

Operations and Management Area Working Group  
Internet-Draft  
Intended status: Informational  
Expires: March 13, 2015

Q. Wu  
M. Wexler  
Huawei  
T. Taylor, Ed.  
PT Taylor Consulting  
September 9, 2014

**Problem Statement for Layer and Technology Independent OAM in a Multi-Layer Environment**  
**draft-edprop-opsawg-multi-layer-oam-ps-00.txt**

**Abstract**

Operations, Administration, and Maintenance (OAM) mechanisms are critical building blocks in network operations. They used for service fulfillment assurance, and for service diagnosis, troubleshooting, and repair. The current practice is that many technologies rely on their own OAM protocols and procedures that are exclusive to a given layer.

At present, there is little consolidation of OAM in the management plane or well-documented inter-layer OAM operation. Vendors and operators dedicate significant resources and effort through the whole OAM life-cycle each time a new technology is introduced. This is exacerbated when dealing with integration of OAM into overlay networks, which require better OAM visibility since there is no method to exchange OAM information between overlay and underlay.

This document analyzes the problem space for multi-layer OAM in the management plane with a focus on layer and technology independent OAM management considerations. It concludes that an attempt to define an architecture for consolidated management should be undertaken, and if this attempt satisfies key objectives, a gap analysis and a program of standardization should follow.

**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 13, 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

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
1.1.	A Vision of Layer and Technology Independent Management .	4
<a href="#">1.2.</a>	Looking Forward . . . . .	<a href="#">5</a>
<a href="#">2.</a>	Terminology . . . . .	<a href="#">5</a>
<a href="#">3.</a>	A Preliminary Set Of Objectives . . . . .	<a href="#">7</a>
<a href="#">4.</a>	Analysis of the Problem . . . . .	<a href="#">8</a>
4.1.	Argument For Consolidated Management . . . . .	8
4.2.	Argument For Layer and Technology Independent Management	9
<a href="#">4.3.</a>	Detailed Issues . . . . .	<a href="#">10</a>
<a href="#">4.3.1.</a>	Strong Technology Dependency For MIB Modules . . . . .	<a href="#">10</a>
<a href="#">4.3.2.</a>	Issues of Abstraction . . . . .	<a href="#">10</a>
<a href="#">4.3.3.</a>	OAM Interworking Issues . . . . .	<a href="#">10</a>
<a href="#">4.3.4.</a>	Multiple (ECMP) Paths OAM Issue . . . . .	<a href="#">11</a>
<a href="#">5.</a>	Problem Statement . . . . .	<a href="#">11</a>
<a href="#">6.</a>	Considerations For the Work On Architecture . . . . .	<a href="#">12</a>
6.1.	Terminology . . . . .	12
<a href="#">6.2.</a>	What the Architecture Must Define . . . . .	<a href="#">13</a>
<a href="#">7.</a>	Security Considerations . . . . .	<a href="#">14</a>
<a href="#">8.</a>	IANA Considerations . . . . .	<a href="#">14</a>
<a href="#">9.</a>	Contributors . . . . .	<a href="#">14</a>
<a href="#">10.</a>	Acknowledgements . . . . .	<a href="#">15</a>
<a href="#">11.</a>	References . . . . .	<a href="#">15</a>
<a href="#">11.1.</a>	Normative References . . . . .	<a href="#">15</a>
<a href="#">11.2.</a>	Informative References . . . . .	<a href="#">15</a>
	Authors' Addresses . . . . .	<a href="#">16</a>



## **1. Introduction**

Operations, Administration, and Maintenance (OAM, [[RFC6291](#)]) mechanisms are critical tools, used for service assurance, fulfillment, or service diagnosis, troubleshooting, and repair, as well as supporting functions such as accounting and security management. The key foundations of OAM and its functional roles in monitoring and diagnosing the behavior of networks have been studied at OSI layers 1, 2 and 3 for many years.

When operating networks with more than one technology in an overlay network, maintenance and troubleshooting are achieved per technology and per layer. As a result, operational processes can be very cumbersome. Stitching together the OAM of adjacent transport segments (as defined in [Section 2](#) in one administrative domain is often not defined and left to proprietary solutions.

Current practice, which consists in enabling specific OAM techniques for each layer, has shown its limits. Concretely, we see today a large number of layer 1/2/3 OAM protocols being well developed and some of them being successfully deployed, but how these OAM protocols in each layer can be applied to overlay networks that are using different encapsulation protocols so as to provide better OAM visibility is still a challenging issue. When no mechanism is defined to exchange performance and liveliness information between the underlay and overlay(s) by a coordination system, it is hard, for instance, to determine whether a fault originates in higher or lower layer.

[Section 1.1 of \[RFC7276\]](#) makes the point that each layer in a multi-layer architecture has its own OAM protocols. From this follows the basic principle that OAM in the data plane cannot cross layer boundaries. A similar constraint holds for boundaries between different transport technologies in the same layer, barring the stitching mentioned above.

One concludes that to simplify OAM and make it more responsive in a multi-layer network requires further consolidation in the management plane. The work on management consolidation would benefit from at least some new standardization. A detailed examination of the potential scope of the work is left for a gap analysis following successful definition of an architecture.

This document further argues that in addition to the ability to retrieve technology specific information from managed entities when following up on problems, consolidated management requires a technology independent view of the network and supporting layers.



How this view is obtained is a key architectural issue outside the scope of the present document.

### **1.1. A Vision of Layer and Technology Independent Management**

What follows is based on the assumption of a network supported by a strict hierarchy of underlying layers in the data plane. There may be multiple layers at a given level of the OSI layer 1-2-3 hierarchy, but that is irrelevant to the vision.

A management application presents to an user a view of this network and its supporting layers that is strictly topological, free of any technology specific information. The user notes a defect along a path serving a particular customer. Looking at the next lower path, the user also sees a defect. Looking the next lower path again, there is also a defect. No lower defect is noted.

At this point it is appropriate to indicate what the user can see along a given path. The path is divided into one or more segments, each spanned by a specific transport technology. However, as already stated, the user does not see any technology specific information. Instead, as well as distinguishing the segments, the user can identify the managed elements at the beginning and end of each segment.

To clarify the situation, the user issues an abstract Continuity Check command, directed toward the initial managed element of the segment in which a fault appears to lie (i.e., in the lowest layer where a defect was observed). By means to be determined by architectural choice, this command is converted into a technology-specific request which is executed across the selected segment. Possible outcomes include:

1. The fault could come clear as a result of the test. The immediate problem is solved (and may have affected multiple upper paths besides the one of initial interest) and the point at which it occurred could be flagged for follow-up maintenance.
2. Local craft action to clear the fault is available in timely fashion.
3. Timely local craft action is not possible, and capacity is reallocated on other paths to ensure that service levels are maintained. Note that capacity reallocation can be done based on the topological view of the network, still on a layer and technology independent basis.



In case (2), technology specific management capabilities are likely to be required by the craftperson following up on the problem.

## **1.2. Looking Forward**

The remainder of this document develops the ideas just stated at a greater level of detail. [Section 2](#) provides terminology that is important to the understanding of the rest of the document. [Section 3](#) establishes preliminary objectives that are key to determining whether a complete program of standardization of consolidated management should be undertaken. [Section 4](#) provides the problem analysis. It is divided into three parts: an argument for consolidated management ([Section 4.1](#)), an argument for layer and technology independent management ([Section 4.2](#)), and an examination of some more detailed issues. [Section 5](#) provides the problem statement, and [Section 6](#) provides some considerations that should be taken into account in the proposed work on architecture.

## **2. Terminology**

[[RFC6291](#)], cited above, provides the official IETF description of Operations, Administration, and Maintenance (OAM) terminology. For a more extensive description of OAM and related terms, see the opening sections, but particularly Sections [2.2.1](#) through [2.2.3](#), of [[RFC7276](#)].

[Section 2.2.4 of \[RFC7276\]](#) introduces the terms data plane, control plane, and management plane.

This document introduces its own interpretation of the following terms, which are in wide use but in that general usage present ambiguities:

Management:

A definition of management can be inferred from [[RFC6123](#)], which in turn refers to [[RFC5706](#)]. Unfortunately the latter chose to divide operations from management, at least from a documentation point of view. The present document chooses to define management as a function that is concerned with all three of operations, administration, and maintenance.

Layer:

The word "layer" has two potential meanings. In the first instance, it is a topological concept, representing a position in a hierarchy of layers. In the second instance, it refers to OSI layers 1, 2 and 3. Within this document, "layer independent OAM





management" as defined below emphasizes the latter meaning when talking about independence, but is intended to extend to all layers of the hierarchy supporting a given network or overlay (the topological view of "layer").

This document makes use of the following additional terms:

Layer independent OAM management:

In a multi-layer network, layer independent OAM management refers to OAM in the management plane that can be deployed independently of media, data protocols, and routing protocols. It denotes the ability to gather OAM information at the different layers, correlate it with layer-specific identifiers and expose it to the management application through a unified interface.

Managed entity:

An architectural concept, an instance of what the management function manages. By definition, a managed entity is capable of communicating with the management function in the management plane.

Local Management Entity (LMgmtE):

An instance of a management function that is restricted in scope to communication with the managed entities associated with a specific transport segment in a specific layer. This term includes legacy management entities in an existing network, and may include entities of a similar scope if they are defined in a consolidated management architecture.

Consolidated Management Entity (CMgmtE):

An instance of the management function that is capable of communicating with all of the LMgmtEs and/or managed entities in a scoped part of the network in order to achieve end-to-end and service-level views of network performance and status and initiate actions when required. The phrase "LMgmtEs and/or managed entities" allows for the possibility that the target architecture allows for direct communication between the CMgmtE and the managed entities or alternatively chooses to assume a distributed management architecture. In any case, as discussed in [Section 6](#), the CMgmtE will have to communicate with legacy LMgmtEs during the transition from the existing to the target architecture.

Management subsystem:



The implementation of the management function in a given network.

Managed device:

A network element associated with at least one technology layer and one managed entity.

Transport segment:

Refers to the portion of a path at a given layer bounded by two points between which a specific transport technology is used and beyond which either a different technology is used or the path is terminated.

Three-dimensional topology:

Refers to a three-dimensional view of the topology of the network and supporting layers. The view of paths along a layer comprises two dimensions. The third dimension is provided by the ordered hierarchy of layers from bottom to top at any point along a path. The three-dimensional topology includes per-path capacity and flow information, permitting layer and technology independent reallocation of capacity as required.

### **3. A Preliminary Set Of Objectives**

Before going further, it is possible to state a preliminary set of objectives for this work. If it does not appear that these can be satisfied, there is no point in undertaking further effort.

As a first objective, the outcome of the work must reduce the time required to respond to and mitigate service-affecting events. The ideal result is that the system be able to do so before the customer notices a service degradation. It is possible that satisfaction of this objective alone is sufficient to carry on.

A second objective relates to the business case for the work and is more difficult for the IETF to judge but crucial for operators attempting to justify changes in their network infrastructure. It should be possible to expect a reduction in life cycle capex and opex as a result of making those changes, even taking account of the potential costs of abandoning or upgrading existing equipment. This objective may influence work on architecture for consolidated management toward minimizing those latter costs (capex). On the positive side, likely savings in craftsperson time implied by the first objective are helpful to the business case (opex).



At a more detailed level, the outcome of the work must allow management to have end-to-end and service-level views of network performance, down to the granularity of service instance. Pre-supposing the arguments made in [Section 4.2](#), it must also allow management to have a layer and technology independent view of the network, at least in the form of the three-dimensional topology, as defined in [Section 2](#).

## **4. Analysis of the Problem**

### **4.1. Argument For Consolidated Management**

Multi-layer OAM actually presents two separate but inter-related issues. The first is technology dependency, at the same or different layers. The second is correlation of events between layers.

OAM mechanisms have a strong technology dependency because each technology (or layer) has its best suited OAM tools. Some of them provide rich functionality with one protocol, while the others provide each function with a different protocol. Today a variety of OAM tools have been developed by different Standards Development Organizations (SDOs) for Optical Transport Network (OTN), Synchronous Digital Hierarchy (SDH), Ethernet, MPLS, and IP networks.

However, orchestrating and coordinating OAM in multi-layer networks to provide better network visibility and efficient OAM operations is still a challenging issue since no mechanisms are defined, for example, to exchange performance and liveliness information between different layers. This means that the required coordination has to happen in the management function through communication with the managed entities.

The development of overlay networks, where one network is the client of another, adds to the magnitude of the problem. To take a specific example, in the Service Function Chaining (SFC) [\[I.D-ietf-sfc-problem-statement\]](#) environment, every Service Function (SF) may operate at a different layer and may use a different encapsulation scheme. When taking into account overlay technologies, the number of encapsulation options increases even more.

At this point, it is useful to recall the preliminary objectives stated in [Section 3](#). To achieve end-to-end and service-level views of network performance requires that the management function be capable of receiving and reacting to related information from every transport segment at every layer in the network. This is a working definition of consolidated management.



A key issue with "management consolidation" is that it may include a requirement for management to interact with every technology used in the network on a per-technology basis either initially or when it has to follow up on detected problems by collecting detailed information. It is an architectural challenge beyond the scope of this document to determine whether consolidated management then becomes an aggregation of local managers of legacy type tied together by a coordination function, or whether simplifications are possible.

#### **4.2. Argument For Layer and Technology Independent Management**

The argument for consolidated management to have a layer and technology independent view of the network and supporting layers is two-pronged. The first argument is fairly straightforward and initially independent of architectural considerations. Some management functions are concerned solely with the topology of the network and supporting layers as represented by the three-dimensional topology defined in [Section 2](#). These include network optimization, efficient enforcement of Traffic Engineering (TE) techniques including assurance of path diversity in one layer and over the complete hierarchy of layers, and fine-grained tweaking. Even in this case management action may require interaction with the managed elements at a technology-specific level, barring an alternative architectural solution.

The second argument for a layer and technology independent view involves considerably more substance than the first one. The three-dimensional topology would be a starting point for this view, but in addition it would include an abstracted view of service-affecting or potentially service-affecting events, identified by layer and reporting managed device. This allows management to correlate events in different layers and identify the devices from which it must seek further information or to which it must direct other requests, without being burdened with excess information. The intention is to ease root cause analysis and improve the ability to maintain end-to-end and service-level visibility.

Where this second version of a technology independent view is created is an architectural issue, beyond the scope of the present document. One possibility is that the work is all done in the "consolidated management" function, in which case the latter just becomes an aggregation of legacy technology-specific managers tied together by a coordination function, as mentioned above. A contrasting possibility is that the managed devices also support the abstraction, with a view to minimizing the amount of technology specific information and management actions the management function has to support.





### **4.3. Detailed Issues**

#### **4.3.1. Strong Technology Dependency For MIB Modules**

OAM protocols rely heavily on the specific network technology they are associated with. For example, ICMPv6 [[RFC4443](#)] and LSP Ping [[RFC4379](#)] provide the same OAM functionality, path discovery, for IPv6 and MPLS Label Switched Path (LSP) technologies respectively.

SNMP MIB modules to manage these protocols were developed on a per OAM protocol basis. As a result, there was little reuse of MIB modules for other existing OAM protocols. To the extent that management operations are being redesigned in terms of YANG modules [[RFC6020](#)] over NETCONF [[RFC6241](#)], the opportunity exists to use the concept of layer and technology independent abstraction to extract the reusable parts, simplifying the work on the remainder.

#### **4.3.2. Issues of Abstraction**

In a multi-layer network, OAM functions are enabled at different layers and OAM information needs to be gathered from various layers independently. Without multi-layer OAM in place, it is hard for management applications to understand what information (e.g., Context, OAM functionalities) at different layers stands for and have a unified view of OAM information at different layers. A mechanism is required to provide this information to management.

The challenge is to abstract in a way that retains in the management plane as much useful information as possible while filtering the data that is not needed. An important part of this effort is a clear understanding of what information is actually needed. There is a close relationship between this issue and the issue already identified in the previous section.

#### **4.3.3. OAM Interworking Issues**

When multiple layer OAMs are used in the different parts of the network, two layer OAMs interworking at the boundaries need to be considered:

- o How one layer OAM in given part of the network interworks with another layer OAM in another part of the network operated by the same administrative entity through a consolidated management interface? e.g., E-LMI used in UNI interworks with Ethernet link OAM used on an IEEE 802.3 link in the same domain?
- o How one layer OAM interworks with another layer OAM in the same part of the network through a consolidated management interface?



e.g., Ethernet OAM interworks with MPLS OAM in the same part of the network? In this case, Ethernet OAM and MPLS OAM are both supported by the same two managed devices in communication.

In these cases, mapping and notifications of defect states between different layer OAMs is required at the boundary nodes of the two parts of the network [[RFC6310](#)] [[RFC7023](#)] [[I-D.ietf-l2vpn-vpws-iw-oam](#)]. Management must provide the interworking function to establish dynamic mapping and translation, supervise defects, and suppress alarms. [Issue for debate. The original text from [draft-ww-oamwg](#) provides for a separate interworking function. To me, that violates the concept of consolidated management. Maybe this is a case of local versus consolidated management as discussed in [Section 6](#) -- PTT as individual contributor]

#### **[4.3.4.](#) Multiple (ECMP) Paths OAM Issue**

Network devices typically use fields in the MAC or IP header or MPLS header and perform hash computations (e.g., 5-tuple hash consisting of IP protocol, source address, destination address, source port, and destination port) on these packet header fields to classify packets into flows and select the forwarding path for the flow among multiple equal cost paths, ECMP becomes more important when network overlay, service chain technology are introduced, e.g., in case of multi-instances of the same service function is invoked for a given chain to provide redundancy, how 5-tuple hash is used based on contents in the outer headers and inner encapsulated packet.

Multiple path OAM requires that Connectivity Check and Continuity Check must follow the same path as the data traffic (e.g., TCP traffic and UDP traffic). Overlay encapsulation allows OAM data to piggyback packets, in the way record route is used in IPv4 options. However, there is no standard way to exercise end to end continuity and connectivity verification that covers all of ECMP paths in the IP networks. Such a standard is desirable.

### **[5.](#) Problem Statement**

OAM functions are used heavily during service and network life-cycle. Today, OAM management requires expertise due to technology dependency despite the similarity in functions (adding to CAPEX and OPEX). Troubleshooting is cumbersome due to protocol variety and lack of multi-layer OAM. This requires expertise and long troubleshooting cycles (OPEX). Last but not least, today's various management interfaces make it difficult to accept and introduce new protocols and technologies



There is value in attempting to define an architecture for consolidated management that may reasonably be argued to meet the objectives stated in [Section 3](#). If this attempt succeeds, it can be followed up with a gap analysis, which in turn will define a further program of standardization.

At the detailed level, [Section 4.3.1](#) and [Section 4.3.2](#) deal with the matter of abstraction and its relationship to the specification of YANG modules. This is work beyond the initial definition of architecture and awaits justification and prioritization by the gap analysis. A similar consideration relates to the solution to the ECMP problem.

The remaining issue is the OAM interworking issue identified in [Section 4.3.3](#). This is architectural in nature, and should be addressed by the proposed work on architecture.

## **6. Considerations For the Work On Architecture**

Definition of an architecture for consolidated management is beyond the scope of the present document. This section instead provides considerations that should be taken into account when defining such an architecture.

### **6.1. Terminology**

The following entities are defined strictly for purposes of the present discussion. This terminology is not meant to restrict other work in any way. This section also uses the term "managed entity" defined in [Section 2](#).

Local Management Entity (LMgmtE):

An instance of a management function that is restricted in scope to communication with the managed entities associated with a specific transport segment in a specific layer. This term includes legacy management entities in an existing network, and may include entities of a similar scope if they are defined in the consolidated management architecture.

Consolidated Management Entity (CMgmtE):

An instance of the management function that is capable of communicating with all of the LMgrtEs and/or managed entities in a scoped part of the network in order to achieve end-to-end and service-level views of network performance and status and initiate actions when required. The phrase "LMgmtEs and/or managed entities" allows for the possibility that the target architecture



allows for direct communication between the CMgmtE and the managed entities or alternatively chooses to assume a distributed management architecture. In any case, as discussed below, the CMgmtE will have to communicate with legacy LMgmtEs during the transition from the existing to the target architecture.

## **6.2. What the Architecture Must Define**

This section is a discussion in the nature of a very general use case rather than a discussion of functions and entities. However, as a preliminary remark, the architecture must be thought through for all five of the FCAPS areas (fault, configuration, accounting, performance, and security management). [RFC 5706 Section 3](#), while nominally directed to protocol design, reviews operational issues associated with each of these areas.

To begin with, previous analysis ([Section 4.2](#)) has indicated that the CMgmtE needs to work with a view of network topology that is layer and technology independent in order to achieve the objectives stated in [Section 3](#). Two questions immediately come to mind: where is this view prepared, taking account of the limited processing power of network devices in particular, and what model is used to present the topology to the CMgmtE? Of course, these questions are evaded if the architecture makes the CMgmtE responsible for creating the abstracted topology from data gathered from the LMgmtEs and/or managed entities within its scope.

Note that from the end-to-end point of view multiple network topologies will typically exist in the network at one time, possibly down to the granularity of a service instance. The relationship of the scope of a CMgmtE to the set of available topologies is subject to the condition that it has end-to-end and service-level views of all paths between the endpoints within its scope, and is otherwise undefined.

The CMgmtE must be aware of all of the LMgmtEs and/or managed entities within its scope. The architecture must define how the CMgmtE identifies the correct sequence of these entities along a path in a given layer, and similarly, must identify the correct ordering of layers from bottom to top. In effect, the CMgmtE requires a three-dimensional topological view of the data plane maintenance infrastructure. Entity identification may be implicit in this work. Note that management actions may alter this topology (e.g., for routine maintenance or installation of new equipment).

The next issue is how the CMgmtE and the other entities discover each other. Bound up in this is the issue of trust. This bootstrapping problem is a hard one, constantly recurring in IETF work but never





yet solved. The architecture work will have to come to its own conclusions on this topic.

Where correlation of events from different layers and transport segments is done is not an issue. By definition it can be done only by the CMgmtE. The architecture must decide whether the necessary data gathering is done as required or continuously.

As a final point, the architecture must specify how an existing network evolves from legacy operation to the target architecture. The existing network will have LMgmtEs in place. The question is whether the CMgmtE simply replaces them or communicates with them. If it simply replaces them, the architecture must define (in an operational considerations section) how testing of the new management configuration takes place before cutover. Considerations of data continuity during cutover should also be addressed.

The above is not an exhaustive list of considerations, but should give a good start to the architectural work.

## **7. Security Considerations**

The architectural work must include work on the security architecture of the whole system. Beyond that, potential future work on individual interfaces must include the appropriate security mechanisms within the architectural framework. The present document cannot be more specific by its nature.

## **8. IANA Considerations**

This document does not require any action from IANA.

## **9. Contributors**

In the understanding of the Editor, the following individuals (listed in alphabetical order by last name) contributed text to or strongly influenced the development of versions of [draft-ww-opsawg-multi-layer-oam](#), from which this document was derived:

- o Sam Aldrin <mailto:aldrin.ietf@gmail.com>, Huawei USA;
- o Mohamed Boucadair <mailto:mohamed.boucadair@orange.com>, France Telecom;
- o Huub van Helvoort <mailto:huubatwork@gmail.com>, Hai Gaoming BV;
- o Tom Herbert <mailto:therbert@google.com>, Google;



- o Pradeep Jain <mailto:pradeep@nuagenetworks.net>, Nuage Networks;
- o Daniel King <mailto:daniel@olddog.co.uk>, Olldog Consulting;
- o Greg Mirsky <mailto:gregory.mirsky@ericsson.com>, Ericsson;
- o Dan Romascanu <mailto:dromasca@avaya.com>, Avaya;
- o Ravi Shekhar <mailto:rshekhar@juniper.net>, Juniper.

## **10. Acknowledgements**

The authors would like to thank Jan Lindblad, Tissa Senevirathne, Yuji Tochio, Ignas Bagdonas, Eric Osborne, Rob Shakir, Georgis Karagiannis, Melinda Shore and Jouni Korhonen for their reviews and suggestions.

## **11. References**

### **11.1. Normative References**

- [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](#), June 2011.
- [RFC7276] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", [RFC 7276](#), June 2014.

### **11.2. Informative References**

- [I-D.ietf-l2vpn-vpws-iw-oam]  
Aissaoui, M., Busschbach, P., Allan, D., Morrow, M., and T. Nadeau, "OAM Procedures for VPWS Interworking", [draft-ietf-l2vpn-vpws-iw-oam-04](#) (work in progress), March 2014.
- [I.D-ietf-sfc-problem-statement]  
Quinn, P., Guichard, J., and S. Surendra, "Network Service Chaining Problem Statement (Work in progress)", ID [draft-ietf-sfc-problem-statement](#), August 2014.
- [RFC4379] Kompella, K. and G. Swallow, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", [RFC 4379](#), February 2006.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", [RFC 4443](#), March 2006.



- [RFC5706] Harrington, D., "Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions", [RFC 5706](#), November 2009.
- [RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), October 2010.
- [RFC6123] Farrel, A., "Inclusion of Manageability Sections in Path Computation Element (PCE) Working Group Drafts", [RFC 6123](#), February 2011.
- [RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", [RFC 6241](#), June 2011.
- [RFC6310] Aissaoui, M., Busschbach, P., Martini, L., Morrow, M., Nadeau, T., and Y(J). Stein, "Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping", [RFC 6310](#), July 2011.
- [RFC7023] Mohan, D., Bitar, N., Sajassi, A., DeLord, S., Niger, P., and R. Qiu, "MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking", [RFC 7023](#), October 2013.

#### Authors' Addresses

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

Email: [bill.wu@huawei.com](mailto:bill.wu@huawei.com)

Mishaël Wexler  
Huawei  
Riesstr. 25  
Munich 80992  
Germany

Email: [mishaël.wexler@huawei.com](mailto:mishaël.wexler@huawei.com)



T. Taylor (editor)  
PT Taylor Consulting  
Ottawa  
Canada

Email: [tom.taylor.stds@gmail.com](mailto:tom.taylor.stds@gmail.com)