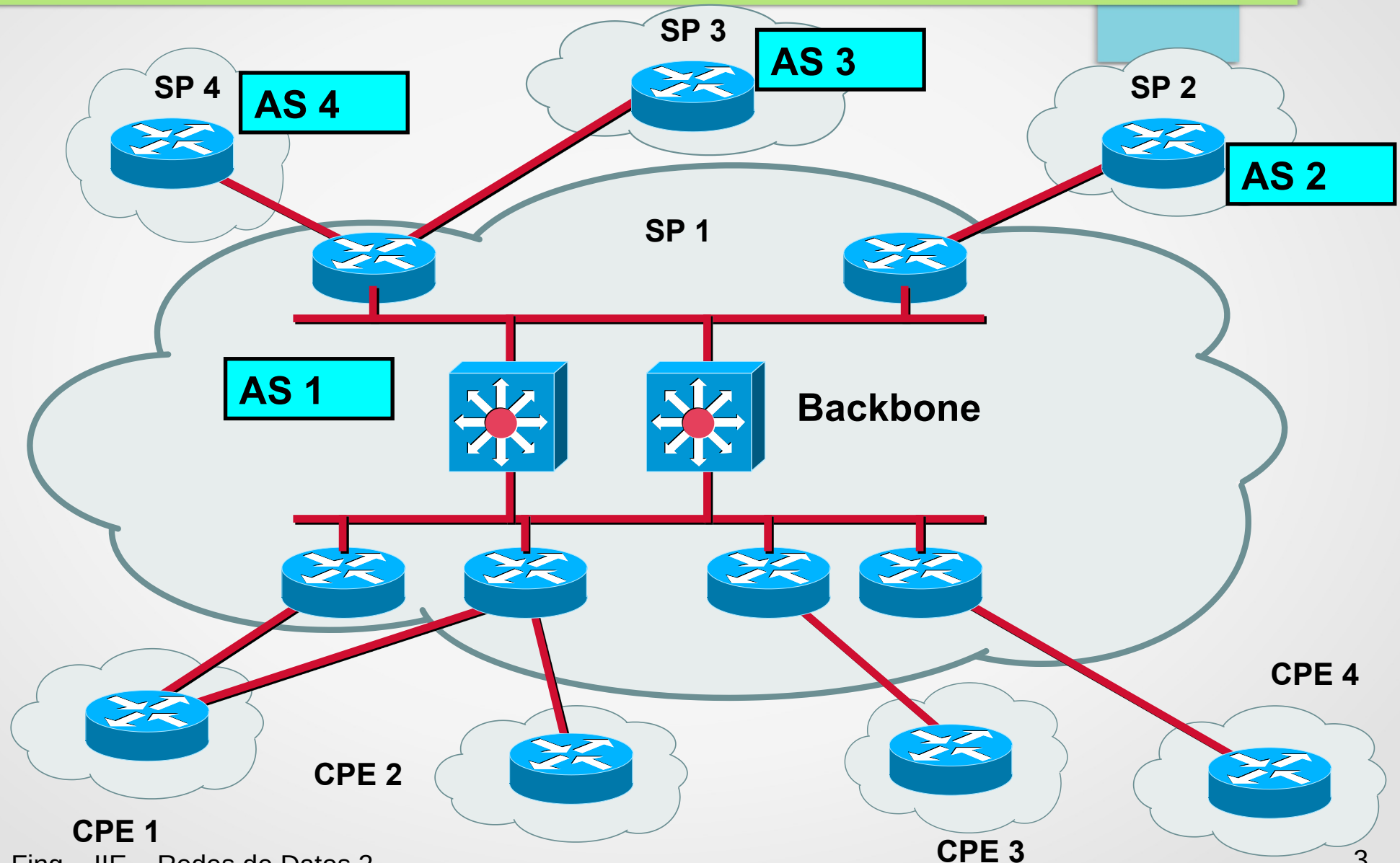
BGP

Border Gateway Protocol

Agenda (1)

- Conceptos Fundamentales de BGP
- Análisis del protocolo (BGP-4)
- Atributos de BGP y políticas de control
- IBGP mesh y Alternativas
- Sumarización y anuncios (CIDR)
- Damping y problemas de convergencia
- Extensiones Multiprotocolo
- Seguridad de BGP
- Salidas reales y datos de actualidad
- Ejemplo y consideraciones prácticas

Paradigma de red de un proveedor



Sistemas Autónomos (AS) (1)

- Conjunto de Redes y enrutadores bajo una política común de enrutamiento, administrados por una única autoridad
- Para el “exterior” el AS se ve como una única entidad
- Cada AS tiene asignado un número en el rango de 1 a 65,535 (privados del 64512 en adelante)

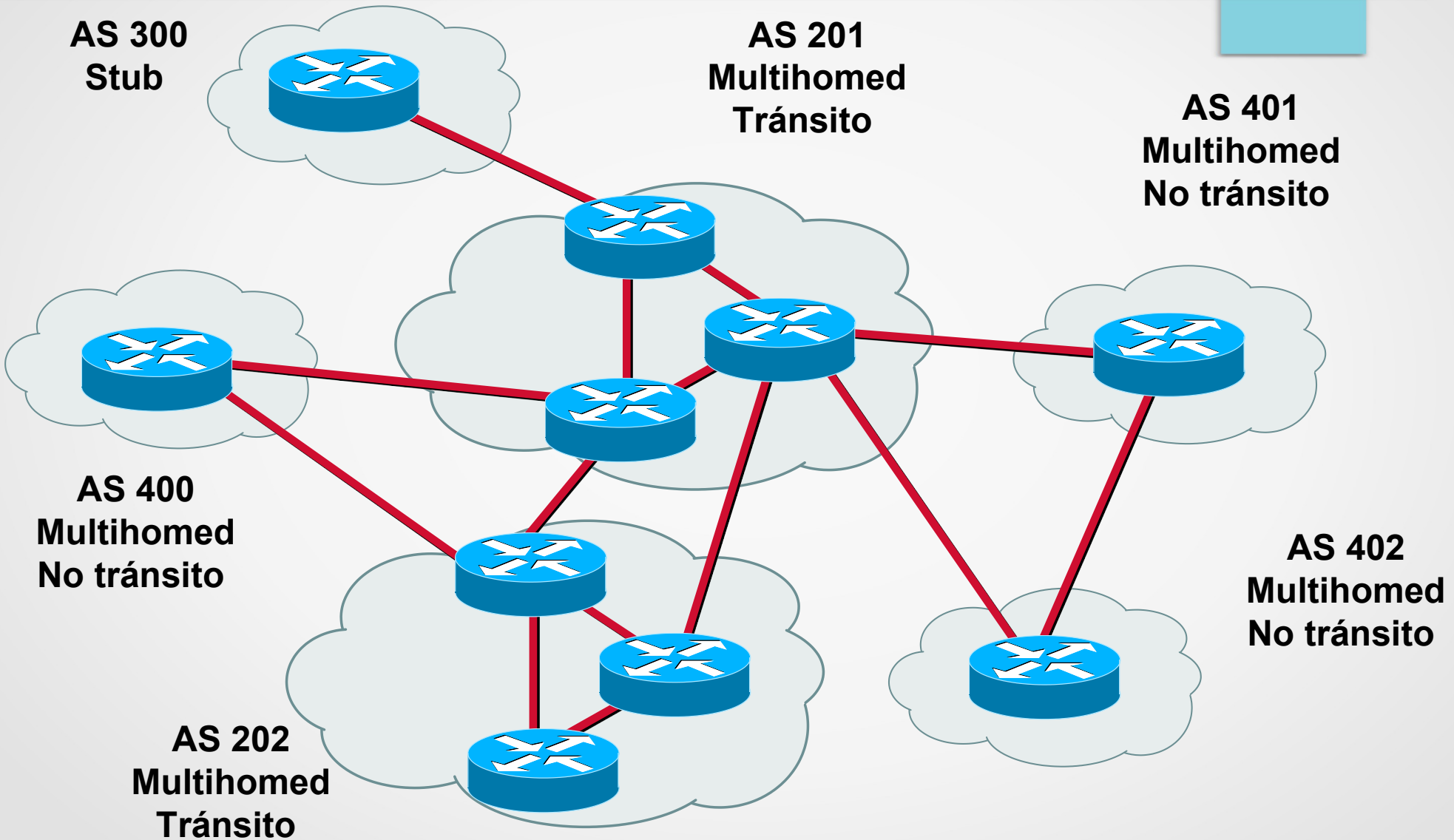
  Existe extensión a 32 bits (RFC 4893)
- Pueden coexistir varios IGP dentro de un mismo AS

Sistemas Autónomos (AS) (2)

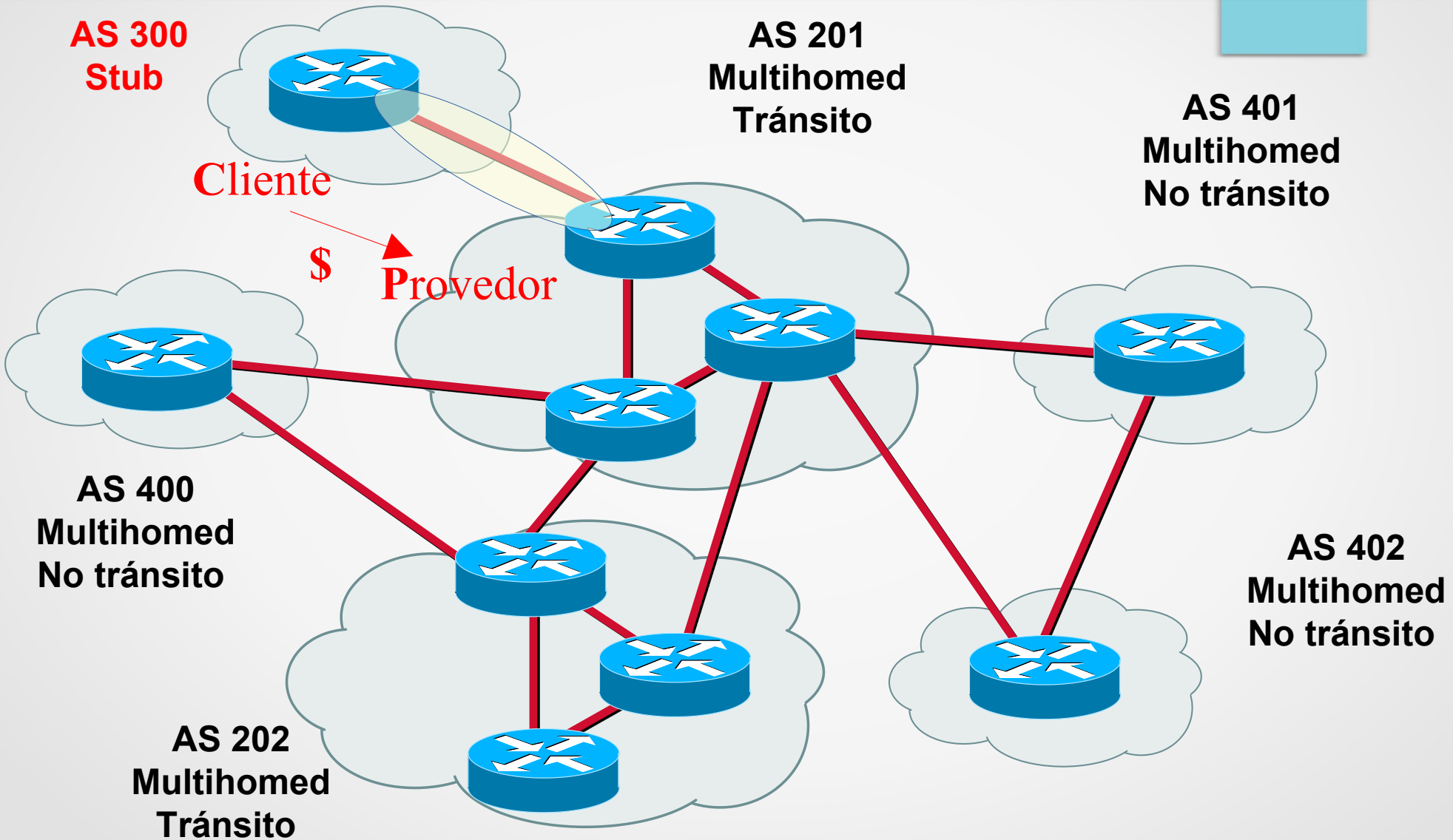
- Esta división (“internet”) en administraciones independientes permite trabajar con redes más pequeñas y “manejables”
- **Tipos de AS:**
  - Stub AS
  - AS Multihomed de no tránsito
  - AS Multihomed de tránsito
- **Relaciones Comerciales:**

¿Qué servicios ofrece un AS a otro AS?

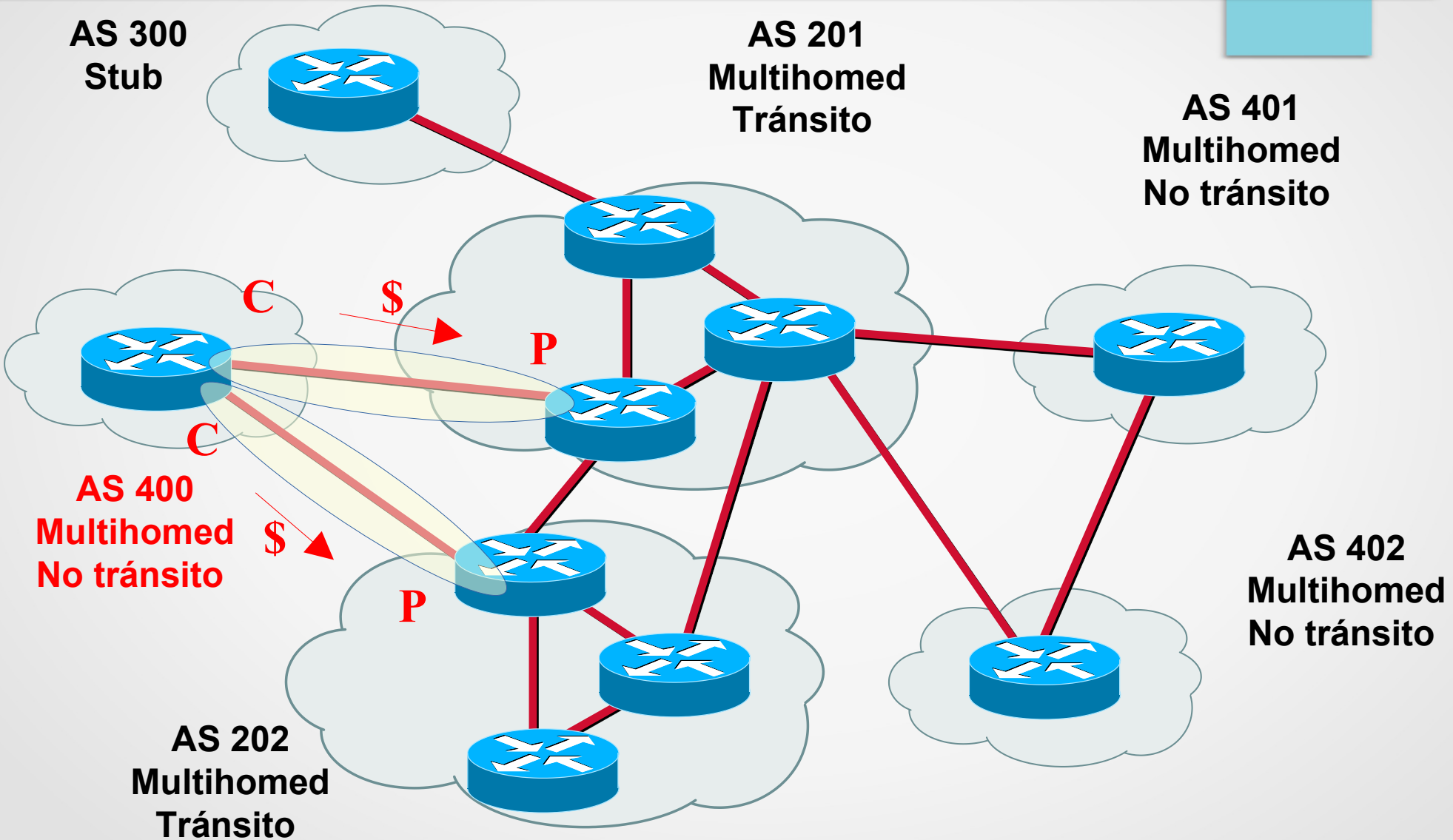
Sistemas Autónomos (AS) (3)



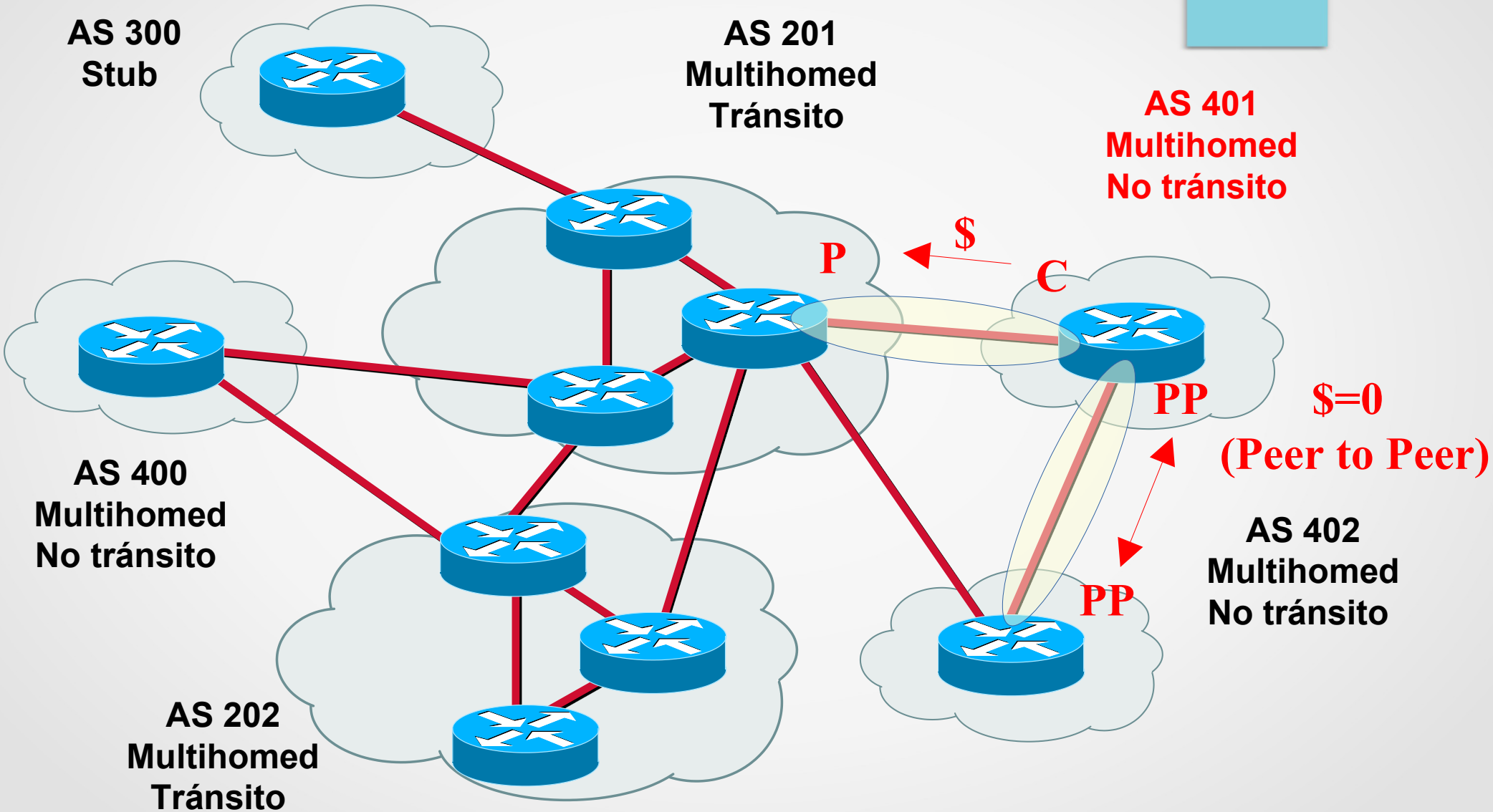
Sistemas Autónomos (AS) (3a)



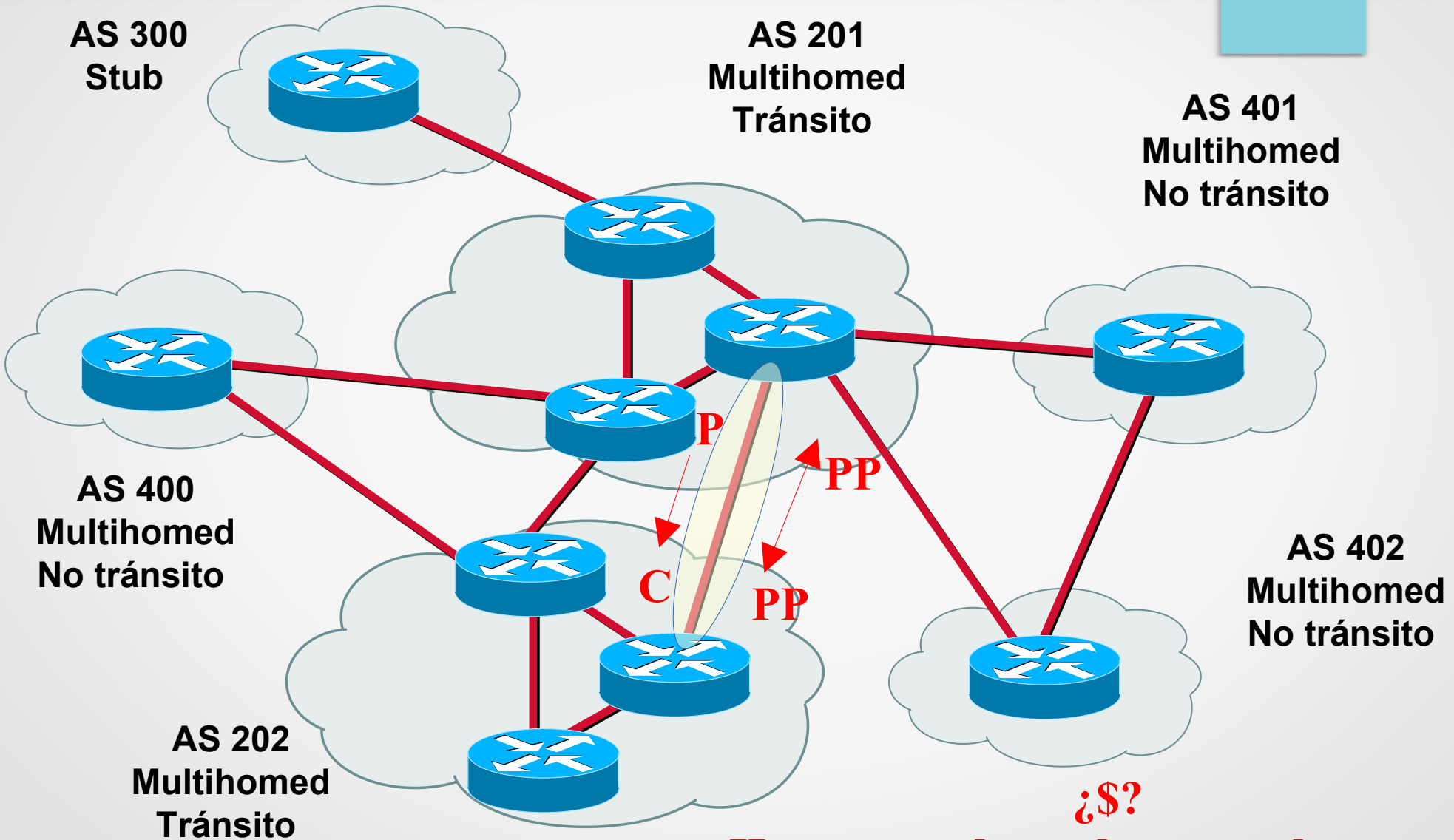
Sistemas Autónomos (AS) (3b)



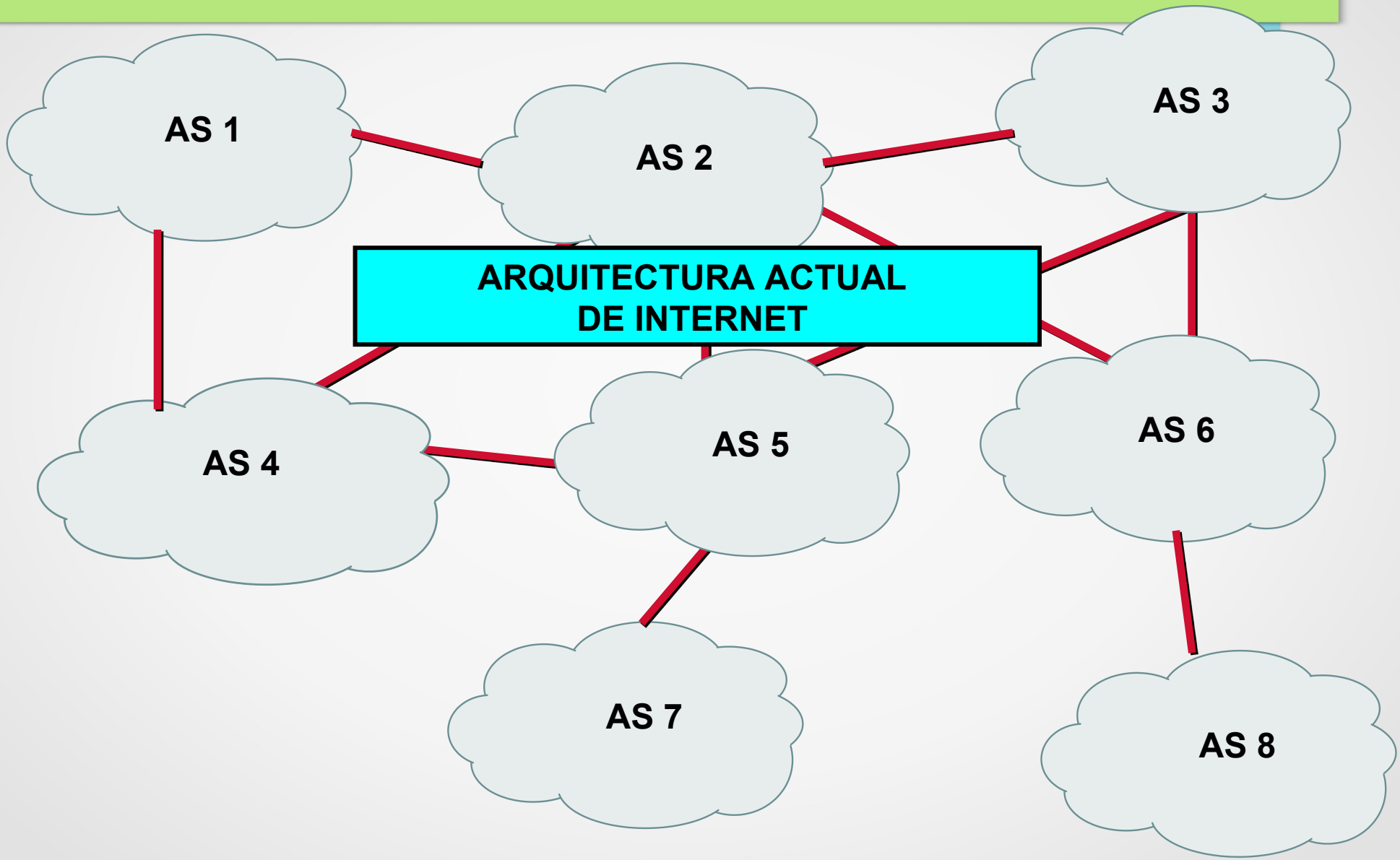
Sistemas Autónomos (AS) (3c)



Sistemas Autónomos (AS) (3d)



Sistemas Autónomos (AS) (4)



Internet (1)

- Interconexión de múltiples AS
- Es una red de redes
- No existe explícitamente un Backbone. Los mayores NSP (Network Service Provider) “hacen” las veces de Backbone
  - Se les suele llamar proveedores “**Tier 1**”
- No hay una definición explícita de “**Tier 1**”, pero en general se asume que son aquellos proveedores que tienen presencia en todo el mundo y que “no le pagan a nadie” para llegar a todos los destinos.
  - RFC 7454 **Tier 1 transit provider:** an IP transit provider that can reach any network on the Internet without purchasing transit services.
- **Idea:** https://www.caida.org/projects/cartography/as-core/2020/

Internet (2)

Pueden plantearse varias preguntas:

- ¿Cómo intercambiar información de ruteo entre los distintos AS?
- ¿Ruteo estático ó dinámico?
- ¿Influye el número de puntos de salida de un AS en la decisión?
- Para el caso dinámico: ¿Estado de enlace o Vector distancia?

Internet (3)

¿Por qué no usar un IGP: RIP, OSPF u otro?

- **Escalabilidad:**
  - Sería como una única red (¿cómo expresar políticas de ruteo?)
  - Máx. número de hops acotado en algunos
  - Tamaño de las tablas de ruteo (y topología) inmanejable
- **Estabilidad:**
  - Capacidad de adaptación a los cambios
  - Convergencia
  - Tráfico de Control inmanejable
- **Necesidad de políticas de enrutamiento**

  Un IGP buscan el camino mas corto, no contempla el camino de menor el costo ($) de utilización o que no transite por determinados AS.

BGP

- BGP permite interconectar múltiples AS conformando la Internet que conocemos (Se dice que es la “goma” que mantiene unida internet)
- BGP es lo que se denomina un Interdomain Routing Protocol o InterAS routing protocol
- BGP es un EGP (Exterior Gateway Protocol)
- BGP-4 es hoy por hoy un estándar de facto en Internet

Ventajas de BGP (1)

- Escalabilidad: diseñado para manejar grandes tablas de rutas (full-routing implica del orden de 940.030 prefijos). No requiere excesivo tráfico de control
- Estabilidad: se adapta a los cambios fácil y rápidamente
- Sencillez: de la familia de los protocolos de vector distancia (no requiere estructura jerárquica ni conocimiento de la topología de red)
- **Full-routing o NDZ (Non Default Route)**: Tabla de rutas para alcanzar a todos los destinos posibles sin tener una ruta por defecto.

Ventajas de BGP (2)

- Soporta políticas de enrutamiento distintas y administración independiente por AS
- No impone restricciones al tamaño del bloque a anunciar
- Es un estándar
- Garantiza intercambio de información de ruteo libre de loops
- Es extensible: “no se puede cambiar de protocolo cada vez que hay surgen requerimientos nuevos”. Es un protocolo que permite **extensiones mediante campos opcionales**, por ello logra adaptarse a nuevas necesidades e interoperar con versiones que difieran en el conjunto de extensiones que soportan.

Agenda (2)

- Conceptos Fundamentales de BGP
- Análisis del protocolo (BGP-4)
- Atributos de BGP y políticas de control
- IBGP mesh y Alternativas
- Sumarización y anuncios (CIDR)
- Damping y problemas de convergencia
- Extensiones Multiprotocolo
- Seguridad de BGP
- Salidas reales y datos de actualidad
- Ejemplo y consideraciones prácticas

BGP versión 4

- Definido originalmente en la **RFC 1771** (1995)

  Las versiones anteriores se manejaban con “clases”

- **RFC 4271** (enero 2006, draft standard), pequeñas modificaciones a RFC 1771. Se adapta al uso real actual
- Se ha extendido sus funcionalidades mediante múltiples RFCs

  Ejemplo: cambio de AS 16 bit a 32 bit, intercambio de prefijos IPv6 y etiquetas para VPN MPLS..

- No se ve un sustituto a BGP en un futuro cercano:
  - La academia no convence a los fabricantes de equipos
  - Tiene que poder coexistir (ganarse confianza)
  - Dudas sobre los impacto en la escala actual
  - Al menos para intercambio entre AS.

“Conexión BGP”

- Concepto de “**peers**” o “**neighbors**” (vecinos)

  No hay descubrimiento, **deben configurarse explícitamente**

- Utiliza TCP, puerto 179

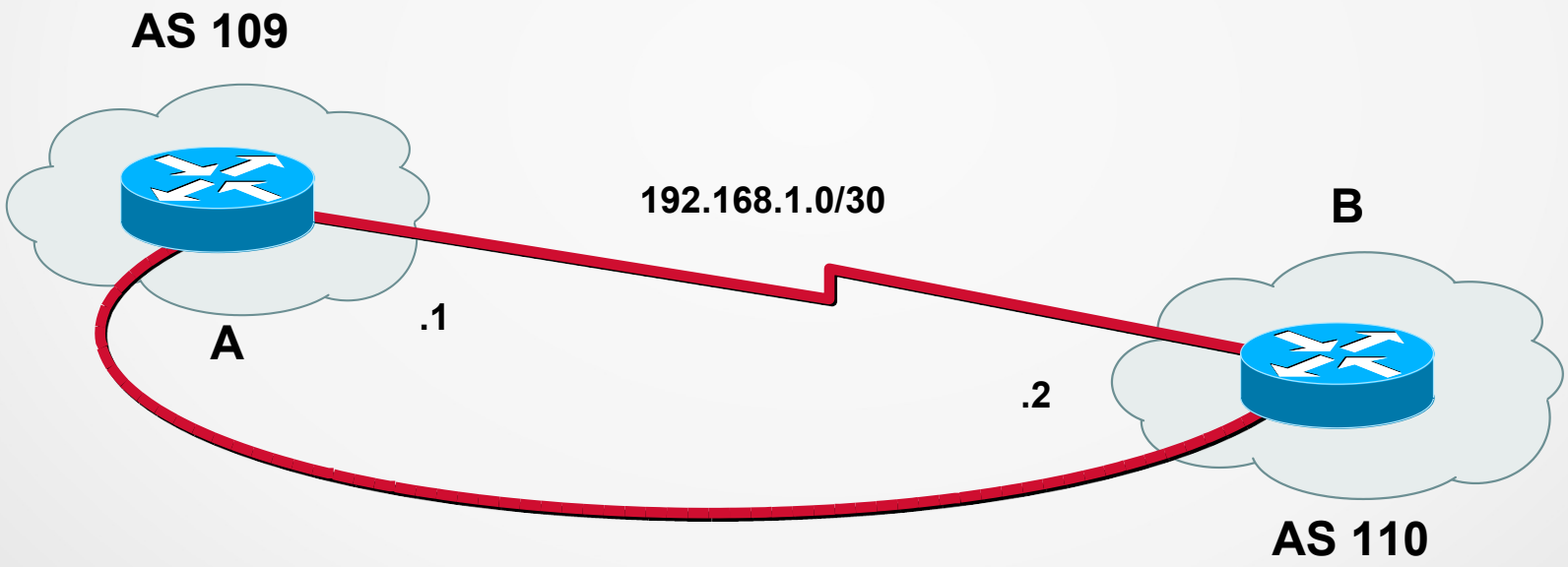
  La confiabilidad recae en la capa de transporte simplificando el protocolo

- La sesión TCP **requiere de una capa de ruteo interno** que me permita llegar al vecino (IGP, estático o directamente conectado)

- Se distingue entre BGP interno (**iBGP**) y externo (**eBGP**)

EBGP: BGP Externo (External BGP)

- Cuando los vecinos BGP pertenecen a distintos AS
- En general los vecinos se encuentran directamente conectados

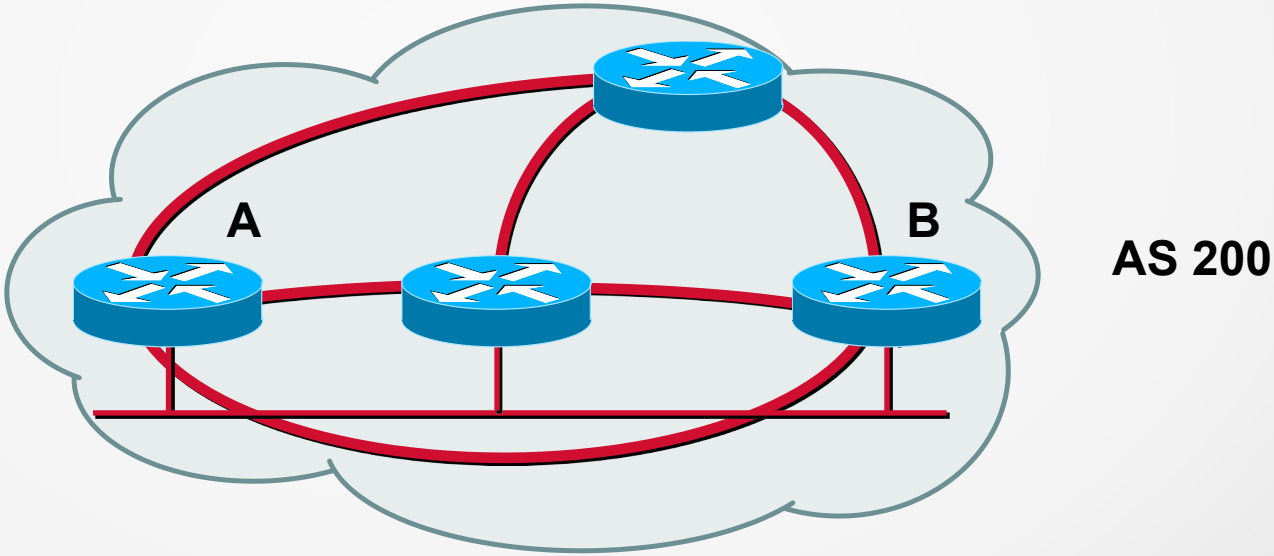


- Enseñar prefijos a otros AS, aprender prefijos de otros AS

IBGP: BGP Interno (Internal BGP)

- Cuando los vecinos pertenecen al mismo AS
- Las conexiones representan sesiones BGP entre vecinos y no necesariamente links físicos

Full mesh en principio

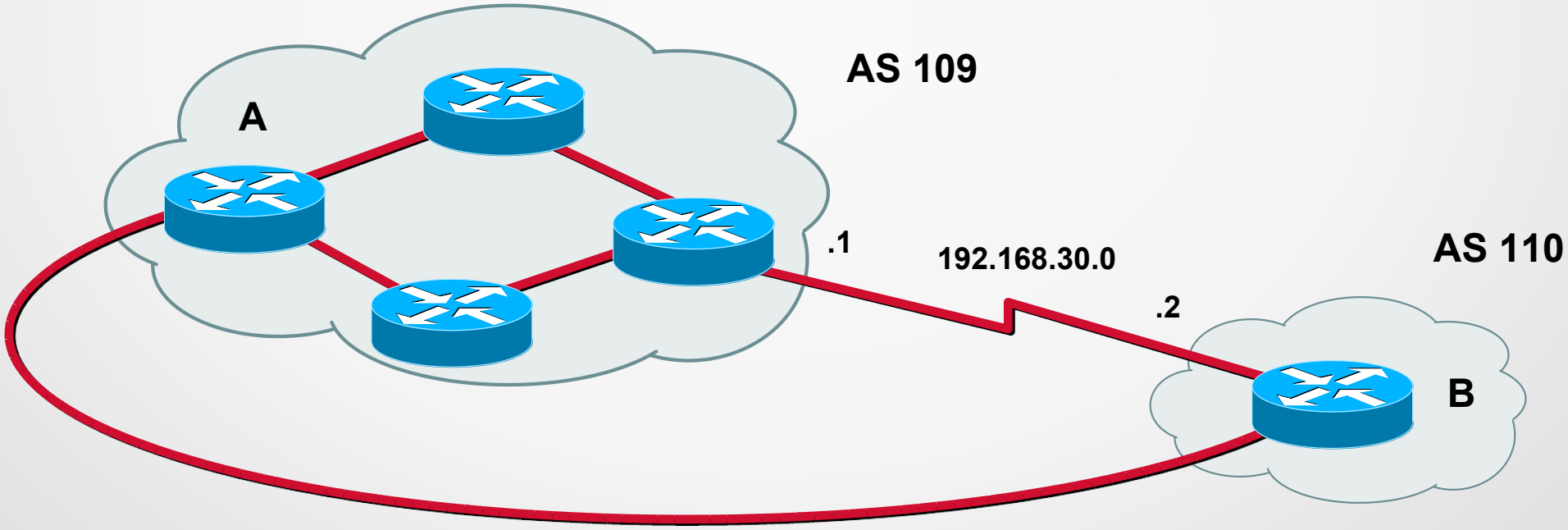


- ¿Como distribuyo prefijos aprendidos de un AS dentro de mi AS o a otro AS (tránsito)?

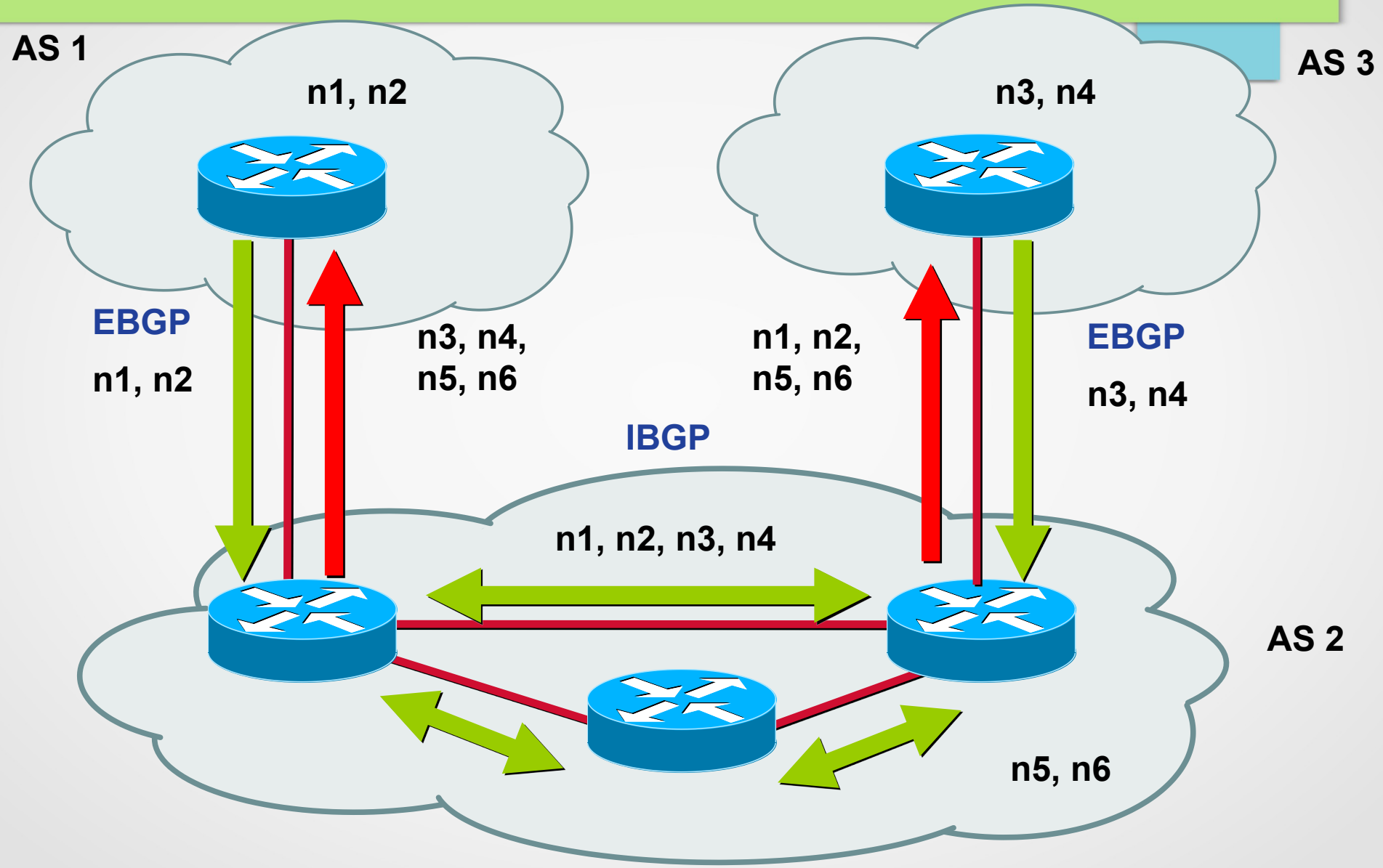
EBGP multihop

- Cuando los vecinos no se encuentran directamente conectados, se dice que la sesión entre ellos es **EBGP-Multihop**

Debe configurarse explícitamente



Ej. **IBGP, EBGP tránsito**



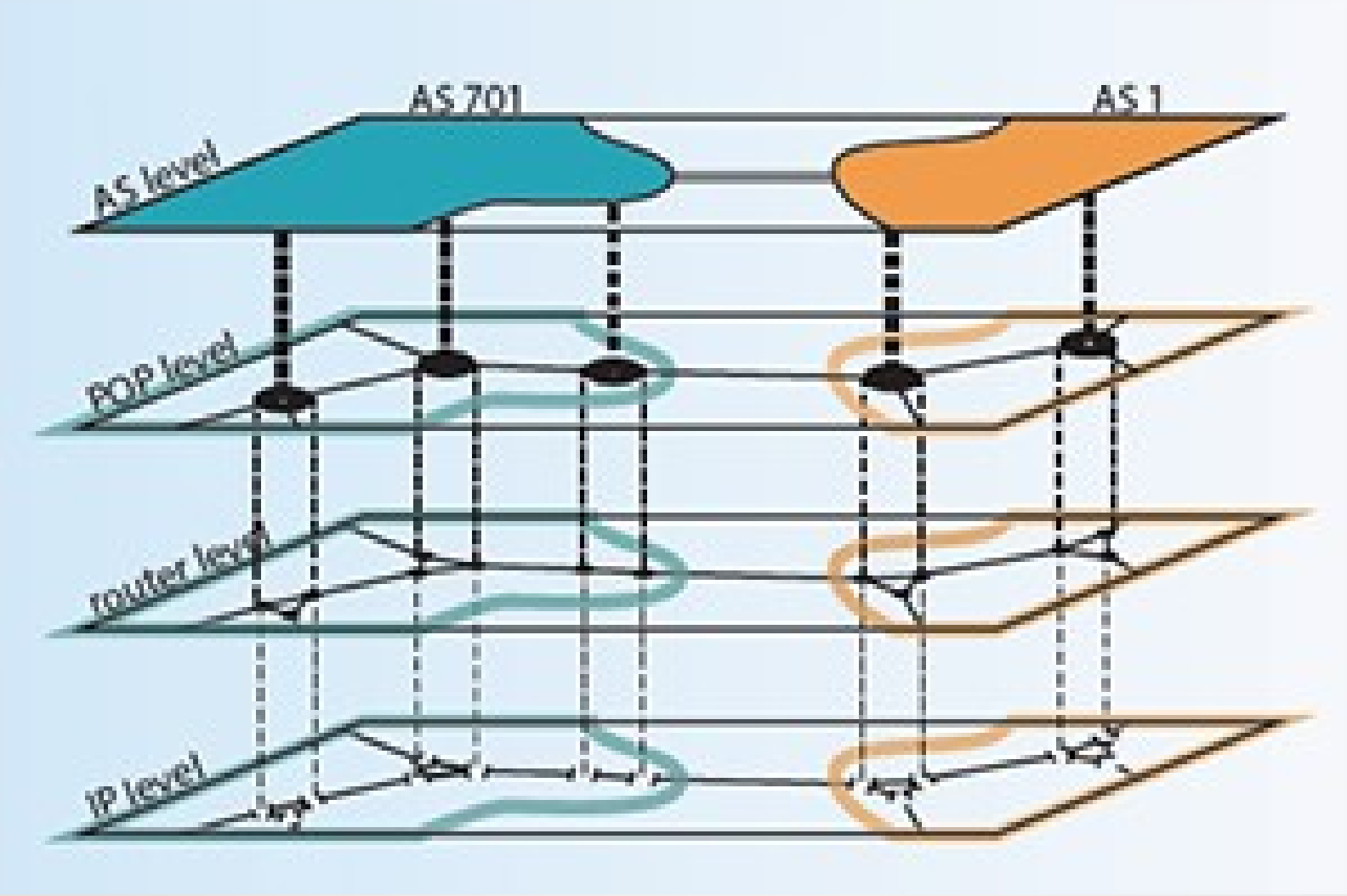
Ej. **IBGP, EBGP no tránsito**

¿Cómo trabaja BGP? (1)

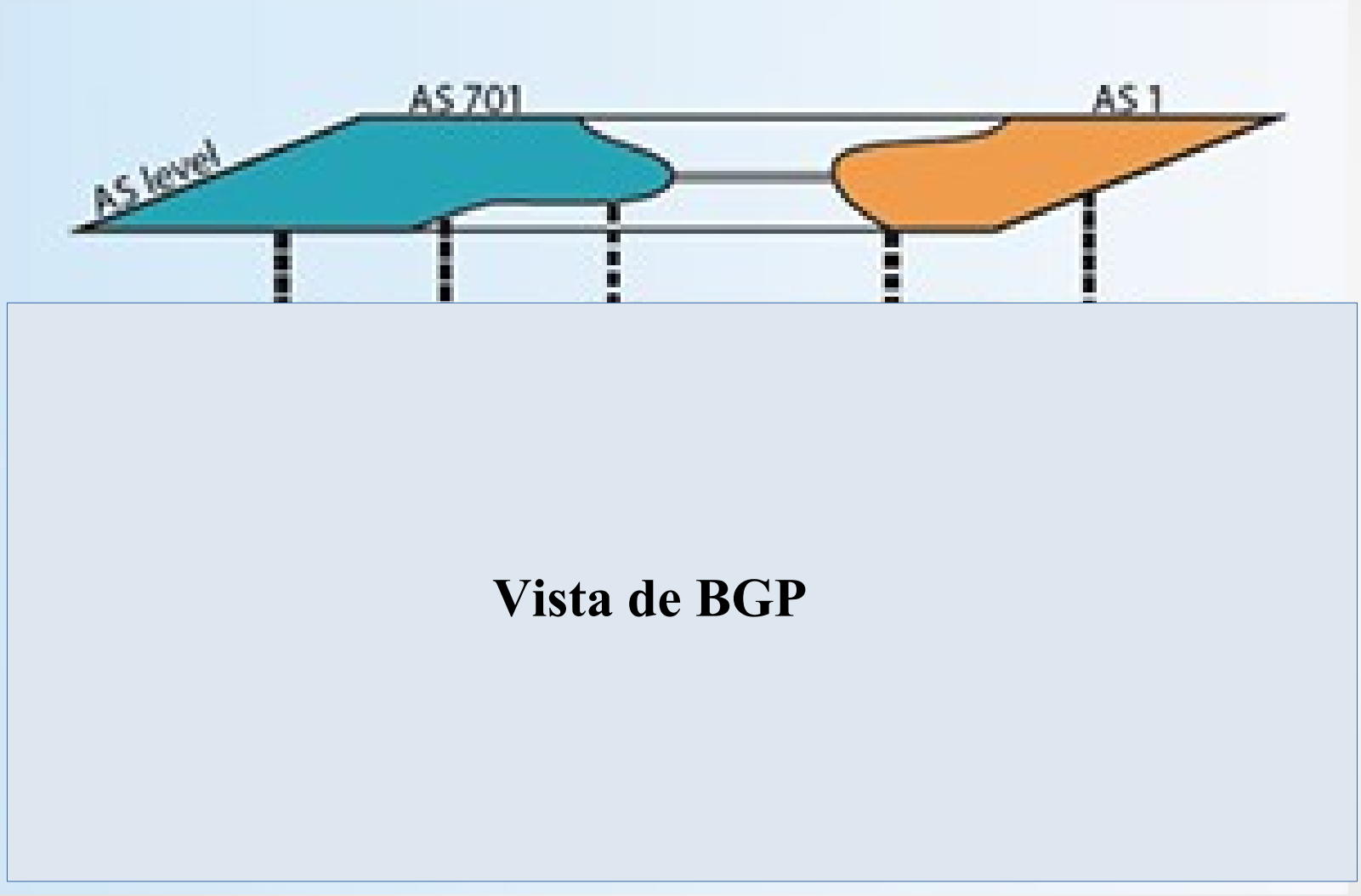
- Se propaga cada prefijo, con un conjunto de atributos, incluyendo el camino de sistemas autónomos por el que pasó el anuncio
- Se dice que BGP es un "**path vector protocol**" (métrica la cantidad de ASs).

  Cada anuncio incluye una lista con la secuencia completa de ASs que un paquete debe atravesar para llegar a la red de destino

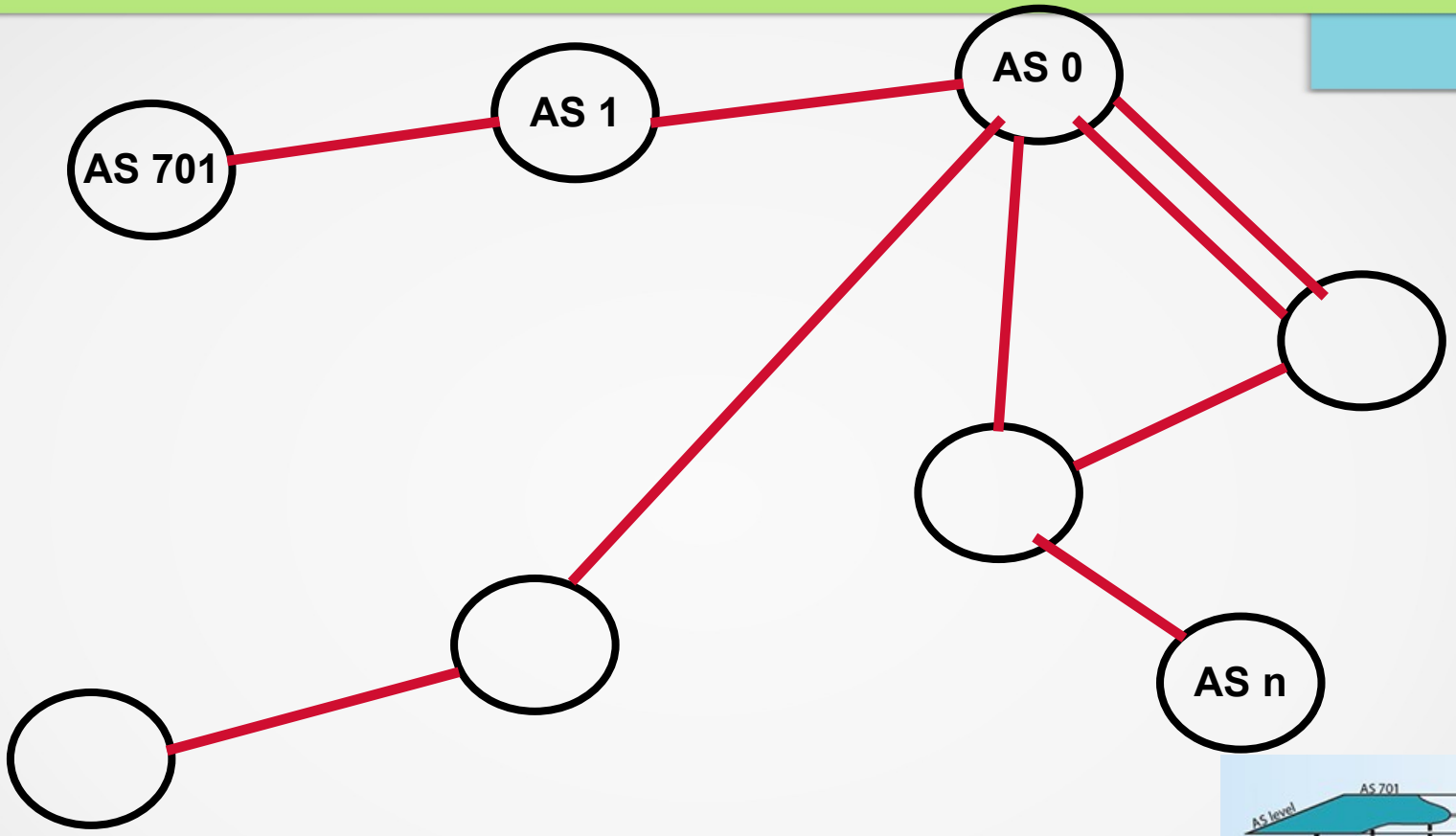
¿Cómo trabaja BGP? (2) **Topología**



¿Cómo trabaja BGP? (2) **Topología**

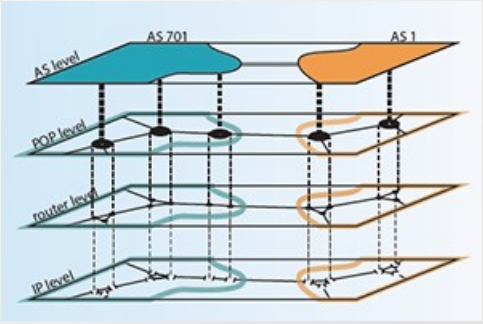


¿Cómo trabaja BGP? (3) **Topología**



AS_Path Tree

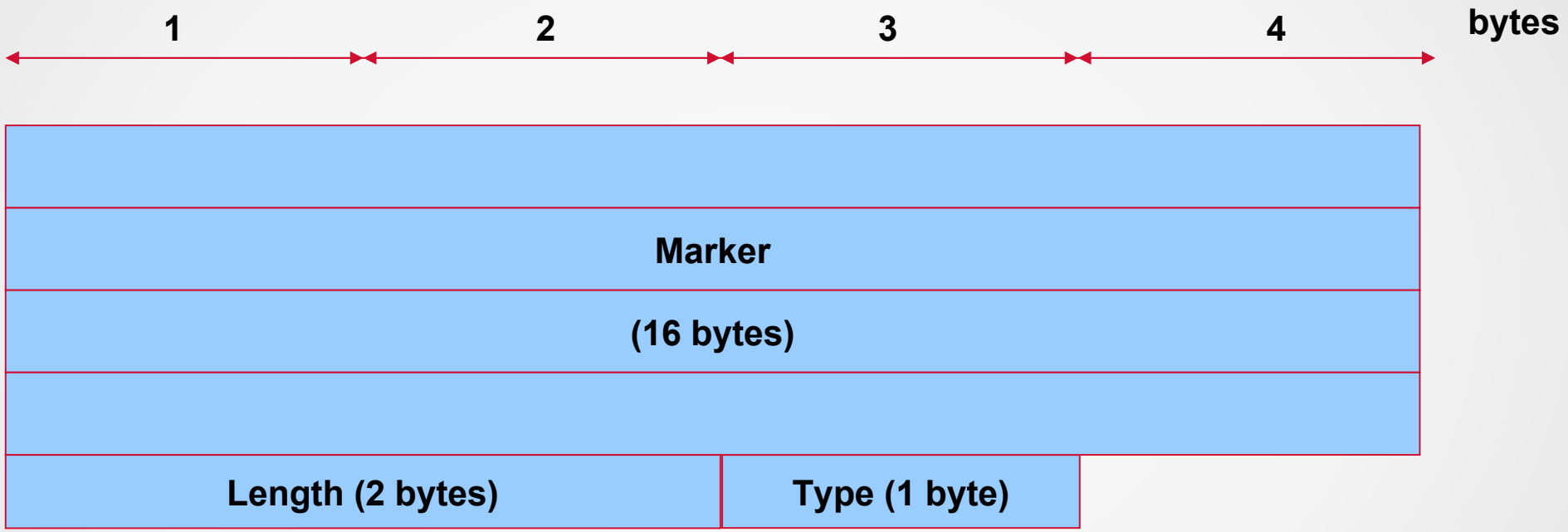
Link BGP



¿Cómo trabaja BGP? (4)

- ¿Cómo establece una sesión BGP sobre TCP entre dos peers?
- ¿Cómo se intercambia información de ruteo en BGP?
- ¿Qué tipo de mensajes intercambian dos peers de BGP?
- ¿Cómo se mantiene “viva” la sesión una vez establecida?

Encabezado (header) de BGP (1)



- Total: 19 bytes

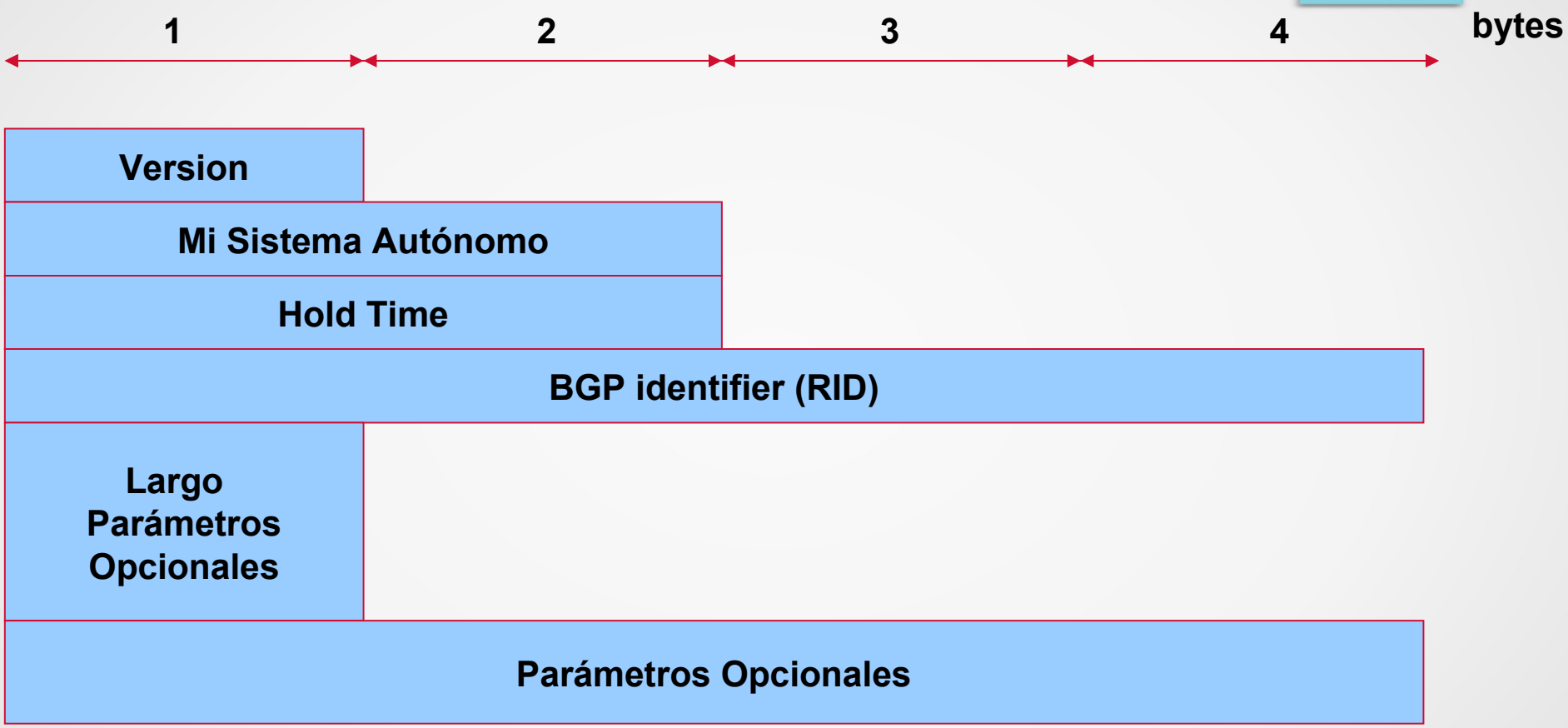
Encabezado de BGP (2)

- **Marker:** contiene una secuencia que puede ser predecida por el peer remoto
  - **RFC 1771** (en desuso) Autenticar los mensajes BGP recibidos en caso de usar autenticación
  - **RFC 4271**:Secuencia de unos (binario). Detectar pérdida de sincronización en caso de no usar autenticación de mensajes
- **Length:** largo total del mensaje incluido el encabezado
- **Type:** Open, Update, Keepalive, Notification

Tipos de mensajes BGP

- **OPEN:** iniciar sesión BGP
- **NOTIFICATION:** condición de error
- **UPDATE:** alta o baja de rutas
- **KEEPALIVE:** confirmación periódica
- Largo de los mensajes:
  - Mínimo 19 bytes (sólo header)
  - Máximo 4096 bytes

Mensaje OPEN (1)



- 10 bytes mínimo
- Formato Opciones: Type, Length, Value (TLV)

Mensaje OPEN (2)

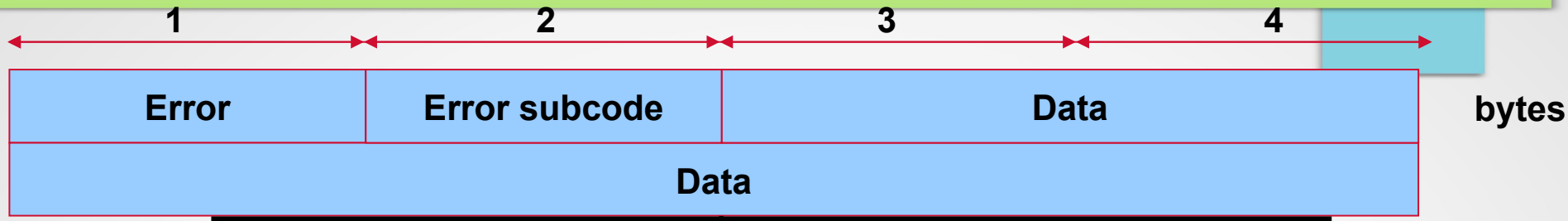
- Version: 4
- Mi AS: Sistema autónomo del emisor
  - Así se distingue IBGP de EBGP
  - Se verifica configuración
- BGP ID: ID del router que envía el mensaje
- Hold time:
  - Tiempo máximo en segundos que puede transcurrir sin recibir mensajes de UPDATE o KEEPALIVE
  - Este tiempo se negocia al iniciar la sesión (mínimo entre ambos extremos, no menos de 3 segundos)

Mensaje OPEN (3)

- **Parámetros Opcionales:** como su nombre lo indica, parámetros opcionales que se negocian al iniciar la relación de vecinos (Por ej. “**capabilities**”), formato TLV (Type 1 byte, Length 1 byte and Value)
- **Largo de parámetros opcionales:**

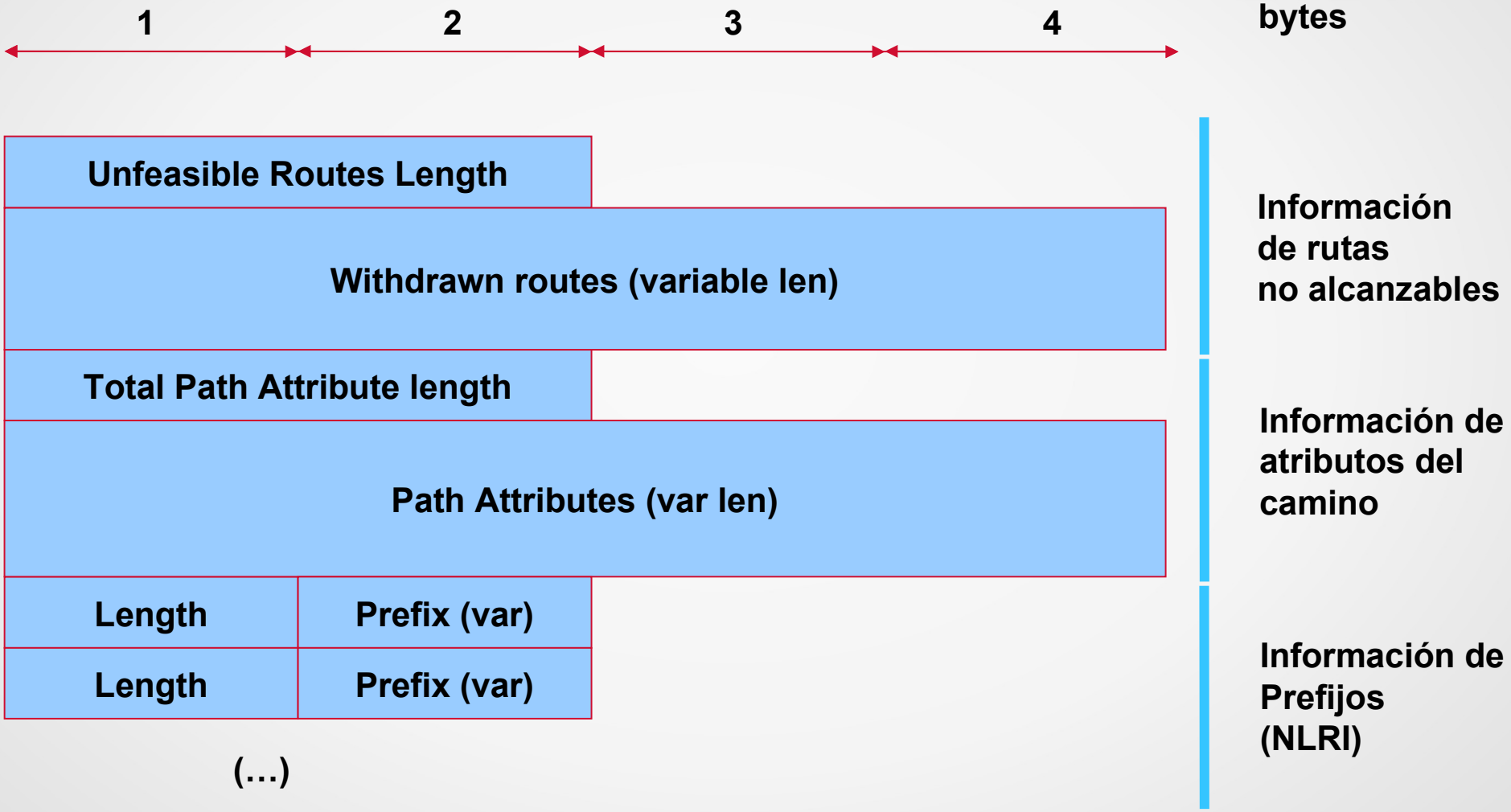
“0” indica que no se negociarán parámetros opcionales

Mensaje NOTIFICATION



Error code	Error subcode
1- message Header Error	1: Connection Not sync 2: Bad message length 3: Bad message type
2-Open message error	1: Unsupported version numb 2: Bad Peer AS 3: Bad BGP identifier 4: Unsupported Optional Par. 5: Authent error 6: Unacceptable hold time
3-UPDATE message error	1: Malformed Attribute-list 2: Unrecognised well-know attr. 3: Missing well-know attribute (…)
4-Hold timer expired	NA
5-Finite state machine error	NA
6-Cease	NA

Mensaje UPDATE (1)



Mensaje UPDATE (2)

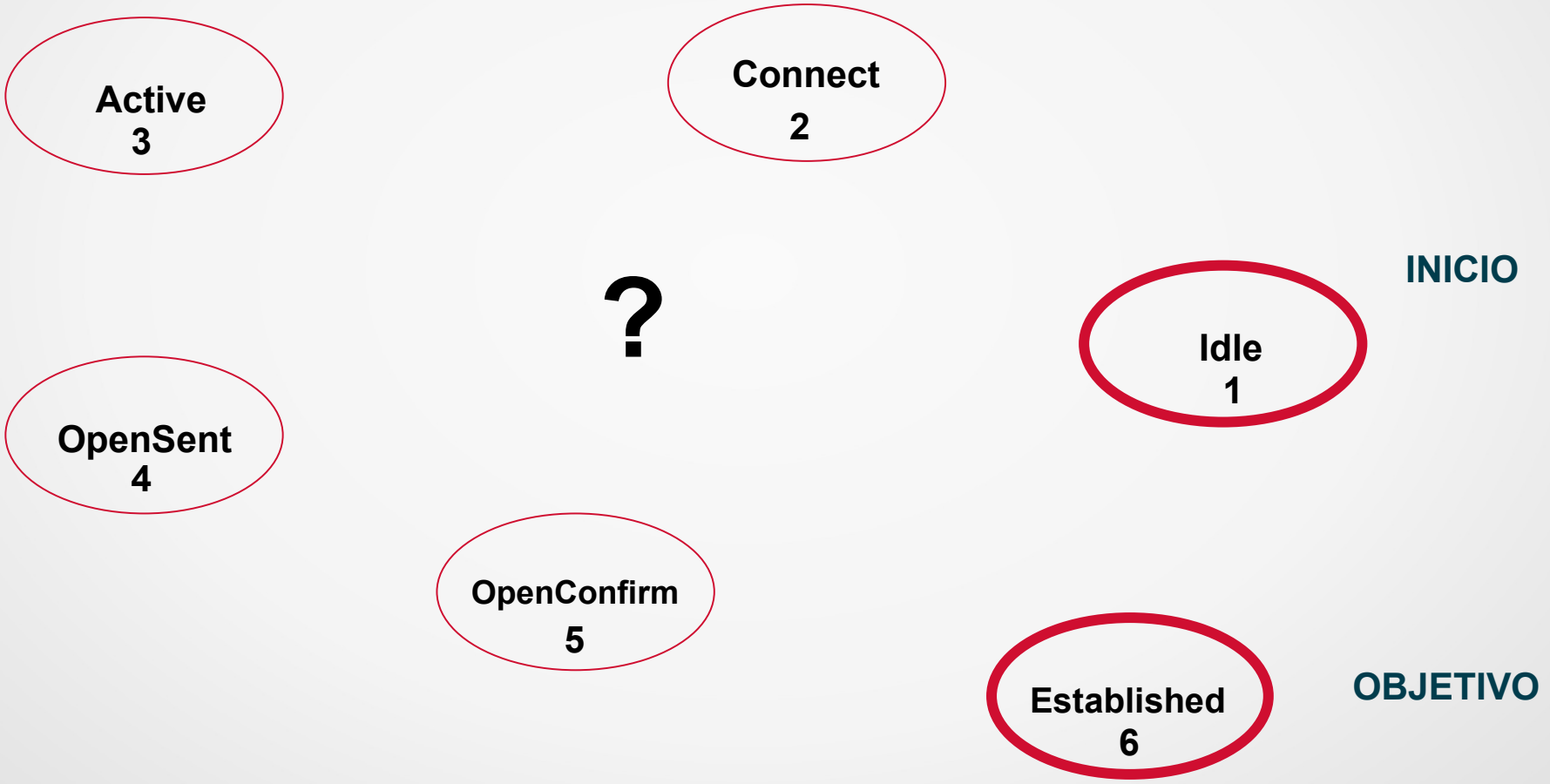
- **Withdrawn routes:** Prefijos anunciados previamente que ya no son alcanzables
- **Path Attributes:** Atributos de un determinado “camino”
- Información de **NLRI** (Network Layer Reachability Information): prefijos que comparten un camino y los atributos
- Pueden haber varios mensajes de Update en un mismo mensaje BGP.

Keepalive message

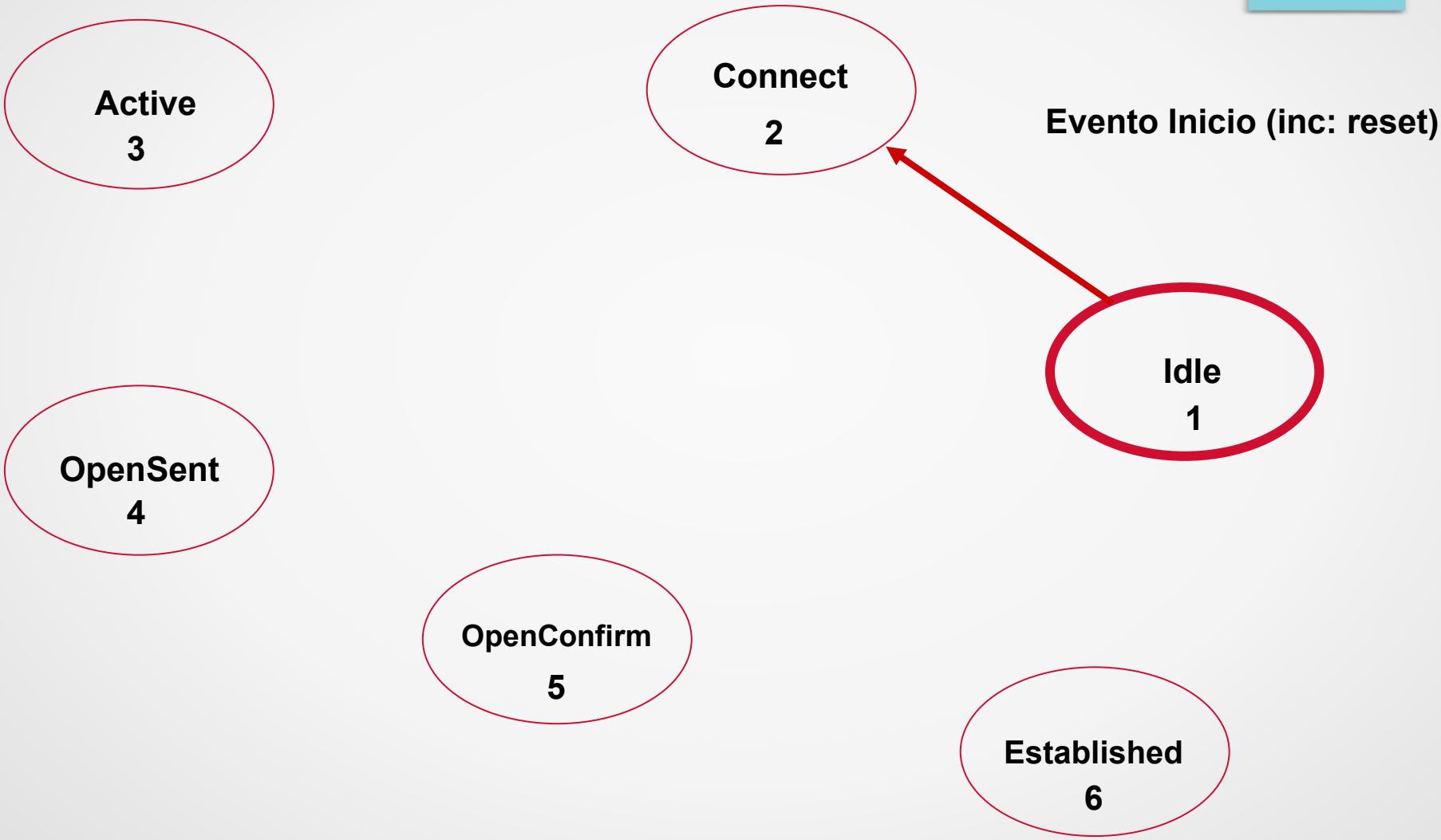
- Consiste simplemente en el Header
- 19 bytes intercambiados periódicamente

Inicio de una sesión BGP

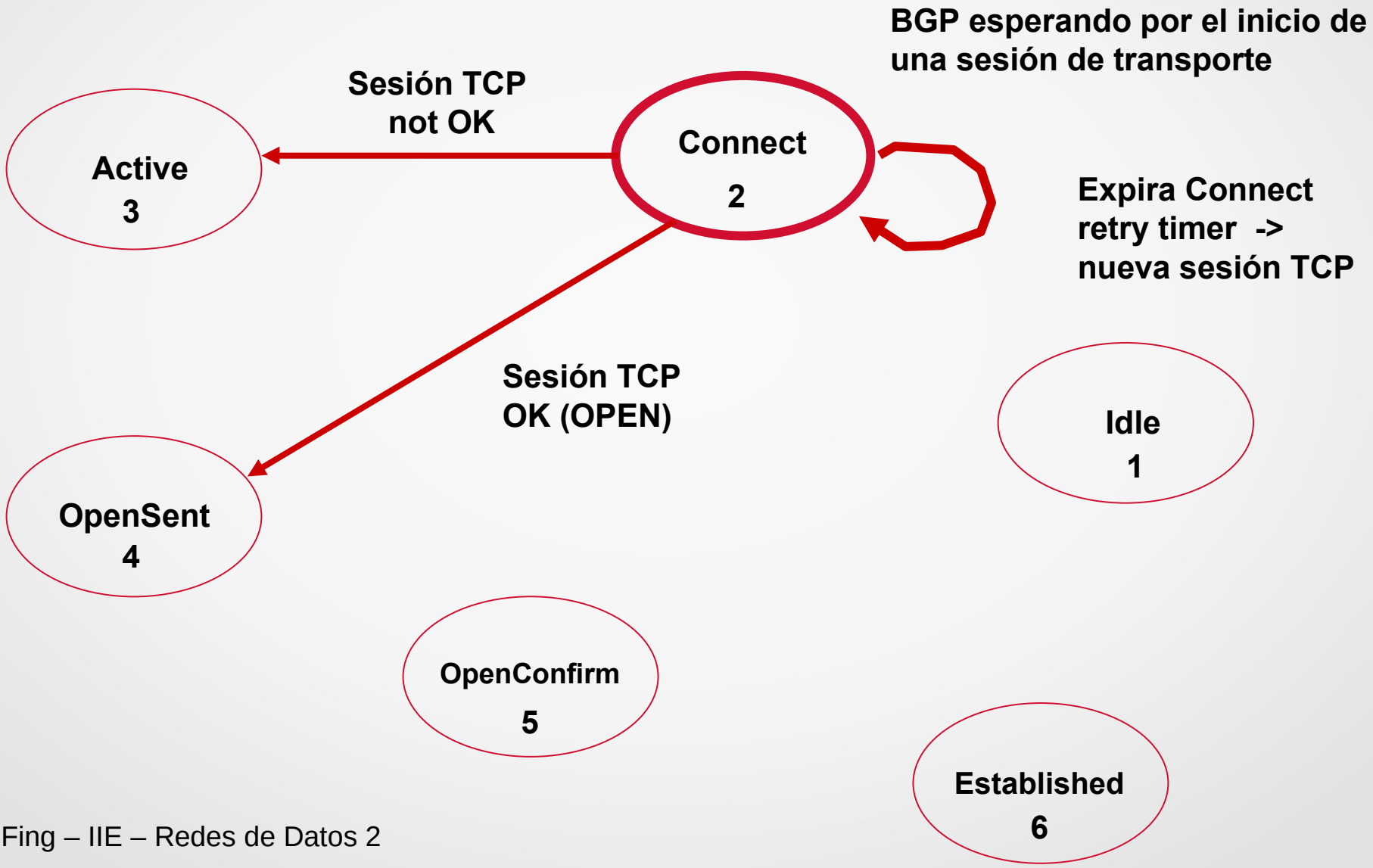
Diagrama de Estados (RFC 1771)



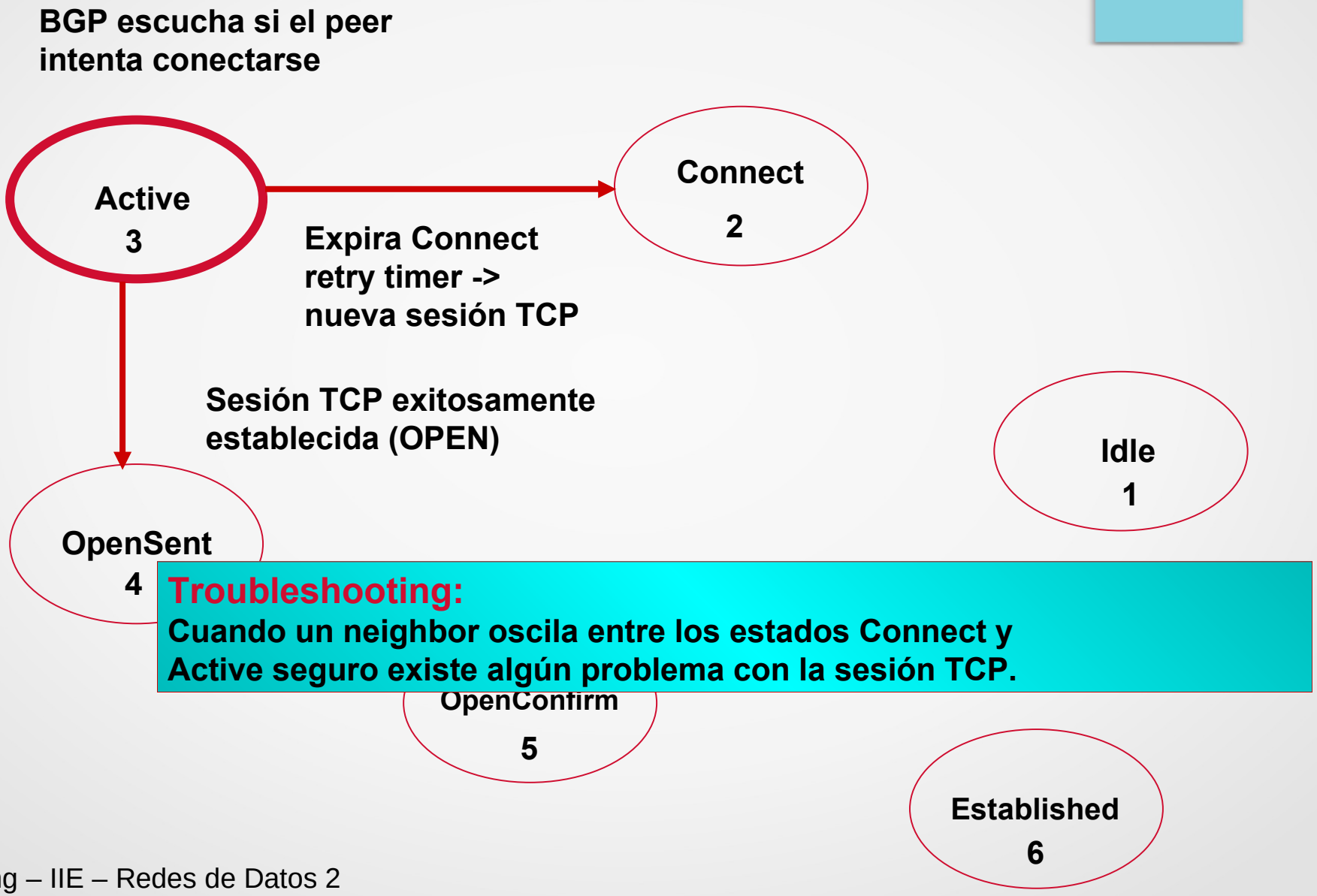
Inicio de una sesión BGP



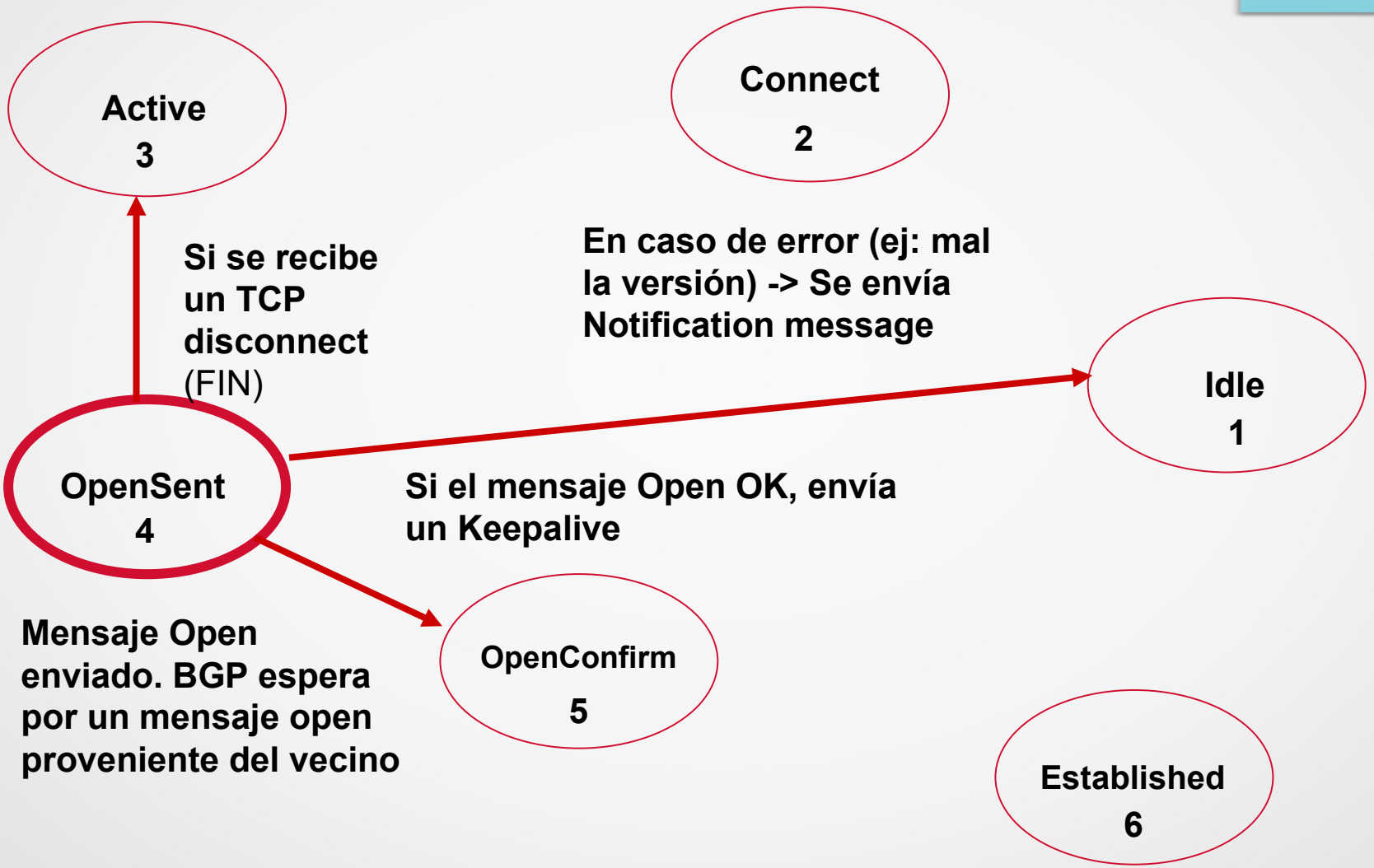
Inicio de una sesión BGP



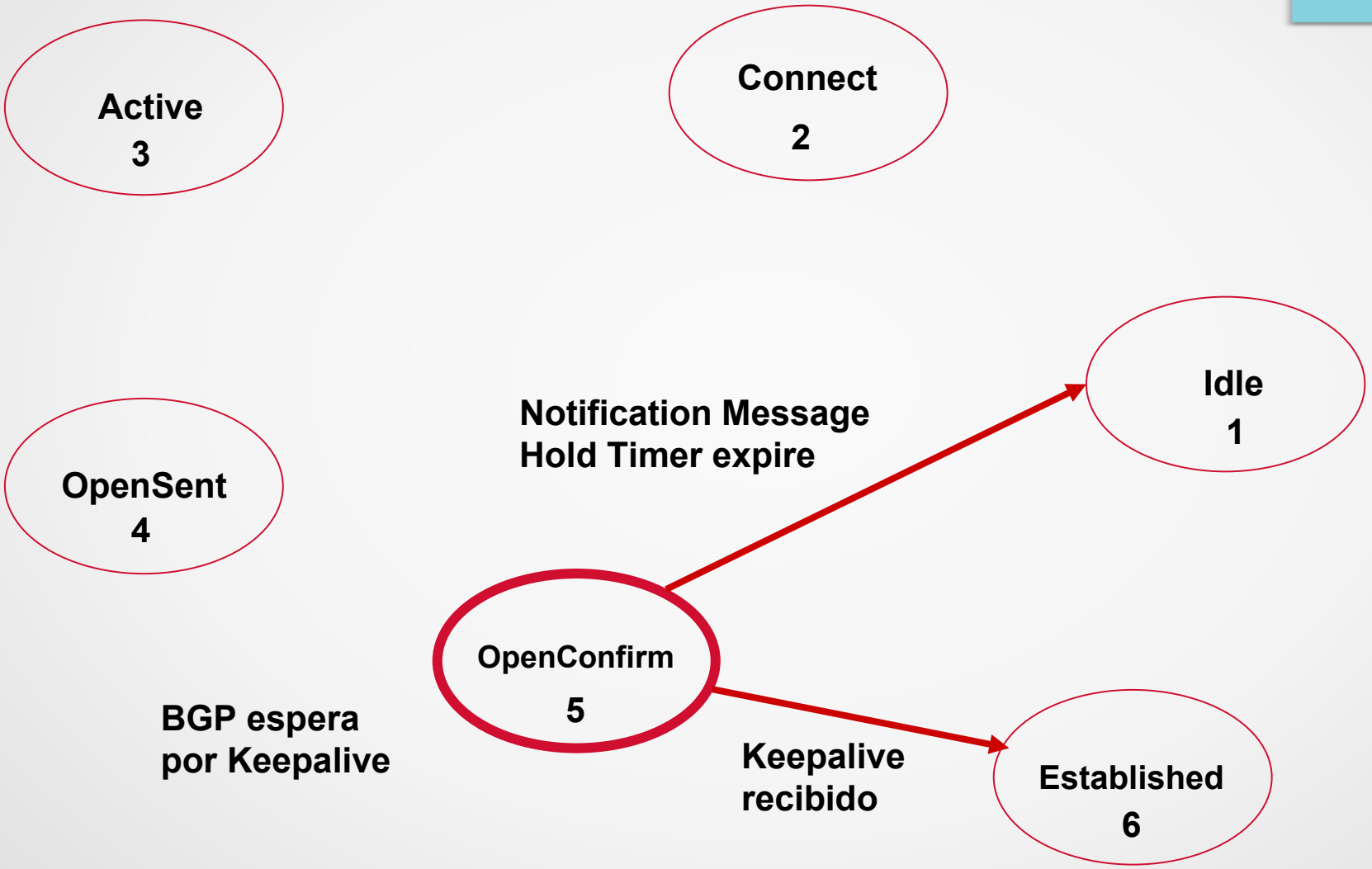
Inicio de una sesión BGP



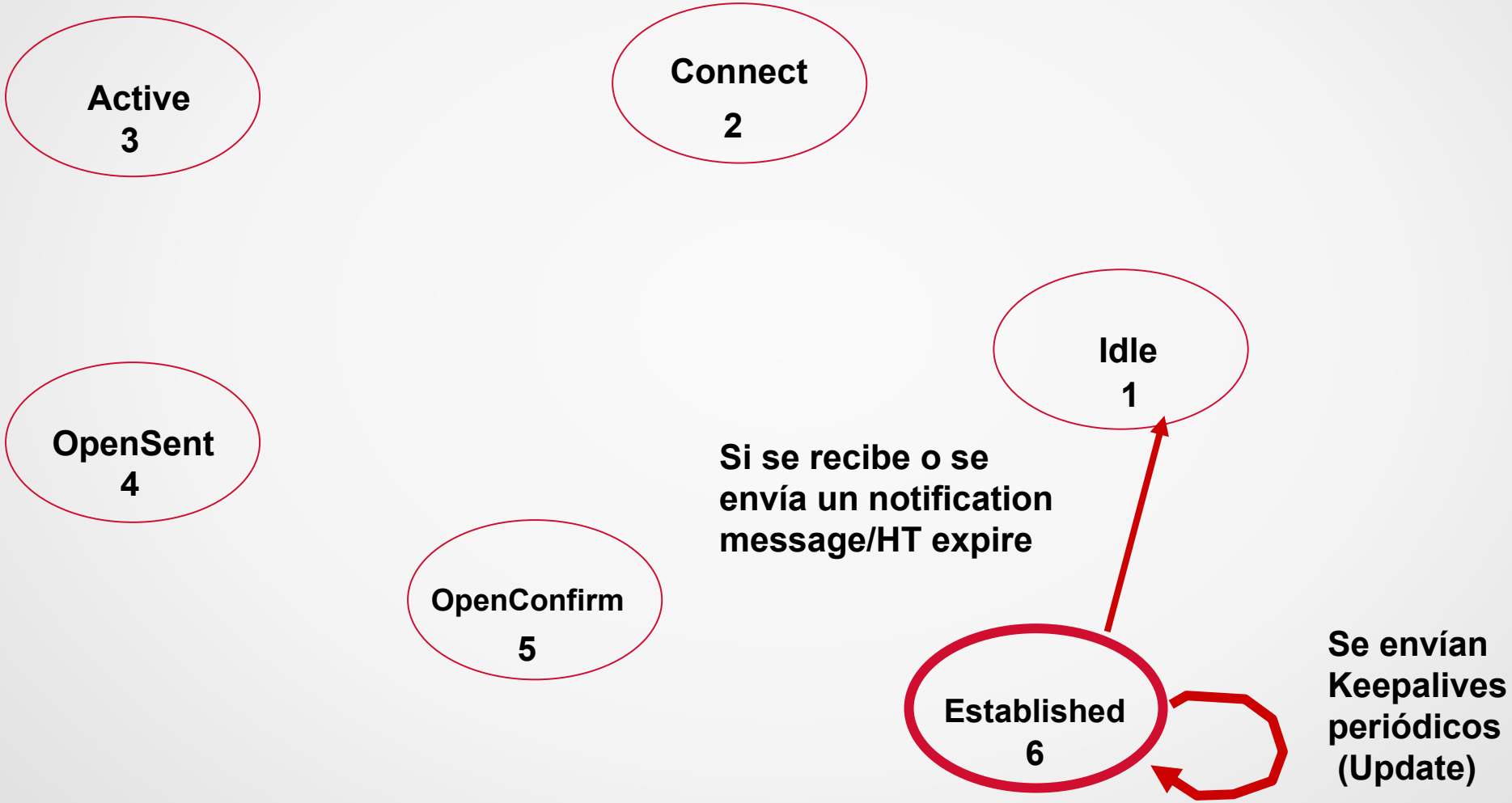
Inicio de una sesión BGP



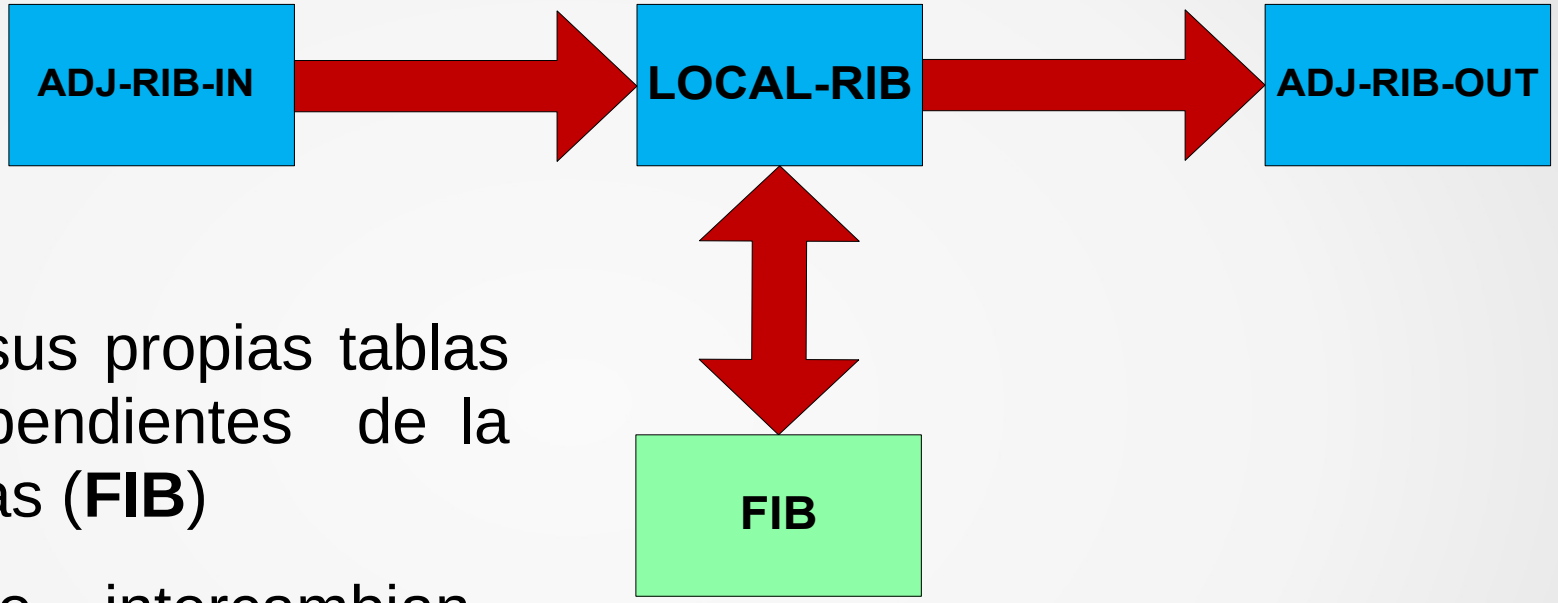
Inicio de una sesión BGP



Inicio de una sesión BGP



Tablas en BGP



- BGP tiene sus propias tablas (**RIB**), independientes de la tabla de rutas (**FIB**)
- Las tablas se intercambian entre peers al inicio de la sesión. Luego sólo actualizaciones incrementales (**Updates**)

- **FIB:** Forwarding Information Base
- **RIB:** Routing Information Base

RIB (Routing Information Base)

- **Tablas de rutas con sus atributos** (AS_PATH, etc)
- Conceptualmente tres conjuntos de tablas:

  Adj-RIB-In: Rutas recibidas de un vecino. Tantas tablas como vecinos exista

  Loc-RIB: Información local (lo que utilizo, luego de aplicarle políticas a las RIB-In)

  Adj-RIB-Out: Rutas para ser enviadas a los vecinos (una por vecino)

- La política se realiza a la entrada entre **Adj-RIB-In y Loc-RIB,** y a la salida entre **Loc-RIB** y **Adj-RIB-Out.**
- Dos entradas son diferentes si difieren en un atributo, por más que refieran al mismo prefijo.

Operación General – Incorporar Anuncios

- Para un mismo **prefijo**, un enrutador aprende múltiples caminos (**paths**) via BGP interno o externo
- En **LOCAL-RIB** están todas las redes (y respectivos atributos) que pasaron el filtro entrante desde las **ADJ-RIB-IN**
- El proceso escoge **“EL MEJOR”** camino en la **LOCAL-RIB** y lo **“instala”** en su tabla de forwarding (**FIB**)
- El algoritmo de selección de mejor camino es conocido (fijo).
- El protocolo es susceptible a Políticas que se aplican para influenciar justamente la selección del **“MEJOR”** camino

Operación General – Anuncios

- Se anuncia **SOLO el MEJOR** camino de la **LOCAL-RIB** a cada destino (recordar que es el que usa la FIB)
- **RFC 7911:** Advertisment Multiple Path in BGP
- **Reglas BGP (sin filtros):**
  - Lo que un enrutador aprende por **EBGP** lo anuncia a **TODOS** sus peers
  - Lo que un enrutador aprende por **IBGP** lo anuncia sólo a sus peers **EBGP**
    - Finalidad: Evitar Loops
    - Fuerza Full-Mesh
- El proceso de anuncio puede ser influenciado por políticas eligiendo que anuncio pasa (**ADJ-RIB-OUT**)

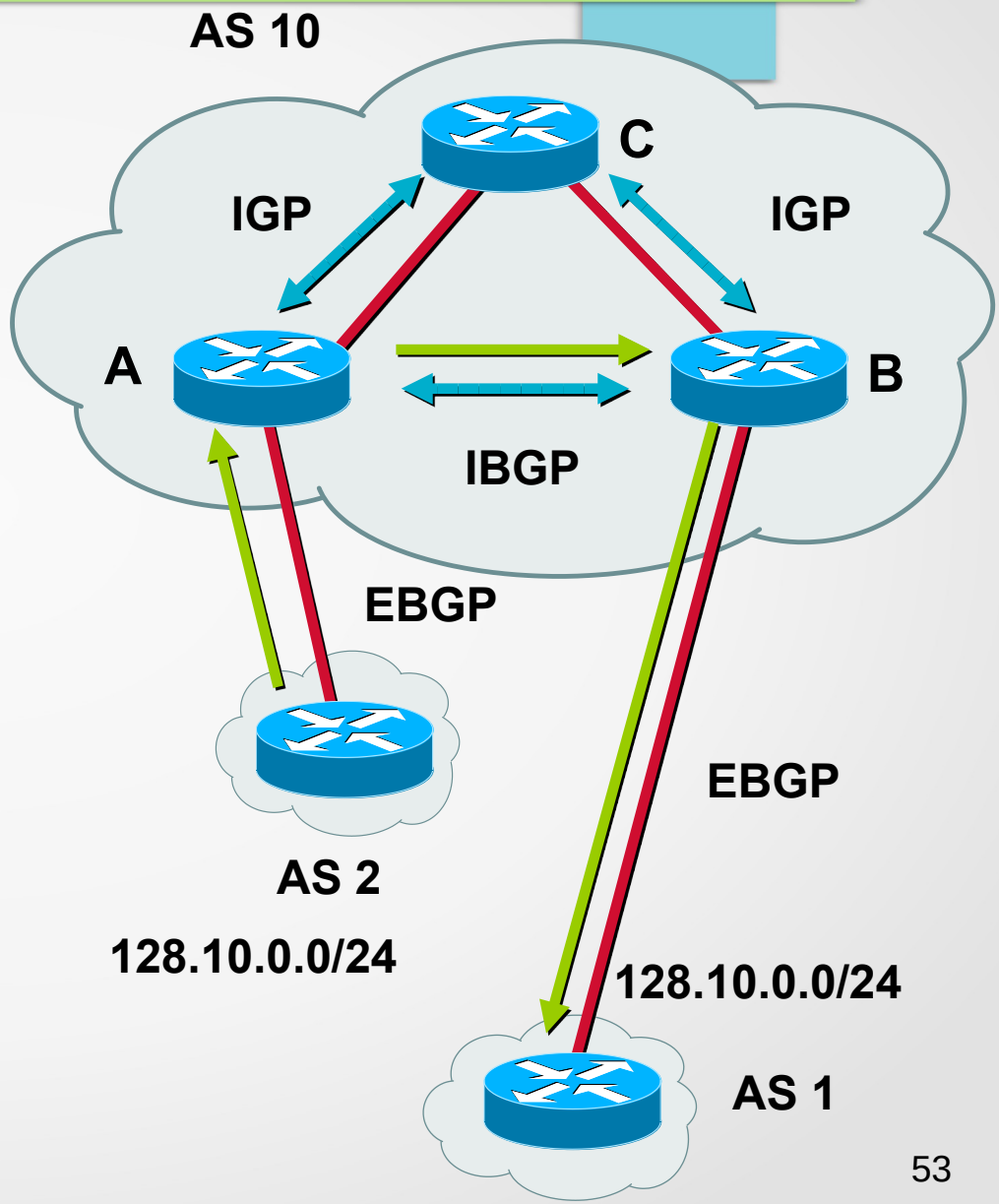
Sincronización (1)

Regla: En los AS Multihomed de tránsito NO usar ni anunciar un prefijo hasta que una ruta que lo contenga haya sido aprendida por IGP

- Asegura la consistencia de la información en el interior del AS
- Evita “black holes” dentro del AS
- Se trata de buscar topologías que permitan deshabilitar la sincronización

Sincronización (2)

- A y B peers **IBGP**
- C **NO** lo es

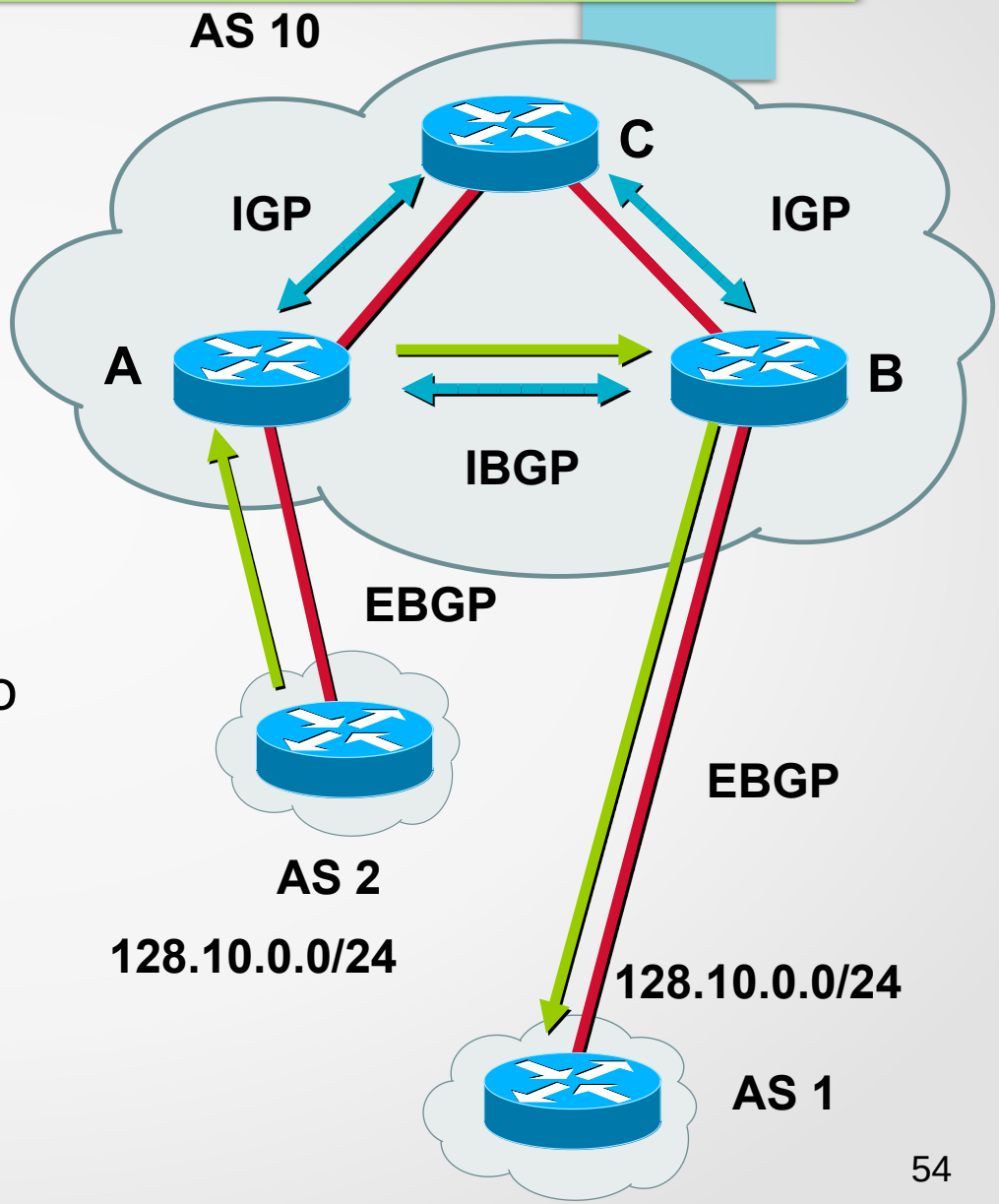


➢ Si la sincronización está **apagada** y el **IGP** no propagó la ruta a **128.10.0.0**:

- B intenta alcanzar la red **128.10.0.0** via C
- C descarta los paquetes ya que no conoce una ruta a la 128.10.0.0
- El AS 1 recibe un anuncio al cual jamás podrá llegar

Sincronización (3)

- A y B peers **IBGP**
- C **NO** lo es



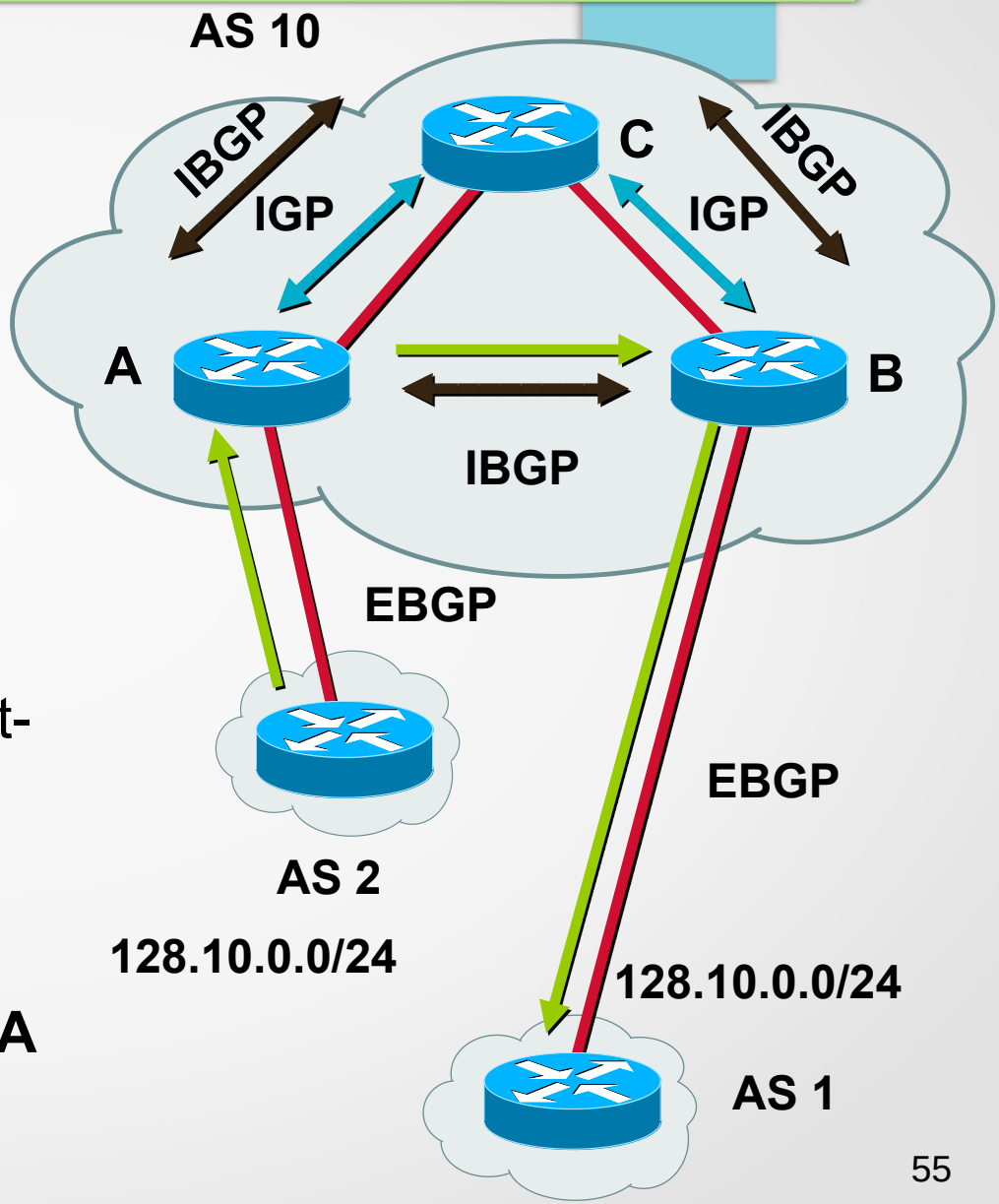
➢ Si la sincronización está **activada**:

- B no anuncia la red al AS 1 hasta no conocerla por IGP
- C debe conocer una ruta a la **128.10.0.0/24** via IGP -> se debe redistribuir la red aprendida por BGP en IGP en el enrutador A

(No recomendable)

Sincronización (4) – iBGP FullMesh

- A,B y C peers **IBGP**



➢ Si la sincronización está **apagada** y el **IGP** no propagó la ruta a **128.10.0.0**:

- B y C conocen por **iBGP** como alcanzar la red **128.10.0.0/24** por next-hop WAN contra el AS2.
- B y C conocen como alcanzar la red WAN AS2 (IGP, IBGP, estática)
- B y C saben que tienen que **dirigir a A** para alcanzar **128.10.0.0/24**.

Sincronización (5)

- El uso de Sincronización implica redistribuir BGP en el IGP, las inestabilidades de BGP se trasladan al IGP (mayor volumen de mensajes, y ejecución SPF).

  Actualmente desaconsejado (deshabilitado por defecto)

- Alternativas para evitar redistribuir al IGP en el enrutador A:

  - Deshabilitar la sincronización y correr BGP en todos los enrutadores del AS (al menos todos en el camino entre otros AS) **Full-Mesh-BGP**
  - **MPLS** (se verá luego)

- Utilizar política en caso de necesitar hacer la redistribución:

  - Hacerlo sólo para las redes de interés!!

- Es posible deshabilitar si nuestro AS no oficiará de tránsito hacia otros AS

Agenda (3)

- Conceptos Fundamentales de BGP
- Análisis del protocolo (BGP-4)
- Atributos de BGP y políticas de control
- IBGP mesh y Alternativas
- Sumarización y anuncios (CIDR)
- Damping y problemas de convergencia
- Extensiones Multiprotocolo
- Seguridad de BGP
- Salidas reales y datos de actualidad
- Ejemplo y consideraciones prácticas

Atributos de BGP (1)

- Los atributos de bgp son los que permiten tomar decisiones “**complejas**” sobre los caminos
- **4 Categorías:**
  - Well-Known Mandatory (Obligatorios, bien conocidos). Deben ser reconocidos por todas las implementaciones de BGP y deben estar presentes en todo mensaje de UPDATE
  - Well-Known Discretionary (bien conocidos, opcionales). Deben ser reconocidos por todas las implementaciones de BGP, pero pueden o no aparecer en un mensaje de UPDATE

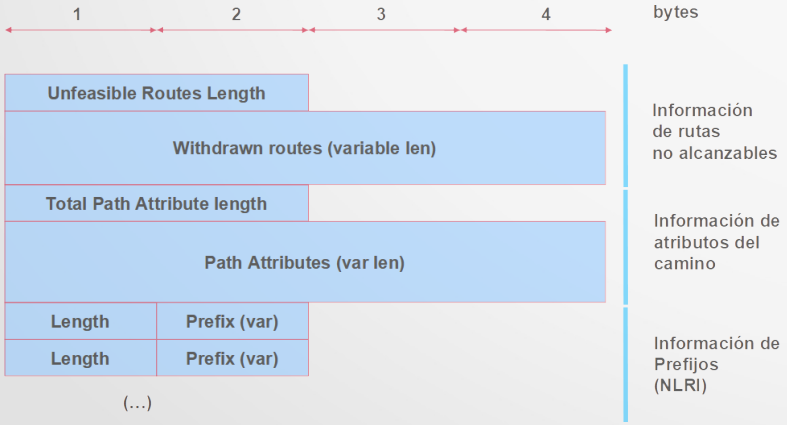
Atributos de BGP (2)

- Optional Transitive (opcional, transitivo): no se requiere que sean soportados por todas las implementaciones de BGP. Deben ser reenviados aún en el caso de no ser soportados

- Optional Nontransitive (opcional, no transitivo): no se requiere que sean soportados por todas las implementaciones de BGP. En caso de no ser reconocido, se ignora y no se pasa a otros vecinos BGP

Atributos de BGP (3)

Attribute flags	Tipo	Largo	Valor



Flags:

- Bit 0: Opcional = 1/bien-conocido = 0
- Bit 1: Transitivo
- Bit 2: Parcial
- Bit 3: Largo Extendido
- Bit 4-7: deben ser 0

Algunos Atributos de BGP (Tipo)

WKM

- **Next_hop**
- **AS_path**
- **Origin**

WKD

- **Local preference**
- **Atomic aggregate**

OT

- **Aggregator**
- **Community**

ONT

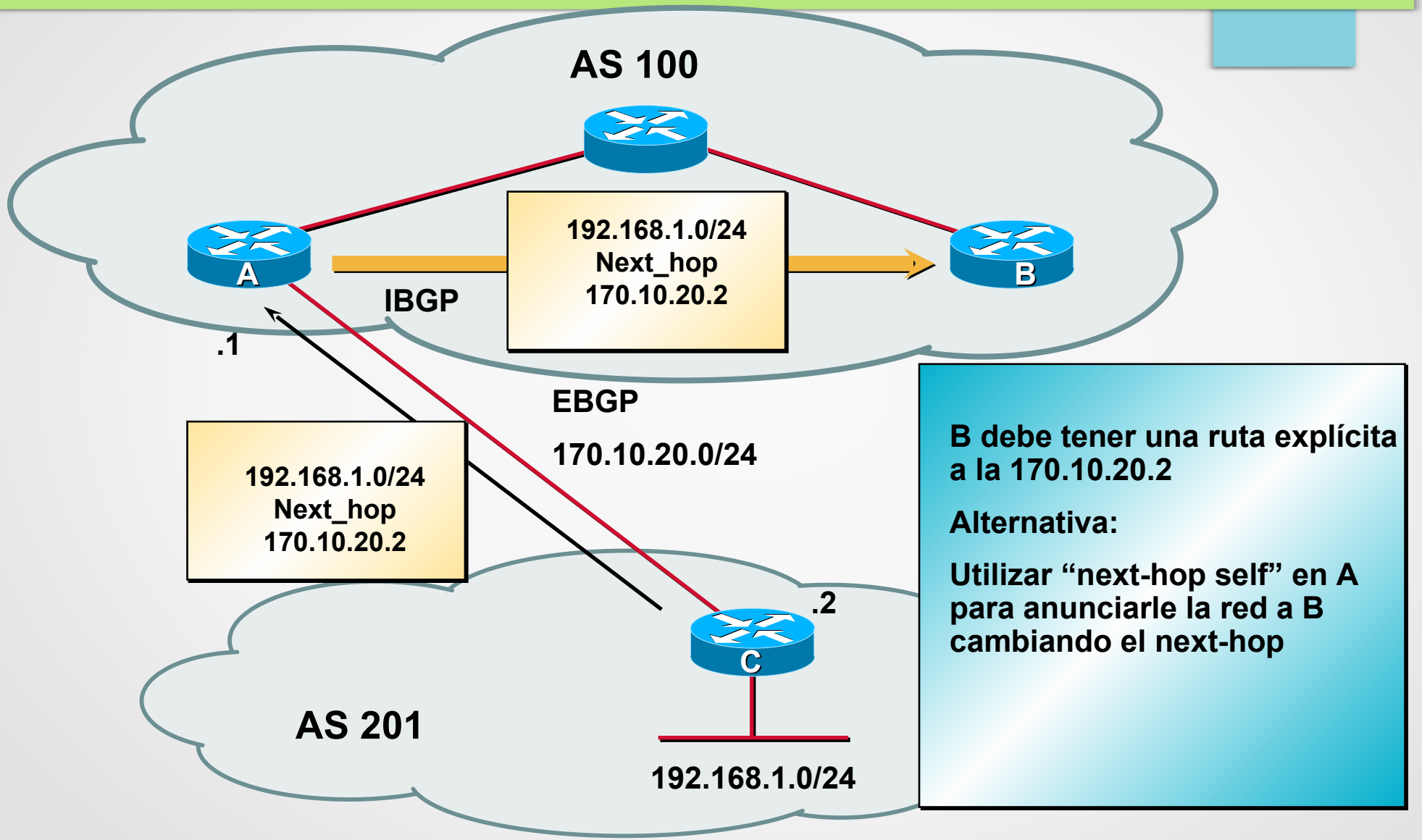
- **Multi Exit Discriminator (MED)**
- **Multiprotocol Reachable NLRI**
- **Cluster_List**
- **otros**

¡¡¡Extensiones!!!

Atributo - NEXT_HOP (1)

- **NEXT_HOP** indica la IP del vecino al cual enviarle los paquetes para alcanzar una red
- Varía según IBGP o EBGP:
  - Para EBGP: dirección IP del peer que anunció la ruta (excepción posible en medios Multiacceso, p. ej. Ethernet)
  - Para IBGP: Redes que fueron inyectadas al AS vía EBGP tienen como NEXT_HOP el anunciado por EBGP y se acarrea inalterado en IBGP
- Si no se tiene una ruta específica a la IP del NEXT_HOP no se debería usar la ruta. **No alcanza con una ruta por defecto!!!**

Atributo - NEXT_HOP (2)



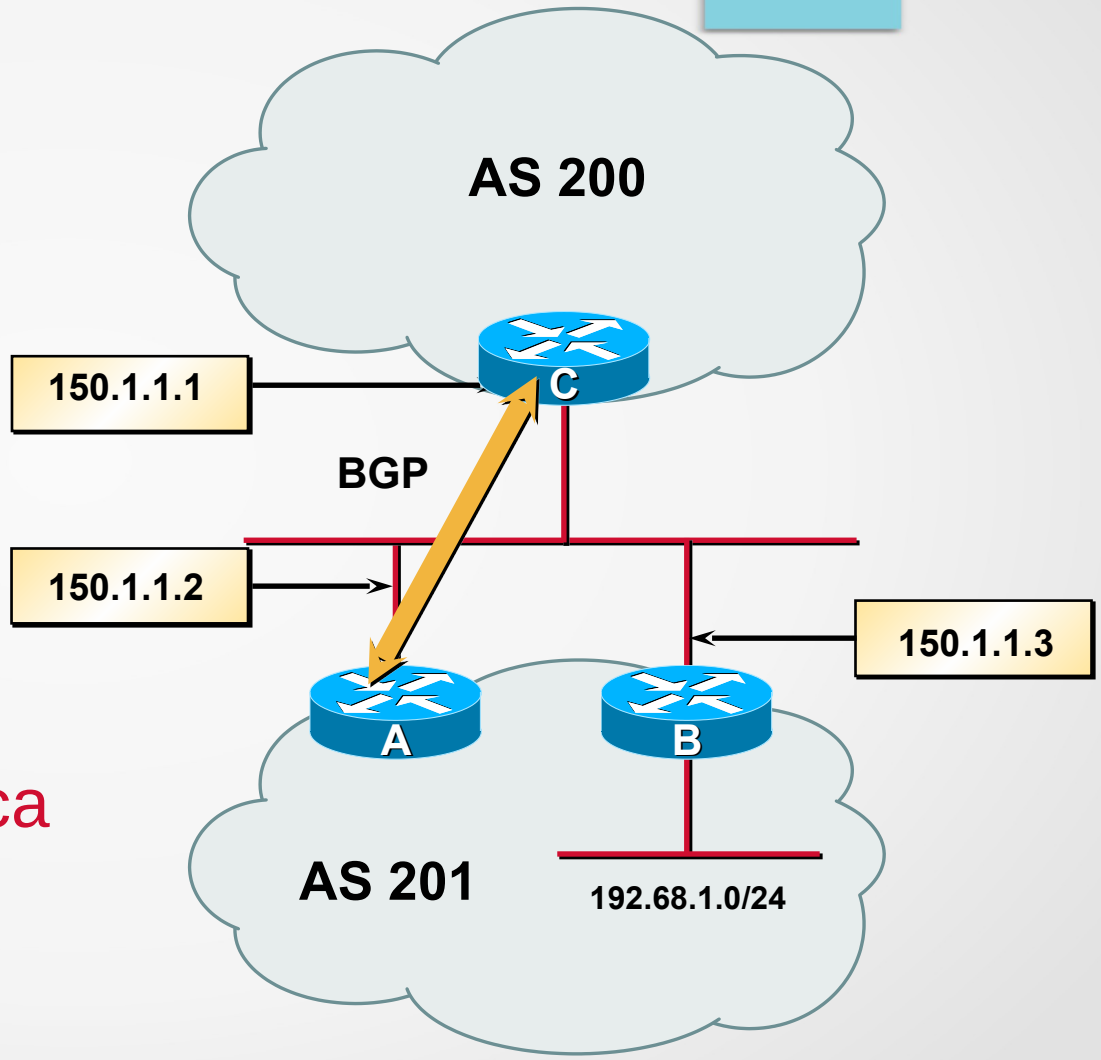
B debe tener una ruta explícita a la 170.10.20.2

Alternativa:

Utilizar "next-hop self" en A para anunciarle la red a B cambiando el next-hop

Third-Party NEXT_HOP en un medio MA

- **Ejemplo:**
  - ➢A y B están en el mismo AS
  - ➢A le anuncia a C
    la red 192.68.1.0/24
    con NEXT_HOP 150.1.1.3.
- Es más eficiente!
- Se implementa de forma explícita mediante una política que modifica el NEXT_HOP

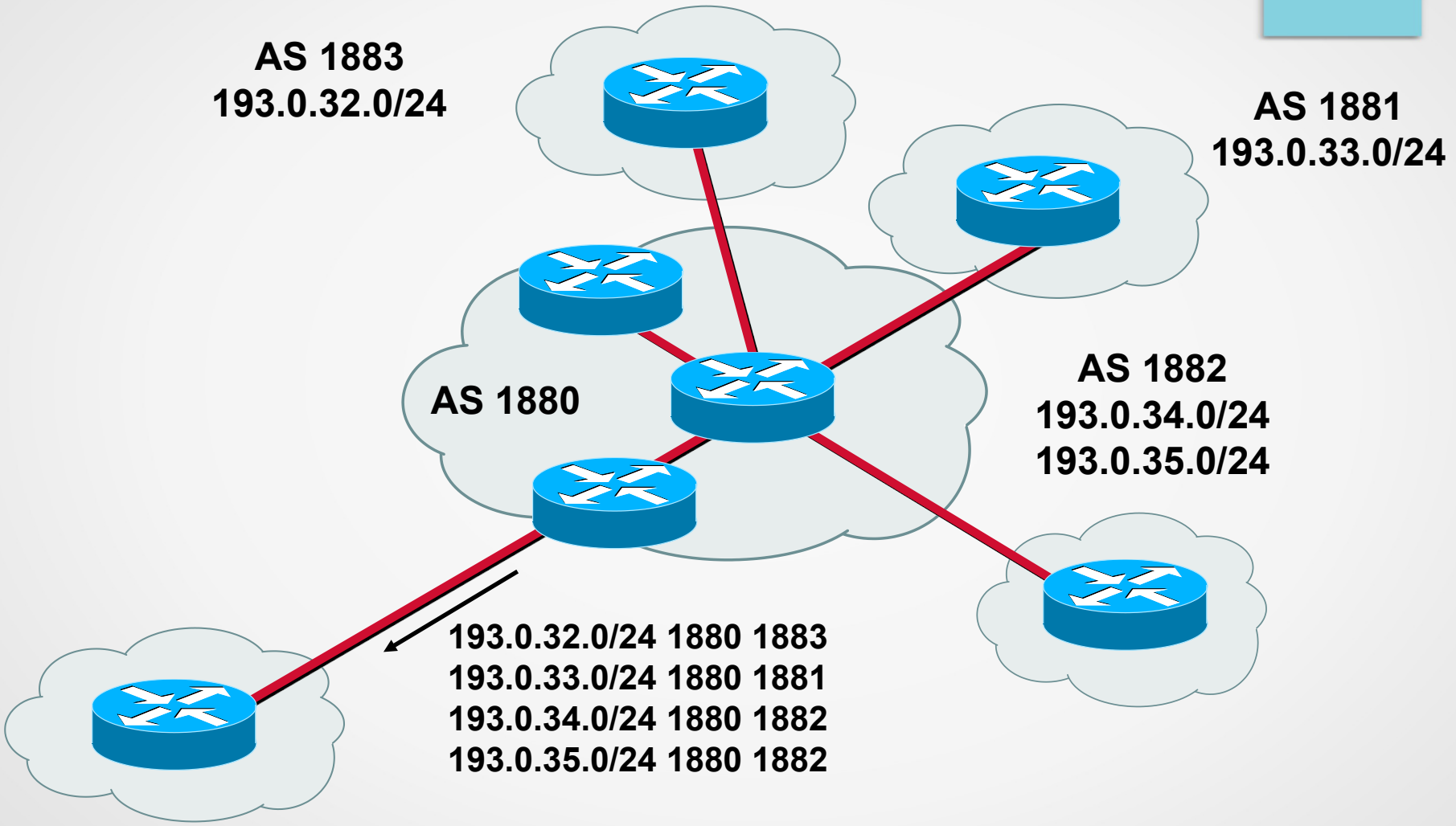


MA – Multi Access Network

Atributo - AS_PATH (1)

- Lista de Sistemas Autónomos (AS) que un anuncio ha atravesado
- **evita loops!!!**
- Es una secuencia de **AS_PATH segment**
- Dos posibles componentes (**segment type**): AS_SEQUENCE y AS_SET
- **AS-SET:** {1881 1882 1883}
- Se usa como uno de los criterios para la elección del mejor camino (se prefiere un **AS_PATH** más corto)

Uso del AS-Set (**sumarización**)



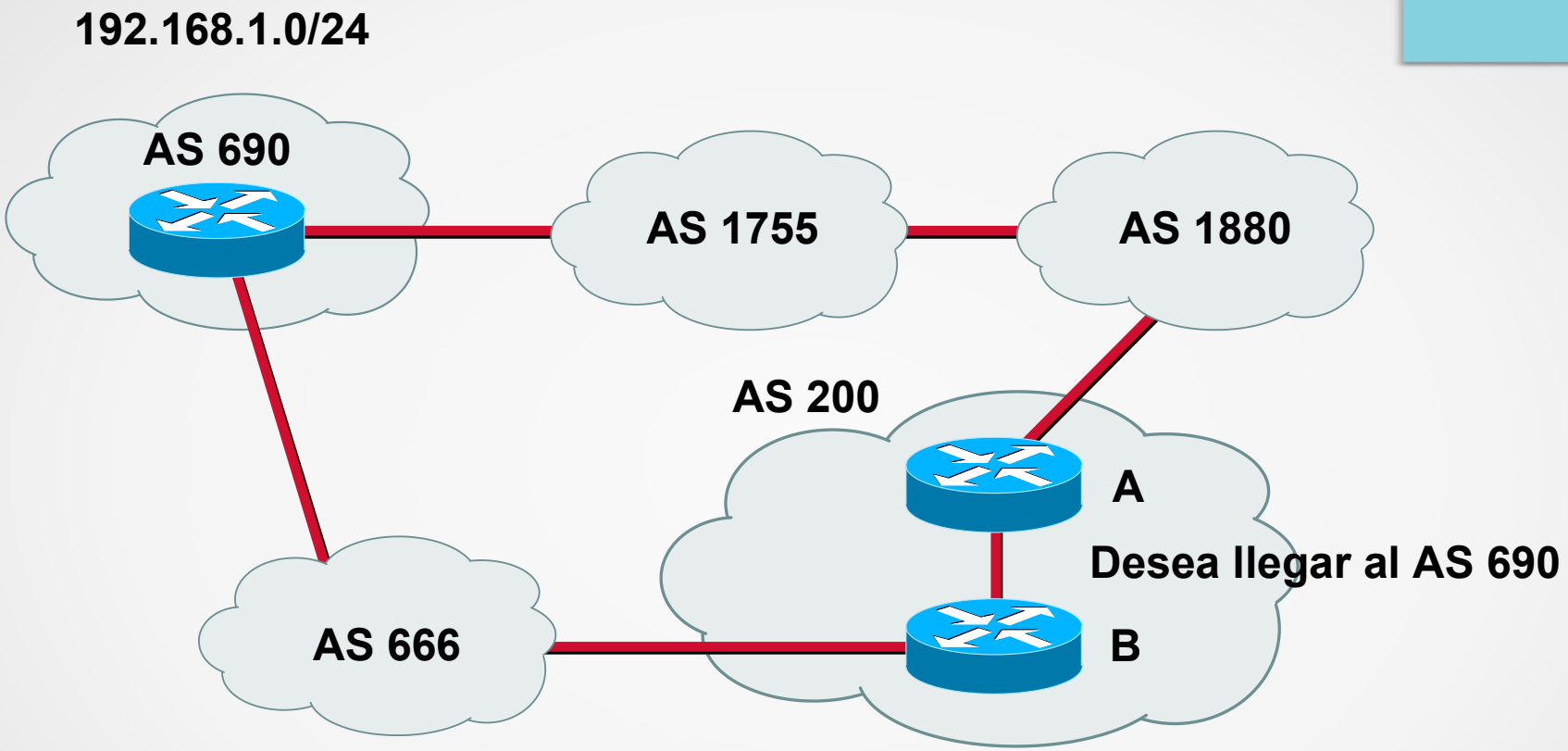
193.0.32.0/22 1880 {1881 1882 1883}

Uso del AS-Set (**sumarización**)

- El **AS-SET** se utiliza para indicar los sistemas autónomos que participaron en la formación del agregado
- La realidad es que esta forma de sumarización se utiliza muy poco
- Router ID en el atributo **AGGREGATOR**
- Se permite eliminar el AS_SET que indica como se formó la sumarización, en este caso debe agregarse el atributo **ATOMIC_AGGREGATE.**
- RFC 6472 recomienda **NO** utilizar AS_SET

  Año 2017, del orden de 13.378 entradas con AS-SET en la tabla global, 420 prefijos diferentes, 199 AS_SET diferentes

AS_PATH – Selección de Camino



- El AS 200 conocerá la red 192.168.1.0/24:

  192.168.1.0/24 **1880 1755 690**

  192.168.1.0/24 **666 690** **<= Preferible (short AS_PATH length)!!!**

Atributo - ORIGIN

- Provee información acerca de cómo se generó la ruta:
- **IGP**

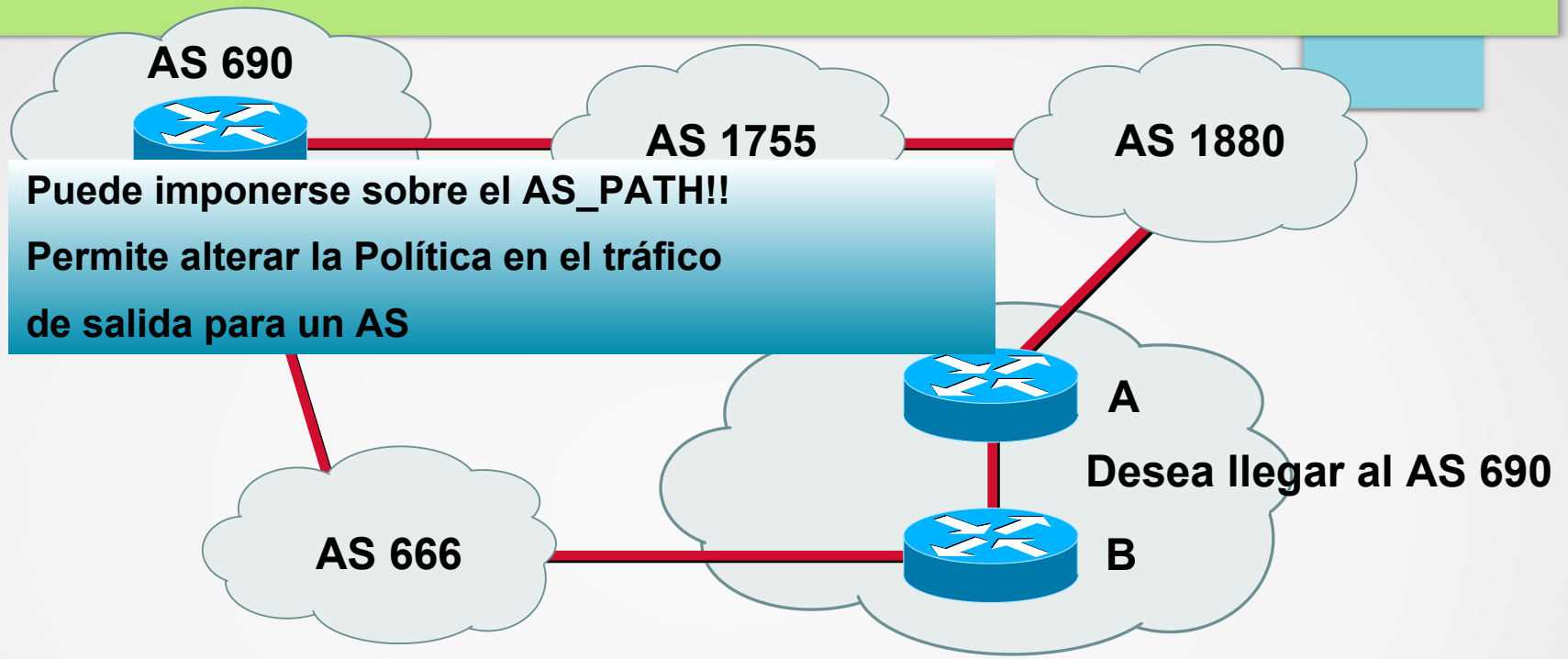
  La ruta se generó en el proceso BGP (configuración)
- **EGP**

  Generado desde EGP (obsoleto)
- **Incomplete**

  La ruta se aprendió por otro mecanismo. Por ejemplo surge de redistribuir rutas IGP en BGP

  Ej. más común en BGP en internet: redistribución de estáticas

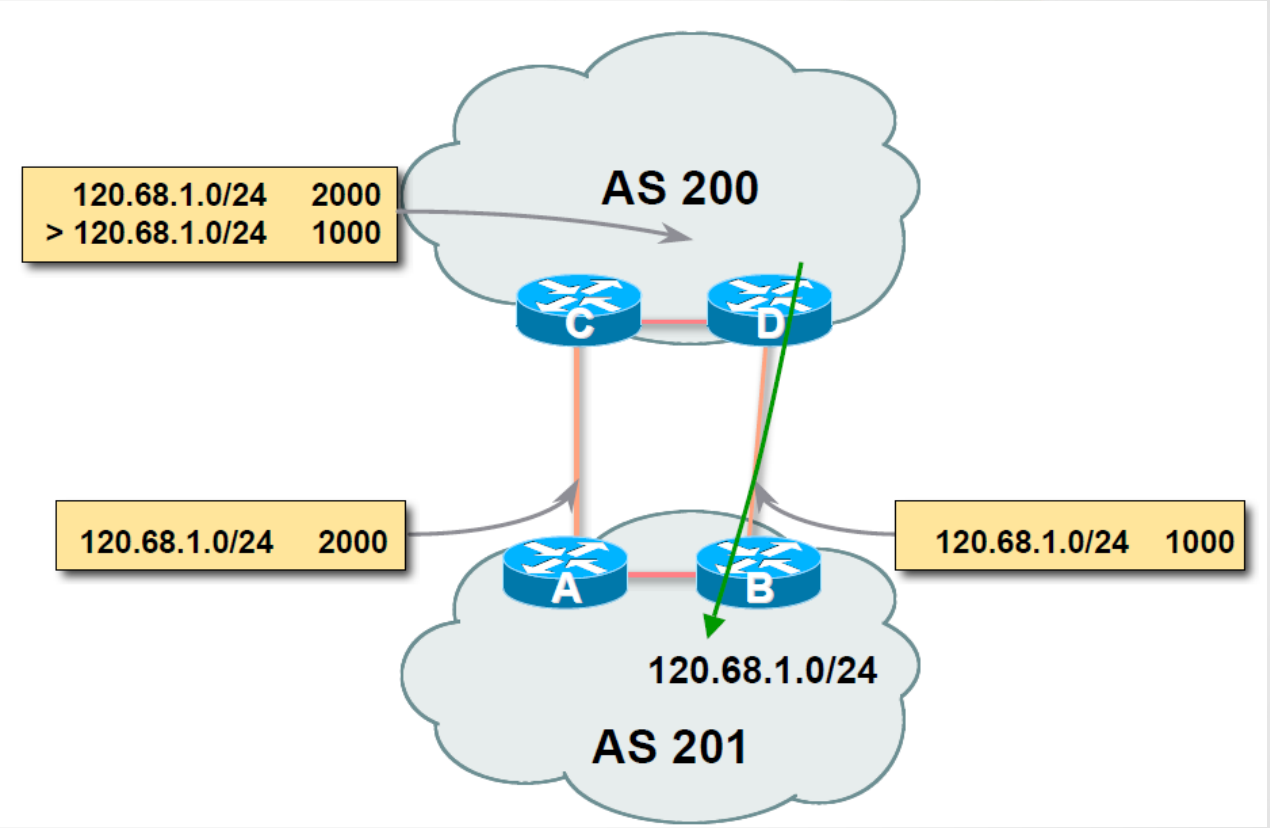
Atributo - **Local Preference** – Influenciando la Salida



- Cuando existen múltiples caminos para el mismo destino, el atributo de **Local Preference** indica el camino preferido administrativamente. Define el punto de salida de mi red.
- El camino con la mayor preferencia local es el elegido
- El atributo **Local Preference** sólo tiene sentido "**local**", se propaga en el interior del AS por **IBGP**, no por **EBGP**

Atributo - Multi Exit Discriminator (MED)

- Para influenciar el camino de vuelta (tráfico entrante al AS).
- Se puede usar para **discriminar entre múltiples caminos al mismo AS.**
- No se propaga a otros vecinos.
- **Limitado** en principio a **múltiples enlaces con un mismo AS.**
- Se prefiere el camino de menor MED. Suele ser llamado “métrica”.



Proceso de selección del mejor camino (1)

1. **No considerar** un prefijo **IBGP** hasta no estar sincronizado (cuando la sincronización está habilitada)
2. **No considerar** un prefijo si no existe una ruta al **next_hop** (o si al agregar el prefijo se genera un loop de resolución)

   200.40.30.0/24 next 10.1.1.1

   10.1.1.1/32 next 172.0.0.1 (cuando falla esta segunda búsqueda)
3. **Preferir** la ruta con mayor **Local Preference** (global dentro del AS)

Proceso de selección del mejor camino(2)

4. **Si** la ruta **NO** fue localmente originada (network o redistributed), elegir el **AS_PATH** de menor largo
5. **Si** los **AS_PATH** son de igual largo, elegir el prefijo con el **menor ORIGIN** type:

   IGP sobre EGP , y EGP sobre Incomplete

Proceso de selección del mejor camino(3)

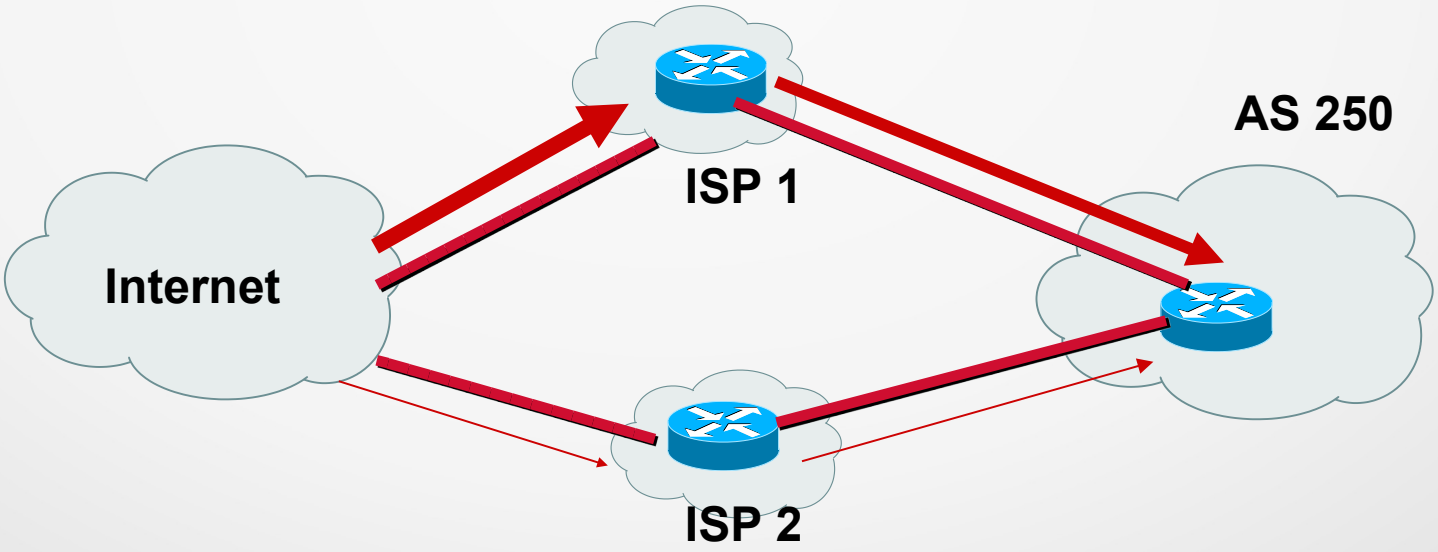
6. **Si** los orígenes son los mismos, y el AS vecino es el mismo, elegir el que tenga el **menor atributo MED** (ojo MED es Opcional!!!)
7. **Preferir** un anuncio externo (eBGP) antes que uno interno (iBGP)
8. **Si** no existe anuncio externo, Preferir el path a través del neighbor **más próximo según el IGP**
9. **Preferir** el path con el menor BGP router id **(desempate,** tiro una moneda!!!**)**

Influenciando el tráfico saliente

- **Observar** que la preferencia local es el primer atributo que se verifica
- Si recibo el mismo prefijo de más de un vecino, puedo elegir el camino de salida fijando un valor mayor de **LOCAL-PREFERENCE**
- Fozando el valor de **LOCAL-PREFERENCE** controlo por donde sale el tráfico para un destino (prefijo).

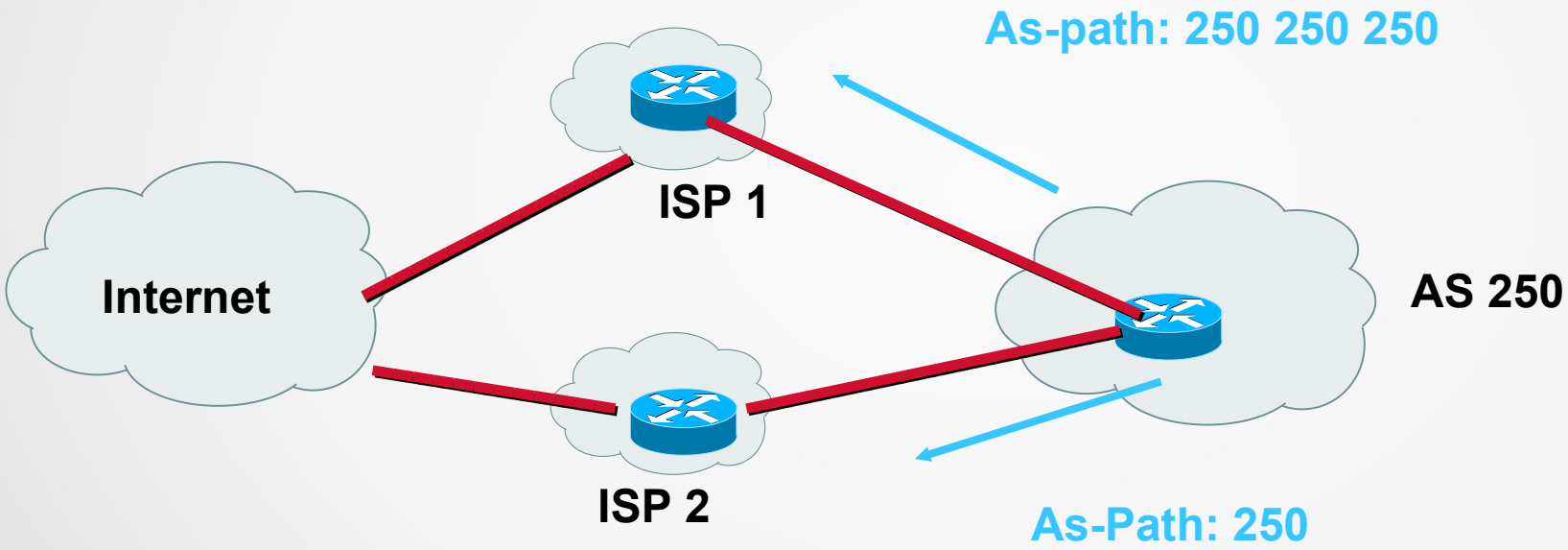
Influenciando el tráfico entrante

- **MED.** Muy limitado el escenario de aplicación
- Práctica habitual: hacer **prepends** al AS-PATH (agregar copias de mi número de AS)
- Problema ejemplo: 80% del tráfico viene por el ISP 1:



Influenciando el tráfico entrante: "Solución" prepends

Los sistemas autónomos remotos verán un camino **más largo** por ISP1



Podemos decidir prefijo a prefijo

```
route-map rmap-bajarprio permit 10
  match ip address prefijoA
  set as-path prepend 250 250
```

Influenciando el tráfico entrante: **Otras soluciones**

- **Influenciando:** Observar que no es posible garantizar 100% que el tráfico destinado a un prefijo llegue por un ISP. ¿Porqué?
- No publicar algunos prefijos por determinados enlaces
  - **Problema:** pierdo respaldo
- Publicar anuncios más específicos por el enlace descargado (10.20.0.0/24 vs 10.20.0.0/20)
- En general todas las soluciones son "**ad-hoc**", analizando el pasado, estimando una corrección y luego verificando el efecto.

Atributo - Comunidades

- Atributo “**Community**”
- Opcional, transitivo
- Una comunidad **es un grupo de destinos** (prefijos) **que comparten una propiedad común**
- Usado para agrupar destinos y aplicar una política común
- Un prefijo puede pertenecer a varias comunidades
- En muchos equipos no se propaga por defecto

  neighbor *ip-address* send-community (en Cisco)

Atributo - Comunidades (2)

- 32 bits
- Recomendación: número de AS en los primeros 16 bits
- set community *AS:community*
- Bien conocidas:

internet	todos los vecinos BGP
no-export	no anunciar a vecinos eBGP
no-advertise	no anunciar a vecinos BGP
local-AS	
no-peering	

- Definidas por cada ISP, el cual le asigna un significado, para luego simplifica las políticas en BGP. Ejemplo prefijos de Europa, información de sitio o Null route para RTBH (Remotely Triggered Black Hole)

Políticas de Control (1)

- **Filtrado de rutas**

  Entrantes o salientes

  Al **filtrar** los anuncios **entrantes**, estoy definiendo el camino del **tráfico saliente**

  Al **filtrar** los anuncios **salientes**, estoy "definiendo/**influenciando**" por donde **vendrá el tráfico** hacia mi AS, y si permito o no tránsito

- **Manipulación de atributos**

  Puedo cambiar los valores de los atributos para influenciar el proceso de decisión (en mi AS o en los vecinos)

- Recordar el Proceso de Selección de mejor Camino: "Juego" que dado una reglas bien conocidas, para cada prefijo puedo manipular atributos mediante políticas, para luego filtrar publicaciones saliente o luego "caer" en determinado lugar del proceso de decisión de BGP en otro equipo.

Políticas de Control (2)

Tres pasos:

1. Identificar las rutas o prefijos (y sus atributos)
2. Permitir o negar las rutas
3. Manipular los atributos

Políticas de Control (3)

Listas de Prefijos

- Por peer BGP
- Basadas en prefijos
- Tanto entrantes como salientes
- Ejemplo: no anunciar al peer 200.108.192.1 el prefijo 172.16.10.128/25

Políticas de Control (4)

Listas de filtrado

- Permite filtrar rutas basándose en el AS_PATH u otros atributos
- Tanto entrantes como salientes
- Basadas en atributos
- Ejemplo: no permitir anuncios cuyo AS_PATH comience con el AS 100

Políticas de control (5)

- Para políticas más complejas y manipulación de atributos, en Cisco se utilizan route-maps

route-map pref permit **10**

match as-path 100

set local-preference 250

route-map pref permit **20**

match ip address 1

set local-preference 300

route-map pref permit 30

Agenda (4)

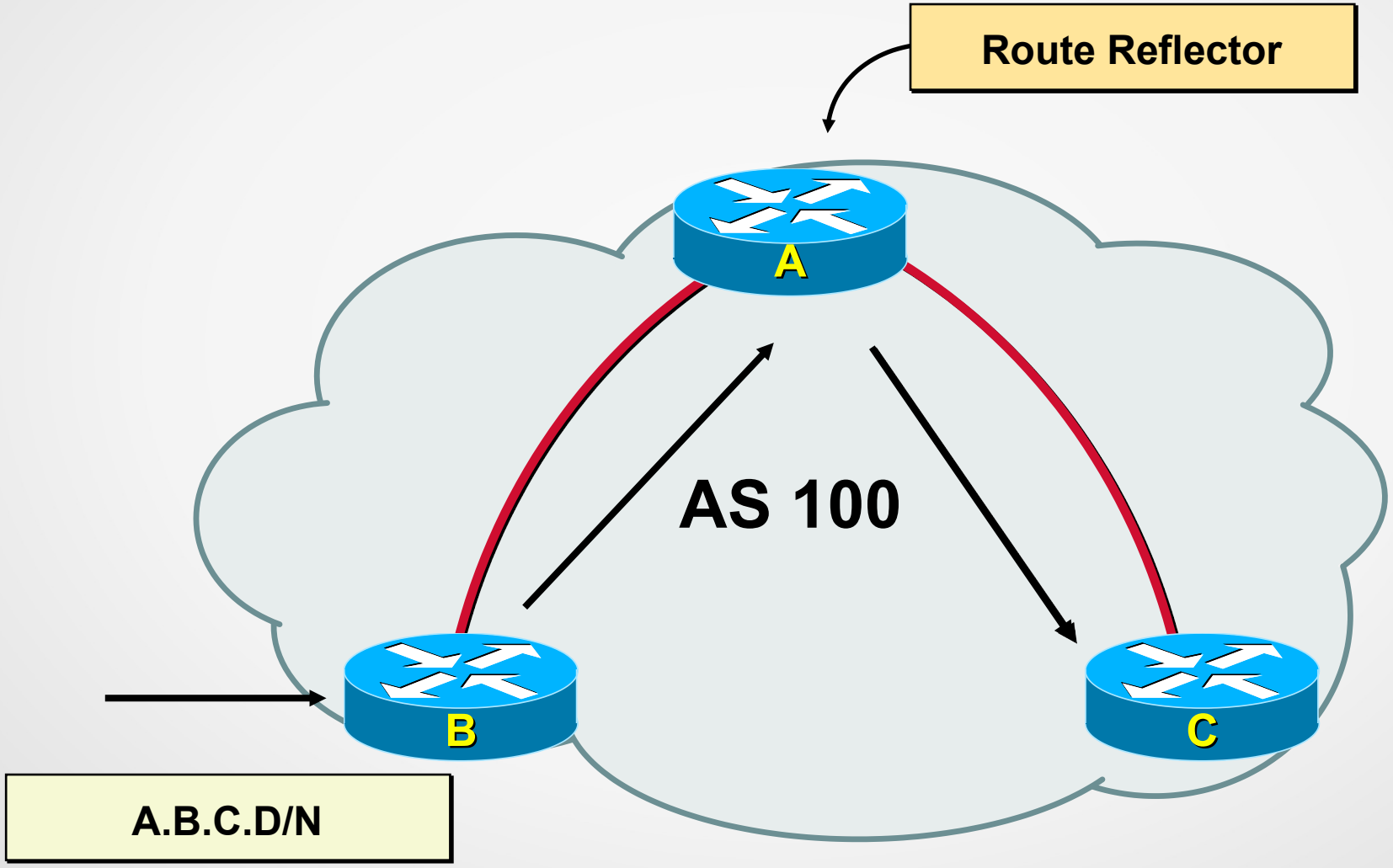
- Conceptos Fundamentales de BGP
- Análisis del protocolo (BGP-4)
- Atributos de BGP y políticas de control
- IBGP mesh y Alternativas
- Sumarización y anuncios (CIDR)
- Damping y problemas de convergencia
- Extensiones Multiprotocolo
- Seguridad de BGP
- Salidas reales y datos de actualidad
- Ejemplo y consideraciones prácticas

IBGP Mesh y soluciones

- **Recordar:** Buscamos "**apagar**" la sincronización
- En principio se precisa una sesión entre cada par de enrutadores hablando IBGP (**Full-Mesh**)

  Por la regla que impide propagar por IBGP lo aprendido por IBGP (para evitar loops)
- **No escala** (agregar un nodo implica n+1 cambios)
- Alternativas:
  - Reflectores de rutas (Route reflectors, **RR**)
  - Confederaciones

Funcionamiento con un Route Reflector



Route Reflector: Beneficios

- Evita el mesh IBGP
- Normalmente no altera el forwarding de los paquetes (**no se modifica el next-hop**)
- Pueden coexistir BGP peers normales
- Pueden configurarse múltiples **RR** por redundancia
- Puede haber una **jerarquía** de **RR** (varios niveles)
- Es relativamente fácil migrar de **mesh** a **RR**

Route Reflector: Definiciones

- **Route reflector** (RR): reflector de rutas
- **Cliente de reflector** (RRC): peer BGP que recibe rutas internas repetidas o reflejadas por un RR
- **RR Cluster**: uno o más RR y sus clientes (RRC)
- **Cluster ID**: identificación del cluster, importante cuando tengo más de un RR
- **No-cliente**: peer iBGP que no es RRC (iBGP normal)
- **Normal BGP peer**: no cliente, o externo

Route Reflector: Funcionamiento

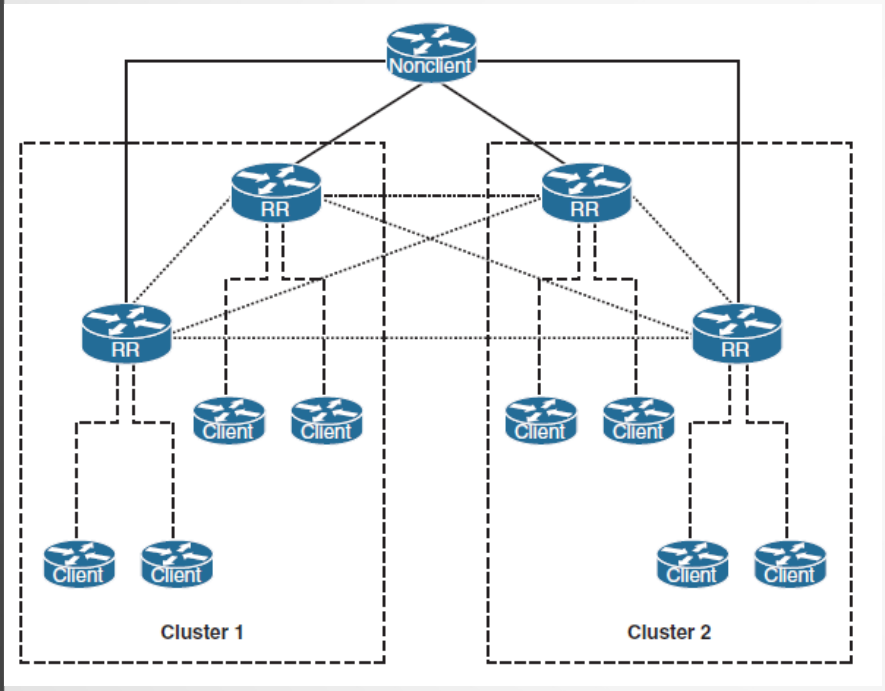
- **Recordar:** BGP solo **anuncia el mejor camino**, independientemente que aprenda varios, y esta ruta es la que “ingresa” a la **FIB** (utiliza lo que anuncia)
- **RR recibe anuncios** de clientes y de no-clientes
- RR elige el **mejor camino**
- Si el mejor camino lo **aprende de un cliente** => lo **refleja** tanto a sus **no-clientes**, como a sus **clientes** (excepto a quien originó el mensaje)
- Si el mejor camino lo **aprende de un no-cliente** => lo **refleja sólo a los clientes**
- Si el mejor camino lo **aprende de eBGP** , lo envía tanto a sus **no-clientes**, como a sus **clientes**

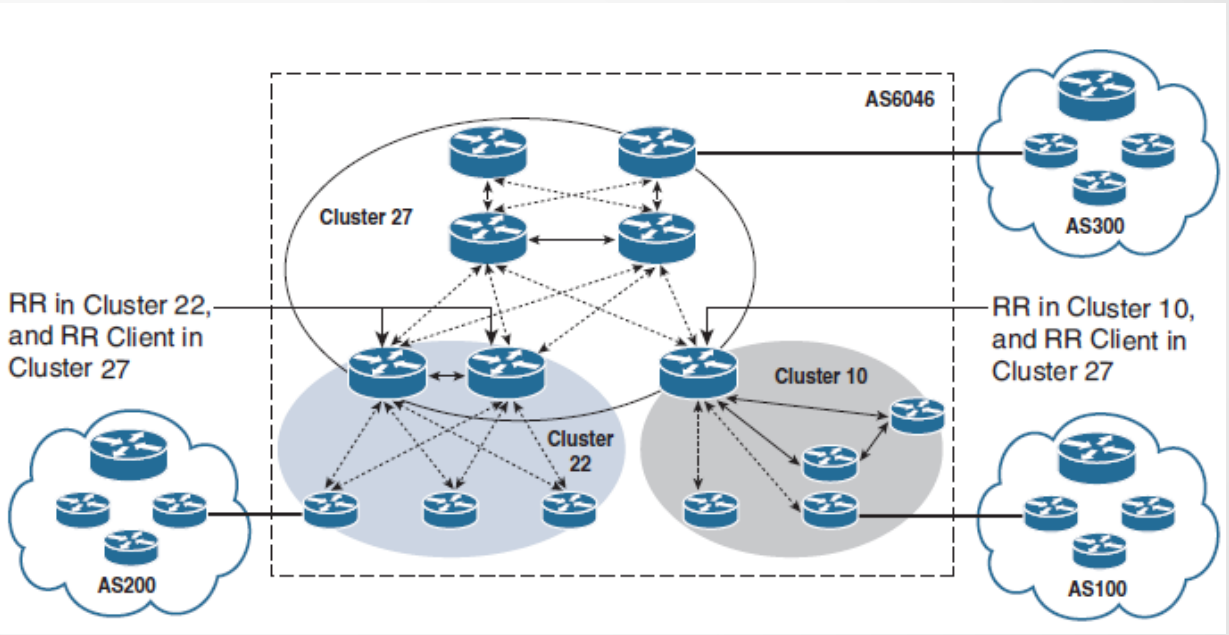
Route Reflector

- Permite dividir el sistema autónomo en múltiples clusters
- Al menos un **RR** y algunos clientes por cluster (RRC)
- **Route reflectors estarán fully-meshed**
- Nuevos atributos: **ORIGINATOR_ID** y **CLUSTER_LIST** opcionales no transitivos
- Cada vez que una ruta es reflejada se agrega el **CLUSTER_ID** (o **ROUTER_ID**) al **CLUSTER_LIST**
- **CLUSTER_LIST** permite detectar loops
- El primer RR que refleja una ruta asigna **ORIGINATOR_ID** = **ROUTER_ID**. Si ya existe no se sobrescribe

Route Reflector - Escalamiento

- Dos formas de escalar con RR: **Full-Mesh de RR o jerarquía de RR**

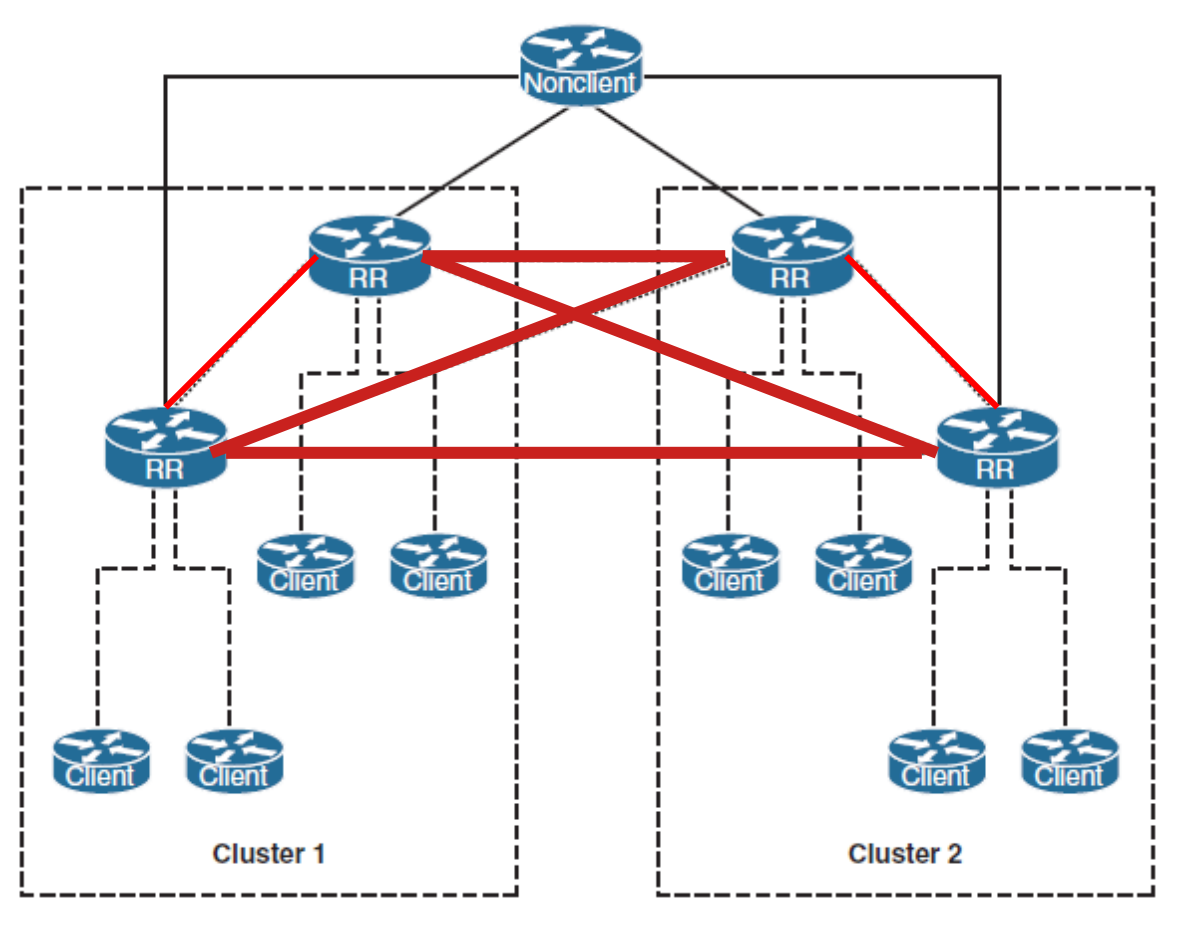




Full-Mesh entre RR de cada cluster

Jerarquía un RR de un Cluster de orden **N** es RRC de un Cluster de orden **N-1**

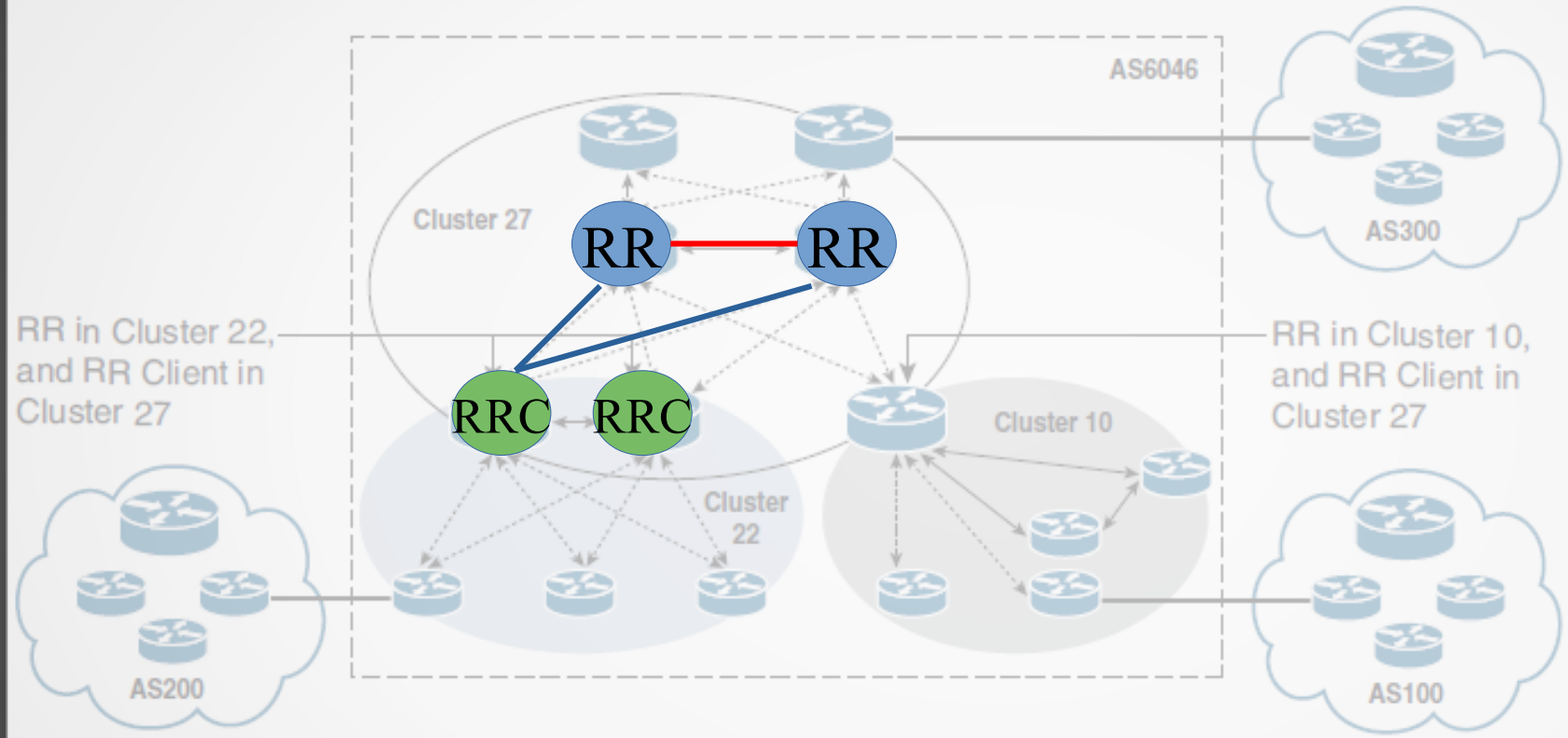
Route Reflector – Escalamiento Mesh RR



IBGP intra Cluster RR

IBGP inter Cluster RR

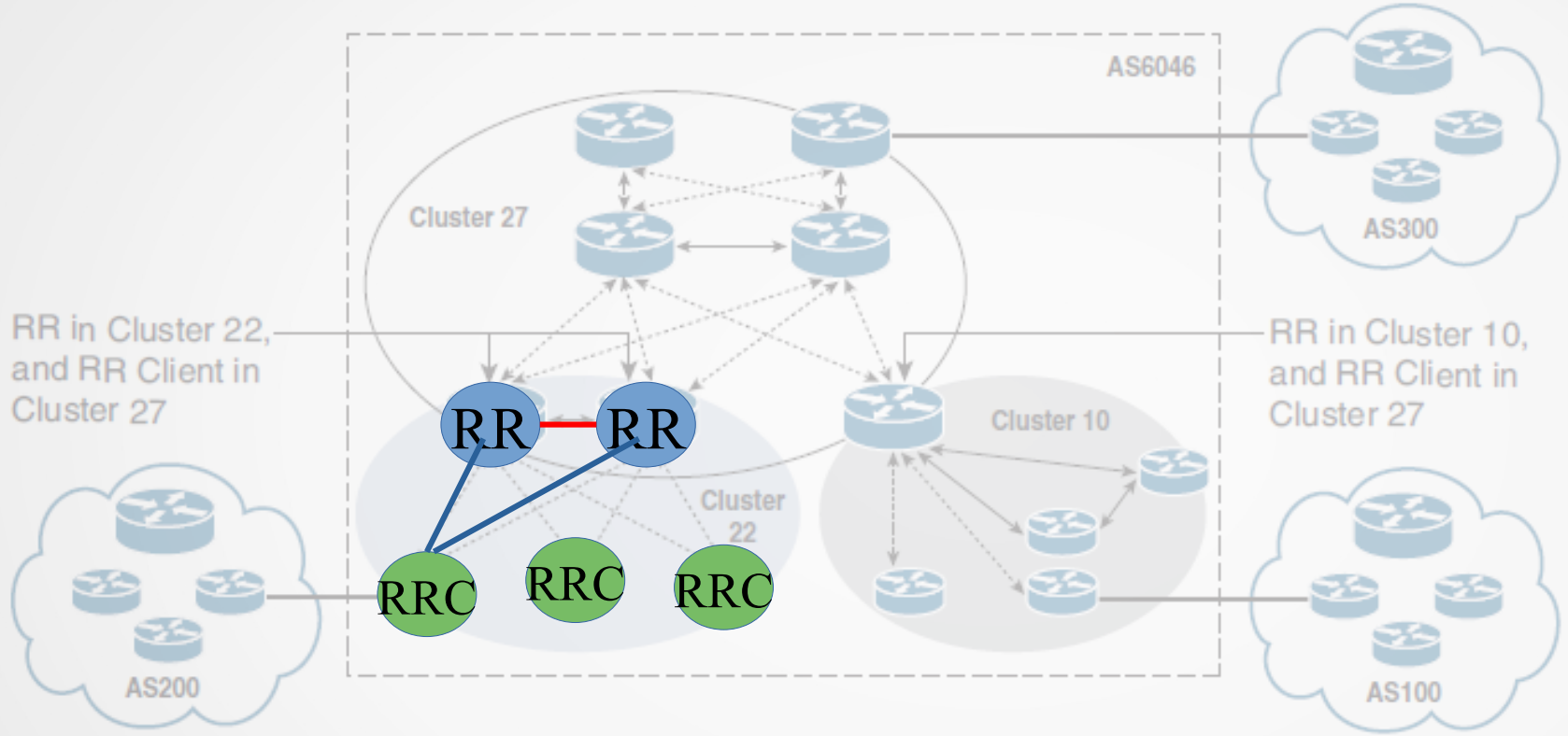
Route Reflector – Escalamiento Jerarquía RR



IBGP intra Cluster RR

IBGP RRC

Route Reflector – Escalamiento Jerarquía RR



La relación RRC-RR es por cada peer iBGP. Los router en azul, son RR de los router del cluster 22, pero son RRC de los RR del cluster 27.

Route Reflector - Recomendaciones

- **Seguir la topología física en la medida de lo posible**

  (los anuncios BGP y el forwarding pueden no coincidir, ejemplo si debo desempatar el mejor camino por métrica del IGP)

- **Evitar modificar los atributos** de las rutas reflejadas

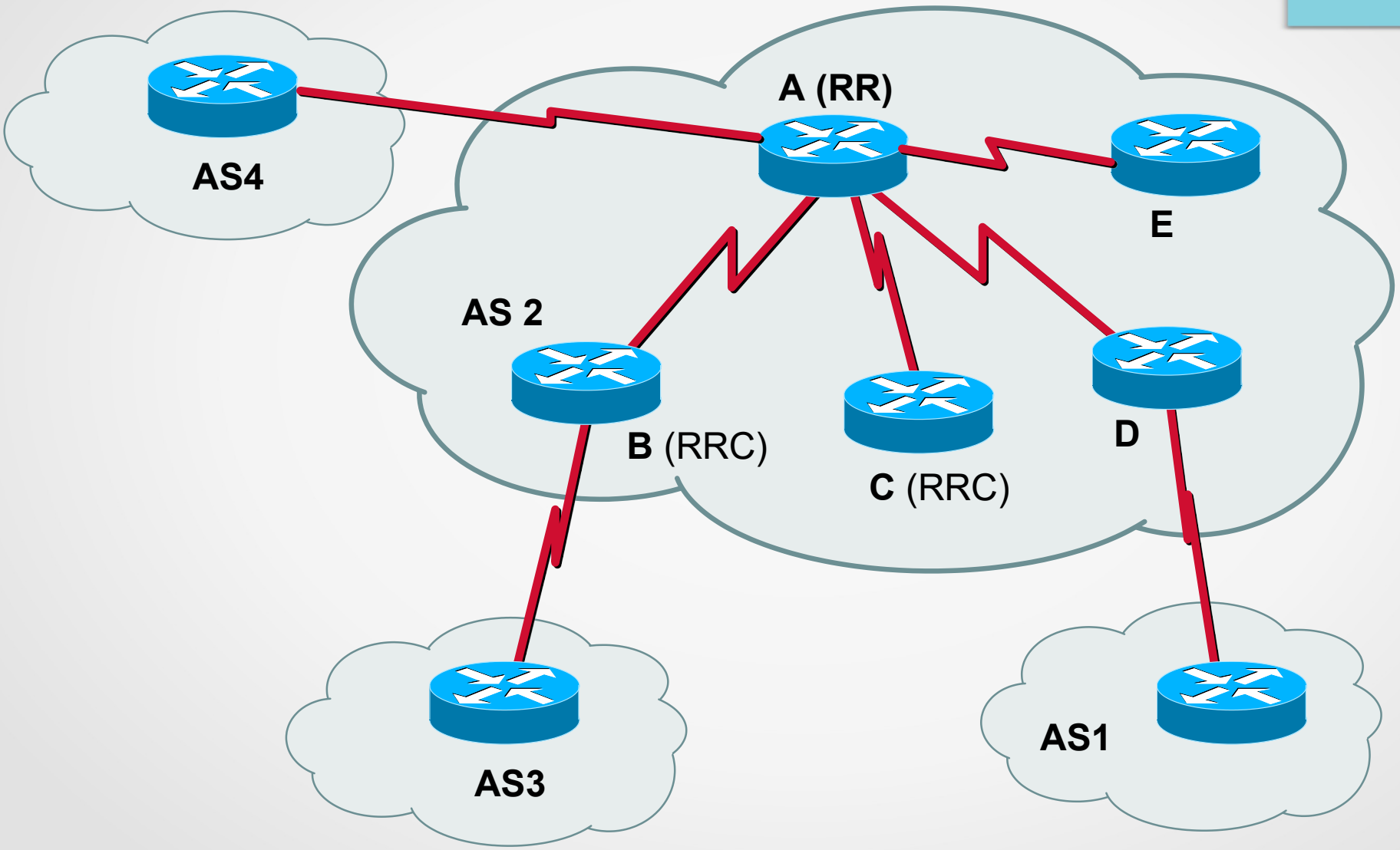
  De ser necesario, tener cuidado para evitar loops

- En caso de múltiples reflectores en un cluster, configurar el mismo CLUSTER_ID

  Esta recomendación está en discusión dependiendo de la topología.

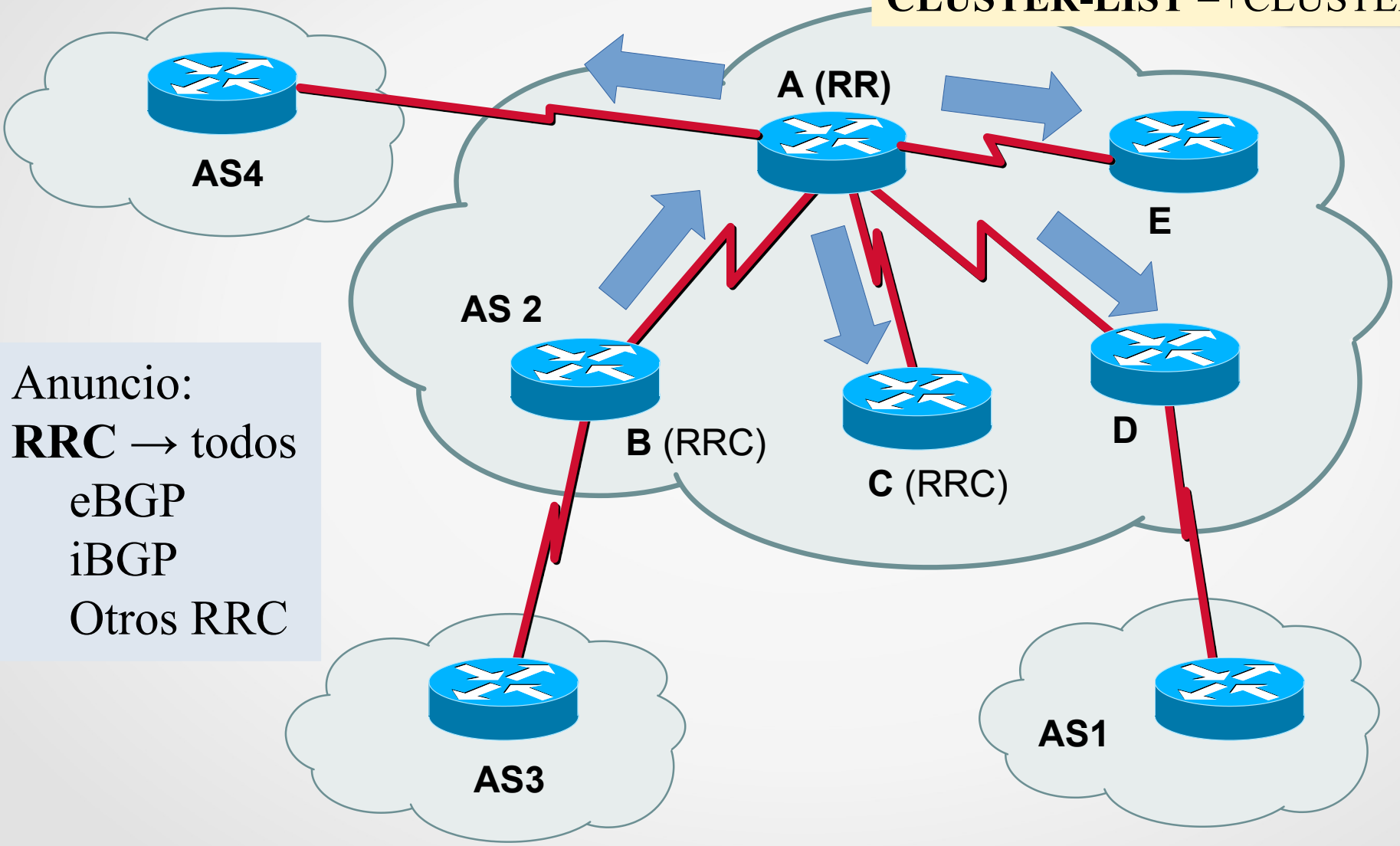
- **Impactos de utilizar RR:**
  - Demoras en propagar la información (los anuncias en vez de ir directos tienen que pasar por la jerarquía de RR)
  - Pérdida de diversidad de caminos (solo veo el mejor camino de los RR).

Route Reflector: Ejemplo



Route Reflector: Ejemplo

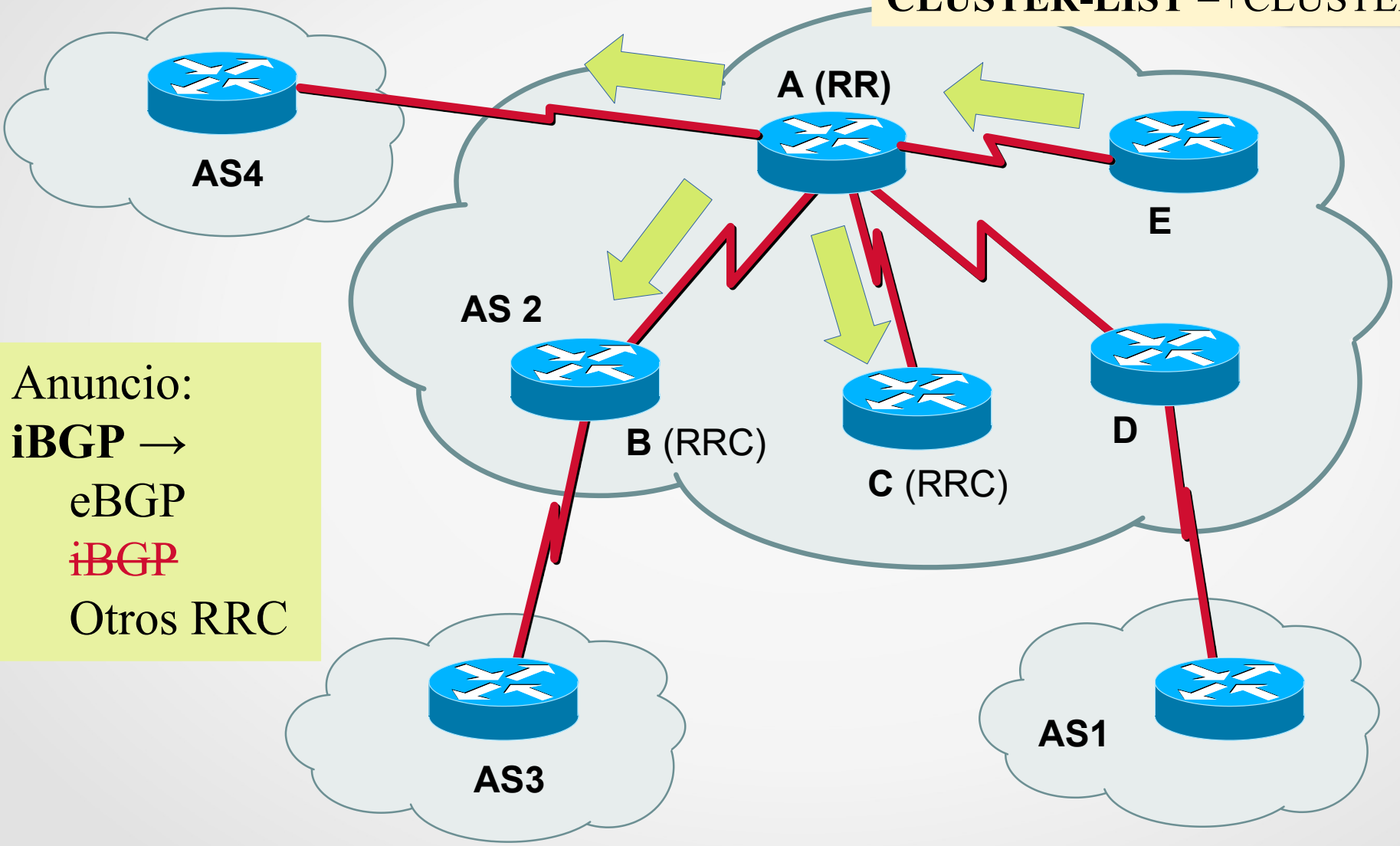
ORIGIN_ID = ROUTER-ID
CLUSTER-LIST =+CLUSTER_ID



Anuncio:
RRC → todos
- eBGP
- iBGP
- Otros RRC

Route Reflector: Ejemplo

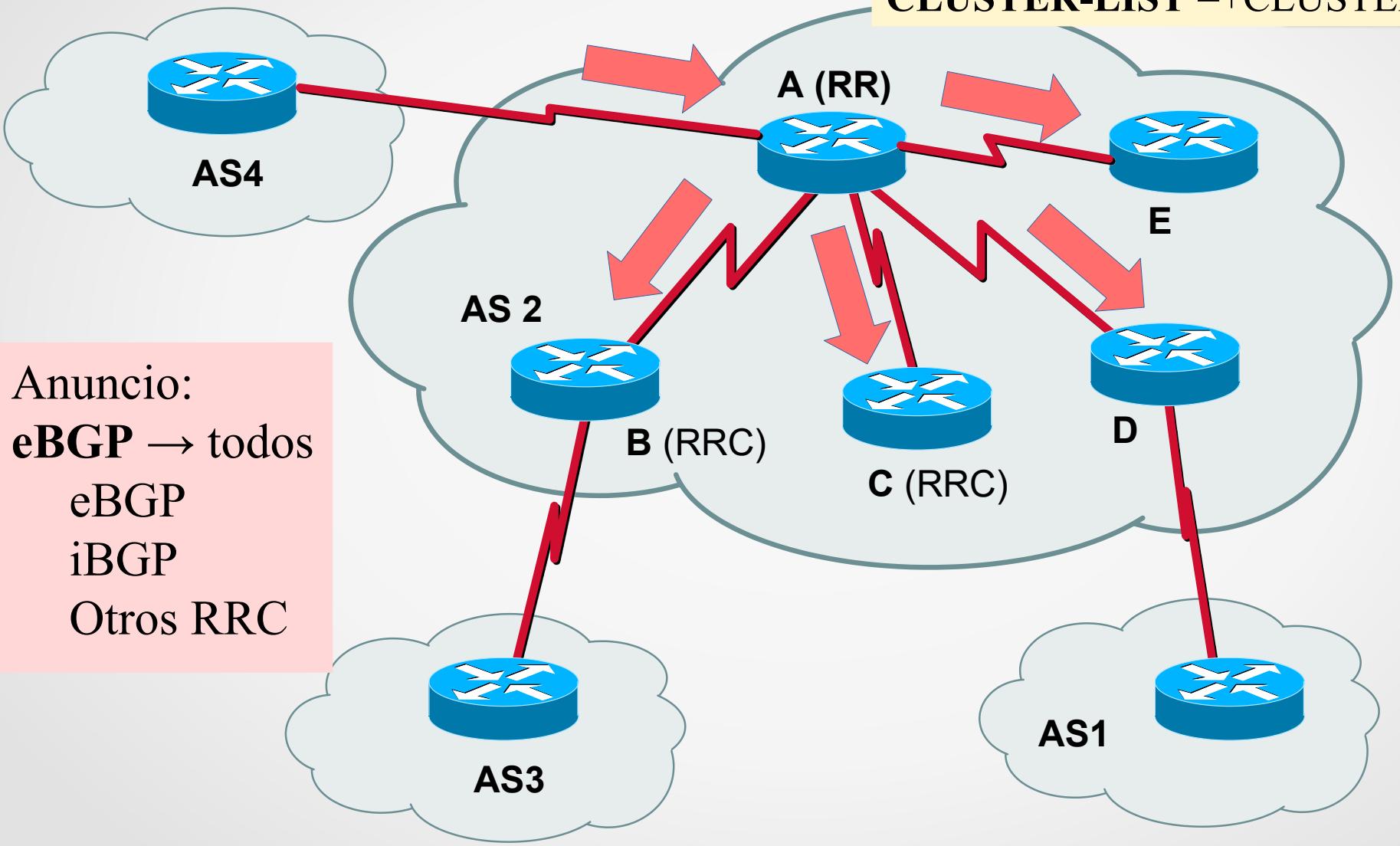
ORIGIN_ID = ROUTER-ID
CLUSTER-LIST =+CLUSTER_ID



Anuncio:
iBGP →
eBGP
~~iBGP~~
Otros RRC

Route Reflector: Ejemplo

ORIGIN_ID = ROUTER-ID
CLUSTER-LIST =+CLUSTER_ID

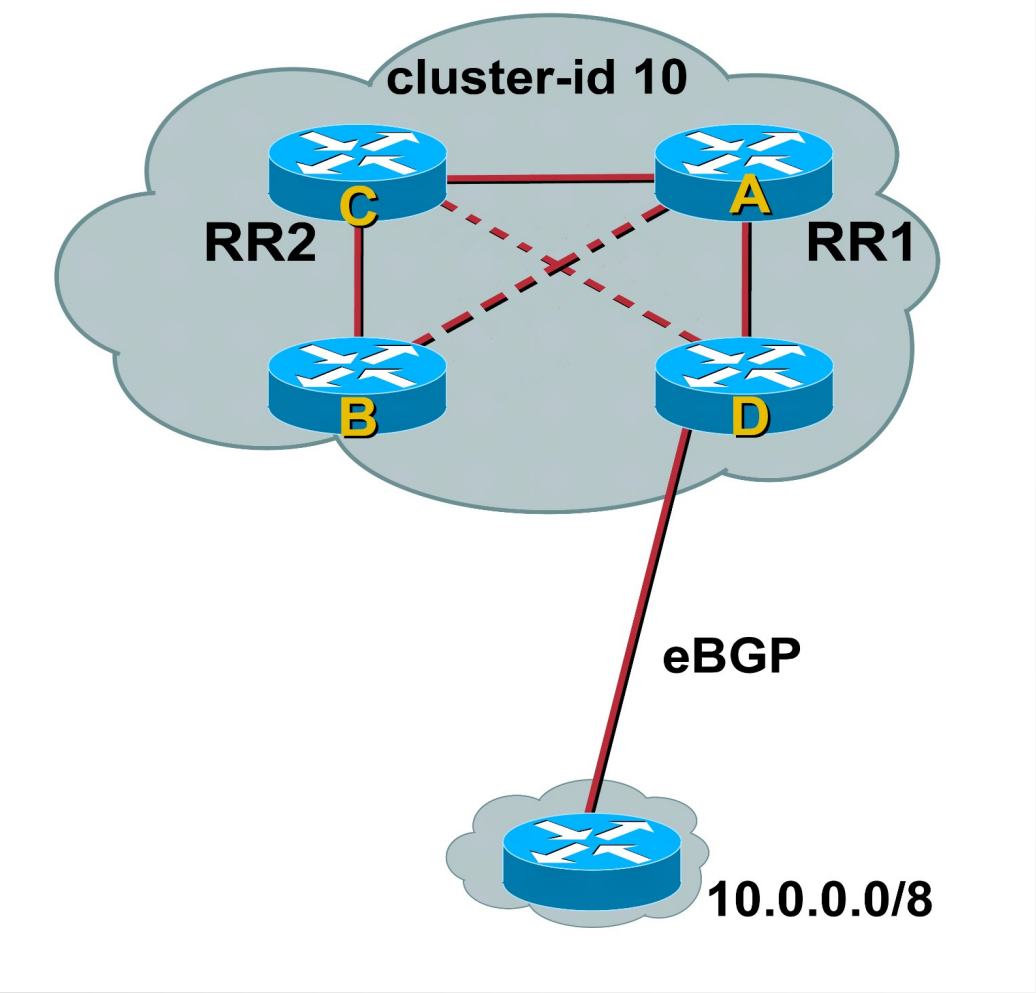


Anuncio:
eBGP → todos
eBGP
iBGP
Otros RRC

(Paréntesis) - Loopback Interface on routing

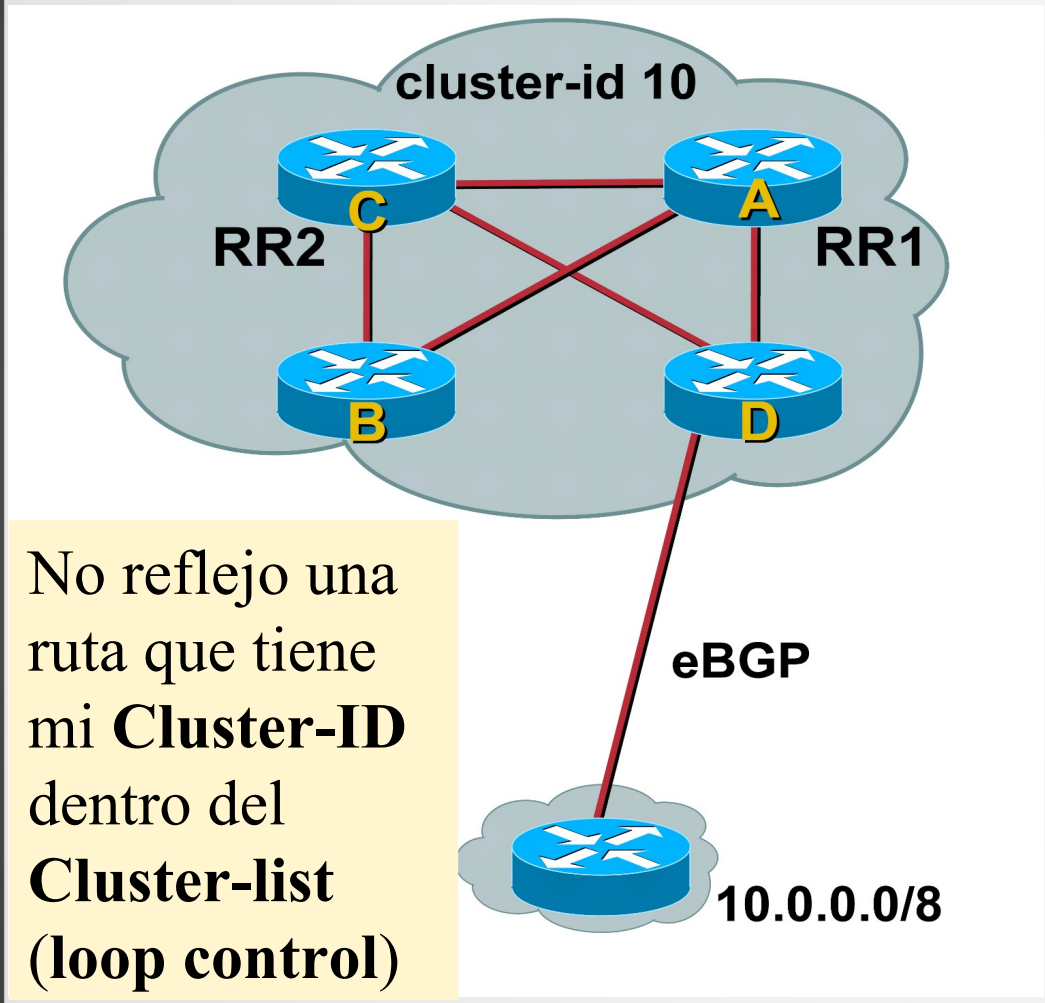
- Una interfaz de **loopback** “pertenece” al router de forma independiente de las interfaces físicas (interfaces lógicas).
- Siempre está “arriba” o disponible, salvo que no lo esté el router.
- En caso de utilizarse para formar adjacencias entre dos routers, **tiene como ventaja de estar disponible mientras al menos una interfaz física** esté disponible y esté participando del IGP.
- Útiles para el acceso a gestión, reportes de alarmas u otros intercambios donde interese dialogar con el router (**independientemente de la interfaz física por la cual ingreso al router**).
- Se suele utilizar como valor de **router-id** en varios protocolos (OSPF ,LDP , BGP)

Route Reflector y **mismo Cluster ID**.



Hay un camino desde B a D, por los links BC, CA, AD.

Route Reflector y **mismo Cluster ID**.



No reflejo una ruta que tiene mi **Cluster-ID** dentro del **Cluster-list** (**loop control**)

1) RR-A y RR-C tienen el **mismo cluster-id**

2) D anuncia a RR-C y RR-A

3) RR-A refleja a B
RR-A refleja a RR-C (IBGP entre RR)

4) RR-C ve que el anuncio contiene **su Cluster-ID** en el **Cluster-list** y descarta para prevenir loops.

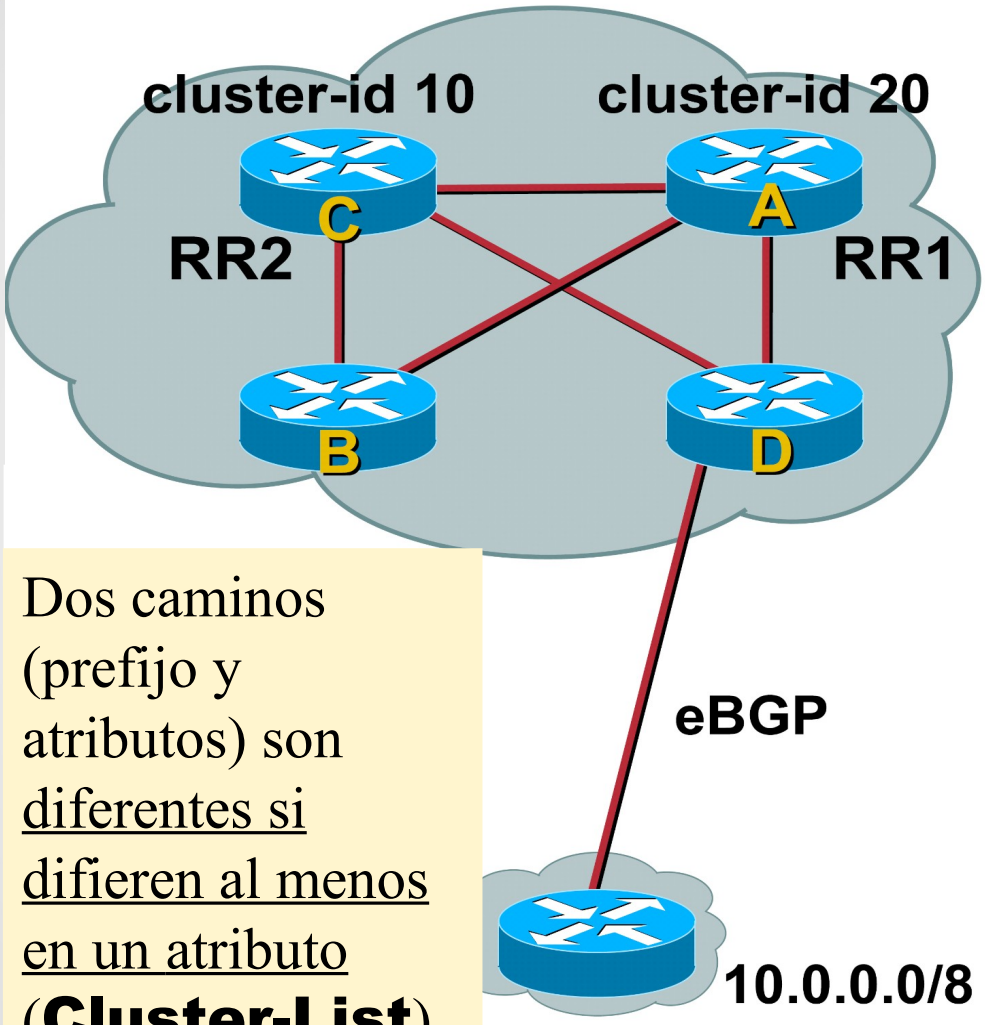
5) RR-C solo aprende un camino para llegar a lo anunciado por D. **¿Si cae el link C-D?**

6) B solo aprende como alcanzar lo anunciado por D solo por RR-A

Problema: Una caída del link entre RR-A y B hace que B no disponga de como llegar a 10.0.0.0/8 por más que haya un camino viable

Solución: BGP desde las Loopbacks y no desde la WAN

Route Reflector y **diferente Cluster ID.**



Dos caminos (prefijo y atributos) son diferentes si difieren al menos en un atributo (**Cluster-List**)

1) RR-A y RR-C tienen **diferente cluster-id**

2) D anuncia a RR-C y RR-A

3) RR-A refleja a B
RR-A refleja a RR-C

4) RR-C ve que el anuncio contiene **cluster-id 20** (!= cluster-id 10) en el **Cluster-list** y lo refleja a B y D.

5) B aprende que llega a 10.0.0.0/8 por RR-C y RR-A.

Observar: B posee dos caminos para llegar al mismo prefijo, por mas que use uno. Estos dos anuncios están en la RIB.

Dos caminos en la RIB: más memoria pero más diversidad (mejor camino de RRC y el de RRA)

Route Reflector: Inconvenientes

- **Diversidad de caminos:** En un fully-mesh un router decide el mejor camino relativos a el, sobre todo si decido por métrica IGP. Como RRC aprendo el mejor camino del RR.
- **Tiempos de Convergencia:** Los anuncios deben propagarse, en el caso de un withdrawn de un prefijo lo conozco directamente en fully-mesh, pero en jerarquía de RR, debe procesarse primero por los RR.
- **Cambio Cultural en la Operativa:** dada una topología, no es “tan” directo saber quién debe anunciar la ruta, sobre todo en escenario de router que son RRC y además tienen sesiones iBGP.

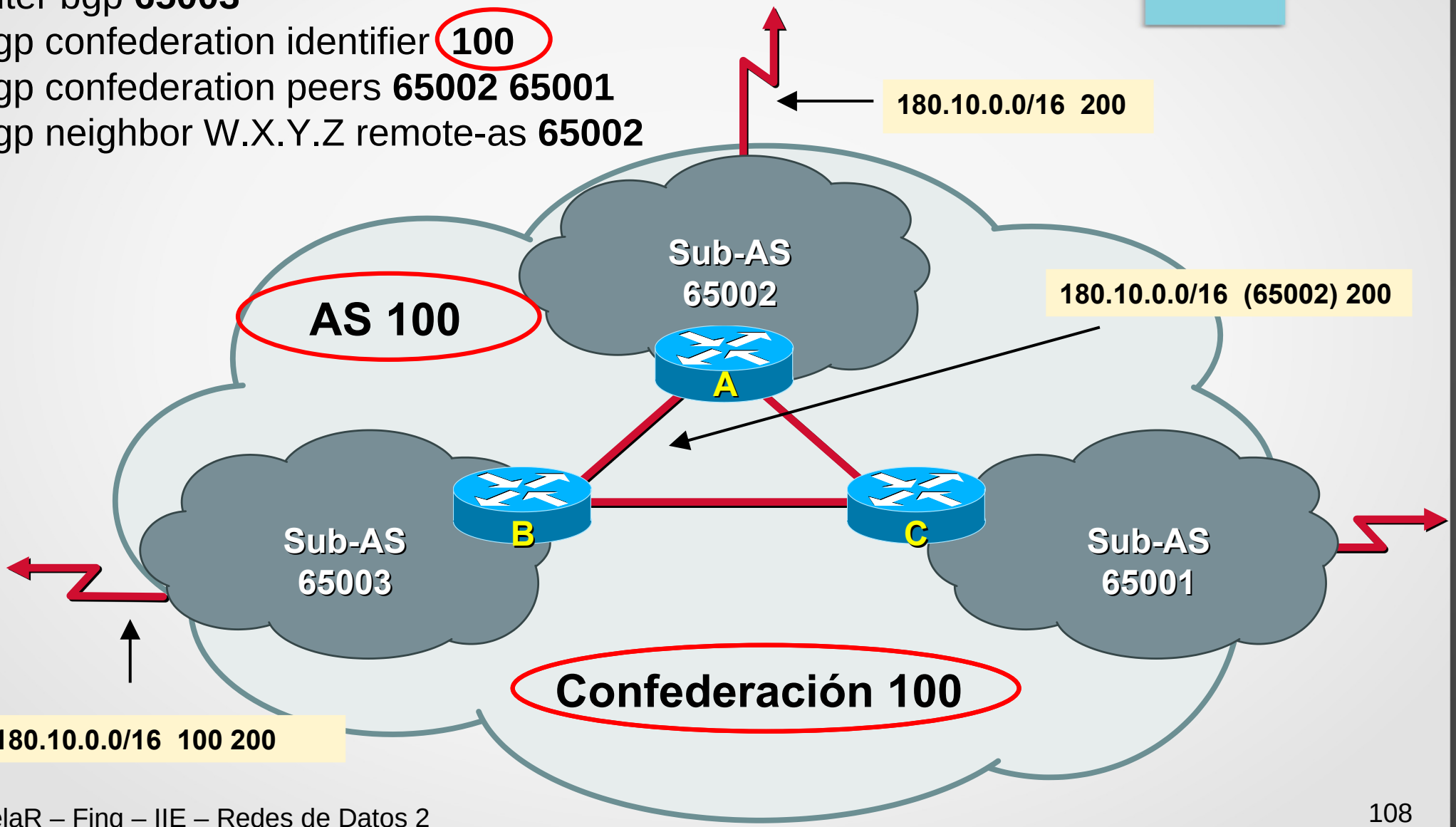
Confederaciones (1)

- Colección de Sistemas Autónomos - **sub-AS**
- Desde el mundo exterior se ve como un único AS
- Se usan los números de AS reservados para los sub-AS internos (AS privados : 64512 - 65535)
- Cada **sub-AS** en arquitectura fully-meshed
- **EBGP entre los sub-AS**
  - **Manteniendo algunas propiedades de iBGP:**

    MED, local-pref y next-hop
  - Surge un nuevo atributo **AS_CONFED_SEQUENCE** para detección de loop

Confederaciones (2)

router bgp **65003**
  bgp confederation identifier **100**
  bgp confederation peers **65002 65001**
  bgp neighbor W.X.Y.Z remote-as **65002**



Confederaciones: Beneficios

- Soluciona y escala el problema de IBGP full-mesh
- Puede ser usado conjuntamente con Route Reflectors
- Admite la aplicación de políticas para enrutar tráfico entre los distintos sub-ASs
- Permite **configurar diferentes IGP** por sub-As
- Dentro de un sub-AS, sigo teniendo que aplicar full-mesh iBGP o RR para obtener topologías que pueda deshabilitar la sincronización
- El **mayor inconveniente** es que no hay transición directa de IBGP (RR) a Confederaciones.
- **Observación:** Es posible utilizar AS privados de forma interna con eBGP sin confederaciones, pero se pierde el MED, local preference y el next-hop. Aunque para MED y local preference es posible definir un par de comunidades equivalentes.

  Se debe remover los AS Privados en el AS-PATH en eBGP a otros AS públicos