YANG Data Models for TE and RSVP

draft-ietf-teas-yang-te-04

https://github.com/ietf-mpls-yang/te

Tarek Saad (Presenter) and Rakesh Gandhi, Cisco Systems
Vishnu Pavan Beeram, Juniper Networks
Xufeng Liu, Ericsson
Himanshu Shah, Ciena
Igor Bryskin, Huawei
Xia Chen, Huawei
Raqib Jones, Brocade
Bin Wen, Comcast

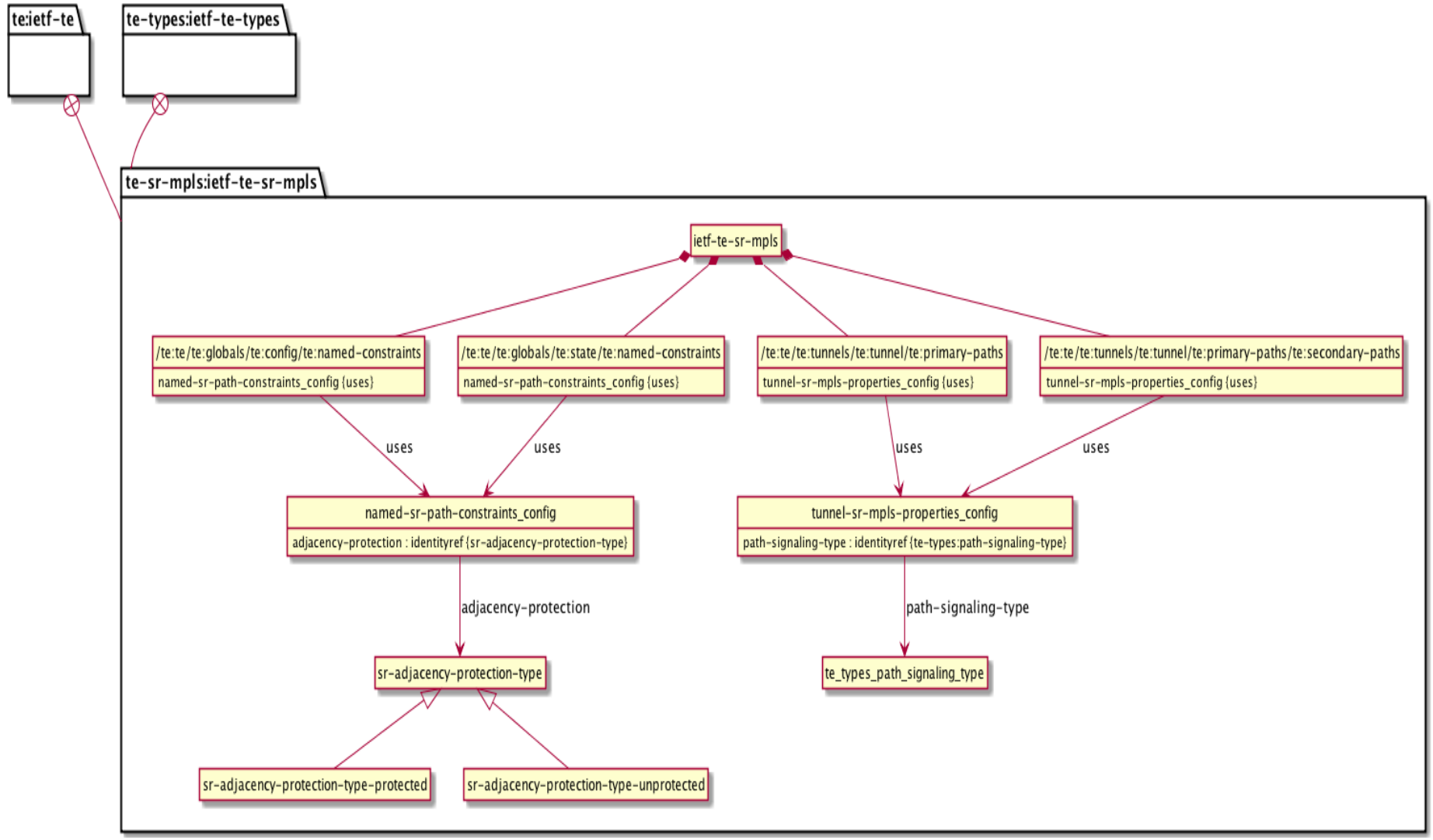
Agenda

- Updates (from previous version)
- Open issues
- Next steps

Since IETF95 Update # 1
MPLS Segment Routing Traffic Engineering

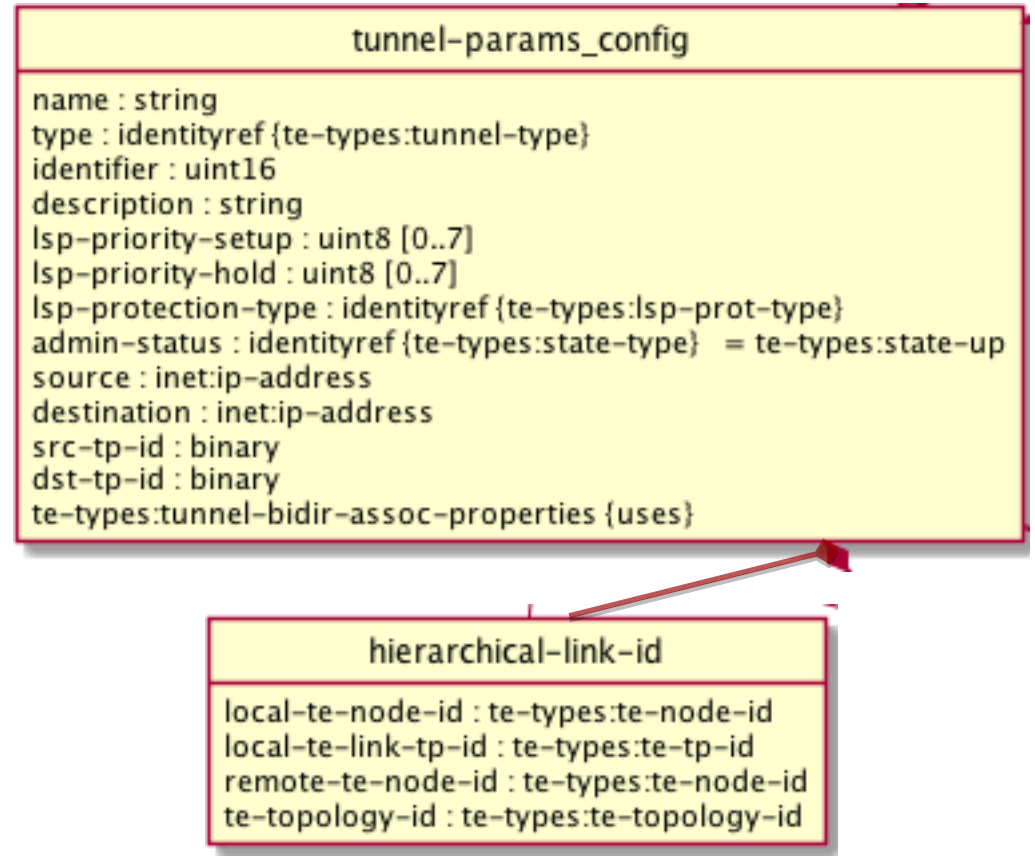
New module that

- Augments the TE module
- Allows per path signaling type as “SR”
- Allows specifying the TE path as set of MPLS label(s)
- Additional path constraints, including protected/unprotected segments



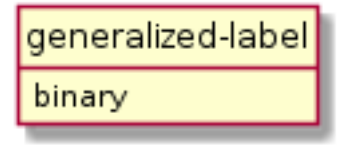
Since IETF95 Update # 2
TE link endpoint for H-LSP

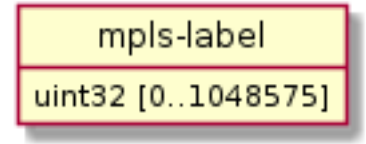
- Support for TE link endpoint for hierarchical LSP
  - Associates the underlay tunnel with the overlay TE link
  - Allows propagating properties of the hierarchical LSP to the overlay TE link

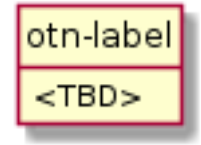


Since IETF95 Update # 3
Generic TE label type

- Need for technology agnostic label
  - Series of bytes with no strict type check
- Technology specific module(s) use technology specific label type
  - Packet MPLS LSP use labels of type mpls-label, OTN LSPs of type otn-label
- The abstract TE model(s) (e.g. abstract TE topology) that may contain data nodes belonging to multiple technologies use TE generic label type
  - Run time check using “must” (e.g. on expected value)







.
.
.

Since IETF95 Update # 4
TE transport Model

- Continious sync-ups with the team driving TE transport service model on possible alignments
- Received some comments from team driving Transport service model on possible changes/alignments
- Topics under discussion
  - Use of strings (or uints) versus URIs for IDs or keys in the model
  - RPCs for creation/deletion of tunnels

Issue# 1

Reuse of TE model for different technologies

- **<u>OPTION # 1:</u>** Reusable data as groupings (tunnels, LSPs, etc.) in generic module(s)
  - Technology models (e.g. MPLS, OTN, etc.) import the generic module and use the generic groupings
- PROs:
  - Allows reuse across technologies
- CONs:
  - Augment of generic groupings **<u>only</u>** possible after grouping is used (in each tech model)
  - Referencing nodes in one generic grouping from another is error prone
    - Relative path in leafrefs can easily break if groupings are arbitrarily used in the tree

```
module ietf-te-mpls {
  uses te-generic-grouping {
    augment "te" {
      uses te-mpls-grouping
    }
  }
  etc.
}
```

```
module ietf-te-otn {
  uses te-generic-grouping {
    augment "te" {
      uses te-otn-grouping
    }
  }
  etc.
}
```

Issue# 1

Reuse of TE model for different technologies

- **<u>OPTION # 2:</u>** TE generic model defined as standalone at the root/TOP of tree
  - Technology models reuse the generic model by “mounting” it under respective path (../mpls/te/…, ../otn/te/…, etc)
- PROs:
  - Augmentation of generic model possible
  - Separation of technology specific TE data
- CONs:
  - Limitation on referencing data nodes outside mount:
    - e.g. needed to reference interfaces for TE device model

Issue# 1

Reuse of TE model for different technologies

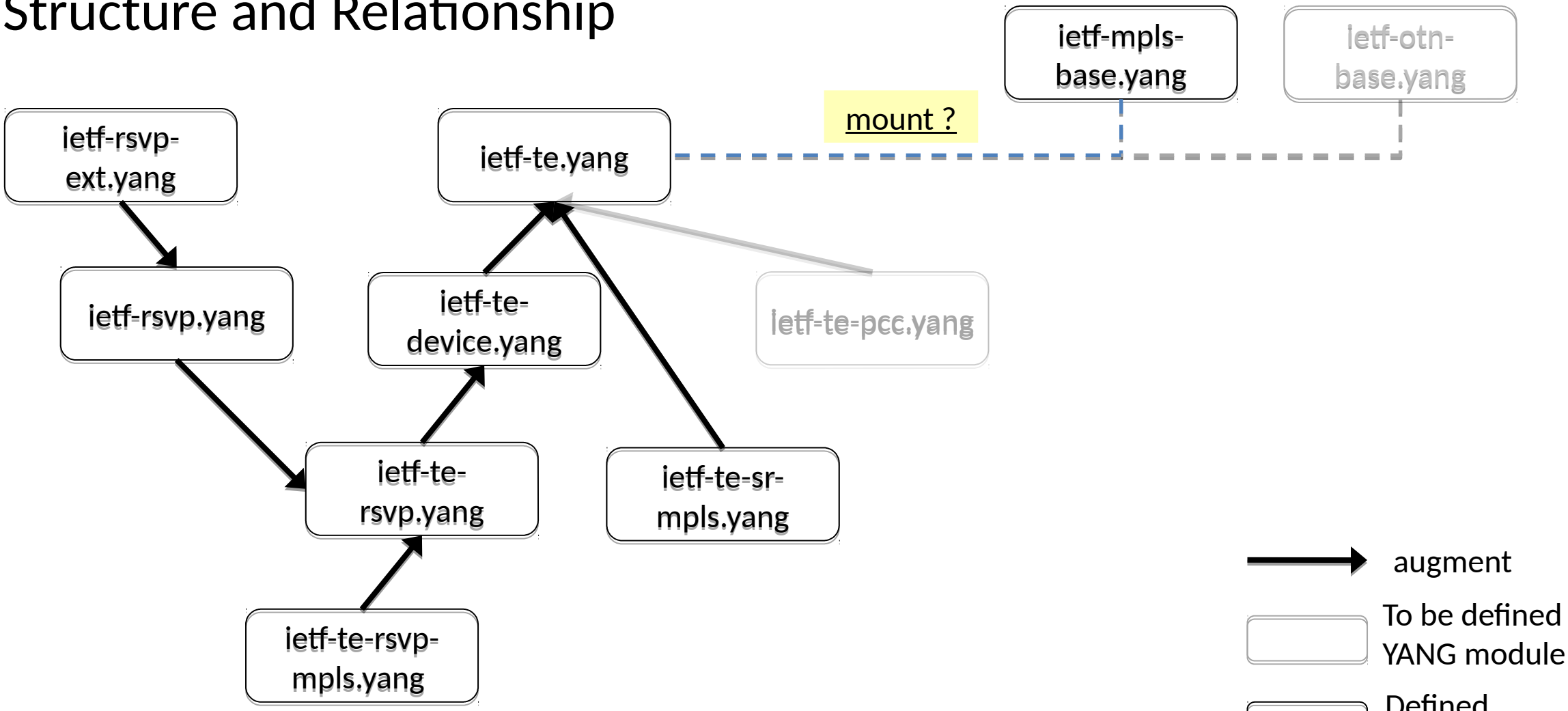
- **<u>OPTION # 3:</u>** TE generic model defined as standalone at the root/TOP of tree with per data node technology type
  - Different TE technology data nodes are represented in same model, e.g.:
    - MPLS, OTN, and other LSPs coexist in same model (in same list too)
    - A new technology type per data node (e.g. LSP) to identify specific tech.
- PROs:
  - Augmentation of generic model possible
- CONs:
  - No separation of technology specific TE data

Summary Issue #1

Option	Reusability of Generic Data	Dependency between data nodes (leafref)	Separation of technology specific data	Augment of generic model
OPTION #1 (groupings)	✝	✝	✝	Only when grouping is used
OPTION #2 (mount)	✝	Not outside mount space	✝	✝
OPTION #3 (multi-technology generic model)	✝	✝	✝	✝

TE/RSVP and MPLS YANG Modules

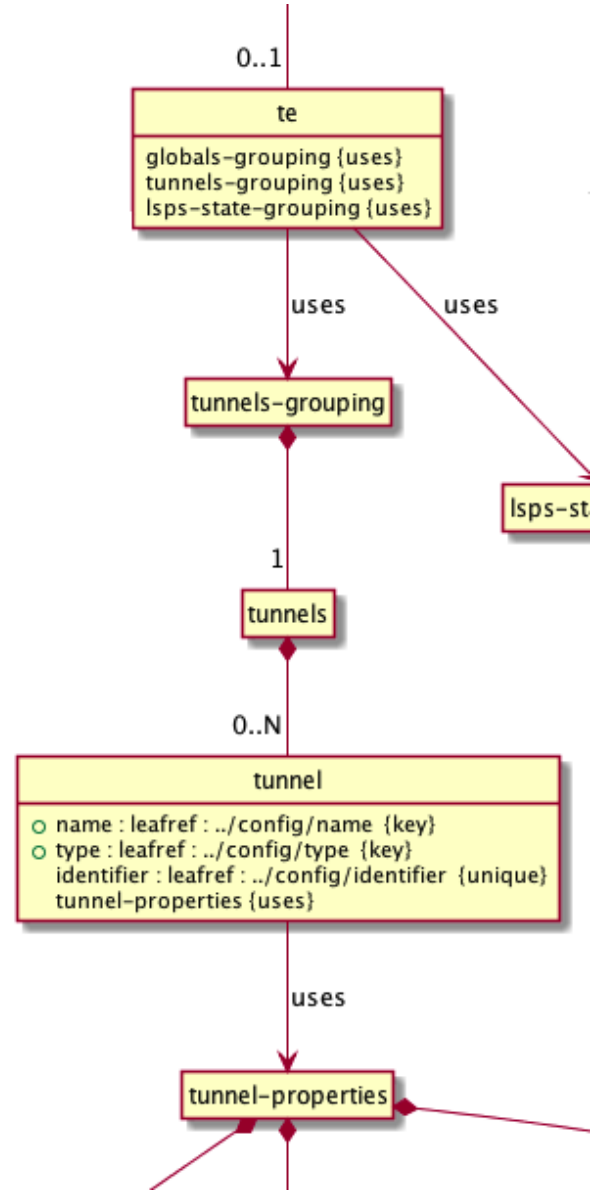
Structure and Relationship



Issue# 2

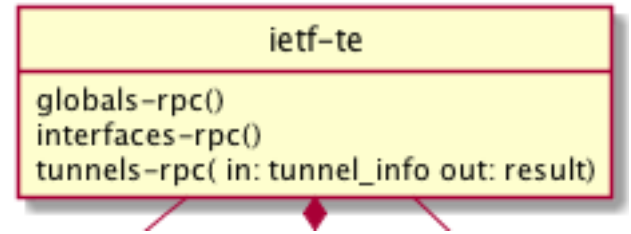
P2MP and P2P TE tunnels lists

- Currently have single list of tunnels and LSPs:
  - Keyed by name and type: P2P or P2MP
  - List contains both P2P and P2MP types
- P2MP tunnels/LSPs have multiplicity of destinations
  - May be simpler to model P2MP separate from P2P LSPs
  - Existing P2P/P2MP MIBs have this separation



Open Issue# 3
RPCs for TE tunnels

- Model (so far) allows TE tunnels creation via:
  - Configuration
  - PCE/controller as ephemeral
- Another option is to allow creation/deletion of TE tunnels via RPC:
  - Also creates ephemeral state
  - User specified tunnel attributes or associated with attribute set
  - Generic question and may apply to apply to many other model that support ephemeral state



Next Steps

- Conclude on open issues
- Request further review and address comments
- Complete the augmentation for module:
  1. PCC-TE data

Thank You