Urgency based Scheduler

Considerations for Low Latency Reserved Streams

Johannes Specht, University of Duisburg-Essen

Markus Jochim, General Motors Research

Introduction

- Scheduled Traffic is the first choice for time critical ultra low latency control applications

- Nonetheless, scheduled traffic has several disadvantages and restrictions w.r.t. flexibility, planning overhead, etc.

- If more flexibility and less planning overhead is required and/or desired while still getting "acceptable" low latency guarantees, there might be better alternatives

- This slide deck proposes considerations on *Latency Balancing* to address these cases

Goals and <u>**no**</u> Goals of this slide deck

- This slide deck serves as a basis for initial discussions in 802.1 TSN to gather opinions and experiences from 802.1 TSN members.

- The shown concept is far from being well analyzed – several significant aspects have not yet been analyzed / considered.

- Nonetheless, some aspects have been considered and some parts have been simulated.

Content

- **Scheduled and Reserved Traffic**
  Brief recap. and considerations of advantages and limitations
- **Traffic Shaping**
  Capabilities of per Queue/Class shaping on egress ports
- **Urgency based Scheduler**
  Basic approach, resulting questions and issues, considerations
- **Simulations**

SCHEDULED AND RESERVED TRAFFIC

Scheduled Traffic

- Fulfills Ultra Low Latency Requirements of e.g. Automotive Control Applications and …

- … is capable to provide low end to end jitter, even across multiple hops

→***Scheduled Traffic is highly deterministic:***

  – transmission times,
  – reception times,
  – transmission and reception periods,
  – frame size limits, etc. …

… nearly everything can be calculated before transmission!

Scheduled Traffic

… nearly everything *must* be calculated!

→ Nearly everything *must be known* for calculation - for "simple" TDMA this means:

- **Period/Cycle duration**

  Must be fixed per stream, harmonized across multiple streams in the network

- **Phase**

  Must be fixed per stream, harmonized across multiple streams in the network

- **Frame size limitations**

  Must be fixed per stream, harmonized across multiple streams in the network

- **etc., etc. (e.g. bridge delays, worst case clock sync. precision)**

Scheduled Traffic

Further issues with Scheduled Traffic

- It's not a good option when more flexibility is needed, e.g. for non-periodic/asynchronous transmissions
  - *even in engineered Automotive Networks!*
- Based on experiences with FlexRay:
  Scheduling needs coordination of all required information, maybe even across company boundaries (e.g. between OEM and Automotive Suppliers).
  - *This can become a tedious and expensive task…*
- The above problems, plus even more, can be expected with dynamic scheduling during runtime …

→ Unless ultra low latency is required and there are simpler alternatives that still provide sufficient latency guarantees, these alternatives may be used

Reserved Traffic

Reserved Traffic (as already found in AVB Gen. 1) …

- … neither fulfills ultra low latency requirements …
- … nor is it as deterministic as scheduled traffic

Nonetheless, it gives certain per-stream guarantees …

- Bandwidth and maximum latency (although not “ultra low”)

… while being more flexible than scheduled traffic

- Only bandwidth and class/priority per stream required for configuration
- No restrictions/requirements for streams like periodic transmission, harmonization of multiple/all streams, etc.
- Only parts of the network must be known and can change during runtime

Per-stream guarantees are essentially assured by asynchronous per class shaping (CBS):

→ ***Without restricting the flexibility!***

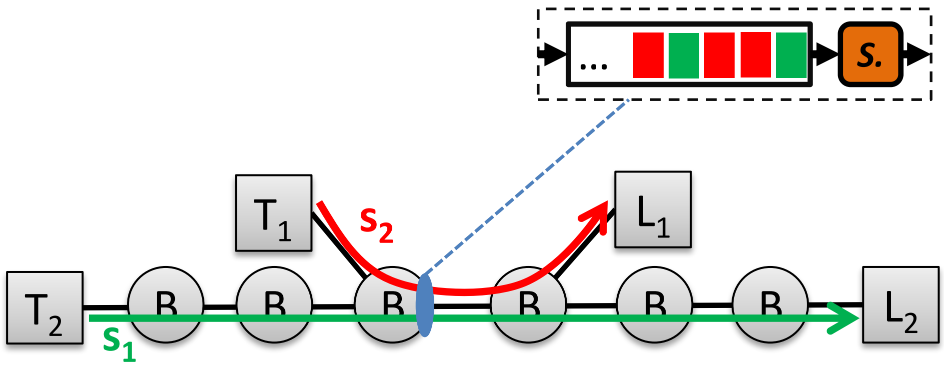
→ ***Without comprehensive information about the whole network!***

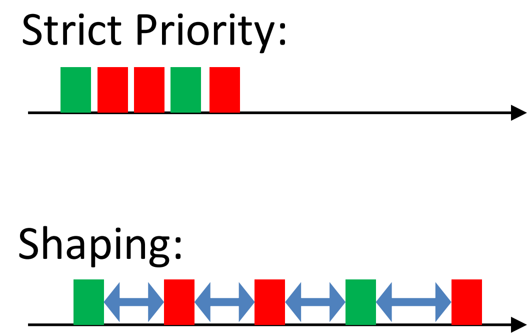
-Capabilities

-Limitations

TRAFFIC SHAPING

What Shapers *won't* do …





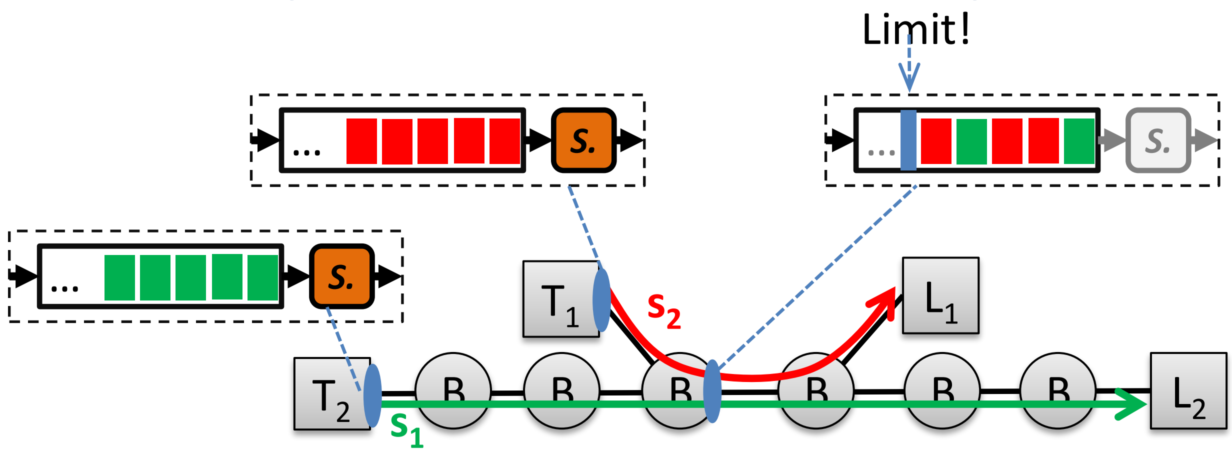
Per queue shapers *won't* reduce the latency of frames in the *local* queue:

- Shaping increases latency – it introduces additional delays between transmissions

  → ***Regardless of the algorithm, shaping can not "magically" reduce the resulting delay of the last frame transmitted from queue!***

- Strict Priority would provide a lower latency, but there are other good reasons for per queue shaping …

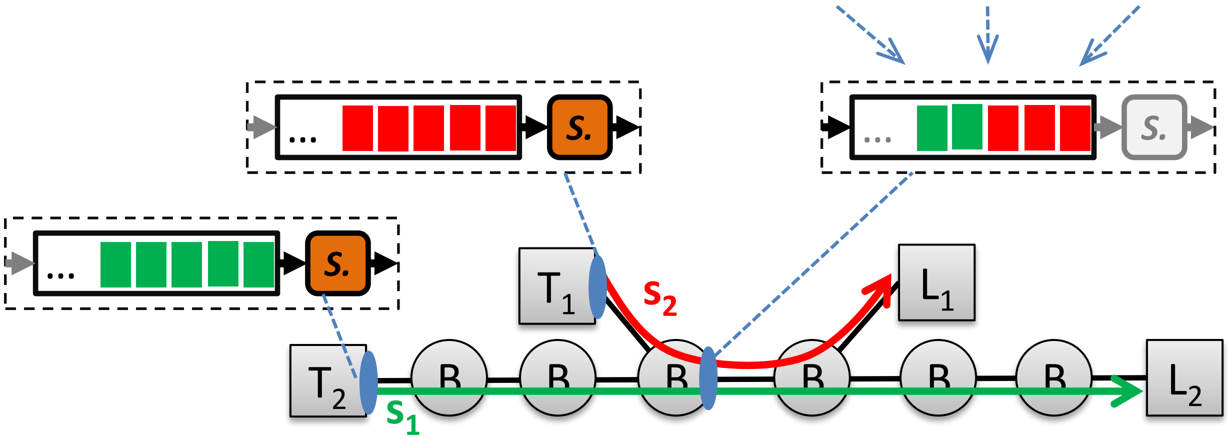
Capabilities of Shapers



Per queue shapers like the Credit Based Shaper (CBS) can:

- Spread frames over time to avoid inacceptably long bursts
- Limit the bandwidth of a certain class and consequently
  - Grant bandwidth to lower priority classes
  - Limit the required egress queue size at consecutive hops

Limitations of Shapers



Recap: To stay flexible, reserved streams should not be synchronized …

- … synchronized shapers could easily limit stream flexibility/introduce burden of scheduled traffic

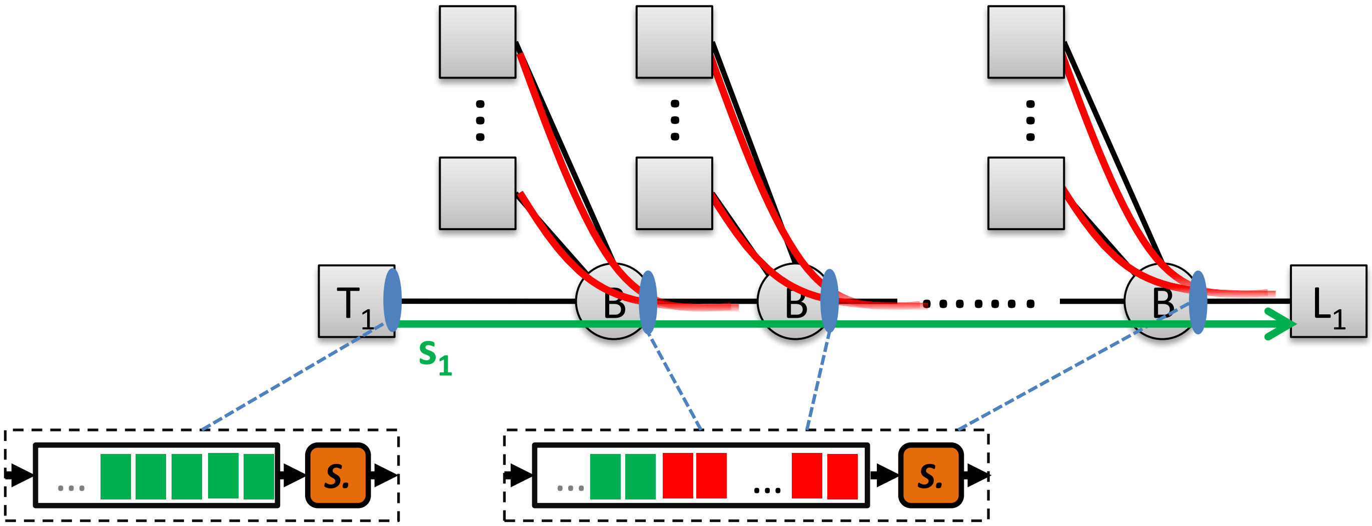
  → ***staying away from synchronized shapers seems to be a good choice!***

- But then streams from different ingress ports can interfere at egress ports, resulting in arbitrary transmission order in one egress queue

  → ***In the worst case, urgent streams are queued behind relaxed streams!***

  (cmp. "fan-in delay" in 802.1Q)

More than 2 talkers



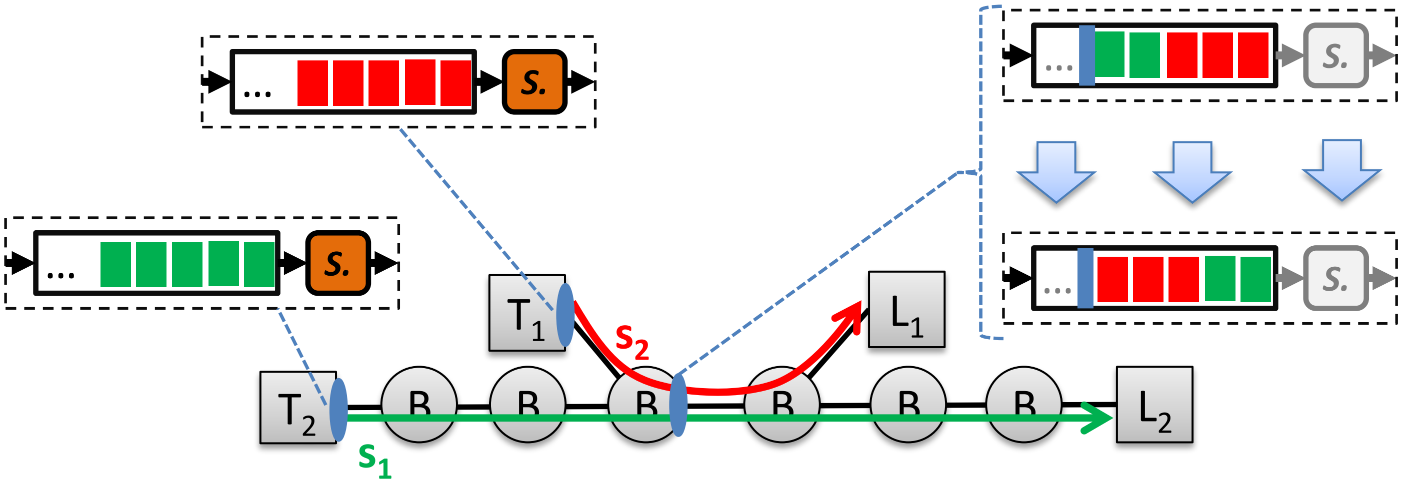
In networks with more than two talkers …

1. Bridges receive streams from a higher number of ingress ports
   → ***The worst case latency of urgent streams increases!***

2. This can happen per bridge per hop on the path of an urgent stream
   → ***The worst case latency of urgent streams increases further!***

URGENCY BASED SCHEDULER

Basic Mechanism



Assumptions:
Only s_1 and s_2 exist and are equal in term of frame size, bandwidth, *maximum acceptable latency*, etc.
s_1: 7 hops s_2: 3 hops

Keep streams asynchronous for maximum flexibility but…

1. Use shaping to limit bandwidth, queue load and grant bandwidth to lower classes

CBS or other algorithms …

AND

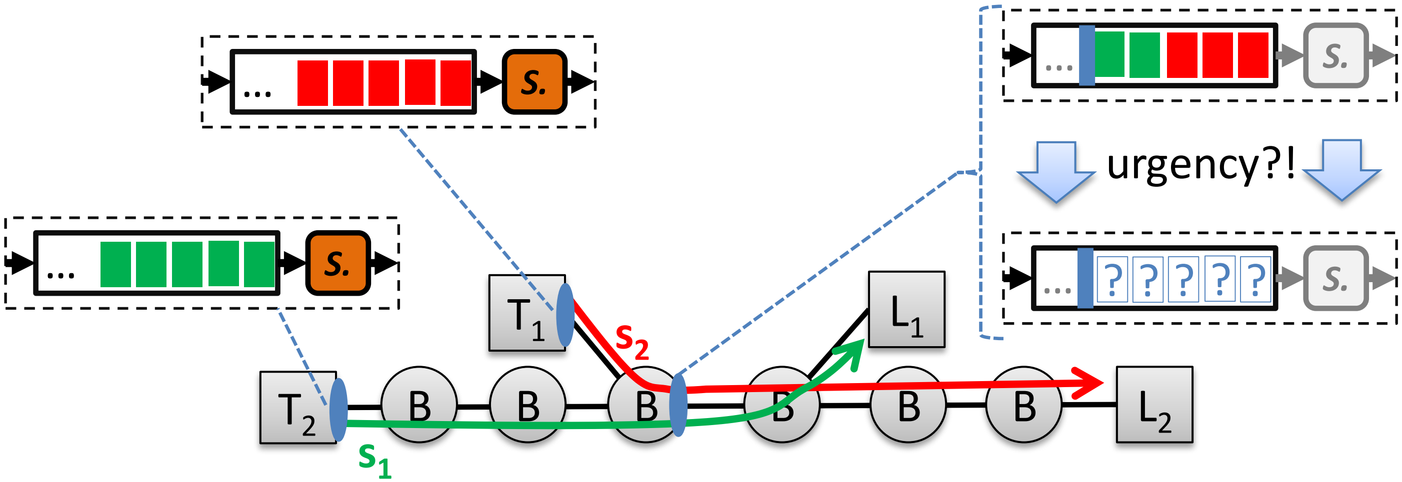
2. Identify and transmit the most urgent frame in queue:

In the above example, sending s_1 frames before sending s_2 frames might be desirable.
(At least until the s_2 frame has been queued for so long, that it becomes more urgent than s_1)

→ ***i.e., introduce "fairness" w.r.t. latency***

→ ***i.e., use suitable traffic scheduling algorithms per queue***
(Very different from static TDMA / "Scheduled Traffic")

What is the most urgent frame?



Assumptions:
Only s_1 and s_2 exist and are equal in term of frame size, bandwidth, *maximum acceptable latency*, etc.
s_1: **5** hops s_2: **5** hops

… minor changes in the previous example make identification of the most urgent frame more complicated:

- <u>Both</u>, s_1 and s_2 now need to be transferred over 5 hops
- Urgency depends on two more criteria:
  1. What latency has already been consumed by a frame in the **past**?
     (could be measured)
  2. What latency will be consumed by a frame in the **future**?
     (can depend on e.g. frame size, remaining hops, other streams, …)

Urgency / Scheduling Strategies

Finding good urgency-criteria / scheduling strategy that can realistically be implemented …

Potential scheduling strategies:

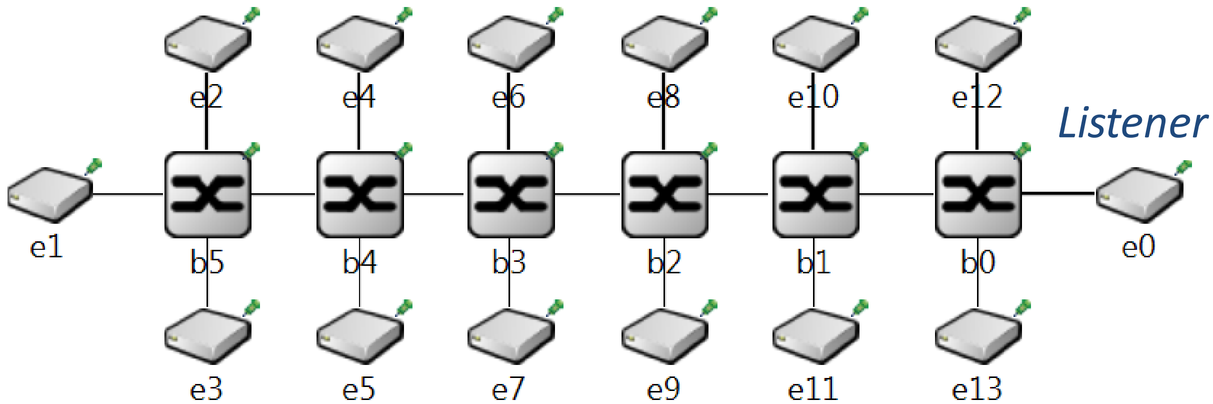
- *Earliest Deadline First*
- *Earliest Due Date*
- *Least Laxity First*
- *…*

Other topics:

- *Compatibility with AVB and other protocols*
- *Low Complexity*
- *Re-use (are there any Traffic Scheduling Algorithms already implemented 802.1 devices)?*

SIMULATIONS

Simulated System



Topology

- Traffic is sent from various end stations to end station *e0* (the rightmost)
- Additional interfering traffic increases link utilization on the path to e0

Algorithms

- Earliest Deadline First (EDF)
- CBS - it's already there and basically shows the desired limiting behavior

Simulated Traffic

Talkers	Max. Latency	Payload	Period
e1,e6,e10	2 ms (Class A)	50 Byte	approx. 125 µs
e3,e5,e9,e13	50 ms (Class B)	400 Byte	approx. 250 µs
e12,e13	none (Best Effort)	1500 Byte	0.5 … 1.5 ms

Traffic sent to e0 (see table)

- AVB-like setup…
- ***Recap: The concept itself basically allows different max. latencies per stream, not only per class (i.e. there could be way more than two different configured maximum latencies)***

Interfering traffic, not sent to e0

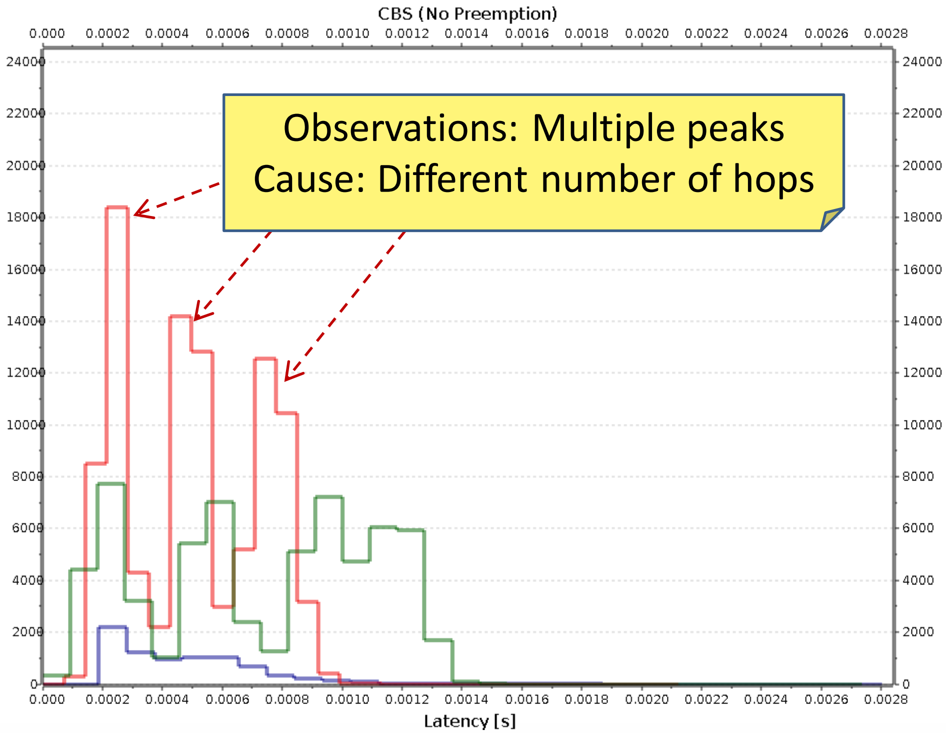
- class A, class B to reach *deltaBandwidth* limits per link (75 %)
- best effort streams to get close to 100% link utilization

Basic Observations

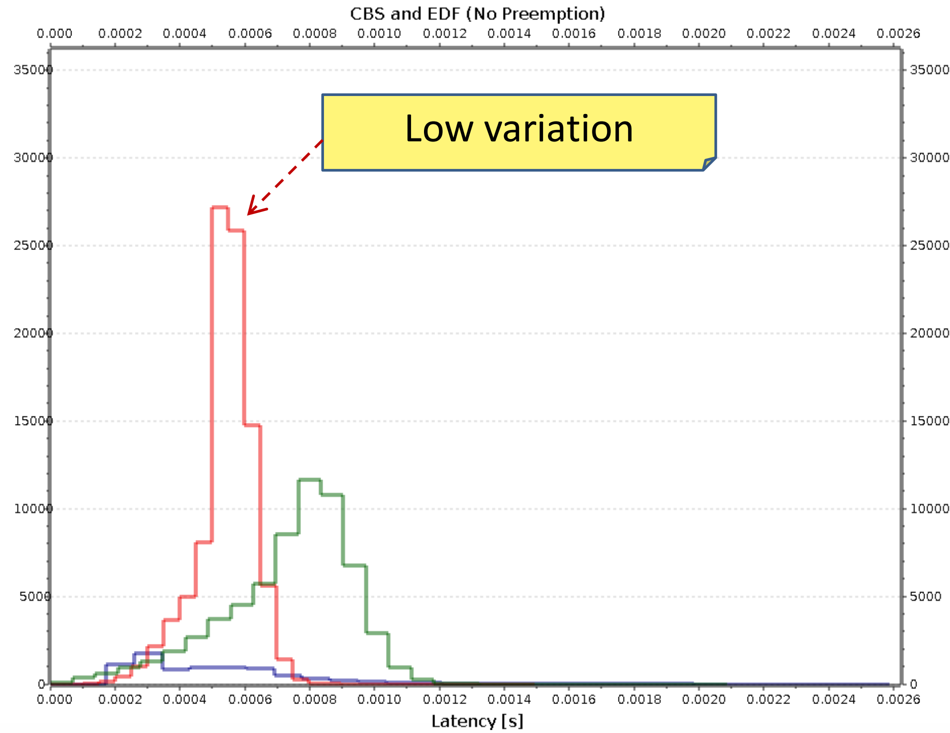
Focus of Observations on this Slide:
Class A
(Similar effect visible for Class B)

Class A (2 ms)
Class B (50 ms)
Best Effort (none)

Without Earliest Deadline First



With Earliest Deadline First



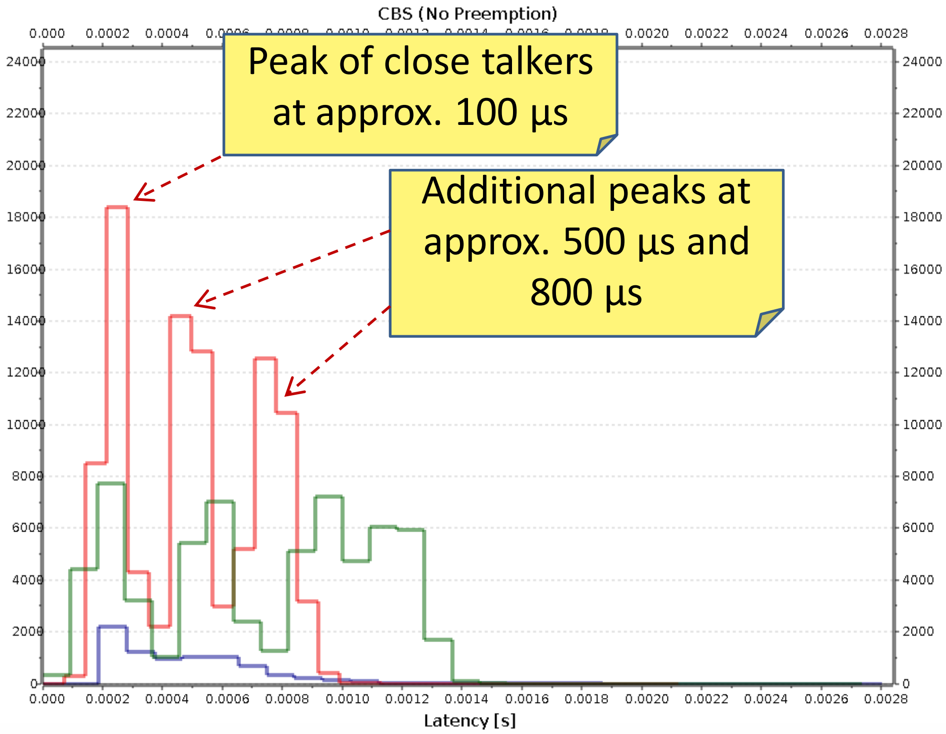
X-Axis: Latency in seconds, Y-Axis: Number of frames.

Basic Observations

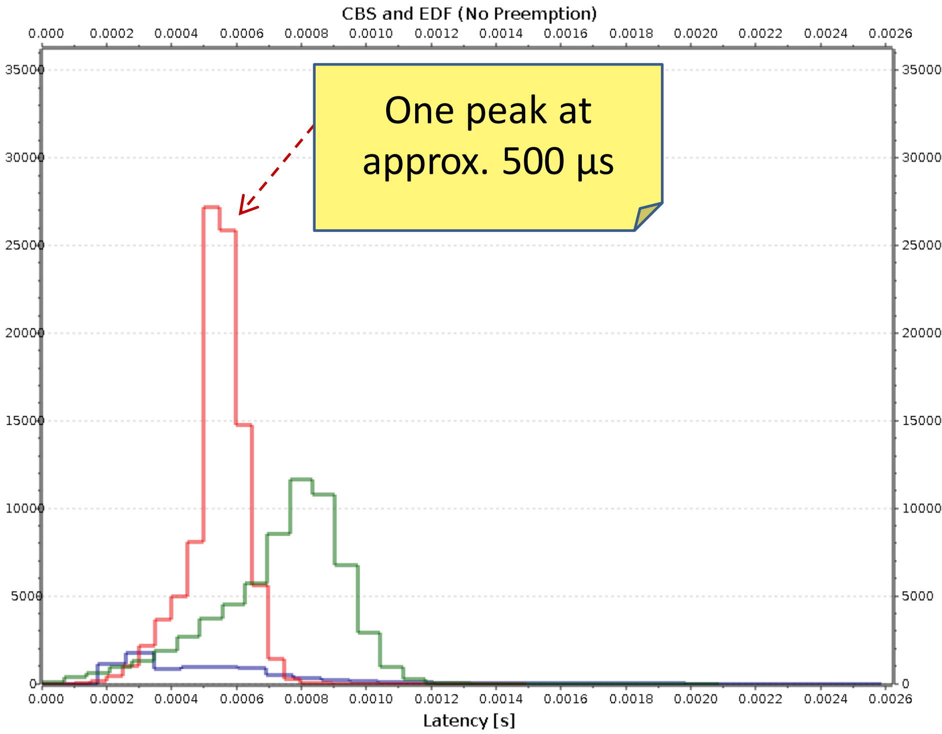
Focus of Observations on this Slide:
Class A only

- Class A (2 ms)
- Class B (50 ms)
- Best Effort (none)

Without Earliest Deadline First



With Earliest Deadline First

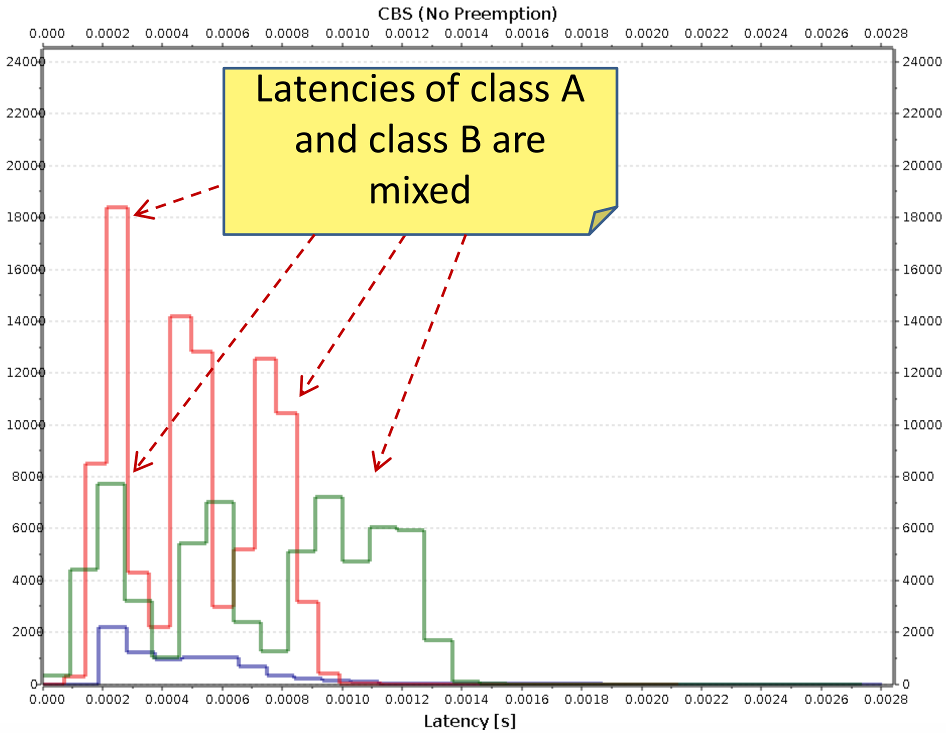


Basic Observations

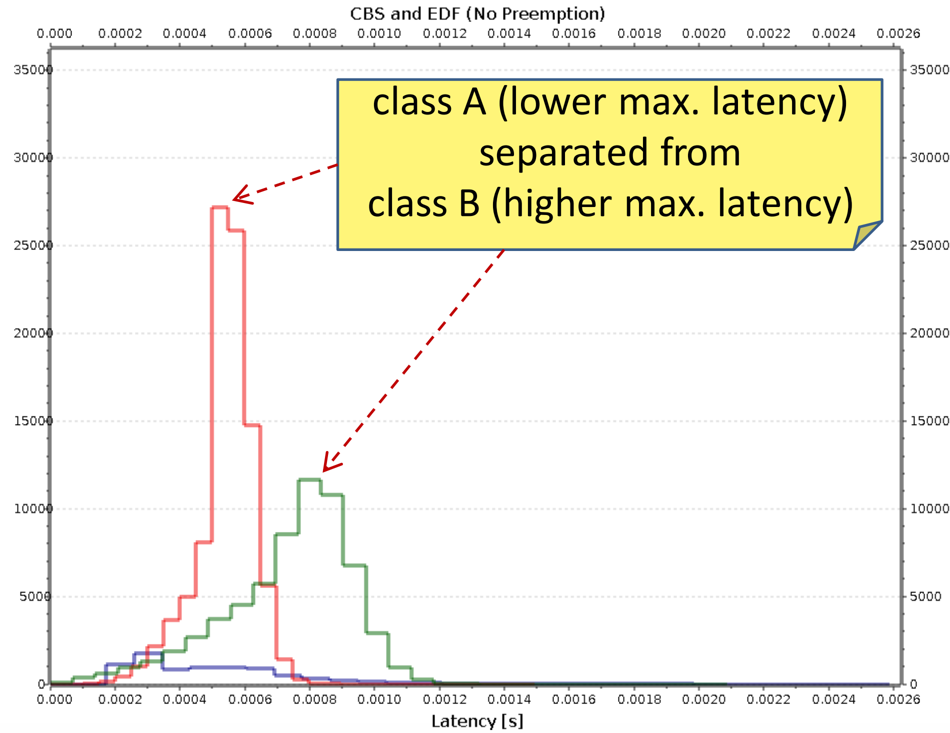
Focus of Observations on this Slide:
class A (2 ms) vs. class B (50 ms)

- Class A (2 ms)
- Class B (50 ms)
- Best Effort (none)

Without Earliest Deadline First



With Earliest Deadline First



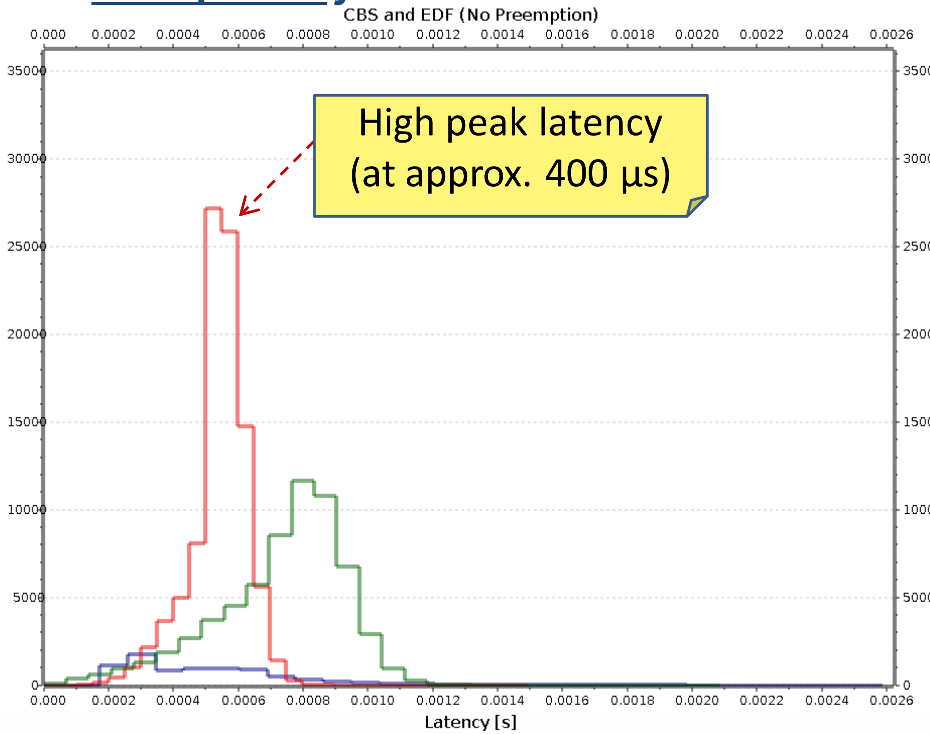
X-Axis: Latency in seconds, Y-Axis: Number of frames.

Basic Observations

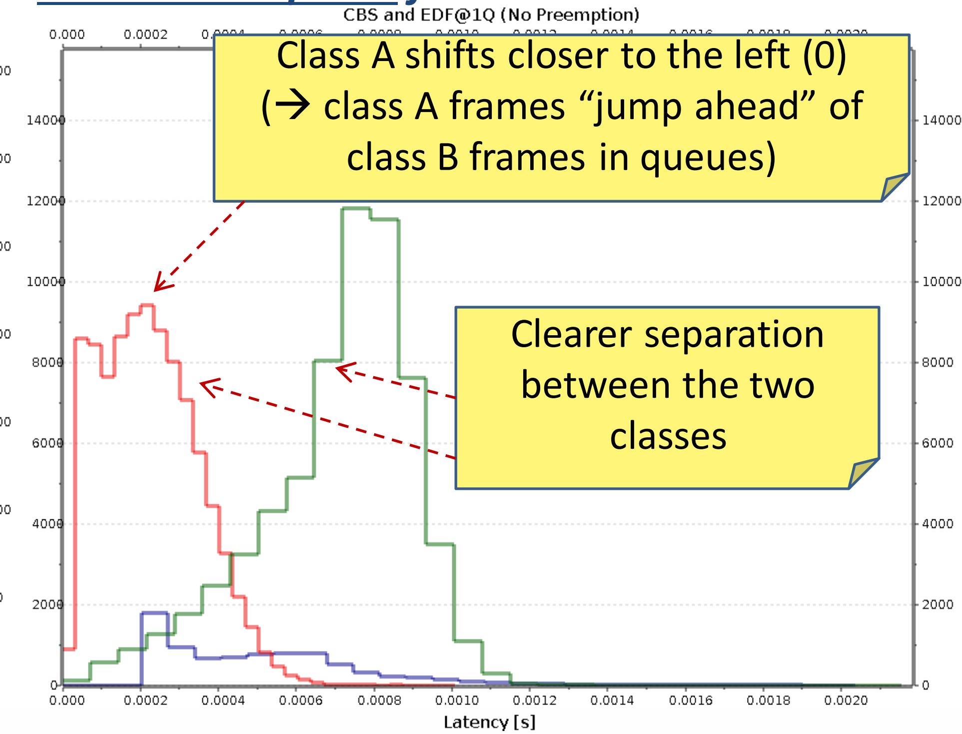
Focus of Observations on this Slide:
class A (2 ms) vs. class B (50 ms)

- Class A (2 ms)
- Class B (50 ms)
- Best Effort (none)

Earliest Deadline First:
Two queues for class A and B



Earliest Deadline First:
One shared queue for class A and B



SUMMARY

Summary 1/3

What are the advantages?

… It can scale better than (FIFO-)queues with static priorities!

- An individual desired maximum latency can be assigned to each individual stream – a single queue would be sufficient for all streams.
- Streams with lower latency requirements can benefit from streams that tolerate higher latencies.
- Latency variations introduced by different path lengths/hop count can be balanced.
  (At common egress ports.)

Summary 2/3

What are the advantages?

… not only in Automotive Systems, but everywhere where …

- there are many different real time streams …
- … with different maximum latency requirements…
- … in networks with bottlenecks and little bandwidth headroom!

The problem is …

- … <u>not only</u> to distribute all bandwidth fair (shaping) …
- … <u>but also</u> to fulfill all different maximum latency requirements …
- … <u>while avoiding</u> the burden of TDMA-scheduled traffic for use cases that do not absolutely require TDMA!

Summary 3/3

- Several aspects have not yet been considered:
  - Which scheduling algorithm(s) would be the best in terms of latency
  - Which traffic shaping algorithm should be preferred or whether traffic shaping is a good idea at all
  - What additional information would be needed in frames, if any
- The even harder part is to find algorithms that:
  - fit into AVB/TSN/802.1Q
  - can be easily implemented or are already available
- Obviously transmitting most urgent frames is very different from First-In-First-Out …

Thank you for your Attention!

Questions, Opinions, Ideas?

Johannes Specht
Dipl.-Inform. (FH)

Dependability of Computing Systems
Institue for Computer Science and
Business Information Systems (ICB)
Faculty of Economics and
Business Administration
University of Duisburg-Essen

Schützenbahn 70
Room SH 502
45127 Essen
GERMANY
T +49 (0)201 183-3914
F +49 (0)201 183-4573

specht@dc.uni-due.de
http://dc.uni-due.de