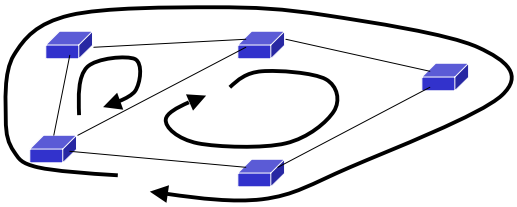
LAN　Switch技術
～冗長化手法とループ防止～

www.poweredcom.net

安藤　雅人

Ethernetにおけるループ発生の発生と弊害

- リンクやノードの故障の影響を防ぐ為に、イーサネットスイッチを冗長を持たせて接続すると、ループ部分ができてしまう。

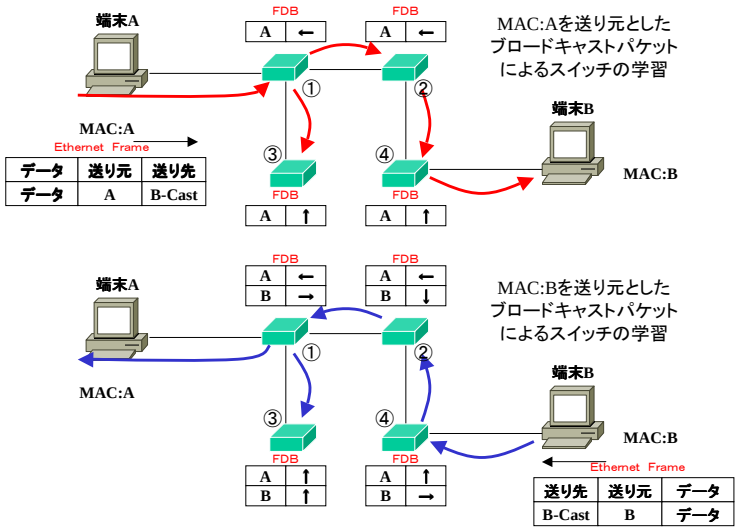


- 何故ループが駄目か？
  （1）**FDB（Forwarding Data Base=MAC学習テーブル）が狂う。**
  ユニキャスト通信が出来なくなる。
  （2）**フレームが増殖する。（トラフィック圧迫）**
  帯域が圧迫される
  アプリケーションへの悪影響（システムダウンする場合もある）

→絶対にループは発生させてはならない！（一瞬でも）
冗長を組みながら、ループを防止する方法を紹介

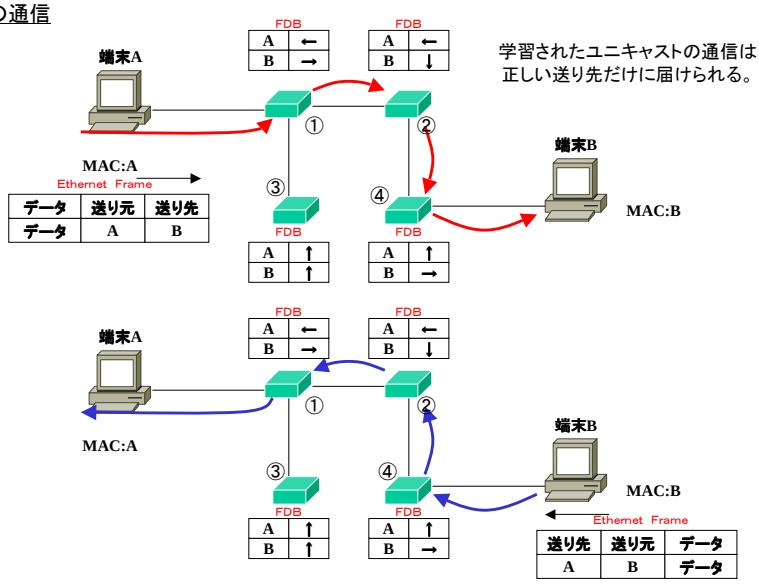
正常時のスイッチ網のFDB学習状況１

通常時の学習



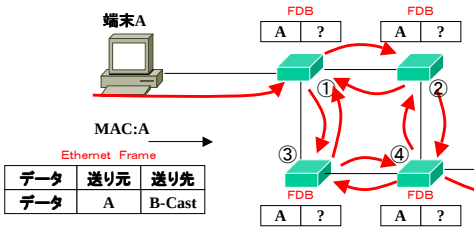
正常時のスイッチ網のFDB学習状況２

通常時の通信



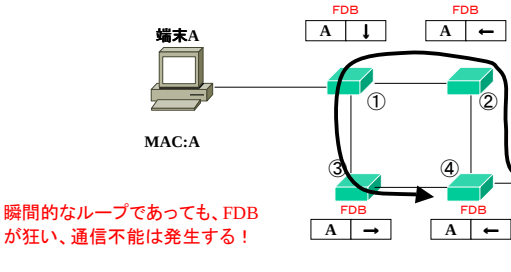
ループ発生時のFDB1

ループ発生時の学習



ループがあるネットワークでは、スイッチにて学習されている内容は時間とともに変化する、狂ったFDBとなる。これはCスイッチのFDB学習機構に負荷を与える事にもなる。

ループ発生時の通信1

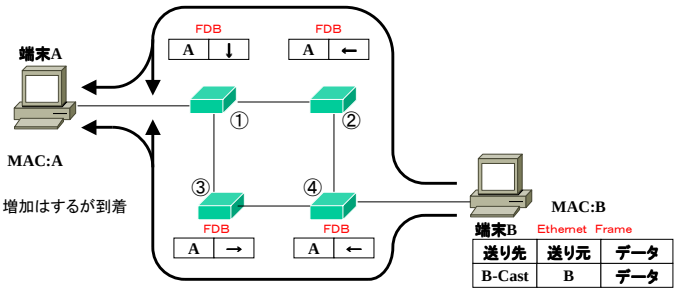


狂ったFDBではユニキャストフレームを正常に送り届ける事は出来ない。
※マルチキャスト、ブロードキャスト、アンノウンユニキャストはFDBを参照しないで転送する為届くことは届く。

瞬間的なループであっても、FDBが狂い、通信不能は発生する！

ループ発生時のFDB2

ループ発生時の通信2

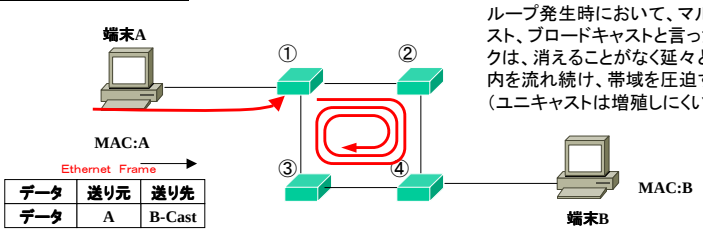


ループ時でもブロードキャストやマルチキャストのフレームは送り先に一応到達する場合が多い。（フラッディングはFDBと関係なく転送されるため）

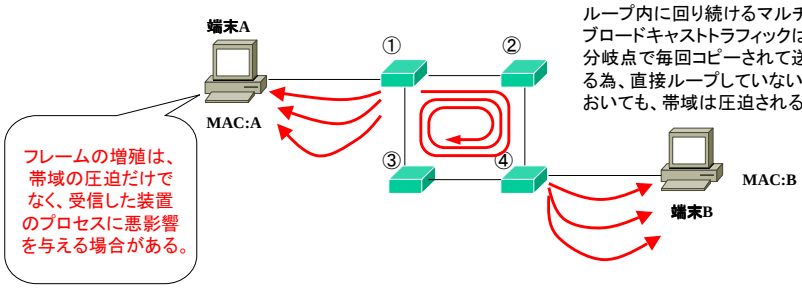
ユニキャストは通信不能となるが、マルチキャストやブロードキャストの通信は転送されているように見える事がある。（OSPFのネイバーは見えるが、pingが通らないなど）

ループによる帯域圧迫

ループによる増殖と帯域圧迫



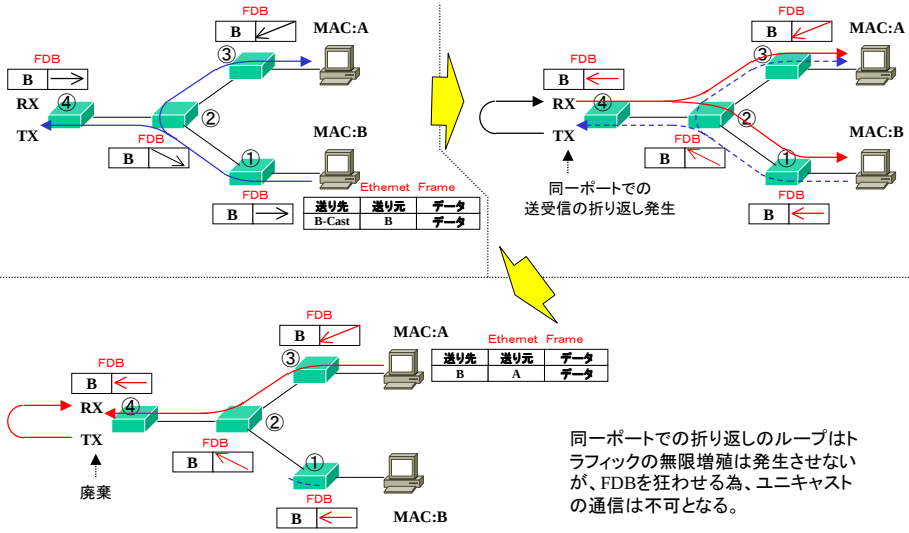
ループ発生時において、マルチキャスト、ブロードキャストと言ったトラフィックは、消えることがなく延々とループ内を流れ続け、帯域を圧迫する。
（ユニキャストは増殖しにくい）



ループ内に回り続けるマルチキャスト、ブロードキャストトラフィックはノードの分岐点で毎回コピーされて送出される為、直接ループしていない箇所においても、帯域は圧迫される。

フレームの増殖は、帯域の圧迫だけでなく、受信した装置のプロセスに悪影響を与える場合がある。

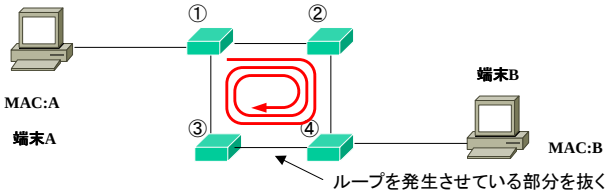
同一ポートでの折り返しのループ



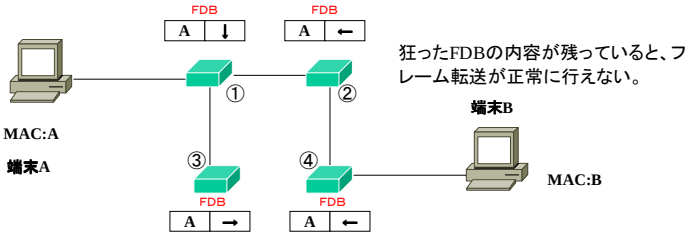
同一ポートでの折り返しのループはトラフィックの無限増殖は発生させないが、FDBを狂わせる為、ユニキャストの通信は不可となる。

ループが発生したら何をすべきか？（手動の場合）

(1)ループを構成している部分のリンクを切断したり、スイッチの転送を止める



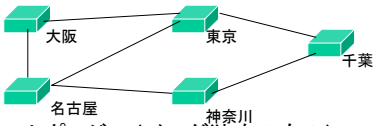
(2)FDBの内容を一度**フラッシュ**する。（フラッシュしなければ、FDBのエージアウトを待つか、内容が上書きされるのを待たなければ正常な通信が行えない）



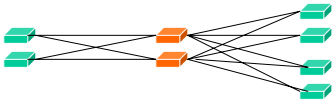
※何もしなければFDBのエージアウトは5分で AgeOutするのが一般的。(設定によって変更可能な物が多い)

イーサネット網における冗長方式の分類

- ネットワーク冗長（STPファミリ、RPR、VPLS、リング型冗長など）
  ネットワークとして、冗長性を確保する。
  爆撃やテロなどに対しても比較的強い。



- ノード冗長化=メッシュトポロジー（ベンダ独自のもの）
  コアのスイッチの冗長化（二重化）
  装置の信頼性、伝送路の信頼性を補う為に主に用いられる。



- ノード内完全冗長+リンクアグリゲーション
  装置内部を完全に冗長化し、冗長単位で交換可能とする。伝送路はリンクアグリゲーションなどで冗長性を確保。

冗長なネットワークでのループを防止する三つの機構

1. ループフリーな論理トポロジーを維持する機構
   STPをはじめとする、様々な論理ネットワーク維持機構

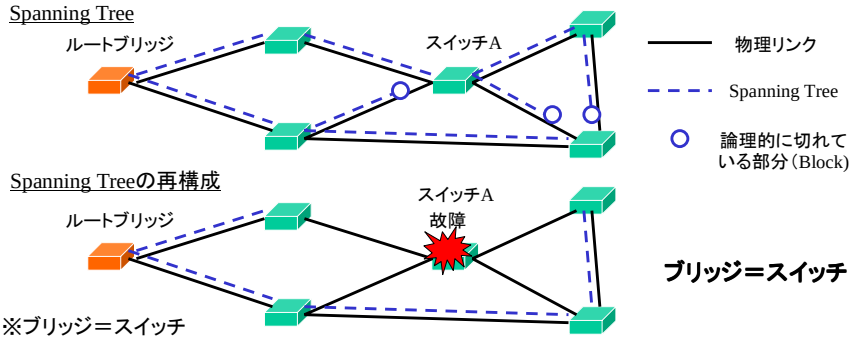
2. ループが発生した場合ループを検出し、論理トポロジーに働きかける機構
   ループの検出を行い、網のトポロジーに働きかける機構

3. ループしたフレームを検出し、フレームを破棄する機構
   TTLを利用したフレーム破棄、フィルタリング

Spanning Tree Protocol
ループフリーな論理トポロジーを維持する機構

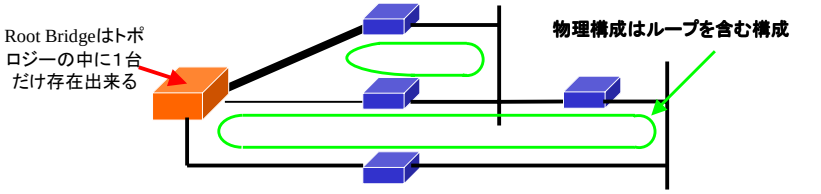
STP（Spanning Tree Protocol）IEEE 802.1D

- ループフリーな論理トポロジーを維持する機構の一つ。
- IEEE 802.1D標準
- 中心となるスイッチはルートブリッジと呼ばれそこから木のように枝分かれしていくので、Spanning Tree (広がる木)と呼ばれる。
- 木は、枝分かれはするが、一度分かれた枝が先で再度くっつく事は（普通）ないので、ループが発生するようなトポロジーとならない。
- 利用中のリンクが断したり、ノードが停止したら、論理トポロジーを自動的に再構成



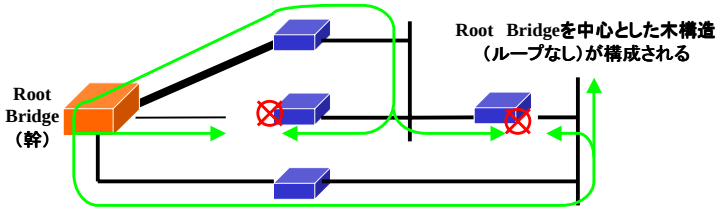
STPでのトポロジーの構築（概要）

（1）木構造の中心となるルートブリッジをどれにするか選ぶ



Root Bridge（木構造の中心となるブリッジ）：このブリッジが中心となるように、トポロジーを構成する。

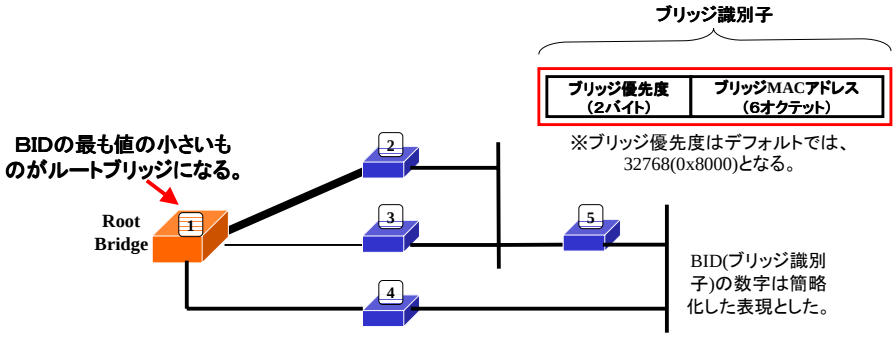
（2）ブロック（転送禁止）にするポートどれにするか選ぶ
（＝転送するポートを選ぶ）



※これらの動作をSTPではスイッチ間でBPDUと呼ばれるパケットの交換し自動的に行う。

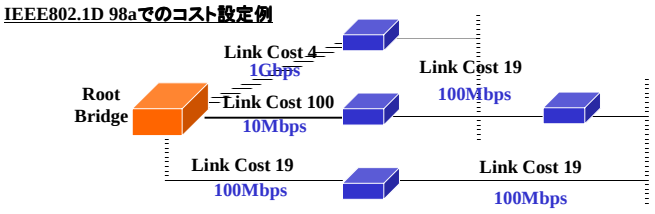
STPでのトポロジーの構築（ルートブリッジの選択）

- ルートブリッジの選択とブリッジ識別子（ブリッジID）
  - ブリッジはそれぞれ固有のブリッジ識別子（BID）を持ちその値のもっとも小さいものがルートブリッジになる。
  - ブリッジ識別子は、2オクテットのブリッジ優先度とブリッジMACアドレス（6オクテット）をつなげたものになる。
  - ブリッジ優先度が同じでも、ブリッジMACアドレスはユニークなので、ブリッジ識別子は必ずユニークとなる。



STPでのトポロジーの構築（転送禁止ポートの選択）

- 転送禁止ポート（転送ポート）の選択とリンクコスト
  - 転送/非転送のポートの選択は、リンクコストの計算によって決定される。
  - 一般的にはポートの速度に応じてリンクのコストを付ける。
  - メディアの高速化に伴い、推奨コストも変化して来ている。
    （かつては1000/速度Mbpsで設定していた）
    16bitショート法(IEEE802.1[IEEE98a])、32bitロング法(IEEE802.1t)
  - Link Aggregationを組む場合には、速度によるリンクコスト/Link Aggregationのコストとなる。(IEEE802.1t)
    ※GbE(1Gbps)のコストを20.000とすると、2本のGbEで組まれたLink Aggregationのリンクコストは、10.000となる。



STPでのトポロジーの構築（リンクコスト表）

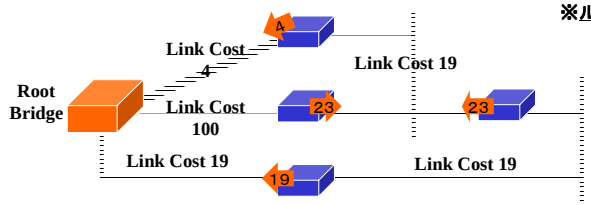
リンクコストの推奨値

データレート	IEEE 802.1D 98a（ショート法） 推奨リンクコスト範囲（推奨値）	IEEE802.1t（ロング法） 推奨リンクコスト範囲（推奨値）
4Mbps	100～1000(250)	
10Mbps	50～600(100)	200.000-20.000.000(2.000.000)
16Mbps	40～400(62)	
100Mbps	10～60(19)	20.000-2.000.000(200.000)
1Gbps	3～10(4)	2.000-200.000(20.000)
10Gbps	1～5(2)	200-20.000(2.000)
100Gbps		20-2.000(200)
1Tbps		2-200(20)
10Tbps		1-20(2)

STPでのトポロジーの構築（転送禁止ポートの選択）

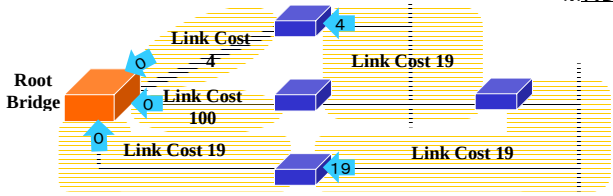
- 各スイッチでリンクコストの積算が最も小さくなるポートを選ぶ（スイッチの出口）

※ルートポート（後述）



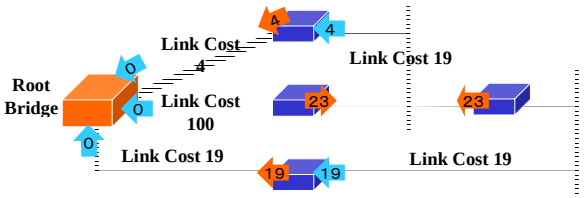
- 各リンクでリンクコストの積算が最も小さくなる接続点を選ぶ（リンクの出口）

※代表ポート（後述）

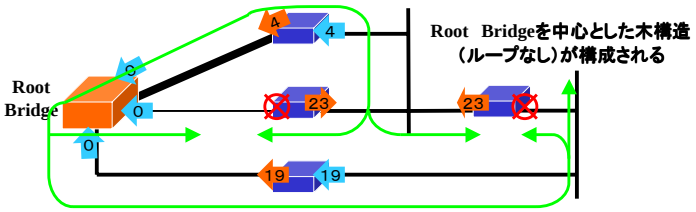


STPでのトポロジーの構築（転送禁止ポートの選択）

- ルートポートと代表ポートは転送状態にする。

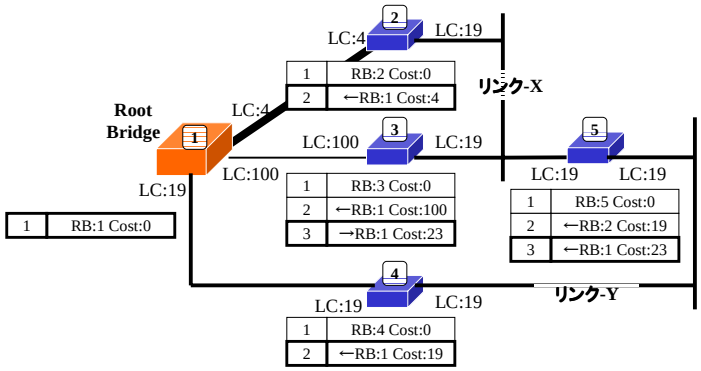


- ルートポートあるいは、代表ポートにならなかったポートはブロック（非転送状態）にする。



STPでのBPDUの交換

- Spanning Treeの構築はConfiguration BPDUの交換によって行われる。
- 各スイッチは、自身が知っている最も条件のいいルートへのコストをBPDUを使って広報する。
- 自身が知っているよりもより条件の良いコストを示すBPDUを受け取ったら受け取ったポートのコストを足して自身が採用するとともに、他のスイッチへその情報を広報する。



STP　BPDU

- BPDU（Bridge Protocol Data Unit）
  - Spanning Treeの作成維持管理を行う。
  - スイッチ（ブリッジ）間で通常2秒間隔で送信、交換される。
- 二種類のBPDU
  - Configuration BPDU（Hello パケットとも呼ばれる）
  - Topology Change Notification BPDU

宛先MAC 01:80:C2:00:00:00
LLC　TYPE1
DSAP,SSAP 0x42

Configuration BPDU（トポロジー構築に使用）

Protocol ID=0000h	2
Protocol Version ID=00h	1
BPDU Type=00000000b	1
Flags	1
Root ID	8
Root Path Cost	4
Bridge ID	8
Port ID	2
Message Age	2
Max Age	2
Hello Time	2
Forward Delay	2

発信するスイッチが知りえる最もブリッジIDが小さいスイッチのID（Root Bridgeと思われるブリッジID）

Root BridgeまでのPathコスト

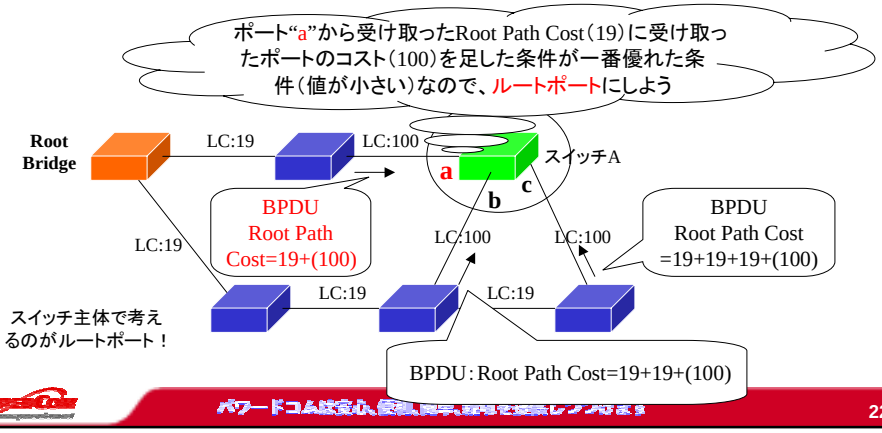
Topology Change Notification BPDU
(トポロジー変更時に使用)

Protocol ID=0000h	2
Protocol Version ID=00h	1
BPDU Type=10000000b	1

Flagには、Topology Change FlagとTopology Change Acknowledgement Flagの二種類が定義されている。

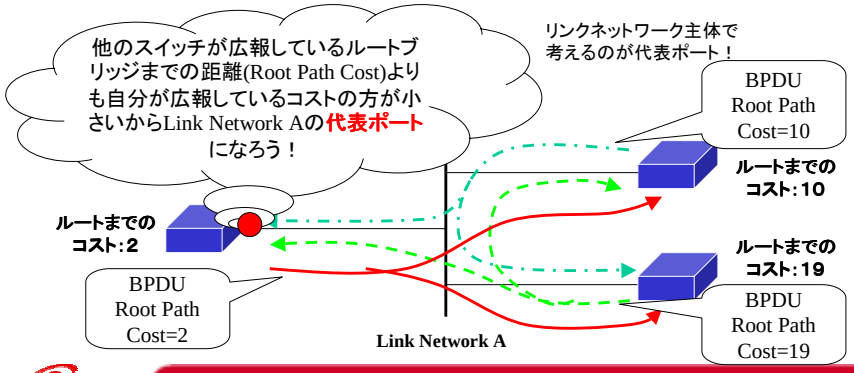
STPでのトポロジーの構築（BPDUでのポートの役割の決定）

- ルートポート（Root Port）
  - スイッチからルートブリッジに到達するのに利用出来るポートを1つだけにする、このポートの事をルートポートと言う。
  - スイッチの中で最もルートブリッジに少ないパスコスト（リンクコストの合計）で行けるポートがルートポートとなる。
  - スイッチが受信したBPDUにその受け取ったポートのコストを足した結果スイッチの中で一番コストが小さくなったポートが一つルートポートとして選択される。



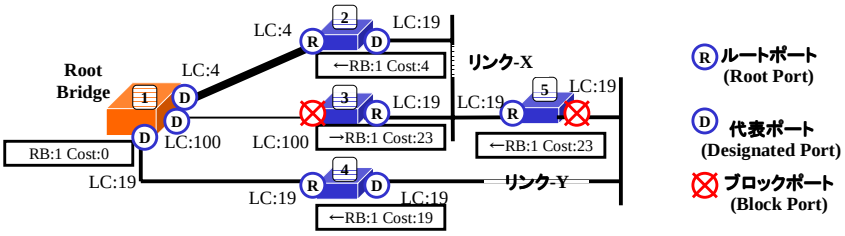
STPでのトポロジーの構築（BPDUでのポートの役割の決定）

- 代表ポート(Designated Port)
  - リンクからルートブリッジに到達するのに利用出来るポートを1つだけにする、このポートの事を代表ポートと言う。
  - ルートブリッジのポートは全て代表ポートとなる。（ルートブリッジ以外の代表ポートを持つブリッジを代表ブリッジ=Designated Bridgeと呼ぶ）
  - リンクに流れているBPDUをそのポートにて観察しそのブリッジが持っているコストより優れたものが流れていない場合に、そのリンクからルートブリッジに到達するのにそのポートが一番有利な条件と考え、そのポートを代表ポートとする。

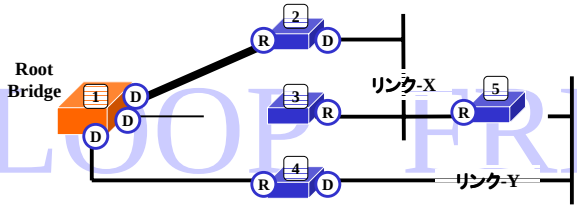


STPでのトポロジーの構築（ポートの役割の決定）

- ルートポートでも代表ポートでもないポートはブロッキング状態（非転送）になる。

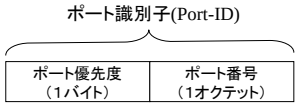


- ルートポートと代表ポートだけがフォワーディング状態（転送）になる。
- フォワーディング状態にあるポートとリンクだけを繋ぐとループのない木構造になっている。

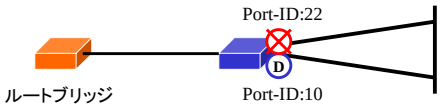


STPでのトポロジー構築

- ポート識別子
  - ブリッジ内のポートはそれぞれシャーシ内で固有のポート識別子を持つ。
  - ポート識別子は、1オクテットのポート優先度と1オクテットポート番号をつなげたものになる。
  - 代表ポートの選択がパスコストの比較だけではつかなかった時にこの数値が低い方のポートが代表ポートになる。



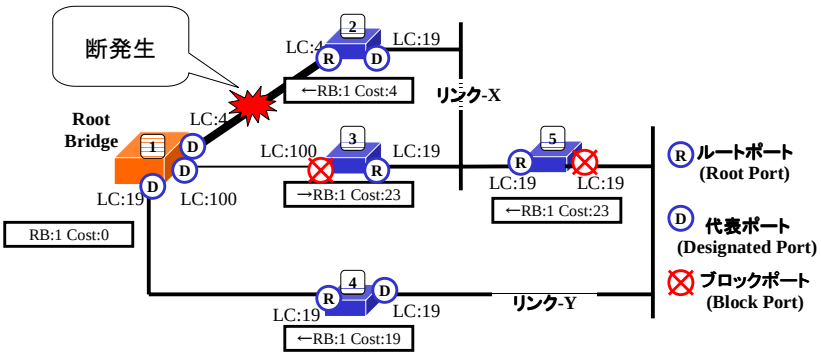
※ポート優先度はデフォルトでは、128(0x80)となっている。



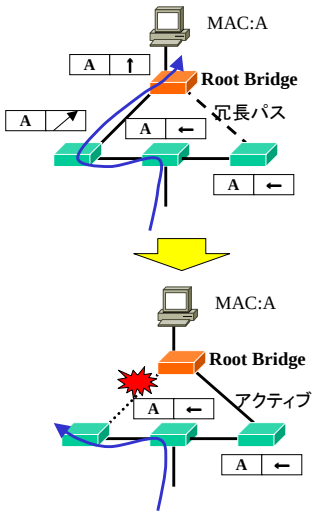
※代表ポートの選択でパスコストが同じ場合（同シャーシの場合など）Port-IDが小さいほうが、代表ポートとなる。

STPでのトポロジー再構築

- 障害発生の場合
  - スイッチ2はルートポートの断を検出すると、Spanning Treeの再計算を開始する
  - ここでの計算の方法は通常のSTP構築と同じ



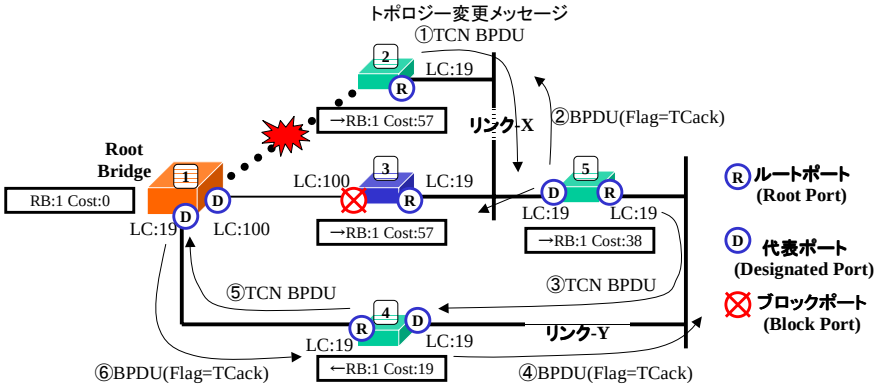
STPでのFDBフラッシュ（消しこみ）の必要性



- 何故トポロジー変更中にFlag=TCNのBPDUをRoot Bridgeが送信して、FDBの内容の消しこみを行う必要があるのか？
- STPのトポロジー変更が発生した後で、FDBに以前学習した内容が残っていると、正常な通信が出来ない。
- FDBの中身を消せば通信出来るようになる。

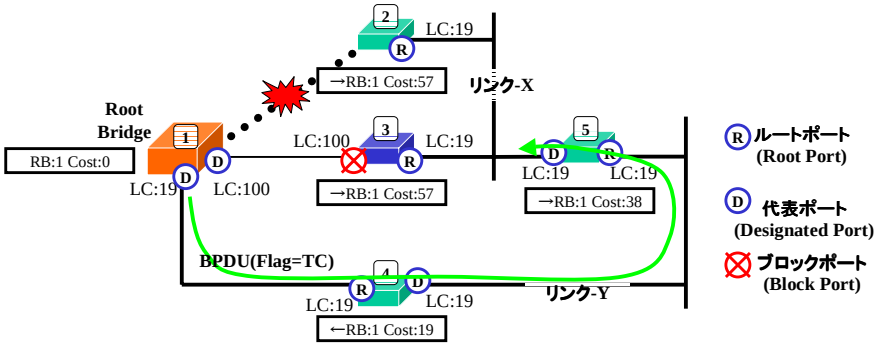
STPでのFDBフラッシュ

- STP構築以外に網全体のFDBをフラッシュする為の機構が必要となる。
- トポロジーの変更を検出したスイッチは、ルートポートより、Topology Change Notification BPDU(TCN BPDU)を送出する。
- 代表ポートはTCN-BPDUを受け取ると、BPDUのFlag=TCack（Topology Change Acknowledgement)を設定し送り返しTCN-BPDUを受信した事を通知し、さらに上位のスイッチにTCN-BPDUを送信する。



STPでのFDBフラッシュ

- TCN BPDUを受信した、ルートブリッジは、一定期の間、送信するConfiguration BPDUをFlag=TC(Topology Change)として送信し、全てのスイッチにトポロジー変更が発生した事を通知する。
- Flag=TCのBPDUを受信したスイッチはフラグが設定されている期間中、FDBの中身をより短い時間でAge Outするようにする。
  - Aging Time(一般には5分)を、Forward Delay(Default=15秒)に変更する事により、速やかにFDBを忘れさせる。



STPでのポート状態

- STPでブロッキングポートが、ルートポートや、代表ポートに変更されたとしてもすぐに転送状態(Forwarding)とはならない。
  - ループ防止
  - 無駄なフラッディングを防ぐ

ルートポートであるとか代表ポートであるとかいったポートの役割とは別に、ポートにはいくつかの状態がある。

- DISABLED状態
  - シャットダウンされているか、電源の入ってない状態。(このポートは使えない)
- BLOCKING状態
  - データフレームの転送を行わない
  - ルートポートや代表ポートになっていないポートはこの状態に落ち着く
  - BPDUの送信は行わないが、BPDUの受信は行っており、その処理も行われる;
  - 電源投入時は全てのポートがこの状態
- LISTENING状態
  - データフレームの転送は行わない
  - BPDUの受信を行う状態、必要であればBPDUの送信も行う
  - Spanning Treeを構築中のスイッチはこの状態にある。

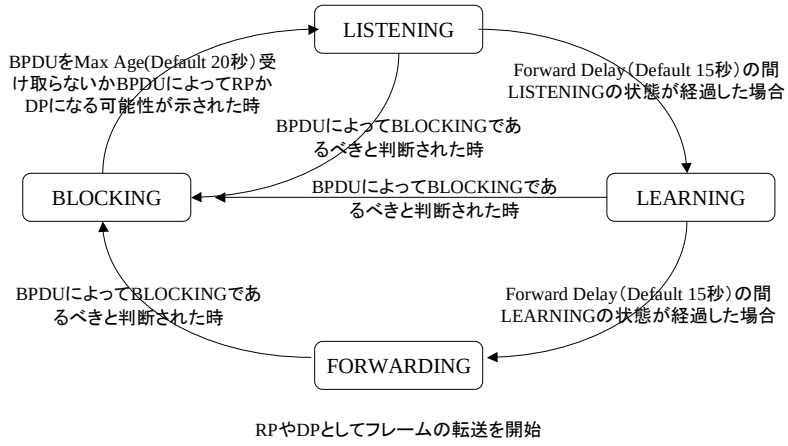
STPでのポート状態

- LEARNING状態
  - 転送を始める前はFDBの内容が空である為そのまま転送をはじめるとフラッディングが多発する。これをおさえる為、転送を開始する前に流れているフレームからFDBの内容の学習を行う。
- FORWARDING状態
  - 通常の転送状態。

	利用可能？	BPDU処理	MAC学習	データ転送
DISABLED状態	×	×	×	×
BLOCKING状態	○	△（受信のみ）	×	×
LISTENING状態	○	○	×	×
LEARNING状態	○	○	○	×
FORWARDING状態	○	○	○	○

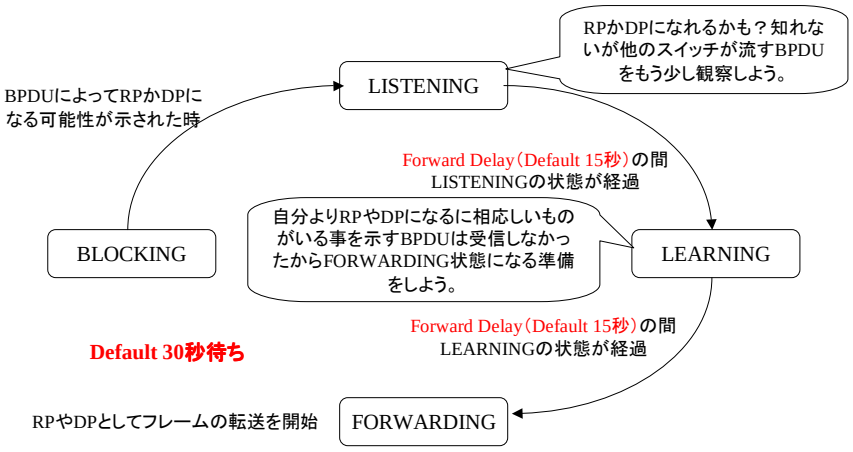
STPでのポート状態

- STPポート状態遷移



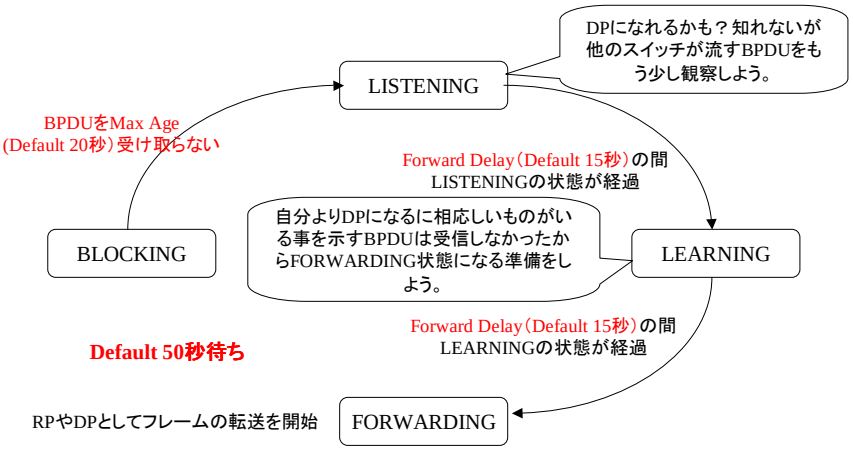
STPでのポート状態

- トポロジー変更により、BLOCKING状態であったポートがルートポート(RP)や代表ポート(DP)に変更され、転送状態になるにはForward Delay x 2待たなくてはならない。
- スイッチの起動時やポートをリンクアップさせてすぐの状態も同様。(スイッチにPCに付けてすぐに通信出来ないのはこれが原因の事もある。)



STPでのポート状態

- 上位のブリッジから、BPDUをMAX Age時間受け取らなかった場合は結果的にMAX Age + (Forward Delay x 2)待たなくては転送を開始出来ない。



STPパラメータの確認

パラメータ	説明	Default値 （設定可能範囲）
Hello time	BPDUの送信間隔	2秒（1-10）
Forward Delay	LISTENNINGやLEARNINGに使う時間	15秒（4-30）
MAX Age	BPDUがタイムアウトする時間	20秒（6-40）
Bridge Priority	スイッチの優先度、小さいほど、ルートブリッジになりやすい	32768（0-65535）
Port Cost	そのポートのコストを示します。小さいほど選択されやすくなる。	速度に応じて設定
Port Priority	ポート間でパスコストが同じだった場合に比較される値、小さいほど選択されやすくなる。	16(0-255)

STPのまとめ

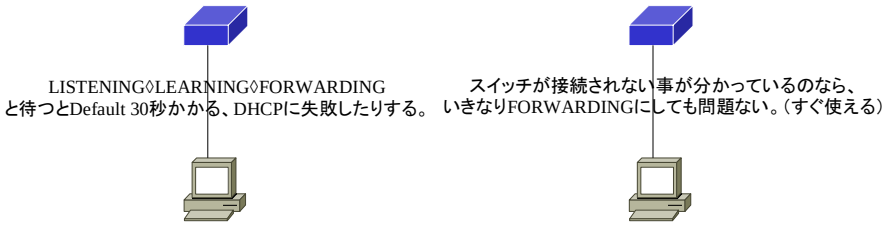
- 通常STPが止まるのは、LISTENING◊LEARNING◊FORWARDINGにかかる、Forward Delay x 2=Default 30秒の時間、ただし、最悪の場合、MAX Age + Forward Delay x 2=Default 50秒の時間止まる。Forward DelayとMAX Ageを設定変更する事も可能。それでも14秒(6+4x2)を切る事は出来ない。
- パラメータはルートブリッジに設定されたものが採用される。
  - 他のスイッチはルートブリッジが流すBPDUに記述されたパラメータを採用する。
- STPは特に何も設定しなくても動作するが最低限ルートブリッジとバックアップでルートブリッジになる装置がネットワークの適切な位置で適切な性能のスイッチになるように設計されなくてはならない。
- STPのパラメータはむやみに変更しない（障害時の解析が大変になる）。変更する時はきちっと設計を行い、全てのスイッチが同じポリシーで動作するようにする事。

- STPのメリット
  - 標準的なプロトコルであり、異ベンダ機器の相互接続が可能になっている。
  - 物理トポロジーを選ばない。

- STPのデメリット
  - 標準のSTPは切り替えに時間がかかる
  - トポロジー全体の事を考えて、オペレーションを行わなくてはならない。
  - PCなどをつなげる場合もForward Delay x 2の時間待たされる。

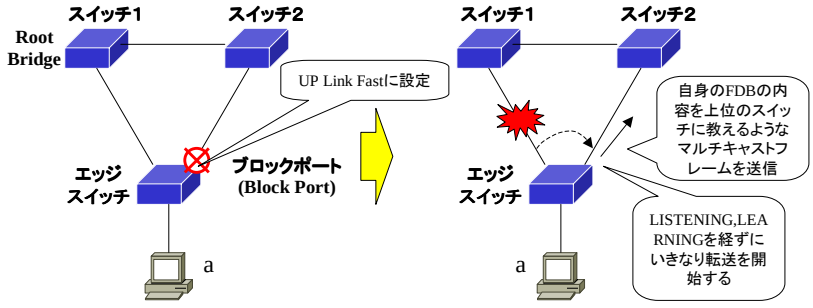
STPの拡張(Port fast)

- スイッチにPCなどを接続した後、LISTENING◊LEARNING◊FORWARDINGと状態変化する時間（通常30秒）待たなくてはならないのは使いにくい。
- スイッチが接続される可能性のないポートに関しては、事前に設定しておくことにより、いきなり、FORWARDINGになるようにしておく。
- BPDUをポートで受信した場合はポートをブロッキング状態にする。
- Ciscoで実装しているが多くのスイッチで同様の効果を得る設定は出来る。



STPの拡張(Uplink fast)

- CiscoによるSTPの拡張
- アクティブなリンクが断となった場合に、バックアップのリンクにすぐ切り替わる機能。（ LISTENING◊LEARNING◊FORWARDINGの状態変化にかかる時間をとばす）
- エッジに設置したスイッチが上位のスイッチに2本のリンクで接続されている場合にエッジに設定出来る
- 上位のスイッチのFDB構築を支援する為、Uplink fastでエッジスイッチが切り替わりを発生させた場合に、エッジスイッチは自身がFDB内に学習済みのアドレスを送り元アドレスとする、マルチキャストフレームを新しくアクティブにしたリンクに流す。

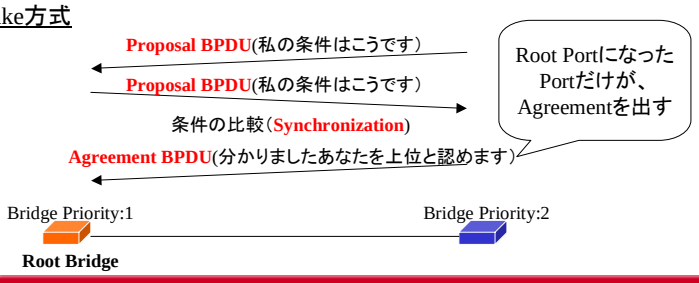


Rapid Spanning Tree Protocol
ループフリーな論理トポロジーを維持する機構
（STP高速化）

RSTP(Rapid Spanning Tree Protocol)802.1w

- IEEE 802.1w標準
- STPの切り替わり動作を高速化する為に作られた。（30秒や50秒かかるのは遅い）
- 構築される論理木構造はSTPと同じ（同じパラメータを使う）
- Max Age、Forward DelayのパラメータはBPDUを受信しないポートを代表ポートとする場合か、あるいはSTPと混在して使う時のみ有効
- 802.1wは802.1Dの上位互換性がある
- Point-to-Pointの接続が基本
- BPDUの交換にHandshake方式の導入
- ポートの役割の種類の追加

Handshake方式



RSTPにおけるポートの役割（Port Role）

ポートの役割	説明	定常状態
ルートポート (Root Port)	Rootブリッジへ最も少ないコストで到達出来る経路を提供するポート、STPと同じ	Forwarding
代表ポート (Designated Port)	リンクからRootブリッジへ最も少ないコストで到達出来る経路を提供するポート、STPと同じ	Forwarding
アルタネートポート (Alternate Port)	ルートポートに変わる二番目に少ないコストでルートブリッジに到達出来るルートブリッジへの経路を提供するポート。（複数あり） Next Root Port	Blocking
バックアップポート (Backup Port)	指定ポートが提供するリンクへの経路に変わるリンクへの経路を提供するポート Next Designated Port	Blocking
ディスエーブルドポート (Disabled Port)	故障しているか、シャットダウンされているポート、STPと同じ	Disabled

RSTPで使用されるBPDU

- RSTP-BPDU
  - BPDU ver2として定義
  - Hello 間隔ごとにスイッチ間で交換（3Hello timeで過去の情報は無効になる）
  - Flagの部分を拡張
  - Topology Change Notification BPDUは使わない。(STPとのインターワークでのみ利用）

RSTP BPDU

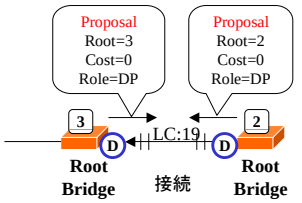
Protocol ID=0000h	2
Protocol Version ID=02h (2)	1
BPDU Type=0000 0010b (2)	1
Flags	1
Root ID	8
Root Path Cost	4
Bridge ID	8
Port ID	2
Message Age	2
Max Age	2
Hello Time	2
Forward Delay	2

FLAGを大きく拡張

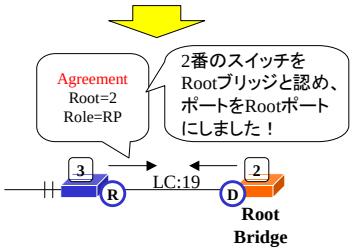
Bit位置	Flagの意味
0	Topology Change
1	Proposal
2-3	Port Role
00	Unknown Port
01	Alternate or Backup Port
10	Root Port
11	Designated Port
4	Learning
5	Forwarding
6	Agreement
7	Topology Change Ack

RSTPでのトポロジー構築

RSTPにおけるHandshake



条件比較(Synchronization)

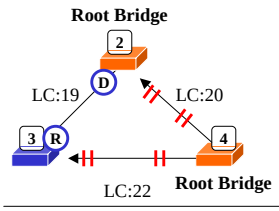


- 二つのスイッチが接続されると、ポートを代表ポートとし状態を非転送(ブロッキング)にした上で双方のスイッチが自身がRoot Bridgeであり、当該ポートが代表ポート(DP)であるとするProposal BPDUを送信する。
- Proposal BPDUを受信すると、自身の持つRoot情報及びコストと比較し。
  - 相手が勝る場合
    - 相手を代表ポート(DP)と認める
    - “Proposal BPDU”を受信したポートをRootポート(RP)として転送状態にし、その他のポートをブロック状態にする。
    - 相手に“Agreement BPDU”を送信する。
  - 自身が勝る場合
    - 相手から“Agreement BPDU”を受け取ると代表ポートとしてすぐに転送を開始する。

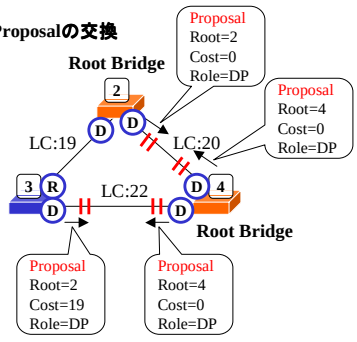
RSTPでのトポロジー構築

RSTPによる論理トポロジーの構築

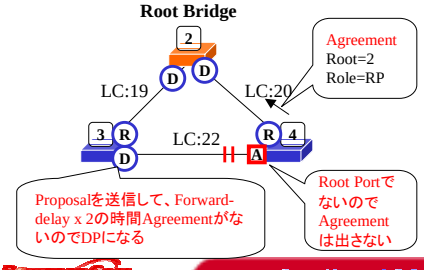
(1)ブリッジ4の追加



(2)Proposalの交換



(3)SynchronizationとAgreement



R ルートポート (Root Port)

|| ブロッキング

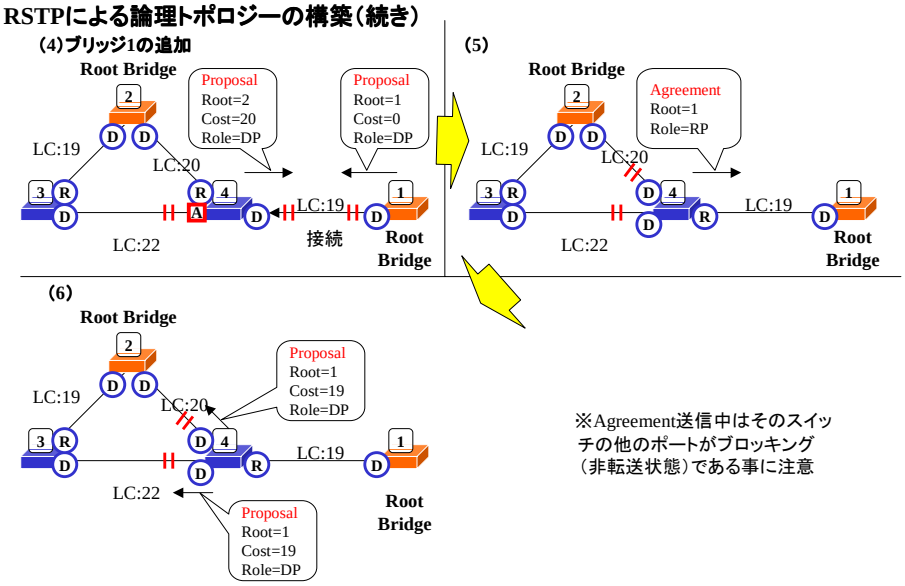
D 代表ポート (Designated Port)

A アルタネートポート (Block Port)

※Proposal交換中のポートがブロッキング(非転送状態)である事に注意

RSTPでのトポロジー構築

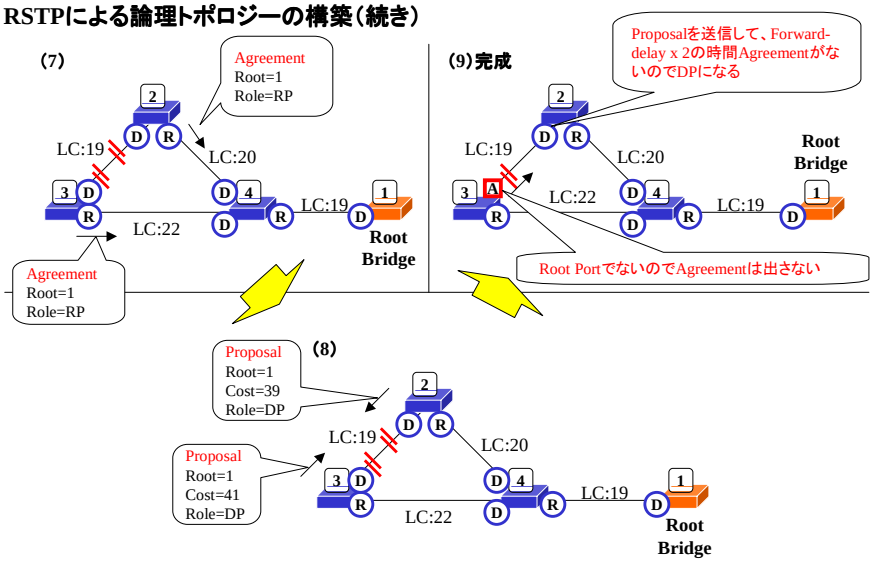
RSTPによる論理トポロジーの構築（続き）



※Agreement送信中はそのスイッチの他のポートがブロッキング（非転送状態）である事に注意

RSTPでのトポロジー構築

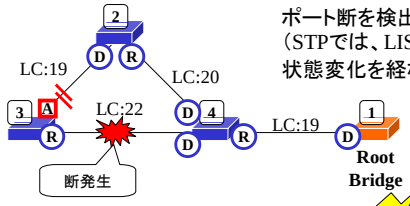
RSTPによる論理トポロジーの構築（続き）



RSTPでの障害回復（アルタネートポート）

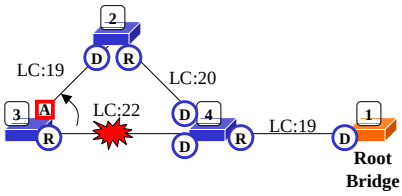
障害発生の場合（アルタネートポートがあるとき）

（1）障害発生



アルタネートポートは二番目にRootブリッジに近いポート。ルートポート断を検出するとすぐに転送状態に切り替える。
（STPでは、LISTENNING->LEARNING->FORWARDINGと言う状態変化を経なくては切り替えられなかった。）

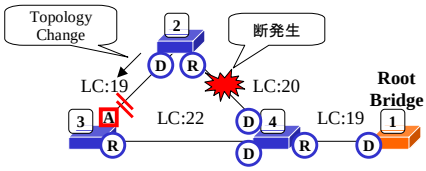
（2）すぐにアルタネートポートを転送状態にする



RSTPでの障害回復

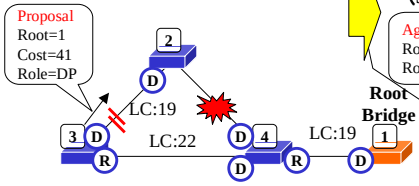
障害発生の場合２

（1）障害発生



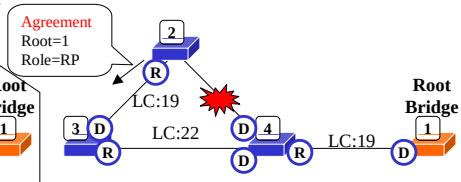
•スイッチ2はルートポートの断を検出すると、トポロジー変更があった事を示すBPDUを送信する。

(2) Proposal



•スイッチ3はTCNを受信後、ただちにHandshakeを開始

(3) Agreement



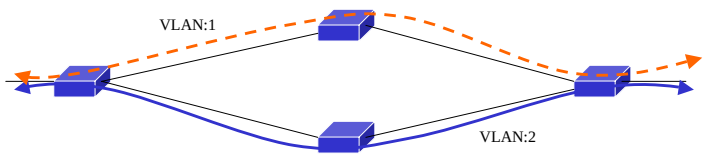
•Handshake完了で、新トポロジー完成

RSTPのまとめ

- トポロジー変更
  - LISTENNING->LEARNING->FORWARDINGのステートをHandshakeを導入したことによりはぶけるのでSTPよりも高速。
- トポロジー変更に伴うFDBのフラッシュ
  - RSTPでは、Topology Change Notification BPDUは使わない。そのかわり、Topology Change Flagを立てたBPDUを使って、他のスイッチにトポロジー変更が発生した事を教える。
  - Topology Change Flagを立てたBPDUを受信したスイッチは他のポートにTopology Change Flagを立てたBPDUを送信するともに、FDBのフラッシュを行う。
- 802.1Dとの接続
  - Proposalを投げて、Agreementを返してこなければ(Forward Delay x 2の時間)、802.1Dの動作をする。

MSTP(Multiple Spanning Tree Protocol)802.1s

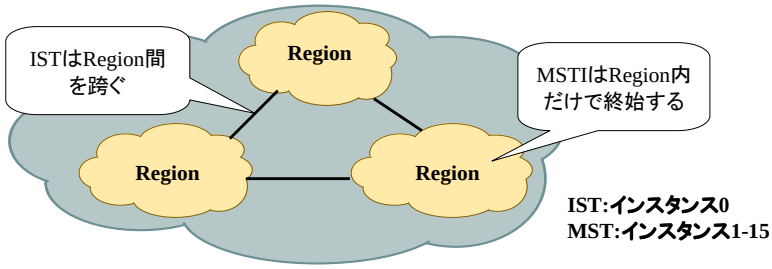
- STPで複数のトポロジー(インスタンス)を扱いたいと言う要求に答える為に登場
- IEEE 802.1s標準
- 802.1sは802.1Dの上位互換性がある。
- RSTP 802.1wと連携して使われる。
- VLANごとに別々のSTPのインスタンスを動作させる方法(PVSTなど)もあるがVLAN数が増えるとそれなりに負荷が大きくなるので、MSTPでは複数のインスタンスを一つのBPDUで扱えるようにしている。



負荷分散などの為にVLANごとに経路を変えたい事がある

MSTP

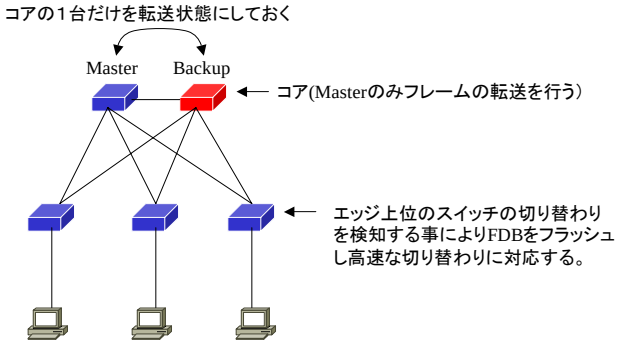
- MSTPには802.1D互換の親玉になる1個のInternal Spanning Tree(IST)と多数のMultiple Spanning Tree Instance(MSTI)がある。
- 個々のVLANはIST（すべてのVLANがマッピングされている）と任意のMSTIインスタンス1つにマッピングされ、それらのインスタンスの挙動に同期した挙動を行う。
- MSTIごとに、BPDU(MSTPのBPDUはversion3)にM-recordと呼ばれるレコードが追加される。
- 1個のBPDUに多数のM-recordが搭載される為、インスタンスが増えてもBPDUは増えない。
- Regionと呼ばれる概念がありMSTIはリージョン内に閉じ込められるが、ISTはRegionをまたいで存在する。



STPファミリ以外の冗長化プロトコル
ノード冗長化（メッシュトポロジー）
ESRP、VSRP、FVRP、GVRP

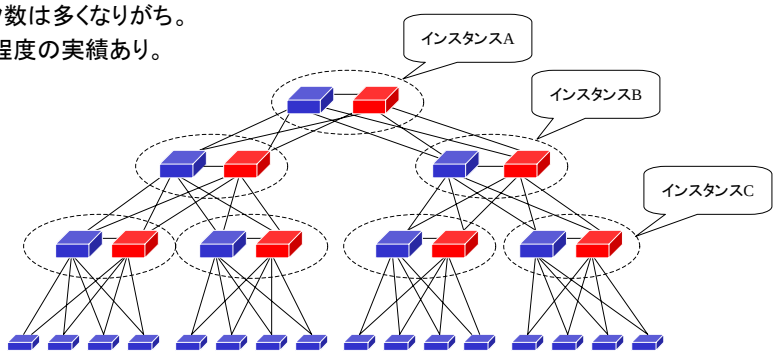
ノード冗長化プロトコル（メッシュトポロジー）

- ネットワークコアにあるスイッチを二重化し、コアに接続されるエッジ側のスイッチはコアに二重帰属する構成が基本
- Edge-Coreトポロジーとも呼ばれる。
- 標準的なプロトコルはない、ベンダ独自の実装
  - ESRP : Extreme
  - VSRP : Foundry
  - FVRP : Force10
  - GVRP（仮称）：日立製作所



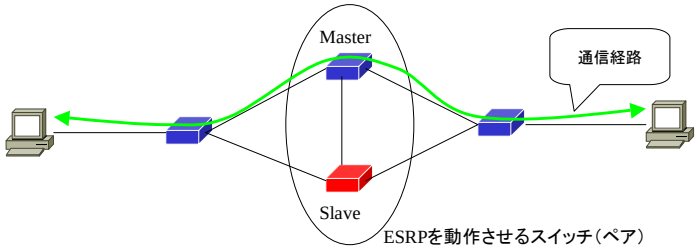
ノード冗長化プロトコルの大規模化

- 大規模な冗長を組む場合は、スイッチのペアごとに独立したのノード冗長化のインスタンスを作成し、そのペアを通過するVLANはそのインスタンスの挙動に同期して動作をするようにする。
- STPと比較して、スイッチやリンクの障害がネットワーク全体のトポロジーの再構成に引き起こさない（トポロジー変更が局所に閉じる）と言うメリットがある。
- リンク数は多くなりがち。
- ある程度の実績あり。



ノード冗長化プロトコル（ESRP）

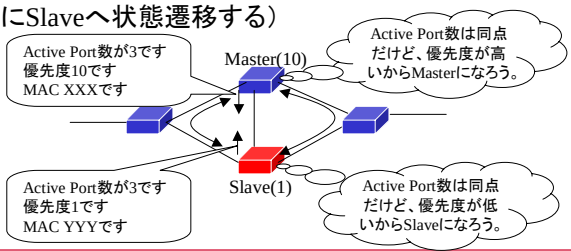
- ESRP（Extreme Standby Router Protocol）
  - Extreme社が開発した、ノード冗長化プロトコル、レイヤー2とレイヤー3の冗長機能の両方の機能を提供している。
  - 冗長機能を必要とするスイッチにESRP機能を持たせ、冗長を持たせている。
    - Master スイッチ： データの送受信を行っているSW
    - Slave スイッチ： データの送受信を行わず、予備状態となっているSW(Standbyとも言う)
    - マスタvlan： ESRPを管理するvlan、マスタvlanのみESRPのアルゴリズムを計算し、他のvlanはマスタvlanの動作に同期してMaster Slaveの選択を行う事が出来る。



ノード冗長化プロトコル(ESRP)

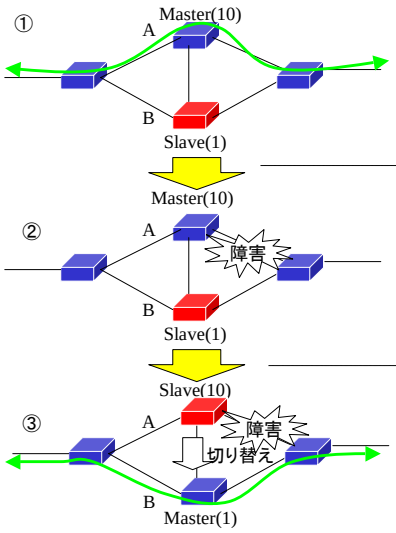
- ESRPマスターの選択
  - ESRPを動作させるスイッチでは、定期的に制御フレーム(ESRP hello packet)を交換し、どのスイッチが最もMasterにふさわしいかを判断している。
- ESRPマスターを決定する要素
  - Active Port数
  - スイッチの優先度(Priority)
  - トラッキング情報(pingなど)
  - システムMACアドレス（大きい番号のものが優先）

  これらの要素をタイブレークルールで比較していく。（比較順は変更可能）
- 相手スイッチがMasterに遷移したと言う通知を受けた時、自身がMasterであったら、即座にSlaveへ状態遷移する）



ノード冗長化プロトコル(ESRP)

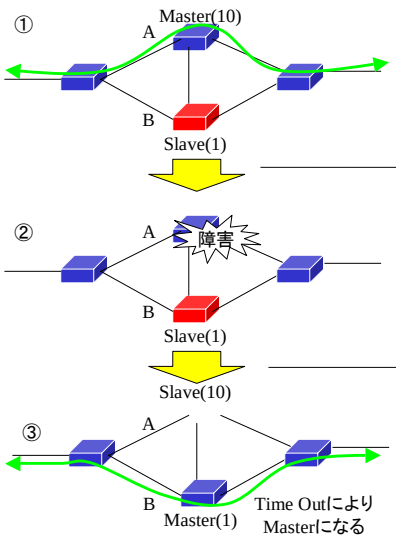
リンク障害による切り替え



- フレームはMasterとなっているスイッチAを経由して流れている。
  - Active port数 3:3 → 同点
  - Priority 数 10:1 →10の勝ち(スイッチAがマスター)
- スイッチAを経由する通信経路で障害が発生。
  - 勝負前
- ESRPの機構により、Master/Slaveの切り替えが発生し、フレームの流れる経路も変わる。
  - Active port数 2:3 →3の勝ち(スイッチBがマスター)

ノード冗長化プロトコル(ESRP)

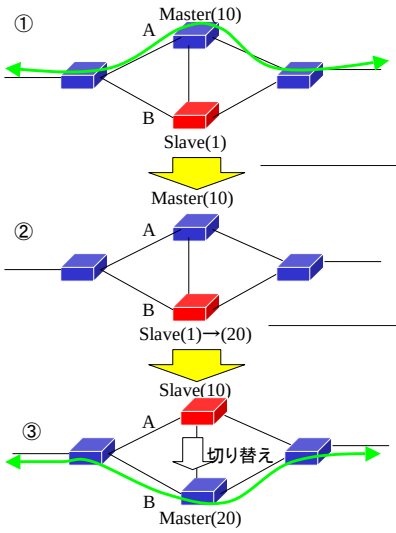
ノード停止による切り替え



- フレームはMasterとなっているスイッチAを経由して流れている。
  - Active port数 3:3 → 同点
  - Priority 値 10:1 →10の勝ち(スイッチAがマスター)
- スイッチAが停止
- スイッチBでhello packetが未受信となってあらかじめ設定した時間を越えると、Slaveスイッチは Masterスイッチに障害が発生したと認識してMasterとなる。

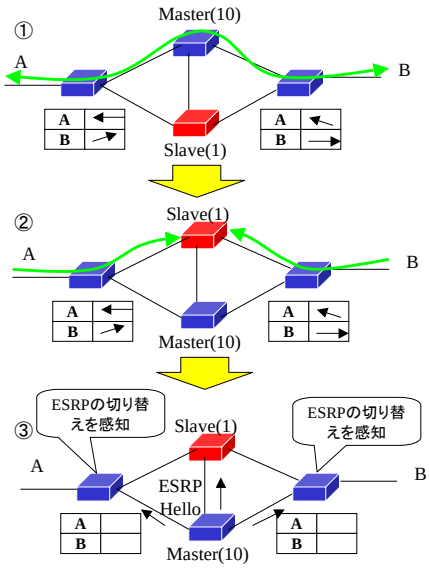
ノード冗長化プロトコル(ESRP)

Priority変更による切り替え



- フレームはMasterとなっているスイッチAを経由して流れている。
  - Active port数 3:3 → 同点
  - Priority 値 10:1 →10の勝ち(スイッチAがマスター)
- スイッチBに系を切り替える為にPriority値を 1から20に変更する。
  - 勝負前
- ESRPの機構により、Master/Slaveの切り替えが発生し、フレームの流れる経路も変わる。
  - Active port数 3:3 →同点
  - Priority値 10:20→20の勝ち(スイッチBがマスター)

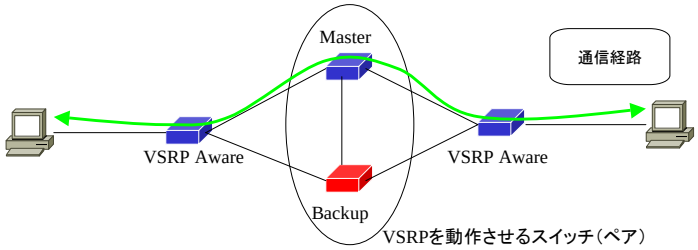
ノード冗長化プロトコル(ESRP AWARE)



- ESRPのMasterを経由してトラフィックが流れると当然その経路に従ったFDBの学習がなされる。
- ESRPのPriorityなどを変更して、Master/Slaveの関係が入れ替わった場合に、そのままでは、古いFDBの内容に従ってフレームの転送が行われる為、フレームが結果的に転送不能となる。
- FDBの矛盾状態を防ぐ為、両端のスイッチは、上位のスイッチがMaster/Slaveの関係を変更した事をESRP Helloの内容変更を見る事により、FDBの内容をFlushする(消す)。これにより再度学習が行われ、正常な通信が行えるようになる。

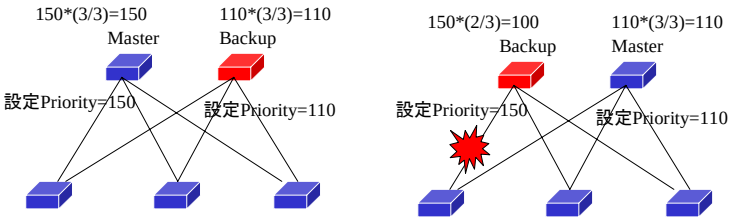
ノード冗長化プロトコル（VSRP）

- VSRP（ Virtual Switch Redundancy Protocol ）
  - Foundry Networks社が開発した、ノード冗長化プロトコル、レイヤー2とレイヤー3の冗長機能の両方の機能を提供している。
  - 冗長機能を必要とするスイッチにVSRP機能を持たせ、冗長を持たせている。
    - Master スイッチ： データの送受信を行っているSW
    - Backup スイッチ： データの送受信を行わず、予備状態となっているSW
    - Topology Group：複数のVLANをグループ化し、Masterを共有する機能。
  - VSRP Awareな装置は、系が切り替わりMasterとなったスイッチが送信するTC packetsを受信すると、FDBをフラッシュするのではなく、Backup側に書き換える。



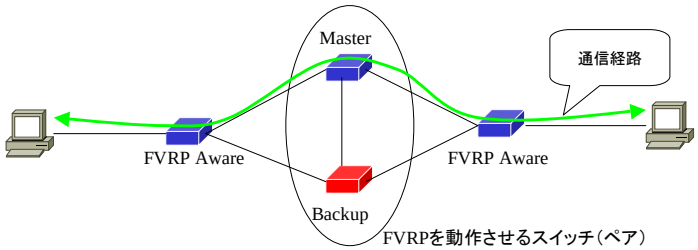
ノード冗長化プロトコル（VSRP）

- VSRPの priority(3-255 Default 100)の高い方がMasterとなる。
- ポートがダウンするとpriorityが減る。Priority x（利用可能Link数/設定Link数)
- Tracking portにより、特定のリンクのDownによりpriority値を制御可能
- VSRP Helloを使って　priority情報を交換(Default 1秒間隔)
- Active 決定後はHelloはMasterからのみ送信
- Backup側のスイッチはMasterから、Dead interval時間Helloを受信しないと、Hello packetを送信しはじめ、さらに、Hold-down interval時間自分よりpriorityの高いHelloを受けとらなければ、Masterとなる。



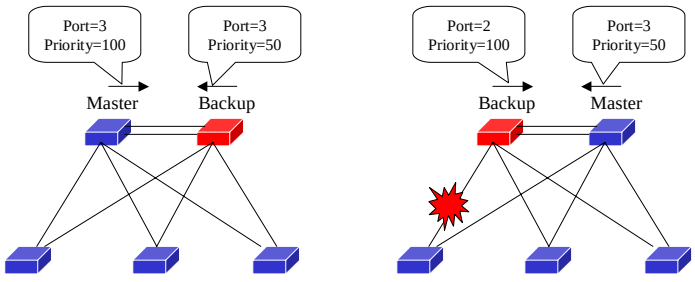
ノード冗長化プロトコル（FVRP）

- FVRP（ Force10 VLAN Redundancy Protocol ）
  - Force10社が開発した、レイヤー2ノード冗長化プロトコル。
  - 冗長機能を必要とするスイッチにFVRP機能を持たせ、冗長を持たせている。
    - Master スイッチ： データの送受信を行っているSW
    - Standby スイッチ： データの送受信を行わず、予備状態となっているSW
    - FVRP Domain：複数のVLANをグループ化し、Masterを共有する機能。
  - FVRP Awareな装置は、コアスイッチより、flush address messageを受信するとFDBをフラッシュする。



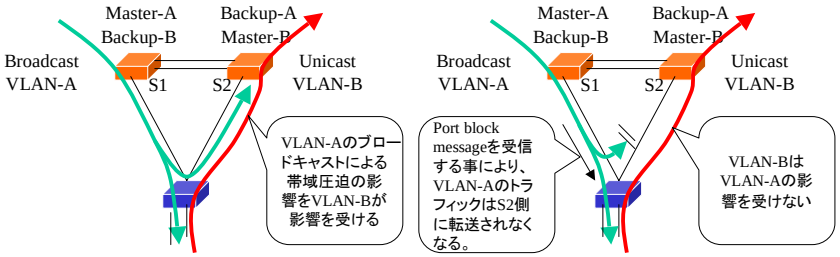
ノード冗長化プロトコル（FVRP）

- ポート数、priority、制御ポートのMACアドレス（低い方が有利）の順でタイブレークルールで比較し、勝った方がMasterとなる。
- priority(1-255 ただし255は強制Slave)は高い方がMasterとなりやすい。
- FVRP Helloを使って　priority情報を交換(Default 1秒間隔)
  - 通常は、Master-Standby間に張られたCore Linkを使ってHelloのやり取りを行う。
  - Core Linkが断になった場合はアクセスリンク上のコントロールVLANを使ってHelloのやり取りを行う。(Dual Masterを防ぐ為）
- Standby側のスイッチはMasterから、Message Age Timer時間Helloを受信しないと、遷移プロセスに移る。

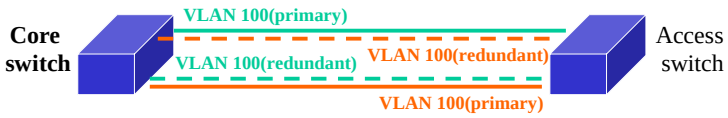


ノード冗長化プロトコル（FVRP）

- ブロードキャストが帯域を無駄使いする事を防止
  - Masterから port block messageを受信すると該当するVLANのStandby側のポートをブロックする。（これによって余計なブロードキャストが回りこまずに、帯域を有効活用出来る）

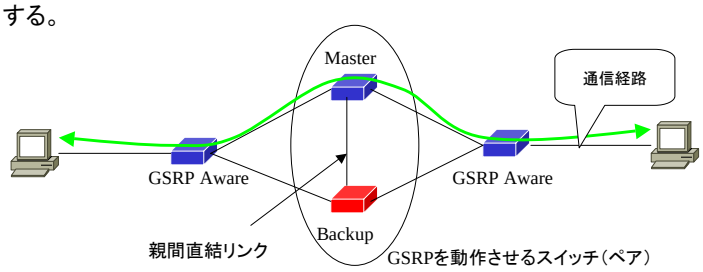


- FVRPはスイッチの対向でも使用出来る！



ノード冗長化プロトコル（GSRP）

- GSRP（ GS4000 Switch Redundant Protocol =仮称 ）
  - （株）日立製作所が開発した、ノード冗長化プロトコル。
  - 冗長機能を必要とするスイッチにGSRP機能を持たせ、冗長を持たせている。
    - Master スイッチ： データの送受信を行っているSW
    - Backup スイッチ： データの送受信を行わず、予備状態となっているSW
    - VLAN Group：複数のVLANをグループ化し、制御する機能。
  - マスターの切り替えが発生すると、新マスタは全隣接スイッチに、GSRP Flush Requestを送信する、GSRP Awareな装置はそれを受信するとFDBをフラッシュする。



ノード冗長化プロトコル（GSRP）

- GSRPマスターの選択
  - GSRPを動作させているスイッチでは、定期的に制御フレーム(GSRP Advertise)を交換し、どのスイッチが最もMasterにふさわしいかを判断している。
- GSRPマスターを決定する要素
  - Active Port 数
  - スイッチの優先度（Priority）
  - 装置MAC（大きい番号のものが優先）

  これらの要素をタイブレークルールで比較していく。（比較順は変更可能）
- デュアルマスター防止策
  - 切り替え時のデュアルマスターの可能性（デュアルマスターはループになる）を排除する為の機構を持っている。（瞬間ループもFDBが狂うので絶対に駄目！）

  （1）マスターになろうとするスイッチはまず、マスター待ち状態になる。（ブロック状態のまま）

  （2）バックアップになろうとするスイッチはすぐにバックアップになり、バックアップになった事をGSRP Advertiseを使って広報

  （3）マスター待ちのスイッチは相手側のスイッチがバックアップになった事を示すGSRP Advertiseを受信すると、マスターとして動作しはじめる。
- GSRP Advertiseを規定回数受信しないと（1-255 Default=3）で相手不定状態になる。
- オプション指定時は、相手不定状態と親間直結リンク断条件組み合わせでバックアップスイッチはマスターとして動作しはじめる。

STPファミリ以外の冗長化プロトコル（リングトポロジー）

RPR

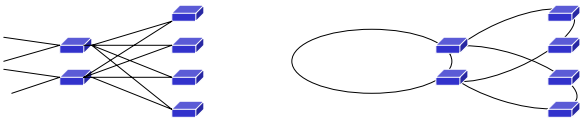
EAPS、MRP、MMRP2

リングトポロジー

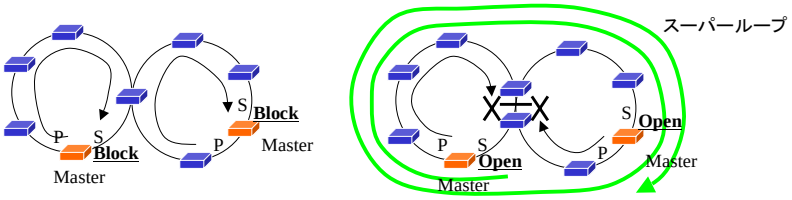
- リングトポロジーはメッシュ（ノード冗長化）トポロジーよりも、伝送路やインターフェースの必要量が少ないと言う特徴がある。
- RPR（Resilient Packet Ring）のように、高度で高価な技術の他にイーサネットスイッチをリング状に配置し、そのリングにHelloパケットを流す事によってリンク断の監視を行い、ブロッキングポートの制御を行うような単純で安価な方式がある。（EAPS、MRP、MMRP2）
- １つのリングだけでは限界がある事が多く、どうやって、複数のリングを冗長を持たせた形で接続しかつループを起こさないか？が課題の一つとなっている。

ノード冗長化プロトコル　リングマルチ接続

- １つのリングだけではスケーラビリティが限られている為、複数のリングを接続したいと言う要求がある。（大規模な接続を行う場合に、ノード二重化プロトコルを利用するよりも、リンク数を減らせると言う考え方もある、昔、DECのFDDIスイッチが流行した時のようなリングの使い方をやろうとするとこれが必要）

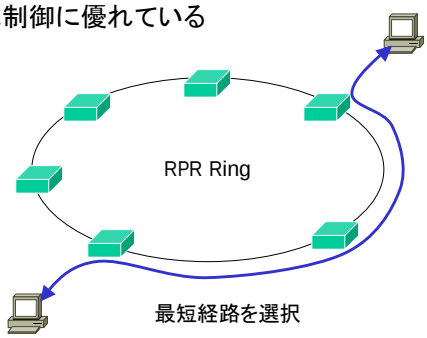


- リングを一箇所で接続するのは問題ないが、冗長の為に2箇所で接続すると、冗長部分の渡りで断が発生した場合にスーパーループが出来てしまう。



RPR(Resilient Packet Ring) 802.17

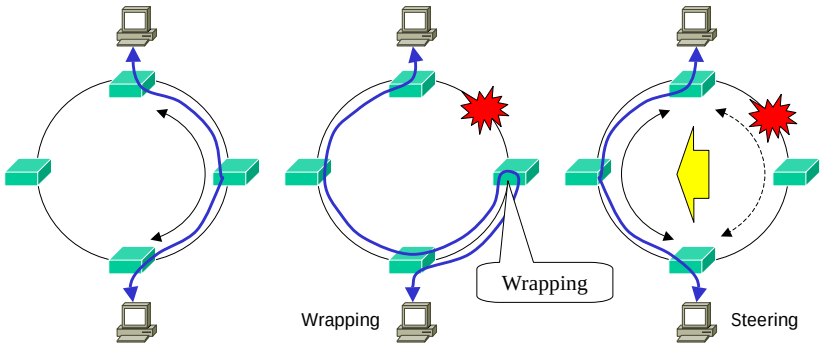
- IEEE 802.17　標準化中
- リング型転送方式（最近のEAPSなどの簡易なリングとは異なる）
- 50msec以内の高速障害回復
- SONET/SDH(C48c、OC192c)で利用可能
- Spatial Reuse 通常時リング内の最短経路で転送を行う（Link State情報）
- 通信の公平性を保つ機構がある
- QoS制御や帯域制御に優れている



RPR(Resilient Packet Ring) 802.17

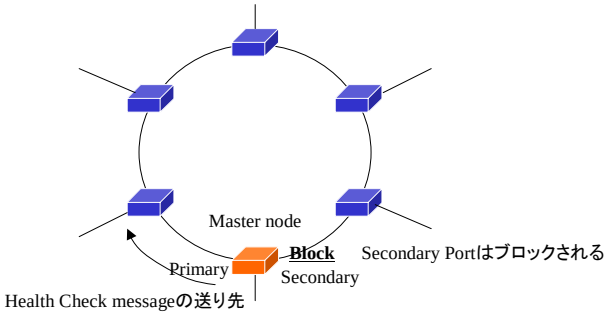
障害発生時の切り替えの方法の種類

- Wrapping
  - 障害が発生部分の直近で折り返したリングを作る事により障害回復を図る。
  - 障害回復がはやい
- Steering
  - 障害部位を通らない方向にリングを切り替える。
  - Wrappingと異なり、切り替え後の遅延の変動も少なく、帯域も有効活用出来る。



ノード冗長化プロトコル　リング構成(EAPS)

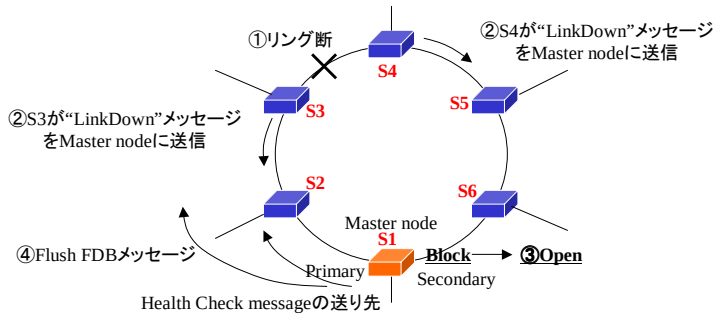
- EAPS（Ethernet Automatic Protection Switching）
  - Extreme社が開発した、リング型冗長化プロトコル
- リング構成で使用する簡易な冗長化プロトコル
  - リングとなるようにスイッチを接続し、その中にMaster nodeを１台指定する（手動）
  - Master nodeのリングに所属するポートの一つをPrimary Portとし、もう一方をSecondary Portとする。
  - Master nodeのSecondary Portをブロッキング状態にする。
  - Primaryポートより、Health Check messageを送信し、Secondaryポートで受信出来るかによってリングの状態を監視する方式



ノード冗長化プロトコル　リング構成(EAPS)

障害発生時の挙動

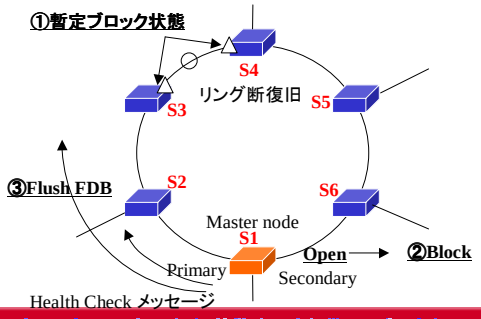
- Master nodeがリングの障害を検出する方法
  - “Link Down”メッセージを転送ノードより受け取る。
  - Health Check messageがSecondaryポートで受信出来なくなる。
- 障害によりリングが切断されたと判断すると、Master nodeはSecondaryポートをBlock状態からOpen(転送)状態に遷移させる。
- トポロジーの変更によるFDBエントリの矛盾を回避する為、Master nodeは転送ノードに対して、“Flush FDB”メッセージを流し、それを受け取った転送ノードはFDBの内容のFlush（消去）を行う。



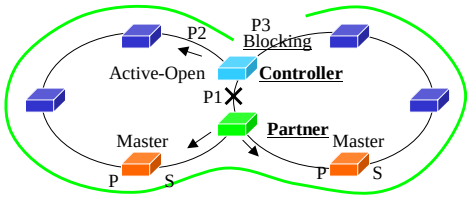
ノード冗長化プロトコル　リング構成(EAPS)

障害復旧時の挙動

- Master nodeはリング障害中もHealth Check メッセージの送信は続け、Secondary Portに届かない限り、障害が継続中であると判断する。
- リングが復旧後、Health Checkメッセージが Secondary Portに到着するまで、ループが発生する可能性がある、これを防ぐ為に、断の復旧を検出した転送ノードはそのポートを暫定ブロック状態とし、実データを通さず、Health Checkのみ通すようにする。
- Master nodeはHealth Check メッセージを Secondary Portで受信すると、Secondary Portをブロッキング状態にし、“Flush FDB”メッセージを送信する。
- 転送ノードは“Flush FDB”メッセージを受信すると、自身のFDBを一旦消去するとともに、暫定ブロック状態のポートを通常の転送状態にし、次のノードに“Flush FDB”メッセージを転送する。



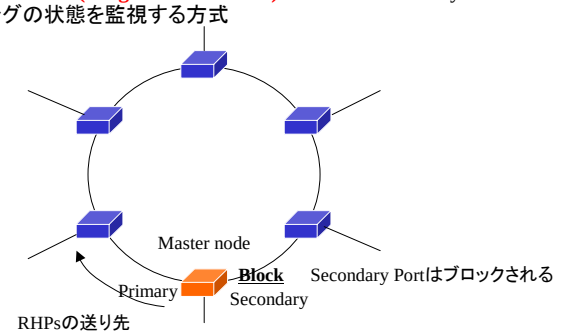
2ノード接続マルチリング構成(EAPS)



- 2つのリングが共有するリンク部分を挟む形で、Controllerとpartnerを設置しておく（ControllerとPartnerは互いにhelloを交換して共有リンクを監視）
- 共有リンク断
  - 共有リンクの断を検出すると、Controllerは1つのポートをActive-Openと呼ばれる状態にして、他のポートをブロッキング状態にする。
- 共有リンク復旧
  - 共有リンクが復旧すると、ControllerはBlocking Stateにしている部分と共有リンク部分をPreforwarding mode（Masterが流す、health-checkのみ通す）にする。（そのまま転送状態にすると一時的なループを構成してしまう為）
  - 双方のリングのMaster nodeがhealth-checkにより、Secondary Portをブロックにし、“Flush FDB”を送出する。
  - ControllerはMaster nodeがSecondary Portを閉塞した事を示す、“Flush FDB”を受信すると、全てのポートを転送状態にする。

リング構成(MRP)

- MRP（Metro Ring Protocol）
  - Foundry Networks社が開発した、リング型冗長化プロトコル
- リング構成で使用する簡易な冗長化プロトコル
  - リングとなるようにスイッチを接続し、その中にMaster nodeを１台指定する（手動）
  - Master nodeのリングに所属するポートの一つをPrimary Portとし、もう一方をSecondary Portとする。
  - Master nodeのSecondary Portをブロッキング状態にする。
  - Primaryポートより、**RHPs(Ring Health Packets)**を送信し、Secondaryポートで受信出来るかによってリングの状態を監視する方式



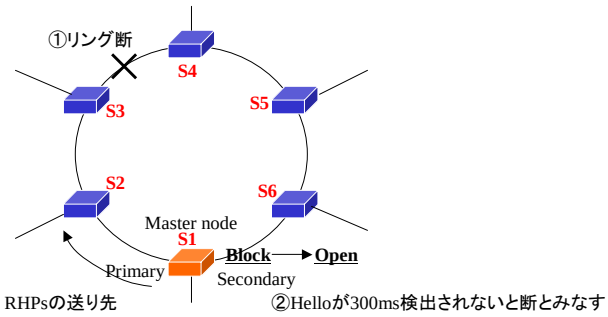
リング構成(MRP)

障害発生時の挙動

- Master nodeがリングの障害を検出する方法
  - RHPsがSecondaryポートで受信出来なくなる。

（RHPsは100ms間隔で送信されており、300ms検出されないと異常と見なす。）

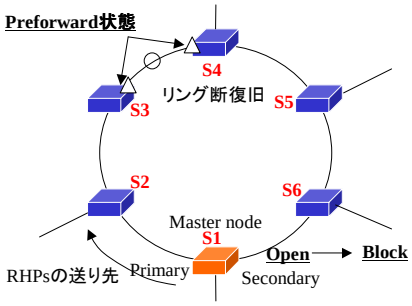
- 障害によりリングが切断されたと判断すると、Master nodeはSecondaryポートをBlock状態からOpen(転送)状態に遷移させる。
- トポロジーの変更によるFDBエントリの矛盾を回避する為、Master nodeは転送ノードに対して、“Flush FDB”メッセージを流し、それを受け取った転送ノードはFDBの内容のFlush（消去）を行う。



リング構成(MRP)

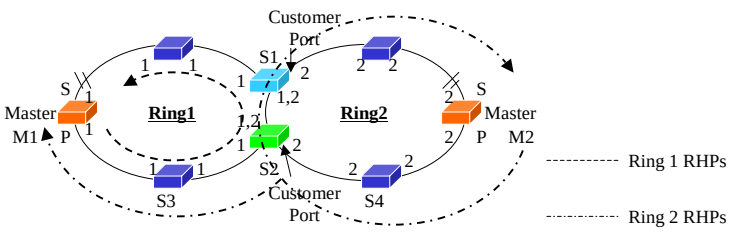
障害復旧時の挙動

- Master nodeはリング障害中もRHPs(Ring Health Packets) の送信は続け、Secondary Portに届かない限り、障害が継続中であると判断する。
- リングが復旧後、RHPsがSecondary Portに到着するまで、ループが発生する可能性がある、これを防ぐ為に、断の復旧を検出した転送ノードはそのポートをPreforward状態とし、実データを通さず、RHPsのみ通すようにする。
- Master nodeは RHPsをSecondary Portで受信すると、Secondary Portをブロッキング状態にし、“Flush FDB”メッセージを送信する。
- 転送ノードは“Flush FDB”メッセージを受信すると、自身のFDBを一旦消去するとともに、Preforward状態のポートを通常の転送状態にし、次のノードに“Flush FDB”メッセージを転送する。



2ノード接続マルチリング構成(MRP)

- 2ノードのマルチリング接続はIronWare Release 07.7.00からサポート
- Ring Priorityと呼ばれる数値が各リングに設定される。
- 二つのリングが接続されている所でRing Priorityの数字が大きい側のリングのポートはCustomer Portと呼ばれる。
- Customer Portから入力された、RHPsだけはRing Priorityの小さい側のリングにも流れ込む。このRHPsは通常Master Nodeで止まる。



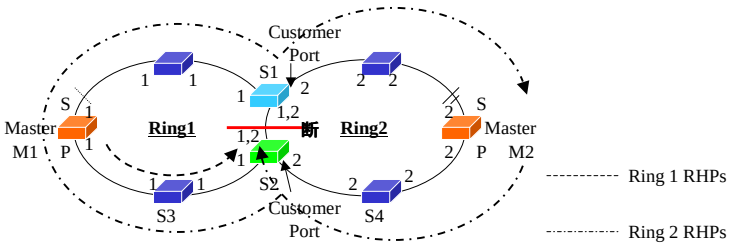
2ノード接続マルチリング構成(MRP)

障害発生時の挙動

- 共有リンク部分S1-S2間での断をM1が検出すると、M1のSecondary portは他のリングのRHPsのみを透過するPreforwarding状態になる。
- この状態ではM2のSecondary Portはブロック状態のままでありこの後、M1がSecondary portをforwarding状態にしても、ループは発生しない。

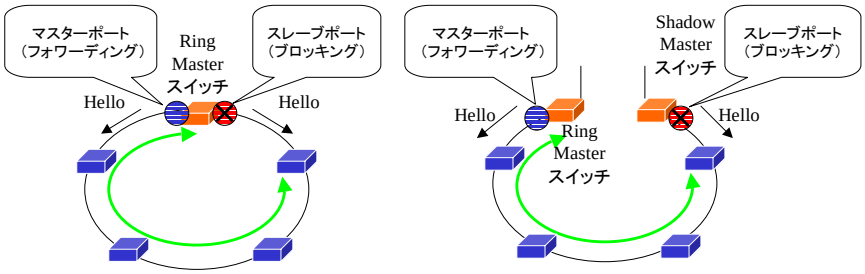
障害復旧時の挙動

- 1リング構成の場合と同じ



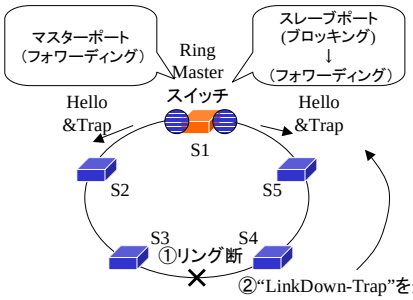
ノード冗長化プロトコル　リング構成(MMRP2)

- MMRP（Multi Master Ring Protocol 2）
  - 日立電線が開発した、リング型冗長化プロトコル
  - Ring Master スイッチには、Masterポート及び、Slaveポートがあり両方のポートで、Health Checkフレームを投げる。
  - Health チェックが相手のポートに届いているかどうかでリングの状況を確認する。



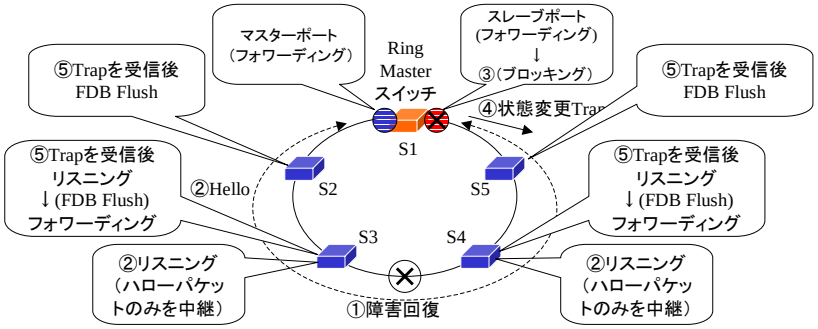
ノード冗長化プロトコル　リング構成(MMRP2)

- 障害発生時の挙動
  - 障害の検出方法(Ring Master node)
    - スレーブポートがHello packetを n秒以上連続して受信しない場合
    - LinkDown-Trapを受信した場合
  - Ring nodeでの障害の検出とFDBフラッシュ
    - 直接LinkDownを検出した場合
    - Ring Masterスイッチがスレーブポートをフォワーディング状態に変更したTrapを受信した場合
    - どちらかのHello Packetをn秒以上連続して受信しない場合



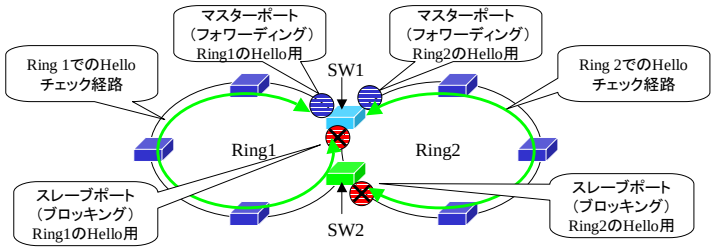
ノード冗長化プロトコル　リング構成(MMRP2)

- 障害回復時の挙動
  - Master nodeでの障害回復の検出
    - Helloパケットを受信した場合
      →スレーブポートをブロッキングに切り替え、ブロッキングにした事を示すTrapを送信
  - Ring nodeでの障害回復の検出とFDBフラッシュ
    - 直接LinkUpを検出したノードはそのポートをすぐに転送状態にはしないで、ハローパケットのみ通す、リスニング状態にする。
    - Ring Masterスイッチのスレーブポートがブロッキングになった事を示す、trapを受信後、リスニング状態のポートがあれば、フォワーディングに変更し、FDBをFlushする。



2ノード接続マルチリング構成(MMRP2)

- MMRP2によるマルチリング
  - Masterスイッチを分散して設置出来る事を利用して、マルチリング接続を行う。
  - 共有部分のリンク断によって、ループ構成とならないように、Masterと Shadow Masterの配置を行えばそれだけで、特別な機能は使わずにマルチリングの構成を構築する事が出来る。

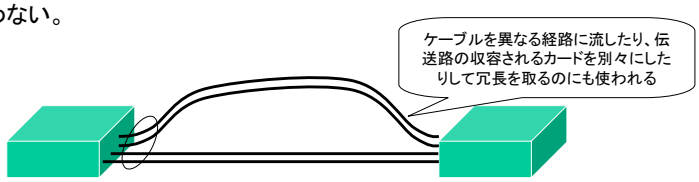


SW1：Ring1のシングルMasterスイッチ、リング2のShadow Masterスイッチ
SW2：Ring2のShadow Masterスイッチ

リンク冗長としてのLink Aggregation

Link Aggregation 802.3ad

- IEEE 802.3adとして標準化
- スイッチ間の複数の物理リンクを論理的に1本にまとめて使う機能
- 負荷分散の他、伝送路の冗長を確保する為にも用いられる
- 制御プロトコルとして、(LACP:Link Aggregation Control Protocol)を規定
  - 論理チャネルを動的に組み上げるためのプロトコルで、リンクの状態のチェックや接続の間違いをチェック出来る。
  - 実装されていない場合は、手動で設定する。
- トラフィックの振り分けは、MAC、IP、ポート番号、入力ポートなどのハッシュやラウンドロビンなどがある。
  - 平均的に分散するわけではないので、リンク数に比例したパフォーマンスを期待出来るわけではない。
  - ラウンドロビンはパケットの順番入れ替えが発生する可能性があるのであまり使わない。

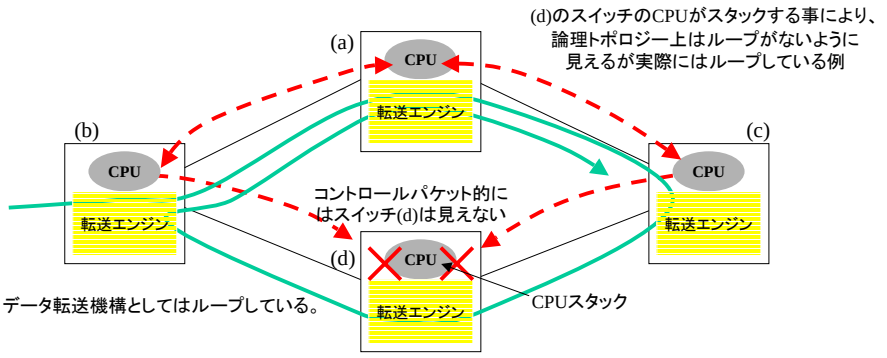


複数の伝送路をまとめて使用

ループを検出し、論理トポロジーに働きかける機構

ループを検出し、論理トポロジーに働きかける機構

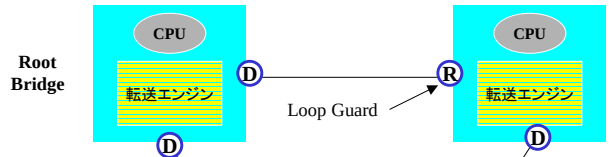
- 論理トポロジーがアルゴリズム通り維持出来ない可能性もある。
  - 多くのスイッチが、データ転送を行う部分（転送エンジン）と、制御パケット（BPDU）などをコントロールする部分（CPU）が分かれている為、CPUの過負荷やソフトウェアの障害により、制御パケットがCPUに転送されるが、処理させず、データの転送のみが実行される状況が発生する可能性が存在する。
  - その他、リブートのタイミングで、設定情報の読み込みに失敗して立ち上がってくるようなスイッチが存在する可能性も存在する。



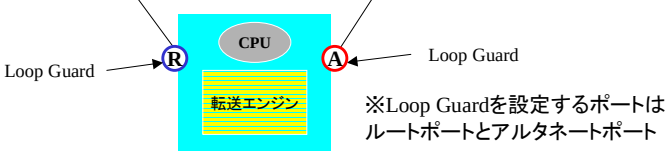
ループを検出し、論理トポロジーに働きかける機構

ループガード（Cisco）の例

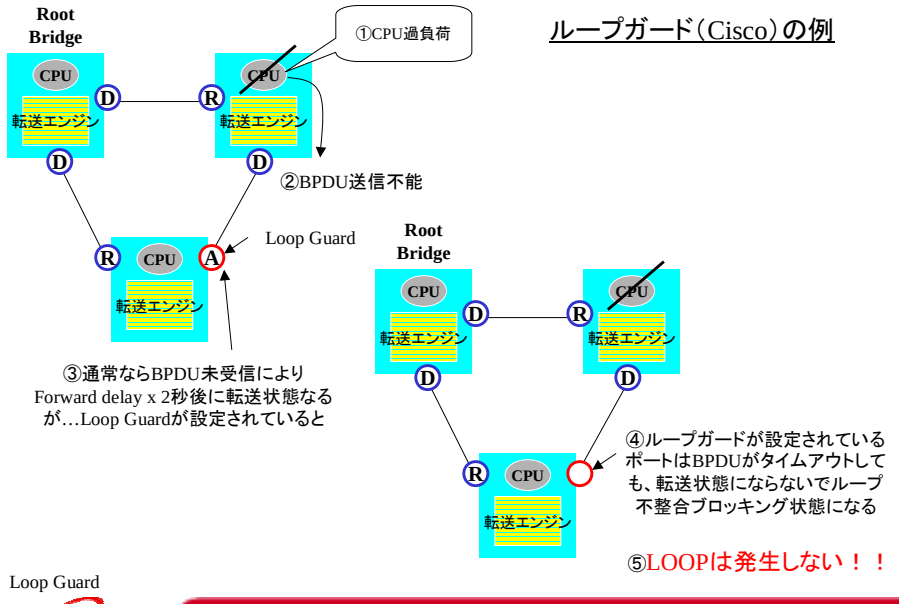
- ループガード（R-STP用）
  - 隣接したスイッチのCPU異常や単方向リンクの発生によりBPDUが受信出来ない場合に、ループの発生を防ぐ機能
  - ループガードを設定したポートでBPDUを受信しなくなっても、代表ポートにはせず、ループ不整合ブロッキングステートにする。



※上位（Root Bride向け）に向いているポートであると分かっていれば、そのポートでBPDUを受信しなくなる事によって、代表ポートになる必要性はない！（Hand shake代表ポートになる可能性はある）



ループを検出し、論理トポロジーに働きかける機構

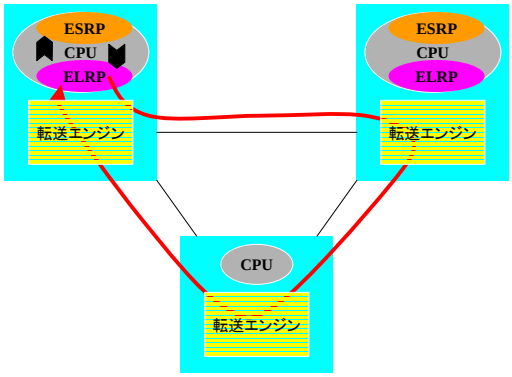


Loop Guard

ループを検出し、論理トポロジーに働きかける機構

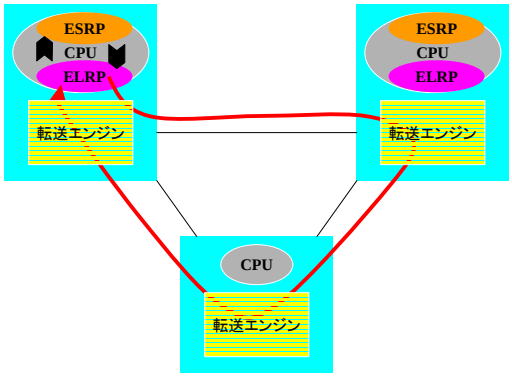
Extremeの例

- ELRP（Extreme Loop Recovery Protocol）
  - ループ検出を行い、論理トポロジーに働きかける機構として、ELRPと呼ばれる機構を提供している。この機構はESRPと組み合わせて利用される。
  - この機能はESRPのMaster/Slaveの関係の中で、両方のスイッチがMasterにならないよう、ループの防止/検出をする。



ループを検出し、論理トポロジーに働きかける機構

- ELRPの動作概要
  - ELRPパケットを呼ばれる監視パケットを送出し、ループの有無を検出
    - 送出されたパケットを受信した場合は、ループありと判定
    - 送出されたパケットを受信しない場合は、ループなしと判定（正常）
  - ELRPのパケットは発信スイッチ以外のスイッチではCPUに転送されず、転送エンジン内で転送される為、他のスイッチのCPUの状況に左右されない。



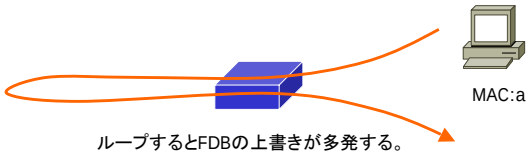
ループを検出し、論理トポロジーに働きかける機構

- ESRPとELRPの連携
  - ELRPはループ検出を行うが、トポロジー維持機構のESRPと連係して動作をする事により、ループを検知し、ループ回避を行う事が出来る。
- ELRP Master-poll機能
  - Master スイッチが定期的に両系Master検出用パケット（宛先マルチキャスト）を送出し、戻りを検出した場合にループが発生していると判断し、Slaveに落ちる機能。
- ELRP premaster-poll機能
  - スイッチがESRPのMasterに遷移する直前に、両系Master検出用パケット（宛先マルチキャストを送出し、戻りを検出した場合に自身がMasterに遷移した場合にループになる事を事前に察知し、Masterにならない機能。

ループ検出機構

ループ検出機構

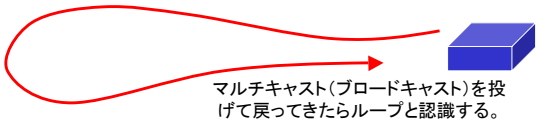
- ループ検出の必要性について
  - 様々な機構を駆使して、ループを防止したとしても、設定ミスなどの可能性やケーブリングのミスなど、ループする可能性を0にする事は難しい。
  - 被害を最小限に食い止める為にループが発生している事にすぐに認識し対策をうつ必要がある。
- ループ検出機構
  - FDBの書き換わりを見張る方法
  - ループするとFDBの書き換わりが多発する、ある程度の時間内にある程度の回数書き換えが発生すると、ループの疑いがあるとして、警報を上げる。（SEIKOのキャリア向けスイッチなどに実装）



ループするとFDBの上書きが多発する。
(MAC:aが右に見えたり左に見えたり)

ループ検出機構

- マルチキャストHelloポーリング
  - マルチキャスト宛てにHelloフレームを投げ、それが戻って来るかどうかを確認する事により、ループがないか確認する方法。（制限はあるがCiscoなどが実装）

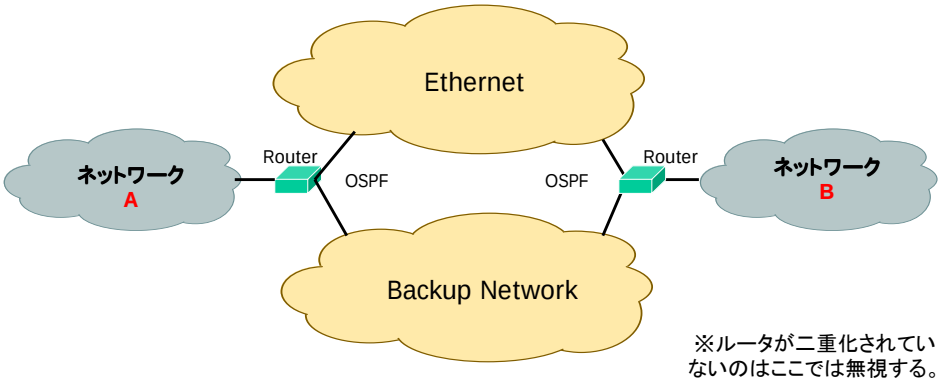


- マルチキャスト、ブロードキャスト、ユニキャストの流量観察
  - ネットワーク上に流れている、マルチキャスト、ブロードキャスト、ユニキャストの定常的な流れのバランスをモニタしておき、マルチキャストやブロードキャストの流量の急激な増加、ユニキャストの減少などよりループを検知する。
  - 運用レベルの検出方法。

大規模イーサネットを介したネットワーク冗長について

大規模イーサネットを介した冗長を組む場合

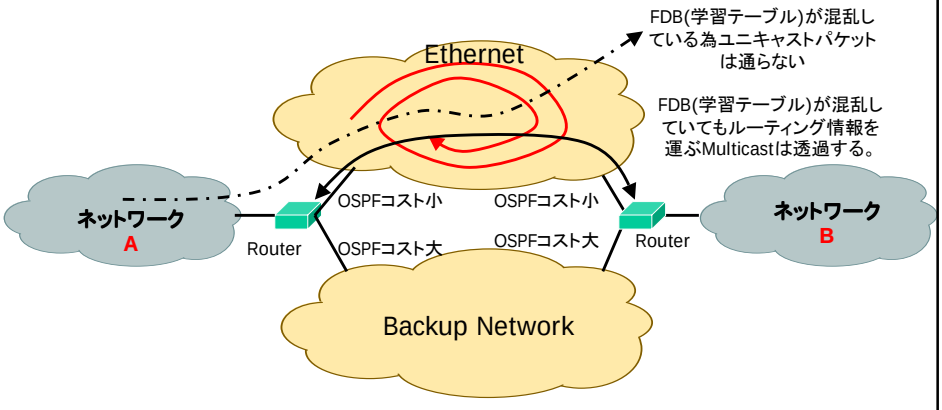
•広域イーサネットや大規模なイーサネットを利用し重要なサービスを動作させるような場合、バックアップの通信経路を持たせるような事がしばしば行われる。
•一般には、OSPFやEIGRPなどをそのまま動作させる方法が取られる。



※ルータが二重化されていないのはここでは無視する。

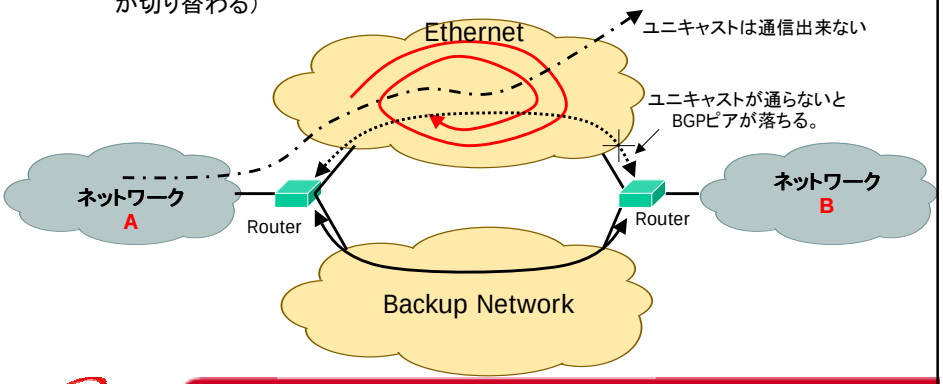
大規模イーサネットを介した冗長とループ

図のEthernet部分でループが発生した場合、
FDBの内容が狂う為、条件によっては、マルチキャストは通るが、ユニキャストが通らないと言う状況が発生する。OSPFはルータ間のプロトコルのやり取りをMulticastで行っている為、Ethernet1側でループによって接続性が不安定になっている事を検出出来ず、結果的に、ネットワークAからネットワークBへの通信に影響を与える可能性がある。



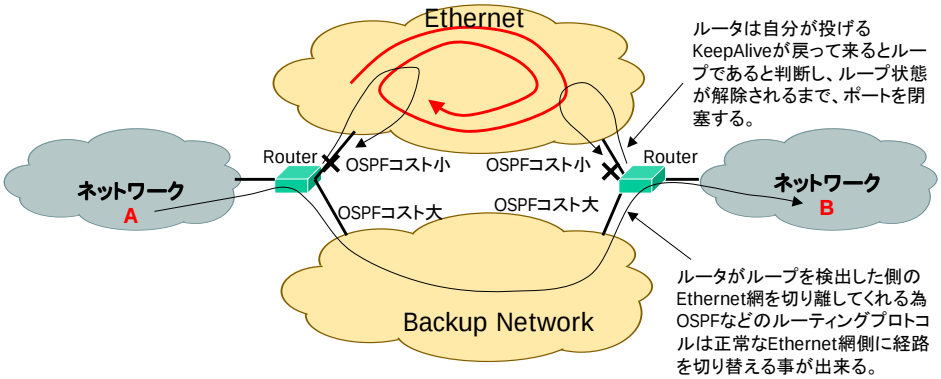
大規模イーサネットを介した冗長とループ

- 拠点数が少ない場合は拠点間でBGPピアを張る場合もある。
- BGPで張る場合のメリット
  - Ethernet網の中間で切れた場合に、検出しやすい。
  - ループの発生によって、使えなくなっているEthernetを検知出来る場合がある。（BGPはユニキャストでルータ間にTCPピアを張るので、FDBが狂っている場合にピア自体が落ちてくれる場合がある、そうするとBACK Up側にネットワークが切り替わる）



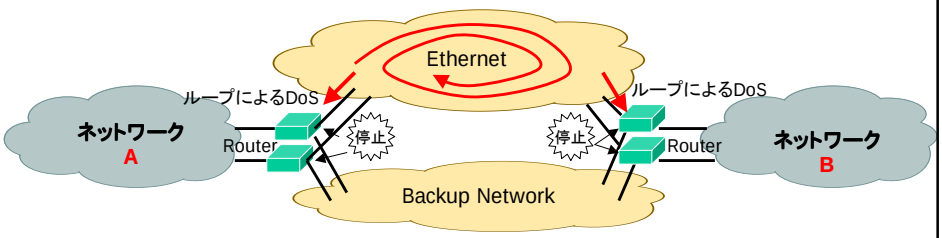
大規模イーサネットのループを検出し経路を切りかえる

例えばCiscoの一部のL3スイッチの機能には、Keep Aliveを定期的にEthernet網側に投げ、そのKeep Aliveが折り返して来た場合にそのEthernet網がループを発生させていると判定してポートに閉塞をかける機能があるものがある。（ループが解除されてしばらくたつとポートの閉塞を解除するような機能もあるものもある）このような機能を利用する事により、複数のEthernet網を使って、冗長を組んだ場合に、ループの発生を検出して、経路の切り替えを、効果的に行う事が出来る可能性がある。このような機能は、ループによる、フレーム増殖より、アプリケーションを守ると言う側面もある。
※ただし、Ethernet網側では、Keep Aliveを透過するような設定をしておかなくてはならない。

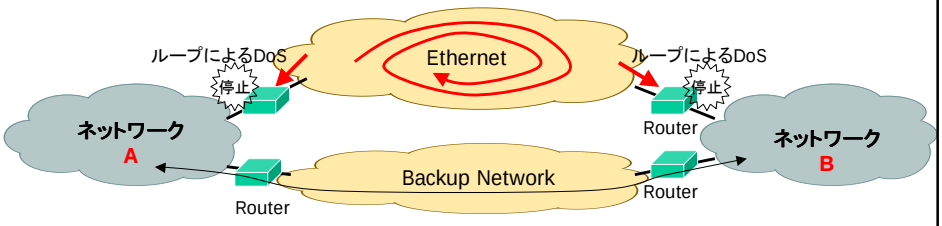


大規模イーサネットでのループによるルータ停止対策

•ループによって増殖したフレームでルータが死ぬ場合もある。

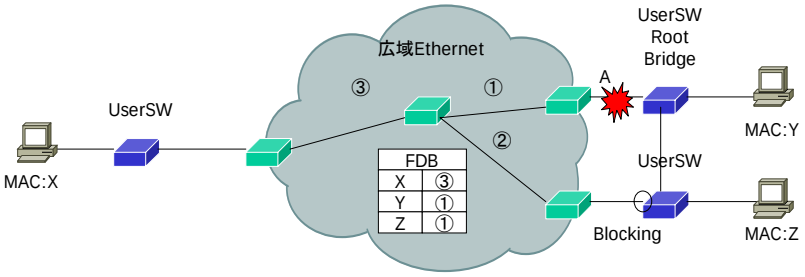


•一箇所のループで、影響を受けないルータの組を考えたほうがいい。



広域イーサネットをまたいだSTP

- 広域イーサネットをまたいだSTP
  - 全てのキャリアが対応しているわけではない。(あんまりおすすめじゃない)
  - 広域イーサネットのスイッチが、ユーザSTPの切り替えを認識しない為切り替え動作にFDBが Age outするまで(一般には5分)待たなくてはならない。(広域イーサネットに接続された機器が定期的にマルチキャストやブロードキャストを送信していればこの問題はある程度解決する)

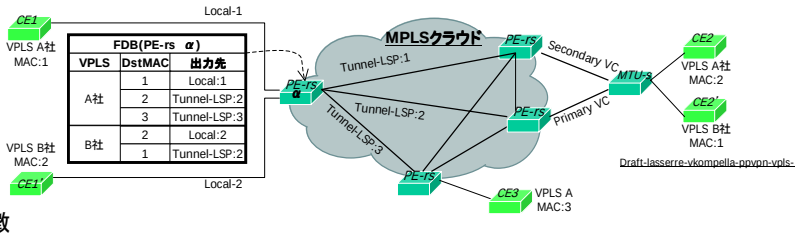


このような構成でAの部分が切断されると、UserSWのトポロジー変更が上手くいったとしても、広域Ethernet網内のBridgeTableが古い状態のまま保持されてしまう為、AgeOutするまで通信が出来なくなってしまう事がある。

付録1：VPLS（Virtual Private LAN Service）
ルーティングを使ってイーサネットの冗長を実現する例

VPLS(Virtual Private LAN Service)

- **EthernetフレームをMPLSを使ってMultipoint to Multipointで転送する技術**
  - PE間でフルメッシュLSPを張り、ブリッジングはPEで行い、MPLSのコアのではラベルスイッチングのみを行う。

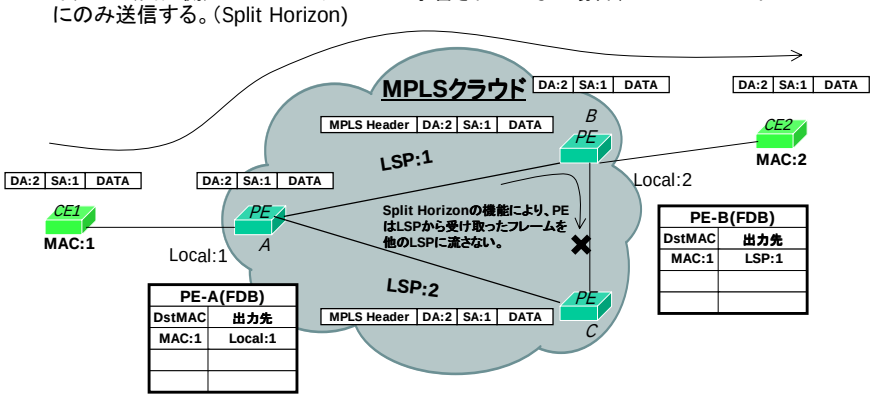


- 特徴
  - PE間でフルメッシュにトンネルを張る。
  - PEは関連したトンネルに関してMACの学習機能を持つ。
  - MPLS網側から受け取ったフレームをMPLS網に戻さない、**Split Horizon** の機能により、Loopを防止する。
  - MPLSベースの強力な冗長化機能が使える。

VPLS(Virtual Private LAN Service)基本動作(1)

FDBにMACアドレスが学習されていない場合のVPLS上のフレーム転送

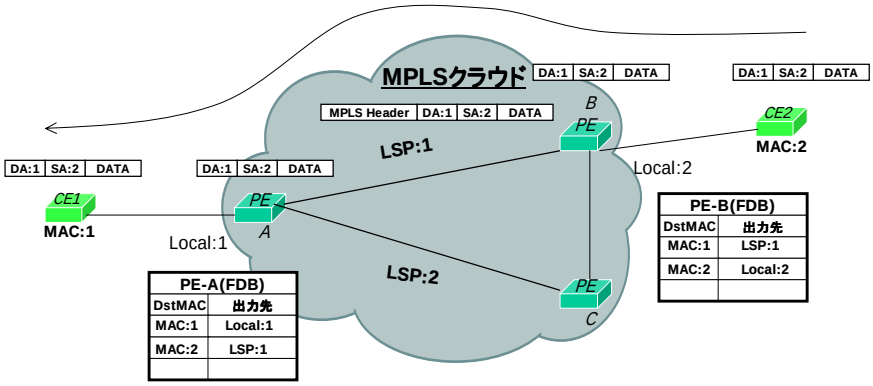
- イングレス（入力側）PEにてMACアドレスが学習されていない場合、PEはフレームをフラッドする。
- イグレス（出力側）PEにてMACアドレスが学習されていない場合、PEはローカルポートにのみ送信する。（Split Horizon）



VPLS(Virtual Private LAN Service)基本動作(2)

FDBにMACアドレスが学習されている場合のVPLS上のフレーム転送

- PEのFDBにMACアドレスが学習されている場合は、学習の内容にそって、フレームが転送される。

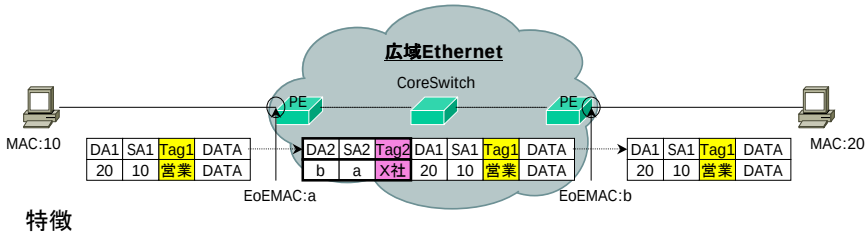


付録2：Ether over Ether
広域イーサネットサービスで利用されている拡張イーサネットの例とループ対策

階層化ブリッジング(Ether over Ether)

802.1Q Tag VLANを使ったVLAN VPNの改良方式

- PEの加入者向けポートそれぞれにユニークなEoEMACアドレスを定義し、加入者から受け取ったEthernetフレームをその入力ポートに定義されたEoEMACアドレスをソースとし送り先のPEのポートのEoEMACアドレスをディスティネーションとするEthernetフレームでカプセル化して転送する方式。

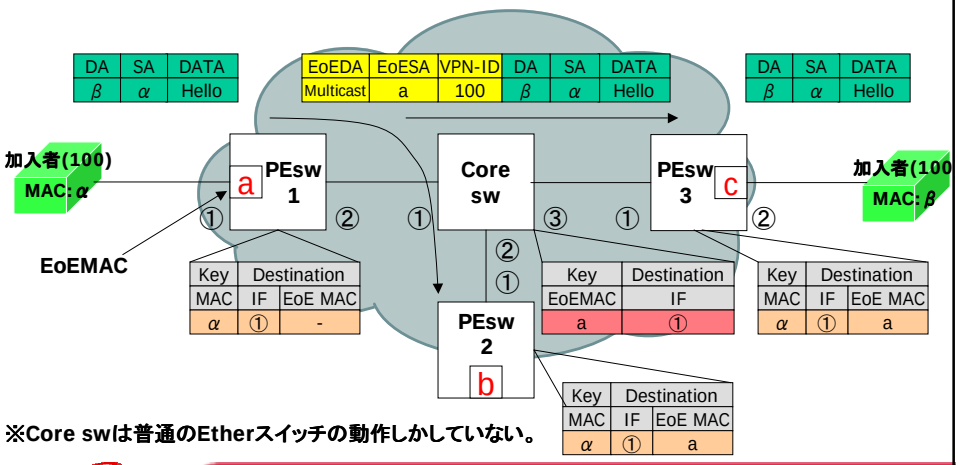


- 特徴
  - コアスイッチで学習しなくてはならないMACアドレスを劇的に減らす事が出来る。
  - EoEMACアドレスを階層的に割り振る事により、ループトラフィックが防止出来る。
  - 特殊な処理を意味するあて先MACアドレスを持つパケットを安全に転送する。
  - コアスイッチは単にジャンボフレームを転送出来る普通のスイッチでかまわない。（過去の資産の継承）

EoE(Ether over Ether)基本動作(1)

FDBにMACアドレスが学習されていない場合のEoE上のフレーム転送

- イングレス（入力側）PEにてMACアドレスが学習されていない場合、PEはEoEDAにマルチキャストアドレスをセットしフレームをフラッドする。

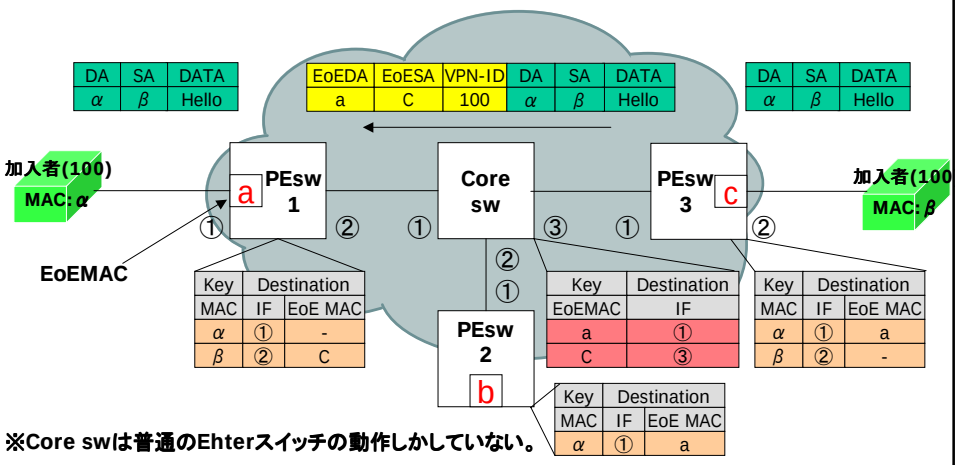


※Core swは普通のEtherスイッチの動作しかしていない。

EoE(Ether over Ether)基本動作(2)

FDBにMACアドレスが学習されている場合のEoE上のフレーム転送

- FDBにMACアドレスが学習されている場合は、学習の内容にそって、フレームが転送される。



※Core swは普通のEhterスイッチの動作しかしていない。

EoEフレームフォーマット

EoEフレーム構造(Etag付きの場合)

(802.1Qにカプセル化する場合)

7bytes	1bytes	6bytes	6bytes	2bytes	2bytes	2bytes	48～1520bytes or more	4bytes
プリアンブル	SFD	宛先EoE MACアドレス (DstEoE)	送信元EoE MACアドレス (SrcEoE)	TPID	TCI	EoE TPID	ペイロード (EoEフレーム)	FCS
プリアンブル								

※1: TPIDについて、プロバイダ内でどの値を利用するかは任意である。802.1Qをその使用するのであれば0x8100を使う事も出来るが、QinQやvMANで一般に0x9100が使われたように、EoEでは一般には0xA100が用いられる。

※1

16bits	3bits	1bits	12bits
0x A100	Pri	CFI	Tag-ID

16bits
0x E0E0

※2

※2:EoEデータフレームのTPID/Typeは0xE0E0になる。これにより、EoEカプセルフレームを判定する

(UntagEthernetフレームにカプセル化する場合)

7bytes	1bytes	6bytes	6bytes	2bytes	48～1520bytes or more	4bytes
プリアンブル	SFD	宛先EoE MACアドレス (DstEoE)	送信元EoE MACアドレス (SrcEoE)	TYPE	ペイロード (EoEフレーム)	FCS
プリアンブル						

EoE フレーム

※3: TTL(Time to Live)はEoE Awareなスイッチを通過するたびに減算され、0になったフレームは転送されず破棄される。(最大255)

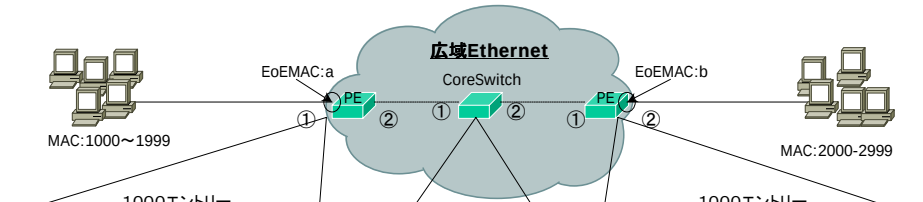
※4:Etag(Extension Tag) オプション
拡張Tagは802.1QがサポートするTag-ID以上のVLAN空間をサポートする為に用いる。
(最大12bits+8bits=20bitsの空間をサポート出来る)

1bytes	1bytes	46～1518bytes or more
EoE TTL	ETag	ペイロード (加入者フレーム)

※3 ※4

EoEによる、CoreSwitchでのMAC学習の低減

- 加入者が使用するMACアドレスの量が増えたとしても、コアスイッチが学習するMACアドレスの量は変わらない。(コアスイッチは数千VPNを扱う為、VPNごとのMAC学習数を減らしたい。)
- エッジスイッチでは、多くてもポート数程度の数のVPNしか存在しない為、スケールし易い。



1000エントリー

Key	Destination		Key	Destination	
MAC	IF	EoE MAC	MAC	IF	EoE MAC
1000	①	-	2000	②	b
1001	①	-	2001	②	b
...	①	-	...	②	b
...	①	-	...	②	b
1999	①	-	2999	②	b

2エントリー

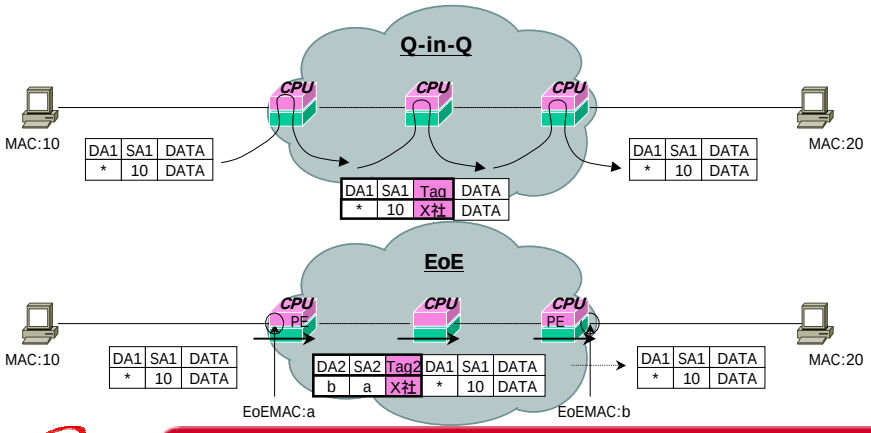
Key	Destination
EoEMAC	IF
a	①
b	②

1000エントリー

Key	Destination		Key	Destination	
MAC	IF	EoE MAC	MAC	IF	EoE MAC
1000	①	a	2000	②	-
1001	①	a	2001	②	-
...	①	a	...	②	-
...	①	a	...	②	-
1999	①	a	2999	②	-

EoEによる制御パケットの安全な透過

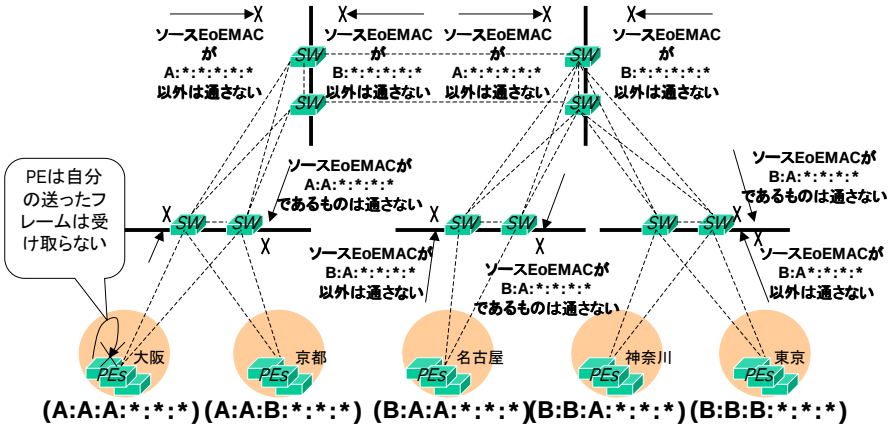
- Q-in-Qでは、特定の制御プロトコルと同じマルチキャストアドレスを宛先に持ったフレームをユーザが送信すると、スイッチのCPUに転送されたり、ブロックされる場合がある。
- EoEではユーザが送信したマルチキャストのアドレスは隠蔽されるので、CPUに転送されず、透過する。



EoE階層化MACアドレッシングによるループ防止(1)

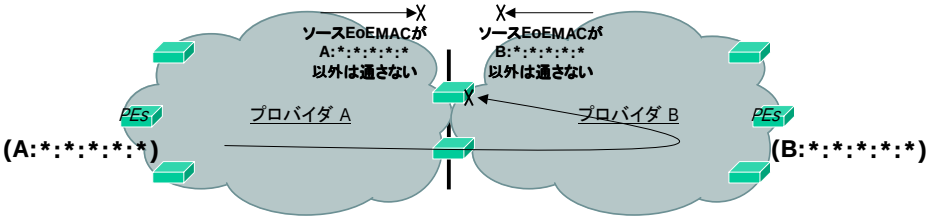
EoEMACアドレスを階層的に割り振る事により、ループの発生を防止する

- EoEMACアドレスを階層的にふり、マスク付きMACアドレスフィルタを使って、ループを防止出来る。(ストリクトなフィルタの例)

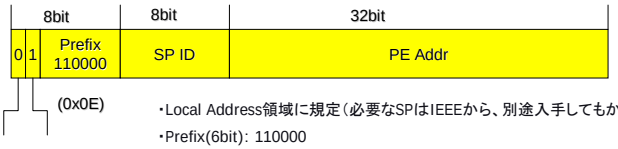


EoE階層化MACアドレッシングによるループ防止(2)

- サービスプロバイダ相互接続点



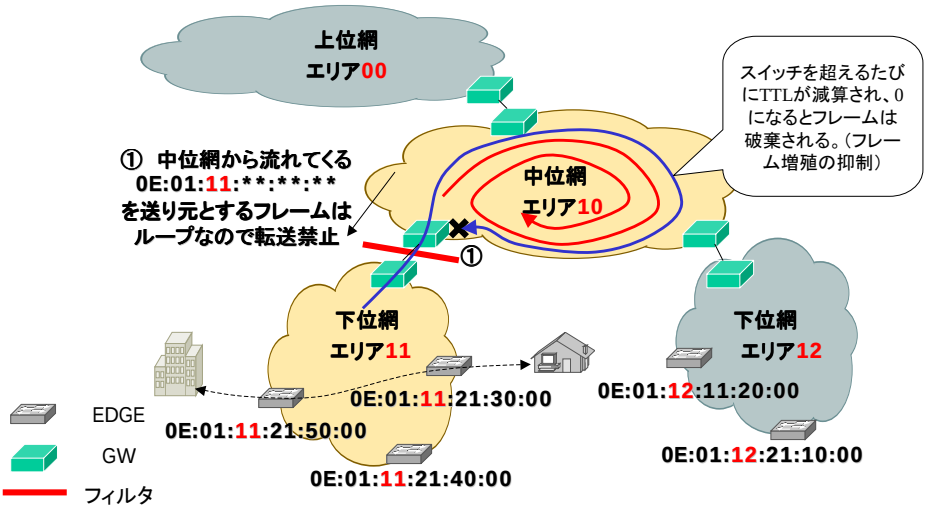
サービスプロバイダEoE MACアドレスの構造



・Local Address領域に規定(必要なSPはIEEEから、別途入手してもかまわない)
・Prefix(6bit): 110000
・SP ID(8bit): Service Providerの識別番号
・PE Addr: Provider Edgeスイッチのアドレス
PEスイッチそのもの、又はPEスイッチの内部の1エンティティ(ラインカードやポート)を表す場合が有る (使用方法はService Provider毎に任意)

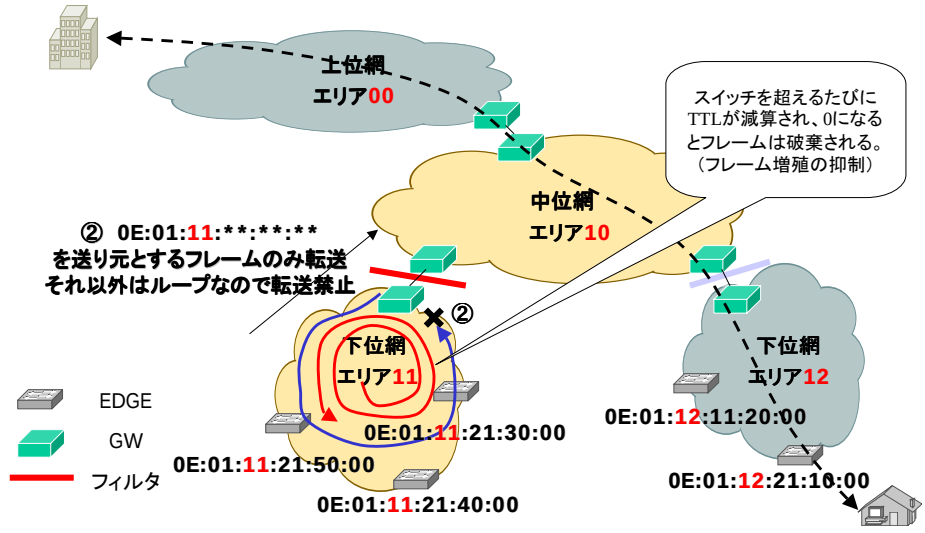
ループ発生時の流入防止(ルーズなフィルタの例)

中位網でループが発生しても下位網内部の通信には影響を与えないようにフィルターを設定する。



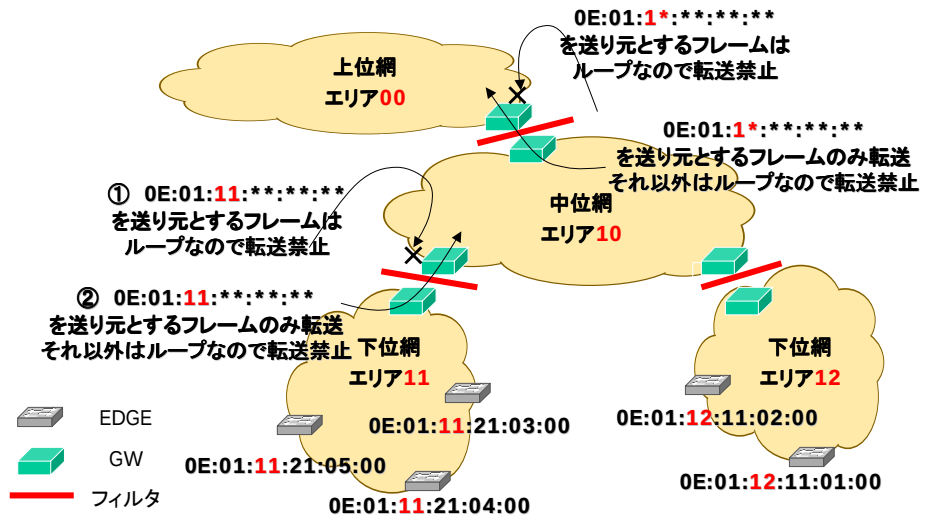
ループ発生時の流出防止（ルーズなフィルタの例）

下位網でループが発生しても他網の通信には影響を与えないようにフィルターを設定する。



階層エリアのフィルタ（ルーズなフィルタの例）

2種類のフィルタの組み合わせにより、ループの影響範囲の限定を行ってみる例。



EoEでのループ位置検出

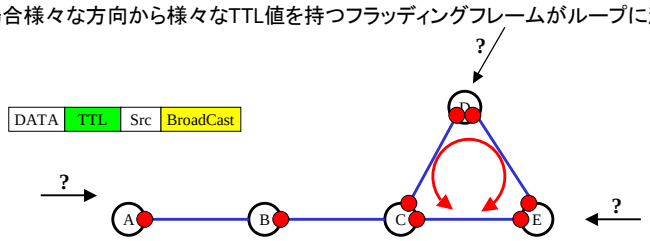
- EthernetなどのMACブリッジングにおいてループを検出する方式はいくつかある
  （1）FDBの書き換わり回数によってループを検出する方法。
  （2）マルチキャストなどのフレームを一定方向に送出しそれが戻ってくるのを観測する事によりループの方向と発生を検出する方法。
  （3）トラフィックに含まれるフラッディングトラフィックの量をモニタする方法。
  （4）TTLをexpireが発生する事によりループの検出を行う方法（拡張イーサネットにおいて）

これらの方法ではループが発生した事の検出は行えるが実際にどのノードがループしているかは判定出来ない！

ループが発生するとループを形成しているノードを発信元として、増殖されたマルチキャストやブロードキャストが多量に送信される為、ループ発生後に多量のトラヒックが流れ出す方向に探索を進める事によりループ箇所を検出する事が出来るが、スイッチの数が多いと時間がかかる。

EoEでのループ位置検出

ループの発生位置検出は既存のEthernetでは困難な場合があったが、
EoEの場合様々な方向から様々なTTL値を持つフラッディングフレームがループに流れ込んでくると・・・



2ポート以上でTTL＝0による廃棄が発生するとループノード

•ループを構成するノードでは、2つ以上のポートから入力されたフレームにてTTL Expireによる、フレーム廃棄が発生する。（ループを構成するノードでは右回りのフレームが流入するポートでも左回りのフレームが流入するポートでもTTL Expireによるフレーム廃棄が発生するため）

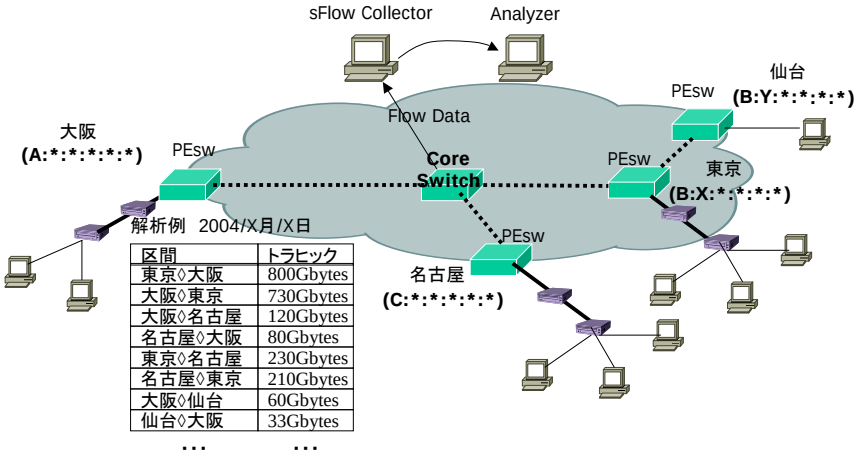
1ポートだけでTTL＝0による廃棄が発生するとループノードではない

•直接ループを構成しないノードでは、ループの発生方向に向いたポートに入力されるフレームにてTTL Expireによるフレーム廃棄が発生する。（ループ方向から入力されるフレームのみが、ループ内を回るうちにTTLを減らしている可能性がある為）

**通信事業者網内でのループであればこのトラップの上げ方を
変える事により瞬時にループしているノードを検出出来る。**

EoE階層化MACアドレッシングとsFlow（NetFlow）

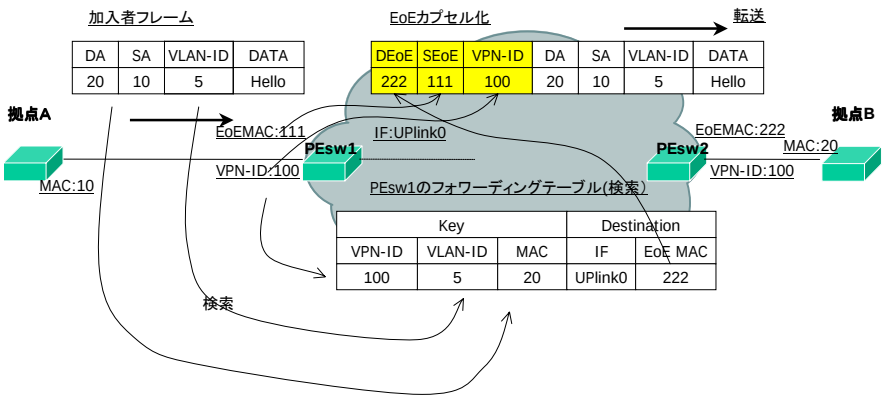
- 既存の方式では網内のMACアドレスがユーザのMACアドレスそのものであったため、そのトラヒックがどの拠点間の通信なのかを特定することが不可能
- EoEの階層化MACアドレッシングとsFlowの様なトラヒック分析システムを組み合わせることで、対地毎のトラヒックの流れを分析する事が可能となる



区間	トラヒック
東京◊大阪	800Gbytes
大阪◊東京	730Gbytes
大阪◊名古屋	120Gbytes
名古屋◊大阪	80Gbytes
東京◊名古屋	230Gbytes
名古屋◊東京	210Gbytes
大阪◊仙台	60Gbytes
仙台◊大阪	33Gbytes
…	…

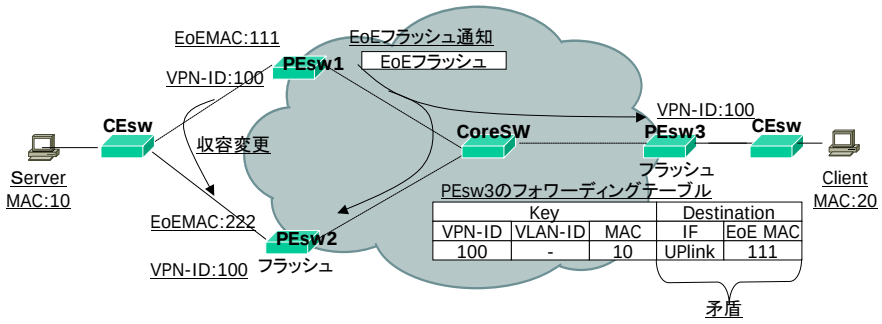
EoEとTag学習について

- 一般にQ-in-Qでは、加入者が設定したVLAN内で、MACアドレスの重複は許されない。
- EoE では、エッジでMACと共にTag-IDを学習する事によって、加入者の設定したVLAN内でのMACの重複を許容する方式を導入。



EoE OAM

- EoE Ping
  PE SW間あるいはEoE終端機能を持った装置間のPing（疎通確認）
- EoE Traceroute
  TTLを用いて EoE経路のトレースを行う（専用の中継Ethernet SWが必要）
- EoE フラッシュ
  EoEのFDBの全部又は一部のエントリのクリア要求をPEスイッチでやりとり



Questions？

masaty@pwd. ad. jp