

BGP

Protocol & Configuration



Scalable Infrastructure
Workshop
AfNOG 2011

Border Gateway Protocol (BGP4)

- ❑ Case Study 1, Exercise 1: Single upstream
- ❑ Part 6: BGP Protocol Basics
- ❑ Part 7: BGP Protocol - more detail
- ❑ Case Study 2, Exercise 2: Local peer
- ❑ Part 8: Routing Policy and Filtering
- ❑ Exercise 3: Filtering on AS-path
- ❑ Exercise 4: Filtering on prefix-list
- ❑ Part 9: More detail than you want
- ❑ Exercise 5: Interior BGP
- ❑ Part 10: BGP and Network Design

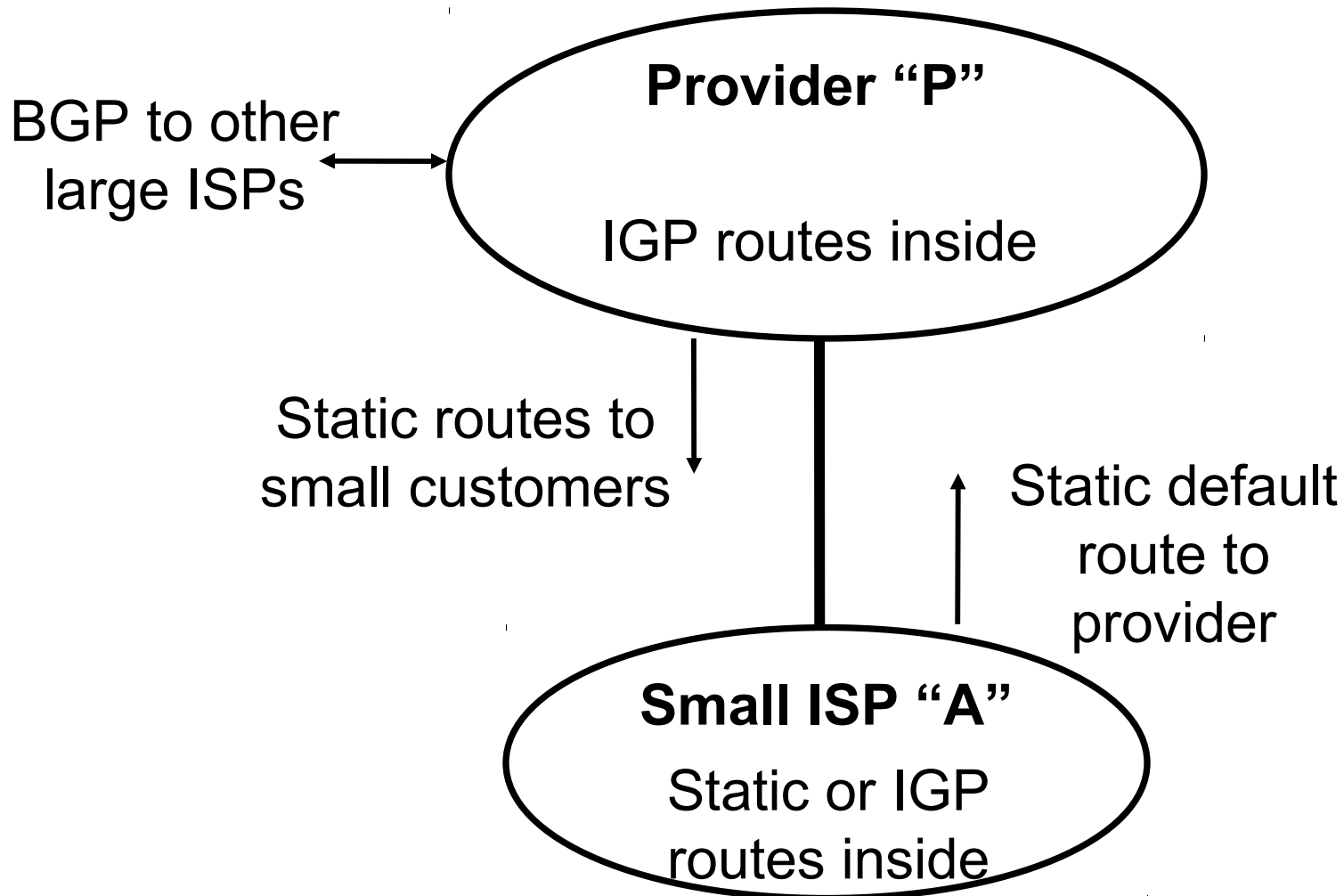
BGP Case Study 1 and Exercise 1

Small ISP with one upstream
provider

Case Study 1: Small ISP with one upstream provider

- Local network
- May have multiple POPs
- Line to Internet
 - International line providing transit connectivity
 - Very, very expensive

Case Study 1: Small ISP with one upstream provider



Case Study 1: Routing Protocols

- ❑ Static routes or IGP inside small ISP "A"
- ❑ Static default route from small ISP "A" to upstream provider "P"
- ❑ IGP inside upstream provider "P"
- ❑ The two IGPs do not know about each other
- ❑ BGP between upstream provider "P" and outside world

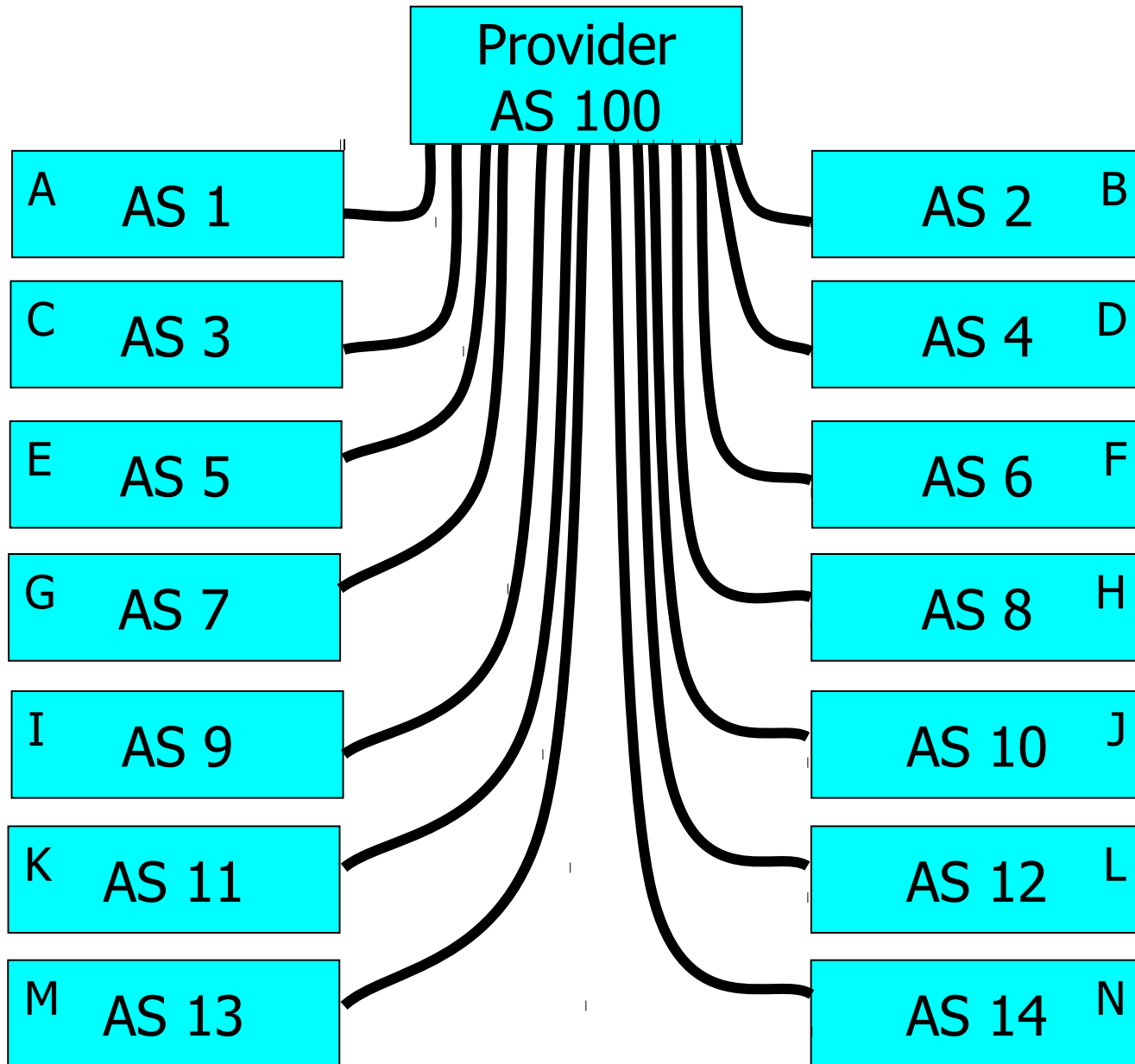
Case Study 1: BGP is not needed

- No need for BGP between small ISP "A" and upstream provider "P"
- The outside world does not need to care about the link between provider "P" and customer "A"
- Hiding that information from the outside world helps with scaling
- **We will do an exercise using BGP even though it is not needed**

Exercise 1: Upstream provider with small customers

- ❑ This is not a realistic exercise
- ❑ In reality, a single-homed network would not use BGP
- ❑ Exercise 2 will be more realistic, adding a connection between two small ISPs in the same country

Exercise 1: Upstream provider & small customers



Exercise 1:

BGP configuration

- ❑ Refer to “BGP cheat sheet”
- ❑ Connect cable to upstream provider
- ❑ “router bgp” for your AS number
- ❑ BGP “network” statement for your network
- ❑ BGP “neighbor” for upstream provider (IP address 196.200.220.xx, remote AS 100)
 - (Your workshop instructor will provide point to point link addresses)
- ❑ Do the same for IPv6

Exercise 1: Transit through upstream provider

- Instructors configure AS 100 to send you all routes to other classroom ASes, and a default route
 - You can send traffic through AS 100 to more distant destinations
 - In other words, AS 100 provides “transit” service to you

Exercise 1:

What you should see

- You should see routes to all other classroom networks
- Try:
 - “show ip route” to see IPv4 routing table
 - “show ipv6 route” to see IPv6 routing table
 - “show ip bgp” to see IPv4 BGP table
 - “show bgp ipv6” to see IPv6 BGP table
- Look at the “next hop” and “AS path”
- Try some pings and traceroutes.

Exercise 1: Did BGP “network” statement work?

- ❑ BGP “network” statement has no effect unless route exists in IGP (or static route)
- ❑ You might need to add a static route to make it work
 - IPv4: ip route x.x.x.x m.m.m.m Null0 250
 - IPv6: ipv6 route x:x::/60 Null0 250
- ❑ 250 is the administrative distance
 - Smaller is “less important”
 - Default for a static route is 1

BGP Part 6



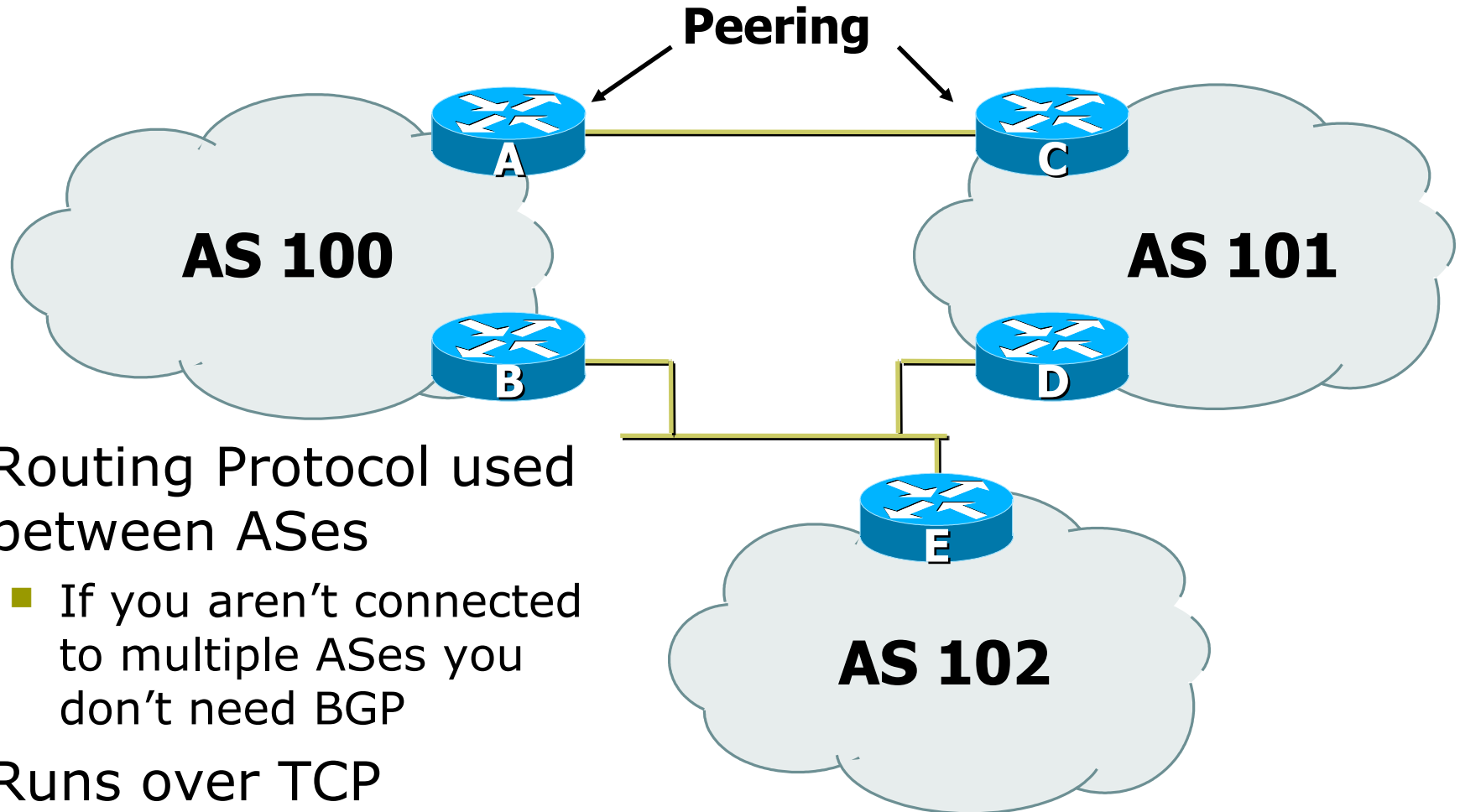
BGP Protocol Basics

Terminology

General Operation

Interior/Exterior BGP

BGP Protocol Basics



- ❑ Routing Protocol used between ASes
 - If you aren't connected to multiple ASes you don't need BGP
- ❑ Runs over TCP

BGP Protocol Basics

- Uses Incremental updates
 - sends one copy of the RIB at the beginning, then sends changes as they happen
- Path Vector protocol
 - keeps track of the AS path of routing information
- Many options for policy enforcement

Terminology

- Neighbour
 - Configured BGP peer
- NLRI/Prefix
 - NLRI – network layer reachability information
 - Reachability information for an IP address & mask
- Router-ID
 - 32 bit integer to uniquely identify router
 - Comes from Loopback or Highest IP address configured on the router
- Route/Path
 - NLRI advertised by a neighbour

Terminology

- ❑ Transit – carrying network traffic across a network, usually for a fee
- ❑ Peering – exchanging routing information and traffic
 - your customers and your peers' customers network information only.
 - not your peers' peers; not your peers' providers.
- ❑ Peering also has another meaning:
 - BGP neighbour, whether or not transit is provided
- ❑ Default – where to send traffic when there is no explicit route in the routing table

BGP Basics ...

- ❑ Each AS originates a set of NLRI (routing announcements)
- ❑ NLRI is exchanged between BGP peers
- ❑ Can have multiple paths for a given prefix
- ❑ BGP picks the best path and installs in the IP forwarding table
- ❑ Policies applied (through attributes) influences BGP path selection

Interior BGP vs. Exterior BGP

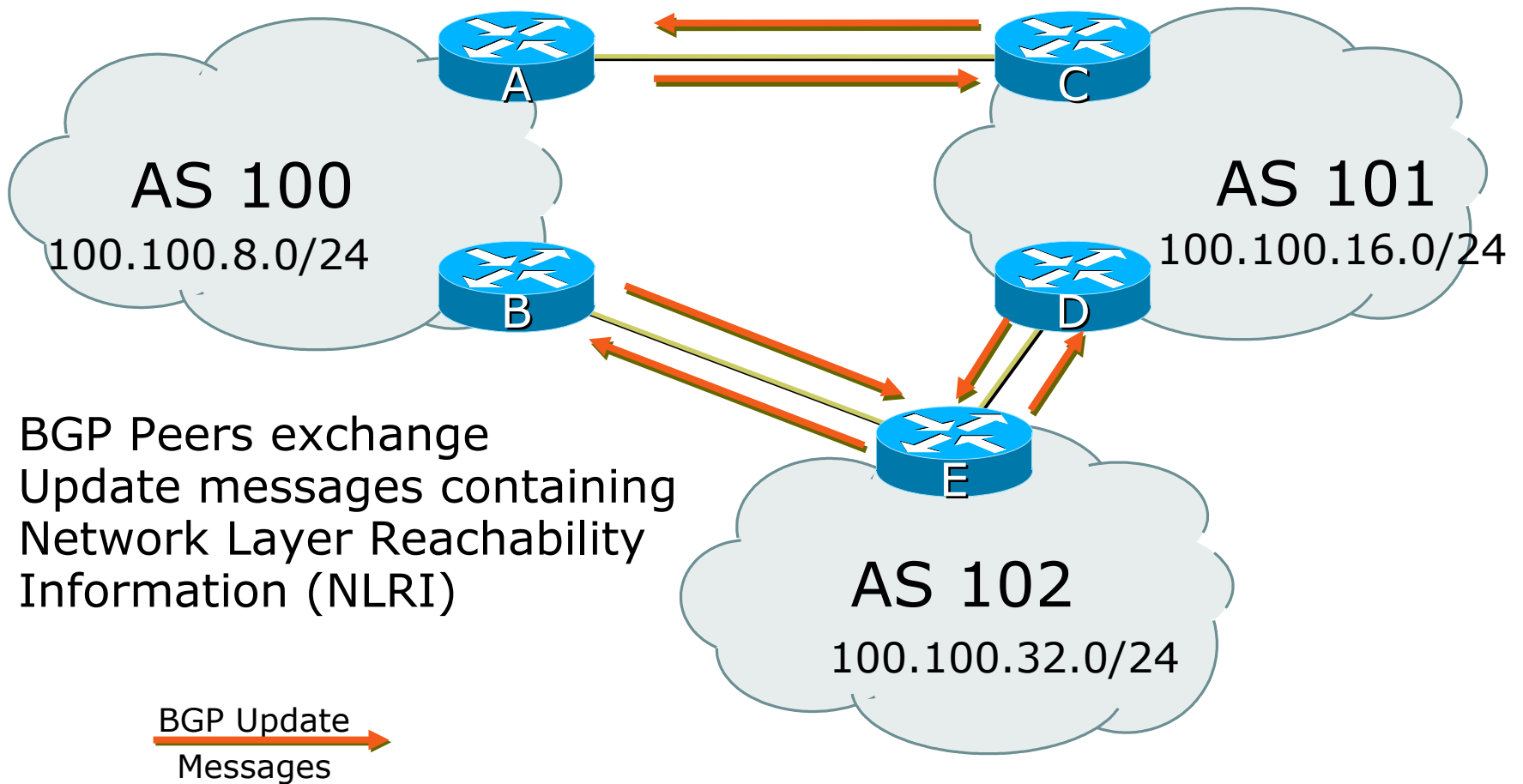
□ Interior BGP (iBGP)

- Between routers in the same AS
- Often between routers that are far apart
- Should be a full mesh: every iBGP router talks to all other iBGP routers in the same AS

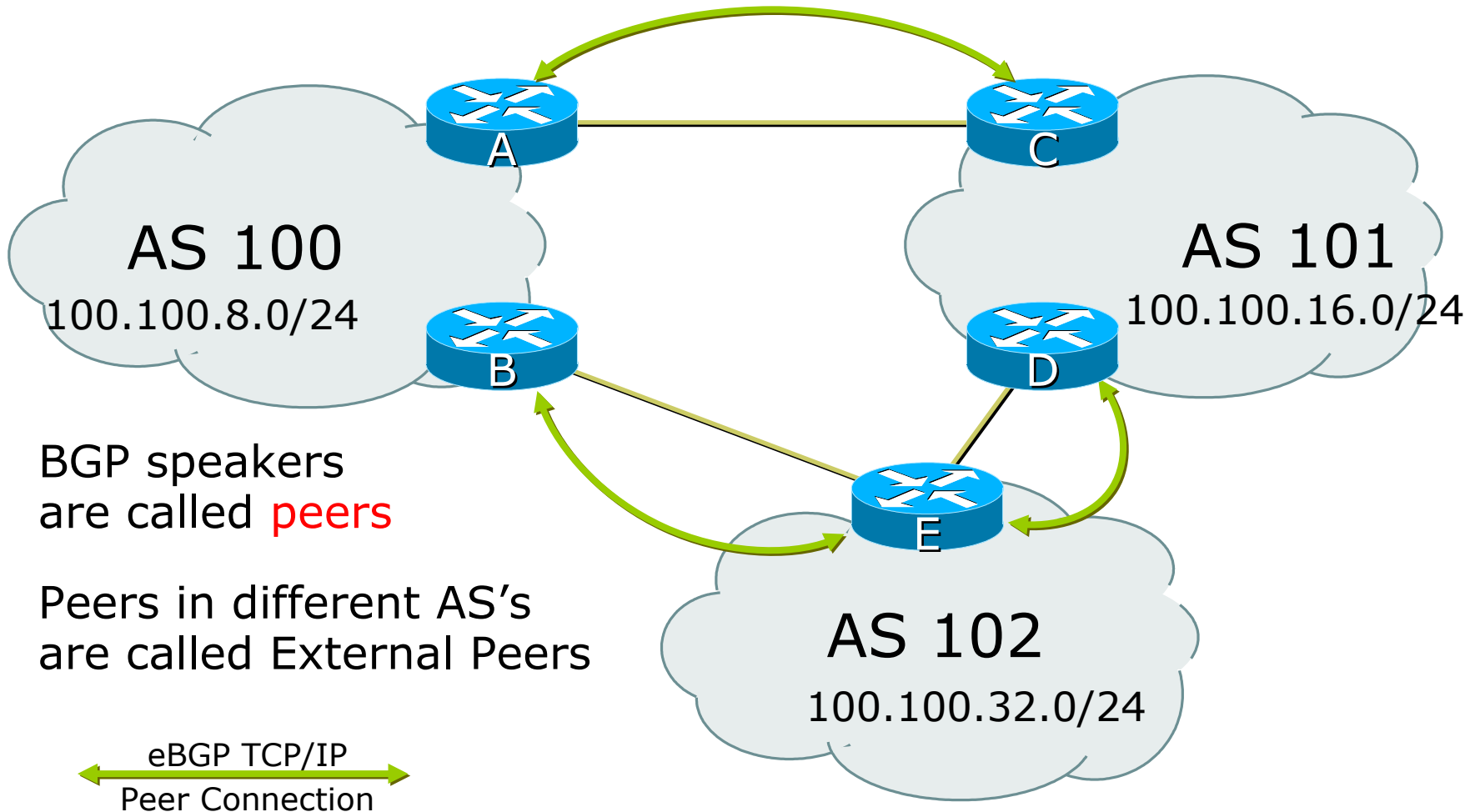
□ Exterior BGP (eBGP)

- Between routers in different ASes
- Almost always between directly-connected routers (ethernet, serial line, etc.)

BGP Peers

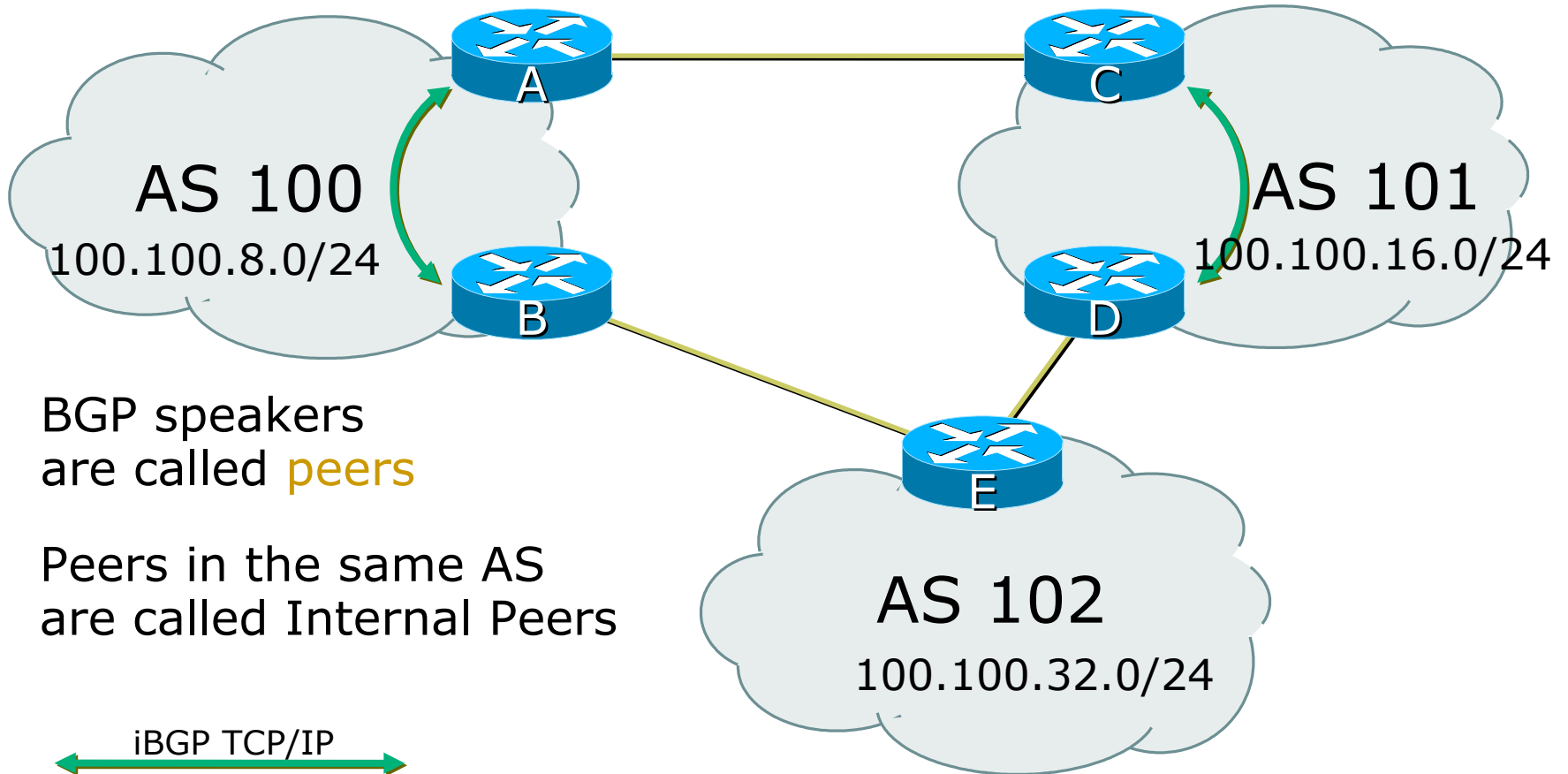


BGP Peers – External (eBGP)



Note: eBGP Peers normally should be directly connected.

BGP Peers – Internal (iBGP)



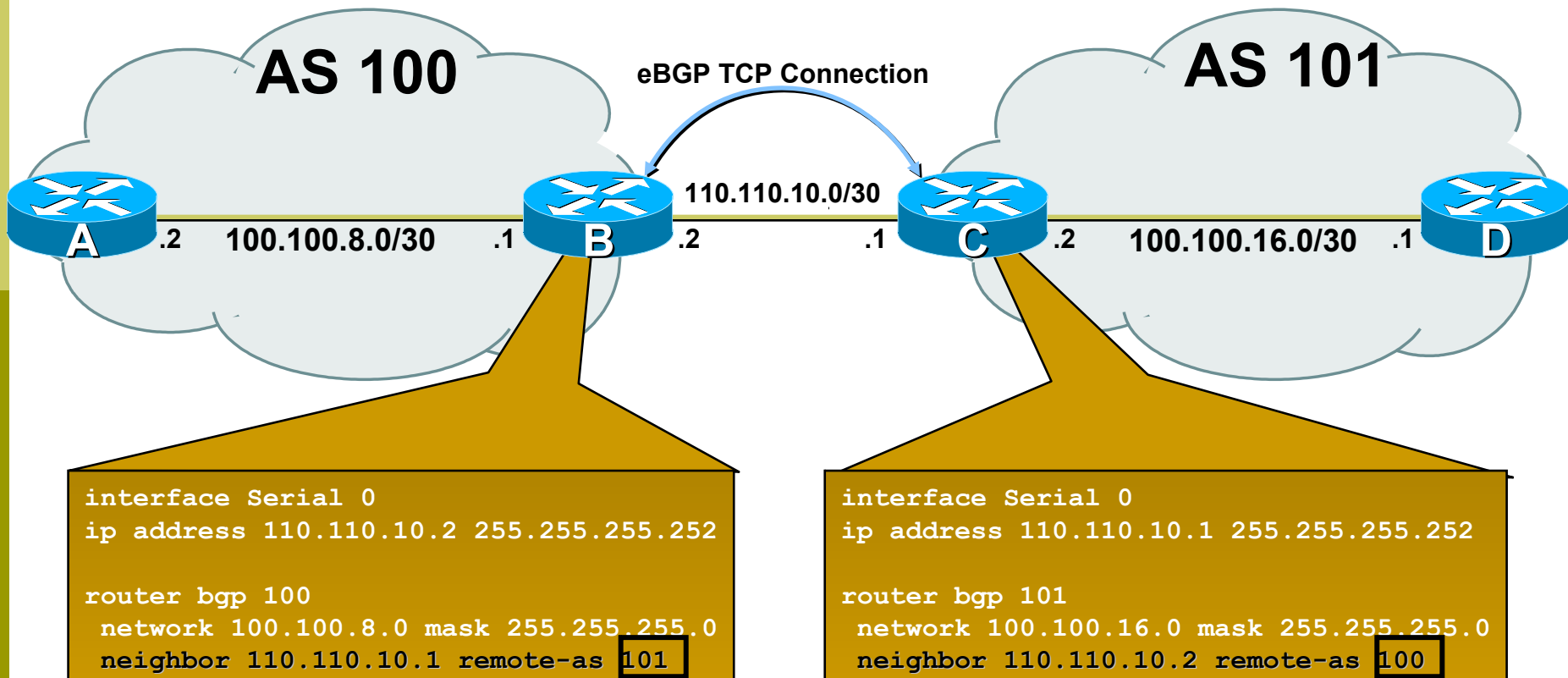
BGP speakers are called **peers**

Peers in the same AS are called Internal Peers

Note: iBGP Peers don't have to be directly connected.

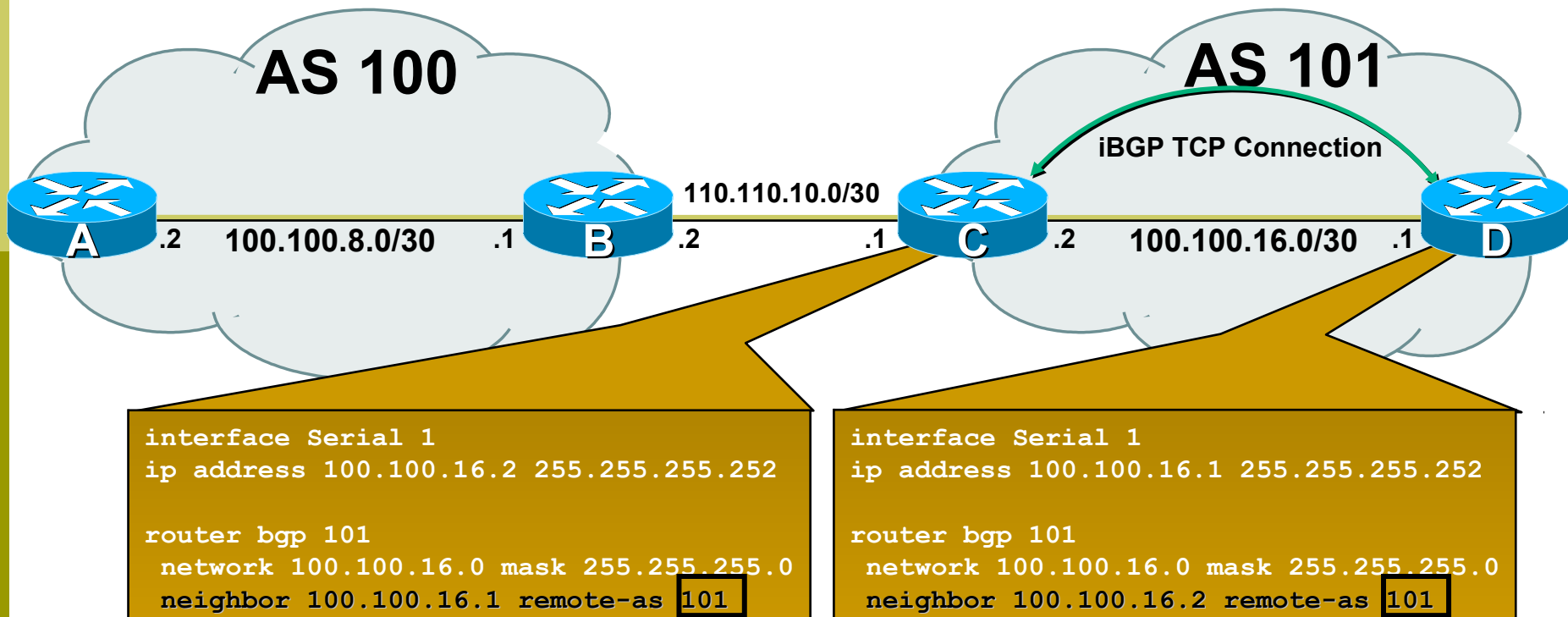
Configuring eBGP peers

- ❑ BGP peering sessions are established using the BGP "neighbor" command
 - eBGP is configured when AS numbers are different



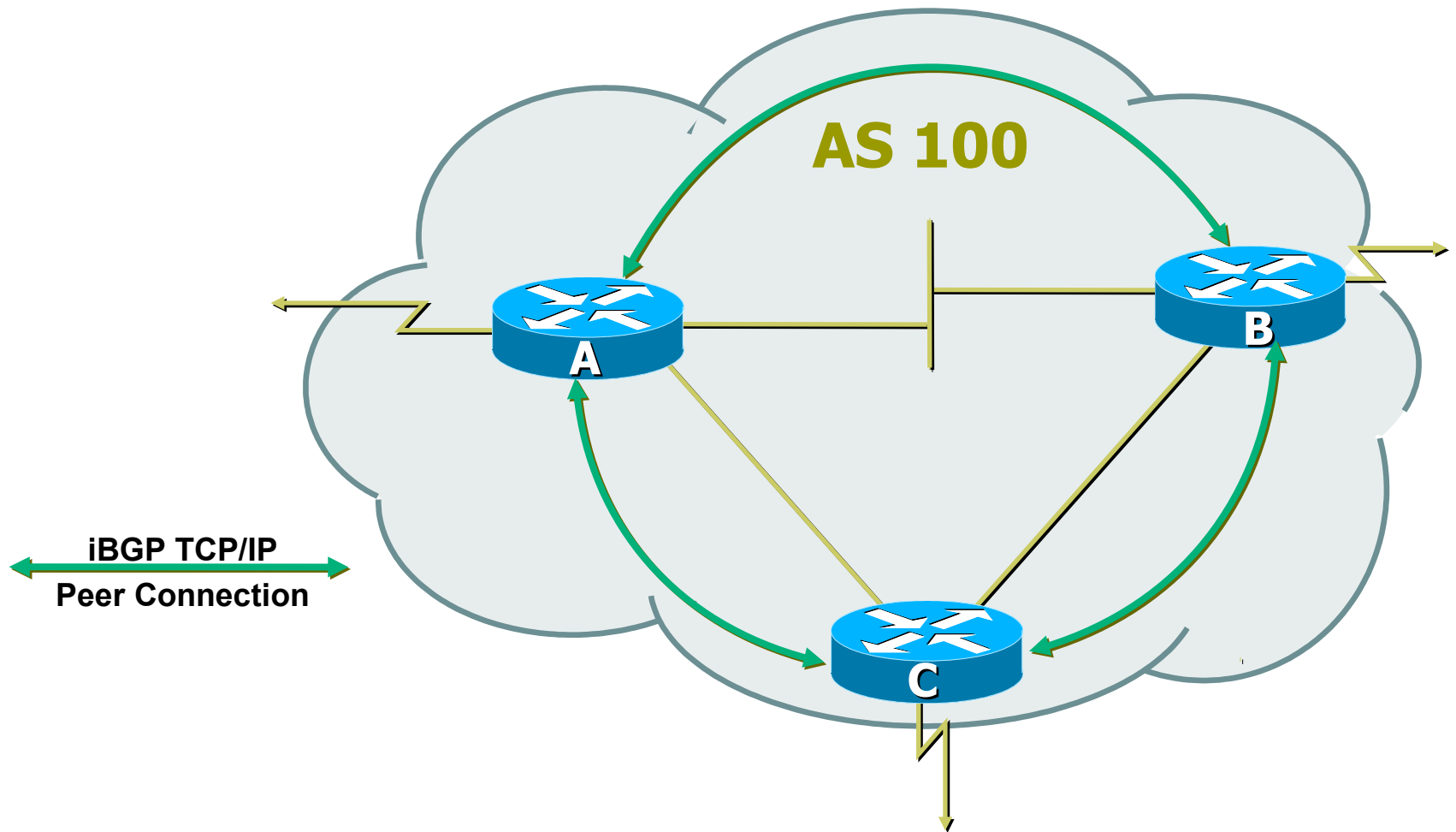
Configuring iBGP peers

- BGP peering sessions are established using the BGP “neighbor” command
 - iBGP is configured when AS numbers are the same



Configuring iBGP peers: Full mesh

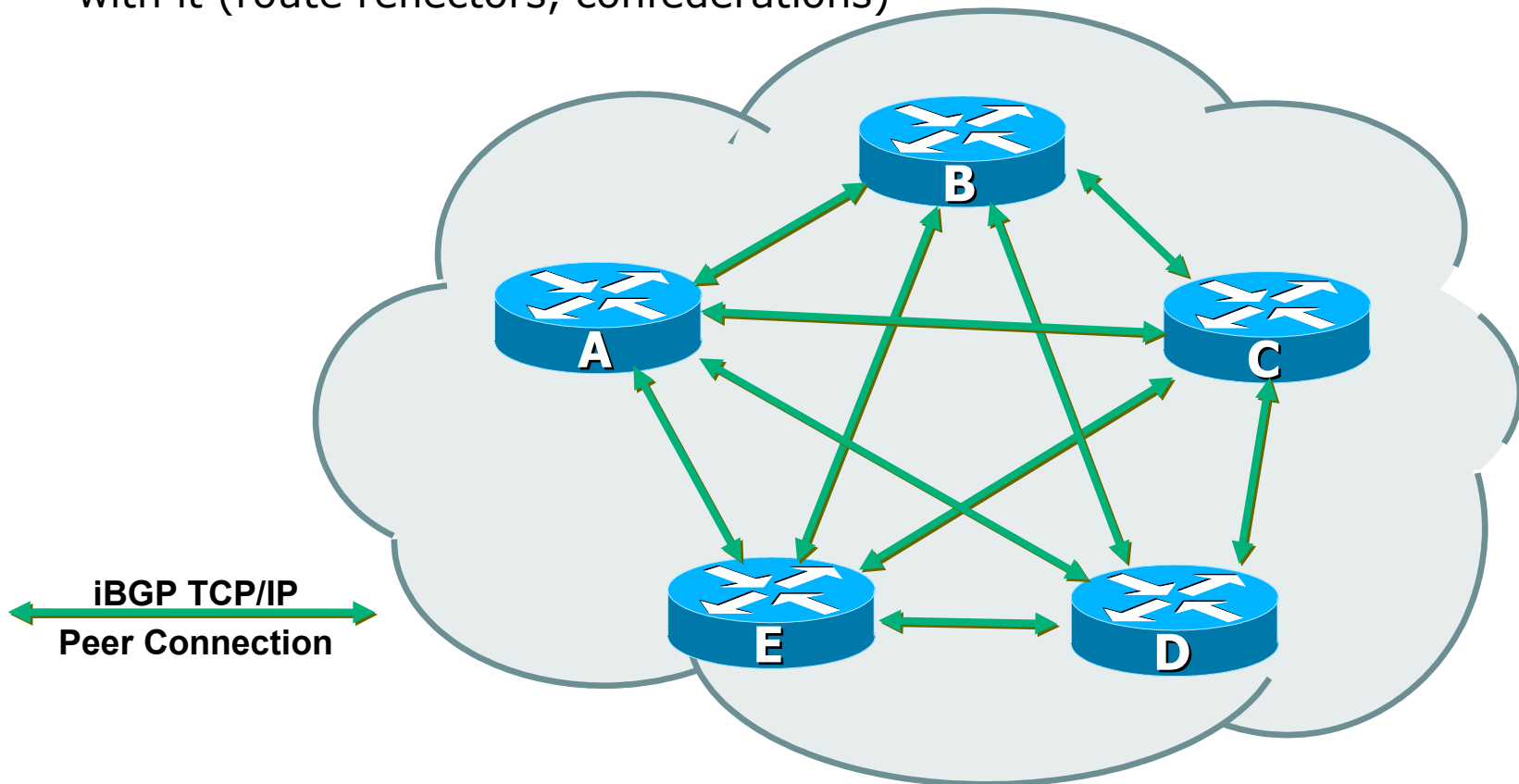
- Each iBGP speaker must peer with every other iBGP speaker in the AS



Configuring iBGP peers:

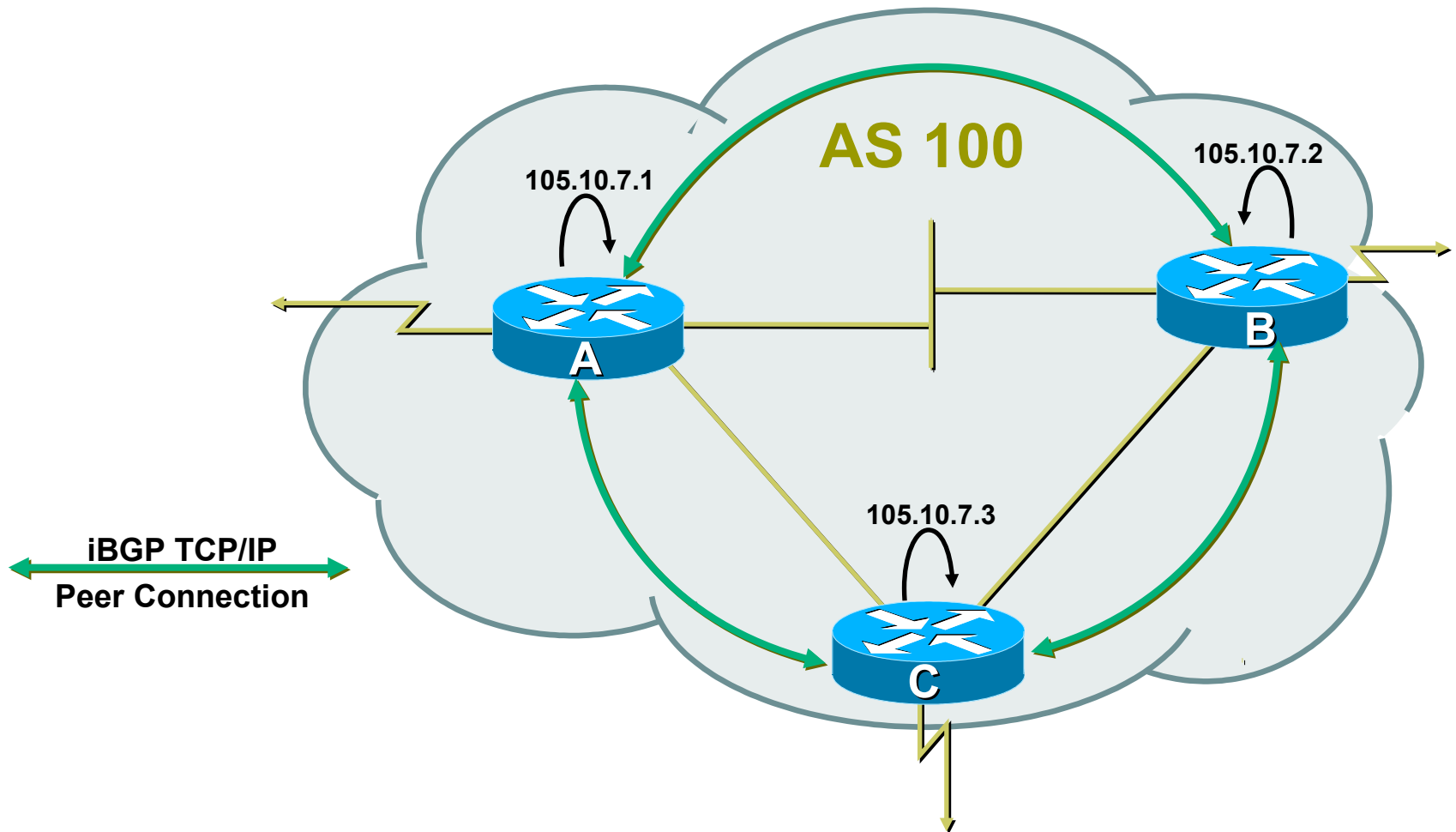
Full mesh

- ❑ Each iBGP speaker must peer with every other iBGP speaker in the AS
- ❑ This can be a pain if there are many routers, but there are ways to deal with it (route reflectors, confederations)

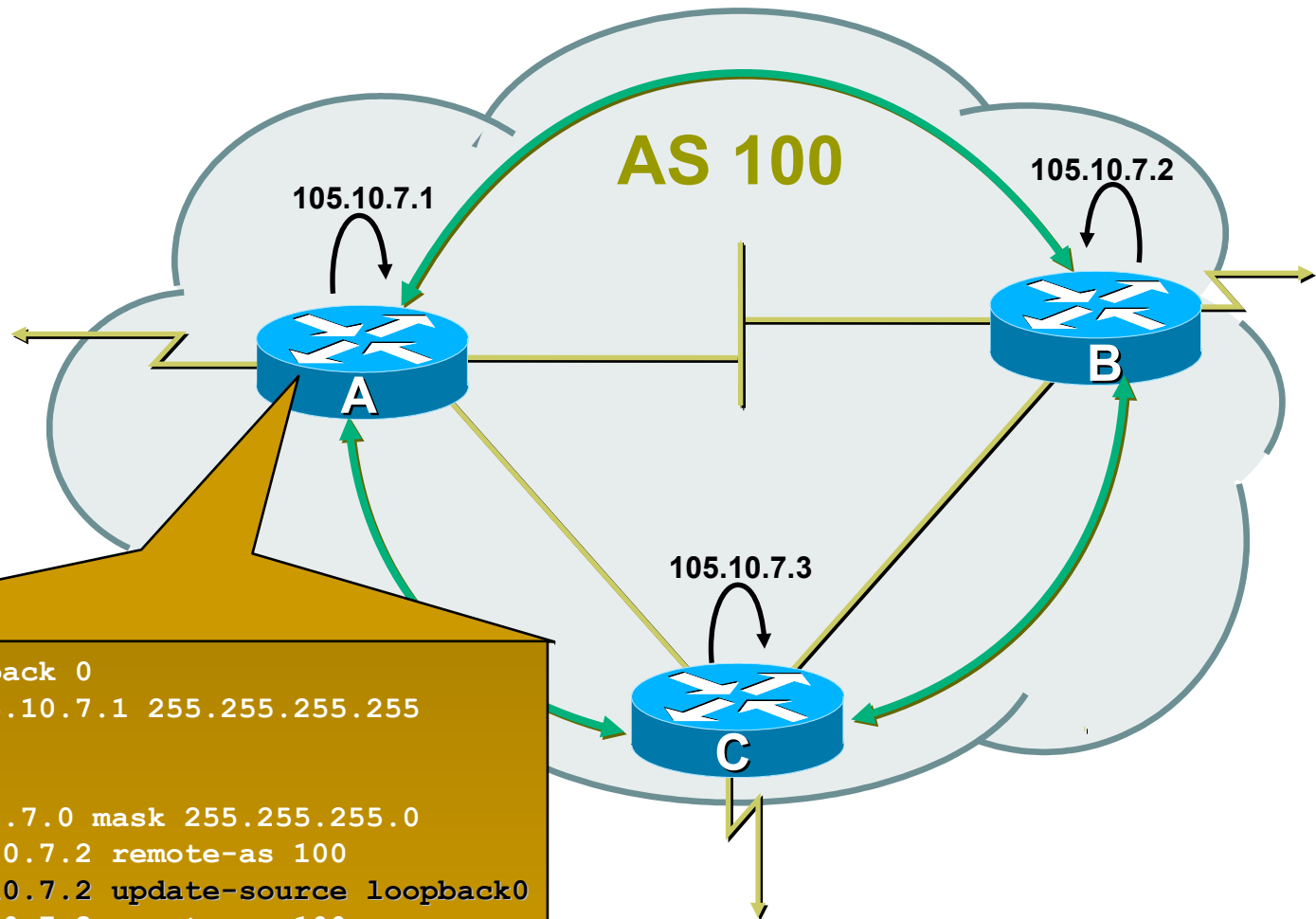


Configuring iBGP peers: Loopback interface

- Loopback interfaces are normally used as the iBGP peer connection end-points



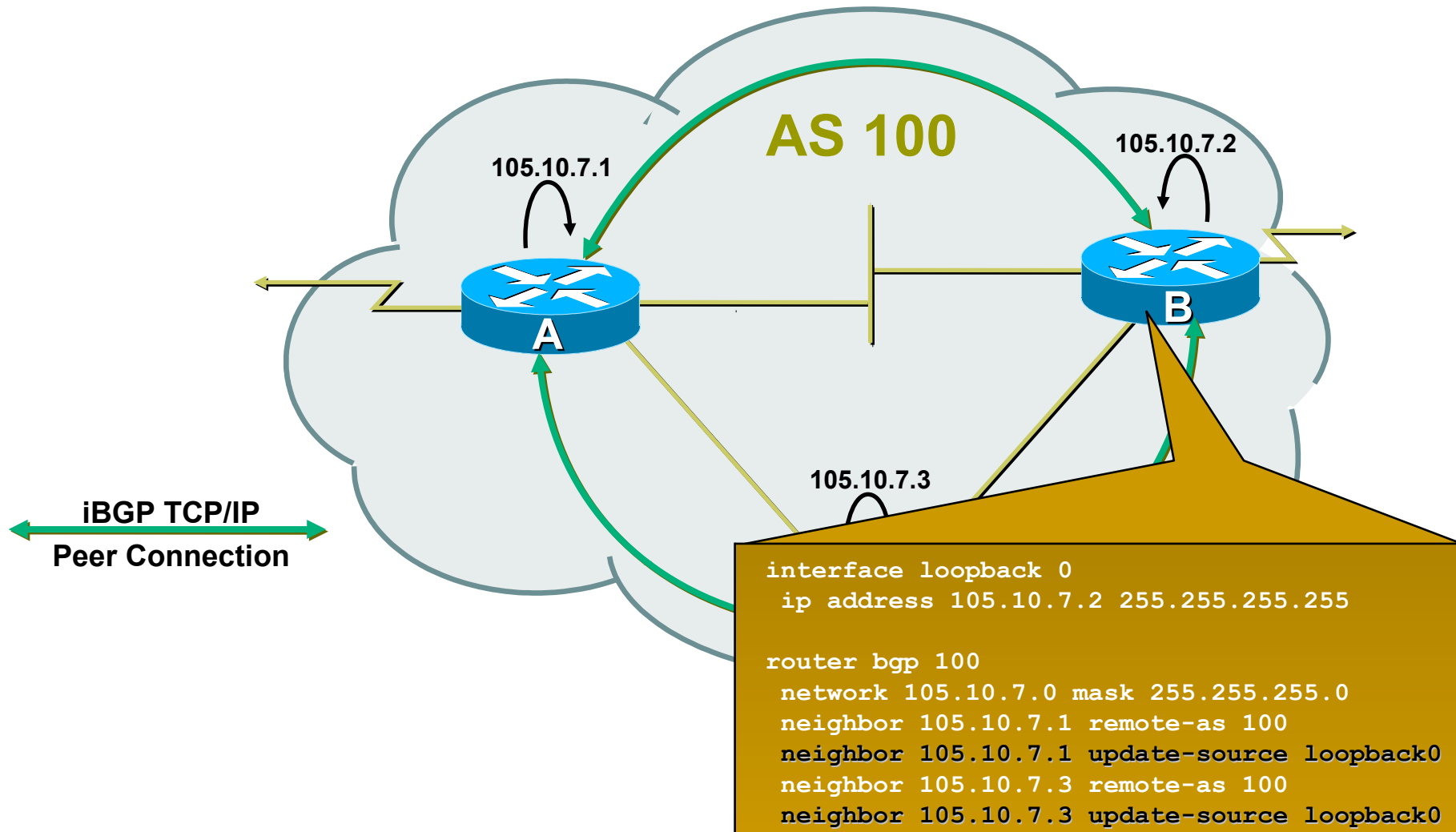
Configuring iBGP peers



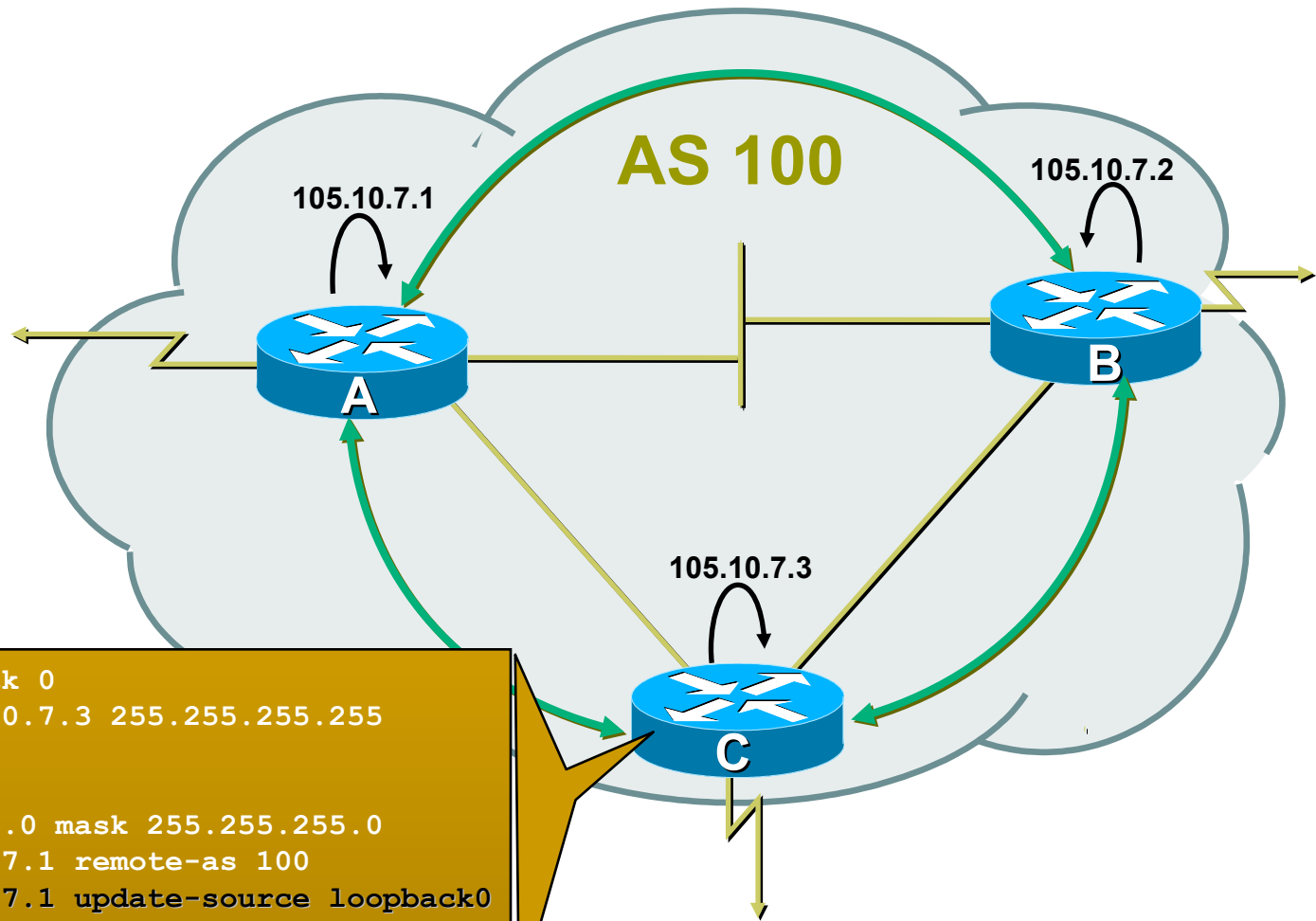
```
interface loopback 0
 ip address 105.10.7.1 255.255.255.255

router bgp 100
 network 105.10.7.0 mask 255.255.255.0
 neighbor 105.10.7.2 remote-as 100
 neighbor 105.10.7.2 update-source loopback0
 neighbor 105.10.7.3 remote-as 100
 neighbor 105.10.7.3 update-source loopback0
```

Configuring iBGP peers



Configuring iBGP peers



```
interface loopback 0
 ip address 105.10.7.3 255.255.255.255

router bgp 100
 network 105.10.7.0 mask 255.255.255.0
 neighbor 105.10.7.1 remote-as 100
 neighbor 105.10.7.1 update-source loopback0
 neighbor 105.10.7.2 remote-as 100
 neighbor 105.10.7.2 update-source loopback0
```

BGP Part 7



BGP Protocol – A little more
detail

BGP Updates — NLRI

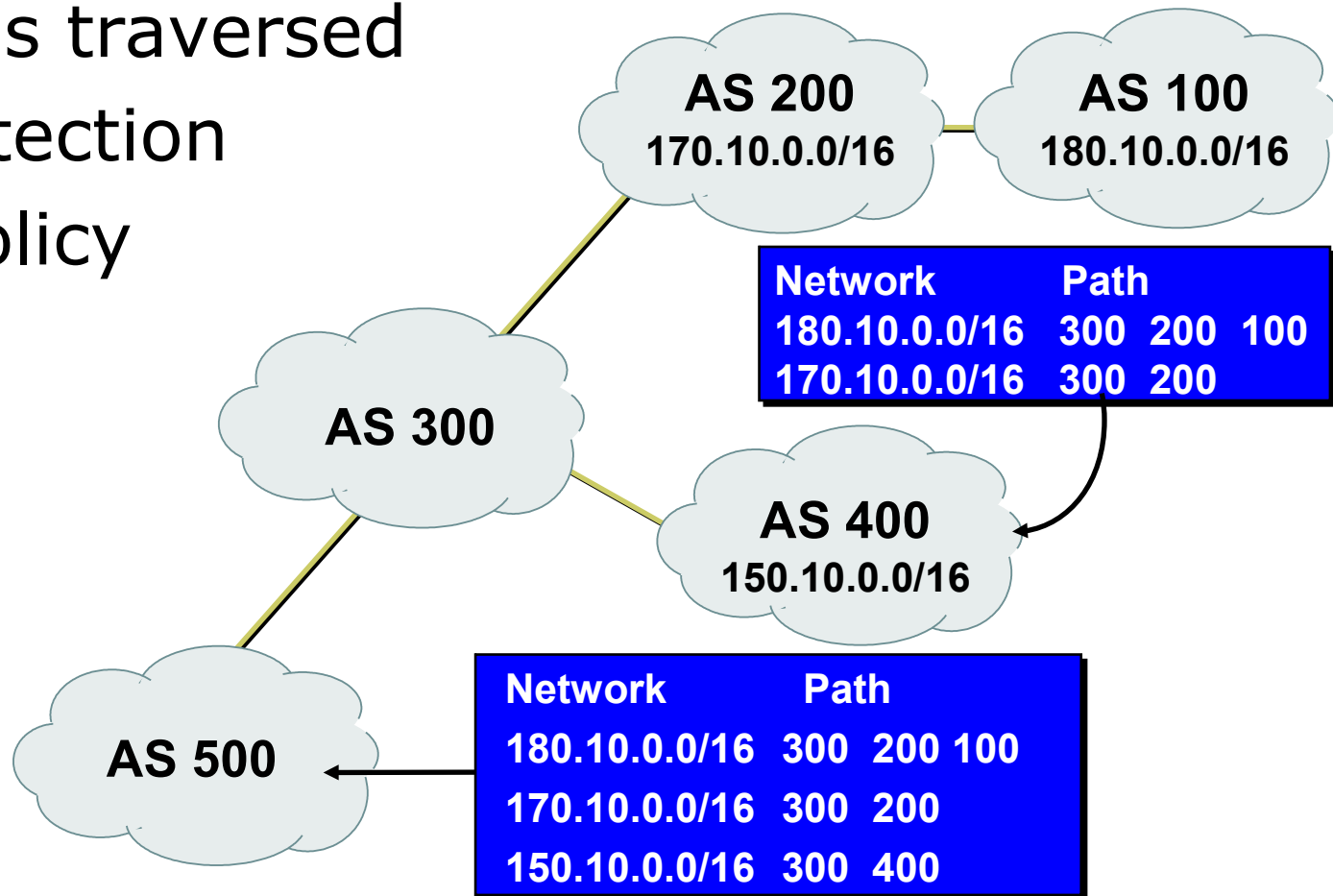
- Network Layer Reachability Information
- Used to advertise feasible routes
- Composed of:
 - Network Prefix
 - Mask Length
 - Attributes of the path between you and the destination

BGP Updates — Attributes

- Used to convey information associated with NLRI
 - AS path
 - Next hop
 - Local preference
 - Multi-Exit Discriminator (MED)
 - Community
 - Origin
 - Aggregator

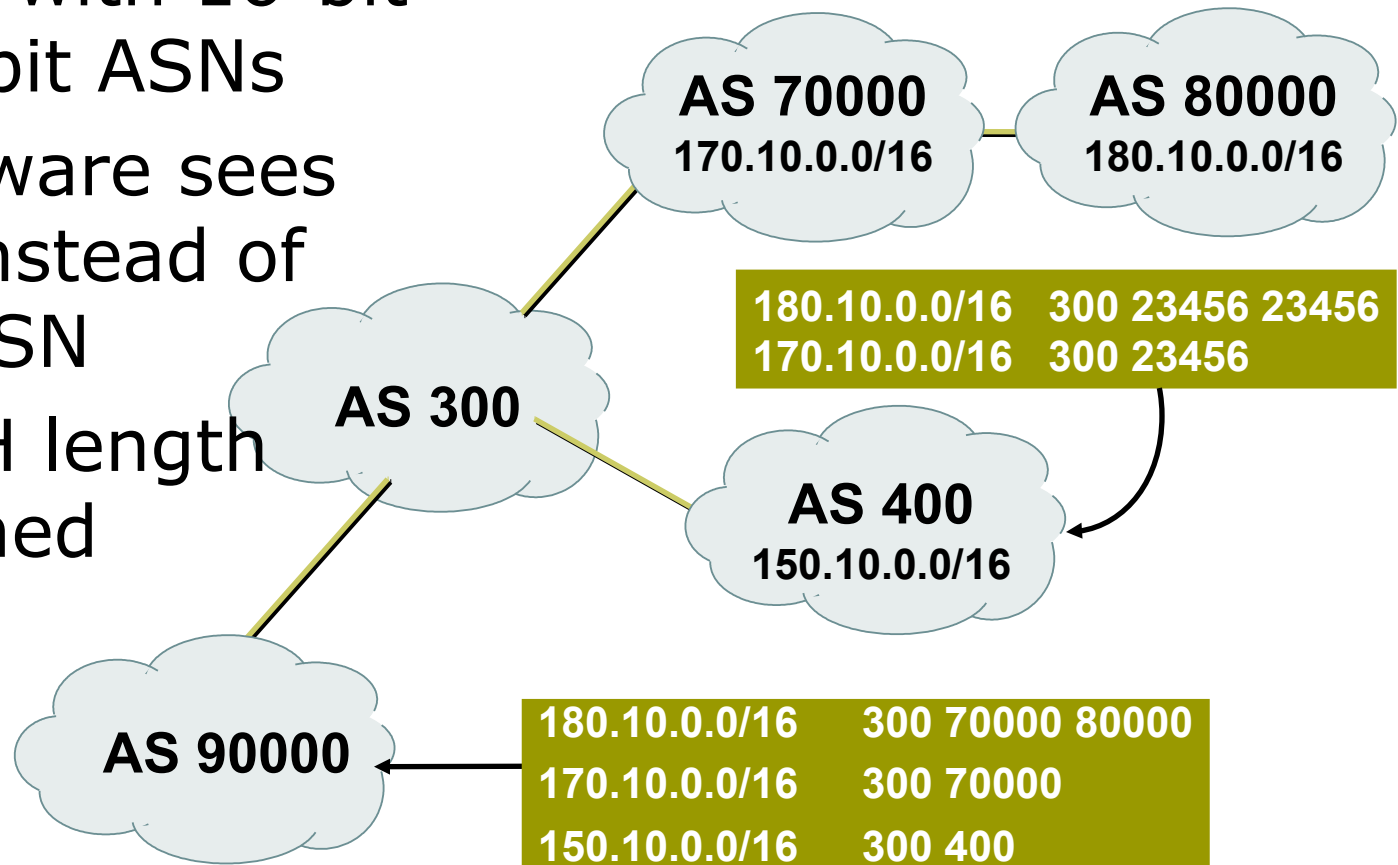
AS-Path Attribute

- ❑ Sequence of ASes a route has traversed
- ❑ Loop detection
- ❑ Apply policy

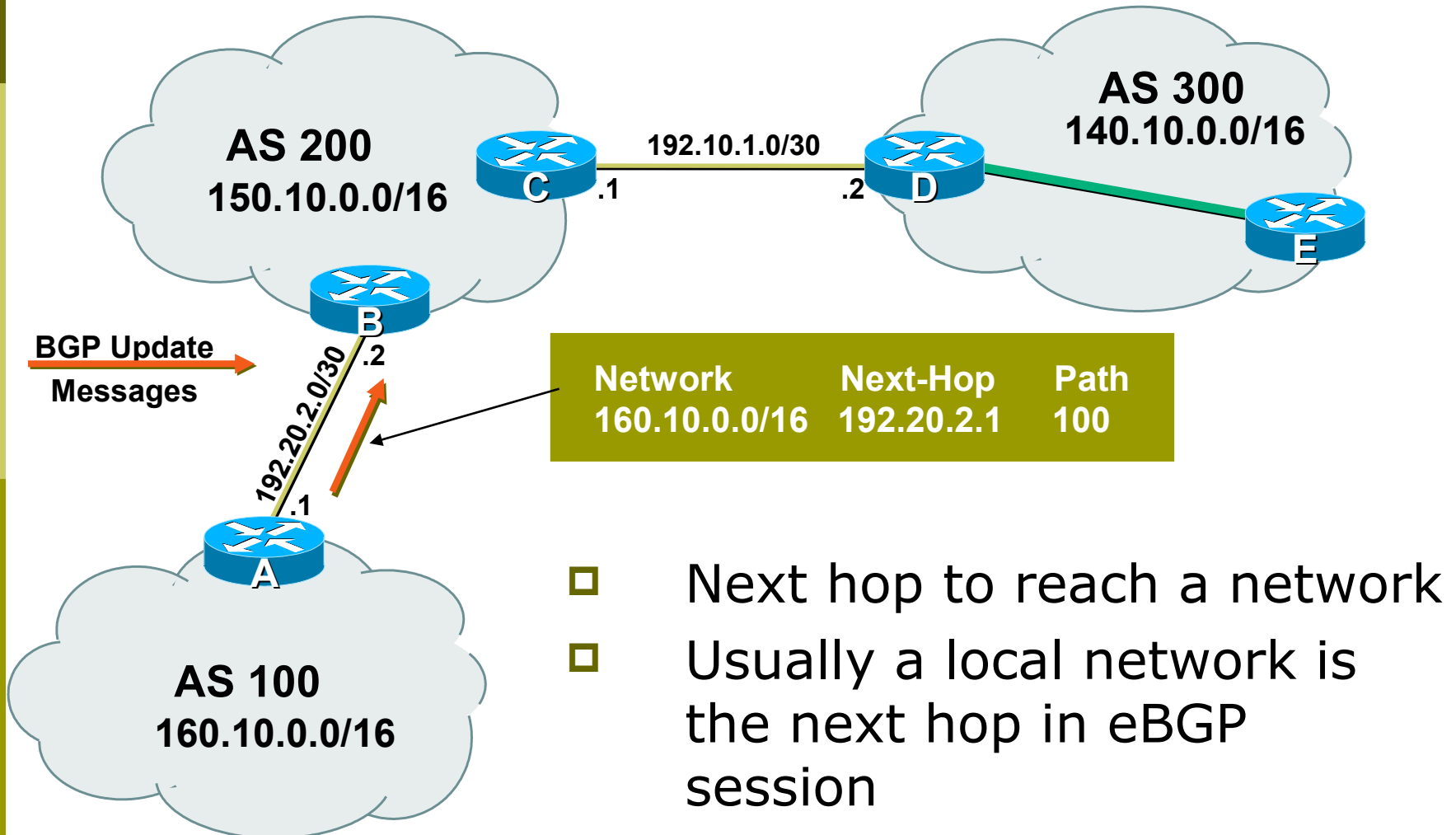


AS-Path (with 16 and 32-bit ASNs)

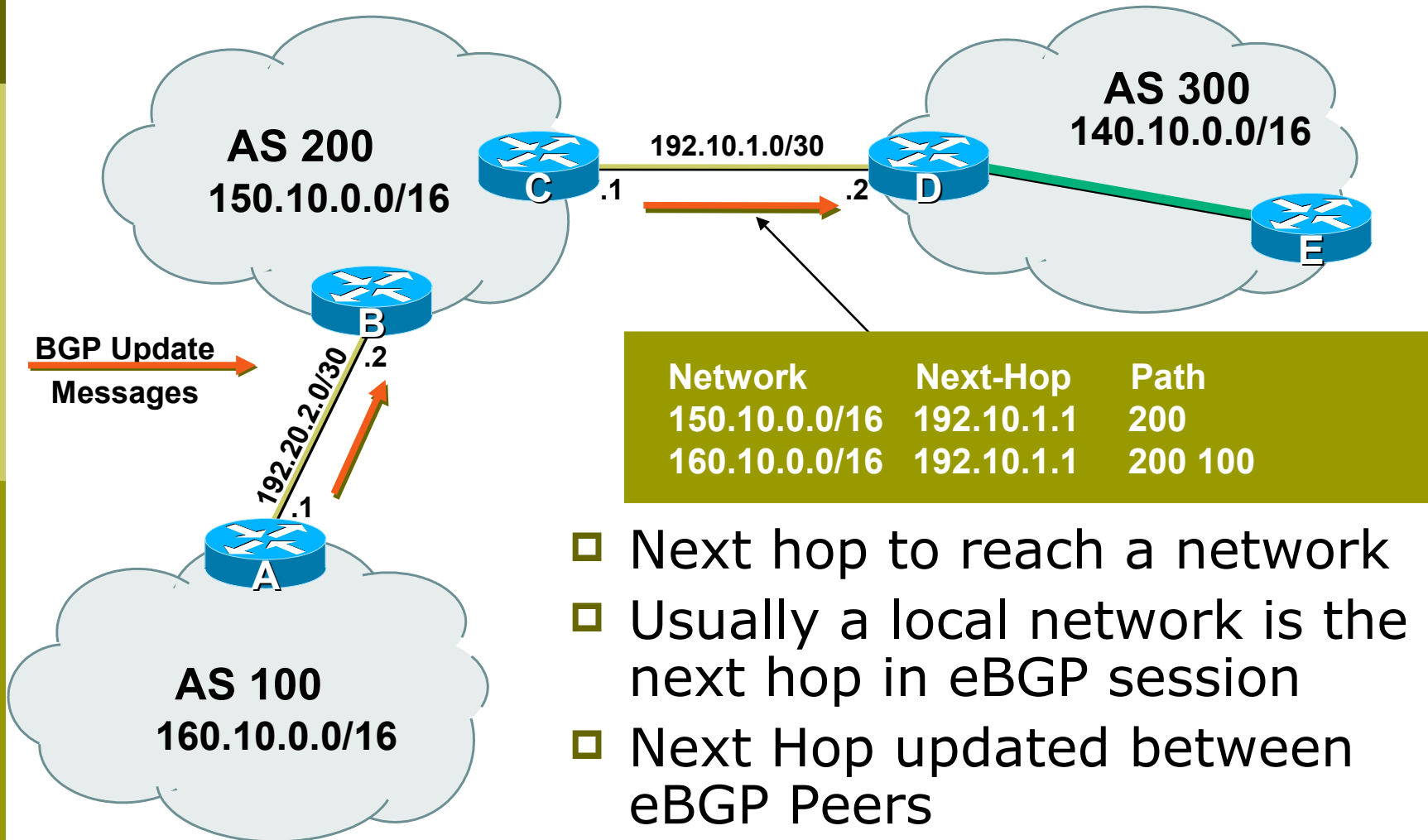
- ❑ Internet with 16-bit and 32-bit ASNs
- ❑ Old software sees 23456 instead of actual ASN
- ❑ AS-PATH length maintained



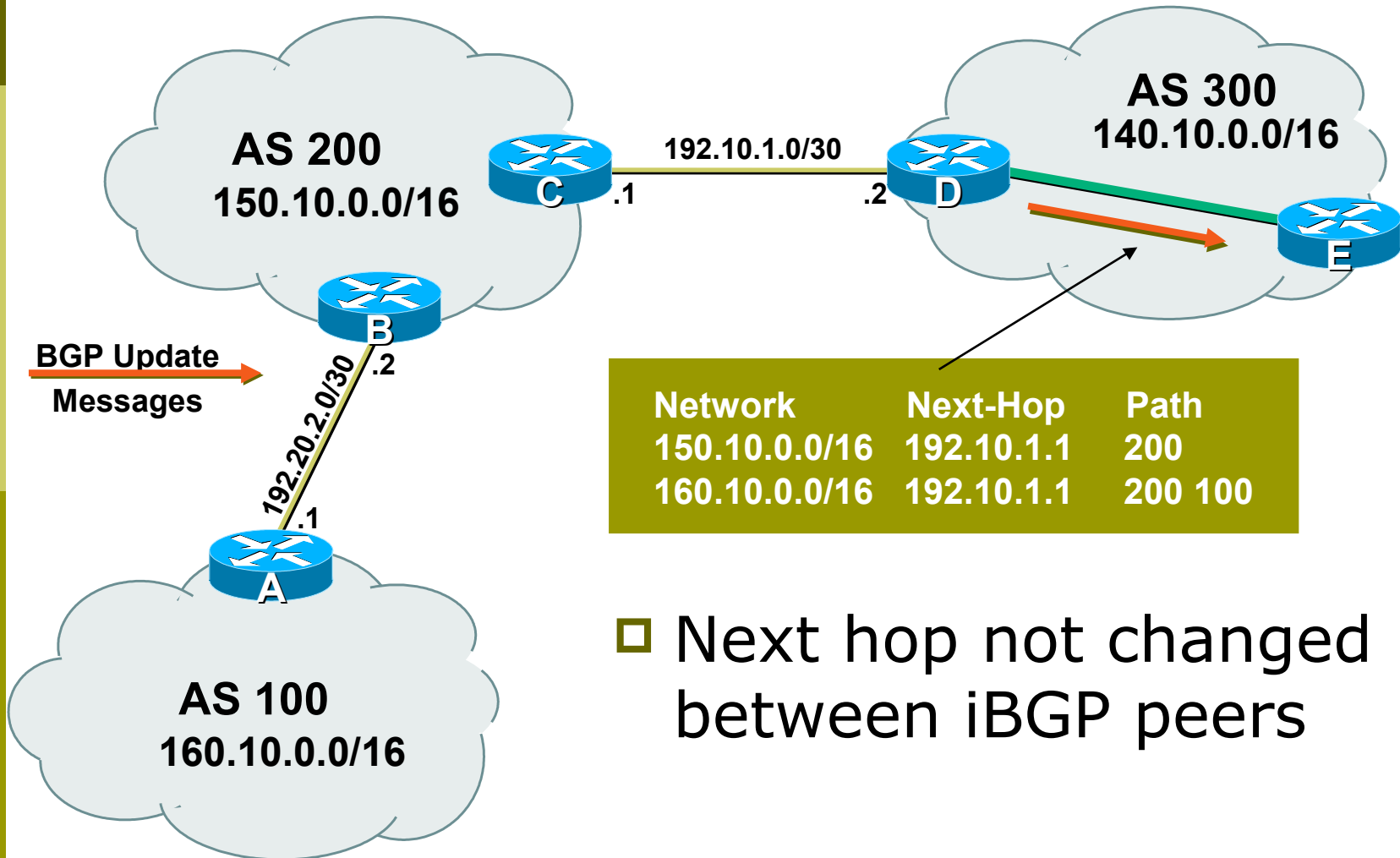
Next Hop Attribute



Next Hop Attribute



Next Hop Attribute



Next Hop Attribute (more)

- ❑ IGP is used to carry route to next hops
- ❑ Recursive route look-up
 - BGP looks into IGP to find out next hop information
 - BGP is not permitted to use a BGP route as the next hop
- ❑ Unlinks BGP from actual physical topology
- ❑ Allows IGP to make intelligent forwarding decision

Next Hop Best Practice

- ❑ Cisco IOS default is for external next-hop to be propagated unchanged to iBGP peers
 - This means that IGP has to carry external next-hops
 - Forgetting means external network is invisible
 - With many eBGP peers, it is extra load on IGP
- ❑ **ISP best practice is to change external next-hop to be that of the local router**

```
neighbor x.x.x.x next-hop-self
```

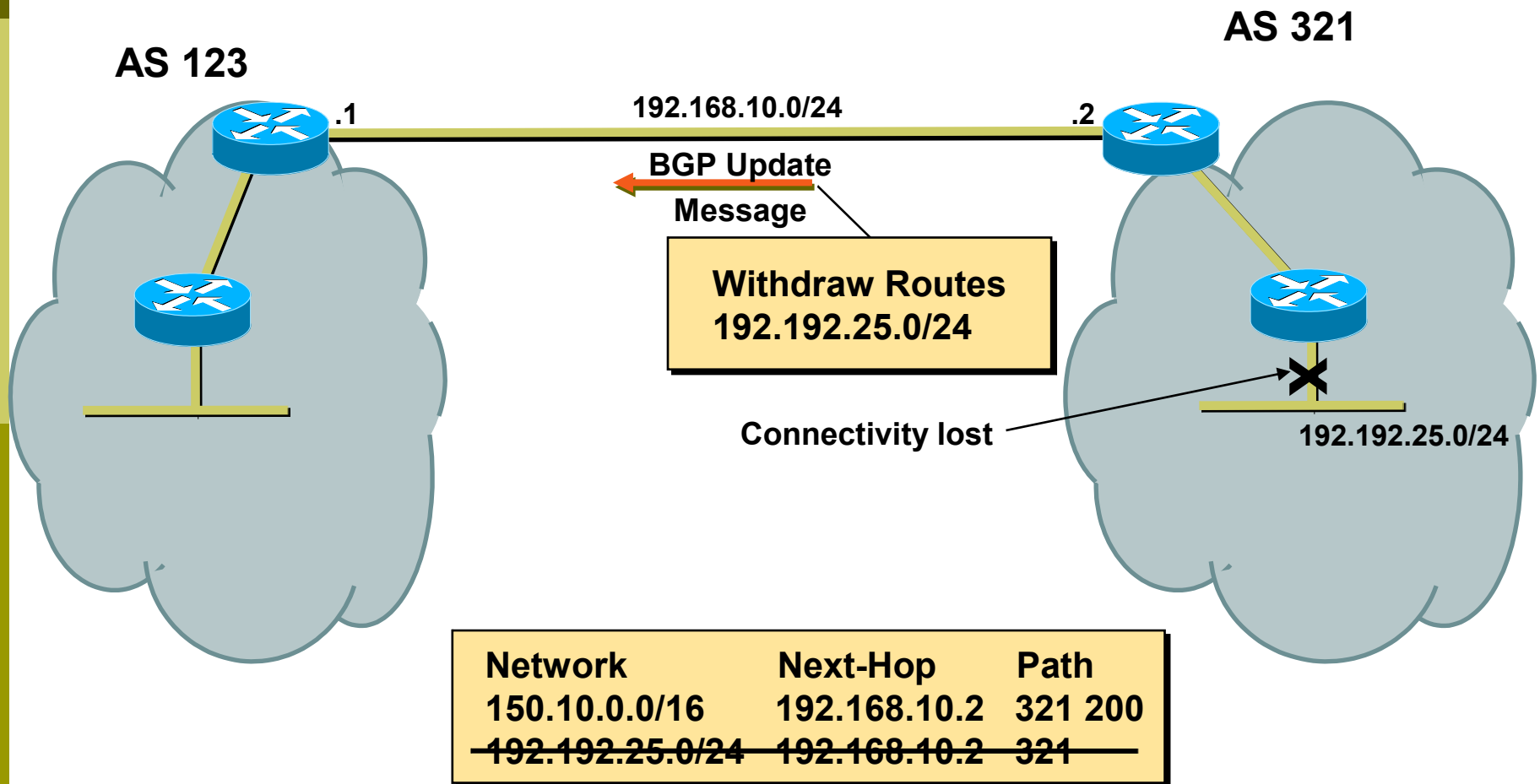
Community Attribute

- 32-bit number
- Conventionally written as two 16-bit numbers separated by colon
 - First half is usually an AS number
 - ISP determines the meaning (if any) of the second half
- Carried in BGP protocol messages
 - Used by administratively-defined filters
 - Not directly used by BGP protocol (except for a few “well known” communities)

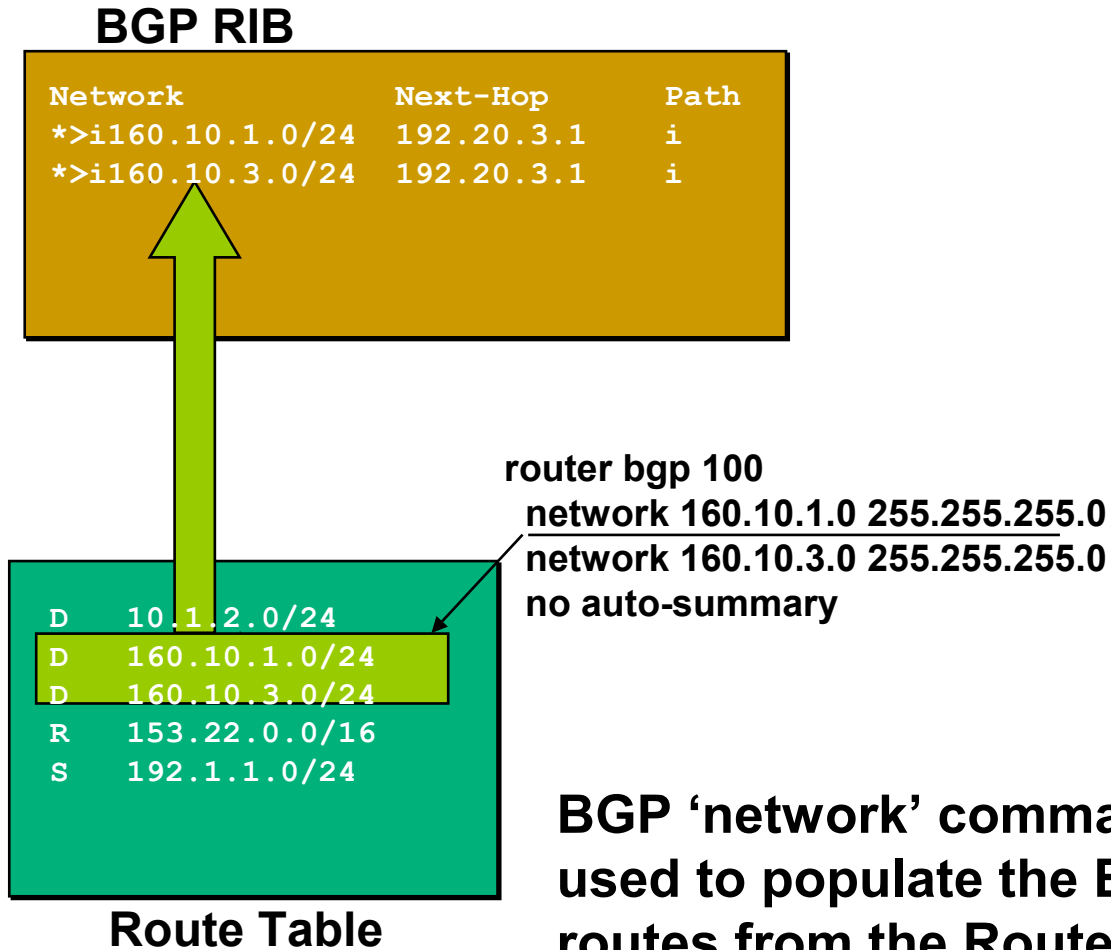
BGP Updates: Withdrawn Routes

- Used to “withdraw” network reachability
- Each withdrawn route is composed of:
 - Network Prefix
 - Mask Length

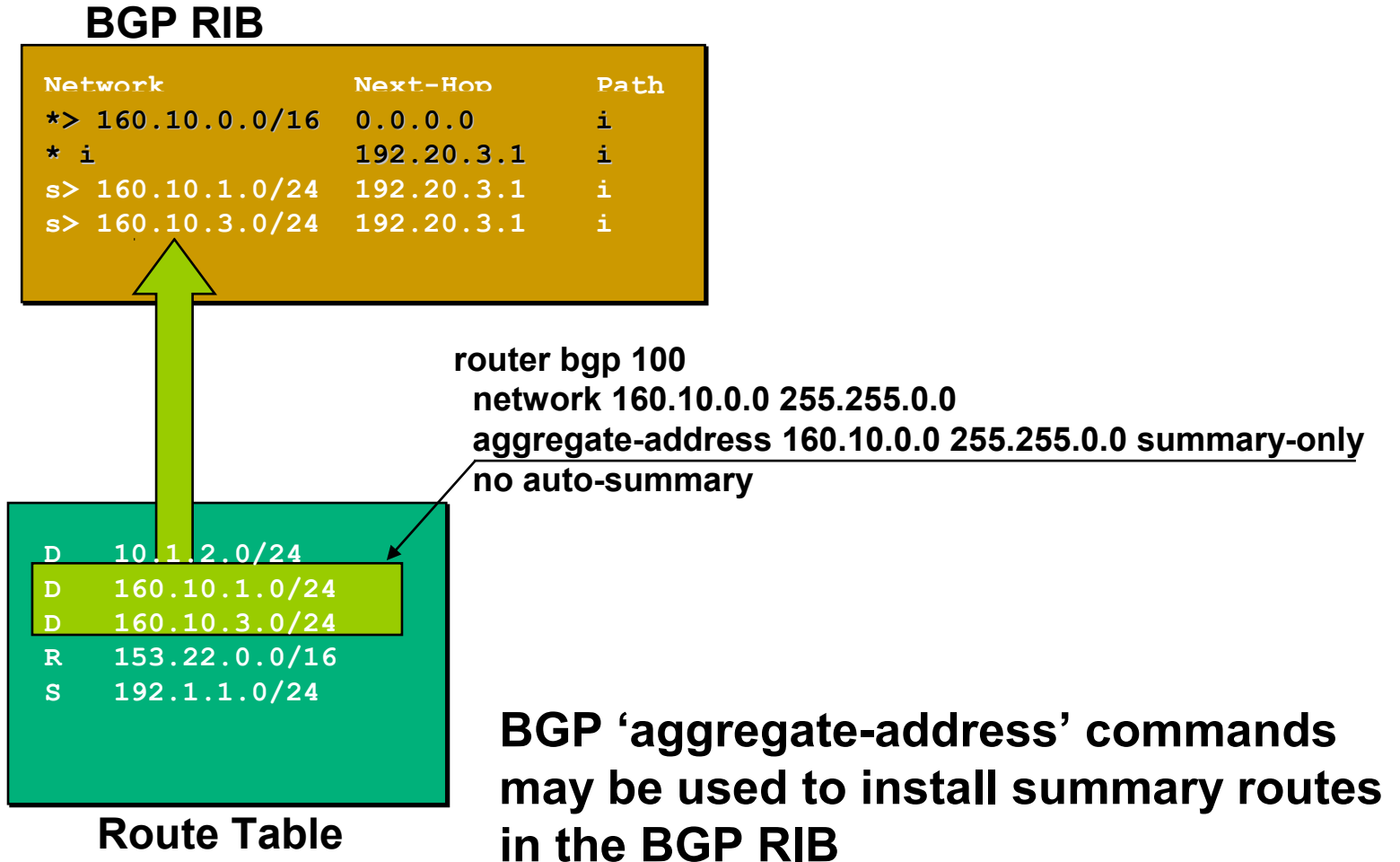
BGP Updates: Withdrawn Routes



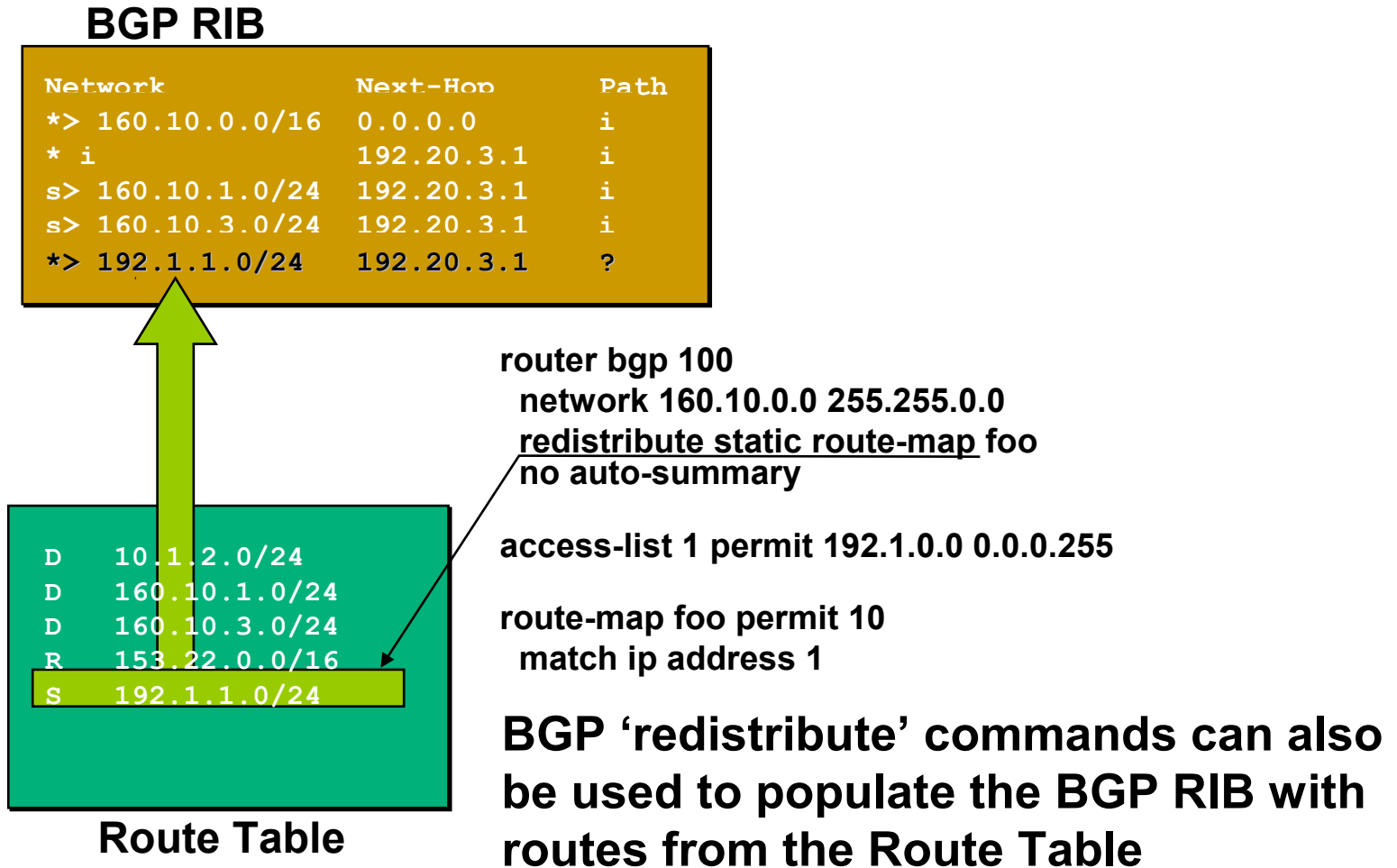
BGP Routing Information Base



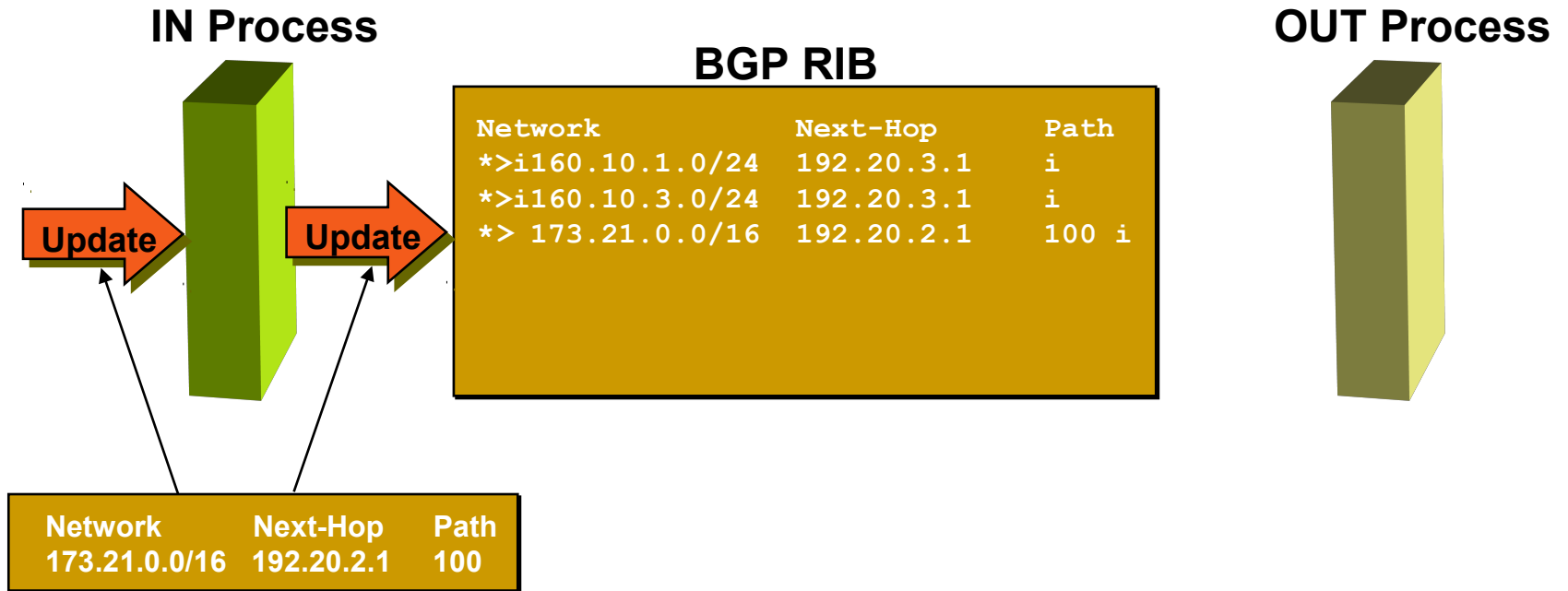
BGP Routing Information Base



BGP Routing Information Base



BGP Routing Information Base



- **BGP “in” process**
 - receives path information from peers
 - results of BGP path selection placed in the BGP table
 - “best path” flagged (denoted by “>”)

BGP Routing Information Base

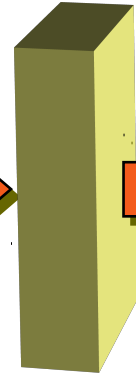
IN Process



BGP RIB

Network	Next-Hop	Path
*>i160.10.1.0/24	192.20.3.1	i
*>i160.10.3.0/24	192.20.3.1	i
*> 173.21.0.0/16	192.20.2.1	100

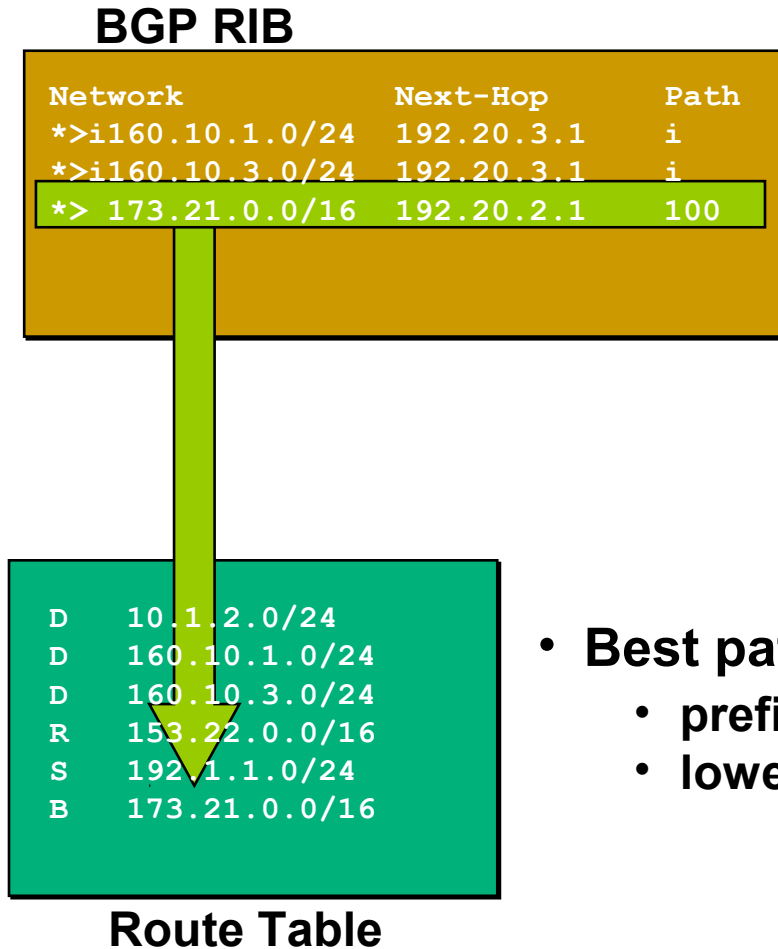
OUT Process



Network	Next-Hop	Path
160.10.1.0/24	192.20.3.1	200
160.10.3.0/24	192.20.3.1	200
173.21.0.0/16	192.20.2.1	200 100

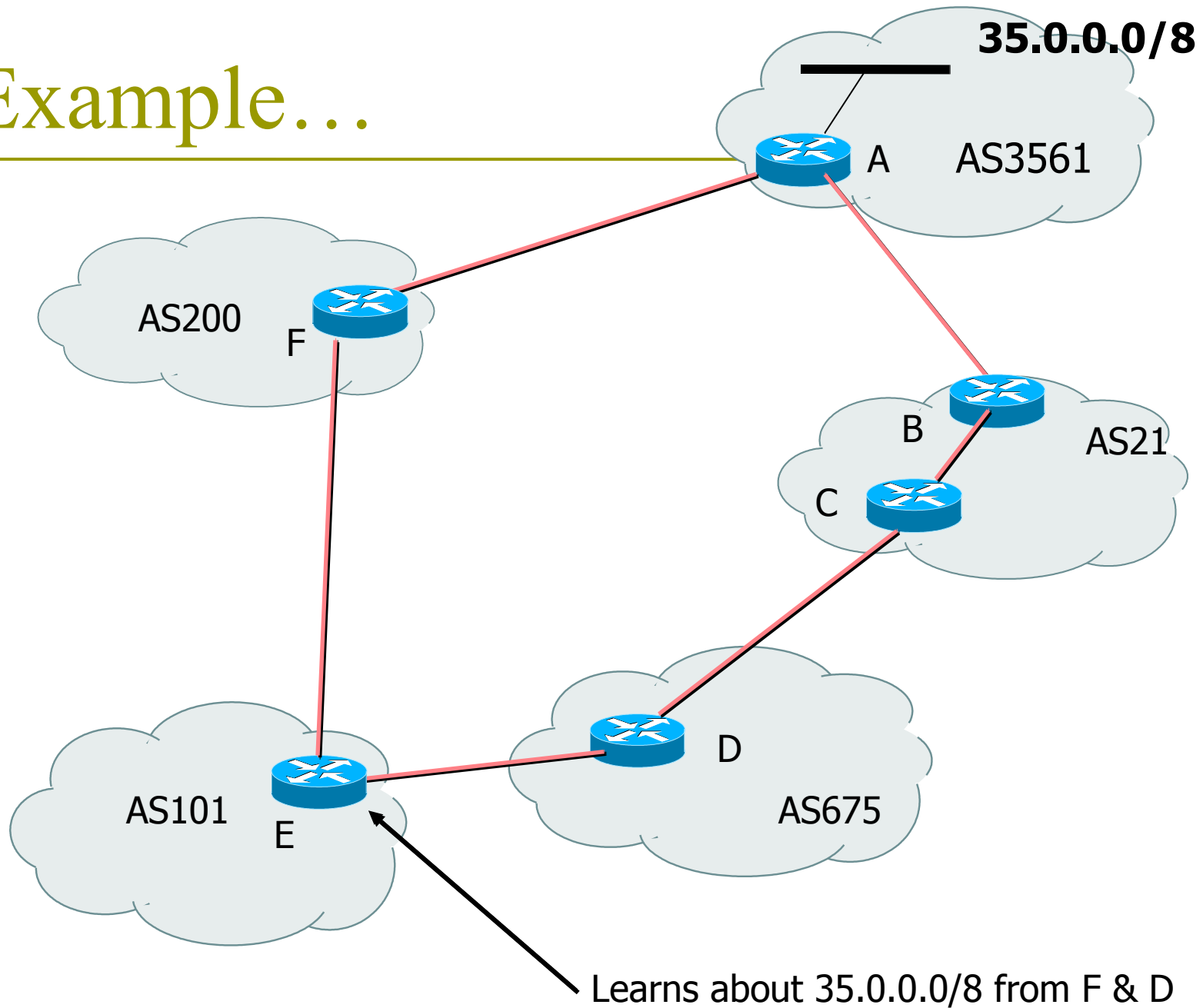
- BGP “out” process
 - builds update using info from RIB
 - may modify update based on config
 - Sends update to peers

BGP Routing Information Base



- **Best paths installed in routing table if:**
 - prefix and prefix length are unique
 - lowest “protocol distance”

An Example...



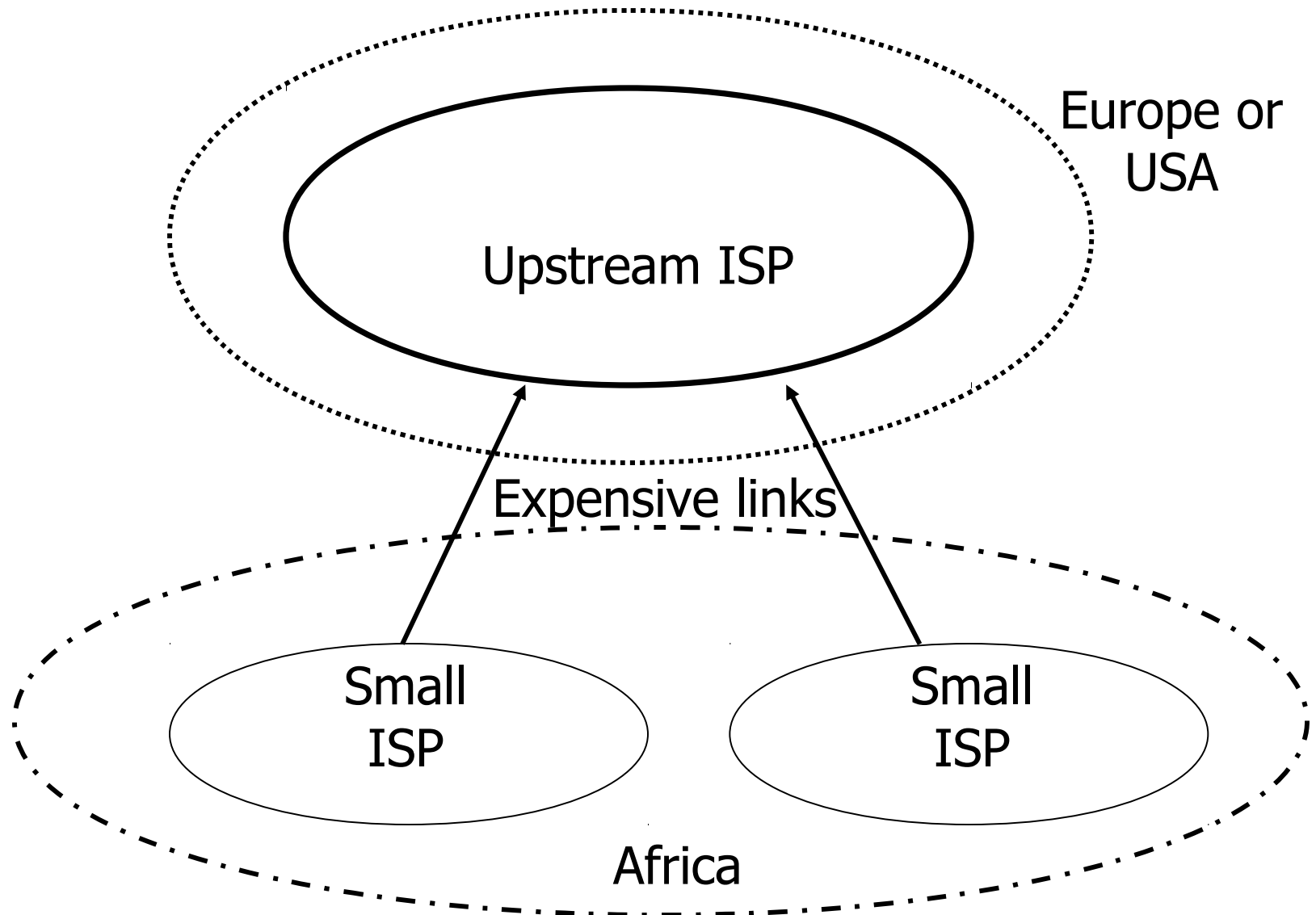
BGP Case Study 2 and Exercise 2

Small ISPs in the same locality
connect to each other

Case Study 2: Another ISP in the same country

- Similar setup
- Traffic between you and them goes over
 - Your expensive line
 - Their expensive line
- Traffic can be significant
 - Same language/culture
 - Traffic between your and their customers
- This wastes money

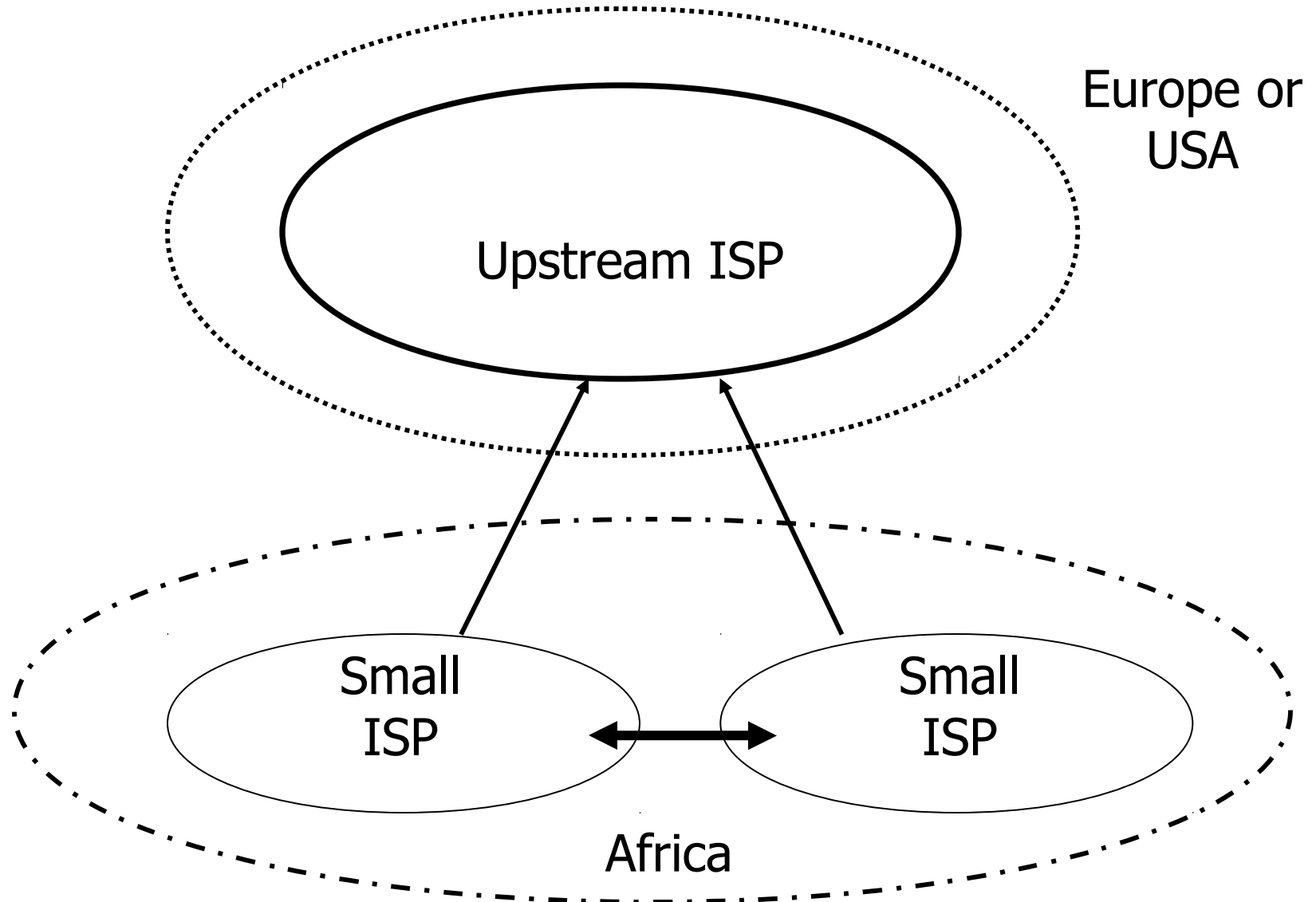
Case Study 2: Another ISP in the same country



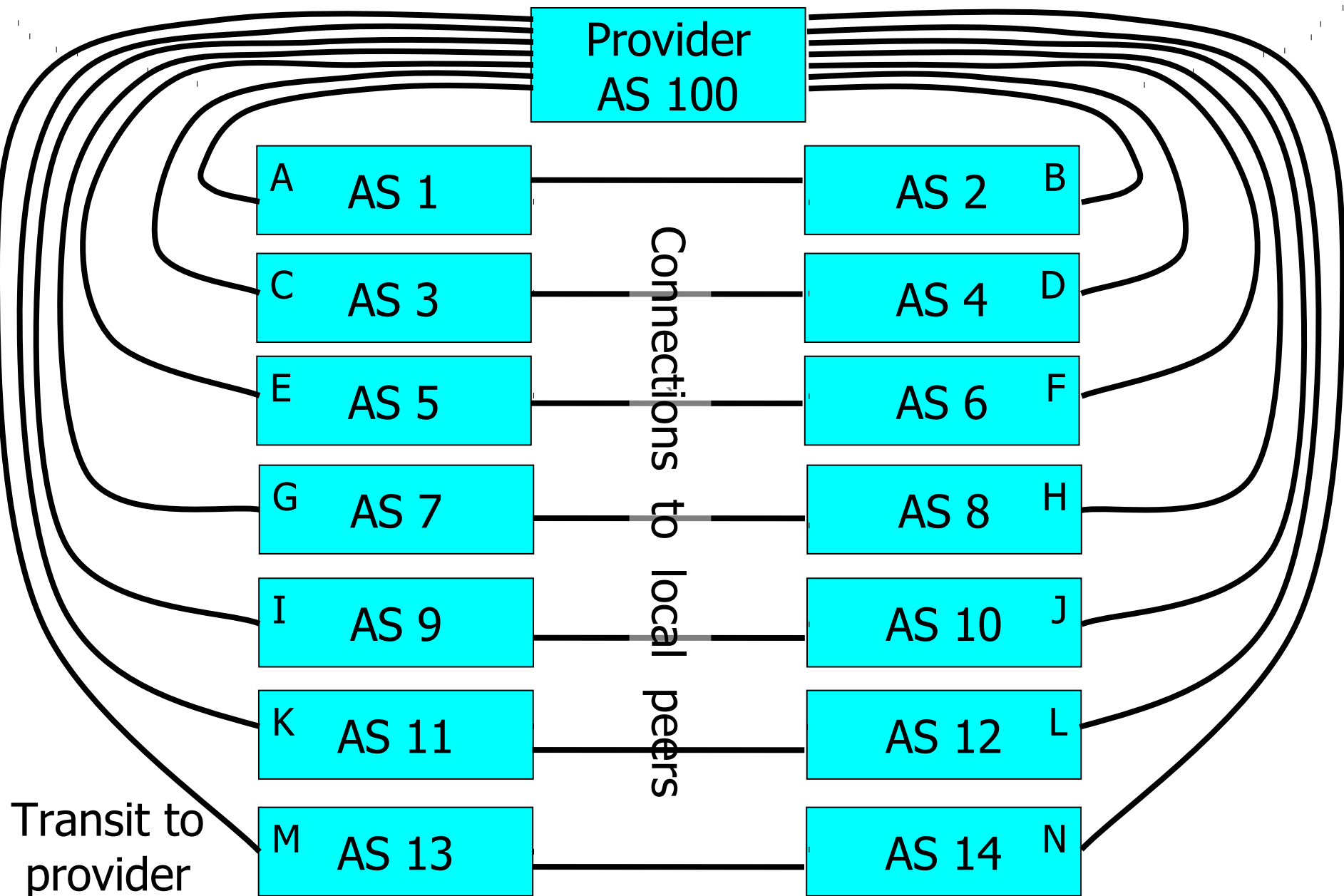
Case Study 2: Bringing down costs

- Local (national) links are usually much cheaper than international ones
- Might be interesting to get direct link between you and them
 - Saving traffic on expensive lines
 - better performance, cheaper
 - No need to send traffic to other ISP down the street via New York!

Case Study 2: Keeping Local Traffic Local



Exercise 2: Connect to another local ISP



Exercise 2: BGP configuration

- Refer to “BGP cheat sheet”.
- Add to previous configuration.
- Connect cable to local peer.
- No filters yet.

Exercise 2: What you should see

- You should see multiple routes to each destination
 - direct route to your peer
 - transit route through provider (AS 100)
 - any more?

Exercise 2: What you should see

- To see forwarding table, try:
 - IPv4: "show ip route"
 - IPv6: "show ipv6 route"
- To see BGP information, try:
 - IPv4: "show ip bgp"
 - IPv6: "show bgp ipv6"
- Look at the "next hop" and "AS path"
- Try some pings and traceroutes.

Exercise 2: Do you see transit routes through your peers?

- Are your peer ASes sending you transit routes as well as peering routes?
 - Do you want transit through them?
- Are you sending transit routes to your peers?
 - Do you want your peers to have transit through you?
- We will fix this later

BGP Part 8



Routing Policy
Filtering

Terminology: “Policy”

- Where do you want your traffic to go?
 - It is difficult to get what you want, but you can try
- Control of how you accept and send routing updates to neighbors
 - prefer cheaper connections, load-sharing, etc.
- Accepting routes from some ISPs and not others
- Sending some routes to some ISPs and not others
- Preferring routes from some ISPs over others

Routing Policy

□ Why?

- To steer traffic through preferred paths
- Inbound/Outbound prefix filtering
- To enforce Customer-ISP agreements

□ How?

- AS based route filtering – filter list
- Prefix based route filtering – prefix list
- BGP attribute modification – route maps
- Complex route filtering – route maps

Filter list rules:

Regular Expressions

- ❑ Regular Expression is a pattern to match against an input string
- ❑ Used to match against AS-path attribute
- ❑ ex: `^3561_.*_100_.*_1$`
- ❑ Flexible enough to generate complex filter list rules

Regular expressions (cisco specific)

^ matches start

\$ matches end

_ matches start, or end, or space (boundary between words or numbers)

.* matches anything (0 or more characters)

.+ matches anything (1 or more characters)

[0-9] matches any number between 0 and 9

^\$ matches the local AS (AS path is empty)

There are many more possibilities

Filter list – using as-path access list

- Listen to routes originated by AS 3561. Implicit deny everything else inbound.
- Don't announce routes originated by AS 35, but announce everything else (outbound).

```
ip as-path access-list 1 permit _3561$  
ip as-path access-list 2 deny _35$  
ip as-path access-list 2 permit .*
```

```
router bgp 100  
  neighbor 171.69.233.33 remote-as 33  
  neighbor 171.69.233.33 filter-list 1 in  
  neighbor 171.69.233.33 filter-list 2 out
```

Policy Control – Prefix Lists

- Per neighbor prefix filter
 - incremental configuration
- High performance access list
- Inbound or Outbound
- Based upon network numbers (using CIDR address/mask format)
- First relevant “allow” or “deny” rule wins
- Implicit Deny All as last entry in list

Prefix Lists – Examples

- ❑ Deny default route

```
ip prefix-list Example deny 0.0.0.0/0
```

- ❑ Permit the prefix 35.0.0.0/8

```
ip prefix-list Example permit 35.0.0.0/8
```

- ❑ Deny the prefix 172.16.0.0/12, and all more-specific routes

```
ip prefix-list Example deny 172.16.0.0/12 ge 12
```

- “ge 12” means “prefix length /12 or longer”. For example, 172.17.0.0/16 will also be denied.

- ❑ In 192.0.0.0/8, allow any /24 or shorter prefixes

```
ip prefix-list Example permit 192.0.0.0/8 le 24
```

- This will not allow any /25, /26, /27, /28, /29, /30, /31 or /32

Prefix Lists – More Examples

- In 192/8 deny /25 and above

```
ip prefix-list Example deny 192.0.0.0/8 ge 25
```

- This denies all prefix sizes /25, /26, /27, /28, /29, /30, /31 and /32 in the address block 192.0.0.0/8

- It has the same effect as the previous example

- In 192/8 permit prefixes between /12 and /20

```
ip prefix-list Example permit 192.0.0.0/8 ge 12 le 20
```

- This denies all prefix sizes /8, /9, /10, /11, /21, /22 and higher in the address block 193.0.0.0/8

- Permit all prefixes

```
ip prefix-list Example 0.0.0.0/0 le 32
```

Policy Control Using Prefix Lists

□ Example Configuration

```
router bgp 200
  network 215.7.0.0
  neighbor 220.200.1.1 remote-as 210
  neighbor 220.200.1.1 prefix-list PEER-IN in
  neighbor 220.200.1.1 prefix-list PEER-OUT out
!
ip prefix-list PEER-IN deny 218.10.0.0/16
ip prefix-list PEER-IN permit 0.0.0.0/0 le 32
ip prefix-list PEER-OUT permit 215.7.0.0/16
ip prefix-list PEER-OUT deny 0.0.0.0/0 le 32
```

- Accept everything except our network from our peer
- Send only our network to our peer

Prefix-lists in IPv6

- Prefix-lists in IPv6 work the same way as they do in IPv4
 - Caveat: ipv6 prefix-lists cannot be used for ipv4 neighbours - and vice-versa
 - Syntax is very similar, for example:

```
ip prefix-list ipv4-ebgp permit 0.0.0.0/0 le 32
ip prefix-list v4out permit 172.16.0.0/16
!
ipv6 prefix-list ipv6-ebgp permit ::/0 le 128
ipv6 prefix-list v6out permit 2001:db8::/32
```


Policy Control – Route Maps

- A route-map is like a “program” for Cisco IOS
- Has “line” numbers, like programs
- Each line is a separate condition/action
- Concept is basically:
 - if *match* then do *expression* and *exit*
 - else
 - if *match* then do *expression* and *exit*
 - else *etc*

Route-map match & set clauses

□ Match Clauses

- AS-path
- Community
- IP address

□ Set Clauses

- AS-path prepend
- Community
- Local-Preference
- MED
- Origin
- Weight
- Others...

Route Map: Example One

```
router bgp 300
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 route-map SETCOMMUNITY out
!
route-map SETCOMMUNITY permit 10
  match ip address 1
  match community 1
  set community 300:100
!
access-list 1 permit 35.0.0.0
ip community-list 1 permit 100:200
```

Route Map: Example Two

- Example Configuration as AS PATH prepend

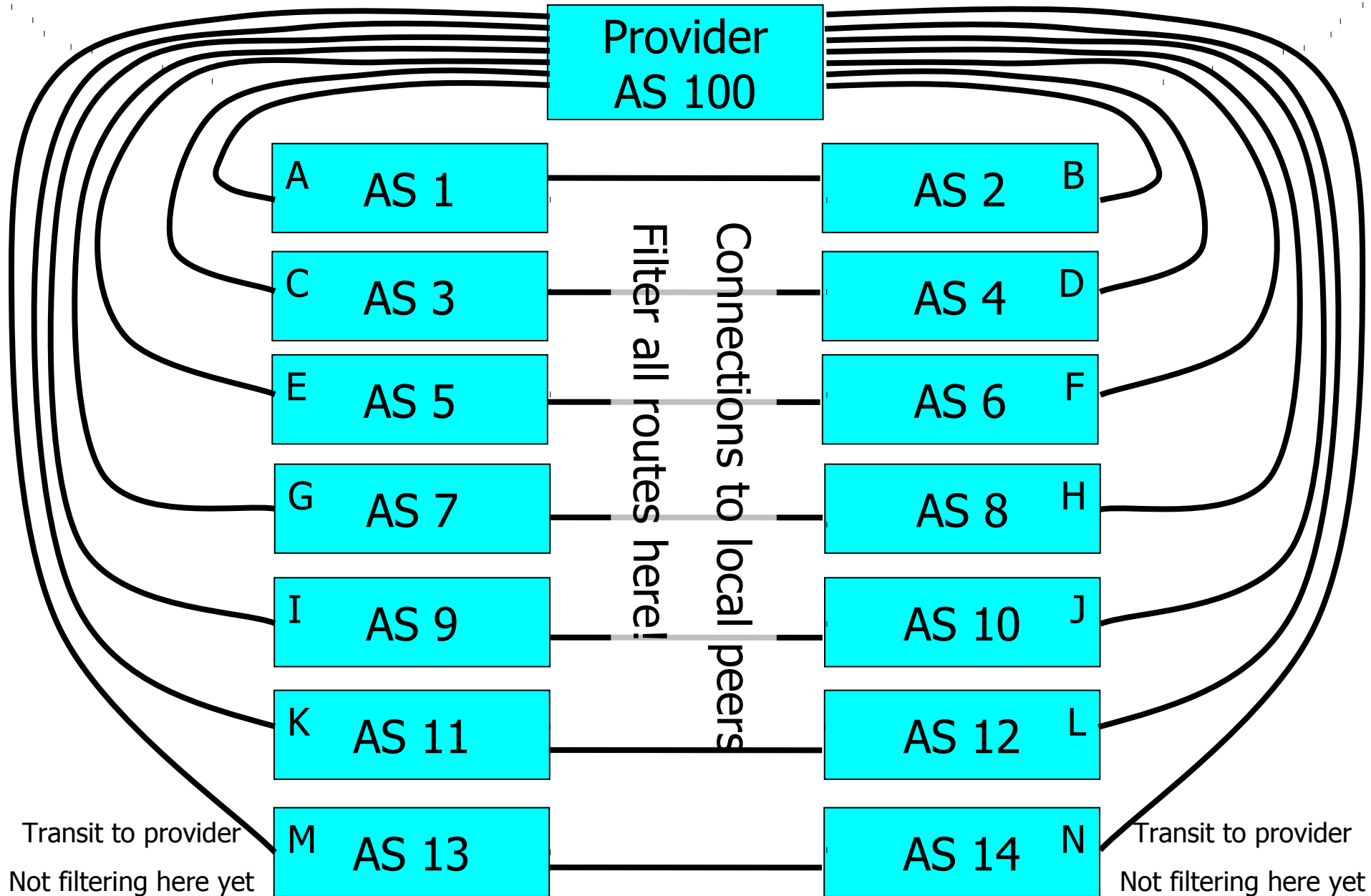
```
router bgp 300
  network 215.7.0.0
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 route-map SETPATH out
!
route-map SETPATH permit 10
  set as-path prepend 300 300
```

- Use your own AS number for prepending
 - Otherwise BGP loop detection will cause disconnects

BGP Exercise 3

Filtering peer routes using AS-path regular expression

Exercise 3: Filtering peer routes using AS-path



Exercise 3: Filtering peer routes using AS-path

- Create “ip as-path access-list <number>” to match your peer’s routes
 - ip as-path access-list 1 permit ^1\$
- Create “ip as-path access-list <number>” to match your own routes
 - ip as-path access-list 2 permit ^\$
- Apply the filters to both IPv4 and IPv6 peers:
 - “neighbor <address> filter-list 1 in”
 - “neighbor <address> filter-list 2 out”
 - As-path filters are protocol independent, so the same filter can be applied to both IPv4 and IPv6 peers!
- Apply the outbound filter to the AS100 upstream
 - “neighbor <upstream-addr> filter-list 2 out”

Exercise 3: What you should see

- From peers: only their routes, no transit
 - They send all routes, but you filter
- To peers: your routes
 - They will ignore the transit routes if you mistakenly send them
- From upstream: all routes
- To upstream: your routes, no transit

Exercise 3: Did it work?

- IPv4 show commands:
 - “show ip route” – your forwarding table
 - “show ip bgp” – your BGP table
 - “show ip bgp neighbor xxx received-routes” – from your neighbour before filtering
 - “show ip bgp neighbor xxx routes” – from neighbour, after filtering
 - “show ip bgp neighbor advertised-routes” – to neighbour, after filtering

Exercise 3: Did it work?

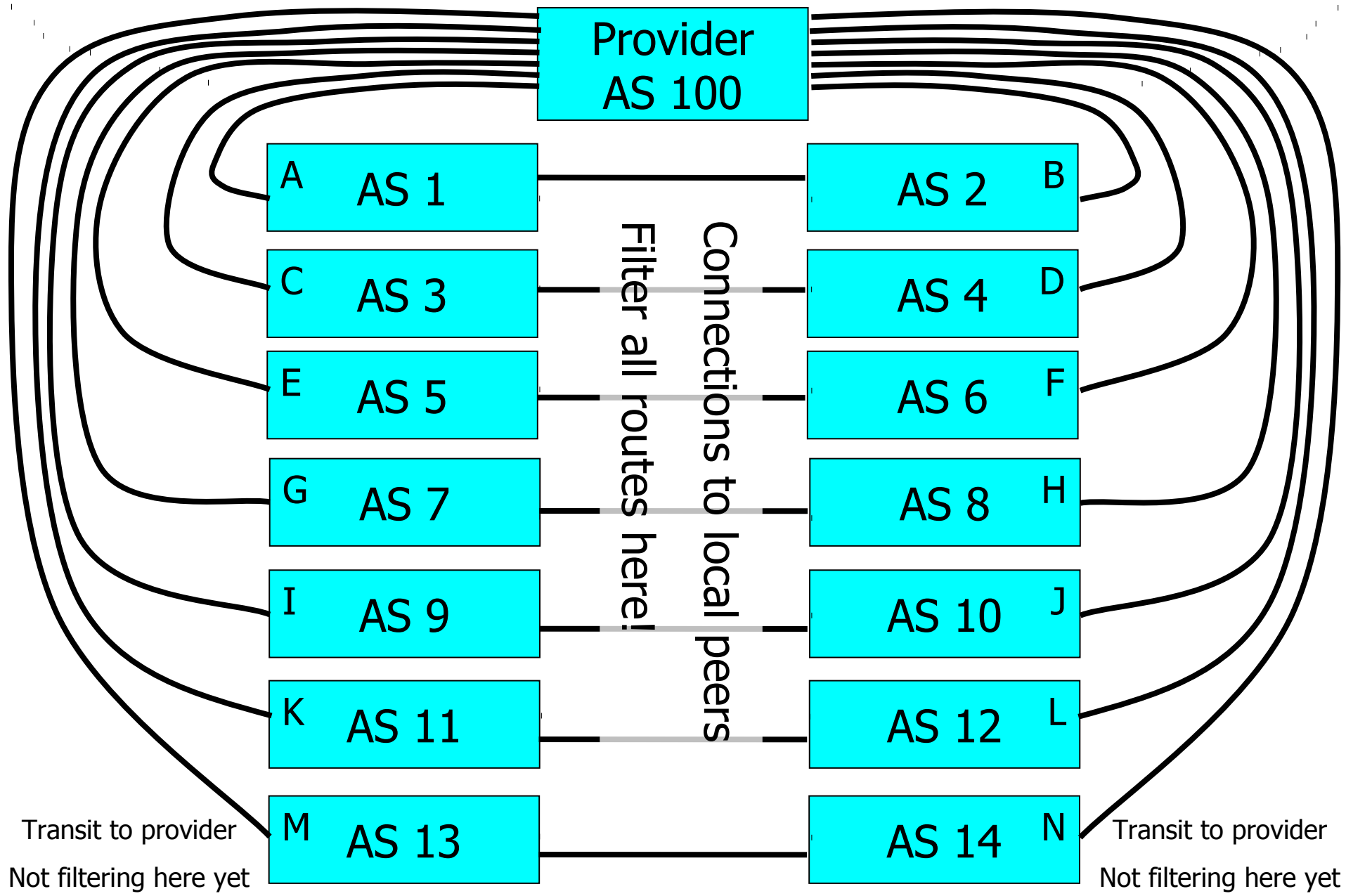
- IPv6 show commands:
 - “show ipv6 route” – your forwarding table
 - “show bgp ipv6” – your BGP table
 - “show bgp ipv6 neighbor xxx received-routes” – from your neighbour before filtering
 - “show bgp ipv6 neighbor xxx routes” – from neighbour, after filtering
 - “show bgp ipv6 neighbor advertised-routes” – to neighbour, after filtering

BGP Exercise 4



Filtering peer routes using
prefix-lists

Exercise 4: Filtering peer routes using prefix-lists



Exercise 4: Filtering peer routes using prefix-list

- ❑ Create "ip prefix-list my-routes" to match your own routes
- ❑ Create "ip prefix-list peer-as-xxx" to match your peer's routes
- ❑ Apply the filters to your peers
 - "neighbor xxx prefix-list my-routes out"
 - "neighbor xxx prefix-list peer-as-xxx in"
- ❑ Apply the outbound filter to your upstream provider
 - "neighbor xxx prefix-list my-routes out"

Exercise 4: Filtering peer routes using prefix-list

- ❑ Create "ipv6 prefix-list myv6-routes" to match your own routes
- ❑ Create "ipv6 prefix-list peer-as-xxx-v6" to match your peer's routes
- ❑ Apply the filters to your IPv6 peers
 - "neighbor xxx prefix-list myv6-routes out"
 - "neighbor xxx prefix-list peer-as-xxx-v6 in"
- ❑ Apply the outbound filter to your upstream provider
 - "neighbor xxx prefix-list myv6-routes out"

Exercise 4: What you should see

- From peers: only their routes, no transit
- To peers: only your routes, no transit
- From upstream: all routes
- To upstream: only your routes, no transit

- We still trust the upstream provider too much. Should filter it too!
 - See “ip prefix-list sanity-filter” and “ipv6 prefix-list v6sanity-filter” in the cheat sheet

Exercise 4: Did it work?

- IPv4 show commands:
 - “show ip route” – your forwarding table
 - “show ip bgp” – your BGP table
 - “show ip bgp neighbor xxx received-routes” – from your neighbour before filtering
 - “show ip bgp neighbor xxx routes” – from neighbour, after filtering
 - “show ip bgp neighbor advertised-routes” – to neighbour, after filtering

Exercise 4: Did it work?

- IPv6 show commands:
 - “show ipv6 route” – your routing table
 - “show bgp ipv6” – your BGP table
 - “show bgp ipv6 neighbor xxx received-routes” – from your neighbour before filtering
 - “show bgp ipv6 neighbor xxx routes” – from neighbour, after filtering
 - “show bgp ipv6 neighbor advertised-routes” – to neighbour, after filtering

BGP Part 9



More detail than you want

BGP Attributes
Synchronization
Path Selection

BGP Path Attributes: Why ?

- ❑ Encoded as Type, Length & Value (TLV)
- ❑ Transitive/Non-Transitive attributes
- ❑ Some are mandatory
- ❑ Used in path selection
- ❑ To apply policy for steering traffic

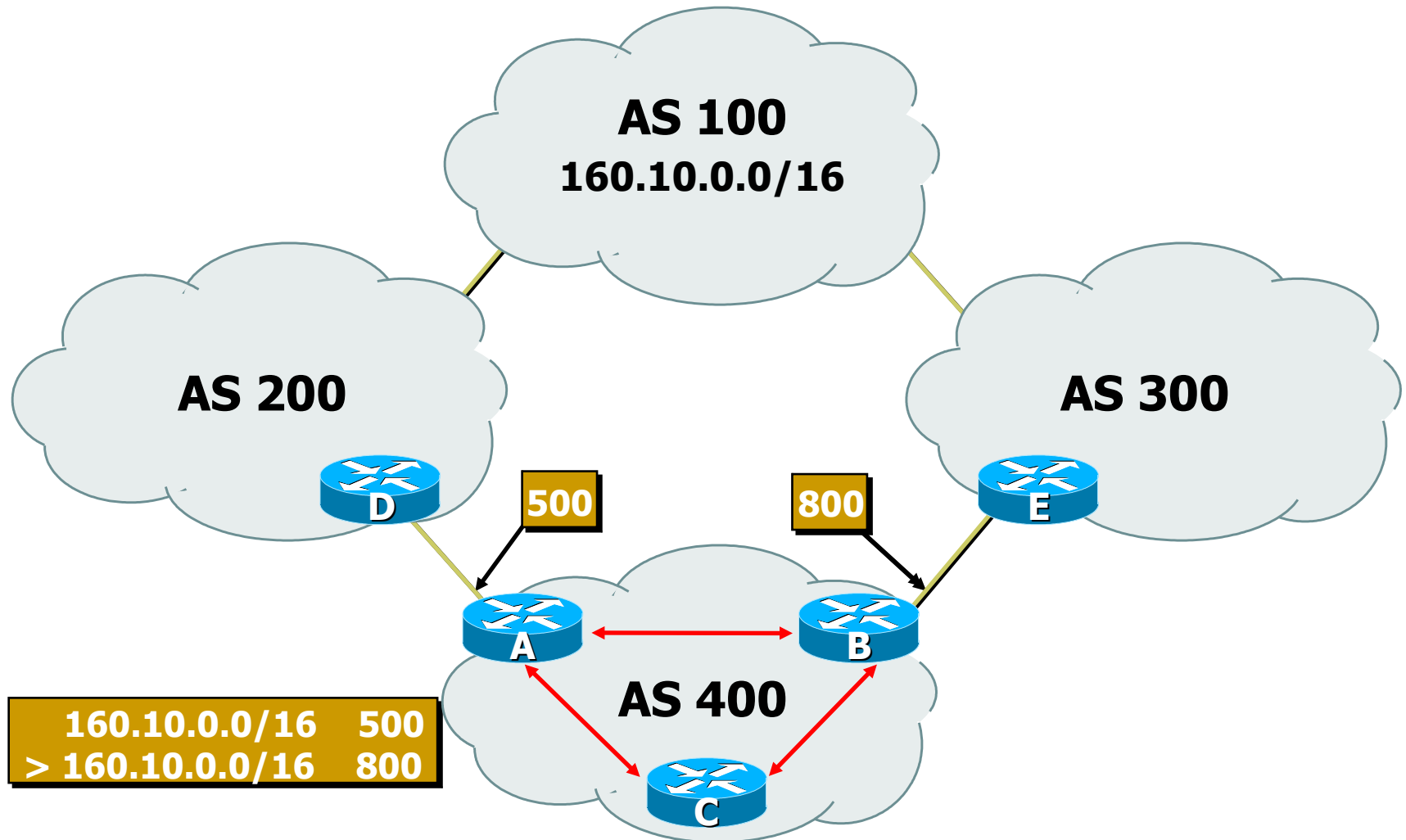
BGP Attributes

- Used to convey information associated with NLRI
 - AS path
 - Next hop
 - Local preference
 - Multi-Exit Discriminator (MED)
 - Community
 - Origin
 - Aggregator

Local Preference

- ❑ Not used by eBGP, mandatory for iBGP
- ❑ Default value of 100 on Cisco IOS
- ❑ Local to an AS
- ❑ Used to prefer one exit over another
- ❑ Path with highest local preference wins

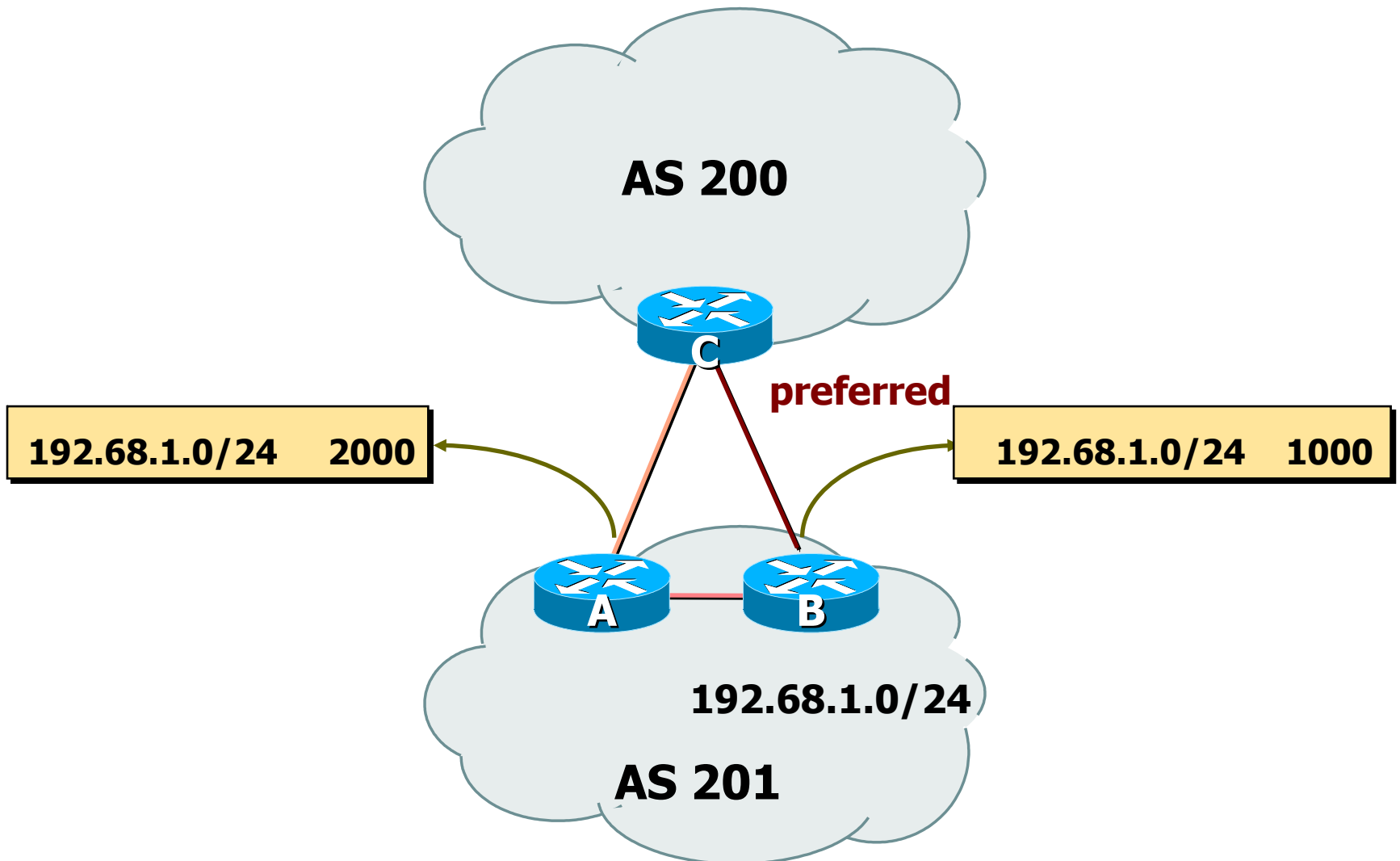
Local Preference



Multi-Exit Discriminator

- Non-transitive
- Represented as a numerical value
 - Range 0x0 – 0xffffffff
- Used to convey relative preference of entry points to an AS
- Comparable if the paths are from the same AS
- Path with the lowest MED wins
- IGP metric can be conveyed as MED

Multi-Exit Discriminator (MED)



Origin

- Conveys the origin of the prefix
 - **Historical** attribute
- Three values:
 - IGP – from BGP network statement
 - E.g. – *network 35.0.0.0*
 - EGP – redistributed from EGP (not used today)
 - Incomplete – redistributed from another routing protocol
 - E.g. – *redistribute static*
- IGP < EGP < incomplete
 - Lowest origin code wins

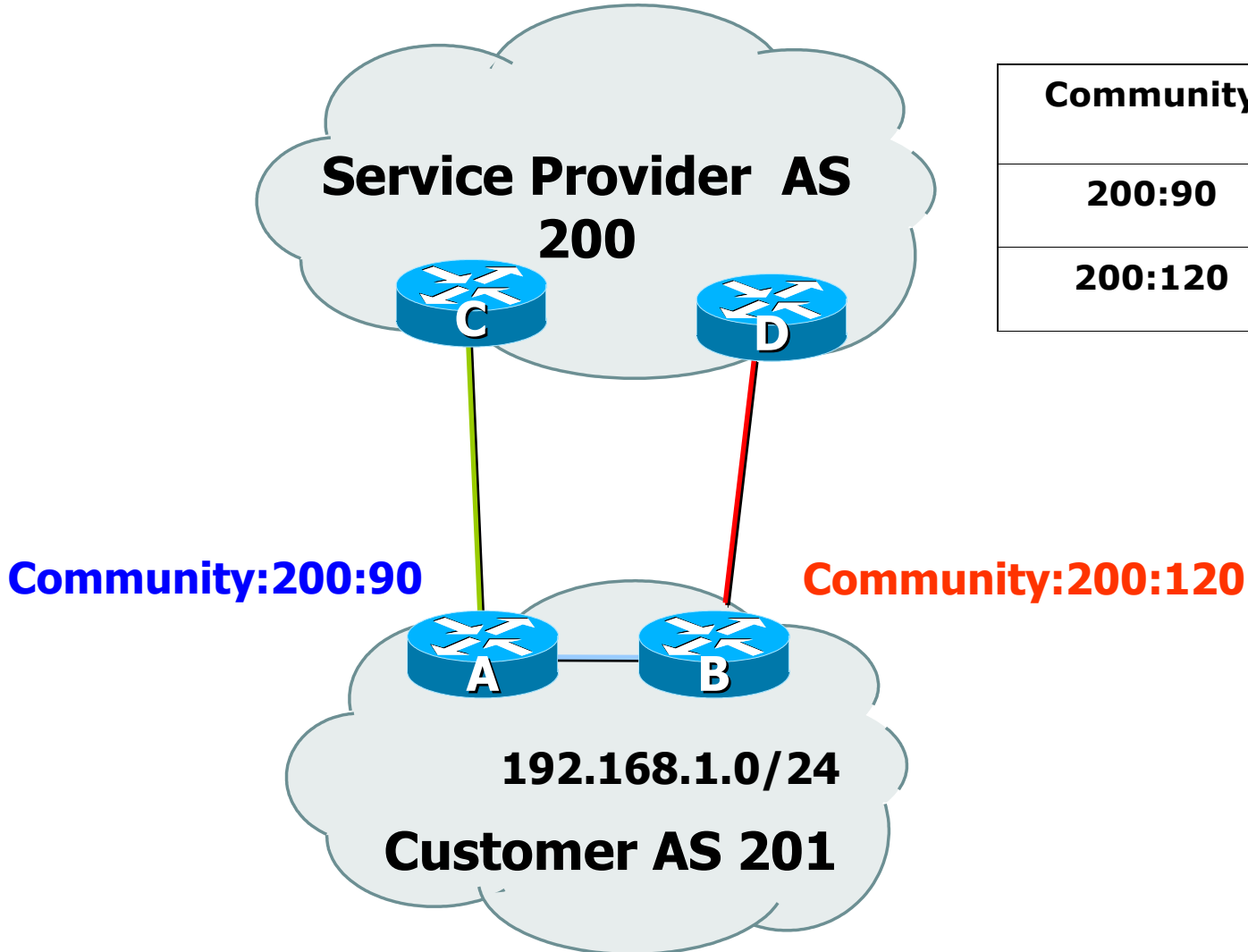
Weight

- ❑ Not really an attribute
- ❑ Used when there is more than one route to same destination
- ❑ Local to the router on which it is assigned, and not propagated in routing updates
- ❑ Default is 32768 for paths that the router originates and zero for other paths
- ❑ Routes with a higher weight are preferred when there are multiple routes to the same destination

Communities

- Transitive, Non-mandatory
- Represented as a numeric value
 - 0x0 – 0xffffffff
 - Internet convention is ASn: <0-65535>
- Used to group destinations
- Each destination could be member of multiple communities
- Flexibility to scope a set of prefixes within or across AS for applying policy

Communities



Community	Local Preference
200:90	90
200:120	120

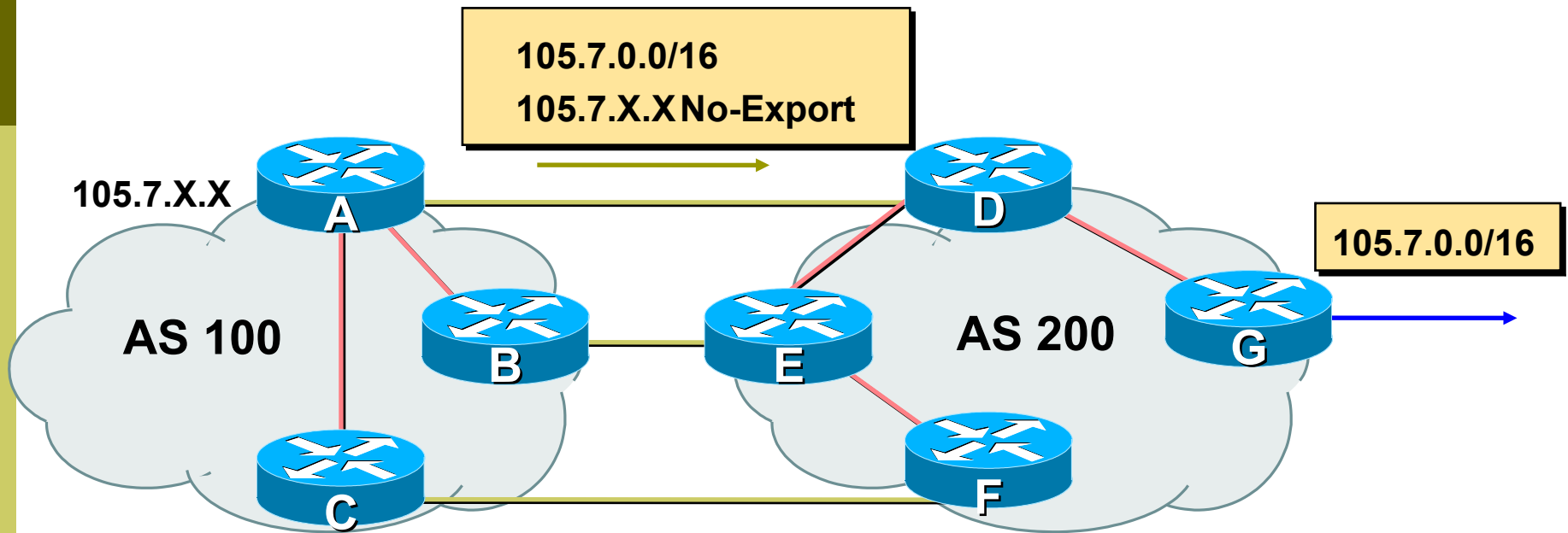
Well-Known Communities

- Several well known communities

www.iana.org/assignments/bgp-well-known-communities

- no-export 65535:65281
 - do not advertise to any eBGP peers
- no-advertise 65535:65282
 - do not advertise to any BGP peer
- no-export-subconfed 65535:65283
 - do not advertise outside local AS (only used with confederations)
- no-peer 65535:65284
 - do not advertise to bi-lateral peers (RFC3765)

No-Export Community

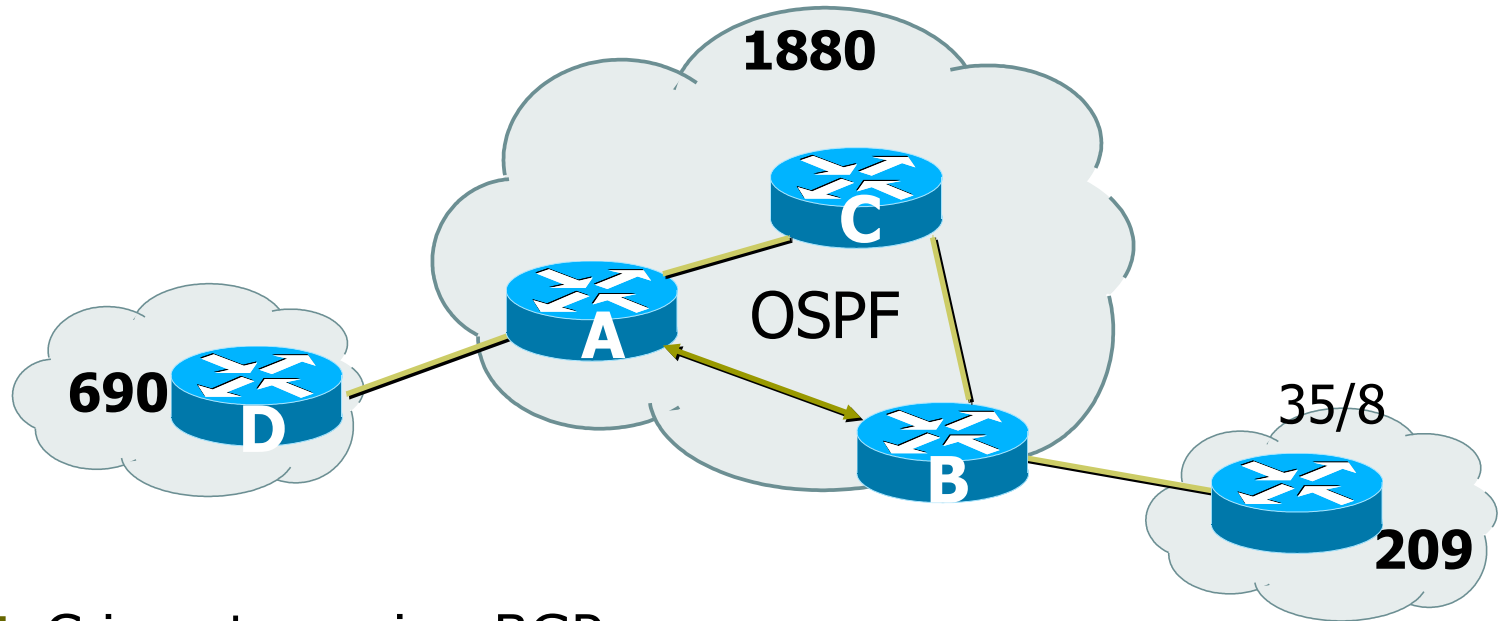


- ❑ AS100 announces aggregate and subprefixes
 - Intention is to improve loadsharing by leaking subprefixes
- ❑ Subprefixes marked with no-export community
- ❑ Router G in AS200 does not announce prefixes with no-export community set

Administrative Distance

- Routes can be learned via more than one protocol
 - Used to discriminate between them
- Route with lowest distance installed in forwarding table
- BGP defaults
 - Local routes originated on router: 200
 - iBGP routes: 200
 - eBGP routes: 20
- Does not influence the BGP path selection algorithm but influences whether BGP learned routes enter the forwarding table

Synchronization



- ❑ C is not running BGP
- ❑ A won't advertised 35/8 to D until the IGP is in sync
- ❑ Turn synchronization off!

```
router bgp 1880
```

```
no synchronization
```


Synchronization

- ❑ In Cisco IOS, BGP does not advertise a route before all routers in the AS have learned it via an IGP
 - Default in IOS prior to 12.4; very unhelpful to most ISPs
- ❑ Disable synchronization if:
 - AS doesn't pass traffic from one AS to another, or
 - All transit routers in AS run BGP, or
 - iBGP is used across backbone
- ❑ You should always use iBGP
 - so, always use "no synchronization"

BGP route selection (bestpath)

- Route has to be synchronized
 - Only if synchronization is enabled
 - Prefix must be in forwarding table
- Next-hop has to be accessible
 - Next-hop must be in forwarding table
- Largest weight
- Largest local preference

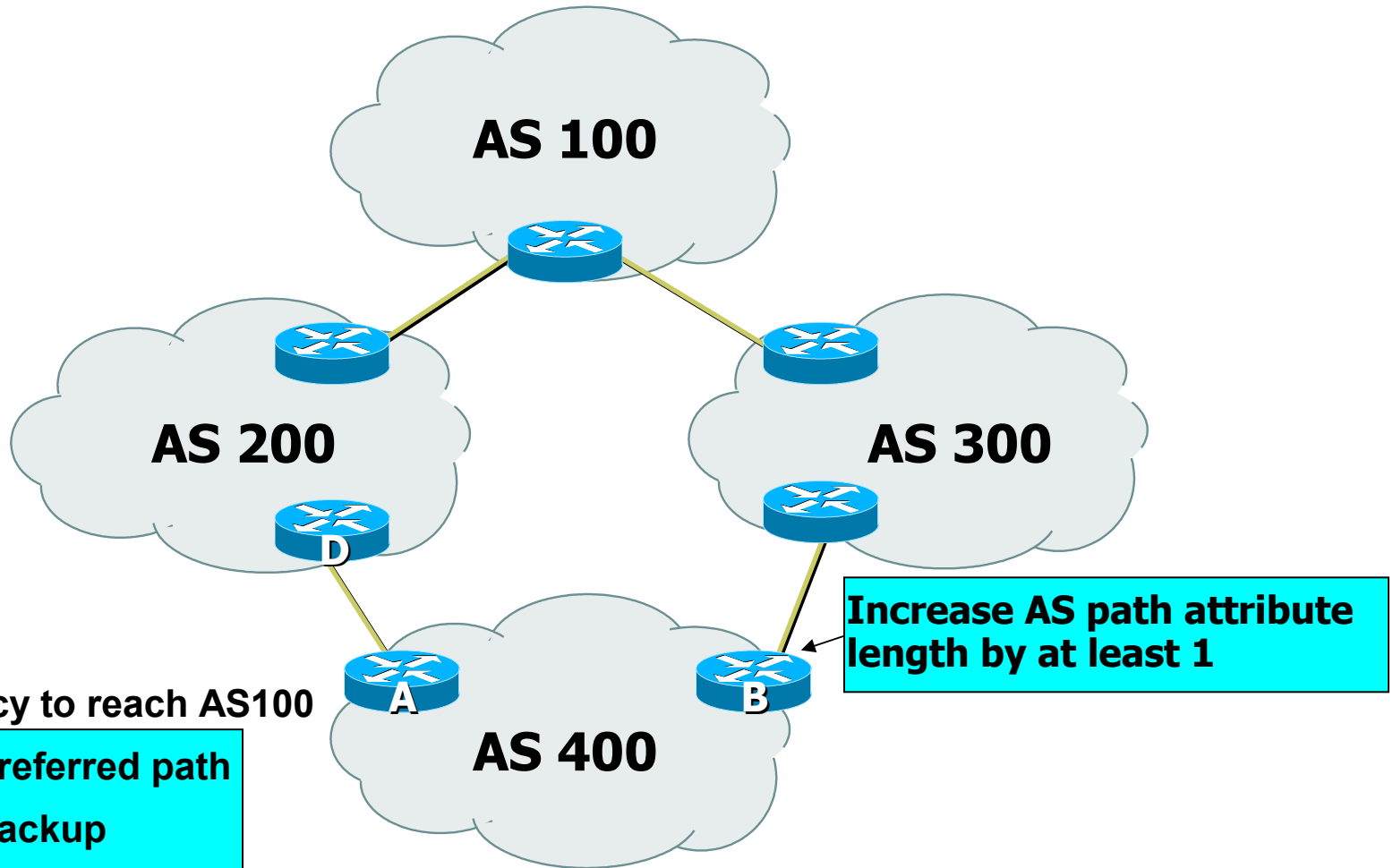
BGP route selection (bestpath)

- Locally sourced
 - Via redistribute or network statement
- Shortest AS path length
 - Number of ASes in the AS-PATH attribute
- Lowest origin
 - IGP < EGP < incomplete
- Lowest MED
 - Compared from paths from the same AS

BGP route selection (bestpath)

- External before internal
 - Choose external path before internal
- Closest next-hop
 - Lower IGP metric, nearest exit to router
- Lowest router ID
- Lowest IP address of neighbour

BGP Route Selection...



BGP Exercise 5

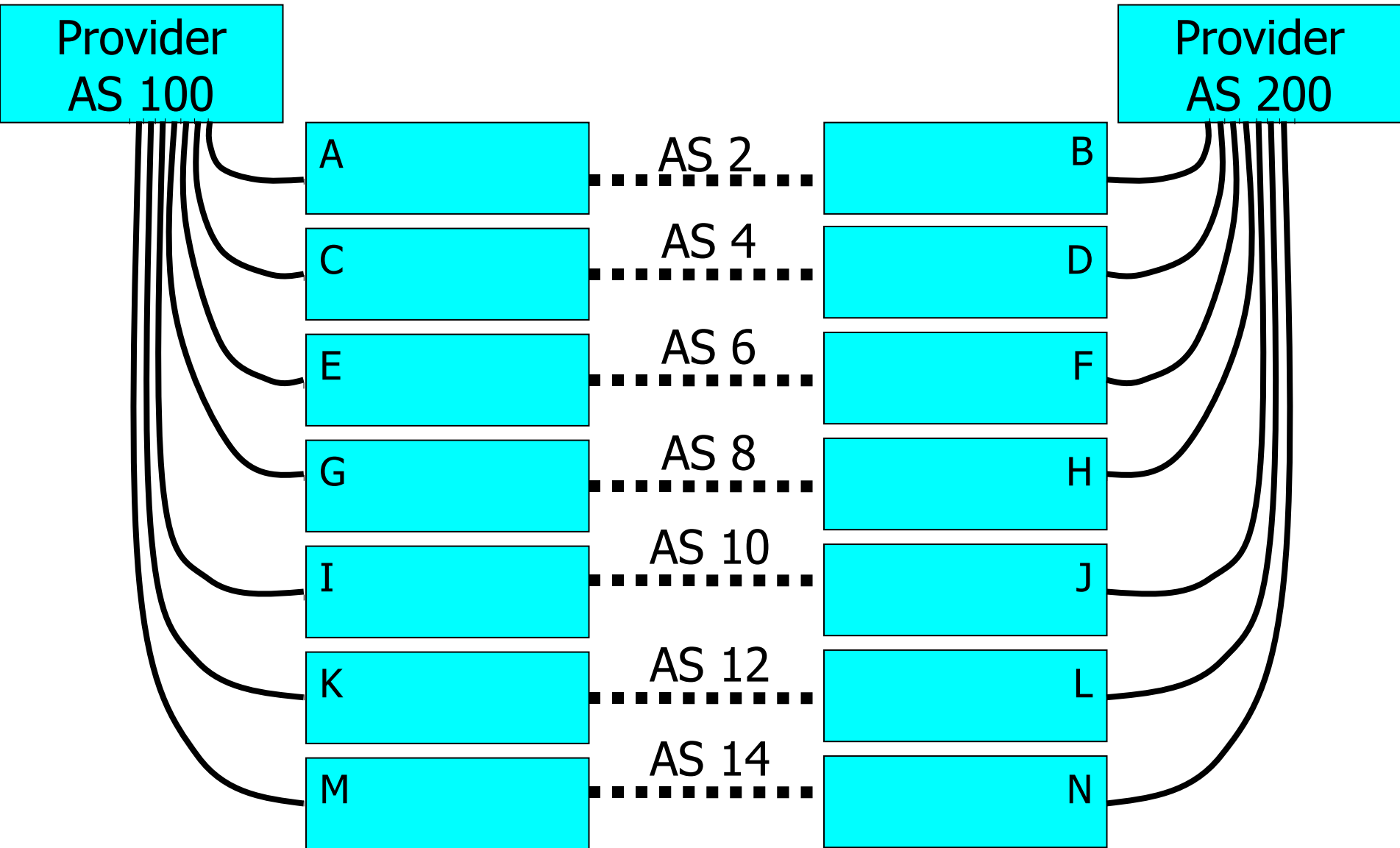


Internal BGP (iBGP)

Exercise 5: Configure iBGP

- ❑ Tables join into pairs, with two routers per AS
- ❑ Each AS has two upstream providers
- ❑ IS-IS and iBGP within your AS
- ❑ eBGP to your upstream provider
- ❑ Filter everything!

Exercise 5: Configure iBGP



Exercise 5: Configure iBGP

- The two routers in your AS should talk iBGP to each other
 - no filtering here
 - use “update-source loopback 0”
- One of your routers talks eBGP to AS 100, and one talks to AS 200.
 - Filter!
 - Send only your routes (use a prefix-list)
 - Accept all except bogus routes (“sanity-filter”)

Exercise 5: What you should see

- ❑ Directly from AS 100: routes to entire classroom
- ❑ Directly from AS 200: routes to entire classroom
- ❑ From your iBGP neighbour: indirect routes through AS 100 or AS 200 to entire classroom
- ❑ Which route do you prefer?

BGP Part 10

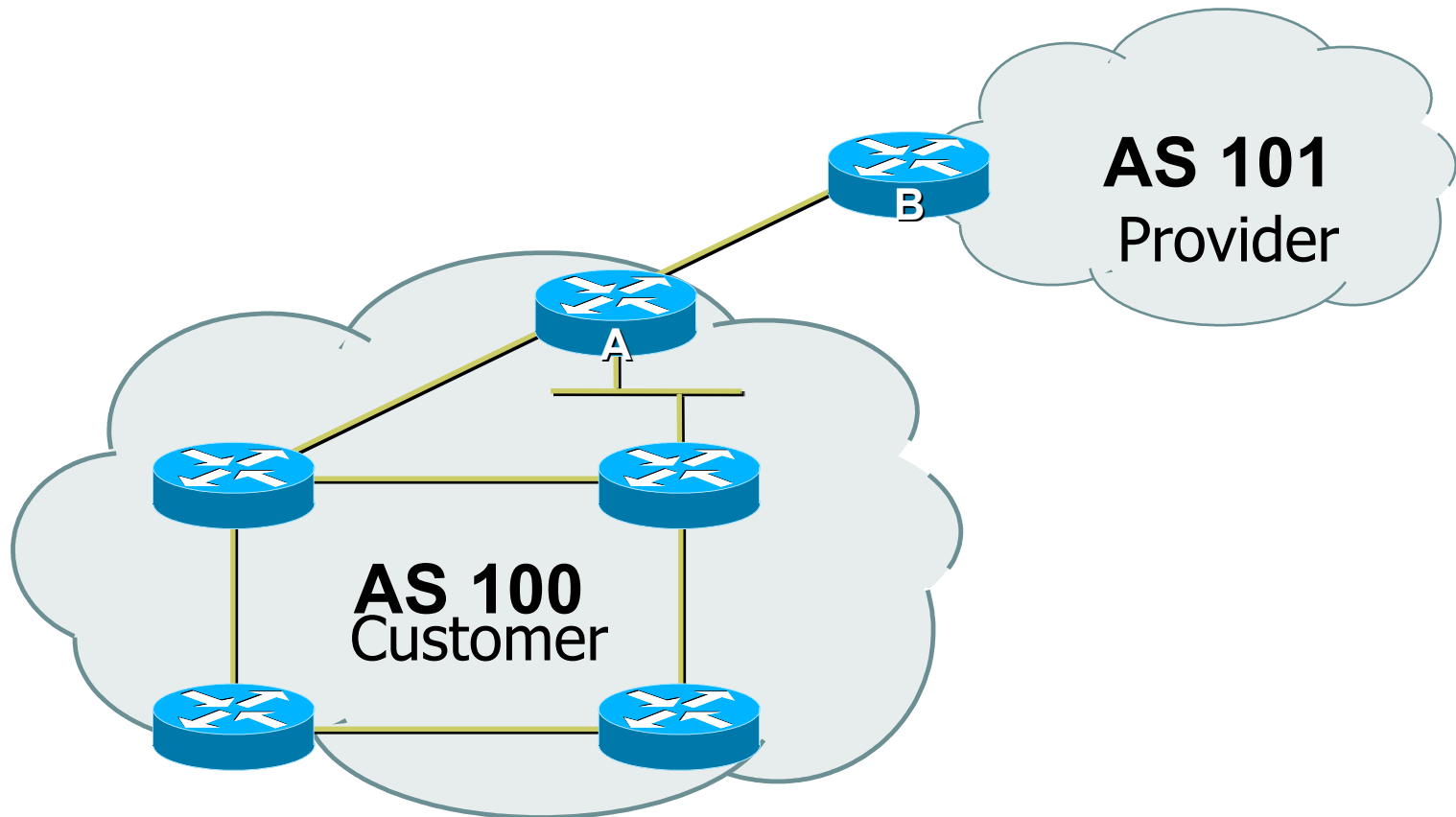


BGP and Network Design

Stub AS

- ❑ Enterprise network, or small ISP
- ❑ Typically no need for BGP
- ❑ Point default towards the ISP
- ❑ ISP advertises the stub network to Internet
- ❑ Policy confined within ISP policy

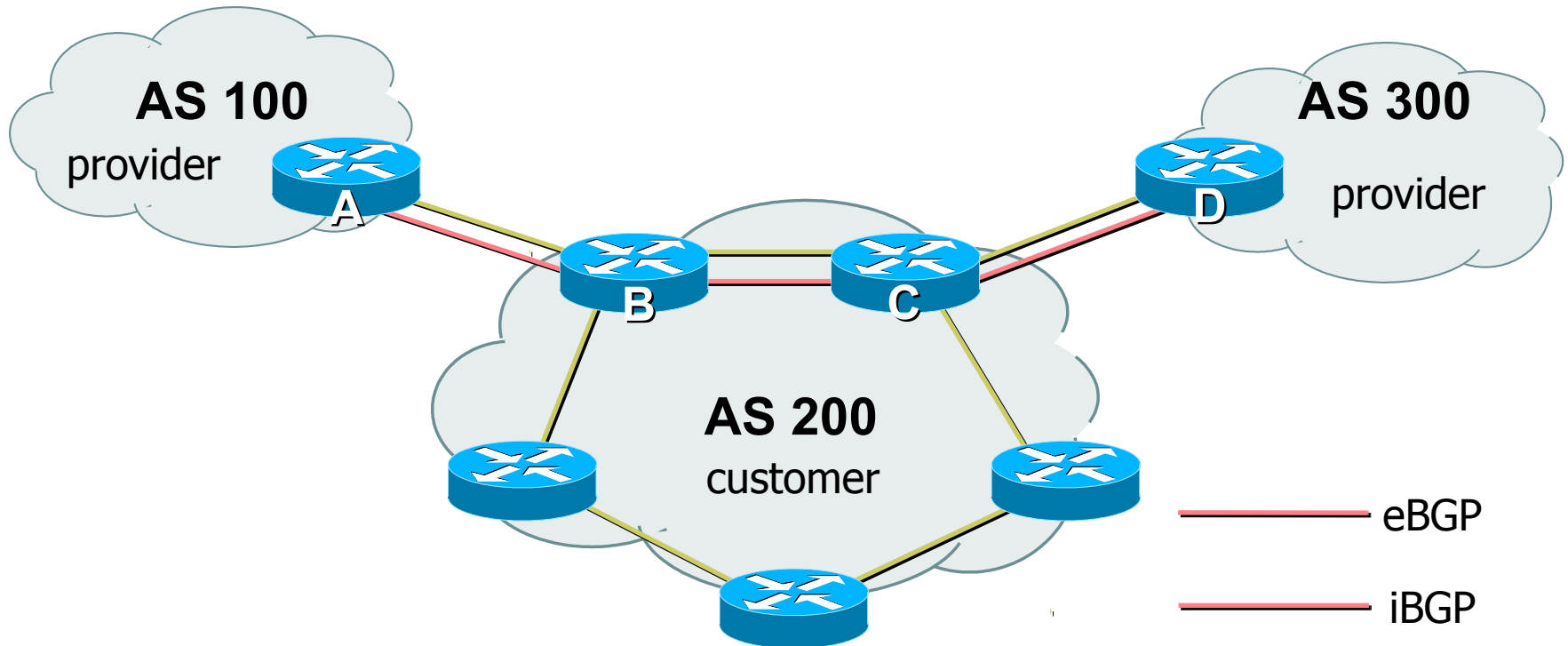
Stub AS



Multihomed AS

- ❑ Enterprise network or small ISP
- ❑ Only border routers speak BGP
- ❑ iBGP only between border routers
- ❑ Rest of network either has:
 - exterior routes redistributed in a controlled fashion into IGP...
 - ...or use defaults (much preferred!)

Multi-homed AS

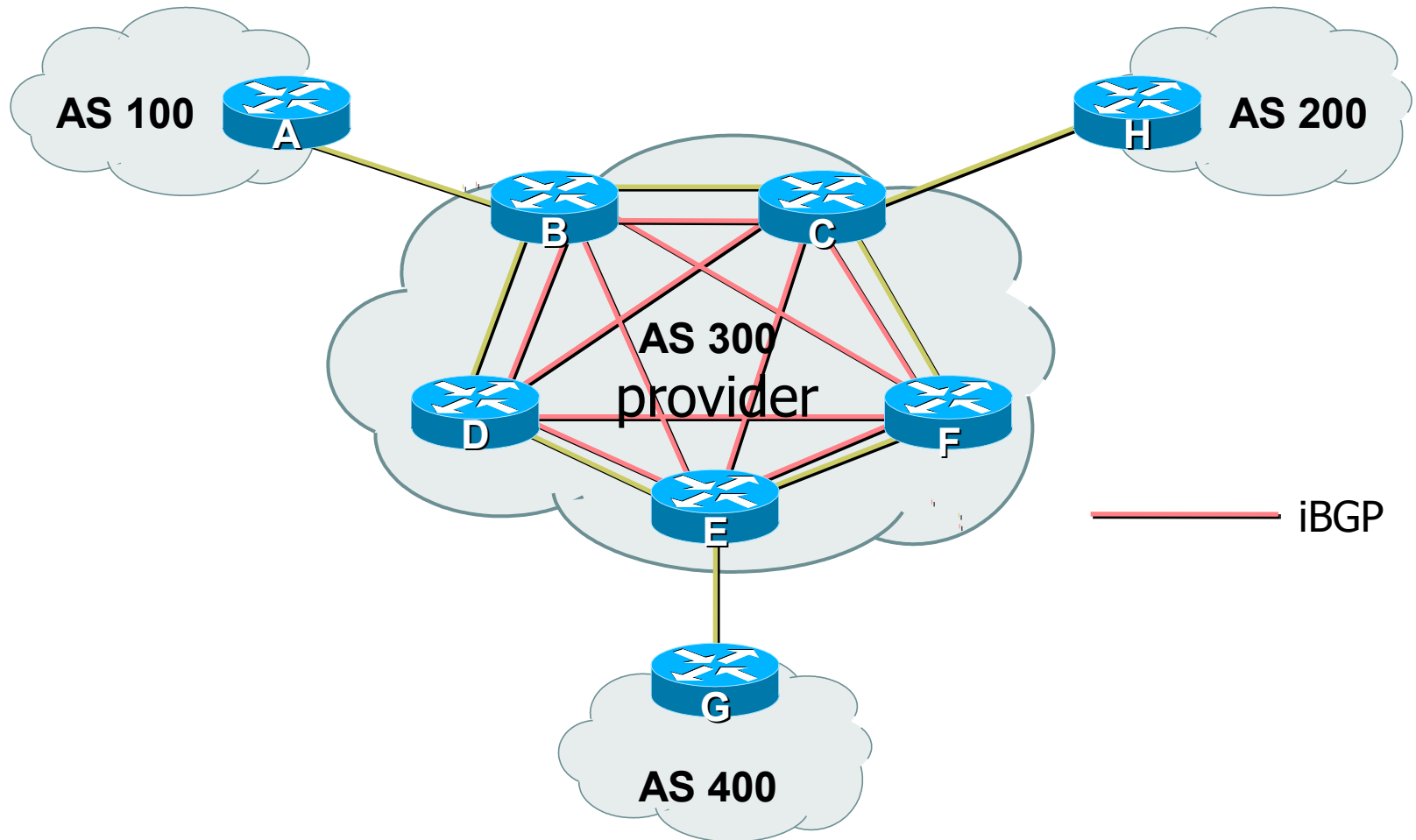


- More details on multihoming coming up...

Service Provider Network

- iBGP used to carrier exterior routes
 - No redistribution into IGP
- IGP used to track topology inside your network
- Full iBGP mesh required
 - Every router in ISP backbone should talk iBGP to every other router
 - This has scaling problems, and solutions (e.g. route reflectors)

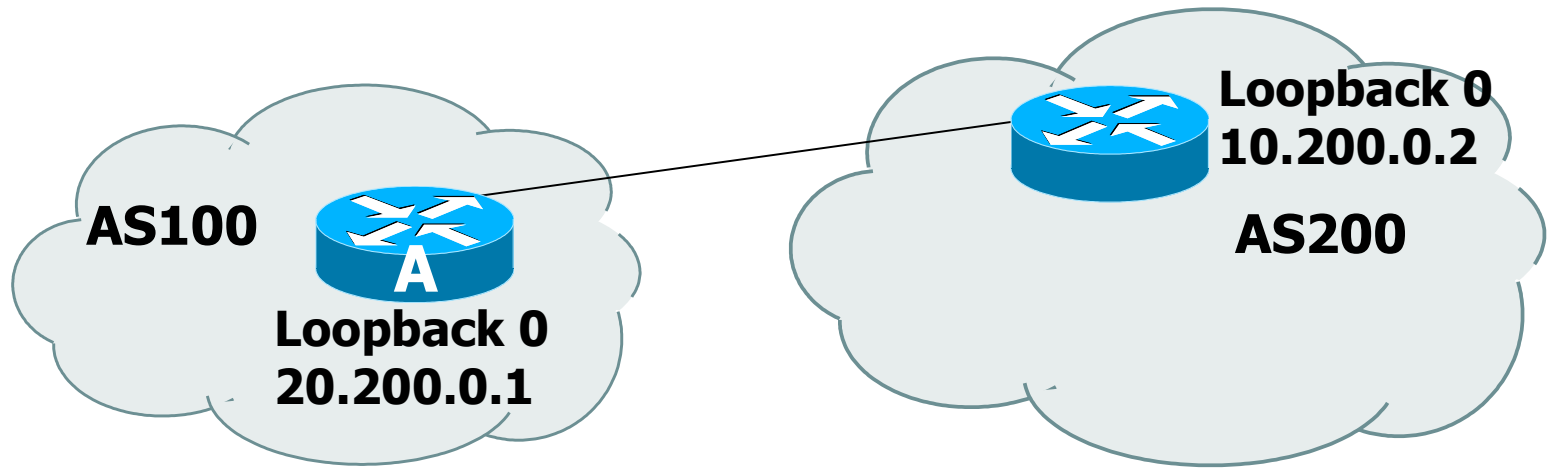
Common Service Provider Network



Load-sharing – single path

❑ Router A:

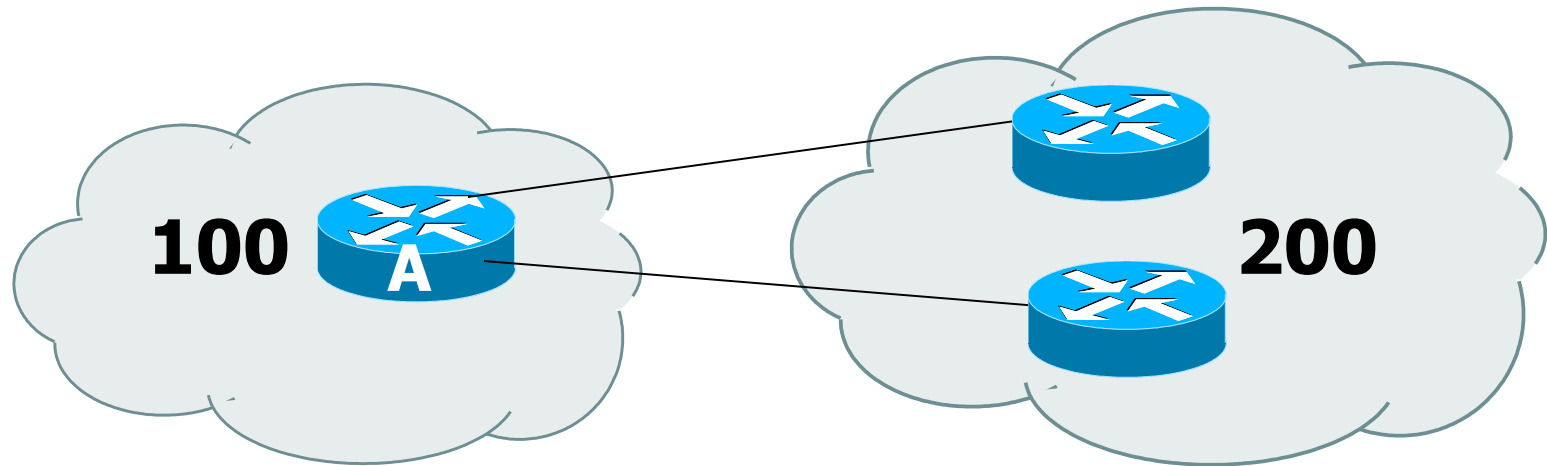
```
interface loopback 0
 ip address 20.200.0.1 255.255.255.255
!
router bgp 100
 neighbor 10.200.0.2 remote-as 200
 neighbor 10.200.0.2 update-source loopback0
 neighbor 10.200.0.2 ebgp-multihop 2
!
ip route 10.200.0.2 255.255.255.255 <DMZ-link1>
ip route 10.200.0.2 255.255.255.255 <DMZ-link2>
```



Load-sharing – multiple paths from the same AS

□ Router A:

```
router bgp 100
  neighbor 10.200.0.1 remote-as 200
  neighbor 10.300.0.1 remote-as 200
  maximum-paths 2
```



Note: A still only advertises one “best” path to ibgp peers

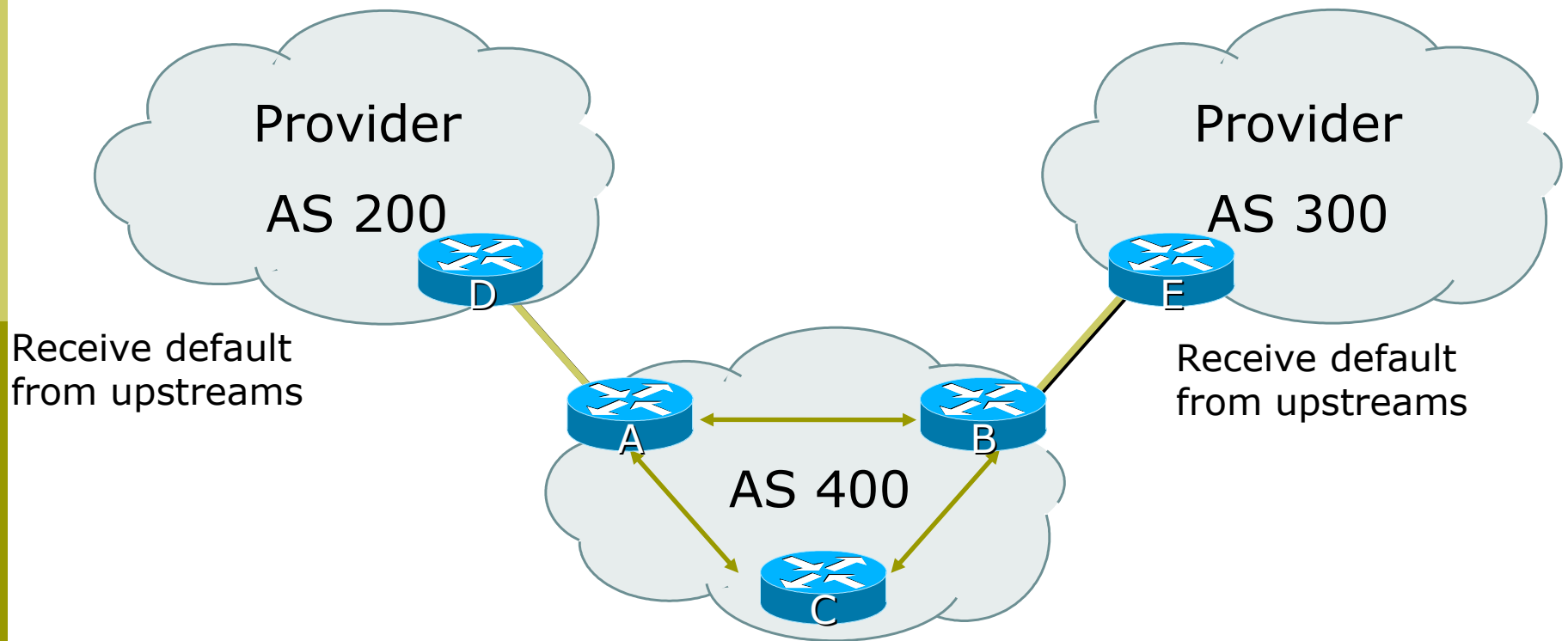
Redundancy – Multi-homing

- Reliable connection to Internet
- 3 common cases of multi-homing
 - default from all providers
 - customer + default from all providers
 - full routes from all providers
- Address Space
 - comes from upstream providers, or
 - allocated directly from registries

Default from all providers

- Low memory/CPU solution
- Provider sends BGP default
 - provider is selected based on IGP metric
- Inbound traffic decided by providers' policy
 - Can influence using outbound policy, example: AS-path prepend

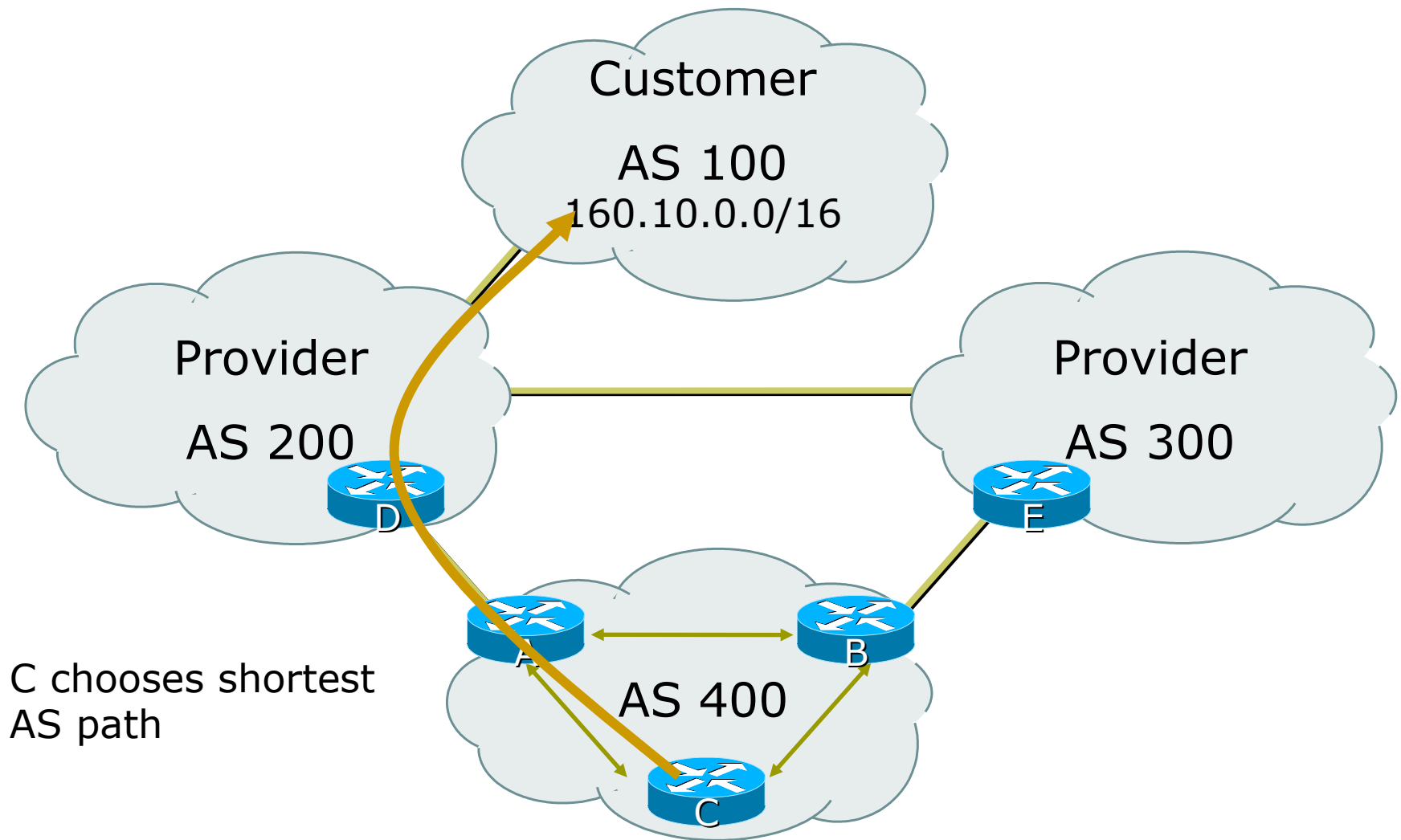
Default from all providers



Customer prefixes plus default from all providers

- Medium memory and CPU solution
- Granular routing for customer routes, default for the rest
 - Route directly to customers as those have specific policies
- Inbound traffic decided by providers' policies
 - Can influence using outbound policy

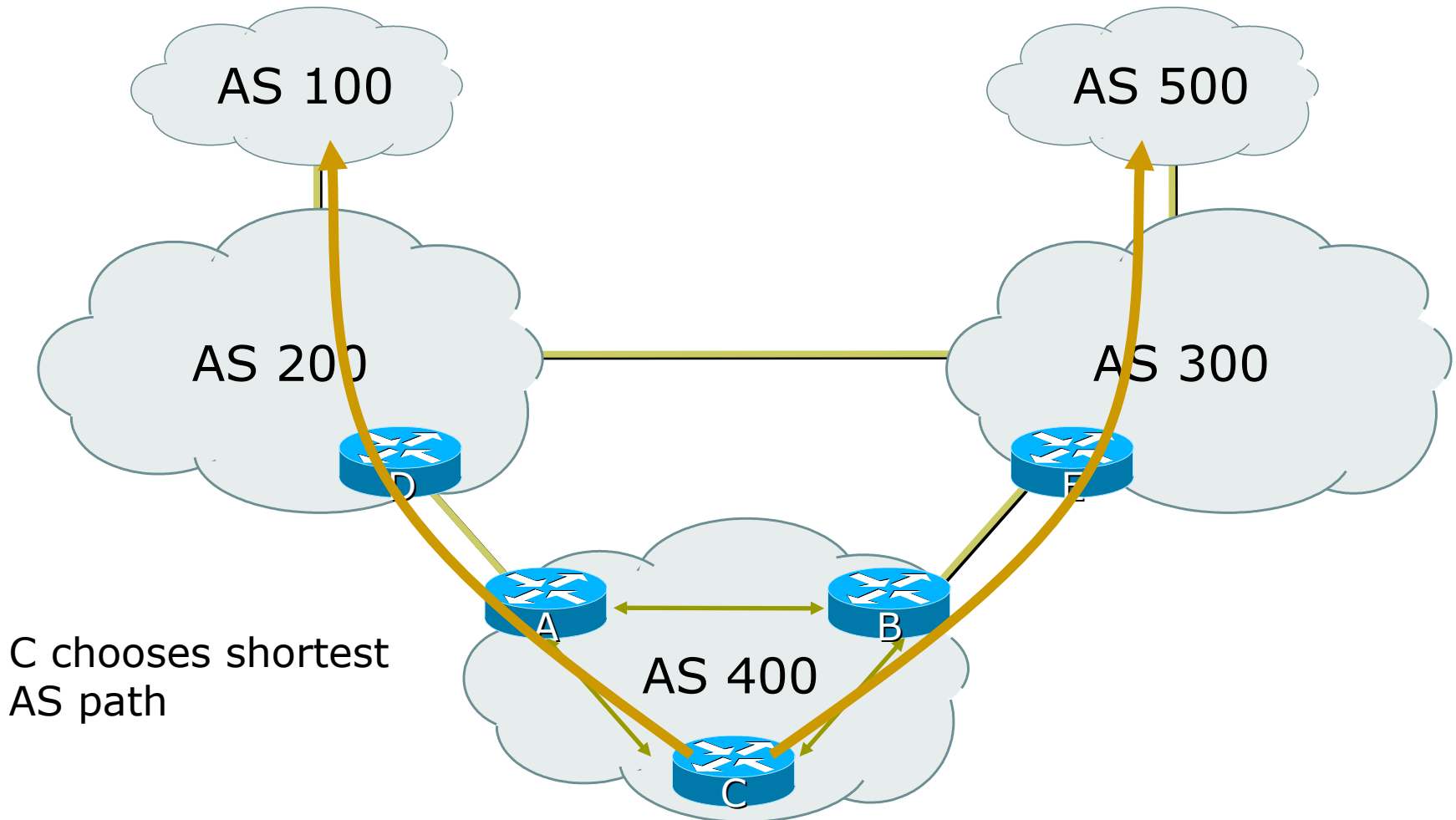
Customer routes from all providers



Full routes from all providers

- ❑ More memory/CPU
- ❑ Fine grained routing control
- ❑ Usually transit ASes take full routes
- ❑ Usually pervasive BGP

Full routes from all providers



Best Practices

IGP in Backbone

- IGP connects your backbone together, not your clients' routes
 - Clients' routes go into iBGP
 - Hosting and service LANs go into iBGP
 - Dial/Broadband/Wireless pools go into iBGP
- IGP must converge quickly
 - The **fewer** prefixes in the IGP the **better**
- IGP should carry netmask information – OSPF, IS-IS, EIGRP

Best Practices

iBGP in Backbone

- iBGP runs between all routers in backbone
- Configuration essentials:
 - Runs between loopbacks
 - Next-hop-self
 - Send-community
 - Passwords
 - All non-infrastructure prefixes go here

Best Practices...

Connecting to a customer

- Static routes
 - You control directly
 - No route flaps
- Shared routing protocol or leaking
 - Strongly discouraged
 - You must filter your customers info
 - Route flaps
- BGP for multi-homed customers
 - Private AS for those who multihome on to your backbone
 - Public AS for the rest

Best Practices...

Connecting to other ISPs

- Advertise only what you serve
- Take back as little as you can
- Take the shortest exit
- **Aggregate your routes!!**
 - Consult RIPE-399 document for recommendations:
 - <http://www.ripe.net/docs/ripe-399.html>
- **FILTER! FILTER! FILTER!**

Best Practices...

The Internet Exchange

- Long distance connectivity is:
 - Expensive
 - Slow (speed of light limitations)
 - Congested
- Connect to several providers at a single point
 - Cheap
 - Fast
- More details later!

Summary

- We have learned about:
 - BGP Protocol Basics
 - Routing Policy and Filtering
 - BGP Best Path Computation
 - Typical BGP topologies
 - Routing Policy
 - BGP Network Design
 - Redundancy/Load sharing
 - Some best practices