Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 29, 2015

L. Zhang
L. Zheng
S. Aldrin
Huawei Technologies

September 25, 2014

YANG Data Model for MPLS-TP Operations, Administration, and Maintenance (OAM)
draft-zhang-mpls-tp-yang-oam-00

Abstract

This document presents the YANG Data model for MPLS-TP OAM, including the basic functions of Fault Management and Performance Monitoring.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 http://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 March 29, 2015.

Copyright Notice

Copyright (c) 2014 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 (http://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

This document presents the YANG Data model for MPLS-TP OAM, including the basic functions of Fault Management and Performance Monitoring.

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 - Packet Delay Measurement

AIS - Alarm Indication Signal

LKR - Lock Report

3. Design of the Data Model

At the top of the Model is the global configuration, which indicate the MPLS-TP OAM basic information.

Under the global configuration is Maintenance Entity Group. Each Maintenance Entity Group is associated with a Maintenance Entity Group Name and a Maintenance Entity Group level, and a certain type of Maintenace Entity, such as PW, TE, SECTION and so on.

Under each MEG, there can be one or more MEs (Maintenance Entity), which should be consistent with the certain ME type assigned above. Each ME has its own indication need to be configurated exactly.

Subsequently, the basic function of Fault Management and Performance Monitoring for each MEG should be offered. 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. MPLS-TP OAM Global Configuration

The container "global" is the top level container. Within the container "global", separate leaf nodes are maintained.

```
   +--rw global
      |  +--rw aisEnable?     Enable
      .
      .
```

Figure 1 Snippet of data hierarchy related to MPLS-TP OAM Global

3.2. Maintenance Entity Group (MEG) Configuration

The container "megs" is the second level container within the MPLS-TP oam module. Within the container "MEGs", separate lists are maintained per MEG. The MEG list uses the key MEG-name for indexing.

```
   +--rw global
      |  +--rw aisEnable?     Enable
      +--rw megs
      |  +--rw meg* [megName]
      |     +--rw megName           string
      |     +--rw meType?           mplstpoamMeType
      |     +--rw megId?            string
      |     +--rw megLevel?         uint8
      |     +--rw oamActiveState?   mplstpoamActiveType

      .
      .
```

Figure 2 Snippet of data hierarchy related to MPLS-TP OAM MEGs

3.3. Maintenance Entities (MEs) Configuration

Within a given Maintenance Entity Group there can be one or more Maintenance Entity (ME), which should be consistent with the certain ME type assigned within megs . Different types of MEs are represented as different list and indexed by theire own key.

3.3.1. Pseudo Wire (PW) Configuration

```
+--rw global
   |  .
   |  .
   +--rw megs
   |  +--rw meg* [megName]
   |     .
   |     .
   |     +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         |  +--rw peerIp                  inet:ip-address
         |  +--rw vcId                    uint32
         |  +--rw vcType                  mplstpoamVctype
         |  +--rw remotePeerIp            inet:ip-address
         |  +--rw remoteVcId              mplstpoamVctype
         |  +--rw remoteVcType            mplstpoamVctype
         |  +--rw mepId?                  uint16
         |  +--rw remoteMepId?            uint16
         |  +--rw vllttl?                 uint8
   .
   .
```

Figure 3 Snippet of data hierarchy related to ME of PW

3.3.2. Traffic Engineering (TE) Configuration

```
+--rw global
   .
   .
   +--rw megs
   |  +--rw meg* [megName]
         .
         .
      +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
        .
        .
         +--rw te* [tunnelName tunnelId ingressLsrId]
         |  +--rw tunnelName              string
         |  +--rw tunnelId                uint32
         |  +--rw ingressLsrId            inet:ip-address
         |  +--rw mepId?                  uint16
         |  +--rw remoteMepId?            uint16
         |  +--rw reverseTunnelName       string
         |  +--rw reverseTunnelId?        uint16
         |  +--rw reverseIngrLsrId?       inet:ip-address

   .
   .
```

Figure 4 Snippet of data hierarchy related to ME of TE

3.3.3. Section Configuration

```
+--rw global
   .
   .
   +--rw megs
   |  +--rw meg* [megName]
         .
         .
         +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         .
         .
         +--rw te* [tunnelName tunnelId ingressLsrId]
         .
         .
         +--rw section* [sectionId]
         |  +--rw sectionId               uint64
         |  +--rw ifName?                 string
         |  +--rw peerIp                  inet:ip-address
         |  +--rw peerLsrId?              inet:ip-address
         |  +--rw mepId?                  uint16
         |  +--rw remoteMepId?            uint16
   .
   .
```

Figure 5 Snippet of data hierarchy related to ME of Section

3.3.4. Virtual Leased Line (VLL) Maintenance Entity Group Intermediate Points(MIPs) Configuration

```
+--rw global
   .
   .
   +--rw megs
   |  +--rw meg* [megName]
         .
         .
         +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         .
         .
         +--rw te* [tunnelName tunnelId ingressLsrId]
         .
         .
         +--rw section* [sectionId]
         .
         .
         +--rw vllMip* [peerIp vcId switchPeerIp switchVcId vcType instanceName]

         |  +--rw peerIp          inet:ip-address
         |  +--rw vcId            uint32
         |  +--rw switchPeerIp    inet:ip-address
         |  +--rw switchVcId      uint32
         |  +--rw vcType          mplstpoamVctype
         |  +--rw instanceName    string
   .
   .
```

Figure 6 Snippet of data hierarchy related to ME of VLL-MIP

3.3.5. Traffic Engineering (TE) Maintenance Entity Group Intermediate Points(MIPs) Configuration

```
+--rw global
   .
   .
   +--rw megs
   |  +--rw meg* [megName]
         .
         .
         +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         .
         .
         +--rw te* [tunnelName tunnelId ingressLsrId]
         .
         .
         +--rw section* [sectionId]
         .
         .
         +--rw vllMip* [peerIp vcId switchPeerIp switchVcId vcType instanceName]
         .
         .
         +--rw teMip* [lspName]
         |  +--rw lspName        string
   .
   .
```

Figure 7 Snippet of data hierarchy related to ME of TE-MIP

3.4. MPLS-TP OAM Fault Management And Performance Moinitoring Configuration

3.4.1. Fault Management Configuration

The container "cc", "cv", "ais" and "lkr" are indicate the Fault Management tools of "Continuity Check","Connectivity Verification". "Alarm Indication Signal" and "Lock Report", which are used by the MEs of MEG.

```
+--rw meg* [megName]
         .
         .
         +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         .
         .
         +--rw te* [tunnelName tunnelId ingressLsrId]
         .
         .
         +--rw section* [sectionId]
         .
         .
         +--rw vllMip* [peerIp vcId switchPeerIp switchVcId vcType instanceName]
         .
         .
         +--rw teMip* [lspName]
         .
         .
         +--rw cc
         |  +--rw ccSessionMode?            mplstpoamCcSessionMode
         |  +--rw ccAuthenticationEnable?   Enable
         |  +--rw ccExp?                    uint8
         |  +--rw ccTransmitInterval?       mplstpoamCcInterval
         |  +--rw ccRecieveInterval?        mplstpoamCcInterval
         |  +--rw ccDetectMultiplier?       mplstpoamCcDetectMultiplier
         |  +--rw ccEnable?                 Enable
         +--rw cv
         |  +--rw cvSessionMode?            mplstpoamCcSessionMode
         |  +--rw cvAuthenticationEnable?   Enable
         |  +--rw cvExp?                    uint8
         |  +--rw cvInterval?               mplstpoamCvInterval
         |  +--rw cvDetectMultiplier?       mplstpoamCvDetectMultiplier
         |  +--rw cvEnable?                 Enable
         +--rw ais
         |  +--rw aisExp?        uint8
         |  +--rw aisInterval?   mplstpoamAisInterval
         +--rw lkr
         |  +--rw lkrExp?        uint8
         |  +--rw lkrInterval?   mplstpoamLkrInterval
         |  +--rw lkrEnable?     Enable
         .
         .
```

Figure 8 Snippet of data hierarchy related to ME of Fault Management

3.4.2. Performance Monitoring Configuration

The performance monitoring configuration is consist of seperate containers of delay, packet loss and jitter measurements.

```
|  +--rw meg* [megName]
         .
         .
         +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         .
         .
         +--rw te* [tunnelName tunnelId ingressLsrId]
         .
         .
         +--rw section* [sectionId]
         .
         .
         +--rw vllMip* [peerIp vcId switchPeerIp switchVcId vcType instanceName]
         .
         .
         +--rw teMip* [lspName]
         .
         .
         +--rw cc
         .
         .
         +--rw cv
         .
         .
         +--rw ais
         .
         .
         +--rw lkr
         .
         .
         +--rw oneWayDmSend
         |  +--rw oneDmSendEnable?   Enable
         |  +--rw oneDmInterval?     mplstpoamDmInterval
         |  +--rw oneDmExp?          uint8
         |  +--rw oneDmPacketSize?   uint16
         |  +--rw oneDmPadValue?     mplstpoamDmPaddingValue
         +--rw oneWayDmRcv
         |  +--rw oneDmRcvEnable?       Enable
         |  +--rw oneDmRcvEnableType?   mplstpoamOneWayRcvType
         +--rw twoWayDmSend
         |  +--rw twoDmSendEnable?   Enable
         |  +--rw twoDmInterval?     mplstpoamDmInterval
         |  +--rw twoDmExp?          uint8
         |  +--rw twoDmPacketSize?   uint16
         |  +--rw twoDmPadValue?     mplstpoamDmPaddingValue
         |  +--rw twoDmTimestamp?    Enable
         +--rw twoWayDmRcv
         |  +--rw twoDmRcvEnable?   Enable
         +--rw singleLmSend
         |  +--rw slmSendEnable?   Enable
         |  +--rw slmInterval?     mplstpoamSlmInterval
         |  +--rw slmExp?          uint8
         +--rw singleLmRcv
         |  +--rw slmRcvEnable?   Enable
         +--rw dualLm
         |  +--rw dlmEnable?   Enable
```

Figure 8 Snippet of data hierarchy related to MEG performance monitoring configuration

4. MPLS-TP OAM Data Hierarchy

The complete data hierarchy related to the MPLS-TP OAM YANG model is presented below. The following notations are used within the data tree and carry the meaning as below.

Each node is printed as:

<status> <flags> <name> <opts> <type>

<status> is one of:

+ for current

x for deprecated

o for obsolete

<flags> is one of:

rw for configuration data

ro for non-configuration data

-x for rpcs

-n for notifications

<name> is the name of the node

If the node is augmented into the tree from another module, its name is printed as <prefix>:<name>.

<opts> is one of:

? for an optional leaf or choice

! for a presence container

* for a leaf-list or list

[<keys>] for a list's keys

<type> is the name of the type for leafs and leaf-lists

```
module: mplstpoam
   +--rw global
   |  +--rw aisEnable?   Enable
   +--rw megs
      +--rw meg* [megName]
         +--rw megName             string
         +--rw meType?             mplstpoamMeType
         +--rw megId?              string
         +--rw megLevel?           uint8
         +--rw oamActiveState?     mplstpoamActiveType
         +--rw pw* [peerIp vcId vcType remotePeerIp remoteVcId remoteVcType]
         |  +--rw peerIp                   inet:ip-address
         |  +--rw vcId                     uint32
         |  +--rw vcType                   mplstpoamVctype
         |  +--rw remotePeerIp             inet:ip-address
         |  +--rw remoteVcId               mplstpoamVctype
         |  +--rw remoteVcType             mplstpoamVctype
         |  +--rw mepId?                   uint16
         |  +--rw remoteMepId?             uint16
         |  +--rw vllttl?                  uint8
         |  +--ro meIndex?                 uint32
         |  +--ro meDirection?             mplstpoamMeDirection
         |  +--ro meState?                 mplstpoamMeState
         |  +--ro localState?              mplstpoamMeState
         |  +--ro remoteState?             mplstpoamMeState
         |  +--ro alarmIndicate?           string
         |  +--ro localDefectStatus?       mplstpoamDefectStatusType
         |  +--ro localInvalidTime?        uint32
         |  +--ro localDefectLocation?     string
         |  +--ro localDefectType?         mplstpoamDefectType
         |  +--ro remoteDefectStatus?      mplstpoamDefectStatusType
         |  +--ro remoteInvalidTime?       uint32
         |  +--ro remoteDefectLocation?    string
         |  +--ro remoteDefectType?        mplstpoamDefectType
         |  +--rw galEnable?               Enable
         |  +--rw galMode?                 mplstpoamGalMode
         +--rw te* [tunnelName tunnelId ingressLsrId]
         |  +--rw tunnelName                string
         |  +--rw tunnelId                  uint32
         |  +--rw ingressLsrId              inet:ip-address
         |  +--rw mepId?                    uint16
         |  +--rw remoteMepId?              uint16
         |  +--rw reverseTunnelName         string
         |  +--rw reverseTunnelId?          uint16
         |  +--rw reverseIngrLsrId?         inet:ip-address
         |  +--ro tunnelDescription?        string
         |  +--ro tunnelType?               mplstpoamTunnelType
         |  +--ro tunnelDirection?          mplstpoamTunnelDirectionType
         |  +--ro meIndex?                  uint32
         |  +--ro meDirection?              mplstpoamMeDirection
         |  +--ro meState?                  mplstpoamMeState
         |  +--ro localState?               mplstpoamMeState
         |  +--ro remoteState?              mplstpoamMeState
         |  +--ro alarmIndicate?            string
         |  +--ro localDefectStatus?        mplstpoamDefectStatusType
         |  +--ro localInvalidTime?         uint32
         |  +--ro localDefectLocation?      string
         |  +--ro localDefectType?          mplstpoamDefectType
         |  +--ro remoteDefectStatus?       mplstpoamDefectStatusType
         |  +--ro remoteInvalidTime?        uint32
         |  +--ro remoteDefectLocation?     string
         |  +--ro remoteDefectType?         mplstpoamDefectType
         |  +--ro meIndexEgress?            uint32
         |  +--ro meDirectEgress?           mplstpoamMeDirection
         |  +--ro statusBoardEgress?        string
         |  +--ro stateEgress?              mplstpoamMeState
         |  +--ro alarmEgress?              string
         +--rw section* [sectionId]
         |  +--rw sectionId                 uint64
         |  +--rw ifName?                   string
         |  +--rw peerIp                    inet:ip-address
         |  +--rw peerLsrId?                inet:ip-address
         |  +--rw mepId?                    uint16
         |  +--rw remoteMepId?              uint16
         |  +--ro meIndex?                  uint32
         |  +--ro meDirection?              mplstpoamMeDirection
         |  +--ro meState?                  mplstpoamMeState
         |  +--ro localState?               mplstpoamMeState
         |  +--ro remoteState?              mplstpoamMeState
         |  +--ro alarmIndicate?            string
         |  +--ro localDefectStatus?        mplstpoamDefectStatusType
         |  +--ro localInvalidTime?         uint32
         |  +--ro localDefectLocation?      string
         |  +--ro localDefectType?          mplstpoamDefectType
         |  +--ro remoteDefectStatus?     mplstpoamDefectStatusType
         |  +--ro remoteInvalidTime?      uint32
         |  +--ro remoteDefectLocation?   string
         |  +--ro remoteDefectType?       mplstpoamDefectType
         +--rw vllMip* [peerIp vcId switchPeerIp switchVcId vcType instanceName]

         |  +--rw peerIp           inet:ip-address
         |  +--rw vcId             uint32
         |  +--rw switchPeerIp     inet:ip-address
         |  +--rw switchVcId       uint32
         |  +--rw vcType           mplstpoamVctype
         |  +--rw instanceName     string
         |  +--ro meIndex?         uint32
         |  +--ro meDirection?     mplstpoamMeDirection
         |  +--ro meState?         mplstpoamMeState
         |  +--rw mipId?           uint16
         +--rw teMip* [lspName]
         |  +--rw lspName         string
         |  +--ro meIndex?        uint32
         |  +--ro meDirection?    mplstpoamMeDirection
         |  +--ro meState?        mplstpoamMeState
         |  +--rw mipId?          uint16
         +--rw cc
         |  +--rw ccSessionMode?            mplstpoamCcSessionMode
         |  +--rw ccAuthenticationEnable?   Enable
         |  +--rw ccExp?                    uint8
         |  +--rw ccTransmitInterval?       mplstpoamCcInterval
         |  +--rw ccRecieveInterval?        mplstpoamCcInterval
         |  +--rw ccDetectMultiplier?       mplstpoamCcDetectMultiplier
         |  +--rw ccEnable?                 Enable
         +--rw cv
         |  +--rw cvSessionMode?            mplstpoamCcSessionMode
         |  +--rw cvAuthenticationEnable?   Enable
         |  +--rw cvExp?                    uint8
         |  +--rw cvInterval?               mplstpoamCvInterval
         |  +--rw cvDetectMultiplier?       mplstpoamCvDetectMultiplier
         |  +--rw cvEnable?                 Enable
         +--rw ais
         |  +--rw aisExp?        uint8
         |  +--rw aisInterval?   mplstpoamAisInterval
         +--rw lkr
         |  +--rw lkrExp?        uint8
         |  +--rw lkrInterval?   mplstpoamLkrInterval
         |  +--rw lkrEnable?     Enable
         +--rw oneWayDmSend
         |  +--rw oneDmSendEnable?   Enable
         |  +--rw oneDmInterval?     mplstpoamDmInterval
         |  +--rw oneDmExp?          uint8
         |  +--rw oneDmPacketSize?   uint16
         |  +--rw oneDmPadValue?     mplstpoamDmPaddingValue
         +--rw oneWayDmRcv
         |  +--rw oneDmRcvEnable?       Enable
         |  +--rw oneDmRcvEnableType?   mplstpoamOneWayRcvType
         +--rw twoWayDmSend
         |  +--rw twoDmSendEnable?   Enable
         |  +--rw twoDmInterval?     mplstpoamDmInterval
         |  +--rw twoDmExp?          uint8
         |  +--rw twoDmPacketSize?   uint16
         |  +--rw twoDmPadValue?     mplstpoamDmPaddingValue
         |  +--rw twoDmTimestamp?    Enable
         +--rw twoWayDmRcv
         |  +--rw twoDmRcvEnable?   Enable
         +--rw singleLmSend
         |  +--rw slmSendEnable?   Enable
         |  +--rw slmInterval?     mplstpoamSlmInterval
         |  +--rw slmExp?          uint8
         +--rw singleLmRcv
         |  +--rw slmRcvEnable?   Enable
         +--rw dualLm
         |  +--rw dlmEnable?   Enable
         +--ro oneWayDmResult
         |  +--ro sendPktNum?      uint32
         |  +--ro recvPktNum?      uint32
         |  +--ro delayMin?        uint32
         |  +--ro delayMax?        uint32
         |  +--ro delayAvg?        uint32
         |  +--ro jitterMin?       uint32
         |  +--ro jitterMax?       uint32
         |  +--ro jitterAvg?       uint32
         |  +--ro oneWayDmDatas
         |     +--ro oneWayDmData* [index]
         |        +--ro index          uint32
         |        +--ro oneDelay?      uint32
         |        +--ro oneDelayVar?   uint32
         |        +--ro errorInfo?     mplstpoamErrorInfo
         +--ro oneWaySendResult
         |  +--ro measureMode?   mplstpoamMeasureMode
         |  +--ro status?        mplstpoamStatisticsStatus
         +--ro twoWayDmResult
         |  +--ro measureMode?     mplstpoamMeasureMode
         |  +--ro status?          mplstpoamStatisticsStatus
         |  +--ro sendPktNum?      uint32
         |  +--ro recvPktNum?      uint32
         |  +--ro delayMin?        uint32
         |  +--ro delayMax?        uint32
         |  +--ro delayAvg?        uint32
         |  +--ro jitterMin?       uint32
         |  +--ro jitterMax?       uint32
         |  +--ro jitterAvg?       uint32
         |  +--ro twoWayDmDatas
         |     +--ro twoWayDmData* [index]
         |        +--ro index          uint32
         |        +--ro twoDelay?      uint32
         |        +--ro twoDelayVar?   uint32
         |        +--ro errorInfo?     mplstpoamErrorInfo
         +--ro singleLmResult
         |  +--ro measureMode?       mplstpoamMeasureMode
         |  +--ro status?            mplstpoamStatisticsStatus
         |  +--ro sendPktNum?        uint32
         |  +--ro recvPktNum?        uint32
         |  +--ro rmtLossRatioMin?   uint32
         |  +--ro rmtLossRatioMax?   uint32
         |  +--ro rmtLossRatioAvg?   uint32
         |  +--ro rmtLossCountMin?   uint32
         |  +--ro rmtLossCountMax?   uint32
         |  +--ro rmtLossCountAvg?   uint32
         |  +--ro singleLmDatas
         |     +--ro singleLmData* [index]
         |        +--ro index            uint32
         |        +--ro slmLossLcl?      uint32
         |        +--ro slmLossLclRat?   string
         |        +--ro slmLossRmt?      uint32
         |        +--ro slmLossRmtRat?   string
         |        +--ro errorInfo?       mplstpoamErrorInfo
         +--ro dualLmDatas
            +--ro dualLmData* [index]
               +--ro index            uint32
               +--ro dlmLossLcl?      uint32
               +--ro dlmLossLclRat?   string
               +--ro dlmLossRmt?      uint32
               +--ro dlmLossRmtRat?   string
               +--ro errorInfo?       mplstpoamErrorInfo
```

5. MPLS-TP OAM YANG module

```
module mplstpoam {
    namespace "urn:ietf:params:xml:ns:yang:mplstpoam";
    //namespace need to be assigned by IANA
    prefix "mplstpoam";
    import ietf-inet-types {
    prefix inet;
    }
    organization "IETF MPLS (Multiprotocol Label Switching) Working Group";
    contact "monica.zhangli@huawei.com
```

```
        vero.zheng@huawei.com
        aldrin.ietf@gmail.com";
    description "MPLS TP OAM Yang Module";
    revision "2014-09-18";

    typedef Enable {
        type enumeration {
            enum "true" {
                value 0;
            }
            enum "false" {
                value 1;
            }
        }
    }
    typedef mplstpoamMeType {
        type enumeration {
            enum "none" {
                value 0;
                description "ME type is valid";
            }
            enum "vll" {
                value 1;
                description "ME type is vll";
            }
            enum "vpls" {
                value 2;
                description "ME type is vpls";
            }
            enum "ingress" {
                value 3;
                description "ME type is ingress";
            }
            enum "egress" {
                value 4;
                description "ME type is egress";
            }
            enum "co-route" {
                value 5;
                description "ME type is co-route bidirectional te";
            }
            enum "associate" {
                value 6;
                description "ME type is associate bidirectional te";
            }
            enum "section" {
                value 7;
                description "ME type is section";
            }
            enum "vllMip" {
            value 8;
                description "ME type is vllMip";
            }
            enum "teMip" {
                value 9;
                description "ME type is teMip";
            }
        }
    }
    typedef mplstpoamTeServiceType {
        type enumeration {
            enum "none" {
                value 0;
                description "Service type is valid";
            }
            enum "te-crlsp" {
                value 1;
                description "Service type is te-crlsp";
            }
            enum "associate-te-lsp" {
                value 2;
                description "Service type is associate-te-lsp";
            }
            enum "te-ingress" {
                value 3;
                description "Service type is te-ingress";
            }
            enum "te-egress" {
                value 4;
                description "Service type is te-egresst";
            }
        }
    }
    typedef mplstpoamCcSessionMode {
        type enumeration {
            enum "coordinated" {
                value 0;
            }
            enum "independent" {
                value 1;
            }
        }
    }
    typedef mplstpoamCcInterval {
        description "The value rang for cc packet transmit and receive interval"
;
        type uint32{
```

```
                       range "1..65535";
        }
    }
    typedef mplstpoamCvInterval {
        description "The value rang for cv packet transmit interval";
        type uint32{
                       range "1..65535";
        }
    }
    typedef mplstpoamCcDetectMultiplier {
        description "The value rang for cv packet detect multiplier";
        type uint8{
                       range "1..255";
        }
    }
    typedef mplstpoamCvDetectMultiplier {
        description "The value rang for cv packet detect multiplier";
        type uint8{
                       range "1..255";
        }
    }
    typedef mplstpoamLkrInterval {
        type enumeration {
            enum "interval1000ms" {
                value 0;
            }
            enum "interval60000ms" {
                value 1;
            }
        }
    }
    typedef mplstpoamAisInterval {
        type enumeration {
            enum "interval1000ms" {
                value 0;
            }
            enum "interval60000ms" {
                value 1;
            }
        }
    }
    typedef mplstpoamMeDirection {
        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";
            }
        }
    }
    typedef mplstpoamMeState {
        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";
            }
        }
    }
    typedef mplstpoamDmInterval {
        description "The value rang for dm packet transmit interval";
        type uint32 {
                       range "1..65535";
        }
    }
    typedef mplstpoamDmPaddingValue {
        type enumeration {
            enum "paddingvalue0" {
                value 0;
            }
            enum "paddingvalue1" {
                value 1;
            }
        }
    }
    typedef mplstpoamSlmInterval {
        description "The value rang for lm packet transmit interval";
        type uint32 {
```

```
                       range "1..65535";
        }
    }
    typedef mplstpoamMeasureMode {
        type enumeration {
            enum "on-demand" {
                value 0;
            }
            enum "proactive" {
                value 1;
            }
        }
    }
    typedef mplstpoamVctype {
        description "The namespace of the vc type of pw";
        type string {
                       length "1..8191";
        }
    }
    typedef mplstpoamStatisticsStatus {
        type enumeration {
            enum "finished" {
                value 0;
            }
            enum "working" {
                value 1;
            }
        }
    }
    typedef mplstpoamErrorInfo {
        type enumeration {
            enum "valid" {
                value 0;
            }
            enum "invalid-loss" {
                value 1;
            }
            enum "invalid-delay" {
                value 2;
            }
        }
    }
    typedef mplstpoamDefectStatusType {
            description "The namespace of defect status type";
        type string {
                       length "1..8191";
        }
    }
    typedef mplstpoamDefectType {
            description "The namespace of defect type";
        type string {
                       length "1..8191";
        }
    }
    typedef mplstpoamTunnelType {
        type enumeration {
            enum "ingress" {
                value 0;
            }
            enum "egress" {
                value 1;
            }
            enum "bidirectional" {
                value 2;
            }
        }
    }
    typedef mplstpoamTunnelDirectionType {
        type enumeration {
            enum "uniDrectional" {
                value 0;
            }
            enum "biDirectional" {
                value 1;
            }
        }
    }
    typedef mplstpoamActiveType {
        type enumeration {
            enum "deactive" {
                value 0;
            }
            enum "active" {
                value 1;
            }
        }
    }
    typedef mplstpoamGalMode {
        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";
            }
        }
    }
    typedef mplstpoamOneWayRcvType {
        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";
            }
        }
    }
    grouping ME-detect-status {
        description "This node indicate detect status of ME";
        leaf meIndex {
            description "The object indicates the index of ME";
            config "false";
            type uint32 {
                range "1..65535";
            }
        }
        leaf meDirection {
            description "The object indicates the direction of ME";
            config "false";
            type mplstpoamMeDirection;
        }
        leaf meState {
            description "The object indicates the state of ME";
            config "false";
            type mplstpoamMeState;
        }
        leaf localState {
            description "The object indicates the local status of ME";
            config "false";
            type mplstpoamMeState;
        }
        leaf remoteState {
            description "The object indicates the remote state of ME";
            config "false";
            type mplstpoamMeState;
        }
        leaf alarmIndicate {
            description "The object indicates the alarm of ME";
            config "false";
            type string {
                length "1..26";
            }
        }
        leaf localDefectStatus {
            description "This object indicates the local defect status";
            config "false";
            default "init";
            type mplstpoamDefectStatusType;
        }
        leaf localInvalidTime {
            description "This object indicates the invalid Time of local detect"
;
            config "false";
            type uint32 {
                range "0..4294967295";
            }
        }
        leaf localDefectLocation {
            description "This object indicates the local defect location";
            config "false";
            type string {
                length "1..32";
            }
        }
        leaf localDefectType {
            description "This object indicates the local defect type";
            config "false";
            type mplstpoamDefectType;
        }
        leaf remoteDefectStatus {
            description "This object indicates the remote defect status";
            config "false";
            default "init";
            type mplstpoamDefectStatusType;
        }
        leaf remoteInvalidTime {
            description "This object indicates the invalid Time of remote detect
";
            config "false";
            type uint32 {
                range "0..4294967295";
            }
        }
        leaf remoteDefectLocation {
            description "This object indicates the remote defect location";
            config "false";
            type string {
                length "1..32";
            }
        }
        leaf remoteDefectType {
            description "This object indicates the remote defect type";
            config "false";
            type mplstpoamDefectType;
        }

    }
    grouping gal-set {
        description "This object indicates the gal set";
        leaf galEnable {
            description "This object indicates the gal flag";
            config "true";
            default "true";
            type Enable;
        }
        leaf galMode {
            description "This object indicates the gal flag";
            config "true";
            type mplstpoamGalMode;
        }
    }

    container global {

        leaf aisEnable {
            description "This object indicates the global ais flag of mpls-tp oa
m";
            config "true";
            default "false";
            type Enable;
        }
    }
    container megs {
        status current;
        description "show the megs";

        list meg {

            key "megName";
            leaf megName {
                description "The object indicates the name of MEG";
                config "true";
                mandatory "true";
                type string {
                    length "1..14";
                }
            }
            leaf meType {
                description "The object indicates the type of ME";
                config "true";
                default "none";
                type mplstpoamMeType;
            }
            leaf megId {
                description "The object indicates the ID of MEG";
                config "true";
                type string {
                    length "1..96";
                }
            }
            leaf megLevel {
                description "The object indicates the level of MEG";
                config "true";
                default "7";
                type uint8 {
                    range "0..7";
                }
            }
            leaf oamActiveState {
                description "This object indicates the oam active state";
                config "true";
                default "deactive";
                type mplstpoamActiveType;
            }
            list pw {

                key "peerIp vcId vcType remotePeerIp remoteVcId remoteVcType";
                leaf peerIp {
                    description "This object indicates the peer IP address";
                    config "true";
                    mandatory "true";
                    type inet:ip-address;
                }
                leaf vcId {
                    description "This object indicates the vc ID of PW type ME";
                    config "true";
                    mandatory "true";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf vcType {
                    description "This object indicates the vc type of VC type ME
";
                    config "true";
                    mandatory "true";
                    type mplstpoamVctype;
                }
                leaf remotePeerIp {
                    description "This object indicates the remote peer IP of PW
type ME";
                    config "true";
                    type inet:ip-address;
                }
                leaf remoteVcId {
                    description "This object indicates the remote vc ID of PW ty
pe ME";
                    config "true";
                    type mplstpoamVctype;
                }
                leaf remoteVcType {
                    description "This object indicates the remote vc type of PW
type ME";
                    config "true";
                    type mplstpoamVctype;
                }
                leaf mepId {
                    description "This object indicates the MEP Id of local ME";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
                leaf remoteMepId {
                    description "This object indicates the MEP Id of remote ME";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
                leaf vllttl {
                    description "This object indicates the VLL ttl of PW type ME
";
                    config "true";
                    type uint8 {
                        range "1..255";
                    }
                }
                uses ME-detect-status;
                uses gal-set;
            }

            list te {
                key "tunnelName tunnelId ingressLsrId";
                leaf tunnelName  {
                    description "The object indicates the name of tunnel";
                    config "true";
                    mandatory "true";
                    type string {
                        length "0..63";
                    }
                }
                leaf tunnelId {
                    description "The object indicates the tunnel id";
                    config "true";
                    type uint32 {
                        range "1..65535";
                    }
                }
                leaf ingressLsrId {
                    description "The object indicates the ingress LSR-ID";
                    config "true";
                    type inet:ip-address;
                }
                leaf mepId {
                    description "This object indicates the MEP Id of local ME";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
                leaf remoteMepId {
                    description "This object indicates the MEP Id of remote ME";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
                leaf reverseTunnelName  {
                    description "The object indicates the name of reverse tunnel
";
                    config "true";
                    mandatory "true";
                    type string {
                        length "0..63";
                    }
                }

                leaf reverseTunnelId {
                    description "The object indicates the ingress reverse tunnel
Id";
                    config "true";
                    default "10";
                    type uint16 {
                        range "1..65535";
                    }
                }
                leaf reverseIngrLsrId {
                    description "The object indicates the ingress reverse LSR-ID
";
                    config "true";
                    type inet:ip-address;
                }
                leaf tunnelDescription {
                    description "The object indicates the description of tunnel"
;
                    config "false";
                    type string {
                        length "1..32";
                    }
                }
                leaf tunnelType {
                    description "The object indicates the type of tunnel";
                    config "false";
                    default "ingress";
                    type mplstpoamTunnelType;
                }
                leaf tunnelDirection {
                    description "The object indicates the direction of tunnel";
                    config "false";
                    type mplstpoamTunnelDirectionType;
                }
                uses ME-detect-status;
                leaf meIndexEgress {
                    description "The object indicates the egress index of ME";
                    config "false";
                    type uint32 {
                        range "1..65535";
                    }
                }
                leaf meDirectEgress {
                    description "The object indicates the direction of egress ME
";
                    config "false";
                    type mplstpoamMeDirection;
                }
                leaf statusBoardEgress {
                    description "The object indicates the selected status board
of ME";
                    config "false";
                    type string {
                        length "1..19";
                    }
                }
                leaf stateEgress {
                    description "The object indicates the status of ME";
                    config "false";
                    type mplstpoamMeState;
                }
                leaf alarmEgress {
                    description "The object indicates the alarm of ME";
                    config "false";
                    type string {
                        length "1..26";
                    }
                }
            }

            list section {

                key "sectionId";
                leaf sectionId {
                    description "This object indicates the section ID";
                    config "true";
                    type uint64 {
                        range "1..2147483647";
                    }
                }
                leaf ifName {
                    description "The object indicates the interface name";
                    config "true";
                    type string {
                        length "1..63";
                    }
                }
                leaf peerIp {
                    description "This object indicates the peer IP address";
                    config "true";
                    mandatory "true";
                    type inet:ip-address;
                }
                leaf peerLsrId {
                    description "This object indicates the peer lsr ID";
                    config "true";
                    type inet:ip-address;
                }
                leaf mepId {
                    description "This object indicates the MEP Id of local ME";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
                leaf remoteMepId {
                    description "This object indicates the MEP Id of remote ME";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
                uses ME-detect-status;
            }

            list vllMip {

                key "peerIp vcId switchPeerIp switchVcId vcType instanceName";
                leaf peerIp {
                    description "This object indicates the peer IP address of PW
 type MIP";
                    config "true";
                    mandatory "true";
                    type inet:ip-address;
                }
                leaf vcId {
                    description "This object indicates the vc ID of PW type MIP"
;
                    config "true";
                    mandatory "true";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf switchPeerIp {
                    description "This object indicates the peer IP address of PW
 switch node";
                    config "true";
                    mandatory "true";
                    type inet:ip-address;
                }
                leaf switchVcId {
                    description "This object indicates the vc id of PW switch no
de";
                    config "true";
                    mandatory "true";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf vcType {
                    description "This object indicates the vc type of PW type MI
P";
                    config "true";
                    mandatory "true";
                    type mplstpoamVctype;
                }
                leaf instanceName {
                    description "This object specifies the VPWS instance name";
                    config "true";
                    mandatory "true";
                    type string {
                        length "1..31";
                    }
                }
                leaf meIndex {
                    description "The object indicates the index of MIP";
                    config "false";
                    type uint32 {
                        range "1..65535";
                    }
                }
                leaf meDirection {
                    description "The object indicates the direction of MIP";
                    config "false";
                    type mplstpoamMeDirection;
                }
                leaf meState {
                    description "The object indicates the state of MIP";
                    config "false";
                    type mplstpoamMeState;
                }
                leaf mipId {
                    description "The object indicates the ID of MIP";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
            }

            list teMip {

                key "lspName";

                leaf lspName {
                    description "This object indicates the name of LSP";
                    type string {
                        length "1..16";
                    }
                }

                leaf meIndex {
                    description "The object indicates the index of te MIP";
                    config "false";
                    type uint32 {
                        range "1..65535";
                    }
                }
                leaf meDirection {
                    description "The object indicates the direction of te MIP";
                    config "false";
                    type mplstpoamMeDirection;
                }
                leaf meState {
                    description "The object indicates the state of te MIP";
                    config "false";
                    type mplstpoamMeState;
                }
                leaf mipId {
                    description "The object indicates the ID of te MIP";
                    config "true";
                    type uint16 {
                        range "1..8191";
                    }
                }
            }

            container cc {

                leaf ccSessionMode {
                     description "This object indicates the session mode of CC";
                     config "true";
                     default "coordinated";
                         type mplstpoamCcSessionMode;
                }
                leaf ccAuthenticationEnable {
                    config "true";
                    default "true";
                    type Enable;
                }
                leaf ccExp {
                    description "This object indicates the exp of CC packet whic
h is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
                leaf ccTransmitInterval {
                    description "The interval of CC packet which is transmit in
the MEG";
                    config "true";
                    default "1";
                    type mplstpoamCcInterval;
                }
                leaf ccRecieveInterval {
                    description "The interval of CC packet which is recieved in
the MEG";
                    config "true";
                    default "1";
                    type mplstpoamCcInterval;
                }
                leaf ccDetectMultiplier {
                    description "The object indicate the detect multiplier of CC
 packet";
                    config "true";
                    default "3";
                    type mplstpoamCcDetectMultiplier;
                }
                leaf ccEnable {
                    description "The object indicates whether CC can be sent by
the MEG";
                    config "true";
                    default "true";
                    type Enable;
                }
            }
            container cv {
                    leaf cvSessionMode {
                     description "This object indicates the session mode of CC";
                     config "true";
                     default "coordinated";
                         type mplstpoamCcSessionMode;
                }
                leaf cvAuthenticationEnable {
                    config "true";
                    default "true";
                    type Enable;
                }
                leaf cvExp {
                    description "This object indicates the exp of CV packet whic
h is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
                    leaf cvInterval {
                    description "The interval of CV packet which is sent in the
MEG";
                    config "true";
                    default "1";
                    type mplstpoamCvInterval;
                }
                leaf cvDetectMultiplier {
                    description "The object indicate the detect multiplier of CV
 packet";
                    config "true";
                    default "3";
                    type mplstpoamCvDetectMultiplier;
                }
                leaf cvEnable {
                    description "The object indicates whether CC can be received
 by the MEG";
                    config "true";
                    default "true";
                    type Enable;
                }

            }

            container ais {

                config "true";

                leaf aisExp {
                    description "This object indicates the exp of AIS packet whi
ch is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
                leaf aisInterval {
                    description "This object indicates the interval of AIS packe
t which is sent in the MEG";
                    config "true";
                    default "interval1000ms";
                    type mplstpoamAisInterval;
                }
            }

            container lkr {

                config "true";

                leaf lkrExp {
                    description "This object indicates the exp of lock report pa
cket which is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
                leaf lkrInterval {
                    description "This object indicates the interval of lock repo
rt packet which is sent in the MEG";
                    config "true";
                    default "interval1000ms";
                    type mplstpoamLkrInterval;
                }
                leaf lkrEnable {
                    description "The object indicates whether lock report is ena
bled in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
            }
            container oneWayDmSend {


                leaf oneDmSendEnable {
                    description "This object indicates the 1DM statistics is ena
bled in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
                leaf oneDmInterval {
                    description "This object indicates the interval of 1DM stati
stics in the MEG";
                    config "true";
                    default "1000";
                    type mplstpoamDmInterval;
                }
                leaf oneDmExp {
                    description "This object indicates the exp of 1DM packet whi
ch is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
                leaf oneDmPacketSize {
                    description "This object indicates the packet size of 1DM pa
cket which is sent in the MEG";
                    config "true";
                    type uint16 {
                        range "64..1518";
                    }
                }
                leaf oneDmPadValue {
                    description "This object indicates the padding value of 1DM
packet which is sent in the MEG";
                    config "true";
                    default "paddingvalue0";
                    type mplstpoamDmPaddingValue;
                }
            }

            container oneWayDmRcv {


                leaf oneDmRcvEnable {
                    description "This object indicates the 1DM receive is enable
d in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
                leaf oneDmRcvEnableType {
                    description "This object indicates the 1DM receive type";
                    config "true";
                    type mplstpoamOneWayRcvType;
                }
            }

            container twoWayDmSend {

                leaf twoDmSendEnable {
                    description "This object indicates the 2DM statistics is ena
bled in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
                leaf twoDmInterval {
                    description "This object indicates the interval of 2DM stati
stics in the MEG";
                    config "true";
                    default "1000";
                    type mplstpoamDmInterval;
                }
                leaf twoDmExp {
                    description "This object indicates the exp of 2DM packet whi
ch is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
                leaf twoDmPacketSize {
                    description "This object indicates the packet size of 2DM pa
cket which is sent in the MEG";
                    config "true";
                    type uint16 {
                        range "64..1518";
                    }
                }
                leaf twoDmPadValue {
                    description "This object indicates the padding value of 2DM
packet which is sent in the MEG";
                    config "true";
                    default "paddingvalue0";
                    type mplstpoamDmPaddingValue;
                }
                leaf twoDmTimestamp {
                    description "This object indicates whether two-way delay mea
surement time stamp is enable in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
            }

            container twoWayDmRcv {

                leaf twoDmRcvEnable {
                    description "This object indicates the 2DM receiving statist
ics is enabled in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
            }

            container singleLmSend {

                leaf slmSendEnable {
                    description "This object indicates whether slm send is enabl
e in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
                leaf slmInterval {
                    description "This object indicates the interval of slm stati
stics in the MEG";
                    config "true";
                    default "1000";
                    type mplstpoamSlmInterval;
                }
                leaf slmExp {
                    description "This object indicates the exp of slm packet whi
ch is sent in the MEG";
                    config "true";
                    default "7";
                    type uint8 {
                        range "0..7";
                    }
                }
            }

            container singleLmRcv {

                leaf slmRcvEnable {
                    description "This object indicates whether slm receive is en
able in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
            }
            container dualLm {

                leaf dlmEnable {
                    description "This object indicates the dual loss statistics
is enabled in the MEG";
                    config "true";
                    default "false";
                    type Enable;
                }
            }

            container oneWayDmResult {
                config "false";

                leaf sendPktNum {
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf recvPktNum {
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf delayMin {
                    description "This object indicates the minimum delay of rece
ived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf delayMax {
                    description "This object indicates the maximum delay of rece
ived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf delayAvg {
                    description "This object indicates the average delay of rece
ived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf jitterMin {
                    description "This object indicates the minimum jitter of rec
eived LB packets in the MEG";
```

```
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf jitterMax {
                    description "This object indicates the average jitter of rec
eived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf jitterAvg {
                    description "This object indicates the average jitter of rec
eived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                container oneWayDmDatas {
                    config "false";

                    list oneWayDmData {

                        key "index";
                        config "false";

                        leaf index {
                            description "This object indicates index of 1DM stat
istics record in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf oneDelay {
                            description "This object indicates delay of 1DM stat
istics in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf oneDelayVar {
                            description "This object indicates delay Variation o
f 1DM statistics in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf errorInfo {
                            description "This object indicates the error info of
 statistics record in the MEG";
                            config "false";
                            type mplstpoamErrorInfo;
                        }
                    }

                }

            }

            container oneWaySendResult {
                config "false";

                leaf measureMode {
                    description "The flag indicates whether the measurement is a
n on-demand or a continue measurement";
                    config "false";
                    default "on-demand";
                    type mplstpoamMeasureMode;
                }
                leaf status {
                    description "The flag indicates whether the measurement is f
inished";
                    config "false";
                    default "finished";
                    type mplstpoamStatisticsStatus;
                }
            }

            container twoWayDmResult {
                config "false";

                leaf measureMode {
                    description "The flag indicates whether the measurement is a
n on-demand or a continue measurement";
                    config "false";
                    default "on-demand";
                    type mplstpoamMeasureMode;
                }
                leaf status {
                    description "The flag indicates whether the measurement is f
inished";
                    config "false";
                    default "finished";
                    type mplstpoamStatisticsStatus;
                }
                leaf sendPktNum {
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf recvPktNum {
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf delayMin {
                    description "This object indicates the minimum delay of rece
ived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf delayMax {
                    description "This object indicates the maximum delay of rece
ived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf delayAvg {
                    description "This object indicates the average delay of rece
ived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf jitterMin {
                    description "This object indicates the minimum jitter of rec
eived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf jitterMax {
                    description "This object indicates the average jitter of rec
eived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf jitterAvg {
                    description "This object indicates the average jitter of rec
eived LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                container twoWayDmDatas {
                    config "false";

                    list twoWayDmData {

                        key "index";
                        config "false";

                        leaf index {
                            description "This object indicates index of 2DM stat
istics record in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf twoDelay {
                            description "This object indicates delay of 2DM stat
istics in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf twoDelayVar {
                            description "This object indicates delay Variation o
f 2DM statistics in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf errorInfo {
                            description "This object indicates the error info of
 statistics record in the MEG";
                            config "false";
                            type mplstpoamErrorInfo;
                        }
                    }

                }

            }

            container singleLmResult {

                config "false";

                leaf measureMode {
                    description "The flag indicates whether the measurement is a
n on-demand or a continue measurement";
                    config "false";
                    default "on-demand";
                    type mplstpoamMeasureMode;
                }
                leaf status {
                    description "The flag indicates whether the measurement is f
inished";
                    config "false";
                    default "finished";
                    type mplstpoamStatisticsStatus;
                }
                leaf sendPktNum {
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf recvPktNum {
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf rmtLossRatioMin {
                    description "This object indicates the minimum loss-ratio of
 received LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf rmtLossRatioMax {
                    description "This object indicates the maximum loss-ratio of
 received LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf rmtLossRatioAvg {
                    description "This object indicates the average loss-ratio of
 received LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf rmtLossCountMin {
                    description "This object indicates the minimum packet lost o
f received LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf rmtLossCountMax {
                    description "This object indicates the average packet lost o
f received LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                leaf rmtLossCountAvg {
                    description "This object indicates the average packet lost o
f received LB packets in the MEG";
                    config "false";
                    type uint32 {
                        range "1..4294967295";
                    }
                }
                container singleLmDatas {

                    config "false";

                    list singleLmData {

                        key "index";
                        config "false";

                        leaf index {
                            description "This object indicates index of slm stat
istics record in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf slmLossLcl {
                            description "This object indicates local packet loss
 of slm statistics in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf slmLossLclRat {
                            description "This object indicates local packet loss
 rate of slm statistics in the MEG";
                            config "false";
                            type string {
                                length "1..24";
                            }
                        }
                        leaf slmLossRmt {
                            description "This object indicates remote packet los
s of slm statistics in the MEG";
                            config "false";
                            type uint32 {
                                range "1..4294967295";
                            }
                        }
                        leaf slmLossRmtRat {
                            description "This object indicates remote packet los
s rate of slm statistics in the MEG";
                            config "false";
                            type string {
                                length "1..24";
                            }
                        }
                        leaf errorInfo {
                            description "This object indicates the error info of
 statistics record in the MEG";
                            config "false";
                            type mplstpoamErrorInfo;
                        }
                    }

                }

            }

            container dualLmDatas {

                config "false";

                list dualLmData {

                    key "index";
                    config "false";

                    leaf index {
                        description "This object indicates index of dlm statisti
cs record in the MEG";
                        config "false";
                        type uint32 {
                            range "1..4294967295";
                        }
                    }
                    leaf dlmLossLcl {
                        description "This object indicates local packet loss of
dlm statistics in the MEG";
                        config "false";
                        type uint32 {
                            range "1..4294967295";
                        }
                    }
                    leaf dlmLossLclRat {
                        description "This object indicates local packet loss rat
e of dlm statistics in the MEG";
                        config "false";
                        type string {
                            length "1..24";
                        }
                    }
                    leaf dlmLossRmt {
                        description "This object indicates remote packet loss of
 dlm statistics in the MEG";
                        config "false";
                        type uint32 {
                            range "1..4294967295";
                        }
                    }
                    leaf dlmLossRmtRat {
                        description "This object indicates remote packet loss ra
te of dlm statistics in the MEG";
                        config "false";
                        type string {
                            length "1..24";
                        }
                    }
                    leaf errorInfo {
                        description "This object indicates the error info of sta
tistics record in the MEG";
                        config "false";
                        type mplstpoamErrorInfo;
                    }
                }

            }

        }

    }
}
```

6. Security Considerations

TBD

7. IANA Considerations

This document registers the following namespace URI in the IETF XML registry.

URI:TBD

8. Acknowledgements

TBD

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6374] Frost, D. and S. Bryant, "Packet Loss and Delay Measurement for MPLS Networks", RFC 6374, September 2011.

[RFC6375] Frost, D. and S. Bryant, "A Packet Loss and Delay Measurement Profile for MPLS-Based Transport Networks", RFC 6375, September 2011.

[RFC6427] Swallow, G., Fulignoli, A., Vigoureux, M., Boutros, S., and D. Ward, "MPLS Fault Management Operations, Administration, and Maintenance (OAM)", RFC 6427, November 2011.

[RFC6428] Allan, D., Swallow Ed. , G., and J. Drake Ed. , "Proactive Connectivity Verification, Continuity Check, and Remote Defect Indication for the MPLS Transport Profile", RFC 6428, November 2011.

9.2. Infomative References

[RFC6371] Busi, I. and D. Allan, "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks", RFC 6371, September 2011.

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
Huawei Technologies
USA

Email: aldrin.ietf@gmail.com