YANG Data Models for TE
<draft-ietf-teas-yang-te-16>

Latest YANG code @ https://github.com/ietf-mpls-yang/te

Tarek Saad (Presenter) and Rakesh Gandhi, Cisco Systems

Vishnu Pavan Beeram, Juniper Networks

Xufeng Liu, Volta Networks

Igor Bryskin, Huawei

Himanshu Shah, Ciena

IETF-102,July 2018, Montreal

Summary of Changes

- Added RFC references to modelled objects

- Updates to the TE tunnel model

Update # 1
LSP Association Types

<draft-ietf-teas-yang-te-16>

```
identity association-type {
  description "Base identity for tunnel association";
  reference "RFC6780, RFC4872, RFC4873";
}
identity association-type-recovery {
  base association-type;
  description
    "Association Type Recovery used to association LSPs of
     same tunnel for recovery";
  reference "RFC4872";
}
identity association-type-resource-sharing {
  base association-type;
  description
    "Association Type Resource Sharing used to enable resource
     sharing during make-before-break.";
  reference "RFC4873";
}
identity association-type-double-sided-bidir {
  base association-type;
  description
    "Association Type Double Sided bidirectional used to associate
     two LSPs of two tunnels that are independently configured on
     either endpoint";
  reference "RFC7551";
}
identity association-type-single-sided-bidir {
  base association-type;
  description
    "Association Type Single Sided bidirectional used to associate
     two LSPs of two tunnels, where a tunnel is configured on one
     side/endpoint, and the other tunnel is dynamically created on
     the other endpoint";
  reference "RFC7551";
}
```

- Added Identities for known LSP ASSOCIATION types in ietf-te-types.yang module
- Additional types can be added if/when needed

3
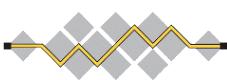
Update # 2
Association Objects

<draft-ietf-teas-yang-te-16>

```
+--rw tunnels
|  +--rw tunnel* [name]
|  |  +--rw name                              string
|  |  +--rw identifier?                       uint16
p_te_model.tree
|     +--rw association-objects
|     |  |  +--rw association-object* [type ID source global-source]
|     |  |  |  +--rw type           identityref
|     |  |  |  +--rw ID             uint16
|     |  |  |  +--rw source         inet:ip-address
|     |  |  |  +--rw global-source  inet:ip-address
|     |  |  +--rw association-object-extended* [type ID source global-source extended-ID]
|     |  |  |  +--rw type           identityref
|     |  |  |  +--rw ID             uint16
|     |  |  |  +--rw source         inet:ip-address
|     |  |  |  +--rw global-source  inet:ip-address
|     |  |  |  +--rw extended-ID    binary
```

```
container association-objects {
  description "TE tunnel associations";
  list association-object {
    key "type ID source global-source";
    description "List of association base objects";
    reference "RFC4872";
    leaf type {
      type identityref {
        base te-types:association-type;
      }
      description "Association type";
      reference "RFC4872";
    }
    leaf ID {
      type uint16;
      description "Association ID";
      reference "RFC4872";
    }
    leaf source {
      type inet:ip-address;
      description "Association source";
      reference "RFC4872";
    }
    leaf global-source {
      type inet:ip-address;
      description "Association global source";
      reference "RFC4872";
    }
  }
}
```

```
list association-object-extended {
  key "type ID source global-source extended-ID";
  description "List of extended association objects";
  reference "RFC6780";
  leaf type {
    type identityref {
      base te-types:association-type;
    }
    description "Association type";
  }
  leaf ID {
    type uint16;
    description "Association ID";
    reference "RFC4872";
  }
  leaf source {
    type inet:ip-address;
    description "Association source";
  }
  leaf global-source {
    type inet:ip-address;
    description "Association global source";
    reference "RFC4872";
  }
  leaf extended-ID {
    type binary;
    description "Association extended ID";
    reference "RFC4872";
  }
}
```

- Added association objects under tunnel:
  - 2 Lists of ASSOCIATION and Extended ASSOCIATION objects, keyed by
    - Type
    - ID
    - Source
    - Global-source
    - Extended ID (for Extended AO only)
- For example:
  - can use it to associate 2 unidirectional LSPs to make a bidirectional tunnel
  - can use as path computation constraint:
    - Associate LSP resources
    - Diverse LSP paths for tunnels (possibly originating at different ingresses)

Update # 3
Combined numbered/unnumbered hop in Explicit Route

<draft-ietf-teas-yang-te-16>

```
case num-unnum-hop {
  container num-unnum-hop {
    leaf node-id {
      type te-types:te-node-id;
      description
        "The identifier of a node in the TE topology.";
    }
    leaf link-tp-id {
      type te-types:te-tp-id;
        description
          "TE link termination point identifier. The combination
           of TE link ID and the TE node ID is used to identify an
           unnumbered TE link.";
    }
    leaf hop-type {
      type te-hop-type;
      description "strict or loose hop";
    }
    leaf direction {
      type te-link-direction;
      default INCOMING;
      description "Link ERO direction";
    }
    description
      "Numbered and Unnumbered link/node explicit route
       subobject";
    reference
      "RFC3209: section 4.3 for EXPLICIT_ROUTE in RSVP-TE
       RFC3477: Signalling Unnumbered Links in RSVP-TE";
  }
```

- Unnumbered Link:
  - identified by TE node ID + TE Link ID
- Numbered Link:
  - Identified by TE Link ID
- Numbered node:
  - TE node ID

```
+--rw explicit-route-include-objects
   +--rw route-object-include-object* [index]
      +--rw index                 uint32
      +--rw (type)?
         +--:(num-unnum-hop)
         |  +--rw num-unnum-hop
         |     +--rw node-id?      te-types:te-node-id
         |     +--rw link-tp-id?   te-types:te-tp-id
         |     +--rw hop-type?     te-hop-type
         |     +--rw direction?    te-link-direction
```

Update # 4
SRLG in Exclude Route List

<draft-ietf-teas-yang-te-16>

```
list route-object-exclude-object {
  key index;
  description
    "List of explicit route objects to be excluded
     in path computation";
  uses te-types:explicit-route-hop {
    augment "type" {
      case srlg {
        container srlg {
          description "SRLG container";
          leaf srlg {
            type uint32;
            description "SRLG value";
          }
        }
        description "An SRLG value to be included or excluded";
      }
      description
        "Augmentation to generic explicit route for SRLG exclusion";
    }
  }
}
```

```
+--rw explicit-route-exclude-objects
|  +--rw route-object-exclude-object* [index]
|     +--rw index                   uint32
|     +--rw (type)?
|        +--:(num-unnum-hop)
|        |  +--rw num-unnum-hop
|        |     +--rw node-id?        te-types:te-node-id
|        |     +--rw link-tp-id?     te-types:te-tp-id

|        +--rw srlg
|           +--rw srlg?    uint32
```

- Added SRLG as entry in the exclude route object

Dependency of other drafts
on TE types module

- TE types are defined ietf-te-types.yang module and included in I-D <draft-ietf-teas-yang-te>

- Other drafts, e.g. I-D <draft-ietf-teas-yang-te-topo> have progressed in IESG review but have dependency on TE types

- Most YANG types modules at IETF are in their own drafts, e.g.
  - RFC8294: Common YANG Data Types for the Routing Area

- Propose to carry TE types in a separate document
  - New I-D: <draft-ietf-teas-yang-te-types> for TE types
  - Next slide for proposed split

TE YANG Modules Documentation Split Proposal
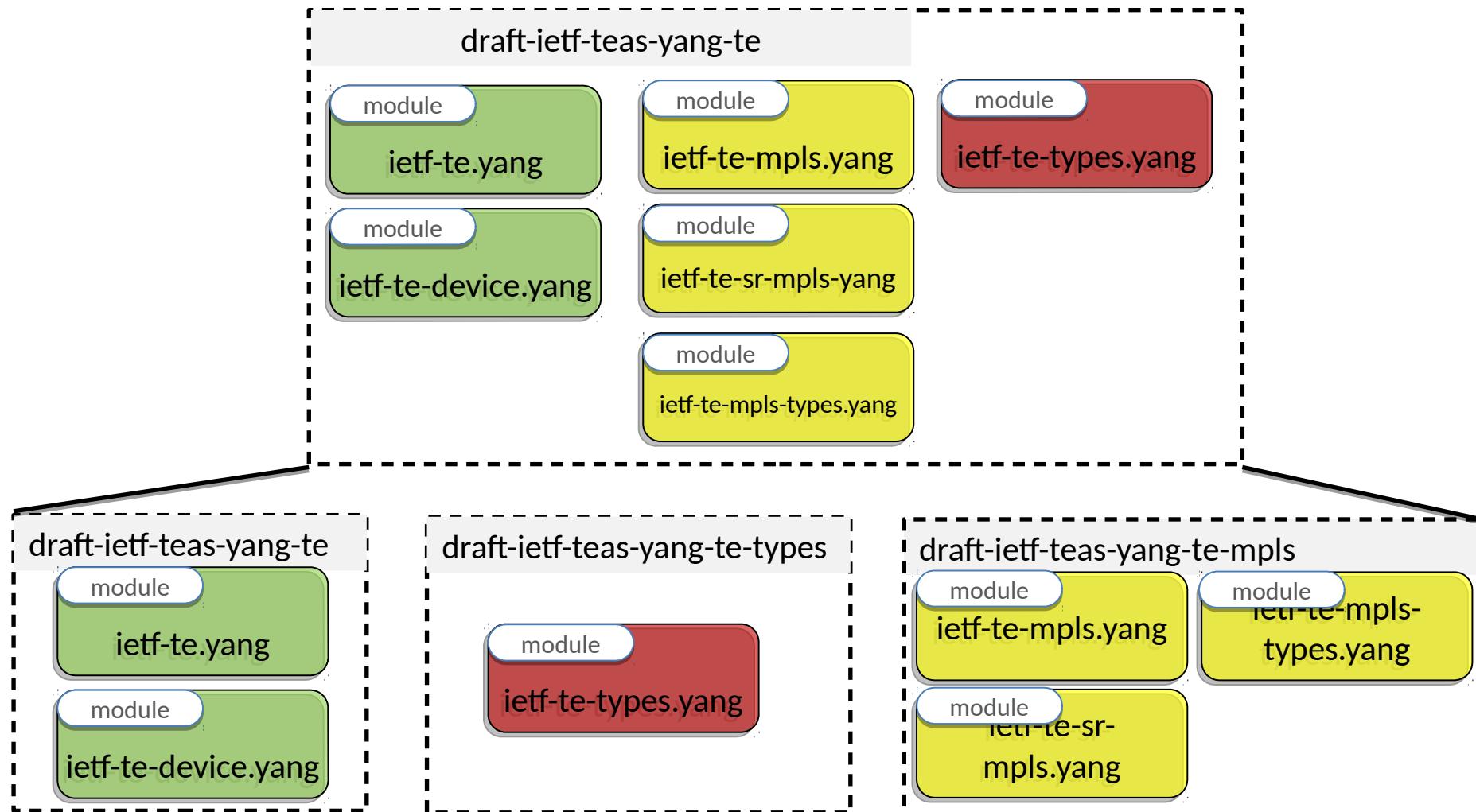
Next Steps

- The following I-Ds will be ready for WGLC after the draft split

  - draft-ietf-teas-yang-te-types

  - draft-ietf-teas-yang-te

- The following I-Ds will undergo a round of review and update

  before asking for LC

  - draft-ietf-teas-yang-te-mpls

Thank You