

LSO L1 Service Schemas and Developer GuideDecember 2024MEF W103 v0.1Contribution Number

Working Draft

MEF W103 v0.1

LSO L1 Service Schemas and Developer Guide

December 2024

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 2024. 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:

1. 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
2. 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
3. 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

[1 List of Contributing Members 1](#_Toc184105018)

[2 Abstract 2](#_Toc184105019)

[3 Terminology and Abbreviations 3](#_Toc184105020)

[4 Compliance Levels 4](#_Toc184105021)

[5 Numerical Prefixes 5](#_Toc184105022)

[6 Introduction 6](#_Toc184105023)

[7 Overview of LSO Services 8](#_Toc184105024)

[8 Overview of Layer 1 Service Model 10](#_Toc184105025)

[9 Overview of L1 Subscriber Services 11](#_Toc184105026)

[9.1 L1 Subscriber UNI 11](#_Toc184105027)

[9.2 L1 Virtual Connection and L1 Virtual Connection End Points 11](#_Toc184105028)

[10 Overview of L1 Operator Services 12](#_Toc184105029)

[10.1 L1 Operator UNI 12](#_Toc184105030)

[10.2 L1 Operator Virtual Connection and Operator Virtual Connection End Points 12](#_Toc184105031)

[10.3 L1 ENNI and ENNI Service 13](#_Toc184105032)

[11 Data Model Design Principles and Assumptions 14](#_Toc184105033)

[12 Data Models for Layer 1 Services 15](#_Toc184105034)

[12.1 Organization and Structure of the Schemas 15](#_Toc184105035)

[12.1.1 Naming Conventions 16](#_Toc184105036)

[13 Relationships Between Entities 17](#_Toc184105037)

[13.1 Subscriber L1 Services Relationships Between Entities 17](#_Toc184105038)

[13.2 Operator Ethernet Services Relationships Between Entities 19](#_Toc184105039)

[14 Subscriber L1 Services Data Model 24](#_Toc184105040)

[14.1 L1SubscriberUni 24](#_Toc184105041)

[14.2 L1SubscriberVcEndPoint 25](#_Toc184105042)

[14.3 L1SubscriberVc 27](#_Toc184105043)

[15 Operator L1 Services Data Model 28](#_Toc184105044)

[15.1 L1OperatorUni 28](#_Toc184105045)

[15.2 L1OperatorVcEndPoint 29](#_Toc184105046)

[15.2.1 L1VcEndPointMap 31](#_Toc184105047)

[15.2.2 TriburtarySlotRate 31](#_Toc184105048)

[15.3 L1OperatorVc 31](#_Toc184105049)

[15.4 L1Enni 33](#_Toc184105050)

[15.4.1 L1NniPhysicalLayer 34](#_Toc184105051)

[15.4.2 L1NniOpticalInterfaceFunction 35](#_Toc184105052)

[15.4.3 L1NniCodingFunction 35](#_Toc184105053)

[15.4.4 OtuKOverHead 36](#_Toc184105054)

[15.4.5 HighOrderOduOverHead 36](#_Toc184105055)

[15.4.6 NniProtection 37](#_Toc184105056)

[15.5 L1EnniService 37](#_Toc184105057)

[15.5.1 PathOverHead 39](#_Toc184105058)

[16 Common Classes and Types 41](#_Toc184105059)

[16.1 ClientProtocol 41](#_Toc184105060)

[16.2 EnabledDisabled 41](#_Toc184105061)

[16.3 Eth1000BaseXOpticalInterfaceFunction 41](#_Toc184105062)

[16.4 Eth10GBaseWOpticalInterfaceFunction 41](#_Toc184105063)

[16.5 Eth10GBaseROpticalInterfaceFunction 42](#_Toc184105064)

[16.6 Eth100GBaseROpticalInterfaceFunction 42](#_Toc184105065)

[16.7 Eth40GBaseROpticalInterfaceFunction 42](#_Toc184105066)

[16.8 Fc100OpticalInterfaceFunction 42](#_Toc184105067)

[16.9 Fc200OpticalInterfaceFunction 42](#_Toc184105068)

[16.10 Fc400OpticalInterfaceFunction 43](#_Toc184105069)

[16.11 Fc800OpticalInterfaceFunction 43](#_Toc184105070)

[16.12 Fc1200OpticalInterfaceFunction 43](#_Toc184105071)

[16.13 Fc1600OpticalInterfaceFunction 43](#_Toc184105072)

[16.14 Fc3200OpticalInterfaceFunction 43](#_Toc184105073)

[16.15 L1UniCodingFunction 43](#_Toc184105074)

[16.16 L1UniOpticalInterfaceFunction 44](#_Toc184105075)

[16.17 L1VcEndPointExternalInterfaceType 45](#_Toc184105076)

[16.18 L1UniPhysicalLayer 45](#_Toc184105077)

[16.19 LineRate 46](#_Toc184105078)

[16.20 Oc3OpticalInterfaceFunction 46](#_Toc184105079)

[16.21 Oc12OpticalInterfaceFunction 46](#_Toc184105080)

[16.22 Oc48OpticalInterfaceFunction 47](#_Toc184105081)

[16.23 Oc192OpticalInterfaceFunction 47](#_Toc184105082)

[16.24 Oc768OpticalInterfaceFunction 48](#_Toc184105083)

[16.25 Stm10OpticalInterfaceFunction 48](#_Toc184105084)

[16.26 Stm4OpticalInterfaceFunction 48](#_Toc184105085)

[16.27 Stm16OpticalInterfaceFunction 49](#_Toc184105086)

[16.28 Stm64OpticalInterfaceFunction 49](#_Toc184105087)

[16.29 Stm256OpticalInterfaceFunction 49](#_Toc184105088)

[17 L1 Service Level Specification 51](#_Toc184105089)

[17.1 L1ServiceLevelSpecification 51](#_Toc184105090)

[17.2 L1VcEndPointRef 53](#_Toc184105091)

[17.3 L1VcRef 53](#_Toc184105092)

[17.4 Sls1wDelayPerformanceMetric 54](#_Toc184105093)

[17.5 Sls1wErroredSecondPerformanceMetric 54](#_Toc184105094)

[17.6 Sls1wSeverelyErroredSecondPerformanceMetric 55](#_Toc184105095)

[17.7 Sls1wUnavailabilitySecondPerformanceMetric 55](#_Toc184105096)

[17.8 Sls1wAvailabilityPerformanceMetric 55](#_Toc184105097)

[17.9 Time 56](#_Toc184105098)

[17.10 TimeIntervalT 56](#_Toc184105099)

[Appendix A Usage examples (Informative) 60](#_Toc184105100)

[A.1 High-level Flow 60](#_Toc184105101)

[A.2 Integration of Service Specification into the Service Order API 61](#_Toc184105102)

[A.3 Use Case 1: Create Service Order 62](#_Toc184105103)

[A.3.1 Create Service Order Request 64](#_Toc184105104)

[A.3.2 Create Service Order Response 65](#_Toc184105105)

[A.4 Use Case 2: Service Order Item to Modify Existing Service 67](#_Toc184105106)

[A.5 Use Case 3: Service Order Item to Delete Existing Service 68](#_Toc184105107)

[A.6 Use Case 4: Retrieve List of Service Orders 69](#_Toc184105108)

[A.7 Use Case 5: Retrieve Service Order by Service Order Identifier 70](#_Toc184105109)

[A.8 Use Case 6: Register for Notifications 70](#_Toc184105110)

[A.9 Use Case 7: Retrieve Event Subscription by Identifier 71](#_Toc184105111)

[A.10 Use Case 8: Send Notification 71](#_Toc184105112)

[18 References 72](#_Toc184105113)

# List of Figures

[Figure 1-LSO Reference Diagram 8](#_Toc184105114)

[Figure 2-LSO Service API Structure 9](#_Toc184105115)

[Figure 3- L1 Service Model Overview 10](#_Toc184105116)

[Figure 4-L1 Subscriber Service Model 11](#_Toc184105117)

[Figure 5-L1 Operator Service Model 12](#_Toc184105118)

[Figure 6-Schema Files Organization 15](#_Toc184105119)

[Figure 7-Subscriber L1 Service Order API Associations 17](#_Toc184105120)

[Figure 8-L1 Subscriber Services Entities and Relationships 19](#_Toc184105121)

[Figure 9-L1 Operator Service Order API Associations 20](#_Toc184105122)

[Figure 10-L1 Operator Entities and Relationships 23](#_Toc184105123)

[Figure 11-L1SubscriberUni Model 24](#_Toc184105124)

[Figure 12-L1SubscriberVcEndPoint Model 26](#_Toc184105125)

[Figure 13-L1SubscriberVc Model 27](#_Toc184105126)

[Figure 14-L1OperatorUni Model 28](#_Toc184105127)

[Figure 15-L1OperatorVcEndPoint Model 30](#_Toc184105128)

[Figure 16-L1OperatorVcModel 32](#_Toc184105129)

[Figure 17-L1Enni Model 33](#_Toc184105130)

[Figure 18-L1EnniService Model 38](#_Toc184105131)

[Figure 19-Service Level Specification Model 52](#_Toc184105132)

[Figure 20-L1 VC Service Level Specification Service Attribute Value Example 57](#_Toc184105133)

[Figure 21-L1 SLS JSON Example 59](#_Toc184105134)

[Figure 22-Service End-to-End Function Flow 60](#_Toc184105135)

[Figure 23-Extension Pattern: L1 Subscriber Service-Specific Extensions 62](#_Toc184105136)

[Figure 24-Extension Pattern: L1 Operator Service-Specific Extensions 62](#_Toc184105137)

[Figure 25-Service Order progress tracking – Notifications (Asynchronous) 63](#_Toc184105138)

[Figure 26-Service Order progress tracking – Polling (Synchronous) 64](#_Toc184105139)

[Figure 27-Service Order Request 64](#_Toc184105140)

[Figure 28-Service Order Response 67](#_Toc184105141)

[Figure 29-Service Order Modify Existing Service Request 68](#_Toc184105142)

[Figure 30-Service Order Modify Existing Service Response 68](#_Toc184105143)

[Figure 31-Service Order to Delete Existing Service 69](#_Toc184105144)

# List of Tables

[Table 1-Terminology and Abbreviations 3](#_Toc184105145)

[Table 2-Numerical Prefix Conventions 5](#_Toc184105146)

[Table 3-Service Relationship Roles L1 Subscriber Service 18](#_Toc184105147)

[Table 4-Place Relationship Role 19](#_Toc184105148)

[Table 5-Service Relationship Roles L1 Operator 20](#_Toc184105149)

[Table 6-Place Relationship Role 22](#_Toc184105150)

[Table 7-L1SubscriberUni Service Attributes 25](#_Toc184105151)

[Table 8-L1SubscriberVcEndPoint Service Attributes 26](#_Toc184105152)

[Table 9-L1SubscriberVc Service Attributes 27](#_Toc184105153)

[Table 10-L1OperatorUni Service Attributes 29](#_Toc184105154)

[Table 11-L1OperatorVcEndPoint Service Attributes 31](#_Toc184105155)

[Table 12-TributarySlotRate Service Attributes 31](#_Toc184105156)

[Table 13-L1OperatorVc Service Attributes 32](#_Toc184105157)

[Table 14-L1Enni Service Attributes 34](#_Toc184105158)

[Table 15-L1NniPhysicalLayer Service Attributes 35](#_Toc184105159)

[Table 16-L1NniCodingFunction Service Attributes 36](#_Toc184105160)

[Table 17-OtuKOverHead Service Attributes 36](#_Toc184105161)

[Table 18-HighOrderOduHead Service Attributes 37](#_Toc184105162)

[Table 19-L1EnniService Attributes 39](#_Toc184105163)

[Table 20-PathOverHead Attributes 40](#_Toc184105164)

[Table 21-L1UniPhysical Service Attributes 45](#_Toc184105165)

[Table 22-L1ServiceLevelSpecification Attributes 53](#_Toc184105166)

[Table 23-L1VcEndPointRef Attributes 53](#_Toc184105167)

[Table 24-L1VcEndPointRef Attributes 54](#_Toc184105168)

[Table 25-Sls1wDelayPerformanceMetric Attributes 54](#_Toc184105169)

[Table 26-Sls1wErroredSecondPerformanceMetric Attributes 54](#_Toc184105170)

[Table 27-Sls1wSeverelyErroredSecondPerformanceMetric Attributes 55](#_Toc184105171)

[Table 28-Sls1wUnavailabilitySecondPerformanceMetric Attributes 55](#_Toc184105172)

[Table 29-Sls1wAvailabilityPerformanceMetric Attributes 56](#_Toc184105173)

[Table 30-Time Attributes 56](#_Toc184105174)

[Table 31-TimeIntervalT Attributes 57](#_Toc184105175)

# 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.

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.

# 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 Service Functions – specifically, Service Order and Service Inventory, for Layer 1 Subscriber and Operator Services. It then provides a basic information model for the MEF L1 Service Attributes. 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 Subscriber and Operator L1 Services Data Model and the schemas provided that embody the Data Model. L1 Services are described by a set of Service Attributes. Each Service Attribute describes an aspect of the service that is agreed between the provider and the user of the service. The document that describes the Service Attributes for Subscriber L1 Services is MEF 63 [5] and Operator L1 Services is MEF 64 [6].

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_l1cs_service>.

# 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** |
| --- | --- | --- |
| Business Applications | The Service Provider functionality supporting Business Management Layer functionality (e.g., product catalog, order management, billing, relationship management, etc.) | MEF 55.1 [9] |
| BUS | See Business Applications | MEF 55.1 [9] |
| 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] |
| ENNI | Used for brevity when referring to an L1 ENNI. | MEF 64 [6] |
| L1CI | Layer 1 Characteristic Information. | MEF 63 [5] |
| L1 ENNI | Layer 1 External Network Network Interface | MEF 64 [6] |
| L1 Service | A connectivity service which delivers Layer 1 Characteristic Information that is specified using Service Attributes. | MEF 63 [5] |
| Information Model | A representation of concepts of interest to an environment in a form that is independent of data repository, data definition language, query language, implementation language, and protocol. | IETF RFC 3444 [3] |
| Order | One or more Service Order Items formulated into a fulfillment request made by a Client to a Server. | This document (derived from MEF 57.2) |
| Service Provider | In the context of this document, a Service Provider is an Ethernet Service Provider. In this document, we use Service Provider to include Super Operator as specified in MEF 26.2 (also referred to as SP/SO). | This Document |

Table -Terminology and Abbreviations

# 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**.**

# 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 | 103 | Ki | 210 |
| M | 106 | Mi | 220 |
| G | 109 | Gi | 230 |
| T | 1012 | Ti | 240 |
| P | 1015 | Pi | 250 |
| E | 1018 | Ei | 260 |
| Z | 1021 | Zi | 270 |
| Y | 1024 | Yi | 280 |

Table -Numerical Prefix Conventions

# Introduction

LSO provides programmatic interfaces for establishing automated exchange of information (i.e., Service Order, Service Inventory) between a Business Application and Service Orchestration Function. 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 (e.g., Service Order or Inventory, etc.) and the inner-most structure contains information relating to the specific service, for example L1 Service.

The specific types of L1 Services are Subscriber and Operator Services. Subscriber L1 Services are requested between a Customer and a Service Provider or a Service Provider and a Partner. Operator L1 Services are requested between a Service Provider (SP) and a Partner. The Service Attributes for Subscriber and Operator L1 Services are defined in MEF 63 [5] and MEF 64 [6] respectively.

This specification is accompanied by a Data Model for Subscriber and Operator L1 Services instantiated as a set of YAML schemas that can be used with MEF LSO APIs to perform Service Order, and request an Inventory for the Subscriber and Operator L1 Services consisting of:

The Data Model for Subscriber L1 Services includes:

* L1 Subscriber VC: A L1 Subscriber VC is an association of two or more L1 VC End Points (L1 VC EPs).
* L1 VC End Point: A L1 VC End Point is a construct at a L1 Subscriber UNI that selects a subset of the Service Frames that pass over the L1 Subscriber UNI. A L1 VC End Point represents the logical attachment of an L1 Subscriber VC to a L1 Subscriber UNI.
* L1 Subscriber UNI: A construct that represents the L1 Network Interface demarcation point between the responsibility of the Service Provider and the responsibility of the Subscriber.

The Data Model for Operator L1 Services includes:

* L1 Operator VC: The L1 Operator VC is the building block for constructing a L1 VC spanning multiple Operator CENs. A L1 Operator VC is an association of L1 Operator VC End Points.
* L1 VC End Point: A logical entity at a given External Interface that is associated with a distinct set of frames passing over that External Interface. A L1 Operator VC End Point represents the logical attachment of an L1 Operator VC to an External Interface (a L1 Operator UNI or L1 ENNI).
* L1 Operator UNI: UNI used in L1 Operator Service solution where attributes are agreed to by the Service Provider/Super Operator and the Operator.
* L1 ENNI: A reference point representing the boundary between two or more Operator CENs that are operated as separate administrative domains.
* L1 ENNI Service: A construct that represents the L1 ENNI Service Attributes for a L1 ENNI used by a particular SP/SO. For each instance of a L1 ENNI, there can be multiple sets of L1 ENNI Service Attributes.

The document contains the following sections:

An overview of LSO Services (Section 7)

An overview L1 Service Model (Section 8)

An overview of L1 Subscriber Services (Section 9)

An overview of L1 Operator Services (Section 10)

Data Model Design Principles and Assumptions (Section 11)

Data Models for L1 Services (Section 12)

Relationship between the Entities (Section 13)

Subscriber L1 Services Data Model (Section 14)

Operator L1 Services Data Model (Section 15)

Common Classes and Types (Section 16)

L1 Service Level Specification (Section 17)

# Overview of LSO Services

MEF 55.1 [7] describes the Reference Architecture for Lifecycle Service Orchestration (LSO) of MEF-defined connectivity services. MEF 55.1 [7] 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).

A screenshot of a cell phone

Description automatically generated

Figure -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.

A close-up of a document

Description automatically generated

Figure -LSO Service API Structure

This document defines the service-specific payload, shown as YAML/JSON Data Model in the figure above, specifically for a L1 Subscriber and L1 Operator Services as defined in MEF 63 [5] and MEF 64 [6] respectively. The envelope resources of the API and association to specific payload resources will be discussed in detail later in this document.

# Overview of Layer 1 Service Model

The L1 Service model has eight main classes, L1SubscriberVc, L1SubscriberVcEndPoint, L1SubscriberUni, L1OperatorVc, L1OperatorVcEndPoint, L1OperatorUni, L1Enni and L1EnniService. A L1 Service is defined as either a L1 Subscriber Service or an L1 Operator Service.

A L1 Subscriber Service has two L1 Subscriber UNIs and corresponding L1 VC and L1 VC End Points provided by a Service Provider to a Customer. In some L1 Services the Subscriber will have locations that are not all served by a single L1 Operator. Specifically, to support all L1 Subscriber’s UNIs one or more L1 Operators are required. This is where an L1 Operator Service is used.

A screenshot of a computer

Description automatically generated

Figure - L1 Service Model Overview

# Overview of L1 Subscriber Services

This specification describes a data model for MEF-defined Subscriber L1 Services. A Subscriber L1 Service is a Layer 1 Service provided to an end user (the Subscriber) by a Service Provider. There is no restriction on the type of organization that can act as a Subscriber; for example, a Subscriber can be an enterprise, a mobile operator, an IT system integrator, a government department, etc. At its most basic, a L1 Subscriber Service provides connectivity between different parts of the Subscriber's network (usually at different physical locations) or between the Subscriber's network and an external network. The subsequent sub-sections provide background on the set of objects that are associated with a L1 Subscriber Service.

A diagram of a program

Description automatically generated

Figure -L1 Subscriber Service Model

## L1 Subscriber UNI

A User Network Interface (UNI) is the demarcation point between the responsibility of the Service Provider (SP) and the responsibility of the Subscriber. A Subscriber is connected to the SP at one or more UNIs.

## L1 Virtual Connection and L1 Virtual Connection End Points

A fundamental aspect of L1 Subscriber Service is the L1 Virtual Connection (L1 VC) is an association of two or more L1 VC End Points (L1 VC EPs). A L1 VC EP is a construct at a L1 UNI that selects a subset of the Service Frames that pass over the L1 UNI.

# Overview of L1 Operator Services

This specification describes a data model for MEF-defined L1 Operator Services. When a Service Provider provides an end-to-end L1 Subscriber Service to a Subscriber, they might not be able to implement the entire service using their own network - for instance, one of the Subscriber UNIs might not be in a geographic region where the Service Provider does not operate. In this case, the Service Provider must partner with another Operator who can reach that UNI. The Operator provides L1 connectivity service between the UNI and a point where they can interconnect with the SP's network as described in [6].

A diagram of a program

Description automatically generated

Figure -L1 Operator Service Model

## L1 Operator UNI

A L1 Operator UNI is the supporting interface between a customer and SP/SO when the SP/SO is serving L1 Operator Services. The L1 Operator UNI is different from the L1 Subscriber UNI. They are different because the value of each Operator UNI Service Attribute is agreed by the SP/SO and the Operator while the value of each Subscriber UNI Service Attribute is agreed to by the Subscriber and Service Provider.

## L1 Operator Virtual Connection and Operator Virtual Connection End Points

The L1 Operator Virtual Connection (L1 VC) is the building block for constructing a L1 VC spanning multiple Operator L1 networks. In the same way that a L1 VC defines an association of UNIs, an L1 VC is and association of L1 VC End Points. An L1 VC End Point is a logical entity at a given External Interface that is associated with a distinct set of frames passing over that External Interface. An L1 VC End Point represents the logical attachment of an L1 VC to an External Interface (a L1 UNI or L1 ENNI) [6].

## L1 ENNI and ENNI Service

An External Network Network Interface (ENNI) is the demarcation point between the responsibility of one Operator and another - in other words, it is the interface where two Operators interconnect. For each instance of an ENNI, there are multiple sets of ENNI Service Attributes. The value for each ENNI Service Attribute in a set for an Operator network is specific to a SP/SO that is using the ENNI per [6]. Each such value is agreed to by the SP/SO and the Operator.

# Data Model Design Principles and Assumptions

A Service Attribute for a Service 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 Service-Specific Attribute. A Service-Specific Attribute could be a Service Attribute (in the case where the Service Attribute itself has a simple type) or it could be a parameter within a Service Attribute (if the Service Attribute is a structured object or a composition of such objects). The classification for each Service-Specific Attribute may be different across Service Function, Service Action, and Service Offering.

The L1 Service data model supports both INSTALL and CHANGE actions for Service Order for L1 Subscriber VC, L1 Operator VC, L1 Subscriber UNI, L1 Operator UNI, L1 Subscriber VC End Point, L1 Operator VC End Point, L1 ENNI and L1 ENNI Service. The L1 Service data model supports the RETRIEVE action for Inventory for all Service Order components.

The location and physical layer of a L1 Subscriber or Operator UNI and L1 ENNI cannot be changed once it is ordered; instead, this is handled as an installation (L1 Subscriber or Operator UNI or L1 ENNI at new location) and disconnect (L1 Subscriber or Operator or L1 ENNI at previous location), as there is often a requirement for a smooth transition with minimum downtime.

# Data Models for Layer 1 Services

The data models for the L1 Service 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.

## Organization and Structure of the Schemas

The schemas are organized into a file structure as shown in Figure 6.

A screenshot of a computer

Description automatically generated

Figure -Schema Files Organization

Both Subscriber and Operator L1 Service schemas are provided in the same directory. The is one file that provides common resources that are shared with Subscriber and Operator service:

* l1Cs/l1CsCommon.yaml – provides classes shared among all L1 services.

These common classes are referenced in the relevant service component schema files. For example, the L1UniPhysicalLayer attribute specified in l1CsCommon.yaml file refers to common Layer 1 Physical Layer definition:

L1UniPhysicalLayer:

type: object

description: >-

The L1 Physical Layer Service Attribute specifies the Client Protocol, the Coding Function and the Optical Interface Function. Reference MEF 63 Section 8.1.2 Physical Layer Service Attribute.

properties:

clientProtocol:

$ref: '#/definitions/ClientProtocol'

l1UniCodingFunction:

$ref: '#/definitions/L1UniCodingFunction'

l1UniOpticalInterfaceFunction:

$ref: '#/definitions/L1UniOpticalInterfaceFunction'

### Naming Conventions

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

# Relationships Between Entities

This section describes the constraints and relationships between the primary Service Order Items for both L1 Subscriber Service (L1 Subscriber VC, L1 Subscriber UNI and L1 Subscriber VC End Point) and L1 Operator Service (L1 Operator VC, L1 Operator UNI, L1 ENNI and L1 Operator VC End Points).

The L1 Service is associated with two or more UNIs each associated with an VC End Point and End Points associated with L1VC. The Operator Ethernet Service is one UNI and one ENNI associated with an VC End Point and End Points associated with L1VC.

## Subscriber L1 Services Relationships Between Entities

A MEF Service Order for Subscriber L1 Service has one or more Service Order Items which are components of the Service-agnostic envelope part of the MEF 99 API[10]. Each Service Order Item is associated to the Service-specific payload components (L1SubscriberVc, L1SubscriberVcEndPoint and L1SubscriberUni. The relationships between the envelope and payload components are shown in Figure 7.

A screenshot of a diagram

Description automatically generated

Figure -Subscriber L1 Service Order API Associations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Source Service | Relationship Type | Cardinality | Target Service |
| 1 | L1SubscriberVcEndPoint | L1\_UNI\_ENDPOINT\_OF\_VC | 1 | L1SubscriberVc |
| 2 | L1SubscriberVcEndPoint | CONNECTS\_TO\_L1\_SUBSCRIBER\_UNI | 1 | L1SubscriberUni |

Table -Service Relationship Roles L1 Subscriber Service

1. For a service listed in Source Service column in Table 3, the Relationship Type field of the Service Order Item Relationship types **MUST** contain one the corresponding values shown in the Relationship Type column.
2. For Order, the relationships to L1 Subscriber UNIs **MUST** be specified for every INSTALL of, or CHANGE to a L1 Subscriber VC.
3. For Order, the relationships to L1 Subscriber UNIs **MUST** be specified for every INSTALL of, or CHANGE to a L1 Subscriber VC End Point.
4. For a L1 Subscriber VC service, the relationship to a UNI **MUST** reference a L1 Subscriber Uni Order Item.
5. For a CHANGE to an L1 Subscriber VC Service, the relationship to the L1 Subscriber UNI **MUST NOT** be changed from the value present in the Service Inventory.

[R5] indicates that once a L1 Subscriber VC and L1 Subscriber VC End Point are associated with a Subscriber UNI, it cannot be associated with a different L1 Subscriber UNI.

A L1SubscriberUni and a L1SubscriberVcEndPoint may be included in the same Service Order as the L1SubscriberVc. The L1SubscriberUni is associated with a specific INSTALL\_LOCATION, which is required at INSTALL and CHANGE. Once a L1SubscriberUni is associated with a specific location, the INSTALL\_LOCATION cannot be changed and as, the same INSTALL\_LOCATION value must be specified for every CHANGE.

The install location is captured in the service-agnostic part of the Service Order API. The value in the Place Relationship Role column in the table below is used in the *role* field of the *RelatedPlaceRefOrValue* type.

|  |  |  |  |
| --- | --- | --- | --- |
| Service | Place Relationship Role | Cardinality | CHANGE |
| L1SubscriberUni | INSTALL\_LOCATION | 1 | Must be same value as Service Inventory. |

Table -Place Relationship Role

1. For a L1SubscriberUni, the Role field (role) of the Related Place (RelatedPlaceRefOrValue) type, MUST contain one of the values shown in Place Relationship Role in Table 4.
2. For Order, the Related Place (*RelatedPlaceRefOrValue)* **MUST** be specified for every INSTALL of, or CHANGE to, a L1SubscriberUni.
3. For a CHANGE to a L1SubscriberUni the Related Place **MUST NOT** be changed from the value present in the Service Inventory.

A diagram of a building

Description automatically generated

Figure -L1 Subscriber Services Entities and Relationships

## Operator Ethernet Services Relationships Between Entities

A MEF Service Order for an L1 Operator Service has one or more Service Order Items which are components of the Service-agnostic envelope part of the MEF 99 API[8]. Each Service Order Item is associated to the Service-specific payload components (L1OperatorVc, L1OperatorVcEndPoint, L1OperatorUni, L1Enni and L1EnniService. The relationships between the envelope and payload components are shown in Figure 9.

A screenshot of a computer

Description automatically generated

Figure -L1 Operator Service Order API Associations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Source Service | Relationship Type | Cardinality | Target Service |
| 1 | L1OperatorVcEndPoint | L1\_UNI\_ENDPOINT\_VC | 1 | L1OperatorVc |
| 2 | L1OperatorVcEndPoint | L1\_ENNI\_ENDPOINT\_VC | 1 | L1OperatorVc |
| 3 | L1OperatorVcEndPoint | CONNECTS\_TO\_L1\_OPERATOR\_UNI | 1 | L1OperatorUni |
| 4 | L1OperatorVcEndPoint | CONNECTS\_TO\_ENNI | 1 | L1Enni |
| 5 | L1Enni | REQUIRES | 1..\* | L1EnniService |

Table -Service Relationship Roles L1 Operator

1. For a service listed in Source Service column in Table 3, the Relationship Type field of the Service Order Item Relationship types **MUST** contain one the corresponding values shown in the Relationship Type column.
2. For Order, the relationships to L1OperatorUni **MUST** be specified for every INSTALL of, or CHANGE to an L1OperatorVc.
3. For Order, the L1Enni **MUST** specify an L1EnniService specific to relationship between SP/SO.
4. For Order, the relationships to L1Enni **MUST** be specified for every INSTALL of, or CHANGE to an L1 Operator VC.
5. For Order, the relationships to L1 Operator UNIs **MUST** be specified for every INSTALL of, or CHANGE to an L1 Operator VC End Point.
6. For Order, the relationships to L1 ENNI **MUST** be specified for every INSTALL of, or CHANGE to an L1 VC End Point.
7. For an L1 VC service, the relationship to an L1 Operator UNI **MUST** reference a L1 Operator UNI Order Item.
8. For an L1 VC service, the relationship to an L1 ENNI **MUST** reference a L1Enni Order Item.
9. For a CHANGE to an L1 VC Service, the relationship to the L1 Operator UNI **MUST NOT** be changed from the value present in the Service Inventory.
10. For a CHANGE to an L1 VC Service, the relationship to the L1 ENNI **MUST NOT** be changed from the value present in the Service Inventory.

[R9] indicates that once an L1 VC and L1 VC End Point are associated with an L1 Operator UNI, it cannot be associated with a different L1 Operator UNI.

[R10] indicates that once an L1 VC and L1 VC End Point are associated with an L1 ENNI, it cannot be associated with a different L1 ENNI.

An L1 Operator UNI and an L1 VC End Point may be included in the same Service Order as the L1 VC. The L1 Operator UNI is associated with a specific INSTALL\_LOCATION, which is required at INSTALL and CHANGE. Once a L1 Operator UNI is associated with a specific location, the INSTALL\_LOCATION cannot be changed and as, the same INSTALL\_LOCATION value must be specified for every CHANGE.

The install location is captured in the service-agnostic part of the Service Order API. The value in the Place Relationship Role column in the table below is used in the *role* field of the *RelatedPlaceRefOrValue* type.

|  |  |  |  |
| --- | --- | --- | --- |
| Service | Place Relationship Role | Cardinality | CHANGE |
| L1OperatorUni | INSTALL\_LOCATION | 1 | Must be same value as Service Inventory. |
| L1Enni | INSTALL\_LOCATION | 1 | Must be same value as Service Inventory. |

Table -Place Relationship Role

1. For a L1OperatorUni, the Role field (role) of the Related Place (RelatedPlaceRefOrValue) type, **MUST** contain one of the values shown in Place Relationship Role in Table 6.
2. For Order, the Related Place (*RelatedPlaceRefOrValue)* **MUST** be specified for every INSTALL of, or CHANGE to, a L1OperatorUni.
3. For a CHANGE to a L1OperatorUni the Related Place **MUST NOT** be changed from the value present in the Service Inventory.
4. For Order, the Related Place (*RelatedPlaceRefOrValue)* **MUST** be specified for every INSTALL of, or CHANGE to, a L1Enni.
5. For a CHANGE to a L1Enni the Related Place **MUST NOT** be changed from the value present in the Service Inventory.

A diagram of a software process

Description automatically generated

Figure -L1 Operator Entities and Relationships

# Subscriber L1 Services Data Model

A Subscriber L1 Service is a Layer 1 Service provided to an end user (the Subscriber) by a Service Provider. There is no restriction on the type of organization that can act as a Subscriber; for example, a Subscriber can be an enterprise, a mobile operator, an IT system integrator, a government department, etc. At its most basic, a Subscriber L1 Service provides connectivity for frames between different parts of the Subscriber’s network (usually at different physical locations).

The L1 Subscriber Services Resources and corresponding Attributes are listed in groups:

* L1SubscriberUni
* L1VcEndPoint
* L1Vc

## L1SubscriberUni

The UNI is the physical demarcation point between the responsibility of the L1 Service Provider and the responsibility of the L1 Subscriber. Reference MEF 63 [5] Section 8.

A screenshot of a diagram

Description automatically generated

Figure -L1SubscriberUni Model

| **Schema File Name: l1Cs/l1SubscriberUni.yaml**  **$id: urn:mef:lso:spec:service:l1-subscriber-uni:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| identifier | String  *Min length=1*  *Max length=45*  *Pattern=pattern: "[\x20-\x7F]+"* | 1 | The value of the UNI ID Service Attribute is a string that is used to allow Subscriber and Service Provider to uniquely identify the UNI. Reference MEF 64 Section 8.1.1 UNI ID Service Attribute. |
| physicalLayer | L1UniPhysicalLayer | 1 | The Physical Layer Service At-tribute specifies the Client Protocol, the Coding Function and the optical interface Function. Reference MEF 63 Section 8.1.2 Physical Layer Service Attribute. |

Table 7-L1SubscriberUni Service Attributes

## L1SubscriberVcEndPoint

A L1SubscriberVcEndPoint is a logical entity at a given L1SubscriberUni that is associated with L1CI passing over that L1SubscriberUni. A L1SubscriberVc is an association of two L1SubscriberVcEndPoints. A L1SubscriberVcEndPoint represents the logical attachment of a L1SubscriberVc to a L1SubscriberUni. Reference MEF 63 [5] Section 8.3 Subscriber L1VC End Point Service Attributes.

A screenshot of a computer

Description automatically generated

Figure -L1SubscriberVcEndPoint Model

| **Schema File Name: l1Cs/l1SubscriberVcEndPoint.yaml**  **$id: urn:mef:lso:spec:service:l1-subscriber-vc-end-point:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| identifier | String  *Min length=1*  *Max length=45*  *Pattern=pattern: "[\x20-\x7F]+"* | 1 |  |

Table -L1SubscriberVcEndPoint Service Attributes

*NOTE: that L1VC EP L1Subscriber UNI attributes is not provided given the relationship to Subscriber UNI is provided in the envelope part of API (i.e., MEF 99).*

## L1SubscriberVc

A L1SubscriberVc is an association of two or more L1VC End Points (L1 VC EPs). Reference MEF 63 Section 8.2 Subscriber L1VC Service Attributes [5].

A screenshot of a computer

Description automatically generated

Figure -L1SubscriberVc Model

| **Schema File Name: l1Cs/l1SubscriberVcEndPoint.yaml**  **$id: urn:mef:lso:spec:service:l1-subscriber-vc:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| identifier | String  *Min length=1*  *Max length=45*  *Pattern=pattern: "[\x20-\x7F]+"* | 1 |  |

Table -L1SubscriberVc Service Attributes

*NOTE: that L1VC List of L1VC EPs attributes is not provided given the relationship to L1VC EP is provided in the envelope part of API (i.e., MEF 99).*

# Operator L1 Services Data Model

An Operator L1 Service is a Layer 1 Service provided to an end user (the Subscriber) by a Service Provider.

The L1 Operator Services Resources and corresponding Attributes are listed in groups:

* L1OperatorUni
* L1OperatorVcEndPoint
* L1OperatorVc
* L1Enni
* L1EnniService

## L1OperatorUni

This class represents the Operator UNI Service Attributes that are agreed on by the SP/SO and the Operator for each UNI. The model below illustrates the payload component association to envelope components.

A screenshot of a diagram

Description automatically generated

Figure -L1OperatorUni Model

| **Schema File Name: l1Cs/l1OperatorUni.yaml**  **$id: urn:mef:lso:spec:service:l1 -operator-uni:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| Identifier | String  *Min length=1*  *Max length=45*  *Pattern=pattern: "[\x20-\x7F]+"* | 1 | An identifier for the UNI intended for SP/SO and Operator to uniquely identify the UNI. Reference MEF 64 Section 8.3.1 Operator UNI Identifier Service Attribute. |
| physicalLayer | L1UniPhysicalLayer | 1 | The Physical Layer Service At-tribute specifies the Client Protocol, the Coding Function and the optical interface Function. Reference MEF 64 Section 8.3.2 Physical Layer Service Attribute. |

Table -L1OperatorUni Service Attributes

## L1OperatorVcEndPoint

An Operator L1VC End Point represents the logical attachment of an Operator L1VC to an EI. Reference MEF 64 Section 8.5 Operator L1VC End Point Service Attributes.

A diagram of a service

Description automatically generated

Figure -L1OperatorVcEndPoint Model

| **Schema File Name: l1Cs/l1OperatorVcEndPoint.yaml**  **$id: urn:mef:lso:spec:service:l1-operator-ovc-end-point:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| identifier | String  Min length=1  Max length=45  Pattern=pattern: "[\x20-\x7F]+" | 1 | An identifier for the Operator L1VC End Point is a string that is used to allow the SP/SO and operator to uniquely identify the Operator L1VC End Point. Reference MEF 64 Section 8.5.1 Operator L1VC End Point Identifier Service Attribute. |
| L1VcEndPointExternalInterfaceType | L1VcEndPointExternalInterfaceType | 1 | Enumeration used to indicate if the L1VC end point is either UNI or ENNI. |
| l1VcEndPointMap | L1VcEndPointMap | 1 | Either Not Applicable or non-empty list of tuples of attributes in Table 16. Reference MEF 64 Section 8.5.4 Operator L1VC End Point Map Service Attribute. |

Table -L1OperatorVcEndPoint Service Attributes

### L1VcEndPointMap

Either Not Applicable or non-empty list of tuples of attributes in Table 16. Reference MEF 64 Section 8.5.4 Operator L1VC End Point Map Service Attribute.

| **Schema File Name: l1Cs/l1OperatorVcEndPoint.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| lineRate | LineRate | 1 | Enumeration representing physical line rate. |
| tributarySlotRate | TributarySlotRate | 1 | Enumeration representing tributary slot rate in Gb/s. |
| tributarySlots | Integer | 1..\* | Tributary Slot is list of integers that represents Tributary Slots occupied in a HO ODUi. |

Table -TributarySlotRate Service Attributes

### TriburtarySlotRate

Enumeration representing tributary slot rate in Gb/s.

* 1.25
* 2.5

## L1OperatorVc

An Operator L1VC is an association of two Operator L1VC End Points. Reference MEF 63 Section 8.4 Operator L1VC Service Attributes.

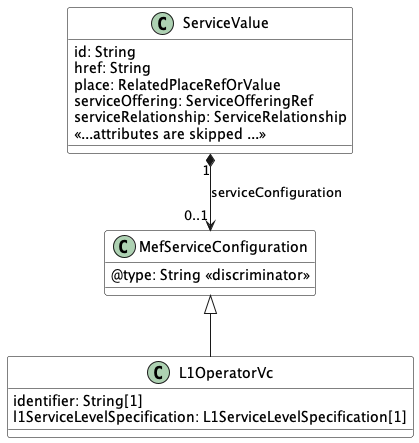


Figure -L1OperatorVcModel

| **Schema File Name: l1Cs/l1OperatorVc.yaml**  **$id: urn:mef:lso:spec:service:l1-operator-vc:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| identifier | String  Min length=1  Max length=45  Pattern=pattern: "[\x20-\x7F]+" | 1 | An identifier for the OVC intended for management purposes. Reference MEF 26.2 Section 12.1 OVC Identifier Service Attribute. |
| l1ServiceLevelSpecification | L1ServiceLevelSpecification | 1 | The Subscriber L1VC Service Level Specification (SLS) Service Attribute is the technical specification of aspects of the service performance agreed to by the Service Provider and Subscriber. Reference MEF 63 Section 8.2.3. |

Table -L1OperatorVc Service Attributes

*NOTE: L1VC List of L1VC EPs attributes is not provided given the relationship to L1VC EP is provided in the envelope part of API (i.e., MEF 99).*

## L1Enni

The ENNI is a reference point representing the boundary between two or more Operator CENs that are operated as separate administrative domains.

A screenshot of a computer

Description automatically generated

Figure -L1Enni Model

| **Schema File Name: 1Cs/l1Enni.yaml**  **$id: urn:mef:lso:spec:service:l1-enni:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| peeringIdentifier | String  *Min length=1*  *Max length=45*  *Pattern=pattern: "[\x20-\x7F]+"* | 1 | The ENNI Peering Identifier value is a string used to allow the Operators at ENNI to uniquely identify the ENNI. Reference MEF 64 Section 8.1.1 ENNI Peering Identifier Common Attribute. |
| listOfPhysicalLayer | L1NniPhysicalLayer | 1..\* | The list of coding function and wavelength structure supporting the ENNI. Reference MEF 64 Section 8.1.2 ENNI List of Physical Layers Common Attribute. |
| protection | NniProtection | 1 | Enumeration representing the protection protocol employed at ENNI for the ODU container exchanged by the operator. The enumeration value is either None or one of the rows as specified in G.873. |

Table -L1Enni Service Attributes

### L1NniPhysicalLayer

The L1 NNI Physical layer Service Attribute is a list of 2-tuples of the ENNI Coding Function and ENNI Optical Interface Function. Reference MEF 64 Section 8.1.2 ENNI List of Physical Layers Common Attribute.

| **Schema File Name: 1Cs/l1Enni.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| l1NniOpticalInterfaceFunction | L1NniOpticalInterfaceFunction | 1 | The L1 NNI Physical layer Service Attribute is a list of 2-tuples of the ENNI Coding Function and ENNI Optical Interface Function. Reference MEF 64 Section 8.1.2 ENNI List of Physical Layers Common Attribute. |
| l1NniCodingFunction | L1NniCodingFunction | 1 | ENNI Coding function is a 3-tuple of the <k, OTUk OH, HO ODUk OH>.   * k is an index repre-senting the physical layer line rate. * OTUk OH is a list of overhead values corresponding to the terminated OTUk. * HO ODUk OH is either None or List where the value represents the overhead values corresponding to the terminated HO ODUk. |

Table -L1NniPhysicalLayer Service Attributes

### L1NniOpticalInterfaceFunction

The L1 NNI Physical layer Service Attribute is a list of 2-tuples of the ENNI Coding Function and ENNI Optical Interface Function. Reference MEF 64 Section 8.1.2 ENNI List of Physical Layers Common Attribute.

*oneOf:*

* Otu1OpticalInterfaceFunction
* Otu2AndEOpticalInterfaceFunction
* Otu3OpticalInterfaceFunction
* Otu4OpticalInterfaceFunction

### L1NniCodingFunction

ENNI Coding function is a 3-tuple of the <k, OTUk OH, HO ODUk OH>. k is an index repre-senting the physical layer line rate. OTUk OH is a list of overhead values corresponding to the terminated OTUk. HO ODUk OH is either None or List where the value represents the overhead values corresponding to the terminated HO ODUk.

| **Schema File Name: 1Cs/l1Enni.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| otukOverHead | OtuKOverHead | 1..\* | A list of overhead values corre-sponding to the terminated OTUk. |
| highOrderOdukOverhead | HighOrderOduOverHead | 1..\* | The overhead values corresponding to the terminated HO ODUk (or SHO ODUk), where each entry in the list has the value Disabled or Enabled. |
| lineRate | LineRate | 1 | Enumeration representing physical line rate. |

Table -L1NniCodingFunction Service Attributes

### OtuKOverHead

OTUk Overhead must be a list of three pairs {field,values} with each field and corresponding values in MEF 64 Section 8.1.2 ENNI List of Physical Layers Common Attribute Table 3 - OTUk Overhead Data Type Attributes.

| **Schema File Name: 1Cs/l1Enni.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| otukTti | EnabledDisabled | 1 | OTUk Trail Trace Identifier. |
| otukGcc0 | EnabledDisabled | 1 | OTUk General Communications Channel. |
| otukOsmc | EnabledDisabled | 1 | OTUk OTN Synchronization Messaging Channel. |

Table -OtuKOverHead Service Attributes

### HighOrderOduOverHead

The value of HO ODUk OH is either None or List of overhead values corresponding to the terminated HO ODUk, where each entry in the list has the value Disabled or Enabled.

| **Schema File Name: 1Cs/l1Enni.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| superHighOrderHighOrderOduTti | EnabledDisabled | 1 | Super High Order/High Order ODUk Trail Trace Identifier. |
| superHighOrderHighOrderOduKGcc1 | EnabledDisabled | 1 | Super High Order/High Order ODUk General Communications Channel 1. |
| superHighOrderHighOrderOduKGcc2 | EnabledDisabled | 1 | Super High Order/High Order ODUk General Communications Channel 2. |
| superHighOrderHighOrderOduAps | EnabledDisabled | 1 | Super High Order/High Order ODUk Automatic Protection Switching. |

Table -HighOrderOduHead Service Attributes

### NniProtection

Enumeration representing the protection protocol employed at ENNI for the ODU container exchanged by the operator. The enumeration value is either *None* or one of the rows as specified in G.873.

Contains Enumeration Literals:

* NONE
* 1\_PLUS\_1\_UNIDIR\_SNC\_I
* 1\_PLUS\_1\_BIDIR\_SNC\_I
* 1\_TO\_N\_BIDIR\_SNC\_I
* 1\_PLUS\_1\_UNIDIR\_SNC\_NE
* 1\_PLUS\_1\_BIDIR\_SNC\_NE
* 1\_PLUS\_1\_UNIDIR\_SNC\_NS
* 1\_PLUS\_1\_BIDIR\_SNC\_NS
* 1\_PLUS\_1\_UNIDIR\_SNC\_S
* 1\_PLUS\_1\_BIDIR\_SNC\_S
* 1\_TO\_N\_BIDIR\_SNC\_S
* 1\_PLUS\_1\_UNIDIR\_CL-SNCG\_1
* 1\_PLUS\_1\_BIDIR\_CL-SNCG\_1
* 1\_TO\_1\_BIDIR\_CL-SNCG\_1

## L1EnniService

The ENNI is a reference point representing the boundary between two or more Operator CENs that are operated as separate administrative domains. For each instance of an ENNI, there are multiple sets of ENNI Service Attributes. The value for each ENNI Service Attribute in a set for an Operator CEN is specific to a SP/SO that is using the ENNI.

A screenshot of a diagram

Description automatically generated

Figure -L1EnniService Model

| **Schema File Name: l1Cs/EnniService.yaml**  **$id: urn:mef:lso:spec:service:l1-enni-service:v0.0.2:all** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| identifier | String  *Min length=1*  *Max length=45*  *Pattern=pattern: "[\x20-\x7F]+"* | 1 | The Operator ENNI Identifier Service Attribute value is a string used to allow the SP/SO and Operator to uniquely identify the ENNI. Reference MEF 64 Section 8.2.1 Operator ENNI Identifier Service Attribute. |
| pathOverHead | PathOverHead | 1 | An ODU path is the connectivity between the locations where the path overhead is terminated. Either None or List. When the value of the Operator Path Overhead Service Attribute is List, the entries are the overhead values corresponding to each of the SHO/HO/LO ODU paths carried across an ENNI which is terminated in an Operator’s network. Reference MEF 64 Section 8.2.3 Operator Path Overhead Service Attribute. |
| multiplexingCapabilityList | one of:   * HighOrderOdu4MultiplexingSequences * HighOrderOdu3MultiplexingSequences * HighOrderOdu2And2eMultiplexingSequences * HighOrderOdu1MultiplexingSequences | | Indicates the Operator's ability to mulitiplex a given LO ODUj into a HO ODUk (single-stage), or multiplex a given LO ODUi into a HO ODUj and into a SHO ODUk (two-stage), or more multiplexing stages. Reference MEF 64 Section 8.2.2 Operator Multiplexing Capability List Service Attribute. |

Table -L1EnniService Attributes

### PathOverHead

An ODU path is the connectivity between the locations where the path overhead is terminated. Either None or List. When the value of the Operator Path Overhead Service Attribute is List, the entries are the overhead values corresponding to each of the SHO/HO/LO ODU paths carried across an ENNI which is terminated in an Operator’s network. Reference MEF 64 Section 8.2.3 Operator Path Overhead Service Attribute.

| **Schema File Name: l1Cs/EnniService.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| oduTti | EnabledDisabled | 1 | Overhead value, corresponding to each of the SHO/HO/LO ODU paths carried across an ENNI, ODU TTI is enabled or disabled. |
| oduGcc1 | EnabledDisabled | 1 | Overhead value, corresponding to each of the SHO/HO/LO ODU paths carried across an ENNI, ODU GCC1 is enabled or disabled. |
| oduGcc2 | EnabledDisabled | 1 | Overhead value, corresponding to each of the SHO/HO/LO ODU paths carried across an ENNI, ODU GCC2 is enabled or disabled. |
| oduAps | EnabledDisabled | 1 | Overhead value, corresponding to each of the SHO/HO/LO ODU paths carried across an ENNI, ODU APS is enabled or disabled. |

Table -PathOverHead Attributes

# Common Classes and Types

This section is structure like the previous section but focuses on common classes and types used by the Service Attributes. Most of these are structured to support a variety of L1 Services. This section details the data types and enumerations that are used by the L1 Service model.

## ClientProtocol

Enumeration representing client protocol of L1 physical layer.

Contains Enumeration Literals:

* ETHERNET
* FIBERCHANNEL
* SDH
* SONET

## EnabledDisabled

Enumeration used to indicate state as ENABLED or DISABLED.

Contains Enumeration Literals:

* ENABLED
* DISABLED

## Eth1000BaseXOpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* SX-PMD
* LX-PMD
* LX10-PMD
* BX10-PMD

## Eth10GBaseWOpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* LW-PWD
* EW-PWD

## Eth10GBaseROpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* LR-PMD
* ER-PMD

## Eth100GBaseROpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* LR4-PMD
* ER4-PMD

## Eth40GBaseROpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* LR4-PMD
* ER4-PMD
* FR-PMD

## Fc100OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-PI-2-100-SM-LC-L

## Fc200OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-PI-2-200-SM-LC-L

## Fc400OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-PI-5-400-SM-LC-L
* FC-PI-5-400-SM-LC-M

## Fc800OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-PI-5-800-SM-LC-L
* FC-PI-5-800-SM-LC-I

## Fc1200OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-10GFC-1200-SM-LL-L

## Fc1600OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-PI-5-1600-SM-LC-L
* FC-PI-5-1600-SM-LZ-I

## Fc3200OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* FC-PI-6-3200-SM-LC-L

## L1UniCodingFunction

MEF 63 Coding Function <c>, functionality which encodes bits for transmission and corresponding decode upon reception.

Contains Enumeration Literals:

* 1000BASE-X
* 10GBASE-W
* 10GBASE-R
* 40GBASE-R
* 100GBASE-R
* FC-100
* FC-200
* FC-400
* FC-800
* FC-1200
* FC-1600
* FC-3200
* STM-1
* STM-4
* STM-16
* STM-64
* STM-256
* OC-3
* OC-12
* OC-48
* OC-192
* OC-768

## L1UniOpticalInterfaceFunction

Optical Interface Functional value. Reference MEF 63 Section 8.1.2.

oneOf:

* Eth1000BaseXOpticalInterfaceFunction
* Eth10GBaseWOpticalInterfaceFunction
* Eth10GBaseROpticalInterfaceFunction
* Eth100GBaseROpticalInterfaceFunction
* Eth40GBaseROpticalInterfaceFunction
* Fc100OpticalInterfaceFunction
* Fc200OpticalInterfaceFunction
* Fc400OpticalInterfaceFunction
* Fc800OpticalInterfaceFunction
* Fc1200OpticalInterfaceFunction
* Fc1600OpticalInterfaceFunction
* Fc3200OpticalInterfaceFunction
* Stm1OpticalInterfaceFunction
* Stm4OpticalInterfaceFunction
* Stm16OpticalInterfaceFunction
* Stm64OpticalInterfaceFunction
* Stm256OpticalInterfaceFunction
* Oc3OpticalInterfaceFunction
* Oc12OpticalInterfaceFunction
* Oc48OpticalInterfaceFunction
* Oc192OpticalInterfaceFunction
* Oc768OpticalInterfaceFunction

## L1VcEndPointExternalInterfaceType

Enumeration used to indicate if the L1VC end point is either UNI or ENNI.

Contains Enumeration Literals:

* UNI
* ENNI

## L1UniPhysicalLayer

The L1 Physical Layer Service Attribute specifies the Client Protocol, the Coding Function and the Optical Interface Function. Reference MEF 63 Section 8.1.2 Physical Layer Service Attribute.

| **Schema File Name: l1Cs/l1CsCommon.yaml** | | | |
| --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| clientProtocol | ClientProtocol | 1 | Pointer to ClientProtocol. |
| L1UniCodingFunction | L1UniCodingFunction | 1 | Pointer to L1UniCodingFunction. |
| L1UniOpticalInterfaceFunction | L1UniOpticalInterfaceFunction | 1 | Pointer L1UniOpticalInterfaceFunction. |

Table -L1UniPhysical Service Attributes

## LineRate

Enumeration representing physical line rate.

* OTU1: SONET OC-48 or STM-16
* OTU2: SONET OC-192, STM-64 or 10GBASE-W
* OTU2e: 10Gigabit Ethernet LAN
* OTU3: SONET OC-768, STM-256 or 40 Gigabit Ethernet
* OTU4: 100 Gigabit Ethernet

Contains Enumeration Literals:

* OTU1
* OTU2
* OTU2E
* OTU3
* OTU4

## Oc3OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* SR-1
* IR-1
* IR-2
* LR-1
* LR-2
* LR-3

## Oc12OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* SR-1
* IR-1
* IR-2
* LR-1
* LR-2
* LR-3
* VR-1
* VR-2
* VR-3
* UR-2
* UR-3

## Oc48OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* SR-1
* IR-1
* IR-2
* LR-1
* LR-2
* LR-3
* VR-2
* VR-3
* UR-2
* UR-3

## Oc192OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* SR-1
* SR-2
* IR-1
* IR-2
* IR-3
* LR-1
* LR-2
* LR-2a
* LR-2b
* LR-2c
* LR-3
* VR-2a
* VR-2b
* VR-3

## Oc768OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* SR-1
* SR-2
* IR-1
* IR-2
* IR-3
* LR-1
* LR-2
* LR-3

## Stm10OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* I-1
* S-1.1
* S-1.2
* L-1.1
* L-1.2
* L-1.3

## Stm4OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* I-4
* S-4.1
* S-4.2
* L-4.1
* L-4.2
* L-4.3

## Stm16OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* I-16
* S-16.1
* S-16.2
* L-16.1
* L-16.2
* L-16.3

## Stm64OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* I-64.LR
* I-64.I
* I-64.2R
* I-64.3
* I-64.5
* S-64.1
* S-64.2
* S-64.3
* S-64.5
* L-64.1
* L-64.2
* L-64.3

## Stm256OpticalInterfaceFunction

Optical Interface Function value. Reference MEF 63 Section 8.1.2.

Contains Enumeration Literals:

* VSR2000-3R1
* VSR2000-3R2
* VSR2000-3R3
* VSR2000-3R5
* VSR2000-3M1
* VSR2000-3M2
* VSR2000-3M3
* VSR2000-3M5
* VSR2000-3H2
* VSR2000-3H3
* VSR2000-3H5

# L1 Service Level Specification

The Subscriber L1VC Service Level Specification (SLS) Service Attribute is the technical specification of aspects of the service performance agreed to by the Service Provider and the Sub-scriber. For any given SLS, a given Performance Metric may or may not be specified.

The value of the Subscriber L1VC SLS Service Attribute is either None or a 3-tuple of the form (t-s, T, PM) where:

1) t-s is a time that represents the date and time for the start of the SLS.

2) T is a duration that is used in conjunction with t-s to specify a contiguous sequence of time intervals for determining when performance objectives are met. The units for T are not constrained. For example, a calendar month is an allowable value. Since the duration of a month varies it could be specified as, e.g. from midnight on the 10th of one month up to but not including midnight on the 10th of the following month.

3) PM is a list where each element in the list consists of a Performance Metric Name, a list of parameter values specific to the definition of the Performance Metric, and Performance Metric Objective.

The following performance metrics are supported as part of an SLS:

1. One-way Delay Performance Metric (Sls1wDelayPerformanceMetric)
2. One-way Errored Second Performance Metric (Sls1wErroredSecondPerformanceMetric)
3. One-way Severely Errored Second Performance Metric (Sls1wSeverelyErroredSecondPerformanceMetric)
4. One-way Unavailable Second Performance Metric (Slsw1UnavailabilitySecondPerformanceMetric)
5. One-way Availability Performance Metric (Slsw1AvailabilityPerformanceMetric)

## L1ServiceLevelSpecification

The Subscriber L1VC Service Level Specification (SLS) Service Attribute is the technical specification of aspects of the service performance agreed to by the Service Provider and the Subscriber. For any given SLS, a given Performance Metric may or may not specified.

A screenshot of a computer

Description automatically generated

Figure -Service Level Specification Model

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| startTime | TimeAndDate | 1 | StartTime is a time that represents the date and time for the start of the SLS. MEF 63: [R22] t-start MUST be specified to the nearest second. MEF 64: [R30] t-start MUST be specified to the nearest second. |
| duration | TimeIntervalT | 1 | Duration is a duration that is used in conjunction with ts to specify a contiguous sequence of time intervals for determining when performance objectives are met. The units for T are not constrained. |
| sls1wDelayPerformanceMetric | Sls1wDelayPerformanceMetric | 0..2 | The One-way Delay for the L1CI that ingresses at UNI 1 and that egresses at UNI 2 is defined as the time elapsed from the reception of the first bit of the ingress L1CI at UNI 1 until the reception of that first bit of the corresponding L1CI egressing at UNI 2. |
| sls1wErroredSecondPerformanceMetric | Sls1wErroredSecondPerformanceMetric | 0..2 | An errored second (ES) is defined as one second sigma-k in Available Time with at least one errored block (EB) and is not a SES. An EB is defined as a block in which one or more bits are in error. |
| sls1wSeverelyErroredSecondPerformanceMetric | Sls1wSeverelyErroredSecondPerformanceMetric | 0..2 | An errored second (ES) is defined as one second sigma-k in Available Time with at least one errored block (EB) and is not a SES. An EB is defined as a block in which one or more bits are in error. |
| sls1wUnavailableSecondPerformanceMetric | Sls1wUnavailableSecondPerformanceMetric | 0..2 | An Unavailable Second (UAS) is defined as a second during Unavailable Time (UAT). |
| sls1wAvailablityPerformanceMetric | Sls1wAvailablityPerformanceMetric | 0..2 | Availability is defined as the percentage of Available Time over a given interval T-l which does not include Maintenance Interval Time (MIT). |

Table 22-L1ServiceLevelSpecification Attributes

## L1VcEndPointRef

First reference VC, then specific VC End Point.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| vcRef | L1VcRef | 1..\* | Pointer to L1VcRef. |
| id | String  Format: uuid | 1 | Points to L1VcEndPoint. |

Table -L1VcEndPointRef Attributes

## L1VcRef

Reference to a L1 Virtual Connection Instance.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| id | String | 1..\* | VC reference. |
| href | String  Format: uri | 1 | VC URI reference. |

Table -L1VcEndPointRef Attributes

## Sls1wDelayPerformanceMetric

The One-way Delay for the L1CI that ingresses at UNI 1 and that egresses at UNI 2 is defined as the time elapsed from the reception of the first bit of the ingress L1CI at UNI 1 until the reception of that first bit of the corresponding L1CI egressing at UNI 2. Reference MEF 63 Section 8.2.3.3 One-way Delay Performance Metric.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| orderedPairs | OrderedPair | 1..\* | A non-empty subset of the ordered pairs of OVC/EVC End Points. |
| oneWayFdPercentile | Percentage | 1 | Frame Delay percentile. |
| oneWayFdObjective | Time | 1 | Frame Delay objective. |

Table -Sls1wDelayPerformanceMetric Attributes

## Sls1wErroredSecondPerformanceMetric

An errored second (ES) is defined as one second sigma-k in Available Time with at least one errored block (EB) and is not a SES. An EB is defined as a block in which one or more bits are in error. Reference MEF 63 Section 8.2.3.4 One-way Errored Second Performance Metric.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| orderedPairs | OrderedPair | 1..\* | A non-empty subset of the ordered pairs of OVC/EVC End Points. |
| timeDuration | Time | 1 | Time duration. |
| oneWayIfdvPercentile | Percentage | 1 | Inter-Frame Delay Variation percentile. |
| oneWayIfdvObjective | Time | 1 | Inter-frame Delay Variation objective. |

Table -Sls1wErroredSecondPerformanceMetric Attributes

## Sls1wSeverelyErroredSecondPerformanceMetric

An errored second (ES) is defined as one second sigma-k in Available Time with at least one errored block (EB) and is not a SES. An EB is defined as a block in which one or more bits are in error. Reference MEF 63 Section 8.2.3.5 One-way Severely Errored Second Performance Metric.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| orderedPairs | OrderedPair | 1..\* | A non-empty subset of the ordered pairs of OVC/EVC End Points. |
| oneWayMfdObjective | Time | 1 | Mean Frame Delay objective. |

Table -Sls1wSeverelyErroredSecondPerformanceMetric Attributes

## Sls1wUnavailabilitySecondPerformanceMetric

An Unavailable Second (UAS) is defined as a second during Unavailable Time (UAT). Reference MEF 63 Section 8.2.3.6 One-way Unavailable Second Performance Metric.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| orderedPairs | OrderedPair | 1..\* | A non-empty subset of the ordered pairs of OVC/EVC End Points. |
| oneWayFdrPercentile | Percentage | 1 | Frame Delay Range percentile. |
| oneWayFdrObjective | Time | 1 | Frame Delay Range objective. |

Table -Sls1wUnavailabilitySecondPerformanceMetric Attributes

## Sls1wAvailabilityPerformanceMetric

Availability is defined as the percentage of Available Time over a given interval T-l which does not include Maintenance Interval Time (MIT). Reference MEF 63 Section 8.2.3.7 One-way Availability Performance Metric.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| orderedPairs | OrderedPair | 1..\* | A non-empty subset of the ordered pairs of OVC/EVC End Points. |
| oneWayFrameLossRatioObjective | Percentage | 1 | Frame Loss Ratio objective. |

Table -Sls1wAvailabilityPerformanceMetric Attributes

## Time

This data type is for the Time and Date in UTC.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| day | Integer | 1 | Denotes the day. |
| hour | Integer | 1 | Denotes the hour. |
| month | Integer | 1 | Denotes the month. |
| second | Integer | 1 | Denotes the second. |
| year | Integer | 1 | Denotes the year. |

Table -Time Attributes

## TimeIntervalT

This attribute sets the time interval to evaluate the performance for the SLS. All performance of this SLS use the same time interval T, which itself may not be constrained, e.g., 1 month.

|  |  |  |  |
| --- | --- | --- | --- |
| **Schema File Name: l1Cs/l1ServiceLevelSpecification.yaml** | | | |
| **Attribute Name** | **Type** | **Multiplicity** | **Description** |
| number | Integer | 1 | This denotes the value (for the unit). |
| unit | String  Enum:   * second * minute * hour * day * month * week * year | 1 | Time interval unit. |

Table -TimeIntervalT Attributes

Figure 20 below provides an example of the value for the L1 VC Service Level Specification Attribute. Some observations:

* A single value of startTime and a single value of duration apply to all Performance Metrics.
* The SLS is defined with a single instance of each of the Performance Metrics between an Ordered Pair:
  + One-way Delay
  + One-way Errored Seconds
  + One-way Severely Errored Seconds
  + One-way Unavailable Seconds
  + One-way Availability

A screenshot of a computer

Description automatically generated

Figure -L1 VC Service Level Specification Service Attribute Value Example

{

"startTime" : {

"day" : 18,

"hour" : 13,

"minute" : 30,

"month" : 11,

"year" : 2024

},

"duration" : {

"number" : 15,

"unit" : "minute"

},

"sls1wDelayPerformanceMetric" : [ {

"percentile" : 95,

"delayObjective" : {

"time" : 10,

"timeUnits" : "milliSeconds"

},

"orderedPairSrc" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "0e87f431-3951-4386-b059-99dbf2267802"

},

"orderedPairDst" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "ecbfbaec-f717-4d37-ba5e-ec6970d0151c"

}

} ],

"sls1wErroredSecondPerformanceMetric" : [ {

"erroredSecondObjective" : 1,

"orderedPairSrc" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "0e87f431-3951-4386-b059-99dbf2267802"

},

"orderedPairDst" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "ecbfbaec-f717-4d37-ba5e-ec6970d0151c"

}

} ],

"sls1wSeverelyErroredSecondPerformanceMetric" : [ {

"severelyErroredSecon-dObjective" : 1,

"orderedPairSrc" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "0e87f431-3951-4386-b059-99dbf2267802"

},

"orderedPairDst" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "ecbfbaec-f717-4d37-ba5e-ec6970d0151c"

}

} ],

"sls1wUnavailableSecondPerformanceMetric" : [ {

"unavailableSecondObjective" : 1,

"orderedPairSrc" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "0e87f431-3951-4386-b059-99dbf2267802"

},

"orderedPairDst" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "ecbfbaec-f717-4d37-ba5e-ec6970d0151c"

}

} ],

"sls1wAvailabilityPerformanceMetric" : [ {

"availabilityObjective" : 99,

"orderedPairSrc" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "0e87f431-3951-4386-b059-99dbf2267802"

},

"orderedPairDst" : {

"vcRef" : {

"id" : "cfc1f664-ac80-4247-9d64-6bd7d37812d3"

},

"id" : "ecbfbaec-f717-4d37-ba5e-ec6970d0151c"

}

} ]

}

Figure -L1 SLS JSON Example

Appendix A Usage examples (Informative)

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

* Different Service Order configuration variants,
* Basic Service Order API walkthrough to order a L1 Service,
* Common modifications,
* Deletion of a L1 Service.

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.
  1. High-level Flow

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

A diagram of a service order

Description automatically generated

Figure -Service End-to-End Function Flow

Service Catalog – allows the Client to query SOF for available Services as well as what attributes are fixed and/or elastics with values/ranges.

Service Order – allows the Client to request the SOF to initiate and complete the fulfillment process of installation of a Service Offering, an update to an existing Service, or a disconnect of an existing Service.

Service Inventory – allows the Client to retrieve information about existing Service instances from the SOF’s Service Inventory.

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.

* 1. Integration of Service Specification into the Service Order API

The Service Order API is service-agnostic in the meaning that they serve as an interaction between the Client and the Server (SOF) and they do not contain any service-specific information in their specifications. To pass the service-specific information, an extension pattern is used. This applies to any of the Legato Service APIs that carry service-specific information: Service Catalog, Service Order and Service Inventory [10].

The extension hosting type in the API data model is MefServiceConfiguration. The @type attribute of that type must be set of a value that uniquely identifies the service specification. See Figure 23 and Figure 24. A unique identifier for MEF standard service 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:service:l1-subscriber-uni:v0.0.2:all** the above-mentioned URN:

**$id: urn:mef:lso:spec:service:l1-subscriber-uni:v0.0.1:all**

**$schema: http://json-schema.org/draft-07/schema#**

**title: MEF LSO Service – L1 Subscriber UNI Specification**

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

* urn:mef:lso:spec:service:l1-subscriber-uni:v0.0.2:order
* urn:mef:lso:spec:service:l1-subscriber-vc:v.0.0.2:order
* urn:mef:lso:spec:service:l1-subscriber-vc-end-point:v.0.0.2:order

A screenshot of a computer

Description automatically generated

Figure -Extension Pattern: L1 Subscriber Service-Specific Extensions

* urn:mef:lso:spec:service:l1-operator-uni:v0.0.2:order
* urn:mef:lso:spec:service:l1-operator-vc:v.0.0.2:order
* urn:mef:lso:spec:service:l1-operator-vc-end-point:v.0.0.2:order
* urn:mef:lso:spec:service:l1-enni:v.0.0.2:order
* urn:mef:lso:spec:service:l1-enni-service:v.0.0.2:order

A diagram of a computer

Description automatically generated with medium confidence

Figure -Extension Pattern: L1 Operator Service-Specific Extensions

* 1. Use Case 1: Create Service Order

The ServiceOrderCreate process is initiated by the Buyer/Client with the determination of the one or more ServiceOrderItems that will be part of a Service Order using the ServiceOrderCreate. Each Service Order Item through the Extension pattern is associated with a MEF Service specific payload component (i.e., L1SubscriberUni, L1VcEndPoint, L1Vc).

The Buyer/Client sends a request with a ServiceOrderCreate type in the body. The SOF performs request validation, assigns an id, and returns ServiceOrder type in the response body, with a state set to acknowledged. From this point, the ServiceOrder 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 ServiceOrder. The two patterns are illustrated in sequence diagrams below.

A screenshot of a computer

Description automatically generated

Figure -Service Order progress tracking – Notifications (Asynchronous)

A screenshot of a computer program

Description automatically generated

Figure -Service Order progress tracking – Polling (Synchronous)

* + 1. Create Service Order Request

A screenshot of a computer

Description automatically generated

Figure -Service Order Request

Entities use for providing a response to a ServiceOrderCreate request are presented in Figure 27. The main types used for response are ServiceOrder and ServiceOrderItem, which add attributes set by the SOF (like id or state) ServiceOrder is the root entity of a response. The response echoes back all attributes as provided by the BUS/CLIENT and contains the same number of ServiceOrderItems as in the request. The following snippet presents the SOF’s response including an example L1CS payload.

This section guides through all the steps of Service Order API that is needed to be performed to successfully order a L1 Subscriber UNI service.

NOTE: SOF is free to mandate some of these steps.

NOTE: As the examples of steps in many cases will replicate the service-specific information, in some of the snippets some parts of it will be omitted for better readability.

There are rules for all request items for creation requests (Service Order):

* item.action must be set to add
* item.service.id must not be provided.
* service.serviceConfiguration must contain all desired configurations
  + 1. Create Service Order Response

The SOF response to the CreateServiceOrder is a ServiceOrder which is shown below. The main types used for the response are ServiceOrder and ServiceOrderItem. The one or more ServiceOrderItems will reference a specific L1CS resource as part of the payload and association(s) to other ServiceOrderItems as an attribute in the envelope. The response echoes back all the attributes as provided by the Client/Buyer and contains the same number of ServiceOrderItems as in the request.

{

"coordinatedAction": [

{

"coordinatedActionDelay": {

"amount": 10,

"units": "businessMinutes"

},

"coordinationDependency": "startToFinish",

"orderId": "0000000-2222-5555-00000000123"

}

],

"description": "l1-cs-so-0001",

"note": [

{

"author": "admin",

"date": "2024-11-17T02:15:00Z",

"id": "0000000-2222-5555-00000000123",

"source": "bus",

"text": "automatic\_so"

}

],

"orderRelationship": [

{

"serviceOrder": {

"href": "0000000-2222-5555-00000000123",

"id": **null**

},

"relationshipType": **null**

}

],

"relatedContactInformation": [

{

"emailAddress": "admin@sof.com",

"name": "admin",

"number": "651-555-0000",

"organization": "SOF Co.",

"postalAddress": {

"city": "Duluth",

"country": "USA",

"streetName": "100 Postal Street"

},

"role": **null**

}

],

"requestedCompletionDate": "2024-11-17T02:15:00Z",

"requestedStartDate": "2024-11-17T02:15:00Z",

"href": "{{baseUrl}}/serviceOrder/0000000-2222-5555-00000000123",**<< added by SOF >>**

"id": "0000000-2222-5555-00000000123",

"serviceOrderItem": [

{

"id": **null**,

"action": "add",

"coordinatedAction": **null**,

"note": **null**,

"service": {

"note": **null**,

"serviceRelationship": **null**,

"relatedContactInformation": **null**,

"place": **null**,

"serviceConfiguration": {

"@type": "urn:mef:lso:spec:service:l1-subscriber-uni:v0.0.2:all",

"identifier": "l1subscriberuni-100"

}

},

"serviceOrderItemRelationship": [

{

"orderItem": **null**,

"relationshipType": "L1\_UNI\_ENDPOINT\_OF\_VC"

}

],

"state": **null**,

"terminationError": **null**

}

],

"state": "acknowledged", **<< added by SOF >>**

"orderDate": "2024-11-17T02:15:00Z"

}

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.

A screenshot of a computer program

Description automatically generated

Figure -Service Order Response

The set of requirements for the Service Order Response are specified in [8].

* 1. Use Case 2: Service Order Item to Modify Existing Service

The following use case shows a request for an order for an existing L1 VC Subscriber Service modification (action equal to modify). A change to L1 Service Level Specification objective is performed.

The following requirements based on [8] apply to serviceOrderItem when action is modify:

1. The modify request **MUST** specify a reference (provide service.id) to an existing service that is a subject of this order and provide the desired service.configuration.
2. The modify request **MUST** repeat the same values (specified or empty) of service.relationship, and service.place as they are available in the inventory for a given service instance. These values cannot be update or deleted.
3. If there is a relationship with another Service Order Item, the serviceOrderItemRelationship **MUST** be also specified unchanged.

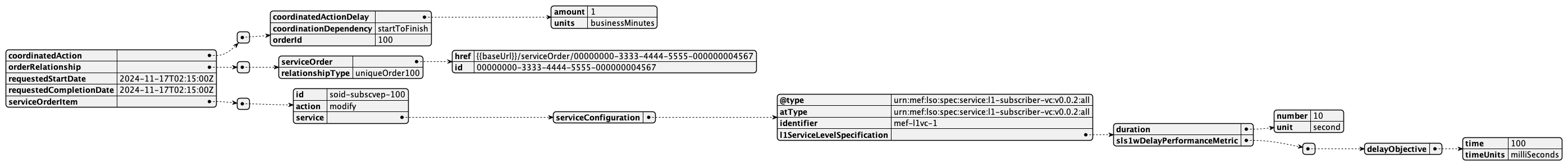


Figure -Service Order Modify Existing Service Request

A screenshot of a computer program

Description automatically generated

Figure 30-Service Order Modify Existing Service Response

* 1. Use Case 3: Service Order Item to Delete Existing Service

The following use case represents a single Service Order request for deletion (action=delete) of an existing service identified by serviceOrderItem.id.

{

"description" : "Example Service Order to Delete L1 VC End Point Service",

"externalId" : "serviceOrder-100",

"requestedCompletionDate" : "2024-11-17T02:15:00Z",

"requestedStartDate" : "2024-11-17T02:15:00Z",

"serviceOrderItem" : [ {

"id" : "so-item-001",

"action" : "delete",

"service" : {

"id" : "00000000-5555-6666-7777-000000009999"

}

} ]

}

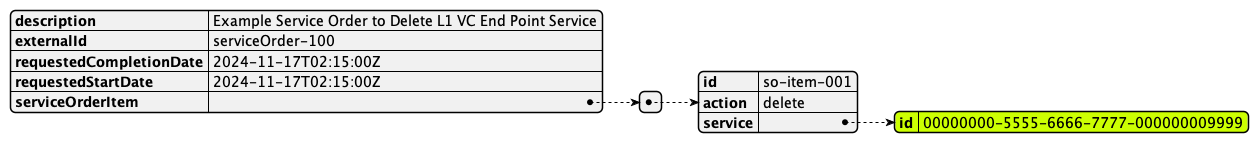


Figure -Service Order to Delete Existing Service

1. service.id **MUST** be provided.
2. The Buyer/Client **MUST NOT** provide any service attributes other than service.id.
   1. Use Case 4: Retrieve List of Service Orders

The Buyer/Client can retrieve a list of ServiceOrders by using a GET /serviceOrders operation with desired filtering criteria. Reference [8] for complete set of mandatory and optional requirements.

1. The Buyer’s/Client’s request MAY contain none or more of the following attributes:

* state
* orderDate.gt
* orderDate.lt
* completionDate.gt
* completionDate.lt
* expectedCompletionDate.gt
* expectedCompletionDate.lt
* startDate.gt
* startDate.lt
  1. Use Case 5: Retrieve Service Order by Service Order Identifier

The Buyer/Client can get detailed information about the Service Order from the SOF by using a GET/serviceOrder/{{id}} operation. The payload returned in the response includes all attributes the Buyer/Client has provided when a Service Order create request was sent. The attributes provided by the SOF depend on the state of the ServiceOrder and may require some additional processing and time to be set.

* 1. Use Case 6: Register for 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 registerListerner 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=serviceOrderCreateEvent,serviceOrderItemStateChangeEvent,serviceOrderInformationRequiredEvent,serviceOrderItemStateChangeEvent"

}

Below is corresponding response from Server:

{

"callback": "https://client.com/listenerEndpoint",

"id": "1e29afb6-6d91-48a1-b4a7-5bca047f8764",**<< added by SOF >>**

"query": "eventType=serviceOrderCreateEvent,serviceOrderItemStateChangeEvent,serviceOrderInformationRequiredEvent,serviceOrderItemStateChangeEvent"

}

* 1. Use Case 7: 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:

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

Below is corresponding response from Server:

{

"callback": "eventType=serviceOrderCreateEvent,serviceOrderItemStateChangeEvent,serviceOrderInformationRequiredEvent,serviceOrderItemStateChangeEvent",

"id": "1e29afb6-6d91-48a1-b4a7-5bca047f8764"

}

* 1. Use Case 8: Send Notification

The Buyer/Client Supports a notification mechanism. The notification mechanism used by the SOF is REST-based and notifications are transmitted by the SOF using a POST /listener/[serviceOrderCreate|serviceOrderStateChangeEvent|serviceOrderInformationRequiredEvent|serviceOrderItemStateChangeEvent].

# References

1. IETF [JSON Schema draft 7](https://json-schema.org/specification-links.html#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 63 Subscriber Layer 1 Service Attributes, August 2018.
6. MEF 64 Operator Layer 1 Service Attributes and Services, February 2020.
7. MEF 55.1, Lifecycle Service Orchestration (LSO): Reference Architecture and Framework, January 2021.
8. MEF 99 LSO Service Ordering Management API-Developer Guide, October 2023.