



Border Gateway Protocol – BGP4

Philip Smith

E2 Workshop, AfNOG2006

Border Gateway Protocol (BGP4)



- Part 0: Why use BGP?
- Part 1: Forwarding and Routing (review)
- Part 2: Interior and Exterior Routing
- Part 3: BGP Building Blocks
- Part 4: Configuring BGP
- Case Study 1, Exercise 1: Single upstream
- Part 5: BGP Protocol Basics
- Part 6: BGP Protocol - more detail
- Case Study 2, Exercise 2: Local peer
- Part 7: Routing Policy and Filtering
- Exercise 3: Filtering on AS-path
- Exercise 4: Filtering on prefix-list
- Part 8: More detail than you want
- Exercise 5: Interior BGP
- Part 9: BGP and Network Design



BGP Part 0

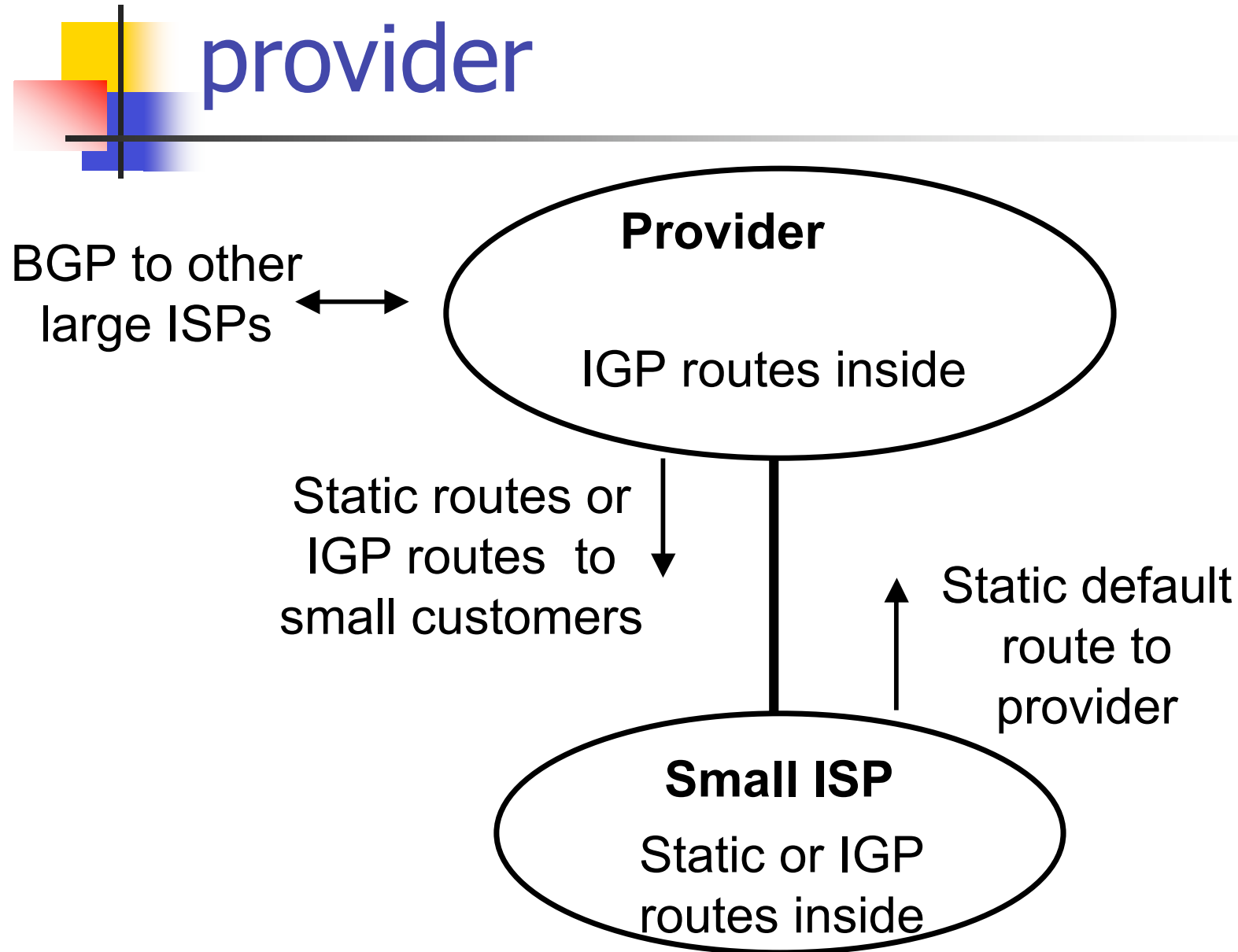
Why use BGP?



Consider a typical small ISP

- Local network in one country
- May have multiple POPs in different cities
- Line to Internet
 - International line providing transit connectivity
 - Very, very expensive international line
- Doesn't yet need BGP

Small ISP with one upstream provider



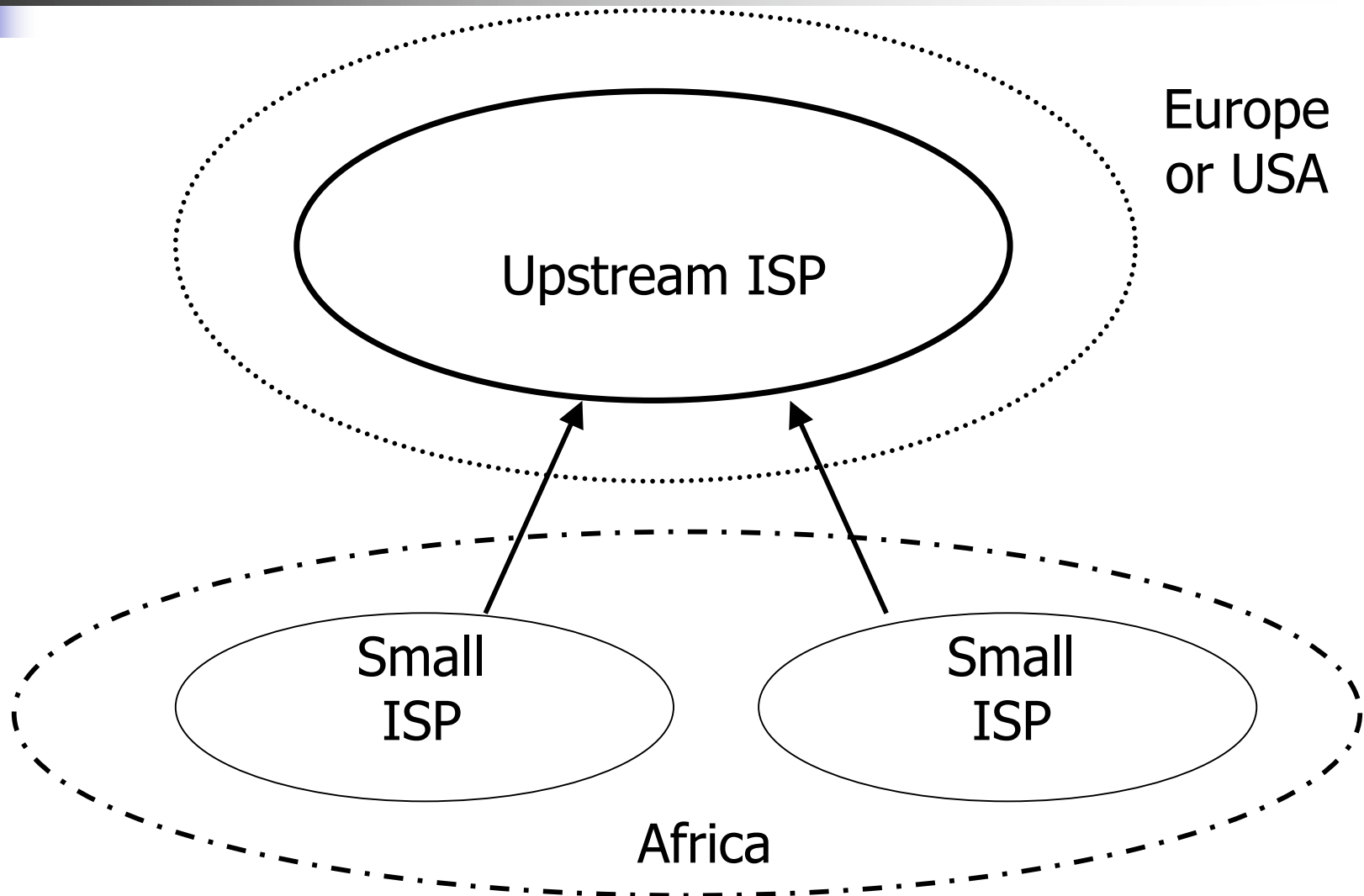


What happens with other ISPs in the same country

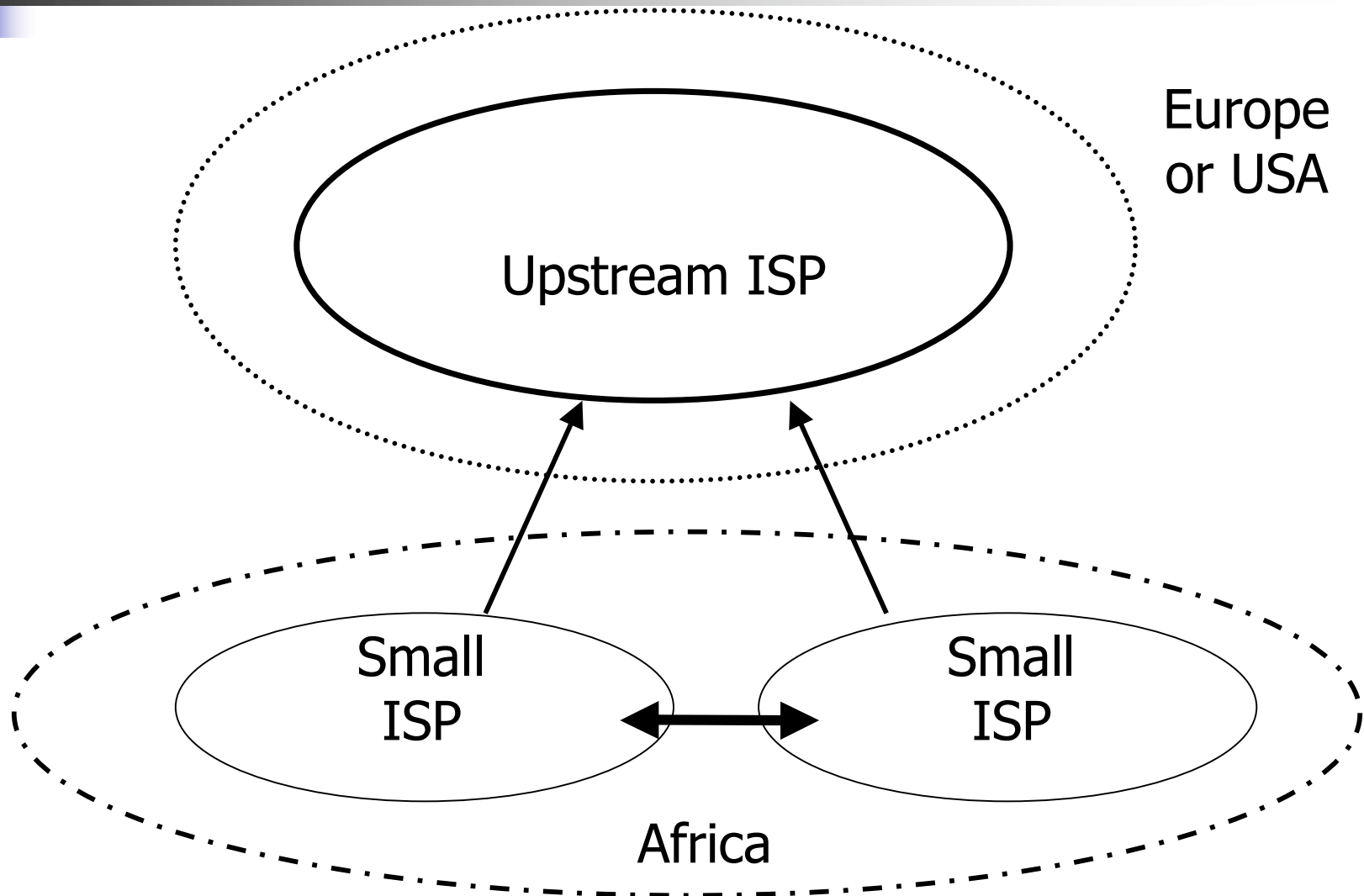
- Similar setup
- Traffic between you and them goes over
 - Your expensive line
 - Their expensive line
- Traffic can be significant
 - Your customers want to talk to their customers
 - Same language/culture
 - Local email, discussion lists, web sites




Keeping Local Traffic Local



Keeping Local Traffic Local

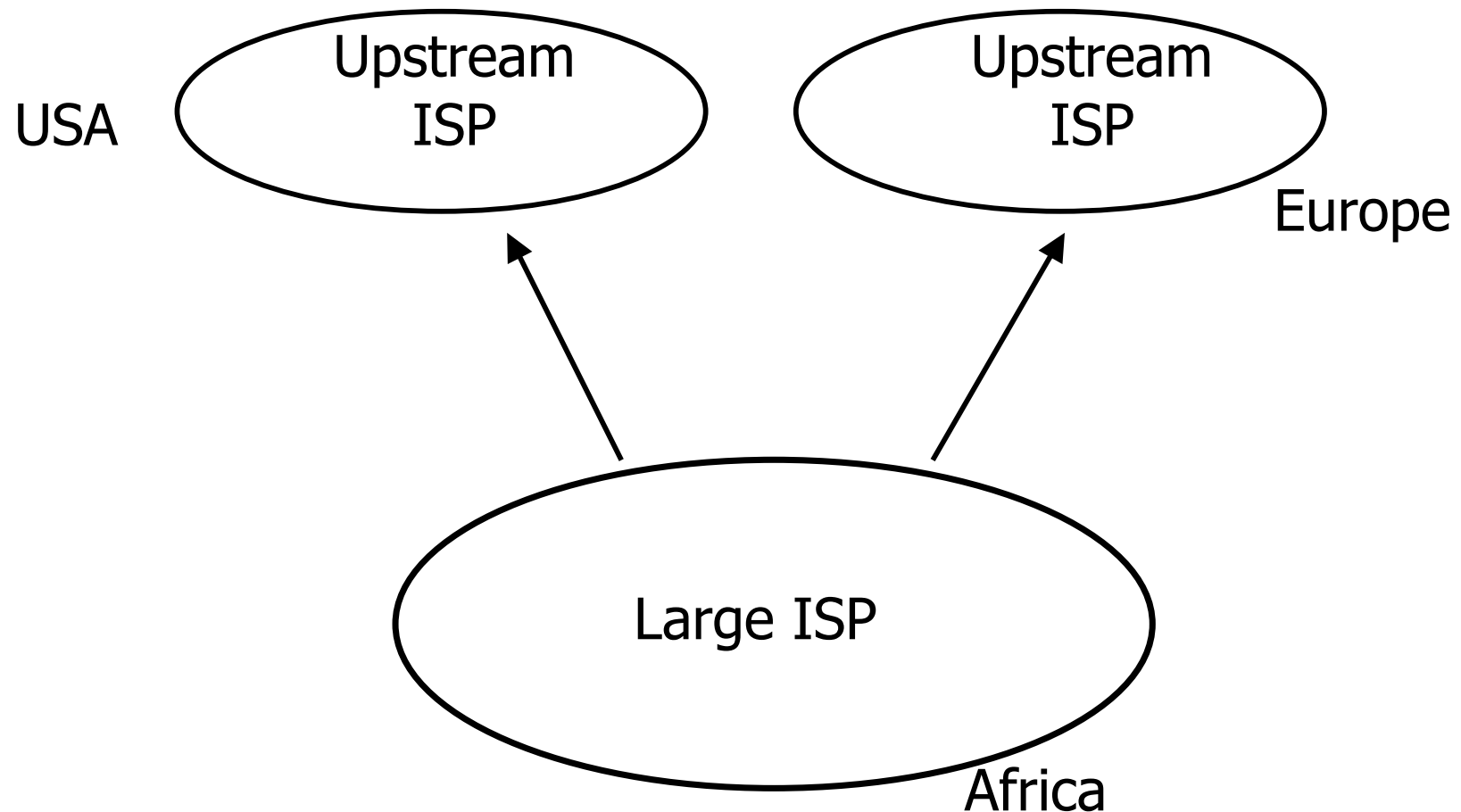




Consider a larger ISP with multiple upstreams

- Large ISP multi-homes to two or more upstream providers
 - multiple connections
 - to achieve:
 - redundancy
 - connection diversity
 - increased speeds
 - Use BGP to choose a different upstream for different destination addresses

A Large ISP with more than one upstream provider





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 neighbours
 - Prefer cheaper connections
 - Prefer connections with better latency
 - Load-sharing, etc



“Policy” (continued)

- Implementing policy:
 - Accepting routes from some ISPs and not others
 - Sending some routes to some ISPs and not to others
 - Preferring routes from some ISPs over those from other ISPs



“Policy” Implementation

- You want to use a local line to talk to the customers of other local ISPs
 - local peering
- You do not want other local ISPs to use your expensive international lines
 - no free transit!
- So you need some sort of control over routing policies
- BGP can do this



Terminology: “Peering” and “Transit”

- **Peering:** getting connectivity to the network of other the ISP
 - ... and just that network, no other networks
 - Frequently at zero cost (zero-settlement)
- **Transit:** getting connectivity though the network of the other ISP to other networks
 - ... getting connectivity to rest of world (or part thereof)
 - Usually at cost (customer-provider relationship)



Terminology: “Aggregation”

- Combining of several smaller blocks of address space into a larger block
- For example:
 - 192.168.4.0/24 and 192.168.5.0/24 are contiguous address blocks
 - They can be combined and represented as 192.168.4.0/23...
 - ...with no loss of information!



“Aggregation” (continued)

- Useful because it hides detailed information about the local network:
 - The outside world needs to know about the range of addresses in use
 - The outside world does not need to know about the small pieces of address space used by different customers inside your network



“Aggregation” (continued)

- A jigsaw puzzle makes up a picture which is easier to see when the puzzle is complete!
- Aggregation is very necessary when using BGP to “talk” to the Internet

Summary:

Why do I need BGP?

- Multi-homing – connecting to multiple providers
 - upstream providers
 - local networks – regional peering to get local traffic
- Policy discrimination
 - controlling how traffic flows
 - do not accidentally provide transit to non-customers

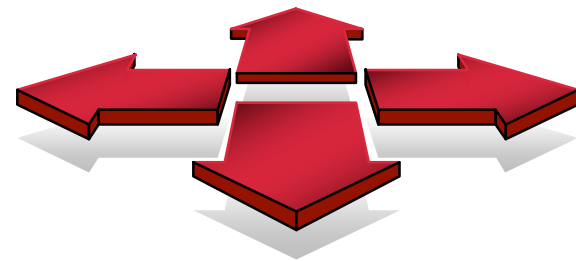


BGP Part 1

Forwarding and Routing

Routing versus Forwarding

- Routing = building maps and giving directions
- Forwarding = moving packets between interfaces according to the “directions”





Routing Table/RIB

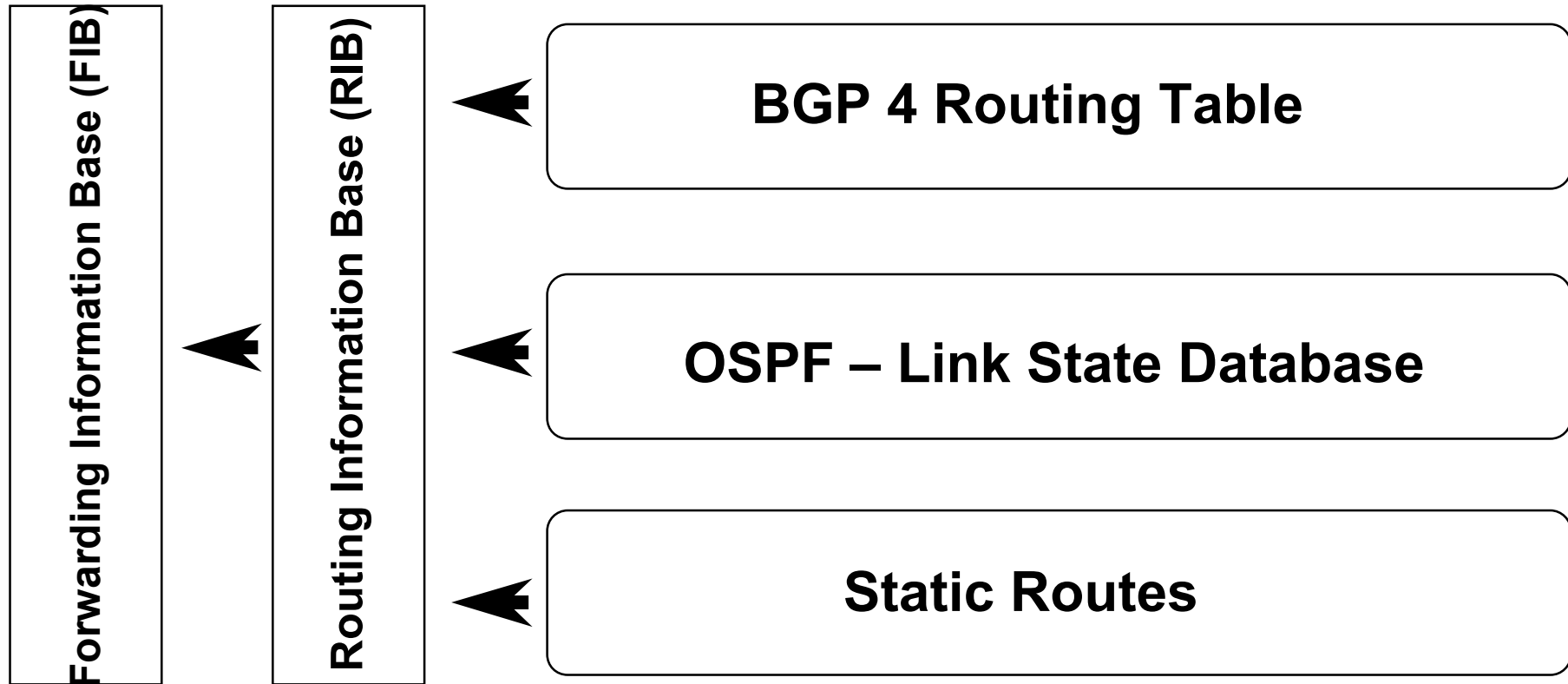
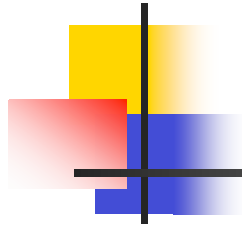
- Routing table is managed by a routing protocol (e.g. OSPF or BGP)
- Often called the RIB – Routing Information Base
- Each routing protocol has its own way of managing its own routing tables
- Each routing protocol has a way of exchanging information between routers using the same protocol



Forwarding Table/FIB

- Forwarding table determines how packets are sent through the router
- Often called the FIB – Forwarding Information Base
- Made from routing table built by routing protocols
 - Best routes from routing tables are installed
- Performs the lookup to find next-hop and outgoing interface
- Switches the packet with new encapsulation as per the outgoing interface

Routing Tables Feed the Forwarding Table





IP Routing

- Each router or host makes its own routing decisions
- Sending machine does not have to determine the entire path to the destination
- Sending machine just determines the next-hop along the path (based on destination IP address)
 - This process is repeated until the destination is reached, or there's an error
- Forwarding table is consulted (at each hop) to determine the next-hop



IP Routing

- Classless routing
 - route entries include
 - destination
 - next-hop
 - mask (prefix-length) indicating size of address space described by the entry
- Longest match
 - for a given destination, find longest prefix match in the routing table
 - example: destination is 35.35.66.42
 - routing table entries are 35.0.0.0/8, 35.35.64.0/19 and 0.0.0.0/0
 - All these routes match, but the /19 is the longest match



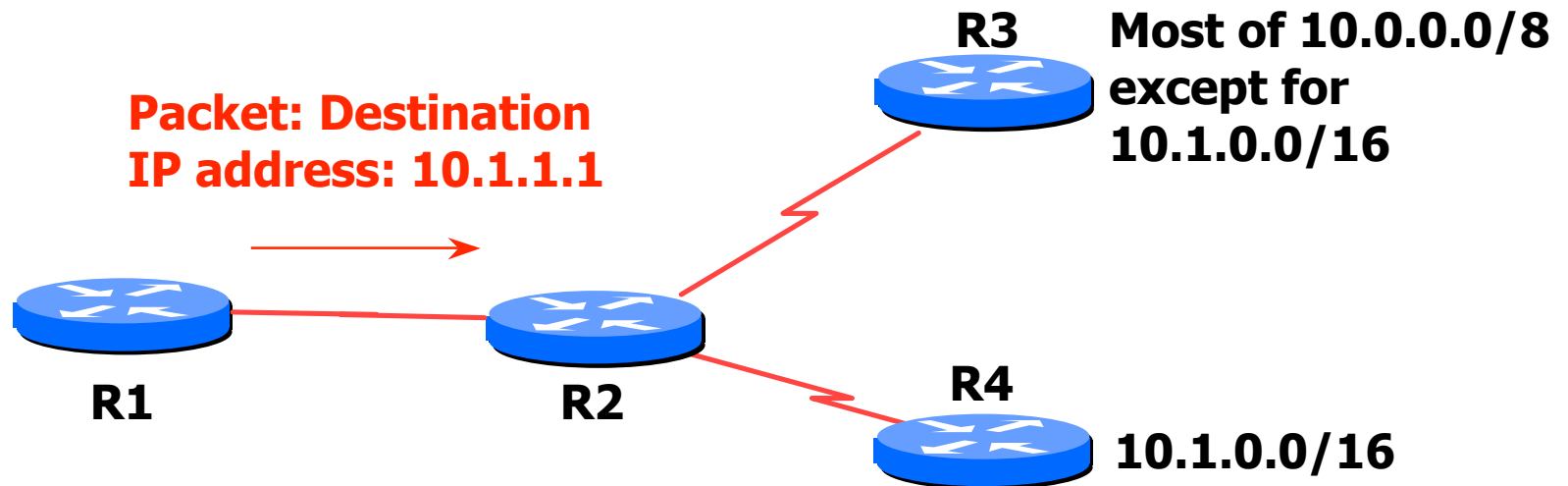
IP routing

- Default route

- where to send packets if there is no entry for the destination in the routing table
- most machines have a single default route
- often referred to as a default gateway

- 0.0.0.0/0
 - matches all possible destinations, but is usually not the longest match

IP route lookup: Longest match routing

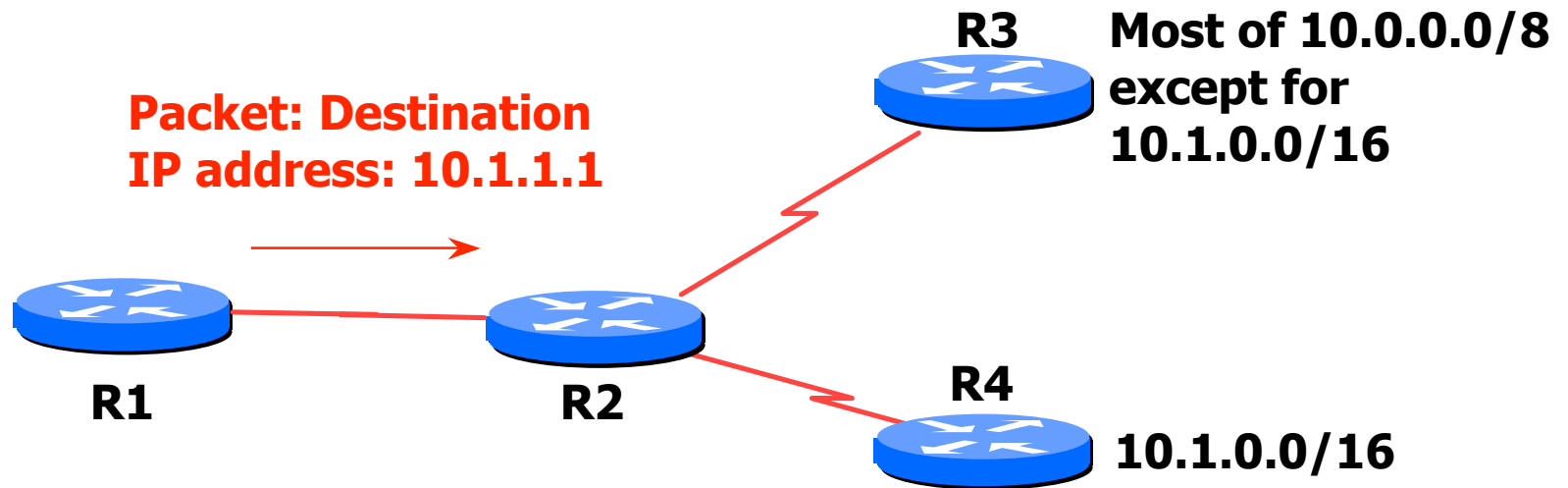


Based on destination IP address

R2's IP forwarding table

10.0.0.0/8	→ R3
10.1.0.0/16	→ R4
20.0.0.0/8	→ R5
0.0.0.0/0	→ R1

IP route lookup: Longest match routing



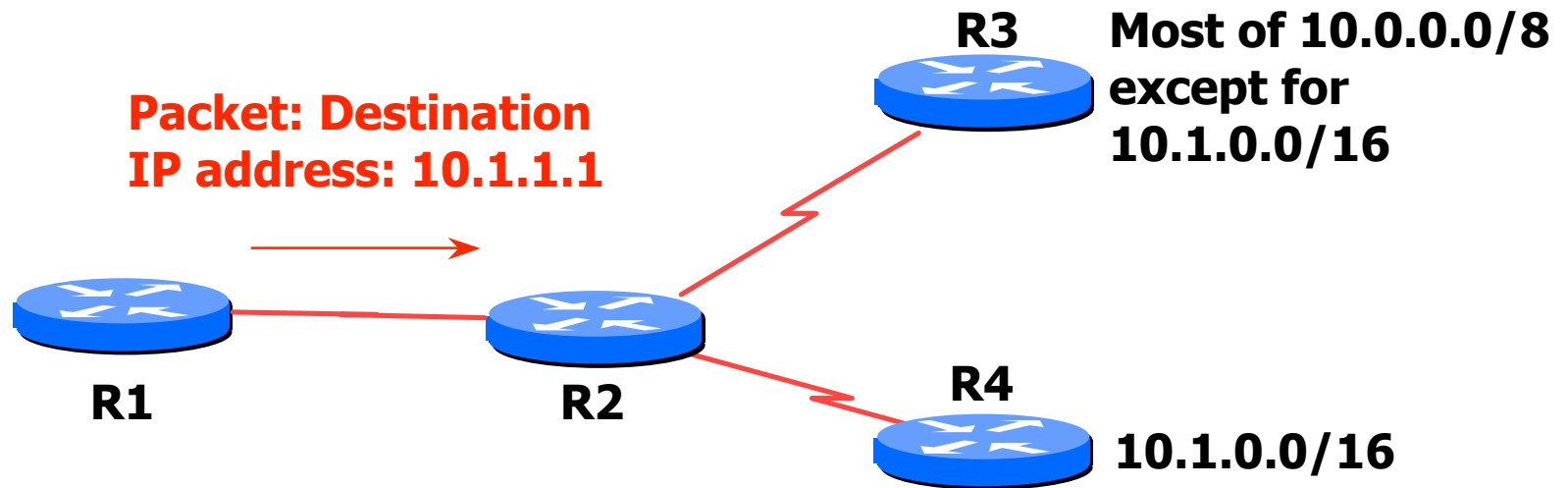
Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

10.1.1.1 & FF.00.00.00
vs.
10.0.0.0 & FF.00.00.00
Match! (length 8)

IP route lookup: Longest match routing



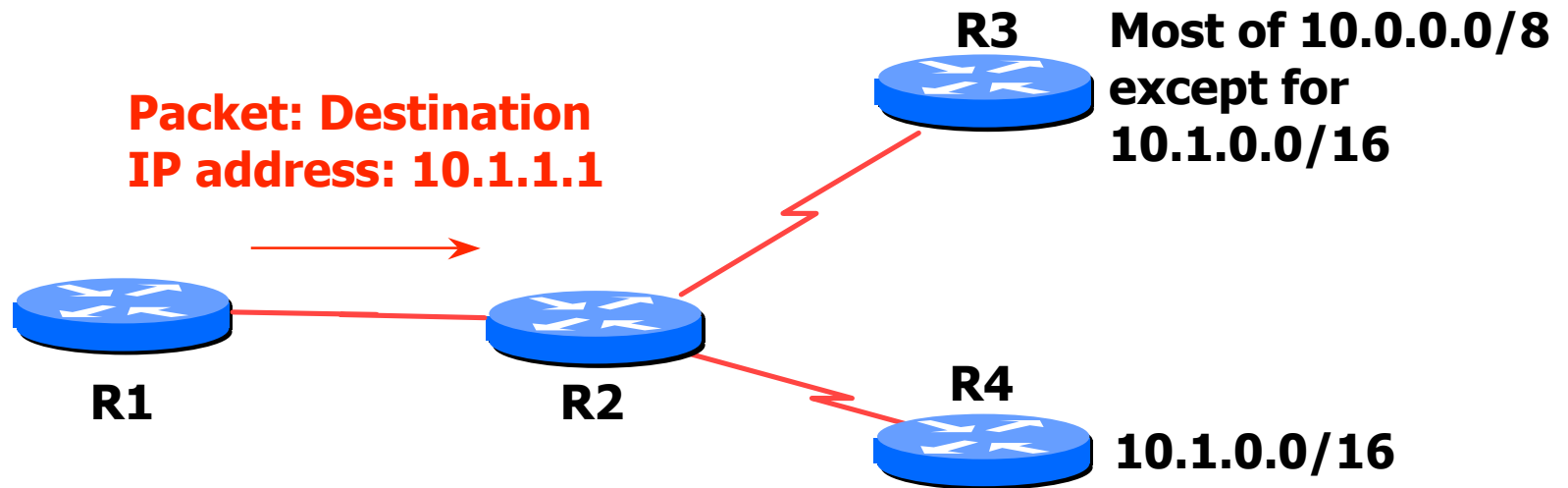
Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

10.1.1.1 & FF.FF.00.00
vs.
10.1.0.0 & FF.FF.00.00
Match! (length 16)

IP route lookup: Longest match routing



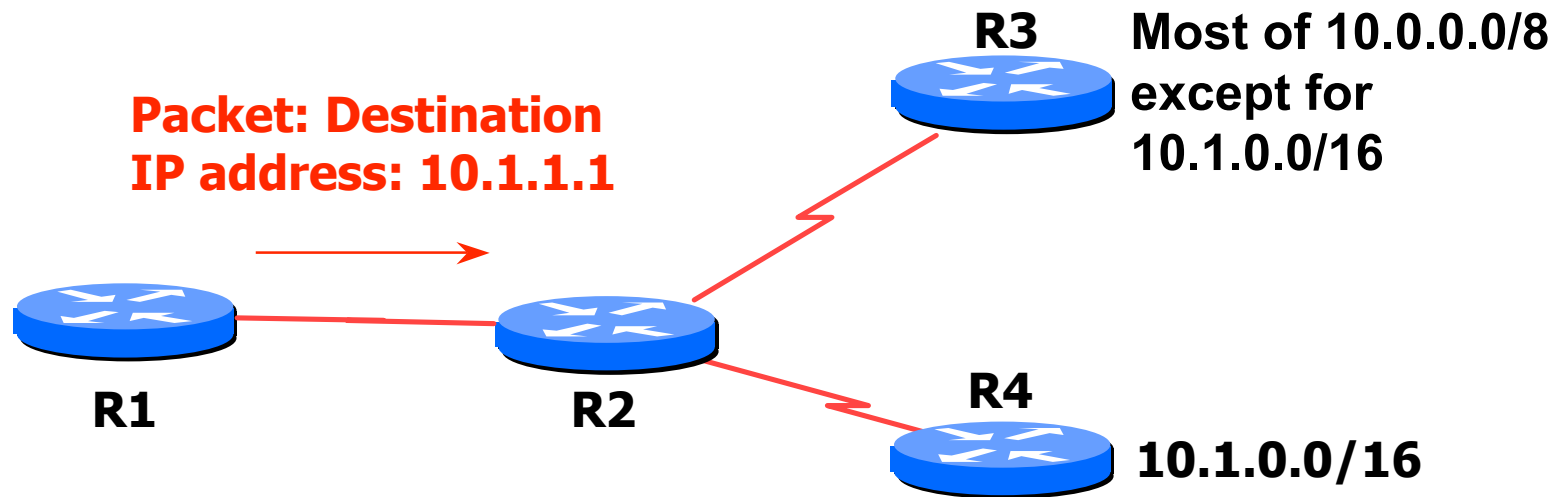
Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

10.1.1.1 & FF.00.00.00
vs.
20.0.0.0 & FF.00.00.00
No Match!

IP route lookup: Longest match routing



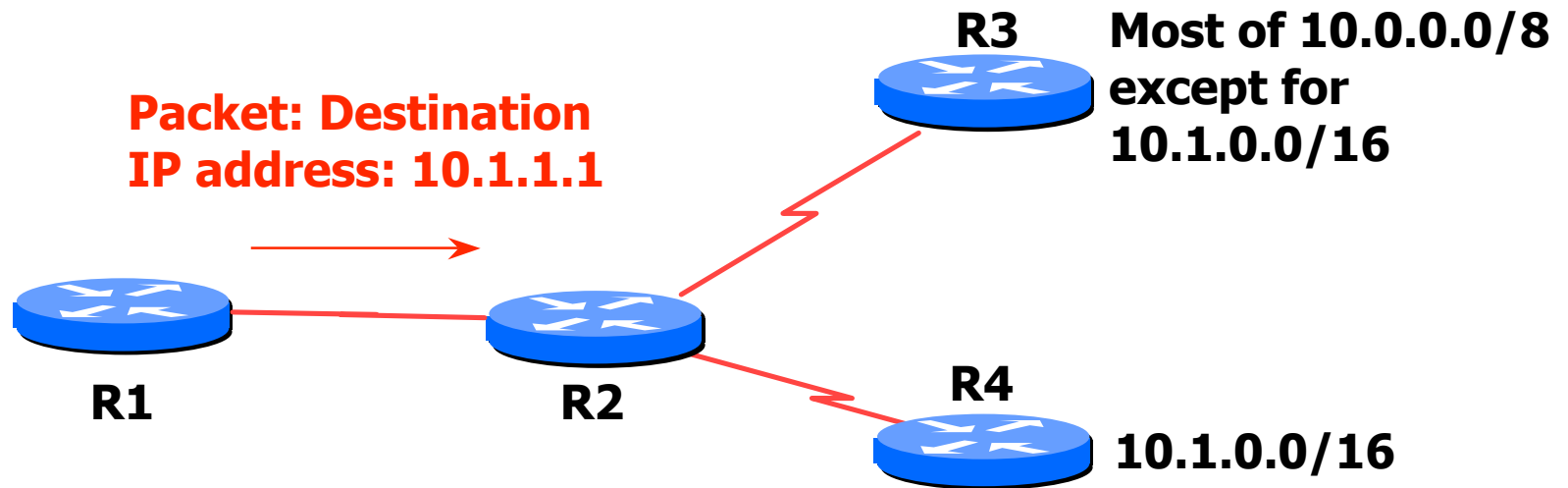
Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

10.1.1.1 & 00.00.00.00
vs.
0.0.0.0 & 00.00.00.00
Match! (length 0)

IP route lookup: Longest match routing



Based on destination IP address

R2's IP forwarding table

10.0.0.0/8	→ R3
10.1.0.0/16	→ R4
20.0.0.0/8	→ R5
0.0.0.0/0	→ R1

This is the longest matching prefix (length 16). "R2" will send the packet to "R4".

IP route lookup:

Longest match routing

- Most specific/longest match always wins!!
 - Many people forget this, even experienced ISP engineers
- Default route is 0.0.0.0/0
 - Can handle it using the normal longest match algorithm
 - Matches everything. Always the shortest match.



Static vs. Dynamic routing

- Static routes

- Set up by administrator
- Changes need to be made by administrator
- Only good for small sites and star topologies
- Bad for every other topology type

- Dynamic routes

- Provided by routing protocols
- Changes are made automatically
- Good for network topologies which have redundant links (most!)



Dynamic Routing

- Routers compute routing tables dynamically based on information provided by other routers in the network
- Routers communicate topology to each other via different protocols
- Routers then compute one or more next hops for each destination – trying to calculate the most optimal path
- Automatically repairs damage by choosing an alternative route (if there is one)



BGP Part 2

Interior and Exterior Routing



Interior vs. Exterior Routing Protocols

- Interior gateway protocol (IGP)
 - Automatic neighbour discovery
 - Under control of a single organisation
 - Generally trust your IGP routers
 - Routes go to all IGP routers
 - Usually not filtered
- Exterior gateway protocol (EGP)
 - Specifically configured peers
 - Connecting with outside networks
 - Neighbours are not trusted
 - Set administrative boundaries
 - Filters based on policy



IGP

- Interior Gateway Protocol
- Within a network/autonomous system
- Carries information about internal prefixes
- Examples – OSPF, ISIS, EIGRP



EGP

- Exterior Gateway Protocol
- Used to convey routing information between networks/ASes
- De-coupled from the IGP
- Current EGP is BGP4



Why Do We Need an EGP?

- Scaling to large network
 - Hierarchy
 - Limit scope of failure
- Define administrative boundary
- Policy
 - Control reachability to prefixes



Scalability and policy issues

- Just getting direct line is not enough
- Need to work out how to do routing
 - Need to get local traffic between ISP's/peers
 - Need to make sure the peer ISP doesn't use us for transit
 - Need to control what networks to announce, what network announcements to accept to upstreams and peers

Scalability:

Not using static routes

- `ip route their_net their_gw`
- Does not scale
- Millions of networks around the world

Scalability:

Not using IGP (OSPF)

- Serious operational consequences:
 - If the other ISP has a routing problem, you will have problems too
 - Your network prefixes could end up in the other ISP's network — and vice-versa
 - Very hard to filter routes so that we don't inadvertently give transit



Using BGP instead

- BGP = Border Gateway Protocol
- BGP is an **exterior** routing protocol
- Focus on routing **policy**, not topology
- BGP can make 'groups' of networks (Autonomous Systems)
- Good route filtering capabilities
- Ability to isolate from other's problems



Border Gateway Protocol

- A Routing Protocol used to exchange routing information between networks
 - exterior gateway protocol
- Described in RFC4271
 - RFC4276 gives an implementation report on BGP-4
 - RFC4277 describes operational experiences using BGP-4
- The Autonomous System is BGP's fundamental operating unit
 - It is used to uniquely identify networks with a common routing policy



BGP Part 3

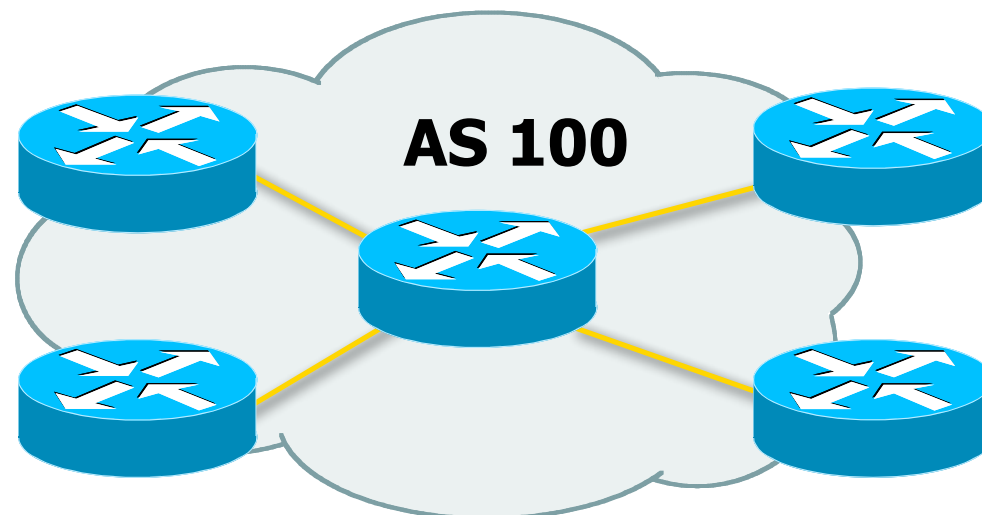
BGP Building Blocks



BGP Building Blocks

- Autonomous System (AS)
- Types of Routes
- IGP/EGP
- DMZ
- Policy
- Egress
- Ingress

Autonomous System (AS)



- Collection of networks with same policy
- Single routing protocol
- Usually under single administrative control
- IGP to provide internal connectivity



Autonomous System (AS)

- Autonomous systems is a misnomer
 - Not much to do with freedom, independence, ...
- Just a handle for a group of networks that is under the same administrative control
- Identified by an AS number



Autonomous System (AS)

- Identified by 'AS number'
 - example: AS16907 (ISPKenya)
- Examples:
 - Service provider
 - Multi-homed customers
 - Anyone needing policy discrimination for networks with different routing policies
- Single-homed network (one upstream provider) does not need an AS number
 - Treated like part of upstream AS



Autonomous System numbers

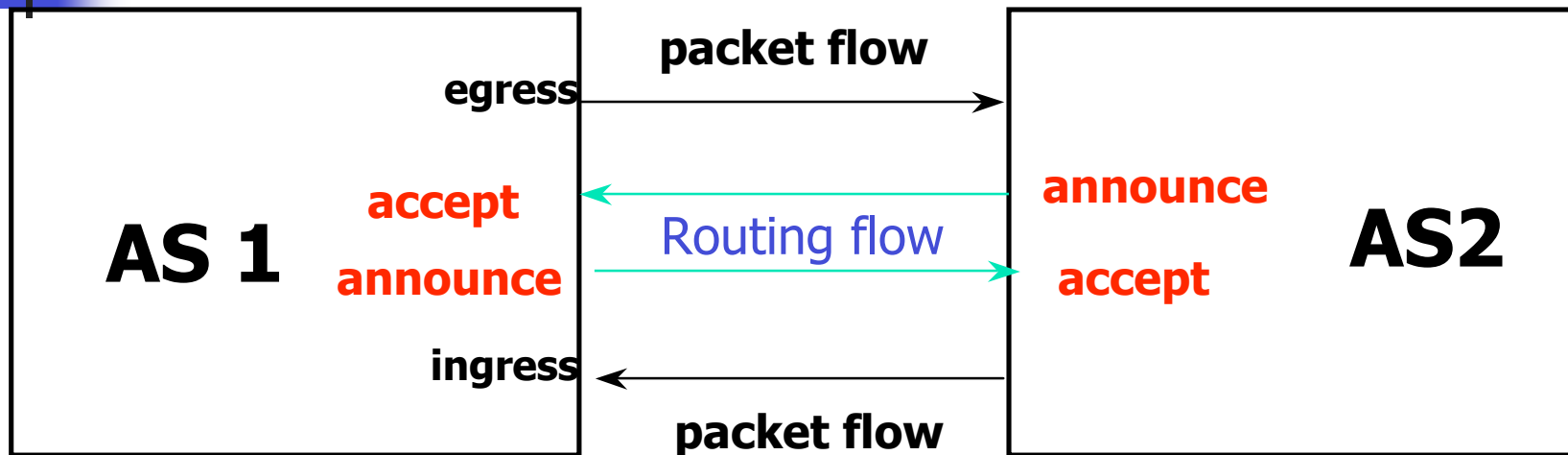
- 16-bit number, 1-65534
 - AS 1 to AS 64511 are for normal use.
 - Assigned by registry, just like IP numbers
- AS 0 and AS 65535 are reserved
- Top 1024 AS numbers (AS64512-AS65534) are private numbers
 - see RFC1930 for details



Using AS numbers

- BGP can filter on AS numbers
 - Get all networks of the other ISP using one handle
 - Include future new networks without having to change routing filters
 - AS number for new network will be same
 - Can use AS numbers in filters with regular expressions
- BGP actually does routing computation on IP numbers

Routing flow and packet flow



- For networks in AS1 and AS2 to communicate:
 - AS1 must announce routes to AS2
 - AS2 must accept routes from AS1
 - AS2 must announce routes to AS1
 - AS1 must accept routes from AS2



Egress Traffic

- Packets exiting the network
- Based on:
 - Route availability (what others send you)
 - Route acceptance (what you accept from others)
 - **Policy** and tuning (what you do with routes from others)
 - Peering and transit agreements



Ingress Traffic

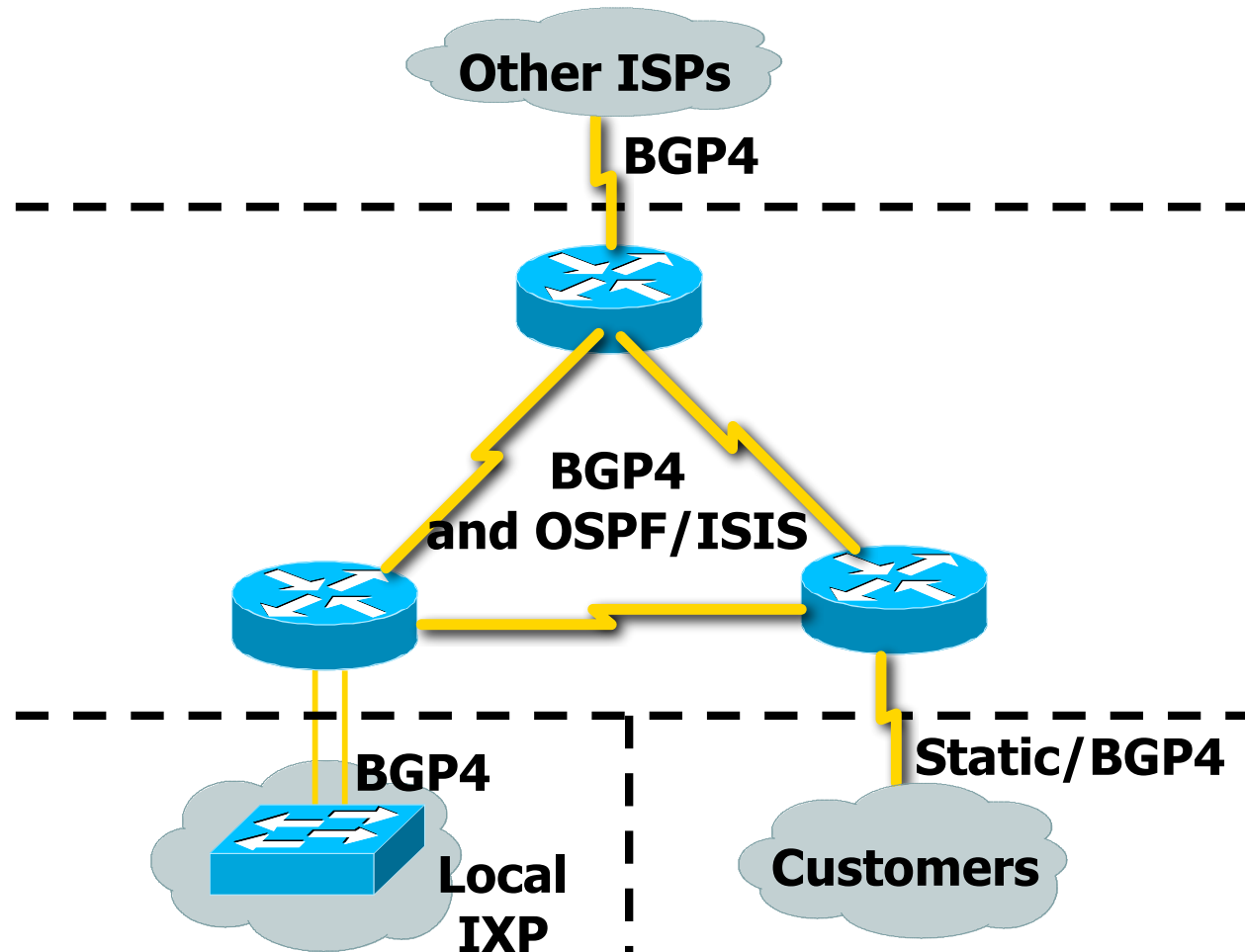
- Packets entering your network
- Ingress traffic depends on:
 - What information you send and to whom
 - Based on your addressing and ASes
 - Based on others' policy (what they accept from you and what they do with it)



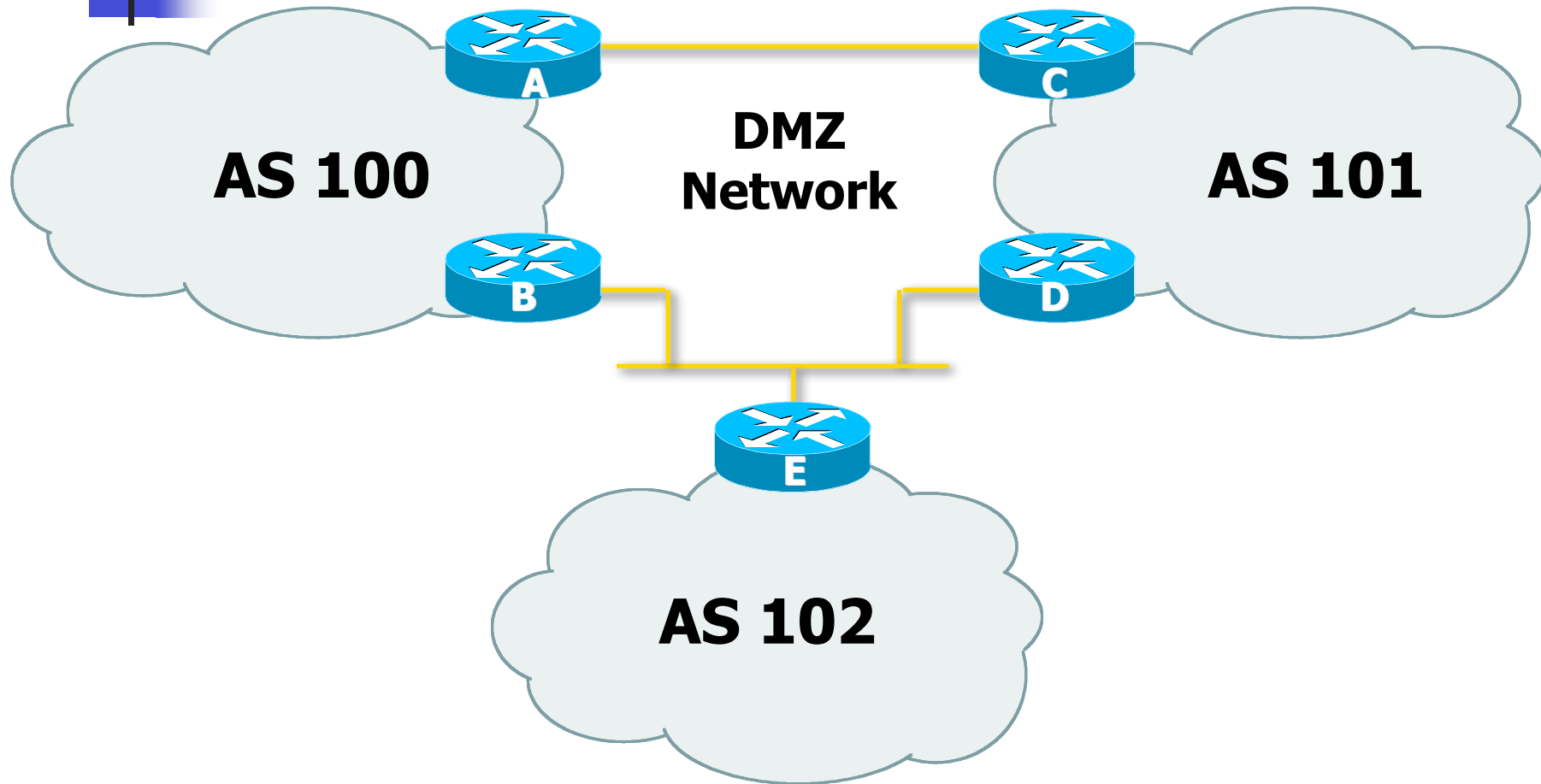
Types of Routes

- Static Routes
 - configured manually
- Connected Routes
 - created automatically when an interface is 'up'
- Interior Routes
 - Routes within an AS
 - learned via IGP (e.g. OSPF)
- Exterior Routes
 - Routes exterior to AS
 - learned via EGP (e.g. BGP)

Hierarchy of Routing Protocols



DeMarcation Zone (DMZ)



- Shared network between ASes



Basics of a BGP route

- Seen from output of "sh ip bgp"
- Prefix and mask — what IP addresses are we talking about?
 - 192.168.0.0/16 or 192.168.0.0/255.255.0.0
- Origin — How did the route originally get into BGP?
 - "?" — incomplete, "e" — EGP, "i" — IGP
- AS Path — what ASes did the route go through before it got to us?
 - "701 3561 1"



BGP Part 4

Configuring BGP
Basic commands
Getting routes into BGP



Basic BGP commands

Configuration commands

```
router bgp <AS-number>
```

```
no auto-summary
```

```
no synchronization
```

```
neighbor <ip address> remote-as <as-number>
```

Show commands

```
show ip bgp summary
```

```
show ip bgp neighbors
```

```
show ip bgp neighbor <ip address>
```



Inserting prefixes into BGP

- Two main ways to insert prefixes into BGP
 - network command
 - redistribute static
- Both require the prefix to be in the routing table



“network” command

- Configuration Example

```
router bgp 1
  network 105.32.4.0 mask 255.255.254.0
  ip route 105.32.4.0 255.255.254.0 serial 0
```

- matching route must exist in the routing table before network is announced!
- Prefix will have Origin code set to “IGP”



“redistribute static”

- Configuration Example:

```
router bgp 1
  redistribute static
  ip route 105.32.4.0 255.255.254.0 serial0
```

- Static route must exist before redistribute command will work
- Forces origin to be “incomplete”
- Care required!
 - This will redistribute all static routes into BGP
 - Redistributing without using a filter is dangerous



“redistribute static”

- Care required with redistribution
 - redistribute <routing-protocol> means everything in the <routing-protocol> will be transferred into the current routing protocol
 - will not scale if uncontrolled
 - best avoided if at all possible
 - redistribute normally used with “route-maps” and under tight administrative control
 - “route-map” is a kind of filter



Aggregates and Null0

- Remember: matching route must exist in routing table before it will be announced by BGP

```
router bgp 1
  network 105.32.0.0 mask 255.255.0.0
  ip route 105.32.0.0 255.255.0.0 null0 250
```
- Static route to null0 often used for aggregation
 - Packets will be sent here if there is no more specific match in the routing table
 - Distance of 250 ensures last resort
- Often used to nail up routes for stability
 - Can't flap! 😊



BGP Case Study 1 and Exercise 1

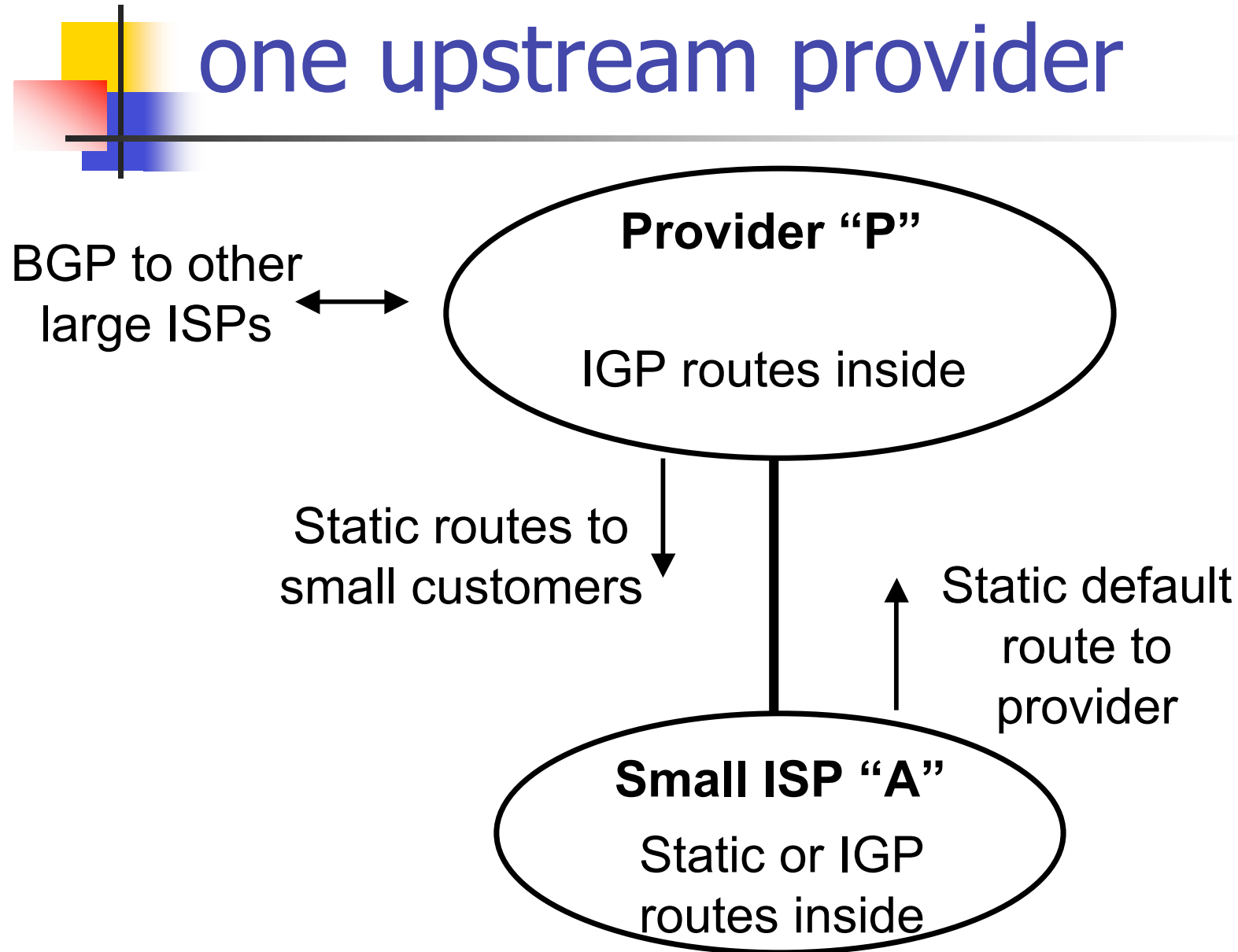
Small ISP with one upstream
provider



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

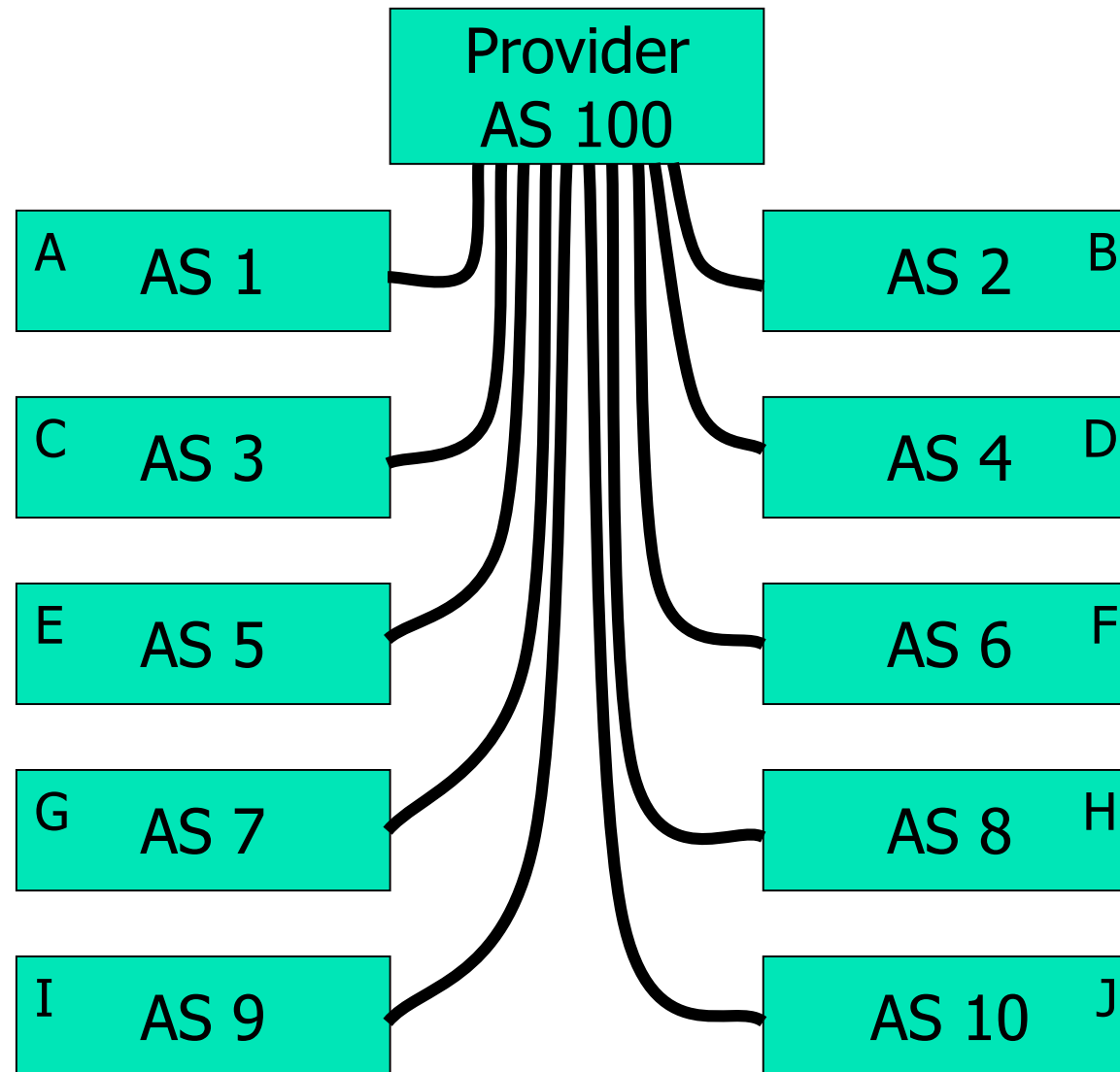
- No need for BGP between small ISP "A" and upstream provider "P"
- The outside world does not need to care about the link between provider "P" and customer "A"
- Hiding that information from the outside world helps with scaling
- **We will do an exercise using BGP even though it is not needed**



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 196.200.220.12, remote AS 100)



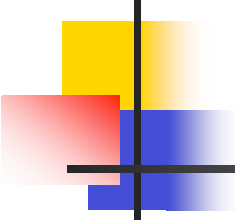
Exercise 1: Transit through upstream provider

- 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 routing table
- Try “show ip bgp” to see 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
 - `ip route x.x.x.x m.m.m.m Null0`



BGP Part 5

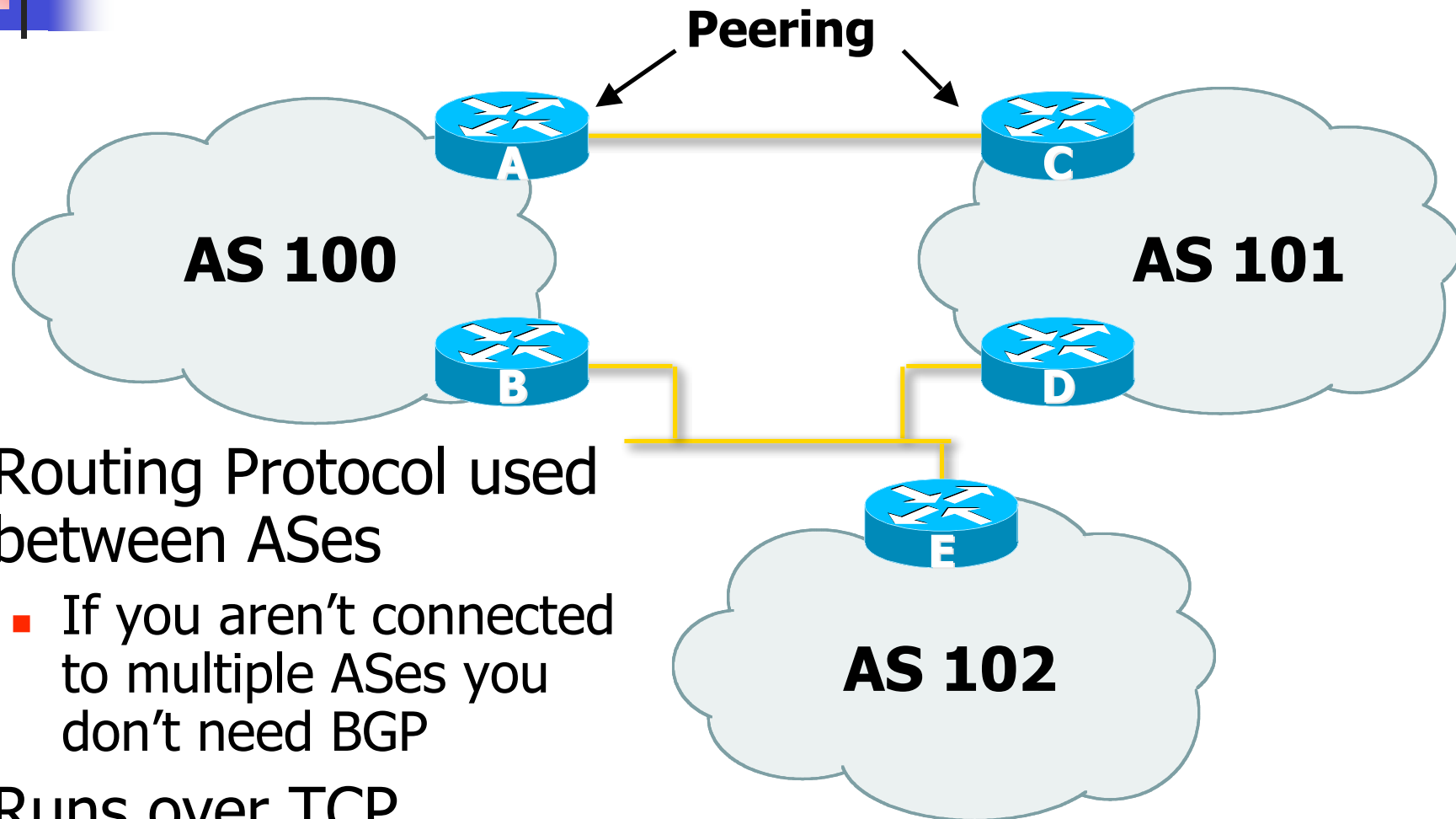
BGP Protocol Basics

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 neighbor



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 ...

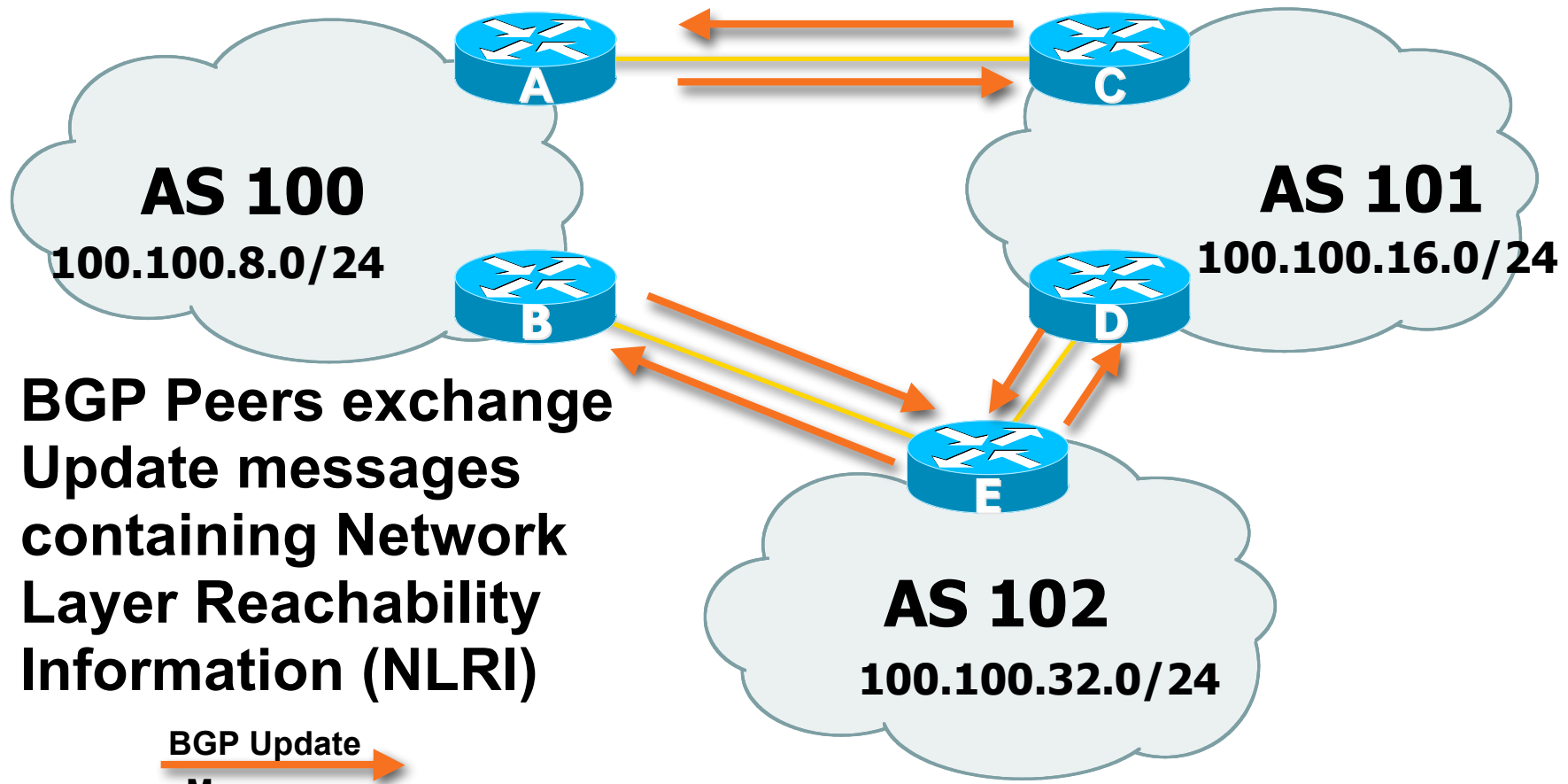
- Each AS originates a set of NLRI (routing announcements)
- NLRI is exchanged between BGP peers
- Can have multiple paths for a given prefix
- BGP picks the best path and installs in the IP forwarding table
- Policies applied (through attributes) influences BGP path selection



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.)

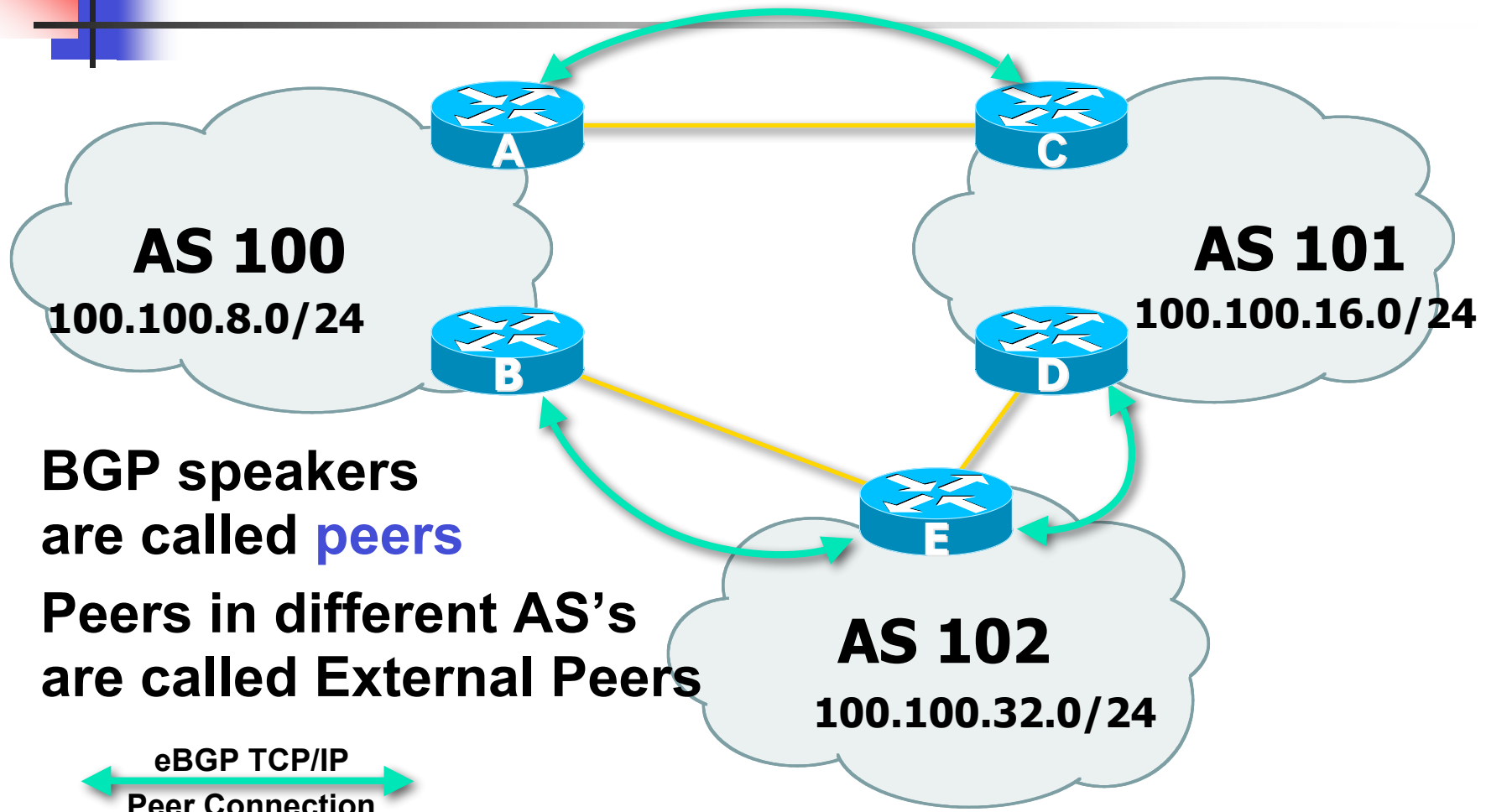
BGP Peers



BGP Peers exchange Update messages containing Network Layer Reachability Information (NLRI)

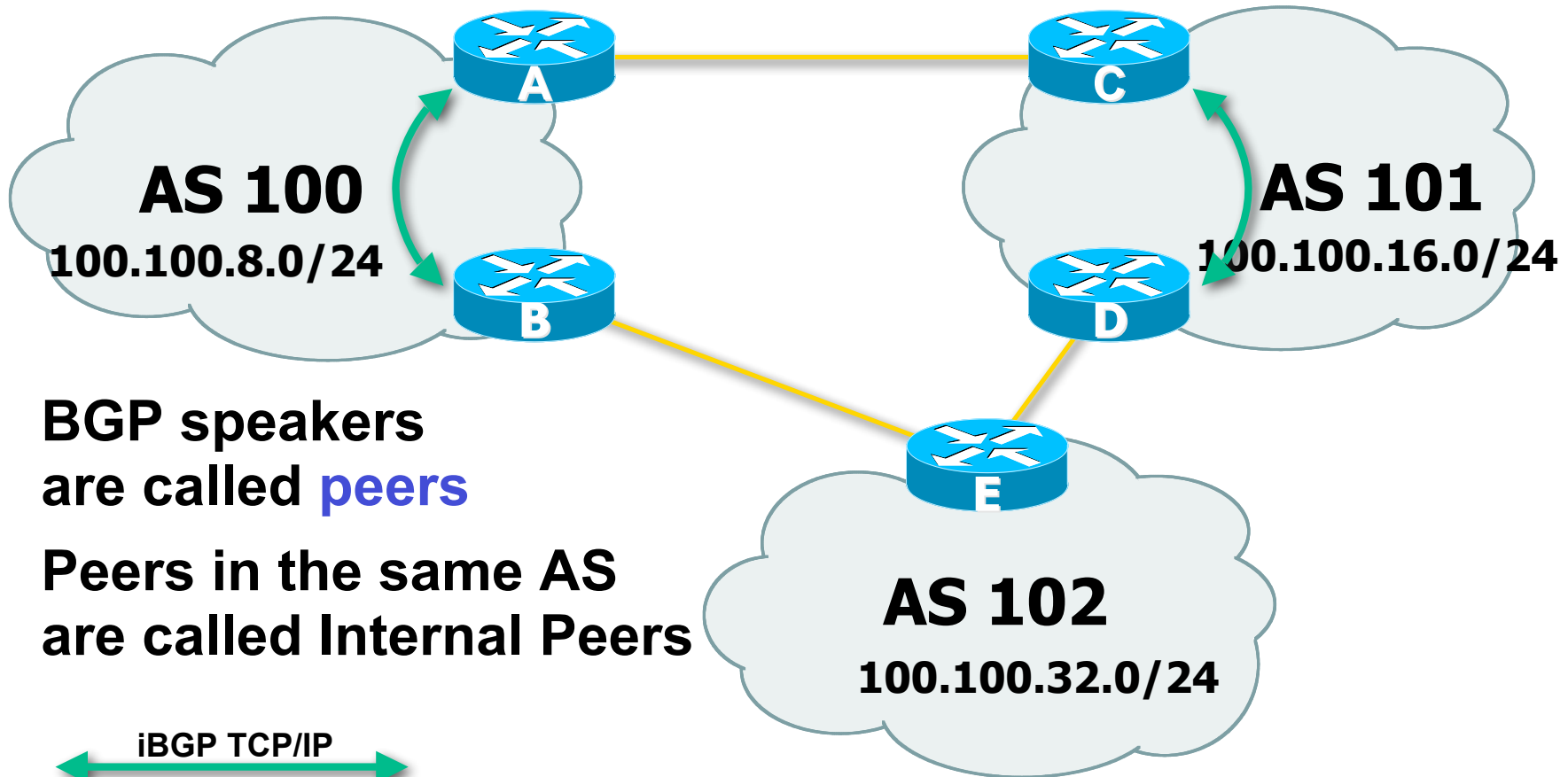
BGP Update Messages →

BGP Peers – External (eBGP)



Note: eBGP Peers normally should be directly connected.

BGP Peers – Internal (iBGP)

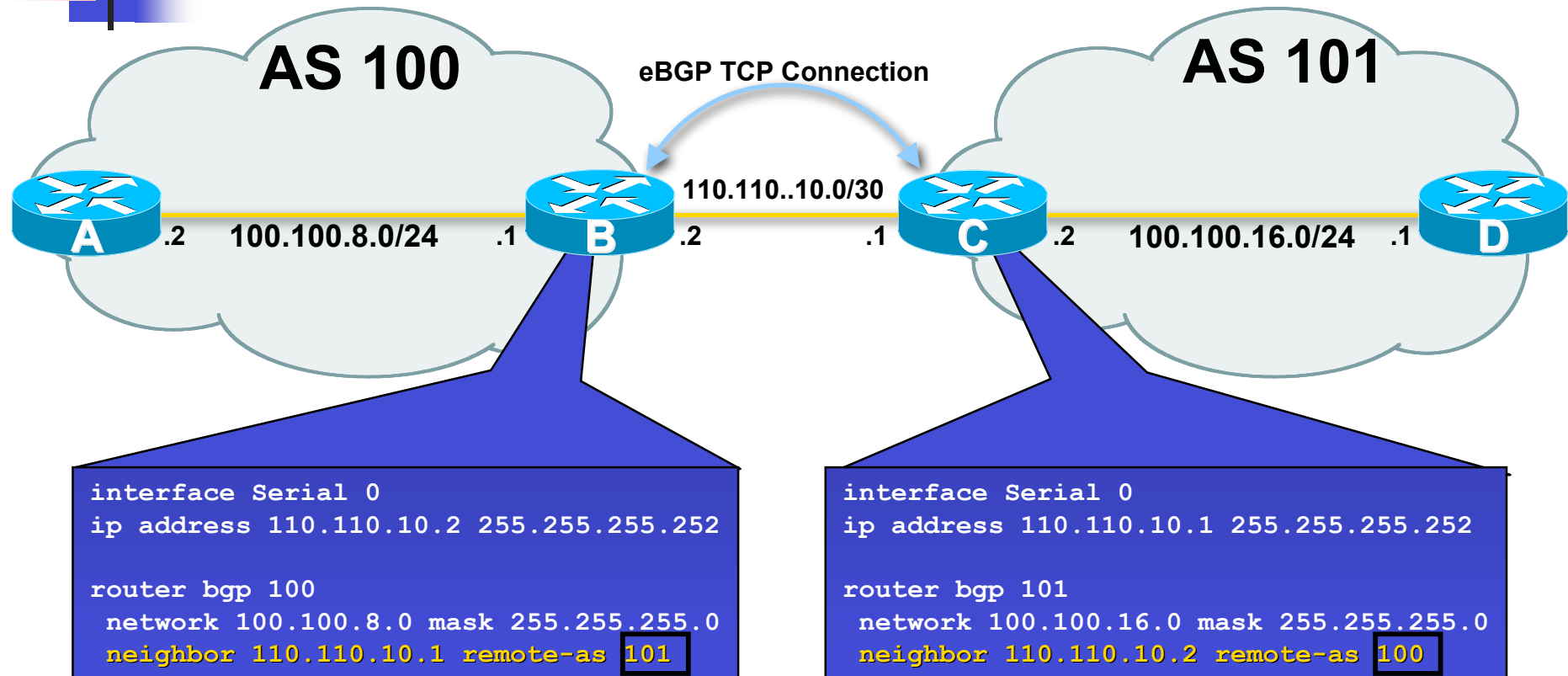


BGP speakers are called **peers**

Peers in the same AS are called **Internal Peers**

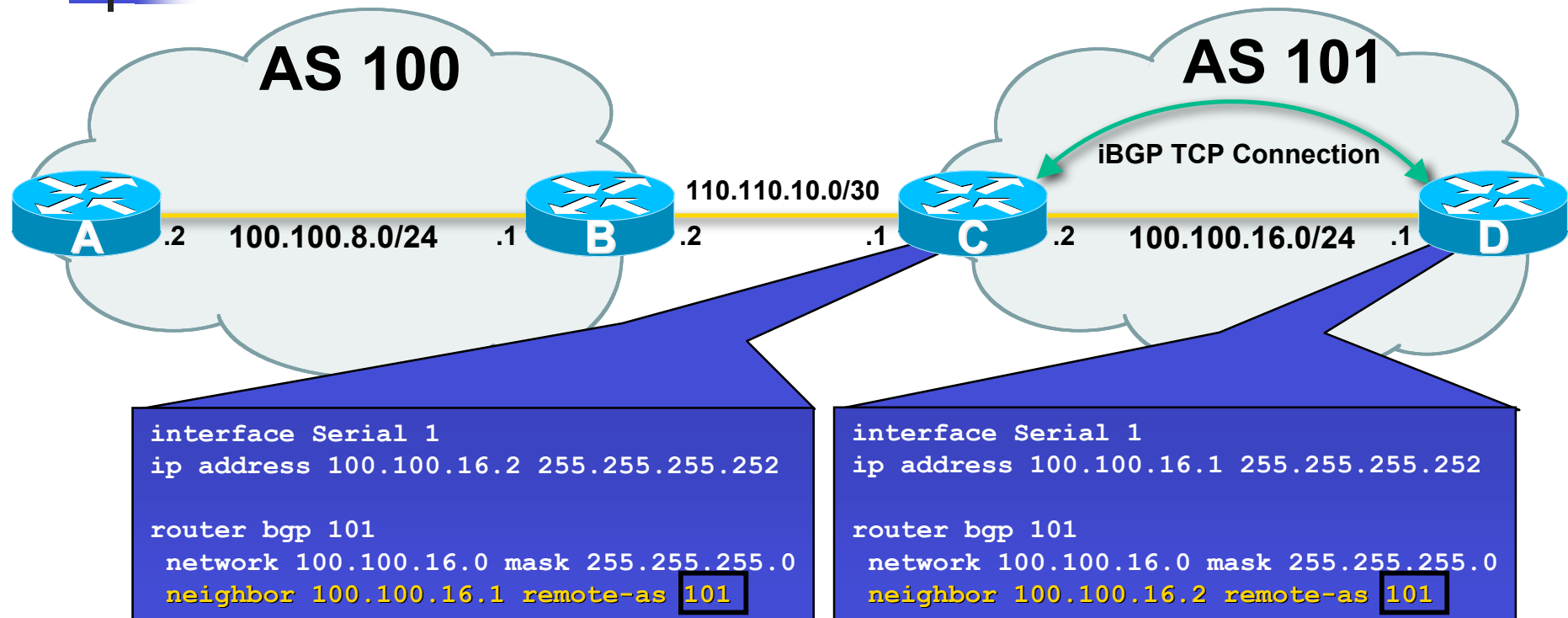
Note: iBGP Peers don't have to be directly connected.

Configuring eBGP peers



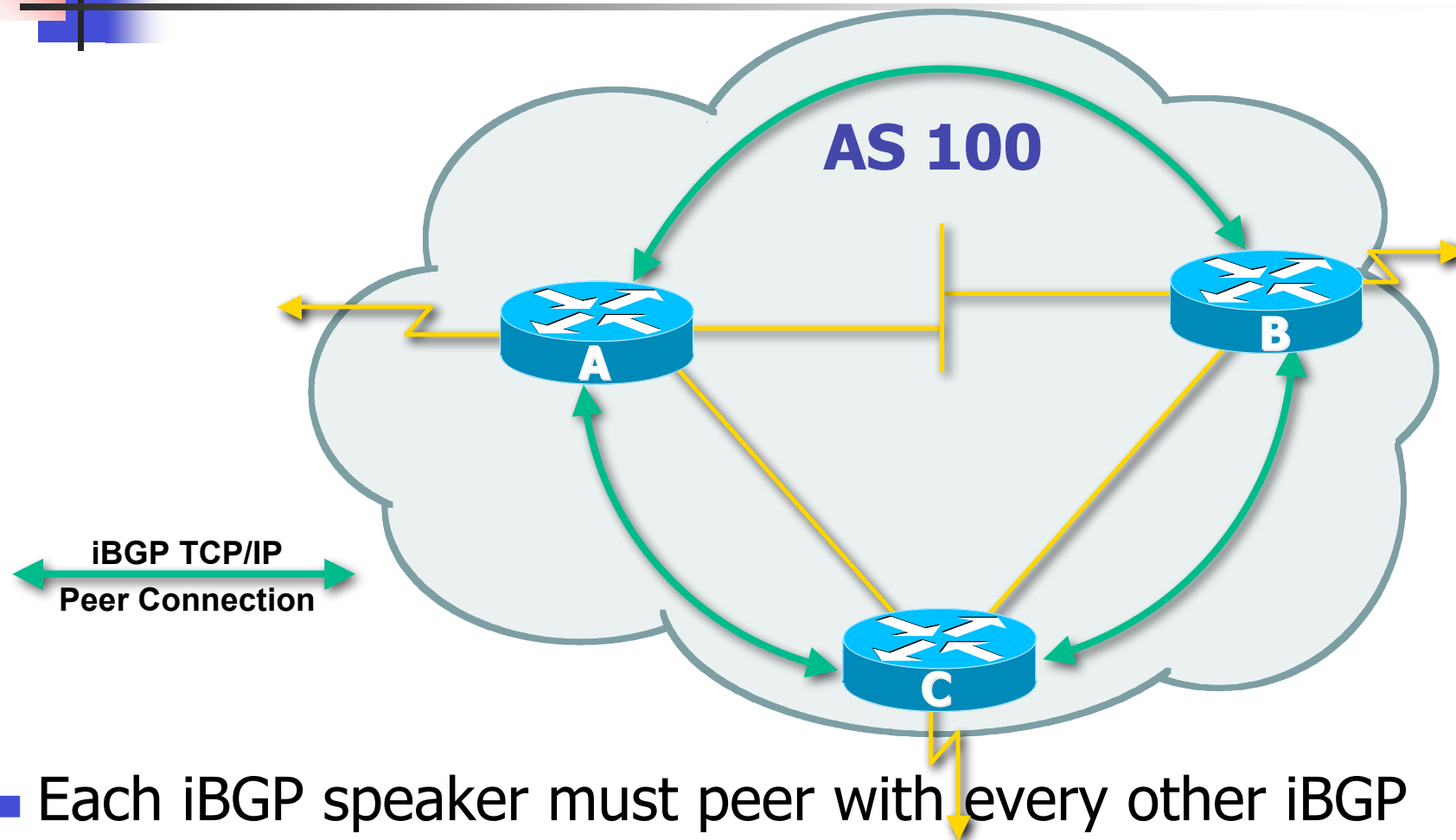
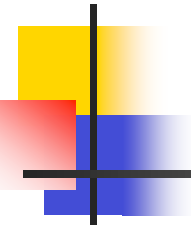
- BGP peering sessions are established using the BGP "neighbor" command
 - eBGP is configured when AS numbers are different

Configuring iBGP peers



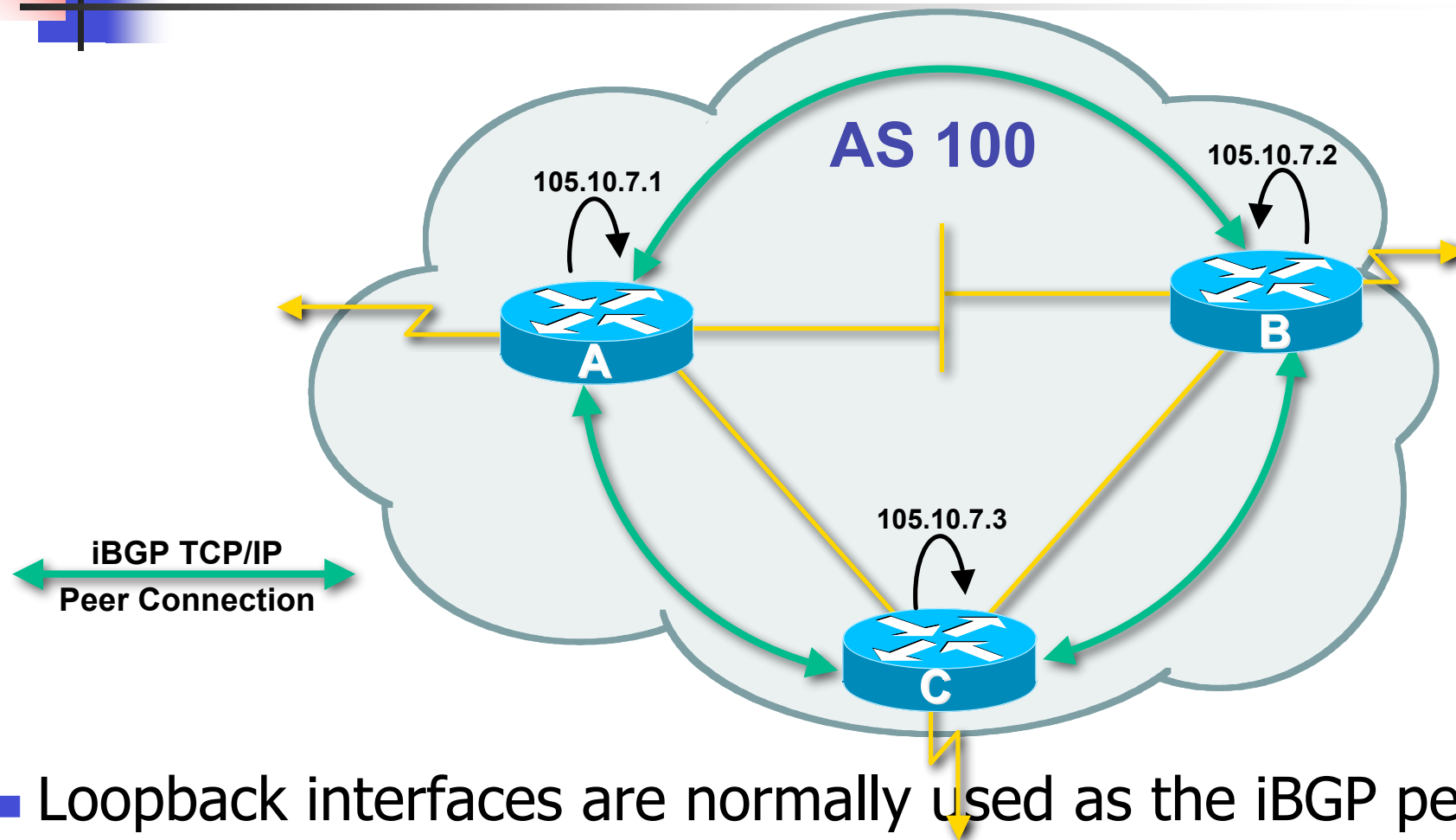
- BGP peering sessions are established using the BGP "neighbor" command
 - iBGP is configured when AS numbers are the same

Configuring iBGP peers: Full mesh



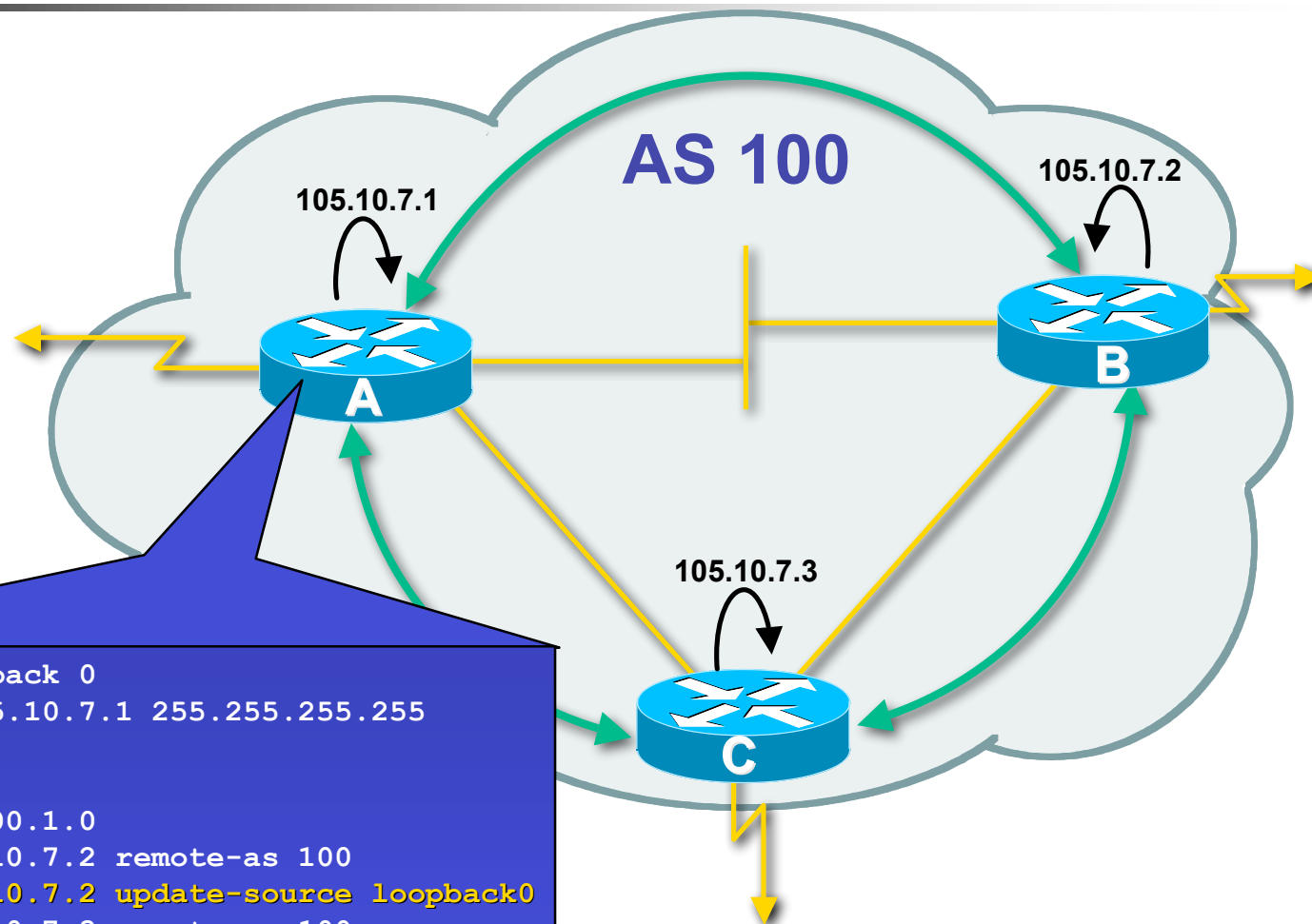
- Each iBGP speaker must peer with every other iBGP speaker in the AS

Configuring iBGP peers: Loopback interface



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

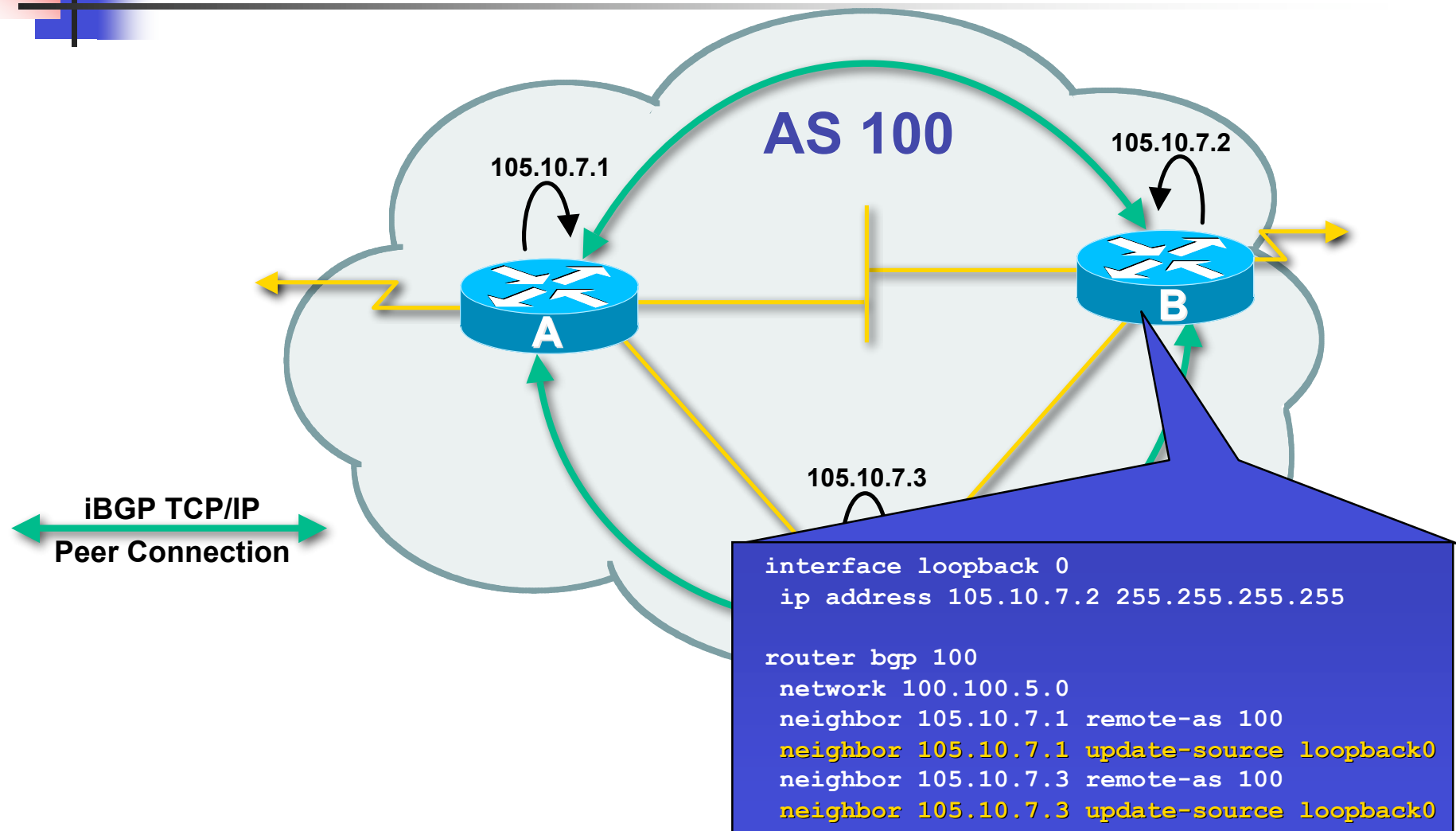
Configuring iBGP peers



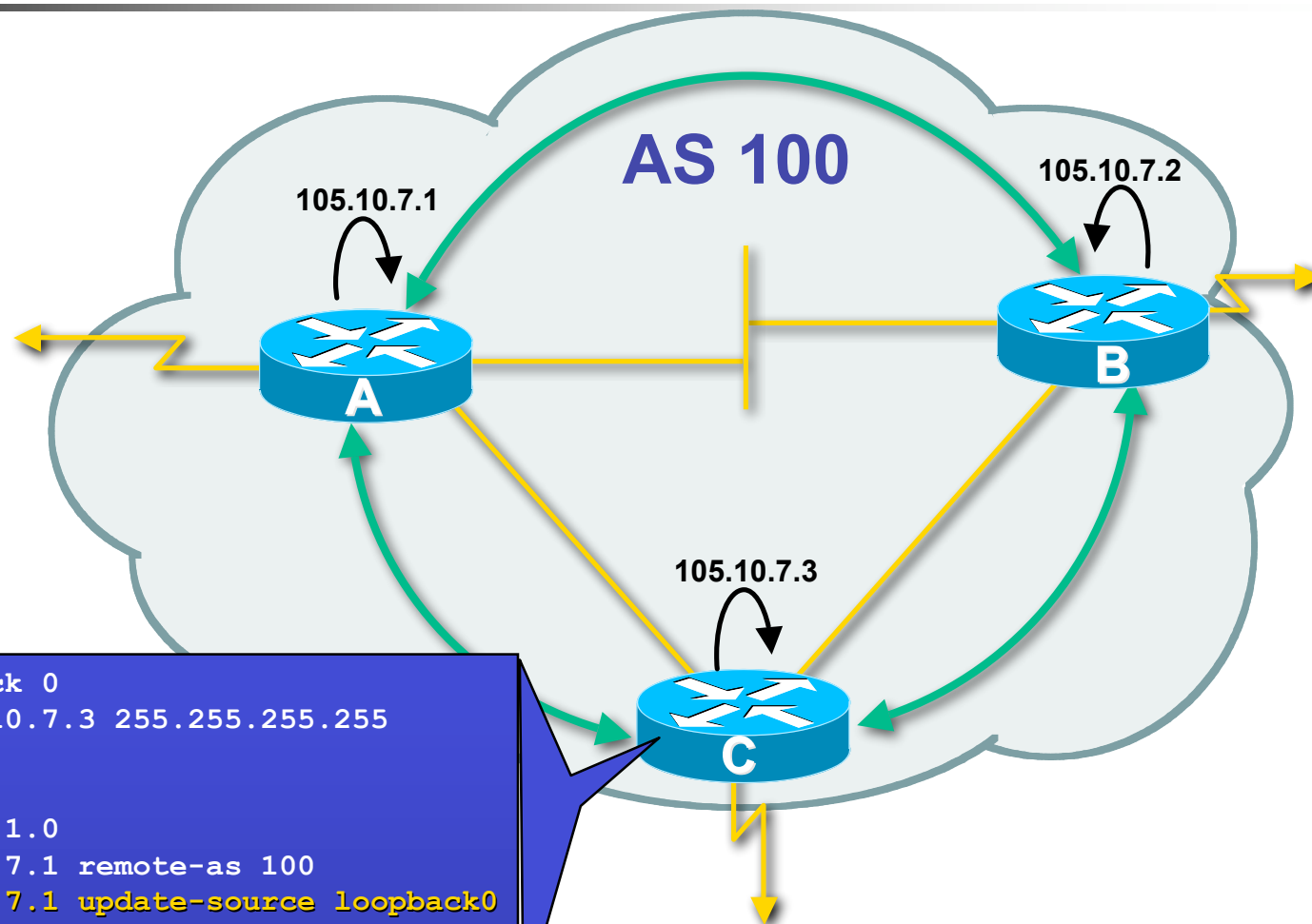
```
interface loopback 0
 ip address 105.10.7.1 255.255.255.255

router bgp 100
 network 100.100.1.0
 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 100.100.1.0
 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 6

BGP Protocol – A little more detail



BGP Updates — NLRI

- Network Layer Reachability Information
- Used to advertise feasible routes
- Composed of:
 - Network Prefix
 - Mask Length

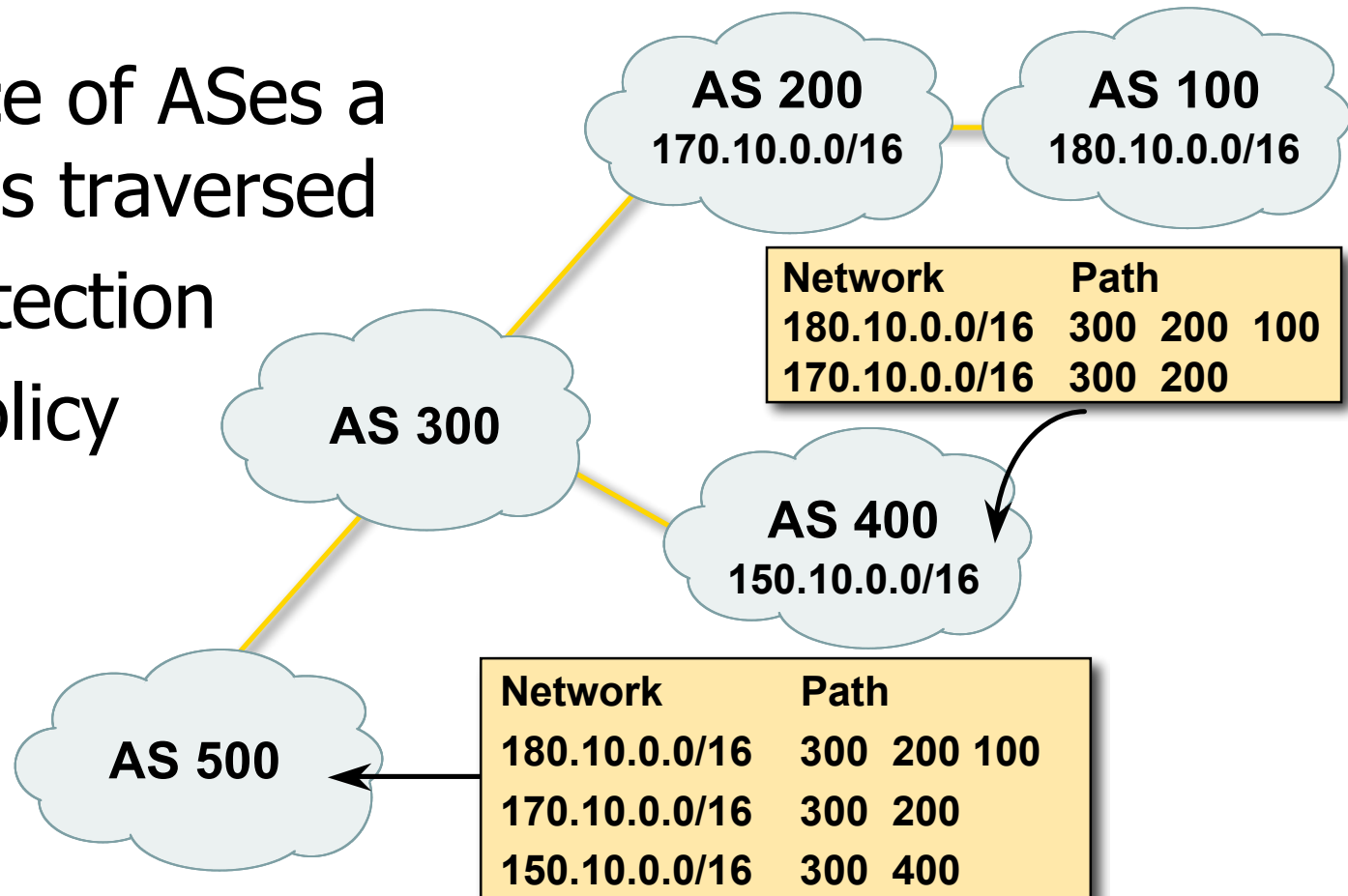


BGP Updates — Attributes

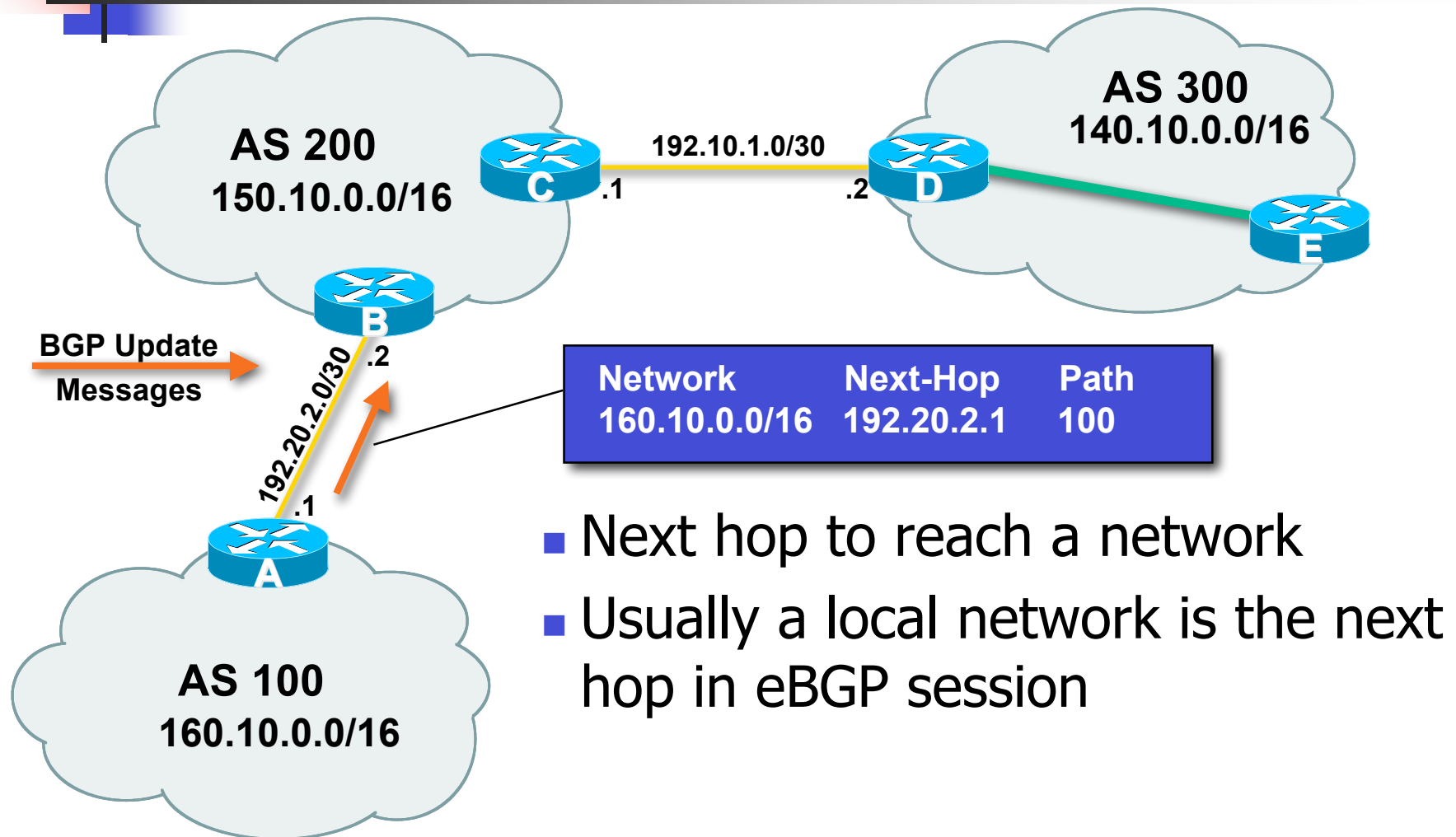
- Used to convey information associated with NLRI
 - AS path
 - Next hop
 - Local preference
 - Multi-Exit Discriminator (MED)
 - Community
 - Origin
 - Aggregator

AS-Path Attribute

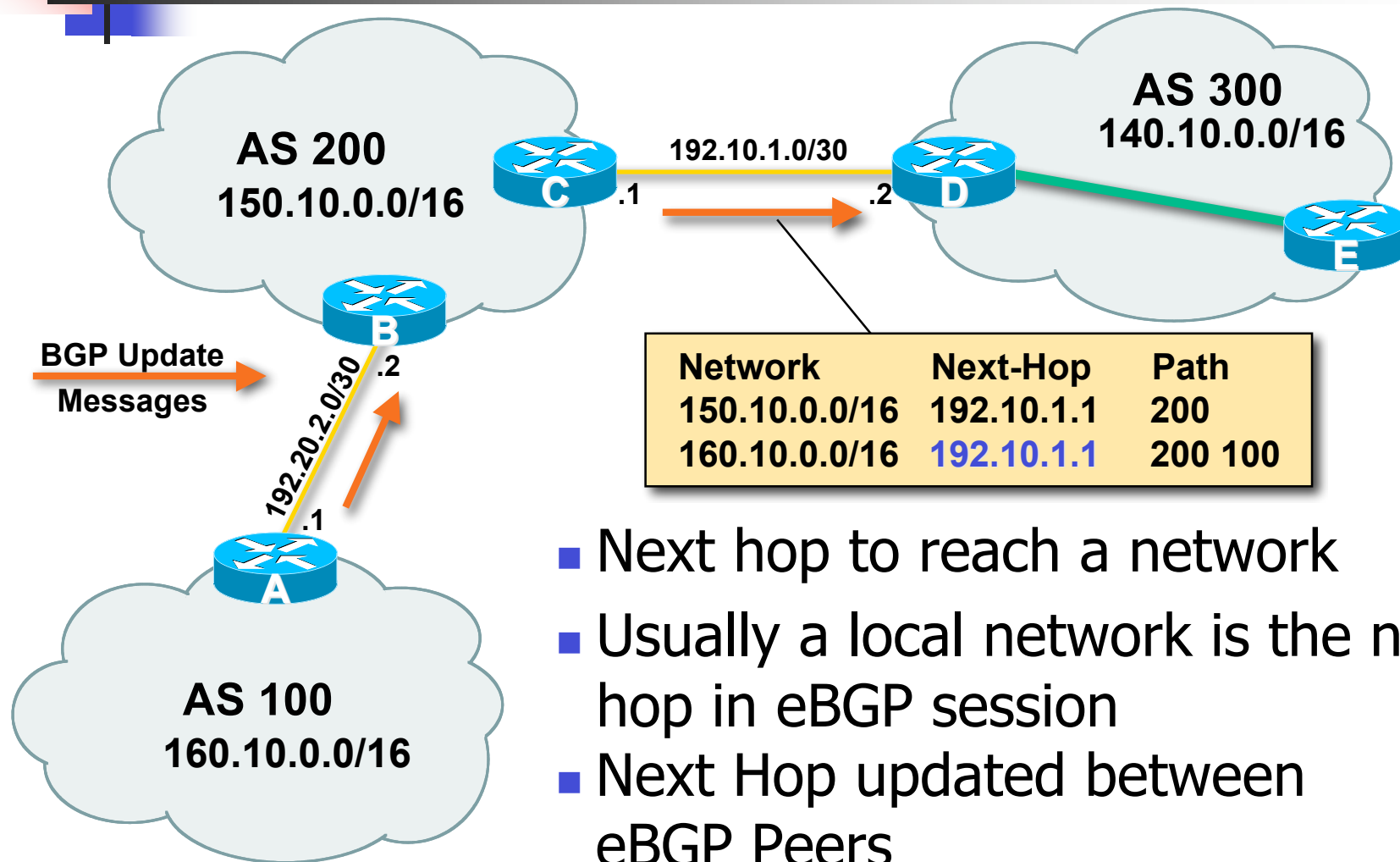
- Sequence of ASes a route has traversed
- Loop detection
- Apply policy



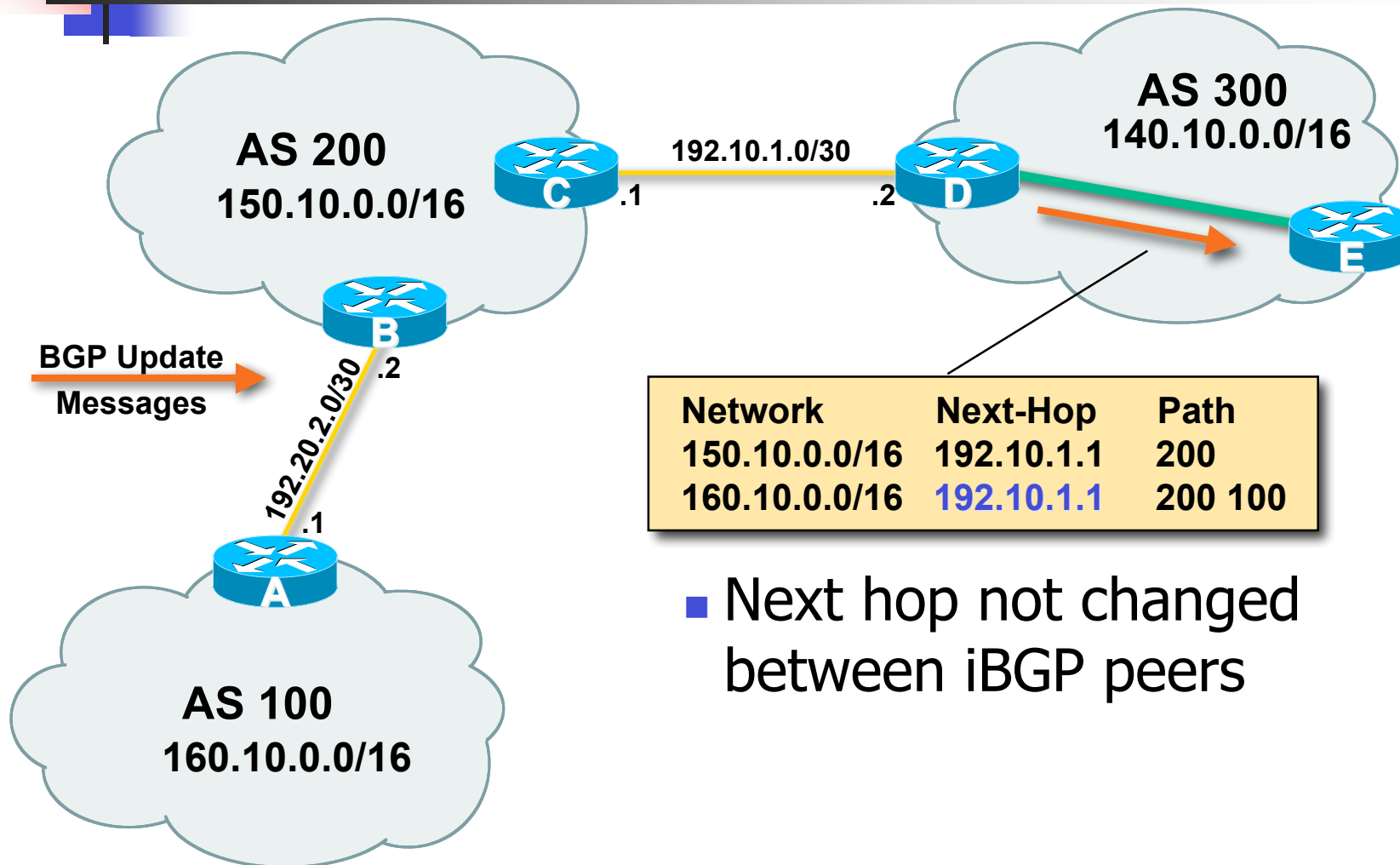
Next Hop Attribute



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



Community Attribute

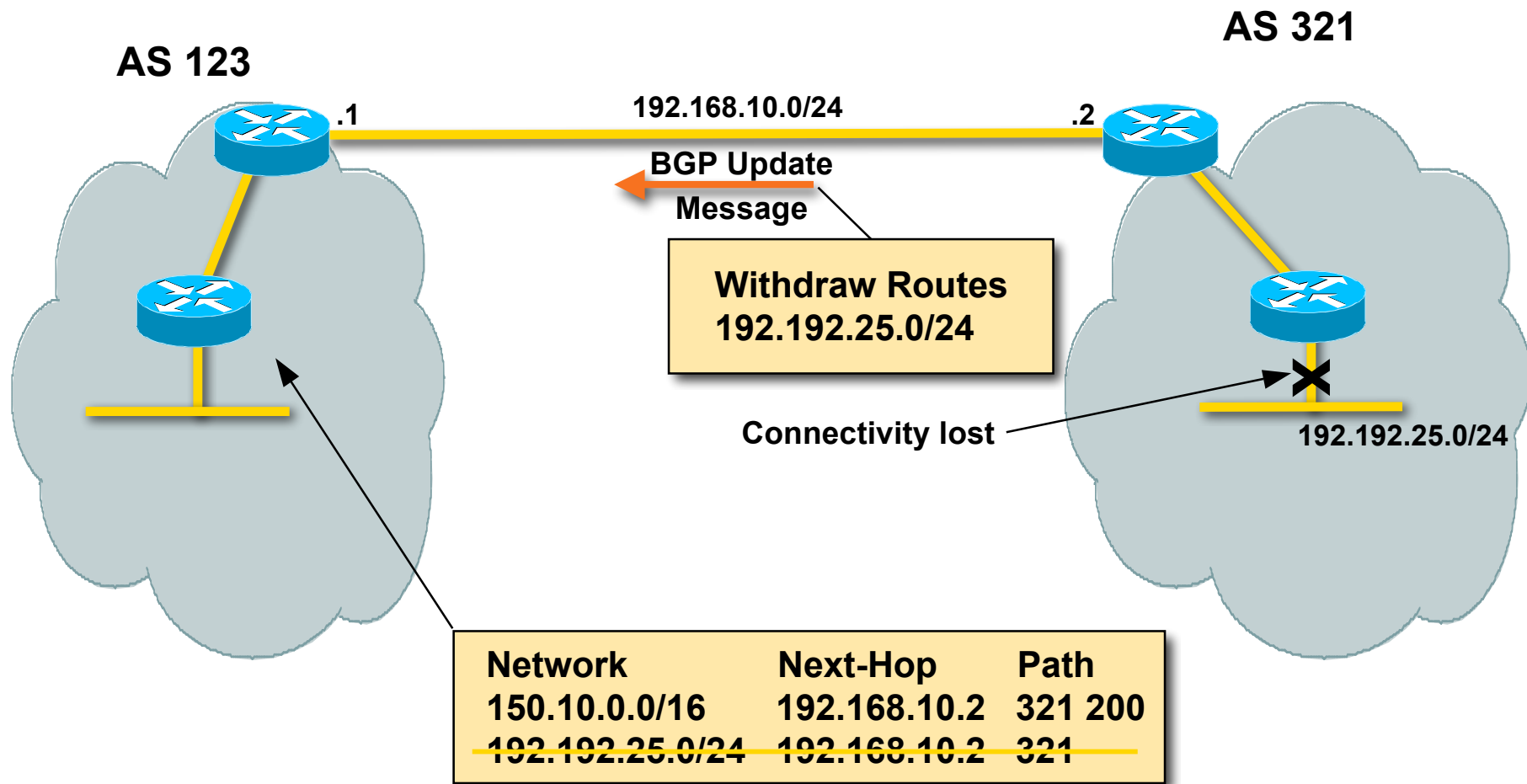
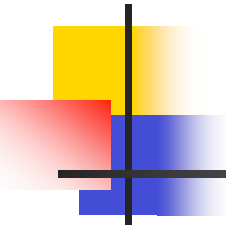
- 32-bit number
- Conventionally written as two 16-bit numbers separated by colon
 - First half is usually an AS number
 - That AS determines the meaning (if any) of the second half
- Carried in BGP protocol messages
 - Used by administratively-defined filters
 - Not directly used by BGP protocol (except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

D	10.1.2.0/24
D	160.10.1.0/24
D	160.10.3.0/24
R	153.22.0.0/16
S	192.1.1.0/24

Route Table

```
router bgp 100
network 160.10.1.0 255.255.255.0
network 160.10.3.0 255.255.255.0
no auto-summary
```

BGP 'network' commands are normally used to populate the BGP RIB with routes from the Route Table

BGP Routing Information Base

BGP RIB

Network	Next-Hop	Path
*> 160.10.0.0/16	0.0.0.0	i
* i	192.20.3.1	i
s> 160.10.1.0/24	192.20.3.1	i
s> 160.10.3.0/24	192.20.3.1	i

router bgp 100

network 160.10.0.0 255.255.0.0

aggregate-address 160.10.0.0 255.255.0.0 summary-only

no auto-summary

D	10.1.2.0/24
D	160.10.1.0/24
D	160.10.3.0/24
R	153.22.0.0/16
S	192.1.1.0/24

Route Table

BGP 'aggregate-address' commands may be used to install summary routes in the BGP RIB

BGP Routing Information Base

BGP RIB

Network	Next-Hop	Path
*> 160.10.0.0/16	0.0.0.0	i
* i	192.20.3.1	i
s> 160.10.1.0/24	192.20.3.1	i
s> 160.10.3.0/24	192.20.3.1	i
*> 192.1.1.0/24	192.20.3.1	?

D	10.1.2.0/24
D	160.10.1.0/24
D	160.10.3.0/24
R	153.22.0.0/16
S	192.1.1.0/24

Route Table

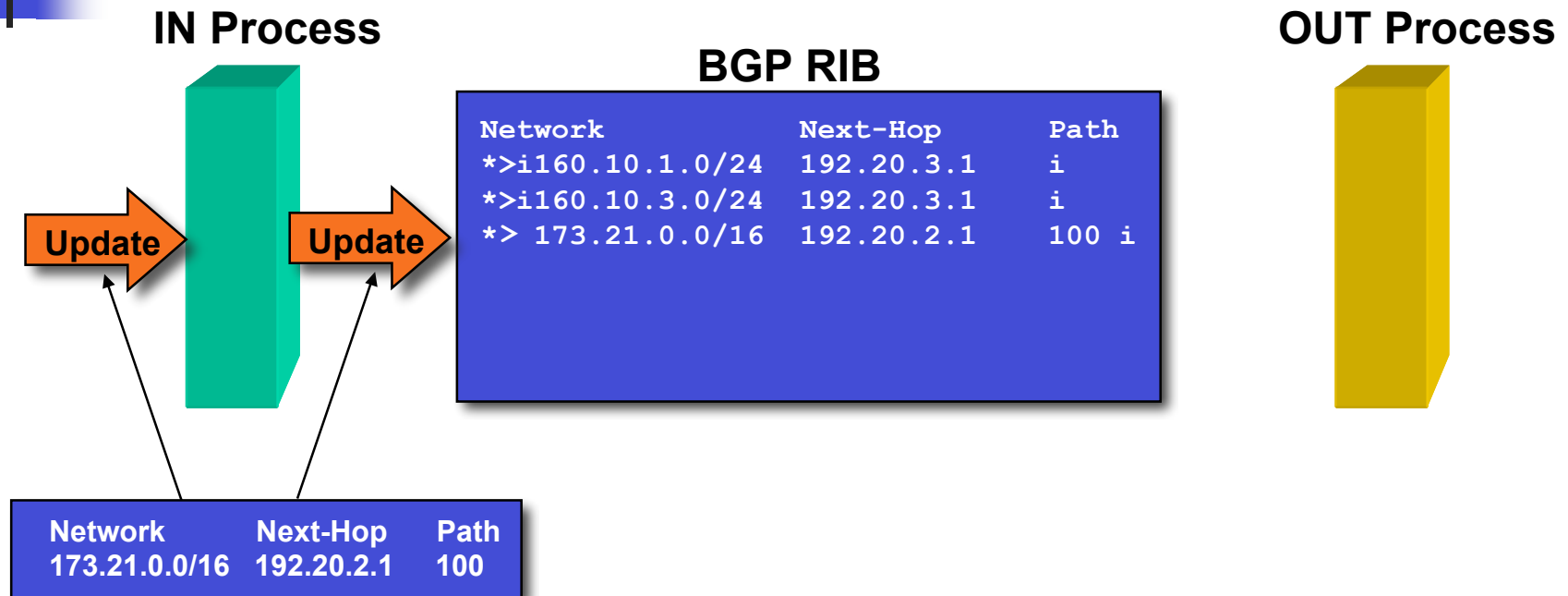
```
router bgp 100
network 160.10.0.0 255.255.0.0
redistribute static route-map foo
no auto-summary

access-list 1 permit 192.1.0.0 0.0.255.255

route-map foo permit 10
match ip address 1
```

BGP 'redistribute' commands can also be used to populate the BGP RIB with routes from the Route Table

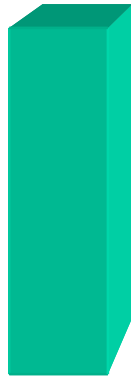
BGP Routing Information Base



- **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

BGP RIB

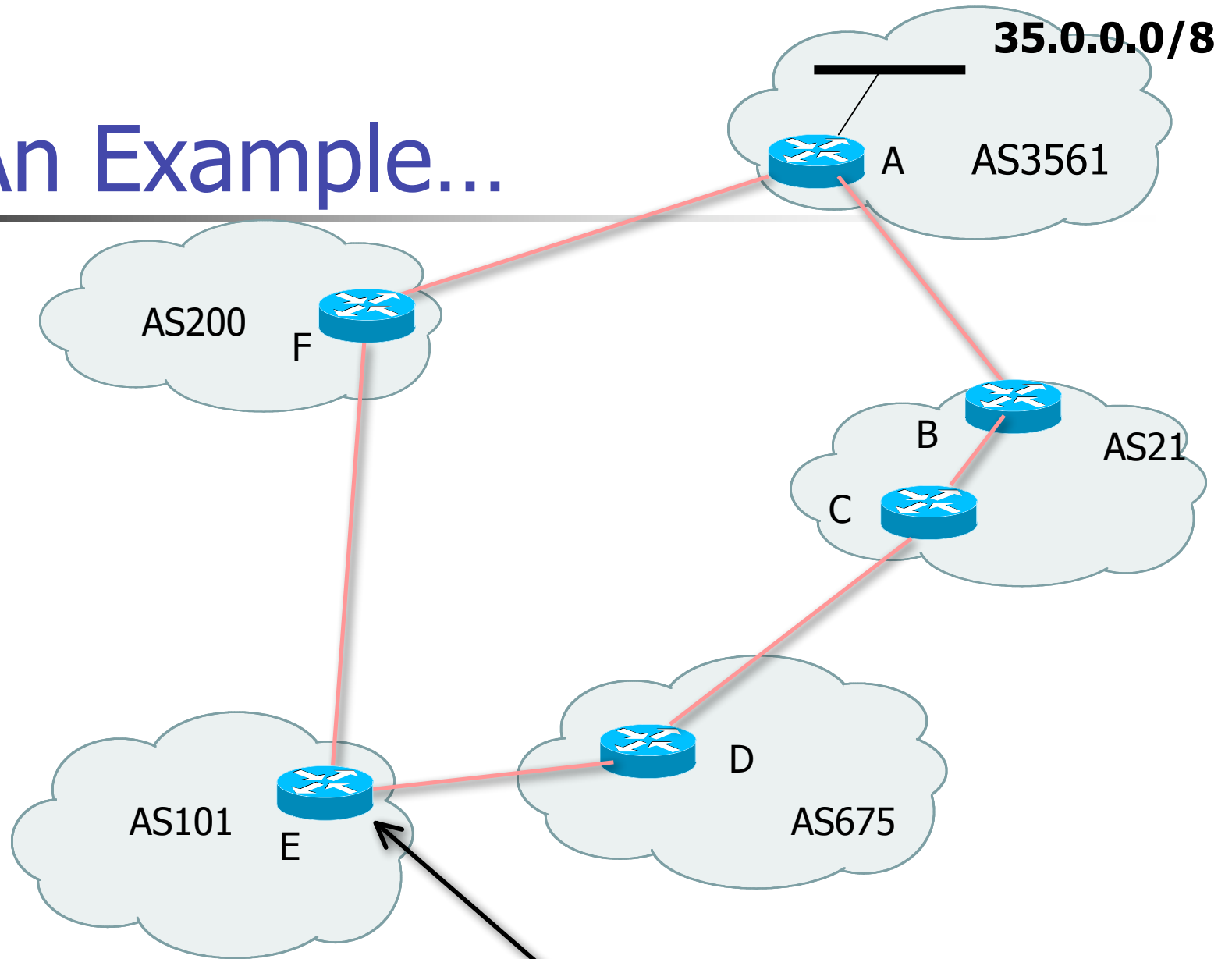
Network	Next-Hop	Path
*>i160.10.1.0/24	192.20.3.1	i
*>i160.10.3.0/24	192.20.3.1	i
*> 173.21.0.0/16	192.20.2.1	100

D	10.1.2.0/24
D	160.10.1.0/24
D	160.10.3.0/24
R	153.22.0.0/16
S	192.1.1.0/24
B	173.21.0.0/16

Route Table

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

An Example...



Learns about 35.0.0.0/8 from F & D



BGP Case Study 2 and Exercise 2

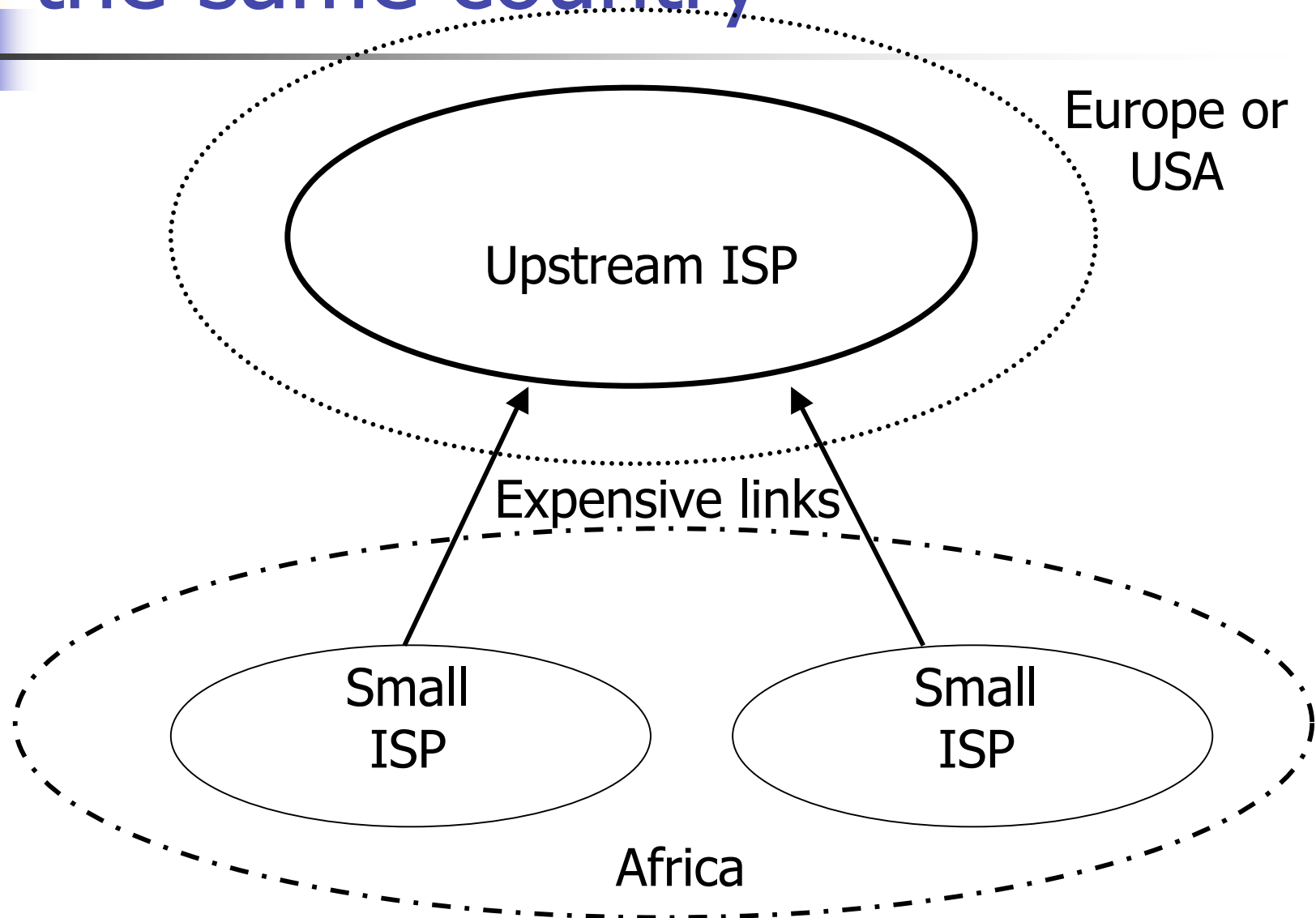
Small ISPs in the same locality
connect to each other



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

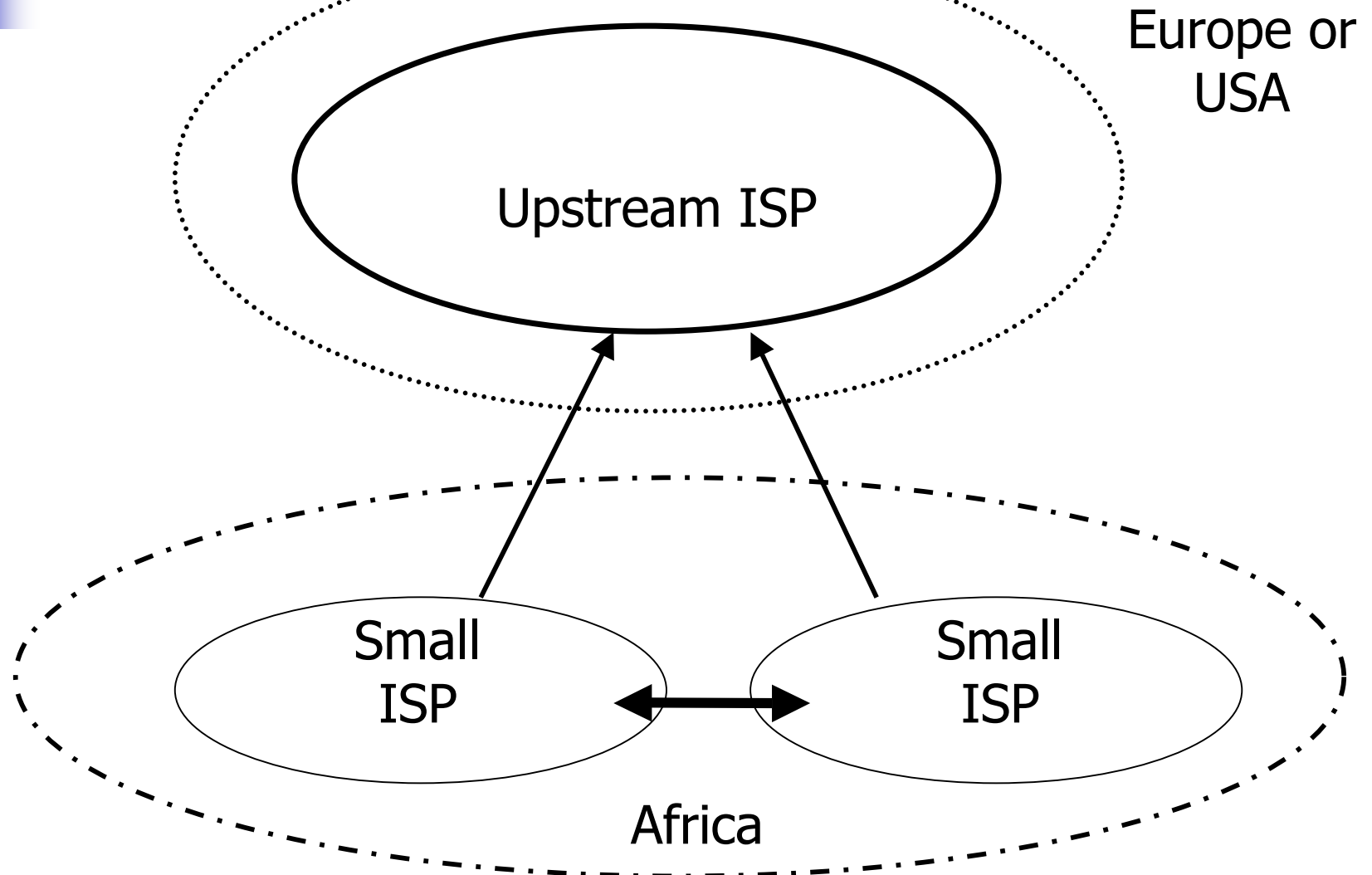




Case Study 2: Bringing down costs

- 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!

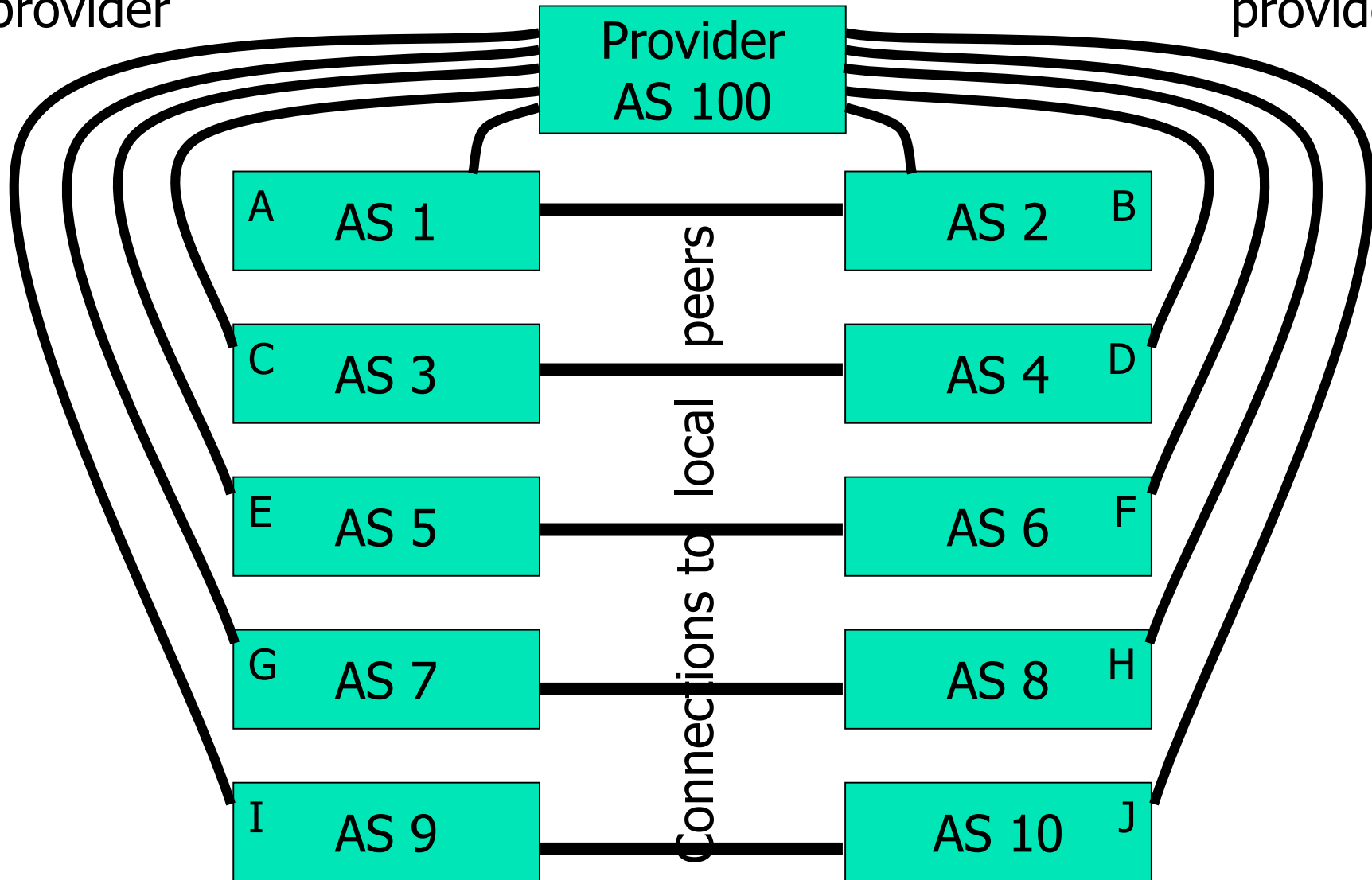
Case Study 2: Keeping Local Traffic Local



Exercise 2: Connect to another local ISP

Transit to
provider

Transit to
provider







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

- 
-
- Try “show ip route” to see forwarding table
 - Try “show ip bgp” to see BGP information
 - 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 7

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)
- `[abc]` matches a, or b, or c.
- There are many more possibilities



Filter list – using as-path access list

```
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
```

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).



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

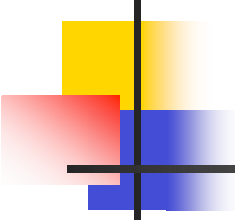
```
ip prefix-list Example 0.0.0.0/0 le 32
```

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



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



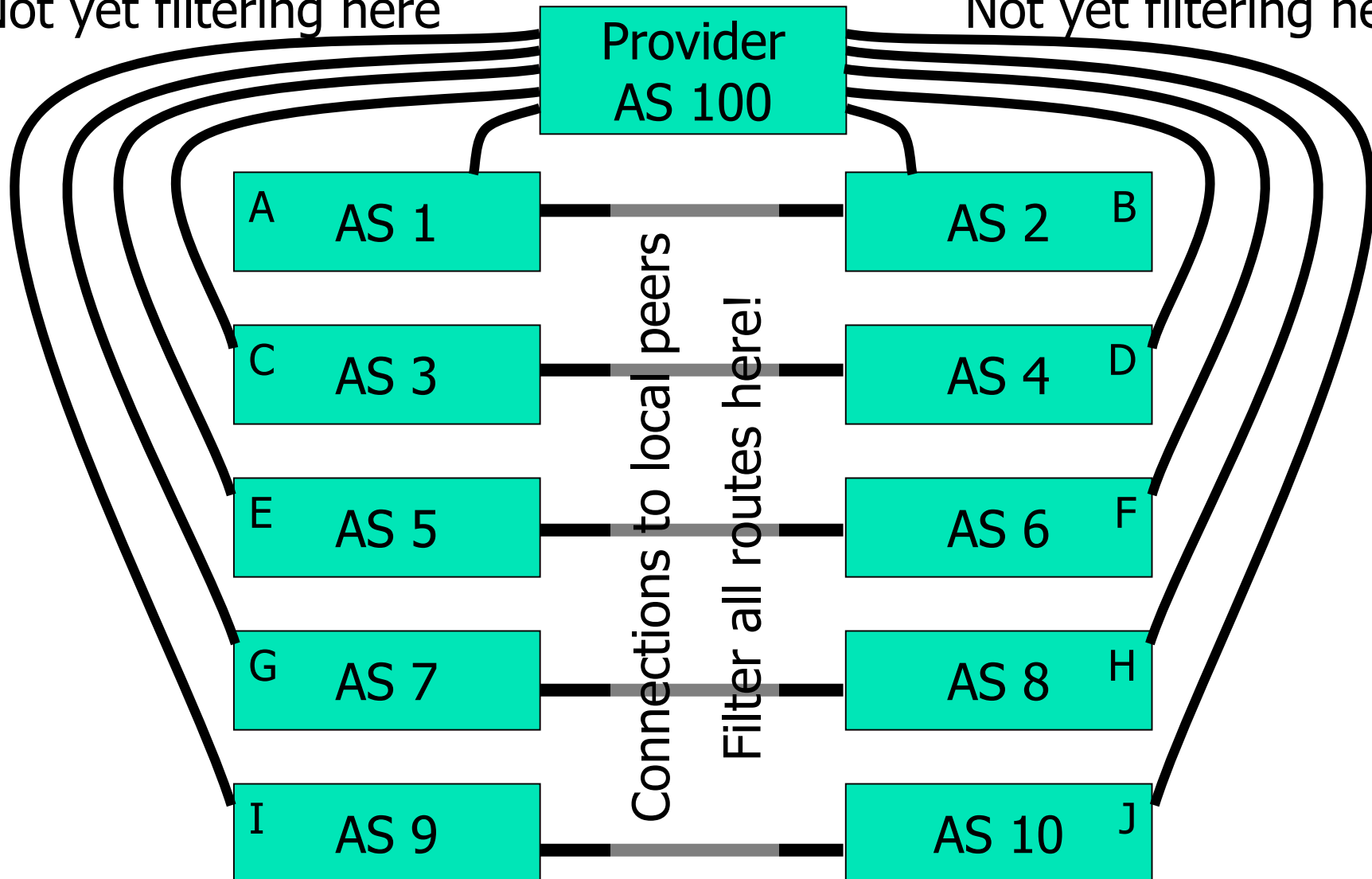
BGP Exercise 3

Filtering peer routes using AS-
path regular expression

Exercise 3: Filtering peer routes using AS-path

Transit to provider
Not yet filtering here

Transit to provider
Not yet filtering here





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\$
- Apply the filters
 - “neighbor <address> filter-list <number> in”

Exercise 3: What you should see

- From peers: only their routes, no transit
 - They send all routes, but you filter
- To peers: your routes and transit routes
 - They should ignore the transit routes
 - But it's bad that you send transit routes
- From upstream: all routes
- To upstream: all routes
 - This is bad



Exercise 3: Did it work?

- “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



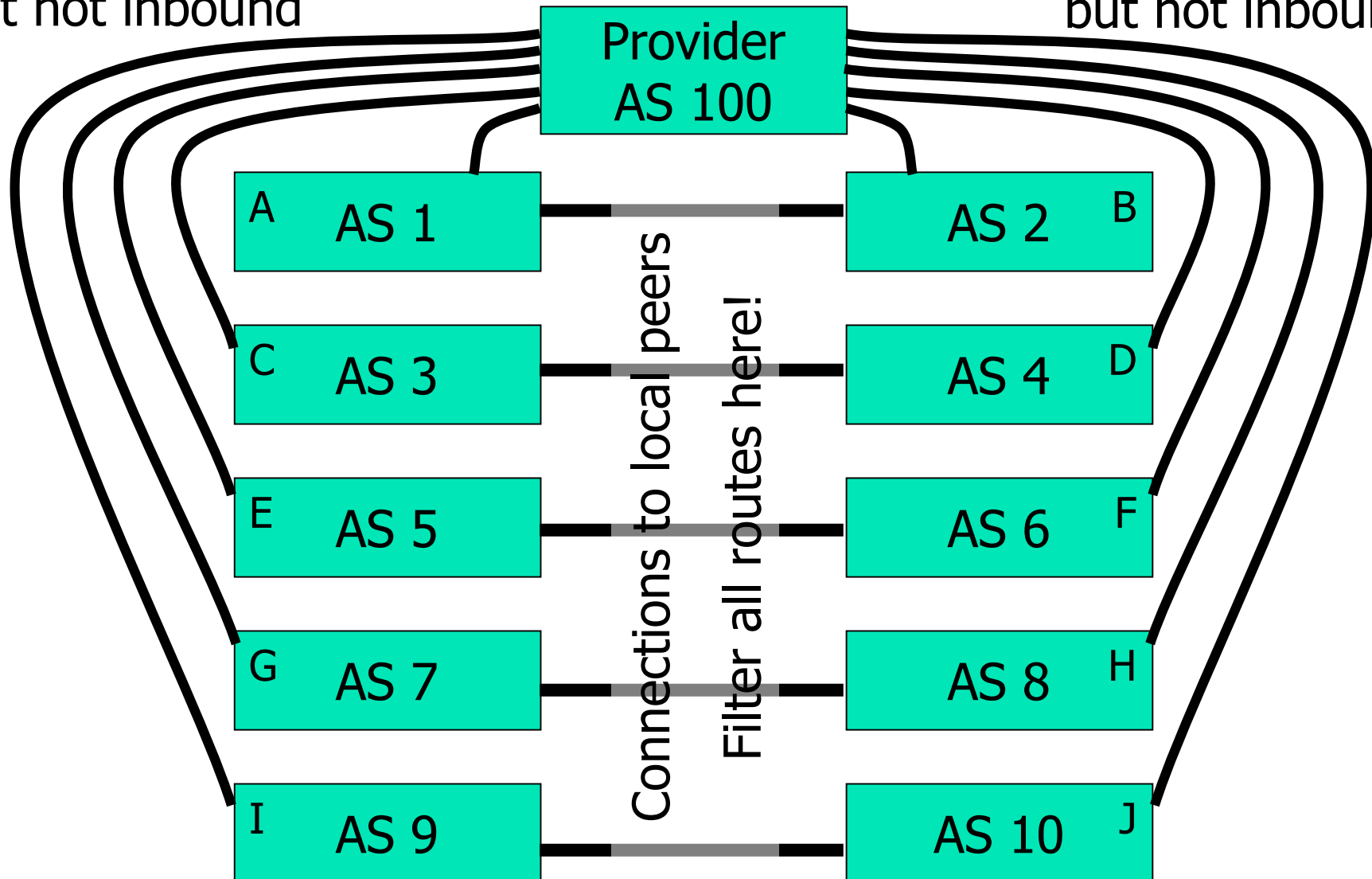
BGP Exercise 4

Filtering peer routes using prefix-
lists

Exercise 4: Filtering peer routes using prefix-lists

Filter outbound
but not inbound

Filter outbound
but not inbound





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

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" in cheat sheet



Exercise 4: Did it work?

- “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 xxx advertised-routes” - to neighbour, after filtering



BGP Part 8

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 Path Attributes...

- Origin
- AS-path
- Next-hop
- Multi-Exit Discriminator (MED)
- Local preference
- BGP Community
- Others...

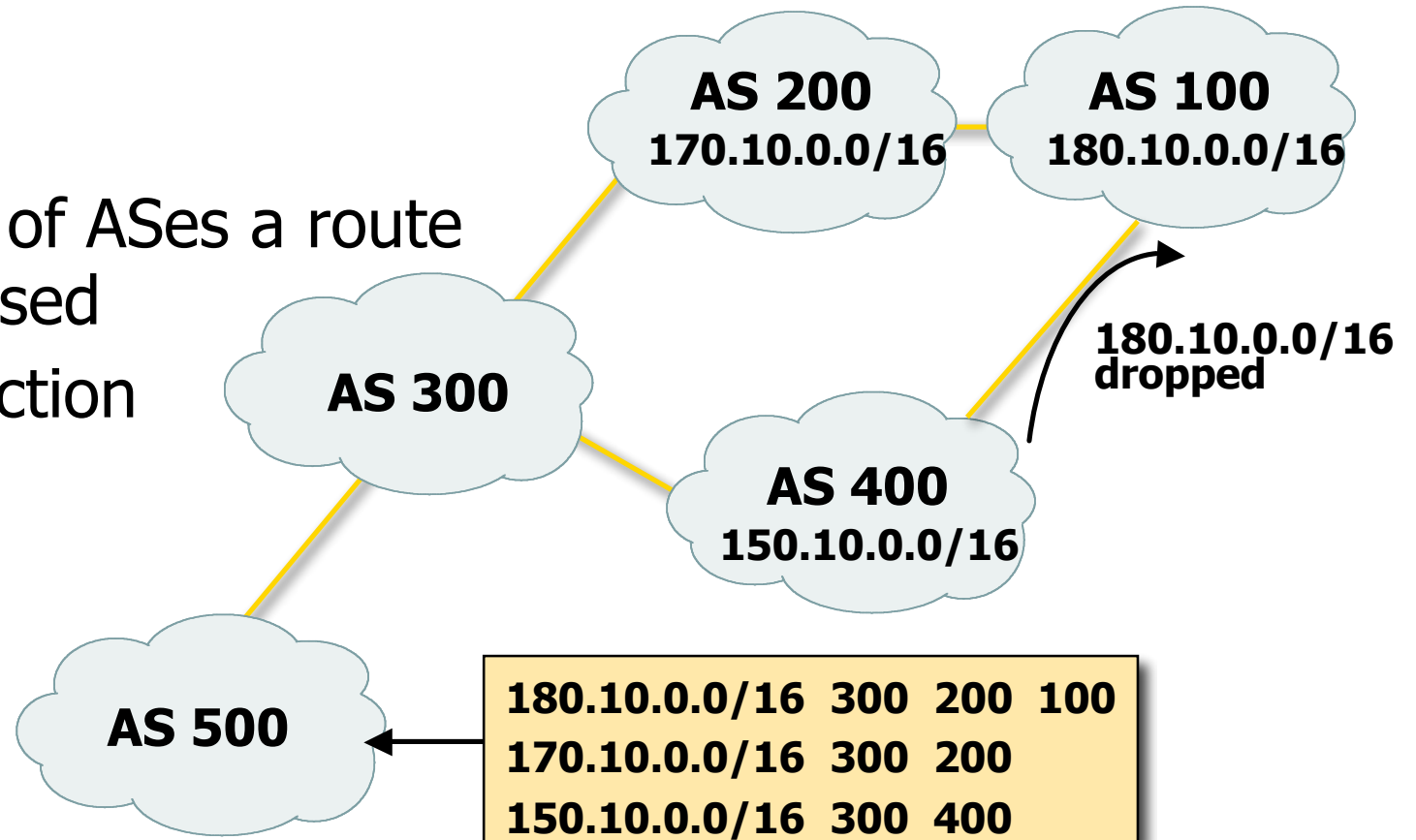


AS-PATH

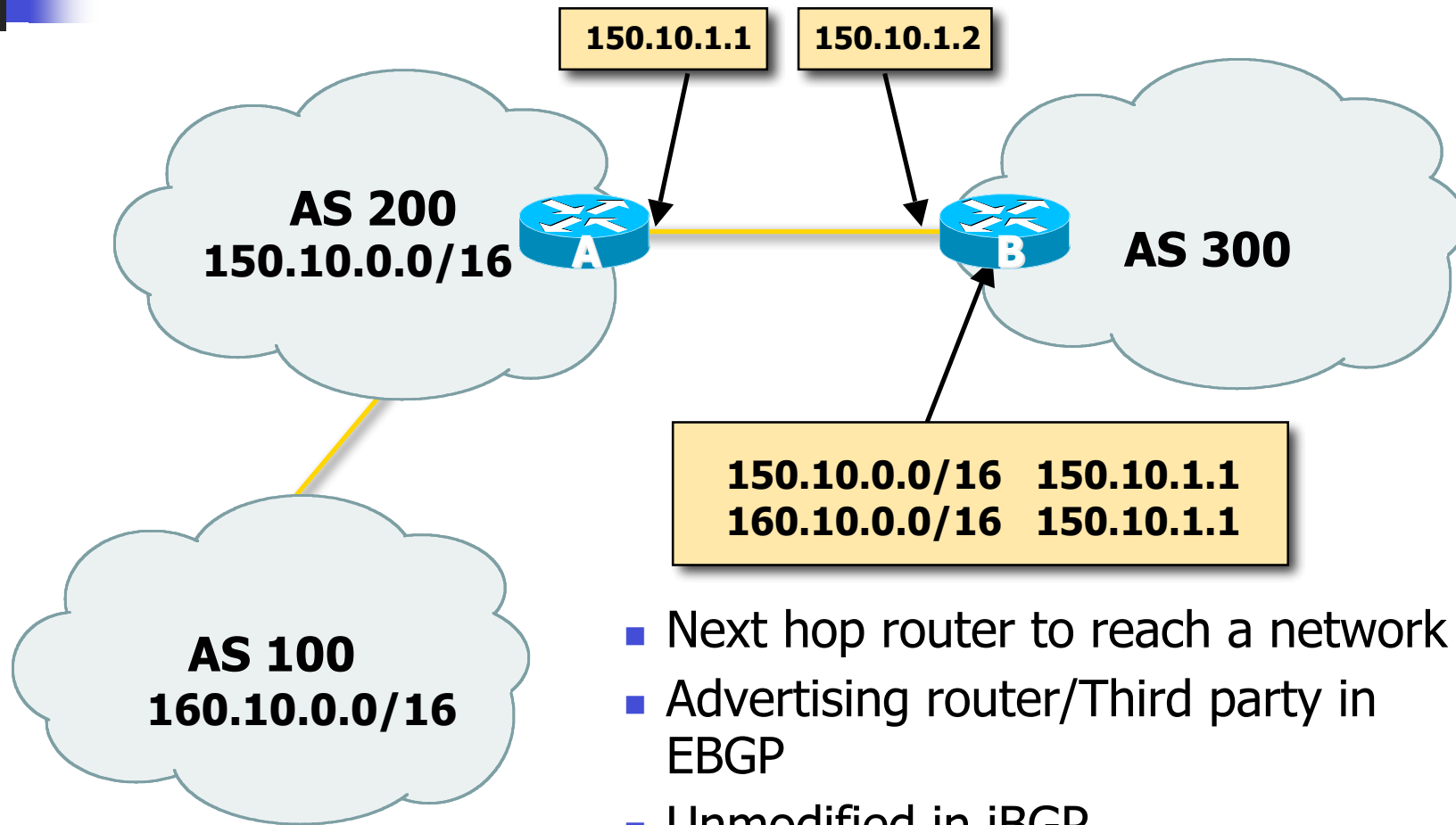
- Updated by the sending router with its AS number
- Contains the list of AS numbers the update traverses.
- Used to detect routing loops
 - Each time the router receives an update, if it finds its own AS number, it discards the update

AS-Path

- Sequence of ASes a route has traversed
- Loop detection



Next-Hop



- Next hop router to reach a network
- Advertising router/Third party in EGP
- Unmodified in iBGP



Next Hop...

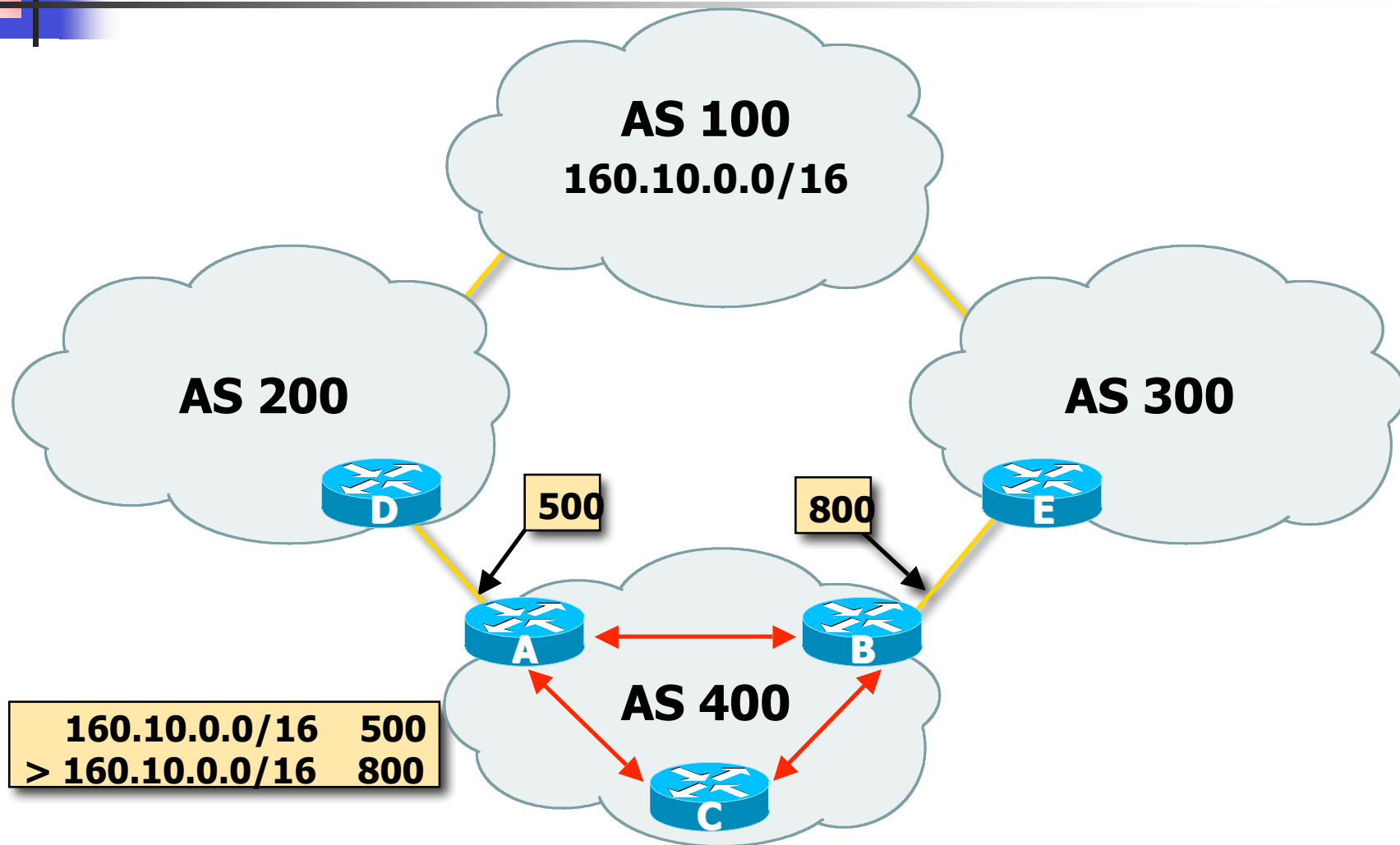
- IGP should carry route to next hops
- Recursive route look-up
- Unlinks BGP from actual physical topology
- Allows IGP to make intelligent forwarding decision



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

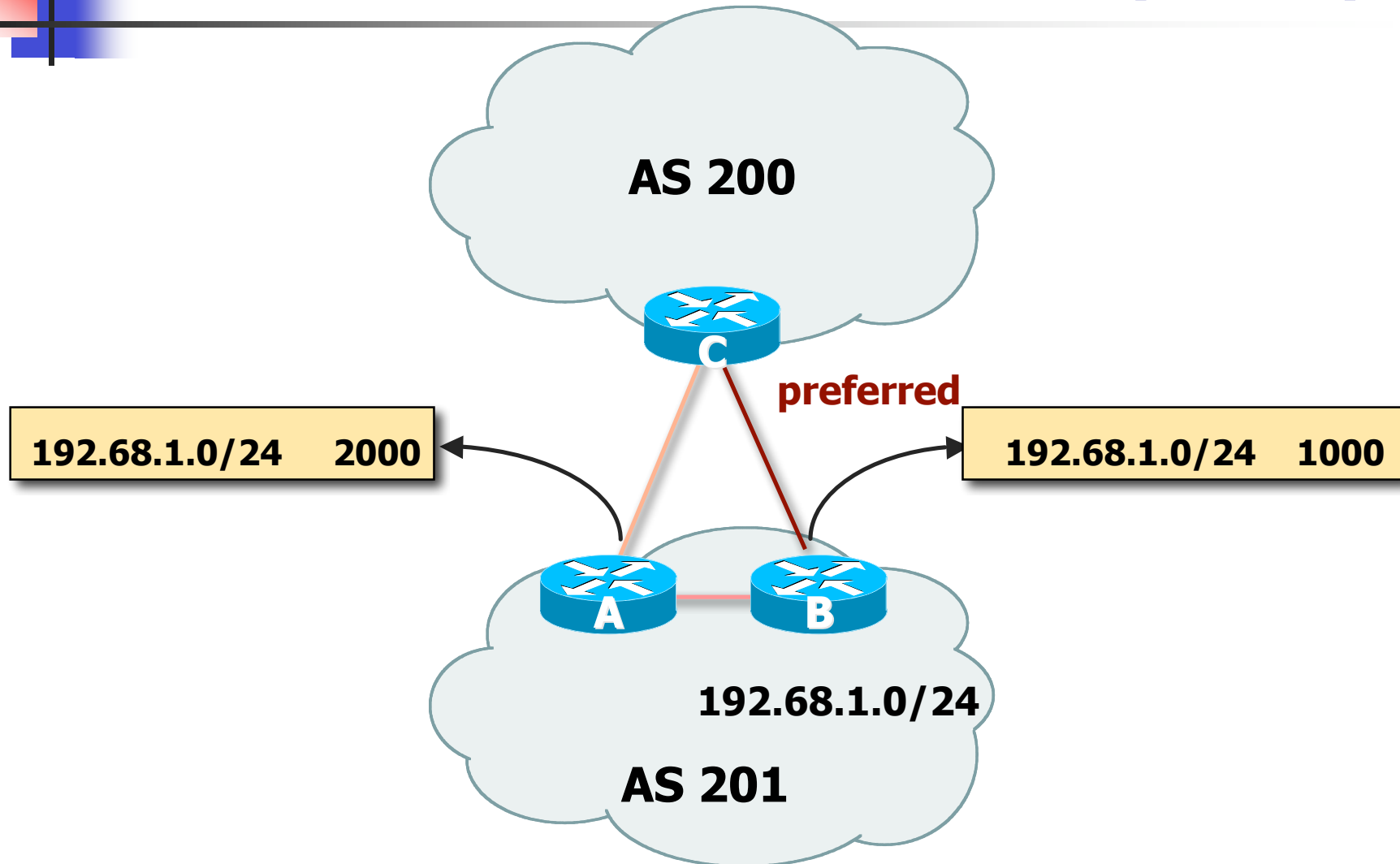




Multi-Exit Discriminator

- 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

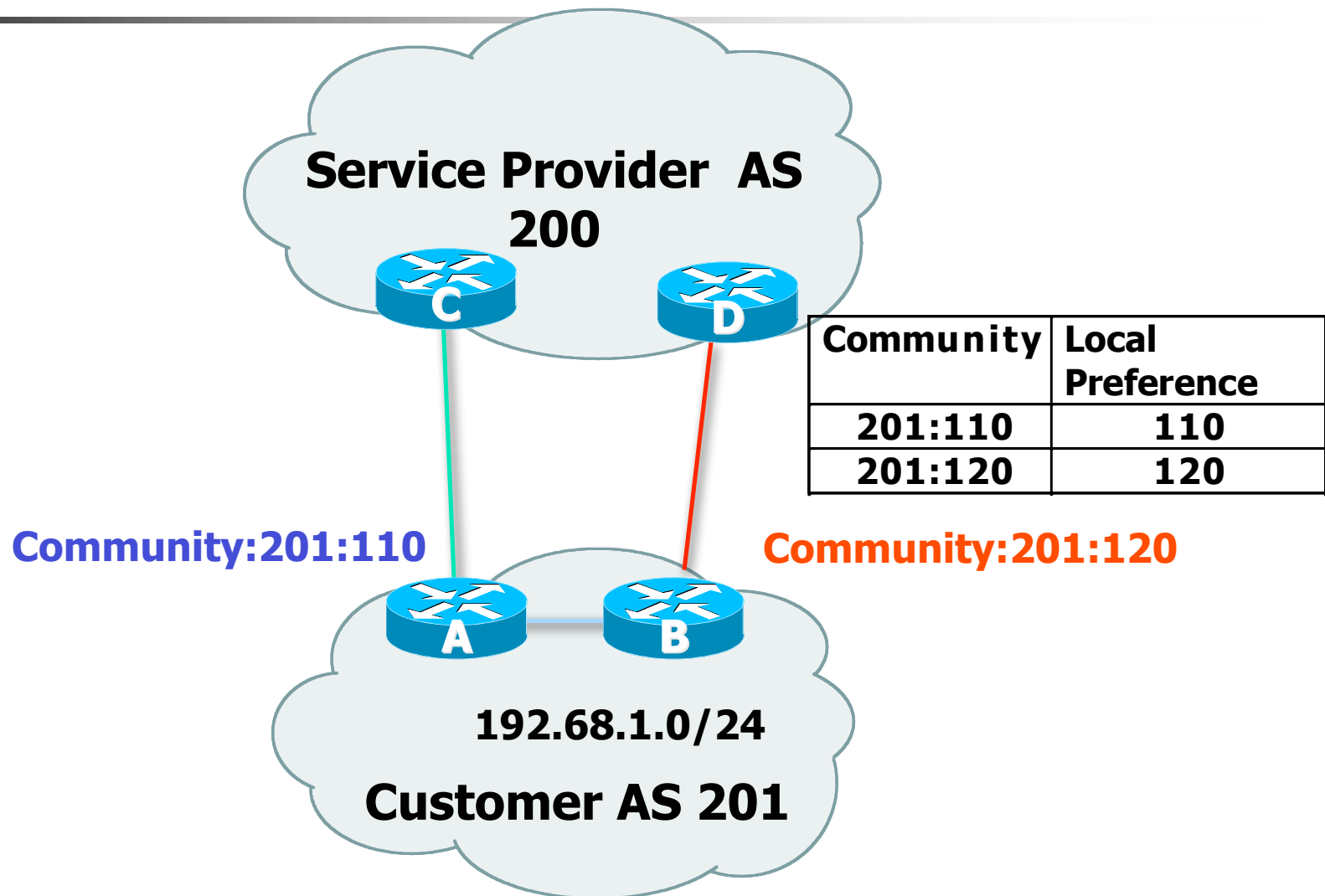
- Conveys the origin of the prefix
- Three values:
 - IGP – from BGP network statement
 - E.g. – *network 35.0.0.0*
 - EGP – redistributed from EGP (not used today)
 - Incomplete – redistributed from another routing protocol
 - E.g. – *redistribute static*
- IGP < EGP < incomplete
 - Lowest origin code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



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





Weight

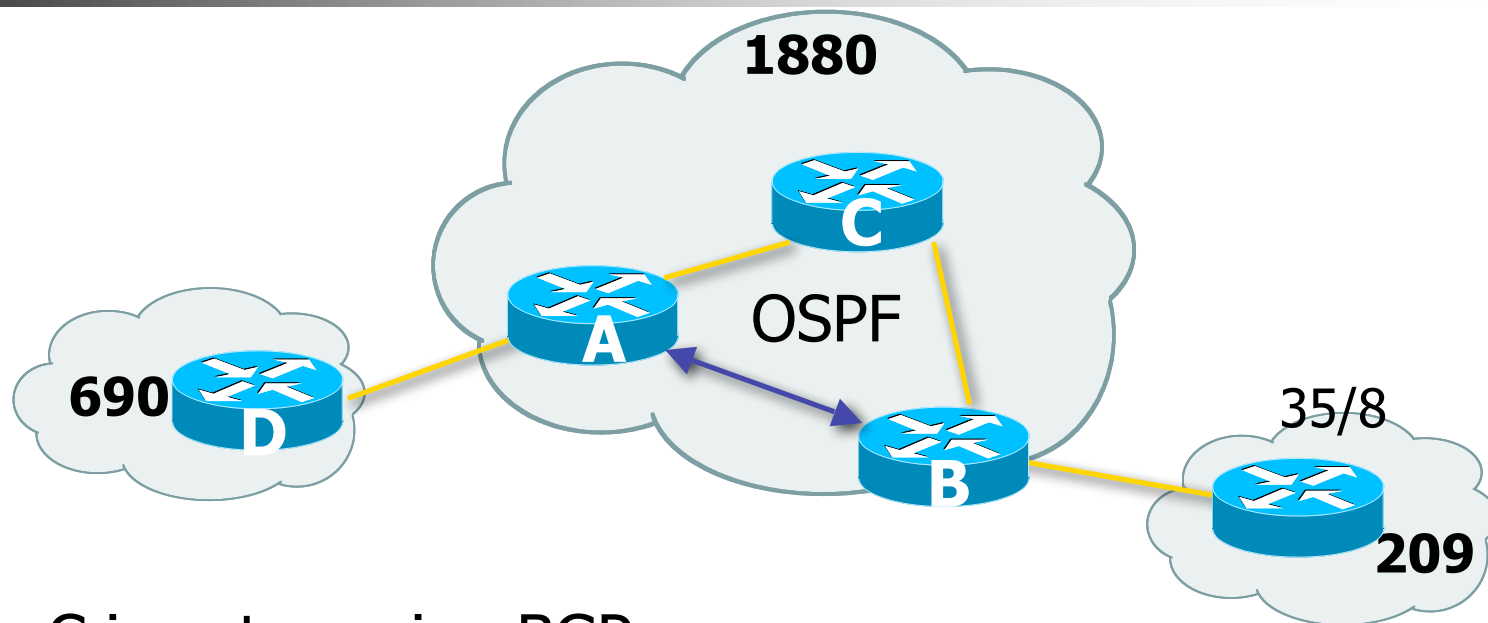
- Special Cisco attribute used when there is more than one route to same destination.
- Local to the router on which it is assigned, and not propagated in routing updates.
- Default is 32768 for paths that the router originates and zero for other paths.
- Routes with a higher weight are preferred when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



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
 - This is a default which is 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)

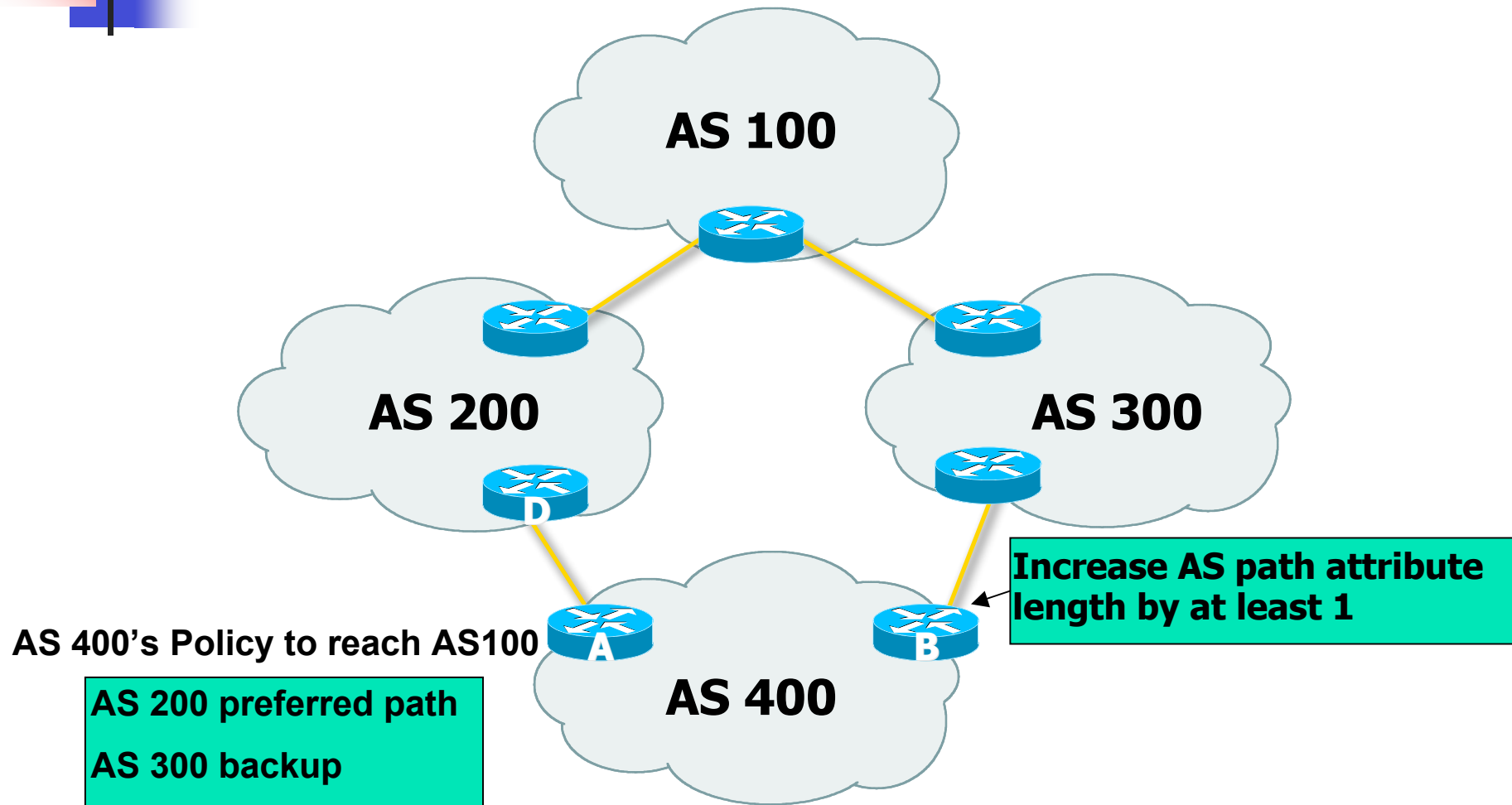
- Locally sourced
 - Via redistribute or network statement
- Shortest AS path length
 - Number of ASes in the AS-PATH attribute
- Lowest origin
 - IGP < EGP < incomplete
- Lowest MED
 - Compared from paths from the same AS



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

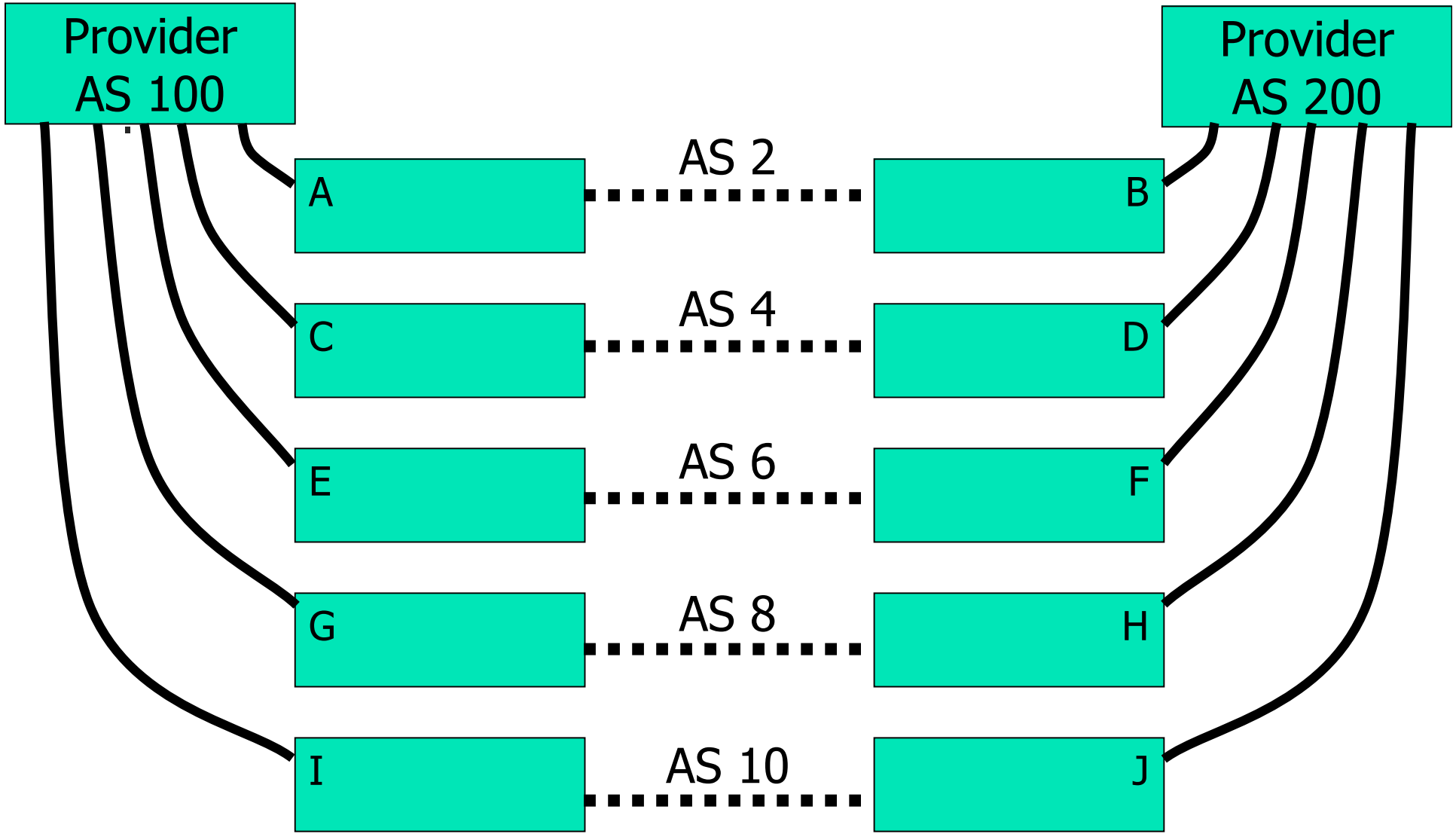
Interior BGP (iBGP)



Exercise 5: Configure iBGP

- Tables join into pairs, with two routers per AS
- Each AS has two upstream providers
- OSPF 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
 - 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 9

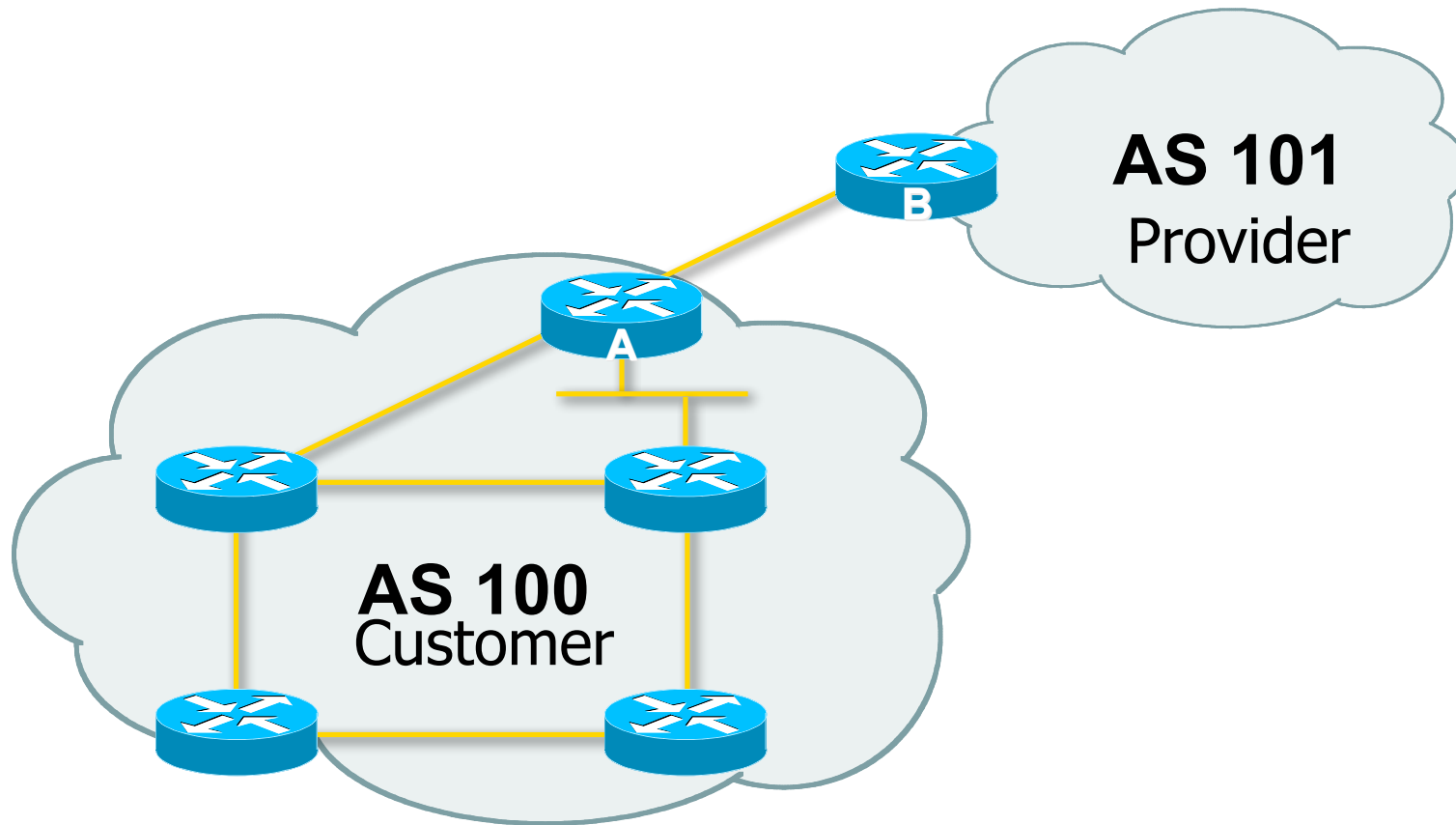
BGP and Network Design



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

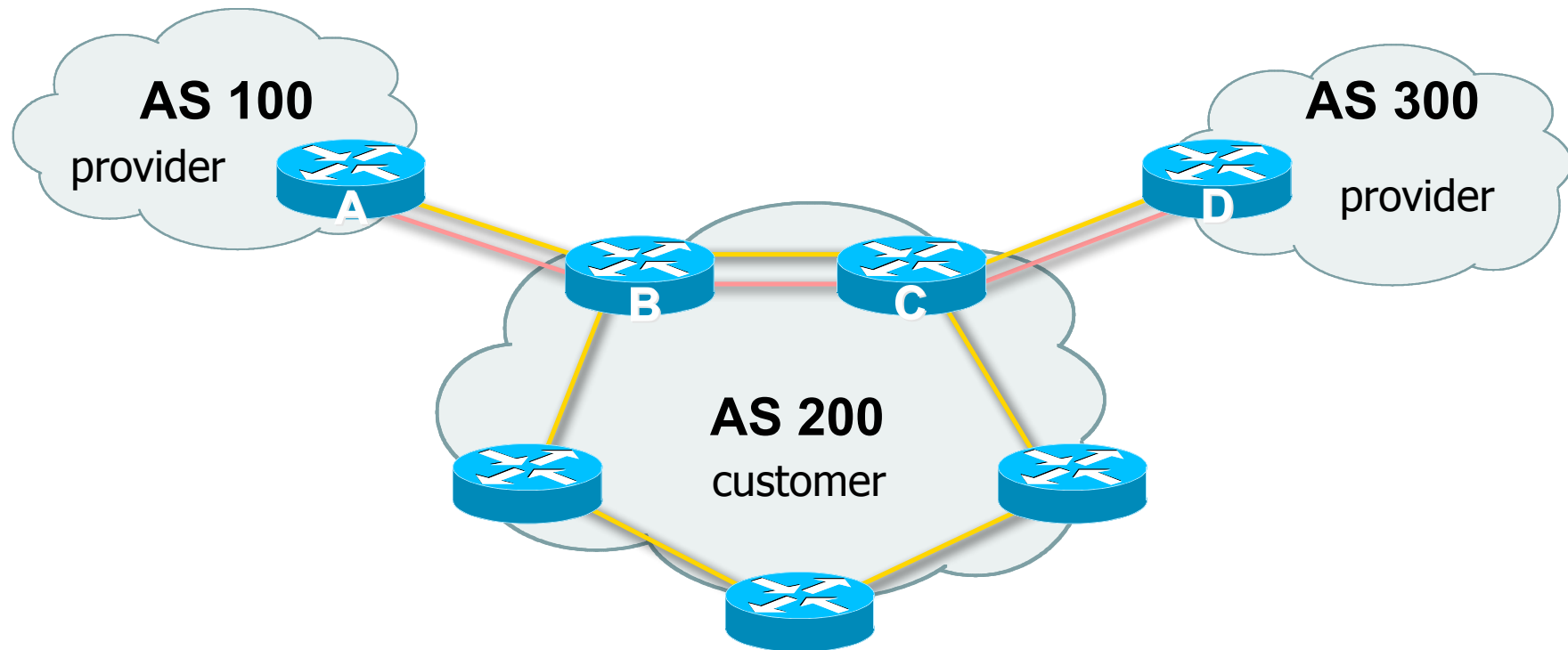




Multi-homed AS

- 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



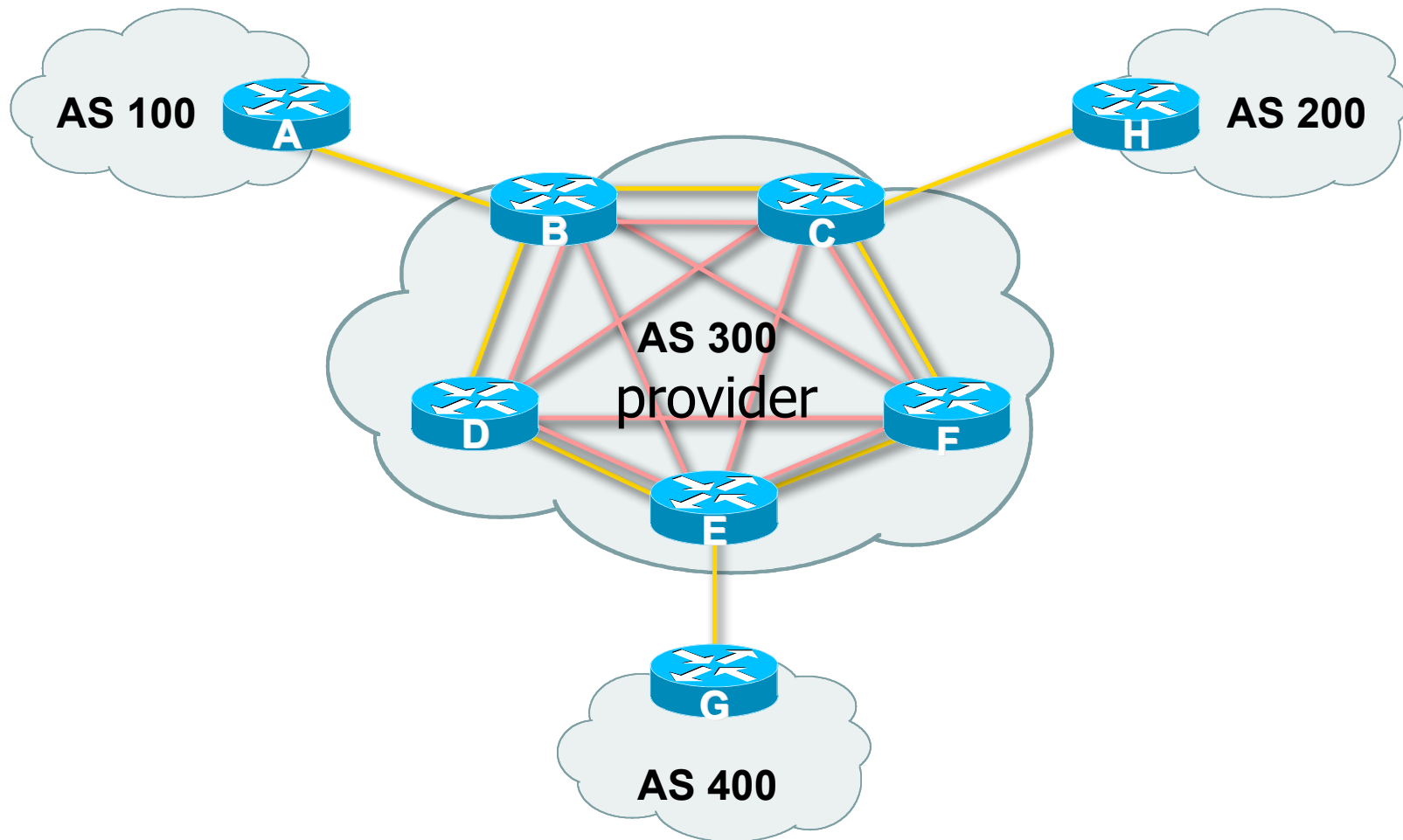
- More details on multihoming coming up...



Service Provider Network

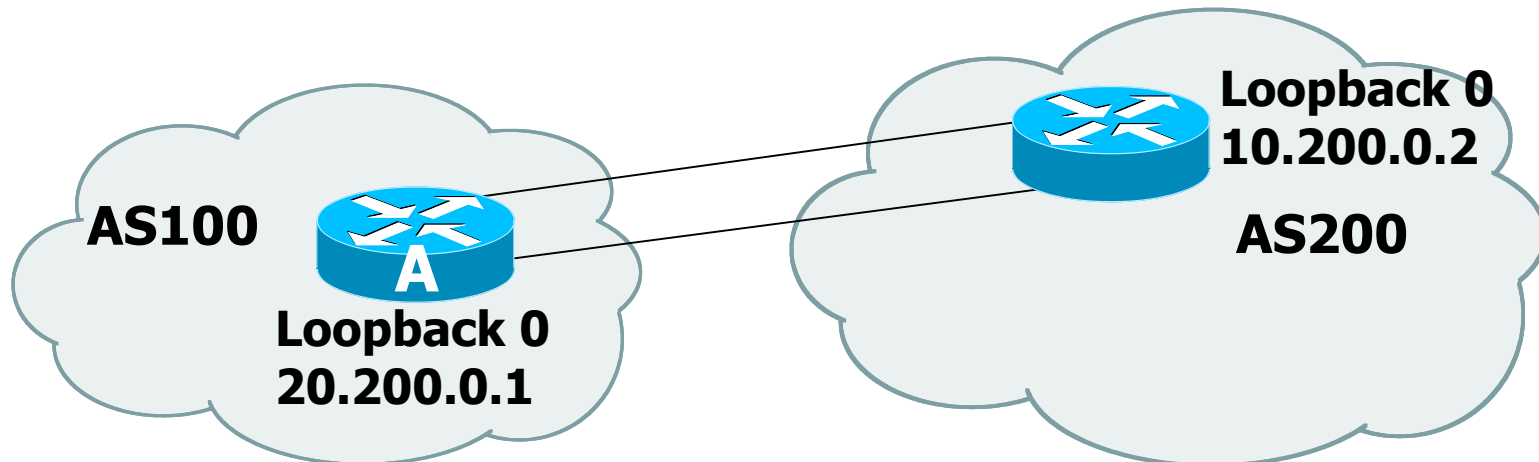
- iBGP used to carry exterior routes
 - No redistribution into IGP
- IGP used to track topology inside your network
- Full iBGP mesh required
 - Every router in ISP backbone should talk iBGP to every other router
 - This has scaling problems, and solutions (e.g. route reflectors)

Common Service Provider Network



Load-sharing – single path

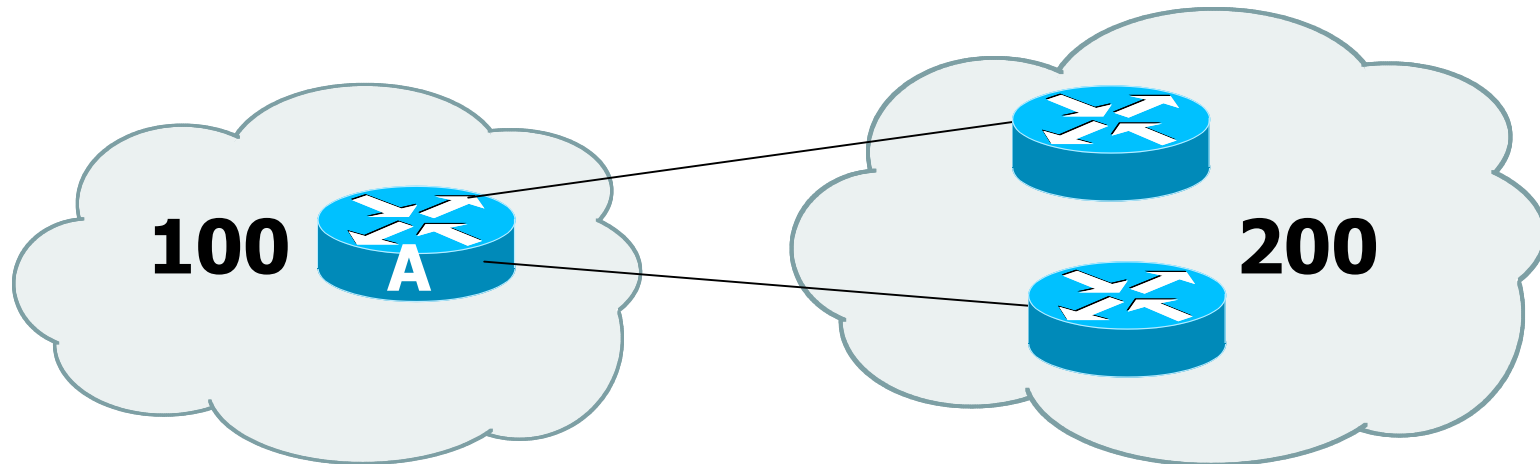
```
Router A:  
interface loopback 0  
  ip address 20.200.0.1 255.255.255.255  
!  
router bgp 100  
  neighbor 10.200.0.2 remote-as 200  
  neighbor 10.200.0.2 update-source loopback0  
  neighbor 10.200.0.2 ebgp-multihop 2  
!  
ip route 10.200.0.2 255.255.255.255 <DMZ-link1, link2>
```



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

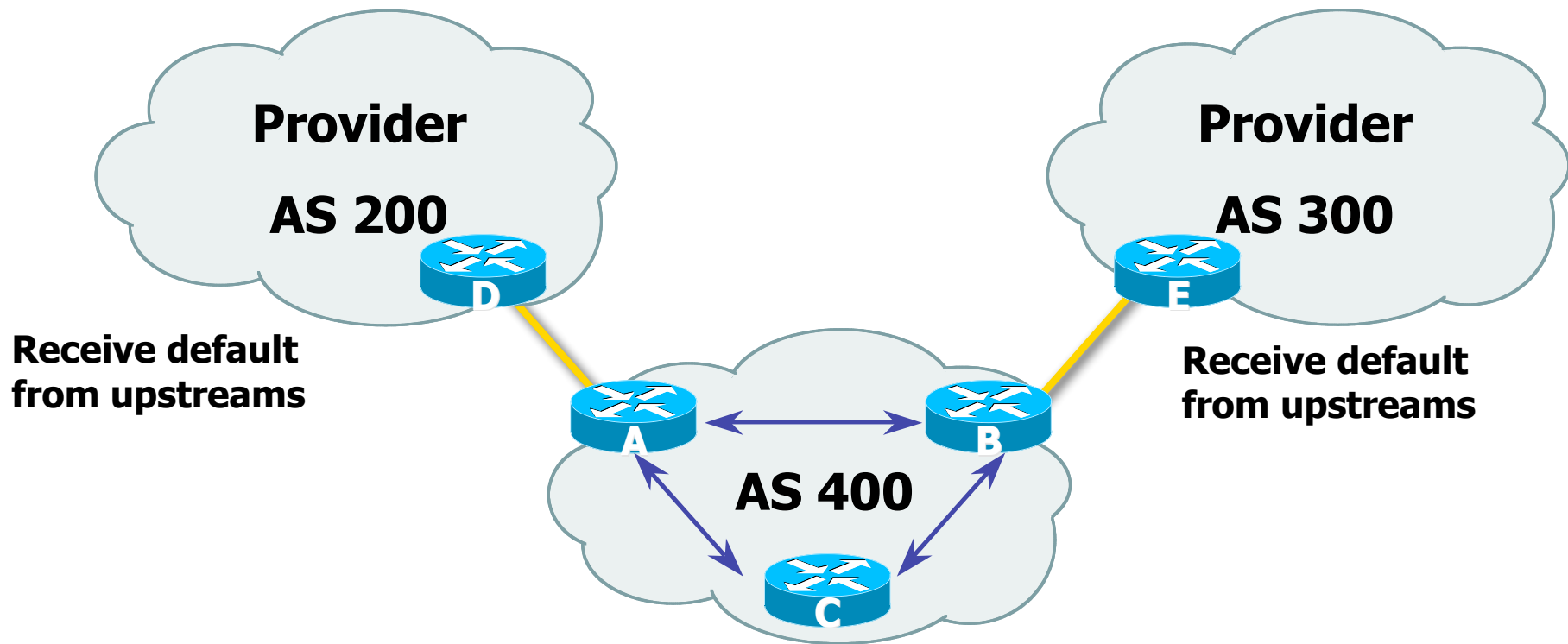
- Reliable connection to Internet
- 3 common cases of multi-homing
 - default from all providers
 - customer + default from all providers
 - full routes from all providers
- Address Space
 - comes from upstream providers, or
 - allocated directly from registries



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

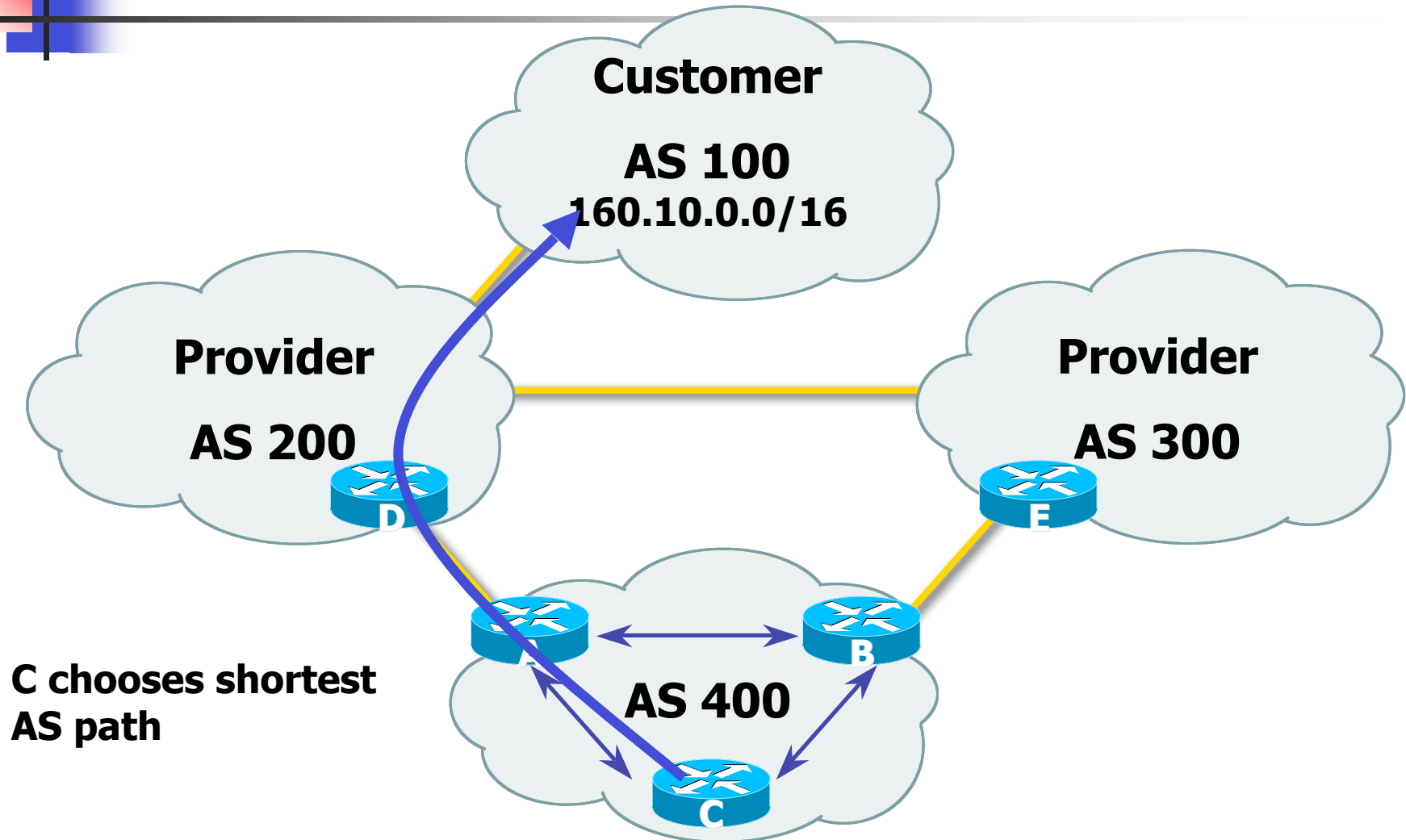




Customer prefixes plus default from all providers

- 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



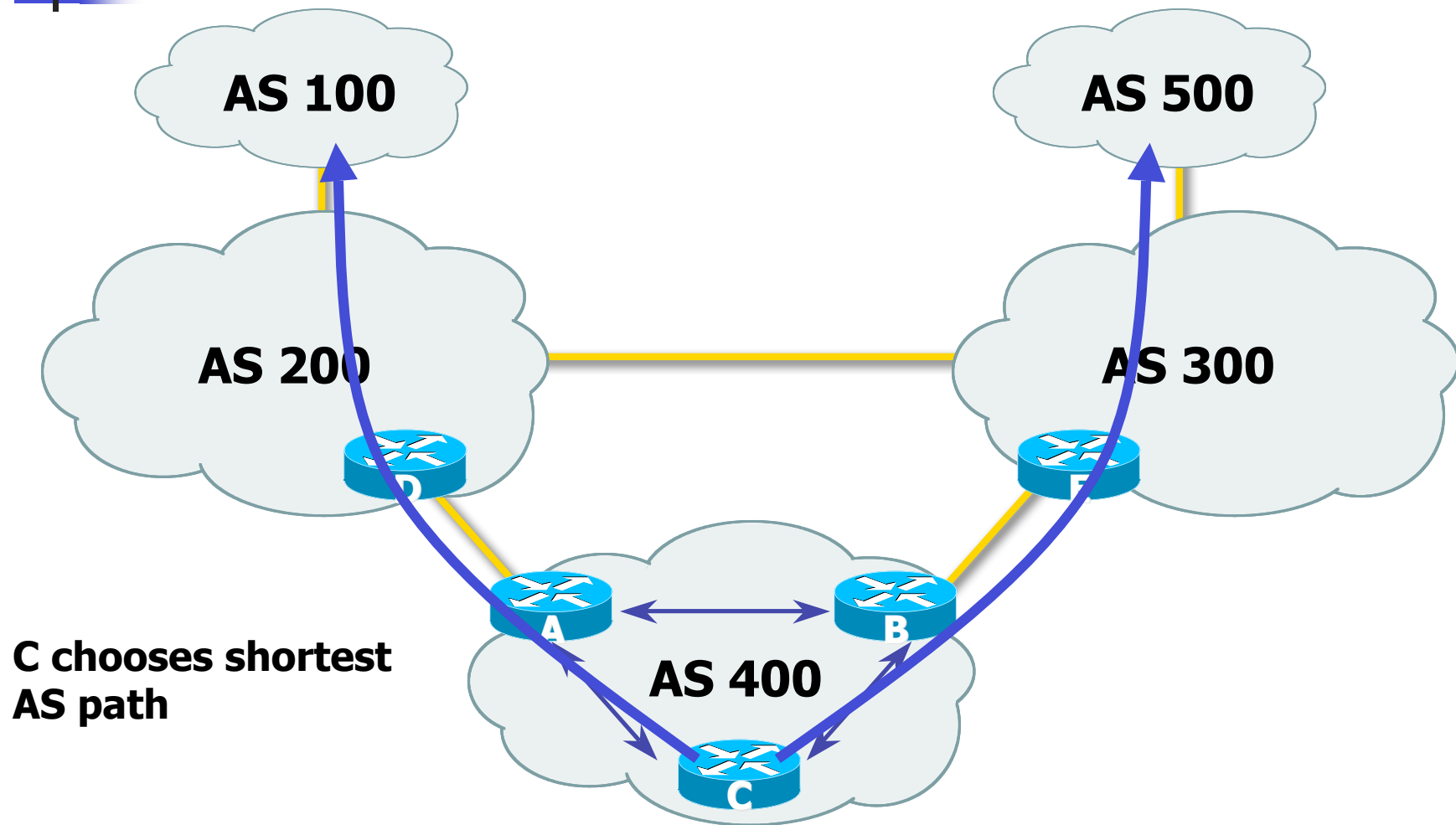
C chooses shortest AS path



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



**C chooses shortest
AS path**



Best Practices

IGP in Backbone

- IGP connects your backbone together, not your clients' routes
 - Your clients' routes 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...

Connecting to a customer

- Static routes
 - You control directly
 - No route flaps
- Shared routing protocol or leaking
 - You must filter your customers info
 - Route flaps
 - Strongly discouraged
- 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!!**
- **FILTER! FILTER! FILTER!**



Best Practices...

The Internet Exchange

- Long distance connectivity is expensive
- Connect to several providers at a single point



Summary

- BGP Building Blocks
- BGP Protocol Basics
- BGP Path Attributes
- BGP Path Computation
- Typical BGP topologies
- Routing Policy
- Redundancy/Load sharing
- Best current practices