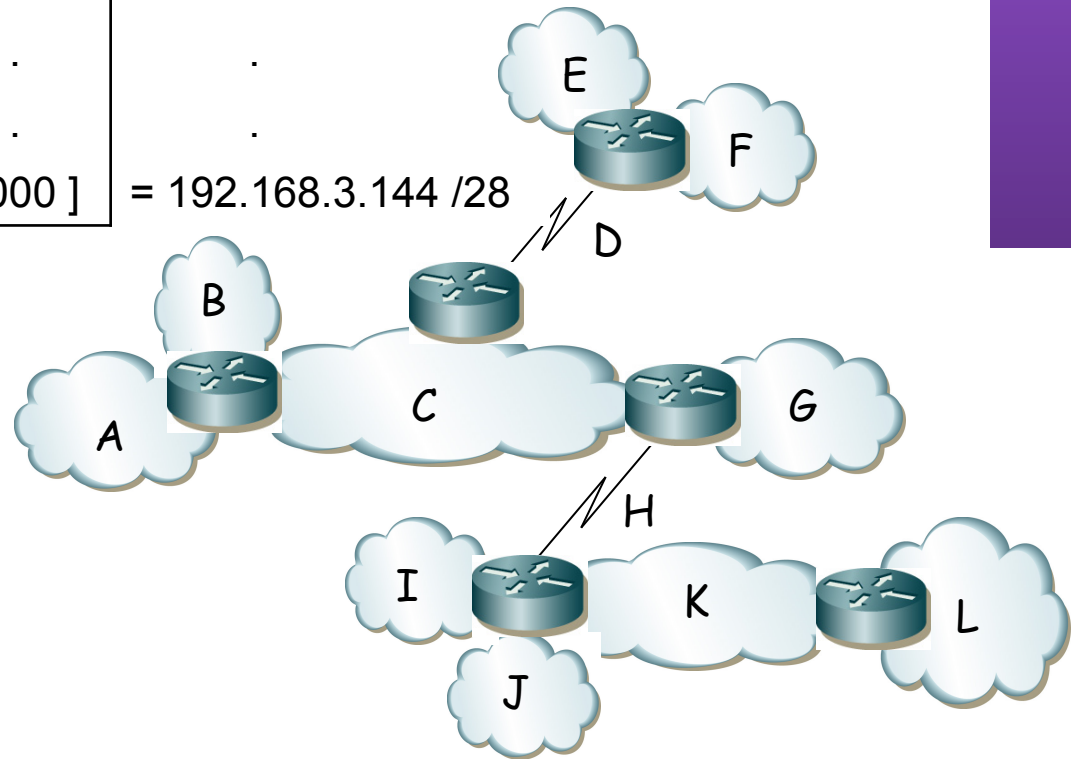
- Máscaras de longitud variable (VLSM o CIDR)
- Red D y H no van a necesitar más de 2 direcciones
- 10 redes de 10 hosts



Ejemplo (2)

- 10 redes de 10 hosts

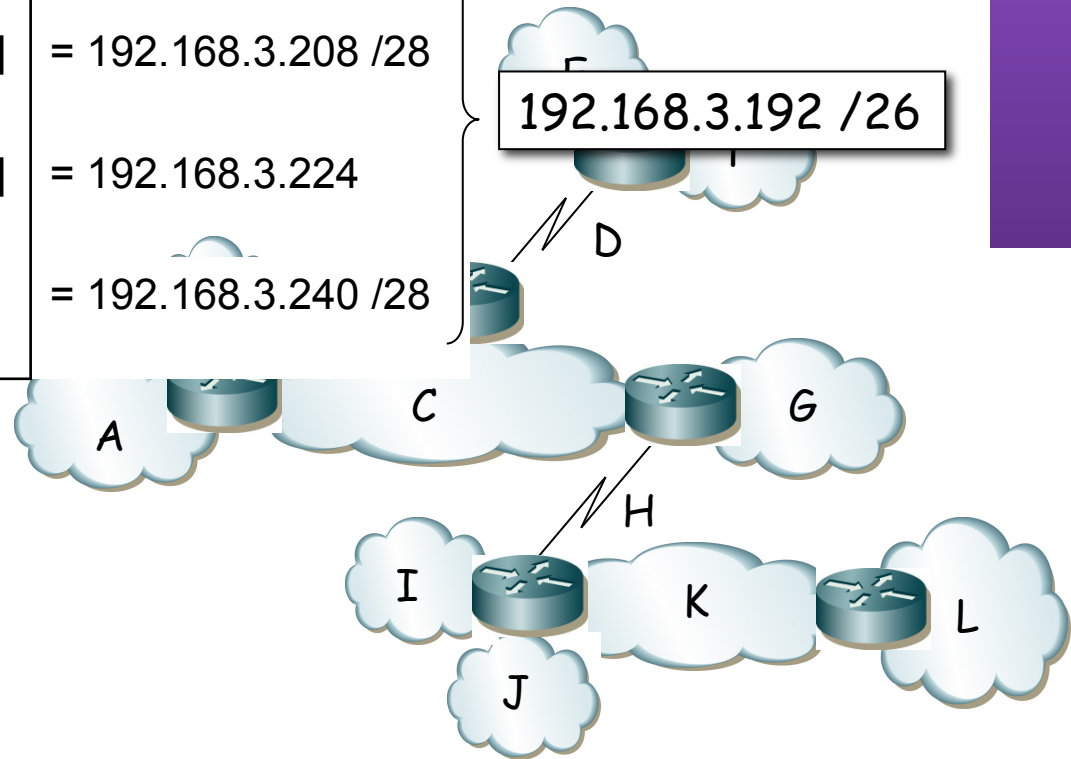
A	192.168.3	.	[0000]	[0000]	= 192.168.3.0 /28
B	192.168.3	.	[0001]	[0000]	= 192.168.3.16 /28
C	192.168.3	.	[0010]	[0000]	= 192.168.3.32 /28
.
.
L	192.168.3	.	[1001]	[0000]	= 192.168.3.144 /28



Ejemplo (2)

- Libres:

192.168.3	.	[1010]	[0000]	= 192.168.3.160 /28	} 192.168.3.160 /27
192.168.3	.	[1011]	[0000]	= 192.168.3.176 /28	
192.168.3	.	[1100]	[0000]	= 192.168.3.192 /28	} 192.168.3.192 /26
192.168.3	.	[1101]	[0000]	= 192.168.3.208 /28	
192.168.3	.	[1110]	[0000]	= 192.168.3.224	
192.168.3	.	[1111]	[0000]	= 192.168.3.240 /28	



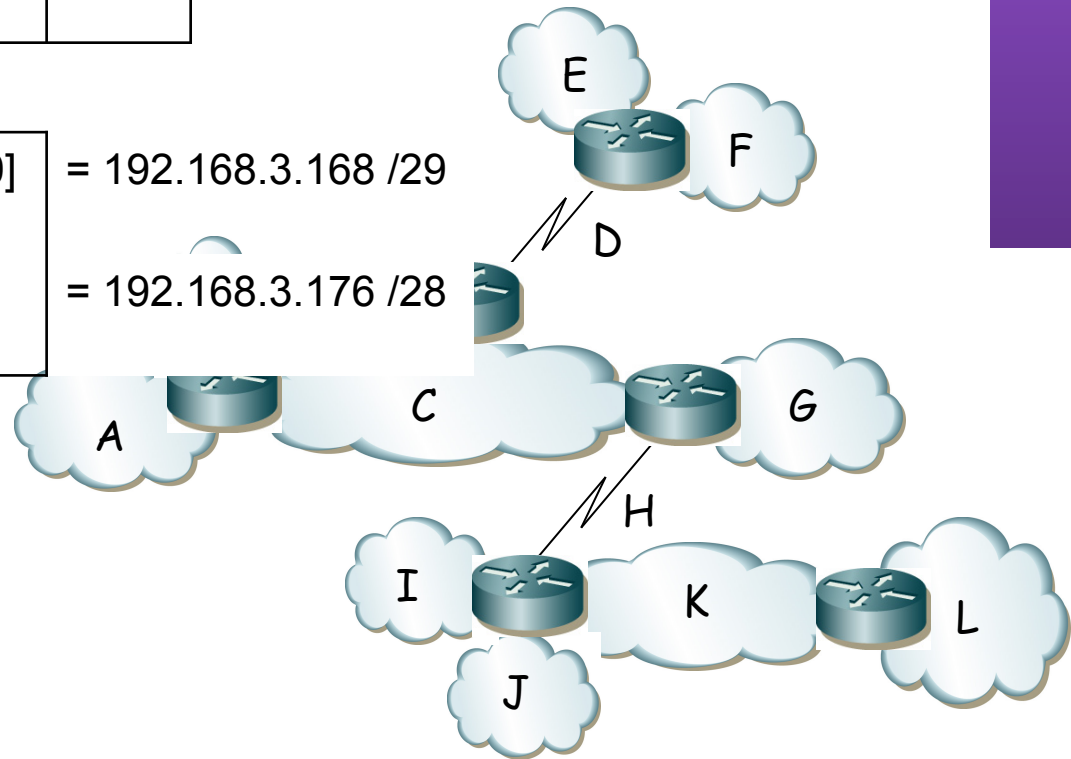
Ejemplo (2)

- 2 redes de 2 hosts
- 2 hosts (+2) → $2^2=4$, 2 bits para el host-id
- Por ejemplo dentro de 192.168.3.160 /27

D	192.168.3	.	[101]	[000]	[00]	= 192.168.3.160 /30
H	192.168.3	.	[101]	[001]	[00]	= 192.168.3.164 /30

Libres en 192.168.3.160 /27:

192.168.3	.	[101]	[0 1]	[000]	= 192.168.3.168 /29
192.168.3	.	[101]	[1]	[0000]	= 192.168.3.176 /28

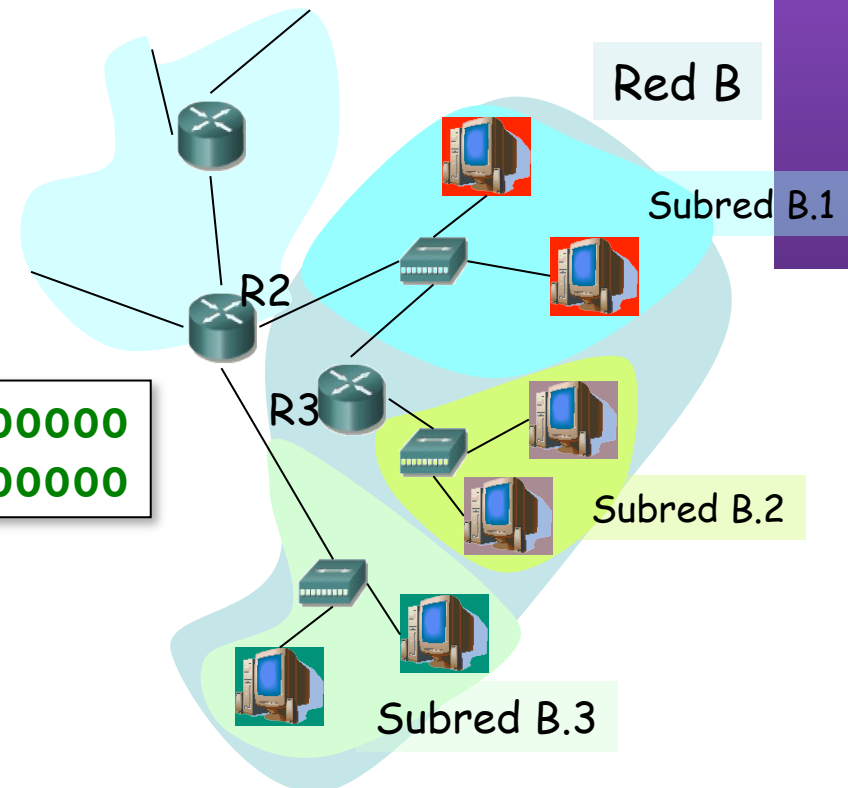


Ejemplo (2)

- Supongamos que nuestra red tiene asignado el espacio de direcciones 193.65.72.0/22
- Queremos subdividir nuestro espacio de direcciones en 3 redes (...)
- La subred B.1 tendrá 320 hosts
- La subred B.2 85 hosts
- La subred B.3 113 hosts
- Subred B.1:
 - $2^8 < 320 < 2^9$
 - 9 bits en el host-ID (...)

Mask=

11000001	01000001	01001000	00000000
11111111	11111111	11111100	00000000



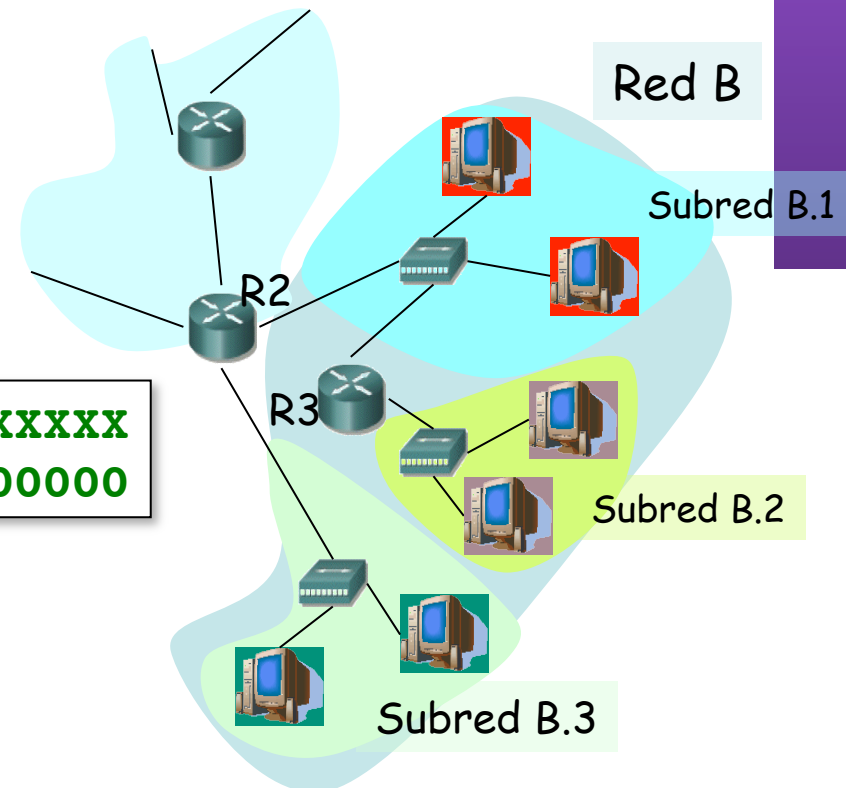
Ejemplo (2)

- Supongamos que nuestra red tiene asignado el espacio de direcciones 193.65.72.0/22
- Queremos subdividir nuestro espacio de direcciones en 3 redes (...)
- La subred B.1 tendrá 320 hosts
- La subred B.2 85 hosts
- La subred B.3 113 hosts
- Subred B.1:
 - $2^8 < 320 < 2^9$
 - 9 bits en el host-ID (...)

11000001	01000001	0100100X	XXXXXXXX
11111111	11111111	11111110	00000000

Mask=

- 193.65.72.0/23



Ejemplo (2)

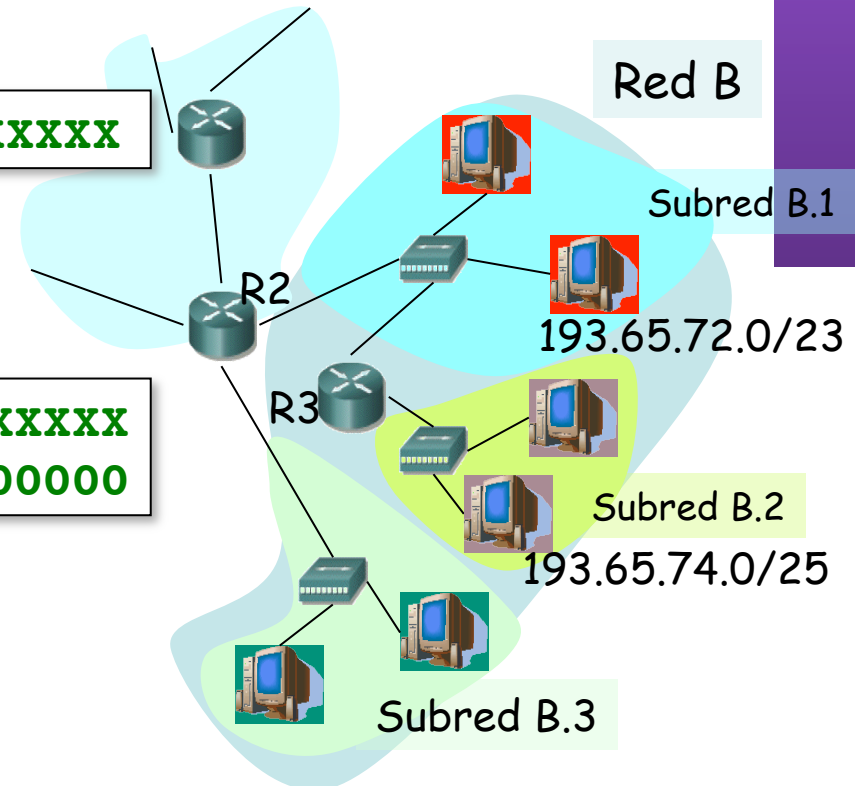
- Supongamos que nuestra red tiene asignado el espacio de direcciones 193.65.72.0/22
- Queremos subdividir nuestro espacio de direcciones en 3 redes (...)
- La subred B.1 tendrá 320 hosts
- La subred B.2 85 hosts
- La subred B.3 113 hosts

B.1 0x xxxxxxxx

- Subred B.2:
 - $2^6 < 85 < 2^7$
 - 7 bits en el host-ID (...)

Mask= **11000001 01000001 01001010 0xxxxxxx**
11111111 11111111 11111111 10000000

- 193.65.74.0/25



Ejemplo (2)

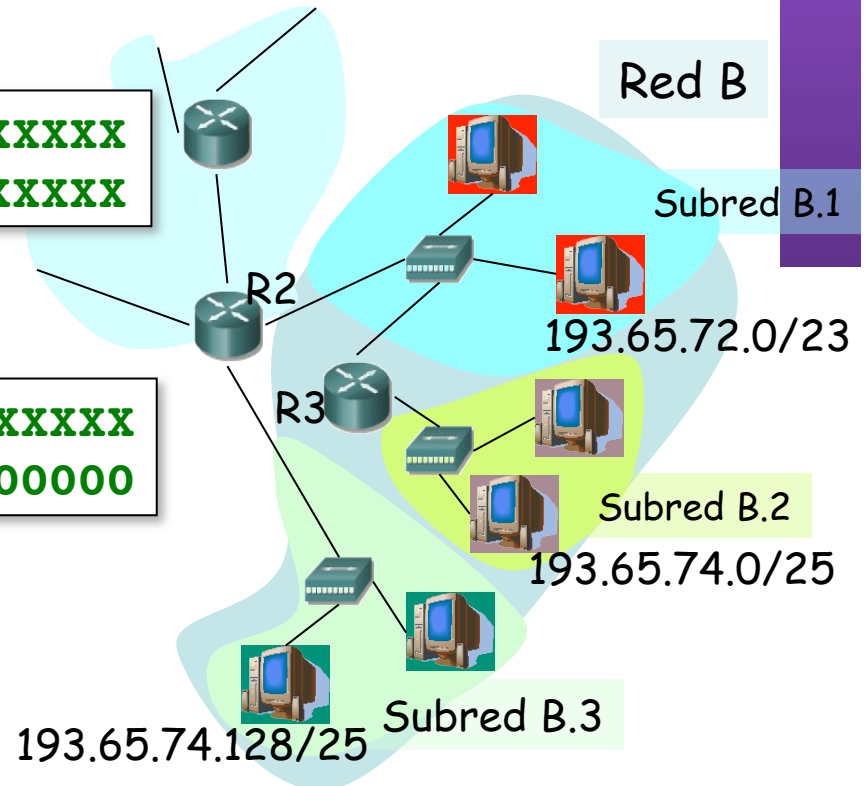
- Supongamos que nuestra red tiene asignado el espacio de direcciones 193.65.72.0/22
- Queremos subdividir nuestro espacio de direcciones en 3 redes (...)
- La subred B.1 tendrá 320 hosts
- La subred B.2 85 hosts
- La subred B.3 113 hosts

- Subred B.3:
 - $2^6 < 113 < 2^7$
 - 7 bits en el host-ID

B.1 0x XXXXXXXX
 B.2 10 OXXXXXXXX

Mask= 11000001 01000001 01001010 1XXXXXXXX
 11111111 11111111 11111111 10000000

- 193.65.74.128/25



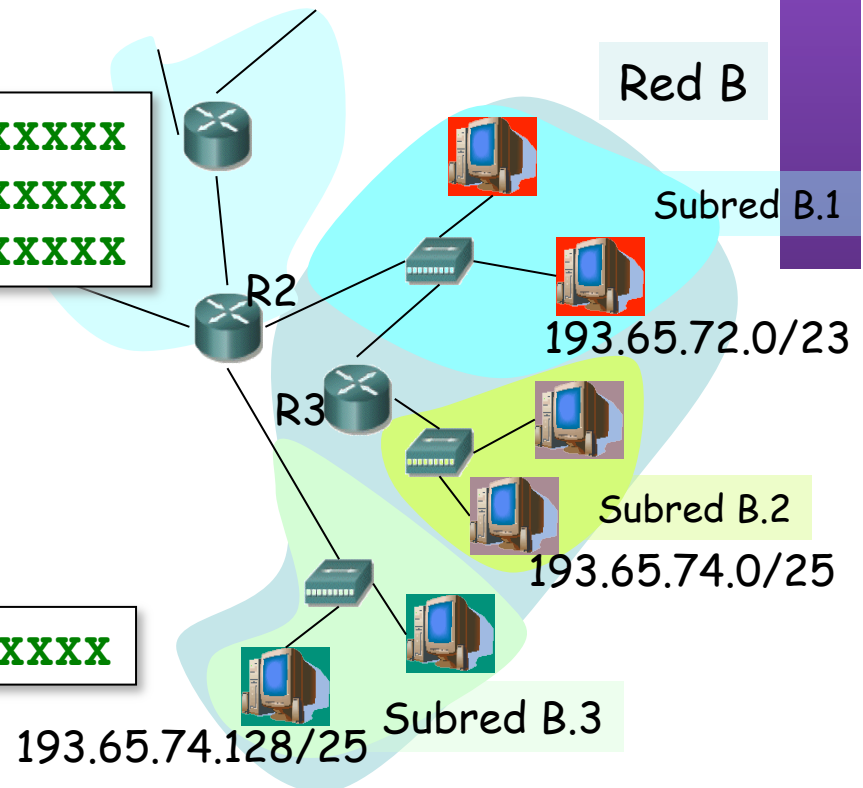
Ejemplo (2)

- Supongamos que nuestra red tiene asignado el espacio de direcciones 193.65.72.0/22
- Queremos subdividir nuestro espacio de direcciones en 3 redes (...)
- La subred B.1 tendrá 320 hosts
- La subred B.2 85 hosts
- La subred B.3 113 hosts

- B.1: 193.65.72.0/23
- B.2: 193.65.74.0/25
- B.3: 193.65.74.128/25
- No hay intersecciones
- Queda libre:
 - 193.65.75.0/24

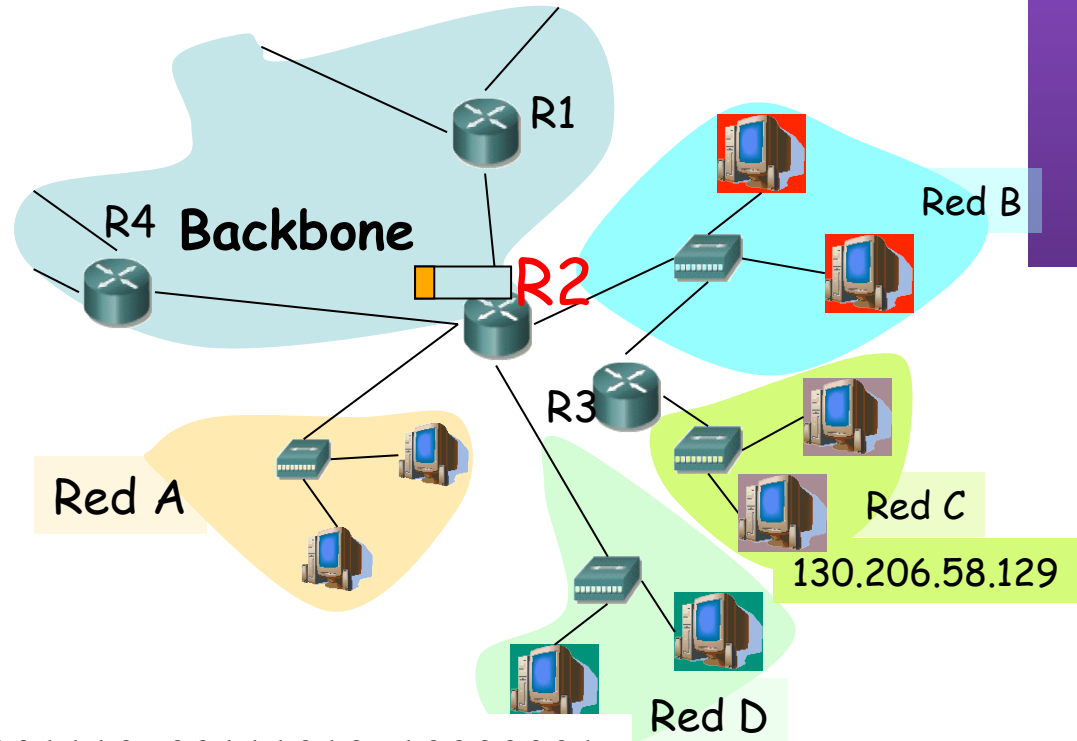
B.1	0X	XXXXXXXXXX
B.2	10	0XXXXXXXXX
B.3	10	1XXXXXXXXX

Libre	11	XXXXXXXXXX
--------------	-----------	-------------------



Ejemplo (3): $IP_d = 130.206.58.129$

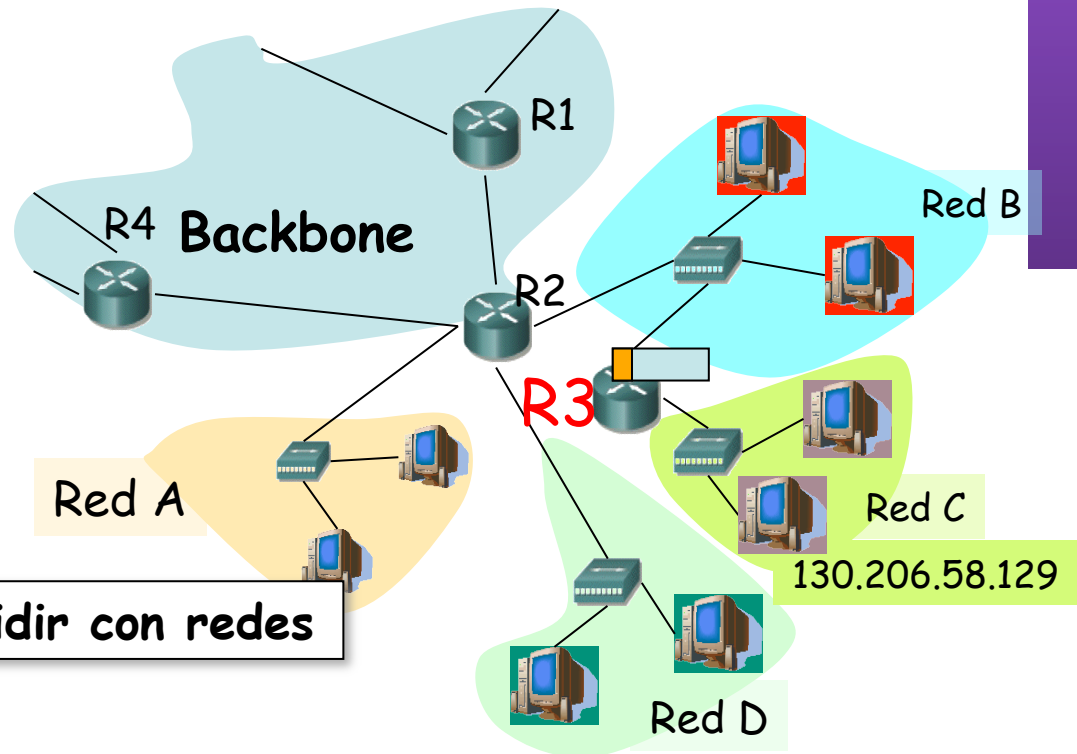
$IP_d =$	Destino	Next-hop	if
10000010 11001110 00111010 10000001	130.206.16.0/20	-	1
10000010 11001110 00010000 00000000	130.206.56.0/21	130.206.16.1 (R3)	1
10000010 11001110 00111000 00000000	130.206.64.0/18	-	2
11001001 00011000 00010000 00000000	201.24.16.0/23	-	3
10000010 00000000 00000000 00000000	201.0.0.0/10	10.50.44.1 (R4)	4
00000000 00000000 00000000 00000000	0.0.0.0/0	10.50.43.13 (R1)	0



$130.206.58.129 = 10000010 \ 11001110 \ 00111010 \ 10000001$

Ejemplo (3): $IP_d = 130.206.58.129$

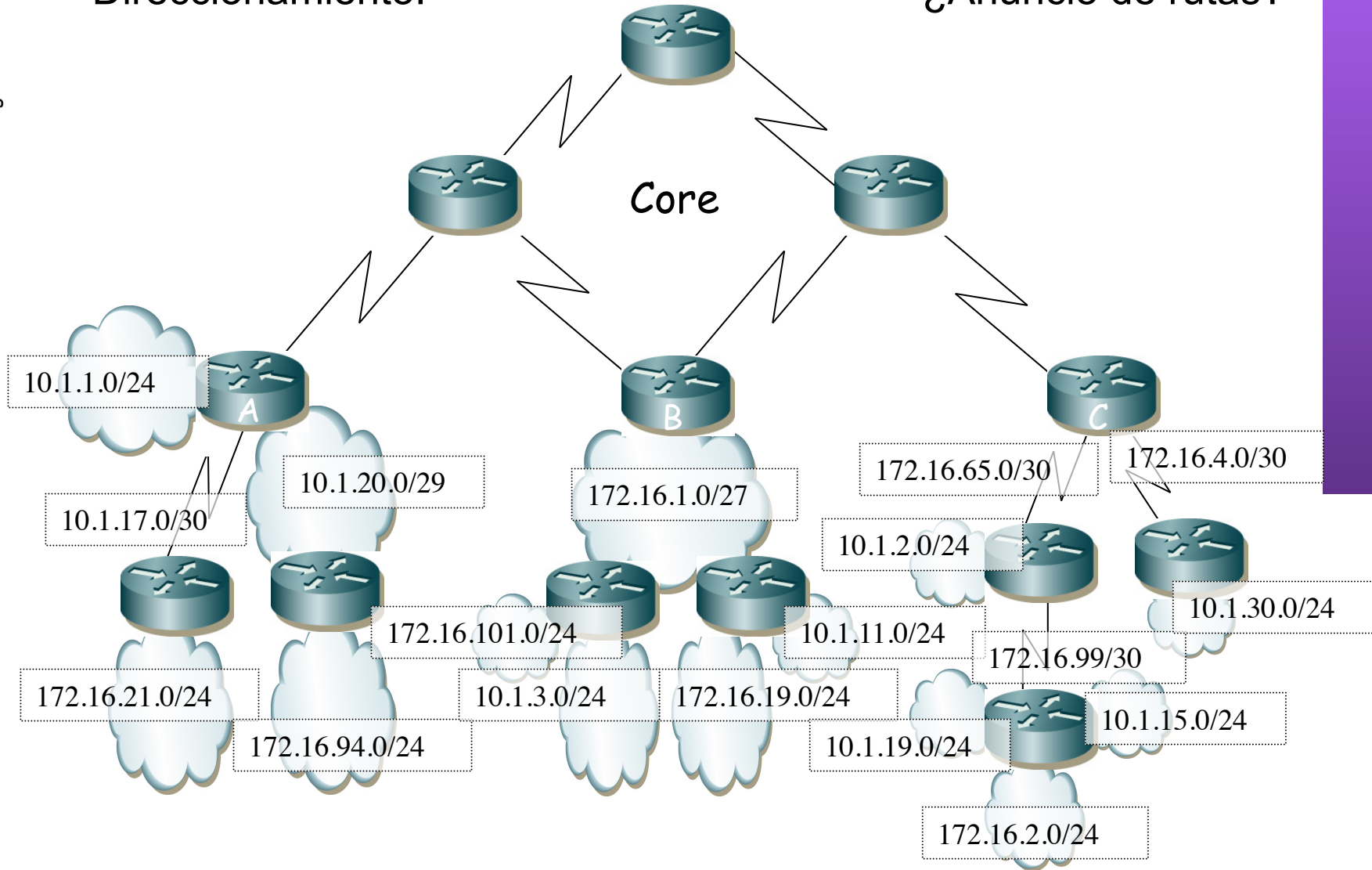
$IP_d =$	Destino	Next-hop	if
10000010 11001110 00111010 10000001	130.206.16.0/20	-	0
10000010 11001110 00010000 00000000	130.206.56.0/21	-	1
00000000 00000000 00000000 00000000	0.0.0.0/0	130.206.16.2 (R2)	0



Las rutas no necesitan coincidir con redes

Ejemplo: Resúmenes

- Direccionamiento:
- ¿Anuncio de rutas?

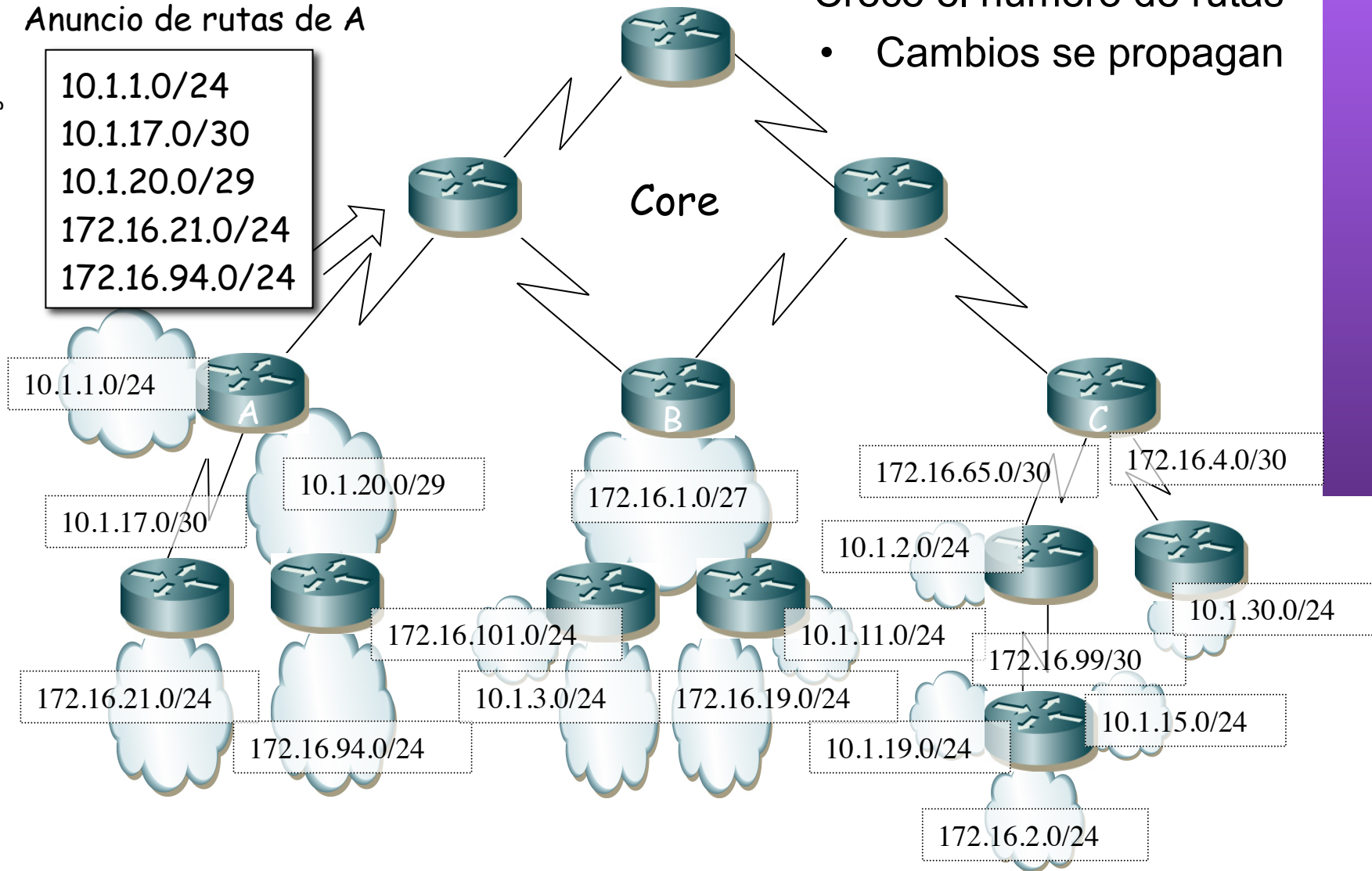


Ejemplo: Resúmenes

- Crece el número de rutas
- Cambios se propagan

Anuncio de rutas de A

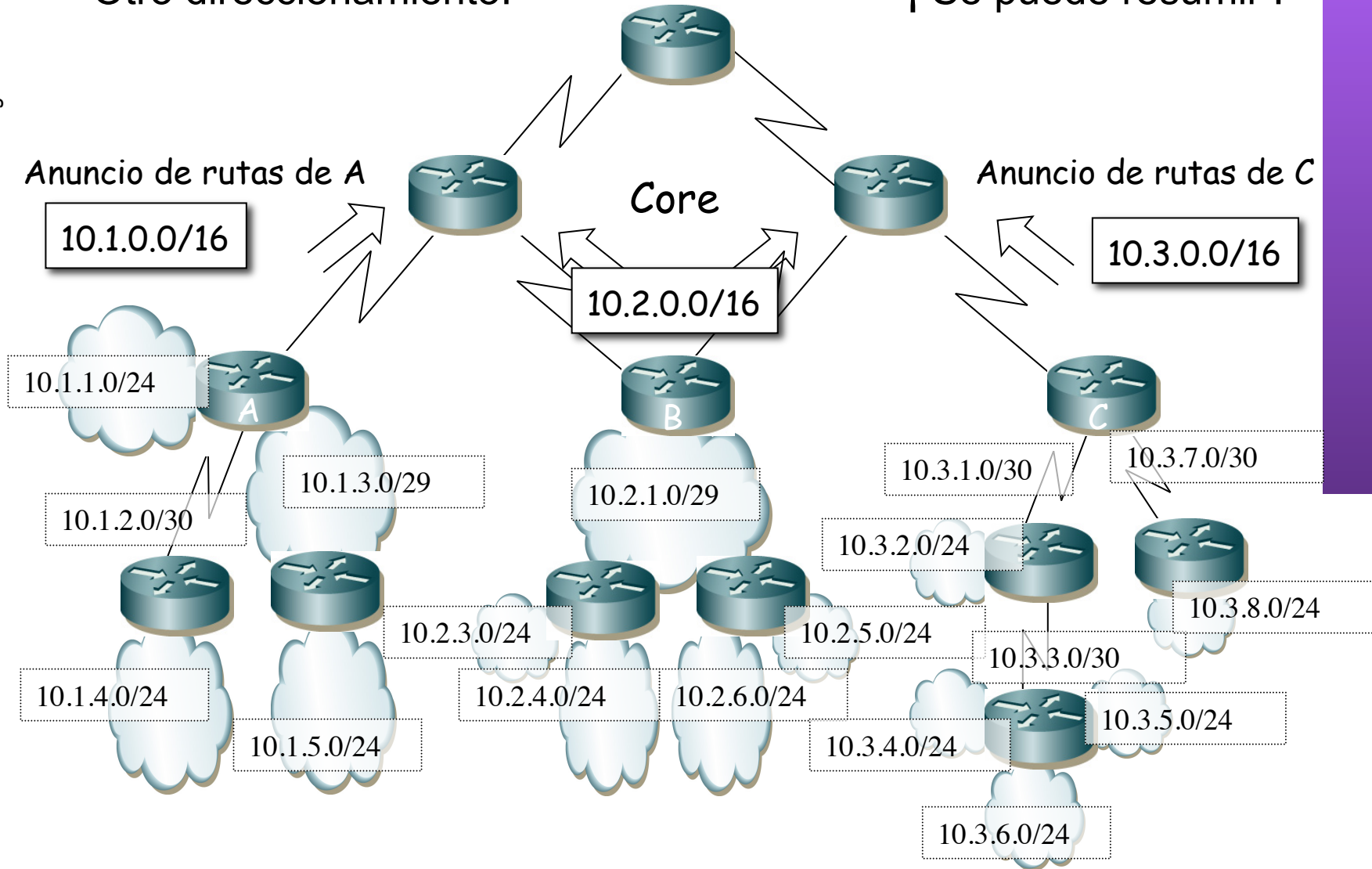
10.1.1.0/24
 10.1.17.0/30
 10.1.20.0/29
 172.16.21.0/24
 172.16.94.0/24



Ejemplo: Resúmenes

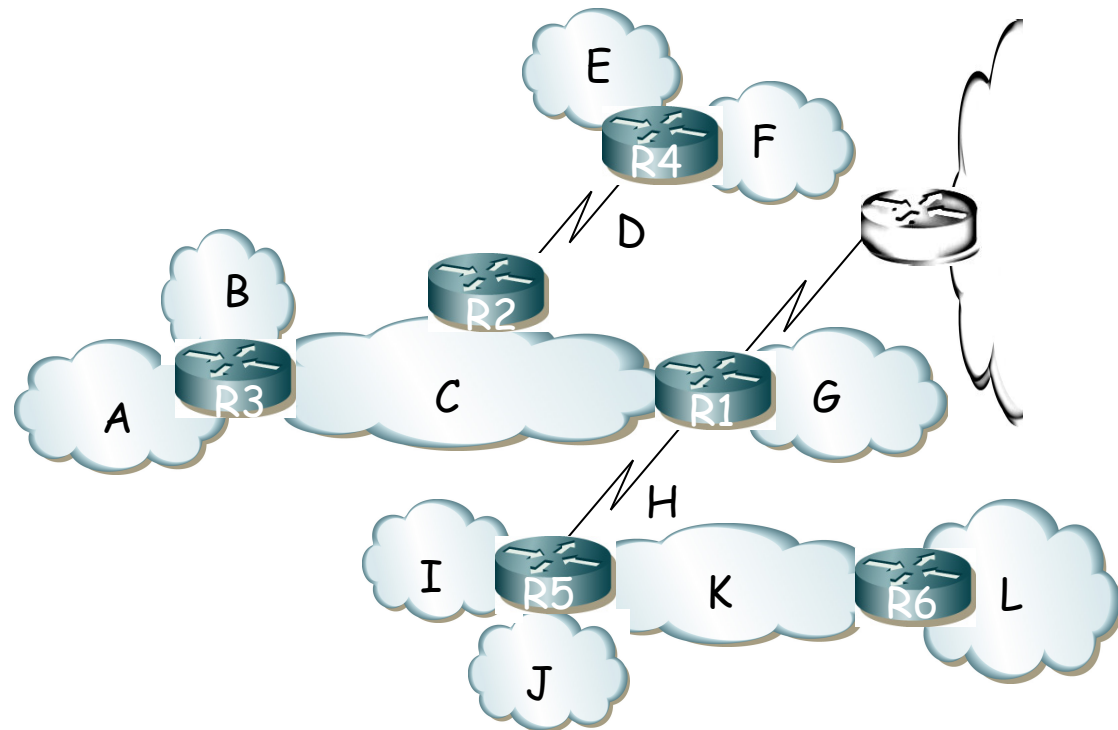
- Otro direccionamiento:

- ¡ Se puede resumir !



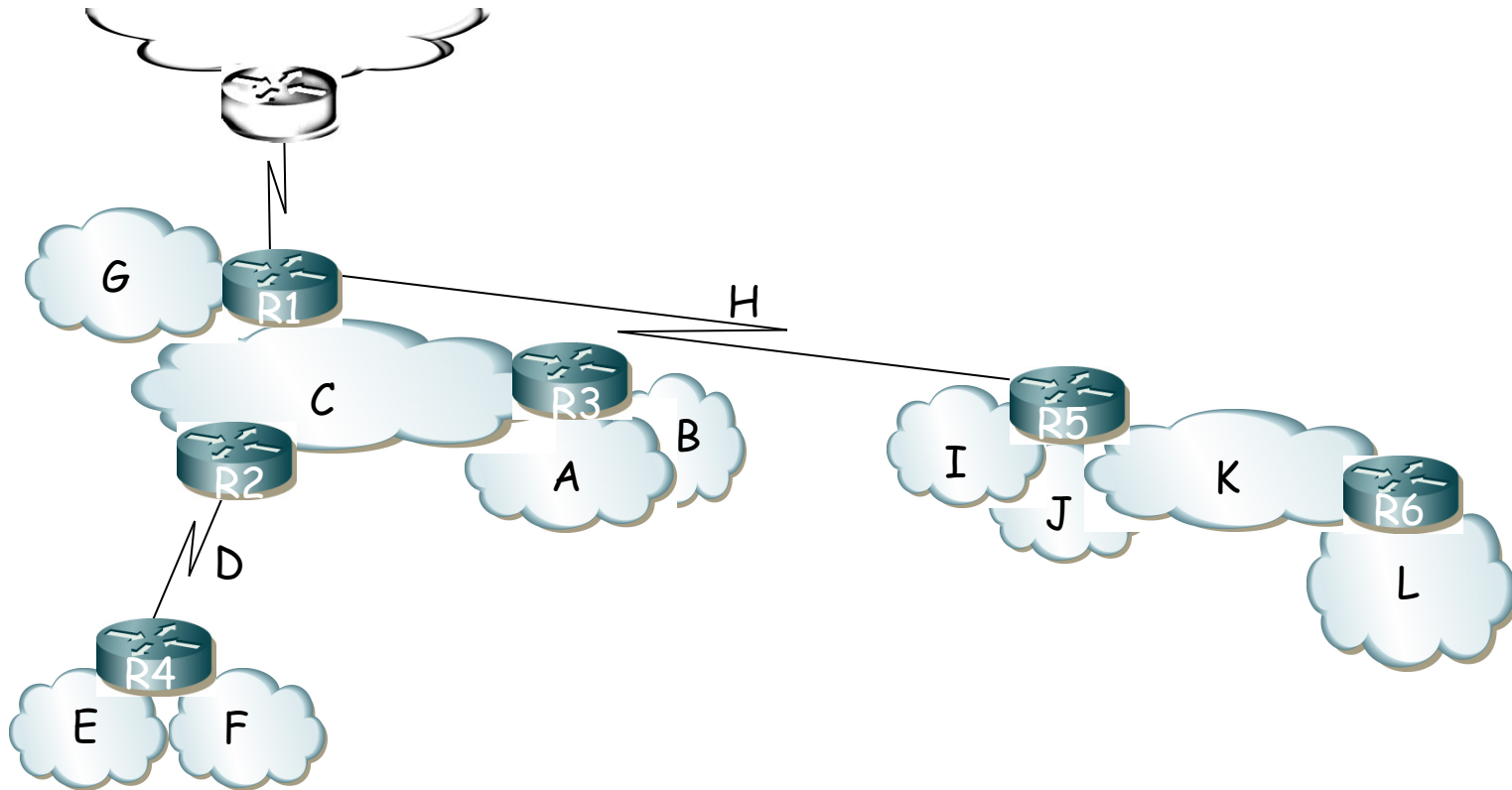
Ejemplo (4)

- 12 Subredes
- Máximo 10 hosts por red
- Red 192.168.3.0/24
- Que se pueda resumir



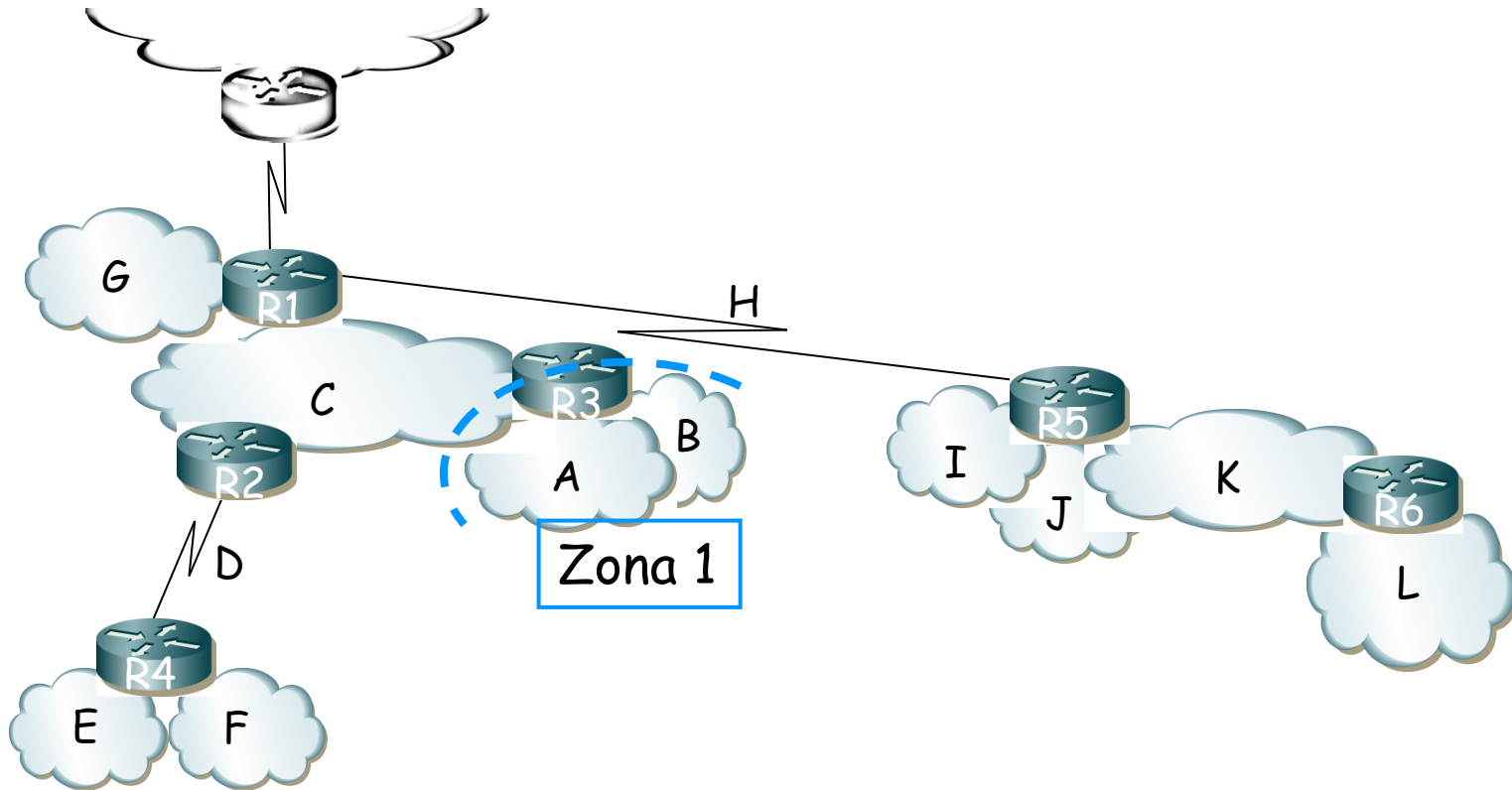
Ejemplo (4)

- Redibujando la topología:



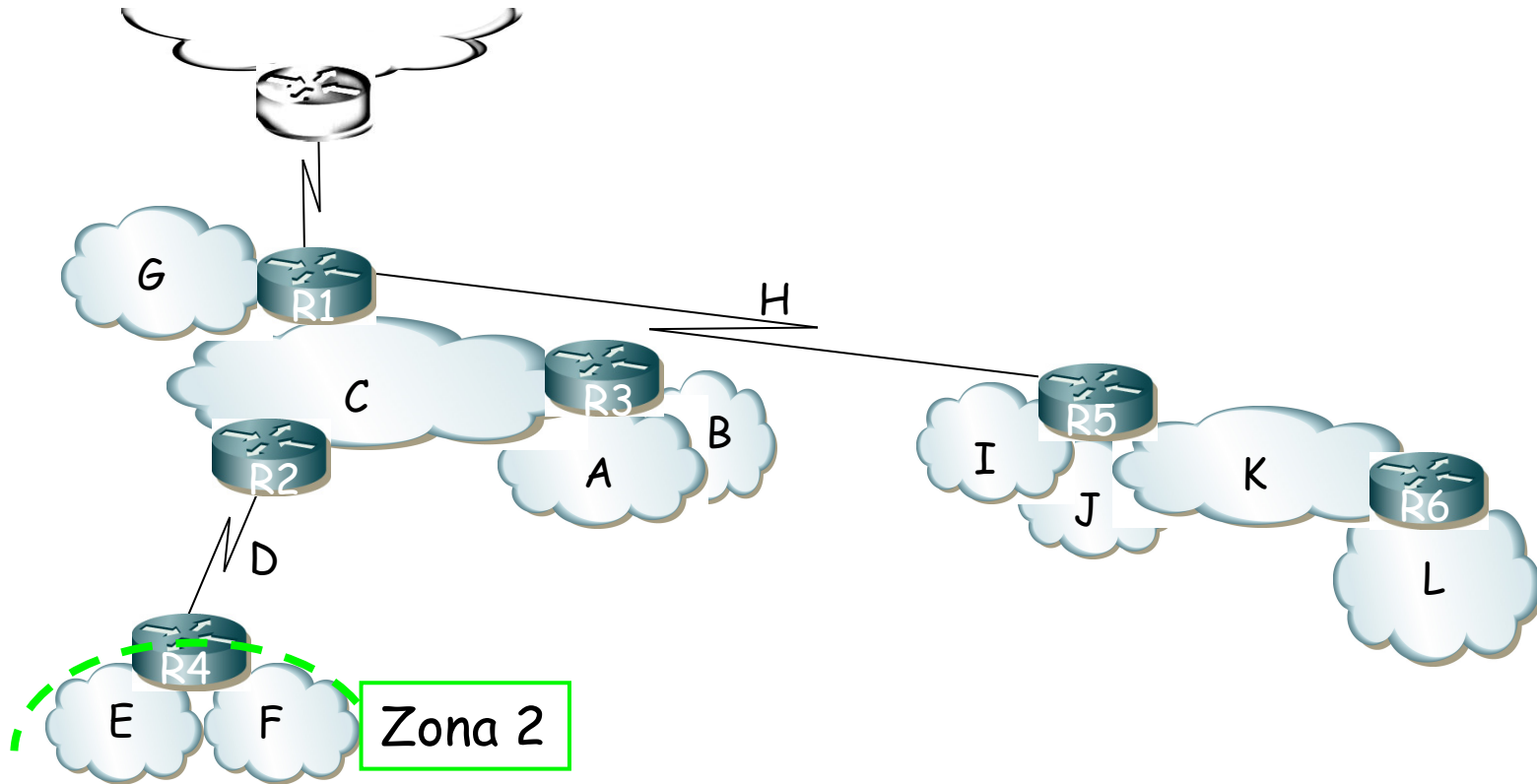
Ejemplo (4)

A	192.168.3	[000]	[0]	[0000]	= 192.168.3.0 /28	} Zona 1: 192.168.3.0 /27
B	192.168.3	[000]	[1]	[0000]	= 192.168.3.16 /28	



Ejemplo (4)

E	192.168.3	[010]	[0]	[0000]	= 192.168.3.64 /28	} Zona 2: 192.168.3.64 /27
F	192.168.3	[010]	[1]	[0000]	= 192.168.3.80 /28	



Ejemplo (4)

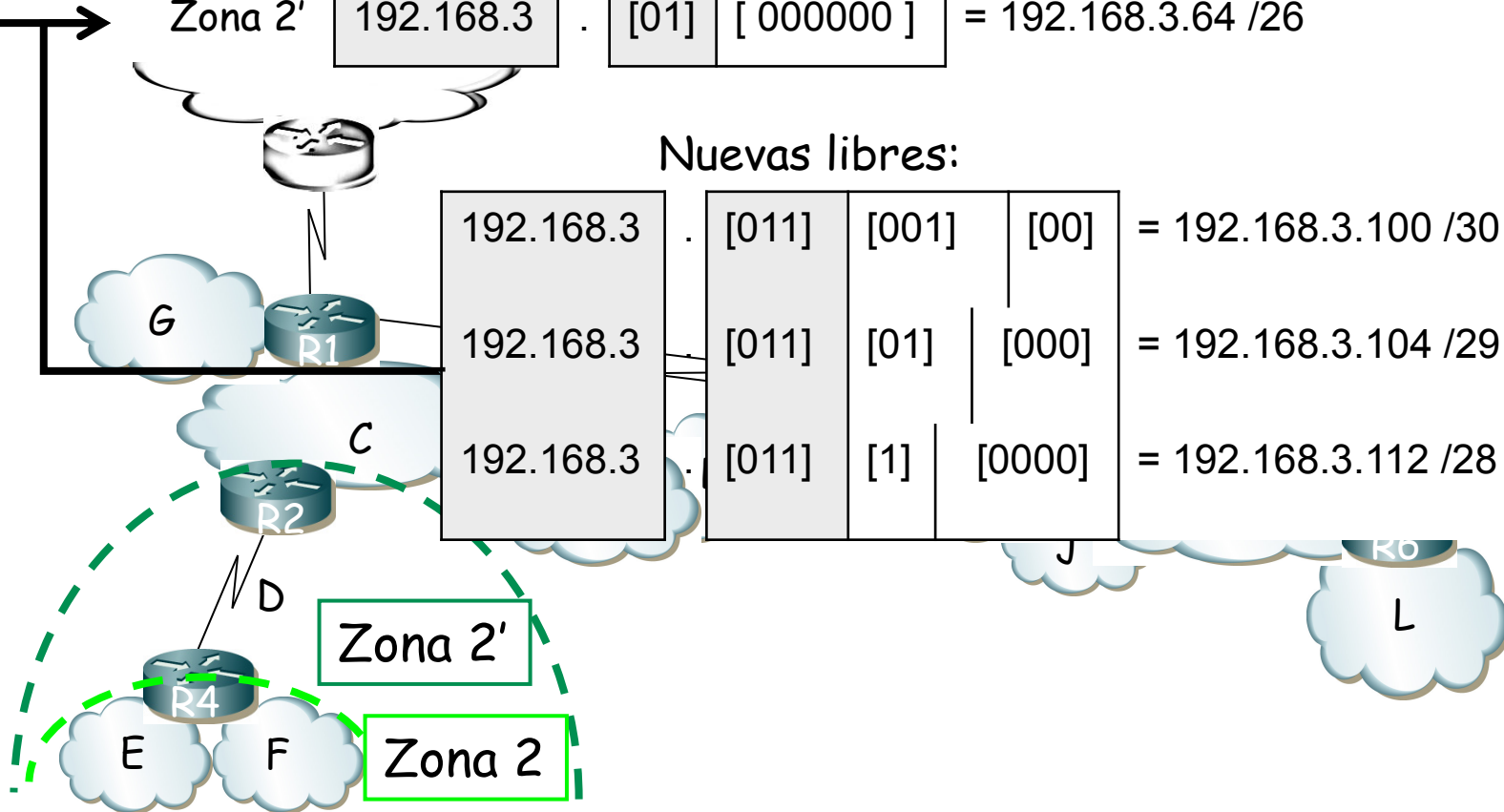
ARQUITECTURA DE REDES,
 SISTEMAS Y SERVICIOS
 Área de Ingeniería Telemática

Zona 2	192.168.3	[010]	[00000]	= 192.168.3.64 /27
	192.168.3	[011]	[000] [00]	= 192.168.3.96 /30

Zona 2'	192.168.3	[01]	[000000]	= 192.168.3.64 /26
---------	-----------	------	------------	--------------------

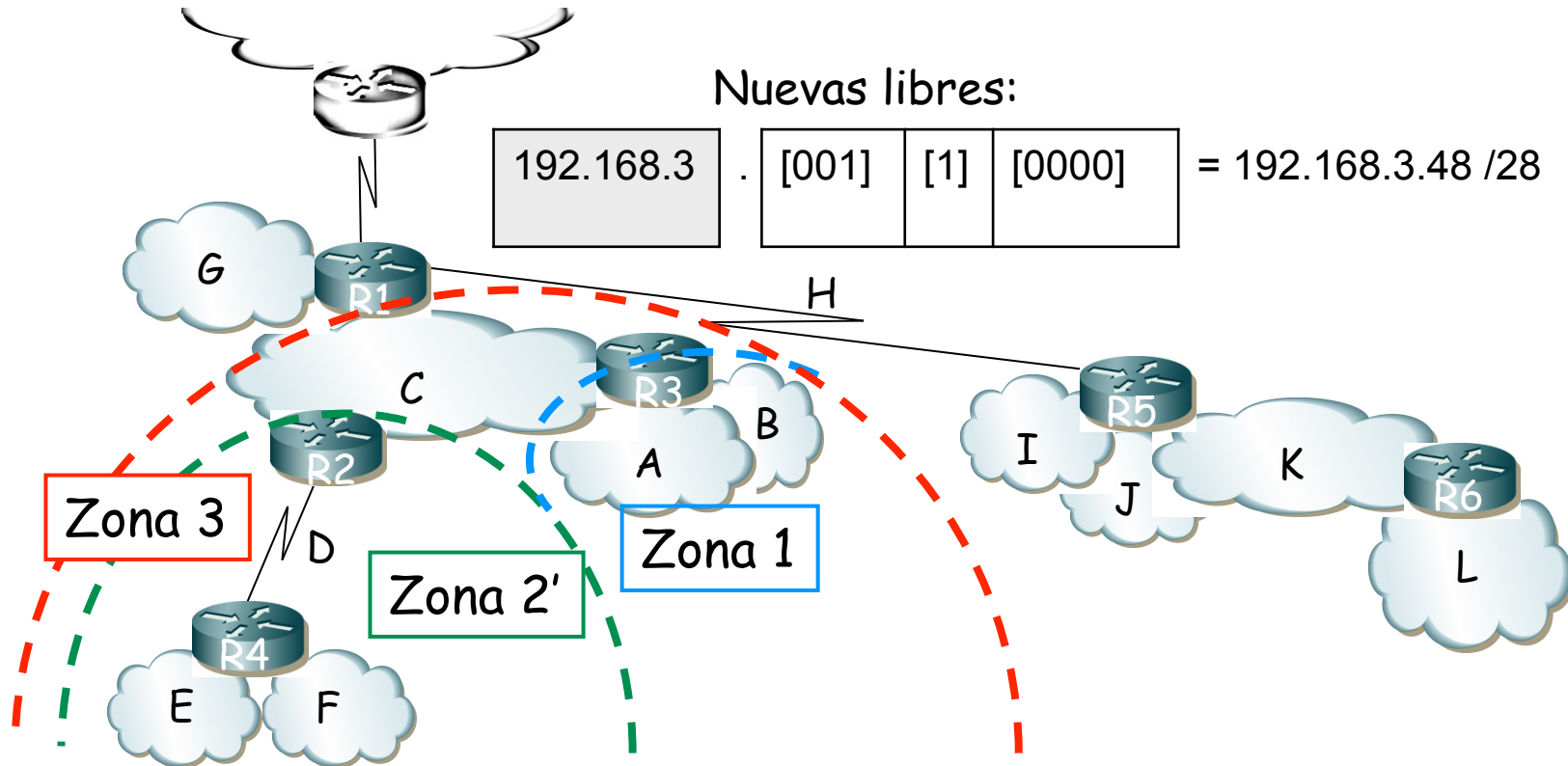
Nuevas libres:

192.168.3	[011]	[001]	[00]	= 192.168.3.100 /30
192.168.3	[011]	[01]	[000]	= 192.168.3.104 /29
192.168.3	[011]	[1]	[0000]	= 192.168.3.112 /28



Ejemplo (4)

Zona 1	192.168.3	[000]	[00000]	= 192.168.3.0 /27
Zona 2'	192.168.3	[01]	[000000]	= 192.168.3.64 /26
C	192.168.3	[001]	[0] [0000]	= 192.168.3.32 /28



Ejemplo (4)

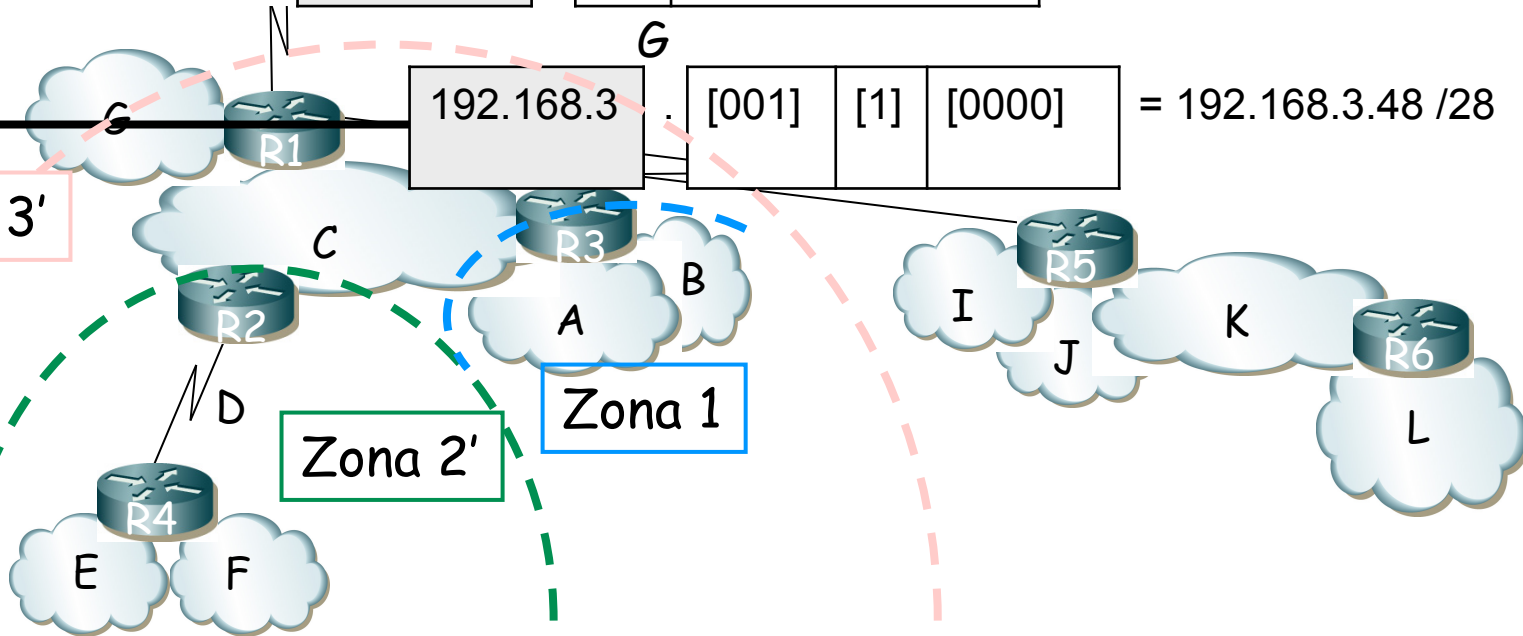
ARQUITECTURA DE REDES,
 SISTEMAS Y SERVICIOS
 Área de Ingeniería Telemática

Zona 1	192.168.3	[000]	[00000]	= 192.168.3.0 /27
Zona 2'	192.168.3	[01]	[000000]	= 192.168.3.64 /26
C	192.168.3	[001]	[0] [0000]	= 192.168.3.32 /28

Zona 3'	192.168.3	[0]	[0000000]	= 192.168.3.0 /25
---------	-----------	-----	-----------	-------------------

192.168.3	[001]	[1]	[0000]	= 192.168.3.48 /28
-----------	-------	-----	--------	--------------------

Zona 3'



Ejemplo (4)

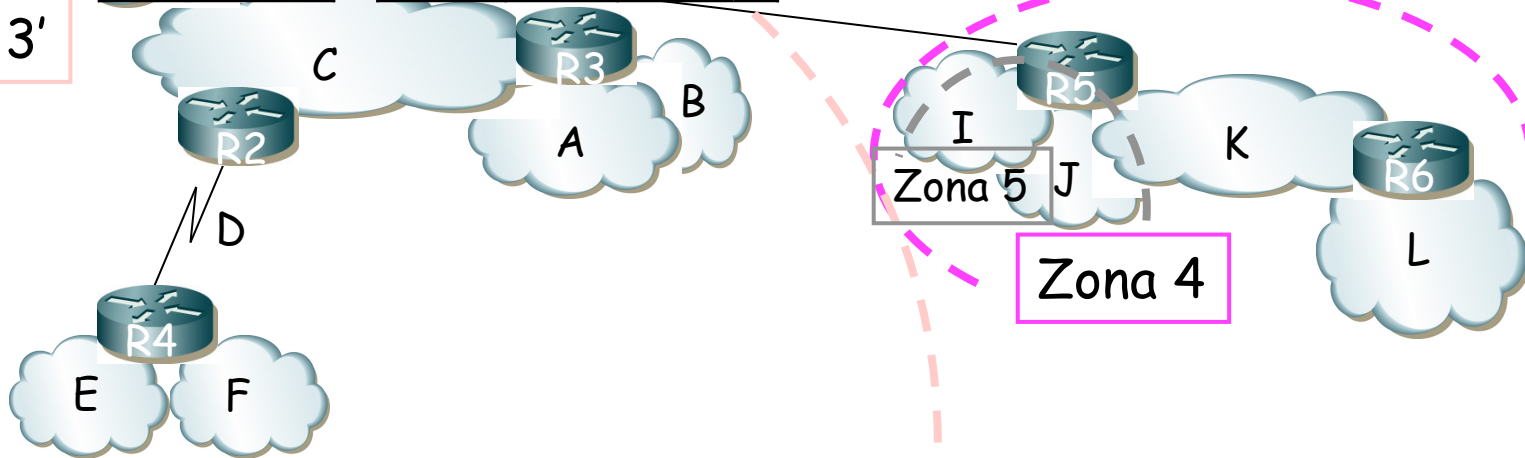
Zona 3'	192.168.3	[0]	[0000000]	= 192.168.3.0 /25	
Zona 4	192.168.3	[10]	[000000]	= 192.168.3.128 /26	
I	192.168.3	[10]	[00]	[0000]	= 192.168.3.128 /28
J	192.168.3	[10]	[01]	[0000]	= 192.168.3.136 /28
K	192.168.3	[10]	[10]	[0000]	= 192.168.3.144 /28
L	192.168.3	[10]	[11]	[0000]	= 192.168.3.152 /28

Zona 5:
 192.168.3.128 /27

Zona 3'

Zona 5

Zona 4



Ejemplo (4)

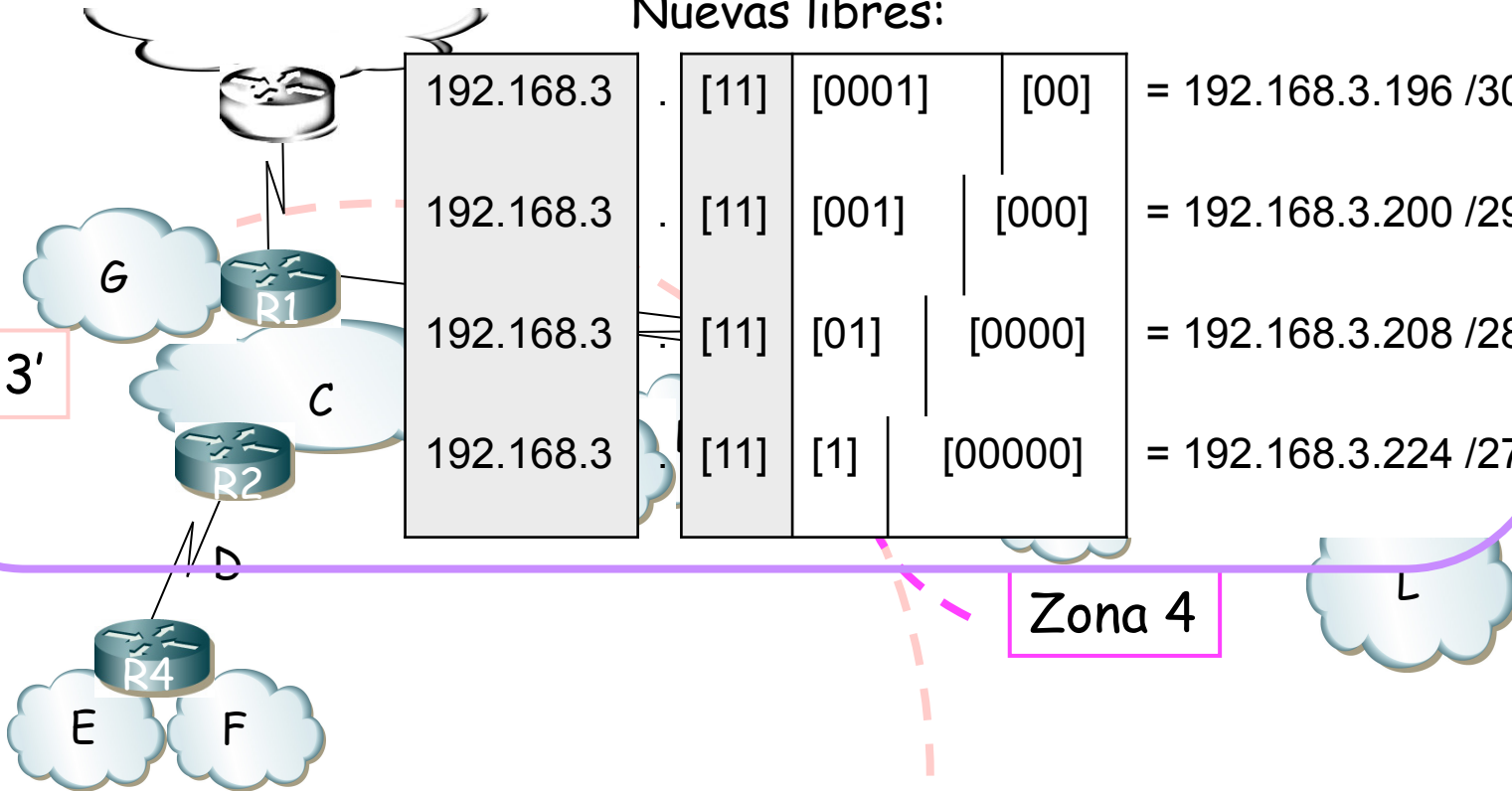
Zona 3'	192.168.3	[0]	[0000000]	= 192.168.3.0 /25
Zona 4	192.168.3	[10]	[000000]	= 192.168.3.128 /26
H	192.168.3	[11]	[0000] [00]	= 192.168.3.192 /30

Nuevas libres:

192.168.3	[11]	[0001]	[00]	= 192.168.3.196 /30
192.168.3	[11]	[001]	[000]	= 192.168.3.200 /29
192.168.3	[11]	[01]	[0000]	= 192.168.3.208 /28
192.168.3	[11]	[1]	[00000]	= 192.168.3.224 /27

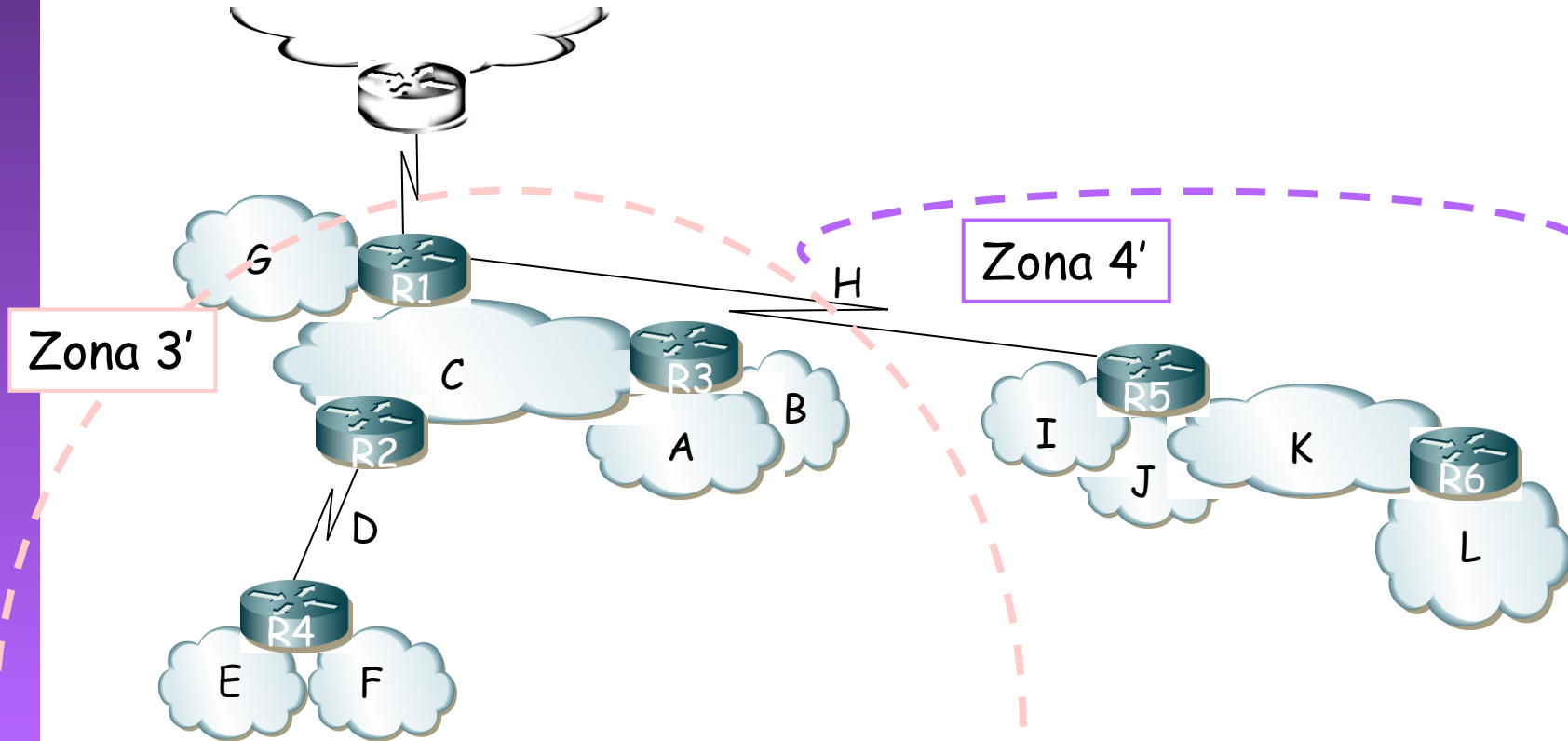
Zona 3'

Zona 4



Ejemplo (4)

Zona 3'	192.168.3	[0]	[0000000]	= 192.168.3.0 /25
Zona 4'	192.168.3	[1]	[0000000]	= 192.168.3.128 /25

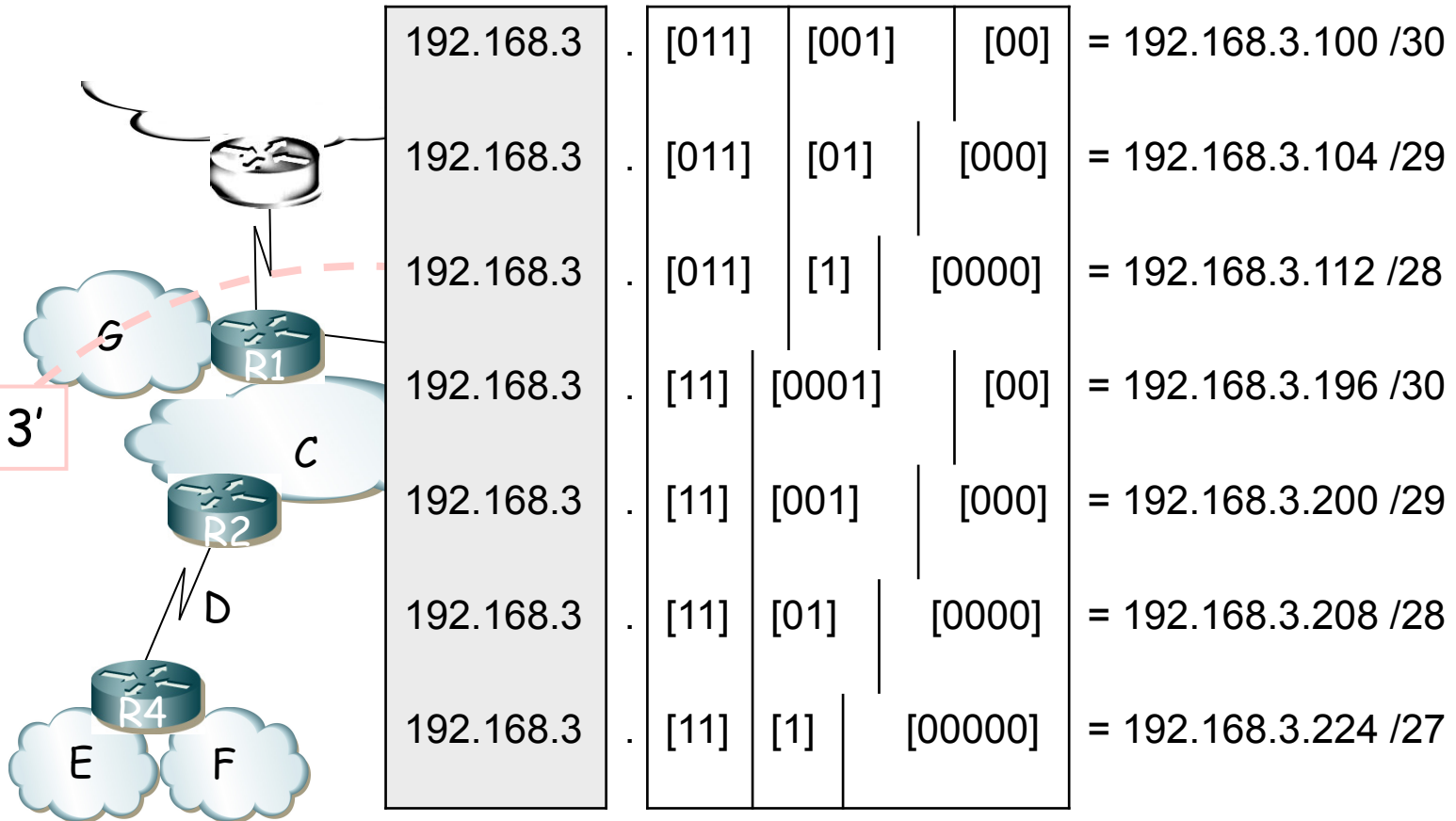


Ejemplo (4)

Zona 3'	192.168.3	[0]	[0000000]	= 192.168.3.0 /25
Zona 4'	192.168.3	[1]	[0000000]	= 192.168.3.128 /25

Libres totales:

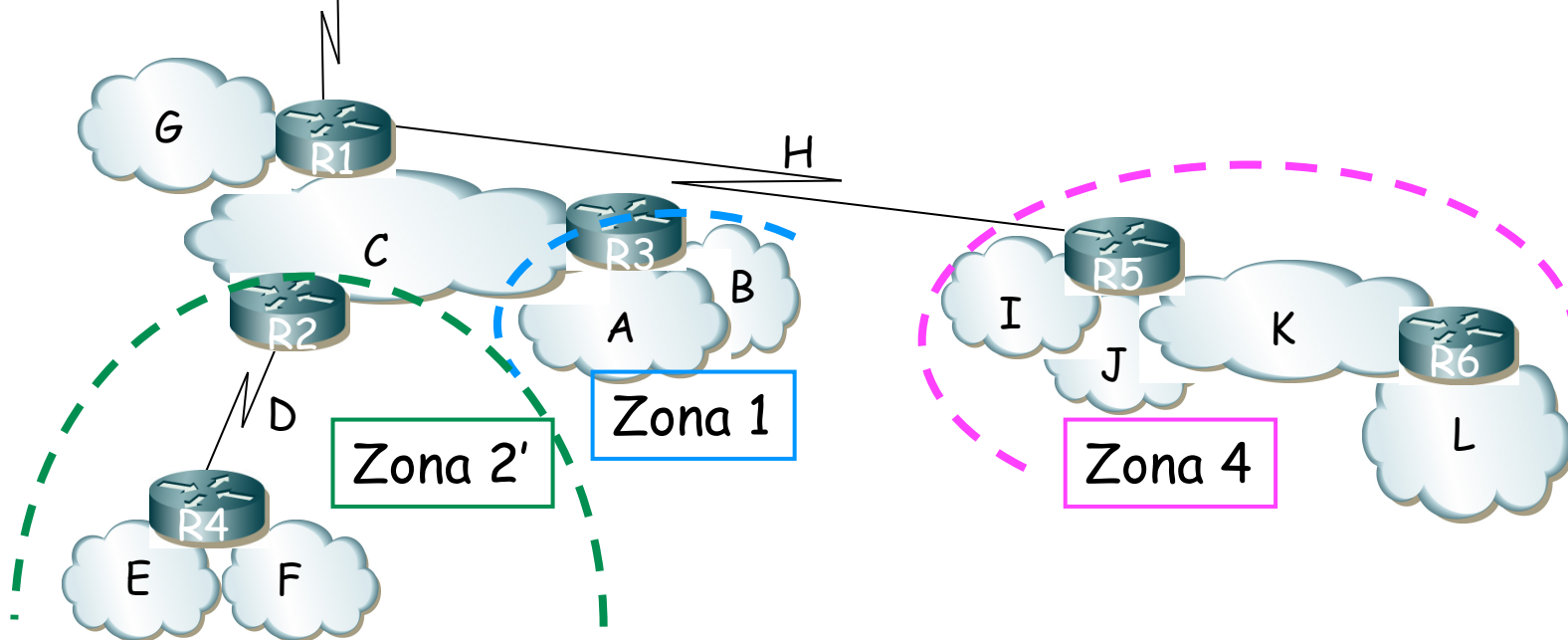
Zona 3'



Ejemplo (4)

Tabla de rutas de R1:

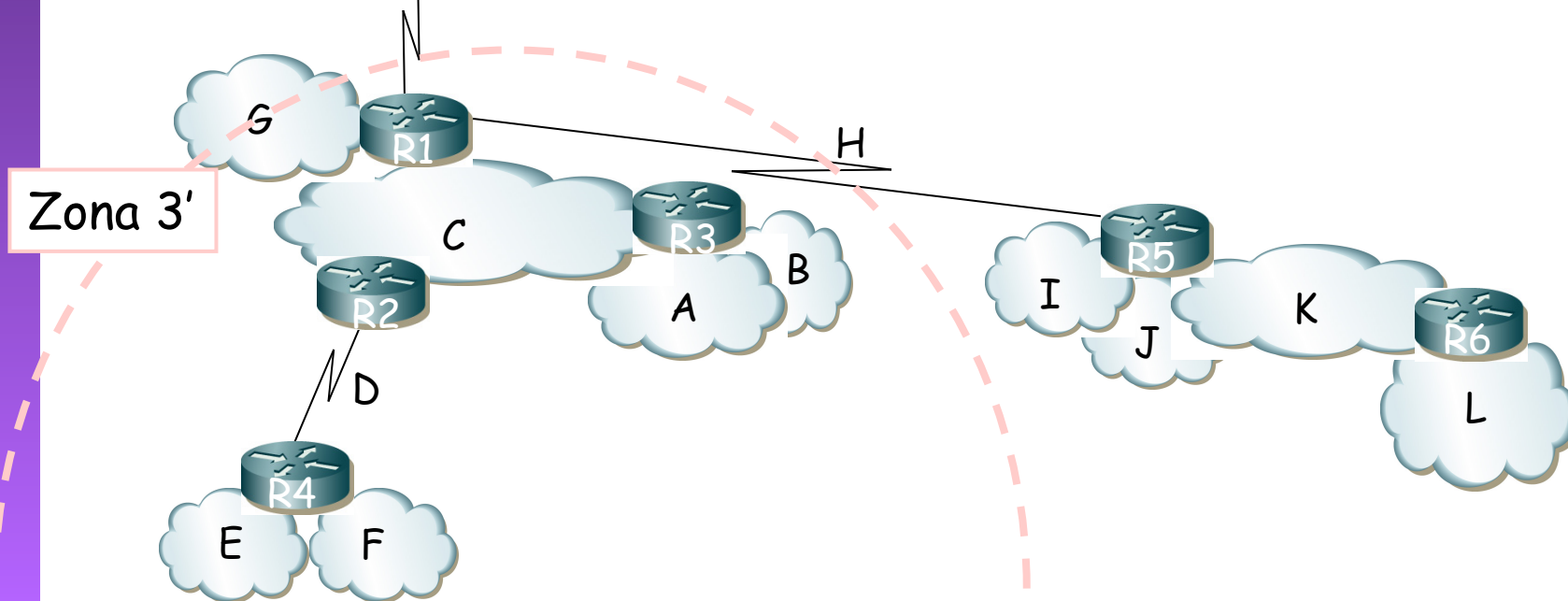
Destino	Next-hop	Interfaz
Red C	192.168.3.32 /28	(dir.connected)
Red G	192.168.3.48 /28	(dir.connected)
Red H	192.168.3.192 /30	(dir.connected)
Zona 4	192.168.3.128 /26	IPR5ifRedH
Zona 2'	192.168.3.64 /26	IPR2ifRedC
Zona 1	192.168.3.0 /27	IPR3ifRedC



Ejemplo (4)

Tabla de rutas de R5:

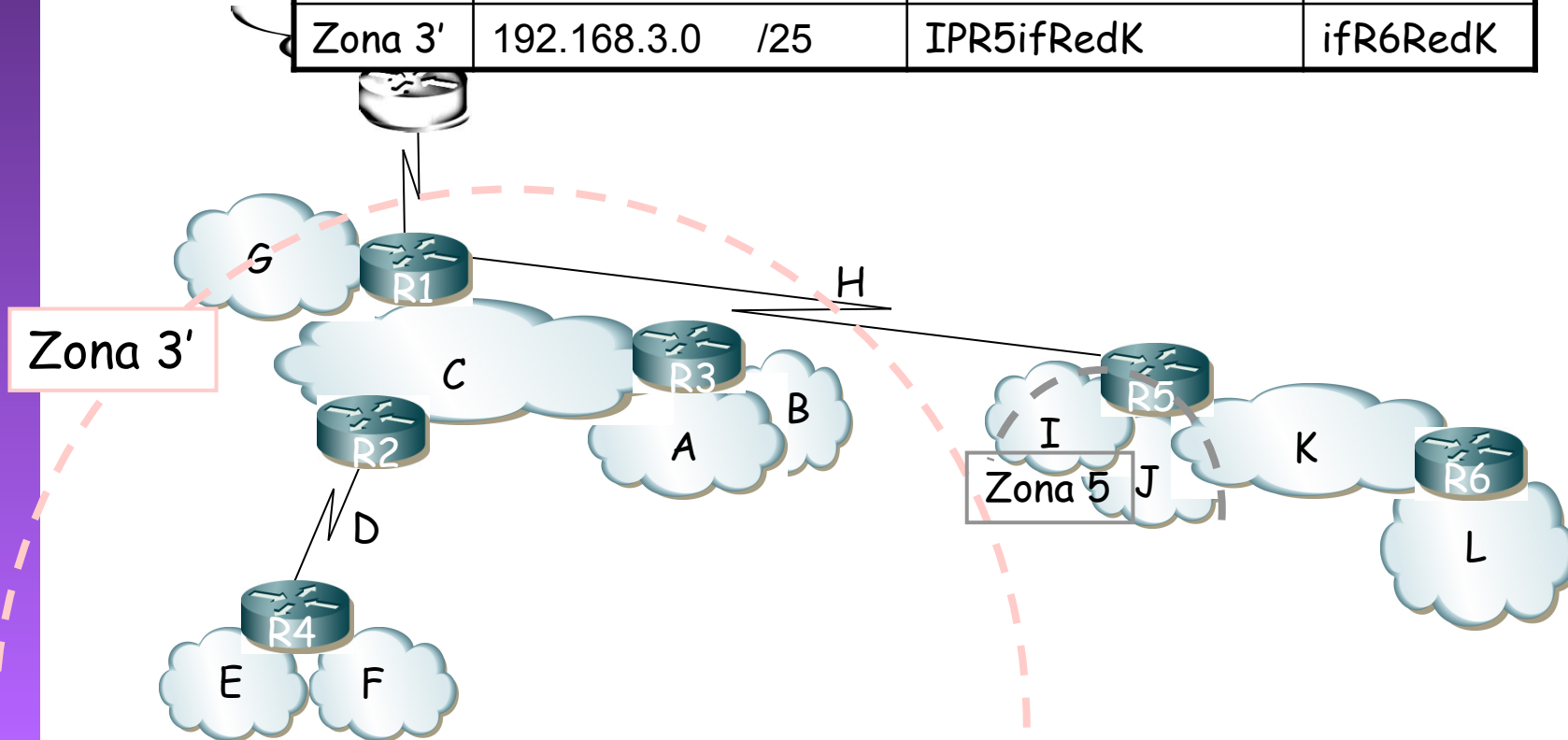
Destino		Next-hop	Interfaz
Red I	192.168.3.128 /28	(dir.connected)	ifR5RedI
Red J	192.168.3.136 /28	(dir.connected)	ifR5RedJ
Red K	192.168.3.136 /28	(dir.connected)	ifR5RedK
Red H	192.168.3.192 /30	(dir.connected)	ifR5RedH
Red L	192.168.3.152 /28	IPR6ifRedK	ifR5RedK
Zona 3'	192.168.3.0 /25	IPR1ifRedH	ifR5RedH



Ejemplo (4)

Tabla de rutas de R6:

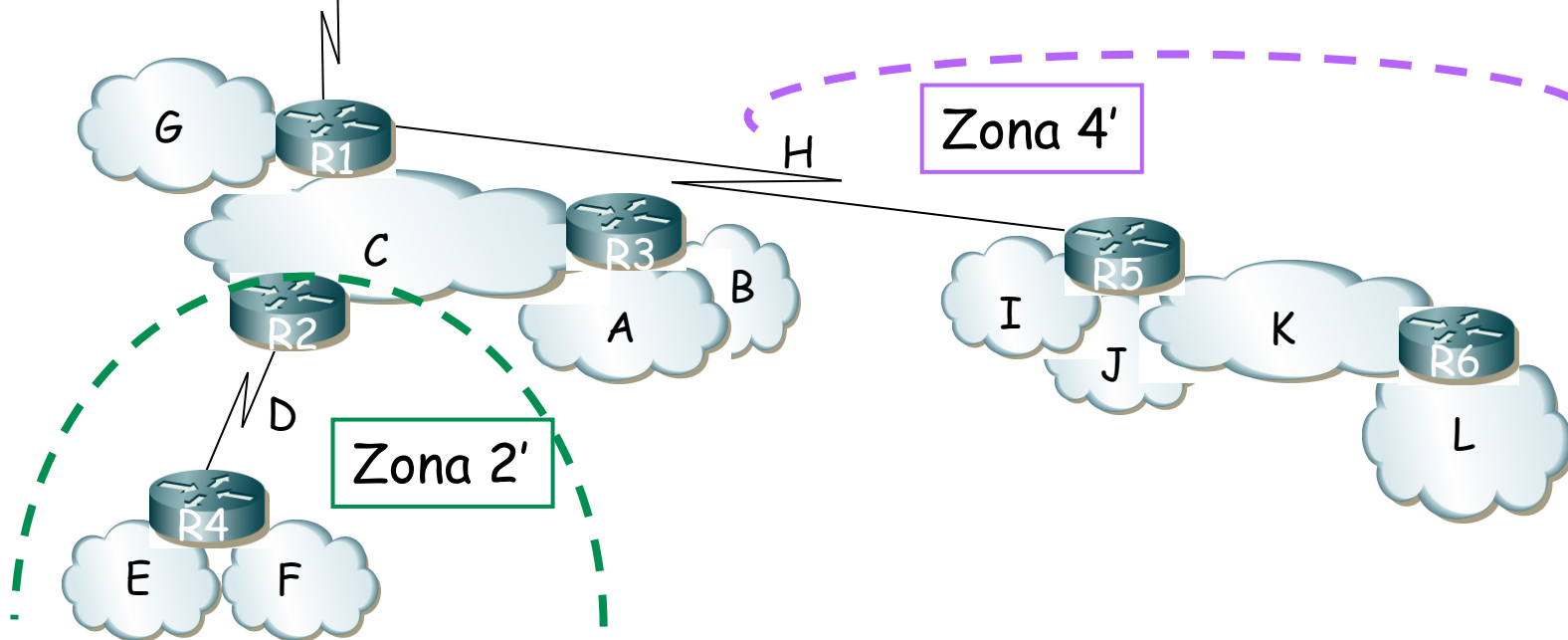
Destino	Next-hop	Interfaz
Red K	192.168.3.136 /28	(dir.connected)
Red L	192.168.3.152 /28	(dir.connected)
Zona 5	192.168.3.128 /27	IPR5ifRedK
Red H	192.168.3.192 /30	IPR5ifRedK
Zona 3'	192.168.3.0 /25	IPR5ifRedK



Ejemplo (4)

Tabla de rutas de R3:

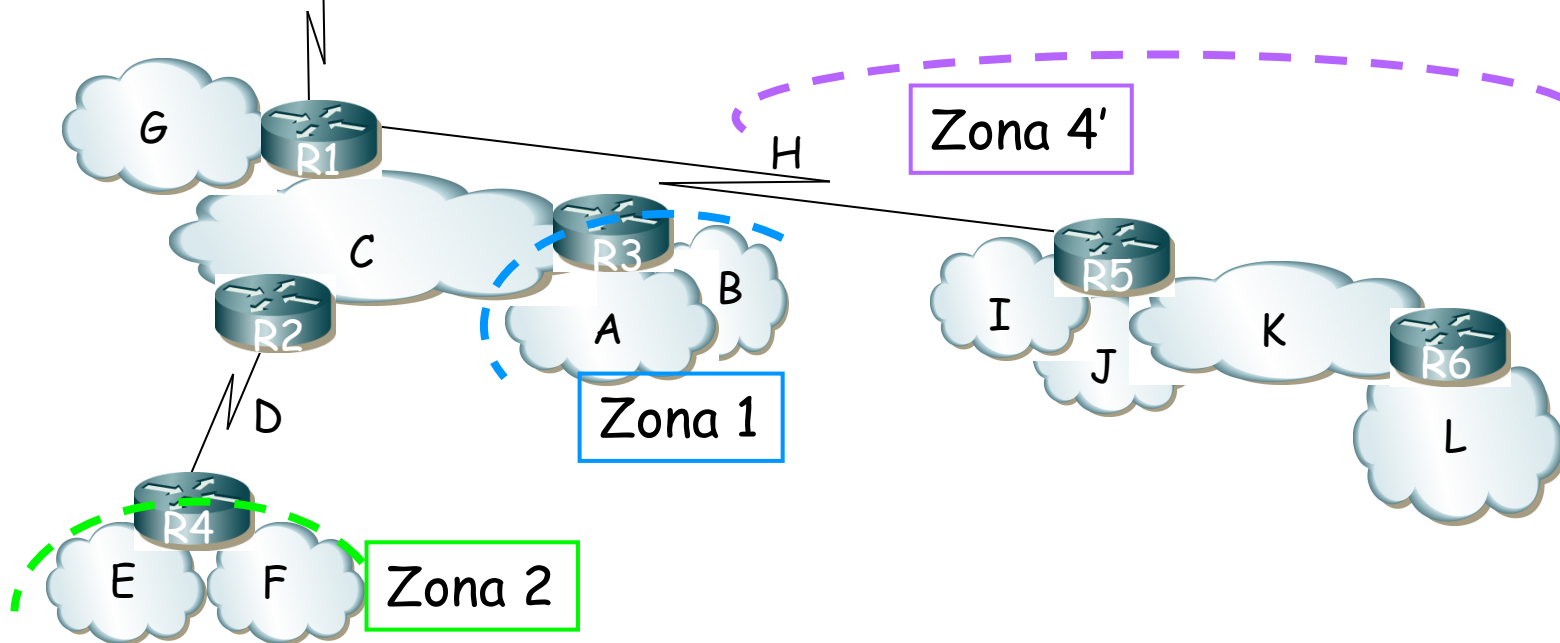
Destino	Next-hop	Interfaz
Red A	192.168.3.0 /28	(dir.connected)
Red B	192.168.3.16 /28	(dir.connected)
Red C	192.168.3.32 /28	(dir.connected)
Zona 2'	192.168.3.64 /26	IPR2ifRedC
Red G	192.168.3.48 /28	IPR1ifRedC
Zona 4'	192.168.3.128 /25	IPR1ifRedC



Ejemplo (4)

Tabla de rutas de R2:

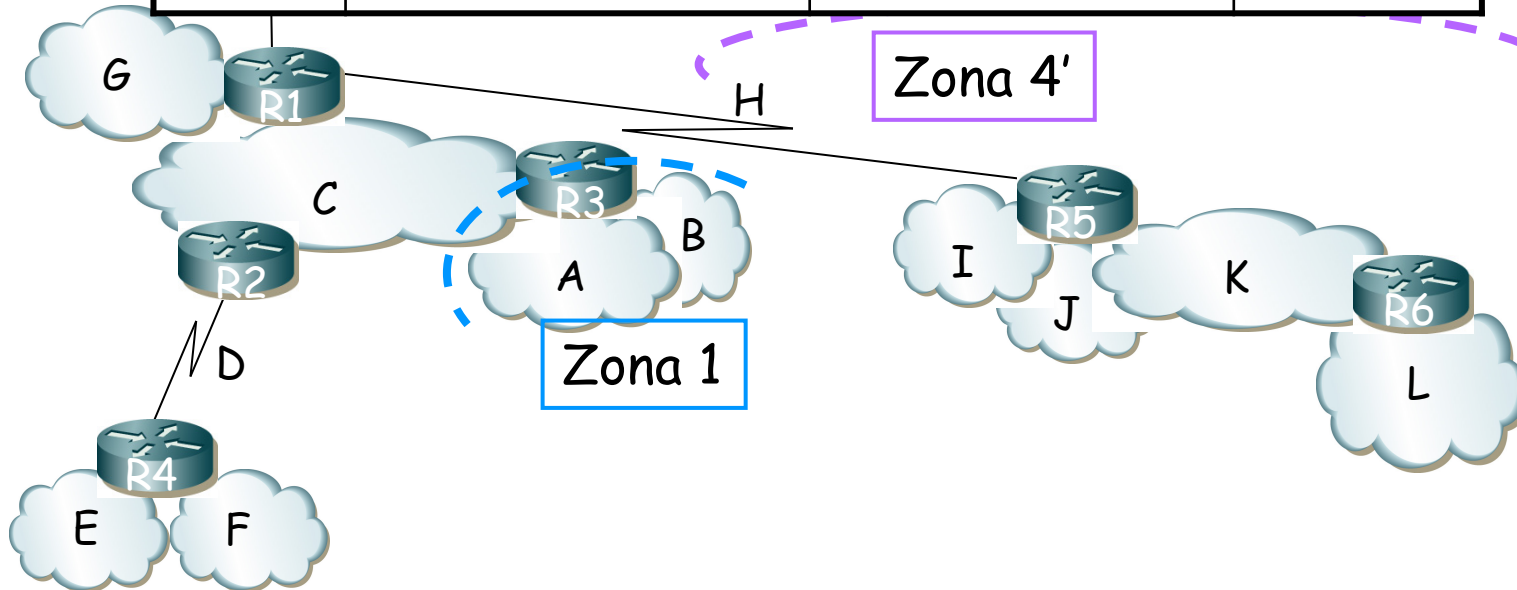
Destino	Next-hop	Interfaz
Red D	192.168.3.96 /30	(dir.connected)
Red C	192.168.3.32 /28	(dir.connected)
Zona 2	192.168.3.64 /27	IPR4ifRedD
Zona 1	192.168.3.0 /27	IPR3ifRedC
Red G	192.168.3.48 /28	IPR1ifRedC
Zona 4'	192.168.3.128 /25	IPR1ifRedC



Ejemplo (4)

Tabla de rutas de R4:

Destino	Next-hop	Interfaz
Red D	192.168.3.96 /30	(dir.connected)
Red E	192.168.3.64 /28	(dir.connected)
Red F	192.168.3.80 /28	(dir.connected)
Red C	192.168.3.32 /28	IPR2ifRedD
Zona 1	192.168.3.0 /27	IPR2ifRedD
Red G	192.168.3.48 /28	IPR2ifRedD
Zona 4'	192.168.3.128 /25	IPR2ifRedD

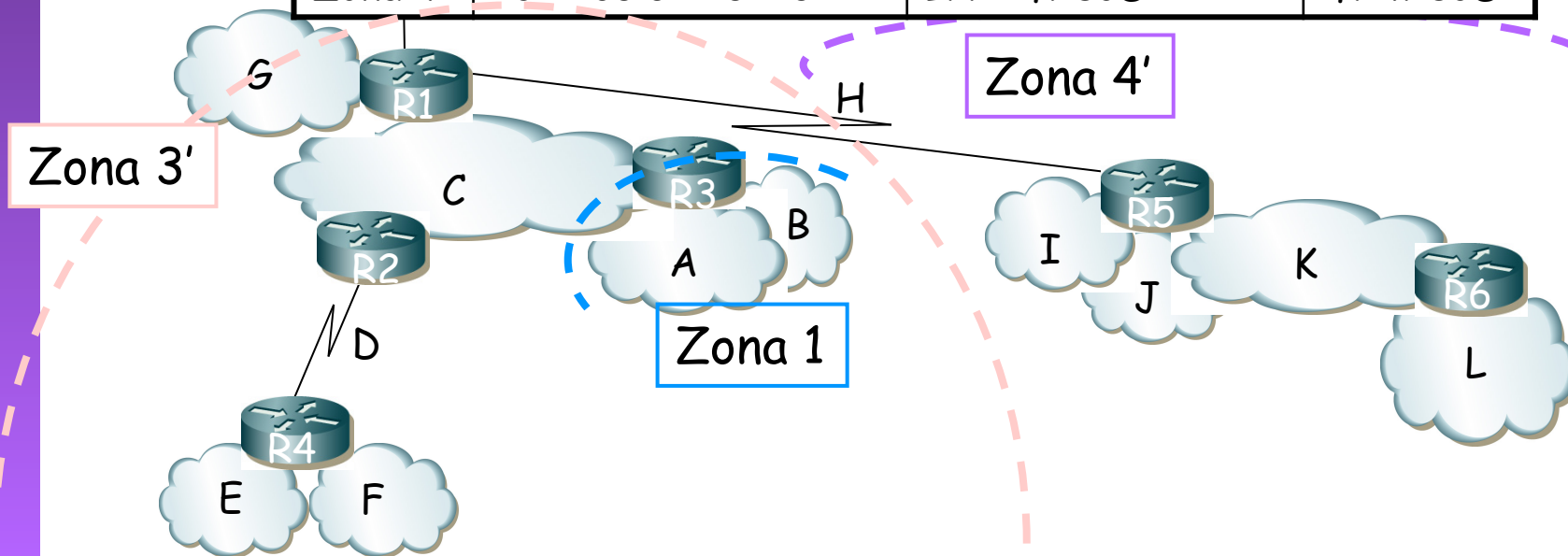


Ejemplo (4)

Mejora 1

Tabla de rutas de R4:

Destino	Next-hop	Interfaz
Red D	192.168.3.96 /30	(dir.connected)
Red E	192.168.3.64 /28	(dir.connected)
Red F	192.168.3.80 /28	(dir.connected)
Red C	192.168.3.32 /28	IPR2ifRedD
Zona 1	192.168.3.0 /27	IPR2ifRedD
Red G	192.168.3.48 /28	IPR2ifRedD
Zona 4'	192.168.3.128 /25	IPR2ifRedD

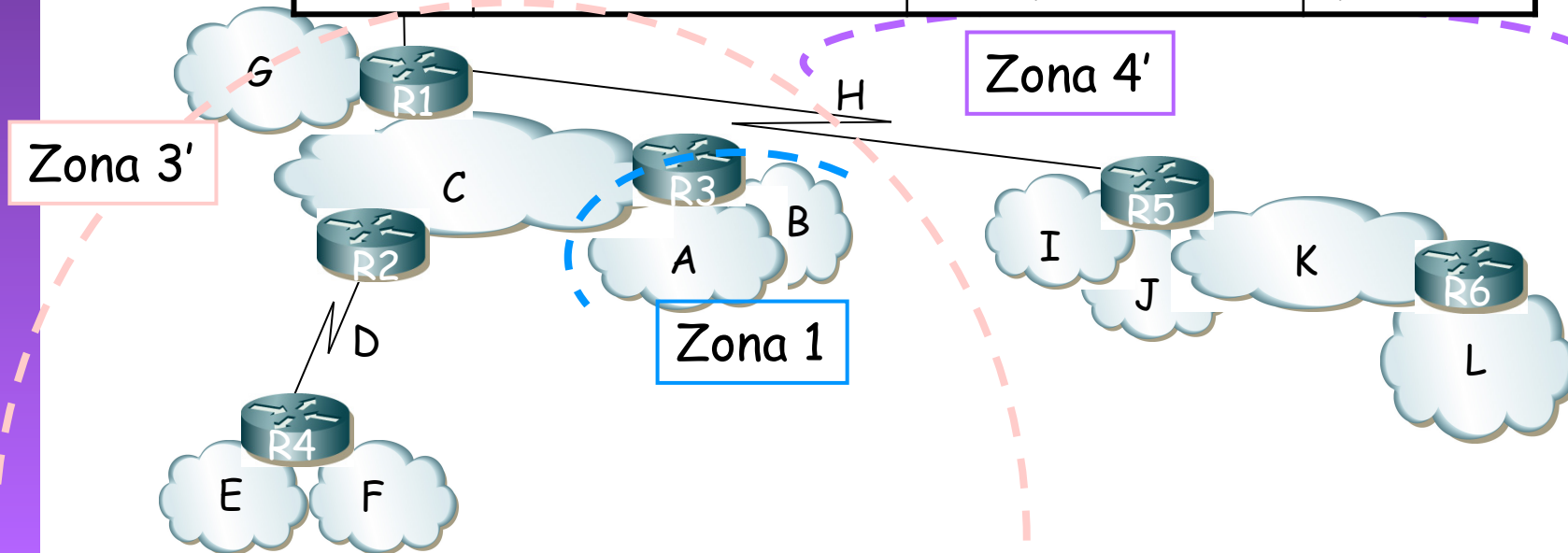


Ejemplo (4)

Mejora 1

Tabla de rutas de R4:

Destino	Next-hop	Interfaz	
Red D	192.168.3.96 /30	(dir.connected)	ifR4RedD
Red E	192.168.3.64 /28	(dir.connected)	ifR4RedE
Red F	192.168.3.80 /28	(dir.connected)	ifR4RedF
Zona 3'	192.168.3.0 /25	IPR2ifRedD	ifR4RedD
Zona 4'	192.168.3.128 /25	IPR2ifRedD	ifR4RedD

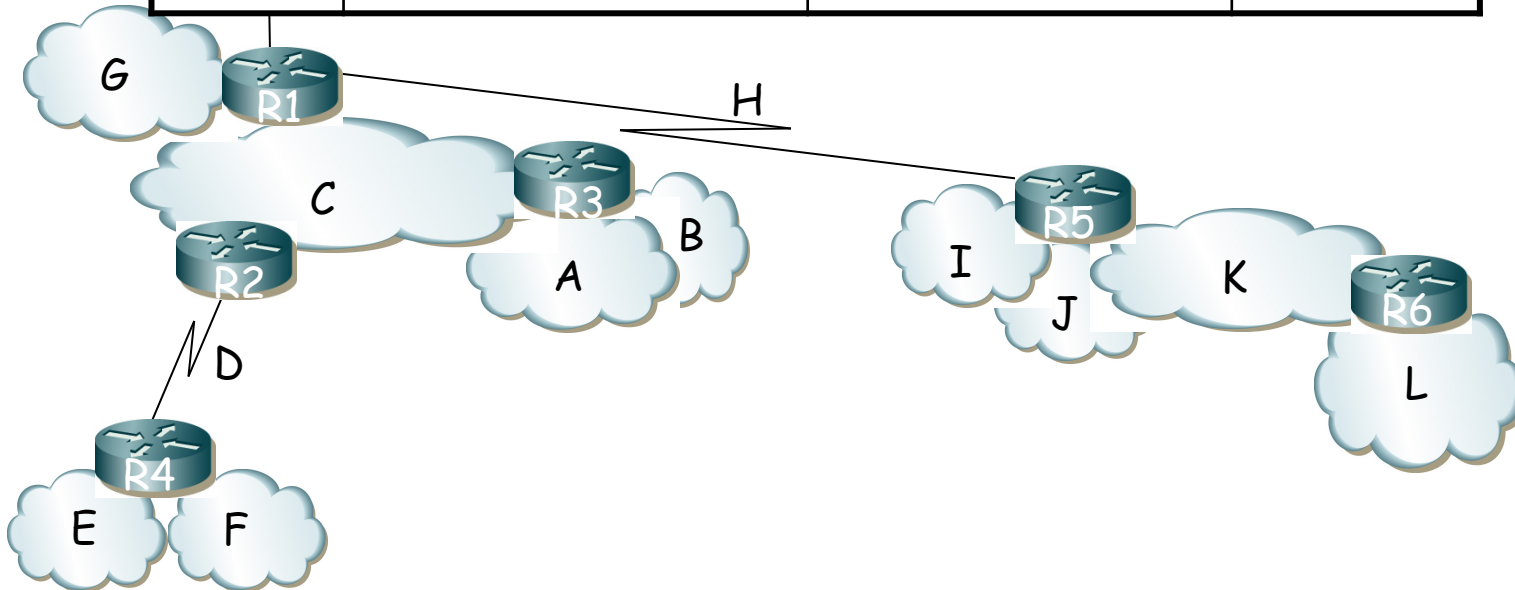


Ejemplo (4)

Mejora 2

Tabla de rutas de R4:

Destino	Next-hop	Interfaz
Red D	192.168.3.96 /30	(dir.connected)
Red E	192.168.3.64 /28	(dir.connected)
Red F	192.168.3.80 /28	(dir.connected)
Red	192.168.3.0 /24	IPR2ifRedD

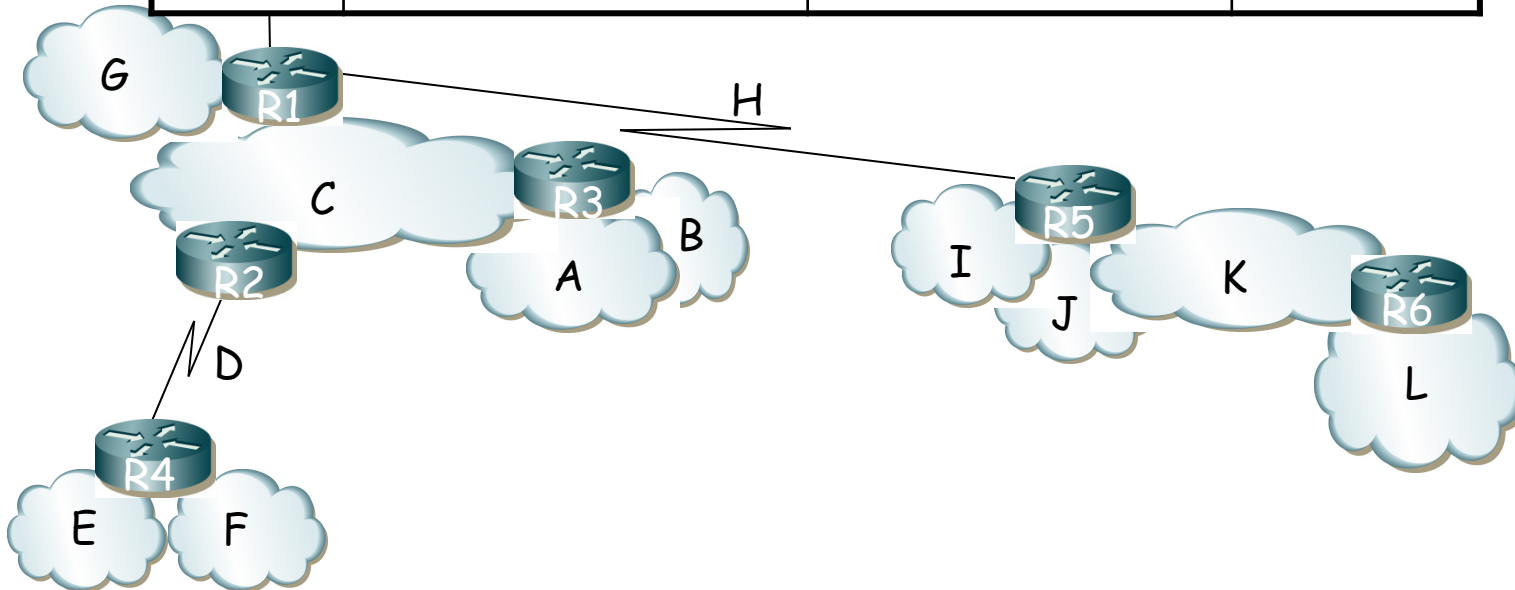


Ejemplo (4)

Mejora 3

Tabla de rutas de R4:

Destino	Next-hop	Interfaz
Red D	192.168.3.96 /30	(dir.connected)
Red E	192.168.3.64 /28	(dir.connected)
Red F	192.168.3.80 /28	(dir.connected)
(def.)	0.0.0.0 /0	IPR2ifRedD



CIDR

Permite:

- Asignar **redes** **más ajustadas** al tamaño necesario
- Bloque puede estar en cualquier rango disponible (**ignora clases**)

Necesita:

- Rutas deben emplear máscara
- El protocolo de enrutamiento debe transportar las máscaras
- Debería hacerse un reparto manteniendo jerarquía

Regional Internet

Registries (RIR):

- RIPE NCC (www.ripe.net)
Europa, Oriente Medio, Asia Central, África norecuatorial
- ARIN (www.arin.net)
América, parte del Caribe y África subecuatorial
- APNIC (www.apnic.net)
Asia y Pacífico
- LACNIC (www.lacnic.net)
América Latina y Caribe

Resumen

- Más flexibilidad en el tamaño de las redes empleando la máscara de subred
- Asignar espacios de direcciones más ajustados a las necesidades
- CIDR ignora el significado de las clases A, B y C
- Resumir varias rutas en una sola siempre que tengan un prefijo común
- Reducir con ello los tamaños de las tablas de rutas