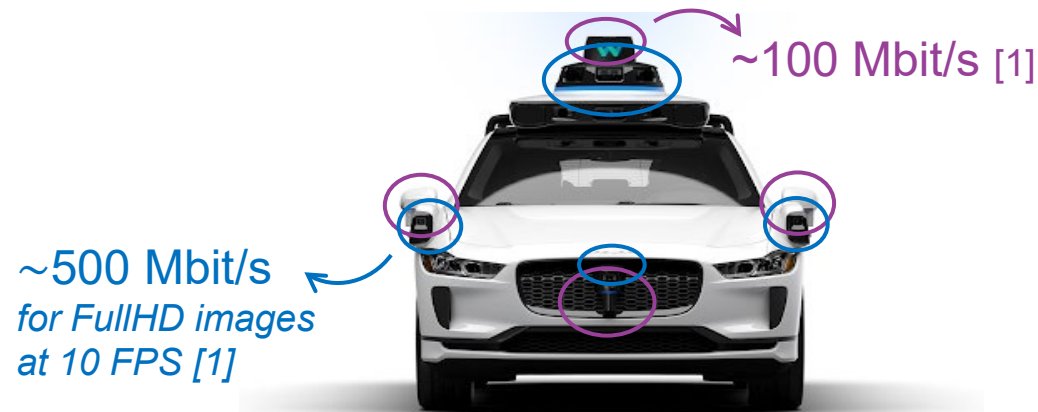# ROS 2 (and DDS) Compatible Selective Large Data Transfer
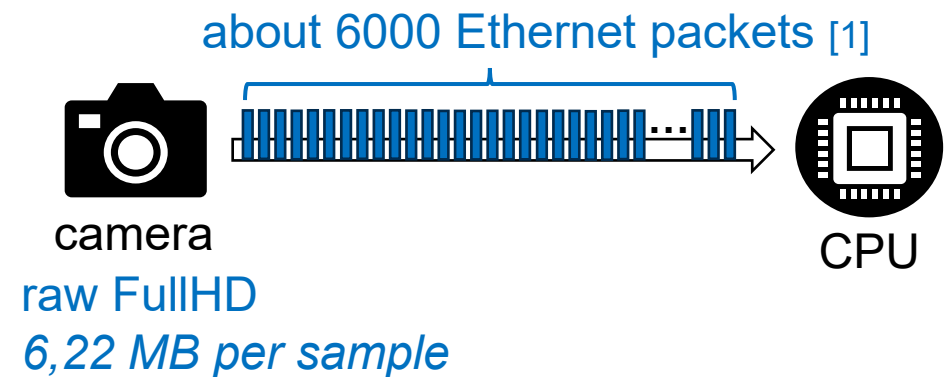
Nora Sperling & Rolf Ernst, at RosCon '24

Large data samples in time- and safety-critical (distributed) systems

- **camera and lidar sensors are often used for perception in autonomous (robotic or vehicular) systems**
  - sample resolution is increasing, creating large data objects at high data rates
  - time- and safety-critical effect chains and process flows

~100 Mbit/s [1]

~500 Mbit/s
for FullHD images at 10 FPS [1]

https://waymo.com/waymo-one
https://support.google.com/waymo/answer/9190838?hl=en

about 6000 Ethernet packets [1]

camera
raw FullHD
6,22 MB per sample

CPU

Sperling | Slide 2

[1] *Reducing Communication Cost and Latency in Autonomous Vehicles with Subscriber-centric Selective Data Distribution,*
Nora Sperling, Rolf Ernst at 99th VTC-Spring, 2024

Technische
Universität
Braunschweig

IDA
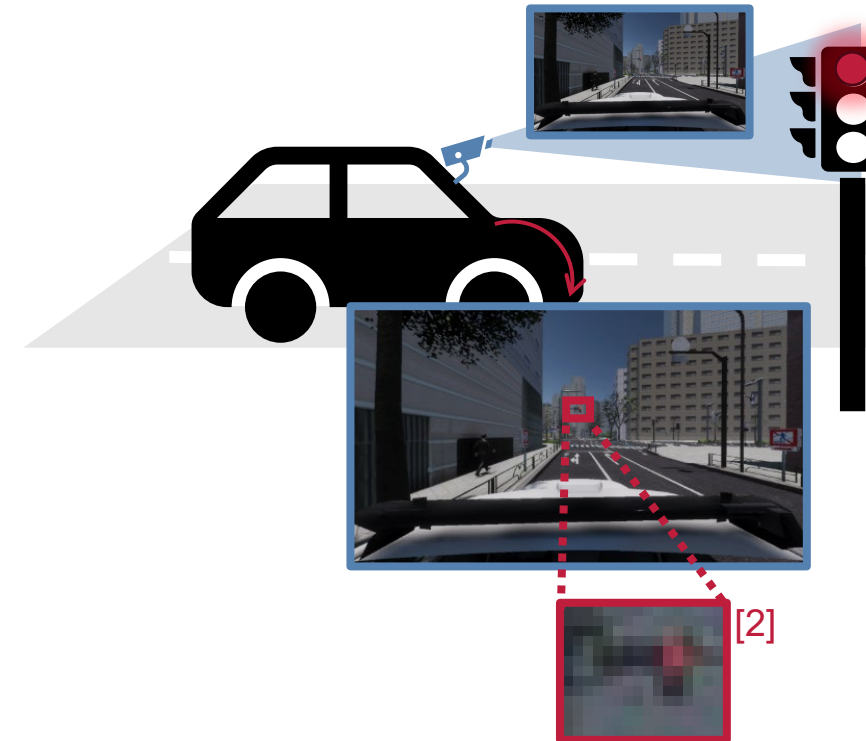INSTITUTE OF
COMPUTER AND
NETWORK ENGINEERING

Problem: Distribution of large data samples

- **publish-subscribe (PS) communication has limited efficiency for large data object distribution under real-time constraints**
  → high network cost and latency

- **selective application-<u>driven</u> data transfer often sufficient**
  - example: selective data transfer for ML based perception

[2]

[2] *Traffic light detection found in Autoware:* https://autowarefoundation.github.io/autoware.universe/main/perception/traffic_light_map_based_detector/

Technische
Universität
Braunschweig

INSTITUTE OF
COMPUTER AND
NETWORK ENGINEERING

Contribution

- **a basic library named Application-centric Data Object Management**
  - first step towards subscriber-driven request/reply-based communication within ROS 2
 - *find the library here: github.com/IDA-TUBS/ApplicationDataObjectManagement*

- **introduction of a virtual sample type**
  - managed like a regular ROS 2 sample and therefore allows to maintain compatibility with ROS 2

Technische Universität Braunschweig

INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Approach – Step 1: Control Data Sample

ROS 2 publishes Virtual Object O_n describing the large data sample and using the regular ROS 2 infrastructure
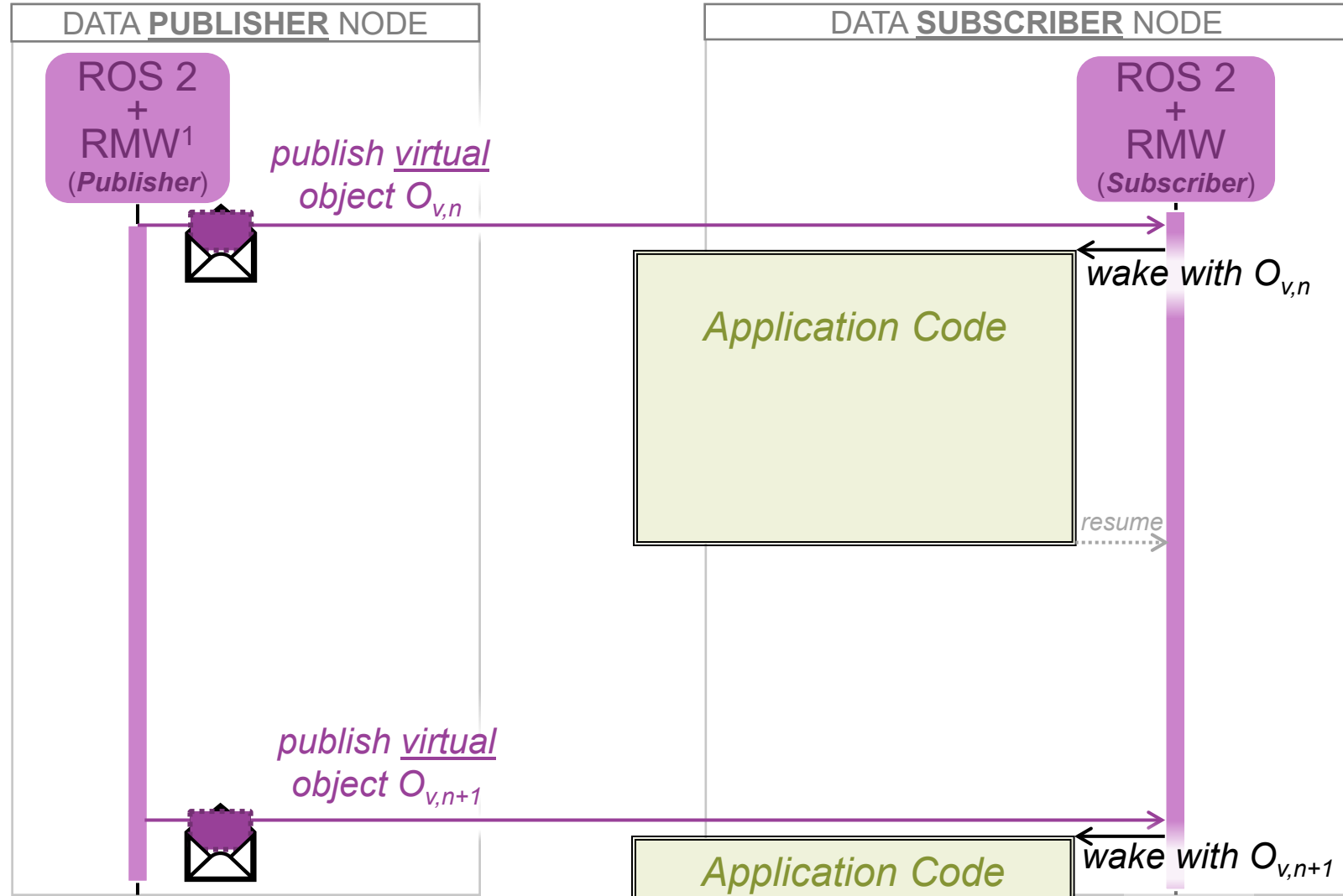
n-th object sample:

O_n

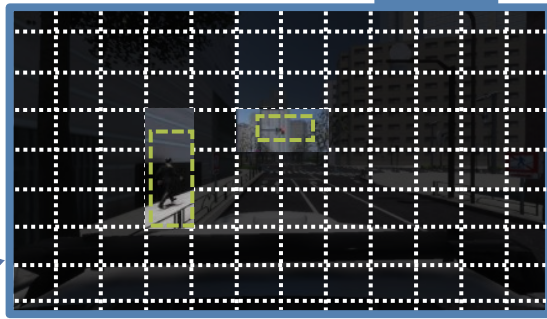*virtual* object sample:

$O_{v.n}$
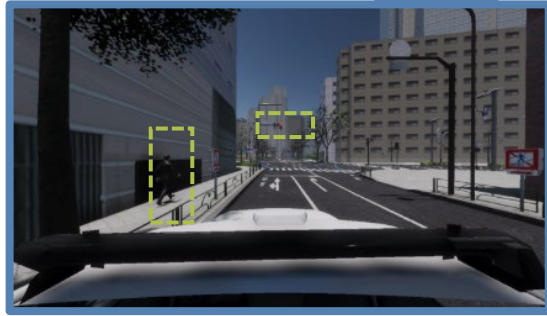
object_structure	object_type	object_seq_nr n

→ **virtual objects keep ROS 2 protocol and semantics**

DATA **PUBLISHER** NODE

DATA **SUBSCRIBER** NODE

ROS 2 + RMW[1] (*Publisher*)

ROS 2 + RMW (*Subscriber*)

publish virtual object $O_{v,n}$

wake with $O_{v,n}$

Application Code

resume

publish virtual object $O_{v,n+1}$

wake with $O_{v,n+1}$

Application Code

footnotes: 1   *ros middleware*

INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Data exchange is then managed by an ***Application-centric Data Object Management (*ADOM) directory**.

n-th object sample:

O_n

***virtual* object sample:**

$O_{v.n}$

| object_structure | object_type | object_seq_nr n |

→ **on-demand selection and transport controlled by subscriber application**

DATA **PUBLISHER** NODE

ROS 2 + RMW (*Publisher*)

DATA **SUBSCRIBER** NODE

ROS 2 + RMW (*Subscriber*)

wake with $O_{v,n}$

Application Code

ADOM DIR (*Publisher*)

ADOM DIR (*Subscriber*)

USER CODE (*Subscriber*)

read selection

block request

block responses

write back

read selection

block request

block responses

write back

resume

Technische Universität Braunschweig

INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Evaluation of ADOM library

- **realistic rosbag for evaluation**
 - recorded <u>camera images</u> and <u>regions of interest</u> during a test drive with the Autoware <u>digital twin simulation</u> (AWSIM) and Autoware <u>AD software stack</u>
  - camera images/ regions of interest are transmitted between two PCs via 1 Gbit Ethernet

test run 1:
state-of-the-art DDS

publishing topics:
*/sensing/camera/**image_raw***
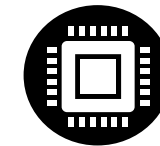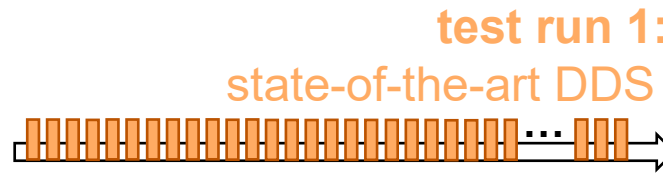
subscribing topics:
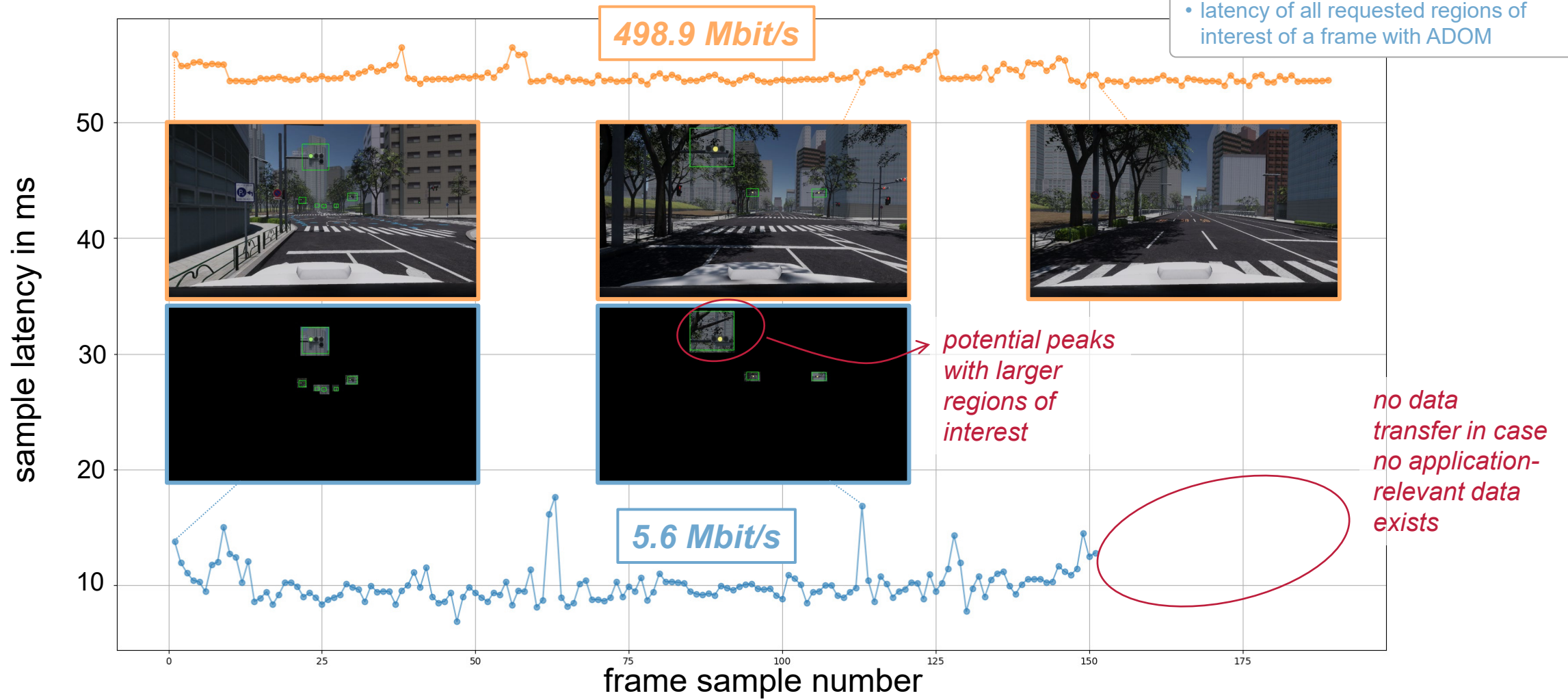*/perception/traffic_light_recognition/**rois***
*/sensing/camera/**image_raw***

test run 2:
transmission of application
relevant data via ADOM

publishing topics:
*/adom/**virtual_object**/image*
+ <u>ADOM library</u>
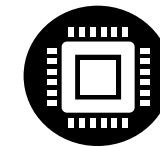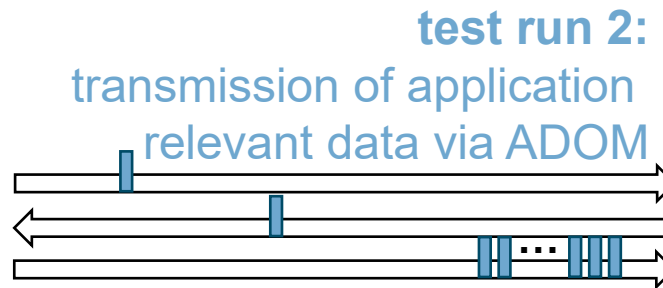
subscribing topics:
*/perception/traffic_light_recognition/**rois***
*/adom/**virtual_object**/image*
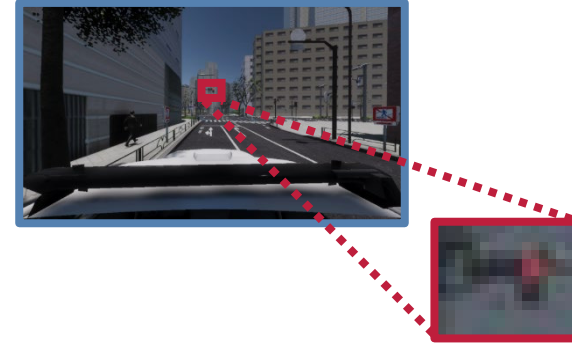**+ <u>ADOM library</u>**

Experimental Results

Conclusion

Motivation:
- ROS 2 distribution (DDS) of <u>large data samples</u> is inefficient with <u>respect to application specific data relevance</u>

- ***Application-centric Data Object Management* adds a new interface that allows applications to address and directly exchange data selections**
  - compatible to ROS 2 control and callback mechanisms due to virtual topics
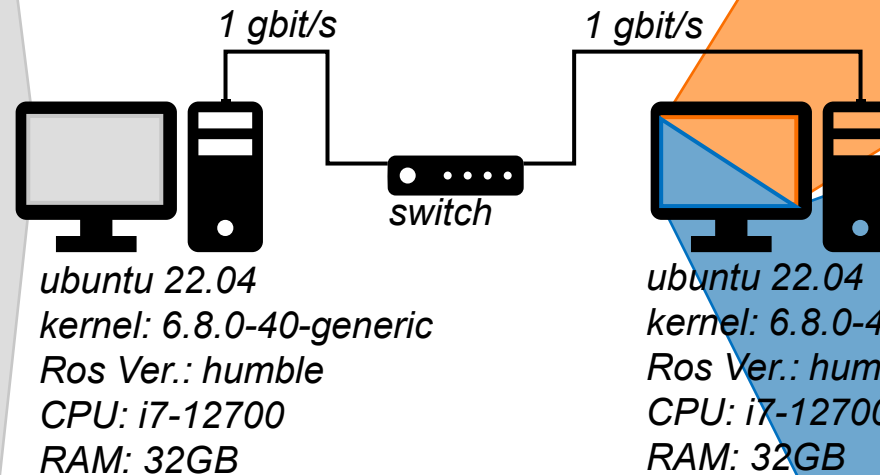 - lean data transport protocol minimizes ROS 2 protocol overhead

Thank you for your attention!

find the library here: *github.com/IDA-TUBS/ApplicationDataObjectManagement*

Technische Universität Braunschweig

IDA INSTITUTE OF COMPUTER AND NETWORK ENGINEERING

Backup – Setup of the Experiment

DATA **PUBLISHER** NODE

- Publish **image** (1080x1920) on topic: */sensing/camera/image_raw* every 100 ms (10 FPS)

- Publish **virtual image** on topic: */adom/virtual_object/image* every 100 ms

- Publish **region of interest** on topic: /perception/traffic_light_recognition/rois every 100 ms

- Running an active instance of an **ADOM directory** to answer any incoming requests

ubuntu 22.04
kernel: 6.8.0-40-generic
Ros Ver.: humble
CPU: i7-12700
RAM: 32GB

1 gbit/s

switch

1 gbit/s

ubuntu 22.04
kernel: 6.8.0-40-generic
Ros Ver.: humble
CPU: i7-12700
RAM: 32GB

test run 1:
state-of-the-art DDS

test run 2:
transmission of application relevant data via ADOM

DATA **SUBSCRIBER** NODE

- **Simplified application: marks** regions of interest with a box

- Subscribed to topic: /sensing/**camera**/image_raw

- Subscribed to topic: /perception/traffic_light_recognition/rois

synchronized

DATA **SUBSCRIBER** NODE

- **Simplified application: marks** regions of interest with a box

- Subscribed to topic: */adom/**virtual_object**/image*

- Subscribed to topic: /perception/traffic_light_recognition/rois

- Running an active instance of an **ADOM directory** to make requests

synchronized

Camera images and regions of interest come from a rosbag recorded during a test drive with the Autoware simulation and Autoware AD software stack.

Note: In a realistic scenario, the image publisher and the RoI publisher would not be located on the same node.

Technische Universität Braunschweig

IDA INSTITUTE OF COMPUTER AND NETWORK ENGINEERING