入門書には載っていない
ルーティング Tips

ルーティング チュートリアル

小島 慎太郎 / @codeout

JANOG34, 2014/07/16

小島 慎太郎

codeout

http://about.me/codeout

- ISP: 5年 (ntt.net / AS2914)
- IX: 4年 (JPNAP)

Agenda

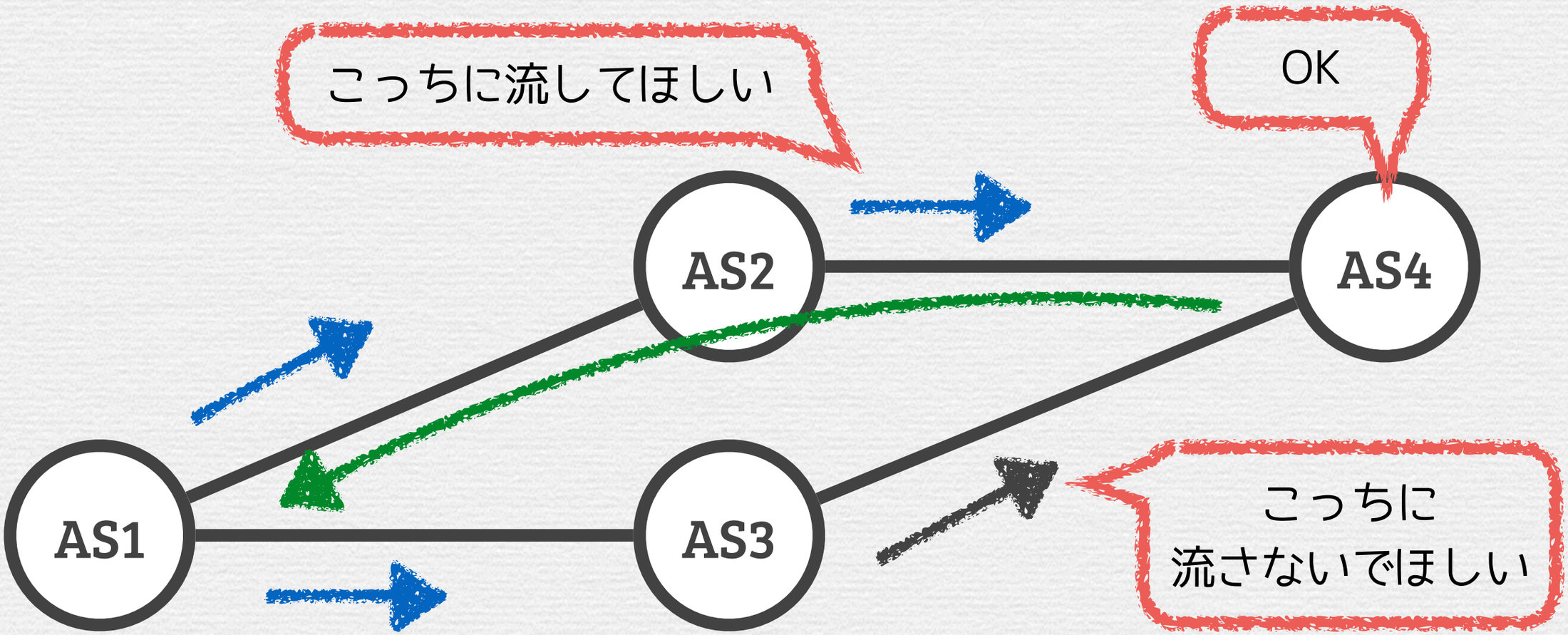
入門書に載ってない，運用経験にもとづくTipsについて話します

- BGPとは?
- BGPの設計を考えよう
- BGPの運用を考えよう
- セキュリティ
- ルーティングエコシステム

BGPとは？

BGP とは?

EGP の一種．異なるAS間でrouting情報を交換する



＝ routing policyを伝える

IGPとの関係

- BGPで解決できるのは，**Internet上のある目的地に達するために，自AS内のどの出口から出ればいいか?** のみ

- **その出口にはどうやったら到達できるか?** はIGP頼み

- IGPの主な用途は，BGPのprotocol nexthopを解決すること
  - むやみにBGP経路をIGPに注入しないほうがいい

A Border Gateway Protocol 4 (BGP-4)

- ***RFC1654*** (July 1994)
- ***RFC1771*** (March 1995)
- ***RFC4271*** **(January 2006)**
- もちろん たくさん拡張されている
  - ***RFC1997*** BGP Communities Attribute
  - ***RFC2385*** Protection of BGP Sessions via the TCP MD5 Signature
  - ***RFC3065*** AS Confederations for BGP
  - ***RFC4451*** BGP MED Considerations
  - ***RFC4456*** BGP Route Reflection
  - ***RFC4360*** BGP Extended Communities Attribute
  - ***RFC5004*** Avoid BGP Best Path Transitions from One External to Another
  - ***RFC5668*** 4-Octet AS Specific BGP Extended Community
  - ...

BGP の経路選択

1. (最も高い **WEIGHT** を持つパスが優先されます．一部メーカーのみ)
2. **最も高い LOCAL_PREF を持つパスが優先されます**
3. network または aggregate BGP サブコマンドによって，あるいは IGP からの再配布を通じて，ローカルで発信されたパスが優先されます
4. **最短の AS_PATH を持つパスが優先されます**
5. 最小のオリジン タイプを持つパスが優先されます
6. **最小の Multi-Exit Discriminator (MED) を持つパスが優先されます**
   - **MED** は remote AS が同じ場合のみ評価される
7. iBGP パスよりも eBGP パスの方が優先されます
8. **BGP ネクストホップへの最小の IGP メトリックを持つパスが優先されます**
9. 両方のパスが外部のときは，先に受信したパス（最も古いパス）が優先されます
10. 最小のルータ ID を持つ BGP ルータから送られたルートが優先されます
11. 発信元 ID またはルータ ID が複数のパスで同じ場合は，最小のクラスタリスト長を持つパスが優先されます
12. 最小の隣接ルータ アドレスから送られたパスが優先されます

http://www.cisco.com/cisco/web/support/JP/100/1008/1008556_25-j.html

今日話すこと

- **BGPの設計時に考えるべきこと**について話します

- “BGP sessionの先にいるAS (顧客 / peering パートナー / transit 提供者) にも個別のrouting policyがある” という環境で, **どうやって自分の思うようにtraffic controlするか?** を理解するために
  - 私が使っている手法の紹介
  - その解説

もします

BGP の設計を考えよう

- eBGP policy / 設計
  - 経路広告
  - 経路受信
- iBGP policy / 設計

eBGP policy (基本)

eBGP Policy を考えるポイント

- **どんな経路** を
  - **どんなeBGP session** からもらうか?
  - **どんなeBGP session** へ広告するか?

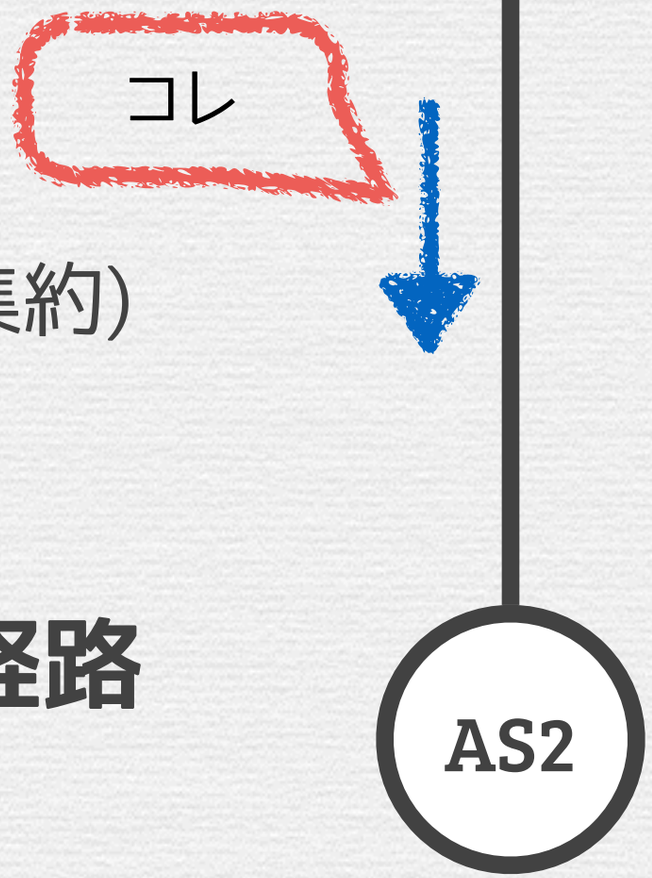
- 経路の各path attributeは誰のものか?
- Full Route 持てないんだけど どうしよう?

どんな経路を

高

低

- **顧客の経路**
  - BGP 顧客
  - static 顧客 (実際はPAに集約)
- **自ASの経路**
  - PA/PI経路
- **peering パートナーの経路**
- **transit 事業者の経路**
- **不要な経路**



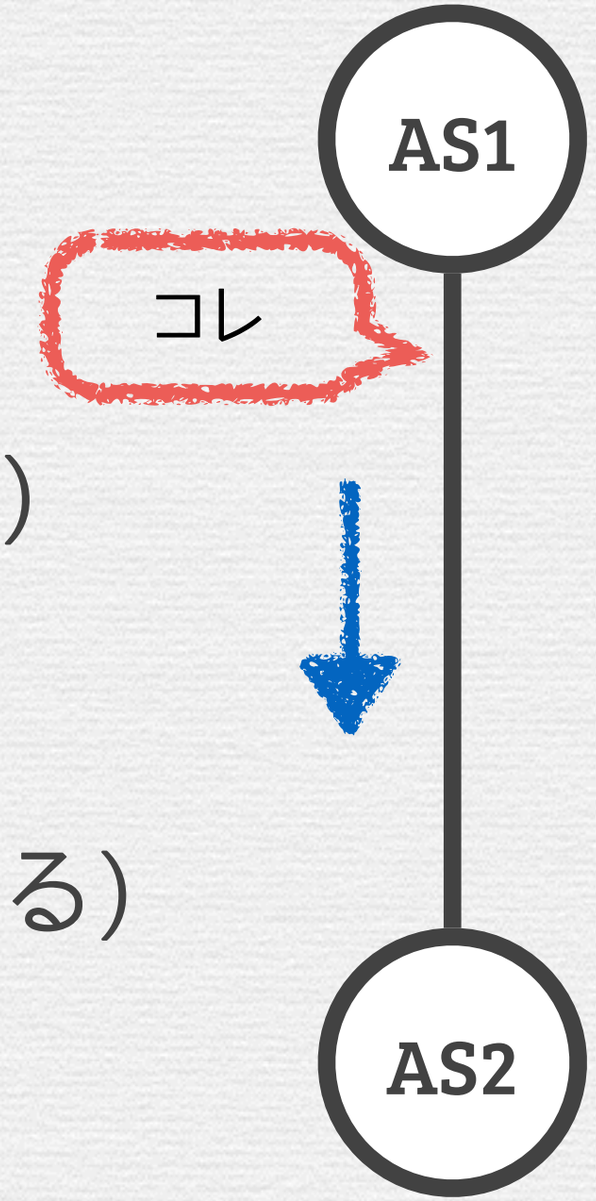
ビジネス上の観点から，基本的には上記の順に優先する
(優先 = なるべくその経路に従ってpacketを流したい / 流してほしい)

どんなeBGP sessionへ

高

低

- **顧客**
- **peer**
  - paid peer（収益を得ている）
  - private peer
  - public peer（IX）
  - paid peer（費用を払っている）
- **transit**



同様に，基本的には上記の順に優先する
(優先 = なるべくその接続/回線にpacketを流したい)

eBGP Policy の基本

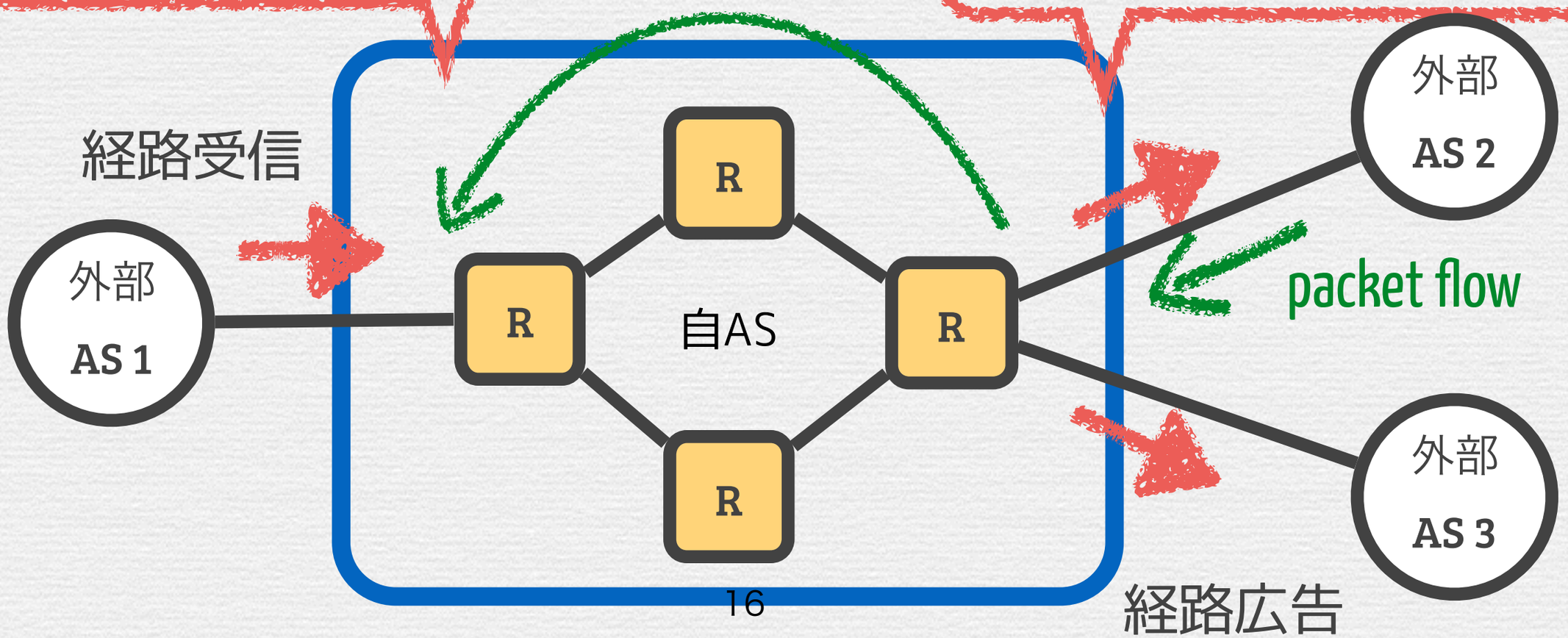
受信 / 広告Policyは何に影響する?

受信Policy

AS内でどのように routing させる?

広告Policy

AS外でどのように routing させる?



eBGP

経路受信

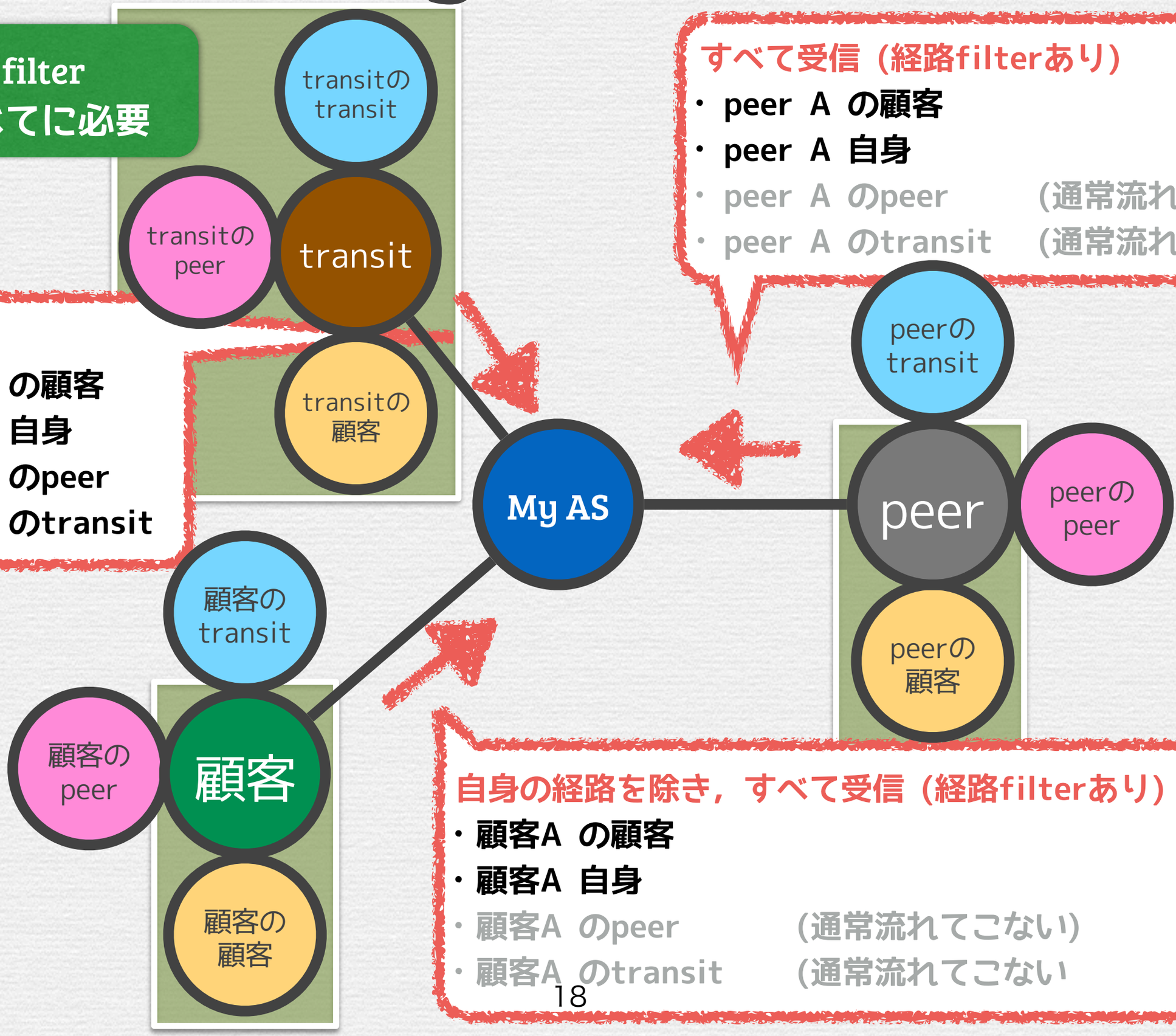
(ちょっと細かく)

eBGP Policy の基本（受信）

bogon filter
はすべてに必要

すべて受信（経路filterあり）
- **peer A の顧客**
- **peer A 自身**
- peer A のpeer　（通常流れてこない）
- peer A のtransit　（通常流れてこない）

すべて受信
- **transit A の顧客**
- **transit A 自身**
- **transit A のpeer**
- **transit A のtransit**



自身の経路を除き，すべて受信（経路filterあり）
- **顧客A の顧客**
- **顧客A 自身**
- 顧客A のpeer　（通常流れてこない）
- 顧客A のtransit　（通常流れてこない

受信経路を扱うときに気にするべきポイント3つ

1. LPとMEDの値

2. 経路Filter

3. Path Attribute は本来の用途に使おう

1. LPとMEDの値

優先度	経路種別	LP	MED
1	顧客	150~300 (*)	上書きしない
2	自AS	200	なし
3	peer	200	上書き (十分大きな値)
4	transit	120	上書き

(*) 200をまたいで数段階 (default: 300)

- 幅に余裕をもって数値を設定
- LPのdefault値が100の実装が多いので,
  安全を期すならLPの値は>100

－ 解説 －

顧客の明確な意図がない限り最優先
常にRIBに保持

経路を指定して他の顧客やpeerに迂回させることができる

peerの200と比較して，必ず次のAS_PATHで優先度が決定

顧客のTE選択肢を増やす

優先度	経路種別	LP	MED
1	顧客	150~300 (*)	上書きしない
2	自AS	200	なし
3	peer	200	上書き (十分大きな値)
4	transit	120	上書き

- AS_PATHが同じならIGPベースのclosest exitを強制
- always-compare-med や peer&顧客AS対策でMEDを高めに
- 2^32 – 1 でもいいかも

private / public peer で2段階あるISPもある

一応>100

AS_PATHが同じならIGPベースのclosest exitを強制

(*) 200をまたいで数段階（default:300）

1. LPとMEDの値

2. 経路Filter

3. Path Attribute は本来の用途に使おう

2. 経路Filter

顧客経路

優先度が非常に高いため，細かくチェックする必要がある

- **IRRベース が理想的**
  - 頻繁にメンテナンスできるのであれば，手動でも問題ないかもしれない
- filter方法の候補
  - exact match な prefix filter
  - exact match な prefix filter & origin AS filter

- **bogon prefix，自ASのprefixは経路filterで除外**
- BGP communityは透過的に扱う

IRRベースって言われても…

- Rubyから使えるやつ
- PerlでいうNet::IRR

https://github.com/codeout/irrc

peer経路

peering: transitより責任範囲が小さい
➔ 少ない情報に基づき，ゆるくfilterする

- **Mis-Origin（経路ハイジャック）の可能性**
- 細かいfilterを設定してしまうと**メンテナンスされなくなるかもしれない**
  - “AS_PATHをメールで伝え合う”しくみが動いていた
  - やはり限界があるので，自動で動くことが望ましい

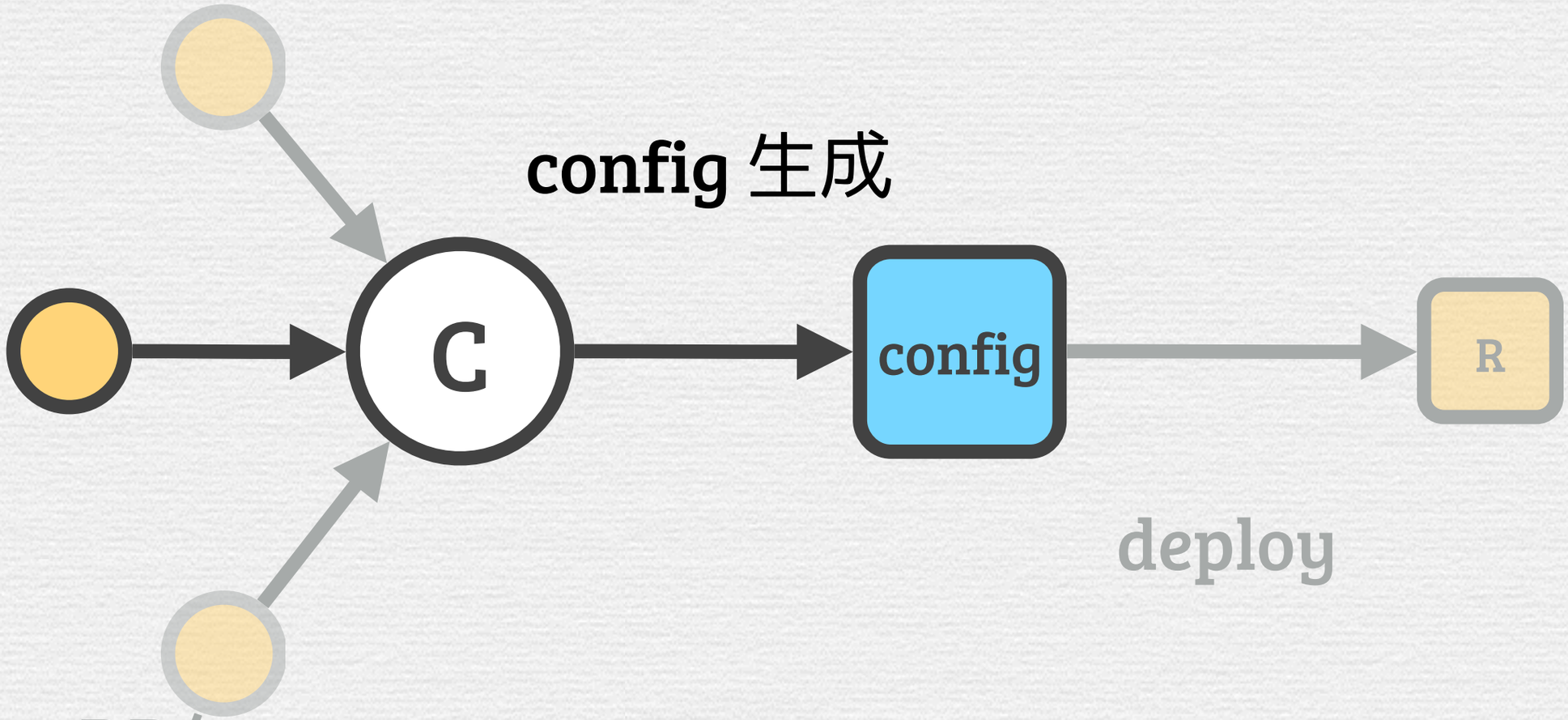
peer経路

他の国では以下の組み合わせが多い

- **maximum-prefix (prefix-limit) filter**
  - 実績ベース（昨日から20%増えたらアウト，など)
  - peering dbベース
- **prefix lengthによるfilter**
  - ipv4: le /24
  - ipv6: le /48 など

bgpq3 / IRR Power Tools

- IRRの展開 ➔ config生成までざっくりやってくれる
- かゆいところに手は届かないが，シンプル



input:

IRR / peering DB/
実経路数 / 顧客連絡

Maximum-Prefix (Prefix-Limit) Filter

⚠ transitに設定すると危険な場合がある

- network上のeventにより急にprefix数が増えると，transitが全断するリスクがある
  - internetから遮断されることになる

経路Filterに関する参考情報

JANOG Comment JC1000~1003 に
大変丁寧にまとめられている

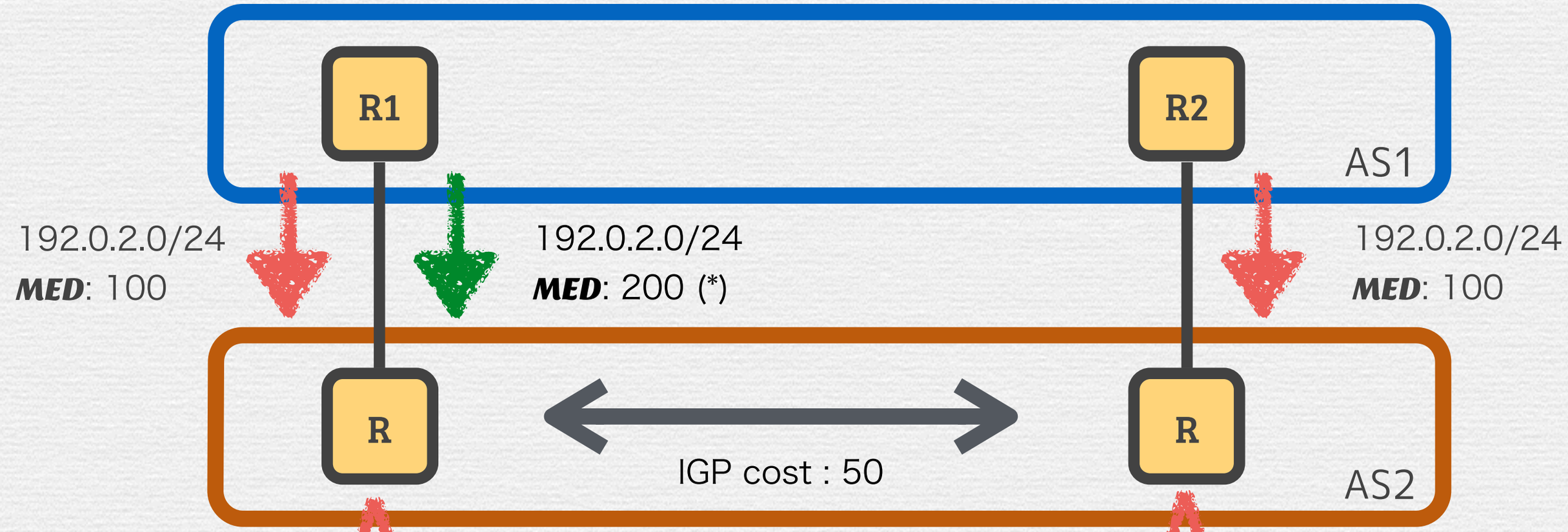
http://www.janog.gr.jp/doc/janog-comment/

1. LPとMEDの値

2. 経路Filter

3. Path Attribute は本来の用途に使おう

3. Path Attribute は本来の用途に使おう
– 例えば closest exit –




```
  prefix          Next Hop   MED     IGP
> 192.0.2.0/24    R1         100     0
  192.0.2.0/24    R2         100     50
```

```
  prefix          Next Hop   MED     IGP
  192.0.2.0/24    R1         100     50
> 192.0.2.0/24    R2         100     0
```

IGPによる制御

(*) MEDが異なる経路でもclosest exitしたい場合はMEDを
上書きする必要がある

IGPを用いる代わりに，MEDを加算することでも一応実現できる

Juniper:	metric add 500
Cisco:	set metric +500

- ✕ router間でMED加算して回る必要がある
  (“壁” になっている 500 という数値が十分かどうかはpeer partnerの広告policy次第)

- 本来のMEDの用途とは異なる

3. Path Attributeは本来の用途に使おう

よくあるパターン

1. 実装できるからといって，本来の用途からはみ出る
   やりたいことに対して大掛かりすぎ
2. おまじない的な設定や(消せそうだが消してはならない)
   運用対処が増える
3. “ **なんかめんどくさくない？** ”
   “ **うわ，壊れた！** ”

http://www.flickr.com/photos/techsavvyed/5926978939/

受信経路に手を入れる
＝ 経路制御する
方法を決めておく
(運用設計)

経路制御の選択肢

- **transit / peer 経路**
  - LPの微調整
  - AS_PATH prepend
  - MED微調整
  - passive IGP
  - 経路を止める

- **顧客経路**
  - 顧客からの依頼に基づく場合を除き，操作しない

- **LP / MED などの数値はなるべく弱くなる方に制御する**
  - ヘンにtrafficを吸い込むのを防ぐ
  - MED:　増やす
    LP:　減らす(多くの実装でdefault：100)
- **AS_PATH prepend**

  prepend された経路がRIBに残ってしまう可能性があるので，なるべく避ける
  - transitを売っている場合など，全体として経路が遠く見えてしまうのは良くない
- **passive IGPは経路ごとの制御ができない**

- **LP / MED / AS_PATH 操作**
  - **なるべくメンテナンス頻度を下げる**

- peer AS
- nexthop
- origin AS
- AS_PATH
- prefix

の順で操作

Juniper の例:

```
protocols {
    bgp {
        neighbor x.x.x.x {
            import [ regular irregular finalize ];
        }
    }
}

policy-options {
    policy-statement regular {
        from { ... }
        then {
            # 通常のpolicy
            next policy;
        }
    }
```

template

```
    #
    # remove me soon !!
    #
    policy-statement irregular {
        from { ... }
        then {
            # irregular なpolicy
            next policy;
        }
    }
    policy-statement finalize {
        then accept;
    }
}
```

hack

Cisco の例:

```
no route-map route-filter
route-map route-filter permit 10
  ! 通常のpolicy
route-map route-filter permit 20
  ! 通常のpolicy
route-map route-filter permit 30
  ! 通常のpolicy
```

template

```
!
! remove me soon !!
!
route-map route-filter permit 25
  ! irregular なpolicy
```

hack

irregularなことは目立つように / 消しやすく

よく使うのは次の3種類くらい

経路制御の選択肢（受信）

– LP 制御 –

一部経路はpeerより
優先させたい

transit

AS1

```
  prefix         LP  MED  AS_PATH
> 192.0.2.1/32  120 1000 1
  192.0.2.1/32  110 1000 2
```

```
prefix         MED AS_PATH
192.0.2.1/32 100 1
```

AS2

My AS

peer

```
prefix         MED AS_PATH
192.0.2.1/32 50  2
```

```
prefix         MED AS_PATH
192.0.2.1/32 100 1
```

顧客

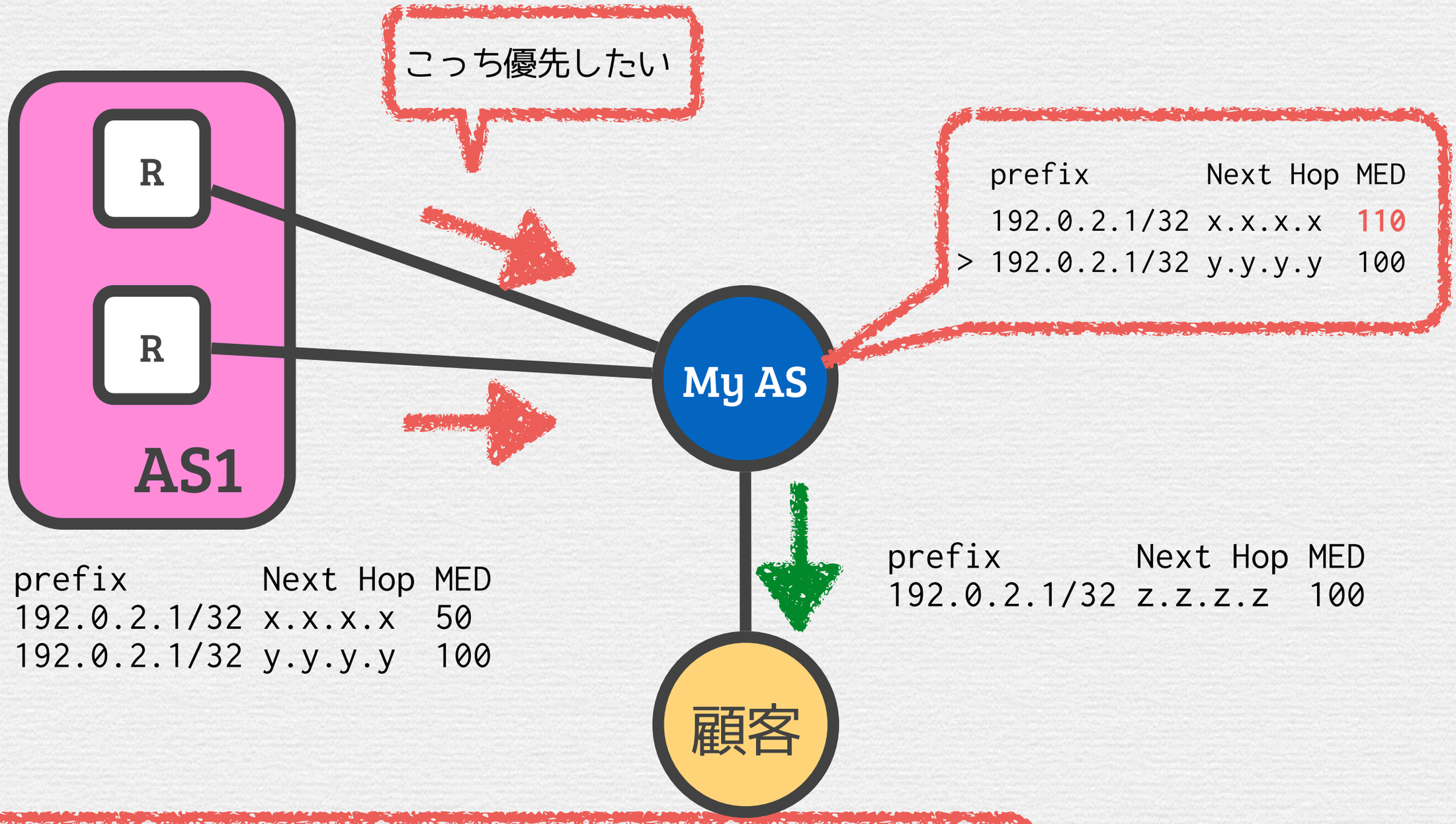
peer からの経路について,
LP を下げる

LP policy

- transit 120
- peer 200

経路制御の選択肢（受信）

– MED 制御 –



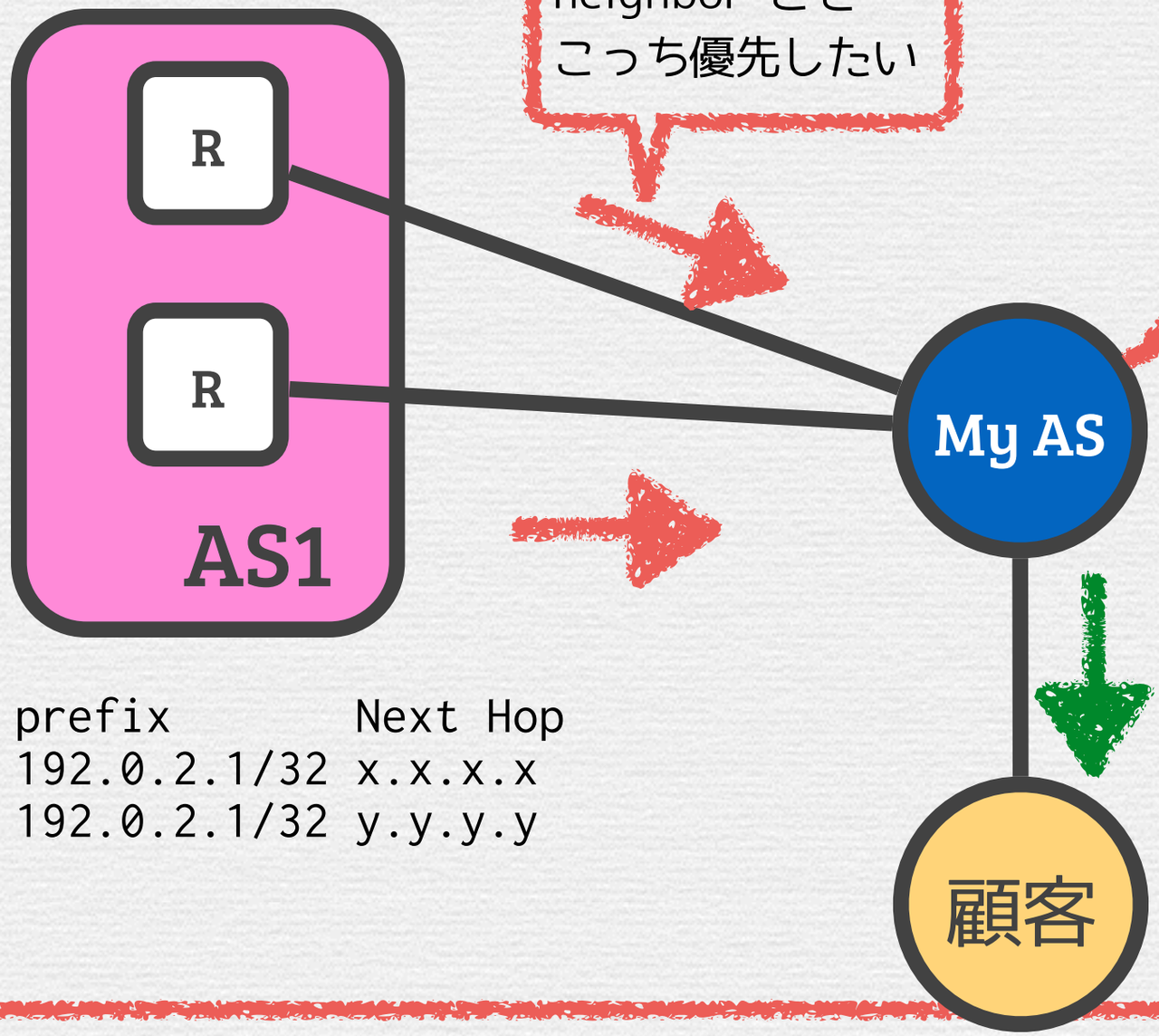
一方のMEDを大きくする
(大きな値をセットする or **加算して大きくする**)

経路制御の選択肢（受信）

－ IGP 制御 －

neighbor ごと
こっち優先したい

```
  prefix          Next Hop  IGP
> 192.0.2.1/32    x.x.x.x   10
  192.0.2.1/32    y.y.y.y   40
```




```
prefix          Next Hop
192.0.2.1/32    x.x.x.x
192.0.2.1/32    y.y.y.y
```

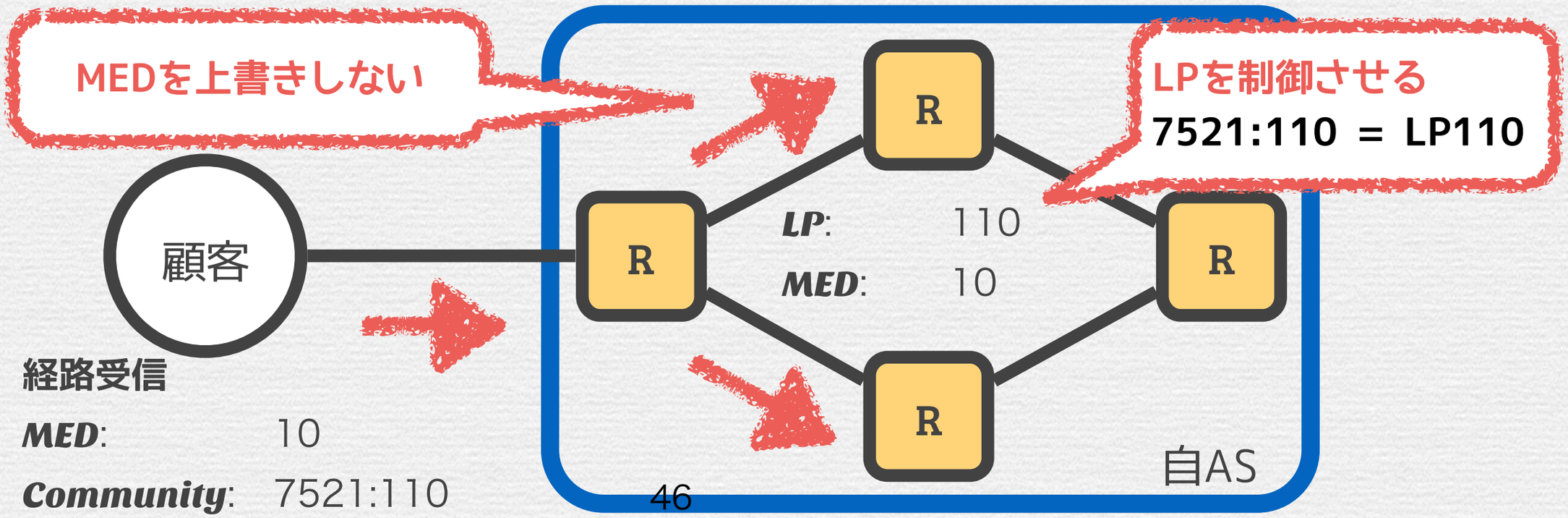
```
prefix          Next Hop
192.0.2.1/32    z.z.z.z
```

通常だと，MY ASから見て x.x.x.x，y.y.y.y へのIGP costは両方 10 だが，一方だけ 40 にする

ところで，
考えてみて
ください

経路の各path attributeは誰のもの?

- 以下のような機能を提供しているISPも
  http://onesc.net/communities/
  - 顧客経路の**MEDを上書きしない**
  - 自AS内での**local preference(LP)を制御するBGP community**を顧客向けに提供する



ここまで経路受信の話

eBGP

経路広告

eBGP Policy の基本（広告）

bogon filter はすべてに必要

顧客経路 ＋ 自分の経路のみ広告

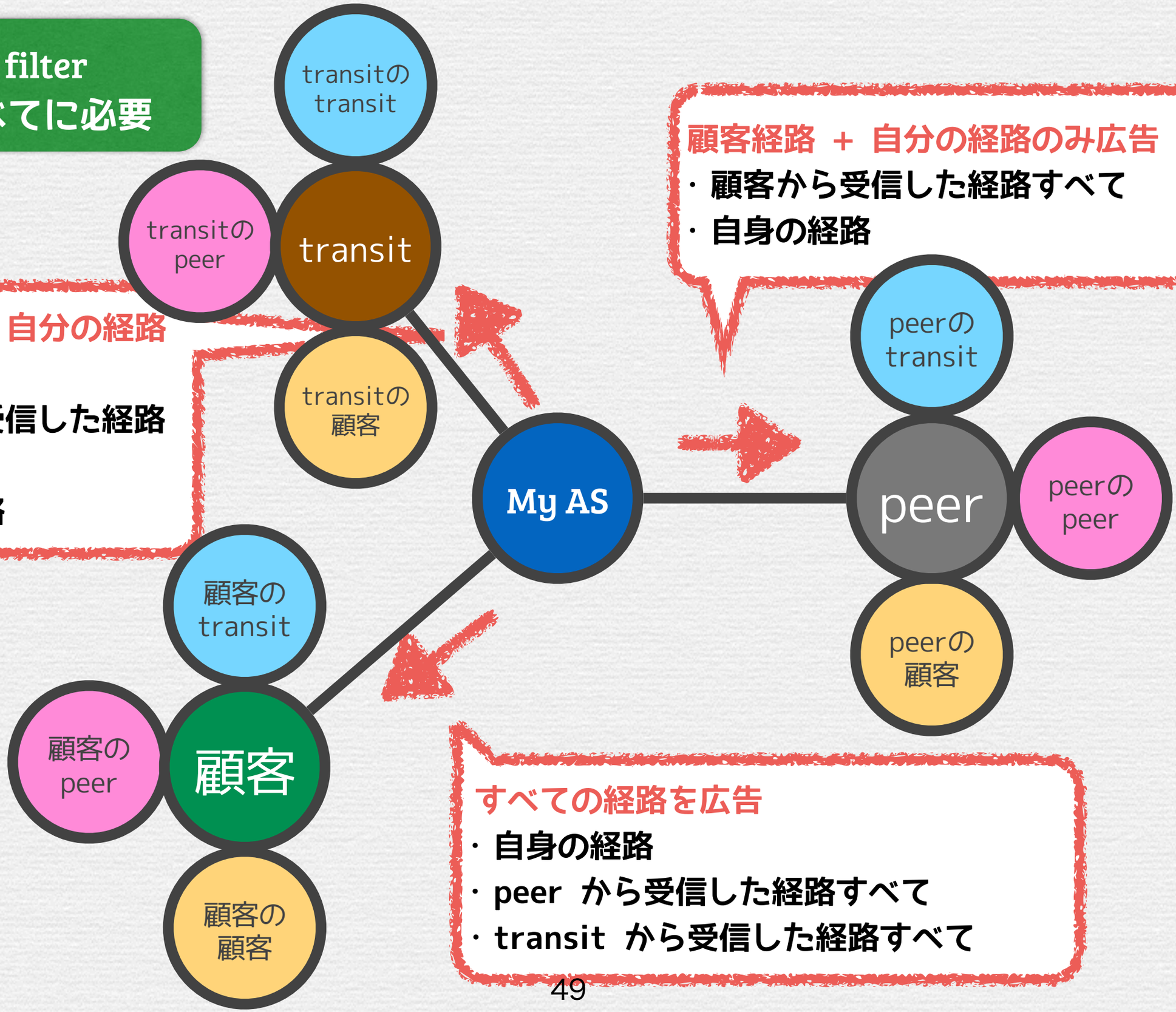
- **顧客から受信した経路すべて**
- **自身の経路**

顧客経路 ＋ 自分の経路のみ広告

- **顧客から受信した経路すべて**
- **自身の経路**

すべての経路を広告

- **自身の経路**
- **peer から受信した経路すべて**
- **transit から受信した経路すべて**



広告経路を扱うときに気にするべきポイント3つ

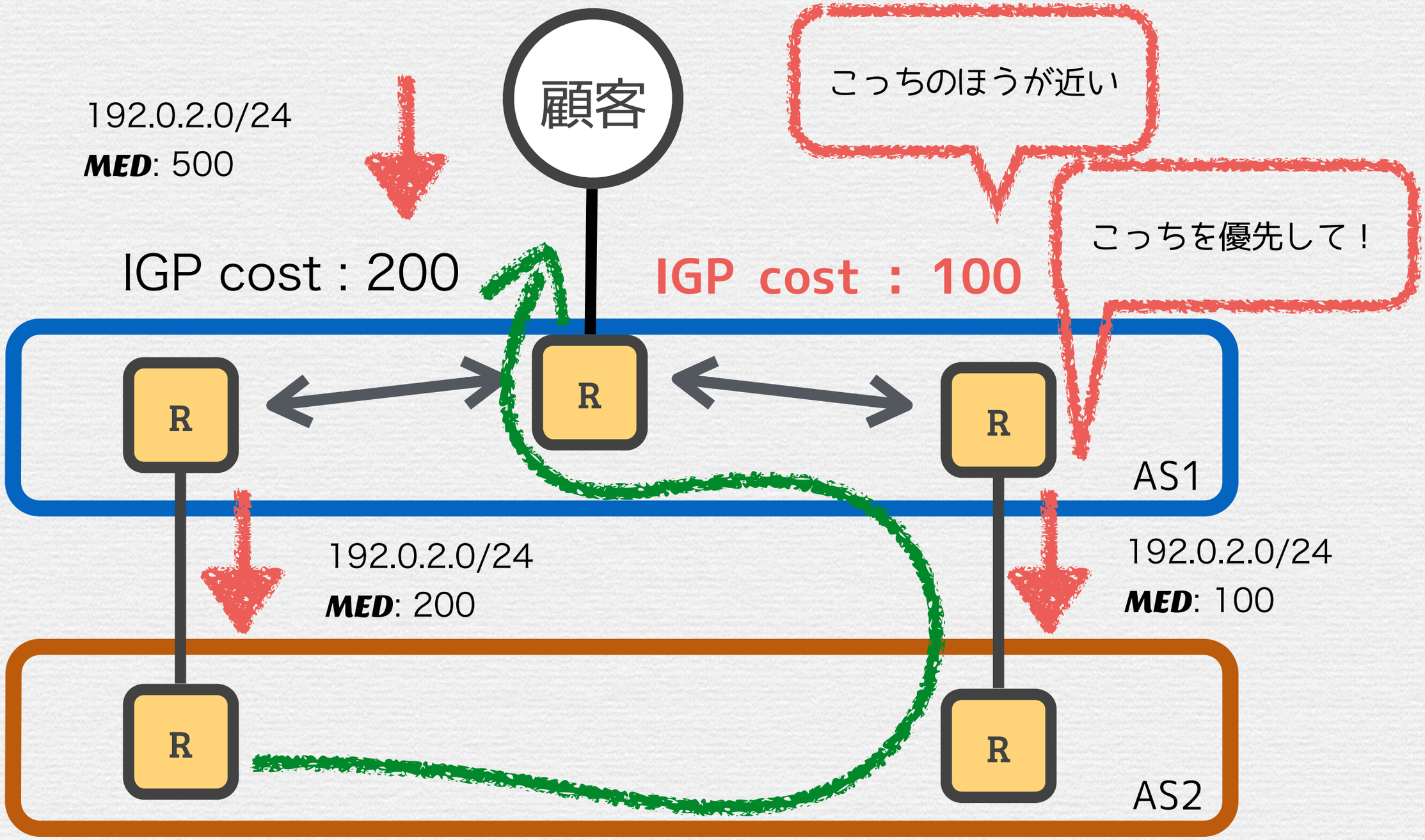
1. MEDをちゃんと付ける

2. 経路Filter

3. ASPATH prepend

ってちょっと強い

1. MEDをちゃんと付ける



AS2 が AS1 のMEDを
評価した場合の
packet flow

- metric-out igp が便利
  - IGP costを広告時のMEDとして利用

Juniper の例:

```
protocols {
    bgp {
        group ebgp {
            metric-out igp;
            neighbor x.x.x.x { ... }
        }
    }
}
```

Cisco の例:

```
router bgp xxxx
  ...
  neighbor y.y.y.y route-map ebgp
route-map ebgp
  ...
  set metric-type internal
```

⚠ 外部ASが常にMEDを評価してくれるとは限らない．でも
- ダメもとでも広告しておく価値あり
- 評価してもらいたいなら，交渉

1. MEDをちゃんと付ける

2. 経路Filter

3. ASPATH prepend

ってちょっと強い

2. 経路Filter

- **内部の細かい経路は広告しない**
  - connected(direct)経路はBGPにredistributeしないなど
    - するときにはno-export communityを付けておく
- **bogonその他，internetに流すべきでない経路が含まれないかを再確認**
- **remove-privateしておく（private ASは削って広告）**

1. MEDをちゃんと付ける

2. 経路Filter

3. ASPATH prepend

ってちょっと強い

3. AS_PATH prepend

- 制御できるが，比較的制御が強い
  - “設計”に含めるのは，やりすぎなイメージ
  - どちらかというとtrouble shootなどの暫定対処用

広告経路に手を入れる
＝ 経路制御する
方法を決めておく
(運用設計)

経路制御の選択肢（広告）

- **transit / peer**
  - AS_PATH prepend
  - MED 微調整
  - BGP community (一部transitのみ)
    - transit AS 内のLP を制御
    - transit AS ➔ 他AS 経路広告を制御 (広告しない / prepend)
  - 経路広告を止める

- **顧客**
  - 顧客からの依頼に基づく場合を除き，操作しない

あまり手がない

経路制御の選択肢（広告）

広告経路の制御が効かない場合も多々ある

- 顧客 / peering パートナー / transit提供者にも彼らなりのpolicyがあるので，やむを得ない
- **接続しているtransit / peer のrouting設計を理解することがすごく重要**
  - MEDは効くか?
  - AS_PATH prepend可能か?
  - best pathはどのpath attributeで決まっているか?
  - 制御系BGP communityはあるか?
  - **そのような設計になっているのはなぜか?**

以下の選択肢は
高い確率で効果が期待できる

- BGP Community
- **経路広告を止める**
  - 多くの場合，冗長性を損なう
  - more specificな経路にする(/20 = 2x /21)
    - 管理が煩雑になる

奥の手！

ところで，
こういうのは
どう思いますか?

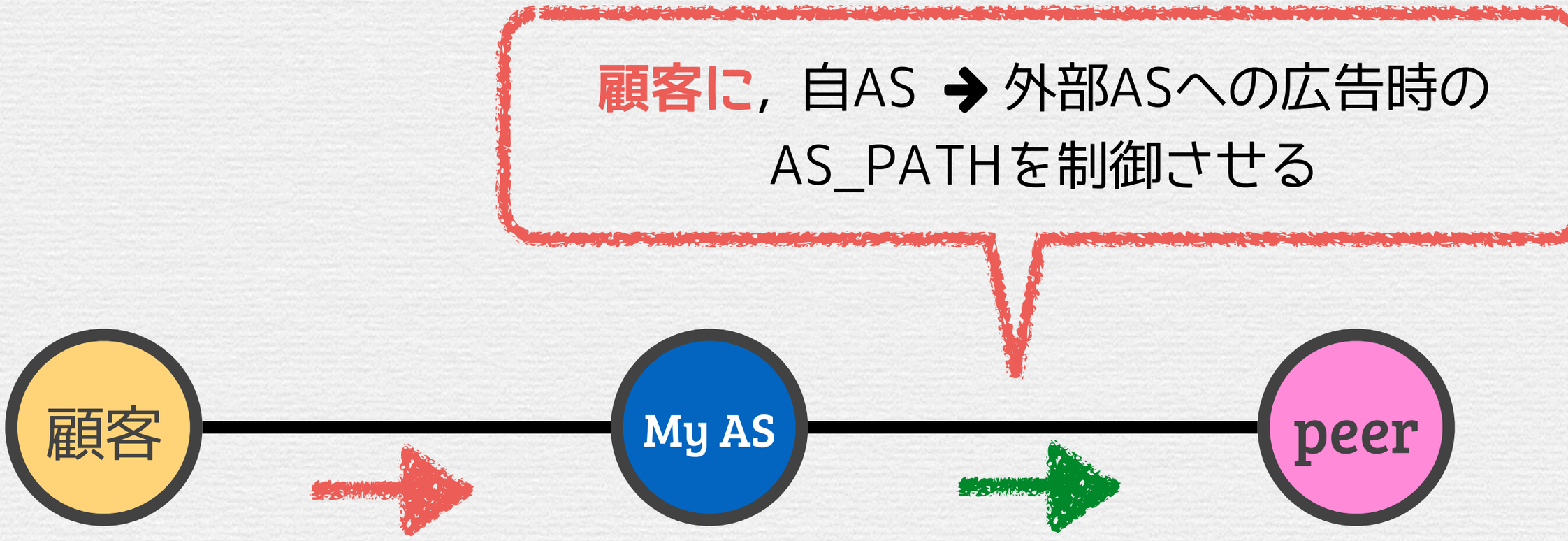
BGP Community 便利

- 経路の種別による ”マーク” をBGP communityとして付与すると顧客は便利に使える

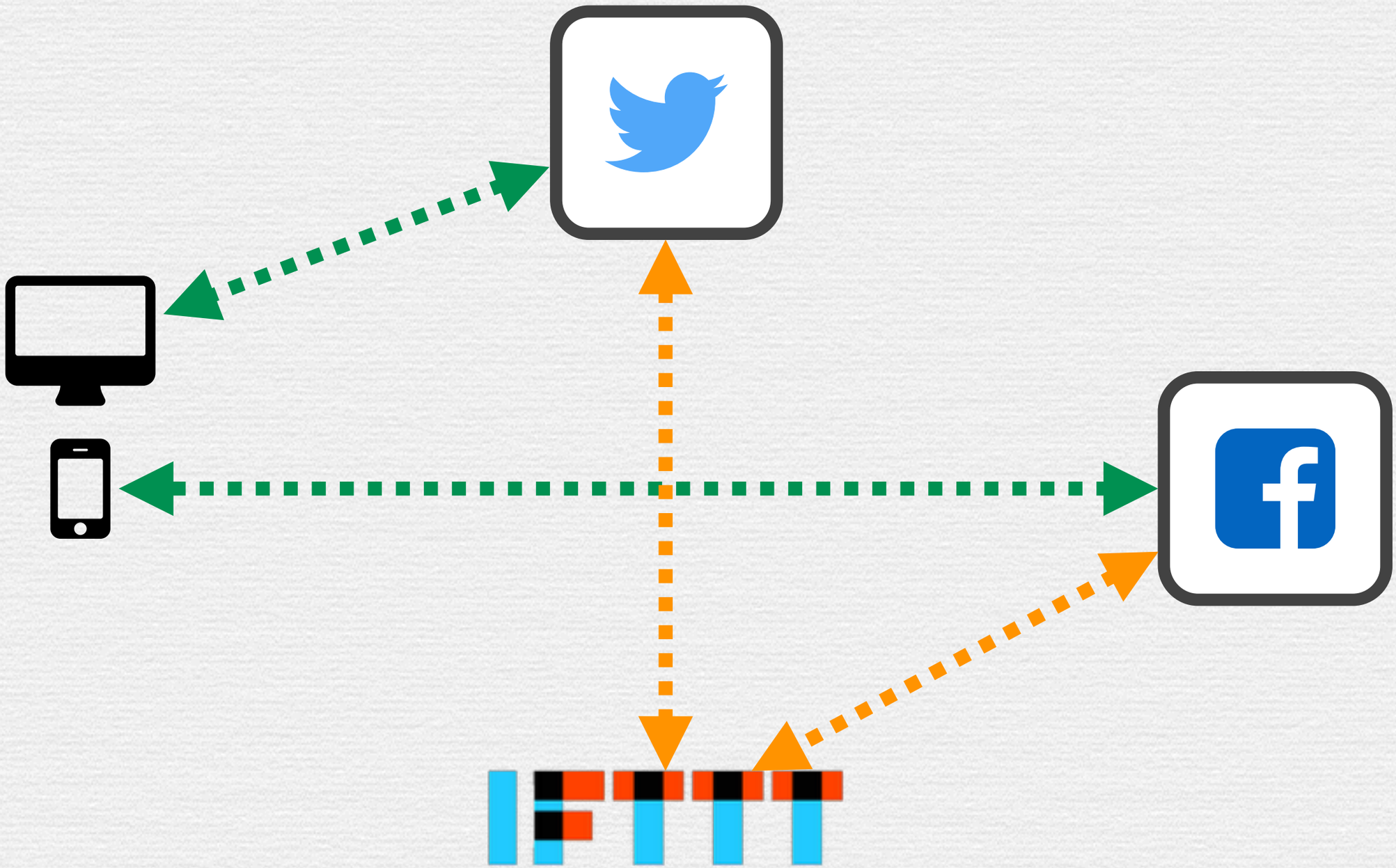
- **どの地域** の経路?
- どんな経路か? **transit**? **peer**? **顧客**?

prepend community とかどうですか?

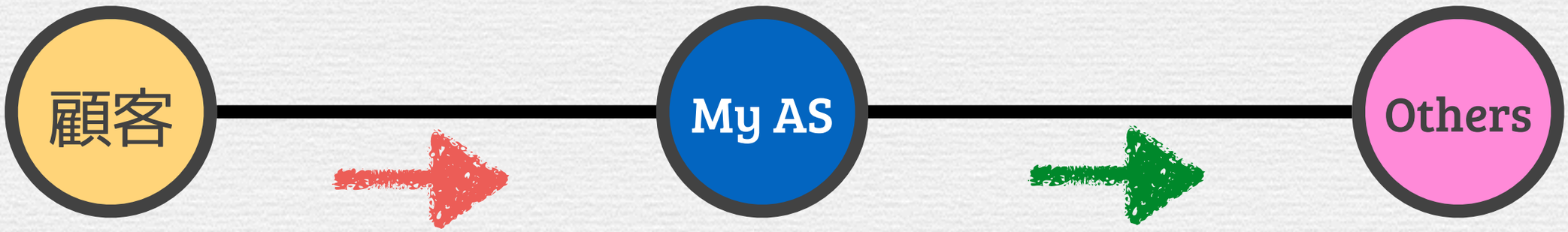
顧客に，自AS ➔ 外部ASへの広告時の
AS_PATHを制御させる



BGP Community = API



BGP Community = API



特別なBGP Community
を付けると

ルーティングを制御できる

ここまで eBGP 経路広告の話

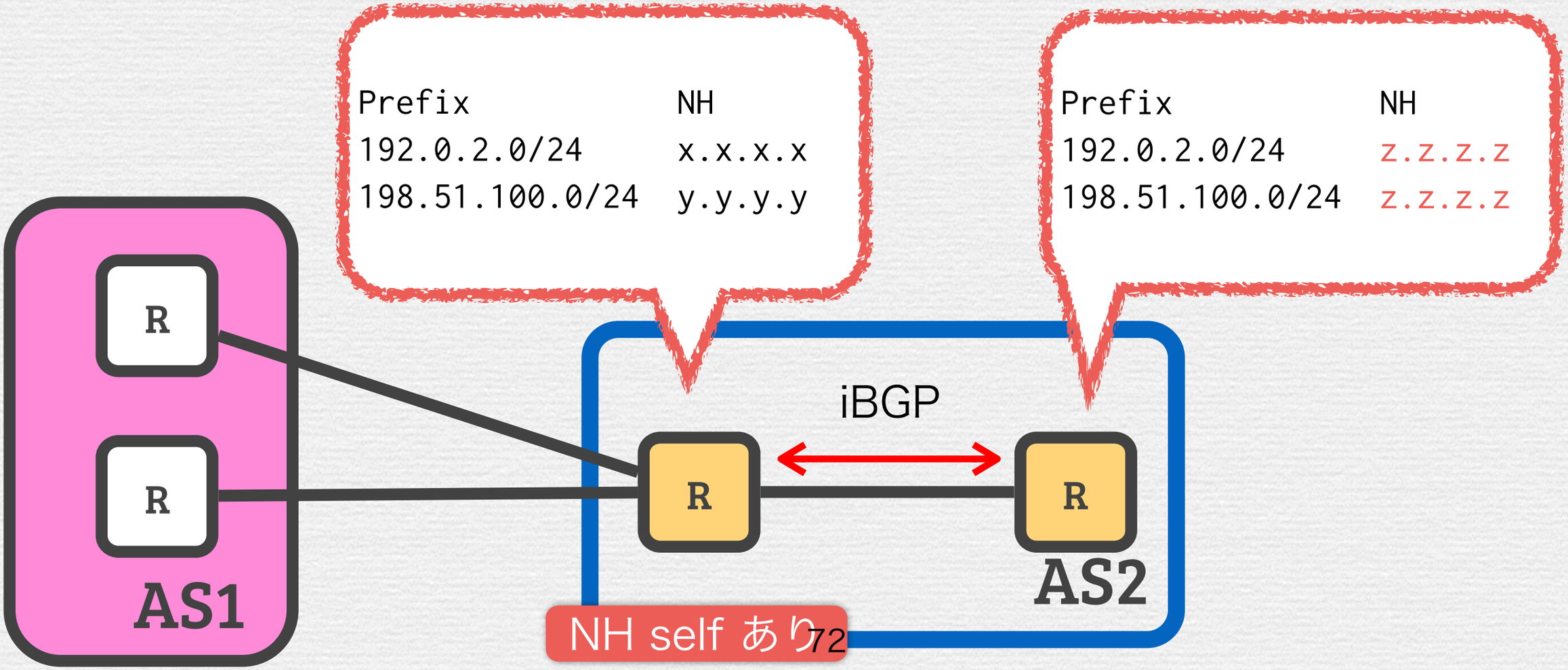
iBGP policy / 設計

あまりたくさん話しませんが，気に留めておくと良さそうなこと

next-hop self

next-hop self

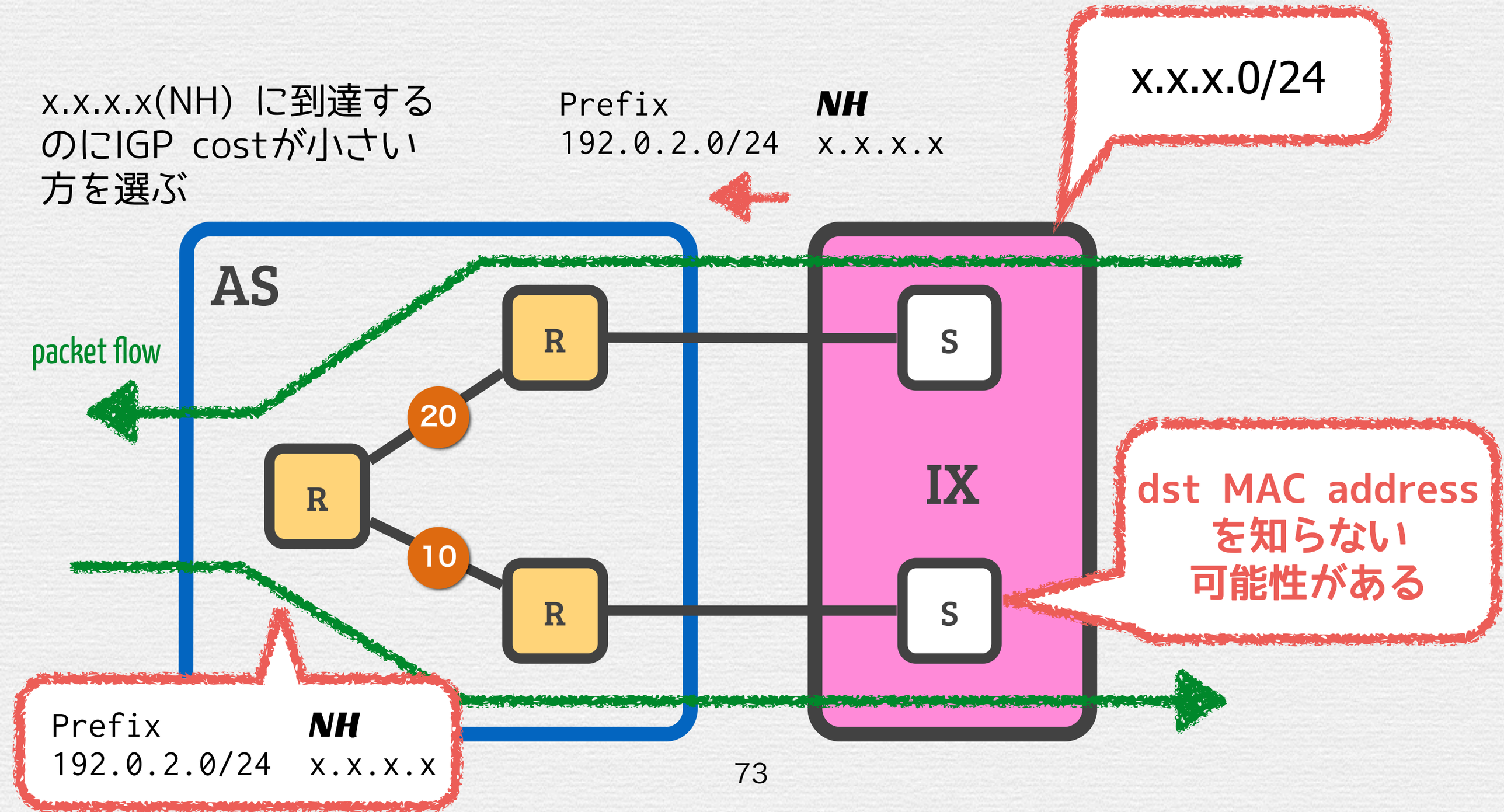
- 自AS内のtrafficを細かく制御したい場合はnext-hop selfしないほうがいい
  - **経路制御の選択肢を1つ失う**
- したほうがいい場合もある（後述）



next-hop self したほうがいい場合

IX経由でもらった経路をiBGPに流すとき

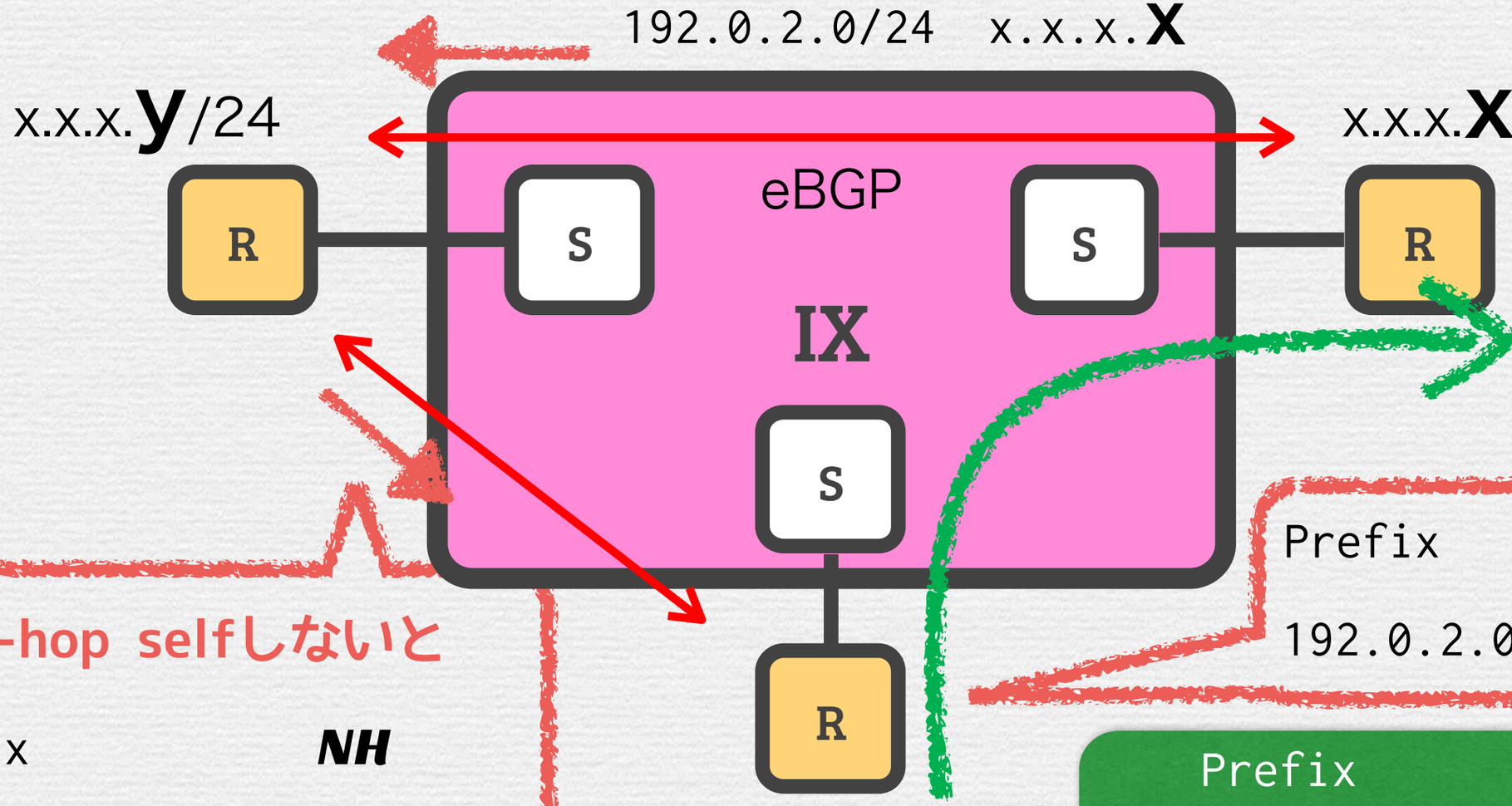
http://www.janog.gr.jp/doc/janog-comment/jc1005.txt



next-hop self したほうがいい場合

IX経由でもらった経路を同じIX上の別のeBGPの流すとき

http://www.janog.gr.jp/doc/janog-comment/jc1005.txt



next-hop selfしないと

Prefix	NH
192.0.2.0/24	x.x.x.X

Prefix	NH
192.0.2.0/24	x.x.x.X

Prefix	NH
192.0.2.0/24	x.x.x.y

であるべき

x.x.x.x(NH) に直接転送してしまう

ここまで，
設計 ＝ 平常運転時のBGP
について話しました

Questions 

BGP の運用を考えよう

Traffic Engineering

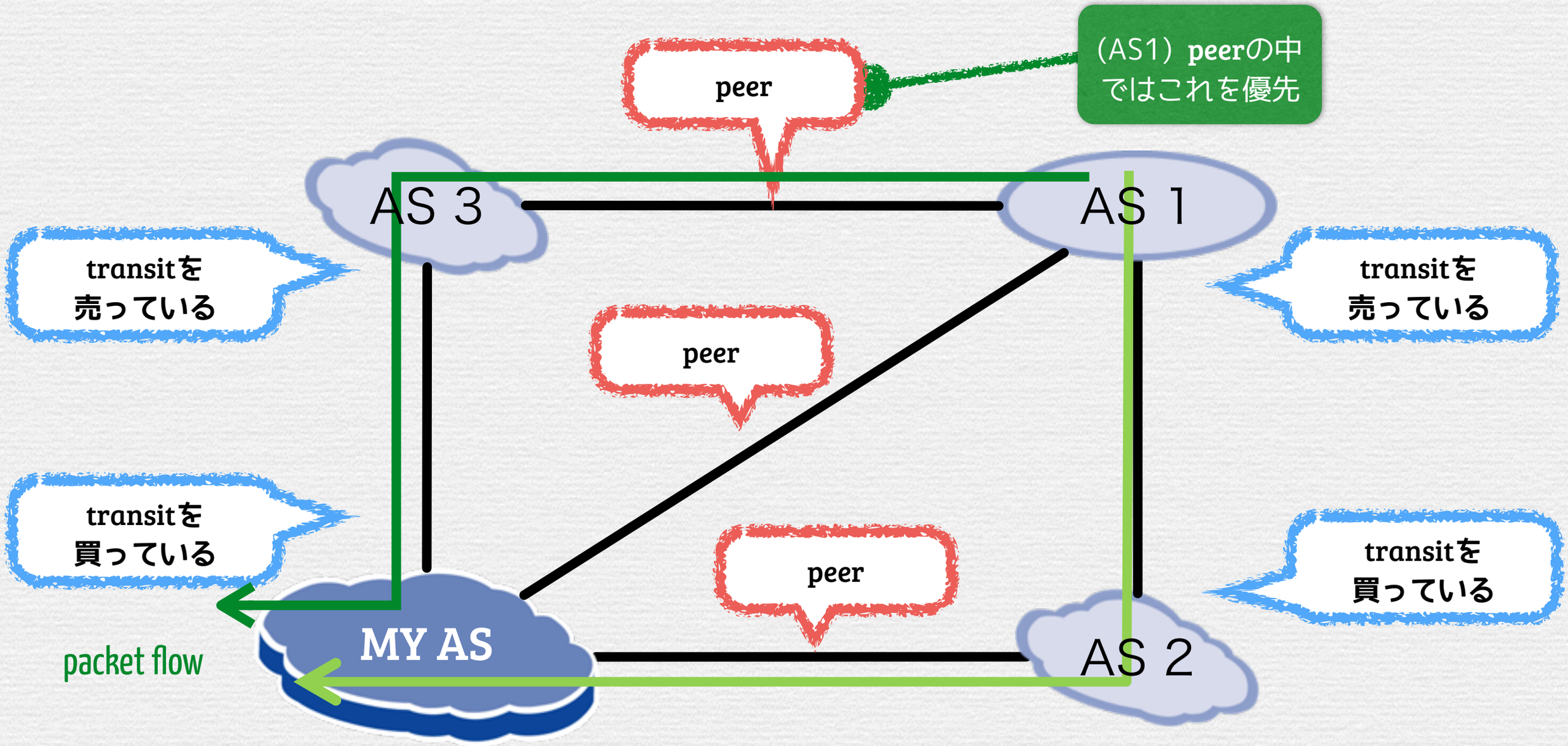
これから，運用上問題になったケースをいくつか挙げます

みなさんも

“自分ならどうするか”

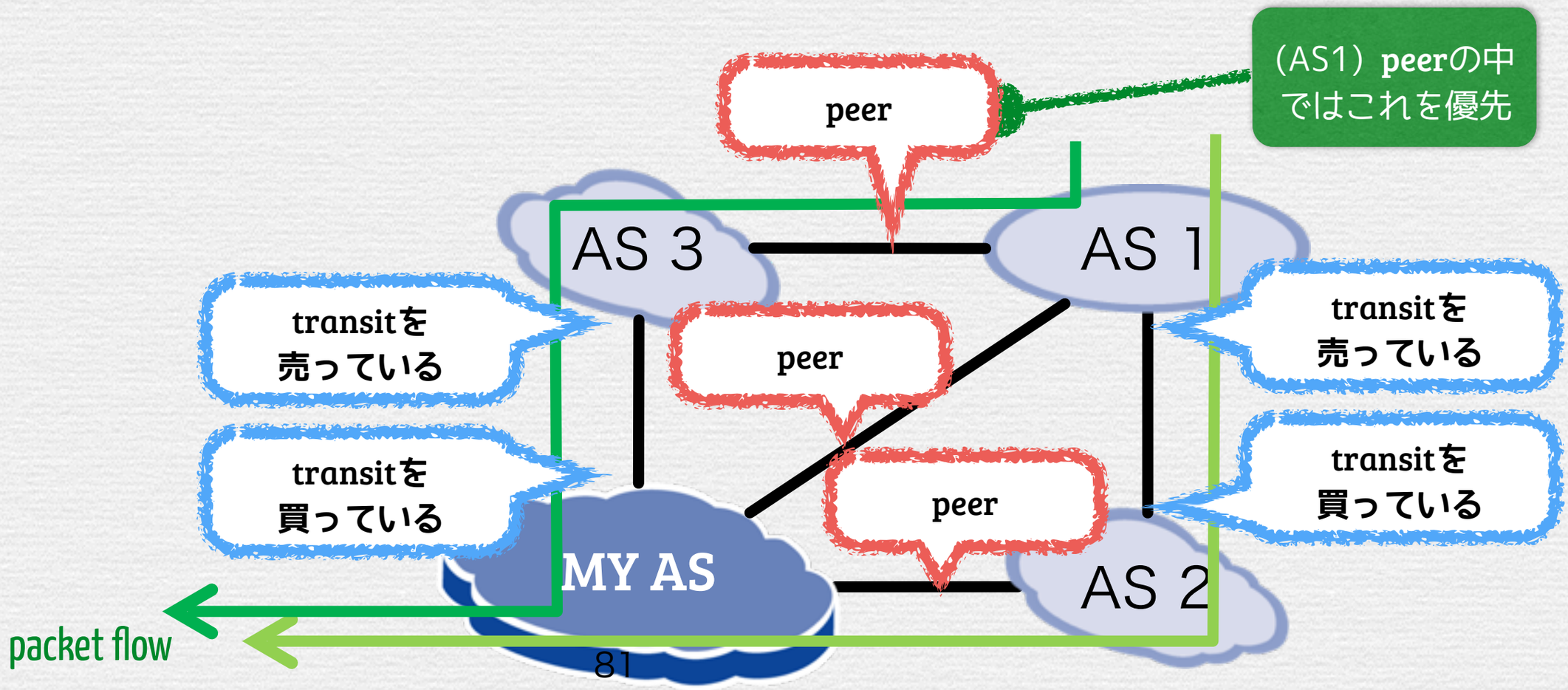
考えてみてください

問題1：直接peerしているのに trafficが流れてこない



問題1：直接peerしているのに trafficが流れてこない

1. AS1からではなく，AS3からtrafficが入ってくる
   - こちらはなんとかなるかも
2. もう一方は AS1の収益に影響するため困難



答え：直接peerしているのに trafficが流れてこない

○ AS1にメールする（またはAS3にメールする）

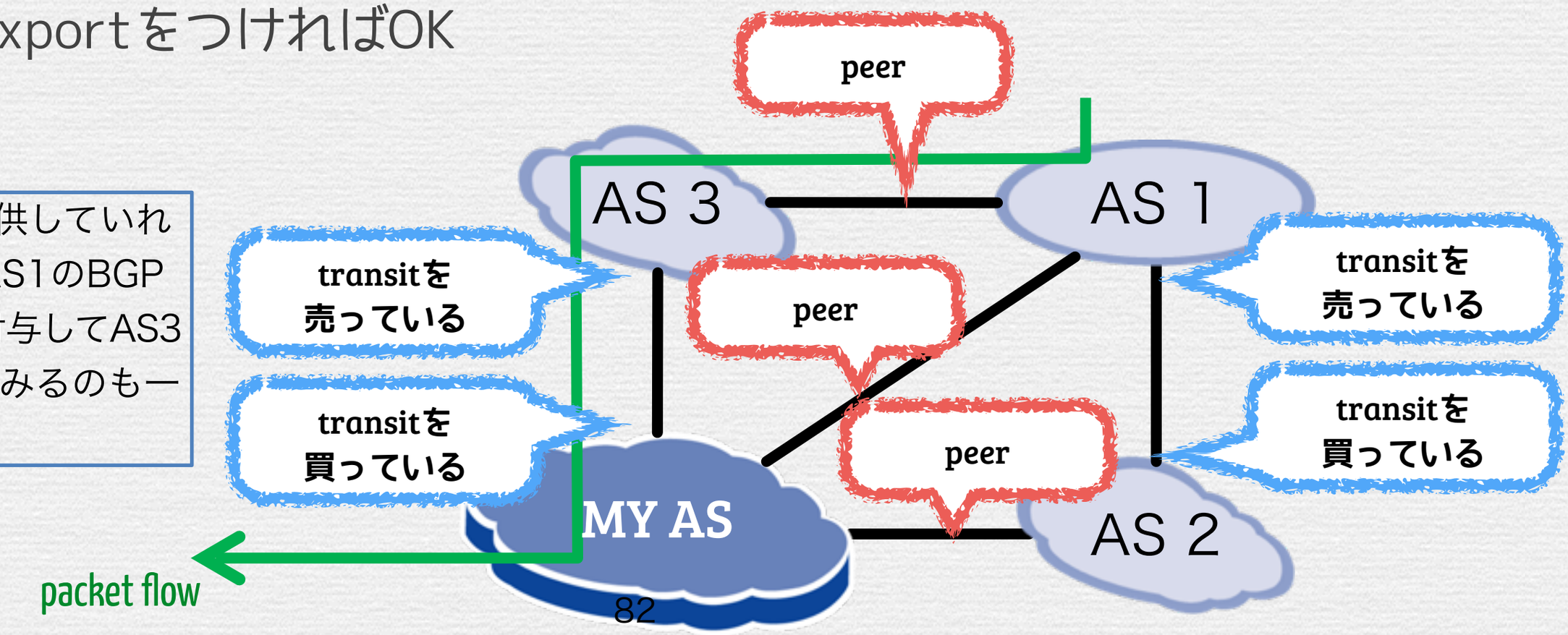
✖ AS3への経路広告を操作する ➔ transit全体に影響するので，
基本的には良くない

○(もし提供していれば) AS3のBGP communityを使い，AS1に対して経路を止める

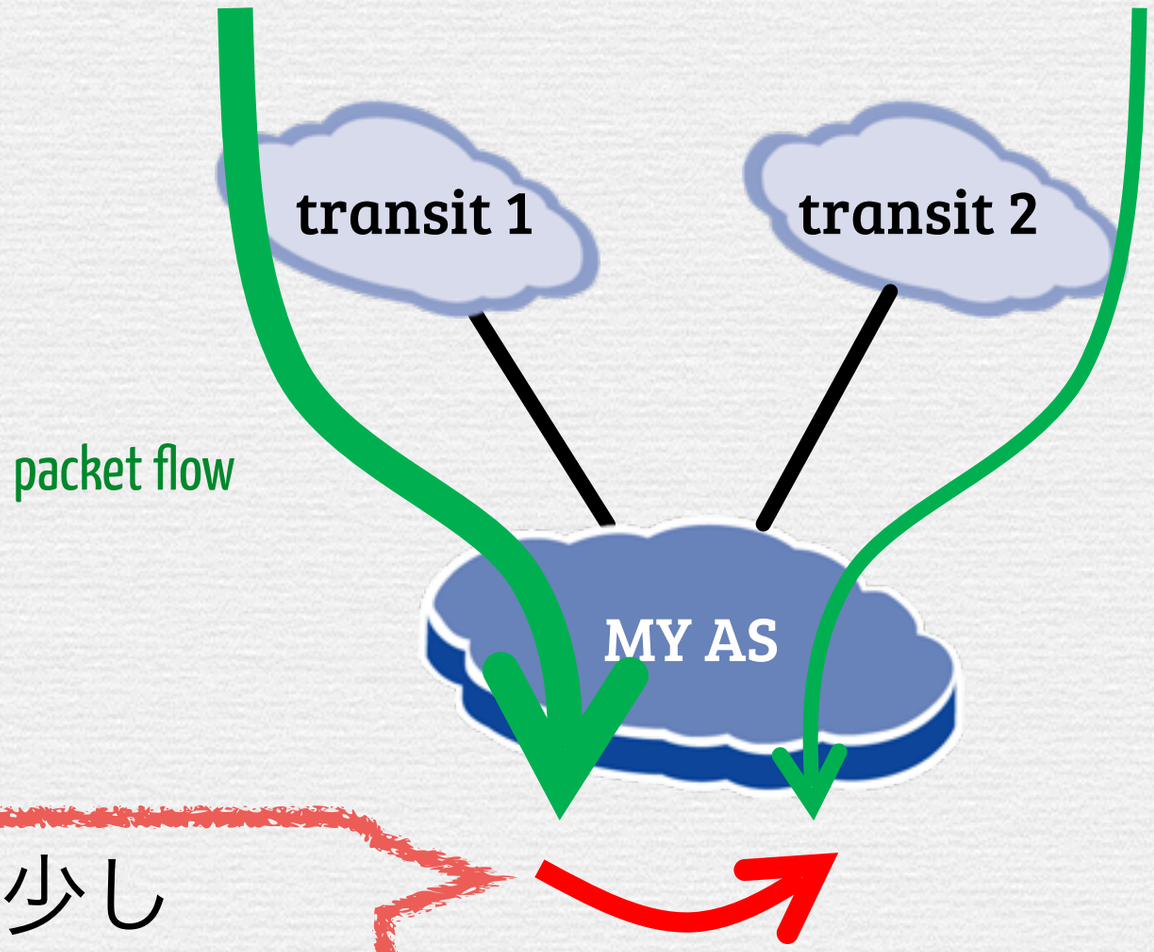
△ AS1への経路広告をmore specificに

- AS3 ➔ MY AS へのtrafficなど，対象外のtrafficも引き込んでしまう
- no-exportをつければOK

(もしAS1 が提供していれば)ダメもとでAS1のBGP communityを付与してAS3に経路広告してみるのも一手



問題2：transit間で trafficを動かしたい



答え：transit間で trafficを動かしたい

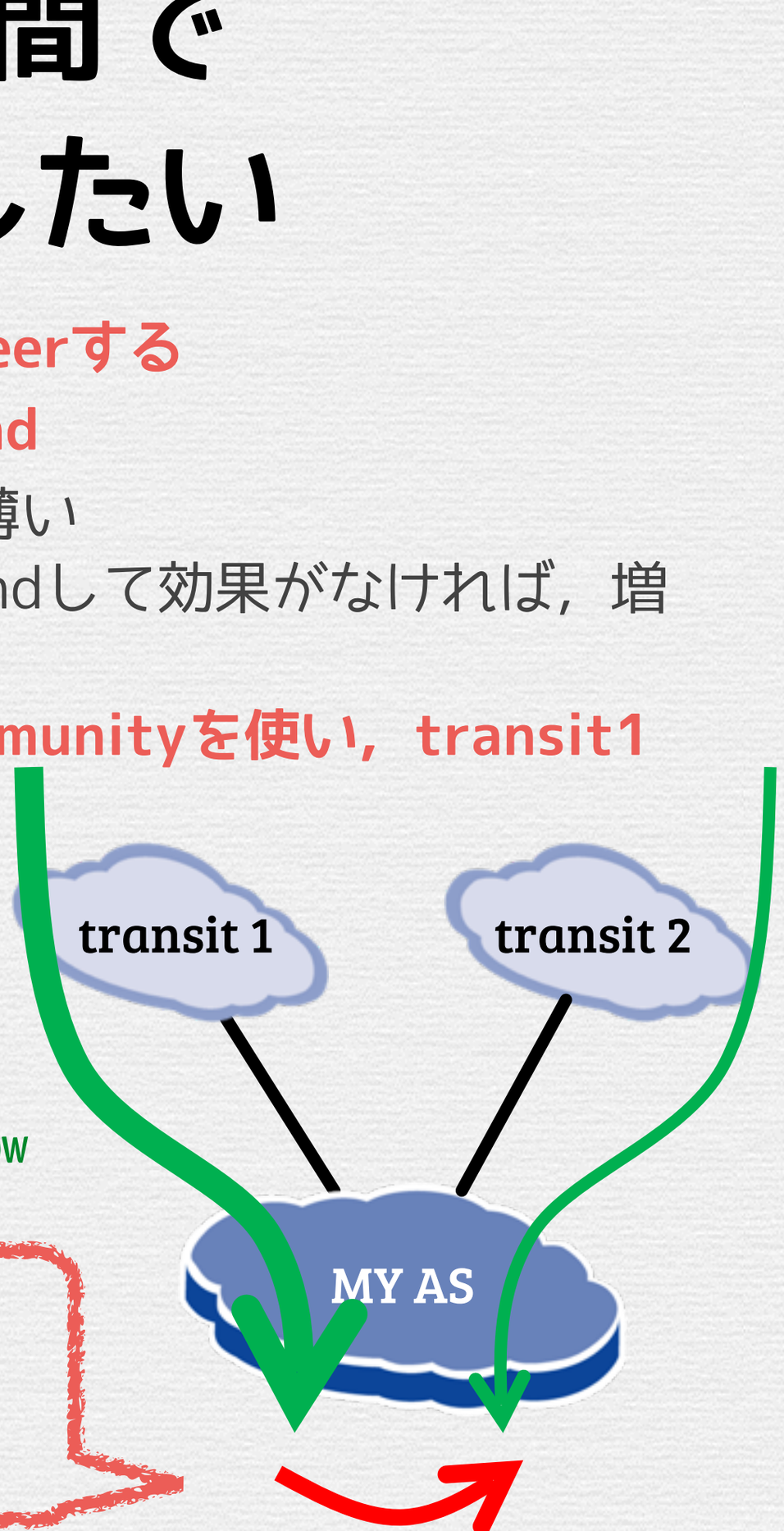
〇(特定ASからのtrafficが大きい場合）直接peerする

〇transit1への経路広告時にAS_PATH prepend

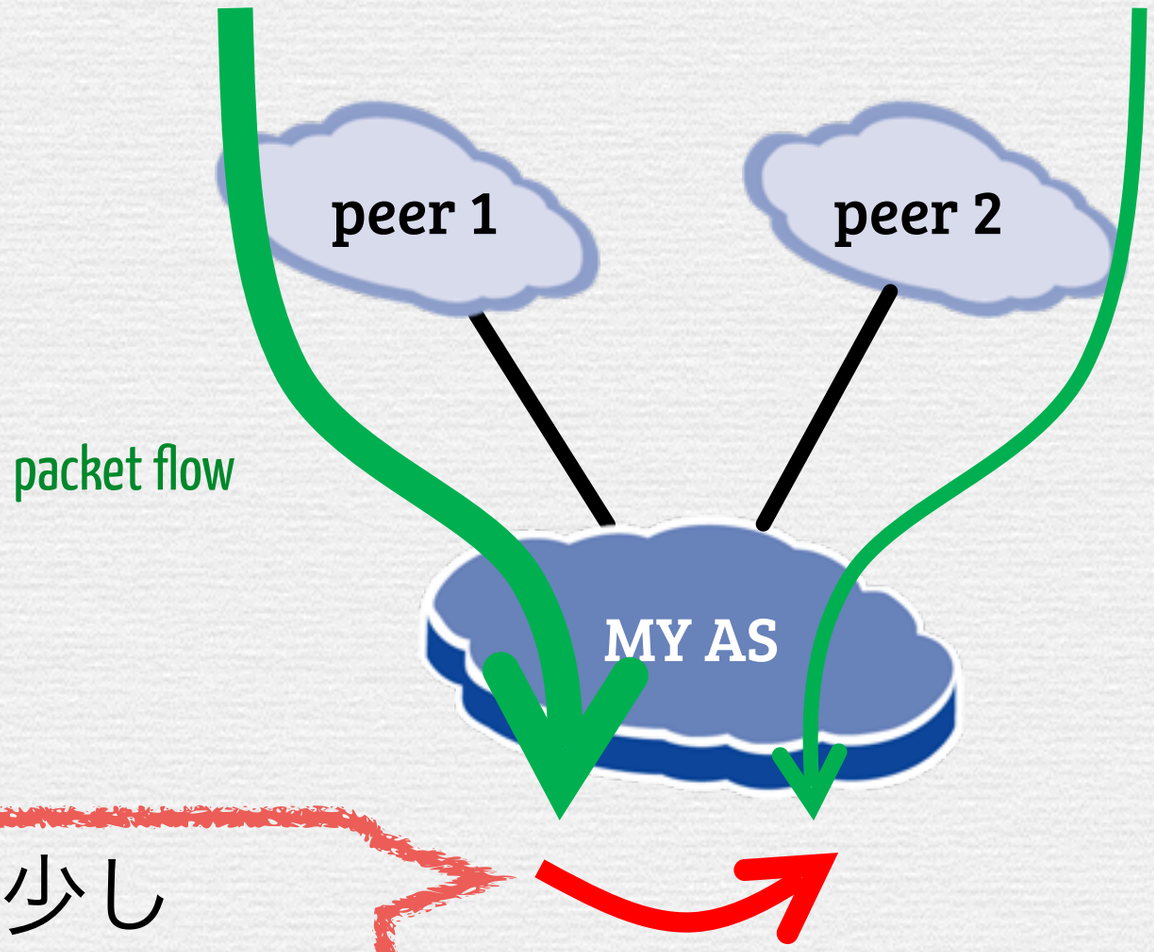
- 経験的にいくつもprependしても効果は薄い
  - 経験的には限界は+3程度．+3 prependして効果がなければ，増やしてもたぶん同じ

〇(もし提供していれば）transit1のBGP communityを使い，transit1内でのLPを下げる

△ 丁度いいvolumeの経路について，
- transit2への広告をmore specificに
- transit1への経路広告を止める



問題3: peer間でtrafficを動かしたい



答え：peer間でtrafficを動かしたい

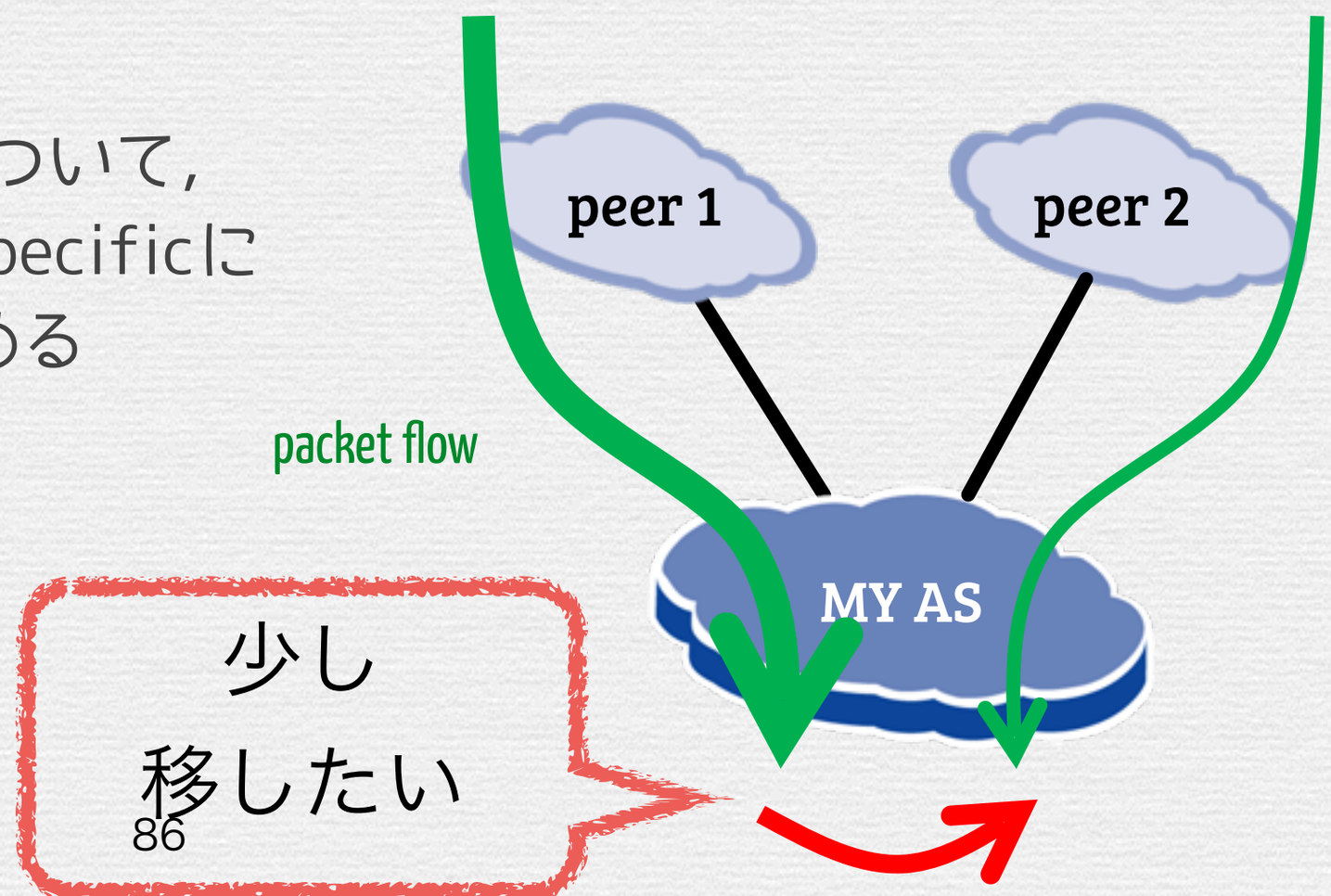
○ peer1への経路広告時にAS_PATH prepend

- 経験的にいくつもprependしても効果は薄い
  - 経験的には限界は+3程度．+3 prependして効果がなければ，増やしてもたぶん同じ

○（もしpeer1と複数peerしているなら）trafficをどけたいpeer1とのeBGP sessionでのみ特定の経路広告を止める

△ 丁度いいvolume の経路について，

- peer2への広告をmore specificに
- peer1への経路広告を止める

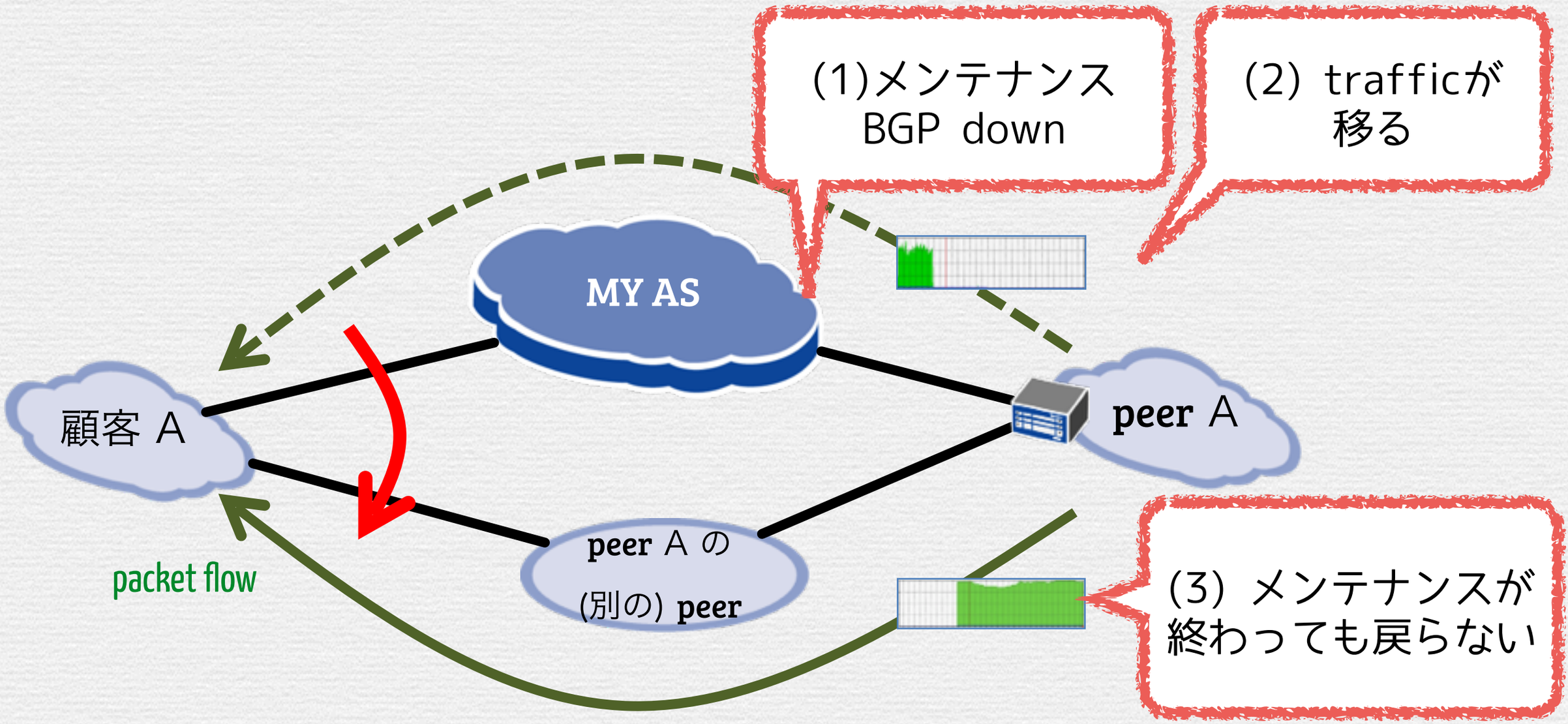


補足：peer間でtrafficを動かしたい

peer間でTEしたい理由

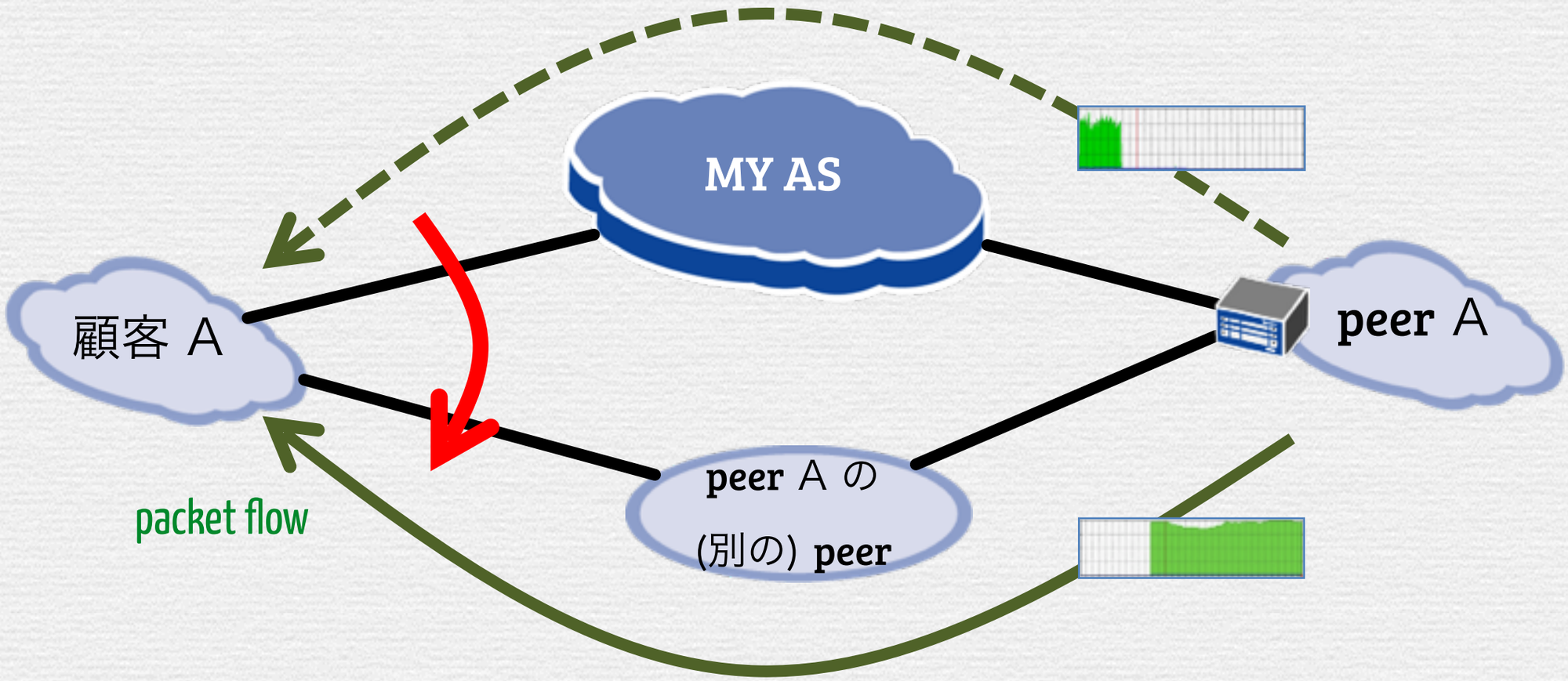
- **品質が悪いから**
  - 時間帯により輻輳する
  - latencyが大きい
- **paid peerだから**

問題4: peer間でよくあるtraffic移動



答え?: peer間などでよくあるtraffic移動

- 戻すのは困難



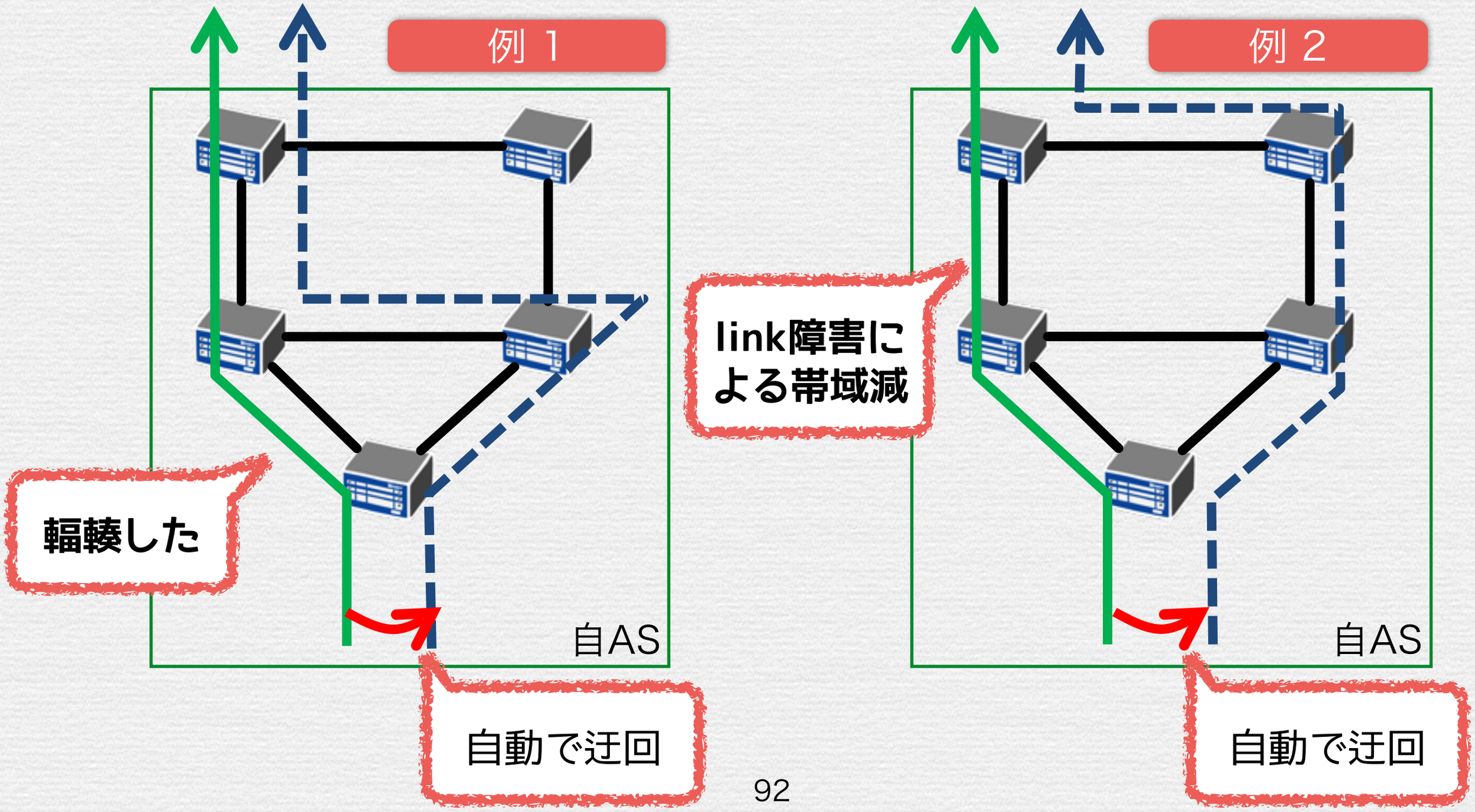
- eBGP間のbest path selectionはrouter IDではなく“経路の生存期間”で決定する場合が多い

MPLS-TE

MPLS-TE

- **AS内のtrafficを自動で制御する技術**
  - 空き帯域を自動で探し，そこへrouting
  - QoSも可能
- **MPLS(Multi-Protocol Label Switching) + RSVP(ReSerVation Protocol)**
- **IP PacketにlabelをつけてカプセルLSP化**
  - 仮想的なトンネル(LSP: Label Switching Path)をつくる

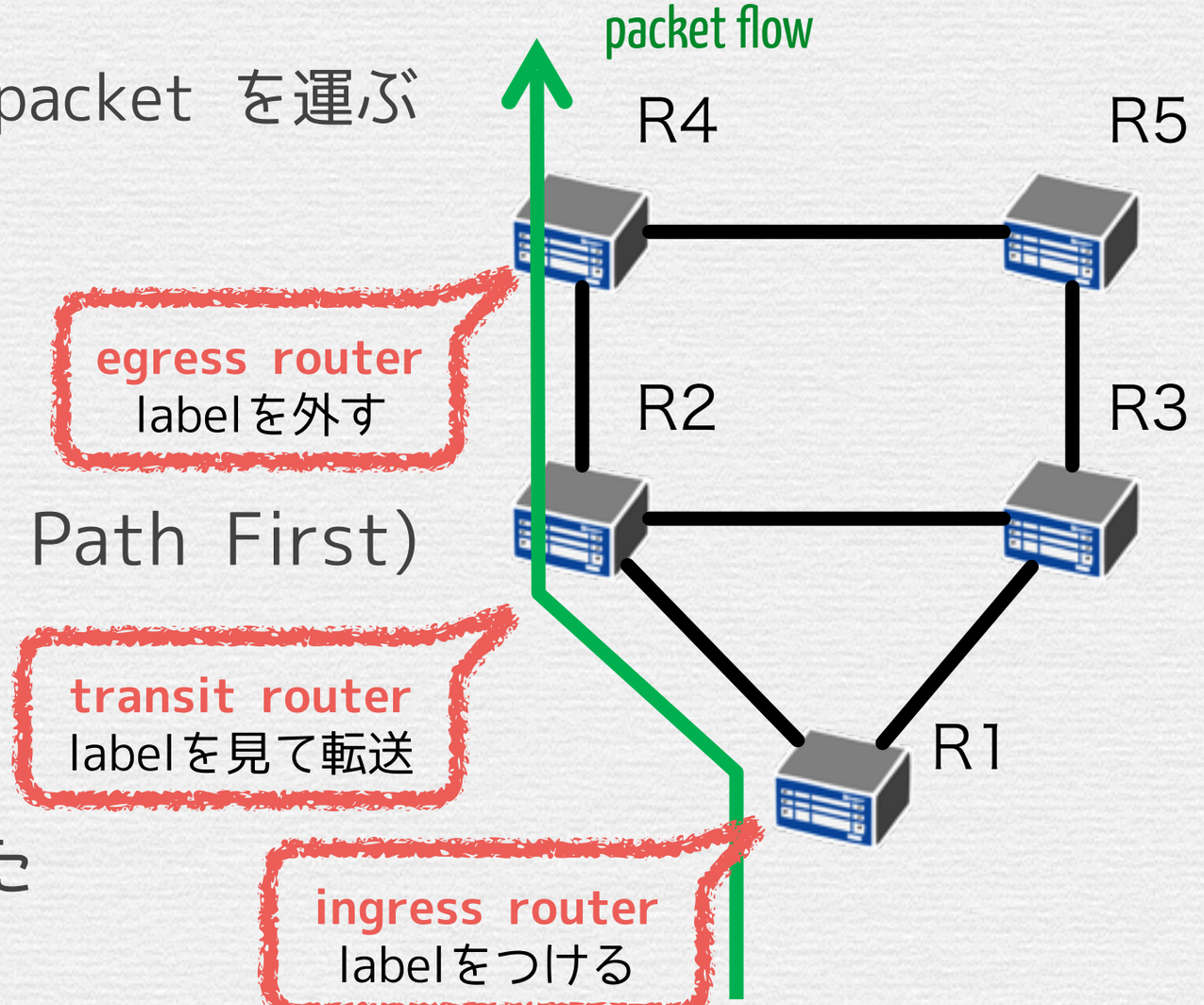
trafficのend-pointは変わらず，rerouteする



MPLS + RSVP のしくみ

- **LSPはLSR(Label Switching Router: MPLSを設定するrouter)群で 2x full mesh分 張る**
  - LSPは片方向通信のみ
    - 通信方向が異なると 別のLSPがpacket を運ぶ
    - R1 → R2 → R4
    - R4 → R2 → R1

    は別物
- **LSPを張る前にRSVP signaling**
  - CSPF(Constrainted Shortest Path First)
    - Link State型のIGPに依存する
      - OSPF
      - ISIS
  - LSP毎にあらかじめ設定された帯域が確保できるかを調べる



MPLS-TE

- **network eventにより帯域が縮退**
  1. RSVP signaling
  2. LSPが空き帯域に移る
- **自動で空き帯域を探索してくれるので，手動によるTE不要**
- **LSPが落ちても即packet lossではない**
  - IGPにfallbackする(BGP のprotocol nexthopの解決のために LSP + IGPを使う. 優先度が LSP > IGP)
- **fast reroute**
  - LSPが落ちたときのconvergenceを早くするために，あらかじめbackup LSPを張っておく

```
koji@test-router> show route 192.0.2.0/24

inet.0: 57 destinations, 57 routes (57 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.0.2.0/24    *[BGP/170] 5d 07:12:01, localpref 100, from 192.0.2.1
                   AS path: I
                 > to 198.51.100.1 via ae1.0, label-switched-path r1-r5-00


koji@test-router> show route 192.0.2.1

inet.3: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.0.2.1/32    *[RSVP/7] 5d 06:47:49, metric 16
                 > to 198.51.100.1 via ae1.0, label-switched-path r1-r5-00
                 [IS-IS/18] 5d 06:47:48, metric 16
                 > to 198.51.100.1 via ae1.0, label-switched-path r1-r5-00
```

MPLS-TE

- **利点**
  - 手動TEからの開放
  - 帯域設計がラク
  - 最悪rerouteしてくれる
- **欠点**
  - overlay model
    - label overhead
  - **LSP sizeを設定するため，日々LSP毎のtraffic量をカウントする必要がある**
    - automatic bandwidth に期待

なんだか便利そうですよね?

JANOG33

- グローバルインターネットにおける大容量トラヒックコントロールとリソースマネジメント
  http://www.janog.gr.jp/meeting/janog33/program/gin.html

- MPLSトラフィックエンジニアリングチュートリアル
  http://www.janog.gr.jp/meeting/janog33/tutorial/mpls.html

ここまで，
Traffic Engineering
について話しました

Questions ?

ルーティング
セキュリティ

- **Mis-Origin（経路ハイジャック）**
  - オペミス
  - MITM
    http://www.renesys.com/2013/11/mitm-internet-hijacking/

- **DDoS**
  - アンプ攻撃
    https://ripe68.ripe.net/presentations/227-RIPE68_2014_CRossow_Amplification_stripped.pdf

DNSキャッシュPoisoning，HeartBleed，SPAM/Virus，MITB などは ”ルーティング” とは違う

Mis-Origin

(経路ハイ ジャ ック)

Mis-Origin（経路ハイジャック）の検知

- **経路奉行**
  http://www.nic.ad.jp/ja/ip/irr/jpirr_exp.html
  - 無料

- **BGPMon**
  http://www.bgpmon.net/
  - 5prefix を超えると有料
  - API
  - twitter
  - blog

Mis-Origin（経路ハイジャック）の対策

- **他AS がMis-Origin されている**
  → Mis-Origin 経路を網内に入れない
  - **Route Filter**
  - RPKI

- **自AS がMis-Origin されている**
  → 奪われた到達性を奪い返す
  - **more specific な経路を流す**

RPKI

- Origin Validation はどんどん実装されているが．．．
  - 現状，まだ普及しているとは言い難い
- 選択肢
  - 自分でROA Cache Server を運用する
    http://www.janog.gr.jp/wp/rpki-routing-wg/
  - ROA Cache Service を待つ

more specific な経路を流す

奪われた到達性を奪い返す

- 本当に流れるか分からないので，**素振り重要**
  - Peering パートナー，トランジット事業者が経路を受け取ってくれないかもしれない

DDoS

DDoSの検知

- **既存のサービス監視 / リソース監視**
  - CPU / Memory / TCP connection / bandwidth のオーバーフロー，ログなどによるリクエスト数の監視
- **トラフィック監視**
  - 商用アプライアンス / サービス
  - OSS
    - flowtools
    - **exaddos**

DDoSの検知より対策が重要.
トラブルの原因がDDoSだと分かった時，対処できるように.

DDoSの予防

- **Anti IP-Spoofing (BCP38)**
  - **uRPF**
  - packet filter
  - http://www.ietf.org/rfc/rfc2827.txt
    http://www.bcp38.info/

- **アンプ除去**
  オープンリゾルバー ないですよね?
  - NTP 大丈夫?
  - SNMP は?

DDoSの対策

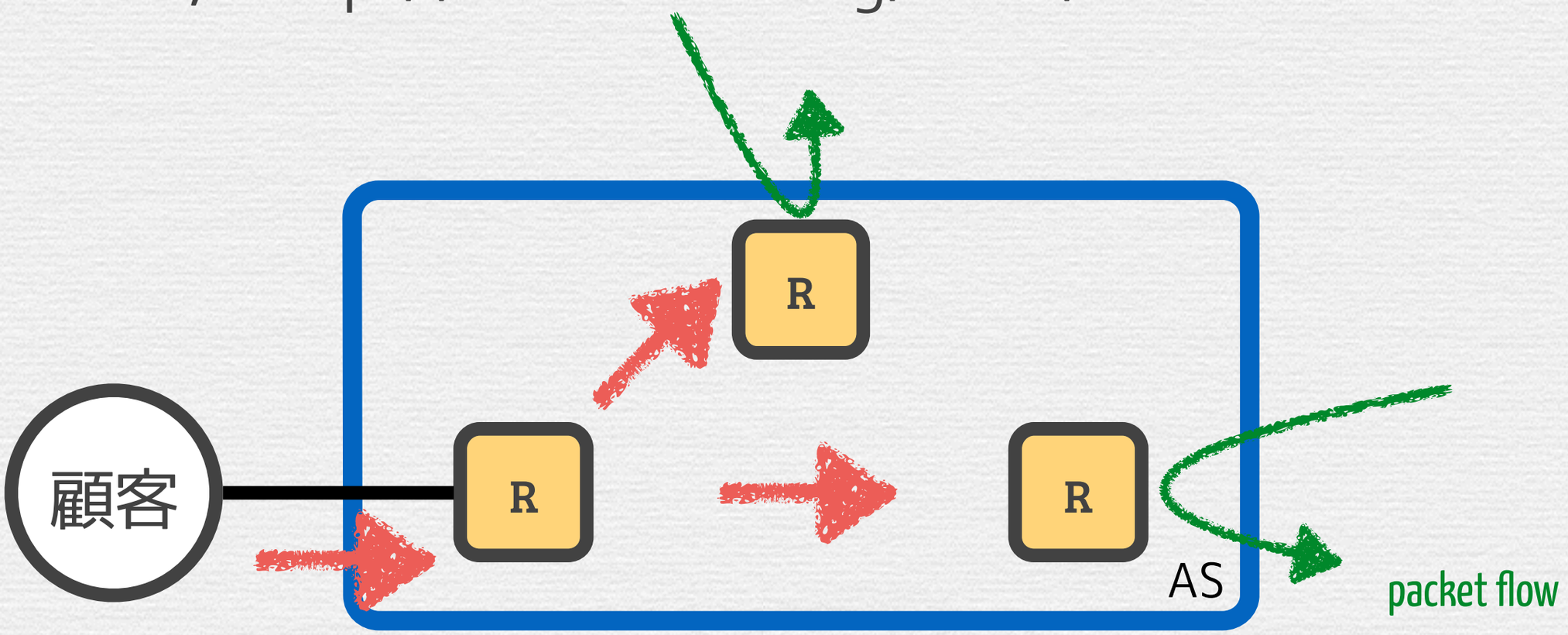
- **Victim 上のサービスを止めてもよい**
  - トランジット事業者へのエスカレーション
  - **Remote Triggered Blackholing**
  - GSLBを使った付加分散と連動できれば理想的
    http://www.slideshare.net/AmazonWebServices/ddos-resiliency-with-amazon-web-services-sec305-aws-reinvent-2013

- **Victim 上のサービスを止めない**
  - サーバー上での対策 (RRLなど)
  - DDoS Mitigation アプライアンス
  - DDoS Mitigation サービス
    - セキュリティプロバイダー
    - トランジットの付加サービス

Remote Triggered Blackholing

特定のBGP Community が付いた経路を受信している間は，
該当経路宛てパケットをネットワークの全エッジルーターで
blackhole するしくみ

RFC5635, http://tools.ietf.org/html/rfc5635



x.x.x.x/32

community: yyyy:zzz

ルーティング

エコシステム

標準化
IETF
ベンダー
メーカー
ICANN
DC事業者
HW
ラック
アドレス
ASN
RIR
NIR
LIR
WCIT
IGF
回線
キャリア
法令
総務省
ガバナンス
トランジット，IX
各種サービス
情報
各種サービス
事業者
IaaS
traffic
ネットワーク
コミュニティ
クラウド
事業者
顧客
peer
111

IRR:
transit providerのfilterを制御する

どこかのInternet Routing Registry(IRR) に登録する必要がある

- 日本でよく使われているもの
  - **JPIRR (jpirr.nic.ad.jp)**
    - JPNIC 管理下のアドレス保有者
  - **RADB (whois.radb.net)**
    - 有償 ($500/y)
  - **NTTCOM (rr.ntt.net)**
    - ntt.net ユーザーのみ

IRR

- http://www.nic.ad.jp/doc/jpnic-01077.html
- お互いにmirror しているため，基本どこか1カ所に登録すればよい
- 冗長化のため +1カ所登録すると安心

		このIRR の情報を持っているか?		
		JPIRR	RADB	NTTCOM
このIRRが	JPIRR	○	○	×
	RADB	○	○	○
	NTTCOM	○	○	○

peering db

“細かいことはpeering dbを見てください”

ですむので便利

- www.peeringdb.com
- 情報閲覧はguestアカウントでOK
- 自ASの情報を登録するには
  ユーザー登録 ➔データベース運営者の承認が必要
  - mail addressのdomainを見られるので，ASとの関連が明らかなmail addressを使う

peering db には いろいろ記載できる

- ASの基本情報
  - 名前
  - ASN
  - 種別(ISP / ICP など)
  - traffic level
  - looking glass / route server URL
  - ipv4 / multicast / ipv6
- **peering policy**
  - open / selective / restrictive / No
  - 複数拠点でのpeerを条件にするか?
  - in / outのtraffic比率を条件にするか?
- 連絡先
- 接続IX (public peer 用)
  - 名前
  - 帯域
- POP (private peer 用)
  - DC名
  - Interface 種別

Questions

付録:
バッド?
ノウハウ集

(応募いただいたみなさま,
ありがとうございました)

cisco のサイトでドキュメントを探すときに一番効率のいい方法

> http://www.cisco.com/cisco/web/psa/default.html?mode=prod

こっから探しますね。
ここからそれぞれのOSのconfiguration guide/comannd reference に行くのが一番固い気がします。

— Cisco Systems 土屋さん

NATになっとった？

> あれはまだBGP以前で、PBRで上り選択をしていたころ、NTT系が売ってた写真立てみたいな、ある無線ルータ配下に、ある上位ISPのPAアドレスからグローバルIPを/29ほど割り当てて、配下の端末からtracerouteしたところ、なぜか、別の上位ISPに上っていく現象。
>
> ルーターのメーカーに聞いても謎で、ダメモトで上位ISPに問い合わせようとした矢先、自分の個人掲示板にアクセスした際に表示されたIPアドレスが、割り当てたものと違う・・・！
>
> 実はこの無線LANルータは、NATをしていたのでした。

トラフィックを頭打たせた
ーウチだけ編

> PBRからBGP運用に変わり、当時高かったATM回線の間でバランス取りをしていたとき、「あとちょっと、/25分だけ、ここの回線に」と思って、その回線のルータから /25を経路広報に追加したところ、大量に流入するトラフィックにより、頭打ちになりました。
>
> その/25は、その回線からしか出ていなかったのが原因です
>
> 後になって、「意外とみんな、/25の経路はフィルタしないのね」と思いました。

トラフィックを頭打たせた
ー国内編

> BGP運用も軌道に乗って数年、ルータ機種を変えるときに、新しいルータのIX接続分configで、BGPの外向きroute-mapが入っておらず、持っているルート全てを、某IXの各ピアに吐いてしまうという事態が発生しました。

横で見ていて、ポートのLEDが異常に光りっぱなしになり、もしやと思ってサマリーを確認してもらったら、ウチ以外のルートも吐いていました。

そして、多くのピア先の上位トランジットにうっかりなってしまったことで、大量に流入・排出するトラフィックにより、頭打ちになりました。

後になって、「意外とみんな、完全な経路フィルタは書かないのね」と思いました。

普段気をつけている
ポイント

“

- 新規ピア／トランジットは、shutdown状態で十分確認、sendする経路数も。
- 物理ルートも気を付ける。キャリアダイバシティから踏み込んで、別条、異ルート・・
- ファイバ接続時は、必ずクリーナで処置。

“

来るべきIPv6時代に備えて、BGPルータのメモリ設定をIPv4/IPv6両対応の配分にしていたところ、IPv4の40万ルート時代が先に来てしまったという苦い話しもあるとかないとか。

FB の経路数が増えすぎて受けきれなくなって・・・・

> お客様からの問い合わせで到達できない経路が出てきていることがわかって，なんだろうと思って調べてみたらフルルートが機器のキャパを超えてこえていたということが昔ありました.

Hitachi GR2000 はわりと安価に手に入っていたのでよく使っていたのですが，メモリが少なく，為す術がなくなってしまったため，フルルートを諦めて default の 1 経路のみに切り替えて運用したりということをしたことがあります.

“

また、某Foundry製品では、

```
#show default values

System Parameters     Default     Maximum     Current

ip-cache              204800      524288      450000

ip-route              204800      524288      450000

(snip)w default values
```

といった感じで変更は可能なのですが、default値が小さすぎて、症状が出始めてからこれに気が付き、かといっていきなりMaxまであげても大丈夫なものなのかなとビクビクしながら数値を拡張していっていました。

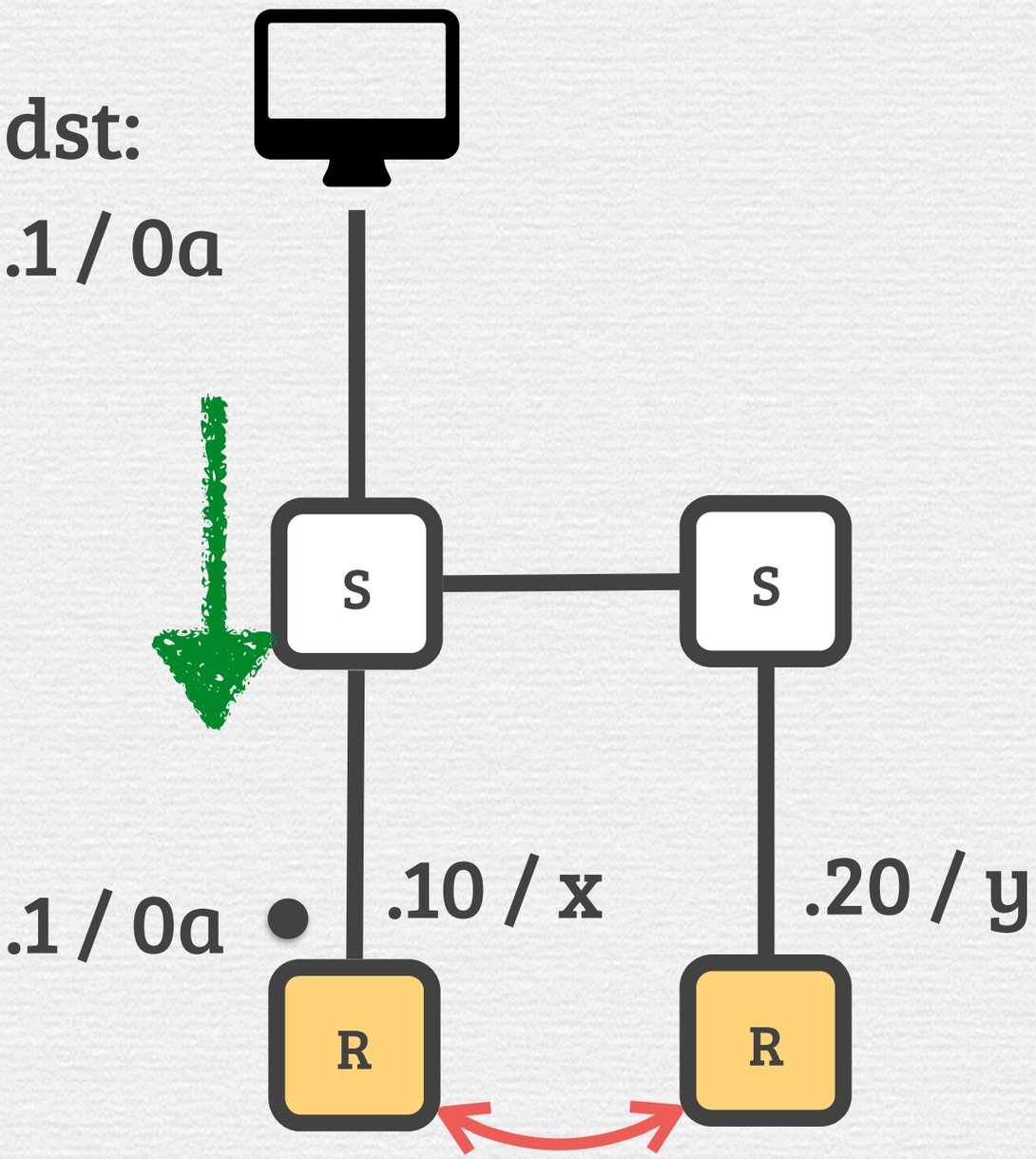
これ、今49万経路くらいだと思うのでもうしばらくしたらMaxでも溢れますね(^^;;

> 数年前なのでもぅ大丈夫だとおもいますが、IPv6のpeer(not transit)が原因でメモリが足りなくなったことがありました。
>
> peer先は、おそらく、address-fa..... ipv4がno activateされておらず、こちらの想定しないIPv4のフルルートがIPv6トランスポート経由で飛んできていました。
>
> それに気づくまで、フィルタをあまり厳密にかけていなかったので、peer up →コンバージェンスの最中にルータ(非力)がびっくりして、、、、、
>
> なんて事がありました。
>
> — JPNIC 岡田さん

無停止で
HSRPのグループIDだけを変更

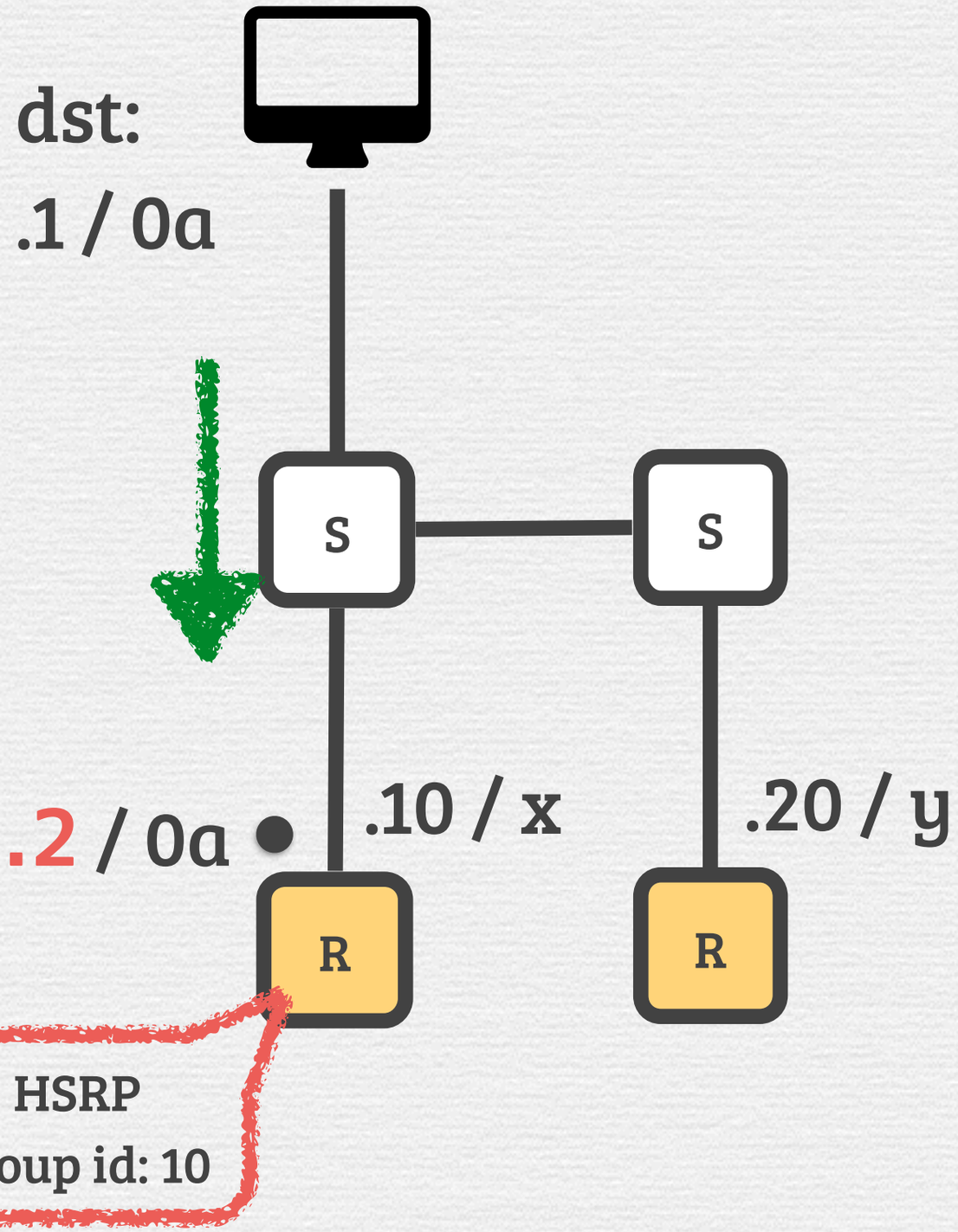
> ルーティングとはちょっと違うけど投げちゃえ。
>
> ふと思い付いて、当時は便利に使ってた無停止でHSRPのグループIDだけを変更する小技。
>
> [janog:12412] Re: [JANOG34チュートリアル] みなさんのノウハウも紹介させてください
>
> — IIJ 松崎さん

(0)

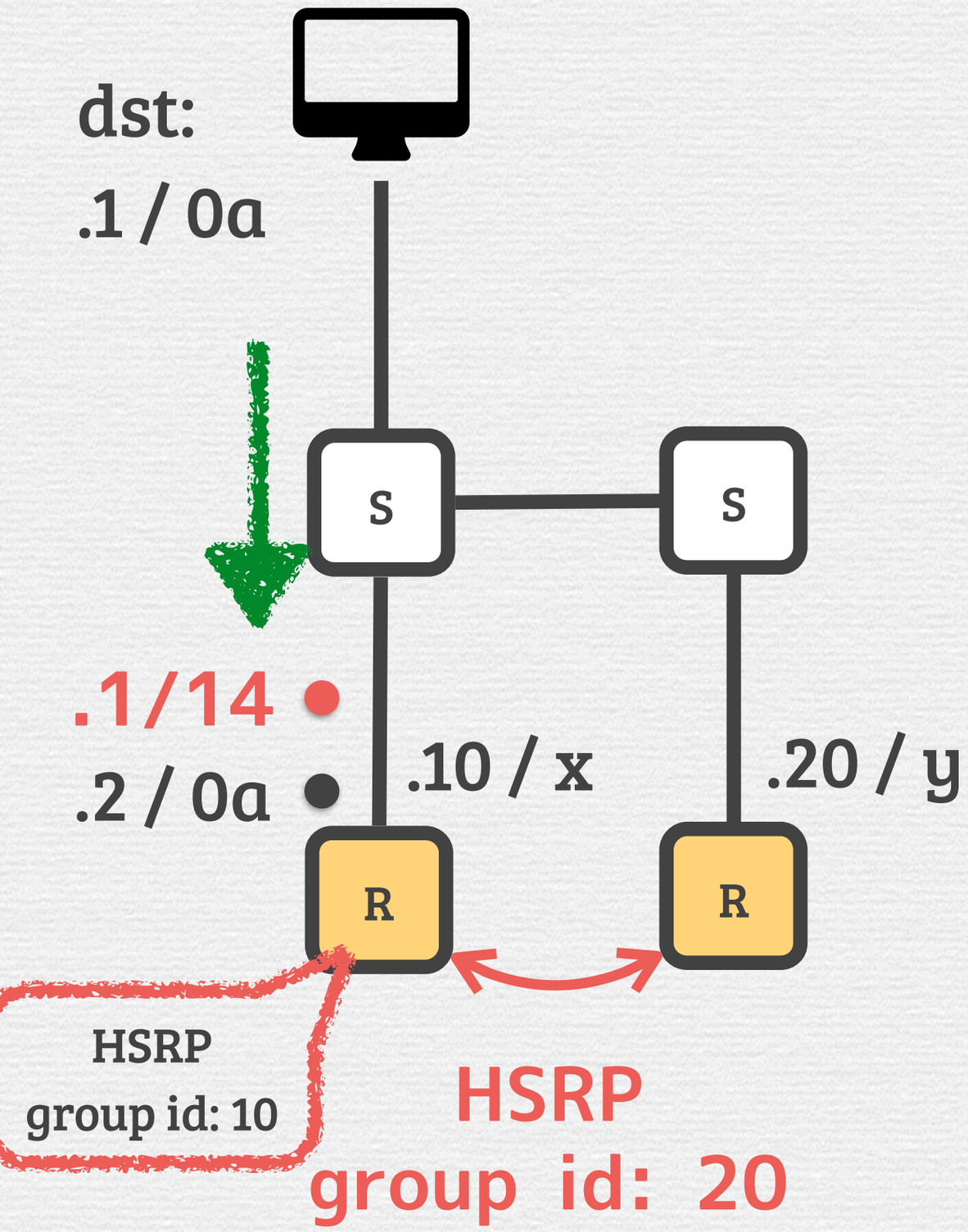


HSRP
group id: 10

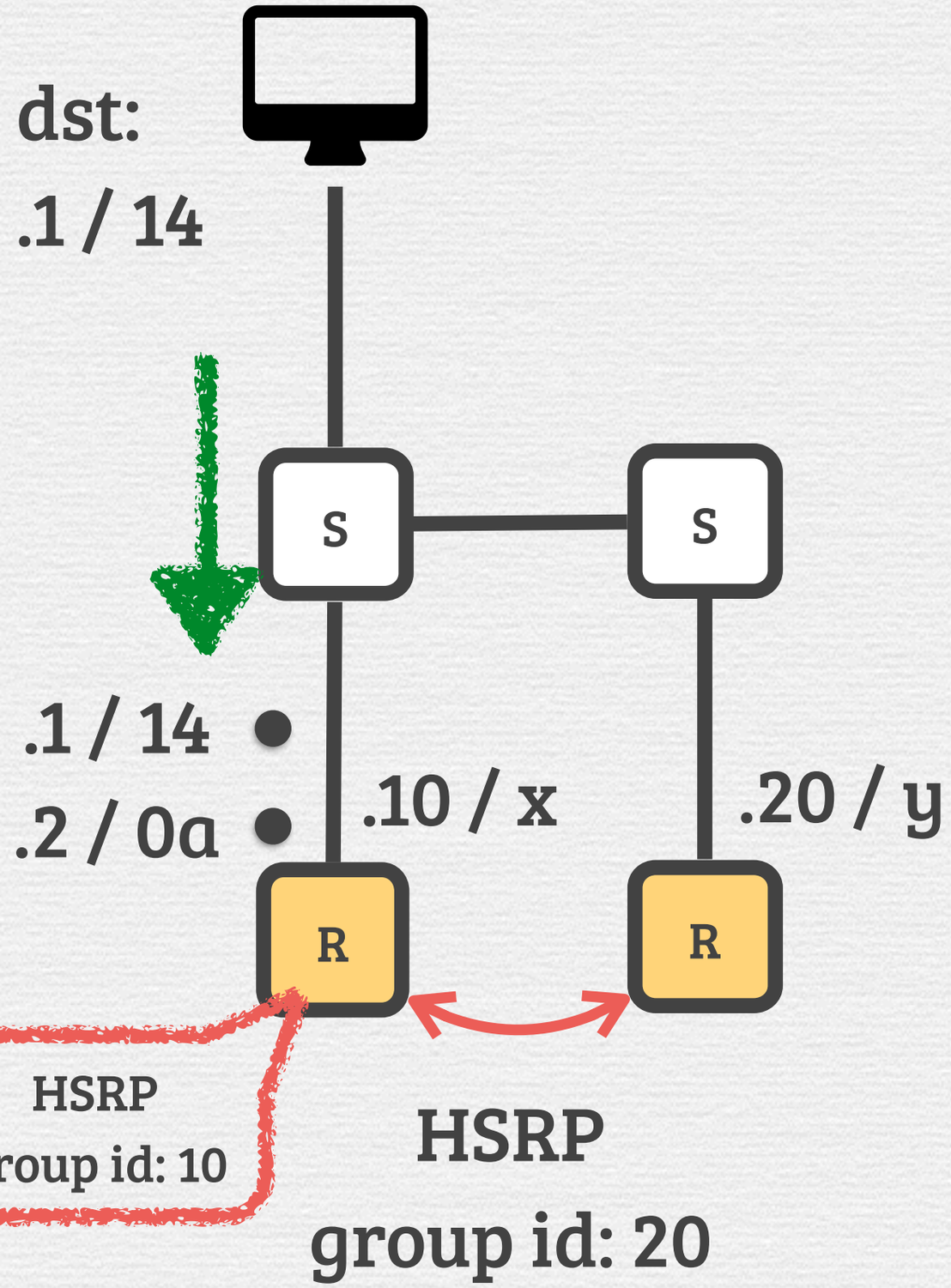
(1)



(2)


dst:
.1 / 0a
S
S
.1/14
.2 / 0a
.10 / x
.20 / y
R
R
HSRP
group id: 10
HSRP
group id: 20

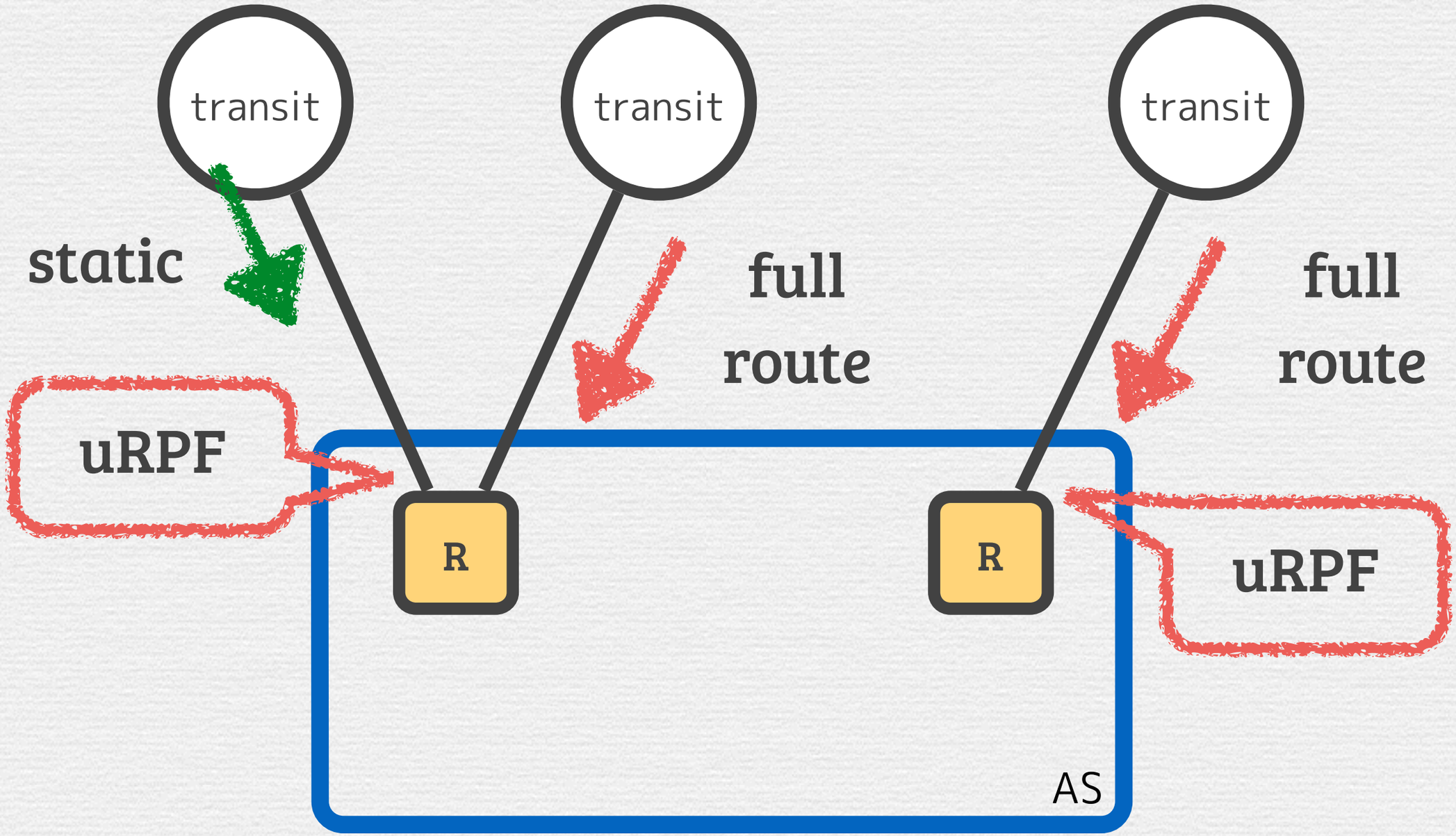
(3)


dst:
.1 / 14
S
S
.1 / 14
.2 / 0a
.10 / x
.20 / y
R
R
HSRP
group id: 10
HSRP
group id: 20

昔あるところに、歴史的経緯により自網内に自社PIアドレスと他社PAアドレスが同居したネットワークを運用している組織がありまして、これも歴史的経緯によりPAなstatic接続回線とBGPトランジット接続回線を同じルータに収容していました。PAのホストからインターネットに出ていく通信だけは、static回線がnexthopになるようにpolicy入れてる感じで。

ある時、uRPFという便利な機能を知ったネットワーク管理者は、static回線とトランジット回線の両方のinboundに、uRPF loose modeの設定を入れました。

数年後、トランジット回線業者からメンテナンスの通知がありまして、でもPAのネットワークは影響ないよね、と思っていたら、すっとこどっこい、トランジット回線切れたらPAネットワークもインターネットと通信ができなくなってしまいました。


transit
transit
transit
static
full
route
full
route
uRPF
R
R
uRPF
AS

昔やっちまったルーティングミス

> clear ip bgp *
> と打ってしまったとか、ですね。
>
> あと、リモートメンテしている時にルータのプロンプトのある画面でうっかりペースト操作しちまったら、全然関係ない画面でどっさりコピーしてたのがべったり貼られたとか。
>
> — ナインレイヤーズ 菊池さん

小ネタなら幾らかありますねー。例えば、コピペのときに、クリップボードが汚染されていたとか、fw設定しっぱいして、繋がらなくなったとかw

— @tomocha

特殊なネットワーク作り時にOSPFのコスト足し算を間違えます。おろ？という報告に迂回しちゃったり。事前に迂回試験をちゃんとやってればそういうミスも防げるのですが。
VLANインタフェースきってるところでよく間違えます。
標準パターンではあまりミスはないんですが
MEDを書き換えちゃいけないPeerで書き換えちゃったりとか、まぁ標準とは違うところでは間違いは起きやすいですね。

あと大した影響はないですがoriginが?になったまま経路出しちゃったりとか。

— Biglobe 川村さん

> ミスの部類に入るかは微妙ですが、ホットポテト採用しているのを気にせずremote peerはってるとそのルータの中ではremote peerがベストパスになってしまって遠い方にまわってしまったりとか。remote peeringってIGPのメトリックが使えないのですよね。
>
> でも一番怖いのはStatic routeですね。
> Staticちょー怖い。あれ？迂回しない？
> とかポヨッと変な経路が浮かんで来たりすると暫定対策で入れたつもりの管理しきれてない古いstaticがそこにあったりします。
>
> — Biglobe 川村さん

BGPのMEDが勝手についてた

> MEDをあまり意識してなくって、route-mapで設定してなかったんですが。。。
> トラフィックが偏るんで、なんでだろ？と思ったら勝手にMEDが付いていました
>
> 犯人を捜したら、実はOSPFのmetricをMEDにコピーしてるということが判明(ciscoの仕様)
> ちゃんとMEDを意識しないとだめだね!!
>
> — 富士通KCN 山口さん