Update on Cut-Through Forwarding (CTF)

Johannes Specht

Preamble

- **This Presentation collects the Author's thoughts on cut-through forwarding.**
- **It is intended to move towards a common view in IEEE 802.1 amongst goals, needs, and operation of potential IEEE 802 standardization activities on CTF.**
- **This is an individual contribution by the presenter (Johannes Specht).**

Review

January 2020

- https://www.ieee802.org/1/files/public/docs2020/new-specht-cut-through-tech-0120-v01.pdf
- Where CTF matters, and where not
- CTF-specific issues and mitigations
- Proposed contents of a Standard

Subsequent Discussions

- Relevant IEEE 802.1 Standards: IEEE 802.1AC, IEEE 802.1CB, IEEE 802.1Q
- More discussion in IEEE 802.1 needed
- Problem description for IEEE 802.3 needed
- Presentations, WYSIWYG document

A Path to Move Forward

CTF standard contents (not a Standards document clause structure)

- **Existing Bridge functions/protocols** (e.g., pieces from the 802.1Q-2018 forwarding process), **and modifications** (if needed)
- **New Bridge functions** (e.g., tail cutting/CRC invalidation)
- **Management Interfaces**, Counters, etc.
- **Where CTF Matters** - and where not?
- **Header corruption mitigation**
- **Device conformance**
- **Network conformance** - IEEE conformant networks using CTF should apply mitigations to handle CTF-specific issues!
- **MAC Service Interface (???)**, frame reception

From https://www.ieee802.org/1/files/public/docs2020/new-specht-cut-through-tech-0120-v01.pdf

WYSIWYG document for subsequent technical discussions in 802.1 & 802.3

- Preparation after this Meeting
- Contributed "lightweight" working document
  - First step towards a new CTF 802.1 Standard
  - High-Level, short, easy to adjust
  - Arrange discussed contents
  - Describes challenges to solve: "Loops", "No standardized MAC", etc.
  - *Get feedback from 802.1 participants!*
- ***NOTE: A potential subsequent "Editor's draft" may or may not look like this!***

Structural Proposal – Top Level

- 1. Overview
- 2. Normative References
- 3. Definitions ⎤
- 4. Abbreviations ⎬ Not now
- 5. Conformance ⎦
- 6. Cut-Through Forwarding in Networks
- 7. Cut-Through Forwarding Relay
- 8. Managed Objects ⎤
- 9. YANG Modules ⎬ Not now
- A. PICS ⎥
- B. Bibliography ⎦

Structural Proposal – Contents

- ### 1. Overview
  - *Introduction, Glue for Subsequent Clauses*

- ### 2. Normative References
  - IEEE Std 802.3
  - IEEE Std 802.1 802.1AC-2016, Cor 1-2018
  - IEEE Std 802.1 802.1Q-2021
  - IEEE Std 802.1 802.1CB-2017, IEEE Std 802.1 CBcv-2021

- ### 6. Cut-Through Forwarding in Networks
  - Introduce Terms (e.g., "Bridges with Cut-Through Forward Relay Support")
  - Network Structure and Elements *(Wired P2P, Extensions in Bridges, No Extensions in End Stations)*
  - High-Level Use-Cases (application independent): Topologies, Traffic Patterns/Scheduling*
  - Performance Considerations *(a.k.a. where CTF matters, and where not)*
  - Loops and Loop Prevention *(Header corruptions, impact, etc.)*
  - Link Speed transitions

  Specific/Heterogenous Structure
  - Compact, Readable, Etc.
  - *To be determined …*

- ### 7. Cut-Through Forwarding Relay
  - Bridge Port Transmit and Receive *(Demultiplexing, etc.)*
  - Augmented Forwarding Process
  - *Forwarding Process Function 1..n (Existing ones included, and new ones)*

  Straight Forward Structure
  - Essentially a Bridge Pipeline
  - Details on next slides

Note *: Term "Scheduling" is used in the broad sense for shaping, coordinated transmission times, TDM, etc. (i.e, not tied to what is called "Scheduled Traffic" in IEEE Std 802.1Q-2018).

High-Level Model: *Transient Frames, Complete Frames, Stalls,* and *Late Discarding*

Properties of *Transient Frames*

- Device Internal
- Content change over time
- *Late discarding* (e.g., FCS errors)
- Only for Relaying path
- Relay stages *stalled* until enough content is available

Distinction in Descriptions

- *Transient Frames* v.s.
- *Complete Frames* (just "Frame" in IEEE 802.1Q)

Transient Frames v.s. Complete Frames

- Receive Timing
  - *Transient Frames*:   At Frame Start from the Wire
  - *Complete Frames*:   After Frame End from the Wire
- Transient Frames can be completed
  - Become Complete Frames
    (if FCS ok, and not yet under transmission)
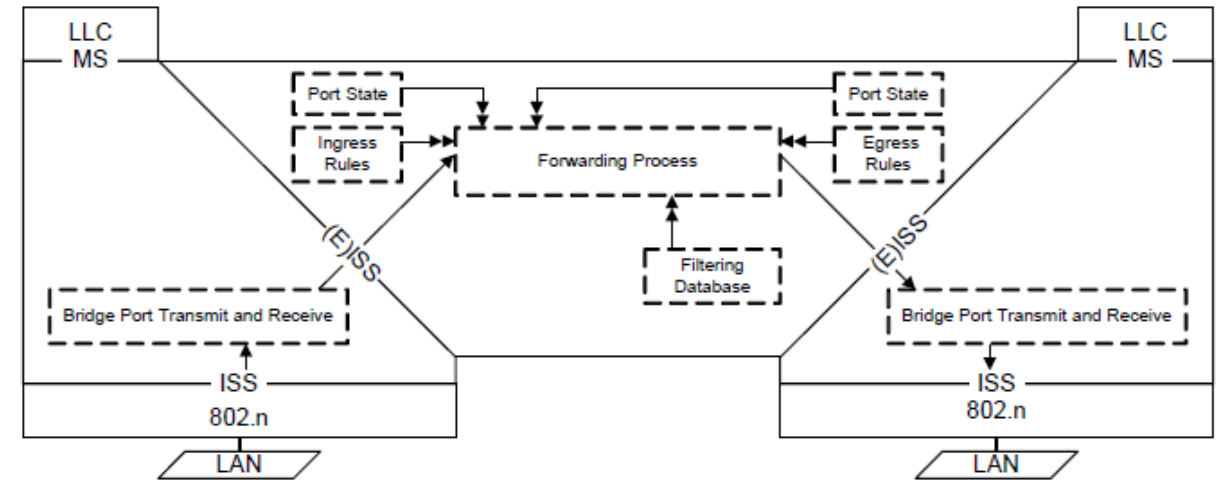 - *Late discarding*
    (e.g., if FCS is not ok)

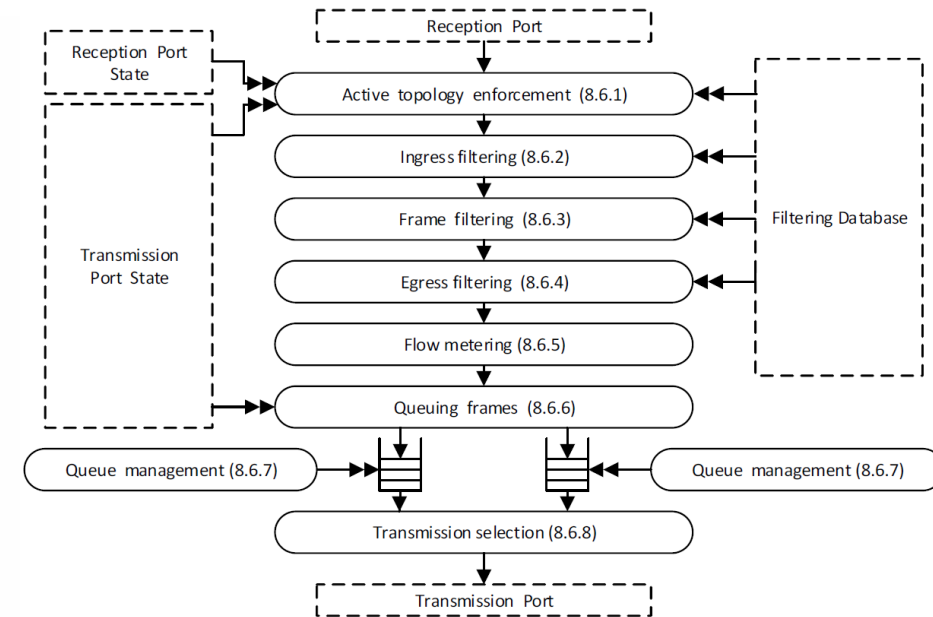

Figure 8-4—Relaying MAC frames

Figure 8-12—Forwarding process functions

Structural Proposal – Initial Refinement

- Relaying of transient frames
- Introduce:
  - "diff"-clause concept
  - Essentially only for transient frames
  - Absent 802.1* functions: unsupported
  - Pipeline stages/stalls/late discarding

- Initial List (subclauses added/removed over time)
- Case-by-case diff to 802.1Q
  - Min: "As described in A.B.C of 802.1Q-20XX."
  - Typical: Different handling of transient frames/late discarding
  - Max: New stages (not illustrated)

7. Cut-Through Forwarding Relay

7.1 General

7.2 Bridge Port Receive

7.3 Active Topology Enforcement

7.4 Ingress Filtering

7.5 Frame Filtering

7.6 Egress Filtering

7.7 Flow Classification and Metering

7.8 Queuing Frames

7.9 Queue Management

7.10 Transmission Selection

7.11 Bridge Port Transmit

- Relationship:
  - Transient Frames v.s.
  - Complete Frames v.s.
  - M_UNITDATA.indication
- Never (!) send transient frames to higher layers
- Stage may stall:
  - Any stage could do so, if transition to Complete Frame required

- Multiplexing:
  - Higher Layer PDUs,
  - Transient Frames,
  - Complete Frames
- Handling late discarding of transient frames

Note: Arrangement of *Bridge Port Receive* and *Bridge Port Transmit* is in a pipeline manner, not in a combined/layered manner (like in IEEE Std 802.1Q). This proposal is a trade-off: On the one hand, the pipelined manner appeared more readable to the author, and it's no obligation to organize contents identical to IEEE Std 802.1Q. On the other hand, symmetry might be helpful for readers familiar with IEEE Std 802.1Q.

Discussion

Update on CTF, Johannes Specht

Document Aspects

1. Call it *Bridge*, or different?

- Dedicated 802.1 Standard proposed (i.e., not an 802.1Q amendment)
- Bridges may have support for this, or not
- Operation may be applicable beyond 802.1Q Bridges

2. Conformance Clause in the working document?

- Not a Standard, however …
- … can summarize mandatory (shall), recommended (should), and optional elements (may) of
  - Devices
  - Networks (?)

3. Other document discussion?
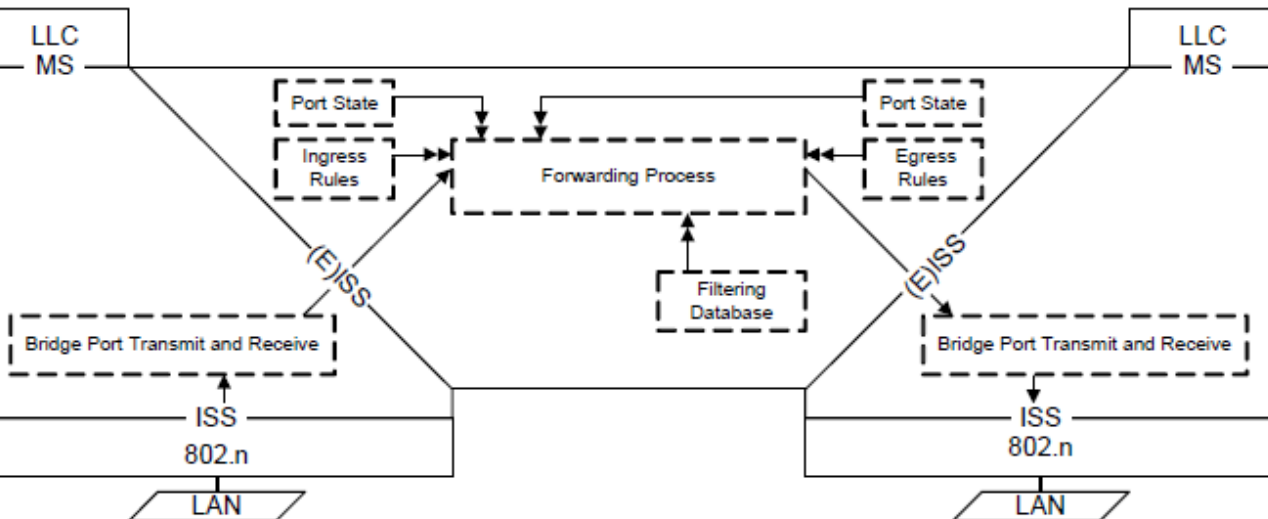
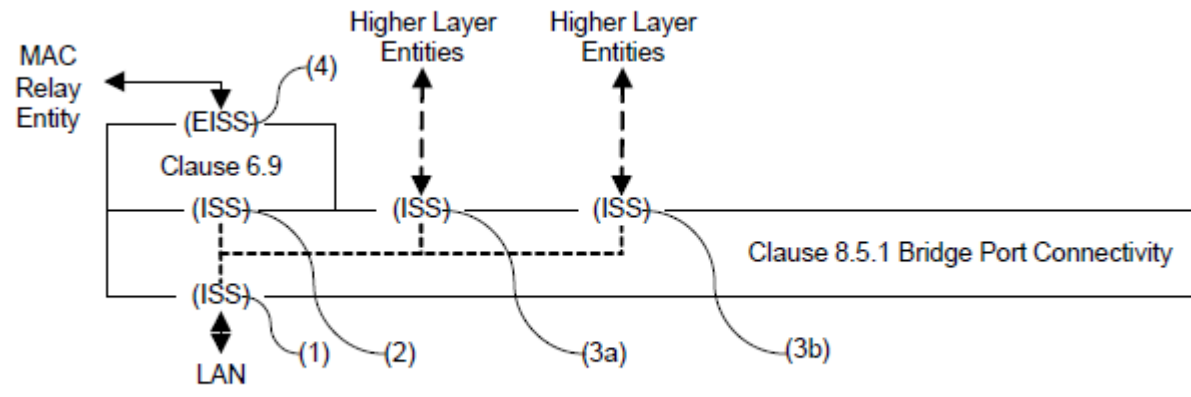# Criteria to stall until completion/discarding

Figure 8-4—Relaying MAC frames

Figure 8-10—Bridge Port Transmit and Receive

Cmp. 8.3 and 8.5 of 802.1Q-2018

Problem
- Stall Transient Frames until they transition to Complete Frames (or are discarded) in several stages
- Avoid transfer to Higher Layer Entities

Discussion
- Tagged Frames (i.e., PCP)?
- Selected Stream Identification Functions (IEEE 802.1CB)?
- FDB?
- Transmit-Port based decision?
- Other?

Other Technical Aspects

1. Preemption: CTF only for non-preemptible traffic assumed - if not …
   - "Late preemption"?

2. Late Discarding (supported/unsupported relay functions)
   - Bad FCS (obvious)
   - Flow Metering
     - Max. SDU Size Filtering?
     - Stream Gates: closed, open→closed, IntervalOctetsLeft?
     - Flow Meters (a.k.a. MEF 10.3 Token Bucket Meters)?
   - drop_eligible?
   - queueMaxSDU?

3. Configuration: Static only assumed - if not …
   - (R)STP?
   - Dynamic Filtering Entries?

3. Other items?

Thank you for your Attention!

Questions, Opinions, Ideas?

Johannes Specht

Dipl.-Inform. (FH)

Kurfürstenwall 2
45657 Recklinghausen
North Rhine-Westphalia
GERMANY
M +49 (0)170 718-4422
johannes.specht.standards@gmail.com