E6998-02: Internet Routing

Lectures 15-20
Interdomain Routing

John Ioannidis

AT&T Labs – Research

ji+ir@cs.columbia.edu

Announcements

- 10/27: Homework 3 is out. Due 11/13 at 3am. No questions answered 48 hours before the homework is due.
- Midterm answers are on the course web page.
- No class on 11/4.

The old days

- Original Arpanet.
  - Single routing domain (GGP, then SPF).
  - Every gateway (router) knew all destinations.
  - Not all that many destinations back then!
- RFC827:
  - Scaling issues identified.
    - High algorithm overhead (given the hardware).
    - Stability.
  - Software engineering issues identified.
    - Different implementations.
    - Different default parameters.
  - Administrative issues.
    - Multiple network administrators.

RFC827: EGP

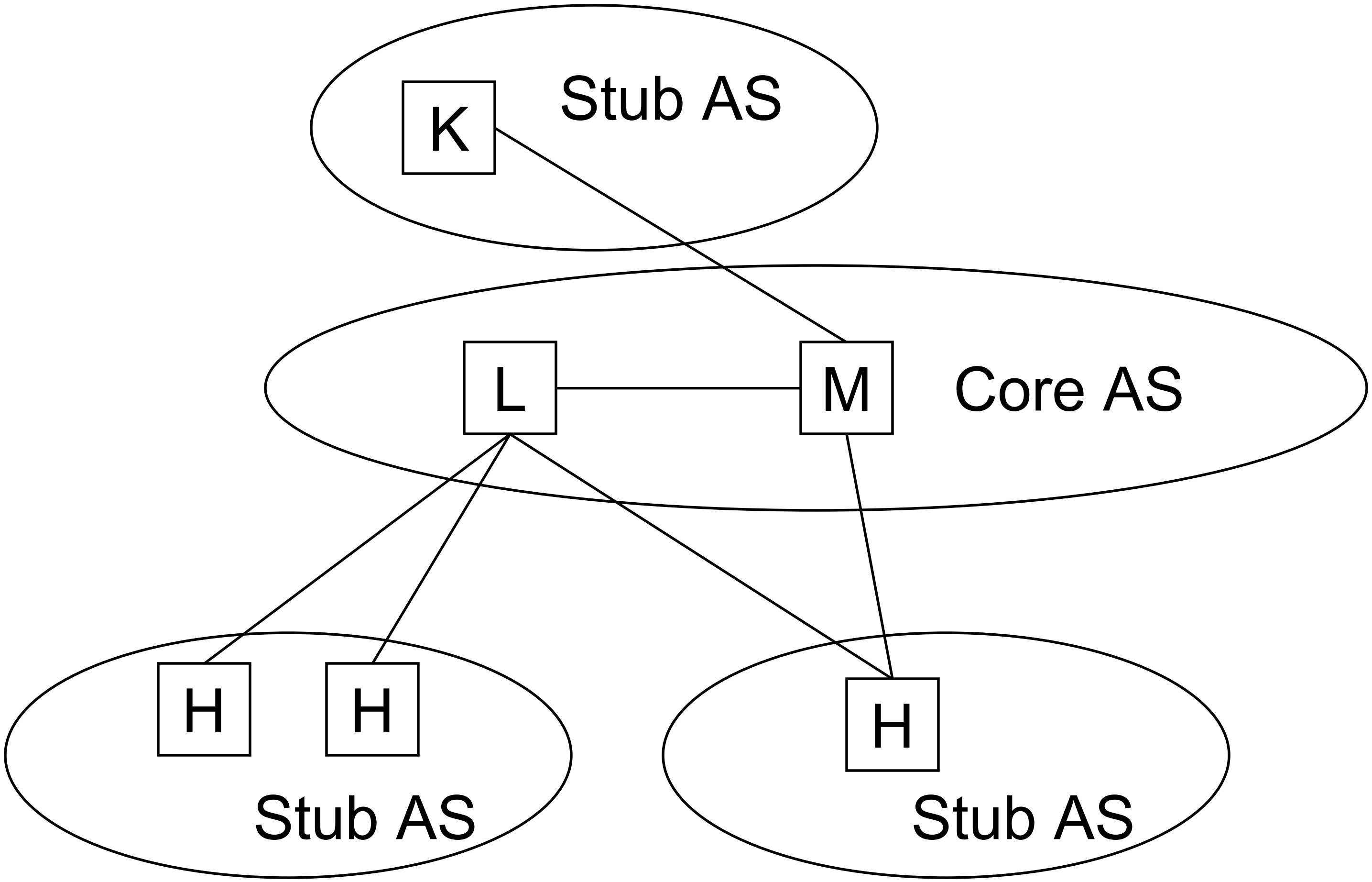
- Replace single routing domain with…
- Multiple interconnected autonomous routing domains.
  - Called "Autonomous Systems" (AS).
- Each AS managed independently.
- Identified by a 16 bit number (ASN).
  - ASN1: BBN, ASN14: Columbia, …
  - 64512 – 65535 (FC00-FFFF) are private.
- ASes run IGPs for their internal routing.
- ASes communicate using an EGP (of which "EGP" is the first one).
- IGPs are concerned with optimizing paths.
- EGPs are concerned with adhering to policy.
  - Different metrics make optimization an ill-defined problem.

Exterior Gateway Protocol

- RFC 827, 888, 904.
- IP Protocol 8
- *Neighbors* (or *peers*): routers exchanging EGP messages.
  - *Interior neighbors*: in the same AS.
  - *Exterior neighbors*: in different ASes.
- All EGP routers accept messages about other ASes.
- *Stub gateways* send messages only about their own AS.
- *Core gateways* send messages about all ASes.

EGP topology

- One Core AS to which Stub ASes connect.
- Avoids loops.



EGP Neighbor Acquisition/Reachability

- Neighbor addresses manually configured.
- There is an *active* and a *passive* neighbor.
- *Neighbor Acquisition Request* unicast to neighbor.
    - *Hello interval* and *Poll interval* specified.
- *Neighbor Acquisition Confirm* and *Refuse*.
- *Neighbor Cease* / *Neighbor Cease Ack*.
- Relationship maintained with periodic *Hello*/*I-Heard-You* messages.

- Nothing surprising here!

EGP Network Reachability Protocol

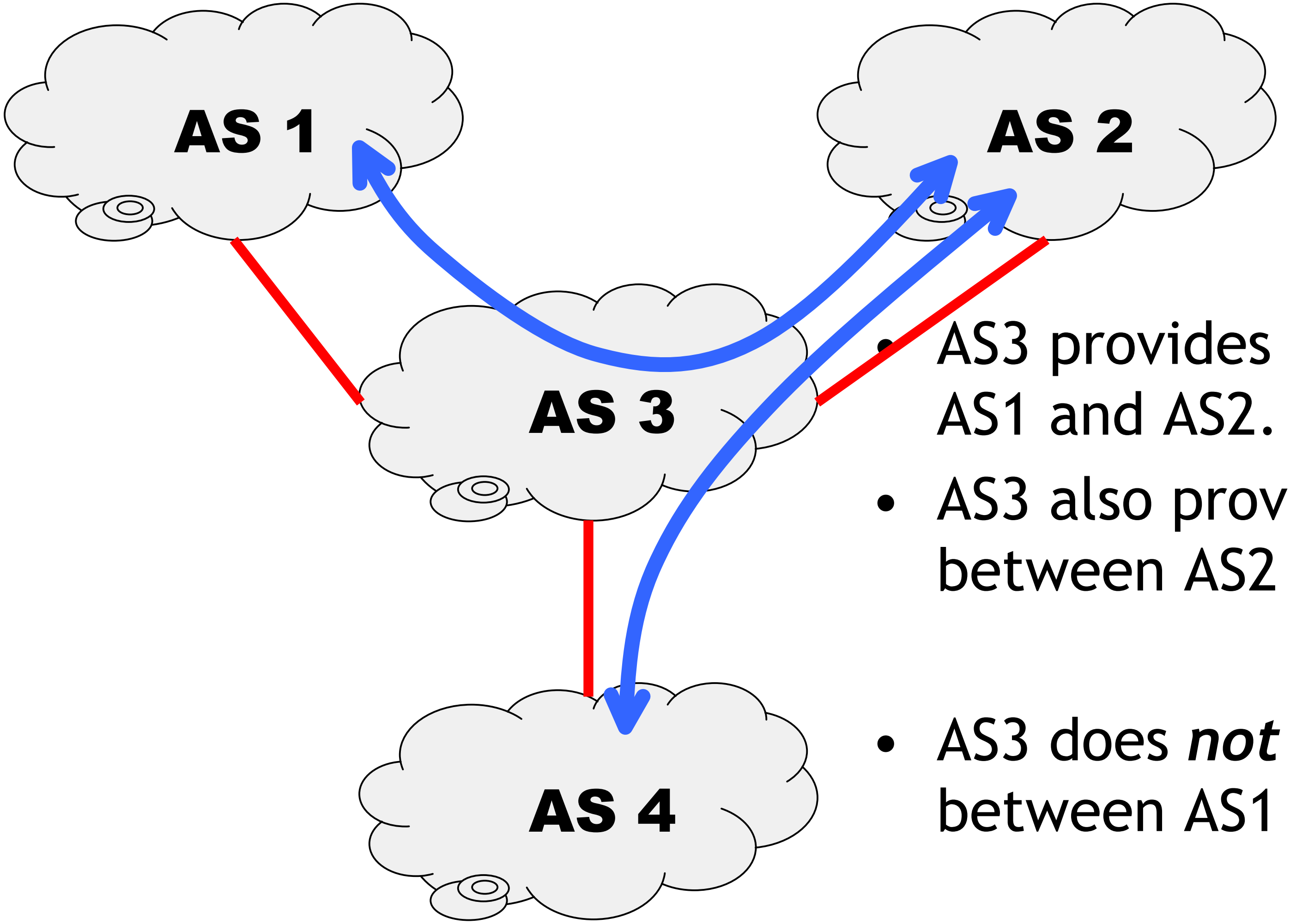
- One neighbor sends a *Poll* message
  - Contains a sequence number.
- The other responds with an *Update* message.
  - Echoes the s/n.
  - Includes list of reachable networks.
- Hello/IHU messages contain the same s/n until an update is received.
  - S/N is then incremented.
- Unsolicited updates are an option.
- Notion of indirect (proxy) updates.
  - Route server.
- Details are not important.

Limitations of EGP

- Inability to detect routing loops.
  - Metrics don’t really mean much.
  - Count-to-infinity too slow.
- Must be engineered loop-free.
- Policy was kludged when NSFNET dictated AUPs.
- Little interaction with IGP to pick best routes.
- Very slow to advertise topology changes.
- Classful.

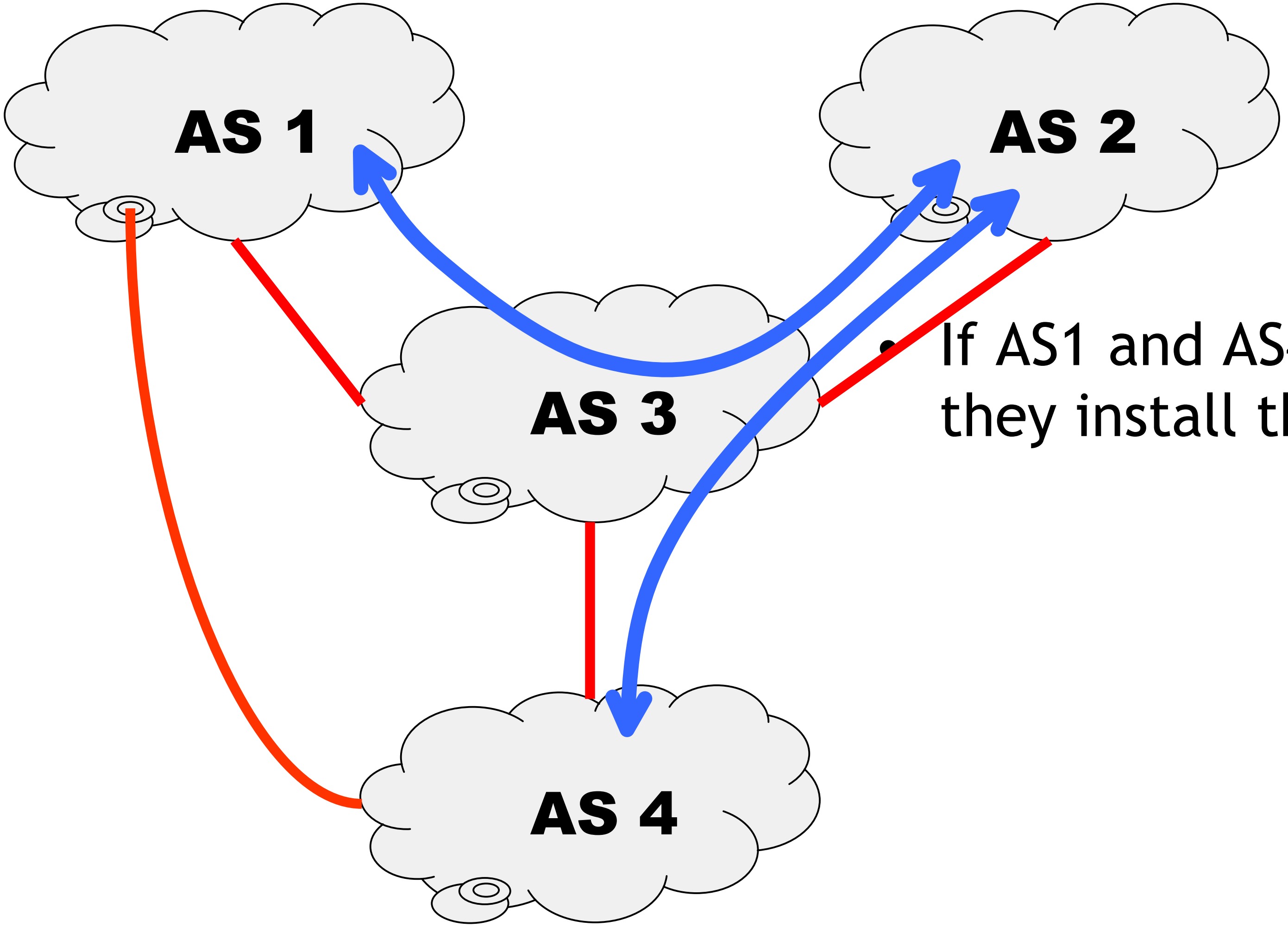
- Abandoned in favor of BGP(-1, -2, -3, -4).

Transit *vs.* Non-transit Networks (review)



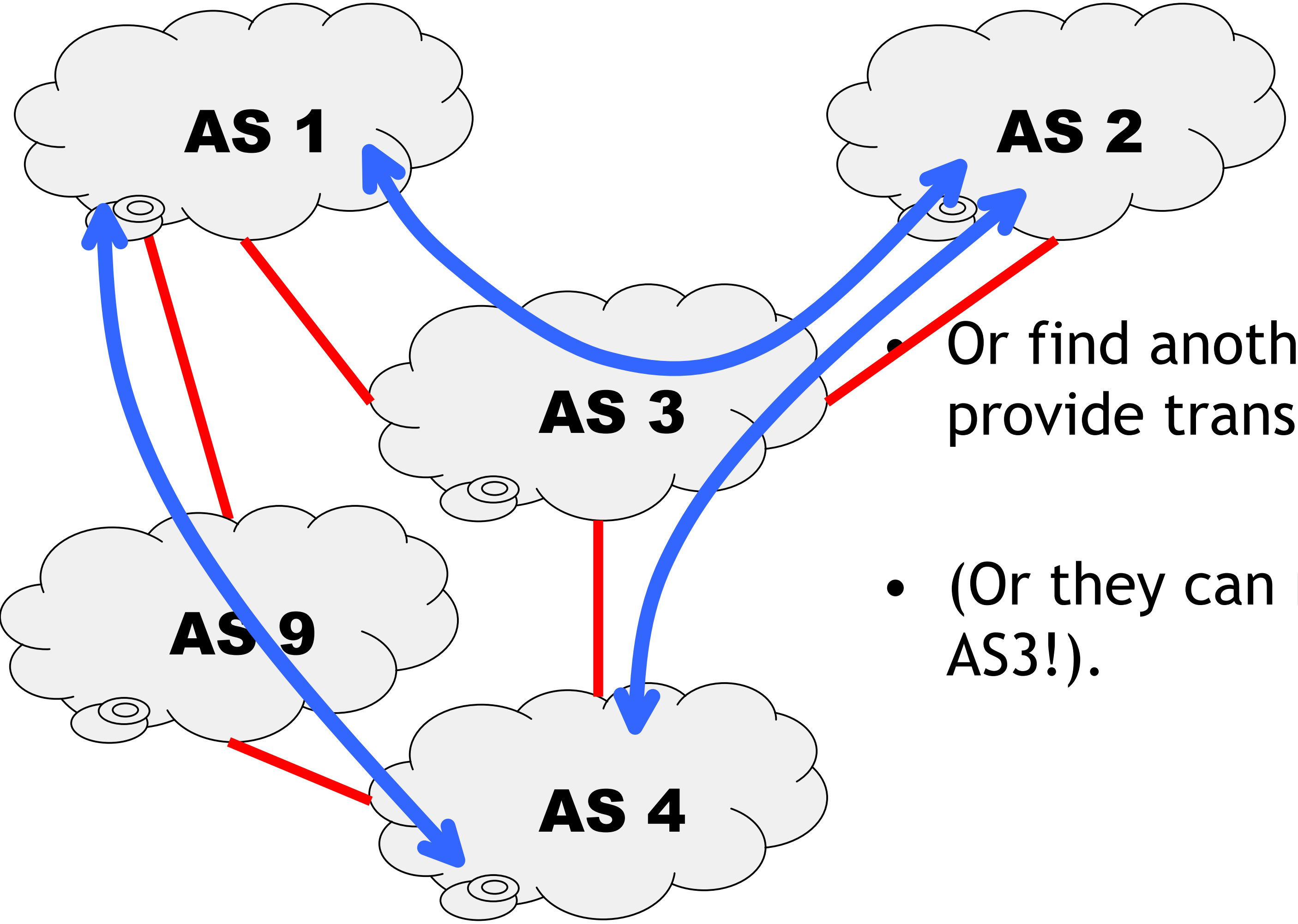
- AS3 provides transit between AS1 and AS2.
- AS3 also provides transit between AS2 and AS4.
- AS3 does ***not*** provide transit between AS1 and AS4.

Transit *vs.* Non-transit Networks (review)



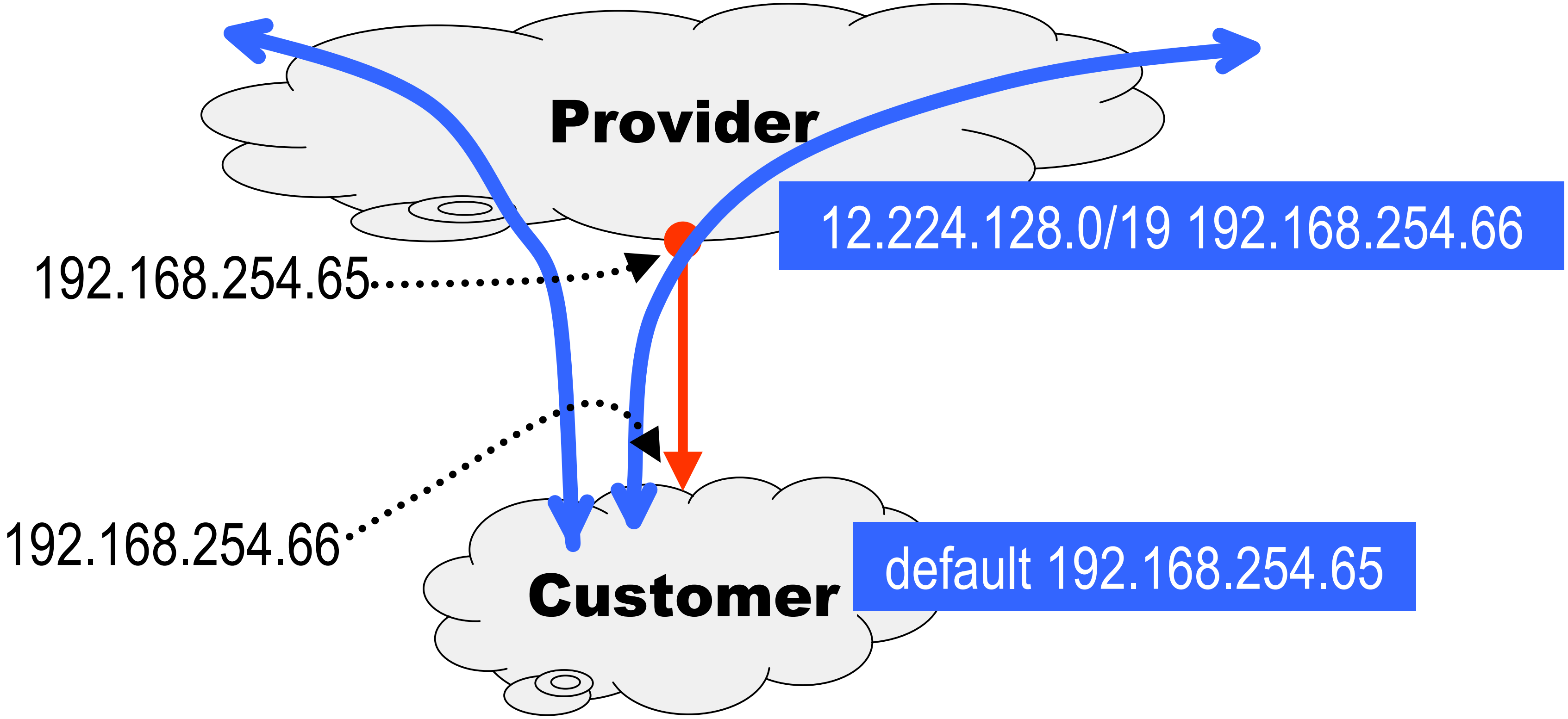
- If AS1 and AS4 need to talk, they install their own link.

Transit *vs.* Non-transit Networks (review)



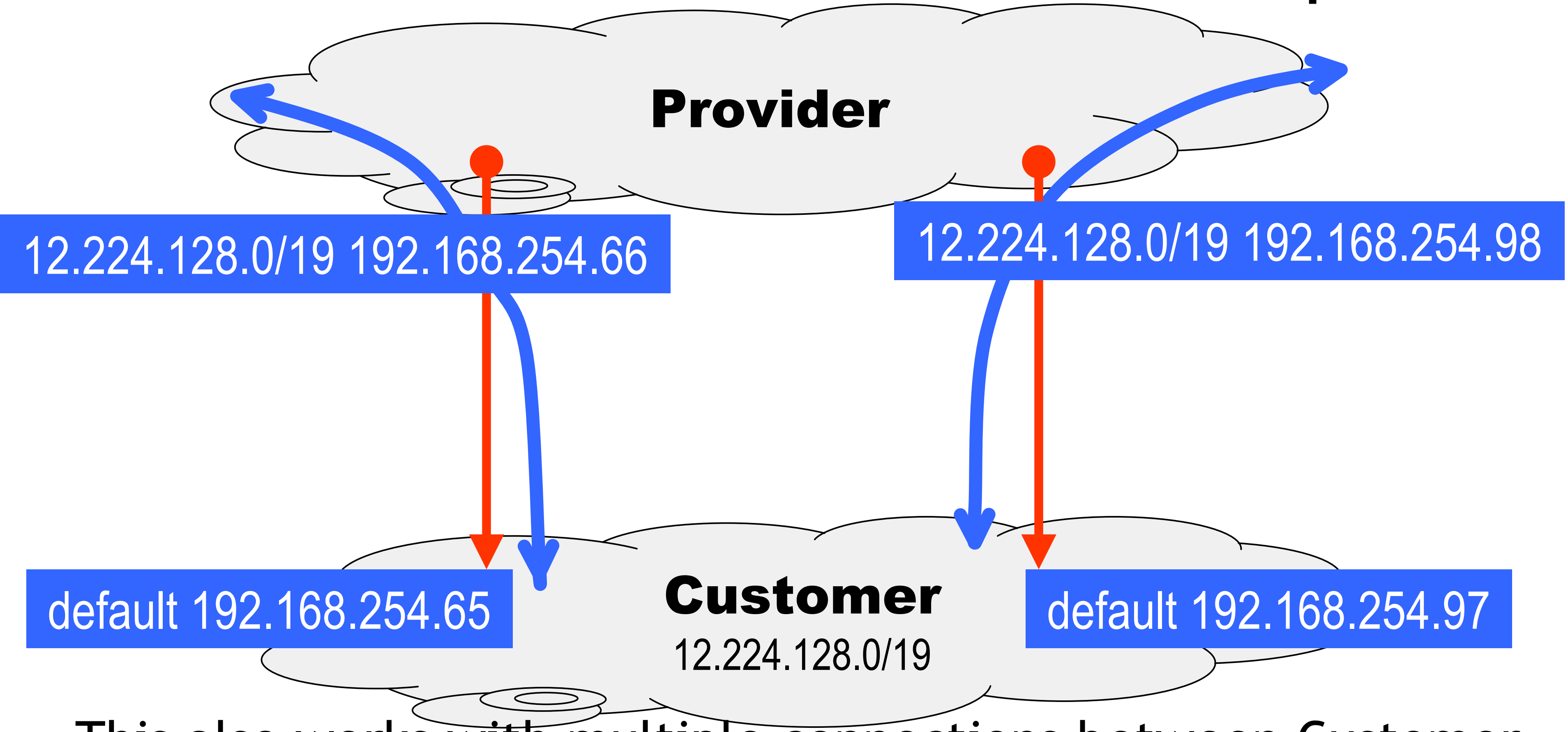
- Or find another network to provide transit traffic.
- (Or they can negotiate with AS3!).

Customer-Provider Relationship



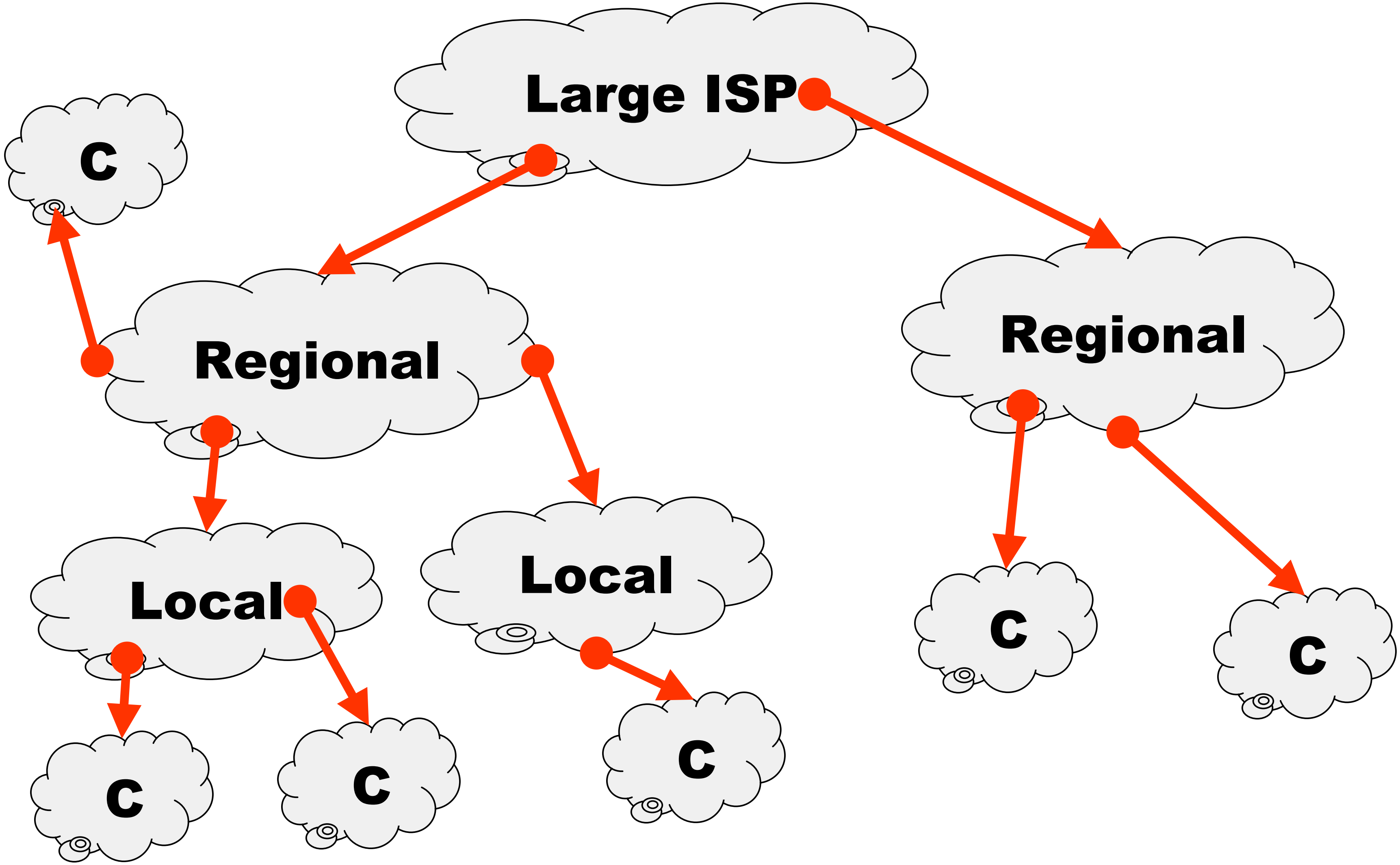
- Customer pays provider for access.
- Customer just has default route pointing to provider.
- Provider has static route pointing to customer.
- Customer does not need BGP.

Customer-Provider Relationship



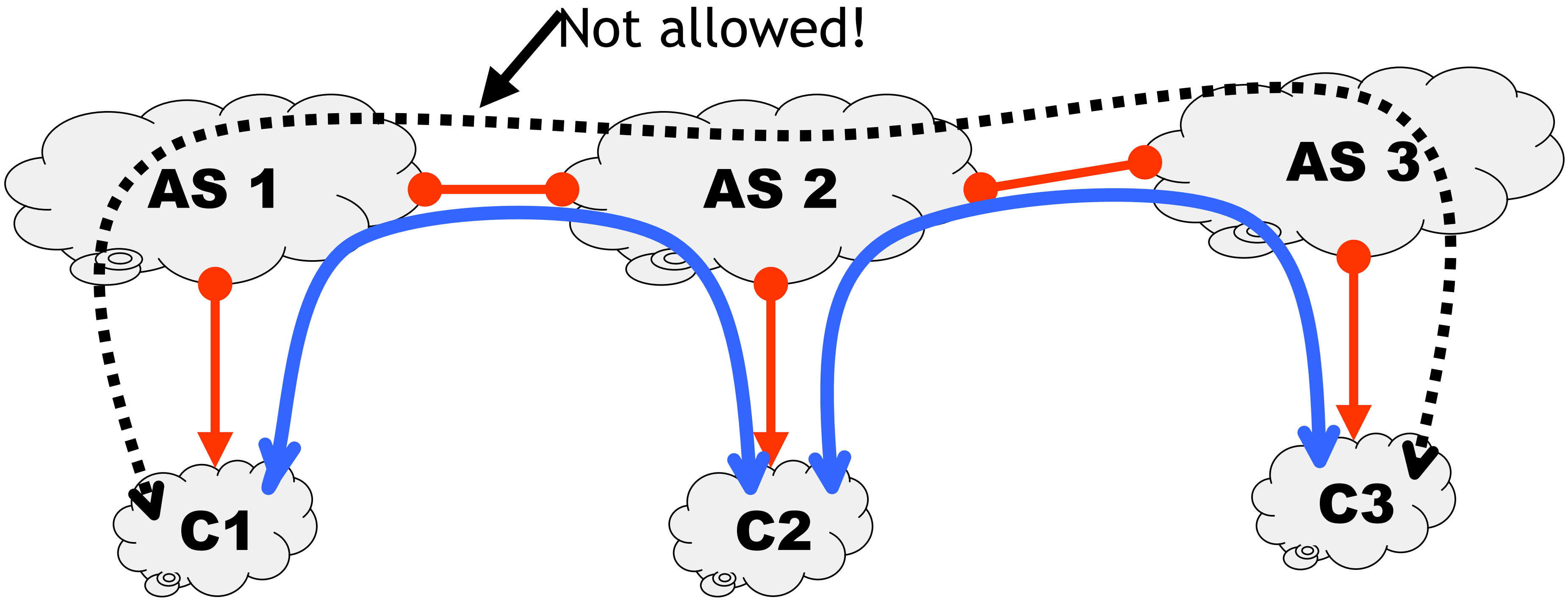
- This also works with multiple connections between Customer and Provider.
- IGP actually takes care of using closest link (how?).

Customer-Provider Hierarchy



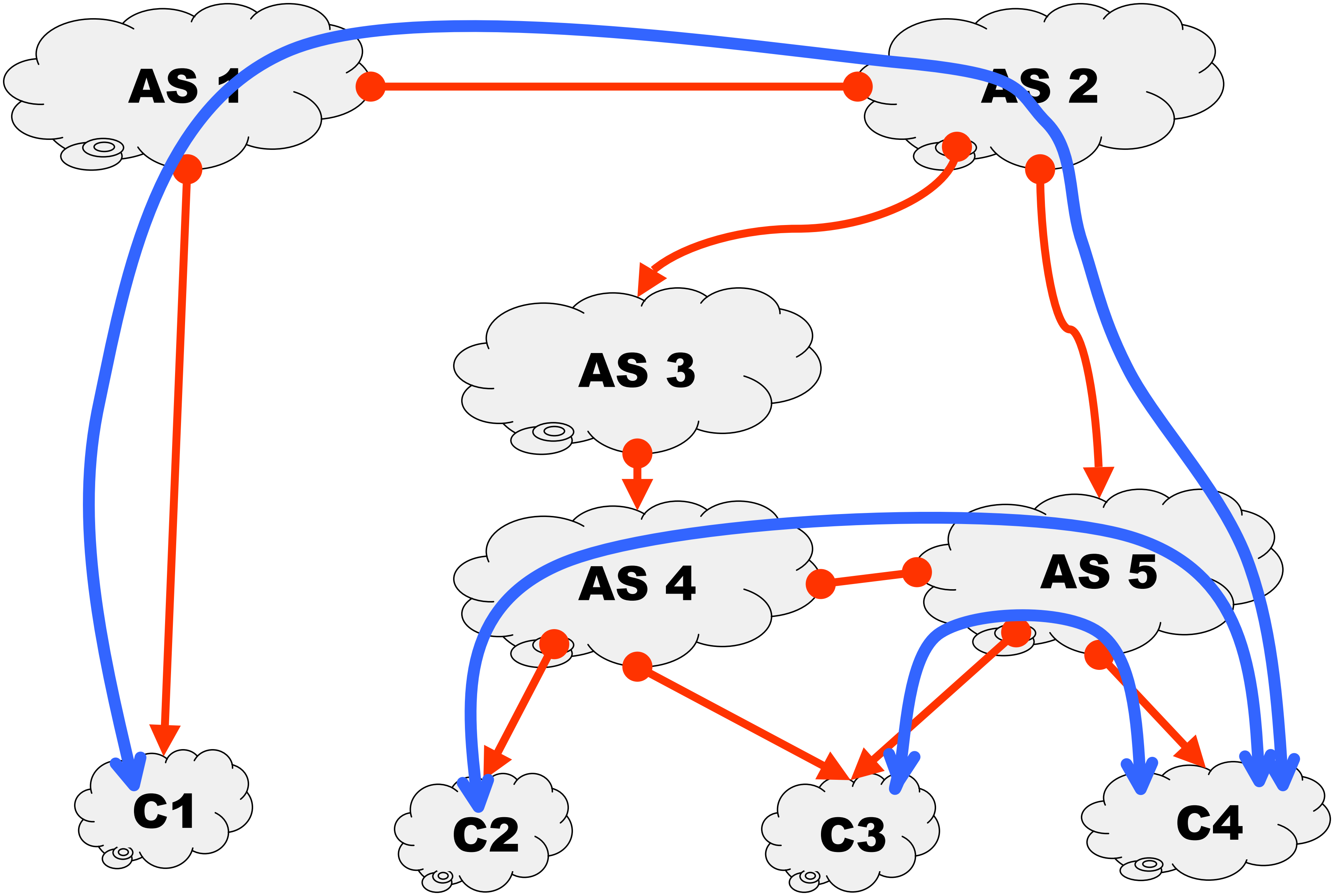
- Customer-Provider relationships can be hierarchical.
- Each network pays their *upstream* provider.

Peering



- Peers provide transit between their respective customers.
- Peers DO NOT provide transit for other peers.
  - They do if they have a customer relationship!
  - How is this enforced?

Peering is About Shortcuts



Peer or Customer?

- Each provider’s customers:
  - Want to “connect” to customers of other providers.
  - Provide services that others may want/need.
- Providers, in response:
  - Should pay to provide upstream service to their customers.
  - Should get paid to make their customers available.
- Peering agreements result from this contention.
  - Peering implies no exchange of money.
  - Your peers are your competitors!
  - Peering agreements are often confidential.
    - And subject to periodic negotiation.

Peer or Customer? Cont’d

- Similar-size providers peer.
  - Tier-1, Tier-2, etc. providers.
- Customers who exchange a lot of traffic may also peer!
- A customer may have multiple upstream providers.
  - Multihoming.
- “Back-doors” may be installed for special customers.
  - Columbia is not Verizon’s customer.
  - But lots of Verizon DSL customers want to connect to Columbia.
  - Verizon may install a private link to Columbia just for their DSL customers.

BGP-4 Overview

- RFC1771.
- BGP runs over TCP (port 179).
- BGP happens between exactly two nodes.
  - *BGP Session* between *BGP Peers*.
    - *BGP Speakers*.
  - A router can have multiple sessions (with multiple peers).
- Maintains the concept of Autonomous System.
- Allows arbitrary AS connectivity.
  - Transit ASes.
  - Non-transit ASes.
  - No such thing as "backbone".
- Objective: find optimal AS paths satisfying policy constraints.

BGP-4 Overview, cont’d

- In a nutshell:
  - Establish connection with peer.
  - Exchange all routes.
  - While link stays up
    - Exchange incremental updates.
- Routes are not refreshed.
  - A route is considered valid until it is changed or withdrawn.
  - Or until the BGP session is terminated.

BGP-4 Overview, cont’d

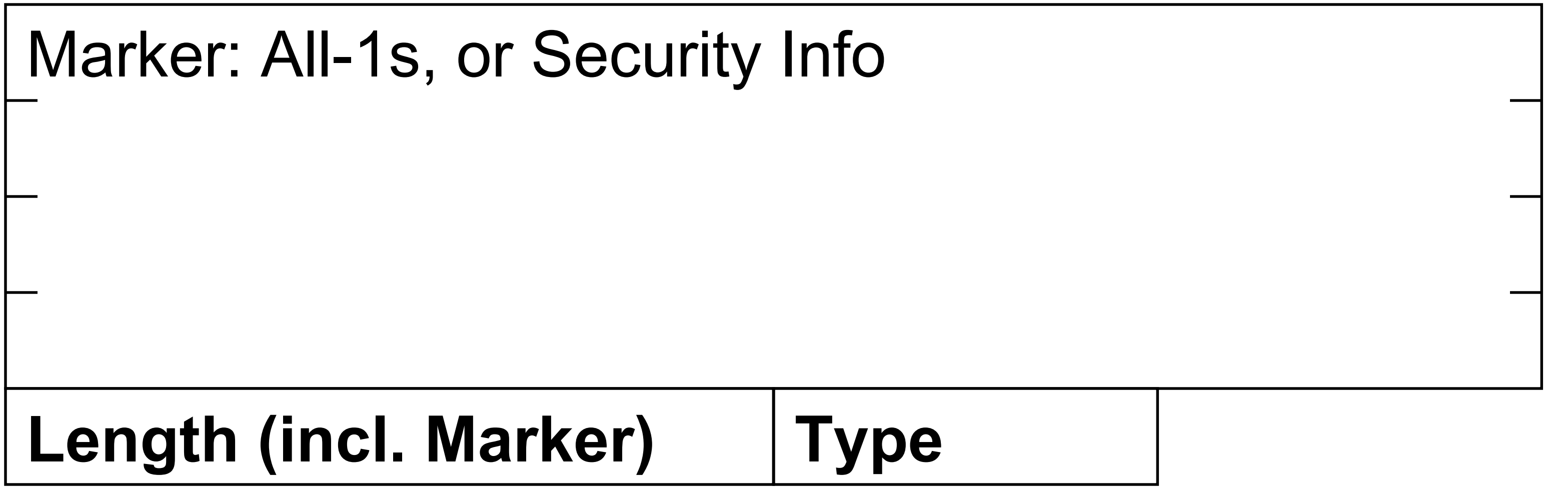
- Advertisements are about reachability.
  - A advertises to B a path for N.
  - B is assured that A uses that path to reach N.
- Path-Vector:
  - Almost like DV, except complete paths are advertised.
    - Loops are prevented this way.
- Attributes:
  - That’s what makes BGP so flexible and extensible …
  - and prone to misconfigurations.
  - Next hops, various metrics, path, …
  - Lots of new attributes defined since RFC1771.

Bringing up BGP

- *BGP Peers*: endpoints of a *BGP Session*.
- BGP Peers are configured.
  - No automatic discovery.
- Start at *Idle* state.
- Attempt TCP connection: *Connect* state.
- While establishing TCP connection: *Active* state.

- Now BGP messages can be sent.
  - While TCP connection is up.

BGP Message Common Header



Marker: All-1s, or Security Info

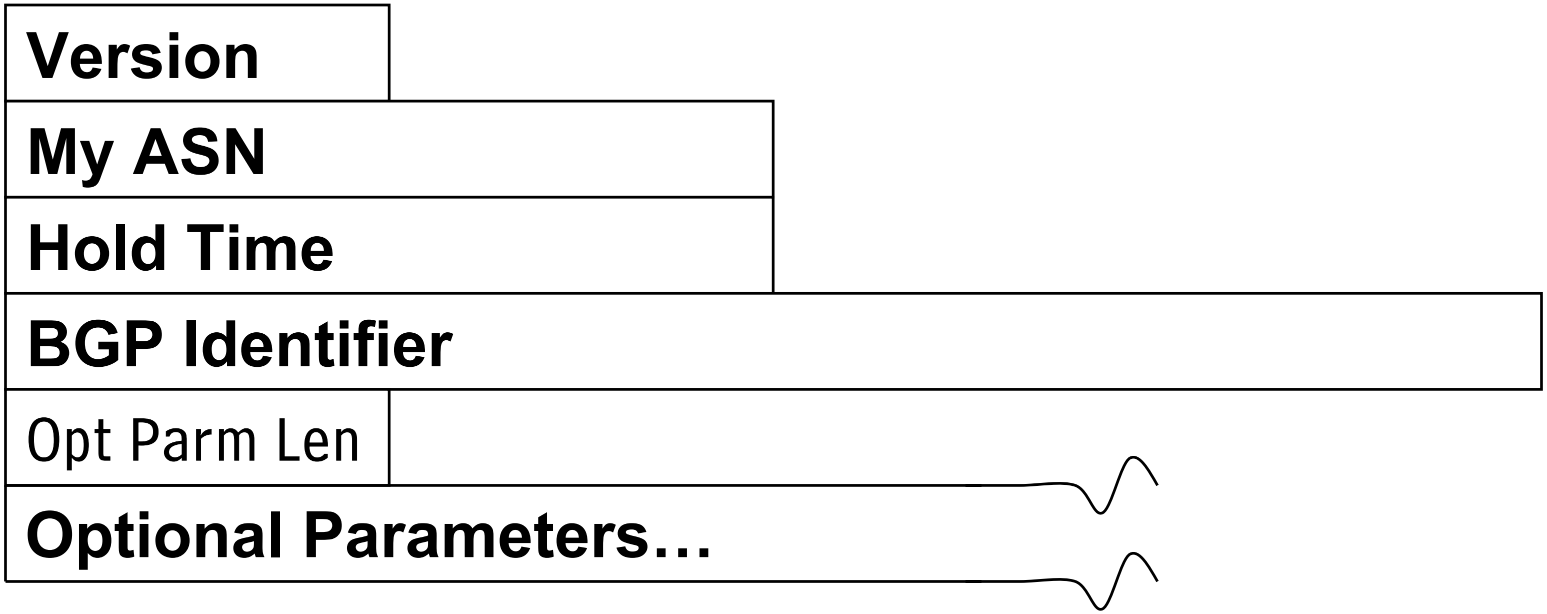
Length (incl. Marker)

Type

- Type is one of:
  - OPEN (1)
  - UPDATE (2)
  - NOTIFICATION (3)
  - KEEPALIVE (4)

BGP OPEN

- BGP speakers identify each other.
  - And verify that they are who they are supposed to be.
- Verify they speak the same version of BGP.
- Inform each other of their ID.
- Exchange/negotiate optional parameters.



Version

My ASN

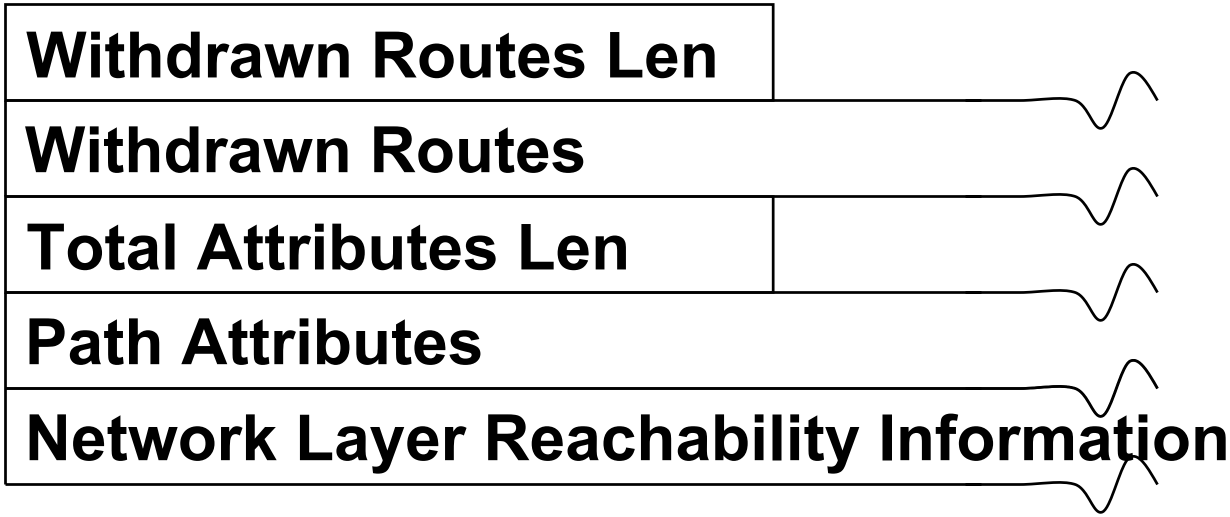
Hold Time

BGP Identifier

Opt Parm Len

Optional Parameters…

BGP UPDATE



Withdrawn Routes Len

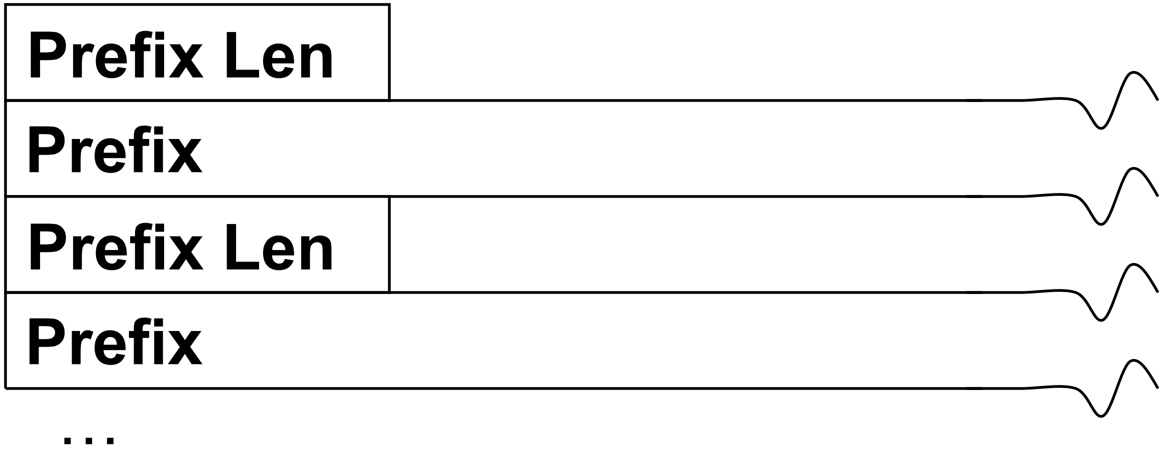
Withdrawn Routes

Total Attributes Len

Path Attributes

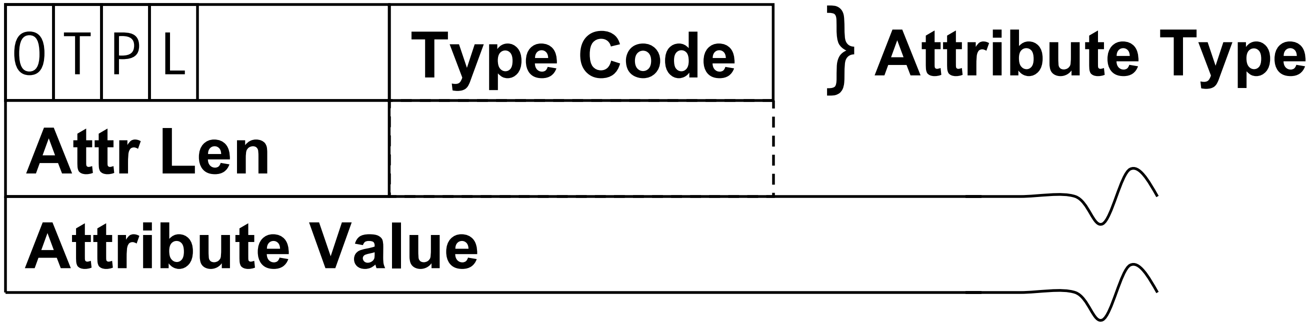
Network Layer Reachability Information

Withdrawn Routes



- List of IP prefixes to withdraw.
- Length is the prefix length.
- Prefix is padded to a multiple of 8 bits.
  - Pad bits ignored.

Path Attributes



O T P L | Type Code } Attribute Type

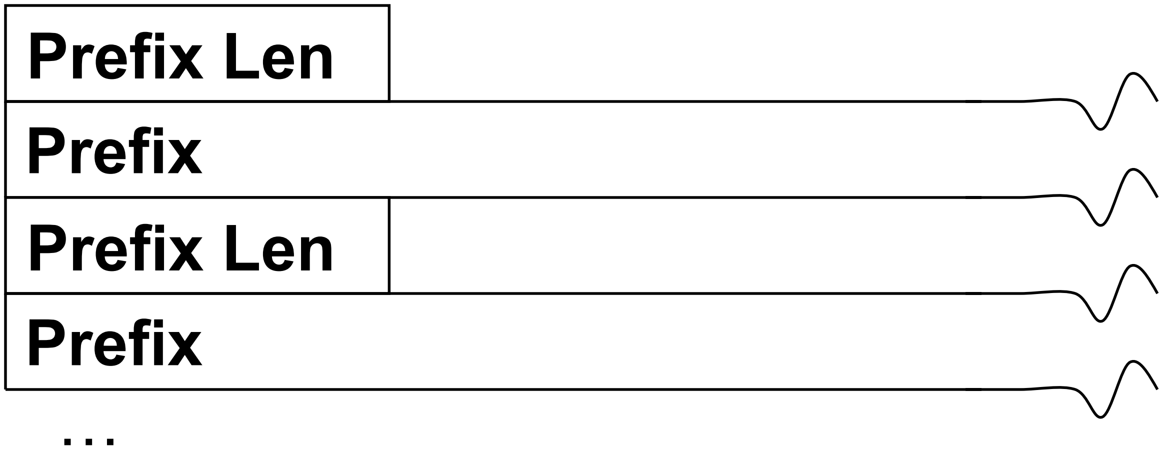
Attr Len

Attribute Value

- O: Optional/Well Known
- T: Transitive/Nontransitive (passed on to peers)
- P: Partial: only some routers in the path understand an Optional and Transitive attribute.
  - If O=0 and T=0 then P must be 0.
- L: Extended Length: L=1 means length field is 2 bytes.

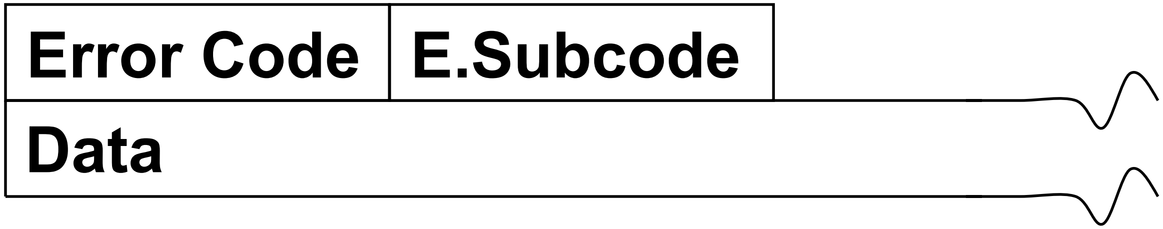
- Attributes apply to all advertised prefixes in the UPDATE message.

Network Layer Reachability Information



- List of advertised prefixes.
- All attributes apply to all prefixes.
- Prefixes with different attributes are advertised in separate UPDATE messages.

BGP NOTIFICATION



Error Code

E.Subcode

Data

- Report errors about:
  - Format of received message.
  - Unexpected state.
  - Timers expiring.
- The TCP connection is closed right after the NOTIFICATION.
  - All notifications are fatal!

BGP KEEPALIVE

- Sent if there have been no updates in the last HoldTimer seconds.
- Syntactically, just a BGP header with Type=4

(About Keepalives)

- Some TCP implementations have the notion of a keepalive:
  - Packet sent periodically to probe the connection.
- What it does keep alive is the underlying link IF the underlying link depends on continuous traffic to stay up (e.g., dialup).

- TCP state is kept only at the endpoints.
  - Intermediate hops do not need to be refreshed.
- If intermediate links go away temporarily, TCP will keep retransmitting until they come back up.
- In most cases, tearing down a link when no other data traffic would have flowed anyway is wasteful.

- Hence the term “makedeads”.

Keepalive

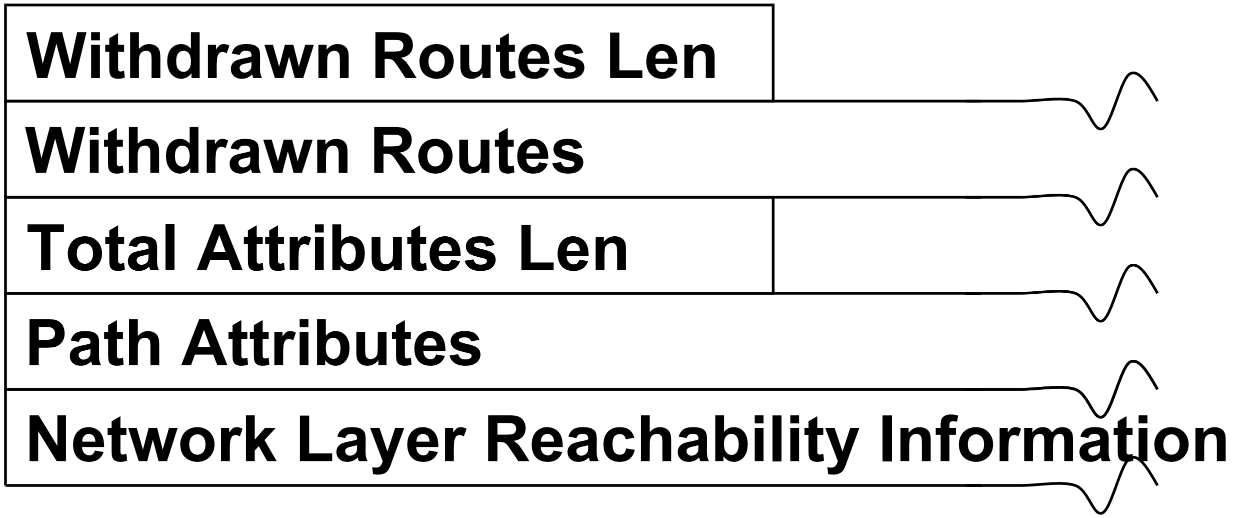
- In BGP, we DO want a Makedead!
- A failed link indicates that routing should change.
  - Since BGP messages are exchanged over the same link that all other traffic would be routed.
  - (There is an exception to this, don’t worry about it yet.)
- Detects if the link has failed, and tears the session down.
- A torn-down BGP session causes routes to be withdrawn
  - This is the desired behavior.

Conceptual Model of Operation

- BGP is about advertising prefixes.
  - Some prefixes are learned from BGP neighbors.
  - Some more prefixes are also learned from the IGP.
  - Some of these prefixes are advertised to neighbors.
- RIB: Routing Information Base.
- Each router keeps:
  - One **Adj-RIB-In** for each peer.
    - Stores prefixes learned from each peer.
  - Prefixes from all the **Adj-RIB-In**s are selected for use.
  - Stored in the **Loc-RIB**.
    - One per router.
  - One **Adj-RIB-Out** for each peer.
    - Stores prefixes to be advertised to each peer.

Back to BGP

- Path Attributes in particular.



Withdrawn Routes Len

Withdrawn Routes

Total Attributes Len

Path Attributes

Network Layer Reachability Information

Path Attributes

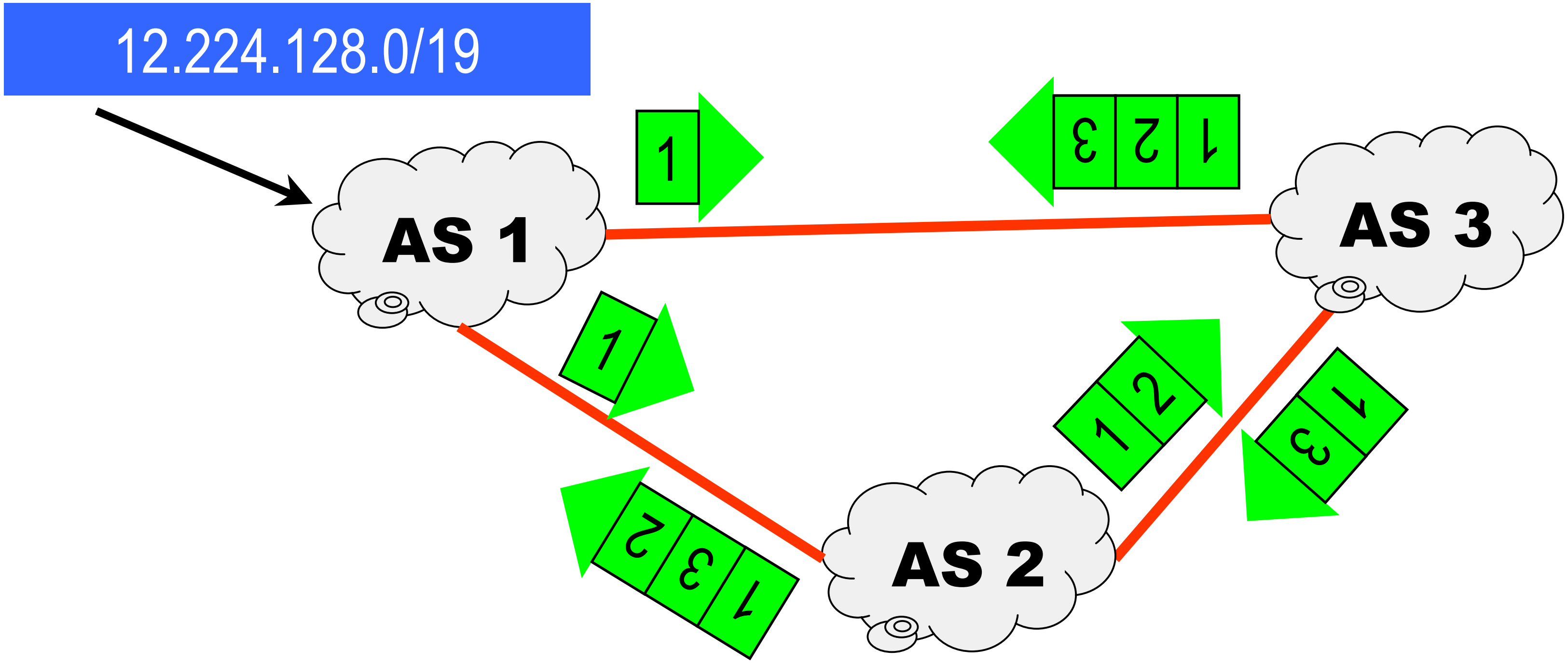
1	ORIGIN	RFC 1771
2	AS_PATH	RFC 1771
3	NEXT_HOP	RFC 1771
4	MULTI_EXIT_DISCRIMINATOR	RFC 1771
5	LOCAL_PREF	RFC 1771
6	ATOMIC_AGGREGATE	RFC 1771
7	AGGREGATOR	RFC 1771
8	COMMUNITY	RFC 1997
9	ORIGINATOR_ID	RFC 2796
10	CLUSTER_LIST	RFC 2796
11	DPA	deprecated
12	ADVERTISER	RFC 1863
13	RCID_PATH/CLUSTER_ID	RFC 1863
14	MP_REACH_NLRI	RFC 2858
15	MP_UNREACH_NLRI	RFC 2858
16	EXTENDED COMMUNITIES	draft-ietf-idr-bgp-ext-communities-06.txt
17	NEW_AS_PATH	draft-ietf-idr-as4bytes-07.txt
18	NEW_AGGREGATOR	draft-ietf-idr-as4bytes-07.txt
...	...	
255	Reserved for development	

ORIGIN

- Well-known, Mandatory. Type=1
- Shows how a prefix was learned.
  - Prefixes are *injected* into BGP
- Length=1
- Value:
  - IGP (=1): Prefix was learned from an IGP.
  - EGP (=2): Prefix was learned from the EGP (BGP).
  - INCOMPLETE (=3): Prefix was learned some other way.
    - Static routes/directly connected networks.

AS_PATH

- ASNs through which the announcement for these prefixes has passed.
- First ASN in the AS_PATH: Origin AS.
- Each AS appends its own ASN before passing on the update.

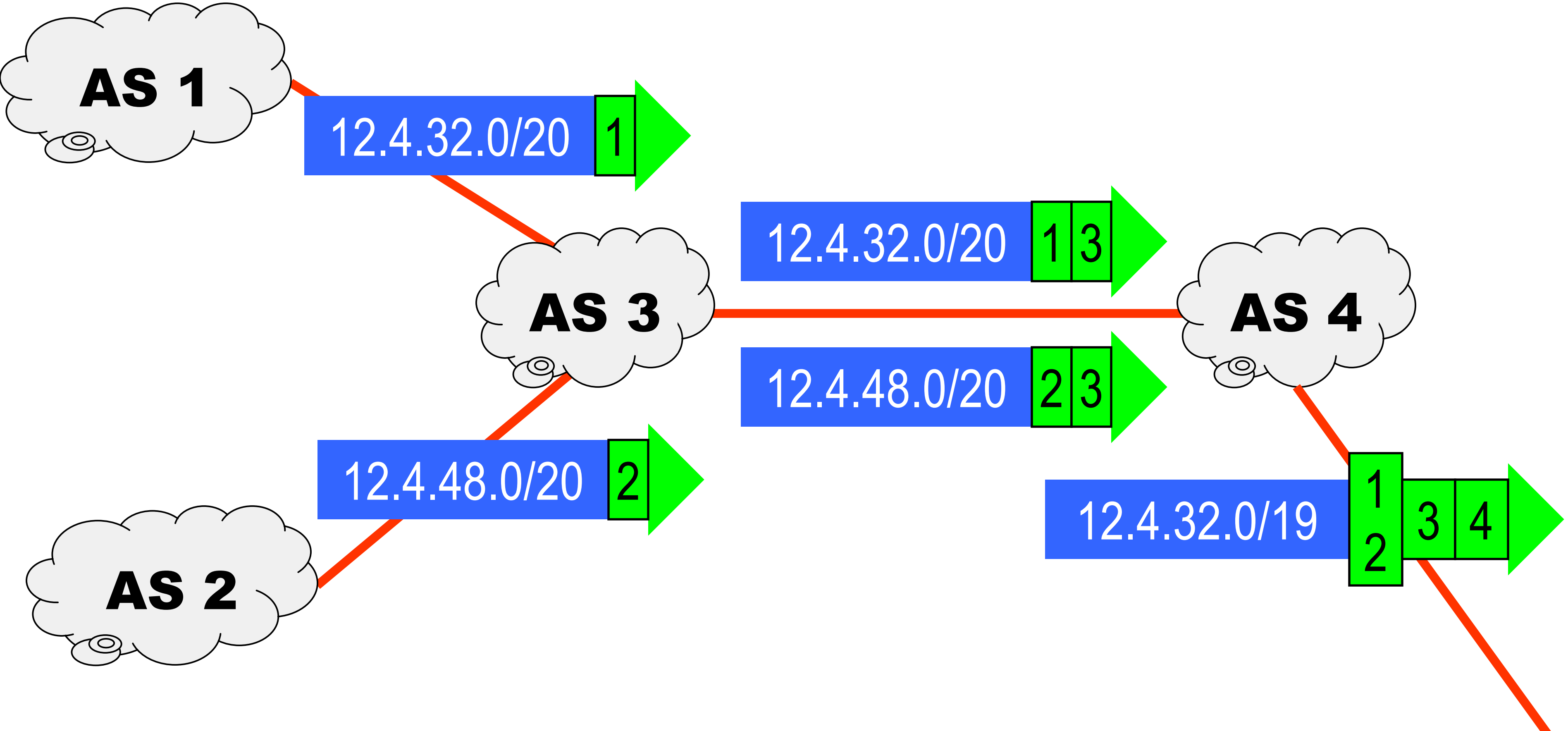


AS_PATH Cont’d

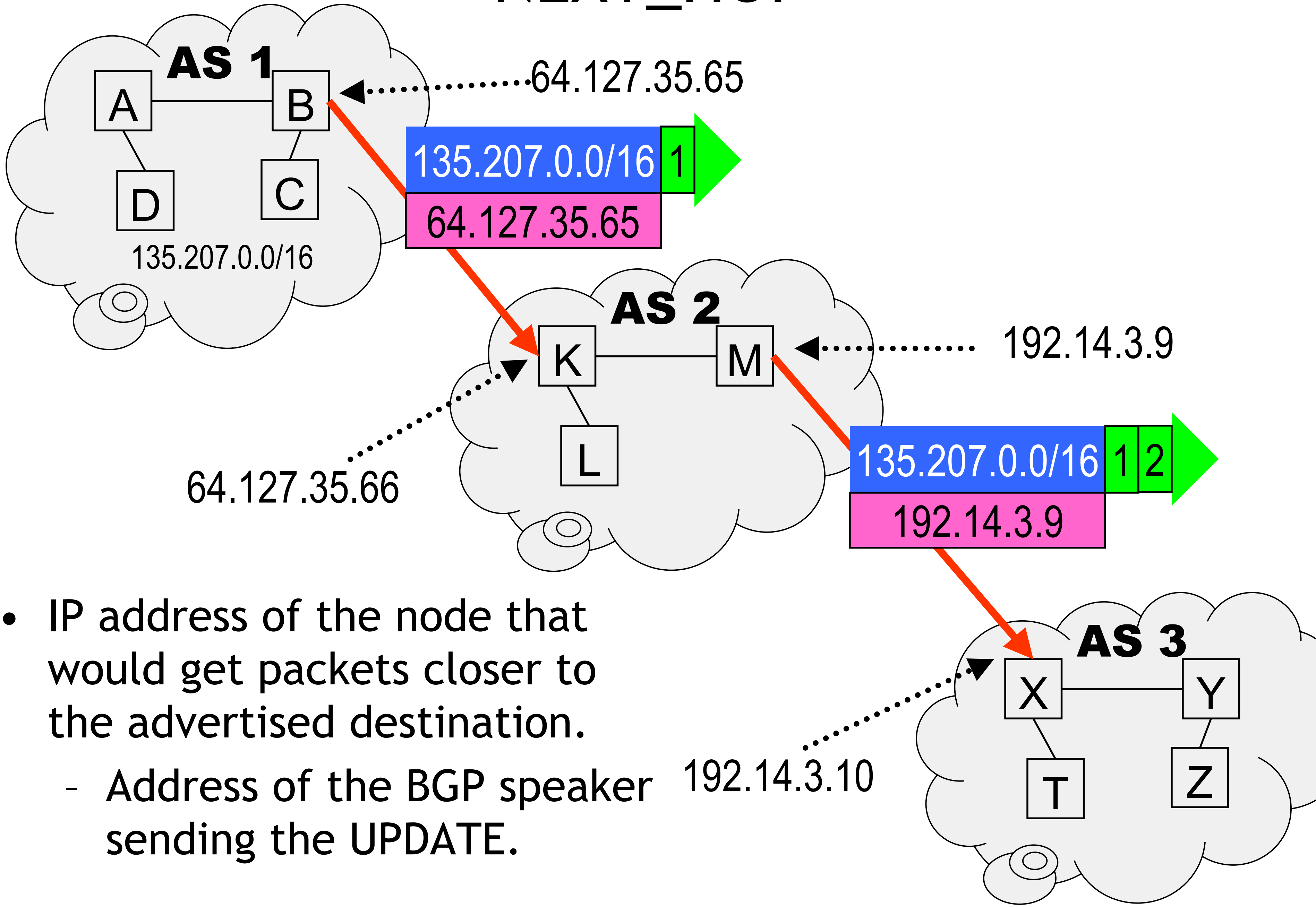
- Well-known, Mandatory. Type=2
- Encoded as sequence of AS_PATH segments.
- Each segment is encoded as:
  - Path Segment Type:
    - AS_SET (1): unordered set of ASNs.
    - AS_SEQUENCE (2): ordered set of ASNs.
  - Path Segment Length: 1 octet, #of ASNs in segment.
  - Path Segment Value: 2*PSL octets, list of ASNs.
- New ASNs are actually **prepended** in the packet.
- If leading segment is AS_SET, a new AS_SEQUENCE is prepended with the ASN as its sole member.
- If leading segment is AS_SEQUENCE, the ASN is just prepended to the sequence.

AS_PATH Cont’d

- Most AS_PATHs are encoded as a single AS_SEQUENCE.
- If a router needs to aggregate, it has to use AS_SET.
- Not common, since most routers aggregate prefixes from their own AS.



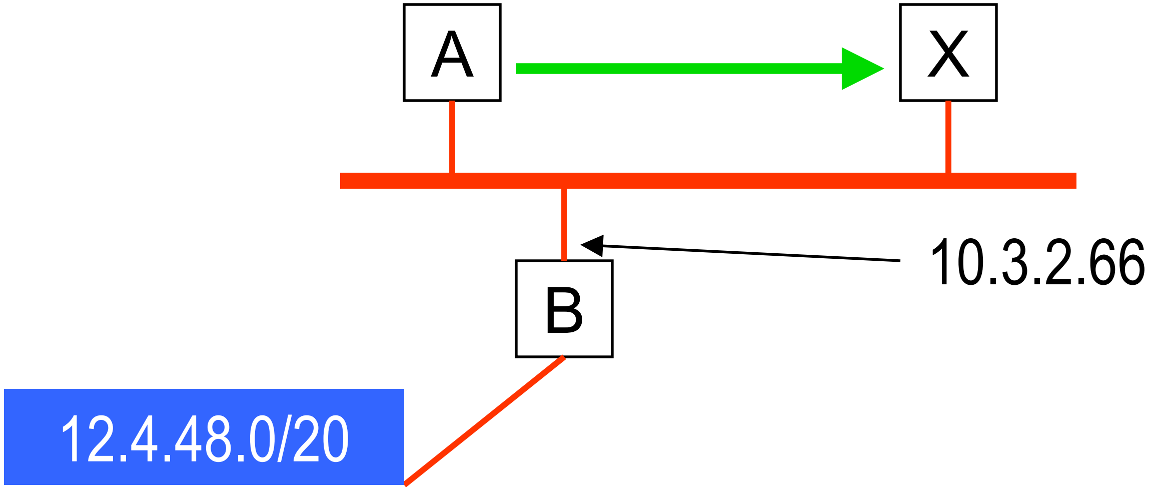
NEXT_HOP



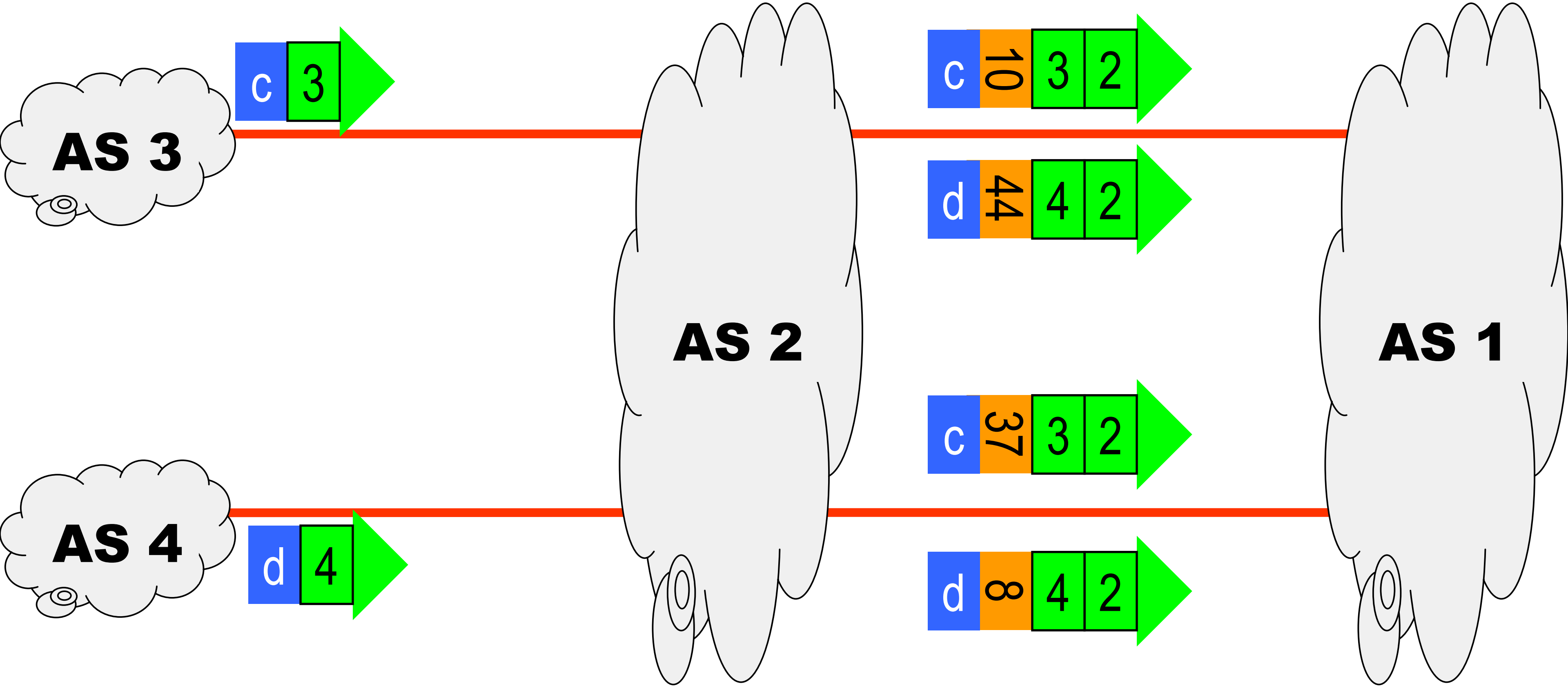
- IP address of the node that would get packets closer to the advertised destination.
  - Address of the BGP speaker sending the UPDATE.

NEXT_HOP cont'd

- Well-known, Mandatory. Type=3
- Encoded as the 4-octet address right after the Type Code.
- IP address of the node that would get packets closer to the advertised destination.
  - Address of the BGP speaker sending the UPDATE.
- Exception: A (BGP speaker) sends X (BGP speaker) an UPDATE indicating B (10.3.2.66 interface) (not a BGP speaker) is the router for 12.4.48.0/20.



MULTI_EXIT_DISCRIMINATOR (MED)



- AS2 includes MED to the updates it sends to AS1.
- AS3 and AS4 are advertised over both links, of course.
- AS1 can now make a better choice about sending packets to AS3 and AS4.

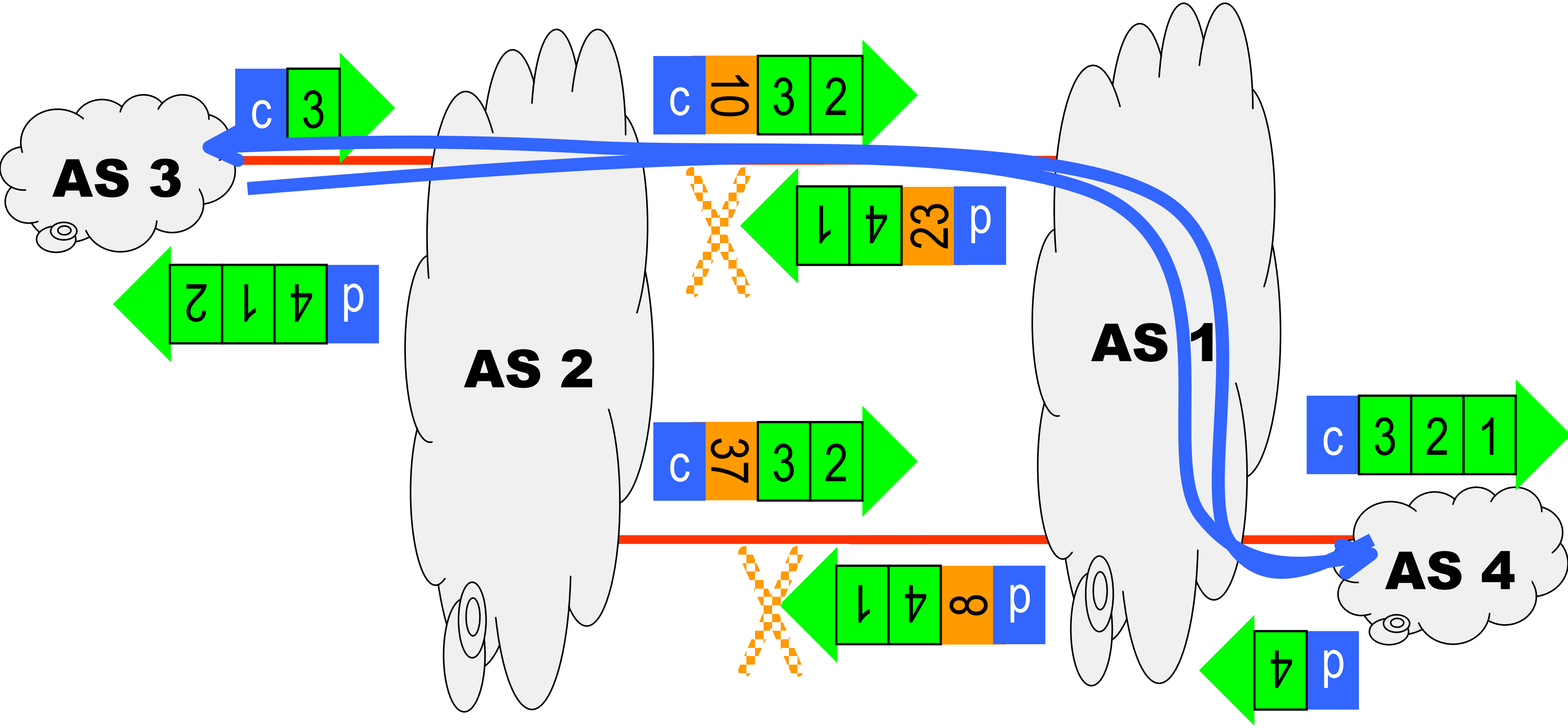
MED Cont’d

- One AS sets MED, but another uses it.
  - MED only used in Customer/Provider relationships (why?).
- Peers usually ignore received MEDs (why?).

- Well-known, discretionary (why?). Type=4
- Length is always 4, encoding is unsigned integer.

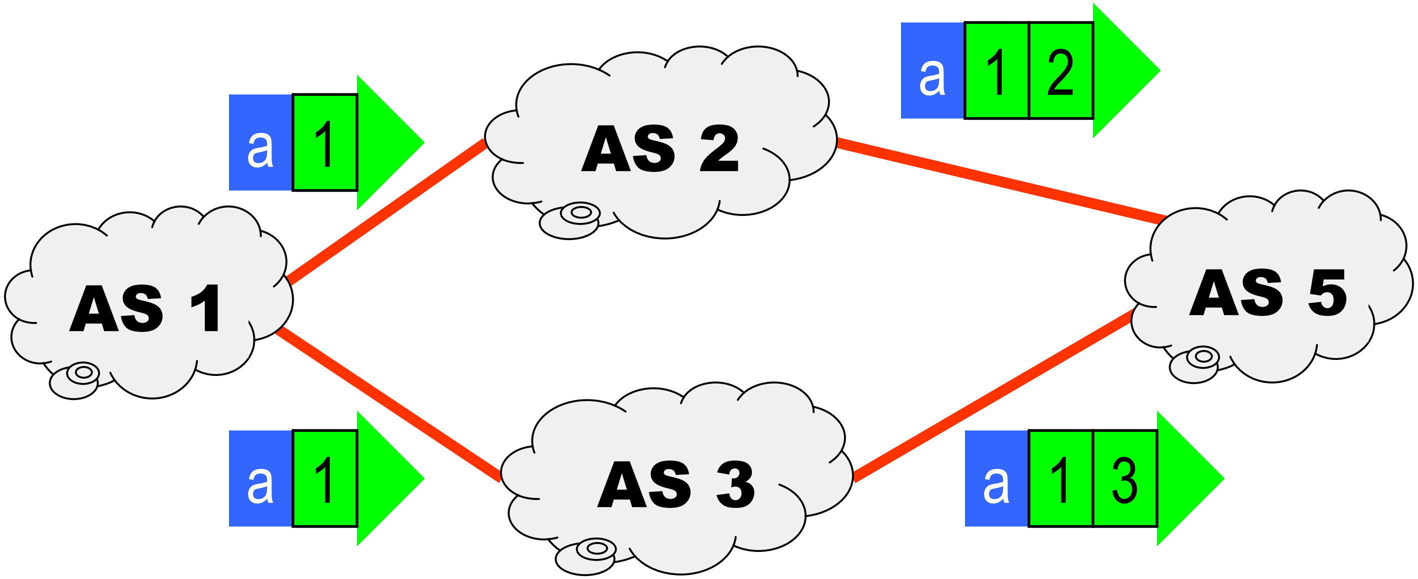
- MED is usually the IGP metric for the advertised prefix.
- MED comparison only makes sense when received from the same AS.

MED Cont'd



- MED can be (ab)used to get one ISP to carry more traffic.
- Traffic from AS3 to AS4 goes to closest link.
- Traffic from AS4 to AS3 obeys MED.

LOCAL_PREF



- How does AS5 decide how to send traffic to prefix a?
- MED doesn’t help here.
  - Only one link between AS pairs.
  - AS5 may want to set its own policy about this.
- AS5 uses the LOCAL_PREF attribute on routes it receives.
- LOCAL_PREF is the first attribute used in route selection.

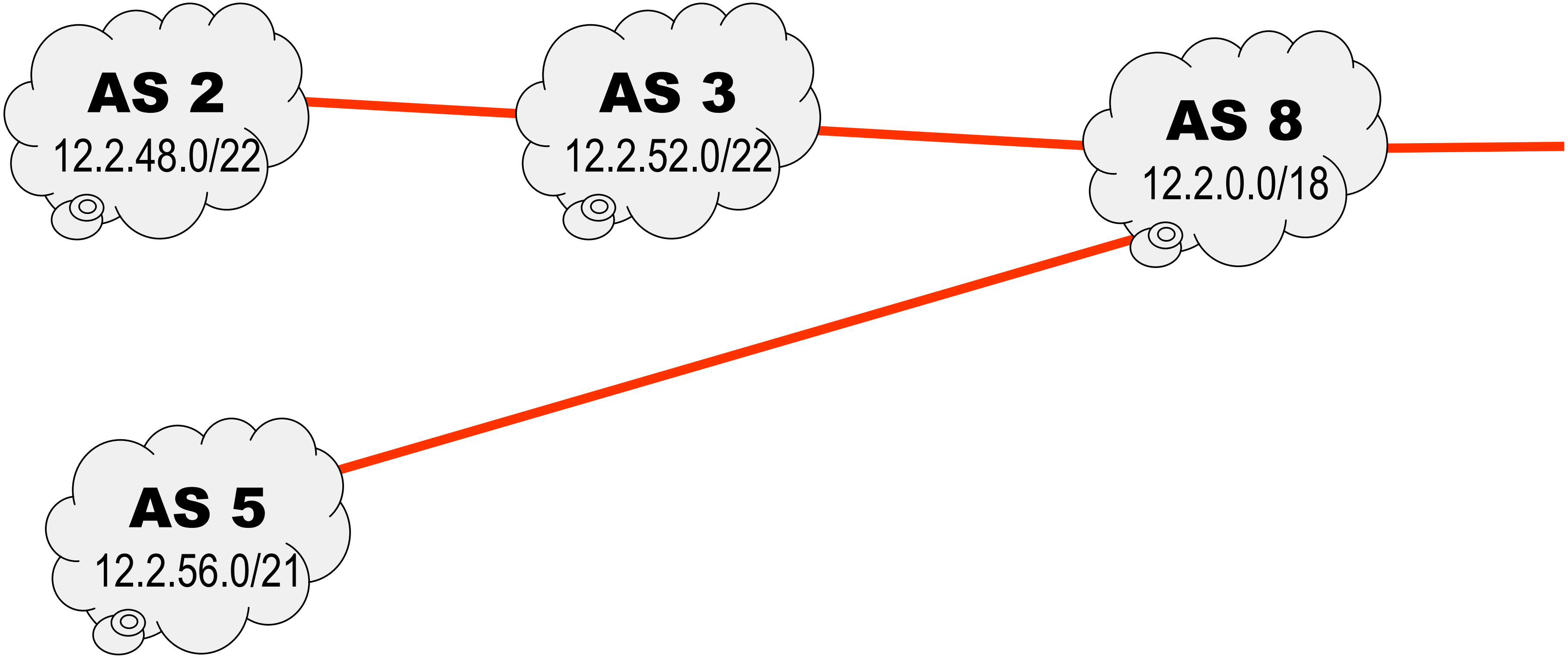
LOCAL_PREF Cont’d

- LOCAL_PREF is computed locally when route received from E-BGP, IGP, or statically assigned.
  - Part of the interface configuration.
  - Stored in the Adj-RIB-In.
- LOCAL_PREF is carried in I-BGP.
  - Don’t worry about this right now!

- Well-known, Discretionary. Type=5
- Length is always 4.
- Encoding is unsigned integer.

Route Aggregation

- AS2 and AS3 can be aggregated into 12.2.48.0/21.
- AS8's space covers that of AS2, AS3, and AS5.
- What should AS8 advertise upstream?

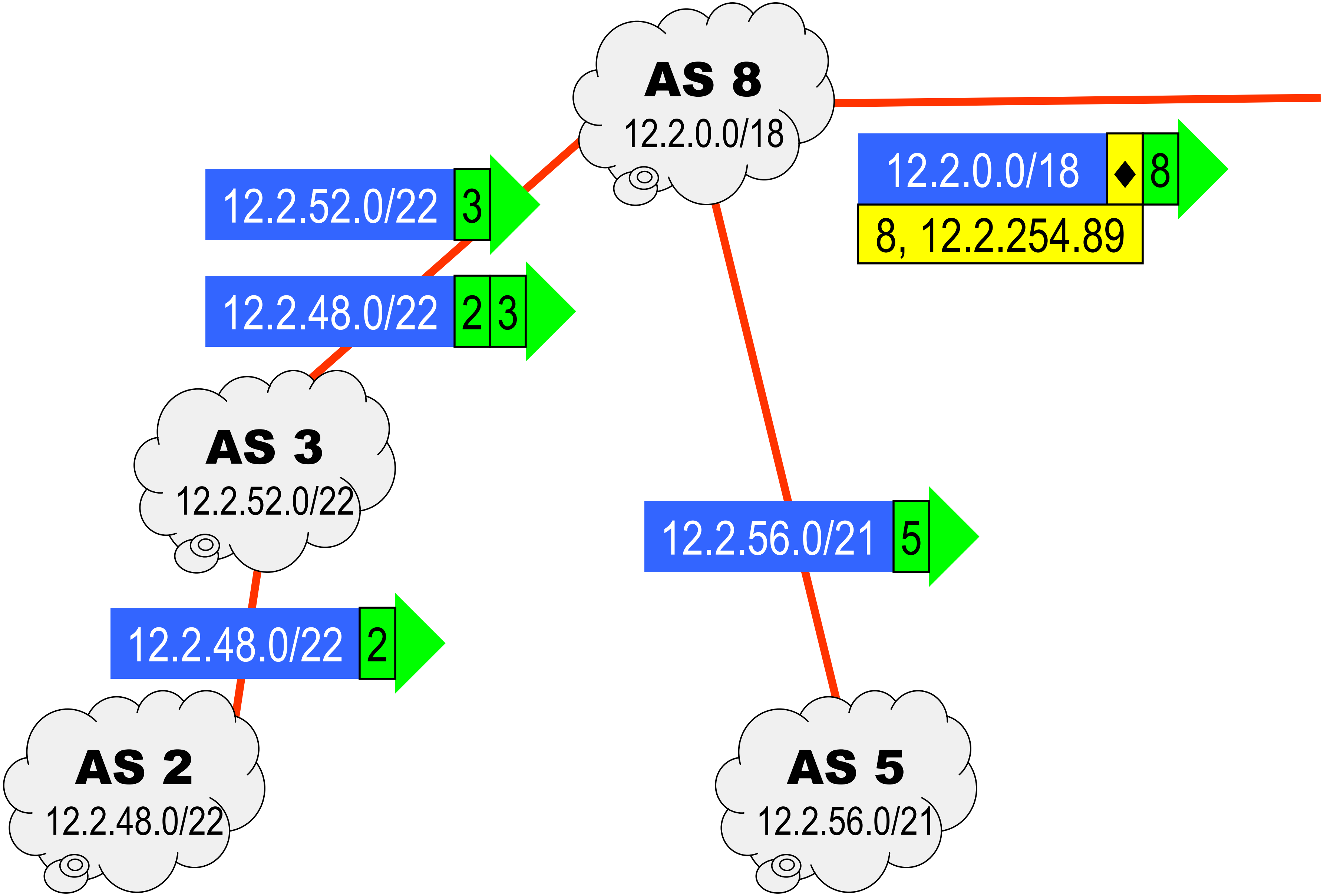


Route Aggregation, Cont’d

- AS8 could advertise:
  - Nothing, or some subset of the routes (subj. to policy).
  - All four routes.
  - Advertise just its own (less-specific) route.
    - 12.2.0.0/18 (AS8)
  - De-aggregate its own prefix and advertise more-specifics:
    - 12.2.0.0/19 (AS8)
    - 12.2.32.0/20 (AS8)
    - 12.2.48.0/22 (AS2, AS3, AS8)
    - 12.2.52.0/22 (AS3, AS8)
    - 12.2.56.0/21 (AS5, AS8)
- Aggregation saves space but destroys information.

ATOMIC_AGGREGATE & AGGREGATOR

- If a BGP speaker aggregates routes.
  - AS_PATH information is lost.
- Following routers must be alerted.
  - So they don't de-aggregate the advertised prefix.
- The ATOMIC_AGGREGATE attribute provides that feature.
  - Well-known, Discretionary. Type=6.
  - Zero length (just a flag).
  - Must remain attached.
- AGGREGATOR attribute:
  - Indicates which AS and router performed the aggregation.
  - Optional, transitive. Type=7.
  - Length is always 6.
  - 2-byte ASN, 4-byte IP address of aggregator.


AS 8
12.2.0.0/18
12.2.0.0/18
◆
8
8, 12.2.254.89
12.2.52.0/22
3
12.2.48.0/22
2
3
AS 3
12.2.52.0/22
12.2.56.0/21
5
12.2.48.0/22
2
AS 2
12.2.48.0/22
AS 5
12.2.56.0/21

COMMUNITY

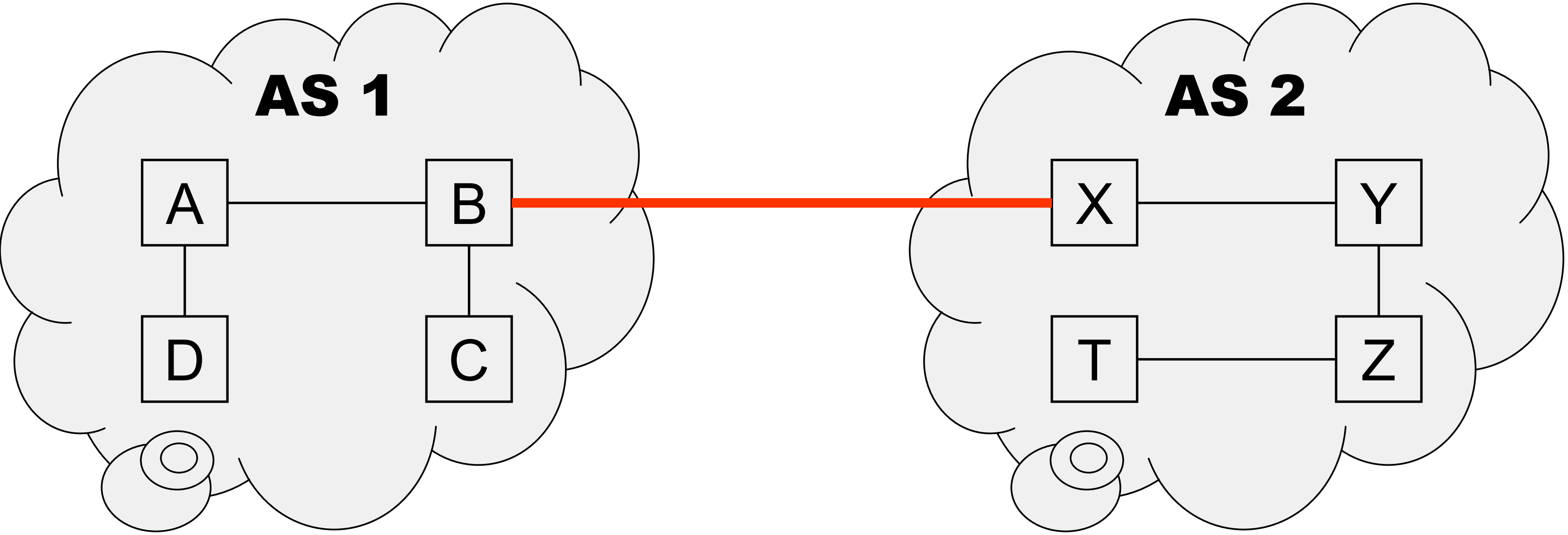
- Specified in RFC 1997.
- Encodes arbitrary properties.
  - E.g., all of customer's routes get a specific COMMUNITY.
- Much of the policy is specified using communities.

- Optional, Transitive. Type=8
- Four bytes: (*e.g.*, 7018:100)
  - 2 bytes ASN (by convention).
  - 2 bytes administratively defined (no predefined meaning).

- We'll talk about this in the next lecture.

Learning External Prefixes

- So far, BGP has been presented as a pure EGP.
  - A protocol that runs between ASs.



- How do A, C and D learn about AS2’s routes?
  - Ditto for Y, Z, T about AS1’s routes?
- I.E., how are prefixes learned by an ASBR distributed inside the AS?

Learning External Prefixes, cont’d

- Inject into the IGP (using AS-External LSAs).
- Small networks can do this.
  - Default route + a few external routes.
- Does not work for large ISPs.
  - They carry a full routing table (100K-400K routes!).
- Would lose policy information.
  - No way to carry attributes.
- IGPs don’t scale well.
  - Computational complexity.
  - Memory requirements.
  - Additional traffic.
    - Fragmented LSAs.
- Clearly need a different way!

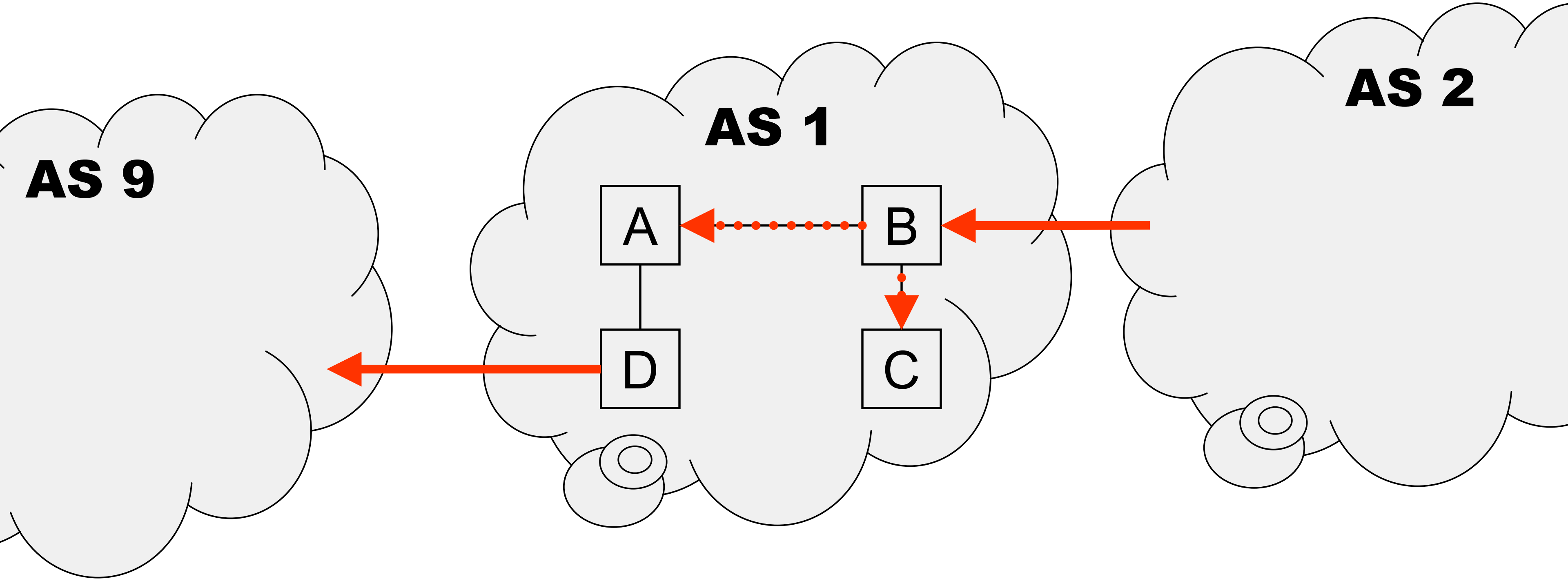
E-BGP and I-BGP

- The solution is called *Internal-BGP (I-BGP)*.
  - As opposed to *External-BGP (E-BGP)*.
- E-BGP is used between ASs.
- I-BGP is used **within** an AS.
  - Is used to distribute routes learned with E-BGP.
- E-BGP and I-BGP are the same protocol.
  - Same messages, attributes, state machine, etc.
- But: different rules about route redistribution:

		Redistribute to	
		I-BGP	E-BGP
Learned from	I-BGP	no	yes
	E-BGP	yes	(yes)

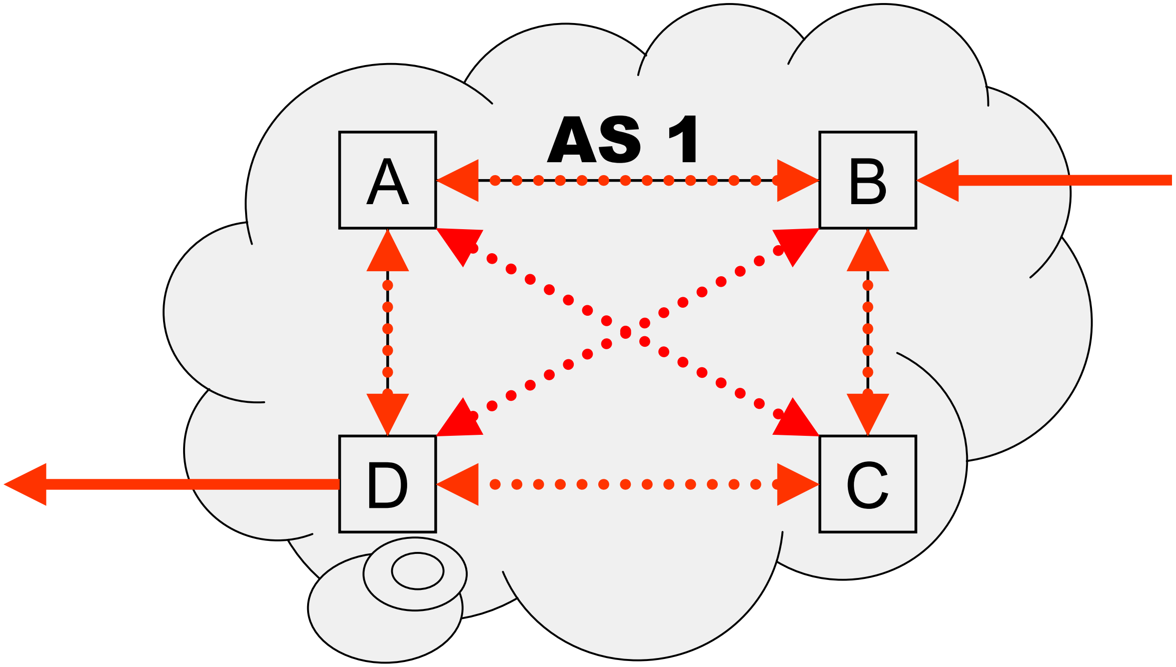
I-BGP Route Redistribution

- How does D learn routes acquired by B?
  - Since A can’t redistribute routes learned over I-BGP?
- If D also had an external connection, how would it redistribute routes learned from other ASs?



I-BGP Route Redistribution, cont’d

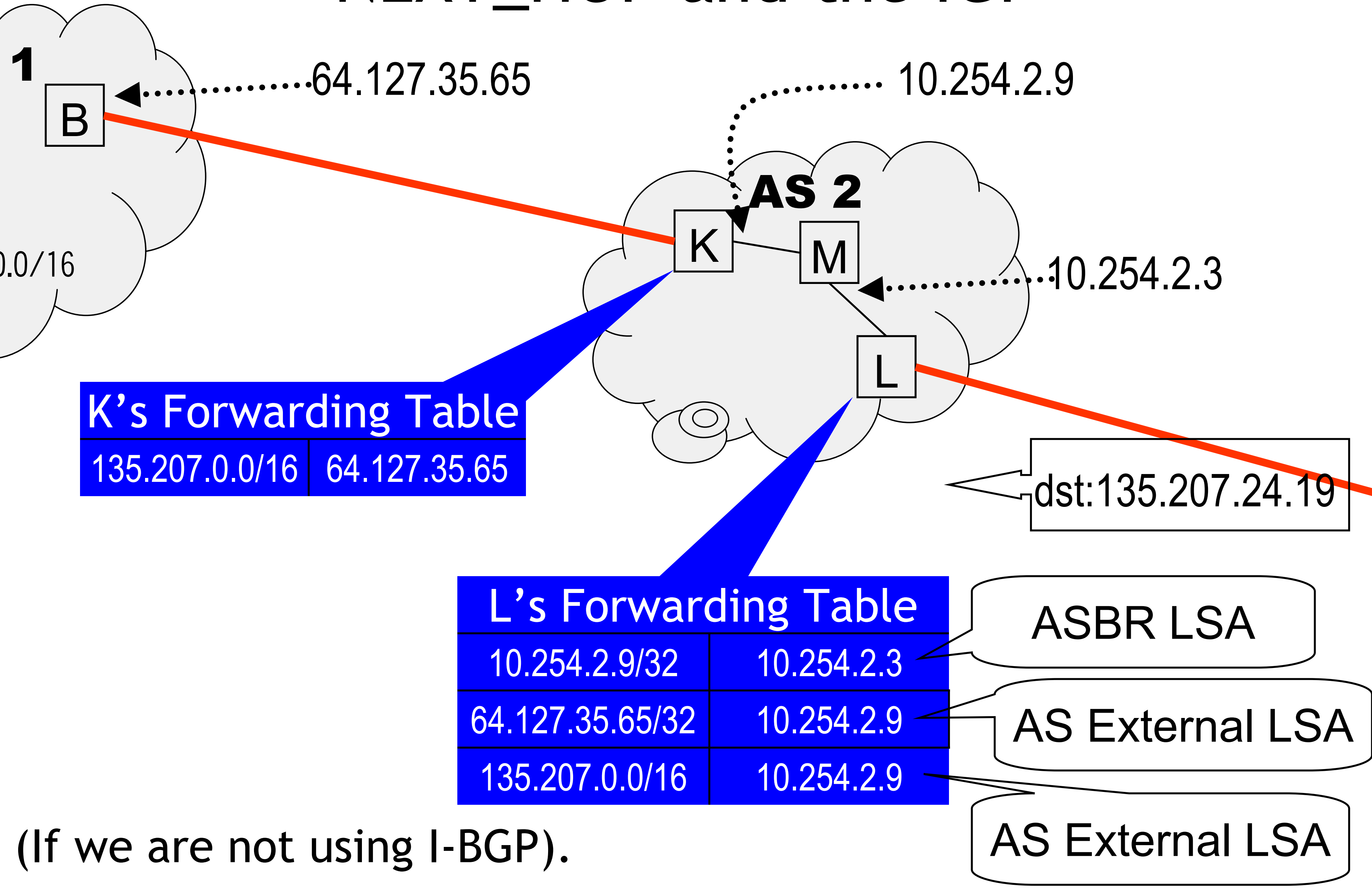
- Remember: BGP is a **routed** protocol.
- Routes between routers already exist.
  - Carried by the IGP.
- I-BGP sessions can be formed between non-adjacent routers.
- I-BGP sessions must form a full mesh:



I-BGP, cont'd

- Full mesh.
- Independent of actual links between (internal) routers.
- TCP src/dst of I-BGP session must be a loopback address.
  - Routing to the router must be independent of interfaces going up/down.
- Full mesh is necessary to prevent loops.
  - AS_PATH is used to detect loops in E-BGP.
  - ASN appended to AS_PATH only when route is advertised to E-BGP peer.
- I-BGP is **NOT** an IGP.
  - Nor can be used as one.

NEXT_HOP and the IGP



K’s Forwarding Table

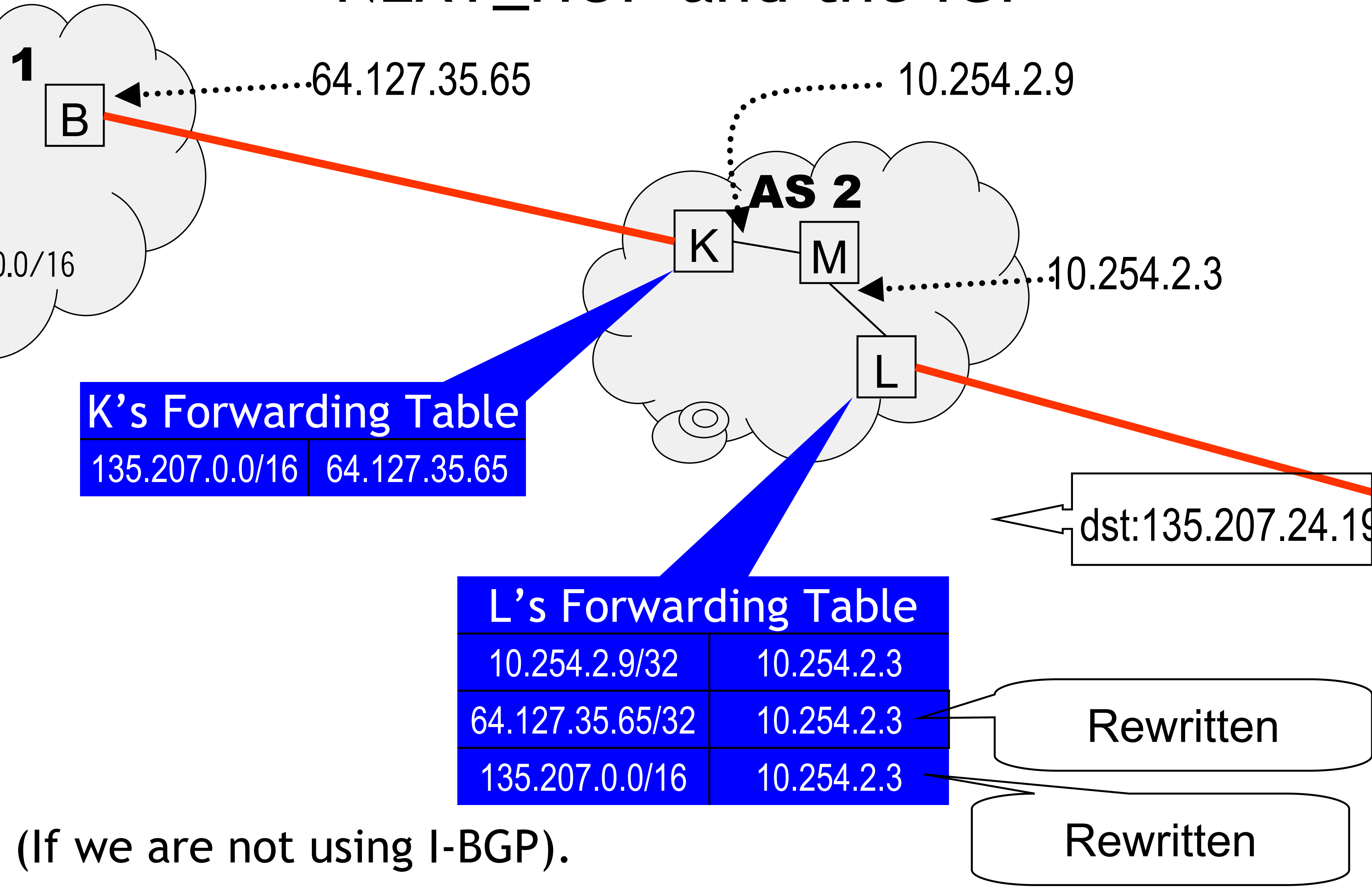
135.207.0.0/16	64.127.35.65

L’s Forwarding Table

10.254.2.9/32	10.254.2.3
64.127.35.65/32	10.254.2.9
135.207.0.0/16	10.254.2.9

(If we are not using I-BGP).

NEXT_HOP and the IGP

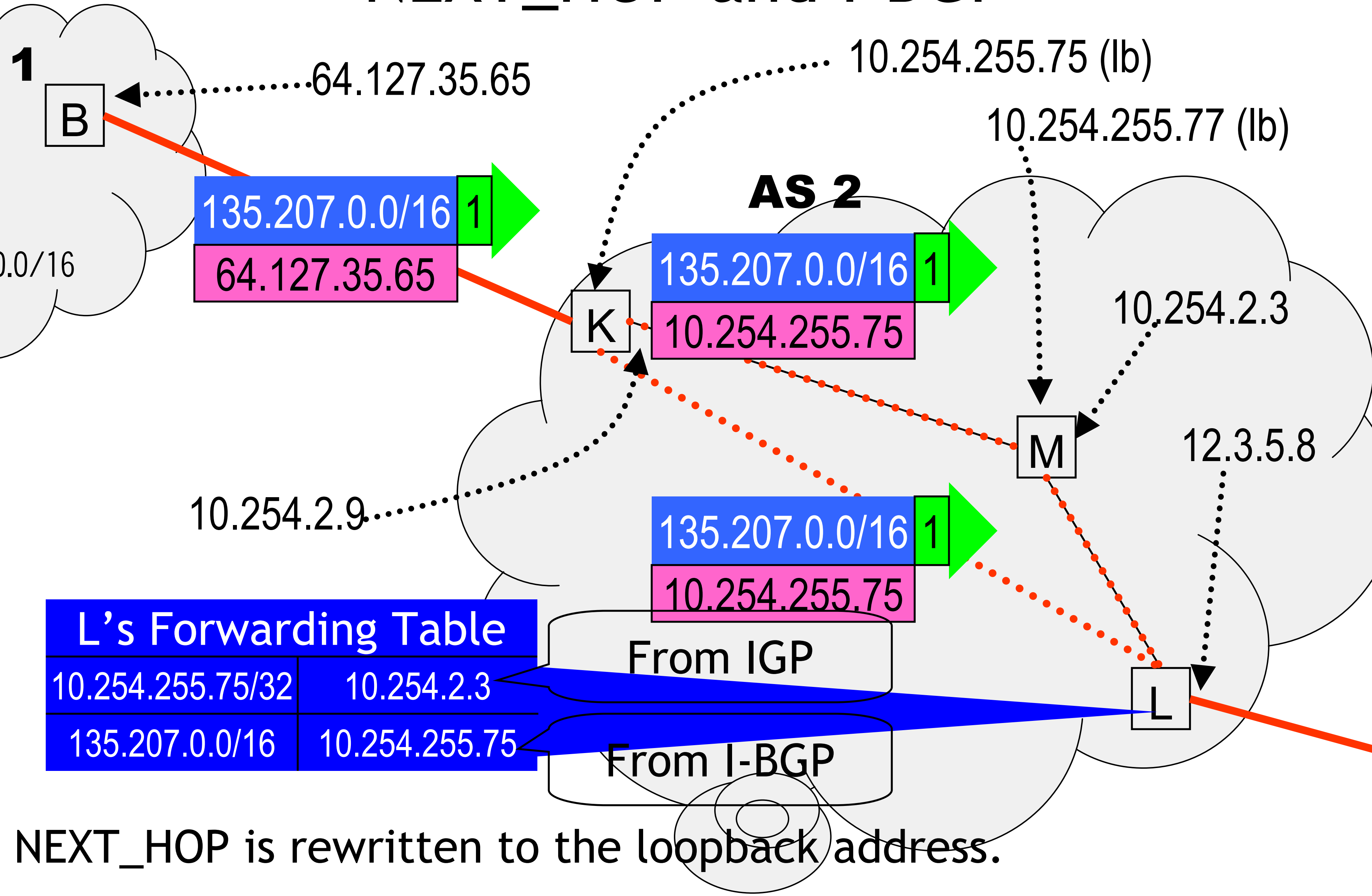


K's Forwarding Table	
135.207.0.0/16	64.127.35.65

L's Forwarding Table	
10.254.2.9/32	10.254.2.3
64.127.35.65/32	10.254.2.3
135.207.0.0/16	10.254.2.3

(If we are not using I-BGP).

NEXT_HOP and I-BGP

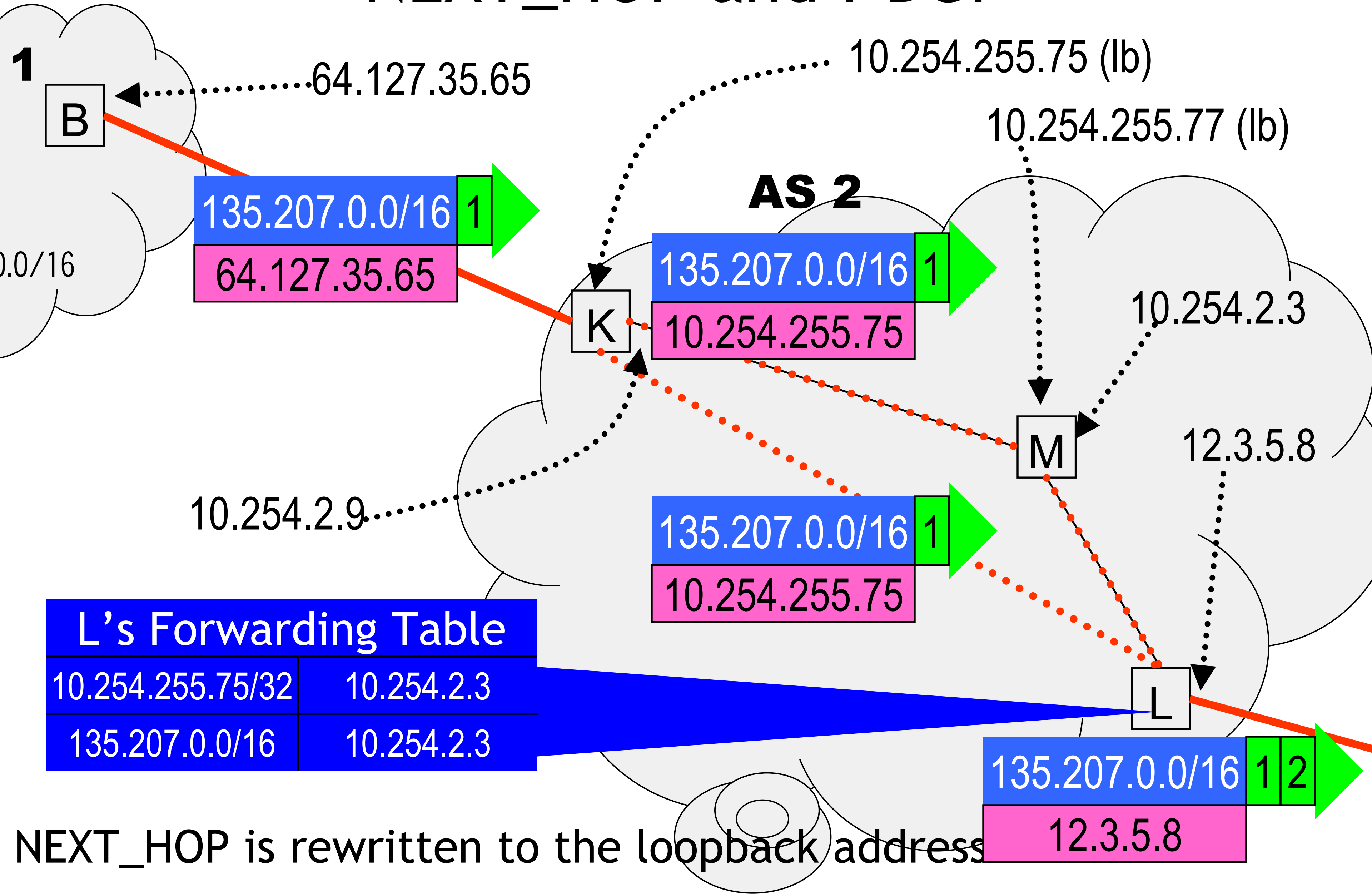


L’s Forwarding Table

10.254.255.75/32	10.254.2.3
135.207.0.0/16	10.254.255.75

NEXT_HOP is rewritten to the loopback address.

NEXT_HOP and I-BGP

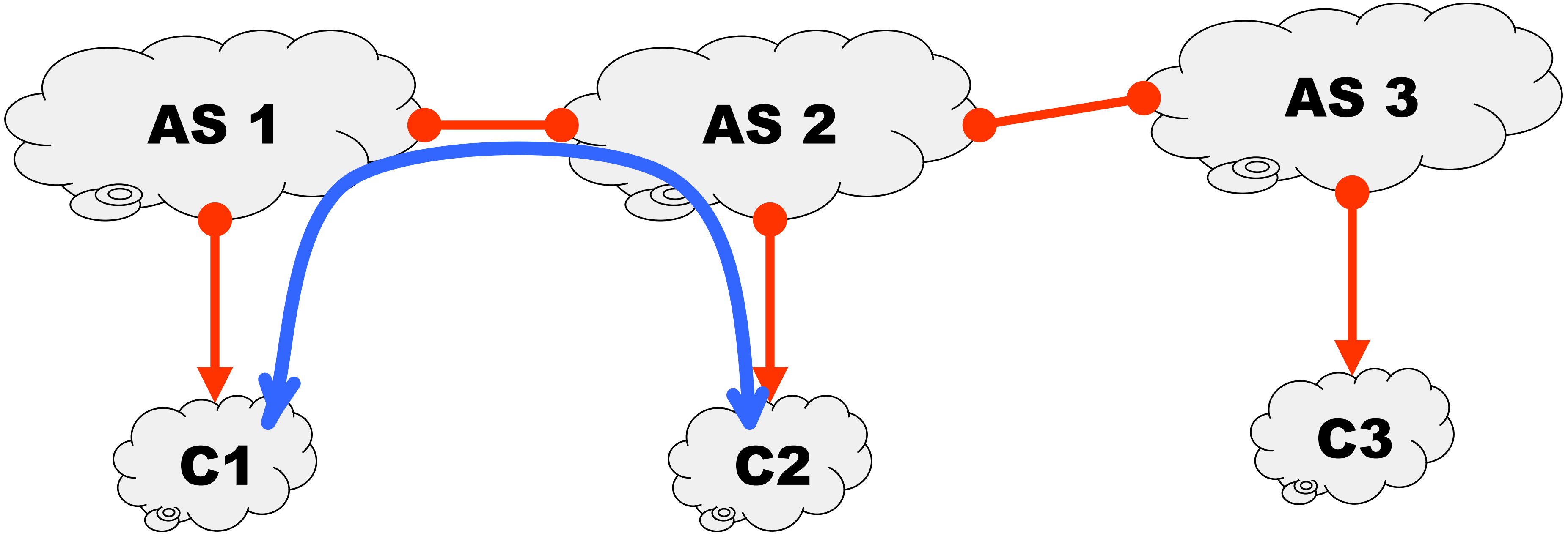


L’s Forwarding Table

10.254.255.75/32	10.254.2.3
135.207.0.0/16	10.254.2.3

NEXT_HOP is rewritten to the loopback address

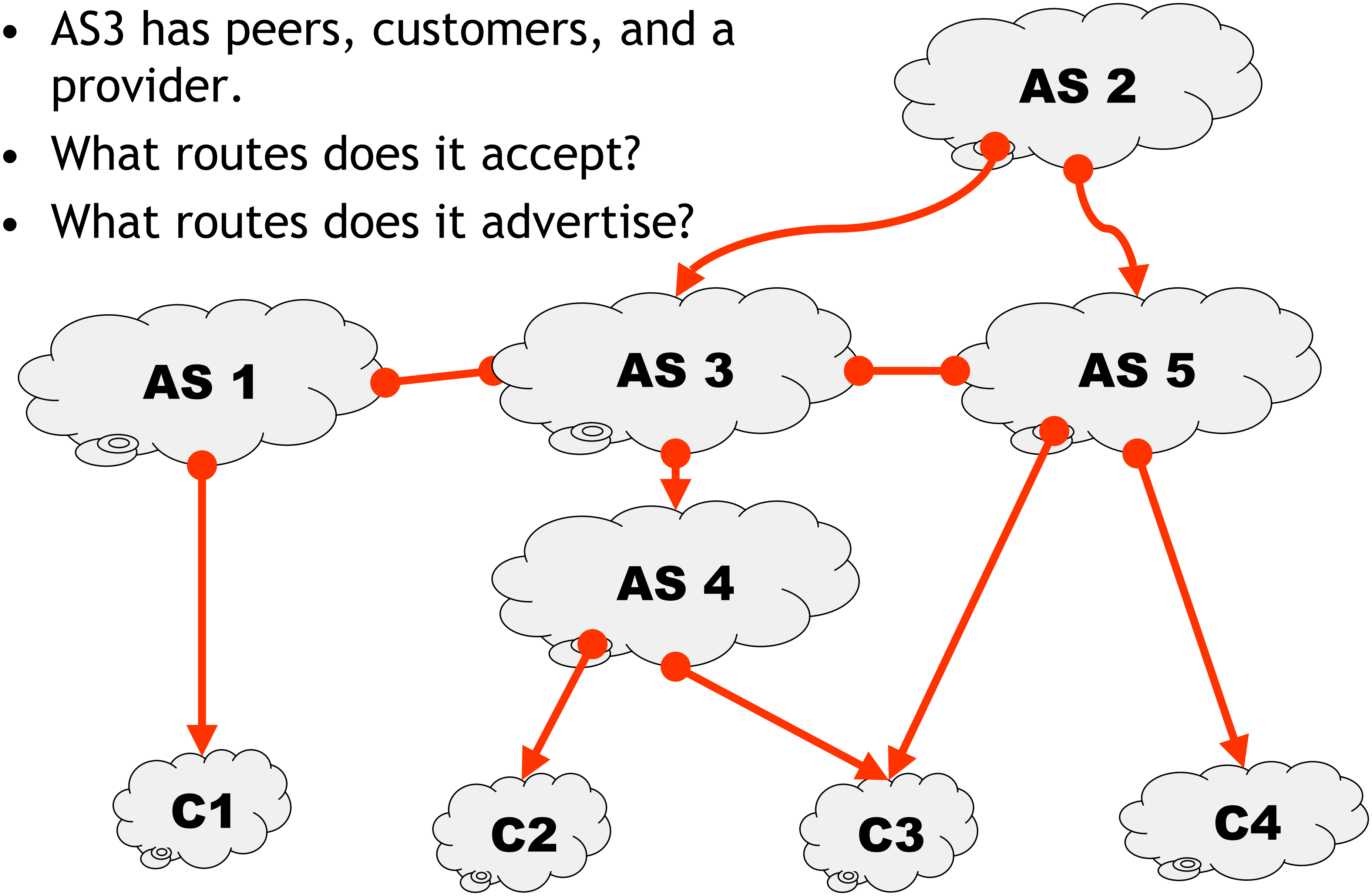
BGP Route Selection is about Policy



- AS1 exports C1’s prefix to AS2.
- AS1 accepts C2’s prefix from AS2.
- AS2 accepts C1’s prefix from AS1
- AS2 does not export any prefixes learned from AS3 to AS1.
- …

How Are Routes Chosen?

- AS3 has peers, customers, and a provider.
- What routes does it accept?
- What routes does it advertise?

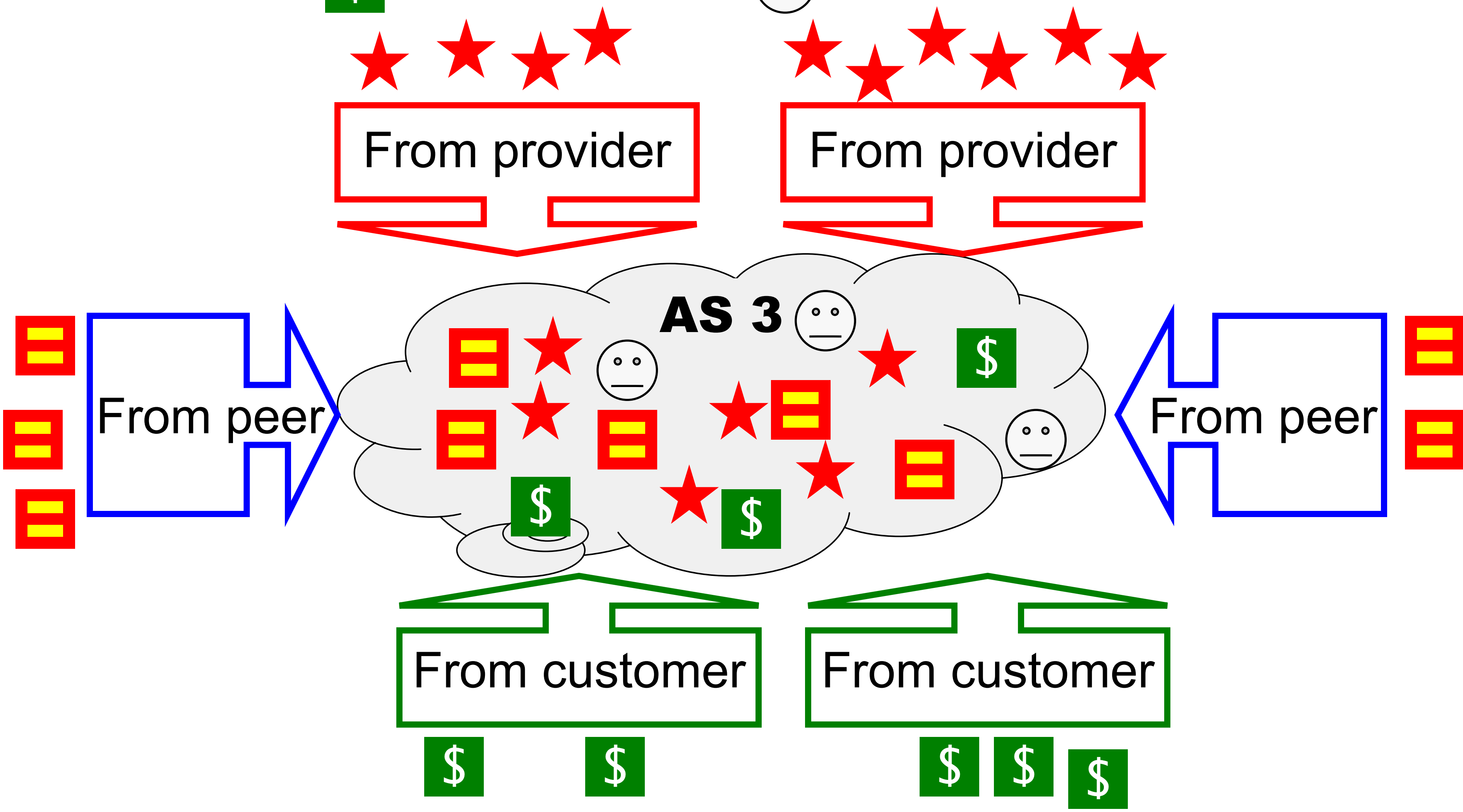


Customer-Provider & Peer-Peer Rltnshps

- Enforce transit relationships:
  - Filter outbound routes.
- Enforce order of route preference:
  - Customer $\succ$ Peer $\succ$ Provider.
  - More rules on route preference later.

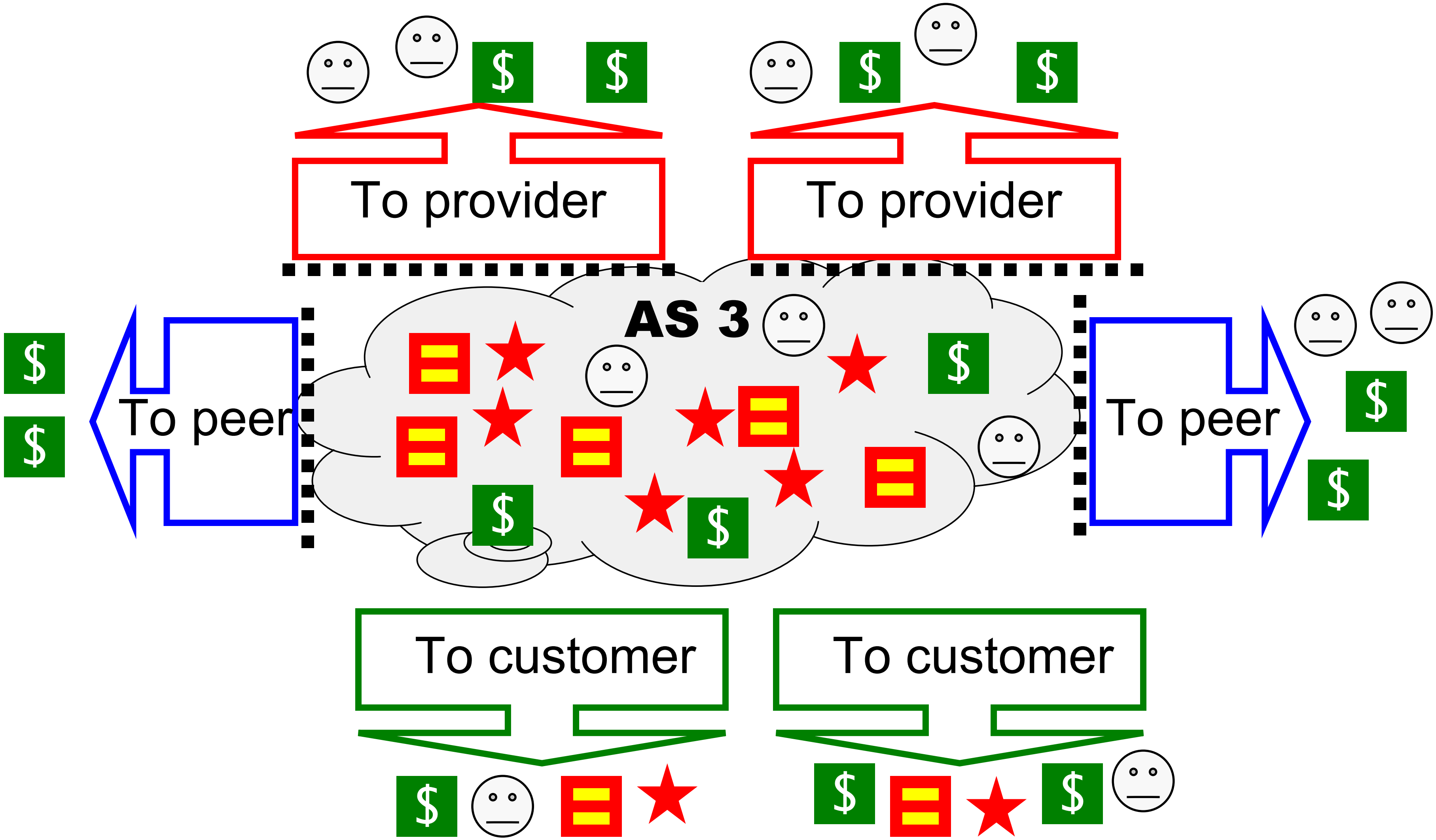
Imported Routes

Routes arrive from various sources: provider (★), peer (🟰), customer ($), and own IGP (😐).



Exported Routes

- Filters (▪▪▪▪▪) block peer and provider routes!



Picking Routes for Redistribution

- How does AS3 know which routes are customer/peer/provider/IGP?
- If AS3 were a single router, it could peek into Adj-RIB-In-x.
- But routes are redistributed with I-BGP.
  - Router that talks to provider is not router that talks to customer.
  - Routers could be (and were) configured with all of an AS's customer/peer/etc ASes to do output filtering.

Better answer:

- COMMUNITY attribute.

COMMUNITY

- Specified in RFC 1997.
- Encodes arbitrary properties.
  - E.g., all of customer's routes get a specific COMMUNITY.
- Much of the policy is specified using communities.

- Optional, Non-transitive. Type=8
- List of community values (length is multiple of 4).
  - Each prefix can belong to multiple communities.
- Each community value is 4 bytes: (*e.g.*, 7018:100)
  - 2 bytes ASN (by convention).
  - 2 bytes administratively defined (no predefined meaning).

COMMUNITY, cont’d

- 0x00000000 through 0x0000FFFF are reserved.
- 0xFFFF0000 through 0xFFFFFFFF are reserved.
- 0xFFFFFF01: NO_EXPORT
- 0xFFFFFF02: NO_ADVERTISE
- 0xFFFFFF03: NO_EXPORT_SUBCONFED

- Community values have local (intra-AS) meaning.
- Community values can also have meaning between two neighboring ASes (following bilateral agreement).

- Terminology: *Route Coloring*.

COMMUNITY Example

- When AS3 imports routes, it colors them with the appropriate community string.
  - From customers ($): 3:100.
  - From peers (=): 3:200.
  - From providers (★): 3:300.

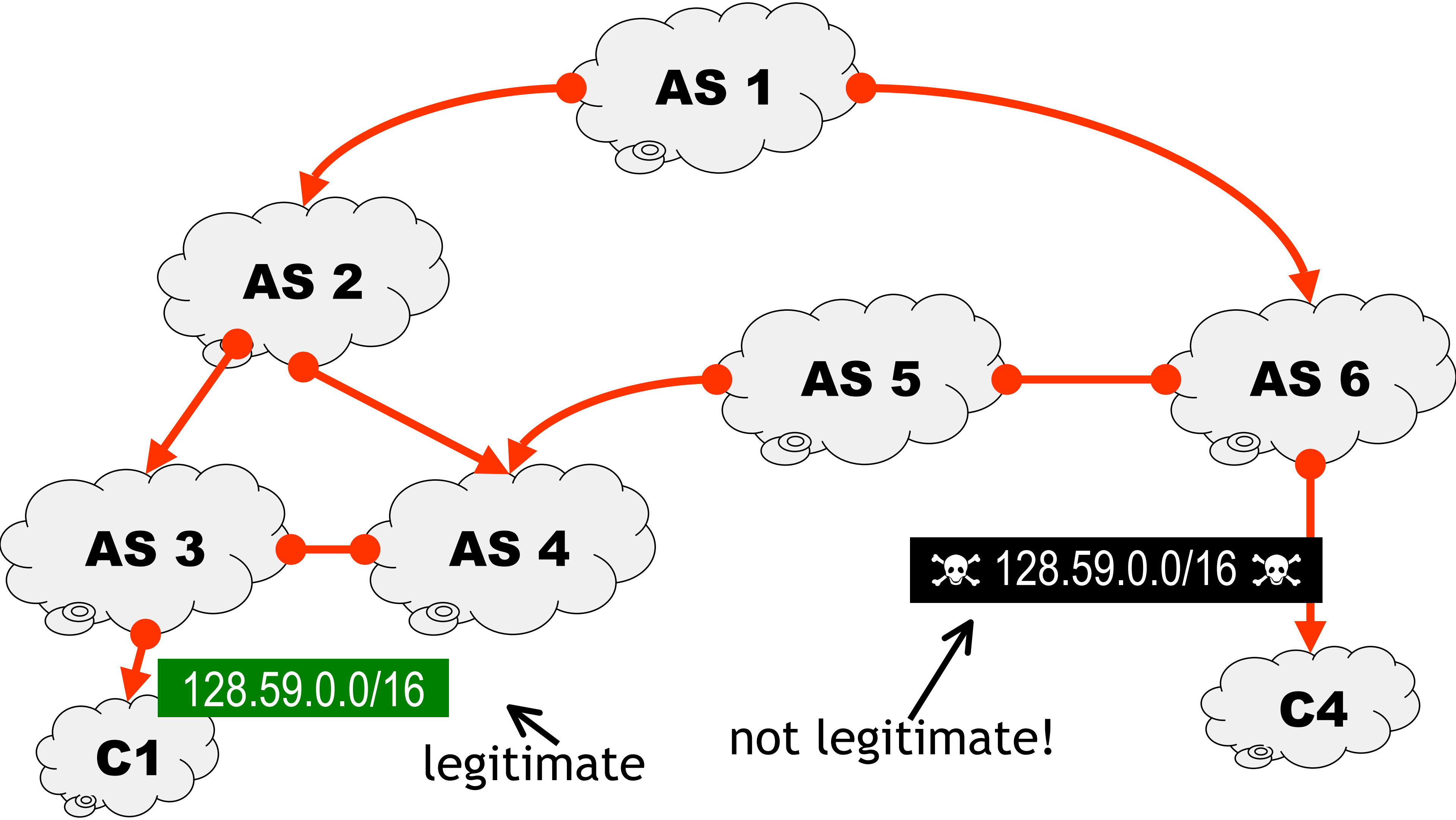
- When AS3 exports routes, it picks them according to their community string.
  - To customers: 3:100, 3:200, 3:300
  - To peers: 3:100
  - To providers: 3:100

Martians (or bogons)

- Some prefixes should not be advertised.
  - Some should not even appear!
  - Default (0.0.0.0/0) routes are never advertised.
  - Site-local (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16).
  - Link-local (169.254.0.0/16).
  - Loopback (127.0.0.0/8).
  - IANA-reserved (128.0.0.0/16, 192.0.0.0/24, etc.).
  - Test networks (192.0.2.0/24, etc.).
  - Class D and E (224.0.0.0/3).
  - Unallocated space.
    - Careful with that!
- Routes to martians are filtered on input.
  - Not that they should ever have been advertised!

Black Holes Are Out of Sight

- If another AS advertises one of our prefixes, bad things happen:

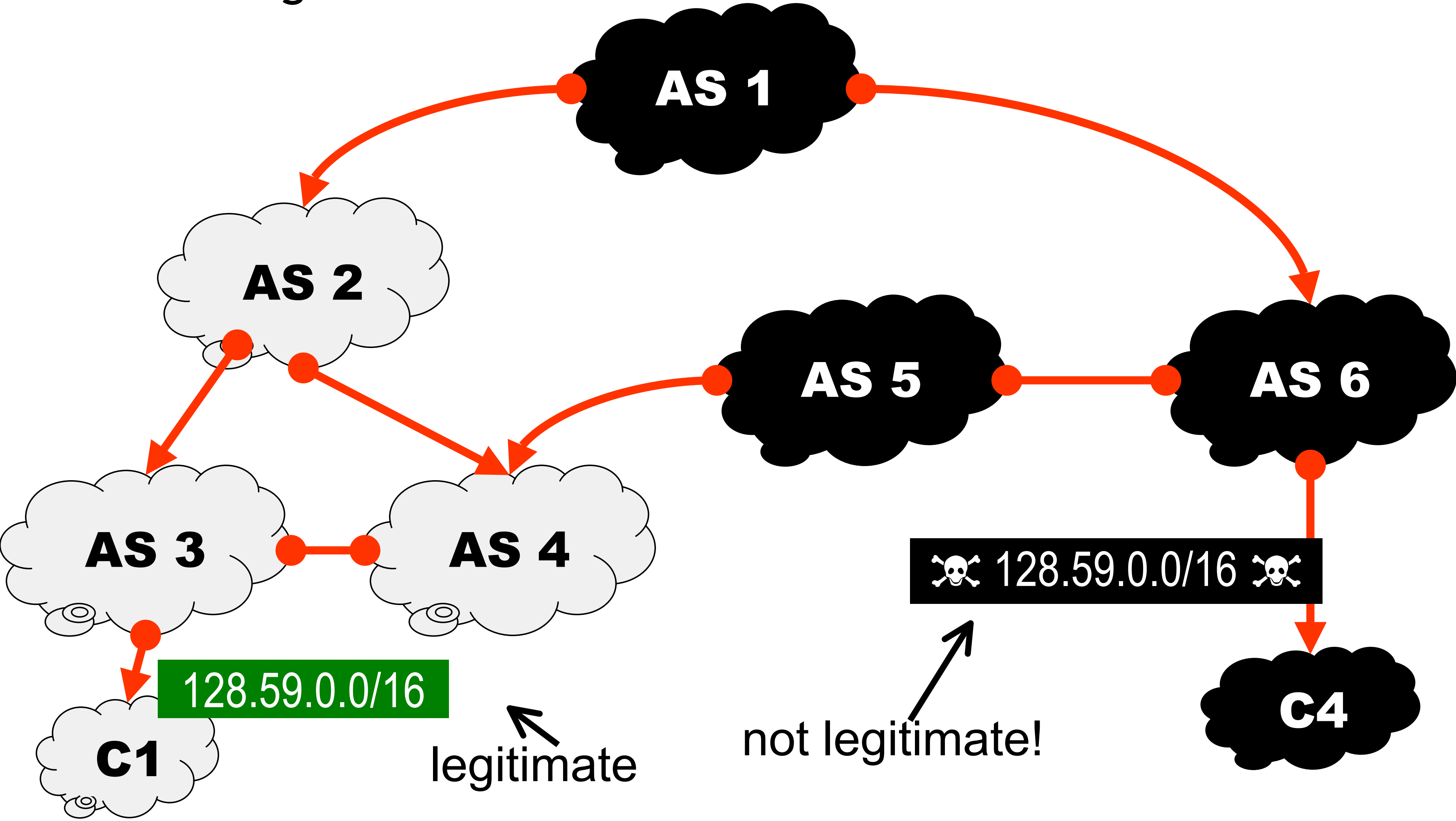


legitimate

not legitimate!

Black Holes Are Out of Sight

- Our prefix becomes unreachable from the part of the net believing C4’s announcement.

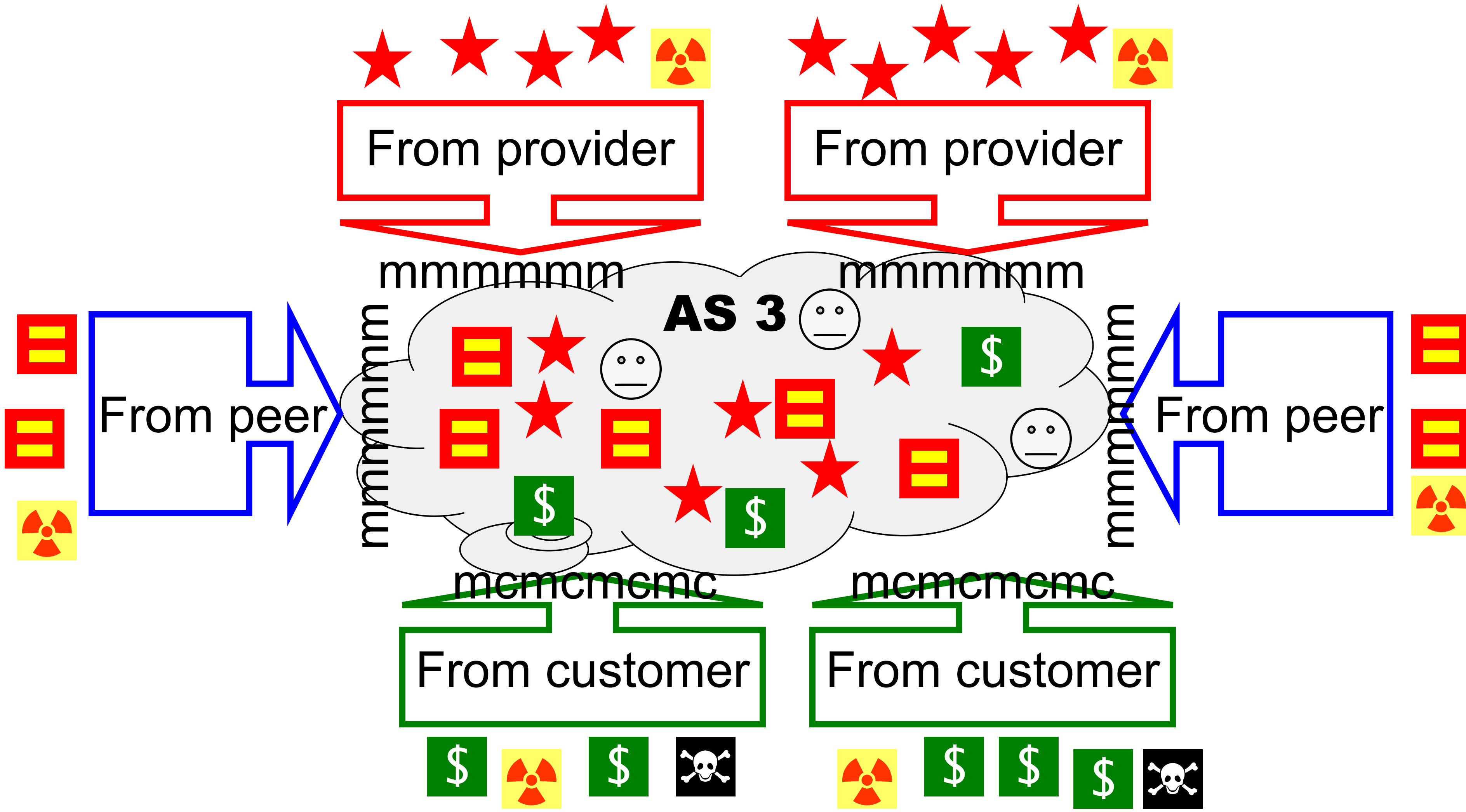


Preventing Bad Routing

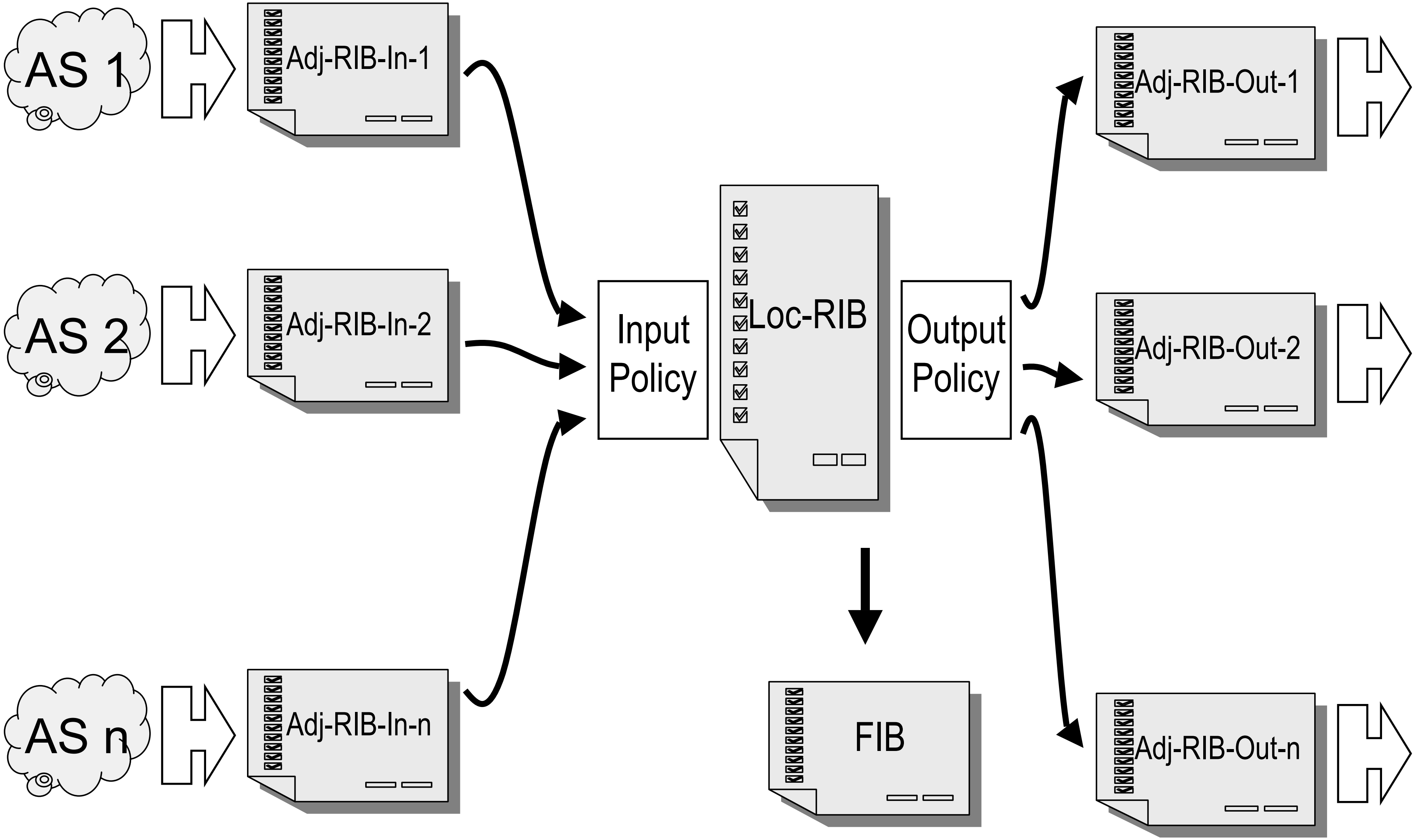
- Preventing black holes:
  - Only accept customer routes advertising customer’s prefixes.
  - AS6 should only accept C4’s real prefixes, not anything C4 advertises.
- Filter out Martians:
  - Private address space is sometimes used for intra-AS management.
    - Should not accept routes for it!
  - Be a good citizen, do not leak martians!

Imported Routes, revisited

When importing, filter martians (☢) and potentially bad customer routes (☠). Also, drop looping AS_PATH.



In/Out Route Processing



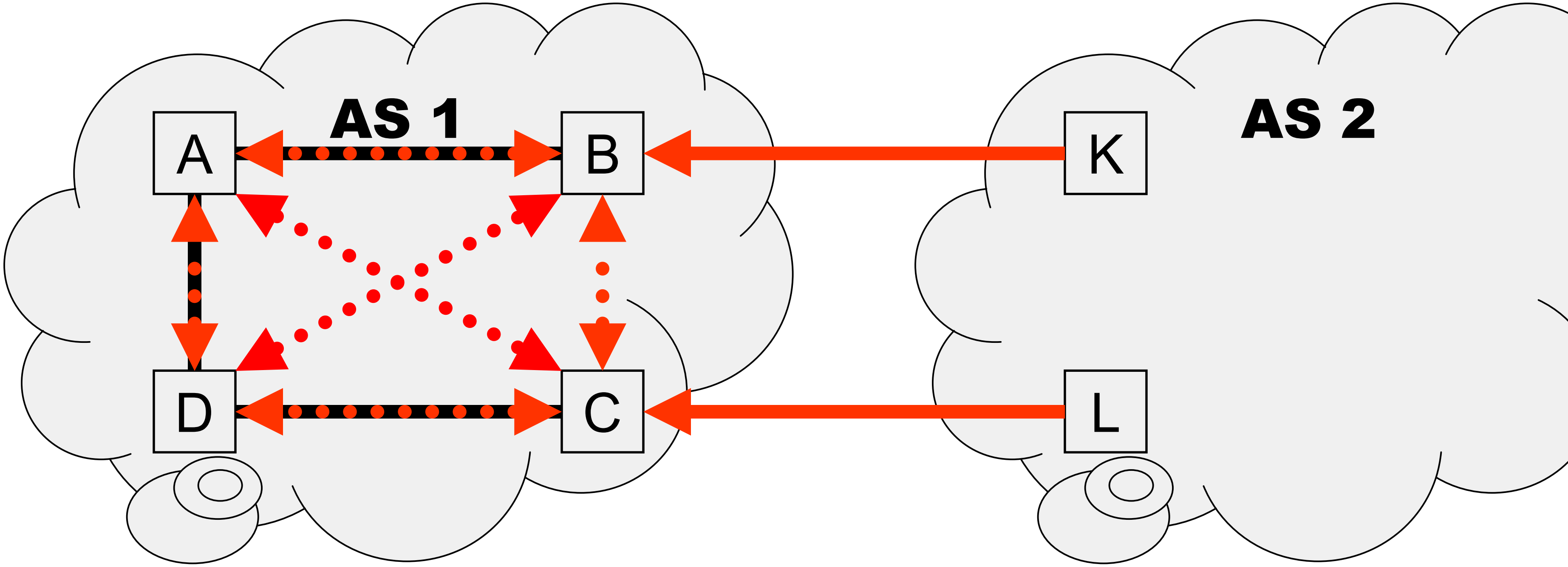
Input Policy

- Apply input filtering.
  - Routes that are dropped here are not used internally.
  - Nor are they advertised.
  - They are dead!
- Tweak attributes:
  - Set LOCAL_PREF, add COMMUNITY
- Select best route.
  - Based on Path Attributes.
- Create Route table.
- Populate Forwarding table.

Best Route Selection

- If NEXT_HOP inaccessible, route is dropped.
- [cisco only] prefer path with highest *weight*.
- Select route with highest LOCAL_PREF.
- Prefer shortest AS_PATH.
- Prefer lowest origin (IGP < EGP < INCOMPLETE).
- If routes received from same AS (or **bgp always-compare-med** enabled), and MED enabled, prefer lowest MED.
- Prefer E-BGP paths over I-BGP paths.
- Prefer shortest IGP path to NEXT_HOP.
- Use lowest router ID as tie-breaker.
  - Some implementations use first installed route instead.

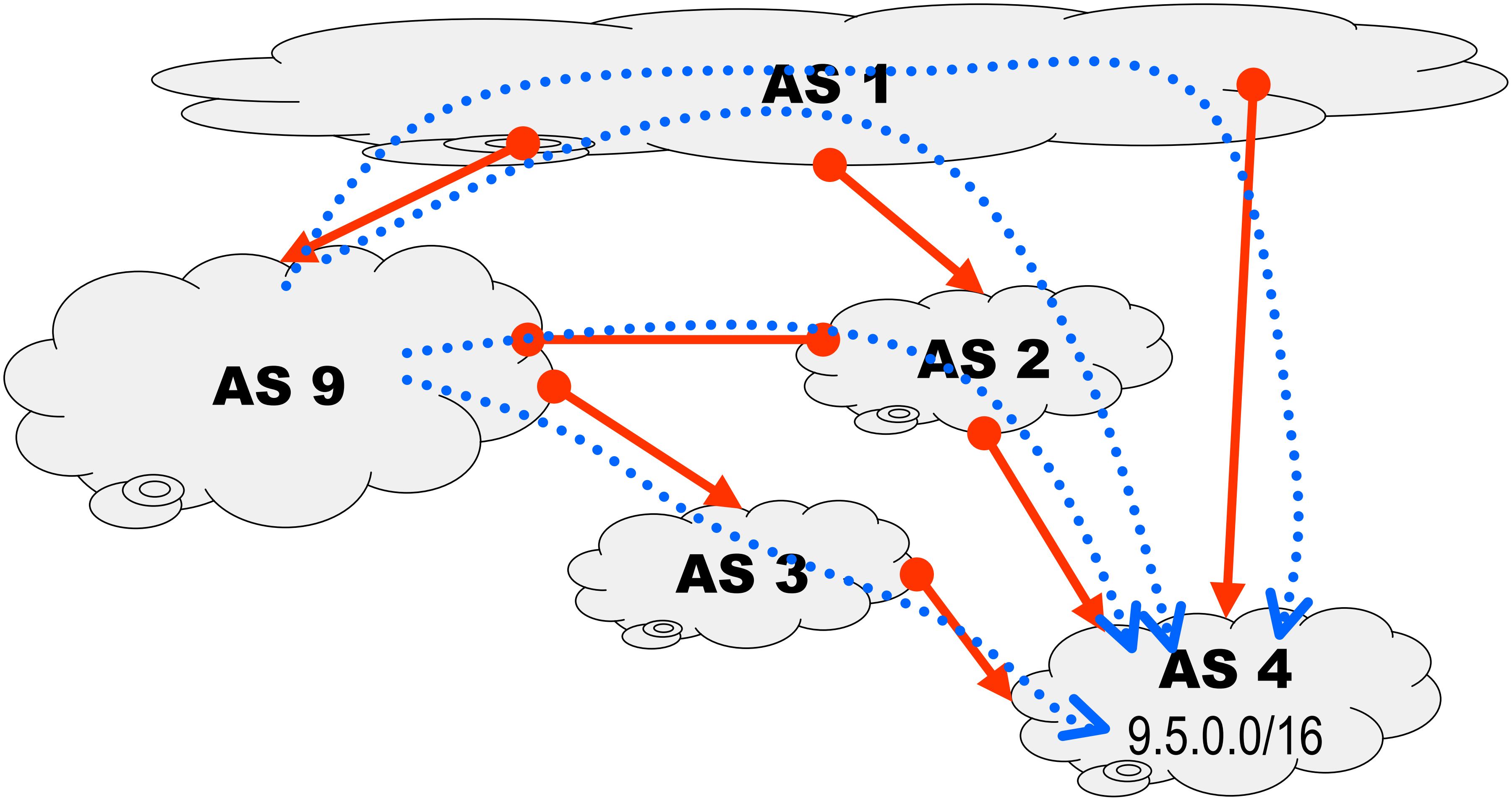
Why prefer E-BGP over I-BGP?



- B learns route to AS2 over E-BGP from K.
- B learns route to AS2 over I-BGP from C
  - (who learned it from L).
- Same local pref, as_path length, origin, etc.
- Obviously should use K!

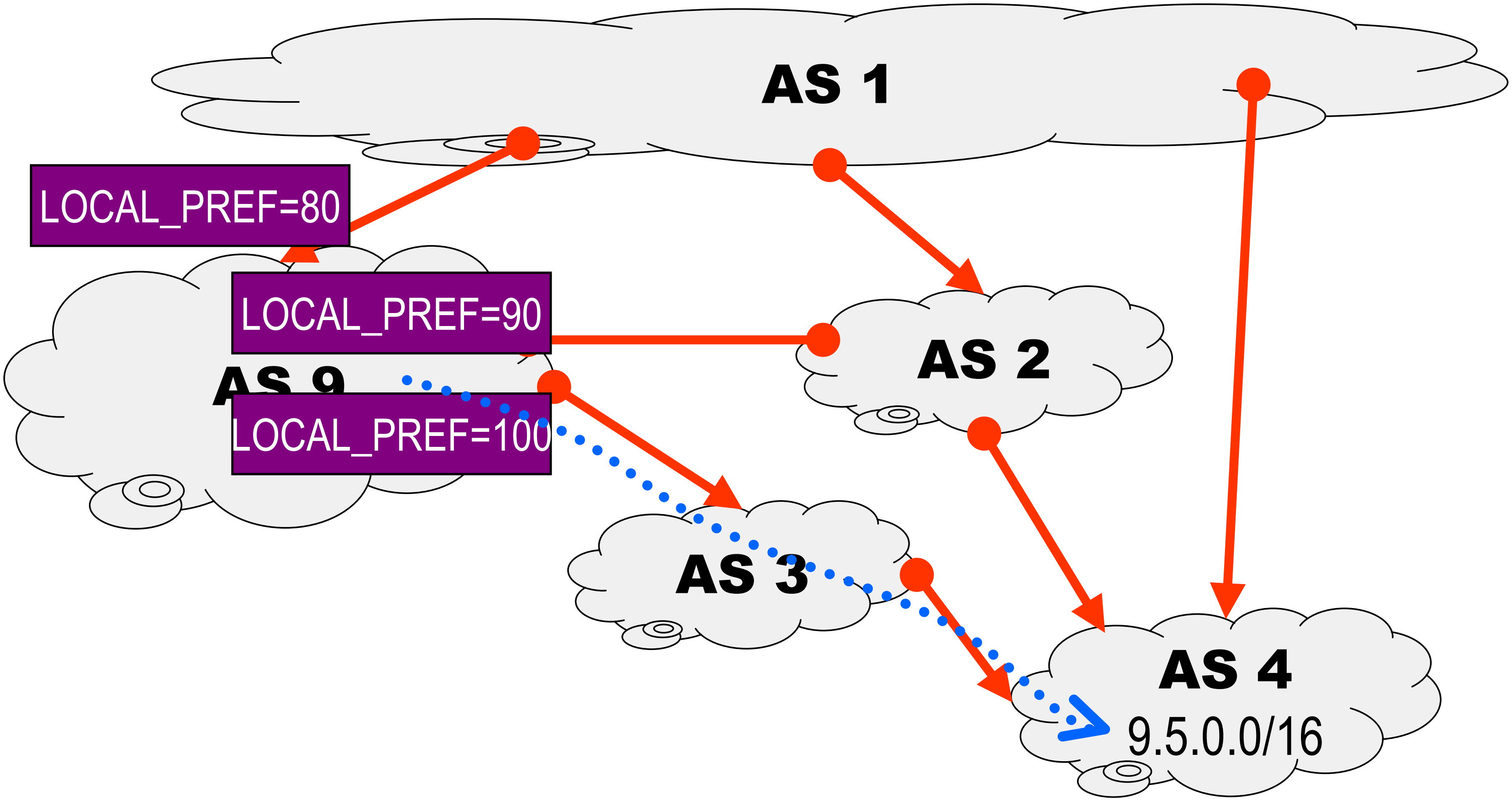
What is the Best Route?

Which of the four possible routes will 9.5.1.2 take to get to AS4?



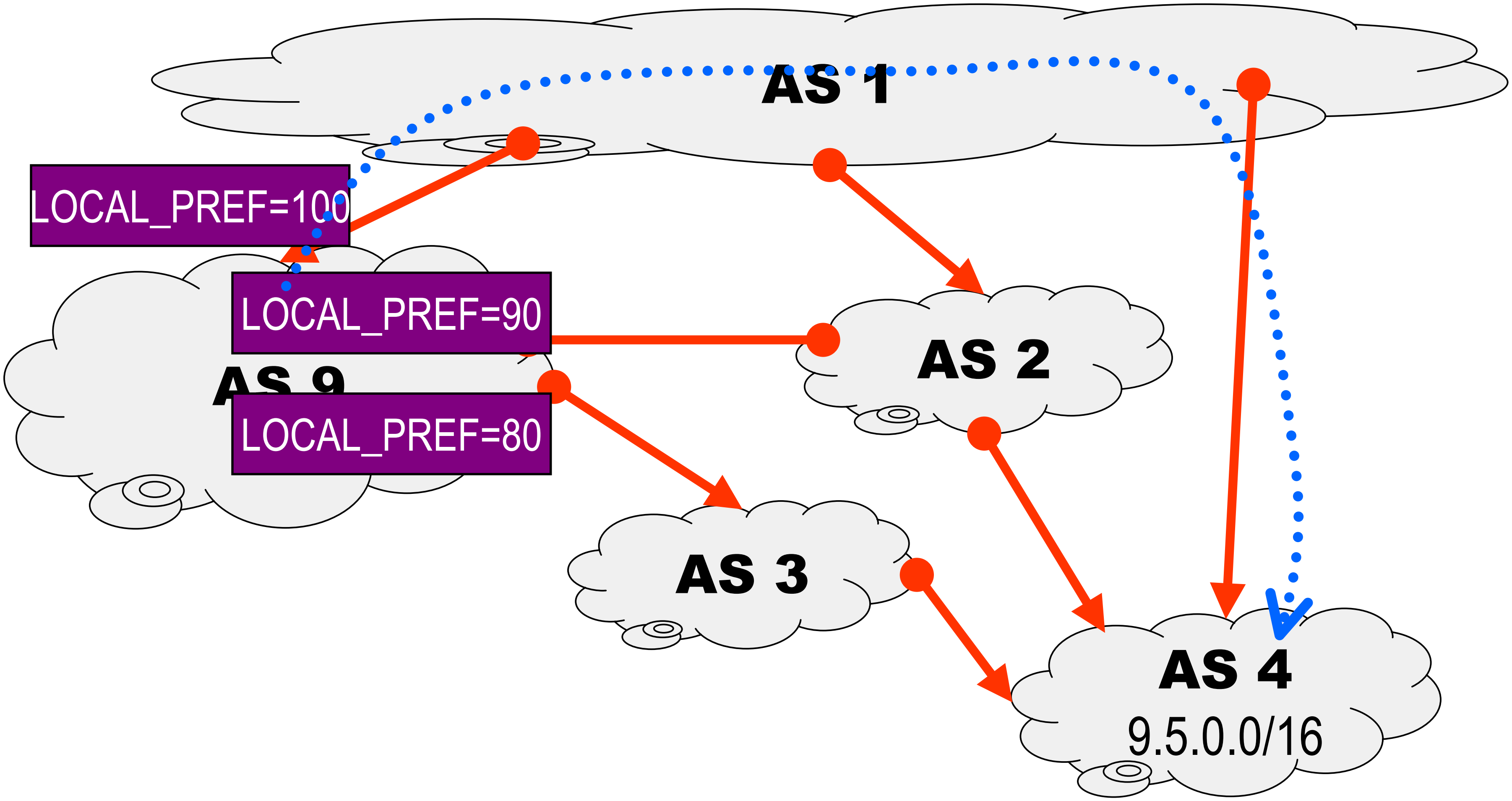
What is the Best Route?

- LOCAL_PREF to the rescue!



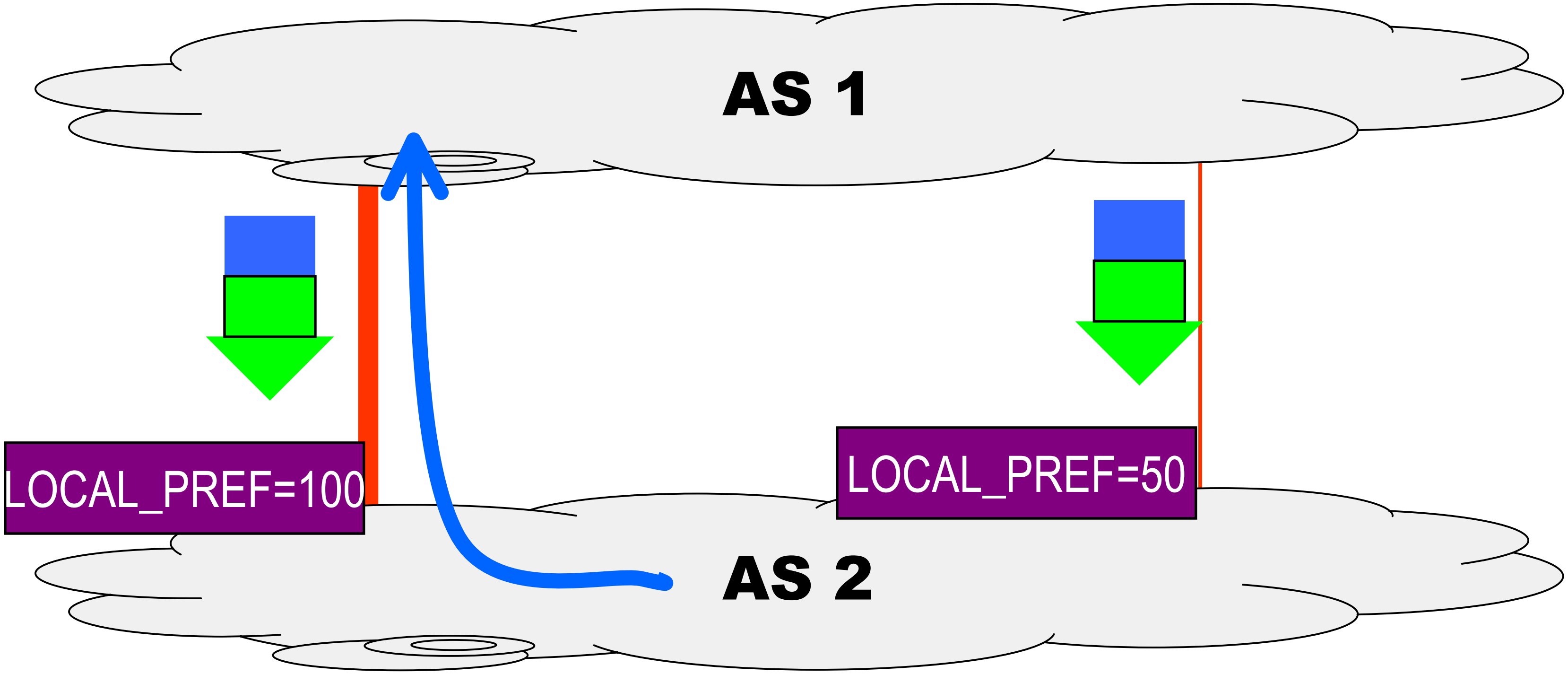
Alternatively…

- Now shortest AS_PATH takes effect!



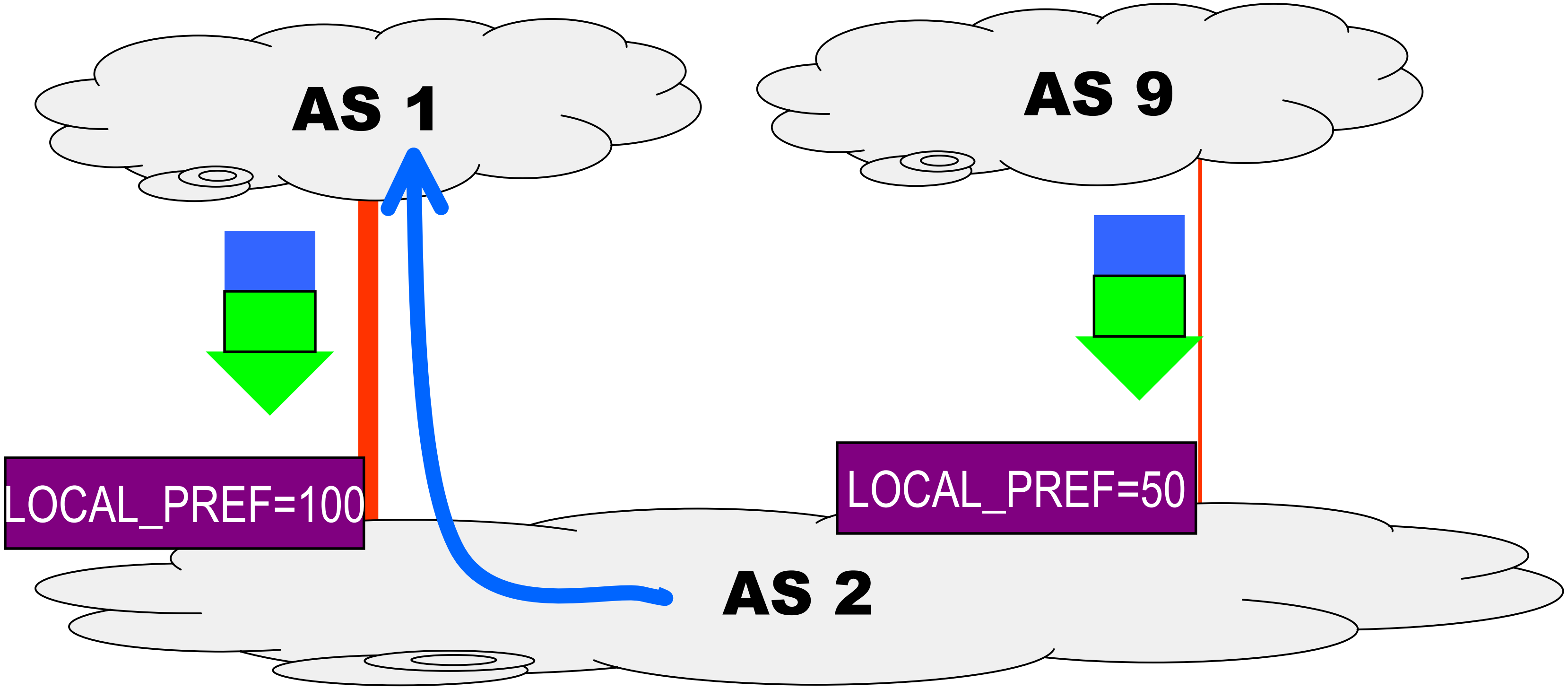
Backup Links (outbound traffic)

- Set higher local pref on primary link on all routes from AS1.
- Forces all traffic to take primary unless it is down.



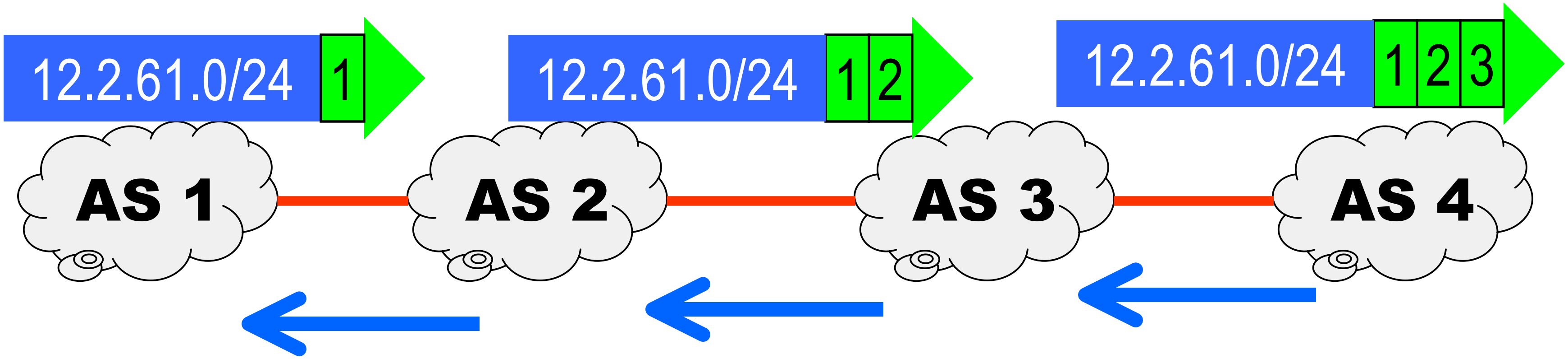
Multihomed Backups (outbound traffic)

- Same idea.

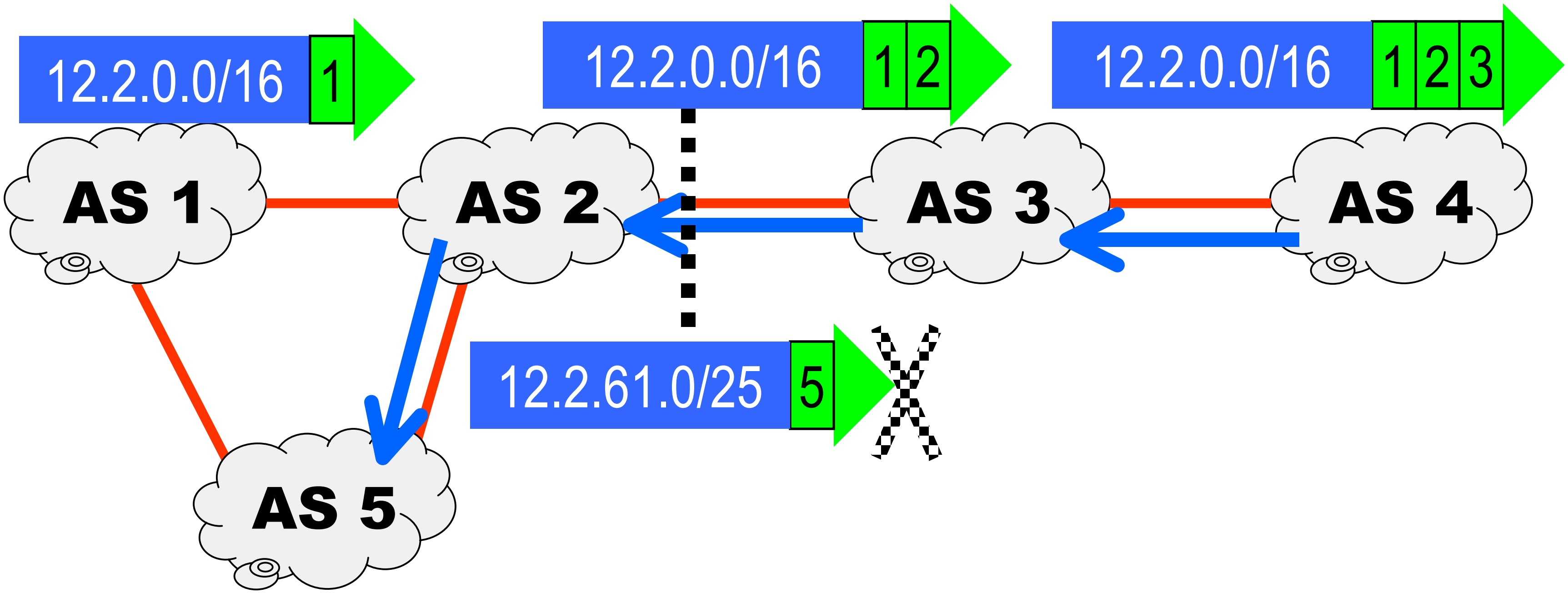


Back to AS_PATH

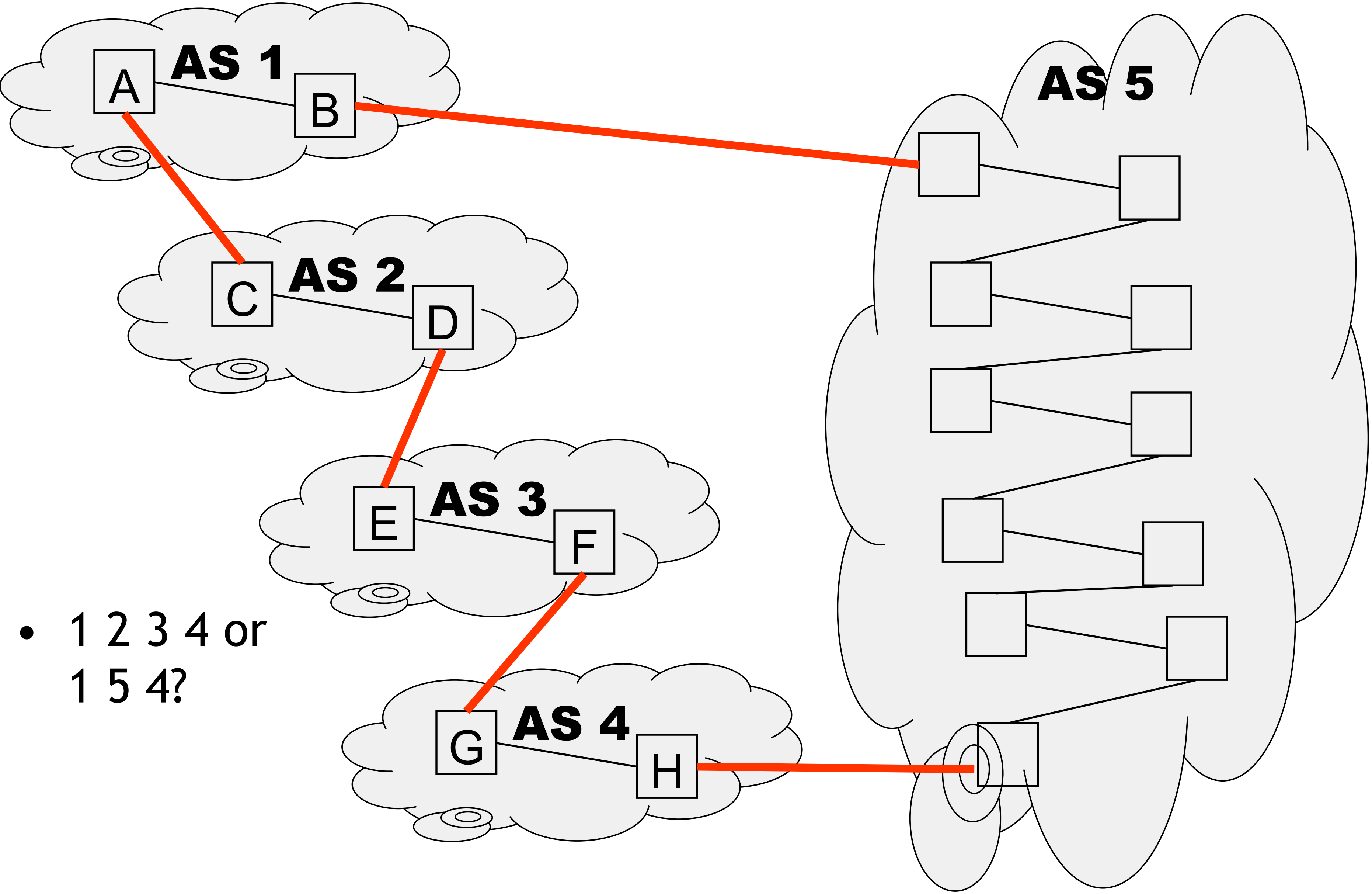
- Traffic often follows reverse of AS_PATH:



- But it might not!
- AS2 filters prefixes longer than /24.
- Packet to 12.2.61.19 actually makes it to AS5.



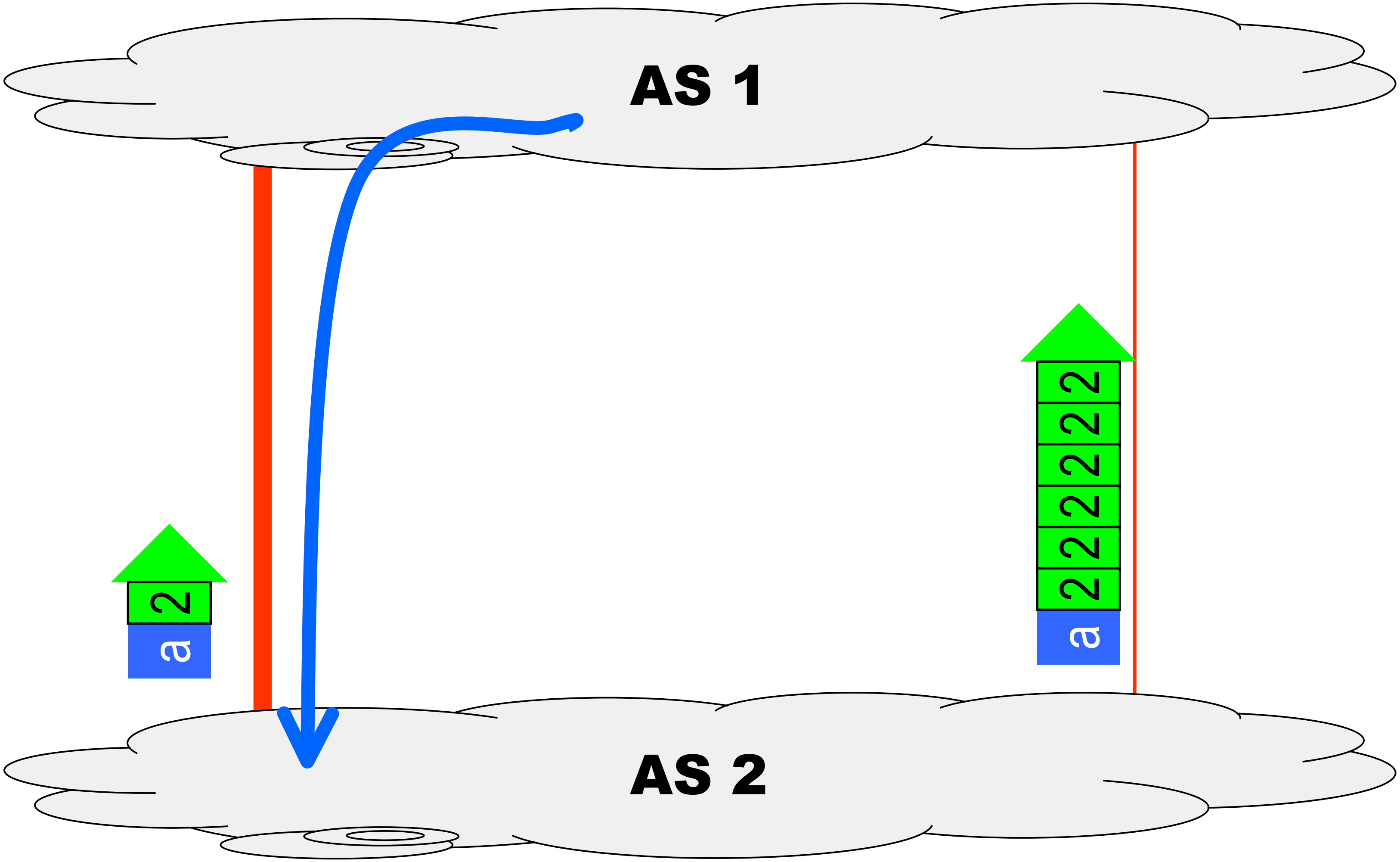
Shortest AS_PATH?



- 1 2 3 4 or
  1 5 4?

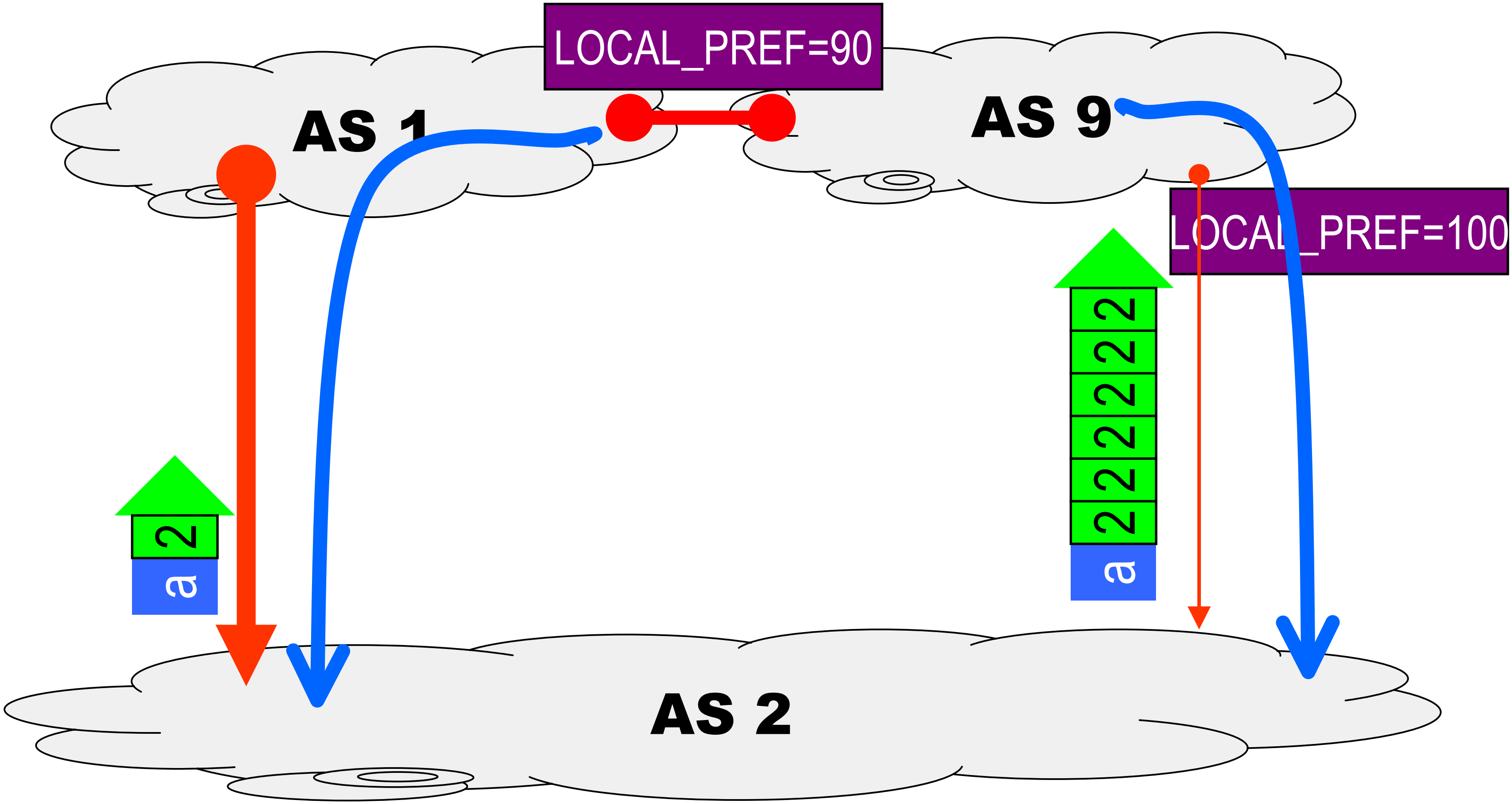
Backup Links (inbound traffic)

- Hack: AS_PATH padding.



Backup Links (inbound traffic)

- AS_PATH padding does not shut off all traffic.
- AS 9 has higher LOCAL_PREF for customer routes.
- Some traffic from AS9 still flows through the backup link.

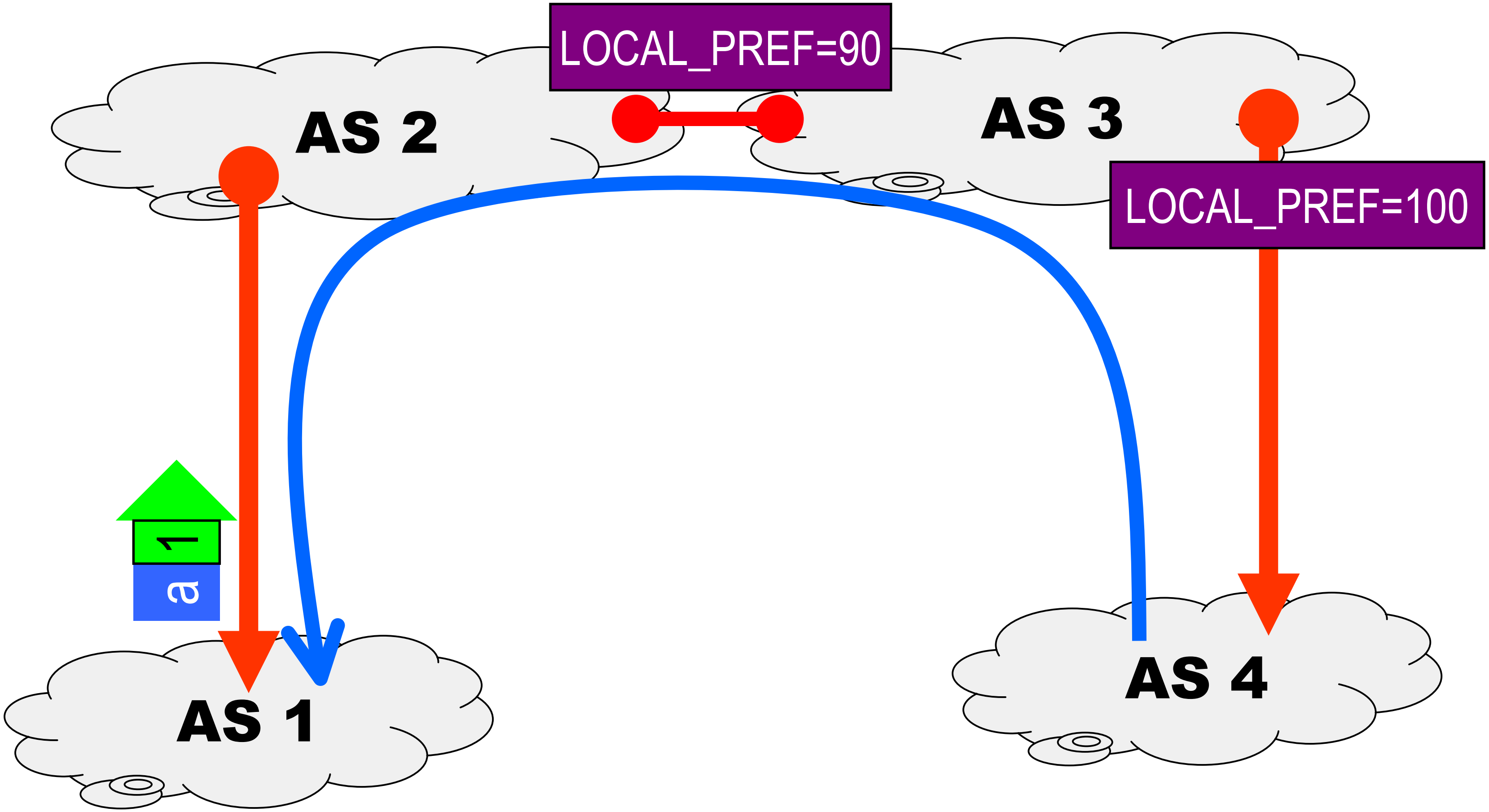


Backup links (inbound traffic)

- COMMUNITY to the rescue!
- AS9 has LOCAL_PREF = 100 for customer and 90 for peer.
- AS9 has the following import policy:
  - If 9:90 in community, set local_pref to 90.
  - If 9:80 in community, set local_pref to 80.
  - If 9:70 in community, set local_pref to 70.
- AS2 advertises its routes (over the backup link to AS9) with community 9:70.
- Now peer has higher local pref and traffic flows as intended!

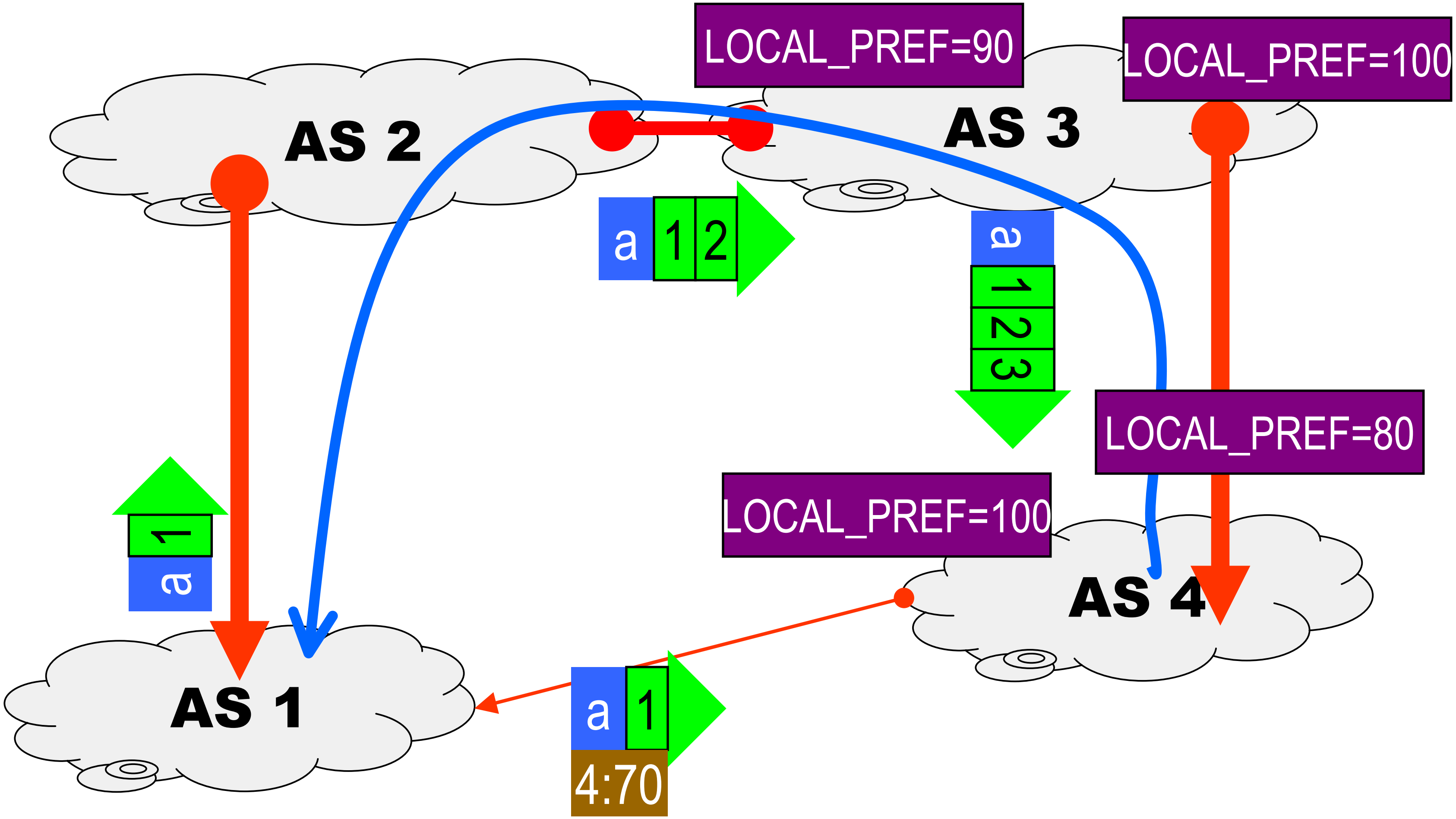
Policy Interaction

- Example: backup route with community hack.
- AS1 advertises prefix a over its (only) link.



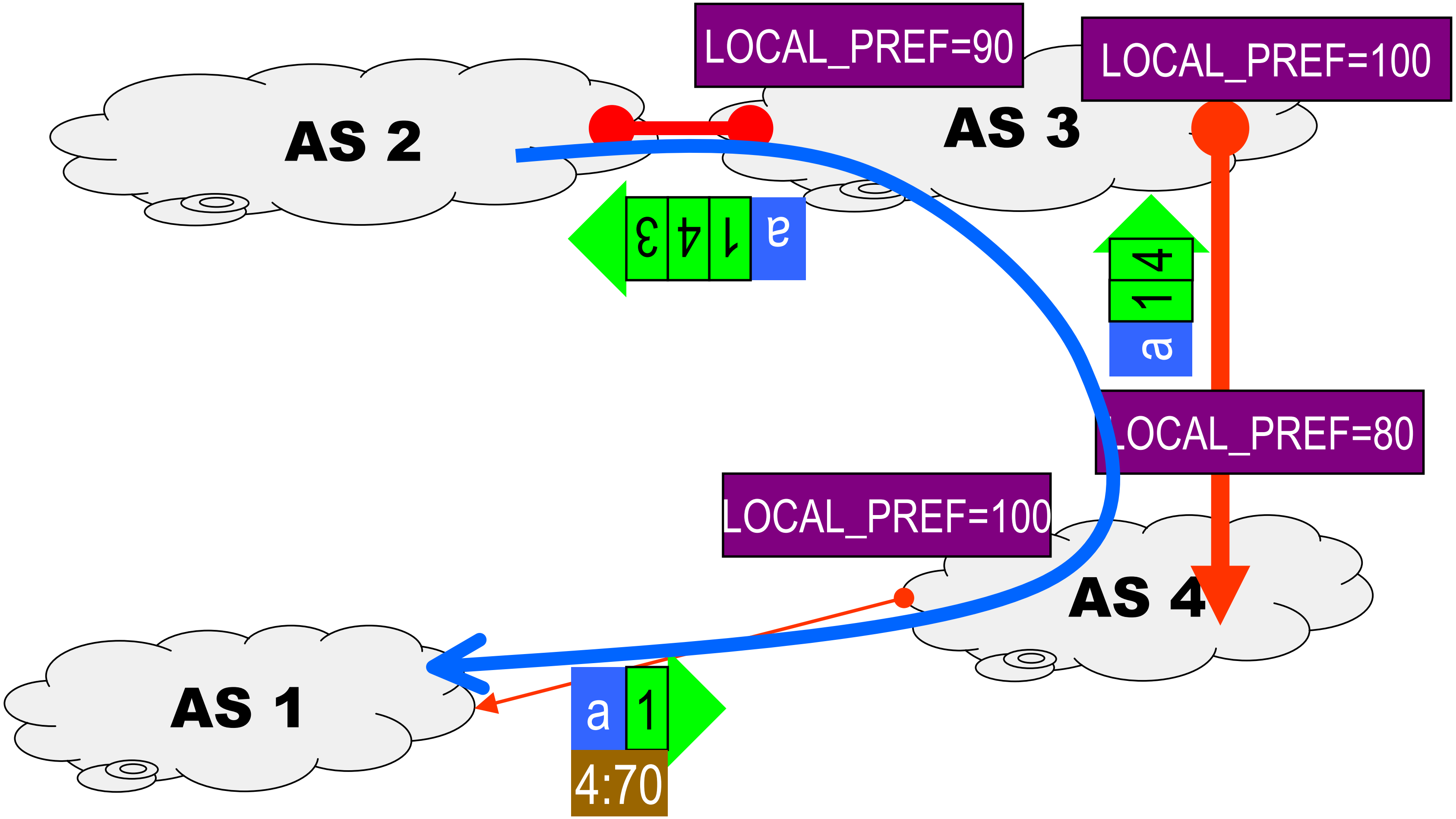
Policy Interaction cont’d

- Backup link gets installed, AS1 advertises community 4:70.
- AS4 still prefers route via AS3 (highest local_pref).



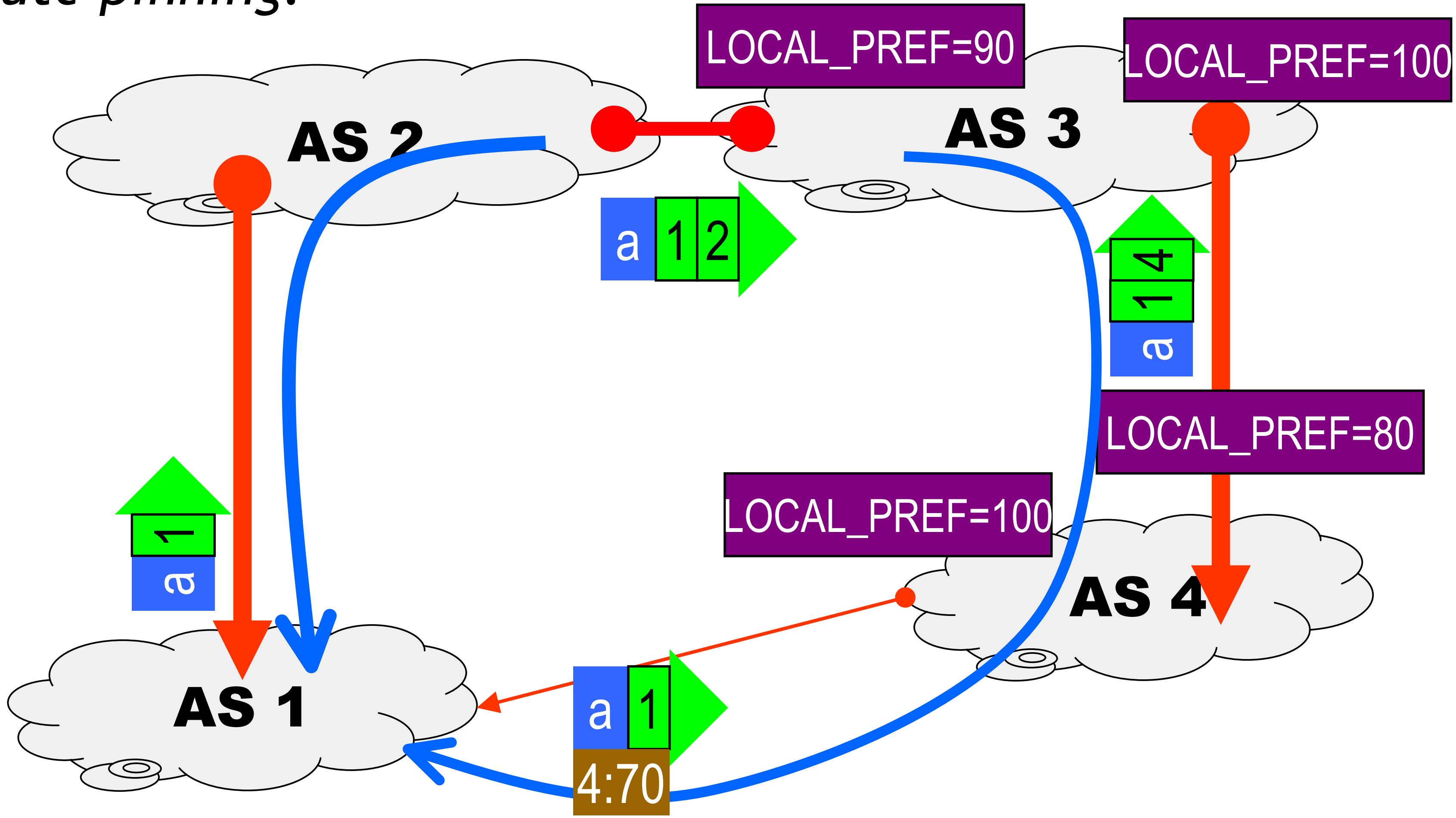
Backhoe Severs Primary Link

- AS2 withdraws route to a.
- Backup link takes over.



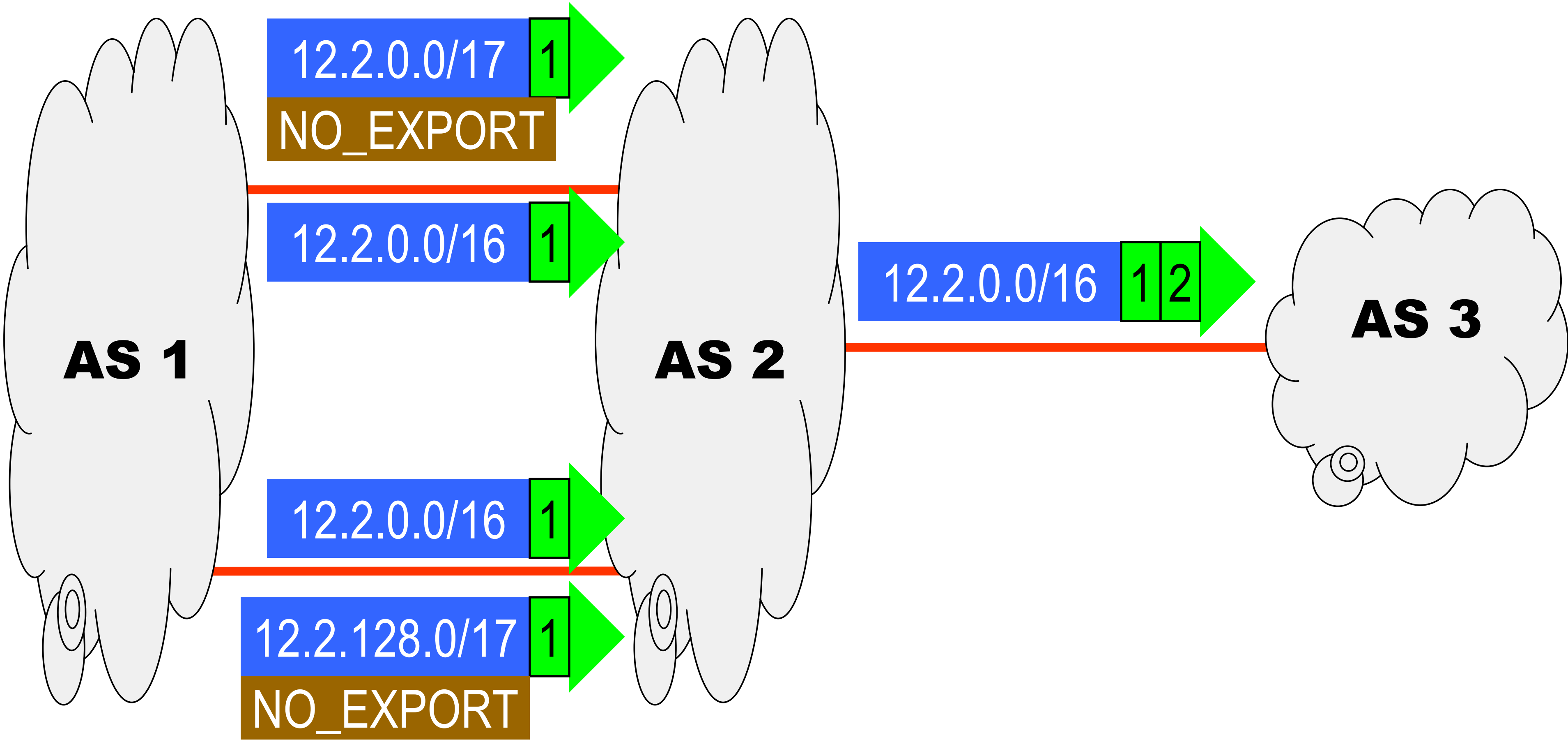
Primary link restored

- AS4 is still advertising route to AS1.
- Route from AS2 has lower local pref, gets ignored!
- *Route pinning.*



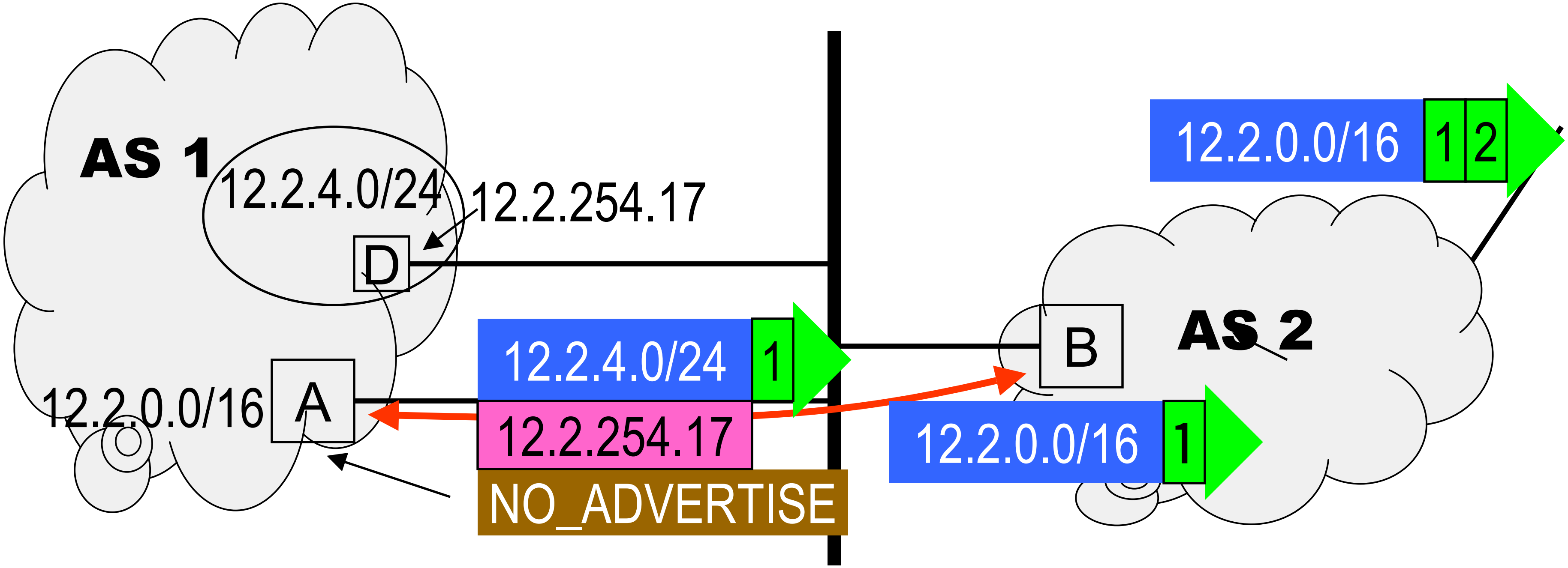
NO_EXPORT (0xFFFFFF01)

- Received routes with the NO_EXPORT community are not re-advertised beyond the receiving AS.



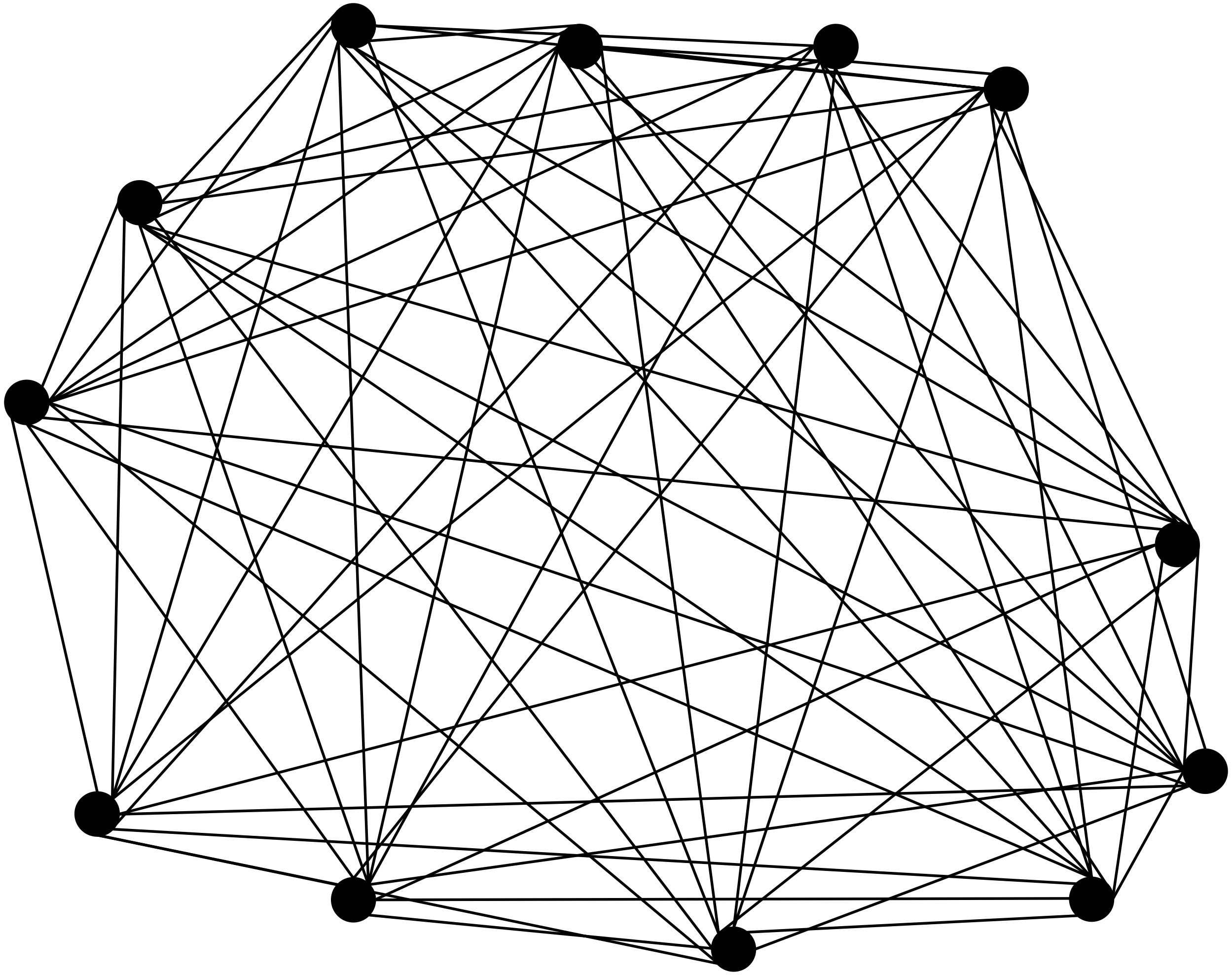
NO_ADVERTISE (0xFFFFFF02)

- Used in conjunction with the third-party NEXT_HOP.
- Most of AS1 is behind A.
- D does not speak BGP.
- AS1 advertises 12.2.4.0/24 with the NO_ADVERTISE.
- B uses D to forward packets to 12.2.4.0/24.
- This fine structure is not exported beyond AS2.



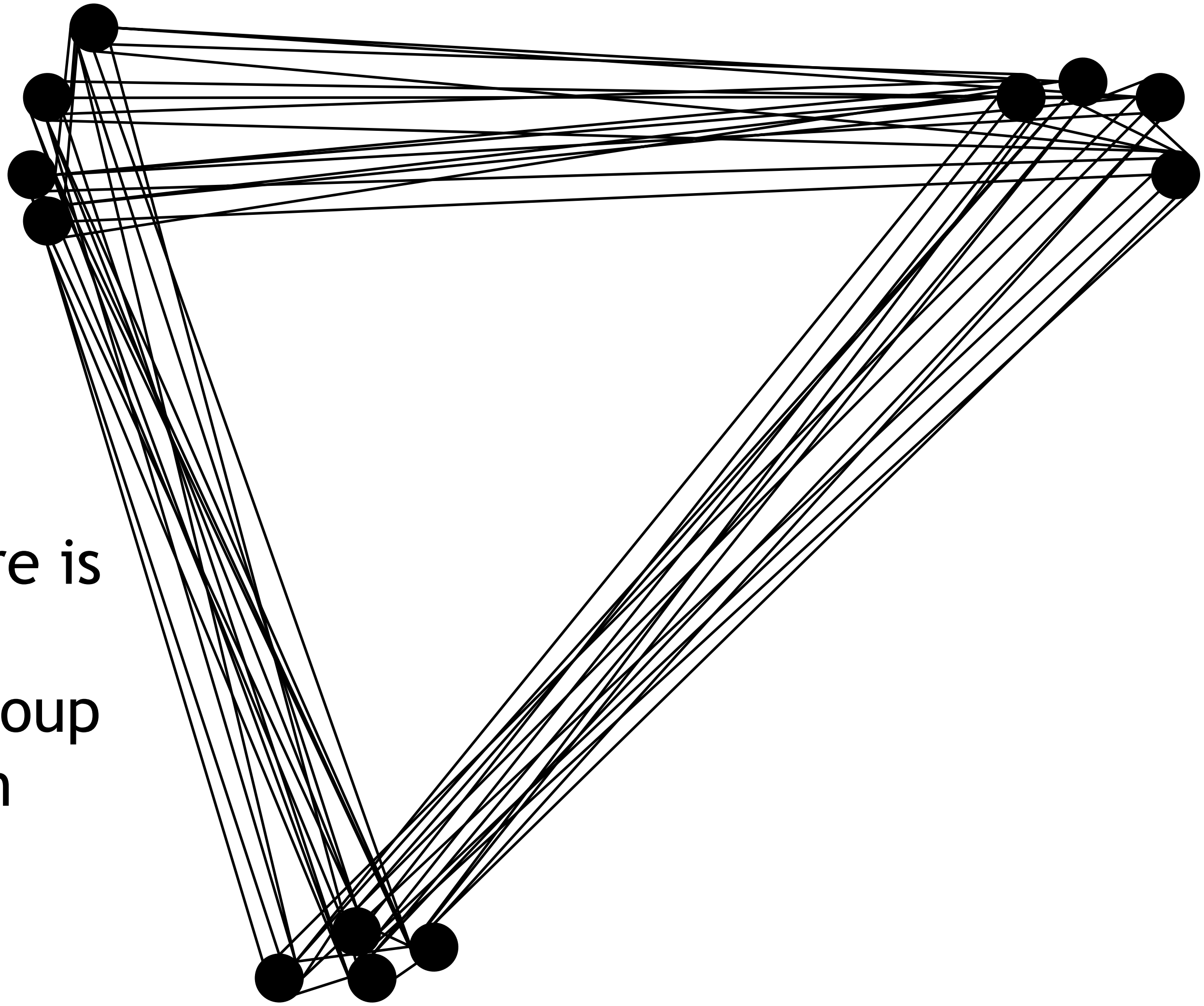
I-BGP Scaling

- I-BGP peering sessions can be wasteful of resources.
  (Lines represent I-BGP sessions, NOT physical links!)



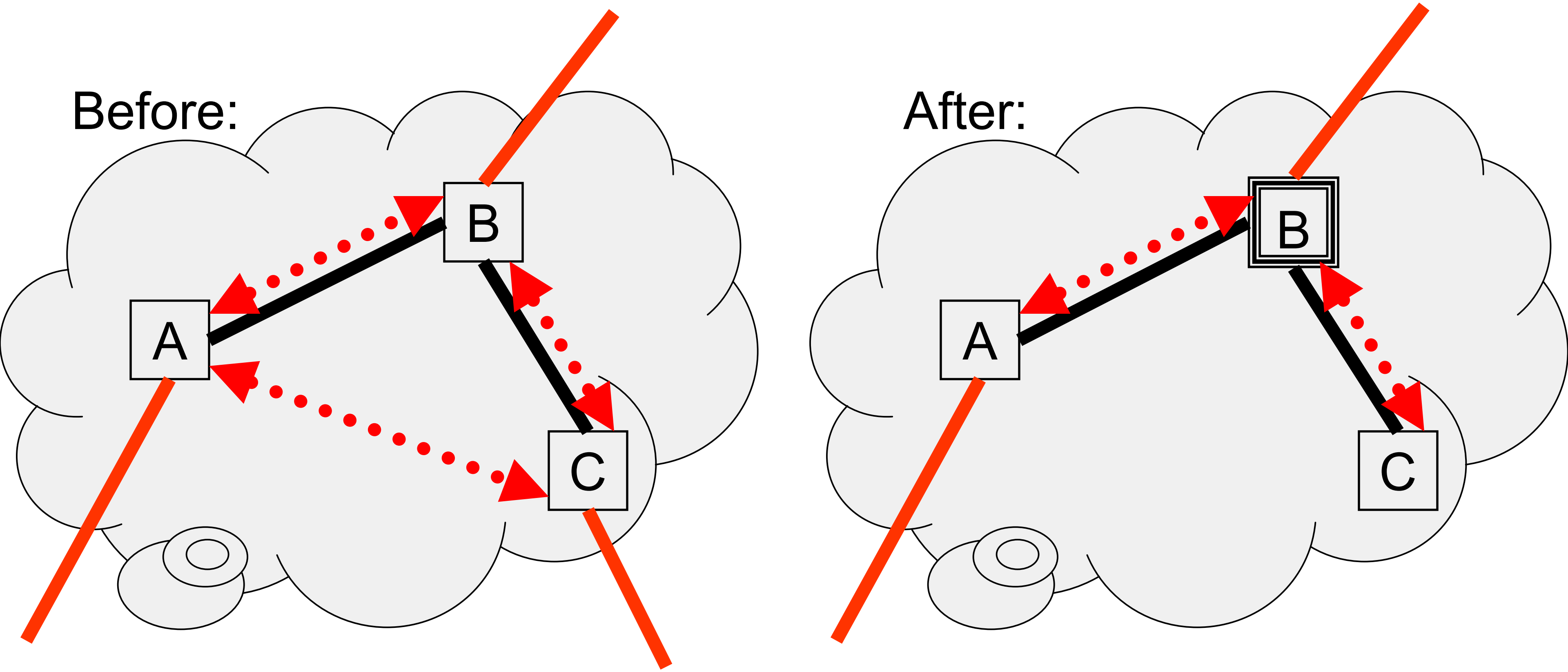
I-BGP Scaling

- Really wasteful!
  - CPU
  - Memory
  - Link capacity
- Poor scaling.
- Replicated traffic.
  - Chances are there is only one link between each group of four routers in the picture!

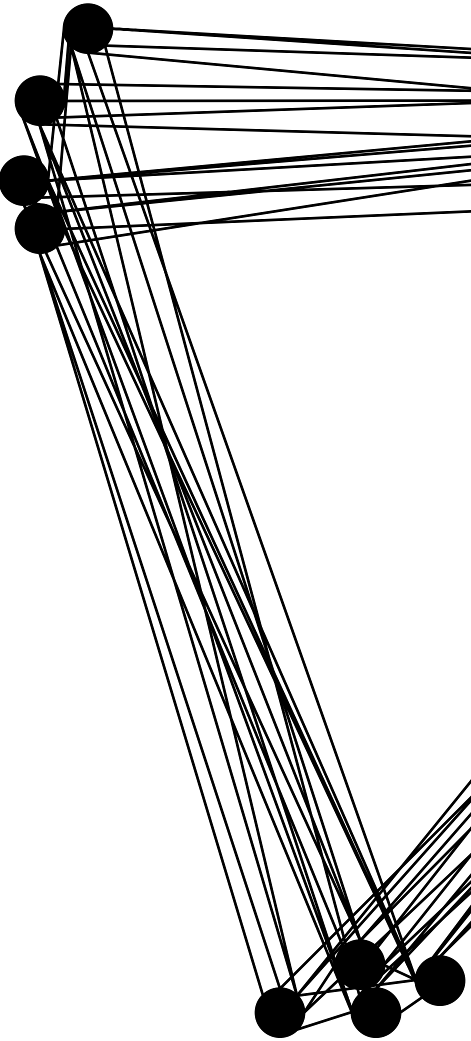


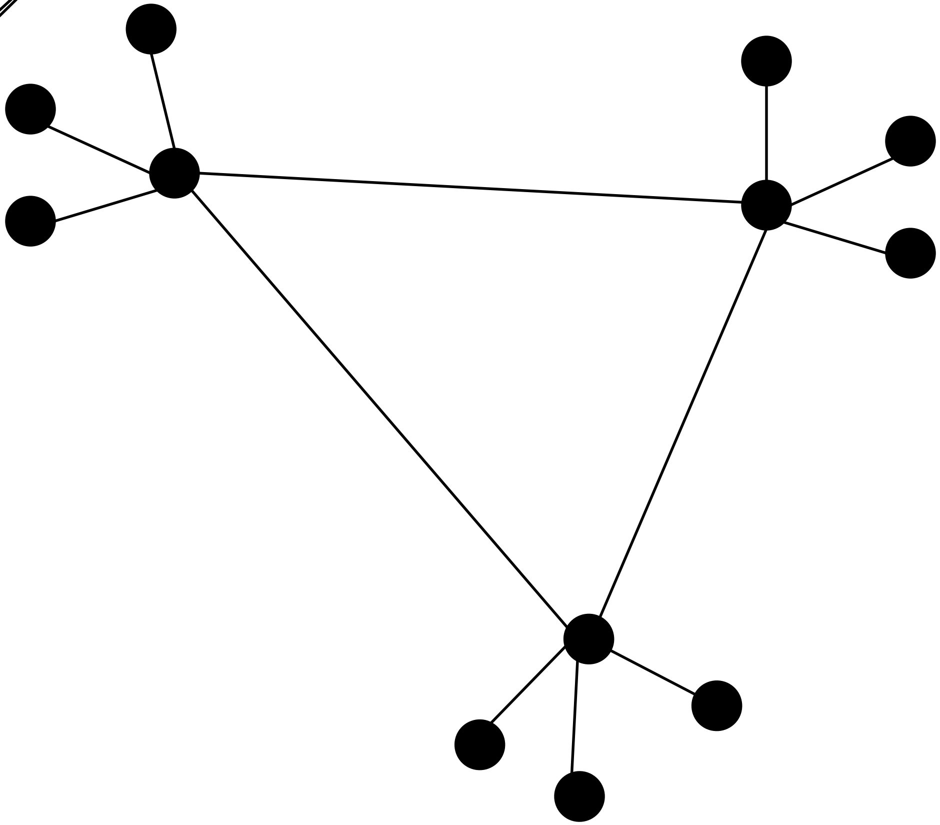
Route Reflection

- Relax the rule about not re-advertising I-BGP-learned routes.
    - Add hierarchy to I-BGP.
- Reduces # of sessions.
- RR can simply copy UPDATE messages (saves CPU).



Before/After





Lines represent IBGP sessions.

Route Reflection, cont’d

- I-BGP peers of a Route Reflector:
  - *Clients*
  - *Non-clients*
- A RR and its clients form a *Cluster*.
- Non-clients still form a full I-BGP mesh with each other.
- Clients only talk to their RR
  - And external peers, of course.
- Clients are normal I-BGP peers.
  - All they know is that they have been configured to peer with the RR.
- Which routers become RR depends on the topology.
  - Ditto for clusters.

Route-Reflector Route Selection

- RR receiving multiple routes to same destination runs regular BGP route selection procedure.

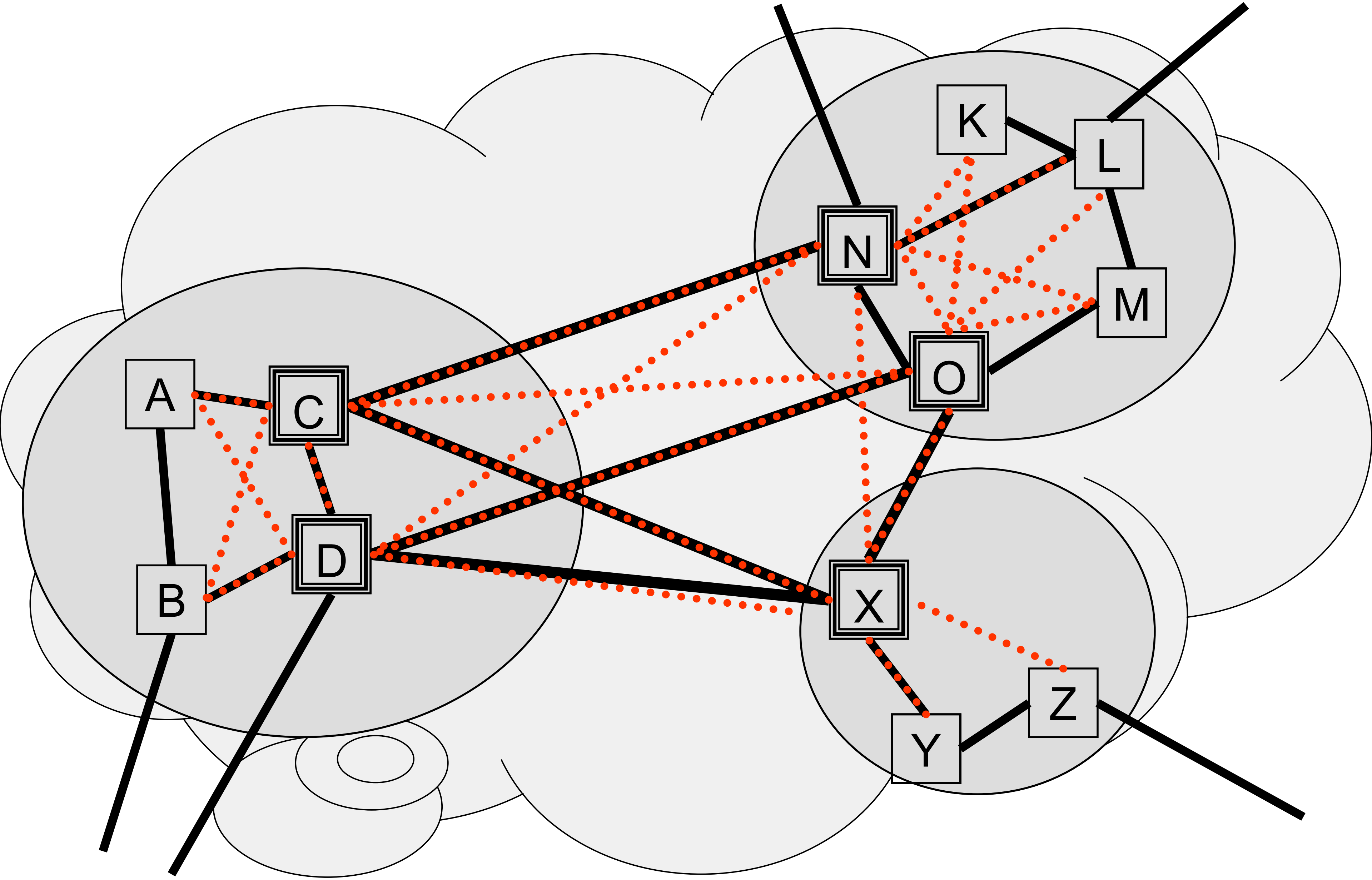
Received from:	**Reflect to:**
nonclient peer (RR or otherwise)	clients only
client	all other clients* all nonclient peers
EBGP	all clients all nonclient peers

*Except when clients are fully-meshed.

Redundancy in RR

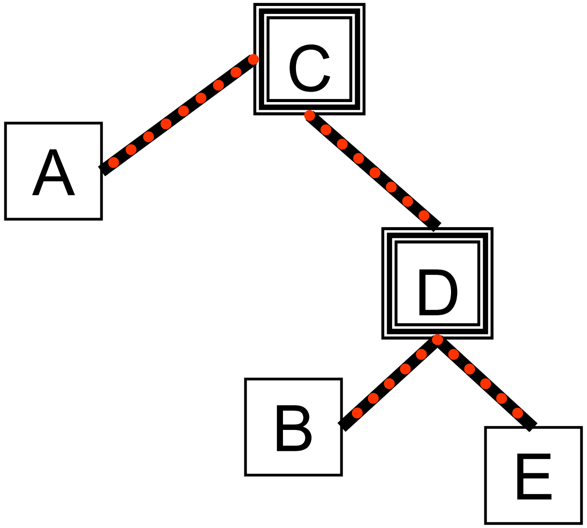
- If a route reflector goes down, I-BGP setup gets partitioned.
  - Not good!
- Redundancy.
- Each cluster gets at least two RRs.
  - Each client in the cluster talks to both RRs.
  - Yes, they get duplicate UPDATEs.
- RRs fully meshed.
- Clients can also be fully meshed inside a cluster.
  - RR must be configured not to readvertise to its own clients.
- Topology considerations.
  - I-BGP sessions should (if possible) flow over distinct links.

RR with Redundancy



Nested RR Configurations

- A client does not know it is a client!
  - A RR can be client of another RR.



- D is C’s client, but B&E’s RR.

RR and Attributes

- RR preserve BGP attributes.
- Necessary to avoid loops due to interactions with the IGP.
- NEXT_HOP in particular.

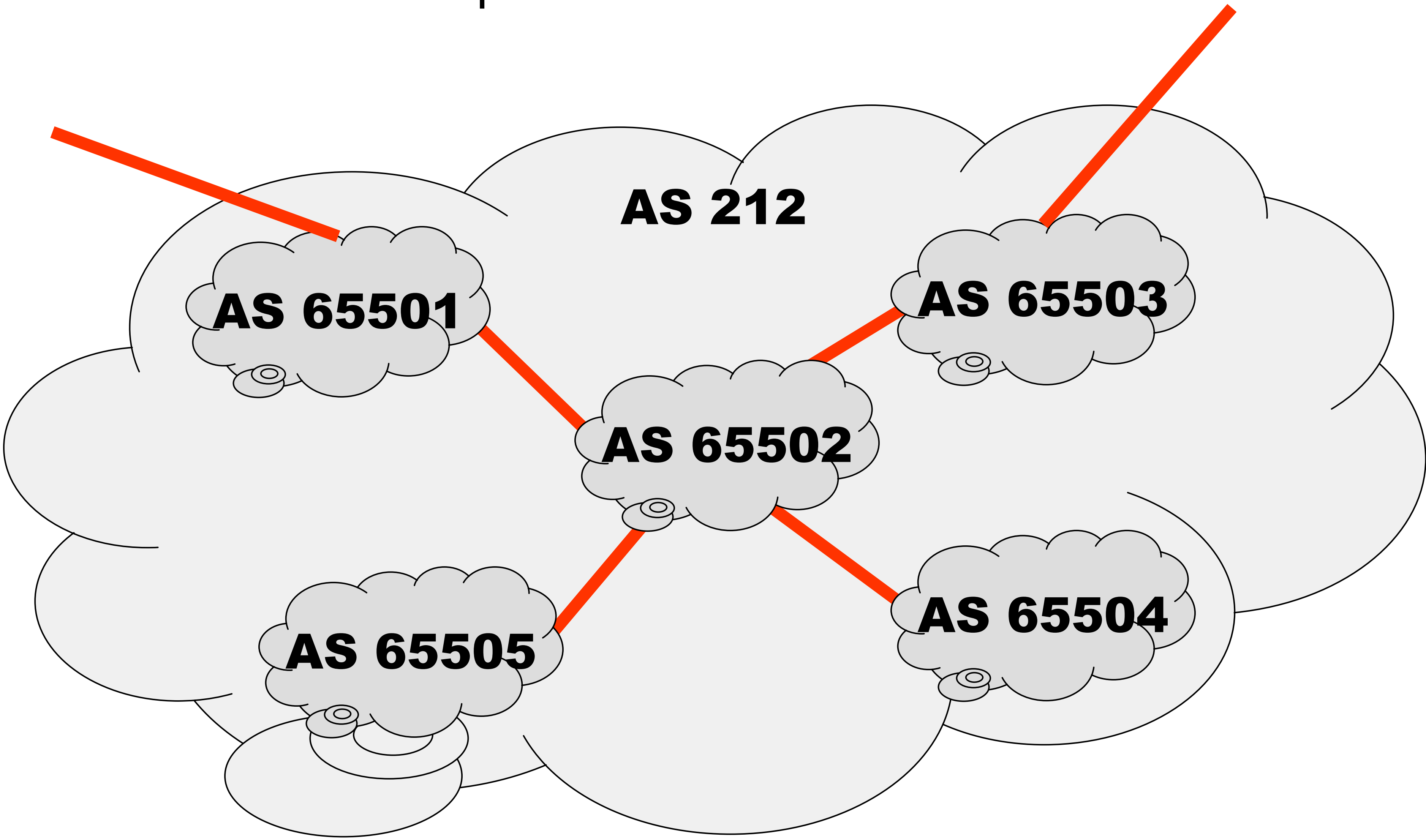
- Fewer actual paths are possible.
- Bizarre interactions can occur.
- RR/Clustering should follow topology.

Avoiding Loops

- Relaxation of the I-BGP re-advertising rule can lead to loops.
  - In cases of misconfiguration.
- ORIGINATOR_ID
  - Optional, non-transitive (type code 9).
  - Router ID of router that injected the route.
  - Added by the RR.
- CLUSTER_LIST
  - Optional, non-transitive (type code 10).
  - List of clusters that an UPDATE has traversed.
    - CLUSTER_ID should be the same in RRs of the same cluster.
  - Also added by the RR.
  - Remind you of anything?

Confederations

- RR enforces hierarchy.
- Alternative: break up AS into smaller ASes:

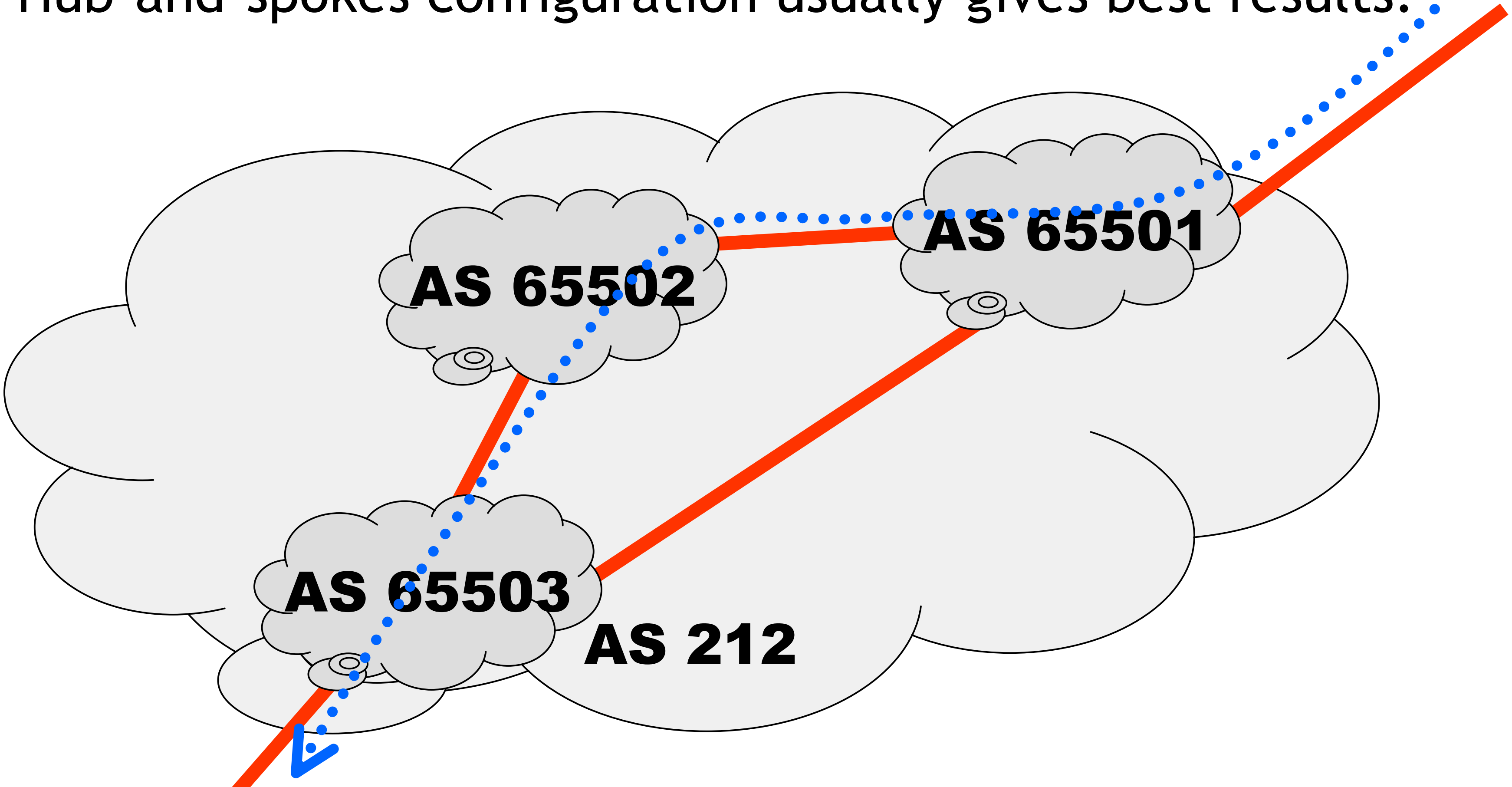


Confederations, cont'd

- Entire AS runs a single IGP.
  - Areas may or may not overlap with sub-ASes.
- Routers inside each sub-AS run normal I-BGP.
- BGP sessions between border routers of sub-ASes in the same confederation: EIBGP (what else!)
- Like E-BGP but with some changes.
  - LOCAL_PREF and MED are carried along.
  - NEXT_HOP is set by the first router, then carried along.
  - New AS_PATH segments:
    - AS_CONFED_SET (type 3).
    - AS_CONFED_SEQUENCE (type 4).
    - Stripped when going over a (real) EBGP session.
  - NO_EXPORT_SUBCONFED community.
- Route selection process is the same as with "regular" BGP.
  - Change: Prefer EBGP over EIBGP over IBGP.

Confederation Topology Considerations

- AS_PATH length stays constant (sub-AS components don't count).
  - Packets may take suboptimal path:
- Confederations should follow physical topology.
- Hub-and-spokes configuration usually gives best results.



RR *vs.* Confederations

- Experience varies.
- In RR, only the reflectors have to support the extension.
  - Not so in Confederations.
- Sub-ASs in a confederation can run individual IGPs.
- You can actually do RR inside a confederation.

Multihoming

- Connecting to multiple providers.
- Backup links (we’ve already examined this).
  - The backup link is idle unless the primary goes down.
  - Slow is better than dead!
  - We’ve already covered this.
- Load sharing / load balancing / redundancy.
  - To the same provider.
  - To different providers.

Redundancy Issues

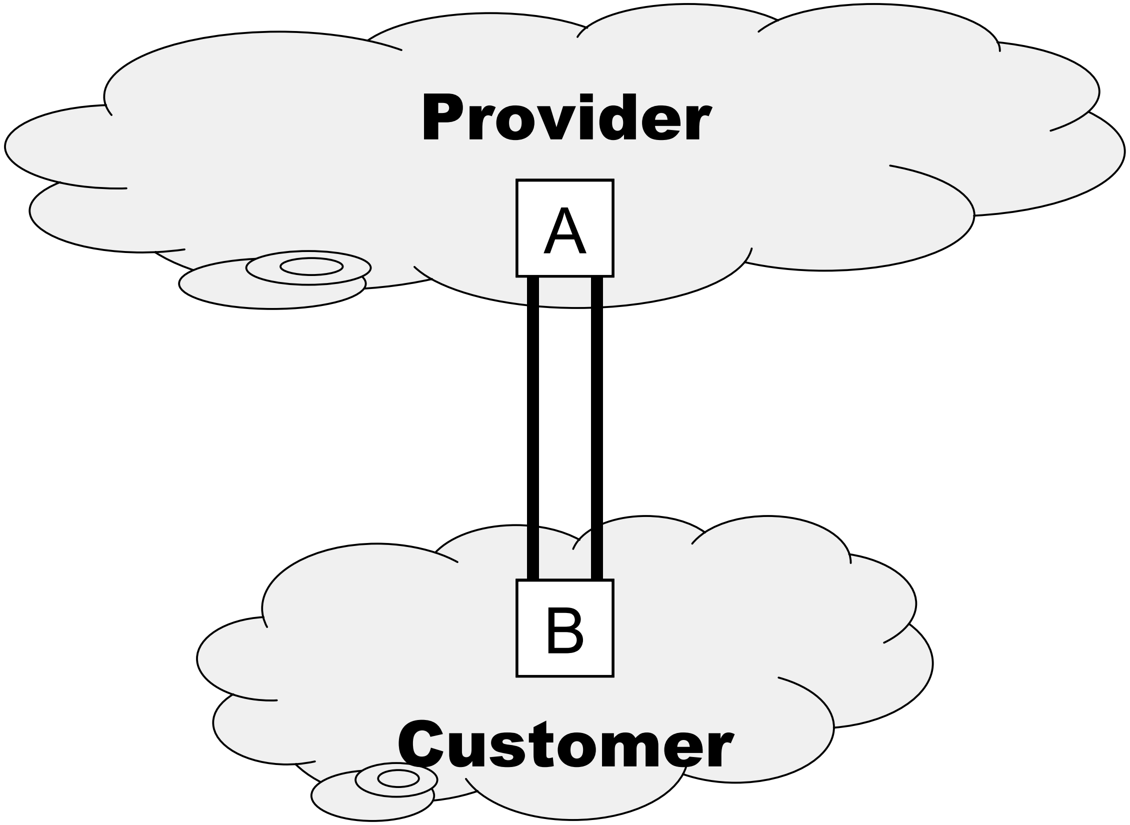
- Not just two ISPs!
- Redundant telco lines.
- Redundant power.
- Redundant exit points from the building!
- Redundant routers.
  - Make sure any additional hardware does not become a single point of failure!
- Redundant …

Multihoming Issues

- Addressing.
  - Pick addresses from upstream (main) provider.
  - Use addresses from both providers.
  - Get addresses allocated from ARIN/RIPE/APNIC.
- Routing.
  - Where/how to advertise prefixes.
    - Affects incoming traffic.
  - Where/how to set up own IGP.
    - Affects outgoing traffic.
- DNS
- Higher-layer protocols.

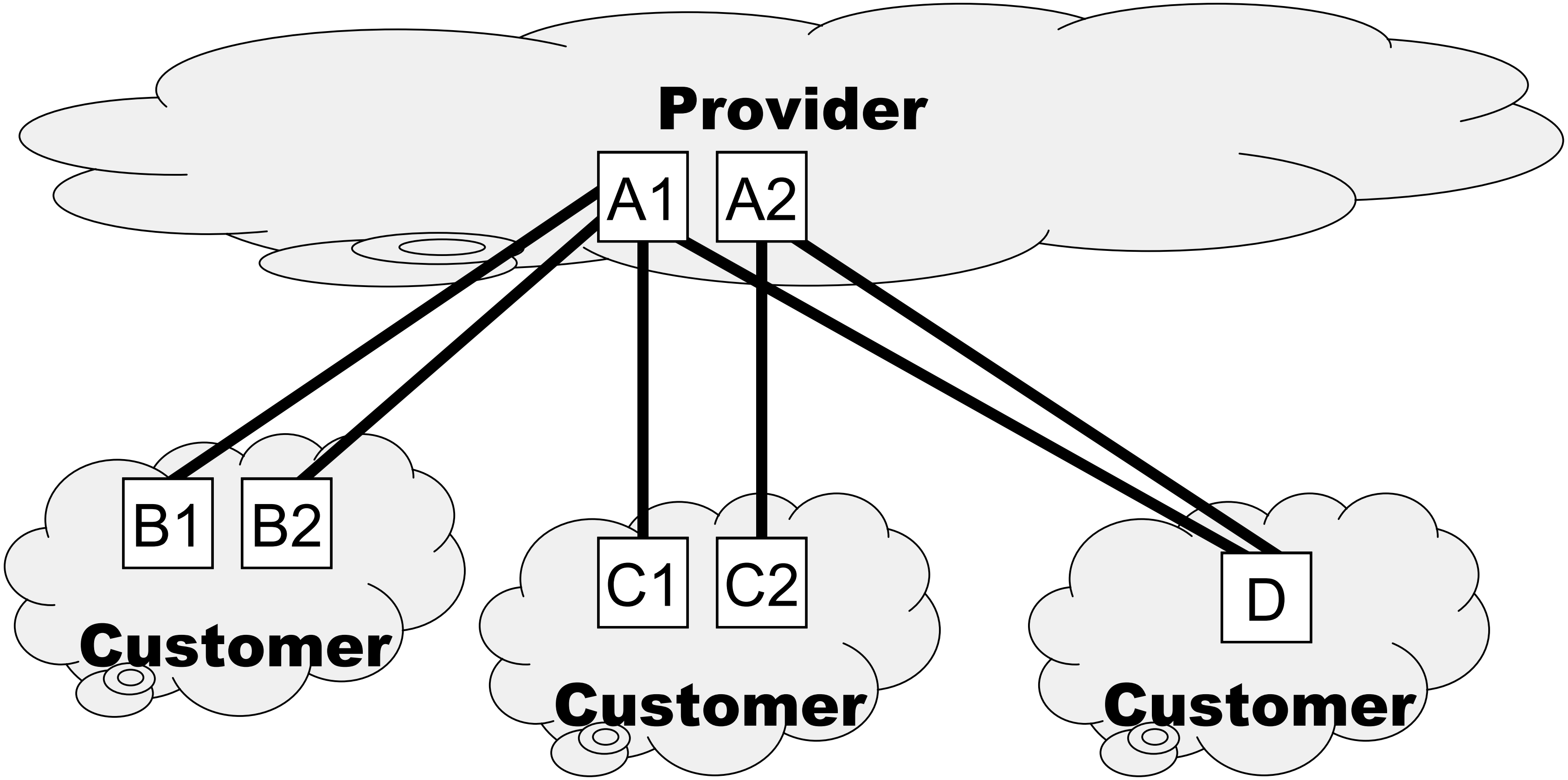
Dual Links

- Simplest cast: two distinct telco lines between the same pair of routers.
- Protects against link failure.



Dual Routers

- Different Configurations protects against router or link failure.
- A1/A2, B1/B2, C1/C2 are "near" each other.
  - IGP handles everything.
  - No BGP tricks involved.

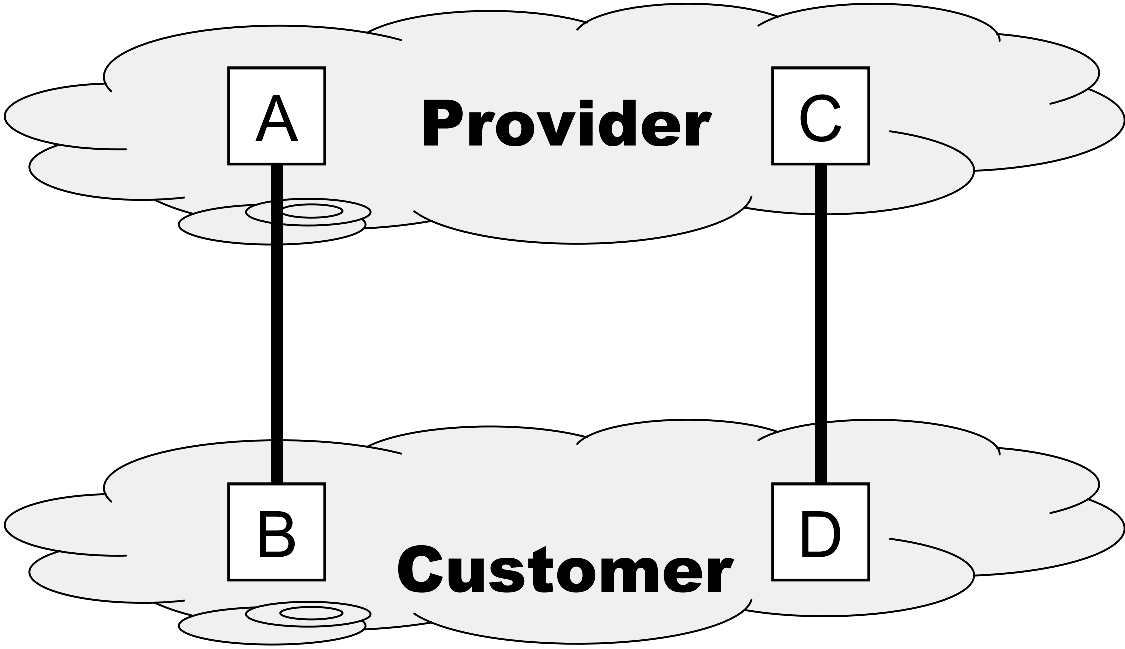


Dual {Links,Routers} cont’d

- These configurations add redundancy.
- Also enable load sharing/load balancing between the links.
- Traffic is (usually) split on a **per-flow** basis.
  - *Flow*: (protocol,src,dst,src-port,dst-port).
  - Performance reasons (can be done on the linecard).
  - Per-packet split possible at much higher CPU burden.
    - Or by using MUXes or multipoint PPP (below the network layer).
  - Packet ordering maintained.
    - At least across the redundant hop.
- OSPF can use equal-cost paths.

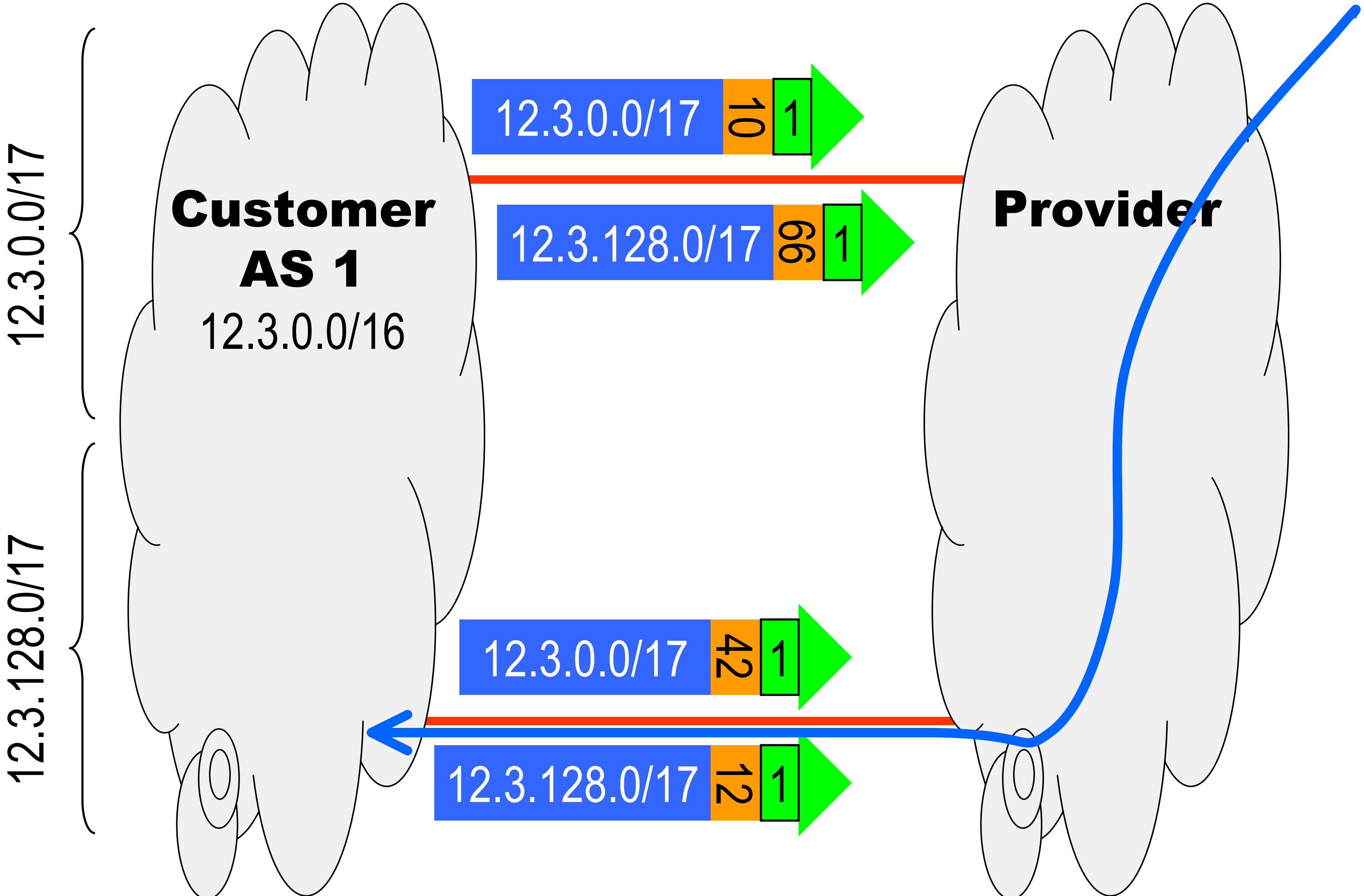
Multihoming to a Single Provider

- … when access links are "far" from each other.
- ISP advertises defaults to customer.
  - Customer's IGP ensures packets take the closest egress router (B or D).
- Customer advertises more-specifics with MED to force cold-potato routing.

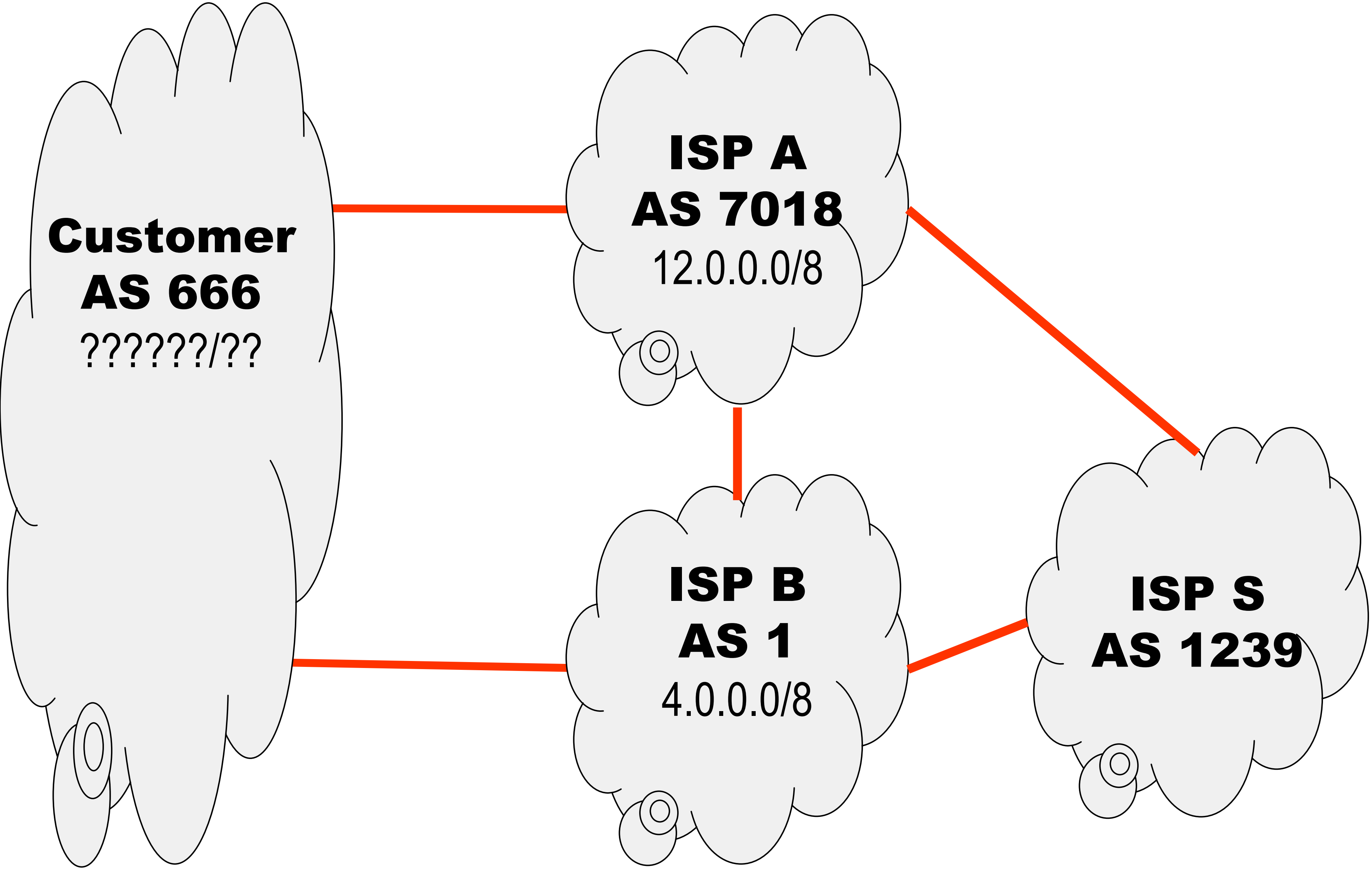


Cold-Potato with MEDs

- MED takes precedence over IGP distance.



Multihoming to Multiple Providers



Own Address Space

- Great if you can get it!
  - And if you’re big enough.
- If the prefix is too long (> /24), it may not get through filters.
  - Lose connectivity from parts of the Internet.
- It does get redundancy.
- Does it get us good load-sharing?
  - Depends on the relative sizes of ISP A and ISP B.
- If equally “important”
  - roughly half the traffic will be coming from each
  - roughly half the announcements will be “better” from one of the two
    - resulting in outbound load sharing.
- Otherwise, may use AS_PATH padding to shed some traffic.

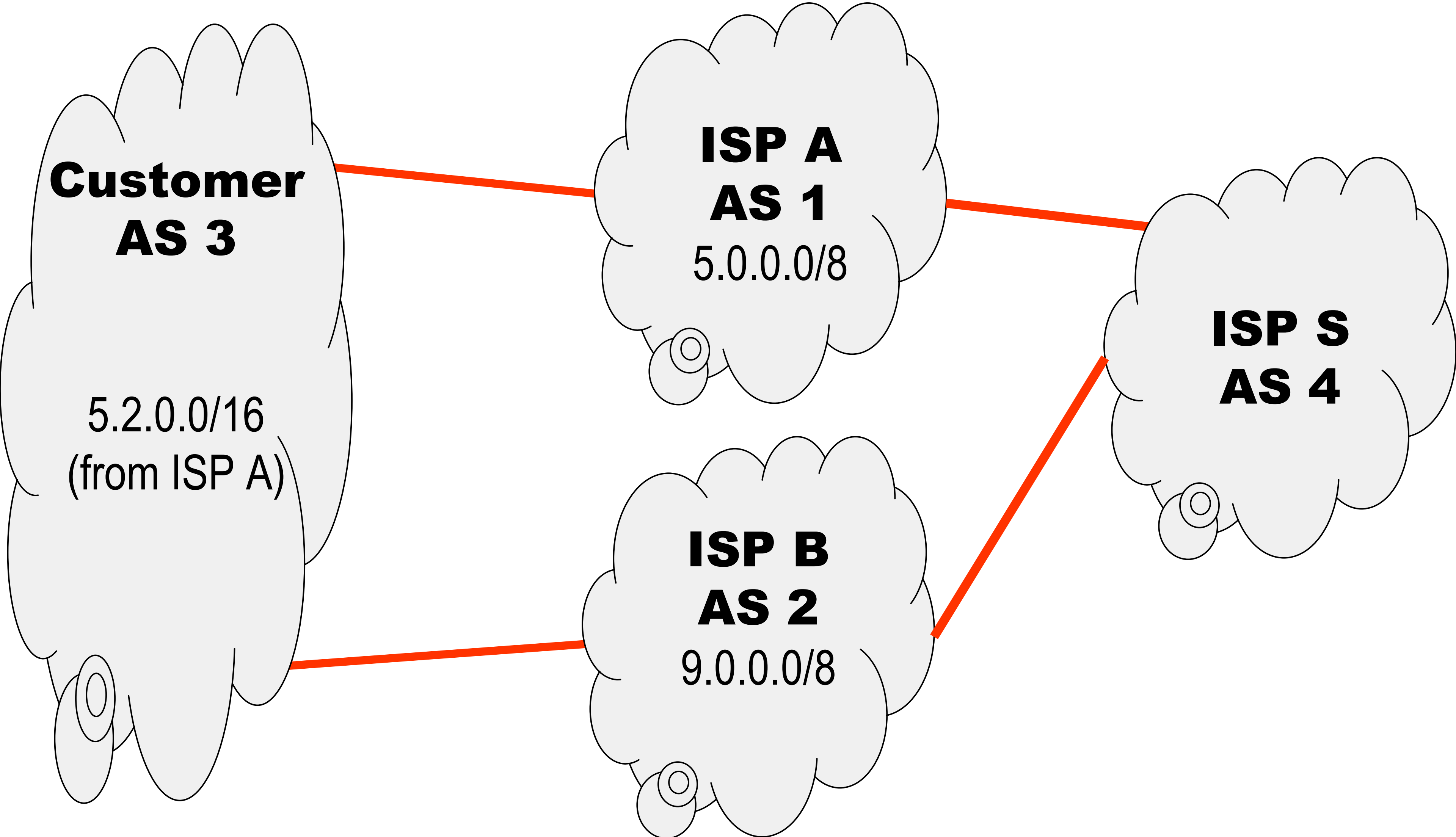
Address Space from Both ISPs

- With the service agreement comes address space.
  - 12.96.16.0/20 from ISP A.
  - 4.99.32.0/21 from ISP B.
- Announce the 12… space to A, and the 4… space to B.
  - (or not announce at all).
- Load sharing depends on source/destination of bulk of traffic.
- No redundancy.
  - If one link goes down, half of Customer’s address space is unreachable.
  - And unusable (no return routes).

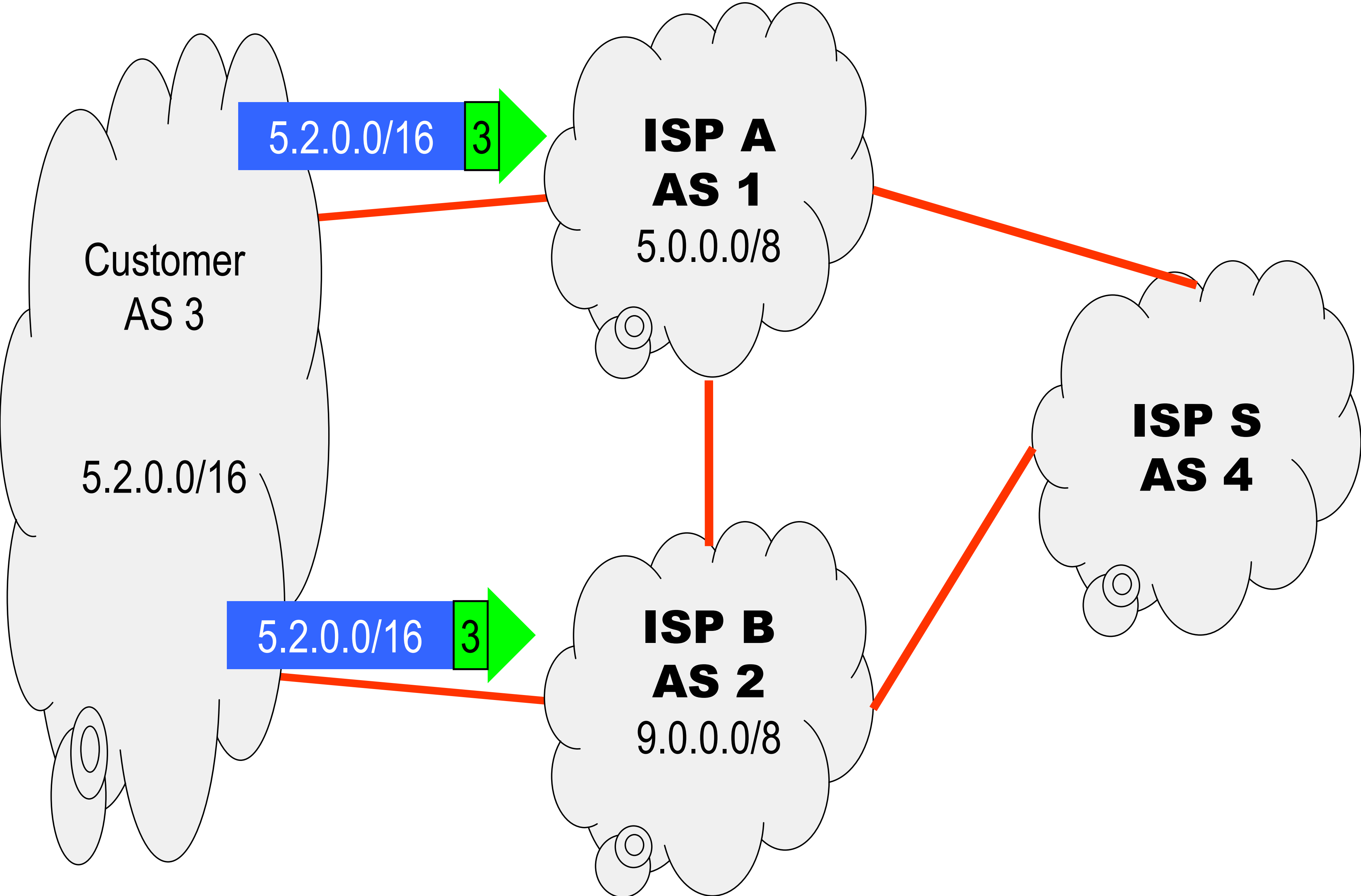
- Use DNS round-robin to respond with addresses from both spaces.
  - Incoming connections will chose an address at random.
  - Not optimal in half the cases.
- How to pick address for outgoing connection?
  - Allocate address by region.
  - Random.
- Problems if ISPs do ingress filtering.
- Use of NAT has been suggested (arrrgggghhhh!)

Address Space from one ISP

- Outgoing traffic **from** Customer is not affected.



What does AS3 Advertise?

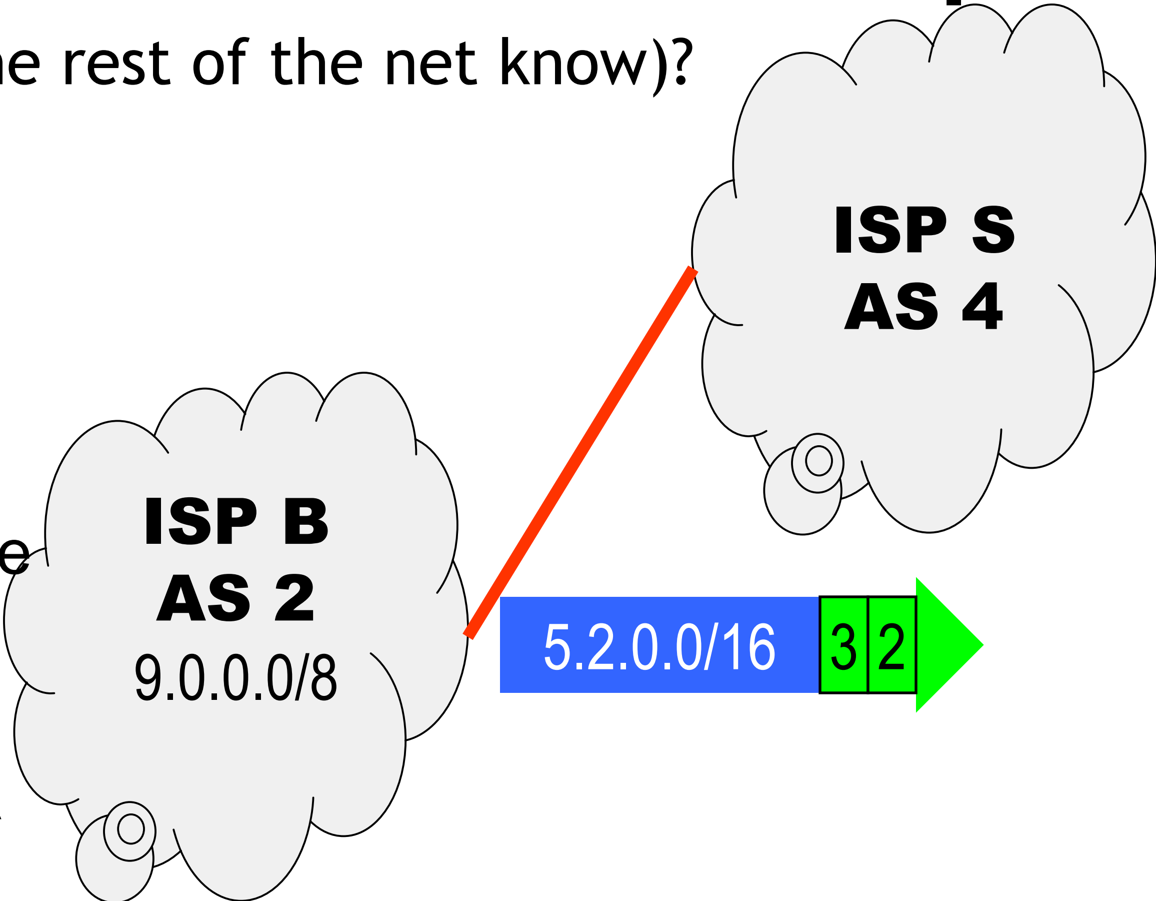


- Customer advertises its prefix to both its ISPs.
- ISP A (and its customers) now knows how to reach 5.2.0.0/16.
- ISB B (and its customers) also knows how to reach 5.2.0.0/16.
  - Although it gets 5.0.0.0/8 from ISP A.
    - Longest-prefix match.

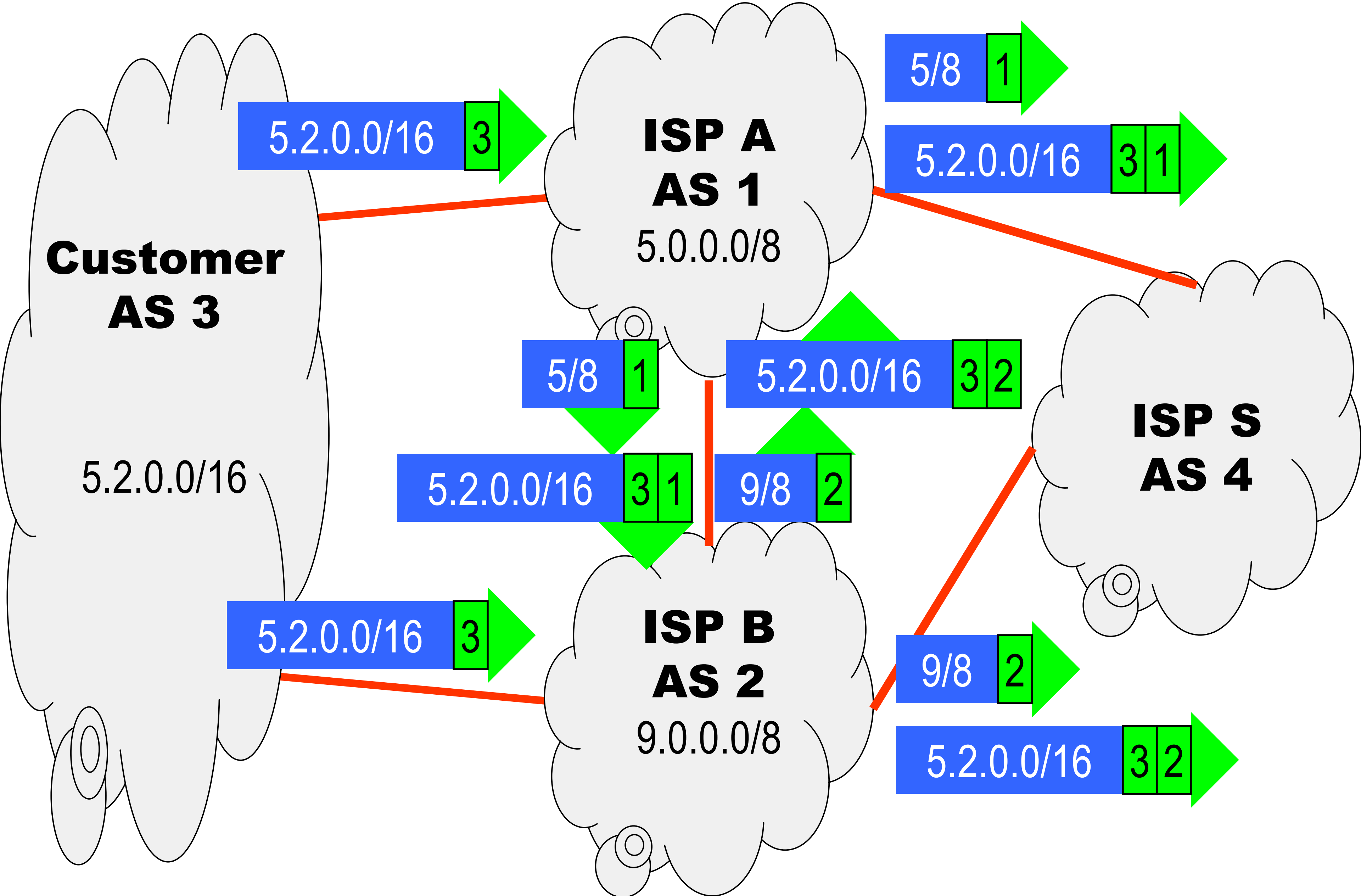
[ISP B could in some situations filter 5.2.0.0]

- What does ISP S (and the rest of the net know)?

- ISP B advertises the longer prefix to S.
- S now sends all traffic for 5.2.0.0/16 via B!
- This can lead to massive asymmetry!
  - Depends on relative amts of traffic from A *vs.* B+S



What is being advertised?



- ISP A had to "**punch a hole**" in its aggregation policy.
- What is carried in ISP A's I-BGP?
  - ISP-A knows that Customer is a proper subset.
  - If the access router does not readvertise inside I-BGP the more-specific, traffic for Customer would go out via ISP B!
    - Access router has to be configured accordingly.
- Customer and ISP A **must** run BGP.
  - I.e., A's access router can't just inject a static route.

- ISP S has the more-specific for Customer from both ISP A and ISP B.
  - Will route traffic for Customer properly.

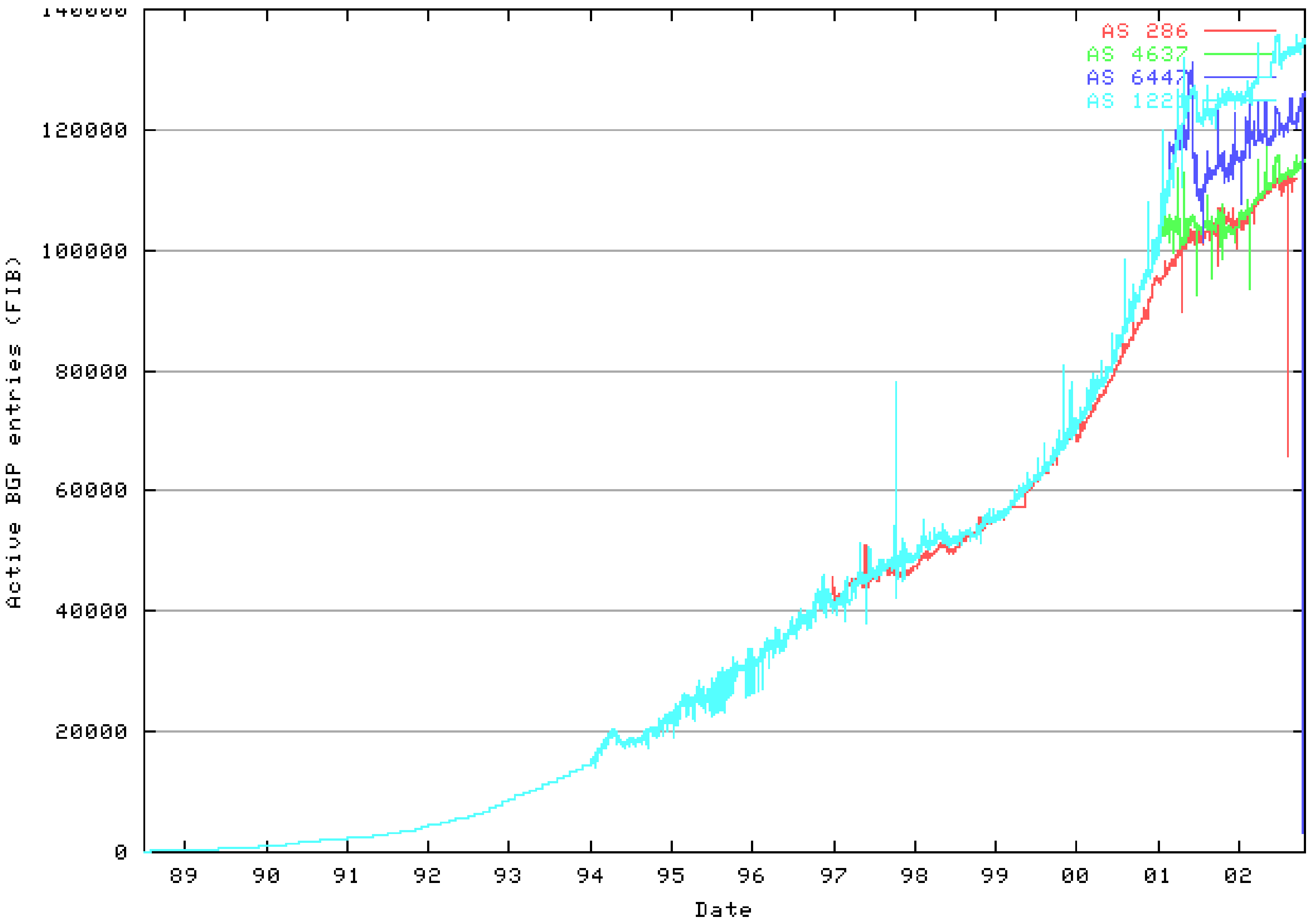
Aggregation

- Address aggregation: announcing one less-specific prefix in lieu of many more-specific prefixes.
- Example:
  - Provider has a /12.
  - Customers are allocated /16s through /24s from that space.
  - Provider **filters** the more-specifics and only announces the /12 to its peers.
- More-specifics may still need to be carried inside I-BGP.
  - Finer-level aggregation on access routers.
  - (e.g.) Sixteen /24 customers are on an access router.
  - Access router advertises a /20 into the I-BGP mesh.
- More-specifics may still be announced (e.g., with NO_EXPORT) to some peers.

Aggregation and Filtering

- External aggregation: provider only announces aggregates to its peers, not individual customer more-specifics.
- Internal aggregation: longer prefixes allocated to access routers, so that fewer routes are carried in I-BGP.
- Many times providers have to de-aggregate.
  - For multi-homed customers.
- Some providers do not allow in (filter) prefixes longer than /19 or /20 from aggregatable address space (post-CIDR allocations).
  - Contentious issue.
- Deaggregation leading cause of BGP table size.
  - "Grazing the commons"

Routing Table Size



- Source: http://bgp.potaroo.net/
- Active (used for the FIB) table.

BGP Scaling Issues

- Previous graph shows **active** routes (in the "Loc-RIB").
- Many more routes floating around.
- Can't just "add more memory".
  - FIB memory is expensive, on linecards.
  - CPU/link capacity still an issue.
- Both the number of routes and the rate of UPDATEs (and their first derivatives) are scaling issues.
- Moore's law only means we have to keep buying new routers!
- For a good time, go to telnet://route-views.oregon-ix.net/
- Chief problem: (at least) one route per advertised prefix.
  - De-aggregation due to multihoming a main source of the problem.
  - Switching to IPv6 doesn't fix this!
  - Need a better routing architecture?

AS Numbers

- About 14K already.
- Increasing faster than linearly.
  - Current derivative: 2K/year.
- Source of new AS numbers:
  - New ISPs.
  - New multihomed customers.
- At this rate, we run out around 2007-2010.
  - IPv6 doesn’t fix this either!
- Suggestions:
  - 4-byte AS numbers (draft-ietf-idr-as4bytes-05.txt).
  - ASE (AS Number Substitution on Egress (AitFotL)).
    - Another cause of MOAS conflicts.

Route Flapping

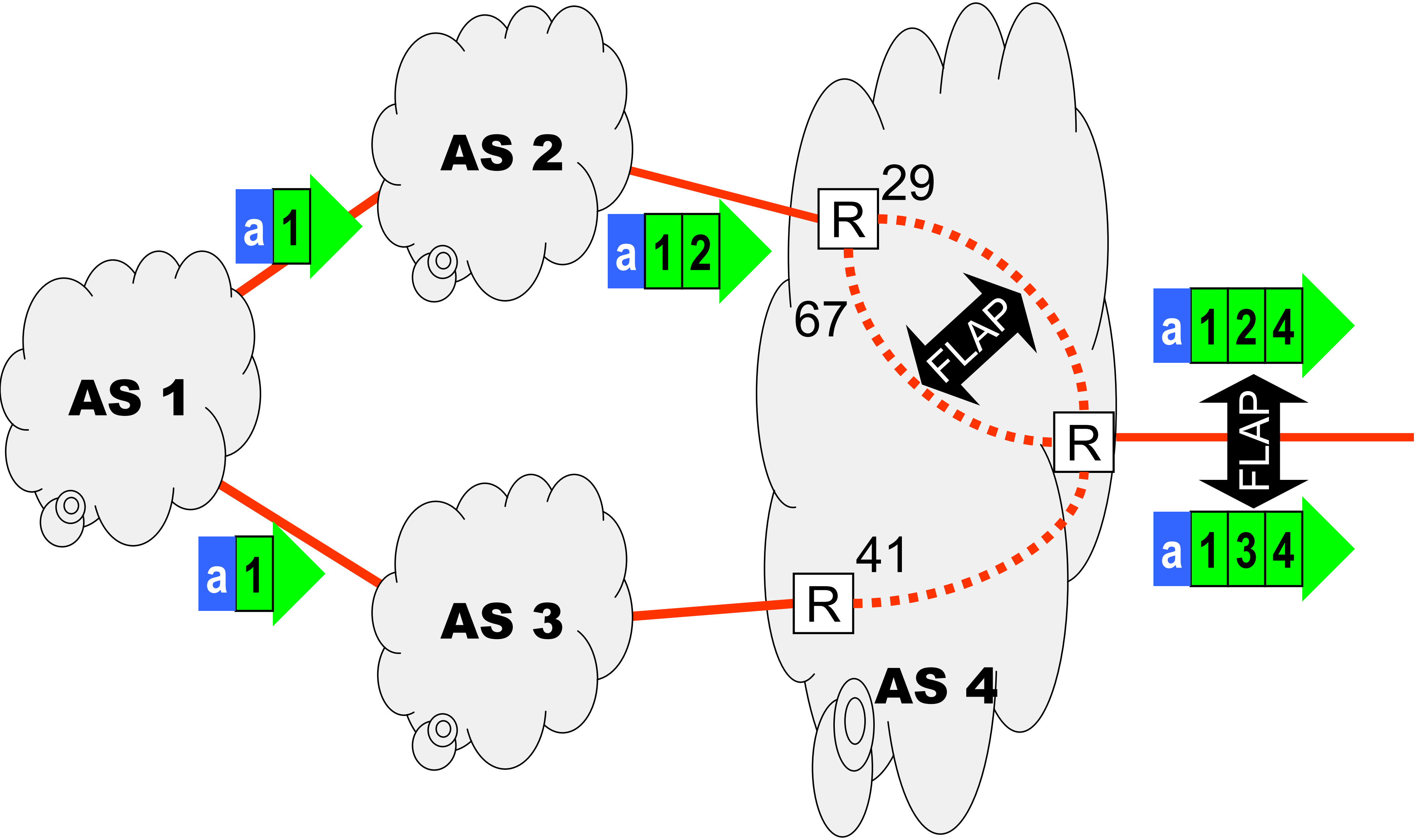
- Routing instability.
- Route disappears, appears again, disappears again…
  - Withdrawal, announcement, withdrawal, announcement…
- Visible to the entire Internet.
  - Wastes resources, triggers more instability.
- Some causes of *Route Flapping*:
  - Flaky inter-AS links.
  - Flaky or insufficient hardware.
  - Link congestion.
  - IGP instability.
  - Operator error.

Link Instability

- The first three are examples of link instability.
  - Link itself fails.
  - Router/router interface fails.
  - Messages can’t get through.
- When a link goes down, routers withdraw routes associated with this link.
  - Customer-ISP.
  - ISP-ISP.
- Announcements travel throughout the default-free zone.
- Aggregation may mask downstream flapping.
  - Does not work for multihoming

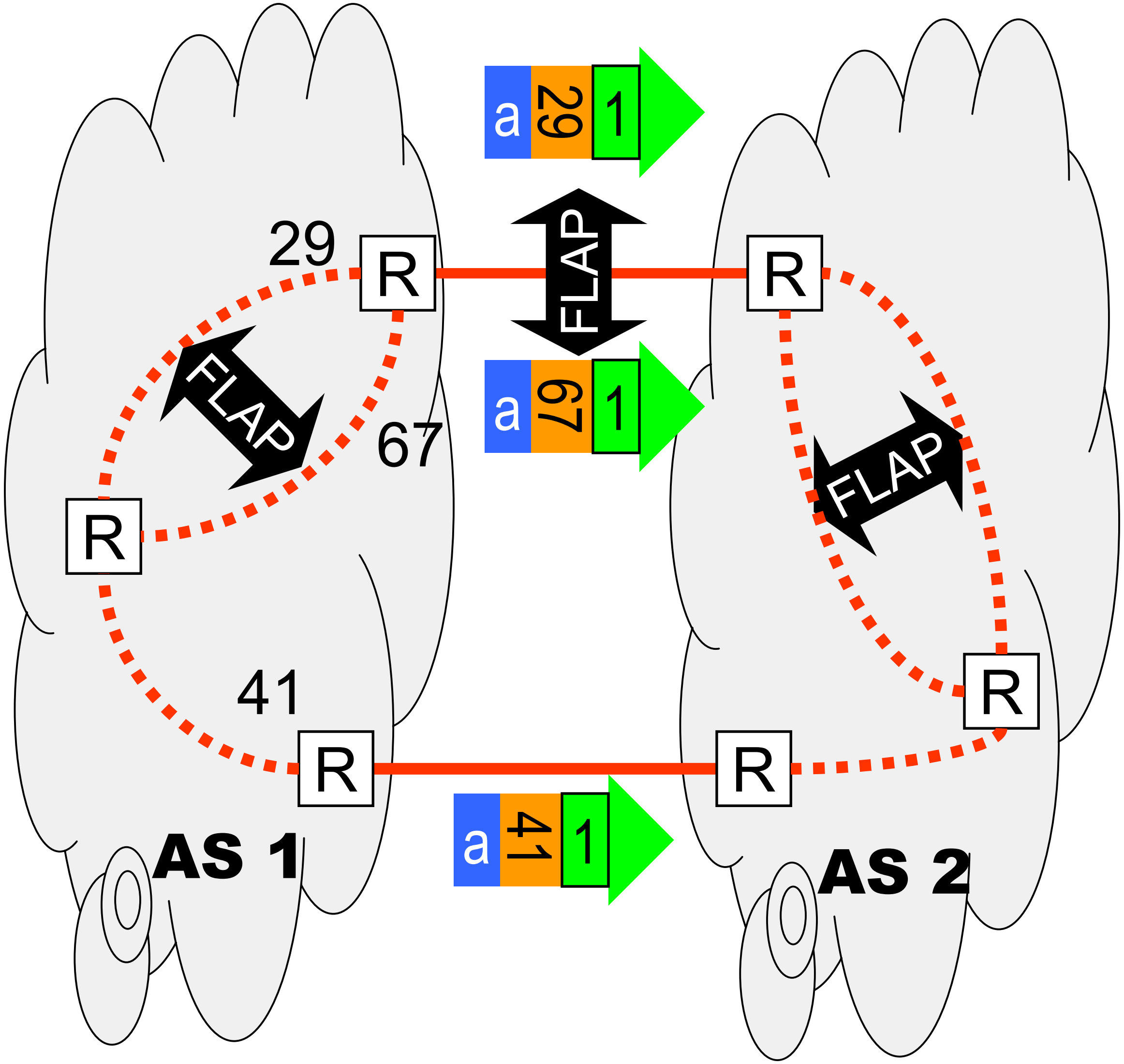
IGP Instability

- IGP route-preference rule exports instability.



IGP Instability

- MEDs can export internal instability.



Route Flap Damping

- Router detects route flapping.
- *Penalty*:
  - Increased each time a route flaps.
  - Decreased over time.
- If penalty threshold exceeded (*suppress limit*), route is suppressed.
- Until penalty drops below a certain level (*reuse limit*).

More BGP Extensions

- HELLO optional parameters:
  1. TCP MD5 Authentication (RFC2385).
  2. Capabilities negotiation (RFC2842).
     - TLVs indicating what optional capabilities the sender supports.
- If receiver does not support, closes connection with appropriate NOTIFICATION.

TCP MD5 Authentication

- TCP option type 19.
- 18 bytes long.
- 16 bytes of MD5 hash, including key, of TCP segment.

- Poor authentication.
- Should have used IPsec (of course).
- Does not make key management any easier.

Route Refresh Capability

- It’s a request to the peer to send its Adj-RIB-Out.
- Used when the inbound policy of a peer changes.
    - All the routes that the peer had gotten (and potentially filtered or changed attributes thereof) have to be re-processed by the input policy engine.
- Alternative: close and reopen BGP session.
    - Causes lots of routes to flap.

- RFC 2918
- New BGP message (Type=5).

Outbound Route Filter Capability

- Request to the peer to send its inbound prefix filters.
- Rationale: why bother sending routes that will be filtered anyway?

- draft-ietf-idr-route-filter-06.txt

Graceful Restart Capability

- Indicates the ability to preserve BGP state across restarts.
- Minimizes distrubance.

- draft-ietf-idr-restart-05.txt

Dynamic Capability

- Capabilities are negotiated during OPEN.
- DC allows capabilities to be negotiated after OPEN.
- CAPABILITY message (Type=6)

- draft-ietf-idr-dynamic-cap-02.txt

Multiprotocol Extensions for BGP-4

- Negotiated capability.
- Extension to allow BGP-4 to carry routes for protocols other than unicast IPv4 (IPv6, multicast, *etc*.)

- Two new attributes:
  - MP_REACH_NLRI (Type=14)
    - Replaces NEXT_HOP attribute and NLRI field.
  - MP_UNREACH_NLRI (Type=15)
    - Replaces list of withdrawn routes.

- RFC2858 and draft-ietf-idr-rfc2858bis-02.txt

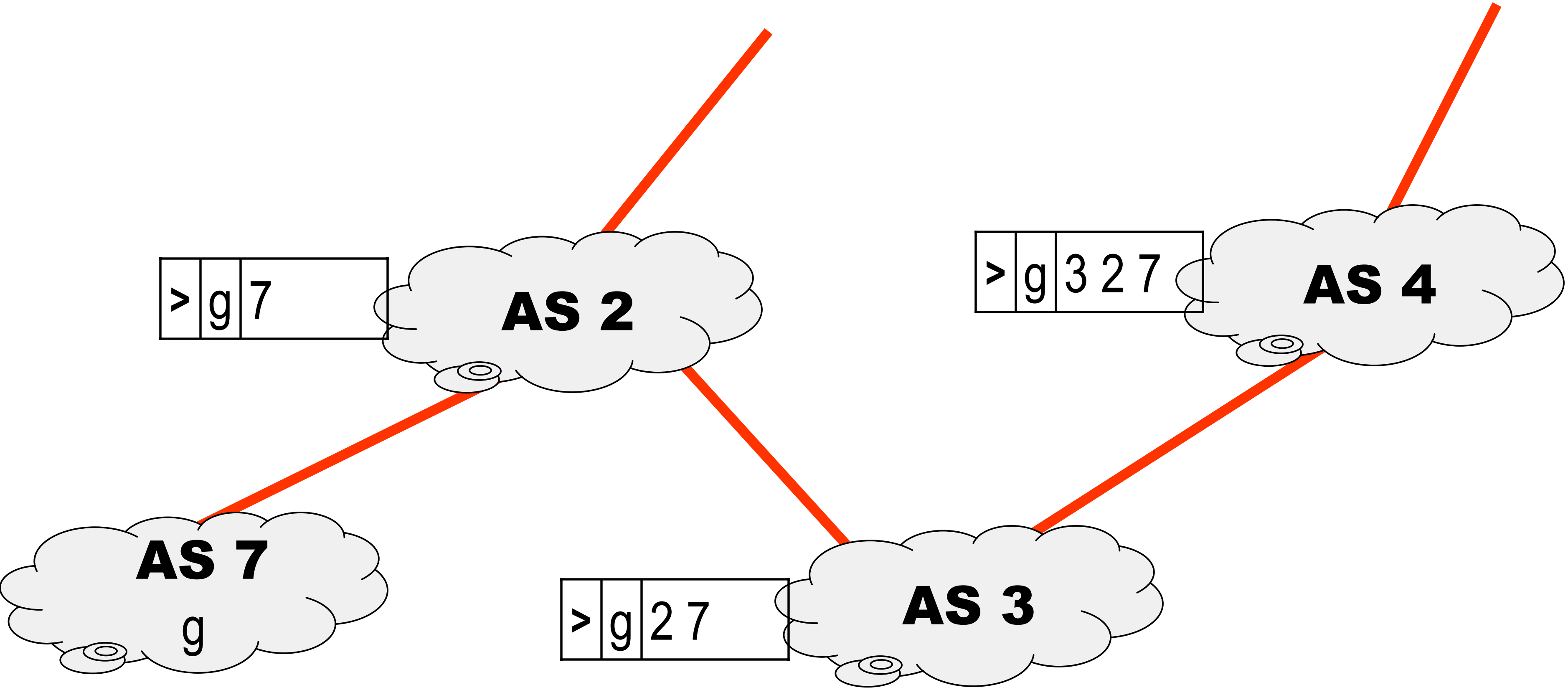
Dynamic Behavior of BGP

- The network is never in steady-state.
- Links break, routers crash, people make mistakes.
  - Routes get withdrawn.
  - New routes get advertised.
- How often do these happen?
- What is the effect on prefix reachability?
- Are they random or do they follow patterns?
- How disruptive are they?
- Can we/do we do anything to protect the network against them?

- Lots of recent and current research.

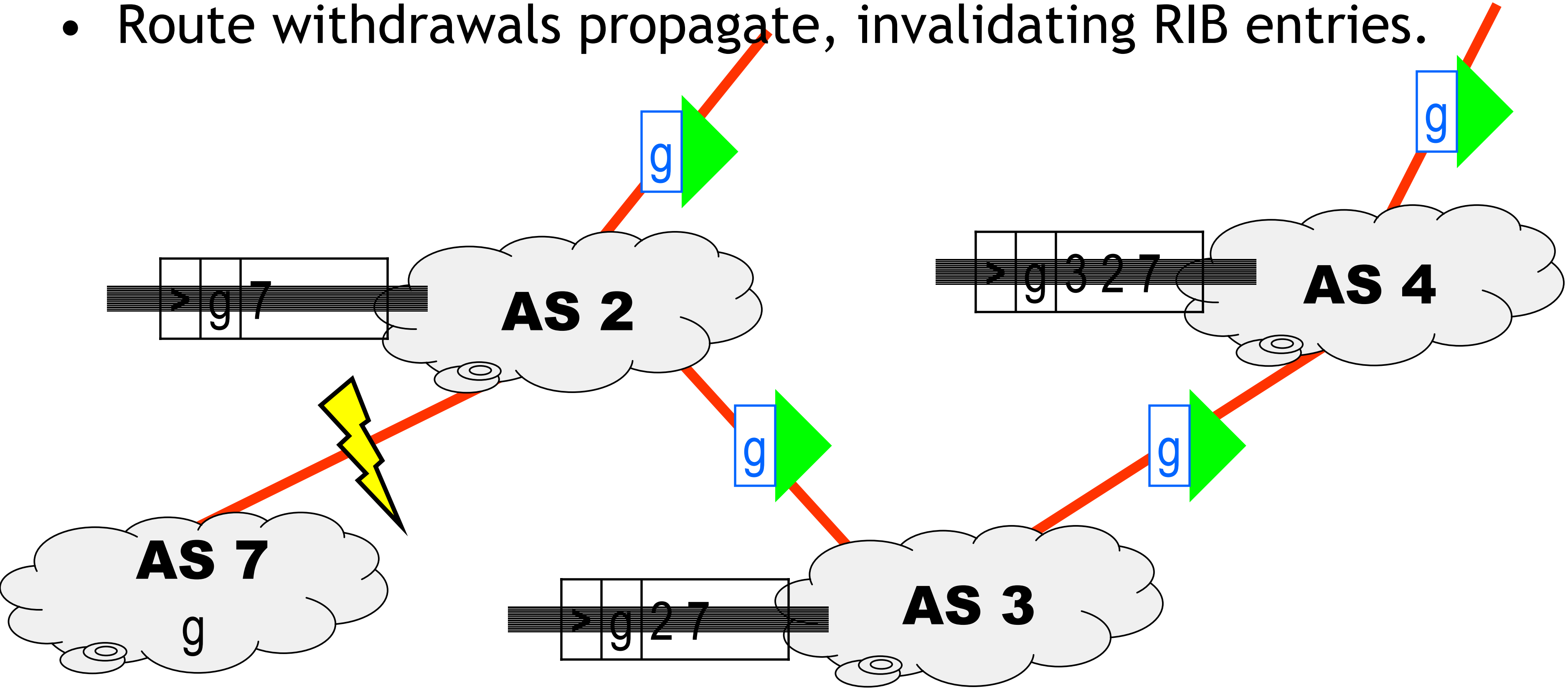
Link Failure (Single-homed system)

- AS 7 (prefix: g) is single-homed.



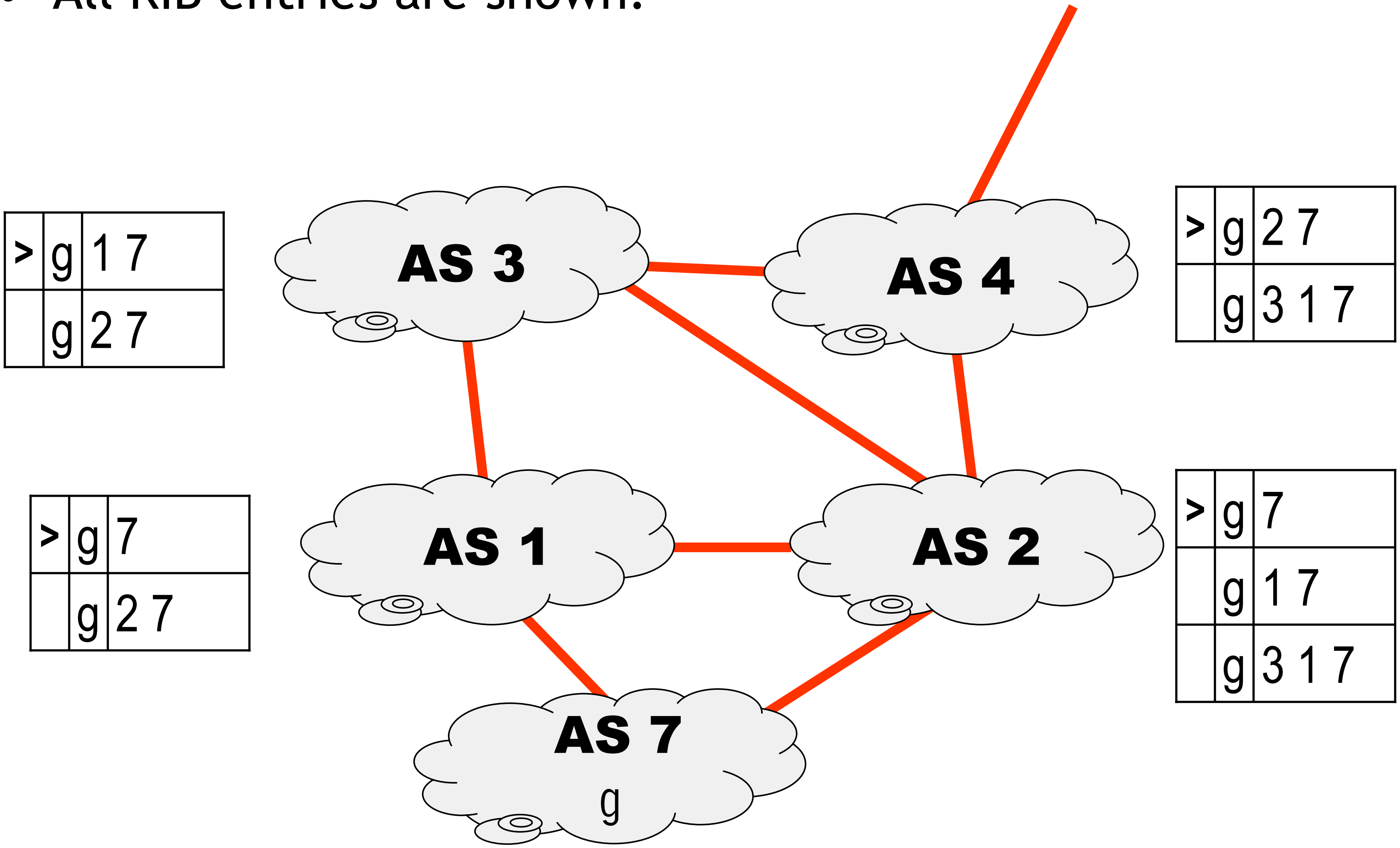
Link Failure (Single-homed system)

- Link between AS2 and AS7 fails.
- AS2 removes g from its RIB (both its Adj-RIB-1 and its Loc-RIB).
- AS2 withdraws route to g.
- Route withdrawals propagate, invalidating RIB entries.



Link Failure (Multihomed system)

- AS 7 (prefix: g) is dual-homed.
- All RIB entries are shown.



>	g	1 7
	g	2 7

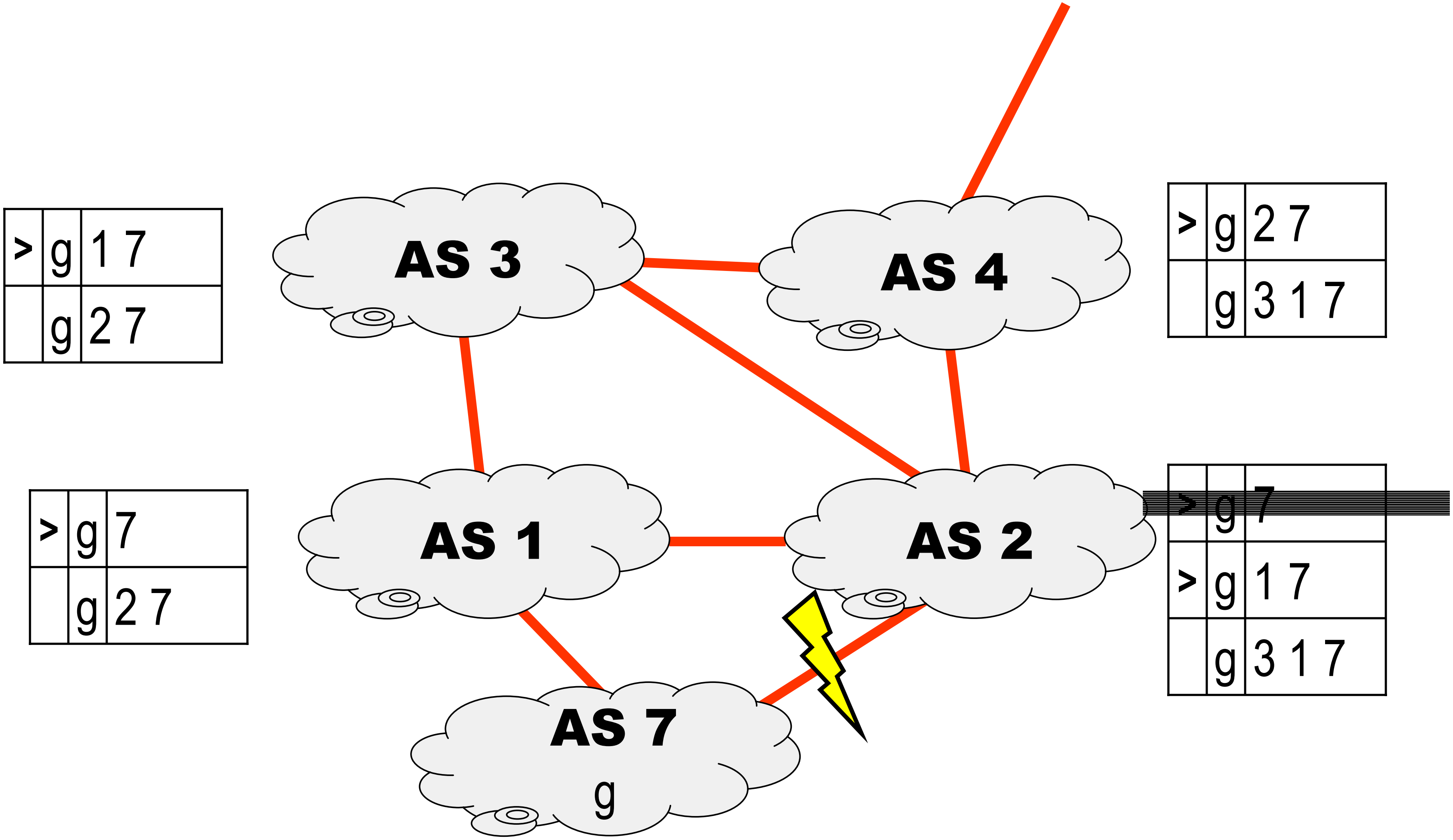
>	g	2 7
	g	3 1 7

>	g	7
	g	2 7

>	g	7
	g	1 7
	g	3 1 7

Link Failure (Multihomed system)

- Link 2-7 fails.
- AS2 selects next best route.



>	g	1 7
	g	2 7

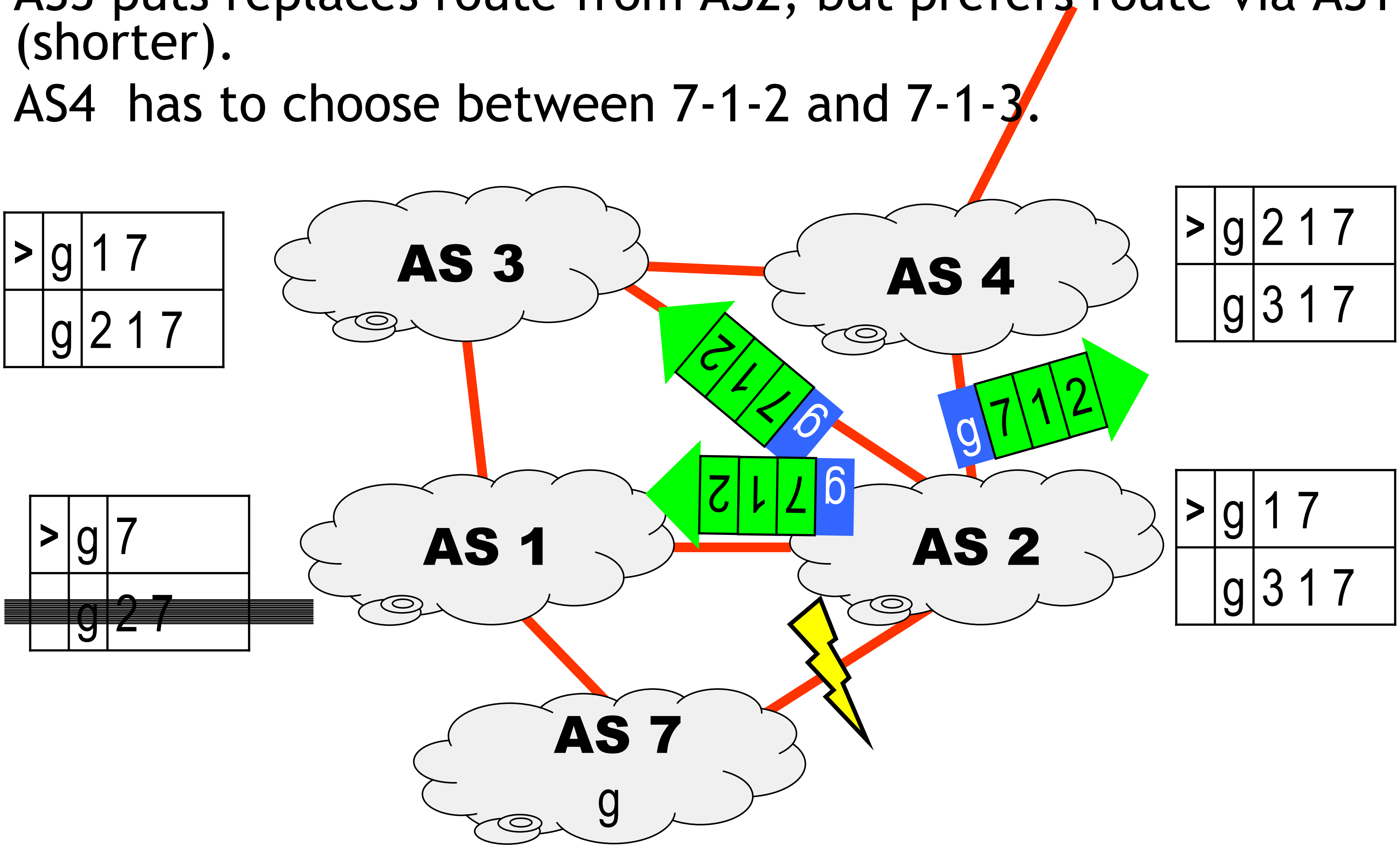
>	g	7
	g	2 7

>	g	2 7
	g	3 1 7

~~>~~	~~g~~	~~7~~
>	g	1 7
	g	3 1 7

Link Failure (Multihomed system)

- New route is advertised.
- AS1 removes route to g (it’s in the AS_PATH).
- AS3 puts replaces route from AS2, but prefers route via AS1 (shorter).
- AS4 has to choose between 7-1-2 and 7-1-3.



>	g	1 7
	g	2 1 7

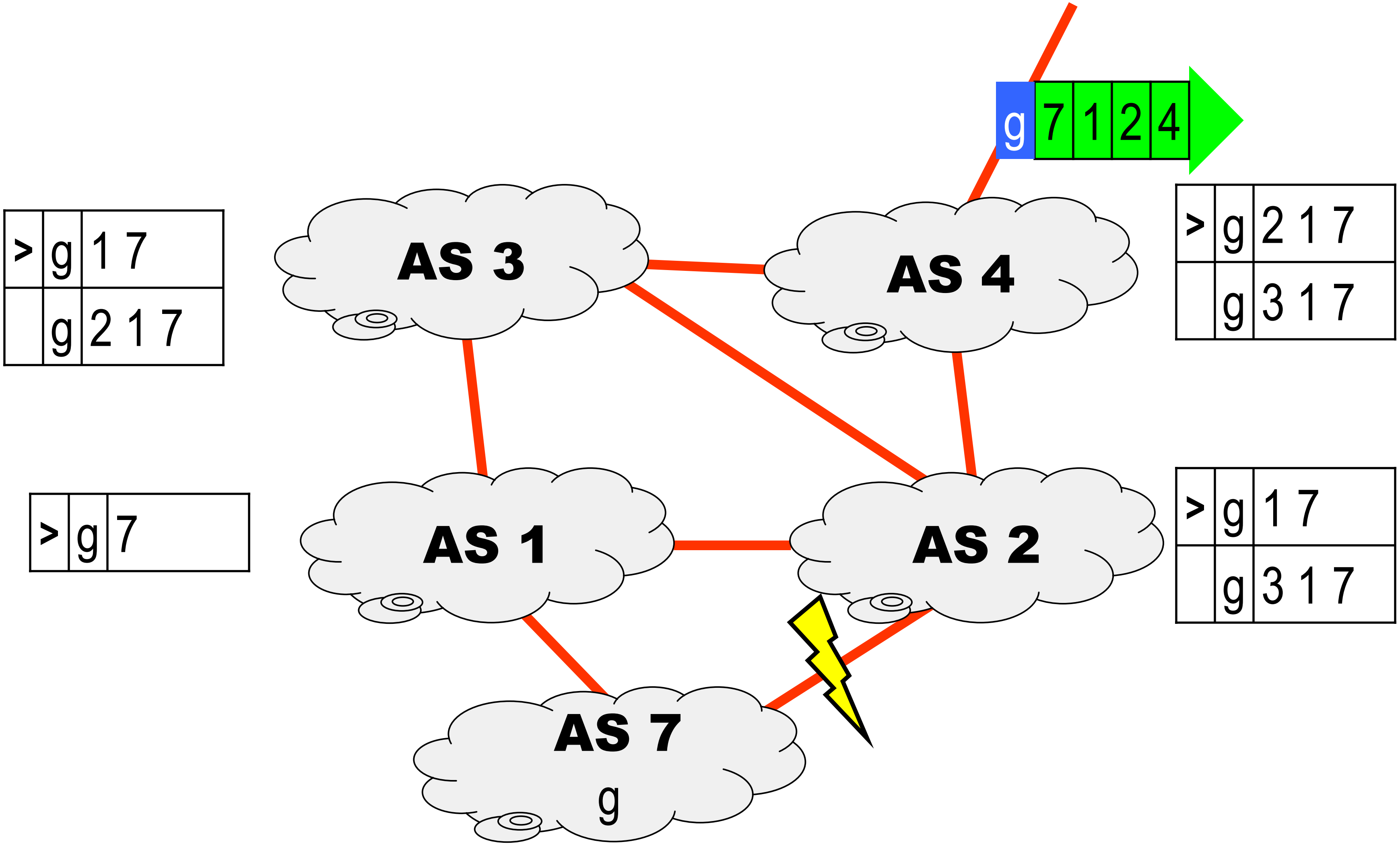
>	g	2 1 7
	g	3 1 7

>	g	7
	~~g~~	~~2 7~~

>	g	1 7
	g	3 1 7

Link Failure (Multihomed system)

- AS4 sticks decides to stick with AS2 (higher LOCAL_PREF).
- Has to advertise new route, since AS_PATH changed.



g	7	1	2	4	

>	g	1 7
	g	2 1 7

>	g	2 1 7
	g	3 1 7

>	g	7

>	g	1 7
	g	3 1 7

Route Flapping

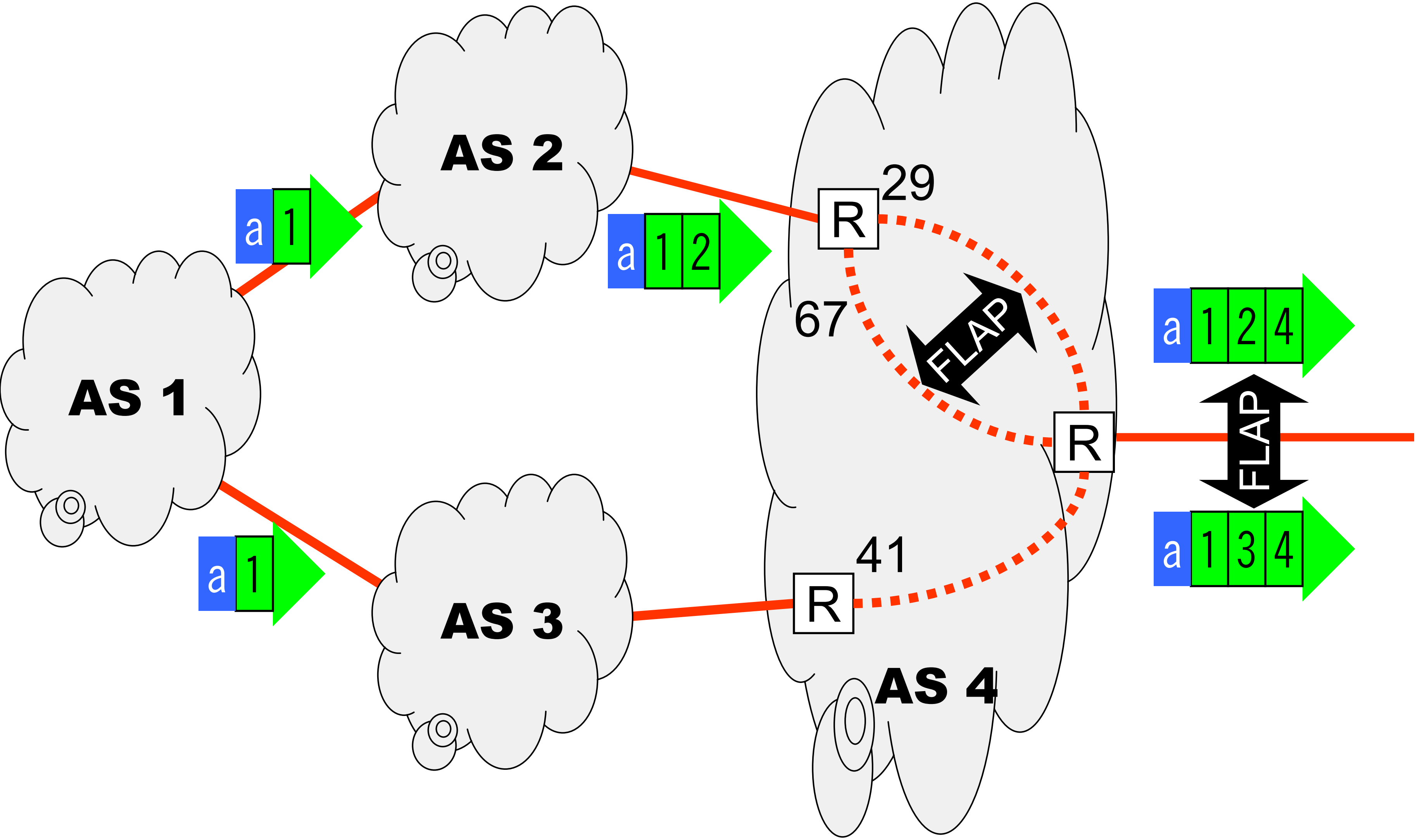
- Routing instability.
- Route disappears, appears again, disappears again…
  - Withdrawal, announcement, withdrawal, announcement…
- Visible to the entire Internet.
  - Wastes resources, triggers more instability.
- Some causes of *Route Flapping*:
  - Flaky inter-AS links.
  - Flaky or insufficient hardware.
  - Link congestion.
  - IGP instability.
  - Operator error.

Link Instability

- The first three are examples of link instability.
  - Link itself fails.
  - Router/router interface fails.
  - Messages can’t get through.
- When a link goes down, routers withdraw routes associated with this link.
  - Customer-ISP.
  - ISP-ISP.
- Announcements travel throughout the default-free zone.
- Aggregation may mask downstream flapping.
  - Does not work for multihoming

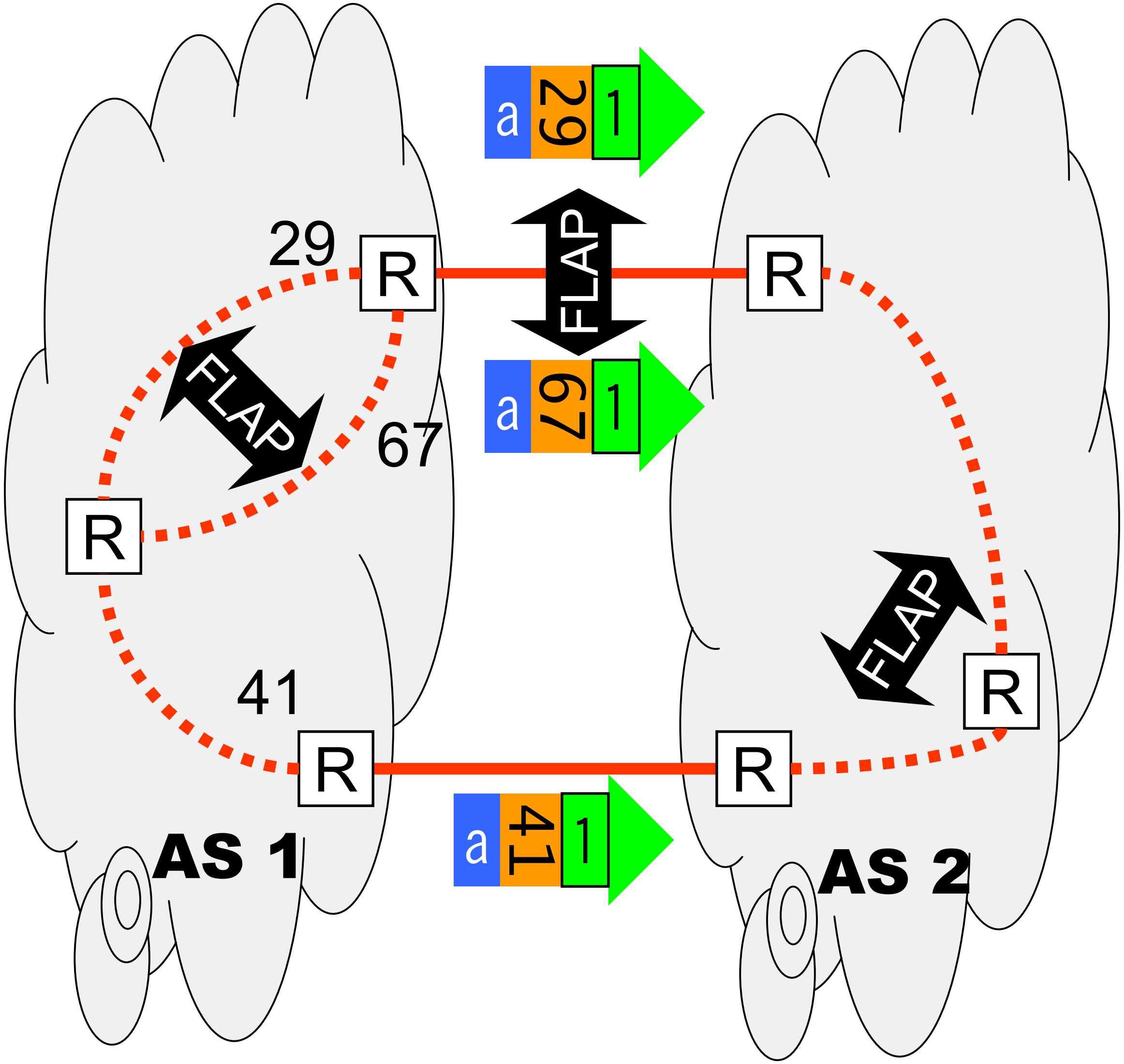
IGP Instability

- IGP route-preference rule exports instability.



IGP Instability

- MEDs can export internal instability.



Route Flap Damping

- RFC2439
- Router detects route flapping.
- *Penalty*:
  - Increased each time a route flaps.
  - Decreased over time.
- If penalty threshold exceeded (*suppress limit*), route is suppressed.
- Until penalty drops below a certain level (*reuse limit*).

- There is evidence that it may be harmful.
  - BGP explores alternate paths when a route is withdrawn.
  - Dampening merely makes the exploration run in slow motion.
  - Too aggressive.

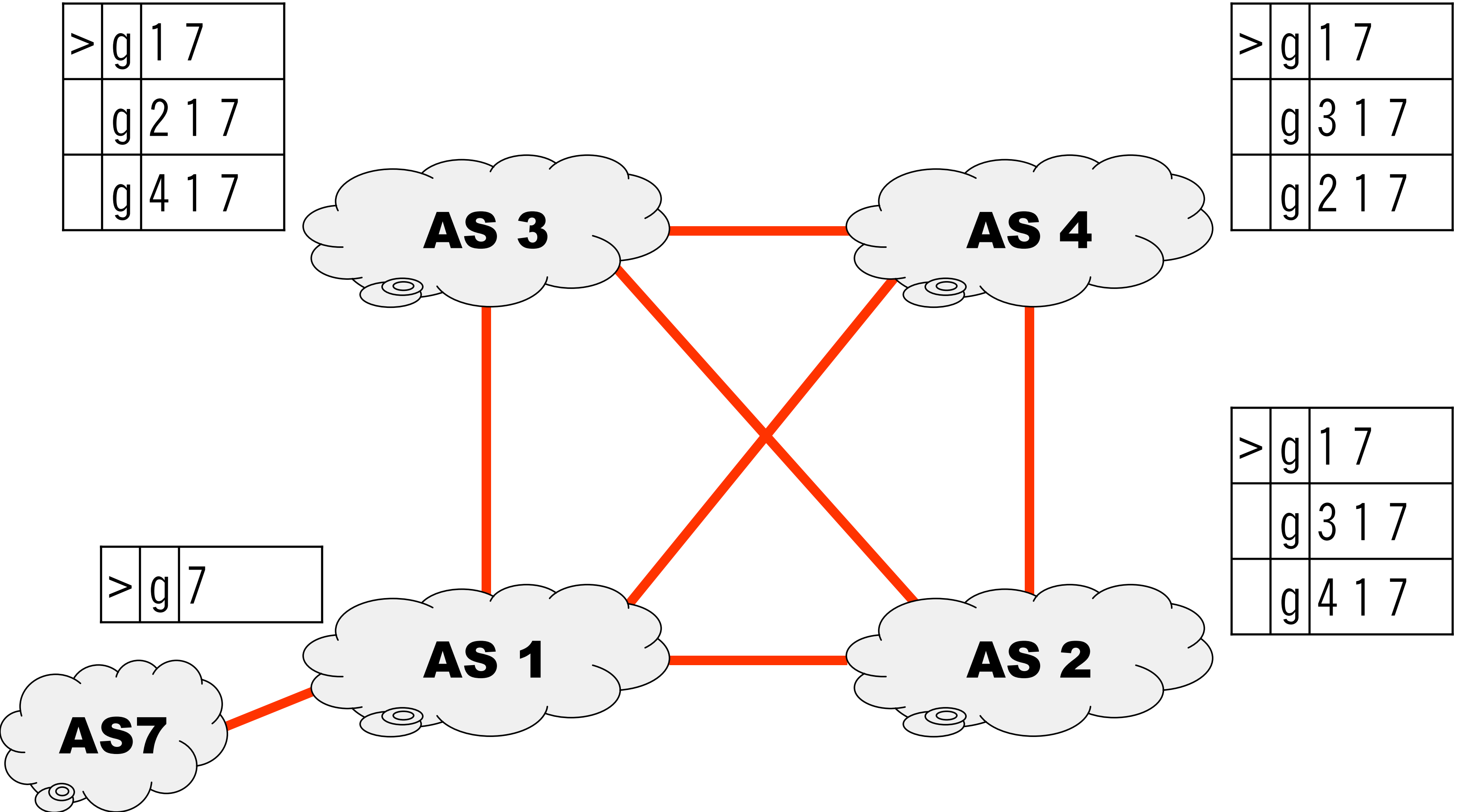
Convergence

- Link-State algorithms avoid loops by running the same computation (Dijkstra SPF) on the same data.
- Distance-Vector (Bellman-Ford-like) algorithms (e.g., RIP) avoid loops by selecting routes with a lower metric.
- Path-Vector algorithms (e.g., BGP) avoid loops by detecting self in path.

- LS converges as soon as new LSAs flooded.
- DV counts to infinity.
  - Split horizon/poison reverse/triggered updates just make the counting-to-infinity faster.
- How about BGP?

BGP Explores All Paths!

- See Labovitz *et al.*, SIGCOMM 2000.



>	g	1 7
	g	2 1 7
	g	4 1 7

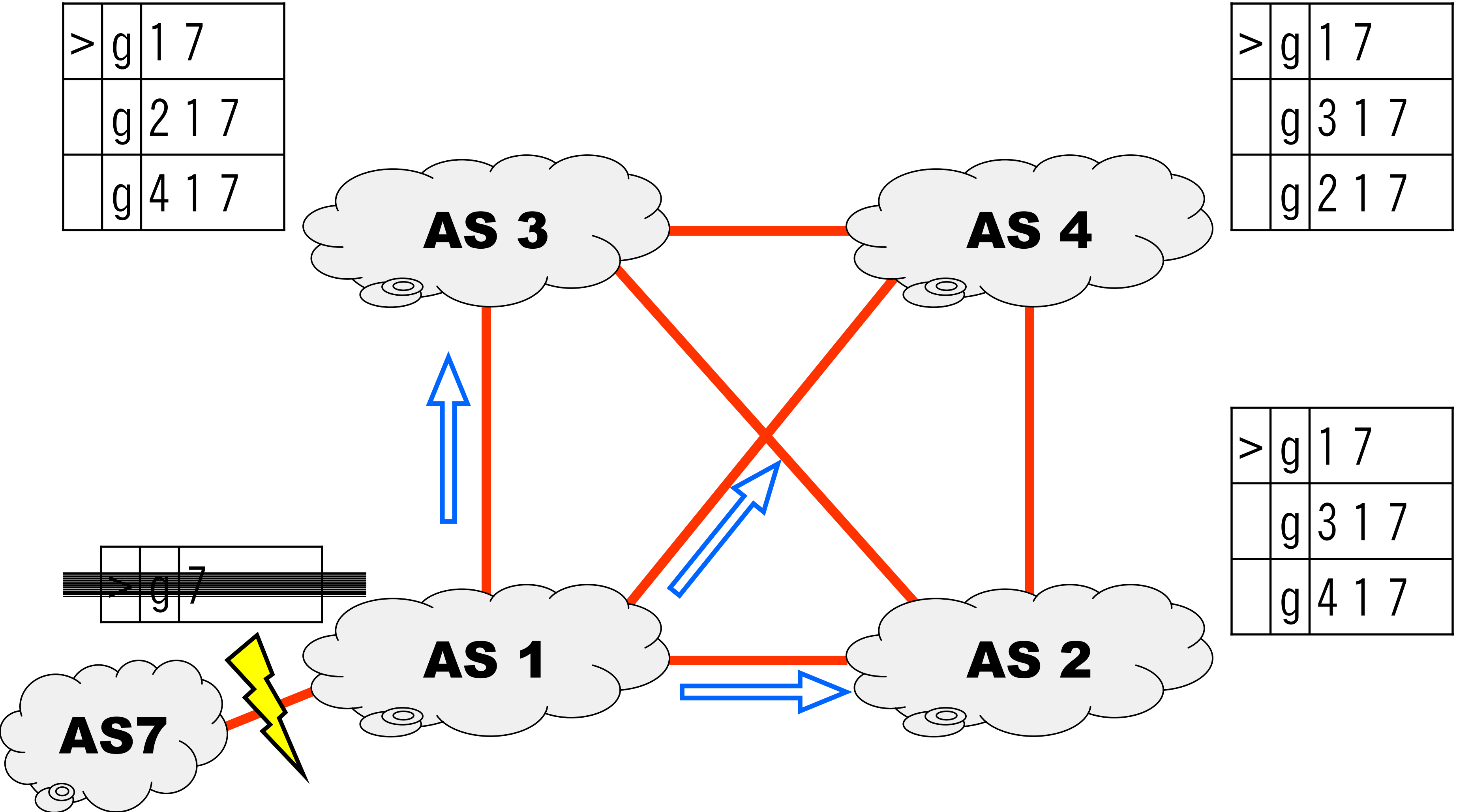
>	g	1 7
	g	3 1 7
	g	2 1 7

>	g	7

>	g	1 7
	g	3 1 7
	g	4 1 7

BGP Explores All Paths!

- Link 7-1 goes down.
- AS1 withdraws the route to prefix g.



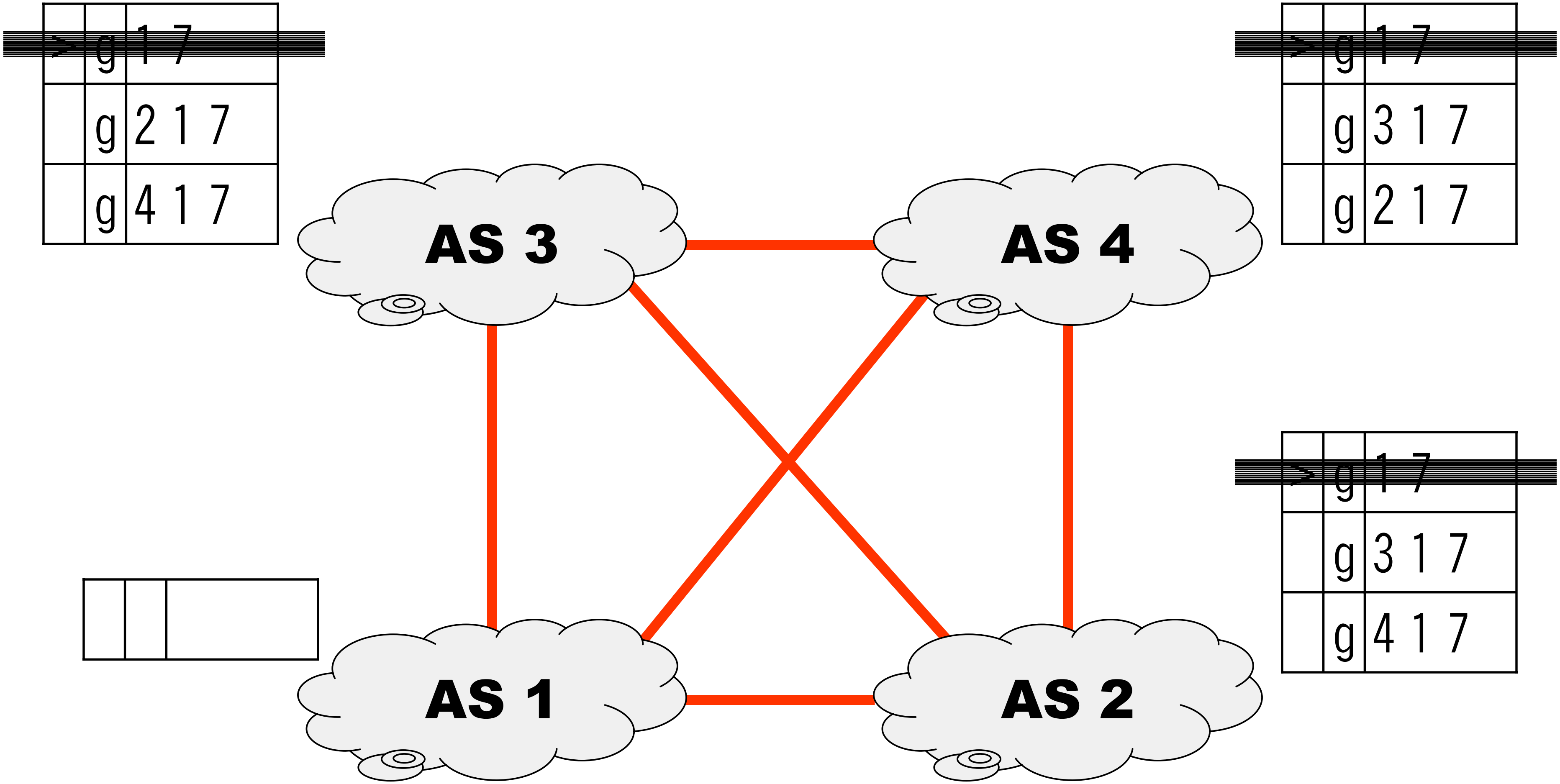
>	g	1 7
	g	2 1 7
	g	4 1 7

>	g	1 7
	g	3 1 7
	g	2 1 7

>	g	1 7
	g	3 1 7
	g	4 1 7

~~>~~	~~g~~	~~7~~

- AS 2, 3, 4 remove [1 7] route.

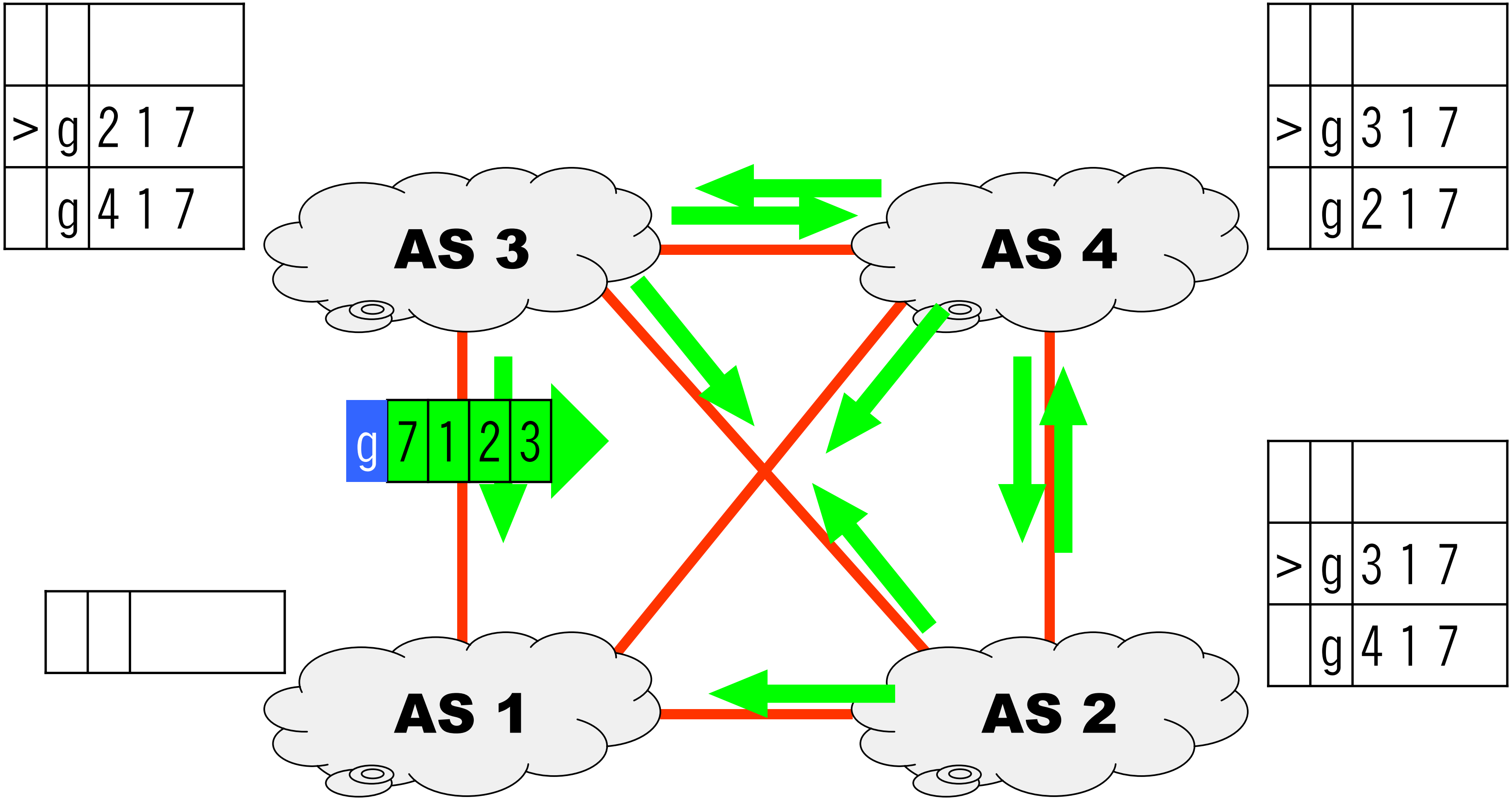


~~>~~	~~g~~	~~1 7~~
	g	2 1 7
	g	4 1 7

~~>~~	~~g~~	~~1 7~~
	g	3 1 7
	g	2 1 7

~~>~~	~~g~~	~~1 7~~
	g	3 1 7
	g	4 1 7

- Select their next best route.
- Advertise it.



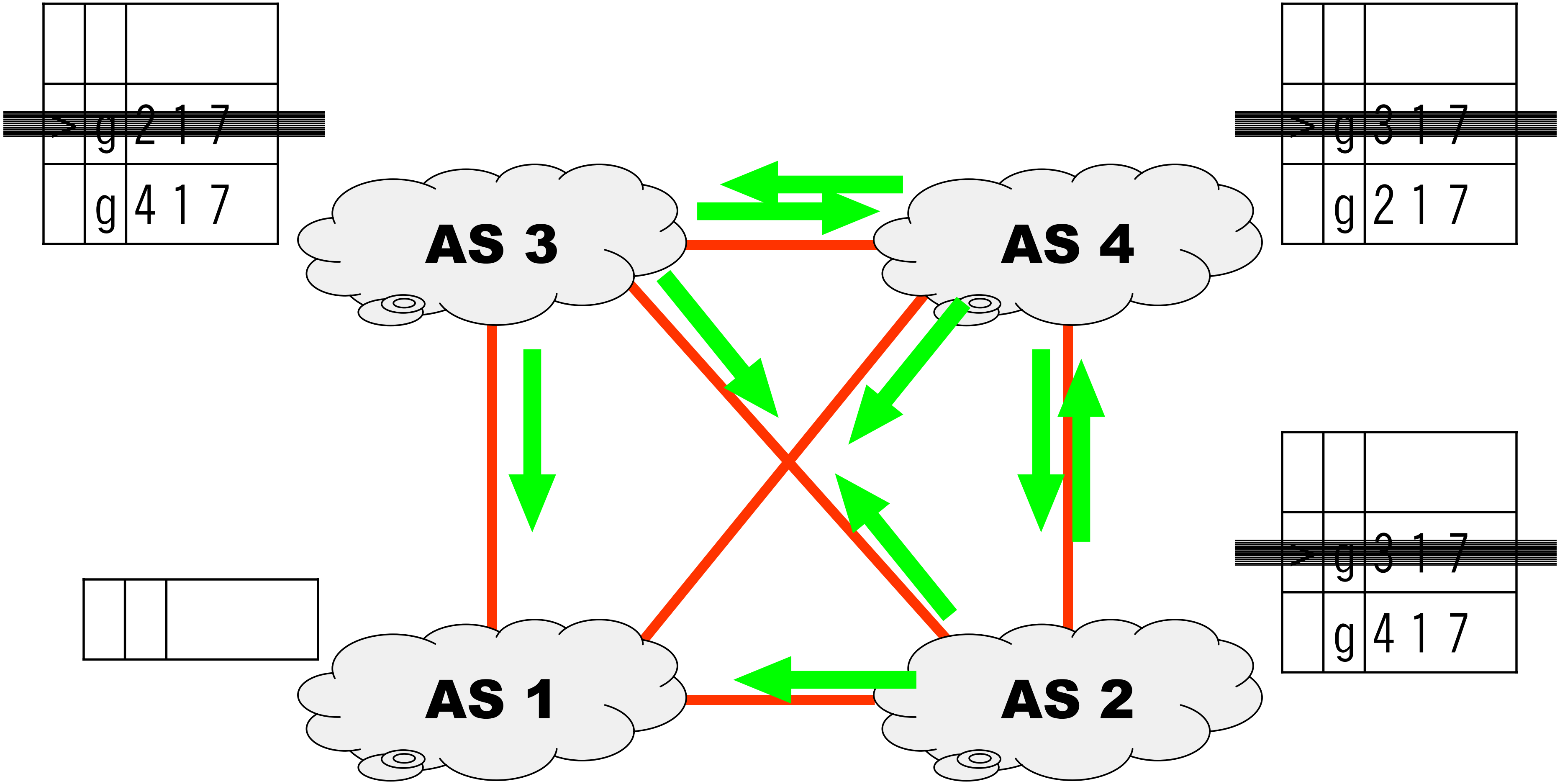
>	g	2 1 7
	g	4 1 7

>	g	3 1 7
	g	2 1 7

>	g	3 1 7
	g	4 1 7

g	7	1	2	3

- AS1 ignores the routes it gets (self in AS_PATH).
- (e.g.) AS2 gets [3 2 1 7] from AS3; treats it as implicit withdrawal of [3 1 7], then rejects it (self in AS_PATH).
- Process repeats one more time, then all ASes lose their routes to g.



~~>~~	~~g~~	~~2 1 7~~
	g	4 1 7

~~>~~	~~g~~	~~3 1 7~~
	g	2 1 7

~~>~~	~~g~~	~~3 1 7~~
	g	4 1 7

BGP Explores n! Paths (cont’d)

- Problem was exacerbated by MinRouteAdvertisementInterval.
- Routers would wait 30 seconds before sending next set of updates.
- Common perception at the time was “BGP converges within 30 seconds”.
- There were paths that took over 15 minutes to converge.

- This sort of behavior creates routing traffic without always benefiting connectivity.

- Lots of other sources of instability.

BGP Conclusion

- Protocol (deceptively) simple.
- Lots of accumulated current practices.
- It mostly works.
- But for how much longer?