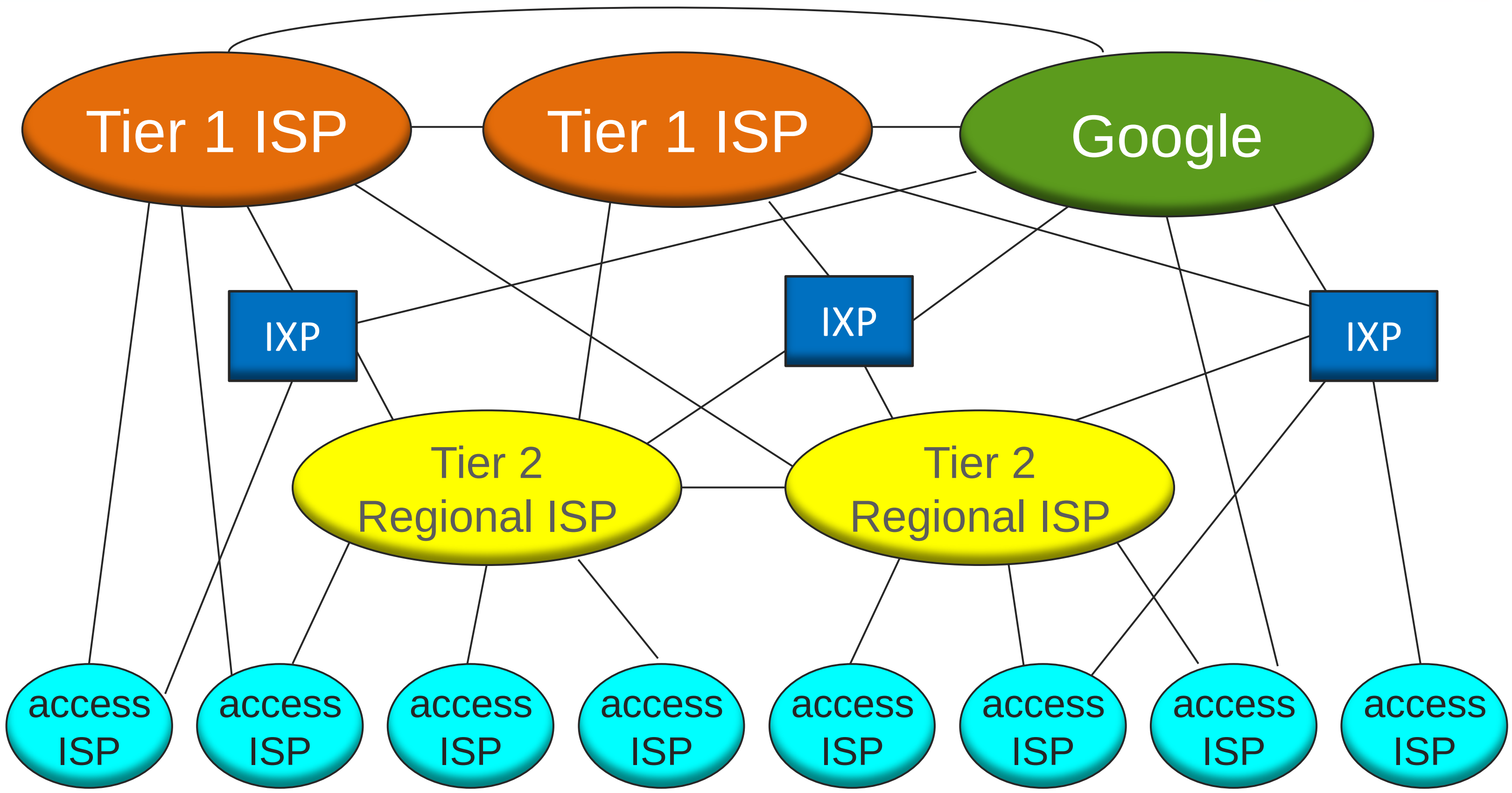
Institute of Telecommunications
Warsaw University of Technology

internet technologies and standards

Piotr Gajowniczek

BGP (Border Gateway Protocol)

structure of the Internet

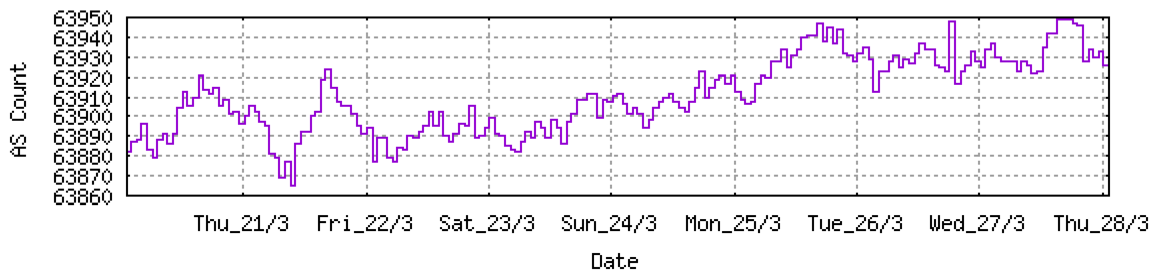


- *at center: small # of well-connected large networks*
  - "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
  - content provider network (e.g, Google): private network that connects its data centers to Internet, often bypassing tier-1, regional ISPs

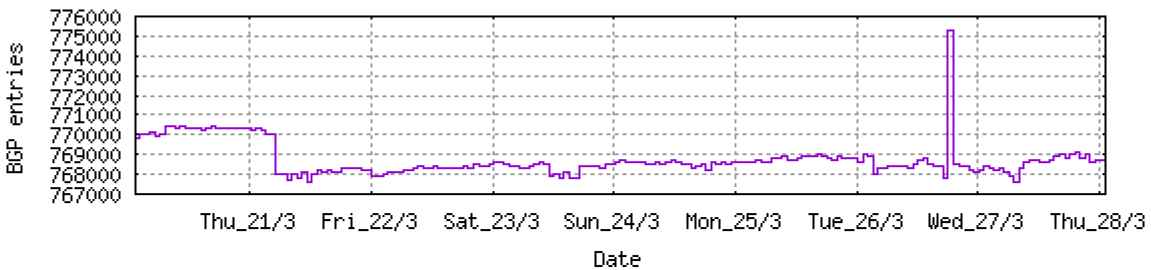
routing in the Internet

- *the Internet is organized as a set of independent Autonomous Systems*
  - the AS is a collection of networks under single technical administration
  - the AS appears to the outside world as having a coherent routing plan and presents unique view on what destinations are reachable through it
- *the AS can use many different routing protocols*
  - the routing protocols inside the AS = Interior Routing Protocols (IGP)
  - the routing protocol between the ASs = Exterior Routing Protocol (EGP)

AS count



BGP table size



www.cidr-report.org

routing protocols for IP networks

Protocol	**Type**	**Scalability**	**Metric**	**IP classes**
RIP-1	Distance vector	Small	Hop count	Classful
RIP-2	Distance vector	Small	Hop count	Classless
OSPF-2	Link state	Large (hierarchical)	Cost	Classless
IS-IS	Link state	**Very large** (hierarchical)	Cost	Classless
IGRP	Distance vector	Medium	Bandwidth, delay, load, MTU, reliability	Classful
EIGRP	Dual	Large	Bandwidth, delay, load, MTU, reliability	Classless
BGP	Distance (Path) vector	Large (**non-hierarchical**)	**Vector of attributes**	Classless

BGP basics

- *BGP is Inter-Autonomous System routing protocol (EGP)*
  - BGP is used to route traffic between different AS systems
  - BGP is used to interconnect ISPs and connect Enterprise networks to ISPs
- *when is the BGP needed?*
  - AS allows to pass packets between different ASs
  - AS has multiple connections to other ASs
  - AS wants to manipulate the flows of traffic leaving or entering this AS
- *BGP is a distance (path) vector routing protocol*
  - BGP peers exchange Path and NLRI for destination-based routing
  - maintains a list of AS's through which this NLRI traverses to prevent loops
  - advertises only the best routes to neighbors
- *BGP is CPU and memory consuming*
  - BGP „working domain" is a whole Internet!

BGP policy-based routing

- *BGP allows **policy-based routing***
  - a key strength of the BGP
  - it means that using BGP you can implement certain rules to manage traffic flows exchanged with other ASs
- *main reasons for using Routing Policy:*
  - business-related aspects
  - protecting the local AS (and other ASs) from bogus and unexpected NLRI from customers and other peers
  - protecting external peers or transit networks from instability inside the AS
  - optimize local AS ingress and egress traffic
  - …

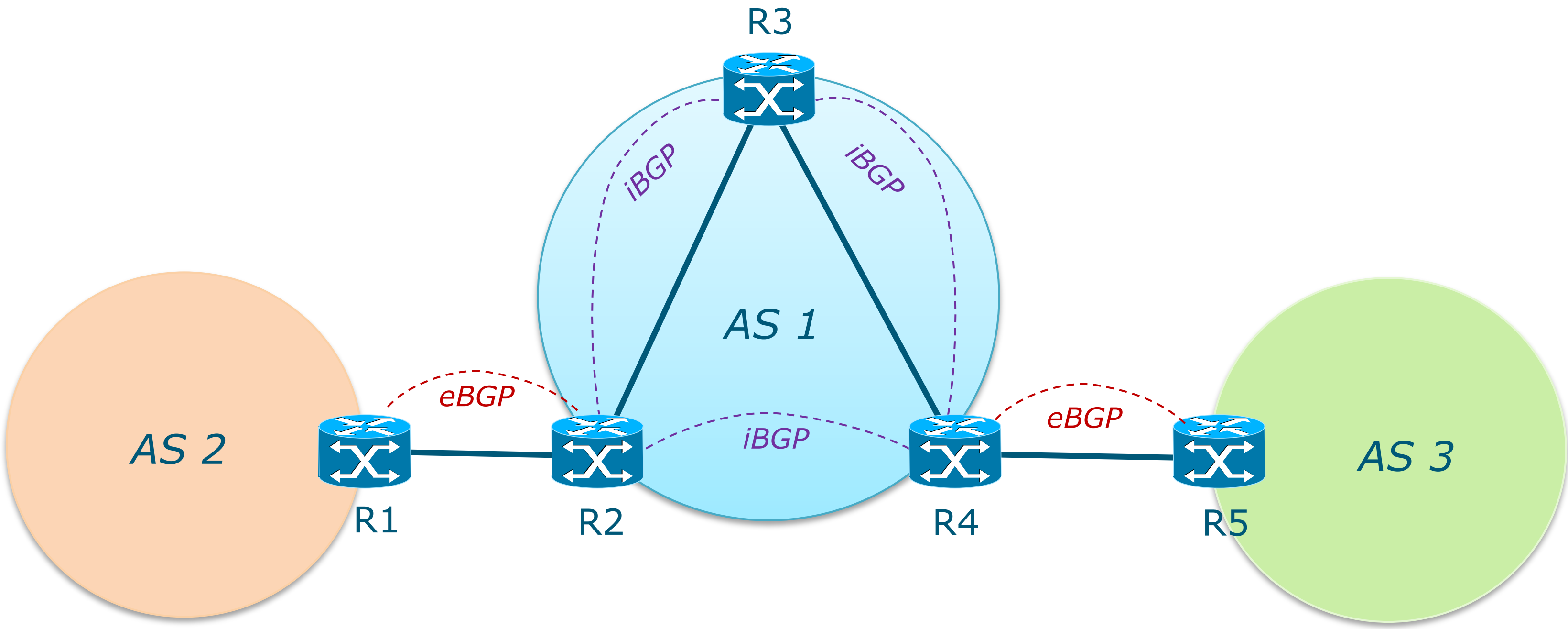
relationships between ASs (ISPs)

- *Peering (private & public) vs Transit*
- *Peering*
  - usually symmetric traffic flows
  - mutual benefit from traffic exchange = no tariffs
  - can be private or public
- *Transit ("paid peering")*
  - ISP pays for a BGP neighbor in upstream network
  - upstream ISP provides up to a full Internet table to the client
  - commercial agreement on conditions
    - tariffs
    - exchange point/access link
    - acceptable use policy

BGP – more details

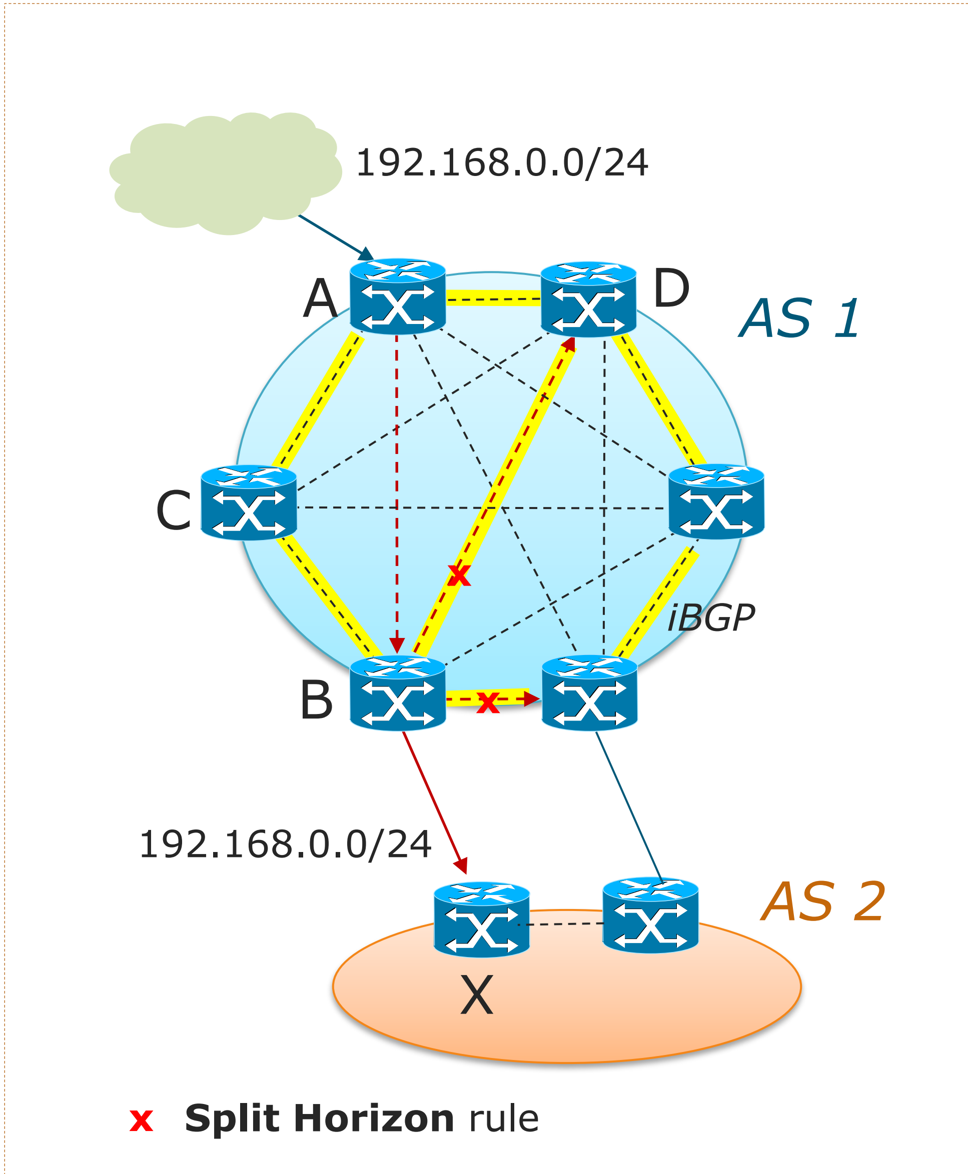
- *router running BGP is called a BGP speaker (or peer)*
- *no peer discovery procedures*
  - peers are configured manually
  - by telling each router what the IP address of its peer is
- *BGP uses triggered updates*
  - at startup BGP peers exchange full routing tables
  - then, only changes are advertised
- *BGP uses TCP on port 179 as its transport protocol*
  - peers establish TCP connection to exchange routing information
  - periodic keep-alive is used to verify TCP connectivity
- *two types of BGP sessions*
  - external BGP (eBGP) – if two peers belong to different ASs (they are usually directly connected)
  - internal BGP (iBGP) – if two peers belong to the same AS (they are usually not directly connected, the IGP protocol must run to assure connectivity)

why do we need iBGP?



- *to synchronize BGP routing information received by various peers*
- *to advertise prefixes from one AS to another (transit)*
- *full iBGP mesh is required*
  - why?
  - what if R3 does not run iBGP?

route propagation; split horizon rule

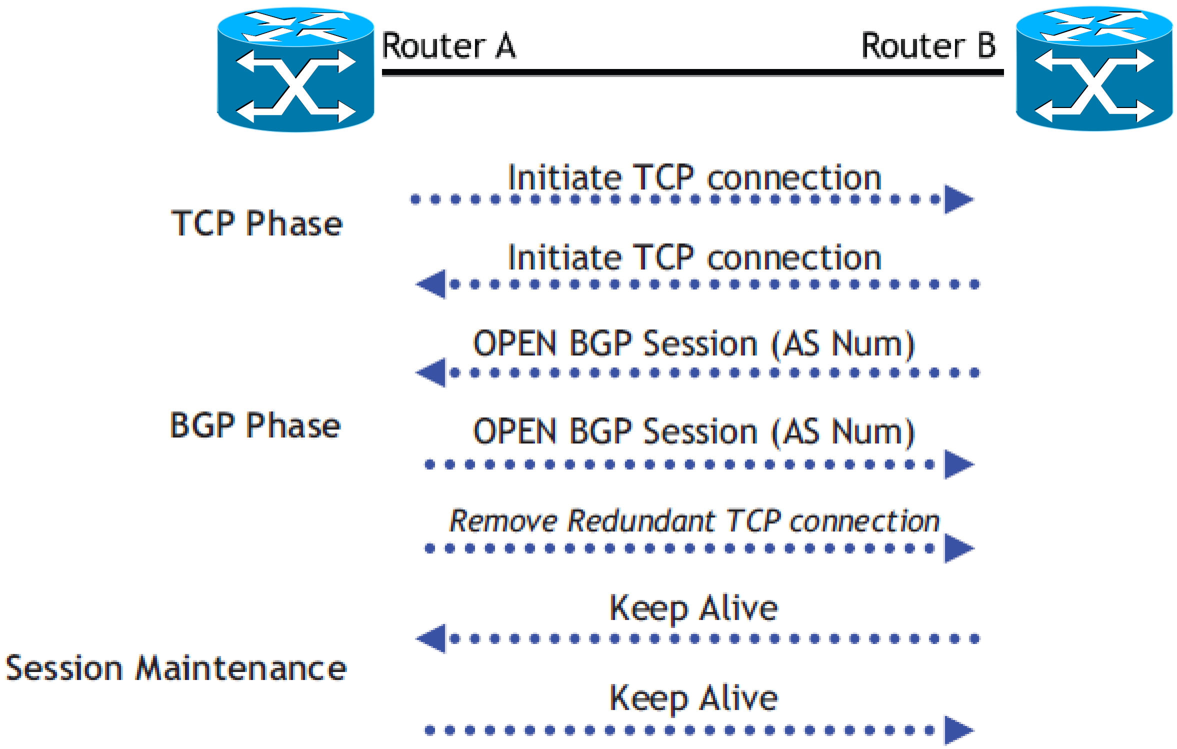


- *Route propagation*
  - occurs at the edges of the AS accepting NLRI into the BGP
    - by exporting directly connected or static routes into the BGP
    - by importing prefixes learned over eBGP sessions
  - policies can be implemented at:
    - the local injection point (router A)
    - AS boundary (router B)
- *Route propagation within an AS*
  - control plane: iBGP with Split Horizon rule (that's why we need full mesh iBGP)
  - data plane forwarding: by IGP
- *IGP optimization:*
  - do not export BGP routes into IGP and vice versa (except static or directly connected networks associated with customer networks)
  - assure fast IGP convergence (eg. using BFD)

BGP state machine

Phase	State	Description	Next state
TCP	*Idle*	Initialization, TCP initiation	*Connect*
TCP	*Connect*	Waiting for completing TCP connection	*OpenSent* or *Active* if no TCP after ConnectRetry
TCP	*Active*	Resetting ConnectRetry timer	*Connect*
BGP	*OpenSent*	Sending OPEN message	*OpenConfirm*
BGP	*OpenConfirm*	Exchanging KEEPALIVE messages	*Established*
BGP	*Established*	Sending/receiving KEEPALIVE, UPDATE, NOTIFICATION; operational state	-

BGP - initialization



BGP messages and databases

- *Open*
  - exchange parameters; sent after the TCP connection is established; includes hold time - the maximum time between consecutive keep alive messages and router ID - highest IP interface address
- *Keep alive*
  - sent periodically to maintain BGP session
- *Update*
  - contains information about one path: prefix (NLRI) and path attributes (set, update or withdraw)
- *Notification*
  - sent in case of error condition

Open

Ver	AS	Hold Time	BGP Id	O-P L	Opt Par
8	8	24	32	8	n

header

Marker	Length	Type
128	16	8

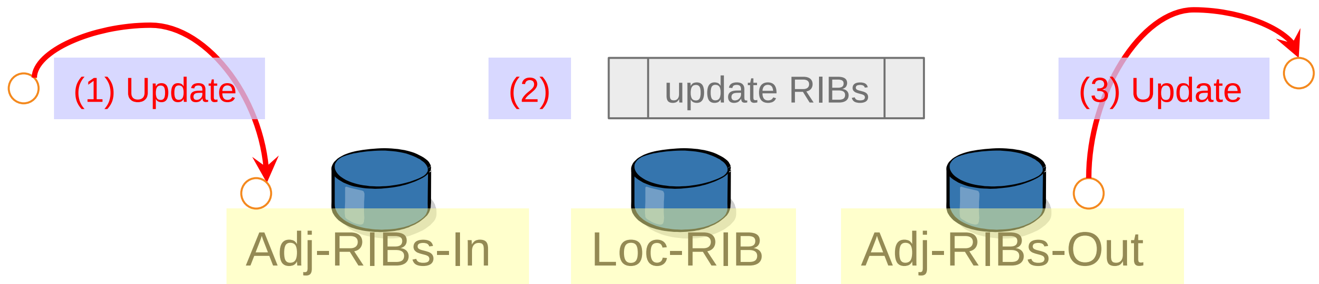
Update

W-P L	Withdrawn Paths	P-A L	Path Attributes	NLRI
16	n	16		

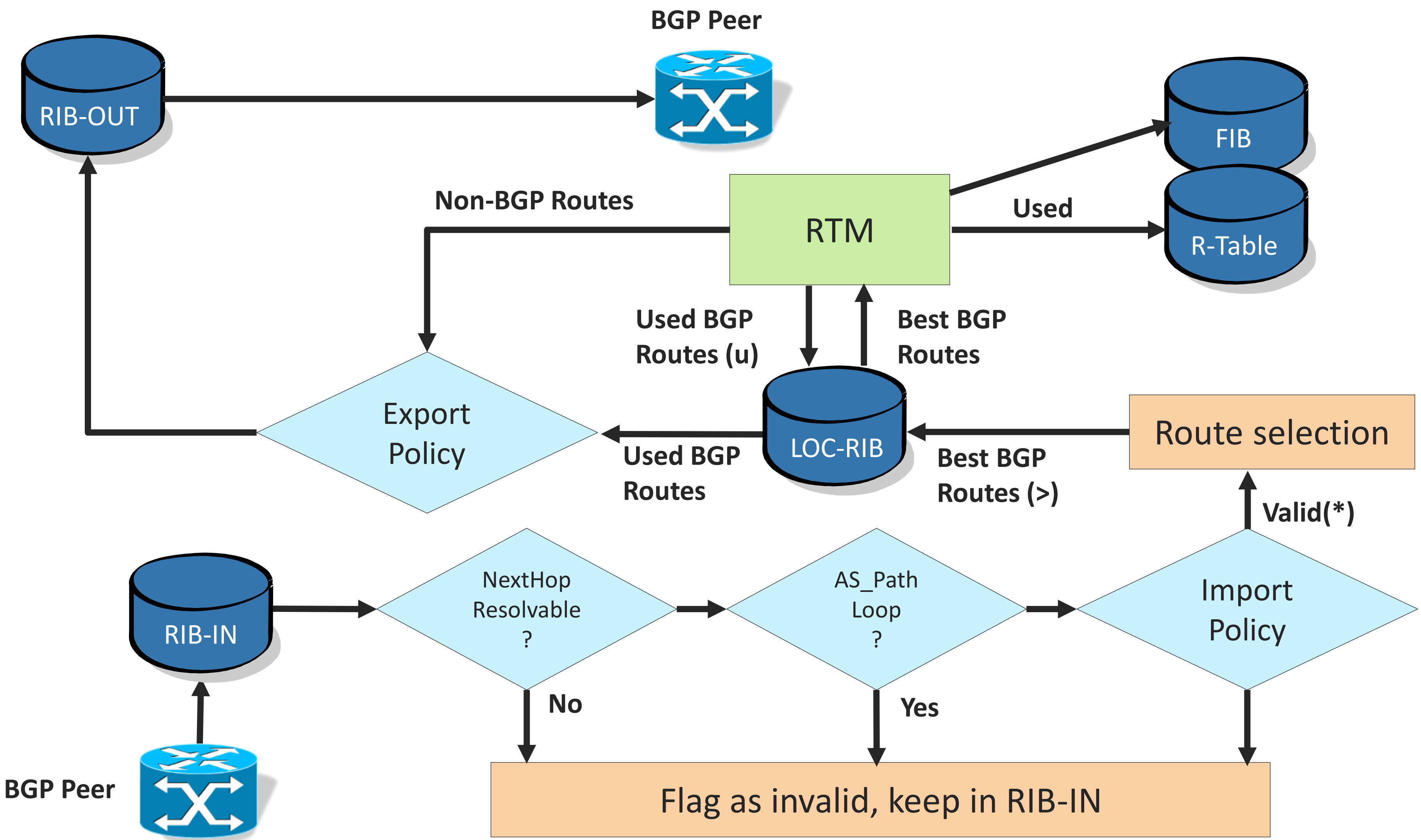
AS Path

Notification

Code	SubC	Data
8	8	48



BGP route processing



BGP attributes

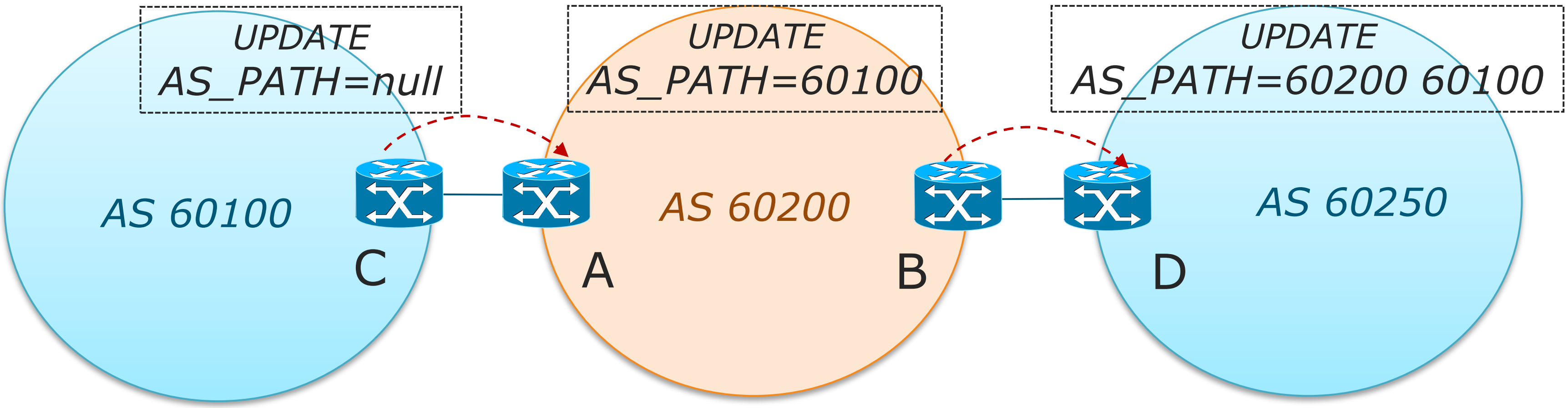
- *BGP metrics are called path attributes*
- *the following path attribute categories exist*
  - well-known mandatory – must be recognised by all implementation and must be included in all update messages
  - well-known discretionary - must be recognised by all implementation but needn't be included in all update messages
  - optional transitive – may not be recognised by some implementation, when not recognised must be propagated to their neighbours
  - optional nontransitive – may not be recognised by some implementation, when not recognised must be dropped

BGP attributes - examples

- *well-known mandatory attributes*
  - AS-path
  - Next-hop
  - Origin
- *well-known discretionary attributes*
  - Local preference
  - Atomic aggregate
- *optional transitive attributes*
  - Aggregator
  - Community
- *optional non-transitive attributes*
  - Multi-exit-discriminator (MED)

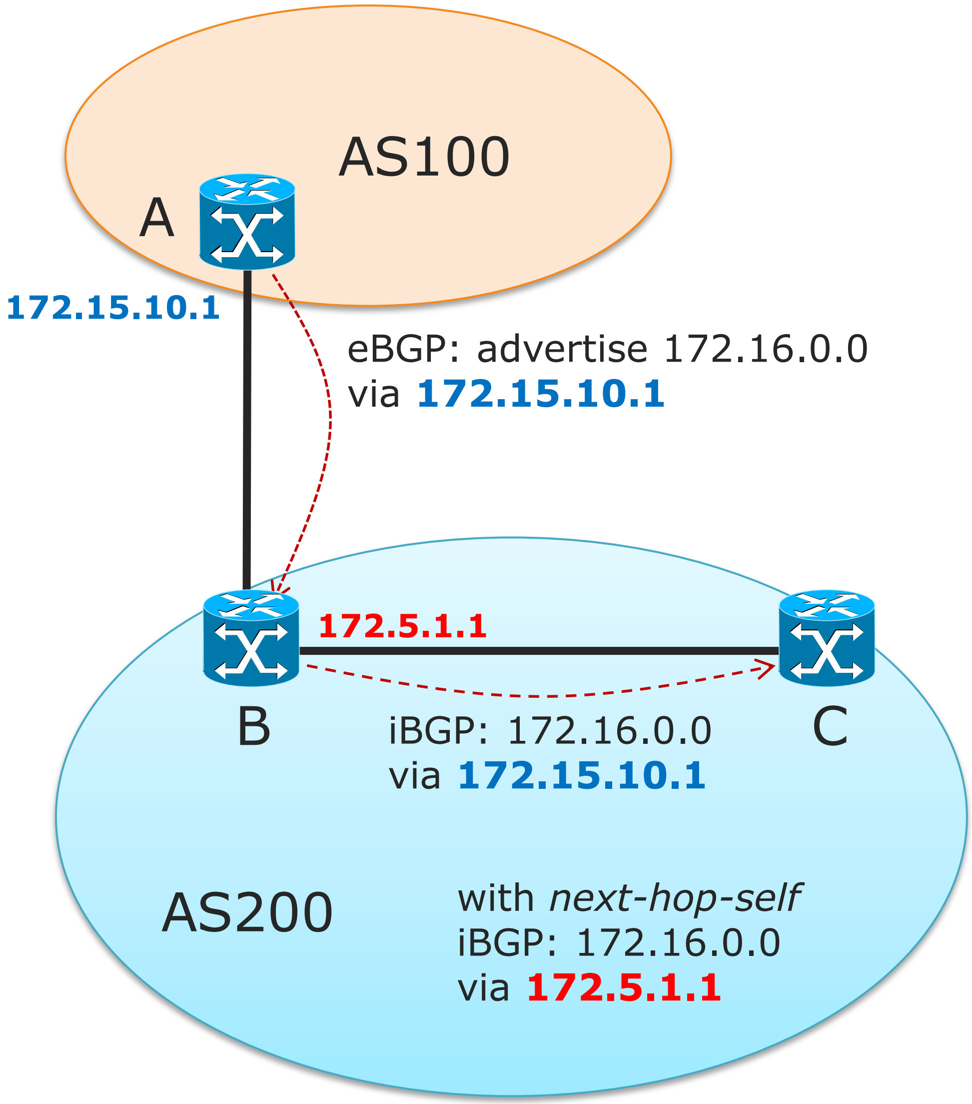
AS Path attribute

- *contains the list of AS identifiers on the path toward the destination*
    - whenever a route passes through AS its identifier is pre-penned to it by the BGP router
- *allows to detect and eliminate route loops*
- *is modified by a border router when propagating an update across an AS boundary*



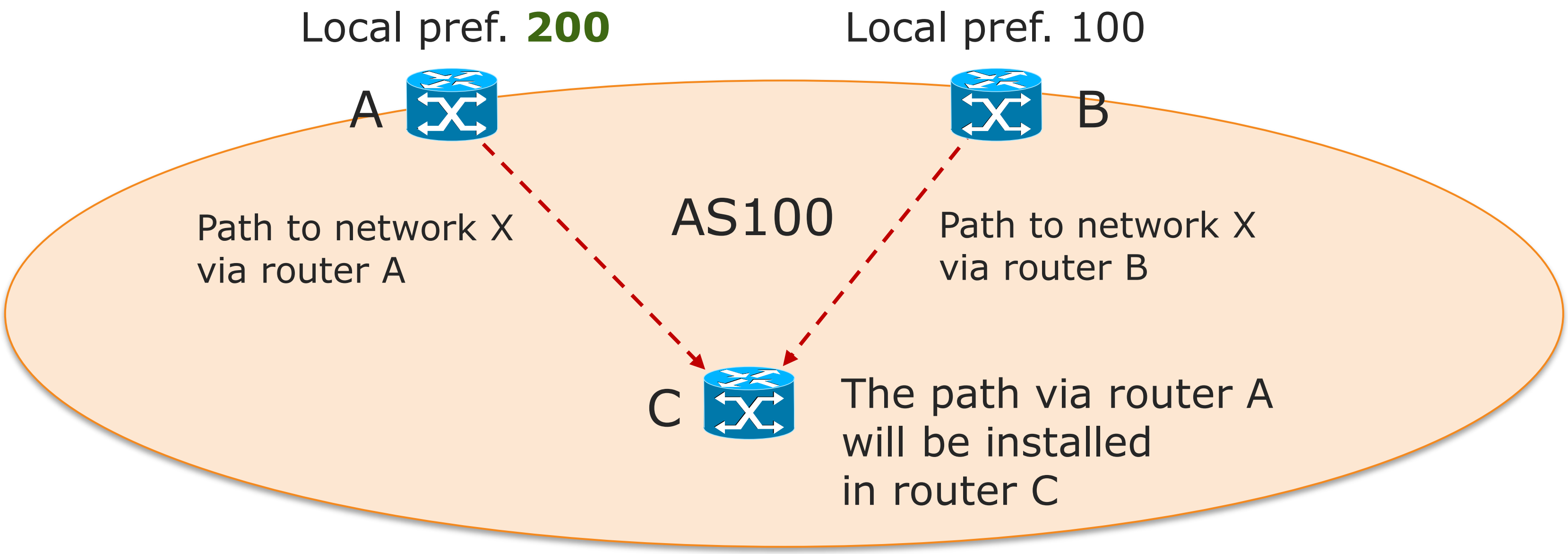
Next-hop attribute

- *address of the next router on the path towards the destination*
- *for eBGP it is the address of the router that has sent the path information (address of the peer)*
  - router A advertises the network 172.16.0.0 to B with next hop 172.15.10.1
- *for iBGP the next hop is carried from eBGP unchanged by default*
  - router B will advertise network 172.16.0.0 to C with next hop 172.15.10.1
- *router C has to know how to reach 172.15.10.1*
  - next-hop-self (change next hop to local interface)



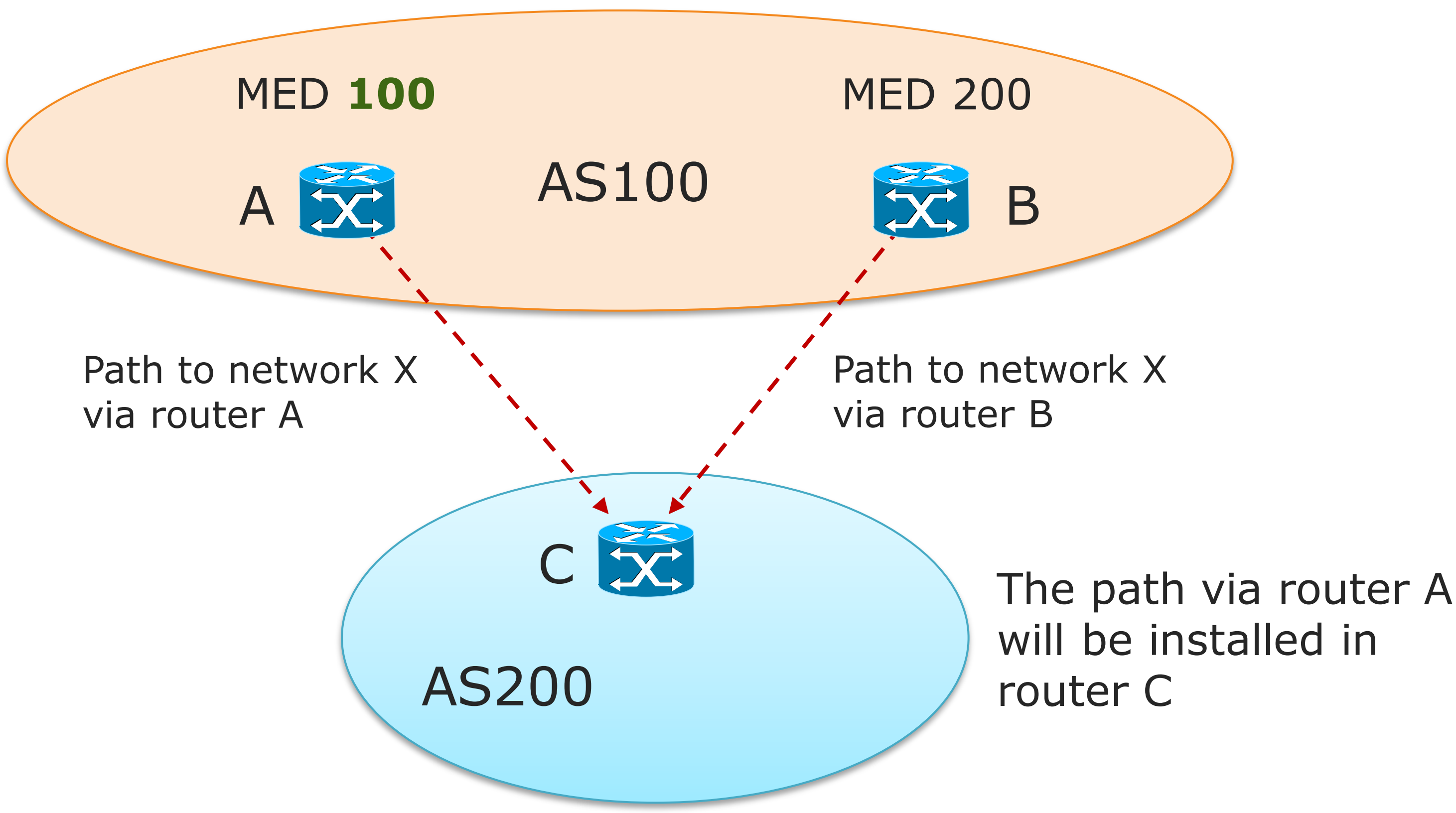
Local Preference attribute (outbound traffic)

- *local preference is an attribute configured on the router and exchanged only inside the AS*
- *the route from router with higher local precedence value will be preferred*



MED attribute (inbound traffic)

- *the MED attribute is configured on the router and exchanged between adjacent ASs*
- *the MED attribute is an indication to external peers about the preferred path into given AS*

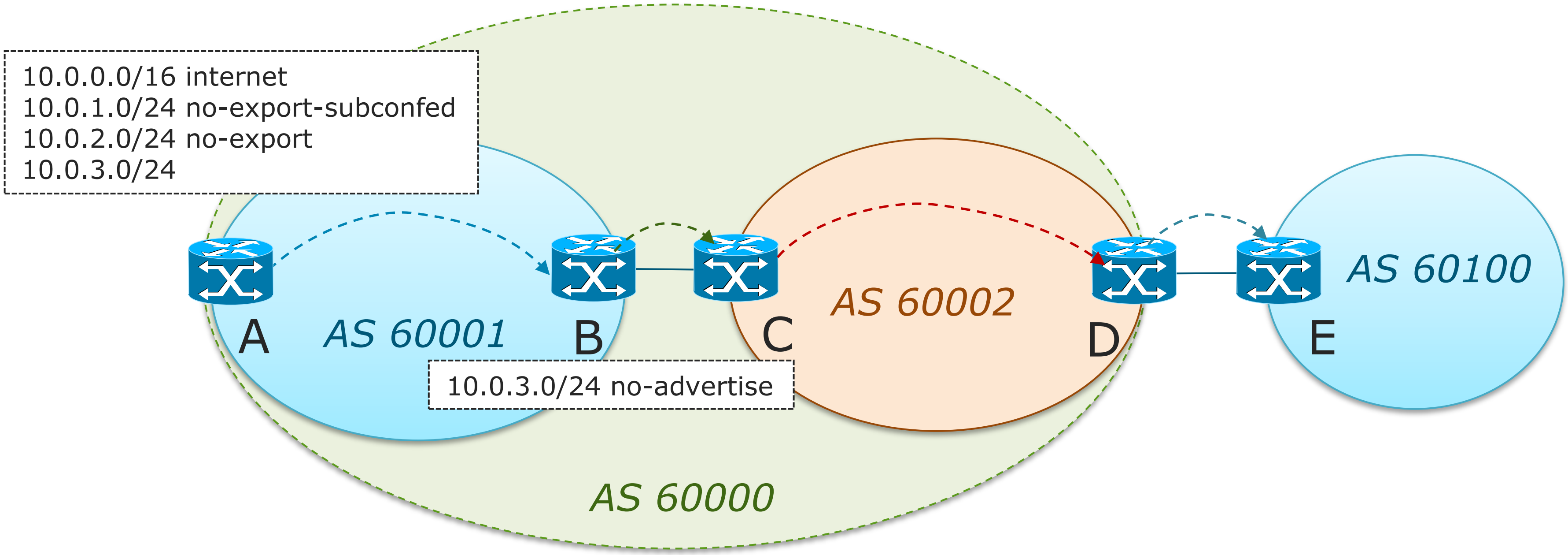


Community attribute

- *Community attributes allow tagging paths*
  - the routes can be tagged on incoming or outgoing interface
  - community is a list of values
- *tagging is used for route filtering and selection*
- *Community attributes are used to implement consistent BGP policy routing rules*
- *routers that understand community attribute must be configured to use it otherwise the attribute is dropped*
- *known communities*
  - no-export – do not advertise the route to external peers
  - no-advertised – do not advertise the route to any peer
  - internet – advertise the route to the Internet
  - no-export-subconfed – used in confederation to prevent sending packets outside AS

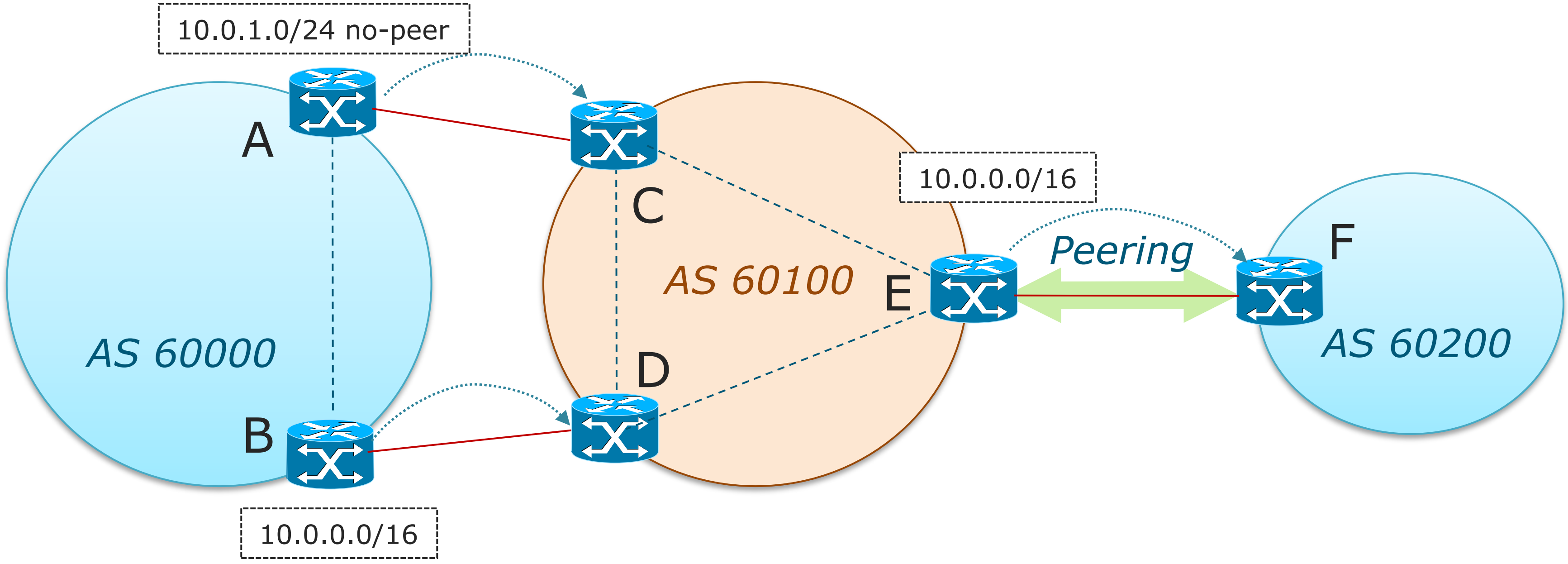
Communities – route propagation example

- *no-export, no-export-subconfed, no-advertise Communities*
- *ASs: 60001 and 60002 are confederated within AS 60000*
- *who gets what?*
  - 10.0.0.0/16 is received by …
  - 10.0.1.0/24 is not advertised to …
  - 10.0.2.0/24 is not advertised to …
  - 10.0.3.0/24 is not advertised to …



Communities – route propagation example

- *no-peer Community*
- *AS in the middle is a transit provider to AS 60000, AS 60100 and 60200 are in peering relationship*
- *the prefix advertised by router A is more specific than the one advertised by B, and has a no-peer community*
- *therefore router F receives the aggregate route only*

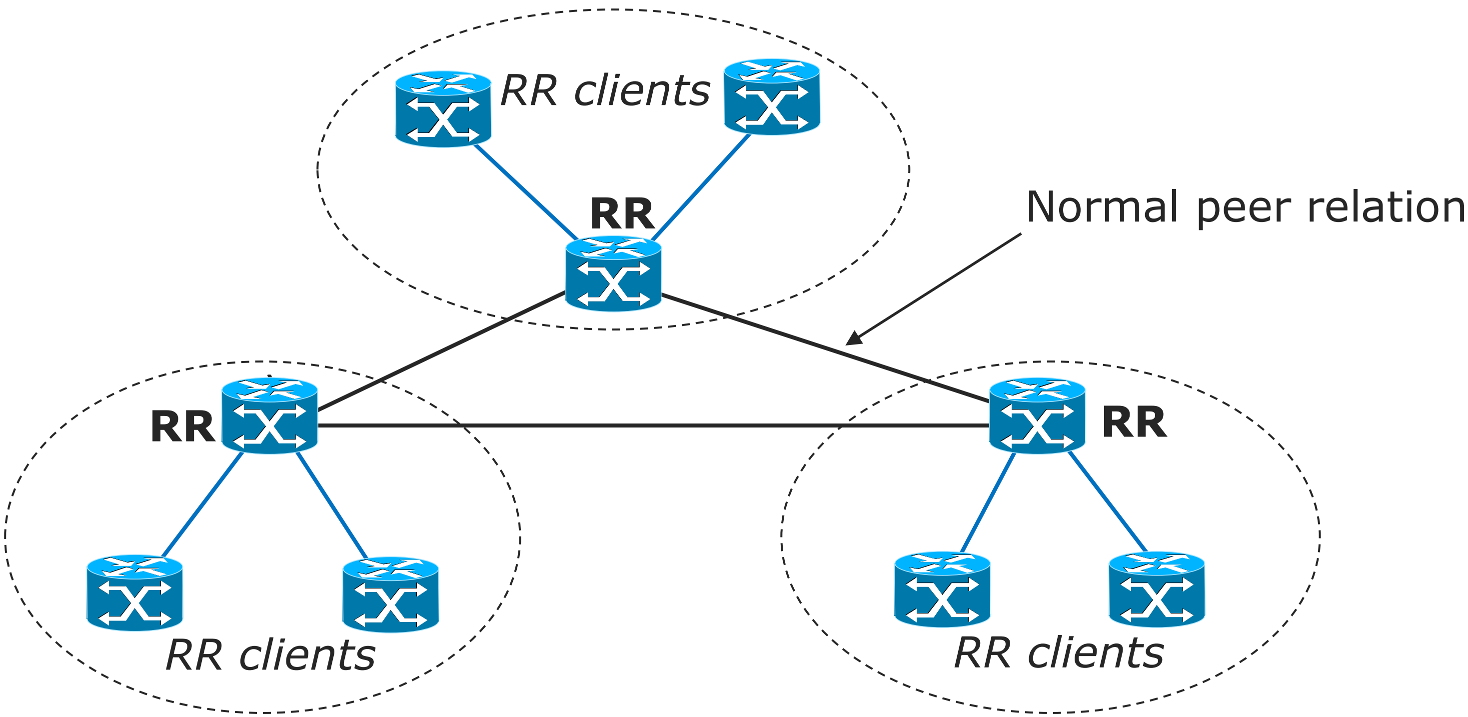


route selection decision

- *consider synchronised routes with no loops and valid next-hop address*
    - prefer routes highest local preference
    - prefer routes originated by local router
    - prefer shortest AS path
    - prefer lowest origin attribute (IGP<EGP<incomplete)
    - prefer lowest MED
    - prefer eBGP over iBGP paths
    - prefer paths through the closest IGP
    - prefer oldest eBGP paths

route reflectors

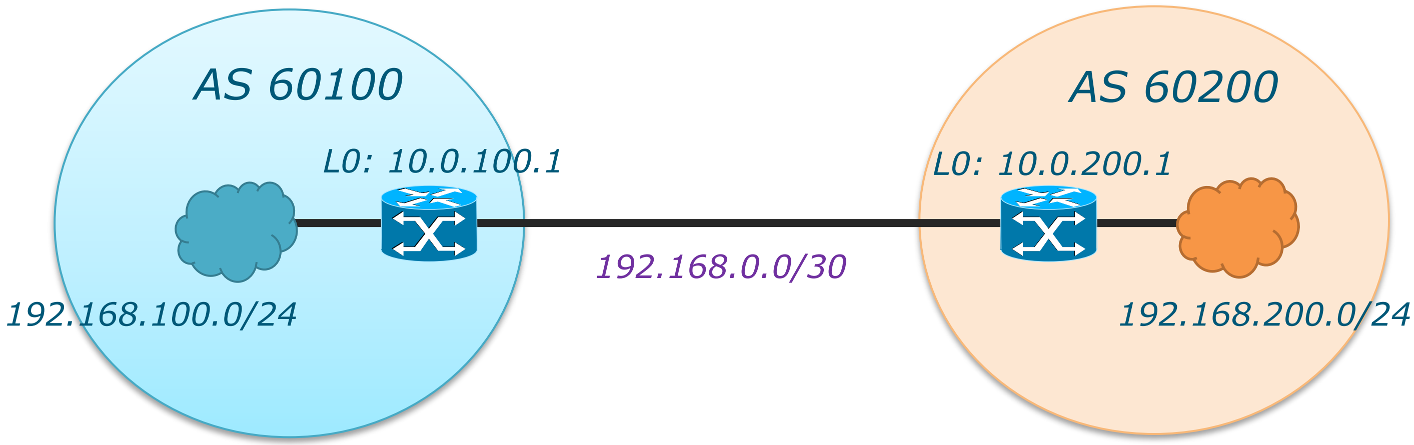
- *route reflectors allows to cope with the iBGP sessions full mesh problem*
- *the route reflector advertises routes learnt via iBGP to other local BGP peers*
  - this reduces the number of point-to-point relations between BGP speakers
- *many route reflectors can be configured in one AS*



BGP (Border Gateway Protocol)

configuring the BGP

BGP configuration example




```
R1(config)#router bgp 60100
R1(config-router)#neighbor 10.0.200.1 remote-as 60200
R1(config-router)#neighbor 10.0.200.1 update-source loopback 0

R1(config-router)#neighbor 10.0.200.1 ebgp-multihop 4

R1#show run | section bgp

R1#show ip bgp
BGP table
version is 2, local router ID is 10.0.100.1
Status codes: s suppressed, d damped, h history, * valid, > best,
i - internal, r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

     Network          Next Hop Metric LocPrf Weight Path
*> 192.168.100.0/24 0.0.0.0       0            32768  i
```

http://www.bgp4.as/looking-glasses

BGP configuration example

- *advertising a prefix*

```
R1(config)#router bgp 60100
R1(config-router)#network 192.168.100.0 mask 255.255.255.0
```

- *null interface*

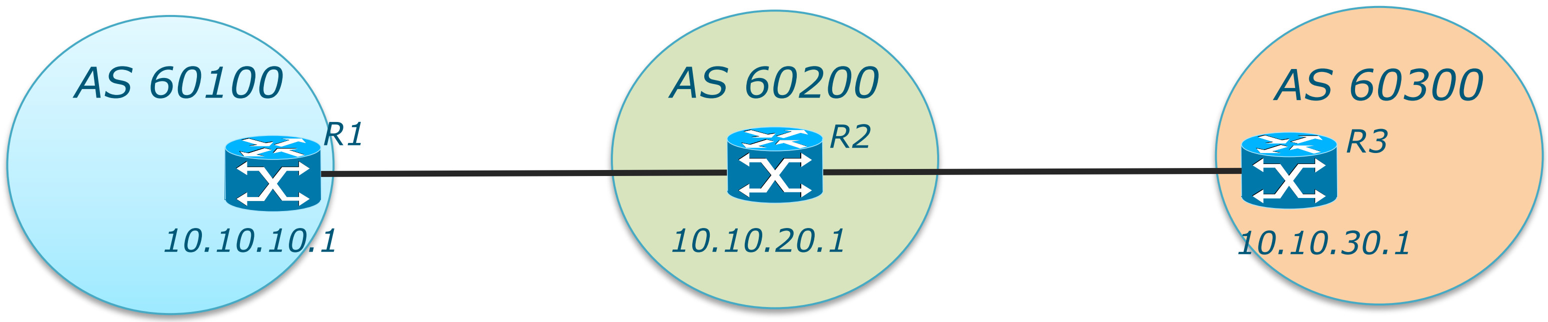
```
R1(config)#ip route 1.0.0.0 255.0.0.0 null 0
```

- *path prepend*

```
R1(config)#route-map MAP1 permit 10
R1(config-route-map)#set as-path prepend 1 1 1 1 1
R1(config-route-map)#exit

R1(config)#router bgp 60100
R1(config-router)#neighbor 10.0.200.1 route-map MAP1 out
```

prefixes filtering



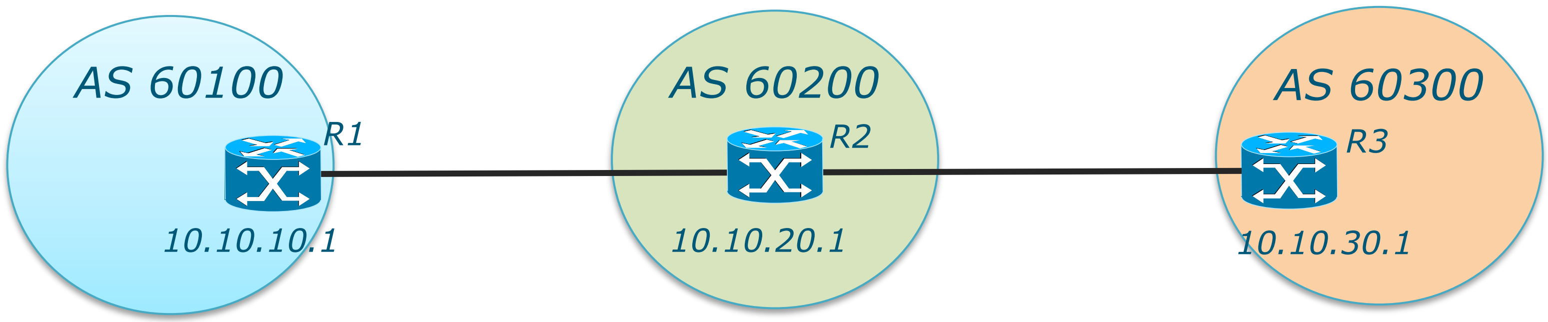
- *no transit in AS60200 – access list*

```
R2(config)#ip as-path access-list 1 permit ^$
R2(config-router)#neighbor 10.10.10.1 filter-list 1 out
R2(config-router)#neighbor 10.10.30.1 filter-list 1 out
```

- *no transit in AS60200 – no-export community*

```
R2(config)#route-map NO-EXPORT
R2(config-route-map)#set community no-export
R2(config)#router bgp 60200
R2(config-router)#neighbor 10.10.10.1 route-map NO-EXPORT in
R2(config-router)#neighbor 10.10.20.1 route-map NO-EXPORT in
```

prefixes filtering



- *Only allow prefixes that originated directly from AS 60100*

```
R2(config)#ip as-path access-list 1 permit ^60100$
R2(config)#route-map ASPATH_FILTER
R2(config-route-map)#match as-path 1
R2(config)#router bgp 60200
R2(config-router)#neighbor 10.10.10.1 route-map ASPATH_FILTER in
```

- *Only allow prefixes from AS 60100 and its directly connected ASs*

```
R2(config)#ip as-path access-list 1 permit ^60100_[0-9]*$
..
```

- *Deny prefixes that originated from AS 60000 and permit everything else*

```
R2(config)#ip as-path access-list 1 deny _60000$
R2(config)#ip as-path access-list 1 permit .*
...
```

BGP good practices

- ISIS and OSPF
  - used for carrying **infrastructure** addresses, not Internet prefixes or customer prefixes
  - **DO NOT:**
    - distribute BGP prefixes into an IGP
    - distribute IGP routes into BGP
    - use IGP to carry customer prefixes
- BGP in Cisco IOS is permissive by default
  - configuring BGP peering without using filters means:
    - all best paths on the local router are passed to the neighbour
    - all routes announced by the neighbour are received by the local router
  - can have disastrous consequences
  - good practice is to ensure that each eBGP neighbour has inbound and outbound filter applied

source: BGP Best Current Practices, ISP Workshops

aggregation

- aggregation = announcing the address block received from the RIR to the other ASes connected to your network
  - subprefixes of the aggregate may be used internally in the ISP network, announced to other ASes only to aid with multihoming
- example: ISP has 101.10.0.0/19 address block
  - put into BGP as an aggregate:

    ```
    router bgp 60101
    network 101.10.0.0 mask 255.255.224.0
    ip route 101.10.0.0 255.255.224.0 null0
    ```

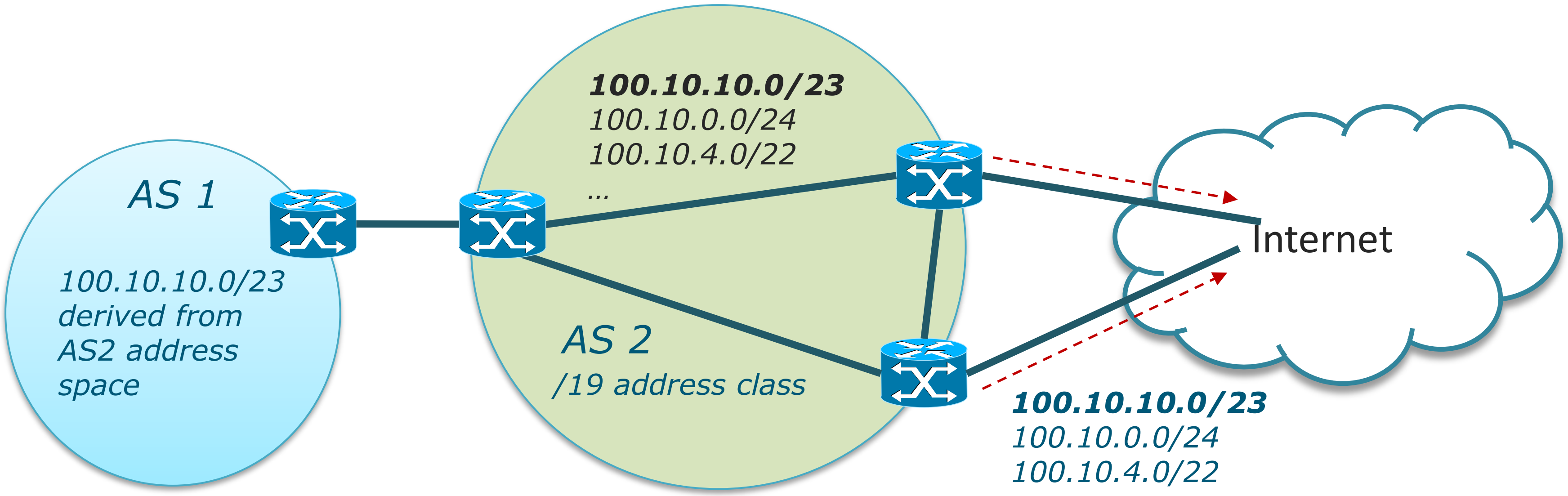
  - more specific prefixes within this address block ensure connectivity to ISP's customers; "longest match" lookup

```
router bgp 60101
neighbor 102.102.10.1 remote-as 60102
neighbor 102.102.10.1 prefix-list out-filter out

ip prefix-list out-filter permit 101.10.0.0/19
```

source: BGP Best Current Practices, ISP Workshops

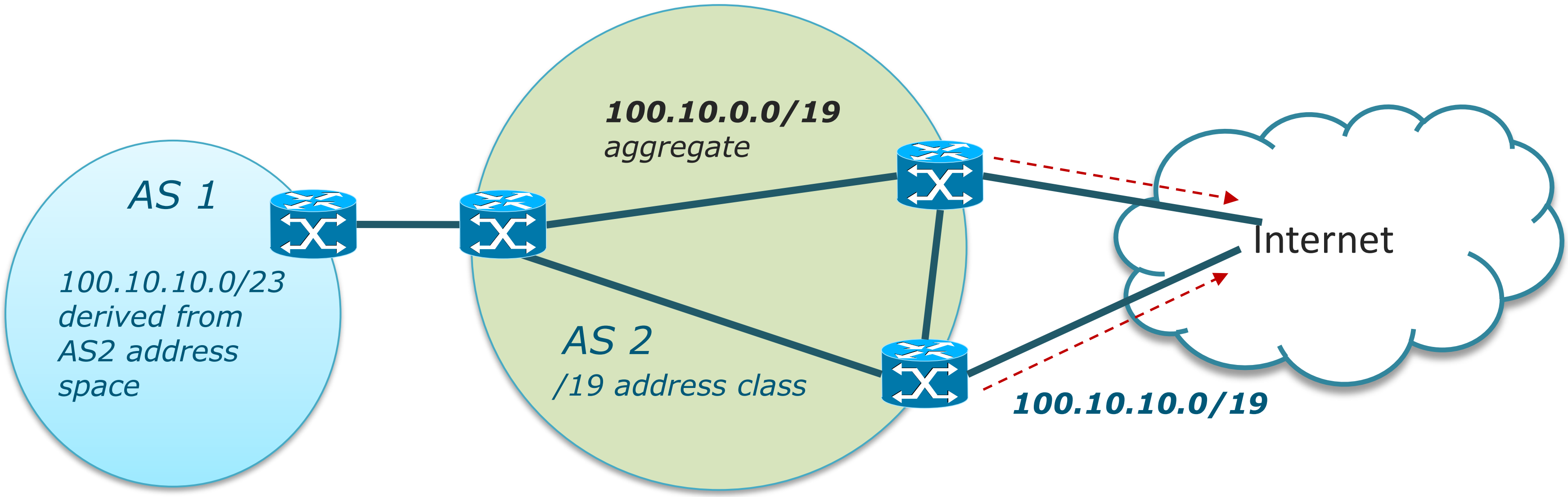
aggregation example



- Customer network advertised to the Internet
  - if customer link goes down, /23 becomes unreachable and is withdrawn from AS2 iBGP
  - the change propagates to Internet
  - when the network is back, the change propagates again
  - an example of bad practice

source: BGP Best Current Practices, ISP Workshops

aggregation example



- AS2 announces aggregate to the Internet
  - if customer link goes down, /23 becomes unreachable and is withdrawn from AS2 iBGP, but /19 aggregate is still advertised over eBGP
  - avoids BGP problems,
  - when the network is back, the /23 is reinjected to AS2 iBGP – the Intenet is visible again immediately
  - an example of **good** practice

source: BGP Best Current Practices, ISP Workshops

aggregation – efforts to improve

- why important?
  - convergence and stability (memory / CPU load less important)
- the CIDR Report
  - initiated and operated for many years by Tony Bates
  - now combined with Geoff Huston's routing analysis

    www.cidr-report.org
- lists the top 30 service providers who could do better at aggregating
  - RIPE Routing WG aggregation recommendations
    - IPv4: RIPE-399 — www.ripe.net/ripe/docs/ripe-399.html
    - IPv6: RIPE-532 — www.ripe.net/ripe/docs/ripe-532.html
- also computes the size of the routing table assuming ISPs performed optimal aggregation
  - website allows searches and computations of aggregation to be made on a per AS basis
  - flexible and powerful tool to aid ISPs

source: BGP Best Current Practices, ISP Workshops

distribution of prefixes

- *Receiving prefixes from: customer, peer, upstream (transit) provider*
- *Customer:*
  - check if they are assigned/allocated to this customer
  - if not assigned by the IPS itself, check (RIR databases)

```
router bgp 100
neighbor 102.102.10.1 remote-as 101
neighbor 102.102.10.1 prefix-list customer in
neighbor 102.102.10.1 prefix-list default out
!
ip prefix-list customer permit 100.50.0.0/20
ip prefix-list default permit 0.0.0.0/0
```

- *Peer*
  - accept only prefixes your peer ISP announced for advertising
  - send only prefixes you have announced
- *Upstream / Transit provider*
  - best practice: receive default route or a prefix to be used as default
  - not the whole Internet table (until necessary)
  - traffic control – via multihoming