

BGP4

プロトコルの概要と運用



IIJ

(株) インターネットイニシアティブ

浅羽登志也

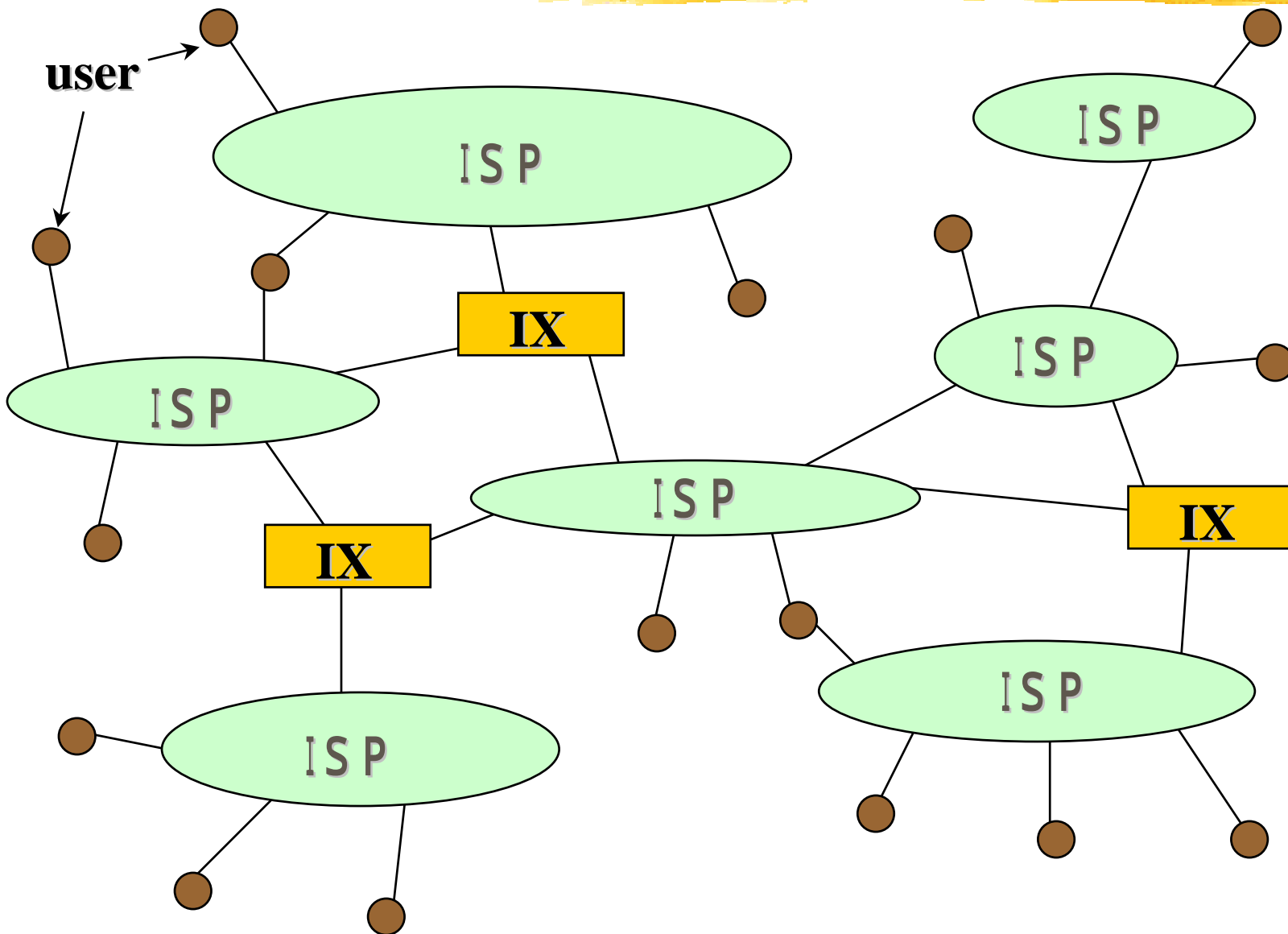
asaba@ij.ad.jp

グローバルな経路制御の概要



現状、問題点、解決策

インターネット全体の構造



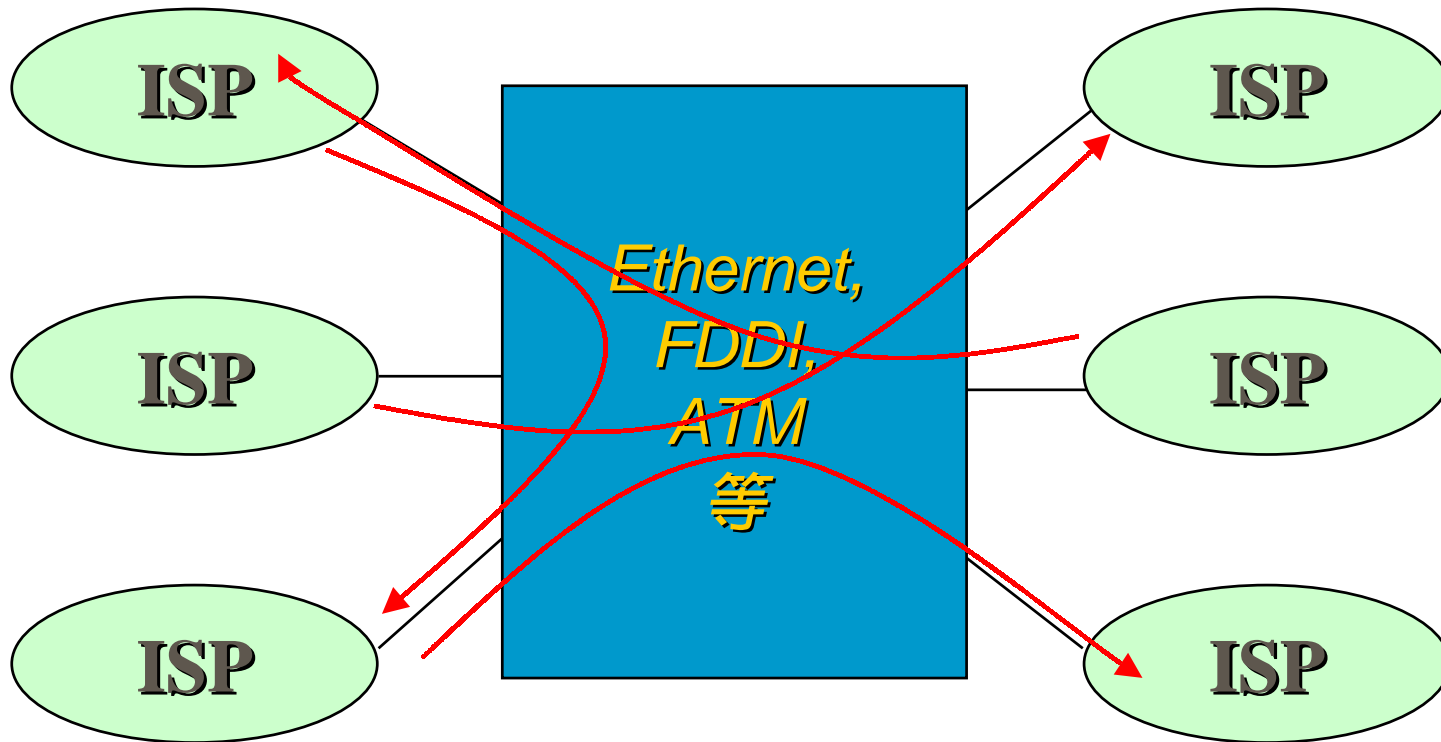
ISPとは？

- インターネットへのコネクティビティを提供
- 複数のISP同士が相互接続して全体を構成
 - “インターネット”
 - 相互接続形態
 - IX経由の接続
 - 直接接続
- ユーザはいずれかのISP経由でコネクティビティを得る

インターネットエクステンジ

- 複数ISP間の相互接続を提供するサービス
 - IX (Internet eXchange)
 - ISP同士がトラフィックを交換する場
 - イーサネット、FDDI、ATMなどのマルチアクセス型のデータリンク接続サービス
 - 同じデータリンクメディアを經由して複数ISPと接続が可能
 - 例
 - Network Access Point (NAP)
 - Metropolitan Area Exchange (MAE)
 - LINX, NSPIXP, JPIX, MEX, HKIX, etc.

IXの概念図



経路制御とは？

- インターネットに接続された任意の2ユーザ間の、ネットワーク層での接続性の確立
 - アドレッシング
 - 経路情報の交換
- インターネット上のトラフィックの流れの制御
 - ロードバランス
 - 代替経路の選択
 - ボトルネックの解消

経路制御の階層

- 2階層の経路制御
 - ISPの内部、ISP間
- Interior Gateway (or Routing) Protocol (IGP)
 - コストに基づく経路選択
 - OSPF, RIP2
- Exterior Gateway (or Routing) Protocol (EGP)
 - ポリシーに基づく経路選択
 - BGP4

スケーラビリティの問題

- 2つの問題
 - アドレス空間の枯渇
 - 経路表(ルーティングテーブル)の爆発
- 短期的解決策
 - CIDR (Class-less Inter-Domain Routing) の推進
 - プライベートアドレスの活用 (RFC1918)
- 長期的解決策
 - IPv6 (RFC1883)
 - アドレス空間の拡張(32ビット 128ビット)
 - 階層的なアドレス割当と経路制御の推進

Classless Inter-Domain Routing

■ 目的

- クラスの概念による弊害の払拭
- IPv4のアドレススペースの有効利用
- 経路表のエントリ数の縮小

■ 階層的アドレス割当

- ビット境界に促したアドレス割当

■ 経路情報の集成

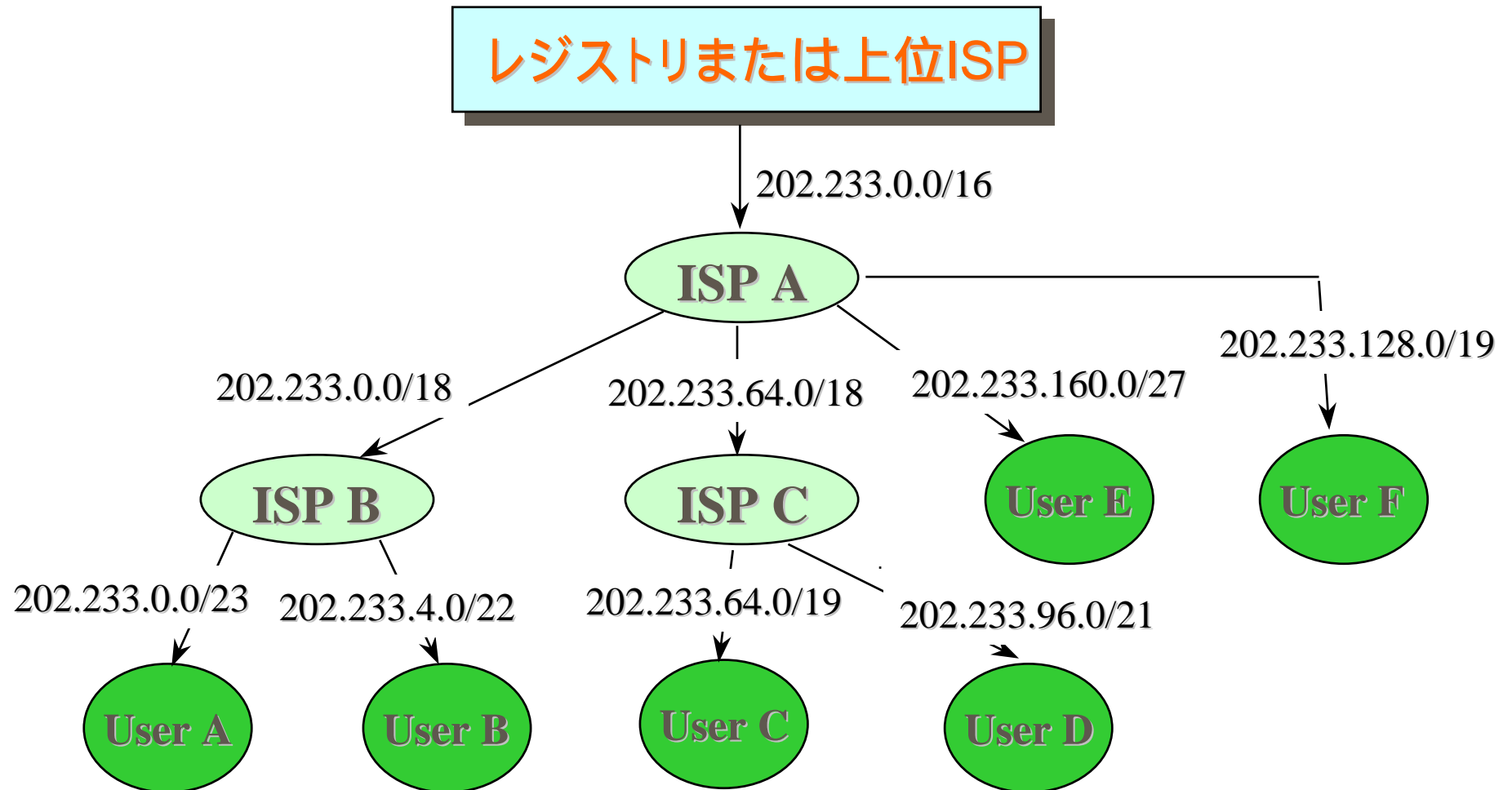
■ アドレスプレフィックス表記

- $202.232.68.0 - 202.232.68.63 = 202.232.68.0/26$

Classlessな経路制御

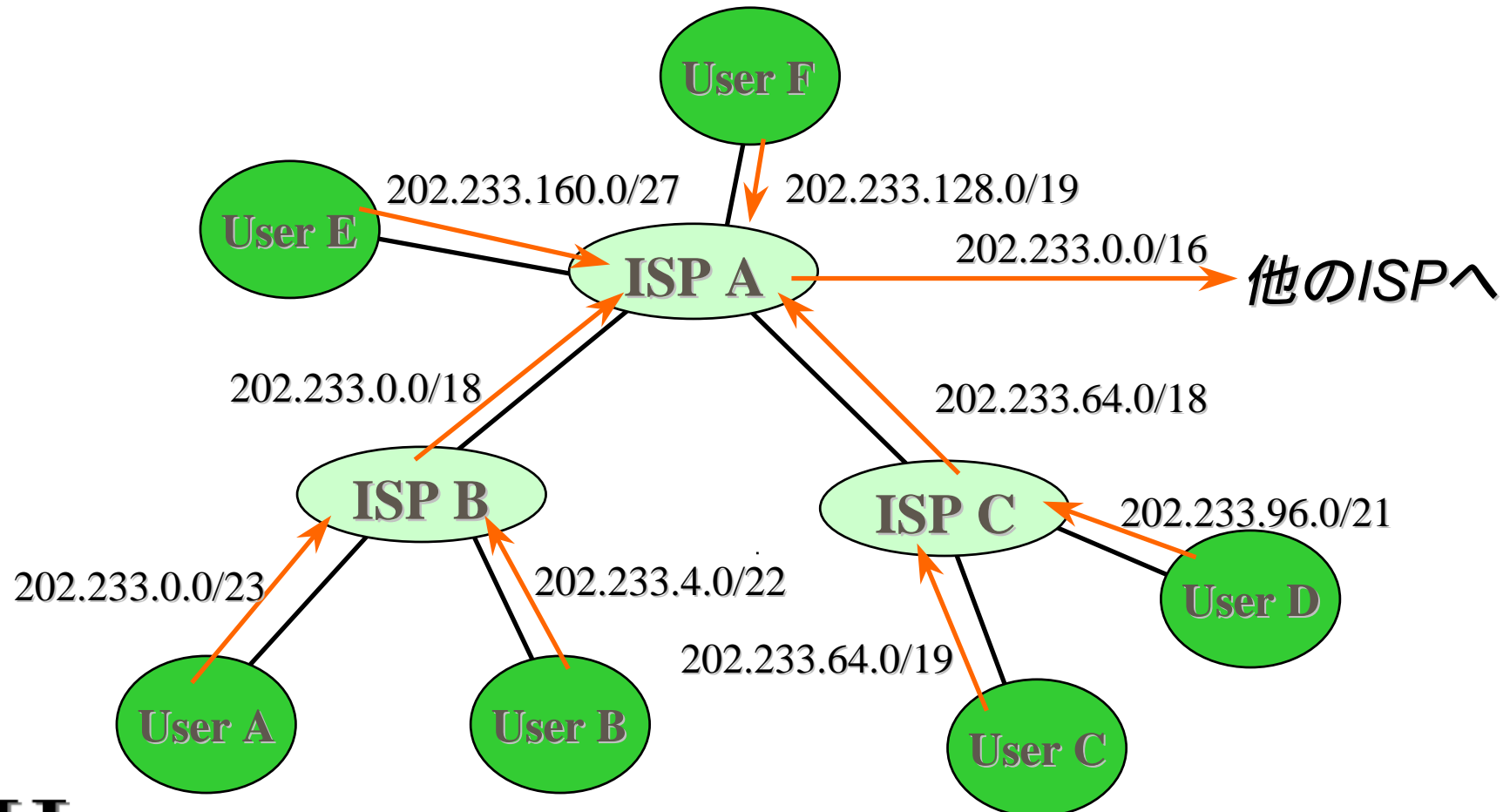
- VLSMのサポート
 - インターフェース / 経路表 / 経路制御プロトコル
- Supernetのサポート
 - アドレス / 経路情報の集成
- “*Classfull*”なアドレス割当と経路制御の概念の排除
 - all-0サブネット, all-1サブネット等
- Classlessな経路情報
 - ネットマスク長の伝播

階層的なアドレス割当

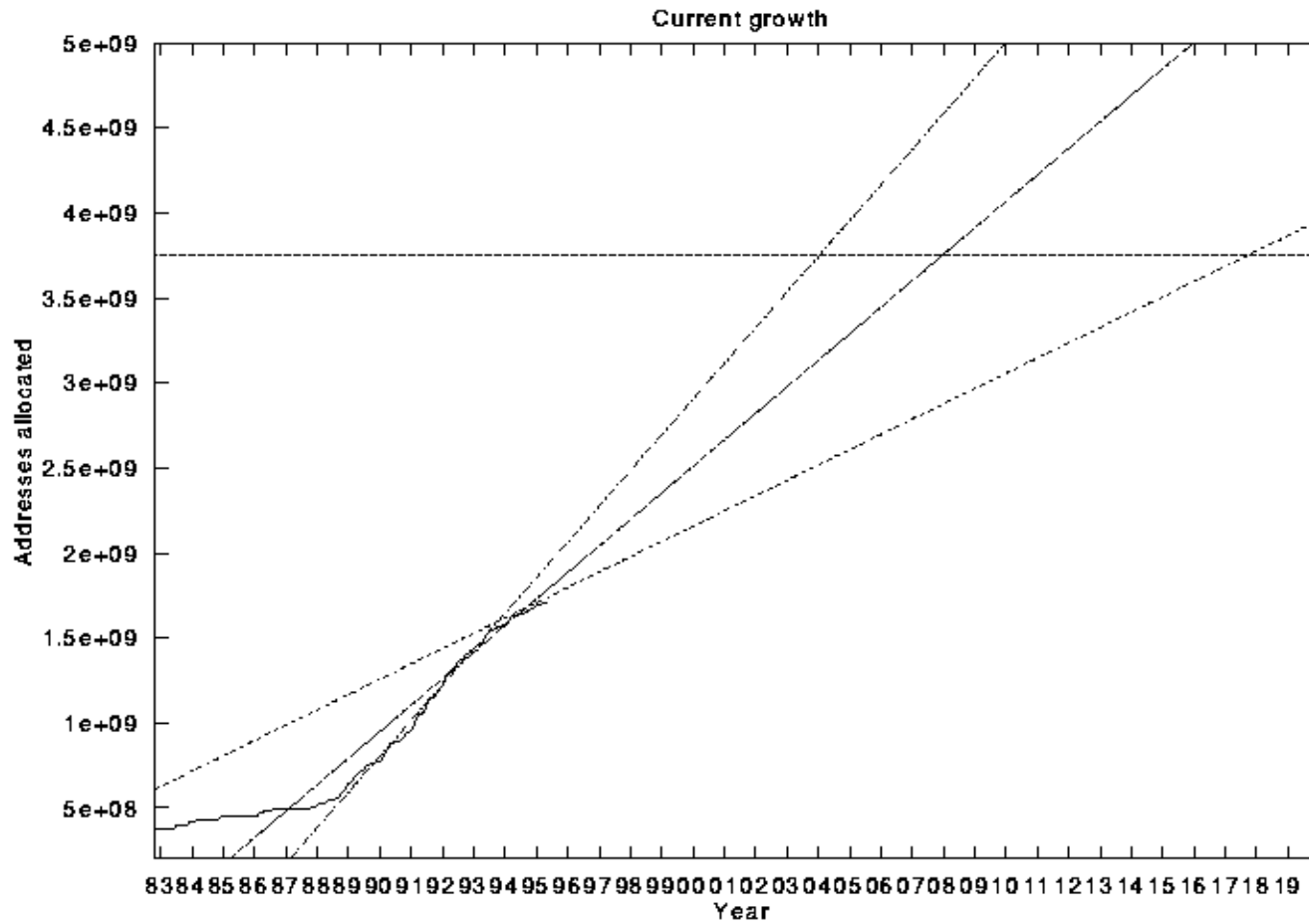


経路情報の集成

■ ネットワークトポロジに応じた階層的な集成

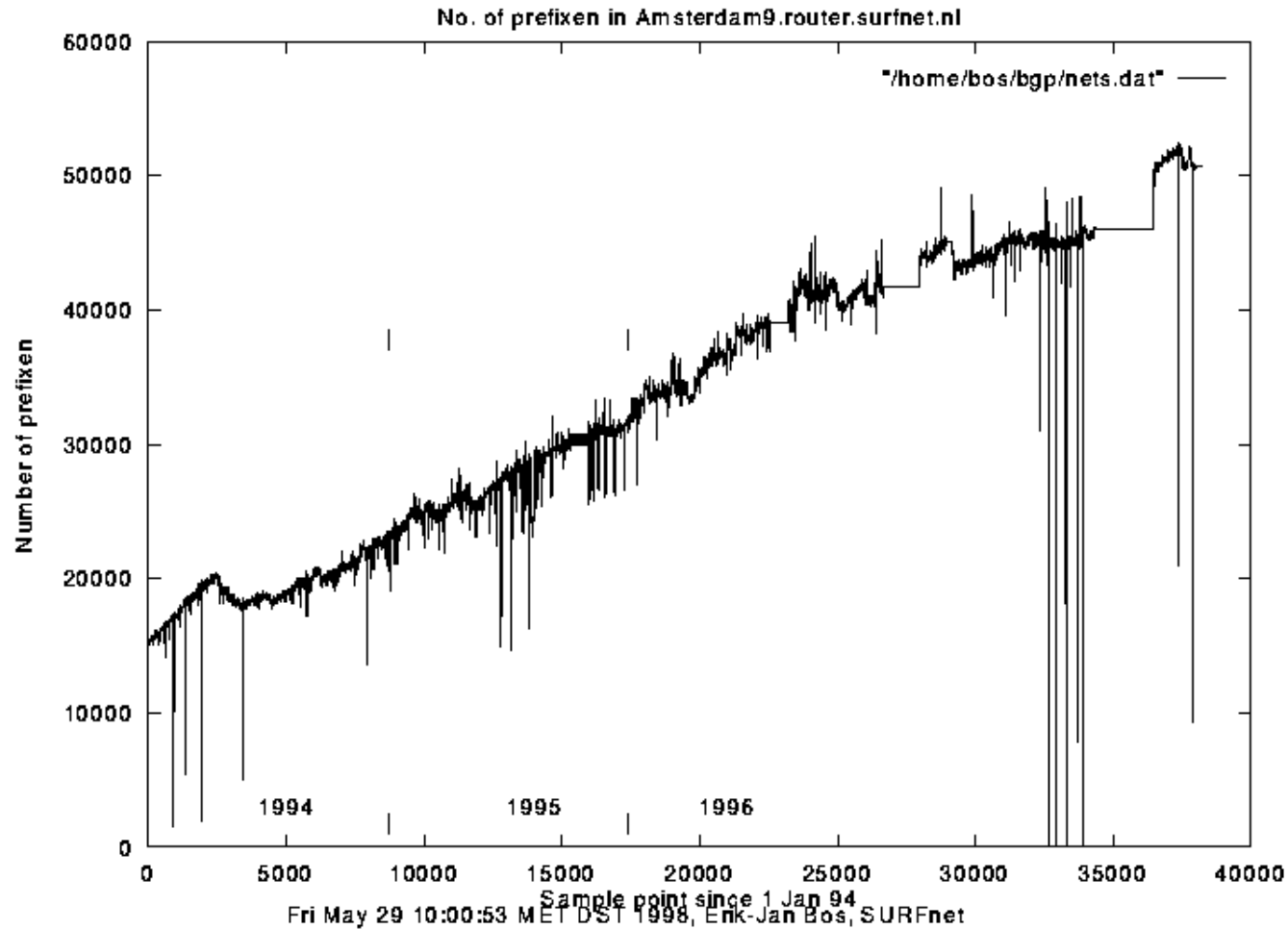


アドレス利用状況



経路表の増大状況

<SURFNET, Eric-Jan Boss氏作成>



BGP4の概要



BGP4(Border Gateway Protocol)

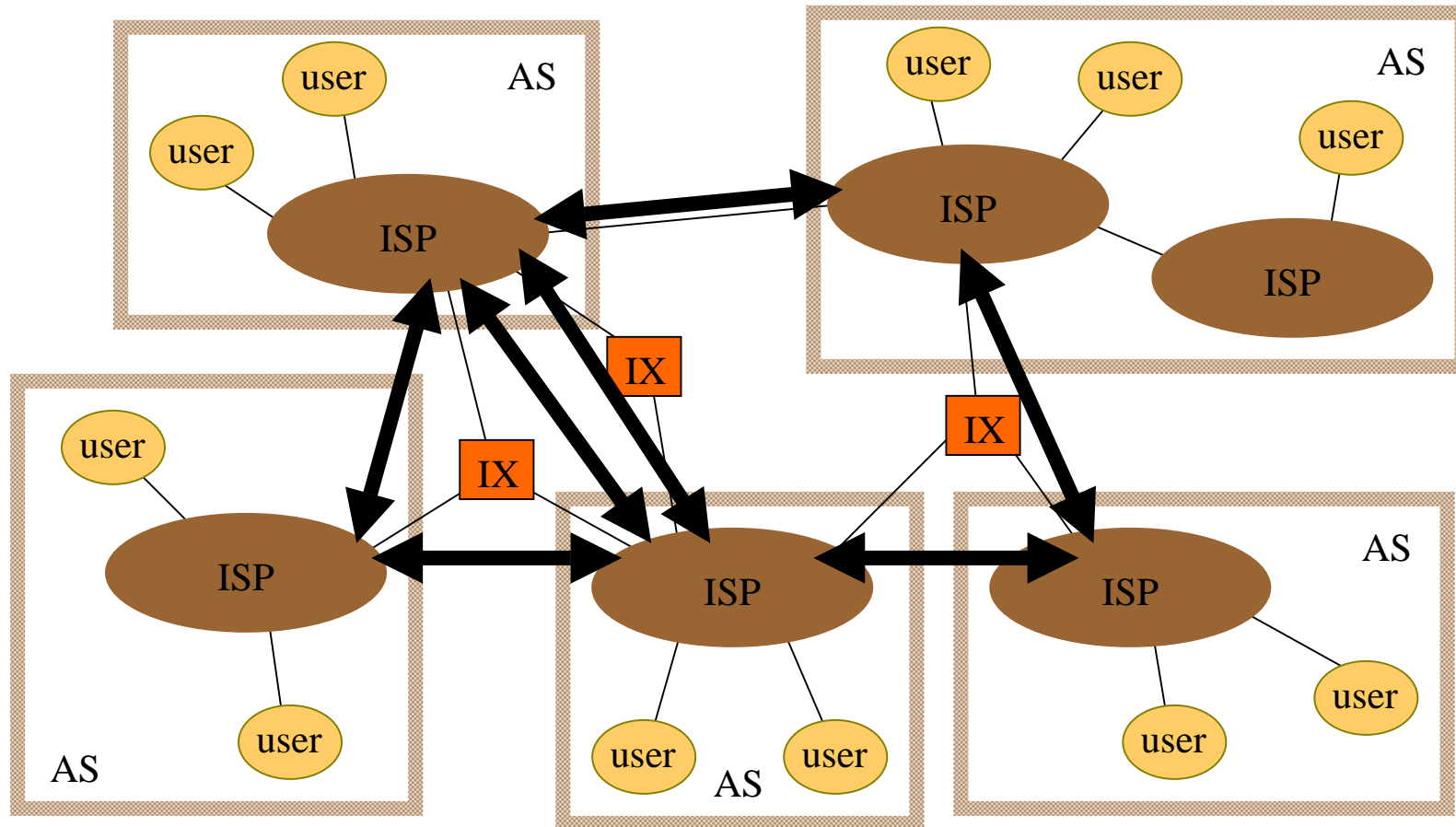
- RFC1771
- AS間経路制御のde-facto標準プロトコル
 - Autonomous System (AS)
 - 単一の管理主体により、単一の経路制御ポリシーにのもとの管理・運用される範囲
 - ISP AS
 - 現在のインターネットはASの集合体とみなすことが可能
- CIDRのサポート
 - CIDRの実現に不可欠

- TCP (ポート179) を用いる
 - コネクションを張ったルータ間(peer)で1対1の経路情報の交換
 - 経路情報の交換に信頼性を保証
 - RIP等と異なり、Incrementalな情報交換
- 16ビットのAS番号 (例: IIJはAS2497)
- Path Vector方式の経路制御プロトコル
 - 経路情報に付加されたパス属性 (Path Attribute) に基づく経路選択
 - AS Path, Origin, Next Hop, Multi-Exit-Discriminator(MED), Local Preference, etc.

EBGPとIBGP

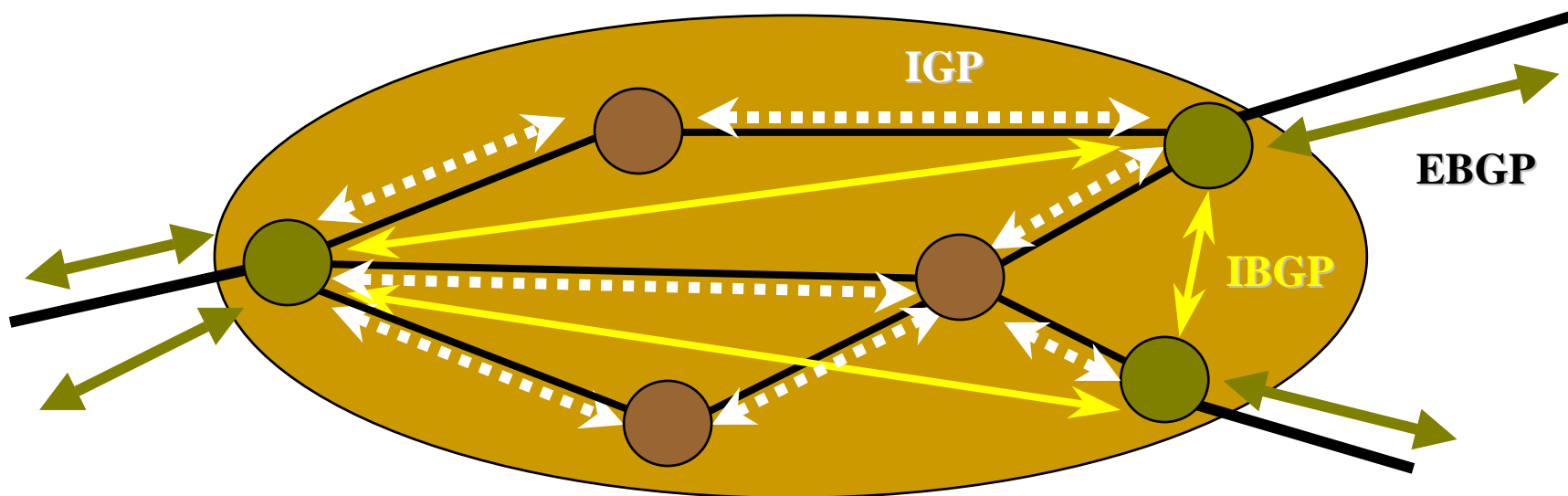
- BGPスピーカ (ボーダールータ)
 - BGPを用いて経路交換をするルータ等
- EBGP (External BGP)
 - 異なるASに属するBGPスピーカ間のBGPセッション
- IBGP (Internal BGP)
 - 同一AS内部のBGPスピーカ間のBGPセッション
 - full mesh
 - BGPスピーカ間で学んだ経路情報を交換する
 - 他のIBGPスピーカから学んだ経路は伝播しない

BGPを用いたAS間経路制御



AS外部とAS内部の階層的経路制御

- AS間
 - EBGPで他のASのボーダールータと経路情報を交換
- AS内
 - IBGPでEBGPでAS外部から学んだ経路情報を伝播
 - IGPでNLRIを伝播
- BGPとIGPの同期が重要



パス属性

- 伝播された各経路の属性を示す
 - 複数経路からの経路選択に用いる
 - ポリシーを表す
- 通過型 (Transitive)属性と非通過型 (Non-Transitive)属性
- 必須 (Mandatory)属性と任意 (Optional)属性

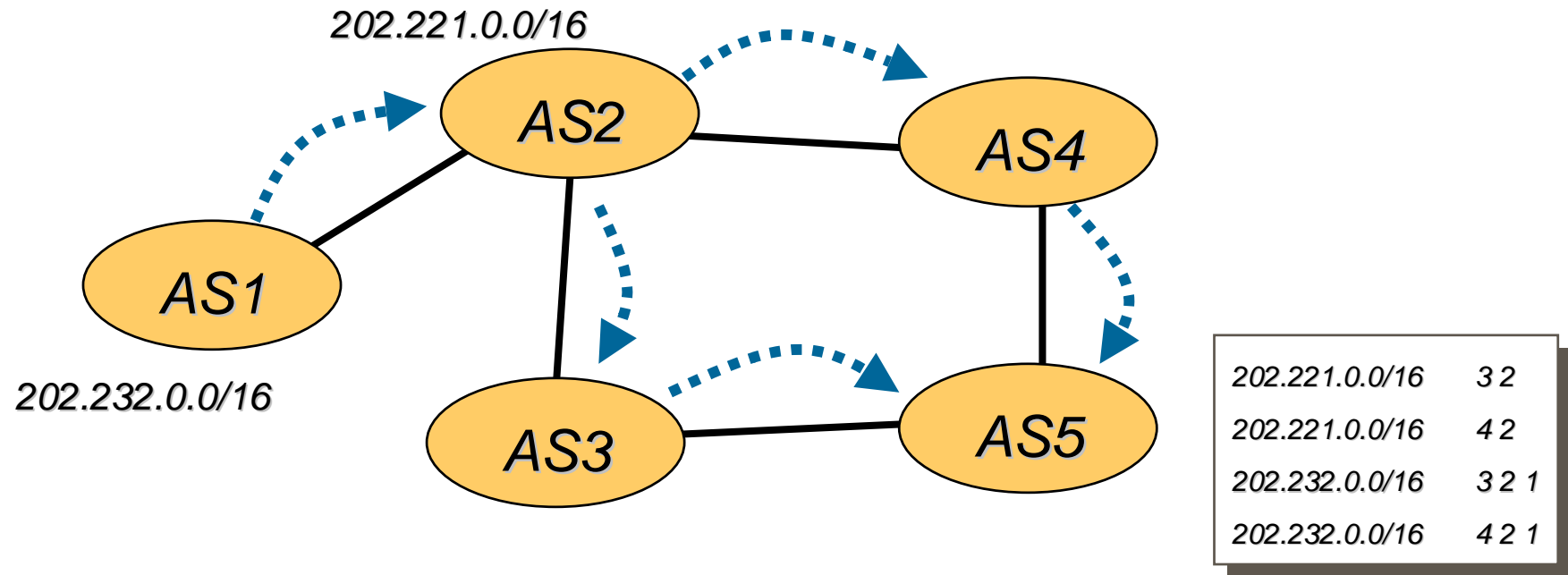
Origin属性

- その経路情報をどこから持ってきたかを表す
- 最初にBGPでアナウンスする時に設定される
- 可能な値
 - IGP
 - EGP
 - Incomplete
- 必須属性

AS Path属性

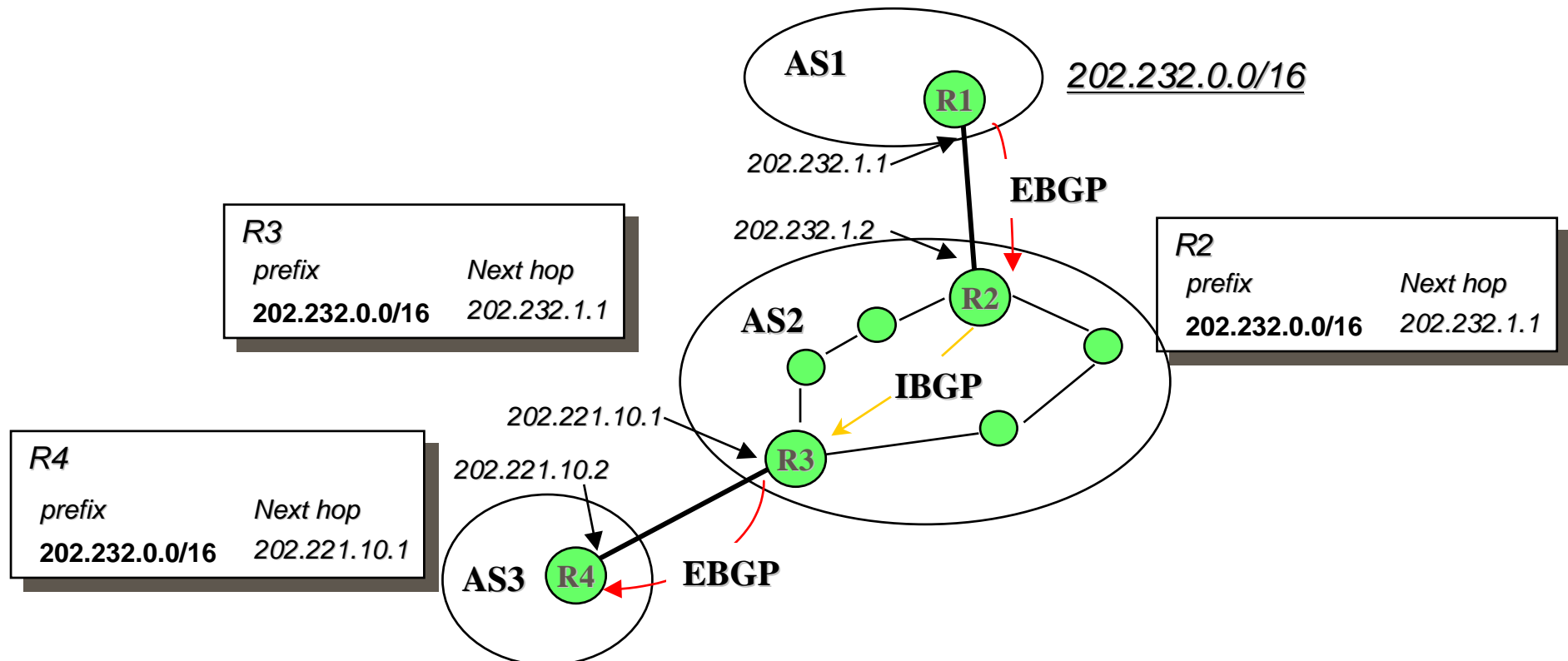
- 経路情報が伝播する際に経由したASの列/組
- ループの検出
- 一般的にはAS Pathの長さが短いほうが選ばれる
 - ポリシーによる
 - prepend, stuffing等の技巧
- 必須属性

AS Path属性の例



- AS1が202.232.0.0/16を、AS2が202.221.0.0/16をアナウンス

Next Hop属性



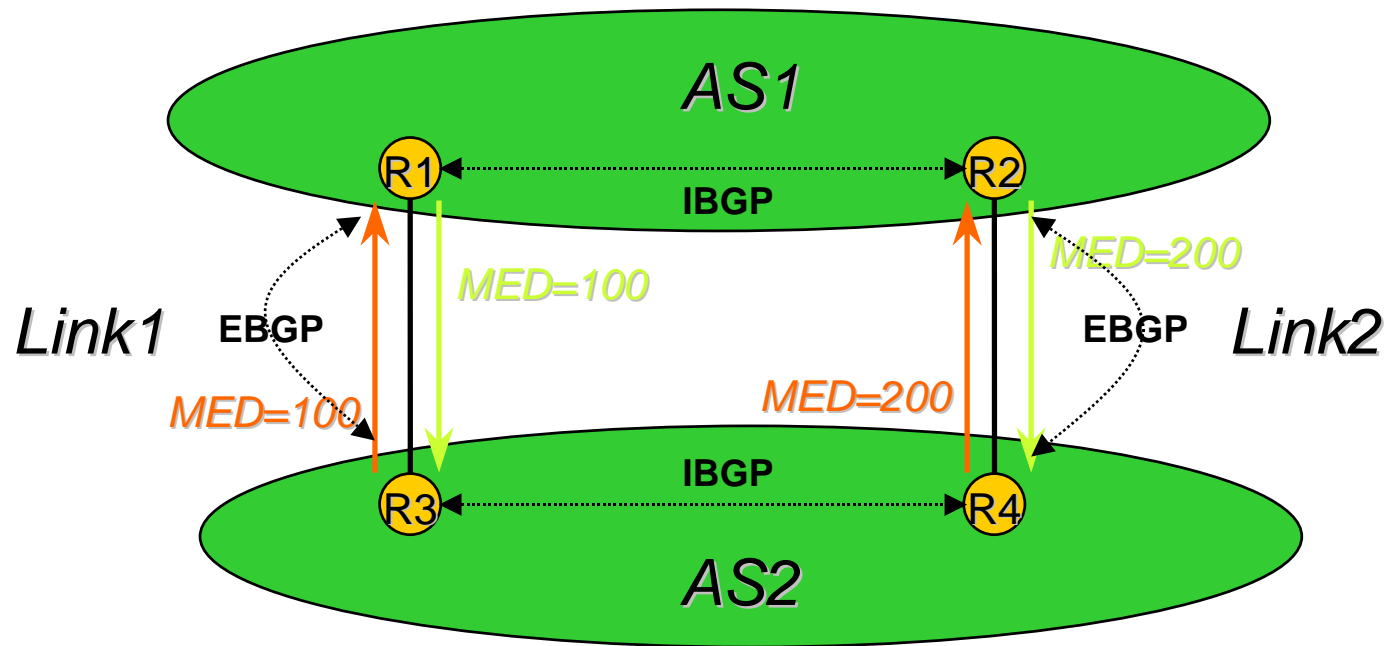
- 経路上の次のASのボーダールータのIPアドレス
- IBGPで伝播するときには値は変わらない
- R3からR1への経路はIGPで解決

Multi-Exit Discriminator(MED)

- 同一隣接ASからの複数経路を区別する
- 値が小さいほうを優先
 - IGPのコストを反映させるも可
 - ロードバランスを考えて設定するも可
- 非通過型属性

MEDの例

- AS1とAS2の間では、Link1を主に用い、Link2をバックアップとする場合

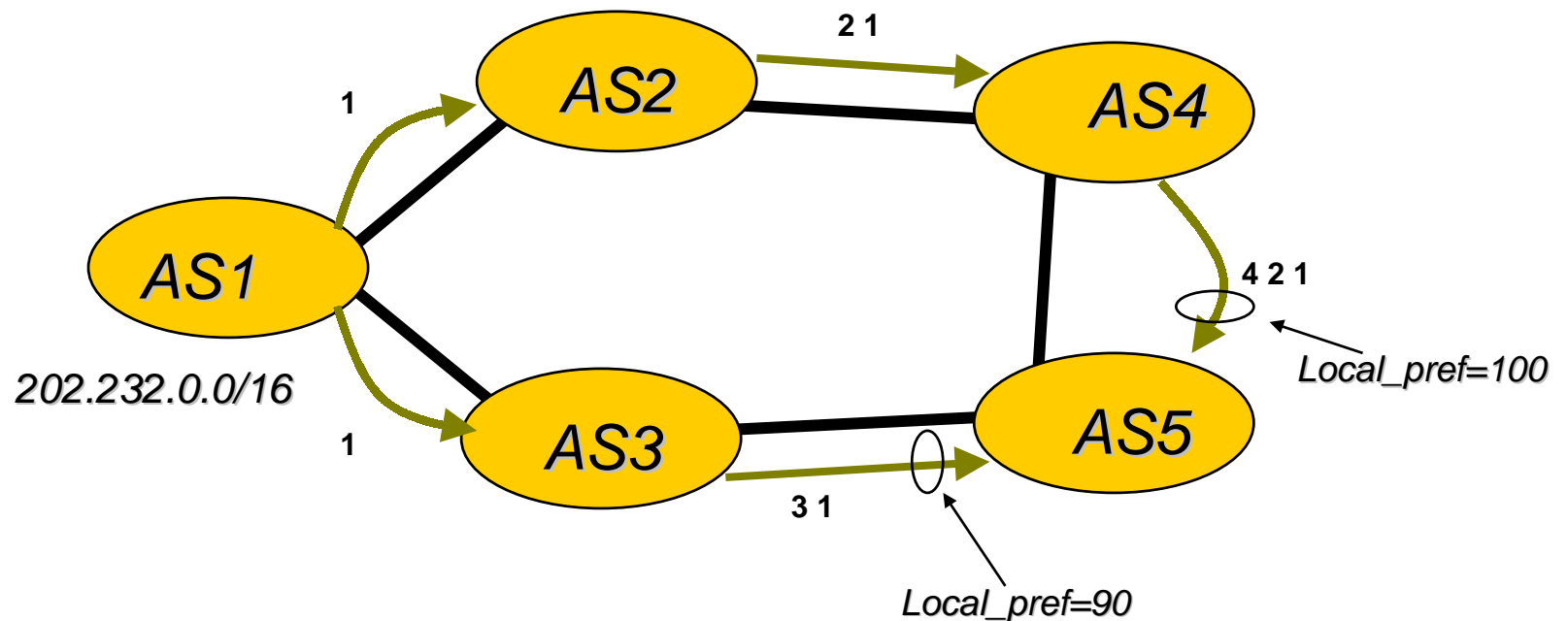


Local Preference属性

- 同一AS内部で複数経路の優先度を表すために用いられる
- 値が大きいほど優先される
- 非通過型属性

Local Preferenceの例

- AS5では、AS1へはAS4経由の経路を優先したい
- AS Path長では、AS3経由のほうが選択されてしまう
- AS5でAS4から受け取る経路に高いLocal_Prefの値を設定



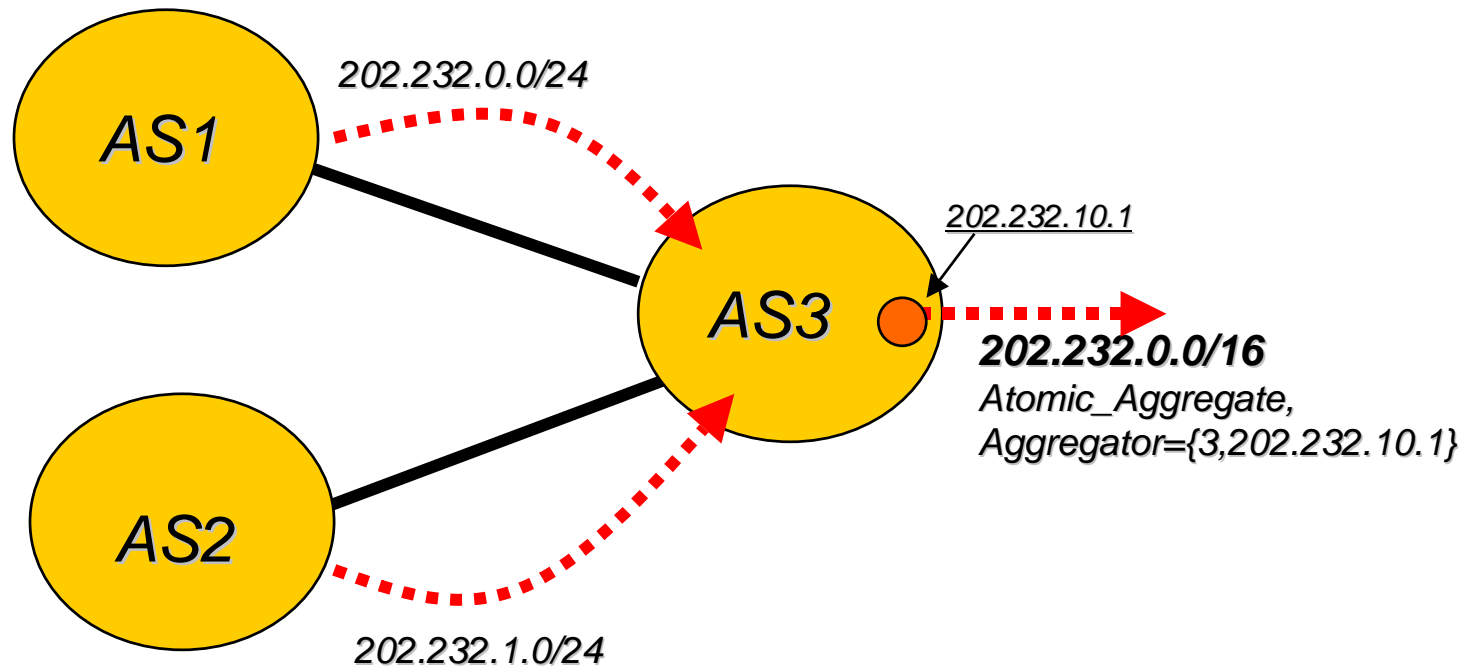
Atomic Aggregate属性

- 経路の集成 (Aggregate)を行ったときに付加される属性
- 集成の際に細かい経路に付加されていた情報が欠落したことを示す
- 再び細かい経路に分けることはできない

Aggregator属性

- 経路の集成を行ったBGPスピーカのIPアドレスと、それが属するAS番号を示す属性

経路の集成



- Atomic Aggregate属性とAggregator属性が設定される

Community属性

- RFC1997
- 経路に色をつける
 - ポリシーに応じて経路をグループ分けする
 - 一つの経路が複数のグループに属することも可
- 32ビットの整数値

Community属性の値(共通)

■ 予約領域

- 0x00000000 - 0x0000FFFF

- 0xFFFF0000 - 0xFFFFFFFF

■ Well-Known Community

- NO_EXPORT(0xFFFFFFFF01)

- ┆ AS外部に出さない

- NO_ADVERTISE(0xFFFFFFFF02)

- ┆ 他のルータに出さない

- NO_EXPORT_SUBCONFED(0xFFFFFFFF03)

- ┆ 同盟中の他のメンバーASに出さない

Communityの値(ユーザ定義)

- 予約されていない値は、AS毎に独自のCommunityを定義できる
 - 上位16ビット: Communityを定義したAS番号
 - 下位16ビット: そのAS内部で用いるCommunity番号
 - 表記法: AS番号:Community番号

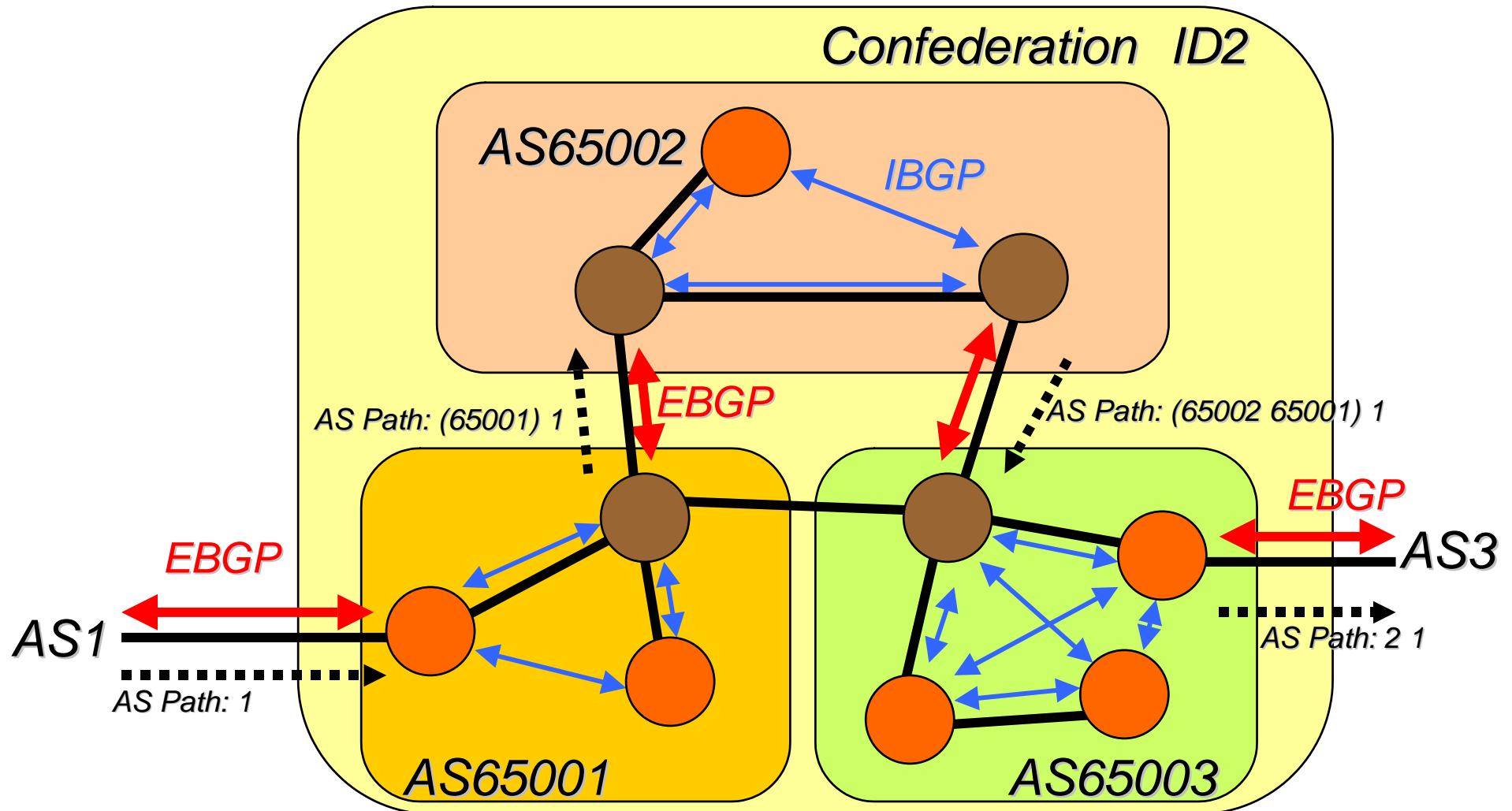
Community属性の利用例

- AS内部での経路のグループ分け
 - 外部への経路アナウンスのポリシーに応じてIngress FilterにてCommunityを定義する
 - 2497:10 顧客の経路
 - 2497:20 peerの経路
 - 個別の経路情報ではなくCommunityの値のみに着目してEgress Filterを設定できる
 - 例えば、2497:10をCommunity値にもつ経路のみupstreamにアナウンスする等
- 他のASに対するポリシーの伝達

AS同盟(Confederation)

- RFC1965
- AS内部を、サブASに分割
 - サブAS間の階層関係、包含関係は無い
 - サブASではAS番号にプライベートAS(64512-65535)を用いる
 - 各サブASでは独立したIGPの利用が可能
- 外部からは一つのASに見える
- サブAS間は、IBGPに近いEBGP
 - サブAS間で経路を渡すときには、Next Hop, MED, Local Preference等の値は保存される
- 大きなASで、IBGPのメッシュを減らすのに役立つ

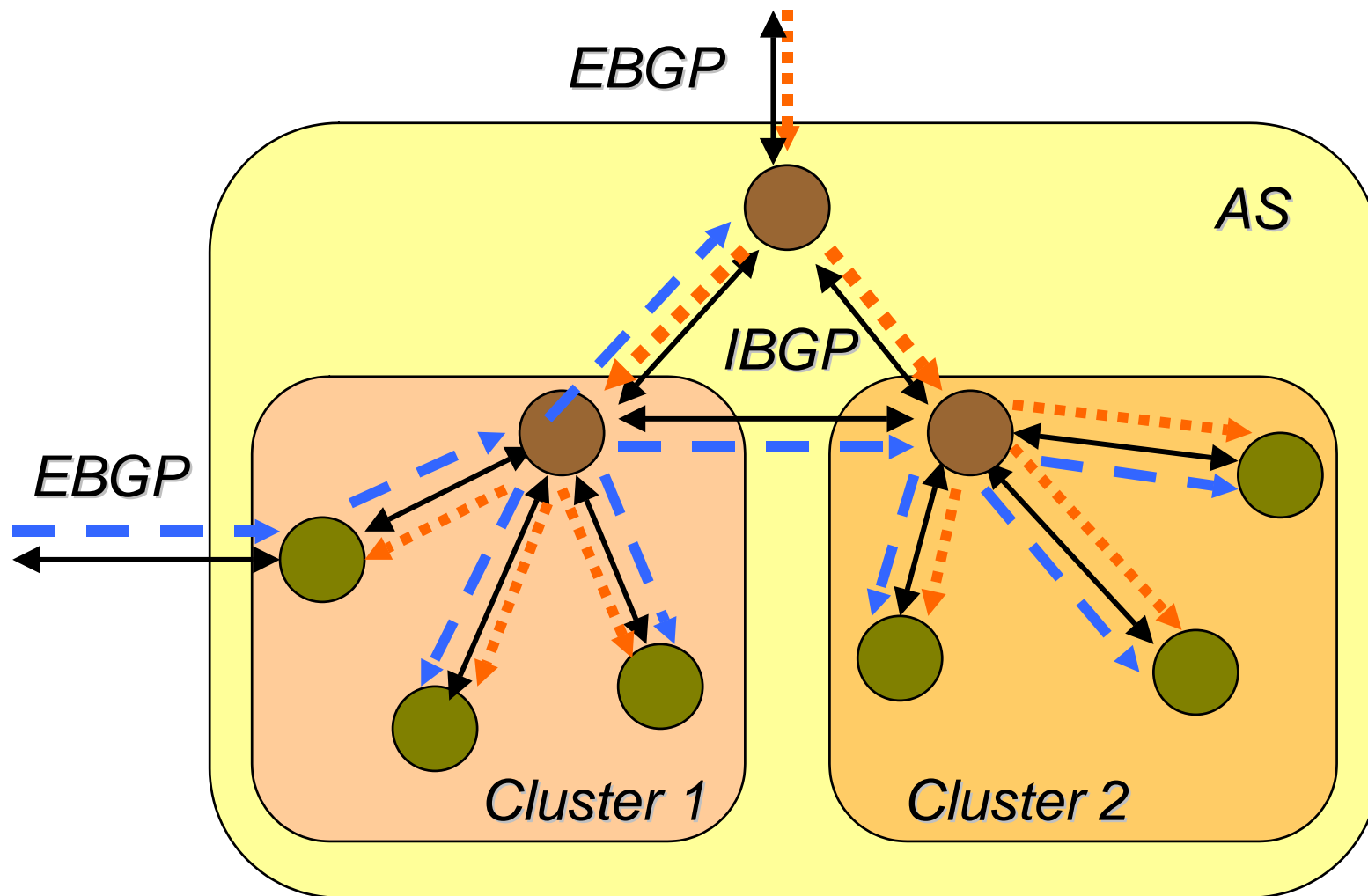
AS Confederationの例




Route Reflector

- RFC1966
- AS内部で用いるルートサーバ的イメージ
- BGPスピーカをグループ(クラスタ)に分ける
 - リフレクタ
 - ┆ AS内の他のクラスタのリフレクタと経路情報を交換
 - ┆ クラスタ内のBGPスピーカに経路情報を供給
 - クライアント
 - ┆ リフレクタからBGPの経路情報をもらう

Route Reflectorの例



ポリシールーティング



考え方とBGPを用いた例

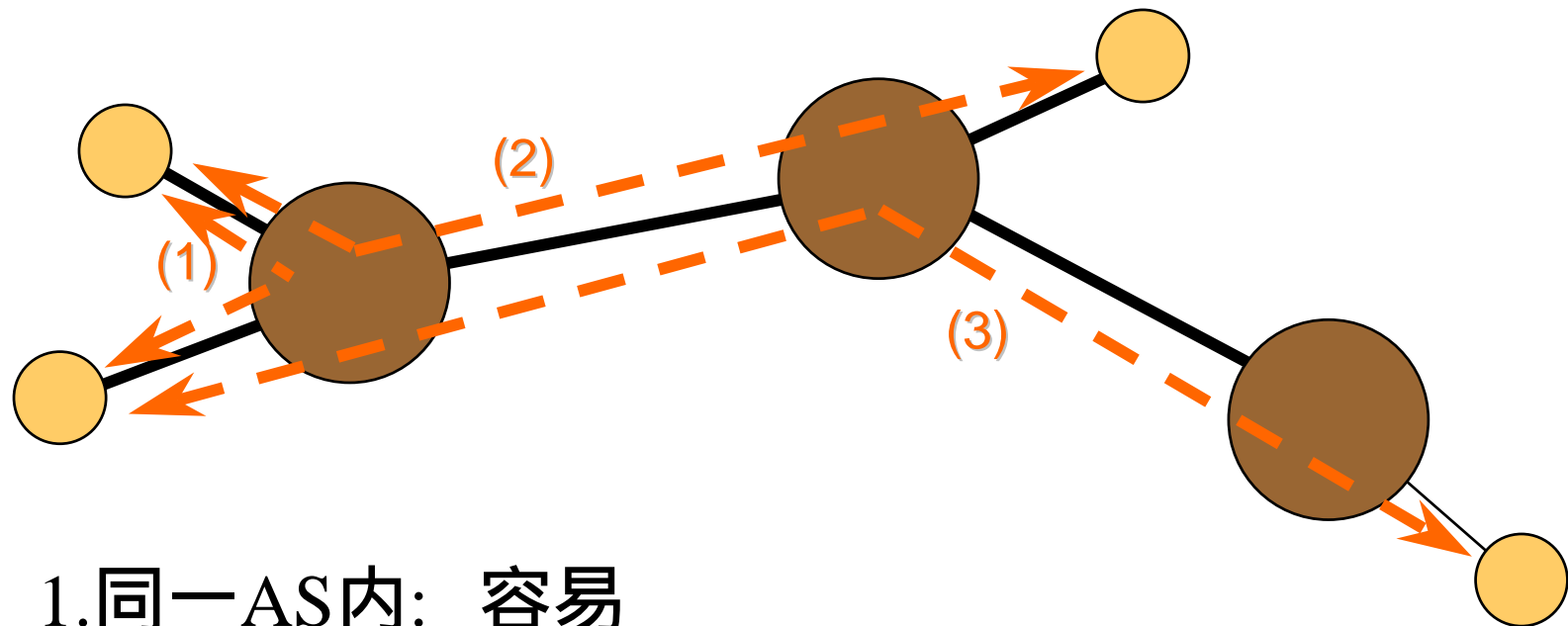
ポリシールーティングとは？

- **ポリシーに基づく経路選択**
 - 他のISP(AS)とどのようにトラフィックをやりとりしたいか
 - 単に近さやコストをもとにした選択ではない
 - 他のISPとの間でどのように経路情報をやり取りするか
 - 個々の目的地ごとに経路を選択する
 - BGPのパス属性を用いる
 - 経路情報のやり取りの制御だけでは実現できないポリシーもある
 - ネットワークトポロジの再考などが必要

ポリシーの例

- 自ASを通過させてよいかどうか
 - 顧客ASは、すべて通過可
 - 非顧客AS(例えばIXでの無償Peer)は顧客には通過させるがupstreamには通過させない
- 隣接ASや経路上のASの使い分け
 - 特定の目的地に至る経路として不安定なASを経由したくない
 - マルチホーム環境下で、料金の高いupstreamはできるだけ使いたくない
- 自AS内のリンクの使い分け
 - 混んでいるリンクにはなるべくトラフィックを乗せたくない

ポリシールーティングの実装



- 1.同一AS内: 容易
- 2.隣接AS間: やや難
- 3.離れたAS間: 難

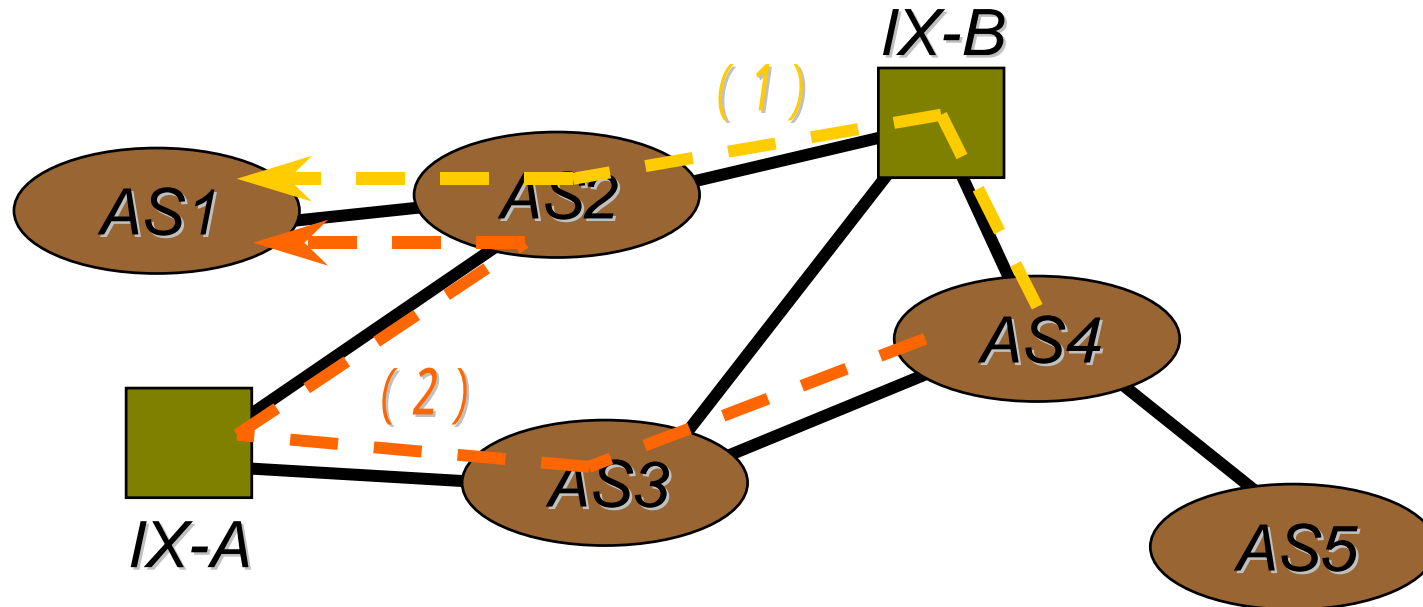
ポリシールーティングの実装(その2)

- 経路情報のフィルタリング
 - パス属性に基づいて特定の経路情報をピックアップ
 - Ingressフィルタ (受信時のフィルタ)
 - Egressフィルタ (送信時のフィルタ)
- BGPパス属性を操作
 - Local_pref, MED, AS Path Prepend等
 - Communityを用いた経路の分類
- 経路情報の操作によって、すべてのポリシーを実現できるわけではない

複数経路の選択

- あるASへ複数の経路がある場合、どの経路を優先するか？
 - AS_PATHの短い経路を優先
 - NEXT_HOPまでのIGP的な距離が短いパスを優先 (Hot Potato)
 - 特定の経路を優先
 - 顧客からの経路を優先
 - 特定IX経由の経路を優先

AS PATHによる経路選択



■ AS4での経路選択

- 経路(1)のAS Path: AS2 AS1

- 経路(2)のAS Path: AS3 AS2 AS1

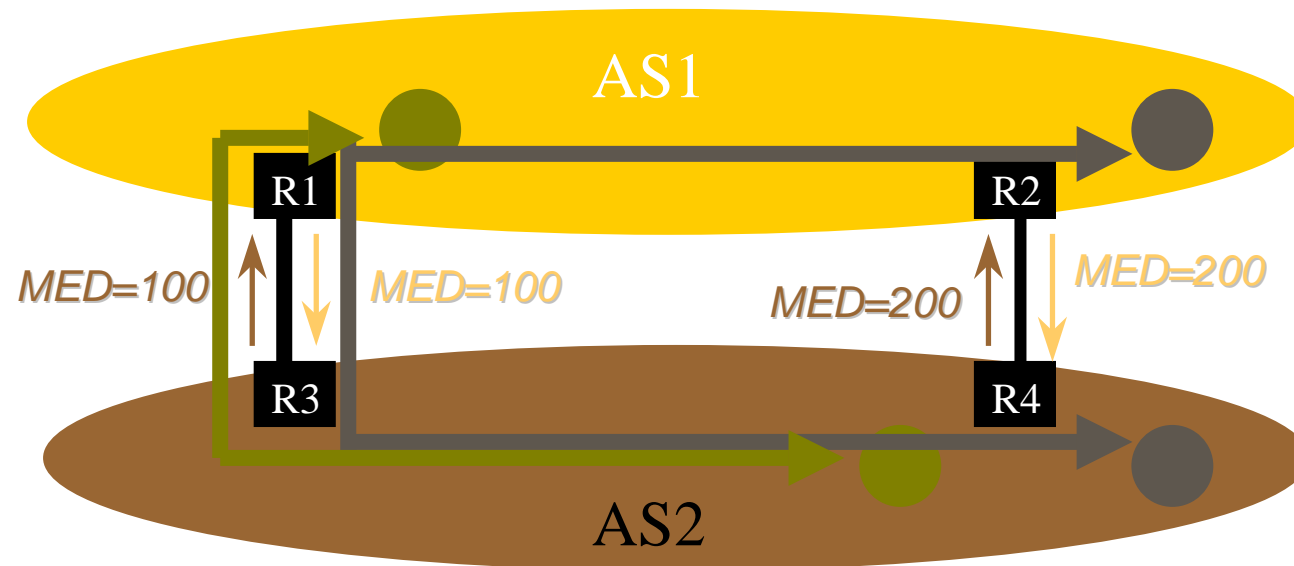
- 通常はAS Pathの短い経路(1)が選択される

- Ingress Filterで経路(2)のAS Pathに高いLocal_Prefの設定も可

■ AS5はAS4と異なるポリシーをもてない

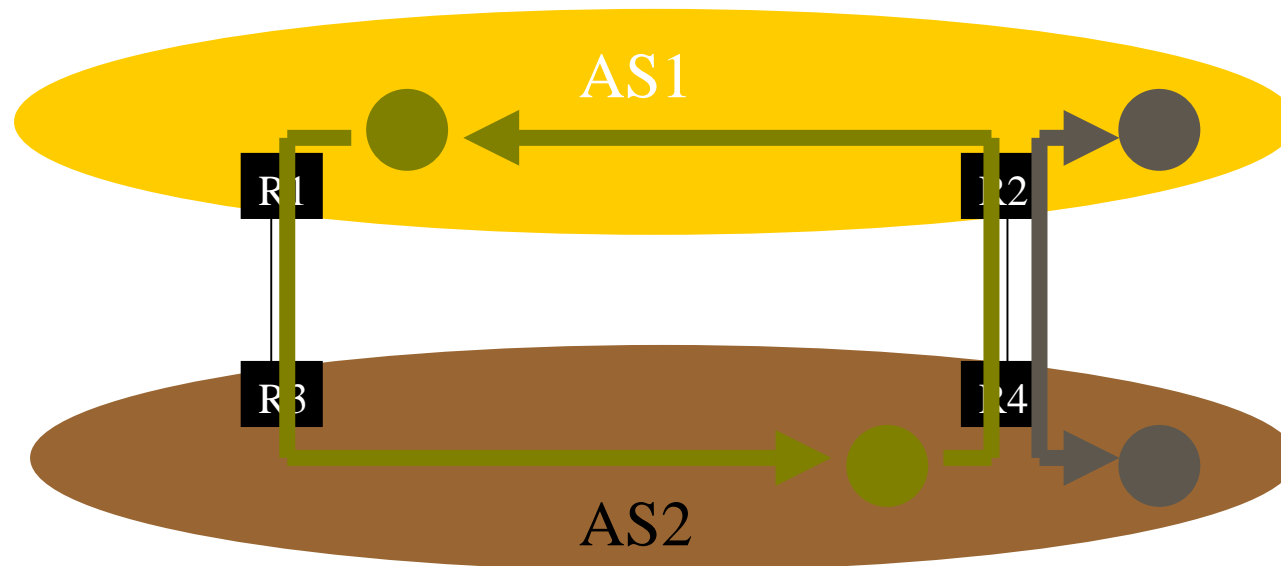
隣接AS間の複数パス

- 一方のパスを優先
- MEDを利用
- 非効率な経路になる場合もあり

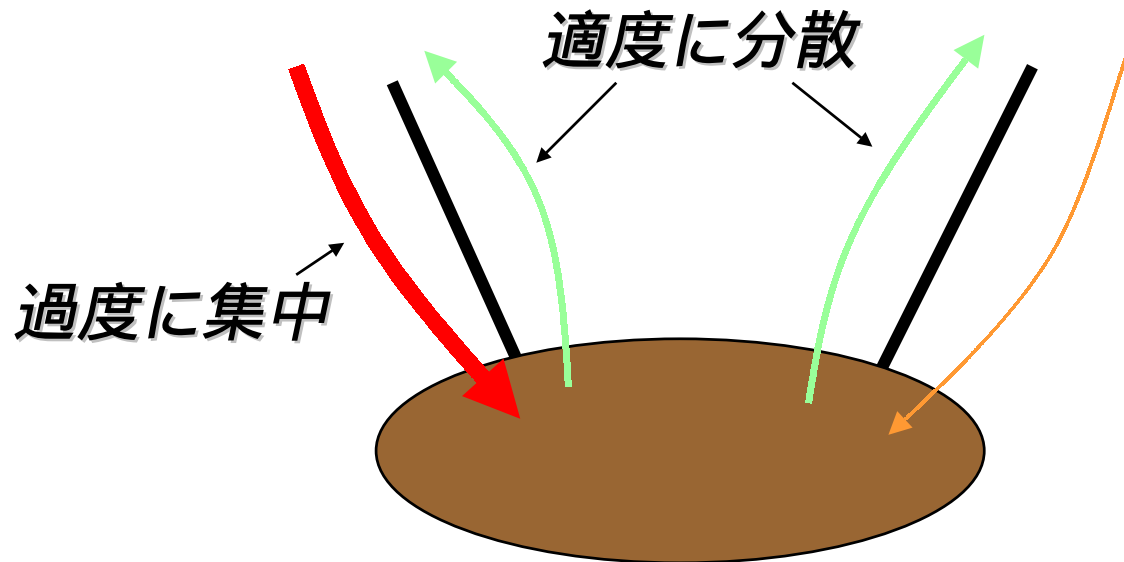


隣接AS間の複数パス(Hot Potato)

- 最も近い出口から次のASに渡してしまう
- MEDの値を等しくして、Next Hopの解決をIGPで行う
- 複数IXで相互接続しているISP間で一般的に行われる方法



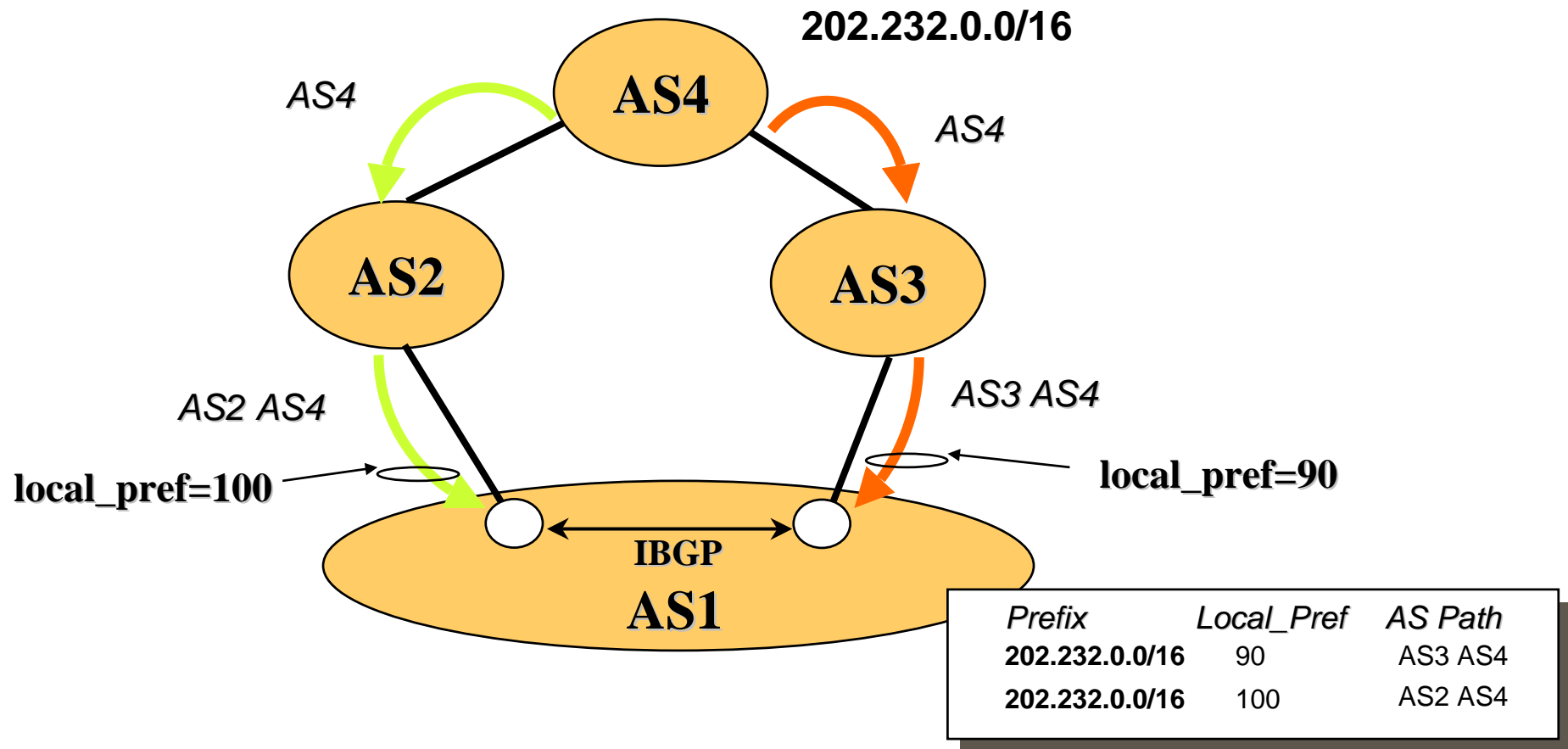
マルチホーム下でのロードバランス



- 他ASに出るトラフィックの調整は容易
 - 自ASで受け取る複数経路間の選択の問題
- 他ASから入ってくるトラフィックの調整は困難
 - 他ASでの複数経路間の選択を制御しなくてはならない
 - MED, AS PATH Prepend, Community等を駆使

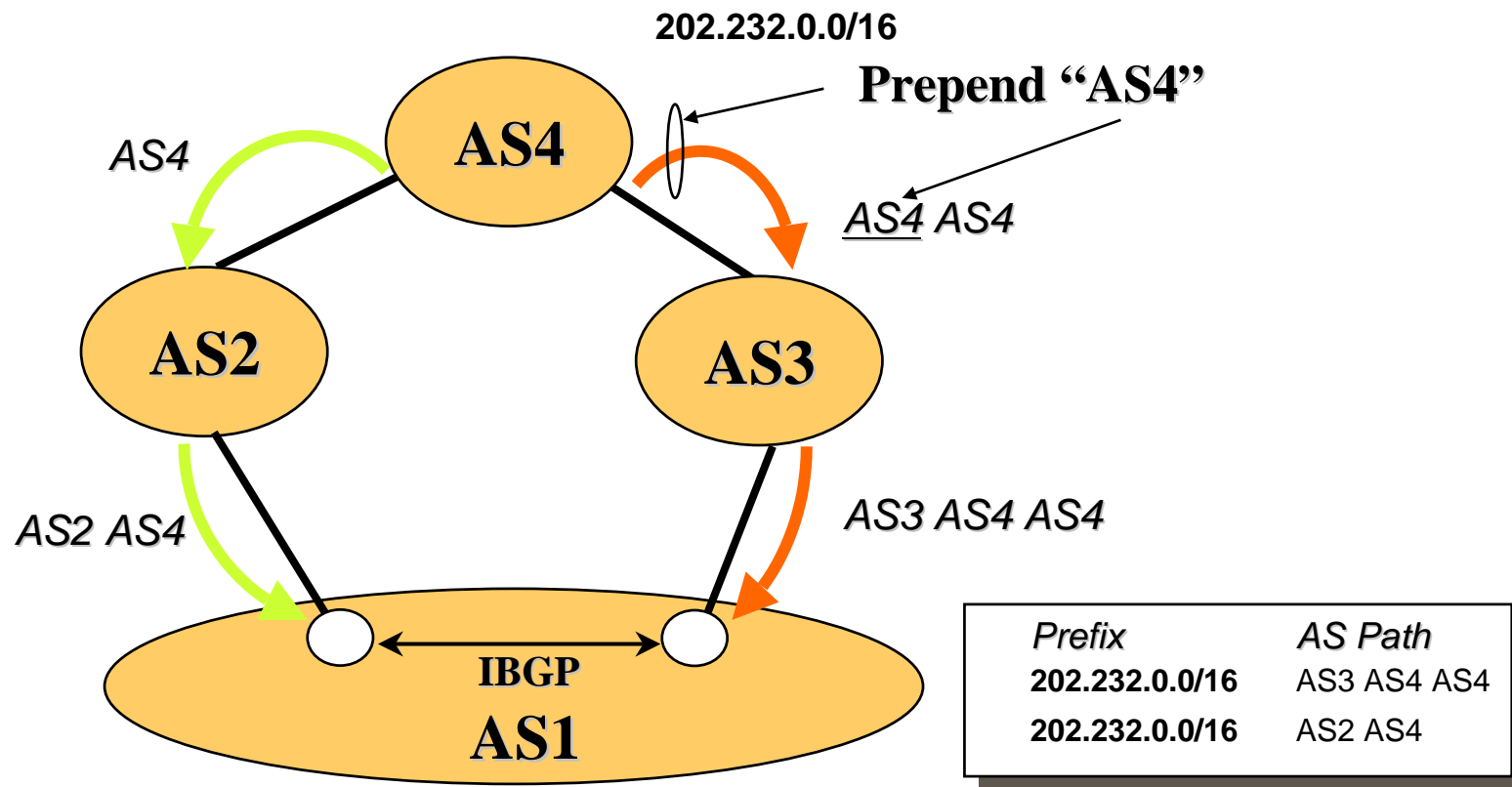
経路情報を受け取るASでの選択

- AS1が、AS4に向かうトラフィックをAS2経由で送りたい場合
- AS1のIngress Filterで特定の経路をピックアップしlocal_prefを設定



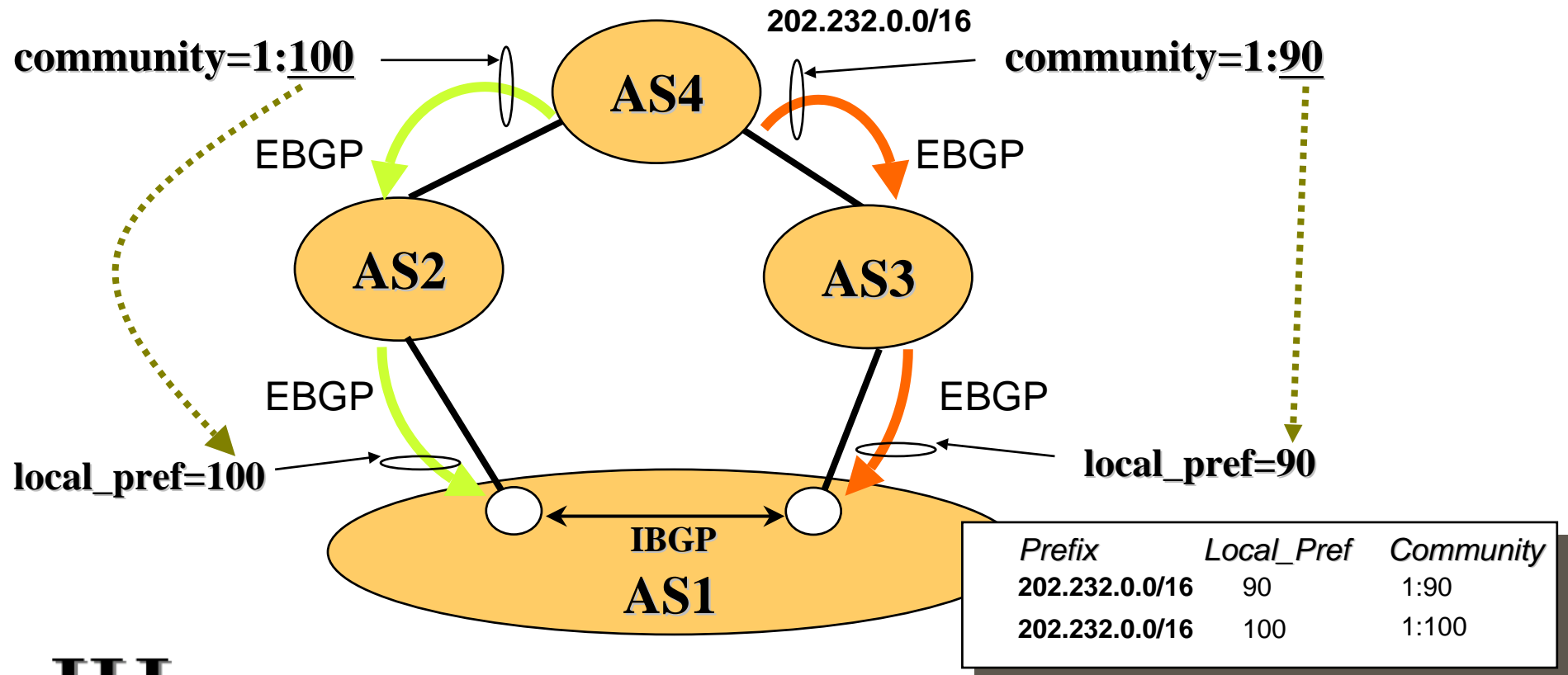
経路情報をアナウンスする側からの調整(1)

- AS4が、AS1からのトラフィックをAS2経由で受け取りたい場合
- AS4のegress filterで、自AS番号をPrepend



経路情報をアナウンスする側からの調整(2)

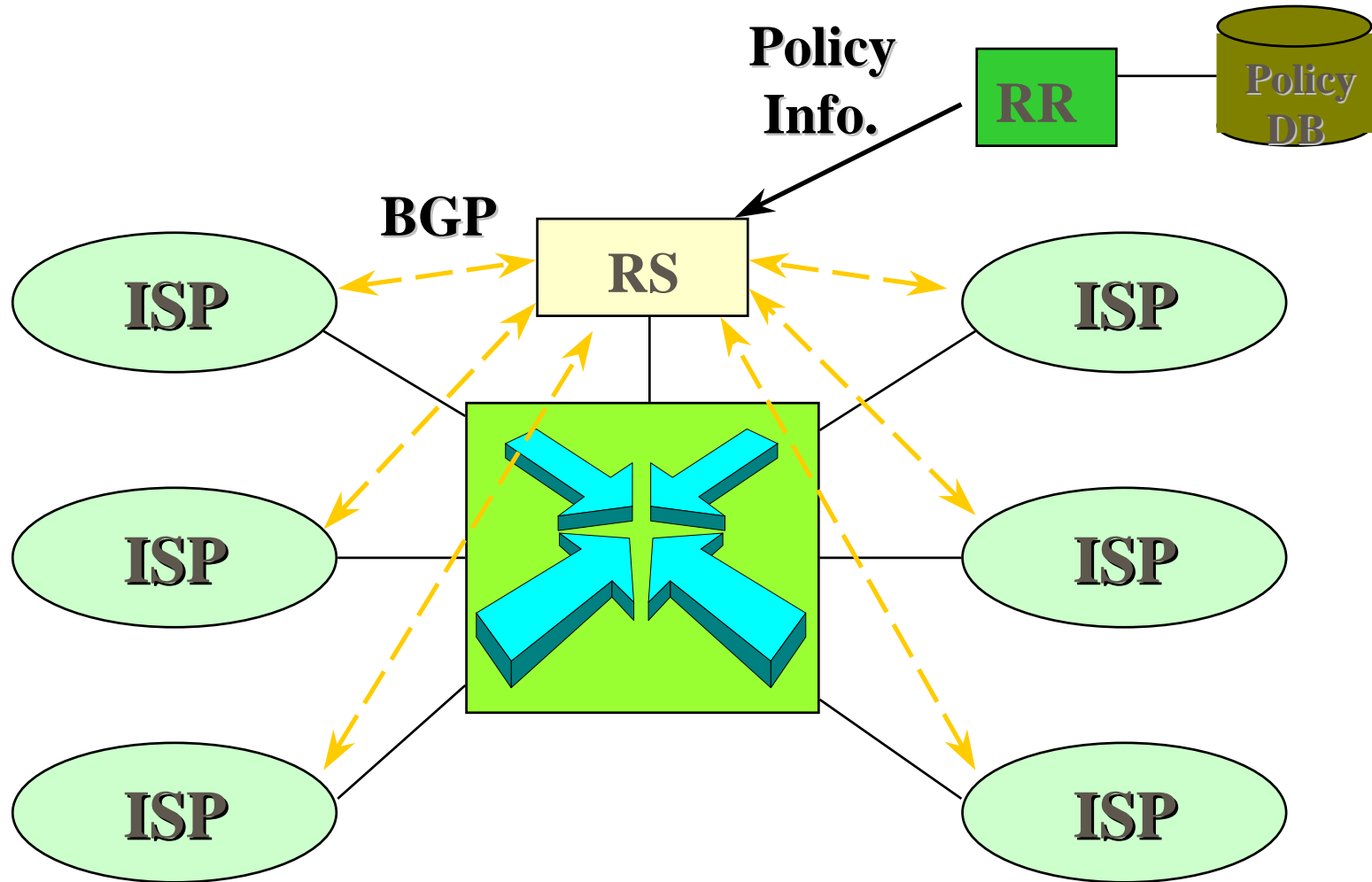
- AS4が、AS1からのトラフィックをAS2経由で受け取りたい場合
- Community属性の利用(RFC1998)
- 但し利用できるISPはごく少数
- AS4のegress filterでcommunityを設定
- AS1のingress filterで、自AS向けのcommunityに指定されたlocal_prefの値を設定



ルーティングレジストリとルートサーバ

- パケット転送と経路選択のプロセスの分離
- ルーティングレジストリ (RR)
 - 各ASの経路制御ポリシーのデータベース
- ルートサーバ(RS)
 - 第2層エクスチェンジに接続するISPとBGPで通信する
 - RRに登録されたポリシーをもとに、各ISPのボーダルータの経路表を計算する
- RRを用いた経路フィルタリング

ルーティングレジストリとルートサーバ



BGPの運用上の問題



IBGPフルメッシュ

■ BGPの仕様

- ボーダールータ間でのみIBGPのフルメッシュを張る
- 外部から学んだ経路はIGPでAS内部のルータに伝播する
- ボーダールータでIBGPとIGPの間で同期を取る
フルルートはIGP(例えばOSPF)で流すのは非現実的
BGPの経路情報の不安定化につながる flapping

■ IBGP “HACK”

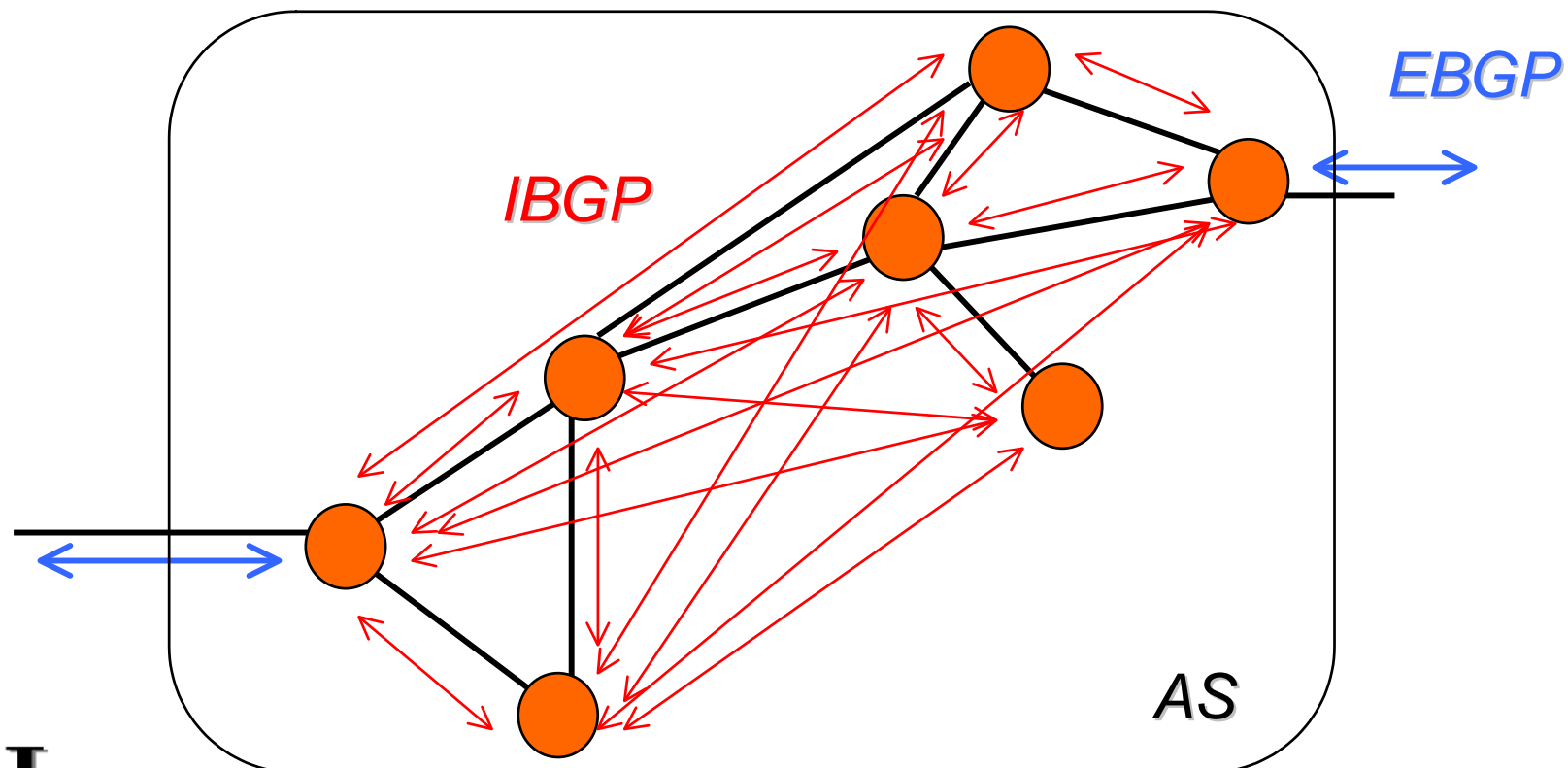
- フルルートを持つ必要があるすべてのルータ間でIBGPのフルメッシュを張る
- IBGPとIGPとの間の同期は取らない
- Next Hopの解決はIGPで行う

IBGPフルメッシュとの戦い

■ IBGPフルメッシュの問題

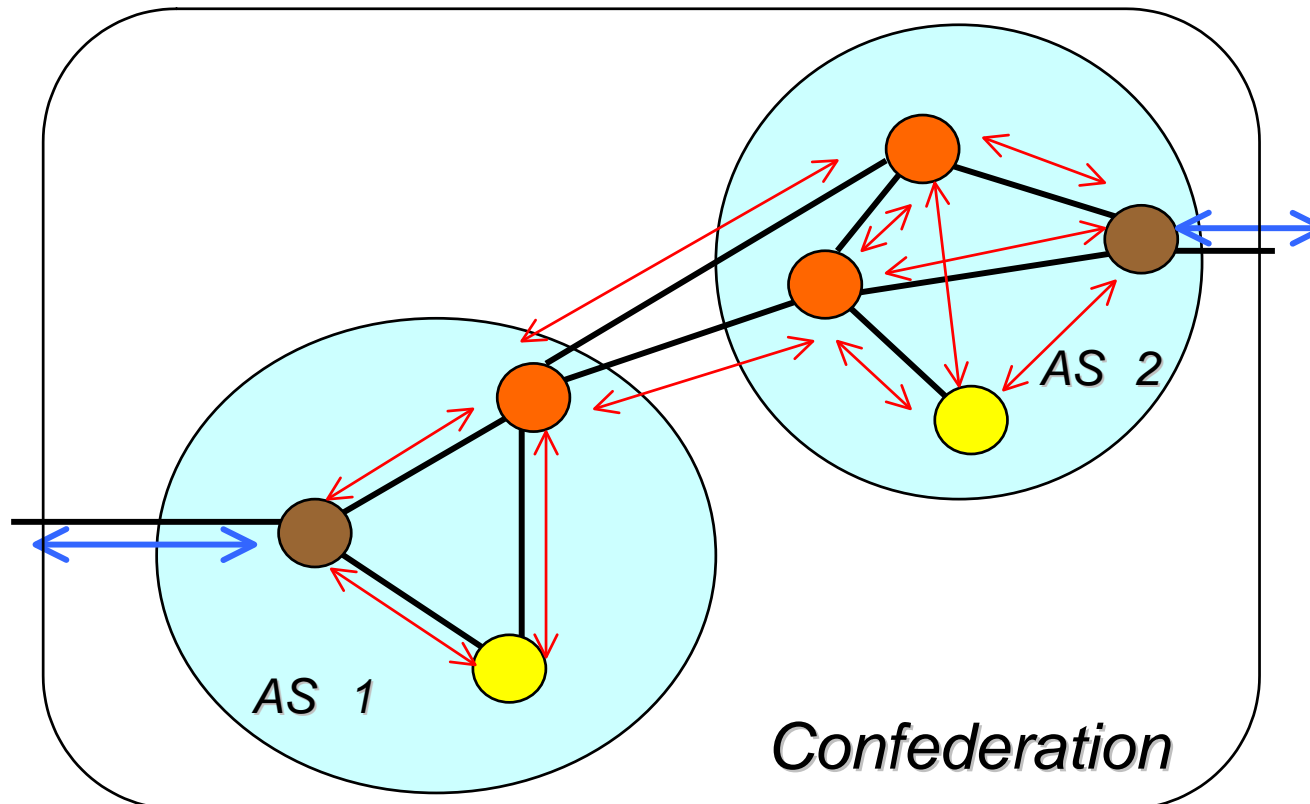
- N^2 のオーダで増えるIBGPセッションの数をいかに減らすか
 - 各ルータで保持するpeerの数と処理するupdateの数
 - 経路収束に要する時間

■ 例 7台のBGPスピーカ、21のIBGPセッション



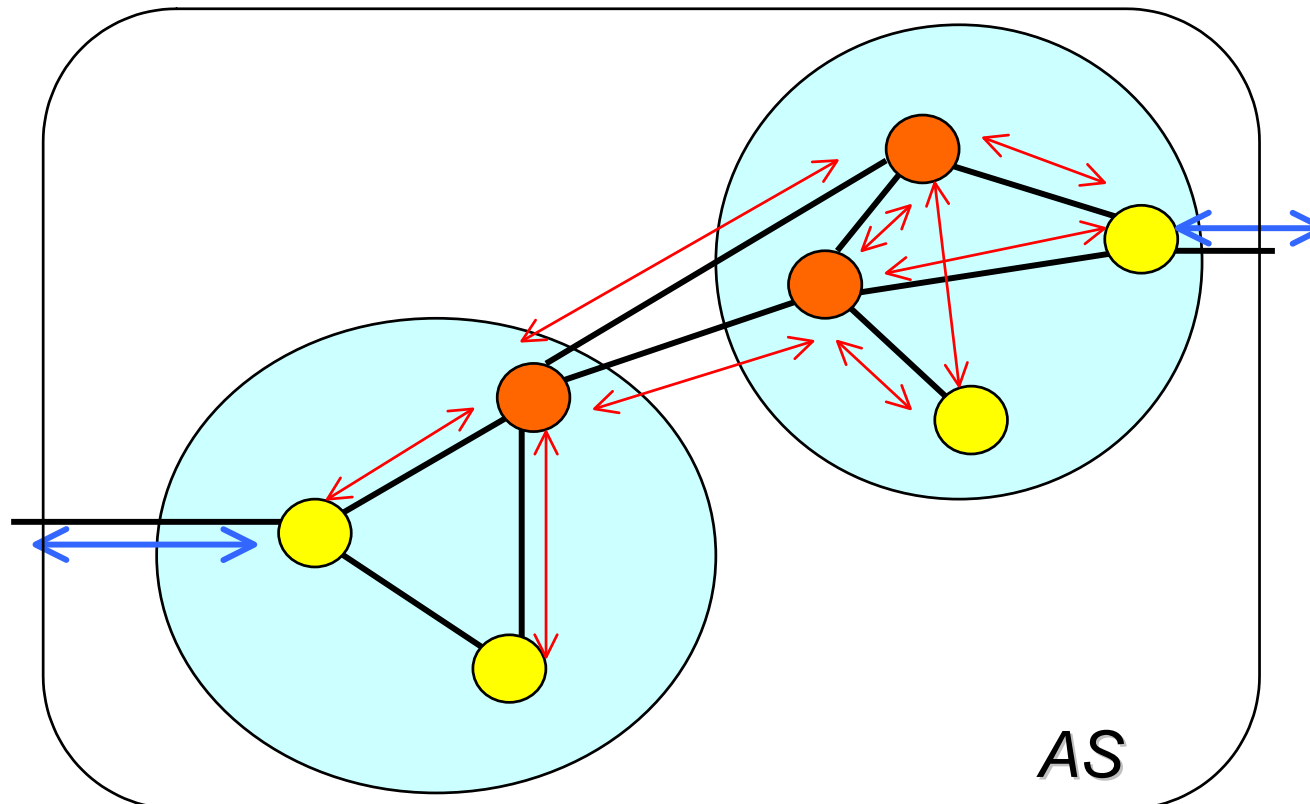
Confederationの利用例

- 2つのサブAS
- 2つのEBGPセッション
- 9のIBGPセッション



Route Reflectorの利用例

- 2つのクラスタ
- 3台のリフレクタ、4台のクライアント
- 9のIBGPセッション

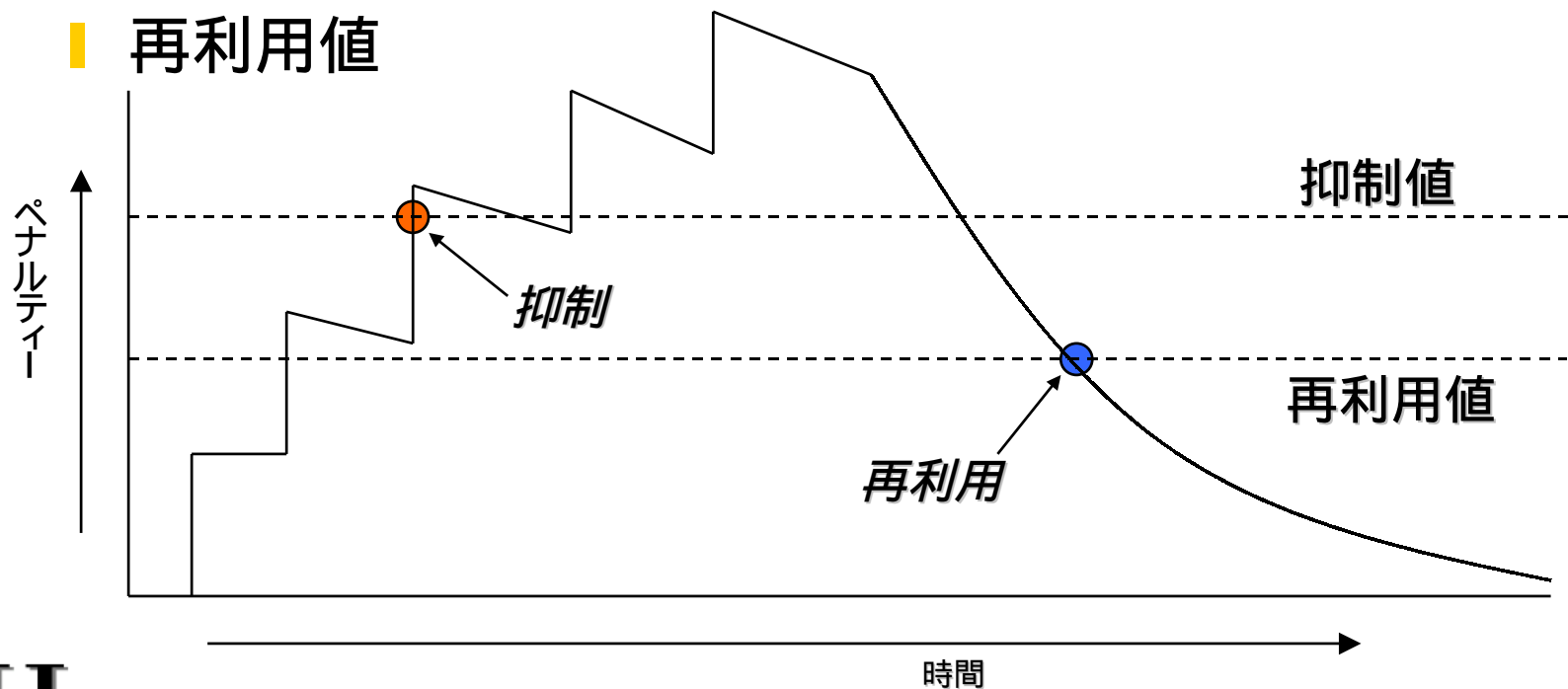


Route FlappingとDampening

- 不安定な経路情報
 - 上がったたり落ちたり、属性が変化したりを繰り返す(Route Flapping)
 - 大量のUPDATEやWITHDRAWメッセージの処理に、ルータのCPUやメモリ資源が浪費される
 - インターネット全体の経路制御システムへの影響
- ある程度以上不安定な経路情報は落ちたものとみなす(Dampening)
 - 一定の条件を満たすまで再びその経路は採用しない
- 時に板ばさみ
 - 経路をflapさせるほうが悪いのか、勝手にdampeningするほうが悪いのか？

Dampening

- パラメータ
 - ペナルティー
 - 半減期
 - 抑制値
 - 再利用値

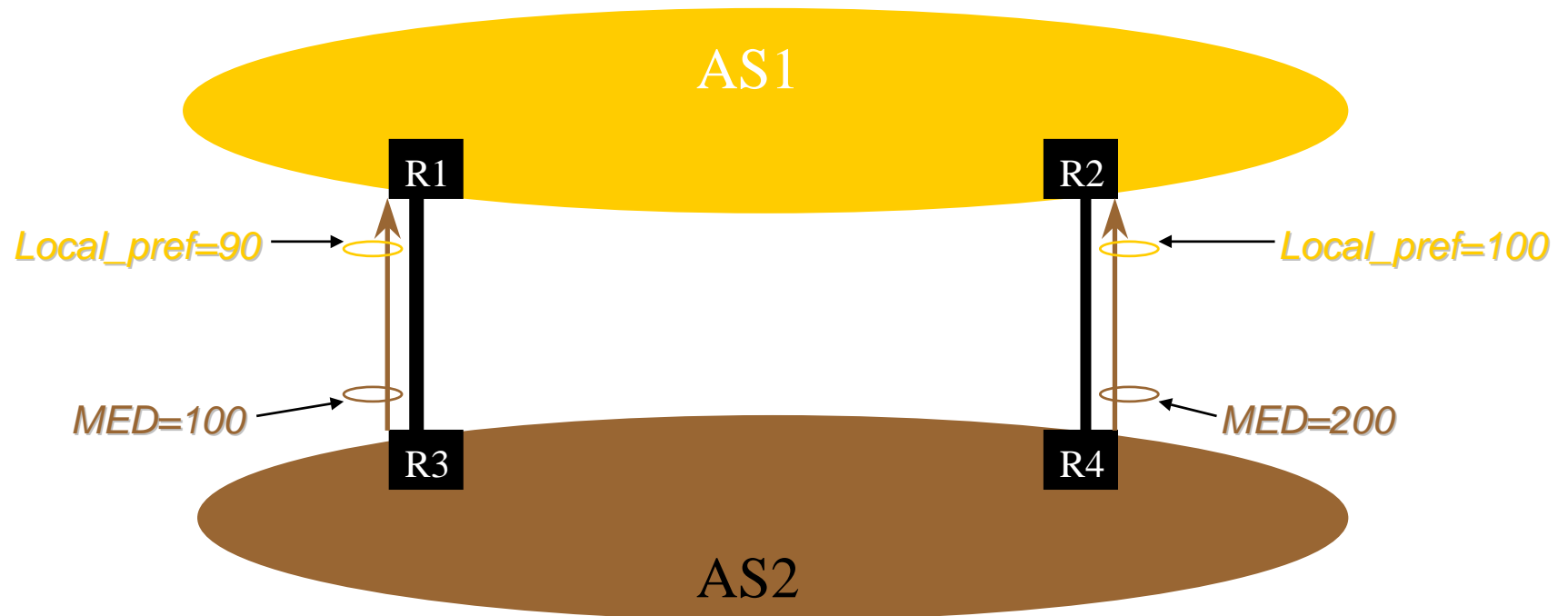


ポリシーの不整合

- 自ASのポリシーと、隣接ASやその他のASとのポリシーが擦り合わない場合がある
 - 隣接AS間
 - ┆ Hot PotatoかCold Potatoか
 - ┆ 相手のアナウンスするMEDを尊重するか否か
 - 顧客の経路とpeerの経路
 - ┆ 顧客から受け取る経路を最優先するISPもあるが、それ以外の経路を優先したいときもある
- 不整合を解決するために、特例的な設定を増やしていくと設定管理上の問題が発生

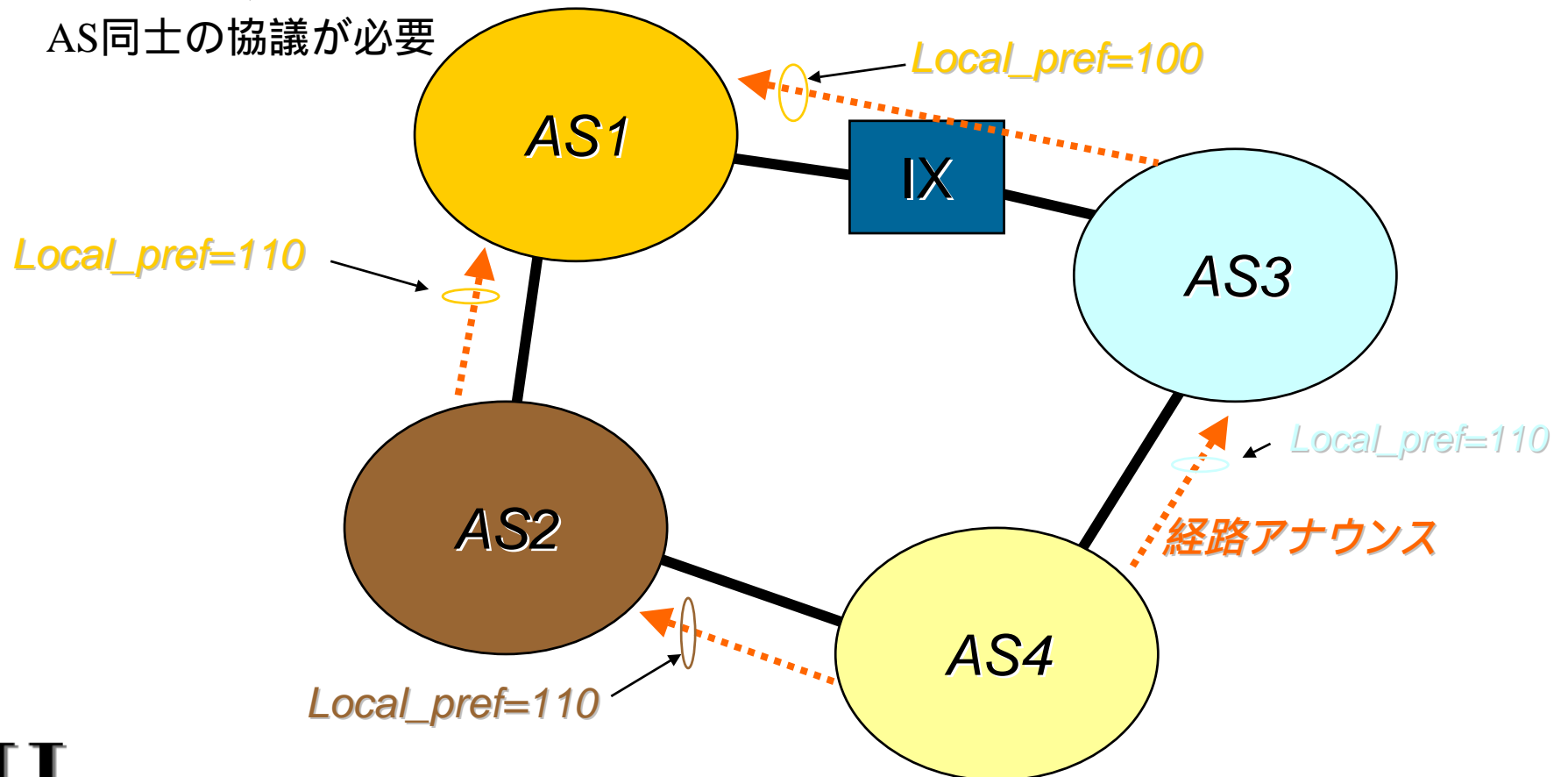
ポリシーの不整合（隣接AS間）

- 相手のMEDを無効化し自ASのポリシーを適用
- AS同士の協議が必要



ポリシーの不整合（顧客かpeerか）

- AS4はAS2とAS3の顧客、AS2はAS1の顧客、AS1とAS3はpeer
- 各ASは顧客からの経路情報を優先
 - AS1からAS4へのトラフィックはAS1 AS2 AS4と流れる
- だがAS4は、AS1 AS3 AS4の経路を優先したい
- AS同士の協議が必要



不正な経路情報

- 不正な経路情報のアナウンスによる事故
 - フルルートを自ASをoriginとして流してしまう
 - ┆ そのASがパケットのブラックホールとなる
 - 不必要なmore specific routeを流してしまう
 - ┆ Aggregateよりもmore specificにひきづられトラフィックが最適な経路を経由しない
 - IXのセグメントの経路をBGPで他のASに流してしまう
 - ┆ Next Hopの解決で最適経路を選べない

プレフィクス・ベース・フィルタリング

- 不正な経路アナウンスを防ぐために、プレフィクス単位でIngressフィルタを設定
 - 特に自分が通過を許可しているASからの経路アナウンスに対して
- フィルタの自動生成
 - RRの情報を元にフィルタを生成
 - RRの信頼性は？ (内容、動作)

経路情報のセキュリティー

- いくつかの提案はあるが、未だ発展途上
- アナウンスの信頼性をDNSを使って認証
 - どのプレフィクスをどのASがアナウンスしてもよいのか
 - DNSにASレコードを追加
 - draft-bates-bgp4-nlri-orig-verif-00.txt
- Secure BGP
 - IPSECを用いたpeerの認証、メッセージの完全性の保証
 - PKI(Public Key Infrastructure)を用いた、経路の生成許可証明と確認
 - <http://www.net-tech.bbn.com/sbgp/sbgp-index.html>

まとめ



まとめ

- BGP4はAS間の経路制御の標準プロトコル
 - 経路の選択にはパス属性を用いる
 - 実現できるポリシーは限られている
 - 細かなポリシーの実現にはネットワークトポロジの再考など必要
- BGP4を使えば良いというわけではない
 - 使わなくても良い場合もあれば、使わないほうが良い場合もある
 - 使ったがために運用管理が煩雑になることもある

まとめ

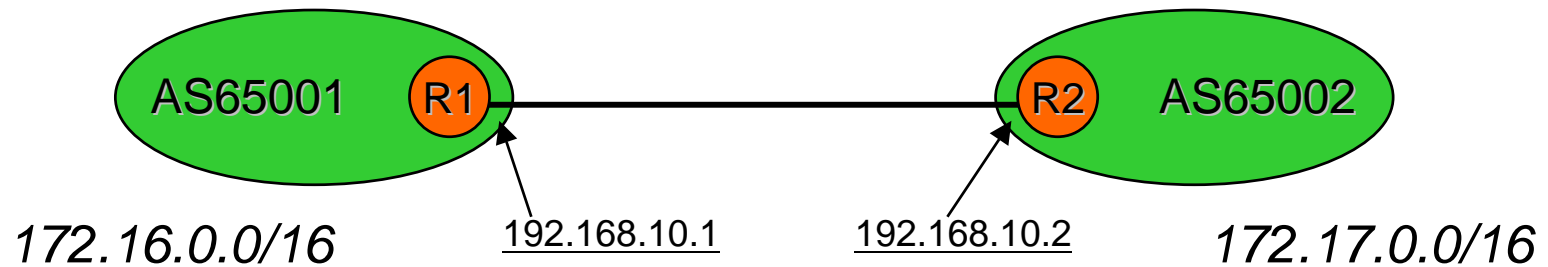
- ポリシールーティングは難しい
 - 単なるマルチホームでもトラフィックをうまく複数のリンクに分散することは難しい
 - できる限り、シンプルなネットワーク構成が望ましい
- 今後も技術開発が必要
 - RR, RS等の管理技術
 - RRの情報のUPDATE
 - 運用技術の確立
 - 設定の自動化
 - route flapping等の問題
 - セキュリティー

付録



サンプルコンフィギュレーション

Ciscoでの基本設定



R1での設定例

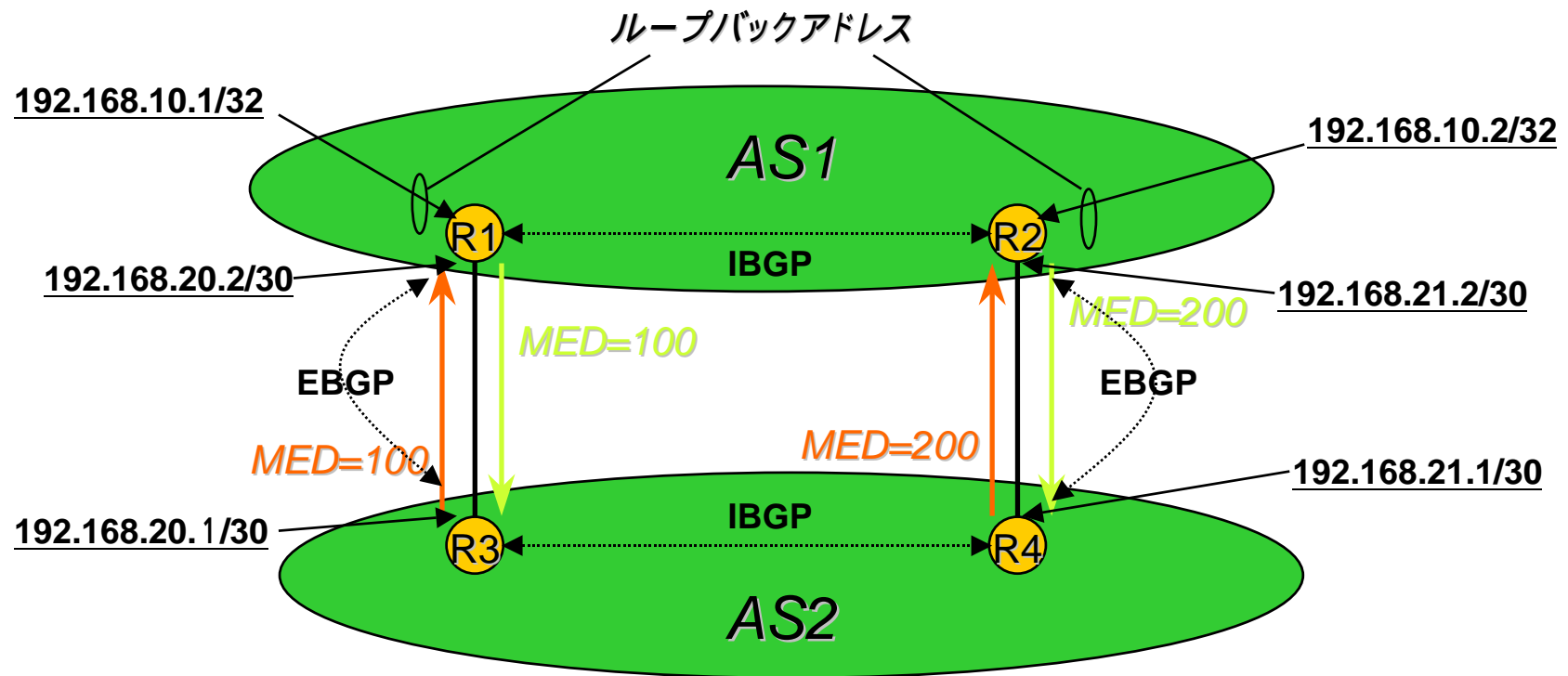
```
router bgp 65001  
network 172.16.0.0  
neighbor 192.168.10.2 remote-as 65002
```

R2での設定例

```
router bgp 65002  
network 172.17.0.0  
neighbor 192.168.10.1 remote-as 65001
```

MED

スライド28の例



MED

R1での設定例

```
interface loopback 0
ip address 192.168.10.1 255.255.255.255

router bgp 1
no synchronization
neighbor 192.168.10.2 remote-as 1
neighbor 192.168.10.2 update-source loopback0
neighbor 192.168.20.1 remote-as 2
neighbor 192.168.20.1 route-map MED-OUT out

route-map MED-OUT permit 10
match as-path 10
set metric 100

ip as-path access-list 10 permit ^$
```

R2での設定例

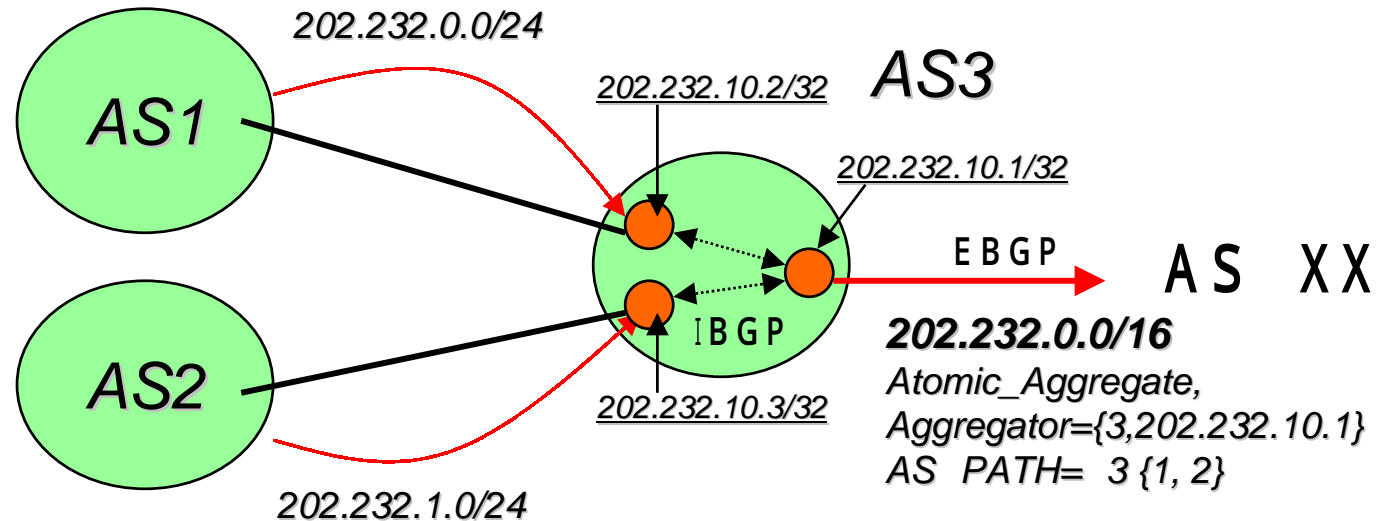
```
interface loopback 0
ip address 192.168.10.2 255.255.255.255

router bgp 1
no synchronization
neighbor 192.168.10.1 remote-as 1
neighbor 192.168.10.1 update-source loopback0
neighbor 192.168.21.1 remote-as 2
neighbor 192.168.21.1 route-map MED-OUT out

route-map MED-OUT permit 10
match as-path 10
set metric 200

ip as-path access-list 10 permit ^$
```

Aggregate

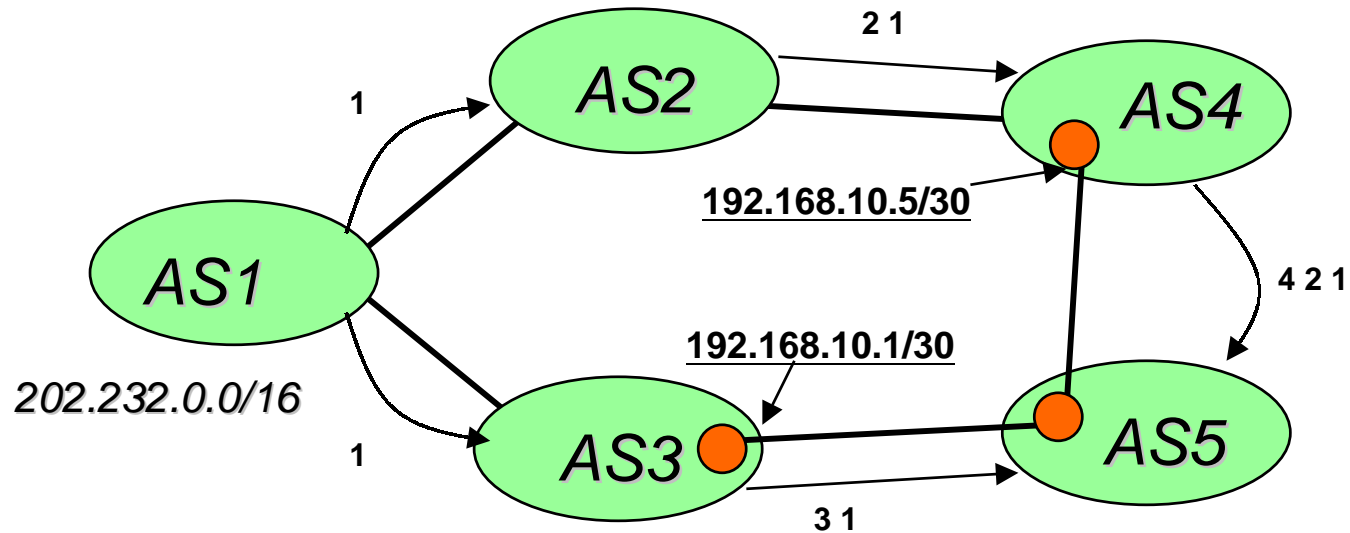


設定例

```
interface loopback 0  
ip address 202.232.10.1 255.255.255.255
```

```
router bgp 3  
no synchronization  
network 202.232.10.0  
aggregate-address 202.232.0.0 255.255.0.0 as-set summary-only  
neighbor 202.232.10.2 remote-as 3  
neighbor 202.232.10.2 update-source loopback0  
neighbor 202.232.10.3 remote-as 3  
neighbor 202.232.10.3 update-source loopback0  
neighbor X.X.X.X remote-as XX
```

Local-preference



AS5のボーダルータでの設定例

```
router bgp 5  
neighbor 192.168.10.1 remote-as 3  
neighbor 192.168.10.1 fromAS3 in
```

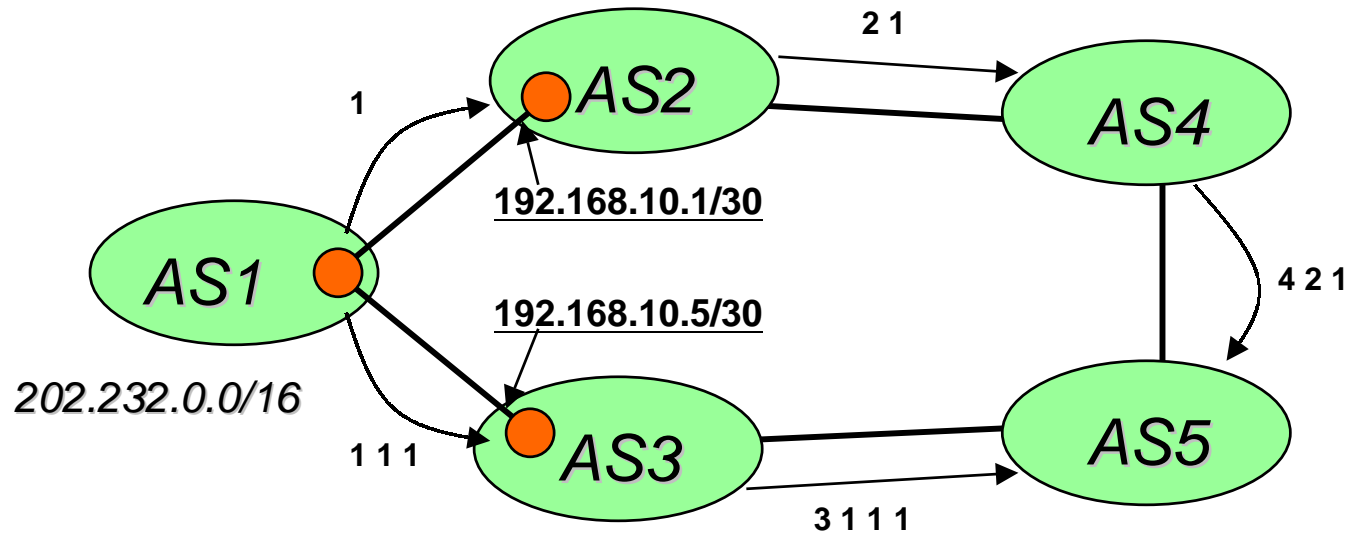
```
ip as-path access-list 10 permit ^3_1$
```

```
route-map fromAS3 permit 10  
match as-path 10  
set local-preference 90
```

AS1へは、AS4経由を優先したい

注: ciscoのlocal-preferenceの
デフォルト値は100

AS PREPEND

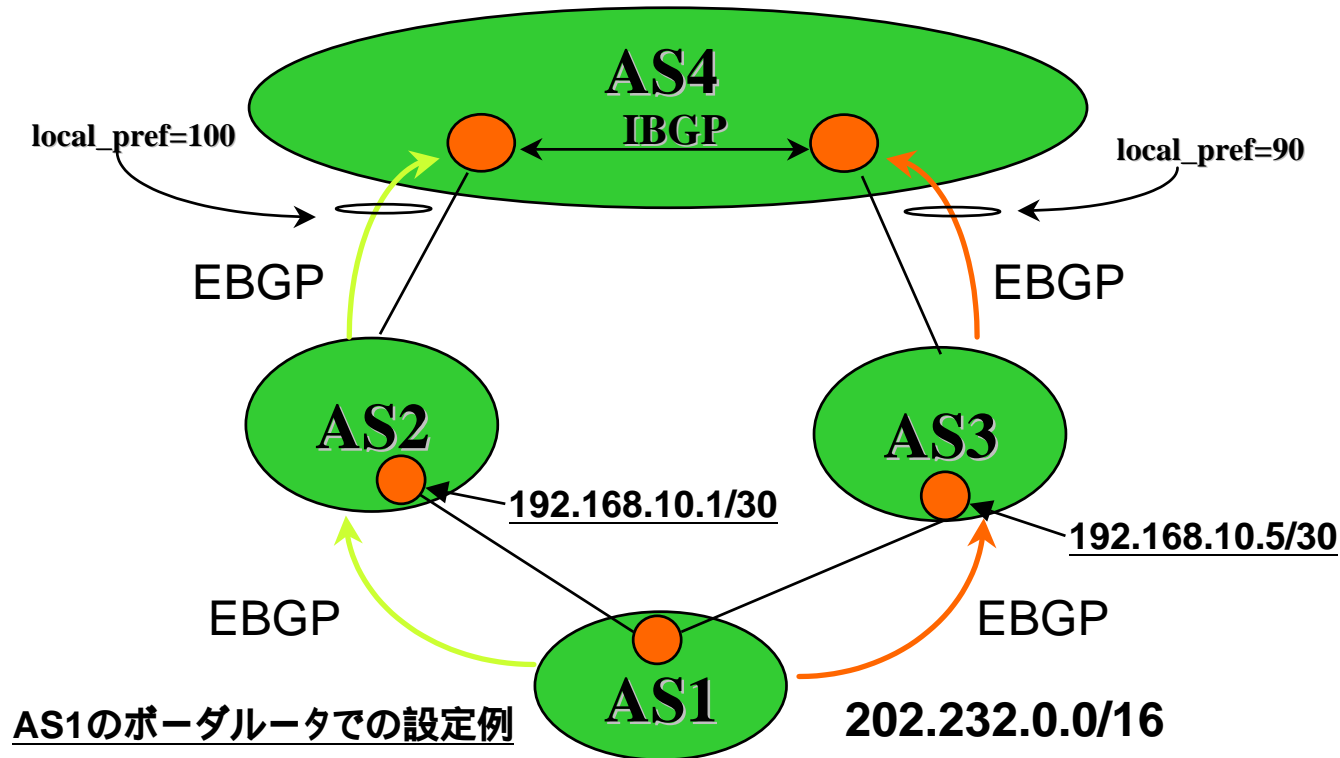


AS1のボーダルータでの設定例

```
router bgp 1
network 202.232.0.0 mask 255.255.0.0
neighbor 192.168.10.1 remote-as 2
neighbor 192.168.10.5 remote-as 3
neighbor 192.168.10.5 route-map PREPEND out
```

```
route-map PREPEND permit 10
set as-path prepend 1 1
```

Community



```
ip bgp new-format
access-list 10 202.232.0.0 0.0.255.255
```

```
router bgp 1
neighbor 192.168.10.1 remote-as 2
neighbor 192.168.10.1 send-community
neighbor 192.168.10.1 route-map toAS2 out
neighbor 192.168.10.5 remote-as 3
```

```
neighbor 192.168.10.5 send-community
neighbor 192.168.10.5 route-map toAS3 out
```

```
route-map toAS2 permit 10
match ip address 10
set community 4:100
```

```
route-map toAS3 permit 10
match ip address 10
set community 4:90
```