



IEEE Standard for Local and Metropolitan Area Networks— Link Aggregation

IEEE Computer Society

Developed by the
LAN/MAN Standards Committee

IEEE Std 802.1AX™-2020
(Revision of IEEE Std 802.1AX-2014)

IEEE Standard for Local and Metropolitan Area Networks— Link Aggregation

Developed by the
**LAN/MAN Standards Committee
of the
IEEE Computer Society**

Approved 30 January 2020
IEEE SA Standards Board

Abstract: Link Aggregation allows parallel point-to-point links to be used as if they were a single link and also supports the use of multiple links as a resilient load-sharing interconnect between multiple nodes in two separately administered networks. This standard defines a MAC-independent Link Aggregation capability and provides general information relevant to specific MAC types.

Keywords: Aggregated Link, Aggregator, Distributed Resilient Network Interconnect, DRNI, interconnect, Link Aggregation, Link Aggregation Group, local area network, management, Network-Network Interface, NNI

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

Copyright © 2020 by The Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 29 May 2020. Printed in the United States of America.

IEEE and IEEE 802 are registered trademarks in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.

PDF: ISBN 978-1-5044-6428-4 STD24045
Print: ISBN 978-1-5044-6429-1 STDPD24045

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

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

IEEE Standards documents (standards, recommended practices, and guides), both full-use and trial-use, are developed within IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (“IEEE SA”) Standards Board. IEEE (“the Institute”) develops its standards through a consensus development process, approved by the American National Standards Institute (“ANSI”), which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE Standards are documents developed through scientific, academic, and industry-based technical working groups. Volunteers in IEEE working groups are not necessarily members of the Institute 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 Standards do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other devices or networks. 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.

IEEE does not warrant or represent the accuracy or content 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. 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: PROCUREMENT OF 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 should be considered 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, or 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 his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide 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 revisions to an IEEE standard is welcome to join the relevant IEEE working group.

Comments on standards should be submitted to the following address:

Secretary, IEEE SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854 USA

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 imply 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.

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 fee, 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. 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 at <https://ieeexplore.ieee.org/> or contact IEEE at the address listed previously. For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website at <https://standards.ieee.org>.

Errata

Errata, if any, for IEEE standards can be accessed via <https://standards.ieee.org/standard/index.html>. 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: <https://ieeexplore.ieee.org/browse/standards/collection/ieee/>. Users are encouraged to periodically check for errata.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE SA Website at <https://standards.ieee.org/about/sasb/patcom/patents.html>. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this revision was submitted to the IEEE SA for approval, the IEEE P802.1 Working Group had the following membership:

Glenn Parsons, *Chair*
John Messenger, *Vice Chair*
János Farkas, *Chair, Time-Sensitive Networking Task Group*
Stephen Haddock and Jessy Rouyer, *Co-Editors*

Astrit Ademaj	Yoshihiro Ito	Maximilian Riegel
Ralf Assmann	Michael Karl	Atsushi Sato
Jens Bierschenk	Stephan Kehrer	Frank Schewe
Christian Boiger	Randy Kelsey	Michael Seaman
Paul Bottorff	Hajime Koto	Maik Seewald
Radhakrishna Canchi	James Lawlis	Johannes Specht
Feng Chen	Christophe Mangin	Marius Stanica
Weiyang Cheng	Scott Mansfield	Guenter Steindl
Paul Congdon	Kenichi Maruhashi	Karim Traore
Rodney Cummings	David McCall	Hao Wang
Josef Dorr	Larry McMillan	Tongtong Wang
Hesham Elbakoury	Tero Mustala	Xinyuan Wang
Thomas Enzinger	Roy Myers	Karl Weber
Donald Fedyk	Hiroki Nakano	Brian Weis
Norman Finn	Bob Noseworthy	Ludwig Winkel
Geoffrey Garner	Tomoki Ohsawa	Jordon Woods
Craig Gunther	Hiroshi Ohue	Takahiro Yamaura
Marina Gutierrez	Donald R. Pannell	Nader Zein
Mark Hantel	Michael Potts	William Zhao
Marc Holness	Dieter Proell	Helge Zinner
Satoko Itaya	Wei Qiu	Harald Zweck
	Karen Randall	

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

Thomas Alexander	Raj Jain	Bansi Patel
Richard Alfvén	Sangkwon Jeong	Brian Petry
Butch Anton	Pranav Jha	David Piehler
Stefan Aust	Lokesh Kabra	Walter Pienciak
Harry Bims	Srinivas Kandala	Rick Pimpinella
Kenneth Bow	Piotr Karocki	Clinton Powell
Rich Boyer	Stuart Kerry	R. K. Rannow
Nancy Bravin	Evgeny Khorov	Maximilian Riegel
Vern Brethour	Yongbum Kim	Robert Robinson
Demetrio Bucaneg	Robert Landman	Jessy Rouyer
William Byrd	Hyeong Ho Lee	John Sargent
Radhakrishna Canchi	John Lemon	Frank Schewe
Paul Cardinal	James Lepp	Michael Seaman
Juan Carreon	Michael Lynch	Thomas Starai
Pin Chang	John Mackay	Walter Struppler
János Farkas	Roger Marks	Michael Thompson
John Fletcher	Arthur Marris	Mark-Rene Uchida
Avraham Freedman	Jeffery Masters	James Van De Ligt
Devon Gayle	Brett McClellan	Dmitri Varsanofiev
Zhigang Gong	Richard Mellitz	George Vlantis
Randall Groves	Jose Morales	Stephen Webb
Stephen Haddock	Ronald Murias	Karl Weber
Marek Hajduczenia	Bruce Muschlitz	Hung-Yu Wei
Marco Hernandez	Nick S. A. Nikjoo	Scott Willy
Werner Hoelzl	Satoshi Obara	Peter Wu
Yasuhiro Hyakutake	Carlos Pardo	Oren Yuen
Satoko Itaya		Nader Zein

When the IEEE-SA Standards Board approved this standard on 30 January 2020, it had the following membership:

Gary Hoffman, Chair
Vacant Position, Vice Chair
Jean-Philippe Faure, Past Chair
Konstantinos Karachalios, Secretary

Ted Burse	Howard Li	Dorothy Stanley
J. Travis Griffith	Dong Liu	Mehmet Ulema
Grace Gu	Kevin Lu	Lei Wang
Guido R. Hiertz	Paul Nikolich	Sha Wei
Joseph L. Koepfinger*	Damir Novosel	Philip B. Winston
John D. Kulick	Jon Walter Rosdahl	Daidi Zhong
David J. Law		Jingyi Zhou

*Member Emeritus

Introduction

(This introduction is not part of IEEE Std 802.1AX-2020, IEEE Standard for Local and Metropolitan Area Networks—Link Aggregation.)

Link Aggregation allows one or more links to be aggregated together to form a Link Aggregation Group (LAG) so that the Link Aggregation Client can treat the LAG as if it were a single link. Link Aggregation was originally published as IEEE Std 802.3ad™-2000 and subsequently incorporated into the IEEE Std 802.3™, 2000 Edition. In 2008 Link Aggregation was removed from IEEE Std 802.3 and published as IEEE Std 802.1AX-2008. These standards specified the aggregation of full-duplex point-to-point links using IEEE Std 802.3 media of the same speed.

An amendment, IEEE Std 802.1AXbk™-2012, specified changes to the addressing used by the link aggregation control and marker protocols to allow a LAG to span Two-Port Media Access Control (MAC) Relays (TPMRs) and to span Provider Bridged Networks and Provider Backbone Bridge Networks.

A revision, IEEE Std 802.1AX-2014, extended Link Aggregation in three areas. First, it explicitly allowed the aggregation of point-to-point links of any speed using any physical media or logical connection capable of supporting the Internal Sublayer Service specified in IEEE Std 802.1AC™. Second, it specified Conversation-Sensitive Collection and Distribution (CSCD) that provides a mechanism to identify the distribution algorithm in use to map data frames to individual links in the LAG and to convey that information to the Link Aggregation Partner via Link Aggregation Control Protocol Data Units (LACPDU)s containing version 2 type/length/values (TLVs). Third, it specified Distributed Resilient Network Interconnect (DRNI) that allows a LAG to terminate at two or three cooperating Systems so that the LAG provides resiliency to System-level failures as well as link level failures. A corrigendum, IEEE Std 802.1AX-2014/Cor 1-2017, provided technical and editorial corrections to CSCD.

This revision, IEEE Std 802.1AX-2020, makes significant refinements and simplifications to the Link Aggregation Control Protocol (LACP) as well as to CSCD and DRNI. In LACP, the Periodic state machine and Transmit state machine are combined to a single machine, and the Mux machine is optimized to reduce the likelihood of excessive Link Aggregation Control Protocol Data Unit (LACPDU) transmissions. CSCD is refined to eliminate the TLVs that led to LACPDU)s greater than 128 bytes in length. DRNI is significantly revised and simplified to support a LAG terminating at just two (not three) cooperating Systems.

Every effort has been made to maintain interoperability, without prior configuration, with LACP implementations conforming to IEEE Std 802.3ad-2000, IEEE Std 802.1AX-2008, or IEEE Std 802.1AX-2014 and with CSCD implementations conforming to IEEE Std 802.1AX-2014. The changes to DRNI, and in particular the Distributed Relay Control Protocol (DRCP), are such that an implementation conforming to this standard will not interoperate with a DRCP implementation conforming to IEEE Std 802.1AX-2014. The DRCP version number in this standard has been changed to version 2, and care has been taken so that a DRCP implementation conformant to IEEE Std 802.1AX-2014 will discard version 2 DRCPDU)s as invalid and that implementations of this standard will discard version 1 DRCPDU)s.

This standard contains state-of-the-art material. The area covered by this standard is undergoing evolution. Revisions are anticipated within the next few years to clarify existing material, to correct possible errors, and to incorporate new related material. Information on the current revision state of this and other IEEE 802 standards may be obtained from

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Contents

1.	Overview.....	18
1.1	Scope.....	18
1.2	Purpose.....	18
1.3	State diagram conventions	19
2.	Normative references	20
3.	Definitions	21
4.	Acronyms and abbreviations	24
5.	Conformance.....	25
5.1	Requirements terminology.....	25
5.2	Protocol implementation conformance statement.....	25
5.3	Link Aggregation.....	25
5.3.2	Link Aggregation options	26
5.4	Distributed Resilient Network Interconnect (DRNI).....	26
5.4.2	DRNI options.....	27
6.	Link Aggregation.....	28
6.1	Overview.....	28
6.1.1	Goals and objectives	28
6.1.2	Positioning of Link Aggregation within the IEEE 802 architecture.....	29
6.1.3	Protocol Parser/Multiplexer.....	29
6.1.3.1	Protocol Parser state diagram	30
6.2	Link Aggregation operation.....	31
6.2.1	Principles of Link Aggregation.....	33
6.2.2	Service interfaces.....	34
6.2.3	Frame Collector	34
6.2.3.1	Frame Collector state diagram.....	35
6.2.4	Frame Distributor.....	36
6.2.4.1	Frame Distributor state diagram	36
6.2.5	Marker Generator/Receiver (optional).....	37
6.2.6	Marker Responder.....	38
6.2.7	Aggregator Parser/Multiplexer	38
6.2.7.1	Aggregator Parser/Multiplexer state diagrams	38
6.2.8	Aggregator	42
6.2.9	LACP Parser/Multiplexer	43
6.2.10	Addressing	43
6.2.10.1	Source address (SA).....	43
6.2.10.2	Destination address (DA).....	43
6.3	Link Aggregation Control.....	44
6.3.1	Characteristics of Link Aggregation Control.....	45
6.3.2	System identification	46
6.3.3	Aggregator identification.....	46
6.3.4	Port identification	46
6.3.5	Capability identification	47
6.3.6	Link Aggregation Group identification	48

	6.3.6.1	Construction of the Link Aggregation Group Identifier	48
	6.3.6.2	Representation of the Link Aggregation Group Identifier.....	48
6.3.7		Selecting a Link Aggregation Group	49
6.3.8		Agreeing on a Link Aggregation Group	49
6.3.9		Attaching a link to an Aggregator.....	50
6.3.10		Signaling readiness to transfer user data.....	50
6.3.11		Enabling the Frame Collector and Frame Distributor	50
6.3.12		MAC_Operational status	51
6.3.13		Monitoring the membership of a Link Aggregation Group.....	51
6.3.14		Detaching a link from an Aggregator	51
6.3.15		Configuration and administrative control of Link Aggregation	52
6.3.16		Link Aggregation Control state information	52
6.4		Link Aggregation Control Protocol	52
6.4.1		LACP design elements.....	52
6.4.2		LACPDU structure and encoding	53
	6.4.2.1	Transmission and representation of octets.....	53
	6.4.2.2	Encapsulation of LACPDU in frames	53
	6.4.2.3	LACPDU structure	54
	6.4.2.4	Version 2 TLVs	57
6.4.3		LACP state machine overview	59
6.4.4		Constants.....	60
6.4.5		Variables associated with each Aggregator	61
6.4.6		Variables associated with each Aggregation Port.....	63
6.4.7		Variables used for managing the operation of the state machines.....	68
6.4.8		Functions.....	69
6.4.9		Timers	72
6.4.10		Messages	72
6.4.11		LACP Receive machine.....	72
6.4.12		Selection Logic	74
	6.4.12.1	Selection Logic—Requirements	75
	6.4.12.2	Selection Logic—Recommended default operation	77
6.4.13		Mux machine	78
6.4.14		LACP Transmit machine	81
6.5		Marker protocol	82
6.5.1		Introduction.....	82
6.5.2		Sequence of operations	83
6.5.3		Marker and Marker Response PDU structure and encoding	83
	6.5.3.1	Transmission and representation of octets.....	83
	6.5.3.2	Encapsulation of Marker and Marker Response PDU in frames.....	83
	6.5.3.3	Marker and Marker Response PDU structure.....	84
6.5.4		Protocol definition	85
	6.5.4.1	Operation of the marker protocol.....	85
	6.5.4.2	Marker Responder state diagram	86
6.6		Conversation-Sensitive Collection and Distribution	87
6.6.1		Port Algorithms and Port Conversation IDs	89
6.6.2		Link numbers and link selection.....	89
6.6.3		Conversation-sensitive LACP.....	90
	6.6.3.1	Per-Aggregator variables	91
	6.6.3.2	Variables associated with each Aggregation Port.....	94
	6.6.3.3	Variables used for managing the operation of the state diagrams	95
	6.6.3.4	Functions.....	96
	6.6.3.5	Update Mask machine	100
6.7		Configuration capabilities and restrictions	102
6.7.1		Use of system and port priorities	102

6.7.2	Dynamic allocation of operational Keys	103
6.7.3	Link Aggregation on shared-medium links	103
6.7.4	Selection Logic variants.....	104
6.7.4.1	Reduced reconfiguration.....	104
6.7.4.2	Limited Aggregator availability.....	104
6.7.5	LACP configuration for dual-homed Systems.....	104
7.	Management.....	106
7.1	Overview.....	106
7.1.1	Systems management overview.....	106
7.1.2	Management model.....	107
7.2	Managed objects	107
7.2.1	Introduction.....	107
7.2.2	Overview of managed objects.....	108
7.2.2.1	Text description of managed objects	108
7.2.3	Containment.....	108
7.2.4	Naming.....	109
7.2.5	Capabilities	109
7.3	Management for Link Aggregation	114
7.3.1	Aggregator managed object class	114
7.3.1.1	Aggregator attributes	115
7.3.1.2	Aggregator Notifications	125
7.3.2	Aggregation Port managed object class.....	125
7.3.2.1	Aggregation Port Attributes.....	125
7.3.2.2	Aggregation Port Extension Attributes.....	132
7.3.3	Aggregation Port Statistics managed object class	133
7.3.3.1	Aggregation Port Statistics attributes	133
7.3.4	Aggregation Port Debug Information managed object class	134
7.3.4.1	Aggregation Port Debug Information attributes	135
7.4	Management for Distributed Resilient Network Interconnect.....	137
7.4.1	DRNI Managed Object Class	137
7.4.1.1	DRNI Attributes.....	137
8.	Distribution algorithms	148
8.1	Distribution algorithm identification	148
8.2	Per-Service Frame Distribution	149
8.2.1	Distribution based on C-VLAN Identifier (C-VID)	149
8.2.2	Distribution based on S-VLAN Identifier (S-VID)	150
8.2.3	Distribution based on Backbone Service Instance Identifier (I-SID).....	150
8.2.4	Distribution based on Traffic Engineering Service Instance Identifier (TE-SID).....	150
8.2.5	Distribution based on Flow Hash.....	150
9.	Distributed Resilient Network Interconnect	151
9.1	Goals	151
9.2	Distributed Relay operation.....	152
9.3	Intra-Relay Connection.....	154
9.4	Using DRNI.....	155
9.4.1	DRNI connectivity.....	155
9.4.2	DRNI fault recovery	157
9.4.3	DRNI configuration	158

9.5	DRNI Gateway	159
9.5.1	DRCP Parser/Multiplexer	160
9.5.2	DRNI Gateway Relay	160
9.5.2.1	Service interfaces	160
9.5.2.2	variables	161
9.5.2.3	Functions	162
9.5.2.4	Messages	162
9.5.2.5	DR_Gateway Collector/Distributor	162
9.5.2.6	DR_Aggregator Parser/Multiplexer	163
9.5.2.7	DR_IRP Parser/Multiplexer	164
9.5.3	DRNI Gateway Control	165
9.5.3.1	DRNI Gateway and DRNI Identification	165
9.5.3.2	Forming the DRNI	166
9.5.3.3	Partner Selection and Forming a LAG	166
9.5.3.4	Port Algorithm and Aggregator Port Selection	166
9.5.3.5	Gateway Algorithm and DRNI Gateway Port selection	167
9.6	Distributed Relay Control Protocol	168
9.6.1	DRCPDU transmission, addressing, and protocol identification	169
9.6.1.1	Destination MAC Address	169
9.6.1.2	Source MAC Address	170
9.6.1.3	Priority	170
9.6.1.4	Protocol Identification	170
9.6.1.5	Encapsulation of DRCPDUs in frames	170
9.6.2	DRCPDU structure and encoding	170
9.6.2.1	Transmission and representation of octets	170
9.6.2.2	DRCP TLV structure	171
9.6.2.3	DRCPDU structure	172
9.6.2.4	Aggregator State TLV	174
9.6.2.5	Gateway State TLV	176
9.6.2.6	Gateway Preference TLV	176
9.6.2.7	Organization-Specific TLV	177
9.6.3	DRCP state machine overview	178
9.6.4	Constants	179
9.6.5	Variables associated with the DRNI Gateway	179
9.6.6	Variables used for managing the operation of the state machines	185
9.6.7	Functions	187
9.6.8	Timers	194
9.6.9	Messages	194
9.6.10	DRCP Receive machine	195
9.6.11	Distributed Relay machine	197
9.6.12	DRNI Gateway and Aggregator machine	198
9.6.13	DRCP Transmit machine	199
	Annex A (normative) Protocol implementation conformance statement (PICS) proforma	200
A.1	Introduction	200
A.1.1	Abbreviations and special symbols	200
A.1.2	Instructions for completing the PICS proforma	201
A.1.3	Additional information	201
A.1.4	Exceptional information	201
A.1.5	Conditional items	202
A.1.6	Identification	202
A.1.6.1	Implementation identification	202
A.1.6.2	Protocol summary	202

A.2	PICS proforma for Clause 6.....	203
A.2.1	Major capabilities/options	203
A.2.3	Protocol Parser/Multiplexer support	204
A.2.4	Frame Collector.....	204
A.2.5	Frame Distributor	204
A.2.2	LLDP Port connectivity	204
A.2.7	Aggregator Parser/Multiplexer.....	205
A.2.8	LACP Parser/Multiplexer.....	205
A.2.6	Marker protocol.....	205
A.2.10	Aggregator identification	206
A.2.11	Port identification.....	206
A.2.12	Capability identification.....	206
A.2.9	System identification.....	206
A.2.14	Detaching a link from an Aggregator.....	207
A.2.15	LACPDU structure.....	207
A.2.16	Receive machine	207
A.2.13	Link Aggregation Group identification.....	207
A.2.18	Mux machine.....	208
A.2.17	Selection Logic.....	208
A.2.20	Marker protocol.....	209
A.2.19	Transmit machine.....	209
A.2.21	Management	210
A.2.23	Conversation-sensitive frame collection and distribution.....	211
A.2.22	Per-Service Frame Distribution.....	211
A.2.25	Link Aggregation on shared-medium links.....	212
A.2.24	Configuration capabilities and restrictions.....	212
A.2.27	DRCPDU structure.....	213
A.2.26	Distributed Resilient Network Interconnect.....	213
Annex B	(informative) Collection and distribution algorithms.....	214
B.1	Introduction.....	214
B.2	Port selection.....	215
B.3	Dynamic reallocation of conversations to different Aggregation Ports	215
B.4	Topology considerations in the choice of distribution algorithm	216
Annex C	(informative) LACP standby link selection and dynamic Key management	218
C.1	Introduction.....	218
C.2	Goals	218
C.3	Standby link selection.....	219
C.4	Dynamic Key management.....	219
C.5	A dynamic Key management algorithm	219
C.6	Example 1	221
C.7	Example 2	221
Annex D	(normative) SMIPv2 MIB definitions for Link Aggregation	223
D.1	Introduction.....	223
D.2	SNMP Management Framework	223
D.3	Security considerations	223
D.4	Structure of the MIB module	224
D.4.1	Relationship to the managed objects defined in Clause 7	225
D.4.2	MIB Subtrees.....	232
D.4.2.1	The dot3adAgg Subtree.....	232
D.4.2.2	The dot3adAggPort Subtree	232

D.4.2.3	The dot3adAggNotifications Subtree.....	232
D.4.2.4	The dot3adDrni Subtree	232
D.5	Relationship to other MIBs.....	233
D.5.1	Relationship to the Interfaces MIB	233
D.5.2	Layering model	233
D.5.3	ifStackTable	234
D.5.4	ifRcvAddressTable.....	234
D.6	Definitions for Link Aggregation MIB.....	234
Annex E (informative)	DRNI on Bridges	325
E.1	DRNI on VLAN Bridges	325
E.2	DRNI on Provider Bridges and Provider Edge Bridges	326
E.3	DRNI on Backbone Edge Bridges	327
Annex F (normative)	Link Aggregation and Link Layer Discovery Protocol.....	328
F.1	Positioning Link Layer Discovery Protocol (LLDP) relative to Link Aggregation	328
F.1.1	LLDP Parser/Multiplexer.....	328
F.2	Link Aggregation TLV	328
F.2.1	aggregation status.....	329
F.2.2	aggregated Port ID	329
F.2.3	Link Aggregation TLV usage rules.....	329
F.2.4	Use of other TLVs on an Aggregator or Aggregation Link	330
Annex G (informative)	Bibliography	331

Figures

Figure 6-1	Architectural positioning of Link Aggregation sublayer	29
Figure 6-2	Protocol Parser state diagram.....	31
Figure 6-3	Link Aggregation sublayer block diagram.....	32
Figure 6-4	Frame Collector state diagram	36
Figure 6-5	Frame Distributor state diagram	37
Figure 6-6	Aggregator Parser state diagram	41
Figure 6-7	Aggregator Multiplexer state diagram	42
Figure 6-8	LACPDU structure.....	54
Figure 6-9	Bit encoding of the Actor_State and Partner_State fields.....	55
Figure 6-10	Port Algorithm TLV.....	57
Figure 6-11	Port Conversation Link Digest TLV	58
Figure 6-12	Port Conversation Service Digest TLV	58
Figure 6-13	LACP state machine overview.....	59
Figure 6-14	LACP Receive state diagram	73
Figure 6-15	Selection of Aggregators.....	78
Figure 6-16	Mux state diagram.....	79
Figure 6-17	LACP Transmit state diagram.....	81
Figure 6-18	Marker protocol time sequence diagram.....	83
Figure 6-19	Marker PDU and Marker Response PDU structure	84
Figure 6-20	Marker Responder state diagram	86
Figure 6-21	Conversation-Sensitive Collection and Distribution overview.....	88
Figure 6-22	Distribution algorithm information flow.....	91
Figure 6-23	Update Mask state diagram	101
Figure 6-24	Dual-homed System examples.....	105
Figure 7-1	Link aggregation entity relationship diagram	109
Figure 8-1	Distribution algorithm identifiers.....	148
Figure 9-1	A DRNI	153
Figure 9-2	Transmission and reception via the IRC	154
Figure 9-3	DRNI with dedicated IRC Link	155
Figure 9-4	DRNI with dedicated IRC LAG.....	155
Figure 9-5	DRNI end station attach	156
Figure 9-6	DRNI network attach	156
Figure 9-7	DRNI Gateway block diagram.....	159
Figure 9-8	DR_Gateway state machine	163
Figure 9-9	DR_Aggregator state machine	164
Figure 9-10	DR_IRP state machine	165
Figure 9-11	Basic TLV format	171
Figure 9-12	DRCPDU structure	172
Figure 9-13	Bit encoding of the Home_IRP_State and Neighbor_IRP_State fields.....	173
Figure 9-14	Aggregator State TLV	174
Figure 9-15	Bit encoding of the Aggregator_CSCD_State fields	175
Figure 9-16	Gateway State TLV.....	176
Figure 9-17	Gateway Preference TLV.....	176
Figure 9-18	Organization-Specific TLV.....	177
Figure 9-19	DRCP state machine overview	178
Figure 9-20	DRCP Receive machine state diagram	195
Figure 9-21	Distributed Relay machine state diagram	197
Figure 9-22	DRNI Gateway and Aggregator machine state diagram.....	198
Figure 9-23	DRCP Transmit state diagram	199
Figure B-1	Link aggregation topology examples	217
Figure C-1	Example 1	221

Figure C-2	Example 2a.....	222
Figure C-3	Example 2b	222
Figure E-1	DRNI on VLAN Bridges	325
Figure E-2	DRNI on Provider Edge Bridges	326
Figure E-3	DRNI on Backbone Edge Bridges	327
Figure F-1	Link Aggregation TLV format.....	328

Tables

Table 6-1	Link Aggregation protocol destination addresses.....	44
Table 6-2	Example Partner parameters	49
Table 6-3	Slow Protocols EtherType Assignment	53
Table 6-4	Type field values of Version 2 TLVs	57
Table 7-1	Link Aggregation capabilities.....	110
Table 8-1	IEEE per-service distribution algorithms	149
Table 9-1	Distributed Relay Control Protocol destination addresses	169
Table 9-2	DRNI EtherType Assignment.....	170
Table 9-3	DRNI Protocol subtypes	170
Table 9-4	Type field values of DRCP TLVs.....	171
Table F.1	Link aggregation capability/status	329

IEEE Standard for Local and Metropolitan Area Networks— Link Aggregation

1. Overview

1.1 Scope

Link Aggregation provides protocols, procedures, and managed objects that allow the following:

- One or more parallel instances of point-to-point links to be aggregated together to form a Link Aggregation Group (LAG) so that a Link Aggregation Client can treat the LAG as if it were a single link.
- Conversation-Sensitive Collection and Distribution (CSCD) that specifies a means to identify the distribution algorithm in use to assign frames to individual links in a LAG and to convey that information to the System at the other end of the LAG.
- Distributed Resilient Network Interface (DRNI) that enables a LAG to terminate at a pair of cooperating Systems in order to provide system-level as well as link-level resiliency.

1.2 Purpose

Link Aggregation allows the establishment of point-to-point links that have a higher aggregate bandwidth than the individual links that form the aggregation and the use of multiple systems at each end of the aggregation. This allows improved utilization of available links in Bridged local area network (LAN) environments, along with improved resilience in the face of failure of individual links or systems. In applications connecting separately administered networks, the networks are isolated from each other's fault recovery events.

1.3 State diagram conventions

This standard uses the state diagram conventions in Annex E of IEEE Std 802.1Q™-2018.¹

When implementation of a state diagram is required for conformance, all states and transitions shown with solid lines shall be implemented. States and transitions shown with dashed lines may be implemented.

If a conflict exists between a state diagram and either the corresponding global transition tables or the textual description associated with the state machine, the state diagram takes precedence.

¹ Information on references can be found in Clause 2.

2. Normative references

The following referenced documents are indispensable for the application of this standard (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this standard 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[®], IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture.^{2, 3}

IEEE Std 802.1AC[™], IEEE Standard for Local and metropolitan area networks—Media Access Control (MAC) Service Definition.

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

IEEE Std 802.3[™], IEEE Standard for Ethernet.

IETF RFC 1213 (IETF STD 17), Management Information Base for Network Management of TCP/IP-based internets: MIB-II, McCloghrie, K., and M. Rose, editors, Mar. 1991.⁴

IETF RFC 1321, The MD5 Message-Digest Algorithm, R. Rivest, Apr. 1992.

IETF RFC 2578 (STD 58), Structure of Management Information Version 2 (SMIv2), McCloghrie, K., D. Perkins, and J. Schoenwaelder, Apr. 1999.

IETF RFC 2579 (STD 58), Textual Conventions for SMIv2, McCloghrie, K., D. Perkins, J. Schoenwaelder, J. Case, M. Rose, and S. Waldbusser, Apr. 1999.

IETF RFC 2580 (STD 58), Conformance Statements for SMIv2, McCloghrie, K., D. Perkins, J. Schoenwaelder, J. Case, M. Rose, and S. Waldbusser, Apr. 1999.

IETF RFC 2863, The Interfaces Group MIB, McCloghrie, K., and F. Kastenholz, June 2000.

IETF RFC 3410, Introduction and Applicability Statements for Internet-Standard Management Framework, Case, J., R. Mundy, D. Partain, and B. Stewart, Dec. 2002.

IETF RFC 3414 (STD 62), User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3), Blumenthal, U., and B. Wijnen, Dec. 2002.

IETF RFC 3415 (STD 62), View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP), Wijnen, B., R. Presuhn, and K. McCloghrie, Dec. 2002.

ISO/IEC 10165-4:1992, Information technology—Open Systems Interconnection—Structure of management information—Part 4: Guidelines for the definition of managed objects.⁵

² IEEE publications are available from The Institute of Electrical and Electronics Engineers (<https://standards.ieee.org/>).

³ The IEEE standards or products referred to in this clause are trademarks of The Institute of Electrical and Electronics Engineers, Inc.

⁴ IETF documents (i.e., RFCs) are available from the Internet Engineering Task Force (<https://www.rfc-archive.org/>).

⁵ ISO/IEC publications are available from the International Organization for Standardization (<https://www.iso.org/>), the International Electrotechnical Commission (<https://www.iec.ch>), and the American National Standards Institute (<https://www.ansi.org/>).

3. Definitions

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

This standard makes use of the following terms defined in IEEE Std 802:

- end station
- logical link
- station

This standard makes use of the following terms defined in IEEE Std 802.1AC:

- client (7.1 of IEEE Std 802.1AC-2016)
- frame (7.2 of IEEE Std 802.1AC-2016)
- Internal Sublayer Service (ISS) (Clause 11 of IEEE Std 802.1AC-2016)
- port (7.4 of IEEE Std 802.1AC-2016)

NOTE 1—It is helpful for the reader of this standard to review IEEE Std 802.1AC's description of the basic architectural concepts and terms used in this standard.⁷

This standard makes use of the following terms defined in IEEE Std 802.1Q:

- Bridge
- service instance

Actor: The local entity in a Link Aggregation Control Protocol exchange.

Aggregateable: A link terminating at an Aggregation Port that is configured to allow that link to form a Link Aggregation Group with other Aggregation Links.

Aggregation Key: A parameter, associated with each Aggregation Port and with each Aggregator of an Aggregation System, that prohibits the participation of multiple Aggregation Ports and Aggregators in the same Link Aggregation Group if their Aggregation Keys differ.

Aggregation Link: A local area network (LAN) that connects two Aggregation Systems.

Aggregation Port: A Service Access Point in an Aggregation System that provides access to a single Aggregation Link.

Aggregation System: A system that includes one or more Aggregators, each capable of using one or more Aggregation Ports to provide service to an Aggregator Client.

Aggregator: A multiplexing/demultiplexing entity that is capable of providing service at an Aggregator Port using the services provided by one or more Aggregation Ports.

Aggregator Client: A protocol entity that makes use of the service provided by a single Aggregator.

Aggregator Port: A port that provides access to the service provided by a single Aggregator.

⁶ *IEEE Standards Dictionary Online* subscription is available at <https://dictionary.ieee.org>. An IEEE Account is required for access to the dictionary, and one can be created at no charge on the dictionary sign-in page.

⁷ Notes in text, tables, and figures of a standard are given for information only and do not contain requirements needed to implement this standard.

conversation: A set of frames transmitted from one end station to another, with the assumption that the communicating end stations require intermediate systems to maintain the ordering of those frames.

Conversation ID: An integer in the range of 0 through 4095, where each value identifies one or more conversations.

Distributed Relay: Transmission and reception of frames by Distributed Resilient Network Interconnect (DRNI) Gateway Ports in a pair of DRNI Systems using Aggregator Ports and Intra-Relay Ports in both DRNI Systems.

Distributed Relay Client: A protocol entity that makes use of the service provided by a DRNI Gateway.

Distributed Resilient Network Interconnect (DRNI): The DRNI Gateways in a pair of Aggregation Systems that are connected by an Intra-Relay Connection and can be used to create a single Link Aggregation Group using Aggregation Ports from either or both of the Aggregation Systems.

Down frame: A frame that has entered a Distributed Relay from the Distributed Relay Client.

DRNI Gateway: A protocol entity in an Distributed Resilient Network Interconnect (DRNI) System that can be paired with a DRNI Gateway in another DRNI System and that provides Distributed Relay service at a DRNI Gateway Port using the services provided by an Aggregator Port in each system.

DRNI Gateway Port: A port that provides access to the service provided by a single Distributed Resilient Network Interconnect (DRNI) Gateway.

DRNI System: An Aggregation System containing one or more Distributed Resilient Network Interconnect (DRNI) Gateways.

Gateway Conversation ID: A Conversation identifier (ID) associated with a frame for the purpose of selecting a Distributed Resilient Network Interconnect (DRNI) Gateway Port.

Home: The local entity in a Distributed Relay Control Protocol (DRCP) exchange.

Intra-Relay Connection (IRC): A logical link that connects a pair of Distributed Resilient Network Interconnect (DRNI) Gateways comprising a Distributed Relay.

Intra-Relay Port (IRP): The service interface of a Distributed Resilient Network Interconnect (DRNI) Gateway provided by an Intra-Relay Connection.

Key: *See: Aggregation Key.*

Link Aggregation Group (LAG): A group of links that appear to an Aggregator Client as a single link.

Link_Number: An Aggregation Link identifier that is unique within a Link Aggregation Group, with an operational value common to both the Actor and Partner Systems.

Neighbor: The remote entity in a Distributed Relay Control Protocol (DRCP) exchange.

Partner: The remote entity in a Link Aggregation Control Protocol (LACP) exchange.

Port Conversation ID: A Conversation identifier (ID) associated with a frame for the purpose of selecting an Aggregation Port.

Service Identifier (Service ID): A value extracted from the header of a frame (e.g., VID, I-SID) that identifies the service instance with which that frame is associated.

NOTE 2—This standard makes use of service identifiers specified in IEEE Std 802.1Q.

Solitary: An Aggregation Link that cannot be included in a Link Aggregation Group with any other Aggregation Link, due to administrative configuration at either of the attached Aggregation Systems.

System: When used in this standard and not preceded by “Aggregation” or “DRNI”, “System” means an Aggregation System.

Up frame: A frame that has entered a Distributed Relay through an Aggregator Port.

4. Acronyms and abbreviations

This standard contains the following acronyms and abbreviations:

CID	Company ID ⁸ (see also OUI)
CSCD	Conversation-Sensitive Collection and Distribution
DA	destination address
DRCP	Distributed Relay Control Protocol
DRCPDU	Distributed Relay Control Protocol Data Unit
DRNI	Distributed Resilient Network Interconnect
DWC	Discard Wrong Conversation
FCS	frame check sequence
IRC	Intra-Relay Connection
IRP	Intra-Relay Port
ISS	Internal Sublayer Service
LACP	Link Aggregation Control Protocol
LACPDU	Link Aggregation Control Protocol Data Unit
LAG	Link Aggregation Group
LAG ID	Link Aggregation Group Identifier
LAN	local area network
LLC	logical link control
LLDP	Link Layer Discovery Protocol
MAC	medium access control
MIB	management information base
MSAP	MAC Service Access Point
NTT	Need To Transmit
OUI	organizationally unique identifier
PDU	Protocol Data Unit
PICS	protocol implementation conformance statement
SA	source address
TLV	type/length/value
TPMR	Two-Port MAC Relay
UCT	unconditional transition

⁸ See <https://standards.ieee.org/develop/regauth/tut/eui.pdf>.

5. Conformance

This clause specifies the mandatory and optional capabilities provided by conformant implementations of this standard.

5.1 Requirements terminology

For consistency with existing IEEE and IEEE 802.1™ standards, requirements placed upon conformant implementations of this standard are expressed using the following terminology:

- a) **Shall** is used for mandatory requirements;
- b) **May** is used to describe implementation or administrative choices (“may” means “is permitted to,” and hence, “may” and “may not” mean precisely the same thing);
- c) **Should** is used for recommended choices (the behaviors described by “should” and “should not” are both permissible but not equally desirable choices).

The protocol implementation conformance statement (PICS) proformas (see Annex A) reflect the occurrences of the words “shall,” “may,” and “should” within the standard.

The standard avoids needless repetition and apparent duplication of its formal requirements by using **is**, **is not**, **are**, and **are not** for definitions and the logical consequences of conformant behavior. Behavior that is permitted but is neither always required nor directly controlled by an implementer or administrator, or whose conformance requirement is detailed elsewhere, is described by **can**. Behavior that never occurs in a conformant implementation or System of conformant implementations is described by **cannot**.

This clause lists requirements and options for certain capabilities, e.g., Link Aggregation. If a System claims to support such a capability for one of its ports, then the System is considered *conformant* if it meets that capability’s requirements in this clause for that port. For example, a claim that a System supports Link Aggregation on a set of ports requires the System’s behavior to meet the requirements specified in 5.3 for the identified ports.

5.2 Protocol implementation conformance statement

The supplier of an implementation that is claimed to conform to this standard shall provide the information necessary to identify both the supplier and the implementation, and shall complete a copy of the PICS proforma provided in Annex A.

5.3 Link Aggregation

5.3.1 Link Aggregation requirements

A conformant Link Aggregation System shall

- a) Support the Link Aggregation Sublayer and conform to the state machines and procedures in 6.2.
- b) Support the Link Aggregation Control Protocol (LACP) and conform to the state machines and procedures in 6.3 and 6.4.
- c) Transmit and receive Link Aggregation Control Protocol Data Unit (LACPDUs) in the formats specified in 6.4.2.
- d) Implement the Marker Responder as specified in 6.5.4.2 and respond to received Marker PDUs as specified in 6.5.1.
- e) Support the LACP configuration for dual-homed Systems described in 6.7.5.

5.3.2 Link Aggregation options

A conformant Link Aggregation System may

- a) Support the Marker Protocol and conform to the state machines and procedures in 6.5 and transmit and receive required Marker and Marker Response PDUs in the formats specified in 6.5.3.
- b) Support the management functionality for Link Aggregation as specified in Clause 7.
- c) Support SMIPv2 management information base (MIB) modules for the management of Link Aggregation capabilities (Annex D).
- d) Support Aggregation Port Debug Information package support as specified in 7.2.5.
- e) Conform to the required specifications of the Link Layer Discovery Protocol (LLDP) of IEEE Std 802.1AB™ [B1].⁹ A conformant System that supports LLDP shall provide an LLDP Parser/Multiplexer to attach zero or more instances of LLDP to each Aggregation Port as specified in F.1.
- f) Implement a Protocol Parser/Multiplexer (6.1.3) for a protocol not explicitly specified in this standard.
- g) Implement the techniques specified in 6.7.1 and 6.7.2 to form Link Aggregation Groups on an Aggregation System with the constraints described.
- h) Support Conversation-Sensitive Collection and Distribution as specified in 6.2.7 and 6.6. An implementation that supports Conversation-Sensitive Collection and Distribution shall support a value of FORCE_FALSE for Admin_Discard_Wrong_Conversation and may support values of FORCE_TRUE and AUTO.
- i) Support Per-Service Frame Distribution as specified in 8.2 and shall
 - 1) Support Conversation-Sensitive Collection and Distribution as specified in 6.2.7 and 6.6, and
 - 2) Transmit and receive version 2 LACPDUs as specified in 6.4.2.4, and
 - 3) Support classification by Customer VLAN Tag for C-tagged interfaces and classification by Service VLAN tag for S-tagged and B-tagged interfaces, and may
 - 4) Support classification by Backbone Service Instance Tag as specified in 8.2.3.

5.4 Distributed Resilient Network Interconnect (DRNI)

5.4.1 DRNI requirements

A conformant DRNI System shall

- a) Conform to the requirements specified for Link Aggregation as specified in 5.3.
- b) Support a DRNI consisting of two DRNI Systems as specified in Clause 9 and shall
 - 1) Support a DRNI Gateway having a single Intra-Relay Port (IRP) and conform to the state machines and procedures in 9.5,
 - 2) Support the Distributed Relay Control Protocol (DRCP) and conform to the state machines and procedures specified in 9.6, and
 - 3) Transmit and receive required Distributed Relay Control Protocol Data Units (DRCPDUs) in the formats specified in 9.6.2.
- c) Support using a dedicated link for the Intra-Relay Connection (IRC) as specified in 9.3.

⁹ The numbers in brackets correspond to the numbers in the bibliography in Annex G.

5.4.2 DRNI options

A conformant DRNI System may

- a) Implement any of the options specified for Link Aggregation (5.3.2).
- b) Support using a LAG for the IRC by supporting the IRP with a dedicated Aggregator as specified in 9.3.
- c) Support using a method other than a dedicated link or LAG for the IRC as specified in 9.3.
- d) Support Organization-Specific TLVs in DRCPDUs as specified in 9.6.2.7.

6. Link Aggregation

6.1 Overview

This clause defines an optional Link Aggregation sublayer for use with links offering the IEEE 802.1AC Internal Sublayer Service (ISS). Link Aggregation allows one or more point-to-point links to be aggregated together to form a Link Aggregation Group (LAG) so that an Aggregator Client can treat the LAG as if it were a single link.

The models of operation in this clause provide a basis for specifying the externally observable behavior of the operation and are not intended to place additional constraints on implementations; these can adopt any internal model of operation compatible with the externally observable behavior specified. Conformance of equipment to this standard is purely with respect to observable protocol.

6.1.1 Goals and objectives

Link Aggregation, as specified in this clause, provides the following:

- a) **Increased bandwidth**—The capacity of multiple links is combined into one logical link.
- b) **Increased availability**—The failure or replacement of a single link within a LAG need not cause failure from the perspective of an Aggregator Client.
- c) **Load sharing**—Aggregator Client traffic can be distributed across multiple links.
- d) **Automatic configuration**—In the absence of manual overrides, an appropriate set of LAGs is automatically configured, and individual links are allocated to those groups. If a set of links can aggregate, they will aggregate.
- e) **Rapid configuration and reconfiguration**—In the event of changes in physical connectivity, Link Aggregation will quickly converge to a new configuration, typically on the order of milliseconds for link down events and 1 s or less for link up events.

NOTE 1—Timing out LACPDU exchanges is the method of last resort for detecting link failures and shifting data flows to other physical links. The MAC_Operational status generated by link hardware is the first choice for link failure detection. Connectivity Fault Management (CFM) (Clause 18 of IEEE Std 802.1Q-2018) provides a method for detecting link failures (also indicated via the MAC_Operational status) when hardware means are not applicable.

- f) **Deterministic behavior**—The link aggregation configuration can be made independent of past events and completely determined by the connectivity provided by the individual links and the administrative configuration of the connected systems.
- g) **Low risk of duplication or misordering**—During both steady-state operation and link (re)configuration, there is a low probability that frames are duplicated or misordered.
- h) **Support of existing medium access control (MAC) Clients**—No change is required to existing higher-layer protocols or applications to use Link Aggregation.
- i) **Backwards compatibility with aggregation-unaware devices**—Links that cannot take part in Link Aggregation because of their inherent capabilities, management configuration, or the capabilities of the devices to which they attach operate as normal, individual links.
- j) **Accommodation of differing capabilities and constraints**—Devices with differing hardware and software constraints on Link Aggregation are, to the extent possible, accommodated.
- k) **No change to frame formats**—Link Aggregation neither adds to, nor changes the contents of, frames exchanged between Aggregator Clients.
- l) **Network management support**—The standard specifies appropriate management objects for configuration, monitoring, and control of Link Aggregation.
- m) **Point-to-point aggregations**—The mechanisms specified in this clause form a Link Aggregation Group providing a point-to-point connectivity service.

- n) **Dissimilar MACs**—Link Aggregation can be used over any physical or logical medium supporting the ISS with a service providing point-to-point connectivity (i.e., the ISS operPointToPoint variable is TRUE).

Aggregating different data rates is not prohibited nor required by this standard. Determining how to distribute traffic across links of different data rates is beyond the scope of this standard.

NOTE 2—Previous versions of this standard restricted the data rate of the aggregated links. However, the mechanisms specified did not depend on whether the links operated at the same rate.

6.1.2 Positioning of Link Aggregation within the IEEE 802 architecture

Link Aggregation comprises an optional sublayer that (at its upper boundary) has an ISS to Aggregator Clients attached to Aggregator Ports and (at its lower boundary) has an ISS for each of its Aggregation Ports. Figure 6-1 depicts a common case where Aggregation Ports in a system are supported directly by a MAC.

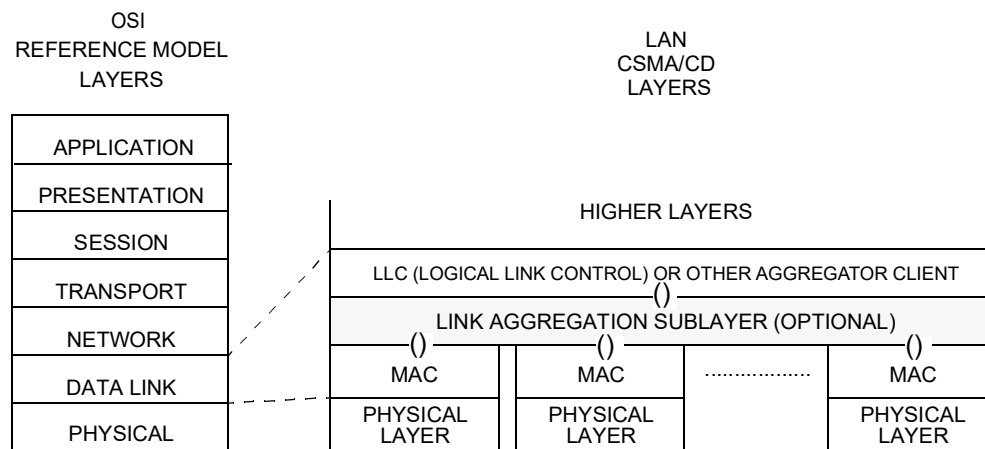


Figure 6-1—Architectural positioning of Link Aggregation sublayer

It is possible to implement the optional Link Aggregation sublayer for some ports within a System while not implementing it for other ports; i.e., it is not necessary for all ports in a System to be subject to Link Aggregation. A conformant implementation is not required to be able to apply the Link Aggregation sublayer to every port.

NOTE—IEEE Std 802.1AX-2008 and Clause 43 in earlier editions of IEEE Std 802.3 used the term “MAC Client” instead of the term “Aggregator Client” and placed Link Aggregation between the IEEE 802.3 MAC sublayer and the IEEE 802.3 MAC Control sublayer. IEEE Std 802.1AX-2014 repositioned Link Aggregation as a sublayer above the MAC sublayer. This rearrangement aligned the standard with common industry practice. It did not cause implementations conformant to earlier editions of the standard to become non-interoperable with those conformant to IEEE Std 802.1AX-2014 and subsequent editions.

6.1.3 Protocol Parser/Multiplexer

A Protocol Parser/Multiplexer is a functional element that separates specific control protocol frames from a sequence of M_UNITDATA.indications received at an ISS interface and combines frames generated for that control protocol with a sequence of M_UNITDATA.requests at an ISS interface. The general purpose Protocol Parser/Multiplexer described here can be customized to a specific control protocol, of which there are three examples in this standard:

- a) The LLDP Parser/Multiplexer (F.1.1)
- b) The LACP Parser/Multiplexer (6.2.9)
- c) The DRCP Parser/Multiplexer (9.5.1)

NOTE—The Aggregator Parser/Multiplexer (6.2.7) and DR_Aggregator Parser/Multiplexer (9.5.2.6) are not instances of a Protocol Parser/Multiplexer. These perform additional parsing functions not included in the general purpose Protocol Parser/Multiplexer and are described by their own state diagrams.

A Protocol Parser/Multiplexer has three service interfaces, each an instance of the ISS, as follows:

- d) One DownPort makes use of an instance of the ISS to pass data and control frames to and from lower layers in the interface stack.
- e) One DataPort offers an instance of the ISS to higher layers in the interface stack.
- f) One ControlPort offers an instance of the ISS to the functional element for the control protocol.

A specific instance of the Protocol Parser/Multiplexer, e.g., the LACP Parser/Multiplexer (6.2.9), ties these three notional service interfaces to specific service interfaces, e.g., those in 6.2.2, and provides a definition of the IsControlFrame function (6.1.3.1.2).

On transmission, the Protocol Multiplexer provides transparent pass-through of frames from the ControlPort or the DataPort to the DownPort.

On receipt of a frame from its DownPort, the Protocol Parser uses the IsControlFrame function to determine whether the frame is a control frame. It passes control frames up through the ControlPort and passes all other frames up through the DataPort. The Protocol Parser implements the function specified by the state diagram shown in Figure 6-2 and the associated definitions contained in 6.1.3.1.

6.1.3.1 Protocol Parser state diagram

6.1.3.1.1 Variables

DA
SA
mac_service_data_unit
priority
 The parameters of the M_UNITDATA.indication primitive.
msdu_type
 The type value of the mac_service_data_unit that is either an EtherType or an LLC Address, determined as described in Clause 9 of IEEE Std 802-2014.
BEGIN
 A Boolean variable that is set to TRUE when the System is initialized or reinitialized and is set to FALSE when (re)initialization has completed.
 Data Type: Boolean

6.1.3.1.2 Functions

IsControlFrame
 Given an input frame (DA, SA, mac_service_data_unit, priority), returns TRUE if the frame is a control frame for the ControlPort or FALSE if it is a data frame for the DataPort.
 Value returned: Boolean

6.1.3.1.3 Messages

ControlPort:M_UNITDATA.indication
DataPort:M_UNITDATA.indication
DownPort:M_UNITDATA.indication

The service primitives used to pass a received frame to a client with the specified parameters.

6.1.3.1.4 State diagram

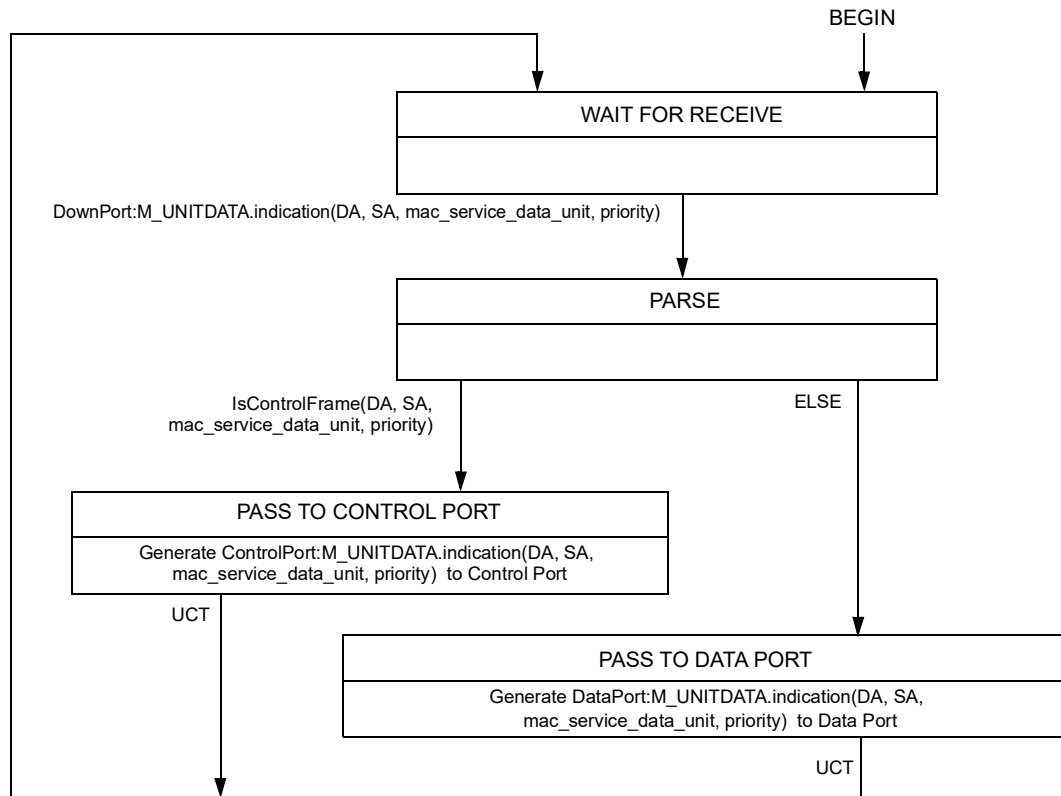


Figure 6-2—Protocol Parser state diagram

6.2 Link Aggregation operation

Figure 6-3 depicts the major blocks that form the Link Aggregation sublayer and their interrelationships. The Link Aggregation sublayer comprises the following functions:

- Frame Distribution.** This block is responsible for taking frames submitted by the Aggregator Client and submitting them for transmission on the appropriate Aggregation Port, based on a frame distribution algorithm employed by the Frame Distributor. Frame Distribution includes a Frame Distributor and an optional Marker Generator/Receiver used for the Marker protocol. (See 6.2.4, 6.2.5, and 6.5.)
- Frame Collection.** This block is responsible for passing frames received from the various Aggregation Ports to the Aggregator Client. Frame Collection includes a Frame Collector and a Marker Responder, used for the Marker protocol. (See 6.2.3, 6.2.6, and 6.5.)

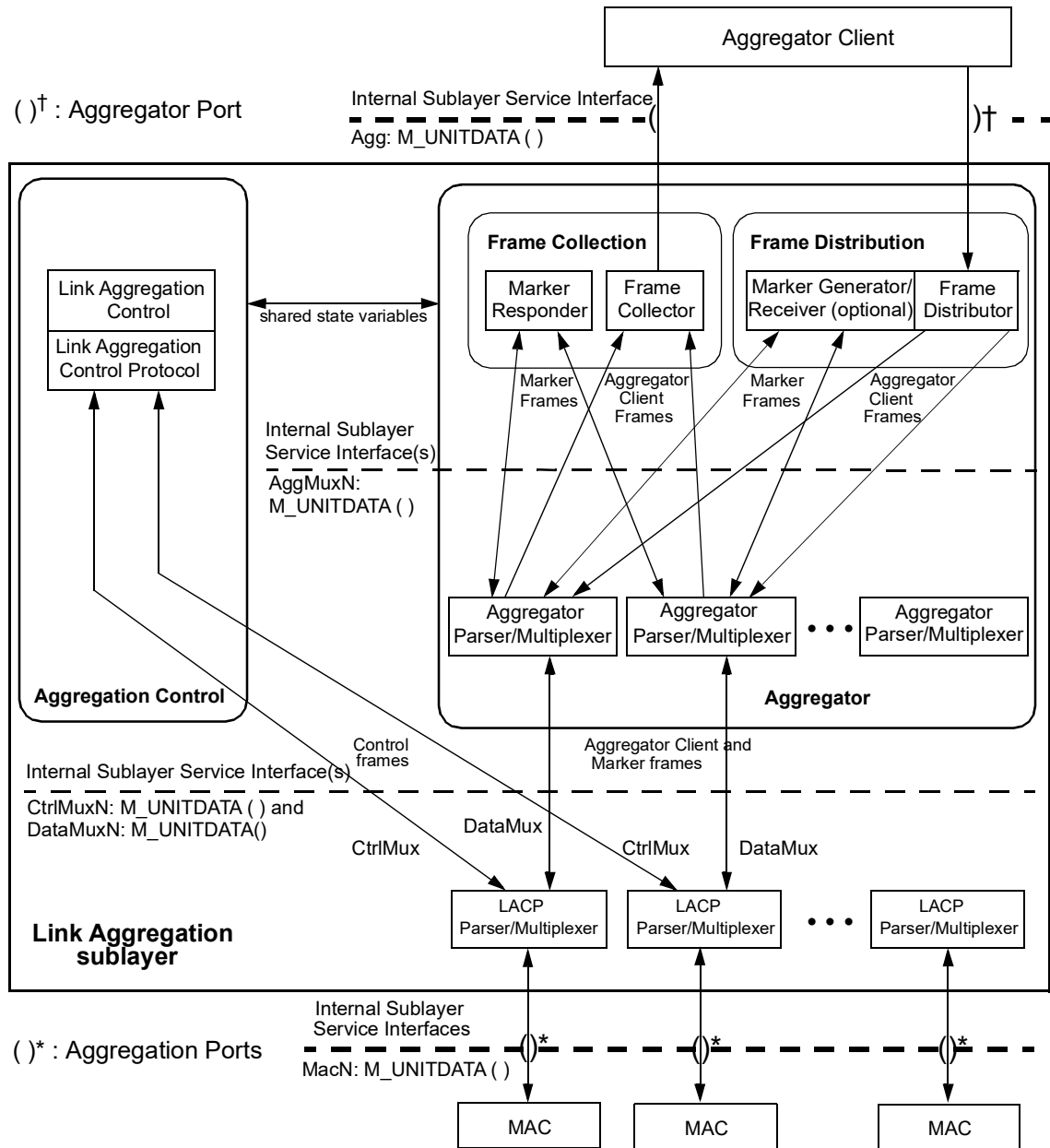


Figure 6-3—Link Aggregation sublayer block diagram

- c) **Aggregator Parser/Multiplexers.** On transmit, these entities simply pass frame transmission requests from the Frame Distributor, Marker Generator, and/or Marker Responder to the appropriate Aggregation Port. On receive, these entities distinguish among Marker Request, Marker Response, and MAC Protocol Data Units and pass each to the appropriate entity (Marker Responder, Marker Receiver, and Frame Collector, respectively).
- d) **Aggregator.** The combination of Frame Distribution and Frame Collection, along with the Aggregator Parser/Multiplexers, is referred to as the Aggregator.
- e) **Aggregation Control.** This block is responsible for the configuration and control of Link Aggregation. It incorporates a Link Aggregation Control Protocol (LACP) that can be used for automatic communication of aggregation capabilities between Systems and automatic configuration of Link Aggregation.

- f) **LACP Parser/Multiplexers.** On transmit, these entities simply pass frame transmission requests from the Aggregator and Control entities to the appropriate Aggregation Port. On receipt, these entities distinguish Link Aggregation Control PDUs from other frames, passing the LACPDUs to the appropriate sublayer entity, and all other frames to the Aggregator.

6.2.1 Principles of Link Aggregation

Link Aggregation allows an Aggregator Client to treat a set of one or more Aggregation Ports as if it were a single port. In doing so, it employs the following principles and concepts:

- a) An Aggregator Client communicates with a set of Aggregation Ports through an Aggregator, which presents a standard ISS interface to the Aggregator Client. One or more Aggregation Ports attach to an Aggregator within a System.
- b) It is the responsibility of the Aggregator to distribute frame transmissions from the Aggregator Client to the various Aggregation Ports, and to collect received frames from the Aggregation Ports and pass them to the Aggregator Client transparently.
- c) A System can contain multiple Aggregators, serving multiple Aggregator Clients. A given Aggregation Port will attach to (at most) a single Aggregator at any time. An Aggregator Client is served by a single Aggregator at a time.
- d) The attachment of Aggregation Ports to Aggregators within a System is managed by the Link Aggregation Control function for that System, which is responsible for determining which links can be aggregated, aggregating them, attaching the Aggregation Ports within the System to an appropriate Aggregator, and monitoring conditions to determine when a change in aggregation is needed.
- e) Such determination and attachment might be under manual control through direct manipulation of the state variables of Link Aggregation (e.g., Keys) by a network manager. In addition, automatic determination, configuration, attachment, and monitoring can occur through the use of a Link Aggregation Control Protocol (LACP). LACP uses peer exchanges across the links to determine, on an ongoing basis, the aggregation capability of the various links and continuously provides the maximum level of aggregation capability achievable between a given pair of Systems.
- f) Frame ordering has to be maintained for certain sequences of frame exchanges between Aggregator Clients (known as conversations, see Clause 3). The Frame Distributor ensures that all frames of a given conversation are passed to a single Aggregation Port. For any given Aggregation Port, the Frame Collector is required to pass frames to the Aggregator Client in the order that they are received from that Aggregation Port. The Frame Collector is otherwise free to select frames received from the Aggregation Ports in any order. Since there are no means for frames to be misordered on a single link, this guarantees that frame ordering is maintained for any conversation.
- g) Conversations can be moved among Aggregation Ports within an aggregation, both for load balancing and to maintain availability in the event of link failures. Frame ordering can be maintained (6.5 and 6.6) when conversations are moved.
- h) This standard does not require the use of a particular distribution algorithm.
- i) Each Aggregation Port is assigned a MAC address, unique over the LAG and the IEEE 802.1Q Bridged LAN (if any) to which the LAG is connected. This MAC address is used as the source address (SA) for frame exchanges that are initiated by entities within the Link Aggregation sublayer itself (i.e., LACP and Marker protocol exchanges).

NOTE—LACP and Marker Protocol use a multicast DA for all exchanges and do not require an Aggregation Port to recognize more than one unicast address on received frames.

- j) Each Aggregator is assigned a MAC address, unique over the LAG and the IEEE 802.1Q Bridged LAN (if any) to which the LAG is connected; this address is used as the MAC address of the aggregation from the perspective of the Aggregator Client, both as a SA for transmitted frames and as the destination address (DA) for received frames. The MAC address of the Aggregator can be one of the MAC addresses of an Aggregation Port in the associated LAG (see 6.2.10).

6.2.2 Service interfaces

The Aggregator Client communicates with the Aggregator using the ISS specified in IEEE Std 802.1AC. Similarly, Link Aggregation communicates internally (between Frame Collection/Distribution, the Aggregator Parser/Multiplexers, the LACP Parser/Multiplexers, and Link Aggregation Control) and with its attached Aggregation Ports using the same service interface. No new interlayer service interfaces are defined for Link Aggregation.

Since Link Aggregation uses several instances of the ISS, it is necessary to introduce a notation convention so that the reader can be clear about which interface is being referenced at any given time. A prefix is therefore assigned to each service primitive to indicate which of the interfaces is being invoked, as depicted in Figure 6-3. The prefixes are as follows:

- a) Agg:, for primitives issued on the interface between the Aggregator Client and the Link Aggregation sublayer.
- b) AggMuxN:, for primitives issued on the interface between Aggregator Parser/Multiplexer N and its internal clients (where N is the Port Number associated with the Aggregator Parser/Multiplexer).
- c) CtrlMuxN:, for primitives issued on the interface between LACP Parser/Multiplexer N and Link Aggregation Control (where N is the Port Number associated with the LACP Parser/Multiplexer).
- d) DataMuxN:, for primitives issued on the interface between LACP Parser/Multiplexer N and Aggregator Parser/Multiplexer N (where N is the Port Number associated with the LACP Parser/Multiplexer).
- e) MacN:, for primitives issued on the interface between underlying MAC N and its LACP Parser/Multiplexer (where N is the Port Number associated with the underlying MAC).

Aggregator Clients can generate Agg:M_UNITDATA.request primitives for transmission on an aggregated link. These are passed by the Frame Distributor to an Aggregation Port selected by the distribution algorithm. MacN:M_UNITDATA.indication primitives signifying received frames are passed unchanged from an Aggregation Port to the Aggregator Client by the Frame Collector.

In order to satisfy the needs of higher layers to access physical ports directly, the Aggregator Parser/Multiplexer (6.2.7) is provided. In particular, if IEEE Std 802.1AB [B1] is implemented in a System supporting Link Aggregation, the Link Aggregation operation (6.2) can attach zero or more instances of LLDP to each Aggregation Port.

6.2.3 Frame Collector

A Frame Collector is responsible for receiving incoming frames (i.e., AggMuxN:M_UNITDATA.indications) from the set of individual links that form the LAG (through each link's associated Aggregator Parser/Multiplexer) and delivering them to the Aggregator Client. Frames received from a given Aggregation Port are delivered to the Aggregator Client in the order that they are received by the Frame Collector. Since the Frame Distributor is responsible for maintaining any frame ordering constraints, there is no requirement for the Frame Collector to perform any reordering of frames received from multiple links.

The Frame Collector shall implement the function specified in the state diagram shown in Figure 6-4 and the associated definitions contained in 6.2.3.1.

6.2.3.1 Frame Collector state diagram

6.2.3.1.1 Constants

CollectorMaxDelay

In tens of microseconds, the maximum time that the Frame Collector can delay the delivery of a frame received from an Aggregator Parser to its Aggregator Client. Value is assigned by management or administration policy.

Data Type: Integer

Alias of aAggCollectorMaxDelay (7.3.1.1.32)

6.2.3.1.2 Variables

DA

SA

mac_service_data_unit

priority

The parameters of the M_UNITDATA.indication primitive.

BEGIN

A Boolean variable that is set to TRUE when the System is initialized or reinitialized and is set to FALSE when (re)initialization has completed.

Data Type: Boolean

6.2.3.1.3 Messages

Agg:M_UNITDATA.indication

AggMuxN:M_UNITDATA.indication

The service primitives used to pass a received frame to a client with the specified parameters.

6.2.3.1.4 State diagram

The architectural models of ISS and Link Aggregation do not make any provision for queuing of frames between the link and the Aggregator Client. However, practical implementations of Link Aggregation will typically incur both queuing and delay in the Frame Collector. In order to ensure that frame delivery is not delayed indefinitely (which could cause a frame ordering problem when moving conversations from one link to another), the Frame Collector shall, upon receiving a frame from an Aggregator Parser, either deliver the frame to its Aggregator Client, or discard the frame within a CollectorMaxDelay time. The Frame Distributor (within the Partner System at the other end of the link) can assume that all frames transmitted on a given link have been either received by its Partner's Aggregator Client or discarded after a CollectorMaxDelay plus the propagation delay of the link. The use of CollectorMaxDelay is further discussed in B.3.

NOTE—Because frame discard due to CollectorMaxDelay is a function of queuing and other entities not specified in this standard, the discard action is a requirement of the system, not just of the Frame Collector, and is therefore not shown in Figure 6-4.

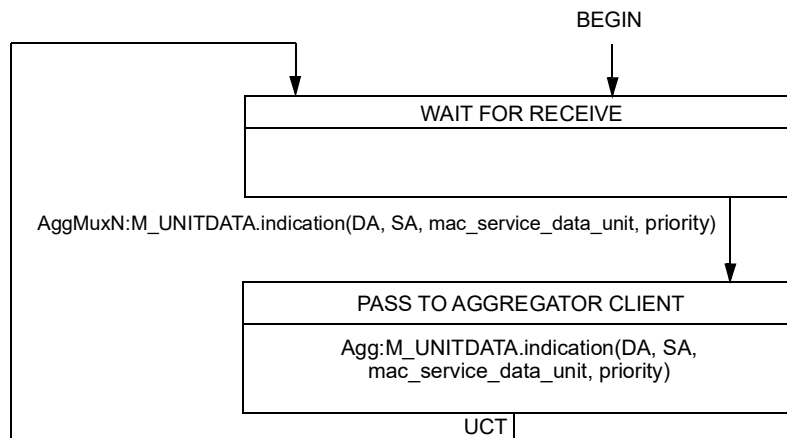


Figure 6-4—Frame Collector state diagram

6.2.4 Frame Distributor

The Frame Distributor is responsible for taking outgoing frames from the Aggregator Client and transmitting them through the set of links that form the LAG. The Frame Distributor implements a distribution function (algorithm) responsible for choosing the link to be used for the transmission of any given frame or set of frames.

This standard allows a wide variety of distribution algorithms. However, practical frame distribution algorithms do not misorder frames that are part of any given conversation, nor do they duplicate frames.

Rather, frame order is maintained by ensuring that all frames that compose a given conversation are transmitted on a single link in the order that they are generated by the Aggregator Client. No addition (or modification) of any information to the MAC frame is necessary, nor is any buffering or processing on the part of the corresponding Frame Collector in order to reorder frames. This approach permits a wide variety of distribution and load balancing algorithms to be used, while also ensuring interoperability between devices that adopt differing algorithms.

NOTE—Distribution algorithms and maintenance of frame ordering are discussed in Annex B.

Conversation-Sensitive Collection and Distribution (6.6) provides a structure for identifying various distribution algorithms, configuring the distribution algorithm to be used, and conveying that information to the LACP partner in LACP Version 2 TLVs (6.4.2.4). Per-Service Frame Distribution (Clause 8) specifies a set of distribution algorithms that can be used by a Link Aggregation implementation.

The Frame Distributor shall implement the function specified in the state diagram shown in Figure 6-5 and the associated definitions contained in 6.2.4.1.

6.2.4.1 Frame Distributor state diagram

6.2.4.1.1 Variables

DA
SA
mac_service_data_unit
priority

The parameters of the M_UNITDATA.request primitive.

BEGIN

A Boolean variable that is set to TRUE when the System is initialized or reinitialized and is set to FALSE when (re)initialization has completed.

Data Type: Boolean

6.2.4.1.2 Messages

Agg:M_UNITDATA.request

AggMuxN:M_UNITDATA.request

The service primitives used to transmit a frame with the specified parameters at the Aggregation Port, N, determined by the distribution algorithm. When Conversation-Sensitive Collection and Distribution is not supported, the algorithm used to determine the value of N is unspecified. When Conversation-Sensitive Collection and Distribution is supported, the algorithm used to determine the value of N is specified by the Actor_Port_Algorithm variable. It is possible that the value of N will be zero, indicating that the service primitive is to be discarded.

6.2.4.1.3 State diagram

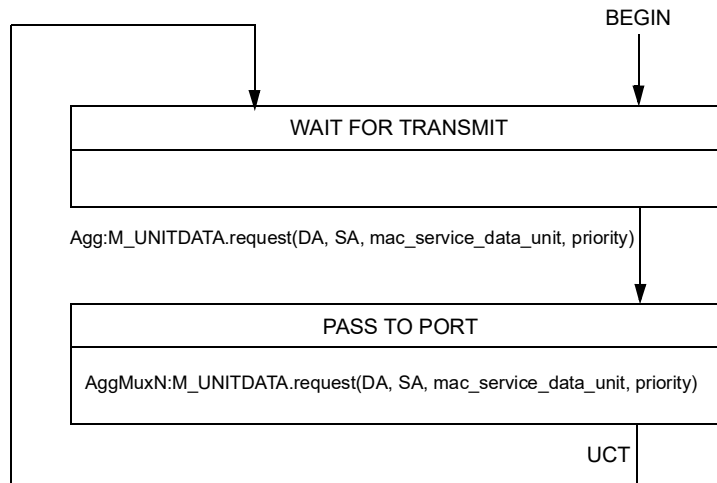


Figure 6-5—Frame Distributor state diagram

6.2.5 Marker Generator/Receiver (optional)

The optional Marker Generator is used by the Marker protocol, as specified in 6.5. When implemented and so requested by the distribution algorithm, the Marker Generator shall issue an AggMuxN:M_UNITDATA.request primitive, with a mac_service_data_unit containing a Marker PDU as defined in 6.5.3, to the Aggregation Port associated with the conversation being marked, subject to the timing restrictions for Slow Protocols specified in IEEE Std 802.3-2018, Annex 57A.

The optional Marker Receiver is used by the Marker protocol, as specified in 6.5. It receives Marker Response PDUs from the Aggregator Parser.

6.2.6 Marker Responder

The Marker Responder is used by the Marker protocol, as specified in 6.5. The Marker Responder receives Marker PDUs (generated by a Partner System's Marker Generator), and transmits a Marker Response PDU through the same Aggregation Port from which the Marker PDU was received. While implementation of the Marker Generator/Receiver is optional, the ability to respond to a Marker PDU (the Marker Responder) is mandatory. An implementation conformant to this clause that is not under a Distributed Relay component (DRNI Gateway, see 9.2, 9.5) shall implement the Marker Responder as specified in 6.5.4.2, thus ensuring that implementations that need to make use of the protocol can do so.

A Marker Responder that is under a DRNI Gateway is not required to respond to a Marker PDU.

NOTE—Since this standard lacks any means for coordinating Marker PDU responses among the DRNI Systems comprising a DRNI, it is safer to ignore a Marker PDU and allow the sender to time out, rather than to return a Marker PDU and run the risk of delivering frames out of order.

6.2.7 Aggregator Parser/Multiplexer

On transmission, the Aggregator Multiplexer shall provide transparent pass-through of frames submitted by the Marker Responder and optional Marker Generator to the Aggregation Port specified in the transmission request. The Aggregator Multiplexer shall provide transparent pass-through of frames submitted by the Frame Distributor to the Aggregation Port specified in the transmission request only when the Aggregation Port state is Distributing (see 6.4.13). Furthermore, if Conversation-Sensitive Collection and Distribution is supported, the value of the `Distribution_Conversation_Mask` for the `Port_Conversation_ID` associated with the frame is TRUE. Otherwise, such frames shall be discarded.

On receipt, the Aggregator Parser decodes frames received from the LACP Parser, passes those frames destined for the Marker Responder or Marker Receiver to the selected entity, and discards frames with invalid Slow Protocol subtype values (see IEEE Std 802.3-2018, Table 57A–2). The Aggregator Parser shall pass all other frames to the Frame Collector for passage to the Aggregator Client only when the Aggregation Port state is Collecting (see 6.4.13). Furthermore, if Conversation-Sensitive Collection and Distribution is supported, the value of the `Collection_Conversation_Mask` for the `Port_Conversation_ID` associated with the frame is TRUE. Otherwise, such frames shall be discarded. The Aggregator Parser shall implement the function specified in the state diagram shown in Figure 6-6 and the associated definitions contained in 6.2.7.1.

6.2.7.1 Aggregator Parser/Multiplexer state diagrams

6.2.7.1.1 Constants

`Slow_Protocols_Type`

The value of the Slow Protocols EtherType (Table 6-3).

`Marker_subtype`

The value of the Subtype field for the Marker protocol. (See 6.5.3.)

Data Type: Integer

Value: 2

`Marker_Information`

The encoding of the Marker Information TLV_Type field. (See 6.5.3.)

Data Type: Integer

Value: 1

`Marker_Response_Information`

The encoding of the Marker Response Information TLV_Type field. (See 6.5.3.)

Data Type: Integer

Value: 2

6.2.7.1.2 Variables

DA

SA

mac_service_data_unit

priority

The parameters of the M_UNITDATA.indication primitive.

Protocol_DA

The address determined by the setting of the aAggPortProtocolDA managed object (7.3.2.2.1).

A particular instance of link aggregation shall use the same DA for LACP as it does for the Marker protocol.

Data Type: 6 octets.

Length/Type

The value of the Length/Type field in a received frame.

Data Type: Integer

Subtype

The value of the octet following the Length/Type field in a Slow Protocol frame.

(See IEEE Std 802.3-2018, Annex 57A.)

Data Type: Integer

TLV_Type

The value contained in the octet following the Version Number in a received Marker or Marker Response frame. This identifies the “type” for the type/length/value (TLV) tuple. (See 6.5.3.)

Data Type: Integer

Collecting

An alias of Actor_Oper_Port_State.Collecting. When TRUE, the Aggregator Parser can pass frames to the Collector.

Data Type: Boolean.

Collection_Conversation_Mask

A Boolean vector, indexed by Port_Conversation_ID, that indicates whether the frame is allowed to reach the Aggregator when received through this Aggregation Port (TRUE = passes). When Conversation-Sensitive Collection and Distribution is not supported, this variable is not used (implicitly contains the value TRUE for all Port_Conversation_IDs). When Conversation-Sensitive Collection and Distribution is supported, the value is determined by the Update Mask machine (6.6.3.5).

Data Type: sequence of Boolean values, indexed by Port_Conversation_ID.

Alias of aAggPortOperConversationCollected (7.3.2.1.26).

Distributing

An alias of Actor_Oper_Port_State.Distributing. When TRUE, the Aggregator Multiplexer can pass frames to the Aggregation Port.

Data Type: Boolean.

Distribution_Conversation_Mask

A Boolean vector, indexed by Port_Conversation_ID, that indicates whether the M_UNITDATA.request is allowed to be passed to the Aggregation Port (TRUE = passes). When Conversation-Sensitive Collection and Distribution is not supported, this variable is not used (implicitly contains the value TRUE for all Port_Conversation_IDs). When Conversation-Sensitive Collection and Distribution is supported, the value is determined by the Update Mask machine (6.6.3.5).

Data Type: sequence of Boolean values, indexed by Port_Conversation_ID.

Alias of aAggPortOperConversationPasses (7.3.2.1.25).

Port_Conversation_ID

A Conversation Identifier (6.6.1) derived from the parameters of the service primitive. When Conversation-Sensitive Collection and Distribution is supported, the mechanism to derive the Conversation Identifier is specified by the Actor_Port_Algorithm (6.6.3.1).

Data Type: Integer

Value: 0 to 4095

BEGIN

A Boolean variable that is set to TRUE when the System is initialized or reinitialized and is set to FALSE when (re)initialization has completed.

Data Type: Boolean

6.2.7.1.3 Functions

DeterminePortConversationID

When Conversation-Sensitive Collection and Distribution is supported, this function determines a Port_Conversation_ID for the frame as specified by the Actor_Port_Algorithm variable using the mechanisms described in 6.6 and Clause 8.

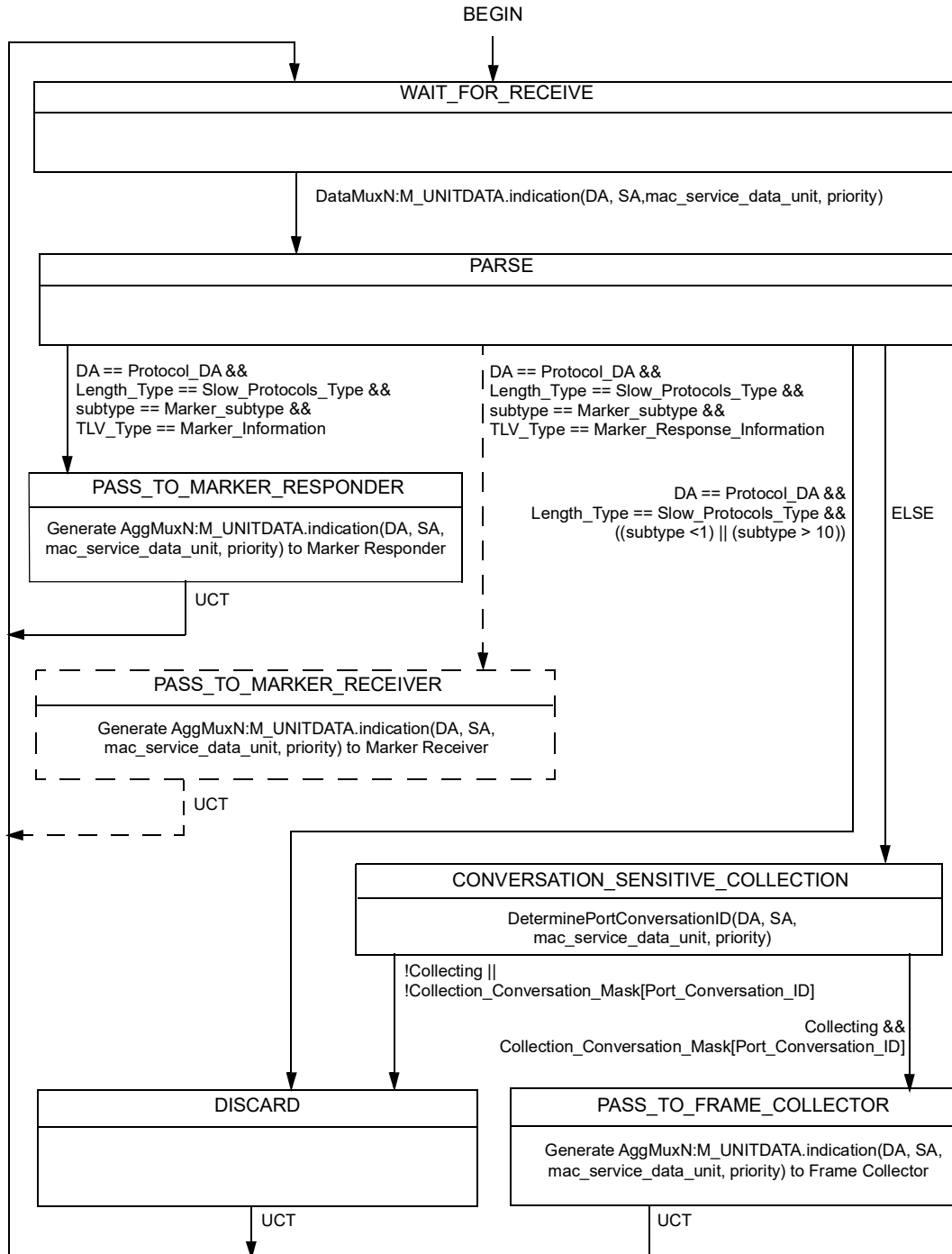
6.2.7.1.4 Messages

DataMuxN:M_UNITDATA.indication

AggMuxN:M_UNITDATA.indication

The service primitives used to pass a received frame to a client with the specified parameters.

6.2.7.1.5 State diagrams



If the optional Marker Receiver is not implemented, Marker Responses go from the PARSE state to the CONVERSATION_SENSITIVE_COLLECTION state.
If Conversation-Sensitive Collection and Distribution is not implemented, transitions from the CONVERSATION_SENSITIVE_COLLECTION state are based solely on the value of the Collecting variable.

Figure 6-6—Aggregator Parser state diagram

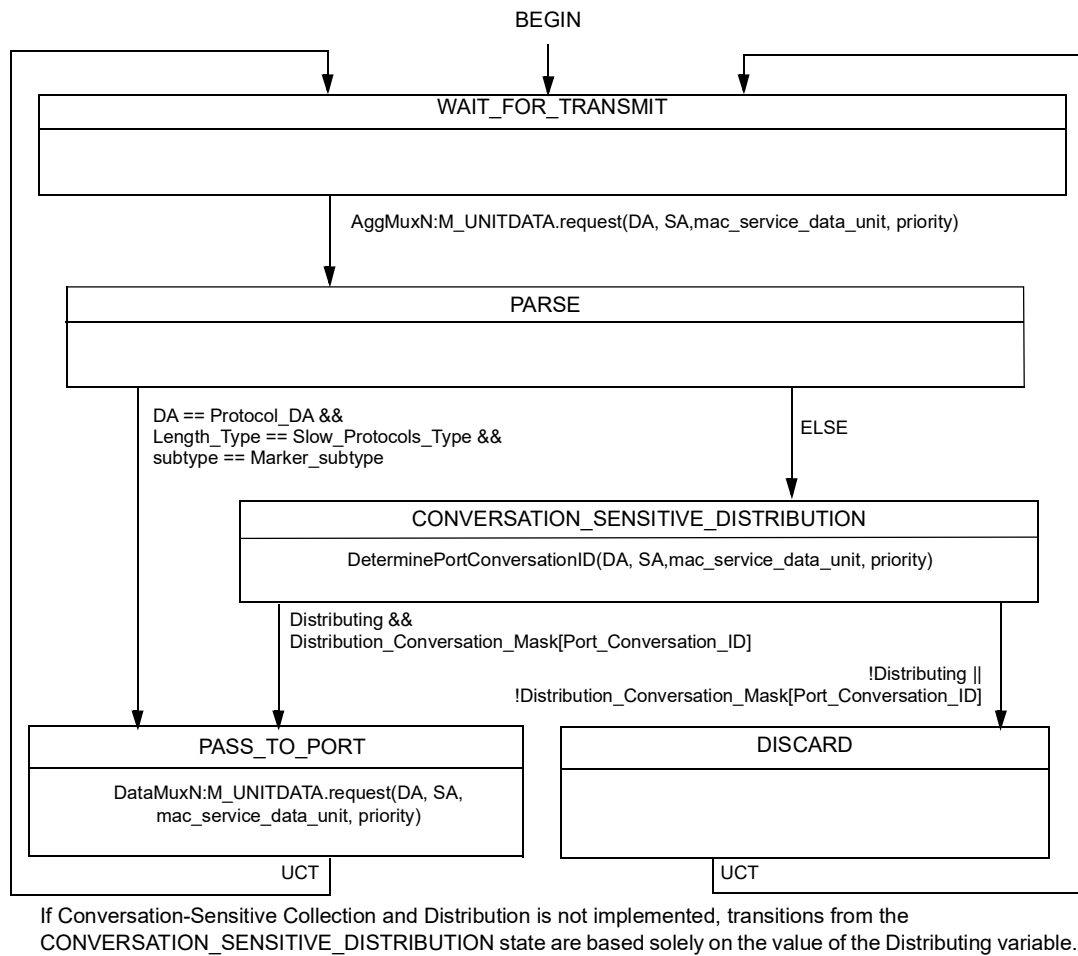


Figure 6-7—Aggregator Multiplexer state diagram

6.2.8 Aggregator

An Aggregator comprises an instance of a Frame Collection function, an instance of a Frame Distribution function, and one or more instances of the Aggregator Parser/Multiplexer function for a LAG. A single Aggregator is associated with each LAG. An Aggregator offers an ISS interface to its associated Aggregator Client; access to the physical media by an Aggregator Client is always achieved via an Aggregator.

A single individual MAC address is associated with each Aggregator (see 6.2.10).

If the following are all true, an Aggregator is available for use by the Aggregator Client, and MAC_Operational at the Aggregator ISS is TRUE (6.3.12):

- The Aggregator has one or more attached Aggregation Ports.
- The Aggregator has not been set to a disabled state by administrative action (see 7.3.1.1.13).
- One or more of the attached Aggregation Ports is Collecting and Distributing (see 7.3.1.1.14).

NOTE—To simplify the modeling and description of the operation of Link Aggregation, it is assumed that there are as many Aggregators as there are Aggregation Ports in a given System; however, this is not a requirement of this standard. Aggregation of two or more Aggregation Ports consists of changing the attachments between Aggregation Ports and Aggregators so that more than one Aggregation Port is attached to a single Aggregator. The creation of any

aggregations of two or more links therefore results in one or more Aggregators that are attached to more than one Aggregation Port and one or more Aggregators that are not attached to any Aggregation Port. An Aggregator that is not attached to any Aggregation Port appears to an Aggregator Client as a MAC interface to an inactive Aggregation Port. During times when the attachments between Aggregation Ports and Aggregators are changing, or as a consequence of particular configuration choices, there might be occasions when one or more Aggregation Ports are not attached to any Aggregator.

6.2.9 LACP Parser/Multiplexer

There is an LACP Parser/Multiplexer for each Aggregation Port. Each LACP Parser/Multiplexer is an instance of the Protocol Parser/Multiplexer described in 6.1.3. Specifically:

- a) The DownPort of the LACP Parser/Multiplexer is MacN (6.2.2) for LACP Parser/Multiplexer N.
- b) The ControlPort of the LACP Parser/Multiplexer is CtrlMuxN (6.2.2) for LACP Parser/Multiplexer N.
- c) The DataPort of the LACP Parser/Multiplexer is DataMuxN (6.2.2) for LACP Parser/Multiplexer N.
- d) The IsControlFrame function returns a value of TRUE if the DA of the service primitive is equal to the aAggPortProtocolDA (7.3.2.2.1) value, the msdu_type is the Slow Protocols EtherType (Table 6-3), the subtype is the LACP_subtype (IEEE Std 802.3-2018, Table 57A-3), and the length of the mac_service_data_unit is greater than or equal to 60 octets.

6.2.10 Addressing

Each IEEE 802 MAC has an associated individual MAC address regardless of whether that MAC is used for Link Aggregation.

Each Aggregator to which one or more Aggregation Ports are attached has an associated individual MAC address (see 6.3.3). The MAC address of the Aggregator can be the individual MAC address of one of the MACs in the associated LAG, or it can be a distinct MAC address. The manner in which such addresses are chosen is not otherwise constrained by this standard.

6.2.10.1 Source address (SA)

The source MAC address of LACPDUs and Marker protocol frames originating within the Link Aggregation sublayer is that of the Aggregation Port used for their transmission. The source MAC address of frames transmitted by the Aggregator Client is that of the Aggregator (for frames originating from the Aggregator Client) or that assigned by the originator of the frame (for frames bridged by the Aggregator Client).

NOTE—This behavior causes both the Aggregation Port and the Aggregator to behave the same way as a standard MAC with regard to frames submitted by their clients.

6.2.10.2 Destination address (DA)

The destination MAC address of LACPDUs and Marker protocol frames originating within the Link Aggregation sublayer is a group address chosen to limit the propagation of those frames to the desired peer. Appropriate addresses include the C-VLAN component reserved addresses (see Table 8-1 in IEEE Std 802.1Q-2018) and the S-VLAN component reserved addresses (see Table 8-2 in IEEE Std 802.1Q-2018). The choice of address used determines the scope of propagation of Link Aggregation control and marker PDUs within a Bridged LAN, as follows:

- a) The Nearest Customer Bridge group address is an address that no conformant C-VLAN component or MAC Bridge forwards. However, this address is relayed by Two-Port MAC Relay (TPMR) components and S-VLAN components. Therefore, Link Aggregation control, Distributed Relay

control, and marker PDUs received on this address represent information about stations that are not separated from the recipient station by any intervening C-VLAN components or MAC Bridges. There can, however, be TPMR components and/or S-VLAN components in the path between the originating and receiving stations.

NOTE 1—This address makes it possible to communicate information between end stations and/or Customer Bridges and for that communication to be transparent to the presence or absence of TPMRs or S-VLAN components in the communication path. The scope of this address is the same as that of a Customer-Bridge-to-Customer-Bridge MACsec connection.

- b) The Slow_Protocols_Multicast group address is an address that no conformant TPMR component, S-VLAN component, C-VLAN component, or MAC Bridge can forward. Link Aggregation control, Distributed Relay control, and marker PDUs transmitted using this DA can therefore travel no further than the stations that can be reached via a single individual LAN from the originating station. Link Aggregation control, Distributed Relay control, and marker PDUs received on this address therefore represent information about stations that are attached to the same individual LAN segment as the recipient station.

NOTE 2—This address makes it possible to transmit a Link Aggregation Control, DRCP, or Marker frame containing information specific to a single individual LAN, and for that information not to be propagated further than the extent of that individual LAN.

- c) The Nearest non-TPMR Bridge group MAC address is an address that no conformant C-VLAN component, S-VLAN component, or MAC Bridge can forward. However, this address is relayed by TPMR components. Therefore, Link Aggregation control, Distributed Relay control, and marker PDUs received on this address represent information about stations that are not separated from the recipient station by any intervening C-VLAN component, S-VLAN component, or MAC Bridge. There can, however, be one or more TPMR components in the path between the originating and receiving stations.

NOTE 3—This address makes it possible to communicate information between end stations and/or Bridges and for that communication to be transparent to the presence or absence of TPMRs in the transmission path. This address is primarily intended for use within Provider Bridged Networks.

The Slow_Protocols_Multicast group address is the default for LACP and Marker Protocol destination address. The use and encoding of the Slow_Protocols_Multicast group address is specified in Annex 57A of IEEE Std 802.3-2018.

NOTE 4—IEEE Std 802.1AX-2014 and IEEE Std 802.1AXbk-2012 restricted the LACP and Marker protocol destination addresses to those in Table 6-1. This restriction has been removed, thus permitting the use of other suitable address scopes without further revision of this standard.

Table 6-1—Link Aggregation protocol destination addresses

Assignment	Value
Nearest Customer Bridge group address	01-80-C2-00-00-00
IEEE 802.3 Slow_Protocols_Multicast group address	01-80-C2-00-00-02
Nearest non-TPMR Bridge group address	01-80-C2-00-00-03

6.3 Link Aggregation Control

Link Aggregation Control uses locally configured information and information exchanged by the Link Aggregation Control Protocol.

For each Aggregation Port in the System, Link Aggregation Control

- a) Maintains configuration information (reflecting the inherent properties of the individual links as well as those established by management) to control aggregation.
- b) Exchanges configuration information with other Systems to allocate the link to a LAG.
NOTE—A given link is allocated to, at most, one LAG at a time. The allocation mechanism attempts to maximize aggregation, subject to management controls.
- c) Attaches the Aggregation Port to the Aggregator used by the LAG, and detaches the Aggregation Port from the Aggregator when it is no longer used by the LAG.
- d) Enables and disables frame collection from the Aggregation Port by the Aggregator and distribution to the Aggregation Port by the Aggregator.

Link Aggregation Control uses the following identifiers to determine whether links provide connectivity to the same peer System, to determine whether the capabilities of the Aggregation Ports attached to those links are compatible with each other and with an available Aggregator, and to support management of LAGs and their associated Aggregators:

- e) A globally unique identifier for each System that participates in Link Aggregation (see 6.3.2).
- f) A locally unique identifier labeling a set of compatible capabilities.
- g) A means of identifying a LAG and its associated Aggregator.

System identification allows the detection of links that are connected in a loopback configuration (i.e., both ends of the same link are connected to the same System).

6.3.1 Characteristics of Link Aggregation Control

Link Aggregation Control provides a configuration capability that is the following:

- a) **Automatic.** In the absence of manual override controls, an appropriate set of LAGs is automatically configured, and individual links are allocated to those LAGs. If a set of links can aggregate, they do aggregate.
- b) **Continuous.** Manual intervention or initialization events are not a requirement for correct operation. The configuration mechanism continuously monitors for changes in state that require reconfiguration. The configuration functions detect and correct misconfigurations by performing reconfiguration and/or by taking misconfigured links out of service.
- c) **Deterministic.** The configuration can resolve deterministically; i.e., the configuration achieved can be made independent of the order in which events occur and be completely determined by the combination of the capabilities of the individual links and their physical connectivity.
- d) **Controllable.** The configuration capabilities accommodate devices with differing hardware and software constraints on Link Aggregation.
- e) **Compatible.** Links that cannot take part in Link Aggregation, either because of their inherent capabilities or of the capabilities of the devices to which they attach, operate as normal IEEE 802 links. The introduction of Link Aggregation capability at one or both ends of a link does not result in a degradation of the perceived performance of the link.
- f) **Rapid.** The configuration resolves rapidly to a stable configuration. Convergence can be achieved by the exchange of three LACPDUs, without dependence on timer values.

With the following:

- g) **Low risk of misdelivery.** The operation of the (re)configuration functions minimizes the risk of frames being delivered to the wrong Aggregator.

- h) **Low risk of duplication or misordering.** The operation of the (re)configuration functions minimizes the risk of frame duplication and frame misordering.
- i) **Low protocol overhead.** The overhead involved in external communication of configuration information between devices is small.

6.3.2 System identification

The System Identifier is the concatenation of the System Priority and the System MAC address. The MAC Address needs to be globally unique or at least unique among any set of Aggregation Systems with potential links to each other; otherwise, LACP can fail to create aggregations and/or can make improper connections between systems. The MAC address chosen can be the individual MAC address associated with one of the Aggregation Ports of the System. For DRNI (Clause 9), each DRNI System in a DRNI has a unique System Identifier, but the Aggregator Ports supporting the DRNI Gateway in each DRNI System have the same System Identifier, which is provided by the concatenation of the DRNI Aggregator Port's MAC address and the DRNI Aggregator Port's System Priority.

Where it is necessary to perform numerical comparisons between System Identifiers, each System Identifier is considered to be an eight-octet unsigned binary number, constructed as follows:

- a) The two most significant octets of the System Identifier comprise the System Priority. The System Priority value is taken to be an unsigned binary number; the most significant octet of the System Priority forms the most significant octet of the System Identifier.
- b) The third most significant octet of the System Identifier is derived from the initial octet of the MAC address; the least significant bit of the octet is assigned the value of the first bit of the MAC address, the next most significant bit of the octet is assigned the value of the next bit of the MAC address, and so on. The fourth through eighth octets are similarly assigned the second through sixth octets of the MAC address.

6.3.3 Aggregator identification

Each Aggregator to which one or more Aggregation Ports are attached shall be assigned a unique, globally or locally administered individual MAC address. The MAC address assigned to the Aggregator can be the same as the MAC address assigned to one of its attached Aggregation Ports. No Aggregator shall be assigned a MAC address that is the same as that of an Aggregation Port attached to a different Aggregator within the System. When receiving frames, an Aggregation Port is never required to recognize more than one unicast address, i.e., the Aggregator's MAC address.

When management is supported, an Aggregator shall be assigned an integer identifier that uniquely identifies the Aggregator within the System. This value will typically be the same as the interface identifier (ifIndex) used for the MIB (D.4.1).

6.3.4 Port identification

Link Aggregation Control uses a Port Identifier, comprising the concatenation of a Port Priority (7.3.2.1.15) and a Port Number (7.3.2.1.14), to identify an Aggregation Port. Port Numbers (and hence, Port Identifiers) shall be uniquely assigned within a System. Port Number 0 shall not be assigned to any Aggregation Port.

When it is necessary to perform numerical comparisons between Port Identifiers, each Port Identifier is considered to be a four-octet unsigned binary number constructed as follows:

- a) The most significant and second most significant octets are the first and second most significant octets of the Port Priority, respectively.
- b) The third and fourth most significant octets are the first and second most significant octets of the Port Number, respectively.

6.3.5 Capability identification

The ability of one Aggregation Port to aggregate with another is summarized by a simple integer parameter, known as a Key. This facilitates communication and comparison of aggregation capabilities, which are determined by a number of factors, including the following:

- a) The Aggregation Port's physical characteristics, such as point-to-point or shared medium.
- b) Configuration constraints established by the network administrator.
- c) Use of the Aggregation Port by higher layer protocols (e.g., assignment of Network Layer addresses).
- d) Characteristics or limitations of the Aggregation Port implementation itself.

Each Aggregation Port has an operational Key and an administrative Key. The operational Key is the Key that is currently in active use for the purposes of forming aggregations. The administrative Key allows manipulation of Key values by management. The administrative and operational Keys assigned to an Aggregation Port can differ under the following circumstances:

- e) When the operation of the implementation is such that an administrative change to a Key cannot be immediately reflected in the operational state of the Aggregation Port.
- f) When the System supports the dynamic manipulation of Keys, as discussed in 6.7.2, either to accurately reflect changes in operational capabilities of the Aggregation Port (for example, as a result of Auto-Negotiation) or to provide a means of handling constraints on aggregation capability.

A given Key is meaningful only in the context of the System that allocates it; there is no global significance to Key values. Similarly, the relationship between administrative and operational Keys is meaningful only in the context of the System that allocates it. When a System assigns an administrative Key to a set of Aggregation Ports, it signifies that those Aggregation Ports have the potential to aggregate together, subject to the considerations discussed in 6.7.2. When a System assigns an operational Key to a set of Aggregation Ports, it signifies that, in the absence of other constraints, any or all of those Aggregation Ports can be aggregated together from the perspective of the System making the assignment. The set of Aggregation Ports that actually aggregate together are those that terminate at a common Partner System and for which that Partner System has assigned a common operational Key, local to that Partner.

A System can determine that a given link is not able to be aggregated with other links. Such links are referred to as Solitary links (as opposed to Aggregateable links). A System can declare a link to be Solitary if the inherent properties of the link allow its use as part of an aggregation, but the system is aware of no other links that are capable of aggregating with this link (e.g., the System has allocated a unique operational Key to the link).

The capability information communicated between Systems, therefore, includes this local knowledge of the aggregation capability of the link in addition to the operational Key, i.e., whether the System considers the link to be Aggregateable or Solitary.

Each Aggregator has an operational Key and an administrative Key. The values of administrative and operational Key for an Aggregator can differ in the same manner as that of Aggregation Port Keys, per item e) and item f) in this subclause. Aggregation Ports with a given operational Key can be attached only to an Aggregator that has the same operational Key.

All Keys are 16-bit identifiers. All values except the null value (all zeros) are available for local use.

NOTE—This model allows for two convenient initial configurations. The first is achieved by assigning each Aggregation Port an initial administrative and operational Key value identical to its Port Number, and assigning the same Port Numbers as Keys to the corresponding Aggregators for each Aggregation Port. A device with this initial

configuration will bring up all links as Solitary, non-aggregated links. The second is achieved by assigning the same administrative and operational Key values to all Aggregation Ports with a common set of capabilities, and also to all Aggregators. A device with this initial configuration will attempt to aggregate together any set of links that have the same Partner System Identifier and operational Key, and for which both Systems are prepared to allow aggregation.

6.3.6 Link Aggregation Group identification

A Link Aggregation Group consists of either

- a) One or more Aggregateable links that terminate in the same pair of Systems (or DRNIs) and whose Aggregation Ports have the same operational Key, or
- b) A Solitary link.

6.3.6.1 Construction of the Link Aggregation Group Identifier

A unique Link Aggregation Group Identifier (LAG ID) is constructed from the following parameters for each of the communicating Systems:

- a) The operational System Identifier
- b) The operational Key assigned to the Aggregation Ports in the LAG
- c) The Port Identifier, if the link is identified as a Solitary link

A compound identifier formed from the System Identifiers and Keys alone is sufficient to identify a LAG comprising Aggregateable links. However, such an identifier is not sufficient for a LAG comprising a single Solitary link where the Partner System Identifier and operational Key can be zero. Even if these are nonzero, there can be multiple Solitary links with the same System Identifier and operational Key combinations, and it is necessary to include Port Identifiers to provide unique LAG IDs.

Given that

- d) S and T are System Identifiers,
- e) K and L are the operational Keys assigned to a LAG by S and T, respectively, and
- f) P and Q are the Port Identifiers of the Aggregation Ports being attached if the LAG comprises a single Solitary link and zero if the LAG comprises one or more Aggregateable links,

then the general form of the unique LAG ID is [(SKP), (TLQ)].

To simplify comparison of LAG IDs, it is conventional to order these so that S is the numerically smaller of S and T.

6.3.6.2 Representation of the Link Aggregation Group Identifier

To allow for convenient transcription and interpretation by human network personnel, this standard provides a convention for representing LAG IDs. Using this format,

- a) All fields are written as hexadecimal numbers, two digits per octet, in canonical format.
- b) Octets are presented in order, from left to right. Within fields carrying numerical significance (e.g., priority values), the most significant octet is presented first, and the least significant octet last.
- c) Within fields that carry MAC addresses, organizationally unique identifiers (OUIs) or Company Identifiers (CIDs), successive octets are separated by hyphens (-), in accordance with the hexadecimal representation for MAC addresses defined in IEEE Std 802.
- d) Parameters of the LAG ID are separated by commas.

For example, consider a link connecting two Aggregation Ports with the parameters shown in Table 6-2.

Table 6-2—Example Partner parameters

	Partner SKP	Partner TLQ
System Parameters (S, T)	System Priority = 0x8000 (see 6.4.2.3) System MACaddr= AC-DE-48-03-67-80	System Priority = 0x8000 (see 6.4.2.3) System MACaddr = AC-DE-48-03-FF-FF
Key Parameter (K, L)	Key = 0x0001	Key = 0x00AA
Aggregation Port Parameters (P, Q)	Port Priority = 0x80 (see 6.4.2.3) Port Number = 0x0002	Port Priority = 0x80 (see 6.4.2.3) Port Number = 0x0002

For a Solitary link, which cannot be aggregated with other links, the LAG ID derived from this information is represented as follows:

$[(SKP), (TLQ)] = [(8000, AC-DE-48-03-67-80, 0001, 80, 0002), (8000, AC-DE-48-03-FF-FF, 00AA, 80, 0002)]$

For a link that is Aggregateable, the LAG ID's Port Identifier components are zero, and the LAG ID derived from this information is represented as follows:

$[(SKP), (TLQ)] = [(8000, AC-DE-48-03-67-80, 0001, 00, 0000), (8000, AC-DE-48-03-FF-FF, 00AA, 00, 0000)]$

6.3.7 Selecting a Link Aggregation Group

Each Aggregation Port is selected for membership in the LAG uniquely identified by the LAG ID (composed of operational information, both derived from local administrative parameters and received through the Link Aggregation Control Protocol). Initial determination of the LAG ID is delayed to allow receipt of information from a peer Link Aggregation Control entity; in the event such information is not received, locally configured administrative defaults are assumed for the remote Aggregation Port's operational parameters.

Where a particular link is known to be Solitary, the complete LAG ID is not required to select the LAG since the link will not be aggregated with any other.

6.3.8 Agreeing on a Link Aggregation Group

Before frames are collected from and distributed to a link, both the local Link Aggregation Control entity and its remote peer (if present) need to agree on the LAG. The Link Aggregation Control Protocol allows each of the communicating entities to check its peer's current understanding of the LAG ID and facilitates rapid exchange of operational parameters while that understanding differs from its own.

The ability of LACP to signal that a particular link is Solitary (by clearing the Aggregation bit in the LACPDU *Actor_State* field; see 6.4.2.3) can accelerate the use of the link since, if both Link Aggregation Control entities know that the link is Solitary, agreement on the LAG ID is not necessary.