BGP Traffic Engineering

Andy Davidson
CTO @ Asteroid
Director @ LONAP, IXLeeds, Euro-IX

andy@nosignal.org
Bangkok Peering Forum 2.0
May 2017

Why do Traffic Engineering?

- Manage capacity demand
- Ensure service quality
- Recover from Failures
- Manage circuit costs
- Handle traffic growth

James Cridland http://www.flickr.com/photos/jamescridland/

Complexity

Life starts simple, "send to peers if possible, then transit providers"

But what about when network grows?

What about when your traffic grows?

Or you add more cities/POPs/IXPs?

ey Hussain Bisson - http://www.flickr.com/people/maisonbisson/

Real examples

- Circuits with **cost difference** >$100/Mbit
- Regional networks - **poor local peering**
- Circuit failure causing **congestion**
- Changing **customer demand/**behaviour
  - Increased quality expectation
  - New high bandwidth services such as video

Internal network TE

- **Simple** compared with Interdomain TE

- You administrate both sides
  - You know the **price** of all paths
  - The IGP knows the **capacity** of all paths
  - IGP protocols let you map price, capacity to shape routing using **cost**.

Inter-domain TE

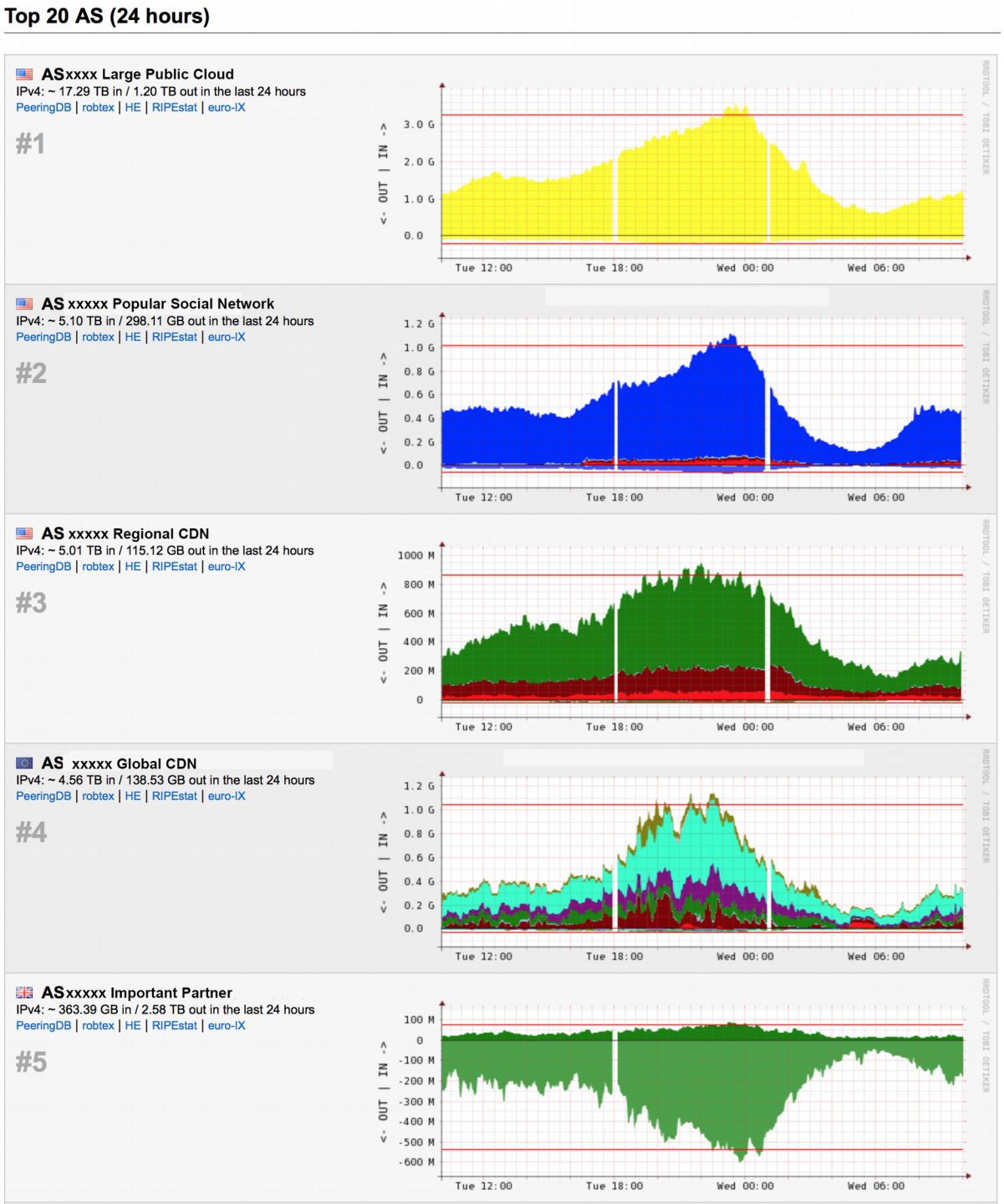
- You do NOT control both sides
  - Path vector protocols hide metric, capacity, cost
  - Simplicity of BGP protocol imposes **limitations**
  - **Volume of traffic** matters, not # of routes
- However, large volume of traffic is usually with a small number of other ASNs

You need **data**

AS-Stats
Manuel Kasper
https://neon1.net/as-stats/

.. But more on this later

14/05/2017



Netflow

- Export information about packets routed through your network
- Normally sampled
- Sent to a collector over UDP
- A variety of commercial and open-source tools sort and display these flow records.

Different Flow protocols

- Netflow – Designed by Cisco in ‘90s, published as a standard (v9 is RFC3954 and supports IPv6)
- IPFIX (RFC5101) Based on Netflow 9, 2008
- sFlow – Nice protocol but incompatible with Netflow, typically implemented on L2 switch.
- Jflow – Essentially Netflow on Junipers

Enabling Netflow (example)

`ip route-cache flow`

Enables Netflow on an Interface

`ip flow-export version X origin-as`

Defines Netflow options

`ip flow-export destination <ip> <port>`

Defines the collector address

`ip flow-export source loopback0`

For consistent source IP addressing

6500/7600 sup720 Netflow

```
mls netflow interface
mls flow ip interface-full
mls flow ipv6 interface-full
mls nde sender
ip flow-capture mac-addresses
ip flow-export version 9 origin-as
ip flow-export destination 192.0.2.100 5500 vrf vrf-netflow
ip flow-top-talkers

interface GigabitEthernet1/1
 ip flow ingress
```

Order that you enter configuration matters.

With special thanks to Nick Hilliard of INEX for this config

XR Flexible Netflow

```
flow exporter-map fem-default
 version v9
  options interface-table timeout 300
  options sampler-table timeout 300
 !
 transport udp 5500
 source Loopback0
 destination 192.0.2.100

flow monitor-map fmm-ipv4
 record ipv4
 exporter fem-default
 cache entries 1000000

sampler-map sm-flow-default
 random 1 out-of 100

interface TenGigE0/0/2/2
 flow ipv4 monitor fmm-ipv4 sampler sm-flow-default ingress

router bgp 65533
 address-family ipv4 unicast
  bgp attribute-download
```

With special thanks to Nick Hilliard of INEX for this config

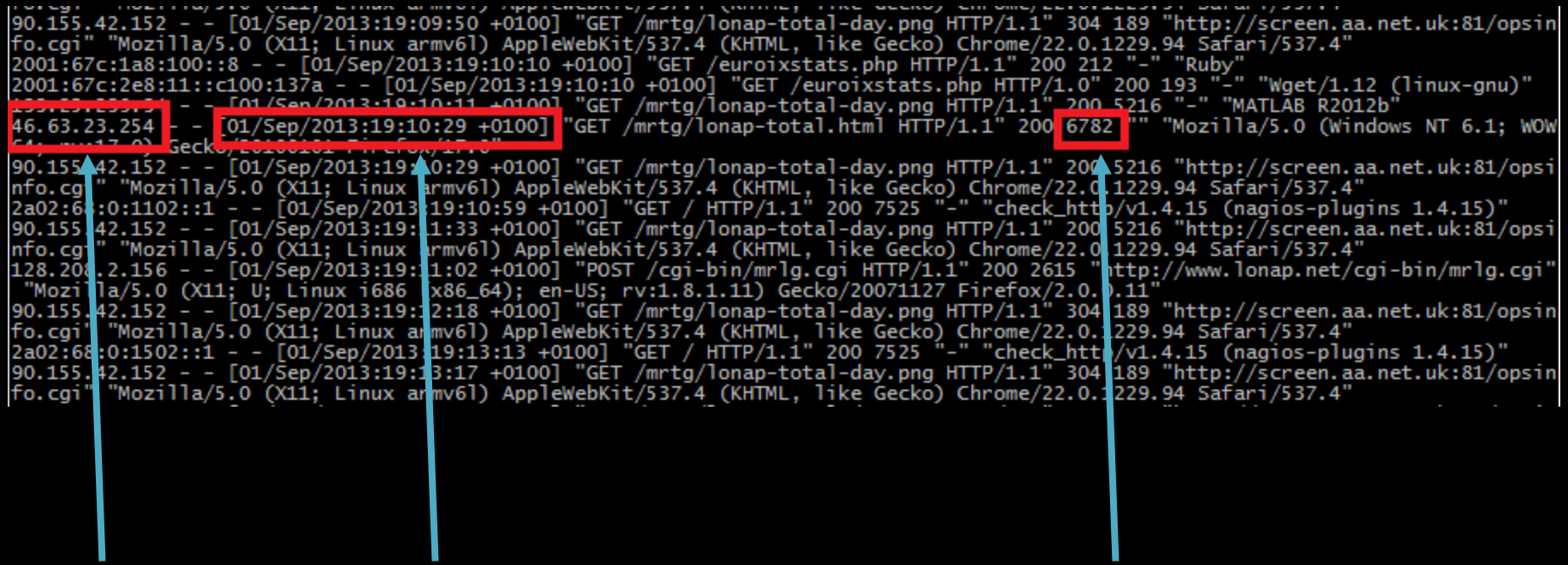
Brocade Netflow

```
sflow enable
sflow source loopback 1 8888
sflow destination x.x.x.x
sflow polling-interval 30

interface Ethernet1/1 to 1/4
    sflow forwarding
```

Other ways to get data

- Log file analysis
  - Useful before you have a network, for working out the benefit of building a network/peering.
  - Best for ‘single service’ networks
    - DNS providers have DNS logs with time & IP
    - Web providers have web logs with time & IP
    - Hosted email providers have mail logs…


90.155.42.152 - - [01/Sep/2013:19:09:50 +0100] "GET /mrtg/lonap-total-day.png HTTP/1.1" 304 189 "http://screen.aa.net.uk:81/opsinfo.cgi" "Mozilla/5.0 (X11; Linux armv6l) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4"
2001:67c:1a8:100::8 - - [01/Sep/2013:19:10:10 +0100] "GET /euroixstats.php HTTP/1.1" 200 212 "-" "Ruby"
2001:67c:2e8:11::c100:137a - - [01/Sep/2013:19:10:10 +0100] "GET /euroixstats.php HTTP/1.0" 200 193 "-" "Wget/1.12 (linux-gnu)"
46.63.23.254 - - [01/Sep/2013:19:10:29 +0100] "GET /mrtg/lonap-total.html HTTP/1.1" 200 6782 "" "Mozilla/5.0 (Windows NT 6.1; WOW
90.155.42.152 - - [01/Sep/2013:19:10:29 +0100] "GET /mrtg/lonap-total-day.png HTTP/1.1" 200 5216 "http://screen.aa.net.uk:81/opsinfo.cgi" "Mozilla/5.0 (X11; Linux armv6l) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4"
2a02:68:0:1102::1 - - [01/Sep/2013:19:10:59 +0100] "GET / HTTP/1.1" 200 7525 "-" "check_http/v1.4.15 (nagios-plugins 1.4.15)"
90.155.42.152 - - [01/Sep/2013:19:11:33 +0100] "GET /mrtg/lonap-total-day.png HTTP/1.1" 200 5216 "http://screen.aa.net.uk:81/opsinfo.cgi" "Mozilla/5.0 (X11; Linux armv6l) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4"
128.208.2.156 - - [01/Sep/2013:19:11:02 +0100] "POST /cgi-bin/mrlg.cgi HTTP/1.1" 200 2615 "http://www.lonap.net/cgi-bin/mrlg.cgi" "Mozilla/5.0 (X11; U; Linux i686 (x86_64); en-US; rv:1.8.1.11) Gecko/20071127 Firefox/2.0.0.11"
90.155.42.152 - - [01/Sep/2013:19:12:18 +0100] "GET /mrtg/lonap-total-day.png HTTP/1.1" 304 189 "http://screen.aa.net.uk:81/opsinfo.cgi" "Mozilla/5.0 (X11; Linux armv6l) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4"
2a02:68:0:1502::1 - - [01/Sep/2013:19:13:13 +0100] "GET / HTTP/1.1" 200 7525 "-" "check_http/v1.4.15 (nagios-plugins 1.4.15)"
90.155.42.152 - - [01/Sep/2013:19:13:17 +0100] "GET /mrtg/lonap-total-day.png HTTP/1.1" 304 189 "http://screen.aa.net.uk:81/opsinfo.cgi" "Mozilla/5.0 (X11; Linux armv6l) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4"
IP Address
Time and date
Amount of Traffic

Other ways to get “data”

Your instinct is better than you think?

- Content networks will talk to eyeballs
- Eyeball networks will talk to content
- Confirm with top talkers, etc.
- But you should use Netflow!
  - Early “quick wins” may provide hard data
  - Hard data provides stronger business case

Data tells you

- Your traffic **direction**
  - Mainly inbound
  - Mainly outbound
  - Balanced
- Your **top traffic originators** or **destinations**

As-stats

- https://neon1.net/as-stats/

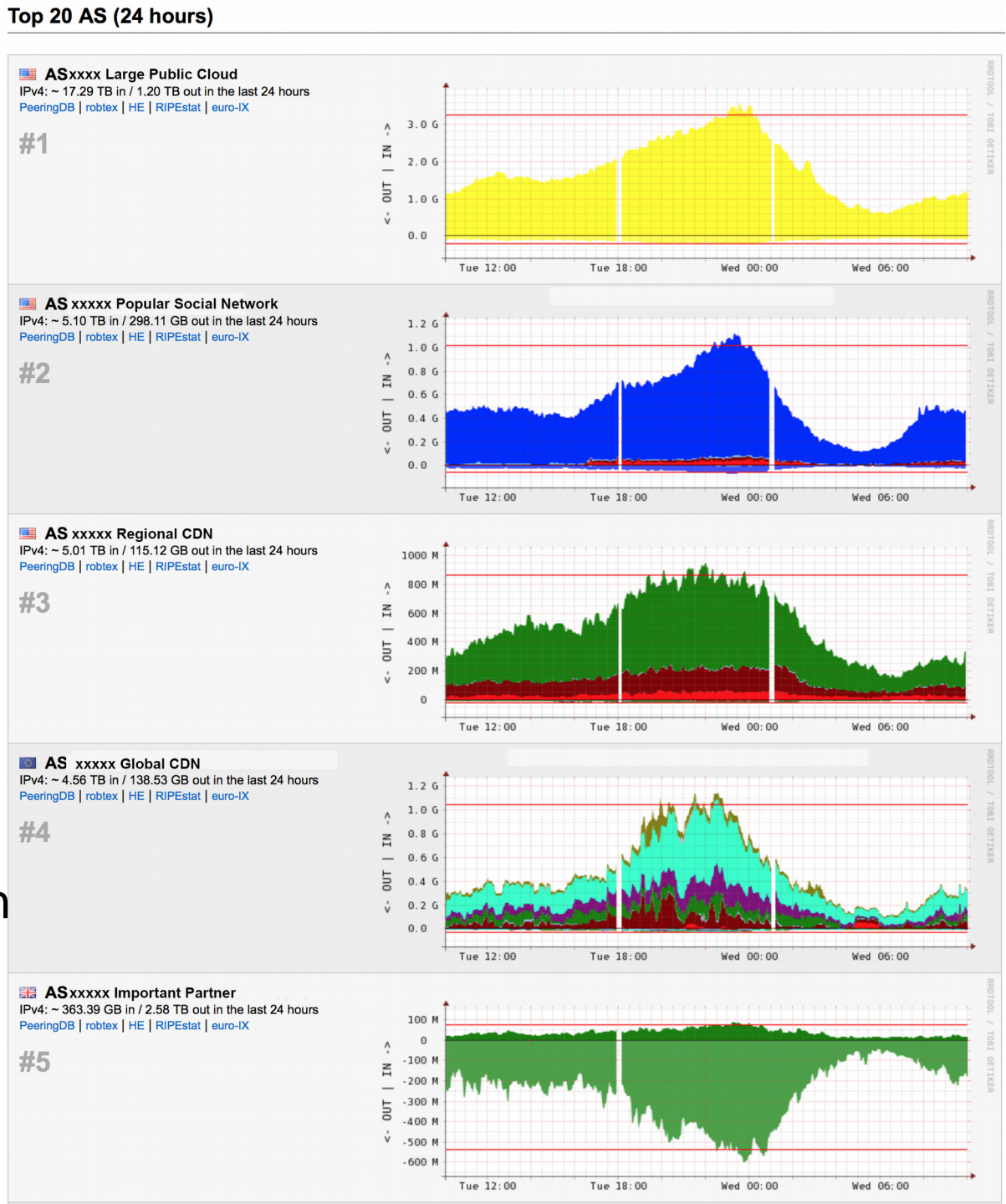
- Open source
- Quick to setup, simple to use
- Resource intensive

Who are my key peers?

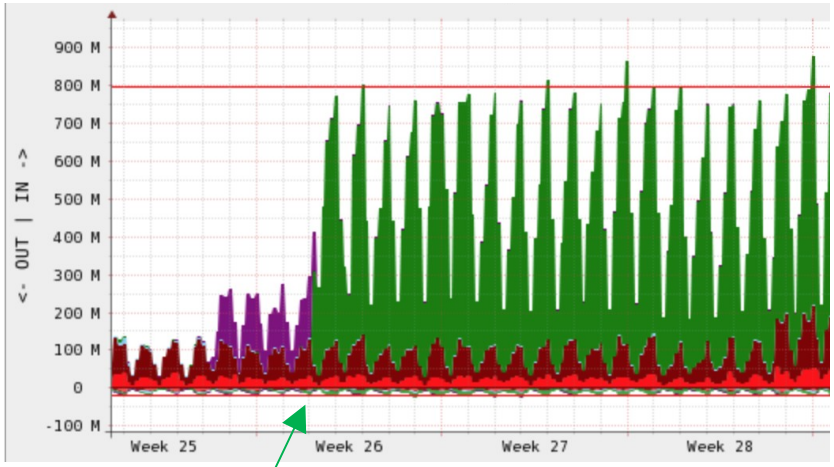
(or potential peers)

Top originators of traffic to me, top consumers of my content

Chart colour relates to an interface on the edge of my network

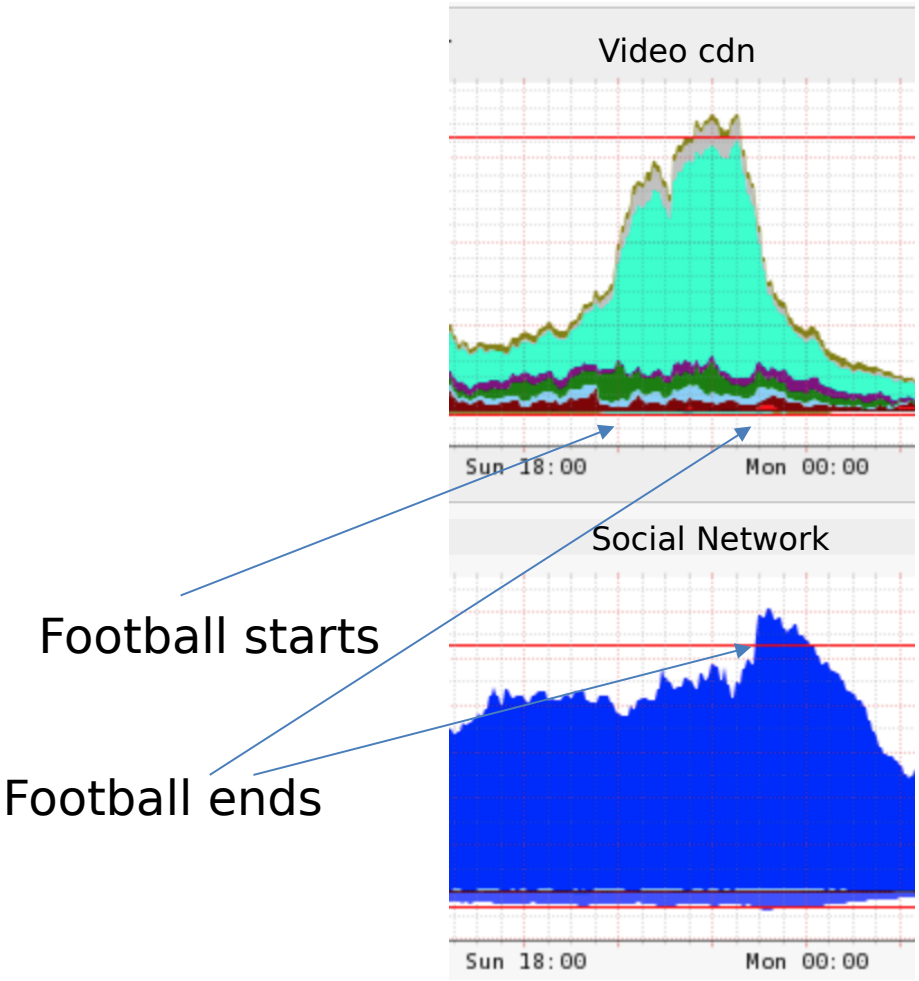


Historical data



New peering added, traffic growth!
= $$$$$ for my network!

Bonus please!

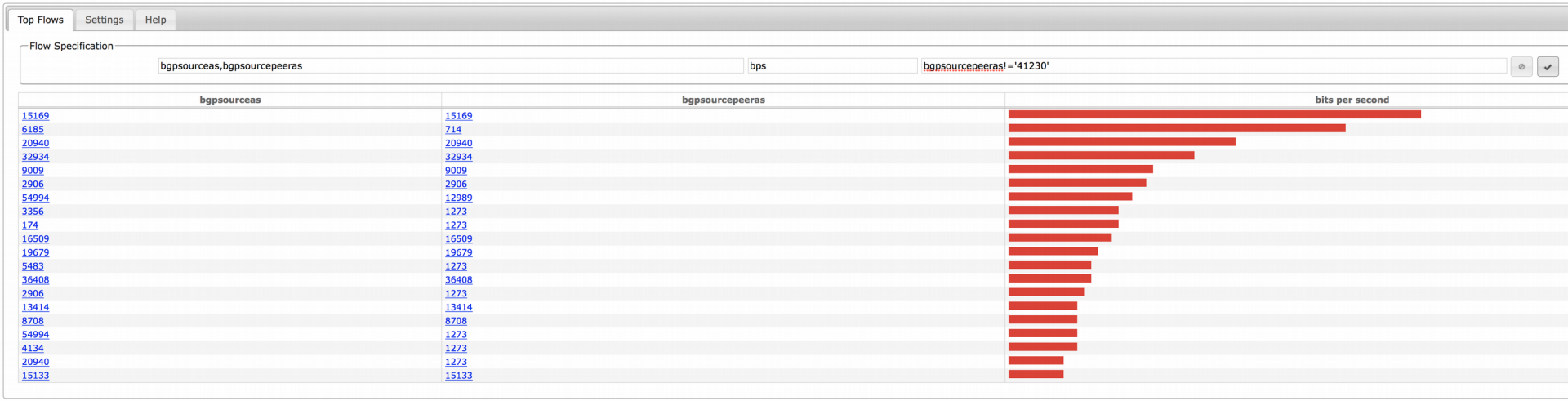

Video cdn
Sun 18:00
Mon 00:00
Social Network
Sun 18:00
Mon 00:00
Football starts
Football ends

Inmon sflow-rt

- http://www.sflow-rt.com/

- Produces realtime traffic graphs
- “What is the situation right now”
- Useful to check peering config has taken effect
- Also useful in abuse mitigation

Realtime aggregate data



Realtime transit analysis

Flow Specification

bgpsourceas,bgpsourcepeeras	bps	bgpsourcepeeras='1273'

bgpsourceas	bgpsourcepeeras	bits per second
3356	1273	
54994	1273	
4134	1273	
4837	1273	
5483	1273	
174	1273	
16625	1273	
4812	1273	
20940	1273	
9506	1273	
12876	1273	
9304	1273	
56046	1273	
32934	1273	
6185	1273	
41164	1273	
3320	1273	
8764	1273	
17816	1273	
17621	1273	

Export from Sflow-RT

- RESTful export into logstash/influxdb/grafana for historical data

- http://www.slideshare.net/pphaal/network-visibility-and-control-using-industry-standard-sflow-telemetry

pmacctd

- http://www.pmacct.net

- Open Source
- High performance, high scale, powerful
- Most flexible, most configuration required

- Collector > own reports

1) Configure a collector

```
sfacctd_port: 2100
sfacctd_as: sflow
sfacctd_renormalize: true
!
plugins: print[testprint]
!
aggregate[testprint]: in_iface, out_iface, proto, peer_src_ip, peer_dst_ip,
peer_dst_as, peer_src_as, src_as, dst_as
!
print_output_file[testprint]: /path/to/spool/blabla-$peer_src_ip-%Y%m%d-%H%M.txt
print_output[testprint]: csv
print_output_separator[testprint]: ;
print_refresh_time[testprint]: 60
print_history[testprint]: 1m
print_history_roundoff[testprint]: m
```

Use the ASN data from the router if it exists, no need to setup BGP flow export

2) Get a report

```
SRC_AS;DST_AS;PEER_SRC_AS;PEER_DST_AS;PEER_SRC_IP;PEER_DST_IP;IN_IFACE;OUT_
IFACE;PROTOCOL;PACKETS;BYTES
41230;224;41230;2603;x.x.x.253;x.x.x.246;3;4;tcp;2048;151552
41230;15169;41230;15169;x.x.x.253;x.x.x.246;3;4;tcp;10240;880640
41230;50247;41230;24724;x.x.x.253;x.x.x.246;3;4;tcp;2048;167936
41230;9269;41230;1273;x.x.x.253;x.x.x.237;3;3;tcp;2048;135168
41230;3356;41230;1273;x.x.x.253;x.x.x.237;3;3;tcp;32768;2375680
41230;209;41230;1273;x.x.x.253;x.x.x.237;3;3;udp;2048;2940928
20940;0;20940;0;x.x.x.253;x.x.x.246;3;4;tcp;43008;65458176
```

Red line represents a single flow with Google on behalf of a user

Keeping historical data

```
plugins: mysql[5mins], mysql[hourly]

sql_optimize_clauses: true
sql_dont_try_update: true
sql_multi_values: 1024000

sql_history_roundoff[5mins]: m
sql_history[5mins]: 5m
sql_refresh_time[5mins]: 300
sql_table[5mins]: acct_bgp_5mins

sql_history_roundoff[hourly]: h
sql_history[hourly]: 1h
sql_refresh_time[hourly]: 3600
sql_table[hourly]: acct_bgp_1hr

plugin_buffer_size: 10240
plugin_pipe_size: 1024000
aggregate: tag, src_as, dst_as, peer_src_as, peer_dst_as, peer_src_ip,
                                             peer_dst_ip, local_pref, as_path
```

Example by pmacct author Paolo Lucente

```
mysql> SELECT * FROM int_tm-20130803_1400 LIMIT 10;
+----------+-------------+-------------+-------------+------------------+-------+
| iface_in | peer_ip_src | peer_ip_dst | peer_dst_as | stamp_inserted   | bytes |
+----------+-------------+-------------+-------------+------------------+-------+
|      212 | 10.0.0.107  | 10.0.0.3    |       65000 | 03-08-2013 14:00 |   859 |
|      212 | 10.0.0.107  | 10.0.0.253  |       65001 | 03-08-2013 14:00 |  5358 |
|      212 | 10.0.0.107  | 10.0.0.234  |       65002 | 03-08-2013 14:00 |  6181 |
|      212 | 10.0.0.107  | 10.0.0.251  |       65003 | 03-08-2013 14:00 | 27002 |
|      205 | 10.0.0.107  | 10.0.0.233  |       65004 | 03-08-2013 14:00 |  1200 |
|      258 | 10.0.0.107  | 10.0.0.240  |       65005 | 03-08-2013 14:00 |   560 |
|      212 | 10.0.0.107  | 10.0.0.252  |       65006 | 03-08-2013 14:00 | 62682 |
|      212 | 10.0.0.107  | 10.0.0.234  |       65007 | 03-08-2013 14:00 |  3843 |
|      212 | 10.0.0.107  | 10.0.0.17   |       65008 | 03-08-2013 14:00 | 21074 |
|      205 | 10.0.0.107  | 10.0.0.254  |       65009 | 03-08-2013 14:00 |  2023 |
```

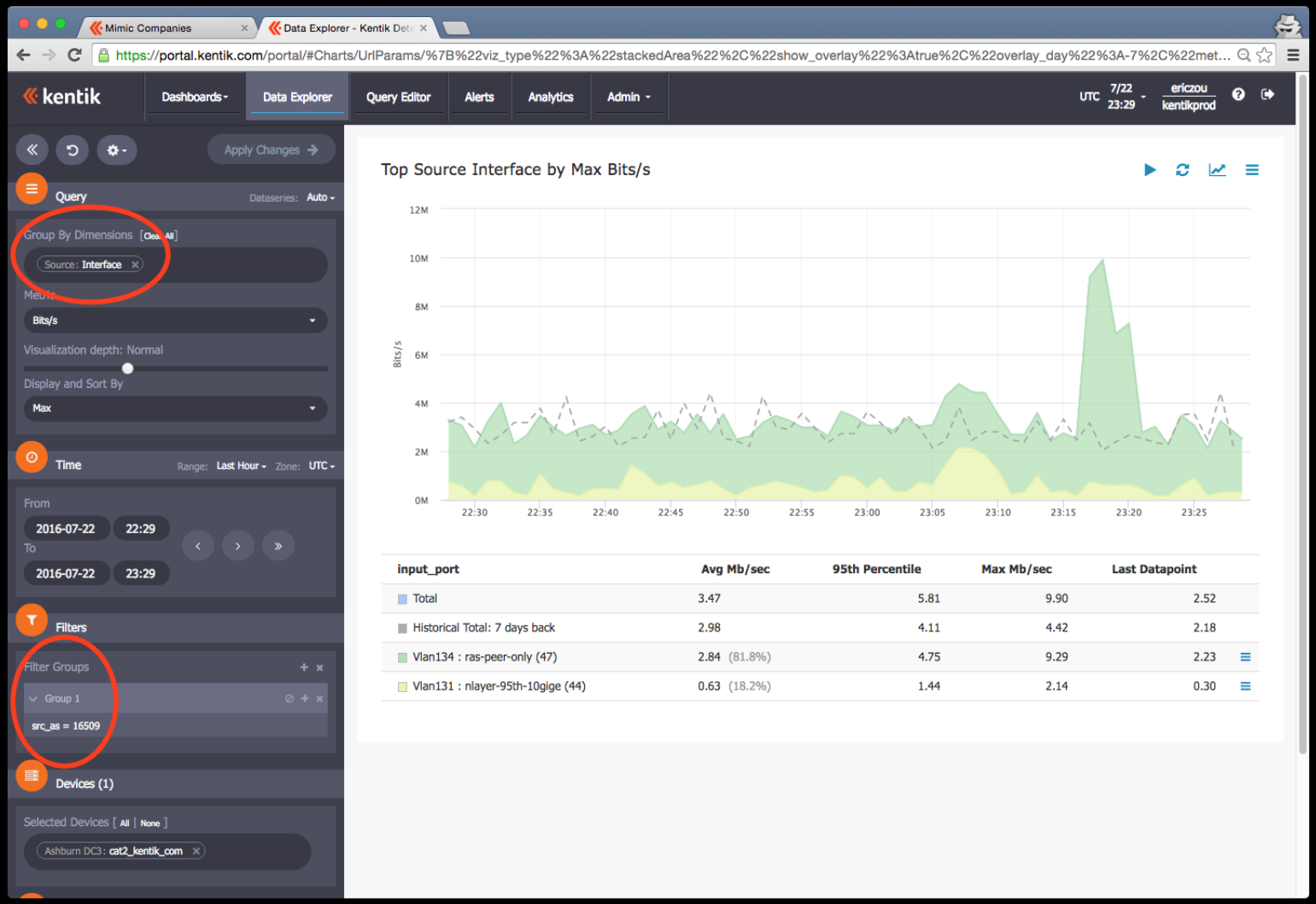
- Export into reports, web interface, spreadsheet
- Multiple back ends supported, including time series databases
- Very flexible approach but needs more setup time

Kentik

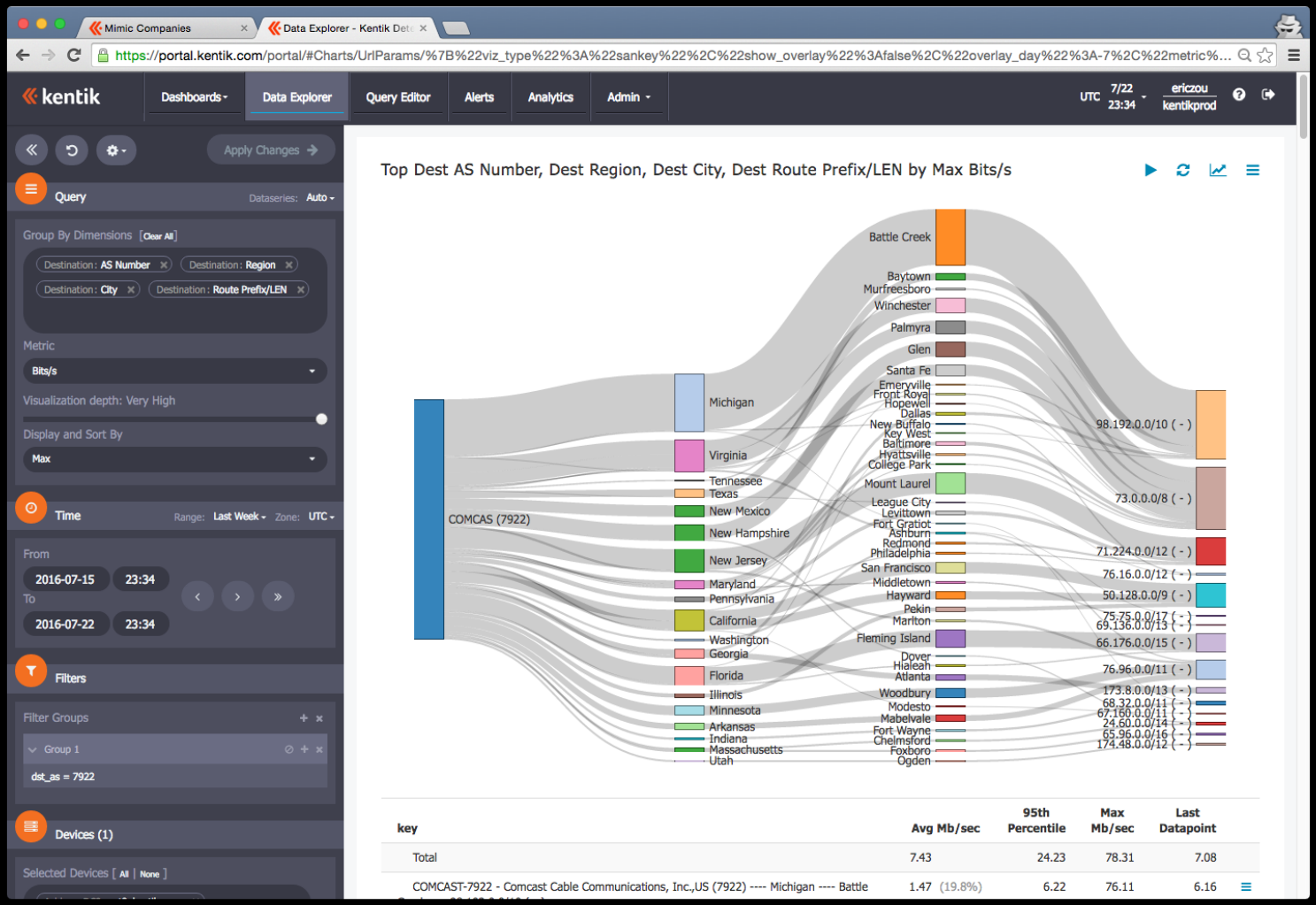
- https://www.kentik.com

- Hosted solution
- Zero configuration, zero equipment needed
- Point Netflow at their collector and reports follow

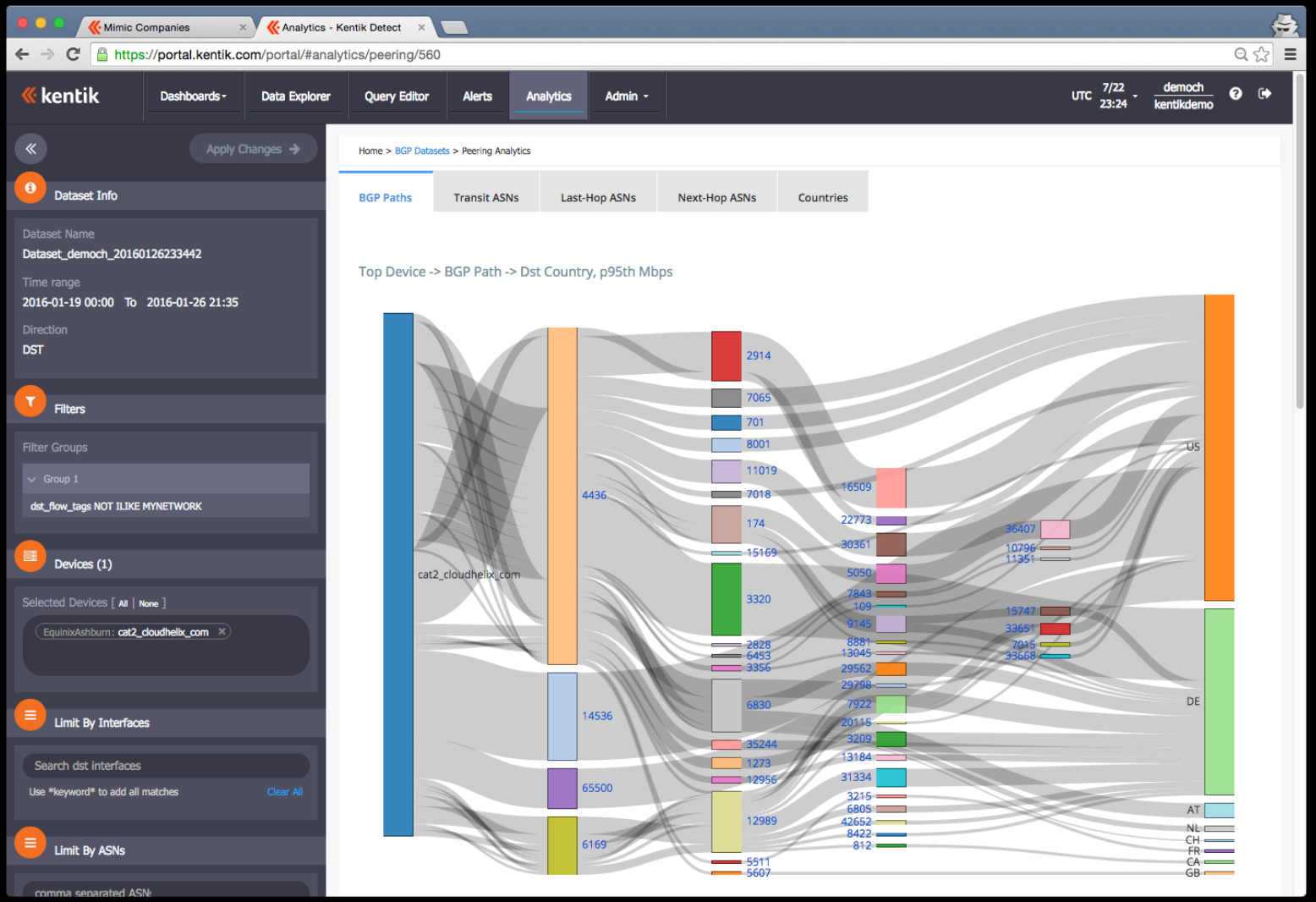
Traffic by Source ASN



Breakdown by region (US)



Top flows per interface



Outbound vs Inbound

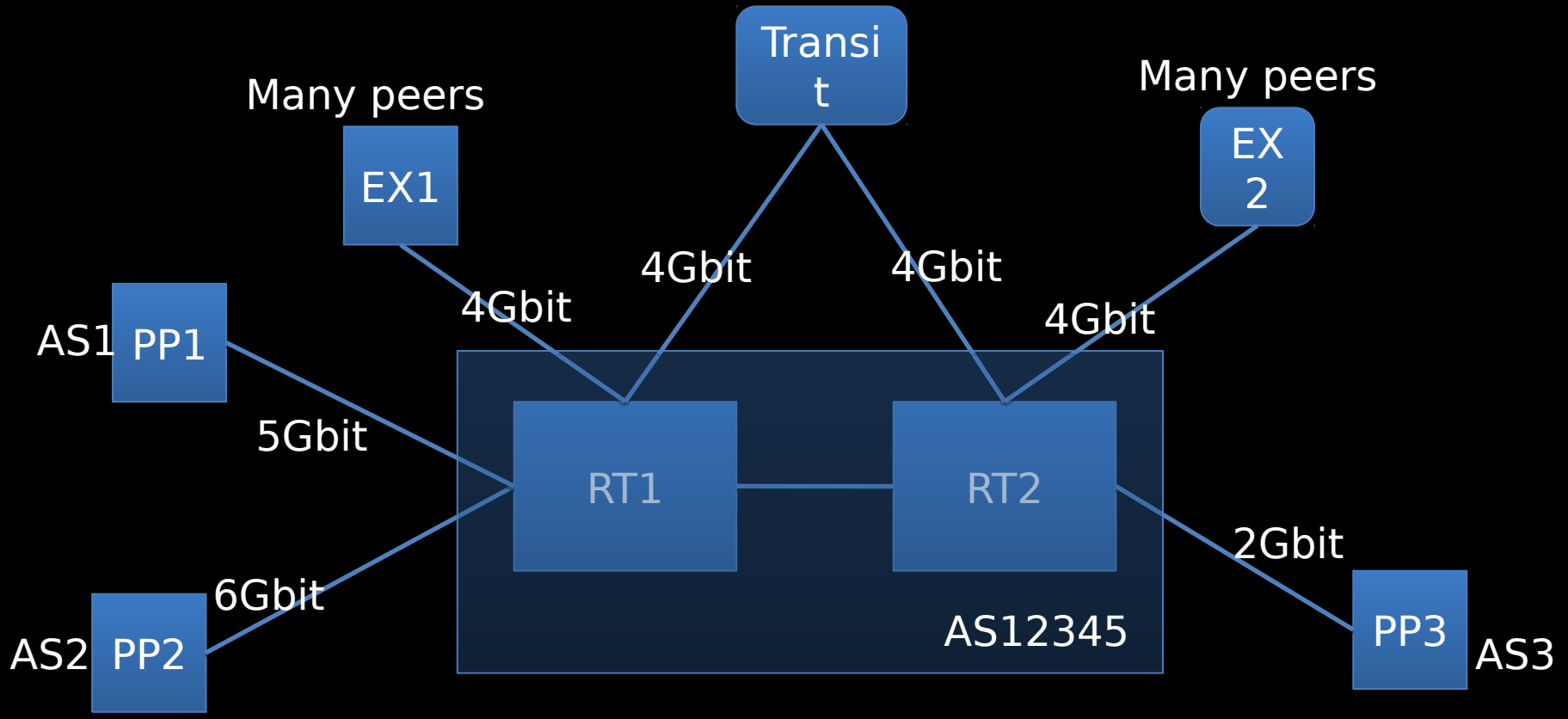
- Outbound heavy networks
  - Somewhat **easier life**
- Inbound heavy networks
  - You must **trick** the Best Path Selection methods of networks sending you traffic.
  - *Their* config change will move ***your*** traffic.

BGP Best Path Selection Algorithm

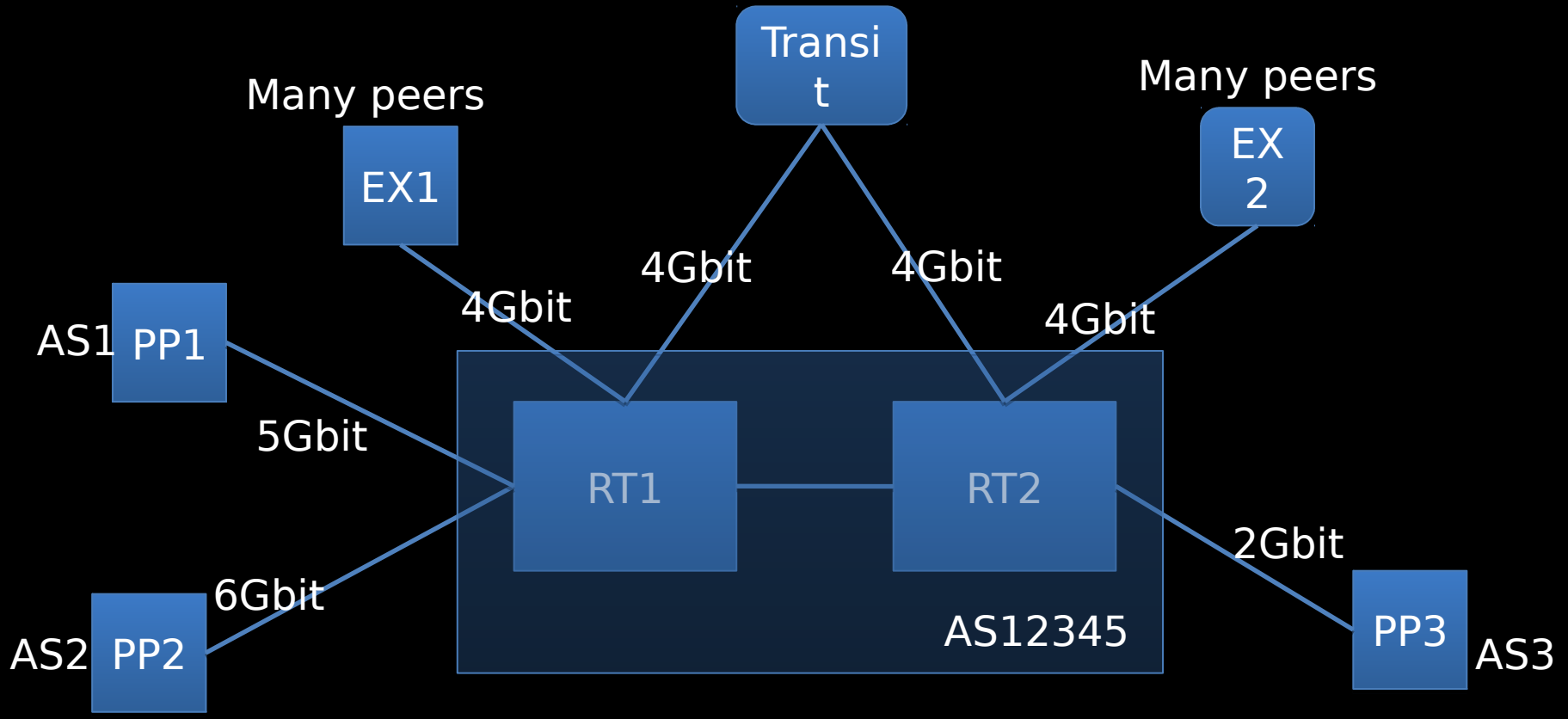
- Traffic engineering is about ‘tricking’ this process
- Affects traffic in outbound direction
  - Local Preference
  - AS PATH length
  - Lowest Origin Type
  - Lowest MED
  - Prefer eBGP paths
  - Lowest IGP Metric
  - Oldest route

Mainly outbound, single POP

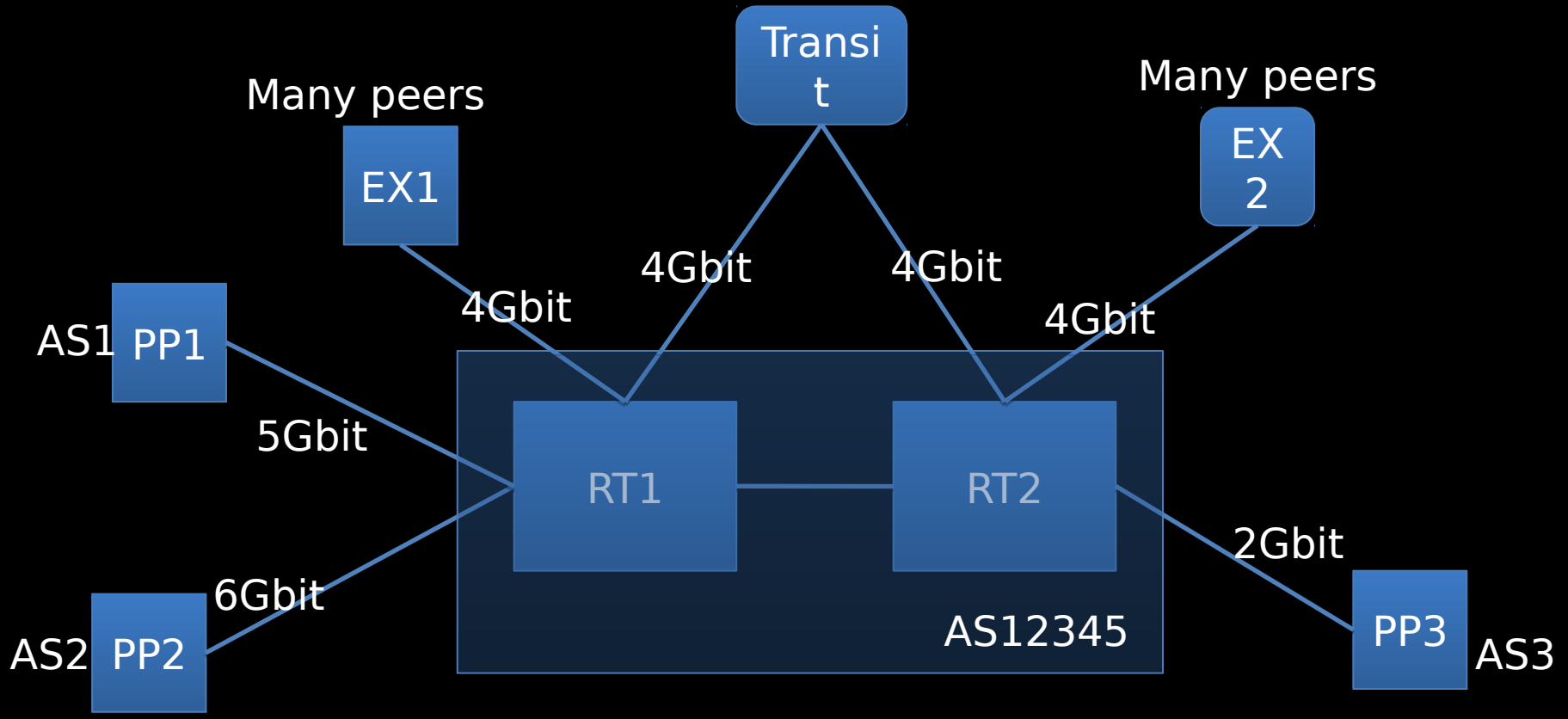
- Localpref
  - A **hammer** – blunt tool, inflexible.. But it is a tool.
  - “Generally” prefer to send traffic to customers, then peers, then transits.
  - Manage top ‘n’ networks, so that there is a **preferred path**, and a **failure path**, with capacity on both circuits.



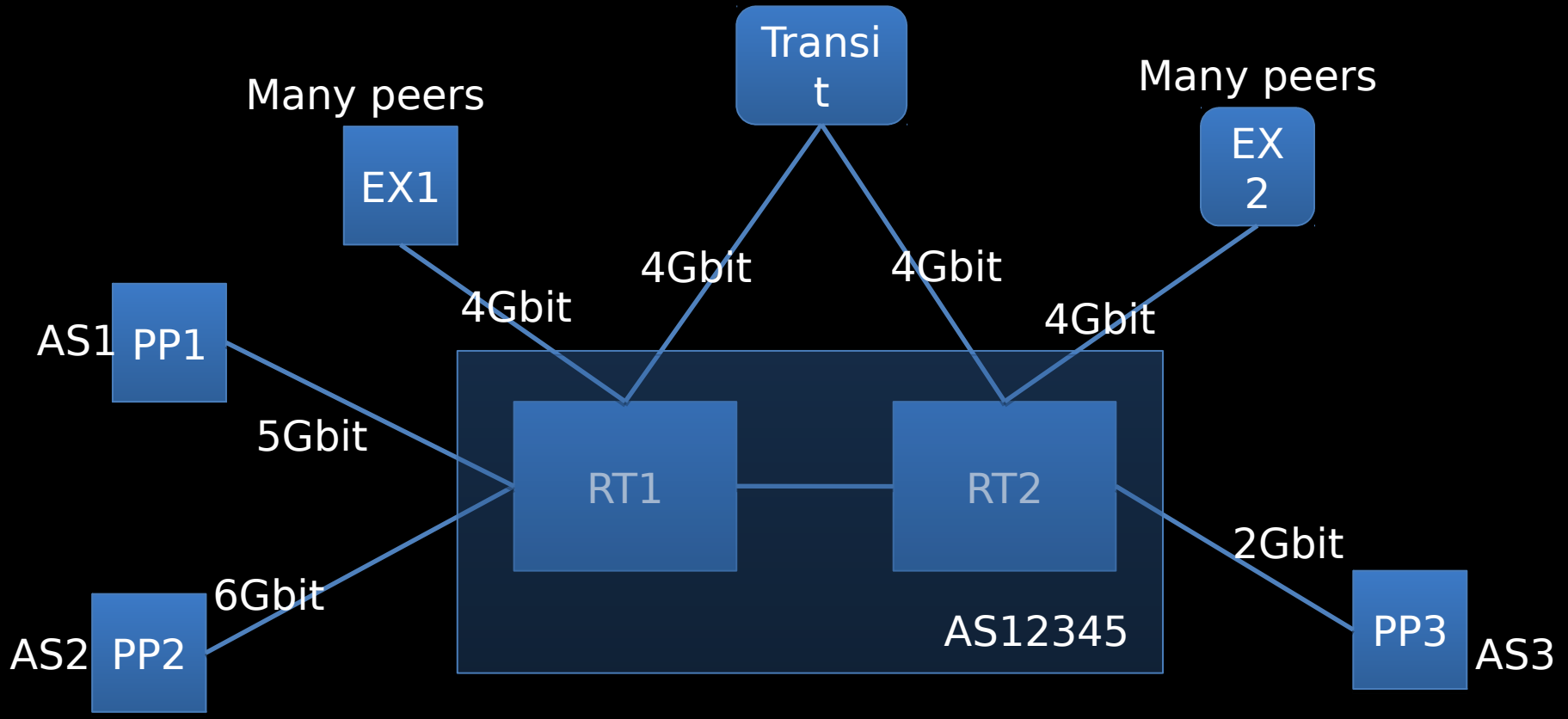
AS2 is your largest flow - via PP2 - maybe needs a second private peer backup



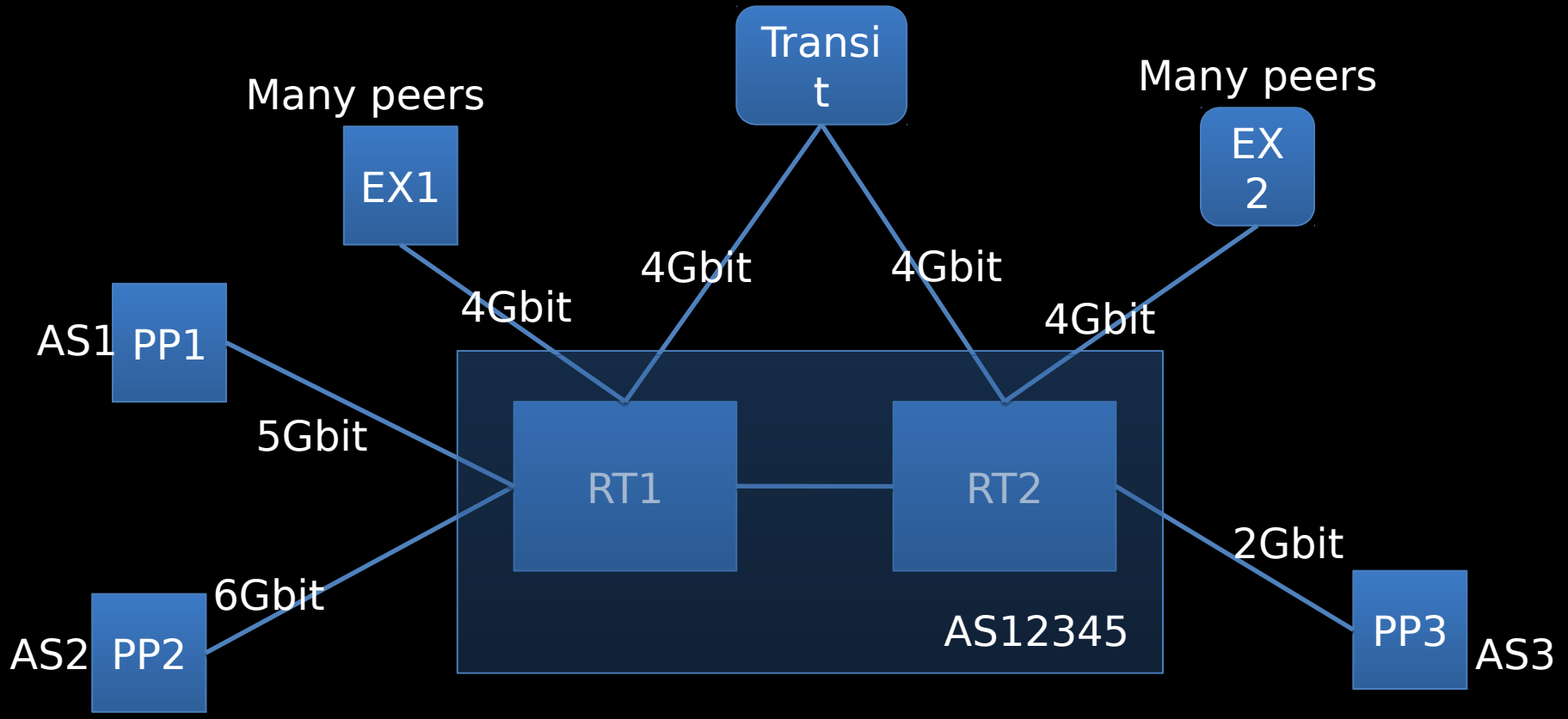
AS2 is your largest flow - via PP2 - maybe needs a second private peer backup
AS1 via PP1, configure a backup over EX1 or EX2 for deterministic



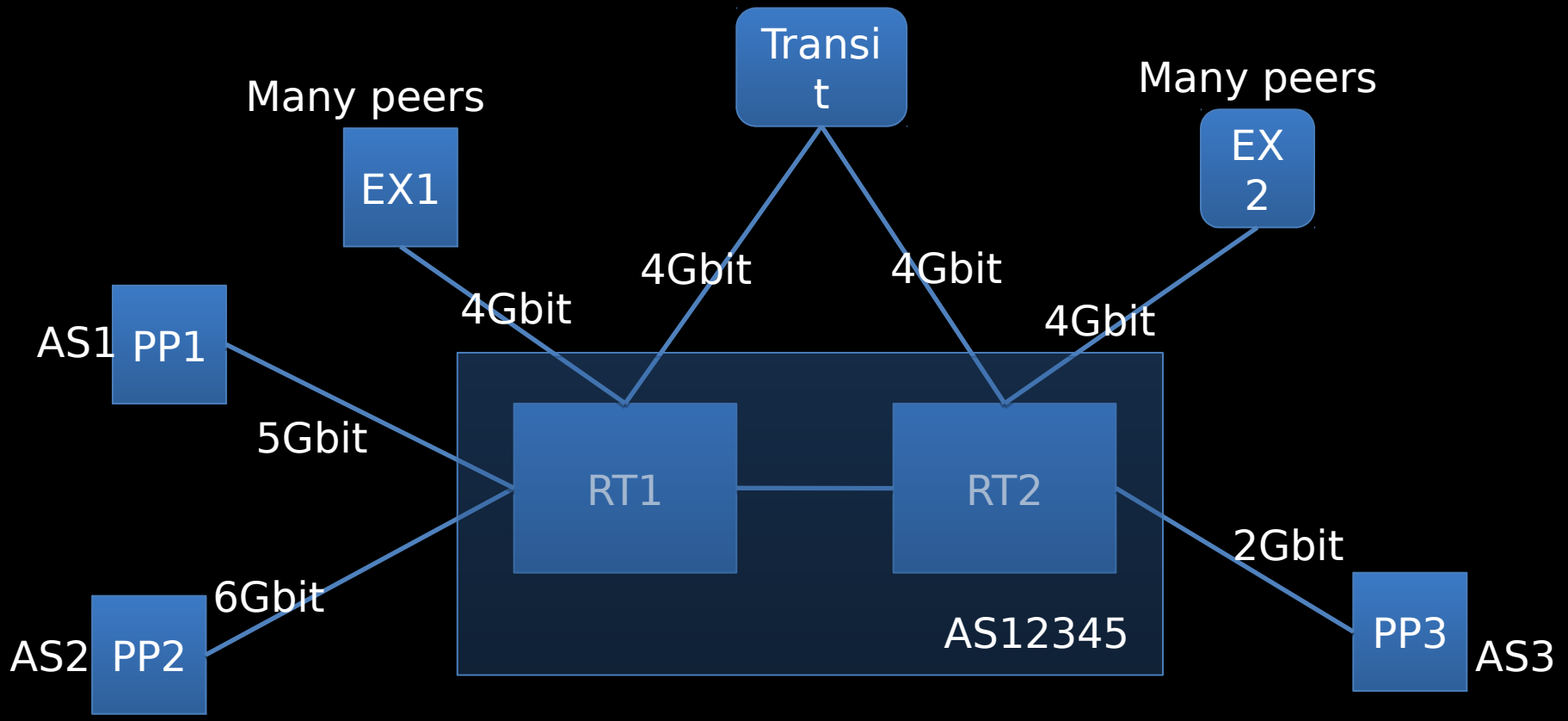
AS2 is your largest flow - via PP2 - maybe needs a second private peer backup
AS1 via PP1, configure a backup over EX1 or EX2 for deterministic
Can you move larger peers behind EX1 and EX2 onto private p



AS2 is your largest flow - via PP2 - maybe needs a second private peer backup
AS1 via PP1, configure a backup over EX1 or EX2 for deterministic
Can you move larger peers behind EX1 and EX2 onto private p
is an exchange failure, where will the traffic go? How big a flow should you care



AS2 is your largest flow - via PP2 - maybe needs a second private peer backup
AS1 via PP1, configure a backup over EX1 or EX2 for deterministic
Can you move larger peers behind EX1 and EX2 onto private p
is an exchange failure, where will the traffic go? How big a flow should you care
If you lose RT2, how will traffic to PP3 and traffic volume via EX2 be de



2 is your largest flow - via PP2 - maybe needs a second private peer backup on R
AS1 via PP1, configure a backup over EX1 or EX2 for deterministic routi
Can you move larger peers behind EX1 and EX2 onto private peeri
n exchange failure, where will the traffic go? How big a flow should you care abo
If you lose RT2, how will traffic to PP3 and traffic volume via EX2 be deliver
If you lose RT1, how will traffic volume via PP3 and EX1 be deliver

Localpref – blunt hammer

10.0.0.0/8 Localpref 100 via 100 123
10.0.0.0/8 Localpref 500 via 300 200 200 200 200 123

Which link will you prefer ?
AS123 here is trying to shape inbound traffic via AS100. Why ?
- Higher capacity link ?
- More reliable ?

What should you do ?

Answer: It depends on the **volume** of traffic, **cost** of capacity, **value** of traffic

Configuration Example

Larger flows are in ASNs
Listed in as-path 30 and 40

Deterministic exits configured

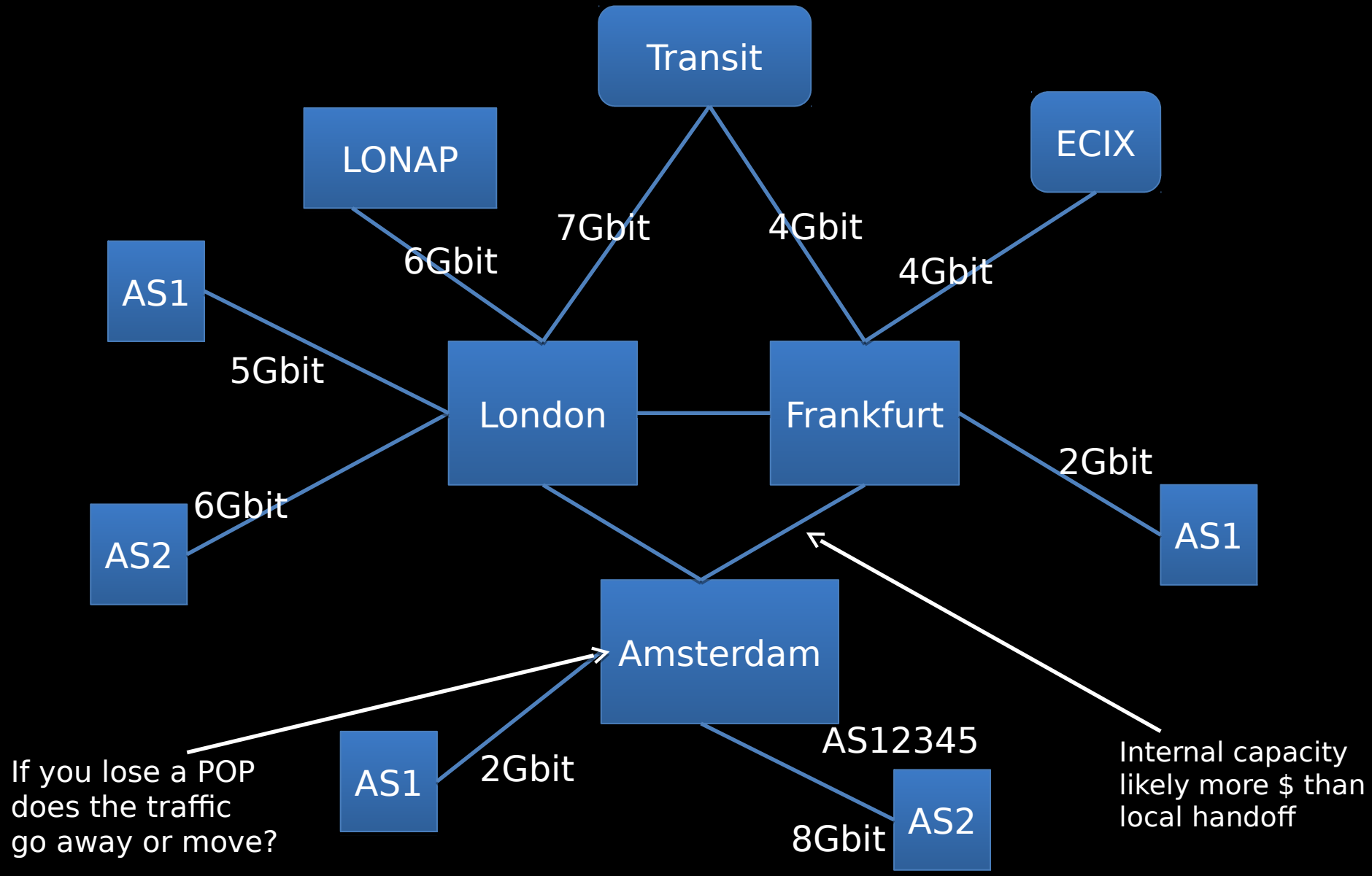
```
ip as-path access-list 30 permit _7018_
ip as-path access-list 30 permit _2828_
ip as-path access-list 30 permit _4323_
ip as-path access-list 30 permit _3561_
ip as-path access-list 30 permit _1668_
ip as-path access-list 40 permit _3330_

route-map PEER_EX1 permit 10
  match as-path 30
  set local-preference 300
route-map PEER_EX1 permit 15
  match as-path 40
  set local-preference 200
route-map PEER_EX1 permit 20
  set local-preference 150

route-map PEER_EX2 permit 10
  match as-path 40
  set local-preference 300
route-map PEER_EX2 permit 15
  match as-path 30
  set local-preference 200
route-map PEER_EX2 permit 20
  set local-preference 150
```

Mainly outbound – Many POPs

- Use hot potato routing to best effect
  - Nearest exit routing
  - Understand who your top traffic sinks are and peer at all POPs
  - Ignore MEDs from others – unless you want to carry the traffic on your backbone



If you understand your top flows, you will cope with traffic growth and failures

Deterministic routing

- Local Preference
- AS PATH length
- Lowest Origin Type
- Lowest MED
- Prefer eBGP paths
- Lowest IGP Metric
- Oldest route

Top flows should leave your network via deterministic means, and not left to BGP Best Path selection (or to chance)

If you are relying on oldest route to make the decision, you risk traffic taking unpredictable routes.

However, oldest routes do break the 'flapping sessions' problem. You need to monitor and manage your top flows constantly.

Inbound traffic engineering

- Much harder
  - Trick others' Best Path calculations
  - You do not administrate origin party router
- But remember…
  - Largest flows come from a small number of networks
  - Content networks want to deliver traffic to you as well as possible!

Selective Announcements

- Shortest prefix
- Local Preference
- AS PATH length
- Lowest Origin Type
- Lowest MED
- Prefer eBGP paths
- Lowest IGP Metric
- Oldest route

Prefix length considered before BGP.

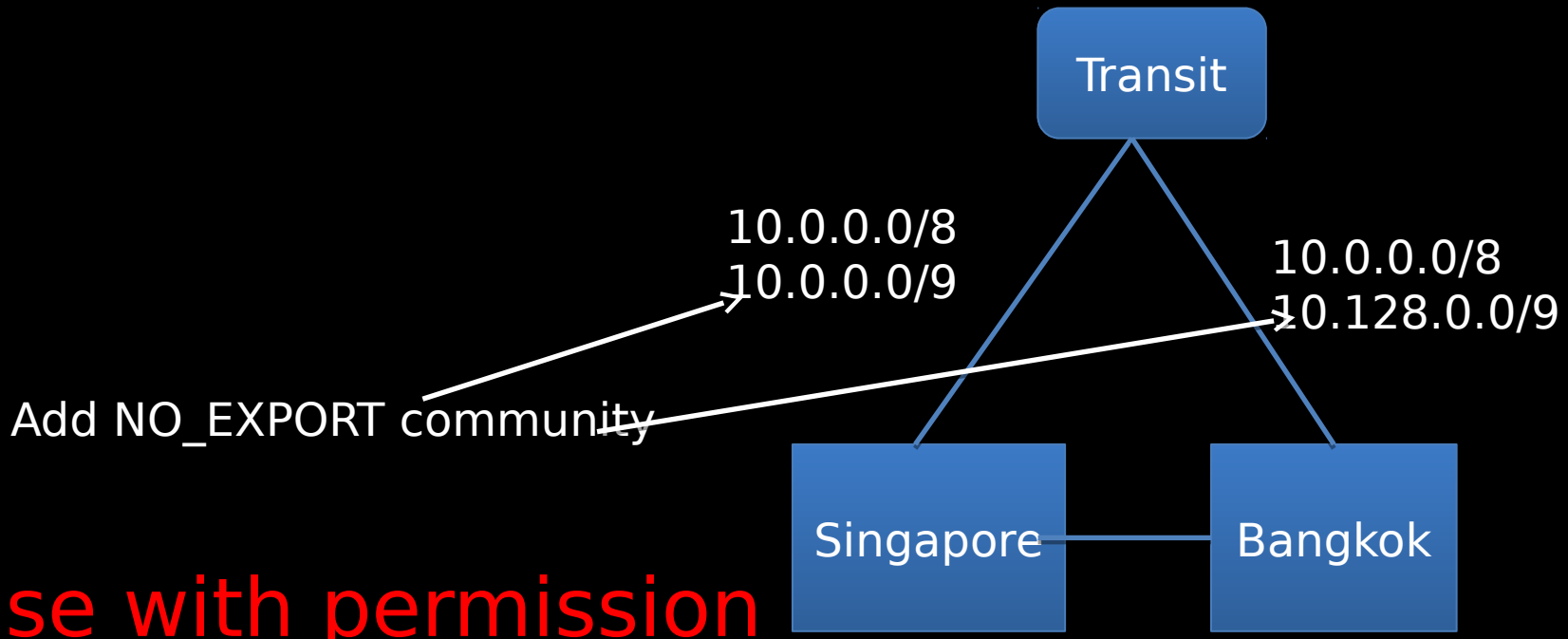
10.0.0.0/16 vs 10.0.0.0/17 & 10.128.0.0/17

Problem of Selective Announcements

- Often filtered
- Considered rude – might lead to depeering
- Never announce ‘globally’

…But can be used to great effect

- To the same peer or transit provider, announce aggregate and regional pfx



Use with permission

AS_PATH prepending

- Signal preferred path by growing AS_PATH on less preferred paths
- Marginal effect which degrades quickly
- Signal backup link to a single AS, but load-balancing capacity is much harder
- May not be heard at ‘distant’ ASNs
- Another ‘blunt’ tool, but can move some traffic.

2.5 AS Path Prepending

AS path prepending is a common way of making routes less attractive since AS path length is usually one of the BGP path selection criteria. A customer network may use these communities to selectively request AS3320 to insert additional copies of the AS number 3320 when propagating the customer routes to neighbors.

Community Value	Name	Description
65012 : X	AS Prepend 2x to AS X	Prepend 3320 two times to named peer (ASN=X)
65013 : X	AS Prepend 3x to AS X	Prepend 3320 three times to named peer (ASN=X)
6501n : 65001	AS Prepend by Class: Peer	Prepend 3320 n times to peers. n=2 or 3.
6501n : 65002	AS Prepend by Class: Upstream	Prepend 3320 n times to upstream.
6501n : 65003	AS Prepend by Class: Peer & Upstream	Prepend 3320 n times to peers and upstream.
6501n : 65004	AS Prepend by Class: Customer	Prepend 3320 n times to customers.
6501n : 65005	AS Prepend by Class: Customer & Peer	Prepend 3320 n times to customers and peers.
6501n : 65006	AS Prepend by Class: Customer & Upstream	Prepend 3320 n times to customers and upstream.
6501n : 65007	AS Prepend by Class: All	Prepend 3320 n times to all AS3320 neighbors.

Community Value	Name	Description
65001 : 100	Standard Local Preference	Set Local Preference to 100 (default).
65001 : 50	Low Priority Local Preference	Set Local Preference to 50.
65001 : 150	High Priority Local Preference	Raise Local Preference value to 150. Requires authorization from AS3320 backbone engineering.

2.4 Restrict Route Propagation

A customer network may use these communities to restrict propagation of its routes to AS3320 peers. However, the well known community NOPEER should be employed instead of these where appropriate.

Community Value	Name	Description
65010 : X	No Export to AS X	Do not advertise route(s) to named AS3320 peer (ASN=X)
65010 : 65001	No Export by Class: Peer	Do not advertise route(s) to AS3320 peers.
65010 : 65002	No Export by Class: Upstream	Do not advertise route(s) to AS3320 upstream.
65010 : 65003	No Export by Class: Peer & Upstream	Do not advertise route(s) to AS3320 peers and upstream.

MEDs

- Lowest MED wins.
  - Opposite of Nearest Exit routing, “carry traffic to me”
  - Only works to the same peer in multiple regions
  - Copy IGP metric to MED
  - Normally subject to negotiation
- Sometimes honoured, often when network traffic is latency or loss sensitive.

MEDs are often filtered

- Many networks set MED to 0 when they learn prefixes, so that hot potato routing will override MED.

```
route-map peers-in permit 10
    set local-preference 200
    set metric 0
```

Origin changing

- IGP
- EGP
- Incomplete

```
route-map PEERS permit 10
  set origin igp

route-route-map TRANSIT permit 10
  set origin incomplete
```

Often peers set to ‘igp’ or ‘egp’ statically on routers to nullify effects of Origin changing.

Inbound – what does work well?

- Overprovisioning
- Peer with top networks widely (buy options!)
  - Failure of single link will not break adjacency
  - Failures can be handled in predictable ways
- Build relationships
- **Constantly monitor and manage**
- If you care about your traffic, let it go.
  - Playing games with peering hurts your customers' traffic

- Affecting distant ASNs is very hard – a region may only see a single next-hop ASN.

What does “manage relationships” mean?

- Go back to your data
  - Collect and share information with peering co-ordinators at forums like this
  - You will stand out if you know exactly how much traffic you will exchange at peak with a peer
  - Protect your peer’s interests
    - Discuss mutual points of interconnection that suit both
    - Respond to abuse complaints promptly
    - Use contacts to reach other peering co-ordinators
    - Respond promptly to BGP session down/flapping
    - List your network on PeeringDB!

Buying transit in a smart way

- Buying from a well peered transit provider:
  - Can improve quality for the reasons discussed
  - Hides capacity problems from you automatically
- Buying from your top traffic destination
  - If your business relies on the traffic quality, it may make sense to pay
  - Data may help you negotiate good terms

Dealing with a “no” to peering

- Paid peering is one option
  - Often more expensive than full IP transit
  - “Once a customer, never a peer”
- Pay for other services in return for peering
  - Transport for example
- Peer around the problem
  - Try to peer directly with downstream customers
  - Try to sell directly to downstream customers
    - If you are better peered, you can sell based on quality

Constantly manage

- Peering on the Internet changes every day.
- Capacity on the Internet grows every day.
- Small networks become large.
- Large networks become larger (consolidation)
- A “bad” path might become good overnight

Questions?

Andy Davidson

andy@nosignal.org

Email me to request a copy of this presentation!

Feedback and introduction to peering co-ordinators welcome

Twitter: @andyd