

# Urgency Based Scheduler

- updates since March 2013 -

Johannes Specht (Univ. Duisburg-Essen)



# Background and Content

- **Content**                      Urgency Based Scheduler (UBS)  
(based on March 2k13 slides, changes based on feedback)
- **UBS**                              Flexible Traffic Class?  
(cmp. AAA2C slides)

## UBS Concepts:

- Sub Shapers and Sub Queues
  - Priority Queues
  - Latency Calculation
  - Ingress Policing
  - Higher and Lower Traffic Classes
  - Scheduling
- } Today's presentation



# AVB1 – THIS MAKE'S LATENCY CALCULATION TRICKY



# Example AVB1 (1)

## Example scenario - simplified environment

- Fast Ethernet, two bridges, store & forward
- One AVB class (CBSA), no other classes  
→no need for traffic selection, etc.
- No further delays (link delays, etc.)
- **Red traffic:** One (or more!) streams, one path, constant frame size

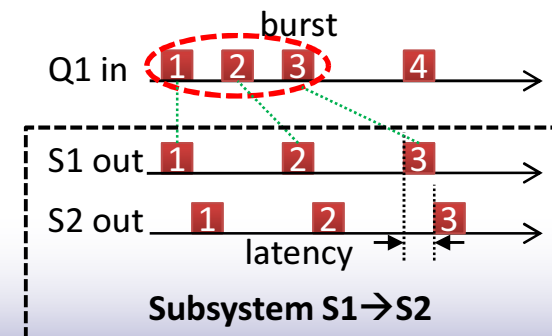
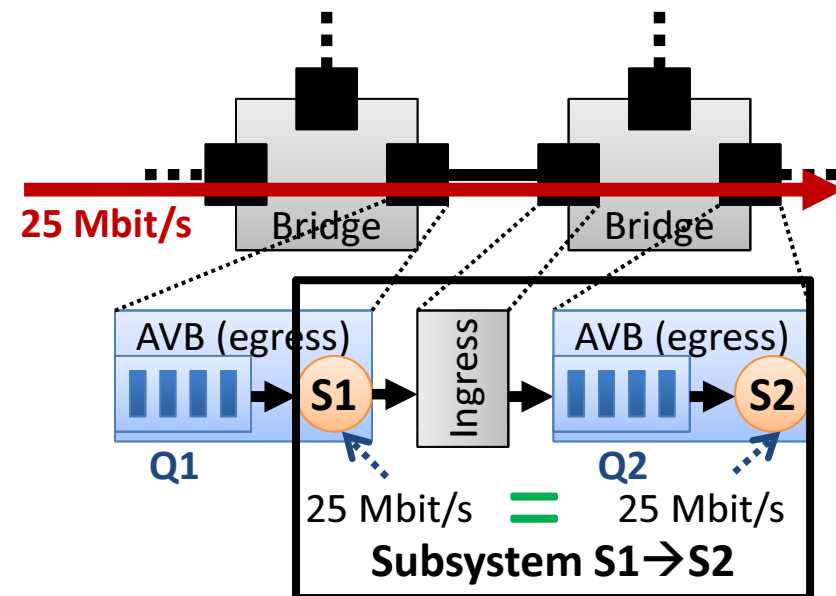
## Focus: Shaper to shaper subsystem (1 Hop)

Shaper S1: Guaranteed input bound

Shaper S2: Guaranteed output bound

## Max. Latency: Isolated subsystem analysis possible!

- input bound  $\approx$  output bound:  
→ “Entering traffic can leave” ...

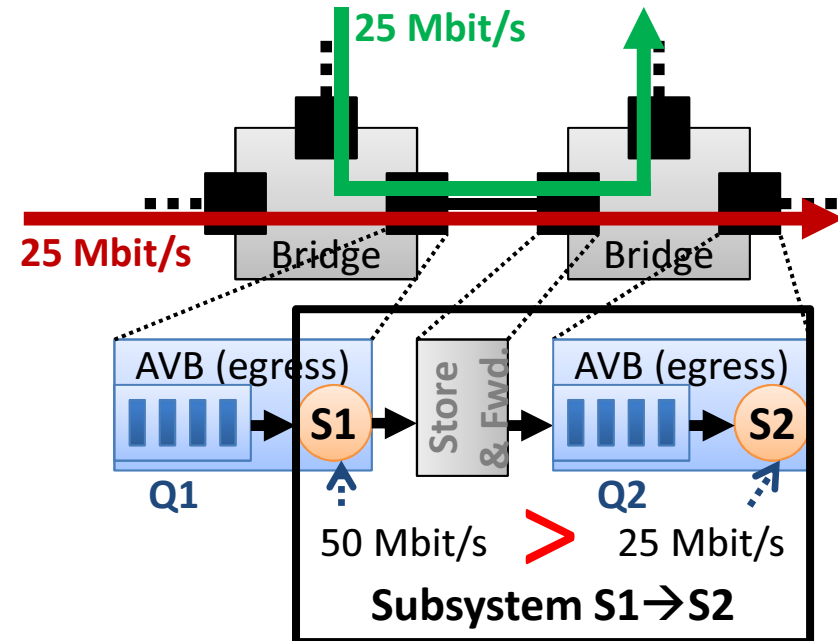


# Example AVB1 (2)

*Extended Scenario: as before, but...*

- **Green traffic:** One (or more!) streams, one path, equal frame size

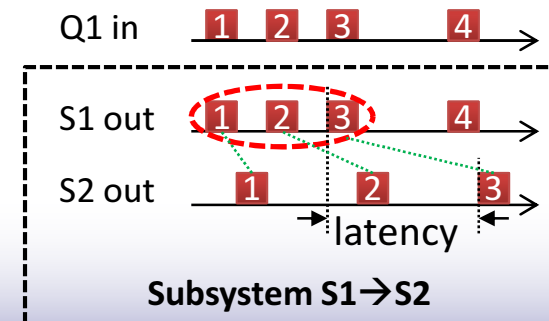
*Note: Reservations are not forced to send!*



**Max. Latency:**

Isolated subsystem analysis **impossible!**

- *Input bound > output bound*  
→ “Entering traffic can’t leave”...
- *Depends on history of the traffic (topology, other traffic)*  
→ **Complicated** → **Undesired**



# SUB SHAPERS AND SUB QUEUES



# Sub Shapers – Purpose

## Always assure:

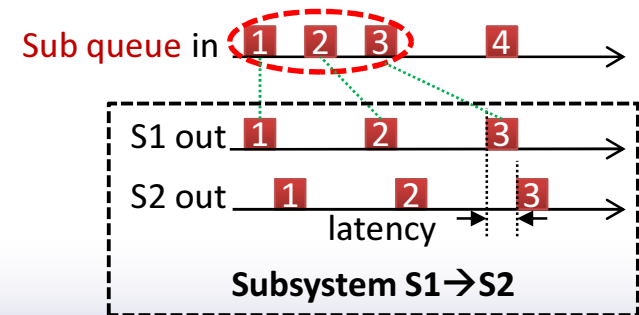
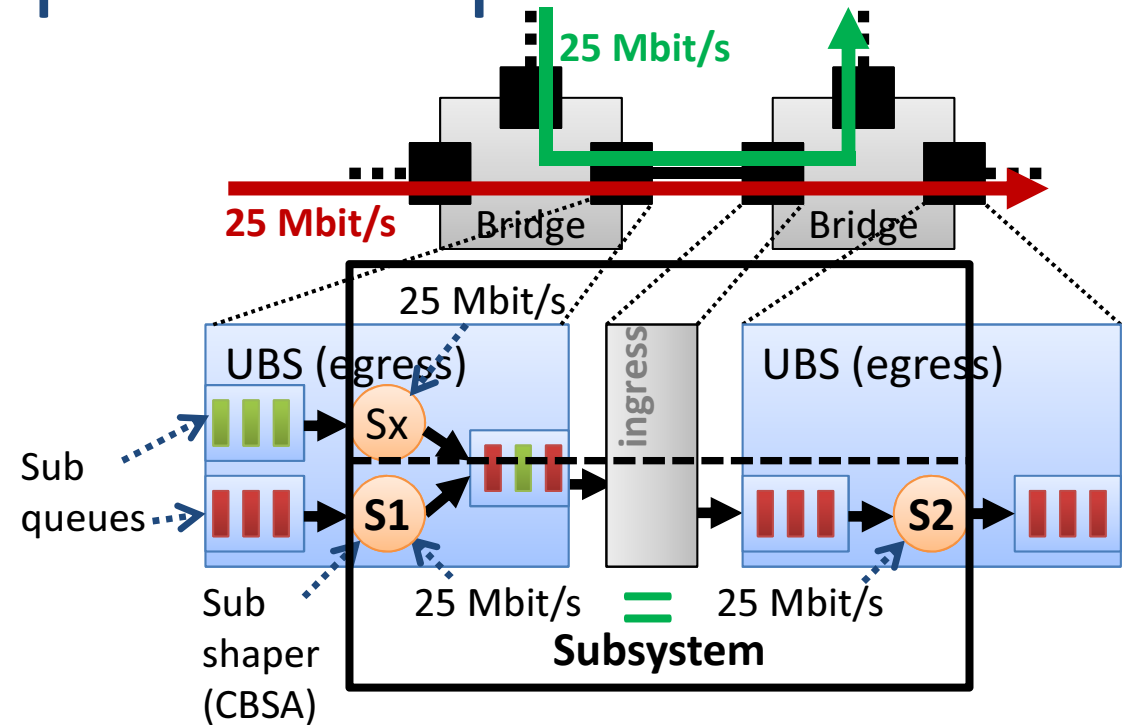
*input rate = output rate*

## Enables:

- Maximum latency calculation per subsystem

## Sub Queues:

- Consequence of Sub Shapers (1 Sub Queue per Sub Shaper)



Notes for correctness: (1) Input rate < output rate is also ok

(2) CBSA, but without being blocked by other frames/classes



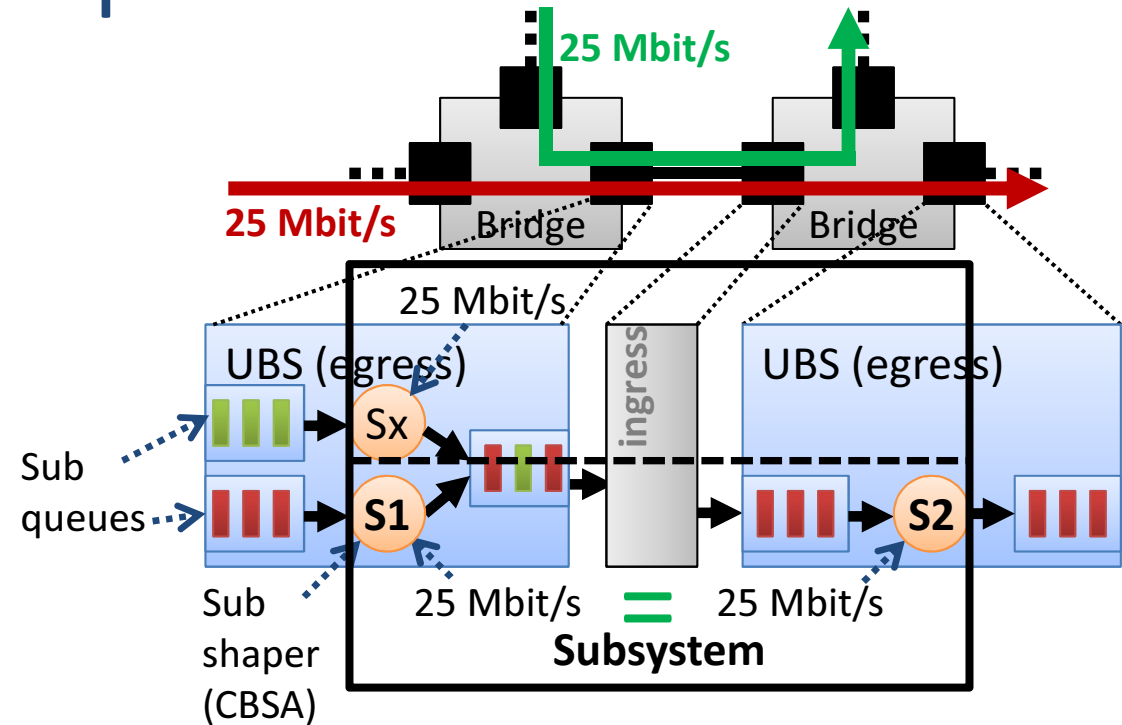
16.10.2013

Johannes Specht - University of Duisburg-Essen



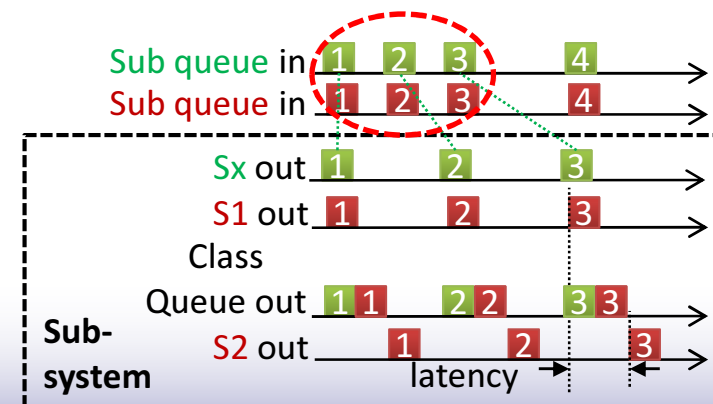
7

# Sub Shapers – Cont.



## Interfering Sub Shapers:

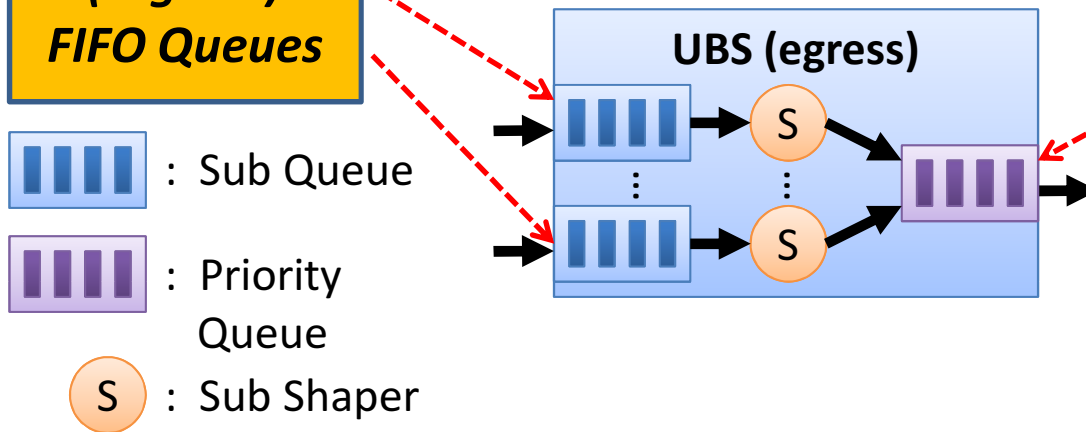
- Guaranteed input bound
- **Maximum latency calculation per subsystem possible**







**Sub Queues:  
(regular)  
FIFO Queues**

**Class Queue:  
Priority Queue**



 : Sub Queue

 : Priority Queue

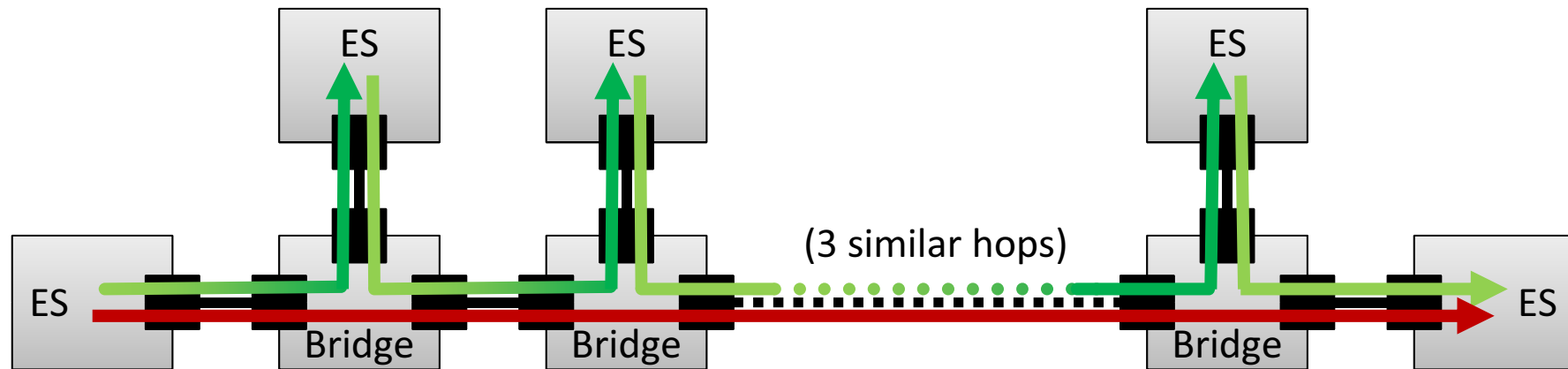
 : Sub Shaper

# PRIORITY QUEUE



# Priority Queue - Background

**Problem: Latency → Function of Hops**



## Simplified

- More hops  $\approx$  more End-2-End latency
- but*
- **Maximum latency is a requirement ...**
- ... **not only** a function of the topology

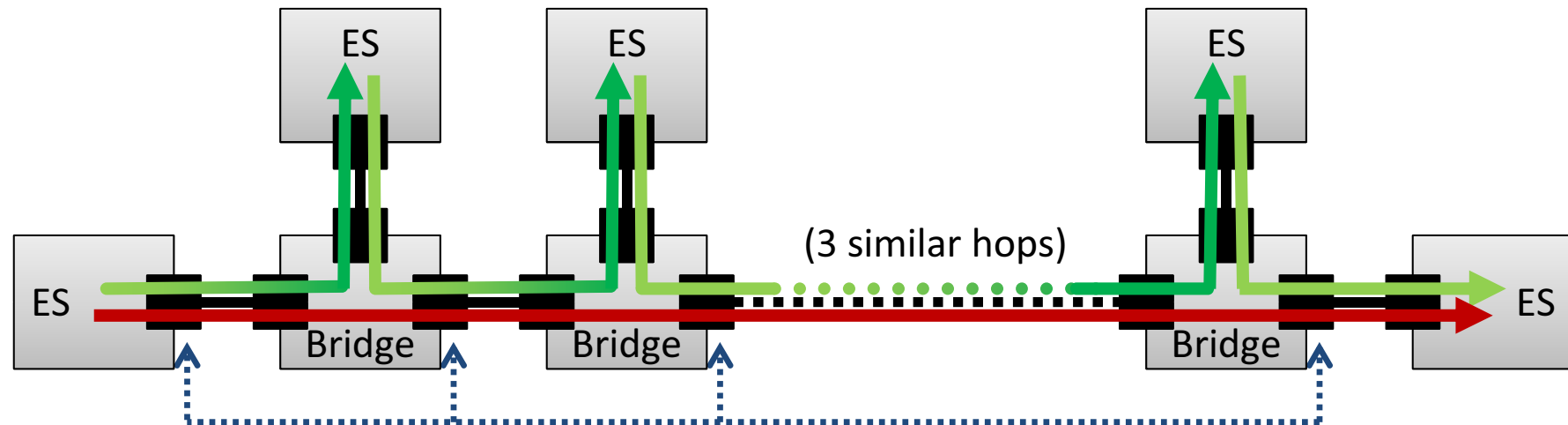
## Example:

Traffic	# Hops	Rate	Max. Required End-2-End Latency
Red	7	25 Mbit/s	1 ms → Urgent
Green	2 or 3	25 Mbit/s	10 ms → Relaxed



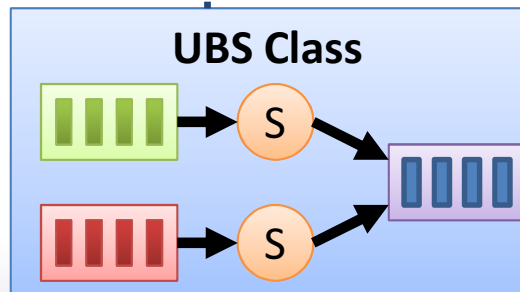
# Priority Queue – Purpose

## Accelerate Urgent Traffic



### Example:

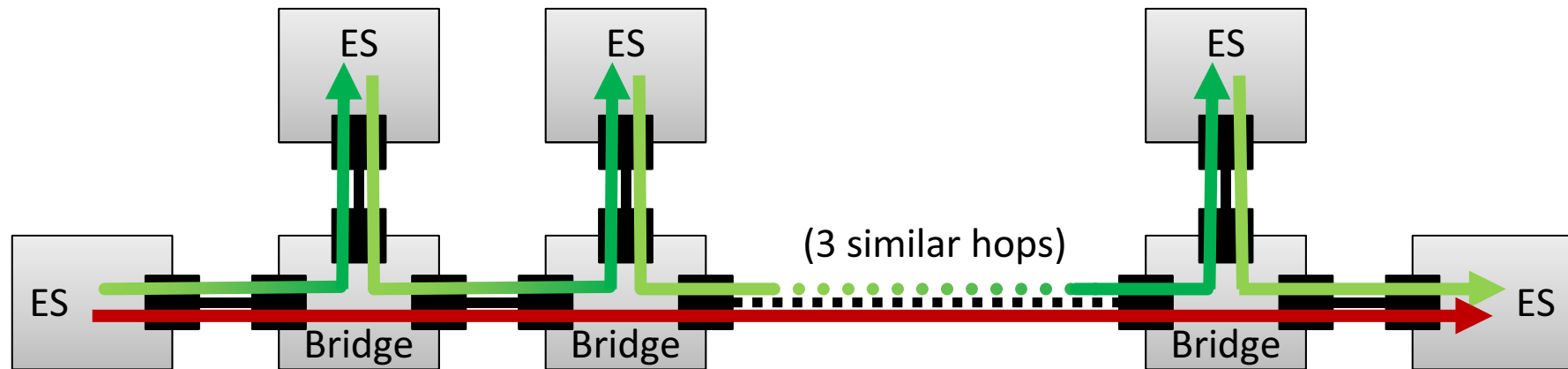
Traffic	# Hops	Rate	Max. Required End-2-End Latency
Red	7	25 Mbit/s	1 ms → Urgent
Green	2 or 3	25 Mbit/s	10 ms → Relaxed



→ Queue urgent traffic “before” more relaxed traffic, i.e. “make **red** traffic faster”

# Priority Queue - Example

## Accelerate Urgent Traffic



Traffic	# Hops	Rate	Max. <u>Required</u> End-to-End Latency	Max. Frame Size	Max. <u>Calculated</u> End-to-End Latency*
Red	7	25 Mbit/s	1 ms → urgent	128 Byte	962,56 μs @ 7 Hops
Green	3 or 2	25 Mbit/s	10 ms → relaxed	512 Byte	1331,20 μs @ 3 Hops 832,85 μs @ 2 Hops

### Assumptions

- 100 Mbit/s Fast Eth.
- No TDMA Class present
- Store & forward
- No link delays
- Preemption of lower classes (max. 128 Byte framelet)
- Simple priority queue algorithm: Strict priority per Sub-Shaper/-Queue

\*: Result of a preliminary worst case calculation



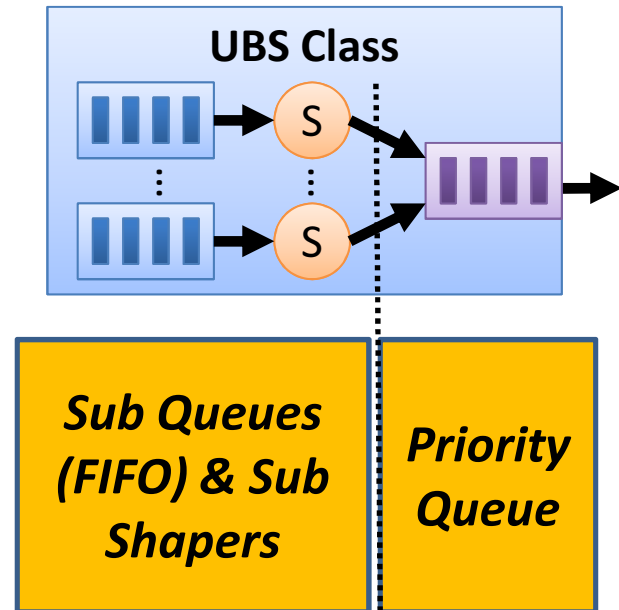
16.10.2013

Johannes Specht - University of Duisburg-Essen

# Summary

## Proposal for Flexible Traffic Class

- Urgency Based Scheduling
  - Sub shaper & Sub queues:
    - Enables latency analysis per Hop
  - Priority queue:
    - Loosens topology dependency on max. Latency
    - Maps Latency Requirements



## Next steps

- Present Latency Calculation
- **Other?**



# Thank you for your Attention!

## *Questions, Opinions, Ideas?*

***Johannes Specht***

***Dipl.-Inform. (FH)***

Dependability of Computing Systems	Schützenbahn 70
Institute for Computer Science and	Room SH 502
Business Information Systems (ICB)	45127 Essen
Faculty of Economics and	GERMANY
Business Administration	T +49 (0)201 183-3914
University of Duisburg-Essen	F +49 (0)201 183-4573

[specht@dc.uni-due.de](mailto:specht@dc.uni-due.de)

<http://dc.uni-due.de>

