

802.1CB-2017-Rev

Draft 0.1, first TG ballot, editor's report

Christophe Mangin

2025/9/16

MITSUBISHI ELECTRIC R&D CENTRE EUROPE

- P802.1CB-2017-Rev rolls up :
 - 802.1CBcv-2021 : MIB and YANG for CB
 - 802.1CBdb-2021 : extended Stream identification
 - 802.1CB/Cor1
- ... and implements
 - Maintenance item #343 : Handling of Stream_handle in Active Stream Identification, clarifications to the FRER "layered" model

- 11 respondents
 - Approve: 3
 - Disapprove: 3
 - Abstain: 5
- 26 comments from 4 commenters
- Most comments are editorial or “editorial-related technical” and are proposed not to be discussed
 - Outdated normative/standard references, use of trademark logo in standard denomination
 - Page layout, typos, broken cross-references, spelling, grammar, ...
- 4 technical comments on
 - YANG types used to represent Stream identification functions parameters
 - 4, 5, 7
 - Stream identification function managed objects definition
 - 6

- Technical

- #4 : Cl. 9.1.5.1 *tsnCpelpIdDestMac*

- MAC address wildcard (all 0s) for Dest MAC in IP Stream identification function
 - Represented using the mac-address type in YANG, that possibly does not allow the "all 0" value

- **Proposed reject**

- The regex used to specify the mac-address type includes the "all-0" value

```
typedef mac-address {  
    type string {  
        pattern '[0-9a-fA-F]{2}(:[0-9a-fA-F]{2}){5}';  
    }  
}
```

- #5 : Cl. 9.1.6.1 *tsnCpeMmIdDestMacMask*

- mac-address YANG type used to represent the MAC address mask in the mask-and-match Stream identification function
 - Hex string representation is not adapted to specify a 48-bit mask

- **Proposed reject**

- Already had the discussion at the time of CBdb development
 - The mac-address type provides a human-readable format for the mask values

- Technical

- #7 : Cl. 9.1.5.4 *tsnCpelpdlpSource*

- Wildcard (all 0s) for Source IP in IP Stream identification function

- Represented using the ip-address type in YANG, that does not allow an "all 0" value

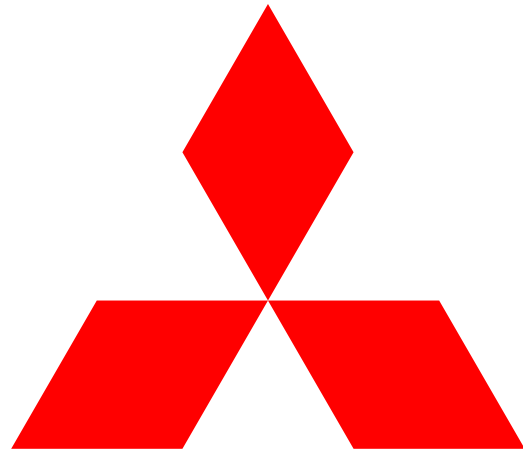
```
typedef ipv4-address {  
    type string {  
        pattern  
            '([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5])\.){3}'  
        + '([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5])'  
        + '(%[\p{N}\p{L}]+)?';  
    }  
}
```

- **Discuss**

- Do we need a new type ?

- Technical

- #6 : Cl. 9.1.5.3 *tsnCpeIpdVlan*
- CB-2017 reads : "a value of 0 indicates that the frame is not to have a VLAN tag".
 - There is already another managed object indicating the presence/absence of the VLAN tag
 - *tsnCpeIpdTagged* (Cl. 9.1.5.2)
 - is this intended to split "priority" into "priority tagged" and "untagged" ?
 - **Discuss**
 - The other stream id functions (null, source MAC + vlan, active destination MAC + vlan) ignore the vlan when the "tsnCpeXXvlan" value equals 0.
 - CB-2017 : "A value of 0 indicates that the vlan_identifier parameter is ignored on EISS indication primitives"
 - The editor assumes that the same should apply to the IP stream id function.
 - Proposal : replace the sentence by "A value of 0 indicates that the vlan_identifier parameter is ignored on EISS indication primitives"



**MITSUBISHI
ELECTRIC**

Changes for the Better