

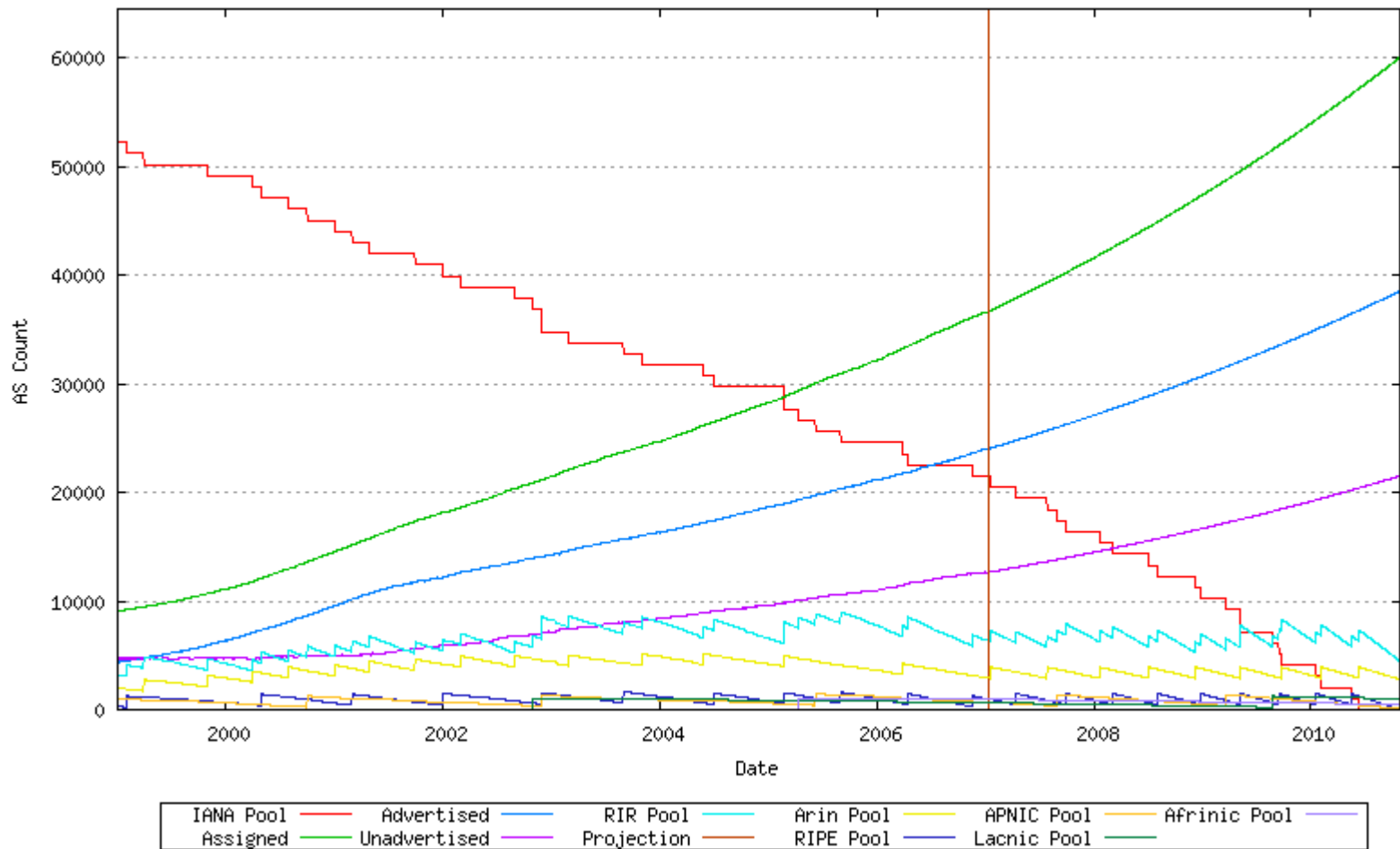
4-Byte AS Numbers



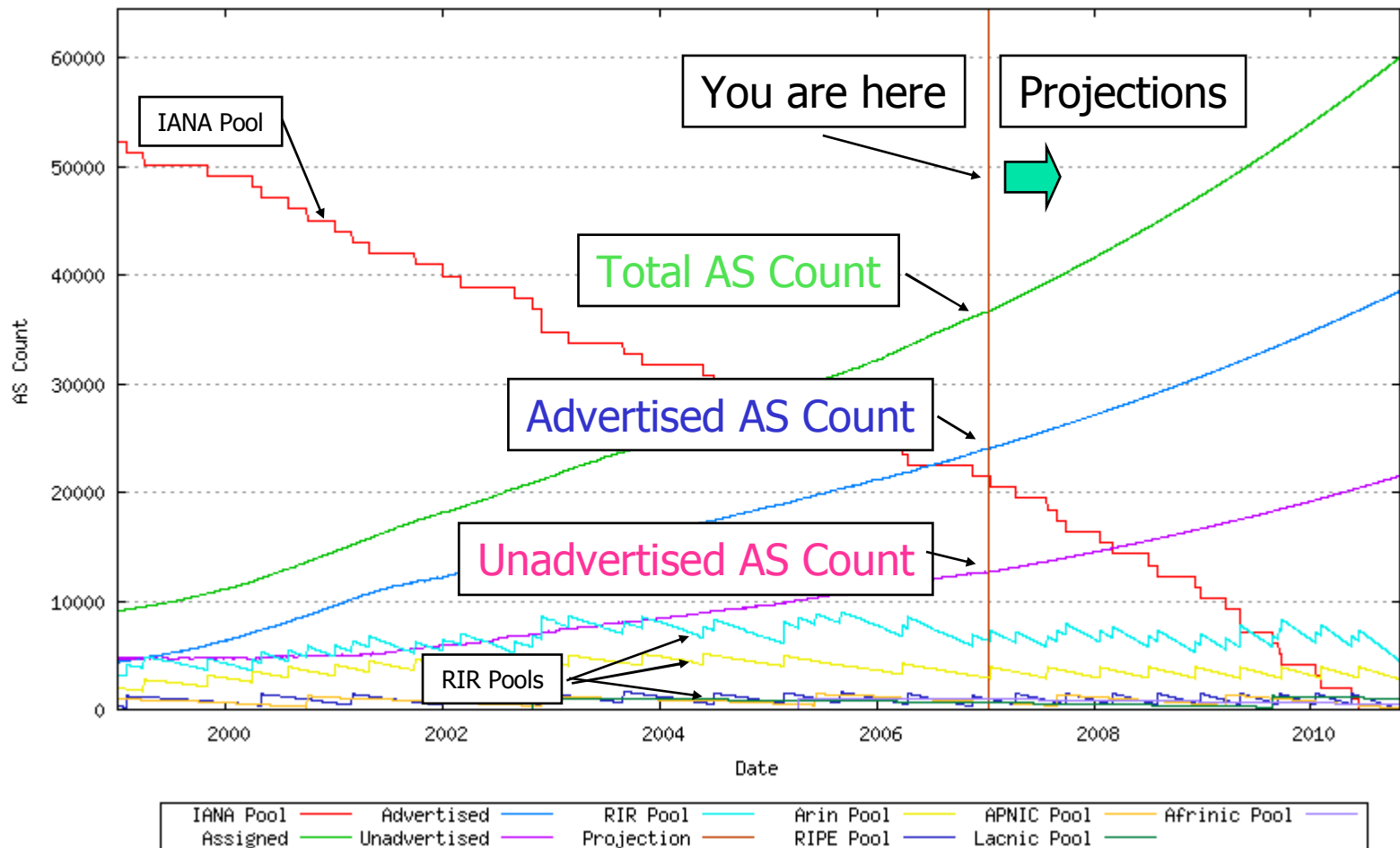
The view from the Old BGP world

Geoff Huston
February 2007
APNIC

AS Number Consumption



AS Number Consumption





4-Byte AS Numbers

- We were running into exhaustion of the 2-Byte AS Number pool
 - Estimated exhaustion time: 2100 UTC 29 October 2010
 - See <http://www.potaroo.net/tools/asns>



RIRs and 4-Byte AS Numbers

- From **1 January 2007** the RIRs are allocating 4-Byte AS numbers (upon specific request)
- From **1 January 2009** the RIRs will be allocating 4-Byte AS numbers by default (leaving some 2-Byte AS numbers available upon specific request)



The 4-Byte ASN Approach

Objectives:

- Change as little as possible in the BGP spec
- Be 'backward compatible' with 2-Byte AS BGP implementations
 - Negotiate 4-Byte capability when opening a BGP session
 - Automatically adjust behaviour when peering with 2-Byte BGP peers
 - Assume a 2-Byte "persona" with 2-Byte peers
- Use 4-Byte "persona" with 4-Byte peers
 - Preserve "basic" AS semantics in BGP when peering with 2-Byte BGP peers
 - Preserve BGP's loop detection properties
 - Preserve AS Path length metric properties
- No 'flag day' transition
 - Allow 2-Byte BGP implementations to continue to operate indefinitely in a mixed 2 / 4-Byte AS world with complete reachability
 - Allow for piecemeal deployment of 4-Byte BGP implementations



What's changed?

- BGP Update messages in the 2-Byte world:
 - May 'lie' in parts of the AS Path
 - May be larger in size
- But prefix reachability information is still communicated between 2-Byte and 4-Byte BGP "realms"



What does this imply?

If you are a 2-Byte AS
as most (all) of you are today

and you don't want to upgrade all your
instances of BGP today
something you probably want to avoid (or at least defer!)

then you don't have to do anything at all!

NOTHING changes!



Thank You



Well, **almost** nothing!



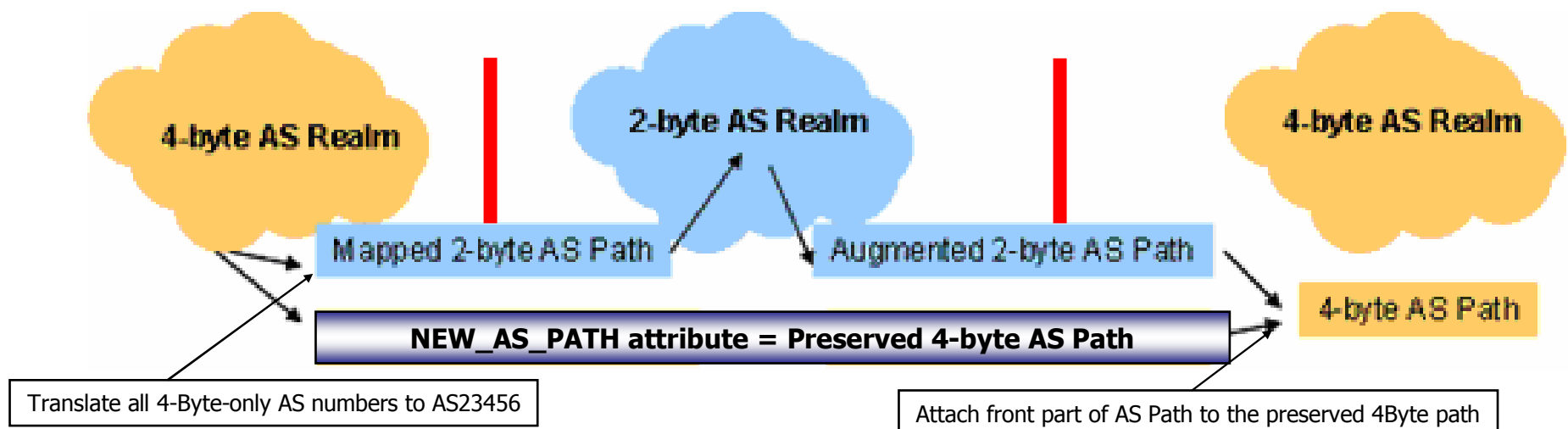
AS Path Semantics in BGP

- It's a *path metric* where the length of the AS Path is used as in path selection
- It's a *loop detector* where the presence of your own AS in a PATH is an indicator of a distance-vector "I'm-going-to-loop-to-infinity-unless-you-stop-me" loop

You don't have to have an **entirely** accurate AS Path – but at a minimum you do have to have path-metric and loop-detecting properties for BGP to function correctly

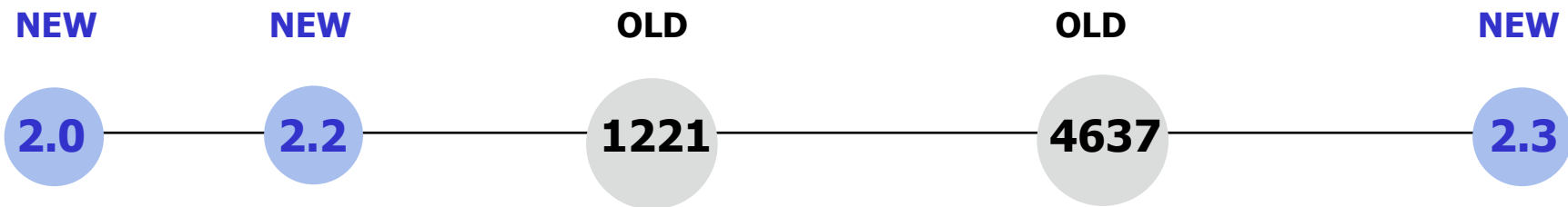
4-Byte AS Transition

- Think about this space as a set of NEW / OLD boundaries
- Define the NEW / OLD and the OLD / NEW transitions
- Preserve all BGP information at the transition interfaces
 - **Translate** 4-Byte AS Path information into a 2-Byte representation
 - **Tunnel** 4-Byte AS Path information through 2-Byte AS domain as an update attribute





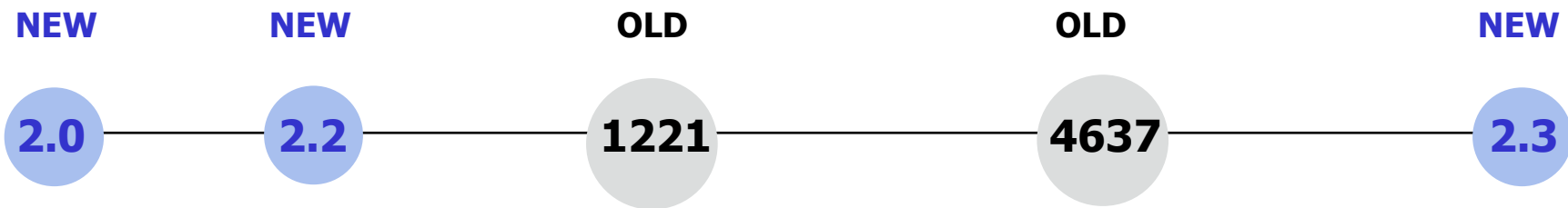
4-Byte AS Example



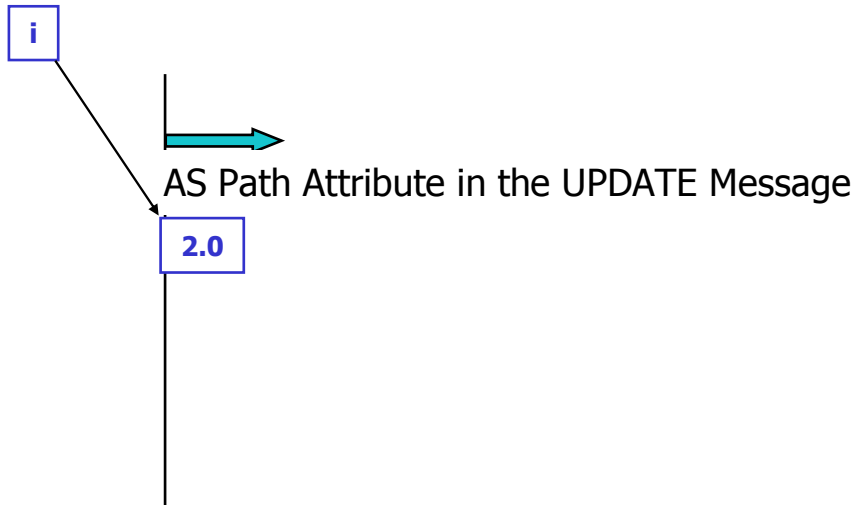
AS Path in the RIB



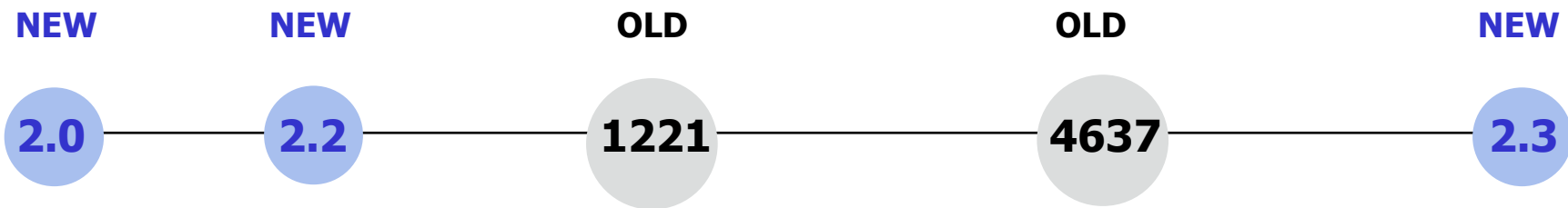
4-Byte AS Example



AS Path in the RIB



4-Byte AS Example



AS Path in the RIB

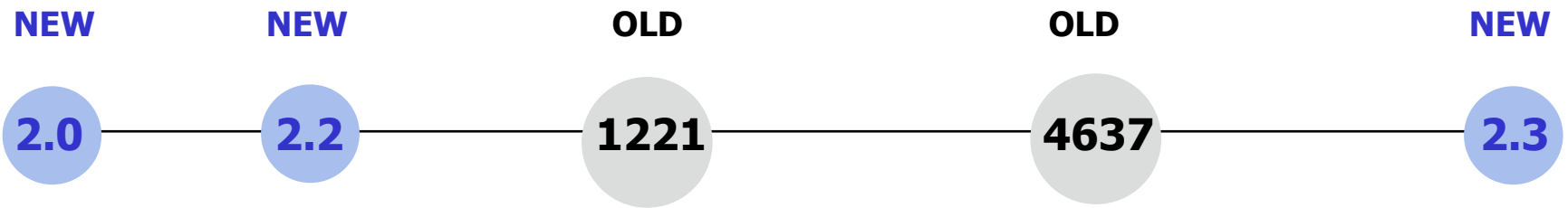


AS Path Attribute in the UPDATE Message

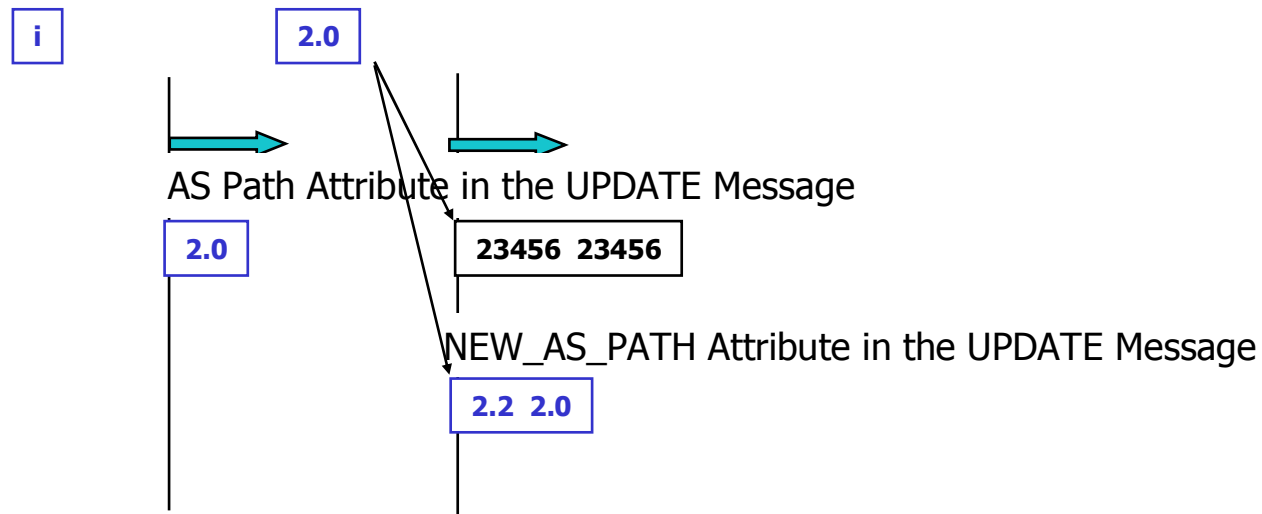




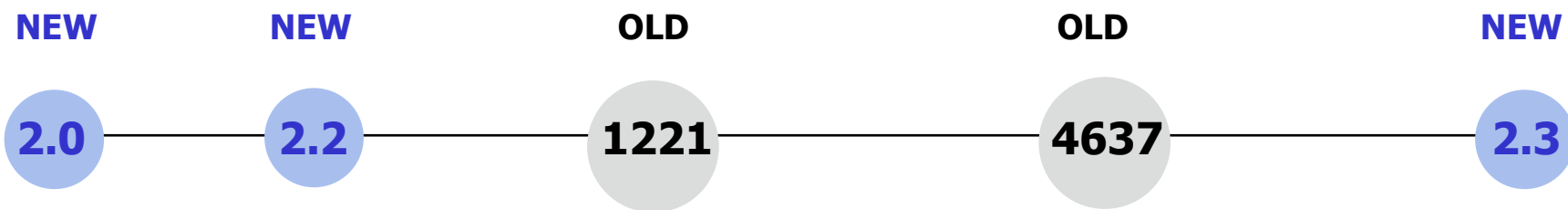
4-Byte AS Example



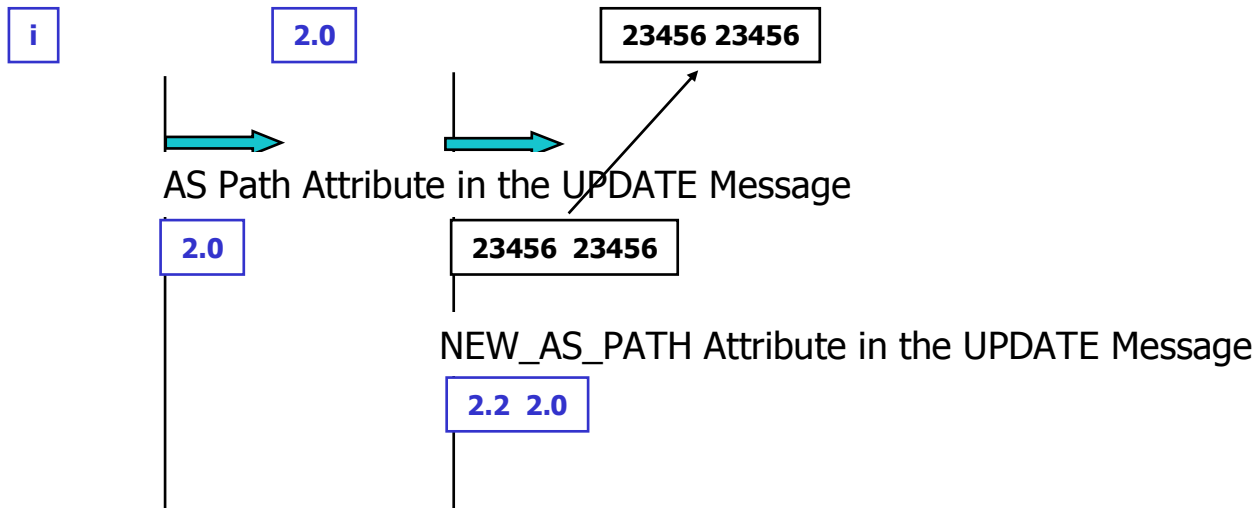
AS Path in the RIB



4-Byte AS Example

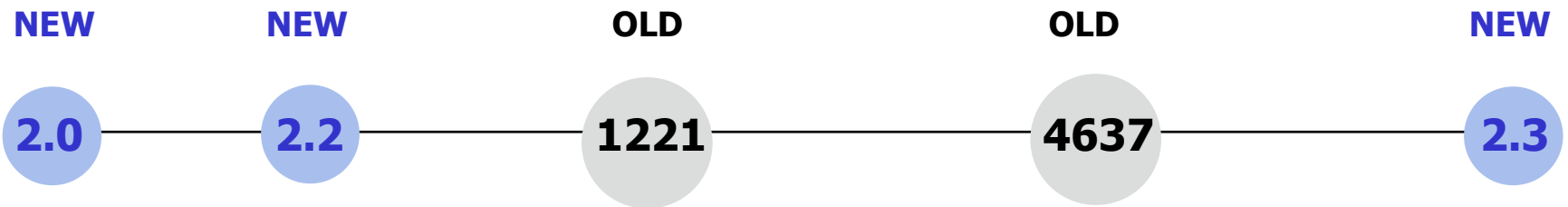


AS Path in the RIB

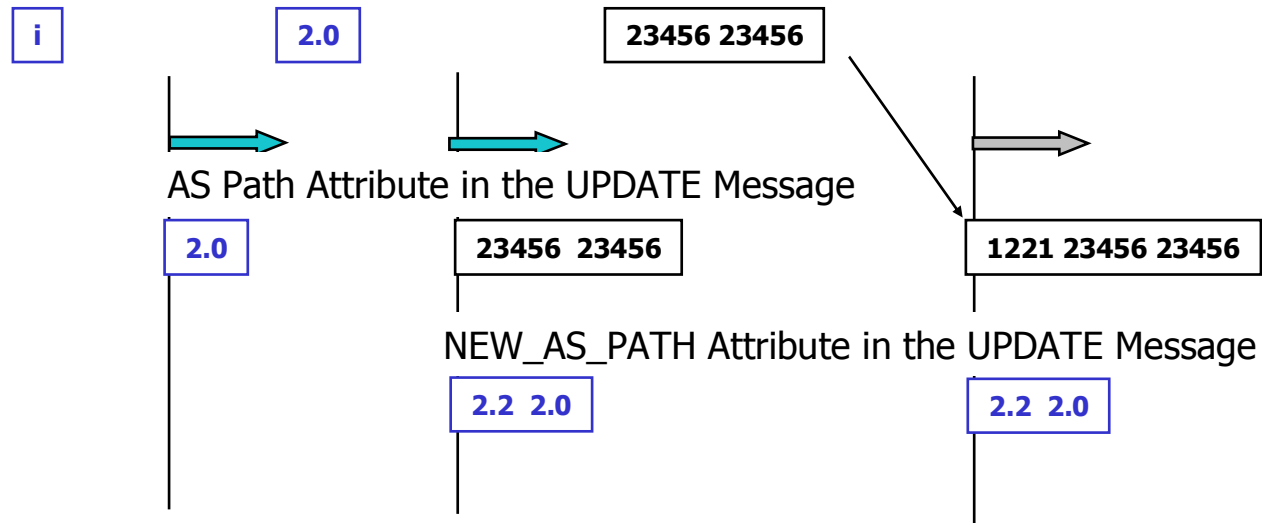




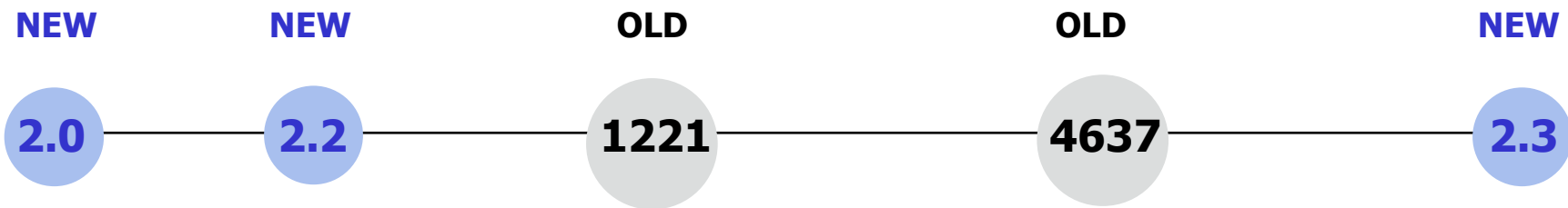
4-Byte AS Example



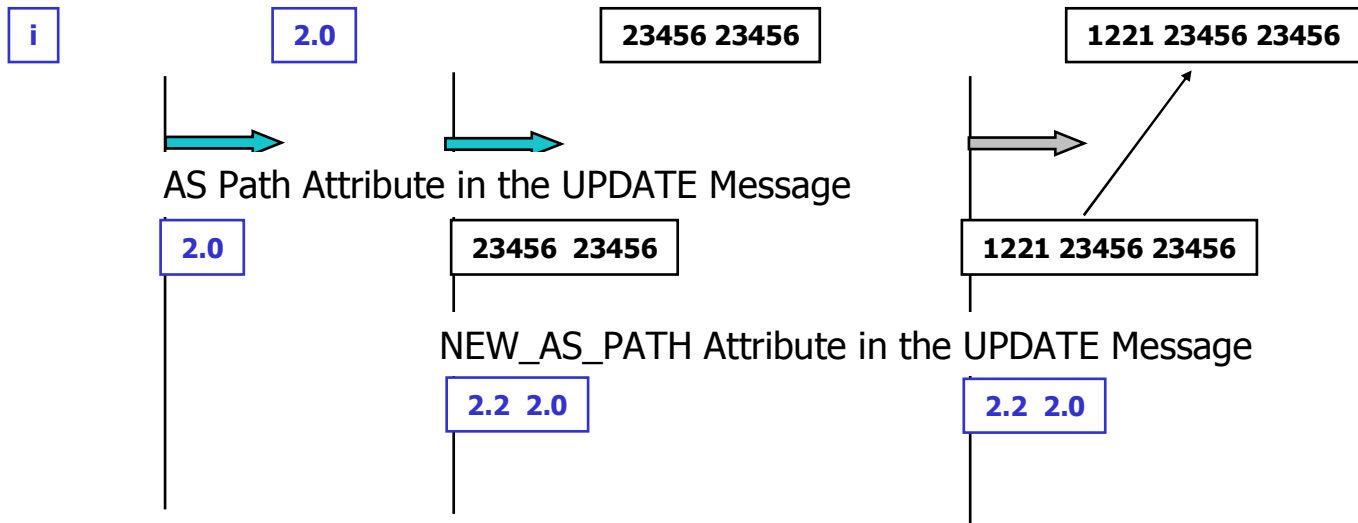
AS Path in the RIB



4-Byte AS Example

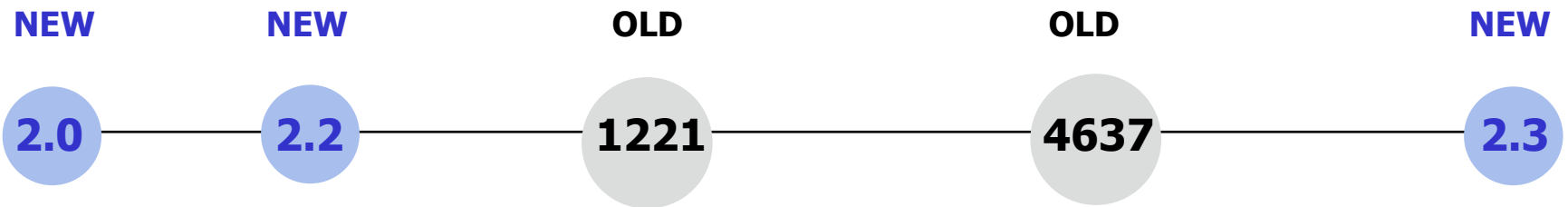


AS Path in the RIB

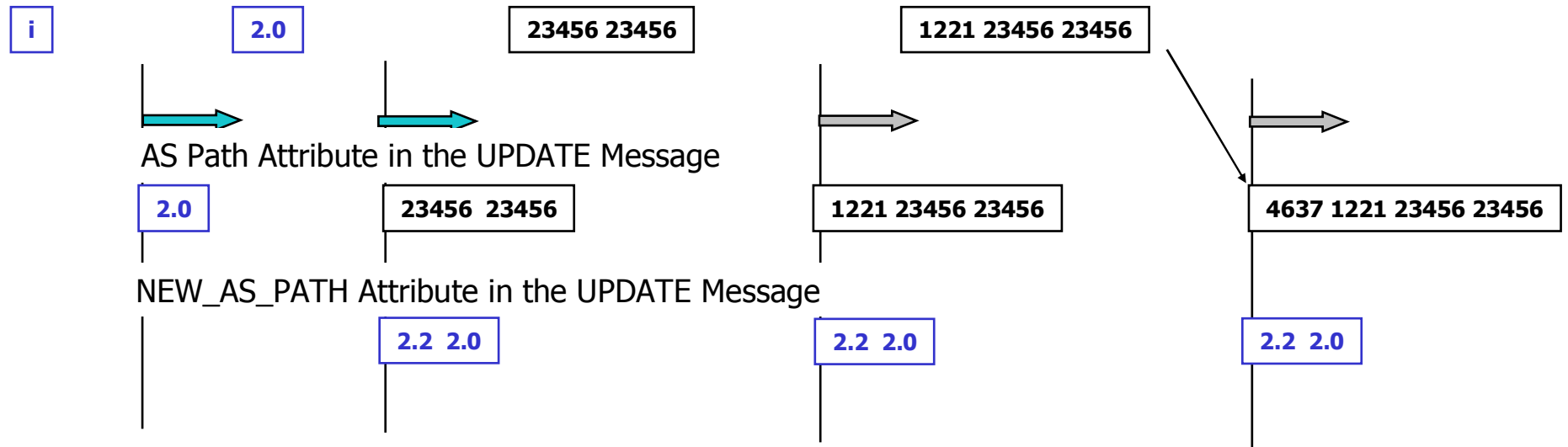




4-Byte AS Example

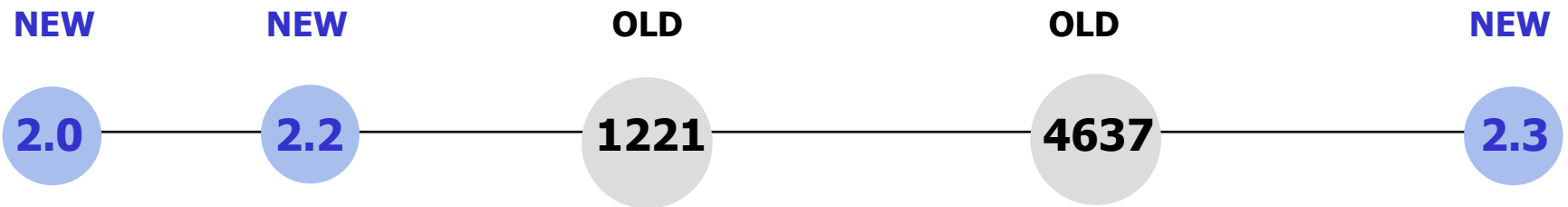


AS Path in the RIB

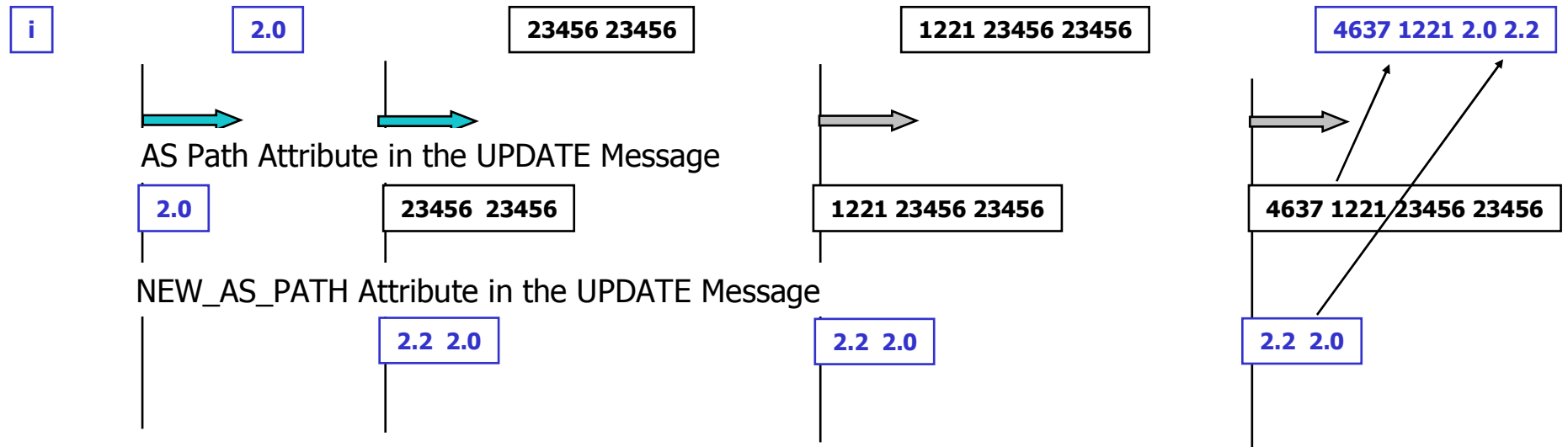


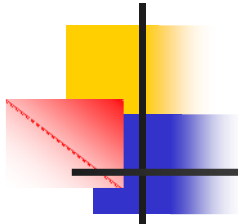


4-Byte AS Example

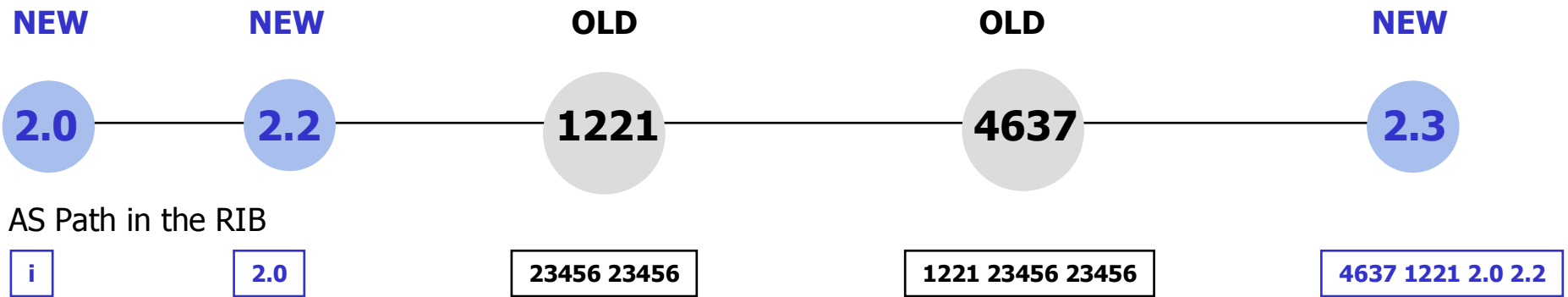


AS Path in the RIB





4-Byte AS Example





What changes with 4-Byte ASs?

- If you are an “old” BGP speaker then what should you look out for?



NEW_AS_PATH Attribute

- BGP speakers in 2-Byte AS domains should support **NEW_AS_PATH** as a transitive optional attribute in UPDATE messages
 - because that's where the 4-byte path is hiding
 - That's a "SHOULD" not a "MUST", by the way
 - Its better if you do, but nothing fatally breaks if you don't
 - Mixed 2 / 4 Byte loops will get detected in the 2-Byte world as a fallback

Default BGP configurations will do the right thing here



NEW_AGGREGATOR Attribute

- BGP speakers in 2-Byte AS domains should support **NEW_AGGREGATOR** as a transitive optional attribute in UPDATE messages
 - because that's where the 4-byte Aggregator AS is hiding
 - That's a "SHOULD" not a "MUST", by the way
 - Its better if you do, but nothing fatally breaks if you don't

Default BGP configurations should do the right thing here



AS 23456

- **AS 23456** is going to appear in many 2-Byte AS paths – both origin and transit

This is not an error – it's a 2-Byte token holder for a 4-Byte AS number

The logo consists of a vertical black line on the left, a horizontal black line at the bottom, and a yellow square in the top-left corner. A red square is partially visible behind the yellow one, and a blue square is partially visible behind the horizontal line.

Netflow

- Netflow analyzers may need to be reviewed
 - Netflow version 9 supports 4-byte AS numbers
 - But may not report the 4-Byte ASN unless the netflow collector is a 4-byte BGP
 - Does your analyzer support 4-Byte AS numbers?
 - Netflow version 8 and earlier are 2-Byte AS constrained
 - Which implies that you'll be seeing AS 23456 more than you may want!



BGP Communities and Extended Communities

- If you want to explicitly signal to a 4-Byte AS using communities in BGP then you will need to explicitly signal the 4-Byte AS using BGP Extended Communities
 - Attempting to use AS23456 in this context will have unintended consequences!

See:

- RFC4630
- draft-rekhter-as4octet-ext-community-01.txt



Memory

- BGP memory requirements will increase
 - 4-Byte BGP speakers will need twice the memory used to hold AS paths¹
 - 2-Byte BGP speakers will need up to three times the memory used to hold AS paths plus NEW_AS_PATH extended community attribute²

1 - Not "twice the memory" but "twice the memory used for AS Path storage"

2 - Not "three times the memory", but "three times the memory used for AS Path Storage"



Bandwidth

- BGP bandwidth requirements will increase
 - 4-Byte BGP speakers will need twice the size used to carry AS paths
 - 2-Byte BGP speakers will need up to three times the size used to carry AS paths (factoring in the NEW_AS_PATH attribute)



Performance

- 4-Byte to 2-Byte BGP session startup may be considerably slower
 - The 4-Byte speaker will need to compress all the AS Paths into their 2-Byte equivalent prior to generating updates
 - (assuming that the 2-Byte Paths for Update messages are generated on demand)
 - This may take some time to compute for some 35,000 distinct AS Paths



Performance

- BGP convergence times may increase in some cases
 - Any instance of 2-Byte BGP world destruction of the tunnelled NEW_AS_PATH attribute implies extended times on loop detection in order to fully complete prefix withdrawal
 - Its not that the withdrawal will loop forever, its that the loop will take additional AS hops before it is detected in the 2-Byte realm
 - The time to complete the withdrawal of a route may be extended



Proxy Aggregation

- If you proxy aggregate in the 2-Byte world then make sure that the aggregate is strictly larger than the components
 - Or loop detection may be harder
 - As the AS Set object generated in the 2-Byte world as a result of this proxy aggregation is not cleanly translatable into the 4-Byte world, so 4-Byte information is lost
- But proxy aggregation is not a common occurrence in today's BGP environment



Mixed environments

- No dynamic capability for 2/4-Byte ASN mode shift
 - You cannot flick from “2-Byte OLD” to “4-Byte NEW” mode within an active BGP session
 - You need to clear the session and then perform a clean start to trigger the initial capability exchange



Transition within an AS

- In a complex iBGP AS that wants to transition to using a 4-Byte “home” AS then you are going to have to think about the transition VERY carefully
 - You can undertake this transition one router at a time, but care and attention are required



Notation Confusion

- We have not (yet) converged to a uniform way of describing 4-Byte AS Numbers:

Numerics

101, 65637

Dotted Short Ints

0.101, 1.101

Dotted Short Ints+

101, 1.101

See [draft-michaelson-4byte-as-representation-02.txt](#)



Operational Support Systems

What happens when you have a customer / transit / peer with a 4-Byte AS Number?

- What's in the route registries and what your customers tell you about their AS and what's in your OSS and your routing system will differ:
 - E.g.: **AS 1.2** needs to be auto-translated into **AS 23456** in a number of places, including in your OSS
 - Your BGP routers may need to peer with **AS 23456**, transit across **AS 23456**, and have multiple customers on **AS 23456** at the same time, while also understanding that these refer to different external parties
 - Your OSS might get terminally confused!



Related Systems

The following systems will need to be revised:

- Internet Route Registries and RPSL
- WHOIS databases, WHOIS query syntax and WHOIS report formats
- Protocol, log and dump analysers
- And anything else that wants to manipulate AS numbers, including your local support systems, scripts and databases



Changing BGP

- Known 4-Byte BGP implementations:
 - Quagga (patches to 0.99.6)
 - OpenBGPD (patches to 3.9, 4.0)
 - JUNOSe 4-1-0 and later
 - Redback



4-Byte AS Implementations

Patches to OpenBGPD and Quagga:

1. Convert BGP to be internal 4-Byte AS in all data structures
2. Alter parser and output routines to support the various notational forms of AS numbers
3. Alter OPEN processing to negotiate 4 Byte AS Capability with BGP Peer
4. Alter UPDATE processing changes to support 2-Byte peers
 - Generated updates include a generated NEW_AS_PATH attribute and a dynamically created 2-Byte AS_PATH (and AGGREGATOR changes)
 - Received updates need to merge NEW_AS_PATH with AS_PATH to form a stored 4-Byte AS PATH (and AGGREGATOR merges) and remove NEW_AS_PATH attribute



4 Byte AS Testing

- Tests have been undertaken using closed BGP networks, and over the public Internet
- Tests of 2-Byte/4-Byte transition boundaries in various permutations of transits and loops
- Current announcement of 203.10.62.0/24 originating from AS 2.2 to assist others in local testing of 4-Byte BGP



The Route-Views View

```
route-views.oregon-ix.net>show ip bgp 203.10.62.0/24
BGP routing table entry for 203.10.62.0/24, version 177310093
Paths: (43 available, best #39, table Default-IP-Routing-Table)
Not advertised to any peer
 3277 3216 3549 4637 1221 23456
   194.85.4.55 from 194.85.4.55 (194.85.4.16)
     Origin IGP, localpref 100, valid, external
     Community: 3216:3000 3216:3004 3277:3216 3549:2141 3549:30840
 7500 2497 4637 1221 23456
   202.249.2.86 from 202.249.2.86 (203.178.133.115)
     Origin IGP, localpref 100, valid, external
 2493 3602 812 812 4637 1221 23456
   206.186.255.223 from 206.186.255.223 (206.186.255.223)
     Origin IGP, localpref 100, valid, external
 2905 701 1239 4637 4637 4637 4637 4637 4637 1221 23456
   196.7.106.245 from 196.7.106.245 (196.7.106.245)
     Origin IGP, metric 0, localpref 100, valid, external
...

```



4-Byte Path Reconstruction

```
srv0# bgpctl show rib 203.10.62.0/24
```

```
flags: * = Valid, > = Selected, I = via IBGP, A = Announced
```

```
origin: i = IGP, e = EGP, ? = Incomplete
```

flags	destination	gateway	lpref	med	aspath	origin
*>	203.10.62.0/24	147.28.0.1	100	0	0.3130 0.1239 0.4637	0.4637 0.4637 0.4637
					0.4637 0.4637 0.1221	
					1.202 i	

Experiment performed on January 11 2007, with the assistance of Randy Bush and George Michaelson, using OpenBGPD 3.9 with 4Byte AS support patches as the origin and the observer points.



Conclusions

- Deployed BGP appears to be entirely capable of supporting incremental 4-Byte AS deployments
 - No, the Internet is probably not going to crash and burn just because of this change in BGP!
- BUT: Will your OSS do the right thing when you need to use 4 Byte AS numbers?
 - What happens if you have 2 or more eBGP customers with 4 byte AS numbers?
 - What happens if you have 2 or more transits and/or peers using 4 byte AS numbers?



Resources

- IETF Specification
 - [draft-ietf-idr-as4bytes-12.txt](#)
- OpenBGPD patches
 - [http://www.potaroo.net/tools/bgpd](#)
- Quagga patches
 - [http://quagga.ncc.eurodata.de/](#)



Thank You

Questions?