



**Working Draft**

**MEF W153 v0.1**

# **LSO IP Schemas and Developer Guide for SOAM**

**January 2025**

**This draft represents MEF work in progress  
and is subject to change.**

EXPORT CONTROL: This document contains technical data. The download, export, reexport or disclosure of the technical data contained in this document may be restricted by applicable U.S. or foreign export laws, regulations and rules and/or applicable U.S. or foreign sanctions ("Export Control Laws or Sanctions"). You agree that you are solely responsible for determining whether any Export Control Laws or Sanctions may apply to your download, export, reexport or disclosure of this document, and for obtaining (if available) any required U.S. or foreign export or reexport licenses and/or other required authorizations.

Disclaimer

© MEF Forum 2025. All Rights Reserved.

The information in this publication is freely available for reproduction and use by any recipient and is believed to be accurate as of its publication date. Such information is subject to change without notice and MEF Forum (MEF) is not responsible for any errors. MEF does not assume responsibility to update or correct any information in this publication. No representation or warranty, expressed or implied, is made by MEF concerning the completeness, accuracy, or applicability of any information contained herein and no liability of any kind shall be assumed by MEF as a result of reliance upon such information.

The information contained herein is intended to be used without modification by the recipient or user of this document. MEF is not responsible or liable for any modifications to this document made by any other party.

The receipt or any use of this document or its contents does not in any way create, by implication or otherwise:

- a) any express or implied license or right to or under any patent, copyright, trademark or trade secret rights held or claimed by any MEF member which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor
- b) any warranty or representation that any MEF members will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor
- c) any form of relationship between any MEF member and the recipient or user of this document.

Implementation or use of specific MEF standards, specifications, or recommendations will be voluntary, and no Member shall be obliged to implement them by virtue of participation in MEF Forum. MEF is a non-profit international organization to enable the development and worldwide adoption of agile, assured and orchestrated network services. MEF does not, expressly or otherwise, endorse or promote any specific products or services.

EXPORT CONTROL: This document contains technical data. The download, export, reexport or disclosure of the technical data contained in this document may be restricted by applicable U.S. or foreign export laws, regulations and rules and/or applicable U.S. or foreign sanctions ("Export Control Laws or Sanctions"). You agree that you are solely responsible for determining whether any Export Control Laws or Sanctions may apply to your download, export, reexport or disclosure of this document, and for obtaining (if available) any required U.S. or foreign export or reexport licenses and/or other required authorizations.

## Table of Contents

<b>1</b>	<b>List of Contributing Members .....</b>	<b>1</b>
<b>2</b>	<b>Abstract.....</b>	<b>2</b>
<b>3</b>	<b>Terminology and Abbreviations .....</b>	<b>4</b>
<b>4</b>	<b>Compliance Levels.....</b>	<b>5</b>
<b>5</b>	<b>Numerical Prefixes .....</b>	<b>6</b>
<b>6</b>	<b>Introduction .....</b>	<b>7</b>
<b>7</b>	<b>Overview of LSO Services .....</b>	<b>9</b>
<b>8</b>	<b>Overview of IP SOAM Model .....</b>	<b>11</b>
<b>9</b>	<b>Overview of IP Performance Monitoring Configuration.....</b>	<b>12</b>
9.1	IpPerformanceMonitoringConfiguration.....	12
<b>10</b>	<b>Overview of IP Performance Monitoring Results.....</b>	<b>13</b>
10.1	IpPerformanceMonitoringConfiguration.....	13
<b>11</b>	<b>Data Model Design Principles and Assumptions .....</b>	<b>14</b>
<b>12</b>	<b>Data Models for IP SOAM.....</b>	<b>15</b>
12.1	Organization and Structure of the Schemas.....	15
12.1.1	Naming Conventions .....	16
<b>13</b>	<b>Relationships Between Entities.....</b>	<b>17</b>
13.1	Performance Monitoring Job Relationships Between Entities .....	17
13.2	Performance Monitoring Result Relationships Between Entities.....	18
<b>14</b>	<b>IP SOAM Configuration and Report Data Models.....</b>	<b>19</b>
14.1	IpPerformanceMonitoringConfiguration.....	19
14.2	IpPerformanceMonitoringResults .....	20
<b>15</b>	<b>Common Classes and Types .....</b>	<b>23</b>
15.1	Ipv4OrIpv6Address .....	23
<b>Appendix A Usage examples (Informative).....</b>		<b>24</b>
A.1	High-level Flow .....	24
A.2	Integration of Service Payload Specific Attributes into the Performance Monitoring API .....	25
A.3	Use Case 1: Create IP Performance Monitoring Configuration .....	27
A.3.1	Create Performance Monitoring Configuration Request for IP Telemetry.....	28

82	A.3.2	Create Performance Monitor Results Response .....	30
83	A.4	Use Case 2: Modify Performance Monitoring Job.....	31
84	A.5	Use Case 3: Register for Performance Monitoring Profile Notifications .....	31
85	A.6	Use Case 4: Retrieve Event Subscription by Identifier .....	31
86	A.7	Use Case 8: Send Notification .....	32
87	16	References .....	33
88			

## List of Figures

89	
90	Figure 1-Performance Monitoring for IP API..... 3
91	Figure 2-LSO Reference Diagram..... 9
92	Figure 3-LSO Performance Monitoring API Structure ..... 10
93	Figure 4- IP SOAM Model Overview..... 11
94	Figure 5-IP Performance Monitoring Configuration Model ..... 12
95	Figure 6-IP Performance Monitoring Result Model ..... 13
96	Figure 7-Schema Files Organization ..... 15
97	Figure 8-Performance Monitoring Job and IP Performance Configuration Associations..... 17
98	Figure 9-Performance Report and IP Performance Configuration Associations ..... 18
99	Figure 10-IpPerformanceMonitoringConfiguration Model..... 19
100	Figure 11-IpPerformanceMonitoringResults Model ..... 21
101	Figure 12-Performance Monitoring End-to-End Function Flow..... 24
102	Figure 13-Extension Pattern: Performance Monitoring Configuration (Job) Extension..... 26
103	Figure 14-Extension Pattern: Performance Monitoring Results Extension ..... 27
104	Figure 15-Performance Monitoring Job progress tracking – Notifications (Asynchronous) ..... 28
105	Figure 16-Performance Monitoring Job progress tracking – Polling (Synchronous)..... 28
106	Figure 17-Performance Monitoring Configuration Request for IP ..... 28
107	Figure 18-IP BFD Performance Configuration JSON Request ..... 29
108	Figure 19-IP Ping Performance Configuration JSON Request..... 29
109	Figure 20-IP TWAMP Performance Configuration JSON Request ..... 30
110	Figure 21-IP Passive Statistics Performance Result JSON Response ..... 30
111	Figure 22-IP BFD Performance Result JSON Response..... 30
112	Figure 23-IP Ping Performance Result JSON Response ..... 30
113	Figure 24-IP TWAMP Performance Result JSON Response ..... 30
114	Figure 25-Performance Monitoring Results Response..... 31
115	
116	

## List of Tables

117	
118	Table 1-Terminology and Abbreviations..... 4
119	Table 2-Numerical Prefix Conventions ..... 6
120	Table 3-IpPerformanceMonitoringConfiguration Attributes..... 20
121	Table 4- IpPerformanceMonitoringResults Attributes ..... 22
122	Table 5- Ipv4OrIpv6Address Attributes ..... 23
123	

## 1 List of Contributing Members

The following members of the MEF participated in the development of this document and have requested to be included in this list.

*Editor Note 1: This list will be finalized before Letter Ballot. Any member that comments in at least one CfC is eligible to be included by opting in before the Letter Ballot is initiated. Note it is the MEF member that is listed here (typically a company or organization), not their individual representatives.*

## 2 Abstract

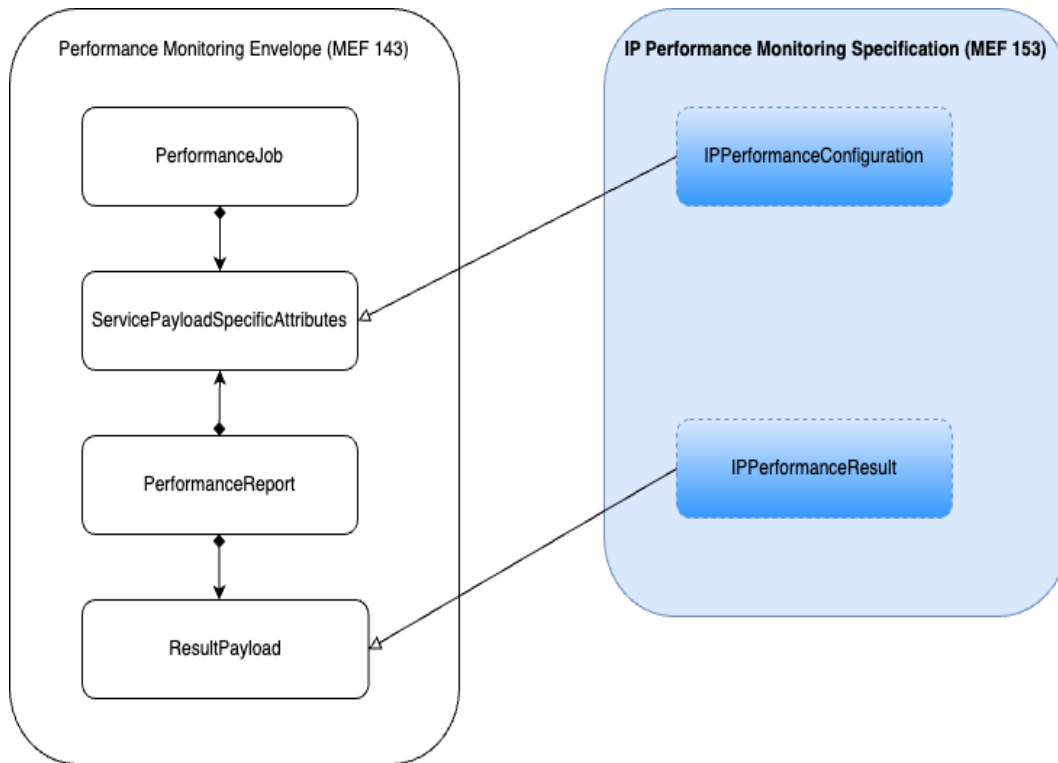
This MEF Standard consisting of this Developer Guide and its associated software artifacts (JSON/YAML Schemas) defines and describes the service-specific payload for the LSO APIs for a set of Performance Monitoring Functions – specifically, Performance Profiles, Performance Job, Performance Notifications and Performance Reporting, for IP Services. It then provides a basic information model for the MEF IP Attributes in support of Performance Profiles, Jobs, Notifications and Reports. The final sections describe the Data Model focused on the JSON/YAML Schemas associated with this specification.

This document can be thought of as a developer's guide for the IP Performance Monitoring Data Models and the schemas supporting the Performance Monitoring of IP Services. IP Performance Profiles, Jobs and Reports are described by a set of Telemetry Attributes. Each Telemetry Attribute describes an aspect of the monitoring and reporting that is agreed between the provider and the user of the service.

This Developer Guide defines the IP performance specifications to be used in combination with MEF 143 LSO Allegro, LSO Interlude and LSO Legato Performance Monitoring API – Developer Guide [7].

Figure 1 represents the relationship between the Performance Monitoring API entities and the IP Service Performance Monitoring specification model. The `ServiceSpecificPayloadAttributes` serves as an extension point for configuring IP service-specific performance monitoring attributes.

`ResultPayload` acts as an extension point for capturing and reporting the outcome of performance monitoring.





151

**Figure 1-Performance Monitoring for IP API**

152

This Standard normatively incorporates the following files by reference as if they were part of this document, from GitHub repository [https://github.com/MEF-GIT/MEF-LSO/tree/develop\\_fm\\_pm](https://github.com/MEF-GIT/MEF-LSO/tree/develop_fm_pm).

153

### 3 Terminology and Abbreviations

This section defines the terms used in this document. In many cases, the normative definitions of terms are found in other documents. In these cases, the third column is used to provide the reference that is controlling, in other MEF or external documents. If the reference includes an asterisk (\*), the definition has been adapted from the original.

Term	Definition	Reference
API Endpoint	The endpoint of a communication channel (the complete URL of an API Resource) to which the HTTP-REST requests are addressed to operation on the API Resource.	MEF W143 [7]
API Resource	A REST Resource. In REST, the primary data representation is called Resource. In this document, API Resource is defined as an OAS SchemaObject with specified API Endpoints.	MEF W143 [7]
Data Model	A representation of concepts of interest to an environment in a form that is dependent on data repository, data definition language, query language, implementation language, and/or protocol (typically, but not necessarily, all five).	IETF RFC 3444 [3]
Notification	A notification is a representation of an event that is exchanged between interested parties. An event is a significant occurrence or change in system state that is important from the perspective of system administration.	MEF W133.1 [6]

**Table 1-Terminology and Abbreviations**

## 4 Compliance Levels

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 (RFC 2119 [2], RFC 8174 [4]) when, and only when, they appear in all capitals, as shown here. All key words must be in bold text.

Items that are **REQUIRED** (contain the words **MUST** or **MUST NOT**) are labeled as **[Rx]** for required. Items that are **RECOMMENDED** (contain the words **SHOULD** or **SHOULD NOT**) are labeled as **[Dx]** for desirable.

Items that are **OPTIONAL** (contain the words **MAY** or **OPTIONAL**) are labeled as **[Ox]** for optional.

## 5 Numerical Prefixes

This document uses the prefix notation to indicate multiplier values as shown in Table 2-Numerical Prefix Conventions.

Decimal		Binary	
Symbol	Value	Symbol	Value
k	$10^3$	Ki	$2^{10}$
M	$10^6$	Mi	$2^{20}$
G	$10^9$	Gi	$2^{30}$
T	$10^{12}$	Ti	$2^{40}$
P	$10^{15}$	Pi	$2^{50}$
E	$10^{18}$	Ei	$2^{60}$
Z	$10^{21}$	Zi	$2^{70}$
Y	$10^{24}$	Yi	$2^{80}$

**Table 2-Numerical Prefix Conventions**

## 6 Introduction

LSO provides programmatic interfaces for establishing automated exchange of information (i.e., Performance Monitoring Profiles, Performance Monitoring Jobs, Performance Monitoring Notifications and Performance Monitoring Results) between a Client and Service Orchestration Function (SOF). These APIs are hierarchically structure. The outer-most structure includes information relating to the access method (e.g., REST), next is information relating to the function being requested and the inner-most structure contains information relating to the specific service, for example IP Performance Monitoring telemetry.

The models defined in this document are intended to be used as payload part of the Performance Monitoring API as defined in MEF W143 [7]. This specification is accompanied by a Data Model for IP-based telemetry instantiated as a set of YAML schemas.

The Data Model for IP Performance Monitoring includes:

- IP Passive Statistics Job Configuration and Collection
- IP Ping Job Configuration and Collection
- IP BFD Job Configuration and Collection
- IP TWAMP Job Configuration and Collection
- Notification Subscription and Events

The document contains the following sections:

An overview of LSO Services (Section 7)

An overview IP SOAM Model (Section 8)

An overview of IP Monitoring Configuration (Section 9)

An overview of IP Monitoring Results (Section 10)

Data Model Design Principles and Assumptions (Section 11)

Data Models for IP SOAM (Section 12)

Relationship between the Entities (Section 13)

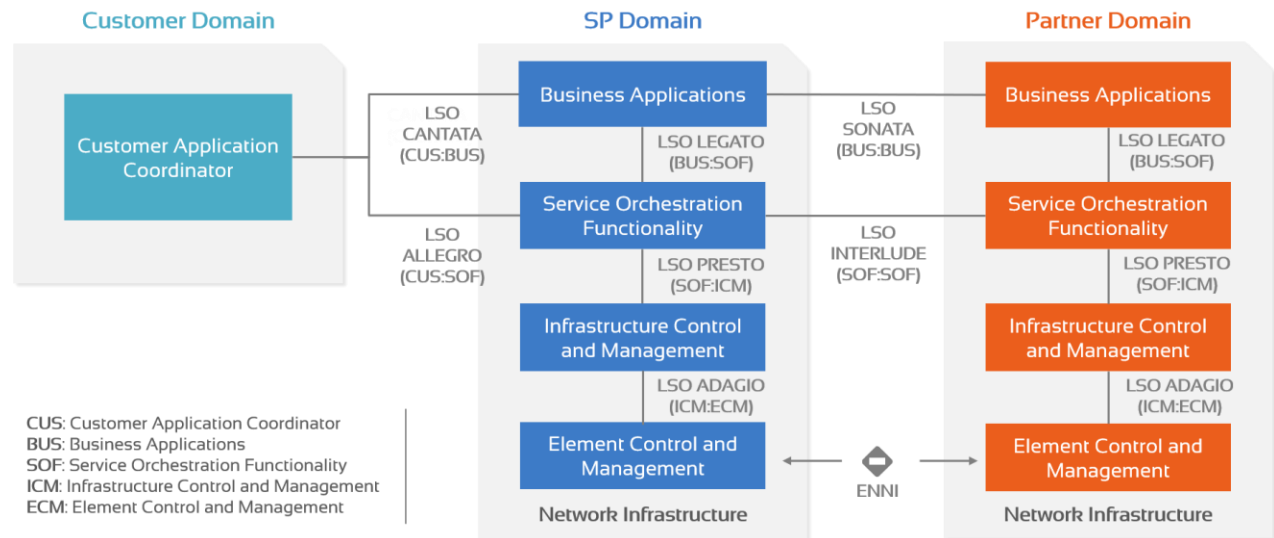
IP SOAM Configuration and Report Data Models (Section 14)

Common Classes and Types (Section 15)

**200 Usage Example (Appendix)**

## 7 Overview of LSO Services

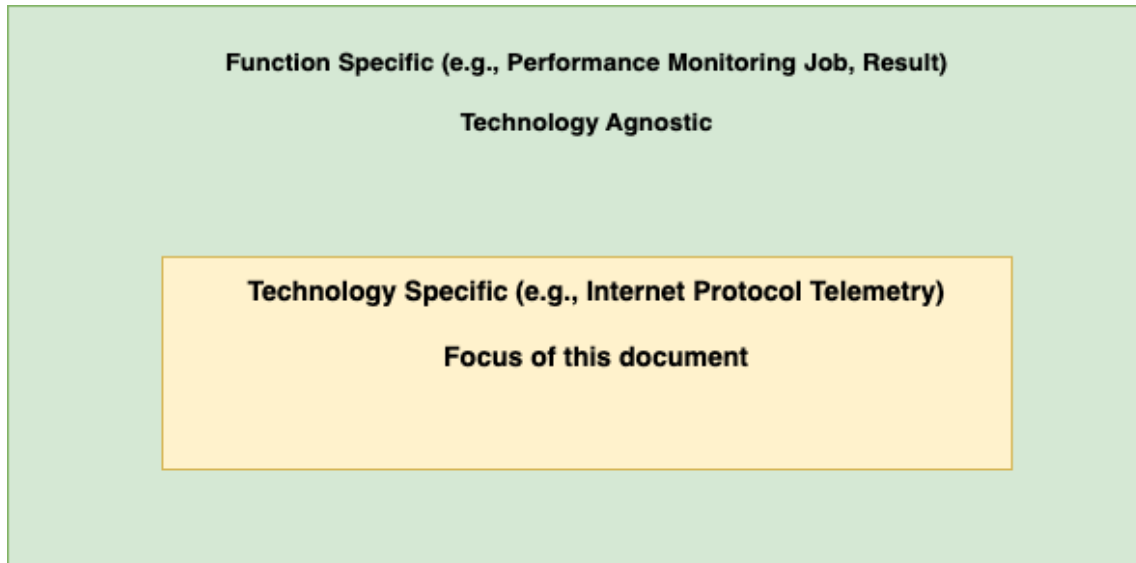
MEF 55.1 [5] describes the Reference Architecture for Lifecycle Service Orchestration (LSO) of MEF-defined connectivity services. MEF 55.1 [5] defines seven LSO Reference Points that are abstract interconnection points between different domains - either within the service provider domain (intra-domain) or between service provider and other business entities (inter-domain).



**Figure 2-LSO Reference Diagram**

The access to automated service provisioning functionality is provided using the Service Provisioning API at multiple LSO Interface Reference Points. LSO provides a suite of APIs for ordering, inventory, etc. which are standardized by MEF as LSO APIs, and which are made available by MEF in a series of releases of the LSO SDKs.

The LSO APIs comprise two parts: one is the service-independent functionality, or Basic API Structure, and the second is the service-specific payload, or Information Payload, as shown in diagram below.



**Figure 3-LSO Performance Monitoring API Structure**

This document defines the technology-specific payload, shown as YAML/JSON Data Model in the figure above, specifically for IP Performance Monitoring applications. The envelope resources of the API and association to specific payload resources will be discussed in detail later in this document.



## 8 Overview of IP SOAM Model

The IP SOAM model has two main classes, `IpPerformanceMonitoringConfiguration` and `IpPerformanceMonitoringResults`. `IpPerformanceMonitoringConfiguration` is responsible for the configuration of the telemetry attributes for an Entity or Service that is to be monitored.

`IpPerformanceMonitoringResults` is responsible for the collected and reporting results of the IP Service or Entity telemetry metrics.

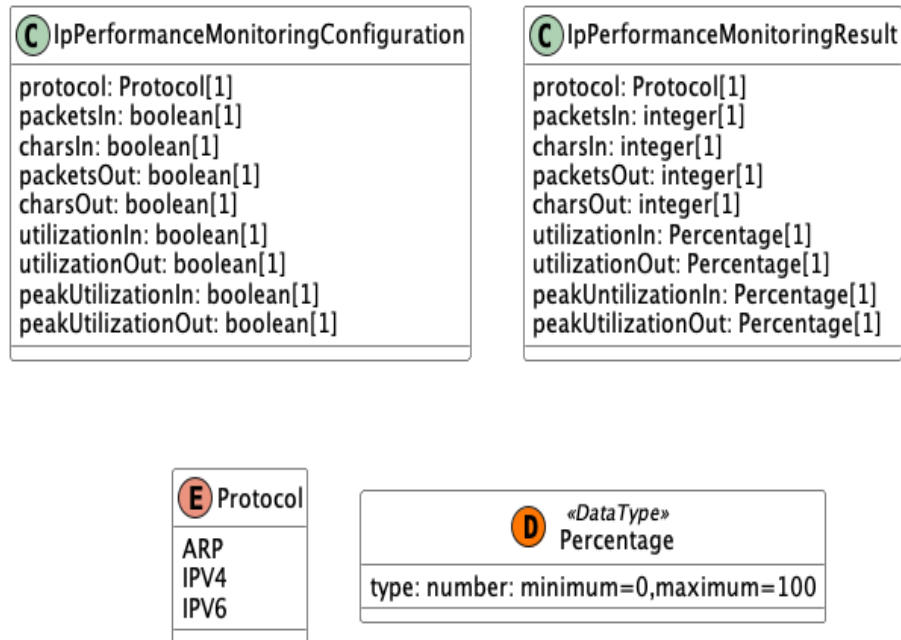


Figure 4- IP SOAM Model Overview

## 9 Overview of IP Performance Monitoring Configuration

This specification describes a data model for MEF-defined IpPerformanceMonitoringConfiguration. An IpPerformanceMonitoringConfiguration is used to select the set of IP specific telemetry attribute that should be monitored and results collected for reporting.

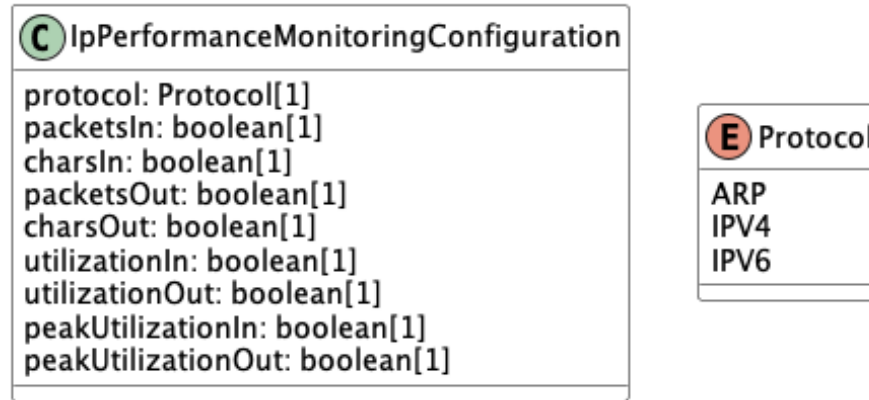


Figure 5-IP Performance Monitoring Configuration Model

### 9.1 IpPerformanceMonitoringConfiguration

An IpPerformanceMonitoringConfiguration has a protocol attribute for specifying which protocol (ARP, IPV4 or IPV6) to collect. The remaining attributes have a Boolean value to specify whether to collect the corresponding telemetry.

## 10 Overview of IP Performance Monitoring Results

This specification describes a data model for MEF-defined IpPerformanceMonitoringResults. An IpPerformanceMonitoringResults is used to report the set of IP specific telemetry attributes that have been monitored and results collected for reporting.

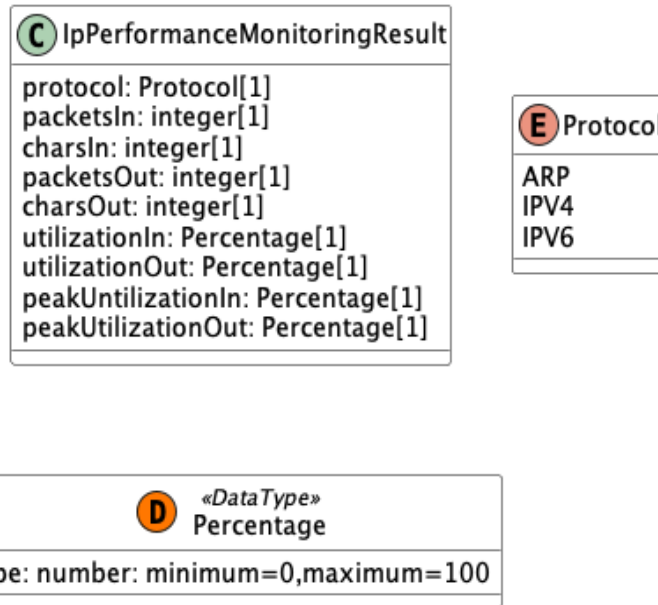


Figure 6-IP Performance Monitoring Result Model

### 10.1 IpPerformanceMonitoringConfiguration

An IpPerformanceMonitoringConfiguration has a protocol attribute for specifying which protocol (ARP, IPv4 or IPv6) to collect. The remaining attributes are telemetry attributes with IP report values.

## 11 Data Model Design Principles and Assumptions

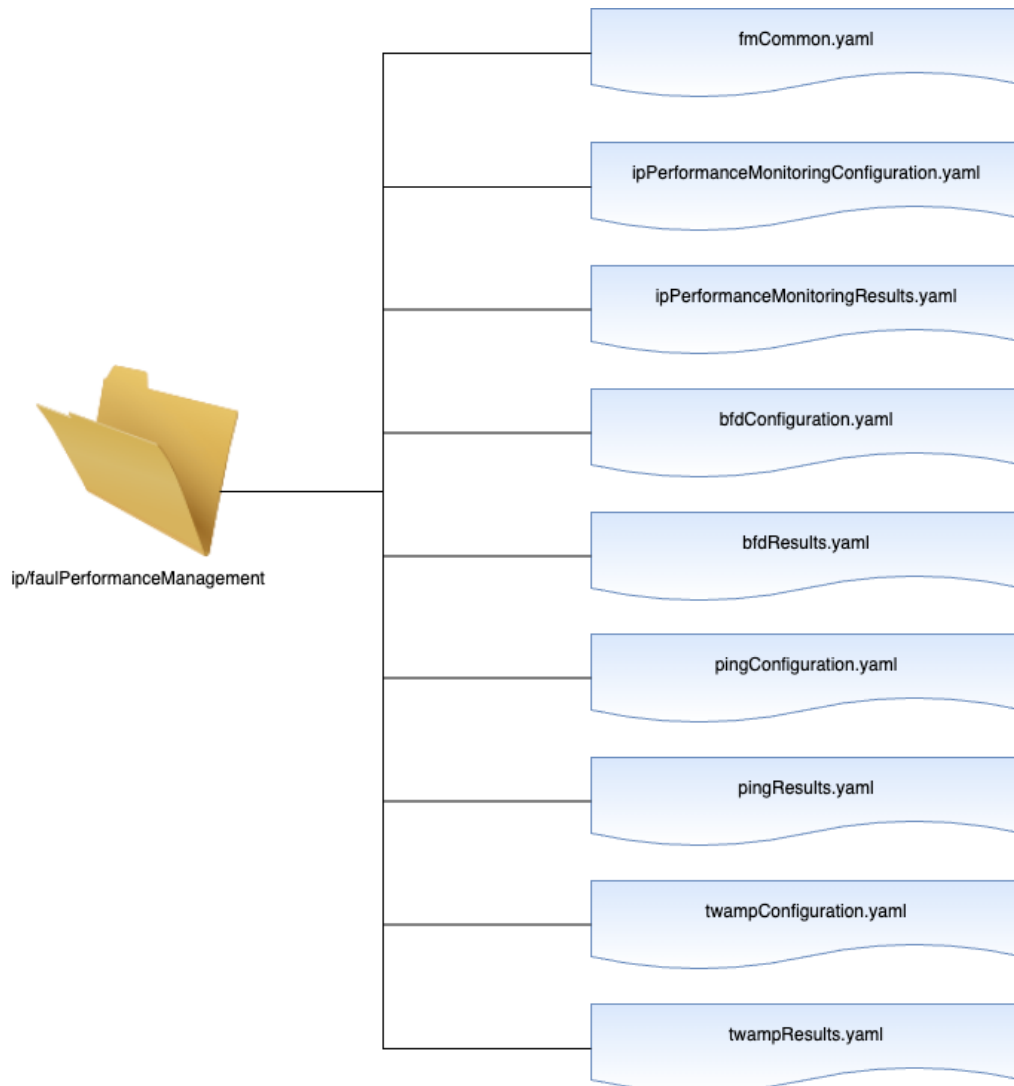
A PM Attribute for a configuration or result can have a value that is a simple datatype such as an integer or string (or list of simple datatypes) or a value that is an object with multiple properties or a composition of objects. Within this document each simple value (integer, string, Boolean, etc.) is referred to as a Technology-Specific Attribute. A Technology-Specific Attribute could be an Attribute, or it could be a parameter within an Attribute.

## 12 Data Models for IP SOAM

The data models for the IP SOAM configuration are expressed as a set of JSON schemas based on JSON schema draft 7 and encoded in YAML. These schemas accompany this document. This section explains the organization and structure of these schemas.

### 12.1 Organization and Structure of the Schemas

The schemas are organized into a file structure as shown in Figure 7.



**Figure 7-Schema Files Organization**

Both ipPerformanceMonitoring and ipPerformanceMonitoringResults schemas are in the same directory. One file that provides common resources that are shared is:

- `performanceMonitoring/fmCommon.yaml` – provides classes shared among all Performance Monitoring technologies for SOAM.

These common classes are referenced in the relevant service component schema files. For example, the **TimeDuration** attribute specified in `fmCommon.yaml` file refers to common definition:

```
TimeDuration:
  type: object
  description: >-
    This class is used to describe durations expressed as a 2-tuple, (value,
    units). The units from nanoseconds to years.
  properties:
    timeDurationValue:
      description: The value of the duration. For example, if the duration is 20
ms, this element is 20.
      type: integer
    timeDurationUnits:
      $ref: '#/definitions/TimeDurationUnits'
```

### 12.1.1 Naming Conventions

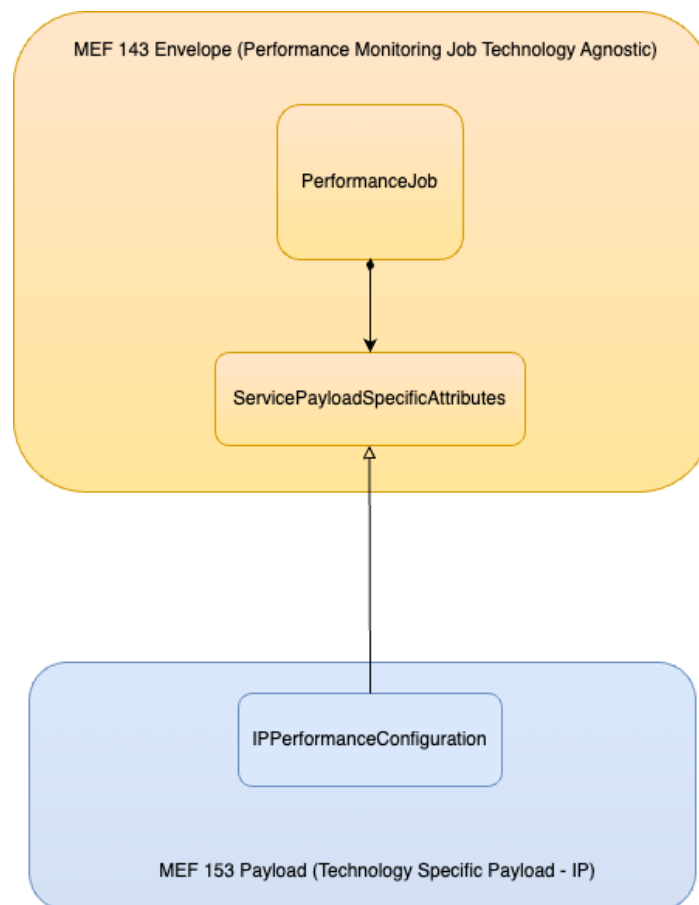
In the schemas, class and type names are UpperCamelCase and Service Attribute/property names are lowerCamelCase.

## 13 Relationships Between Entities

This section describes the constraints and relationships between the envelope operation specific, technology agnostic PerformanceJob and PerformanceReport Items with payload technology specific IPPerformanceConfiguration and IPPerformanceReport respectively. The PerformanceJob and PerformanceReport resources are defined in [7].

### 13.1 Performance Monitoring Job Relationships Between Entities

The Performance Monitoring API is a generic envelope that is technology (i.e., CE, IP, SD-WAN) agnostic. The envelope API itself does not provide explicit definitions of specific performance monitoring resources. The relationships between the envelope and configuration payload components are shown in Figure 8.



**Figure 8-Performance Monitoring Job and IP Performance Configuration Associations**

**[R1]** `ServicePayloadSpecificAttributes` types are extension points the **MUST** be used to integrate IP service performance monitoring properties into a request/response payload.

[R2] The `@type` property of `ServicePayloadSpecificAttributes` MUST be used to specify the type of the extending entity.

[R3] Attributes specified in the payload MUST conform to the performance definition specified in the `@type` property.

## 13.2 Performance Monitoring Result Relationships Between Entities

The Performance Monitoring API is a generic envelope that is technology (i.e., CE, IP, SD-WAN) agnostic. The envelope API itself does not provide explicit definitions of specific performance result resources. The relationships between the envelope and results payload components are shown in Figure 9.

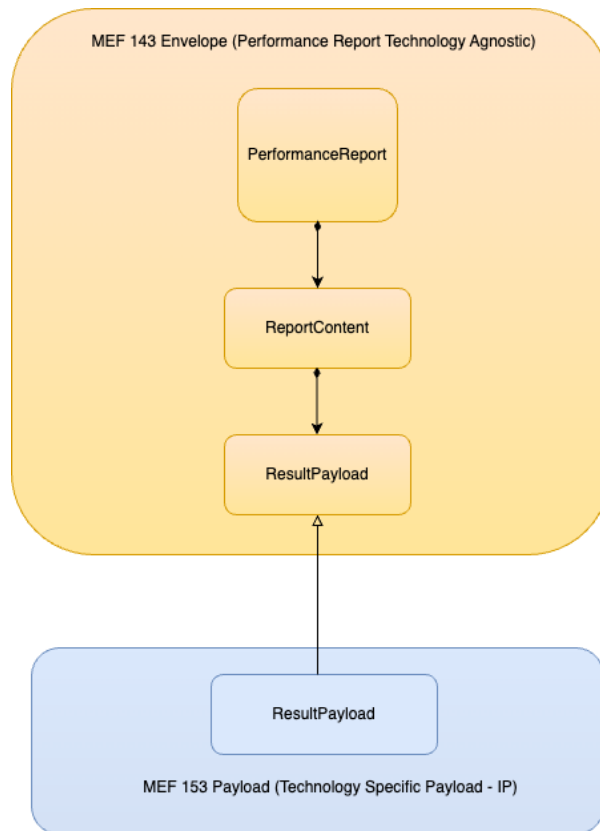


Figure 9-Performance Report and IP Performance Configuration Associations

[R4] `ResultPayload` types are extension points the **MUST** be used to integrate IP service performance monitoring properties into a request/response payload.

[R5] The `@type` property of `ResultPayload` **MUST** be used to specify the type of the extending entity.

[R6] Attributes specified in the payload **MUST** conform to the performance definition specified in the `@type` property.



## 14 IP SOAM Configuration and Report Data Models

An IP SOAM Configuration Job is used to set up the telemetry measurements. The IpPerformanceMonitoringConfiguration is the payload used to select the telemetry metrics to monitor. The PerformanceJob is the envelope resource that has an attribute for selecting the technology-specific payload. ServicePayloadSpecificAttributes is subclassed to the IpPerformanceMonitoringConfiguration.

An IP SOAM Report is used to provide the collected set of telemetry metrics that were initiated by the IP SOAM Configuration Job. The PerformanceReport is the envelope resource that has an attribute for the collection. ReportContent has multiple records using the technology specific payload, IpPerformanceMonitoringResults which is a sub-class of ResultPayload.

### 14.1 IpPerformanceMonitoringConfiguration

The IpPerformanceMonitoringConfiguration is the ServicePayloadSpecificAttributes extension for a PerformanceJob used to performance monitor IP telemetry.

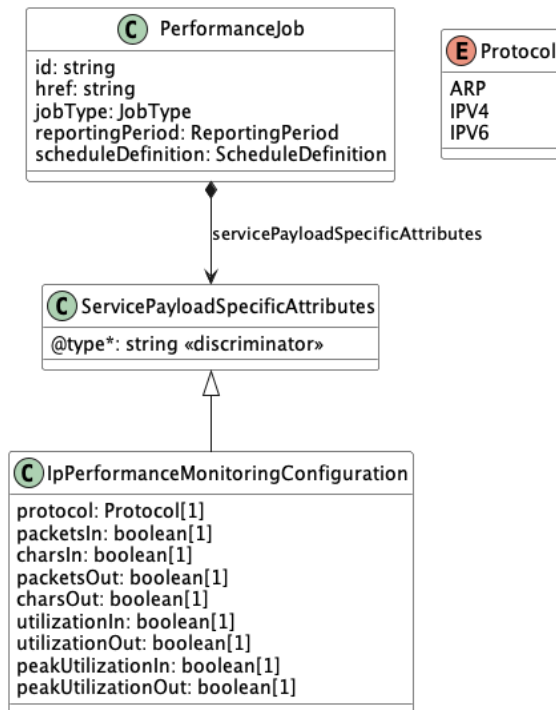


Figure 10-IpPerformanceMonitoringConfiguration Model

Schema File Name: ip/faultPerformanceManagement/ipPerformanceMonitoringConfiguration.yaml			
\$id: urn:mef:iso:spec:service:ip-performance-monitoring-configuration:v0.0.2:all			
Attribute Name	Type	Multiplicity	Description
protocol	String Enum:	1	Protocol type.

<b>Schema File Name:</b> <b>ip/faultPerformanceManagement/ipPerformanceMonitoringConfiguration.yaml</b> <b>\$id: urn:mef:iso:spec:service:ip-performance-monitoring-configuration:v0.0.2:all</b>			
	<ul style="list-style-type: none"> <li>IPv4</li> <li>IPv6</li> <li>ARP</li> </ul>		
packetsIn	boolean	1	Enable or disable the collection of input IP packets on monitored interface.
charsIn	boolean	1	Enable or disable the collection of input IP characters on monitored interface.
packetsOut	boolean	1	Enable or disable the collection of output IP packets on monitored interface.
charsOut	boolean	1	Enable or disable the collection of output IP characters on monitored interface.
utilizationIn	boolean	1	Enable or disable the collection of input bandwidth utilization on monitored interface.
utilizationOut	boolean	1	Enable or disable the collection of input bandwidth utilization on monitored interface.
peakUtilizationIn	boolean	1	Enable or disable the collection of input peak bandwidth utilization on monitored interface.
peakUtilizationOut	boolean	1	Enable or disable the collection of output peak bandwidth utilization on monitored interface.

**Table 3-IpPerformanceMonitoringConfiguration Attributes**

*NOTE: The interface that is being monitored for telemetry is provided in the envelope part of API (i.e., MEF 143).*

## 14.2 IpPerformanceMonitoringResults

The IpPerformanceMonitoringResults is the ResultPayload extension for a PerformanceReport used to report the results of the monitored IP telemetry.

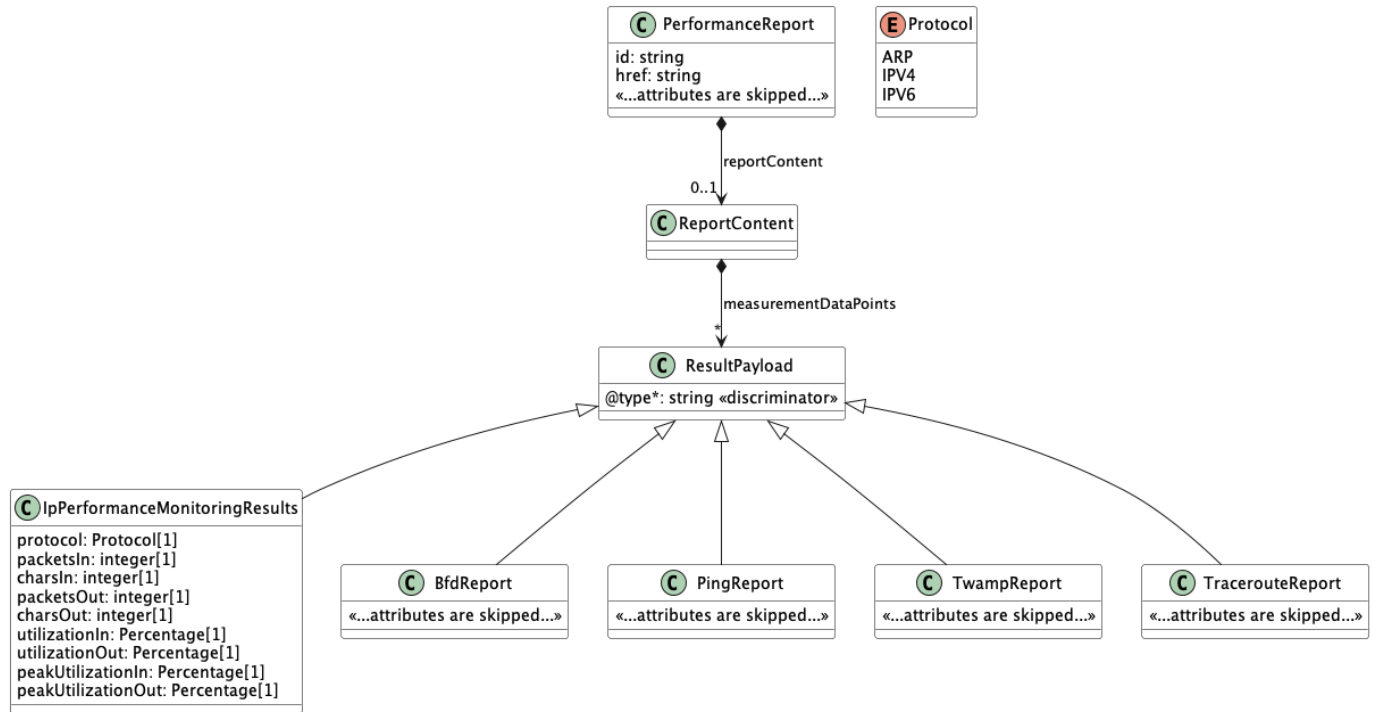


Figure 11-IpPerformanceMonitoringResults Model

Schema File Name: ip/faultPerformanceManagement/ipPerformanceMonitoringResults.yaml \$id: urn:mef:iso:spec:service:ip-performance-monitoring-results:v0.0.2:all			
Attribute Name	Type	Multiplicity	Description
protocol	String Enum: • IPv4 • IPv6 • ARP	1	Protocol type.
packetsIn	integer	1	Packets measured in input interface.
charsIn	integer	1	Characters measured on input interface.
packetsOut	integer	1	Packets measured on output interface.
charsOut	integer	1	Characters measured on output interface.
utilizationIn	Percentage	1	Input interface utilization.
utilizationOut	Percentage	1	Output interface utilization.
peakUtilizationIn	Percentage	1	Input interface peak utilization.
peakUtilizationOut	Percentage	1	Output interface peak utilization.

**Table 4- IpPerformanceMonitoringResults Attributes**

## 15 Common Classes and Types

This section is structure like the previous section but focuses on common classes and types used by the IP Configuration and Result Attributes.

### 15.1 Ipv4OrIpv6Address

Data type representing IPv4 or IPv6 address.

Schema File Name: ip/faultPerformanceManagement/fmCommon.yaml			
Attribute Name	Type	Multiplicity	Description
ipv4	Ipv4Address	0..*	IPv4 address.
ipv6	Ipv6Address	0..*	Ipv6 address.

Table 5- Ipv4OrIpv6Address Attributes

## Appendix A Usage examples (Informative)

This appendix aims to provide an extensive set of examples to cover:

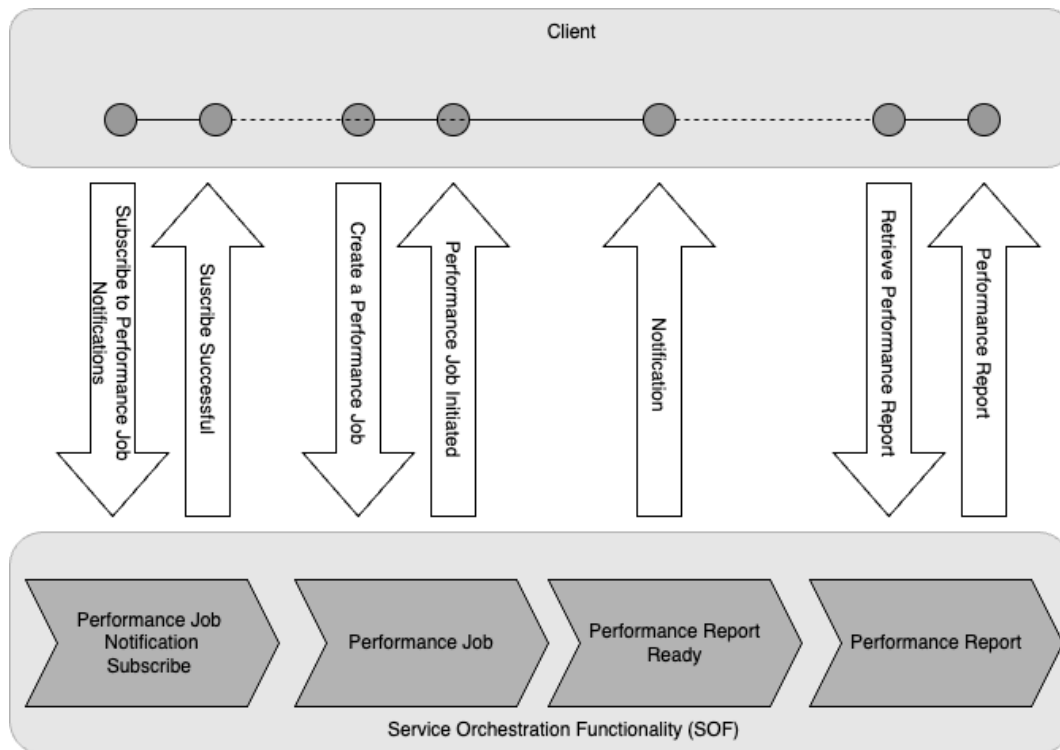
- Performance Monitoring Profile and Performance Monitoring Job configuration variants,
- Retrieval of Performance Monitoring Profile and Performance Monitoring Results.
- Notifications
- Deletion of Performance Monitoring Profile and Performance Monitoring Job.

The examples are delivered in two forms:

- As part of this document – to allow comments and rich explanation.
- As a Postman collection – for ease of use in testing.

### A.1 High-level Flow

The Interface Reference Points each form a set of APIs that service different functions in the end-to-end flow.



**Figure 12-Performance Monitoring End-to-End Function Flow**

Performance Job Notification Subscribe – allows the Client to subscribe to Performance Job Notifications.

Performance Job – allows the Client to create a Performance Job with the technology-specific performance telemetry metrics to measure and collect.

Performance Report Ready Notification – an asynchronous notification generated and sent by the Server to the Client to inform that the Performance Report is ready.

Performance Report – allows the Client to retrieve the Performance Report and measured telemetry metric results.

All the above-mentioned APIs are provided in the SDK together with accompanying Developer Guides. Please refer to those documents for more details and examples of functional APIs.

## A.2 Integration of Service Payload Specific Attributes into the Performance Monitoring API

The Performance Monitoring API is technology-agnostic in the meaning that they serve as an interaction between the Client and the Server (SOF) and they do not contain any technology-specific information in their specifications. To pass the technology-specific information, an extension pattern is used. This applies to any of the Performance Monitoring APIs that carry technology-specific information for both the Performance Monitoring Configuration Job request and Performance Monitoring Results.

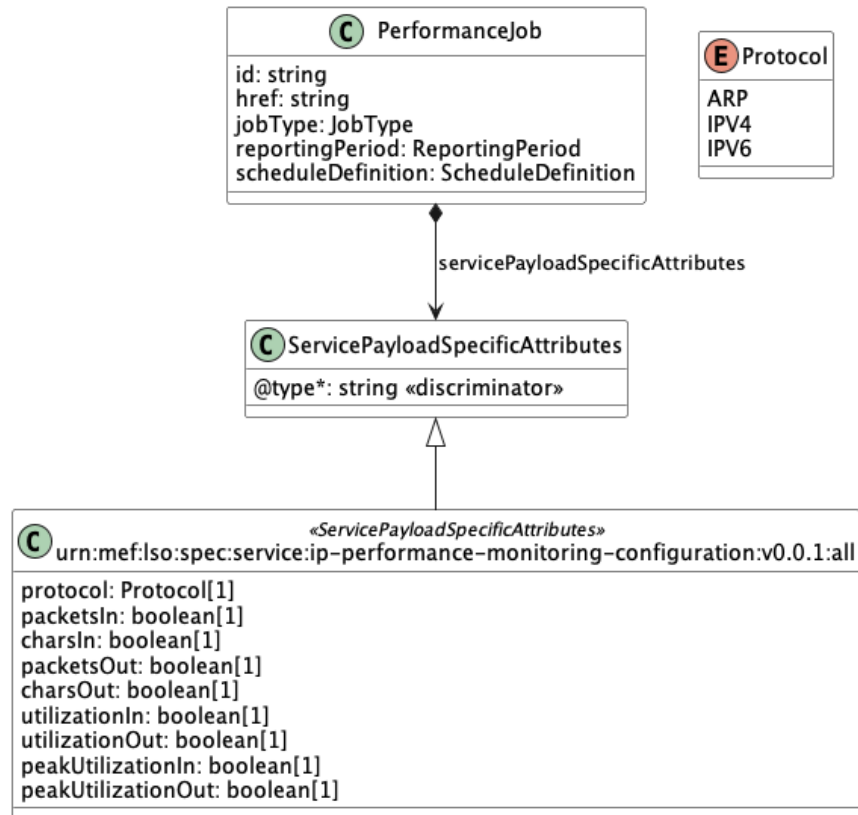
The extension hosting type in the API data model is `ServicePayloadSpecificAttributes` for IP Performance Monitoring Job configuration request. The extension type for IP Performance Monitoring Results is `ResultPayload`.

The `@type` attribute of that type must be set of a value that uniquely identifies the service specification. See Figure 13 and Figure 14. A unique identifier for MEF standard specifications is in URN format and is assigned by MEF. This identifier is provided as root schema `$id` and in service specification documentation. The example below shows a header of a Carrier Ethernet Subscriber UNI, where `$id: urn:mef:lso:spec:legato:ip-performance-monitoring-configuration:v0.0.2:all` the above-mentioned URN:

```
$id: urn:mef:lso:spec:legato:ip-performance-monitoring-configuration:v0.0.2:all  
$schema: http://json-schema.org/draft-07/schema#  
title: MEF LSO Legato – IP Performance Monitoring Configuration
```

In this case, this will be in format of examples below:

- `urn:mef:lso:spec:legato:ip-performance-monitoring-configuration:v0.0.2:all`



**Figure 13-Extension Pattern: Performance Monitoring Configuration (Job) Extension**

- urn:mef:lso:spec:legato:ip-performance-monitoring-results:v0.0.1:all



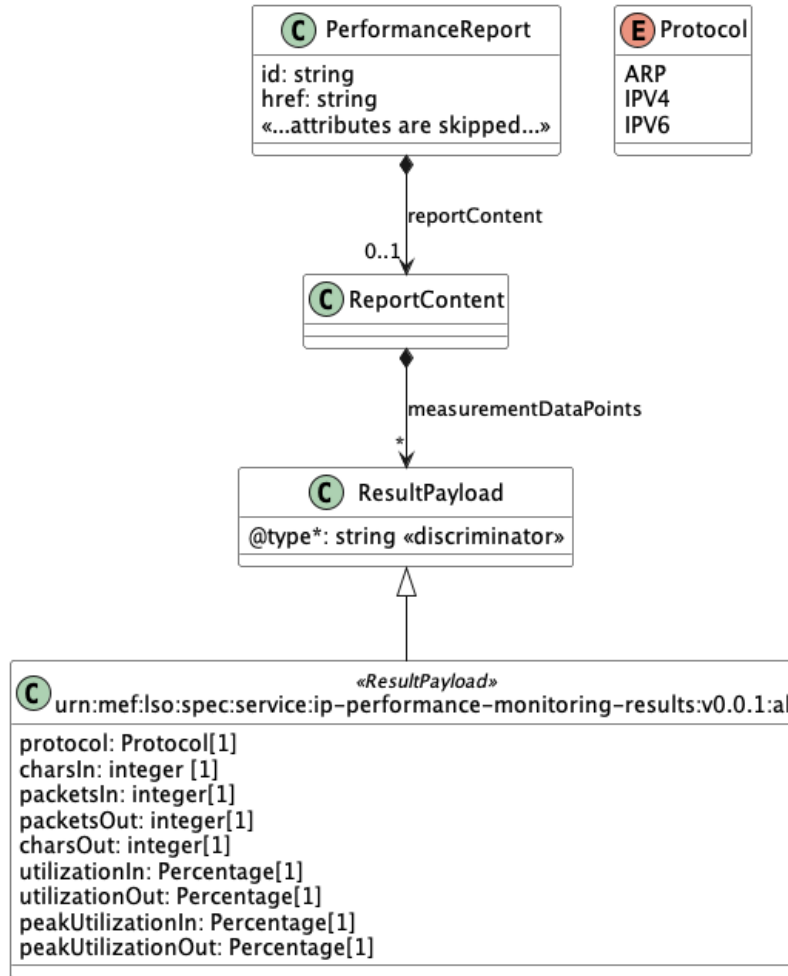


Figure 14-Extension Pattern: Performance Monitoring Results Extension

### A.3 Use Case 1: Create IP Performance Monitoring Configuration

The `PerformanceJob_Create` process is initiated by the Buyer/Client with the determination of the IP Job Configuration payload resource and associated set of payload specific attributes (i.e, `IpPerformanceMonitoringConfiguration`, `BfdConfiguration`, `PingConfiguration`, `TwampConfiguration`).

The Buyer/Client sends a request with a `PerformanceJob_Request` type in the body. The SOF performs request validation, assigns an `id`, and returns `PerformanceJob` type in the response body, with a state set to `acknowledged`. From this point, the `PerformanceJob` is ready for further processing. The Buyer/Client can track the progress of the process by either subscribing for notifications (Asynchronous operation) or by periodically polling (Synchronous operation) the `PerformanceJob`. The two patterns are illustrated in sequence diagrams below.

Figure 15-Performance Monitoring Job progress tracking – Notifications (Asynchronous)

Figure 16-Performance Monitoring Job progress tracking – Polling (Synchronous)

## A.3.1 Create Performance Monitoring Configuration Request for IP Telemetry

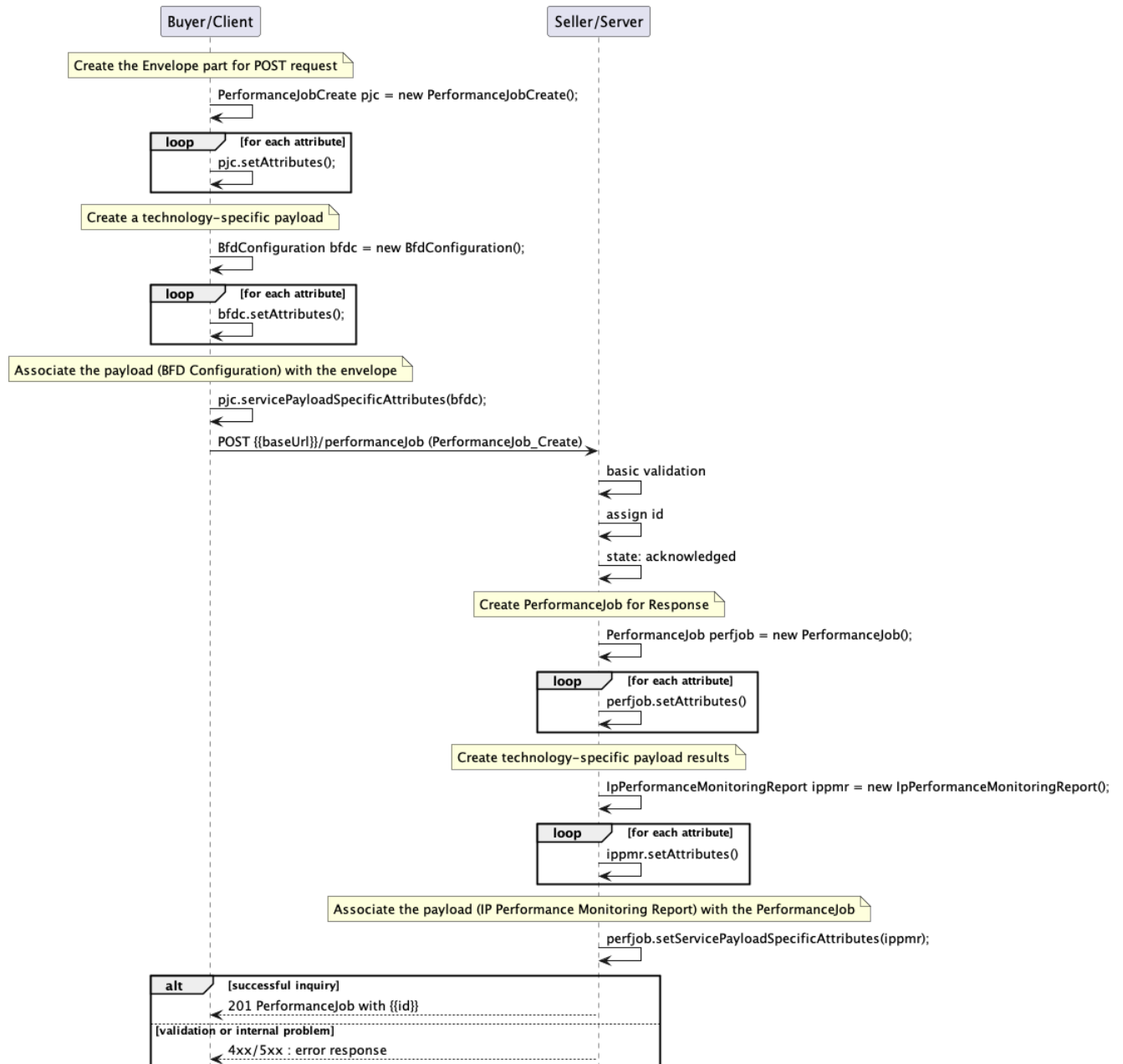


Figure 17-Performance Monitoring Configuration Request for IP

Figure 18-IP BFD Performance Configuration JSON Request

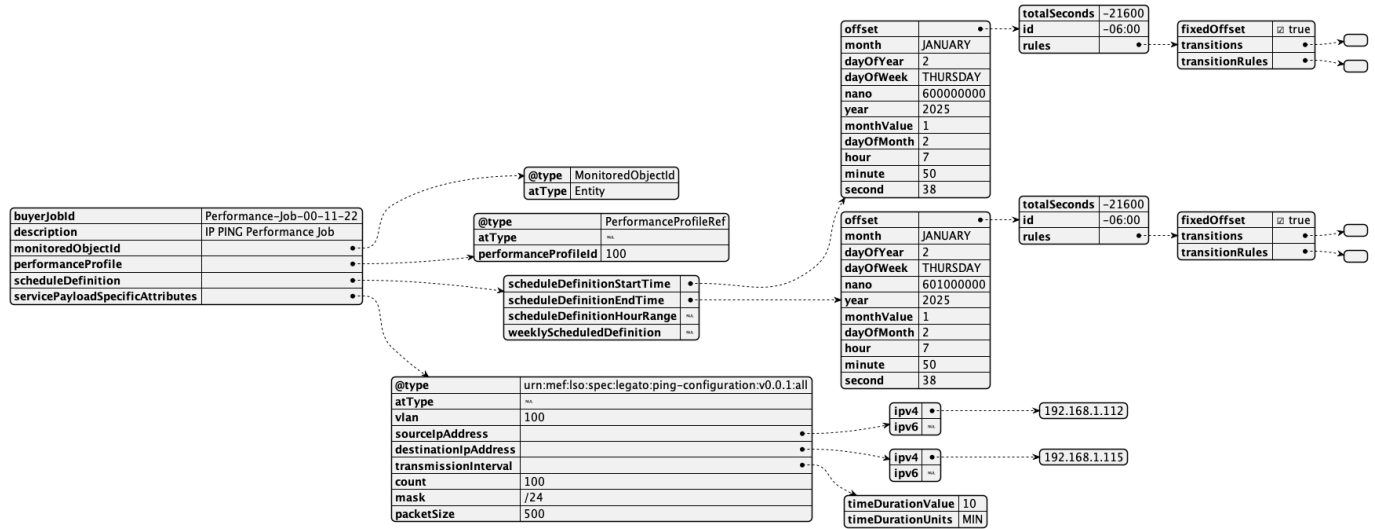


Figure 19-IP Ping Performance Configuration JSON Request

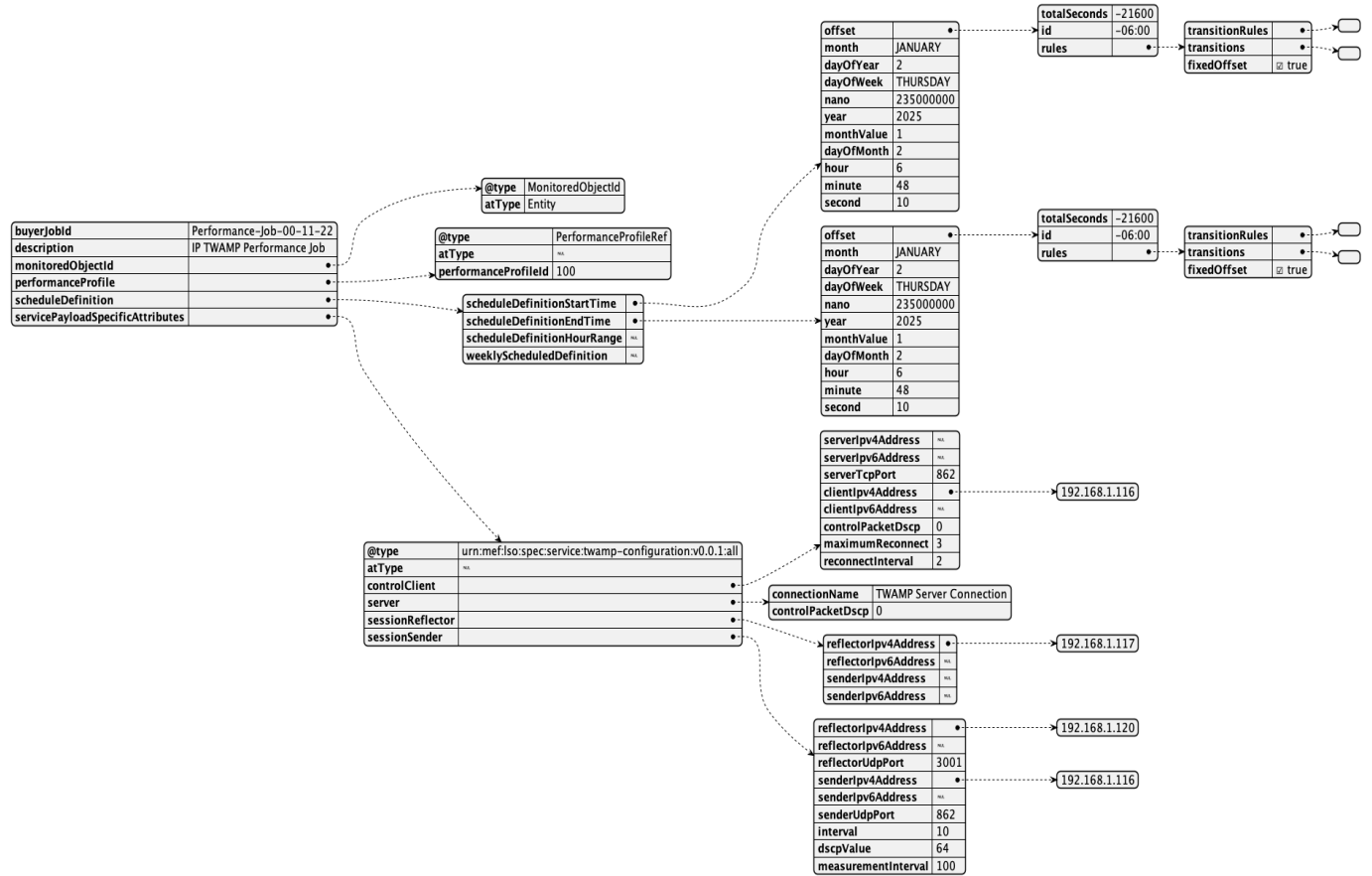


Figure 20-IP TWAMP Performance Configuration JSON Request

### A.3.2 Create Performance Monitor Results Response

The SOF response to the `PerformanceJob_Create` is a `PerformanceJob` which is shown below. The response echoes back all the attributes as provided by the Client/Buyer and contains the same number of `PerformanceJob_Create` as in the request.

Figure 21-IP Passive Statistics Performance Result JSON Response

Figure 22-IP BFD Performance Result JSON Response

Figure 23-IP Ping Performance Result JSON Response

Figure 24-IP TWAMP Performance Result JSON Response

The attributes that are set by the SOF in the response are marked with the **<<added by SOF>>** tag. The response to create request does not contain all possible attributes.

**Figure 25-Performance Monitoring Results Response**

#### **A.4 Use Case 2: Modify Performance Monitoring Job**

The following use case shows a request for a modification to an existing IP Performance Monitoring Job. A change to L1 Service Level Specification objective is performed.

#### **A.5 Use Case 3: Register for Performance Monitoring Profile Notifications**

The SOF communicates asynchronously with the Buyer/Client using Notifications provided that:

- Buyer/Client support a notification mechanism
- Buyer/Client has registered to receive notifications from the SOF

To register for notifications the Buyer/Client uses the **registerListener** operation from the **API: POST /hub**. The request contains two attributes:

- **callback** – mandatory, to provide the callback address the events will be notified to,
- **query** – optional, to provide the required types of event(s).

Below is an example Register for Notifications for all supported Notifications:

```
http://{server.com}:port/mefApi/service/serviceOrderingManagement/v5/hub
{
  "callback": "https://client.com/listenerEndpoint",
  "query": "eventType=performanceReportStateChangeEvent,
performanceJobCreateEvent"
```

Below is corresponding response from Server:

```
{
  "callback": "https://client.com/listenerEndpoint",
  "id": "1e29afb6-6d91-48a1-b4a7-5bca047f8764", << added by SOF >>
  "query": "eventType= "
```

#### **A.6 Use Case 4: Retrieve Event Subscription by Identifier**

The Buyer/Client can get detailed information about their subscribed to Notification(s) from the SOF by using a **GET/hub/{id}** operation. The payload returned in the response includes all attributes the Buyer/Client has provided when a Hub register a listener request was sent.

Below is an example of retrieval of Event Subscription by Identifier request:

466 `http://localhost:8080/mefApi/legato/serviceOrderingManagement/v5/hub/1e29afb6-`  
467 `6d91-48a1-b4a7-5bca047f8764`

468 Below is corresponding response from Server:

469 **A.7 Use Case 8: Send Notification**

470 The Buyer/Client Supports a notification mechanism. The notification mechanism used by the SOF  
471 is REST-based and notifications are transmitted by the SOF.

## 16 References

- [1] IETF JSON Schema draft 7, *JSON Schema: A Media Type for Describing JSON Documents* and associated documents, by Austin Wright and Henry Andrews, March 2018. Copyright © 2018 IETF Trust and the persons identified as the document authors. All rights reserved.
- [2] IETF RFC 2119, *Key words for use in RFCs to Indicate Requirement Levels*, by Scott Bradner, March 1997
- [3] IETF RFC 3444, *On the Difference between Information Models and Data Models*, January 2003
- [4] IETF RFC 8174, *Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words*, by Barry Leiba, May 2017. Copyright © IETF Trust and the persons identified as the document authors (2017). All Rights Reserved
- [5] MEF 55.1, Lifecycle Service Orchestration (LSO): Reference Architecture and Framework, January 2021.
- [6] MEF W133.1 Allegro, Interlude and Legato Fault Management and Performance Monitoring BR&UC, February 2024.
- [7] MEF W143 LSO Allegro, LSO Interlude and LSO Legato Performance Monitoring API-Developer Guide, December 2023.