INTERNATIONAL STANDARD

ISO/ IEC/IEEE 8802-1CB

First edition
2019-02

AMENDMENT 2
2023-02

Information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks — Specific requirements —

Part 1CB: Frame replication and elimination for reliability

AMENDMENT 2: Extend stream identification functions

Technologies de l'information — Télécommunications et échange d'information entre systèmes — Réseaux locaux et métropolitains — Exigences spécifiques —

Partie 1CB: Duplication de trame et son élimination pour la fiabilité

AMENDEMENT 2: Fonctions d'identification de flux étendu

Reference number
ISO/IEC/IEEE 8802-1CB:2019/Amd.2:2023(E)

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

Email: stds.ipr@ieee.org
Website: www.ieee.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO/IEC documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

ISO/IEC/IEEE 8802-1CB:2019/Amd2 was prepared by the LAN/MAN of the IEEE Computer Society (as IEEE 802-1CBdb™-2021) and drafted in accordance with its editorial rules. It was adopted, under the "fast-track procedure" defined in the Partner Standards Development Organization cooperation agreement between ISO and IEEE, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

A list of all parts in the ISO/IEC/IEEE 8802 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

IEEE Std 802.1CBdb™-2021
(Amendment to IEEE Std 802.1CB™-2017
as amended by IEEE Std 802.1CBcv™-2021)

IEEE Standard for
Local and metropolitan area networks—

Frame Replication and Elimination for Reliability

Amendment 2: Extended Stream Identification Functions

Developed by the

LAN/MAN Standards Committee
of the
IEEE Computer Society

Approved 8 December 2021

IEEE SA Standards Board

Abstract: This amendment specifies procedures, managed objects, and protocols for bridges and end systems that provide identification and replication of packets for redundant transmission, identification of duplicate packets, and elimination of duplicate packets. It is not concerned with the creation of the multiple paths over which the duplicates are transmitted.

Keywords: amendment, Bridged Local Area Networks, Bridges, Bridging, Extended Stream identification, Frame Elimination, Frame Replication, IEEE 802®, IEEE 802.1CB™, IEEE 802.1CBdb™, IEEE 802.1Q™, local area networks (LANs), MAC Bridges, Redundancy, Stream identification, Time-Sensitive Networking, TSN, Virtual Bridged Local Area Networks (virtual LANs)

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

PDF: ISBN 978-1-5044-8305-6 STD25175
Print: ISBN 978-1-5044-8306-3 STDPD25175

Important Notices and Disclaimers Concerning IEEE Standards Documents

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

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

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

IEEE makes no warranties or representations concerning its standards, and expressly disclaims all warranties, express or implied, concerning this standard, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement. In addition, IEEE does not warrant or represent that the use of the material contained in its standards is free from patent infringement. IEEE standards documents are supplied "AS IS" and "WITH ALL FAULTS."

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

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

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

Translations

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

Official statements

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

Comments on standards

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

Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its Societies and Standards Coordinating Committees are not able to provide an instant response to comments, or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in evaluating comments or in revisions to an IEEE standard is welcome to join the relevant IEEE working group. You can indicate interest in a working group using the Interests tab in the Manage Profile & Interests area of the IEEE SA myProject system.[1] An IEEE Account is needed to access the application.

Comments on standards should be submitted using the Contact Us form.[2]

Laws and regulations

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

Data privacy

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

1. Available at: https://development.standards.ieee.org/myproject-web/public/view.html#landing.
2. Available at: https://standards.ieee.org/content/ieee-standards/en/about/contact/index.html.

Copyrights

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

Photocopies

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

Updating of IEEE Standards documents

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

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

In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit IEEE Xplore or contact IEEE.[3] For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website.

Errata

Errata, if any, for all IEEE standards can be accessed on the IEEE SA Website.[4] Search for standard number and year of approval to access the web page of the published standard. Errata links are located under the Additional Resources Details section. Errata are also available in IEEE Xplore. Users are encouraged to periodically check for errata.

Patents

IEEE Standards are developed in compliance with the IEEE SA Patent Policy.[5]

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

3. Available at: https://ieeexplore.ieee.org/browse/standards/collection/ieee.
4. Available at: https://standards.ieee.org/standard/index.html.
5. Available at: https://standards.ieee.org/about/sasb/patcom/materials.html.

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.

IMPORTANT NOTICE

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

Participants

At the time this amendment was submitted to the IEEE SA Standards Board for approval, the IEEE 802.1 Working Group had the following membership:

Glenn Parsons, *Chair*
Jessy Rouyer, *Vice-Chair*
János Farkas, *Chair, Time-Sensitive Networking Task Group*
Christophe Mangin, *Editor*

Astrit Ademaj
Ralf Assmann
Rudy Belliardi
Christian Boiger
Paul Bottorff
Radhakrishna Canchi
David Chen
Feng Chen
Paul Congdon
Rodney Cummings
Josef Dorr
Hesham Elbakoury
Anna Engelmann
Thomas Enzinger
Donald Fedyk
Norman Finn
Geoffrey Garner
Amrit Gopal
Craig Gunther
Marina Gutierrez
Stephen Haddock
Mark Hantel
Jerome Henry
Marc Holness
Daniel Hopf
Woojung Huh
Satoko Itaya
Yoshihiro Ito
Michael Karl
Stephan Kehrer
Randy Kelsey
Marcel Kiessling
Gavin Lai
James Lawlis
Joao Lopes
Lily Lv
Scott Mansfield
Kenichi Maruhashi
Olaf Mater
David McCall
Larry McMillan
John Messenger
Hiroki Nakano
Bob Noseworthy
Hiroshi Ohue
Donald R. Pannell
Michael Potts
Dieter Proell
Wei Qiu
Karen Randall
Maximilian Riegel
Silvana Rodrigues
Atsushi Sato
Frank Schewe
Michael Seaman
Maik Seewald
Ramesh Sivakolundu
Johannes Specht
Marius Stanica
Guenter Steindl
Liyang Sun
Karim Traore
Max Turner
Balazs Varga
Ganesh Venkatesan
Tongtong Wang
Xinyuan Wang
Karl Weber
Ludwig Winkel
Jordon Woods
Takahiro Yamaura
Yue Yin
Nader Zein
William Zhao
Helge Zinner

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

Thomas Alexander
Harry Bims
Christian Boiger
Vern Brethour
William Byrd
Paul Cardinal
Pin Chang
Diego Chiozzi
Janos Farkas
Avraham Freedman
Craig Gunther
Stephen Haddock
Marco Hernandez
Werner Hoelzl
Oliver Holland
Pranav Jha
Lokesh Kabra
Piotr Karocki
Stephan Kehrer
Randy Kelsey
Stuart Kerry
Evgeny Khorov
Yongbum Kim
Hyeong Ho Lee
Ting Li
Christophe Mangin
Scott Mansfield
Jonathon Mclendon
Satoshi Obara
Glenn Parsons
Bansi Patel
Arumugam Paventhan
Clinton Powell
Dieter Proell
Alon Regev
Maximilian Riegel
Jessy Rouyer
Frank Schewe
Michael Seaman
Eugene Stoudenmire
Walter Struppler
Mitsutoshi Sugawara
Bo Sun
Max Turner
John Vergis
Stephen Webb
Karl Weber
Scott Willy
Yu Yuan
Oren Yuen

When the IEEE SA Standards Board approved this standard on 8 December 2021, it had the following membership:

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

Edward A. Addy	Howard Li	Mehmet Ulema
Doug Edwards	Daozhuang Lin	Lei Wang
Ramy Ahmed Fathy	Kevin Lu	F. Keith Waters
J. Travis Griffith	Daleep C. Mohla	Karl Weber
Thomas Koshy	Chenhui Niu	Sha Wei
Joseph L. Koepfinger*	Damir Novosel	Howard Wolfman
David J. Law	Annette Reilly	Daidi Zhong
	Dorothy Stanley	

*Member Emeritus

Introduction

This introduction is not part of IEEE Std 802.1CBdb-2021, IEEE Standard for Local and metropolitan area networks—Frame Replication and Elimination for Reliability—Amendment 2: Extended Stream Identification Functions.

This Standard defines Extended Stream identification functions.

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 can be obtained from

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

Contents

3. Definitions 14

5. Conformance 15

5.4 Stream identification component recommended behavior 15
5.5 Stream identification component optional behaviors 15
5.7 Talker end system recommended behaviors 15
5.8 Talker end system optional behaviors 15
5.10 Listener end system recommended behavior 15
5.11 Listener end system optional behaviors 16
5.13 Relay system recommended behaviors 16
5.15 FRER C-component required and optional behaviors 16

6. Stream identification 17

6.1 Stream service subparameters 18
6.2 Stream identification function 18
6.3 Stream identification in systems 18
6.4 Null Stream identification 18
6.5 Source MAC and VLAN Stream identification 18
6.6 Active Destination MAC and VLAN Stream identification 19
6.7 IP Stream identification 19
6.8 Mask-and-match Stream identification 19

7. Frame Replication and Elimination for Reliability 21

7.4 Sequencing function 21
7.4.1 Sequence generation function 21
7.4.1.3 SequenceGenerationReset 21
7.4.3 Base recovery function 21
7.4.3.3 SequenceRecoveryReset 21
7.4.3.4 VectorRecoveryAlgorithm 21
7.4.3.5 MatchRecoveryAlgorithm 24
7.8 Redundancy tag 25
7.9 HSR sequence tag 25
7.10 PRP sequence trailer 25

8. Frame Replication and Elimination for Reliability in Bridges 26

8.1 Limiting options 26
8.2 FRER C-component input transformations 26
8.3 Frame Replication and Elimination for Reliability and VLAN tags 27

9. Stream Identification Management 29

9.1 Stream identity table 29
9.1.1 tsnStreamIdEntry 29
9.1.1.6 tsnStreamIdIdentificationType 29
9.1.4 Managed objects for Active Destination MAC and VLAN Stream identifications 29
9.1.5 Managed objects for IP Stream identification 29
9.1.5.1 tsnCpeIpIdDestMac 29
9.1.6 Managed objects for Mask-and-match Stream identification 30
9.1.6.1 tsnCpeMmIdDestMacMask 30
9.1.6.2 tsnCpeMmIdDestMacMatch 30
9.1.6.3 tsnCpeMmIdSrcMacMask 30

9.1.6.4 tsnCpeMmIdSrcMacMatch 30
9.1.6.5 tsnCpeMmIdMsduMaskLength 30
9.1.6.6 tsnCpeMmIdMsduMask 30
9.1.6.7 tsnCpeMmIdMsduMatch 31
9.2 Operational per-port per-Stream Stream identification counters 31
9.3 Operational per-port Stream identification counters 31
9.4 Per-port managed object 31
9.4.1 tsnPpMmIdMsduMaskMaxLength 31

10. Frame Replication and Elimination for Reliability management 32

10.2 Additional tsnStreamIdEntry managed objects 32
10.4 Sequence recovery table 32
10.4.1 frerSeqRcvyEntry 32
10.4.1.4 frerSeqRcvyReset 32
10.4.1.10 frerSeqRcvyIndividualRecovery 32
10.8 Operational per-port and per-Stream FRER counters 32
10.8.8 frerCpsSeqRcvyTaglessPackets 32
10.9 Operational per-port FRER counters 33

11. Management Information Base (MIB) 34

11.2 Structure of the MIB 34
11.2.1 Structure of the IEEE8021-STREAM-IDENTIFICATION-MIB 36
11.5 MIB modules, ... 38
11.5.1 Definitions for the IEEE8021-STREAM-IDENTIFICATION-MIB 38

12. YANG Data Model 70

12.2 IEEE Std 802.1CB YANG model 70
12.2.1 Stream identification model 70
12.3 Structure of the YANG model 71
12.3.1a Structure of the ieee802-dot1cb-mask-and-match YANG module 71
12.4 Relationship to other YANG modules 72
12.4.6a IEEE 802.1CB Extended Stream identification module 72
12.6 Definition of 802.1CB YANG modules, ... 72
12.6.1 YANG data scheme tree definitions 72
12.6.1.1a YANG data scheme definition for ieee802-dot1cb-mask-and-match YANG module 72
12.6.2 YANG data module definitions 73
12.6.2.1 Definition for the ieee802-dot1cb-stream-identification-types YANG module 73
12.6.2.2a Definition for the ieee802-dot1cb-mask-and-match module 75

Annex A (normative) Protocol Implementation Conformance Statement (PICS) proforma 80

A.2 PICS proforma for Frame Replication and Elimination for Reliability 80
A.2.2 Stream identification component 80
A.2.3 Talker end system 81
A.2.4 Listener end system 82
A.2.5 Relay system 83
A.2.9 YANG 83

Annex C (informative) Frame Replication and Elimination for Reliability in systems 84

C.2 Example 2: Various stack positions 84
C.5 Example 5: Protocol interworking 85
C.9 FRER and reserved bandwidth 85
C.10 Use of the Individual recovery function 85

Annex C1 (informative) Mask-and-match Stream identification function 86

C1.1 Example 1: Generic Stream identification for industrial communication networks 86
C1.2 Example 2: MPLS DetNet flow identification 86

Annex D (informative) Bibliography 88

IEEE Standard for
Local and metropolitan area networks—

Frame Replication and Elimination for Reliability

Amendment 2: Extended Stream Identification Functions

(This amendment is based on IEEE Std 802.1CB™-2017 as amended by IEEE Std 802.1CBcv™-2021.)

NOTE—The editing instructions contained in this amendment define how to merge the material contained therein into the existing base standard and its amendments to form the comprehensive standard.

The editing instructions are shown in ***bold italic***. Four editing instructions are used: change, delete, insert, and replace. ***Change*** is used to make corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed by using ~~strikethrough~~ (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Deletions and insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make changes in figures or equations by removing the existing figure or equation and replacing it with a new one. Editing instructions, change markings, and this NOTE will not be carried over into future editions because the changes will be incorporated into the base standard.[6]

[6] Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.

3. Definitions

Change the definition of "Internal Sublayer Service (ISS)" in Clause 3 as follows:

Internal Sublayer Service (ISS): An augmented version of the MAC Service, defined in ~~6.6 of IEEE Std 802.1Q-2014~~IEEE Std 802.1AC.

Insert the following new definition Clause 3 in alphabetical order:

Mask: A set of consecutive bits used to select bits of a particular significance in a bit field of the same length as the mask.

5. Conformance

5.4 Stream identification component recommended behavior

Change the lettered list in 5.4 as follows:

a) A Source MAC and VLAN Stream identification function (6.5);
b) ~~a)~~ An Active Destination MAC and VLAN Stream identification function (6.6).

5.5 Stream identification component optional behaviors

Change the lettered list in 5.5 as follows:

a) The items in 5.3 and 5.4 on more than one port;
b) The items in 5.3 and 5.4 for some number of Compound Streams greater than 1;
c) An IP Stream identification function (6.7); ~~and/or~~
d) A Mask-and-match Stream identification function (6.8); and/or
e) ~~d)~~ Additional types of Stream identification functions.

5.7 Talker end system recommended behaviors

Change the lettered list in 5.7 as follows:

a) A Source MAC and VLAN Stream identification function (6.5);
b) ~~a)~~ An Active Destination MAC and VLAN Stream identification function (6.6); and
c) ~~b)~~ A Stream splitting function (7.7).

5.8 Talker end system optional behaviors

Change the lettered list in 5.8 (as changed by IEEE Std 802.1CBcv-2021) as follows:

a) The items in 5.6 and 5.7 on more than one port;
b) The items in 5.6 and 5.7 for some number of Compound Streams greater than 1;
c) An IP Stream identification function (6.7);
d) A Mask-and-match Stream identification function (6.8);
e) ~~d)~~ Additional types of Stream identification functions;
f) ~~e)~~ The HSR sequence tag (7.9);
g) ~~f)~~ The PRP sequence trailer (7.10);
h) ~~g)~~ Additional types of Sequence encode/decode functions;
i) ~~h)~~ Support SMIv2 MIB modules for the management of talker end system functions (Clause 11) as required by item e in 5.6; and/or
j) ~~i)~~ Support YANG modules for the management of talker end system functions (Clause 12) as required by item e in 5.6.

5.10 Listener end system recommended behavior

Change the lettered list in 5.10 as follows:

a) A Source MAC and VLAN Stream identification function (6.5);
b) ~~a)~~ An Active Destination MAC and VLAN Stream identification function (6.6).

5.11 Listener end system optional behaviors

Change the lettered list in 5.11 (as changed by IEEE Std 802.1CBcv-2021) as follows:

a) The items in 5.9 and 5.10 on more than one port;

b) The items in 5.9 and 5.10 for some number of Compound Streams greater than 1;

c) An IP Stream identification function (6.7);

d) A Mask-and-match Stream identification function (6.8);

e) ~~d)~~ Additional types of Stream identification functions;

f) ~~e)~~ The HSR sequence tag (7.9);

g) ~~f)~~ The PRP sequence trailer (7.10);

h) ~~g)~~ Additional types of Sequence encode/decode functions;

i) ~~h)~~ At least two instances of Individual recovery functions (7.5), each using the VectorRecoveryAlgorithm (7.4.3.4);

j) ~~i)~~ Support SMIv2 MIB modules for the management of listener end system functions (Clause 11) as required by item f in 5.9; and/or

k) ~~j)~~ Support YANG modules for the management of listener end system functions (Clause 12) as required by item f in 5.9.

5.13 Relay system recommended behaviors

Change 5.13 as follows:

A relay system should be able to instantiate the following in-facing functions on at least two ports, for both transmit and receive, for at least one Stream:

a) A Source MAC and VLAN Stream identification function (6.5);

b) ~~a)~~ Active Destination MAC and VLAN Stream identification functions (6.6) for encoding and decoding packets; ~~and~~

c) Either of the following:

   1) ~~b)~~ IP Stream identification functions (6.7);

   2) Mask-and-match Stream identification function (6.8) for identifying packets.

NOTE—IP Stream identification and Mask-and-match Stream identification enable~~s~~ a relay system to proxy for a FRER-unaware end system.

5.15 FRER C-component required and optional behaviors

Change list item a) in 5.15 as follows:

a) Shall meet all of the required and any or none of the optional behaviors for an IEEE 802.1Q C-VLAN component (5.5 of IEEE Std 802.1Q-~~2014~~2018).

6. Stream identification

Change the first paragraph of the introduction of Clause 6 as follows:

Clause 7 of IEEE Std 802.1AC-2016 describes the IEEE 802.1 layering model, that Frame Replication and Elimination for Reliability (FRER) follows. Stream identification utilizes a single Service Access Point (SAP) to a connectionless packet service offered by the layer below it [e.g., the Intermediate Sublayer Service (ISS) of Clause 11 of IEEE Std 802.1AC-2016, or the Enhanced Internal Sublayer Service (EISS) of 6.8 of IEEE Std 802.1Q-2018], and offers an array of SAPs to the layers above it, corresponding to different Streams. The Stream identification model is illustrated in Figure 6-1.

Change the note after the second paragraph of the introduction of Clause 6 as follows:

NOTE—In principle, any number of different methods for identifying and encoding Streams can be defined. Several ~~required~~ methods are specified in the following subclauses (6.4, 6.5, 6.6, 6.7, and 6.8).

Change item c) in the lettered list in the introduction of Clause 6 as follows:

c) ~~Four~~Five specific Stream identification functions are described: Null Stream identification (6.4), Source MAC and VLAN Stream identification (6.5), Active Destination MAC and VLAN Stream identification (6.6), ~~and~~ IP Stream identification (6.7), and Mask-and-match Stream identification (6.8).

Change Table 6-1 as follows:

Table 6-1—Stream identification functions

Stream identification function	Active/passive	Examines	Overwrites	Reference
Null Stream identification	Passive	destination_address, vlan_identifier	None	6.4, 9.1.2
Source MAC and VLAN Stream identification	Passive	source_address, vlan_identifier	None	6.5, 9.1.3
Active Destination MAC and VLAN Stream identification	Active	destination_address, vlan_identifier	destination_address, vlan_identifier, priority	6.6, 9.1.4
IP Stream identification	Passive	destination_address, vlan_identifier, IP source address, IP destination address, DSCP, IP next protocol, source port, destination port	None	6.7, 9.1.5
Mask-and-match Stream identification function	Passive	destination_address, source_address, mac_service_data_unit	None	6.8, 9.1.6

6.1 Stream service subparameters

Change the first paragraph of 6.1 as follows:

The ISS defined in IEEE Std 802.1AC and the EISS defined in IEEE Std 802.1Q includes a connection_identifier parameter that is of local significance (to a system) only. The parameter is not carried across the underlying service. Stream identification makes use of this parameter to carry parametrized information. Stream identification has need for more than one subparameter, but an implementor can create mathematical algorithms to combine those subparameters (and/or other subparameters for other layers) into a single connection_identifier parameter, especially since the connection_identifier's values are undefined outside the system implementing them. In this document, parameters that are assumed to be encoded in the connection_identifier are deemed *subparameters*.

6.2 Stream identification function

Change the note at the end of 6.2 into two notes as follows:

NOTE 1—The Stream identification method does not necessarily involve an actual transformation of the packet. See 6.4 and 6.6 for examples of both cases.

NOTE 2—The order of precedence of Stream identification functions and the invocation of their instances, when more than one matches the same frame, is out of scope of this standard.

6.3 Stream identification in systems

Change the third sentence of the first paragraph of 6.3 as follows:

Diagram a in Figure 6-5 illustrates a relay system (e.g., 8.6 of IEEE Std 802.1Q-~~2014~~2018) with two ports, that has its FRER capabilities embedded in its forwarding function, and not shown explicitly in the diagram.

6.4 Null Stream identification

Change the second sentence of the first paragraph of 6.4 as follows:

It can be defined using the Enhanced Internal Sublayer Service (EISS) described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, in which case it is enhanced with the extra stream_handle subparameter of the connection_identifier, specified in 6.1 of the present standard.

6.5 Source MAC and VLAN Stream identification

Change the second sentence of the first paragraph of 6.5 as follows:

It can be defined using the EISS described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, in which case it is enhanced with the extra stream_handle subparameter of the connection_identifier, specified in 6.1 of the present standard.

6.6 Active Destination MAC and VLAN Stream identification

Change the second sentence of the first paragraph of 6.6 as follows:

It can be defined using the EISS described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, in which case it is enhanced with the extra stream_handle subparameter of the connection_identifier, specified in 6.1 of the present standard.

Change the second paragraph of 6.6 as follows:

In the Active Destination MAC and VLAN Stream identification, the destination_address, vlan_identifier, and priority parameters of the frame passed down the stack from the upper layers or up the stack from the lower layers are replaced with alternate values. The replacement values for frames transmitted down the stack to the Active Destination MAC and VLAN Stream identification, and used to recognize frames passed up the stack to the Active Destination MAC and VLAN Stream identification function, ~~are those listed in 9.1.2~~as well as . ~~T~~the replacement values for frames passed up the stack ~~(not including the priority parameter)~~ are described in 9.1.4. The replacement values for frames transmitted down the stack to the Active Destination MAC and VLAN Stream identification function are determined by the stream_handle subparameter passed down in the EISS request's connection_identifier.

6.7 IP Stream identification

Change the second sentence of the first paragraph of 6.7 as follows:

It can be defined using the union of the IP address primitives listed in 9.1.5 and parameters defined by the EISS described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, in which case it is enhanced with the stream_handle subparameter of the connection_identifier, specified in 6.1 of the present standard.

Insert new subclause 6.8 after 6.7 as follows:

6.8 Mask-and-match Stream identification

The Mask-and-match Stream identification is a passive Stream identification that operates at the frame level. It can be defined using the Internal Sublayer Service (ISS) described in IEEE Std 802.1AC, in which case it is enhanced with the extra stream_handle subparameter of the connection_identifier, specified in 6.1 of the present standard.

It discards the stream_handle subparameter passed down the stack. It generates a stream_handle subparameter on frames passed up the stack based on information extracted from a subset of the frame's ISS parameters. It does not change any of a packet's parameters, other than the connection_identifier.

Input information for the Mask-and-match Stream identification function is obtained by masking one or more of the following ISS parameters:

— Source_address,
— Destination_address, and
— The first *N* octets of mac_service_data_unit, *N* being a system-dependent variable.

Stream identification is then done by comparing these masked ISS parameters with specific match values.

The Mask-and-match Stream identification function supports Stream identification schemes ranging from very simple, using a single mask applied to one of its input ISS parameters, to complex schemes involving

masks applied to all input ISS parameters, and in particular a specific mac_service_data_unit mask to identify communication flows supported by various higher-layer protocols.

In order to instantiate the Mask-and-match Stream identification function, the tsnStreamIdIdentificationType managed object (9.1.1.6) is encoded using the OUI (00-80-C2) and the type values as shown in Table 9-1.

The managed objects for Mask-and-match Stream identification are described in 9.1.6.

Mask-and-match Stream identification can be coupled, for example, with Active Destination MAC and VLAN Stream identification (6.6) to assign a particular {MAC address, VLAN, priority} triplet to packets belonging to a particular unicast application Stream identified by the combination of the values of particular fields in the MAC service data unit, as shown in Figure 8-1, Port A, where Mask-and-match Stream identification would be in the box labeled “Passive Upper Stream identification functions (6.2).”

7. Frame Replication and Elimination for Reliability

7.4 Sequencing function

7.4.1 Sequence generation function

7.4.1.3 SequenceGenerationReset

Insert the following note at the end of 7.4.1.3:

NOTE—SequenceGenerationReset can cause packet drop.

7.4.3 Base recovery function

Change the third paragraph and NOTE 1 in 7.4.3 as follows:

The sequence recovery function has two sequence recovery algorithms (7.4.3.4, 7.4.3.5) so that it can be used either for Intermittent Streams (item c in 7.1.1) or for Bulk Streams (item d in 7.1.1). As long as the maximum difference in the number of packets in flight among all of the paths taken by the Member Streams served by a given sequence recovery function does not exceed the size of the SequenceHistory variable (7.4.3.2.2, 10.4.1.6), the packets are delivered without duplication, but can be delivered out-of-order. If the difference exceeds this value, then ~~duplicate packets can be delivered~~ packets can be discarded.

NOTE 1—The requirements for conformance to this standard, particularly item c in 5.9 and item e in 5.12, specify that the sequence recovery function need only support a minimum difference in path length for the correct elimination of duplicates in Bulk Streams. If the actual path difference in a given network exceeds the capability of a sequence recovery function, ~~then duplicate packets will be delivered~~ then packets will be discarded. It is up to the user to see that the capabilities of the systems purchased match the needs of the particular network in which they are employed.

7.4.3.3 SequenceRecoveryReset

Insert the following note at the end of 7.4.3.3:

NOTE—Reset of Sequence recovery function may cause temporary duplicate delivery.

7.4.3.4 VectorRecoveryAlgorithm

Change the second, third, and fourth paragraphs of 7.4.3.4 as follows:

~~One can observe that if the managed object frerSeqRcvyHistoryLength contains the value 1, then this algorithm is suitable for Intermittent Streams (item c in 7.1.1). In that case, the SequenceHistory variable (7.4.3.2.2) merely records whether the RecovSeqNum (7.4.3.2.3) does or does not record the sequence_number subparameter of a received packet. [It does not immediately after SequenceRecoveryReset (7.4.3.3) is called.] If frerSeqRcvyHistoryLength contains a value greater than one, VectorRecoveryAlgorithm serves as the more complex algorithm suitable for Bulk Streams (item d in 7.1.1).~~

Immediately after SequenceRecoveryReset (7.4.3.3) is called, the VectorRecoveryAlgorithm accepts the first packet received as valid. After the first packet has been accepted, all subsequent packets that are in the ~~window last packet number accepted ± frerSeqRcvyHistoryLength are accepted, and~~ recovery window (i.e., last packet number accepted – frerSeqRcvyHistoryLength + 1 to last packet number accepted + frerSeqRcvyHistoryLength) are accepted, and those packets with sequence_number values outside that range are discarded. Each packet accepted and passed up the stack resets the timer variable RemainingTicks (7.4.3.2.4). If that variable ticks down to 0, meaning that no packet has been accepted in frerSeqRcvyResetMSec milliseconds (10.4.1.7), then SequenceRecoveryReset again resets the algorithm, and the next packet received is accepted.

This timeout mechanism means that:

a) "Rogue" packets, meaning packets outside the ~~frerSeqRcvyHistoryLength window,~~ recovery window, are discarded as invalid.
b) If a Base recovery function somehow gets out of step with its corresponding Sequence generation function, then after frerSeqRcvyResetMSec milliseconds, the Base recovery function will be reset and data will again be passed.
c) If a Sequence generation function is reset, perhaps by rebooting a system, then Base recovery functions that have not been reset are likely to discard packets until the timeout has occurred.
d) If a Talker ~~or a relay system~~ fails in such a way as to repeatedly transmit packets with the same sequence_number subparameter (perhaps repeating exactly the same packet), ~~those packets will continue to be discarded, at least until the sequence_number wraps around~~then in the case of Individual recovery (7.5), all repeated packets are discarded. If only Sequence recovery (7.4.2) is used, then all except one repeated packets are discarded per RemainingTicks interval. If a relay system fails the same way, then in the case of Individual recovery (7.5), all packets are discarded. If only Sequence recovery (7.4.2) is used and if there is a correct Member Stream, then the failed Member Stream pollutes the correct Member Stream.

Change the pseudo-code at the end of 7.4.3.4 as follows:

```
void VectorRecoveryAlgorithm () {
   //  Check that sequence number is present in the packet
   unsigned int sequence_number;
   if (sequence_number == frerSeqRcvyInvalidSequenceValue) {
      frerCpsSeqRcvyTaglessPackets = frerCpsSeqRcvyTaglessPackets + 1;
      if (frerSeqRcvyTakeNoSequence) {
         frerCpsSeqRcvyPassedPackets = frerCpsSeqRcvyPassedPackets + 1;
         frerCpSeqRcvyPassedPackets = frerCpSeqRcvyPassedPackets  + 1;
         RemainingTicks              =
                     ((frerSeqRcvyResetMSec*TicksPerSecond)+999)/1000;
         PRESENT_DATA;
      } else {
         frerCpsSeqRcvyDiscardedPackets =
                                    frerCpsSeqRcvyDiscardedPackets + 1;
         frerCpSeqRcvyDiscardPackets = frerCpSeqRcvyDiscardPackets + 1;
      }
      return;
   }
   // Compute signed difference modulo RecovSeqSpace.
   int delta = (sequence_number-RecovSeqNum) & (RecovSeqSpace - 1);
   if (0 != (delta & (RecovSeqSpace/2)))
      delta = delta - RecovSeqSpace;
   // Here, -(RecovSeqSpace/2)<=delta<=((RecovSeqSpace/2)-1)
   // After reset, accept any packet
   if (TakeAny) {
      TakeAny                     = false;
      SequenceHistory[0]          = 1;    // Shift, adding a "seen" bit
      RecovSeqNum                 = sequence_number;
      frerCpsSeqRcvyPassedPackets = frerCpsSeqRcvyPassedPackets + 1;
      frerCpSeqRcvyPassedPackets  = frerCpSeqRcvyPassedPackets  + 1;
      RemainingTicks              =
         ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
      PRESENT_DATA;
   } else if (delta >= frerSeqRcvyHistoryLength  ||
```

```
            delta <= -frerSeqRcvyHistoryLength     )
    {
       // Packet is out-of-range.  Count and discard it.
       frerCpsSeqRcvyRoguePackets  = frerCpsSeqRcvyRoguePackets  + 1;
       frerCpSeqRcvyDiscardPackets = frerCpSeqRcvyDiscardPackets + 1;
       // Reset timer if working on an individual Stream
       if (frerSeqRcvyIndividualRecovery)
          RemainingTicks           =
             ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
    } else if (delta <= 0) {
       // Packet is old and in SequenceHistory; have we seen it before?
       if (0 == SequenceHistory[-delta]) {
          // Packet has not been seen.  Take it.
          SequenceHistory[-delta]     = 1;
          frerCpsSeqRcvyOutOfOrderPackets =
                                    frerCpsSeqRcvyOutOfOrderPackets  + 1;
          frerCpsSeqRcvyPassedPackets = frerCpsSeqRcvyPassedPackets + 1;
          frerCpSeqRcvyPassedPackets  = frerCpSeqRcvyPassedPackets  + 1;
          RemainingTicks              =
             ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
          PRESENT_DATA;
       } else {
          // Packet has been seen.  Do not forward.  Count the discard.
          frerCpsSeqRcvyDiscardedPackets =
                                     frerCpsSeqRcvyDiscardedPackets + 1;
          frerCpSeqRcvyDiscardPackets = frerCpSeqRcvyDiscardPackets + 1;
          // Reset timer if working on an individual Stream
          if (frerSeqRcvyIndividualRecovery)
             RemainingTicks        =
                ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
       }
    } else {
       // Packet is not too far ahead of the one we want.
       // Packet is out-of-order unless it directly follows RecovSeqNum
       if (delta != 1)
          frerCpsSeqRcvyOutOfOrderPackets =
                                     frerCpsSeqRcvyOutOfOrderPackets + 1;
       // Shift the history until bit 0 refers to sequence_number.
       while (0 != (delta = delta - 1))
          ShiftSequenceHistory(0);   // Shift, adding a "not seen" bit
       ShiftSequenceHistory(1);      // Shift, adding a "seen" bit
       RecovSeqNum                 = sequence_number;
       frerCpsSeqRcvyPassedPackets = frerCpsSeqRcvyPassedPackets + 1;
       frerCpSeqRcvyPassedPackets  = frerCpSeqRcvyPassedPackets  + 1;
       RemainingTicks              =
          ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
       PRESENT_DATA;
    }
}
```

7.4.3.5 MatchRecoveryAlgorithm

Change the second and third paragraphs of 7.4.3.5 as follows:

Immediately after SequenceRecoveryReset (7.4.3.3) is called, the MatchRecoveryAlgorithm accepts the first packet received as valid. After the first packet has been accepted, all subsequent packets either match the last packet number accepted, and are therefore discarded, or do not, in which case they are accepted. Each packet accepted and passed up the stack resets the timer variable RemainingTicks (7.4.3.2.4). Dropped packets also reset the timer variable RemainingTicks (7.4.3.2.4) in the case of Individual recovery. If that variable ticks down to 0, meaning that no packet has been accepted in frerSeqRcvyResetMSec milliseconds (10.4.1.7), then SequenceRecoveryReset again resets the algorithm, and the next packet received is accepted.

~~The timer mechanism prevents the MatchRecoveryAlgorithm from getting stuck forever, blocking packet 1, in case a Talker fails or is reset soon after initialization.~~

Change the pseudo-code at the end of 7.4.3.5 as follows:

```
void MatchRecoveryAlgorithm () {
   //  Check that sequence number is present in the packet
   unsigned int sequence_number;
   if (sequence_number~~,~~ == frerSeqRcvyInvalidSequenceValue) {
      frerCpsSeqRcvyTaglessPackets = frerCpsSeqRcvyTaglessPackets + 1;
      frerCpsSeqRcvyPassedPackets  = frerCpsSeqRcvyPassedPackets  + 1;
      frerCpSeqRcvyPassedPackets   = frerCpSeqRcvyPassedPackets   + 1;
      PRESENT_DATA;
      return;
   }
   // After reset, accept any packet
   if (TakeAny) {
      TakeAny                    = false;
      RecovSeqNum                = sequence_number;
      frerCpsSeqRcvyPassedPackets = frerCpsSeqRcvyPassedPackets + 1;
      frerCpSeqRcvyPassedPackets  = frerCpSeqRcvyPassedPackets  + 1;
      RemainingTicks             =
         ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
      PRESENT_DATA;
      return;
   }
   // Compute signed difference modulo RecovSeqSpace.
   int delta = (sequence_number-RecovSeqNum) & (RecovSeqSpace - 1);
   if (delta == 0) {
      // Packet has been seen.  Do not forward.  Count the discard.
      frerCpsSeqRcvyDiscardedPackets =
                                  frerCpsSeqRcvyDiscardedPackets + 1;
      frerCpSeqRcvyDiscardPackets    = frerCpSeqRcvyDiscardPackets + 1;
      // Reset timer if working on an individual Stream
      if (frerSeqRcvyIndividualRecovery)
         RemainingTicks          =
            ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
   } else {
      // Packet has not been seen; accept it.
      // Packet is out-of-order unless it directly follows RecovSeqNum
      if (delta != 1)
```

```
            frerCpsSeqRcvyOutOfOrderPackets =
                                frerCpsSeqRcvyOutOfOrderPackets + 1;
        RecovSeqNum                 = sequence_number;
        frerCpsSeqRcvyPassedPackets = frerCpsSeqRcvyPassedPackets + 1;
        frerCpSeqRcvyPassedPackets  = frerCpSeqRcvyPassedPackets  + 1;
        RemainingTicks              =
            ((frerSeqRcvyResetMSec * TicksPerSecond) + 999) / 1000;
        PRESENT_DATA;
    }
}
```

7.8 Redundancy tag

Change the second sentence of the first paragraph of 7.8 as follows:

It operates at the frame level and can be defined using the ISS defined by IEEE Std 802.1AC, or the EISS described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, enhanced with the extra stream_handle and sequence_number subparameters specified in 6.1.

7.9 HSR sequence tag

Change the second sentence of the first paragraph of 7.9 as follows:

It operates at the frame level and can be defined using the ISS defined by IEEE Std 802.1AC, or the EISS described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, enhanced with the extra stream_handle and sequence_number subparameters specified in 6.1.

7.10 PRP sequence trailer

Change the second sentence of the first paragraph of 7.10 as follows:

It operates at the frame level and can be defined using the ISS defined by IEEE Std 802.1AC, or the EISS described in 6.9 of IEEE Std 802.1Q-~~2014~~2018, enhanced with the extra stream_handle and sequence_number subparameters specified in 6.1.

8. Frame Replication and Elimination for Reliability in Bridges

8.1 Limiting options

Change item 2) in the numbered list in 8.1 as follows:

2) **Augmented forwarding:** The Bridge Forwarding Process, described in 8.6 of IEEE Std 802.1Q-~~2014~~2018 (shown in white boxes in Figure 8-1), augmented with Stream identification functions (6.2, both incoming and outgoing), Sequence encode/decode functions (7.6, both incoming and outgoing), Sequence recovery functions (7.4.2), Individual recovery functions (7.5), and their associated infrastructure, all shown in shaded boxes in Figure 8-1.

Change the sixth paragraph of 8.1 as follows (lettered list items f) and g) remain unchanged):

Figure 8-1 does not imply that the Augmented forwarding functions have to be placed in expanded ports; they can be integrated with the Bridge forwarding function (8.6 of IEEE 802.1Q-~~2014~~2018) as shown in Figure 8-2. The ordering of this functions with respect to 8.6 of IEEE 802.1Q-~~2014~~2018 can be important. In particular:

Change lettered list items h), i), and j) in 8.1 as follows:

h) If Input transformations are performed, the frame is forwarded (8.6 of IEEE Std 802.1Q-~~2014~~2018) according to its post-transformation parameters.

i) Flow metering (8.6.5 of IEEE Std 802.1Q-~~2014~~2018) is placed after the passive Stream identification function (6.2) and before the Individual recovery functions (7.5). This makes the stream_handle subparameter available for Flow metering, and means that Flow metering can be applied to the individual Streams feeding an instance of the Sequence recovery function. As a result, Flow metering can be applied to frames that will be discarded by the Individual recovery functions or Sequence recovery functions.

j) Output transformations following the Sequence recovery function take place after all forwarding (except Queuing frames, 8.6.6 of IEEE Std 802.1Q-~~2014~~2018). Thus, output transformations can appear to cause violations of the normal forwarding rules, e.g., Egress filtering (8.6.4 of IEEE Std 802.1Q-~~2014~~2018).

8.2 FRER C-component input transformations

Change 8.2 as follows:

The Input transformations, marked with white ~~boxed with~~boxes and boldface type in Figure 8-1, enable a Bridge to proxy for a non-FRER-capable end system. The expanded input port identifies packets belonging to a Stream (e.g., using IP Stream identification, 6.7, or Mask-and-match Stream identification, 6.8), serializes the packets with a Sequence generation function (7.4.1), encodes the sequence number with an R-TAG (7.8), and then gives the packets belonging to this Stream a {vlan_identifier, destination_mac_address} pair that is unique, at least inside this Bridge, using Active Destination MAC and VLAN Stream identification (6.6). The IEEE 802.1Q Forwarding Process, enhanced with the Individual recovery function (7.5) and Sequence recovery function (7.4.2), then forwards the frame.

8.3 Frame Replication and Elimination for Reliability and VLAN tags

Change the text of 8.3 as follows (Figure 8-3 remains unchanged):

As illustrated in Figure 8-1 and Figure 8-3, FRER in an IEEE 802.1Q C-VLAN Component is above the Bridge Port Transmit and Receive function (8.5 of IEEE Std 802.1Q-~~2014~~2018) in the protocol stack. As a consequence of this placement, a frame containing both an IEEE 802.1Q C-VLAN tag and an R-TAG but no other IEEE 802.1Q tags, would be as illustrated in Figure 8-3.

9. Stream Identification Management

9.1 Stream identity table

9.1.1 tsnStreamIdEntry

9.1.1.6 tsnStreamIdIdentificationType

Change Table 9-1 as follows:

Table 9-1—Stream identification types

OUI/CID	Type number	Stream identification function	Controlling parameters
00-80-C2	0	Reserved	—
00-80-C2	1	Null Stream identification (6.4)	9.1.2
00-80-C2	2	Source MAC and VLAN Stream identification (6.5)	9.1.3
00-80-C2	3	Active Destination MAC and VLAN Stream identification (6.6)	9.1.4
00-80-C2	4	IP Stream identification (6.7)	9.1.5
00-80-C2	5	Mask-and-match Stream identification (6.8)	9.1.6
00-80-C2	~~5~~6–255	Reserved	—
other	—	Defined by entity owning the OUI or CID	—

9.1.4 Managed objects for Active Destination MAC and VLAN Stream identifications

Change 9.1.4 as follows:

When instantiating an instance of the Active Destination MAC and VLAN Stream identification function (6.6) for a particular output Stream, the managed objects in the following subclauses~~, along with those listed in 9.1.2,~~ serve as the tsnStreamIdParameters managed object (9.1.1.7).

9.1.5 Managed objects for IP Stream identification

9.1.5.1 tsnCpeIpIdDestMac

Change 9.1.5.1 as follows:

Specifies the destination_address parameter that identifies a packet in an EISS indication primitive. An address of all 0 indicates that the MAC destination address is to be ignored on packets received from lower layers.

Insert new subclause 9.1.6 after 9.1.5 as follows:

9.1.6 Managed objects for Mask-and-match Stream identification

When instantiating an instance of the Mask-and-match Stream identification function (6.8) for a particular input Stream, the managed objects in the following subclauses serve as the tsnStreamIdParameters managed object (9.1.1.2).

9.1.6.1 tsnCpeMmIdDestMacMask

Specifies a 48-bit mask. A bitwise AND operation is performed between tsnCpeMmIdDestMacMask and the destination_address parameter passed by the ISS indication primitive to the Mask-and-match Stream identification function. The resulting 48-bit information is the masked destination_address that is used as input for the instance of the Mask-and-match Stream identification function. If tsnCpeMmIdDestMacMask has a value of 0, the destination_address parameter is ignored.

9.1.6.2 tsnCpeMmIdDestMacMatch

Specifies the 48-bit value of the masked destination_address, as defined in 9.1.6.1, to be matched by the instance of the Mask-and-match Stream identification function.

9.1.6.3 tsnCpeMmIdSrcMacMask

Specifies a 48-bit mask. A bitwise AND operation is performed between tsnCpeMmIdSrcMacMask and the source_address parameter passed by the ISS indication primitive to the Mask-and-match Stream identification function. The resulting 48-bit information is the masked source_address that is used as input for the instance of the Mask-and-match Stream identification function. If tsnCpeMmIdSrcMacMask has a value of 0, the source_address parameter is ignored.

9.1.6.4 tsnCpeMmIdSrcMacMatch

Specifies the 48-bit value of the masked source_address, as defined in 9.1.6.3, to be matched by the instance of the Mask-and-match Stream identification function.

9.1.6.5 tsnCpeMmIdMsduMaskLength

Is an integer value that specifies the length, in octets, of tsnCpeMmIdMsduMask (9.1.6.6) and tsnCpeMmIdMsduMatch (9.1.6.7). The behavior of the Mask-and-match Stream identification function is undefined on any port where the value of tsnCpeMmIdMsduMaskLength exceeds the per-port mac_service_data_unit mask maximum length (9.4.1). tsnCpeMmIdMsduMaskLength has a minimum value of 2. If the length of the frame's mac_service_data_unit is shorter than tsnCpeMmIdMsduMaskLength, then the identification fails.

9.1.6.6 tsnCpeMmIdMsduMask

Specifies a mask of tsnCpeMmIdMsduMaskLength octets. A bitwise AND operation is performed between tsnCpeMmIdMsduMask and the tsnCpeMmIdMsduMaskLength (9.1.6.5) first octets of the mac_service_data_unit parameter passed by the ISS indication primitive to the Mask-and-match Stream identification function. The resulting tsnCpeMmIdMsduMaskLength octets of information are the masked mac_service_data_unit that is used as input for the instance of the Mask-and-match Stream identification function. If tsnCpeMmIdMsduMask has a value of 0, the mac_service_data_unit parameter is ignored.

9.1.6.7 tsnCpeMmIdMsduMatch

Specifies the value of the masked mac_service_data_unit, as defined in 9.1.6.5 and 9.1.6.6, to be matched by the instance of the Mask-and-match Stream identification function.

NOTE—The Mask-and-match Stream identification function takes as input the ISS indication primitive parameters, which include the mac_service_data_unit parameter. If specific part(s) of the mac_service_data_unit are required for the identification of a Stream, the ad-hoc tsnCpeMmIdMsduMask and tsnCpeMmIdMsduMatch have to be defined. For instance, when a VLAN ID is required for Stream identification, it is to be masked within the mac_service_data_unit by setting 16 bits to 1 at the offset where the tag is expected (to isolate the EtherType), followed by 4 bits set to 0 (to ignore the PCP and DEI fields), and ending with 12 bits set to 1 (to mask the full VID) in tsnCpeMmIdMasduMask. In a Customer Bridged Network, the C-VLAN Tag is expected at an offset of 0, and the tsnCpeMmIdMsduMatch value used in the Mask-and-match Stream identification function instance contains, at the same offset, a pattern matching the C-VLAN EtherType (81-00), followed by 4 don't-care bits (set to 0), followed by 12 bits representing the VID value to be matched.

9.2 Operational per-port per-Stream Stream identification counters

Change 9.2 as follows:

The ~~following~~ counters in this subclause are instantiated for each port on which the Stream identification function (6.2) is configured. They ~~counters~~ are indexed by port number, facing (in-facing or out-facing), and stream_handle value (tsnStreamIdHandle, 9.1.1.1). ~~All counters are unsigned integers. If used on links faster than 650 000 000 bits per second, they shall be 64 bits in length to ensure against excessively short wrap times~~Encoded as unsigned integers, they shall be 64 bits in length to ensure against excessively short wrap times when used on links on which the transmission rate exceeds 650 000 000 bits per second.

9.3 Operational per-port Stream identification counters

Insert the following paragraph in 9.3:

The counters in this subclause are instantiated for each port on which the Stream identification function (6.2) is configured. They are indexed by port number. Encoded as unsigned integers, they shall be 64 bits in length to ensure against excessively short wrap times when used on links on which transmission rate exceeds 650 000 000 bits per second.

Insert subclause 9.4 after 9.3 as follows:

9.4 Per-port managed object

9.4.1 tsnPpMmIdMsduMaskMaxLength

tsnPpMmIdMsduMaskMaxLength is a read-only variable defined for each port the Mask-and-match Stream identification function is to be run on. It defines the maximum length a mac_service_data_unit mask can have. It is an integer value, in octets, with a minimum value of 2 and a maximum value of 1984 (maximum ISS mac_service_data_unit length for an IEEE 802.3 frame).

10. Frame Replication and Elimination for Reliability management

Change the heading for 10.2 as follows:

10.2 Additional tsnStreamIdEntry managed objects

10.4 Sequence recovery table

10.4.1 frerSeqRcvyEntry

10.4.1.4 frerSeqRcvyReset

Change 10.4.1.4 as follows:

A Boolean object indicating that the Sequence recovery function (7.4.2) or Individual recovery function (7.5) is to be reset by calling its corresponding ~~SequenceGenerationReset~~SequenceRecoveryReset function (~~7.4.1.3~~7.4.3.3). Writing the value True to frerSeqRcvyReset triggers a reset; writing the value False has no effect. When read, frerSeqRcvyReset always returns the value False.

10.4.1.10 frerSeqRcvyIndividualRecovery

Change 10.4.1.10 as follows:

A Boolean value specifying whether this entry describes a Sequence recovery function (7.4.2) or Individual recovery function (7.5).

a) **True:** The entry describes an Individual recovery function (7.5). Packets discarded by the ~~SequenceGenerationAlgorithm (7.4.1.4)~~ VectorRecoveryAlgorithm (7.4.3.4) or the MatchRecoveryAlgorithm (7.4.3.5) will cause the variable RemainingTicks (7.4.3.2.4) to be reset. There is no Latent error detection function (7.4.4) associated with this entry, so frerSeqRcvyLatentErrorDetection (10.4.1.11) cannot also be True.

b) **False:** The entry describes a Sequence recovery function (7.4.2). Packets discarded by the ~~SequenceGenerationAlgorithm (7.4.1.4)~~ VectorRecoveryAlgorithm (7.4.3.4) or the MatchRecoveryAlgorithm (7.4.3.5) will not cause the variable RemainingTicks (7.4.3.2.4) to be reset.

10.8 Operational per-port and per-Stream FRER counters

10.8.8 frerCpsSeqRcvyTaglessPackets

Change 10.8.8 as follows:

The frerCpsSeqRcvyTaglessPackets counter is incremented once for each packet received by the VectorRecoveryAlgorithm (7.4.3.4) or the MatchRecoveryAlgorithm (7.4.3.5) that has no sequence_number subparameter (item b in 6.1).

10.9 Operational per-port FRER counters

Change the first paragraph of 10.9 as follows:

The following counters are instantiated for each port on which any of the Stream identification function (6.2), Sequencing function (7.4), or Sequence encode/decode function (7.6) is configured. The counters are indexed by port number~~ and facing (in-facing or out-facing)~~. ~~All counters are unsigned integers. If used on links faster than 650 000 000 bits per second, they shall be 64 bits in length to ensure against excessively short wrap times.~~ Encoded as unsigned integers, they shall be 64 bits in length to ensure against excessively short wrap times when used on links on which the transmission rate exceeds 650 000 000 bits per second. A Stream identification component (5.3) shall implement the first two counters tsnCpSidInputPackets (9.3.1) and tsnCpSidOutputPackets (9.3.2); the remainder of the counters in 10.9 are optional for such a system.

Clause 11 was added by IEEE Std 802.1CBcv-2021.

11. Management Information Base (MIB)

11.2 Structure of the MIB

Replace Figure 11-1 with the figure shown on the next page (with "MaskAndMatch IdentificationTable" and "PerPort MsduMaskMaxLength Table" added):

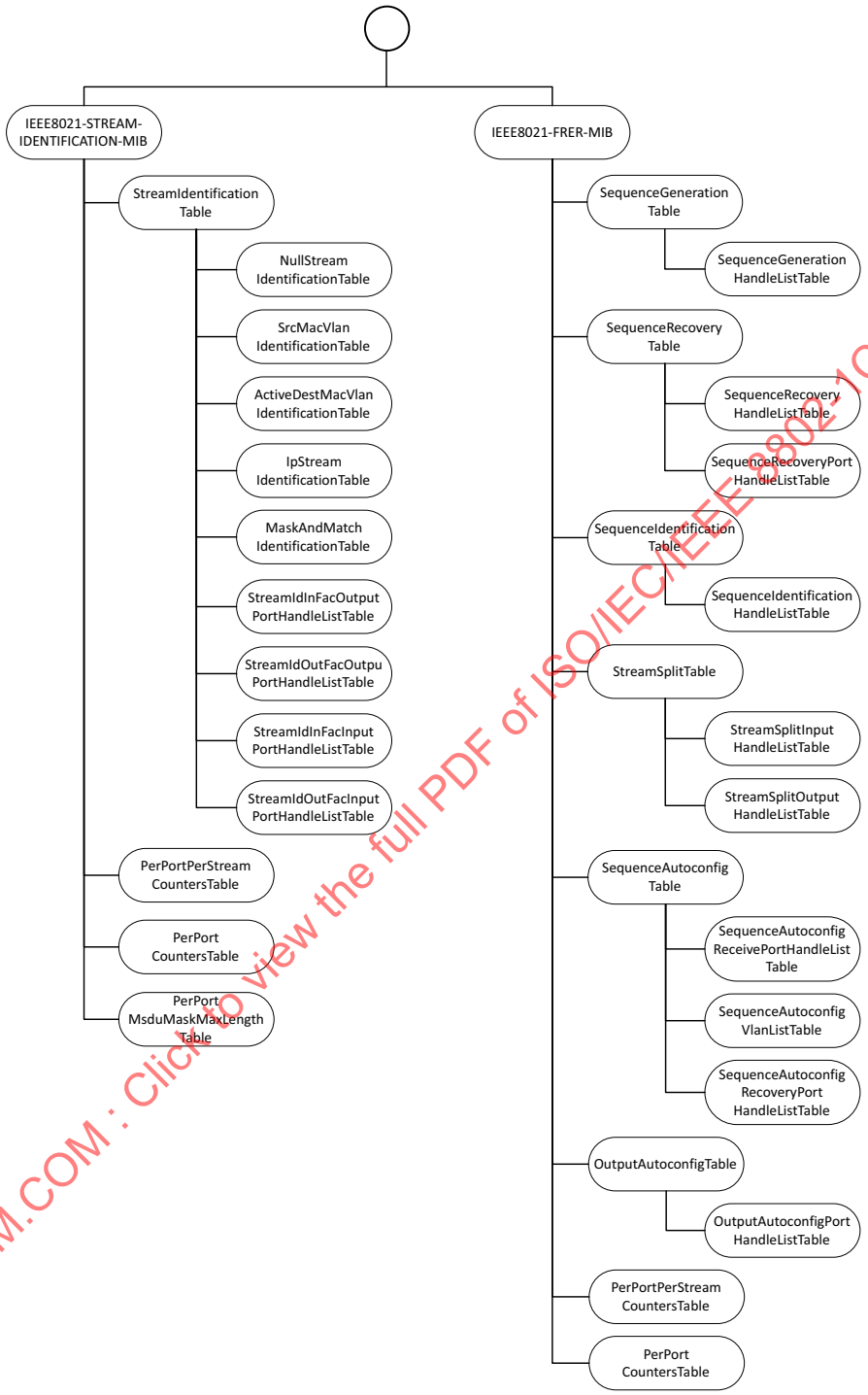

Figure 11-1—MIB modules structure

11.2.1 Structure of the IEEE8021-STREAM-IDENTIFICATION-MIB

Change Table 11-2 as follows:

Table 11-2—IEEE8021-STREAM-IDENTIFICATION-MIB structure and relationship to this standard

Clause 11 MIB table/object	Reference
ieee8021StreamIdStreamIdentificationTable	9.1, 9.1.1
ieee8021StreamIdStreamIdentificationIndex	—
ieee8021StreamIdStreamIdIdentificationType	—
ieee8021StreamIdStreamIdIdentificationTypeOUI	—
ieee8021StreamIdStreamIdIdentificationCustomType	—
ieee8021StreamIdStreamIdIdentificationCustomTypeOUI	—
ieee8021StreamIdStreamIdIdentificationTypeSelect	—
ieee8021StreamIdStreamIdIdentificationTypeSelect	—
ieee8021StreamIdStreamIdInFacOutputPortList	—
ieee8021StreamIdStreamIdOutFacOutputPortList	—
ieee8021StreamIdStreamIdInFacInputPortList	—
ieee8021StreamIdStreamIdOutFacInputPortList	—
ieee8021StreamIdAutoConfigured	—
ieee8021StreamIdLanPathId	—
ieee8021StreamIdStatus	—
ieee8021StreamIdNullStreamIdentificationTable	9.1.2
ieee8021StreamIdCpeNullDownDestMac	—
ieee8021StreamIdCPENullDownTagged	—
ieee8021StreamIdCpeNullDownVlan	—
ieee8021StreamIdSrcMacVlanIdentificationTable	9.1.3
ieee8021StreamIdCpeSmacVlanDownSrcMac	—
ieee8021StreamIdCpeSmacVlanDownTagged	—
ieee8021StreamIdCpeSmacVlanDownVlan	—
ieee8021StreamIdActiveDestMacVlanIdentificationTable	9.1.4
ieee8021StreamIdCpeDmacVlanDownDestMac	—
ieee8021StreamIdCpeDmacVlanDownTagged	—
ieee8021StreamIdCpeDmacVlanDownVlan	—
ieee8021StreamIdCpeDmacVlanDownPriority	—
ieee8021StreamIdCpeDmacVlanUpDestMac	—

Clause 11 MIB table/object	Reference
ieee8021StreamIdCpeDmacVlanUpTagged	—
ieee8021StreamIdCpeDmacVlanUpVlan	—
ieee8021StreamIdCpeDmacVlanUpPriority	—
Ieee8021StreamIdIpStreamIdentificationTable	9.1.5
ieee8021StreamIdCpeIpIdDestMac	—
ieee8021StreamIdCpeIpIdTagged	—
ieee8021StreamIdCpeIpIdVlan	—
ieee8021StreamIdCpeIpIdIpSourceType	—
ieee8021StreamIdCpeIpIdIpSource	—
ieee8021StreamIdCpeIpIdIpDestinationType	—
ieee8021StreamIdCpeIpIdIpDestination	—
ieee8021StreamIdCpeIpIdDscp	—
ieee8021StreamIdCpeIpIdNextProtocol	—
ieee8021StreamIdCpeIpIdSourcePort	—
ieee8021StreamIdCpeIpIdDestinationPort	—
ieee8021StreamIdMaskAndMatchIdentificationTable	9.1.6
ieee8021StreamIdCpeMmIdDestMacMask	—
ieee8021StreamIdCpeMmIdDestMacMatch	—
ieee8021StreamIdCpeMmIdSrcMacMask	—
ieee8021StreamIdCpeMmIdSrcMacMatch	—
ieee8021StreamIdCpeMmIdMsduMaskLength	—
ieee8021StreamIdCpeMmIdMsduMask	—
ieee8021StreamIdCpeMmIdMsduMatch	—
ieee8021StreamIdStreamIdInFacOutputPortHandleListTable	9.1.1.2
ieee8021StreamIdStreamIdInFacOutputPortHandleListIndex	—
ieee8021StreamIdStreamIdInFacOutputPortHandle	—
ieee8021StreamIdStreamIdInFacOutputPortHandleListStatus	—
ieee8021StreamIdStreamIdOutFacOutputPortHandleListTable	9.1.1.3
ieee8021StreamIdStreamIdOutFacOutputPortHandleListIndex	—
ieee8021StreamIdStreamIdOutFacOutputPortHandle	—
ieee8021StreamIdStreamIdOutFacOutputPortHandleListStatus	—
ieee8021StreamIdStreamIdInFacInputPortHandleListTable	9.1.1.2
ieee8021StreamIdStreamIdInFacInputPortHandleListIndex	—
ieee8021StreamIdStreamIdInFacInputPortHandle	—

Clause 11 MIB table/object	Reference
ieee8021StreamIdStreamIdInFacInputPortHandleListStatus	—
ieee8021StreamIdStreamIdOutFacInputPortHandleListTable	9.1.1.5
ieee8021StreamIdStreamIdOutFacInputPortHandleListIndex	—
ieee8021StreamIdStreamIdOutFacInputPortHandle	—
ieee8021StreamIdStreamIdOutFacInputPortHandleListStatus	—
ieee8021StreamIdPerPortPerStreamCountersTable	9.2
ieee8021StreamIdPerPortPerStreamDirection	—
ieee8021StreamIdPerPortPerStreamInputPackets	—
ieee8021StreamIdPerPortPerStreamOutputPackets	—
ieee8021StreamIdPerPortCountersTable	9.3
ieee8021StreamIdPerPortInputPackets	—
ieee8021StreamIdPerPortOutputPackets	—
ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthTable	9.4.1
ieee8021StreamIdMaskAndMatchMsduMaskMaxLength	—

11.5 MIB modules[7, 8]

11.5.1 Definitions for the IEEE8021-STREAM-IDENTIFICATION-MIB

Change 11.5.1 as follows:

```
IEEE8021-STREAM-IDENTIFICATION-MIB DEFINITIONS ::= BEGIN

-- ============================================================
-- MIB for support of Stream identification of IEEE 802.1CB
-- Frame Replication and Elimination for Reliability
-- ============================================================

IMPORTS
    OBJECT-GROUP,
    MODULE-COMPLIANCE
        FROM SNMPv2-CONF
    MODULE-IDENTITY,
    OBJECT-TYPE,
    OBJECT-IDENTITY,
    Counter64,
    Unsigned32,
    Integer32
        FROM SNMPv2-SMI
    TEXTUAL-CONVENTION,
    TruthValue,
```

[7] Copyright release for MIBs: Users of this standard may freely reproduce the MIBs contained in this subclause so that they can be used for their intended purpose.

[8] An ASCII version of the MIB modules are attached to the PDF version of this standard, and can be obtained by Web browser from the IEEE 802.1 Website at https://1.ieee802.org/mib-modules/.

```
    RowStatus,
    MacAddress,
    AutonomousType,
    VariablePointer
        FROM SNMPv2-TC
    ifIndex
        FROM IF-MIB
    InetAddressType,
    InetAddress,
    InetPortNumber
        FROM INET-ADDRESS-MIB
    IEEE8021PriorityValue,
    ieee802dot1mibs
        FROM IEEE8021-TC-MIB
    ;

ieee8021StreamIdMib MODULE-IDENTITY
    LAST-UPDATED "202112080000Z" -- December 8, 2021
    LAST-UPDATED "202112090000Z" -- December 9, 2021
    ORGANIZATION "IEEE 802.1 Working Group"
    CONTACT-INFO
                "WG-URL: http://ieee802.org/1/
                WG-EMail: stds-802-1-l@ieee.org

    Contact: IEEE 802.1 Working Group Chair
    Postal: C/O IEEE 802.1 Working Group
            IEEE Standards Association
            445 Hoes Lane
            Piscataway, NJ 08854
            USA

    E-mail: stds-802-1-chairs@ieee.org"

    DESCRIPTION
        "The Management Information Base module for IEEE 802.1CB Stream
         identification. Unless otherwise indicated, the references in
         this MIB module are to IEEE Std 802.1CB-2017 as amended by
         IEEE Std 802.1CBcv-2021 and IEEE Std 802.1CBdb-2021.
         Copyright (C) IEEE (2021 2022).
         This version of this MIB module is part of IEEE Std 802.1CB-2021;
         see the standard itself for full legal notices."

    REVISION "202112090000Z" -- December 9, 2021
    DESCRIPTION
        "Published as part of IEEE Std 802.1CBdb-2021. Adds extended
         Stream identification function objects."

    REVISION "202112080000Z" -- December 8, 2021
    DESCRIPTION
        "Initial revision published as part of IEEE Std 802.1CBcv-2021."

    ::= { ieee802dot1mibs 34 }

-- ================================================================
-- Textual Conventions
-- ================================================================

Ieee8021CBStreamIdentificationType ::= TEXTUAL-CONVENTION
    STATUS       current
```

```
    DESCRIPTION
        "An enumerated value indicating the method used to identify
         packets belonging to the Stream. The values 0-255 are reserved
         for use by IEEE 802."
    REFERENCE    "9.1.1.6 and Table 9-1"
    SYNTAX      INTEGER {
                    reserved(0),
                    nullStreamIdentification(1),
                    srcMacVlanStreamIdentification(2),
                    activeDstMacVlanStreamIdentification(3),
                    ipStreamStreamIdentification(4),
                    maskAndMatchStreamIdentification(5),
                    nonIEEESpecified(256)
                }

Ieee8021CBTaggedType ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
        "An enumerated value indicating whether a packet in an EISS
         indication primitive to a Stream identification function is
         permitted to have a VLAN tag."
    REFERENCE    "9.1.2.2, 9.1.3.2, 9.1.4.2, 9.1.4.6, 9.1.5.2, and
                  10.7.1.1.3"
    SYNTAX      INTEGER {
                    tagged(1),
                    priority(2),
                    all(3)
                }

Ieee8021CBVlanIdentifier ::= TEXTUAL-CONVENTION
    DISPLAY-HINT "d"
    STATUS       current
    DESCRIPTION
        "Specifies the vlan_identifier. A value of 0 indicates a special
         treatment, depending on the use of Ieee8021CBVlanIdentifier."
    SYNTAX      Integer32 (0..4094)

Ieee8021CBIpDscpType ::= TEXTUAL-CONVENTION
    DISPLAY-HINT "d"
    STATUS   current
    DESCRIPTION
        "The code point used for discriminating a traffic
         Stream."
    SYNTAX      Integer32 (-1 | 0..63)

Ieee8021CBIdNextProtocolType ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
        "An enumerated value indicating an IP next protocol."
    SYNTAX      INTEGER {
                    none(1),
                    udp(2),
                    tcp(3),
                    sctp(4)
                }

Ieee8021CBLanPathIdType ::= TEXTUAL-CONVENTION
    DISPLAY-HINT "d"
    STATUS       current
```

```
    DESCRIPTION
        "An integer specifying a path or LAN. If and only if a packet
        matches an entry in the Sequence identification table that
        specifies HSR or PRP in its frerSeqEncEncapsType  object,
        tsnStreamIdLanPathId specifies the LanId or PathId value that
        must be matched for this tsnStreamIdEntry to apply. A value
        of -1 indicates that the LanId or PathId are to be ignored."
    REFERENCE    "10.22"
    SYNTAX      Integer32 (-1 | 0..15)

-- ============================================================
-- subtrees in the Stream identification MIB
-- ============================================================

ieee8021StreamIdNotifications
    OBJECT IDENTIFIER ::= { ieee8021StreamIdMib 0 }

ieee8021StreamIdObjects
    OBJECT IDENTIFIER ::= { ieee8021StreamIdMib 1 }

ieee8021StreamIdConformance
    OBJECT IDENTIFIER ::= { ieee8021StreamIdMib 2 }

ieee8021StreamIdStreamIdentification
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 1 }

ieee8021StreamIdStreamIdInFacOutputPortHandleList
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 2 }

ieee8021StreamIdStreamIdOutFacOutputPortHandleList
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 3 }

ieee8021StreamIdStreamIdInFacInputPortHandleList
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 4 }

ieee8021StreamIdStreamIdOutFacInputPortHandleList
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 5 }

ieee8021StreamIdPerPortPerStreamCounters
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 6 }

ieee8021StreamIdPerPortCounters
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 7 }

ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLength
    OBJECT IDENTIFIER ::= { ieee8021StreamIdObjects 8 }

-- ============================================================
-- the ieee8021StreamIdStreamIdentification table
-- ============================================================

-- ============================================================
-- the ieee8021StreamIdStreamIdentificationTypes
-- ============================================================
ieee8021StreamIdStreamIdentificationTypes
    OBJECT IDENTIFIER ::= { ieee8021StreamIdStreamIdentification 1 }

ieee8021StreamIdNullStream OBJECT-IDENTITY
       STATUS current
```

```
        DESCRIPTION
            "The Stream identification type for the Null Stream
             identification."
        ::= { ieee8021StreamIdStreamIdentificationTypes 1 }

ieee8021StreamIdSrcMacVlan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The Stream identification type for the Source MAC and VLAN
             Stream identification."
        ::= { ieee8021StreamIdStreamIdentificationTypes 2 }

ieee8021StreamIdActiveDestMacVlan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The Stream identification type for the Active Destination
             and VLAN Stream identification."
        ::= { ieee8021StreamIdStreamIdentificationTypes 3 }

ieee8021StreamIdIpStream OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The Stream identification type for the IP Stream
             identification."
        ::= { ieee8021StreamIdStreamIdentificationTypes 4 }

ieee8021StreamIdMaskAndMatch OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The Stream identification type for the Mask-and-match
             Stream identification."
        ::= { ieee8021StreamIdStreamIdentificationTypes 5 }

-- ================================================================
-- the ieee8021StreamIdStreamIdentification
-- ================================================================

ieee8021StreamIdStreamIdentificationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdStreamIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of tsnStreamIdEntry objects each
         relating to a single Stream, specifying the points in the
         system where Stream identification functions are to be
         instantiated."
    REFERENCE   "9.1"
    ::= { ieee8021StreamIdStreamIdentification 2 }

Ieee8021StreamIdStreamIdentificationEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdStreamIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects, all applying to the Stream specified
        by tsnStreamIdHandle, and all using the same Stream
        identification types and parameters."
    REFERENCE   "9.1.1"
    INDEX  { ieee8021StreamIdStreamIdentificationIndex }
```

```
    ::= { ieee8021StreamIdStreamIdentificationTable 1 }

Ieee8021StreamIdStreamIdentificationEntry ::=
    SEQUENCE {
        ieee8021StreamIdStreamIdentificationIndex
            Unsigned32,
        ieee8021StreamIdStreamIdIdentificationType
            Ieee8021CBStreamIdentificationType,
        ieee8021StreamIdStreamIdIdentificationTypeOUI
            OCTET STRING,
        ieee8021StreamIdStreamIdIdentificationCustomType
            Integer32,
        ieee8021StreamIdStreamIdIdentificationCustomTypeOUI
            OCTET STRING,
        ieee8021StreamIdStreamIdIdentificationTypeSelect
            AutonomousType,
        ieee8021StreamIdStreamIdHandle
            Unsigned32,
        ieee8021StreamIdStreamIdInFacOutputPortList
            AutonomousType,
        ieee8021StreamIdStreamIdOutFacOutputPortList
            AutonomousType,
        ieee8021StreamIdStreamIdInFacInputPortList
            AutonomousType,
        ieee8021StreamIdStreamIdOutFacInputPortList
            AutonomousType,
        ieee8021StreamIdAutoConfigured
            TruthValue,
        ieee8021StreamIdLanPathId
            Ieee8021CBLanPathIdType,
        ieee8021StreamIdStatus
            RowStatus
    }

ieee8021StreamIdStreamIdentificationIndex OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Index for the Stream Identity table."
    ::= { ieee8021StreamIdStreamIdentificationEntry 1 }

ieee8021StreamIdStreamIdIdentificationType OBJECT-TYPE
    SYNTAX      Ieee8021CBStreamIdentificationType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This managed object is an enumerated value indicating the
         method used to identify packets belonging to the Stream.
         If the value used is nonIEEESpecified(256), the managed objects
         ieee8021StreamIdStreamIdIdentificationCustomType and
         ieee8021StreamIdStreamIdIdentificationCustomTypeOUI are used to
         specify OUI/CID and the Stream identification method defined by
         the entity owning the OUI/CID."
    REFERENCE   "9.1.1.6"
    DEFVAL  { nullStreamIdentification }
    ::= { ieee8021StreamIdStreamIdentificationEntry 2 }

ieee8021StreamIdStreamIdIdentificationTypeOUI OBJECT-TYPE
```

```
        SYNTAX      OCTET STRING (SIZE (3))
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "This managed object specifies the OUI/CID used together with
             the Stream identification method specified in
             ieee8021StreamIdStreamIdIdentificationType.
             It is used if and only if the value for
             ieee8021StreamIdStreamIdIdentificationType is in the range of
             0-255. In this case it always takes the value 00-80-C2."
        REFERENCE   "9.1.1.6"
        ::= { ieee8021StreamIdStreamIdentificationEntry 3 }

    ieee8021StreamIdStreamIdIdentificationCustomType OBJECT-TYPE
        SYNTAX      Integer32 (256..2147483647)
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "This managed object is an enumerated value indicating the
             method used to identify packets belonging to the Stream. It is
             used if and only if the value for
             ieee8021StreamIdStreamIdIdentificationType is 256."
        REFERENCE   "9.1.1.6"
        ::= { ieee8021StreamIdStreamIdentificationEntry 4 }

    ieee8021StreamIdStreamIdIdentificationCustomTypeOUI OBJECT-TYPE
        SYNTAX      OCTET STRING (SIZE (3))
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "This managed object specifies the OUI/CID used together with
             the Stream identification method specified in
             ieee8021StreamIdStreamIdIdentificationType. It is used if and
             only if the value for
             ieee8021StreamIdStreamIdIdentificationType is 256."
        REFERENCE   "9.1.1.6"
        ::= { ieee8021StreamIdStreamIdentificationEntry 5 }

    ieee8021StreamIdStreamIdIdentificationTypeSelect OBJECT-TYPE
        SYNTAX      AutonomousType
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "An indication of the type of Stream identification method that
             is used.

             If this value is
             'ieee8021StreamIdNullStream  ieee8021StreamIdObjects 1' then
             an entry exists in the
             ieee8021StreamIdStreamIdentificationTable
             which corresponds to the Null Stream identification.

             If this value is
             'ieee8021StreamIdSrcMacVlan  ieee8021StreamIdObjects 2' then
             an entry exists in the
             ieee8021StreamIdStreamIdentificationTable
             which corresponds to the Source MAC and VLAN Stream
             identification.
```

```
            If this value is
            'ieee8021StreamIdActiveDestMacVlan  ieee8021StreamIdObjects 3'
            then an entry exists in the
            ieee8021StreamIdStreamIdentificationTable
            which corresponds to the Active Destination MAC and VLAN Stream
            identification.

            If this value is
            'ieee8021StreamIdIpStream  ieee8021StreamIdObjects 4' then
            an entry exists in the
            ieee8021StreamIdStreamIdentificationTable
            which corresponds to the IP Stream identification.

            If this value is
            'ieee8021StreamIdMaskAndMatch  ieee8021StreamIdObjects 5' then
            an entry exists in the
            ieee8021StreamIdStreamIdentificationTable
            which corresponds to the Mask-and-match Stream
            identification."
        ::= { ieee8021StreamIdStreamIdentificationEntry 6 }

ieee8021StreamIdStreamIdHandle OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The objects in a given entry of the Stream identity table are
             used to control packets whose stream handle subparameter is
             equal to the entry's tsnStreamIdHandle object."
        REFERENCE   "9.1.1.1"
        ::= { ieee8021StreamIdStreamIdentificationEntry 7 }

ieee8021StreamIdStreamIdInFacOutputPortList OBJECT-TYPE
        SYNTAX      AutonomousType
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The list of ports on which an in-facing Stream identification
             function using this identification method is to be placed for
             this Stream in the output (towards the system forwarding
             function) direction.
             For each port in this list, there exists an entry in the
             ieee8021StreamIdStreamIdInFacOutputPortHandleListTable."
        REFERENCE   "9.1.1.2"
        ::= { ieee8021StreamIdStreamIdentificationEntry 8 }

ieee8021StreamIdStreamIdOutFacOutputPortList OBJECT-TYPE
        SYNTAX      AutonomousType
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The list of ports on which an out-facing Stream identification
             function using this identification method is to be placed for
             this Stream in the output (towards the physical interface)
             direction.
             For each port in this list, there exists an entry in the
             ieee8021StreamIdStreamIdOutFacOutputPortHandleListTable."
        REFERENCE   "9.1.1.3"
        ::= { ieee8021StreamIdStreamIdentificationEntry 9 }
```

```
ieee8021StreamIdStreamIdInFacInputPortList OBJECT-TYPE
    SYNTAX      AutonomousType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The list of ports on which an in-facing Stream identification
         function using this identification method is to be placed for
         this Stream in the input (coming from the system forwarding
         function) direction.
         For each port in this list, there exists an entry in the
         ieee8021StreamIdStreamIdInFacInputPortHandleListTable."
    REFERENCE   "9.1.1.4"
    ::= { ieee8021StreamIdStreamIdentificationEntry 10 }

ieee8021StreamIdStreamIdOutFacInputPortList OBJECT-TYPE
    SYNTAX      AutonomousType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The list of ports on which an out-facing Stream identification
         function using this identification method is to be placed for
         this Stream in the input (coming from the physical interface)
         direction.
         For each port in this list, there exists an entry in the
         ieee8021StreamIdStreamIdOutFacInputPortHandleListTable."
    REFERENCE   "9.1.1.5"
    ::= { ieee8021StreamIdStreamIdentificationEntry 11 }

ieee8021StreamIdAutoConfigured OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A read-only Boolean value, supplied by the system, specifying
         whether this entry was created explicitly (False) or via the
         Sequence autoconfiguration table (True)."
    REFERENCE   "10.2.1"
    ::= { ieee8021StreamIdStreamIdentificationEntry 12 }

ieee8021StreamIdLanPathId OBJECT-TYPE
    SYNTAX      Ieee8021CBLanPathIdType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "An integer specifying a path or LAN. If and only if a packet
         matches an entry in the Sequence identification table that
         specifies HSR or PRP in its frerSeqEncEncapsType object,
         tsnStreamIdLanPathId specifies the LanId or PathId value that
         must be matched for this tsnStreamIdEntry to apply. A value
         of -1 indicates that the LanId or PathId are to be ignored."
    REFERENCE   "10.2.2"
    ::= { ieee8021StreamIdStreamIdentificationEntry 13 }

ieee8021StreamIdStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
```

```
        "Row-Status for the Stream Identity table."
    ::= { ieee8021StreamIdStreamIdentificationEntry 14 }

-- ============================================================
-- the ieee8021StreamIdNullStreamIdentificationTable
-- ============================================================

ieee8021StreamIdNullStreamIdentificationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdNullStreamIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of controlling parameters for the
         Null Stream identification method."
    REFERENCE   "9.1.2"
    ::= { ieee8021StreamIdStreamIdentification 3 }

ieee8021StreamIdNullStreamIdentificationEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdNullStreamIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects that serve as the Stream
         identification parameters when using the Null Stream
         identification method."
    REFERENCE   "9.1.2"
    INDEX  { ieee8021StreamIdStreamIdentificationIndex }
    ::= { ieee8021StreamIdNullStreamIdentificationTable 1 }

Ieee8021StreamIdNullStreamIdentificationEntry ::=
    SEQUENCE {
        ieee8021StreamIdCpeNullDownDestMac
            MacAddress,
        ieee8021StreamIdCPENullDownTagged
            Ieee8021CBTaggedType,
        ieee8021StreamIdCpeNullDownVlan
            Ieee8021CBVlanIdentifier
    }

ieee8021StreamIdCpeNullDownDestMac OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the destination_address that identifies a packet in
         an EISS indication primitive, to the Null Stream identification
         function."
    REFERENCE   "9.1.2.1"
    ::= { ieee8021StreamIdNullStreamIdentificationEntry 1 }

ieee8021StreamIdCPENullDownTagged OBJECT-TYPE
    SYNTAX      Ieee8021CBTaggedType
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "An enumerated value indicating whether a packet in an EISS
         indication primitive to the Null Stream identification function
         is permitted to have a VLAN tag."
    REFERENCE   "9.1.2.2"
```

```
    ::= { ieee8021StreamIdNullStreamIdentificationEntry 2 }

ieee8021StreamIdCpeNullDownVlan OBJECT-TYPE
    SYNTAX      Ieee8021CBVlanIdentifier
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the vlan_identifier parameter that identifies a
         packet in an EISS indication primitive to the Null Stream
         identification function. A value of 0 indicates that the
         vlan_identifier parameter is ignored on EISS indication
         primitives."
    REFERENCE   "9.1.2.3"
    ::= { ieee8021StreamIdNullStreamIdentificationEntry 3 }

-- ================================================================
-- the ieee8021StreamIdSrcMacVlanIdentificationTable
-- ================================================================

ieee8021StreamIdSrcMacVlanIdentificationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdSrcMacVlanIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of controlling parameters for the
         Source MAC and VLAN Stream identification method."
    REFERENCE   "9.1.3"
    ::= { ieee8021StreamIdStreamIdentification 4 }

ieee8021StreamIdSrcMacVlanIdentificationEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdSrcMacVlanIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects that serve as the Stream
         identification parameters when using the Source MAC and VLAN
         Stream identification method."
    REFERENCE   "9.1.3"
    INDEX  { ieee8021StreamIdStreamIdentificationIndex }
    ::= { ieee8021StreamIdSrcMacVlanIdentificationTable 1 }

Ieee8021StreamIdSrcMacVlanIdentificationEntry ::=
    SEQUENCE {
        ieee8021StreamIdCpeSmacVlanDownSrcMac
            MacAddress,
        ieee8021StreamIdCpeSmacVlanDownTagged
            Ieee8021CBTaggedType,
        ieee8021StreamIdCpeSmacVlanDownVlan
            Ieee8021CBVlanIdentifier
    }

ieee8021StreamIdCpeSmacVlanDownSrcMac OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the source_address that identifies a packet in an
         EISS indication primitive, to the Source MAC and VLAN Stream
         identification function."
```

```
    REFERENCE   "9.1.3.1"
    ::= { ieee8021StreamIdSrcMacVlanIdentificationEntry 1 }

ieee8021StreamIdCpeSmacVlanDownTagged OBJECT-TYPE
    SYNTAX      Ieee8021CBTaggedType
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "An enumerated value indicating whether a packet in an EISS
         indication primitive to the Source MAC and VLAN Stream
         identification function is permitted to have a VLAN tag."
    REFERENCE   "9.1.3.2"
    ::= { ieee8021StreamIdSrcMacVlanIdentificationEntry 2 }

ieee8021StreamIdCpeSmacVlanDownVlan OBJECT-TYPE
    SYNTAX      Ieee8021CBVlanIdentifier
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the vlan_identifier parameter that identifies a
         packet in an EISS indication primitive to the Source MAC and
         VLAN Stream identification function. A value of 0 indicates
         that the vlan_identifier parameter is ignored on EISS
         indication primitives."
    REFERENCE   "9.1.3.3"
    ::= { ieee8021StreamIdSrcMacVlanIdentificationEntry 3 }

-- ================================================================
-- the ieee8021StreamIdActiveDestMacVlanIdentificationTable
-- ================================================================

ieee8021StreamIdActiveDestMacVlanIdentificationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdActiveDestMacVlanIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of controlling parameters for the
         Active Destination MAC and VLAN Stream identification method."
    REFERENCE   "9.1.4"
    ::= { ieee8021StreamIdStreamIdentification 5 }

ieee8021StreamIdActiveDestMacVlanIdentificationEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdActiveDestMacVlanIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects that serve as the Stream
         identification parameters when using the Active Destination
         MAC and VLAN Stream identification method."
    REFERENCE   "9.1.4"
    INDEX  { ieee8021StreamIdStreamIdentificationIndex }
    ::= { ieee8021StreamIdActiveDestMacVlanIdentificationTable 1 }

Ieee8021StreamIdActiveDestMacVlanIdentificationEntry ::=
    SEQUENCE {
        ieee8021StreamIdCpeDmacVlanDownDestMac
            MacAddress,
        ieee8021StreamIdCpeDmacVlanDownTagged
            Ieee8021CBTaggedType,
```

```
            ieee8021StreamIdCpeDmacVlanDownVlan
                Ieee8021CBVlanIdentifier,
            ieee8021StreamIdCpeDmacVlanDownPriority
                IEEE8021PriorityValue,
            ieee8021StreamIdCpeDmacVlanUpDestMac
                MacAddress,
            ieee8021StreamIdCpeDmacVlanUpTagged
                Ieee8021CBTaggedType,
            ieee8021StreamIdCpeDmacVlanUpVlan
                Ieee8021CBVlanIdentifier,
            ieee8021StreamIdCpeDmacVlanUpPriority
                IEEE8021PriorityValue
        }

    ieee8021StreamIdCpeDmacVlanDownDestMac OBJECT-TYPE
        SYNTAX      MacAddress
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Specifies the destination_address parameter to use in the
             EISS request primitive for output packets sent to lower layers
             by the Active Destination MAC and VLAN Stream identification
             function, and the destination_address that identifies an input
             packet in an EISS indication primitive to the Active
             Destination MAC and VLAN Stream identification function."
        REFERENCE   "9.1.4.1"
        ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 1 }

    ieee8021StreamIdCpeDmacVlanDownTagged OBJECT-TYPE
        SYNTAX      Ieee8021CBTaggedType
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "An enumerated value indicating whether a packet in an EISS
             indication or request primitive between the Active Destination
             MAC and VLAN Stream identification function and the lower
             layers is to have a VLAN tag.
             This variable is not used in an FRER C-component."
        REFERENCE   "9.1.4.2"
        ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 2 }

    ieee8021StreamIdCpeDmacVlanDownVlan OBJECT-TYPE
        SYNTAX      Ieee8021CBVlanIdentifier
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Specifies the vlan_identifier parameter to use in the EISS
             request primitive for output packets sent to lower layers by
             the Active Destination MAC and VLAN Stream identification
             function, and the vlan_identifier that identifies an input
             packet in an EISS indication primitive to the Active
             Destination MAC and VLAN Stream identification function.
             A value of 0 indicates that the vlan_identifier parameter is
             ignored on EISS indication primitives."
        REFERENCE   "9.1.4.3"
        ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 3 }

    ieee8021StreamIdCpeDmacVlanDownPriority OBJECT-TYPE
        SYNTAX      IEEE8021PriorityValue
```

```
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the priority parameter to use in the EISS request
         primitive for output packets sent to lower layers by the Active
         Destination MAC and VLAN Stream identification function for all
         packets in a particular Stream."
    REFERENCE   "9.1.4.4"
    ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 4 }

ieee8021StreamIdCpeDmacVlanUpDestMac OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the destination_address parameter to use in the EISS
         indication primitive for input packets offered to upper layers
         by the Active Destination MAC and VLAN Stream identification
         layer. This address replaces the address that was used to
         identify the packet (tsnCpeDmacVlanDownDestMac)."
    REFERENCE   "9.1.4.5"
    ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 5 }

ieee8021StreamIdCpeDmacVlanUpTagged OBJECT-TYPE
    SYNTAX      Ieee8021CBTaggedType
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "An enumerated value indicating whether a packet in an EISS
         indication or request primitive between the Active Destination
         MAC and VLAN Stream identification function and the upper
         layers is to have a VLAN tag.
         This variable is used only by an end system and not by a
         relay system."
    REFERENCE   "9.1.4.6"
    ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 6 }

ieee8021StreamIdCpeDmacVlanUpVlan OBJECT-TYPE
    SYNTAX      Ieee8021CBVlanIdentifier
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the vlan_identifier parameter to use in the EISS
         indication primitive for packets offered to upper layers, or
         the VLAN ID field for an IEEE 802.1Q tag in an
         EISS mac_service_data_unit. This address replaces the VLAN ID
         that was used to identify the packet (tsnCpeDmacVlanDownVlan)."
    REFERENCE   "9.1.4.7"
    ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 7 }

ieee8021StreamIdCpeDmacVlanUpPriority OBJECT-TYPE
    SYNTAX      IEEE8021PriorityValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the priority parameter to use in the EISS indication
         primitive for packets offered to upper layers."
    REFERENCE   "9.1.4.8"
    ::= { ieee8021StreamIdActiveDestMacVlanIdentificationEntry 8 }
```

```
-- ============================================================
-- the ieee8021StreamIdIpStreamIdentificationTable
-- ============================================================

ieee8021StreamIdIpStreamIdentificationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdIpStreamIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of controlling parameters for the
         IP Stream identification method."
    REFERENCE   "9.1.5"
    ::= { ieee8021StreamIdStreamIdentification 6 }

ieee8021StreamIdIpStreamIdentificationEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdIpStreamIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects that serve as the Stream
         identification parameters when using the IP Stream
         identification method."
    REFERENCE   "9.1.5"
    INDEX  { ieee8021StreamIdStreamIdentificationIndex }
    ::= { ieee8021StreamIdIpStreamIdentificationTable 1 }

Ieee8021StreamIdIpStreamIdentificationEntry ::=
    SEQUENCE {
        ieee8021StreamIdCpeIpIdDestMac
            MacAddress,
        ieee8021StreamIdCpeIpIdTagged
            Ieee8021CBTaggedType,
        ieee8021StreamIdCpeIpIdVlan
            Ieee8021CBVlanIdentifier,
        ieee8021StreamIdCpeIpIdIpSourceType
            InetAddressType,
        ieee8021StreamIdCpeIpIdIpSource
            InetAddress,
        ieee8021StreamIdCpeIpIdIpDestinationType
            InetAddressType,
        ieee8021StreamIdCpeIpIdIpDestination
            InetAddress,
        ieee8021StreamIdCpeIpIdDscp
            Ieee8021CBIpDscpType,
        ieee8021StreamIdCpeIpIdNextProtocol
            Ieee8021CBIdNextProtocolType,
        ieee8021StreamIdCpeIpIdSourcePort
            InetPortNumber,
        ieee8021StreamIdCpeIpIdDestinationPort
            InetPortNumber
    }

ieee8021StreamIdCpeIpIdDestMac OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the destination_address parameter that identifies a
```

```
            packet in an EISS indication primitive."
        REFERENCE   "9.1.5.1"
        ::= { ieee8021StreamIdIpStreamIdentificationEntry 1 }

    ieee8021StreamIdCpeIpIdTagged OBJECT-TYPE
        SYNTAX      Ieee8021CBTaggedType
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "An enumerated value indicating whether a packet in an EISS
             indication or request primitive to the IP Stream identification
             function is to have a VLAN tag."
        REFERENCE   "9.1.5.2"
        ::= { ieee8021StreamIdIpStreamIdentificationEntry 2 }

    ieee8021StreamIdCpeIpIdVlan OBJECT-TYPE
        SYNTAX      Ieee8021CBVlanIdentifier
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Specifies the vlan_identifier parameter that identifies a
             packet in an EISS indication primitive. A value of 0 indicates
             that the frame is not to have a VLAN tag."
        REFERENCE   "9.1.5.3"
        ::= { ieee8021StreamIdIpStreamIdentificationEntry 3 }

    ieee8021StreamIdCpeIpIdIpSourceType OBJECT-TYPE
        SYNTAX      InetAddressType
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Specifies the type of the source address parameter supplied
             in ieee8021StreamIdCpeIpIdIpSource."
        REFERENCE   "9.1.5.4"
        ::= { ieee8021StreamIdIpStreamIdentificationEntry 4 }

    ieee8021StreamIdCpeIpIdIpSource OBJECT-TYPE
        SYNTAX      InetAddress
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Specifies the IPv4 (RFC 791) or IPv6 (RFC 2460) source address
             parameter that must be matched to identify packets coming up
             from lower layers. An address of all 0 indicates that the
             IP source address is to be ignored on packets received from
             lower layers."
        REFERENCE   "9.1.5.4"
        ::= { ieee8021StreamIdIpStreamIdentificationEntry 5 }

    ieee8021StreamIdCpeIpIdIpDestinationType OBJECT-TYPE
        SYNTAX      InetAddressType
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "Specifies the type of the destination address parameter
             supplied in ieee8021StreamIdCpeIpIdIpDestination."
        REFERENCE   "9.1.5.4"
        ::= { ieee8021StreamIdIpStreamIdentificationEntry 6 }
```

```
ieee8021StreamIdCpeIpIdIpDestination OBJECT-TYPE
    SYNTAX      InetAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the IPv4 (RFC 791) or IPv6 (RFC 2460) destination
        address parameter that must be matched to identify packets
        coming up from lower layers."
    REFERENCE   "9.1.5.5"
    ::= { ieee8021StreamIdIpStreamIdentificationEntry 7 }

ieee8021StreamIdCpeIpIdDscp OBJECT-TYPE
    SYNTAX      Ieee8021CBIpDscpType
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the IPv4 (RFC 791) or IPv6 (RFC 2460) differentiated
         services codepoint (DSCP, RFC 2474) that must be matched to
         identify packets coming up from the lower layers. A value of
         64 decimal indicates that the DSCP is to be ignored on packets
         received from lower layers."
    REFERENCE   "9.1.5.6"
    ::= { ieee8021StreamIdIpStreamIdentificationEntry 8 }

ieee8021StreamIdCpeIpIdNextProtocol OBJECT-TYPE
    SYNTAX      Ieee8021CBIdNextProtocolType
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the IP next protocol parameter that must be matched
         to identify packets coming up from lower layers. The value of
         this parameter must specify either none, UDP (RFC 768),
         TCP (RFC 793), or SCTP (RFC 4960). If 'none', then the
         tsnCpeIpIdSourcePort and tsnCpeIpIdDestinationPort managed
         objects are not used."
    REFERENCE   "9.1.5.7"
    ::= { ieee8021StreamIdIpStreamIdentificationEntry 9 }

ieee8021StreamIdCpeIpIdSourcePort OBJECT-TYPE
    SYNTAX      InetPortNumber
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the TCP or UDP Source Port parameter that must be
         matched to identify packets coming up from lower layers.
         A value of 0 indicates that the Source Port number of the
         packet is to be ignored on packets received from lower layers."
    REFERENCE   "9.1.5.8"
    ::= { ieee8021StreamIdIpStreamIdentificationEntry 10 }

ieee8021StreamIdCpeIpIdDestinationPort OBJECT-TYPE
    SYNTAX      InetPortNumber
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the TCP or UDP Destination Port parameter that must
         be matched to identify packets coming up from lower layers.
         A value of 0 indicates that the Destination Port number of the
         packet is to be ignored on packets received from lower layers."
```

```
    REFERENCE   "9.1.5.9"
    ::= { ieee8021StreamIdIpStreamIdentificationEntry 11 }

-- ============================================================
-- the ieee8021StreamIdMaskAndMatchIdentificationTable
-- ============================================================

ieee8021StreamIdMaskAndMatchIdentificationTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdMaskAndMatchIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of controlling parameters for the
         Mask-and-match Stream identification method."
    REFERENCE   "9.1.6"
    ::= { ieee8021StreamIdStreamIdentification 7 }

ieee8021StreamIdMaskAndMatchIdentificationEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdMaskAndMatchIdentificationEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects that serve as the Stream
         identification parameters when using the Mask-and-match Stream
         identification method."
    REFERENCE   "9.1.6"
    INDEX  { ieee8021StreamIdStreamIdentificationIndex }
    ::= { ieee8021StreamIdMaskAndMatchIdentificationTable 1 }

Ieee8021StreamIdMaskAndMatchIdentificationEntry ::=
    SEQUENCE {
        ieee8021StreamIdCpeMmIdDestMacMask
            MacAddress,
        ieee8021StreamIdCpeMmIdDestMacMatch
            MacAddress,
        ieee8021StreamIdCpeMmIdSrcMacMask
            MacAddress,
        ieee8021StreamIdCpeMmIdSrcMacMatch
            MacAddress,
        ieee8021StreamIdCpeMmIdMsduMaskLength
            Integer32,
        ieee8021StreamIdCpeMmIdMsduMask
            OCTET STRING,
        ieee8021StreamIdCpeMmIdMsduMatch
            OCTET STRING
    }

ieee8021StreamIdCpeMmIdDestMacMask OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies a 48-bit mask. A bitwise AND operation is performed
         between MmIdDestMacMask and the destination address parameter
         passed by the ISS indication primitive to the Mask-and-match
         Stream identification function. The resulting 48-bit
         information is the masked destination address that is used as
         input for the instance of the Mask-and-match Stream
         identification function. If MmIdDestMacMask has a value of 0,
```

```
        the destination_address parameter is ignored."
    REFERENCE   "9.1.6.1"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 1 }

ieee8021StreamIdCpeMmIdDestMacMatch OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the 48-bit value of the masked destination_address,
        as defined in 9.1.6.1, to be matched by the instance of the
        Mask-and-match Stream identification function."
    REFERENCE   "9.1.6.2"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 2 }

ieee8021StreamIdCpeMmIdSrcMacMask OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies a 48-bit mask. A bitwise AND operation is performed
        between CpeMmIdSrcMacMask and the source_address parameter
        passed by the ISS indication primitive to the Mask-and-match
        Stream identification function. The resulting 48-bit
        information is the masked source_address that is used as input
        for the instance of the Mask-and-match Stream identification
        function. If CpeMmIdSrcMacMask has a value of 0, the
        source_address parameter is ignored."
    REFERENCE   "9.1.6.3"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 3 }

ieee8021StreamIdCpeMmIdSrcMacMatch OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the 48-bit value of the masked source_address, as
        defined in 9.1.6.3, to be matched by the instance of the
        Mask-and-match Stream identification function."
    REFERENCE   "9.1.6.4"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 4 }

ieee8021StreamIdCpeMmIdMsduMaskLength OBJECT-TYPE
    SYNTAX      Integer32(2..1984)
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Is an integer value that specifies the length, in octets, of
        CpeMmIdMsduMask (9.1.6.6) and CpeMmIdMsduMatch (9.1.6.7). The
        behavior of the Mask-and-match Stream identification function
        is undefined on any port where the value of
        CpeMmIdMsduMaskLength exceeds the per-port
        mac_service_data_unit_mask_maximum_length(9.4.1). If the
        length of the frame's mac_service_data_unit is shorter than
        CpeMmIdMsduMaskLength, then the identification fails."
    REFERENCE   "9.1.6.5"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 5 }

ieee8021StreamIdCpeMmIdMsduMask OBJECT-TYPE
```

```
    SYNTAX      OCTET STRING (SIZE(2..1984))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies a mask of CpeMmIdMsduMaskLength octets. A bitwise
         AND operation is performed between CpeMmIdMsduMask and the
         CpeMmIdMsduMaskLength (9.1.6.6) first octets of the
         mac_service_data_unit parameter passed by the ISS indication
         primitive to the Mask-and-match Stream identification
         function. The resulting CpeMmIdMsduMaskLength octets of
         information are the masked mac_service_data_unit that is used
         as input for the instance of the Mask-and-match Stream
         identification function. If CpeMmIdMsduMask has a value of 0,
         the mac_service_data_unit parameter is ignored."
    REFERENCE   "9.1.6.6"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 6 }

ieee8021StreamIdCpeMmIdMsduMatch OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(2..1984))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Specifies the value of the masked mac_service_data_unit, as
         defined in 9.1.6.5 and 9.1.6.6, to be matched by the instance
         of the Mask-and-match Stream identification function."
    REFERENCE   "9.1.6.7"
    ::= { ieee8021StreamIdMaskAndMatchIdentificationEntry 7 }

-- ============================================================
-- the ieee8021StreamIdStreamIdInFacOutputPortHandleList
-- table
-- ============================================================
ieee8021StreamIdStreamIdInFacOutputPortHandleListTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF Ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a list of ports on which an in-facing Stream
         identification function using this identification method is to
         be placed for this Stream in the output (towards the system
         forwarding function) direction, referenced in
         ieee8021StreamIdStreamIdentificationIndex."
    REFERENCE   "9.1.1.2"
    ::= { ieee8021StreamIdStreamIdInFacOutputPortHandleList 2 }

ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects providing the ports on which an
         in-facing Stream identification function using this
         identification method is to be placed for this Stream in the
         output (towards the system forwarding function) direction,
         referenced in ieee8021StreamIdStreamIdentificationIndex."
    REFERENCE   "9.1.1.2"
    INDEX   { ieee8021StreamIdStreamIdentificationIndex,
              ieee8021StreamIdStreamIdInFacOutputPortHandleListIndex
            }
```

```
    ::= { ieee8021StreamIdStreamIdInFacOutputPortHandleListTable 1 }

Ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry ::=
    SEQUENCE {
        ieee8021StreamIdStreamIdInFacOutputPortHandleListIndex
            Unsigned32,
        ieee8021StreamIdStreamIdInFacOutputPortHandle
            VariablePointer,
        ieee8021StreamIdStreamIdInFacOutputPortHandleListStatus
            RowStatus
    }

ieee8021StreamIdStreamIdInFacOutputPortHandleListIndex OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Index for the In-Facing Output Port handle list table."
    ::= { ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry 1 }

ieee8021StreamIdStreamIdInFacOutputPortHandle OBJECT-TYPE
    SYNTAX      VariablePointer
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A pointer to an interface that is an element of the
         ieee8021StreamIdStreamIdInFacOutputPortList instance in the
         ieee8021StreamIdStreamIdentificationTable."
    REFERENCE   "9.1.1.2"
    ::= { ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry 2 }

ieee8021StreamIdStreamIdInFacOutputPortHandleListStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Row-Status for the In-Facing Output Port handle list table."
    ::= { ieee8021StreamIdStreamIdInFacOutputPortHandleListEntry 3 }

-- ============================================================
-- the ieee8021StreamIdStreamIdOutFacOutputPortList
-- table
-- ============================================================
ieee8021StreamIdStreamIdOutFacOutputPortHandleListTable OBJECT-TYPE
   SYNTAX  SEQUENCE OF Ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a list of ports on which an out-facing
         Stream identification function using this identification method
         is to be placed for this Stream in the output (towards the
         system forwarding function) direction, referenced in
         ieee8021StreamIdStreamIdentificationIndex."
    REFERENCE   "9.1.1.3"
    ::= { ieee8021StreamIdStreamIdOutFacOutputPortHandleList 3 }

ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
    DESCRIPTION
        "A set of managed objects providing the ports on which an
         out-facing Stream identification function using this
         identification method is to be placed for this Stream in the
         output (towards the system forwarding function) direction,
         referenced in ieee8021StreamIdStreamIdentificationIndex."
    REFERENCE   "9.1.1.3"
    INDEX   { ieee8021StreamIdStreamIdentificationIndex,
              ieee8021StreamIdStreamIdOutFacOutputPortHandleListIndex
            }
    ::= { ieee8021StreamIdStreamIdOutFacOutputPortHandleListTable 1 }

Ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry ::=
    SEQUENCE {
        ieee8021StreamIdStreamIdOutFacOutputPortHandleListIndex
            Unsigned32,
        ieee8021StreamIdStreamIdOutFacOutputPortHandle
            VariablePointer,
        ieee8021StreamIdStreamIdOutFacOutputPortHandleListStatus
            RowStatus
    }

ieee8021StreamIdStreamIdOutFacOutputPortHandleListIndex OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Index for the Out-Facing Output Port handle list table."
    ::= { ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry 1 }

ieee8021StreamIdStreamIdOutFacOutputPortHandle OBJECT-TYPE
    SYNTAX      VariablePointer
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A pointer to an interface that is an element of the
         ieee8021StreamIdStreamIdOutFacOutputPortList instance in the
         ieee8021StreamIdStreamIdentificationTable."
    REFERENCE   "9.1.1.3"
    ::= { ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry 2 }

ieee8021StreamIdStreamIdOutFacOutputPortHandleListStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Row-Status for the Out-Facing Output Port handle list table."
    ::= { ieee8021StreamIdStreamIdOutFacOutputPortHandleListEntry 3 }

-- ============================================================
-- the ieee8021StreamIdStreamIdInFacInputPortList
-- table
-- ============================================================
ieee8021StreamIdStreamIdInFacInputPortHandleListTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ieee8021StreamIdStreamIdInFacInputPortHandleListEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
```

```
            "A table containing a list of ports on which an in-facing Stream
             identification function using this identification method is to
             be placed for this Stream in the input (coming from the system
             forwarding function) direction, referenced in
             ieee8021StreamIdStreamIdentificationIndex."
        REFERENCE   "9.1.1.4"
        ::= { ieee8021StreamIdStreamIdInFacInputPortHandleList 4 }

    ieee8021StreamIdStreamIdInFacInputPortHandleListEntry OBJECT-TYPE
        SYNTAX      Ieee8021StreamIdStreamIdInFacInputPortHandleListEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "A set of managed objects providing the ports on which an
             in-facing Stream identification function using this
             identification method is to be placed for this Stream in the
             input (coming from the system forwarding function) direction,
             referenced in ieee8021StreamIdStreamIdentificationIndex."
        REFERENCE   "9.1.1.4"
        INDEX   { ieee8021StreamIdStreamIdentificationIndex,
                  ieee8021StreamIdStreamIdInFacInputPortHandleListIndex
                }
        ::= { ieee8021StreamIdStreamIdInFacInputPortHandleListTable 1 }

    Ieee8021StreamIdStreamIdInFacInputPortHandleListEntry ::=
        SEQUENCE {
            ieee8021StreamIdStreamIdInFacInputPortHandleListIndex
                Unsigned32,
            ieee8021StreamIdStreamIdInFacInputPortHandle
                VariablePointer,
            ieee8021StreamIdStreamIdInFacInputPortHandleListStatus
                RowStatus
        }

    ieee8021StreamIdStreamIdInFacInputPortHandleListIndex OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "Index for the In-Facing Input Port handle list table."
        ::= { ieee8021StreamIdStreamIdInFacInputPortHandleListEntry 1 }

    ieee8021StreamIdStreamIdInFacInputPortHandle OBJECT-TYPE
        SYNTAX      VariablePointer
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "A pointer to an interface that is an element of the
             ieee8021StreamIdStreamIdOutFacInputPortList instance in the
             ieee8021StreamIdStreamIdentificationTable."
        REFERENCE   "9.1.1.4"
        ::= { ieee8021StreamIdStreamIdInFacInputPortHandleListEntry 2 }

    ieee8021StreamIdStreamIdInFacInputPortHandleListStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "Row-Status for the In-Facing Input Port handle list table."
```

```
    ::= { ieee8021StreamIdStreamIdInFacInputPortHandleListEntry 3 }

-- ================================================================
-- the ieee8021StreamIdStreamIdOutFacInputPortList
-- table
-- ================================================================
ieee8021StreamIdStreamIdOutFacInputPortHandleListTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF Ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a list of ports on which an out-facing
         Stream identification function using this identification method
         is to be placed for this Stream in the input (coming from the
         physical interface) direction, referenced in
         ieee8021StreamIdStreamIdentificationIndex."
    REFERENCE   "9.1.1.5"
    ::= { ieee8021StreamIdStreamIdOutFacInputPortHandleList 5 }

ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of managed objects providing the ports on which an
         out-facing Stream identification function using this
         identification method is to be placed for this Stream in the
         input (coming from the physical interface) direction,
         referenced in ieee8021StreamIdStreamIdentificationIndex."
    REFERENCE   "9.1.1.5"
    INDEX   { ieee8021StreamIdStreamIdentificationIndex,
              ieee8021StreamIdStreamIdOutFacInputPortHandleListIndex
            }
    ::= { ieee8021StreamIdStreamIdOutFacInputPortHandleListTable 1 }

Ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry ::=
    SEQUENCE {
        ieee8021StreamIdStreamIdOutFacInputPortHandleListIndex
            Unsigned32,
        ieee8021StreamIdStreamIdOutFacInputPortHandle
            VariablePointer,
        ieee8021StreamIdStreamIdOutFacInputPortHandleListStatus
            RowStatus
    }

ieee8021StreamIdStreamIdOutFacInputPortHandleListIndex OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Index for the Out-Facing Input Port handle list table."
    ::= { ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry 1 }

ieee8021StreamIdStreamIdOutFacInputPortHandle OBJECT-TYPE
    SYNTAX      VariablePointer
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A pointer to an interface that is an element of the
```

```
            ieee8021StreamIdStreamIdOutFacInputPortList instance in the
            ieee8021StreamIdStreamIdentificationTable."
        REFERENCE   "9.1.1.5"
        ::= { ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry 2 }

    ieee8021StreamIdStreamIdOutFacInputPortHandleListStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "Row-Status for the Out-Facing Input Port handle list table."
        ::= { ieee8021StreamIdStreamIdOutFacInputPortHandleListEntry 3 }

    -- ================================================================
    -- the ieee8021StreamIdPerPortPerStreamCountersTable
    -- ================================================================

    ieee8021StreamIdPerPortPerStreamCountersTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF Ieee8021StreamIdPerPortPerStreamCountersEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
                "A table containing a set of counters for Stream
                 identification that are instantiated per port, Stream, and
                 direction."
        REFERENCE   "9.2"
        ::= { ieee8021StreamIdPerPortPerStreamCounters 6 }

    ieee8021StreamIdPerPortPerStreamCountersEntry OBJECT-TYPE
        SYNTAX      Ieee8021StreamIdPerPortPerStreamCountersEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
                "A set of managed objects for Stream identification
                 counters that are instantiated per port, Stream, and
                 direction."
        REFERENCE   "9.2"
         INDEX  { ifIndex,
                  ieee8021StreamIdStreamIdHandle,
                  ieee8021StreamIdPerPortPerStreamDirection
                }
        ::= { ieee8021StreamIdPerPortPerStreamCountersTable 1 }

    Ieee8021StreamIdPerPortPerStreamCountersEntry ::=
        SEQUENCE {
            ieee8021StreamIdPerPortPerStreamDirection
                TruthValue,
            ieee8021StreamIdPerPortPerStreamInputPackets
                Counter64,
            ieee8021StreamIdPerPortPerStreamOutputPackets
                Counter64
        }

    ieee8021StreamIdPerPortPerStreamDirection OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "A managed object specifying the facing of a given Stream
```

```
            on a port. The Stream can be either in-facing (False) or
            out-facing (True)."
        REFERENCE   "9.2"
        ::= { ieee8021StreamIdPerPortPerStreamCountersEntry 1 }

    ieee8021StreamIdPerPortPerStreamInputPackets OBJECT-TYPE
        SYNTAX      Counter64
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "A managed object serving as a counter that is incremented once
             for each packet identified by the Stream identification
             function."
        REFERENCE   "9.2.1"
        ::= { ieee8021StreamIdPerPortPerStreamCountersEntry 2 }

    ieee8021StreamIdPerPortPerStreamOutputPackets OBJECT-TYPE
        SYNTAX      Counter64
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "A managed object serving as a counter that is incremented once
             for each packet passed down the stack by the Stream
             identification function."
        REFERENCE   "9.2.2"
        ::= { ieee8021StreamIdPerPortPerStreamCountersEntry 3 }

    -- ================================================================
    -- the ieee8021StreamIdPerPortCountersTable
    -- ================================================================

    ieee8021StreamIdPerPortCountersTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF Ieee8021StreamIdPerPortCountersEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
                "A table containing a set of counters for Stream
                 identification that are instantiated per port and
                 direction."
        REFERENCE   "9.3"
        ::= { ieee8021StreamIdPerPortCounters 7 }

    ieee8021StreamIdPerPortCountersEntry OBJECT-TYPE
        SYNTAX      Ieee8021StreamIdPerPortCountersEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
                "A a set of managed objects for Stream identification
                 counters that are instantiated per port and direction."
        REFERENCE   "9.3"
         INDEX  { ifIndex }
        ::= { ieee8021StreamIdPerPortCountersTable 1 }

    Ieee8021StreamIdPerPortCountersEntry ::=
        SEQUENCE {
            ieee8021StreamIdPerPortInputPackets
                Counter64,
            ieee8021StreamIdPerPortOutputPackets
                Counter64
```

```
    }

ieee8021StreamIdPerPortInputPackets OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A managed object serving as a counter that is incremented once
         for each packet identified by any Stream identification
         function on this port. Its value equals the sum (modulo the
         size of the counters) of all of the
         ieee8021StreamIdPerPortPerStreamInputPackets counters on this
         same port."
    REFERENCE   "9.3.1"
    ::= { ieee8021StreamIdPerPortCountersEntry 1 }

ieee8021StreamIdPerPortOutputPackets OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A managed object serving as a counter that is incremented once
         for each packet passed down the stack by any Stream
         identification function on this port. Its value equals the sum
         (modulo the size of the counters) of all of the
         ieee8021StreamIdPerPortPerStreamOutputPackets counters on this
         same port."
    REFERENCE   "9.3.2"
    ::= { ieee8021StreamIdPerPortCountersEntry 2 }

-- ================================================================
-- the ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthTable
-- ================================================================

ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthTable OBJECT-TYPE
   SYNTAX     SEQUENCE OF Ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthEn-
try
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing a set of maximum msdu mask length
         parameters that are defined per port."
    REFERENCE   "9.4.1"
    ::= { ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLength 8 }

ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthEntry OBJECT-TYPE
    SYNTAX      Ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A a set of managed objects specifying the maximum msdu
         mask length per port, applicable to all the instances
         of Mask-And-Match Stream identification functions"
    REFERENCE   "9.4.1"
    INDEX  { ifIndex }
    ::= { ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthTable 1 }

Ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthEntry ::=
    SEQUENCE {
```

```
        ieee8021StreamIdMaskAndMatchMsduMaskMaxLength
        Integer32
    }

ieee8021StreamIdMaskAndMatchMsduMaskMaxLength OBJECT-TYPE
    SYNTAX      Integer32(2..1984)
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "A managed object specifying the maximum msdu mask length for
         a given port."
    REFERENCE   "9.4.1"
    ::= { ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthEntry 1 }

-- ============================================================
-- IEEE802 STREAM IDENTIFICATION MIB - Conformance Information
-- ============================================================

ieee8021StreamIdCompliances
    OBJECT IDENTIFIER ::= { ieee8021StreamIdConformance 1 }
ieee8021StreamIdGroups
    OBJECT IDENTIFIER ::= { ieee8021StreamIdConformance 2 }

-- ============================================================
-- units of conformance
-- ============================================================

-- ============================================================
-- the ieee8021StreamIdStreamIdentification group
-- ============================================================

ieee8021StreamIdStreamIdentificationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdStreamIdIdentificationTypeSelect,
        ieee8021StreamIdStreamIdIdentificationType,
        ieee8021StreamIdStreamIdIdentificationTypeOUI,
        ieee8021StreamIdStreamIdIdentificationCustomType,
        ieee8021StreamIdStreamIdIdentificationCustomTypeOUI,
        ieee8021StreamIdStreamIdHandle,
        ieee8021StreamIdStreamIdInFacOutputPortList,
        ieee8021StreamIdStreamIdOutFacOutputPortList,
        ieee8021StreamIdStreamIdInFacInputPortList,
        ieee8021StreamIdStreamIdOutFacInputPortList,
        ieee8021StreamIdStreamIdInFacOutputPortHandle,
        ieee8021StreamIdStreamIdInFacOutputPortHandleListStatus,
        ieee8021StreamIdStreamIdOutFacOutputPortHandle,
        ieee8021StreamIdStreamIdOutFacOutputPortHandleListStatus,
        ieee8021StreamIdStreamIdInFacInputPortHandle,
        ieee8021StreamIdStreamIdInFacInputPortHandleListStatus,
        ieee8021StreamIdStreamIdOutFacInputPortHandle,
        ieee8021StreamIdStreamIdOutFacInputPortHandleListStatus
    }
    STATUS      current
    DESCRIPTION
        "Objects that are part of the Stream identification."
    ::= { ieee8021StreamIdGroups 1 }

-- ============================================================
-- the ieee8021StreamIdNullStreamIdentification group
```

```
-- ================================================================

ieee8021StreamIdNullStreamIdentificationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdCpeNullDownDestMac,
        ieee8021StreamIdCPENullDownTagged,
        ieee8021StreamIdCpeNullDownVlan
    }
    STATUS      current
    DESCRIPTION
        "Objects that are part of the null Stream identification
         method."
    ::= { ieee8021StreamIdGroups 2 }

-- ================================================================
-- the ieee8021StreamIdSrcMacVlanIdentification group
-- ================================================================

ieee8021StreamIdSrcMacVlanIdentificationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdCpeSmacVlanDownSrcMac,
        ieee8021StreamIdCpeSmacVlanDownTagged,
        ieee8021StreamIdCpeSmacVlanDownVlan
    }
    STATUS      current
    DESCRIPTION
        "Objects that are part of the Source MAC and VLAN Stream
         identification method."
    ::= { ieee8021StreamIdGroups 3 }

-- ================================================================
-- the ieee8021StreamIdActiveDestMacVlanIdentification group
-- ================================================================

ieee8021StreamIdActiveDestMacVlanIdentificationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdCpeDmacVlanDownDestMac,
        ieee8021StreamIdCpeDmacVlanDownTagged,
        ieee8021StreamIdCpeDmacVlanDownVlan,
        ieee8021StreamIdCpeDmacVlanDownPriority,
        ieee8021StreamIdCpeDmacVlanUpDestMac,
        ieee8021StreamIdCpeDmacVlanUpTagged,
        ieee8021StreamIdCpeDmacVlanUpVlan,
        ieee8021StreamIdCpeDmacVlanUpPriority
    }
    STATUS      current
    DESCRIPTION
        "Objects that are part of the Active Destination MAC and
         VLAN Stream identification method."
    ::= { ieee8021StreamIdGroups 4 }

-- ================================================================
-- the ieee8021StreamIdIpStreamIdentification group
-- ================================================================

ieee8021StreamIdIpStreamIdentificationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdCpeIpIdDestMac,
        ieee8021StreamIdCpeIpIdTagged,
```

```
        ieee8021StreamIdCpeIpIdVlan,
        ieee8021StreamIdCpeIpIdIpSourceType,
        ieee8021StreamIdCpeIpIdIpSource,
        ieee8021StreamIdCpeIpIdIpDestinationType,
        ieee8021StreamIdCpeIpIdIpDestination,
        ieee8021StreamIdCpeIpIdDscp,
        ieee8021StreamIdCpeIpIdNextProtocol,
        ieee8021StreamIdCpeIpIdSourcePort,
        ieee8021StreamIdCpeIpIdDestinationPort
    }
    STATUS      current
    DESCRIPTION
        "Objects that are part of the IP Stream identification method."
    ::= { ieee8021StreamIdGroups 5 }

-- ================================================================
-- the ieee8021StreamIdAutoConfiguration group
-- ================================================================

ieee8021StreamIdAutoConfigurationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdAutoConfigured,
        ieee8021StreamIdLanPathId,
        ieee8021StreamIdStatus
    }
    STATUS      current
    DESCRIPTION
        "Objects that are used if auto configuration for Streams is
        used."
    ::= { ieee8021StreamIdGroups 6 }

-- ================================================================
-- the ieee8021StreamIdPerPortPerStreamCounters group
-- ================================================================

ieee8021StreamIdPerPortPerStreamCountersGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdPerPortPerStreamInputPackets,
        ieee8021StreamIdPerPortPerStreamOutputPackets
    }
    STATUS      current
    DESCRIPTION
        "Objects that provide information on the count of packets
         handled by Stream identification per port, Stream, and
         direction ."
    ::= { ieee8021StreamIdGroups 7 }

-- ================================================================
-- the ieee8021StreamIdPerPortCounters group
-- ================================================================

ieee8021StreamIdPerPortCountersGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdPerPortInputPackets,
        ieee8021StreamIdPerPortOutputPackets
    }
    STATUS      current
    DESCRIPTION
        "Objects that provide information on the count of packets
```

```
            handled by Stream identification per port and direction."
        ::= { ieee8021StreamIdGroups 8 }

-- ============================================================
-- the ieee8021StreamIdMaskAndMatchIdentification group
-- ============================================================

ieee8021StreamIdMaskAndMatchIdentificationGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdCpeMmIdDestMacMask,
        ieee8021StreamIdCpeMmIdDestMacMatch,
        ieee8021StreamIdCpeMmIdSrcMacMask,
        ieee8021StreamIdCpeMmIdSrcMacMatch,
        ieee8021StreamIdCpeMmIdMsduMaskLength,
        ieee8021StreamIdCpeMmIdMsduMask,
        ieee8021StreamIdCpeMmIdMsduMatch
    }
    STATUS      current
    DESCRIPTION
        "Objects that are part of the Mask-and-Match Stream
         identification method."
    ::= { ieee8021StreamIdGroups 9 }

-- ============================================================
-- the ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLength group
-- ============================================================

ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthGroup OBJECT-GROUP
    OBJECTS {
        ieee8021StreamIdMaskAndMatchMsduMaskMaxLength
    }
    STATUS      current
    DESCRIPTION
        "Objects that provide information on the maximum msdu mask
         handled by the Mask-and-match Stream identification function
         per port."
    ::= { ieee8021StreamIdGroups 10 }

-- ============================================================
-- compliance statements
-- ============================================================

ieee8021StreamIdCompliance MODULE-COMPLIANCE
    STATUS      current
    DESCRIPTION
        "The compliance statement for devices supporting
        Stream identification."

    MODULE -- this module
        MANDATORY-GROUPS {
            ieee8021StreamIdStreamIdentificationGroup,
            ieee8021StreamIdNullStreamIdentificationGroup,
            ieee8021StreamIdSrcMacVlanIdentificationGroup,
            ieee8021StreamIdActiveDestMacVlanIdentificationGroup,
            ieee8021StreamIdIpStreamIdentificationGroup,
            ieee8021StreamIdPerPortPerStreamCountersGroup,
            ieee8021StreamIdPerPortCountersGroup,
            ieee8021StreamIdMaskAndMatchIdentificationGroup,
            ieee8021StreamIdMaskAndMatchPerPortMsduMaskMaxLengthGroup
```

```
        }

        GROUP ieee8021StreamIdAutoConfigurationGroup
        DESCRIPTION
            "Implementation of this group is mandatory if the
             auto configuration feature is implemented."

    ::= { ieee8021StreamIdCompliances 1 }

END
```

Clause 12 was added by IEEE Std 802.1CBcv-2021.

12. YANG Data Model

12.2 IEEE Std 802.1CB YANG model

12.2.1 Stream identification model

Insert the following note after the first paragraph of 12.2.1 and before Figure 12-3:

NOTE—The {XOR} UML constraint denotes that the stream-identity parameters can be either one of the identification-types described on the right-hand side of the diagram.

Replace Figure 12-3 with the following figure (with darker-grey background boxes added):

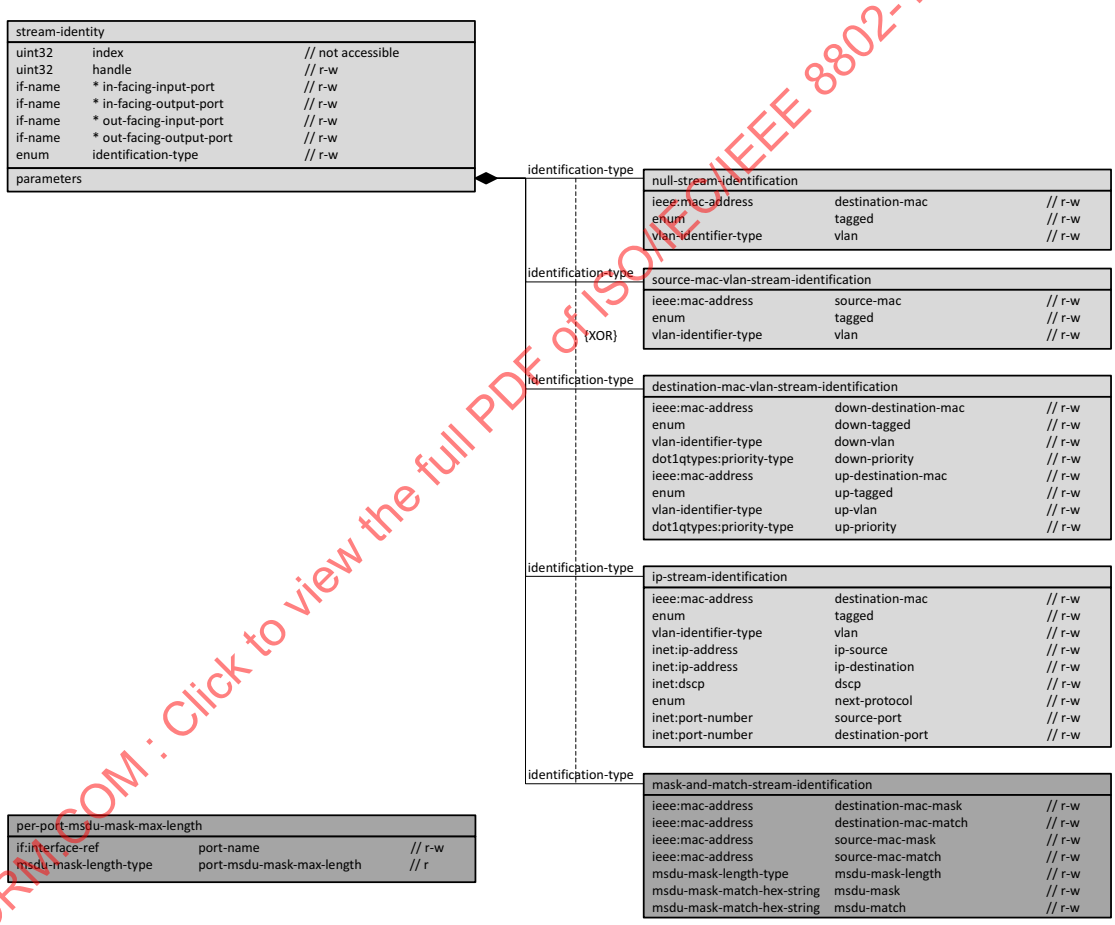

Figure 12-3—Stream identification model

12.3 Structure of the YANG model

Change Table 12-1 as follows:

Table 12-1—Structure of the YANG modules

Module	References	Notes
ieee802-dot1cb-stream-identification-types	12.6.2.1	General type definitions used by IEEE Std 802.1CB Stream identification.
ieee802-dot1cb-stream-identification	12.6.2.2	YANG model for Stream identification.
ieee802-dot1cb-mask-and-match	12.6.2.2a	YANG model for the extended Stream identification.
ieee802-dot1cb-frer-types	12.6.2.3	General type definitions used by IEEE Std 802.1CB frame replication and elimination for reliability.
ieee802-dot1cb-frer	12.6.2.4	YANG model for FRER.

Insert new subclause 12.3.1a after 12.3.1 as follows:

12.3.1a Structure of the ieee802-dot1cb-mask-and-match YANG module

The Extended Stream Identification model consists of the *ieee802-dot1cb-mask-and-match* YANG module along with all the dependencies (YANG imports) that the module uses. 12.6.1.1a contains the YANG data schema tree for the *ieee802-dot1cb-mask-and-match* module.

The high-level structure of the *ieee802-dot1cb-mask-and-match* YANG module is found in Table 12-3a.

The list of YANG modules directly imported by the *ieee802-dot1cb-mask-and-match* YANG module is found in Table 12-3b.

Table 12-3a—ieee802-dot1cb-mask-and-match structure and relationship to this standard

Module	References	Notes
ieee802-dot1cb-mask-and-match		—
stream-identity parameters	9.1.6	Mask-and-match Stream identification parameters.
per-port-msdu-mask-max-length	9.4	Per-port Stream identification parameter.

Table 12-3b—YANG module dependencies for the Extended Stream identification model

YANG module
ieee802-types
ietf-interfaces
ieee802-dot1cb-stream-identification-types
ieee802-dot1cb-stream-identification