

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 2, 2018

L. Zhang
L. Zheng
Huawei Technologies
S. Aldrin
Google
G. Mirsky
ZTE Corp.
October 29, 2017

YANG Data Model for MPLS-TP Operations, Administration, and Maintenance (OAM)

[draft-zhang-mpls-tp-yang-oam-05](#)

Abstract

The Transport Profile of Multiprotocol Label Switching (MPLS-TP), specified in [RFC 5921](#), is a packet-based transport technology based on the MPLS Traffic Engineering (MPLS-TE) and pseudowire (PW) data-plane architectures. A comprehensive set of Operations, Administration, and Maintenance (OAM) procedures that fulfill the MPLS-TP OAM requirements for fault, performance, and protection-switching management had been defined. YANG, defined in [RFC 6020](#) and [RFC 7950](#), is a data model definition language that was introduced to define the content of a conceptual data stores that allows networked devices to be managed using NETCONF, as specified in [RFC 6241](#). This document presents the YANG Data model for MPLS-TP OAM, including the basic functions of Fault Management and Performance Monitoring.

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 May 2, 2018.

Internet-Draft

MPLS-TP OAM YANG

October 2017

Copyright Notice

Copyright (c) 2017 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
2.	Conventions used in this document	3
2.1.	Terminology	3
2.2.	Requirements Language	4
3.	Design of the Data Model	4
3.1.	Maintenance Entity Group (MEG) Configuration	4
3.2.	Maintenance Entities (MEs) Configuration	4
3.3.	MPLS-TP OAM Fault Management And Performance Moinitoring Configuration	6
3.4.	Display of ME Status	7
3.5.	Display of Detect Result	10
4.	MPLS-TP OAM Data Hierarchy	12
5.	Interaction with other MPLS OAM Tools Models	17
6.	MPLS-TP OAM YANG module	17
7.	Examples	47
8.	Security Considerations	47
9.	IANA Considerations	48
10.	Acknowledgements	48
11.	References	48
11.1.	Normative References	48
11.2.	Infomative References	49
	Authors' Addresses	49

[1.](#) Introduction

The Transport Profile of Multiprotocol Label Switching (MPLS-TP)

[[RFC5921](#)] is a packet-based transport technology based on the MPLS Traffic Engineering (MPLS-TE) and pseudowire (PW) data-plane architectures. A comprehensive set of Operations, Administration, and Maintenance (OAM) procedures that fulfill the MPLS-TP OAM requirements for fault, performance, and protection-switching

management had been defined. YANG [[RFC6020](#)] is a data definition language that was introduced to define the contents of a conceptual data store that allows networked devices to be managed using NETCONF [[RFC6241](#)]. This document presents the YANG Data model for MPLS-TP OAM, including the basic functions of Fault Management and Performance Monitoring.

The rest of this document is organized as follows. [Section 2](#) presents the conventions used in this document. [Section 3](#) provides the design of the MPLS-TP OAM data model in details. [Section 4](#) presents the complete data hierarchy of LSP-Ping YANG model. [Section 5](#) discusses the interaction between MPLS-TP OAM data model and other MPLS tools data models. [Section 6](#) specifies the YANG module and [section 7](#) lists examples which conform to the YANG module specified in this document. Finally, security considerations are discussed in [Section 8](#).

[2](#). Conventions used in this document

[2.1](#). Terminology

CC - Continuity Check

CV - Conectivity Verification

LM - Loss Measurement

ME - Maintenance Entity

MEG - Maintenance Entity Group

MEP - Maintenance Entity Group End Point

MIP - Maintenance Entity Group Intermediate Point

PM - Performance Monitoring

PW - Pseudowire

DM - Delay Measurement

AIS - Alarm Indication Signal

LKR - Lock Report

[2.2.](#) Requirements Language

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.

[3.](#) Design of the Data Model

This YANG data model is defined to be used to configure and manage MPLS-TP OAM. Under the top level container `mplstp-oam` is the container `meg`, which contains the configuration and detect result information of multi instances of Maintenance Entity Group (MEG). Under `meg` container, configuration of each Maintenance Entity (ME) type are defined in corresponding list for a particular MEG. Different OAM function configuration are also defined for each MEG. The ME status and detect result information is also shown on per MEG basis. In order to facilitate zero-touch experience, this document defines a default value of the related detect parameters, such as detection intervals, the exp of OAM packet and OAM packet size.

[3.1.](#) Maintenance Entity Group (MEG) Configuration

The container "meg" holds the configuration and detect result information of multi instances of Maintenance Entity Group (MEG). Each MEG is indexed by meg-name. The data hierarchy for MEG configuration is presented below:

```

module: ietf-mplstp-oam
  +--rw mplstp-oam
    +--rw ais-enable?   enable
    +--rw meg* [meg-name]
      +--rw meg-name      string
      +--rw me-type?      me-type
      +--rw meg-id?       string
      +--rw meg-level?    uint8
      +--rw oam-active-state? active-type

```

[3.2.](#) Maintenance Entities (MEs) Configuration

Within a given Maintenance Entity Group there could be one or more type of Maintenance Entity (ME), configuration of different types of MEs are represented in its corresponding list and indexed by its own key. The data hierarchy for ME configuration is presented below:

```

module: ietf-mplstp-oam

```

```

+--rw mplstp-oam
  +--rw ais-enable?   enable
  +--rw meg* [meg-name]
    +--rw meg-name      string
    +--rw me-type?      me-type
    +--rw meg-id?       string
    +--rw meg-level?    uint8
    +--rw oam-active-state? active-type
    +--rw pw* [local-peer-ip local-vc-id local-vc-type
remote-peer-ip remote-vc-id remote-vc-type]
      | +--rw local-peer-ip    inet:ip-address
      | +--rw local-vc-id      uint32
      | +--rw local-vc-type    vc-type
      | +--rw remote-peer-ip   inet:ip-address
      | +--rw remote-vc-id     uint32
      | +--rw remote-vc-type   vc-type
      | +--rw mep-id?          uint16
      | +--rw remote-mep-id?   uint16
      | +--rw vll-ttl?         uint8
      | +--rw gal-enable?      enable
      | +--rw gal-mode?        gal-mode
    +--rw lsp* [tunnel-name tunnel-id ingress-lsr-id]

```

```

|   +---rw tunnel-name          string
|   +---rw tunnel-id            uint32
|   +---rw ingress-lsr-id       inet:ip-address
|   +---rw mep-id?              uint16
|   +---rw remote-mep-id?       uint16
|   +---rw reverse-tunnel-name  string
|   +---rw reverse-tunnel-id?   uint16
|   +---rw reverse-ingress-lsr-id? inet:ip-address
|   +---rw tunnel-description?  string
|   +---rw tunnel-type?         tunnel-type
|   +---rw tunnel-direction?    tunnel-direction-type
+---rw section* [section-id]
|   +---rw section-id          uint64
|   +---rw if-name?            string
|   +---rw peer-ip             inet:ip-address
|   +---rw peer-lsr-id?        inet:ip-address
|   +---rw mep-id?             uint16
|   +---rw remote-mep-id?      uint16
+---rw pw-spme* [local-peer-ip local-vc-id switch-peer-ip
switch-vc-id vc-type instance-name]
|   +---rw local-peer-ip       inet:ip-address
|   +---rw local-vc-id         uint32
|   +---rw switch-peer-ip      inet:ip-address
|   +---rw switch-vc-id        uint32
|   +---rw vc-type             vc-type
|   +---rw instance-name       string

```

```

+---rw lsp-spme* [tunnel-id local-lsr-id remote-lsr-id]
|   +---rw tunnel-id          uint32
|   +---rw local-lsr-id       inet:ip-address
|   +---rw remote-lsr-id      inet:ip-address

```

[3.3.](#) MPLS-TP OAM Fault Management And Performance Monitoring Configuration

Different OAM function configuration are also defined for each MEG. The data hierarchy for OAM function configuration is presented below:

```

module: ietf-mplstp-oam
  +---rw mplstp-oam
    +---rw ais-enable?  enable

```

```

+--rw meg* [meg-name]
|   +--rw meg-name          string
|   +--rw me-type?          me-type
|   +--rw meg-id?           string
|   +--rw meg-level?        uint8
|   +--rw oam-active-state?  active-type
|   +--rw pw* [local-peer-ip local-vc-id local-vc-type
remote-peer-ip remote-vc-id remote-vc-type]
|   ...
+--rw lsp* [tunnel-name tunnel-id ingress-lsr-id]
|   ...
+--rw section* [section-id]
|   |   +--rw section-id      uint64
|   |   +--rw if-name?       string
|   |   +--rw peer-ip         inet:ip-address
|   |   +--rw peer-lsr-id?    inet:ip-address
|   |   +--rw mep-id?         uint16
|   |   +--rw remote-mep-id?  uint16
|   +--rw pw-spme* [local-peer-ip local-vc-id switch-peer-ip
switch-vc-id vc-type instance-name]
|   ...
+--rw lsp-spme* [tunnel-id local-lsr-id remote-lsr-id]
|   ...
+--rw cc
|   |   +--rw cc-session-mode?      cc-session-mode
|   |   +--rw cc-authentication-enable?  enable
|   |   +--rw cc-exp?               uint8
|   |   +--rw cc-transmit-interval?    cc-interval
|   |   +--rw cc-recieve-interval?    cc-interval
|   |   +--rw cc-detect-multiplier?   cc-detect-multiplier
|   |   +--rw cc-enable?              enable
+--rw cv
|   |   +--rw cv-session-mode?        cc-session-mode

```

```

|   +--rw cv-authentication-enable?  enable
|   +--rw cv-exp?                    uint8
|   +--rw cv-interval?               cv-interval
|   +--rw cv-detect-multiplier?     cv-detect-multiplier
|   +--rw cv-enable?                 enable
+--rw ais
|   +--rw ais-exp?                   uint8
|   +--rw ais-interval?              ais-interval

```

```

+--rw lkr
|   +--rw lkr-exp?          uint8
|   +--rw lkr-interval?    lkr-interval
|   +--rw lkr-enable?      enable
+--rw one-way-dm-send
|   +--rw one-dm-send-enable?  enable
|   +--rw one-dm-interval?    dm-interval
|   +--rw one-dm-exp?        uint8
|   +--rw one-dm-packet-size? uint16
|   +--rw one-dm-pad-value?   dm-padding-value
+--rw one-way-dm-rcv
|   +--rw onr-dm-rcv-enable?    enable
|   +--rw one-dm-rcv-enable-type? one-way-rcv-type
+--rw two-way-dm-send
|   +--rw two-dm-send-enable?  enable
|   +--rw two-dm-interval?    dm-interval
|   +--rw two-dm-exp?        uint8
|   +--rw two-dm-packet-size? uint16
|   +--rw two-dm-pad-value?   dm-padding-value
|   +--rw two-dm-time-stamp?  enable
+--rw two-way-dm-rcv
|   +--rw two-dm-rcv-enable?  enable
+--rw single-lm-send
|   +--rw slm-send-enable?    enable
|   +--rw slm-interval?      lm-interval
|   +--rw slm-exp?          uint8
+--rw single-lm-rcv
|   +--rw slm-rcv-enable?    enable
+--rw dual-lm
|   +--rw dlm-enable?      enable

```

[3.4.](#) Display of ME Status

The data hierarchy for display of ME status is presented below:

```

module: ietf-mplstp-oam
+--rw mplstp-oam
|   +--rw ais-enable?  enable
|   +--rw meg* [meg-name]

```

```

+--rw meg-name          string

```



```

+--rw me-type?                me-type
+--rw meg-id?                 string
+--rw meg-level?             uint8
+--rw oam-active-state?      active-type
+--rw pw* [local-peer-ip local-vc-id local-vc-type
remote-peer-ip remote-vc-id remote-vc-type]
...
+--rw lsp* [tunnel-name tunnel-id ingress-lsr-id]
...
+--rw section* [section-id]
...
+--rw pw-spmex [local-peer-ip local-vc-id switch-peer-ip
switch-vc-id vc-type instance-name]
...
+--rw lsp-spmex [tunnel-id local-lsr-id remote-lsr-id]
...
+--rw cc
...
+--rw cv
...
+--rw ais
...
+--rw lkr
...
+--rw one-way-dm-send
...
+--rw one-way-dm-rcv
...
+--rw two-way-dm-send
...
+--rw two-way-dm-rcv
...
+--rw single-lm-send
...
+--rw single-lm-rcv
...
+--rw dual-lm
...
+--ro status-info
|   +--ro pw*
|   |   +--ro me-index?                uint32
|   |   +--ro me-direction?           me-direction
|   |   +--ro me-state?               me-state
|   |   +--ro local-state?            me-state
|   |   +--ro remote-state?           me-state
|   |   +--ro alarm-indicate?         string
|   |   +--ro local-defect-status?    defect-status-type

```

```
| | +--ro local-invalid-time?      uint32
| | +--ro local-defect-location?  string
| | +--ro local-defect-type?     defect-type
| | +--ro remote-defect-status?   defect-status-type
| | +--ro remote-invalid-time?    uint32
| | +--ro remote-defect-location? string
| | +--ro remote-defect-type?    defect-type
| +--ro lsp*
| | +--ro me-index?              uint32
| | +--ro me-direction?         me-direction
| | +--ro me-state?             me-state
| | +--ro local-state?          me-state
| | +--ro remote-state?         me-state
| | +--ro alarm-indicate?       string
| | +--ro local-defect-status?   defect-status-type
| | +--ro local-invalid-time?    uint32
| | +--ro local-defect-location? string
| | +--ro local-defect-type?    defect-type
| | +--ro remote-defect-status?   defect-status-type
| | +--ro remote-invalid-time?    uint32
| | +--ro remote-defect-location? string
| | +--ro remote-defect-type?    defect-type
| | +--ro me-index-egress?       uint32
| | +--ro me-direct-egress?     me-direction
| | +--ro status-board-egress?   string
| | +--ro state-egress?         me-state
| | +--ro alarm-egress?         string
| +--ro section*
| | +--ro me-index?              uint32
| | +--ro me-direction?         me-direction
| | +--ro me-state?             me-state
| | +--ro local-state?          me-state
| | +--ro remote-state?         me-state
| | +--ro alarm-indicate?       string
| | +--ro local-defect-status?   defect-status-type
| | +--ro local-invalid-time?    uint32
| | +--ro local-defect-location? string
| | +--ro local-defect-type?    defect-type
| | +--ro remote-defect-status?   defect-status-type
| | +--ro remote-invalid-time?    uint32
| | +--ro remote-defect-location? string
| | +--ro remote-defect-type?    defect-type
| +--ro pw-spme*
| | +--ro me-index?              uint32
| | +--ro me-direction?         me-direction
| | +--ro me-state?             me-state
```

		+-ro mip-id?	uint16
		+-ro lsp-spme*	

Internet-Draft

MPLS-TP OAM YANG

October 2017

		+-ro me-index?	uint32
		+-ro me-direction?	me-direction
		+-ro me-state?	me-state
		+-ro mip-id?	uint16

[3.5.](#) Display of Detect Result

The data hierarchy for display of detect result is presented below:

```

module: ietf-mplstp-oam
  +-rw mplstp-oam
    +-rw ais-enable?  enable
    +-rw meg* [meg-name]
      +-rw meg-name      string
      +-rw me-type?      me-type
      +-rw meg-id?       string
      +-rw meg-level?    uint8
      +-rw oam-active-state?  active-type
      +-rw pw* [local-peer-ip local-vc-id local-vc-type
remote-peer-ip remote-vc-id remote-vc-type]
      ...
      +-rw lsp* [tunnel-name tunnel-id ingress-lsr-id]
      ...
      +-rw section* [section-id]
      ...
      +-rw pw-spme* [local-peer-ip local-vc-id switch-peer-ip
switch-vc-id vc-type instance-name]
      ...
      +-rw lsp-spme* [tunnel-id local-lsr-id remote-lsr-id]
      ...
      +-rw cc
      ...
      +-rw cv
      ...
      +-rw ais
      ...
      +-rw lkr
      ...

```

```

+---rw one-way-dm-send
...
+---rw one-way-dm-rcv
...
+---rw two-way-dm-send
...
+---rw two-way-dm-rcv
...
+---rw single-lm-send

```

```

...
+---rw single-lm-rcv
...
+---rw dual-lm
...
+---ro status-info
...
+---ro detect-result
  +---ro one-way-dm-result
    | +---ro send-pkt-num?      uint32
    | +---ro rcv-pkt-num?      uint32
    | +---ro delay-min?        uint32
    | +---ro delay-max?        uint32
    | +---ro delay-avg?        uint32
    | +---ro jitter-min?       uint32
    | +---ro jitter-max?       uint32
    | +---ro jitter-avg?       uint32
    | +---ro one-way-dm-data
    |   +---ro one-way-dm-data* [index]
    |     +---ro index          uint32
    |     +---ro one-delay?      uint32
    |     +---ro one-delay-var?  uint32
    |     +---ro error-info?     error-info
  +---ro one-way-send-result
    | +---ro measure-mode?      measure-mode
    | +---ro status?            statistics-status
  +---ro two-way-dm-result
    | +---ro measure-mode?      measure-mode
    | +---ro status?            statistics-status
    | +---ro send-pkt-num?      uint32
    | +---ro rcv-pkt-num?       uint32
    | +---ro delay-min?         uint32

```

```

|   +--ro delay-max?          uint32
|   +--ro delay-avg?          uint32
|   +--ro jitter-min?         uint32
|   +--ro jitter-max?         uint32
|   +--ro jitter-avg?         uint32
|   +--ro two-way-dm-data
|       +--ro two-way-dm-data* [index]
|           +--ro index          uint32
|           +--ro two-delay?      uint32
|           +--ro two-delay-var?  uint32
|           +--ro error-info?     error-info
+--ro single-lm-result
|   +--ro measure-mode?        measure-mode
|   +--ro status?              statistics-status
|   +--ro send-pkt-num?        uint32
|   +--ro rcv-pkt-num?         uint32

```

```

|   +--ro rmt-loss-ratio-min?  uint32
|   +--ro rmt-loss-ratio-max?  uint32
|   +--ro rmt-loss-atio-avg?   uint32
|   +--ro rmt-loss-count-min?  uint32
|   +--ro rmt-loss-count-max?  uint32
|   +--ro rmt-loss-count-avg?  uint32
|   +--ro single-lm-data
|       +--ro single-lm-data* [index]
|           +--ro index          uint32
|           +--ro slm-loss-lcl?   uint32
|           +--ro slm-loss-lcl-rat? string
|           +--ro slm-loss-rmt?   uint32
|           +--ro slm-loss-rmt-rat? string
|           +--ro error-info?     error-info
+--ro dual-lm-data
|   +--ro dual-lm-data* [index]
|       +--ro index          uint32
|       +--ro dlm-loss-lcl?   uint32
|       +--ro dlm-loss-lcl-rat? string
|       +--ro dlm-loss-rmt?   uint32
|       +--ro dlm-loss-rmt-rat? string
|       +--ro error-info?     error-info

```

[4.](#) MPLS-TP OAM Data Hierarchy

The complete data hierarchy related to the MPLS-TP OAM YANG model is presented below.

```

module: ietf-mplstp-oam
  +--rw mplstp-oam
    +--rw ais-enable?    enable
    +--rw meg* [meg-name]
      +--rw meg-name      string
      +--rw me-type?      me-type
      +--rw meg-id?       string
      +--rw meg-level?    uint8
      +--rw oam-active-state? active-type
      +--rw pw* [local-peer-ip local-vc-id local-vc-type
remote-peer-ip remote-vc-id remote-vc-type]
        | +--rw local-peer-ip    inet:ip-address
        | +--rw local-vc-id      uint32
        | +--rw local-vc-type    vc-type
        | +--rw remote-peer-ip   inet:ip-address
        | +--rw remote-vc-id     uint32
        | +--rw remote-vc-type   vc-type
        | +--rw mep-id?         uint16
        | +--rw remote-mep-id?   uint16
        | +--rw vll-ttl?        uint8

```

```

  | +--rw gal-enable?    enable
  | +--rw gal-mode?      gal-mode
+--rw lsp* [tunnel-name tunnel-id ingress-lsr-id]
  | +--rw tunnel-name    string
  | +--rw tunnel-id      uint32
  | +--rw ingress-lsr-id inet:ip-address
  | +--rw mep-id?        uint16
  | +--rw remote-mep-id? uint16
  | +--rw reverse-tunnel-name string
  | +--rw reverse-tunnel-id? uint16
  | +--rw reverse-ingress-lsr-id? inet:ip-address
  | +--rw tunnel-description? string
  | +--rw tunnel-type?    tunnel-type
  | +--rw tunnel-direction? tunnel-direction-type
+--rw section* [section-id]
  | +--rw section-id      uint64
  | +--rw if-name?        string
  | +--rw peer-ip         inet:ip-address

```

```

|   +---rw peer-lsr-id?      inet:ip-address
|   +---rw mep-id?          uint16
|   +---rw remote-mep-id?   uint16
+---rw pw-spmex [local-peer-ip local-vc-id switch-peer-ip
switch-vc-id vc-type instance-name]
|   +---rw local-peer-ip     inet:ip-address
|   +---rw local-vc-id       uint32
|   +---rw switch-peer-ip    inet:ip-address
|   +---rw switch-vc-id      uint32
|   +---rw vc-type           vc-type
|   +---rw instance-name     string
+---rw lsp-spmex [tunnel-id local-lsr-id remote-lsr-id]
|   +---rw tunnel-id         uint32
|   +---rw local-lsr-id      inet:ip-address
|   +---rw remote-lsr-id     inet:ip-address
+---rw cc
|   +---rw cc-session-mode?   cc-session-mode
|   +---rw cc-authentication-enable? enable
|   +---rw cc-exp?            uint8
|   +---rw cc-transmit-interval? cc-interval
|   +---rw cc-recieve-interval? cc-interval
|   +---rw cc-detect-multiplier? cc-detect-multiplier
|   +---rw cc-enable?         enable
+---rw cv
|   +---rw cv-session-mode?   cc-session-mode
|   +---rw cv-authentication-enable? enable
|   +---rw cv-exp?            uint8
|   +---rw cv-interval?       cv-interval
|   +---rw cv-detect-multiplier? cv-detect-multiplier
|   +---rw cv-enable?         enable

```

```

+---rw ais
|   +---rw ais-exp?          uint8
|   +---rw ais-interval?     ais-interval
+---rw lkr
|   +---rw lkr-exp?          uint8
|   +---rw lkr-interval?     lkr-interval
|   +---rw lkr-enable?       enable
+---rw one-way-dm-send
|   +---rw one-dm-send-enable? enable
|   +---rw one-dm-interval?   dm-interval
|   +---rw one-dm-exp?        uint8

```

```

|   +---rw one-dm-packet-size?    uint16
|   +---rw one-dm-pad-value?      dm-padding-value
+---rw one-way-dm-rcv
|   +---rw onr-dm-rcv-enable?     enable
|   +---rw one-dm-rcv-enable-type? one-way-rcv-type
+---rw two-way-dm-send
|   +---rw two-dm-send-enable?    enable
|   +---rw two-dm-interval?       dm-interval
|   +---rw two-dm-exp?            uint8
|   +---rw two-dm-packet-size?    uint16
|   +---rw two-dm-pad-value?      dm-padding-value
|   +---rw two-dm-time-stamp?     enable
+---rw two-way-dm-rcv
|   +---rw two-dm-rcv-enable?    enable
+---rw single-lm-send
|   +---rw slm-send-enable?       enable
|   +---rw slm-interval?          lm-interval
|   +---rw slm-exp?               uint8
+---rw single-lm-rcv
|   +---rw slm-rcv-enable?        enable
+---rw dual-lm
|   +---rw dlm-enable?            enable
+---ro status-info
|   +---ro pw*
|   |   +---ro me-index?           uint32
|   |   +---ro me-direction?       me-direction
|   |   +---ro me-state?           me-state
|   |   +---ro local-state?        me-state
|   |   +---ro remote-state?       me-state
|   |   +---ro alarm-indicate?     string
|   |   +---ro local-defect-status? defect-status-type
|   |   +---ro local-invalid-time? uint32
|   |   +---ro local-defect-location? string
|   |   +---ro local-defect-type?   defect-type
|   |   +---ro remote-defect-status? defect-status-type
|   |   +---ro remote-invalid-time? uint32
|   |   +---ro remote-defect-location? string

```

```

|   |   +---ro remote-defect-type?   defect-type
|   +---ro lsp*
|   |   +---ro me-index?              uint32
|   |   +---ro me-direction?          me-direction

```


		+++ro me-state?	me-state
		+++ro local-state?	me-state
		+++ro remote-state?	me-state
		+++ro alarm-indicate?	string
		+++ro local-defect-status?	defect-status-type
		+++ro local-invalid-time?	uint32
		+++ro local-defect-location?	string
		+++ro local-defect-type?	defect-type
		+++ro remote-defect-status?	defect-status-type
		+++ro remote-invalid-time?	uint32
		+++ro remote-defect-location?	string
		+++ro remote-defect-type?	defect-type
		+++ro me-index-egress?	uint32
		+++ro me-direct-egress?	me-direction
		+++ro status-board-egress?	string
		+++ro state-egress?	me-state
		+++ro alarm-egress?	string
		+++ro section*	
		+++ro me-index?	uint32
		+++ro me-direction?	me-direction
		+++ro me-state?	me-state
		+++ro local-state?	me-state
		+++ro remote-state?	me-state
		+++ro alarm-indicate?	string
		+++ro local-defect-status?	defect-status-type
		+++ro local-invalid-time?	uint32
		+++ro local-defect-location?	string
		+++ro local-defect-type?	defect-type
		+++ro remote-defect-status?	defect-status-type
		+++ro remote-invalid-time?	uint32
		+++ro remote-defect-location?	string
		+++ro remote-defect-type?	defect-type
		+++ro pw-spme*	
		+++ro me-index?	uint32
		+++ro me-direction?	me-direction
		+++ro me-state?	me-state
		+++ro mip-id?	uint16
		+++ro lsp-spme*	
		+++ro me-index?	uint32
		+++ro me-direction?	me-direction
		+++ro me-state?	me-state
		+++ro mip-id?	uint16
		+++ro detect-result	
		+++ro one-way-dm-result	

```

| +--ro send-pkt-num?      uint32
| +--ro recv-pkt-num?      uint32
| +--ro delay-min?         uint32
| +--ro delay-max?         uint32
| +--ro delay-avg?         uint32
| +--ro jitter-min?        uint32
| +--ro jitter-max?        uint32
| +--ro jitter-avg?        uint32
| +--ro one-way-dm-data
|   +--ro one-way-dm-data* [index]
|     +--ro index          uint32
|     +--ro one-delay?     uint32
|     +--ro one-delay-var? uint32
|     +--ro error-info?    error-info
+--ro one-way-send-result
| +--ro measure-mode?      measure-mode
| +--ro status?            statistics-status
+--ro two-way-dm-result
| +--ro measure-mode?      measure-mode
| +--ro status?            statistics-status
| +--ro send-pkt-num?      uint32
| +--ro rcv-pkt-num?       uint32
| +--ro delay-min?         uint32
| +--ro delay-max?         uint32
| +--ro delay-avg?         uint32
| +--ro jitter-min?        uint32
| +--ro jitter-max?        uint32
| +--ro jitter-avg?        uint32
| +--ro two-way-dm-data
|   +--ro two-way-dm-data* [index]
|     +--ro index          uint32
|     +--ro two-delay?     uint32
|     +--ro two-delay-var? uint32
|     +--ro error-info?    error-info
+--ro single-lm-result
| +--ro measure-mode?      measure-mode
| +--ro status?            statistics-status
| +--ro send-pkt-num?      uint32
| +--ro rcv-pkt-num?       uint32
| +--ro rmt-loss-ratio-min? uint32
| +--ro rmt-loss-ratio-max? uint32
| +--ro rmt-loss-atio-avg?  uint32
| +--ro rmt-loss-count-min? uint32
| +--ro rmt-loss-count-max? uint32
| +--ro rmt-loss-count-avg? uint32
| +--ro single-lm-data
|   +--ro single-lm-data* [index]

```

	++-ro index	uint32
--	-------------	--------

Internet-Draft

MPLS-TP OAM YANG

October 2017

	++-ro slm-loss-lcl?	uint32
	++-ro slm-loss-lcl-rat?	string
	++-ro slm-loss-rmt?	uint32
	++-ro slm-loss-rmt-rat?	string
	++-ro error-info?	error-info
++-ro dual-lm-data		
++-ro dual-lm-data* [index]		
	++-ro index	uint32
	++-ro dlm-loss-lcl?	uint32
	++-ro dlm-loss-lcl-rat?	string
	++-ro dlm-loss-rmt?	uint32
	++-ro dlm-loss-rmt-rat?	string
	++-ro error-info?	error-info

5. Interaction with other MPLS OAM Tools Models

TBA.

6. MPLS-TP OAM YANG module

```
<CODE BEGINS> file "ietf-mplstpoam@2017-10-29.yang"
module ietf-mplstpoam {
  namespace "urn:ietf:params:xml:ns:yang:ietf-mplstpoam";
  //namespace need to be assigned by IANA
  prefix "mplstpoam";
  import ietf-inet-types {
    prefix inet;
  }
  organization "IETF MPLS Working Group";
  contact "draft-zhang-mpls-tp-yang-oam";
  description "MPLS TP OAM YANG Module";
  revision "2017-10-29" {
    description "05 revision";
    reference "draft-zhang-mpls-tp-yang-oam";
  }

  typedef enable {
    type boolean;
    description "enable";
```

```

}
typedef me-type {
  type enumeration {
    enum "none" {
      value 0;
      description "ME type is not assigned.";
    }
    enum "section" {

```

```

    value 1;
    description "ME type is MPLS-TP Sections (between MPLS
    LSRs).";
  }
  enum "lsp" {
    value 2;
    description "ME type is an end-to-end LSP (between LERs).";
  }
  enum "pw" {
    value 3;
    description "ME type is an end-to-end Single-Segment
    Pseudowire (SS-PW) or MS-PW (between T-PEs).";
  }
  enum "lsp-spme" {
    value 4;
    description "ME type is an SPME (between a given pair
    of LERs and/or LSRs along an LSP).";
  }
  enum "pw-spme" {
    value 5;
    description "ME type is an SPME (between a given pair
    of T-PEs and/or S-PEs along an (MS-)PW).";
  }
}
description "ME type";
}
typedef cc-session-mode {
  type enumeration {
    enum "coordinated" {
      value 0;
      description "coordinated";
    }
    enum "independent" {

```

```

        value 1;
        description "independent";
    }
}
description "CC session mode";
}
typedef cc-interval {
    type uint32{
        range "1..65535";
    }
    description "The value rang for cc packet transmit and receive
interval.";
}
typedef cv-interval {
    type uint32 {

```

```

        range "1..65535";
    }
    description "The value rang for cv packet transmit interval.";
}
typedef cc-detect-multiplier {
    type uint8{
        range "1..255";
    }
    description "The value rang for cv packet detect multiplier";
}
typedef cv-detect-multiplier {
    type uint8{
        range "1..255";
    }
    description "The value rang for cv packet detect multiplier";
}
typedef lkr-interval {
    type enumeration {
        enum "interval1000ms" {
            value 0;
            description "1000ms";
        }
        enum "interval60000ms" {
            value 1;
            description "60000ms";
        }
    }
}

```

```

    }
    description "lkr-interval";
}
typedef ais-interval {
    type enumeration {
        enum "interval1000ms" {
            value 0;
            description "1000ms";
        }
        enum "interval60000ms" {
            value 1;
            description "60000ms";
        }
    }
    description "ais-interval";
}
typedef me-direction {
    type enumeration {
        enum "ingress" {
            value 0;
            description "The direction to the ME is ingress";
        }
    }
}

```

```

        enum "egress" {
            value 1;
            description "The direction to the ME is egress";
        }
        enum "dual" {
            value 2;
            description "The direction to the ME is dual";
        }
        enum "none" {
            value 3;
            description "The direction to the ME is none";
        }
    }
    description "me-direction";
}
typedef me-state {
    type enumeration {
        enum "init" {
            value 0;
        }
    }
}

```

```

        description "The me state is init";
    }
    enum "down" {
        value 1;
        description "The me state is down";
    }
    enum "up" {
        value 2;
        description "The me state is up";
    }
}
description "me-state";
}
typedef dm-interval {
    type uint32 {
        range "1..65535";
    }
    description "The value rang for dm packet transmit interval";
}
typedef dm-padding-value {
    type enumeration {
        enum "paddingvalue0" {
            value 0;
            description "0";
        }
        enum "paddingvalue1" {
            value 1;
            description "1";
        }
    }
}

```

```

    }
    description "dm-padding-value";
}
typedef lm-interval {
    type uint32 {
        range "1..65535";
    }
    description "The value rang for lm packet transmit interval";
}
typedef measure-mode {
    type enumeration {
        enum "on-demand" {

```

```

        value 0;
        description "on-demand";
    }
    enum "proactive" {
        value 1;
        description "proactive";
    }
}
description "measure mode";
}
typedef vc-type {
    type uint16 {
        range "1..65535";
    }
    description "The namespace of the vc type of pw";
}
typedef statistics-status {
    type enumeration {
        enum "finished" {
            value 0;
            description "finished";
        }
        enum "working" {
            value 1;
            description "working";
        }
    }
}
description "statistics status";
}
typedef error-info {
    type enumeration {
        enum "valid" {
            value 0;
            description "valid";
        }
        enum "invalid-loss" {

```

```

        value 1;
        description "invalid-loss";
    }
    enum "invalid-delay" {
        value 2;

```



```

        description "invalid-delay";
    }
}
description "error-info";
}
typedef defect-status-type {
    type string {
        length "1..8191";
    }
    description "The namespace of defect status type";
}
typedef defect-type {
    type string {
        length "1..8191";
    }
    description "The namespace of defect type";
}
typedef tunnel-type {
    type enumeration {
        enum "ingress" {
            value 0;
            description "ingress";
        }
        enum "egress" {
            value 1;
            description "egress";
        }
        enum "bidirectional" {
            value 2;
            description "bidirectional";
        }
    }
    description "tunnel type";
}
typedef tunnel-direction-type {
    type enumeration {
        enum "unidirectional" {
            value 0;
            description "unidirectional";
        }
        enum "bidirectional" {
            value 1;
            description "bidirectional";
        }
    }
}

```

```

    }
}
description "tunnel direction type";
}
typedef active-type {
    type enumeration {
        enum "deactive" {
            value 0;
            description "deactive";
        }
        enum "active" {
            value 1;
            description "active";
        }
    }
}
description "active-type";
}
typedef gal-mode {
    type enumeration {
        enum "with-13" {
            value 0;
            description "Gal mode is with label 13";
        }
        enum "without-13" {
            value 1;
            description "Gal mode is without label 13";
        }
    }
}
description "gal mode";
}
typedef one-way-rcv-type {
    type enumeration {
        enum "on-demand" {
            value 0;
            description "The switch of receive eanble takes effect
on-demand one-way delay-measure";
        }
        enum "proactive" {
            value 1;
            description "The switch of receive eanble takes effect
proactive one-way delay-measure";
        }
    }
}
description "one way receive type";
}
grouping pw {
    leaf local-peer-ip {
        type inet:ip-address;
    }
}

```

```
        mandatory "true";
        description "This object indicates the peer IP address";
    }
    leaf local-vc-id {
        type uint32 {
            range "1..4294967295";
        }
        mandatory "true";
        description "This object indicates the vc ID of PW
        type ME";
    }
    leaf local-vc-type {
        type vc-type;
        mandatory "true";
        description "This object indicates the vc type of VC
        type ME";
    }
    leaf remote-peer-ip {
        type inet:ip-address;
        description "This object indicates the remote peer IP of
        PW type ME";
    }
    leaf remote-vc-id {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the remote vc ID of
        PW type ME";
    }
    leaf remote-vc-type {
        type vc-type;
        description "This object indicates the remote vc type of
        PW type ME";
    }
    description "pw";
}
grouping lsp {
    leaf tunnel-name {
        type string {
            length "0..63";
        }
        mandatory "true";
    }
}
```

```

    description "The object indicates the name of tunnel";
}
leaf tunnel-id {
    type uint32 {
        range "1..65535";
    }
}

```

```

    description "The object indicates the tunnel id";
}
leaf ingress-lsr-id {
    type inet:ip-address;
    description "The object indicates the ingress LSR-ID";
}
description "lsp";
}
grouping pw-spme {
    leaf local-peer-ip {
        type inet:ip-address;
        mandatory "true";
        description "This object indicates the peer IP address of
        PW type MIP";
    }
    leaf local-vc-id {
        type uint32 {
            range "1..4294967295";
        }
        mandatory "true";
        description "This object indicates the vc ID of PW type MIP";
    }
    leaf switch-peer-ip {
        type inet:ip-address;
        mandatory "true";
        description "This object indicates the peer IP address of
        PW switch node";
    }
    leaf switch-vc-id {
        type uint32 {
            range "1..4294967295";
        }
        mandatory "true";
        description "This object indicates the vc id of PW switch
        node";
    }
}

```

```

}
leaf vc-type {
    type vc-type;
    mandatory "true";
    description "This object indicates the vc type of PW type MIP";
}
leaf instance-name {
    type string {
        length "1..31";
    }
    mandatory "true";
    description "This object specifies the VPWS instance name";
}

```

```

    description "pw spme";
}
grouping me-detect-status {
    leaf me-index {
        type uint32 {
            range "1..65535";
        }
        description "The object indicates the index of ME";
    }
    leaf me-direction {
        type me-direction;
        description "The object indicates the direction of ME";
    }
    leaf me-state {
        type me-state;
        description "The object indicates the state of ME";
    }
    leaf local-state {
        type me-state;
        description "The object indicates the local status of ME";
    }
    leaf remote-state {
        type me-state;
        description "The object indicates the remote state of ME";
    }
    leaf alarm-indicate {
        type string {
            length "1..26";
        }
    }
}

```

```

    }
    description "The object indicates the alarm of ME";
}
leaf local-defect-status {
    type defect-status-type;
    default "init";
    description "This object indicates the local defect status";
}
leaf local-invalid-time {
    type uint32 {
        range "0..4294967295";
    }
    description "This object indicates the invalid Time of
    local detect";
}
leaf local-defect-location {
    type string {
        length "1..32";
    }
    description "This object indicates the local defect location";
}

```

```

}
leaf local-defect-type {
    type defect-type;
    description "This object indicates the local defect type";
}
leaf remote-defect-status {
    type defect-status-type;
    default "init";
    description "This object indicates the remote defect status";
}
leaf remote-invalid-time {
    type uint32 {
        range "0..4294967295";
    }
    description "This object indicates the invalid Time of
    remote detect";
}
leaf remote-defect-location {
    type string {
        length "1..32";
    }
}

```

```

        description "This object indicates the remote defect location";
    }
    leaf remote-defect-type {
        type defect-type;
        description "This object indicates the remote defect type";
    }
    description "This node indicate detect status of ME";
}
grouping gal-set {
    leaf gal-enable {
        type enable;
        default "true";
        description "This object indicates the gal flag";
    }
    leaf gal-mode {
        type gal-mode;
        description "This object indicates the gal flag";
    }
    description "This object indicates the gal set";
}

container mplstp-oam {
    description "Top level container.";
    leaf ais-enable {
        type enable;
        default "false";
        description "This object indicates the global ais flag

```

```

    of mplstp-oam";
}

list meg {
    key "meg-name";
    description "meg multi instances.";
    leaf meg-name {
        type string {
            length "1..14";
        }
        mandatory "true";
        description "The object indicates the name of MEG";
    }
    leaf me-type {

```

```

    type me-type;
    default "none";
    description "The object indicates the type of ME";
}
leaf meg-id {
    type string {
        length "1..96";
    }
    description "The object indicates the ID of MEG";
}
leaf meg-level {
    type uint8 {
        range "0..7";
    }
    default "7";
    description "The object indicates the level of MEG";
}
leaf oam-active-state {
    type active-type;
    default "deactive";
    description "This object indicates the oam active state";
}

list pw {
    key "local-peer-ip local-vc-id local-vc-type remote-peer-ip
remote-vc-id remote-vc-type";
    description "PW";
    uses pw;
    leaf mep-id {
        type uint16 {
            range "1..8191";
        }
        description "This object indicates the MEP Id of local ME";
    }
}

```

```

leaf remote-mep-id {
    type uint16 {
        range "1..8191";
    }
    description "This object indicates the MEP Id of remote
ME";
}

```



```

leaf vll-ttl {
    type uint8 {
        range "1..255";
    }
    description "This object indicates the VLL ttl of PW
type ME";
}
uses gal-set;
}

list lsp {
    key "tunnel-name tunnel-id ingress-lsr-id";
    description "LSP";
    uses lsp;
    leaf mep-id {
        type uint16 {
            range "1..8191";
        }
        description "This object indicates the MEP Id of
local ME";
    }
    leaf remote-mep-id {
        type uint16 {
            range "1..8191";
        }
        description "This object indicates the MEP Id of
remote ME";
    }
    leaf reverse-tunnel-name {
        type string {
            length "0..63";
        }
        mandatory "true";
        description "The object indicates the name of
reverse tunnel";
    }
    leaf reverse-tunnel-id {
        type uint16 {
            range "1..65535";
        }
        description "The object indicates the ingress

```

```

        reverse tunnelId";
    }
    leaf reverse-ingress-lsr-id {
        type inet:ip-address;
        description "The object indicates the ingress
reverse LSR-ID";
    }
    leaf tunnel-description {
        type string {
            length "1..32";
        }
        description "The object indicates the description
of tunnel";
    }
    leaf tunnel-type {
        type tunnel-type;
        default "ingress";
        description "The object indicates the type of tunnel";
    }
    leaf tunnel-direction {
        type tunnel-direction-type;
        description "The object indicates the direction of tunnel";
    }
}

list section {
    key "section-id";
    description "Section";
    leaf section-id {
        type uint64 {
            range "1..2147483647";
        }
        description "This object indicates the section ID";
    }
    leaf if-name {
        type string {
            length "1..63";
        }
        description "The object indicates the interface name";
    }
    leaf peer-ip {
        type inet:ip-address;
        mandatory "true";
        description "This object indicates the peer IP address";
    }
    leaf peer-lsr-id {
        type inet:ip-address;
        description "This object indicates the peer lsr ID";
    }
}

```

Internet-Draft

MPLS-TP OAM YANG

October 2017

```
    }
    leaf mep-id {
      type uint16 {
        range "1..8191";
      }
      description "This object indicates the MEP Id of
        local ME";
    }
    leaf remote-mep-id {
      type uint16 {
        range "1..8191";
      }
      description "This object indicates the MEP Id of
        remote ME";
    }
  }
}

list pw-spme {
  key "local-peer-ip local-vc-id switch-peer-ip
    switch-vc-id vc-type instance-name";
  description "PW-SPME";
  uses pw-spme;
}

list lsp-spme {
  key "tunnel-id local-lsr-id remote-lsr-id";
  description "LSP-SPME";
  leaf tunnel-id {
    type uint32 {
      range "1..65535";
    }
    description "The object indicates the tunnel id";
  }
  leaf local-lsr-id {
    type inet:ip-address;
    description "The object indicates the ingress LSR-ID";
  }
  leaf remote-lsr-id {
    type inet:ip-address;
    description "The object indicates the egress LSR-ID";
  }
}
```

```
container cc {
  description "CC";
  leaf cc-session-mode {
    type cc-session-mode;
    default "coordinated";
  }
}
```

```
  description "This object indicates the session
  mode of CC";
}
leaf cc-authentication-enable {
  type enable;
  default "true";
  description "CC authentication enable";
}
leaf cc-exp {
  type uint8 {
    range "0..7";
  }
  default "7";
  description "This object indicates the exp of CC
  packet which is sent in the MEG";
}
leaf cc-transmit-interval {
  type cc-interval;
  default "1";
  description "The interval of CC packet which is
  transmit in the MEG";
}
leaf cc-recieve-interval {
  type cc-interval;
  default "1";
  description "The interval of CC packet which is
  recieved in the MEG";
}
leaf cc-detect-multiplier {
  type cc-detect-multiplier;
  default "3";
  description "The object indicate the detect
  multiplier of CC packet";
}
leaf cc-enable {
  type enable;
}
```

```

        default "true";
        description "The object indicates whether CC can be
        sent by the MEG";
    }
}
container cv {
    description "CV";
    leaf cv-session-mode {
        type cc-session-mode;
        default "coordinated";
        description "This object indicates the session
        mode of CC";
    }
}

```

```

    }
    leaf cv-authentication-enable {
        type enable;
        default "true";
        description "CV authentication enable";
    }
    leaf cv-exp {
        type uint8 {
            range "0..7";
        }
        default "7";
        description "This object indicates the exp of CV packet
        which is sent in the MEG";
    }
    leaf cv-interval {
        type cv-interval;
        default "1";
        description "The interval of CV packet which is sent
        in the MEG";
    }
    leaf cv-detect-multiplier {
        type cv-detect-multiplier;
        default "3";
        description "The object indicate the detect multiplier
        of CV packet";
    }
    leaf cv-enable {
        type enable;
        default "true";
    }
}

```

```

        description "The object indicates whether CC can be
        received by the MEG";
    }
}

container ais {
    description "AIS";
    leaf ais-exp {
        type uint8 {
            range "0..7";
        }
        default "7";
        description "This object indicates the exp of AIS packet
        which is sent in the MEG";
    }
    leaf ais-interval {
        type ais-interval;
        default "interval1000ms";
        description "This object indicates the interval of AIS

```

```

        packet which is sent in the MEG";
    }
}

container lkr {
    description "LKR";
    leaf lkr-exp {
        type uint8 {
            range "0..7";
        }
        default "7";
        description "This object indicates the exp of lock report
        packet which is sent in the MEG";
    }
    leaf lkr-interval {
        type lkr-interval;
        default "interval1000ms";
        description "This object indicates the interval of lock
        report packet which is sent in the MEG";
    }
    leaf lkr-enable {
        type enable;
    }
}

```

```

        default "false";
        description "The object indicates whether lock report
        is enabled in the MEG";
    }
}

container one-way-dm-send {
    description "One way delay measurement send";
    leaf one-dm-send-enable {
        type enable;
        default "false";
        description "This object indicates the 1DM statistics
        is enabled in the MEG";
    }
    leaf one-dm-interval {
        type dm-interval;
        default "1000";
        description "This object indicates the interval of
        1DM statistics in the MEG";
    }
}
leaf one-dm-exp {
    type uint8 {
        range "0..7";
    }
    default "7";
}

```

```

        description "This object indicates the exp of 1DM
        packet which is sent in the MEG";
    }
    leaf one-dm-packet-size {
        type uint16 {
            range "64..1518";
        }
        description "This object indicates the packet size
        of 1DM packet which is sent in the MEG";
    }
    leaf one-dm-pad-value {
        type dm-padding-value;
        default "paddingvalue0";
        description "This object indicates the padding value
        of 1DM packet which is sent in the MEG";
    }
}

```

```

    }
}

container one-way-dm-rcv {
    description "One way delay measurement received";
    leaf onr-dm-rcv-enable {
        type enable;
        default "false";
        description "This object indicates the 1DM receive
            is enabled in the MEG";
    }
    leaf one-dm-rcv-enable-type {
        type one-way-rcv-type;
        description "This object indicates the 1DM receive type";
    }
}

container two-way-dm-send {
    description "Two way delay measurement send";
    leaf two-dm-send-enable {
        type enable;
        default "false";
        description "This object indicates the 2DM statistics
            is enabled in the MEG";
    }
    leaf two-dm-interval {
        type dm-interval;
        default "1000";
        description "This object indicates the interval of
            2DM statistics in the MEG";
    }
    leaf two-dm-exp {
        type uint8 {

```

```

        range "0..7";
    }
    default "7";
    description "This object indicates the exp of 2DM
        packet which is sent in the MEG";
}
leaf two-dm-packet-size {
    type uint16 {

```



```

        range "64..1518";
    }
    description "This object indicates the packet size of
    2DM packet which is sent in the MEG";
}
leaf two-dm-pad-value {
    type dm-padding-value;
    default "paddingvalue0";
    description "This object indicates the padding value of
    2DM packet which is sent in the MEG";
}
leaf two-dm-time-stamp {
    type enable;
    default "false";
    description "This object indicates whether two-way delay
    measurement time stamp is enable in the MEG";
}
}

container two-way-dm-rcv {
    description "Two way delay measurement recieved";
    leaf two-dm-rcv-enable {
        type enable;
        default "false";
        description "This object indicates the 2DM receiving
        statistics is enabled in the MEG";
    }
}

container single-lm-send {
    description "Single loss measurment send";
    leaf slm-send-enable {
        type enable;
        default "false";
        description "This object indicates whether slm send
        is enable in the MEG";
    }
    leaf slm-interval {
        type lm-interval;
        default "1000";
    }
}

```

description "This object indicates the interval of

```

    slm statistics in the MEG";
}
leaf slm-exp {
    type uint8 {
        range "0..7";
    }
    default "7";
    description "This object indicates the exp of slm
    packet which is sent in the MEG";
}
}

container single-lm-rcv {
    description "Single loss measurment received";
    leaf slm-rcv-enable {
        type enable;
        default "false";
        description "This object indicates whether slm
        receive is enable in the MEG";
    }
}

container dual-lm {
    description "Dual loss measurement";
    leaf dlm-enable {
        type enable;
        default "false";
        description "This object indicates the dual loss
        statistics is enabled in the MEG";
    }
}

container status-info {
    config "false";
    description "Status info";
    list pw {
        uses me-detect-status;
        description "PW";
    }
    list lsp {
        uses me-detect-status;
        leaf me-index-egress {
            type uint32 {
                range "1..65535";
            }
            description "The object indicates the egress index
            of ME";
        }
    }
}

```

```
}
leaf me-direct-egress {
    type me-direction;
    description "The object indicates the direction of
    egress ME";
}
leaf status-board-egress {
    type string {
        length "1..19";
    }
    description "The object indicates the selected status
    board of ME";
}
leaf state-egress {
    type me-state;
    description "The object indicates the status of ME";
}
leaf alarm-egress {
    type string {
        length "1..26";
    }
    description "The object indicates the alarm of ME";
}
description "LSP";
}
list section {
    uses me-detect-status;
    description "Section";
}
list pw-spme {
    leaf me-index {
        type uint32 {
            range "1..65535";
        }
        description "The object indicates the index of MIP";
    }
    leaf me-direction {
        type me-direction;
        description "The object indicates the direction of MIP";
    }
    leaf me-state {
        type me-state;
        description "The object indicates the state of MIP";
    }
    leaf mip-id {
        type uint16 {
```

```
        range "1..8191";
    }
```

```
        description "The object indicates the ID of MIP";
    }
    description "PW-SPME";
}
list lsp-spme{
    leaf me-index {
        type uint32 {
            range "1..65535";
        }
        description "The object indicates the index of te MIP";
    }
    leaf me-direction {
        type me-direction;
        description "The object indicates the direction of
te MIP";
    }
    leaf me-state {
        type me-state;
        description "The object indicates the state of te MIP";
    }
    leaf mip-id {
        type uint16 {
            range "1..8191";
        }
        description "The object indicates the ID of te MIP";
    }
    description "LSP-SPME";
}
}
container detect-result {
    config "false";
    description "Detect result";
    container one-way-dm-result {
        config "false";
        description "One way delay measurement result";
        leaf send-pkt-num {
            type uint32 {
                range "1..4294967295";
            }
        }
    }
}
```

```

        description "Send packet number";
    }
    leaf recv-pkt-num {
        type uint32 {
            range "1..4294967295";
        }
        description "Recieved packet number";
    }
    leaf delay-min {

```

```

        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the minimum delay
of received LB packets in the MEG";
    }
    leaf delay-max {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the maximum delay
of received LB packets in the MEG";
    }
    leaf delay-avg {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the average delay
of received LB packets in the MEG";
    }
    leaf jitter-min {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the minimum jitter
of received LB packets in the MEG";
    }
    leaf jitter-max {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the average jitter

```

```

    of received LB packets in the MEG";
}
leaf jitter-avg {
    type uint32 {
        range "1..4294967295";
    }
    description "This object indicates the average jitter
of received LB packets in the MEG";
}

container one-way-dm-data {
    config "false";
    description "One way delay measurement data";
    list one-way-dm-data {
        key "index";
        leaf index {

```

```

    type uint32 {
        range "1..4294967295";
    }
    description "This object indicates index of 1DM
statistics record in the MEG";
}
leaf one-delay {
    type uint32 {
        range "1..4294967295";
    }
    description "This object indicates delay of 1DM
statistics in the MEG";
}
leaf one-delay-var {
    type uint32 {
        range "1..4294967295";
    }
    description "This object indicates delay Variation
of 1DM statistics in the MEG";
}
leaf error-info {
    type error-info;
    description "This object indicates the error info
of statistics record in the MEG";
}
}

```

```

        description "One way delay measurement data";
    }
}

container one-way-send-result {
    config "false";
    description "One way send result";
    leaf measure-mode {
        type measure-mode;
        default "on-demand";
        description "The flag indicates whether the measurement
            is an on-demand or a continue measurement";
    }
    leaf status {
        type statistics-status;
        default "finished";
        description "The flag indicates whether the measurement
            is finished";
    }
}

container two-way-dm-result {

```

```

    config "false";
    description "Two way delay measurement result.";
    leaf measure-mode {
        type measure-mode;
        default "on-demand";
        description "The flag indicates whether the measurement
            is an on-demand or a continue measurement";
    }
    leaf status {
        type statistics-status;
        default "finished";
        description "The flag indicates whether the measurement
            is finished";
    }
    leaf send-pkt-num {
        type uint32 {
            range "1..4294967295";
        }
    }

```

```

        description "Send packet number";
    }
    leaf rcv-pkt-num {
        type uint32 {
            range "1..4294967295";
        }
        description "Received packet number";
    }
    leaf delay-min {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the minimum delay
of received LB packets in the MEG";
    }
    leaf delay-max {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the maximum delay
of received LB packets in the MEG";
    }
    leaf delay-avg {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the average delay
of received LB packets in the MEG";
    }
    leaf jitter-min {

```

```

        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the minimum jitter
of received LB packets in the MEG";
    }
    leaf jitter-max {
        type uint32 {
            range "1..4294967295";
        }
        description "This object indicates the average jitter

```



```

    of received LB packets in the MEG";
}
leaf jitter-avg {
    type uint32 {
        range "1..4294967295";
    }
    description "This object indicates the average jitter
of received LB packets in the MEG";
}
container two-way-dm-data {
    config "false";
    description "Two way delay measurement data";
    list two-way-dm-data {
        key "index";
        leaf index {
            type uint32 {
                range "1..4294967295";
            }
            description "This object indicates index of 2DM
statistics record in the MEG";
        }
        leaf two-delay {
            type uint32 {
                range "1..4294967295";
            }
            description "This object indicates delay of 2DM
statistics in the MEG";
        }
        leaf two-delay-var {
            type uint32 {
                range "1..4294967295";
            }
            description "This object indicates delay Variation
of 2DM statistics in the MEG";
        }
        leaf error-info {
            type error-info;

```

```

        description "This object indicates the error info
of statistics record in the MEG";
    }
    description "Two way delay measurement data";

```

```

    }
  }
}

container single-lm-result {
  config "false";
  description "Single loss measurement result.";
  leaf measure-mode {
    type measure-mode;
    default "on-demand";
    description "The flag indicates whether the measurement
      is an on-demand or a continue measurement";
  }
  leaf status {
    type statistics-status;
    default "finished";
    description "The flag indicates whether the measurement
      is finished";
  }
  leaf send-pkt-num {
    type uint32 {
      range "1..4294967295";
    }
    description "Send packet number";
  }
  leaf rcv-pkt-num {
    type uint32 {
      range "1..4294967295";
    }
    description "Received packet number";
  }
  leaf rmt-loss-ratio-min {
    type uint32 {
      range "1..4294967295";
    }
    description "This object indicates the minimum loss-ratio
      of received LB packets in the MEG";
  }
  leaf rmt-loss-ratio-max {
    type uint32 {
      range "1..4294967295";
    }
    description "This object indicates the maximum loss-ratio
      of received LB packets in the MEG";
  }
}

```

```
}
leaf rmt-loss-atio-avg {
  type uint32 {
    range "1..4294967295";
  }
  description "This object indicates the average loss-ratio
of received LB packets in the MEG";
}
leaf rmt-loss-count-min {
  type uint32 {
    range "1..4294967295";
  }
  description "This object indicates the minimum packet
lost of received LB packets in the MEG";
}
leaf rmt-loss-count-max {
  type uint32 {
    range "1..4294967295";
  }
  description "This object indicates the average packet
lost of received LB packets in the MEG";
}
leaf rmt-loss-count-avg {
  type uint32 {
    range "1..4294967295";
  }
  description "This object indicates the average packet
lost of received LB packets in the MEG";
}

container single-lm-data {
  config "false";
  description "Single loss measurement data";
  list single-lm-data {
    key "index";
    leaf index {
      type uint32 {
        range "1..4294967295";
      }
      description "This object indicates index of slm
statistics record in the MEG";
    }
    leaf slm-loss-lcl {
      type uint32 {
        range "1..4294967295";
      }
      description "This object indicates local packet
```

loss of slm statistics in the MEG";

```
    }
    leaf slm-loss-lcl-rat {
      type string {
        length "1..24";
      }
      description "This object indicates local packet
        loss rate of slm statistics in the MEG";
    }
    leaf slm-loss-rmt {
      type uint32 {
        range "1..4294967295";
      }
      description "This object indicates remote packet
        loss of slm statistics in the MEG";
    }
    leaf slm-loss-rmt-rat {
      type string {
        length "1..24";
      }
      description "This object indicates remote packet
        loss rate of slm statistics in the MEG";
    }
    leaf error-info {
      type error-info;
      description "This object indicates the error
        info of statistics record in the MEG";
    }
    description "Single loss measurement data";
  }
}

container dual-lm-data {
  config "false";
  description "Dual loss measurement data";
  list dual-lm-data {
    key "index";
    leaf index {
      type uint32 {
        range "1..4294967295";
      }
    }
  }
}
```

```

    }
    description "This object indicates index of dlm
    statistics record in the MEG";
  }
  leaf dlm-Loss-lcl {
    type uint32 {
      range "1..4294967295";
    }
  }

```

```

    description "This object indicates local packet
    loss of dlm statistics in the MEG";
  }
  leaf dlm-loss-lcl-rat {
    type string {
      length "1..24";
    }
    description "This object indicates local packet
    loss rate of dlm statistics in the MEG";
  }
  leaf dlm-loss-rmt {
    type uint32 {
      range "1..4294967295";
    }
    description "This object indicates remote packet
    loss of dlm statistics in the MEG";
  }
  leaf dlm-loss-rmt-rat {
    type string {
      length "1..24";
    }
    description "This object indicates remote packet
    loss rate of dlm statistics in the MEG";
  }
  leaf error-info {
    type error-info;
    description "This object indicates the error info
    of statistics record in the MEG";
  }
  description "Dual loss measurement data";
}
}
}

```

```
    }  
  }  
}  
<CODE ENDS>
```

[7.](#) Examples

Examples of using YANG module to configure and manage MPLS-TP OAM will be given here in the update.

[8.](#) Security Considerations

The configuration and state data defined in this document is designed to be accessed via the NETCONF protocol [[RFC6241](#)]. The lowest NETCONF layer is the secure transport layer and the mandatory-to-

Zhang, et al.

Expires May 2, 2018

[Page 47]

Internet-Draft

MPLS-TP OAM YANG

October 2017

implement secure transport is SSH [[RFC6242](#)]. The authors recommend to implement the NETCONF access control model [[RFC6536](#)] to restrict access for particular NETCONF users to a pre-configured subset of all available NETCONF protocol operations and content. There are a number of config true nodes defined in the YANG module which are writable/creatable/deletable. These data nodes may be considered sensitive or vulnerable in some network environments. Write operations to these data nodes without proper protection can have a negative effect on network operations.

[9.](#) IANA Considerations

The IANA is requested to as assign a new namespace URI from the IETF XML registry.

URI:TBA

[10.](#) Acknowledgements

TBD

[11.](#) References

[11.1.](#) 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>>.

- [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>>.
- [RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", [RFC 6536](#), DOI 10.17487/RFC6536, March 2012, <<https://www.rfc-editor.org/info/rfc6536>>.

Zhang, et al.

Expires May 2, 2018

[Page 48]

Internet-Draft

MPLS-TP OAM YANG

October 2017

- [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>>.

[11.2.](#) Informative References

- [RFC5921] Bocci, M., Ed., Bryant, S., Ed., Frost, D., Ed., Levrau, L., and L. Berger, "A Framework for MPLS in Transport Networks", [RFC 5921](#), DOI 10.17487/RFC5921, July 2010, <<https://www.rfc-editor.org/info/rfc5921>>.

Authors' Addresses

Li Zhang
Huawei Technologies
China

Email: monica.zhangli@huawei.com

Lianshu Zheng
Huawei Technologies
China

Email: vero.zheng@huawei.com

Sam K. Aldrin
Google
USA

Email: aldrin.ietf@gmail.com

Greg Mirsky
ZTE Corp.
USA

Email: gregimirsky@gmail.com