

TRILL Working Group
INTERNET-DRAFT
Intended status: Proposed Standard
Updates: [6325](#)

Donald Eastlake
Huawei
Manoj Wadekar
QLogic
Anoop Ghanwani
Dell
Puneet Agarwal
Broadcom
Tal Mizrahi
Marvell
January 4, 2013

Expires: July 3, 2013

TRILL: Support of IEEE 802.1 Congestion Notification
<[draft-eastlake-trill-cn-00.txt](#)>

Abstract

This document briefly explains the IEEE 802.1 Congestion Notification standard and specifies the support of this standard in TRILL switches (devices that implement the IETF TRILL protocol standard). It updates [RFC 6325](#).

Status of This Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Distribution of this document is unlimited. Comments should be sent to the authors.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>. The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

INTERNET-DRAFT

TRILL: CN Support

Table of Contents

1. Introduction.....	3
1.1 Overview of These Standards.....	3
1.2 Terminology.....	4
2. Congestion Notification.....	5
2.1 Congestion Notification Domains.....	7
2.2 Congestion Notification Tag Details.....	9
2.3 Congestion Notification Message Details.....	9
3 Addition to TRILL to Support Congestion Notification....	11
3.1 TRILL Switch Ingress Details.....	12
3.2 Transit TRILL Switch Details.....	15
3.2.1 Transit TRILL Switch Input Port.....	15
3.2.2 Transit TRILL Switch Output Port.....	15
3.3 TRILL Switch Egress Details.....	16
4. Management Considerations.....	17
5. IANA Considerations.....	17
6. Security Considerations.....	17
7. References.....	18
7.1 Normative References.....	18
7.2 Informative References.....	18
Authors' Addresses.....	20

INTERNET-DRAFT

TRILL: CN Support

1. Introduction

IEEE 802.1 has developed various standards as part of its Data Center Bridging (DCB) activity. The intent of thee of these standards is (1) to efficiently minimize data loss due to queue overflow for selected classes of traffic within Local Area Networks (LANs) meeting certain conditions and (2) to provide limited means to allocate the available bandwidth to different classes of traffic. Those three standards are Priority Based Flow Control (IEEE [[802.1Qbb](#)]), Enhanced Tramission Selection (IEEE [[802.1Qaz](#)]), and the Congestion Notification (CN) feature in the IEEE [[802.1Q](#)] standard. Intended uses include the support of loss sensitive services, such as Fiber Channel over Ethernet [[FCoE](#)], in data centers.

PFC and ETS and their support in TRILL, which requires no changes to TRILL, are described in [[PfcEts](#)].

This document concerns the Congestion Notification (CN) feature in the IEEE [[802.1Q](#)] standard. To support CN, a change to TRILL is required as specified herein and for that reason this document updates [[RFC6325](#)].

The existing optional PAUSE feature of IEEE 802.3 (Annex 31B of [[802.3](#)]) can, with appropriate engineering, also provide Ethernet service without loss of frames due to queue overflow. However, PAUSE has problems as described in [[PfcEts](#)].

1.1 Overview of These Standards

An overview of the CN standard covered herein is given below.

Congestion Notification (CN) provides signaling of congestion on a per flow basis to the end station source of the flow as specified in [\[802.1Q\]](#). As a part of CN, participating end stations are required to implement per flow rate limiting. CN is enabled on a per priority basis and, with appropriate engineering, minimizes frame drops due to queue overflow in a LAN Congestion Notification Domain within which all switches and end stations implement it. CN and Priority-based Flow Control (PFC, see [\[PfcEts\]](#)) complement each other to help eliminate such frame drops. CN reduces congestion by proactively reducing frame ingress rates at the source end station(s) involved in the congestion. For some congestion cases this may be insufficient to stop buffer overflow at a congestion point. PFC provides an emergency brake for such cases and avoids frame loss. Frames that require drops due to congestion for flow control, such as those using TCP [\[RFC793\]](#) can be assigned a priority for which CN is not enabled. CN is not normally used to limit priorities 6 and 7 to avoid interfering with time sensitive control frames that commonly use such priorities. And

CN acts by restraining end station flow sources rather than blocking transmission on intermediate switch ports, which avoids congestion spreading as discussed in [\[PfcEts\]](#). [Section 2](#) below provides additional information on CN and specifies additions to the TRILL protocol to support it.

The PFC, ETS, and CN standards may be implemented independently or in any combination except that implementation of any of them implies implementation of DCBX, specified in IEEE [\[802.1Qaz\]](#) and discussed in [\[PfcEts\]](#).

[1.2](#) Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [\[RFC2119\]](#).

The following acronyms are used in this document in addition to those defined in [\[RFC6325\]](#).

CN - Congestion Notification [\[802.1Q\]](#)

CNM - Congestion Notification Message

CNtag - Congestion Notification tag

DCB - Data Center Bridging [[802.1Qaz](#)]

DCBX - DCB Exchange protocol [[802.1Qaz](#)]

ETS - Enhanced Transmission Selection [[802.1Qaz](#)] [[PfcEts](#)]

FCoE - Fiber Channel over Ethernet [[FCoE](#)]

LLDP - Link Layer Discovery Protocol [802.1AB]

PFC - Priority-based Flow Control [[802.1Qbb](#)] [[802.3bd](#)] [[PfcEts](#)]

RBridge - "Routing Bridge", an alternative name for a TRILL switch

TRILL Switch - A device implementing the TRILL protocol [[RFC6325](#)]

[2.](#) Congestion Notification

Congestion Notification (CN) can limit flows to minimize frame loss by having congestion points that detect congestion send Congestion Notification Messages (CNMs) back to reaction points in end stations that can limit flows. See [[802.1Q](#)] for the specification of the CN algorithms to perform at congestion and reaction points. Congestion Notification is designed to operate best in minimizing frame loss of unicast flows in a LAN composed of point-to-point physical links where all switches have implemented Congestion Notification.

A TRILL switch that implements Congestion Notification may act as an end point, for example when sourcing or sinking SNMP management frames, and thus may contain one or more reaction points, as well as implementing congestion points at its output queues.

Reaction points are in end stations where flows originate and are the mechanism to limit flows. The granularity of reaction point flows is beyond the scope of CN and this document but cannot be larger than a priority at a source MAC address. If the granularity is smaller and there are multiple reaction points in an end station for a given priority, then the end station must label outgoing frames with a Congestion Notification tag (CNtag) that includes an end station flow ID. This flow ID is an opaque field to the rest of the network.

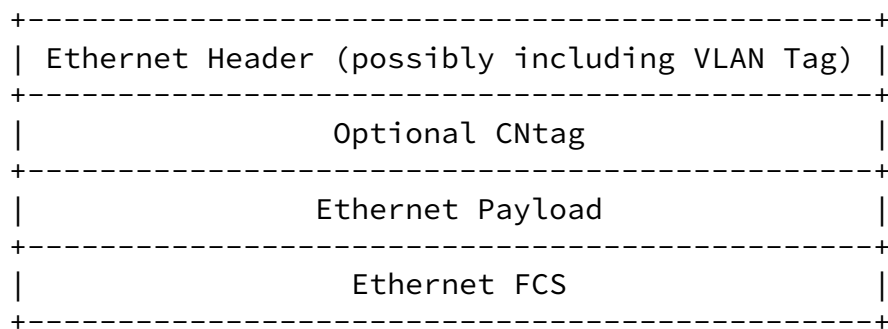


Figure 1: Native Ethernet Frame in a CN Domain

Congestion points are at queues in forwarding devices, normally port output queues. The functions of a congestion point are (1) to conditionally send Congestion Notification Messages (CNMs) to the source of a frame and (2) to conditionally strip Congestion Notification tags (CNtags) out of a frame being forwarded, for example if it is being forwarded out of a congestion notification domain.

When a frame is to be inserted into an output queue with a congestion point, the procedures specified in IEEE [[802.1Q](#)] are used to determine if a CNM should be sent to the frame's source and if so to determine various fields in that CNM. When a frame is to be inserted

into an output queue with a congestion point, the congestion point may remove any CNtag in the frame as discussed in [Section 2.1](#). Congestion points are implemented within the logic associated with a port and require no changes to TRILL for the output of native frames, as TRILL is implemented above such port logic as described in [[RFC6325](#)]; however, when outputting a TRILL Data frame, any CNM generated needs to be for the TRILL encapsulated frame rather than

for the entire TRILL Data frame. In that case there are some differences between the details of the creation of a CNM at an TRILL switch output port and at a bridge output port. This CNM also needs to be TRILL encapsulated but this will happen automatically as the CNM is specified by [802.1Q] to be treated as a native frame arriving at the port.

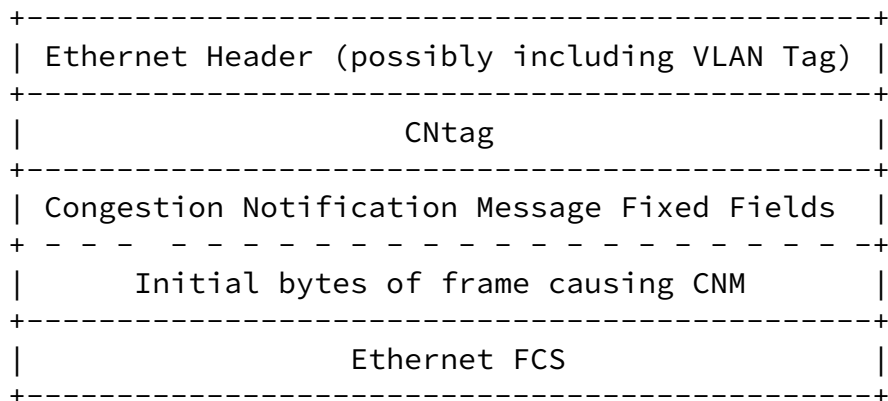


Figure 2: Native Congestion Notification Message

Within a contiguous part of the TRILL campus where Congestion Notification is enabled (see [Section 2.1](#)), you would see the same frames with the same tags as in a similar bridged LAN except that those frames will be TRILL encapsulated as shown in Figures 3 and 4. The exception is when a TRILL-ignorant bridge within the campus produces a CNM in response to a TRILL data frame as shown in Figure 6. The resulting CNM is corrected by the first TRILL switch it encounters, which will be the previous-hop TRILL switch.

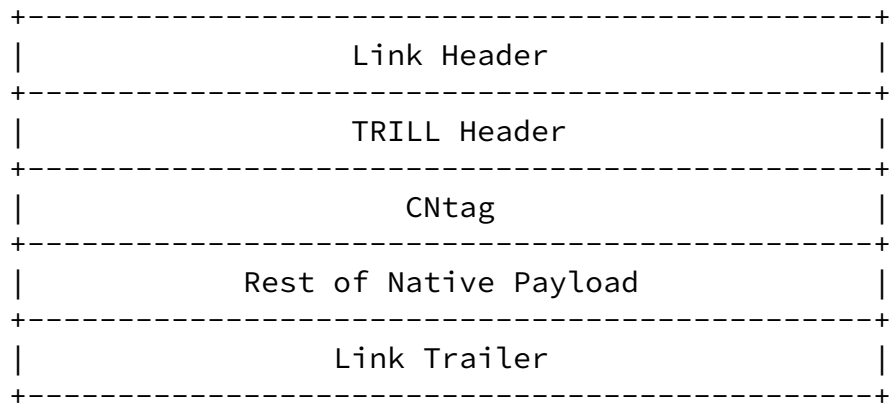


Figure 3. TRILL Data Form of CNtagged Native Frame

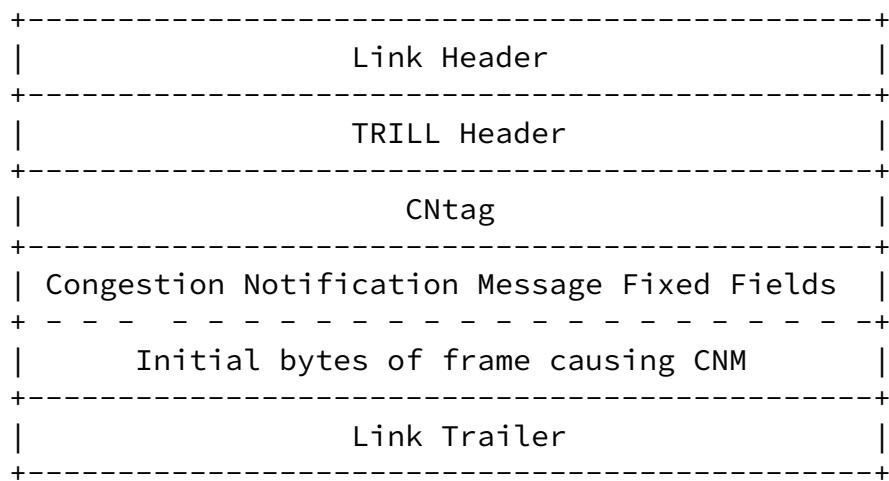


Figure 4: TRILL Data Form of Congestion Notification Message

[2.1](#) Congestion Notification Domains

Congestion Notification (CN) reduces frame drops due to output queue overflow in a Congestion Notification Domain. There could be many such domains, each specified for a particular priority and contiguous set of network stations (end stations, TRILL switches, or bridges), within a TRILL campus. For example, two Congestion Notification Domains, one at priority X and one at priority Y, could cover the same set of contiguous stations, overlapping but different sets of such stations, or completely disjoint sets of such stations, in a campus.

CN includes mechanisms to "defend" Congestion Notification Domains, that is, make sure only congestion managed flows of frames enter congestion point queues. The edge of a domain, i.e. the set of station ports in the domain directly connected to a station not in

INTERNET-DRAFT

TRILL: CN Support

the domain, is determined by a combination of auto-detection using LLDP (see Section 4 of [[PfcEts](#)]) and management configuration. Bridges that implement Congestion Notification defend a domain by the following:

1. Prohibiting priority mapping inside the domain.
2. Mapping the priority of any frame entering the domain from a station outside the domain to a priority that is not a congestion managed priority.
3. Prohibiting the mapping of the priority of any frame entering the domain from a station outside the domain to the domain's priority.

The station containing the reaction-point-equipped source of a flow must be part of a Congestion Notification Domain at the flow's priority along with all stations along the path to the flow's destination and all of the queues involved with the flow must be congestion-point-equipped in order for CN to be able to meet its goals.

Because of item 2 in the list above, a station can be a member of no more than 7 different Congestion Notification Domains because there must be at least one priority that is not congestion managed for use as the mapped priority of entering frames from outside the domain and which are therefore not part of a congestion managed flow. As a practical matter, it is unlikely that a station would be a member of more than 4 or 5 different Congestion Notification Domains as priorities 6 and 7 are normally used for high priority control frames that are not congestion controlled and at least one low priority is kept non-congestion managed for mapping as above.

The per port per priority state of a switch or end station will be one of the following four values, which have the effects indicated:

- o Disabled:
 - On native frame input, frame priority can be mapped to or from this priority.
 - If this is an end-station output port, CNtags are not added.
 - If this is a switch output port, CNtags are not stripped.
- o Edge:

- On native frame input, a frame with this priority is mapped to a non-CN priority and no native frame can be mapped to this priority, regardless of the priority-mapping table at the port. For TRILL Data frames, this also applies to the Inner.VLAN priority.
- If this is an end-station output port, CNtags are not added.
- If this is a switch output port, CNtags are stripped including any CNtag in the encapsulated frame if a TRILL Data frame is

being output.

- o Interior:
 - On frame input, a frame in this priority is not mapped to another priority and no frame can be mapped to this priority, regardless of the priority-mapping table at this port. For TRILL Data frames, this also applies to the Inner.VLAN priority.
 - If this is an end-station output port, CNtags are not added.
 - If this is a switch output port, CNtags are stripped including any CNtag in the encapsulated frame if a TRILL Data frame is being output.
- o InteriorReady:
 - On frame input, a frame in this priority is not mapped to another priority and no frame can be mapped to this priority, regardless of the priority-mapping table at this port. For TRILL Data frames, this also applies to the Inner.VLAN priority.
 - If this is an end-station output port, CNtags may be added.
 - If this is a switch output port, CNtags are not stripped.

Note that when the priority of a TRILL encapsulated frame is mapped, the priority field in the Inner.VLAN tag MUST be changed.

[2.2](#) Congestion Notification Tag Details

An end station originating a native frame may add a Congestion Notification tag (CNtag) to identify the native frame's reaction point in that end station, if the end station and the next hop device are part of a Congestion Notification Domain. A CNtag is 4 bytes long, consisting of a 2 bytes Ethertype (0x22E9) followed by a 2

bytes flow ID, and appears after any VLAN tag but before the frame body. This CNtag flow ID is an opaque quantity only meaningful to the originating end station. The inclusion of a CNtag is optional as the originating end station may be able to identify the corresponding reaction point from other information returned in a Congestion Notification Message such as the priority.

As described in [Section 2.3](#), CNtags are always added to Congestion Notification Messages (CNM) when they are created.

[2.3](#) Congestion Notification Message Details

A Congestion Notification Message (CNM) is, under certain circumstances, created by a congestion point, as described in IEEE

D. Eastlake, et al

[Page 9]

INTERNET-DRAFT

TRILL: CN Support

[[802.1Q](#)], when a frame is entered into the queue associated with that congestion point. The CNM frame always includes a Congestion Notification tag (CNtag, see [Section 2.2](#)). The CNtag includes a zero flow ID if the frame provoking the CNM did not have a CNtag. The body of the CNM itself, after the CNtag, starts with the CNM Ethertype (0x22E7) followed by the information below:

- CNM version information, currently zero
- Quantized congestion feedback information as specified in [[802.1Q](#)]
- An 8 byte opaque ID of the congestion point generating the CNM
- The priority of the frame causing the CNM
- The destination MAC address of the frame causing the CNM
- The number of bytes included from the beginning of the body of the frame causing the CNM
- The first up to 64 bytes of the body of the frame causing the CNM

Except that input bytes/frame counters are not incremented, a CNM generated at an output queue for a port is treated as if it had been received on that port. CNMs are considered to be in the same VLAN as the frame that provoked them and have configurable priority that defaults to priority 6.

It is undesirable, but not an error, for a CNM to be sent in response to a CNM frame which encounters congestion. This is normally avoided

by sending CNM frames with a priority which does not have congestion notification enabled.

As described in [Section 3.1.3](#) below, when a CNM is generated by an TRILL switch when queuing a TRILL data frame, it is generated for the enclosed frame, not for the entire TRILL data frame. This will cause the CNM to be addressed to the source end station of the data.

[3](#) Addition to TRILL to Support Congestion Notification

The figure below is used in the discussion in this section. The assumption is that a frame is generated at End Station "a" (ESa) destined for End Station "b" (ESb) and this frame is forwarded through the sequence of 802.1 bridges (Bn) and TRILL switches (RBridges, RBn) shown. For native frames from ESa, RB1 acts as the ingress TRILL switch, encapsulating and directing them to egress TRILL switch RB3 for decapsulation and delivery to ESb. The arrows indicate the flow of a data frame. Any resulting CNM would flow in the opposite direction; however, such a CNM would be independently routed towards ESa and would not be constrained to follow the same sequence of switches shown below.



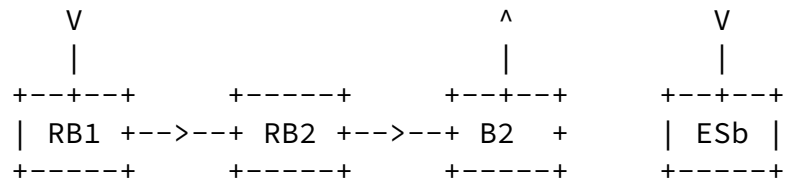


Figure 5: Example Frame Path

TRILL can make no change to the actions at any reaction points in ESa or any congestion points at the output queues of B1, B2, or B3, since they are not TRILL switches. Any CNM generated at B2 will be in response to a TRILL frame and will need to be corrected by the previous hop TRILL switch. The situation at the output queue of RB3 is actually the same as B3 since, as egress, RB3 will have decapsulated any traffic for ESb before it tries to insert it in an output queue. Thus the frame RB3 is enqueueing will be a native frame, a congestion point at the RB3 output can act, for such a frame, exactly as an IEEE 802.1 congestion point, and any CNM generated in the RB3 output from that native frame will be treated as if it was received by the RB3 port.

A CNM created at the RB1 or RB2 output queue is straightforward. Assume the CNM is created in response to TRILL Data frame 1 (TDF1) and the TDF1 encapsulates native frame 1 (NF1). The CNM would be created as a TRILL encapsulated CNM with the ingress TRILL switch of NF1 as its egress. The Inner.MacDA would be ESa. The Inner.MacSA would be the MAC address of the port on which the TRILL encapsulated CNM was initially sent, that is, the same as the Outer.MacSA. The encapsulated CNM itself would be filled in as if in response to NF1, not TDF1.

Similarly, a CNM created at B3 would have ESa as its destination address and would be TRILL encapsulated when it arrived at RB3 as RB3 would be its ingress TRILL switch.

[3.1](#) TRILL Switch Ingress Details

This section specifies special actions for CN at a TRILL switch input port receiving a native frame, that is, the TRILL switch ingress

function. The usual processing on the priority of the input TRILL data frame, modified as described in [Section 2.1](#), is done. Special actions are required only when the native frame received is a CNM.

The ingress process at a TRILL switch, say RB2, supporting CN MUST detect the case of a native CNM created by a bridge in response to a TRILL frame, say by B2 in Figure 4, and transform or discard it as described below. No other changes are needed in the TRILL switch ingress process. If such a CNM was generated in response to a TRILL control (IS-IS) frame, it is discarded.

A native CNM requiring special actions is easily recognized on ingress as it's MAC destination address will be the TRILL switch and it will have the CNM Ethertype. (A CNM not addressed to the TRILL switch must have been generated in response to an unencapsulated native frame, for example at B3 in the diagram above, and can be encapsulated by its Ingress TRILL switch and generally forwarded by transit TRILL switches in the same way as other native frame.)

Such a native CNM resulting from a TRILL data frame at B2 has the contents generally shown in Figure 6 and listed further below.

```
+-----+
| Ethernet Header (possibly including VLAN Tag) |
```

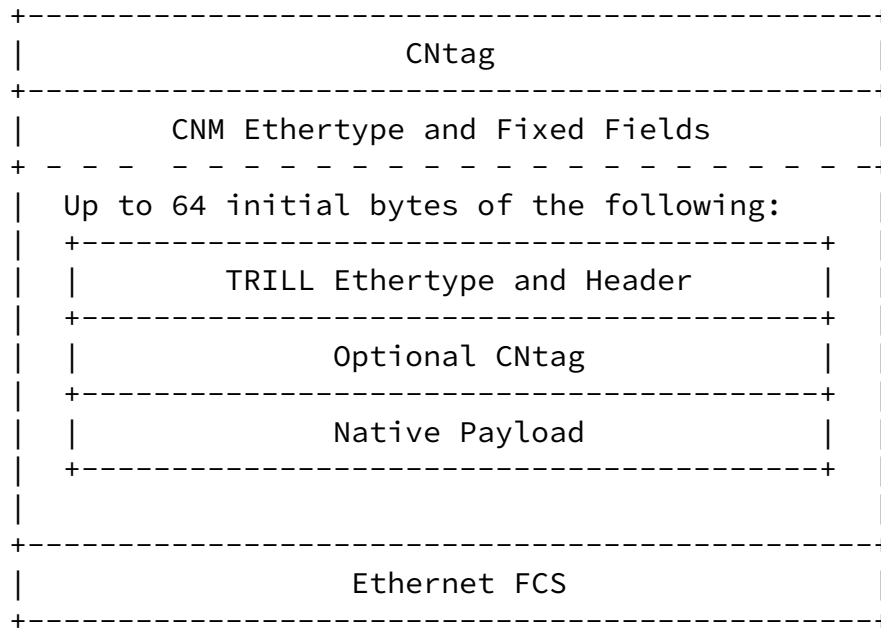


Figure 6: Native CNM Caused by a TRILL Data Frame

- 1 + Outer.MacDA, MAC address of RB2
- 2 + Outer.MacSA, MAC address of port on which B2, the bridge generating this CNM, sent the CNM
- 3 + Outer.VLAN tag for the designated VLAN on the RB2 to RB3 link with the priority configured at B2 for CNMs (default priority 6)
- 4 + CNtag (CNtag Ethertype 0x22E9 followed by Flow ID of zero)
- + CNM
 - 5 o CNM Ethertype 0x22E7
 - 6 o CNM version information, quantized congestion feedback information, and an 8 byte opaque ID of the congestion point generating the CNM
 - 7 o The priority of the TRILL encapsulated frame causing the CNM
 - 8 o The destination MAC address of the TRILL encapsulation frame causing the CNM, RB3 in this case
 - 9 o The number of bytes included below from the beginning of the body of the TRILL encapsulation frame causing the CNM
- + Initial bytes of body of TRILL encapsulation Data frame causing the CNM
 - o TRILL Header of the frame causing the CNM
 - 10 - TRILL Ethertype 0x22F3
 - 11 - Flags, hop count, options length
 - 12 - Egress nickname, RB3 in this case
 - 13 - Ingress nickname, RB1 in this case
 - 14 - Options, if any
 - 15 o Inner.MacDA, MAC address of ESb
 - 16 o Inner.MacSA, MAC address of ESa

- 17 o Inner.VLAN tag of the TRILL encapsulated frame causing the CNM
- 18 o Optional CNtag
- 19 o Encapsulated native frame body

The ingressing TRILL switch RB2 transforms this CNM above into the following TRILL encapsulated CNM.

- + Outer.MacDA, MAC address of next hop RBridge (RB1) toward originating end station
- + Outer.MacSA, MAC address of RB2 port on which this TRILL encapsulated CNM frame is to be sent
- + Outer.VLAN tag for the designated VLAN on the RB2 to RB1 link with priority copied from incoming Outer.VLAN, field #3 above
- + TRILL Header to get the CNM to the right end station
 - o TRILL Ethertype 0x22F3
 - o Flags, hop count, options length
 - o Egress nickname, RB1 in this case, from ingress nickname in the TRILL header in the received CNM, field #13 above
 - o Ingress nickname, RB2 in this case, the nickname of the RBridge doing this transformation
 - o Options, if any
- + Inner.MacDA, MAC address of ESa, field #16 above
- + Inner.MacSA, MAC address of B2, field #2 above
- + Inner.VLAN Tag with VLAN ID from field #17 above and priority from field #3 above
- + CNtag, with flow ID from field #18 above, if #18 is present, otherwise flow ID of zero
- + CNM
 - o CNM Ethertype 0x22E7
 - o CNM version information, quantized congestion feedback information, and an 8 byte opaque ID of the congestion point generating the CNM, field #6 above
 - o The priority of the native frame whose encapsulated form caused the CNM, from Inner.VLAN, field #17 above
 - o The destination MAC address of the frame whose encapsulated form caused the CNM, the Inner.MacDA, field #15 above
 - o The number of bytes included below from the beginning of the body of the frame whose encapsulated form caused the CNM. This will be 24 smaller (but not less than zero) than the same field (#9) in the CNM tag received due to dropping the TRILL Header (8 bytes), MAC addresses (12 bytes), and Inner.VLAN (4 bytes).
- + Initial bytes of the body of the frame whose encapsulated form caused the CNM, field #19 above

Because of the reduction in the number of bytes of the body of the frame that would have caused the CNM if it weren't TRILL encapsulated, it is RECOMMENDED that bridges and TRILL switches implementing Congestion Notification in a TRILL campus be configured

to include the maximum (64) number of bytes when generating a CNM.

[3.2](#) Transit TRILL Switch Details

The subsections below describe transit TRILL switch support of Congestion Notification at input and output ports. As this is a TRILL switch in its transit role, only the handling of TRILL Data frames is discussed. If the TRILL switch is receiving a native frame, it will be an ingress as described in [Section 3.1](#) and if it is sending a native frame, it will be an egress as described in [Section 3.3](#). However, this section does apply to the output of an encapsulated frame that was ingressed at a TRILL switch and to the input, in TRILL encapsulated form, of a frame to be egressed at a TRILL switch.

[3.2.1](#) Transit TRILL Switch Input Port

The usual 802.1Q processing on the priority of the input TRILL data frame, modified as described in [Section 2.1](#), is done.

[3.2.2](#) Transit TRILL Switch Output Port

As discussed in [Section 2.1](#), a CNtag is stripped under some circumstances; however, such a CNtag will appear as part of the encapsulated frame, not on the outside of the TRILL data frame, so the CNtag is stripped from deeper in the frame. When there is a Congestion Point enabled at a TRILL switch output queue, a CNM is not generated as the result of trying to queue a TRILL control (IS-IS) frame for output at a TRILL switch port. A TRILL encapsulated CNM is generated in response to a TRILL Data frame composed as below, when to do so is specified by [\[802.1Q\]](#). The TRILL Data frame causing the CNM is referred to as TDF1 and its encapsulated native frame as NF1.

- + Outer.MacDA - MAC address of the next hop RBridge towards the egress nickname used in the TRILL Header (see below)
- + Outer.MacSA - MAC address of the output port on which the TRILL encapsulated CNM is to be sent
- + Outer.VLAN - Designated VLAN of the link on which the TRILL encapsulated CNM is to be sent
- + TRILL Header
 - o TRILL Ethertype 0x22F3
 - o Flags, hop count, options length
 - o Egress nickname, from ingress nickname in TDF1
 - o Ingress nickname, a nickname of the RBridge generating the CNM

- o Options, if any
- + Inner.MacDA - set to the Inner.MacSA of TDF1, that is, the source MAC address of NF1
- + Inner.MacSA - same as Outer.MacSA of TDF1
- + Inner.VLAN - same as the Inner.VLAN of TDF1, that is, the VLAN tag of NF1
- + CNtag - with flow ID from the CNtag of NF1 or zero if NF1 did not have a CNtag
- + CNM - message generated for NF1

[3.3](#) TRILL Switch Egress Details

After decapsulation, processing of the decapsulated native frame is the same as at any CN equipped output port. As discussed in [Section 5.1](#), any CNtag present is stripped under some circumstances. If the output queue is congested, then a native CNM may be generated in response to the decapsulated native frame. This native CNM will then be treated as if it had been received on the port.

INTERNET-DRAFT

TRILL: CN Support

[4.](#) Management Considerations

---TBD---

[5.](#) IANA Considerations

This document requires no IANA actions. RFC Editor: Please delete this section before publication.

[6.](#) Security Considerations

See [[RFC6325](#)] for general RBridge Security Considerations.

---more TBD---

INTERNET-DRAFT

TRILL: CN Support

[7](#). References

Normative and informational references for this document are given below.

[7.1](#) Normative References

[802.1Q] - IEEE, "IEEE Standard for Local and metropolitan area networks / Virtual Bridged Local Area Networks", IEEE 802.1Q-2011, May 2011.

[RFC2119] - Bradner, S., "Key words for use in RFCs to Indicate

Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997

[RFC6325] - Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (Rbridges): Base Protocol Specification", [RFC 6325](#), July 2011.

[7.2](#) Informative References

[802.1Qaz] - IEEE, "Draft Standard for Local and Metropolitan Area Networks / Virtual Bridged Local Area Networks / Amendment XX: Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes", IEEE Std 802.1Qaz-2011, June 2011.

[802.1Qbb] - IEEE, "Draft Standard for Local and Metropolitan Area Networks / Virtual Bridged Local Area Networks / Amendment: Priority-based Flow Control", IEEE Std 802.1Qbb-2011, June 2011.

[802.3] IEEE, "IEEE Standard for Information technology / Telecommunications and information exchange between systems / Local and metropolitan area networks / Specific requirements Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications", IEEE 802.3-2008, 26 December 2008.

[802.3bd] - IEEE 802.3, "Draft Standard for Information technology / Telecommunications and information exchange between systems / Local and Metropolitan Area Networks / Specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications / Amendment: MAC Control Frame for Priority-based Flow Control", IEEE Std 802.3bd-2011, June 2011.

[FCoE] - <http://fcoe.com/>

[RFC793] - Postel, J., "Transmission Control Protocol", STD 7, [RFC 793](#), September 1981

[PfcEts] - [draft-eastlake-trill-pfc-ets](#), work in progress.

Authors' Addresses

Donald Eastlake 3rd
Huawei Technologies
155 Beaver Street
Milford, MA 01757 USA

Tel: +1-508-333-2270
Email: d3e3e3@gmail.com

Manoj Wadekar
QLogic Corporation
26650 Aliso Viejo Pkwy
Aliso Viejo, CA 92656 USA

Tel: +1-949-389-6000
Email: manoj.wadekar@qlogic.com

Anoop Ghanwani
Dell
350 Holger Way
San Jose, CA 95134 USA

Phone: +1-408-571-3500
Email: anoop@alumni.duke.edu

Puneet Agarwal
Broadcom
3975 Freedom Circle
Santa Clara, CA 95054 USA

Phone: +1-949-926-5000
Email: pagarwal@broadcom.com

Tal Mizrahi
Marvell
6 Hamada Street
Yokneam, 20692 Israel

Email: talmi@marvell.com

INTERNET-DRAFT

TRILL: CN Support

Copyright, Disclaimer, and Additional IPR Provisions

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of [RFC 5378](#). No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under [RFC 5378](#), shall have any effect and shall be null and void, whether published or posted by such Contributor, or included with or in such Contribution.

