

# Yang Data Model for TE Topologies

draft-ietf-teas-yang-te-topo-05

Github: <https://github.com/ietf-mpls-yang/te/blob/master/ietf-te-topology.yang>

Xufeng Liu (Ericsson)

Vishnu Pavan Beeram (Juniper Networks)

Igor Bryskin (Huawei Technologies)

Tarek Saad (Cisco)

Himanshu Shah (Ciena)

Oscar Gonzalez De Dios (Telefonica)

Contributors:

Sergio Belotti (Alcatel-Lucent)

Diete Beller (Alcatel-Lucent)

# Summary of Changes

- Draft re-organization.
- Added support for multi-layer topology.
- Added support for protection on tunnel termination point.
- Added support for label restrictions on connectivity matrix.
- Added support for topology optimization options.
- Enhanced support for inter-domain topology.

# Draft Re-organization

- Moved shared type definitions to `ietf-te-types.yang` in `draft-ietf-teas-yang-te`.
  - To share type definitions by multiple models.
- Moved YANG module `ietf-te-topology-psc` from `draft-ietf-teas-yang-te-topo` to `draft-liu-teas-yang-l3-te-topo`.
  - Kept `draft-ietf-teas-yang-te-topo-05` technology agnostic.
- Extended `ietf-teas-yang-te-topo` to cover SR topology and SR TE topology
  - Submitted `draft-liu-teas-yang-sr-te-topo`.
  - Will describe more at end of this presentation.

# Added Support for Multi-layer Topology




- Transitional link
  - Connects link termination points at different layers.
- Modeling abstraction
  - Added switch-layer attributes to TE Link Termination Point.
  - Added a flag to TE Link to indicate transitional.

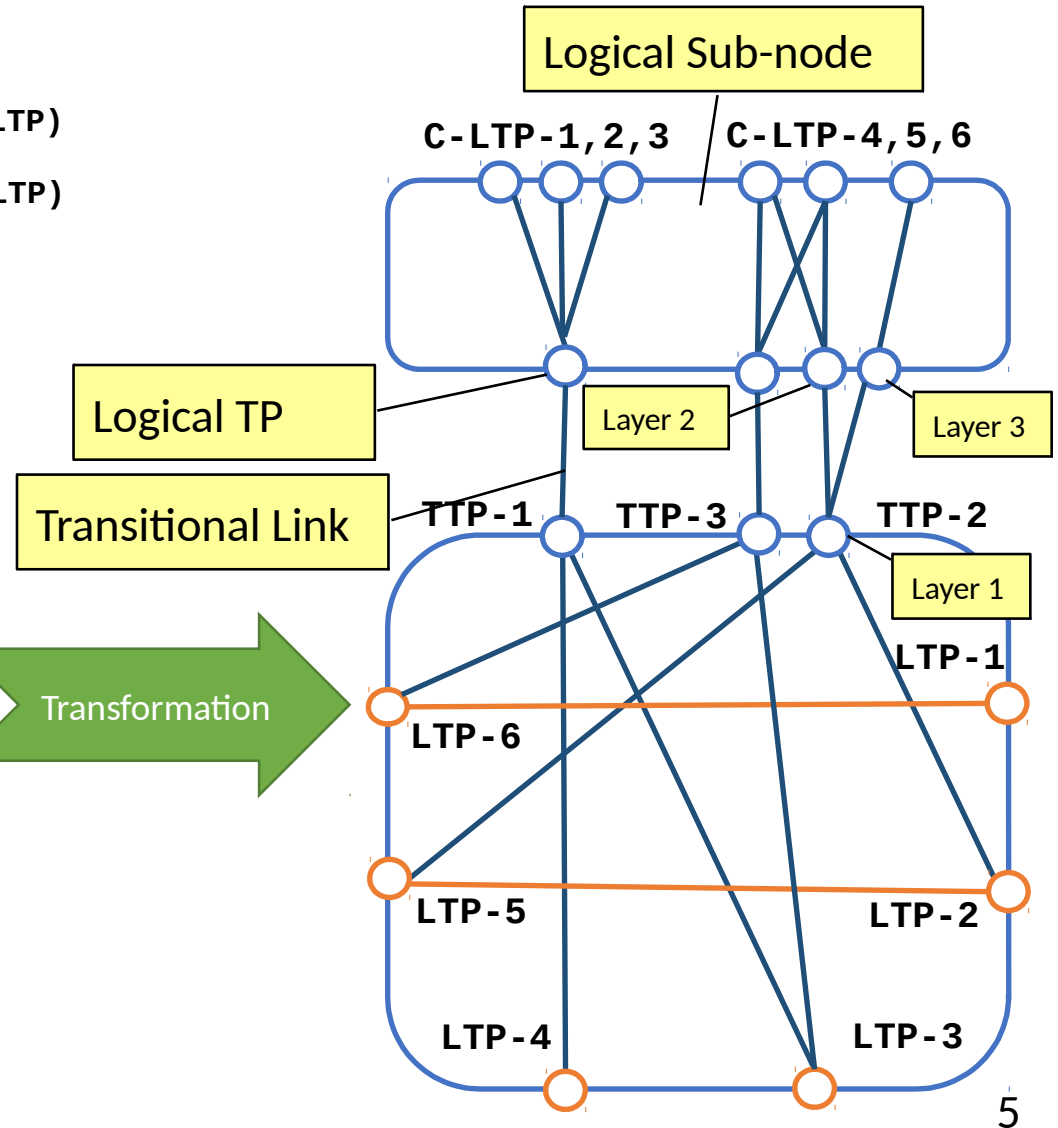
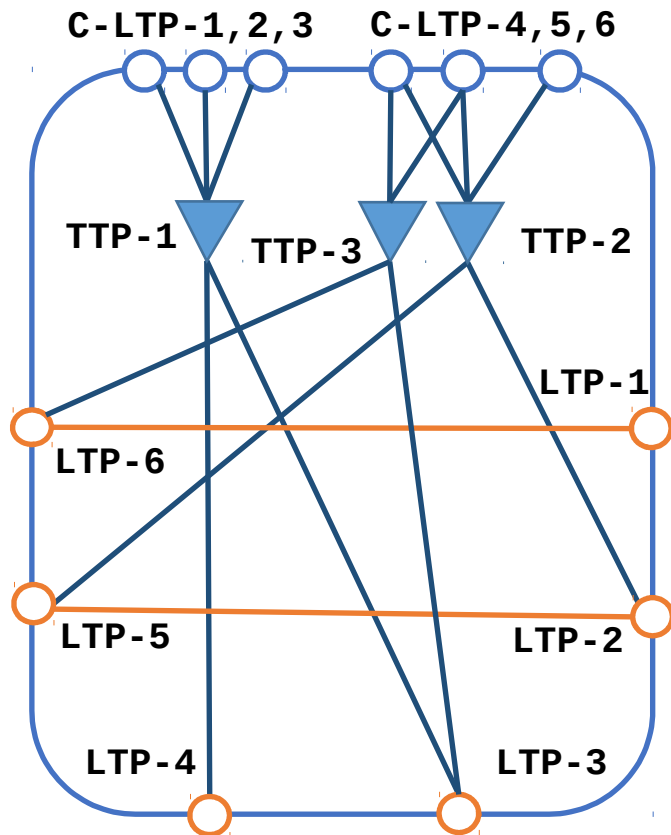
```
augment /nw:networks/nw:network/nw:node:
  +--rw te!
    +--rw tunnel-termination-point* [tunnel-tp-id]
      +--rw tunnel-tp-id    binary
      +--rw config
        | +--rw switching-capability?  identityref
        | +--rw encoding?              identityref
      +--ro state
        +--ro switching-capability?  identityref
        +--ro encoding?              identityref
```

“Layer 1” at one end,  
“Layer 2” at other end.

```
augment /nw:networks/nw:network/nt:link:
  +--rw te!
    +--rw config
      | +--rw te-link-attributes
      |   +--rw interface-switching-capability* [switching-capability]
      |     | +--rw switching-capability      identityref
      |     | +--rw encoding?                 identityref
      |     | +--rw max-lsp-bandwidth* [priority]
    +--ro state
      +--ro is-transitional?              empty
```

# Transitional Link

-  Tunnel Termination Point (TTP)
-  Server Link Termination Point (S-LTP)
-  Client Link Termination Point (C-LTP)



# Added Support for Multi-layer Topology

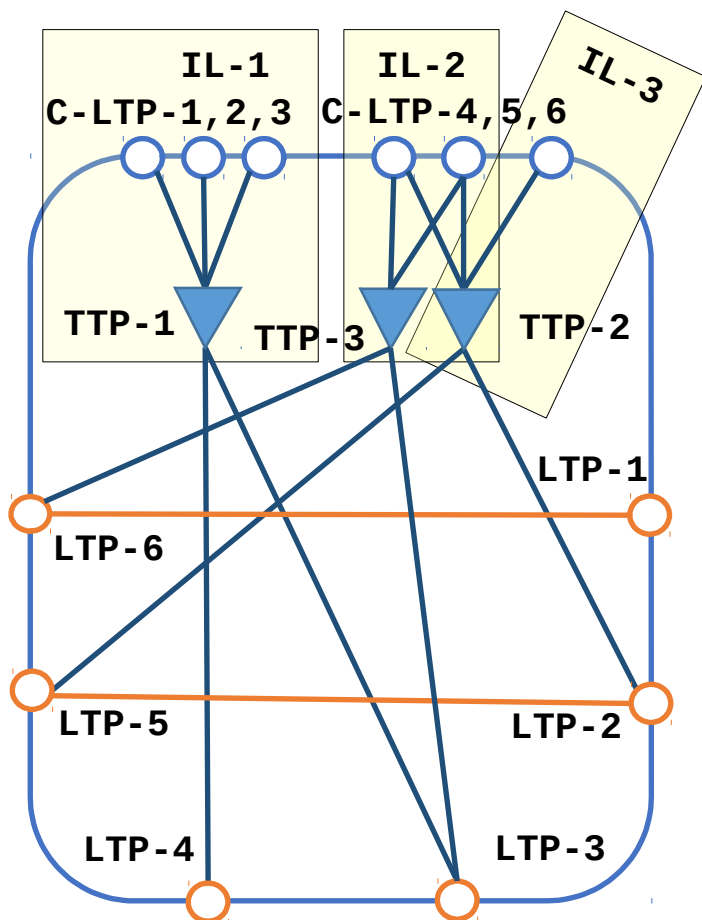
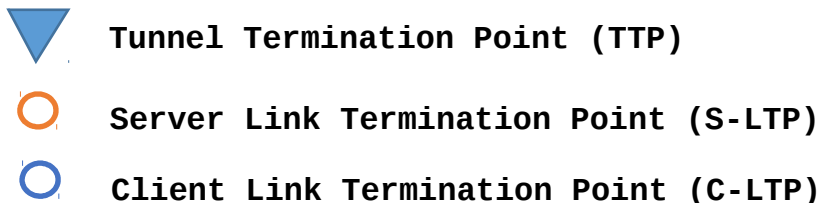
- Inter layer lock
  - Describes client-server layer adaptation relationships
- Modeling abstraction
  - Added inter-layer-lock-id to TE Tunnel Termination Point.

```

augment /nw:networks/nw:network/nw:node:
  +--rw te!
    +--rw te-node-id          te-node-id
    +--rw tunnel-termination-point* [tunnel-tp-id]
      +--rw tunnel-tp-id      binary
      +--rw config
        | +--rw inter-layer-lock-id?      uint32
        | +--rw termination-capability* [link-tp]
        |   +--rw link-tp      leafref
      +--ro state
        +--ro inter-layer-lock-id?      uint32
        +--ro termination-capability* [link-tp]
        | +--ro link-tp      leafref

augment /nw:networks/nw:network/nw:node/nt:termination-point:
  +--rw te!
    +--rw te-tp-id      te-tp-id
    +--rw config
      | +--rw schedules
      | | +--rw schedule* [schedule-id]
      | +--rw interface-switching-capability* [switching-capability]
      | +--rw inter-layer-lock-id      uint32
    +--ro state
      +--ro schedules
      | +--ro schedule* [schedule-id]
      +--ro inter-layer-lock-id?      uint32
  
```

# Inter-layer Lock



- Describes client-server layer adaptation relationship.
- It is an association of M client layer LTPs and N server layer TTPs.
- Each association is uniquely identified by an inter-layer lock ID.

TTP1	IL-1	C-LTP-1	IL-1
TTP2	IL-2, IL-3	C-LTP-2	IL-1
TTP3	IL-2	C-LTP-3	IL-1
		C-LTP-4	IL-2
		C-LTP-5	IL-2
		C-LTP-6	IL-3

# Added Support for Protection on Tunnel Termination Point

- Tunnel Termination Point to Support Protection
  - Added attribute `protection-type` to specify the type of protection that the TTP is capable of, as specified in RFC4872.

```

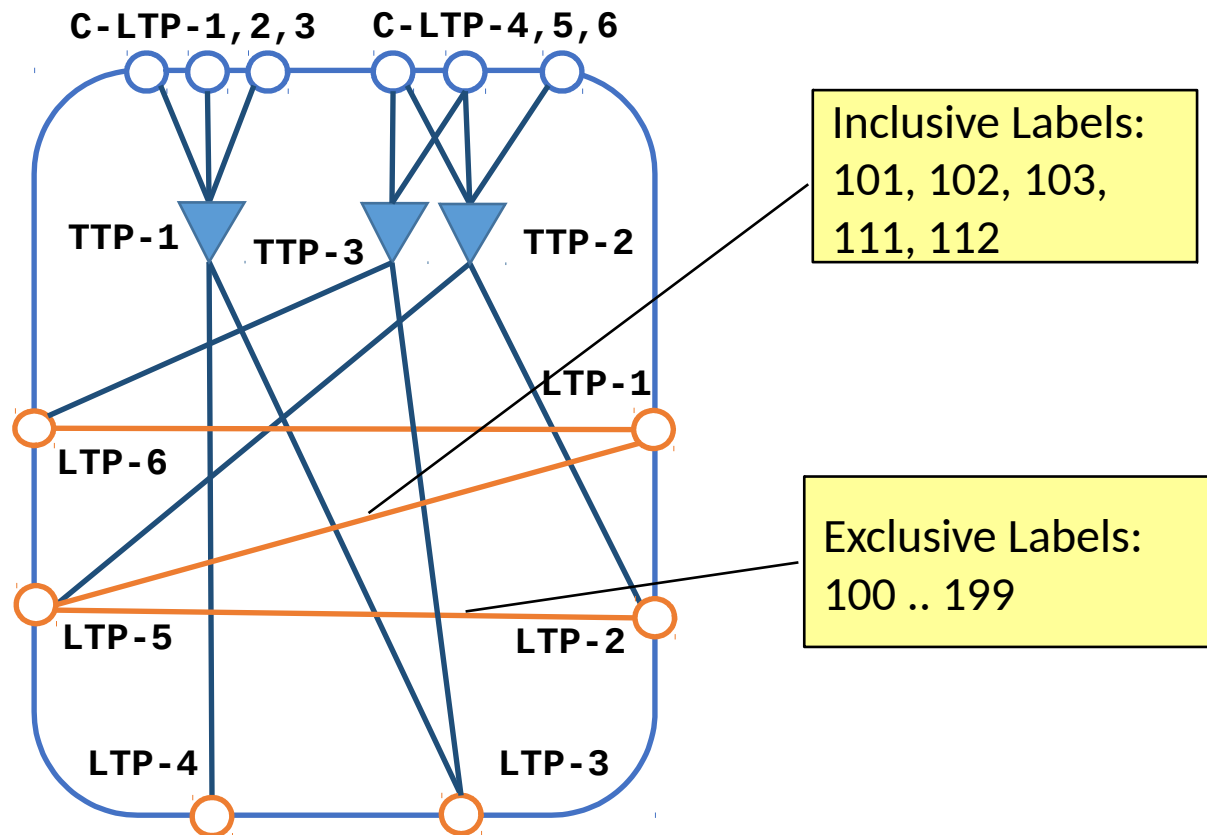
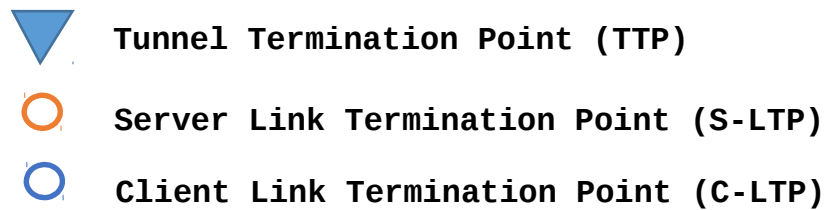
augment /nw:networks/nw:network/nw:node:
  +--rw te!
    +--rw tunnel-termination-point* [tunnel-tp-id]
      +--rw tunnel-tp-id    binary
      +--rw config
        | +--rw protection-type?           identityref
      +--ro state
        +--ro protection-type?           identityref
  
```

## protection-type value range:

- `lsp-prot-unprotected`
- `lsp-prot-reroute-extra`
- `lsp-prot-reroute`
- `lsp-prot-1-for-n`
- `lsp-prot-unidir-1-to-1`
- `lsp-prot-bidir-1-to-1`



# Added Support for Label Restrictions on Connectivity Matrix



# Added Support for Label Restrictions on Connectivity Matrix

- RFC7579 specifies label constraints on connectivity matrix.
  - Inclusion and exclusion.
  - Simple label, label range, and label set.

```

augment /nw:networks/nw:network/nw:node:
  +--rw te!
    +--rw config
      | +--rw te-node-attributes
      |   +--rw connectivity-matrix* [id]
      |     +--rw id                               uint32
      |     +--rw label-restriction* [inclusive-exclusive label-start]
      |       | +--rw inclusive-exclusive         enumeration
      |       | +--rw label-start                 te-types:generalized-label
      |       | +--rw label-end?                  te-types:generalized-label
      |       | +--rw range-bitmap?               binary
    +--ro state
      | +--ro te-node-attributes
      |   +--ro connectivity-matrix* [id]
      |     +--ro id                               uint32
      |     +--ro label-restriction* [inclusive-exclusive label-start]
      |       | +--ro inclusive-exclusive         enumeration
      |       | +--ro label-start                 te-types:generalized-label
      |       | +--ro label-end?                  te-types:generalized-label
      |       | +--ro range-bitmap?               binary

```

# Added Support for Topology Optimization Options

- The optimization objectives may be different for different situations, and may change over time [RFC3272].
- Added attribute `optimization-criterion` to specify the optimization options.

```
augment /nw:networks/nw:network:  
  +--rw te!  
    +--rw provider-id      te-types:te-global-id  
    +--rw client-id       te-types:te-global-id  
    +--rw te-topology-id  te-types:te-topology-id  
    +--rw config  
      | +--rw optimization-criterion?  identityref  
    +--ro state  
      +--ro optimization-criterion?  Identityref
```

**optimization-criterion value range:**

- **not-optimized**
- **cost**
- **delay**

# Enhanced Support for Inter-domain Topology

Domain TE topologies  
renamed and merged into client's native TE topology

C-11

C-21

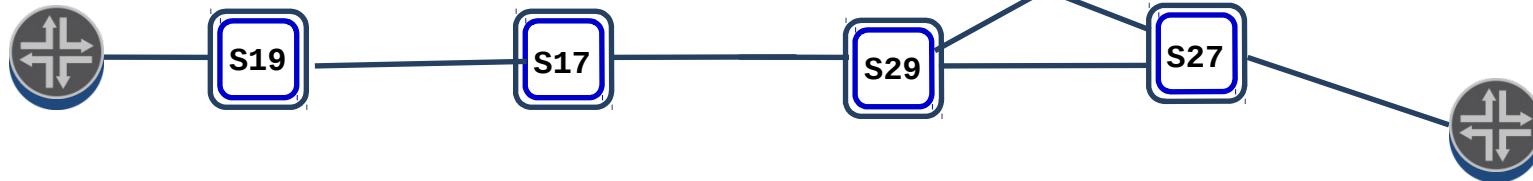


C-12

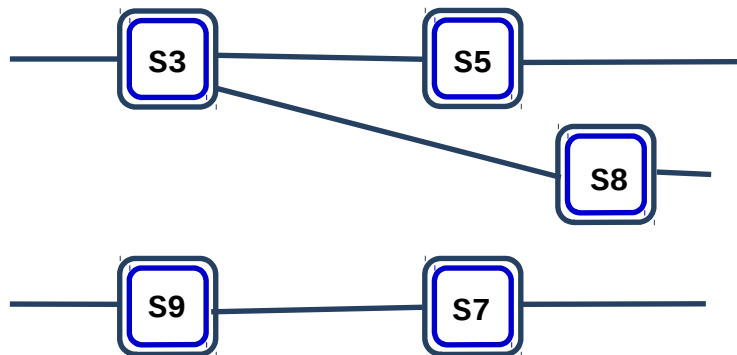
C-22



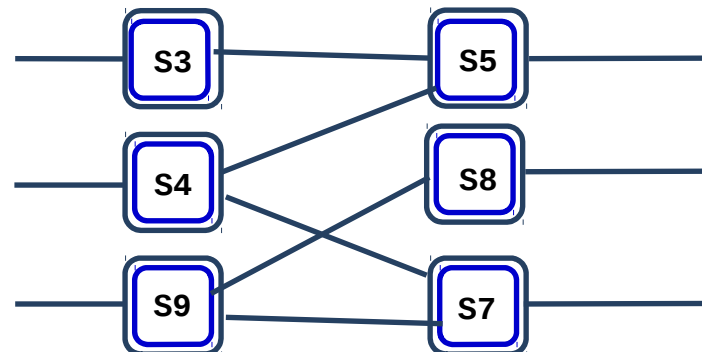
C-23



Domain 1 TE topology

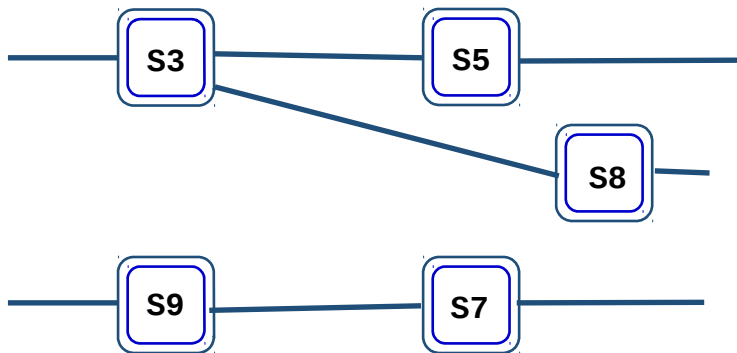


Domain 2 TE topology

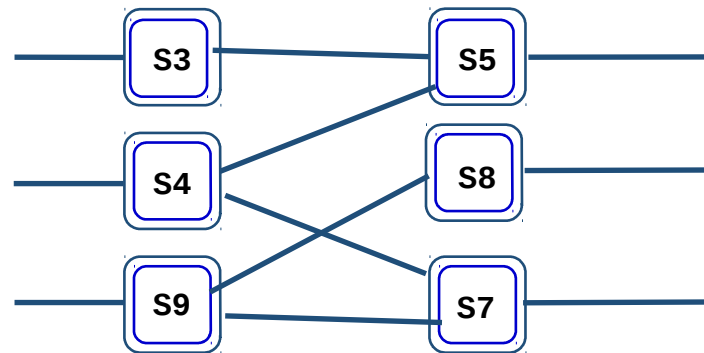


# Topology Abstractions for Inter-domain Topology

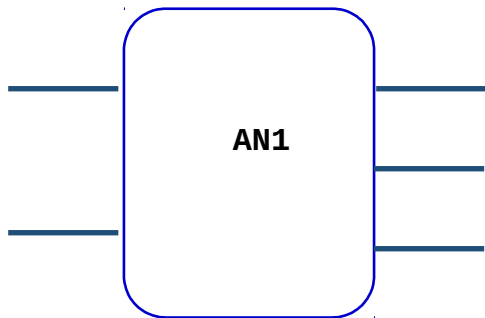
Domain 1 abstract TE topology 1



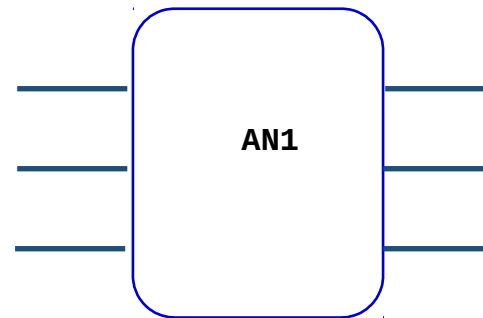
Domain 2 abstract TE topology 1



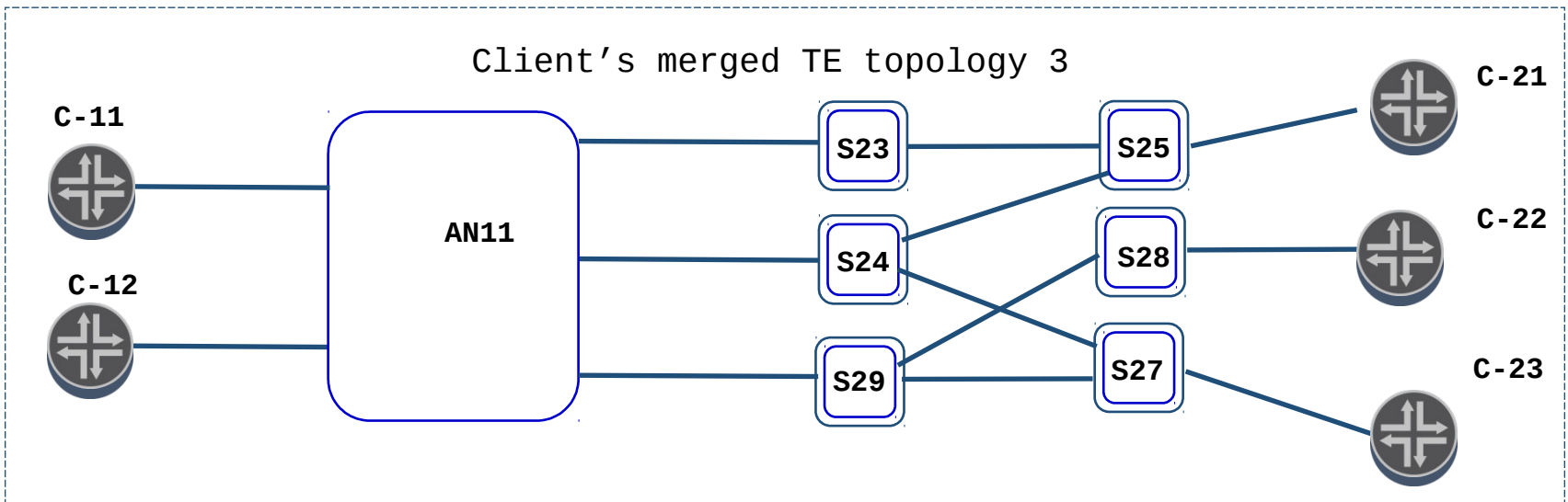
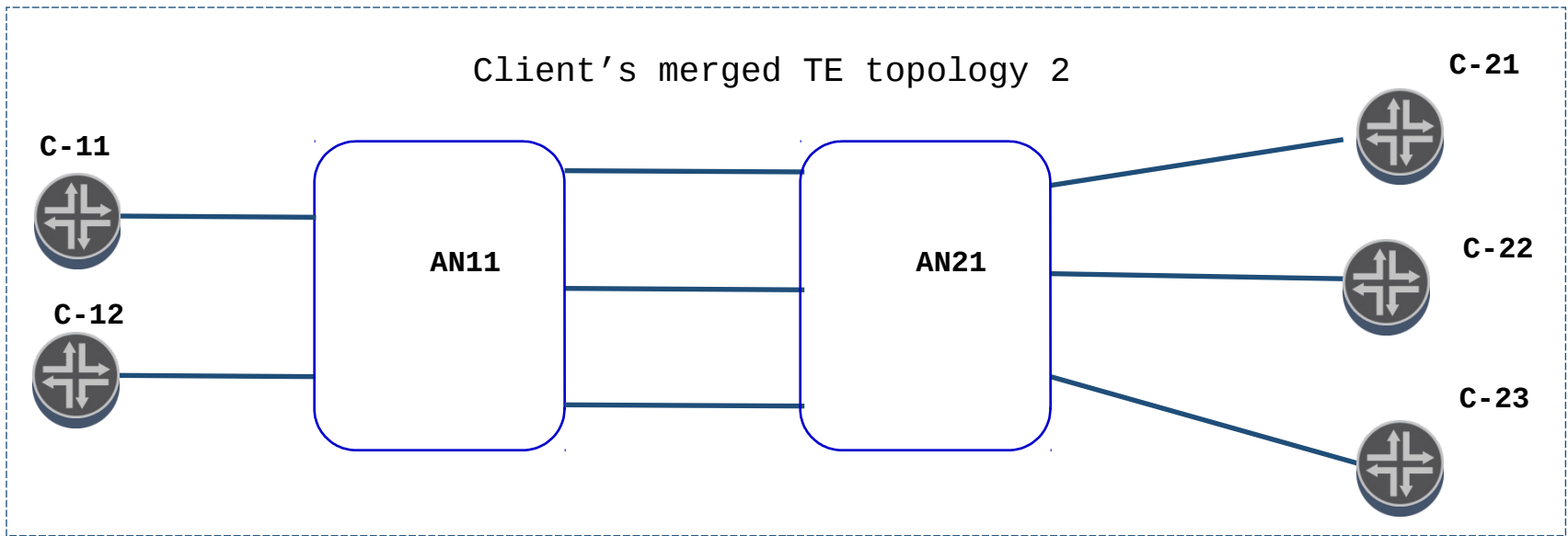
Domain 1 abstract TE topology 2



Domain 2 abstract TE topology 2



# Multiple Merged Native TE Topologies



# Modeling Support for Inter-domain Topology

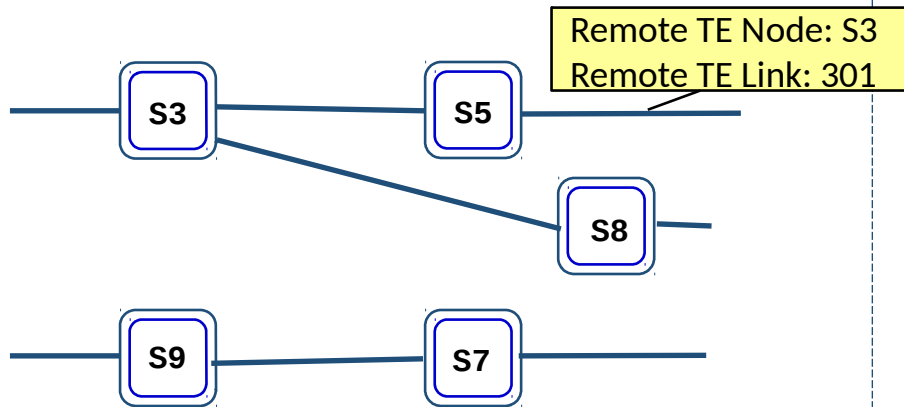
- Specify remote TE node ID and TE link ID.
- Specify inter-domain plug ID.

```

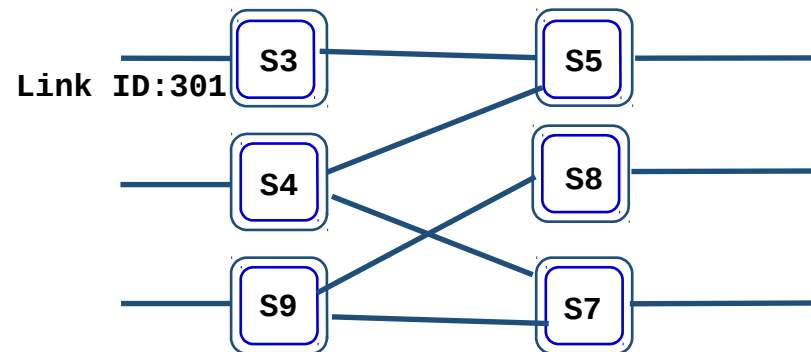
augment /nw:networks/nw:network/nt:link:
  +--rw te!
    +--rw config
      | +--rw te-link-attributes
      |   +--rw schedules
      |     | +--rw schedule* [schedule-id]
      |   +--rw external-domain
      |     | +--rw remote-te-node-id?      te-types:te-node-id
      |     | +--rw remote-te-link-tp-id?   te-types:te-tp-id
      |     | +--rw plug-id?                uint32
      |   +--rw is-abstract?                empty
    +--ro state
      +--ro te-link-attributes
      | +--ro schedules
      |   | +--ro schedule* [schedule-id]
      | +--ro external-domain
      |   | +--ro remote-te-node-id?      te-types:te-node-id
      |   | +--ro remote-te-link-tp-id?   te-types:te-tp-id
      |   | +--ro plug-id?                uint32
      | +--ro is-abstract?                empty
  
```

# Topology Abstractions for Inter-domain Topology

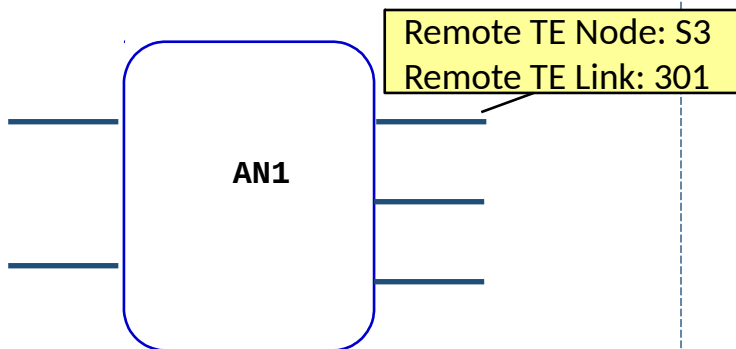
Domain 1 abstract TE topology 1



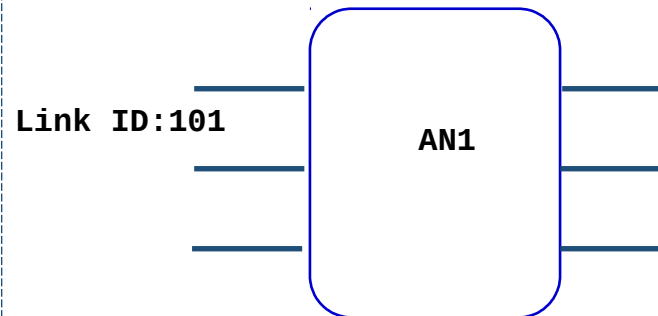
Domain 2 abstract TE topology 1



Domain 1 abstract TE topology 2



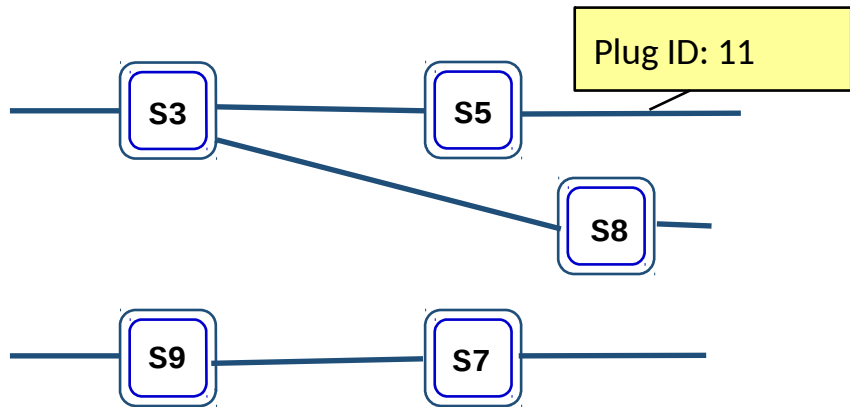
Domain 2 abstract TE topology 2



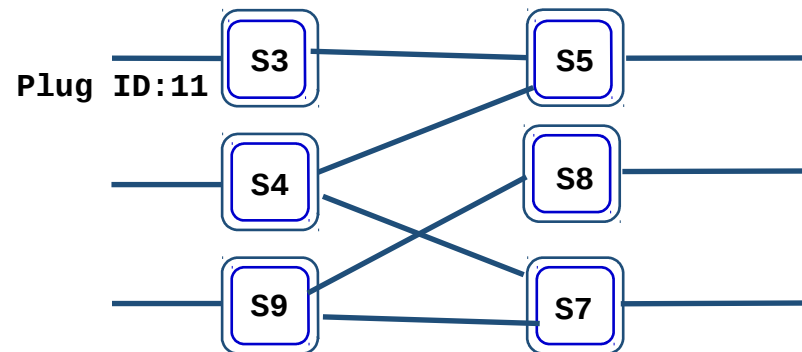


# Topology Abstractions for Inter-domain Topology

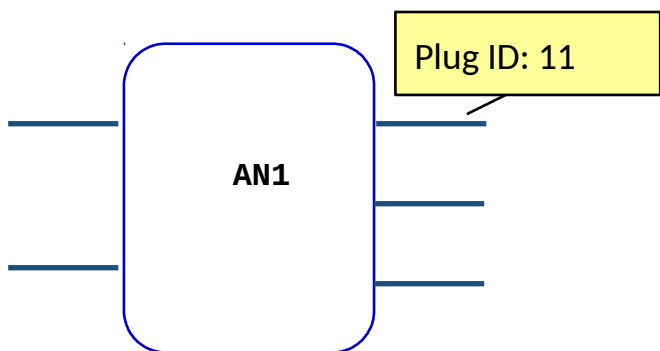
Domain 1 abstract TE topology 1



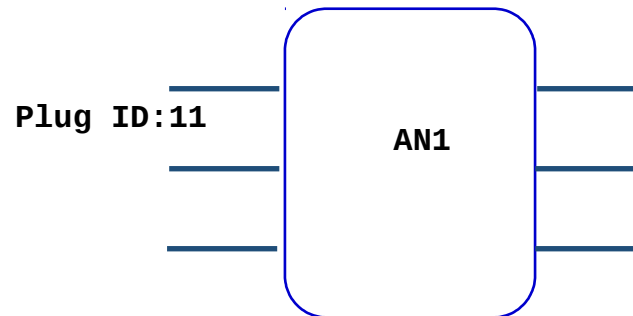
Domain 2 abstract TE topology 1



Domain 1 abstract TE topology 2



Domain 2 abstract TE topology 2



# ID Type Debate

- It is being debated on the YANG data types for the IDs. Need WG consensus.
  1. The type for plug ID:
    - uint32
    - URI
    - String
  2. More generally, the types for other IDs, including TE Node ID and TE Link ID:
    - More specific types: dotted-quad, uint32
    - URI

# ID Type Debate

- Arguments for number type (uint32 or dotted-quad):
  - Compatible to current implementations and RFCs.
  - More efficient to implement (e.g. searching, sorting, and indexing).
  - Easier to advertise.
  - Easier to do automation (e.g. getting the next available value).
  - The model is mostly for machine-to-machine interface. Client software can do the mapping if user friendly formats are needed.
  - No need for parsing and conversion, and no ambiguity for interpretation.
- Arguments for URI:
  - Forward thinking.
  - More human readable.
  - Flexible for conversion to other formats.

# Next Steps

- Address review comments.
  - To Do List:
    - <https://github.com/ietf-mpls-yang/te/blob/master/ietf-te-topology-todo-list.txt>
- Request further review.

# Yang Data Model for Layer 3 TE Topologies

draft-liu-teas-yang-l3-te-topo-01

Xufeng Liu (Ericsson)

Vishnu Pavan Beeram (Juniper Networks)

Igor Bryskin (Huawei Technologies)

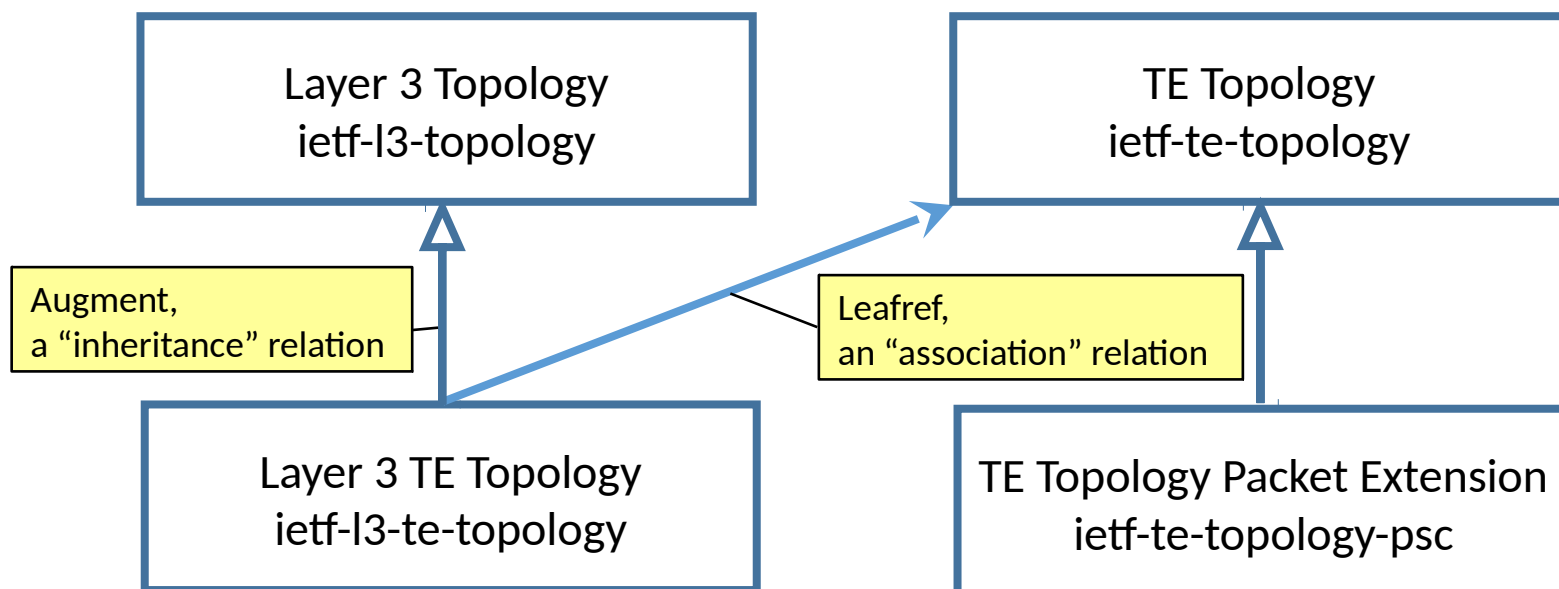
Tarek Saad (Cisco)

Himanshu Shah (Ciena)

Oscar Gonzalez De Dios (Telefonica)

# Augmentation Hierarchy

- L3 TE Topology augments L3 Topology and references TE Topology.
- Packet extension module augments ietf-te-topology.
  - Asking feedback on the module name. Options can be:
    - ietf-te-topology-pcs
    - ietf-te-topology-packet
    - ietf-te-topology-mpls



# Augmenting L3 Network Topology Model

- Add references to TE topology instances.

```
module: ietf-l3-te-topology
augment /nw:networks/nw:network/nw:network-types/l3t:l3-unicast-igp-
topology:
  +--rw l3-te!
augment /nw:networks/nw:network/l3t:igp-topology-attributes:
  +--rw l3-te-topology-attributes
    +--rw network-ref? leafref
augment /nw:networks/nw:network/nw:node/l3t:igp-node-attributes:
  +--rw l3-te-node-attributes
    +--rw node-ref? leafref
    +--rw network-ref? leafref
augment /nw:networks/nw:network/nw:node/nt:termination-point/l3t:igp-
termination-point-attributes:
  +--rw l3-te-tp-attributes
    +--rw tp-ref? leafref
    +--rw node-ref? leafref
    +--rw network-ref? leafref
augment /nw:networks/nw:network/nt:link/l3t:igp-link-attributes:
  +--rw l3-te-link-attributes
    +--rw link-ref? leafref
    +--rw network-ref? leafref
```

# Augmenting Base TE Topology Model

- Add attributes that are specific to packet switching technology.

```

module: ietf-te-topology-psc
augment /nw:networks/nw:network/nt:link/tet:te/tet:config/tet:te-link-attributes/tet:interface-switching-capability:
  +--rw packet-switch-capable
    +--rw minimum-lsp-bandwidth?   decimal64
    +--rw interface-mtu?           Uint16

augment /nw:networks/nw:network/nt:link/tet:te/tet:state/tet:te-link-attributes/tet:interface-switching-capability:
  +--ro packet-switch-capable
    +--ro minimum-lsp-bandwidth?   decimal64
    +--ro interface-mtu?           Uint16

augment /nw:networks/nw:network/nt:link/tet:te/tet:state/tet:information-source-entry/tet:interface-switching-capability:
  +--ro packet-switch-capable
    +--ro minimum-lsp-bandwidth?   decimal64
    +--ro interface-mtu?           Uint16

augment /tet:te-link-event/tet:te-link-attributes/tet:interface-switching-capability:
  +---- packet-switch-capable
    +---- minimum-lsp-bandwidth?   decimal64
    +---- interface-mtu?           uint16

augment /tet:te-link-event/tet:information-source-entry/tet:interface-switching-capability:
  +---- packet-switch-capable
    +---- minimum-lsp-bandwidth?   decimal64
    +---- interface-mtu?           uint16

```



# Next Steps

- Request further review.
- Ask for WG adoption.

# Yang Data Model for SR and SR TE Topologies

draft-liu-teas-yang-sr-te-topo-00

Xufeng Liu (Ericsson)

Vishnu Pavan Beeram (Juniper Networks)

Igor Bryskin (Huawei Technologies)

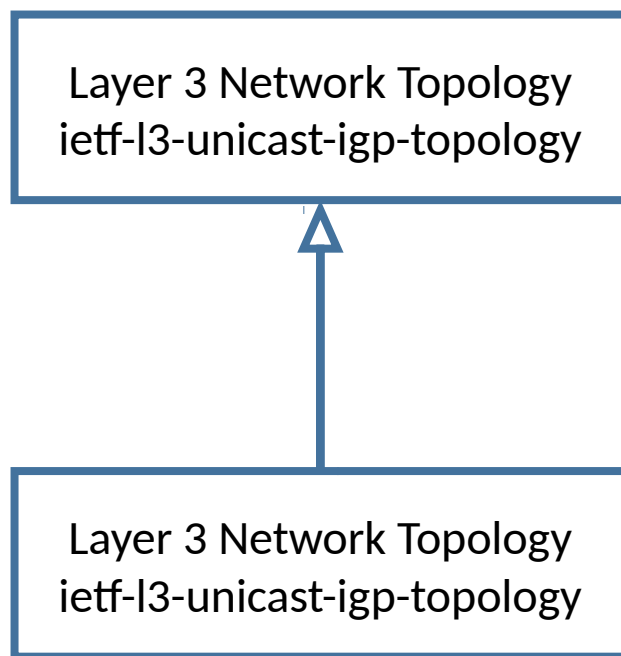
Tarek Saad (Cisco)

Himanshu Shah (Ciena)

Stephane Litkowski (Orange)

# SR (Segment Routing) Topology

- Augment layer 3 network topology model.



# SR Topology Structure

- Define a new network-type “sr”.
- Add network-wide SRGB.

```
module: ietf-l3-unicast-igp-topology
augment /nw:networks/nw:network/nw:network-types:
  +--rw l3-unicast-igp-topology!
    +--rw srt:sr!

augment /nw:networks/nw:network:
  +--rw igp-topology-attributes
    +--rw name?      string
    +--rw flag*     flag-type
    +--rw srt:sr
      +--rw srt:config
        +--rw srt:srgb* [lower-bound upper-bound]
          +--rw srt:lower-bound  uint32
          +--rw srt:upper-bound  uint32
```

# SR Topology Structure

- Augment node to add per-node attributes.

```
augment /nw:networks/nw:network/nw:node:
  +--rw igp-node-attributes
    +--rw name?          inet:domain-name
    +--rw flag*         flag-type
    +--rw router-id*    inet:ip-address
    +--rw prefix* [prefix]
      | +--rw prefix    inet:ip-prefix
      | +--rw metric?   uint32
      | +--rw flag*     flag-type
      | +--rw srt:sr!
      |   +--rw srt:value-type?      enumeration
      |   +--rw srt:start-sid        uint32
      |   +--rw srt:range?           uint32
      |   +--rw srt:last-hop-behavior? enumeration {sid-last-hop-behavior}?
      |   +--rw srt:is-local?        boolean
      |   +--rw srt:algorithm?       sr-algorithm
    +--rw srt:sr
      +--rw srt:config
        | +--rw srt:srgb* [lower-bound upper-bound]
        | | +--rw srt:lower-bound   uint32
        | | +--rw srt:upper-bound   uint32
        | +--rw srt:capabilities
        |   +--rw srt:transport-planes* [transport-plane]
        |   | +--rw srt:transport-plane identityref
        |   +--rw srt:segment-stack-push-limit?   uint8
        |   +--rw srt:readable-label-stack-depth? uint8
      +--ro srt:state
        +--ro srt:information-source?      enumeration
        +--ro srt:information-source-state
        | +--ro srt:credibility-preference?  uint16
        +--ro srt:capabilities
          +--ro srt:transport-planes* [transport-plane]
          | +--ro srt:transport-plane identityref
          +--ro srt:segment-stack-push-limit?   uint8
          +--ro srt:readable-label-stack-depth? uint8
```

# SR Topology Structure

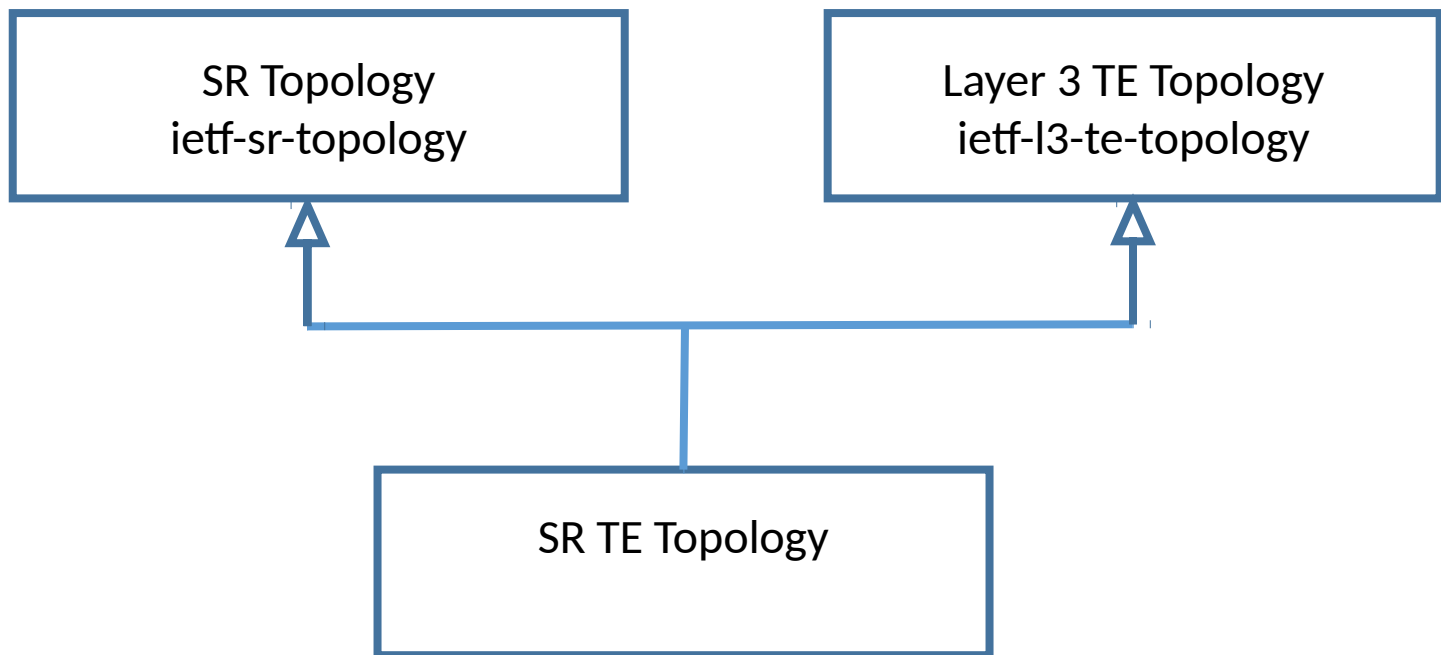
- Augment link to add per-link attributes.

```

augment /nw:networks/nw:network/nt:link:
  +--rw igp-link-attributes
    +--rw name?      string
    +--rw flag*     flag-type
    +--rw metric?   uint32
  +--rw srt:sr
    +--rw srt:config
      | +--rw srt:sid?          uint32
      | +--rw srt:value-type?  enumeration
      | +--rw srt:is-local?    boolean
      | +--ro srt:is-part-of-set? boolean
      | +--ro srt:is-on-lan?   boolean
    +--ro srt:state
      +--ro srt:information-source?      enumeration
      +--ro srt:information-source-state
        +--ro srt:credibility-preference?  uint16
  
```

# SR (Segment Routing) TE Topology

- Multiple inheritance:
  - Is both SR topology and layer 3 TE topology model.
  - Uses multiple network types: “l3-te” and “sr”.



# Next Steps

- Work with authors of draft-ietf-spring-sr-yang to share YANG types.
- Add support for protection on link.
- Add support for link bundle.