UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

WueWoWAS'23 | 29.06.2023 | Würzburg

On Data Plane Multipath Scheduling for Connected Mobility Applications

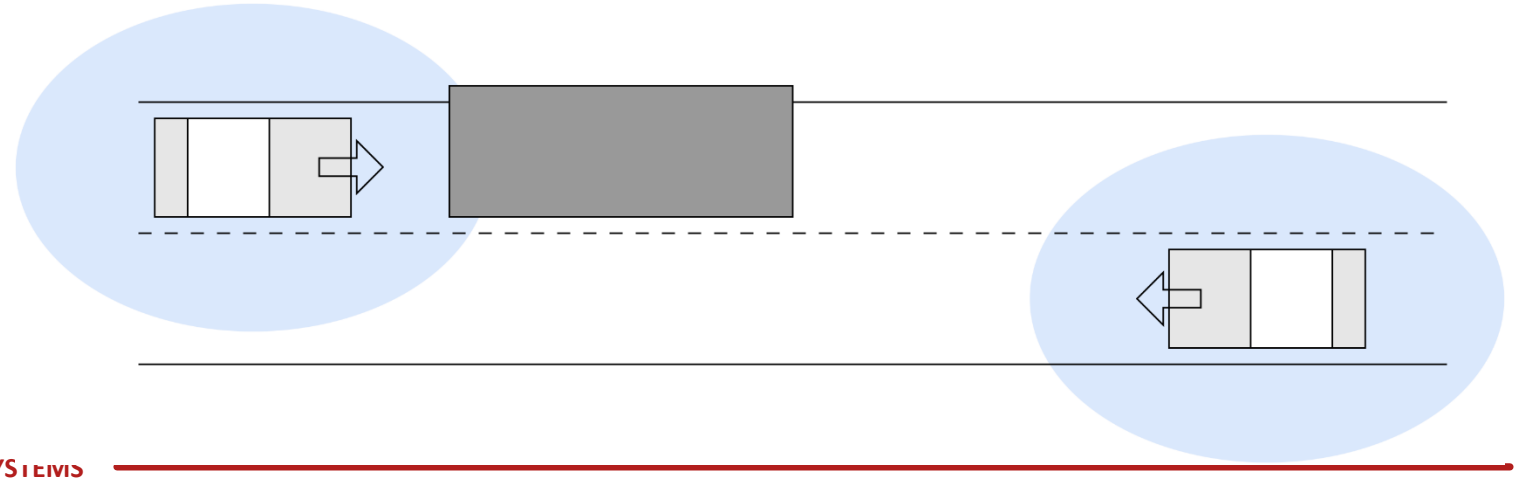
Martin Herrmann, Amr Rizk

PODIUM

PALUNO
The Ruhr Institute for Software Technology

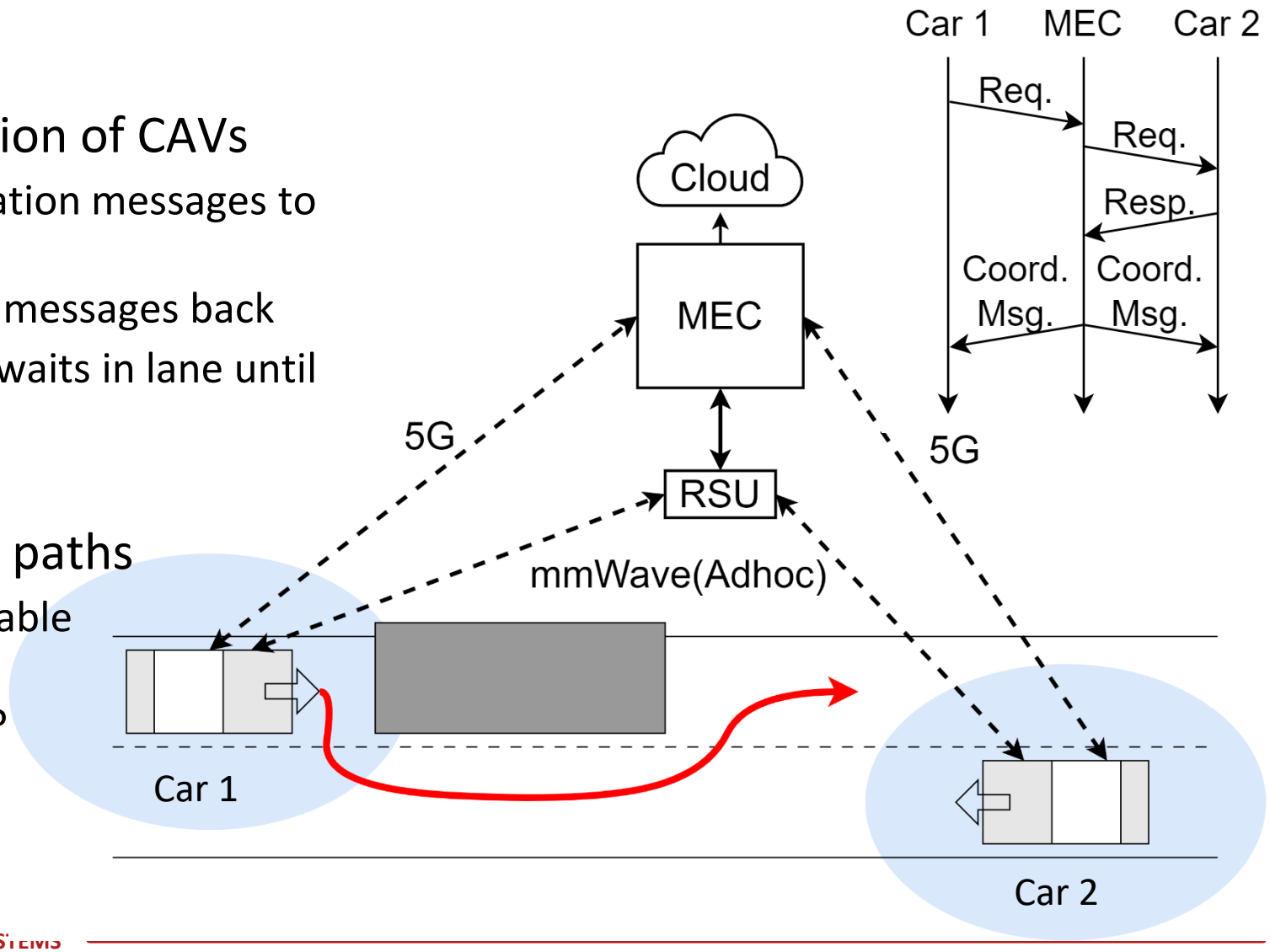
Motivation – Cooperative Connected Automated Mobility Example: Corridor Management

- Object blocks the road
- Connected Automated Vehicles (CAVs) only have a limited Field of View
- CAV can not safely pass the object
- Naive solution: human operator has take over to resolve the situation



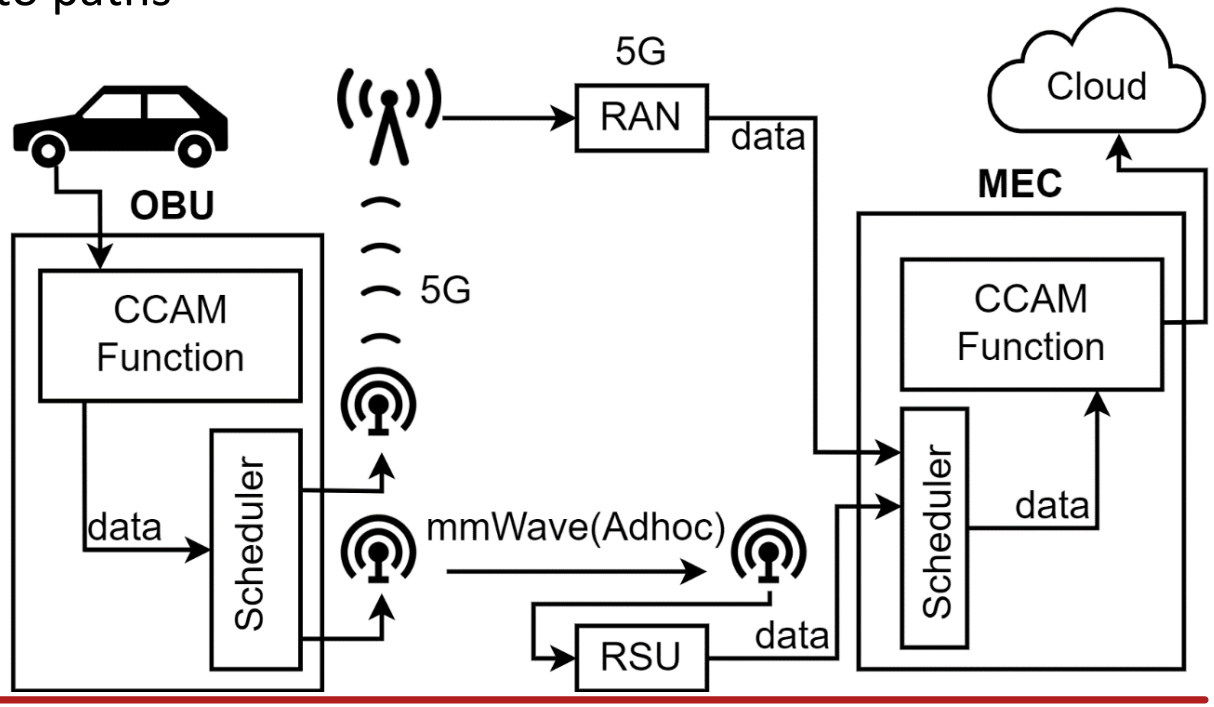
Scenario – Corridor Management

- Better solution: Coordination of CAVs
  - Both CAVs send coordination messages to the MEC
  - MEC sends coordination messages back
  - Result: Right CAV slows/waits in lane until left CAV passes
- Multipath communication paths
  - Allow for redundant/reliable communication
  - Do we always want that?

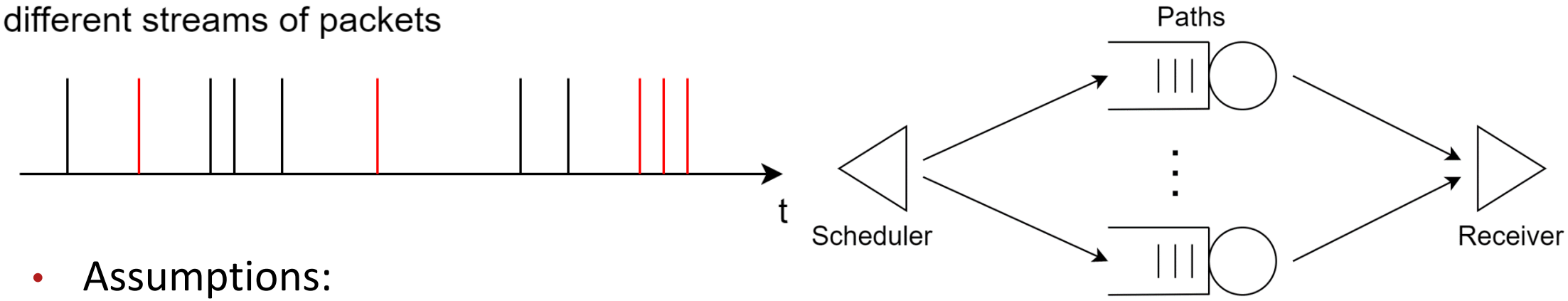


Problem statement (1)
The Setup

- Multiple communication technologies are availble
  - Each with their own characteristics e.g. 5G NR, ITS-G5, mmWave WiFi
- Scheduler
  - Possibly duplicates and maps packets to paths
  - Drops duplicated packets on arrival



Problem statement (2)



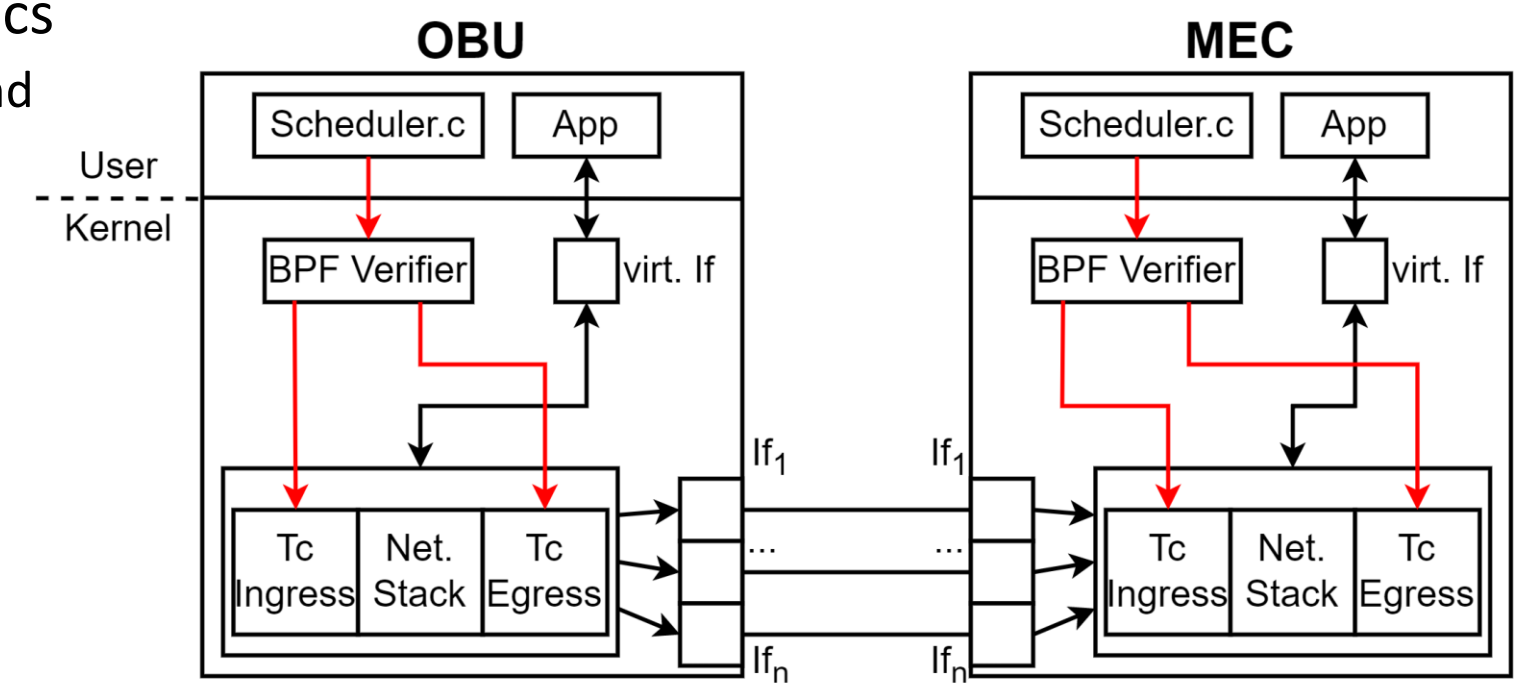
- Assumptions:
  - A stream is not split on paths (no fork then join)
  - A stream may be replicated on multiple paths
  - Locally FIFO per path; priority scheduling per path possible (if N streams per path)
- Goals:
  - Explore the scheduler design space in terms of throughput vs. reliability (through replication)
  - We are mainly concerned with a metric of the form $\mathrm{P}\left[delay \geq x\right] \leq \varepsilon_x$
  - Adaptively decide on mapping of streams to paths

Options for realizing Multipath schedulers

- Scheduler in user space
  - Flexible
  - Slower
- Scheduler in kernel space (tc/XDP)
  - Restricted
  - Fast
  - Programming in terms of Traffic Control (tc) or eXpress Data Path (XDP)
- Realization on the data plane
  - Transparency

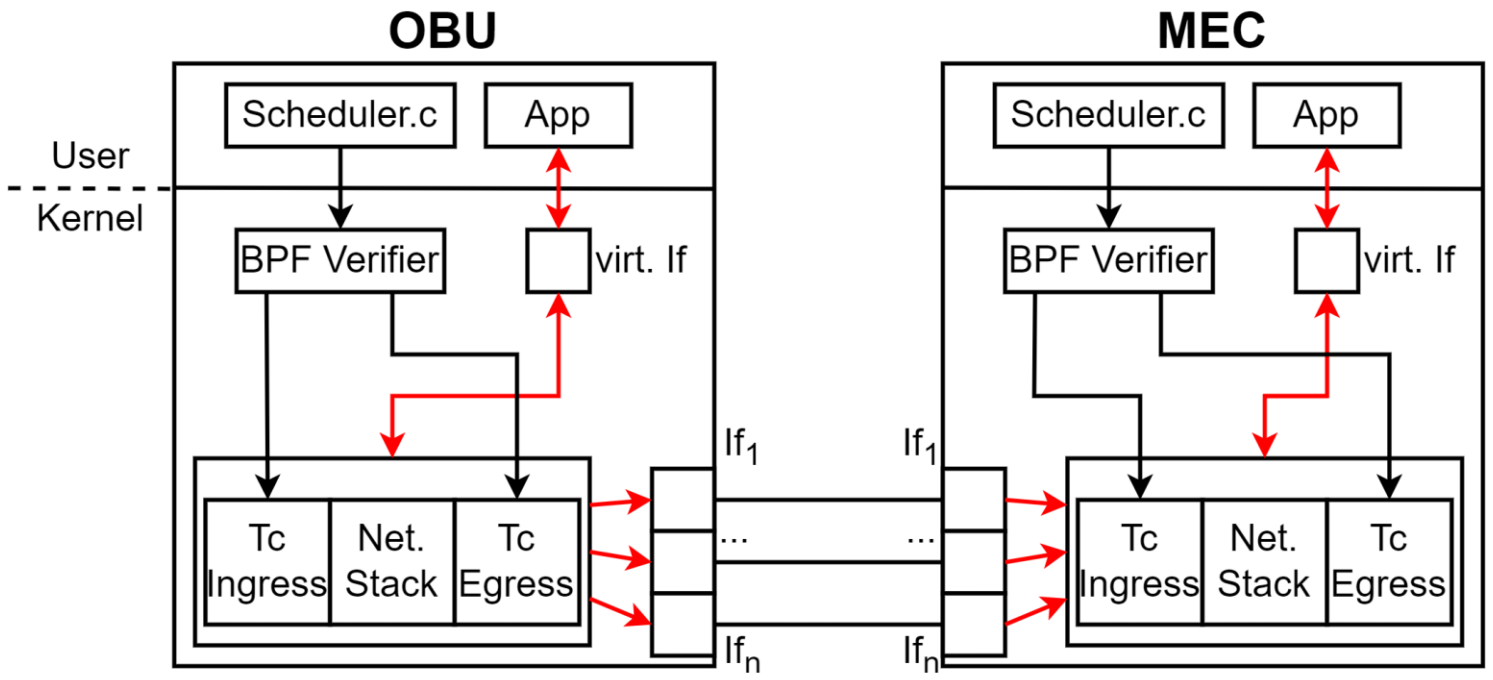
Realization (1)

- Scheduling programs are written in C
- The programs have to pass a bpf (Berkeley packet filter) verifier
- Programs are loaded onto ingress and egress as filters for qdiscs
  - Can be changed beforehand and possibly at runtime

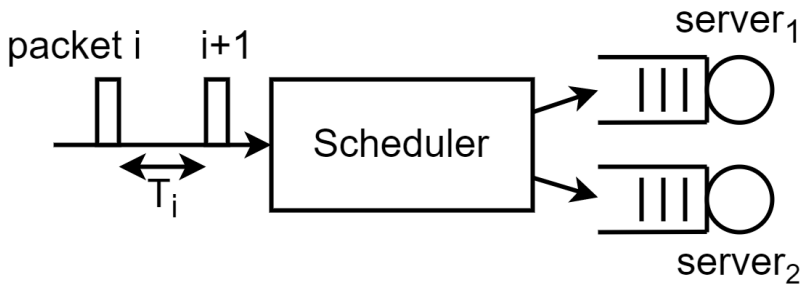


Realization (2)

- Sender side: packets are cloned or written on specified interfaces
  - Actions and interfaces are specified in the programs
- Receiver side: Remove clones from data stream
  - **Highly non-trivial** for TCP
  - Can also be used for **in-band monitoring**



Analytical Model



- Analytical model shows the latency improvement through packets replication
- Assumptions:
  - 2 paths only
  - inter-packet times T is iid exponential exp(λ)
  - service times X are iid exponential exp(μ)
- Model limitations:
  - Pure replication
  - Open loop
- Model can be adapted to encompass more realistic traffic and scheduling

- Response time distribution
  - from the response time recursion

$$r \overset{D}{=} \max_{n \geq 0} \left\{ \sum_{i=1}^{n+1} \min_{j \leq 2} \{X_{i,j}\} - \sum_{i=1}^{n} T_{i-1} \right\}$$

- Bound on the tail of the response time distribution
  - using Doob's inequality [Ciucu'21]

$$\mathsf{P}\left[r \geq \sigma\right] \leq \mathsf{E}\left[e^{\theta \min_j X_{1,j}}\right] e^{-\theta\sigma}$$

$$\mathsf{P}\left[r \geq \sigma\right] \leq \frac{2\mu}{\lambda} e^{-(2\mu-\lambda)\sigma}$$

Summary and conclusions

- Different CCAM scenarios require different grades of throughput and reliability
  - adaptive scheduling of streams to mutliple available communication technologies comes to the rescue
- Goal: provide **flexible model-based multipath schedulers on the data plane** to
  - explore the scheduling design space of throughput vs. reliability
  - allow optimizations later on (e.g. using the model shown)
- Currently first realizations will be tested in a joint 5G testbed for autonomous vehicles in the city of Ulm

UNIVERSITÄT DUISBURG ESSEN

Offen im Denken

WueWoWAS'23 | 29.06.2023 | Würzburg

Thank you for your attention

Martin Herrmann, Amr Rizk

PALUNO
The Ruhr Institute for Software Technology