

# IETF98 RTG関連報告

KDDI 株式会社  
宮坂 拓也

# 自己紹介

---

- 宮坂 拓也 (みやさか たくや)
- 所属
  - KDDI株式会社 (AS2516)
- お仕事
  - MPLSバックボーンネットワークのネットワークエンジニア
- IETF
  - Routing Areaに生息しています
    - mpls, teas, pce, spring, idr, ospf, rtgwg, sfcとか

# 今回報告するもの

---

- **報告する基準・内容**

- 各WGが何をしているのか・どういうステータスにあるのか

- **MPLS WG**

- MPLSに関する全体的な提案を行うWG

- **TEAS WG**

- MPLS Traffic Engineeringに関する提案を行うWG

- **PCE WG**

- PCEに関する提案を行うWG

- 参考：<https://www.slideshare.net/TakuyaMiyasaka/pce-mplssdn>

- **SPRING WG**

- Segment Routingに関する提案を行うWG

# 今回報告するもの

---

- **報告する基準・内容**

- 各WGが何をしているのか・どういうステータスにあるのか

- **MPLS WG**

- MPLSに関する主体的な提案を行うWG

というわけでMPLS関係の発表です！

- **TEAS WG**

- MPLS Traffic Engineeringに関する提案を行うWG

- **PCE WG**

- PCEに関する提案を行うWG

- 参考：<https://www.slideshare.net/TakuyaMiyasaka/pce-mplssdn>

- **SPRING WG**

- Segment Routingに関する提案を行うWG

# MPLS WG

---

- どんなことをやっているWG?

- MPLSに関する全体的な提案を行うWG

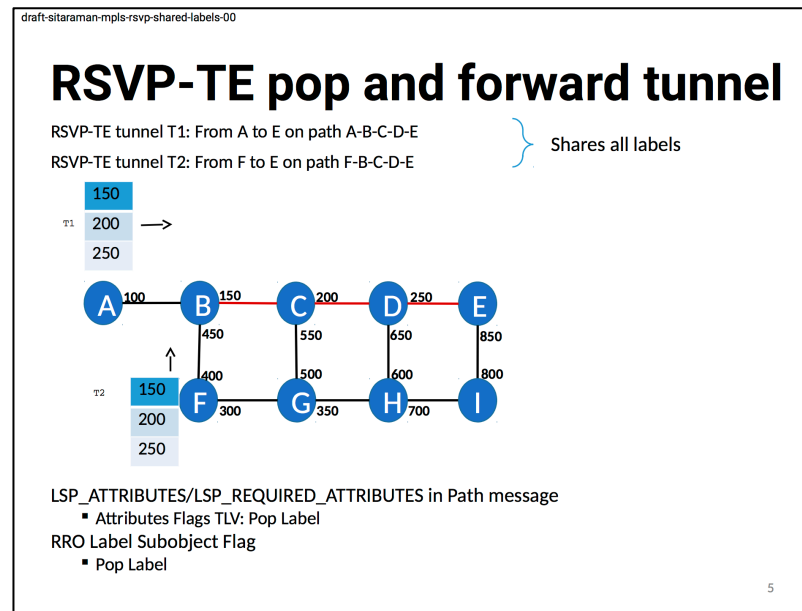
[MPLS WG Charter]

The MPLS working group is responsible for standardizing technology for label switching and for the implementation of label-switched paths over packet based link-level technologies.

- MPLSは成熟したものであるため、「新MPLS！」みたいな流れはない(YANGはもちろん別)
- なので、各々が単発的にMPLSを利用した新規提案をしていることが多いイメージ
  - 本報告ではイメージが伝わるようにその一例を紹介します

# Signaling RSVP-TE tunnels on a shared MPLS forwarding plane

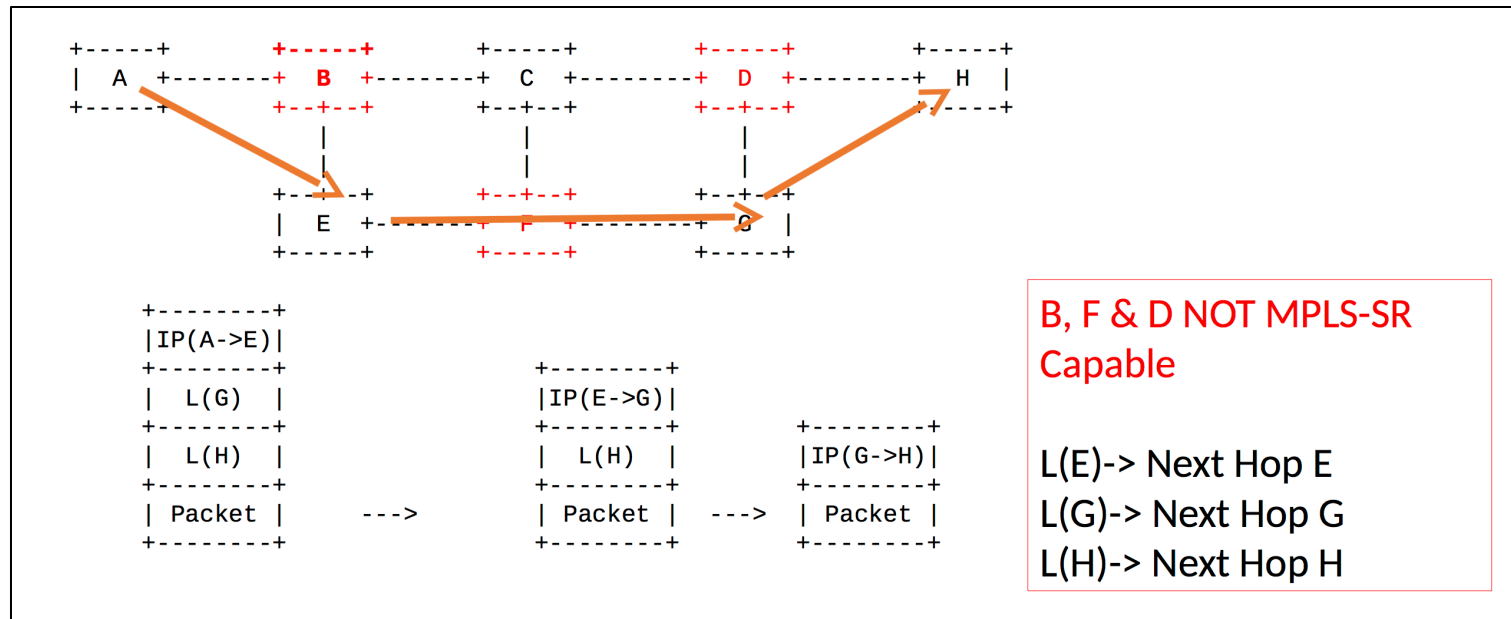
- draft-sitaraman-mpls-rsvp-shared-labels-00
- RSVPでシグナルされたLSPがとて多くなると、ルーターの利用できる最大ラベル数を超過してしまうのでは？
  - 同じリンクを通るなら同じラベルをRSVPにて広報すればいい
  - Ingress LSRはRROで書かれた全てのラベルをスタックしてパケットを送信し、Transit LSRは対応するTop LabelをPOP
- 個人的にはRSVPでやらずに、SRでいいのでは・・・？な感じ



<https://www.ietf.org/proceedings/98/slides/slides-98-mpls-sessa-05-mpls-wg-slides-98-mpls-draft-sitaraman-mpls-rsvp-shared-labels-00.pdf>

# MPLS-SR over non-MPLS nodes

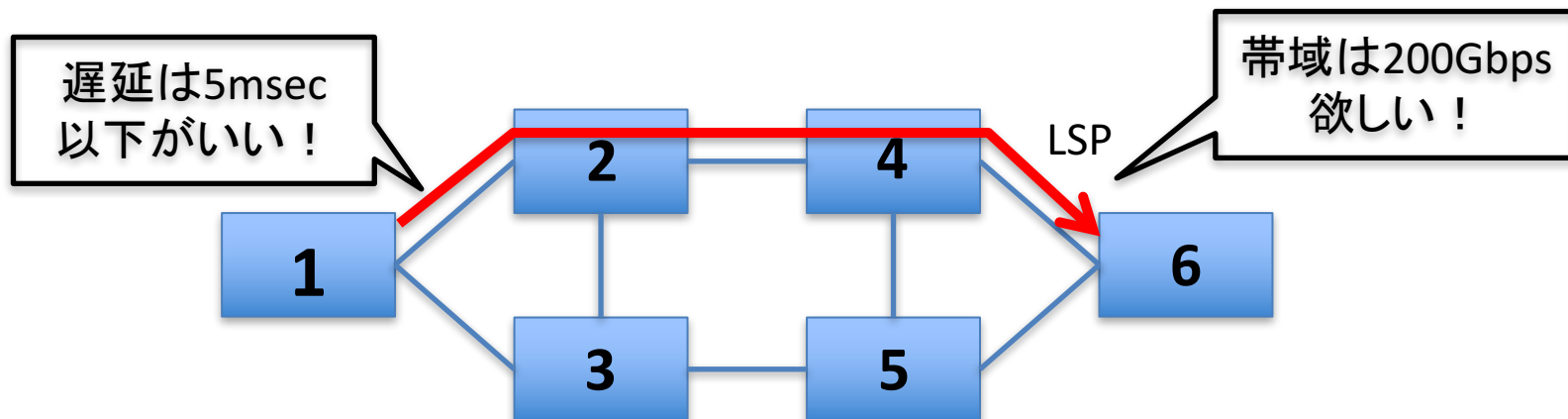
- <https://tools.ietf.org/html/draft-xu-mpls-unified-source-routing-instruction-00>
- MPLS capableでないノードがあってもMPLS-SRを動作できるようにする提案
- MPLS over IP / MPLS over UDPはすでにRFC化されているのでそれを利用するだけでよいとのこと



# TEAS WG

## • どんなことをやっているWG?

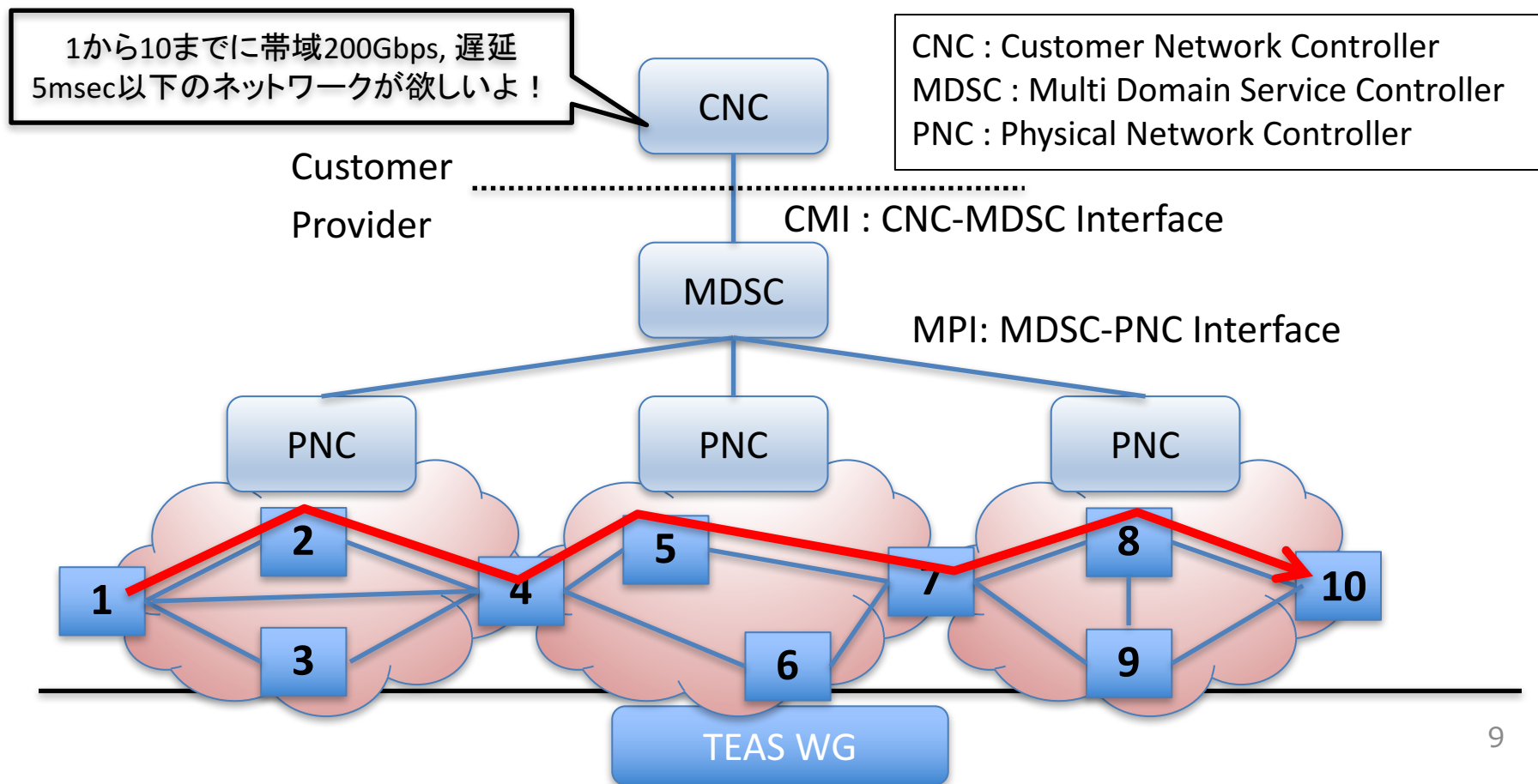
- MPLS Traffic Engineeringに関する提案を行うWG
  - Traffic Engineering Architecture and Signaling
- 下図のような、帯域・遅延といった制約を満たすMPLS LSPを確立するためにはどうすればいいか? という問題を解決するWG
- 皆さんに馴染みがある(はずの?) **RSVP-TE**はココ





# ACTN

- ACTN : Abstraction and Control of Traffic Engineering (TE) Networks
  - TEネットワークの抽象化・制御を実施
  - ACTN draftでは全体のアーキテクチャを定義し、各InterfaceをTEASやPCEなどのWGにおいて拡張している

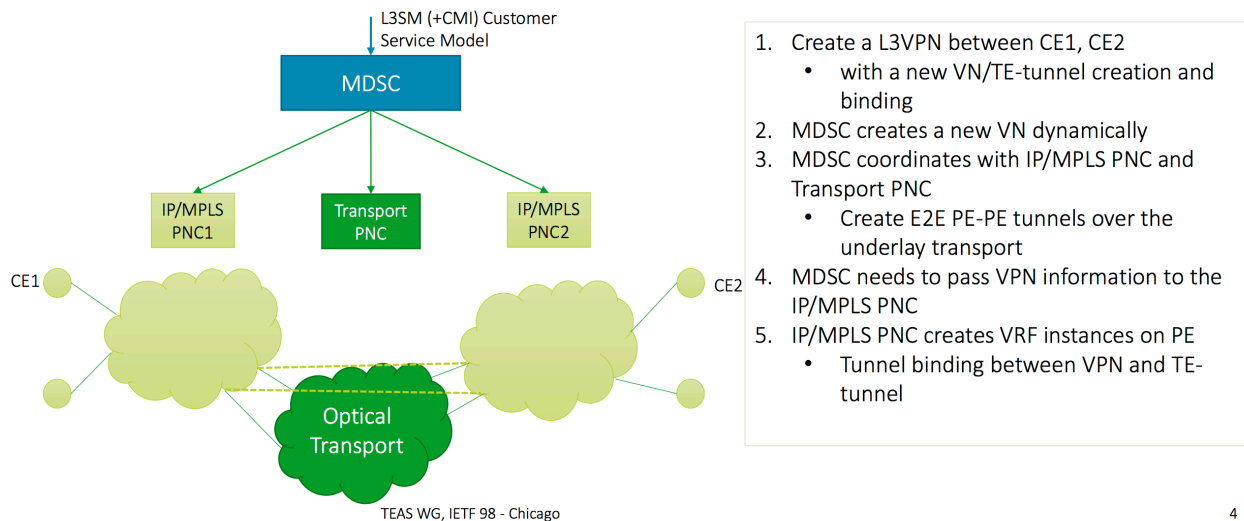


# TE / Service Mapping

- Traffic Engineering and Service Mapping Yang Model
  - **draft-lee-teas-te-service-mapping-yang-00**
  - L3SMといった**Service Model**とTE LSP/ACTN VN(Virtual Networkといった**Transport**を紐づける(YANGで)
  - CustomerからサービスとTransport Networkの**確立**をACTN frameworkにおいて実現



## Sample Flow

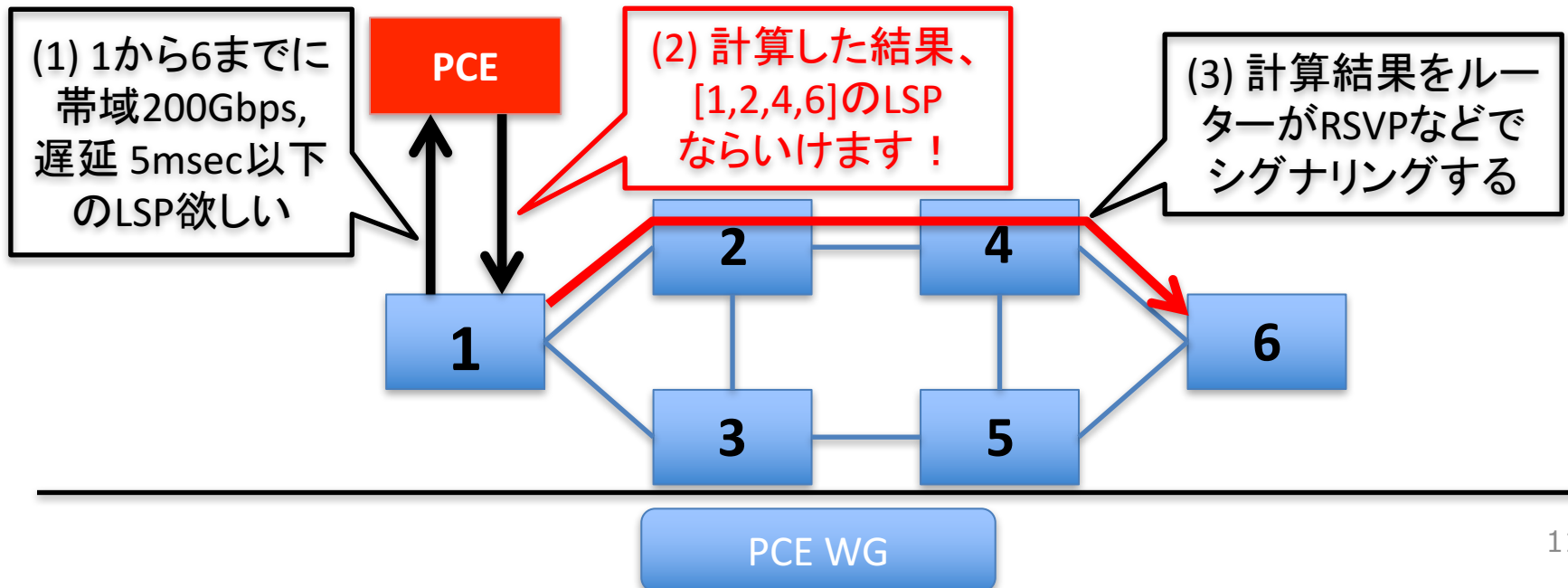


[https://www.ietf.org/proceedings/98/slides/slides-98-teas-09\\_te\\_service\\_mapping\\_yang-02.pdf](https://www.ietf.org/proceedings/98/slides/slides-98-teas-09_te_service_mapping_yang-02.pdf)

# PCE WG

## • どんなことをやっているWG?

- PCE(Path Computation Element)に関する提案を行うWG
  - 参考：<https://www.slideshare.net/TakuyaMiyasaka/pce-mplssdn>
- 下図のように、MPLS LSPの経路計算をIngress LSRではなく、外部コントローラー(PCE)が計算する
- SDNの流行(?)につれて、PCEがMPLS界のSDN Controllerとする流れが近年盛んになっている



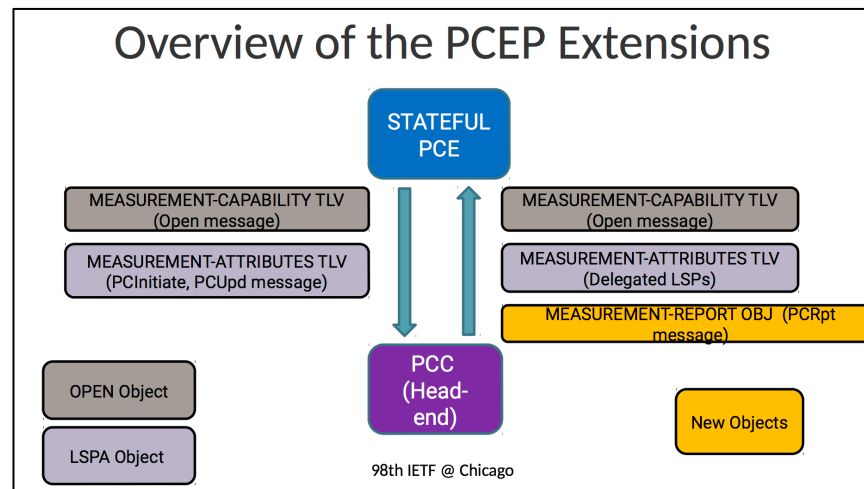
# Stateful PCE

---

- PCEは現在 2 種類存在する
  - Stateless PCE (RFC 5440)
    - TEDBは対象ネットワークと同期 (BGP-LSとかで)
    - PCEはネットワーク内のLSP情報を管理しない
    - PCEはパス計算のリクエストを受けたら答えるだけ
  - Stateful PCE (ietf-pce-stateful-pce)
    - PCEはTEDBだけでなく、ネットワーク内のLSP情報を管理する
    - PCEから以下のようなパス確立要求も行える
      - 新規LSPの確立
      - 既存LSPのパス変更
- Stateful PCEはIETFにおけるSDNコントローラー的な位置付けとなっており、近年盛んに様々な標準化が進んでいる状況
  - なので、近年のPCE WGではPCEをSDN的に使うときに、こういう拡張があるといいよね！的な提案が多いです

# TE LSP Performance Metric Reporting

- **draft-gandhi-pce-pm-07**
- PCEPでMPLS-TE LSPの遅延・パケロス率・利用帯域といった品質情報を送る提案
- BGP/OSPF/IS-ISでは上記機能が存在するので、PCEPでも！という提案。
  - 最近は、何でもかんでも(?)PCEPでやろう系なものも多い。（例えばBGP-LSあるんだからPCEP-LSも！とか）



<https://www.ietf.org/proceedings/98/slides/slides-98-pce-33-pcep-extensions-for-pm-reporting-00.pdf>

# Stateful PCE and ACTN

- Stateful PCEをACTNに用いる
  - Stateful H-PCE and ACTN
    - 各NWドメインに子PCEを置く(これがPNC)
    - それぞれのPCEを束ねる親PCEを中央に配置する(これがMDSC)
    - Customerからの要求をもとに、親PCEが子PCEにLSPの確立要求メッセージをPCEPで広報する

### Applicability of PCE for ACTN

- ACTN – Abstraction and Control of TE networks
- Four Functions in ACTN
  - Multi domain coordination function
  - Virtualization/Abstraction function
  - Customer mapping/translation function
  - Virtual service coordination function
- We list all ways PCEP could be used

CNC	Customer Network Controller
MDSC	Multi-Domain Service Coordinator
PNC	Physical Network Controller
CMI	CNC-MDSC Interface
MMI	MDSC-MDSC Interface
MPI	MDSC-PNC Interface

PCE WG, IETF 98 Chicago

### Stateful H-PCE

- H-PCE + Stateful PCE
- Hierarchy of Stateful PCE
- -00 version was discussed in the IETF 95 (BA)
- Marked 'Informational'

Maintain the domain topology map and LSPDB

Child Stateful PCE with per domain TEDB and LSPDB

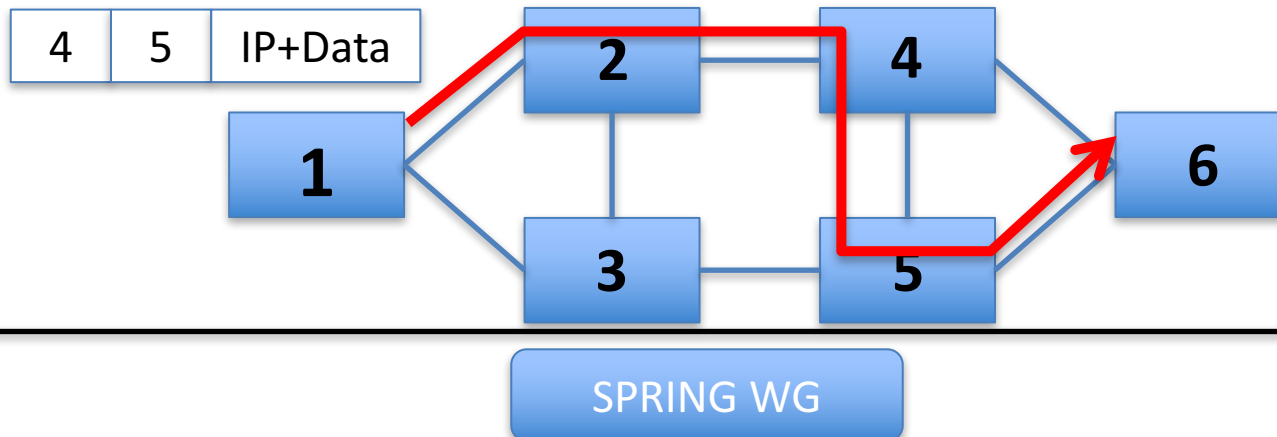
PCE WG, IETF 98 Chicago

<https://www.ietf.org/proceedings/98/slides/slides-98-pce-25-stateful-h-pce-and-actn-00.pdf>

# SPRING WG

## • どんなことをやっているWG?

- Segment Routingに関する提案を行うWG
- 下図のように、Ingress Routerが途中に通るべきSegmentを指定してRoutingさせる：Source Routing！
  - 参考
    - <http://www.mpls.jp/2013/presentations/sr-shitsuchi.pdf>
    - <https://www.janog.gr.jp/meeting/janog32/doc/janog32-lt-segment-kohno-01.pdf>
- LSP/RSVPといったプロトコルを使う必要はなく、OSPFv2/OSPFv3/IS-ISのIGPのみでSegmentを広報
- DataplaneとしてMPLS(Label)とIPv6 (Routing Header)の2種類を利用できる



# Close?

## • やりきったからWG閉じる？

- WGの開始時にSegment Routingに関する全体アーキテクチャに関する標準化についてはほぼ終了し、Milestoneも終わりそうなので、WGをcloseするかどうかChairから問いかけがあった

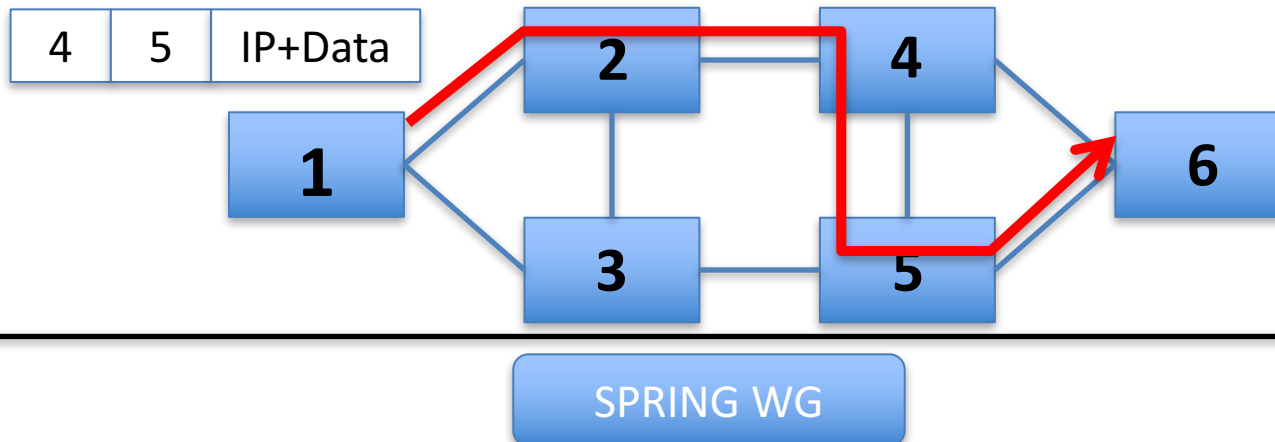
種別	RFC / I-D	状態
Problem Statement	RFC 7855	RFC
Architecture	draft-ietf-spring-segment-routing-11	IESG
IS-IS extension	draft-ietf-isis-segment-routing-extensions-12	WG LC
OSPF extension	draft-ietf-ospf-segment-routing-extensions-14 draft-ietf-ospf-ospfv3-segment-routing-extensions-09	v2:IESG v3:WG doc

- 結果としては、「クローズせずにこのWGを維持する」というものであった
  - 標準化はほぼ終了したかもしれないが、Segment Routingの実運用例はまだ多くなく、そこで出てくる問題がないかどうか確認してからにしようという意見が多数



# Segment Routing MPLS dataplane

- MPLS dataplane
  - [draft-ietf-spring-segment-routing-mpls-08](#)
  - MPLS Label(20bit)を用いてSegmentを表す
  - Label StackによってSegment Listを明示
  - 通常のMPLSの動作である、push/swap/popにてSegment Routingを実現
  - SR-TEなどのために、Label Stackしすぎて各ルーターの対応する最大スタック数を超えてしまう恐れがあるのでは？という懸念の解決策：
    - IGPで自身の最大スタック数を広報する(OSPF/IS-ISのdraftあります)
    - PCEなどのCentral Controllerを用いて各ルーターの最大スタック数を超えないようにする→(でもけっこう難しそう・・・)



# Segment Routing IPv6 dataplane

- IPv6 dataplane
  - draft-ietf-6man-segment-routing-header-06
  - IPv6 Routing Headerを用いて実現する
  - MPLSと比べて、Segmentあたりのbit長が長い(128bit)
    - Endpoint Nodeにて実施してほしい動作(Function)を埋め込むことでNetworkにProgrammabilityを与える提案が今回なされた！(draft-filsfils-spring-srv6-network-programming)
      - Linux kernel 4.10に実装済らしい！→NFV/SFCとかに使えるそう！？

```
0          1          2          3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+
| Next Header | Hdr Ext Len | Routing Type | Segments Left |
+-----+-----+-----+-----+
| First Segment | Flags | RESERVED |
+-----+-----+-----+-----+
| Segment List[0] (128 bits IPv6 address) |
+-----+-----+-----+-----+
| ... |
+-----+-----+-----+-----+
| Segment List[n] (128 bits IPv6 address) |
+-----+-----+-----+-----+
// Optional Type Length Value objects (variable) //
```



Segment Left Fieldで現在のActiveなSegmentを決定

# SRv6 Network Programming

---

- <https://tools.ietf.org/html/draft-filsfils-spring-srv6-network-programming-00>

End	Endpoint function The SRv6 instantiation of a prefix SID
End.X	Endpoint function with Layer-3 cross-connect The SRv6 instantiation of a Adj SID
End.T	Endpoint function with specific IPv6 table lookup
End.DX2	Endpoint with decapsulation and Layer-2 cross-connect L2VPN use-case
End.DX6	Endpoint with decapsulation and IPv6 cross-connect IPv6 L3VPN use (equivalent of a per-CE VPN label)
End.DX4	Endpoint with decapsulation and IPv4 cross-connect IPv4 L3VPN use (equivalent of a per-CE VPN label)
End.DT6	Endpoint with decapsulation and IPv6 table lookup IPv6 L3VPN use (equivalent of a per-VRF VPN label)
End.DT4	Endpoint with decapsulation and IPv4 table lookup IPv4 L3VPN use (equivalent of a per-VRF VPN label)
End.B6	Endpoint bound to an SRv6 policy SRv6 instantiation of a Binding SID
End.B6.Encaps	Endpoint bound to an SRv6 encapsulation Policy SRv6 instantiation of a Binding SID
End.BM	Endpoint bound to an SR-MPLS Policy SRv6/SR-MPLS instantiation of a Binding SID
End.S	Endpoint in search of a target in table T
End.AS	Endpoint to SR-unaware APP via static proxy
End.AM	Endpoint to SR-unaware APP via masquerading

# Segment Routing IPv6 dataplane

- Forwarding Example

## Endpoint Node Rule

1. IF DA = myself (segment endpoint)
2. IF Segments Left > 0 THEN  
decrement Segments Left  
update DA with Segment List[Segments Left]
3. ELSE continue IPv6 processing of the packet  
End of processing.
4. Forward the packet out

