Inter-domain routing and BGP

Olof Hagsand KTH/CSC

Literature

- Practical BGP
  - Follow reading instructions
- RFC 4271
- Many vendor pages

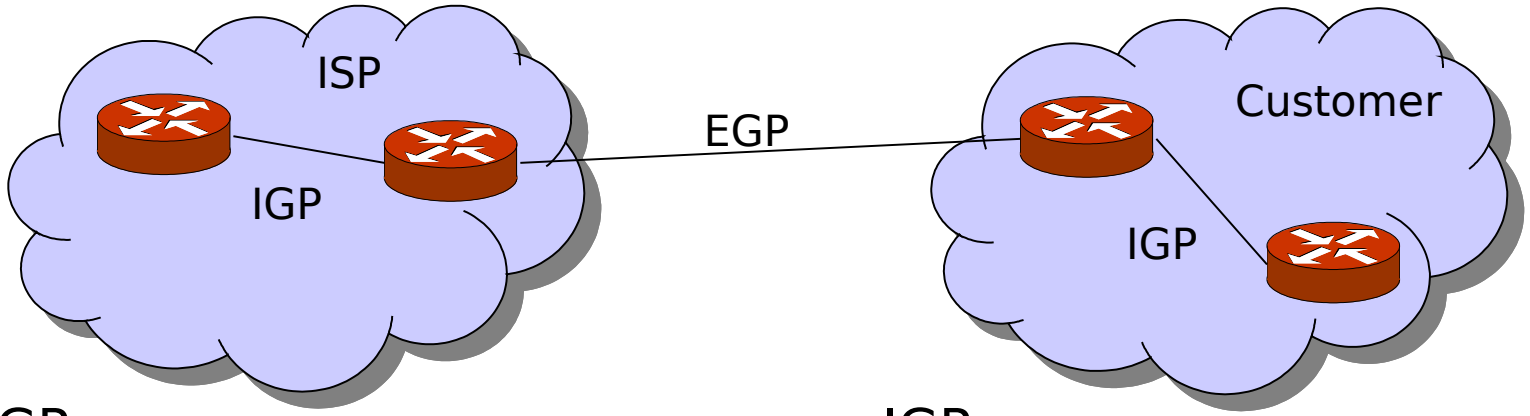
Inter-domain routing

- The objective of inter-domain routing is to bind together all the thousands of independent IP networks that constitute the Internet
- Perspective from one network
  - Decide how to receive information (packets) from the outside world
  - Decide how to spread information to the outside world
- Handling of prefixes
  - Receive and choose (filter) between prefixes from other domains
  - Announce prefixes to other domains
- Address aggregation

What is BGP?

- Border Gateway Protocol version 4
- Defined in RFC 4271
- An inter-domain routing protocol
- Uses the *destination-based* forwarding paradigm
  - No other relations can be expressed: sources, tos, link load
- Uses *path-vector* routing
- Views the Internet as a collection autonomous systems
- Exchanges information between *peers* using TCP as underlying protocol
- Maintains a database (RIBs) of *network layer reachability information* (NLRI:s)
- Supports a toolkit of mechanisms to express and enforce policies decisions at the AS level

IGP/EGP



EGP

- Exterior Gateway Protocol.
- Runs between networks/domains (inter-domain)
- Examples: BGP, static routing
- Note that BGP can also run internally in a network: IBGP

IGP

- Interior Gateway Protocol.
- Runs within a network/domain (intra-domain)
- Examples: RIP, OSPF, IS-IS.

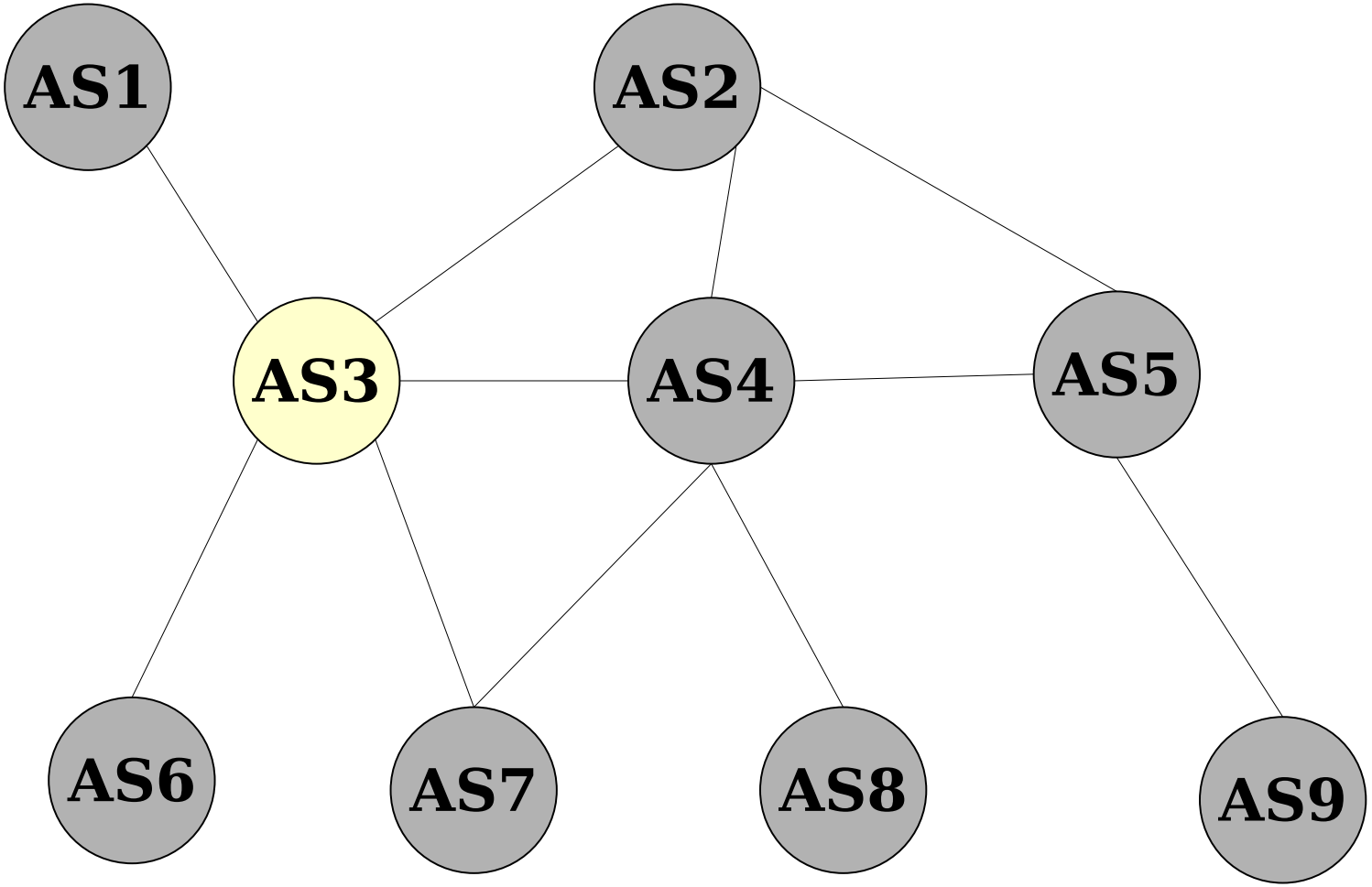
Why cant we use an IGP?

- On page 3 in the book, there is a chapter:
  - Why not use a single protocol for both internal and external routing?

Autonomous Systems (AS)

- A set of routers that has a single routing policy, that run under a single technical administration
  - A single network or group of networks
  - University, business, organization, operator
- This is viewed by the outside world as an Autonomous System
  - All interior policies, protocols, etc are hidden within the AS
- Represented in the Internet by an Autonomous System Number (ASN). 0-65535
  - Example: ASN 1653 for SUNET
- Currently, operators are switching to four-byte ASNs
  - RFC 4893: BGP Support for Four-octet AS Number Space

General AS graph



Whois example

```
gelimer.kthnoc.net> whois -h whois.ripe.net AS1653
aut-num:        AS1653
as-name:        SUNET
descr:          SUNET Swedish University Network
import:         from AS42 accept AS42
export:         to AS42 announce AS-SUNET
import:         from AS702 accept AS702:RS-EURO AS702:RS-CUSTOMER
export:         to AS702 announce AS-SUNET
import:         from AS2603 accept any
export:         to AS2603 announce AS-SUNET
import:         from AS2831 accept AS2831 AS2832
export:         to AS2831 announce any
import:         from AS2833 accept AS2833
export:         to AS2833 announce any
import:         from AS2834 accept AS2834
export:         to AS2834 announce any

gelimer.kthnoc.net> whois -h whois.ripe.net AS-SUNET
as-set:         AS-SUNET
descr:          SUNET AS Macro
descr:          ASes served by SUNET
members:        AS1653, AS2831, AS2832, AS2833, AS2834, AS2835, AS2837
members:        AS2838, AS2839, AS2840, AS2841, AS2842, AS2843, AS2844
members:        AS2845, AS2846, AS3224, AS5601, AS8748, AS8973, AS9088
members:        AS12384, AS15980, AS16251, AS20513, AS25072, AS28726
members:        AS-NETNOD
```


(Edited example)

ISP Tiers

Tier-1 (TeliaSonera, Sprint, Verizon, NTT, Level3, Google,...)

- Huge ISPs. They do not pay anyone else for transit, since they exchange traffic (peer) with all other tier-1 networks. Everyone else pays to peer with tier-1 ISPs, to get connectivity the whole Internet.

Tier-2

- Large regional ISPs. Do not peer with all tier-1 networks. Buys transit from a few tier-1 network.

Tier-3

- Smaller ISPs. Buys transit form either tier-1 or tier-2 networks. Exchanges traffic at a single IX.

RIPE

- All public IP addresses and AS numbers is provided by IANA (Internet Assigned Numbers Authority). www.iana.org.
  - IANA also handles the top level domains and protocol numbers.
- IANA delegate IP blocks and AS numbers to RIRs (Regional Internet Registry) so they in turn can delegate space to LIRs (Local Internet Registry).
- RIPE NCC (Réseaux IP Européens Network Coordination Centre) is the European RIR
  - Also handles non European countrys like Israel, Iraq, Iran, Russia and many more.
  - www.ripe.net

RIPE

- RIPE NCC is one of five RIRs in the world
  - ARIN, American Registry for Internet Numbers, www.arin.net
  - LACNIC, Latin American and Caribbean Internet Addresses Registry, www.lacnic.net
  - APNIC, Asia Pacific Network Information Centre, www.apnic.net
  - AfriNIC, the African Network Information Centre, www.afrinic.net

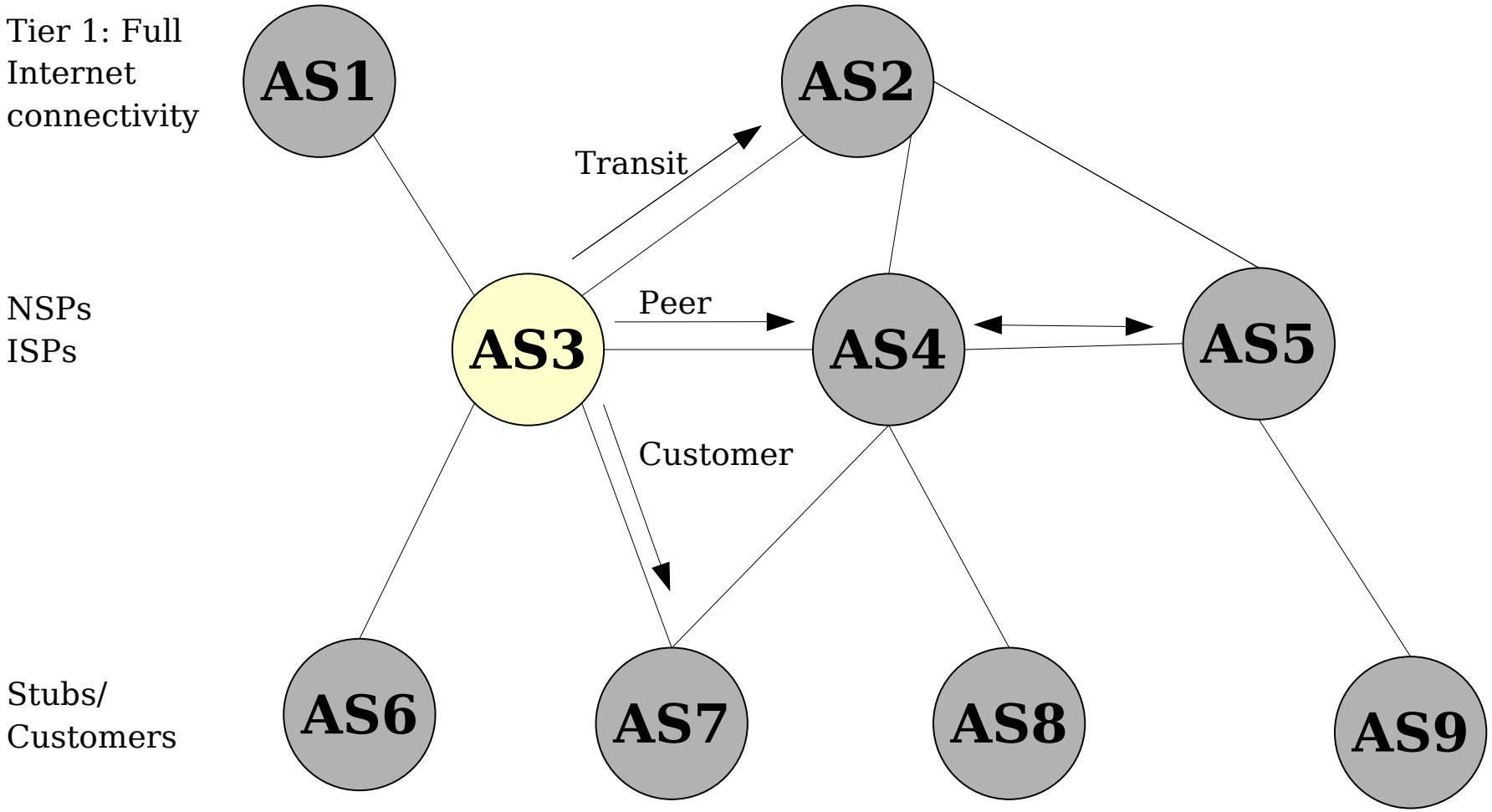
RIPE

- There are two “classes” of IP addresses
  - Provider Independent (PI)
  - Provider Aggregatable (PA)
- PA space is block of prefixes delegated to a LIR (Local Internet Registry) and to be used by themselves and their customers.
  - If you use PA space and decide to change ISP the address space have to be returned.

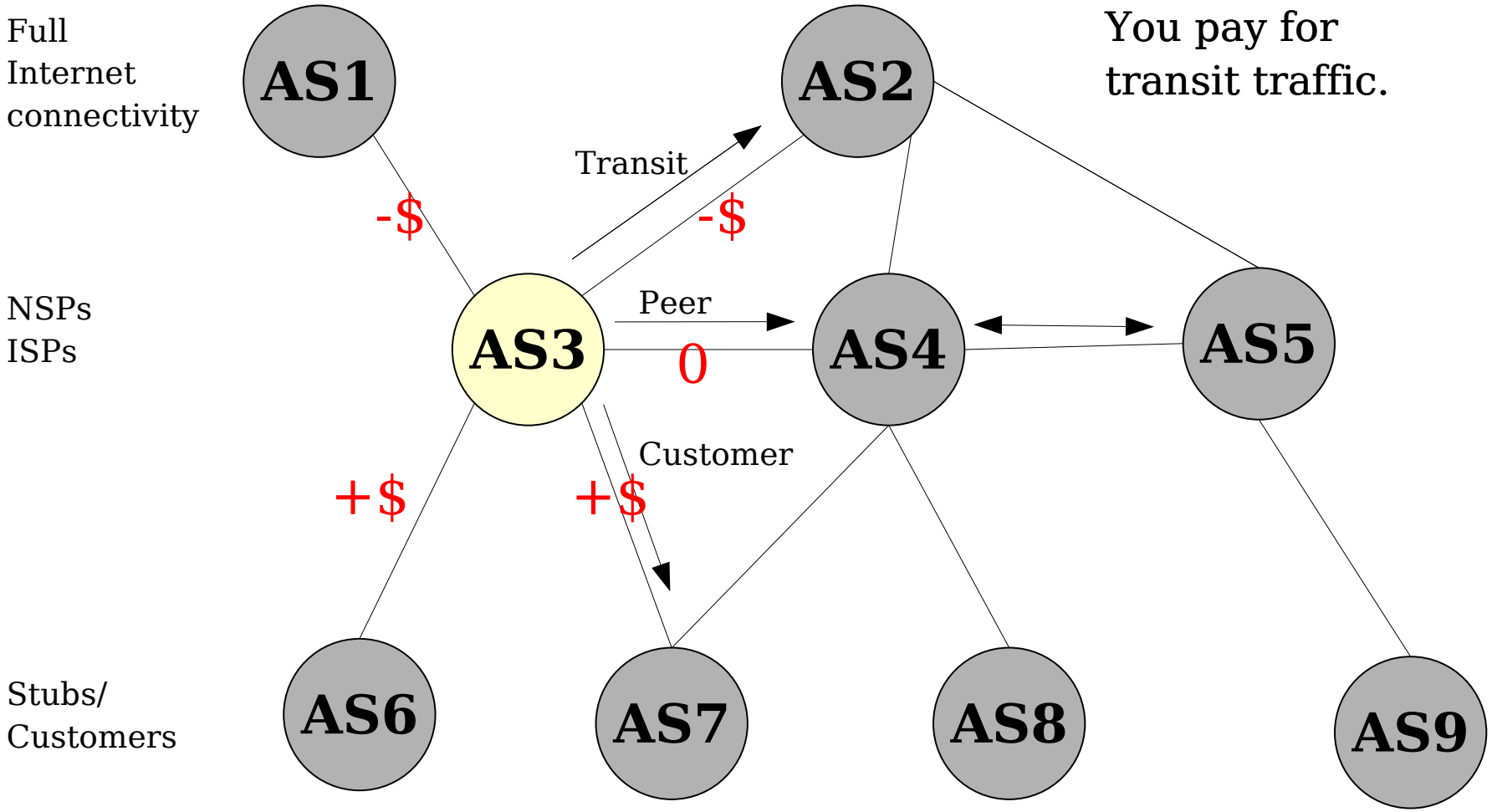
RIPE

- PI space is addresses that end customers can request directly from RIPE.
  - Good for the end customer, but bad for the ISP.
  - RIPE will also start charging for PI space
- You have to make a strong case against RIPE to get the requested amount of IP addresses.
  - There are organizations that can help you with the paper work towards RIPE.

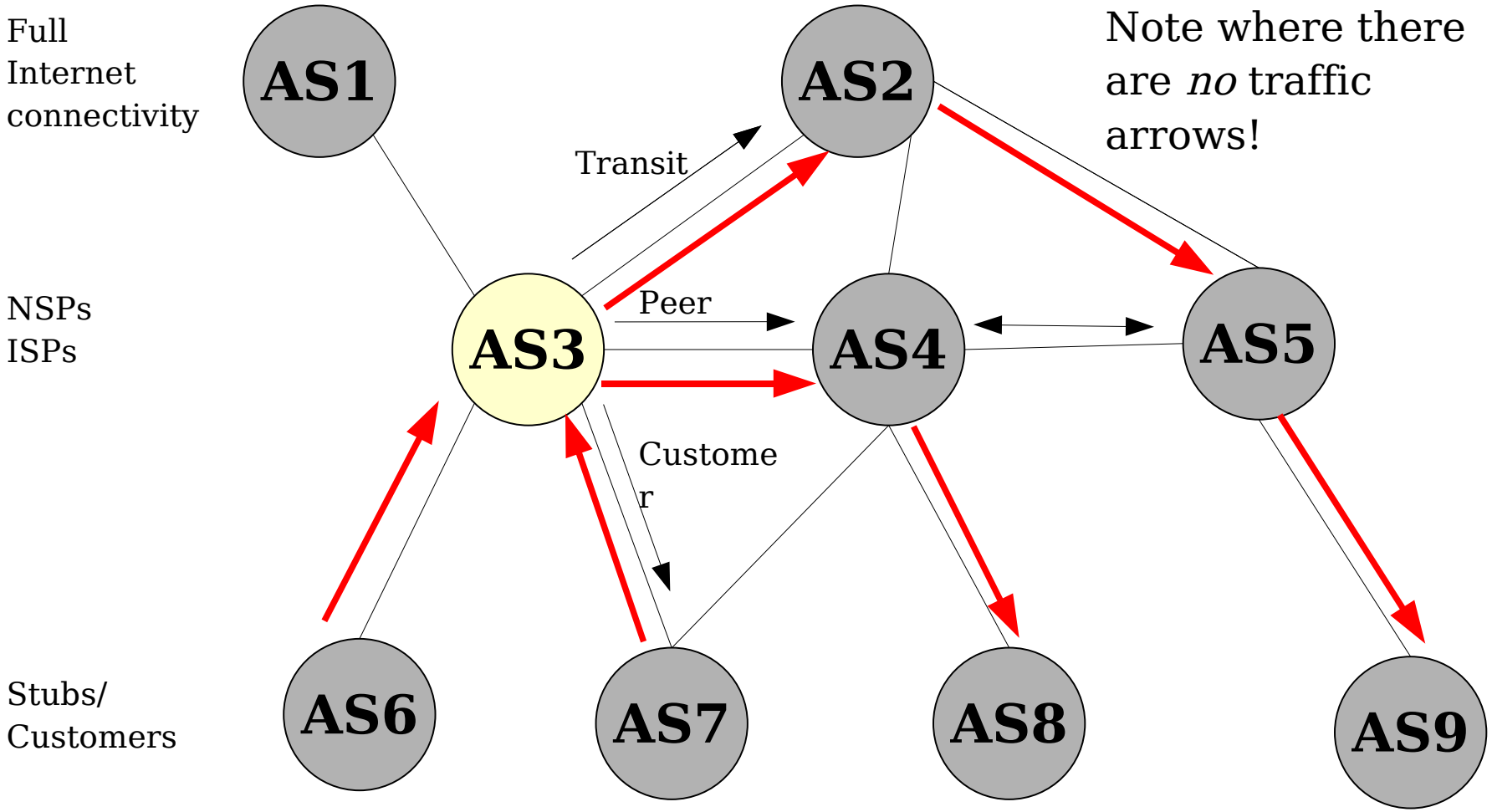
AS graph and peering relations



Cost and peering relations



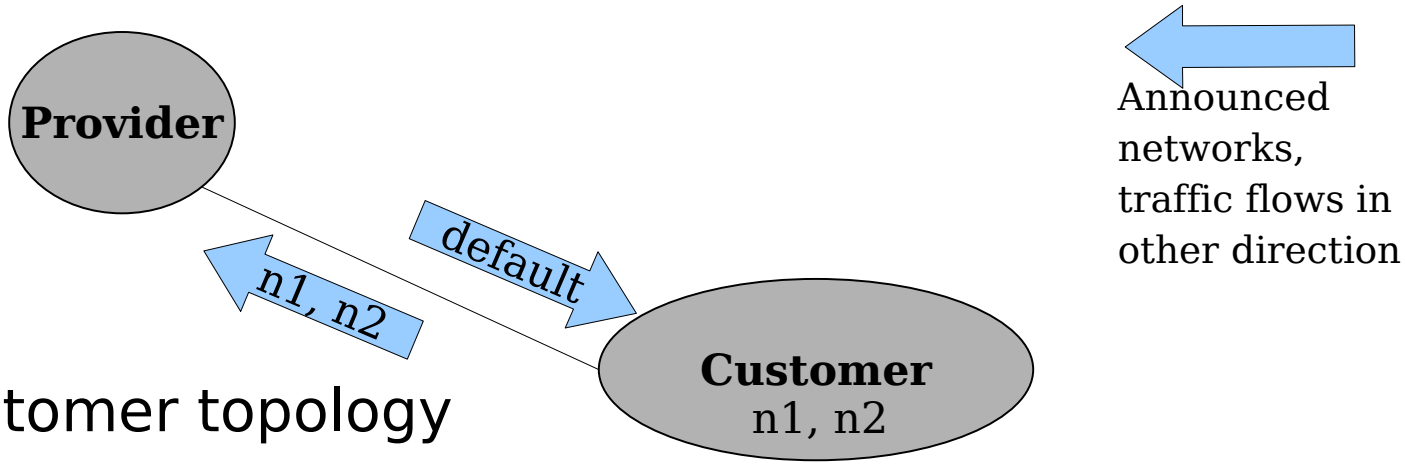
Traffic patterns



Peering relations

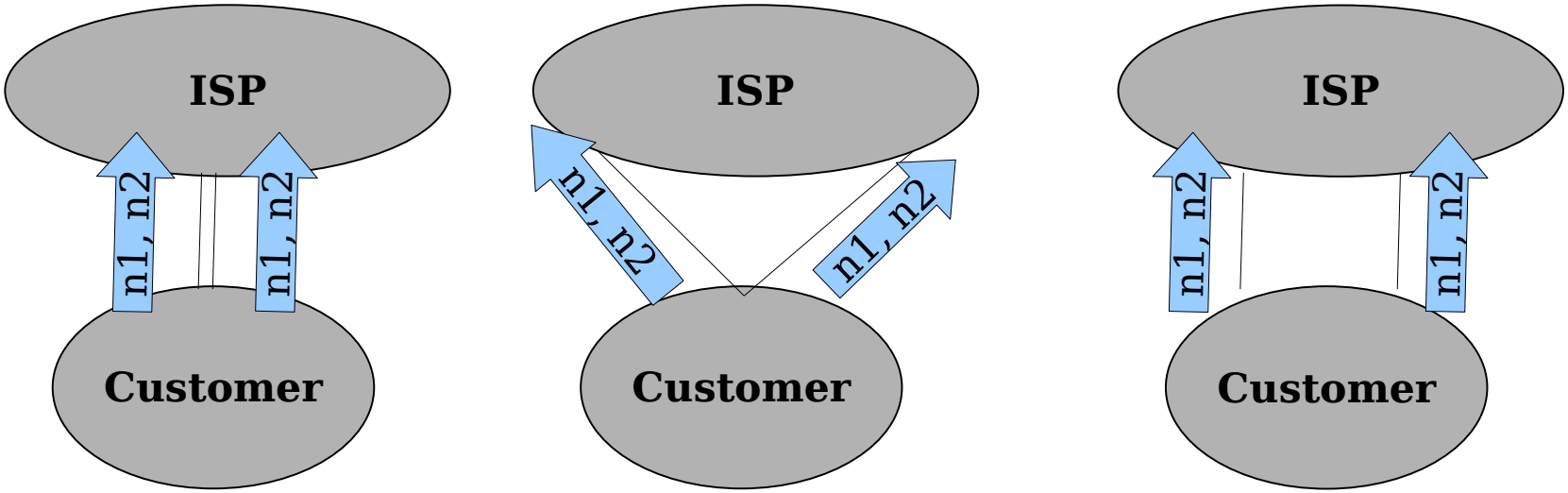
- An abstract way of defining peering relations is for example:
- Prefix sets:
  - Define a customer set, a peering set and a transit set
- Example rules:
  - Customer prefixes should be announced to transit and peers
  - Peer and transit prefixes should be announced to customers
  - Prefer prefixes from peers over prefixes from transit
  - Do not accept illegal prefixes (RFC 1918 for example), or unknown prefixes from customers
  - Load balance over several transit providers
  - Filter traffic (eg src addresses) according to the prefixes announced

Customer / ISP Relations: Stub AS



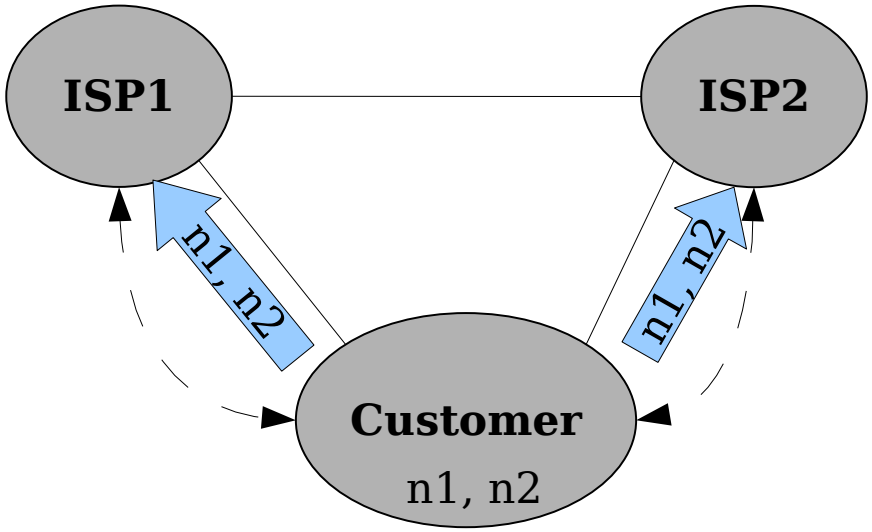
- Typical customer topology
- Can use default route to reach the Provider and Internet
- Customer can use address block of provider
- Customer does not need to be a separate AS
- Typically use static routing but can also use BGP
  - Less common: Use a separate IGP (eg RIP) only to exchange routes between border routers.

Multi-homed customer



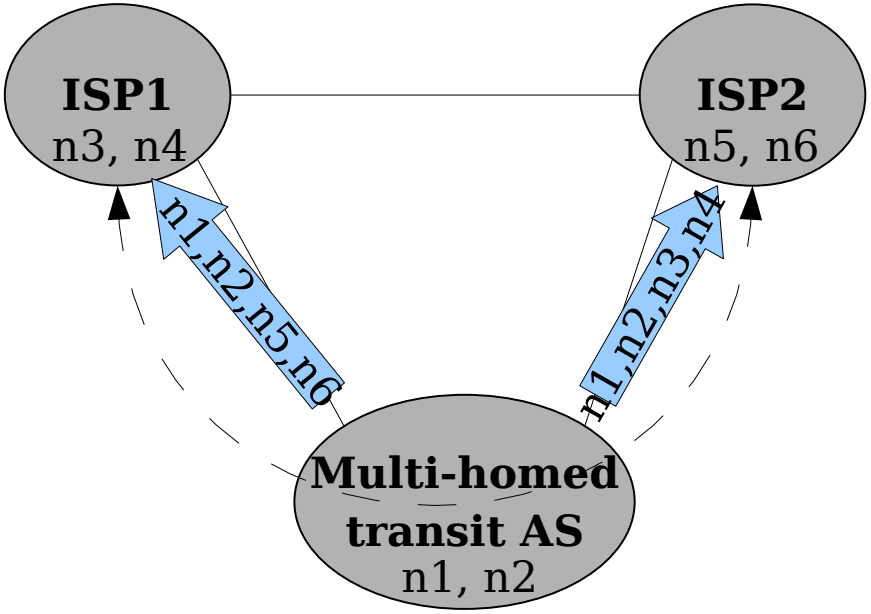
- Customer can be multi-homed for reliability and performance reasons
  - Load sharing or geographical traffic distribution
- Non Transit AS does now allow external traffic to pass through
- Multi-homed non-transit AS
- You have to think about: How to announce the prefixes? Default routes? Symmetrical routing? Packet filtering, address aggregation, etc

Customer with multiple providers



- For several upstream providers address aggregation is an important issue:
- Which address block should the Customer use?
  - From ISP1 or ISP2?
  - From both?
  - Or an independent address block?

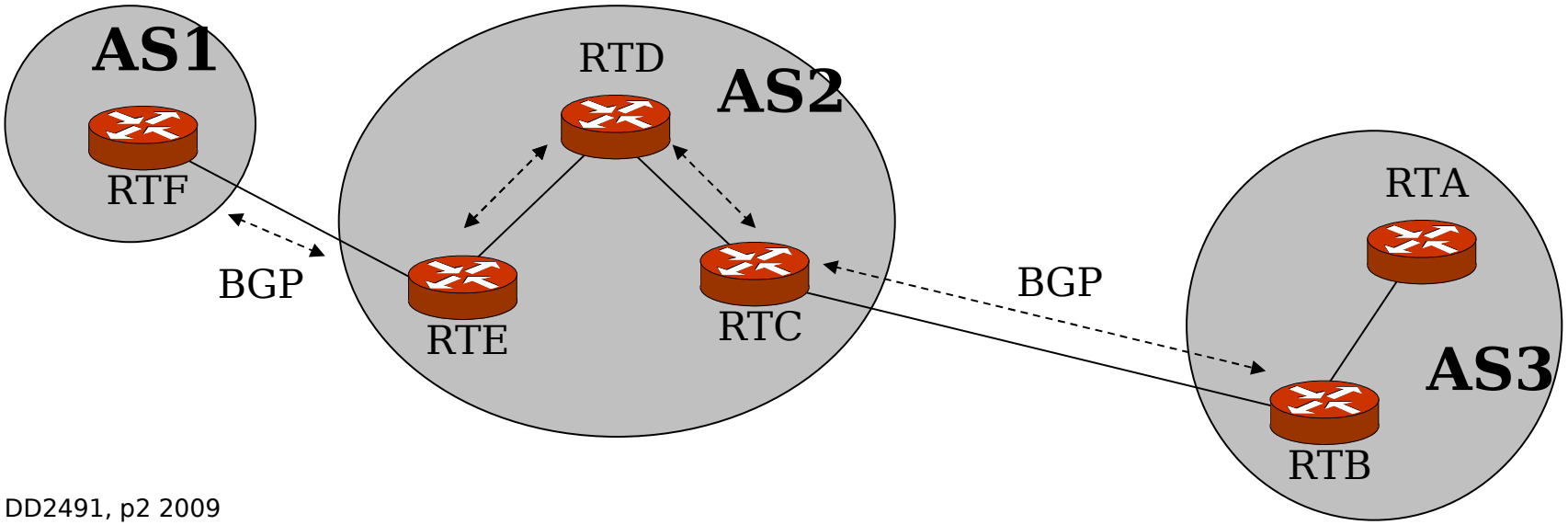
Provider: Multi-homed Transit AS



- Transits traffic within own network
- This the most general configuration and is how a provider works.

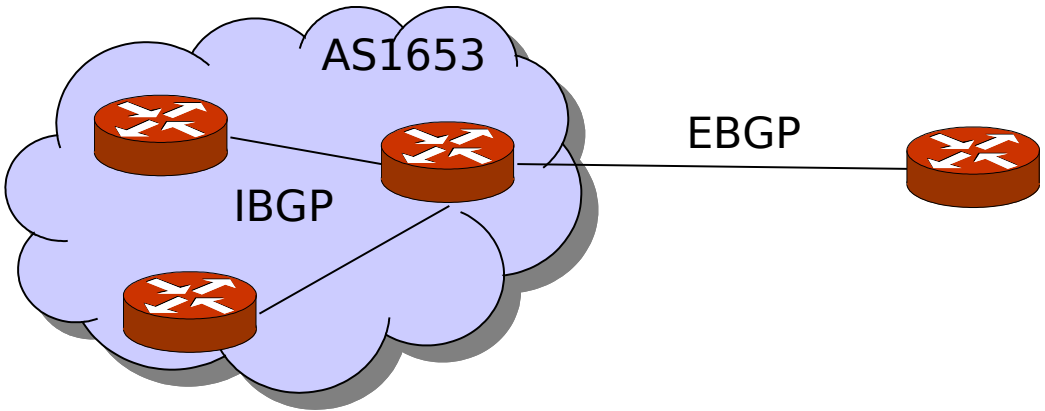
BGP sessions

- BGP connections (peerings) are setup manually using TCP
- Two peers must have IP connectivity
- Things to think about (see RTE):
  - How are routes imported into AS2?
  - How are routes propagated to AS3?
  - Which are the BGP nexthops?
  - How is external traffic sent through AS2?



Example JunOS configuration

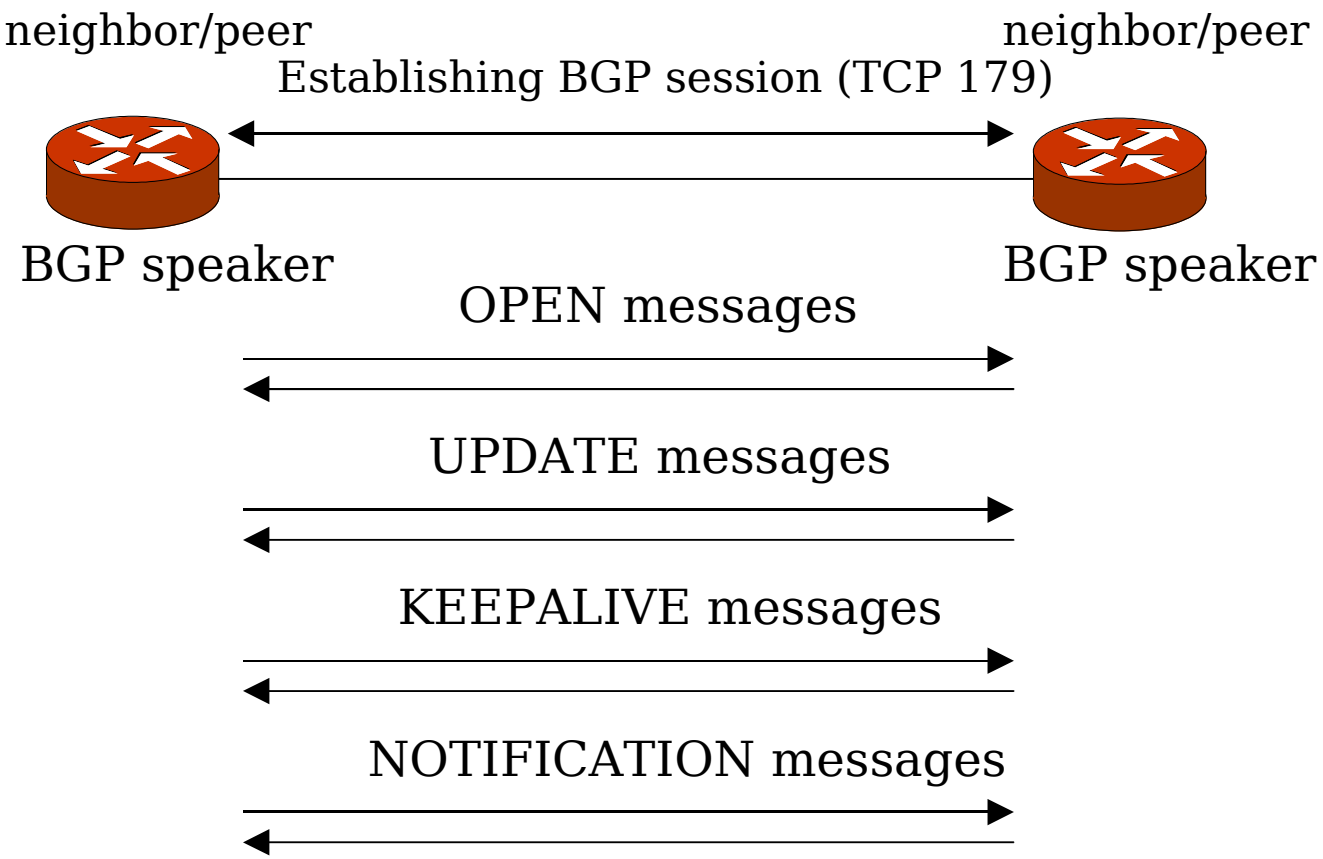
```
routing-options {
    autonomous-system 1653
}
protocol bgp {
    group external-peers {
        type external;
        peer-as 42;
        neighbor 192.168.200.13;
    }
    group internal-peers {
        type internal;
        local-address 192.168.24.1;
        neighbor 192.168.16.1;
        neighbor 192.168.6.1;
    }
}
```



Path vector protocol

- In a distance-vector protocol, vectors with destination information are distributed between routers:
- Example:
  - <dst: 10.1.10/24, metric: 5, nexthop: 10.2.3.4>
- Distance-vector has problems with converging
  - Example: count-to-infinity
- Path-vector extends the information with a *path* to the destination
  - This enables immediate *loop detection*
  - Several other attributes associated with path
- Also, in BGP, the path vector uses AS:s, not IP addresses
  - This hides internal structure in the domains
  - *Loop detection only on AS-numbers!*
- Example: <dst: 10.1.10/24, path: AS1:AS3:AS5, nexthop: 10.2.3.4>

BGP Operation



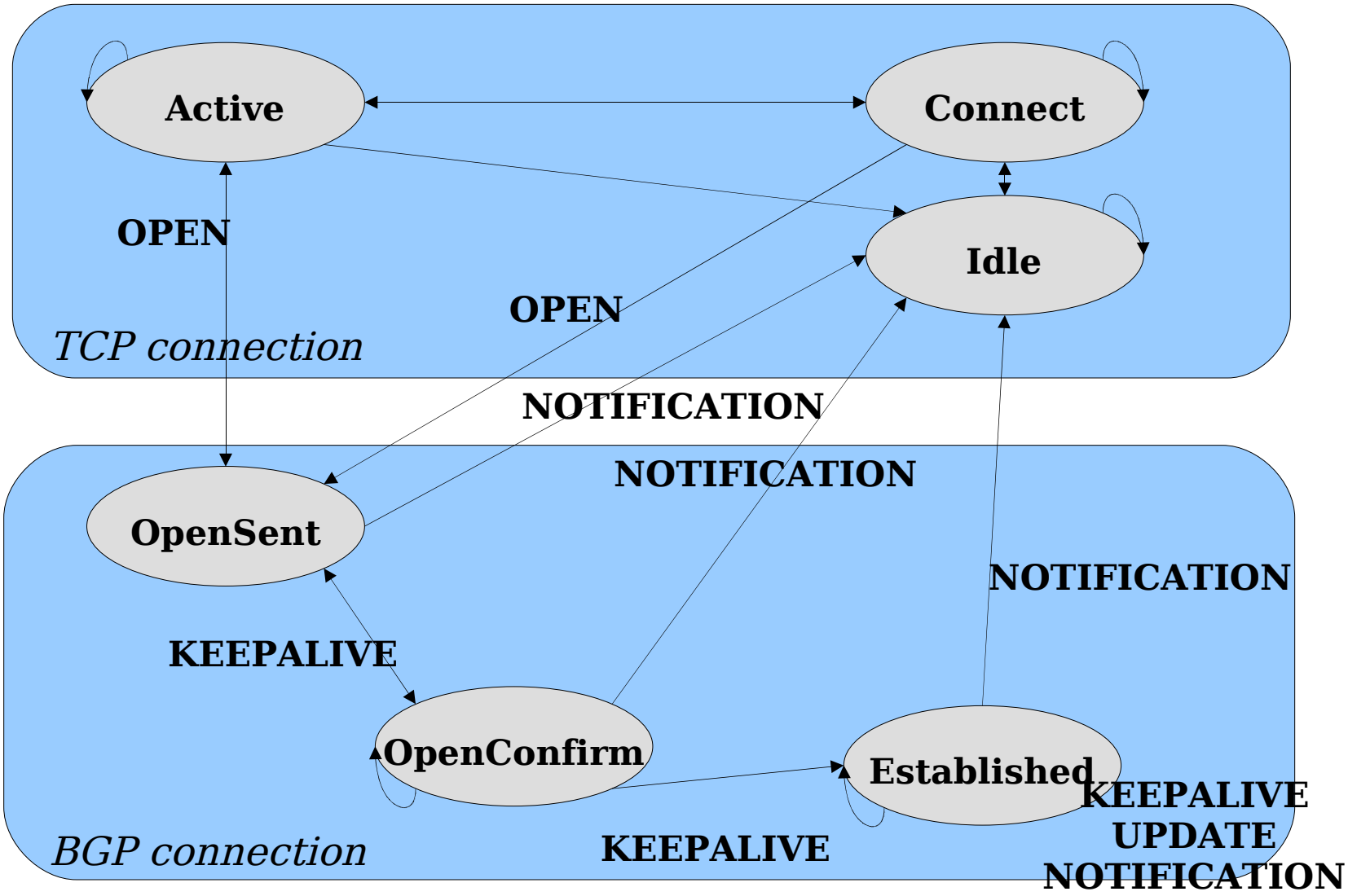
BGP protocol operation

- OPEN messages to initiate a connection and exchange capabilities
- UPDATES contain
  - A set of path attributes
  - A set of prefixes sharing the path attributes
  - A set of withdrawn routes
- BGP compares the AS path and other attributes to select the best path for a prefix
  - Same prefix may be received from several peers
- Path attributes describes properties of the route
  - How it was generated, which is the nexthop, various metrics, etc
- NOTIFICATION to signal errors
- KEEPALIVE to check liveness of peer

BGP Connections

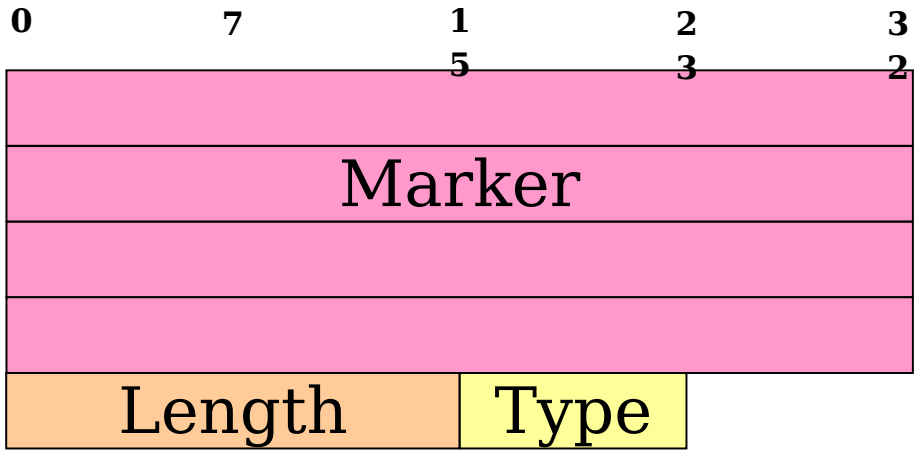
- All updates are incremental
  - Basic case: No refreshes
  - Extension: RFC 2918: Route Refresh Capability fro BGP-4
- This assumes reliable delivery!
- Therefore BGP runs over TCP
  - Fragmentation,
  - Acks,
  - Flow control,
  - Congestion control
  - Byte stream
  - No automatic neighbour discovery
- This assumes IP connectivity between peers!
  - But via other mechanism than BGP
  - Either IGP, static, or directly connected.
- So BGP connections can rely on an IGP
- BGP does not use the TCP keepalives (which by default is on the order of hours)

BGP Finite State Machine



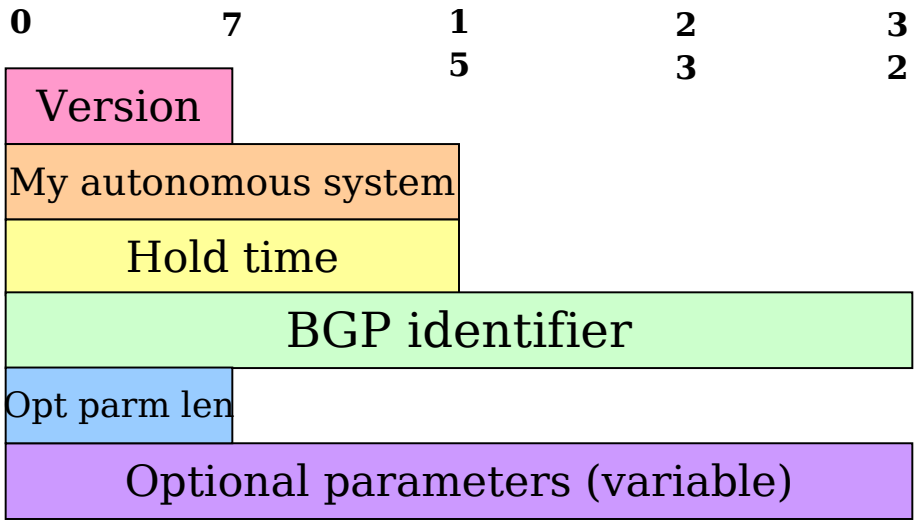
BGP message header

- BGP message header format



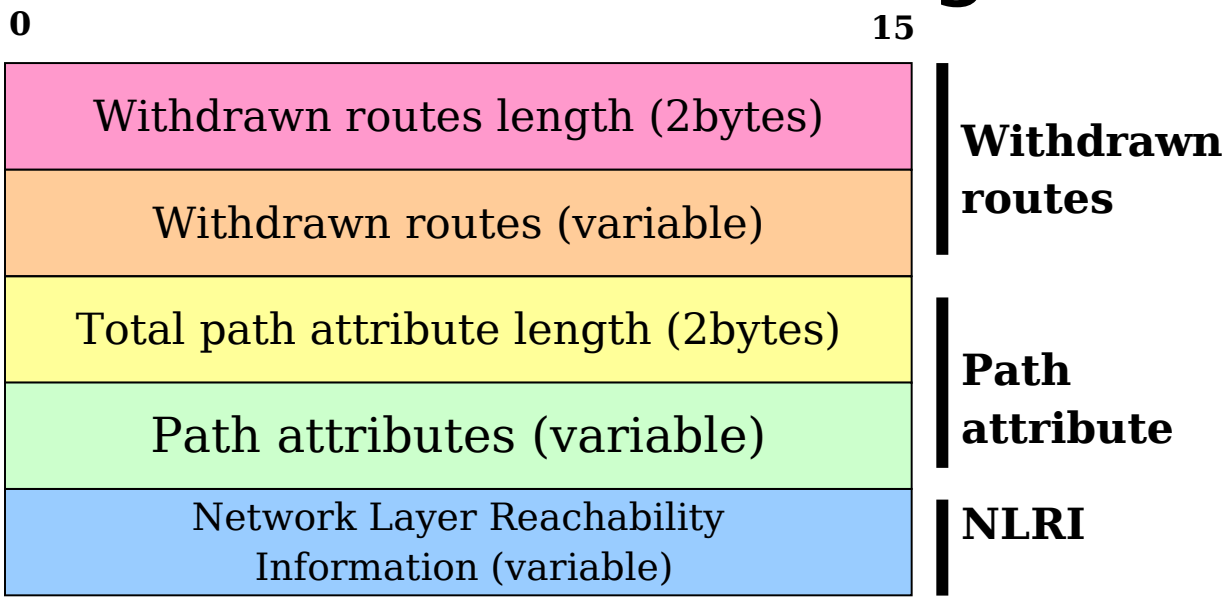
- Marker field:
  - Authentication of incoming BGP messages
  - Detect loss of synchronization between two BGP peers
- Length field: total message length including the header
- Type field: indicates the message type

OPEN message



- Version: version of BGP message (current is 4)
- My AS: ASN of the BGP speaker
- Hold Time: Maximum interval between KEEPALIVE or UPDATE messages
- BGP Identifier: Sender’s BGP ID
- Optional Parameter length: total length of the Optional Parameter field
- Optional Parameter: use in BGP session negotiation

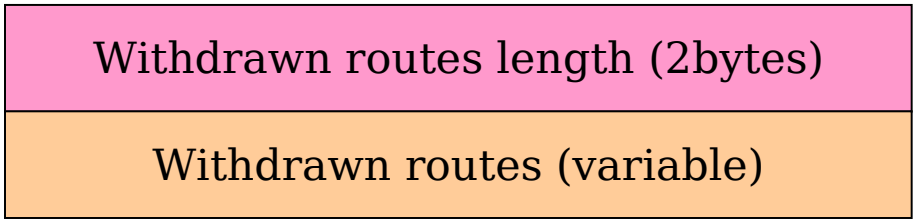
UPDATE message



- 3 basic blocks of UPDATE message:
  - Withdrawn routes
  - Path attributes
  - Network Layer Reachability Information (NLRI)

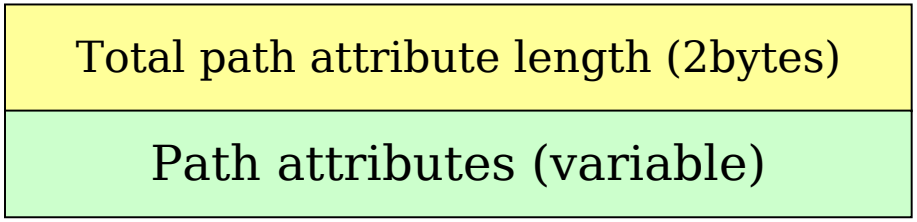
Withdrawn Routes

- Withdrawn Routes Length: total length of Withdrawn Routes field
  - 0 means no routes being withdrawn and Withdrawn Routes field is not present in this UPDATE message
- Withdrawn Routes field: contains list of prefixes that are being withdrawn
  - Each prefix is encode as a 2-tuple of <length, prefix> (CIDR)



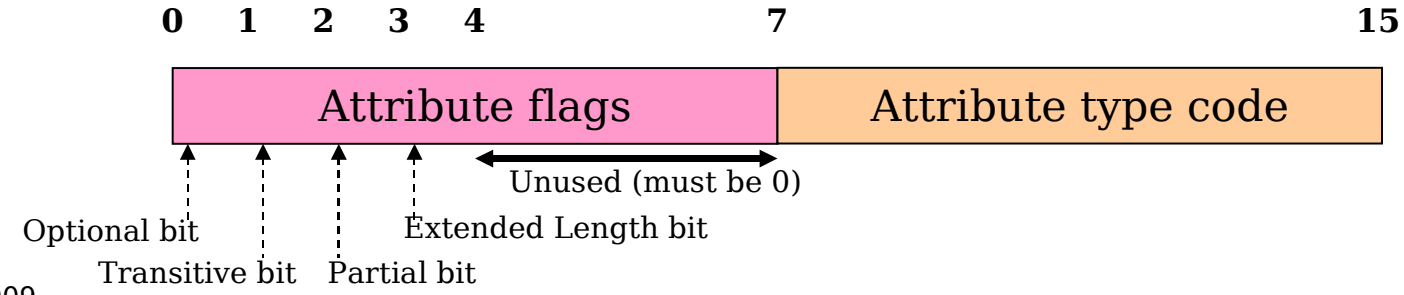
Path Attributes

- Total Path Attribute Length: total length of the Path Attribute field
  - '0' indicates that neither NLRI field nor the Path Attribute field is present in the UPDATE message
- Path Attributes:
  - A sequence of path attributes is presents in every UPDATE message except message that carries only withdrawn routes
  - Each part attribute is a triple of <attribute type, attribute length, attribute value>



Path Attribute type

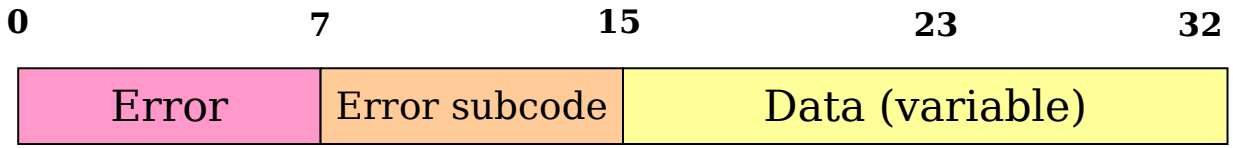
- Attribute type: consists of type code and flags
- Type code: contains the attribute code maintained by IANA
- Flags
  - Bit 0: well-known (0) or optional (1)
    - *Universally known?*
  - Bit 1: for optional attribute; non-transitive (0) or transitive (1)
    - *Pass the attribute to other neighbors?*
  - Bit 2: for optional transitive attribute; complete (0) or partial (1)
    - *Is attribute known by all on the path?*
  - Bit 3: for attribute length; one octet (0) or two octets (1)
  - Lower-order four bits: unused and always set to 0



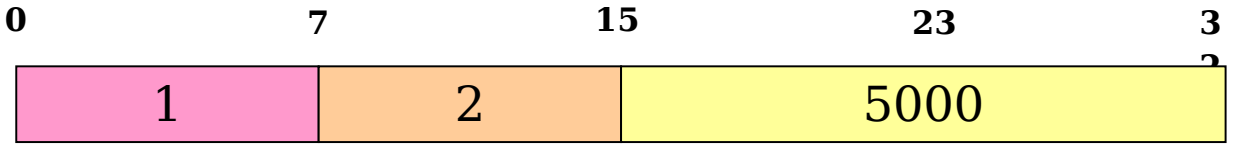
Network Layer Reachability Information (NLRI)

- Contains list of prefixes that are being advertised
  - Each prefix is encoded as a 2-tuple of <length, prefix>
- NLRI length = UPDATE message length – HDR length – Total length of Path Attributes field – Total length of Withdrawn Routes field
- NOTE: RFC 4760: Multiprotocol extensions for BGP-4 places a generalized NLRIs in an NLRI- *attribute*
  - This means that NLRI for non-IPv4 protocols is obsolete!

NOTIFICATION message



- Error code: indication the type of notification
- Error subcode: more specific information about the nature of the error
- Data: contains data relevant to the error e.g. Bad header, illegal ASN
- Data Length = Message Length - 21
- Example



  - Error: 1 Message Header Error
  - Error subcode: 2 Bad Message Length
  - Data: 5000 is the erroneous length

KEEPALIVE message

- Periodically sent to determine whether peers are reachable
- Sent at a rate that ensures that hold time will not expire
  - Recommended rate is one-third of the Hold Timer
  - Must not be sent more frequently than one per second
  - If Hold Timer is 0, periodic KEEPALIVE must not be sent

Path attributes categories

- Path attributes are characterize according to categories:
- Well-known: All BGP implementations must recognize them
  - Mandatory: Must always be present in all updates
  - Discretionary: May or may not be sent in an UPDATE
- Transitive: Must be passed on to next peer
- Optional: All BGP implementations need not recognize them
- Optional + Transitive has proven to be an excellent way to introduce new features seamlessly!
  - If a router does not recognize it, it just passes it along to its peers.
  - Therefore, almost all novel attributes are optional + transitive

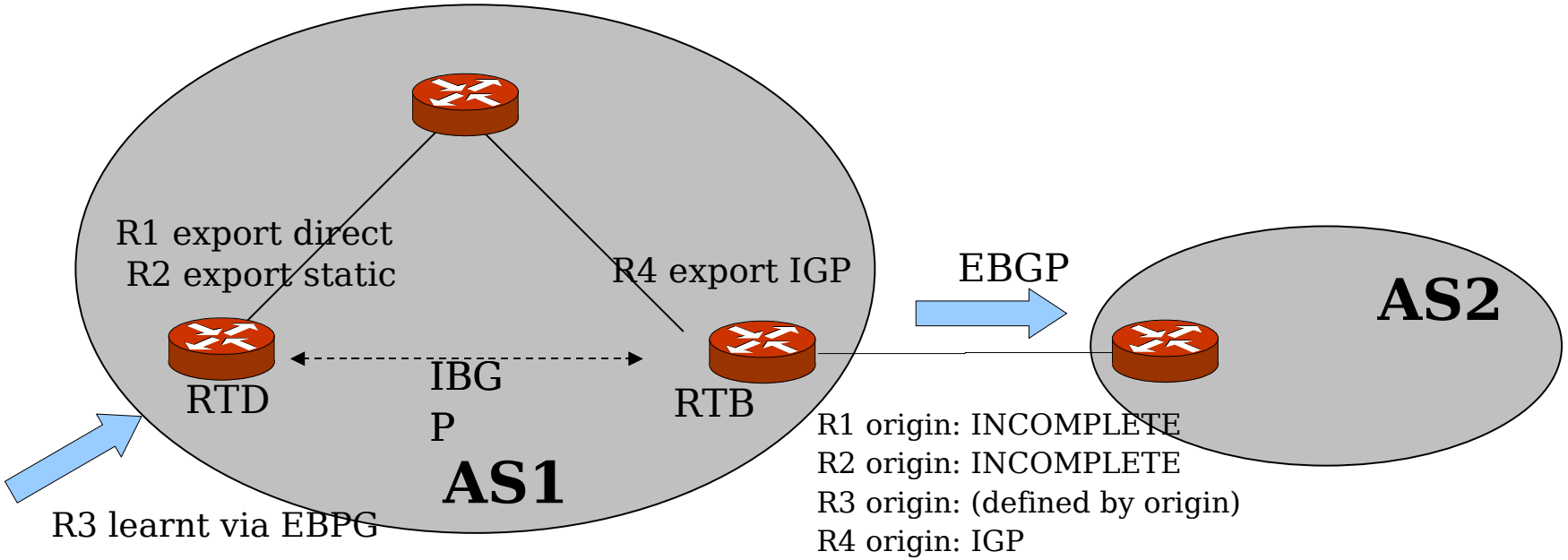
Early BGP path attributes

- Type code 1: ORIGIN (RFC4271)
- Type code 2: AS_PATH (RFC4271)
- Type code 3: NEXT_HOP (RFC4271)
- Type code 4: MULTI_EXIT_DISC (RFC4271)
- Type code 5: LOCAL_PREF (RFC4271)
- Type code 6: ATOMIC_AGGREGATE (RFC4271)
- Type code 7: AGGREGATOR (RFC4271)
- Type code 8: COMMUNITY (RFC1997)

ORIGIN

- Well-known mandatory
- Defines the origin of the path information
- Types:
  - IGP (0) — NLRI is internal to the originating AS (eg learnt via IGP)
  - INCOMPLETE (2) — NLRI is learned by some other means (eg static route)
- BGP prefers the path with the lowest origin type
- In JunOS, for example, ORIGINs are 0 by default

ORIGIN example



R1-R4 are routes imported into BGP in four different ways

- R1/R2: Direct / static
- R3: EBGP
- R4: IGP

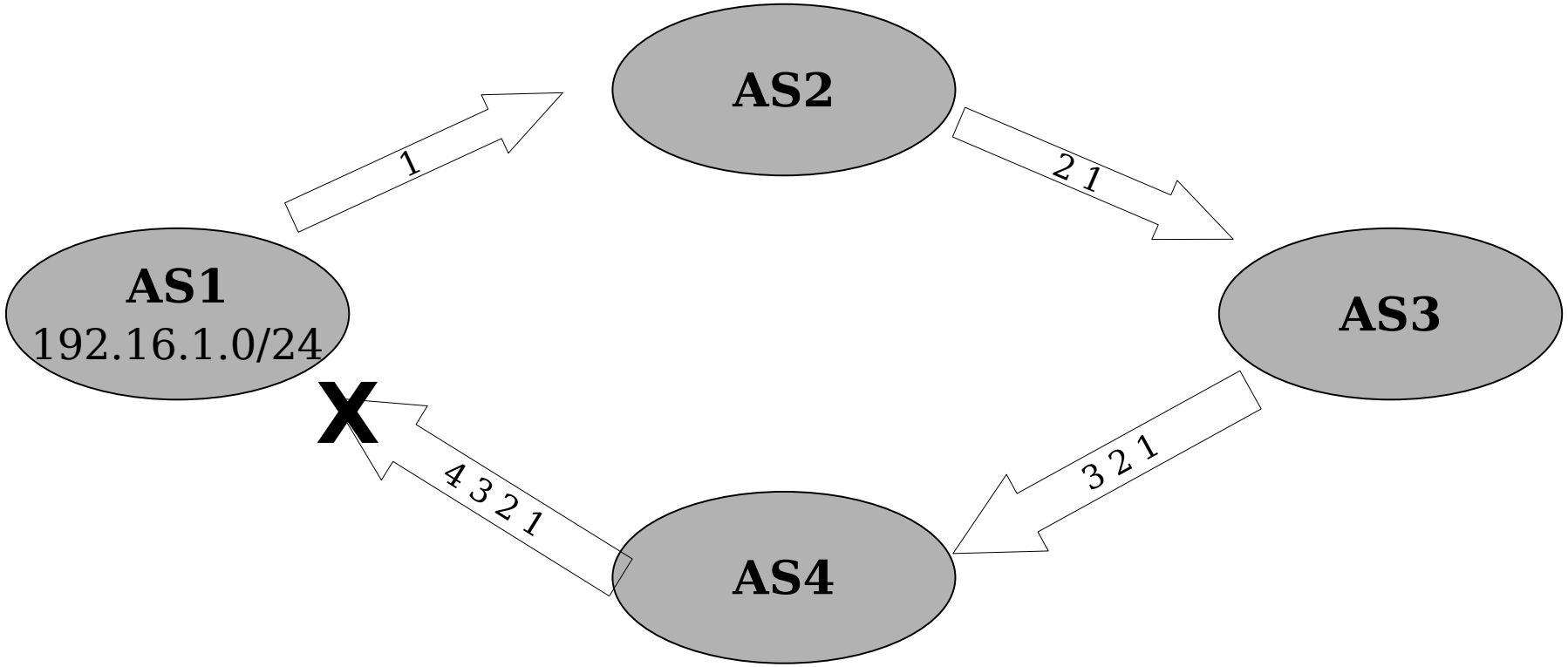
AS_PATH

- Well-known mandatory
- Contains a sequence of AS path segments

  <path segment type, path segment length, path segment value>

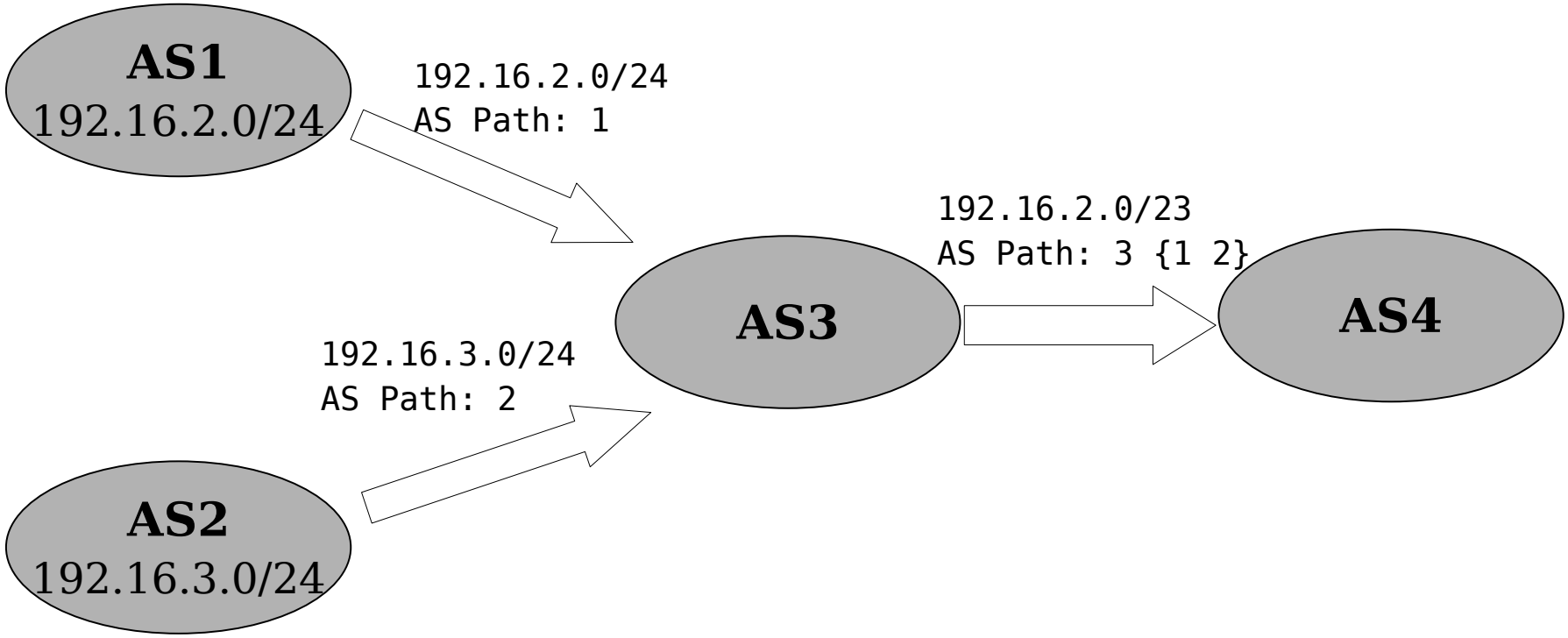
  - AS_SET (1): unordered set of ASes a route traversed
  - AS_SEQUENCE (2): ordered set of ASes a route traversed
- A BGP speaker prepends its ASN to the AS_PATH list when sending routes to external BGP peers (not to internal peers)
- Loop detection
- Shorter AS_PATH is preferred

AS_PATH (cont.)



- The AS-PATH is used to break loops (between AS:s)
- AS1 announces 192.16.1.0/24 to AS2 and detects its own ASN when received from AS4

AS SET

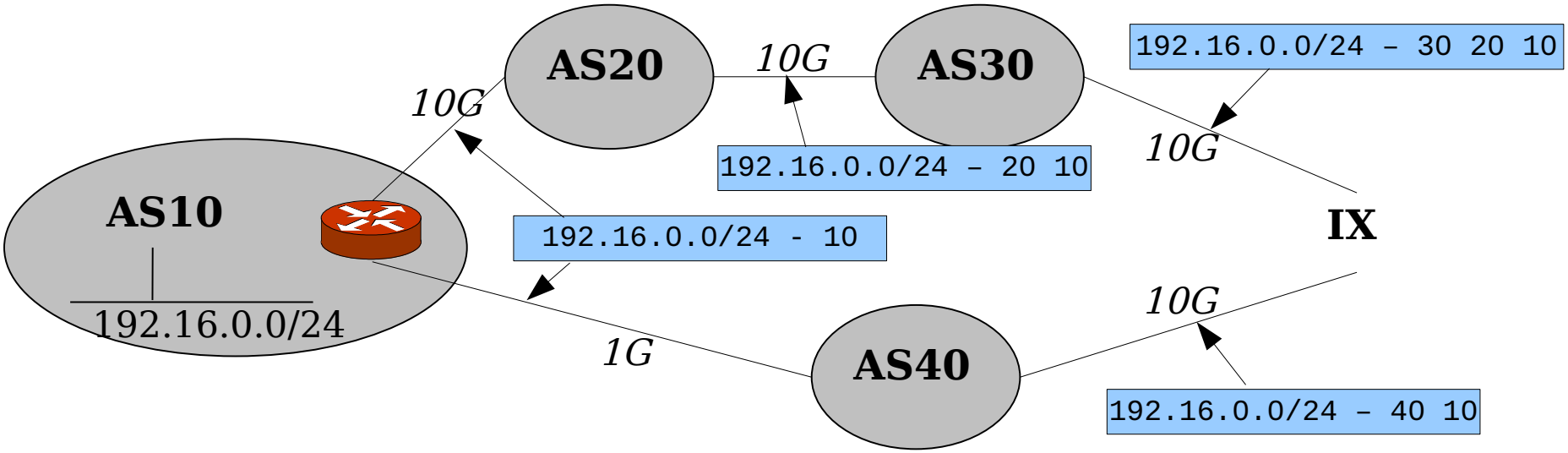


- As alternative to sequences of AS:s, a set denotes a set of AS:s.
- Useful in *aggregation* at AS-level
- Necessary to detect loops

AS_PATH Manipulation

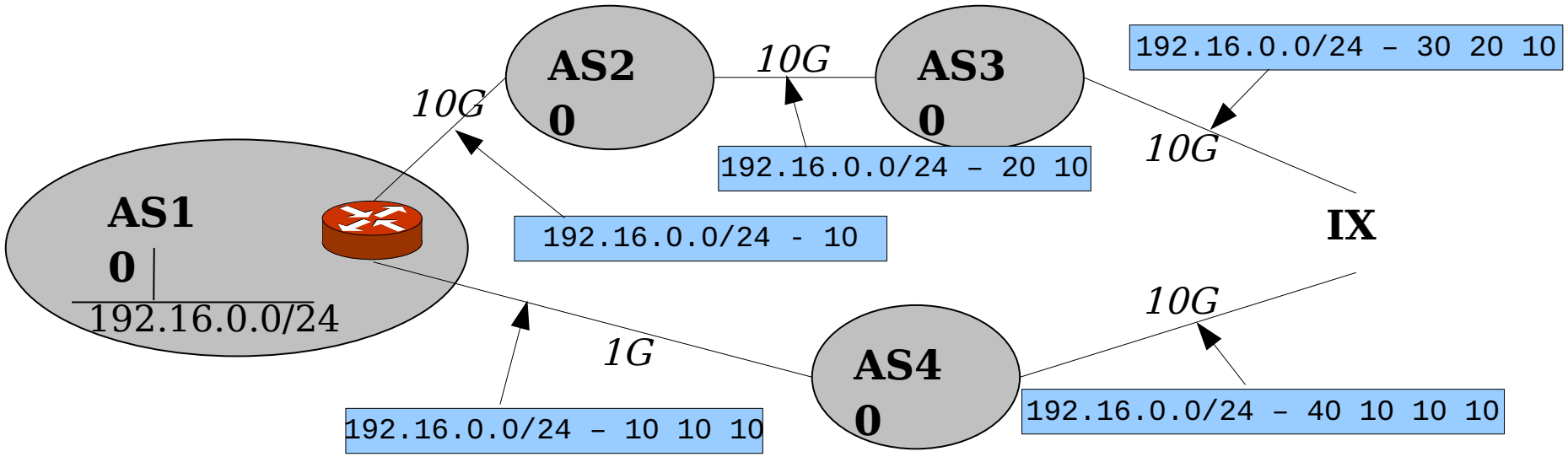
- The AS_PATH can be manipulated to affect inter-domain routing behavior
- The AS_PATH can be *lengthened* to make a path less preferable
- This affects all ASes that receives this prefix update
- Unlike the MED that only can affect how a neighboring AS sends traffic to you
- Affects how incoming traffic is routed
- Is achieved by *prepending* dummy ASNs to the AS_PATH

Routing case *before* Manipulation



- AS10 has two links to the rest of the world
  - 10Gbit/s and 1Gbit/s
- In this case, traffic from the Internet Exchange IX will follow shortest AS_PATH
  - Traffic will thus use the incoming 1G link
- How can AS10 steer traffic from the IX to take the 10G link instead?

Routing case *after* manipulation

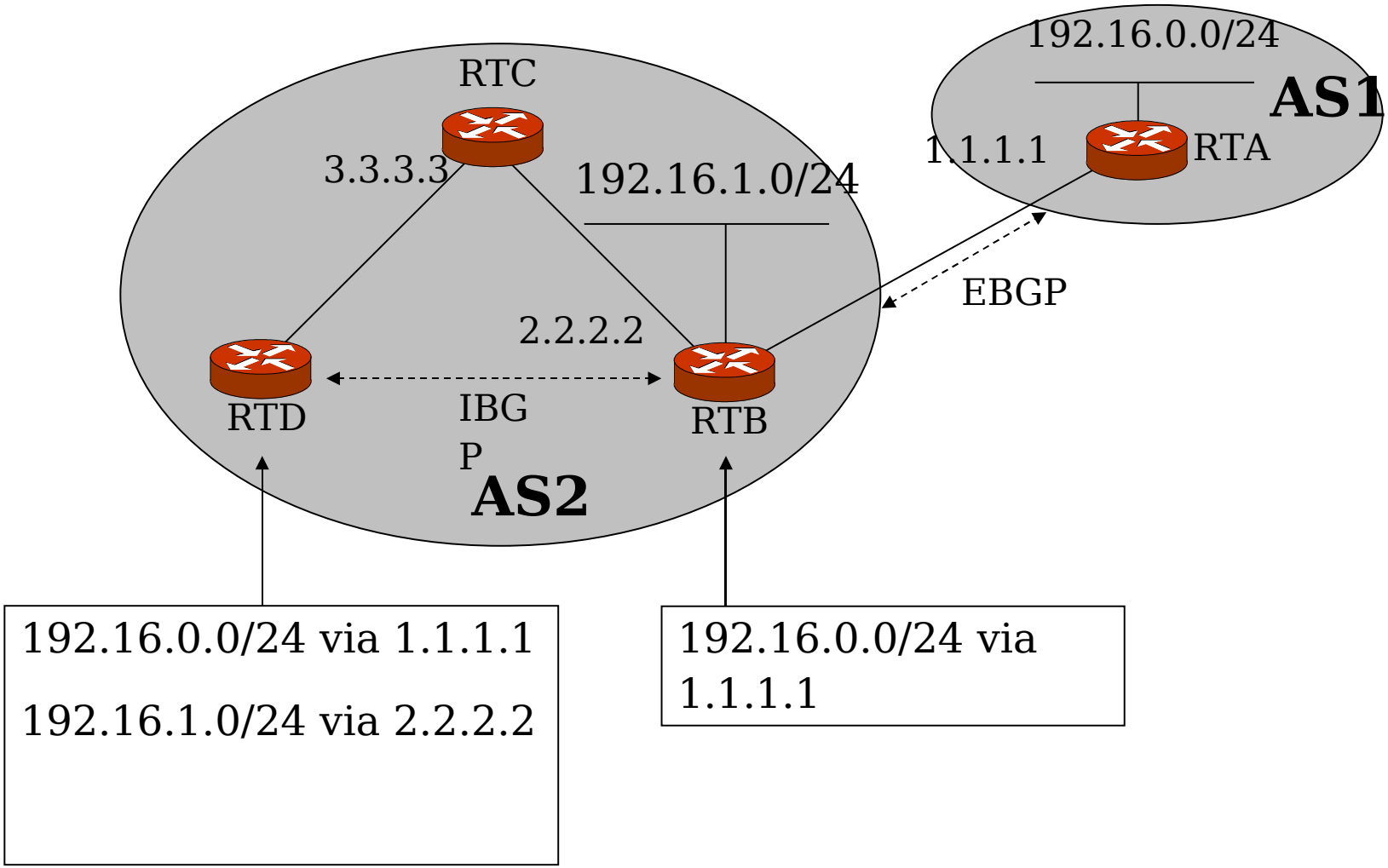


- To force incoming traffic to go through the 10G link, AS10 manipulates the AS_PATH
  - Insert dummy AS numbers when sending UPDATEs to AS40
  - Make the AS_PATH over the 10G link become the shorter one
- Best practice: bogus AS number should be duplicate of own
  - Otherwise, the number can be misleading and cause routing loops

NEXT_HOP

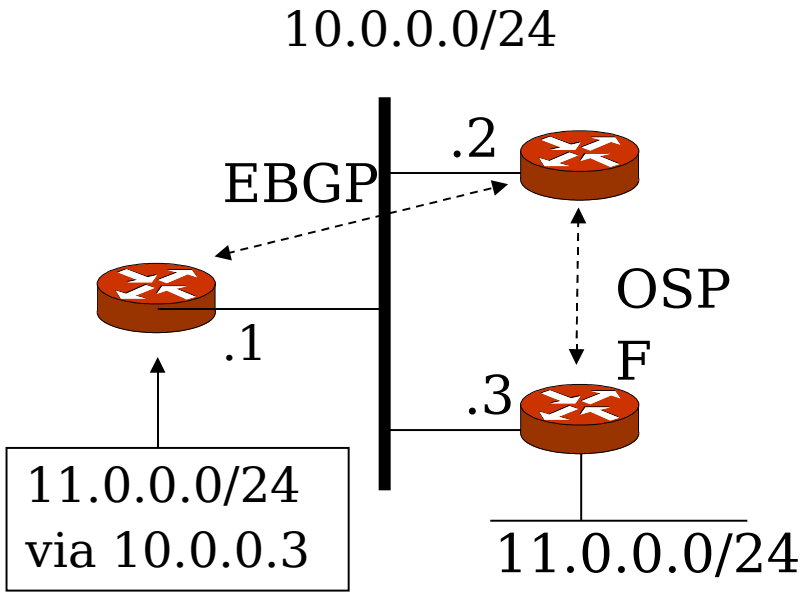
- Well-known mandatory
- Defined IP address of the router that should be used as the next hop to the destinations listed in NLRI
- Next-hop concept for BGP
  - External peer: IP address of the peer that announced the route
  - Internal peer:
    - Locally originated routes: IP address of the peer that announced the route
    - Routes learned from external: IP address of the external peer from which the route was learned
  - Route on multiaccess medium: IP address of interface of the router connected to the medium that originated the route

NEXT_HOP (cont.)



NEXT_HOP on Multiaccess Media

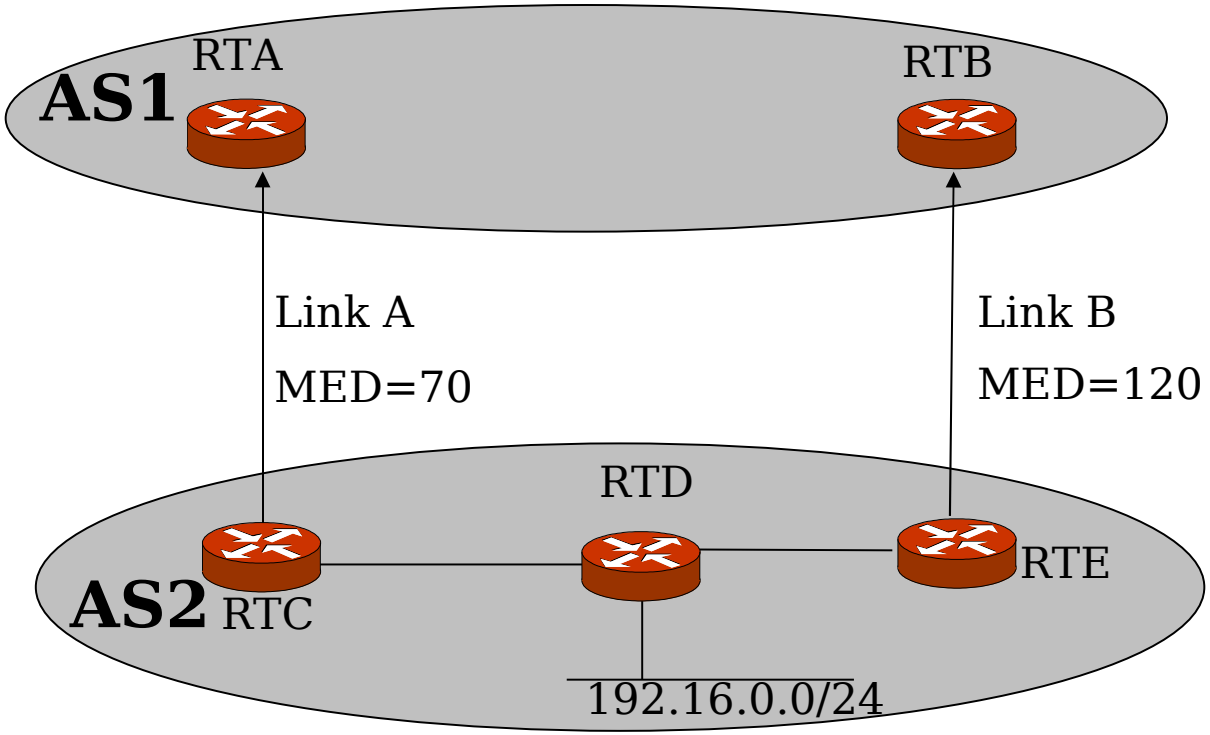
- When advertising route on a multi-access media , the next hop can be an IP address of the interface of the router connected to the medium that originated the route
  - This is called third-party next-hop



MULTI-EXIT-DISCRIMINATOR (MED)

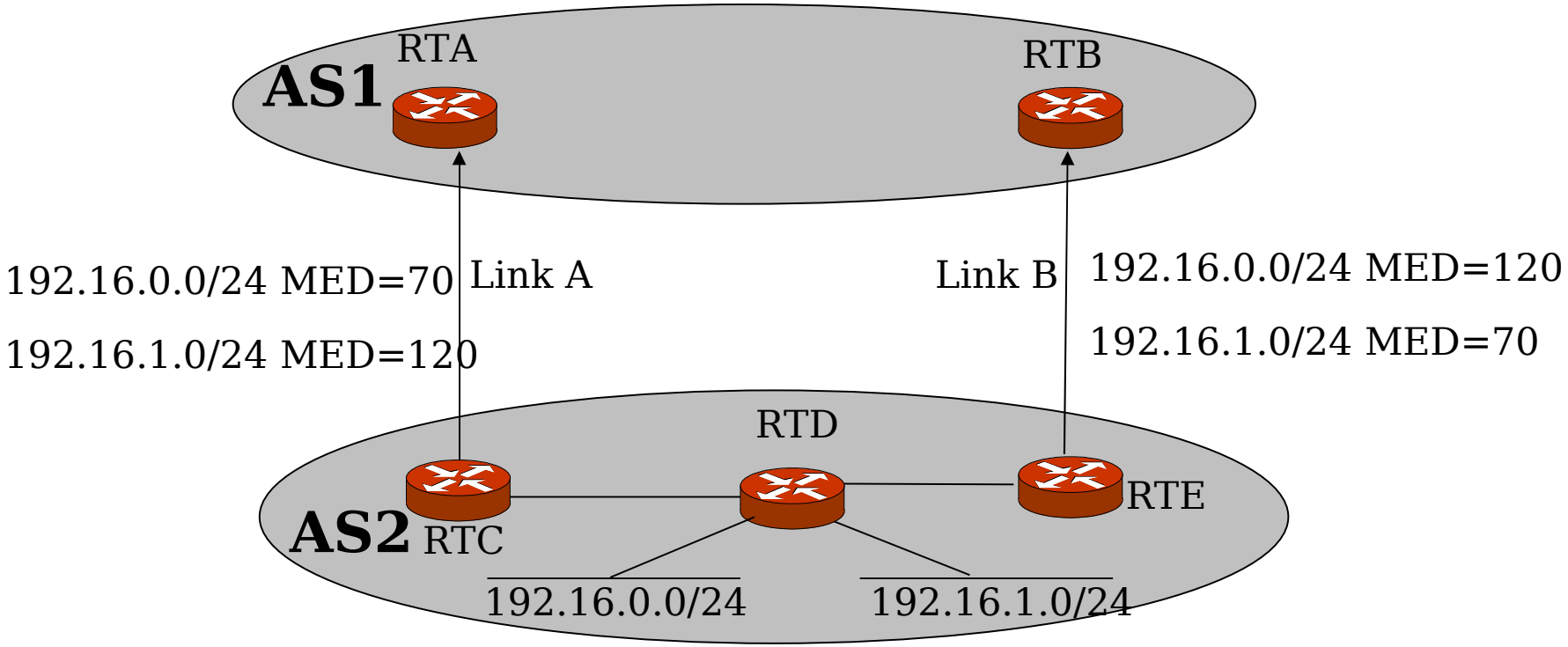
- Optional non-transitive
- Used on external links to discriminate among multiple links to the same neighboring AS
- Lower MED is preferred
- MED received from external peer must not be propagated to other neighboring AS:s

MULTI-EXIT-DISCRIMINATOR (cont.)



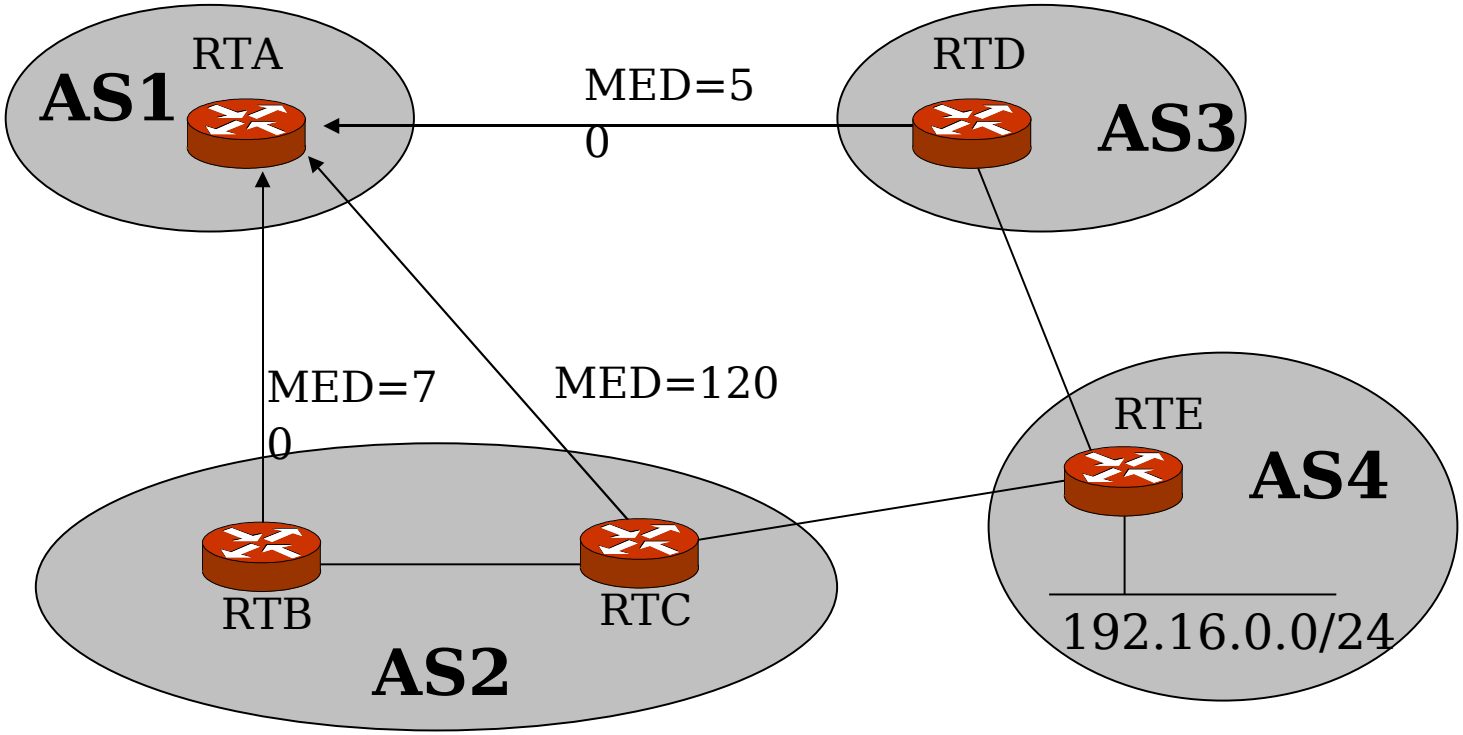
- AS2 prefers to receive traffic to 192.16.0.0/24 on link A and use link B as backup
- RTC announces the prefix with low MED, RTE announces the prefix with high MED.

MULTI-EXIT-DISCRIMINATOR (cont.)



- AS2 wishes to load-balance traffic using MED.
- Receive traffic to 192.16.0.0/24 on Link A
- Receive traffic to 192.16.1.0/24 on Link B

MULTI-EXIT-DISCRIMINATOR (cont.)



- AS1 will select RTB over RTC, but chooses between RTB and RTD using other means
- MEDs can not be compared from different ASs!

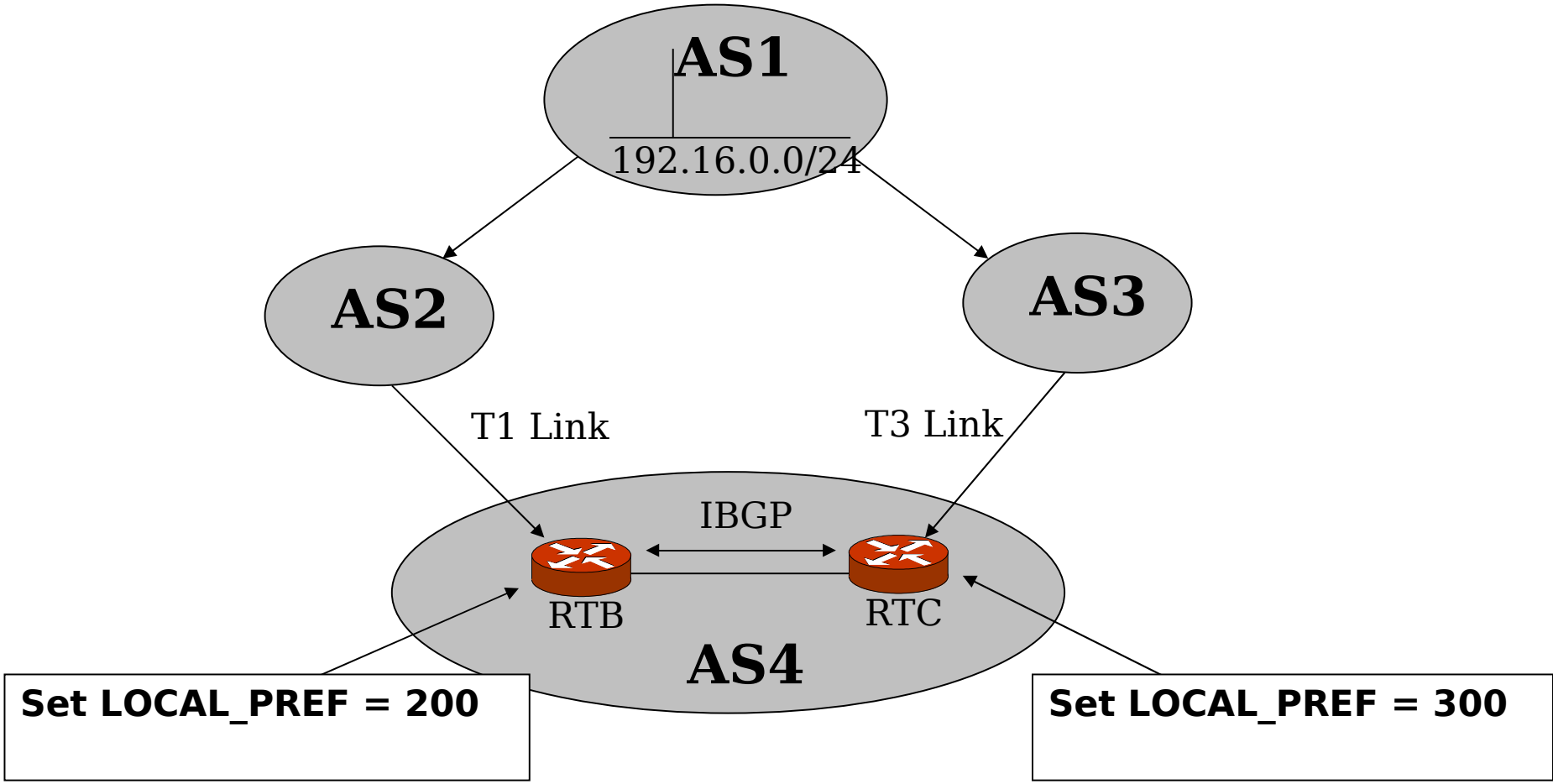
Using MED as tie-breaker

- The use of MED as tie-breaker is controlled by several sub-settings
- CISCO for example, has a non-deterministic comparison by default based on age of the routes (newer routes are pairwis compared).
  - `cisco-non-deterministic` parameter in JunOS.
  - `deterministic-med` in CISCO
- You can also set `always-use-med` to use MED comparisons from different AS:s
  - Can be useful if you are among a group of AS:s that trust each other.

LOCAL-PREF

- Well-known discretionary
- Used as local policy to set degree of preference of routes when announcing to other internal peers
- Used locally within the AS
- A *higher* local preference is preferred(!)

LOCAL_PREF (cont.)



- AS4 prefers to send traffic to 192.16.0.0/24 on the T3 Link.

MED versus LOCAL_PREF

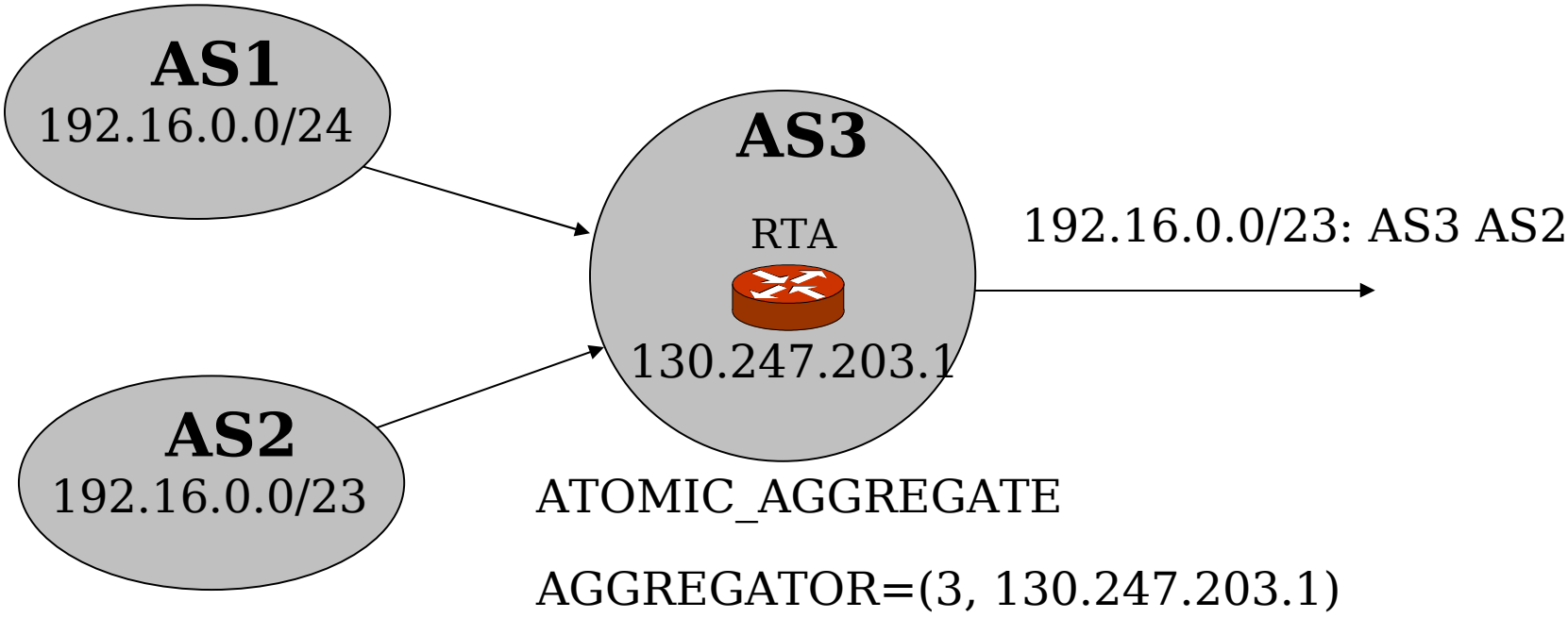
- MED is announced to other AS:s
  - Used by your neighbors to tell you how they want to receive traffic
- LOCAL_PREF announced internally
  - Used by you to steer traffic internally (to your neighbors)
- LOCAL_PREF overrides MED
- Lower MED preferred
- Higher LOCAL_PREF preferred

ATOMIC_AGGREGATE

- Well-known discretionary
- Set to indicate information loss
  - There may be longer prefixes to AS:s not in AS_PATH
  - Alternative to using AS_SET
- Receiving BGP speaker must not de-aggregate the route
- BGP speaker that receives a route with this attribute needs to be aware of that actual path to destinations may not be the path specified in the AS_PATH attribute of the route
- Should not be set when the aggregate carries some extra information that indication from where the aggregated information came

AGGREGATOR

- Used in combination with ATOMIC_AGGREGATE
- Optional transitive
- Contains ASN and IP address of BGP speaker that aggregates the route



Example: ATOMIC_AGGREGATE and AGGREGATOR

BGP Decision Process

1. If next hop inaccessible, ignore route
2. Prefer highest local preference value
3. Prefer shortest AS_PATH
4. Prefer lowest origin type (IGP, EGP, INCOMPLETE)
5. Prefer lowest MED value (if from same AS)
6. Prefer routes from EBGP over IBGP
7. Prefer routes with lowest IGP metric Nexthop
8. Prefer route from peer with lowest router id
9. Prefer route from peer with lowest address

Vendor specific tie-break

- CISCO has several own rules
  - Weight (cisco-specific)
  - Routes are compared by default pairwise in the order they arrived (non-deterministic).

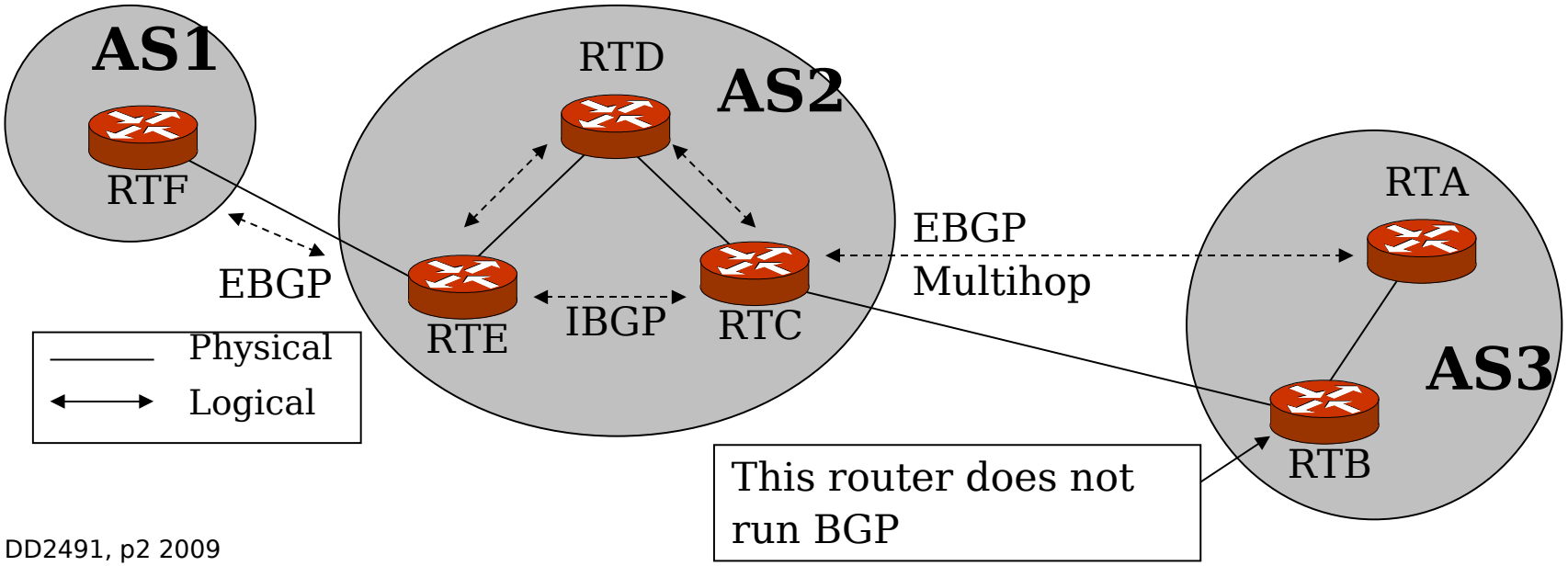
    Example:

    entry1: AS(PATH) 500, med 150, external, routerid 172.16.13.1
    entry2: AS(PATH) 100, med 200, external, routerid 1.1.1.1
    entry3: AS(PATH) 500, med 100, internal, routerid 172.16.8.4

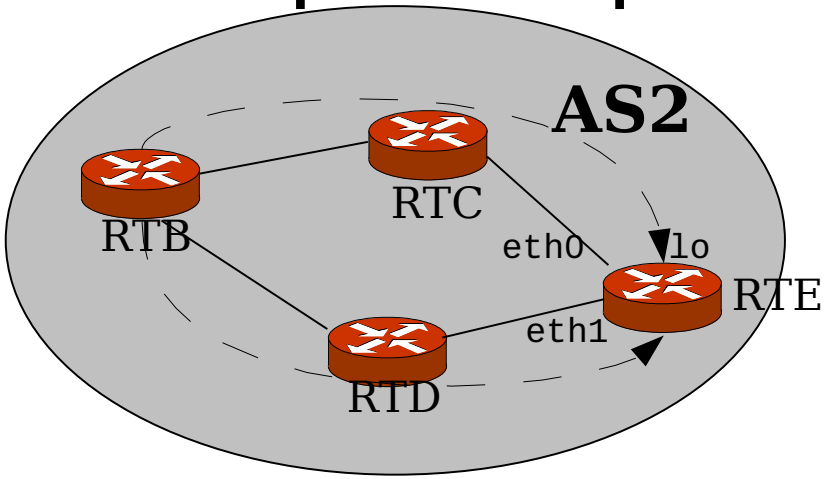
  - Entry3 is chosen. Why?
  - Turn this off with `bgp deterministic-med`
- Juniper is somewhat more standard compliant

Peering sessions

- Neighbor negotiation of IBGP and EBGP are the same
  - IBGP peering: within an AS
  - EBGP peering: between AS:s
- Two peers must have IP connectivity
  - Simple check: they should be able to ping each other
- If EBGP peers are not physically connected: Multihop EBGP

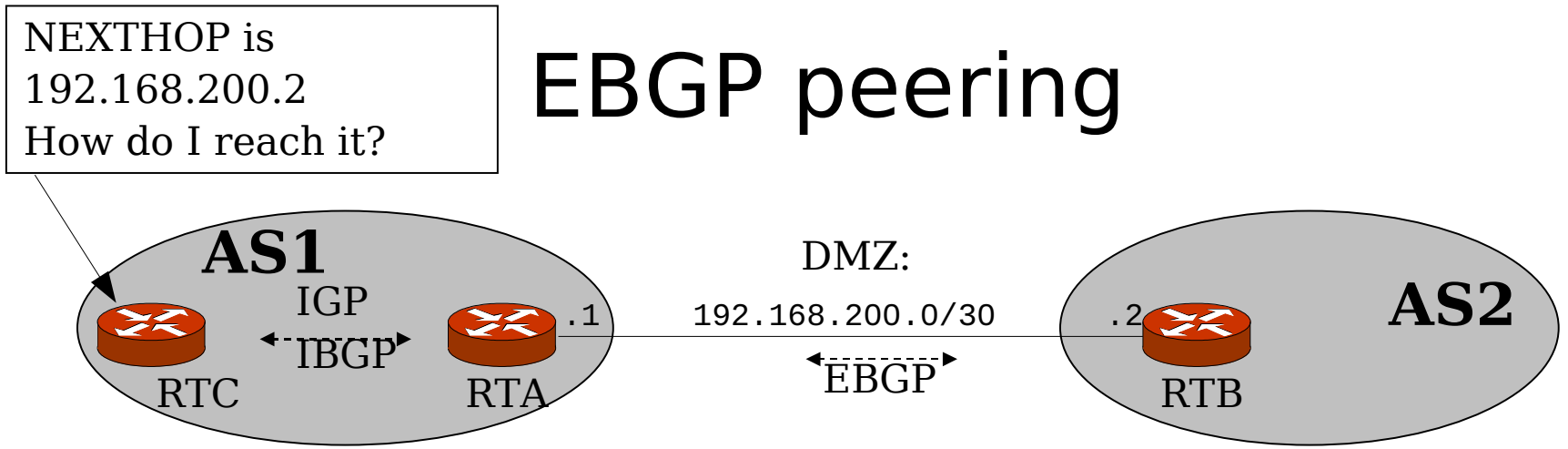


IBGP loopback peering



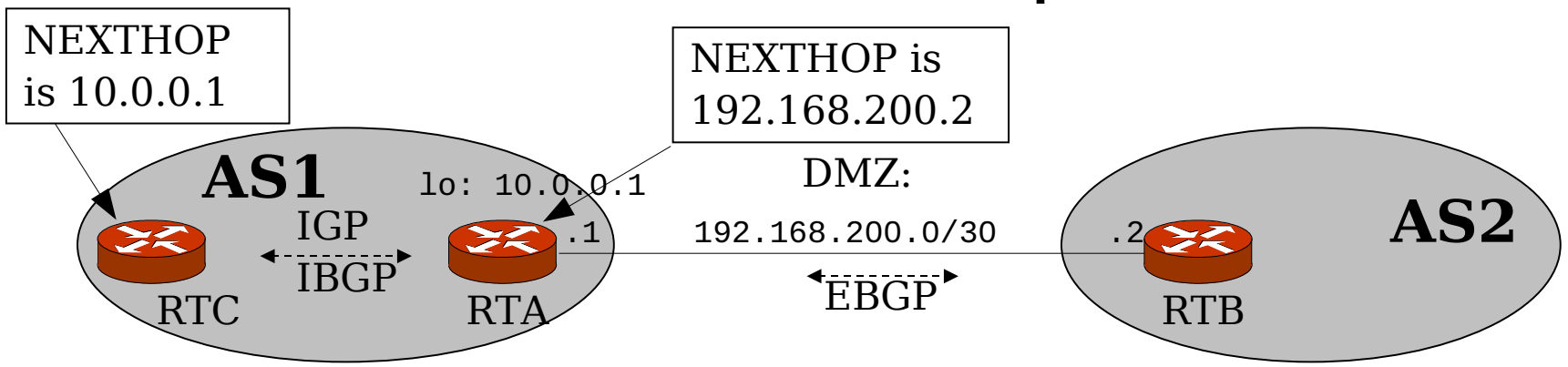
- IBGP peering is usually done with loopbacks. Why?
  - More stable: Not tied to single physical path, if a link/interface goes down, another route may be chosen.
- But IBGP needs an IGP so that the loopbacks can be reached
  - And TCP connections can be established

EBGP peering



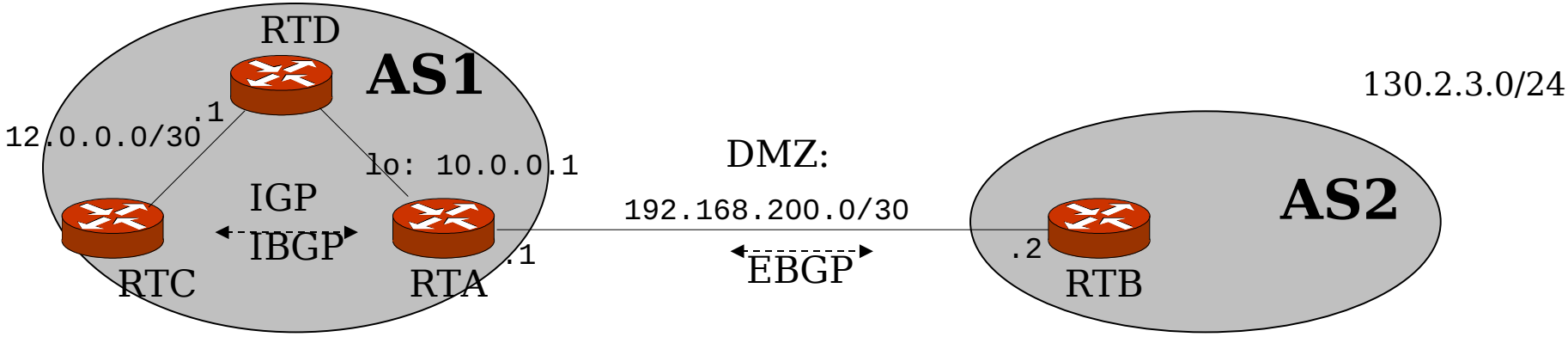
- EBGP peering is typically made over a physical directly connected link
  - Demilitarized zone (DMZ) that does not “belong” to either AS
- But if routes learned via EBGP uses the *DMZ address as nexthop*
  - The DMZ must be redistributed via IGP!
  - But the DMZ is not really part of the AS,...
    - Although the DMZ is usually a part of one of the AS:s address ranges

EBGP: Next-hop self



- Alternative:
  - Set next-hop-self
  - Announce routes using the loopback address of the border router as next-hop
  - DMZ does not need to be distributed within the AS
- But RTA still uses the directly connected DMZ address

EBGP nexthop: recursive lookup



RTC:s routing table alternatives

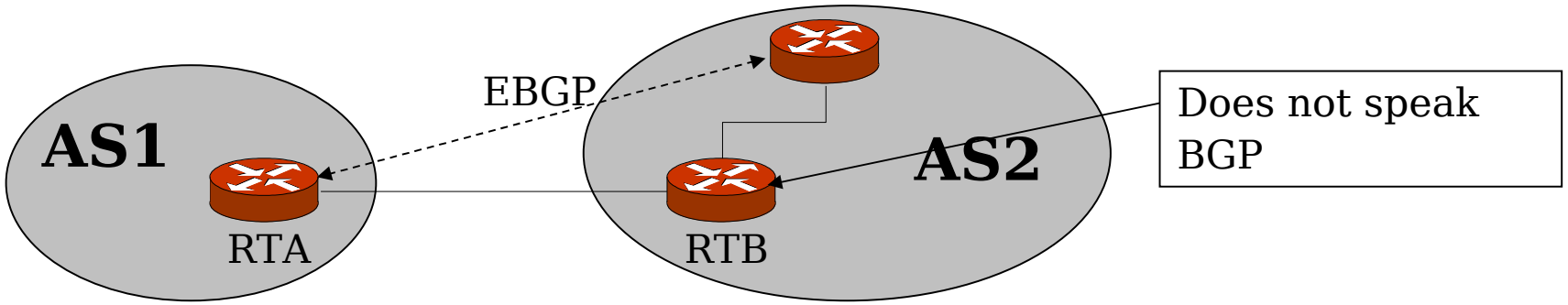
Route	*Nexthop*	*Protocol*
130.2.3.0/24	192.168.200.2	IBGP
192.168.200.0/30	12.0.0.1	IGP
12.0.0.0/30	-	direct

DMZ nexthop

Route	*Nexthop*	*Protocol*
130.2.3.0/24	10.0.0.1	IBGP
10.0.0.1/32	12.0.0.1	IGP
12.0.0.0/30	-	direct

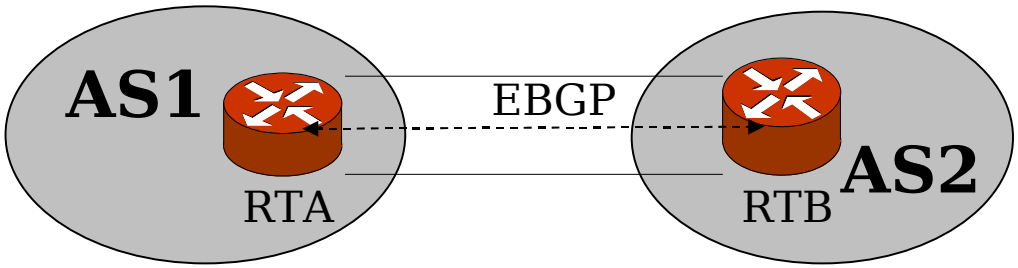
Next-hop self

Alternative EBGP peering: multi-hop



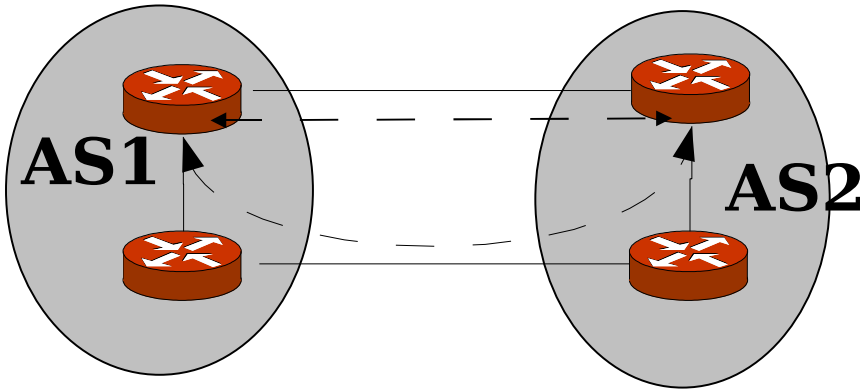
- Problems: EBGP peering dependent on routing
- Nexthop address is in other AS
  - You need to learn the remote next-hop address
  - And you may be dependent on the other AS:s IGP for delivery that may cause instabilities

Alternative EBGP peering: redundancy/load balancing



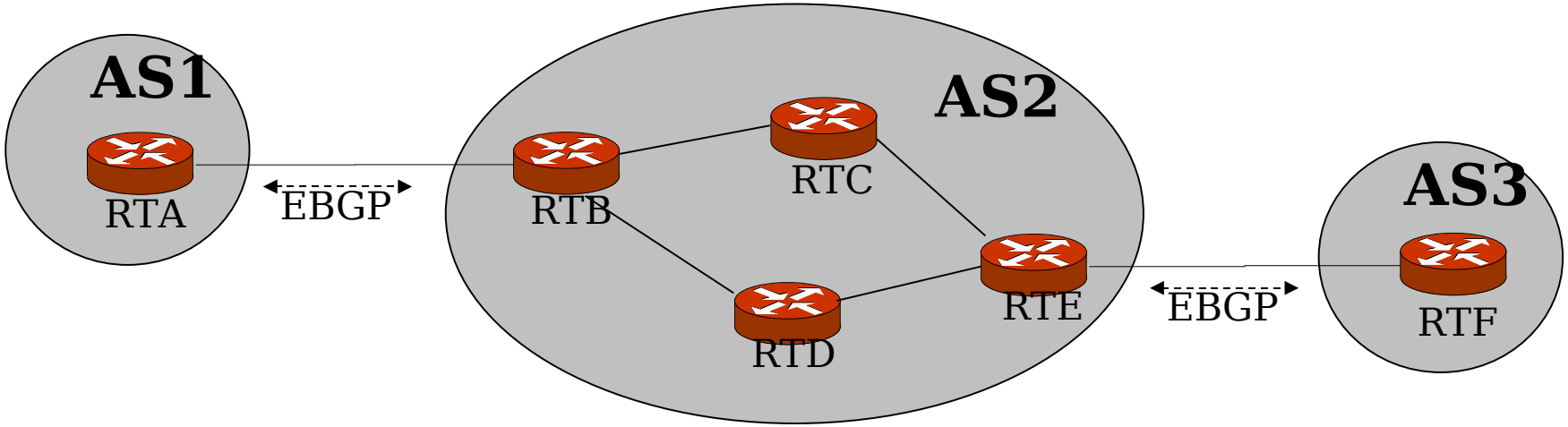
- Peer between loopback address
- Set equal cost to nexthop using IGP or static routes
- Load balance between links
- Or use one link as redundant link

Alternative EBGP peering: redundancy/load balancing



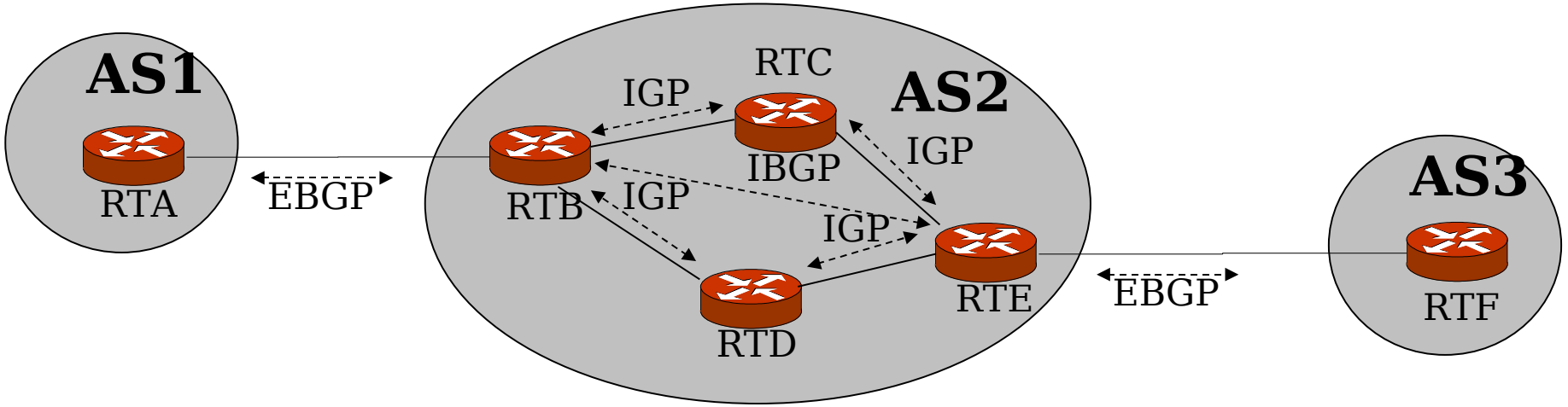
- Peer between loopback address
- Load balancing between several links/redundant link
- But next-hop may now be dependent on IGP/IBGP in other AS

How to transit traffic?



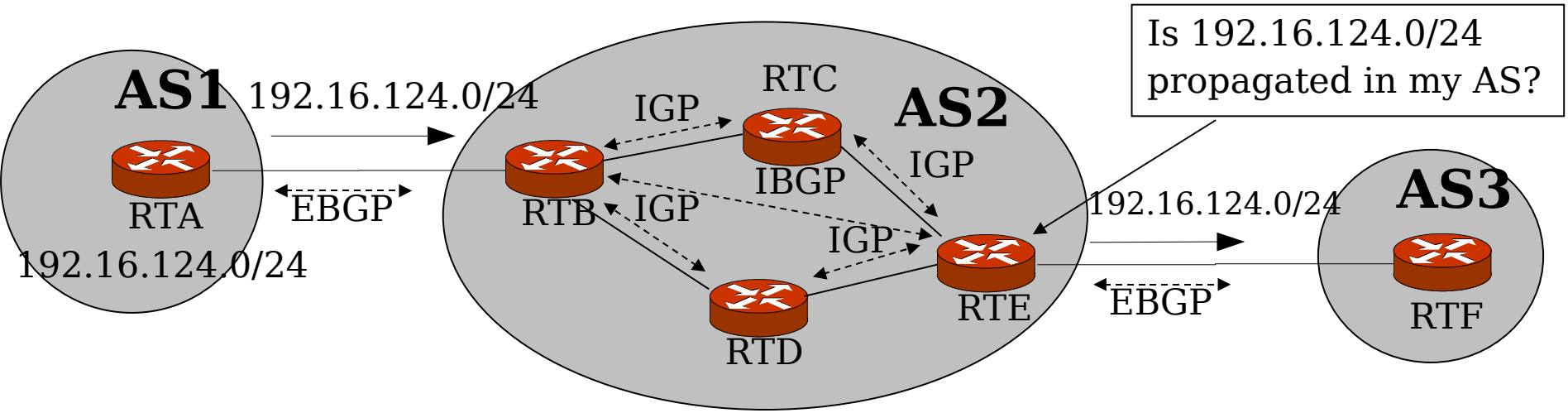
- How does RTC and RTD know how to forward transit traffic between RTA and RTF?
- You cannot use default routes. Why?
- You may use IGP to distribute external routes.
- But most common nowadays is to use IBGP to distribute external routes internally.

How to transit traffic using IGP



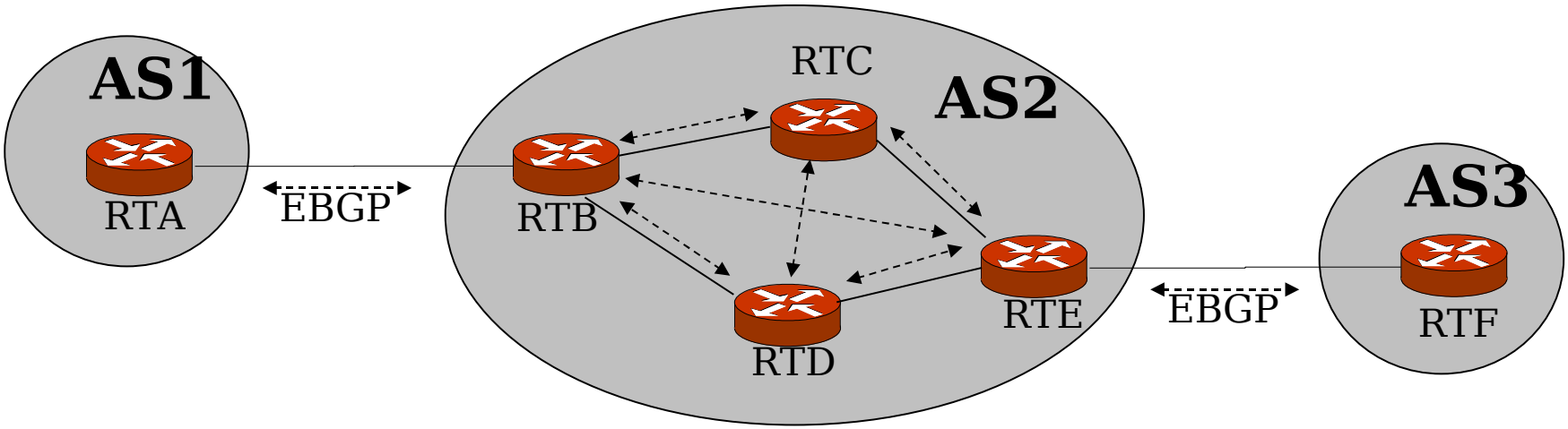
- You can inject all BGP routes into your IGP as external routes
- Scales badly – High memory consumption for the IGP and will take time to converge
- There is also a problem with synchronization between the IGP and EBGP
  - Can you announce a route even though your IGP has not converged?

Synchronization between IGP and BGP



- If an AS provides transit service to another AS, a BGP speaker should not advertise a route to external peer unless all routers within an AS learned about that route via IGP
  - RTE checks that 192.16.124.0/24 is reachable via IGP before announcing it to RTF via BGP
- If you ignore this: no synchronization

Using IBGP: Full mesh



- With IBGP all transit routers know all external routes.
  - Note that all transit routers need to speak IBGP (what happens if they dont?)
- But loop prevention in BGP is via AS_PATH
  - There is no change in AS_PATH between internal peers!
- Loop prevention in IBGP:
  - All IBGP speakers are *fully meshed*
  - *Never reannounce routes to an IBGP peer learned from another IBGP peer*

IBGP full mesh

- So IBGP needs to be fully meshed in order for:
  - All internal routes to receive all external routes
  - Loop prevention (no difference in AS_PATH)
- The number of TCP connections in an AS:
  - n*(n-1)/2
- Not practical for large transit networks, but new routers are pushing the limit upwards due to higher route-processor performance
- Two ways to remove this scaling limitation (later lecture):
  - Route reflectors (RFC 4456)
  - AS confederations (RFC 5065)

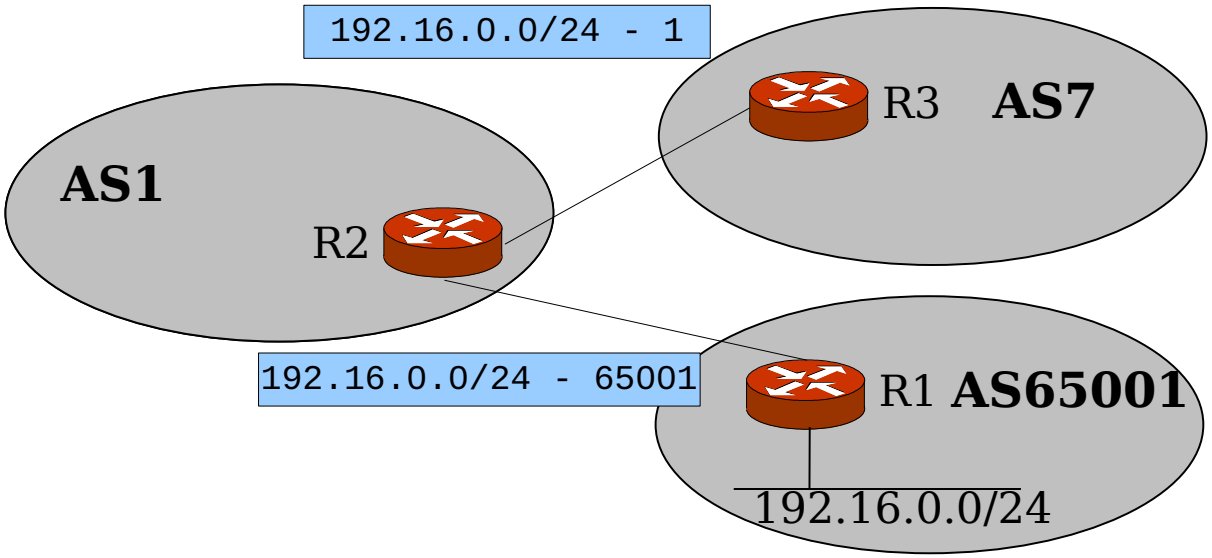
Route advertisement rules

- BGP next-hop must be reachable
  - Consequence: If IGP fails, BGP route is not announced
- Advertise only active BGP routes to peers
  - Juniper specific
  - Consequence: If same route from IGP is active, BGP route is not announced!
  - Turn off in JunOS with: `advertise inactive`
- Never forward IBGP routes to IBGP peers (full mesh)

Private ASes

- Sometimes it is not necessary to have a public ASN
- IANA has reserved the ASN range 64512 – 65535 for internal use within a system
- Can be used for customers that are single-homed or multi-homed to the same provider
- Private ASNs must not be announced globally
- Providers must strip private ASNs before announcing the prefixes on to the rest of the Internet
- Purpose of private AS:s is to conserve AS numbers and hide networks

Private ASes, cont'd



- Prefixes originating from AS65001: AS_PATH of 65001
- AS1 propagates the prefix but strips the private ASN

Extensions

- BGP is under constant development
- New operational problems and new technologies require extensions to the protocol
- Extensions are introduced, standardized, and implemented
- Implementation of extensions:
  - Negotiated via BGP capabilities when peering is set up.
  - Sent as optional transitive attributes and either recognized or not

Extensions example

- BGP extensions
  - BGP communities attribute (RFC1997)
  - Route refresh capability (RFC2918)
  - BGP multipath (RFC3107)
  - Capabilities advertisement (RFC3392)
  - BGP route reflection (RFC4456)
  - Multi-protocol extensions (RFC4760)
  - Graceful restart (RFC4724)
  - Four-byte AS (RFC4893)
  - Autonomous system confederations (RFC5065)
- TCP extension
  - TCP MD5 signature option (RFC2385)

Capabilities Advertisement

- An 'extension to negotiate extensions'
- Announce supported capabilities to the peer with OPEN message using options parameter
- Some capabilities are (~same as previous slide)

Value	Description	Reference
0	Reserved	RFC3392
1	Multiprotocol Extensions for BGP-4	RFC4760
2	Route Refresh Capability for BGP-4	RFC2918
3	Cooperative Route Filtering Capability	
4	Multiple routes to a destination capability	RFC3107
5-63	Unassigned	
64	Graceful Restart Capability	RFC4724
65	Support for 4-octet AS number capability	RFC4893
66	Deprecated (2003-03-06)	
67	Support for Dynamic Capability (capability specific)	
68-127	Unassigned	
128-255	Vendor Specific	

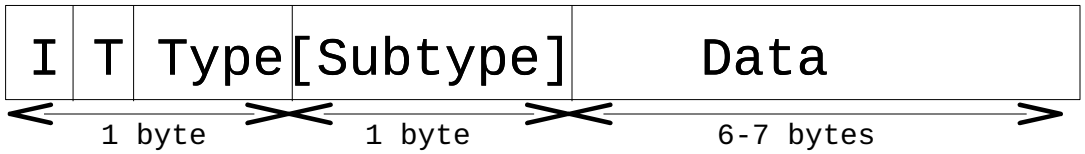
The COMMUNITY attribute

- RFC 1997 defines a 4-byte COMMUNITY attribute as optional transitive
- A group of destinations that share some common property
- Used to simplify routing policies based on *logical property* rather than IP prefix or ASN
- Format
  - First 2-bytes ASN, last 2-bytes defines a value (*ASN*:*value*)
  - Example 5678:90 (0x162E005A)
- A route can have more than one community attribute
- BGP speaker can add and modify a community attribute before passing routes on to other peers

NO_EXPORT example

- NO_EXPORT is a well-known community (0xFFFF FF01)
  - a route should not be advertised to peers outside an AS
- Other well-known:
  - NO_ADVERTISE: a route should not be advertised to other BGP peers

Extended communities



- Extended communites are 8-bytes with more structure
  - RFC 4360
- Type 0: <ASN:2; Data:4>
- Type 1: <IPv4:4; Data:2>
- Type 2: <ASN:4; Data:2>
- Example: Used in VPNs to tag VPN-specific information

Use of Communities

- Communities are used extensively in modern networks for defining policies
  - Both internally and between networks (if they have agreed)
- You “tag” a route with a community and use this information to implement a policy
  - Tag backbone routes
  - Tag routes you wish to advertise to peers
  - Tag routes with VPN label
  - Eg: SUNET uses communities to tag routes with academic and non-academic sites
- See Practical BGP p 217 ff, for more examples

Community configuration example (1)

Tagging a community at the edge (or by the other peer):

```
community academic members 3244:11;
policy-statement add-academic {
        route-filter 172.16.0.0/8 upto /16 {
            community add academic;
            next policy;
        }
        route-filter 192.168.0.0/8 upto /24 {
            community add academic;
            next policy;
        }

    }
}
```

Community configuration example (2)

Using the community to implement a policy:

```
community academic members [3244:11]; # regexp
policy-statement from-academic {
    from {
        protocol bgp;
        community academic;
    }
    then
        as-path-prepend “201;201”;
}
```

Multiprotocol extension for BGP-4

- Support routing of other network layer protocols than IPv4
- NLRI and NEXTHOP fields in the UPDATE message are IPv4 specific
- Use a generalized address form using:
    - AFI - Address Family Identifier
    - SAFI - Subsequent Address Family Identifier
- Two new attributes replace NLRI and NEXTHOP
  - Multiprotocol Reachable NLRI (MP_REACH_NLRI)
    - Contains NLRI and nexthop
  - Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI)
- Used in multicast, IPv6, VPNs, etc.

Multiprotocol extension for BGP-4

- Examples
  - IPv4 unicast: AFI=1, SAFI=1
  - IPv4 multicast: AFI=1, SAFI=2
  - L3VPN: AFI=1, SAFI=128
  - IPv6 unicast: AFI=2, SAFI=128
  - IPX: AFI=11

Route refresh capability

- By default, BGP has no mechanism to dynamically request for re-advertisement of routes from a peer
- If a route (its attributes) does not change, a BGP speaker does not re-announce it
- Therefore, a receiver needs to cache all previous routes
  - Even if not required at a specific moment
- This places a lot of load (memory) on the receiver
- Suppose a router changes input policy and does not want to store all data from all neighbors “just in case”
- With the ROUTE-REFRESH message, a router can request to get the complete Adj-RIB-Out from a neighboring router

TCP MD5 Signature option

- Provides a mechanism for TCP to carry a digest message in each TCP segment using a shared secret
  - MD5 message digest algorithm
  - Verification of authenticy (no encryption)
  - manually configured
- Protects tcp header, parts of the ip header (addresses) and the whole TCP payload (ie whole BGP message)
- Helps BGP protect itself from spoofed TCP segments, TCP SYN/ RST, data injection,
- Problem
  - MD5 algorithm has been found to be vulnerable to collision search attack
  - Performance issue from calculating and comparing digests
  - Few actually use it (less than 10% - G Huston 2009)

Practical BGP: pages 333-336

Graceful restart

- When a BGP peering goes down (eg TCP connection is reset, closed, or Keepalive fires) a BGP peer stops forwarding to that peer
  - Ie, if the *control* plane fails, it is seen as a sign that the *data*-plane fails
- The other peers switches over the routes to other peers
- Then when the router comes back up (BGP peering established) they may switch over to the restarted router again
- But most hw routers have separate forwarding and control planes
- So forwarding can (in principle) continue without the control plane
  - At least for a limited period of time before forwarding data becomes stale
- Graceful restart:
  - Tell your peers that: You are going down but you will be back, please continue forwarding to me until I am back.

Practical BGP: pages 269-276

Route flap damping

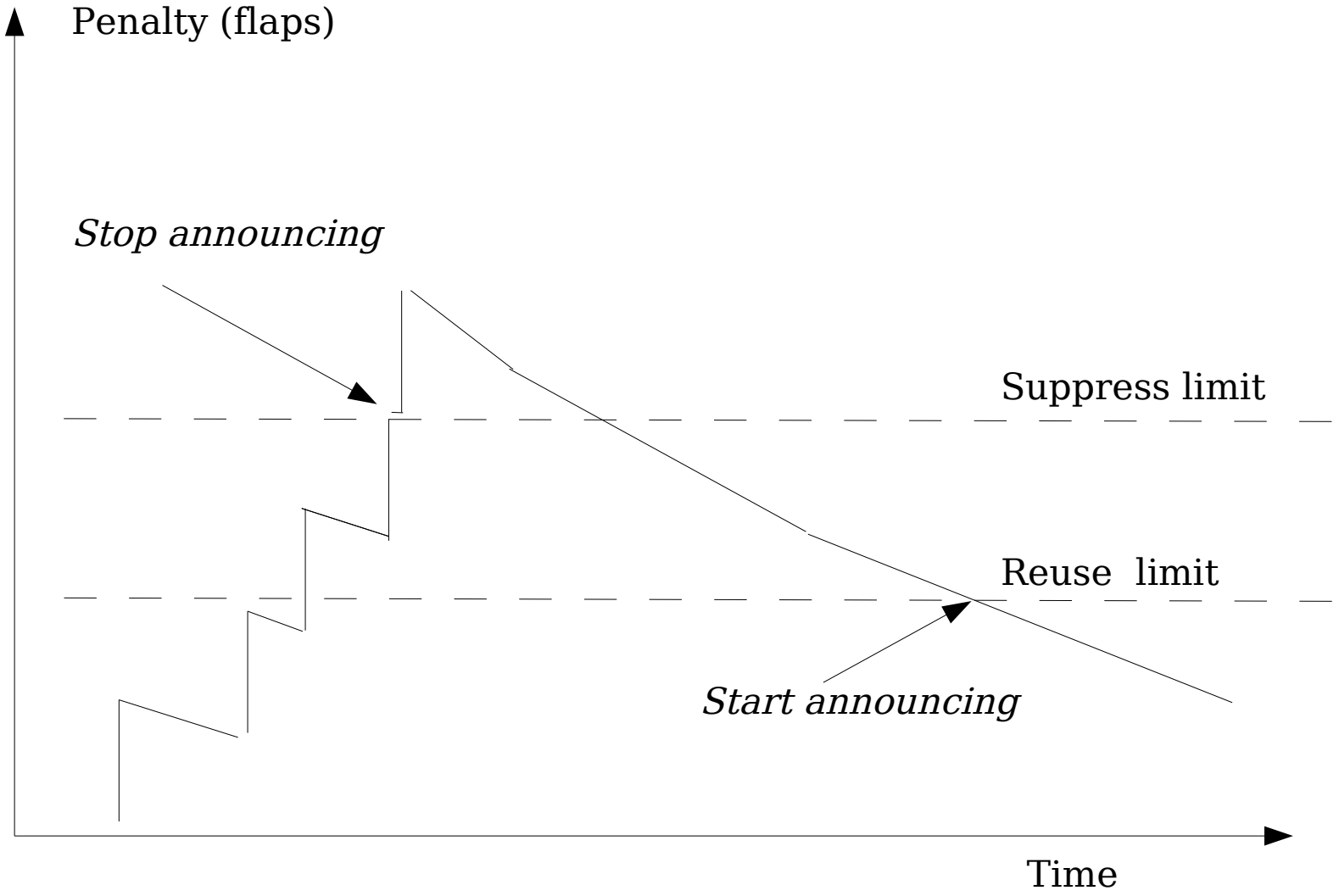
- A route that changes (withdrawn, announce or attribute change) leads to a new UPDATE message.
- A route that changes too often leads to *route flaps* that can ripple through the Internet
- A route can be a bad link that goes up and down, or an instable routing state
  - Some scenarios (clusters of routers) can even magnify flapping
- Only a small percentage of the Internet routes cause the majority of the flaps
- Flaps cause many UPDATE messages, that causes recomputation of FIBs that cause change in traffic.

Practical BGP: pages 238-241

Route flap damping (cont)

- Idea is to use history of route to predict future
- Introduce a penalty every time a route changes (eg 1000)
- Decay penalty exponentially using a half-time (eg 15 minutes)
- Stop announcing route using two levels:
  - Suppress/Cutoff-threshold to stop announcing (eg 3000)
  - Reuse threshold to start announcing (eg 750)
- Note: If one specific route is suppressed, a less specific route can be used for traffic
- Route flap dampening is most effective if everyone uses it towards the edges. Why?
  - But is usually installed towards upstream to protect from instable remote prefixes.

Route flap damping example



Four-byte AS numbers

- 2-byte ASNs are quickly running out
- 4-byte ASNs have been standardized re-using the AS_PATH and a migration technique using a special 2-byte ASN: 23456.
- The migration technique maps all 4-byte ASNs to 23456 when NEW speakers (that have four-byte AS capability) talk to OLD speakers (those that do not have 4-byte AS capability)

AS numbers in BGP

- Where does BGP carry AS numbers?
  - In the UPDATE message (my ASN)
  - In the AS_PATH attribute
  - In the Aggregator attribute
  - In Communities attributes
- A NEW speaker announces 4-byte AS as a capability
  - The capabilty also includes myASN.
- NEW speakers use the AS_PATH attribute for 4-byte ASNs

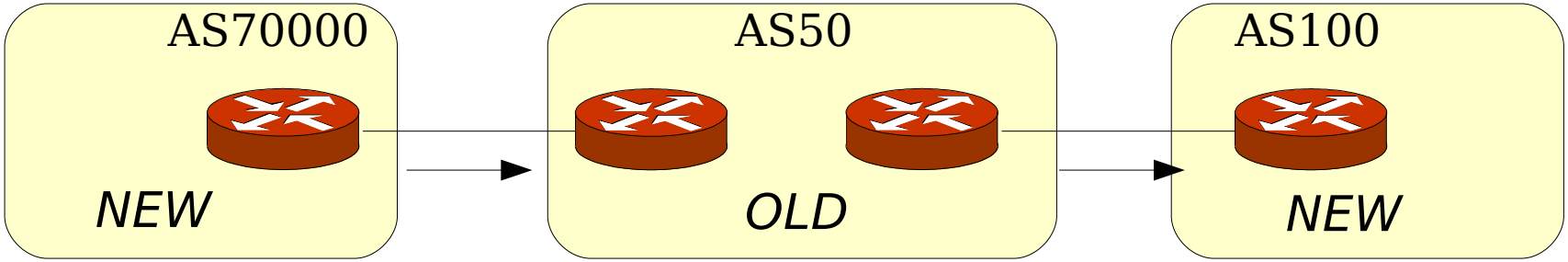
New attributes

- Two new attributes (optional transitive)
  - AS4_PATH
  - AS4_AGGREGATOR
- Only used to “tunnel” 4-byte AS information over non 4-byte AS clouds.

NEW speaking with OLD

- NEW converts all 4-byte ASNs in AS_PATH to 23456
- NEW creates the attribute AS4_PATH to “tunnel” the 4-byte AS-path to other NEW speakers.
- When NEW receives a route from OLD with an AS4_PATH attribute, it constructs a new AS_PATH replacing all 23456 with the corresponding 4-byte AS:s in AS4_PATH.

Example



AS_PATH:
532
90000

AS_PATH: 23456
532
23456
AS4_PATH: 70000
532
90000

AS_PATH:
50
70000
532
90000

- How can loop detection work?
- Because you only make loop detection concerning your own AS. And that is never AS_TRANS.
- But AS_PATH looks strange: 23456 may appear many times