DDCFM contribution

Draft Standard for
Local and Metropolitan Area Networks—

Virtual Bridged Local Area Networks — Amendment ?: Management of data driven and data dependent connectivity faults

Sponsor

LAN MAN Standards Committee of the IEEE Computer Society

This is an individual contribution in anticipation of the future approval of a proposed project and has no official standing whatsoever. The intent of this contribution is to provide supporting information on the integration of the proposed project into 802.1Q and its amendments. It has not been prepared by the Interworking Task Group of IEEE 802.1

Abstract: This amendment extends the connectivity fault management capabilities introduced in P802.1ag to include diagnosis and isolation of faults sensitive to, or caused by, particular data patterns in frames transmitted by a service user.

See the introductory notes for what this draft tries to accomplish, and for project scope, status, and history.

Keywords: For keywords refer to the title page proper, following the editors' foreword.

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.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

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. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, 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 publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received 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 interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: 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 with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

IEEE is the sole entity that may authorize the use of certification marks, trademarks, or other designations to indicate compliance with the materials set forth herein.

Editors' Foreword

<<Notes>>

<<Throughout this document, all notes such as this one, presented between angle braces, are temporary notes inserted by the Editors for a variety of purposes; these notes and the Editors' Foreword will all be removed prior to publication and are not part of the normative text.>>

<<Comments and participation in 802.1 standards development

Comments on this draft are encouraged. **PLEASE NOTE: All issues related to IEEE standards presentation style, formatting, spelling, etc. are routinely handled between the 802.1 Editor and the IEEE Staff Editors prior to publication, after balloting and the process of achieving agreement on the technical content of the standard is complete.** Readers are urged to devote their valuable time and energy only to comments that materially affect either the technical content of the document or the clarity of that technical content. Comments should not simply state what is wrong, but also what might be done to fix the problem.

Full participation in the development of this draft requires individual attendance at IEEE 802 meetings. Information on 802.1 activities, working papers, and email distribution lists etc. can be found on the 802.1 website:

http://ieee802.org/1/

Use of the email distribution list is not presently restricted to 802.1 members, and the working group has had a policy of considering ballot comments from all who are interested and willing to contribute to the development of the draft. Individuals not attending meetings have helped to identify sources of misunderstanding and ambiguity in past projects. Non-members are advised that the email lists exist primarily to allow the members of the working group to develop standards, and are not a general forum.

Comments on this document may be sent to the 802.1 email exploder, to the editor, or to the Chairs of the 802.1 Working Group and Link Security Task Group.

This draft was prepared by:

Linda Dunbar
Huawei Technologies Inc.(US)
1700 Alma Dr., Suite 100
Plano, TX 75075
USA
Email:ldunbar@huawei.com

Mick Seaman
160 Bella Vista Ave
Belvedere
CA 94920
USA
Email:mick_seaman@ieee.org

Chairs of the 802.1 Working Group and Interworking Task Group.

Mick Seaman
Chair, 802.1 Interworking Task Group
160 Bella Vista Ave
Belvedere
CA 94041
USA
Email:mick_seaman@ieee.org

Tony Jeffree
Chair, 802.1 Working Group
11A Poplar Grove
Sale
Cheshire
M33 3AX
UK
+44 161 973 4278 (Tel)
+44 161 973 6534 (Fax)
Email: tony@jeffree.co.uk

PLEASE NOTE: Comments whose distribution is restricted in any way cannot be considered, and may not be acknowledged.
>>

<<The draft text and accompanying information

This document currently comprises:

— A temporary cover page, preceding the Editors' Forewords. This cover page will be removed following working group approval of this draft, i.e. prior to sponsor ballot.

— IEEE boilerplate text.

— The editors' forewords, including this text. These include an unofficial and informal appraisal of history and status, introductory notes to each draft that summarize the progress and focus of each successive draft, and requests for comments and contributions on major issues.

— A title page for the proposed standard including an Abstract and Keywords. This title page will be retained following approval.

— IEEE boilerplate text (identical to the above).

— The introduction to this standard.

— A record of participants (not included in early drafts but added prior to publication).

— The proposed revision proper.

— An Annex Z comprising the editors' discussion of issues. This annex will be deleted from the document prior to sponsor ballot.

IEEE P802.1 editors have a responsibility to attempt to craft technically coherent drafts from the resolutions of ballot comments and the other discussions that take place in the working group meetings. Preparation of drafts often exposes inconsistencies in editors instructions or exposes the need to make choices between approaches that were not fully apparent in the meeting. Choices and requests by the editors' for contributions on specific issues will be found in the editors' introductory notes to the current draft, at appropriate points in the draft, and in Annex Z. Significant discussion of more difficult topics will be found in the last of these.

The ballot comments received on each draft, and the editors' proposed and final disposition of comments, are part of the audit trail of the development of the standard and are available, along with all the revisions of the draft on the 802.1 website (for address see above).

>>

<<History and Scope

A first draft of a PAR (Project Authorization Request) was considered at the July 2006 802.1 meeting, which authorized its September 2006 interim meeting to revise that draft and the accompanying five criteria for pre-circulation to 802.0 in anticipation of PAR approval at the November 2006 802 plenary meeting. The following PAR Scope, Reason, and Why Needed, and Five Criteria are taken from that first draft. <<This section will be updated with the final text when approved.>>.

Scope of Proposed Project:

> This standard specifies connectivity fault management protocols, procedures, and managed objects that provide confirmation of successful transmission of frames conveying specified data. This capability supports diagnosis of faults sensitive to, or caused by, particular data patterns, and their isolation to part of the transmission path. Connectivity verification can be carried out from any single point with bridged connectivity to maintenance points on the path, can isolate failures to communicate in a specific direction, and can be carried out while service is being provided to other users of the data path.

Purpose of Proposed Project:

> While bridged networks are notionally transparent to the users' data, they are often deployed as part of a service offering that selectively filters data frames (e.g. firewall functionality), automatically configures some aspect of service in response to data frames (e.g. IGMP snooping), or is supported by transmission in a data sensitive way (e.g. IEEE Std 802.3ad Link Aggregation). This standard will define the protocols (including CFM OpCodes) and managed objects required for data sensitive connectivity verification that is multi-vendor and uses the framework provided by IEEE P802.1ag Connectivity Fault Management.

WhyNeeded:

> There is considerable demand, from the service providers that currently use or plan to use IEEE 802.1 bridging standards, for diagnostic functionality that is at least equivalent to that provided by loopback for other network technologies, and operates in a broadly similar way. A straight forward application of loopback to IEEE 802.1Q networks is known to cause problems that can be hard to diagnose while not addressing complex fault scenarios, but is likely to be widely implemented in the absence of a better standard solution. The proposed amendment offers that solution.

1. Broad Market Potential

A standards project authorized by IEEE 802 shall have a broad market potential. Specifically, it shall have the potential for:

a) Broad sets of applicability.

> IEEE 802.1 bridging standards have been widely adopted by the service provider community. The proposed standard will address their need to use operate their new IEEE 802.1/IEEE 802.3 networks while retaining familiar procedures derived from past experience. The connectivity fault management capability provided by the proposed standard can be used with the minimum of management access to the equipment supporting user services, consistent with the approach developed in P802.1ag with joint membership collaboration with the ITU. As with P802.1ag as a whole, improvements in connectivity fault management and the ability to diagnose connectivity failures with no or little management access to network equipment is expected to be of utility to the broad community of IEEE Std 802.1Q users.

b) Multiple vendors and numerous users.

There is broad interest from numerous vendors in IEEE 802.1 in meeting the need expressed by multiple service provider customers needs for a CFM capability equivalent to ‘payload loopback’.

c) Balanced costs.

This capability is not expected to materially increase the cost of individual VLAN bridges that are suitable for service provider applications, and in part standardization is required so that two specific CFM OpCodes can be defined as being ignored by bridges that simply have to forward diagnostic traffic.

2. Compatibility

IEEE 802 defines a family of standards. All standards shall be in conformance with the IEEE 802.1 Architecture, Management and Internetworking documents as follows: 802. Overview and Architecture, 802.1D, 802.1Q and parts of 802.1f. If any variances in conformance emerge, they shall be thoroughly disclosed and reviewed with 802.

Each standard in the IEEE 802 family of standards shall include a definition of managed objects which are compatible with systems management standards.

This amendment will not change the conformance of IEEE Std 802.1Q to Std 802. Overview and Architecture, or its relationship to that specification.

Equipment conforming to the proposed amendment to IEEE Std 802.1Q will be compatible and interoperable with bridge implementations that conform to IEEE Std 802.1D and prior revisions of IEEE Std 802.1Q, and support of existing network configurations will be retained in parallel with use of the additional capabilities provided by this amendment. No change to end stations will be required to take advantage of these capabilities.

This amendment will include extensions to MIBs, existing or under development as part of other 802.1 projects, to allow management of DDCFM as a natural extension of existing capabilities.

3. Distinct Identity

Each IEEE 802 standard shall have a distinct identity. To achieve this, each authorized project shall be:

a) Substantially different from other IEEE 802 standards

IEEE Std 802.1Q is the sole and authoritative specification for VLANs and VLAN-aware Bridges, and for Connectivity Fault Management of networks constructed using that technology.

b) One unique solution per problem (not two solutions to a problem).

The proposed amendment will extend existing VLAN technology and has not been anticipated by any other specification, in IEEE 802 or elsewhere.

c) Easy for the document reader to select the relevant specification.

IEEE Std 802.1Q is the natural reference for VLAN bridging technology, which will make the capabilities added by this amendment easy to locate.

4. Technical Feasibility

For a project to be authorized, it shall be able to show its technical feasibility. At a minimum, the proposed project shall show:

a) Demonstrated system feasibility.

The proposed amendment is based on known 802.1Q VLAN technology.

b) Proven technology, reasonable testing.

The proposed amendment is based on known 802.1Q VLAN technology.

c) Confidence in reliability.

The reliability of this solution is anticipated to be the same as that of others based on existing 802.1Q VLAN technology.

d) Coexistence of 802 wireless standards specifying devices for unlicensed operation.

Not applicable.

5. Economic Feasibility

For a project to be authorized, it shall be able to show economic feasibility (so far as can reasonably be estimated), for its intended applications. At a minimum, the proposed project shall show:

a) Known cost factors, reliable data.

The proposed technology is no expected to materially alter individual VLAN Bridge equipment costs, while addressing an operational need in service provider networks that use that equipment. Relative to fostering the development of proprietary solutions with differing approaches and concepts the proposed standard will help to contain operational costs.

b) Reasonable cost for performance.

The operational practice that requires the development of the proposed standard has a long history, perceived utility, and considerable cost experience by the users' of 802.1 standards that want it supported by IEEE 802.1 conformant equipment.

c) Consideration of installation costs.

Installation costs of VLAN Bridges are not expected to be affected in any way.

>>

<<Introductory notes to the current draft

This contribution in the form of a draft reflects the information in .../docs2006/new-seaman-ddcfm-note-0306-01.pdf and .../docs2006/new-seaman-ddcfm-0306-02.pdf which were based on the discussions resulting from prior presentations by Linda Dunbar.**>>**

<<Notes to prior drafts (excerpts of continuing relevance).

At present there are no prior drafts.

>>

<<Editors' final checklist (items noted in development, to be applied to final text.

The IEEE approved format for the headings of Annexes (split across multiple lines) is not compatible with automatic production of the Table of Contents, and requires that the latter be manually edited after updating. Indeed the Framemaker user guide specifically cautions against constructing multi-line formats of the type used. Accordingly drafts of this standard use a slightly different format for Annex titles, though it contains the same information.

>>

Introduction to IEEE Std 802.1aw

This introduction is not part of IEEE Std 802.1aw-2006, IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks—Amendment?: Management of data driven and data dependent connectivity faults.

IEEE802.1aw provides additional capability to Connection Fault Management (CFM 802.1ag) to achieve data driven and data dependent connection fault management.

<<The "Introduction to IEEE Std xxx" clause found in the front matter for all IEEE 802.1 standards is not really an introduction to the standard per se, but really exists so that the revision history of the current standard and its relationship to prior editions, together with its relationship to (revisions of) companion standards can be easiy grasped. For a first edition this text is necessarily brief, and is mainly a repetition of the current scope together with a note that it is indeed the first edition.>>

Contents

IEEE P802.1??

Draft Standard for
Local and Metropolitan Area Networks—
Amendment ? to 802.1Q:
Virtual Bridged Local Area Networks

Management of data driven and data dependent connectivity faults

Editorial Note

This standard specifies changes to be applied to the base text of IEEE Std 802.1Q-2005 as amended by IEEE Std 802.1ad-2005, P802.1ag, P802.1ah, P802.1aj, P802.1ak, and P802.1aq[1]. Text shown in bold italics in this amendment defines the editing instructions necessary to changes to this base text. Three editing instructions are used: ***change***, ***delete***, and ***insert***. ***Change*** is used to make a change to existing material. The editing instruction specifies the location of the change and describes what is being changed. Changes to existing text may be clarified using ~~strikeout~~ markings to indicate removal of old material, and underscore markings to indicate addition of new material). ***Delete*** removes existing material. ***Insert*** adds new material without changing the existing material. Insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. Editorial notes will not be carried over into future editions of IEEE Std. 802.1Q.

1. Overview

Insert the following after the initial paragraphs of Clause 1.

The connectivity fault management capabities provided by this standard include include diagnosis and isolation of faults sensitive to, or caused by, particular data patterns in transmitted frames.

[1]This amendment has no strong dependencies on any of these current projects apart from P802.1ag. It is proposed not to devote significant resource to this project until P802.1ag is in sponsor ballot. As a practical matter the project editors intend to maintain a P802.1Q—Consolidated Edition as amendments are approved, to allow the effect of further amendments to be appreciated,.

1.1 Scope

Insert the following at end of subclause 1.1, relettering the bullet points so that they follow in order from those in the existing text.

This standard specifies connectivity fault management protocols, procedures, and managed objects that provide confirmation of successful transmission of frames conveying specified data. This capability supports diagnosis of faults sensitive to, or caused by, particular data patterns, and their isolation to part of the transmission path. Connectivity verification can be carried out from any single point with bridged connectivity to maintenance points on the path, can isolate failures to communicate in a specific direction, and can be carried out while service is being provided to other users of the data path. To this end it

a) Defines the extendsion to connectivity fault management capabilities introduced in P802.1ag to include diagnosis and isolation of faults sensitive to, or caused by, particular data patterns in frames transmitted by a service user
b) Describes the protocols and procedures used by Maintenance Points to loopback payload frames while allowing application running as usual.

<<The additions to the Scope clause, as partly illustrated by the above, (will) comprise(s) the following logical parts:

The introductory lines (and possibly the initial list items) reiterate the scope of the PAR. To ensure smooth progression through the final approval stages of the draft, these should remain very little modified from the above or some other way of clearly expressing the PAR Scope text.

The following list items are really a step by step contents list, delineating exactly what is in scope, showing how each clause contributes to meeting the objective of the standard, and how each builds on its predecessors. The current list items are not accurate, and are merely place holders for the first draft items. In early drafts this list serves as a top down view of what is to be said, in later drafts it should accurately reflect what is done in each clause, and thus serve as a way of highlighting gaps.

The next set of list items state what conformance requirements are set out. This becomes important if the amount of necessary context and other supporting material is such that it is easy to miss the actual normative provisions, i.e. what implementation aspects are actually prescribed. It is less important if there is only one obvious conformance item, and should be omitted in that case.

The remaining text is concerned (where necessary) with clarifying the scope of this standard in relation to other complementary standards, some of which may appear on first sight to overlap in scope. These can be broken down into the following categories (a) other standards that can be used by this standard, and which require this standard to make choices about how they are used, while not changing them (b) standards which may make use of this standard, but whose own operation (including option restriction, selection, or definition) is wholly outside the present scope (c) generally standards or industry practices that can use this standard in a way that merits some informative (non-normative) discussion to encourage such use.>>

2. References

Insert the following references at the appropriate points:

<<as needed, not forgetting that references that are not absolutely required for provisions for confoormance should go in the Bibliography and not in the References.>>

DDCFM contribution

Draft Standard for
Local and Metropolitan Area Networks—

Virtual Bridged Local Area Networks — Amendment ?: Management of data driven and data dependent connectivity faults

Sponsor

LAN MAN Standards Committee of the IEEE Computer Society

This is an individual contribution in anticipation of the future approval of a proposed project and has no official standing whatsoever. The intent of this contribution is to provide supporting information on the integration of the proposed project into 802.1Q and its amendments. It has not been prepared by the Interworking Task Group of IEEE 802.1

Abstract: This amendment extends the connectivity fault management capabilities introduced in P802.1ag to include diagnosis and isolation of faults sensitive to, or caused by, particular data patterns in frames transmitted by a service user.

Keywords: local area networks, LANs, metropolitan area networks, MANs, transparent bridging, MAC Bridges, VLANS, OAM, fault management.

3. Definitions

Insert the following definitions, renumbering to place them in the appropriate collating order.

3.1 **<<DDCFM>>:** <<definition>>.

3.2 DDF: Data Dependent and Data Driven Faults

3.3 DDCFM Data Dependent and Data Driven Connectivity Fault Management

3.4 RFM Reflected Frame Message

3.5 SFM Send Frame Message

3.6 FPT Forward Path Testing

3.7 RPT Return Path Testing

4. Abbreviations

Insert the following definitions, placing them in the appropriate collating order(alphabetical).

DDCFM Data driven and Data dependent Connectivity Fault Management

5. Conformance

In subclause 5.3.1 VLAN-aware Bridge options, insert the following additional bullet (re-numbered) after current bullet (c):

a) Support DDCFM operation ()

Insert the following subclause after Clause 5.3.1.3

5.3.1.4 VLAN-aware Bridge DDCFM requirements (optional)

A VLAN-aware Bridge implementation that conforms to the provisions of this standard (40, 19, 20)for the management of connectivity faults that are driven by or dependent upon specific data in the user frames conveyed, shall:

a) Support Forward Path Testing configuration, which include creation and deletion of Reflection points and reflection target points, reflection filter definition, the continuation option, and Forward Path Testing Scheduler.
b) Support intercepting the filtered traffic flow at reflection point, encapsulate OAM headers, forward the encapsulated frames to the Reflection Target, and/or continue forwarding the traffic flow to their original destinations.
c) Support Return Path Testing (RPT) configuration, which include creation and deletion of Return Path Testing Starting Point, and its scheduler.
d) Support decapsulate Return Path Testing’s Send Frame Messages, and forward them to their destinations.

<<The above only addresses Bridge requirements for DDCFM. Other clauses will be required to handle the legal behavior of (classes of) other systems participating in DDCFM exchanges. These might be structured simply as one clause per participant type, or there might be a clause (or two) for DDCFM capable systems in general (including bridges) and these might be referred to by the systems specific clauses.>>

6. Support of the MAC Service

<<This amendment is not expected to make any changes to this clause.>>

7. Principles of network operation

<<This amendment is not expected to make any changes to this clause.>>

8. Principles of bridge operation

<<Any significant additions to this clause are most likely to take the form of (a) new subclause(s) rather than tweaking of existing text. However any architectural description of the entities participating in DDCFM protocol is more likely to be handled in an entirely new clause added to the standard by this project.>>

8.12 Bridge management entity

Insert a sixth point immediately following point e):

f) CFM MIB.

<<It seems more likely (?) that DDCFM will have its own MIB to control operation of the mirror etc. rather than seeking to extend the CFM MIB.>>

12. Bridge management

<<Any significant additions to this clause are most likely to take the form of (a) new subclause(s) rather than tweaking of existing text. However any architectural description of the entities participating in DDCFM protocol is more likely to be handled in an entirely new clause added to the standard by this project.>>

12.12 Bridge management entity

Insert a sixth point immediately following point e):

f) CFM MIB.

<<It seems more likely (?) that DDCFM will have its own MIB to control operation of the mirror etc. rather than seeking to extend the CFM MIB.>>

15. Support of the MAC Service by Provider Bridged Networks

Insert a new clause after the existing 15.x level clauses, renumbering as necessary, as follows.

15.10 DDCFM

Data Dependent and Data Driven Connection Fault Management (DDCFM) provides tools for network operators to detect and isolate data dependent and data driven faults in Virtual Bridged Local Networks under multiple administration domains.

Clause 40 introduces the principle of DDCFM. The DDCFM entities are also added to Clause 19. The DDCFM protocols are added to Clause 20. Together, these 3 clauses provide a means whereby diverse administrations can detect, isolate, and correct data dependent and data driven connectivity faults in the MAC Service with a minimum of access to each others' equipment.

18. Principles of Connectivity Fault Management operation

<<This amendment makes no, or minimal, changes to Clause 18. Instead, this amendment introduces a new clause (40) to cover the basic concepts of DDCFM operation.

The option of extending Clause 18 to cover DDCFM has been considered. However this would have obscured some of the things that the clause currently says. Too often the latest amendments to a standard have the effect of diminishing the clarity of the base standard, as if the subject of the amendment were the most important thing the standard had to say. Tight integration of DDCFM into Clause 18 would have required caveats on the transparency of CFM that might make it harder to see the need for architecting CFM as a shim, while not removing that need.

If DDCFM were to be integrated into Clause 18 the following would be required:

a) Caveats and extensions to the second paragraph of the introductory text.
b) Additions to the dashed list of CFM functions (Path discovery, faultdetection, ...)
c) An introductory paragraph following the above, similar to that for Path discovery etc.

The following DDCFM specific considerations would seem to have no natural home in Clause 18 (even with the addition of subclauses, because they don’t fit within the current purpose of the clause):

d) Detailed discussion of DDCFM specific security issues.

It is not currently obvious that these items would fit naturally within other existing clauses—so a new clause likely to be required in any case.

>>

This amendment makes no changes to Clause 18.

<<Consider a note after the third paragraph to point forward to DDCFM in Clause 40, and state that 40 makes use of the architectural principles in 18.>>

19. CFM Entity operation

19.2.2 MEP Functions

Insert the following list items, renumbered these and subsequent list items as necessary, after the first sublist in clause 19.2.2:

k) May decapsulate and transmit the payload of received Send Frame Messages (SFMs) as defined in Clause 40;

l) May encapsulate selected data frames to be reflected to a specific location as defined in Clause 40. Reflected Frame Messages (RFMs)

m) May implement a Continuation function as defined in Clause 40 to allow the selected data frames to be reflected also to be forwarded to their original destinations;

19.2.3 MEP Architecture

Add the following paragraph after the initial paragraph of clause 19.2.3, and the figure after the existing Figure 19-2:

Figure 19-3 illustrates the addition of the optional Reflection Responder and Decapsulator Responder DDCFM (Clause 40) functions to the MEP shown in <ref existing Figure 19-2>.

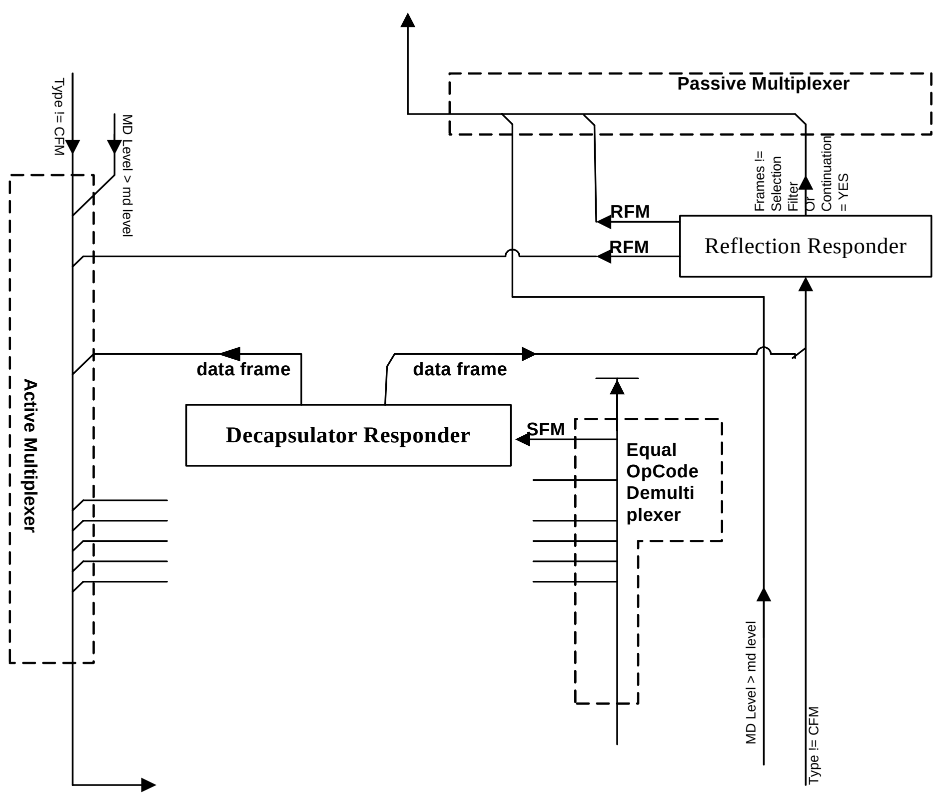


Figure 19-3—MEP with DDCFM functions

<<Specification alternatives:

1) Instead of providing the Reflection functionality within a CFM shim, as shown above, that could be done within the Forwarding Process of the bridge’s MAC Relay Entity, using the Filtering Database (possibly extended) to provide the necessary controls. That approach might be a natural extension of the existing

“mirroring port/VLAN” functionality provided in some existing products. Needs discussion. Given the extent to which implementation models affect Implementers’ views as to what functionality is acceptable, supportable, or even essential, it might be necessary to explore both models for a while

2) The Decapsulator is addressed both by the (encapsulating) MAC address of the SFM and the MD Level (thus preventing inappropriate access from outside the Maintenance Domain), so it could be specified as being attached as an end station style application. Containment of SFMs would happen by default by the existing level filtering performed by CFM shims. On the other hand it is unclear what practical improvement would arise from first checking the MAC Address and then the MD Level, as opposed to the other way round, and whether (once the alternative architecture was spelled out) whether it would look any different to the above. Needs discussion.

Add the following entries to Table 19-1:

Table 19 -1—Actions taken by OpCode Demultiplexers

OpCode	MEP Equal OpCode Demultiplexer	MEP Low OpCode Demultiplexer	MHF OpCode Demultiplexer
SFM	Sets SFMreceived and SFMPDU for <Clause 40>	Discards PDU	<t.b.d>
RFM	Forward as ordinary data frames	Discards PDU	<t.b.d>

Insert new clauses after the existing 19.2.x level clauses, renumbering as necessary, as follows.

19.2.17 DDCFM Send Frame Decapsulator

The Return Path Testing Decapsulator Responder is a CFM shim for decapsulating frames with the DDCFM SendFrameMessage OpCode (SFM), and sending the decapsulated frames towards their destinations.

19.2.18 DDCFM Reflection

The DDCFM Reflection Responder is a CFM shim for selecting frames to be reflected, mirroring the selected frames to be continued if the Continuation option is set, and encapsulating the selected frames with Forward Path Testing DDCFM OpCode. The encapsulated frames are called Reflected Frame Message (RFM).

19.3 MIP Half Function

DDCFM Reflection can be configured at either ingress port or egress port. If configured at ingress port, DDCFM shim behaves as MIP up half function. If configured at egress port, DDCFM shim behaves as MIP down half function.

<<Need some caveats around implementation of Selective Mirror at MIPs, since it requires configuration and hence state, even for trivial mirrors e.g. reflect all traffic back on this service instance (RFM encapsulated of course) while continuing to forward on to destination. Service providers are unlikely to want to put this functionality in MIPs for their customers, though its use in MIPs for themselves is far more likely. The most common usage is likely to be at the customer MD Level.>>

19.3.1 MHF functions

<<Add to the list of MHF functions.>>

19.3.2 MHF Architecture

Figure 19-5 illustrates the addition of the optional Selective Mirror and Send Frame Decapsulator DDCFM (<ref>) functions to the MHF shown in <existing Figure 19-3>.

<<Figure showing additions in the same way as Figure 19-3>>

Figure 19-5—MEP with DDCFM functions

Insert new clauses after the existing 19.2.x level clauses, renumbering as necessary, as follows.

19.3.12 DDCFM Send Frame Decapsulator

The Send Frame Decapsulator is a down MHF function. For SFM frames with the flag indicating to be decapsulated at the output port, the SFM messages are forwarded to the Relay Entity and then to the output port to be decapsulated.

NOTE—MIP and MEP Send Frame Decapsulators are separate options. The capabilities provided can differ.

19.3.13 DDCFM Reflection Responder

The DDCFM Reflection Responder can either up MHF or down MHF depending on where Reflection Point is configured. When the Reflection Point is configured on the input port of the frames to be reflected, the Reflection Responder is serving as an up MHF. When the Reflection Point is configured on the output port of the frames to be reflected, the Reflection Responder is serving as a down MHF.

NOTE—MIP and MEP Reflection Responder are separate options. The capabilities provided can differ.

19.4 Maintenance Point addressing

<<Add a following two rows to the Table 19-2>>

Table 20—Extra entries added to Table 19-2 (Received destination_address filtering by CFM)

Entity	PDU	Received frame destination_address comparison		
		Unicast	CCM Multicast	LMT Multicast
Reflection Responder	RFM	Match	Not used	Not used
Decapsulator Responder	SFM	Match	Not used	Not used

<<Add a following sentence at the end of Clause 19.4:>>

Both DDCFM SFM and RFM messages can only have uni-cast address. Multicast address is not supported by SFM and RFM.

20. Connectivity Fault Management protocols

Change the first sentence of Clause 20 as follows:

Maintenance association End Points (MEPs) and Maintenance association Intermediate Point (MIPs) can participate in ~~three~~ the following CFM Protocols, specified in this clause:

Convert the first three lettered bullets in Clause 20 to a dashed list (so the following lettered list begins with (a)), and add list items for the Send Frame and Mirror protocols, as follows:

— the Continuity Check protocol (20.1);
— the Loopback Protocol (20.2); ~~and~~
— the Linktrace protocol (20.3);
— the DDCFM Send Frame Decapsulator Protocol (40.8)
— the DDCFM Reflection protocols(40.8).

Add the following list items to the lettered list introduced by "This Clause specifies the", appropriately lettered, and immediately prior to the existing bullet (r) "Requirements for decrementing the timer counters ..":

r) Send Frame Decapsulator (40.8.3), variables
s) Reflection (40.8.3), variables

Add the following clauses following the last of the present 20.n numbered clauses:

20.45 Send Frame Decapsulator protocol

<<The Send Frame Decapsulator Protocol is described in detail in Clause 40.8.>>

20.46 Reflection protocol

<<The Reflection Protocol is described in detail in Clause 40.8>>

20.47 Send Frame Decapsulator variables

<< Details are in 40.8.3>>

20.48 Send Frame Decapsulator procedures

<<Details are in 40.7.3>>

20.49 Send Frame Decapsulator state machines

<< Details are in 40.8.2>>

20.50 Reflection and Continuation Encapsulator variables

<< Details are in 40.8.3>>

20.51 Reflection and Continuation Encapsulator procedures

<< Details are in 40.7.2>>

20.52 Reflection and Continuation Encapsulator state machines

<<Details are in 40.8.1>>

21. Encoding of CFM Protocol Data Units

Change the list following the clause heading, and the following NOTE, as follows:

a) The format used to encapsulate or decapsulate a CFM PDU in a frame (21.2);
b) The format of the Common CFM Header, used in all CFM PDUs (21.4);
c) The format used for all Type, Length, Value (TLV) information elements that can be included in CFM PDUs (21.5);
d) The formats of the Continuity Check Message (CCM, 21.6), the Loopback Message and Loopback Reply (LBM and LBR, 21.7), the Linktrace Message (LTM, 21.8), and Linktrace Reply (LTR, 21.9)~~.~~;
e) The formats of the DDCFM Send Frame Message and Reflected Frame Message (SFM and RFM, 21.10).

NOTE—Clause 18 introduces the principles of CFM operation and the network architectural concepts that support it.Clause 19 breaks down the CFM protocol entities into their components. Clause 20 specifies the protocols operated by the components of each MP. The use of CFM within systems and networks is further described in Clause 22.

Insert the following clauses after the last 21.n level clause:

21.10 Send Frame and Reflected Frame Message formats

<<The format of SFM and RFM are described in Clause 40.9>>

22. CFM in systems

<<What needs to be said here?>>

40. Principles of DDCFM operation

Data Dependent and Data Driven Connection Fault Management (DDCFM) comprises capabilities for operators to detect and isolate data dependent and data driven faults in Virtual Bridged Local Networks. This Clause describes the functions of DDCFM and how they can be operated and managed. The DDCFM is an extension to Connection Fault Management defined by Clauses 18 through 22 (802.1ag). As in the case of CFM, DDCFM capabilities can be used in networks operated by multiple independent organizations, each with restricted management access to each other's equipment.

40.1 What Are Data Dependent and Data Driven Faults?

There are two broad types of faults in Bridged Networks that affect only frames or sequence of frames carrying certain data, addresses, or combination of them. Simple data dependent faults are those that result in the repetitive loss of each of those frames, independent of any other frames, and are usually the result of simple misconfiguration or of a failure to appreciate the consequences of a configuration option—installing protocol specific filters, for example. Data driven faults are more complex: the presence (or absence) of some data frames cause or contribute to the loss of other frames. While the services supported by bridged networks are notionally data independent, the use of data driven techniques enables enhanced service delivery. To give three examples: multicast frame filtering and consequent bandwidth saving is facilitated by IGMP snooping; stateful firewalls are used to protect users connected to managed services; and efficient allocation of frames to the individual links of an aggregation (802.3ad Link Aggregation) is often based on spotting conversations by looking at frame data.

40.2 Basic Principle To Detect and Isolate Data Dependant and Data Driven Faults

The major work of detecting data dependent and data driven faults (DDF) is to find out where those faults (DDF) actually occur. Once DDF are isolated to a small enough network segment, such as a bridge port or Maintenance Point, the next step of detecting why or how those data patterns or sequences actually cause the DDF at this location, is much simpler.

The basic procedure to achieve isolation of a DDF is to divide the network into multiple segments and verify if the suspected data frames can traverse through each segment as expected. When a network segment is identified as responsible, the segment is further divided into smaller segments until a bridge, a port, or a CFM Maintenance Point (Clause 20) is identified as responsible for not passing through the service instances or the suspected data frames with expected quality.

DDF may not be apparent in absence of live traffic (that is, when test data are used). Therefore, diagnosis must be carried out while the network is actually running, and the diagnostic tools themselves must not introduce further data dependent faults.

DDCFM is a tool for operators to detect, isolate, and verify data dependent and data driven faults. There are two types of DDF testing: Forward Path Testing (FPT) and Return Path Testing (RPT).

40.2.1 Forward Path Testing

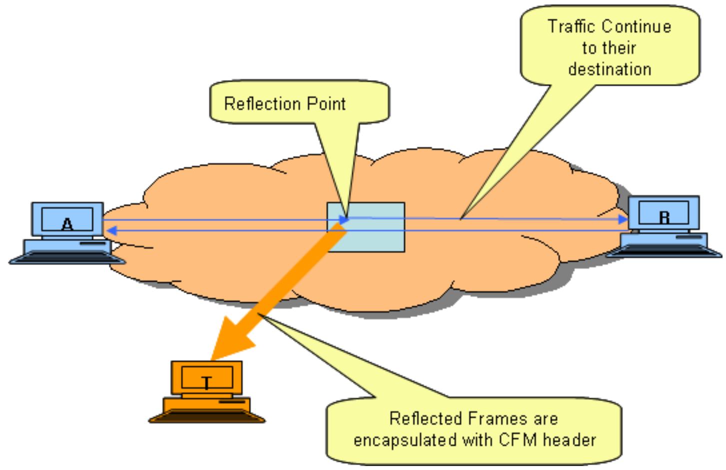


Figure 40-1—Forward Path Test

The goal of Forward Path Testing (FPT) is to determine whether a specified traffic flow (e.g. frames associated with a service instance or selected data frames with same Destination Address, etc.) can reach a particular location (which could be a bridge port, or a Maintenance Point, etc.) without error (like dropping packets, etc.). Forward Path Testing is achieved by reflecting (or turn around) the identified traffic flow (e.g. a service instance or selected data frames) to a specific target location, which could a bridge, a test equipment, or the source node. There are many ways for the target location to verify the reflected data frames. One example is comparing the reflected frames with the original ones to verify if there are any errors. Another example is running a proxy application at the target location to simulate the handshakes as if those packets actually reach their original destinations.

In order for intermediate and/or source bridges to distinguish the reflected flow from other traffic, reflected frames are encapsulated with a CFM header. The data frames to be reflected can also be continued to their original destination(s). This option, which is called Continuation Option throughout this document, is to protect applications which require both Source and Destination nodes having periodic handshakes.

Forward Path Testing (FPT) consists of configuration, action to reflect identified data frames, and analysis of the reflected data frames. The configuration and the action to reflect identified data frames are specified in this standard. But methods of analyzing the reflected data frames is beyond the scope of this standard.

The Forward Path Testing requires configuration of the following information at the Reflection Point:

1) Reflection Point Address, which is a MAC address of a bridge port at which the selected flow is reflected. A bridge usually has multiple ports. Reflection Point can be configured at any one of the ports.
2) Reflection Filter Definition, which is to specify what data frames are selected to be reflected.
3) The Reflection Target Address, which is a MAC address to which the reflected frames are targeted.
4) Option of Continuation, which is to indicate whether or not the reflection point allows the reflected frames to be continued towards the DA specified in the frame header.
5) Forward Path Testing Scheduler which specifies the starting time and ending time of the Forward Path Testing.

40.2.2 Return Path Testing

Return Path Testing (RPT) is to determines whether a flow can be sent without error from a specific point within a network to a station or stations specified by the DA associated with the frames of the Flow-Under-Test. RPT is performed by encapsulating each frame of the Flow-Under-Test with a CFM header at an Originating station. The destination of the encapsulated flow is the Starting Point of Return Path Testing. At the RPT Starting Point, the CFM encapsulation is removed and the frames are sent to the station or stations specified by the DA field in frames of the Flow-Under-Test. This procedure is illustrated in the diagram below.

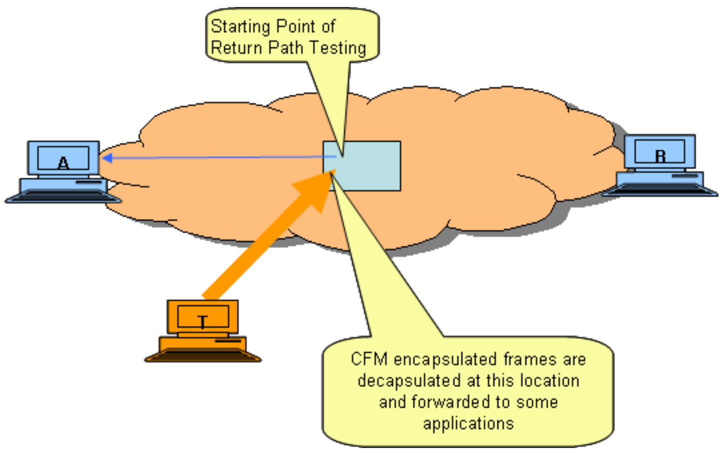


Figure 40-2—Return Path Testing

The “Starting Point of Return Path Testing” can be also called the Return Path Testing Reflection Point. For FPT, raw frames go from Source to Reflection Point. They are CFM-encapsulated by the Reflection Point and reflected towards the Target, where they are decapsulated. For RPT, frames are encapsulated at the Origin and sent to the Reflection Point. The Reflection Point decapsulates the frames and reflects the raw frames towards the station or stations specified by the destination address of the test frames. Different from FPT, RPT does not have the Continuation Option.

<<Note: However, RPT could consider an option of stopping frames coming from the Source Address which are associated with the flow-under-test from forwarding. Need some discussion on the merit of this option>>

The Return Path Testing consists of configuration, action to decapsulate and forward, and final analysis of received data frames at their destination(s). The configuration and the action to decapsulate and forward are specified in this standard. As in the case of FPT, the methods of analyzing the received data frames at destination station(s) is beyond the scope of this standard.

The Return Path Testing requires the configuration of the following at the RPT Starting (Reflection) Point:

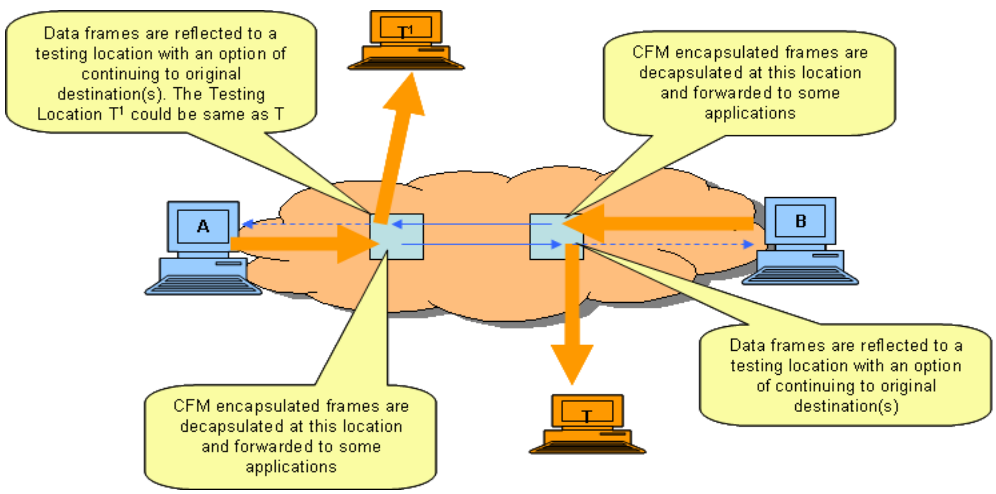
1) RPT Starting (Reflection) Point Address, which is MAC Address of a bridge at which CFM-encapsulated frames are decapsulated and the decapsulated frames are forwarded to the station or stations specified by the DA field of the flow-under-test.
2) RPT scheduler which defines the starting time and ending time of the testing.

The Return Path Testing allows network operator to inject specific traffic flow(s) into network for test purpose.

40.2.3 Derived Testing Scenarios

Forward Path Testing and Return Path Testing can be used together in various ways to achieve more sophisticated testing to detect and isolate data dependent and data driven faults. It is beyond the scope of this standard to elaborate different ways of combining Forward Path Testing and Return Path Testing. This section only illustrates one example of using both Forward Path Testing and Return Path Testing.

When Forward Path Testing and Return Path Testing are used at both end of a network segment of a specific data flow, it can test if the network segment can forward the specified data frames without any error, as depicted in the following figure. This type of testing is especially useful to diagnose data dependent and data driven faults out of fire wall.



40.3 DDCFM and Maintenance Association

A network administrator can only set up Forward Path Testing or Return Path Testing at its own level. A network administrator can set Reflection Point or Return Path Starting Point at DSAP (Domain Service Access Points) and its own ISAP (Intermediate Service Access Points). But A network administrator can't set Reflection Point or Return Path Starting Point at other maintenance domains either higher or lower than its own.

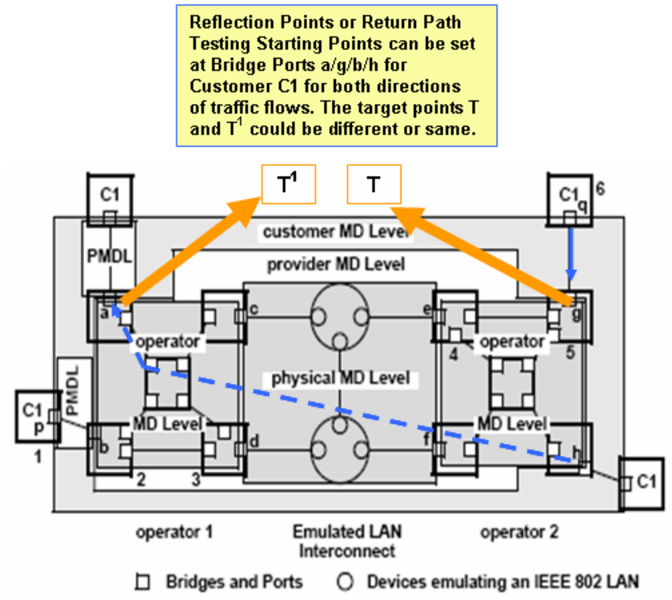


Figure 40-3—DDCFM and Maintenance Domain

There are 4 Maintenance Domains shown in the figure above: a Maintenance Domain for Customer C1, a Maintenance Domain for Operator 1, a Maintenance Domain for Operator 2, and a Maintenance Domain for Provider MD Level. The administrator at each maintenance domain can set up either Forward Path Testing or Return Path Testing at its own domain's DSAP and ISAP for either direction of data flows. But the Administrator for C1 can't set up any DDCFM testing points on the ISAP of Operator 1, Operator 2, or Provider MD.

In the diagram above, operator can set up one Reflection Point at port "g" for traffic from C1/q and another Reflection Point at port "a" for traffic from operator's interior bridges. The operator can set the target of both reflection points as one location, then the reflection target can easily compare the data frames pumped into the Operator2's network with the data frames sent out of Operator1's network to figure out what data frames are dropped by mistake within the operator's networks.

40.4 Forward Path Testing Configuration

40.4.1 Forward Path Testing Reflection Point

The Forward Path Testing Reflection Point is associated with one MA. If a bridge port supports multiple MA Levels, multiple Reflection Points can be established on this bridge port. However, each Reflection Point only reflects service instances within the associated MA.

40.4.2 Forward Path Testing Reflection Filter Definition

Forward Path Testing Reflection Filter is for selecting data frames to be reflected. Reflection Filter has an option of letting the selected frames to be copied and forwarded to Relay Entity for continuing to their original destinations. This option is called "Continuation".

The Continuation option can be set to be either "Yes" or "No". When "Yes" is chosen for Continuation option, the selected data frames are branched to two directions: one direction to the "Reflection Responder", and another direction to the normal Relay Entity. When “No” is chosen for the Continuation option, the selected frames will only be forwarded to the “Reflection Responder” function.

The following selection algorithms shall be supported by all Bridges for reflection filter:

a) Reflect All: all frames within the associated MA level are selected.
b) Service Instance based selection: all data frames belonging to the specified service instances are selected.
c) Destination address based selection: all data frames with the specified Destination Address are selected.
d) Source address based selection: all data frames with the specified Source Address are selected.

Manufactures may define more reflection selection algorithms for their equipment.

40.4.3 Forward Path Testing Reflection Target (or Receiver)

Reflection Target is an address to which the DDCFM encapsulated reflected frames are forwarded. The Reflection Target could be a MEP, a MIP, a bridge or a test equipment which doesn’t participate in CFM protocols.

40.5 DDCFM Scheduler Configuration

Both Forward Path Testing and Return Path Testing needs a scheduler to be configured to mark the Starting and Ending Time of the testing. Following options of DDCFM scheduler shall be supported by all bridges:

a) Starting Time and Duration of Testing. For example, the starting time is Day-Hour-Minute, and duration is 3 hours.
b) Starting Time and Ending Timing. For example, the starting time is Day-Hour-Minute and the ending time is Day-Hour-Minute

Both Starting Time and Ending Time can be specified as absolute time, such as Day-Hour-Minute, or relative time, such as xxx Hours or/and Days from the time when the scheduler is configured.

40.6 Return Path Testing Configuration

The configuration for Return Path Testing configuration consists of:

a) Return Path Testing Decapsulating Point Address definition, which is a MIP/MEP within the same Maintenance Domain.
b) Return Path Testing scheduler, which is defined in the section above.

40.7 DDCFM Entity Operation

This sub-clause specifies:

a) How the Maintenance Points (MP) with DDCFM enabled attach to ISS-SAPs or EISS-SAPs.
b) Forward Path Testing Responders
c) Return Path Testing Responders

40.7.1 DDCFM Maintenance Points

The Data Dependent and Data Driven Connectivity Fault Management (DDCFM) is another type of Connectivity Fault Management Shim. DDCFM can be on either MEP or MHF. Similar to CFM, an Active SAP process DDCFM PDUs, whereas a Passive SAP pass through unprocessed DDCFM PDUs.

Different from CFM, a Maintenance Association configuration doesn't automatically instantiate a DDCFM shim. DDCFM shim has to be created after a CFM Maintenance Association is created. DDCFM shim is specific to a particular CFM Maintenance Association. When a bridge port is configured with multiple Maintenance Associations, multiple DDCFM shims can be created with each DDCFM shim mapped to one Maintenance Association.

40.7.2 Forward Path Testing Reflection Responder

The Forward Path Testing Reflection Responder is a CFM shim for selecting frames to be reflected, mirroring the selected frames to be continued if the Continuation option is set, and encapsulating the selected frames with Forward Path Testing DDCFM OpCode. The encapsulated frames are called Reflected Frame Message (RFM). The following figure describes the detailed functions of Reflection Responder.

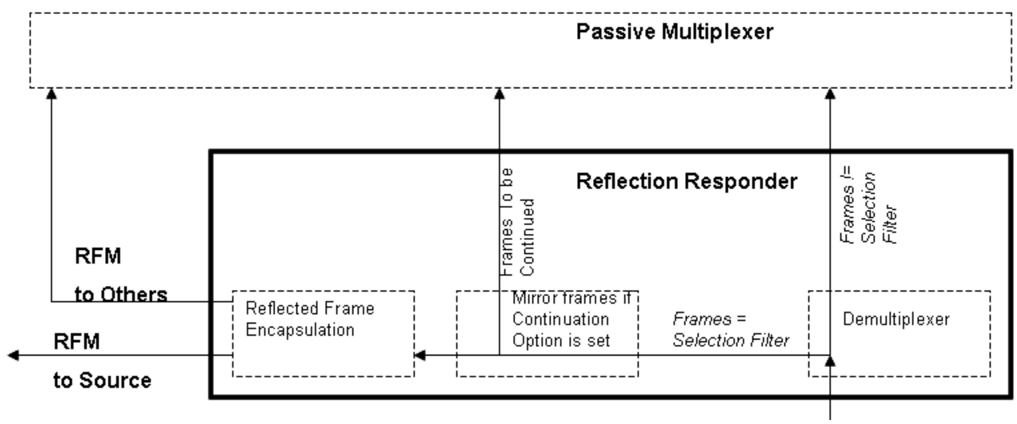


Figure 40-4—Detailed Functions of Reflection Responder

The Demultiplexer function within the Reflection Responder is for filtering frames which match the Reflection Filtering.

The Mirror function within the Reflection Responder is for branching the frames to two flows, one towards Relay Entity, another one towards Reflected Frame Encapsulation function.

The Encapsulation function within the Reflection Responder is for encapsulating the selected frames with Forward Path Test OpCode. When the Reflection Target Address is set different than the source address of the reflected frames, the encapsulated Reflected Frame Messages (RFM) have to be forwarded to the Relay Entity to be forwarded to appropriate output ports. When the Reflection Target Address is same as the source address of the reflected frames, the encapsulated Reflected Frame Messages (RFM) have to be sent to the down MHF Active Multiplexer function.

40.7.3 Return Path Decapsulator Responder

The Return Path Testing Decapsulator Responder is a CFM shim for decapsulating frames with the DDCFM SendFrameMessage OpCode (SFM), and sending the decapsulated frames towards the bridge port. The data frames towards the bridge port are passed to CFM's Active Multiplexer first.

There is a Flag in the SendFrameMessage OpCode to determine if frames are to be decapsulated at the receiving port or output port. If it is at the receiving port, the frames are decapsulated before the Relay Entity/Forwarding processing and the decapsulated frames are sent back to the receiving port via CFM's Active Multiplexer. If the decapsulating is to be done at the output port, the SendFrameMessage encoded frames are forwarded to Relay Entity/Forwarding first, and decapsulated at the output port.

The Forward Path Testing Reflection Responder doesn't have to be configured together with the Return Path Decapsulator Responder. If Forward Path Testing Reflection Responder is not present, the SendFrameMessage encoded frames towards the Relay Entity are actually passed to CFM's Passive Multiplexer first. If Forward Path Testing reflection Responder is present, then the frames towards Relay Entity are passed to the reflection Responder first.

The following figure describes the detailed functions of Decapsulator Responder.

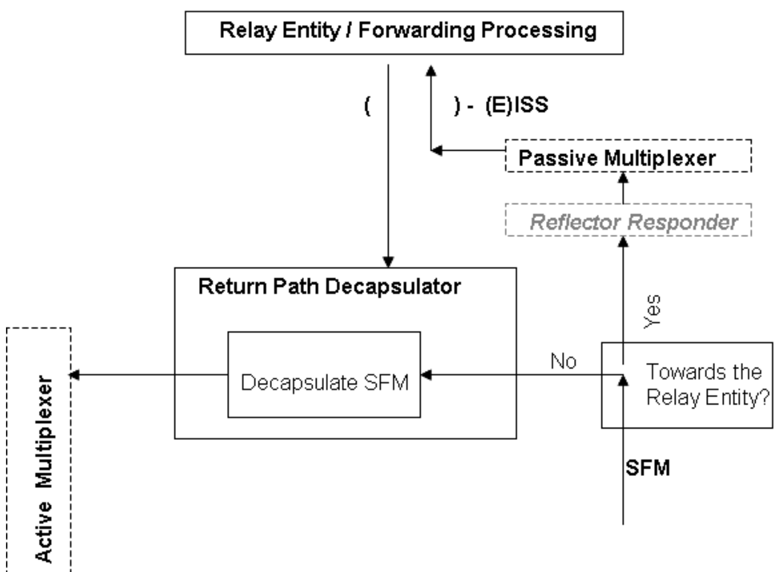


Figure 40-5—Return Path Decapsulator Responder

40.7.4 Forwarding of DDCFM Encapsulated Frames

All bridges shall relay DDCFM encapsulated frames (or messages), such as RFM (Reflected Frame Message) and SFM (Send Frame Message), in the same way as ordinary data frames.

40.8 DDCFM Protocols

This sub-clause specifies the DDCFM protocols in terms of a number of state machines, and the variables and procedures associated with each machine.

40.8.1 Forward Path Testing State Machine

Once Forward Path Testing is configured by network administrator, the actual testing is triggered by scheduler and performed by Reflection Responder associated with a bridge and Reflection Target which conducts analysis of reflected data frames.

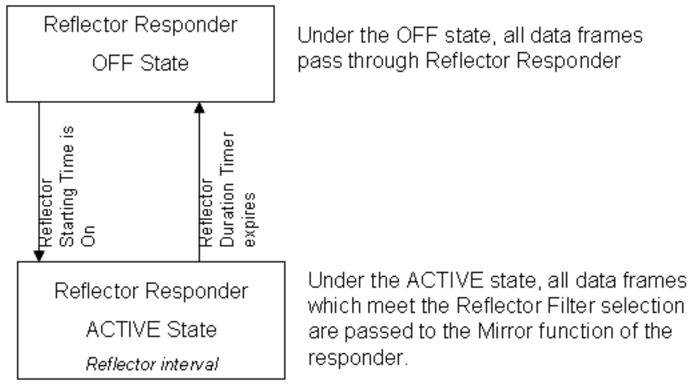


Figure 40-6—Forward Path Testing State Machine

Reflection Interval is the duration of the scheduled Reflection Responder.

40.8.2 Return Path Testing State Machine

Once the Return Path Testing is configured by network administrator, the actual testing is triggered by the scheduler and performed by the Decapsulator Responder. Depending on the SendFrameMessage OpCode, frames can be either decapsulated at the receiving port or the output port.

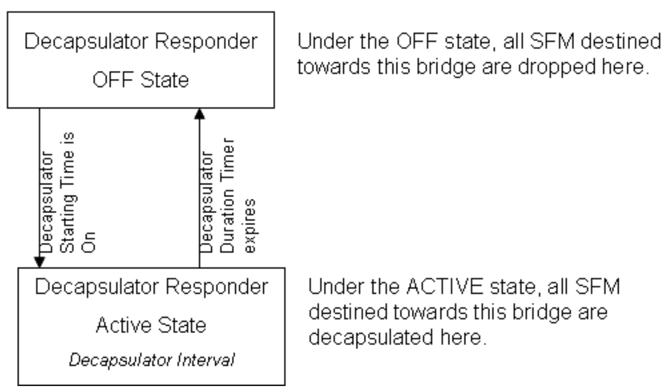


Figure 40-7—Return Path Testing State Machine

Decapsulator Interval is the duration of the scheduled Return Path Testing.

40.8.3 DDCFM Variables

a) ReflectionActive: When set to “TRUE”, the Reflection Responder actively selects frames to be reflected. When set to “False”, the Reflection Responder just pass all data frames towards Relay Entity.

b) DecapsulatorActive: When set to “TRUE”, the Decapsulator Responder is actively decapsulate SFM OpCode frames destined to the bridge. When set to “False”, the Decapsulator Responder just drop all SFM OpCode frames destined towards this bridge.
c) ReflectionStartingTime: This variable is the starting time when the ReflectionActive is set to TRUE.
d) ReflectionInterval: This variable is the duration of ReflectionActive being set to TRUE.
e) DecapsulatorStartingTime: This variable is the starting time when DecapsulatorActive is set to TRUE.
f) DecapsulatorInterval: This variable is the duration of DecapsulatorActive being set to TRUE.

40.9 Encoding of DDCFM Protocol Data Units

This sub-clause specifies the method of encoding DDCFM PDUs. The specifications include:

a) The format used to encapsulate reflected frames
b) The format of SendFrameMessage DDCFM header

40.9.1 DDCFM Header

The DDCFM Header should be same as Common CFM Header defined by Table 21.3, with extra OpCodes defined for ReflectedFrameMessage and SendFrameMessage. With those two messages added, two extra rows are added to the Table 21.4:

Table 40-1—OpCode Field Range Assignment with DDCFM Added

CFM PDU or Organization	OpCode Range
Reflected Frame Message (RFM)	6
Send Frame Message (SFM)	7

40.9.2 Reflection Protocol Data Unit

Reflection Protocol Data Units (PDUs) are identified by the Type value whose format is described by Table 21-1 and Table 21-2.

40.9.2.1 Reflection Target Address Parameter

Reflection Target Address can be one of the following:

a) A single specific MP, or an address of a specific bridge port in the network.
b) Source Address of the frames to be reflected. For a flow of frames to be reflected, there could be different Source Addresses among the frames.

40.9.2.2 Reflected Frame Message Format

Table 41—Reflected Frame Message Format

	0		1	2	3
1	MDLevel	Version	OpCode	Flags	FirstTLVOffset
5	Sequence Number				
9	Reflection Responder Identifier				
13					
17+	Original Data Frame				
last	End TLV (0)				

40.9.3 Send Frame Message PDU

40.9.3.1 Send Frame Message Address Parameter

The Destination Address of SendFrameMessage is a single specific MP.

The Source Address of the SendFrameMessage is a bridge where the SendFrameMessage is initiated. It is not necessary for SendFrameMessage originator to participate in CFM protocols.

40.9.3.2 Send Message Frame Format

Table 42—Send Message Format

	0		1	2	3
1	MDLevel	Version	OpCode	Flags	FirstTLVOffset
5	Sequence Number				
9	Egress Identifier				
13					
17+	Data Frame to be sent				
last	End TLV (0)				

The Flag will identify if the SendFrameMessage is to be decapsulated before or after the Relay Entity.

40.10 DDCFM in Systems

Annex A

(normative)

PICS Proforma[1]

A.5 Major capabilities *(continued)*

Insert the following items at the end of A.5:

Item	Feature	Status	References	Support
DDCFM	Is management of data driven and data dependent connectivity faults implemented?	O	<5.3(x)>, <x.x, y.y>, <A.X>	Yes[ ] No [ ]

A.14 IBridge Management

Insert the following items at the end of A.14:

Item	Feature	Status	References	Support
DDCFM-1	Does the Bridge provide control all of the required managed objects?	CFM:M	<12.x>	Yes[ ]
DDCFM-2		M	<x.x>	Yes[ ]
DDCFM-3		M	<x.x>	No [ ]

Add the following as the last clause of Annex A, renumbering from A.24 as appropriate.

A.24 DDCFM

Item	Feature	Status	References	Support
DDCFM-4		M	<x.x>	Yes[ ]
DDCFM-5		M	<x.x>	Yes[ ]
DDCFM-6		O	<x.x>	Yes[ ] No [ ]

PREDICATES:

<placeholder for the editor’s convenience, no predicates defined at present>

[1] *Copyright release for PICS proformas:* Users of this standard may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.

Annex B

(informative)

Bibliography

Add the following references to the Bibliography, renumbering appropriately.

[B1] <<As needed>>

Annex C

(informative)

DDCFM protocol design and use

<<Editor’s note: This is a permanently retained annex (Z disappears at sponsor ballot time) to describerationale etc. beyond the description appropriate for the main body of the standard.>>

C.1 Introduction

<<Stuff.>>

Annex Z

(informative)

Commentary

<<Editor's Note: This is a temporary Annex, a place to record technical issues and their disposition. This annex will be removed prior to Sponsor Ballot, and preserved on the 802.1 web site for future reference[1].>>

<<The order of discussion of issues is intended to help the reader understand first what is the draft, secondly what may be added, and thirdly what has been considered but will not be included. In pursuit of this goal, issues where the proposed disposition is “no change” will be moved to the end. The description of issues is updated to reflect our current understanding[2] of the problem and its solution: where it has been considered useful to retain the original comment, in whole or part, either to ensure that its author does not feel that it has not been sufficiently argued or the editor suspects there may be further aspects to the issue, that has been done as a footnote.>>

[1]The footnotes in this annex provide further background to its development. Most of the highly subjective material, who said what and were they were right etc. together with temporary notes on blind alleys will be put into the footnotes so that they can be easily stripped out when the final annex is preserved.

[2]This annex is not intended therefore to be a complete historical record of the development of the draft. The formal record is largely captured in the Disposition of Comments on each ballot.