大規模ネットワークにおける経路制御設計

2001年12月 7日
NTTコミュニケーションズ 友近 剛史
イクアント 前村 昌紀

発表内容

タイトル	分	担当
(1) IGPのシステム設計論	80	友近
(2) BGPのシステム設計論	80	前村
(3) 大規模な経路制御設計の実際	20	
(3-1) 概要	5	前村
(3-2) Confederationの実例	5	前村
(3-3) static-to-bgpの実例	10	友近

(1) IGPのシステム設計論

内容

- ルーティングとは　～基本の復習～
- RIP
- OSPF
  - OSPFの基礎
  - OSPFの設定
  - OSPFの網設計
  - OSPFの仕組み
    ～大規模ネットワークにおいてOSPFの何が響くのか～
- IS-IS（時間の都合上参考資料のみ）

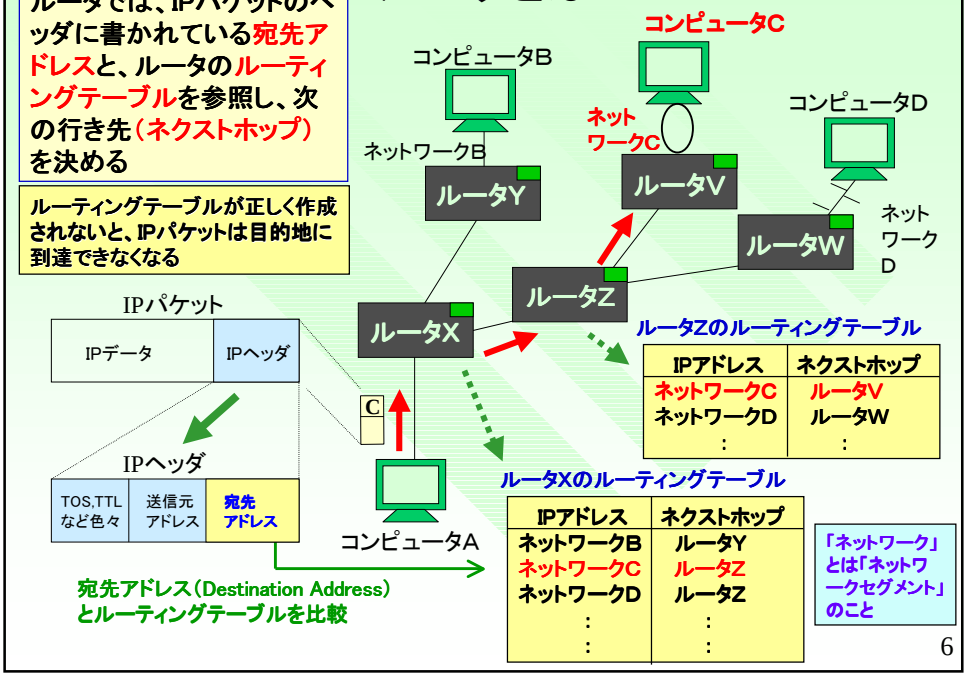
ルーティングとは

～基本の復習～

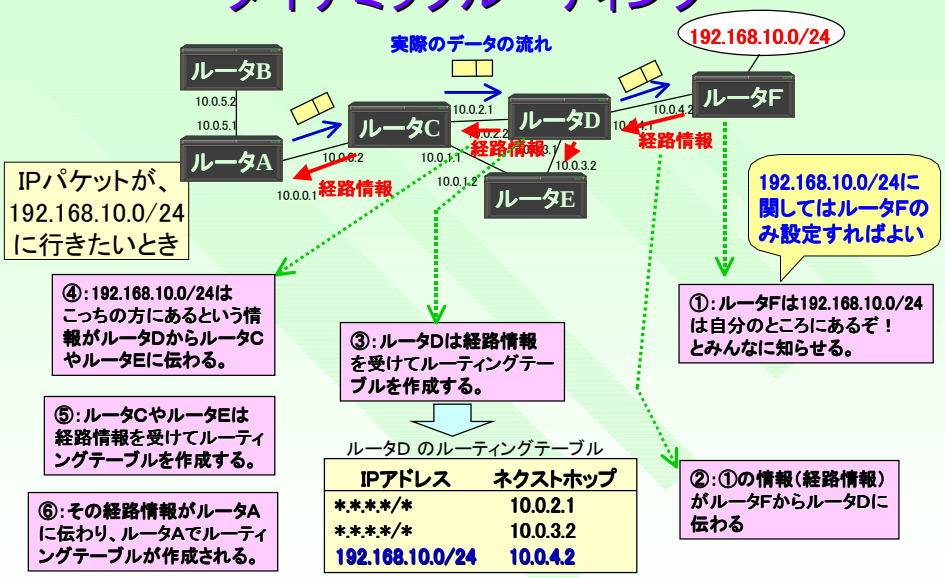
ルータとは

ルータでは、IPパケットのヘッダに書かれている宛先アドレスと、ルータのルーティングテーブルを参照し、次の行き先（ネクストホップ）を決める

ルーティングテーブルが正しく作成されないと、IPパケットは目的地に到達できなくなる

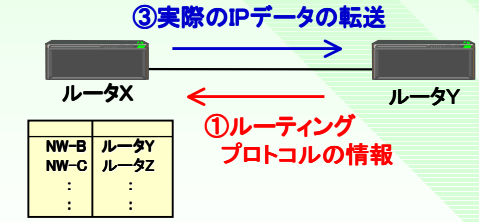


ダイナミックルーティング



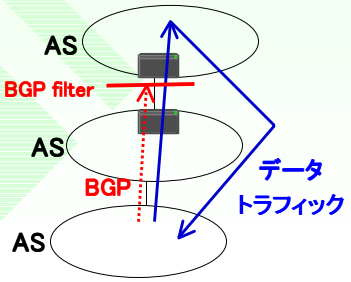
ダイナミックルーティング

(1)経路情報が伝わり、
(2)ルーティングテーブルができ、
(3)それに基づいてトラフィックが流れる。
→経路情報と実際のデータの向きは逆になる



②ルーティングテーブルの作成(ルータX)

トラブルシュートの例
下りのデータトラフィックが意図するところを通らないとき
→上りのBGP経路の伝わり方に問題あり



IGPとEGP

- IGPs（Interior Gateway Protocols）
  - 同一AS（Autonomous System:自律システム）内で使用されるルーティングプロトコル
  - RIP（Routing Infomation Protocol ）
  - OSPF（Open Shortest Path First ）
  - IS-IS（Intermediate System-to-Intermediate System）
- EGPs（Exterior Gateway Protocols）
  - AS間で使用されるルーティングプロトコル
  - BGP（Boarder Gateway Protocol ）

ルーティングプロトコル

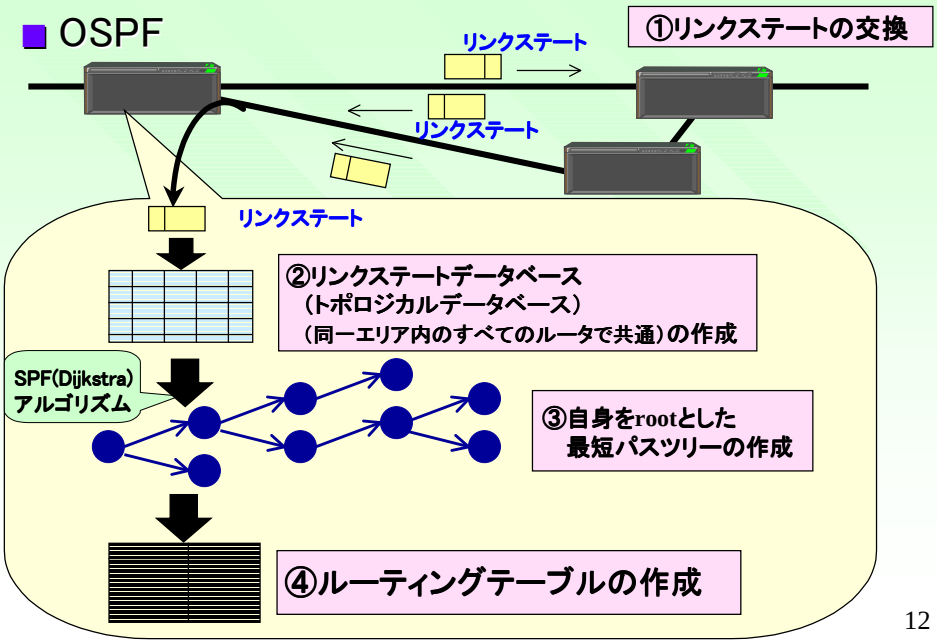
- ディスタンスベクターアルゴリズム
  - 隣接ルータ同士で経路情報を交換することでネットワーク情報を知る
  - 他のルータから受信したルーティングテーブルに自分が直接接続しているネットワークを加え、受信したインタフェース以外のインタフェースに流す
- リンクステートアルゴリズム
  - それぞれのルータが自分の接続しているネットワークについての情報等をネットワーク全体に通知する
  - 各ルータで共通のトポロジーデータベースを持つ
- パスベクターアルゴリズム
  - 経路情報が伝わっていく際に、経路情報にパス属性と呼ばれる付加情報がついて伝わる

ディスタンスベクターアルゴリズム

- RIP
- それぞれのルータが隣接しているルータとルーティング情報を交換することによって、ルーティングテーブルを構築する仕組み
- ルータは自分のもっているルーティングテーブルを接続しているネットワークに30秒ごとにブロードキャストする
  - 隣接したルータから受け取った情報（ネットワークアドレス）に自分の知っている情報を付加し送信する
- これが全ルータの間で繰り返し行われることでルータは接続されたすべてのネットワークとそこへの道筋を知ることができる
  - 収束に時間がかかる

リンクステートアルゴリズム

- OSPF

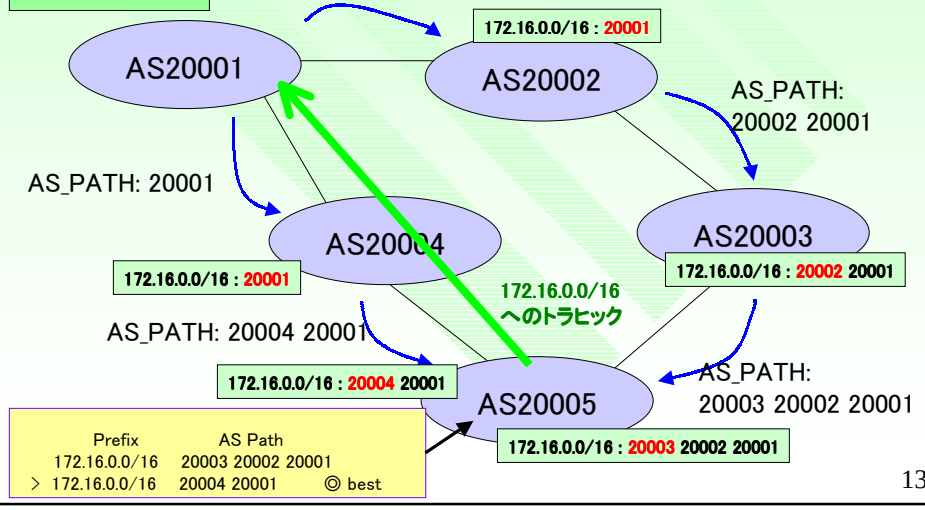


パスベクターアルゴリズム

■経路情報に付加されたPath属性（Path Attribute）を基づいて経路選択

AS20001が172.16.0.0/16を広告

- ルーティング情報は、ASを通り抜けるたびに自分のAS番号を付加していく
- AS_PATHの短い方が優先



RIP

RIP

- Routing Information Protocol
- ディスタンスベクターアルゴリズム
- UDP 520番を使用
- サブネットの情報を運ばない
- ルータは自分のもっているルーティングテーブルを接続しているネットワークに30秒ごとにブロードキャストする
  - 隣接したルータから受け取った情報（ネットワークアドレス）に自分の知っている情報を付加し送信する
  - これが全ルータの間で繰り返し行われることでルータは接続されたすべてのネットワークとそこへの道筋を知ることができる
  - 収束に時間がかかる
- 古くからBSD UNIXシステム上でroutedという形で実装されている
- 実装は簡単で、多くの機器で実装されている

RIPのメリットとデメリット

- メリット
  - 多くのネットワーク機器で対応されている
  - 処理の負荷が小さい
- デメリット
  - サブネットマスクの情報を運ばない
    - » VLSM非対応
  - ディスタンスベクター方式のため、網変更等の際、収束に時間がかかる
  - デフォルトの設定で、30秒に1回、各ルータは自分のもっているすべてのルーティング情報を隣接ルータへブロードキャストで送出する
    - » 経路情報のトラヒックが多い
    - » RIPに参加していないノードも無関係な情報の処理で無駄を生じる
  - 最大のホップ数は15までしか対応できない
  - ホップ数で比較なので、回線の帯域に応じて適切な経路を選ぶことが難しい

VLSM

- Variable Length Subnet Mask
- VLSMとは1つのネットワークをサブネットに分割する場合に複数の長さのサブネットマスクを使用する方法
- 例えば、あるクラスCを分割するときに/26と/27を同時に利用したりすること
- 例えば、同じクラスCでは同じprefix長しか使えない、というのはVLSMに対応していない、という
  - 逆に言うと、RIPでも、あるルータであるクラスCをすべて/26で使用し、また他のあるクラスCをすべて/27で使用する、ということはできる。
- なお、クラスCで/24しか使えないというのはサブネットに対応していない、という状況
  - ip classless
  - ip subnet-zero

  は忘れないように！

RIP2

- RIP1と完全後方互換性
- RIP1を少し直した感じ
- サブネットマスクの情報を運ぶ
  - VLSM対応
- 経路情報をブロードキャストだけでなくマルチキャストでも行える
- 認証機構を提供
- しかし、RIP1と同じくディスタンスベクター方式である
  - デフォルトで30秒に1回、各ルータは自分のもっているすべてのルーティング情報を隣接ルータへ送出する
  - 網変更等の際、収束に時間がかかる

RIP1,RIP2ともに大規模ネットワークには適さない

※ただし、大きくないネットワークではお手軽に使える便利なルーティングプロトコル

OSPF

OSPFの基礎

OSPFについて

- RFC 1247(July 1991)
- →RFC 1583 (March 1994)(9箇所変更backward-compatible)
- →RFC 2178 (July 1997) (10箇所変更backward-compatible)
- →RFC 2328 (April 1998) (4箇所変更backward-compatible)

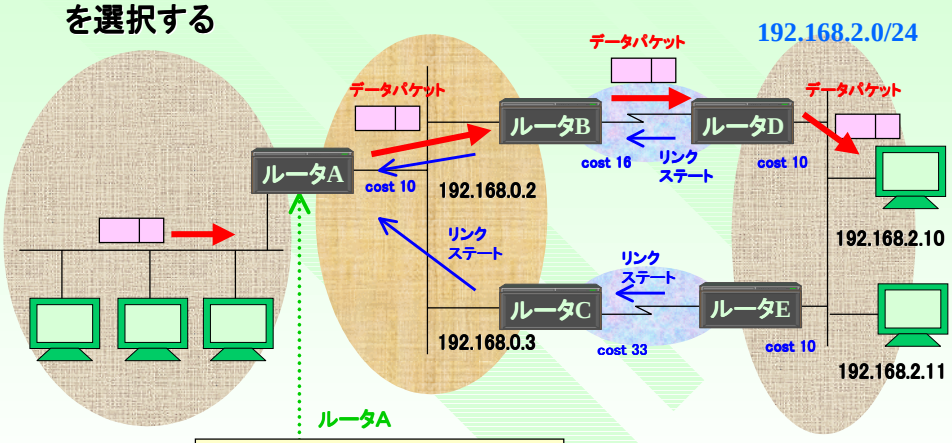
- Open Shortest Path Fast
- version 2
- リンクステートアルゴリズム
- IPを直接使用し、プロトコル番号89
- VLSM対応
- マルチキャストでlink-stateを配布

リンクステート

- トポロジーの変更があったときだけ、link-stateのupdateが送信される
  - リンクステートとはリンクのステートの情報のこと
    - » あるルータのリンク(インタフェース)のステート、つまりIPアドレス、マスク、接続されるネットワークタイプ、そのネットワークに接続されるルータ、等のこと
    - » それらのリンクステートが集まって、トポロジーDBを形成する
      - ルーティングテーブルを交換しない
  - トポロジー変化のないときでも定期的に30分に一回LSAをrefreshする

コスト

- 同じネットワークが複数見える場合、コストが一番低い経路を選択する



ルータA

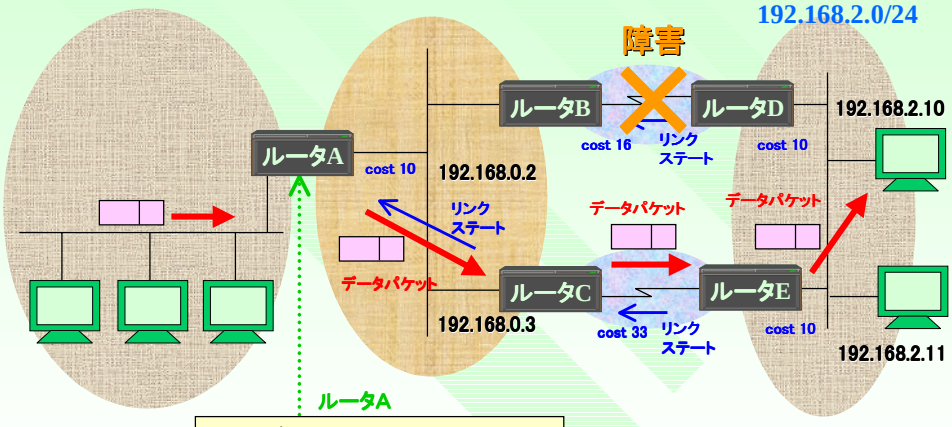
	IPアドレス	ネクストホップ	コスト
◎	192.168.2.0/24	192.168.0.2	36
	192.168.2.0/24	192.168.0.3	53

コストが一番低い経路を選択

※正確にはルーティングテーブル(フォワーディングテーブル)にのるのがそれだけになる

障害時にはバックアップ経路に切り替わる

- 障害時には、コストの高いバックアップ経路に切り替わる



ルータA

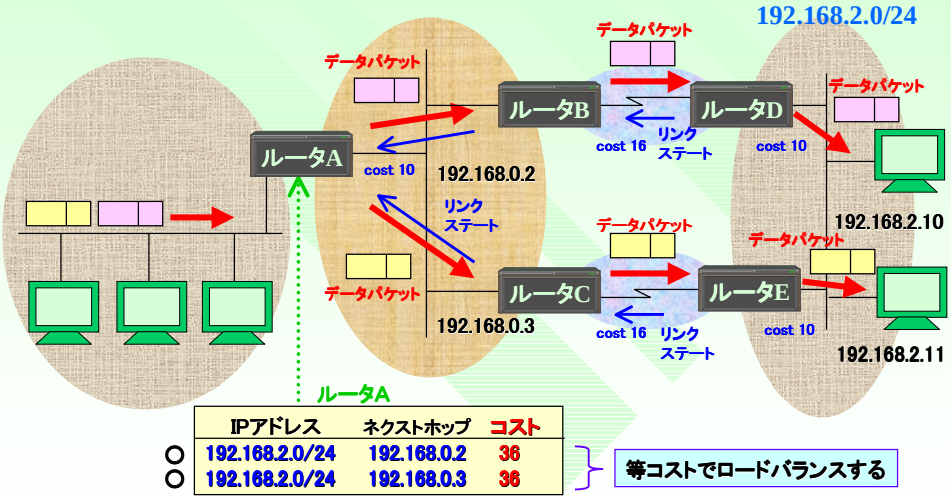
	IPアドレス	ネクストホップ	コスト
	~~192.168.2.0/24~~	~~192.168.0.2~~	~~36~~
◎	192.168.2.0/24	192.168.0.3	53

コストの高いバックアップ経路に切り替わる

※正確にはコストの低い経路がルーティングテーブルから消え、コストの高い経路が現れる

ロードバランス

- 同じネットワークが同じコストで見えるネットワークに対しては、トラフィックが半分ずつ分散する（ロードバランス）



ルータA

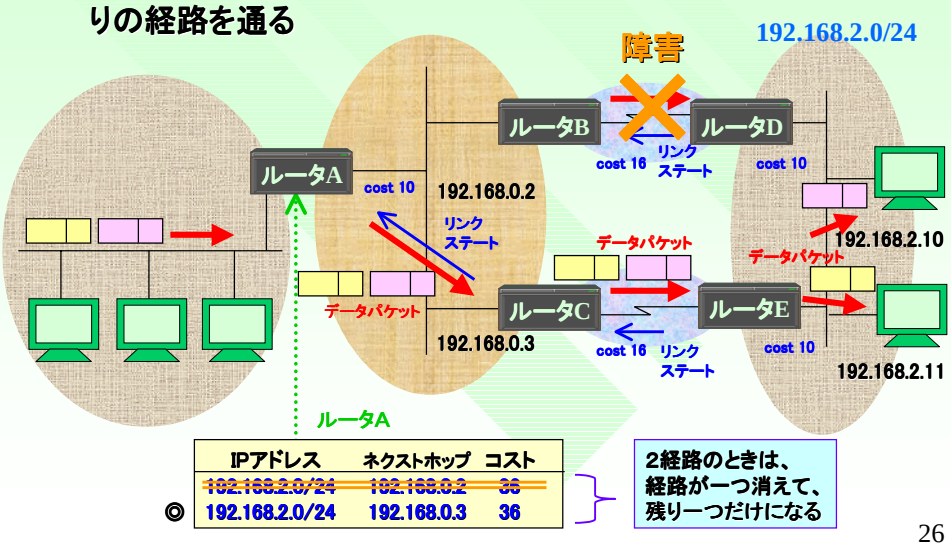
	IPアドレス	ネクストホップ	コスト
○	192.168.2.0/24	192.168.0.2	36
○	192.168.2.0/24	192.168.0.3	36

等コストでロードバランスする

※OSPF的にはイコールコストマルチパスをサポートし、そして、一般的なルータでは、
イコールコストマルチパスについてはトラフィックを半分ずつに分散する

障害時は全てのトラヒックが残りの経路を通る
～ロードバランス時～

- 障害時には、障害した経路が消えて全てのトラヒックが残りの経路を通る



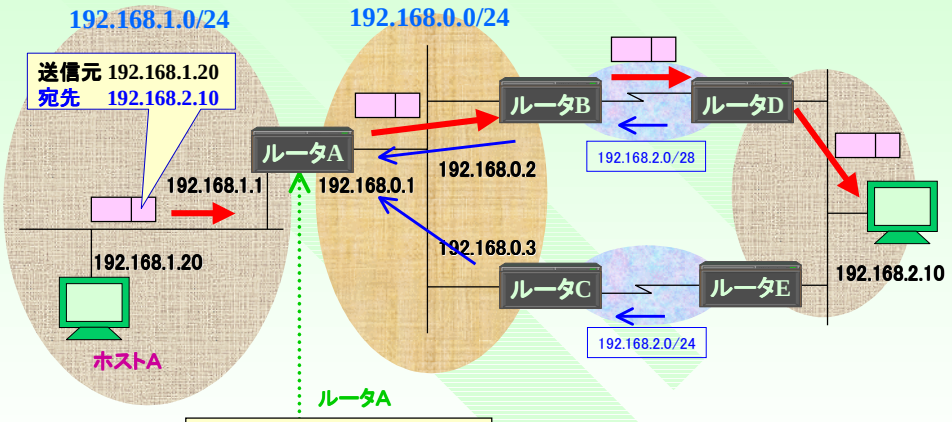
ルータA

	IPアドレス	ネクストホップ	コスト
~~◎~~	~~192.168.2.0/24~~	~~192.168.0.2~~	~~36~~
◎	192.168.2.0/24	192.168.0.3	36

2経路のときは、
経路が一つ消えて、
残り一つだけになる

最長一致(longest match)ルーティング規則

- IPパケットの宛先アドレスを調べて、一致するネットワークアドレスが複数ある場合には、ビット列が長い方のネットワークアドレスを選択する



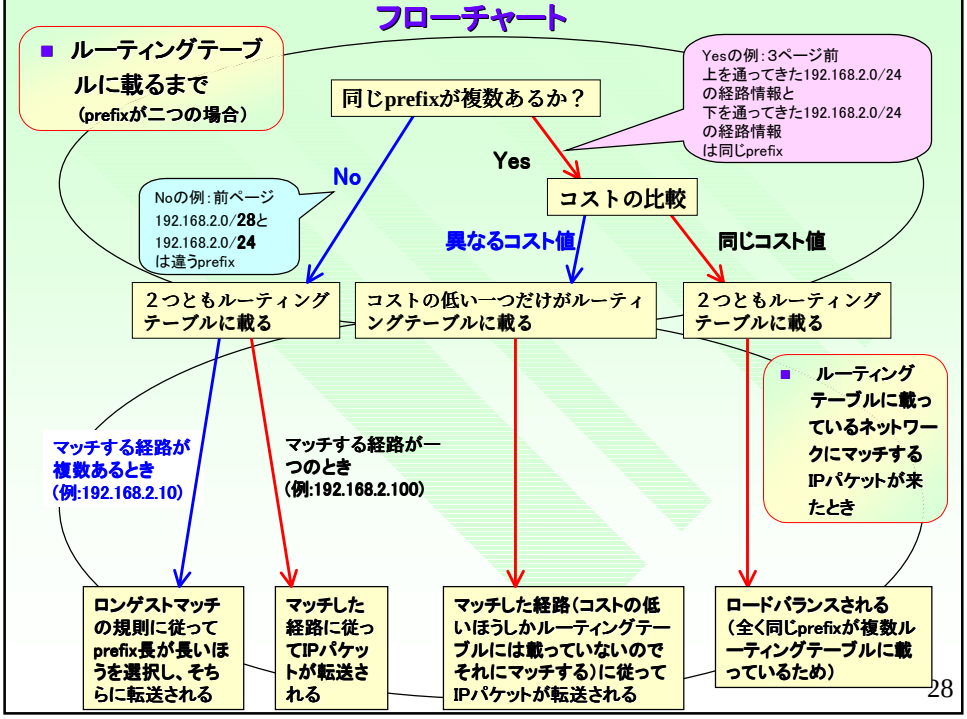
ルータA

	IPアドレス	ネクストホップ
C	192.168.0.0/24	192.168.0.1
C	192.168.1.0/24	192.168.1.1
	192.168.2.0/28	192.168.0.2
	192.168.2.0/24	192.168.0.3

Longest matchで192.168.0.2を選択する

フローチャート

- ルーティングテーブルに載るまで
(prefixが二つの場合)



- ルーティングテーブルに載っているネットワークにマッチするIPパケットが来たとき

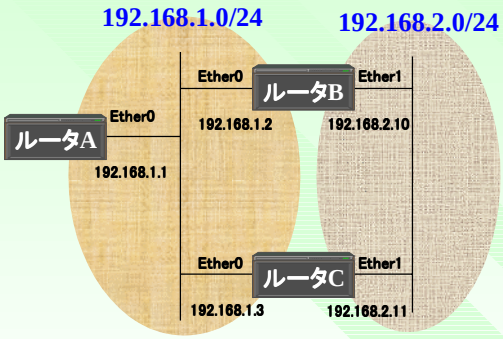
OSPFの設定

OSPF設定(C社の例)

- router ospf <process ID>
  - 自分のASと同じ番号にすることが多い
    - » 一つのAS内で一つしかOSPF processを走らせない場合
    - » process IDは1～65535の何番にしてもいい
- network 192.168.0.0 0.0.0.15 area 0
  - 0.0.0.15はワイルドカードマスク
    - » アドレスのうち無視する部分をマスクする
  - 192.168.0.0～192.168.0.15の範囲にあるアドレスのインタフェースで
    - » OSPFを area 0 で話す
    - » そのインタフェースのネットワークをOSPFに広告する

上記2つが基本で、最低限のOSPFのconfig

OSPFの基本設定(C社の例)



ルータAの設定

```
interface Ethernet0
 ip address 192.168.1.1  255.255.255.0
!
router ospf 1
 network 192.168.1.0  0.0.0.255  area  0
```

ルータBの設定

```
interface Ethernet0
 ip address 192.168.1.2  255.255.255.0
!
interface Ethernet1
 ip address 192.168.2.10  255.255.255.0
!
router ospf 1
 network 192.168.1.0  0.0.0.255  area  0
 network 192.168.2.0  0.0.0.255  area  0
```

ルータCの設定

```
interface Ethernet0
 ip address 192.168.1.3  255.255.255.0
!
interface Ethernet1
 ip address 192.168.2.11  255.255.255.0
!
router ospf 1
 network 192.168.1.0  0.0.0.255  area  0
 network 192.168.2.0  0.0.0.255  area  0
```


ワイルドカードマスクについて

ルータCの設定

```
interface Ethernet0
 ip address 192.168.1.3  255.255.255.0
!
interface Ethernet1
 ip address 192.168.2.11  255.255.255.0
!
router ospf 1
 network 192.168.1.0  0.0.0.255  area  0
 network 192.168.2.0  0.0.0.255  area  0
```

OSPFのnetworkコマンドの
アドレス　192.168.1.0
ワイルドビットマスク　0.0.0.255

2進数に直すと

11000000.01010100.00000001.00000000
00000000.00000000.00000000.11111111

論理和

11000000.01010100.00000001.11111111

(A)とする

Ethernet0

・Ethernet0のアドレス　192.168.1.3
・OSPF networkコマンドの
ワイルドビットマスク　0.0.0.255

2進数に直すと

11000000.01010100.00000001.00000011
00000000.00000000.00000000.11111111

論理和

11000000.01010100.00000001.11111111

(A)とマッチするのでEthernet0でOSPFを話す

Ethernet1

・Ethernet1のアドレス　192.168.2.11
・OSPF networkコマンドの
ワイルドビットマスク　0.0.0.255

2進数に直すと

11000000.01010100.00000010.00001011
00000000.00000000.00000000.11111111

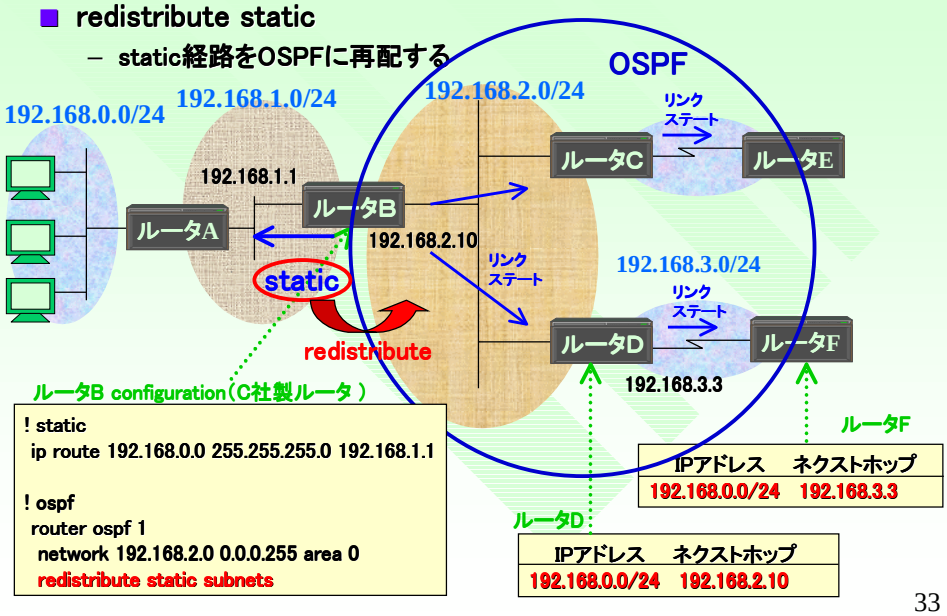
論理和

11000000.01010100.00000010.11111111

(A)とマッチしないのでEthernet1でOSPFを話さない
※この例では2行目のnetworkコマンドにマッチするので結局OSPFは話す

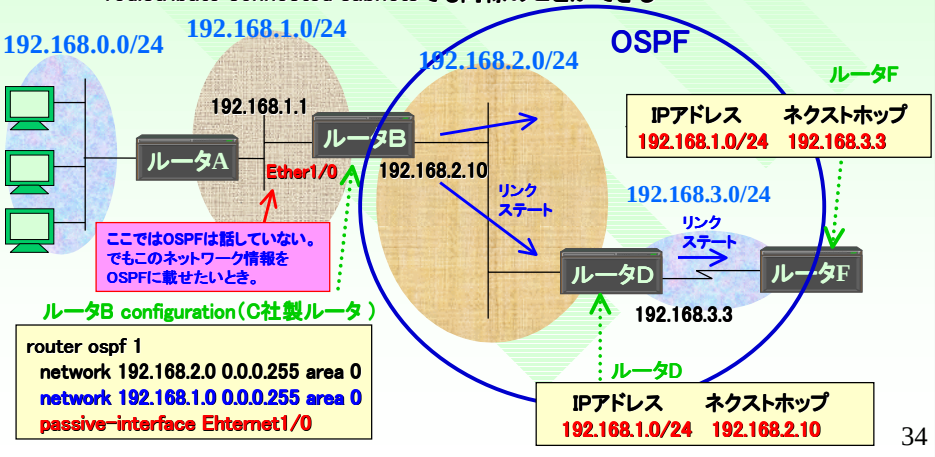
redistribute

- redistribute static
  - static経路をOSPFに再配する



passive-interface

- passive-interface Ethernet1/0
  - そのインタフェースでOSPFを話さない
  - OSPFにとってstubなnetworkな場合、そのネットワークをOSPFに広告したいが、そのネットワークでOSPFを話さない方がいい、ということが多い。 そのときに、networkコマンド + passive-interface でやる
  - redistribute connected subnetsでも同様のことができる



Interfaceの設定(C社の例)

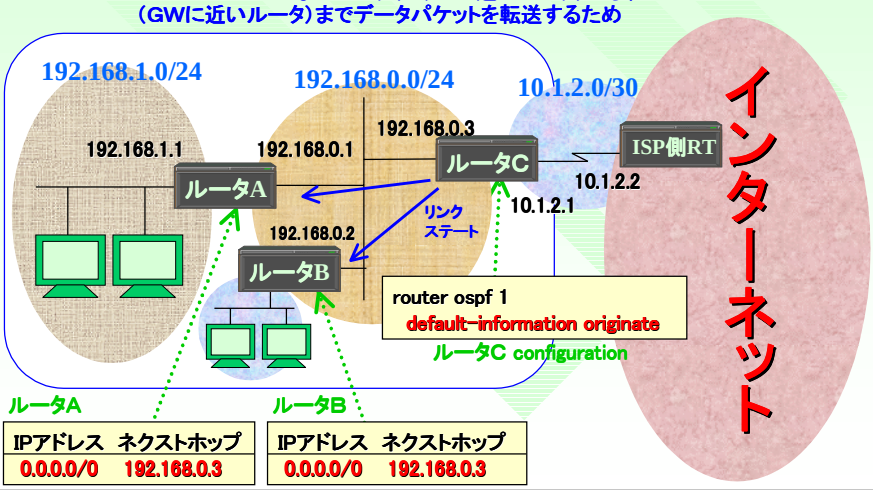
- コストによるネットワークごとの重み付けができる
  - デフォルト 100M/回線速度(bps)
  - ip ospf cost <cost>
  - そのインタフェースからデータパケットが出るときのためのコスト
    - » 非対称でもよい
- 通常は流れるのは10秒に一回のHelloだけ
  - ブロードキャストネットワーク(Ether)やPoint-to-Pointで10秒
    - » ノンブロードキャストネッワーク(X.25公衆網など)では30秒
  - ip ospf hello-interval <seconds>
- デッドタイマー
  - HELLOtを受け取らなければ障害だと判断
  - デフォルト HELLOインターバルの4倍
  - ip ospf dead-interval <seconds>

OSPF設定(C社の例)

- その他
- 同一コストの複数パスを同時に使用できる
  - ロードバランス
  - 6つまで
  - maximum-paths 6　　(rotuer ospf **で)
- 認証
  - ip ospf authentication-key *****　(interfaceで)
  - area ** authentication　　(router ospf **で)

デフォルトルートの生成

- デフォルトルートの生成
  - デフォルトルートを広告する
  - そのルータにデフォルトが向く
  - 外部に近いルータで設定する
    - » BGPスピーカーでないルータ(エッジに近いルータ)から、BGPスピーカー(GWに近いルータ)までデータパケットを転送するため



デフォルトルート(C社の例)

- デフォルトルートの生成
  - default-information originate
    - » そのルータにデフォルトルートの情報が既にある場合だけ広告
  - default-information originate always
    - » そのルータにデフォルトルートの情報がない場合はalwaysが必要
  - デフォルトルートを広告するルータに、知らないアドレス向け(例:プライベートアドレス)にパケットが来た場合そのパケットを廃棄しなくてはならない。この処理はかなり重いため、できればインタフェースで廃棄できるようなルータ(例:GSR)でデフォルトルートを広告すべき
    - » C7513+RSP4でも、廃棄パケットが20～30Mbpsでかなり苦しい
    - » CPU負荷検証などでも、廃棄専用のルータを用意すべき
  - default-route広告ルータは他のdefault-route広告ルータが生成したdefault-routeを受け取らない

External routesとメトリックのタイプ

- External routes
  - staticや他のルーティングプロトコルからredistributeされた経路
  - そのルータはAS*境界ルータになる

*ここでいうASとは共通の経路制御プロトコルを用いて経路情報を交換しているルータのグループのこと。いわゆるBGP等でいうASとは違う

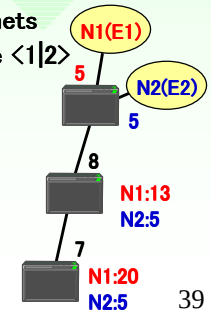
- メトリックのタイプ
  - type1：externalのコストにinternalコストを加えていく
  - type2：externalのコストのまま
- ルータconfiguration
  - redistribute **** metric <metric> metric-type <1|2> subnets
  - default-information originate metric <metric> metric-type <1|2>
    - » これもexternalになる
  - デフォルトはtype2
  - 同じネットワークに関しては常に(メトリックに関わらず)

intra-area > inter-area > external E1 > external E2

(O　O IA　O E1　O E2)

の順番で優先される

sh ip routeの出力



OSPFの網設計

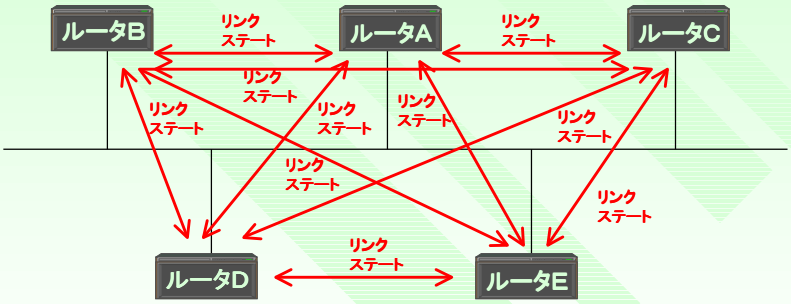
網設計においての基本

- まずは、要望条件を整理し、ポリシーを策定する
- 例
  - 基本機能の実現
    - » 静的状態での接続性
    - » 迂回機能の実現
  - 信頼性の向上
    - » リンク障害
    - » ノード障害
    - » 機種レベルでの冗長化
    - » メディアレベルの冗長化（例：Giga-EtherとFDDI）
    - » ビル障害

要望条件、ポリシー（続き）

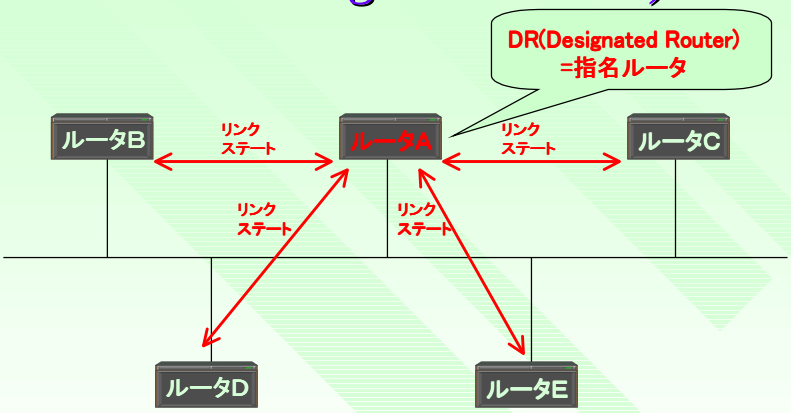
- コストの低減
  - » 回線数、回線帯域の削減
  - » アクトスタンバイにするか、ロードバランスにするか
  - » 1 + 1 にするか、n + 1 にするか
- 保守運用性の向上
  - » 物理的にシンプルであること
  - » 論理的にシンプルであること
  - » 地域的、サービス的に分離可能であること
  - » 移行が容易であること
- 将来性
  - » ビル数、ノード数、ユーザ数の増大対応
  - » サービス種類の増大対応

DR（Designated Router）



- DRがないとあるネットワークセグメントでのリンクステートのやりとりがフルメッシュ的になってしまう

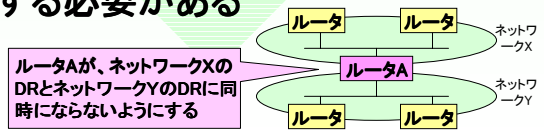
DR（Designated Router）



- ネットワークセグメント上で、一つのルータをDRにすることによりリンクステートのやりとりを減らす

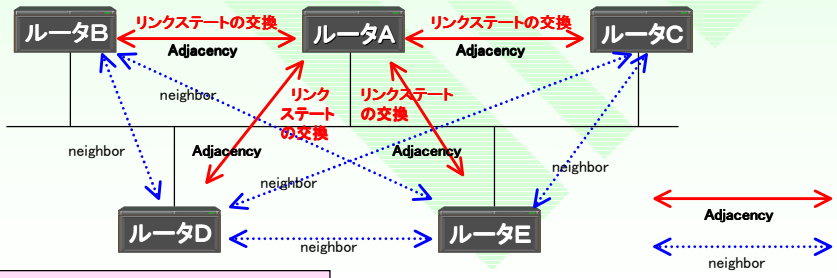
DRとBDR

- Designated Router：指名ルータ
- Backup Designated Router：バックアップ指名ルータ
- マルチアクセスネットワーク上で必ず１つ存在
- BDRはDRがダウンしたときのバックアップ
- DRとBDR以外のルータをDROTHERという
- DRは結構負荷がかかるので、処理能力のあるルータや、他の処理が重くかかっていないものになるなど、選定に考慮する必要がある
- 一つのルータが複数のネットワークのDRになるべくならないように考慮する必要がある



DRとBDR

- Helloプロトコルで決定される
- AdjacencyはLink-stateをやりとりする関係
- 単純にHelloパケットをやりとりするのはNeighbor関係
- よって、DROTHER同士はNeighborであるがAdjacencyではない

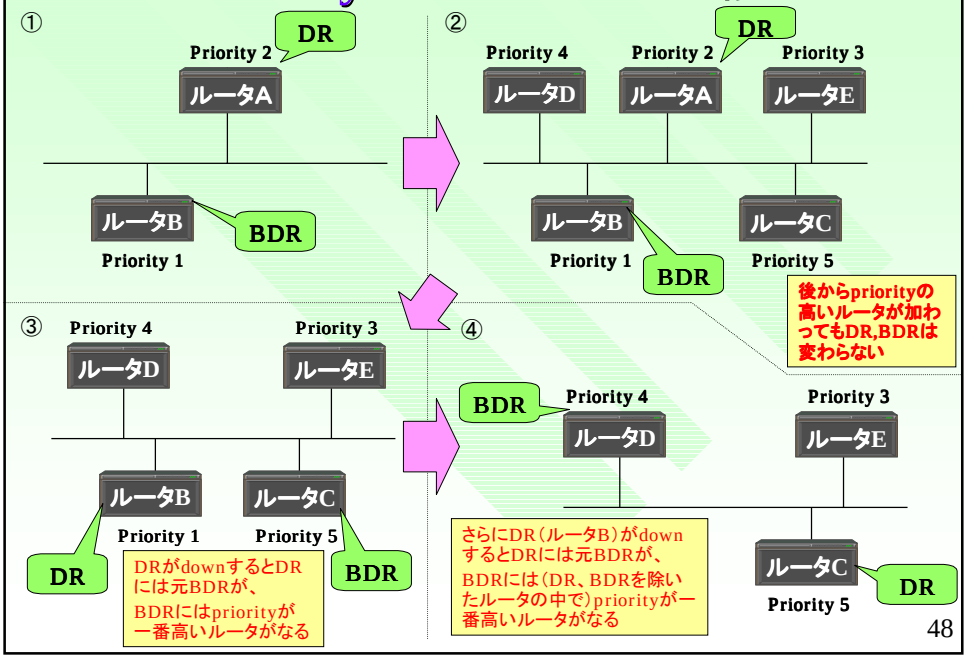


注：BDRとDROTHERはAdjacentになっているが、BDRからのfloodingは省略される

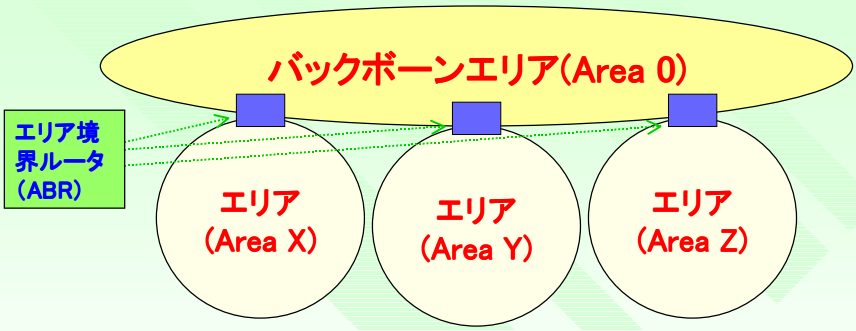
priority

- DRになりやすさの値
- ip ospf priority ＊　　(interface)
- ospf priorityの値が高いほど優先される
- しかし、対象とするネットワークですでにDR/BDRが存在するときにはDROTHERとなる
  - 結局最初に立ち上げた2つのルータがDR/BDRとなる
- よって、ネットワークを新規に立ち上げる時などは、priorityが高いものから起動させるのが望ましい
- ospf priority 0はDR/BDRに選ばれない
  - 負荷が大きくなると困るルータなどは0にする

PriorityとDR/BDRの遷移



エリアについて



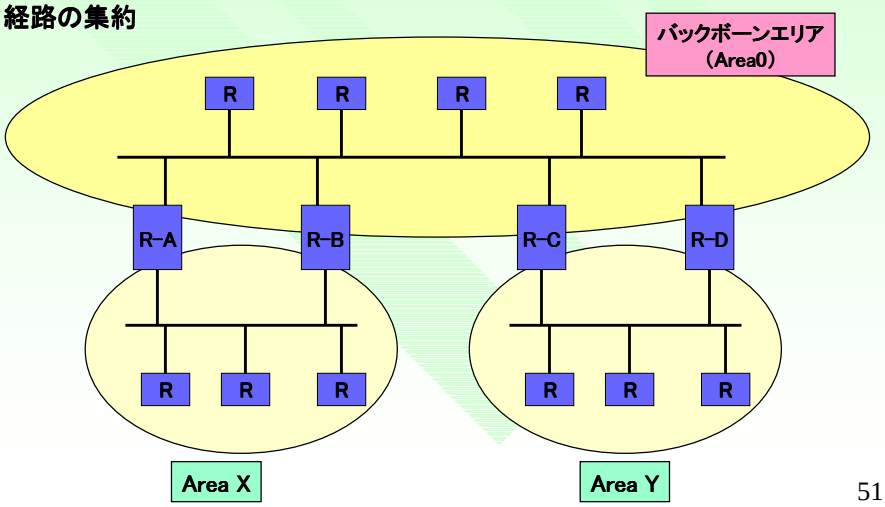
- エリア
  - 同一エリア内のすべてのルータでトポロジーデータベースは共通
  - バックボーンエリアに他のエリアがぶら下がる形
- エリア境界ルータ
  - エリアを結ぶルータをエリア境界ルータと呼ぶ
  - 必ずバックボーンエリア(エリア0)には属する

エリアについての設計

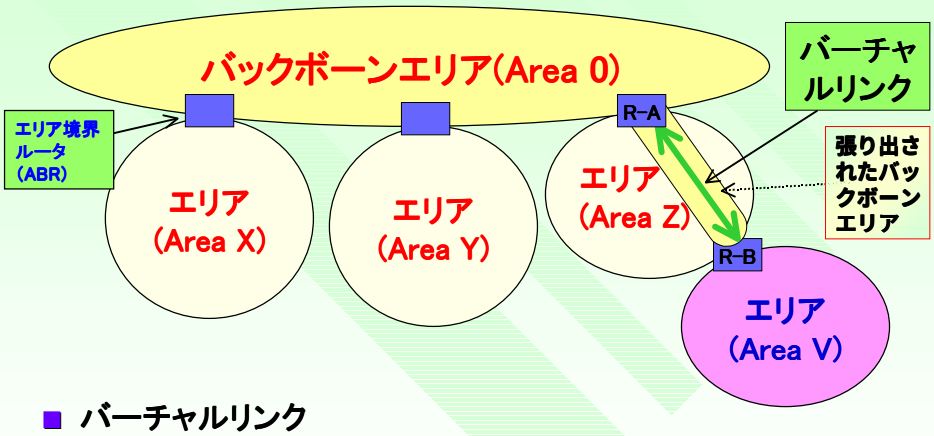
- まず、エリア0を構築して(または考えて)、それ後その他のエリアを拡張していく(または考えていく)
  - エリア0は全てのエリアの中心
- 信頼性を必要とするNWであれば、リダンダンシーのため一つのエリアでは複数のエリア境界ルータを置くべき
- 一つのエリア境界ルータが所属するエリアはなるべく2つまでにすべき
  - つまりエリア0ともう一つのエリア、というようになる
- 経路の集約
  - エリア境界ルータにて経路の集約をする
  - エリアごとに経路を集約できるように、アドレス設計をする
    - » area ** range <address> <mask>　(エリア境界ルータ)
  - OSPFにredistributeされる経路も集約できるように、アドレス設計する
    - » summary-address <address> <mask>　(AS境界ルータ)

エリアについての設計

- リダンダンシーのため、一つのエリアでは複数のエリア境界ルータ
- 一つのエリア境界ルータが所属するエリアはなるべく2つまで
- 経路の集約



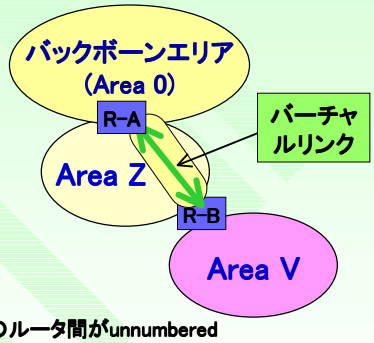
バーチャルリンク



- バーチャルリンク
  - バックボーンに対して物理的コネクションを持たないエリアを取り込む

エリアについての設計（バーチャルリンク）

- バーチャルリンクをあてにしてネットワークを設計すべきでない
  - 設計が複雑になる
  - 冗長性確保が難しい
  - AreaVを0以外にするとRouter-Bが3つのエリアに所属してしまう。これはあまり好ましくない。
  - よってAreaVをArea0とするが、するとArea0が大きくなって、規模対応性に関してはあまり効果が得られない
  - Area0につなげるというより、Area0を拡大するイメージ



・Virtual linkはArea0の一部であり、2つのルータ間がunnumbered なp-to-pネットワークで接続されているように振る舞う

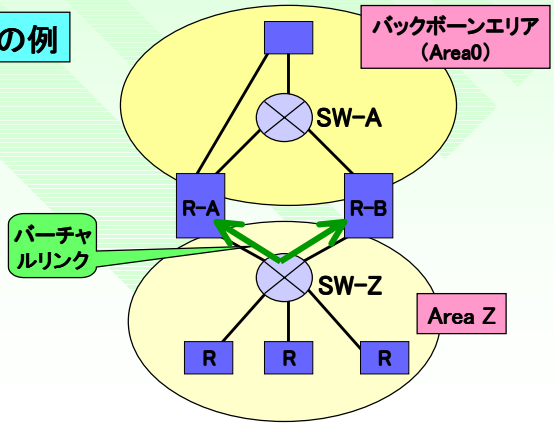
- R-A
  - area Z virtual-link <Router-BのR-ID(loopbackアドレス)>
- R-B
  - area Z virtual-link <Router-AのR-ID(loopbackアドレス)>

エリアについての設計（バーチャルリンク）

- バーチャルリンクはArea0に対して物理的コネクションを持たないエリアを取り込むとき
- 万が一のときのエリア0が切断されてしまう場合にバックボーンをつなぐため(patching)に使用するときや網変更の際の緊急措置対応のための使用にとどめておくべき

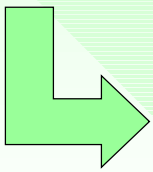
patchingの例

SW-Aが故障したときにR-Bがarea0から切断されないように、R-AとR-BでAreaZを介したバーチャルリンクを張っておく



ルータID

- loopbackアドレスがあるときはloopbackアドレス
- そうでないときは最大のIPアドレス(C社で)(RFC的には"最小のアドレスとする実装戦略が考えられる(One possible implementation strategy would be to use the smallest IP interface address belonging to the router)"となっている)
- ルータIDが変わると、link-stateしゃべり直し



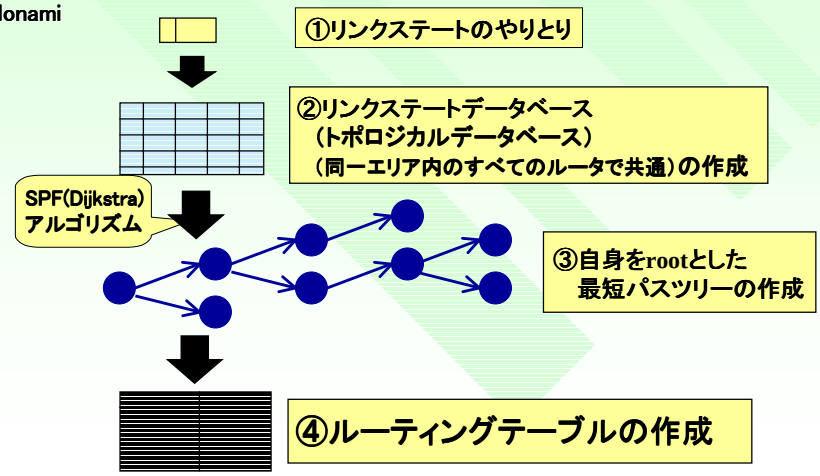
- loopbackアドレスを設定するべき
  - 絶対ダウンしない
  - 安定している
  - iBGPピアリングのためにも
  - ルータIDとしてなにかと使う
    - telnet
    - syslog, tftpのソースアドレスとして
  - /32で十分

OSPFの仕組み

～大規模ネットワークにおいてOSPFの何が響くのか～

ルーティングテーブルの作成まで

大規模ネットワークにおいてOSPFの何が響くのか理解するため、OSPFプロトコルについて知りたいなぁ



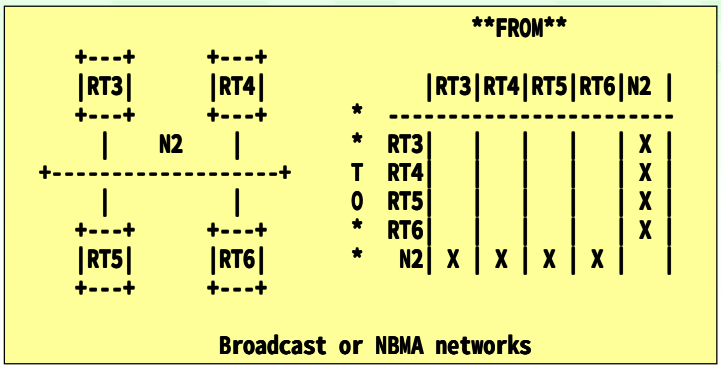
リンクステートデータベース*

* RFC1583 : The Topological Database
RFC 2178, 2328 : The Link-state Database

- 有向グラフ
- ルータとネットワークで構成される
- ルータがネットワークにインタフェースを持っているときは、ルータとネットワークをつなぐ
- 2つのルータが物理的にpoint-to-pointで結ばれているときは、ルータ同士をつなぐ

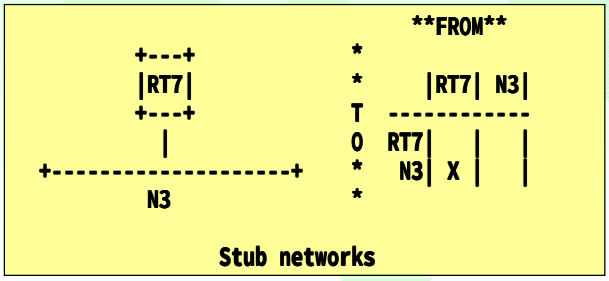
リンクステートデータベース（トランジットネットワーク）

- トランジットネットワーク
  - マルチアクセスネットワークにおいて複数のルータがあるとき
  - ルータとネットワークを双方向でつなぐ



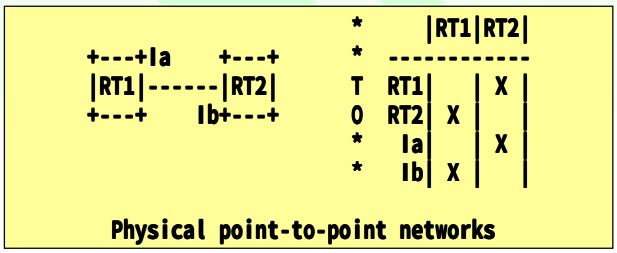
リンクステートデータベース（スタブネットワーク）

- スタブネットワーク
  - マルチアクセスネットワークにおいてルータが一つだけのとき
  - ルータからネットワークへの片方向でつなぐ



リンクステートデータベース（point-to-point）

- point-to-point
  - 2つのルータが物理的にpoint-to-pointで結ばれているときは、ルータ同士をつなぐ。双方向。
  - Numberedのときは、そのインタフェースは各ルータにstub networkでくっついているようにみなす
    - » ルータからインタフェースの片方向
  - Unnumberedのときはルータだけ

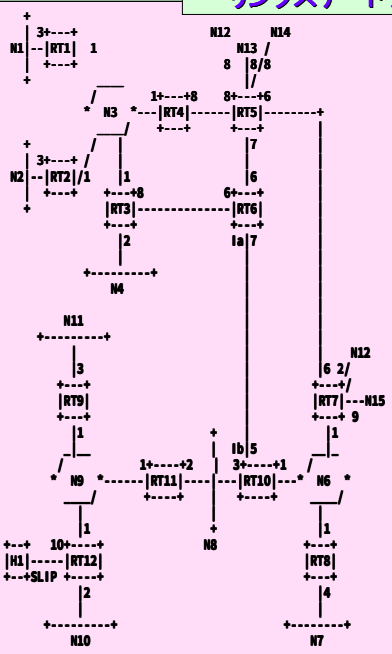


※"I"はルータインタフェースの接頭詞を示す

リンクステートデータベース

- データベースの中はコストを値とする
- コストはルータインタフェースにリンクステートが入ってくるところで効く
  - つまり、データトラフィックが出るところで効く
- ネットワークからルータに向かうところは常にコスト0
- 同一エリア内のすべてのルータで共通
  - 次ページの例はエリアが一つだけの例

リンクステートデータベースの例



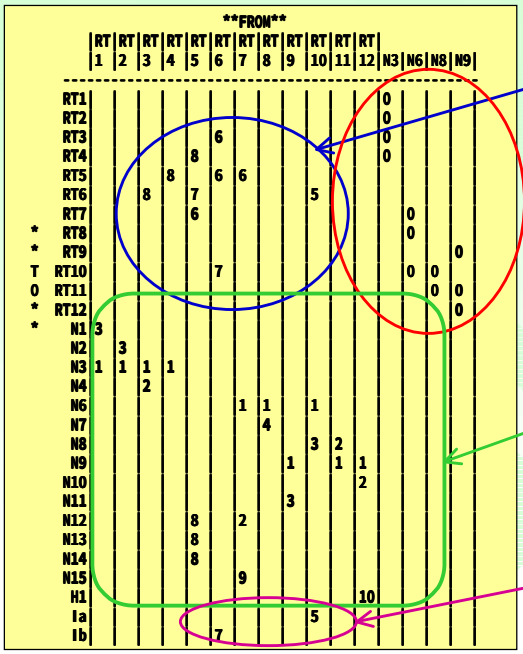
A sample Autonomous System

FROM

TO	RT1	RT2	RT3	RT4	RT5	RT6	RT7	RT8	RT9	RT10	RT11	RT12	N3	N6	N8	N9
RT1													0			
RT2													0			
RT3						6							0			
RT4					8								0			
RT5				8		6	6									
RT6			8		7					5						
RT7					6									0		
* RT8														0		
* RT9																0
T RT10						7								0	0	
O RT11															0	0
* RT12																0
* N1	3															
N2		3														
N3	1	1	1	1												
N4			2													
N6							1	1		1						
N7								4								
N8										3	2					
N9									1		1	1				
N10												2				
N11									3							
N12					8		2									
N13					8											
N14					8											
N15							9									
H1												10				
Ia										5						
Ib						7										

The resulting directed graph

リンクステートデータベースの内容



FROM

TO	RT1	RT2	RT3	RT4	RT5	RT6	RT7	RT8	RT9	RT10	RT11	RT12	N3	N6	N8	N9
RT1													0			
RT2													0			
RT3						6							0			
RT4				8									0			
RT5				8		6	6									
RT6			8		7					5						
RT7					6									0		
* RT8														0		
* RT9																0
T RT10						7								0	0	
O RT11															0	0
* RT12																0
* N1	3															
N2		3														
N3	1	1	1	1												
N4			2													
N6							1	1		1						
N7								4								
N8										3	2					
N9									1		1	1				
N10												2				
N11									3							
N12					8		2									
N13					8											
N14					8											
N15							9									
H1												10				
Ia										5						
Ib						7										

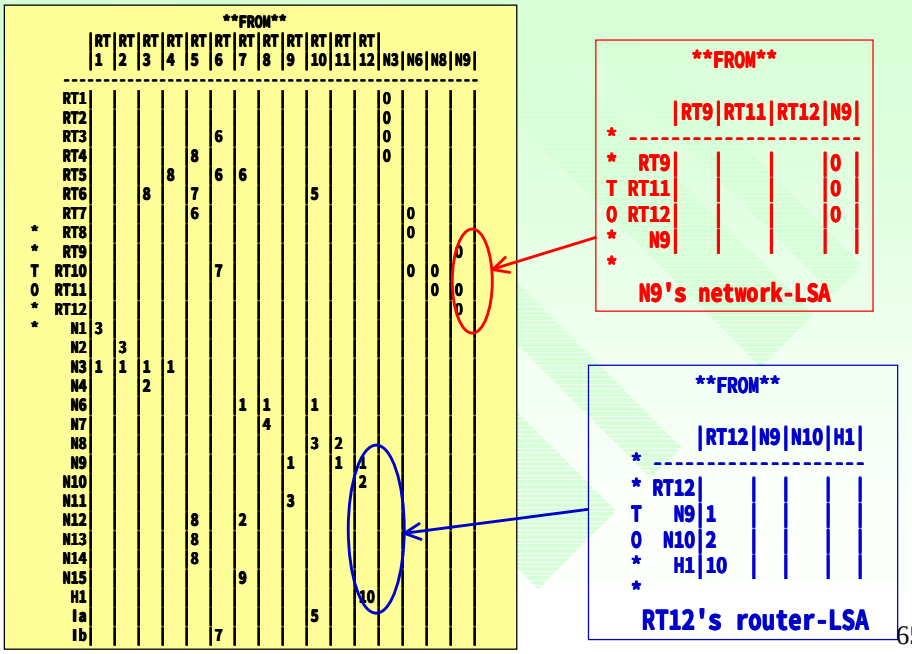
p-to-pはRT同士の辺となる

・FROMでNWがあるところは
複数のルータがあるマルチ
アクセスネットワークとなる
・NWからRTに向かうのは常に0

・RTからNやHに向かうのはその
ルータがそのネットワークに
インタフェースを持つことを意味
する
・値はそのインタフェースでのコス
トを示す

・RTからIIに向かうのは、P2Pに
おいて対抗のルータのインタフ
ェースにアドレスが割り当てら
れていることを示す

リンクステートデータベースとLSA



OSPFのパケットの種類

前ページでnetwork LSAとかrouter LSAってでてきたけど、そもそもLink-stateってどんな内容なんだろう？

Type	パケット名
1	HELLO
2	Database Description
3	Link-state Request
4	Link-state Update
5	Link-state Acknowledgment

OSPFのパケットの種類Type1～3

- HELLO(Type1)
  - neighborの検出、維持
  - DR/BDRの決定
  - すべてのルータより周期的(10sec)に送信
    - » デッドタイマー: ルータのダウン、削除時などの構成変更の発見
- Database Description(2) & Link-state Request(3)
  - ネットワークにルータが新たに参加したときに、DRとのデータベースの違いのチェックを行う
  - LS age(Link-stateの作成されてからの時間)をチェックしてどちらが最新のものを保持しているか判断
  - 自分のもっているものが古い、もしくは持っていない場合にはLink-state Requestを送信し、詳細な情報を得る

以上の動作でAdjacencyが確立される

OSPFのType5,4とLSA

- Link-state Acknowledgment(Type5)
  - Link-state Updateを受信したときの受信確認
- Link-state Update(Type4)
  - 最も重要(OSPFを理解するためには)
  - OSPFでは情報Link-stateを交換するが、それがこれ
  - ひとつのLink-state UpdateはOSPFヘッダとそれに続く複数のLink-state Advertisementでできている

Link-state Advertisementの種類

LS Type	LSAの名前
1	ルータLSA
2	ネットワークLSA
3, 4	サマリLSA
5	AS-external LSA

※LSAヘッダについてはType4 Link-state Updateのほかに、Type2のDatabase Description、Type5のLink-state Acknowledgmentの中でも使われるが、Type2,5についてはLASヘッダだけが使用され、LSAの中身は使用されない

LSAの種類Type1,2

- ルータLSA(Type1)
  - 全てのルータで生成する
  - ルータの接続情報
    - » そのルータにどういうリンクがついているか、それぞれのリンクの種類*とリンクの情報(Link ID,Link DATA)とコストを情報としてもつ
  - エリア内しか伝わらない
  - これにより、エリア内の各ルータが各ネットワークにどのように接続されているかが分かる
- ネットワークLSA(Type2)
  - DRが作成する
  - そのネットワークに接続しているルータのリスト
  - エリア内しか伝わらない

・OSPFのType4のLS-updateの話題の中でLSAの話しになって、
・LSAのType1のルータLSAの話題の中でルータについているリンクのTypeの話しになって、
・そのLink Typeの表である

*参考：ルータLSAの中で表すリンクのType

Link Type	Description
1	他のルータとp-to-p接続**
2	透過ネットワーク***への接続
3	stubネットワークへの接続
4	virtual link

** RFC2178から、p-to-pはType3でも表してよいことになっている
*** 2台以上のルータが接続されているマルチアクセスネットワーク

LSAの種類Type3～5

- サマリLSA(Type3,4)
  - エリア境界ルータによって生成される
  - エリア外の情報
  - Type 3 はエリア外のネットワークの情報
  - Type 4 はAS境界ルータの情報（AS外部のネットワークについてはType5）
- AS external LSA(Type5)
  - AS境界ルータによって生成される
  - 他のASの経路を記述
  - redistribute された経路
  - default-information originate

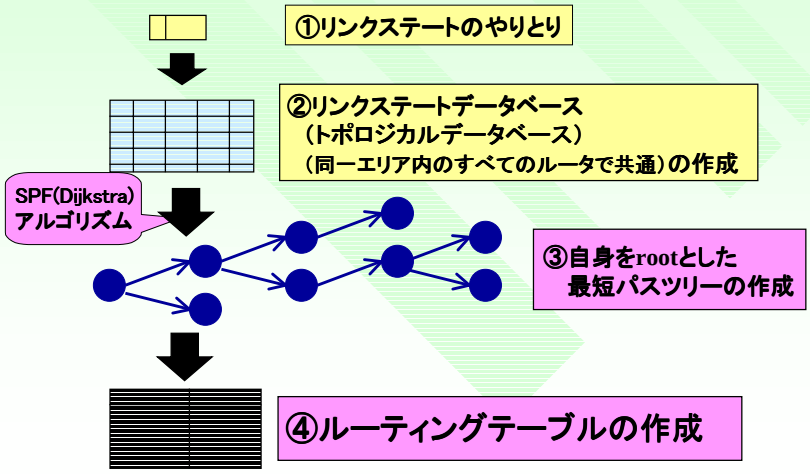
NSSA

- RFC1587 "The OSPF NSSA Option"
- 準スタブエリア(Not So Stubby Area)
- Type 7 LSAを使う
  - スタブエリアは、AS externalな経路(Type 5)は流れない。よって、スタブエリアにはAS境界ルータは置くことはできない
    →例えばstaticをredistributeするところなどでは使えない
- NSSA は上記の制限をなくす仕組み
- NSSAではType7 LSAを流すことができる
- NSSAのAS境界ルータでType7 LSAとしてredistributeすることによって、AS境界ルータを置くことができるという仕組み
  - Type 7 LSAsはNSSAのASBRでしか生成できない
  - Type 7 LSAsはNSSAの中でしか流れない
  - NSSAから他のareaに行くときは、ABRでType 7 LSAsをType 5 LSAsに変更する。そのときサマライズやフィルターすることもできる。

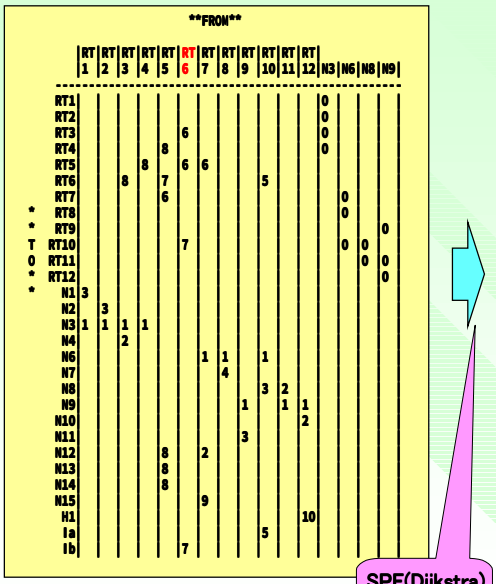
  →エッジの方でメモリの少ないルータとかある場合に使える
- C社ではIOS 11.2あたりから対応

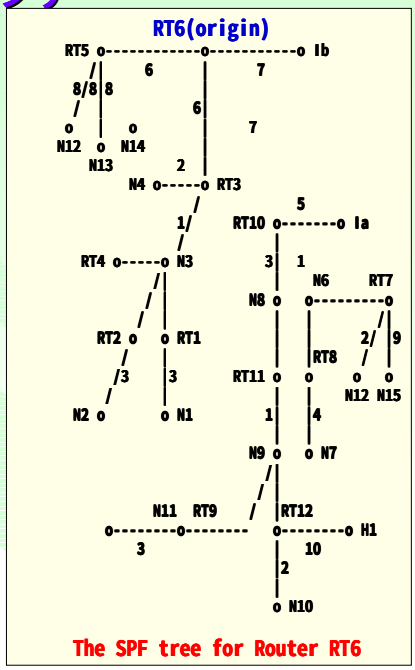
ルーティングテーブルの作成まで

ここまでで、Link-stateについてと、それをもとにどのようにしてリンクステートデータベースができるかがわかった。ではそれからどうやってルーティングテーブルができるのかなぁ？



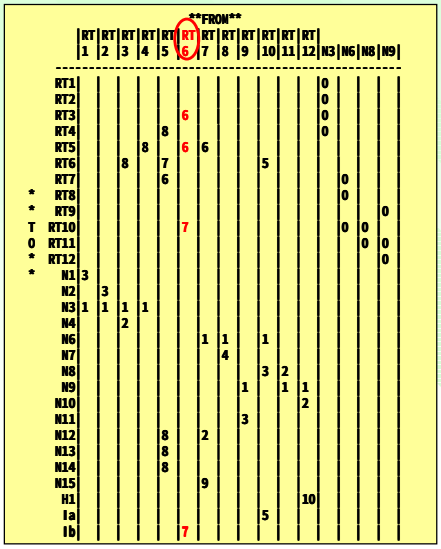
最短パスツリー





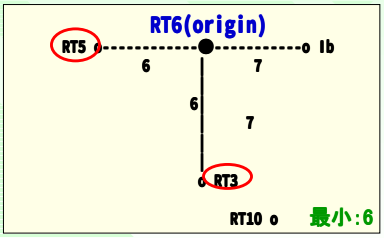
SPF(Dijkstra)アルゴリズム(1)

すべての中で最小のものを確定していき、次はそこから次のノードまでを加えていく



データベースを見て、RT6から次のノードまでのツリーを作る

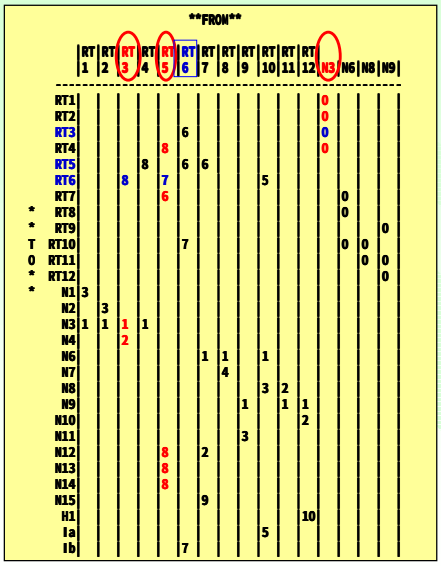
1回目



:確定

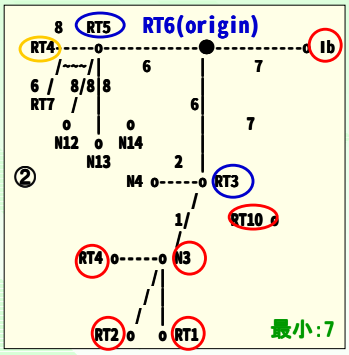
現在リーフにあるノードの中でRT6からのコストが最小である6のノードを確定する

SPF(Dijkstra)アルゴリズム(2)



確定したところからDBを見て次のノードまで伸ばす
(RT6などの既に確定しているノードは除く)

2回目

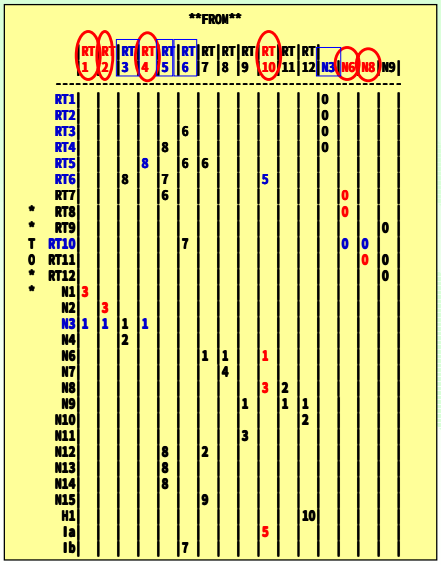


:旧確定 :新確定 :消去

現在リーフにあるノードの中でRT6からの
コストが最小である7のノードを確定する

RT4はRT6→RT3→N3→RT4で確定したので
RT5→RT4のところは消去する

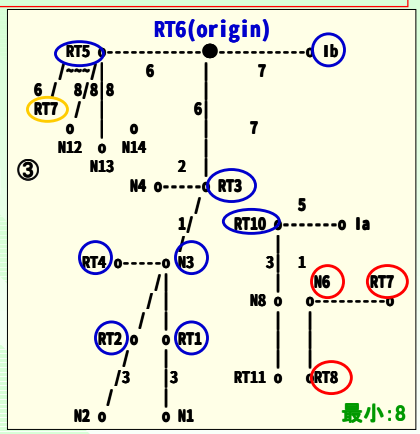
SPF(Dijkstra)アルゴリズム(3)



こういう感じで繰り返していく

確定したところからDBを見て次のノードまで伸ばす

3回目

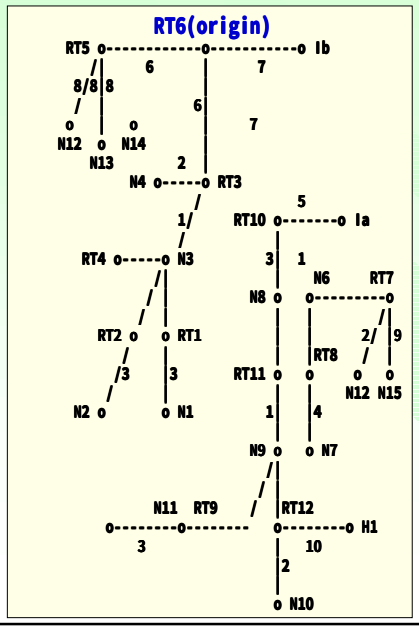


:旧確定 :新確定 :消去

現在リーフにあるノードの中でRT6からの
コストが最小である8のノードを確定する

RT7はRT6→RT10→N6→RT7で確定したので
RT5→RT7のところは消去する

ルーティングテーブルの作成

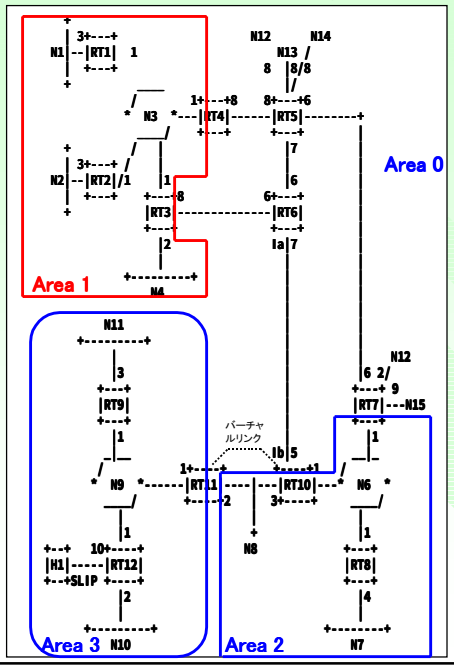


- 最短パスツリーからルーティングテーブルが作成される

Destination	Next Hop	Distance
N1	RT3	10
N2	RT3	10
N3	RT3	7
N4	RT3	8
Ib	*	7
Ia	RT10	12
N6	RT10	8
N7	RT10	12
N8	RT10	10
N9	RT10	11
N10	RT10	13
N11	RT10	14
H1	RT10	21
RT5	RT5	6
RT7	RT10	8

The portion of Router RT6's routing table listing local destinations.

エリアで分けられている場合



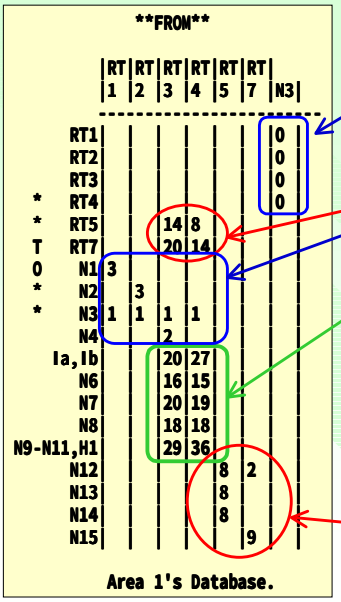
Area 1

FROM

	RT1	RT2	RT3	RT4	RT5	RT7	N3
RT1							0
RT2							0
RT3							0
* RT4							0
* RT5			14	8			
T RT7			20	14			
0 N1	3						
* N2		3					
* N3	1	1	1	1			
N4			2				
Ia,Ib			20	27			
N6			16	15			
N7			20	19			
N8			18	18			
N9-N11,H1			29	36			
N12					8	2	
N13					8		
N14					8		
N15						9	

Area 1's Database.

リンクステートデータベースの内容



LS Type2: ネットワークLSA

エリア境界ルータから
AS境界ルータまで
LS Type4: サマリ LSA

LS Type1:
ルータLSA

エリア境界ルータからエリア外の
ネットワーク(inter-area)まで
LS Type3: サマリ LSA

これは以下の情報からわかる。
・エリア境界ルータ(RT3、RT4)から全部のエリア境界ルータ(RT7、RT10など)までのコストがバックボーンエリアのSPF treeから計算される。
・各エリアのエリア境界ルータからバックボーンにサマリ情報を流している。
これがつまりエリア境界ルータがバックボーンに属していなければならない理由でもある。

AS境界ルータから
AS externalなネットワークまで
Type5: AS external LSA

SPF(Dijkstra)アルゴリズムの負荷

- リーフにあるノードは候補リストに入っていて、コストの低い順に並べてある
- 最もコストの低いノードを確定して、そこから新たなリーフを継ぎ足していく
- その新たなリーフを候補リストのしかるべき位置に入れるのは現在の候補リストにのっているノード数を m とすると O(log(m))となる
- すべてのリンクは必ず1度づつ調べられている
- よって、そのエリアの全リンクの数を l とすると
- O(l * log(m))
- となる。よって、エリア内のノードの数だけではなく、リンクの数、つまりネットワーク構成によって大きく左右される
- 同じノード数でもリンクの数が多いような構成だときつい

OSPFの負荷について

- OSPFでネックになるのはSPFアルゴリズムだけではない
- むしろLink-stateの交換がかなりの負荷がかかっているように見えるときもある
  - 安定しないネットワークではなかなかadjacencyも確立しない
  - sh ip ospf neighbor で見ても、DRやBDRともなかなかFULLにならない
    - » Exchange → Init
- メモリが足りないから不安定になっているわけではない、ということがよくある
- インプリマターだし、はっきりしたことは誰にもわからない

大規模ネットワークにおけるOSPF設計

- どのくらいの大きさまでOSPFが耐えられるかは、ルータの機種・メモリ、ネットワークの構成、安定度などによるので一概に言えない
- また、検証も困難
  - それだけの台数を集めるのは難しい
- したがって基本的に経験則となる
- また以下のような著名な人のドキュメントも参考になる
  - OSPF Anatom of an Internet Routing Protocol
    - » J. Moy
      - RFC 著者
    - » January 1998
  - OSPF DESIGN GUIDE
    - » Bassam Halabi -Cisco Systems Network Consulting Engineer
      - (”インターネットルーティングアーキテクチャ“の著者)
    - » April 1996
    - » http://www.cisco.com/warp/public/104/1.html

大規模ネットワークにおけるOSPF設計Tips

- 一つのAreaに持てる台数
  - よくある質問で、その度に「一概に言えない」というのが決まり文句だが…
  - C7513 RSP4 256Mとかで100台くらいは十分安定していけそう？
  - ただ、今までの説明の通り、かなりネットワーク構成によって左右される
  - 実際は増やしていって、例えばどこかのリンクをシャットダウンしたときとかに、増やす前に比べてコンバージェンス時間（CPUが落ち着くまでの時間）が明らかに大きくなるとそろそろ限界だと思うべき
    - » これはわかります
  - トラヒックが非常にかかっていて、ただでさえ負荷の重いルータに注意する
    - » こういうルータは大事なルータでもある。最も注意すべき。
  - 性能の低いルータが入っているだろうから、それも注意する必要がある
  - Halabi：50台まで。60台とか70台は避けた方がいい
  - Moy：1991年に多くて200台と言ったが、ベンダによって350というところもあれば、50とかそれ以下とかいうところもある。ただ、あまり少なくしすぎないべきだ。

大規模ネットワークにおけるOSPF設計まとめ

- リンク数
  - あまりリンクを持つような構成はよくない
  - 例：p2pでフルメッシュにするよりも、マルチアクセスのSWにする
- メモリ
  - メモリが足りているといって安心してはいけない
  - しかし、メモリが多いに越したことはない
  - OSPFのルートマップが占有するメモリ容量は，1エントリ当たり200～300B。オーバーヘッドは，1LSA当たり100B
    - » 5万経路で15M+α Byteとなってメモリは足りているのだが…
- DR/BDR
  - DRは結構（かなり）負荷がかかるので、処理能力のあるルータや、他の処理が重くかかっていないものになるなど、考慮する必要がある
  - 一つのルータが複数のネットワークのDRにならないように考慮する必要がある
    - » ip ospf priority
- loopbackアドレス
  - 安定したルータIDのためにloopbackアドレスを持つようにする

大規模ネットワークにおけるOSPF設計まとめ

- エリア
  - area 0 を中心としてそこから拡大していくようにする
  - リダンダンシーのため、一つのエリアでは複数のエリア境界ルータを置くべき
  - エリア境界ルータがもつエリアの数はなるべく2つまでにする
  - virtual linkをあてにして設計しないようにする
- 経路数
  - なるべく経路が集約できるようにIPアドレスの設計をする
- デフォルトルート
  - デフォルトルートをうまく使う
    - » default-information originate
  - 多くの経路をOSPFにredistributeはしない
    - » あまり負荷に関係なさそうなAS externalの経路でさえも、多くなるとメモリが足りているにもかかわらず不安定になる
- まずは、ちゃんとポリシーを策定するのが基本

危なくなったときどうするか？

- 機器の性能をアップグレードする
  - 劇的に変わることが多い（例：C7513 RSP2→RSP4）
- ノード数とリンク数を少なくするため大容量ルータにする
  - バックボーンエリア内の台数の削減
  - それでもどんどん大きくなっていく…
- それができない、またはそれでも間に合わないなら工夫すればよい
  - 状況に応じて手を打つ
  - confederation
  - static-to-bgp
  - 他の候補
    - » エリア境界ルータにもっと多くのエリア
    - » OSPFプロセス分け
    - » IS-IS化
    - » virtual link
    - » ネットワーク分けて他のプロトコルで結ぶ
    - » etc...

IS-IS

すみませんが、IS-ISのプレゼンは
時間の都合上省略させていただきます。
しかし一部要望があるため、参考資料ということで
添付しておきます。宜しくお願い致します。

IS-IS

- 米国のビッグISPでOSPFではなく、IS-ISを使っているところが結構ある
- OSPFよりスケールするという噂もある
  - 本質的にはOSPFと大差ないはずなので、実装がしっかりしているのだろう
- 米国ISP向けにチューニングされているらしい
- 日本のISPや企業で使っているところはないだろう

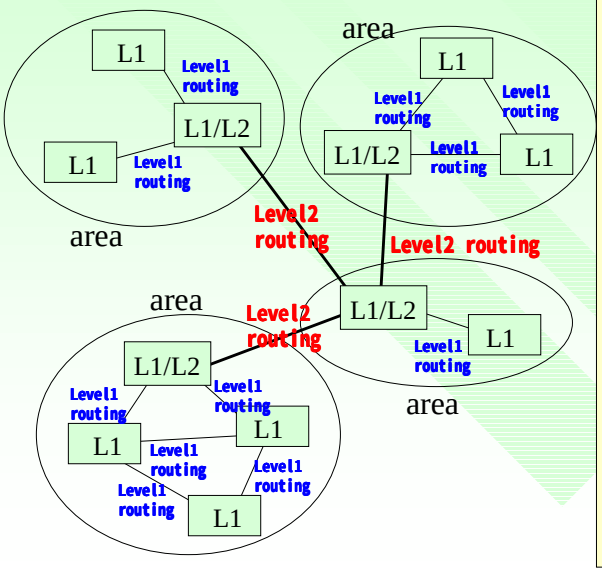
とりあえず、どんなものか見てみよう

IS-ISの特徴

注意：この資料でIS-ISと言っているのは、正確には「Integrated IS-IS」のことである。

Link state routing protocol	OSIスタックのLinks State Routing Protocol - OSPFに非常によく似ている - DRの仕組みも存在する
Level1 and Level2	2つの階層をもつ
Cost-based routing protocol	1linkでMetric 0～63 default値は10（すべてのIF） 積算されたmetricの最大値：1023
NSAP address	使用するアドレスNSAP Address
その他	VLSM対応 OSI CLNS と TCP/IPネットワークをサポート ロードバランスはCiscoでは6pathまで

ネットワーク構成例



- areaの概念がある
- areaのボーダーはOSPFと異なり、リンクになる
- area内のroutingをlevel1 routingという
- area間のroutingをlevel2 routingという
- level1 routingを話すルータをlevel1(L1)ルータという。Level1 ISともいう。
- leve2 routingを話すルータをlevel2(L2)ルータという。Level2 ISともいう。
- level1 routing、level2 routingどちらも話すルータをlevel1/level2(L1/L2)ルータという
- 多くのL2ルータは、自areaのためlevel1 routingを話すため、L1/L2ルータとなる

用語の簡単な説明

- CLNS(Connectionless Network Service)
  - OSIのものだが、いわば「IPの世界でのアドレスや伝送の仕組み」というのと同じような感じで「OSIの世界でのアドレスや伝送の仕組み」ということ
- NSAP(Network Service Access Point)address
  - CLNSで使うアドレス

プロトコルスタック	TCP/IP	OSI
アドレスや伝送の仕組み	IP	CLNS
アドレス	IPアドレス	NSAPアドレス

IS-IS Routing Protocolの仕組み

- IS-ISのLSP(Link State PDU)はOSIのノード間のやり取りとして認識される
  - IS-ISのやり取りは、OSIのネットワークレイヤ即ちCLNSで行われる。
  - よって、各ルータでは、OSIでのアドレスすなわちNSAPアドレスで表現されるNETを持つ必要がある。NETはOSPFでいうルータIDにあたる。
  - IPはIS-ISのLSPに乗る情報としてやり取りされる。
- つまり、
  1. CLNSにおいてIS-ISのやり取りをし、データベースができる
  2. NETに基づいたツリーを作る
  3. IP(及びCLNS)のルーティングテーブルを作る
- OSPFとIS-ISの比較

ルーティングプロトコル	OSPF	IS-IS
使用するネットワークレイヤ	IP	CLNS
ノードのID	ルータID (IPアドレスに基づく)	NET (NSAPアドレスに基づく)
できるルーティングテーブル	IP	IP及びCLNS

Level1 and Level2 Routing

- Dijkstra'sアルゴリズム
  - Level1とLevel2両方それぞれに関して独立に走る
- Level1 IS ルータにおいて
  - エリア内への通信に関しては、Level1 IS-ISで認識し、普通にrouting tableにのっけることによって通信が可能となる
  - 他エリアへの通信に関しては、metric的に最も近いL1/L2ルータに向けてdefault routeを向けることによって通信が可能となる。
    - » routing tableにそこに向けて 0.0.0.0/0 が生成されるわけ。
    - » L1/L2ルータからL1へのLSPのATT(Attached) bitを1にすることによって、知らされる。
- Level1/Level2 IS ルータにおいて
  - 他エリアへの通信に関しては、Level2 IS-ISで認識
  - 自エリアへの通信に関しては、Level1 IS-ISで認識

NSAP address

- NSAP address

Example: 47.0004.004D.0003.0000.0C00.62E6.00

IS-IS area address (可変長:1～13byte)

System address (=System ID + セレクタ) (固定長:7byte)

- NET
  - System IDは自由に振ることができるが、一般的に次のような形で割り当てられることが多い
  - MACアドレスを割り当てる
    - » system IDはセレクタ抜かして6bytesのため、ぴったり
  - loopbackのIP addressを割り当てる
    - » 6bytesを16進数表記すると数字が12個になる。その12個の数字を、3桁の10進数表記4つに当てる。

例:loopbackのIP addressが192.168.10.1の場合
→system IDを 1921.6801.0001 にする。

192.168.10. 1

Config例

```
clns routing
!
interface loopback0
 ip address 10.1.0.2 255.255.255.255
 ip router isis ****
...
!
interface serial0
 description isis level-1 connection
 ip address 10.1.2.1 255.255.255.0
 ip router isis ****                 →IPアドレスの情報をこのIFでやり取りする
                                       ＋ このIFのNWを広告する
                                           (OSPFのnetworkコマンドと同様だろう)
 clns router isis ****               →CLNSアドレスの情報をこのIFでやり取りする
                                       (IP情報で十分のときは必要なし)
 isis circuit-type level-1           →このリンクでLevel 1 routingだけ話す場合
!
router isis ****
 redistribute static metric 0        →staticユーザ収容ルータにおいて
 net 47.0000.0100.0100.0002.00       →loopback IPアドレス10.1.0.2とsystemIDが対応
 is-type level-1                     →level 1 ルータとする場合
```


基本config

・基本的なコンフィグ

1) ISISプロセスをあげる

```
router isis
 net xx.xxxx.xxxx.xxxx.xxxx.00
```

2) インタフェースにISISをしゃべらす。
そのインタフェースのNWも広告する。

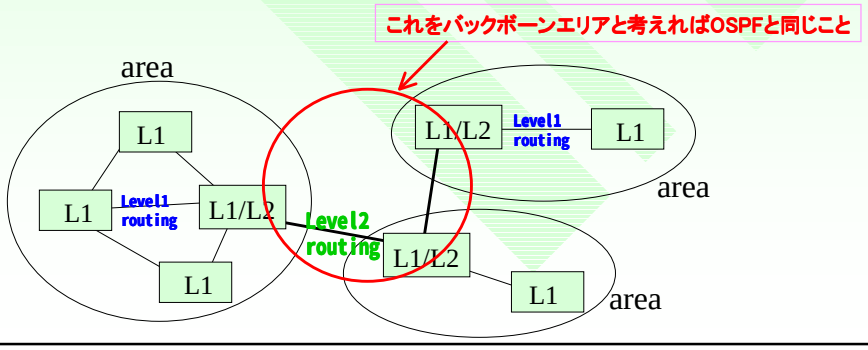
```
int xxx
 ip router isis
```

以上が最小限のコマンド

この場合、このルータはLevel-1/2ルータとして動く。
ルータ自体やリンクをLevel-1やLevel-2-onlyにするコマンド、
メトリックを設定するコマンドなどなどがある

IS-ISとOSPFとの本質的な違い

- SPFをダイクストラアルゴリズムで作るときに、OSPFではルータIDを元に計算するが、その代わりにNSAPで表現されるNETでやる（これは本質的な違いではない）
- ISISエリア境界がルータとルータの間にあるが、Level-1/2ルータをOSPFの場合のエリア境界ルータと思うと、本質的な差はない



OSPF と IS-ISの比較

- 米国ISPで昔IS-ISに使っていて慣れているので今も使っている、という理由でIS-ISを使っているところもある
- 日本でIS-ISに慣れている人なんていない
  - 教えてくれる人も、サポートしてくれる人もいない
- 本もドキュメントもあまりない
- CLNSも使いたい人にはうれしいがそんな人はいない
- いまさらIS-ISには変更できない
  - オペレーション的にも、ノウハウ的にも、ネットワーク的にも

- 普通、これらのデメリットに逆らってまでOSPFでなくIS-ISにすることはほとんどない
- しかし、OSPFでどうしようもなくなった場合は、ルータ製品的にIS-ISのほうがOSPFよりスケールしやすい実装になっているということを信じてIS-ISを試してみるのも一案ではある
- ただし、通常、上記のデメリットがでかく、また、工夫すればOSPFでなんとかなるケースが多い

(2) BGPのシステム設計論

概要

- 関連事項の整理
- BGP・プロトコル概説
- ISPネットワーク拡大に沿った規模対応
- ポリシルーティング
- ポリシルーティングの実際

関連事項の整理

AS(Autonomous System)とIGP, EGP

- AS==単一のルーティングポリシで運用される範囲
  - 一般的にはひとつのISP。Routing Domain とも呼ばれる
  - 16ビット(1～65535)の番号空間を持つ
    - » AS2914== *NTT Verio*, AS5511==FT/*Opentransit Internet*
    - » 64512～65535はプライベートAS
    - » 現在最大値は22000程度，11,000個程度が観測される
  - The Internet とは、ASが相互接続された全体
- AS境界を基準に経路制御プロトコルが異なる
  - IGP: Interior Gateway Protocol – OSPF, IS-IS, RIP
    - » AS内(Intra-AS)の経路制御に用いる
  - EGP: Exterior Gateway Protocol – BGP, IDRP
    - » AS間(Inter-AS)の経路制御に用いる

CIDRの復習(1)

- CIDR – Classless Inter-Domain Routing
- クラスレスなAS間の経路制御
  - クラスレスとは、、
    - » classA, classB, classCなどのクラスの考え方を除いたもの
    - » 対義語==クラスフル(classful)

CIDRの復習(2)

- クラスフル(classful) という考え方
  - IPアドレスの先頭オクテットの値でネットワークアドレスの範囲を判断する
    - » class A = 1～126— 第一オクテットだけがネットワーク
    - » class B = 128～191— 第二オクテットまでネットワーク
    - » class C = 192～223— 第三オクテットまでネットワーク
  - ネットワークアドレス単位でしか扱わない（扱えない，伝えない，伝えるすべがない）
  - その中を更に分割したものをサブネットと言う
    - » 分割する大きさも自分にしか定義できず、伝えるすべがない
    - » クラスフルネットワークの中は統一したサブネットのサイズにしないと扱えない

CIDRの復習(3)

- クラスレス(classless)という考え方
  - どこまでがネットワークを示すのかを明示して扱う
  - ネットワークを示すものをプリフィクス(Prefix)と呼ぶ
  - プリフィクスの長さは一般的にビット数で表される
    - » Class Cの 202.216.40.0 – 202.216.40/24 (202.216.40.0/24)
- つまりクラスレスだと、
  - 連続するclass Cアドレスを任意の大きさでひとかたまりで扱える
  - Class Aのサブネットも全く同様に扱える
  - Class Cより小さいアドレスブロックも全く同様に、任意の大きさで扱える
    - » これがいわゆるVLSM(Variable Length Subnet Mask)

CIDRの復習(4)

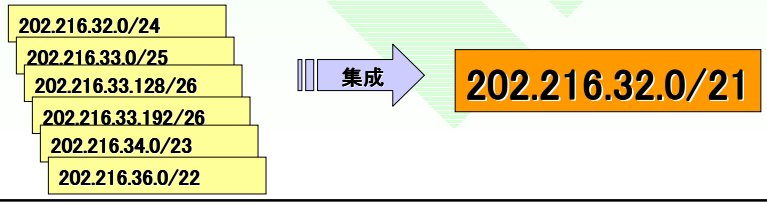
- CIDR—クラスレスなAS間経路制御
  - プリフィクス＋プリフィクス長で経路情報を扱う
  - 複数のClassC(=/24)アドレスも（あらゆるアドレスが）、任意の大きさでひとかたまりに扱える
  - AS内の小さなネットワークセグメント，ユーザネットワークをひとかたまりにして他のASに広告できる
    - » 経路集成—aggregation

The Internetにおける階層的経路制御(1)

- 全インターネットを3つに階層化して、それぞれ独立して経路制御を扱う
  - InterAS
    - » AS間，Default-Freeゾーン，EGPで制御
  - IntraAS
    - » AS内，AS内の全経路，IGPで制御
  - End-User
    - » ユーザサイト内。StaticやIGPで制御

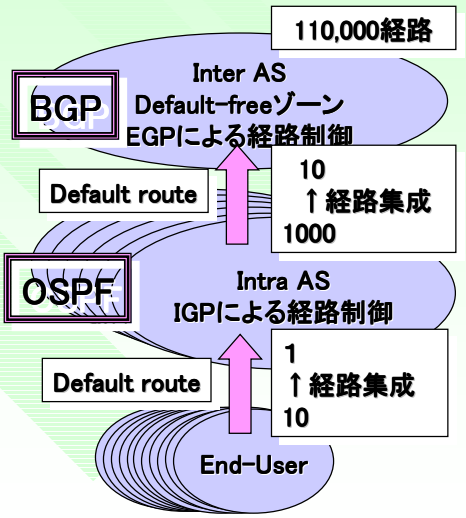
The Internetにおける階層的経路制御(2)

- 経路集成 – Aggregation
  - 複数の経路情報をひとかたまりにして、より大きなサイズの（より短いプリフィクスの）単一の経路情報にすること
  - 現在IPアドレスの割り振りはISP毎に行われているので、そこからユーザに割り当てるIPアドレスは割り振りブロックで集成することができる。



The Internetにおける階層的経路制御(3)

- それぞれの境界で経路集成=情報量の縮退
- 上流の経路は全てdefault route で制御する
- 下流の詳細構成は気にせず、ひとかたまりの経路で制御する



The Internetにおける階層的経路制御(4)

- その内在的矛盾？
  - CIDRは非階層的アドレス形態であったIPアドレスに階層構造を持ち込んだ
  - 階層構造を厳格に推し進めようとすると…
    - » 電話番号のように局番固定割り当てのような構造が望ましい
  - 末端に近くなるほどマルチホームがしにくい
    - » 小さいアドレスブロックでマルチホームをするのは難しい
  - 実際問題としては、小さいアドレスブロックでマルチホームすることも容認されつつある
    - » 階層的経路制御の崩壊の兆し。。。。

BGP・プロトコル概説

基本事項の確認

- 現在のバージョンは4 – BGP4, RFC1771
- AS間経路制御に用いる
- ピアリング(peering) – 明示的に定義した隣接ルータとの間にTCP上でセッションを確立し、経路情報を交換する
- パスベクター型プロトコルと呼ばれ、プリフィクス単位の経路情報レコードにはパス属性と呼ばれる属性情報が添えられている。
- パス属性により経路の優先制御を行う。
- パス属性を調整することで経路制御ポリシを実装することができる。

BGPとOSPFの比較(1)

OSPF	BGP
IGP : Interior Gateway Protocol	EGP : Exterior Gateway Protocol
IP上に直接乗るプロトコル Protocol number: 89	TCP上に乗るプロトコル Port number: 179
リンクステート型プロトコル リンクステート情報を伝播	パスベクター型プロトコル パス情報を伝播
状態変更毎にLSA，連鎖伝播	状態変更毎にUPDATE，連鎖伝播

BGPとOSPFの比較(2)

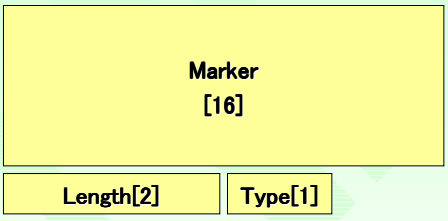
OSPF	BGP
基本的に、OSPFを起動した隣接ルータ全てと経路交換 マルチキャストでセグメント上の全OSPFルータとやりとり	明示的に定義した隣接ルータのみと経路交換 隣接ルータ毎にBGPセッションを確立（ピアリング）
あるネットワーク（ルータ）の状態変更は、全ルータのパスツリー再作成を引き起こす 30分でリフレッシュ--flooding	あるネットワークの状態変化は基本的にはそのプリフィクスだけの問題 リフレッシュなし

BGPとOSPFの比較(3)

OSPF	BGP
トポロジの管理に主眼を置く	プリフィクス（ネットワーク）の パス属性に着目
エリア内共通のLSDBを全ルータが作成し、LSDBから各ルータそれぞれがパスツリーを作成	受領したUPDATEは 各AS，ルータのポリシに 基づいて処理，以遠伝播する
経路個別のポリシ付加は不可	経路個別にポリシ付加が可能 →パス属性値として プリフィクスに付加
精密で敏速な経路制御	ポリシに基づいた経路制御

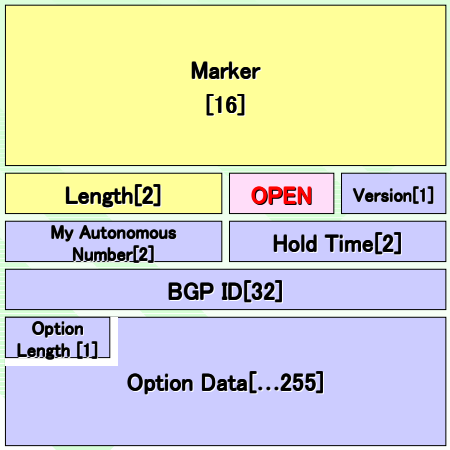
メッセージヘッダ

- マーカー(Marker)
  - セキュリティ目的に利用
- 長さ(length)
- タイプ(Type)
  - メッセージタイプ
    - » オープン(OPEN)
    - » 更新(UPDATE)
    - » 通知(NOTIFICATION)
    - » キープアライブ(KEEPALIVE)



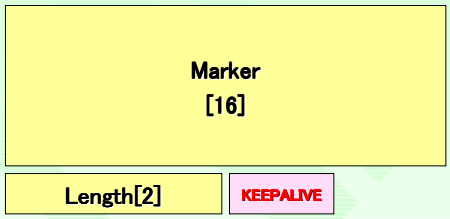
OPENメッセージ

- BGPセッション開設に利用
- 以下のパラメータ提示してネゴシエーションを実施
  - バージョン
    - » 現在はバージョン4
  - 自AS番号
  - ホールドタイム
    - » キープアライブの間隔を指定
  - BGP ID
    - » 自分が持つIPアドレスからひとつをIDとして利用する
  - オプションは現在のところ認証情報のみが定義されている
- 受諾にはKEEPALIVE, 拒否にはNOTIFICATIONを返す



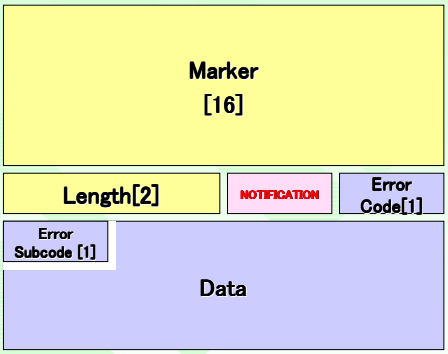
KEEPALIVEメッセージ

- セッションの正常性確認に利用
- UPDATEが一定期間以上発生しない場合に送信
  - Hold Time より頻繁に



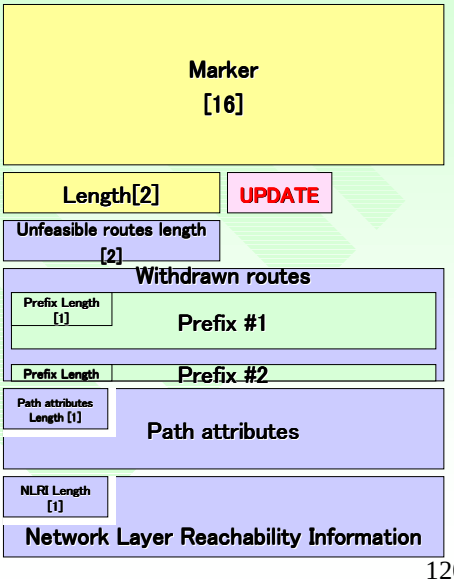
NOTIFICATIONメッセージ

- エラー通知し、セッションを終了する
  - フォーマット不良
  - 無効値
  - 状態遷移エラー
  - Hold Time 満了
  - セッション中止



UPDATEメッセージ

- 経路情報の交換に利用
- 取り消される(withdrawn)プリフィクスを複数、及びパス属性と到達可能プリフィクス(NLRI)の組一対を伝達可能
  - 経路情報を消すのは一気にできる
  - パス属性の変更や新しい経路情報は1プリフィクスずつUPDATEを送信



Path Attributes(パス属性)

- プリフィクスに括りつけられた経路選択制御用の属性値群
- 必須，任意，透過性，非透過性の4つに分類
  - 必須 – Well-known mandatory
    - » 全てのBGPルータで解釈可能で、全ての経路レコードに必要
  - 任意 – Well-known discretionary
    - » 全てのBGPルータで解釈可能で、必ずしもつけなくても良い
  - 透過性 – Optional transitive
    - » 一部のBGPルータで解釈されない可能性があり、次のASへも伝播される
  - 非透過性 – Optional non-transitive
    - » 一部のBGPルータで解釈されない可能性があり、次のASへ伝播されない

Path Attributes(パス属性)
Well-known mandatory

- ORIGIN
  - 生成元のASでどういう形でBGP上に生成されたか
    - » IGP, EGP, INCOMPLETE の3値
- AS_PATH
  - 生成元ASまでの経過ASのリスト
- NEXT_HOP
  - そのプリフィクスへの次のホップとなるIPアドレス

Path Attributes（パス属性）
ポリシ制御のプレイヤーたち

- LOCAL_PREF – 任意
  - Local Preference
  - AS内で他ASから受け取った経路に関する優先度をつけるのに用いる
- MULTI_EXIT_DISC – 非透過性
  - Multi Exit Discriminator
  - 複数相互接続点を持つ隣接ASに対してそれぞれの優先度を伝える
- COMMUNITY – 透過性
  - 任意の32ビットの情報を伝達する

eBGPとiBGP

- eBGP – External BGP
  - 他のASとの間でセッションを張り経路情報の交換を行う
- iBGP – Internal BGP
  - 同じASの複数のBGPルータの間で、それぞれがeBGPを介して入手した（あるいは自AS内から生成した）外部経路を交換し、AS内の経路情報の同期を取る
  - 基本的には、iBGPで入手した経路情報はiBGPで以遠伝播しない
    - » 全てのBGPルータとiBGPセッションを確立する必要がある（回避方法は後ほど）

ISPネットワーク拡大に沿った規模対応設計

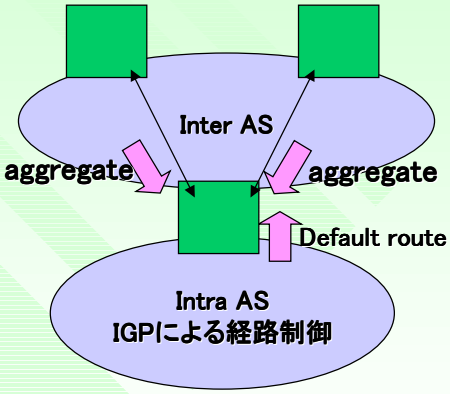
BGPの導入

AS番号はどうやって割り当てを受けるのか

- JPNICが割り当てを行う
  - ftp://ftp.nic.ad.jp/jpnic/ipaddress/as-application.txt
    - » 現在、正式サービス化に向けて作業中
- AS割り当ての条件
  - RFC1930
    - » 日本語訳も一応ある
    - » ftp://ftp.nic.ad.jp/jpnic/ipaddress/rfc1930-jp.txt
    - » マルチホーム接続（IX接続を含む）となっていることが条件

最も単純なBGPの導入

- IGPでデフォルトルートが指されるルータが単一のボーダルータ
- BGP→AS→独自の経路制御ポリシだから、2つ以上のASに接続



問題点:
single point of failure
複数箇所で他のASと接続したい

BGP導入の実際

- 2つ以上の国内大手ISPを上流としてマルチホーム接続
- NSPIXP, JPIX, JPNAP, PihanaIXなどのインターネットエクスチェンジに加入して、国内到達性を確保。別途国際ゲートウェイISP（あるいは国内大手ISP）に加入して海外到達性を確保
  - アドレスブロックは、JPNICなどから割り当てをうける

BGPの基本的コンフィグレーション(1)

```
router bgp 20000
 no synchronization
 no auto-summary
 network 172.16.0.0
 network 192.0.1.0
 aggregate-address 223.224.0.0 255.255.0.0 summary-only
 neighbor 202.249.2.60 remote-as 4689
 neighbor 202.249.2.60 route-map AS4689-in in
 neighbor 202.249.2.60 route-map ixp-out out
```

BGP起動

BGPグローバルコマンド

IGPで経路があればBGPで広告

含まれるプリフィクスがIGPにあれば集成経路を広告

集成経路以外を抑制

Peer確立

Route-mapでポリシを記述

BGPの基本的コンフィグレーション(2)

- Inbound方向のルートマップの例

```
route-map AS4689-in permit 10
 match as-path 10
 set local-preference 110
!
route-map AS4689-in permit 20
 match as-path 20
 set local-preference 100
!
```

シーケンス番号順に適用

それぞれのシーケンスで適合条件とアクションを定義

BGPの
基本的コンフィグレーション(3)

- Outbound方向のルートマップの例

```
route-map ixp-out permit 10
 match as-path 30
 set metric 1000
!
```


ISPネットワーク拡大に沿った
規模対応設計

iBGPシステムの構築

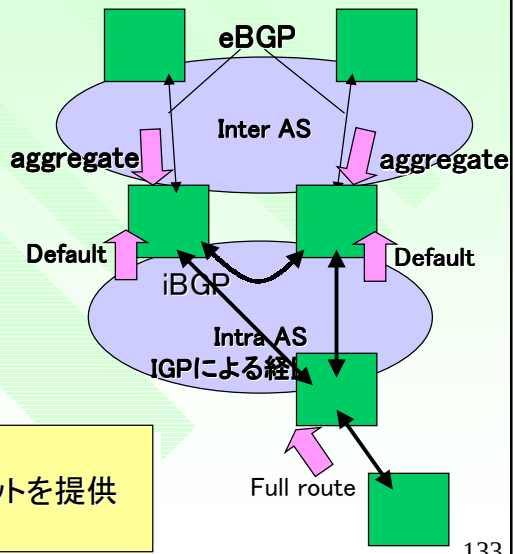
2つのボーダルータを置く

- デフォルトが2つ
  - IGP的に近いほうを選択する
- ボーダルータ間の経路情報の同期？

↓

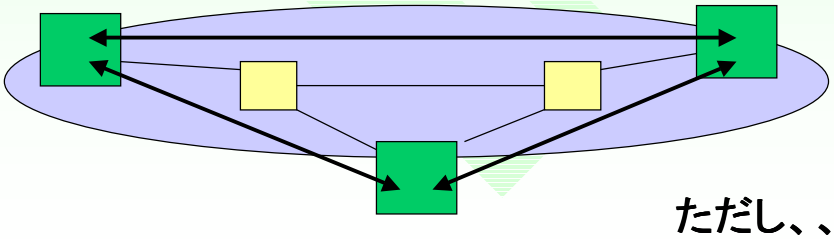
iBGPの確立

次の課題：
BGP加入者にトランジットを提供



iBGPの注意点

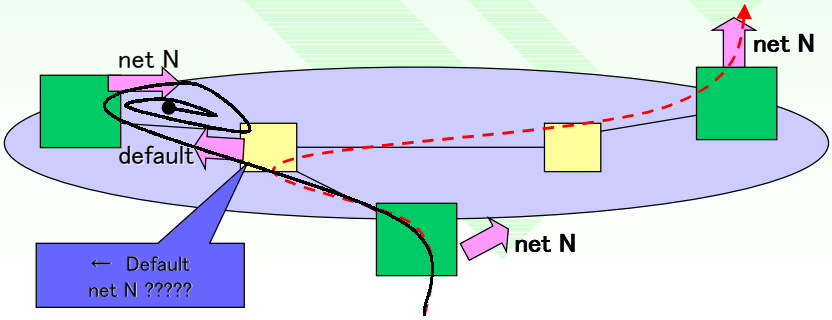
- eBGPは直接隣接を必要とするが、iBGPはAS内での同期が目的なので離れていても確立可能
- iBGPは全てのボーダルータとセッションを張る必要がある



ただし、、

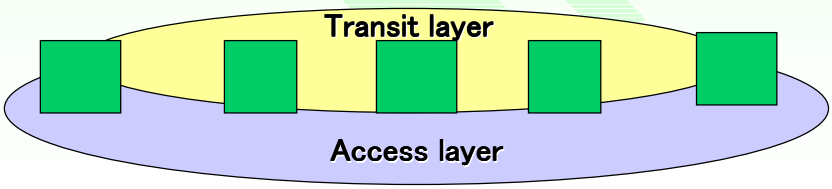
iBGP・仕様上の問題点

- Synchronization問題
  - トランジットしようとする経路はIGPで観測されていなければならない
    - Next-hopが別のボーダルータだった場合
    - 途中のIGPノードではdefaultしか知らない



iBGPシステムの解

- No synchronization
  - IGP synchronizationの縛りを解くコマンド(c社)
  - IGPで経路観測されない経路も利用可能
    - » つまり、BGPルータ間に非BGPルータがあると矛盾が発生
- トランジット層の総BGPノード化
  - トランジット層とアクセス層の二層構造へ
  - BGPユーザが多い場合、「総トランジット層」に近づく



iBGP問題のまとめ

- iBGPは隣接していなくても確立可能
- 仕様では、中間ノードが経路制御できないと問題があるので、IGPでBGP経路を知っている必要があった
- がしかし、それでは経路制御階層化の意味がないので、IGPとの同期を外すほうがよい
- IGP同期を外す結果、全てのBGPルータは隣接する必要がある
- BGPルータ（トランジット）層と非BGPルータ（アクセス）層の二層に階層化
- 総トランジット層へ

iBGPシステムの基本(1) NEXT_HOPをIGPで観測する

- iBGPで伝播される外部経路では、基本的にNEXT_HOPの値は変わらない
  - eBGPの隣接ルータのIPアドレス
- BGP経路は、NEXT_HOPがIGPでreachableでなければ有効とならない。 そこで、、
  - IXやプライベートピアリングのセグメントをIGPで認識させる
    - » 例えばpassive-interfaceでOSPFプロセスに定義する
  - eBGPルータで、iBGPピアに対してnexthop-selfを設定して、自分のIPアドレスをNEXT_HOPとして使う

iBGPシステムの基本(2)
loopbackをピア設定に利用する

- iBGPピアの設定では、loopbackアドレスを利用するのが「基本」
  - 1oopbackインターフェースはダウンしない
    - » 隣接ルータと対面するインターフェースが落ちても迂回して到達することが可能
    - » Loopbackインターフェースにも IGPを起動することを忘れずに
  - 全BGPルータで同じIPアドレスで対象ルータを認識することが可能

iBGPの
基本的コンフィグレーション

```
Interface Loopback 0
 ip address 202.216.41.1 255.255.255.255
!
Interface FastEthernet 2/0
 description NSPIXP2 Segment
 ip address 202.249.2.41 255.255.255.0
!
Router ospf 4689
 network 202.216.41.1 0.0.0.0 area 0
 network 202.249.2.0 0.0.0.255 area 0
 passive-interface loopback 0
 passive-interface FastEtheret 2/0
!
Router bgp 4689
 neighbor IBGP peer-group
 neighbor IBGP remote-as 4689
 neighbor IBGP update-source Loopback 0
 neighbor 202.216.41.2 peer-group IBGP
 neighbor 202.216.41.3 peer-group IBGP
 neighbor 202.216.41.4 peer-group IBGP
!
```

Loopback 0 の設定　/32で構わない

FastE2/0 がIXセグメントだったとする

LoopbackとIXセグメントをOSPF上で定義、かつ非活性とする。これによって他のBGPルータでもそれぞれがIGP上で認識される

peer-groupを利用してみる。
等質なコンフィグには非常に有効

Update-source で、ピアリングに利用するIPアドレスを定義する

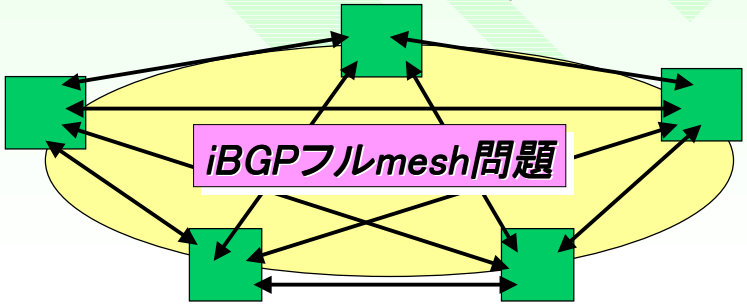
iBGPにloopbackアドレスを利用すると、BGPルータをIPアドレスで認識できるので運用上非常に便利

ISPネットワーク拡大に沿った規模対応設計

iBGPシステムのスケーラビリティ

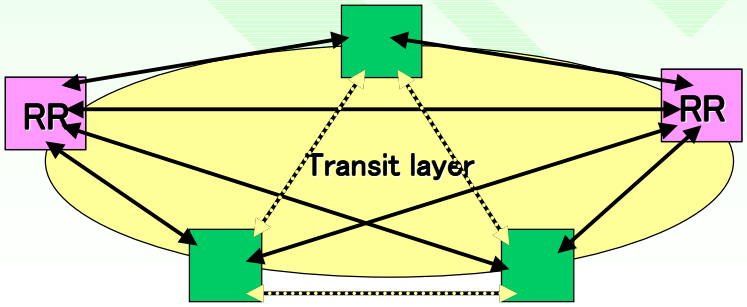
iBGPシステムのスケーラビリティ

- iBGPで得た経路は他のiBGPpeerに再伝播しないため、全ノードをmesh状にpeerする
  - ボーダルータ5ノードで既に10peer
  - 10ノードでは? $_{10}C_2 = 45$
    - » 11ノード目の増設にあたって10peerの追加



iBGPフルmesh問題解決策 iBGPルートリフレクタ(1)

- リフレクタとリフレクタクライアントの2階層化
- リフレクタからクライアントにはiBGPで得た経路を再分配する



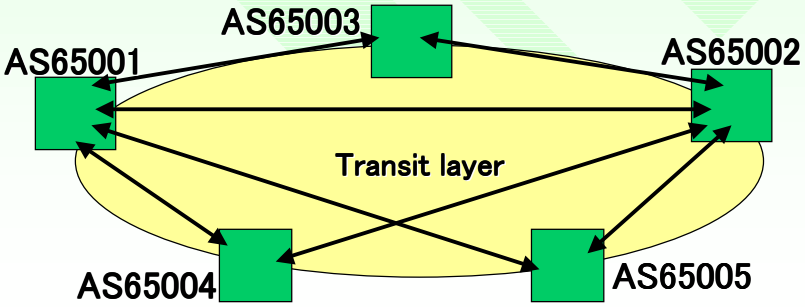
iBGPフルmesh問題解決策 iBGPルートリフレクタ(2)

- コンフィグレーション
  - リフレクタ側で以下のように設定
  - クライアント側では設定不要
    - » 階層化可能

```
router bgp 4689
 bgp cluster-id FOUR-BYTE-CLUSTER-ID
 neighbor CLI.ENT.IPA.DDR remote-as 4689
 neighbor CLI.ENT.IPA.DDR route-reflector-client
```

iBGPフルmesh問題解決策 BGPコンフェデレーション(1)

- BGPコンフェデレーション(confederation)
  - ASの中を更に小さい単位でsubASに分け、その間をeBGPで結ぶ
  - フルmeshにはる必要はなくなる



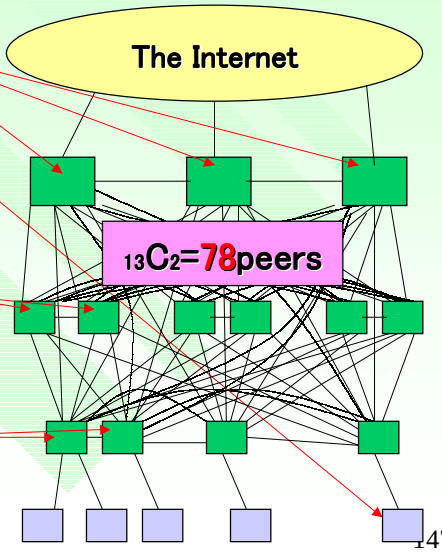
iBGPフルmesh問題解決策 BGPコンフェデレーション(2)

- コンフィグレーション
  - プライベートASを利用するのが普通
  - Confed内部となるAS番号をconfed peersで定義

```
router bgp 65000
  bgp confederation identifier 4689
  bgp confederation peers 65001 65002 65003 65004
  network …….
```

AS内BGPスケーラビリティ問題の実際

- 複数の対外接続
- 地域/POP毎にBGP接続加入者がいる
  - それぞれBGPノードが必要
- 冗長性確保が必要
  - POPにコアルータを2台
- BGP加入者増加
  - BGP加入者収容ルータの増加

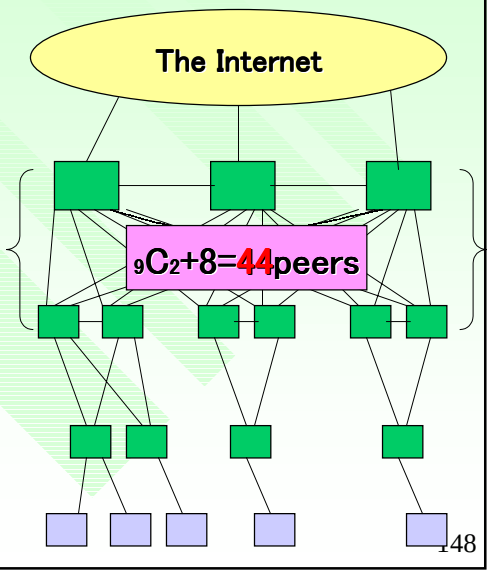


AS内BGPスケーラビリティ問題の実際 —RRによる解法

- RRの導入

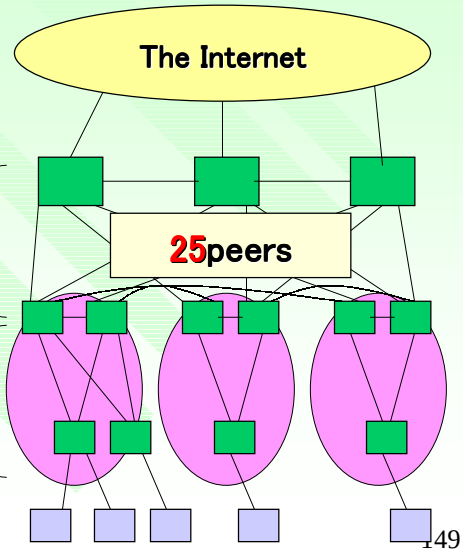
POPコアルータと対外接続ルータをフルメッシュ

加入者ルータがクライアント



AS内BGPスケーラビリティ問題の実際 —コンフェデレーションによる解法

- 地域・POPごとにsubASを設定
- BGP加入者収容ルータとの間にiBGPを設定
- IGPは分割，単一どちらでもOK



ISPネットワーク拡大に沿った規模対応設計

スケーラビリティとトラブル回避

eBGPのスケーラビリティ

- 経路数
  - 110,000 -- CIDR Report by Tony Bates**
  - 所要メモリサイズに影響
    - » 256MB必要
- Peerの数
  - IXで多数のpeerを張るとメモリ所要に影響
  - 50peer程度+upsteamで10MB程度余分に消費

** http://www.employees.org/~tbates/

eBGPの問題回避技術(1)

- 誤広告対策
  - » 隣接ASが広告する経路は完全にいつも正しいとは限らない
    - 誤った経路受領は障害の原因となる
  - AS-pathによるフィルタリング
    - » 隣接ASが広告するAS-pathを予め知らせてもらい、知らせてもらったAS-pathの経路しか受け取らない
  - プリフィクスフィルタリング
    - » (主に顧客の場合)顧客が広告するプリフィクスを予め知らせてもらい、フィルタする
  - Maximum-prefix を絞る
    - » Neighbor NE.IG.HB.OR maximum-prefix 1000 900
      - C社コマンド。1000経路までしか受けず、900でアラーム

eBGPの問題回避技術(2)

- Route flapping
  - リンク不安定などによる経路広告のばたつき
  - 経路更新，消去の連続でCPUリソースを浪費
  - 対処策：Flap Dampening
    - » ..(config-router)# bgp dampening　c社コマンド
    - » ばたつく経路に一定時間のペナルティを課して、経路テーブルから消す

eBGPの問題回避技術(3)

- ポリシ変更の負担軽減
  - ポリシ変更を反映には、peerのクリアが必要
    - » Upstreamの場合、full route を受けるため負担
  - 対処策：　soft-reconfiguration　　c社機能
    - » クリアなしに経路に対するポリシ反映
    - » Outbound はコンフィグそのままで実行可能
      - Clear ip bgp PEER soft out
      - 一旦広告していた経路を取り消して、再広告
    - » Inbound はneighbor定義が必要
      - Neighbor ADDRESS soft-reconfiguration inbound
      - ネイバから受けたそのものを蓄えておき、それに対して新たなポリシを適用
      - メモリが余分に必要なので注意。Full routeで10MB程度

ポリシルーティング

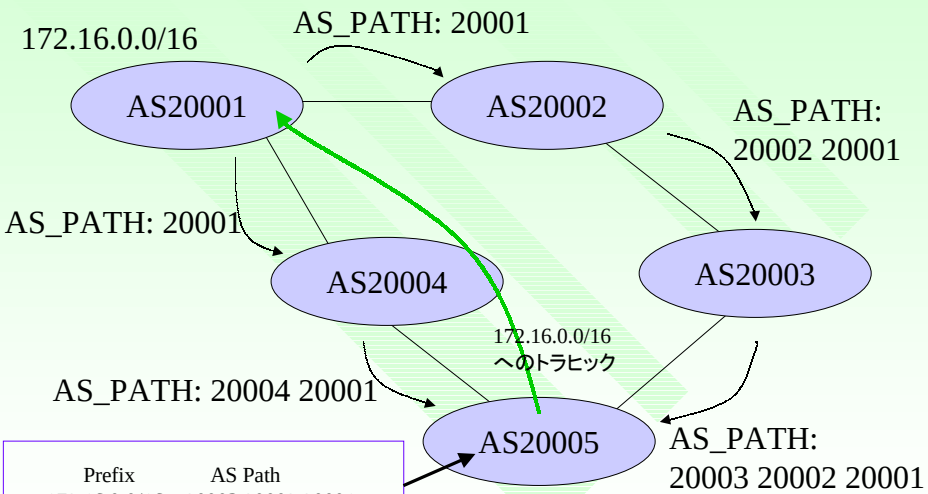
ポリシルーティング

- BGPにおける経路情報の扱い
  - プリフィクス(NLRI)＋パス属性
  - パス属性値の調整，パス属性値に基づく経路選択を行うことができる
- ルーティングポリシ
  - 複数peerを持つASとの間でどのようにトラヒックを交換するか
  - セキュリティのために経路をフィルタする
  - 複数のupstreamに対するトラヒックバランス

ポリシルーティングを可能にするパス属性値

- AS_PATH
  - 経過AS列，短いほうが優先。
  - AS-path prependでAS列長の調整が可能
- LOCAL_PREF – Local Preference
  - 設計者意図の優先順位付け
- MULTI_EXIT_DISC – Multi Exit Discriminator
  - 隣接する同一ASの複数peerの優先度
- COMMUNITY – Community Attribute
  - 32ビットの値を付加できる。プロトコル上、値に意味はないが、有効な利用法がカレントプラクティスに存在

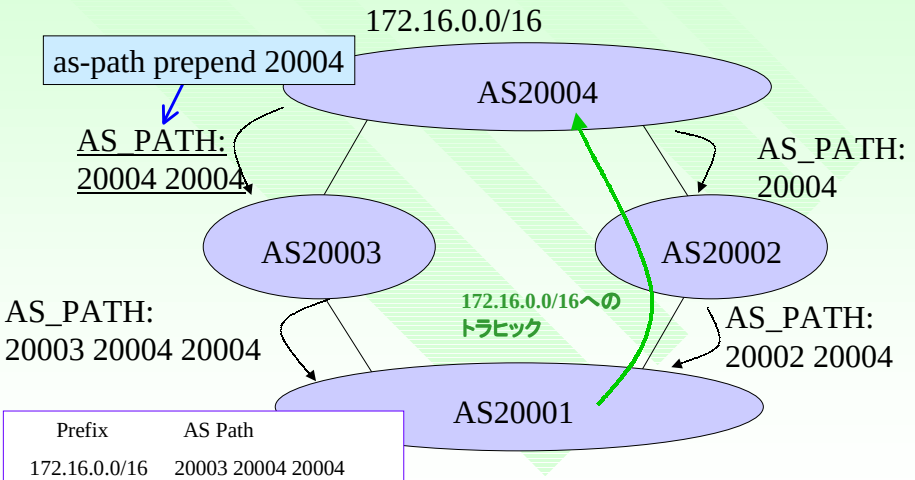
AS_PATH



通常、AS_PATHが短い（AS数が少ない）ものを選択する

AS Path Prepend

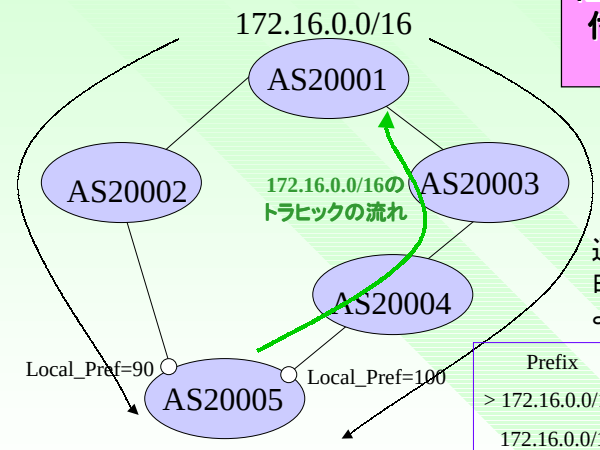
ASを余計につけて、AS_PATH_lengthを長く見せるテクニック



Prefix	AS Path	
172.16.0.0/16	20003 20004 20004	
> 172.16.0.0/16	20002 20004	◎

LOCAL_PREF

他ASから受領する経路に付加し、大きいほうが優先される



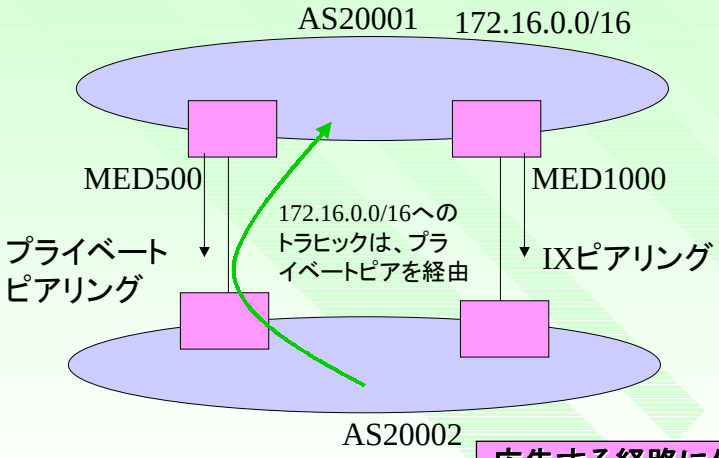
通常 AS_Pathの小さいAS2経由が優先だが、Local_Prefにより、AS4の経路を優先する。

Prefix	AS Path	LocPref	
> 172.16.0.0/16	20004 20003 20001	100	◎
172.16.0.0/16	20002 20001	90	

AS_PATH:
20002 20001

AS_PATH:
20004 20003 20001

MULTI_EXIT_DISC



	Prefix	AS Path	MED	
	172.16..0.0/24	20001	200	
>	172.16..0.0/24	20001	100	◎

広告する経路に付加し隣接ASに伝えられ、隣接ASにおける経路選択で使われる。数値が小さい方が優先

COMMUNITY(1)

- 32ビットの整数値，透過性
- Well-known Community
  - No-export:
    - » 自AS以外に広告しない
  - No-advertise:
    - » 受領したルータ以降に広告しない
- Well-known ではないCommunity
  - 経路情報を受領したAS，ルータで解釈させ、何らかのポリシ付加を発生させる

COMMUNITY(2)

- 一般的な利用法
  - New-format – 32ビットを16ビットずつに二分
    - » 5511:1000
  - 上位 – ターゲットAS
  - 下位 – ターゲットASでの動作
- 例1:RFC1998　MCI(現CWnet)における実装例
  - 3561:70　そのプリフィクスにLocPref=70付与
  - 3561:80　そのプリフィクスにLocPref=80付与
  - . . . .
  - そのASからの戻りトラヒックの制御に便利！

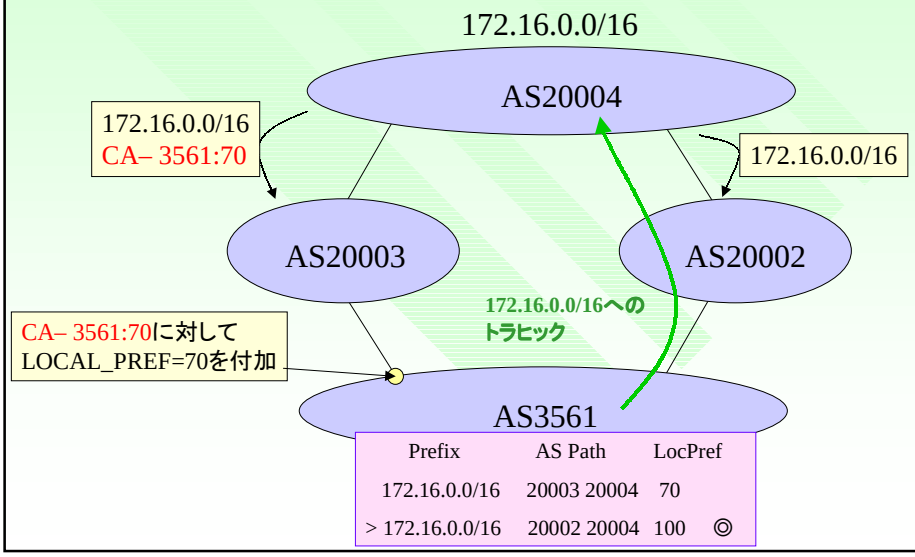
COMMUNITY(3)

- AS5511 Opentransit Internet の例

5511:1000	米国ピアに非広告	5511:2000	欧州ピアに非公告
5511:1001	米国ピアにプリペンド1	5511:2001	欧州ピアにプリペンド1
5511:1002	米国ピアにプリペンド2	5511:2002	欧州ピアにプリペンド2
5511:1101	Sprintlinkに非公告	5511:2101	Equantに非公告
5511:1102	ICMに非公告	5511:2102	Eboneに非公告
5511:1201	Sprintlinkにプリペンド1	5511:2201	Equantにプリペンド1
5511:1202	ICMにプリペンド1	5511:2202	Eboneにプリペンド1
5511:1301	Sprintlinkにプリペンド2	5511:2301	Equantにプリペンド2
5511:1302	ICMにプリペンド2	5511:2302	Eboneにプリペンド2
5511:1401	SprintlinkにRelayPOPで非公告	5511:3000	アジア太平洋に非公告
		5511:3001	アジア太平洋にプリペンド1
		5511:3002	アジア太平洋にプリペンド2
		5511:4000	欧米亜太以外に広告しない

COMMUNITYの利用方法

経路情報に付加して広告することで、対地における経路選択を制御することができる



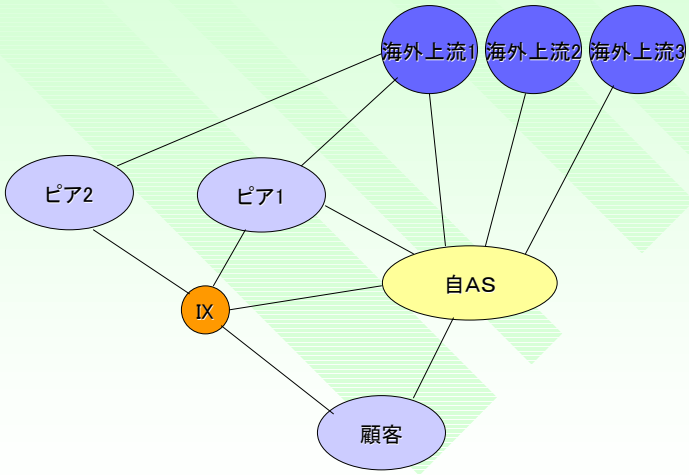
Prefix	AS Path	LocPref	
172.16.0.0/16	20003 20004	70	
> 172.16.0.0/16	20002 20004	100	◎

BGPの最適経路の決定プロセス

- 同一プリフィクスの経路情報が複数があるとき、パス属性値に拠って最適方路を決定
  - » 以下、ciscoの例
  - – 1. Local Preferenceが大きい
  - – 2. AS_PATHが短い
  - – 3. MEDが小さい
  - – 4. IGP上でNext-hopが近い(cost/metric)
  - – 5. BGPのルータIDが小さい

ポリシルーティングの実際

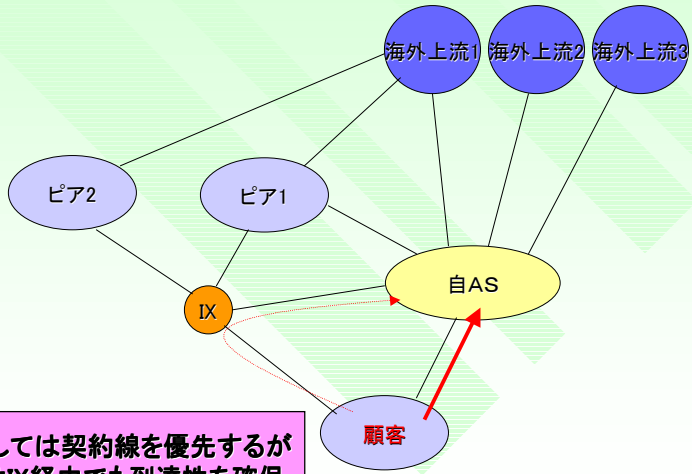
相互接続の例



ポリシルーティングの基本検討

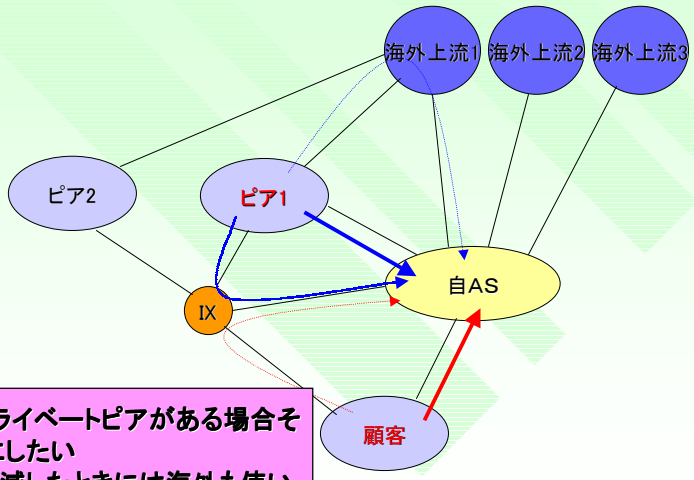
- 相互接続別・対地別の基本ポリシ付け
  - » Outbound/Inbound を対にして、どういう経路を交換するか
  - – 相互接続別
    - » 顧客　　フルルート供給，顧客経路のみ受領
    - » ピア相手　　自網顧客経路のみを相互に交換
    - » 海外上流　　自網顧客経路のみ供給，フルルート受領
  - – 対地別（優先する順番にパスを並べる）
    - » 顧客　　直接，IX経由，Upstream経由
    - » 国内対地　　プライベートピア経由，IXピア経由，Upstream経由
    - » 海外対地　　安い順番，品質の良い順番，とりあえず無制御

受領経路優先順序検討（国内）



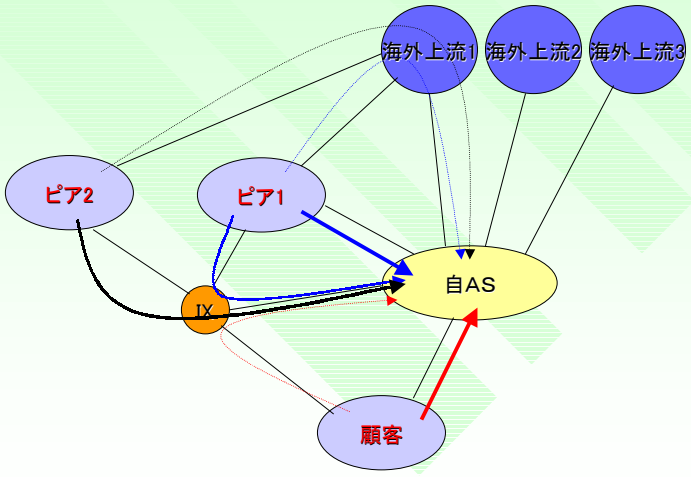
・顧客に対しては契約線を優先するが障害時にはIX経由でも到達性を確保したい

受領経路優先順序検討（国内）



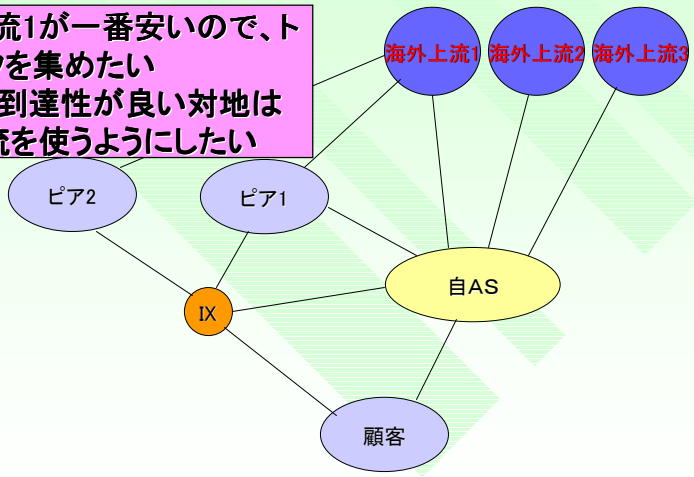
・ピアはプライベートピアがある場合そこを優先にしたい
・国内が全滅したときには海外も使いたい

受領経路優先順序検討（国内）



受領経路優先順序検討（海外）

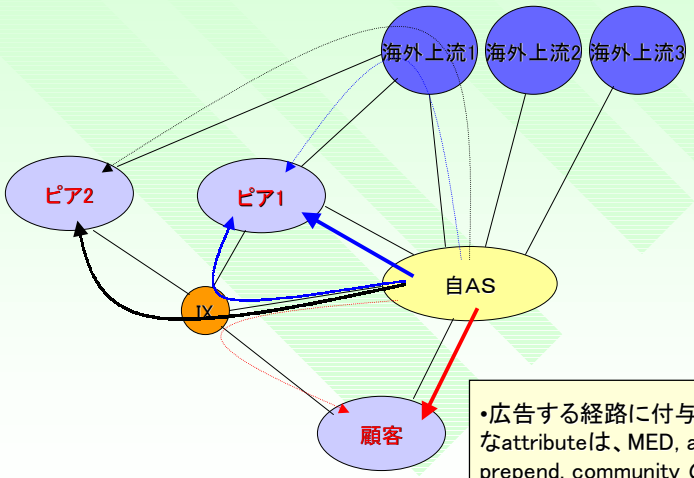
•海外上流1が一番安いので、トラフィックを集めたい
•しかし、到達性が良い対地は他の上流を使うようにしたい



受領経路に関する ルーティングポリシ実装案

- 各eBGPピアで、受領経路に対して以下の通りLOCAL_PREFを付与する
  - 顧客 110
  - プライベートピアリング 100
  - IXピアリング 95
  - 海外上流 90
- 海外上流に関して、上流2，上流3から受領する経路にAS-path prepend を1hop掛ける

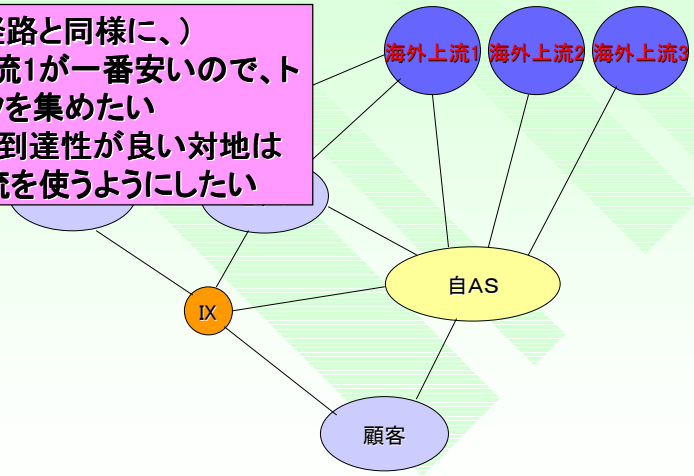
広告経路ポリシ検討（国内）



・広告する経路に付与して有効なattributeは、MED, as-path prepend, community のみ

広告経路ポリシ検討（海外）

・（受領経路と同様に、）
・海外上流1が一番安いので、トラフィックを集めたい
・しかし、到達性が良い対地は他の上流を使うようにしたい

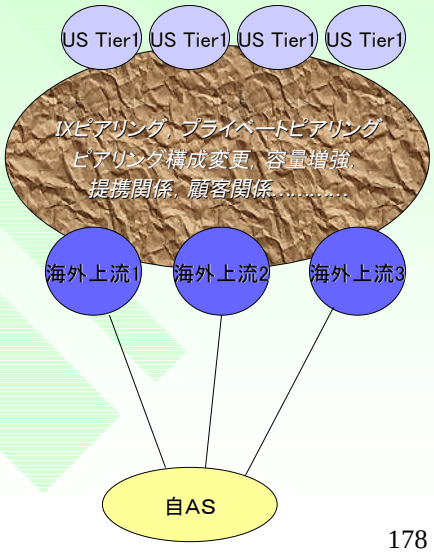


広告経路に関する ルーティングポリシ実装案

- 各eBGPピアで、広告経路に対して以下の通りMULTI_EXIT_DISCを付与する
  - 顧客 500
  - プライベートピアリング 900
  - IXピアリング 1000
- 海外上流に関して、上流2，上流3に広告する経路にAS-path prepend を1hop掛ける

海外上流のトラフィック制御の難しさ

- 海外上流からのinboundトラフィックのバランスは日々変化する
  - 接続構成は常に更新される
- 調整方法としては、
  - as-path prepend,
  - Community
  - プリフィクスの一部のみ適用
- 精密な調整が不要な工夫が必要
  - 全部従量課金サービスにしてコストへのインパクトを少なくする
  - 十分なキャパシティを準備して突出しても性能低下にならないようにする



(3)大規模な経路制御設計の実際

(3-1)
概要・設計指針

参考文献

- RFC2791 - Scalable Routing Design Principles
- 著者: Jessica Yu, CoSine
- Informational RFC
- IIJ近藤邦昭氏，友近，前村で元となるインターネットドラフトを和訳
  - http://www.janog.gr.jp/doc/draft-yu-routing-scaling-01-j.txt
- 大規模ネットワークの経路制御システムにおける問題点を概説し、設計上の指針を示すもの。

問題点

- ルータのリソース消費
  - メモリ消費要因
    - » 経路数過多，IX，顧客集線ルータにおける方路過多，iBGPセッション過多
    - » BGPのプリフィクスフィルタリング，IGPの肥大化したLSDB
  - CPU資源消費要因
    - » 不安定なネットワークのflapping
    - » フラッディング-全ネットワークへのLSA伝搬
    - » 過負荷の悪循環

スケーラビリティ確保のための指針

- 階層構造化
- 区画化
- 適切なトレードオフの設定
- 経路制御処理の負担を軽減
- スケーラブルな経路制御ポリシ，実装
- …

階層構造化

- 単一階層，フルメッシュ構成はスケールしない
- Transit Core Network と Access Network の二層に分けると分かりやすい
  - OSPFのバックボーンエリアとその他のエリア
  - IS-ISのlevel1, level2
  - iBGPルートリフレクタの階層化
- 構造を過度に複雑にしないこと

区画化

- 階層構造化においては、二層目が区画化されている
  - OSPFのエリア分割
  - BGP Confederation によるIGPドメインの分割
- 問題・障害の局所化効果
- 経路の集成

適切なトレードオフの設定

- 冗長性　対　スケーラビリティ
  - 過度の冗長性を持たせない。
- 収束性　対　安定性
  - Flap dampeningなど、収束性を犠牲にしながらそれを最小にする努力

経路制御処理の負担を軽減

- 経路情報の削減
  - 適切な aggregate, summarize
  - できる限りdefault route を利用する
    - » Single-homeの加入者
  - 過度な冗長構成を取らない
    - » 代用方路は2つ以上いらないのでは?

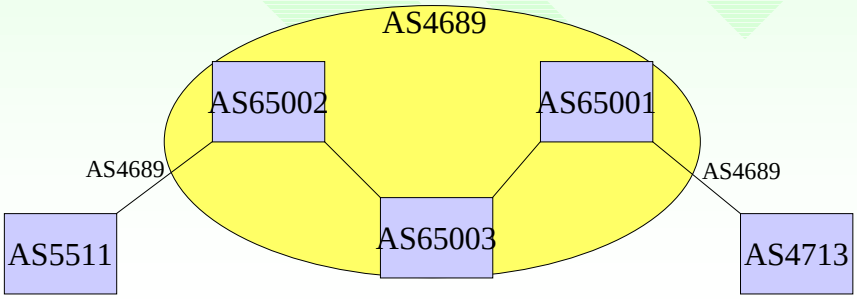
スケーラブルな経路制御ポリシ，実装

- 要件を満たす範囲で可能な限りポリシを簡素にする
- 間違いの起こりやすい手作業を避け、可能な限り自動化する
- 経路制御の完全性のためにプリフィックスによる経路フィルタリングを実施することは例外として、プリフィックス毎のポリシは可能な限り避ける
- 例外を作ることを避ける

(3-2)
Confederationの実例

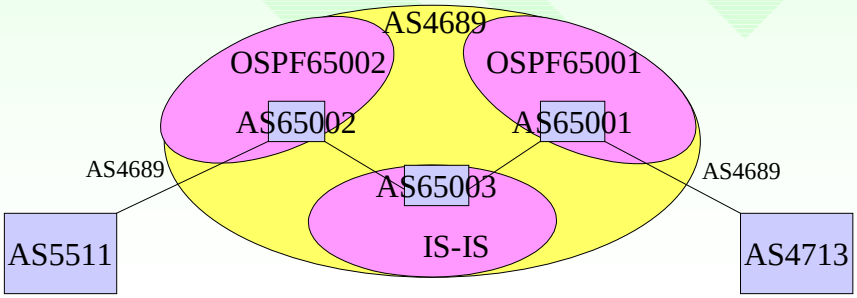
BGP Confederationとは?

- 複数のASナンバのBGPスピーカを、外から見たときに単一のASナンバとして見せることができる



IGPスケーラビリティ解決に利用

- subAS毎に別のIGPプロセスを起動
  - 一つのASをsubASに分割する
  - AS内のIGPが巨大化しても分ければ大丈夫
    - » OSPFが耐えられなくなったら分ければ良い



Confederationの起動

```
router bgp 65001
  bgp confederation identifier 4689
  bgp confederation peers 65002 65003 65004
  network …….
```

Confederationの利点

- OSPFプロセス肥大化への対策
  - OSPFプロセスを小さくする!!
  - 大きくなったら分割すれば良い
- 地域ごとにポリシ制御可能
- 障害の局所化
  - 全網規模になるのだけは避けたい
- ネットワークの統合や、内部での管理分割など

Confederationにおける経路の扱い

- confedの中のsubAS間はeBGP，subASの中でもiBGPは張れる
- LocPref, MED, NextHopは、subASをまたいでも保存する(iBGP的扱い)
- confed内のsubASは　ASpathとして観測できるがhop数評価には利用されない。

Confederationにおける経路制御設計tips

- subAS毎のnetwork定義は事実上不可能
  - OSPFをredistributeして、aggregateする
- AS全体の集成経路の生成
  - 中央にaggregate generator
  - 対外接続ルータでspecificをfilter out
- Next-hop
  - nexthop-self でsubASをより普通のASのように扱う
  - Inter-subAS領域でIGPを立ち上げればnexthop-selfは要らない
- 対外接続ルータを単独1ASにする
  - OSPFを起動が不要，BGPハンドリングに専念

(3-3)
static-to-bgpの実例

概要

- OSPF経路数の増大とその影響
- OSPF経路削減の諸方法
- static経路のBGPへのredistribute
- その他付随するテクニック
- 結果
- 考察

OSPF経路数の増大

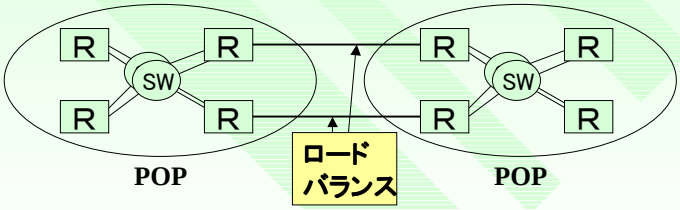
- AS4713(OCN)では、OSPFの経路数が非常に増えていた
  - 90%強がexternal経路。これはcustomerへのstaticの経路をOSPFにredistributeしていた経路
- あまり効率よくaggregateできない
  - JPNICおかわり制限

OSPF経路数の増大の影響

- OSPFにはexternalとはいえある程度以上の数の経路を流すべきでない
  - 疑似環境において検証してみたところ、ある程度以上のexternal経路が流れるとOSPFが不安定になることが確認できた
  - Exchange→init

適用ネットワークの特徴と条件

1. トラヒックのロードバランスをしながら
   - リダンダンシーをとるため様々なところでトラヒックのロードバランスをはかっている



2. サービスの停止がなく
3. 運用の手順の変更を極力少なく

OSPF経路削減の諸方法

- OSPFを分割する（リンク部分で）
  - Confederation等
    - » ロードバランス困難
      - 一つ手前のルータでバランスさせないといけない
    - » サービス停止、運用変更
- OSPFに変えてIS-ISにする
  - 設計・運用ノウハウが足りない
  - 実際効くのかどうかわからない
- その他

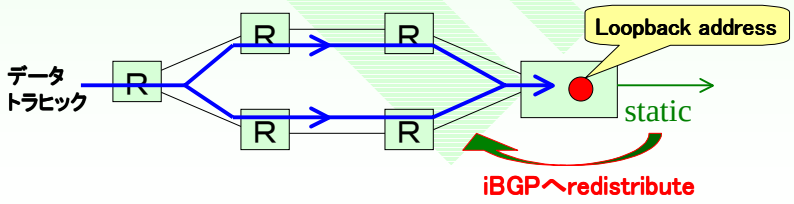
- static経路をOSPFでなく直接iBGPにredistributeさせる

Static経路のBGPへのredistribute

- static経路をOSPFでなく直接iBGPにredistribute
  - IGPとしてのBGP（external経路はBGP、トポロジはOSPF）
  - 前々ページ、1.2.3.などの前提条件を満たし、かつOSPFの経路数を削減する方法
  - BGPは経路数についてスケーラビリティが高い
- 前提
  - iBGPセッションは当然（元々）loopback同士
  - ルータのloopbackアドレスなどは当然（元々）OSPFに流れている
  - staticを設定しているルータもBGPをしゃべらす

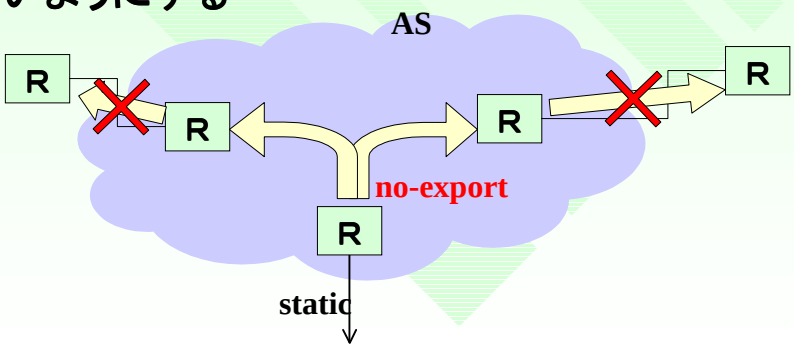
仕組み

- その経路へデータが行くためにはBGP next-hopであるredistributeしたルータのloopbackアドレスへ向かおうとする
- BGP next-hopへ向けてOSPFで作られたルーティングテーブルをrecursive lookupする
  - ロードバランスする



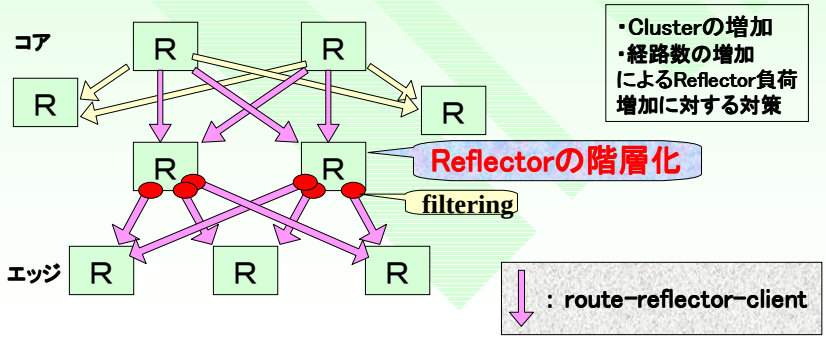
その他付随するテクニック(1)

- BGPで、no-exportのcommunityをつけることによりspecificな経路をAS外部に流れないようにする



その他付随するテクニック(2)

- Route Reflectorの階層化を用いることによってReflectorの処理を軽くする
- フルルート必要でないところはfilteringする



結果

- 実際にこれらの方法を用いることによってそれ以来AS4713の内部ルーティングの安定性が増した
- 運用手順もほとんど変化なし

- Static経路はiBGPに流し
- OSPFはトポロジーの情報をもつだけでよい！！！！！！

参考文献

参考文献

- RFC2791 - Scalable Routing Design Principles
  - Jessica Yu
- インターネットルーティングアーキテクチャ 第2版
  - Sam Halabi / Danny McPherson著, 鈴木 訳
- インターネットルーティング入門
  - 友近・池尻・小早川 著, 翔泳社
- インターネットルーティング
  - C. Huitema 著, 前村 監修・エクストランス 訳, 翔泳社

ご静聴ありがとうございました。

--大規模ネットワークにおける経路制御設計--

NTTコミュニケーションズ ビジネスユーザ事業部
友近 剛史 tomo@byd.ocn.ad.jp

イクアント
前村 昌紀　maem@gip.ad.jp