Extensions on the TSN UNI traffic specification

Konstantinos Alexandris, Lihao Chen, Tongtong Wang
Huawei Technologies
Objective

- TSN UNI TSpec to handle TokenBucket traffic model [1,2]
  - Need for a standard way to receive stream requirements
  - Only basic and TimeAware Tspec elements are included in 802.1Q-2022
- Enable TSN UNI to support the TokenBucket traffic model in conjunction with centralized configuration [*]
  - End-station/CUC needs to send the TokenBucket Tspec via TSN UNI
  - Current projects and standards do not define specific YANG models
  - Centralized configuration involves CNC assistance support
  - To be complementary to RAP (P802.1Qdd) that uses distributed configuration

[*] Both fully centralized and centralized network/distributed user configuration models

Proposal (1/2)

**Tspec** definition is not **complete**: Addition of parameters for the TokenBucket model

**Sub-clauses to be extended:**

- **46.2.3.5**: Extension of the existing Tspec incorporating the relevant parameters (**currently missing**)

<table>
<thead>
<tr>
<th>Name</th>
<th>Data type</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaxFrameSize</td>
<td>uint16</td>
<td>46.2.3.5.3</td>
</tr>
<tr>
<td>MinFrameSize</td>
<td>uint16</td>
<td>46.2.3.5.8</td>
</tr>
<tr>
<td>CommittedInformationRate</td>
<td>uint64</td>
<td>46.2.3.5.9</td>
</tr>
<tr>
<td>CommittedBurstSize</td>
<td>uint32</td>
<td>46.2.3.5.10</td>
</tr>
</tbody>
</table>

- **46.2.3.5.8 - 10**: Explanatory text related to Table 46-10 parameters to be added (**currently missing**)
Proposal (2/2)

Existing YANG models do not support centralized configuration including the TokenBucket Tspec

Sub-clauses to be extended:

- **48.5.23**: Extension of the respective YANG schema tree related to the `ieee802-dot1q-tns-config-uni` YANG module *(currently missing)*
  - `traffic-specification [3]`: To include TokenBucket TLV parameters [TokenBucket Tspec]

- **48.6.3**: Extension of the `ieee802-dot1q-tns-types` YANG module *(currently missing)*
  - `container token-bucket`: To be added under `container traffic-specification` including the relevant parameters as leaf statement:
    - `min-frame-size`, `committed-information-rate`, `committed-burst-size`

[3] [https://1.ieee802.org/tsn/802-1qdi/](https://1.ieee802.org/tsn/802-1qdi/)
Extensions on TSN UNI YANG schema tree – Proposal #1

Updates on schema for the ieee802-dot1q-tsn-config-uni YANG module [*]

module: ieee802-dot1q-tsn-config-uni

```
  +--rw traffic-specification
  |   +--rw (tspec-options)
  |     |   +--: (tspec-option-1)
  |     |     |   +--rw interval
  |     |     |     |   +--rw numerator uint32
  |     |     |     |   +--rw denominator uint32
  |     |     |   +--rw max-frames-per-interval uint16
  |     |     |     |   +--rw time-aware!
  |     |     |     |     |   +--rw earliest-transmit-offset uint32
  |     |     |     |     |   +--rw latest-transmit-offset uint32
  |     |     |   +--rw jitter uint32
  |     |     |     |   +--: (tspec-option-2)
  |     |     |   +--rw token-bucket
  |     |     |     |   +--rw committed-information-rate uint64
  |     |     |     |   +--rw committed-burst-size uint32
  |     |     |     |   +--rw min-frame-size uint16
  |     |     |     |   +--rw max-frame-size uint16
  |     |     |   +--rw transmission-selection uint8
```

TsSpec Option 1 “xor” 2

- Isolation of TsSpec models (token bucket and others) in case of parameters dependency.

- What about introducing “choice” nodes in the respective YANG module?

- The objective is to exclude any correlation while extending the existing YANG module.

[*] Similar changes apply to ieee802-dot1q-tsn-types YANG module.
Updates on schema for the ieee802-dot1q-tsn-config-uni YANG module

```yang
module: ieee802-dot1q-tsn-config-uni
 :
   +++-rw traffic-specification
     | | +++-rw interval
     | | | | +++-rw numerator uint32
     | | | | +++-rw denominator uint32
     | | +++-rw max-frames-per-interval uint16
     | |+++rw time-aware!
     | | | | +++-rw earliest-transmit-offset uint32
     | | | | +++-rw latest-transmit-offset uint32
     | | | | +++-rw jitter uint32
     | |++++-rw token-bucket!
     | | | | +++-rw committed-information-rate uint64
     | | | | +++-rw committed-burst-size uint32
     | | | | +++-rw min-frame-size uint16
     | | | | +++-rw max-frame-size uint16
     | | | | +++-rw transmission-selection uint8
```

- Other way is to introduce the token bucket model as a “presence container” in the YANG module [4].

- Following that methodology, redundancy may be introduced.

**Open discussion:** What is the best way to implement it?

The way TimeAware Tspec is conveyed from user to CNC has already been defined.

We should follow the same methodology with TokenBucket Tspec.
Conclusion

Need to extend TSN UNI

1. Motion for new PAR development in 2023 July Plenary meeting.

2. Any questions?
Thank you.