BGP勉強会

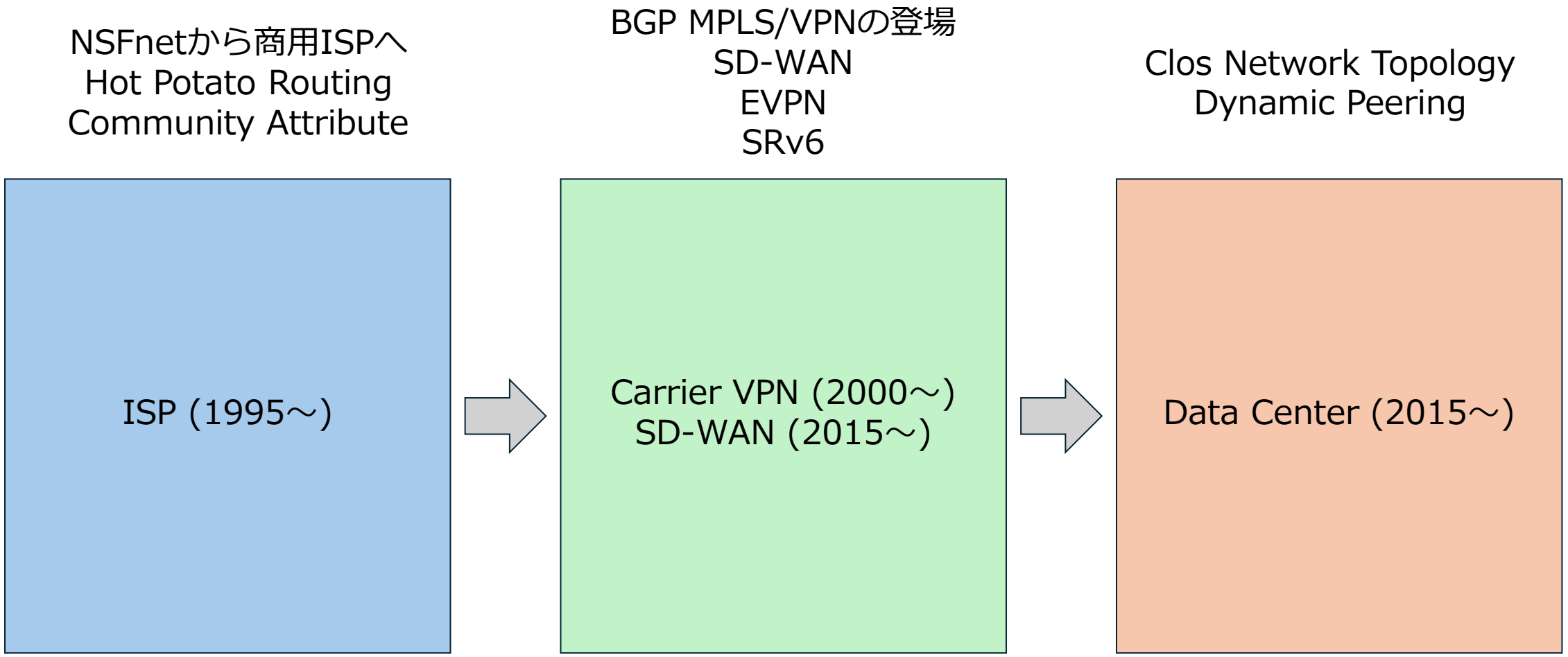
2024/7/25

石黒邦宏

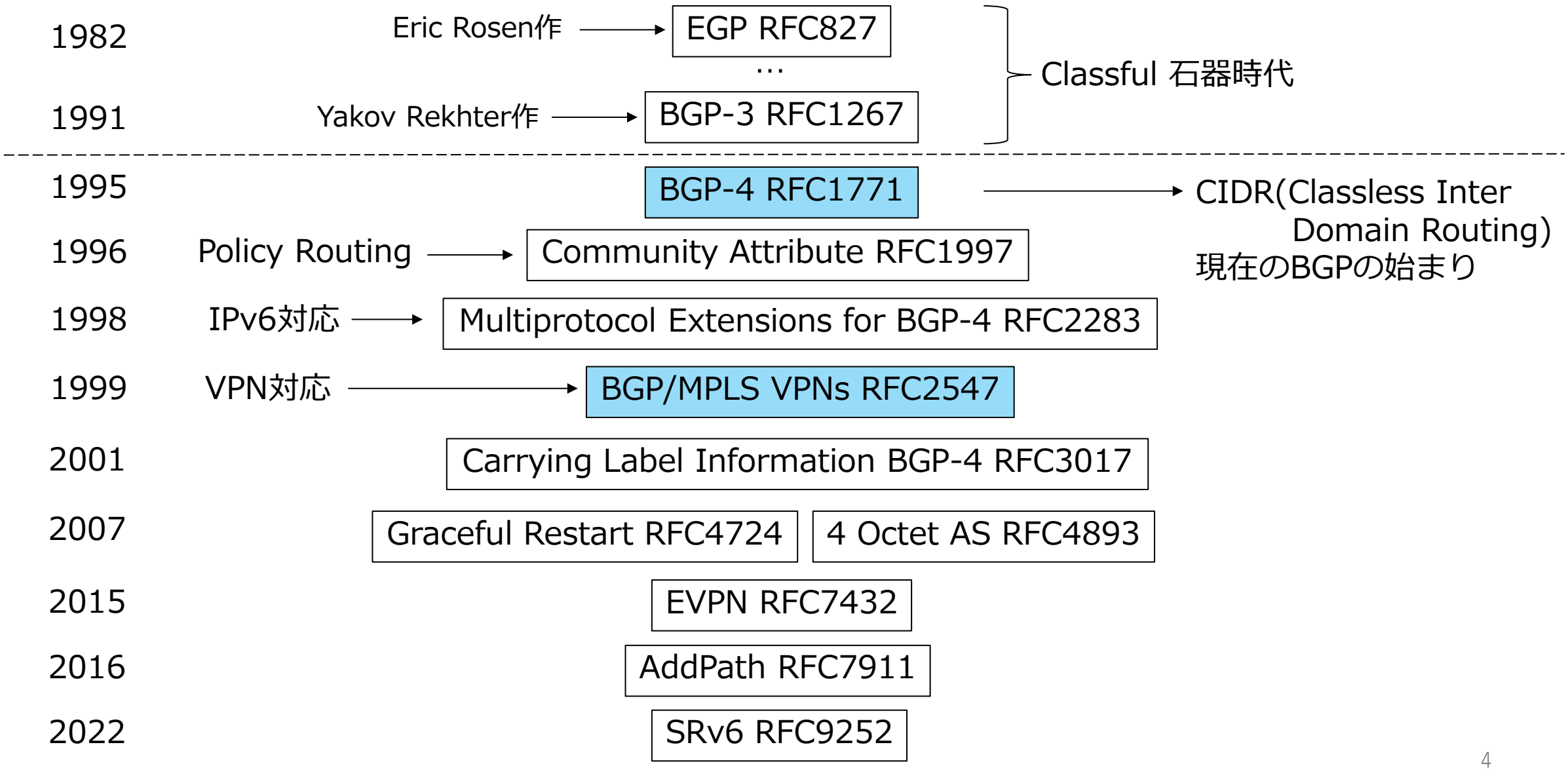
逆出世魚BGP

- 同じ魚なのに成長すると名前が変わる魚を出世魚と呼びますが
  - BGPの場合機能が進化してほぼ別物になっていても常に「BGP」
- BGPの拡張性の高さからなんでも入れられる”Kitchen Sink Protocol”と呼ばれることも
  - EGPの一つから、VPNサービスへ、そしてSD-WANでアプリケーションコントローラへ
- 運用で初めて見つかった課題を後づけで改良してきた歴史がある
  - BGPの特定の機能を理解するためには運用の課題を理解する必要がある
- 顧客へ解決策を提案するためには、広い範囲でのBGPの理解が必要
  - セグメント化が進んでいるため、隣接セグメントについてはほとんど知らないということも

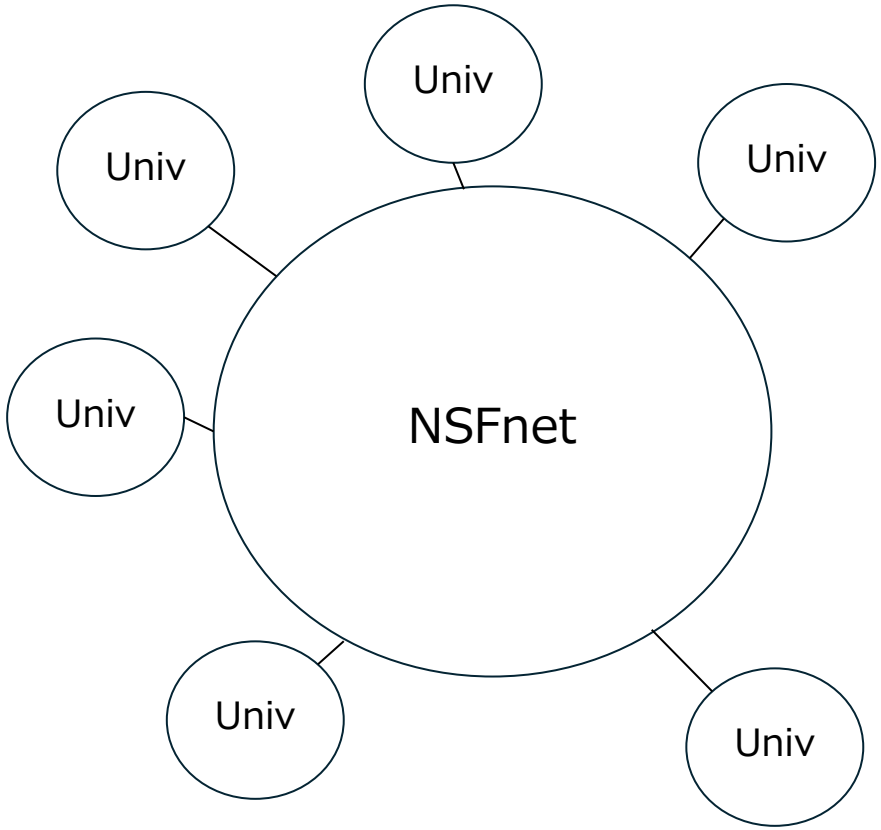
BGPのユースケースの進化

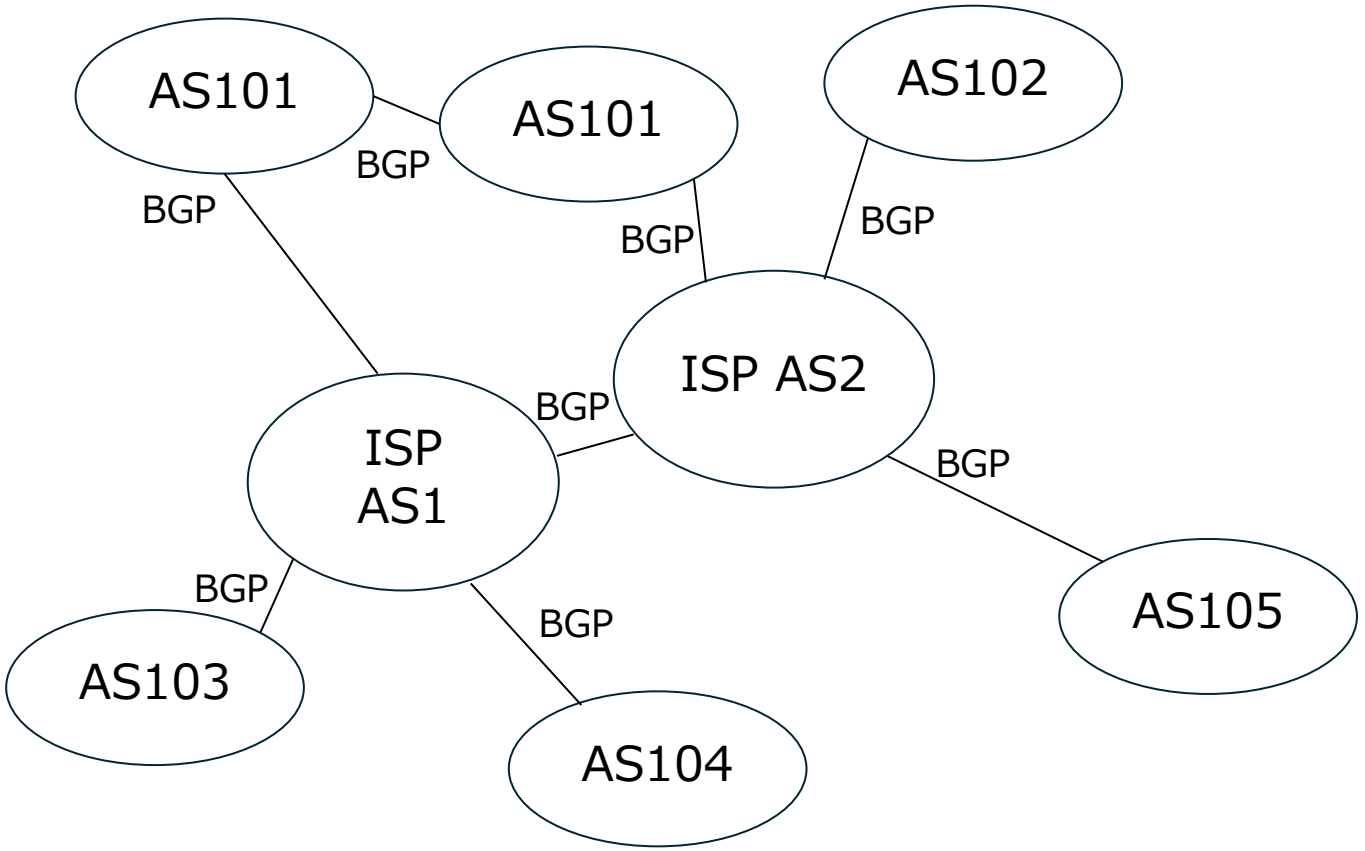


BGPの歴史


1982
Eric Rosen作
EGP RFC827
…
1991
Yakov Rekhter作
BGP-3 RFC1267
Classful 石器時代
1995
BGP-4 RFC1771
CIDR(Classless Inter Domain Routing)
現在のBGPの始まり
1996
Policy Routing
Community Attribute RFC1997
1998
IPv6対応
Multiprotocol Extensions for BGP-4 RFC2283
1999
VPN対応
BGP/MPLS VPNs RFC2547
2001
Carrying Label Information BGP-4 RFC3017
2007
Graceful Restart RFC4724
4 Octet AS RFC4893
2015
EVPN RFC7432
2016
AddPath RFC7911
2022
SRv6 RFC9252

ISPの誕生とHot Potato Routing





- 90年代初頭に商用ISPの誕生とともにトランジットネットワークという概念が生まれた
- 基本ISP宛のトラフィックがないため、受け取ったトラフィックはできるだけ早く外にだしたい
- そのことを指して、熱いイモを渡されたらすぐに次の人に渡したいよね？という意味を込めて Hot Potato Routingと呼ぶ
- ISP間の経路交換を目的につくられたプロトコルがBGP

BGP Peer Establishmentへの流れ

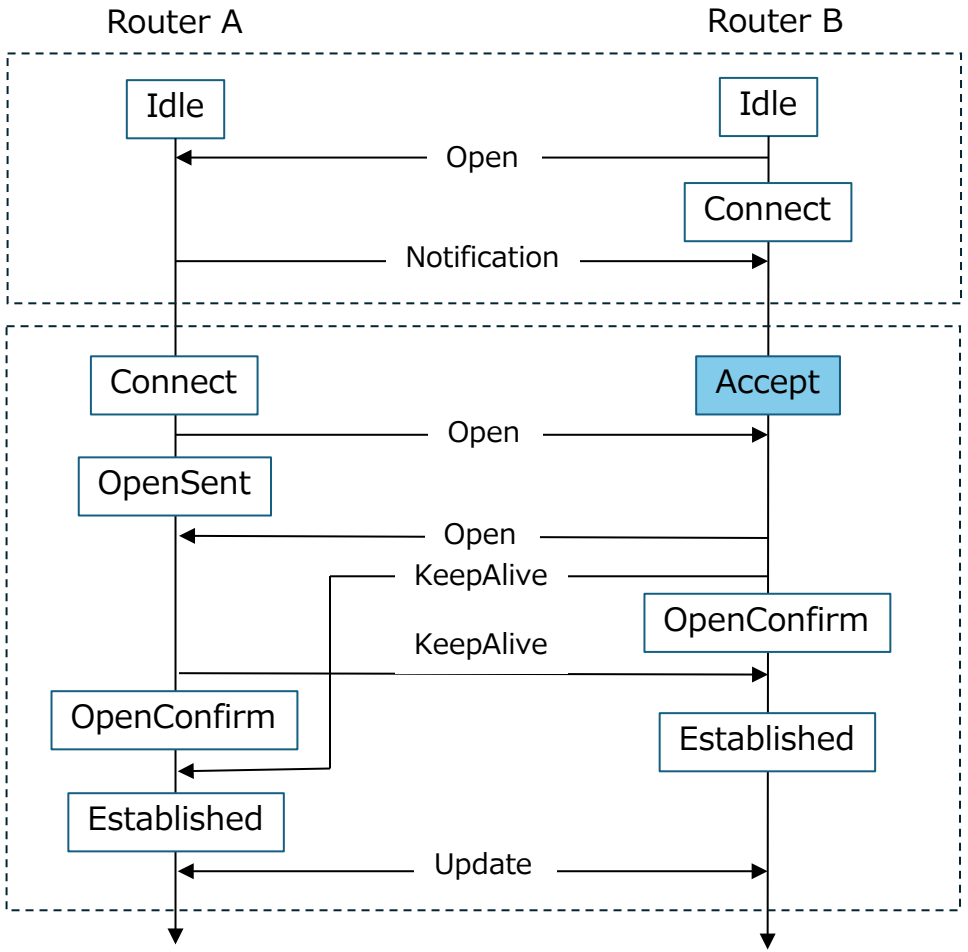
- TCPのPort番号179で接続
- ルーターのどちらも側もTCPのServerになれるしClientにもなれる(HTTP,Telnetとは違う)
- BGPのPeerがEstablishedになったあとはTCPのServer/Clientの区別はない
- 同時にPeerのセッションが確立される可能性があるためそれを解決するCollision Detectというメカニズムがある

BGP State

Idle
Connect
Accept
OpenSent
OpenConfirm
Established

BGP Message Type

Value	Name	Reference
0	Reserved	
1	OPEN	[RFC4271]
2	UPDATE	[RFC4271]
3	NOTIFICATION	[RFC4271]
4	KEEPALIVE	[RFC4271]
5	ROUTE-REFRESH	[RFC2918]
6	DYNAMIC-CAPABILITY	[draft-ietf-idr-dynamic-cap-16]
7-255	Unassigned	

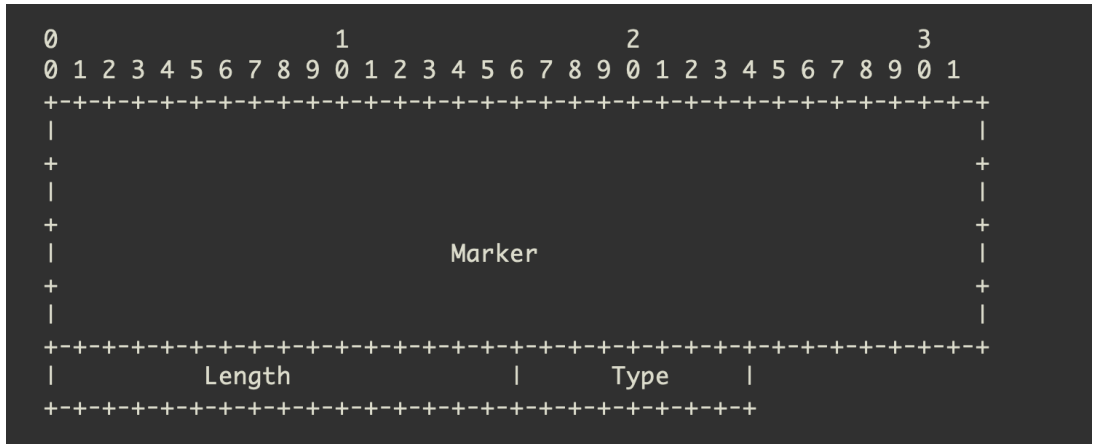


どちらかのルーターがAcceptになる必要がある

Acceptに対してConnectを行う

AcceptでOpenを受信したらOpenとKeepAliveを連続して送る

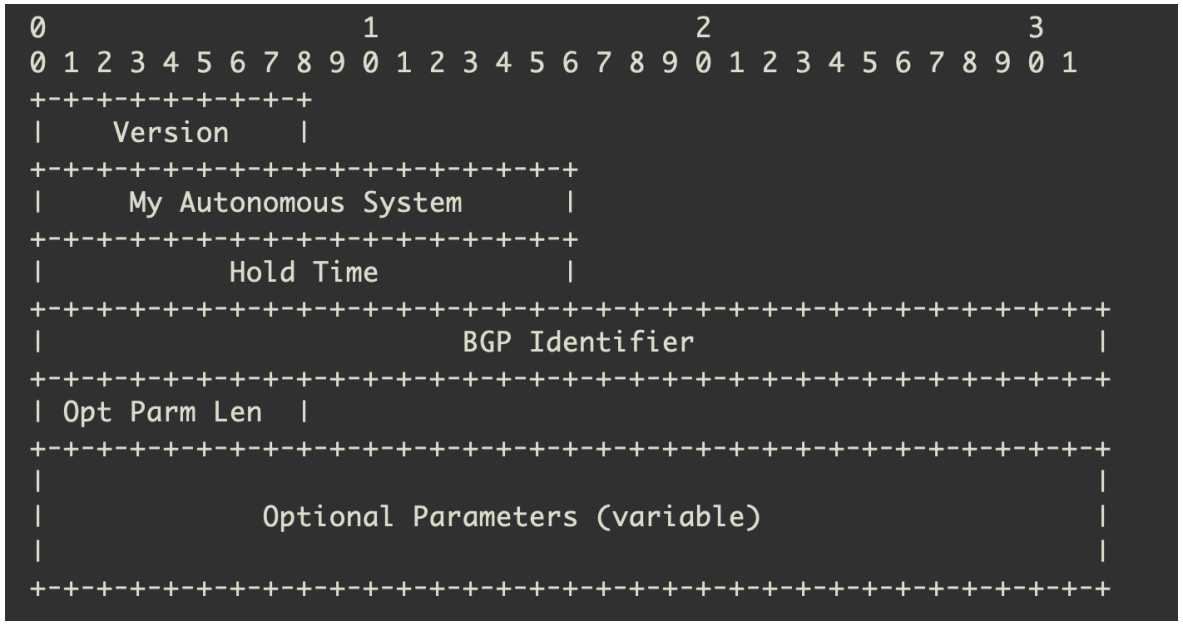
BGP Packet Format



Marker: 16byteで0xFFで埋める、当初は認証に使おうという野望があったが、野望は果たせず
　　世界遺産に
Length: ヘッダーも含めたパケット長。最小19byte、最大4096byte
Type: BGP Packet Type

```
    12 1.160773    10.211.55.65    10.211.55.2     BGP      85 KEEPALIVE Message
    13 1.160866    10.211.55.2     10.211.55.65    TCP      66 62963 → 179 [ACK] Seq=81 Ack=150 Win=131584 Len=0 TSval=3692709829 TSecr=713551874
    14 2.265685    10.211.55.65    10.211.55.2     BGP     141 UPDATE Message
    15 2.265776    10.211.55.2     10.211.55.65    TCP      66 62963 → 179 [ACK] Seq=81 Ack=225 Win=131520 Len=0 TSval=3692710934 TSecr=713552978

> Frame 12: 85 bytes on wire (680 bits), 85 bytes captured (680 bits) on interface bri
> Ethernet II, Src: Parallels_b6:61:4d (00:1c:42:b6:61:4d), Dst: 5e:e9:1e:08:4d:64 (5e
> Internet Protocol Version 4, Src: 10.211.55.65, Dst: 10.211.55.2
> Transmission Control Protocol, Src Port: 179, Dst Port: 62963, Seq: 131, Ack: 81, Le
v Border Gateway Protocol - KEEPALIVE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 19
    Type: KEEPALIVE Message (4)

0000  5e e9 1e 08 4d 64 00 1c  42 b6 61 4d 08 00 45 c2   ^···Md·· B·aM··E·
0010  00 47 74 1a 40 00 01 06  80 ec 0a d3 37 41 0a d3   ·Gt·@··· ····7A··
0020  37 02 00 b3 f5 f3 50 21  c1 29 35 7a 92 56 80 18   7·····P! ·)5z·V··
0030  01 fd de 7e 00 00 01 01  08 0a 2a 87 f0 02 dc 1a   ···~···· ··*·····
0040  47 c3 ff ff ff ff ff ff  ff ff ff ff ff ff ff ff   G······ ········
0050  ff ff 00 13 04                                    ·····
```

BGP Open Message

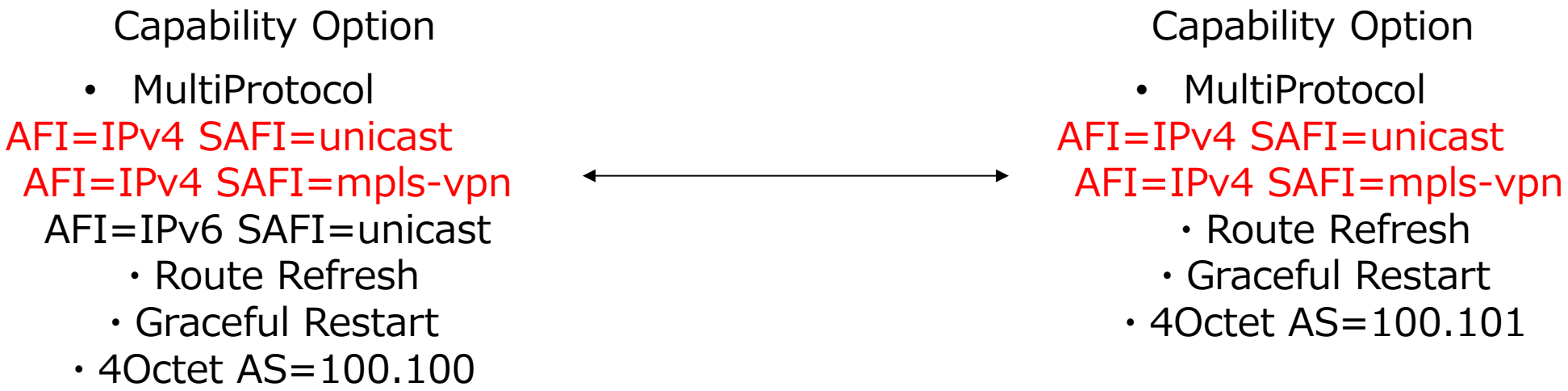


Version: 4に固定
My AS: 自AS番号 – 4Octet拡張時はCapabilityにAS番号を指定
Hold Time: Keepaliveがexpireする時間
BGP Identifier: 自ルーターID
Opt Parm: オプショナルパラメータ、認証に使うという野望もあったが、結局Capabilityだけが使われている

BGP Open Message

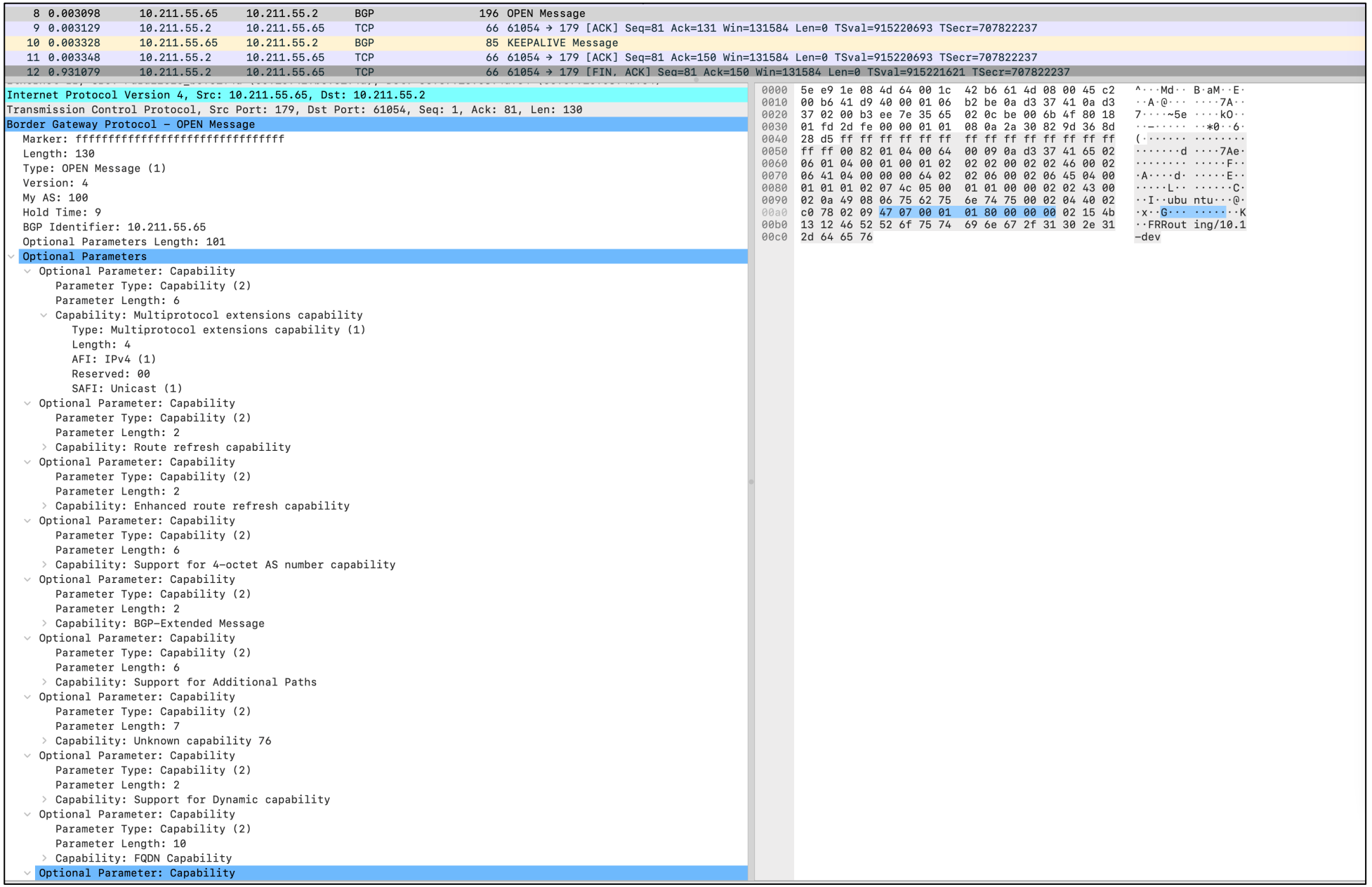
Openメッセージは相互に送りつけて確認と共通部分の認識を行う(ネゴーシエーションはない)

AS番号,Router IDの確認
HoldTimeはどちらか少ないほうを採用
HoldTime=0はKeepalive無し
HoldTime>=3はHoldTime/3がKeepaliveタイマー

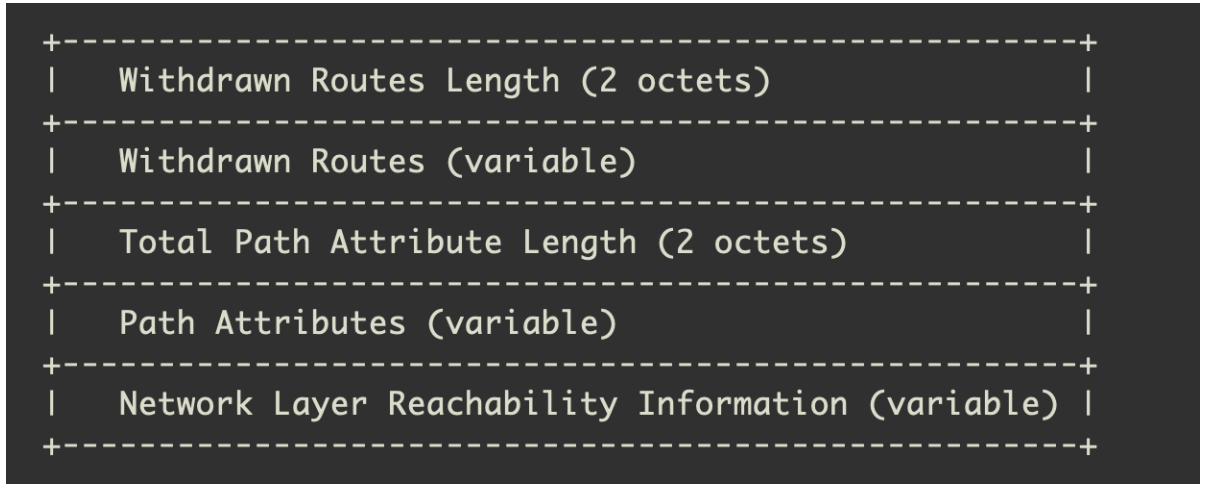


MultiProtocolはsend/receiveの
両方がある場合に有効

BGP Open Message Capability



BGP Update Message



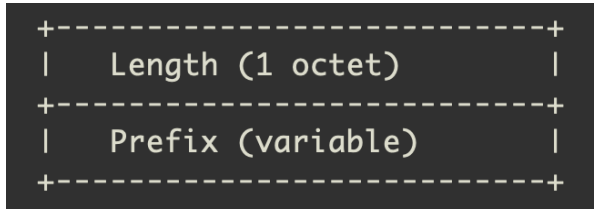
経路削除の場合

ひたすら削除する経路をNLRIのフォーマットでWithdrawn Routesに入れて送る
Path Attributesは無し

経路更新の場合

ひたすら更新する経路をNLRIのフォーマットでNLRIに入れて送る
Path Attributesは必須

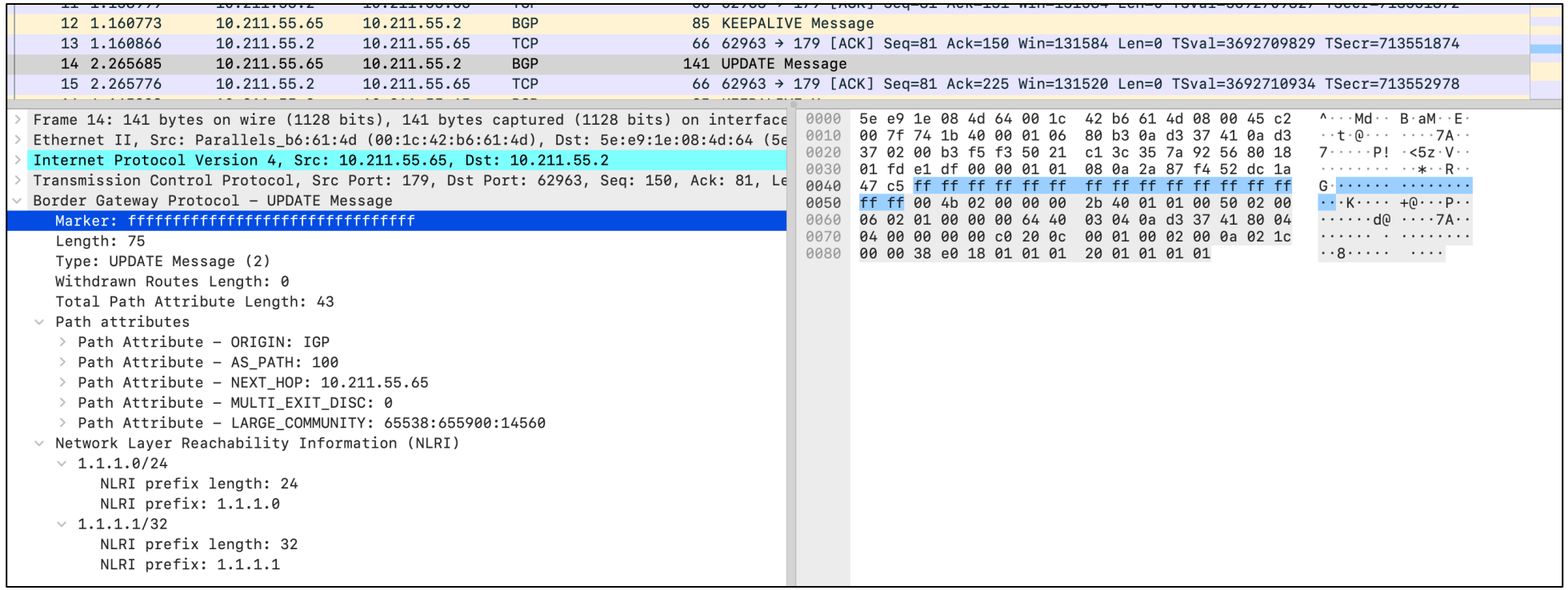
NLRIのフォーマット



IPv4,IPv6,VPN関係なくPrefix LengthとPrefixを並べる形
PrefixはLengthに必要なサイズにTrimされる
0.0.0.0/0の場合Length=0, Prefix無し
10.0.0.0/8の場合Length=8, Prefix(1byte)=10
192.168.0.0/24の場合Length=24, Prefix(3byte)=192,168,0

BGP Attribute

1 ORIGIN well-known mandatory
2 AS_PATH well-known mandatory
3 NEXT_HOP well-known mandatory
4 MED optional non-transitive
5 LOCAL_PREF well-known
6 ATOMIC_AGGREGATE well-known discretionary
7 AGGREGATOR optional transitive
8 COMMUNITIES optional transitive



BGP Path Selection

NexthopがReachableな経路のうち

1. Cisco独自のWEIGHTアトリビュートが最も大きいルートを優先
2. LOCAL_PREFアトリビュートが最も大きいルートを優先
3. ローカルルートが発生元であるルート（networkコマンドで生成したルート）を優先
4. AS_PATHアトリビュートが最も短いルートを優先
5. ORIGINアトリビュートが最も小さいルートを優先（ IGP < EGP < Incomplete ）
6. MED（MULTI_EXIT_DISC）アトリビュートが最も小さいルートを優先
7. IBGPで学習したルートよりもEBGPで学習したルートを優先
8. Nexthopに対して最小のIGPメトリックを持つルートを優先
9. EBGPネイバーから受信したルートのうち、最も古いルート（先に受信したルート）を優先
10. Router IDが最小のBGPピアから受信したルートを優先
11. BGPピアのIPアドレスが最小のルートを優先

IBGPとEBGPの違い

- 同じAS間の接続をIBGP、異なったAS間の接続をEBGPと呼ぶ
- 一番大きな違いはNexthopをResolveするかどうか？
- EBGPの場合基本Immediate Nexthopのみ使用、BGPのTCPもTTL=1で張る
- IBGPの場合NexthopをResolveすることにより柔軟なネットワーク設計が可能に

BGPに必要なStatic Routeの話

- Floating Nexthop(異なったメトリックのStaticがNexthopの有効性により選択される)
  - S 10.0.0.0/24 1 1.1.1.1
  - S 10.0.0.0/24 100 2.2.2.2
  - C 1.1.1.1/24 Down
  - C 2.2.2.2/24 Up
- Recursive Nexthop
  - 経路を再帰的にLookUpする
- Aggregate経路をNull 0 255にする理由
  - Discard経路 –マッチしたパケットが破棄される

BGPで一番重要なIBGP – Next Hopの解決と抽象化

Connected Nexthop

```
B 192.168.0.0/24 10.0.0.2
C 10.0.0.0/24    GigabitEthernet 0/0
```

Recursive Nexthop

```
B 192.168.0.0/24 172.0.0.2
O 172.0.0.2/32   10.0.0.2
C 10.0.0.0/24    GigabitEthernet 0/0
```

Recursive Nexthop

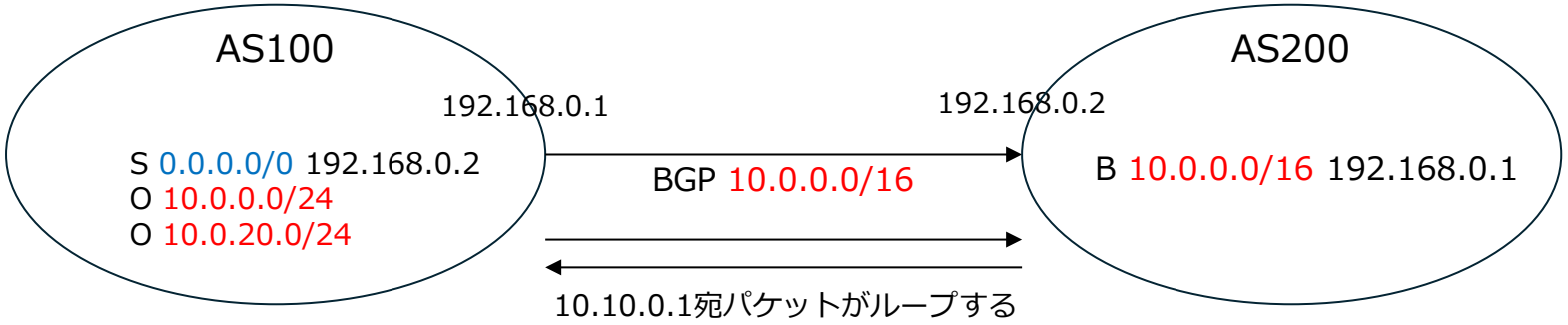
```
B 192.168.0.0/24 172.0.0.2
S 172.0.0.2/32   172.100.0.1
S 172.100.0.0/24 172.0.0.2
```

無限Recursive

Recursive Nexthop
(Resolve to LSP)

```
B 192.168.0.0/24 172.0.0.2
L 172.0.0.2/32   Push 100
O 172.0.0.2/32   10.0.0.2
C 10.0.0.0/24    GigabitEthernet 0/0
```

Aggregate(経路集約)と5秒で作れる経路ループ



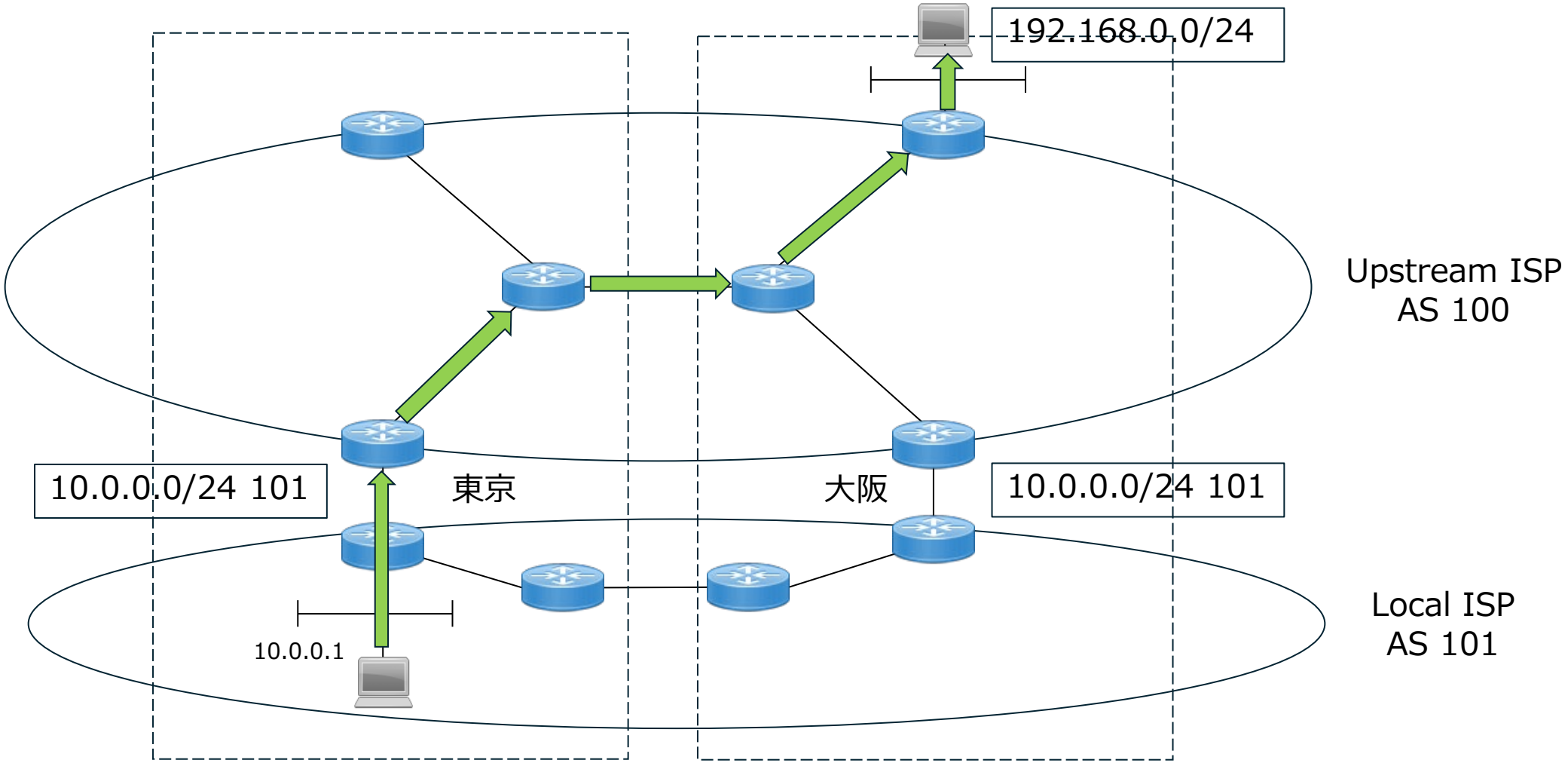
- AS100は10.0.0.0/24と10.0.20.0/24を集約(Aggregate)して10.0.0.0/16をAS200に公告する
- 同時にAS100はStatic RouteでDefault RouteをAS200に向けたとする
- IGPに存在しない経路宛10.10.0.1のパケットをAS100から出すと
  - Default RouteによりAS200に転送される
  - AS200ではBGP経路にマッチするのでAS100に転送される
  - これを繰り返す
  - めでたく経路ループの完成！
- Ciscoの場合これを防ぐために以下のStatic Routeを設定するのがBest Current Practiceとされている

```
ip route 10.0.0.0 255.255.0.0 Null0 255
```

- CiscoのEIGRPというIGPプロトコルではsummary-route (BGPのaggregateと同じ)設定をすると自動的にNull0宛経路が作成される
- Juniperでは以下の設定で自動的にnexthopがrejectの経路が作成される(nexthopをdiscardに設定も可能)

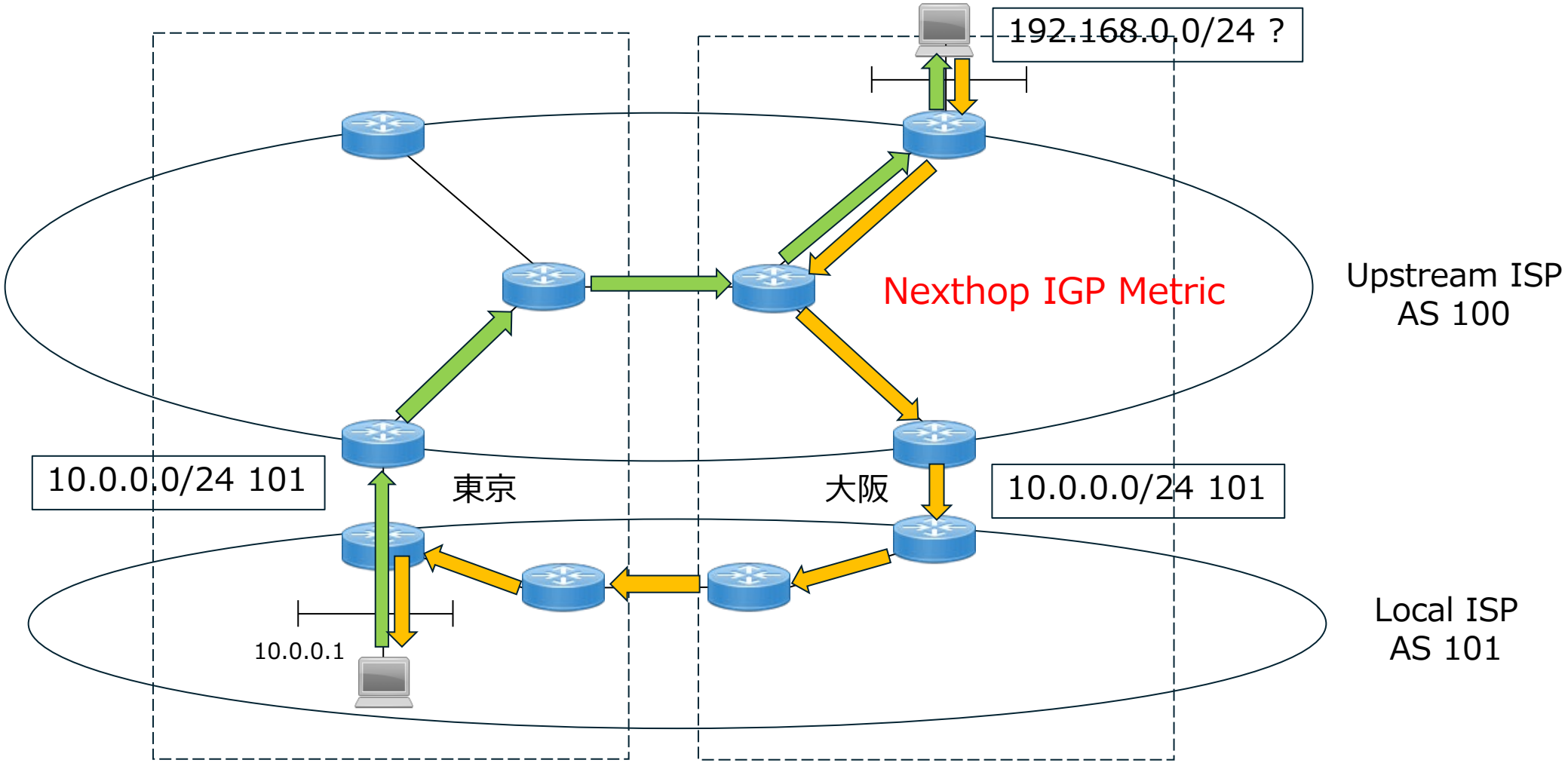
```
routing-options {
  aggregate {
    route 10.10.0.0/16;
  }
}
```

BGPで接続時に戻りのトラフィックをコントロールしたい



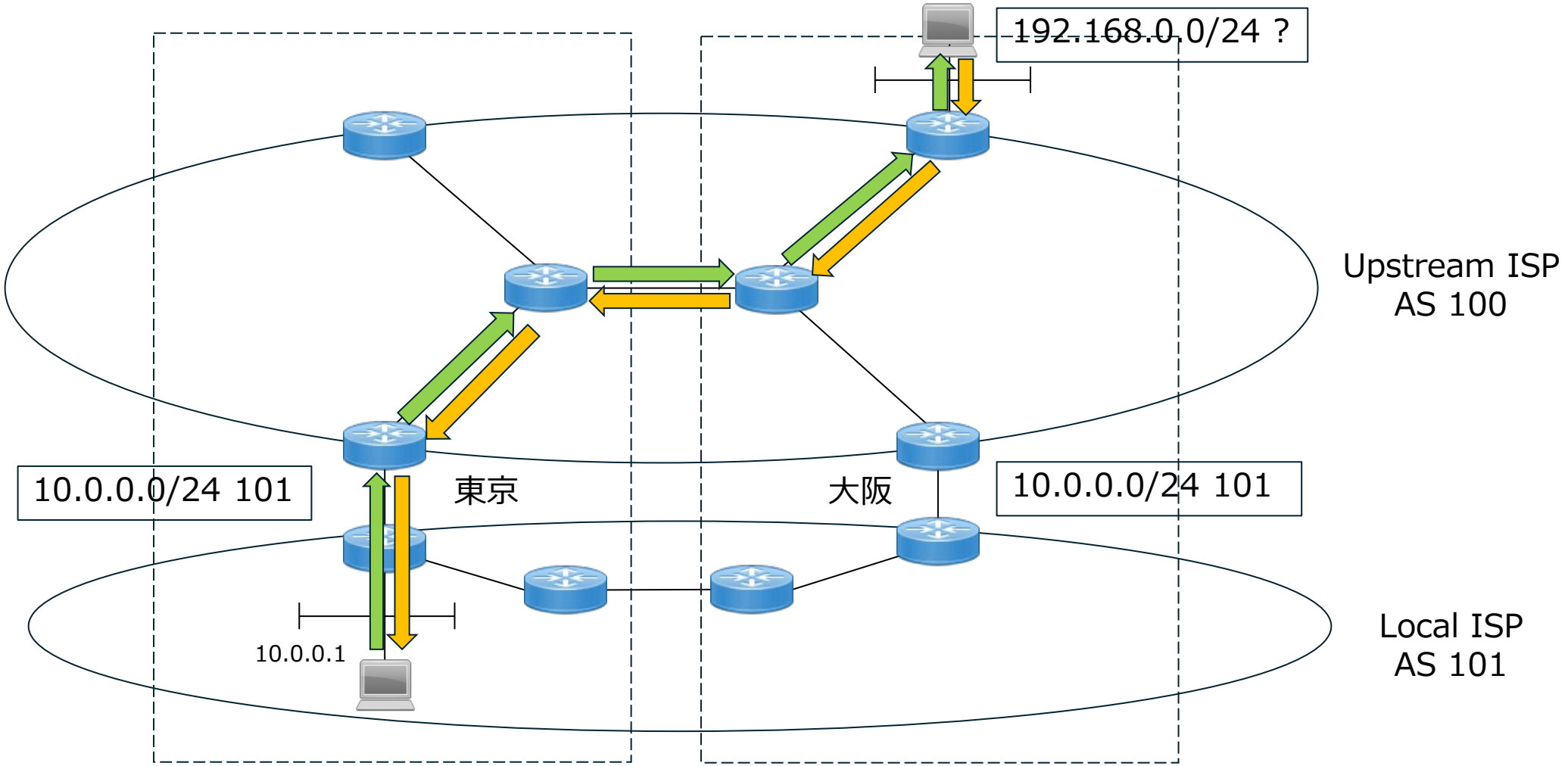
- Upstream ISPとは東京と大阪の２箇所で接続しており同じく10.0.0.0/24を公告しているとする
- 東京の10.0.0.0/24から大阪の192.168.0.0/24への通信を行なった場合
- 行きはコントロールできるが帰りはUpstream ISP次第

Hot Potato Routingの場合(一番近い出口から外に出す)



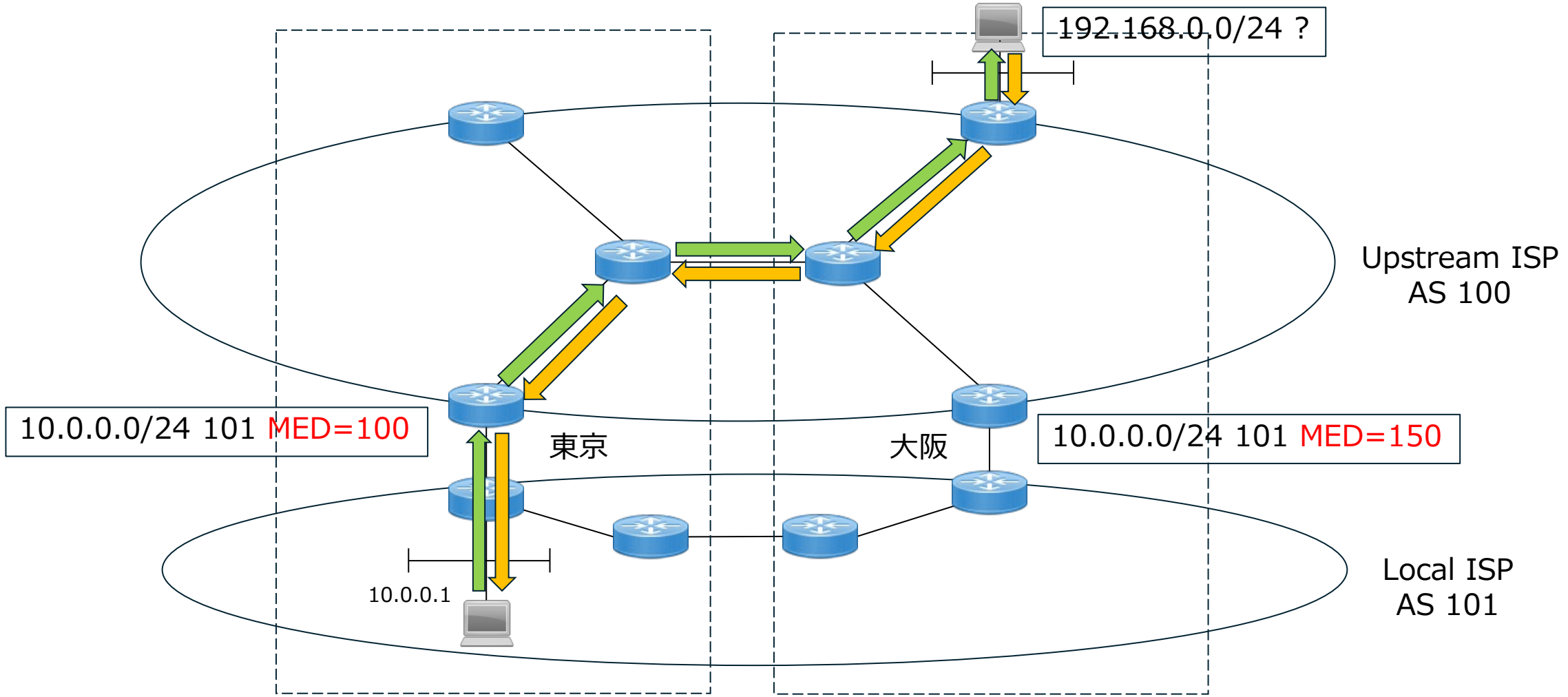
- Upstream ISPは一番近い出口(Closest Exit)からパケットを外に出そうとする
- 自己資源をできるだけ使わないような経路制御
- 戻りのパケットが大阪のルーターに戻ってくる可能性が高いがこれはLocal ISPとしては望ましくない

東京のお客さんのトラフィックは東京のルーターから戻ってきて欲しい



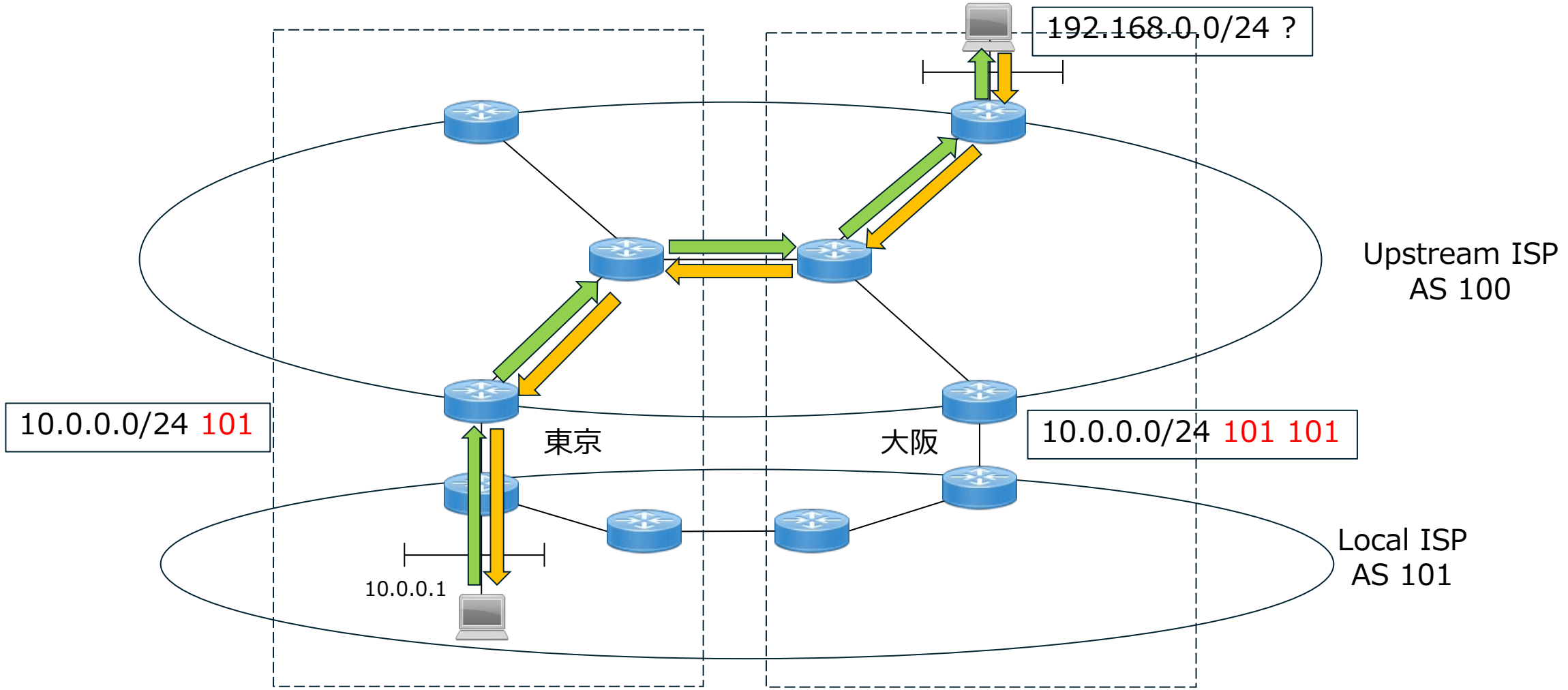
- 複数接続ポイントがある場合に戻りのトラフィックをコントロールしたい
- 上の図の場合10.0.0.0/24のパケットは東京のルーターから戻ってきて欲しい
- 北米だとロスアンジェルスのトラフィックがニューヨークから戻ってきたりする

MEDを使う場合



- MEDをつけて公告することにより、少ないMEDの値の経路が優先される
- しかしMEDはNon-TransitiveなのでASをまたがっては公告されない
- Upstream ISPでMEDを指定しているとうまく動かないが、今でも有効な手法である

AS Path Prepend



- 優先度を低くしたい経路に自ASを追加して公告する
- ASをまたがって公告されるが。。。
- いろいろ問題があることが知られている

Community Attributeの歴史

- AS#:LOCAL_PREFで、特定のAS内のLOCAL_PREFの値を設定できる
- Community Attribute 174:120をつけることでAS174のLOCAL_PREFを120に設定可能
- Transitive Attributeなので直接接続していないネットワークのコントロールができる

Services/Features

Local Preference

All customer routes announced to Cogent will have a local pref of 130.

The customer can control the local preference for their announcements by using a community string that is passed to Cogent in the BGP session. The following table lists the community strings and the corresponding local preference that will be set when they are used.

Community String	Local Pref	Effect
174:10	10	Set customer route local preference to 10 (below everything-least preferred)
174:70	70	Set customer route local preference to 70 (below peers)
174:120	120	Set customer route local preference to 120 (below customer default)
174:125	125	Set customer route local preference to 125 (below customer default)
174:135	135	Set customer route local preference to 135 (above customer default)
174:140	140	Set customer route local preference to 140 (above customer default)

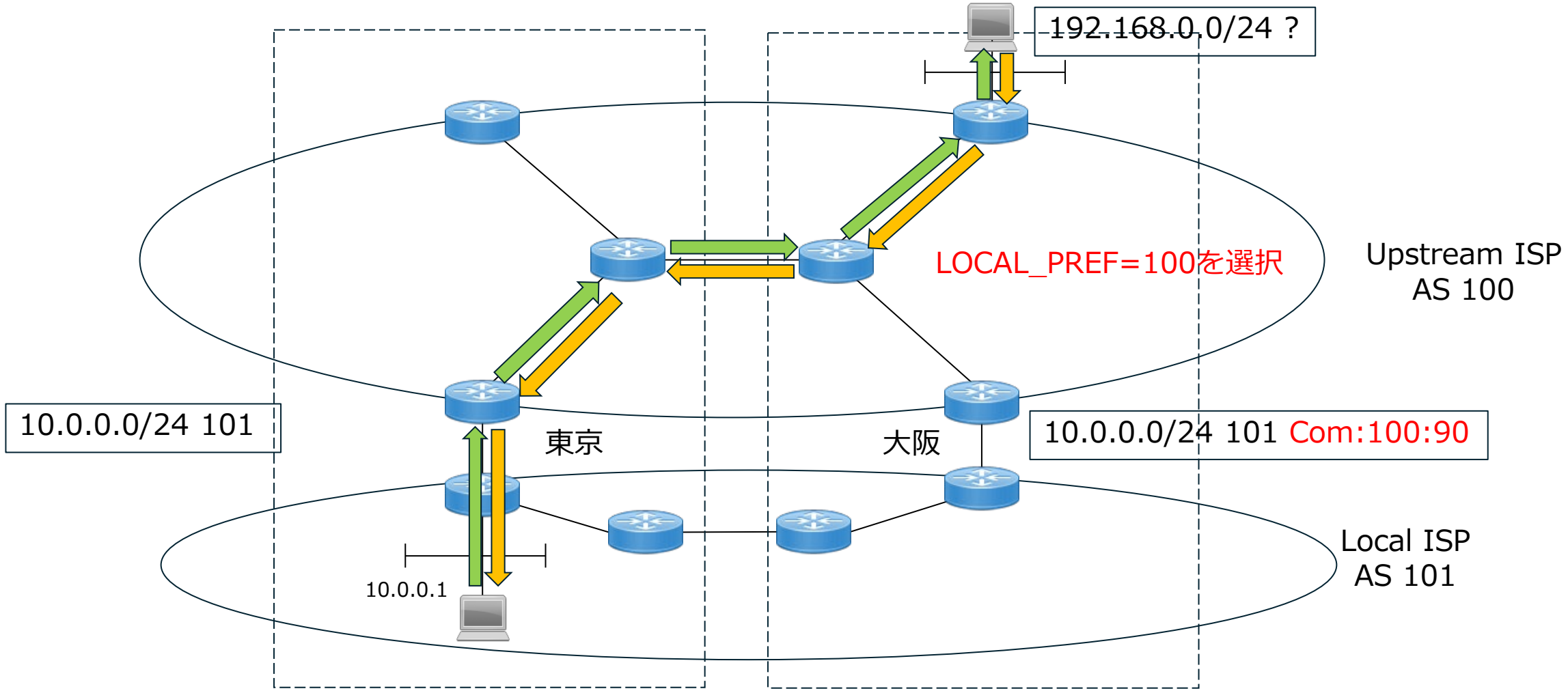
Community Attributeの設定

何もしなくて動くわけではなく明示的な設定が必要

```
route-map community permit 10
  match community 174:80
  set local-preference 80
route-map community permit 20
  match community 174:90
  set local-preference 90
```

1995年～2000年はLOCAL_PREF=80, 90の設定が主流(Default=100)
IX以外のPrivate PeerやPaid Peerが増加することによりLOCAL_PREF=110, 120, 130の設定が登場

まともな戻り経路のコントロールがしたい – Community Attributeの誕生



- Upstream AS:Local PreferenceのCommunityをつけて公告
- 100:90はAS=100の内部でLocal Preferenceを90に設定
- Defaultが100なので、このCommunityがついた経路は優先度が下がる
- CommunityはTransitiveなのでASをまたいで公告される

Community AttributeとDPA

Communityと同時期の提案 AS#, DPA value でpreferenceを指定というCommunityと同じ内容

INTERNET-DRAFT
<draft-ietf-idr-bgp-dpa-05.txt>
Expires September 1996

Enke Chen
Tony Bates
MCI
March 1996

Destination Preference Attribute for BGP
<draft-ietf-idr-bgp-dpa-05.txt>

Destination Preference Attribute (DPA)

This document proposes the DPA path attribute, which is an optional transitive attribute of fixed length. The attribute is represented by a pair <AS#, DPA value>. The AS# is a two octet non-negative integer, which denotes the AS that specifies the preference. The DPA value is a four octet non-negative integer.

The DPA attribute has Type Code 11.

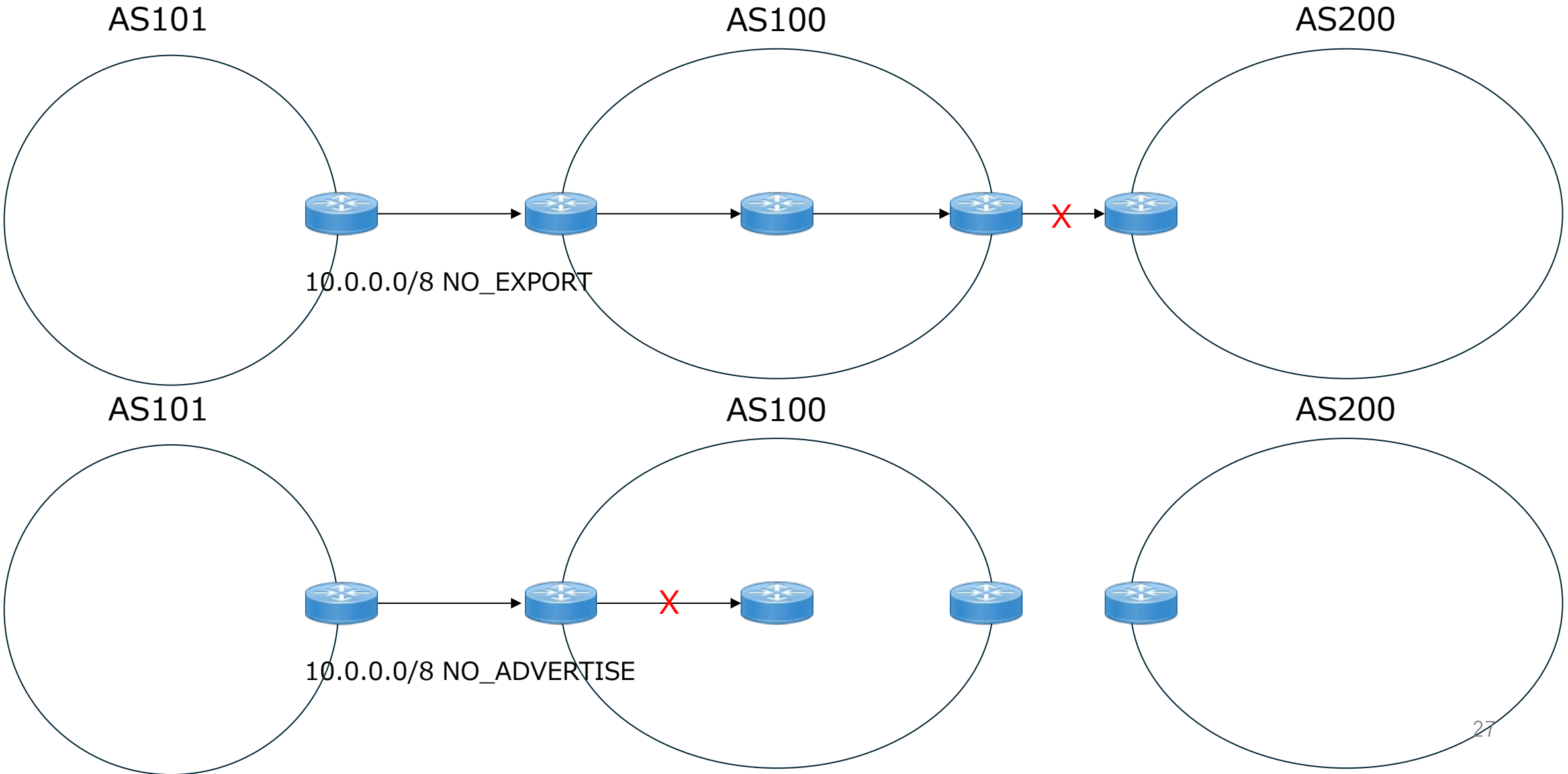
Community Attributeのアイデア

DPAに勝った理由

```
Well-known Communities

   The following communities have global significance and their
   operations shall be implemented in any community-attribute-aware BGP
   speaker.

      NO_EXPORT (0xFFFFFF01)
         All routes received carrying a communities attribute
         containing this value MUST NOT be advertised outside a BGP
         confederation boundary (a stand-alone autonomous system that
         is not part of a confederation should be considered a
         confederation itself).
      NO_ADVERTISE (0xFFFFFF02)
         All routes received carrying a communities attribute
         containing this value MUST NOT be advertised to other BGP
         peers.
      NO_EXPORT_SUBCONFED (0xFFFFFF03)
         All routes received carrying a communities attribute
         containing this value MUST NOT be advertised to external BGP
         peers (this includes peers in other members autonomous
         systems inside a BGP confederation).
```

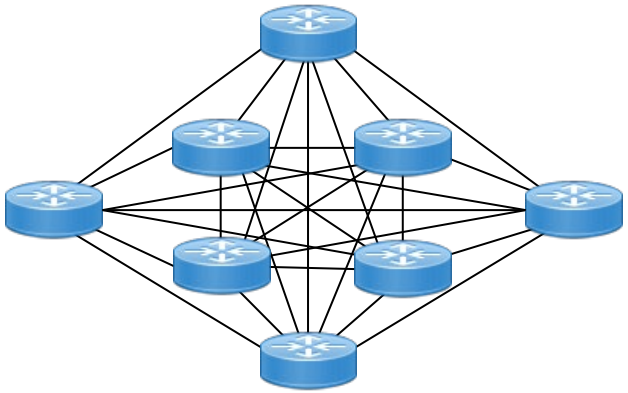
NO_EXPORTとNO_ADVERTISE

特別な設定なしで動作する



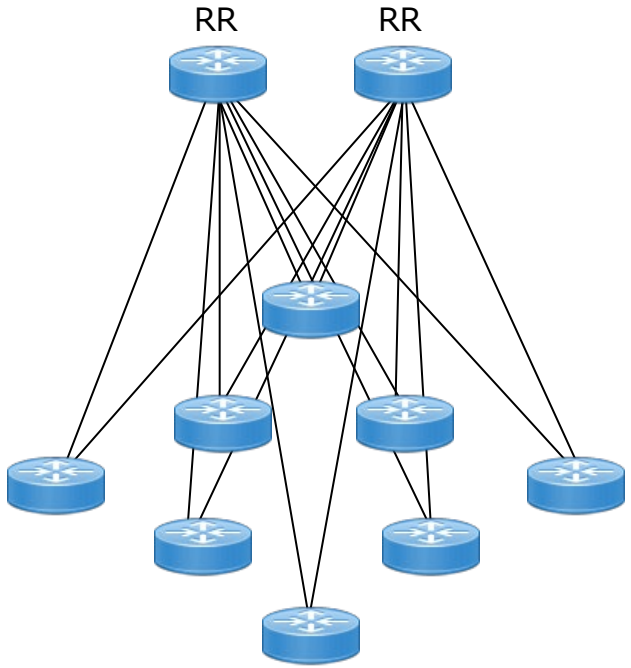
Route Reflectorの登場

- BGP Speakerは全員Full MeshのIBGP peeringが必要
- N*(N-1)/2のBGP Peeringが必要になる
- IBGP Sessionの爆発
- IBGP経路の交換をするためのBGPとしてRoute Reflectorが登場した



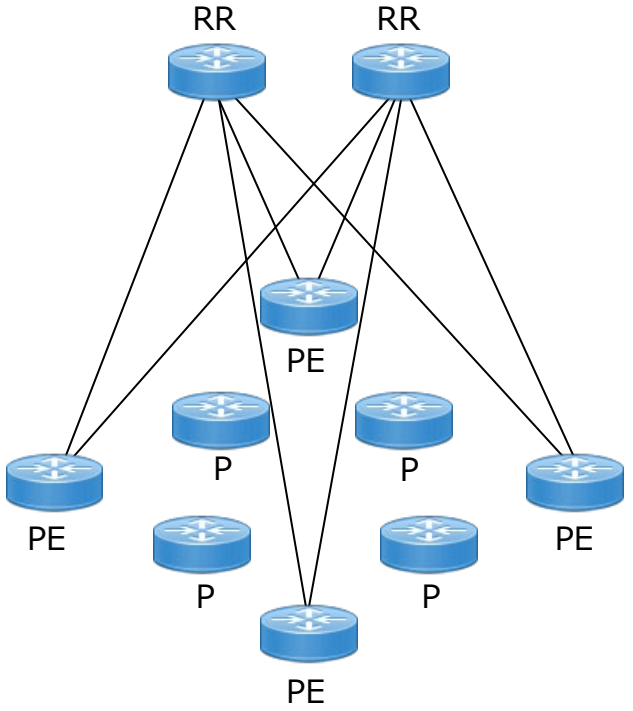
IBGP Full Mesh
N*(N-1)/2

8*7/2 = 28
100*99/2 = 4950



IBGP with 2RR
N*2

8*2 = 16
100*2 = 200

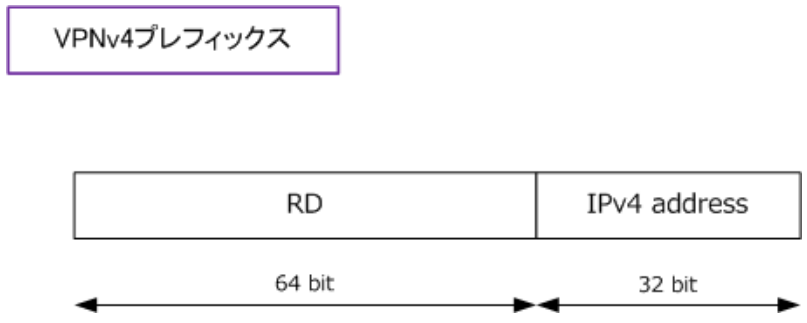


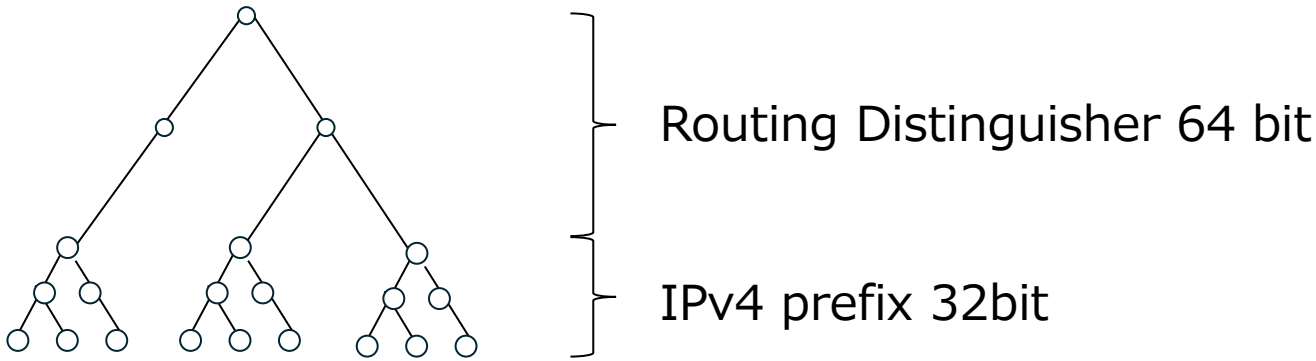
PE with 2RR
PE*2

4*2 = 8
50?*2 = 100

RFC2547 – BGP/MPLS VPNs

- EGPのEric RosenとBGPのYakov Rekhterの1999年の作品
- ルーティングプロトコル設計の傑作
- 顧客プレフィックスを巨大なツリーの一部として嵌め込むという斬新なアイデア
- Community Attributeの技法をExtended Community Attributeとして進化させた
- MPLSへのNext Hop ResolutionをPayloadの仮想化に使えるというアイデアを初めて示した
- MP-BGP, MPLSといった(当時の)最新の技術をフル活用した
- VPNサービスという新たなユースケースを生み出しその後のSD-WAN,EVPNに繋がった
- RD – Routing Distinguisher, RT – Route Target, SOO – Site of Originという新概念

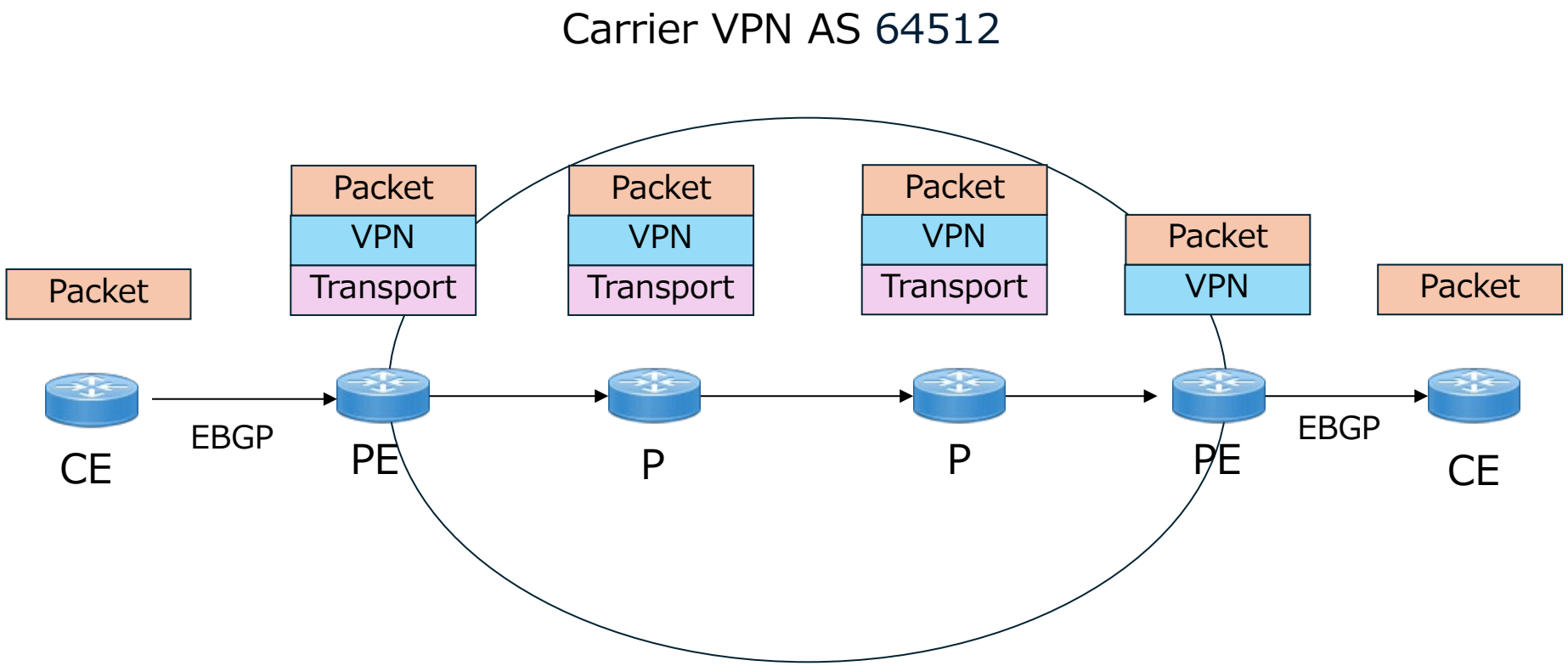




MPLSのラベルスタックとNexthopのLSPへの解決

Network	Labels	Next Hop	AS_PATH
1014:17635:10.0.0.0/9	[100]	198.19.7.97	65000

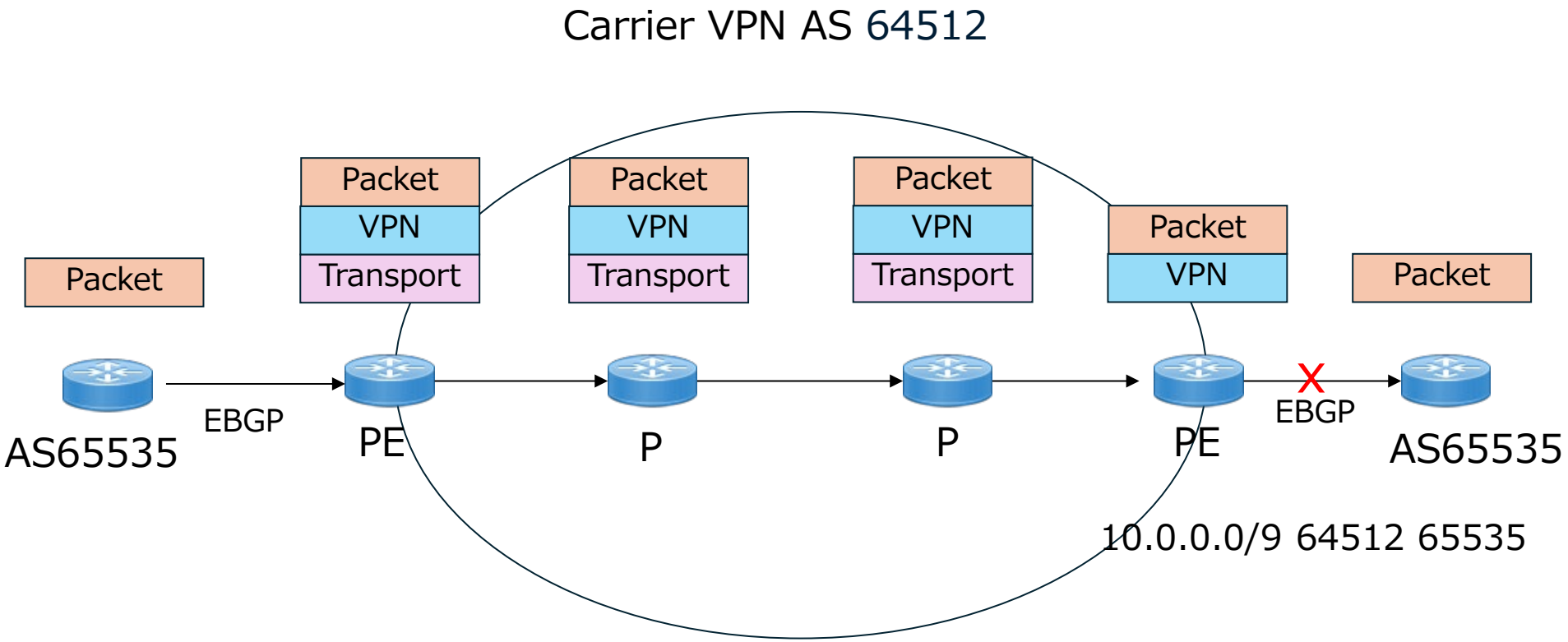
BGPで受け取ったVPN LabelをPushしたのちNexthopでTransport LSPをLookupする



PルーターはTransportを解釈できれば十分なので、BGPをしゃべる必要さえない

Allow-AS-InとAS Override

顧客が同じAS番号を他拠点で使う場合AS PathのLoopチェックにひっかかる
➔ Allow-AS-InでAS Path Loopのチェックを緩和する
➔ AS Overrideで65535を別のAS番号に置き換える



Loopが怖いので、SOO=Site Of OriginというCommunityで別途Loopのチェックができる

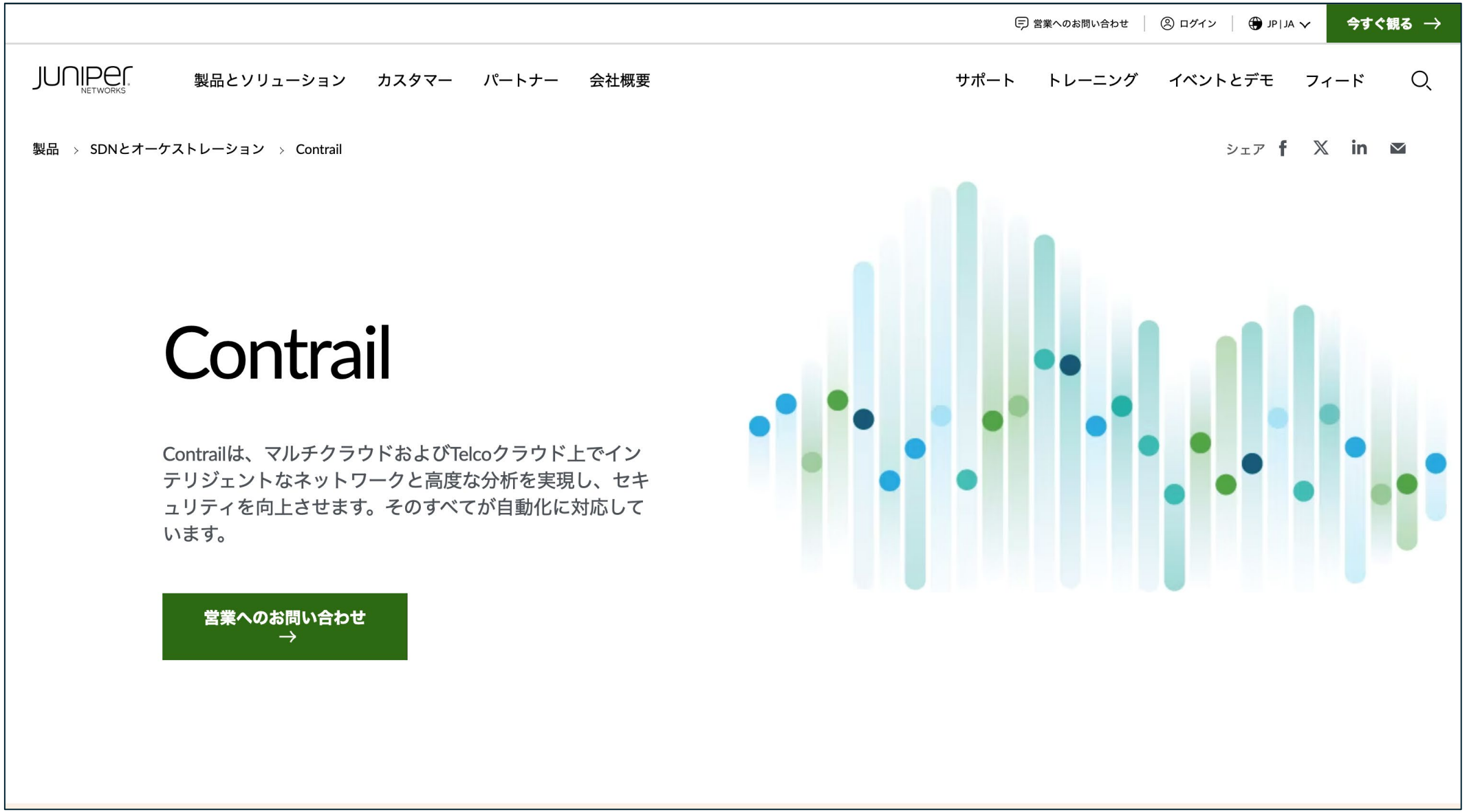
MPLS/VPNからSD-WANへ

Open Networking USER GROUPが掲げる10要件
①WAN回線をActive-Activeに利用できること
②アプリケーション種別、ポリシー、パフォーマンスに応じて、 双方のWAN回線間でダイナミックにトラフィックエンジニアリングが可能であること
③セキュリティ/ガバナンス/コンプライアンスに基づいて、 重要かつ即時性の高いアプリケーションに対する可視化や優先度に応じた トラフィックエンジニアリングが可能であること
④可用性と耐障害性に優れたハイブリッドWANであること
⑤CPEをハード、ソフトの両方で提供可能であること
⑥Layer 2 及び 3で(既存)物理スイッチ、ルーターとの相互接続性があること
⑦拠点/アプリケーション/VPNごとのパフォーマンスを管理する ダッシュボードがあること
⑧特定のログを監視装置やSIEM(セキュリティインシデントマネージャ)へ転送するためのオープンなAPIがあること
⑨ゼロタッチもしくは最小限の設定で迅速に拠点側の構築が可能であること
⑩FIPS140-2(暗号モジュールに関するセキュリティ要件の規格)に準拠していること

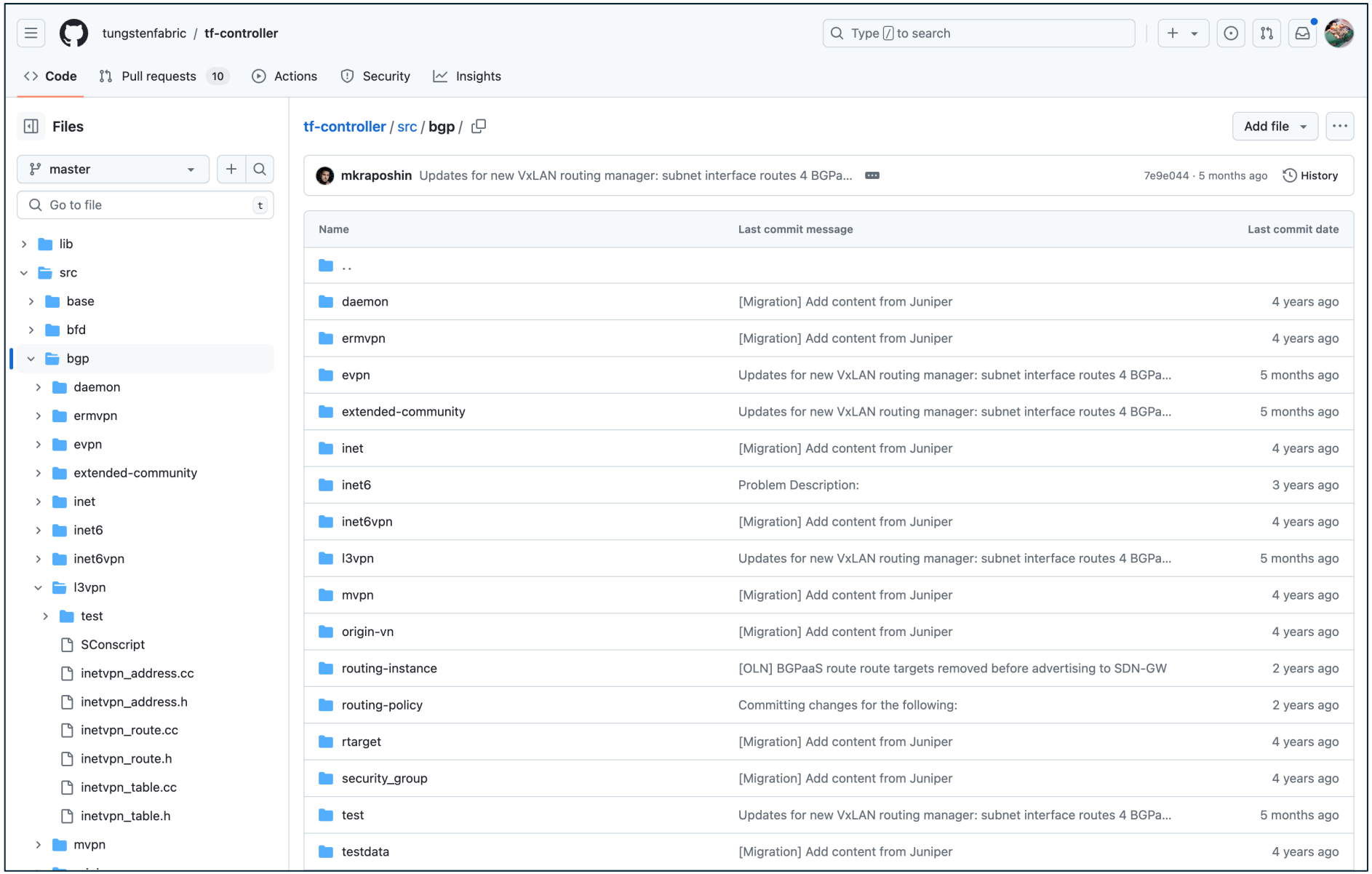
SD-WAN

- Hybrid WAN
  - 複数WANをActive/Activeで同時に使用
  - Jitter/Packet Loss/Delayによって使い分け
- Local Breakout & Remote Breakout
  - 直接ローカルの接続からInternetへNAT接続
  - あるいはリモートのノードからInternetへ抜ける
- サービスベースのルーティング
  - O365やVoIPを特定のゲートウエイ経由でルーティング
- こうしたサービスをBGPで実現
  - Juniper Contrail
  - Cisco Viptela
  - VMWare VeloCloud
  - NTT Cloudwan
  - Huawei SD-WAN

ApplicationとしてのBGP - Contrail

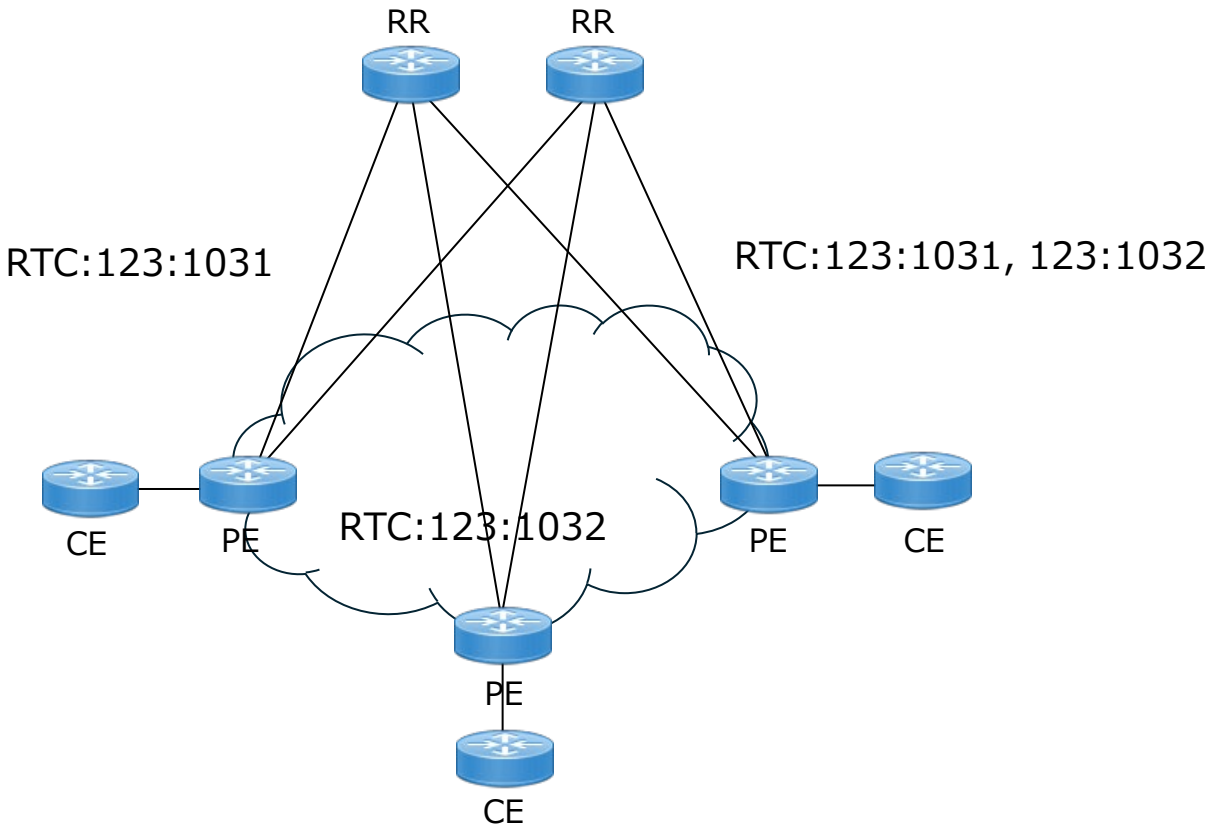


Contrail – C++によるBGP ControllerでSD-WANを実現



RTC – Route Target Constraint

- MPLS/VPNおよびSD-WANで顧客の数が多くなった時にすべてのPEへ全Routing Distinguisherの経路を渡すと負荷が高くなる
- 自分が接続しているCEのRoute Target以外は送らないでくれとメッセージをRRへ送付
- これによりPEの負荷を軽減



LLGR – Long Lived Graceful Restart

- 顧客ネットワークはほぼStatic Routeと同等
- RR(Controller)との接続が切れてもCPE間の経路は保持したままでData Trafficは維持したい
- Graceful Restart
  - 最大Timeoutが12bit=68分16秒
- LLGR – Long Lived Graceful Restart
  - 最大Timeoutが24bit=194日
  - これによりコントローラのソフトウエアアップデートや工事の際でもData Trafficは影響をうけにくく
  - LLGR-Stale Communityが付与される

```
S*>1    45589:6857:192.168.3.0/24       [0]         198.19.30.238
        1d 02:58:37 [{Origin: ?} {LocalPref: 100} {Communities: llgr-stale}
        {Extcomms: [45589:6857]}]
```

AddPath – Reachability InformationとForwarding Decisionの分離

- BGPではPath Selectionを行わずに、すべてのNexthopをFIBにインストール
- ForwarderでPolicyによりどのNexthopを使うか決定する
- 下の実装ではFIBモジュールへBGP Attributeの一部を渡している(AddPathのpathID)

```
6bd6bc4a-601f-4c09-8110-086bf2546b9b>sh ip route vrf vrf1
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

                                    I
B        180.62.216.224/30 [200/0] via 198.19.0.14 pathid 4
B        180.62.216.224/30 [200/0] via 198.19.0.14 hoplimit 255 pathid 1
B        180.62.216.224/30 [200/0] via 198.19.0.14 hoplimit 255 pathid 3
B        180.62.216.224/30 [200/0] via 198.19.0.14 hoplimit 255 pathid 2
C        180.62.220.0/30 is directly connected lan-1
C        198.18.0.0/15 is directly connected sproute3
```

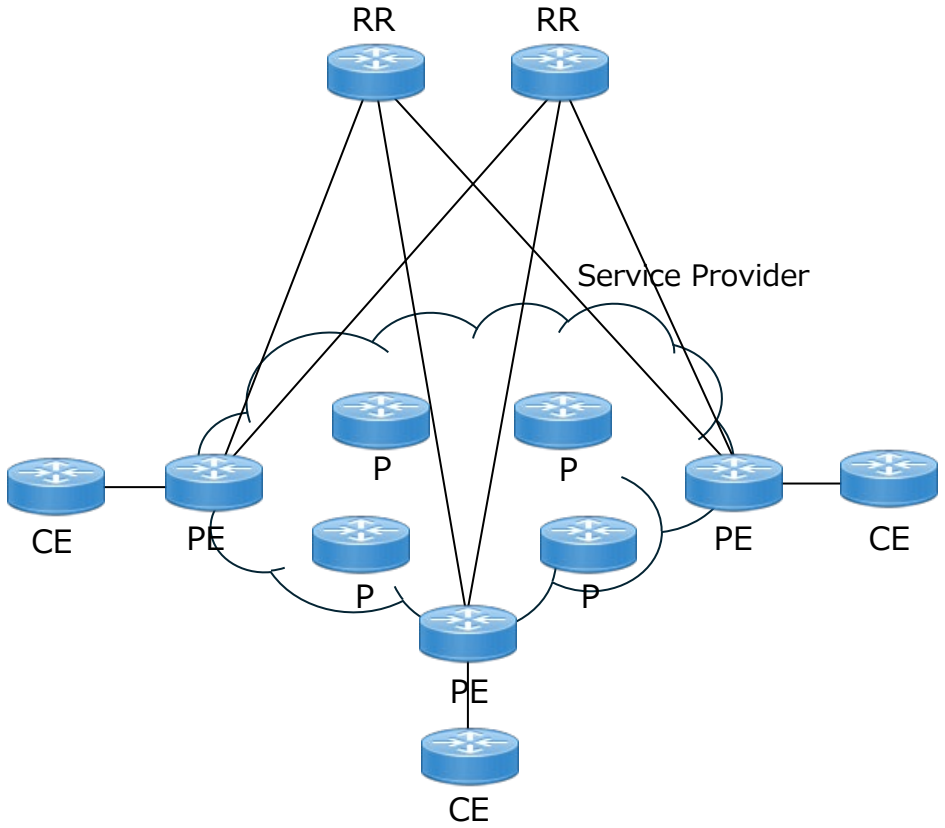
Local Breakout & Remote Breakout

Default VRFからDefault Routeを特定VRFへImport

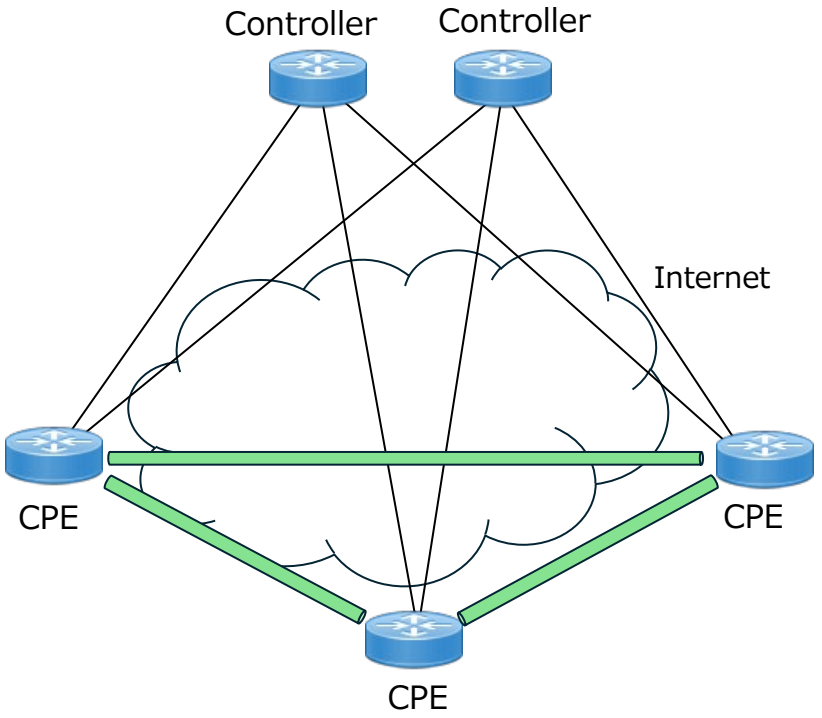
```
6bd6bc4a-601f-4c09-8110-086bf2546b9b>sh ip route vrf vrf1
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

                                    I
I         0.0.0.0/0 [20/1000] via 10.10.126.32
B         180.62.216.224/30 [200/0] via 198.19.0.14 pathid 4
B         180.62.216.224/30 [200/0] via 198.19.0.14 hoplimit 255 pathid 1
B         180.62.216.224/30 [200/0] via 198.19.0.14 hoplimit 255 pathid 3
B         180.62.216.224/30 [200/0] via 198.19.0.14 hoplimit 255 pathid 2
B         180.62.216.228/30 [200/0] via 198.19.0.1 pathid 4
B         180.62.216.228/30 [200/0] via 198.19.0.1 hoplimit 255 pathid 1
B         180.62.216.228/30 [200/0] via 198.19.0.1 hoplimit 255 pathid 3
B         180.62.216.228/30 [200/0] via 198.19.0.1 hoplimit 255 pathid 2
C         180.62.220.0/30 is directly connected lan-1
C         198.18.0.0/15 is directly connected sproute3
```

SD-WAN Data PlaneのFull Mesh課題

Hub-SpokeやP2Pトンネル



PEおよびPはService Providerの内部ネットワークに属しているのでMPLSやSegment Routingなどを適用しやすい



SD-WANではCEとPEの機能が同居したCPEがRRの機能を拡張したControllerと接続する。
CPE間はInternet上でTunnelを張る形が一般的である。
CPE間はFull Meshになるためscale問題が起きやすい