Core Information Model (CoreModel)

TR-512.1

Overview

Version 1.2
September 20, 2016

ONF Document Type: Technical Recommendation
ONF Document Name: Core Information Model version 1.2

Disclaimer

Open Networking Foundation
2275 E. Bayshore Road, Suite 103, Palo Alto, CA 94303
www.opennetworking.org

Table of Contents

List of Figures

Document History

Version	Date	Description of Change
1.0	March 30, 2015	Initial version of the base document of the "Core Information Model" fragment of the ONF Common Information Model (ONF-CIM).
1.1	November 24, 2015	Version 1.1
1.2	September 20, 2016	Version 1.2 [Note Version 1.1 was a single document whereas 1.2 is broken into a number of separate parts]

1 Introduction

This ONF Technical Recommendation (TR) focuses on the Core Information Model (CoreModel) of the ONF Common Information Model (ONF-CIM). An information model describes the things in a domain in terms of objects, their properties (represented as attributes), and their relationships.

The ONF-CIM is expressed in a formal language called Unified Modeling Language (UML). UML defines a number of basic model elements, called UML artifacts. In order to assure consistent modeling, only a subset of these artifacts were used in the development of the ONF-CIM according to guidelines for creating an information model expressed in UML (documented in [ONF TR-514].

The ONF-CIM is formed from a number of pieces and is focused on the CoreModel. At its heart, the CoreModel provides a representation of network forwarding resources[1] from a management-control perspective. The CoreModel is independent of

- Specific forwarding technology, i.e. the CoreModel is forwarding technology neutral.
- Specific management-control interface protocol, i.e. the CoreModel is management-control interface protocol neutral (as described in [ONF TR-513]).

The ONF-CIM supports forwarding technology specific properties via application of the specification model (see 2.1.6 Core Specification Model (TR-512.7) on page 14) enabling reuse of existing technology specific standards definitions (e.g., from [ITU-T G.874.1]), pruned and refactored as appropriate (see [ONF TR-513]). The technology specific content, acquired in a runtime solution via "filled in" cases of specification, augment the CoreModel to provide a forwarding technology specific representation.

From an interfacing perspective, considering the SDN architecture [ONF TR-521] as an example, a controller may expose a view of the network in terms of ONF-CIM entities to client SDN controllers or applications to meet the needs of that client. The interface may expose the information in a client specific form where that form can be deterministically mapped to the ONF-CIM[2]. Tooling is used to generate an interface specific form from the UML[3] model.[4].

This document acts as a guide to the set of documents that describe the CoreModel of the ONF-CIM, providing:

- An introduction to the CoreModel of the ONF-CIM in the form of a brief overview of the model with links to the other documents in the set (see section 2.1 Model on page 8 and

[1] It is focused on representation of the functions/resources that have the primary purpose of supporting information forwarding (transfer and transform functions), that form a network that realizes virtual adjacency, for the purpose of control of those functions/resources. Those resources are referred to as network forwarding resources. The information model is not intended to cover functional resources that have a primary purpose of supporting storage or compute solutions.
[2] The Transport API (TAPI) provides an interface oriented representation (in UML) derived from the CoreModel using the “Pruning & Refactoring” process [ONF TR-513] supported by tooling.
[3] UML is not an interface protocol language.
[4] For example TAPI [OSSDN-SNOWMASS] uses Eagle [OSSDN-EAGLE] tooling to generate interface specific form (Yang and Jason).

section 2.2 Other documents on page 15 including a reference to the data dictionary (see section 2.2.1 Data Dictionary (TR-512.8) on page 15)).

- A brief explanation of how to introduce attributes and structure related to a specific network technology (see section 2.1.6 Core Specification Model (TR-512.7) on page 14).
- A terminology translation table (see section 2.2.2 Terminology mapping (TR-512.9) on page 16)
- An explanation of supporting guidelines with references to the guideline documents (see section 2.3 Supporting Guidelines on page 17).
- A summary of the main changes from the previous version (see section 3 Summary of main changes between version 1.1 and 1.2on page 1818).
- A list of references used in the document set (see section 4 References on page 19).
- A list of definitions used in the document set (see section 5 Definitions on page 21)
- A list of abbreviations used in the document set (see section 5.3 Abbreviations and acronyms on page 21).
- Some conventions used in the document set including key stereotypes and keys to the diagram symbol sets (see section 6 Conventions on page 24).
- A summary of future work (see section 7 Future CoreModel work areas on page 26)

Separate work activities are taking the CoreModel and deriving interface models (see [OSSDN-SNOWMASS]).

In addition to the documentation referenced above and throughout this document, the TR-512 delivery package includes the CoreModel in Papyrus UML (see OnfModel folder). The ongoing intention is to publish using the environment versions as stated in the guidelines [ONF TR-515]. The precise versions are stated below. The OpenModelProfile is not the latest available from GitHub at the time of publication due to a number of issues related to version migration.

Table 1: Tooling and Profile Versions

	[ONF TR-515] Version	GitHub Version	Version used for TR-512	Comments
Eclipse	4.5.x "Mars"		4.5.0.201506211200 "Mars"	
Papyrus	1.1.x		1.1.1.201508071204	
Gendoc	0.5.x (0.5.1)		0.5.0.201504200502	Minor misalignment
OpenModelProfile		0.2.4	0.2.0	Version to be aligned during next release

.

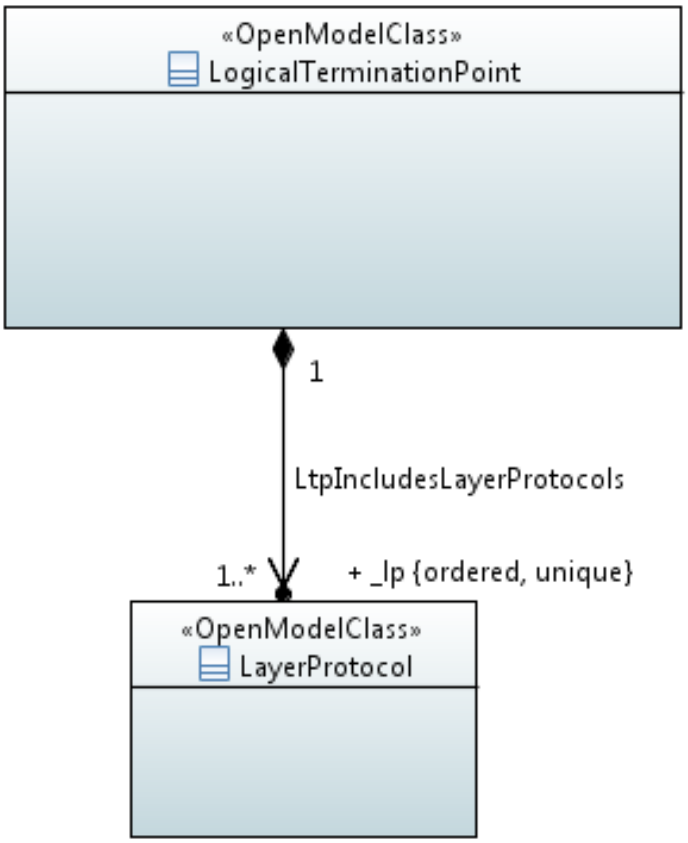
In addition an Experimental profile has been used for some of the Experimental model. This profile will be integrated into the formal profile structure in the next release.

1.1 Viewing UML diagrams

Some of the UML diagrams in figures are very dense. To view them either zoom (sometimes to 400%) or open the associated image file (and zoom appropriately) or open the corresponding UML diagram via Papyrus (for each figure with a UML diagram the UML model diagram name is provided under the figure or within the figure).

The UML diagram convention is provided in [ONF TR-514]. There are some key aspects of the diagrams that need to be emphasized.

- Association end attribute (the name of which always starts with "_") highlighted in the diagrams by the navigable end of the association (arrow head) is an attribute of the class at the non-navigable end of the association. It is the convention not to show the attribute in the class in the diagrams. The attributes for non-navigable ends (owned by the association) are not shown.
- On some occasions, other properties of the association end attribute are also shown.



CoreModel diagram: Overview-LtpAndLp

Figure 1-1 Illustrating navigable association end attributes

In the diagram above, the text at the arrow head end _lp… is an attribute of the Logical Termination Point.

This attribute is shown in the fragment of abbreviated data dictionary below for LogicalTerminationPoint.

Table 2: Attributes for LogicalTerminationPoint

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_lp		Ordered list of LayerProtocols that this LTP is comprised of where the first entry in the list is the lowest server layer (e.g. physical).

This sort of table is used in each of the documents on a section of the model and only provides summary information. For full information the reader should refer to the data dictionary (see section 2.2.1 Data Dictionary (TR-512.8) on page 15) or the model itself (see OnfModel folder).

2 Model Overview

This section provides an overview of the ONF Core Information Model and of the structure of the model description documentation. Each document described has a hyperlink that will take you to the document in your system[5]. The documents are all in the "ModelDescriptions" folder and are covered by two subsections:

- The documents referred to in Section 2.1 describe the core model artifacts progressing through the model from the basics of forwarding and termination through to a description of the augmentation mechanism of the specification model.
- The documents referred to in Section 2.2 provide additional supporting material including the Data Dictionary.

The remaining subsections provide:

- Related guidelines for model generation and usage (section 2.3)
- Key references (section 2.4)
- A brief overview of the Papyrus files (section 2.5)
- Some modeling principles applied (section2.6)

2.1 Model Document Structure

The Core Model of the ONF-CIM consists of model artifacts that are intended for use by multiple applications and/or forwarding technologies. For navigability, the Core Model is further sub-structured into Core Network Model (CNM), Core Foundation Model, Core Physical Model, and the Core Specification Model.

The Core Network Module (CNM) consists of artifacts that model the essential network aspects that are neutral to the forwarding technology of the network. The CNM currently encompasses Forwarding, Termination, Topology, and Resilience aspects (subsets of the CNM).

[5] The link will only work if you have unzipped the whole package as one.

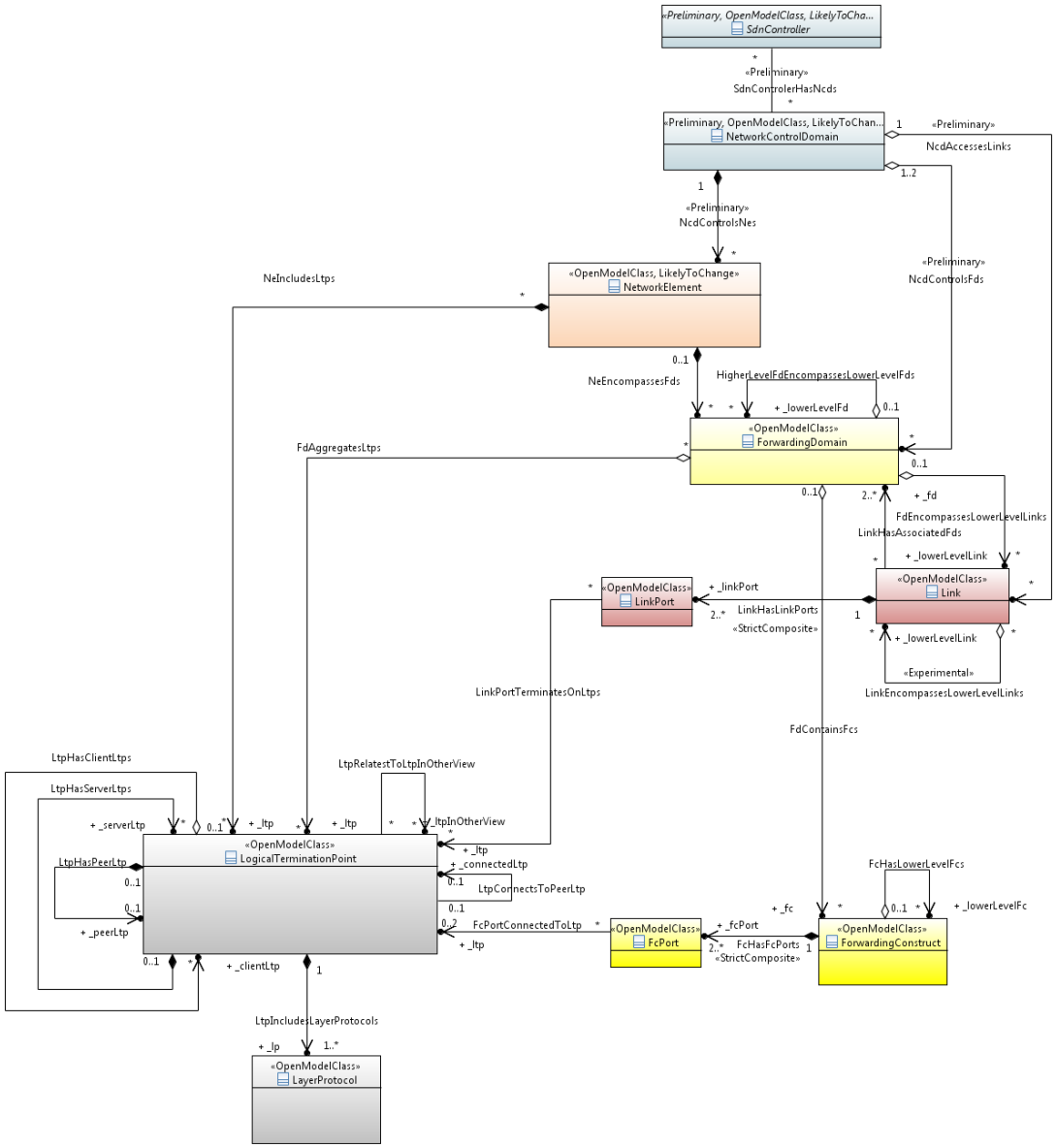
This section provides a list of all associated documents that describe the model. Each of the following sub-sections provides some brief highlights from the associated document and a link to that associated document.

The model documentation is broken down into a number of key parts which relate to but do not exactly match the model breakdown:

- Core Network Model:
  - Forwarding and Termination (see section 2.1.1)
  - Core Network Model: Topology (see section 2.1.3)
  - Core Network Model: Resilience (see section 2.1.4)
- Core Foundation Model (see section 2.1.2)
- Core Physical Model (see section 2.1.5)
- Core Specification Model (see section 2.1.6)
- Core Model Enhancements (see section 2.2.3)

2.1.1 Core Network Model – Forwarding and Termination Model (TR-512.2)

The Forwarding and Termination document provides a high-level overview of the Termination and Forwarding aspects of the CoreNetworkModel. This model is essentially a canonical model of networking from a management-control perspective. The figure below is a skeleton class diagram illustrating the interrelationships between key object classes defined in the CoreNetworkModel of the CoreModel. The classes are colored to help recognize key groupings in the model. The colors are chosen to match the key entity colors in Figure 6-1 Network diagram symbol set (with the Link in the alternative color for clarity). This color scheme for class diagrams is used in some of the figures in the associated documents.



CoreModel diagram: Forwarding-LtpInterLayerSkeletonOverview

Figure 2-1 Skeleton Class Diagram of key object classes

2.1.2 Core Foundation Model (TR-512.3)

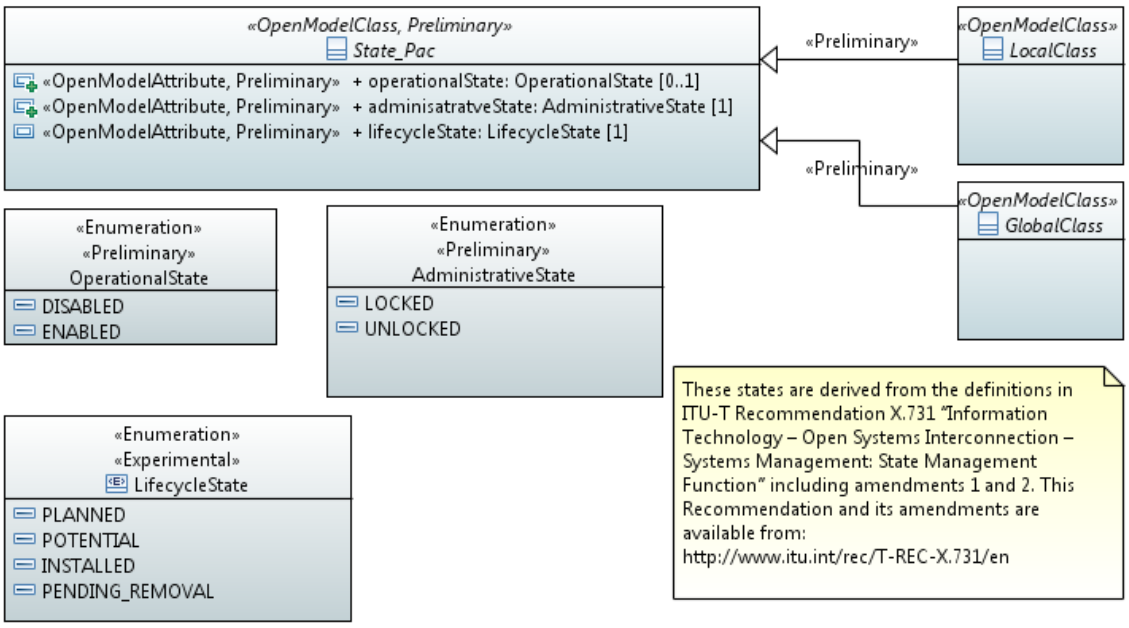
The Foundation document provides a detailed view of all aspects of the CoreModel that are relevant to all other parts of the ONF-CIM. Currently this model includes coverage of naming and identifiers as well as states.

2.1.2.1 Naming and Identifiers

Rationalizing the approach to naming, identification and addressing of entities described in the ONF-CIM

2.1.2.2 States

Basic states applicable to a majority of entities in the ONF-CIM

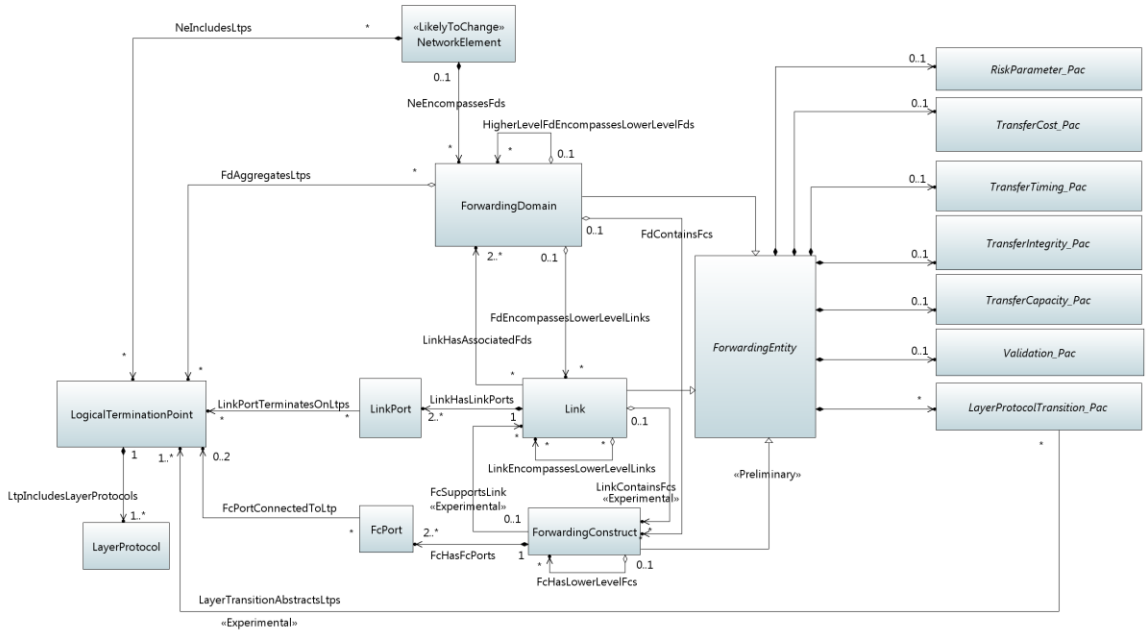


CoreModel diagram: GeneralizedStates

Figure 2-2 States for all Objects

2.1.3 Core Network Model – Topology Model (TR-512.4)

The topology document provides a detailed view of the topology model covering both the basic topology pattern with detailed attributes as well as the combination of layered topology and topology views.



CoreModel diagram: Topology-HighLevelOverviewOfStructureAndPacs-LargeText

Figure 2-3 Key classes that form the network topology

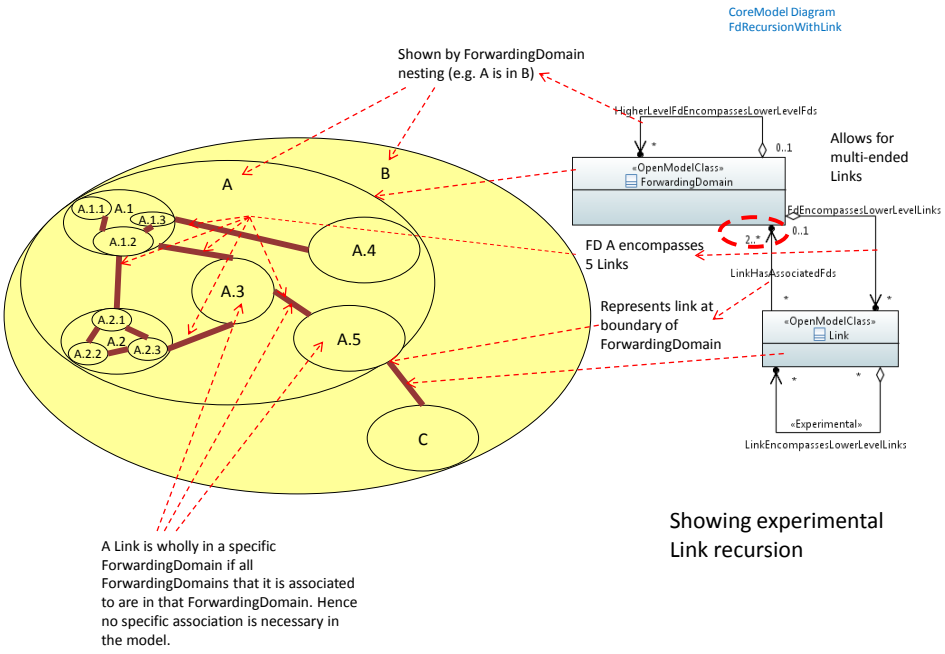


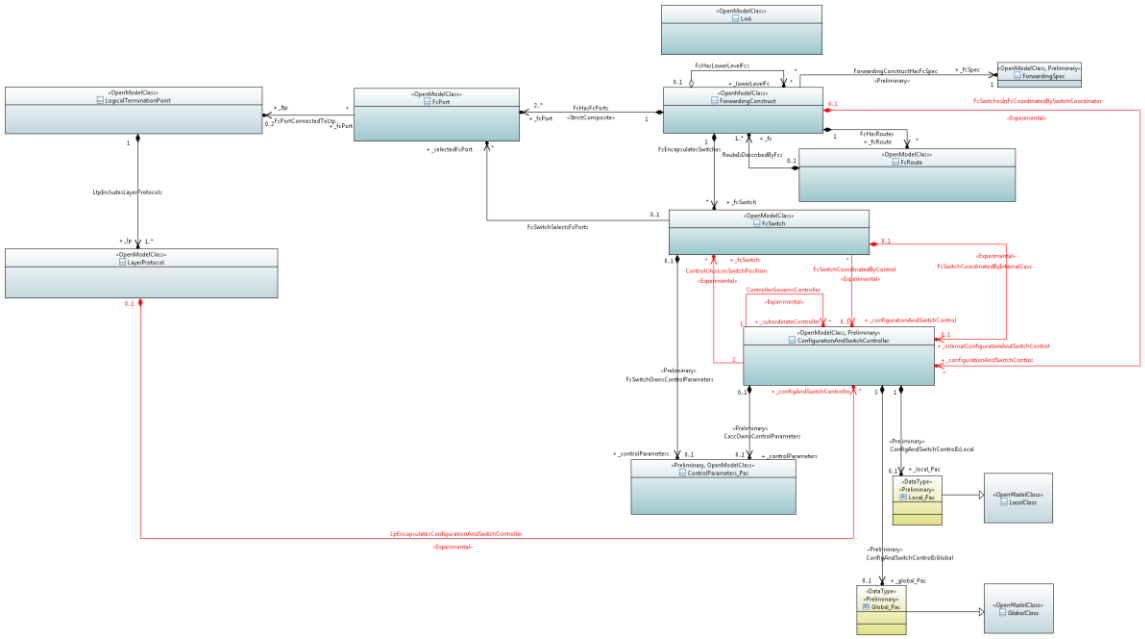
Figure 2-4 ForwardingDomain recursion with Link[6]

[6] The numbering of the FDs on the figure implies strict and fixed hierarchy. It should be noted that the association is aggregation and hence the hierarchy can change and an FD may move from being encompassed by one FD to being encompassed by another. Consider the numbering as simply a view of the current structure.

2.1.4 Core Network Model – Resilience Model (TR-512.5)

The Resilience document provides a view of the model for resilience (including protection and restoration) and encompasses:

- The basic resilience model structure
- The key attributes relevant to resilience
- The application of the resilience model to various cases



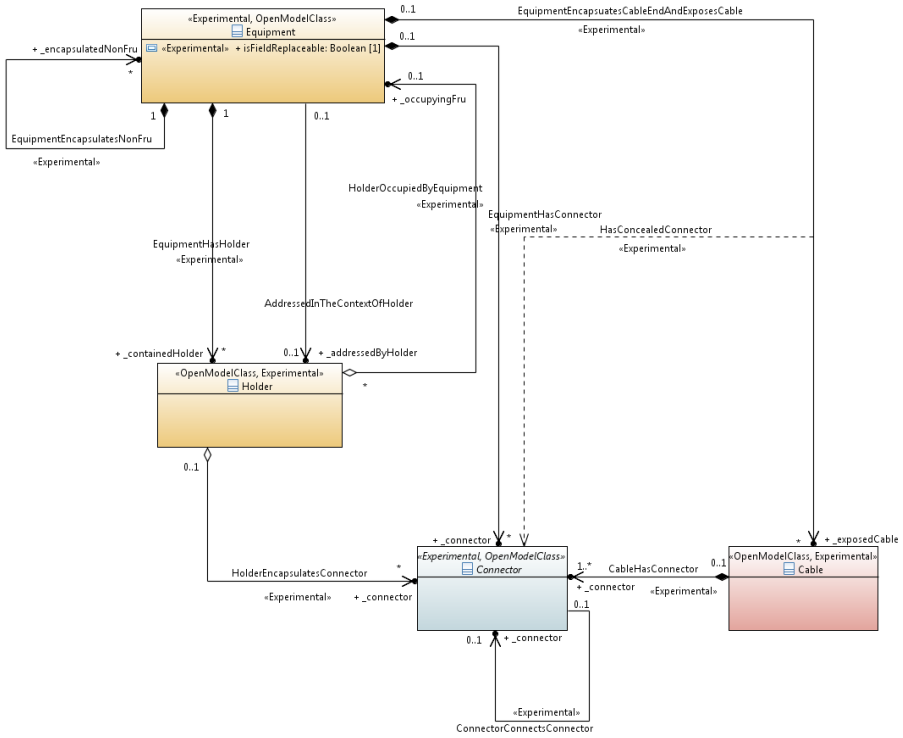
CoreModel diagram: Resilience-Pattern

Figure 2-5 Basic resilience pattern

2.1.5 Core Physical Model (TR-512.6)

The Physical model document provides a view of the model for physical entities (including equipment, holders and connectors). The document:

- Introduces the Physical model structure
- Describes the key classes of the Physical model
- Explains the attributes of the Physical model
- Describes the relationship between the connector and the LTP
- Shows how the model deals with the relationship between physical and functional views
- Explains how the Specification model describes equipment schemes (e.g. rules, etc.)
- Highlights work in progress to further advance the Equipment model



CoreModel diagram: Equipment-Pattern

Figure 2-6 Basic equipment pattern

2.1.6 Core Specification Model (TR-512.7)

There are several related needs that have given rise to the Specification model:

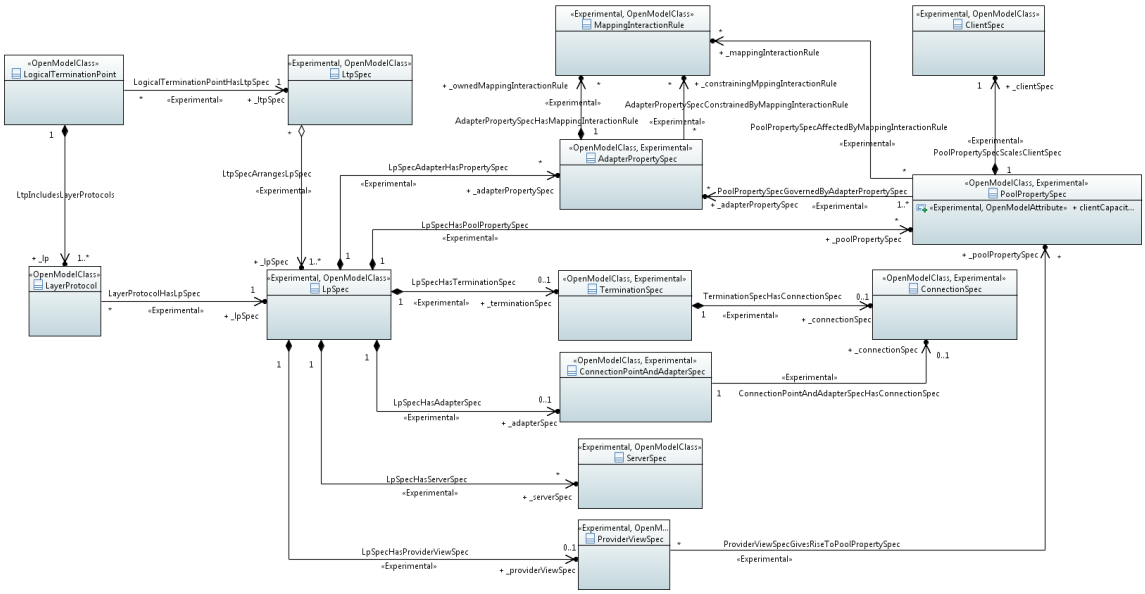
- Provide machine readable form of specific localized behavior:
  - Representing rules related to restrictions of specific cases of use of the model
  - Representing capabilities of specific cases of use
- Enable the introduction of run time schema where the essential structure of the model is known up front (at compile time) but the details are not
- Reduce the clutter in a representation where a set of details take the same values for all instances that related to a specific case
- Allow leverage of existing standards definitions (e.g., technology/application specific) in a machine readable language

The combination of the above resulted in a separation in the model of definitions of structure and content such that an instance of a class from one model fragment could have an association instance to another model fragment to enable the provision of a fragment of definition of the class and of subordinates.

The aim of all specification definitions is that they be rigorous definitions of specific cases of usage and enable machine interpretation where traditional interface designs would only allow human interpretation.

The following dedicated spec structures have been considered:

- FC spec: Main focus to provide a representation of the effective internal structure of a ForwardingConstruct (FC)
- LTP and LP spec: Main focus to provide a representation of Layer Protocol (LP) specific parameters for the Logical Termination Point (LTP)
- FD and Link spec: Main focus on capacity and forwarding enablement restrictions
- Equipment spec: Main focus to provide a representation of equipping constraints



CoreModel diagram: Spec-LtpCapabilitySpec

Figure 2-7 Class Diagram of the Spec Model of LTP and LP

In addition there is work on a generalized spec pattern with the main focus to provide a common representation of the mechanism for relating a class to its spec, accounting for implementation needs.

2.2 Other documents

2.2.1 Data Dictionary (TR-512.8)

The data dictionary provides details of the classes, attributes and data types (i.e. syntax) that are used in the model. The individual "model focuses" documents provide details on key classes and attributes but do not provide all details to avoid clutter and replication.

An extract from the data dictionary is shown below.

5.1.1.6 ForwardingConstruct

Qualified Name: CoreModel::CoreNetworkModel::ObjectClasses::ForwardingConstruct

The ForwardingConstruct (FC) object class models enabled potential for forwarding between two or more LTPs at a particular specific layerProtocol. Like the LTP the FC supports any transport protocol including all circuit and packet forms. It is used to effect forwarding of transport characteristic (layer protocol) information. An FC can be in only one FD. The ForwardingConstruct is a Forwarding entity. At a low level of the recursion, a FC represents a cross-connection within an NE. It may also represent a fragment of a cross-connection under certain circumstances. The FC object can be used to represent many different structures including point-to-point (P2P), point-to-multipoint (P2MP), rooted-multipoint (RMP) and multipoint-to-multipoint (MP2MP) bridge and selector structure for linear, ring or mesh protection schemes.

Applied stereotypes:

- OpenModelClass
  - objectCreationNotification: NA
  - objectDeletionNotification: NA
  - support: MANDATORY

Table 1: Attributes for ForwardingConstruct

Attribute Name	Type	Multiplicity	Access	Stereotypes	Description
layerProtocolName	LayerProtocolName	1	RW	OpenModelAttribute • AVC: NA • valueRange: no range constraint • support: MANDATORY	The layerProtocol at which the FC enables potential for forwarding.
_lowerLevelFcRefList	ForwardingConstruct	0..*	RW	OpenModelAttribute • AVC: NA • valueRange: no range constraint • support: MANDATORY	An FC object supports a recursive aggregation relationship such that the internal construction of an FC can be exposed as multiple lower level FC objects (partitioning). Aggregation is used as for the FD to allow changes in hierarchy. FC aggregation reflects FD aggregation. The FC represents a Cross-Connection in an NE. The Cross-Connection in an NE is not necessarily the lowest level of FC partitioning.

Figure 2-8 Extract from data dictionary

2.2.2 Terminology mapping (TR-512.9)

The terminology mapping document contains a table that provides overview translations from classes in the ONF-CIM to classes (and concepts) in other models. It will be helpful for someone who is familiar with one of the other industry standard terminology sets when working through the ONF-CIM.

2.2.3 Core Model Enhancements (TR-512.10)

This document provides fragments of ongoing work. The data dictionary document does NOT include entities from this document. All the work in this document is experimental.

The document covers:

- Modeling enhancements including adding rules to the core model (loop and spiral)
- Management-Control-Component model including Processing Construct
- Operations model patterns
- Information architecture and patterns
- Additional inter-view interrelationship considerations
- Further Resilience model enhancements (including details of the support for [ITU-T G.8032])

2.2.4 Gendoc fragment definitions (TR-512.11)

All Gendoc templates are provided in the Gendoc folder. TR-512.11 provides a base document from which all other documents are derived along with examples of usage.

2.3 Supporting Guidelines

Several guideline documents have been constructed to maintain consistency in the models generated by ONF. These guidelines have also been shared with organizations outside ONF.

- **[ONF TR-513]:** This document specifies the principles and guidelines for the development and use of the ONF-CIM, including guidelines for deriving purpose-specific information model views (through pruning and refactoring selected subsets of artifacts from the ONF-CIM), and mapping to data schemas for protocol-specific control interfaces.
- **[ONF TR-514]:** The ONF-CIM is expressed in a formal language called UML (Unified Modeling Language). UML has a number of basic model elements, called UML artifacts. In order to assure consistent modeling, only a subset of the UML artifacts is used in the development of the ONF-CIM. The selected subset of UML artifacts is documented.
- **[ONF TR-515]**: This document specifies the guidelines for using the Papyrus tool used in the development of the ONF-CIM. It also describes how the Common IM modeling teams can cooperate in the GitHub environment for separate and coordinated development of the ONF-CIM fragments.

2.4 Key reference material

In the development of the CoreModel, information model work from other SDOs has been used as input, including [TMF TR215], [TMF TR225], [TMF SID 5LR], [ITU-T G.7711], [ITU-T G.874.1], [ITU-T G.8052], and [ITU-T G.8152]. The CoreModel is being shared with other bodies via various mechanisms including publication of a view of the model as an IETF draft [draft-lam].

2.5 Papyrus File

This section provides the link to the information model file and the companion Open Model Profile file specified using the "Papyrus" modeling tool.

Link to the Core Model files: OnfModel folder.

The file structure is as follows:

- .project,
- CoreModel.di,
- CoreModel.notation
- CoreModel.uml
- OpenModel_Profile.profile.di
- OpenModel_Profile.profile.notation
- OpenModel_Profile.profile.uml
- Experimental.profile.di[7]
- Experimental.profile.notation
- Experimental.profile.uml

[7] The Experimental profile provides some stereotypes related to experimental rules (e.g. in the Physical model). The relevant stereotypes in this profile will be moved to a formal profile in the next release.

In order to view and further extend or modify the information model, install the open source Eclipse software and the Papyrus tool. The installation guide for Eclipse and Papyrus can be found in [ONF TR-515].

2.6 Some modeling principles

2.6.1 Encapsulation

- If the positional bounds of two related concept instances are coincident for their entire lifecycle then they may be merged into a single entity instance representing the composite concept and hence share an identifier etc
- If the positional bound of one concept instance is a subset of the positional bound of another concept to which it is related for its entire lifecycle and where that larger concept can be considered as a dominate definition then it may be subsumed into the entity representing the larger concept and hence be identified as part of the entity for that larger concept in terms of attributes of that larger concept
- If the positional bounds of several instances of a concept are all subsets of the positional bound of a another concept to which they are related for their entire lifecycle and where that larger concept can be considered as a dominate definition then they may be subsumed into the entity via a composition relationship
- If a concept instance that bridges two other concept instances (of the same or different types) is, in the particular case, devoid of anything but identity then it may be represented simply by associations between the entities representing the two other concept instances
    - The associations may be two way navigable or one way navigable depending upon the original associations
- If a concept instance that is a leaf is devoid of anything but identity then it may be omitted

2.7 Boundary of the work

As noted, the ONF Core IM does not cover interface definition. As a consequence certain stereotype values are not relevant and hence are left at default including objectCreationNotification, objectDeletionNotification, passByReference. A majority of the attributes are read/write as in most cases a view can be conceived that will allow the attribute to be written.

3 Summary of main changes between version 1.1 and 1.2

Changes to the model and/or related documentation:

- General
    - Change to doc structure
    - Change to gendocs
- Forwarding and Termination
    - Minor corrections made to multiplicities

    - Improvements made to documentation
- Foundation
    - An address structure has been added as «Experimental»
- Topology
    - Change of name of TopologicalEntity to ForwardingEntity
    - Incorporation of FC under ForwardingEntity and consideration of FC as closely related to topology
    - Capacity
- Resilience
    - Protection, restoration and recovery attributes added
    - Structure enhanced
    - Association to LTP from protection added in preparation for G.8032 modeling
    - Various examples of usage developed to both prove and document the resilience model
    - Corrections made to some multiplicities
    - Lifecycle stereotypes adjusted to reflect the advancing maturity of the model
- Equipment
    - New model added as «Experimental»
    - Focus of the model is on the pattern but all experimental work has been published as equal
- Specification
    - Addition of FD/Link spec details in terms of FD/Link capability statements
    - Refinement of FC spec to accommodate the Link (removal of Fc from class names in the spec and generalization to Forwarding recognizing the Link as a Forwarding entity).
    - Addition of sketches of the generalized spec model
    - Enhancements to details on LTP/LP spec and discussion on migration

4 References

[draft-lam]	IETF draft-lam-teas-usage-info-model-net-topology-02, Usage of IM for network topology to support TE Topology YANG Module Development
[IETF RFC4122]	IETF RFC 4122 (July 2005) A Universally Unique Identifier (UUID) URN Namespace
[ISO/IEC 19505]	ISO/IEC 19505:2012 Information technology -- Object Management Group Unified Modeling Language (OMG UML)
[ITU-T G.800]	Recommendation ITU-T G.800 (02/2012), *Unified functional architecture of transport networks*
[ITU-T G.805]	Recommendation ITU-T G.805 (03/2000), *Generic functional architecture of transport networks*
[ITU-T G.808.1]	Recommendation ITU-T G.808.1 (05/2014), *Generic protection switching – Linear trail and subnetwork protection*

[ITU-T G.852.1]	Recommendation ITU-T G.852.1 (11/1996), *Enterprise viewpoint for simple subnetwork connection management*
[ITU-T G.852.2]	Recommendation ITU-T G.852.1 (11/1996), *Enterprise viewpoint description of transport network resource model*
[ITU-T G.874]	Recommendation ITU-T G.874 (08/2013), *Management aspects of optical transport network elements*
[ITU-T G.874.1]	Recommendation ITU-T G.874.1 (10/2012), *Optical transport network: Protocol-neutral management information model for the network element view*, plus Amendment 1 (08/2013)
[ITU-T G.7711]	Recommendation ITU-T G.7711/Y.1702 (ex. G.gim) (08/2015), *Generic Protocol-Neutral Information Model for Transport Resources*
[ITU-T G.8001]	Recommendation ITU-T G.8001/Y.1354 (04/16), *Terms and definitions for Ethernet frames over transport*
[ITU-T G.8032]	Recommendation ITU-T G.8032/Y.1344 (08/15), *Ethernet Ring Protection Switching*
[ITU-T G.8052]	Recommendation ITU-T G.8052/Y.1346 (08/2013), *Protocol-neutral management information model for the Ethernet Transport capable network element*
[ITU-T G.8081]	Recommendation ITU-T G.8081 (02/2012), *Terms and definitions for automatically switched optical networks*
[ITU-T G.8152]	Recommendation ITU-T G.8152/Y.1375 (draft), *Protocol-neutral management information model for the MPLS-TP network element*
[ITU-T M.3100]	Recommendation ITU-T M.3100 (04/2015) , *Generic network information model*
[ITU-T M.3400]	Recommendation ITU-T M.3400 (02/2000), *TMN management functions*
[ITU-T X.731]	Recommendation ITU-T X.731 (01/1992), *Information technology - Open Systems Interconnection - Systems management: State management function*
[ONF TR-521]	TR-521 (V1.1):: ONF SDN Architecture 1.1, June 2014 (https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-521_SDN_Architecture_issue_1.1.pdf)
[ONF TR-513]	TR-513 (V1.2): ONF Common Information Model Overview (https://www.opennetworking.org/sdn-resources/technical-library)
[ONF TR-514]	TR-514 (V1.2): ONF UML Model Guidelines (https://www.opennetworking.org/sdn-resources/technical-library)
[ONF TR-515]	TR-515 (V1.2): ONF Papyrus Guidelines (https://www.opennetworking.org/sdn-resources/technical-library)
[OSSDN-EAGLE]	Project EAGLE: ONF Open Model, Profiles and Tools (https://github.com/OpenNetworkingFoundation/EAGLE-Open-Model-Profile-and-Tools and https://community.opensourcesdn.org/wg/EAGLE/workgroup)

[OSSDN-SNOWMASS]	Project SNOWMASS: ONF Transport API (https://github.com/OpenNetworkingFoundation/Snowmass-ONFOpenTransport and https://groups.opensourcesdn.org/wg/SNOWMASS/dashboard)
[TMF 612]	TM Forum MTOSI (4.0), Multi-Technology OS Interface
[TMF TR215]	TMF TR215 (V0.5.3) Logical Resource Network Model Advancements and Insights (liaised to ONF)
[TMF TR225]	TM Forum TR225 (R15.0.0), Logical Resource: Network Function Model (liaised to ONF)
[TMF SID 5LR]	TM Forum GB922 (R15.0.0) Information Framework (SID) Addendum 5LR (liaised to ONF)

5 Definitions

5.1 Terms defined elsewhere

This document uses terms defined elsewhere. These terms are highlighted in section 5.3 Abbreviations and acronyms below by referring to the definition source document.

5.2 Terms defined in this TR

The primary purpose of this document is to define terms and hence terms are defined throughout the document. Key terms are highlighted in section 5.3 Abbreviations and acronyms below by referring to the section in this document where the term is defined.

5.3 Abbreviations and acronyms

This TR uses the following abbreviations and acronyms (Note that some cross references are included here rather than in the References section where the cross reference is only relevant for abbreviation/acronym interpretation purposes):

AP	Access Point [ITU-T G.805]
API	Application Programmer's Interface
C&SC	Configuration and Switch Controller (model entity)
CASC	C&SC
CNM	Customer Network Management
CP	Connection Point [ITU-T G.805]
CRUD	Create Read Update Delete
CTP	Connection Termination Point. Note that definitions differ between TM Forum [TMF 612] and [ITU-T M.3100]. Both usages apply here when referring to legacy cases and the abbreviation is qualified in all cases of use.

ECC	Embedded Communications Channel [ITU-T G.874]
EMS	Element Management System [definition reference ITU-T M.3400 - TMN][8]
ETH	Ethernet MAC Layer [definition reference ITU-T G.8001]
ETY	Ethernet Physical Layer [definition ITU-T G.8001]
FC	ForwardingConstruct (defined in the ONF-CIM - see TR-512.2). • Note that at this point the definition is subtly different to that in [TMF TR225]. The aim is to align the terms usage
FDFr	FlowDomainFragment [TMF 612]
FRE	ForwardingRelationshipEncapsulation [TMF TR215]
FTP	FloatingTerminationPoint [TMF 612]
GitHub	See www.github.com
GUID	Globally Unique IDentifier (see www.wikipedia .org/Globally_unique_identifier)
IM	Information Model (see section 1 Introduction above)
IMP	Inverse MultiPlexing [ITU-T G.805]
ISO	International Organization for Standardization (see www.iso.org)
ITU	International Telecommunications Union (see www.itu.int)
ITU-T	Telecommunications Standardization Sector of ITU-T
LP	LayerProtocol (defined in the ONF-CIM – see TR-512.2). Note that there are two related terms: • layer-protocol: used to refer to the information transfer protocol (or Characteristic Information of the signal) • layerProtcolName: used to refer to the attribute in the LP class that carries the value that identifies the characteristic layer-protocol of the LP • LayerProtocolName: used to refer to the data type that holds the formal name of the layer-protocol
LTP	LogicalTerminationPoint (defined in the ONF-CIM - see TR-512.2)
MAC	Media Access Control
MEG	Maintenance End Group
MEP	MEG End Point
MFDFr	MatrixFlowDomainFragment
MLSN	MultiLayerSubNetwork [TMF 612]
MP2MP	Multi-Point to Multi-Point
MPLS-TP	Multi-Protocol Label Switching Transport Profile [definition reference RFC6378]
NCD	NetworkControlDomain
NE	NetworkElement
OAM	Operations Administration and Maintenance

[8] This term is not intended for use other than in reference to legacy systems.

OCh	Optical Channel
ODU	Optical Data Unit
OMS	Optical Multiplex Section
ONF-CIM	ONF Common Information Model
OPS	Optical Protection Switch
OS	Operations System (essentially OSS - Operation Support System)
OTN	Optical Transport Network
OTS	Optical Transmission Section
OTU	Optical channel Transport Unit
P2MP	Point to Multi-Point
P2P	Point to Point
PON	Passive Optical Network
PTP	Physical Termination Point [TMF 612]
RMP	Rooted Multi-Point
SDN	Software Defined Networking [ONF]
SDO	Standards Development Organization
SNC	SubNetworkConnection [TMF 612]
SNP	SubNetworkPoint [ITU-T G.8081]
TAPI	Transport API
TBD	To Be Defined
TCP	Termination Connection Point [ITU-T G.805]
TDM	Time Division Multiplex
TMF	TeleManagement Forum (see www.tmforum.org)
TP	Termination Point [ITU-T M.3100]
TPE	TerminationPointEncapsulation [TMF TR215]
TR	Technical Recommendation [ONF] Technical Report [TM Forum]
TRI	Transport Resource Identifier [ITU-T G.8081]
TTP	Trail Termination Point [ITU-T M.3100]
UML	Unified Modelling Language (see www.omg.org)
UUID	Universally Unique IDentifier (see https://en.wikipedia.org/wiki/Universally_unique_identifier)
VCAT	Virtual Concatenation
VNE	Virtual Network Element
XC	CrossConnection

6 Conventions

6.1 Lifecycle Stereotypes

Lifecycle stereotypes (see [ONF TR-514]) are applied to entities in the model to indicate their degree of maturity[9]. These are made visible in many of the figures in this document.

The following stereotypes appear in TR-512:

- «Experimental»: Indicates that the entity is at a very early stage of development and will almost certainly change. The entity is NOT mature enough to be used in implementation[10].
- «Preliminary»: Indicates that the entity is at a relatively early stage of development and is likely to change but is mature enough to be used in implementation.

If no stereotype is shown the entity is mature. Other Lifecycle Stereotypes are defined in [ONF TR-514] .

6.2 Key to diagram symbol set

This document set includes a number of UML diagrams. The UML symbol set is suitably explained in [ONF TR-514]. Many of the UML diagrams in this document have small font (due to density of information conveyed). It will be necessary for the reader to zoom in and pan across the figure to see the detail[11].

This document set also contains a number of non-UML diagrams, which use the symbols highlighted below in pictorial representations of network examples. The symbol set is an advanced partial hybrid of symbols used by other bodies (see [TMF TR215] and [ITU-T G.805]).

[9] The whole model including all degrees of work in progress has been published to allow the user maximum opportunity to set a most consistent direction with the work at hand. It is considered important to expose work in progress especially where this may have an impact on a choice of implementation. There may be some experimental structure that contains some very stable parts, without that structure those parts might be quite uninterpretable. A user who decides to take a low risk approach can ignore preliminary and experimental parts. A user who is more inclined to take a risk or who is looking for inspiration for their work can take the experimental and preliminary parts, understanding the risk involved.

[10] The implementer can clearly choose to use the item at risk (expecting change and accounting for this in deployments etc.)

[11] The aim is to improve the figure readability in future releases.

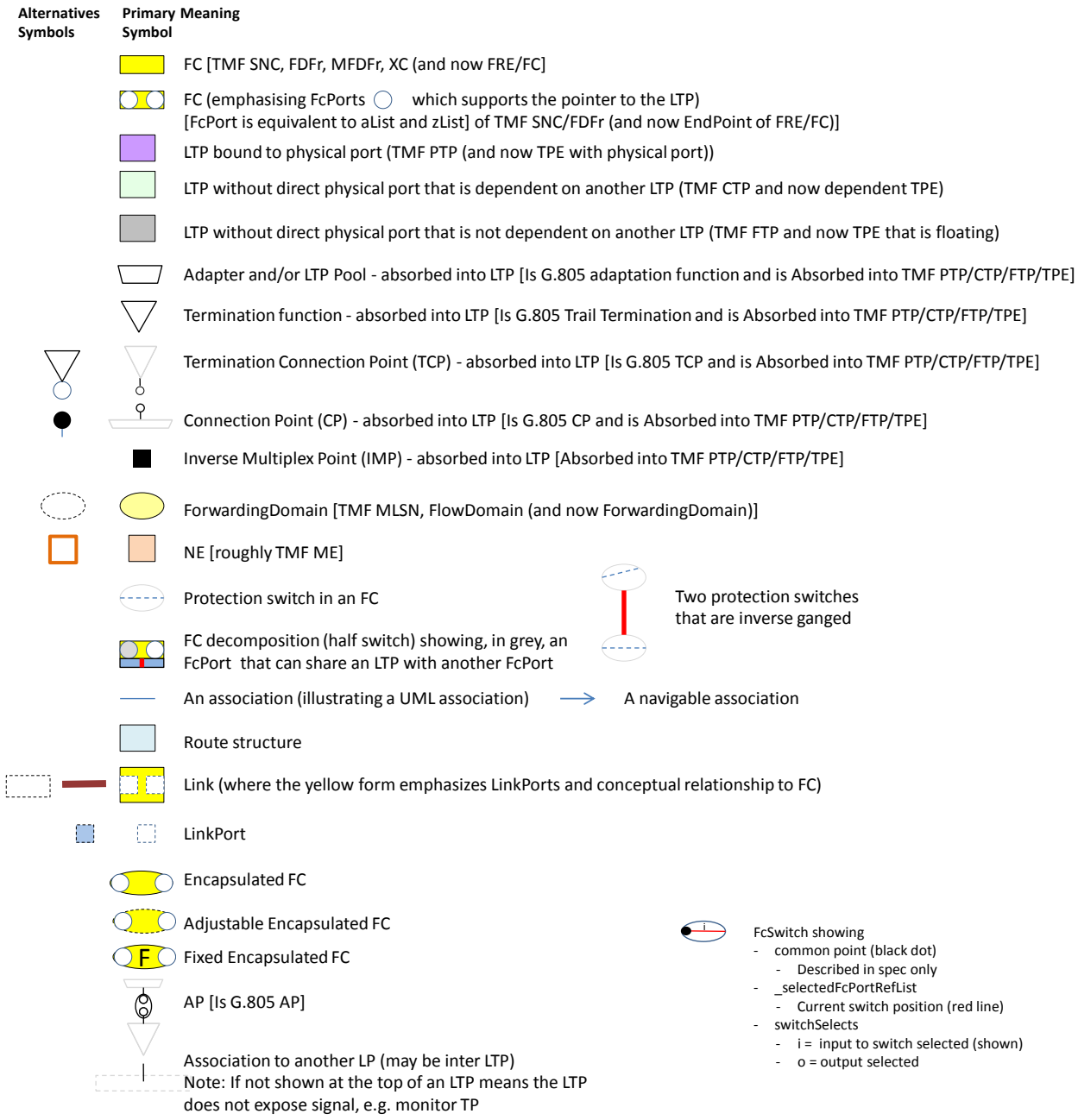


Figure 6-1 Network diagram symbol set[12]

[12] It should be noted that in this version and future versions the terms ForwardingDomain (FD) and ForwardingConstruct (FC) are used in place of SubNetwork (SN) and SubNetworkConnection (SNC) (used in the earlier versions of the ONF-CIM).

7 Future CoreModel work areas

Potential future areas of work in the CoreModel include, not in any particular order:

- Management-Control Components
  - Development of the Controller model (including NE, NCD, Controller component, Network core control, etc.)
- Signaling
  - Developing models for signaling in the context of ECC and protection
- Interface patterns
  - Completion of the generalized operations pattern covering range of cases including intend and CRUD
  - Understand the relationship with Dynamic APIs and Strategic Mediation
- Support for specific interface development
  - TAPI
  - Intent model
- OAM functions
  - Generalization of OAM functions, e.g., generalized MEPs
  - Consider use cases and scenarios to guide development of the generalized model
- Assurance
  - Modeling of events and the reporting of events
- General processing construct model (and the component system-pattern)
  - Developing the model of the recursion of function abstractions from the base equipment through functional protection to the supporting of LTPs etc.
- Patterns and architectures
  - Construction of models that explore the pattern underlying Link/FC/FD and minimally represent that pattern and show derivation of Link/FC/FD from that pattern.
- Dependency graph representation of telecommunications technology (including flow semantics)
  - For expression of detailed processes of a telecommunications technology to enable interpretation of a new technology
- Profiles, Templates and Specifications
  - Completion of spec model and addition of profiles model in the spec context
  - Further development of constraint models (also covering policy)
  - Complete pattern and migrate model to use pattern
  - Develop class based rule mechanism and consider more fluid approach to Core model
  - Provide further examples of usage
  - Develop detailed rules
  - Refine model to deal with rule interaction
  - Consider FD/FC spec convergence
    - FD ports would be necessary but these have essentially been subsumed by the LTP (this relates to the general component-system pattern)

  - When dealing with Compound Links we need to consider whether rules are necessary for Link (the same structure will apply but an additional association from Link to FD will be necessary)
  - Development of a specification toolkit including standardized rules and structures
  - DSGL (Domain Specific Graphical Language to ease spec construction)
  - Model v specification:
    - Implication of the work so far is that the specification structure is the model structure and that the schema for any particular case has some parts of the structure in compile time form and other parts in runtime only form where the runtime form may have static parts only in the spec form
    - Is a replication of the model structure in formal model but that formal model should be decoupled at various points and extensible in a constrained way at various points.
    - Considerations of "model viscosity" (all models are fluid over some timeframe)
  - Dealing with LTP and LP formal sub-structuring challenge
    - Related to the previous bullet… should the LP sub structuring of the spec model be part of the LP model
  - Migration of operations from non-spec to spec
    - Continuum of usage approaches from "phrase book user" to "orator"
  - FcSpec refinements
    - Should the FC spec be rationalized to recognise that MSUF is essential an FC (implications etc)?
  - Moving to the generalized spec
    - How should constraints on each class spec be expressed in the context of a generalized spec?
  - Understanding the relationship to Dynamic APIs, Strategic Mediation and the Operations Patterns
  - Enhance the extension mechanism
- Resilience
  - Validation of the protection model for support of G.8032 and for other ring schemes not yet covered
  - Development of protection scheme specifications and generalization of these to deal with any network structure specification
  - Add details from resilience spreadsheet to the documentation (and model comments where appropriate).
  - Carry out further work on the unexpected flow query
  - Develop a model of signalling (note that this should be part of the MCC work as the C&SC is a controller
  - Refine documentation on relationship to protectionGroups
  - Complete route feed port relationship
  - Document the route lifecycle
  - Develop resilience scheme spec as the start of a general network structure spec model
  - Indication of encapsulated resilience on FC and Link
- Timing and Synchronization model (frequency and time/phase)

    - Construction and development of a model of synchronization based on the FC and LTP derived from work in ITU-T
- Physical Equipment
    - Completion of the equipment model
    - Expectation v actual
    - Attribute details
    - Rationalize attribute groupings
    - Look for source for physical properties
    - Separate out functional work into other work areas (ProcessingConstruct and OAM functions)
    - Separate out Management-Control parts into Management-Control model
    - Refine and move specification model detail from Physical model document to Specification model document (and move model as appropriate).
- FC, FD, Link and Topology
    - Various detailed enhancements including considerations of merging of FC and Link
    - Consider merging FD and Link
    - Derive FD/FC/Link from Component-System pattern
    - Further clarification of off-network "things" (could be a link topology)
    - Various illustration of FD/Link lifecycle
        - Consider generalized lifecycle state and state interaction
    - Serial compound link
    - Cost algorithm
    - Illustrate use of model for inverse multiplexing cases
    - Improve documentation on terminationState
- View abstraction
    - Enhancements to view abstraction examples and cases, including FD view, FC view, Call view, Service view, Connection view
    - Further work on rules for virtualization (e.g. what from one view can be grouped in the same link from another view)
- Mapping to other models
    - Enhancements to the mapping to OpenFlow
    - Development of mappings to IETF models
    - Aim for convergence of industry models
- Temporal properties
    - Determining how to inject temporal aspects into the model process
- Tooling
    - Enhancement to UML YANG to cover specification models
    - Enhancements to pruning and refactoring process and tooling
- Minor enhancements
    - Rename the LayerProtocol class
- Documentation
    - Ongoing improvements

8 Terminology Translation table

The translations provided in this release are early draft (see TR-512.9 - ONF Core IM - Terminology Translation Table). There may be errors in the table and the table is not complete. It should be used for guidance only.

9 Documentation structure

The following figure depicts the structure of TR-512. The UmlFigures folder has a subfolder for each document that includes a UML figure where each subfolder contains the UML figures from the corresponding document. These figures have been included to aid viewing as some are very detailed. The Gendoc Templates are used to generate the TR-512 documents. They do not provide any model information that is not in the .pdf documents and hence they can be ignored. They have been delivered to ensure inter-release continuity and will be used as a basis for the construction of the documentation for the next release of model.

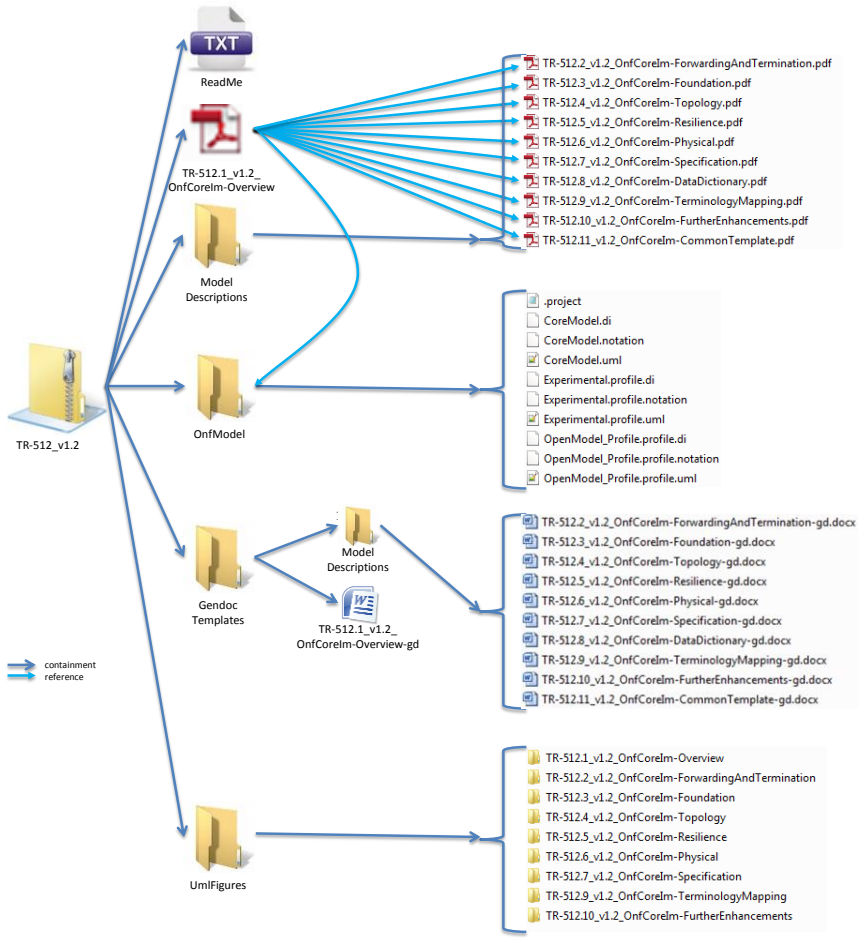


Figure 9-1 TR-512 structure

10 Individuals engaged

10.1 Editors

Nigel DAVIS, Ciena
Kam LAM, Nokia

10.2 Contributors

Hui DING	CATR, China
Yun Bin XU	CATR, China
Nigel DAVIS	Ciena
Christopher HARTLEY	Cisco
Jonathan SADLER	Coriant
Bernd ZEUNER	Deutsche Telekom
Erez SEGEV	ECI
Dave HOOD	Ericsson
Meral SHIRAZIPOUR	Ericsson
Scott MANSFIELD	Ericsson
Xiang YUN	FiberHome
Yuji TOCHIO	Fujitsu
Italo BUSI	Huawei
Maarten VISSERS	Huawei
Raymond CHEH	Juniper
Karthik SETHURAMAN	NEC
Andre MAZZINI	Nokia
Dieter BELLER	Nokia
Sergio BELOTTI	Nokia
Kam LAM	Nokia
Eve VARMA	Nokia
Germano GASPARINI	SM-Optics
Malcolm BETTS	ZTE
Rod LU	ZTE
Qi Lei WANG	ZTE
Xiong QUAN	ZTE

End of Document