

IEEE Standard for Control and Management of Virtual Links in Ethernet-based Subscriber Access Networks

IEEE Communications Society

Developed by the
Standards Development Board

IEEE Std 1904.2™-2021

STANDARDS

IEEE Standard for Control and Management of Virtual Links in Ethernet-based Subscriber Access Networks

Developed by the

**Standards Development Board
of the
IEEE Communications Society**

Approved 16 June 2021

IEEE SA Standards Board

Abstract: Control mechanisms and management methods necessary to enable creation and operation of virtual links in Ethernet-based subscriber access networks are described. The key characteristics of the Virtual Link Control (VLC) mechanisms are:

- The ability to transit Layer 2 bridges in a single IEEE 802 Media Access Control (MAC) domain to allow remote device management
- Extensibility to accommodate new protocols and new types of devices
- The ability to simultaneously send messages to multiple VLC-aware stations using broadcast or multicast addressing

The message format, as well as processing operations at the stations participating in the VLC protocol, are described.

Keywords: IEEE 1904.2™, organization specific extensions, remote station management, Virtual Link Control (VLC) protocol, VLC Protocol Data Unit (VLC PDU)

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2021 by The Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 13 July 2021. Printed in the United States of America.

IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.

PDF: 978-1-5044-7710-9 STD24800
Print: 978-1-5044-7711-6 STDPD24800

IEEE prohibits discrimination, harassment, and bullying.

For more information, visit <https://www.ieee.org/about/corporate/governance/p9-26.html>.

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

Important Notices and Disclaimers Concerning IEEE Standards Documents

IEEE Standards documents are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page (<https://standards.ieee.org/ipr/disclaimers.html>), appear in all standards and may be found under the heading “Important Notices and Disclaimers Concerning IEEE Standards Documents.”

Notice and Disclaimer of Liability Concerning the Use of IEEE Standards Documents

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE SA) Standards Board. IEEE develops its standards through an accredited consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE Standards are documents developed by volunteers with scientific, academic, and industry-based expertise in technical working groups. Volunteers are not necessarily members of IEEE or IEEE SA, and participate without compensation from IEEE. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

IEEE does not warrant or represent the accuracy or completeness of the material contained in its standards, and expressly disclaims all warranties (express, implied and statutory) not included in this or any other document relating to the standard, including, but not limited to, the warranties of: merchantability; fitness for a particular purpose; non-infringement; and quality, accuracy, effectiveness, currency, or completeness of material. In addition, IEEE disclaims any and all conditions relating to results and workmanlike effort. In addition, IEEE does not warrant or represent that the use of the material contained in its standards is free from patent infringement. IEEE Standards documents are supplied “AS IS” and “WITH ALL FAULTS.”

Use of an IEEE standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity, nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

IN NO EVENT SHALL IEEE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

Translations

The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE is the approved IEEE standard.

Official statements

A statement, written or oral, that is not processed in accordance with the IEEE SA Standards Board Operations Manual shall not be considered or inferred to be the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that the presenter's views should be considered the personal views of that individual rather than the formal position of IEEE, IEEE SA, the Standards Committee, or the Working Group.

Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE or IEEE SA. However, **IEEE does not provide interpretations, consulting information, or advice pertaining to IEEE Standards documents.**

Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its Societies and Standards Coordinating Committees are not able to provide an instant response to comments, or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in evaluating comments or in revisions to an IEEE standard is welcome to join the relevant IEEE working group. You can indicate interest in a working group using the Interests tab in the Manage Profile & Interests area of the [IEEE SA myProject system](#). An IEEE Account is needed to access the application.

Comments on standards should be submitted using the [Contact Us](#) form.

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not constitute compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Data privacy

Users of IEEE Standards documents should evaluate the standards for considerations of data privacy and data ownership in the context of assessing and using the standards in compliance with applicable laws and regulations.

Copyrights

IEEE draft and approved standards are copyrighted by IEEE under US and international copyright laws. They are made available by IEEE and are adopted for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making these documents available for use and adoption by public authorities and private users, IEEE does not waive any rights in copyright to the documents.

Photocopies

Subject to payment of the appropriate licensing fees, IEEE will grant users a limited, non-exclusive license to photocopy portions of any individual standard for company or organizational internal use or individual, non-commercial use only. To arrange for payment of licensing fees, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400; <https://www.copyright.com/>. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Updating of IEEE Standards documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect.

Every IEEE standard is subjected to review at least every 10 years. When a document is more than 10 years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit [IEEE Xplore](#) or [contact IEEE](#). For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website.

Errata

Errata, if any, for all IEEE standards can be accessed on the [IEEE SA Website](#). Search for standard number and year of approval to access the web page of the published standard. Errata links are located under the Additional Resources Details section. Errata are also available in [IEEE Xplore](#). Users are encouraged to periodically check for errata.

Patents

IEEE Standards are developed in compliance with the [IEEE SA Patent Policy](#).

IMPORTANT NOTICE

IEEE Standards do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other devices or networks. IEEE Standards development activities consider research and information presented to the standards development group in developing any safety recommendations. Other information about safety practices, changes in technology or technology implementation, or impact by peripheral systems also may be pertinent to safety considerations during implementation of the standard. Implementers and users of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

Participants

At the time this draft standard was submitted to the IEEE-SA Standards Board for approval, the IEEE 1904.2 Working Group had the following membership:

Glen Kramer, *Working Group Chair*
Kevin Noll, *Working Group Vice-Chair*
Zhen Zhou, *Executive Secretary*
Marek Hajduczenia, *Chief Editor*
Zhen Zhou, *Treasurer*

Kevin Noll, *Task Force 2 Chair*

Steve Burroughs
Curtis Knittle

Pradeep Kondamuri

Jean-Christophe Marion
William Powell

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Marek Hajduczenia
Werner Hoelzl
Piotr Karocki
Stuart Kerry
Yongbum Kim
Curtis Knittle

Glen Kramer
Mark Laubach
Haiying Lu
Jean-Christophe Marion
Rajesh Murthy
Kevin Noll
William Powell

R. K. Rannow
Maximilian Riegel
John Vergis
Karl Weber
Yu Yuan
Zhen Zhou

When the IEEE-SA Standards Board approved this standard on 16 June 2021, it had the following membership:

Gary Hoffman, *Chair*
Jon Walter Rosdahl, *Vice Chair*
John D. Kulick, *Past Chair*
Konstantinos Karachalios, *Secretary*

Edward A. Addy
Doug Edwards
Ramy Ahmed Fathy
J. Travis Griffith
Thomas Koshy
Joseph L. Koepfinger*
David J. Law
Howard Li

Daozhuang Lin
Kevin Lu
Daleep C. Mohla
Chenhui Niu
Damir Novosel
Annette Reilly
Jon Walter Rosdahl

Dorothy Stanley
Mehmet Ulema
Lei Wang
F. Keith Waters
Karl Weber
Sha Wei
Howard Wolfman
Daidi Zhong

*Member Emeritus

Contents

Notice and Disclaimer of Liability Concerning the Use of IEEE Standards Documents	3
Translations	3
Official statements.....	4
Comments on standards	4
Laws and regulations	4
Data privacy.....	4
Copyrights.....	4
Photocopies	5
Updating of IEEE Standards documents	5
Errata	5
Patents	5
IMPORTANT NOTICE	5
1 OVERVIEW	13
1.1 Scope	13
1.2 Coverage.....	13
1.3 Overview of clauses	14
1.4 Word usage.....	14
2 NORMATIVE REFERENCES	15
3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS.....	16
3.1 Definitions	16
3.2 Acronyms and abbreviations	16
3.3 Special terms	17
3.4 Notation for state diagrams.....	17
3.4.1 General conventions	17
3.4.1.1 Representation of states.....	17
3.4.1.2 Transitions.....	17

3.4.2	State diagrams and accompanying text	18
3.4.3	Actions inside state blocks	18
3.4.4	State diagram variables	18
3.4.5	Operators	18
3.4.6	Timers.....	19
3.4.7	Hexadecimal notation	19
3.4.8	Binary notation.....	20
3.5	Notation for PICS	20
3.5.1	Abbreviations and special symbols	20
3.5.2	Instructions for completing the PICS proforma.....	20
3.5.3	Additional Information	21
3.5.4	Exception Information.....	21
3.5.5	Conditional items	21
4	VIRTUAL LINK CONTROL (VLC) OVERVIEW AND ARCHITECTURE.....	23
4.1	Principles of operation.....	23
4.1.1	VLC discovery protocol.....	23
4.2	VLC sublayer	24
4.3	VLC service interfaces	24
4.3.1	Definitions of VLC primitives	24
4.3.1.1	<i>MACCSI:MA_DATA</i> primitives	25
4.3.1.1.1	<i>MACCSI:MA_DATA.request</i>	26
4.3.1.1.2	<i>MACCSI:MA_DATA.indication</i>	26
4.3.1.2	<i>VLCSI:MA_DATA</i> primitives.....	26
4.3.1.2.1	<i>VLCSI:MA_DATA.request</i>	26
4.3.1.2.2	<i>VLCSI:MA_DATA.indication</i>	26
4.3.1.3	<i>VLCSI:VLC PDU</i> primitives.....	26
4.3.1.3.1	<i>VLCSI:VLC PDU.request</i>	26
4.3.1.3.1.1	Function	26
4.3.1.3.1.2	Semantics of the service primitive	27
4.3.1.3.1.3	When generated	27
4.3.1.3.1.4	Effect of receipt.....	27
4.3.1.3.2	<i>VLCSI:VLC PDU.indication</i>	27
4.3.1.3.2.1	Function	27
4.3.1.3.2.2	Semantics of the service primitive	27
4.3.1.3.2.3	When generated	27
4.3.1.3.2.4	Effect of receipt.....	27
4.3.1.4	<i>VLCSI:OMCI</i> primitives	28
4.3.1.4.1	<i>VLCSI:OMCI.request</i>	28
4.3.1.4.1.1	Function	28

4.3.1.4.1.2	Semantics of the service primitive	28
4.3.1.4.1.3	When generated	28
4.3.1.4.1.4	Effect of receipt.....	28
4.3.1.4.2	VLCSI:OMCI.indication	28
4.3.1.4.2.1	Function	28
4.3.1.4.2.2	Semantics of the service primitive	29
4.3.1.4.2.3	When generated	29
4.3.1.4.2.4	Effect of receipt.....	29
5	VIRTUAL LINK CONTROL PROTOCOL DATA UNITS (VLCPDUS).....	30
5.1	VLCPDU structure	30
5.2	VLCPDU Subtype encoding.....	31
5.2.1	VLC configuration subtype	32
5.2.2	OAM subtype.....	32
5.2.3	OMCI subtype.....	32
5.2.4	L2 subtype	33
5.2.5	EPD subtype.....	34
5.2.6	Organization-specific extension subtypes	35
5.3	VLAN-tagged VLCPDU	37
6	VLC SUBLAYER	38
6.1	VLC Functional Block Diagram.....	38
6.2	VLC Classification and Translation Engine	38
6.2.1	CTE rule structure.....	39
6.2.1.1	CTE rule classification conditions.....	39
6.2.1.1.1	Comparison operators	40
6.2.1.1.2	Classification fields.....	40
6.2.1.2	CTE rule modification actions	42
6.2.2	CTE rule categories.....	43
6.2.3	CTE rules involving operations on the VLAN tags.....	44
6.3	Receive path specification	44
6.3.1	Principles of operation	44
6.3.2	Constants.....	45
6.3.3	Variables	46
6.3.4	Functions	47
6.3.5	Primitives.....	47
6.3.6	State diagram	47
6.4	Transmit path specification.....	49
6.4.1	Principles of operation	49

6.4.2	Constants	49
6.4.3	Variables	49
6.4.4	Functions	50
6.4.5	Primitives	51
6.4.6	State diagram	51
7	PROTOCOL-SPECIFIC BEHAVIOR	53
7.1	Introduction.....	53
7.2	Support for OAM remote loopback.....	53
7.2.1	Overview	53
7.2.2	OAM loopback over VLC tunnel	53
8	VLC MANAGEMENT	58
8.1	VLC configuration	58
8.1.1	Configuration VLCPDU	58
8.1.2	Rule identification	60
8.1.3	<i>CTE Rule</i> TLV structure	60
8.1.4	VLC configuration protocol.....	62
8.1.4.1	Request and Response for the ‘Query all rules’ operation	62
8.1.4.2	Request and Response for the ‘Add a rule’ operation.....	63
8.1.4.3	Request and Response for the ‘Remove a rule’ operation	64
8.2	Management attributes.....	65
8.2.1	Introduction.....	65
8.2.2	VLC management using extended OAM.....	65
8.2.3	Branch 0xD6— <i>CTE Table Context</i> TLV	66
8.2.4	Branch 0xA7—VLC Service Interface counters	66
8.2.4.1	Attribute <i>aVlcFramesSiVlcpdu</i> (0xA7/0x00-00)	67
8.2.4.2	Attribute <i>aVlcFramesSiMaData</i> (0xA7/0x00-01)	68
8.2.4.3	Attribute <i>aVlcFramesSiOmci</i> (0xA7/0x00-02)	68
8.2.4.4	Attribute <i>aVlcOctetsSiVlcpdu</i> (0xA7/0x80-00)	69
8.2.4.5	Attribute <i>aVlcOctetsSiMaData</i> (0xA7/0x80-01).....	69
8.2.4.6	Attribute <i>aVlcOctetsSiOmci</i> (0xA7/0x80-02).....	70
8.2.5	Branch 0xA8—VLC Rule Counters	71
8.2.5.1	Attribute <i>aVlcFramesUnmatched</i> (0xA8/0x00-00)	71
8.2.5.2	Attribute <i>aVlcFramesMatchedByRuleN</i> (0xA8/0x00-01 to 0xA8/0x7F-FF)	72
8.2.5.3	Attribute <i>aVlcOctetsUnmatched</i> (0xA8/0x80-00).....	73
8.2.5.4	Attribute <i>aVlcOctetsMatchedByRuleN</i> (0xA8/0x80-01 to 0xA8/0xFF-FF).....	74

9 PROTOCOL IMPLEMENTATION CONFORMANCE STATEMENT (PICS) PROFORMA FOR VIRTUAL LINK CONTROL (VLC) SPECIFICATION..... 75

9.1	Introduction.....	75
9.2	Implementation identification	75
9.3	Protocol summary	75
9.4	VLCPDU encoding	76
9.5	VLC configuration protocol	77
9.6	Receive process	78
9.7	Transmit process.....	78
9.8	CTE.....	78
9.9	Management attributes.....	78

ANNEX 8A (INFORMATIVE) VIRTUAL LINK CONTROL (VLC) CONFIGURATION EXAMPLES 81

8A.1	Operations, Administration, and Maintenance (OAM) over VLC use case, VLC-unaware end points.....	81
8A.1.1	Introduction.....	81
8A.1.2	VLC provisioning to establish tunnels.....	82
8A.1.2.1	Addition of tunnel entrance rule at the ingress of Bridge X, port 3.....	82
8A.1.2.2	Addition of tunnel exit rule at the egress of Bridge Y, port 0	84
8A.1.2.3	Addition of VLC tunnel entrance rule at the ingress of Bridge Y, port 0	86
8A.1.2.4	Addition of VLC tunnel exit rule at the egress of Bridge X, port 3	88
8A.2	OAM over VLC use case, VLC-aware end points	90
8A.2.1	Introduction.....	90
8A.2.2	VLC provisioning to establish tunnels.....	91
8A.2.2.1	Addition of tunnel entrance rule at the egress of Manager M.....	92
8A.2.2.2	Addition of VLC tunnel entrance rule at the egress of Station S	94
8A.3	OAM over VLC use case, VLC-aware end point and VLC-unaware end point	96
8A.3.1	Introduction.....	96
8A.3.2	VLC provisioning to establish tunnels.....	97
8A.3.2.1	Addition of tunnel entrance rule at the egress of Manager M.....	98
8A.3.2.2	Addition of tunnel exit rule at the egress of Bridge Y, port 0	98
8A.3.2.3	Addition of VLC tunnel entrance rule at the ingress of Bridge Y, port 0	98
8A.4	Remote ITU-T PON Management over VLC use case	98
8A.4.1	Introduction.....	98
8A.4.2	VLC provisioning to establish tunnels.....	98

8A.4.2.1	Addition of tunnel entrance rule at the egress of Manager for OLT OAM messages....	100
8A.4.2.2	Addition of tunnel entrance rule at the egress of Manager for ONU OMCI messages .	101
8A.4.2.3	Addition of tunnel entrance rule at the egress of OLT for OLT OAM messages	101
8A.4.2.4	Addition of tunnel entrance rule at the egress of OLT for ONU OMCI messages.....	102
8A.5	Remote IEEE EPON Management over VLC use case	102
8A.5.1	Introduction.....	102
8A.5.2	VLC provisioning to establish tunnels.....	104
8A.5.2.1	Rules for the forward VLC tunnel for the management of the OLT	105
8A.5.2.2	Rules for the reverse VLC tunnel for the management of the OLT.....	106
8A.5.2.3	Rules for the forward VLC tunnel for the management of a VLC-unaware ONU	106
8A.5.2.4	Rules for the reverse VLC tunnel for the management of a VLC-unaware ONU	107
8A.5.2.5	Rules for the forward VLC tunnel for the management of a VLC-aware ONU.....	108
8A.5.2.6	Rules for the reverse VLC tunnel for the management of a VLC-aware ONU	109
8A.5.3	Scalable management of an EPON using OAM over VLC	109

IEEE Standard for Control and Management of Virtual Links in Ethernet-based Subscriber Access Networks

1 Overview

1.1 Scope

This standard describes control mechanisms and management methods necessary to enable creation and operation of virtual links in Ethernet-based subscriber access networks. The key characteristics of the Virtual Link Control (VLC) mechanisms are:

- The ability to transit Layer 2 bridges in a single IEEE 802 Media Access Control (MAC) domain to allow remote device management;
- Extensibility to accommodate new protocols and new types of devices;
- The ability to simultaneously send messages to multiple VLC-aware stations using broadcast or multicast addressing.

The standard describes the message format, as well as processing operations at the stations participating in the VLC protocol.

1.2 Coverage

In their quest to find the optimal balance between the performance of subscriber access networks and their cost, network operators increasingly combine optical distribution section with a copper-based drop section, which typically includes a twisted pair, a Category-5 cable, or a coaxial cable. Network operators require a management system that would allow them to efficiently access and manage the subscriber demarcation device as well as the various devices that interconnect their optical and copper sections of the network.

In addition, to improve service quality, access network operators find it necessary to extend their management domains past the typical subscriber demarcation device, such as an Optical Network Unit (ONU), a Coaxial Network Unit (CNU), Cable or DSL modem, or a Residential Gateway (RGW).

Often certain tunneling techniques become necessary in order to continue using the existing management protocols over the extended management domains. This standard defines the concept of virtual links to allow IEEE 802.3 OAM, ITU-T G.988 OMCI, or other management protocols to be extended over layer-2

bridged networks. This specification provides network operators with uniform, consistent, and interoperable methods to create, control, and manage virtual links in subscriber access networks. The new functionality is confined to the Virtual Link Control (VLC) sublayer of the Data Link layer.

1.3 Overview of clauses

This subclause provides an overview of the scope of individual clauses included in this specification, namely:

- Clause 1 provides an overview of the IEEE 1904.2 specification, including the scope and purpose of the specification and the scope of individual clauses.
- Clause 2 lists normative references used within this specification.
- Clause 3 presents definitions of specific terms as used in this standard. Terms may be introduced in this specification or may exist with multiple industry definitions. Additionally, a list of acronyms used in this standard is included.
- Clause 4 defines the overall architecture of VLC and defines the service interfaces of VLC.
- Clause 5 defines the structure of a VLC PDU.
- Clause 6 defines the operation of the VLS sublayer.
- Clause 7 defines behavior of certain protocols when operating over VLC.
- Clause 8 defines VLC management, including the VLC configuration protocol.
- Clause 9 is the PICS Proforma for VLC.
- Annex 8A is an informative collection of VLC use cases.

1.4 Word usage

The word *shall* indicates mandatory requirements strictly to be followed in order to conform to the standard and from which no deviation is permitted (shall equals is required to).^{1, 2}

The word *should* indicates that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required (should equals is recommended that).

The word *may* is used to indicate a course of action permissible within the limits of the standard (may equals is permitted to).

The word *can* is used for statements of possibility and capability, whether material, physical, or causal (can equals is able to).

¹ The word *must* is deprecated and cannot be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.

² The word *will* is deprecated and cannot be used when stating mandatory requirements; *will* is only used in statements of fact.

2 Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

IEEE Std 802.3™-2018, IEEE Standard for Ethernet.^{3, 4}

IEEE Std 802.1Q™-2018, IEEE Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks.

IEEE Std 1904.1™-2017, IEEE Standard for Service Interoperability in Ethernet Passive Optical Networks (SIEPON).

ITU-T Recommendation G.988, ONU management and control interface (OMCI) specification.⁵

³ The IEEE standards or products referred to in Clause 2 are trademarks owned by The Institute of Electrical and Electronics Engineers, Incorporated.

⁴ IEEE publications are available from The Institute of Electrical and Electronics Engineers (<https://standards.ieee.org/>).

⁵ ITU-T publications are available from the International Telecommunications Union (<https://www.itu.int/>).

3 Definitions, acronyms, and abbreviations

3.1 Definitions

For the purposes of this document, the following terms and definitions apply. The *IEEE Standards Dictionary Online* should be consulted for terms not defined in this clause.⁶

end station: A functional unit in an IEEE 802® network that acts as a source of, and/or destination for, link-layer data traffic carried on the network.

ITU-T PON: ITU-T PON architecture as defined in ITU-T G.98x series of recommendations.

Network Management System (NMS): In the scope of IEEE Std 1904.2, any network management, control, information storage, and other type of entities, located in the same or different geographical locations, functionally combined to a single point of reference. This entity is responsible for controlling, managing, and supervising the operation of a Virtual Link Control (VLC)-aware L2 network. NMS combines, terminates, proxies, or snoops a number of different control and management protocols (outside the scope of this standard), providing Fault, Configuration, Accounting, Performance, Security (FCAPS) management functionality for a network operator.

3.2 Acronyms and abbreviations

CTE	Classification and Translation Engine
FCAPS	Fault, Configuration, Accounting, Performance, Security
MAC	Media Access Control
NMS	Network Management System
OAM	Operations, Administration, and Maintenance
OLT	Optical Line Terminal
OMCI	ONU Management Control Interface
ONU	Optical Network Unit
PDU	Protocol Data Unit
VLC	Virtual Link Control

⁶ IEEE Standards Dictionary Online subscription is available at
http://www.ieee.org/portal/innovate/products/standard/standards_dictionary.html.

3.3 Special terms

Term: Definition

3.4 Notation for state diagrams

All the state diagrams used in this standard meet the set of requirements included in the following subclauses.

3.4.1 General conventions

The operation of any protocol defined in this standard can be described by subdividing the protocol into a number of interrelated functions. The operation of the functions can be described by state diagrams. Each diagram represents the domain of a function and consists of a group of connected, mutually exclusive states. Only one state of a function is active at any given time (see Figure 3-1).

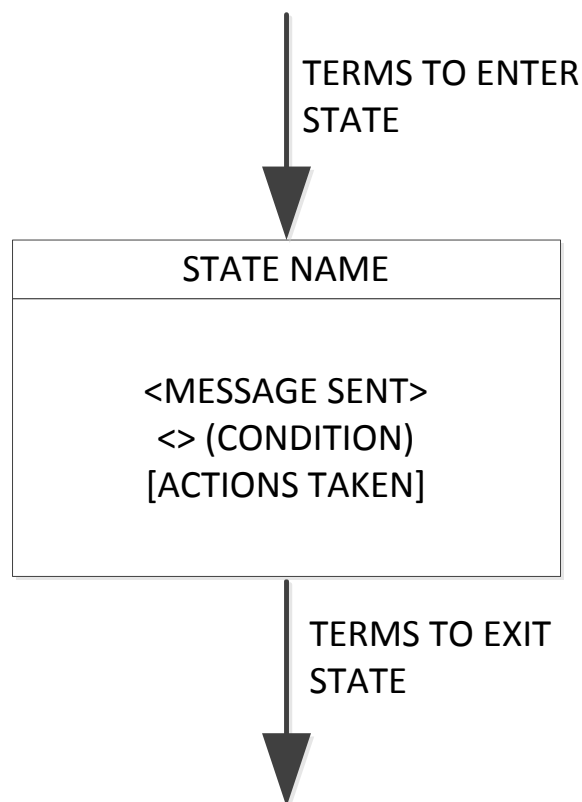


Figure 3-1—State diagram notation example

3.4.1.1 Representation of states

Each state that the function can assume is represented by a rectangle. These are divided into two parts by a horizontal line. In the upper part, the state is identified by a name in capital letters. The lower part contains the body of the given state, containing description of the actions taken in this state as defined in 3.4.3.

3.4.1.2 Transitions

All permissible transitions between the states of a function are represented graphically by arrows between them. A transition that is global in nature (for example, an exit condition from all states to the IDLE or

RESET state) is indicated by an open arrow (an arrow with no source block). Global transitions are evaluated continuously whenever any state is evaluating its exit conditions. When the condition for a global transition becomes true, it supersedes all other transitions, including Unconditional Transition (UCT), returning control to the block pointed to by the open arrow.

Labels on transitions are qualifiers that are required to be fulfilled before the transition is taken. The label UCT designates an unconditional transition. Qualifiers described by short phrases are enclosed in parentheses.

The following terms are valid transition qualifiers:

- Boolean expressions
- An event such as the expiration of a timer: `timer_done`
- An event such as the reception of a message: `MA_DATA.indication`
- An unconditional transition: UCT
- A branch taken when other exit conditions are not satisfied: ELSE

State transitions occur instantaneously. No transition in the state diagram can cross another transition. When possible, any two transitions with different logical conditions are not joined together into a single transition line.

3.4.2 State diagrams and accompanying text

State diagrams take precedence over text.

3.4.3 Actions inside state blocks

The actions inside a state block execute instantaneously. Actions inside state blocks are atomic (i.e., uninterruptible).

After performing all the actions listed in a state block one time, the state diagram then continuously evaluates exit conditions for the given state block until one is satisfied, at which point control passes through a transition arrow to the next block. While the state awaits fulfillment of one of its exit conditions, the actions inside do not implicitly repeat.

Valid state actions may include generation of `indication` and `request` primitives.

No actions are taken outside of any blocks of the state diagram.

3.4.4 State diagram variables

Once set, variables retain their values as long as succeeding blocks contain no references to them.

Setting the parameter of a formal interface message assures that, on the next transmission of that message, the last parameter value set is transmitted.

Testing the parameter of a formal interface message tests the value of that message parameter that was received on the last transmission of said message. Message parameters may be assigned default values that persist until the first reception of the relevant message.

3.4.5 Operators

The state diagram operators are shown in Table 3-1.

Table 3-1—State diagram operators

Character	Meaning
AND	Boolean AND
OR	Boolean OR
XOR	Boolean XOR
!	Boolean NOT
<	Less than
>	More than
≤	Less than or equal to
≥	More than or equal to
==	Equals (a test of equality)
!=	Not equals
()	Indicates precedence
=	Assignment operator
	Concatenation operation that combines several sub-fields or parameters into a single aggregated field or parameter
else	No other state condition is satisfied
true	Designation of a Boolean value of TRUE
false	Designation of a Boolean value of FALSE

3.4.6 Timers

Some of the state diagrams use timers for various purposes, e.g., measurement of time, and confirmation of activity. All timers operate in the same fashion.

A timer is reset and starts counting upon entering a state where [start x_timer, x_timer_value] is asserted. Time “x” after the timer has been started, “x_timer_done” is asserted and remains asserted until the timer is reset. At all other times, “x_timer_not_done” is asserted.

When entering a state where [start x_timer, x_timer_value] is asserted, the timer is reset and restarted even if the entered state is the same as the exited state.

Any timer can be stopped at any time upon entering a state where [stop x_timer] is asserted, which aborts the operation of the “x_timer” asserting “x_timer_not_done” indication until the timer is restarted again.

3.4.7 Hexadecimal notation

Numerical values designated by the 0x prefix indicate a hexadecimal notation of the corresponding number, with the least significant bit shown on the right. For example: 0x0F represents an 8-bit hexadecimal value of the decimal number 15; 0x00-00-00-00 represents a 32-bit hexadecimal value of the decimal number 0; 0x11-AB-11-AB represents a 32-bit hexadecimal value of the decimal number 296423851.⁷

⁷ MAC addresses shown in this standard also use the prefix 0x, different from the format used in IEEE Std 802.

3.4.8 Binary notation

Numerical values designated by the 0b prefix indicate a binary notation of the corresponding number, with the least significant bit shown on the right. For example: 0b00001000 represents an 8-bit binary value of the decimal number 8.

3.5 Notation for PICS

The supplier of a device implementation that is claimed to conform to this standard is required to complete a protocol implementation conformance statement (PICS) proforma.

A completed PICS proforma is the PICS for the implementation in question. The PICS is a statement of which capabilities and options of this standard have been implemented. The PICS can be used for a variety of purposes by various parties, including the following:

- a) As a checklist by the protocol implementer, to reduce the risk of failure to conform to the standard through oversight
- b) As a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma, by the supplier and acquirer, or potential acquirer, of the implementation
- c) As a basis for initially checking the possibility of interworking with another implementation by the user, or potential user, of the implementation (note that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICS)
- d) As the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation, by a protocol tester

Each PICS entry is uniquely identified by an item number, comprising a three-letter mnemonic representing a functional area, followed by a sequential number.

3.5.1 Abbreviations and special symbols

The following symbols are used in the PICS proforma:

M	mandatory field/function
!	negation
O	optional field/function
O.<n>	optional field/function, but at least one of the group of options labeled by the same numeral <n> is required
O/<n>	optional field/function, but one and only one of the group of options labeled by the same numeral <n> is required
X	prohibited field/function
<item>:	simple-predicate condition, dependent on the support marked for <item>
<item1>*<item2>:	AND-predicate condition, the requirement needs to be met if both optional items are implemented

3.5.2 Instructions for completing the PICS proforma

The first part of the PICS proforma, Implementation Identification and Protocol Summary, is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation.

The main part of the PICS proforma is a fixed-format questionnaire divided into subclauses, each containing a group of items. Answers to the questionnaire items are to be provided in the right-most

column, either by simply marking an answer to indicate a restricted choice (usually Yes, No, or Not Applicable), or by entering a value or a set or range of values. (Note that there are some items where two or more choices from a set of possible answers can apply; all relevant choices are to be marked.)

Each item is identified by an item reference in the first column; the second column contains the question to be answered; the third column contains the reference or references to the material that specifies the item in the main body of the standard; the fourth column contains values and/or comments pertaining to the question to be answered. The remaining columns record the status of the items—whether the support is mandatory, optional or conditional—and provide the space for the answers.

The supplier may also provide, or be required to provide, further information, categorized as either Additional Information or Exception Information. When present, each kind of further information is to be provided in a further subclause of items labeled A<i> or X<i>, respectively, for cross-referencing purposes, where <i> is any unambiguous identification for the item (e.g., simply a numeral); there are no other restrictions on its format or presentation.

A completed PICS proforma, including any Additional Information and Exception Information, is the protocol implementation conformance statement for the implementation in question.

Note that where an implementation is capable of being configured in more than one way, according to the items listed under Major Capabilities/Options, single PICS may be able to describe all such configurations. However, the supplier has the choice of providing more than one PICS, each covering some subset of the implementation's configuration capabilities, if that would make presentation of the information easier and clearer.

3.5.3 Additional Information

Items of Additional Information allow a supplier to provide further information intended to assist the interpretation of the PICS. It is not intended or expected that a large quantity be supplied, and the PICS can be considered complete without any such information. Examples might be an outline of the ways in which a (single) implementation can be set up to operate in a variety of environments and configurations; or a brief rationale, based perhaps upon specific application needs, for the exclusion of features that, although optional, are nonetheless commonly present in implementations.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception Information.

3.5.4 Exception Information

It may occasionally happen that a supplier wishes to answer an item with mandatory or prohibited status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No pre-printed answer is found in the Support column for this; instead, the supplier is required to write into the Support column an X<i> reference to an item of Exception Information, and to provide the appropriate rationale in the Exception item itself.

An implementation for which an Exception item is required in this way does not conform to this standard. Note that a possible reason for the situation described above is that a defect in the standard has been reported, a correction for which is expected to change the requirement not met by the implementation.

3.5.5 Conditional items

The PICS proforma may contain conditional items. These are items for which both the applicability of the item itself, and its status if it does apply—mandatory, optional, or prohibited—are dependent upon whether or not certain other items are supported.

Individual conditional items are indicated by a conditional symbol of the form “<item>:<s>” in the Status column, where “<item>” is an item reference that appears in the first column of the table for some other item, and “<s>” is a status symbol, M (Mandatory), O (Optional), or X (Not Applicable).

If the item referred to by the conditional symbol is marked as supported, then:

- a) The conditional item is applicable,
- b) Its status is given by “<s>”, and
- c) The support column is to be completed in the usual way

Each item whose reference is used in a conditional symbol is indicated by an asterisk in the Item column.

4 Virtual Link Control (VLC) overview and architecture

4.1 Principles of operation

Virtual Link Control (VLC) defines the method of encapsulating various Protocol Data Units (xPDUs) in Ethernet frames with VLC Ethertype (0xA8-C8⁸). An Ethernet frame with VLC Ethertype is called a *Virtual Link Control Protocol Data Unit (VLC PDU)*.

That portion of the network path that xPDUs traverse while they are encapsulated as VLC PDUs is referred to as a *VLC tunnel*. VLC tunnels are defined in one direction only (unidirectional). A bidirectional VLC tunnel comprises two unidirectional tunnels between the same two stations.

The xPDU-to-VLC PDU and VLC PDU-to-xPDU conversions take place within the VLC sublayer (see 4.2). Both VLC client and VLC sublayer are optional in an end station. In a multi-port device, the VLC sublayer and VLC client may be implemented in any subset of the device ports. Devices that implement the VLC sublayer in at least one of the ports are said to be VLC-aware.

Devices that do not implement the VLC sublayer in any of the ports are called *VLC-unaware*. VLC-unaware devices are able to relay VLC PDUs as generic Ethernet frames using existing L2 forwarding mechanisms but are unable to consume or generate VLC PDUs.

The VLC sublayer includes the Classification and Translation Engine (CTE) that converts xPDUs into VLC PDUs and vice versa. The CTE behavior is governed by a set of rules that are either statically configured or dynamically provisioned by the Network Management System (NMS) (see 6.1).

The VLC sublayer provides a service interface to the Operations, Administration, and Maintenance (OAM) sublayer and the VLC client, and may provide a service interface to other L2 protocol-specific clients. The only messages that are passed to and received from the VLC client are VLC configuration messages (see *VLC_CONFIG VLC PDU* in 8.1.1).

All VLC PDUs, except *VLC_CONFIG* VLC PDUs, carry tunneling payloads associated with specific protocols (xPDU). Any non-*VLC_CONFIG* VLC PDU that is consumed by a device is first converted into its native xPDU format and then passed to a specific client associated with that xPDU protocol type. Correspondingly, any non-*VLC_CONFIG* VLC PDU that is generated by a device originates in a protocol-specific client as an xPDU and is then converted into a VLC PDU within the VLC sublayer.

A device port where xPDUs are converted into VLC PDUs (within the VLC sublayer) is referred to as a *VLC entrance point* and a port where the opposite conversion takes place is referred to as a *VLC exit point*.

4.1.1 VLC discovery protocol

The tunnel entrance and exit points may be pre-configured or provisioned via *VLC_CONFIG* VLC PDUs based on known network topology and L2 device addresses. An automatic VLC discovery protocol is out-of-scope for this revision of the standard.

⁸ The current list of Ethertype assignments is available at <http://standards-oui.ieee.org/ethertype/eth.txt>.

4.2 VLC sublayer

VLC functionality is confined to the VLC sublayer. Figure 4-1 depicts architectural positioning of the VLC sublayer, which is a client of the Media Access Control (MAC) Control sublayer (see IEEE Std 802.3, Clause 31).⁹

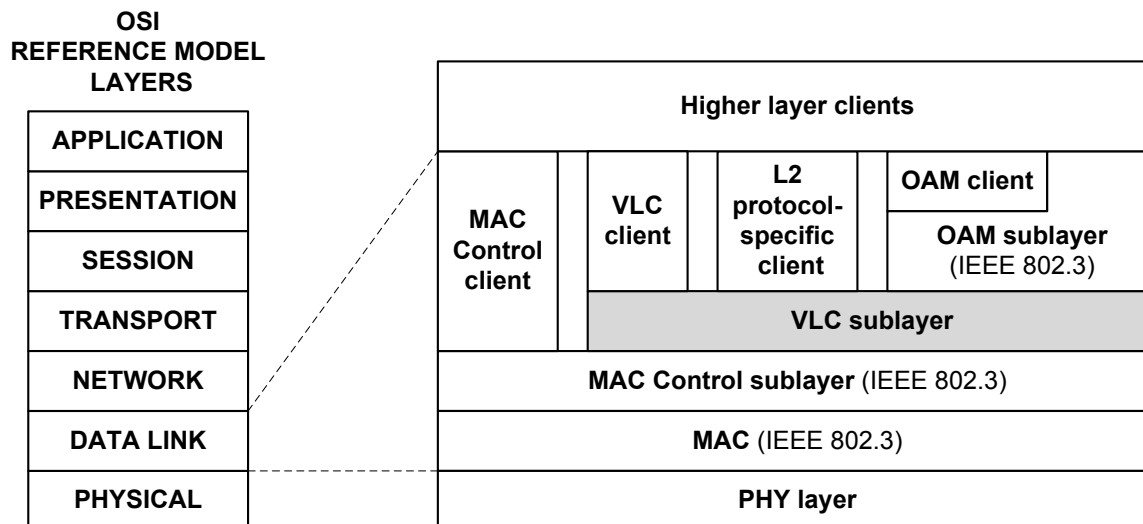


Figure 4-1—VLC sublayer relationship to the ISO/IEC Open Systems Interconnection (OSI) reference model and the IEEE Std 802.3 Ethernet model

4.3 VLC service interfaces

4.3.1 Definitions of VLC primitives

The VLC sublayer is a client of the MAC Control sublayer and implements a standard IEEE Std 802.3 MAC service interface (see IEEE Std 802.3, Clause 2).

The VLC sublayer provides a VLC service interface (VLCSI) to the OAM sublayer, the VLC client, and to other L2 protocol-specific clients (see Figure 4-2). To the OAM sublayer, the VLC sublayer presents a standard IEEE Std 802.3 MAC service interface (*VLCSI:MA_DATA*). To the VLC client, the VLC sublayer presents a VLC-specific service interface (*VLCSI:VLC_PDU*). To the L2 protocol-specific clients, the VLC sublayer presents a protocol-specific service interface. The only protocol-specific client defined in this standard is the Optical Network Unit (ONU) Management Control Interface (OMCI) client (see 5.2.3).

Inter-layer interfaces are depicted in Figure 4-2.

⁹ Information on references can be found in Clause 2.

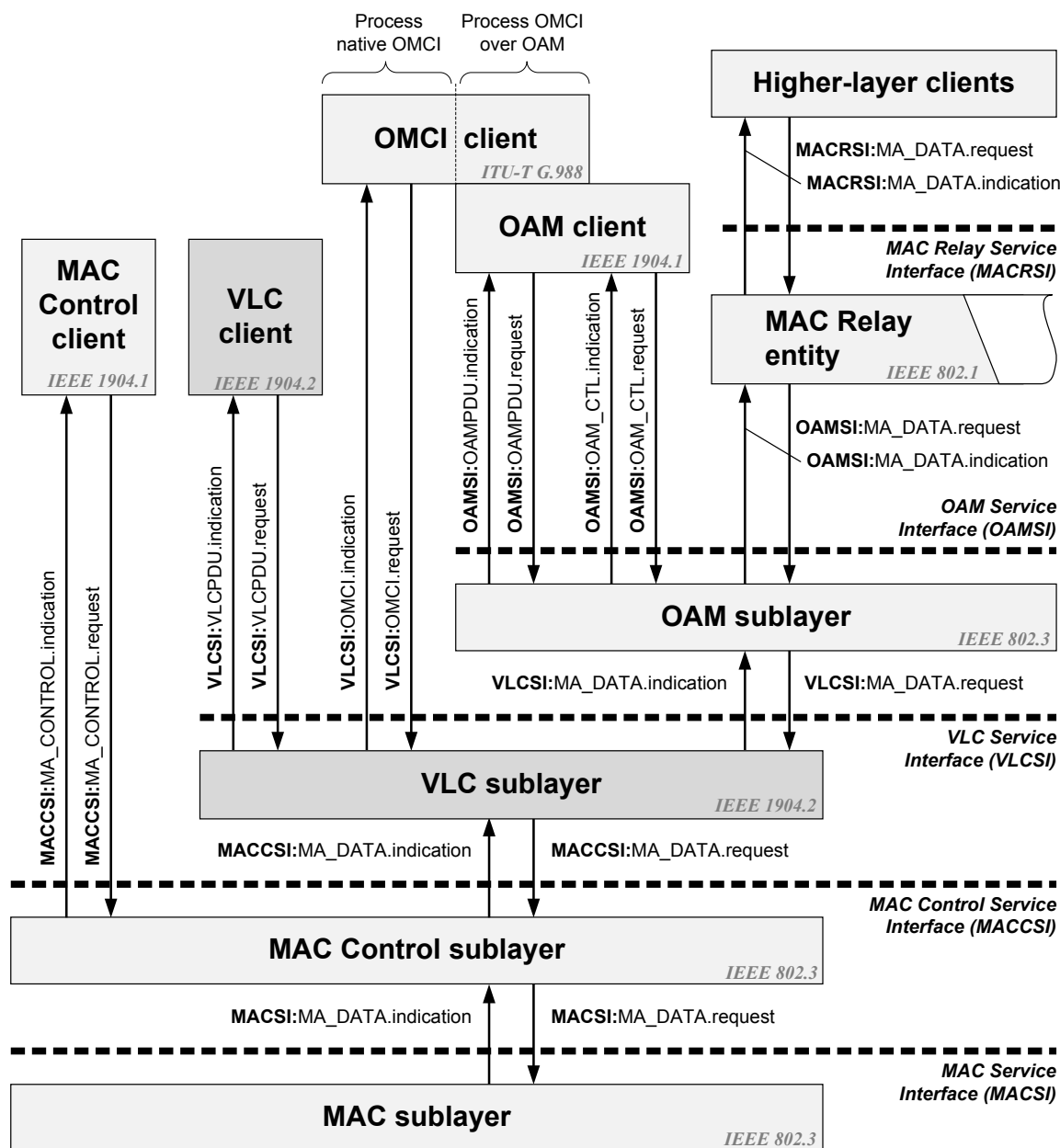


Figure 4-2—Positioning of VLC sublayer and service interfaces

4.3.1.1 MACCSI:MA_DATA primitives

The *MACCSI:MA_DATA* is a mandatory interface between the MAC Control sublayer and the VLC sublayer (see Figure 4-2). The MAC Control sublayer communicates with the VLC sublayer using the following service primitives:

- MACCSI:MA_DATA.request
- MACCSI:MA_DATA.indication

4.3.1.1.1 MACCSI:MA_DATA.request

The MACCSI:MA_DATA.indication service primitive is used to pass a properly-formed and valid MAC frame from the VLC sublayer to the MAC Control sublayer. This primitive is defined in IEEE Std 802.3, 2.3.1.

4.3.1.1.2 MACCSI:MA_DATA.indication

The MACCSI:MA_DATA.indication service primitive is used to pass a properly-formed and valid MAC frame from the VLC sublayer to the MAC Control sublayer. This primitive is defined in IEEE Std 802.3, 2.3.2.

4.3.1.2 VLCSI:MA_DATA primitives

The VLCSI:MA_DATA is an interface between the VLC sublayer and the OAM sublayer, if this sublayer is present (see Figure 4-2). Otherwise, the VLCSI:MA_DATA is an interface between the VLC sublayer and MAC client. The VLC sublayer communicates with the OAM sublayer or MAC client using the following service primitives:

- VLCSI:MA_DATA.request
- VLCSI:MA_DATA.indication

4.3.1.2.1 VLCSI:MA_DATA.request

The VLCSI:MA_DATA.indication service primitive is used to pass a properly-formed and valid MAC frame from the OAM sublayer or a MAC client to the VLC sublayer. This primitive is defined in IEEE Std 802.3, 2.3.1.

4.3.1.2.2 VLCSI:MA_DATA.indication

The VLCSI:MA_DATA.indication service primitive is used to pass a properly-formed and valid MAC frame from the VLC sublayer to the OAM sublayer or to a MAC client. This primitive is defined in IEEE Std 802.3, Clause 2.

4.3.1.3 VLCSI:VLC PDU primitives

The VLCSI:VLC PDU is a mandatory interface between the VLC sublayer and the VLC client (see Figure 4-2). The VLC sublayer communicates with the VLC client using the following service primitives:

- VLCSI:VLC PDU.request
- VLCSI:VLC PDU.indication

4.3.1.3.1 VLCSI:VLC PDU.request

4.3.1.3.1.1 Function

This primitive defines the transfer of control commands from a VLC client entity to the local VLC sublayer entity.

4.3.1.3.1.2 Semantics of the service primitive

The semantics of the primitive are as follows:

```
VLCSI:VLCPPDU.request (
    opcode,
    request_operand_list
)
```

The `opcode` specifies the control operation requested by the VLC client entity. The `request_operand_list` is an opcode-specific set of parameters. The only valid opcode value is `SUBTYPE_VLC` as defined in Table 5-1.

4.3.1.3.1.3 When generated

This primitive is generated by a VLC client whenever it wishes to use the services of the VLC sublayer entity.

4.3.1.3.1.4 Effect of receipt

Upon the receipt of this primitive, the Transmit Process of the VLC sublayer entity forms a complete VLCPPDU as described in 6.4. The VLCPPDU may be further modified by the egress CTE entity, before it is passed to MAC Control sublayer via the MACCSI:MA_DATA interface.

4.3.1.3.2 VLCSI:VLCPPDU.indication

4.3.1.3.2.1 Function

This primitive defines the transfer of control information from the VLC sublayer entity to the VLC client entity.

4.3.1.3.2.2 Semantics of the service primitive

The semantics of the primitive are as follows:

```
VLCSI:VLCPPDU.indication (
    opcode,
    indication_operand_list
)
```

The `opcode` specifies the control information being transferred to the VLC client entity. The `indication_operand_list` is an opcode-specific set of parameters. The only valid opcode value is `SUBTYPE_VLC` as defined in Table 5-1.

4.3.1.3.2.3 When generated

The `VLCSI:VLCPPDU.indication` is generated by the VLC sublayer when it receives from the `MACCSI:MA_DATA` interface a valid VLCPPDU with subtype equal to `SUBTYPE_VLC`.

4.3.1.3.2.4 Effect of receipt

The VLC client performs the requested operation, as encoded in the `indication_operand_list`. The format and encoding of the `indication_operand_list` is specified in 8.1, but the formal specification of VLC client operation is beyond the scope of this standard.

4.3.1.4 *VLCSI:OMCI* primitives

The OMCI client communicates with the VLC sublayer using the following service primitives:

- `VLCSI:OMCI.request`
- `VLCSI:OMCI.indication`

The *VLCSI:OMCI* interface (see Figure 4-2) is optional, but if it is implemented, the `VLCSI:OMCI.request` and `VLCSI:OMCI.indication` service primitives described in this subclause need to be supported.

4.3.1.4.1 *VLCSI:OMCI.request*

4.3.1.4.1.1 Function

This primitive defines the transfer of data from the OMCI client entity to the VLC sublayer. This primitive is only relevant in the egress direction.

4.3.1.4.1.2 Semantics of the service primitive

The semantics of the primitive are as follows:

```
VLCSI:OMCI.request (
    omci_serial_number,
    omci_frame_sdu
)
```

The `omci_serial_number` parameter specifies the 8-octet serial number, defined in ITU-T Rec G.988, 9.1.1, assigned to the ONU that is the intended destination of this OMCI frame.

The `omci_frame_sdu` parameter contains the OMCI frame (according to ITU-T Rec G.988) that is related to the ONU identified by the `omci_serial_number`.

4.3.1.4.1.3 When generated

This primitive is generated by the OMCI client entity whenever an OMCI frame is to be transferred to a peer entity.

4.3.1.4.1.4 Effect of receipt

The receipt of this primitive causes the VLC sublayer to apply the rules installed in the egress CTE instance to perform any required parsing and transformations of the request parameters necessary to encapsulate and transmit the OMCI frame as a VLCPDU. After performing these actions, the VLC sublayer asserts the `MACCSI:MA_DATA.request` primitive according to the procedures described in 4.3.1.1.1.

4.3.1.4.2 *VLCSI:OMCI.indication*

4.3.1.4.2.1 Function

This primitive defines the transfer of data from the VLC sublayer to the OMCI client entity. This primitive is only relevant in the ingress direction.

4.3.1.4.2.2 Semantics of the service primitive

The semantics of the primitive are as follows:

```
VLCSI:OMCI.indication (  
    omci_serial_number,  
    omci_frame_sdu  
)
```

The `omci_serial_number` parameter specifies the 8-octet serial number, defined in ITU-T Rec G.988, 9.1.1, assigned to the ONU that is the intended destination of this OMCI frame.

The `omci_frame_sdu` parameter contains the OMCI frame (according to ITU-T Rec G.988) that is related to the ONU identified by the `omci_serial_number`.

4.3.1.4.2.3 When generated

This primitive is passed from the VLC sublayer entity to the OMCI client entity to indicate the arrival of a VLCPDU to the local VLC sublayer entity that is destined for the OMCI client. Such VLCPDUs are reported only if they are validly formed and received without error.

4.3.1.4.2.4 Effect of receipt

The receipt of this primitive causes the OMCI client to process the received parameters. The specific actions to be taken by the OMCI client are out of the scope of this standard.

5 Virtual Link Control Protocol Data Units (VLCPDUs)

5.1 VLCPDU structure

A Virtual Link Control Protocol Data Unit (VLCPDU) is an Ethernet MAC frame with the value of `LengthType` field equal to the VLC Ethertype (0xA8-C8). The Ethernet MAC frame format is shown in IEEE Std 802.3, Clause 3.

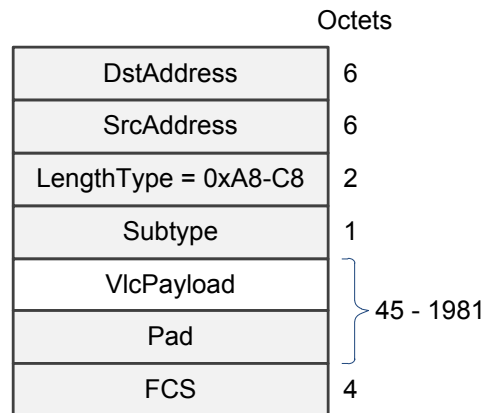


Figure 5-1—Generic VLCPDU format

The VLCPDU structure is shown in Figure 5-1 and it includes the following fields:

— DstAddress:

In a VLCPDU, the `DstAddress` is the MAC address associated with the device consuming the xPDU carried within the VLCPDU.

NOTE—The station identified by `DstAddress` might not be VLC-aware, in which case the VLC tunnel is expected to be terminated before the VLCPDU reaches that station.¹⁰

— SrcAddress:

In VLCPDUs, the `SrcAddress` is the individual MAC address associated with the device that generated xPDU.

NOTE—The station identified by `SrcAddress` might not be VLC-aware, in which case the VLC tunnel is expected to be initiated after the xPDU leaves that station.

— LengthType:

The `LengthType` field in a VLCPDU carries the VLC Ethertype value 0xA8-C8.

— Subtype:

The `Subtype` field identifies the type of xPDU being encapsulated in the VLCPDU. `Subtype` field values are defined in Table 5-1.

¹⁰ Notes in text, tables, and figures of a standard are given for information only and do not contain requirements needed to implement this standard.

— **VlcPayload:**

The **VlcPayload** field represents a set of fields associated with the subtype-specific protocols as defined in 5.2.

— **Pad:**

The **Pad** field is added to bring the VLCPDU length up to the minimum frame size (see IEEE Std 802.3, 4A.2.3.2.4). This field is filled with zeros on transmission and is ignored on reception.

— **FCS:**

This field contains the Frame Check Sequence, typically generated by the MAC.

Fields within a frame are transmitted from top to bottom. When consecutive octets are used to represent a single numerical value, the most significant octet is transmitted first, followed by successively less significant octets. Bits within each octet are transmitted from least significant bit (LSB) to most significant bit (MSB).

5.2 VLCPDU subtype encoding

The value encoding of the **Subtype** field shall be as defined in Table 5-1.

Table 5-1—Subtype field encoding

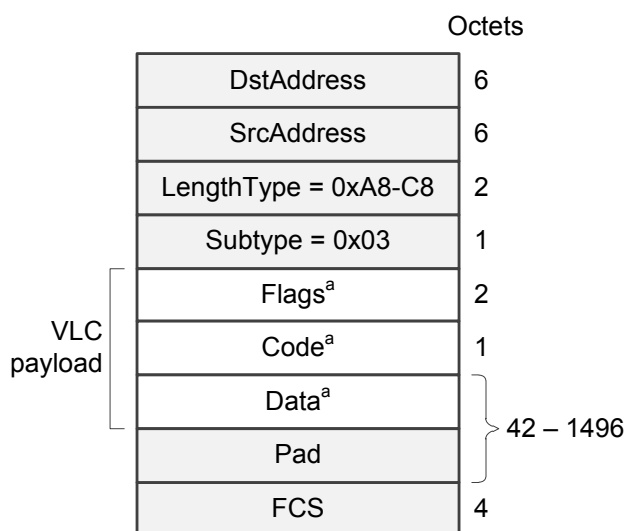
Value	Designation	Description
0x00	SUBTYPE_VLC	SUBTYPE_VLC identifies <i>VLC_CONFIG</i> VLCPDUs used for configuring the VLC Classification and Translation Engine (see 6.1).
0x01, 0x02	n/a	Reserved for Slow Protocol subtypes 0x01 and 0x02 (see IEEE Std 802.3, Table 57A-3). Ignored on reception.
0x03	SUBTYPE_OAM	SUBTYPE_OAM identifies the OAMPDU payload carried within the VLCPDU (see 5.2.2).
0x04 to 0x0A	n/a	Reserved for Slow Protocol subtypes 0x04 through 0x0A (see IEEE Std 802.3, Table 57A-3). Ignored on reception.
0x0B	SUBTYPE_EPD	SUBTYPE_EPD identifies a generic TPID field followed by a TPID-dependent payload carried within the VLCPDU (see 5.2.5).
0x0C	SUBTYPE_OMCI	SUBTYPE_OMCI identifies the OMCI payload carried within the VLCPDU (see 5.2.3).
0x0D	SUBTYPE_L2	SUBTYPE_L2 identifies a generic Ethernet frame carried within the VLCPDU (see 5.2.4).
0x0E to 0xFD	n/a	Reserved; ignored on reception.
0xFE	SUBTYPE_OUI24	SUBTYPE_OUI24 identifies an organization-specific payload carried within the VLCPDU. The organization is identified by a unique OUI/CID value (see 5.2.6).
0xFF	SUBTYPE_OUI36	SUBTYPE_OUI36 identifies an organization-specific payload carried within the VLCPDU. The organization is identified by a unique OUI-36 value (see 5.2.6).

5.2.1 VLC configuration subtype

A VLCPDU with VLC configuration subtype (Subtype field = 0x00) identifies a *VLC_CONFIG* VLCPDU used for configuring the VLC Classification and Translation Engine (see 6.1). This VLCPDU is defined in 8.1.1.

5.2.2 OAM subtype

A VLCPDU with OAM subtype (Subtype field = 0x03) is an instantiation of a generic VLCPDU as defined in 5.1 that carries an Operations, Administration, and Maintenance (OAM) payload (see IEEE Std 802.3, 57.4). The frame structure of a VLCPDU with OAM subtype shall be as depicted in Figure 5-2.



^a This field is defined in IEEE Std 802.3, 57.4

Figure 5-2—Format of VLCPDU with OAM subtype

The structure of the VLC payload in the VLCPDU with OAM subtype is defined as follows:

— **Flags:**

This field carries the value of the *Flags* field as defined in IEEE Std 802.3, 57.4.

— **Code:**

This field carries the value of the *Code* field as defined in IEEE Std 802.3, 57.4.

— **Data:**

This field carries the payload portion of the OAMPDU as defined IEEE Std 802.3, 57.4.

5.2.3 OMCI subtype

A VLCPDU with OMCI subtype (Subtype field = 0x0C) is an instantiation of a generic VLCPDU as defined in 5.1 that carries an ONU Management and Control Interface (OMCI) payload (see ITU-T Rec G.988). The frame structure of a VLCPDU with OMCI subtype shall be as depicted in Figure 5-3.

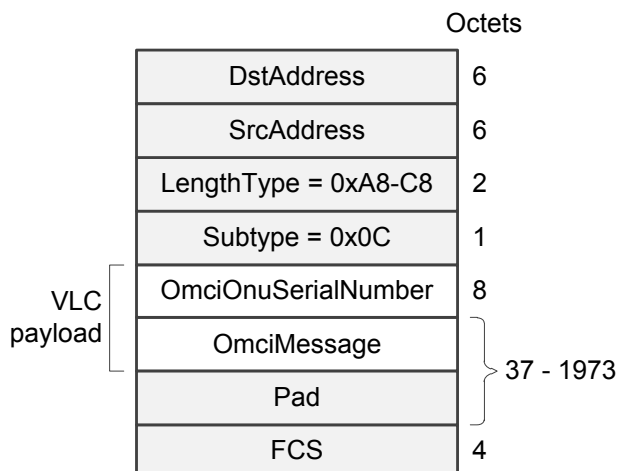


Figure 5-3—Format of VLCPDU with OMCI subtype

The structure of the VLC payload in the VLCPDU with OMCI subtype is defined as follows:

— OmciOnuSerialNumber:

This field carries the serial number of the ONU associated with the OMCI message. The serial number is defined in ITU-T Rec G.988, 9.1.1.

— OmciMessage:

This field carries one OMCI message in baseline or extended format. The OMCI baseline and extended message formats are defined in ITU-T Rec G.988, Clause 11.

5.2.4 L2 subtype

A VLCPDU with L2 subtype (Subtype field = 0x0D) is an instantiation of a generic VLCPDU as defined in 5.1 that carries a complete L2 frame as its payload. The frame structure of a VLCPDU with L2 subtype shall be as depicted in Figure 5-4.