大規模ネットワークにおける経路制御設計

1999年12月16日
NTTコミュニケーションズ（株） 友近 剛史
日本電気（株） 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	75	友近
(2) BGPのシステム設計論	40	前村
(3) Jessica's I-D の紹介	35	前村
(4) static-to-bgpの設計の実際	15	友近
(5) Confederationの設計の実際	15	前村

(1) IGPのシステム設計論

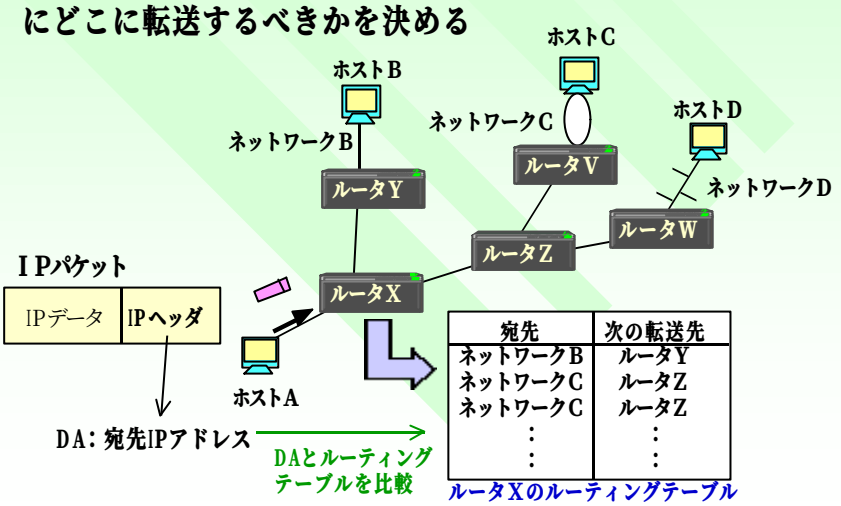
- 概要
- RIP
- OSPF
  - OSPF基礎
  - OSPF設定
  - エリア
  - DRとBDR
  - ルーティングテーブルの作成まで
  - 負荷について
  - 設計Tips
- IS-IS

概要

--基本の復習--

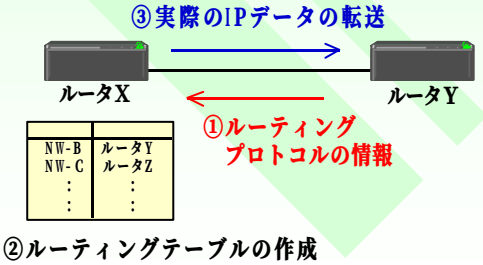
ルータ

- ルータでは、ＩＰパケットのヘッダに書かれている宛先ＩＰアドレスと、ルータのルーティングテーブルを参照し、次にどこに転送するべきかを決める



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

IGPとEGP

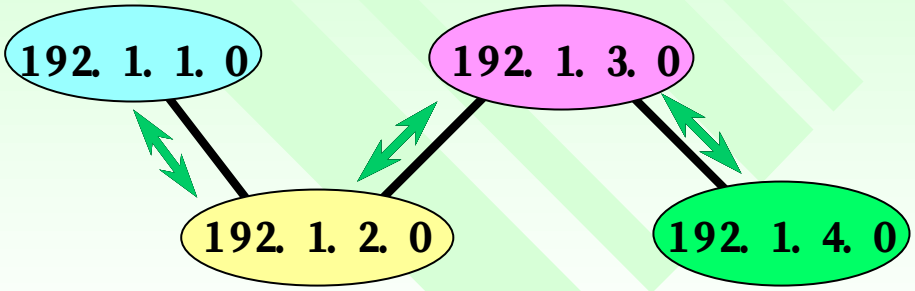
- **IGPs（Interior Gateway Protocols）**
  - 同一ＡＳ（Autonomous System:自律システム）内で使用されるルーティングプロトコル
  - RIP（Routing Infomation Protocol）
  - OSPF（Open Shortest Path First）
  - IS-IS (Intermediate System-to-Intermediate System)
- **EGPs（Exterior Gateway Protocols）**
  - ＡＳ間で使用されるルーティングプロトコル
  - BGP（Boarder Gateway 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をマルチキャストでネットワークで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ーベースを持つ
- **パスベクターアルゴリズム**
  - 近隣ルータ間で交換される情報に基づき、BGPの経路情報がたどっていくパスを示す一連のAS番号を運ぶ
  - それをもとにASパスツリーをつくり、経路選択する

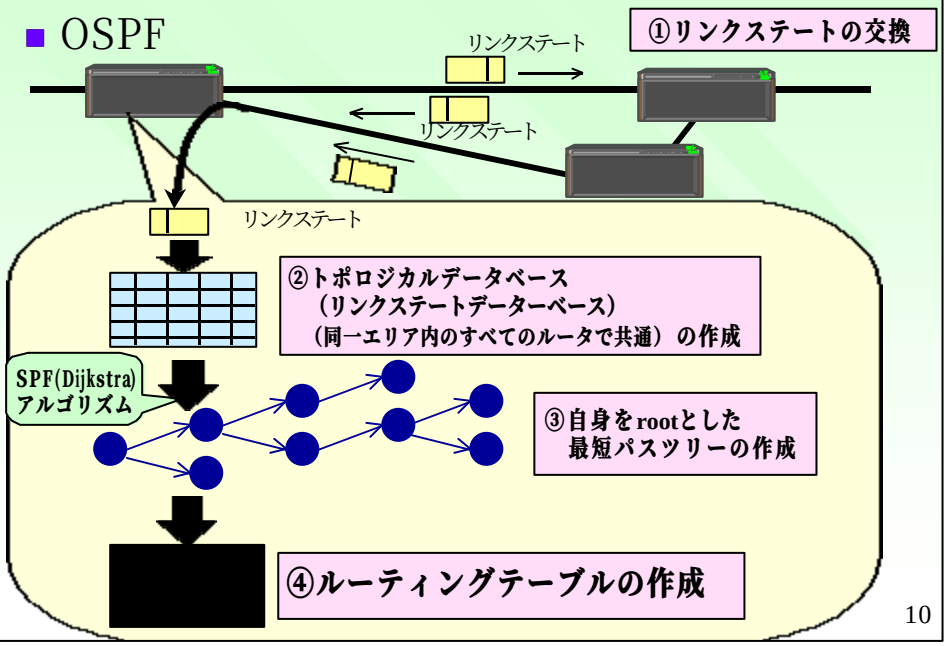
ディスタンスベクターアルゴリズム

- RIP



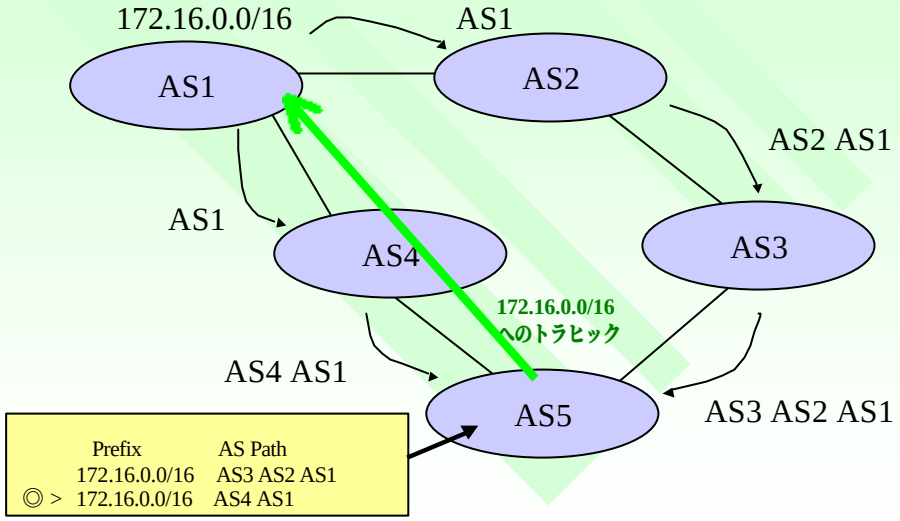
リンクステートアルゴリズム

- OSPF



パスベクターアルゴリズム

- BGP



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
- 送信元と宛先の間で最適な経路を探すときにホップ数を比較
- 最大のホップ数を15と制限している
- デフォルトで３０秒に１回各ルータはルーティング情報をブロードキャストで送出
- 古くからBSD UNIXシステム上でroutedという形で実装されていた
- 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
  - 処理の負荷が小さい
  - 多くのネットワーク機器で対応されている
- 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
  - 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
  - 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

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
- 認証機構を提供
- 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
- 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

OSPF

OSPF基礎

- RFC 1583 (March 1994)
- →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

OSPF基礎（続き）

- 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

OSPF設定(C社の例)

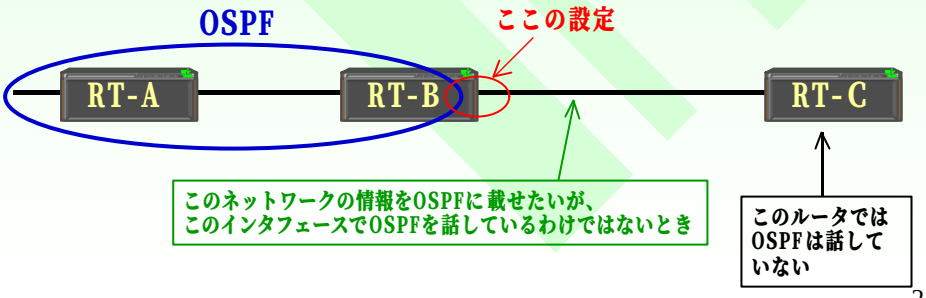
- router ospf <process ID>
  - 一つのAS内で一つしかOSPF processを走らせない場合、process IDは1～65535の何番にしてもいいが、自分のASと同じ番号にすることが多い
- network 192.168.0.0 0.0.0.15 area 0
  - このコマンドは大きく言って２つの意味がある
    - » そのネットワークをOSPFに広告すること
    - » そのネットワークがあるインタフェースでOSPFを話すこと

  上記２つが基本で、最低限のOSPFのconfig

- ospf log-adjacency-changes
  - OSPFのSTATEの変化をlogに残す設定

OSPF設定(C社の例)

- 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
    - » “subnets” を忘れないように



OSPF設定(C社の例)

- Interfaceコマンド
- 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NWで（非ブロードキャストNW：30秒)
  - ip ospf hello-interval <seconds>
- デッドタイマー
  - HELLOトを受け取らなければ傷害だと判断
  - デフォルト HELLOインターバルの４倍
  - ip ospf dead-interval <senconds>

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)

OSPF設定(C社の例)

- デフォルトルートの生成
  - デフォルトルートはredistributeされない
  - 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
  - BGPスピーカーでないルータ（エッジに近いルータ）が、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、設定する
    - » 外部に近いルータで設定する
  - ただし、デフォルトルートを広告するBGPスピーカーに、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を受けることができない
    - » これはネットワーク設計上、注意を要することである

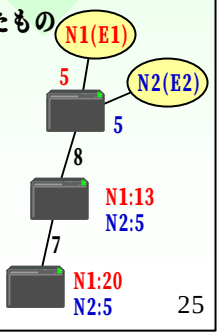
OSPF設定(C社の例)

- 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）には、type1、type2がある
  - redistribute **** metric <metric>metric-type <1|2>
  - default-information originate metric <metric> metric-type <1|2>
    - » これもexternalになる
  - type1：externalのコストにそこまでのinternalコストを加えたもの
  - type2：externalのコストのまま
  - 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
    の順番で優先される

*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

(O O IA O E1 O E2)

sh ip routeの出力



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

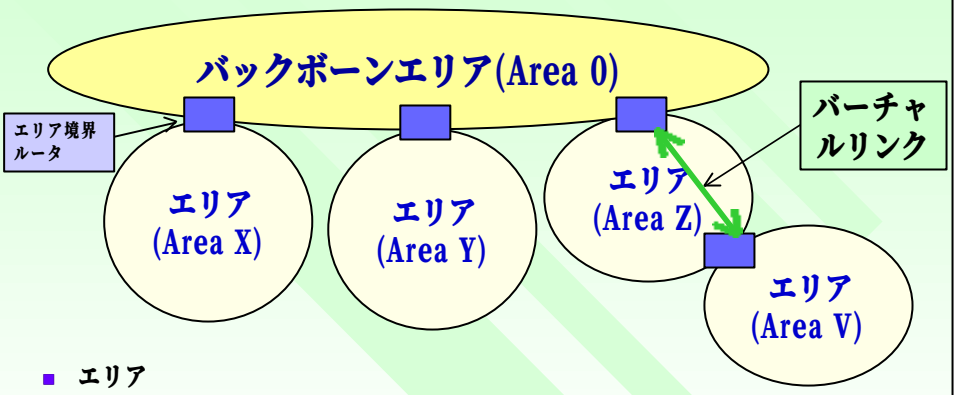
網設計においての基本

- 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
  - コストの低減
    - » 回線数、量の削減
    - » BackUp回線も、1:1でアクトスタンバイよりn:1でロードバランス

要望条件、ポリシー（続き）

- 信頼性の向上
  - » ノード傷害
  - » リンク傷害
  - » 機種レベルでの冗長化
  - » 技術レベルの冗長化（例：Giga-EtherとFDDI）
  - » ビル傷害
- 保守運用性の向上
  - » 物理的、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

エリアについて



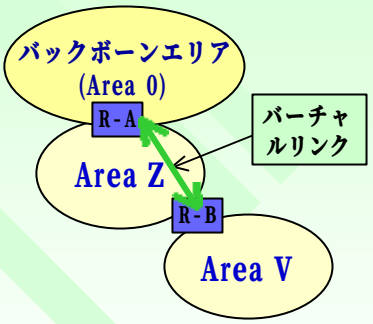
- エリア
  - 同一エリア内のすべてのルータでトポロジーデーターベースは共通
  - バックボーンエリアに他のエリアがぶら下がる形
- バーチャルリンク
  - バックボーンに対して物理的コネクションを持たないエリアを取り込む
  - エリア0が切断されてしまう場合にバックボーンをつなぐため(patching)

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
- 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
- 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
- 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（バーチャルリンク）

- バーチャルリンクをあてにして大規模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

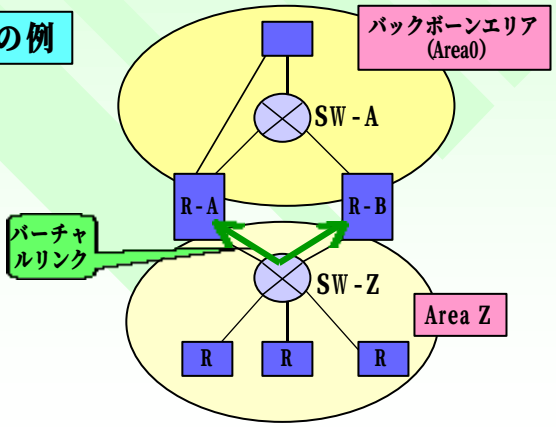
- R-A
  - area Z virtual-link <Router-BのR-ID(loopbackアドレス)>
- R-B
  - area Z virtual-link <Router-AのR-ID(loopbackアドレス)>

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patchingで使用するときや網変更の際の緊急措置対応のため、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



DRとBDR

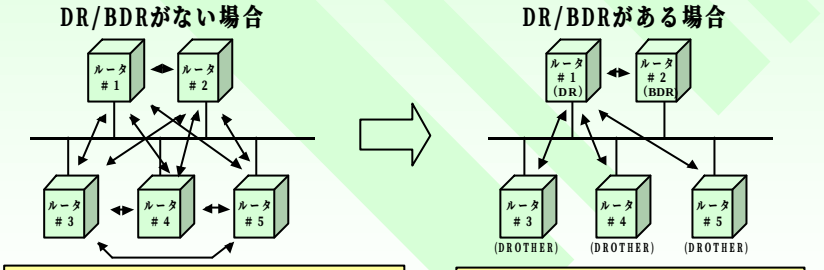
- Designated Router：指名ルータ
- Backup Designated Router：バックアップ指名ルータ
- マルチアクセスネットワーク上で必ず１つ存在
- BDRはDRがダウンしたときのバックアップ
- それ以外の各ルータ(DROTHER)はDRと情報を交換する
- DRは結構負荷がかかるので、処理能力のあるルータや、他の処理が重くかかっていないものになるなど、考慮する必要がある
- 一つのルータが複数のネットワークのDRにならないように考慮する必要がある

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HelloPacket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

DR/BDRについて

- 隣接しているルータは１度DRに対してLink-state情報を送ると、DRがその他全ての隣接ルータに対してLink-state情報を送信する
- この仕組みにより、少ない情報交換量（トラフィック）でルータ同士が相互にLink-state情報を交換することが可能となる



N台のルータがあれば、Link-state情報を交換するのに N(N-1)/2のリンクが必要

DR(BDR)があると、Link-state情報を交換するのに N-1のリンクですむ

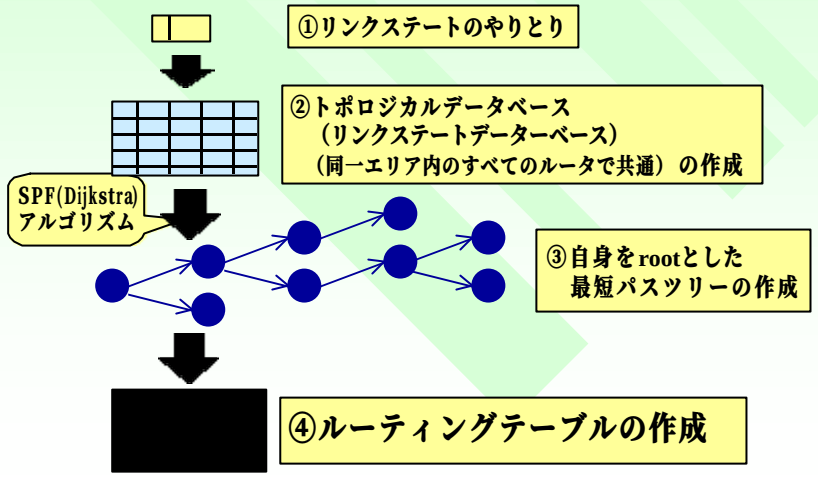
注：BDRとDROTHERはadjacentになっているが、link-stateのやりとりはしない

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

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



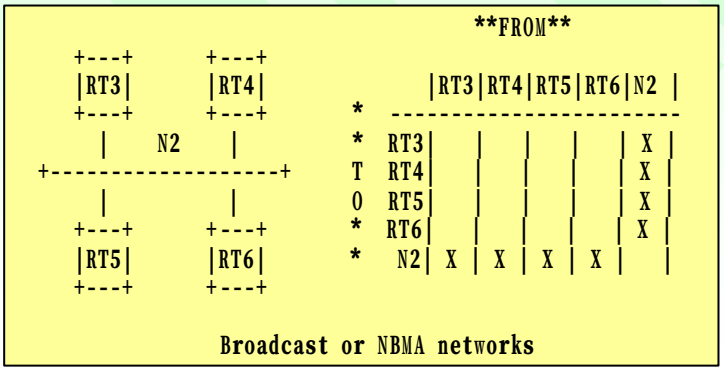
トポロジカルデータベース*

* 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

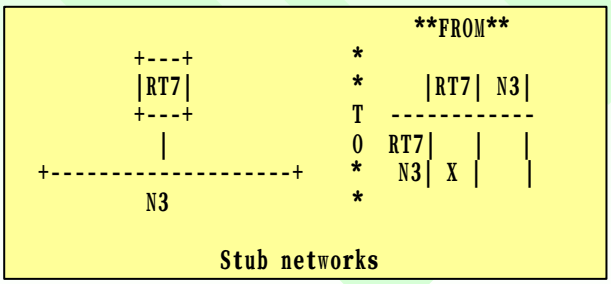
トポロジカルデータベース
(マルチアクセスネットワーク)

- **マルチアクセスネットワーク**
  - **ルータとネットワークをつなぐ**
  - **複数のルータがあるとき(transit network) →双方向**



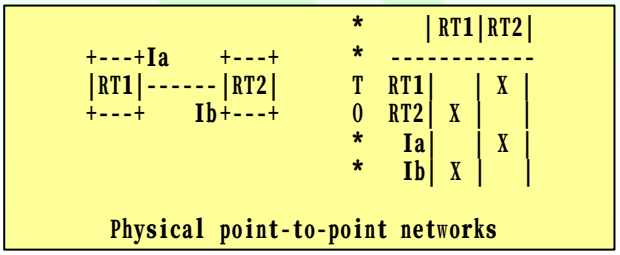
トポロジカルデータベース
(マルチアクセスネットワーク)

- **ルータが一つだけのとき(stub network)**
  →ルータからネットワークへの片方向



トポロジカルデータベース (point-to-point)

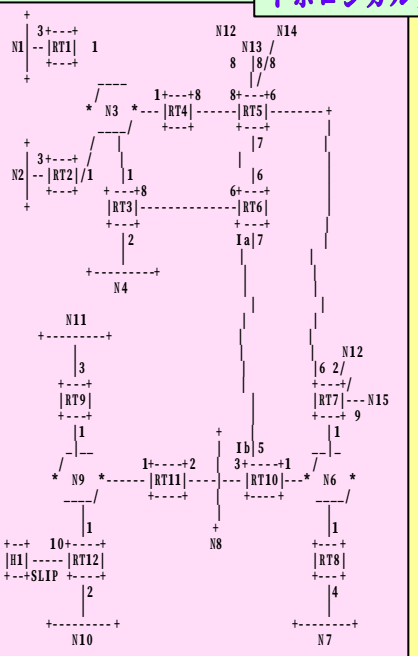
- point-to-point
  - 2つのルータが物理的にpoint-to-pointで結ばれているときは、ルータ同士をつなぐ。双方向。
  - Unnumberedのときはルータだけ
  - Numberedのときは、そのインタフェースは各ルータにstub networkでくっついているようにみなす
    - » ルータからインタフェースの片方向



トポロジカルデータベース

- データベースの中はコストを値とする
- コストはインタフェースの出力側に関するもの
- 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

トポロジカルデータベースの例



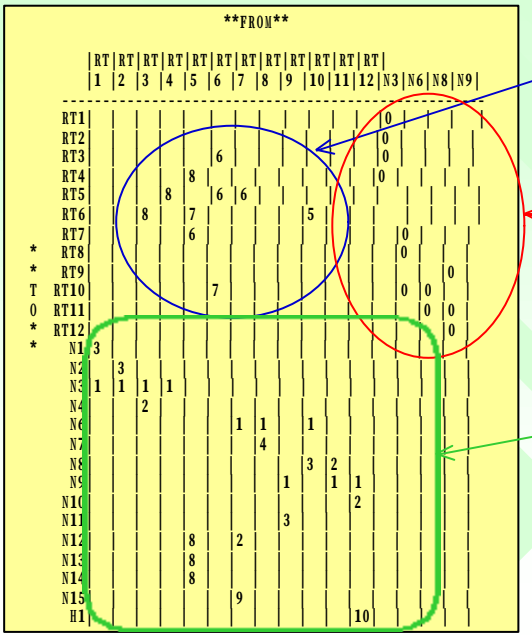
A sample Autonomous System

FROM

	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		
0 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				

The resulting directed graph

トポロジカルデータベースの内容

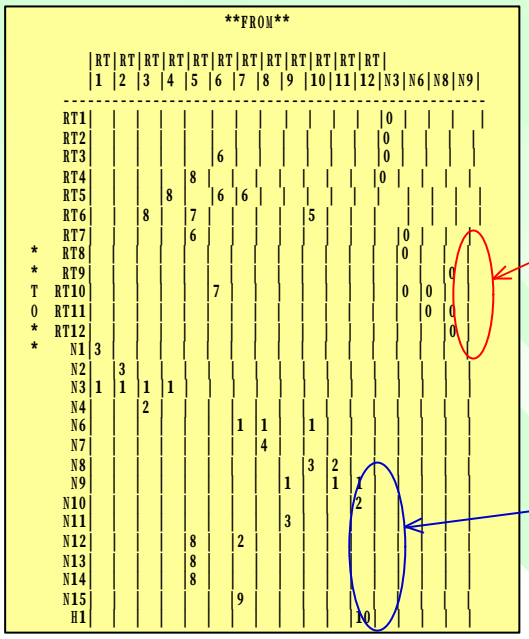


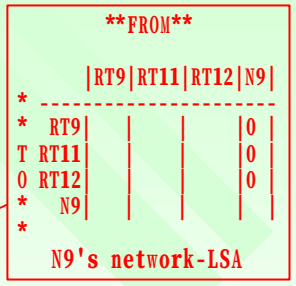
p-to-pはRT同士の辺となる

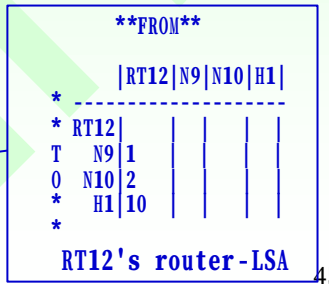
・FROMでNWがあるところは
複数のルータがあるマルチ
アクセスネットワークとなる
・NWからRTに向かうのは常に0

・RTからNW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
・値はコスト

トポロジカルデータベースと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？

Yorimichi F.

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)とメトリックを情報としてもつ
  - 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

*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
  - AS内にあるが、エリアの外にある経路（つまりエリア間経路）を記述
  - Type 3 はネットワークへの経路
  - Type 4 はAS境界ルータへの経路
- 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
  - default-information originate

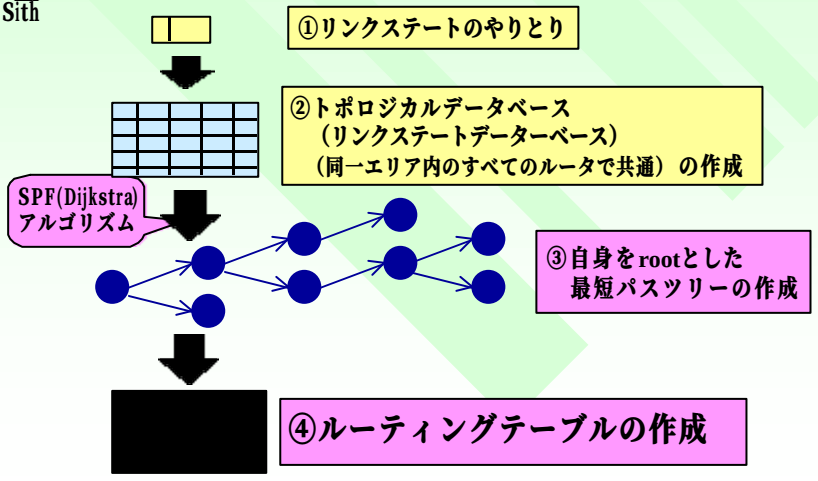
NSSA

- 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。

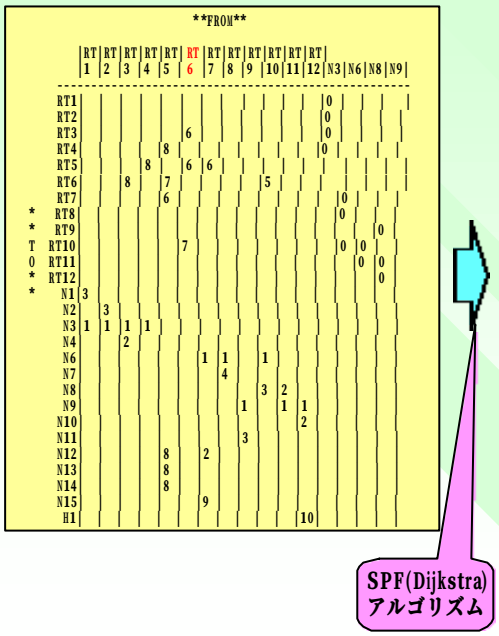
  → area0の負荷を減らすわけではないが
  →エッジの方でメモリの少ないルータとかある場合に使える
- C社ではIOS 11.2あたりから対応

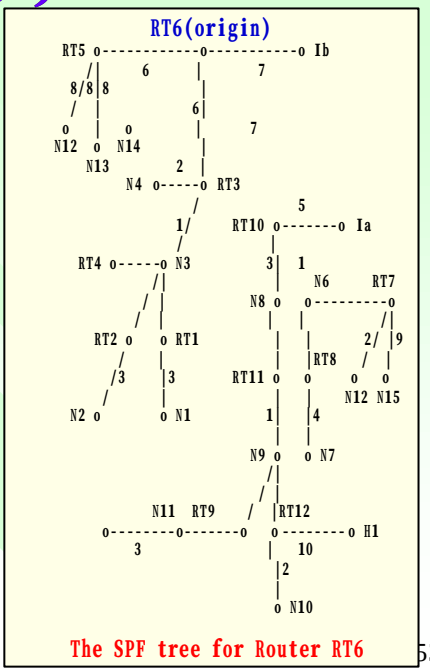
ルーティングテーブルの作成まで

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トポロジカルデーター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？



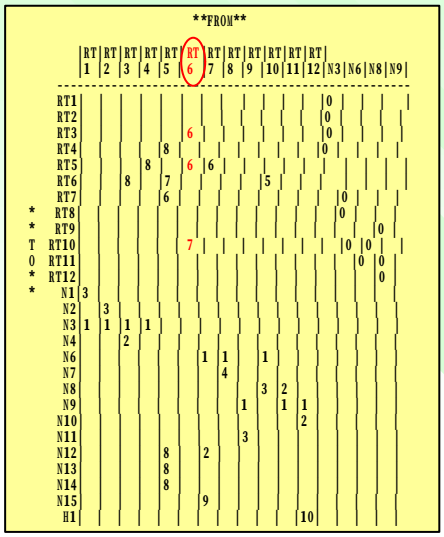
最短パスツリー





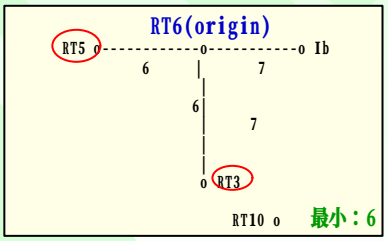
SPF(Dijkstra)アルゴリズム (1)

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

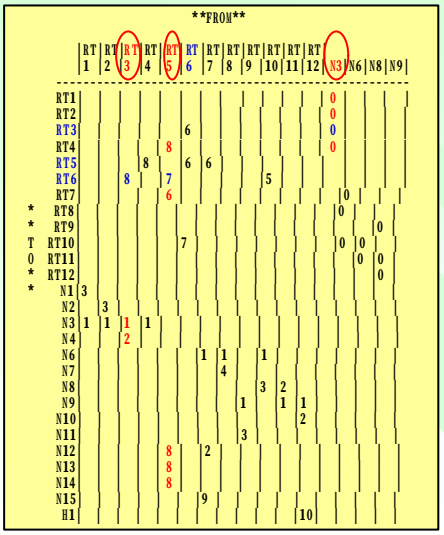


：確定

*p-to-pのlbを忘れないよう

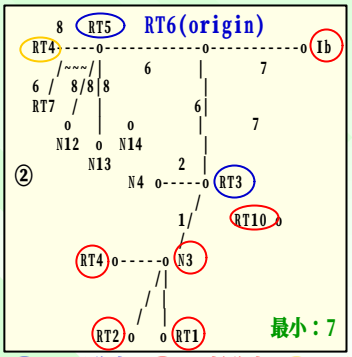
現在リーフにあるノードの中でRT6からのコストが最小である6のノードを確定する

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

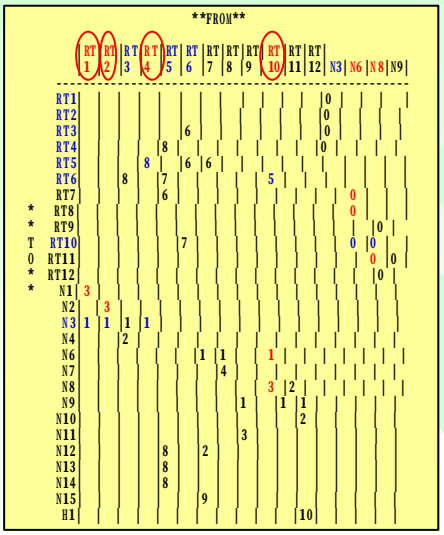


：旧確定　：新確定　：消去

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

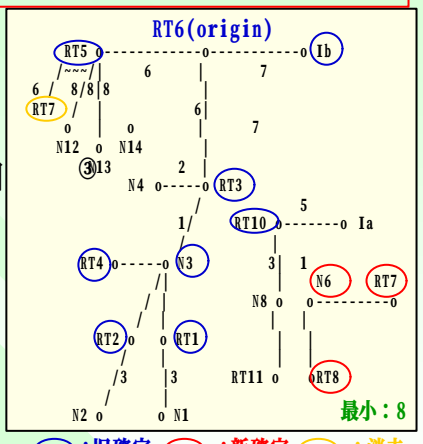
SPF(Dijkstra)アルゴリズム (3)



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

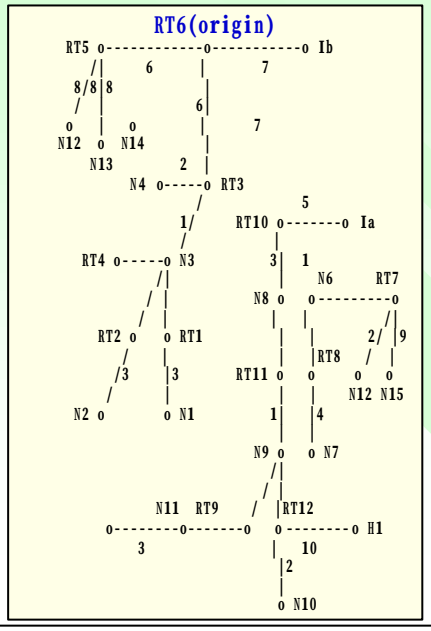


：旧確定　：新確定　：消去

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

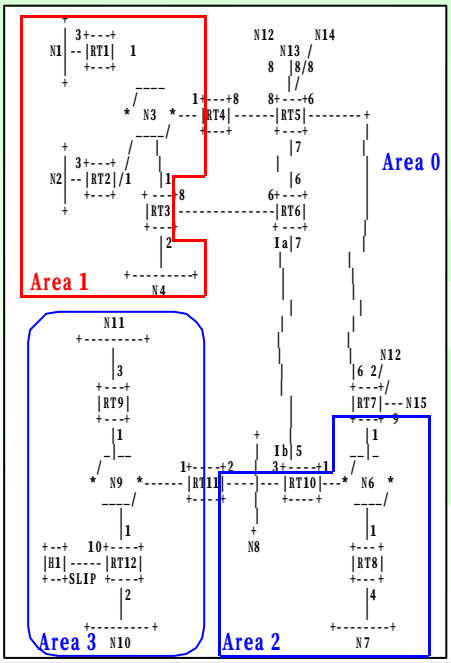


- 最短パスツリーからルーティングテーブルが作成される

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

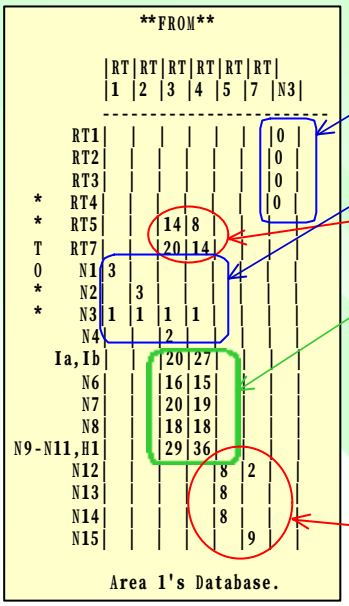


FROM

	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.

トポロジカルデーターベースの内容



Area1内（intra-area）の情報

エリア境界ルータからAS境界ルータまで

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

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

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。エリア内のノードの数を n とすると、mはnを越えることがないので
- O(l * log(n))
- といえる。ただし、ネットワーク構成にかなり左右される。
- 例えば、フルメッシュなネットワークな構成だと
- $O(n^2)$
- となる。

新確定	候補リスト
RT-a	RT-b (1) RT-c (2) RT-d (5)
RT-b	RT-c (2) RT-e (3) RT-d (5) RT-f (8)
RT-c	RT-e (3) RT-d (5) RT-h (6) RT-f (8) RT-g (10)

RT-e(3)を候補リストに入れるのにO(log(m)) かかる

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
  - 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るか？（一切責任は持てません…）
  - 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Tips

- リンク数
  - あまりリンクを持つような構成はよくない
  - フルメッシュになるATM-SW等よりも、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
  - 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Tips

- 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
  - BGP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

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
  - RSP2からRSP4にすると劇的に変わります
- 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
  - static-to-bgp
  - confederation
  - 他の候補
    - » OSPFプロセス分け
    - » IS-IS化
    - » エリア境界ルータにもっと多くのエリア
    - » virtual link
    - » ネットワーク分けて他のプロトコルで結ぶ
    - » etc...

IS-IS

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と大差ないはずだが
- 米国ISP向けにチューニングされているらしい
- 日本で使っているところはないだろう
- C社は最近はOSPFに力を注いでIS-ISにはあまり力を注いでいない、という噂もある

McQueen's grandma

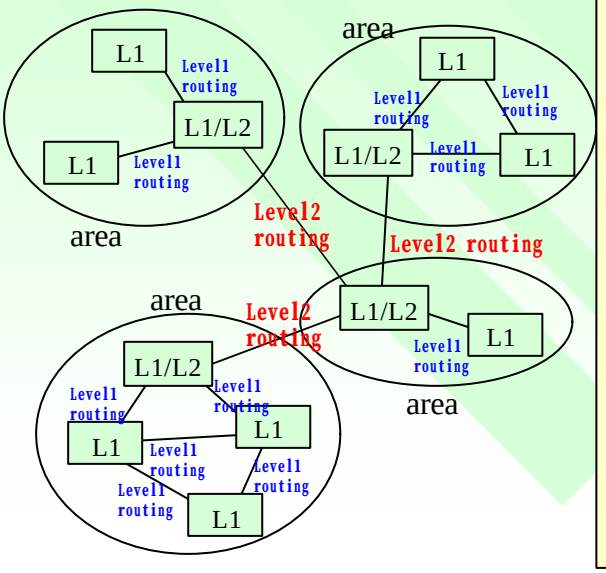
とりあえず、どんなものか見てみよう

IS-ISの特徴

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

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。

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(Connection Network Layer 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

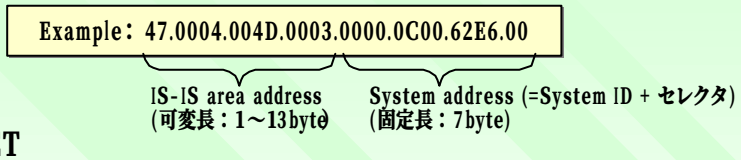
ルーティングプロトコル	OSPF	IS-IS
使用するネットワークレイヤ	IP	CLNS
ノードのID	ルータID (IPアドレスに基づく)	NET (NSAPアドレスに基づく)
できるルーティングテーブル	IP	IP及びCLNS

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



- 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。

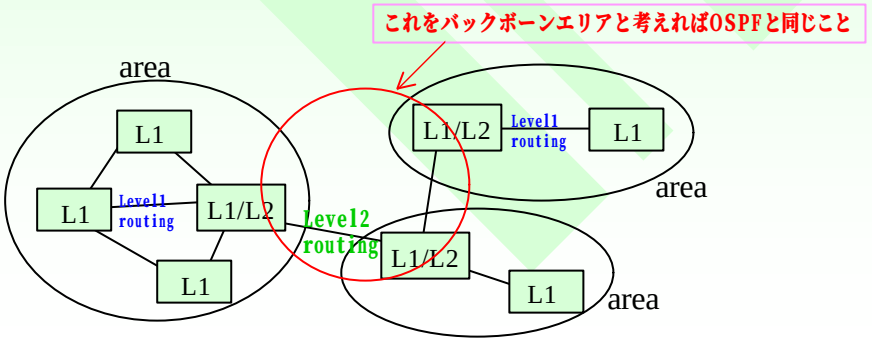
```
int xxx
 ip router isis
```

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
  - ネットワーク的にも、ノウハウ的にも

特別にIS-ISでメリットが見あたるわけでもないのでOSPFのほうがいい

(2) BGPのシステム設計論

BGPとOSPFの比較(1)

OSPF	BGP
リンクステート型プロトコル 状態変更毎にLSA，連鎖伝播	パスベクター型プロトコル 状態変更毎にUPDATE，連鎖伝播
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 生死とパス属性に着目
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
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	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
30分でリフレッシュ--flooding	リフレッシュなし

BGPとOSPFの比較(2)

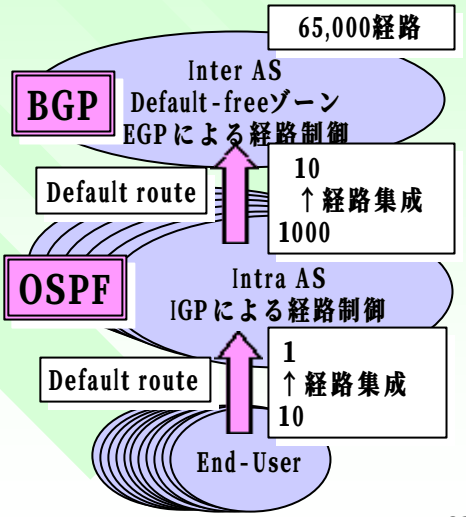
OSPF	BGP
基本的に、OSPFを起動した隣接ルータ全てと経路交換	明示的に定義した隣接ルータのみと経路交換
経路個別にポリシの付加は不可	経路個別にポリシ付加が可能 →パス属性値として プリフィクスに付加

AS (Autonomous System)

- 単一のルーティングポリシで運用される範囲
- 簡単にいえば、ひとつのISP。
- 16ビット (1～65535)の番号空間を持つ
  - BIGLOBE (AS2518)　OCN (AS4713)
  - 64512～65535はプライベートAS

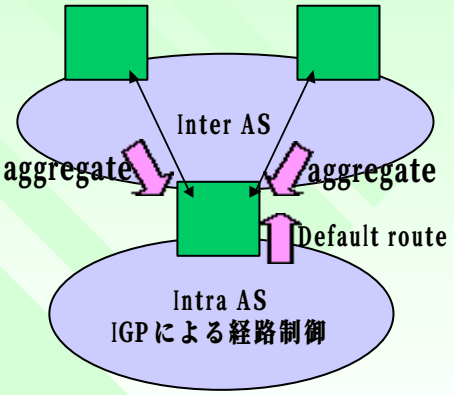
The Internetにおける階層的経路制御

- Inter AS, Intra AS, End-Userの３階層
- それぞれの境界で経路集成=情報量の縮退
- 上は細かいことを気にしない，下は分からないものはゲートウェイに投げる。



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

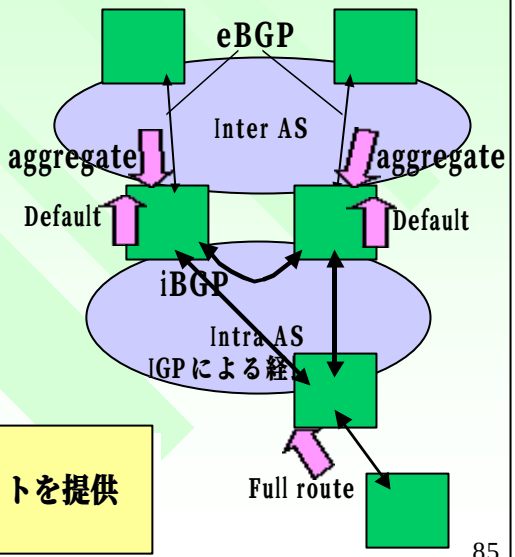
２つのボーダルータを置く

- 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？

↓

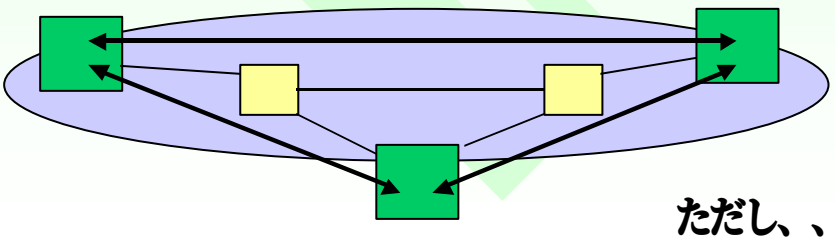
iBGPの確立

次の課題：
BGP加入者にトランジットを提供



iBGPの注意点(1)

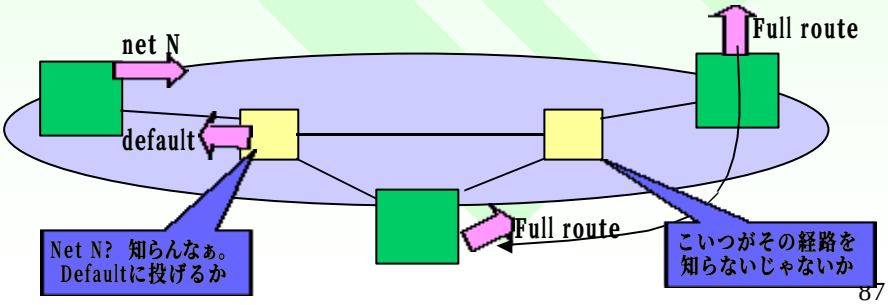
- eBGPは直接隣接を必要とするが、iBGPは離れていても確立可能
- iBGPは全てのボーダルータとセッションを張る必要がある



ただし、、

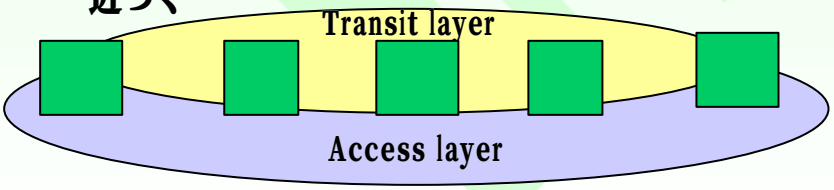
iBGPの注意点(2)

- 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



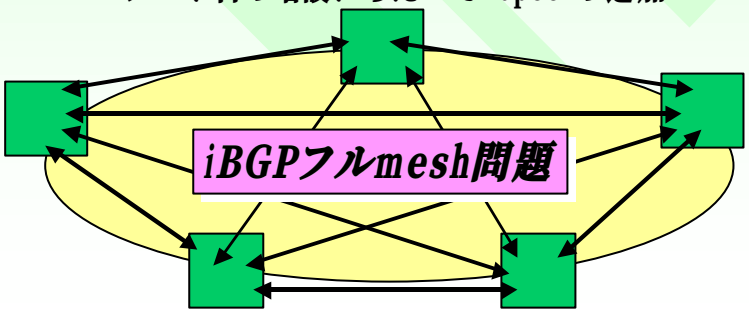
iBGPシステムの解

- No synchronization
  - IGP synchronizationの縛りを解くコマンド(c社)
- 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



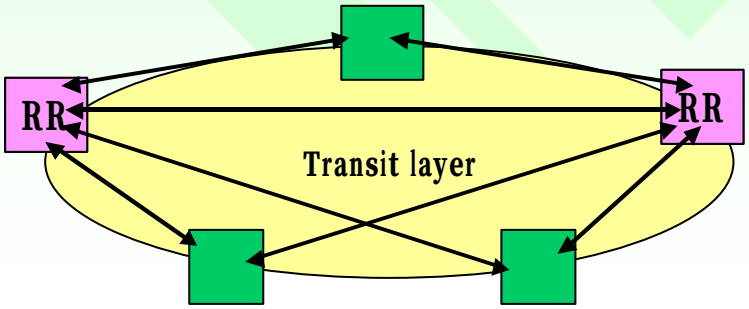
iBGPシステムのスケーラビリティ

- 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



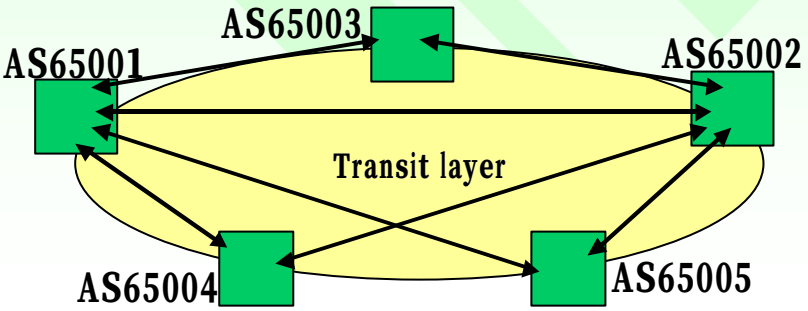
iBGPフルmesh問題解決策(1)

- iBGPルートリフレクタ(RR)
  - 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



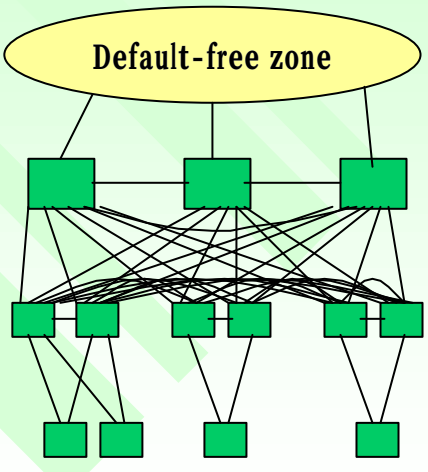
iBGPフルmesh問題解決策(2)

- BGPコンフェデレーション
  - 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



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

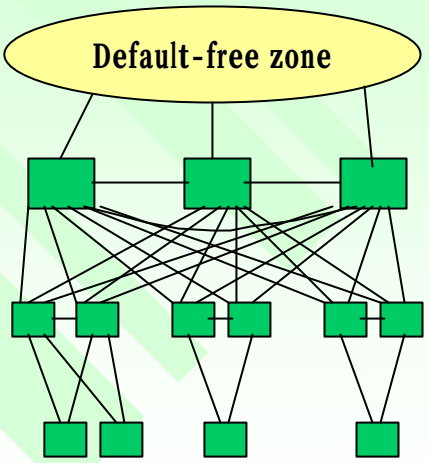


AS内BGPスケーラビリティ問題の実際 —RRによる解法

- 階層的RRの導入

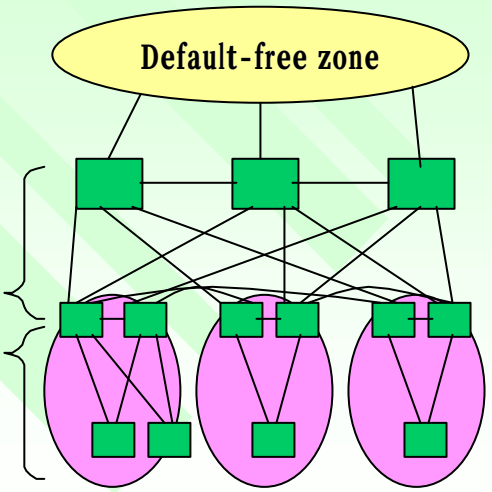
POPコアルータに対してRR→

加入者集線ルータに対してRR→



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加入者主要ルータとの間にiBGPを設定
- IGPは分割，単一どちらでもOK



eBGPスケーラビリティの問題(1)

- 経路数
  - 65,000 -- CIDR Report by Tony Bates**
  - 所要メモリサイズに影響
    - » 128MBで大丈夫，64MBでは足りない局面が多い
- Peerの数
  - IXで多数のpeerを張るとメモリ所要に影響
  - NSPIXP2接続ルータ（50peer程度+upsteam）で10MB程度余分に消費

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

eBGPスケーラビリティの問題(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スケーラビリティの問題(3)

- ポリシ変更の反映
  - 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

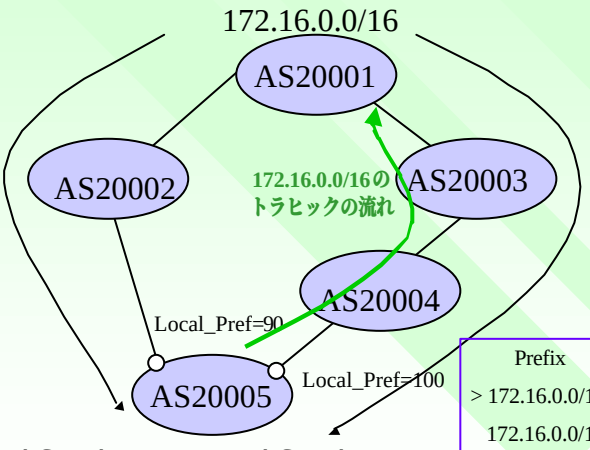
トラヒックバランス，ポリシ実現(1)

- 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

トラヒックバランス，ポリシ実現(2)

- **パス属性値（調整可能なもののみ）**
  - – Local Preference
    - » 設計者意図の優先順位付け
  - – 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
  - – MED
    - » 隣接する同一ASの複数peerの優先度
  - – 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

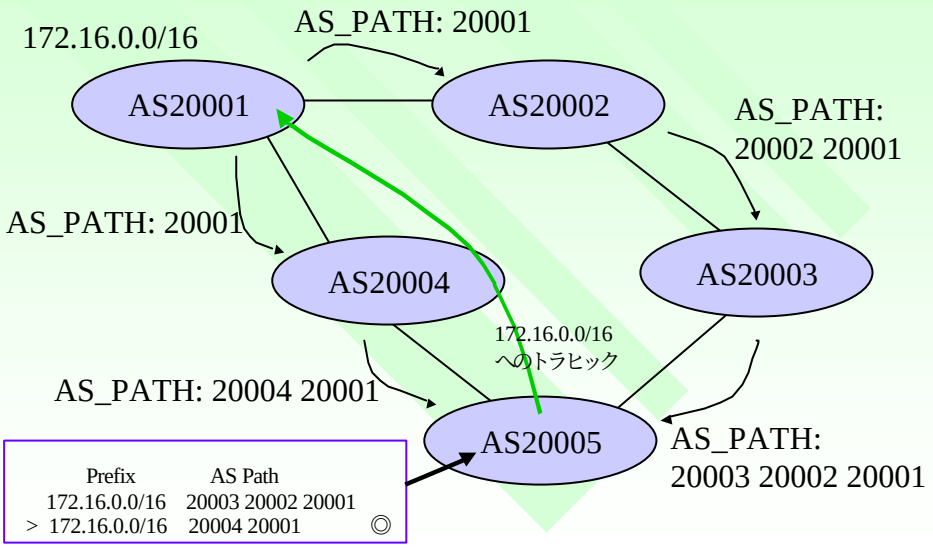
Local Preference



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。

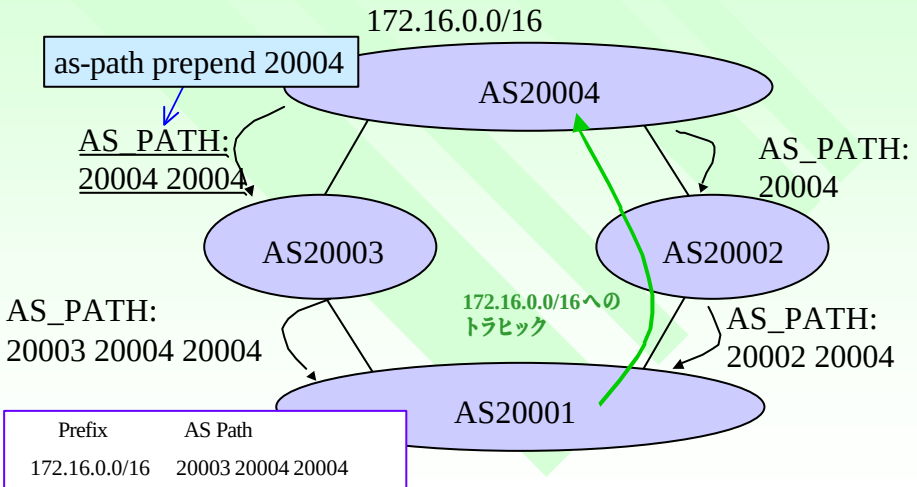
Prefix	AS Path	LocPref	
> 172.16.0.0/16	20004 20003 20001	100	◎
172.16.0.0/16	20002 20001	90	

AS_PATH

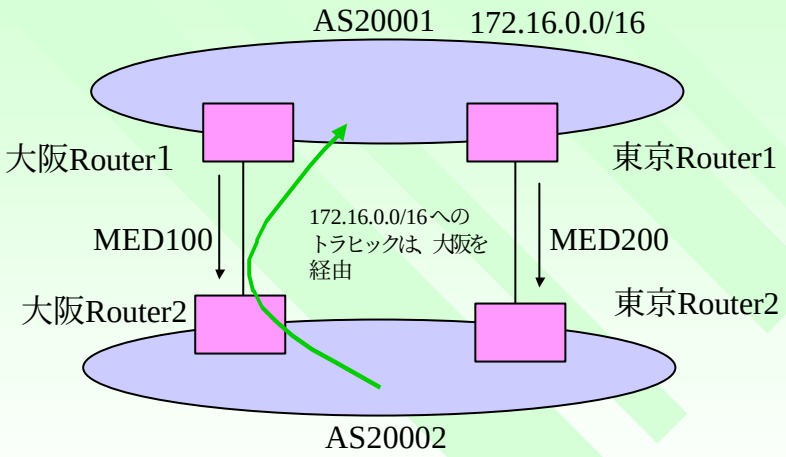


AS Path Prepend

自分のASを余計につけて、AS_PATH_lengthを長く見せるテクニック



MED



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

例えばPrivate PeeringとIXで接続しているISPにおいて、Private Peeringを優先するために使用したりする

Community

- ３２ビットの整数値，透過型属性
  - ただし、数値自体にプロトコル上意味はなし
- 経路情報を受領したAS，ルータで何らかの作用させる
- 一般的な利用法
  - New-format – 32ビットを16ビットずつに二分
    - » 2518:1000
  - 上位 – ターゲットAS
  - 下位 – ターゲットASでの動作

Community

- 例1：RFC1998 – MCI(現C&W)の手法
  - 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！
- 例2：C&W IDC が採用している手法
  - 4694:10ab　　a: 隣のAS識別番号
    b:　AS-path Prependの数(1～3)
  - トラヒック調整の柔軟性が非常に高い！！

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

トラヒックバランス，ポリシ実現まとめ

- 大事なのはoutboundよりもinboundのトラヒック
  - ユーザ向けのコンテンツトラヒック
  - Outboundは受領した経路情報に対するポリシ実装だが、inboundトラヒックは目的対地のポリシに大きく左右される
  - As-path prepend, community の駆使して、逐一調整

(3) Jessica's I-D の紹介

Internet Draft: draft-yu-routing-scaling-01
“Scalable Routing Design Principles”
「規模対応性の高い経路制御設計の指針」

概要説明

- draft-yu-routing-scaling-01
- 著者：Jessica Yu, UUnet
- IIJ近藤邦昭氏，友近，前村で和訳
  - 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。

経路制御設計の一般的目的

- スケーラビリティが高いこと
- 冗長性があり、かつ、強靭であること
- 妥当な収束時間であること
- 経路情報が完全であること
- 経路制御ポリシが実用的かつ管理可能であること

今日の大規模ネットワークの特徴（著者の想定=米国Tier1の現状）

- 数百ノード，数千ユーザ，ほとんどがBGP接続
- 冗長性確保の結果複雑なトポロジ
- フルルート(60,000経路—現在は65,000)の伝搬
- 顧客集線ルータには、数百のユーザがつながることも。。。

問題点1 – 一般論

- ルータのリソース消費
  - 経路数過多
  - 不安定なネットワークのflapping
    - » 経路更新，経路削除を繰り返すことでルータが疲弊
- BGPハンドリング上の問題
  - Default-freeなIXにおけるpeer→方路過多でメモリ圧迫
  - プリフィクスフィルタリング
  - 顧客集線ルータでのpeer過多
- IGPのハンドリング
  - 大規模ネットワークにおける巨大なLSDB

問題点2 – IGPの問題

- IGP – OSPF, IS-IS
  - リンクステートDBの肥大化
  - フラッディング（一斉広告）
    - » 一台のルータの状態変化が全ルータへのLSA伝搬となる
    - » OSPFにおける30分に一度のリフレッシュ
  - 経路計算の複雑化
  - 過負荷の悪循環
    - » 過負荷でHELLOパケットを受け損なったら、LSA発生，復旧してもまた発生
    - » 過負荷の上塗りへ

問題点3 – BGPの問題

- iBGPフルメッシュ問題
  - iBGPでは、BGPノードは全て直接peerを張る必要がある
    - » セッション過多となる
- Flapping による update/withdrawの連続
  - CPUリソース消費
- プリフィクスフィルタリング

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
- out-of-band 経路処理

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
- 二層以上の階層化?
  - IS-ISでは多重階層にトライ
- 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
- 問題・障害の局所化効果
- 経路の集成
- BGP Confederation によるIGPドメインの分割

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

- Out-Of-Band 経路制御 – Route Server の導入
- 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
- 可能であれば out-of-band な経路制御ポリシプロセスを使う。

out-of-band 経路処理

- いわゆる「ルータ」の２つの機能
  - Routing ---- 経路選択，ポリシの処理，経路表完全性の維持
  - Forwarding ---- 経路表に基づくパケットの転送
- トラヒック処理と経路演算を別デバイスで実施
  - Routing --- ルートサーバを使い、できあがった経路表をルータに供給
  - Forwording --- 経路制御から解放されたルータが頑張る

読後雑感

- 経路制御のスケーラビリティに関して非常に良くまとめられた名著
- 米国Tier1と日本の環境の違い
  - BGP加入者が非常に多い
- 階層化，区画化　–当然といえば当然だが
  - 自分たちがやっていることは正しい
- ルートサーバは。。。
  - まだまだ使える気がしない
  - 使えるようになれば、ベンダーとしても朗報か？

(4)static-to-bgpの設計の実際

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

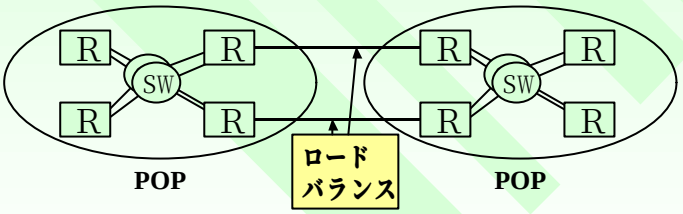
- AS4713(OCN)では、OSPFの経路数が非常に増えていた
  - 90%強がexternal経路。これはcustomerへのstaticの経路をOSPFにredistributeしていた経路
- あまり効率よくaggregateできない
  - JPNICおかわり制限

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

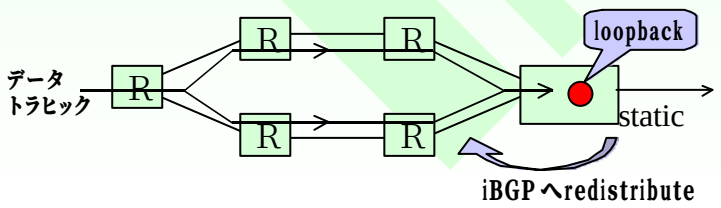
- OSPFを分割する（リンク部分で）
  - Confederation等
    - » ロードバランス困難
      - 一つ手前のルータでバランスさせないといけない
    - » サービス停止、運用変更
- OSPFに変えてIS-ISにする
  - 設計・運用ノウハウが足りない
  - 実際効くのかどうかわからない
- その他
- static経路をOSPFでなく直接iBGPにredistributeさせる

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）
  - 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

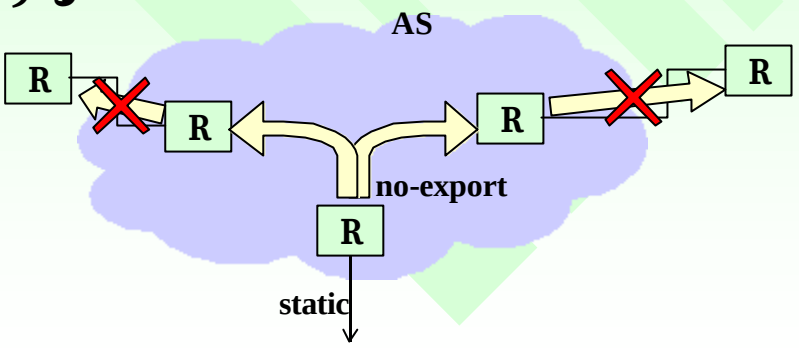
仕組み

- 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
- 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



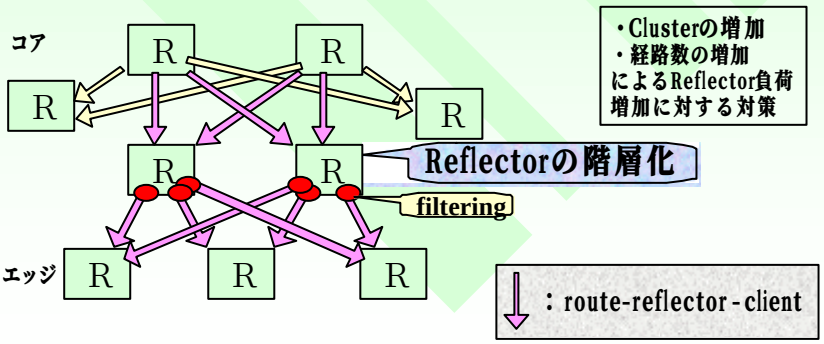
その他付随するテクニック(1)

- 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



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

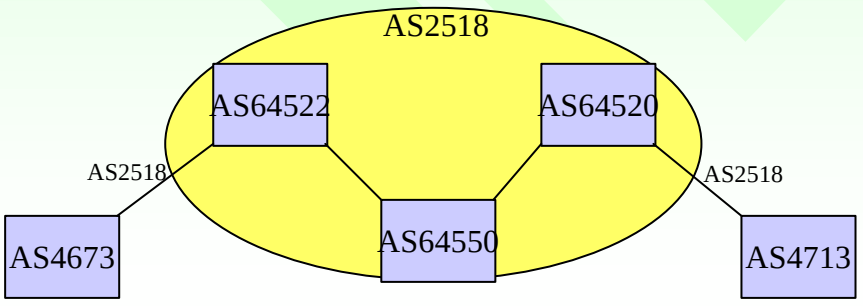
(5) Confederationの設計の実際

Confederationの一般的な取り扱われ方

- iBGPフルmesh解消の手だて -- Confederation or Route Reflector??
  - iBGPで知った経路は他のiBGP peerには広告しない
    → 全てのBGPスピーカとpeerする必要がある。
- Route Reflector
  - 支配下のBGPスピーカに対してiBGPで知った経路を広告するしくみ
    - » リフレクタ同士をフルmeshにすれば、全てのBGPスピーカが全経路を持つ

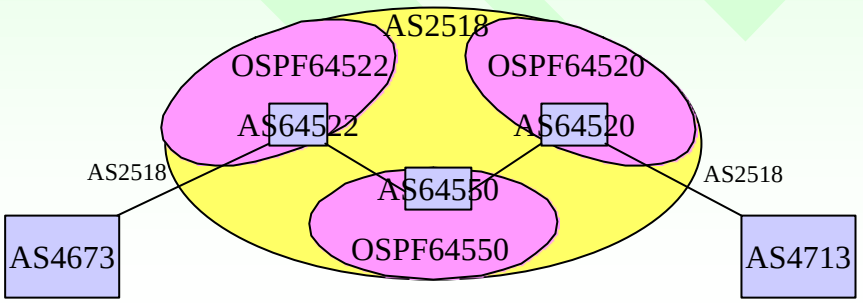
BGP Confederationとは?

- 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 64551
  bgp confederation identifier 2518
  bgp confederation peers 64520 64521 64522 64523
    64530 64551 64552 64553 64570 64580 64582
  network …….
```

BIGLOBEにおける Confederation移行の動機

- OSPFプロセスの肥大化によって、アクセスポイントに設置した小さなルータでメモリ不足，経路制御不全の危機
- 未来永劫、経路不全で悩まないデザインへの移行
  - 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に。。
  - 東阪に外部接続，and more...
- 障害の局所化
  - 全網規模になるのだけは避けたい

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。

BIGLOBEにおける経路制御設計

- subAS毎にnetwork定義をするのはとてもじゃないけどできない
  - OSPFをredistributeして、aggregateする
- 中央にaggregate generator, 対外接続ルータでspecificをfilter out
- nexthop-self でeBGP的に処理

Confederationに移行して嬉しかったこと

- ちゃんと動く!!
  - Internet Routing Architectureの「推奨デザイン」じゃなくても大丈夫
- 対外接続ルータは、ルータ一つで1ASを割り当て、OSPFを起動していない
  - BGPハンドリングだけに専念させられる
- OSPFで一切悩んでない!!

Confederationによる IGP分割の不便な点

- confed内のsubASに関してhop数評価をしない
  - 別のattributeでコントロールする必要あり
    - » IOS11.1(20)CC以降、MEDでコントロールが可能
- subASの毎にルーティングポリシが必要
- subAS毎にdefault route
  - subASボーダルータを複数置く場合。。。

移行の実際

- まずはsubASボーダルータとなるルータとBGPを張る
  - まだまだOSPFでも経路制御可能
- OSPFプロセスの分割
- 同時にOSPF経路のredistribution

雑感

- iBGPフルmesh解消策としてRoute Reflectorよりも。。。
  - 「リフレクタiBGPフルmesh問題」が。。。。
    - » IGPを単一の場合でも、confedのほうが楽かも??
- もう少し「AS分割利用モード」があっても。。。
  - confed内でASpath評価があると嬉しい

結論

- スケーラビリティ解決に朗報!!!
- ちゃんと動く!!!
- もっともっと利用しましょう!!!
  - OSPFのスケーラビリティで困っている人
  - mergerでAS統合が課題になっている人

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
IPネットワークサービス部 サービス開発担当
友近 剛史 tomo@byd.ocn.ad.jp

NECマルチメディアサービス構築運営本部
BIGLOBE/C&Cインターネットサービスmesh
前村 昌紀　maem@mesh.ad.jp
maem@maem.org