PFC Enhancement Discussion ---PFC Management

July 2022

Lily Lv

Topics

1. PFC interface stack diagram
2. PFC and link aggregation
3. MAC privacy protection on PFC
4. Where to specify PFC shim?
5. PFC management

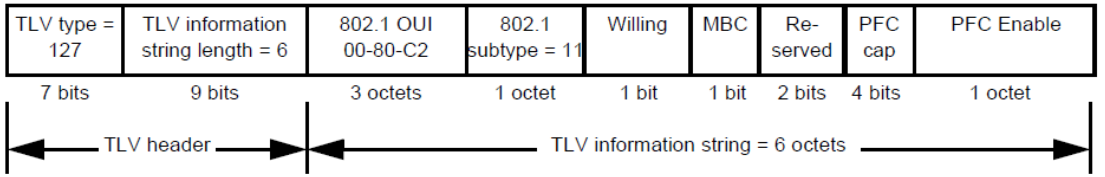
Topic 5: PFC Management

Review current PFC management and determine new content for enhanced PFC functions?

Existing PFC Management Contents in Spec

Variables of existing PFC management in **802.1Q(Rev d1-02)** includes managed objects and DCBX TLVs.

Managed objects	• PFCLinkDelayAllowance • PFCRequests • PFCIndications	**12.23 Priority-based Flow Control objects** **Table 12-21—Priority-based Flow Control objects** Name / Data type / Operations supported[a] / Conformance[b] PFCLinkDelayAllowance — unsigned integer — RW — BE PFCRequests — unsigned integer — R — BE PFCIndications — unsigned integer — R — BE [a] R = Read only access; RW = Read/Write access. [b] B = Required for Bridge or Bridge component support of PFC; E = Required for end station support of PFC.

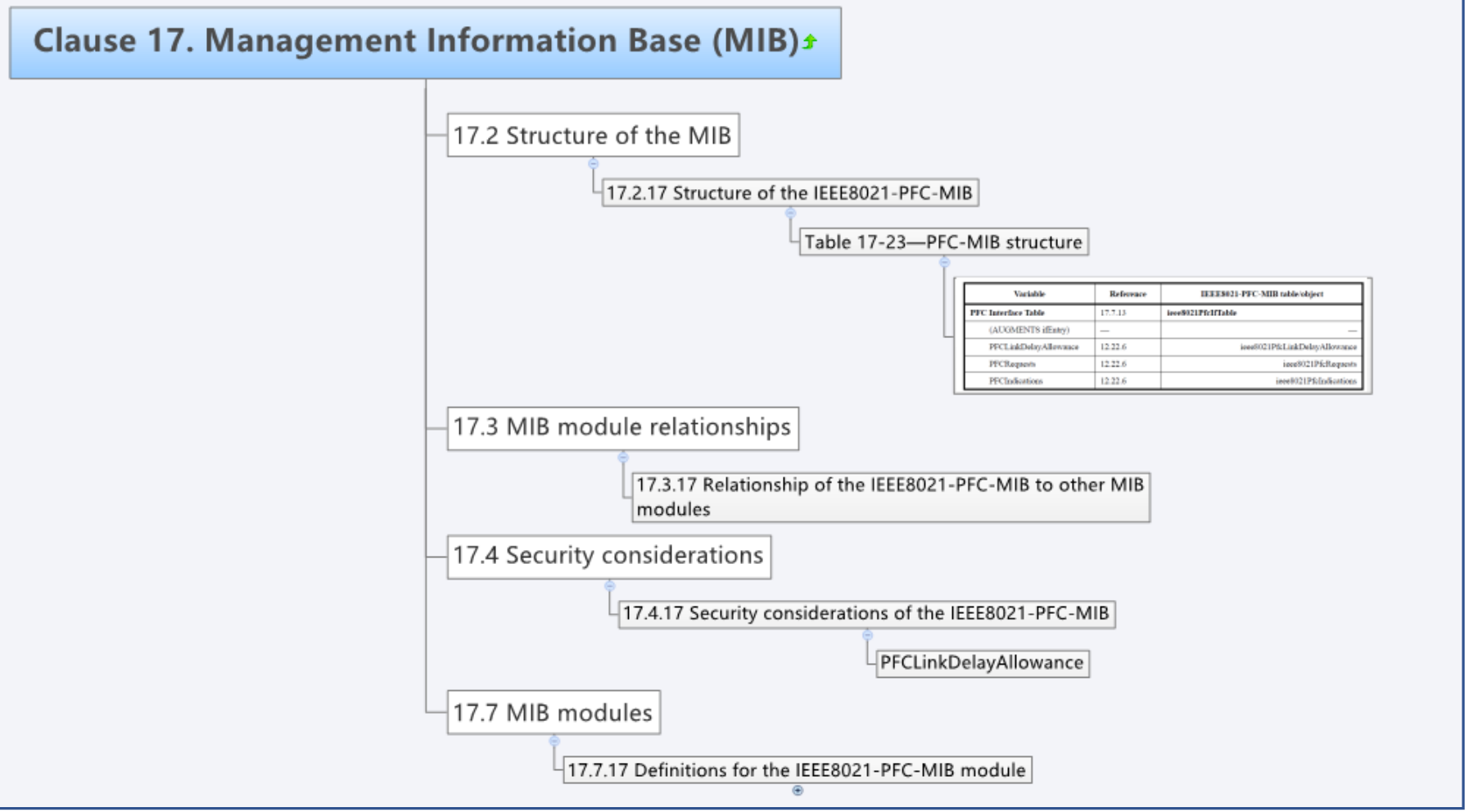
DCBX TLVs	• dcbxSet
	  • PFC configuration TLV
	  • Willing
	  • MBC
	  • PFC cap
	  • PFC Enable
		D.1 Requirements of the IEEE 802.1 Organizationally Specific TLV sets
		Table D-1—IEEE 802.1 Organizationally Specific TLVs
		IEEE 802.1 subtype / TLV name / TLV set name / TLV reference / Feature clause reference
		0B — Priority-based Flow Control Configuration TLV — dcbxSet — D.2.10 — Clause 38
		D.2.10 Priority-based Flow Control Configuration TLV
		 
		Figure D-10—Priority-based Flow Control Configuration TLV format

Existing PFC Management Contents in Spec

MIB is defined for managed objects in **802.1Q clause 17** .

Managed objects

- PFCLinkDelayAllowance
- PFCRequests
- PFCIndications

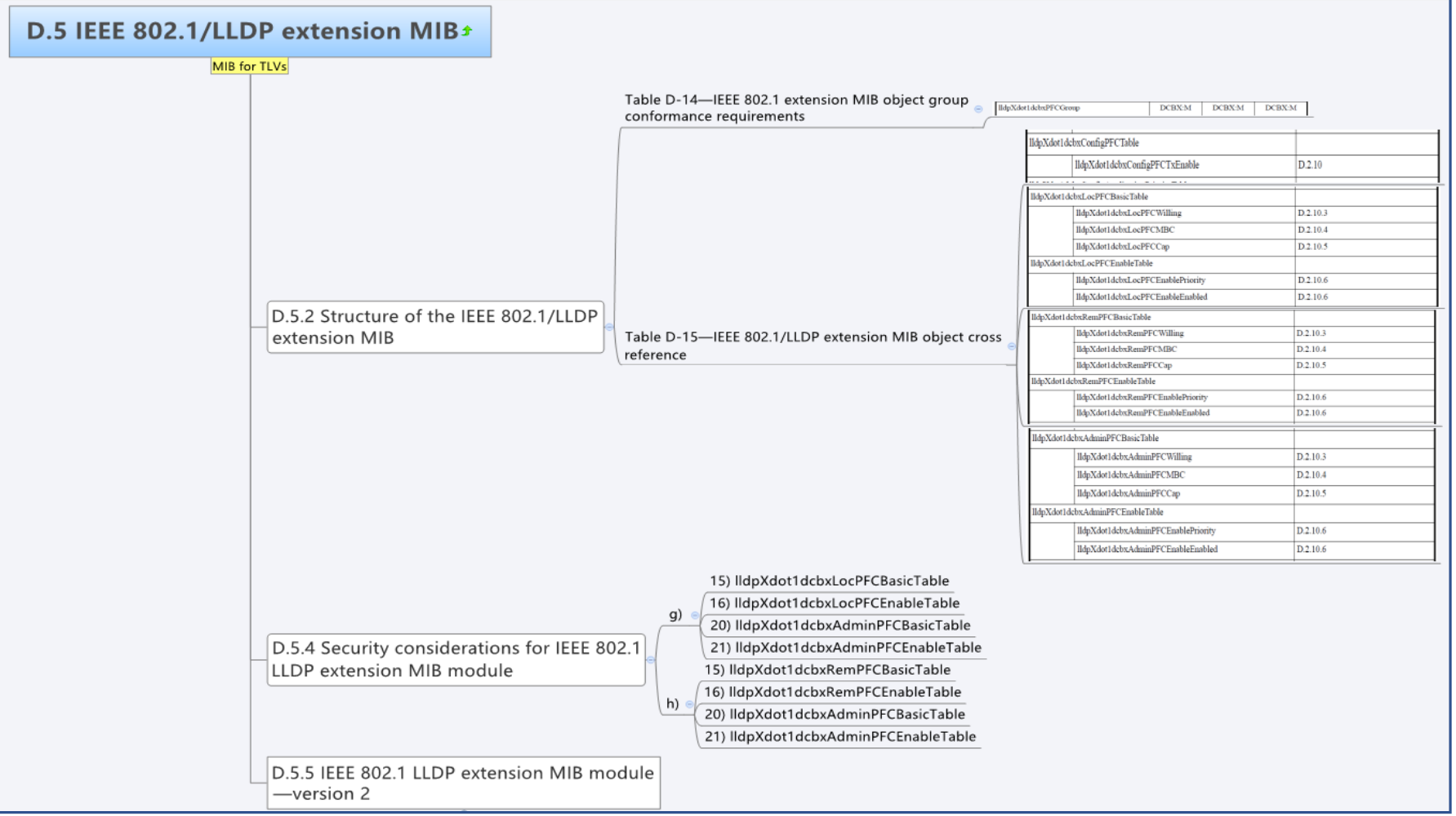


Existing PFC Management Contents in Spec

MIB is defined for DCBX TLVs in **802.1Q Annex D**.

DCBX TLVs

- dcbxSet
  - PFC configuration TLV
    - Willing
    - MBC
    - PFC cap
    - PFC Enable

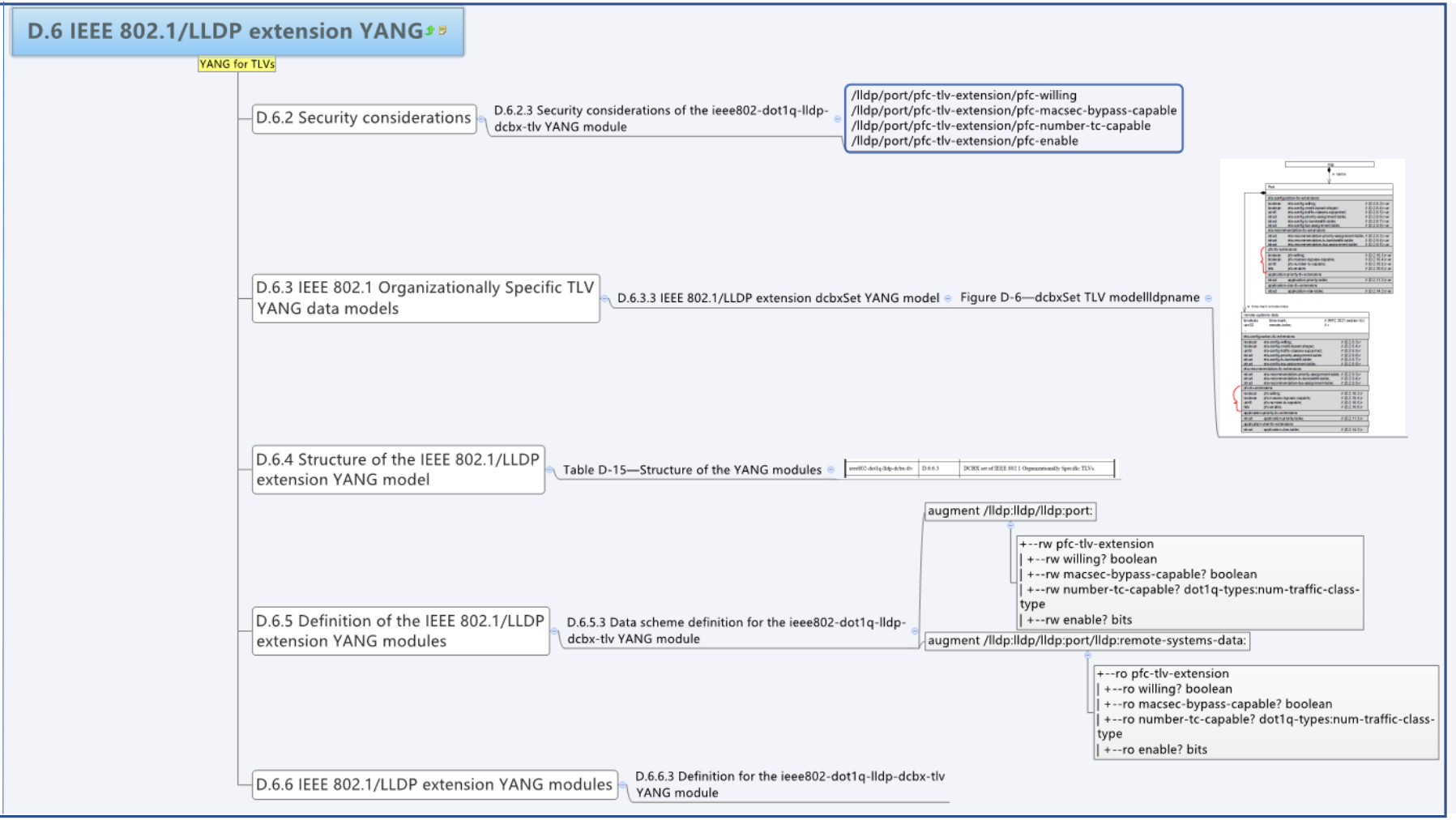


Existing PFC Management Contents in Spec

YANG is defined for DCBX TLVs in **P802.1Qcz Annex D**.

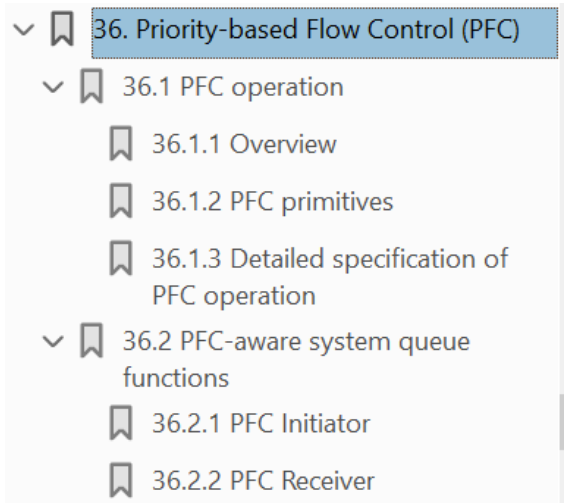
DCBX TLVs

- dcbxSet
  - PFC configuration TLV
    - Willing
    - MBC
    - PFC cap
    - PFC Enable

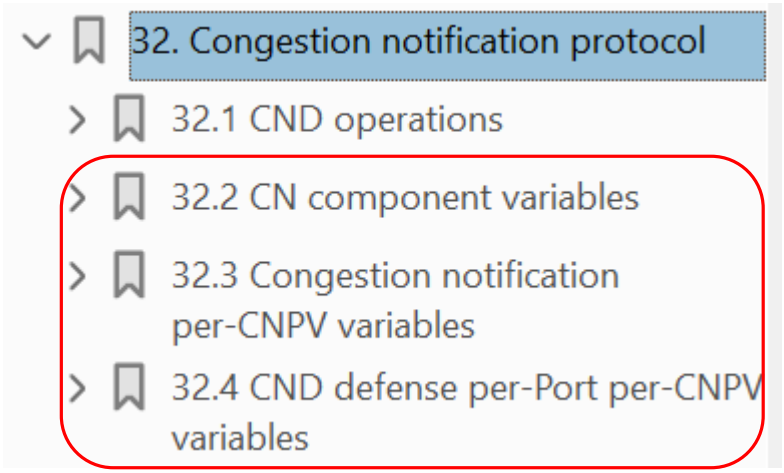


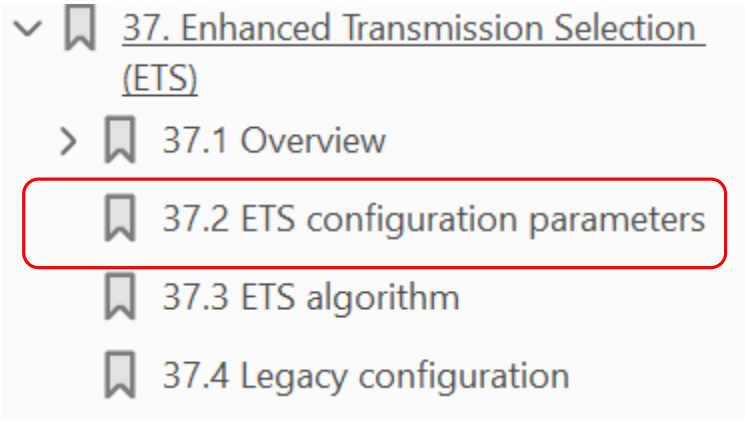
Problem of Existing PFC Management Contents in Spec

1) In clause 36, there is no subclause of PFC variables while other datacenter functions(e.g congestion notification, ETS) have.



vs.





PFC *CN (QCN)* *ETS*

Proposal : Subclause for PFC variables should be added.

Problem of Existing PFC Management Contents in Spec

2) Besides managed objects, PFC variables should contain internal variables.

There are internal variables related description in 36.1.3 Detailed specification of PFC operation, see figure 36-2.

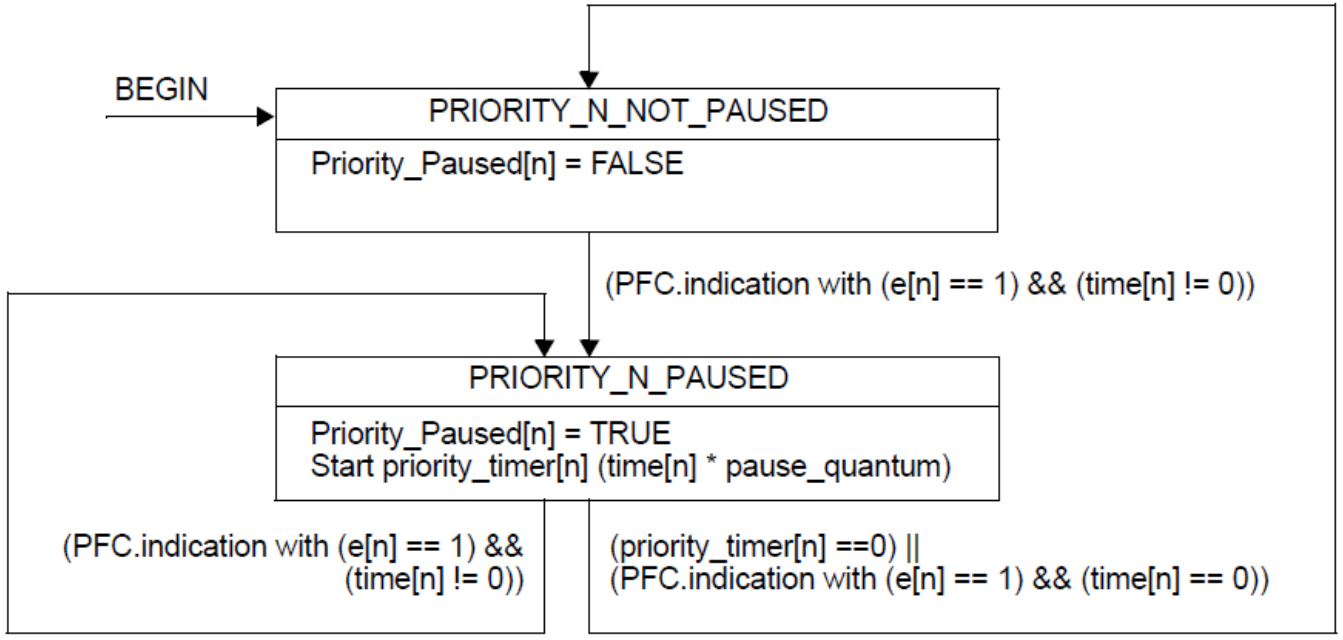


Figure 36-2—PFC Receiver state diagram for priority n

Figure 36-2 shows several variables.

- Priority_Paused[n]
- priority_timer[n]
- pause_quantum
- e[n]
- time[n]

Problem of Existing PFC Management Contents in Spec

2) Besides managed objects, PFC variables should contain internal variables.

Priority_Paused[n] :

- "The PFC Receiver entity maintains … the Priority_Paused[n] variables, indicating **the state of each of the eight priorities.**"

priority_timer[n]:

- No clear definition, only could be inferred from below description.
- "**priority_timer[n] (time[n] * pause_quantum)**"

pause_quantum

- No clear definition, only could be inferred from below description in later subclause (36.1.3.3 Timing considerations)
- "This delay is equivalent to **12 pause quanta (i.e., 6144 bit times) at the speed of 10 Gb/s**, 48 pause quanta (i.e., 24 576 bit times) at the speed of 40 Gb/s, and 120 pause quanta (i.e., 61 440 bit times) at the speed of 100 Gb/s. "

Problem of Existing PFC Management Contents in Spec

2) Besides managed objects, PFC variables should contain internal variables.

e[n]

- "priority_enable_vector: a 2-octet field, with the most significant octet being reserved (i.e., set to zero on transmission and ignored on receipt). **Each bit of the least significant octet indicates if the corresponding field in the time_vector parameter is valid. The bits of the least significant octet are named e[0] (the LSB) to e[7] (the MSB)**. Bit e[n] refers to priority n. For each e[n] bit set to one, the corresponding time[n] value is valid. For each e[n] bit set to zero, the corresponding time[n] value is invalid."

time[n]

- "time_vector: a list of eight 2-octet fields, named time[0] to time[7]. The eight time[n] values are always present regardless of the value of the corresponding e[n] bit. **Each time[n] field** is a 2-octet, unsigned integer **containing the length of time for which the receiving station is requested to inhibit transmission of data frames associated with priority n**. "

The request_operand_list of a PFC M_CONTROL.request and the indication_operand_list of a PFC M_CONTROL.indication are composed of the following operands:
priority_enable_vector
time_vector

Problem of Existing PFC Management Contents in Spec

2) Besides managed objects, PFC variables should contain internal variables.

e[n] and time[n] come from M_CONTROL primitives.
e[n] and time[n] are used to form PFC pause frame.

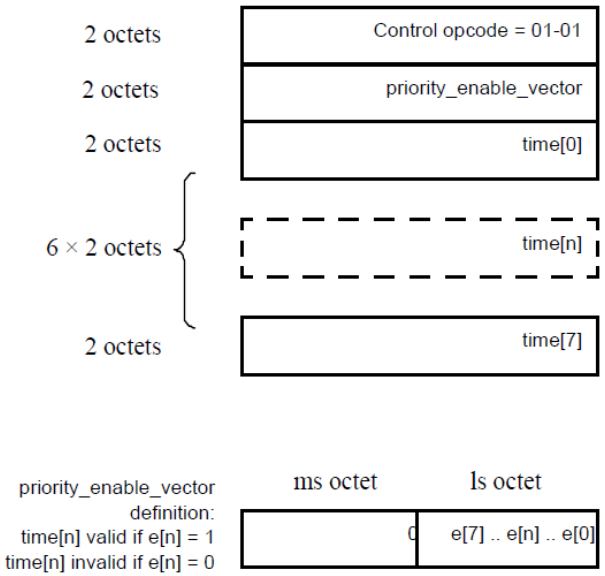


Figure M-1—PFC PDU format

e[n] is not the same value as TLV field " PFC Enable", but should be aligned with "PFC Enable".

D.2.10.6 PFC Enable

Table D-6 shows the layout of the PFC Enable bit vector.

Table D-6—PFC Enable bit vector

Octet:	1							
	Priority 7	Priority 6	Priority 5	Priority 4	Priority 3	Priority 2	Priority 1	Priority 0
Bits:	7	6	5	4	3	2	1	0

A bit vector of 8 bits, one per priority:

a) A one indicates PFC is enabled on the priority.
b) A zero indicates that PFC is disabled on the priority.
c) Local policy in each end of the link decides whether to use the priority if the configuration does not match.

Proposal :

PFC internal variables should include Priority_Paused[n], priority_timer[n], pause_quantum, e[n] and time[n]

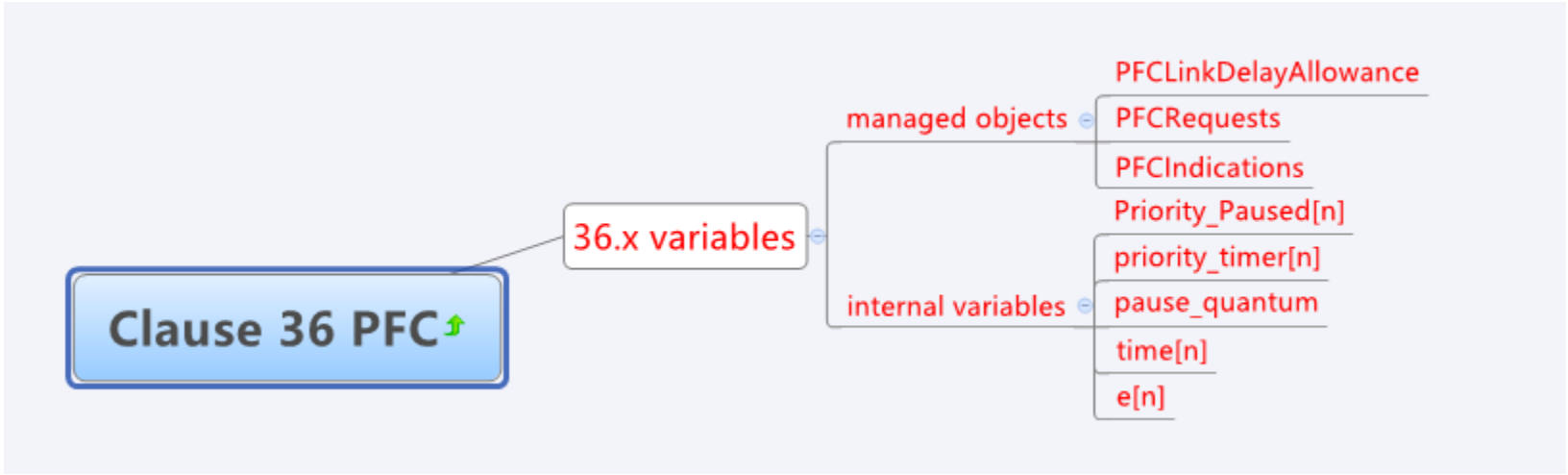
Problem of Existing PFC Management Contents in Spec

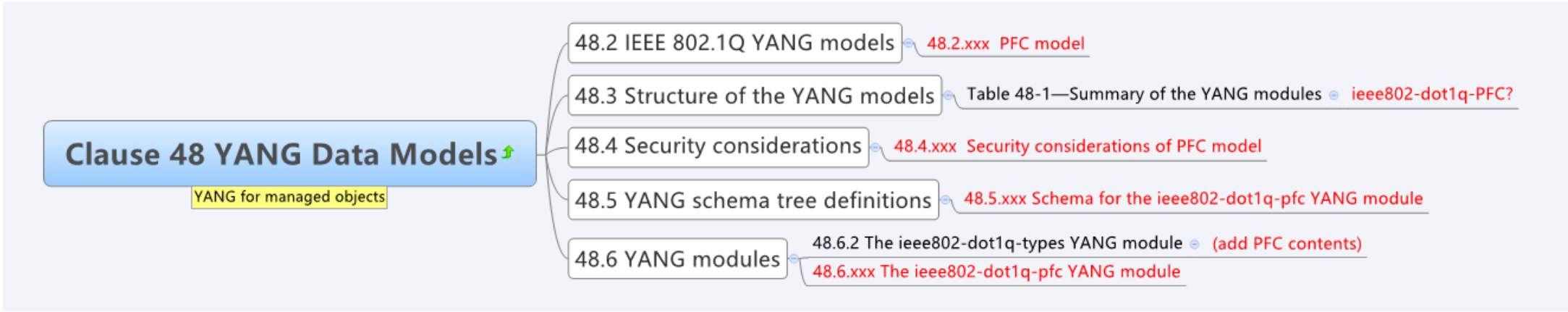
3) Clause 48 YANG Data Models do not have PFC contents.

Clause 48 is YANG models for managed objects. Existing PFC has 3 managed objects.

Proposal:
YANG models of PFC managed objects should be added in clause 48.

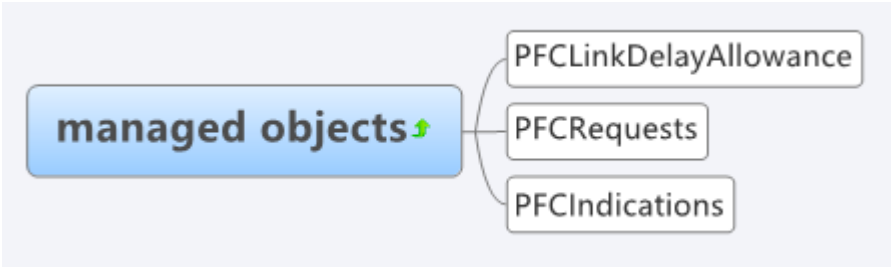
Proposed Fix for Existing PFC Management





New(Qdt--Headroom) for PFC Management in Spec

1) Do we need a new managed object for automatic calculated headroom value?



- PFCLinkDelayAllowance is an existing managed object.

- The definition of PFCLinkDelayAllowance is ,
"PFCLinkDelayAllowance: the allowance made for round-trip propagation delay of the link in bits"

Headroom has the same meaning.

- There is a note to describe the function of PFCLinkDelayAllowance.
"NOTE—The PFC Initiator (see 36.2.1) can use the PFCLinkDelayAllowance parameter as one of the factors to determine when to issue a PFC M_CONTROL.request in order to not discard frames. The parameter can be written to adjust to different link characteristics that affect the link delay (e.g., link length or link technology). See Annex N for an example of how to compute this parameter. "

- PFCLinkDelayAllowance is manually set by administrator currently.

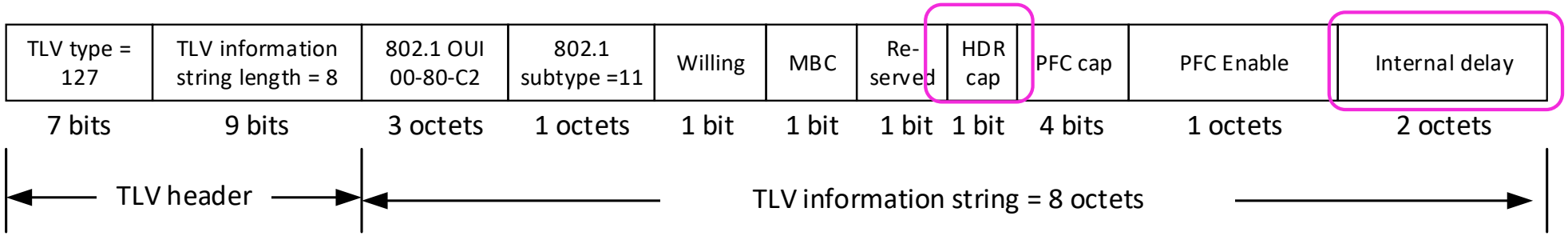
Headroom has the same function, but is calculated automatically

Proposal :
Use PFCLinkDelayAllowance to represent headroom.
Describe how to use PFCLinkDelayAllowance ------- manual setting will override automatic calculated value.

New(Qdt--Headroom) for PFC Management in Spec

2) New added TLVs.

- Two new fields in DCBX TLVs are proposed for automatic headroom calculation.
  HDR cap:
  1 bit taken from Reserved field, indicating if automatic headroom calculation is supported.
  Internal delay:
  2 octets added at the tail, representing the length of time for which the device process received PFC pause frame.



Proposal :

Update D.2.10 Priority-based Flow Control Configuration TLV, adding the 2 new fields in TLV

New(Qdt--Headroom) for PFC Management in Spec

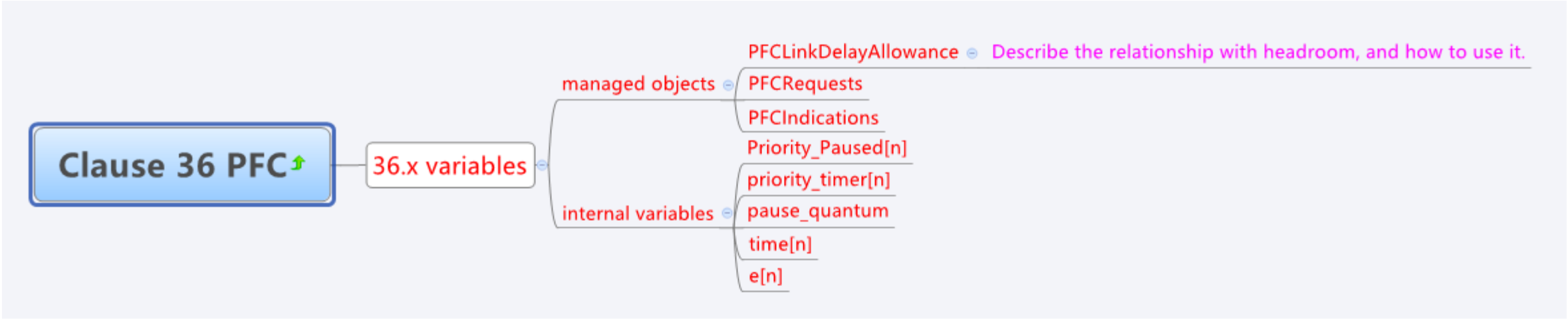
3) MIB for new added TLVs.

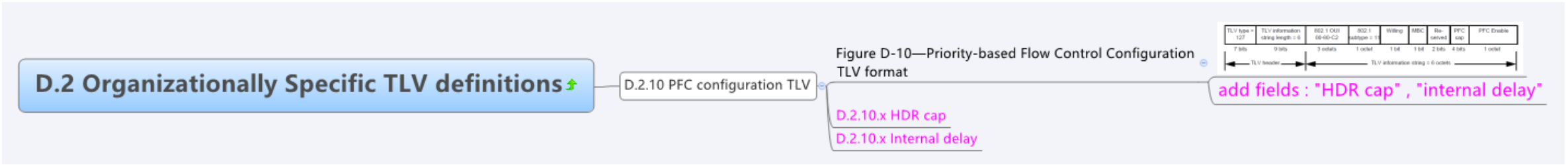
Proposal :
Update D.5 IEEE 802.1/LLDP extension MIB， adding contents for new added TLVs.

4) YANG for new added TLVs.

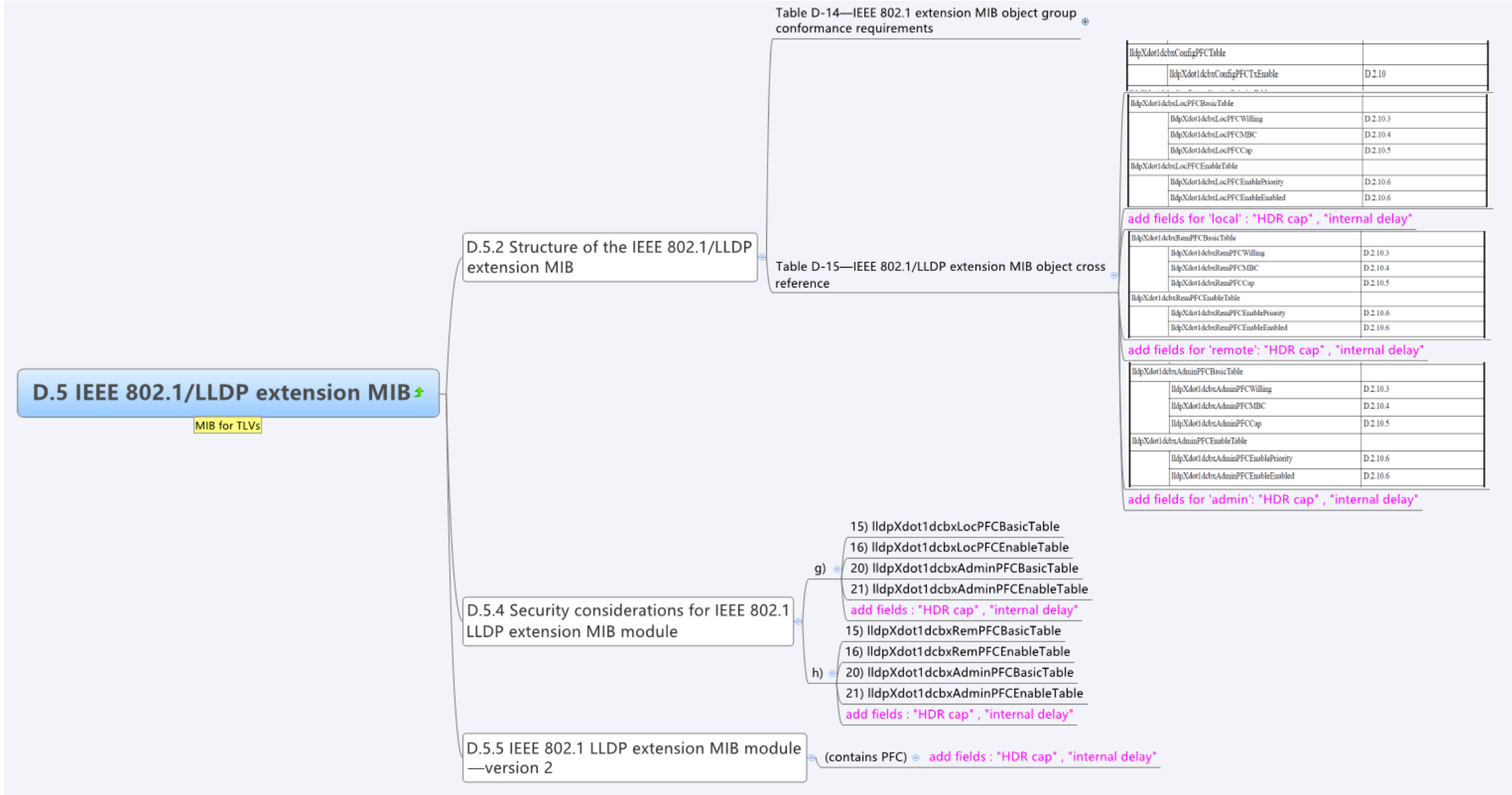
Proposal :
Update D.6 IEEE 802.1/LLDP extension YANG （in P802.1Qcz） ， adding contents for new added TLVs.

Proposed Updates for New(Qdt--Headroom) PFC Management





Proposed Updates for New(Qdt--Headroom) PFC Management

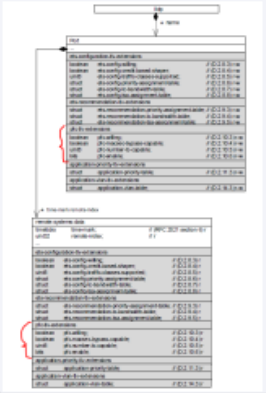


Proposed Updates for New(Qdt--Headroom) PFC Management

D.6 IEEE 802.1/LLDP extension YANG

YANG for TLVs

- D.6.2 Security considerations
  - D.6.2.3 Security considerations of the ieee802-dot1q-lldp-dcbx-tlv YANG module
    - /lldp/port/pfc-tlv-extension/pfc-willing
    - /lldp/port/pfc-tlv-extension/pfc-macsec-bypass-capable
    - /lldp/port/pfc-tlv-extension/pfc-number-tc-capable
    - /lldp/port/pfc-tlv-extension/pfc-enable
    - add fields : "HDR cap" , "internal delay"
- D.6.3 IEEE 802.1 Organizationally Specific TLV YANG data models
  - D.6.3.3 IEEE 802.1/LLDP extension dcbxSet YANG model
  - Figure D-6—dcbxSet TLV modelIlldpname



  - add fields : "HDR cap" , "internal delay"
- D.6.4 Structure of the IEEE 802.1/LLDP extension YANG model
  - Table D-15—Structure of the YANG modules

ieee802-dot1q-lldp-dcbx-tlv	D.6.3	DCBX set of IEEE 802.1 Organizationally Specific TLVs

- D.6.5 Definition of the IEEE 802.1/LLDP extension YANG modules
  - D.6.5.3 Data scheme definition for the ieee802-dot1q-lldp-dcbx-tlv YANG module
    - augment /lldp:lldp/lldp:port:

```
+--rw pfc-tlv-extension
| +--rw willing? boolean
| +--rw macsec-bypass-capable? boolean
| +--rw number-tc-capable? dot1q-types:num-traffic-class-type
| +--rw enable? bits
```

    - add fields : "HDR cap" , "internal delay"
    - augment /lldp:lldp/lldp:port/lldp:remote-systems-data:

```
+--ro pfc-tlv-extension
| +--ro willing? boolean
| +--ro macsec-bypass-capable? boolean
| +--ro number-tc-capable? dot1q-types:num-traffic-class-type
| +--ro enable? bits
```

    - add fields : "HDR cap" , "internal delay"
- D.6.6 IEEE 802.1/LLDP extension YANG modules

New(Qdt--MACsec) for PFC Management in Spec

Do we need a variable to enable/disable MACsec protection on PFC frame?

Discussion: how does MACsec enabled on normal MAC data frames? Any variable is defined?

Proposal : Reflect MACsec protection capability in PFC configuration TLV

1) Option 1: Reuse existing field ‘MBC’ in TLV

2) Option 2: Add a new field in TLV

New(Qdt--MACsec) for PFC Management in Spec

1) Option 1: Reuse existing field ‘MBC’ in TLV

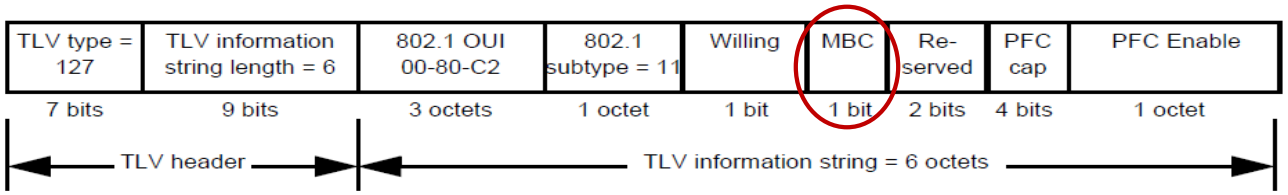


Figure D-10—Priority-based Flow Control Configuration TLV format

- What is MBC?

“The MACsec Bypass Capability Bit. If set to zero, the sending station is capable of bypassing MACsec processing when MACsec is disabled. If set to one, the sending station is not capable of bypassing MACsec processing when MACsec is disabled (see Clause 36).”

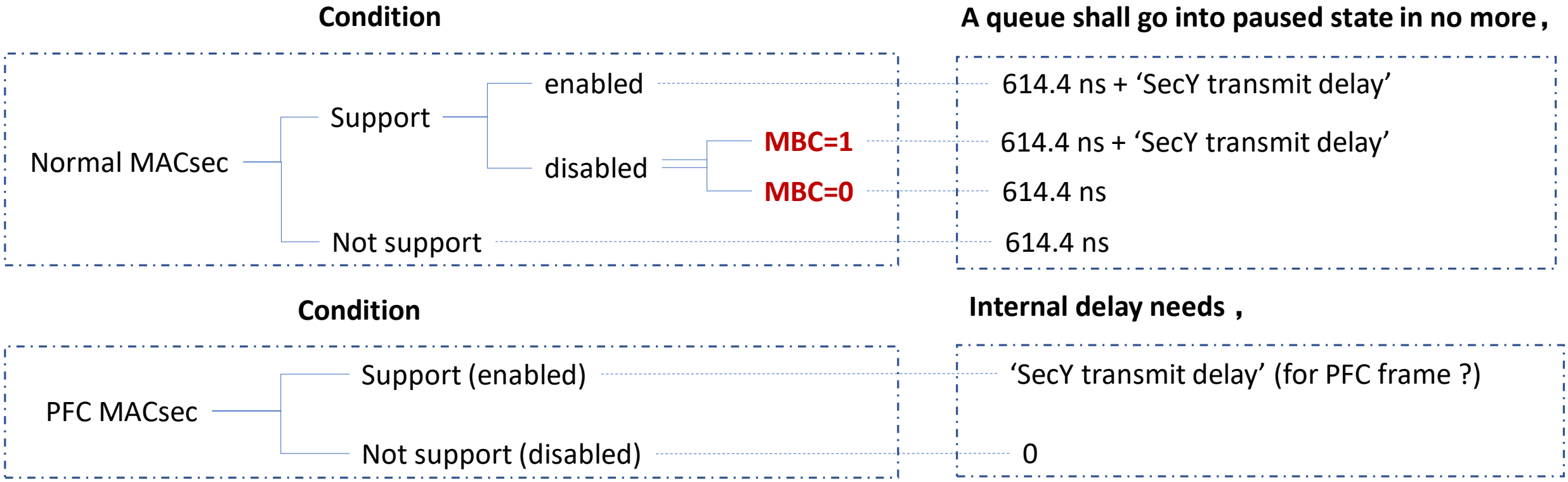
MACsec does not support PFC. So “MACsec is disable’ talks about MACsec protection on normal MAC data frames. The station supports MACsec but it can enable or disable the capability. If MACsec is disabled, extra delay still has to be considered for headroom when MBC is set to one.

This is aligned with description in clause 36.1.3.3 Timing considerations.
“If MACsec is supported but not used, the delay computation has to take into account the MACsec Bypass Capability (MBC) bit in the PFC configuration TLV of DCBX (see IEEE Std 802.1Qaz subclause 38.5.4), that indicates if the link peer needs the extra time for MACsec. If the MBC bit is set to zero, the maximum PFC delay is 614.4 ns. If the MBC bit is set to one, the maximum PFC delay is 614.4 ns + ‘SecY transmit delay’. ”

New(Qdt--MACsec) for PFC Management in Spec

1) Option 1: Reuse existing field ‘MBC’ in TLV

- ‘MBC’ has different meaning with PFC MACsec capability.



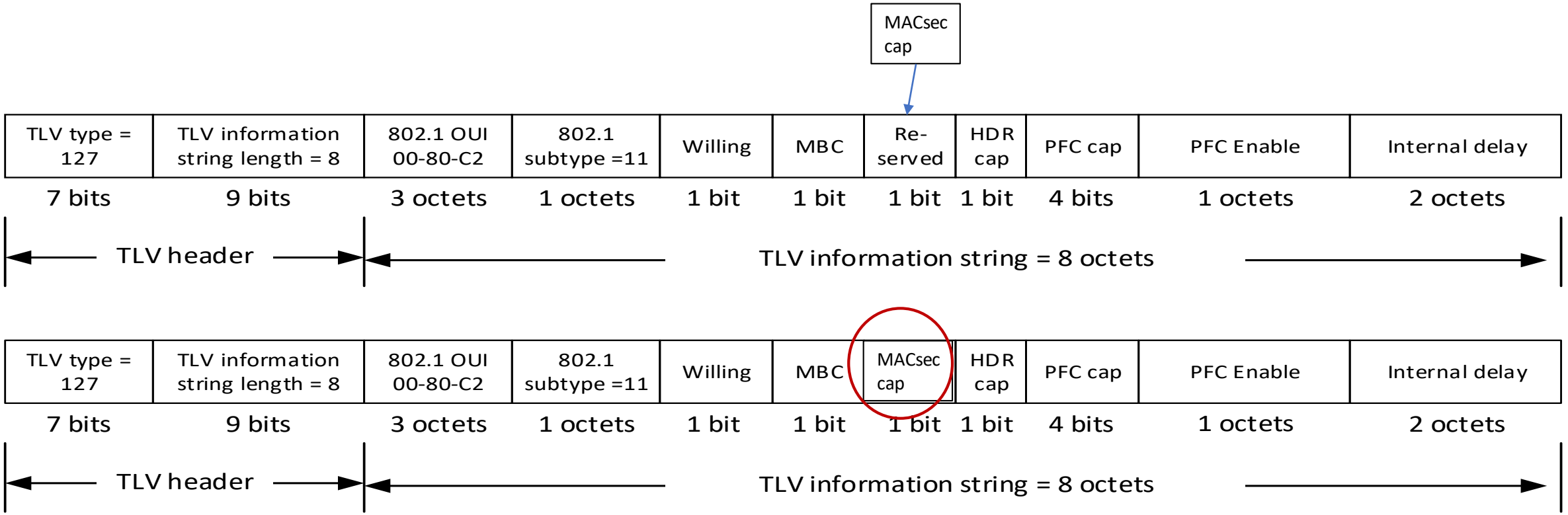
Option 1 has issue, MBC cannot be reused.

“‘SecY transmit delay’ is defined as the wire transmit time for a maximum sized MPDU + 4 times the wire transmit time for 64 octet MPDUs. ”

New(Qdt--MACsec) for PFC Management in Spec

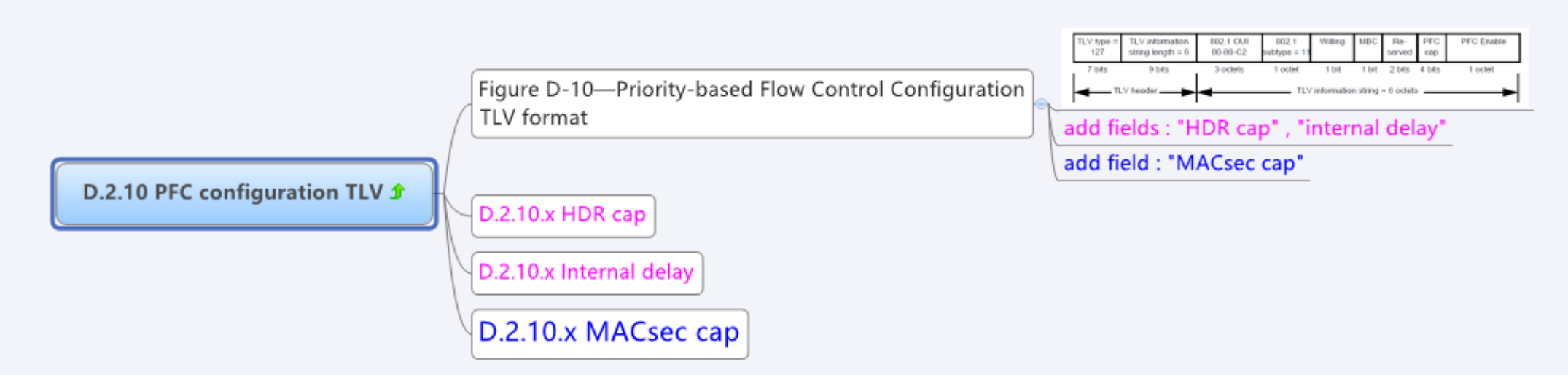
2) Option 2: Add a new field for MACsec protection on PFC frame in TLV

- MACsec cap: The MACsec capability bit. If set to zero, the sending station is NOT capable of protecting PFC frame by MACsec. If set to one, the sending station is capable of protecting PFC frame by MACsec.
- Take the reserved 1 bit as MACsec cap field. TLV information string length does not change.

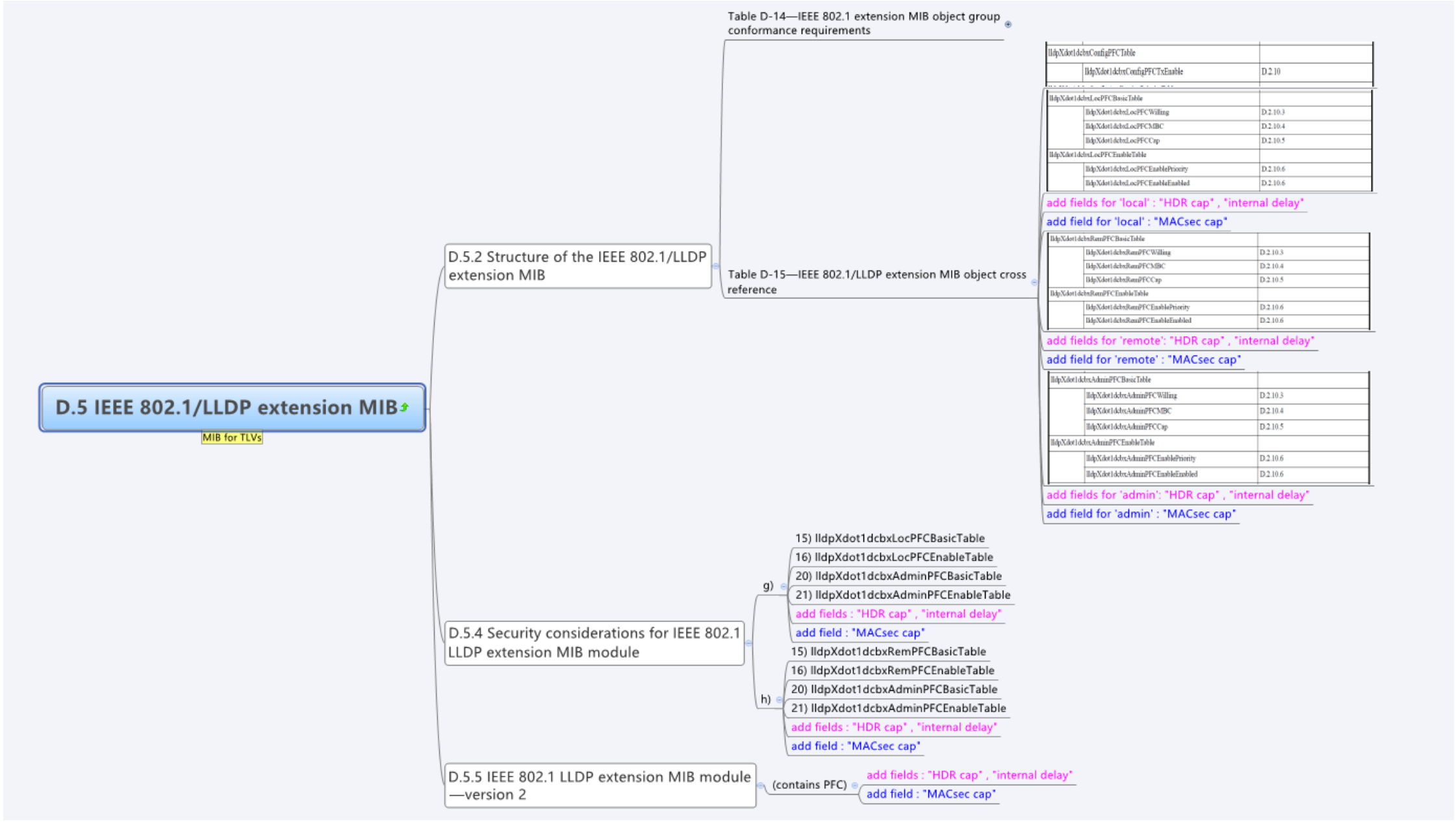


Option 2 is preferred.

Proposed Updates for New(Qdt--MACsec) PFC Management (option 2)

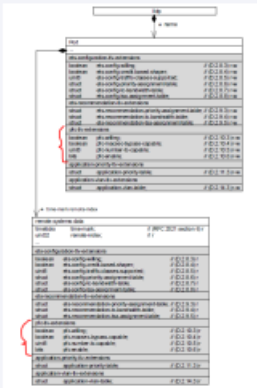


Proposed Updates for New(Qdt--MACsec) PFC Management (option 2)



Proposed Updates for New(Qdt--MACsec) PFC Management (option 2)

- D.6 IEEE 802.1/LLDP extension YANG
  - YANG for TLVs
  - D.6.2 Security considerations
    - D.6.2.3 Security considerations of the ieee802-dot1q-lldp-dcbx-tlv YANG module
      - /lldp/port/pfc-tlv-extension/pfc-willing
      - /lldp/port/pfc-tlv-extension/pfc-macsec-bypass-capable
      - /lldp/port/pfc-tlv-extension/pfc-number-tc-capable
      - /lldp/port/pfc-tlv-extension/pfc-enable
      - add fields : "HDR cap" , "internal delay"
      - add field : "MACsec cap"
  - D.6.3 IEEE 802.1 Organizationally Specific TLV YANG data models
    - D.6.3.3 IEEE 802.1/LLDP extension dcbxSet YANG model
      - Figure D-6—dcbxSet TLV modelIlldpname



      - add fields : "HDR cap" , "internal delay"
      - add field : "MACsec cap"
  - D.6.4 Structure of the IEEE 802.1/LLDP extension YANG model
    - Table D-15—Structure of the YANG modules

ieee802-dot1q-lldp-dcbx-tlv	D.6.3	DCBX set of IEEE 802.1 Organizationally Specific TLVs

  - D.6.5 Definition of the IEEE 802.1/LLDP extension YANG modules
    - D.6.5.3 Data scheme definition for the ieee802-dot1q-lldp-dcbx-tlv YANG module
      - augment /lldp:lldp/lldp:port:
        - +--rw pfc-tlv-extension
          - | +--rw willing? boolean
          - | +--rw macsec-bypass-capable? boolean
          - | +--rw number-tc-capable? dot1q-types:num-traffic-class-type
          - | +--rw enable? bits
        - add fields : "HDR cap" , "internal delay"
        - add field : "MACsec cap"
      - augment /lldp:lldp/lldp:port/lldp:remote-systems-data:
        - +--ro pfc-tlv-extension
          - | +--ro willing? boolean
          - | +--ro macsec-bypass-capable? boolean
          - | +--ro number-tc-capable? dot1q-types:num-traffic-class-type
          - | +--ro enable? bits
        - add fields : "HDR cap" , "internal delay"
        - add field : "MACsec cap"
  - D.6.6 IEEE 802.1/LLDP extension YANG modules

Backup Slides

Topic 1: PFC interface stack diagram (1/2)

Figure 36-1 in 802.1Q seems to illustrate a simple PFC interface stack. However, the figure has issues.

- It draws the wrong boundary between 802.1 and 802.3.
- It does not reflect the correct logic of the PFC mechanism.
  - It shows communication between transmission queues to the same LAN, but PFC asserts back pressure from a reception queue on the LAN to a transmission queue. Or propagates back through a bridge through the reception queue to the transmission queue.

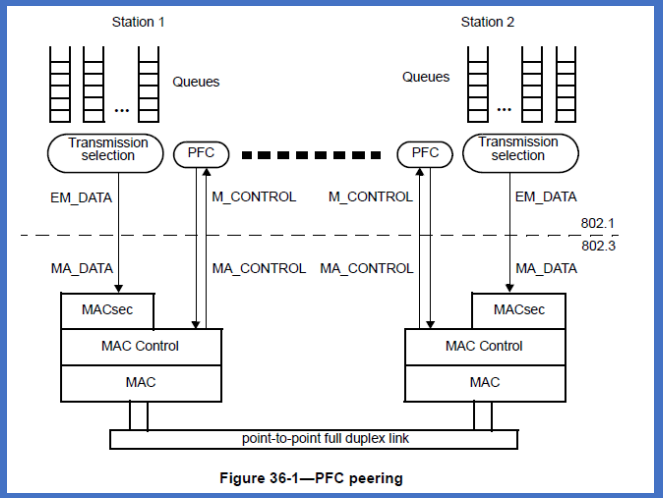
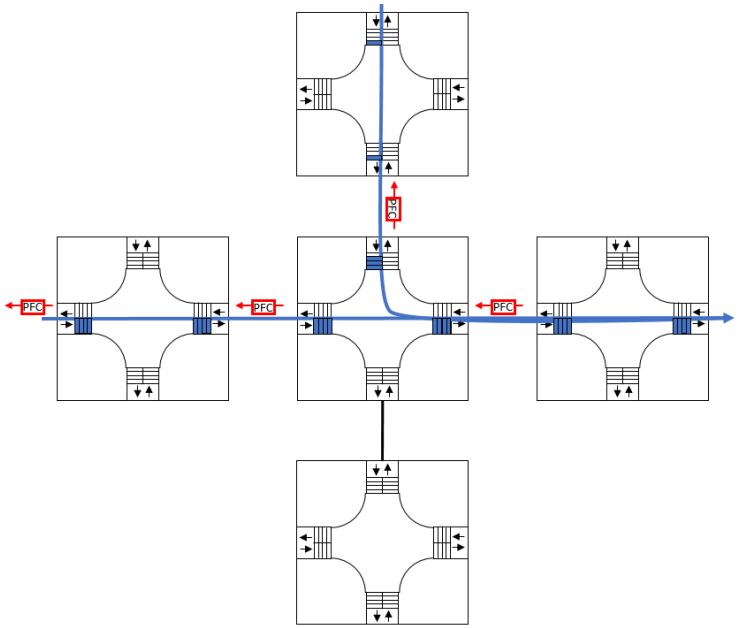


Figure 36-1—PFC peering

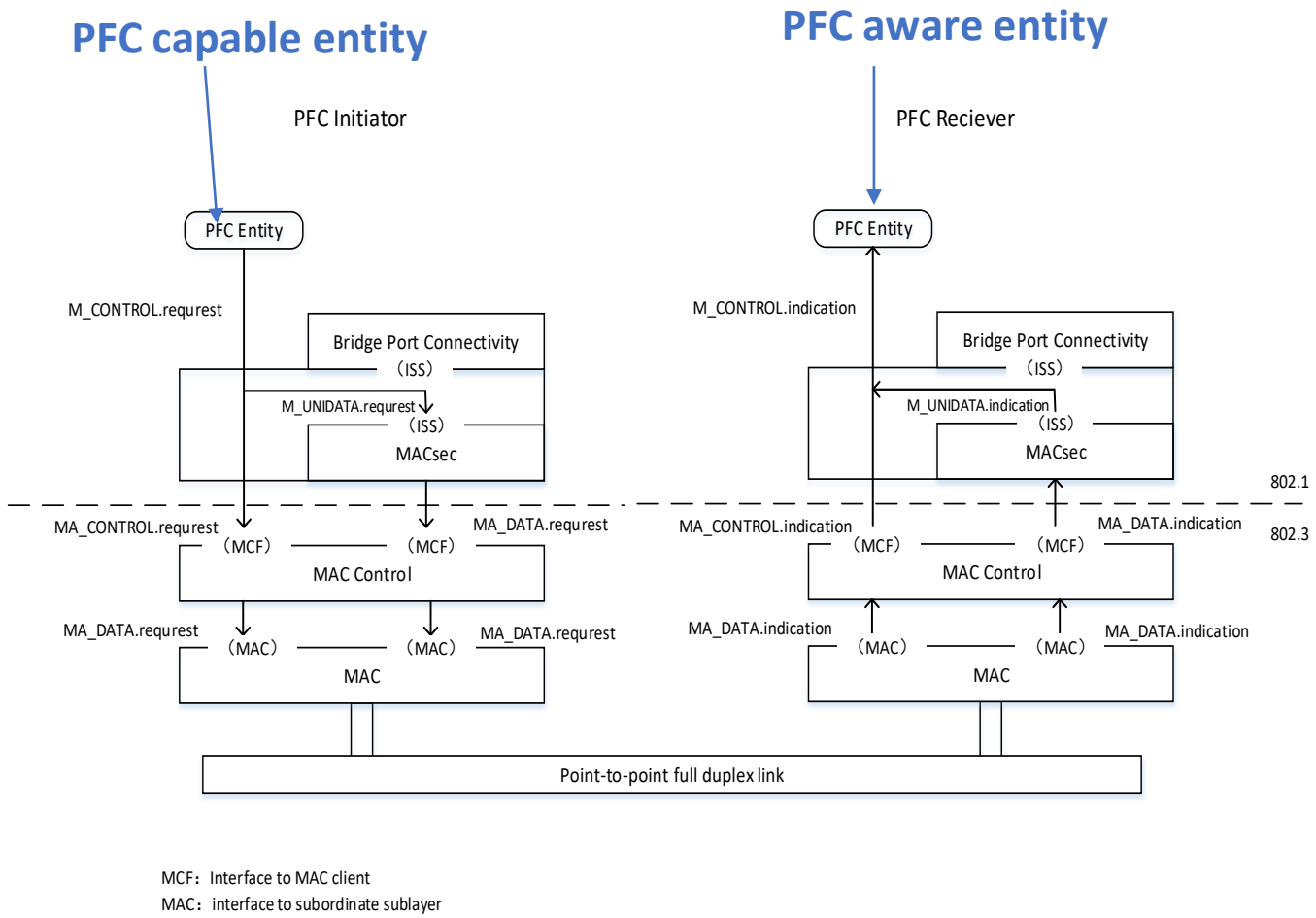
Explanation/Solution:

- Add a new figure showing how PFC propagate hop by hop across the network.
  - PFC pause frame is initiated when ingress port receiving queue is above headroom threshold.
  - Pause frame stops upstream port egress transmit queue.
  - The pause on the port egress transmit queue impacts different port ingress receive queues of the same switch. This is internal backpressure.
    - Internal backpressure is implementation dependent.
  - Higher layer entities (e.g. spanning tree) have no direct interaction with the PFC entity. When higher layer entity frames are put in a PFC enabled queue it may be paused by PFC.
    - Most likely, higher layer entity frames are put into a high priority queue which does not apply PFC.
- Add informative text
  - Describe bridge internal backpressure which is important in PFC propagation, but implementation dependent.
  - Describe higher layer entity relationship with PFC entity.



Topic 1: PFC interface stack diagram (2/2)

- Redraw figure 36-1, still focus on PFC peering.
  - 802.1/802.3 boundary is between MACsec and MAC control.
  - Clearly distinct reception queue and transmission queue on each peer.
  - Add MACsec protection on PFC into the figure



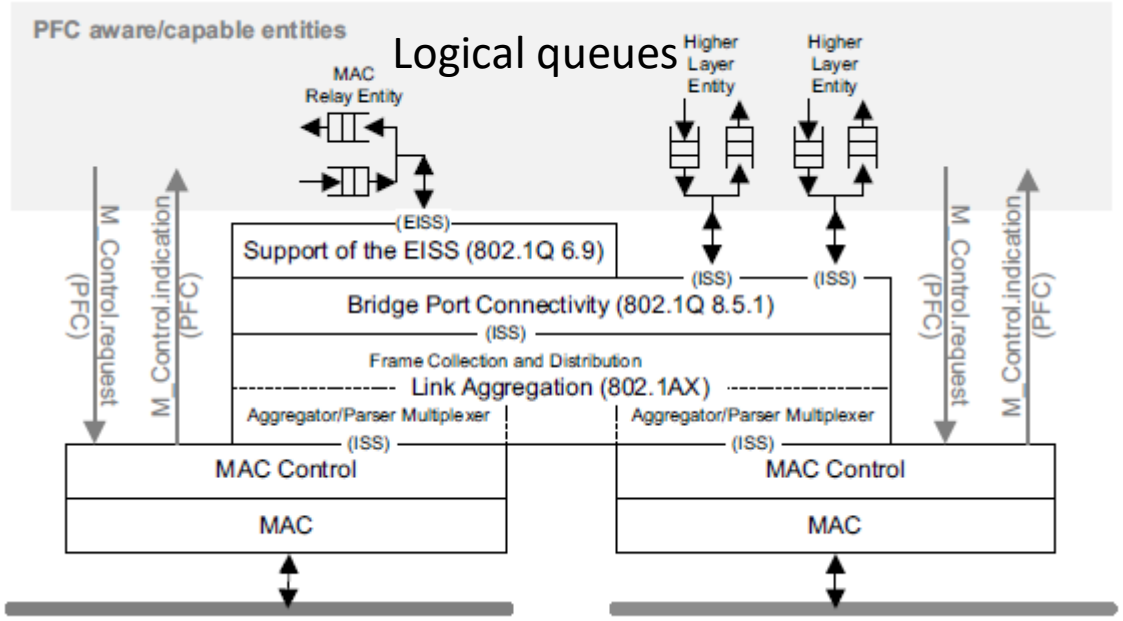
MCF：Interface to MAC client
MAC：interface to subordinate sublayer

Topic 2: PFC and link aggregation

How does PFC function when the link is an aggregated link? Do we pause each physical queue independently?

Explanation:

- 802.1 has no clear description how PFC works together with link aggregation.
- Implementations typically assert PFC on a single physical link, not the logical link.
- .1Q clause 36.1 specifies "PFC is a function defined only for a pair of full duplex MACs (e.g., IEEE 802.3 MACs operating in point-to-point full-duplex mode) connected by one point-to-point link"
- Figure on the right implies the queue on the logical port, is not aware of PFC status of individual physical ports.



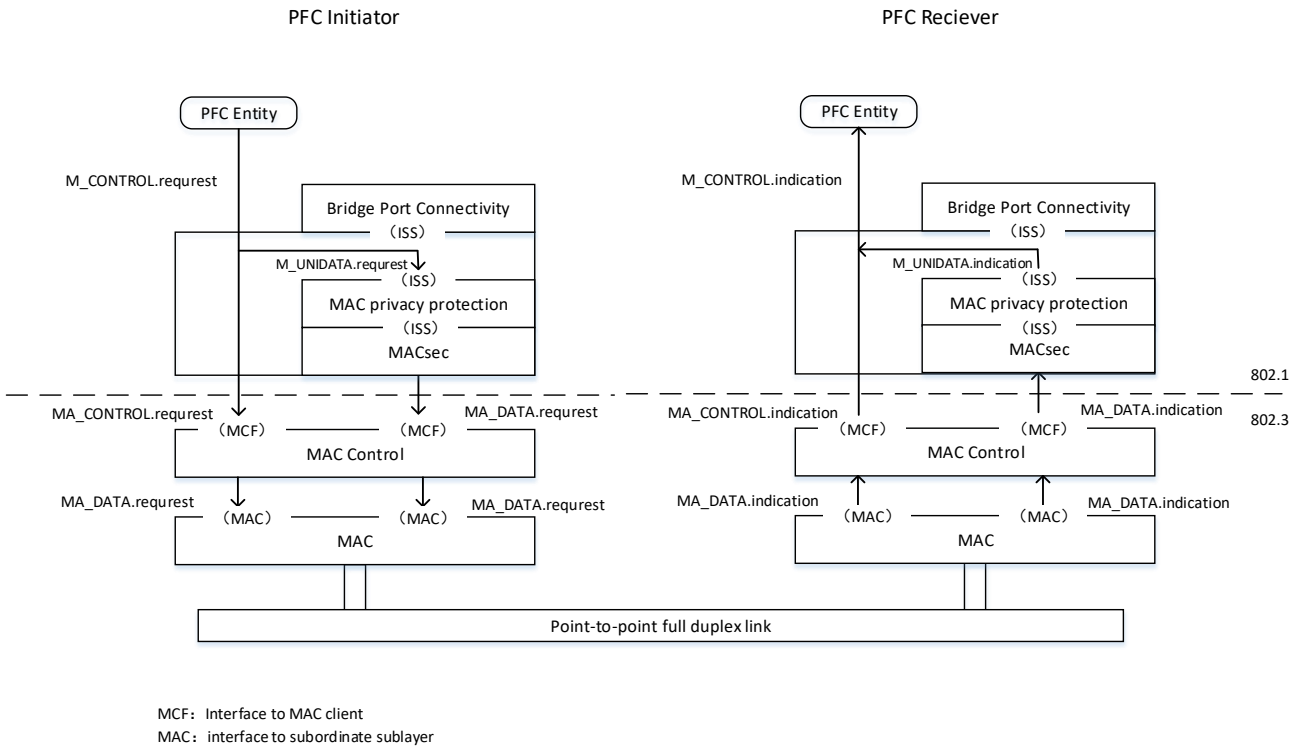
Proposal: add PFC and link aggregation relationship contents in annex, describing pause one link leading to pause all aggr links

Topic 3: MAC privacy protection(802.1AEdk) on PFC

MACsec protects PFC payload, but it is still possible for an attacker to observe the pattern of PFC frames (transmission frequency, packet size etc.) and obtain privacy information. Important in high security cases (e.g. government, financial). Do we need more secured way to protect PFC?

Explanation/Solution:

- Using MAC privacy protection on PFC has Pros & Cons.
  - Pros: Protect privacy information, more secured.
  - Cons: Introduce extra delays for transmission, hard to get headroom, may require a larger buffer as headroom; Privacy channel will tunnel PFC to remote and possibly multiple destinations.
- Solution: PFC stays above MAC Privacy protect layer
  - By default, PFC passes through the layer
  - If PrY is enable for PFC, describe the limitation.



Topic 4: Where to specify PFC shim?

In previous contributions, the shim (used for mapping MAC control primitives to MAC service primitives) is proposed to be specified in .1Q clause 6.7 “Support of the MAC Service” . Perhaps this is not the proper place, otherwise most of 802.1Q (PBNs, PBBNs, CFM, ...) would have been in 6.7 together with 802.1AX, 802.1AE.

Explanation/Solution:

- CFM adds a new clause to specify the shim.
  - “CFM Entities (Clause 19) are specified as shims that make use of and provide the ISS or EISS (IEEE Std 802.1AC, 6.8, 6.17) at SAPs within the network. ”
  - **“19. CFM entity operation ”**
- For PFC shim, propose to add a new subclause under clause 36.