

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: June 13, 2016

T. Taylor, Ed.  
PT Taylor Consulting  
Y. Zhuang, Ed.  
Huawei  
December 11, 2015

**Applicability of Generic YANG Data Model for layer Independent OAM  
Management  
draft-zhuang-lime-yang-oam-model-applicability-02**

**Abstract**

A generic YANG data model for Operations, Administration, and Maintenance (OAM) has been defined in [[GENYANGOAM](#)], with the intention that technology-specific extensions will be developed to be able reference/use the Generic YANG model. In this document, we describe the applicability of the generic YANG OAM data model to specific OAM technologies. To be concrete, we also demonstrate the usability and extensibility of the generic YANG OAM model with OAM protocols such as IP Ping, traceroute, BFD and MPLS LSP Ping.

**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 June 13, 2016.

**Copyright Notice**

Copyright (c) 2015 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

<a href="#">1.</a>	<a href="#">Introduction . . . . .</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Conventions Used In This Document . . . . .</a>	<a href="#">3</a>
<a href="#">2.1.</a>	<a href="#">Terminology . . . . .</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">Basic Structure of Generic YANG Model for OAM . . . . .</a>	<a href="#">4</a>
<a href="#">3.1.</a>	<a href="#">Performance Management Support . . . . .</a>	<a href="#">6</a>
<a href="#">4.</a>	<a href="#">Guidelines For Extending the LIME Base Data Model . . . . .</a>	<a href="#">6</a>
<a href="#">4.1.</a>	<a href="#">Extend configuration structure with technology specific parameters . . . . .</a>	<a href="#">7</a>
<a href="#">4.1.1.</a>	<a href="#">Maintenance domain (MD) at the root level . . . . .</a>	<a href="#">8</a>
<a href="#">4.1.2.</a>	<a href="#">Maintenance Association (MA) at the second level . . . . .</a>	<a href="#">9</a>
<a href="#">4.1.3.</a>	<a href="#">Maintenance Association Endpoint (MEP) at the third level . . . . .</a>	<a href="#">10</a>
<a href="#">4.1.4.</a>	<a href="#">Session at the fourth level . . . . .</a>	<a href="#">10</a>
<a href="#">4.1.5.</a>	<a href="#">Interface at the fifth level . . . . .</a>	<a href="#">10</a>
<a href="#">4.2.</a>	<a href="#">Extend RPC structure with technology specific parameters . . . . .</a>	<a href="#">10</a>
<a href="#">4.3.</a>	<a href="#">Extend Notification structure with technology specific parameters . . . . .</a>	<a href="#">12</a>
<a href="#">4.4.</a>	<a href="#">Define New RPCs and Notifications . . . . .</a>	<a href="#">12</a>
<a href="#">5.</a>	<a href="#">Applicability of LIME Model to Various Technologies . . . . .</a>	<a href="#">12</a>
<a href="#">5.1.</a>	<a href="#">Generic YANG Model extension for IP OAM . . . . .</a>	<a href="#">13</a>
<a href="#">5.1.1.</a>	<a href="#">MD Configuration Extension . . . . .</a>	<a href="#">13</a>
<a href="#">5.1.2.</a>	<a href="#">MA Configuration Extension . . . . .</a>	<a href="#">13</a>
<a href="#">5.1.3.</a>	<a href="#">MEP Configuration Extension . . . . .</a>	<a href="#">14</a>
<a href="#">5.1.4.</a>	<a href="#">RPC Extension . . . . .</a>	<a href="#">14</a>
<a href="#">5.1.5.</a>	<a href="#">Performance Monitoring Extension . . . . .</a>	<a href="#">15</a>
<a href="#">5.2.</a>	<a href="#">Generic YANG Model extension for TRILL OAM . . . . .</a>	<a href="#">16</a>
<a href="#">5.2.1.</a>	<a href="#">MD Configuration Extension . . . . .</a>	<a href="#">16</a>
<a href="#">5.2.2.</a>	<a href="#">MA Configuration Extension . . . . .</a>	<a href="#">16</a>
<a href="#">5.2.3.</a>	<a href="#">MEP Configuration Extension . . . . .</a>	<a href="#">17</a>
<a href="#">5.2.4.</a>	<a href="#">RPC Extension . . . . .</a>	<a href="#">18</a>
<a href="#">5.2.5.</a>	<a href="#">Performance Management (PM) Extension . . . . .</a>	<a href="#">19</a>
<a href="#">5.2.6.</a>	<a href="#">Usage example . . . . .</a>	<a href="#">19</a>
<a href="#">5.3.</a>	<a href="#">Generic YANG Model extension for MPLS OAM . . . . .</a>	<a href="#">23</a>
<a href="#">5.3.1.</a>	<a href="#">MD Configuration Extension . . . . .</a>	<a href="#">23</a>
<a href="#">5.3.2.</a>	<a href="#">MA Configuration Extension . . . . .</a>	<a href="#">24</a>
<a href="#">5.3.3.</a>	<a href="#">MEP Configuration Extension . . . . .</a>	<a href="#">25</a>
<a href="#">5.3.4.</a>	<a href="#">RPC Extension . . . . .</a>	<a href="#">26</a>
<a href="#">5.3.5.</a>	<a href="#">Performance Management Extension . . . . .</a>	<a href="#">26</a>
<a href="#">5.3.6.</a>	<a href="#">Usage Example . . . . .</a>	<a href="#">27</a>
<a href="#">5.4.</a>	<a href="#">Generic YANG Model extension for MPLS-TP OAM . . . . .</a>	<a href="#">28</a>



5.4.1.	MD Configuration Extension . . . . .	28
5.4.2.	MA Configuration Extension . . . . .	29
5.4.3.	MEP Configuration Extension . . . . .	30
5.4.4.	RPC Extension . . . . .	30
5.4.5.	Performance Monitoring Extension . . . . .	31
5.5.	Generic YANG Model extension for NV03 OAM . . . . .	31
5.5.1.	Technology Type Extension . . . . .	31
5.5.2.	Sub Technology Type Extension . . . . .	32
5.5.3.	MEP Configuration Extension . . . . .	32
5.5.4.	Connectivity-Context Extension . . . . .	33
5.5.5.	RPC Extension . . . . .	33
5.5.6.	ECMP Extension . . . . .	33
5.6.	Generic YANG Model extension for BFD . . . . .	34
5.6.1.	MD Level configuration extension . . . . .	34
5.6.2.	MA configuration extension . . . . .	35
5.6.3.	MEP configuration extension . . . . .	36
6.	Open Issues . . . . .	37
7.	Security Considerations . . . . .	38
8.	IANA Considerations . . . . .	38
9.	Acknowledgements . . . . .	38
10.	References . . . . .	38
10.1.	Normative References . . . . .	38
10.2.	Informative References . . . . .	38
Appendix A.	Contributing Authors Information . . . . .	40
Authors' Addresses	. . . . .	41

## 1. Introduction

The Generic YANG [[RFC6020](#)] over NETCONF [[RFC6241](#)] data model for OAM defined in [[GENYANGOAM](#)], aims at providing consistent configuration, reporting and representation of OAM mechanisms at any layer for any technology.

In this document, we discuss the applicability of the generic YANG OAM model to various OAM technologies and demonstrates that the YANG model(s) developed in the LIME WG are usable and extensible for those technologies. The demonstration uses IP Ping, traceroute, BFD and LSP Ping as specific examples.

## 2. Conventions Used In This Document

This document contains no normative language.

### 2.1. Terminology

MP Maintenance Point [[IEEE802.1Q](#)].



MEP Maintenance association End Point [[RFC7174](#)] [[IEEE802.1Q](#)] [[RFC6371](#)].

MIP Maintenance domain Intermediate Point [[RFC7174](#)] [[IEEE802.1Q](#)] [[RFC6371](#)].

MA Maintenance Association [[IEEE802.1Q](#)] [[RFC7174](#)].

MD Maintenance Domain [[IEEE802.1Q](#)].

OAM Operations, Administration, and Maintenance [[RFC6291](#)].

TRILL Transparent Interconnection of Lots of Links [[RFC6325](#)].

RPC Remote Procedure Call[RFC6020].

### **3. Basic Structure of Generic YANG Model for OAM**

As the basis of this document, the generic YANG model for OAM specified as the LIME base model is shown in Figure 1.



```

module: ietf-gen-oam
  +--rw domains
    +--rw domain* [technology MD-name-string]
      +--rw technology      identityref
      +--rw MD-name-string  MD-name-string
      ...
    +--rw MAs
      +--rw MA* [MA-name-string]
        +--rw MA-name-string  MA-name-string
        ...
      +--rw MEP* [mep-name]
        | +--rw mep-name      MEP-name
        | ...
        | +--rw session* [session-cookie]
        | ...
      +--rw MIP* [interface]
        | +--rw interface    if:interface-ref
      +--rw related-oam-layer* [offset]
      ...

  rpcs:
    +---x continuity-check
    |   ...
    +---x continuity-verification {connectivity-verification}?
    |   ...
    +---x path-discovery
    ...

  notifications:
    +---n defect-condition-notification
    ...

```

Figure 1: Structure of the Generic LIME Base Model

The generic YANG OAM model comprises three definitions for configuration and operational state data:

- o configuration model definition;
- o Remote procedure call (RPC) definition;
- o and notification definition.

The configuration model definition provides hierarchical structure to describe fault domain (i.e., maintenance domain), test point (i.e., maintenance point), technology type, layering, and session context for trouble-shooting. This basic configuration model enables users to select corresponding layers and nodes serving as anchor points to define their specific technology OAM YANG models.





The RPC definition provides uniform APIs for common OAM functions such as continuity check, connectivity verification, path discovery, performance measurement and their equivalents. These APIs are used by the network management system (NMS) to control OAM tools and functionalities on network elements for measuring and monitoring the data plane (e.g., LSP Ping, IP performance measurement protocol) and troubleshooting (e.g., fault localization). These OAM tools activation can be pro-active and on-demand.

The notification definition also provides a uniform API to report defects, faults, and network failures at different layers. This API is used by network elements to report to the network management system (NMS). The content of each notification includes the fault domain and the test point(s) that detected the fault and may generate the error message. This API must be activated proactively.

### **3.1. Performance Management Support**

To support OAM Performance Management, the generic YANG Data Model for OAM needs to be extended by adding loss and delay measurements support with the following model structure:

```
/* MEP Configuration extension */
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP:
  +--rw delay-measurements?
  augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP:
    +--rw loss-measurements?
/* New rpcs */
rpcs:
  +---x create-loss-measurement
  |   ...
  +---x abort-loss-measurement
  |   ...
  +---x create-delay-measurement
  |   ...
  +---x abort-delay-measurement
  |   ...
```

Both pro-active and on-demand loss and delay measurement are supported by augment MEP configuration and RPCs with session type parameter. The details of Performance management extension is specified in the [[I-D.wang-lime-yang-pm](#)]

## **4. Guidelines For Extending the LIME Base Data Model**

YANG allows a module to reference external modules to reuse data already defined in those modules. Therefore a technology-specific model can import data definitions from the LIME base model.



The import statements are used to make definitions available inside other modules [[RFC6020](#)]. Users who want to develop a technology-specific OAM model should import the ietf-gen-oam YANG model with the following statements:

```
module example-ietf-xxx-oam {  
    namespace "urn:foo:params:xml:ns:yang:ietf-xxx-oam";  
    prefix xxxoam;  
  
    import ietf-gen-oam {  
        prefix goam;  
    }  
    .....
```

As described in [Section 3](#), the LIME base model provides a hierarchical structure for configuration, notification and RPCs. Each of these three aspects should be extended with technology-specific features and parameters relating to each technology of interest.

YANG allows a module to insert additional nodes into data models, including both the current module (and its submodules) or an external module. This is useful to let specific technologies add specific parameters into the LIME base model.

Here we summarize four ways to extend the LIME base model for specific technologies:

- o Extend structure for configuration with technology specific parameters
- o Extend structure for notification with technology specific parameters
- o Extend structure for RPC with technology specific parameters
- o Define new RPCs and notifications in the technology specific OAM data model.

#### **[4.1.](#) Extend configuration structure with technology specific parameters**

As described in [[RFC6020](#)], the "augment" statement defines the location in the data model hierarchy where new nodes are inserted.

By using the "augment" statement, the hierarchy of configuration structure can be extended with new data nodes that express technology-specific parameters to meet the requirements of the respective technologies. The technology-specific model developer



must take care to select the right layers and nodes in the configuration structure as anchor points to insert these additional data.

For example, assume a technology- specific OAM YANG model A. An "a" node needs to be inserted within the MA (Maintenance Association):

```
augment /goam:domains/goam:domain/goam:MA/goam:MA:
  +--a?   foo
```

Corresponding YANG encoding:

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA"{
  leaf a{
    type foo
    description
      "foo";
  }
}
```

There are the following five levels in the hierarchy of configuration structure which we can choose as anchor point to insert additional data definitions:

- o Maintenance domain (MD) at the root level;
- o Maintenance Association (MA) at the second level;
- o Maintenance Association Endpoint (MEP) and Maintenance Association Intermediate point(MIP) at the third level;
- o Session at the fourth level;
- o Interface at the fifth level;

#### **4.1.1.1. Maintenance domain (MD) at the root level**

At the Maintenance Domain level, domain data node at root level can be augmented with technology type. [[GENYANGOAM](#)] defines a new globally unique, abstract, and untyped "technology-types" base identity by using the "identity" statement. "identity" and "identityref" are used to Identify New Technology Types. Each technology-specific module then can extend technology type in the base model and specifies a corresponding concrete identity using this base: ipv4, ipv6, trill, mpls, etc.



#### **4.1.2. Maintenance Association (MA) at the second level**

At the Maintenance Association level, an MA data node can be augmented with connectivity context information. For example:

```
+--rw MAs
  +--rw MA* [MA-name-string]
    ...
    +--rw (connectivity-context)?
    | +--:(context-null)
    |   +--rw context-null?          Empty
```

Corresponding YANG encoding:

```
choice connectivity-context {
  default "context-null";
  case context-null {
    description
      "this is a place holder when no context is needed";
    leaf context-null {
      type empty;
      description
        "there is no context defined";
    }
  }
}
description
  "connectivity context";
}
```

ietf-gen-oam YANG model users who want to define a specific OAM technology model can augment the corresponding choice node by defining a new case to carry technology specific extensions.

For example, for a specific OAM technology YANG model A, an "a" node is needed to indicate the connectivity context for this specific OAM technology. To achieve this, it is only necessary to augment the connectivity-context choice node in the ietf-gen-OAM YANG model by defining a "connectivity-context-A" case as:





```
augment /goam:domains/goam:domain/goam:MA/goam:MA
/goam:connectivity-context:
  +--:(connectivity-context-A)
    +--a?   foo
```

Corresponding YANG encoding:

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA"
+ "/goam:connectivity-context" {
  case connectivity-context-A {
    leaf a{
      type foo;}
  }
}
```

In some case when technology type in the Maintenance Domain level is not sufficient to identify OAM technology with different encapsulation method, MA data node can be further augmented with technology sub type (see an example in the [section 5.5](#)).

#### **[4.1.3.](#) Maintenance Association Endpoint (MEP) at the third level**

At the Maintenance Association Endpoint level, a MEP data node can be augmented with connectivity-context information, ECMP information and session information respectively.

#### **[4.1.4.](#) Session at the fourth level**

At the session level, Session data node can be augmented with technology specific information such as Session type, Session interval, etc.

#### **[4.1.5.](#) Interface at the fifth level**

At the interface level under MEP/MIP or under session, the interface data node can be augmented with technology specific information such as context information, interface type, disable/enable button, etc.

### **[4.2.](#) Extend RPC structure with technology specific parameters**

[GENYANGOAM] defines rpc model which abstracts OAM specific commands in a technology independent manner. In this RPC model, three generic RPC commands are specified. By using the "augment" statement, the RPC structure for each OAM command can be extended with new data nodes that express technology-specific OAM command parameters to meet the requirements of the respective technologies. The technology-specific model developer must take care to select the right layers and nodes in the RPC structure as anchor points to insert these additional



data. There are two places which we can choose as anchor point to insert additional data definitions:

- o Input data node

Input data node can be augmented with technology type, sub-command type, session type and other technology specific parameters. Here is an example of sub-command type:

[GENYANGOAM] defines a "command-sub-type" abstract identity for different RPC commands, e.g., to distinguish the types of IP ping [RFC792], LSP ping [RFC4379]. Use of this identity is optional for most cases.

The corresponding statements are shown as below.

```
identity command-sub-type {
  description
    "defines different rpc command subtypes, e.g rfc792 IP
    ping, rfc4379 LSP ping, this is
    optional for most cases";
}

identity icmp-rfc792 {
  base command-sub-type;
  description
    "Defines the command subtypes for ICMPv4 ping";
  reference "RFC 792";
}

identity icmp-rfc4443 {
  base command-sub-type;
  description
    "Defines the command subtypes for ICMPv6 ping";
  reference "RFC 4443";
}

identity icmp-rfc4379 {
  base command-sub-type;
  description
    "Defines the command subtypes for LSP ping";
  reference "RFC 4379";
}
```

- o Output data node

Similarly, output data node can be augmented with technology specific test results information collected by executing OAM command.



### **4.3. Extend Notification structure with technology specific parameters**

[GENYANGOAM] defines one notification model which abstracts defects notification in a technology independent manner. By using the "augment" statement, the notification structure can be extended with new data nodes that express technology-specific notification parameters to meet the requirements of the respective technologies. The technology-specific model developer must take care to select the right layers and nodes in the notification structure as anchor points to insert these additional data.

### **4.4. Define New RPCs and Notifications**

The LIME base model presents three basic RPCs: continuity check, connectivity verification and path discovery. Technology-specific OAM models can either extend the existing RPCs and notifications defined in the LIME base model or define new RPCs and notifications if generic RPCs and notifications cannot be reused to meet their requirements.

For example, a Multicast Tree Verification (MTV) [[TRILLOAMYANG](#)] RPC command is defined in the TRILL OAM model to verify connectivity as well as data-plane and control-plane integrity of TRILL multicast forwarding as follows:

```

RPCs:
+---x mtv
  +--ro input
    | +--ro technology          identityref
    | +--ro MD-name-string      MD-name-string
    | +--ro MA-name-string?     MA-name-string
    | ...
  +--ro output
    +--ro response* [mep-address mep-id]
      +--ro hop-count?          uint8
      +--ro mep-id              tril-rb-nickname
      +--ro mep-address          tril-rb-nickname
      ...

```

## **5. Applicability of LIME Model to Various Technologies**

As mentioned above, the ietf-gen-oam model describes the abstract common core configuration, statistics, RPCs, and notifications for layer independent OAM management.

Following guidelines stated in [Section 4](#), ietf-gen-oam YANG model users can augment this base model by defining and adding new data nodes with technology specific functions and parameters into proper



anchor points of the ietf-gen-oam model, so as to develop a technology-specific OAM model.

With these guidelines in hand, this section further demonstrates the usability of the ietf-gen-oam YANG model to various OAM technologies. Note that, in this section, we only present several snippets of technology-specific data model extensions for illustrative purposes. The complete model extensions should be worked on in respective protocol working groups.

## **5.1. Generic YANG Model extension for IP OAM**

### **5.1.1. MD Configuration Extension**

MD level configuration parameters are management information which can be inherited in the TRILL OAM model and set by LIME base model as default values. For example domain name can be set to area-ID in the IP OAM case. In addition, at the Maintenance Domain level, domain data node at root level can be augmented with technology type.

Note that MD level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures. MD level configuration parameters MUST not be carried using IP Ping and traceroute protocol since IP Ping and traceroute doesn't support transport of these management information.

#### **5.1.1.1. Technology Type Extension**

The technology types ipv4 and ipv6 have already been defined in the LIME base model. Therefore no technology type extension is required in the IP OAM model.

### **5.1.2. MA Configuration Extension**

MA level configuration parameters are management information which can be inherited in the IP OAM model and set by LIME base model as default values. In addition, at the Maintenance Association(MA) level, MA data node at the second level can be augmented with connectivity-context extension.

Note that MA level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures. MA level configuration parameters MUST not be carried using IP Ping and traceroute protocol





since IP Ping and traceroute doesn't support transport of these management information.

#### **5.1.2.1. Connectivity-Context Extension**

In IP OAM, one example of the connectivity-context is a 12 bit VLAN ID. The LIME base model defines a placeholder for connectivity-context. This allows other technologies to easily augment it to include technology specific extensions. The snippet below depicts an example of augmenting context-id to include VLAN ID.

```
augment /goam:domains/goam:domain/goam:MA/goam:MA
/goam:MEP/goam:connectivity-context:
  +--:(context-id-vlan)
    +--rw context-id-vlan?   vlan
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP
/goam:session/goam:connectivity-context:
  +--:(context-id-vlan)
    +--rw context-id-vlan?   vlan
```

#### **5.1.3. MEP Configuration Extension**

MEP configuration in the LIME base model already supports configuring the interface on which the MEP is located with an IP address. There is no additional MEP configuration extension needed for IP OAM.

However, IP Ping, traceroute do not use the MEPID in their message headers. Therefore it is important to have method to derive the MEPID in an automatic manner with no user intervention.

##### **5.1.3.1. ECMP extension**

The flow-entropy parameter in the LIME OAM configuration model is an optional parameter. Since standard IP OAM protocols, e.g., IP Ping and Traceroute, don't support ECMP path selection, the flow-entropy parameter does not need to be supported in the IP OAM model.

#### **5.1.4. RPC Extension**

Technology type in the RPC definition has already been defined in the LIME OAM base model. Therefore no technology type extension is required in the RPC definition. For IP OAM, IP Ping and IP Traceroute RPCs need to be supported. For the IP OAM model, the continuity-check RPC with IPv4 or IPv6 as technology type can be mapped to the IP Ping RPC, while the path-discovery RPC with IPv4 or IPv6 as technology type can be mapped to IP Traceroute.



#### **5.1.5. Performance Monitoring Extension**

Editor Note: IP performance measurement (IPPM) and IP Ping and Traceroute are discussed separately based on the [[RFC7276](#)] classification of OAM technologies. Although IPPM and IP OAM are both applied to the IP network, based on Table 4 of [[RFC7276](#)], IP OAM does not support performance measurement. It is necessary to use OWAMP and TWAMP, defined in IPPM, for that purpose.

##### **5.1.5.1. MEP PM Configuration Extension**

To support IP performance measurement, MEP configuration in the LIME base model can be extended with:

- o loss-stats-group: grouping object for loss measurement session statistics.
- o measurement-timing-group: grouping object used for proactive and on-demand scheduling of PM measurement sessions.
- o delay-measurement-configuration-group: grouping configuration object for the delay measurement function.
- o delay-measurement-stats-group: grouping object for delay measurement session statistics.
- o loss-measurement-configuration-group: grouping configuration object for the loss measurement function.
- o loss-measurement-stats-group: grouping object for loss measurement session statistics.

##### **5.1.5.2. RPC PM Extension**

To support IP performance measurement, it is recommended that four RPCs are defined in the IPPM model:

- o create-loss-measurement RPC: allows scheduling of one-way or two-way on-demand or proactive performance monitoring loss measurement sessions.
- o abort-loss-measurement RPC: allows aborting of currently running or scheduled loss measurement session.
- o create-delay-measurement RPC: allows scheduling of one-way or two-way on-demand or proactive performance monitoring delay measurement sessions.



- o abort-delay-measurement RPC: allows aborting of currently running or scheduled delay measurement sessions.

## **5.2. Generic YANG Model extension for TRILL OAM**

### **5.2.1. MD Configuration Extension**

MD level configuration parameters are management information which can be inherited in the TRILL OAM model and set by LIME base model as default values. For example domain name can be set to area-ID in the TRILL OAM case. In addition, at the Maintenance Domain level, domain data node at root level can be augmented with technology type.

Note that MD level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures.

#### **5.2.1.1. Technology Type Extension**

No TRILL technology type has been defined in the LIME base model. Therefore a technology type extension is required in the TRILL OAM model. The technology type "trill" is defined as an identity that augments the base "technology-types" defined in the LIME base model:

```
identity trill{
  base goam:technology-types;
  description
    "trill type";
}
```

### **5.2.2. MA Configuration Extension**

MA level configuration parameters are management information which can be inherited in the TRILL OAM model and set by LIME base model as default values. In addition, at the Maintenance Association(MA) level, MA data node at the second level can be augmented with connectivity-context extension.

Note that MA level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures.



#### **5.2.2.1. Connectivity-Context Extension**

In TRILL OAM, one example of connectivity-context is either a 12 bit VLAN ID or a 24 bit Fine Grain Label. The LIME base model defines a placeholder for context-id. This allows other technologies to easily augment that to include technology specific extensions. The snippet below depicts an example of augmenting connectivity-context to include either VLAN ID or Fine Grain Label.

```
augment /goam:domains/goam:domain/goam:MA:
/goam:MA /goam:connectivity-context:
  +--:(connectivity-context-vlan)
  |   +--rw connectivity-context-vlan?   vlan
  +--:(connectivity-context-fgl)
  |   +--rw connectivity-context-fgl?    fgl

augment /goam:domains/goam:domain/goam:MA:goam:MA/goam:MEP
/goam:session/goam:connectivity-context:
  +--:(connectivity-context-vlan)
  |   +--rw connectivity-context-vlan?   vlan
  +--:(connectivity-context-fgl)
  |   +--rw connectivity-context-fgl?    fgl
```

#### **5.2.3. MEP Configuration Extension**

The MEP configuration definition in the LIME base model already supports configuring the interface of MEP with either MAC address or IP address. In addition, the MEP address can be represented using a 2 octet RBridge Nickname in TRILL OAM. Hence, the TRILL OAM model augments the MEP configuration in base model to add a nickname case into the MEP address choice node as follows:

```
augment /goam:domains/goam:domain/goam:MA:
/goam:MA/ goam:MEP/goam:mep-address:
  +--:( mep-address-trill)
  |   +--rw mep-address-trill?  trill-rb-nickname
```

In addition, at the Maintenance Association Endpoint(MEP) level, MEP data node at the third level can be augmented with ECMP extension.

##### **5.2.3.1. ECMP Extension**

The flow-entropy parameter in the LIME base model is an optional parameter. Since TRILL supports ECMP path selection, flow-entropy in TRILL is defined as a 96 octet field. The snippet below illustrates its extension.





```

augment /goam:domains/goam:domain/goam:MA/goam:MEP
/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--rw flow-entropy-trill?   flow-entropy-trill
augment /goam:domains/goam:domain/goam:MA/goam:MEP
/goam:session/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--rw flow-entropy-trill?   flow-entropy-trill

```

#### 5.2.4. RPC Extension

In the TRILL OAM YANG model, the continuity-check and path-discovery RPC commands are extended with TRILL specific requirements. The snippet below illustrates the TRILL OAM RPC extension.

```

augment /goam:continuity-check/goam:input:
  +--ro (out-of-band)?
  | +--:(ipv4-address)
  | | +--ro ipv4-address?   inet:ipv4-address
  | +--:(ipv6-address)
  | | +--ro ipv6-address?   inet:ipv6-address
  | +--:(trill-nickname)
  |   +--ro trill-nickname?  trill-rb-nickname
  +--ro diagnostic-vlan?    boolean
augment /goam:continuity-check/goam:input/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--ro flow-entropy-trill?  flow-entropy-trill
augment /goam:continuity-check/goam:output:
  +--ro upstream-rbridge?    trill-rb-nickname
  +--ro next-hop-rbridge*    trill-rb-nickname
augment /goam:path-discovery/goam:input:
  +--ro (out-of-band)?
  | +--:(ipv4-address)
  | | +--ro ipv4-address?   inet:ipv4-address
  | +--:(ipv6-address)
  | | +--ro ipv6-address?   inet:ipv6-address
  | +--:(trill-nickname)
  |   +--ro trill-nickname?  trill-rb-nickname
  +--ro diagnostic-vlan?    boolean
augment /goam:path-discovery/goam:input/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--ro flow-entropy-trill?  flow-entropy-trill
augment /goam:path-discovery/goam:output/goam:response:
  +--ro upstream-rbridge?    trill-rb-nickname
  +--ro next-hop-rbridge*    trill-rb-nickname

```



### **5.2.5. Performance Management (PM) Extension**

#### **5.2.5.1. MEP PM Configuration Extension**

To support performance measurement for TRILL, MEP configuration in the LIME base model can be extended with:

- o loss-stats-group: grouping statistics object for TRILL Loss measurement sessions;
- o measurement-timing-group: grouping object used for proactive and on-demand scheduling of PM measurement sessions;
- o delay-measurement-configuration-group: grouping configuration object for TRILL delay measurement function;
- o delay-measurement-stats-group: grouping statistics object for TRILL delay measurement sessions.

#### **5.2.5.2. RPC PM Extension**

To support performance measurement for TRILL, it is recommended that four new RPCs are defined in the TRILL OAM PM model:

- o create-loss-measurement RPC: allows scheduling of one-way or two-way on-demand or proactive performance monitoring loss measurement sessions.
- o abort-loss-measurement RPC: allows aborting of currently running or scheduled loss measurement sessions.
- o create-delay-measurement RPC: allows scheduling of one-way or two-way on-demand or proactive performance monitoring delay measurement sessions.
- o abort-delay-measurement RPC: allows aborting of currently running or scheduled delay measurement sessions.

#### **5.2.6. Usage example**

This part gives a simple example of implementing the TRILL OAM model onto network devices.

The scenario is shown in Figure 2, in which there are two companies: A and B. Both have departments in City 1 and City 2. Meanwhile, different departments within the same company should be able to communicate with each other. However, the communication services of these two companies should be separated from each other.



To meet the requirements above, two Ethernet Lease line, E-Line-1 and E-Line-2, are set between NE1 and NE3: to isolate the communication traffic between two companies. VLAN 100 associates port 3-EFF8-1 of NE1 facing with company A while VLAN 200 associates port 3-EF8-2 of NE1 facing with company B. For network maintenance, NE1, NE2 and NE3 are within a same maintenance domain: MD1. Two maintenance associations MA1 and MA2 are configured and stand for E-Line-1 and E-Line-2 under MD1. The MAC addresses of NE1, NE2, NE3 are MAC-F001, MAC-F002, MAC-F003 respectively.

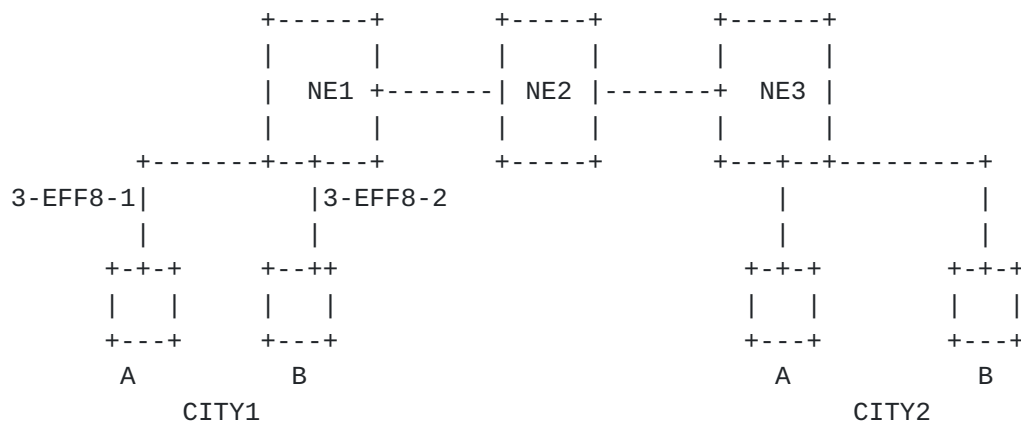


Figure 2: TRILL OAM scenario

#### 5.2.6.1. TRILL OAM Extension

To fulfill the TRILL OAM configuration, the LME base model should be extended by augmenting the connectivity-context and inserting a port node in the MEP list. The snippet below illustrates an example of TRILL OAM model extension.

```
augment /goam:domains/goam:domain/goam:MA:
/goam:MA/goam:MEP /goam:mep-address:
  +--:( mep-address-trill)
  | +--rw mep-address-trill?  trill-rb-nickname
augment /goam:domains/goam:domain/goam:MA/goam:MA
/goam:connectivity-context:
  +--:(connectivity-context-vlan)
  | +--rw connectivity-context-vlan?  vlan
  +--:(connectivity-context-fgl)
  | +--rw connectivity-context-fgl?  fgl

augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP
/goam:session/goam:connectivity-context:
  +--:(connectivity-context-vlan)
  | +--rw connectivity-context-vlan?  vlan
  +--:(connectivity-context-fgl)
```



```

      +--rw connectivity-context-fgl?    fgl
augment /goam:domains/goam:domain/goam:MAS/goam:MA
/goam:MEP/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--rw flow-entropy-trill?    flow-entropy-trill
augment /goam:domains/goam:domain/goam:MAS/goam:MA/goam:MEP
/goam:session/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--rw flow-entropy-trill?    flow-entropy-trill
augment /goam:continuity-check/goam:input:
  +--ro (out-of-band)?
  | +--:(ipv4-address)
  | | +--ro ipv4-address?      inet:ipv4-address
  | +--:(ipv6-address)
  | | +--ro ipv6-address?      inet:ipv6-address
  | +--:(trill-nickname)
  |   +--ro trill-nickname?    trill-rb-nickname
  +--ro diagnostic-vlan?    boolean
augment /goam:continuity-check/goam:input/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--ro flow-entropy-trill?    flow-entropy-trill
augment /goam:continuity-check/goam:output:
  +--ro upstream-rbridge?    trill-rb-nickname
  +--ro next-hop-rbridge*    trill-rb-nickname
augment /goam:path-discovery/goam:input:
  +--ro (out-of-band)?
  | +--:(ipv4-address)
  | | +--ro ipv4-address?      inet:ipv4-address
  | +--:(ipv6-address)
  | | +--ro ipv6-address?      inet:ipv6-address
  | +--:(trill-nickname)
  |   +--ro trill-nickname?    trill-rb-nickname
  +--ro diagnostic-vlan?    boolean
augment /goam:path-discovery/goam:input/goam:flow-entropy:
  +--:(flow-entropy-trill)
    +--ro flow-entropy-trill?    flow-entropy-trill
augment /goam:path-discovery/goam:output/goam:response:
  +--ro upstream-rbridge?    trill-rb-nickname
  +--ro next-hop-rbridge*    trill-rb-nickname

```

#### 5.2.6.2. Corresponding XML Instance Example

This section gives an example of the corresponding XML instance for devices to implement the example TRILL OAM data models in [Section 5.2.6.1](#).

```

<domains>
  <domains>

```





```
<technology> ethernet </techonology>
<MD-name-string> MD1 </MD-name-string>
<MAS>
  <MA>
    <MA-name-string>MA1</MA-name-string>
    <connectivity-context>
      <connectivity-context-vlan>
        100
      </connectivity-context-vlan>
    </connectivity-context>
    <MEP>
      <mep-name>NE1</mep-name>
      <mp-address>
        <mac-address>
          00-1E-4C-84-22-F1
        </mac-address>
      </mp-address>
    </MEP>
    <MEP>

      <mep-name>NE3</mep-name>
      <port>3-EFF8-1</port>
      <mp-address>
        <mac-address>
          00-1E-4C-84-22-F3
        </mac-address>
      </mp-address>
    </MEP>
    <MIP>NE2</MIP>
  </MA>
  <MA>
    <MA-name-string>MA2</MA-name-string>
    <connectivity-context>
      <connectivity-context-vlan>
        200
      </connectivity-context-vlan>
    </connectivity-context>
    <MEP>
      <mep-name>NE1</mep-name>
      <mp-address>
        <mac-address>
          00-1E-4C-84-22-F1
        </mac-address>
      </mp-address>
    </MEP>
    <MEP>
      <mep-name>NE3</mep-name>
      <mp-address>
```



```
    <mac-address>
      00-1E-4C-84-22-F3
    </mac-address>
  </mp-address>
</MEP>
<MIP>NE2</MIP>
</MA>
</MAs>
</domains>
</domains>
```

### **5.3. Generic YANG Model extension for MPLS OAM**

#### **5.3.1. MD Configuration Extension**

MD level configuration parameters are management information which can be inherited in the MPLS OAM model and set by LIME base model as default values. For example domain name can be set to area-ID in the MPLS OAM case. In addition, at the Maintenance Domain level, domain data node at root level can be augmented with technology type and sub-technology type.

Note that MD level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures. MD level configuration parameters MUST not be carried using MPLS OAM protocol(e.g., LSP Ping) since MPLS OAM protocol doesn't support transport of these management information.

##### **5.3.1.1. Technology Type Extension**

No MPLS technology type has been defined in the LIME base model, hence it is required in the MPLS OAM model. The technology type "mpls" is defined as an identity that augments the base "technology-types" defined in the LIME base model:

```
identity mpls{
  base goam:technology-types;
  description
    "mpls type";
}
```

##### **5.3.1.2. Sub Technology Type Extension**

In MPLS, since different encapsulation types such as IP/UDP Encapsulation, PW-ACH encapsulation can be employed, the "technology-sub-type" data node is defined and added into the MPLS OAM model to



further identify the encapsulation types within the MPLS OAM model. Based on it, we also define a technology sub-type for IP/UDP encapsulation and PW-ACH encapsulation. Other Encapsulation types can be defined in the same way.

```
identity technology-sub-type {
    description
        "certain implementations can have different
        encapsulation types such as ip/udp, pw-ach and so on.
        Instead of defining separate models for each
        encapsulation, we define a technology sub-type to
        further identify different encapsulations. Technology
        sub-type is associated at the MA level";
}

identity technology-sub-type-udp {
    base technology-sub-type;
    description
        "technology sub-type is IP/UDP encapsulation";
}

identity technology-sub-type-ach {
    base technology-sub-type;
    description
        "technology sub-type is PW-ACH encapsulation";
}

augment "/goam:domains/goam:domain/goam:MA/goam:MA" {
    leaf technology-sub-type {
        type identityref {
            base technology-sub-type;
        }
    }
}
```

### **5.3.2. MA Configuration Extension**

MA level configuration parameters are management information which can be inherited in the MPLS- OAM model and set by LIME base model as default values. In addition, at the Maintenance Association(MA) level, MA data node at the second level can be augmented with connectivity-context extension.

Note that MA level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures. MA level configuration



parameters MUST not be carried using MPLS OAM protocol(e.g., LSP Ping) since MPLS OAM protocol doesn't support transport of these management information.

#### **5.3.2.1. Connectivity-Context Extension**

In MPLS, one example of context-id is a 20 bit MPLS label. The LIME base model defines a placeholder for context-id. This allows other technologies to easily augment that to include technology specific extensions. The snippet below depicts an example of augmenting context-id to include per VRF MPLS labels in IP VPN or per CE MPLS labels in IP VPN.

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA
/goam:connectivity-context"
{
    case connectivity-context-mpls {
        leaf vrf-label {
            type vrf-label;
        }
    }
}
```

#### **5.3.3. MEP Configuration Extension**

In MPLS, the MEP address is either an IPv4 or IPV6 address in case IP/UDP encapsulation. MEP-ID is either a 2 octet unsigned integer value in case IP/UDP encapsulation or a variable length label value in case of G-ACH encapsulation. In the LIME base model, MEP-ID is defined as a variable length label value and the same definition can be used for MPLS with no further modification. In addition, at the Maintenance Association Endpoint(MEP) level, MEP data node at the third level can be augmented with Session extension and interface extension.

##### **5.3.3.1. ECMP Extension**

Since MPLS supports ECMP path selection, the flow-entropy should be defined in MPLS OAM model. Technology type is used to extend the YANG model to specific usage.





```
augment "/goam:domains/goam:domain/goam:MA/goam:MA
/goam:flow-entropy" {
    case flow-entropy-mpls {
        leaf flags-mpls {
            type flags-mpls;
        }
        leaf flow-entropy-mpls{
            type flow-entropy-mpls;
        }
    }
}
```

#### **5.3.3.2. Per interface Configuration Extension**

TBC.

#### **5.3.4. RPC Extension**

##### **5.3.4.1. CV extension for LSP Ping**

##### **5.3.4.2. Path Discovery Extension for LSP Ping**

##### **5.3.4.3. New RPC Alarm Indication Signal (AIS)**

See [[RFC6427](#)].

##### **5.3.4.4. New RPC for Lock Report (LKR)**

See [[RFC6427](#)].

#### **5.3.5. Performance Management Extension**

##### **5.3.5.1. MEP Configuration Extension**

To support performance monitoring for MPLS, MEP configuration in the LIME base model can be extended with:

- o TBC.

##### **5.3.5.2. RPC Extension**

To support performance monitoring for MPLS, it is recommended that five new RPCs are defined in the MPLS OAM PM model:

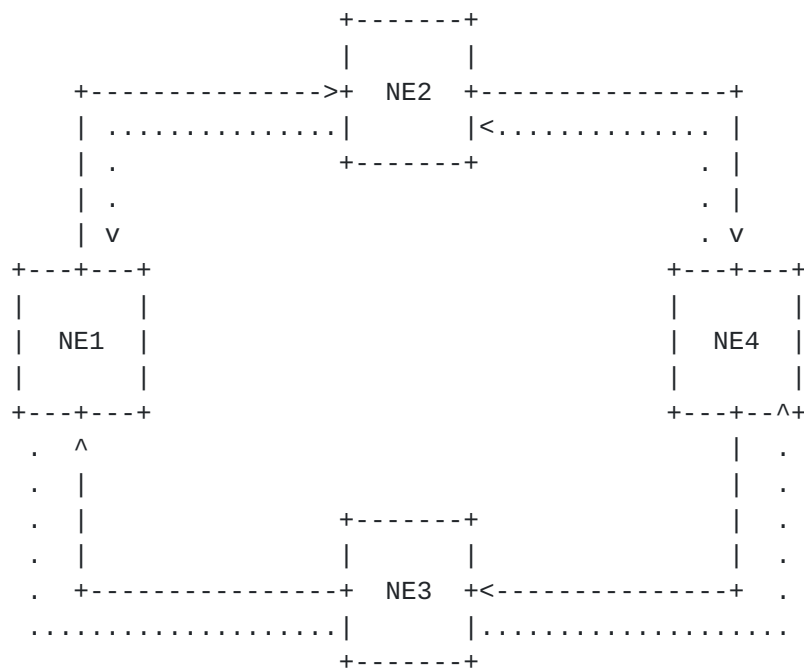
- o MPLS Direct Loss Measurement (DLM) RPC [[RFC6374](#)];
- o MPLS Inferred Loss Measurement (ILM) RPC [[RFC6374](#)];



- o MPLS Delay Measurement (DM) RPC [[RFC6374](#)];
- o MPLS Direct Loss and Delay Measurement RPC [[RFC6374](#)];
- o MPLS Inferred Loss and Delay Measurement RPC [[RFC6374](#)].

### 5.3.6. Usage Example

In the MPLS tunnel scenario (see Figure 3): tunnel\_1 is a static LSP tunnel passing through NE1-NE2-NE4. It is used to perform LSP PING. tunnel\_3 is another static LSP tunnel passing through NE4-NE2-NE1, used to bring back the LSP PING result. tunnel\_2 is a third static LSP tunnel passing through NE1-NE3-NE4, used to perform LSP Traceroute. tunnel\_4 is a fourth static LSP tunnel passing through NE4-NE3-NE1, used to bring back the LSP Traceroute result.



----- forward direction LSP tunnel  
 .....backward direction LSP tunnel

Figure 3: MPLS OAM scenario

#### 5.3.6.1. MPLS OAM Model Extension

TBD.



#### **5.3.6.2. Corresponding XML Instance Example**

TBD.

### **5.4. Generic YANG Model extension for MPLS-TP OAM**

#### **5.4.1. MD Configuration Extension**

MD level configuration parameters are management information which can be inherited in the MPLS-TP OAM model and set by LIME base model as default values. For example domain name can be set to area-ID or the provider's Autonomous System Number (ASN) [[RFC6370](#)] in the MPLS-TP OAM case. In addition, at the Maintenance Domain level, domain data node at root level can be augmented with technology type and sub-technology type.

Note that MD level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures

##### **5.4.1.1. Technology Type Extension**

No MPLS-TP technology type has been defined in the LIME base model, hence it is required in the MPLS OAM model. The technology type "mpls-tp" is defined as an identity that augments the base "technology- types" defined in the LIME base model:

```
identity mpls-tp{
    base goam:technology-types;
    description
        "mpls-tp type";
}
```

##### **5.4.1.2. Sub Technology Type Extension**

In MPLS-TP, since different encapsulation types such as IP/UDP Encapsulation, PW-ACH encapsulation can be employed, the "technology-sub-type" data node is defined and added into the MPLS OAM model to further identify the encapsulation types within the MPLS-TP OAM model. Based on it, we also define a technology sub-type for IP/UDP encapsulation and PW-ACH encapsulation. Other Encapsulation types can be defined in the same way.



```
identity technology-sub-type {
    description
        "certain implementations can have different
        encapsulation types such as ip/udp, pw-ach and so on.
        Instead of defining separate models for each
        encapsulation, we define a technology sub-type to
        further identify different encapsulations. Technology
        sub-type is associated at the MA level";
}

identity technology-sub-type-udp {
    base technology-sub-type;
    description
        "technology sub-type is IP/UDP encapsulation";
}

identity technology-sub-type-ach {
    base technology-sub-type;
    description
        "technology sub-type is PW-ACH encapsulation";
}

augment "/goam:domains/goam:domain/goam:MA/goam:MA" {
    leaf technology-sub-type {
        type identityref {
            base technology-sub-type;
        }
    }
}
```

#### **5.4.2. MA Configuration Extension**

MA level configuration parameters are management information which can be inherited in the MPLS-TP OAM model and set by LIME base model as default values. One example of MA Name is MEG LSP ID or MEG Section ID or MEG PW ID[RFC6370]. In addition, at the Maintenance Association(MA) level, MA data node at the second level can be augmented with connectivity-context extension.

Note that MA level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures.





#### **5.4.2.1. Connectivity-Context Extension**

In MPLS-TP, one example of context-id is a 20 bit MPLS label. The LIME base model defines a placeholder for context-id. This allows other technologies to easily augment that to include technology specific extensions. The snippet below depicts an example of augmenting context-id to include per VRF MPLS labels in IP VPN [[RFC4364](#)] or per CE MPLS labels in IP VPN [[RFC4364](#)].

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA
/goam:connectivity-context"
{
    case connectivity-context-mpls {
        leaf vrf-label {
            type vrf-label;
        }
    }
}
```

#### **5.4.3. MEP Configuration Extension**

In MPLS-TP, MEP-ID is either a variable length label value in case of G-ACH encapsulation or a 2 octet unsigned integer value in case of IP/UDP encapsulation. One example of MEP-ID is MPLS-TP LSP\_MEP\_ID [[RFC6370](#)]. In case of using IP/UDP encapsulation, the MEP address can be either an IPv4 or IPV6 address. In the LIME base model, MEP-ID is defined as a variable length label value and the same definition can be used for MPLS-TP with no further modification. In addition, at the Maintenance Association Endpoint(MEP) level, MEP data node at the third level can be augmented with Session extension and interface extension.

##### **5.4.3.1. ECMP Extension**

The flow-entropy parameter in the LIME OAM configuration model is an optional parameter. Standard MPLS-TP OAM protocol does not support ECMP path selection, so the flow-entropy parameter does not need to be supported in the MPLS-TP OAM model.

##### **5.4.3.2. Per interface Configuration Extension**

TBC.

##### **5.4.4. RPC Extension**



**5.4.4.1. CC extension for MPLS-TP BFD CC Message****5.4.4.2. CV extension for MPLS-TP BFD CV Message****5.4.4.3. CV extension for On-Demand LSP CV with Non-IP Encapsulation****5.4.4.4. CV extension for On-Demand LSP CV with IP Encapsulation****5.4.4.5. New RPC for Remote Defect Indication**

See [[RFC6435](#)].

**5.4.4.6. New RPC for Lock Instruct**

See [[RFC6435](#)].

**5.4.5. Performance Monitoring Extension****5.4.5.1. MEP Configuration Extension**

To support performance monitoring for MPLS-TP, MEP configuration in the LIME base model can be extended with:

- o TBC.

**5.4.5.2. RPC Extension**

To support performance monitoring for MPLS-TP, it is recommended that five new RPCs are defined in the MPLS OAM PM model:

- o MPLS-TP Loss Measurement (LM) Message RPC [[RFC6375](#)];
- o MPLS-TP Test Message RPC [[RFC6375](#)];
- o MPLS-TP Delay Measurement(DM) Message RPC [[RFC6375](#)];

**5.5. Generic YANG Model extension for NV03 OAM****5.5.1. Technology Type Extension**

No NV03 technology type has been defined in the LIME base model. Therefore technology type extension is required in the NV03 OAM model. The technology type "nvo3" is defined as an identity that augments the base "technology-types" defined in the LIME base model:



```
identity nvo3{
  base goam:technology-types;
  description
    "nvo3 type";
}
```

### **5.5.2. Sub Technology Type Extension**

In NV03, since different overlay encapsulation types such as VxLAN, NVGRE can be employed, the "technology-sub-type" data node is defined and added into the NV03 OAM model to further identify the overlay types within the NV03 model. Based on it, we also define a technology sub-type for VxLAN encapsulation. NVGRE and GENEVE, sub-types can be defined in the same way.

```
identity technology-sub-type {
  description
    "certain implementations such as nvo3 can have different
    encapsulation types such as vxlan, nvgre and so on.
    Instead of defining separate models for each
    encapsulation, we define a technology sub-type to
    further identify different encapsulations. Technology
    sub-type is associated at the MA level";
}

identity technology-sub-type-vxlan {
  base technology-sub-type;
  description
    "technology sub-type is vxlan";
}

augment "/goam:domains/goam:domain/goam:MA/goam:MA" {
  leaf technology-sub-type {
    type identityref {
      base technology-sub-type;
    }
  }
}
```

### **5.5.3. MEP Configuration Extension**

In NV03, the MEP address is either an IPv4 or IPV6 address. In the LIME base model, MEP address is defined as an IP address and the same definition can be used for NV03 with no further modification.



#### **5.5.4. Connectivity-Context Extension**

In NV03, one example of context-id is a 24 bit virtual network identifier (VNI). The LIME base model defines a placeholder for context-id. This allows other technologies to easily augment that to include technology specific extensions. The snippet below depicts an example of augmenting context-id to include VNI.

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA
/goam:connectivity-context"
{
    case connctivity-context-nvo3 {
        leaf vni {
            type vni;
        }
    }
}
```

#### **5.5.5. RPC Extension**

In the NV03 OAM YANG model, the End-Station-Locator RPC command is defined. This command locates an end-station within the NV03 deployment. [PTT -- what other tools are applicable??? Presumably one can use ICMP Ping, LSP Ping for CV, and the PM extensions, per [RFC 7276](#) Table 4.]

#### **5.5.6. ECMP Extension**

In NV03, flow-entropy depends on the technology sub-type, e.g., VxLAN. Technology sub-type is used to extend the base model to specific usage. The snippet below illustrates the extension for VxLAN.

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA
/goam:flow-entropy" {
    case flow-entropy-vxlan {
        leaf flags-vxlan {
            type flags-vxlan;
        }
        leaf flow-entropy-vxlan {
            type flow-entropy-vxlan;
        }
    }
}
```





## **5.6. Generic YANG Model extension for BFD**

### **5.6.1. MD Level configuration extension**

MD level configuration parameters are management information which can be inherited in the BFD model and set by LIME base model as default values. For example domain name can be set to area-ID in the BFD case. In addition, at the Maintenance Domain level, domain data node at root level can be augmented with technology type and sub-technology type.

Note that MD level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures. MD level configuration parameters MUST not be carried using BFD protocol since BFD doesn't support transport of these management information.

#### **5.6.1.1. Technology Type Extension**

No BFD technology type has been defined in the LIME base model. Therefore a technology type extension is required in the BFD OAM model. The technology type "bfd" is defined as an identity that augments the base "technology-types" defined in the LIME base model:

#### **5.6.1.2. Sub Technology Type Extension**

In BFD, since different encapsulation types such as IP/UDP Encapsulation, PW-ACH encapsulation can be employed.

In lime-bfd-extension yang data model, we define an identity: "technology-sub-type" to further identify the encapsulation types within the BFD. And based on it, we also define four identity encapsulation types:

- o technology-sub-type-sh-udp: technology sub-type is single hop with IP/UDP encapsulation;
- o technology-sub-type-mh-udp: technology sub-type is multiple hop with IP/UDP encapsulation;
- o technology-sub-type-sh-ach: technology sub-type is single hop with PW-ACH encapsulation;
- o technology-sub-type-mh-ach: technology sub-type is multiple hop with PW-ACH encapsulation;



In MD level, we define a sub-technology leaf with an identityref type which base on the technology-sub-type:

```
augment "/goam:domains/goam:domain/" {  
  leaf sub-technology{  
    type identityref {  
      base technology-sub-type;  
    }  
  }  
}
```

### **5.6.2. MA configuration extension**

MA level configuration parameters are management information which can be inherited in the BFD model and set by LIME base model as default values. In addition, at the Maintenance Association(MA) level, MA data node at the second level can be augmented with connectivity-context extension.

Note that MA level configuration parameters provides context information for management system to correlate faults, defects, network failures with location information, which helps quickly identify root causes of network failures. MA level configuration parameters MUST not be carried using BFD protocol since BFD doesn't support transport of these management information.

#### **5.6.2.1. Connectivity-Context Extension**

In BFD, one example of context-id is a 32bit local discriminator. The LIME base model defines a placeholder for context-id. This allows other technologies to easily augment that to include technology specific extensions. The snippet below depicts an example of augmenting context-id to include local discriminator.

```
augment "/goam:domains/goam:domain/goam:MA/goam:MA  
/goam:connectivity-context"  
{  
  case connectivity-context-bfd {  
    leaf local-discriminator {  
      type local-discriminator;  
    }  
  }  
}
```



### **5.6.3. MEP configuration extension**

In BFD, the MEP address is either an IPv4 or IPV6 address. MEP-ID is either a 2 octet unsigned integer value or a variable length label value. In the LIME base model, MEP-ID is defined as a variable length label value and the same definition can be used for BFD with no further modification. In addition, at the Maintenance Association Endpoint(MEP) level, MEP data node at the third level can be augmented with Session extension and interface extension.

#### **5.6.3.1. Session Configuration Extension**

At the Session level, Session data node at the fourth level can be augmented with 3 interval parameters and 2 TTL parameters. In [\[draft-zheng-bfd-yang\]](#), source and destination address in the bfd-session-cfg can be corresponding to Session configuration extension as source MEP and destination MEP.

```
augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session:
+--rw (interval-config-type)?
|  +--:(tx-rx-intervals)
|  |  +--rw desired-min-tx-interval      uint32
|  |  +--rw required-min-rx-interval     uint32
|  +--:(single-interval)
|      +--rw min-interval                 uint32

augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session:
+--rw tx-ttl?                             ttl
+--rw rx-ttl                             ttl
```

#### **5.6.3.2. Interface configuration extension**

At the Interface level, interface data node at the fifth level can be augmented with the same parameters defined in per-interface configuration of [\[draft-zheng-bfd-yang\]](#).



augment /goam:domains/goam:domain/goam:MA/goam:MA/goam:MEP/goam:session/goam:  
outgoing-interface:

```

+--rw local-multiplier?                multiplier
+--rw (interval-config-type)?
|  +--:(tx-rx-intervals)
|  |  +--rw desired-min-tx-interval    uint32
|  |  +--rw required-min-rx-interval   uint32
|  +--:(single-interval)
|  |  +--rw min-interval               uint32
+--rw demand-enabled?                 boolean
+--rw enable-authentication?          boolean
+--rw authentication-parms {bfd-authentication}?
|  +--rw key-chain-name?              string
|  +--rw algorithm?                   bfd-auth-algorithm
+--rw desired-min-echo-tx-interval?   uint32
+--rw required-min-echo-rx-interval?  uint32

```

#### 5.6.3.3. New Notification definition

[GENYANGOAM] defines a notification model which abstracts defects notification in a technology independent manner. However what BFD is required is state change notification, therefore a new notification definition can be specified to meet BFD requirement.

notifications:

```

+---n state-change-notification
  +--ro local-discriminator?    uint32
  +--ro remote-discriminator?   uint32
  +--ro new-state?              enumeration
  +--ro state-change-reason?     string
  +--ro time-in-previous-state?  string
  +--ro dest-addr?              inet:ip-address
  +--ro source-addr?            inet:ip-address
  +--ro session-cookie?         leafref
  +--ro technology-sub-type?     identityref
  +--ro interface?              leafref
  +--ro echo-enabled?           boolean

```

In this state-change-notification, technology-sub-type is used to identify whether the notification is for single hop or multi-hop or other types.

## 6. Open Issues

Do we need to specify usage examples for each technology-specific OAM model?

Applicability of LIME base model structure on BFD in details





Applicability of LIME base model structure on MPLS OAM and MPLS-TP OAM.

## **7. Security Considerations**

TBD.

## **8. IANA Considerations**

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

URI:TBD

## **9. Acknowledgements**

The authors would like to thank Gregory Mirsky for his valuable comments and suggestions on this document.

## **10. References**

### **10.1. Normative References**

[GENYANGOAM]

Senevirathne , T., Finn, N., Kumar, D., Salam, S., Wu, Q., and Z. Wang, "Generic YANG Data Model for Operations, Administration, and Maintenance (OAM)", ID <https://datatracker.ietf.org/doc/draft-tissa-lime-yang-oam-model/>, June 2015.

[IEEE802.1Q]

"Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks", IEEE Std 802.1Q-2011, August 2011.

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

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", June 2011.

### **10.2. Informative References**



[I-D.wang-lime-yang-pm]

Wang, Z., Wu, Q., Kumar, D., and T. Taylor, "Generic YANG Data Model for Operations, Administration, and Maintenance (OAM) Performance Management", [draft-wang-lime-yang-pm-01](#) (work in progress), November 2015.

[RFC4379] Kompella, K. and G. Swallow, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", [RFC 4379](#), DOI 10.17487/RFC4379, February 2006, <<http://www.rfc-editor.org/info/rfc4379>>.

[RFC6291] Andersson, L., van Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", [BCP 161](#), [RFC 6291](#), DOI 10.17487/RFC6291, June 2011, <<http://www.rfc-editor.org/info/rfc6291>>.

[RFC6325] Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBriges): Base Protocol Specification", [RFC 6325](#), DOI 10.17487/RFC6325, July 2011, <<http://www.rfc-editor.org/info/rfc6325>>.

[RFC6371] Busi, I., Ed. and D. Allan, Ed., "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks", [RFC 6371](#), DOI 10.17487/RFC6371, September 2011, <<http://www.rfc-editor.org/info/rfc6371>>.

[RFC7174] Salam, S., Senevirathne, T., Aldrin, S., and D. Eastlake 3rd, "Transparent Interconnection of Lots of Links (TRILL) Operations, Administration, and Maintenance (OAM) Framework", [RFC 7174](#), DOI 10.17487/RFC7174, May 2014, <<http://www.rfc-editor.org/info/rfc7174>>.

[RFC7276] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", [RFC 7276](#), DOI 10.17487/RFC7276, June 2014, <<http://www.rfc-editor.org/info/rfc7276>>.

[RFC792] Postel, J., "Internet Control Message Protocol", [RFC 792](#), September 1981.

[TRILLOAMYANG]

Kumar, D., Senevirathne, T., Finn, N., Salam, S., Xia, L., and W. Hao, "YANG Data Model for TRILL Operations, Administration, and Maintenance (OAM) (Work in progress)", May 2015.



**Appendix A. Contributing Authors Information**

Huub van Helvoort  
Hai Gaoming BV  
Netherlands

Email: huubatwork@gmail.com

Roland Schott  
Deutsche Telekom  
Deutsche-Telekom-Allee 7  
Darmstadt 64295  
Germany

Email: Roland.Schott@telekom.de

Qin Wu  
Huawei  
101 Software Avenue, Yuhua District  
Nanjing, Jiangsu 210012  
China

Email: bill.wu@huawei.com

Deepak Kumar  
CISCO Systems  
510 McCarthy Blvd  
Milpitas, CA 95035  
USA

Email: dekumar@cisco.com

Yuji Tochio  
Fujitsu  
Japan

Email: tochio@jp.fujitsu.com

Guangying Zheng  
Huawei  
101 Software Avenue, Yuhua District  
Nanjing, Jiangsu 210012  
China

Email: zhengguangying@huawei.com

Daniel King  
Lancaster University



UK

Email: daniel@olddog.co.uk

Zitao Wang  
Huawei Technologies, Co., Ltd  
101 Software Avenue, Yuhua District  
Nanjing 210012  
China

Email: wangzitao@huawei.com

#### Authors' Addresses

Tom Taylor (editor)  
PT Taylor Consulting  
Ottawa  
Canada

Email: tom.taylor.stds@gmail.com

Yan Zhuang (editor)  
Huawei Technologies, Co., Ltd  
101 Software Avenue, Yuhua District  
Nanjing 210012  
China

Email: zhuangyan.zhuang@huawei.com



