

Ethernet PHY  
Information Model

Version 1.0.0-2

7th of March 2019

ONF TR‑541

ONF Document Type: Technical Recommendation

ONF Document Name: Ethernet PHY Information Model Version 1.0

Disclaimer

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

Any marks and brands contained herein are the property of their respective owners.

Open Networking Foundation

1000 El Camino Real, Suite 100, Menlo Park, CA 94025

[www.opennetworking.org](http://www.opennetworking.org)

©2017 Open Networking Foundation. All rights reserved.

Open Networking Foundation, the ONF symbol, and OpenFlow are registered trademarks of the Open Networking Foundation, in the United States and/or in other countries. All other brands, products, or service names are or may be trademarks or service marks of, and are used to identify, products or services of their respective owners.

Table of Contents

[Disclaimer 2](#_Toc2841690)

[Open Networking Foundation 2](#_Toc2841691)

[Table of Contents 3](#_Toc2841692)

[List of Figures 5](#_Toc2841693)

[List of Tables 5](#_Toc2841694)

[Document History 6](#_Toc2841695)

[1 Introduction 6](#_Toc2841696)

[2 Definitions 7](#_Toc2841697)

[2.1 Terms 7](#_Toc2841698)

[2.2 Abbreviations and Acronyms 7](#_Toc2841699)

[3 Compliance Statement 7](#_Toc2841700)

[4 Overview of the Ethernet PHY Information Model 8](#_Toc2841701)

[4.1 Coverage 8](#_Toc2841702)

[4.2 Overview 8](#_Toc2841703)

[5 Conventions 10](#_Toc2841704)

[5.1 UML Modeling Conventions 10](#_Toc2841705)

[5.2 Default Values 11](#_Toc2841706)

[5.3 Comments 11](#_Toc2841707)

[5.4 ONF Stereotypes 11](#_Toc2841708)

[5.4.1 attributeValueChangeNotification 11](#_Toc2841709)

[5.4.2 objectCreationNotification and objectDeletionNotification 11](#_Toc2841710)

[5.4.3 isInvariant 12](#_Toc2841711)

[5.4.4 valueRange 12](#_Toc2841712)

[5.4.5 partOfObjectKey 12](#_Toc2841713)

[5.4.6 unit 12](#_Toc2841714)

[5.4.7 Support Qualifier 13](#_Toc2841715)

[5.4.8 bitLength 13](#_Toc2841716)

[5.4.9 Unsigned, Encoding and Counter 13](#_Toc2841717)

[6 Special elements 13](#_Toc2841718)

[6.1 PMD Definitions 13](#_Toc2841719)

[6.2 Shared \*\_Pacs 13](#_Toc2841720)

[6.3 Problem Definitions, Associations and Inheritances 14](#_Toc2841721)

[6.4 Performance Values 14](#_Toc2841722)

[6.5 Maintenance Attributes 14](#_Toc2841723)

[6.6 Transceiver Information 14](#_Toc2841724)

[6.7 Expected Transceiver 16](#_Toc2841725)

[7 Attachment to the Core Information Model 17](#_Toc2841726)

[7.1 Attachment of \*\_Pacs 19](#_Toc2841727)

[8 Non-Backward Compatibility of the next Version 19](#_Toc2841728)

[9 Future Work 20](#_Toc2841729)

[10 UML Model Files 20](#_Toc2841730)

[11 YANG Model Files 20](#_Toc2841731)

[12 Interface Simulator 21](#_Toc2841732)

[13 Data Dictionary 21](#_Toc2841733)

[13.1 WireInterface\_Pac 22](#_Toc2841734)

[13.1.1 WireInterface\_Pac 23](#_Toc2841735)

[13.1.2 WireInterfaceConfiguration 24](#_Toc2841736)

[13.1.3 WireInterfaceCurrentProblems 29](#_Toc2841737)

[13.1.4 WireInterfaceCurrentPerformance 30](#_Toc2841738)

[13.1.5 WireInterfaceHistoricalPerformances 30](#_Toc2841739)

[13.1.6 WireInterfaceCapability 31](#_Toc2841740)

[13.1.7 WireInterfaceStatus 36](#_Toc2841741)

[13.2 PureEthernetStructure\_Pac 40](#_Toc2841742)

[13.3 EthernetContainer\_Pac 41](#_Toc2841743)

[13.4 Data Types 42](#_Toc2841744)

[13.4.1 PmdType 42](#_Toc2841745)

[13.4.2 WireInterfaceCurrentPerformanceType 43](#_Toc2841746)

[13.4.3 WireInterfaceCurrentProblemType 43](#_Toc2841747)

[13.4.4 WireInterfaceHistoricalPerformanceType 44](#_Toc2841748)

[13.4.5 WireInterfacePerformanceType 44](#_Toc2841749)

[13.4.6 WireInterfaceProblemSeverityType 46](#_Toc2841750)

[13.5 Enumeration Types 46](#_Toc2841751)

[13.5.1 DuplexType 46](#_Toc2841752)

[13.5.2 LoopBackType 47](#_Toc2841753)

[13.5.3 MdiKindType 47](#_Toc2841754)

[13.5.4 MediumKindType 48](#_Toc2841755)

[13.5.5 MiiKindType 48](#_Toc2841756)

[13.5.6 RxSyncPreferenceType 49](#_Toc2841757)

[13.5.7 RxSyncRoleType 50](#_Toc2841758)

[13.5.8 SeverityType 50](#_Toc2841759)

[13.5.9 SignalOrderingType 50](#_Toc2841760)

[13.5.10 pmdNegotiationStateType 51](#_Toc2841761)

[13.6 Super Classes 51](#_Toc2841762)

[13.6.1 WireCurrentProblem 51](#_Toc2841763)

[13.7 Notifications 53](#_Toc2841764)

[13.7.1 AttributeValueChangedNotification 53](#_Toc2841765)

[13.7.2 ObjectCreationNotification 55](#_Toc2841766)

[13.7.3 ObjectDeletionNotification 56](#_Toc2841767)

[13.7.4 ProblemNotification 57](#_Toc2841768)

[14 References 59](#_Toc2841769)

[15 Back matter 59](#_Toc2841770)

[15.1 Editors 59](#_Toc2841771)

[15.2 Contributors 59](#_Toc2841772)

List of Figures

[Figure 1: Model Overview 9](#_Toc2841773)

[Figure 2: Example for the sub-structuring of \*\_Pacs 10](#_Toc2841774)

[Figure 3: Equipment part of the Core Information Model 15](#_Toc2841775)

[Figure 4: Usage of ONF Core Information Model 1.2 18](#_Toc2841776)

[Figure 5: Associations between LayerProtocol and \*\_Pac 19](#_Toc2841777)

[Figure 6: WireInterface\_Pac 22](#_Toc2841778)

[Figure 7: Exemplary PureEthernetStructure\_Pac 40](#_Toc2841779)

[Figure 8: Exemplary EthernetContainer\_Pac 41](#_Toc2841780)

List of Tables

[Table 1: ManufacturerProperties Class 15](#_Toc2841781)

[Table 2: EquipmentType Class 16](#_Toc2841782)

[Table 3: EquipmentInstance Class 16](#_Toc2841783)

[Table 4: PhysicalProperties Class 16](#_Toc2841784)

[Table 5: Attributes for WireInterface\_Pac 23](#_Toc2841785)

[Table 6: Attributes for WireInterfaceConfiguration 25](#_Toc2841786)

[Table 7: Attributes for WireInterfaceCurrentProblems 29](#_Toc2841787)

[Table 8: Attributes for WireInterfaceCurrentPerformance 30](#_Toc2841788)

[Table 9: Attributes for WireInterfaceHistoricalPerformances 31](#_Toc2841789)

[Table 10: Attributes for WireInterfaceCapability 31](#_Toc2841790)

[Table 11: Attributes for WireInterfaceStatus 36](#_Toc2841791)

[Table 12: Attributes for PmdType 42](#_Toc2841792)

[Table 13: Attributes for WireInterfaceCurrentPerformanceType 43](#_Toc2841793)

[Table 14: Attributes for WireInterfaceCurrentProblemType 44](#_Toc2841794)

[Table 15: Attributes for WireInterfaceHistoricalPerformanceType 44](#_Toc2841795)

[Table 16: Attributes for WireInterfacePerformanceType 44](#_Toc2841796)

[Table 17: Attributes for WireInterfaceProblemSeverityType 46](#_Toc2841797)

[Table 18: Attributes for WireCurrentProblem 52](#_Toc2841798)

[Table 19: Attributes for AttributeValueChangedNotification 53](#_Toc2841799)

[Table 20: Attributes for ObjectCreationNotification 55](#_Toc2841800)

[Table 21: Attributes for ObjectDeletionNotification 56](#_Toc2841801)

[Table 22: Attributes for ProblemNotification 57](#_Toc2841802)

Document History

|  |  |  |
| --- | --- | --- |
| Version | Date | Description of Change |
| 1.0.0-0 | 2017-12-09 | Initial version of the Ethernet PHY Information Model as technology specific extension to the TR-512 ONF Core Information Model [ONF Core IM] |
| 1.0.0-1 | 2019-02-16 | Consolidating all improvements discussed after first implementation and testing during the 5th ONF PoC; provided for review on project level. |
| 1.0.0-2 | 2019-03-05 | Consolidating all feedback provided during the latest review; provided for review again on project level. |
|  |  |  |

# Introduction

This ONF Technical Recommendation (ONF TR‑541) describes a management interface for vendor agnostic management of the Ethernet OSI Layer 1.

It is mainly based on IEEE 802.3 Standard for Ethernet [IEEE802.3], but consolidates also elements of specifications provided by the SFF Committee (now a Technology Affiliate of the Storage Networking Industry Association) and other sources.

The comprised UML information model represents a technology specific extension to the TR-512 ONF Core Information Model 1.2 [ONF Core IM] and it integrates well with the ONF TR‑532v1.1 Microwave Information Model [ONF MW IM] for providing a unified Ethernet transport.

# Definitions

## Terms

The primary purpose of this document is to define terms and hence terms are defined throughout the document. Key terms are highlighted in section 2.2 Abbreviations and Acronyms.

## Abbreviations and Acronyms

| **Term** | **Explanation** |
| --- | --- |
| CWDM | Coarse Wavelength Division Multiplex |
| EFM | Ethernet First Mile |
| EPON | Ethernet Passive Optical Network |
| MAU | Medium Attachment Unit |
| ONF | Open Networking Foundation |
| OSI | Open Systems Interconnection Model |
| \*\_Pac | Technology Specific Conditional Package |
| PHY | Physical Layer |
| PMD | Physical Medium Dependent |
| SDN | Software Defined Network |
| SFF | Small Form Factor |
| TR | ONF Technical Recommendation |

# Compliance Statement

**Completeness**

A device's interface must not be denoted as compliant to this Ethernet PHY Information Model, if it doesn't implement all components.

**Support**

The hardware does not necessarily need to make available all functionalities covered by this modeling. In case some functionality is not available at the hardware, the device's interface shall answer the default values defined in this model. All functionalities, which are available at the hardware and covered by the Ethernet PHY Information Model, must be manageable with this model (to the extent of the comprised attributes).

**Proprietary Extensions**

Interfaces implementing elements of this Ethernet PHY Information Model plus additional components must not be denoted as compliant. Such interfaces might be called "based on" or "expanding" it.

# Overview of the Ethernet PHY Information Model

## Coverage

Basically, the structure of the Ethernet PHY Information Model is capable of covering all kinds of point-to-point Physical Medium Dependent (PMD).

The current set of attributes allows managing at least the following PMDs:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1BASE5 | 100BASE-T4 | 1000BASE-BX10 | 10GBASE-LX4 | 40GBASE-KR4 | 100GBASE-CR4 |
| 10BASE2 | 100BASE-TX | 1000BASE-LX | 10GBASE-CX4 | 40GBASE-CR4 | 100GBASE-KR4 |
| 10BROAD36 | 100BASE-BX10 | 1000BASE-LX10 | 10GBASE-KX4 | 40GBASE-SR4 | 100GBASE-KP4 |
| 10BASE-T | 100BASE-FX | 1000BASE-SX | 10GBASE-ER | 40GBASE-LR4 | 100GBASE-CR10 |
| 10BASE-FP | 100BASE-LX10 | 1000BASE-CX | 10GBASE-LR | 40GBASE-ER4 | 100GBASE-SR4 |
| 10BASE-FB | 100BASE-T2 | 1000BASE-KX | 10GBASE-SR | 40GBASE-FR | 100GBASE-SR10 |
| 10BASE-FL |  | 1000BASE-T | 10GBASE-LRM |  | 100GBASE-LR4 |
|  |  |  | 10GBASE-KR |  | 100GBASE-ER4 |
|  |  |  | 10GBASE-PR |  |  |

In principle, the Ethernet PHY Information Model allows definition of additional PMDs during implementation, but very sophisticated, highest capacity PMDs might require additional attributes, which are not yet included.

The Ethernet PHY Information Model does not cover any components of the Ethernet MAC layer.

The following PMDs are not yet cover, but might become subject to future expansion, if requested:

* Point-to-multipoint Ethernet for Ethernet Passive Optical Network (EPON)  
  like 1000BASE-PX and 10/1GBASE-PRX
* PMDs with variable line speed for Ethernet First Mile (EFM)  
  like 2BASE-TL and 10PASS-TS
* PMDs for Ethernet over SDH  
  like 10GBASE-W

The Ethernet PHY Information Model provides the necessary attributes for

* the device informing the SDN Controller about its capabilities
* the Controller configuring the device
* the device providing status, problem and performance information

## Overview

The Ethernet PHY Information Model consists of the following three layers, which are represented by \*\_Pacs (see also chapter 7.1 Attachment of \*\_Pacs):

**Interface**

The WireInterface\_Pac contains the characteristics of the physical layer of the wire based Ethernet interface. It represents the combination of the interface at the internal Ethernet switch and the "transmitter" (e.g. SFP). It is specified within this document.

**Structure**

Basic purpose of the Structure layer is segmenting the physical resource provided by the Interface layer into logical pieces that can be booked by Containers. In case of nowadays wire based Ethernet interfaces, just a single segment is provided. The Ethernet PHY Information Model is not defining any new Structure, but referencing on the PureEthernetStructure\_Pac defined in the ONF TR‑532v1.1 Microwave Information Model [ONF MW IM].

**Container**

The Container layer offers transport services to higher protocol layers. It represents the upper demarcation of the modeling. The Ethernet PHY Information Model is not defining any new Container, but referencing on the EthernetContainer\_Pac defined within the ONF TR‑532v1.1 Microwave Information Model [ONF MW IM].

The following picture shall help understanding the layering of the Technology Specific Conditional Packages (\*\_Pacs), which are shared between ONF TR‑532v1.1 Microwave Information Model [ONF MW IM] and ONF TR‑541‑v1.0 Ethernet PHY Information Model. (It has to be noted that this picture is for explanatory purposes only. The associations between the \*\_Pacs are implemented in a more complex way, which is described in chapter 7.1 Attachment of \*\_Pacs.)

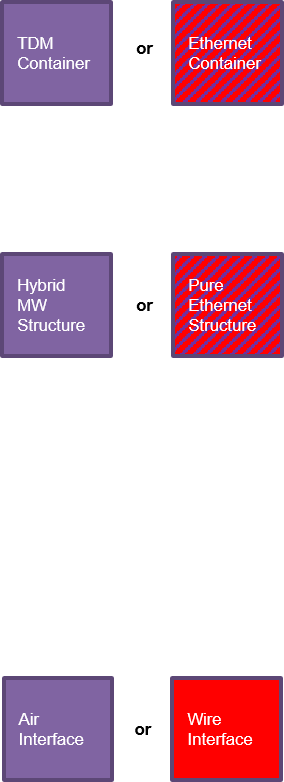
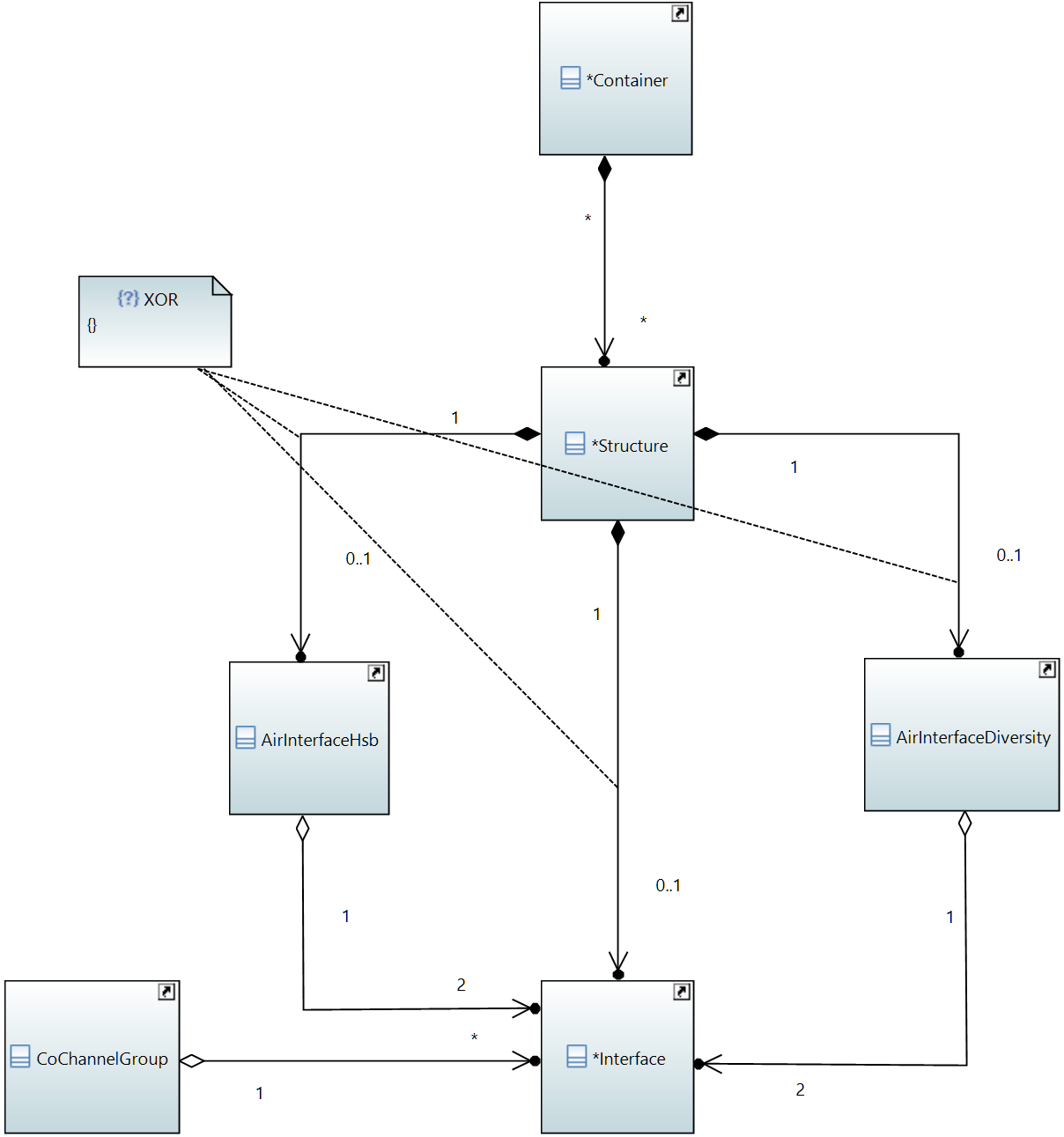


Figure 1: Model Overview

All Technology Specific Conditional Package (\*\_Pacs) are sub-structured into the following classes:

* Capabilities – The device informs about its features, characteristics and tuning ranges
* Configuration – The Controller configures the device and the device informs about its current configuration
* Status – The device informs about measurement values and its current operational status
* Problems – The device proactively informs about events
* Current Performance – The device informs about the current status of its performance counters
* Historical Performance – The device informs about the status of its performance counters at the end of a well-defined time period

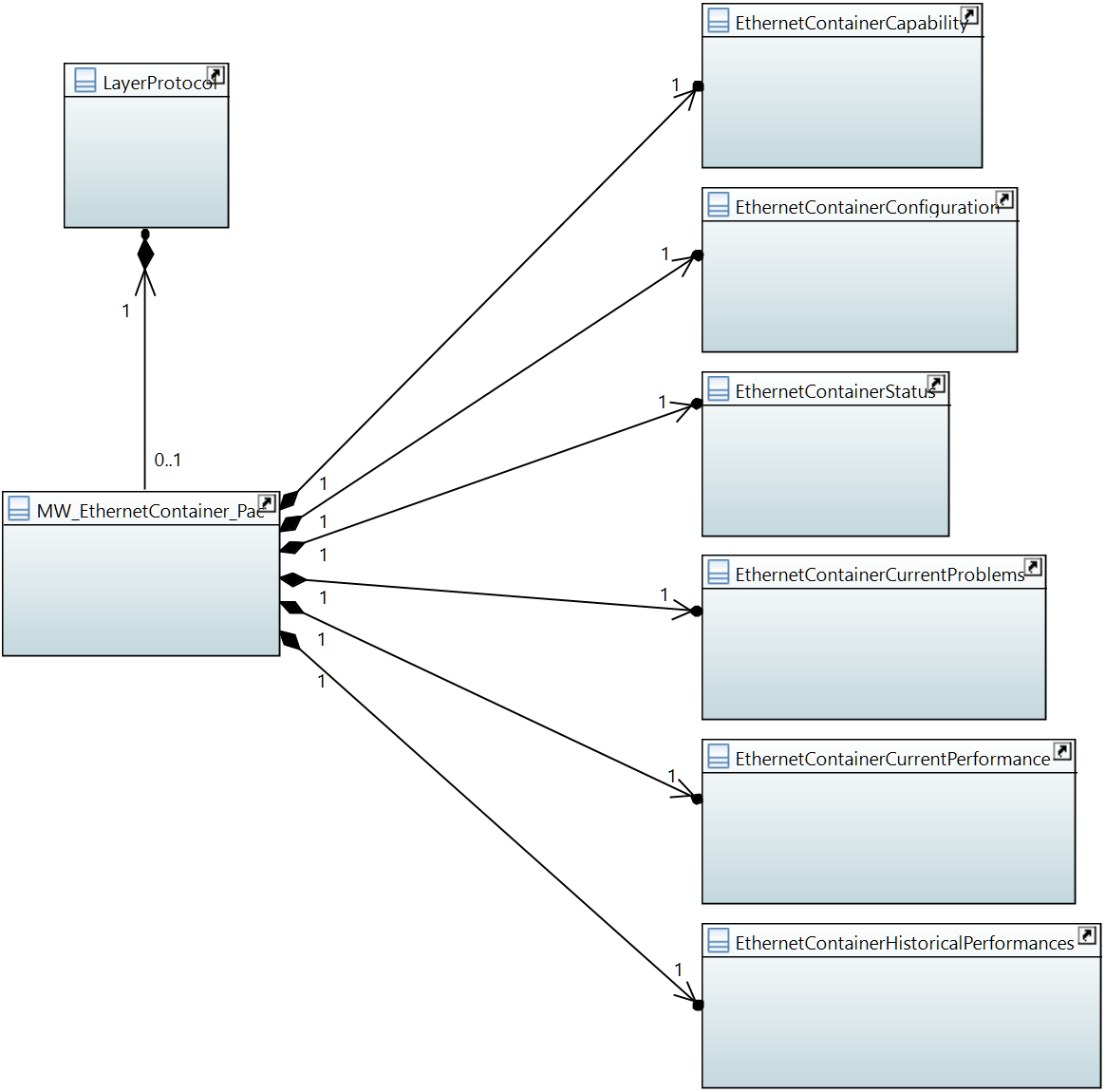


Figure 2: Example for the sub-structuring of \*\_Pacs

Each Technology Specific Conditional Package (\*\_Pac) is sufficiently described by one set of capability, configuration, status and current performance attributes, but several current problem and historical performance value sets might be required. Nevertheless, \*CurrentProblems and \*HistoricalPerformances classes are in a one to one association to the \*\_Pac. The necessary multiplicity is generated by lists inside these classes. This additional hierarchy is required by the interface implementation after conversion to YANG.

Equipment

Existing structures of the TR-512 ONF Core Information Model 1.2 [ONF Core IM] are used for documenting equipment information about the "transceiver" (e.g. SFP). Reference tables are provided in this document.

# Conventions

For seamless integration of components of this ONF TR‑541‑v1.0 Ethernet PHY Information Model and ONF TR‑532v1.1 Microwave Information Model [ONF MW IM] both models are following the same conventions.

As a consequence, this chapter about conventions is identical with chapter 5 Conventions in ONF TR‑532v1.1, except the exact content of the unit table in chapter 5.4.6 unit. Because ONF TR‑541‑v1.0 does not contain alternative Structure\_Pacs or alternative Container\_Pacs, chapter 5.4.7 Support Qualifier differs from ONF TR‑532v1.1, too.

## UML Modeling Conventions

This TR follows the conventions as described in the TR-512 ONF Core Information Model 1.2 [ONF Core IM] and the TR-514 ONF UML Modeling Guidelines [ONF UMGL].

## Default Values

The default values in the Ethernet PHY Information Model have been defined in accordance to the following basic principles:

* Every attribute (except keys, which have to be unique) shall have a default value.
* The default value shall be inside the value range of the data type of the attribute.
* Capability attributes:
  + The default value shall either indicate unavailability of the functionality (if applicable)
  + or be outside the range of reasonable values of the attribute.
* Configuration, status and performance attributes:
  + The default value shall either represent the configuration, status or performance measurement value right after starting the device (in case such a "neutral" value is applicable to the attribute)
  + or be outside the range of reasonable values of the attribute.

Lists of data types shall contain the minimum multiplicity number of elements. This also means that lists of data types with a minimum multiplicity of zero shall just be empty as a default.

## Comments

The comprised comments are meant to explain the attributes in such a way that further documentation is not required for understanding the attributes' meaning.

In rare cases (e.g. Problems) the modeling is open for amending device specific elements. This is done by Capability attributes of data type String, which are foreseen to contain an unspecified number of elements separated by comma. In such cases, the comment field is also used for defining a minimum set of elements, which must be supported. Of course, adding additional elements to these lists is not seen as a Proprietary Extensions according to chapter 3.

Double quotes have been avoided within the comments for supporting the conversion to YANG.

## ONF Stereotypes

### attributeValueChangeNotification

This stereotype defines whether a notification has to be raised, when the attribute changes its value.

It has been set on "true" for all attributes, which are comprised in Configuration classes or in data types used by attributes of the Configuration classes.

It has also been set on "true" for status attributes, which might be subject to automated changes, but do not represent gradually changing measurement values.

The attributeValueChangeNotification stereotype has been set on "false" for status attributes, which are exclusively following configuration activities. This is for avoiding double messaging.

### objectCreationNotification and objectDeletionNotification

These stereotypes define whether a notification has to be raised when an instance of a class has been created, respectively deleted.

It has been set on "true" for all \*\_Pac classes, which are potentially attached to the LayerProtocol class of the Core Information Model (see chapter 7.1 Attachment of \*\_Pacs for details).

### isInvariant

This stereotype defines whether the value of the attribute can be changed, or not, after it has been created.

It has been set on "true" (means: cannot be changed) for the following attributes:

* \*Ids, which are representing target addressed for referencing data types or classes, (except ContainerIDs, which are required for connecting logical traffic from outside the reach of the modeling with a Container)
* All attributes, which are comprised in Capability classes or in data types used by attributes of the Capability classes
* All attributes, which are comprised in data types that are attached to the CurrentProblems classes.
* All attributes, which are comprised in data types that are attached to the \*Performance classes.
* All attributes, which are comprised in Notifications

It has been set on "false" (means: can be changed) for the following attributes:

* All attributes, which are comprised in Configuration classes
* All attributes, which are comprised in Status classes or in data types used by attributes of the Status classes

### valueRange

This stereotype identifies the allowed values for the attribute.

It has not been used within the Ethernet PHY Information Model, because of a conflict with the policy defined for default values in chapter 5.2

### partOfObjectKey

This stereotype indicates whether the attribute is part of the object key or not. Value "0" (default) means the attribute is not part of the object key. Values > "0" indicate that the attribute is part of the object key and the value defines the order of the attribute in case the key is composed of more than one attribute.

The partOfObjectKey stereotype has been set on "1" for all \*Id attributes, which are of data type UniversalId, but not the \*IdRef attributes, which are referring on them.

It does also not apply on containerID attributes that represent a configurable string and are required for associating traffic flows to the transport resources (Container) provided by the Ethernet PHY Information Model.

### unit

This optional stereotype contains a textual definition of the unit associated with the attribute's value.

The following units have been used in the Ethernet PHY Information Model

| Unit | Meaning |
| --- | --- |
| % | Percentage |
| Bytes | Total number of Bytes |
| Bytes/min | Bytes per minute |
| Bytes/s | Bytes per second |
| Celsius | Degree Celsius |
| dB | Decibel |
| dBm | Decibel milliwatt |
| kbit/s | 1000 bit per second |
| kHz | 1000 Hertz |
| m | Meter |
| mJ | Millijoule |
| ms | Millisecond |
| mW | Milliwatt |
| octets | Total number of Octets |
| pm | Picometer |
| ps | Picosecond |
| s | Second |
| symbols | Number of symbols |

### Support Qualifier

The SupportQualifier has always been left on its default value "mandatory":

### bitLength

The bitLength stereotype has been used to define Integer primitives smaller than 64bit.

### Unsigned, Encoding and Counter

The unsigned, encoding and counter stereotypes have not been applied.

# Special elements

## PMD Definitions

PMDs are defined by name, speed and duplex type in a separated data type in the Capability section of the Ethernet PHY Information Model. The comment field of the PmdType::pmdName attribute prescribes a list of PMD names (according to [IEEE802.3]). During interface implementation, the supported PMDs have to be named according to this prescription. Only in case additional PMDs, which are not listed in the comment, would be defined, additional names could be applied. In case such additional PMD would be described in any future version of IEEE 802.3 Ethernet [IEEE802.3], the name stated there shall be applied.

## Shared \*\_Pacs

The PureEthernetStructure\_Pac of this ONF TR‑541‑v1.0 Ethernet PHY Information Model is equivalent to the PureEthernetStructure\_Pac of the ONF TR‑532v1.1 Microwave Information Model [ONF MW IM].

The EthernetContainer\_Pac of this ONF TR‑541‑v1.0 is equivalent to the EthernetContainer\_Pac of the ONF TR‑532v1.1.

## Problem Definitions, Associations and Inheritances

The problem definitions, associations and inheritances are identical to the definitions made ONF TR‑532v1.1 (see chapter 6.3 of [ONF MW IM]).

## Performance Values

The performance value definitions, associations and inheritances are identical to the definitions made ONF TR‑532v1.1 (see chapter 6.4 of [ONF MW IM]).

## Maintenance Attributes

Some features are required for maintenance. Some of these maintenance features potentially break the connection to the Controller. ONF TR‑541 facilitates configuring a timer. After this maintenance timer counted to zero, the values of all attributes related to maintenance features have to be set to their default values again. The affected attributes are marked by the phrase "Maintenance Feature." at the beginning of their comments.

| Maintenance Attributes |
| --- |
| \_loopBackKindOn |
| isolationIsOn |

## Transceiver Information

Tracking equipment information about transceivers (e.g. SFPs) is of particular interest to operators, because transceivers can easily be pulled and plugged somewhere else. Prescriptions, which are made in IEEE 802.3 Ethernet [IEEE802.3], cover the necessary equipment information. The following definitions shall assure a common understanding about where to put/find the information about the actual hardware in the equipment part of the Core Information Model 1.2 [ONF Core IM].

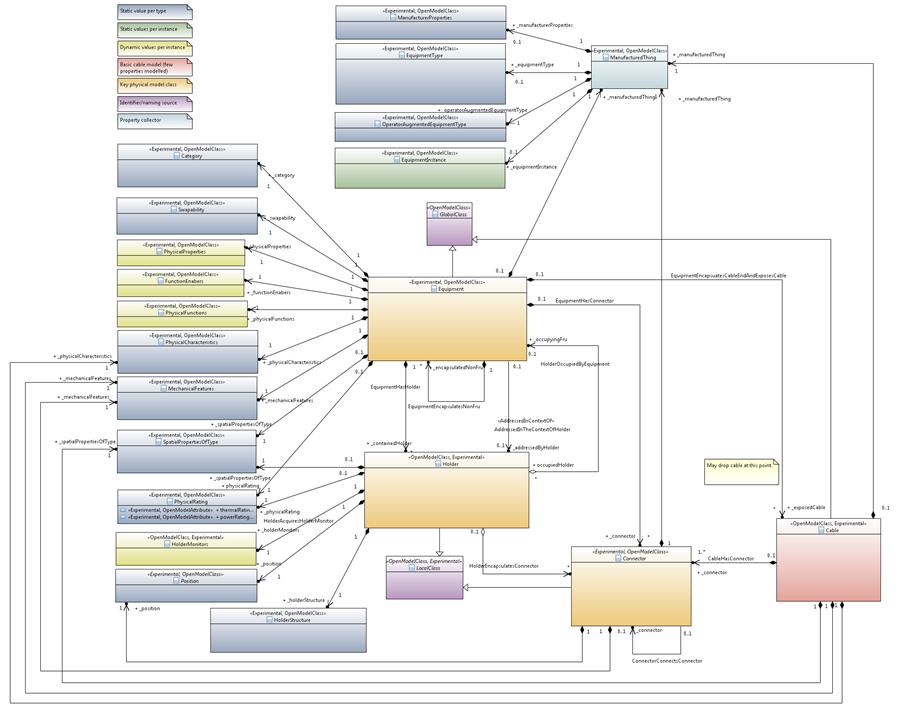


Figure 3: Equipment part of the Core Information Model

Information about the manufacturer of the transceiver shall be located in the ManufacturerProperties Class.

| Attribute Name in Core IM | To be filled with following information from 802.3 |
| --- | --- |
| manufacturerIdentifier | Describes the IEEE Company identifier of the vendor of the transceiver (1st part of 802.3 ResourceTypeID) according to 802.3 22.2.4.3.1 PHY Identifier and also referenced in 45.2.1.13 PMA/PMD package identifier. |
| manufacturerName | No entry |

Table 1: ManufacturerProperties Class

The type of transceiver shall be described in the EquipmentType Class.

| Attribute Name in Core IM | To be filled with following information from 802.3 |
| --- | --- |
| description | Text describing the type of Equipment.  The following information shall be provided:   * Highest PMD * MiiKindType * MdiKindType   All values shall be separated by "/".  Example:  1000BASE-T/SOLDERED\_CONNECTOR/RJ45 |
| modelIdentifier | No entry |
| partTypeIdentifier | Uniquely identifies the transceiver in the vendor's product lists according to 802.3 22.2.4.3.1 PHY Identifier and also referenced in 45.2.1.13 PMA/PMD package identifier as six bit model number. |
| typeName | No entry |
| version | Identifies the revision number of the transceiver (3rd part of 802.3 ResourceTypeID) according to 802.3 22.2.4.3.1 PHY Identifier and also referenced in 45.2.1.13 PMA/PMD package identifier as four-bit revision number. |

Table 2: EquipmentType Class

The concrete item shall be identified with the values stated in the EquipmentInstance Class.

| Attribute Name in Core IM | To be filled with |
| --- | --- |
| manufactureDate | Vendor's date code for the transceiver. |
| serialNumber | Vendor's serial number for the transceiver (0 = not applicable). |
| assetInstanceIdentifier | This attribute represents the asset identifier of this instance from the manufacturer's perspective. |

Table 3: EquipmentInstance Class

The current temperature of the transceiver shall be read from the PhysicalProperties Class.

| Attribute Name in Core IM | To be filled with |
| --- | --- |
| temperature | Current temperature (in degree Celsius) inside the transceiver. |

Table 4: PhysicalProperties Class

## Expected Transceiver

The Capability class of the WireInteface\_Pac holds information, which is specific to the type of transceiver. Because the attributes of the Capability class are invariant, the wireInteface\_Pac cannot be instantiated before an expected equipment has already been defined.

On the other hand, the actually operative transceiver might break. In this situation, the instance of the wireInterface\_Pac shall continue existing. This is why the wireInterface\_Pac is associated with the expected equipment, but not with the actual equipment.

If the expected equipment would also contain attributes like manufactureDate or serialNumber, there would be a mismatch between expected and actual equipment after the replacement.

This means that the expected equipment has to hold sufficient information for unambiguously determining the Capability information, respectively assuring compatibility of all attributes of the WireInteface\_Pac. On the other hand, as much information as possible has to be masked to allow replacing the hardware as flexibly as possible.

As a compromise the expected equipment shall be exclusively defined by the values of the attributes of the ManufacturerProperties class and the EquipmentType class as they are described above.

# Attachment to the Core Information Model

The Core Information Model is the basis of the Ethernet PHY Information Model. However not all object classes of the Core Information Model are necessary for a control interface between Devices and SDN Controller. For example the root element of the Ethernet PHY Information Model is the NetworkElement object class. The SdnController object class must not be implemented in network elements, because a network element cannot offer any control for SDN Controllers.

The following classes of the Core Information Model are not used by the Ethernet PHY Information Model:

* SdnController
* NetworkControlDomain
* Link
* LinkPort
* FcRoute

Figure 4 highlights the classes of the Core Information Model, which are part of the Ethernet PHY Information Model.

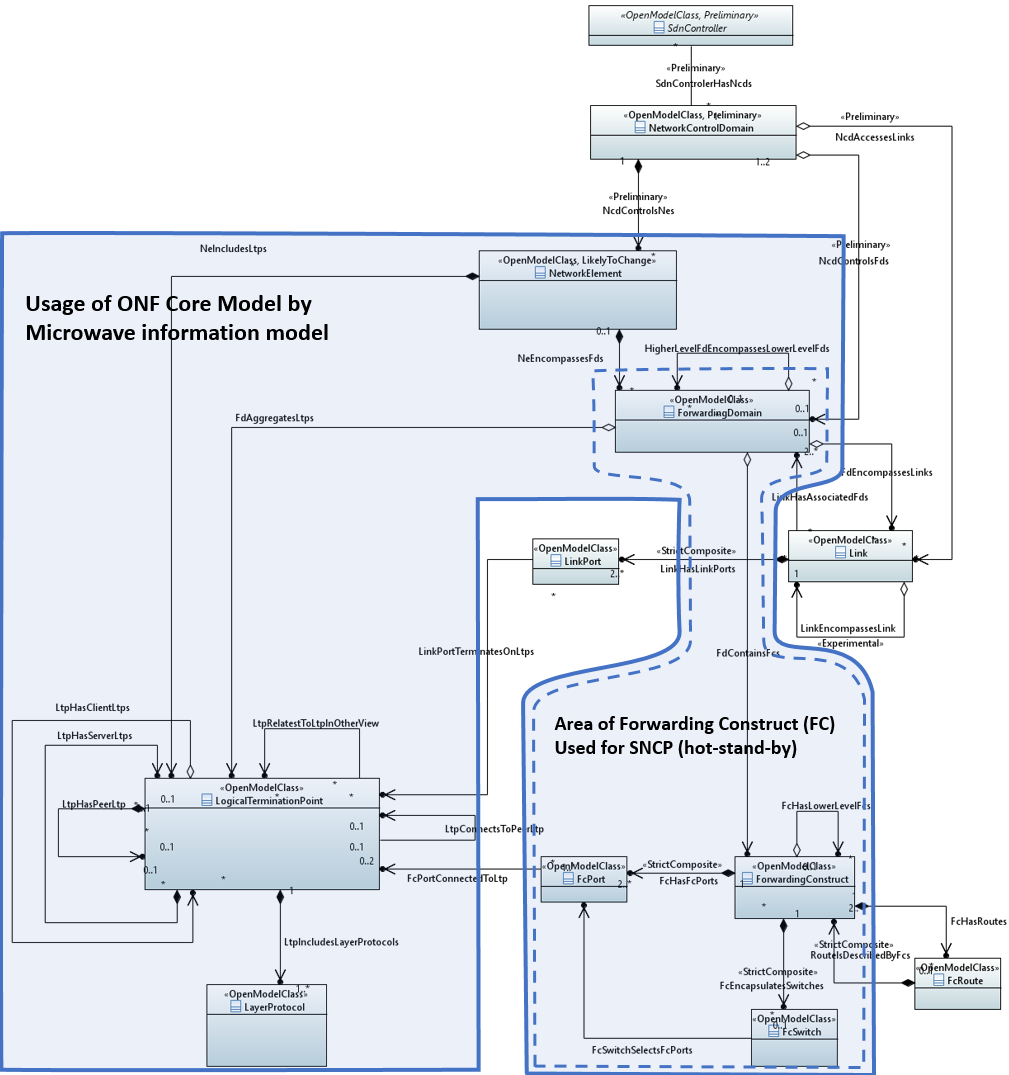


Figure 4: Usage of ONF Core Information Model 1.2

The LogicalTerminationPoints (LTP) is the most important class. Its associations allow complex hierarchies of LTPs. A LayerProtocol class provides content to the generic LTP class and describes its function.

The following layerProtocolName values have to be applied:

| layerProtocolName | Type of \*\_Pac |
| --- | --- |
| ETC | EthernetContainer |
| MWS | PureEthernetStructure |
| PHYPS | WireInterface |

## Attachment of \*\_Pacs

Technology specific extensions of the Core Information Model are implemented by extending the LayerProtocol class with conditional packages. These conditional packages are called \*\_Pacs.

The Ethernet PHY Information Model defines three technology specific conditional packages, pointing to the LayerProtocol:

* The WireInterface\_Pac defines an LTP as Ethernet PHY section trail termination point (PHYPS-TTP).
* The PureEthernetStructure\_Pac defines an LTP as Ethernet PHY and microwave section trail termination point (MWS-TTP) for pure Ethernet transport systems.
* The EthernetContainter\_Pac defines an LTP as a Ethernet PHY and microwave client connection termination point (MW-Client-CTP) for an Ethernet client signal.

Figure 5 shows the associations between the LayerProtocol class of the Core Information Model and the three microwave specific conditional packages. Please note that for simplicity reasons the ForwardingConstruct class is not displayed.

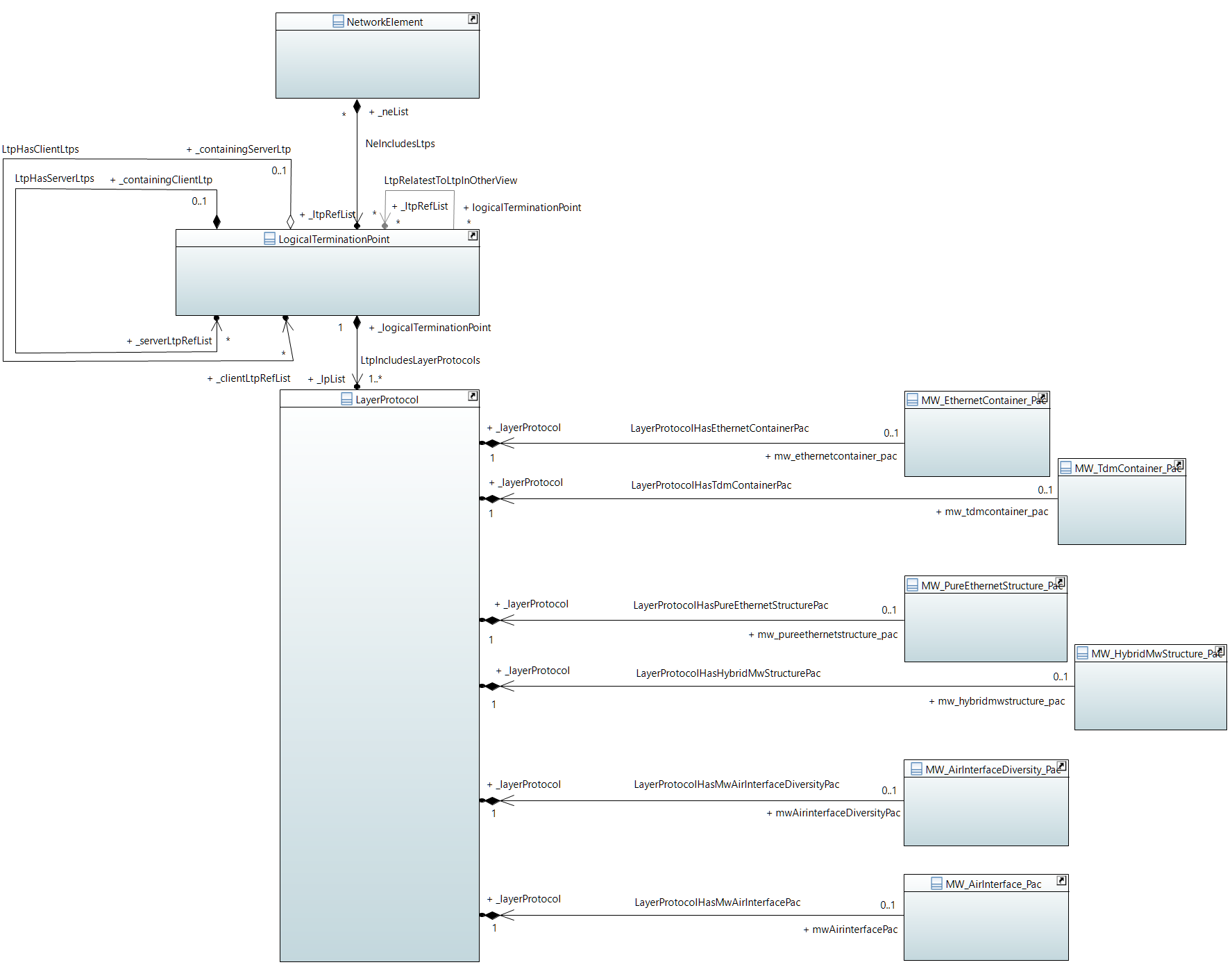


Figure 5: Associations between LayerProtocol and \*\_Pac

# Non-Backward Compatibility of the next Version

ONF TR‑541‑v1.0 comprises not just the technology specific \*\_Pacs for managing the Ethernet PHY interface, but also elements of the ONF Core Information Model. Because the next version of ONF TR‑541 will require to comprise an updated version of the Core Information Model, which is not backward compatible, also the next version of ONF TR‑541 will not be backward compatible.

Please, see further information about future work, which will also impact backward compatibility, in chapter 9.

# Future Work

It is seen as a problem that ONF TR‑532 and ONF TR‑541 are not disjunctive in regards to pureEthernetStructure\_Pac and EthernetContainer\_Pac. It is considered to decompose both information models into \*\_Pacs, make these \*\_Pacs available in separate repositories, and to just reference these \*\_Pacs inside the respective TR documents. Apart from avoiding redundancy, this decomposition would also allow updating the \*\_Pacs independently from each other. As a consequence, each \*\_Pac would be delivered as a separate YANG file.

Current version of ONF TR‑541 is limited on technology specific conditional packages to the LayerProtocol class of the Core Information Model. This results in a comprehensive model for managing devices. From the controller's and the applications' points of view, these devices have to be connected to describe a network topology. The necessary classes for describing such topologies are also comprised in the Core Information Model, but they require technology specific conditional packages in the same way the LayerProtocol class does. It is considered to define Ethernet PHY specific conditional packages also for the ForwardingConstruct and to reference them in the next version of ONF TR‑541.

The next version of ONF TR‑541 shall also take benefit from the more advanced modeling of equipment, connectors and topology in later versions of the Core Information Model. E.g. several attributes, which are detailing characteristics of the SFP, will be replaced to equipment related classes then.

# UML Model Files

The Papyrus export, which is holding the UML modeling files of the ONF TR‑541‑v1.0 Ethernet PHY Information Model can be found on

<https://github.com/openBackhaul/wireInterface/tree/TR541v1>

The available .zip file comprises the

* WireInterface\_Pac

and all related datatypes, notifications and imported artefacts.

Please note that YANG Model files, Data Dictionary (GenDoc export) and Interface Simulator are just implementations of the UML modeling files.

In case of any divergence, the UML modeling files are always the relevant data base.

# YANG Model Files

The YANG files generated from the UML model referenced in chapter 10 can be found on

[not](https://github.com/openBackhaul/airInterface/tree/TR532v1_1) yet generated

# Interface Simulator

An Interface Simulator is for supporting application development without own hardware. The Interface Simulator, which is implementing the UML model referenced in chapter 10 can be found on

[not](https://github.com/openBackhaul/airInterface/tree/TR532v1_1) yet generated

# Data Dictionary

Please be aware that the following lists are showing not all the attribute's characteristics and stereotypes, e.g. default value and unit are missing.

## WireInterface\_Pac

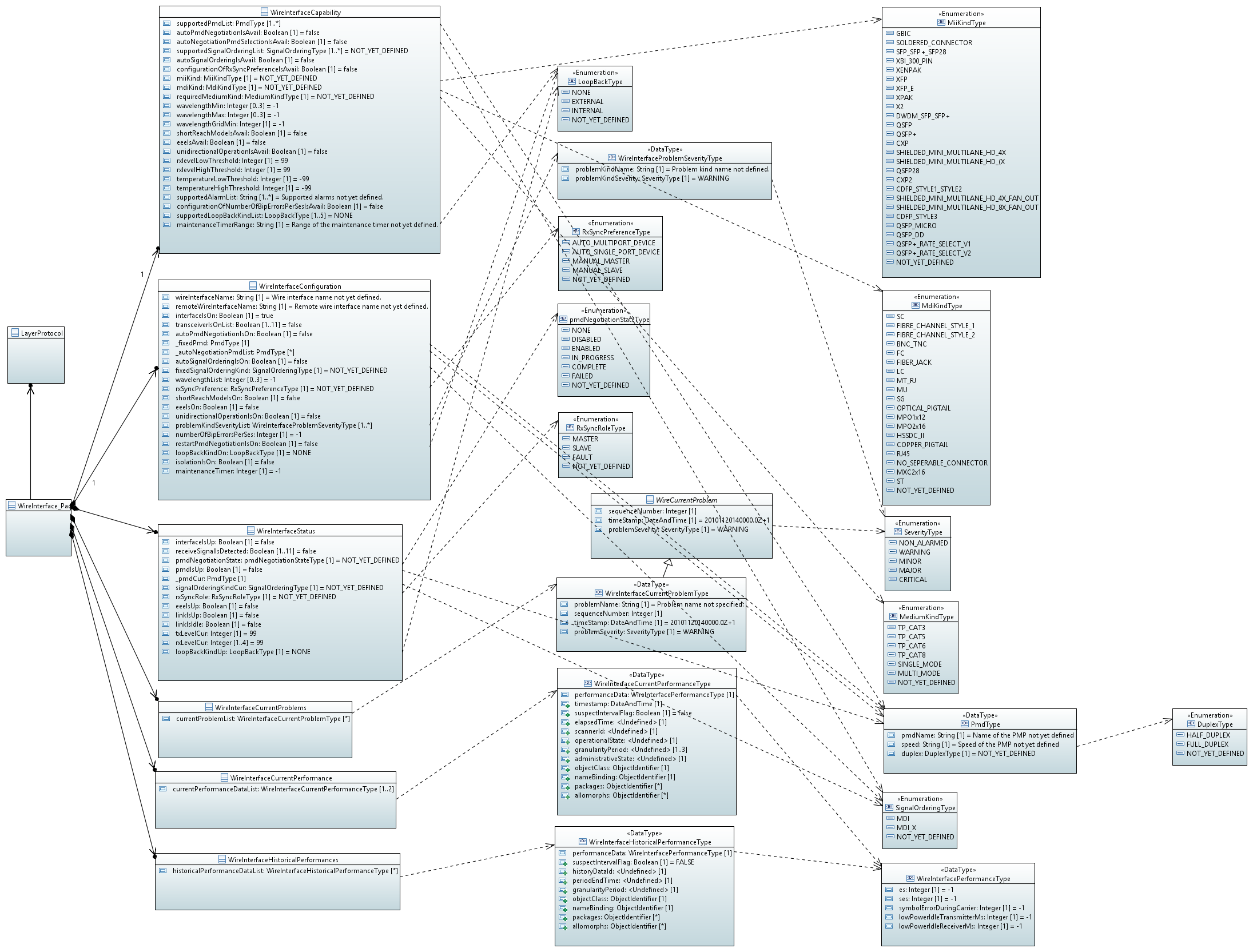


Figure 6: WireInterface\_Pac

### WireInterface\_Pac

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterface\_Pac

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: YES
* objectDeletionNotification: YES
* support: MANDATORY

Table 5: Attributes for WireInterface\_Pac

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| \_layerprotocol | LayerProtocol  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 1 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |
| \_wirebasedinterfacecapability | WireInterfaceCapability  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |
| \_wirebasedinterfaceconfiguration | WireInterfaceConfiguration  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |
| \_wirebasedinterfacestatus | WireInterfaceStatus  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |
| \_wirebasedinterfacecurrentproblems | WireInterfaceCurrentProblems  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |
| \_wirebasedinterfacecurrentperformance | WireInterfaceCurrentPerformance  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |
| \_wirebasedinterfacehistoricalperformances | WireInterfaceHistoricalPerformances  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | See referenced class |

### WireInterfaceCapability

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterfaceCapability

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 10: Attributes for WireInterfaceCapability

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| supportedPmdList | PmdType  ./. | 1..\* | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | List of Physical Medium Dependent (PMD) that can be operated |
| autoPmdNegotiationIsAvail | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = device is supporting auto-negotiation |
| autoNegotiationPmdSelectionIsAvail | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = device is supporting restricting auto-negotiation on a pre-defined list of PMDs |
| supportedSignalOrderingList | SignalOrderingType  NOT\_YET\_DEFINED | 1..\* | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Describes the different (e.g. MDI, MDI-X) ways of ordering the signals on the physical medium |
| autoSignalOrderingIsAvail | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = there is a mechanism for automatically crossing over tx and rx implemented |
| configurationOfRxSyncPreferenceIsAvail | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = Configuration of the behavior during the synchronization of transmitter and receiver is available. This attribute has nothing to do with clock signals. |
| miiKind | MiiKindType  NOT\_YET\_DEFINED | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Kind of Medium Independent Interface (MII) provided by this Medium Attachment Unit (MAU) (e.g. SFP, moldered port) |
| mdiKind | MdiKindType  NOT\_YET\_DEFINED | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Kind of Medium Dependent Interface (MDI) provided by this Medium Attachment Unit (MAU) |
| requiredMediumKind | MediumKindType  NOT\_YET\_DEFINED | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Kind of medium required for operating this Medium Attachment Unit (MAU), more like an information field |
| wavelengthMin | Integer  -1 | 0..3 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: pm * support: MANDATORY | Source: SFF-8690. Minimum laser wavelength in pico meter, -1 = not applicable, 0 = not known, if (wavelengthMin==wavelengthMax): wavelength cannot be configured; multiplicity=0..3 for 10GBASE-LX4 according to 802.3 53.5. Value to be read from the EPROM of the SFP. |
| wavelengthMax | Integer  -1 | 0..3 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: pm * support: MANDATORY | Source: SFF-8690. Maximum laser wavelength in pico meter, -1 = not applicable, 0 = not known, if (wavelengthMax==wavelengthMin): wavelength cannot be configured; multiplicity=0..3 for 10GBASE-LX4 according to 802.3 53.5. Value to be read from the EPROM of the SFP. |
| wavelengthGridMin | Integer  -1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: pm * support: MANDATORY | Source: SFF-8690. Minimum grid spacing supported by the transceiver, -1 = not applicable, 0 = not known |
| shortReachModeIsAvail | Boolean  false | 1 | R | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = Indicates that Short Reach Mode for 10GBASE-T according to 802.3 45.2.1.64 is available |
| eeeIsAvail | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = Indicates that Energy-Efficient Ethernet (EEE) is available at the device. |
| unidirectionalOperationIsAvail | Boolean  false | 1 | R | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3. 1 = Medium Attachment Unit (MAU) able to transmit from Media Independent Interface (MII) regardless of whether the MAU has determined that a valid link has been established, 0 = MAU able to transmit from MII only when the MAU has determined that a valid link has been established |
| rxlevelLowThreshold | Integer  99 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_8\_BIT * unit: dBm * support: MANDATORY | Threshold for alarming low RX levels. Value pre-defined by SFP manufacturer (SFF- 8472) |
| rxlevelHighThreshold | Integer  99 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_8\_BIT * unit: dBm * support: MANDATORY | Threshold for alarming high RX levels. Value pre-defined by SFP manufacturer (SFF- 8472) |
| temperatureLowThreshold | Integer  -99 | 1 | R | LikelyToChange  OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_8\_BIT * unit: Celsius * support: MANDATORY | Threshold for alarming low temperature values. Value pre-defined by SFP manufacturer (SFF- 8472)  Might move to somewhere in the Physical Segment of the Core IM |
| temperatureHighThreshold | Integer  -99 | 1 | R | LikelyToChange  OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_8\_BIT * unit: Celsius * support: MANDATORY | Threshold for alarming high temperature values. Value pre-defined by SFP manufacturer (SFF- 8472)  Might move to somewhere in the Physical Segment of the Core IM |
| supportedAlarmList | String  Supported alarms not yet defined. | 1..\* | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Available alarms to be listed. Mandatory: 'rxLos'. Optional: 'txFault', 'tempHigh', 'tempLow', 'rxLevelHigh', 'rxLevelLow', 'vccHigh', 'vccLow', 'txBiasHigh', 'txBiasLow', 'txPowerHigh', 'txPowerLow', 'laserTempHigh', 'laserTempLow', 'tecCurrentHigh', 'tecCurrentLow'. Further alarms might be added by the device. |
| configurationOfNumberOfBipErrorsPerSesIsAvail | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = SET operation on the Line SES threshold defined by aLineSESThreshold in 30.8.1.1.11 of 802.3-2015 is available. |
| supportedLoopBackKindList | LoopBackType  NONE | 1..5 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3 45.2.1.12.1 PMA remote loopback ability. List of supported kinds of looping back of header information to the remote site. |
| maintenanceTimerRange | String  Range of the maintenance timer not yet defined. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: s * support: MANDATORY | Available time periods for maintenance configurations (e.g. the loop back) to be described. Concrete values shall be separated by commas (e.g. '10, 60, 360'). Ranges shall be expressed as two values separated by a minus (e.g. '10-360'). |

### WireInterfaceConfiguration

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterfaceConfiguration

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 6: Attributes for WireInterfaceConfiguration

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| wireInterfaceName | String  Wire interface name not yet defined. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Text field for the wire interface being named by the operator. Ideally used for entering unique numbers or names for unambiguously identifying the connection within the network |
| remoteWireInterfaceName | String  Remote wire interface name not yet defined. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Text field for defining the wire interface this one is connected with. Ideally used for entering unique numbers or names for unambiguously identifying the connection within the network |
| interfaceIsOn | Boolean  true | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3 according 30.3.2.2.1 acPhyAdminControl. 1 = Activation of the interface (it gets powered and can be managed even if the transceiver is not yet transmitting or receiving). In case there is no Medium Attachment Unit (MAU) (e.g. no SFP in the cage) SETting (interfaceIsOn=1) must be ignored and GETing must return (interfaceIsOn=0) |
| transceiverIsOnList | Boolean  false | 1..11 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3 45.2.1.8 PMD transmit disable register (Register 1.9). 1 = Activation of the transmitter and receiver (e.g. laser) of the PHY; transceiverIsOnList[0]:total interface; transceiverIsOnList[1..10] different lanes of a multilane Medium Attachment Unit (MAU) |
| autoPmdNegotiationIsOn | Boolean  false | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Only relevant if (autoPmdNegotiationIsAvail == 1). 1 = Auto-negotiation is switched on |
| \_fixedPmd | PmdType  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | If (autoPmdNegotiationIsOn==0) : Configuration of the concrete kind of Physical Medium Dependent (PMD). If (autoNegotiationIsOn=1) value of this field becomes irrelevant |
| \_autoNegotiationPmdList | PmdType  ./. | 0..\* | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | If (autoPmdNegotiationIsOn==1) AND (autoNegotiationPmdSelectionIsAvail==1), this list defines the selection of PMDs the automated negotiation process is allowed to choose from. If no entry, the automated negotiation process is allowed to choose from all available PMDs. |
| autoSignalOrderingIsOn | Boolean  false | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Only relevant if (autoSignalOrderingIsAvail == 1). 1 = e.g. auto-MDI-X is switched on |
| fixedSignalOrderingKind | SignalOrderingType  NOT\_YET\_DEFINED | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | If (autoSignalOrderingIsOn==0) configuration of the concrete kind of signal ordering on the media (e.g. MDI, or MDI-X). If (autoSignalOrderingIsOn=1) value of this field becomes irrelevant |
| wavelengthList | Integer  -1 | 0..3 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: pm * support: MANDATORY | Source: SFF-8690. Wavelength of the signal of laser in pico meter; multiplicity=0..3 for 10GBASE-LX4 according to 802.3 53.5 |
| rxSyncPreference | RxSyncPreferenceType  NOT\_YET\_DEFINED | 1 | RW | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3. Only relevant if (configurationOfRxSyncPreferenceIsAvail == 1). Configuration of the behavior during the negotiation of the wire interface (master), which is sending a continuous stream of symbols for the remote site (slave) to synchronize its receiver on it. This attribute is for synchronizing transmitter and receiver and has nothing to do with clock signals. |
| shortReachModeIsOn | Boolean  false | 1 | RW | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Only relevant if (shortReachModeIsAvail == 1). Activation of the Short Reach Mode for 10GBASE-T according to 802.3 45.2.1.64 |
| eeeIsOn | Boolean  false | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Only relevant if (eeeIsAvail == 1). 1 = Energy Efficient Ethernet is activated |
| unidirectionalOperationIsOn | Boolean  false | 1 | RW | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3. Only relevant if (unidirectionalOperationIsAvail == 1). 1 = Enable transmit from media independent interface regardless of whether the PHY has determined that a valid link has been established, 0 = Enable transmit from media independent interface only when the PHY has determined that a valid link has been established |
| problemKindSeverityList | WireInterfaceProblemSeverityType  ./. | 1..\* | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Severity of the problem to be configured. |
| numberOfBipErrorsPerSes | Integer  -1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: LENGTH\_16\_BIT * unit: no unit defined * support: MANDATORY | Only relevant if (configurationOfNumberOfBipErrorsPerSesIsAvail == 1). SET operation on the Line SES threshold defined by aLineSESThreshold in 30.8.1.1.11 of 802.3-2015. This attribute configures the minimum number of BIP errors that have to occur in one second to count this second as a SES instead of an ES. According to 802.3, a value of 9835 should be the hardware default value, which should also apply after restarting the interface. |
| restartPmdNegotiationIsOn | Boolean  false | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3. Restarts the auto negotiation process |
| loopBackKindOn | LoopBackType  NONE | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Maintenance Feature. Source:802.3 according 22.2.4.1.2 Loopback. The currently configured type of looping back of the wire interface header shall be expressed here. The received header is returned to the remote site. |
| isolationIsOn | Boolean  false | 1 | RW | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Maintenance Feature. Source: 802.3. 1 = Activation of the separation of the PHY from higher network layers. Maintenance feature. |
| maintenanceTimer | Integer  -1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: s * support: MANDATORY | Time of existence of any maintenance configuration (e.g. the loop back). Valid values are defined in WireInterface::WireInterfaceCapability::maintenanceTimerRange. In case the hardware would implement just a single timer for maintenance at all kinds and instances of interfaces, this attribute should affect this single timer. |

### WireInterfaceStatus

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterfaceStatus

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 11: Attributes for WireInterfaceStatus

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| interfaceIsUp | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3 according 30.3.2.1.7 aPhyAdminState. 1 = A Physical layer entity (PHY) exists (including Medium Attachment Unit (e.g. SFP) ) and it is powered and can be managed |
| receiveSignalIsDetected | Boolean  false | 1..11 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3 45.2.1.9 PMD receive signal detect. 1 = Receiver (e.g. laser) detects signal; receiveSignalIsDetected[0]:total interface; receiveSignalIsDetected[1..10] different lanes of a multilane Medium Attachment Unit (MAU) |
| pmdNegotiationState | pmdNegotiationStateType  NOT\_YET\_DEFINED | 1 | R | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3. Status of the Physical Medium Dependent (PMD) negotiation process (auto-neg) |
| pmdIsUp | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: Inverse of 802.3 45.2.1.2.3 Fault (1.1.7). If (interfaceIsUp==1) BUT (pmdIsUp==0) : there is a fault in either transmit or receive path |
| \_pmdCur | PmdType  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Indicates the kind of Physical Medium Dependent (PMD) currently operated at this interface |
| signalOrderingKindCur | SignalOrderingType  NOT\_YET\_DEFINED | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Reference on a SignalOrderingType for expressing the currently active way of ordering the signals on the physical medium. |
| rxSyncRole | RxSyncRoleType  NOT\_YET\_DEFINED | 1 | R | Experimental  OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Indicates the result of the negotiation of the wire interface (master), which is sending a continuous stream of symbols for the remote site (slave) synchronizing its receiver on it |
| eeeIsUp | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | 1 = Energy Efficient Ethernet is supported at both ends of the link and it is activated |
| linkIsUp | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | If (transceiverIsUp==1) AND (linkIsUp==1) : communication is established to the remote site |
| linkIsIdle | Boolean  false | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | If (linkIsUp==1) AND (eeeIsAvail==1) AND (eeeIsOn==1) AND (linkIsIdle==1) : link is currently in idle mode. If Energy Efficient Ethernet is not supported or switched off, this attribute must be 0. |
| txLevelCur | Integer  99 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: LENGTH\_8\_BIT * unit: dBm * support: MANDATORY | Current transmit power |
| rxLevelCur | Integer  99 | 1..4 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: LENGTH\_8\_BIT * unit: dBm * support: MANDATORY | Current receive power; Also used for receive signal power measured at the Medium Dependent Interface (MDI) of 10GBASE-T during training as described in 802.3 55.4.3.1 |
| loopBackKindUp | LoopBackType  NONE | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | The currently active (not just configured) type of looping back of the wire interface header shall be expressed here. The received header is returned to the remote site. |

### WireInterfaceCurrentProblems

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterfaceCurrentProblems

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 7: Attributes for WireInterfaceCurrentProblems

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| currentProblemList | WireInterfaceCurrentProblemType  ./. | 0..\* | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |

### WireInterfaceCurrentPerformance

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterfaceCurrentPerformance

Aggregated performance information of the air interface at a particular moment.

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 8: Attributes for WireInterfaceCurrentPerformance

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| currentPerformanceDataList | WireInterfaceCurrentPerformanceType  ./. | 1..2 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | At least values of the counters, which are reset every 15 minutes, are to be provided. If available, the current values of the counters, which are reset every 24 hour, can be provided, too. |

### WireInterfaceHistoricalPerformances

Qualified Name: WireModel::ObjectClasses::WireInterface::WireInterfaceHistoricalPerformances

Aggregated performance information of the air interface for a pre-defined measurement interval.

Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 9: Attributes for WireInterfaceHistoricalPerformances

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| historicalPerformanceDataList | WireInterfaceHistoricalPerformanceType  ./. | 0..\* | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |

## PureEthernetStructure\_Pac

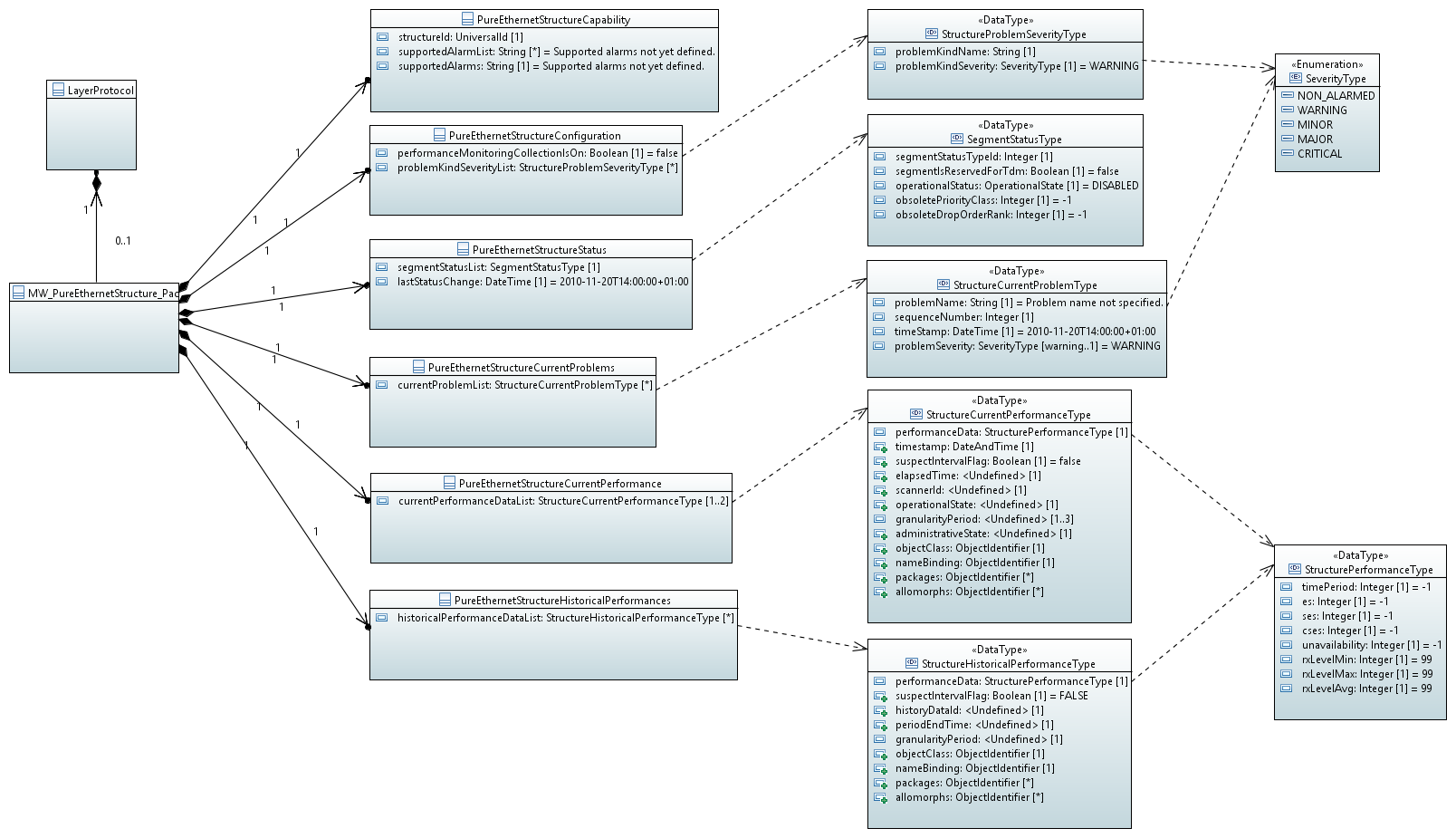


Figure 7: Exemplary PureEthernetStructure\_Pac

Please, see chapter 15.2.1 in ONF TR‑532v1.1 Microwave Information Model [ONF MW IM] for the exact definition of the PureEthernetStructure\_Pac.

## EthernetContainer\_Pac

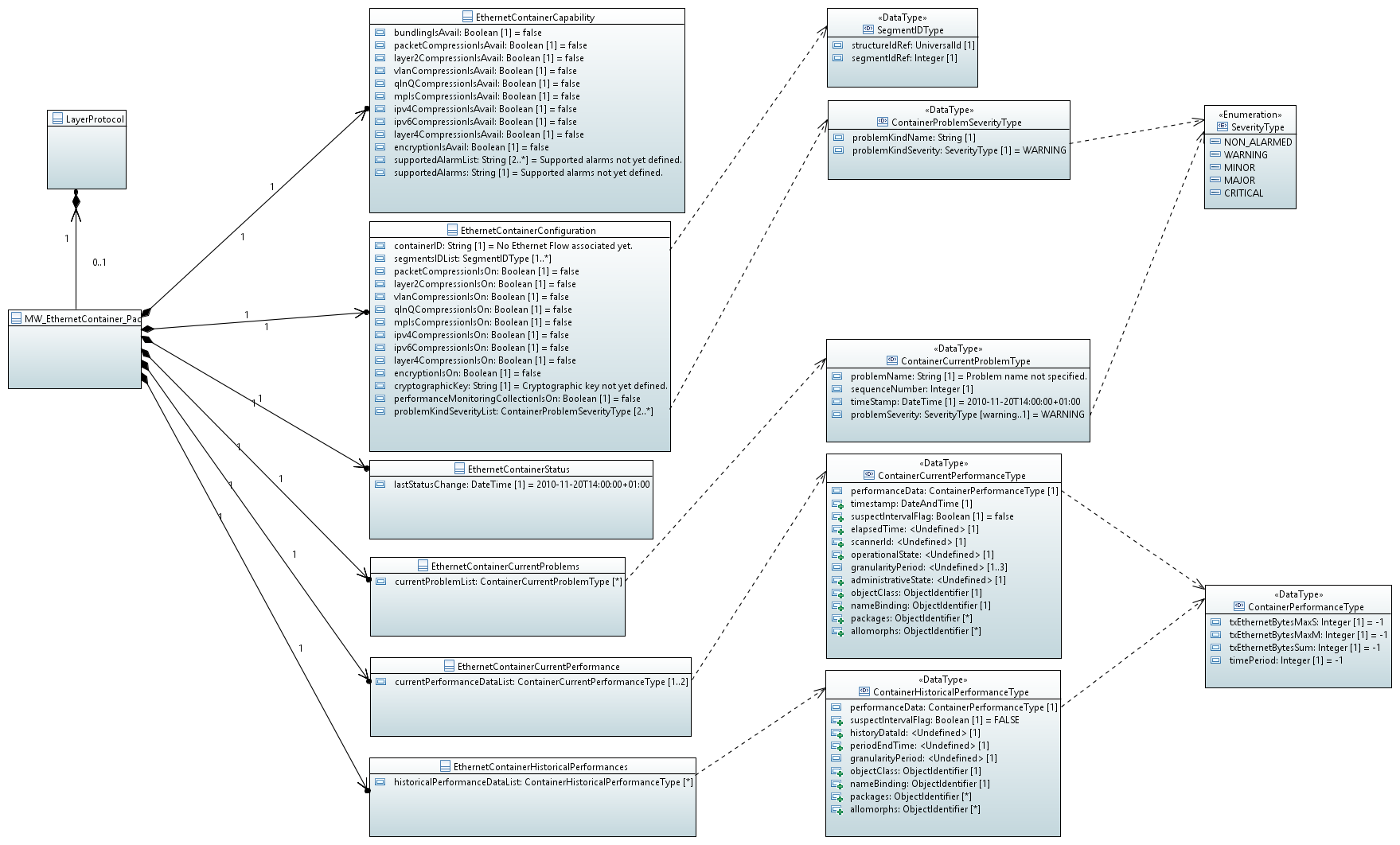


Figure 8: Exemplary EthernetContainer\_Pac

Please, see chapter 15.3.1 in ONF TR‑532v1.1 Microwave Information Model [ONF MW IM] for the exact definition of the EthernetContainer\_Pac.

## Data Types

### PmdType

Qualified Name: WireModel::TypeDefinitions::PmdType

Table 12: Attributes for PmdType

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| pmdName | String  Name of the PMP not yet defined | 1 | R | OpenModelAttribute   * partOfObjectKey: 1 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | To be chosen from the following list of PMD names according to IEEE 802.3 '10BASE5', 'FOIRL', '10BASE2', '10BROAD36', '10BASE-T', '10BASE-FP', '10BASE-FB', '10BASE-FL', '100BASE-T4', '100BASE-TX', '100BASE-BX10D', '100BASE-BX10U', '100BASE-FX', '100BASE-LX10', '100BASE-T2', '1000BASE-X', '1000BASE-BX10D', '1000BASE-BX10U', '1000BASE-LX', '1000BASE-LX10', '1000BASE-SX', '1000BASE-CX', '1000BASE-KX', '1000BASE-T', '10GBASE-X', '10GBASE-LX4', '10GBASE-KX4', '10GBASE-R', '10GBASE-ER', '10GBASE-LR', '10GBASE-SR', '10GBASE-LRM', '10GBASE-KR', '10GBASE-T', '10GBASE-PR-D1', '10GBASE-PR-D2', '10GBASE-PR-D3', '10GBASE-PR-D4', '10GBASE-PR-U1', '10GBASE-PR-U2', '10GBASE-PR-U3', '10GBASE-PR-U4', '40GBASE-R', '40GBASE-KR4', '40GBASE-CR4', '40GBASE-SR4', '4x10GBASE-SR', '40GBASE-LR4', '40GBASE-ER4', '40GBASE-FR', '100GBASE-R', '100GBASE-CR4', '100GBASE-KR4', '100GBASE-KP4', '100GBASE-CR10', '100GBASE-SR4', '100GBASE-SR10', '2x40GBASE-SR', '10x10GBASE-SR', '12x10GBASE-SR', '100GBASE-LR4', '100GBASE-ER4' and 'NOT\_YET\_DEFINED' |
| speed | String  Speed of the PMP not yet defined | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: Mbit/s * support: MANDATORY | Line speed of the PMD. Value to be chosen from '2Mbit/s', '10Mbit/s', '100Mbit/s', '1000Mbit/s', '10Gbit/s', '40Gbit/s' or '100Gbit/s' |
| duplex | DuplexType  NOT\_YET\_DEFINED | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | To be expressed, whether the PMD is full duplex or just half duplex |

### WireInterfaceProblemSeverityType

Qualified Name: WireModel::TypeDefinitions::WireInterfaceProblemSeverityType

Table 17: Attributes for WireInterfaceProblemSeverityType

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| problemKindName | String  Problem kind name not defined. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Name of the alarm according to WireInterface::WireInterfaceCapability::supportedAlarms |
| problemKindSeverity | SeverityType  WARNING | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: YES * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Severity of this type of alarm. |

### WireInterfaceCurrentProblemType

Qualified Name: WireModel::TypeDefinitions::WireInterfaceCurrentProblemType

Table 14: Attributes for WireInterfaceCurrentProblemType

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| problemName | String  Problem name not specified. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Name of the alarm according to supportedAlarmList attribute in MauType datatype |

### WireInterfacePerformanceType

Qualified Name: WireModel::TypeDefinitions::WireInterfacePerformanceType

Table 16: Attributes for WireInterfacePerformanceType

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| es | Integer  -1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: s * support: MANDATORY | Number of errored seconds according to 802.3 30.8.1.1.13 aLineESs |
| ses | Integer  -1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: s * support: MANDATORY | Number of severely errored seconds according to 802.3 30.8.1.1.12 aLineSESs |
| symbolErrorDuringCarrier | Integer  -1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Source: 802.3 according to 30.3.2.1.5 aSymbolErrorDuringCarrier. Number of times when valid carrier was present and an invalid data symbol occured. |
| lowPowerIdleTransmitterMs | Integer  -1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: ms * support: MANDATORY | Source: 802.3 similar to 30.3.2.1.8 aTransmitLPIMicroseconds. Number of milliseconds (original counter expresses microseconds), during which the transmitter was in power save mode |
| lowPowerIdleReceiverMs | Integer  -1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: ms * support: MANDATORY | Source: 802.3 similar to 30.3.2.1.9 aReceiveLPIMicroseconds. Number of milliseconds (original counter expresses microseconds), during which the receiver was in power save mode |

### WireInterfaceCurrentPerformanceType

Qualified Name: WireModel::TypeDefinitions::WireInterfaceCurrentPerformanceType

Turns performance information into current performance information by inheriting from OTN\_CurrentData.

Table 13: Attributes for WireInterfaceCurrentPerformanceType

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| performanceData | WireInterfacePerformanceType  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: false * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |

### WireInterfaceHistoricalPerformanceType

Qualified Name: WireModel::TypeDefinitions::WireInterfaceHistoricalPerformanceType

Turns performance information into historical performance information by inheriting from OTN\_HistoryData.

Table 15: Attributes for WireInterfaceHistoricalPerformanceType

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| performanceData | WireInterfacePerformanceType  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |

## Enumeration Types

### LoopBackType

Qualified Name: WireModel::TypeDefinitions::LoopBackType

Contains Enumeration Literals:

* NONE:
* EXTERNAL:
  + Parameter
  + If the internal loopback test succeeds but the external loopback fails, the Medium Attachment Unit (MAU) is faulty (e.g. SFP has to be replaced)
* INTERNAL:
  + If the internal loopback test fails, the Data Terminal Equipment (DTE) is faulty (e.g. board has to be replaced)
  + Parameter
* NOT\_YET\_DEFINED:

### SeverityType

Qualified Name: WireModel::TypeDefinitions::SeverityType

According to ITU-T M.3160

Contains Enumeration Literals:

* NON\_ALARMED:
* WARNING:
* MINOR:
* MAJOR:
* CRITICAL:

### MdiKindType

Qualified Name: WireModel::TypeDefinitions::MdiKindType

Contains Enumeration Literals:

* SC:
  + Subscriber Connector
* FIBRE\_CHANNEL\_STYLE\_1:
  + Copper connector
* FIBRE\_CHANNEL\_STYLE\_2:
  + Copper connector
* BNC\_TNC:
  + Bayonet/Threaded Neill-Concelman
* FC:
  + Fibre Channel coax headers
* FIBER\_JACK:
* LC:
  + Lucent Connector
* MT\_RJ:
  + Mechanical Transfer - Registered Jack
* MU:
  + Multiple Optical
* SG:
* OPTICAL\_PIGTAIL:
* MPO1x12:
  + Multifiber Parallel Optic
* MPO2x16:
  + Multifiber Parallel Optic
* HSSDC\_II:
  + High Speed Serial Data Connector
* COPPER\_PIGTAIL:
* RJ45:
  + 8P8C, according to Clause 3 and Figures 1 through 5 of IEC 60603-7
* NO\_SEPERABLE\_CONNECTOR:
* MXC2x16:
* ST:
  + according to IEC 60874-10:1992, also often called BFOC/2.5
* NOT\_YET\_DEFINED:

### MediumKindType

Qualified Name: WireModel::TypeDefinitions::MediumKindType

Contains Enumeration Literals:

* TP\_CAT3:
* TP\_CAT5:
* TP\_CAT6:
* TP\_CAT8:
* SINGLE\_MODE:
* MULTI\_MODE:
* NOT\_YET\_DEFINED:

### MiiKindType

Qualified Name: WireModel::TypeDefinitions::MiiKindType

Contains Enumeration Literals:

* GBIC:
* SOLDERED\_CONNECTOR:
* SFP\_SFP+\_SFP28:
* XBI\_300\_PIN:
* XENPAK:
* XFP:
* XFP\_E:
* XPAK:
* X2:
* DWDM\_SFP\_SFP+:
* QSFP:
* QSFP+:
* CXP:
* SHIELDED\_MINI\_MULTILANE\_HD\_4X:
* SHIELDED\_MINI\_MULTILANE\_HD\_(X:
* QSFP28:
* CXP2:
* CDFP\_STYLE1\_STYLE2:
* SHIELDED\_MINI\_MULTILANE\_HD\_4X\_FAN\_OUT:
* SHIELDED\_MINI\_MULTILANE\_HD\_8X\_FAN\_OUT:
* CDFP\_STYLE3:
* QSFP\_MICRO:
* QSFP\_DD:
* QSFP+\_RATE\_SELECT\_V1:
* QSFP+\_RATE\_SELECT\_V2:
* NOT\_YET\_DEFINED:

### DuplexType

Qualified Name: WireModel::TypeDefinitions::DuplexType

Contains Enumeration Literals:

* HALF\_DUPLEX:
* FULL\_DUPLEX:
* NOT\_YET\_DEFINED:

### RxSyncPreferenceType

Qualified Name: WireModel::TypeDefinitions::RxSyncPreferenceType

* Experimental

Contains Enumeration Literals:

* AUTO\_MULTIPORT\_DEVICE:
  + Master/Slave relation is determined by auto-negotiation and this device has a preference for being Master
* AUTO\_SINGLE\_PORT\_DEVICE:
  + Master/Slave relation is determined by auto-negotiation and this device has a preference for being Slave
* MANUAL\_MASTER:
  + Master/Slave relation is determined by manual configuration and this device shall be Master
* MANUAL\_SLAVE:
  + Master/Slave relation is determined by manual configuration and this device shall be Slave
* NOT\_YET\_DEFINED:

### RxSyncRoleType

Qualified Name: WireModel::TypeDefinitions::RxSyncRoleType

* Experimental

Contains Enumeration Literals:

* MASTER:
* SLAVE:
* FAULT:
* NOT\_YET\_DEFINED:

### SignalOrderingType

Qualified Name: WireModel::TypeDefinitions::SignalOrderingType

Contains Enumeration Literals:

* MDI:
* MDI\_X:
* NOT\_YET\_DEFINED:

### pmdNegotiationStateType

Qualified Name: WireModel::TypeDefinitions::pmdNegotiationStateType

* Experimental

Contains Enumeration Literals:

* NONE:
  + Device does not support auto negotiation
* DISABLED:
  + Auto negotiation is disabled.
* ENABLED:
  + Auto negotiation is enabled, but currently nothing is connected
* IN\_PROGRESS:
  + The auto-negotiation protocol is running and negotiation is currently in-progress
* COMPLETE:
  + The auto-negotation protocol has completed successfully
* FAILED:
  + The auto-negotation protocol has failed
* NOT\_YET\_DEFINED:
  + Device supports autonegotiation, but state is currently unknown, e.g. because device just booted

## Super Classes

### WireCurrentProblem

Qualified Name: WireModel::ObjectClasses::SuperClasses::WireCurrentProblem

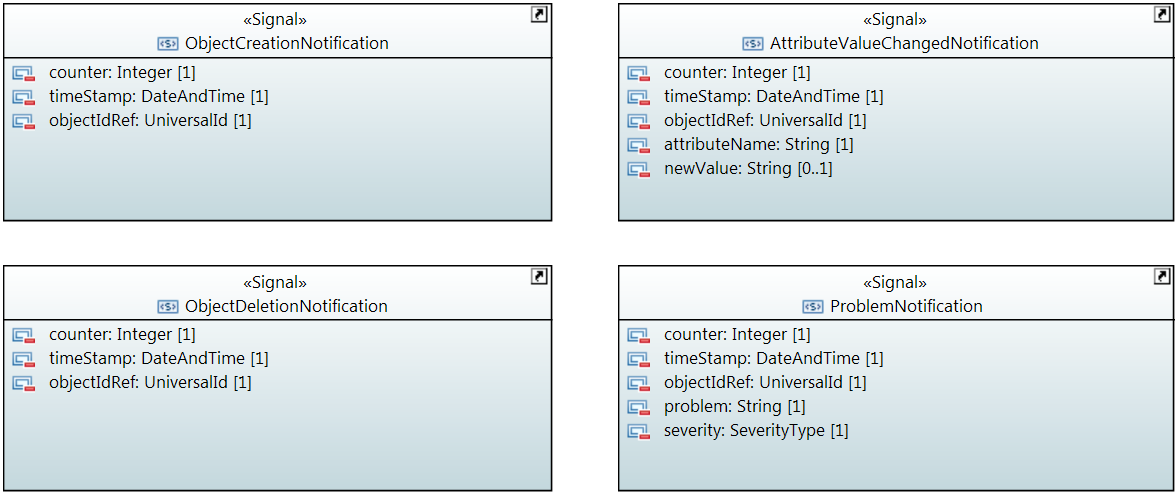
Applied stereotypes:

* OpenModelClass
* objectCreationNotification: NO
* objectDeletionNotification: NO
* support: MANDATORY

Table 18: Attributes for WireCurrentProblem

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| sequenceNumber | Integer  ./. | 1 | R | OpenModelAttribute   * partOfObjectKey: 1 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Unique sequence number of the current problem object. |
| timeStamp | DateAndTime  20101120140000.0Z+1 | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Time and date of the problem. \_format:yyyyMMddhhmmss.s[Z|{+|-}HHMm]; yyyy='0000'..'9999' year; MM='01'..'12' month; dd='01'..'31' day; hh='00'..'23' hour; mm='00'..'59' minute; ss='00'..'59' second; s='.0'..'.9'tenth of second (set to '.0' if EMS or NE cannot support this granularity); Z='Z' indicates UTC (rather than local time); {+|-}='+' or '-' delta from UTC; HH='00'..'23' time zone difference in hours; Mm='00'..'59' time zone difference in minutes. |
| problemSeverity | SeverityType  WARNING | 1 | R | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Severity of the alarm. |

## Notifications



### AttributeValueChangedNotification

Qualified Name: WireModel::Notifications::AttributeValueChangedNotification

To be sent when the value of an attribute has changed.

Applied stereotypes:

* OpenModelNotification
* triggerConditionList: invalid
* support: MANDATORY

Table 19: Attributes for AttributeValueChangedNotification

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| counter | Integer  -1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: no unit defined * support: MANDATORY | Counts attribute value changed notifications. |
| timeStamp | DateAndTime  20101120140000.0Z+1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |
| objectIdRef | UniversalId  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | ID of the affected \*\_Pac. |
| attributeName | String  Attribute name not specified. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Name of the attribute that has been changed. |
| newValue | String  New value not specified. | 0..1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Attribute value converted to a string (xml, json, ...) |

### ObjectCreationNotification

Qualified Name: WireModel::Notifications::ObjectCreationNotification

To be sent when a new \*\_Pac has been instancieted.

Applied stereotypes:

* OpenModelNotification
* triggerConditionList: invalid
* support: MANDATORY

Table 20: Attributes for ObjectCreationNotification

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| counter | Integer  -1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: no unit defined * support: MANDATORY | Counts object creation notifications. |
| timeStamp | DateAndTime  20101120140000.0Z+1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |
| objectIdRef | UniversalId  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | ID of the affected \*\_Pac. |
| objectType | String  Type of created object not specified. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Type of Object to be chosen from the following list of values: 'WireInterface\_Pac', 'MW\_PureEthernetStructure\_Pac' or 'MW\_EthernetContainer\_Pac'. |

### ObjectDeletionNotification

Qualified Name: WireModel::Notifications::ObjectDeletionNotification

To be sent when an instance of a \*\_Pac has been deleted.

Applied stereotypes:

* OpenModelNotification
* triggerConditionList: invalid
* support: MANDATORY

Table 21: Attributes for ObjectDeletionNotification

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| counter | Integer  -1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: no unit defined * support: MANDATORY | Counts object deletion notifications. |
| timeStamp | DateAndTime  20101120140000.0Z+1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |
| objectIdRef | UniversalId  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | ID of the affected \*\_Pac. |

### ProblemNotification

Qualified Name: WireModel::Notifications::ProblemNotification

To be sent when a problem occured.

Applied stereotypes:

* OpenModelNotification
* triggerConditionList: invalid
* support: MANDATORY

Table 22: Attributes for ProblemNotification

| **Attribute Name** | **Type DefaultValue** | **Multiplicity** | **Access** | **Stereotypes** | **Description** |
| --- | --- | --- | --- | --- | --- |
| counter | Integer  -1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: LENGTH\_32\_BIT * unit: no unit defined * support: MANDATORY | Counts problem notifications |
| timeStamp | DateAndTime  20101120140000.0Z+1 | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY |  |
| objectIdRef | UniversalId  ./. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | ID of the affected \*\_Pac. |
| problem | String  Problem name not specified. | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Name of the problem according to WireInterface::WireInterfaceCapability::supportedAlarmList. |
| severity | SeverityType  WARNING | 1 | RW | OpenModelAttribute   * partOfObjectKey: 0 * AVC: NO * isInvariant: true * valueRange: no range constraint * bitLength: NA * unit: no unit defined * support: MANDATORY | Severity of the problem according to WireInterface::wireInterfaceConfiguration::problemSeverityList. |

# References

|  |  |
| --- | --- |
| Reference | Comment |
| [IEEE802.3] | IEEE 802.3-2018 IEEE Standard for Ethernet, June 2018 (https://standards.ieee.org/standard/802\_3-2018.html) |
| [ONF Core IM] | TR-512 ONF Core Information Model Version 1.2, September 2016 (https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-512\_CIM\_(CoreModel)\_1.2.zip) |
| [ONF UMGL] | TR-514 UML Modeling Guidelines; Version 1.2, September 2016  (https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-514\_UML\_Modeling\_Guidelines\_v1.2.pdf and https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/IISOMI\_514\_UML\_Modeling\_Guidelines\_v1.2.pdf) |

# Back matter

## Editors

Thorsten Heinze

Telefonica Germany GmbH & Co. OHG

Martin Skorupski

highstreet technologies GmbH

## Contributors

Many thanks to Michael Binder, Giorgio Cazzaniga, George Clapp, Petr Jurcik, Danilo Pala, James Ries, Daniela Spreafico, Alexandru Stancu, Beatriz Ugrinovic, Tracy Van Brakle, Yossi Victor, Yang Yonggang, Bernd Zeuner and Nader Zein for reviewing and commenting of the Ethernet PHY Model.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_