Extensions on the traffic specification and status for TSN UNI

Konstantinos Alexandris, Lihao Chen, Tongtong Wang
Huawei Technologies
Introduction

- **Deviation** in traffic characteristics sourced by different types of applications has been limitedly considered in traffic specification (TSpec) TLVs [1].

- **New streams** arrival circumvents the need for dynamic (online) scheduling mechanisms.

- **Resource allocation** techniques need to be revised to cover network traffic variability.

- Whether current Tspec TLVs meet the needs for static allocation, there is no approach that adds flexibility for dynamic network resource management.

- Revisit TSN Tspec is urged towards enhancements in flexible traffic engineering.

- In such scenarios, an adaptive **QoS** admission control scheme is required.

Background in IEEE 802.1 TSN

- MSRP is limited to basic traffic parameterization in Talkers
  REGISTER_STREAM.request Tspec [1]:
  - MaxFramesPerInterval
  - MaxFrameSize

- Optional Tspec: TimeAware TLV covers the case of Scheduling traffic (ST) based on centralized scheduling in Qcc [1]:
  - EarliestTransmitOffset
  - LatestTransmitOffset

- More advanced [2] compared to MSRP traffic parameterization in Talkers
  ANNOUNCE_STREAM.request Tspec (i.e., Token Bucket Tspec sub-TLV):
  - Minimum Transmitted Frame Size
  - Committed Information Rate
  - Committed Burst Size

Extensions of the current UNI traffic specification (1/2)

Static vs Dynamic resource allocation: Could we further broaden the TSN UNI scope?

- **Adaptability** of the network behavior is urged, where new streams are introduced on the fly.
- **Fairness** in resource reservation has to be guaranteed in such dynamic environment.
- **TSN UNI** capabilities needs to cover dynamic resource allocation under network performance variability.
- Talkers could ask for a range of resources **up to a maximum value**, but still with less resources **bounded by a minimum value** could sustain the desired QoS.
- **Feasibility** in admission of streams to be guaranteed with flexible talkers QoS.
- Proposed **traffic parameterization** shall be kept simple and build on top of the current TSpec configuration.
Objective: Flexibility in allocating network resources. More streams to be admitted dynamically!

- Would be useful to add data rates and burst size with ranges in the Tspec of UNI?
  + Minimum Information Rate (MIR)
  + Minimum Burst Size (MBS)

- Flexible & agile network management to return a target value of information rate $R(t)$ and burst size $S(t)$ tailored to the talkers announced range of values and the availability of resources:

  $$\text{MIR} \leq R(t) \leq \text{CIR}$$
  $$\text{MBS} \leq S(t) \leq \text{CBS}$$

- Either talkers send traffic with the limitation of the target value of $R(t)$ and $S(t)$, or the target value of $R(t)$ and $S(t)$ is used for the traffic shaping of talkers.
Talker to Network – New stream request (1/2)

- **Talkers** signal to the network via UNI the stream Tspec min/max values, i.e., a pre-defined range.
- **Admission** control and **resource allocation** for the stream request is performed by the network.
- The **target values** to be returned are **discrete** and chosen by a concrete set within the min/max values.
- A **withdraw stream** notification shall be sent in case a stream **cannot be admitted** in the network.
Talker to Network – New stream request (2/2)

1. New request:
   - A new stream reservation request is received by the network carrying the Tspec min/max values.

2. Admission control:
   - If the network resources suffice to satisfy the announced Tspec at its max value, then such target value is returned.
   - Elsewhere, adaptation of the target value within a range is performed for the streams that is sustained starting from the new one.

3. Feasibility check:
   - If a feasible solution is reached, the new stream is admitted and configured (Stream configuration).
   - Elsewhere, the stream is withdrawn and a notification is sent to the talker (Withdraw stream notification).
Example - New stream request (1/2)

Talker application requirements [3]

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Target rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>360p</td>
<td>( R_1 = 0.7 \text{ Mbps} )</td>
</tr>
<tr>
<td>480p</td>
<td>( R_2 = 1.1 \text{ Mbps} )</td>
</tr>
<tr>
<td>720p</td>
<td>( R_3 = 2.5 \text{ Mbps} )</td>
</tr>
<tr>
<td>1080p</td>
<td>( R_4 = 5 \text{ Mbps} )</td>
</tr>
<tr>
<td>4K (2160p)</td>
<td>( R_5 = 20 \text{ Mbps} )</td>
</tr>
<tr>
<td>8K (4320p)</td>
<td>( R_6 = 100 \text{ Mbps} )</td>
</tr>
</tbody>
</table>

Talker 5
- MIR = 20 Mbps, CIR = 100 Mbps
- Target Rate = 20 Mbps

Capacity bottleneck 100 Mbps was reached!

Talker 1-4
- MIR = 2.5 Mbps, CIR = 20 Mbps
- Target Rate = 20 Mbps

Existing streams
- Talker 1
- Talker 2
- Talker 3
- Talker 4

New stream
- Talker 5

Example - New stream request (2/2)

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Target rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>360p</td>
<td>R_1 = 0.7 Mbps</td>
</tr>
<tr>
<td>480p</td>
<td>R_2 = 1.1 Mbps</td>
</tr>
<tr>
<td>720p</td>
<td>R_3 = 2.5 Mbps</td>
</tr>
<tr>
<td>1080p</td>
<td>R_4 = 5 Mbps</td>
</tr>
<tr>
<td>4K (2160p)</td>
<td>R_5 = 20 Mbps</td>
</tr>
<tr>
<td>8K (4320p)</td>
<td>R_6 = 100 Mbps</td>
</tr>
</tbody>
</table>

Live streaming

Network

Target Rate = 20 Mbps

Talker 3-4
MIR = 2.5 Mbps, ✓ CIR = 20 Mbps
Target Rate = 20 Mbps

Talker 5-6
✓ MIR = 20 Mbps, × CIR = 100 Mbps
Target Rate = 20 Mbps

Total allocation = 90 Mbps

Capacity bottleneck 100 Mbps

Existing streams

New stream

Talker application requirements [3]

Where are we now in YANG modules?

IEEE802-Dot1Dj-TSN-Config-UNI YANG module (Section 48.5.13 in Qdj/D0.3)

```yaml
+++rw traffic-specification
   +++rw time-aware!
      +++rw earliest-transmit-offset? uint32
      +++rw latest-transmit-offset? uint32
   +++rw jitter? uint32

+++ro interface-configuration
   +++ro interface-list* [mac-address interface-name]
      +++ro mac-address string
      +++ro interface-name string
      +++ro config-list* [index]
      +++ro index uint32
      +++ro (config-value)?
      +++:(time-aware-offset)
         +++ro time-aware-offset? uint32
```

Ref. to Section 46.2.5.3.5 TimeAwareOffset in IEEE Std 802.1Qcc-2018

### 46.2.5.3.5 TimeAwareOffset

If the `TspecTimeAware` group is present in the `TrafficSpecification` group (46.2.3.5) of the Talker, this configuration value shall be provided by the network to the Talker.

If the `TspecTimeAware` group is not present in the `TrafficSpecification` group (46.2.3.5) of the Talker, this configuration value shall not be provided by the network.

This configuration value shall not be provided to Listeners as it is not applicable.

TimeAwareOffset specifies the offset that the Talker shall use for transmit. The network returns a value between EarliestTransmitOffset and LatestTransmitOffset of the Talker’s TrafficSpecification. The value is expressed as nanoseconds after the start of the Talker’s Interval. The data type is `uint32`.

What about extending those YANG modules?

Similar YANG definitions apply to CIR/MIR, CBS/MBS

```yaml
+++rw traffic-specification
   ...  
      +++rw rate-burst!
         +++rw cir uint16
         +++rw mir uint16
         +++rw cbs uint16
         +++rw mbs uint16
   ...

+++ro interface-configuration
   ...  
      +++ro cir-chosen uint16
      +++ro cbs-chosen uint16
```

Recommended text to be added:

If the `rate-burst` group is present in the `TrafficSpecification` group of the Talker, this configuration value shall be provided by the network to the Talker, and the value of "interval" and "max-frames-per-interval" can be neglected.

This configuration value shall not be provided to Listeners as it is not applicable.

cir-chosen and cbs-chosen specifies the Committed Information Rate and the Committed Burst Size of the Talker shall use as a limit for transmit. The network returns a value of cir-chosen between cir and mir, and a value of cbs-chosen between cbs and mbs. The value of cir, mir, and cir-chosen is expressed as bit per second. The value of cbs, mbs, and cbs-chosen is expressed as bit/Byte (?)
Summary – Contribution & Next steps

- **TSN Tspec** has been revisited towards enhancements in network resource management.
- A dynamic scheduling scheme is proposed based on adaptive QoS traffic engineering within a pre-determined range of values.
- Such methodology accelerates computational convergence and provides flexibility in allocating the network resources.
- The presented mechanism can be applicable to any of the TSN network configuration models.

**How to proceed?**

- **Proposal 1:** Current Tspec definition is not complete: An addition of CIR/CBS parameters is urged to cover more use-cases.
  - TSpec can only handle streams reservation for scheduled traffic and not burst one, i.e, time-aware vs rate-burst Talkers.
  - Token Bucket TSpec sub-TLV includes those parameters in IEEE Std P802.1Qdd.
  - Ongoing work in present YANG modules lacks of the aforementioned system characteristics.
- **Proposal 2:** In addition to that, CIR/MIR & CBS/MBS parameters inclusion is suggested as an upcoming extension.
  - Flexibility in dynamic (online) scheduling is given by those parameters that define a range within QoS is sustainable.
  - A new stream arrives, do the current resources suffice? Adjustment of new and existing streams resources improves schedulability.
Thank you.