Practical Implementation of BGP Community with Geotags and Traffic Engineering based on Geotags

Muhammad Moinur Rahman

BGP routes with location information

- Routers don’t have a builtin GPS device
- Need a process to manually Geocode it’s route-origin
- Need to add a tag to the route objects or group of routes
- Answer : BGP Communities

BGP – Border Gateway Protocol

- An INTER-AS routing protocol
- Building a robust scalable network
- Customers can chose
  - How to exit the network
  - How to bring the return traffic
- But to give that control we need to make maximum utilization of

BGP Community

Cause

- A scalable network needs them for its own use
  - Be able to identify customers, transits, peers, etc
  - To perform traffic engineering and export controls
  - There is no other truly acceptable implementation
- But customers love using them as well
  - “Power user” customers demand high level of control.
  - Having self-supporting customers doesn’t hurt either.
  - The more powerful you make your communities, the more work it will save you in the long run.

Controls and Caveats

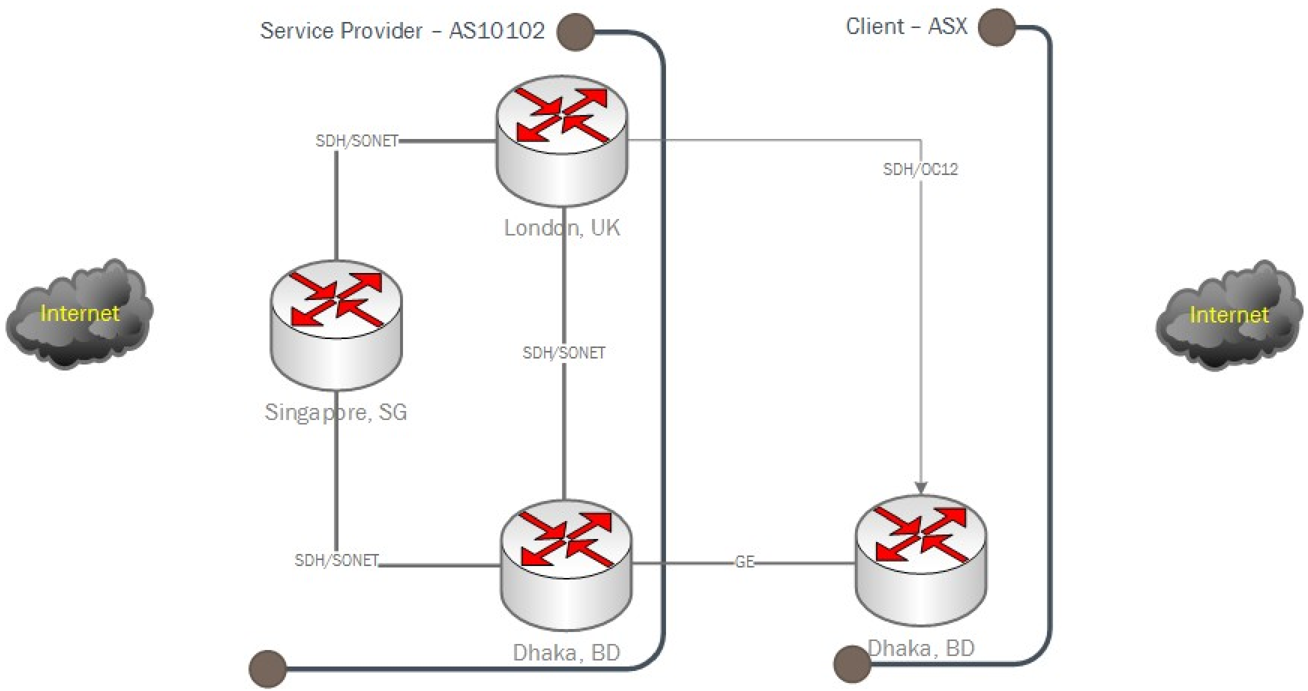
- Exit Traffic
  - You cannot decide
  - Customer needs to decide
  - But you can help a little more the customer to decide
- Return Traffic
  - You can control
  - You can help the customer decide

Problems faced by Service Provider

Customer Requires

1. Inbound Load Balancing
2. Upstream’s outbound traffic is preferring more expensive links
3. Upstream outbound traffic is preferring a higher latency/packet loss link
4. Traffic is not returning network efficiently (shortest/best path)
5. Return traffic load balancing due to cost/latency or Multiple geographical connectivity with same Transit Provider
6. Remotely Triggered Blackhole Route

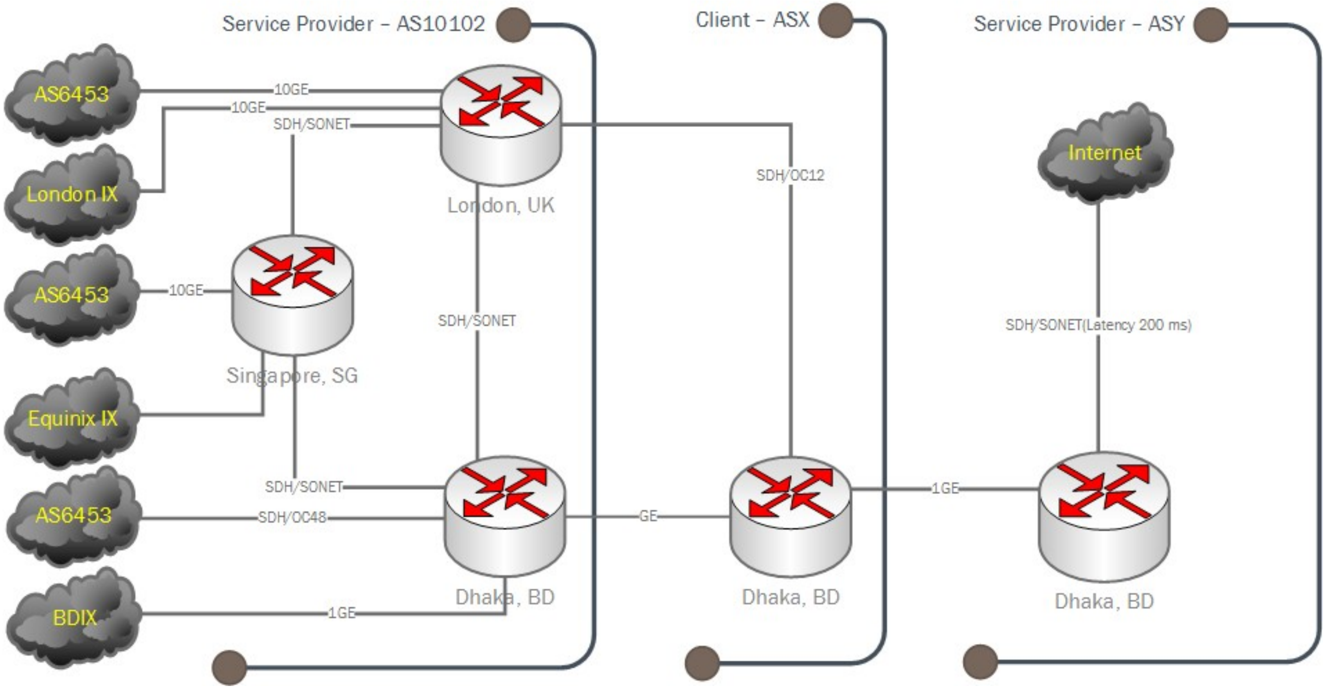
Problem 1 : Inbound Load Balancing



Problem 1 : Details

- Client has multiple connectivity
- To ensure Redundancy
- Advertising same prefixes in both links
- Downstream traffic is received by both links
- Client wants to specify specific link for incoming traffic

Problem 2: Outbound Traffic Preferring Expensive Links



Problem 2: Details

- Customer has Multiple upstreams
- Upstream AS10102 has connectivity with EQUINIX IX Singapore, London IX, BDIX
- Upstream ASY has no connectivity with any IX for Public Peering of Private peering. Only transit connectivity through another upstream
- ASY inbound routes are won in Customer's Router whereas AS10102 has so many better routes through IXP
- AS10102 provided BGP Communities for identifying Transit Routes, Private Peering Routes, Public Peering Routes and Customer Routes
- AS10102 defines different Local Preference for different routes

Problem 2 - Links – Types

- Transit
- Peers
  - Public
  - Private
- Customers

Problem 2 - Links – Types - Transit

- The network operator pays money (or *settlement*) to another network (Transit Provider) for Internet access (or *transit*)
- Pays Money
- Costliest

Problem 2 - Links – Types - Peers

- Public
  - Two networks exchange traffic between their users freely, and for mutual benefit
  - Connected at a public Internet eXchange Point like LINX, DE-CIX, AMS-IX
  - Internet eXchange Point charges a nominal fee for equipment financing and maintenance
  - Less Costlier than transit
- Private
  - Same as public except the two networks connectivity takes place privately
  - Only link costs are involved
  - Less Costlier than Public Peering

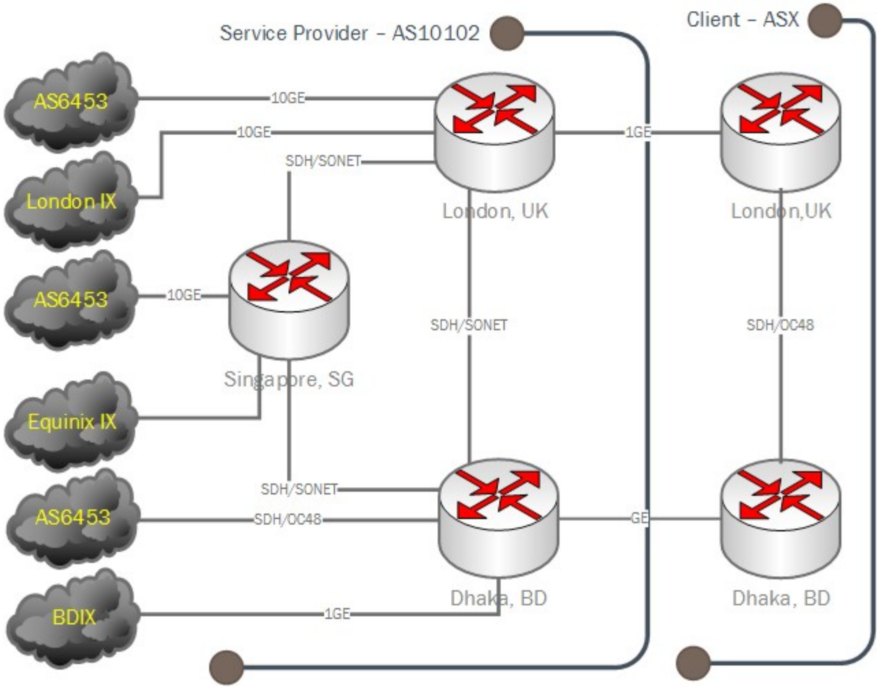
Problem 2 - Links – Types - Customer

- A network pays another network money to be provided with Internet access
- You are earning money
- No Costs involved as we are generating revenue

Problem 2 - Links – Budget Decision

- Customer Routes are the least expensive so Customer Routes should have higher preference
- Private Peering Routes are more expensive than Customer Routes so these should have lesser preference
- Public Peering Routes are more expensive than Private Peering Routes so these should have lesser preference
- Transit Routes are most expensive so these should have the least preference

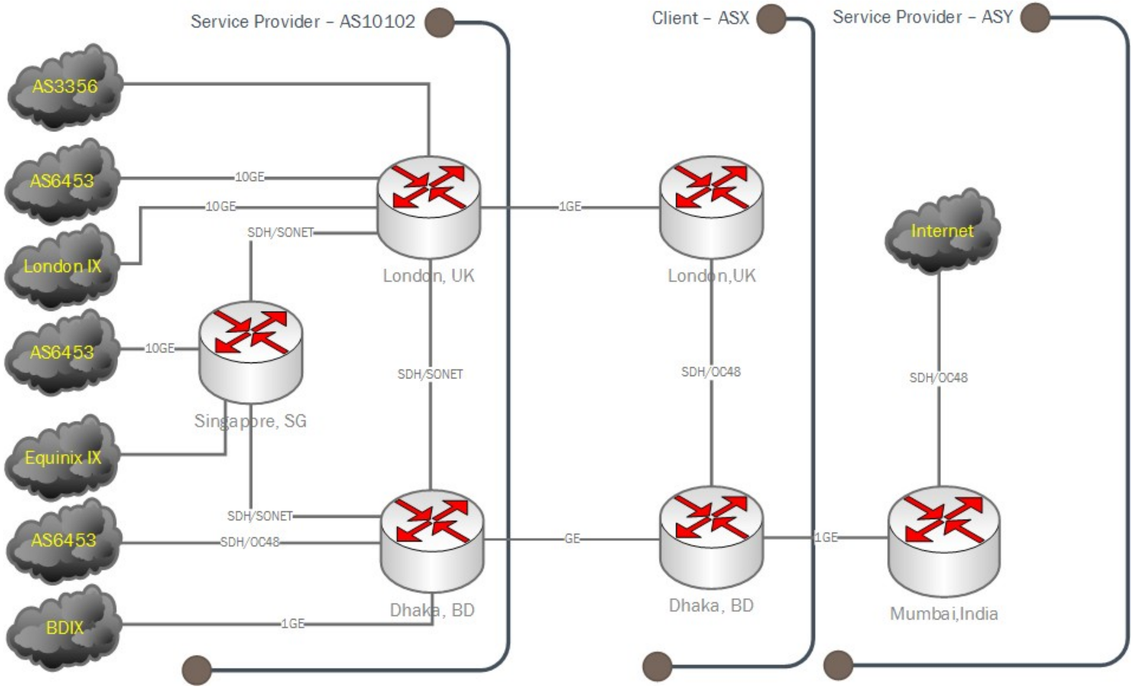
Problem 3 – Outbound Traffic Preferring Higher Latency Links



Problem 3 – Outbound Traffic Preferring Higher Latency Links

- Customer Routes are traversing through Higher Latency/Ping Loss Links
- Customer is connected with AS10102 in multiple location (Dhaka & London)
- Customer’s AP destined traffic from Dhaka is traversing High Latency London Router then going to Singapore
- Customer’s EU destined traffic from Dhaka is traversing High Latency Singapore Router then going to London
- Customer’s AP destined traffic from London is traversing High Latency Dhaka Router then going to Singapore

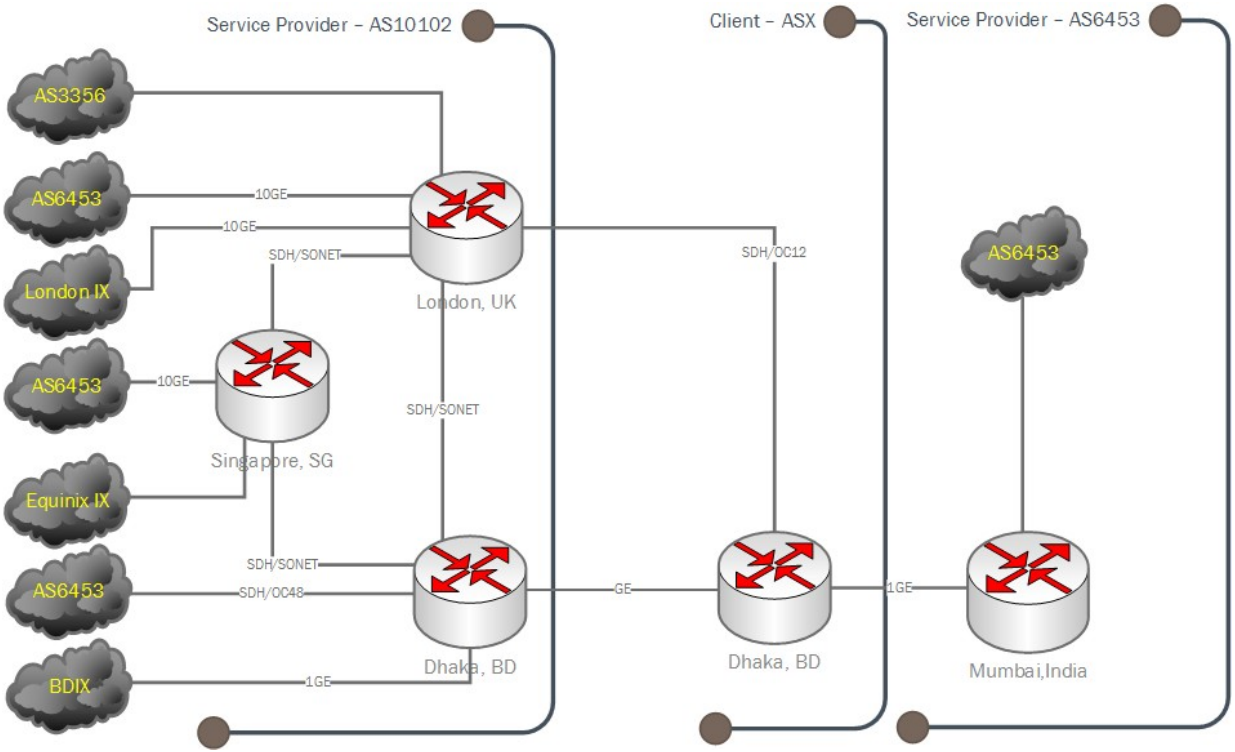
Problem 4: Return Traffic is not exiting network efficiently



Problem 4: Return Traffic is not Exiting Efficiently

- Customer is Connected with AS10102 in Dhaka & London
- Customer is also connected with ASY in Dhaka
- Customer is advertising same prefixes across all upstream from both the routers
- Customers Dhaka destined traffic from London Router(AS10102) is coming through Customer’s London Router instead of Upstream’s Dhaka Router
- Customers London destined traffic from Dhaka Router(AS10102) is coming through Customer’s London Router instead of Upstream’s Dhaka Router
- Customer cannot change Local Preference. Changing it causes all it’s traffic routed via ASY

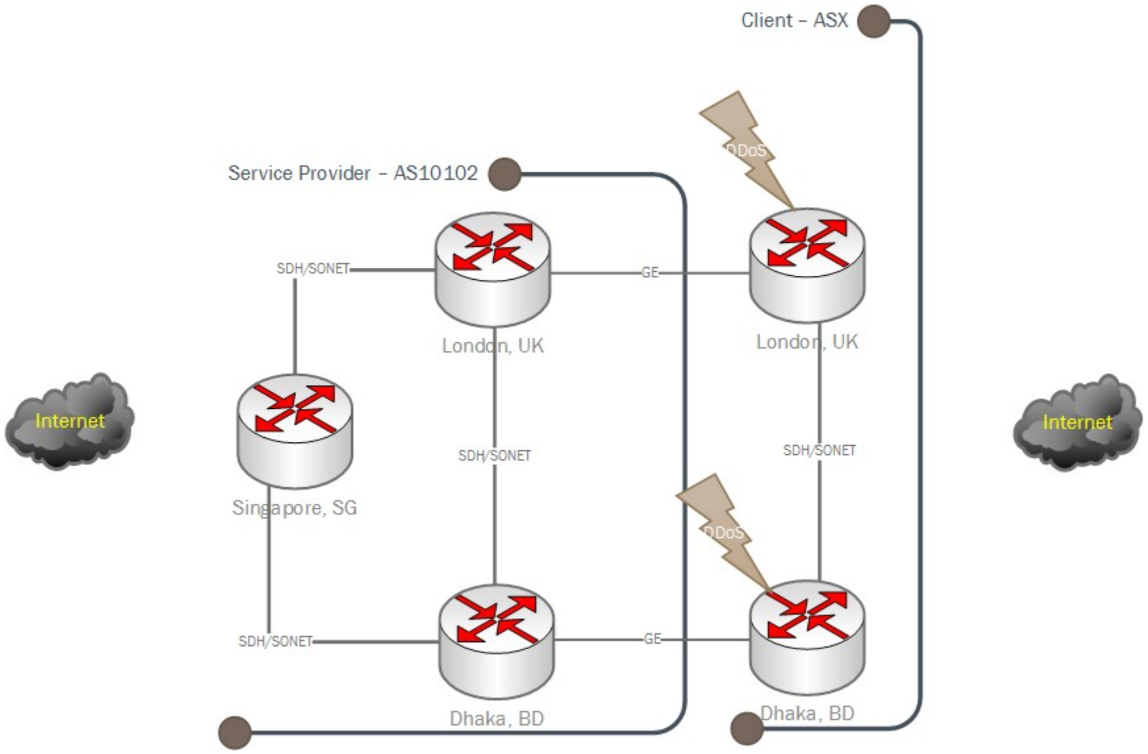
Problem 5: Return Traffic Load Balancing



Problem 5: Return Traffic Load Balancing

- Customer is connected with AS10102 and AS6453 in both London and Dhaka
- AS6453 is Tire-1 Provider whereas AS10102 is not
- As Customer has direct connectivity with AS6453; Customer doesn’t want it’s AP AS6453 return traffic from Dhaka router(AS10102) to come through AS10102 only except last resort
- Customer doesn’t want EU return traffic from AS6453 via AS10102 as they have better latency through other tertiary provider

Problem 6: Remotely Triggered BlackHole Route



Problem 6: Remotely Triggered BlackHole Route

- Client suddenly starts facing massive DDoS attacks
- All of Client's backbone links are saturated
- Client needs to let the upstream know about the IP
- Upstream will NULL route that specific IP

BGP Community Analysis – AS6453

- Accepts STANDARD Communities
  - NO_EXPORT, NO_ADVERTISE
- Local Preference Adjustment
  - Accepts 70, 80, 90, and 110
  - Standard for Customer Routes 100
  - Standard for Peer Routes 90
  - Nothing defined for Transit Routes as they are a TRANSIT Free Operator
- Mitigation of DDoS attacks/ Remotely Triggered Blackhole Routes
  - /32 prefixes can be blackholed with 64999:0(Uses PRIVATE number)

BGP Community Analysis – AS6453(Cont)

- Distribution to PEERS
  - 6500n:ASN n={1,2,3} prepend 6453 n times to peer ASN
  - 65009:ASN do not redistribute to peer ASN
  - 6500n:0 n={1,2,3} prepend 6453 to all peers
  - 65009:0 do not redistribute to any peers
- Informational Community
  - Peer Routes classified as 6453:86
  - Nothing defined for Customer Routes
  - Nothing defined for Transit Routes as they are transit free operator
  - Geographical Information encoded within 4 characters with limited visibility

Solution 1: Inbound Load Balancing

- Changing the Local Preference on one link
- But Local Preference does not travers across different AS
- Upstream needs to change the Local Preference while configuring BGP inbound route policy
- But Upstream can set Local Preference based on conditional BGP community check

Solution 1: Inbound Load Balancing :: Upstream IOS-XR

```
route-policy CUSTOMER-IN
if community matches-any (10102:75) then
    set local-preference 75
elseif community matches-any (10102:85) then
    set local-preference 85
elseif community matches-any (10102:95) then
    set local-preference 95
elseif community matches-any (10102:105) then
   set local-preference 105
else
    done
endif
end-policy
```

Solution 1: Inbound Load Balancing :: Upstream IOS

```
ip community-list 1 permit 10102:75
ip community-list 2 permit 10102:85
ip community-list 3 permit 10102:95
ip community-list 4 permit 10102:105
route-map CUSTOMER-IN permit 10
 match community 1
 set local-preference 75
route-map CUSTOMER-IN permit 20
 match community 2
 set local-preference 85
route-map CUSTOMER-IN permit 30
 match community 3
 set local-preference 95
route-map CUSTOMER-IN permit 40
 match community 4
 set local-preference 105
```

Solution 1: Inbound Load Balancing :: Customer IOS :: London Link

```
route-map AS10102-OUT-LONDON permit 1000
 set community 10102:{75,85,95,105}
```

Solution 2: Outbound Traffic Preferring Expensive Links

- Upstream has two tasks
  - Setting Local preference
    - Transit Local Preference – 90
    - Public/Private Peering Local Preference – 95
    - Client Local Preference – 100
- Setting BGP Community to isolate routes
  - Transit Routes – 1^{st} digit of 2^{nd} octet is 1 (10102:1XXXX)
  - Public/Private Peer Routes – 1^{st} digit of 2^{nd} octet is 2/3 (10102:2XXXX/10102:3XXXX)
  - Client Routes – 1^{st} digit of 2^{nd} octet is 4 (10102:4XXXX)
  - Define GeoTAGS

Solution 2: Outbound Traffic Preferring Expensive Links :: Upstream IOS-XR :: DHAKA

```
route-policy CUSTOMER-IN
  set community 10102:41011 additive
end-policy
route-policy PRIVATE-PEER-IN
  set community 10102:31011 additive
  set local-preference 90
end-policy
route-policy PUBLIC-PEER-IN
  set community 10102:21011 additive
  set local-preference 90
end-policy
route-policy TRANSIT-IN
  set community 10102:11011 additive
  set local-preference 80
end-policy
```

Solution 2: Outbound Traffic Preferring Expensive Links :: Customer

- Upstream has defined optimized outbound routing
- But customer has to route selective traffic upto upstreams router
- Local Preference rests to vendor neutral default value across AS

Solution 2: Outbound Traffic Preferring Expensive Links :: Customer IOS

```
ip community-list 100 permit 10102:[2-3].{4}
ip community-list 101 permit 10102:1.{4}
route-map AS10102-IN permit 10
 match community 100
 set local-preference 90
route-map AS10102-IN permit 20
 match community 101
 set local-preference 80
route-map ASY-IN permit 10
 set local-preference 80
```

Solution 3: Outbound Traffic Preferring Higher Latency Links

- Upstream is TAGGING routes based on it's origin as follows
  - ASIAN region – 2nd digit on 2nd octet is 1(10102:X1XXX)
  - AFRICAN region – 2nd digit on 2nd octet is 2(10102:X2XXX)
  - EUROPEAN region – 2nd digit on 2nd octet is 3(10102:X3XXX)
  - NORTH AMERICAN region – 2nd digit on 2nd octet is 4(10102:X4XXX)
  - SOUTH AMERICAN region – 2nd digit on 2nd octet is 5(10102:X5XXX)
  - AUSTRALIAN region – 2nd digit on 2nd octet is 6(10102:X6XXX)
  - ANTARCTICA region – 2nd digit on 2nd octet is 7(10102:X7XXX)

Solution 3: Outbound Traffic Preferring Higher Latency Links :: UPSTREAM IOS-XR :: DHAKA

- Peer Configuration

```
route-policy SETCMTY-PEER
  set community 10102:21011 additive
  end-policy
```

- Transit Configuration

```
route-policy SETCMTY-TRANSIT
   set community 10102:11011 additive
   end-policy
```

- Client configuration

```
route-policy SETCMTY-CUSTOMER
   set community 10102:41011 additive
   end-policy
```

Solution 3: Outbound Traffic Preferring Higher Latency Links :: UPSTREAM IOS-XR :: SINGAPORE

- Peer Configuration

```
route-policy SETCMTY-PEER
  set community 10102:21021 additive
  end-policy
```

- Transit Configuration

```
route-policy SETCMTY-TRANSIT
  set community 10102:11021 additive
  end-policy
```

- Client configuration

```
route-policy SETCMTY-CUSTOMER
  set community 10102:41021 additive
  end-policy
```

Solution 3: Outbound Traffic Preferring Higher Latency Links :: UPSTREAM IOS-XR :: LONDON

- Peer Configuration

```
route-policy SETCMTY-PEER
  set community 10102:23011 additive
  end-policy
```

- Transit Configuration

```
route-policy SETCMTY-TRANSIT
  set community 10102:13011 additive
  end-policy
```

- Client configuration

```
route-policy SETCMTY-CUSTOMER
  set community 10102:43011 additive
  end-policy
```

Solution 3: Outbound Traffic Preferring Higher Latency Links :: Customer :: London ::IOS

```
ip community-list 100 permit 10102:[1-5]1.{3}
ip community-list 101 permit 10102:[1-5]3.{3}
route-map AS10102-IN permit 10
 match community 101
 set weight 1000
route-map ASCUSTOMER-IN permit 10
 match community 100
 set weight 1000
```

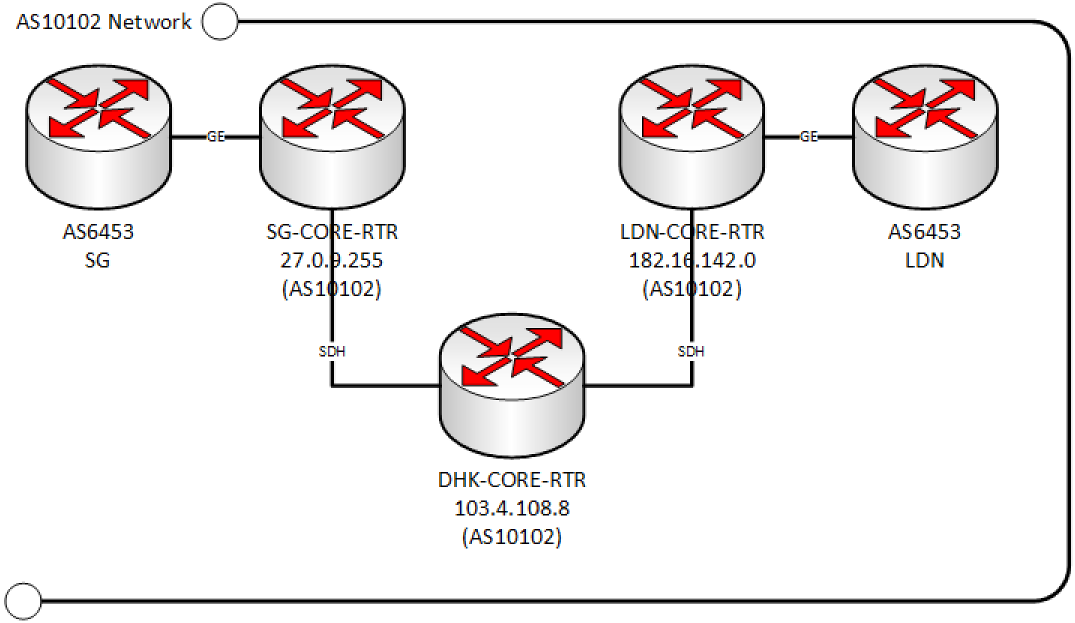
Solution 3: Outbound Traffic Preferring Higher Latency Links :: Customer :: Dhaka :: IOS

```
ip community-list 100 permit 10102:[1-5]1.{3}
ip community-list 101 permit 10102:[1-5]3.{3}
route-map AS10102-IN permit 10
 match community 100
 set weight 1000
route-map ASCUSTOMER-IN permit 10
 match community 101
 set weight 1000
```

Solution 3: Outbound Traffic Preferring Higher Latency Links :: Real Life Example

- This has been a debate where it really gives added advantage
- AS10102 is using this sort of setup
- AS10102 thinks this gives better result

Solution 3: Outbound Traffic Preferring Higher Latency Links :: Real Life Example :: Scenario



Solution 3: Outbound Traffic Preferring Higher Latency Links :: Real Life Example

- 6453:3000 – AP Region
- 6453:2000 – EU Region

Case1 : 1.0.4.1 (BGP Community 6453:3000)

```
Path #4: Received by speaker 0
  Advertised to update-groups (with more than one peer):
    0.3 0.9 0.16 0.21 0.29
  Advertised to peers (in unique update groups):
    103.4.108.138   103.4.109.170   103.4.109.134
  9498 7545 56203
    125.23.197.26 from 125.23.197.26 (203.101.87.173)
      Origin IGP, localpref 90, valid, external, best, group-best, import-candidate
      Received Path ID 0, Local Path ID 1, version 369496799
      Community: 10102:11000 10102:11010 10102:11011
      Origin-AS validity: not-found
```

Case1 : 1.0.4.1 (Default BGP Table TraceRoute)

```
Sun Dec 14 21:49:24.004 UTC
Type escape sequence to abort.
Tracing the route to 1.0.4.1
 1  125.17.2.53 45 msec  45 msec
 2  125.62.187.113 436 msec  447 msec  448 msec
 3  any2ix.coresite.com (206.72.210.141) 621 msec  622 msec  607 msec
 4  203-29-129-130.static.tpgi.com.au (203.29.129.130) 615 msec  614 msec  613 msec
 5  203-29-140-110.static.tpgi.com.au (203.29.140.110) 613 msec  620 msec  617 msec
 6  203-29-140-110.static.tpgi.com.au (203.29.140.110) 612 msec  619 msec  623 msec
 7   *  *  *
 8   *  *  *
 9   *  *  *
 10  *  *  *
 11 203-26-30-91.static.tpgi.com.au (203.26.30.91) 632 msec  *  *
```

Case1 : 1.0.4.1 (via London)

```
Sun Dec 14 21:56:45.596 UTC
Type escape sequence to abort.
Tracing the route to 1.0.4.1
 1  182.16.142.13 175 msec  168 msec
 2  if-3-1-1.core4.LDN-London.as6453.net (195.219.51.85) 167 msec  168 msec  173 msec
 3  if-0-1-3-0.tcore2.LDN-London.as6453.net (80.231.62.25) [MPLS: Label 614247 Exp 0] 324 msec  318 msec  316 msec
 4   if-15-2.tcore2.L78-London.as6453.net (80.231.131.117) 324 msec  335 msec
 5  if-20-2.tcore2.NYY-New-York.as6453.net (216.6.99.13) [MPLS: Label 467080 Exp 0] 321 msec  319 msec  321 msec
 6  *  *  *
 7  if-1-2.tcore1.PDI-Palo-Alto.as6453.net (66.198.127.5) [MPLS: Label 353936 Exp 0] 317 msec  317 msec  319 msec
 8  if-2-2.tcore2.PDI-Palo-Alto.as6453.net (66.198.127.2) [MPLS: Label 344256 Exp 0] 320 msec  319 msec  319 msec
 9  if-5-2.tcore2.SQN-San-Jose.as6453.net (64.86.21.1) 317 msec  316 msec  316 msec
 10 64.86.21.58 516 msec
11 syd-sot-ken-crt1-be-10.tpgi.com.au (203.219.35.1) 532 msec
 12 203-29-140-110.static.tpgi.com.au (203.29.140.110) 565 msec  565 msec  556 msec
 13 203-29-140-110.static.tpgi.com.au (203.29.140.110) 537 msec  545 msec  545 msec
 14   203-26-30-91.static.tpgi.com.au (203.26.30.91) 530 msec  530 msec
```

Case1 : 1.0.4.1 (After route-policy)

```
Sun Dec 14 21:54:55.800 UTC
Type escape sequence to abort.
Tracing the route to 1.0.4.1
 1  if-0-9-0-2.core01.PSB-Dhaka.1asiacom.net (27.0.9.18) 88 msec  95 msec
 2  ix-0-1-3-565.tcore1.SVQ-Singapore.as6453.net (120.29.215.25) 91 msec  91 msec  88 msec
 3  if-20-2.tcore2.SVW-Singapore.as6453.net (180.87.96.22) 118 msec  *
 4  if-1-2.tcore1.HK2-Hong-Kong.as6453.net (180.87.112.1) 116 msec  117 msec  116 msec
 5  116.0.67.34 233 msec  235 msec  233 msec
 6  203-29-129-193.static.tpgi.com.au (203.29.129.193) 244 msec  243 msec  265 msec
 7  203-29-140-110.static.tpgi.com.au (203.29.140.110) 287 msec  264 msec  252 msec
 8  203-29-140-110.static.tpgi.com.au (203.29.140.110) 251 msec  252 msec  252 msec
 9   *  *  *
 10  *  *  *
 11 203-26-30-91.static.tpgi.com.au (203.26.30.91) 259 msec  *  *
```

Case1 : 1.0.4.1 (ROUTE-POLICY)

```
route-policy AS10102-IN
  if source in (27.0.9.255) and (community
matches-any AS6453-AP) then
    set weight 2000
else
    done
  endif
end-policy
```

Case2 : 1.8.52.1 (BGP Community 6453:2000)

```
  Path #1: Received by speaker 0
  Advertised to update-groups (with more than one peer):
    0.3 0.9 0.16 0.21 0.29
  Advertised to peers (in unique update groups):
    103.4.108.138   103.4.109.170   103.4.109.134
  6453 38345, (aggregated by 38345 209.58.75.66), (Received from a RR-client),
(received & used)
    27.0.9.255 (metric 10) from 27.0.9.255 (27.0.9.255)
      Origin IGP, localpref 90, valid, internal, best, group-best, import-candidate
      Received Path ID 0, Local Path ID 1, version 377221249
      Community: 6453:50 6453:1000 6453:1100 6453:1112 10102:11000 10102:11020
10102:11021
```

Case2 : 1.8.52.1 (Default BGP Table TraceRoute)

```
 Sun Dec 14 22:24:20.319 UTC
Type escape sequence to abort.
Tracing the route to 1.8.152.1
 1  if-0-7-1-2.core01.EDC-Singapore.1asiacom.net (27.0.9.6) 90 msec  89 msec
 2  ix-0-1-3-565.tcore1.SVQ-Singapore.as6453.net (120.29.215.25) 87 msec  87 msec  88 msec
 3  if-20-2.tcore2.SVW-Singapore.as6453.net (180.87.96.22) [MPLS: Label 702807 Exp 0] 293 msec  294 msec  *
 4  if-2-2.tcore1.SVW-Singapore.as6453.net (180.87.12.1) [MPLS: Label 383568 Exp 0] 291 msec  292 msec  292 msec
 5  if-6-2.tcore2.TV2-Tokyo.as6453.net (180.87.12.110) [MPLS: Label 722261 Exp 0] 305 msec  303 msec  303 msec
 6  if-2-2.tcore1.TV2-Tokyo.as6453.net (180.87.180.1) [MPLS: Label 475028 Exp 0] 306 msec  305 msec  *
 7  if-9-2.tcore2.PDI-Palo-Alto.as6453.net (180.87.180.17) 313 msec  305 msec
 8  if-2-2.tcore1.PDI-Palo-Alto.as6453.net (66.198.127.1) [MPLS: Label 338135 Exp 0] 303 msec  304 msec  304 msec
 9  if-1-2.tcore1.NYY-New-York.as6453.net (66.198.127.6) [MPLS: Label 747619 Exp 0] 305 msec  303 msec  305 msec
 10 if-4-0-0.mse1.NW8-New-York.as6453.net (216.6.90.42) 307 msec  306 msec  306 msec
 11 ix-9-5.mse1.NW8-New-York.as6453.net (209.58.75.66) 302 msec  303 msec  302 msec
 12 gns1.zdnscloud.net (1.8.152.1) 306 msec  304 msec  304 msec
```

Case2 : 1.8.52.1 (After route-policy)

```
Sun Dec 14 22:22:00.881 UTC
Type escape sequence to abort.
Tracing the route to 1.8.152.1
 1  182.16.142.13 168 msec  167 msec
 2  if-3-1-1.core4.LDN-London.as6453.net (195.219.51.85) 187 msec  195 msec  199 msec
 3  if-2-3-1-0.tcore1.LDN-London.as6453.net (80.231.76.122) [MPLS: Label 506579 Exp 0] 183 msec  172 msec  172 msec
 4  if-17-2.tcore1.L78-London.as6453.net (80.231.130.129) [MPLS: Label 412885 Exp 0] 172 msec  173 msec  *
 5  if-11-2.tcore2.SV8-Highbridge.as6453.net (80.231.139.41) [MPLS: Label 441702 Exp 0] 173 msec  172 msec  186 msec
 6  if-0-3-6-5.thar2.HW1-London.as6453.net (195.219.101.49) 172 msec
 7  ibercom-gw.as6453.net (195.219.101.38) 172 msec  172 msec  175 msec
 8  gns1.zdnscloud.net (1.8.152.1) 172 msec  171 msec  172 msec
```

Case2 : 1.8.52.1 (ROUTE-POLICY)

```
route-policy AS10102-IN
  if source in (182.16.142.0) and (community
matches-any AS6453-EU) then
    set weight 2000
  else
    done
  endif
end-policy
```

Solution 3: Outbound Traffic Preferring Higher Latency Links :: Real Life Example :: Results

In both cases:

- Lower latency
- Lower hop count
- Better performance

Solution 4: Return Traffic is not exiting network efficiently

- Changing Local Preference in any side might force Customer’s all traffic to come through ASY
- Changing Local Preference in any side might result in AS10102’s transit routes to win over Customer’s route
- Safest attribute to handle is MED(Multi Exit Discriminator)
- Lower the metric value; more preferable the route is

Solution 4: Return Traffic is not exiting network efficiently :: Upstream :: IOS-XR

```
route-policy CUSTOMER-IN
if community matches-any (10102:4000) then
    set metric 0
else
    done
endif
end-policy
```

Solution 4: Return Traffic is not exiting network efficiently :: Customer :: IOS :: London

```
ip prefix-list LONDON ..
ip prefix-list DHAKA ..
route-map AS10102-OUT-LONDON permit 10
 match ip address prefix-list LONDON
 set community 10102:4000
route-map AS10102-OUT-LONDON permit 20
 match ip address prefix-list DHAKA
```

Solution 4: Return Traffic is not exiting network efficiently :: Customer :: IOS :: Dhaka

```
ip prefix-list LONDON ..
ip prefix-list DHAKA ..
route-map AS10102-OUT-DHAKA permit 10
 match ip address prefix-list DHAKA
 set community 10102:4000
route-map AS10102-OUT-DHAKA permit 20
 match ip address prefix-list LONDON
```

Solution 5: Return Traffic Load Balancing

- Customer will use AS10102 for transit to AS6453 AP region as the last resort
- Customer's prefix has to be AS path prepended to AS6453 in AP region
- Customer prefix has not to be advertised to AS6453 EU region but to other upstream and peers

Solution 5: Return Traffic Load Balancing

- AS10102 has community for
  - AS path-prepend once (10102:1CTP)
  - AS path-prepend twice (10102:2CTP)
  - AS path-prepend thrice (10102:3CTP)
  - Do not redistribute (5CTP)
    - C = {0..7}
      - 0 – Globally
      - 1-7 – Region Code
    - TP Provider code
      - 00 – Globally
      - 01 – AS6453
      - 02 – AS3356

Solution 5: Return Traffic Load Balancing :: Upstream :: IOS-XR :: Dhaka

```
route-policy PROCCMTY-AS6453-OUT
if community matches-any (10102:1001, 10102:1101) then
    prepend as-path 10102 1
  elseif community matches-any (10102:2001, 10102:2101) then
    prepend as-path 10102 2
  elseif community matches-any (10102:3001, 10102:3101) then
    prepend as-path 10102 3
  elseif community matches-any (10102:5000, 10102:5001, 10102:5101) then
    drop
  else
    done
  endif
end-policy
```

Solution 5: Return Traffic Load Balancing :: Upstream :: IOS-XR :: Singapore

```
route-policy PROCCMTY-AS6453-OUT
if community matches-any (10102:1001, 10102:1101) then
    prepend as-path 10102 1
  elseif community matches-any (10102:2001, 10102:2101) then
    prepend as-path 10102 2
  elseif community matches-any (10102:3001, 10102:3101) then
    prepend as-path 10102 3
  elseif community matches-any (10102:5000, 10102:5001, 10102:5101) then
    drop
  else
    done
  endif
end-policy
```

Solution 5: Return Traffic Load Balancing :: Upstream :: IOS-XR :: London

```
route-policy PROCCMTY-AS6453-OUT
if community matches-any (10102:1000,10102:1001, 10102:1301) then
    prepend as-path 10102 1
  elseif community matches-any (10102:2000, 10102:2001, 10102:2301) then
    prepend as-path 10102 2
  elseif community matches-any (10102:3000, 10102:3001, 10102:3301) then
    prepend as-path 10102 3
  elseif community matches-any (10102:5000, 10102:5001, 10102:5301) then
    drop
  else
    done
  endif
end-policy
```

Solution 5: Return Traffic Load Balancing :: Customer :: IOS

```
route-map AS10102-OUT permit 10
 match <CLAUSE>
 set community 10102:3101 10102:5301
```

Solution 6: Remotely Triggered BlackHole Route

- Customer is facing DoS or DDoS
- Customer has to let it's upstream or upstream's upstream know about the destination address
- AS10102 has community for
  - Remotely Triggering Blackhole Route (10102:0)

Solution 6: Remotely Triggered BlackHole Route :: Upstream :: IOS-XR

```
interface Null 0 ipv4 unreachables disable
router static
 address-family ipv4 unicast
  192.0.2.0/32 Null0
router bgp 10102
address-family ipv4 unicast
  redistribute static route-policy black-hole
route-policy CUSTOMER-IN
  if community matches-any (10102:0) then
        set tag 66
        set local-preference 200
        set origin igp
        set next-hop 192.0.2.0
        set community (no-export)
  endif
end-policy
```

```
route-policy black-hole
  if tag eq 66 then
        set local-preference 200
        set origin igp
        set next-hop 2001:db8:0:ff::abcf
        set community (no-export)
  else
    drop
  endif
end-policy
```

Solution 6: Remotely Triggered BlackHole Route :: Customer :: IOS

```
ip route 10.10.10.10 255.255.255.255 null0
router bgp 65535
address-family ipv4 unicast
  network 10.10.10.10 mask 255.255.255.255
ip prefix-list BLACKHOLE 10.10.10.10/32
route-map AS10102-OUT permit 1000
  match ip address prefix-list BLACKHOLE
  set community 10102:0
```

Designing Internal Community : Practical consideration

- Most routers parse BGP communities as strings rather than integers, using Regular Expressions.
  - Design your community system with this in mind.
  - Think strings and character positions, not numbers.
  - For Example, 10102:1234 can easily be parsed as
    - Field #1, Value 1
    - Field #2, Value 23
    - Field #3, Value 4
  - But can’t easily be parsed numerically
    - For example as “larger than 1233”.
  - Remember not to exceed 65535 as a 16-bit value. (65536 options) to represent
- Carried across AS

Types of Implementation

- Practical BGP Communities Implementation can essentially be classified into two types:
- Informational tags
    - Communities set by and sent from a provider network, to tell their customers (or other interested parties) something about that route.
- Action tags
    - Communities set by and sent from a customer network, to influence the routing policies of the provider network
    - Alter route attributes on demand
    - Both globally and within own network
    - Control the import/export of routes

Informational tags

- Information communities typically focus on
  - Where the route was learned
    - AKA Geographic data (continent, country, region, city, etc in short geotag)
  - How the route was learned
    - AKA Relationship data (transit, peer, customer, internal, etc)
  - There is no other good way to pass on this data
- This data is then used to make policy decisions
  - Either by you, your customer, or an unknown third party.
  - Exporting this data to the Internet can provide invaluable assistance to third party networks you may never even know about. This is usually a good thing for everyone.

Ways to encode Information

- Encode simple arbitrary data
  - Each network defines its own mapping
    - Which must be published somewhere like ASN description in IRR for others to use
  - Ex: Continent (1 = Asia, 2 = Africa, etc)
  - Ex: Relationship (1 = Transit, 2 = Public Peer, etc)
- Standards based encoding
  - Ex: ISO 3166 encodes Country Codes into 3 digits

Providing information

- As always, the exact design decision depends on specific network and footprint.
- Networks in only a few major cities may want to focus on enumerating those cities in a short list.
- Networks in a great number of cities may want to focus on regional aggregation specific to their scope.
- Plan for the future!
    - Changing community design after it is already being used by customers may prove impossible.

Practical Use of Informational Tags

- Make certain that Informational Tags from your Action Tags can easily be distinguished
- Ex: Make Informational Tags always 5 characters in length, and action tags to be 4 characters or less.
- This allow to easily match Info tags: “10102:.{5}”
- Filter communities from neighbors
    - None is allowed to send Informational tags, these should only be set by Service Provider, and these should be stripped from all BGP neighbors (customers, transits, peers, etc).
    - Otherwise there is a massive security problem.

Providing Information

- For example: 10102:TCCCP
  - T Type of Relationship
  - C Continent Code
  - CC Country Code
  - P POP Code
- The community 10102:21021 could be parsed as:
  - Public Peer
  - Asia
  - Singapore
  - Equinix

Definitions - Types

- Type of routes
  - 1XYYP – Transit/Upstream
  - 2XYYP – Public Peer
  - 3XYYP – Private Peer
  - 4XYYP – Customer
  - 5XYYP – Internal

Definitions (Contd)

- Continents
  - T1YYP – Asia
  - T2YYP – Africa
  - T3YYP – Europe
  - T4YYP – North America
  - T5YYP – South America
  - T6YYP – Australia
  - T7YYP – Antarctica

Definitions (Contd)

- Countries
  - T101P – Bangladesh
  - T102P – Singapore
  - T201P – United Kingdom
  - T202P – France
  - So on ..

Definitions (Contd)

- PoP
  - T1011 – Central NOC
  - T1021 – Singapore Global Switch
  - T1022 – Singapore Equinix
  - T2011 – United Kingdom Telehouse North
  - T3011 – United States TelX
  - So on ..

Providing Access to Action

- Remotely Triggered Blackhole Route
  - 10102:0
- Changing Local Preference
  - 10102:75 (Lower than Transit Routes)
  - 10102:85 (Lower than Peer Routes/Higher than Transit Routes)
  - 10102:95 (Lower than Customer Routes/Higher than Peer Routes)
  - 10102:105 (Higher than Customer Routes)
- Reset MED value
  - 10102:4000 (Resets MED value to 0)
  - Another BGP attribute to prefer Main Link/Backup Link
  - If Local Preference is not a solution

Providing Access to Action(Cont)

- Applying NO_EXPORT (Keeping within AS)
  - 10102:5000
- Applying NO_ADVERTISE (Keeping within Local Router)
  - 10102:6000

Providing Access to Action(Cont)

- Redistribution to Other Peers/Transits
  - 10102:5CTP (Do NOT Redistribute to Transit Provider/Peering ASN Code)
  - 10102:1CTP (Prepend 10102 once)
  - 10102:2CTP (Prepend 10102 twice)
  - 10102:3CTP (Prepend 10102 twice)

Providing Access to Action(Cont)

- Where CTP Stands as follows
  - C – Region or Continent
    - 0 – Globally
    - 1-7 As mentioned in Continents in previous
  - TP – Transit Provider or Peering ASN Code
    - 00 – Globally
    - 01 – TATA Communications (AS6453)
    - 02 – Level3 Communications (AS3356)
    - 03 – Cogent (AS174)
    - 04 – Bharti Airtel (AS9498)
    - etc

Caveats

- RFC 4384 : BGP Communities for Data Collection (BCP 114)
- Uses a similar approach but in bit levels
- For community matching we have to consider numerical value rather than strings (Only IOS-XR and Junos supports matching numerical Community Ranges)
- No space for Action communities unless we use extended communities
- Can point only upto Country Level Geolocations not city or PoP like this

References

1. Using Communities for Multihoming (http://bgp4all.com/ftp/isp-workshops/BGP%20Presentations/09-BGP-Communities.pdf)
2. BGP Techniques for Internet Service Providers – Philip Smith
3. BGP Communities: A guide for Service Providers – Richard A. Steenbergen & Tom Scholl