TEAS Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 6, 2020

T. Saad
Juniper Networks
R. Gandhi
Cisco Systems Inc
X. Liu
Volta Networks
V. Beeram
Juniper Networks
I. Bryskin
Huawei Technologies

July 05, 2019

Traffic Engineering Common YANG Types
draft-ietf-teas-yang-te-types-10

Abstract

This document defines a collection of common data types and groupings in YANG data modeling language. These derived common types and groupings are intended to be imported by modules that model Traffic Engineering (TE) configuration and state capabilities.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 6, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
1.1. Terminology 2
1.2. Prefixes in Data Node Names 3
2. Acronyms and Abbreviations 3
3. Overview 4
3.1. TE Types Module Contents 4
3.2. Packet TE Types Module Contents 8
4. TE Types YANG Module 8
5. Packet TE Types YANG Module 68
6. IANA Considerations 76
7. Security Considerations 77
8. Acknowledgement 77
9. Contributors 78
10. References 78
10.1. Normative References 78
10.2. Informative References 79
Authors' Addresses 86

1. Introduction

YANG [RFC6020] and [RFC7950] is a data modeling language used to model configuration data, state data, Remote Procedure Calls, and notifications for network management protocols such as NETCONF [RFC6241]. The YANG language supports a small set of built-in data types and provides mechanisms to derive other types from the built-in types.

This document introduces a collection of common data types derived from the built-in YANG data types. The derived types and groupings are designed to be the common types applicable for modeling Traffic Engineering (TE) features in model(s) defined outside of this document.

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP

14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The terminology for describing YANG data models is found in [RFC7950].

1.2. Prefixes in Data Node Names

In this document, names of data nodes and other data model objects are prefixed using the standard prefix associated with the corresponding YANG imported modules, as shown in Table 1.

Prefix	YANG module	Reference
yang	ietf-yang-types	[RFC6991]
inet	ietf-inet-types	[RFC6991]
rt-types	ietf-routing-types	[RFC8294]
te-types	ietf-te-types	this document
te-packet-types	ietf-te-packet-types	this document

Table 1: Prefixes and corresponding YANG modules

2. Acronyms and Abbreviations

GMPLS: Generalized Multiprotocol Label Switching

LSP: Label Switched Path

LSR: Label Switching Router

LER: Label Edge Router

MPLS: Multiprotocol Label Switching

RSVP: Resource Reservation Protocol

TE: Traffic Engineering

DS-TE: Differentiated Services Traffic Engineering

SRLG: Shared Link Risk Group

NBMA: Non-Broadcast Multiple-access Network

APS: Automatic Protection Switching

SD: Signal Degrade

SF: Signal Fail

WTR: Wait to Restore

PM: Performance Metrics

3. Overview

This document defines two YANG modules for common TE types: ietf-te-types for TE generic types and ietf-te-packet-types for packet specific types. Other technology specific TE types are outside the scope of this document.

3.1. TE Types Module Contents

The ietf-te-types module contains common TE types that are independent and agnostic of any specific technology or control plane instance.

The ietf-te-types module contains the following YANG reusable types and groupings:

te-bandwidth:

A YANG grouping that defines the generic TE bandwidth. The modeling structure allows augmentation for each technology. For un-specified technologies, the string encoded te-bandwidth type is used.

te-label:

A YANG grouping that defines the generic TE label. The modeling structure allows augmentation for each technology. For un-specified technologies, rt-types:generalized-label is used.

performance-metrics-attributes:

A YANG grouping that defines one-way and two-way measured performance metrics and anomalous indication on link(s) or the path as defined in [RFC7471], [RFC8570], and [RFC7823].

performance-metrics-throttle-container:

A YANG grouping that defines configurable thresholds for advertisement suppression and measurement intervals.

te-ds-class:

A type representing the Differentiated-Services (DS) Class-Type of traffic as defined in [RFC4124].

te-label-direction:

An enumerated type for specifying the forward or reverse direction of a label.

te-hop-type:

An enumerated type for specifying hop as loose or strict.

te-global-id:

A type representing the identifier that uniquely identify an operator, which can be either a provider or a client. The definition of this type is taken from [RFC6370] and [RFC5003]. This attribute type is used solely to provide a globally unique context for TE topologies.

te-node-id:

A type representing the identifier for a node in a TE topology. The identifier is represented as 32-bit unsigned integer in the dotted-quad notation. This attribute MAY be mapped to the Router Address described in Section 2.4.1 of [RFC3630], the TE Router ID described in Section 3 of [RFC6827], the Traffic Engineering Router ID described in Section 4.3 of [RFC5305], or the TE Router ID described in Section 3.2.1 of [RFC6119]. The reachability of such a TE node MAY be achieved by a mechanism such as Section 6.2 of [RFC6827].

te-topology-id:

A type representing the identifier for a topology. It is optional to have one or more prefixes at the beginning, separated by colons. The prefixes can be the network-types, defined in ietf-network [RFC8345], to help user to understand the topology better before further inquiry.

te-tp-id:

A type representing the identifier of a TE interface link termination endpoint (TP) on a specific TE node where the TE link connects. This attribute is mapped to local or remote link identifier in [RFC3630] and [RFC5305].

te-path-disjointness:

A type representing the different resource disjointness options for a TE tunnel path as defined in [RFC4872].

admin-groups:

A union type for TE link's classic or extended administrative groups as defined in [RFC3630] and [RFC5305].

srlg:

A type representing the Shared Risk Link Group (SRLG) as defined in [RFC4203] and [RFC5307].

te-metric:

A type representing the TE link metric as defined in [RFC3785].

te-recovery-status:

An enumerated type for the different status of a recovery action as defined in [RFC4427] and [RFC6378].

path-attribute-flags:

A base YANG identity for supported LSP path flags as defined in [RFC3209], [RFC4090], [RFC4736], [RFC5712], [RFC4920], [RFC5420], [RFC7570], [RFC4875], [RFC5151], [RFC5150], [RFC6001], [RFC6790], [RFC7260], [RFC8001], [RFC8149], and [RFC8169].

link-protection-type:

A base YANG identity for supported link protection types as defined in [RFC4872], [RFC4427]

restoration-scheme-type:

A base YANG identity for supported LSP restoration schemes as defined in [RFC4872].

protection-external-commands:

A base YANG identity for supported protection external commands for trouble shooting purposes as defined in [RFC4427].

association-type:

A base YANG identity for supported Label Switched Path (LSP) association types as defined in [RFC6780], [RFC4872], [RFC4873].

objective-function-type:

A base YANG identity for supported path computation objective functions as defined in [RFC5541].

te-tunnel-type:

A base YANG identity for supported TE tunnel types as defined in [RFC3209] and [RFC4875].

lsp-encoding-types:

base YANG identity for supported LSP encoding types as defined in [RFC3471].

lsp-protection-type:

A base YANG identity for supported LSP protection types as defined in [RFC4872] and [RFC4873].

switching-capabilities:

A base YANG identity for supported interface switching capabilities as defined in [RFC3471].

resource-affinities-type:

A base YANG identity for supported attribute filters associated with a tunnel that must be satisfied for a link to be acceptable as defined in [RFC2702] and [RFC3209].

path-metric-type:

A base YANG identity for supported path metric types as defined in [RFC3785] and [RFC7471].

explicit-route-hop:

A YANG grouping that defines supported explicit routes as defined in [RFC3209] and [RFC3477].

te-link-access-type:

An enumerated type for the different TE link access types as defined in [RFC3630].

3.2. Packet TE Types Module Contents

The ietf-te-packet-types module covers the common types and groupings specific packet technology.

The ietf-te-packet-types module contains the following YANG reusable types and groupings:

backup-protection-type:

A base YANG identity for supported protection types that a backup or bypass tunnel can provide as defined in [RFC4090].

te-class-type:

A type that represents the Diffserv-TE class-type as defined in [RFC4124].

bc-type:

A type that represents the Diffserv-TE Bandwidth Constraint (BC) as defined in [RFC4124].

bc-model-type:

A base YANG identity for supported Diffserv-TE bandwidth constraint models as defined in [RFC4125], [RFC4126] and [RFC4127].

te-bandwidth-requested-type:

An enumerated type for the different options to request bandwidth for a specific tunnel.

performance-metrics-attributes-packet:

A YANG grouping for the augmentation of packet specific metrics to the generic performance metrics grouping parameters.

4. TE Types YANG Module

The ietf-te-types module imports from the following modules:

- ietf-yang-types and ietf-inet-types defined in [RFC6991]
- ietf-routing-types defined in [RFC8294]

In addition to the references cross-referenced in Section 3.1, this model also references the following RFCs in defining the types and YANG grouping of the YANG module: [RFC3272], [RFC4202], [RFC4328], [RFC4657], [RFC5817], [RFC6004], [RFC6511], [RFC6205], [RFC7139], [RFC7308], [RFC7551], [RFC7571], [RFC7579], [RFC4090], [RFC4561] and [RFC7951].

```
<CODE BEGINS> file "ietf-te-types@2019-07-05.yang"
module ietf-te-types {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-te-types";

  /* Replace with IANA when assigned */
  prefix "te-types";

  import ietf-inet-types {
    prefix inet;
    reference "RFC6991: Common YANG Data Types";
  }

  import ietf-yang-types {
    prefix "yang";
    reference "RFC6991: Common YANG Data Types";
  }

  import ietf-routing-types {
    prefix "rt-types";
    reference "RFC8294: Common YANG Data Types for the Routing Area";
  }

  organization
    "IETF Traffic Engineering Architecture and Signaling (TEAS)
     Working Group";

  contact
    "WG Web:   <http://tools.ietf.org/wg/teas/>
     WG List:  <mailto:teas@ietf.org>

     WG Chair: Lou Berger
               <mailto:lberger@labn.net>

     WG Chair: Vishnu Pavan Beeram
               <mailto:vbeeram@juniper.net>

     Editor:   Tarek Saad
               <mailto:tsaad@juniper.net>

     Editor:   Rakesh Gandhi
```

```
               <mailto:rgandhi@cisco.com>

     Editor:   Vishnu Pavan Beeram
               <mailto:vbeeram@juniper.net>

     Editor:   Himanshu Shah
               <mailto:hshah@ciena.com>

     Editor:   Xufeng Liu
               <mailto:xufeng.liu.ietf@gmail.com>

     Editor:   Igor Bryskin
               <mailto:Igor.Bryskin@huawei.com>

     Editor:   Young Lee
               <mailto:leeyoung@huawei.com>";

  description
    "This module contains a collection of generally useful TE
     specific YANG data type definitions. The model fully conforms
     to the Network Management Datastore Architecture (NMDA).

     Copyright (c) 2018 IETF Trust and the persons
     identified as authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject
     to the license terms contained in, the Simplified BSD License
     set forth in Section 4.c of the IETF Trust's Legal Provisions
     Relating to IETF Documents
     (https://trustee.ietf.org/license-info).
     This version of this YANG module is part of RFC XXXX; see
     the RFC itself for full legal notices.";

  // RFC Ed.: replace XXXX with actual RFC number and remove this
  // note.

  // RFC Ed.: update the date below with the date of RFC publication
  // and remove this note.

  revision "2019-07-05" {
    description "Latest revision of TE types";
    reference
      "RFC XXXX: A YANG Data Model for Common Traffic Engineering
       Types";
  }

  /**
```

```
   * Typedefs
   */
  typedef admin-group {
    type yang:hex-string {
      /* 01:02:03:04 */
      length "1..11";
    }
    description
      "Administrative group/Resource class/Color representation in
       hex-string type.";
    reference "RFC3630 and RFC5305";
  }

  typedef admin-groups {
    type union {
      type admin-group;
      type extended-admin-group;
    }
    description "TE administrative group derived type";
  }

  typedef extended-admin-group {
    type yang:hex-string;
    description
      "Extended administrative group/Resource class/Color
       representation in hex-string type";
    reference "RFC7308";
  }

  typedef path-attribute-flags {
    type union {
      type identityref {
        base session-attributes-flags;
      }
      type identityref {
        base lsp-attributes-flags;
      }
    }
    description "Path attributes flags type";
  }

  typedef performance-metrics-normality {
    type enumeration {
      enum "unknown" {
        value 0;
        description
          "Unknown.";
      }
```

```
      enum "normal" {
        value 1;
        description
          "Normal.";
      }
      enum "abnormal" {
        value 2;
        description
          "Abnormal. The anomalous bit is set.";
      }
    }
    description
      "Indicates whether a performance metric is normal, abnormal, or
       unknown.";
    reference
      "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
       RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
       RFC7823: Performance-Based Path Selection for Explicitly
       Routed Label Switched Paths (LSPs) Using TE Metric
       Extensions";
  }

  typedef srlg {
    type uint32;
    description "SRLG type";
    reference "RFC4203 and RFC5307";
  }

  typedef te-admin-status {
    type enumeration {
      enum up {
        description
          "Enabled.";
      }
      enum down {
        description
          "Disabled.";
      }
      enum testing {
        description
          "In some test mode.";
      }
      enum preparing-maintenance {
        description
          "Resource is disabled in the control plane to prepare for
           graceful shutdown for maintenance purposes.";
        reference
          "RFC5817: Graceful Shutdown in MPLS and Generalized MPLS
```

```
          Traffic Engineering Networks";
      }
      enum maintenance {
        description
          "Resource is disabled in the data plane for maintenance
           purposes.";
      }
      enum unknown {
        description
          "Status is unknown";
      }
    }
    description
      "Defines a type representing the administrative status of
       a TE resource.";
  }

  typedef te-bandwidth {
    type string {
      pattern
        '0[xX](0((\.0?)?[pP](\+)?0?|(\.0?))|'
      + '1(\.([\da-fA-F]{0,5}[02468aAcCeE]?)?)?[pP](\+)?(12[0-7]|'
      + '1[01]\d|0?\d?\d)?)|0[xX][\da-fA-F]{1,8}|\d+'
      + '(,(0[xX](0((\.0?)?[pP](\+)?0?|(\.0?))|'
      + '1(\.([\da-fA-F]{0,5}[02468aAcCeE]?)?)?[pP](\+)?(12[0-7]|'
      + '1[01]\d|0?\d?\d)?)|0[xX][\da-fA-F]{1,8}|\d+))*';
    }
    description
      "This is the generic bandwidth type that is a string containing
       a list of numbers separated by commas, with each of these
       number can be non-negative decimal, hex integer, or hex float:
       (dec | hex | float)[*(','(dec | hex | float))]
       For packet switching type, a float number is used, such as
       0x1p10.
       For OTN switching type, a list of integers can be used, such
       as '0,2,3,1', indicating 2 odu0's and 1 odu3.
       For DWDM, a list of pairs of slot number and width can be
       used, such as '0, 2, 3, 3', indicating a frequency slot 0 with
       slot width 2 and a frequency slot 3 with slot width 3.
       Canonically, the string is represented as all lowercase and in
       hex where the prefix '0x' precedes the hex number";
  } // te-bandwidth

  typedef te-ds-class {
    type uint8 {
      range "0..7";
    }
    description
```

```
      "The Differentiated Class-Type of traffic.";
    reference "RFC4124: section-4.3.1";
  }

  typedef te-global-id {
    type uint32;
    description
      "An identifier to uniquely identify an operator, which can be
       either a provider or a client.
       The definition of this type is taken from RFC6370 and RFC5003.
       This attribute type is used solely to provide a globally
       unique context for TE topologies.";
  }

  typedef te-hop-type {
    type enumeration {
      enum loose {
        description
          "loose hop in an explicit path";
      }
      enum strict {
        description
          "strict hop in an explicit path";
      }
    }
    description
     "enumerated type for specifying loose or strict
      paths";
    reference "RFC3209: section-4.3.2";
  }

  typedef te-link-access-type {
    type enumeration {
      enum point-to-point {
        description
          "The link is point-to-point.";
      }
      enum multi-access {
        description
          "The link is multi-access, including broadcast and NBMA.";
      }
    }
    description
      "Defines a type representing the access type of a TE link.";
    reference
      "RFC3630: Traffic Engineering (TE) Extensions to OSPF
       Version 2.";
  }
```

```
  typedef te-label-direction {
    type enumeration {
      enum forward {
        description
          "Label allocated for the forward LSP direction";
      }
      enum reverse {
        description
          "Label allocated for the reverse LSP direction";
      }
    }
    description
     "enumerated type for specifying the forward or reverse
     label";
  }

  typedef te-link-direction {
    type enumeration {
      enum incoming {
        description
          "explicit route represents an incoming link on a node";
      }
      enum outgoing {
        description
          "explicit route represents an outgoing link on a node";
      }
    }
    description
     "enumerated type for specifying direction of link on a node";
  }

  typedef te-metric {
    type uint32;
    description "TE link metric";
    reference "RFC3785";
  }

  typedef te-node-id {
    type yang:dotted-quad;
    description
      "A type representing the identifier for a node in a TE
       topology.
       The identifier is represented as 32-bit unsigned integer in
       the dotted-quad notation.
       This attribute MAY be mapped to the Router Address described
       in Section 2.4.1 of [RFC3630], the TE Router ID described in
       Section 3 of [RFC6827], the Traffic Engineering Router ID
       described in Section 4.3 of [RFC5305], or the TE Router ID
```

```
       described in Section 3.2.1 of [RFC6119].
       The reachability of such a TE node MAY be achieved by a
       mechanism such as Section 6.2 of [RFC6827].";
  }

  typedef te-oper-status {
    type enumeration {
      enum up {
        description
        "Operational up.";
      }
      enum down {
        description
        "Operational down.";
      }
      enum testing {
        description
        "In some test mode.";
      }
      enum unknown {
        description
        "Status cannot be determined for some reason.";
      }
      enum preparing-maintenance {
        description
          "Resource is disabled in the control plane to prepare for
           graceful shutdown for maintenance purposes.";
        reference
          "RFC5817: Graceful Shutdown in MPLS and Generalized MPLS
           Traffic Engineering Networks";
      }
      enum maintenance {
        description
          "Resource is disabled in the data plane for maintenance
           purposes.";
      }
    }
    description
      "Defines a type representing the operational status of
       a TE resource.";
  }

  typedef te-path-disjointness {
    type bits {
      bit node {
        position 0;
        description "Node disjoint.";
      }
```

```
      bit link {
        position 1;
        description "Link disjoint.";
      }
      bit srlg {
        position 2;
        description "SRLG (Shared Risk Link Group) disjoint.";
      }
    }
    description
      "Type of the resource disjointness for a TE tunnel path.";
    reference
      "RFC4872: RSVP-TE Extensions in Support of End-to-End
       Generalized Multi-Protocol Label Switching (GMPLS)
       Recovery";
  } // te-path-disjointness

  typedef te-recovery-status {
    type enumeration {
      enum normal {
        description
          "Both the recovery and working spans are fully
           allocated and active, data traffic is being
           transported over (or selected from) the working
           span, and no trigger events are reported.";
      }
      enum recovery-started {
        description
          "The recovery action has been started, but not completed.";
      }
      enum recovery-succeeded {
        description
          "The recovery action has succeeded. The working span has
           reported a failure/degrade condition and the user traffic
           is being transported (or selected) on the recovery span.";
      }
      enum recovery-failed {
        description
          "The recovery action has failed.";
      }
      enum reversion-started {
        description
          "The reversion has started.";
      }
      enum reversion-succeeded {
        description
          "The reversion action has succeeded.";
      }
```

```
      enum reversion-failed {
        description
          "The reversion has failed.";
      }
      enum recovery-unavailable {
        description
          "The recovery is unavailable -- either as a result of an
           operator Lockout command or a failure condition detected
           on the recovery span.";
      }
      enum recovery-admin {
        description
          "The operator has issued a command switching the user
           traffic to the recovery span.";
      }
      enum wait-to-restore {
        description
          "The recovery domain is recovering from a failure/degrade
           condition on the working span that is being controlled by
           the Wait-to-Restore (WTR) timer.";
      }
    }
    description
      "Defines the status of a recovery action.";
    reference
      "RFC4427: Recovery (Protection and Restoration) Terminology
       for Generalized Multi-Protocol Label Switching (GMPLS).
       RFC6378: MPLS Transport Profile (MPLS-TP) Linear Protection";
  }

  typedef te-template-name {
    type string {
      pattern '/?([a-zA-Z0-9\-_.]+)(/[a-zA-Z0-9\-_.]+)*';
    }
    description
      "A type for the name of a TE node template or TE link
       template.";
  }

  typedef te-topology-event-type {
    type enumeration {
      enum "add" {
        value 0;
        description
          "A TE node or te-link has been added.";
      }
      enum "remove" {
        value 1;
```

```
        description
          "A TE node or te-link has been removed.";
      }
      enum "update" {
        value 2;
        description
          "A TE node or te-link has been updated.";
      }
    }
    description "TE Event type for notifications";
  } // te-topology-event-type

  typedef te-topology-id {
    type union {
        type string {
          length 0; // empty string
        }
        type string {
          pattern
            '([a-zA-Z0-9\-_.]+:)*'
          + '/?([a-zA-Z0-9\-_.]+)(/[a-zA-Z0-9\-_.]+)*';
        }
    }
    description
      "An identifier for a topology.
       It is optional to have one or more prefixes at the beginning,
       separated by colons. The prefixes can be the network-types,
       defined in ietf-network.yang, to help user to understand the
       topology better before further inquiry.";
    reference "RFC8345";
  }

  typedef te-tp-id {
    type union {
      type uint32;          // Unnumbered
      type inet:ip-address; // IPv4 or IPv6 address
    }
    description
      "An identifier for a TE link endpoint on a node.
       This attribute is mapped to local or remote link identifier in
       RFC3630 and RFC5305.";
  }

  /* TE features */
  feature p2mp-te {
    description
      "Indicates support for P2MP-TE";
    reference "RFC4875";
```

```
  }

  feature frr-te {
    description
      "Indicates support for TE FastReroute (FRR)";
    reference "RFC4090";
  }

  feature extended-admin-groups {
    description
      "Indicates support for TE link extended admin
      groups.";
    reference "RFC7308";
  }

  feature named-path-affinities {
    description
      "Indicates support for named path affinities";
  }

  feature named-extended-admin-groups {
    description
      "Indicates support for named extended admin groups";
  }

  feature named-srlg-groups {
    description
      "Indicates support for named SRLG groups";
  }

  feature named-path-constraints {
    description
      "Indicates support for named path constraints";
  }

  feature path-optimization-metric {
    description
      "Indicates support for path optimization metric";
  }

  feature path-optimization-objective-function {
    description
      "Indicates support for path optimization objective function";
  }

  /*
   * Identities
   */
```

```
  identity session-attributes-flags {
    description
      "Base identity for the RSVP-TE session attributes flags";
  }
  identity local-protection-desired {
    base session-attributes-flags;
    description "Fastreroute local protection is desired.";
    reference "RFC3209";
  }
  identity se-style-desired {
    description
      "Shared explicit style to allow the LSP to be
       established sharing resources with the old LSP.";
    reference "RFC3209";
  }
  identity local-recording-desired {
    description "Local recording desired";
    reference "RFC3209";
  }
  identity bandwidth-protection-desired {
    base session-attributes-flags;
    description
      "Request FRR bandwidth protection on LSRs if
      present.";
    reference "RFC4090";
  }
  identity node-protection-desired {
    base session-attributes-flags;
    description
      "Request FRR node protection on LSRs if
      present.";
    reference "RFC4090";
  }
  identity path-reevaluation-request {
    base session-attributes-flags;
    description
      "This flag indicates that a path re-evaluation (of the
       current path in use) is requested. Note that this does
       not trigger any LSP Reroute but instead just signals a
       request to evaluate whether a preferable path exists.";
    reference "RFC4736";
  }
  identity soft-preemption-desired {
    base session-attributes-flags;
    description
      "Soft-preemption of LSP resources is desired";
    reference "RFC5712";
  }
```

```
  identity lsp-attributes-flags {
    description "Base identity for per hop attribute flags";
  }
  identity end-to-end-rerouting-desired {
    base lsp-attributes-flags;
    description
      "Indicates end-to-end re-routing behavior for an
       LSP under establishment.  This MAY also be used for
       specifying the behavior of end-to-end LSP recovery for
       established LSPs.";
    reference "RFC4920, RFC5420, RFC7570";
  }
  identity boundary-rerouting-desired {
    base lsp-attributes-flags;
    description
      "Indicates boundary re-routing behavior for an LSP under
       establishment. This MAY also be used for specifying the
       segment-based LSP recovery through nested crankback for
       established LSPs.  The boundary ABR/ASBR can either decide
       to forward the PathErr message upstream to an upstream boundary
       ABR/ASBR or to the ingress LSR.
       Alternatively, it can try to select another egress boundary
       LSR.";
    reference "RFC4920, RFC5420, RFC7570";
  }
  identity segment-based-rerouting-desired {
    base lsp-attributes-flags;
    description
      "Indicates segment-based re-routing behavior for an LSP under
       establishment. This MAY also be used to specify the segment-
       based LSP recovery for established LSPs.";
    reference "RFC4920, RFC5420, RFC7570";
  }
  identity lsp-integrity-required {
    base lsp-attributes-flags;
    description "Indicates LSP integrity is required";
    reference "RFC4875, RFC7570";

  }
  identity contiguous-lsp-desired {
    base lsp-attributes-flags;
    description "Indicates contiguous LSP is desired";
    reference "RFC5151, RFC7570";
  }
  identity lsp-stitching-desired {
    base lsp-attributes-flags;
    description "Indicates LSP stitching is desired";
    reference "RFC5150, RFC7570";
```

```
  }
  identity pre-planned-lsp-flag {
    base lsp-attributes-flags;
    description
      "Indicates the LSP MUST be provisioned in the
       control plane only.";
    reference "RFC6001, RFC7570";
  }
  identity non-php-behavior-flag {
    base lsp-attributes-flags;
    description
      "Indicates non-php behavior for the LSP is desired";
    reference "RFC6511, RFC7570";
  }
  identity oob-mapping-flag {
    base lsp-attributes-flags;
    description
      "Indicates signaling of the egress binding information
       is out-of-band , (e.g., via Border Gateway Protocol (BGP))";
    reference "RFC6511, RFC7570";
  }
  identity entropy-label-capability {
    base lsp-attributes-flags;
    description "Indicates entropy label capability";
    reference "RFC6790, RFC7570";
  }
  identity oam-mep-entity-desired {
    base lsp-attributes-flags;
    description "OAM MEP entities desired";
    reference "RFC7260";
  }
  identity oam-mip-entity-desired {
    base lsp-attributes-flags;
    description "OAM MIP entities desired";
    reference "RFC7260";
  }
  identity srlg-collection-desired {
    base lsp-attributes-flags;
    description "SRLG collection desired";
    reference "RFC8001, RFC7570";
  }
  identity loopback-desired {
    base lsp-attributes-flags;
    description
      "This flag indicates a particular node on the LSP is
       required to enter loopback mode.  This can also be
       used for specifying the loopback state of the node.";
    reference "RFC7571";
```

```
  }
  identity p2mp-te-tree-eval-request {
    base lsp-attributes-flags;
    description "P2MP-TE tree re-evaluation request";
    reference "RFC8149";
  }
  identity rtm-set-desired {
    base lsp-attributes-flags;
    description
      "Residence Time Measurement (RTM) attribute flag";
    reference "RFC8169";
  }

  identity link-protection-type {
    description "Base identity for link protection type.";
  }
  identity link-protection-unprotected {
    base link-protection-type;
    description "Unprotected link type";
    reference "RFC4872";
  }
  identity link-protection-extra-traffic {
    base link-protection-type;
    description "Extra-traffic protected link type";
    reference "RFC4427.";
  }
  identity link-protection-shared {
    base link-protection-type;
    description "Shared protected link type";
    reference "RFC4872";
  }
  identity link-protection-1-for-1 {
    base link-protection-type;
    description "One for one protected link type";
    reference "RFC4872";
  }
  identity link-protection-1-plus-1 {
    base link-protection-type;
    description "One plus one protected link type";
    reference "RFC4872";
  }
  identity link-protection-enhanced {
    base link-protection-type;
    description "Enhanced protection protected link type";
    reference "RFC4872";
  }

  identity association-type {
```

```
    description "Base identity for tunnel association";
  }
  identity association-type-recovery {
    base association-type;
    description
      "Association Type Recovery used to association LSPs of
       same tunnel for recovery";
    reference "RFC6780, RFC4872";
  }
  identity association-type-resource-sharing {
    base association-type;
    description
      "Association Type Resource Sharing used to enable resource
       sharing during make-before-break.";
    reference "RFC6780, 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 "RFC6780,RFC7551";
  }

  identity objective-function-type {
    description "Base objective function type";
  }
  identity of-minimize-cost-path {
    base objective-function-type;
    description
        "Minimize cost of path objective function";
    reference "RFC5541";
  }
  identity of-minimize-load-path {
    base objective-function-type;
    description
        "Minimize the load on path(s) objective
         function";
```

```
    reference "RFC5541";
  }
  identity of-maximize-residual-bandwidth {
    base objective-function-type;
    description
        "Maximize the residual bandwidth objective
         function";
    reference "RFC5541";
  }
  identity of-minimize-agg-bandwidth-consumption {
    base objective-function-type;
    description
        "minimize the aggregate bandwidth consumption
         objective function";
    reference "RFC5541";
  }
  identity of-minimize-load-most-loaded-link {
    base objective-function-type;
    description
        "Minimize the load on the most loaded link
         objective function";
    reference "RFC5541";
  }
  identity of-minimize-cost-path-set {
    base objective-function-type;
    description
        "Minimize the cost on a path set objective
         function";
    reference "RFC5541";
  }

  identity path-computation-method {
    description
     "base identity for supported path computation
      mechanisms";
  }
  identity path-locally-computed {
    base path-computation-method;
    description
      "indicates a constrained-path LSP in which the
      path is computed by the local LER";
    reference "RFC3209";
  }
  identity path-externally-queried {
    base path-computation-method;
    description
     "Constrained-path LSP in which the path is
      obtained by querying an external source, such as a PCE server.
```

```
      In the case that an LSP is defined to be externally queried, it
      may also have associated explicit definitions (provided
      to the external source to aid computation); and the path that is
      returned by the external source is not required to provide a
      wholly resolved path back to the originating system - that is to
      say, some local computation may also be required";
    reference "RFC4657";
  }
  identity path-explicitly-defined {
    base path-computation-method;
    description
     "constrained-path LSP in which the path is
      explicitly specified as a collection of strict or/and loose
      hops";
    reference "RFC3209";
  }

  identity lsp-metric-type {
    description
      "Base identity for types of LSP metric specification";
  }
  identity lsp-metric-relative {
    base lsp-metric-type;
    description
      "The metric specified for the LSPs to which this identity refers
      is specified as a relative value to the IGP metric cost to the
      LSP's tail-end.";
    reference "RFC4657";
  }
  identity lsp-metric-absolute {
    base lsp-metric-type;
    description
      "The metric specified for the LSPs to which this identity refers
      is specified as an absolute value";
    reference "RFC4657";
  }
  identity lsp-metric-inherited {
    base lsp-metric-type;
    description
      "The metric for the LSPs to which this identity refers is
      not specified explicitly - but rather inherited from the IGP
      cost directly";
    reference "RFC4657";
  }

  identity te-tunnel-type {
    description
      "Base identity from which specific tunnel types are
```

```
      derived.";
  }
  identity te-tunnel-p2p {
    base te-tunnel-type;
    description
      "TE point-to-point tunnel type.";
    reference "RFC3209";
  }
  identity te-tunnel-p2mp {
    base te-tunnel-type;
    description
      "TE point-to-multipoint tunnel type.";
    reference "RFC4875";
  }

  identity tunnel-action-type {
    description
      "Base identity from which specific tunnel action types
       are derived.";
  }
  identity tunnel-action-resetup {
    base tunnel-action-type;
    description
      "TE tunnel action resetup. Tears the
      tunnel's current LSP (if any) and
      attempts to re-establish a new LSP";
  }
  identity tunnel-action-reoptimize {
    base tunnel-action-type;
    description
      "TE tunnel action reoptimize.
       Reoptimizes placement of the tunnel LSP(s)";
  }
  identity tunnel-action-switchpath {
    base tunnel-action-type;
    description
      "TE tunnel action switchpath
       Switches the tunnel's LSP to use the specified path";
  }

  identity te-action-result {
    description
      "Base identity from which specific TE action results
       are derived.";
  }
  identity te-action-success {
    base te-action-result;
    description "TE action successful.";
```

```
  }
  identity te-action-fail {
    base te-action-result;
    description "TE action failed.";
  }
  identity tunnel-action-inprogress {
    base te-action-result;
    description "TE action inprogress.";
  }

  identity tunnel-admin-state-type {
    description
      "Base identity for TE tunnel admin states";
  }
  identity tunnel-admin-state-up {
    base tunnel-admin-state-type;
    description "Tunnel administratively state up";
  }
  identity tunnel-admin-state-down {
    base tunnel-admin-state-type;
    description "Tunnel administratively state down";
  }

  identity tunnel-state-type {
    description
      "Base identity for TE tunnel states";
  }
  identity tunnel-state-up {
    base tunnel-state-type;
    description "Tunnel state up";
  }
  identity tunnel-state-down {
    base tunnel-state-type;
    description "Tunnel state down";
  }

  identity lsp-state-type {
    description
      "Base identity for TE LSP states";
  }
  identity lsp-path-computing {
    base lsp-state-type;
    description
      "State path compute in progress";
  }
  identity lsp-path-computation-ok {
    base lsp-state-type;
    description
```

```
      "State path compute successful";
  }
  identity lsp-path-computation-failed {
    base lsp-state-type;
    description
      "State path compute failed";
  }
  identity lsp-state-setting-up {
    base lsp-state-type;
    description
      "State setting up";
  }
  identity lsp-state-setup-ok {
    base lsp-state-type;
    description
      "State setup successful";
  }
  identity lsp-state-setup-failed {
    base lsp-state-type;
    description
      "State setup failed";
  }
  identity lsp-state-up {
    base lsp-state-type;
    description "State up";
  }
  identity lsp-state-tearing-down {
    base lsp-state-type;
    description
      "State tearing down";
  }
  identity lsp-state-down {
    base lsp-state-type;
    description "State down";
  }

  identity path-invalidation-action-type {
    description
      "Base identity for TE path invalidation action types";
  }
  identity path-invalidation-action-drop-type {
    base path-invalidation-action-type;
    description
      "TE path invalidation action drop";
  }
  identity path-invalidation-action-drop-tear {
    base path-invalidation-action-type;
    description
```

```
      "TE path invalidation action tear";
  }

  identity lsp-restoration-type {
    description
      "Base identity from which LSP restoration types are
       derived.";
  }
  identity lsp-restoration-restore-any {
    base lsp-restoration-type;
    description
      "Restores when any of the LSPs is affected by a failure";
  }
  identity lsp-restoration-restore-all {
    base lsp-restoration-type;
    description
      "Restores when all the tunnel LSPs are affected by failure";
  }

  identity restoration-scheme-type {
    description
      "Base identity for LSP restoration schemes";
  }
  identity restoration-scheme-preconfigured {
    base restoration-scheme-type;
    description
      "Restoration LSP is preconfigured prior to the failure";
    reference "RFC4427";
  }
  identity restoration-scheme-precomputed {
    base restoration-scheme-type;
    description
      "Restoration LSP is precomputed prior to the failure";
    reference "RFC4427";
  }
  identity restoration-scheme-presignaled {
    base restoration-scheme-type;
    description
      "Restoration LSP is presignaled prior to the failure";
    reference "RFC4427";
  }

  identity lsp-protection-type {
    description
      "Base identity from which LSP protection types are
      derived.";
    reference "RFC4872";
  }
```

```
  identity lsp-protection-unprotected {
    base lsp-protection-type;
    description
      "LSP protection 'Unprotected'";
    reference "RFC4872";
  }
  identity lsp-protection-reroute-extra {
    base lsp-protection-type;
    description
      "LSP protection '(Full) Rerouting'";
    reference "RFC4872";
  }
  identity lsp-protection-reroute {
    base lsp-protection-type;
    description
      "LSP protection 'Rerouting without Extra-Traffic'";
    reference "RFC4872";
  }
  identity lsp-protection-1-for-n {
    base lsp-protection-type;
    description
      "LSP protection '1:N Protection with Extra-Traffic'";
    reference "RFC4872";
  }
  identity lsp-protection-unidir-1-for-1 {
    base lsp-protection-type;
    description
      "LSP protection '1:1 Unidirectional Protection'";
    reference "RFC4872";
  }
  identity lsp-protection-bidir-1-for-1 {
    base lsp-protection-type;
    description
      "LSP protection '1:1 Bidirectional Protection'";
    reference "RFC4872";
  }
  identity lsp-protection-unidir-1-plus-1 {
    base lsp-protection-type;
    description
      "LSP protection '1+1 Unidirectional Protection'";
    reference "RFC4872";
  }
  identity lsp-protection-bidir-1-plus-1 {
    base lsp-protection-type;
    description
      "LSP protection '1+1 Bidirectional Protection'";
    reference "RFC4872";
  }
```

```
  identity lsp-protection-extra-traffic {
    base lsp-protection-type;
    description
      "LSP protection 'Extra-Traffic'";
    reference
      "RFC4427.";
  }

  identity lsp-protection-state {
    description
      "Base identity of protection states for reporting
       purposes.";
  }
  identity normal {
    base lsp-protection-state;
    description "Normal state.";
  }
  identity signal-fail-of-protection {
    base lsp-protection-state;
    description
        "There is a signal fail condition on the protection
         transport entity which has higher priority than the
         forced switch command.";
    reference
        "RFC4427";
  }
  identity lockout-of-protection {
    base lsp-protection-state;
    description
        "A Loss of Protection (LoP) command is active.";
    reference
        "RFC4427";
  }
  identity forced-switch {
    base lsp-protection-state;
    description
        "A forced switch command is active.";
    reference
        "RFC4427";
  }
  identity signal-fail {
    base lsp-protection-state;
    description
        "There is a signal fail condition on either the working
        or the protection path.";
    reference
        "RFC4427";
  }
```

```
  identity signal-degrade {
    base lsp-protection-state;
    description
        "There is an signal degrade condition on either the working
         or the protection path.";
    reference
        "RFC4427";
  }
  identity manual-switch {
    base lsp-protection-state;
    description
        "A manual switch command is active.";
    reference
        "RFC4427";
  }
  identity wait-to-restore {
    base lsp-protection-state;
    description
        "A wait time to restore (WTR) is running.";
    reference
        "RFC4427";
  }
  identity do-not-revert {
    base lsp-protection-state;
    description
        "A DNR condition is active because of a non-revertive
         behavior.";
    reference
        "RFC4427";
  }
  identity failure-of-protocol {
    base lsp-protection-state;
    description
        "The protection is not working because of a failure of
         protocol condition.";
    reference
        "RFC4427";
  }

  identity protection-external-commands {
    description
      "Base identity from which protection external commands
       for trouble shooting purposes are derived.";
  }
  identity action-freeze {
    base protection-external-commands;
    description
      "A temporary configuration action initiated by an operator
```

```
       command to prevent any switch action to be taken and as such
       freezes the current state.";
    reference
      "RFC4427";
  }
  identity clear-freeze {
    base protection-external-commands;
    description
      "An action that clears the active freeze state.";
    reference
      "RFC4427";
  }
  identity action-lockout-of-normal {
    base protection-external-commands;
    description
      "A temporary configuration action initiated by an operator
       command to ensure that the normal traffic is not allowed
       to use the protection transport entity.";
    reference
      "RFC4427";
  }
  identity clear-lockout-of-normal {
    base protection-external-commands;
    description
      "An action that clears the active lockout of normal state.";
    reference
      "RFC4427";
  }
  identity action-lockout-of-protection {
    base protection-external-commands;
    description
      "A temporary configuration action initiated by an operator
       command to ensure that the protection transport entity is
       temporarily not available to transport a traffic signal
       (either normal or extra traffic).";
    reference
        "RFC4427";
  }
  identity action-forced-switch {
    base protection-external-commands;
    description
        "A switch action initiated by an operator command to switch
         the extra traffic signal, the normal traffic signal, or the
         null signal to the protection transport entity, unless an
         equal or higher priority switch command is in effect.";
    reference
        "RFC4427";
  }
```

```
  identity action-manual-switch {
    base protection-external-commands;
    description
        "A switch action initiated by an operator command to switch
         the extra traffic signal, the normal traffic signal, or
         the null signal to the protection transport entity, unless
         a fault condition exists on other transport entities or an
         equal or higher priority switch command is in effect.";
    reference
        "RFC4427";
  }
  identity action-exercise {
    base protection-external-commands;
    description
        "An action to start testing if the APS communication is
         operating correctly. It is lower priority than any other
         state or command.";
    reference
        "RFC4427";
  }
  identity clear {
    base protection-external-commands;
    description
        "An action that clears the active near-end lockout of
         protection, forced switch, manual switch, WTR state,
         or exercise command.";
    reference
        "RFC4427";
  }

  identity switching-capabilities {
    description
      "Base identity for interface switching capabilities";
    reference "RFC3471";
  }
  identity switching-psc1 {
    base switching-capabilities;
    description
      "Packet-Switch Capable-1 (PSC-1)";
    reference "RFC3471";
  }
  identity switching-evpl {
    base switching-capabilities;
    description
      "Ethernet Virtual Private Line (EVPL)";
    reference "RFC6004";
  }
  identity switching-l2sc {
```

```
    base switching-capabilities;
    description
      "Layer-2 Switch Capable (L2SC)";
    reference "RFC3471";
  }
  identity switching-tdm {
    base switching-capabilities;
    description
      "Time-Division-Multiplex Capable (TDM)";
    reference "RFC3471";
  }
  identity switching-otn {
    base switching-capabilities;
    description
      "OTN-TDM capable";
    reference "RFC7138";
  }
  identity switching-dcsc {
    base switching-capabilities;
    description
      "Data Channel Switching Capable (DCSC)";
    reference "RFC6002";
  }
  identity switching-lsc {
    base switching-capabilities;
    description
      "Lambda-Switch Capable (LSC)";
    reference "RFC3471";
  }
  identity switching-fsc {
    base switching-capabilities;
    description
      "Fiber-Switch Capable (FSC)";
    reference "RFC3471";
  }

  identity lsp-encoding-types {
    description
      "Base identity for encoding types";
    reference "RFC3471";
  }
  identity lsp-encoding-packet {
    base lsp-encoding-types;
    description
      "Packet LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-ethernet {
```

```
    base lsp-encoding-types;
    description
      "Ethernet LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-pdh {
    base lsp-encoding-types;
    description
      "ANSI/ETSI LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-sdh {
    base lsp-encoding-types;
    description
      "SDH ITU-T G.707 / SONET ANSI T1.105 LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-digital-wrapper {
    base lsp-encoding-types;
    description
      "Digital Wrapper LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-lambda {
    base lsp-encoding-types;
    description
      "Lambda (photonic) LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-fiber {
    base lsp-encoding-types;
    description
      "Fiber LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-fiber-channel {
    base lsp-encoding-types;
    description
      "Fiber Channel LSP encoding";
    reference "RFC3471";
  }
  identity lsp-encoding-oduk {
    base lsp-encoding-types;
    description
      "G.709 ODUk (Digital Path) LSP encoding";
    reference "RFC4328";
  }
  identity lsp-encoding-optical-channel {
```

```
    base lsp-encoding-types;
    description
      "G.709 Optical Channel LSP encoding";
    reference "RFC4328";
  }
  identity lsp-encoding-line {
    base lsp-encoding-types;
    description
      "Line (e.g., 8B/10B) LSP encoding";
      reference "RFC6004";
  }

  identity path-signaling-type {
    description
      "base identity from which specific LSPs path
       setup types are derived";
  }
  identity path-setup-static {
    base path-signaling-type;
    description
      "Static LSP provisioning path setup";
  }
  identity path-setup-rsvp {
    base path-signaling-type;
    description
      "RSVP-TE signaling path setup";
    reference "RFC3209";
  }
  identity path-setup-sr {
    base path-signaling-type;
    description
      "Segment-routing path setup";
  }

  identity path-scope-type {
    description
      "base identity from which specific path
       scope types are derived";
  }
  identity path-scope-segment {
    base path-scope-type;
    description
      "Path scope segment";
    reference "RFC4873";
  }
  identity path-scope-end-to-end {
    base path-scope-type;
    description
```

```
      "Path scope end to end";
    reference "RFC4873";
  }

  identity route-usage-type {
    description
      "Base identity for route usage";
  }
  identity route-include-object {
    base route-usage-type;
    description
      "Include route object";
  }
  identity route-exclude-object {
    base route-usage-type;
    description
      "Exclude route object";
    reference "RFC4874";
  }
  identity route-exclude-srlg {
    base route-usage-type;
    description "Exclude SRLG";
    reference "RFC4874";
  }

  identity path-metric-type {
    description
      "Base identity for path metric type";
  }
  identity path-metric-te {
    base path-metric-type;
    description
      "TE path metric";
    reference "RFC3785";
  }
  identity path-metric-igp {
    base path-metric-type;
    description
      "IGP path metric";
    reference "RFC3785";
  }
  identity path-metric-hop {
    base path-metric-type;
    description
      "Hop path metric";
  }
  identity path-metric-delay-average {
    base path-metric-type;
```

```
    description
      "Unidirectional average link delay";
    reference "RFC7471";
  }
  identity path-metric-delay-minimum {
    base path-metric-type;
    description
      "Unidirectional minimum link delay";
    reference "RFC7471";
  }
  identity path-metric-residual-bandwidth {
    base path-metric-type;
    description
      "Unidirectional Residual Bandwidth, which is defined to be
       Maximum Bandwidth [RFC3630] minus the bandwidth currently
       allocated to LSPs.";
    reference "RFC7471";
  }
  identity path-metric-optimize-includes {
    base path-metric-type;
    description
      "A metric that optimizes the number of included resources
       specified in a set";
  }
  identity path-metric-optimize-excludes {
    base path-metric-type;
    description
      "A metric that optimizes the number of excluded resources
       specified in a set";
  }

  identity path-tiebreaker-type {
    description
      "Base identity for path tie-breaker type";
  }
  identity path-tiebreaker-minfill {
    base path-tiebreaker-type;
    description
      "Min-Fill LSP path placement";
  }
  identity path-tiebreaker-maxfill {
    base path-tiebreaker-type;
    description
      "Max-Fill LSP path placement";
  }
  identity path-tiebreaker-random {
    base path-tiebreaker-type;
    description
```

```
      "Random LSP path placement";
  }

  identity resource-affinities-type {
    description
      "Base identity for resource affinities";
    reference "RFC2702";
  }
  identity resource-aff-include-all {
    base resource-affinities-type;
    description
      "The set of attribute filters associated with a
      tunnel all of which must be present for a link
      to be acceptable";
    reference "RFC2702 and RFC3209";
  }
  identity resource-aff-include-any {
    base resource-affinities-type;
    description
      "The set of attribute filters associated with a
      tunnel any of which must be present for a link
      to be acceptable";
    reference "RFC2702 and RFC3209";
  }
  identity resource-aff-exclude-any {
    base resource-affinities-type;
    description
      "The set of attribute filters associated with a
      tunnel any of which renders a link unacceptable";
    reference "RFC2702 and RFC3209";
  }

  identity te-optimization-criterion {
    description
      "Base identity for TE optimization criterion.";
    reference
      "RFC3272: Overview and Principles of Internet Traffic
       Engineering.";
  }
  identity not-optimized {
    base te-optimization-criterion;
    description "Optimization is not applied.";
  }
  identity cost {
    base te-optimization-criterion;
    description "Optimized on cost.";
    reference "RFC5541";
  }
```

```
  identity delay {
    base te-optimization-criterion;
    description "Optimized on delay.";
    reference "RFC5541";
  }

  identity path-computation-srlg-type {
    description
      "Base identity for SRLG path computation";
  }
  identity srlg-ignore {
    base path-computation-srlg-type;
    description
      "Ignores SRLGs in path computation";
  }
  identity srlg-strict {
    base path-computation-srlg-type;
    description
      "Include strict SRLG check in path computation";
  }
  identity srlg-preferred {
    base path-computation-srlg-type;
    description
      "Include preferred SRLG check in path computation";
  }
  identity srlg-weighted {
    base path-computation-srlg-type;
    description
      "Include weighted SRLG check in path computation";
  }

  /**
   * TE bandwidth groupings
   **/
  grouping te-bandwidth {
    description
      "This grouping defines the generic TE bandwidth.
       For some known data plane technologies, specific modeling
       structures are specified. The string encoded te-bandwidth
       type is used for un-specified technologies.
       The modeling structure can be augmented later for other
       technologies.";
    container te-bandwidth {
      description
        "Container that specifies TE bandwidth. The choices
         can be augmented for specific dataplane technologies.";
      choice technology {
        default generic;
```

```
        description
          "Data plane technology type.";
        case generic {
          leaf generic {
            type te-bandwidth;
            description
              "Bandwidth specified in a generic format.";
          }
        }
      }
    }
  }

  /**
   * TE label groupings
   **/
  grouping te-label {
    description
      "This grouping defines the generic TE label.
       The modeling structure can be augmented for each technology.
       For un-specified technologies, rt-types:generalized-label
       is used.";
    container te-label {
      description
        "Container that specifies TE label. The choices can
         be augmented for specific dataplane technologies.";
      choice technology {
        default generic;
        description
          "Data plane technology type.";
        case generic {
          leaf generic {
            type rt-types:generalized-label;
            description
              "TE label specified in a generic format.";
          }
        }
      }
      leaf direction {
        type te-label-direction;
        default 'forward';
        description "Label direction";
      }
    }
  }

  grouping te-topology-identifier {
    description
```

```
      "Augmentation for TE topology.";
    container te-topology-identifier {
      description "TE topology identifier container";
      leaf provider-id {
        type te-global-id;
        default 0;
        description
          "An identifier to uniquely identify a provider. If omitted,
           it assumes the default topology provider ID=0";
      }
      leaf client-id {
        type te-global-id;
        default 0;
        description
          "An identifier to uniquely identify a client. If omitted,
           it assumes the default topology client ID=0";
      }
      leaf topology-id {
        type te-topology-id;
        default '';
        description
          "When the datastore contains several topologies, the
           topology-id distinguishes between them. If omitted, the
           default empty string topology-id is assumed";
      }
    }
  }

  /**
   * TE performance metric groupings
   **/
  grouping performance-metrics-one-way-delay-loss {
    description
      "Performance Metric (PM) information in real time that can
      be applicable to links or connections. PM defined
      in this grouping is applicable to generic TE performance
      metrics as well as packet TE performance metrics.";
    reference
      "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
      RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
      RFC7823: Performance-Based Path Selection for Explicitly
      Routed Label Switched Paths (LSPs) Using TE Metric
      Extensions";
    leaf one-way-delay {
      type uint32 {
        range 0..16777215;
      }
      description "One-way delay or latency in micro seconds.";
```

```
    }
    leaf one-way-delay-normality {
      type te-types:performance-metrics-normality;
      description "One-way delay normality.";
    }
  }

  grouping performance-metrics-two-way-delay-loss {
    description
      "Performance metric information in real time that can
      be applicable to links or connections. PM defined
      in this grouping is applicable to generic TE performance
      metrics as well as packet TE performance metrics.";
    reference
      "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
      RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
      RFC7823: Performance-Based Path Selection for Explicitly
      Routed Label Switched Paths (LSPs) Using TE Metric
      Extensions";
    leaf two-way-delay {
      type uint32 {
        range 0..16777215;
      }
      description "Two-way delay or latency in micro seconds.";
    }
    leaf two-way-delay-normality {
      type te-types:performance-metrics-normality;
      description "Two-way delay normality.";
    }
  }

  grouping performance-metrics-one-way-bandwidth {
    description
      "Performance metric information in real time that can
      be applicable to links. PM defined
      in this grouping is applicable to generic TE performance
      metrics as well as packet TE performance metrics.";
    reference
      "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
      RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
      RFC7823: Performance-Based Path Selection for Explicitly
      Routed Label Switched Paths (LSPs) Using TE Metric
      Extensions";

    leaf one-way-residual-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      default '0x0p0';
      description
```

```
        "Residual bandwidth that subtracts tunnel
         reservations from Maximum Bandwidth (or link capacity)
         [RFC3630] and provides an aggregated remainder across QoS
         classes.";
    }
    leaf one-way-residual-bandwidth-normality {
      type te-types:performance-metrics-normality;
      default 'normal';
      description "Residual bandwidth normality.";
    }
    leaf one-way-available-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      default '0x0p0';
      description
        "Available bandwidth that is defined to be residual
         bandwidth minus the measured bandwidth used for the
         actual forwarding of non-RSVP-TE LSP packets.  For a
         bundled link, available bandwidth is defined to be the
         sum of the component link available bandwidths.";
    }
    leaf one-way-available-bandwidth-normality {
      type te-types:performance-metrics-normality;
      default 'normal';
      description "Available bandwidth normality.";
    }
    leaf one-way-utilized-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      default '0x0p0';
      description
        "Bandwidth utilization that represents the actual
         utilization of the link (i.e. as measured in the router).
         For a bundled link, bandwidth utilization is defined to
         be the sum of the component link bandwidth
         utilizations.";
    }
    leaf one-way-utilized-bandwidth-normality {
      type te-types:performance-metrics-normality;
      default 'normal';
      description "Bandwidth utilization normality.";
    }
  }

  grouping one-way-performance-metrics {
    description
      "One-way performance metrics throttle grouping.";
    leaf one-way-delay {
      type uint32 {
        range 0..16777215;
```

```
      }
      default 0;
      description "One-way delay or latency in micro seconds.";
    }
    leaf one-way-residual-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      default '0x0p0';
      description
        "Residual bandwidth that subtracts tunnel
         reservations from Maximum Bandwidth (or link capacity)
         [RFC3630] and provides an aggregated remainder across QoS
         classes.";
    }
    leaf one-way-available-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      default '0x0p0';
      description
        "Available bandwidth that is defined to be residual
         bandwidth minus the measured bandwidth used for the
         actual forwarding of non-RSVP-TE LSP packets.  For a
         bundled link, available bandwidth is defined to be the
         sum of the component link available bandwidths.";
    }
    leaf one-way-utilized-bandwidth {
      type rt-types:bandwidth-ieee-float32;
      default '0x0p0';
      description
        "Bandwidth utilization that represents the actual
         utilization of the link (i.e. as measured in the router).
         For a bundled link, bandwidth utilization is defined to
         be the sum of the component link bandwidth
         utilizations.";
    }
  }

  grouping two-way-performance-metrics {
    description
      "Two-way performance metrics throttle grouping.";
    leaf two-way-delay {
      type uint32 {
        range 0..16777215;
      }
      default 0;
      description "Two-way delay or latency in micro seconds.";
    }
  }

  grouping performance-metrics-thresholds {
```

```
    description
      "Grouping for configurable thresholds for measured attributes";
    uses one-way-performance-metrics;
    uses two-way-performance-metrics;
  }


  grouping performance-metrics-attributes {
    description
      "A container containing performance metric attributes.";
    container performance-metrics-one-way {
      description
        "One-way link performance information in real time.";
      reference
        "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
         RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
         RFC7823: Performance-Based Path Selection for Explicitly
         Routed Label Switched Paths (LSPs) Using TE Metric
         Extensions";
      uses performance-metrics-one-way-delay-loss;
      uses performance-metrics-one-way-bandwidth;
    }
    container performance-metrics-two-way {
      description
        "Two-way link performance information in real time.";
      reference
        "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
         RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
         RFC7823: Performance-Based Path Selection for Explicitly
         Routed Label Switched Paths (LSPs) Using TE Metric
         Extensions";
      uses performance-metrics-two-way-delay-loss;
    }
  }

  grouping performance-metrics-throttle-container {
    description
      "A container controlling performance metric throttle.";
    container throttle {
      must "suppression-interval >= measure-interval" {
        error-message
          "suppression-interval cannot be less then
           measure-interval.";
        description
          "Constraint on suppression-interval and
           measure-interval.";
      }
      description
```

```
        "Link performance information in real time.";
      reference
        "RFC7471: OSPF Traffic Engineering (TE) Metric Extensions.
         RFC8570: IS-IS Traffic Engineering (TE) Metric Extensions.
         RFC7823: Performance-Based Path Selection for Explicitly
         Routed Label Switched Paths (LSPs) Using TE Metric
         Extensions";
      leaf one-way-delay-offset {
        type uint32 {
          range 0..16777215;
        }
        default 0;
        description
          "Offset value to be added to the measured delay value.";
      }
      leaf measure-interval {
        type uint32;
        default 30;
        description
          "Interval in seconds to measure the extended metric
           values.";
      }
      leaf advertisement-interval {
        type uint32;
        default 0;
        description
          "Interval in seconds to advertise the extended metric
           values.";
      }
      leaf suppression-interval {
        type uint32 {
          range "1 .. max";
        }
        default 120;
        description
          "Interval in seconds to suppress advertising the extended
           metric values.";
      }
      container threshold-out {
        uses performance-metrics-thresholds;
        description
          "If the measured parameter falls outside an upper bound
           for all but the min delay metric (or lower bound for
           min-delay metric only) and the advertised value is not
           already outside that bound, anomalous announcement will be
           triggered.";
      }
      container threshold-in {
```

```
        uses performance-metrics-thresholds;
        description
          "If the measured parameter falls inside an upper bound
           for all but the min delay metric (or lower bound for
           min-delay metric only) and the advertised value is not
           already inside that bound, normal (anomalous-flag cleared)
           announcement will be triggered.";
      }
      container threshold-accelerated-advertisement {
        description
          "When the difference between the last advertised value and
           current measured value exceed this threshold, anomalous
           announcement will be triggered.";
        uses performance-metrics-thresholds;
      }
    }
  } // performance-metrics-throttle-container

  /**
   * TE tunnel generic groupings
   **/
  grouping explicit-route-hop {
    description
      "The explicit route entry grouping";
    choice type {
      description
        "The explicit route entry type";
      case numbered-node-hop {
        container numbered-node-hop {
          leaf node-id {
            type te-node-id;
            mandatory true;
            description
              "The identifier of a node in the TE topology.";
          }
          leaf hop-type {
            type te-hop-type;
            default 'strict';
            description "strict or loose hop";
          }
          description "Numbered node route hop";
          reference
            "RFC3209: section 4.3 for EXPLICIT_ROUTE in RSVP-TE
             RFC3477: Signalling Unnumbered Links in RSVP-TE";
        }
      }
      case numbered-link-hop {
        container numbered-link-hop {
```

```
          leaf link-tp-id {
            type te-tp-id;
            mandatory true;
            description
              "TE link termination point identifier.";
          }
          leaf hop-type {
            type te-hop-type;
            default 'strict';
            description "strict or loose hop";
          }
          leaf direction {
            type te-link-direction;
            default 'outgoing';
            description "Link route object direction";
          }
          description
            "Numbered link explicit route hop";
          reference
            "RFC3209: section 4.3 for EXPLICIT_ROUTE in RSVP-TE
             RFC3477: Signalling Unnumbered Links in RSVP-TE";
        }
      }
      case unnumbered-link-hop {
        container unnumbered-link-hop {
          leaf link-tp-id {
            type te-tp-id;
            mandatory true;
            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 node-id {
            type te-node-id;
            mandatory true;
            description
              "The identifier of a node in the TE topology.";
          }
          leaf hop-type {
            type te-hop-type;
            default 'strict';
            description "strict or loose hop";
          }
          leaf direction {
            type te-link-direction;
            default 'outgoing';
            description "Link route object direction";
```

```
          }
          description
            "Unnumbered link explicit route hop";
          reference
            "RFC3209: section 4.3 for EXPLICIT_ROUTE in RSVP-TE
             RFC3477: Signalling Unnumbered Links in RSVP-TE";
        }
      }
      case as-number {
        container as-number-hop {
          leaf as-number {
            type inet:as-number;
            mandatory true;
            description "The AS number";
          }
          leaf hop-type {
            type te-hop-type;
            default 'strict';
            description "strict or loose hop";
          }
          description
            "Autonomous System explicit route hop";
        }
      }
      case label {
        container label-hop {
          description "Label hop type";
          uses te-label;
        }
        description
          "The label explicit route hop type";
      }
    }
  }

  grouping record-route-state {
    description
      "The record route grouping";
    leaf index {
      type uint32;
      description
        "Record route hop index. The index is used to
         identify an entry in the list. The order of entries
         is defined by the user without relying on key values";
    }
    choice type {
      description
        "The record route entry type";
```

```
      case numbered-node-hop {
        container numbered-node-hop {
          description "Numbered node route hop container";
          leaf node-id {
            type te-node-id;
            mandatory true;
            description
              "The identifier of a node in the TE topology.";
          }
          leaf-list flags {
            type path-attribute-flags;
            description "Record route per hop flags";
            reference "RFC3209, RFC4090, RFC4561";
          }
        }
        description "Numbered node route hop";
      }
      case numbered-link-hop {
        container numbered-link-hop {
          description "Numbered link route hop container";
          leaf link-tp-id {
            type te-tp-id;
            mandatory true;
            description
              "Numbered TE link termination point identifier.";
          }
          leaf-list flags {
            type path-attribute-flags;
            description "Record route per hop flags";
            reference "RFC3209, RFC4090, RFC4561";
          }
        }
        description "Numbered link route hop";
      }
      case unnumbered-link-hop {
        container unnumbered-link-hop {
          leaf link-tp-id {
            type te-tp-id;
            mandatory true;
            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 node-id {
            type te-node-id;
            description
              "The identifier of a node in the TE topology.";
```

```
          }
          leaf-list flags {
            type path-attribute-flags;
            description "Record route per hop flags";
            reference "RFC3209, RFC4090, RFC4561";
          }
          description
            "Unnumbered link record route hop";
          reference
            "RFC3477: Signalling Unnumbered Links in
             RSVP-TE";
        }
        description "Unnumbered link route hop";
      }
      case label {
        container label-hop {
          description "Label route hop type";
          uses te-label;
          leaf-list flags {
            type path-attribute-flags;
            description "Record route per hop flags";
            reference "RFC3209, RFC4090, RFC4561";
          }
        }
        description
          "The Label record route entry types";
      }
    }
  }

  grouping label-restriction-info {
    description "Label set item info";
    leaf restriction {
      type enumeration {
        enum inclusive {
          description "The label or label range is inclusive.";
        }
        enum exclusive {
          description "The label or label range is exclusive.";
        }
      }
      default 'inclusive';
      description
        "Whether the list item is inclusive or exclusive.";
    }
    leaf index {
      type uint32;
      description
```

```
        "The index of the label restriction list entry.";
      }
      container label-start {
        must "(not(../label-end/te-label/direction) and" +
                     " not(te-label/direction))"
          + " or "
          +  "(../label-end/te-label/direction = te-label/direction)"
          + " or "
          +  "(not(te-label/direction) and" +
                     " (../label-end/te-label/direction = 'forward'))"
          + " or "
          +  "(not(../label-end/te-label/direction) and" +
                     " (te-label/direction = 'forward'))" {
          error-message
            "label-start and label-end must have the same direction.";
        }
        description
          "This is the starting label if a label range is specified.
           This is the label value if a single label is specified,
           in which case, attribute 'label-end' is not set.";
        uses te-label;
      }
      container label-end {
        must "(not(../label-start/te-label/direction) and" +
                     " not(te-label/direction))"
          + " or "
          +  "(../label-start/te-label/direction = te-label/direction)"
          + " or "
          +  "(not(te-label/direction) and" +
                     " (../label-start/te-label/direction = 'forward'))"
          + " or "
          +  "(not(../label-start/te-label/direction) and" +
                     " (te-label/direction = 'forward'))" {
          error-message
            "label-start and label-end must have the same direction.";
        }
        description
          "The ending label if a label range is specified;
           This attribute is not set, If a single label is
           specified.";
        uses te-label;
      }
      container label-step {
        description
          "The step increment between labels in the label range.
           The label start/end values will have to be consistent
           with the sign of label step. For example,
           label-start < label-end enforces label-step > 0
```

```
         label-start > label-end enforces label-step < 0";
      choice technology {
        default generic;
        description
          "Data plane technology type.";
        case generic {
          leaf generic {
            type int32;
            default 1;
            description "Label range step";
          }
        }
      }
    }
    leaf range-bitmap {
      type yang:hex-string;
      description
        "When there are gaps between label-start and label-end,
         this attribute is used to specify the positions
         of the used labels. This is represented in big-endian as
         hex-string.

         Each bit-position in the range-bitmap hex-string maps to a
         label in the range derived from the label-start.

         For example, assuming label-start=16000 and
         range-bitmap=0x01000001, then:
          - bit-position(0) is set, and the corresponding mapped label
            from the range is: 16000 + (0 * label-step) or
            16000 for default label-step=1.
          - bit-position(24) is set, and the corresponding mapped label
            from the range is: 16000 + (24 * label-step) or
            16024 for default label-step=1";
    }
  }

  grouping label-set-info {
    description
      "Grouping for List of label restrictions specifying what labels
       may or may not be used on a link connectivity.";
    container label-restrictions {
      description
        "The label restrictions container";
      list label-restriction {
        key "index";
        description
          "The absence of label-set implies that all labels are
           acceptable; otherwise only restricted labels are
```

```
          available.";
        reference
          "RFC7579: General Network Element Constraint Encoding
          for GMPLS-Controlled Networks";
        uses label-restriction-info;
      }
    }
  }

  grouping optimization-metric-entry {
    description "Optimization metrics configuration grouping";
    leaf metric-type {
      type identityref {
        base path-metric-type;
      }
      description
        "Identifies an entry in the list of metric-types to
         optimize the TE path for.";
    }
    leaf weight {
      type uint8;
      default 1;
      description "TE path metric normalization weight";
    }
    container explicit-route-exclude-objects {
      when "../metric-type = " +
           "'te-types:path-metric-optimize-excludes'";
      description
        "Container for the exclude route object list";
      uses path-route-exclude-objects;
    }
    container explicit-route-include-objects {
      when "../metric-type = " +
           "'te-types:path-metric-optimize-includes'";
      description
        "Container for the include route object list";
      uses path-route-include-objects;
    }
  }

  grouping common-constraints {
    description
      "Common constraints grouping that can be set on
       a constraint set or directly on the tunnel";

    uses te-bandwidth {
      description
        "A requested bandwidth to use for path computation";
```

```
    }

    leaf link-protection {
      type identityref {
        base link-protection-type;
      }
      default te-types:link-protection-unprotected;
      description
        "Link protection type required for the links included
         in the computed path";
      reference
        "RFC4202: Routing Extensions in Support of
         Generalized Multi-Protocol Label Switching (GMPLS).";
    }

    leaf setup-priority {
      type uint8 {
        range "0..7";
      }
      default 7;
      description
        "TE LSP requested setup priority";
      reference "RFC3209";
    }
    leaf hold-priority {
      type uint8 {
        range "0..7";
      }
      default 7;
      description
        "TE LSP requested hold priority";
      reference "RFC3209";
    }
    leaf signaling-type {
      type identityref {
        base path-signaling-type;
      }
      default te-types:path-setup-rsvp;
      description "TE tunnel path signaling type";
    }
  }

  grouping tunnel-constraints {
    description
      "Tunnel constraints grouping that can be set on
       a constraint set or directly on the tunnel";
    uses te-topology-identifier;
    uses common-constraints;
```

```
  }

  grouping path-constraints-route-objects {
    description
      "List of route entries to be included or excluded when performing
       path computation.";
    container explicit-route-objects-always {
      description
        "Container for the exclude route object list";
      list route-object-exclude-always {
        key index;
        ordered-by user;
        description
          "List of route objects to always exclude
           from path computation";
        leaf index {
          type uint32;
          description
            "Explicit route object index. The index is used to
             identify an entry in the list. The order of entries
             is defined by the user without relying on key values";
        }
        uses explicit-route-hop;
      }
      list route-object-include-exclude {
        key index;
        ordered-by user;
        description
          "List of route objects to include or exclude in path
           computation";
        leaf explicit-route-usage {
          type identityref {
            base route-usage-type;
          }
          default te-types:route-include-object;
          description
            "Include or exclude usage. Default is to include";
        }
        leaf index {
          type uint32;
          description
            "Route object include-exclude index. The index is used to
             identify an entry in the list. The order of entries
             is defined by the user without relying on key values";
        }
        uses 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";
          }
        }
      }
    }
  }

  grouping path-route-include-objects {
    description
      "List of route object(s) to be included when performing
       the path computation.";
    list route-object-include-object {
      key index;
      ordered-by user;
      description
        "List of explicit route objects to be included
         in path computation";
      leaf index {
        type uint32;
        description
          "Route object entry index. The index is used to
           identify an entry in the list. The order of entries
           is defined by the user without relying on key values";
      }
      uses explicit-route-hop;
    }
  }

  grouping path-route-exclude-objects {
    description
      "List of route object(s) to be excluded when performing
       the path computation.";
    list route-object-exclude-object {
      key index;
      ordered-by user;
      description
        "List of explicit route objects to be excluded
```

```
         in path computation";
      leaf index {
        type uint32;
        description
          "Route object entry index. The index is used to
           identify an entry in the list. The order of entries
           is defined by the user without relying on key values";
      }
      uses 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";
        }
      }
    }
  }

  grouping generic-path-metric-bounds {
    description "TE path metric bounds grouping";
    container path-metric-bounds {
      description "TE path metric bounds container";
      list path-metric-bound {
        key metric-type;
        description "List of TE path metric bounds";
        leaf metric-type {
          type identityref {
            base path-metric-type;
          }
          description
            "Identifies an entry in the list of metric-types
             bound for the TE path.";
        }
        leaf upper-bound {
          type uint64;
          default 0;
          description
            "Upper bound on end-to-end TE path metric. A zero indicate
             an unbounded upper limit for the specific metric-type";
```

```
        }
      }
    }
  }

  grouping generic-path-optimization {
    description "TE generic path optimization grouping";

    container optimizations {
      description
        "The objective function container that includes
         attributes to impose when computing a TE path";

      choice algorithm {
        description "Optimizations algorithm.";
        case metric {
          if-feature path-optimization-metric;
          /* Optimize by metric */
          list optimization-metric {
            key "metric-type";
            description "TE path metric type";
            uses optimization-metric-entry;
          }
          /* Tiebreakers */
          container tiebreakers {
            description
              "Container for the list of tiebreaker(s)";
            list tiebreaker {
              key "tiebreaker-type";
              description
                "The list of tiebreaker criterion to apply
                 on an equally favored set of paths to pick best";
              leaf tiebreaker-type {
                type identityref {
                  base path-metric-type;
                }
                description
                  "Identifies an entry in the list of tiebreakers.";
              }
            }
          }
        }
        case objective-function {
          if-feature path-optimization-objective-function;
          /* Objective functions */
          container objective-function {
            description
              "The objective function container that includes
```

```
              attributes to impose when computing a TE path";
            leaf objective-function-type {
              type identityref {
                base objective-function-type;
              }
              default te-types:of-minimize-cost-path;
              description "Objective function entry";
            }
          }
        }
      }
    }
  }

  grouping generic-path-affinities {
    description
      "Path affinities grouping";
    container path-affinities-values {
      description
        "Path affinities values representation";
      list path-affinities-value {
        key "usage";
        description
          "List of named affinity constraints";
        leaf usage {
          type identityref {
            base resource-affinities-type;
          }
          description
            "Identifies an entry in the list of value affinities
             constraints";
        }
        leaf value {
          type admin-groups;
          default '';
          description "The affinity value. The default is empty.";
        }
      }
    }
    container path-affinity-names {
      description
        "Path affinities named representation style";
      list path-affinity-name {
        key "usage";
        description "List of named affinity constraints";
        leaf usage {
          type identityref {
            base resource-affinities-type;
```

```
          }
          description
            "Identifies an entry in the list of named affinities
             constraints";
        }
        list affinity-name {
          key "name";
          leaf name {
            type string;
            description "Identify a named affinity entry.";
          }
          description "List of named affinities";
        }
      }
    }
  }

  grouping generic-path-srlgs {
    description
      "Path SRLG grouping";
    container path-srlgs-lists {
      description
        "Path SRLG properties container";
      list path-srlgs-list {
        key "usage";
        description
          "List entries of value SRLGs to be included or excluded";
        leaf usage {
          type identityref {
            base route-usage-type;
          }
          description
            "Identifies an entry of list of SRLGs to either include
             or exclude";
        }
        leaf-list values {
          type srlg;
          description "List of SRLG values";
        }
      }
    }
    container path-srlgs-names {
      description "Container for named SRLG list";
      list path-srlgs-name {
        key "usage";
        description
          "List entries of named SRLGs to be included or excluded";
        leaf usage {
```

```
          type identityref {
            base route-usage-type;
          }
          description
            "Identifies an entry of list of named SRLGs to either
             include or exclude";
        }
        leaf-list names {
          type string;
          description "List named SRLGs";
        }
      }
    }
  }

  grouping generic-path-disjointness {
    description "Path disjointness grouping";
    leaf disjointness {
      type te-path-disjointness;
      description
        "The type of resource disjointness.
         Under primary path, disjointness level applies to
         all secondary LSPs. Under secondary, disjointness
         level overrides the one under primary";
    }
  }

  grouping common-path-constraints-attributes {
    description
      "Common path constraints configuration grouping";
    uses common-constraints;
    uses generic-path-metric-bounds;
    uses generic-path-affinities;
    uses generic-path-srlgs;
  }

  grouping generic-path-constraints {
    description
      "Global named path constraints configuration
      grouping";
    container path-constraints {
      description "TE named path constraints container";
      uses common-path-constraints-attributes;
      uses generic-path-disjointness;
    }
  }

  grouping generic-path-properties {
```

```
    description "TE generic path properties grouping";
    container path-properties {
      config false;
      description "The TE path properties";
      list path-metric {
        key metric-type;
        description "TE path metric type";
        leaf metric-type {
          type identityref {
            base path-metric-type;
          }
          description "TE path metric type";
        }
        leaf accumulative-value {
          type uint64;
          description "TE path metric accumulative value";
        }
      }
      uses generic-path-affinities;
      uses generic-path-srlgs;
      container path-route-objects {
        description
          "Container for the list of route objects either returned by
           the computation engine or actually used by an LSP";
        list path-route-object {
          key index;
          ordered-by user;
          description
            "List of route objects either returned by the computation
             engine or actually used by an LSP";
          leaf index {
            type uint32;
            description
              "Route object entry index. The index is used to
               identify an entry in the list. The order of entries
               is defined by the user without relying on key values";
          }
          uses explicit-route-hop;
        }
      }
    }
  }
}
<CODE ENDS>
```

Figure 1: TE basic types YANG module

5. Packet TE Types YANG Module

The ietf-te-packet-types module imports from the following modules:

o ietf-te-types defined in this document.

```
<CODE BEGINS> file "ietf-te-packet-types@2019-07-05.yang"
module ietf-te-packet-types {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-te-packet-types";

  /* Replace with IANA when assigned */
  prefix "te-packet-types";

  /* Import TE generic types */
  import ietf-te-types {
    prefix te-types;
    reference
      "RFC XXXX: A YANG Data Model for Common Traffic Engineering
       Types";
  }

  organization
    "IETF TEAS Working Group";

  contact
    "WG Web:   <http://tools.ietf.org/wg/teas/>
     WG List:  <mailto:teas@ietf.org>

     WG Chair: Lou Berger
               <mailto:lberger@labn.net>

     WG Chair: Vishnu Pavan Beeram
               <mailto:vbeeram@juniper.net>

     Editor:   Tarek Saad
               <mailto:tsaad@juniper.net>

     Editor:   Rakesh Gandhi
               <mailto:rgandhi@cisco.com>

     Editor:   Vishnu Pavan Beeram
               <mailto:vbeeram@juniper.net>

     Editor:   Himanshu Shah
               <mailto:hshah@ciena.com>

     Editor:   Xufeng Liu
```

<mailto:xufeng.liu.ietf@gmail.com>

Editor: Igor Bryskin
<mailto:Igor.Bryskin@huawei.com>

Editor: Young Lee
<mailto:leeyoung@huawei.com>";

```
  description
    "This module contains a collection of generally useful MPLS TE
     specific YANG data type definitions. The model fully conforms
     to the Network Management Datastore Architecture (NMDA).

     Copyright (c) 2018 IETF Trust and the persons
     identified as authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject
     to the license terms contained in, the Simplified BSD License
     set forth in Section 4.c of the IETF Trust's Legal Provisions
     Relating to IETF Documents
     (https://trustee.ietf.org/license-info).
     This version of this YANG module is part of RFC XXXX; see
     the RFC itself for full legal notices.";

  // RFC Ed.: replace XXXX with actual RFC number and remove this
  // note.

  // RFC Ed.: update the date below with the date of RFC publication
  // and remove this note.

  revision "2019-07-05" {
    description "Latest revision of TE MPLS types";
    reference
      "RFC XXXX: A YANG Data Model for Common Traffic Engineering
       Types";
  }

  /**
   * Typedefs
   */
  typedef te-bandwidth-requested-type {
    type enumeration {
      enum specified {
        description
          "Bandwidth is explicitly specified";
      }
      enum auto {
```

```
        description
          "Bandwidth is automatically computed";
      }
    }
    description
      "enumerated type for specifying whether bandwidth is
       explicitly specified or automatically computed";
  }

  typedef te-class-type {
    type uint8;
    description
      "Diffserv-TE class-type that defines a set of Traffic
       Trunks crossing a link that is governed by a specific
       set of bandwidth constraints. CT is used for the
       purposes of link bandwidth allocation, constraint-
       based routing and admission control.";
    reference "RFC4124: Protocols for Diffserv-aware TE";
  }

  typedef bc-type {
    type uint8 {
      range "0..7";
    }
    description
      "Diffserv-TE bandwidth constraint as defined in RFC4124";
    reference "RFC4124: Protocols for Diffserv-aware TE";
  }

  typedef bandwidth-kbps {
    type uint64;
    units "Kbps";
    description
      "Bandwidth values expressed in kilobits per second";
  }

  typedef bandwidth-mbps {
    type uint64;
    units "Mbps";
    description
      "Bandwidth values expressed in megabits per second";
  }

  typedef bandwidth-gbps {
    type uint64;
    units "Gbps";
    description
      "Bandwidth values expressed in gigabits per second";
```

```
  }

  identity backup-protection-type {
    description
      "Base identity for backup protection type";
  }

  identity backup-protection-link {
    base backup-protection-type;
    description
      "backup provides link protection only";
  }

  identity backup-protection-node-link {
    base backup-protection-type;
    description
      "backup offers node (preferred) or link protection";
  }

  identity bc-model-type {
    description
      "Base identity for Diffserv-TE bandwidth constraint
      model type";
    reference "RFC4124: Protocols for Diffserv-aware TE";
  }

  identity bc-model-rdm {
    base bc-model-type;
    description
      "Russian Doll bandwidth constraint model type.";
    reference "RFC4127: Russian Dolls Model for DS-TE";
  }

  identity bc-model-mam {
    base bc-model-type;
    description
      "Maximum Allocation bandwidth constraint
      model type.";
    reference "RFC4125: Maximum Allocation Model for DS-TE";
  }

  identity bc-model-mar {
    base bc-model-type;
    description
      "Maximum Allocation with Reservation
      bandwidth constraint model type.";
    reference "RFC4126: MAR Bandwidth Constraints Model for DS-TE";
  }
```

```
   grouping performance-metrics-attributes-packet {
     description
       "A container containing performance metric attributes.";
     uses te-types:performance-metrics-attributes {
       augment performance-metrics-one-way {
         leaf one-way-min-delay {
           type uint32 {
             range 0..16777215;
           }
           description
             "One-way minimum delay or latency in micro seconds.";
         }
         leaf one-way-min-delay-normality {
           type te-types:performance-metrics-normality;
           default "normal";
           description "One-way minimum delay or latency normality.";
         }
         leaf one-way-max-delay {
           type uint32 {
             range 0..16777215;
           }
           description
             "One-way maximum delay or latency in micro seconds.";
         }
         leaf one-way-max-delay-normality {
           type te-types:performance-metrics-normality;
           default "normal";
           description "One-way maximum delay or latency normality.";
         }
         leaf one-way-delay-variation {
           type uint32 {
             range 0..16777215;
           }
           description "One-way delay variation in micro seconds.";
         }
         leaf one-way-delay-variation-normality {
           type te-types:performance-metrics-normality;
           default "normal";
           description "One-way delay variation normality.";
         }
         leaf one-way-packet-loss {
           type decimal64 {
             fraction-digits 6;
             range "0 .. 50.331642";
           }
           description
             "One-way packet loss as a percentage of the total traffic
              sent over a configurable interval. The finest precision is
```

```
            0.000003%.";
        }
        leaf one-way-packet-loss-normality {
          type te-types:performance-metrics-normality;
          default "normal";
          description "Packet loss normality.";
        }
        description
          "PM one-way packet specific augmentation to generic PM
           grouping";
      }
      augment performance-metrics-two-way {
        leaf two-way-min-delay {
          type uint32 {
            range 0..16777215;
          }
          default 0;
          description
            "Two-way minimum delay or latency in micro seconds.";
        }
        leaf two-way-min-delay-normality {
          type te-types:performance-metrics-normality;
          default "normal";
          description "Two-way minimum delay or latency normality.";
        }
        leaf two-way-max-delay {
          type uint32 {
            range 0..16777215;
          }
          default 0;
          description
            "Two-way maximum delay or latency in micro seconds.";
        }
        leaf two-way-max-delay-normality {
          type te-types:performance-metrics-normality;
          default "normal";
          description "Two-way maximum delay or latency normality.";
        }
        leaf two-way-delay-variation {
          type uint32 {
            range 0..16777215;
          }
          default 0;
          description "Two-way delay variation in micro seconds.";
        }
        leaf two-way-delay-variation-normality {
          type te-types:performance-metrics-normality;
          default "normal";
```

```
          description "Two-way delay variation normality.";
        }
        leaf two-way-packet-loss {
          type decimal64 {
            fraction-digits 6;
            range "0 .. 50.331642";
          }
          default 0;
          description
            "Two-way packet loss as a percentage of the total traffic
             sent over a configurable interval. The finest precision is
             0.000003%.";
        }
        leaf two-way-packet-loss-normality {
          type te-types:performance-metrics-normality;
          default "normal";
          description "Two-way packet loss normality.";
        }
        description
          "PM two-way packet specific augmentation to generic PM
           grouping";
      }
    }
  }

  grouping one-way-performance-metrics-packet {
    description
      "One-way packet performance metrics throttle grouping.";
    leaf one-way-min-delay {
      type uint32 {
        range 0..16777215;
      }
      default 0;
      description "One-way minimum delay or latency in micro seconds.";
    }
    leaf one-way-max-delay {
      type uint32 {
        range 0..16777215;
      }
      default 0;
      description "One-way maximum delay or latency in micro seconds.";
    }
    leaf one-way-delay-variation {
      type uint32 {
        range 0..16777215;
      }
      default 0;
      description "One-way delay variation in micro seconds.";
```

```
      }
      leaf one-way-packet-loss {
        type decimal64 {
          fraction-digits 6;
          range "0 .. 50.331642";
        }
        default 0;
        description
          "One-way packet loss as a percentage of the total traffic sent
           over a configurable interval. The finest precision is
           0.000003%.";
      }
    }

    grouping two-way-performance-metrics-packet {
      description
        "Two-way packet performance metrics throttle grouping.";
      leaf two-way-min-delay {
        type uint32 {
          range 0..16777215;
        }
        default 0;
        description "Two-way minimum delay or latency in micro seconds.";
      }
      leaf two-way-max-delay {
        type uint32 {
          range 0..16777215;
        }
        default 0;
        description "Two-way maximum delay or latency in micro seconds.";
      }
      leaf two-way-delay-variation {
        type uint32 {
          range 0..16777215;
        }
        default 0;
        description "Two-way delay variation in micro seconds.";
      }
      leaf two-way-packet-loss {
        type decimal64 {
          fraction-digits 6;
          range "0 .. 50.331642";
        }
        default 0;
        description
          "Two-way packet loss as a percentage of the total traffic sent
           over a configurable interval. The finest precision is
           0.000003%.";
```

```
    }
  }

  grouping performance-metrics-throttle-container-packet {
    description
      "Packet performance metrics threshold grouping";
    uses te-types:performance-metrics-throttle-container {
        augment "throttle/threshold-out" {
          uses one-way-performance-metrics-packet;
          uses two-way-performance-metrics-packet;
          description
            "PM threshold-out packet augmentation to
             generic grouping";
        }
        augment "throttle/threshold-in" {
          uses one-way-performance-metrics-packet;
          uses two-way-performance-metrics-packet;
          description
            "PM threshold-in packet augmentation to
             generic grouping";
        }
        augment "throttle/threshold-accelerated-advertisement" {
          uses one-way-performance-metrics-packet;
          uses two-way-performance-metrics-packet;
          description
            "PM accelerated advertisement packet augmentation to
             generic grouping";
        }
    }
  }
}
<CODE ENDS>
```

Figure 2: TE packet types YANG module

6. IANA Considerations

This document registers the following URIs in the IETF XML registry [RFC3688]. Following the format in [RFC3688], the following registration is requested to be made.

URI: urn:ietf:params:xml:ns:yang:ietf-te-types XML: N/A, the requested URI is an XML namespace.

URI: urn:ietf:params:xml:ns:yang:ietf-te-packet-types XML: N/A, the requested URI is an XML namespace.

This document registers two YANG modules in the YANG Module Names registry [RFC6020].

name: ietf-te-types namespace: urn:ietf:params:xml:ns:yang:ietf-te-types prefix: ietf-te-types reference: RFCXXXX

name: ietf-te-packet-types namespace: urn:ietf:params:xml:ns:yang:ietf-te-packet-types prefix: ietf-te-packet-types reference: RFCXXXX

7. Security Considerations

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

The YANG module in this document defines common TE type definitions (i.e., typedef, identity and grouping statements) in YANG data modeling language to be imported and used by other TE modules. When imported and used, the resultant schema will have data nodes that can be writable, or readable. The access to such data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations.

The security considerations spelled out in the YANG 1.1 specification [RFC7950] apply for this document as well.

8. Acknowledgement

The authors would like to thank the members of the multi-vendor YANG design team who are involved in the definition of these data types.

The authors would also like to thank Tom Petch, Jan Lindblad, Sergio Belotti, Italo Busi, Carlo Perocchio, Francesco Lazzeri, and Aihua Guo for their review comments and for providing valuable feedback on this document.

9. Contributors

Himanshu Shah
Ciena

Email: hshah@ciena.com

Young Lee
Huawei Technologies

Email: leeyoung@huawei.com

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, DOI 10.17487/RFC7951, August 2016, <https://www.rfc-editor.org/info/rfc7951>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8294] Liu, X., Qu, Y., Lindem, A., Hopps, C., and L. Berger, "Common YANG Data Types for the Routing Area", RFC 8294, DOI 10.17487/RFC8294, December 2017, <https://www.rfc-editor.org/info/rfc8294>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

10.2. Informative References

[RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, DOI 10.17487/RFC2702, September 1999, <https://www.rfc-editor.org/info/rfc2702>.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[RFC3272] Awduche, D., Chiu, A., Elwalid, A., Widjaja, I., and X. Xiao, "Overview and Principles of Internet Traffic Engineering", RFC 3272, DOI 10.17487/RFC3272, May 2002, <https://www.rfc-editor.org/info/rfc3272>.

[RFC3471] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description", RFC 3471, DOI 10.17487/RFC3471, January 2003, <https://www.rfc-editor.org/info/rfc3471>.

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, DOI 10.17487/RFC3477, January 2003, <https://www.rfc-editor.org/info/rfc3477>.

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <https://www.rfc-editor.org/info/rfc3630>.

[RFC3785] Le Faucheur, F., Uppili, R., Vedrenne, A., Merckx, P., and T. Telkamp, "Use of Interior Gateway Protocol (IGP) Metric as a second MPLS Traffic Engineering (TE) Metric", BCP 87, RFC 3785, DOI 10.17487/RFC3785, May 2004, <https://www.rfc-editor.org/info/rfc3785>.

[RFC4090] Pan, P., Ed., Swallow, G., Ed., and A. Atlas, Ed., "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, DOI 10.17487/RFC4090, May 2005, <https://www.rfc-editor.org/info/rfc4090>.

[RFC4124] Le Faucheur, F., Ed., "Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering", RFC 4124, DOI 10.17487/RFC4124, June 2005, <https://www.rfc-editor.org/info/rfc4124>.

[RFC4125] Le Faucheur, F. and W. Lai, "Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering", RFC 4125, DOI 10.17487/RFC4125, June 2005, <https://www.rfc-editor.org/info/rfc4125>.

[RFC4126] Ash, J., "Max Allocation with Reservation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering & Performance Comparisons", RFC 4126, DOI 10.17487/RFC4126, June 2005, <https://www.rfc-editor.org/info/rfc4126>.

[RFC4127] Le Faucheur, F., Ed., "Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering", RFC 4127, DOI 10.17487/RFC4127, June 2005, <https://www.rfc-editor.org/info/rfc4127>.

[RFC4202] Kompella, K., Ed. and Y. Rekhter, Ed., "Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4202, DOI 10.17487/RFC4202, October 2005, <https://www.rfc-editor.org/info/rfc4202>.

[RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, DOI 10.17487/RFC4203, October 2005, <https://www.rfc-editor.org/info/rfc4203>.

[RFC4328] Papadimitriou, D., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for G.709 Optical Transport Networks Control", RFC 4328, DOI 10.17487/RFC4328, January 2006, <https://www.rfc-editor.org/info/rfc4328>.

[RFC4427] Mannie, E., Ed. and D. Papadimitriou, Ed., "Recovery (Protection and Restoration) Terminology for Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4427, DOI 10.17487/RFC4427, March 2006, <https://www.rfc-editor.org/info/rfc4427>.

[RFC4561] Vasseur, J., Ed., Ali, Z., and S. Sivabalan, "Definition of a Record Route Object (RRO) Node-Id Sub-Object", RFC 4561, DOI 10.17487/RFC4561, June 2006, <https://www.rfc-editor.org/info/rfc4561>.

[RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <https://www.rfc-editor.org/info/rfc4657>.

[RFC4736] Vasseur, JP., Ed., Ikejiri, Y., and R. Zhang, "Reoptimization of Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Loosely Routed Label Switched Path (LSP)", RFC 4736, DOI 10.17487/RFC4736, November 2006, <https://www.rfc-editor.org/info/rfc4736>.

[RFC4872] Lang, J., Ed., Rekhter, Y., Ed., and D. Papadimitriou, Ed., "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, DOI 10.17487/RFC4872, May 2007, <https://www.rfc-editor.org/info/rfc4872>.

[RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel, "GMPLS Segment Recovery", RFC 4873, DOI 10.17487/RFC4873, May 2007, <https://www.rfc-editor.org/info/rfc4873>.

[RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <https://www.rfc-editor.org/info/rfc4875>.

[RFC4920] Farrel, A., Ed., Satyanarayana, A., Iwata, A., Fujita, N., and G. Ash, "Crankback Signaling Extensions for MPLS and GMPLS RSVP-TE", RFC 4920, DOI 10.17487/RFC4920, July 2007, <https://www.rfc-editor.org/info/rfc4920>.

[RFC5003] Metz, C., Martini, L., Balus, F., and J. Sugimoto, "Attachment Individual Identifier (AII) Types for Aggregation", RFC 5003, DOI 10.17487/RFC5003, September 2007, <https://www.rfc-editor.org/info/rfc5003>.

[RFC5150] Ayyangar, A., Kompella, K., Vasseur, JP., and A. Farrel, "Label Switched Path Stitching with Generalized Multiprotocol Label Switching Traffic Engineering (GMPLS TE)", RFC 5150, DOI 10.17487/RFC5150, February 2008, <https://www.rfc-editor.org/info/rfc5150>.

[RFC5151] Farrel, A., Ed., Ayyangar, A., and JP. Vasseur, "Inter-Domain MPLS and GMPLS Traffic Engineering -- Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 5151, DOI 10.17487/RFC5151, February 2008, <https://www.rfc-editor.org/info/rfc5151>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <https://www.rfc-editor.org/info/rfc5307>.

[RFC5420] Farrel, A., Ed., Papadimitriou, D., Vasseur, JP., and A. Ayyangarps, "Encoding of Attributes for MPLS LSP Establishment Using Resource Reservation Protocol Traffic Engineering (RSVP-TE)", RFC 5420, DOI 10.17487/RFC5420, February 2009, <https://www.rfc-editor.org/info/rfc5420>.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, DOI 10.17487/RFC5541, June 2009, <https://www.rfc-editor.org/info/rfc5541>.

[RFC5712] Meyer, M., Ed. and JP. Vasseur, Ed., "MPLS Traffic Engineering Soft Preemption", RFC 5712, DOI 10.17487/RFC5712, January 2010, <https://www.rfc-editor.org/info/rfc5712>.

[RFC5817] Ali, Z., Vasseur, JP., Zamfir, A., and J. Newton, "Graceful Shutdown in MPLS and Generalized MPLS Traffic Engineering Networks", RFC 5817, DOI 10.17487/RFC5817, April 2010, <https://www.rfc-editor.org/info/rfc5817>.

[RFC6001] Papadimitriou, D., Vigoureux, M., Shiomoto, K., Brungard, D., and JL. Le Roux, "Generalized MPLS (GMPLS) Protocol Extensions for Multi-Layer and Multi-Region Networks (MLN/MRN)", RFC 6001, DOI 10.17487/RFC6001, October 2010, <https://www.rfc-editor.org/info/rfc6001>.

[RFC6004] Berger, L. and D. Fedyk, "Generalized MPLS (GMPLS) Support for Metro Ethernet Forum and G.8011 Ethernet Service Switching", RFC 6004, DOI 10.17487/RFC6004, October 2010, <https://www.rfc-editor.org/info/rfc6004>.

[RFC6119] Harrison, J., Berger, J., and M. Bartlett, "IPv6 Traffic Engineering in IS-IS", RFC 6119, DOI 10.17487/RFC6119, February 2011, <https://www.rfc-editor.org/info/rfc6119>.

[RFC6205] Otani, T., Ed. and D. Li, Ed., "Generalized Labels for Lambda-Switch-Capable (LSC) Label Switching Routers", RFC 6205, DOI 10.17487/RFC6205, March 2011, <https://www.rfc-editor.org/info/rfc6205>.

[RFC6370] Bocci, M., Swallow, G., and E. Gray, "MPLS Transport Profile (MPLS-TP) Identifiers", RFC 6370, DOI 10.17487/RFC6370, September 2011, <https://www.rfc-editor.org/info/rfc6370>.

[RFC6378] Weingarten, Y., Ed., Bryant, S., Osborne, E., Sprecher, N., and A. Fulignoli, Ed., "MPLS Transport Profile (MPLS-TP) Linear Protection", RFC 6378, DOI 10.17487/RFC6378, October 2011, <https://www.rfc-editor.org/info/rfc6378>.

[RFC6511] Ali, Z., Swallow, G., and R. Aggarwal, "Non-Penultimate Hop Popping Behavior and Out-of-Band Mapping for RSVP-TE Label Switched Paths", RFC 6511, DOI 10.17487/RFC6511, February 2012, <https://www.rfc-editor.org/info/rfc6511>.

[RFC6780] Berger, L., Le Faucheur, F., and A. Narayanan, "RSVP ASSOCIATION Object Extensions", RFC 6780, DOI 10.17487/RFC6780, October 2012, <https://www.rfc-editor.org/info/rfc6780>.

[RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <https://www.rfc-editor.org/info/rfc6790>.

[RFC6827] Malis, A., Ed., Lindem, A., Ed., and D. Papadimitriou, Ed., "Automatically Switched Optical Network (ASON) Routing for OSPFv2 Protocols", RFC 6827, DOI 10.17487/RFC6827, January 2013, <https://www.rfc-editor.org/info/rfc6827>.

[RFC7139] Zhang, F., Ed., Zhang, G., Belotti, S., Ceccarelli, D., and K. Pithewan, "GMPLS Signaling Extensions for Control of Evolving G.709 Optical Transport Networks", RFC 7139, DOI 10.17487/RFC7139, March 2014, <https://www.rfc-editor.org/info/rfc7139>.

[RFC7260] Takacs, A., Fedyk, D., and J. He, "GMPLS RSVP-TE Extensions for Operations, Administration, and Maintenance (OAM) Configuration", RFC 7260, DOI 10.17487/RFC7260, June 2014, <https://www.rfc-editor.org/info/rfc7260>.

[RFC7308] Osborne, E., "Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)", RFC 7308, DOI 10.17487/RFC7308, July 2014, <https://www.rfc-editor.org/info/rfc7308>.

[RFC7471] Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", RFC 7471, DOI 10.17487/RFC7471, March 2015, <https://www.rfc-editor.org/info/rfc7471>.

[RFC7551] Zhang, F., Ed., Jing, R., and R. Gandhi, Ed., "RSVP-TE Extensions for Associated Bidirectional Label Switched Paths (LSPs)", RFC 7551, DOI 10.17487/RFC7551, May 2015, <https://www.rfc-editor.org/info/rfc7551>.

[RFC7570] Margaria, C., Ed., Martinelli, G., Balls, S., and B. Wright, "Label Switched Path (LSP) Attribute in the Explicit Route Object (ERO)", RFC 7570, DOI 10.17487/RFC7570, July 2015, <https://www.rfc-editor.org/info/rfc7570>.

[RFC7571] Dong, J., Chen, M., Li, Z., and D. Ceccarelli, "GMPLS RSVP-TE Extensions for Lock Instruct and Loopback", RFC 7571, DOI 10.17487/RFC7571, July 2015, <https://www.rfc-editor.org/info/rfc7571>.

[RFC7579] Bernstein, G., Ed., Lee, Y., Ed., Li, D., Imajuku, W., and J. Han, "General Network Element Constraint Encoding for GMPLS-Controlled Networks", RFC 7579, DOI 10.17487/RFC7579, June 2015, <https://www.rfc-editor.org/info/rfc7579>.

[RFC7823] Atlas, A., Drake, J., Giacalone, S., and S. Previdi, "Performance-Based Path Selection for Explicitly Routed Label Switched Paths (LSPs) Using TE Metric Extensions", RFC 7823, DOI 10.17487/RFC7823, May 2016, <https://www.rfc-editor.org/info/rfc7823>.

[RFC8001] Zhang, F., Ed., Gonzalez de Dios, O., Ed., Margaria, C., Hartley, M., and Z. Ali, "RSVP-TE Extensions for Collecting Shared Risk Link Group (SRLG) Information", RFC 8001, DOI 10.17487/RFC8001, January 2017, <https://www.rfc-editor.org/info/rfc8001>.

[RFC8149] Saad, T., Ed., Gandhi, R., Ed., Ali, Z., Venator, R., and Y. Kamite, "RSVP Extensions for Reoptimization of Loosely Routed Point-to-Multipoint Traffic Engineering Label Switched Paths (LSPs)", RFC 8149, DOI 10.17487/RFC8149, April 2017, <https://www.rfc-editor.org/info/rfc8149>.

[RFC8169] Mirsky, G., Ruffini, S., Gray, E., Drake, J., Bryant, S., and A. Vainshtein, "Residence Time Measurement in MPLS Networks", RFC 8169, DOI 10.17487/RFC8169, May 2017, <https://www.rfc-editor.org/info/rfc8169>.

[RFC8570] Ginsberg, L., Ed., Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 8570, DOI 10.17487/RFC8570, March 2019, <https://www.rfc-editor.org/info/rfc8570>.

Authors' Addresses

Tarek Saad
Juniper Networks

Email: tsaad@juniper.net

Rakesh Gandhi
Cisco Systems Inc

Email: rgandhi@cisco.com

Xufeng Liu
Volta Networks

Email: xufeng.liu.ietf@gmail.com

Vishnu Pavan Beeram
Juniper Networks

Email: vbeeram@juniper.net

Igor Bryskin
Huawei Technologies

Email: Igor.Bryskin@huawei.com