Universität Rostock

Traditio et Innovatio

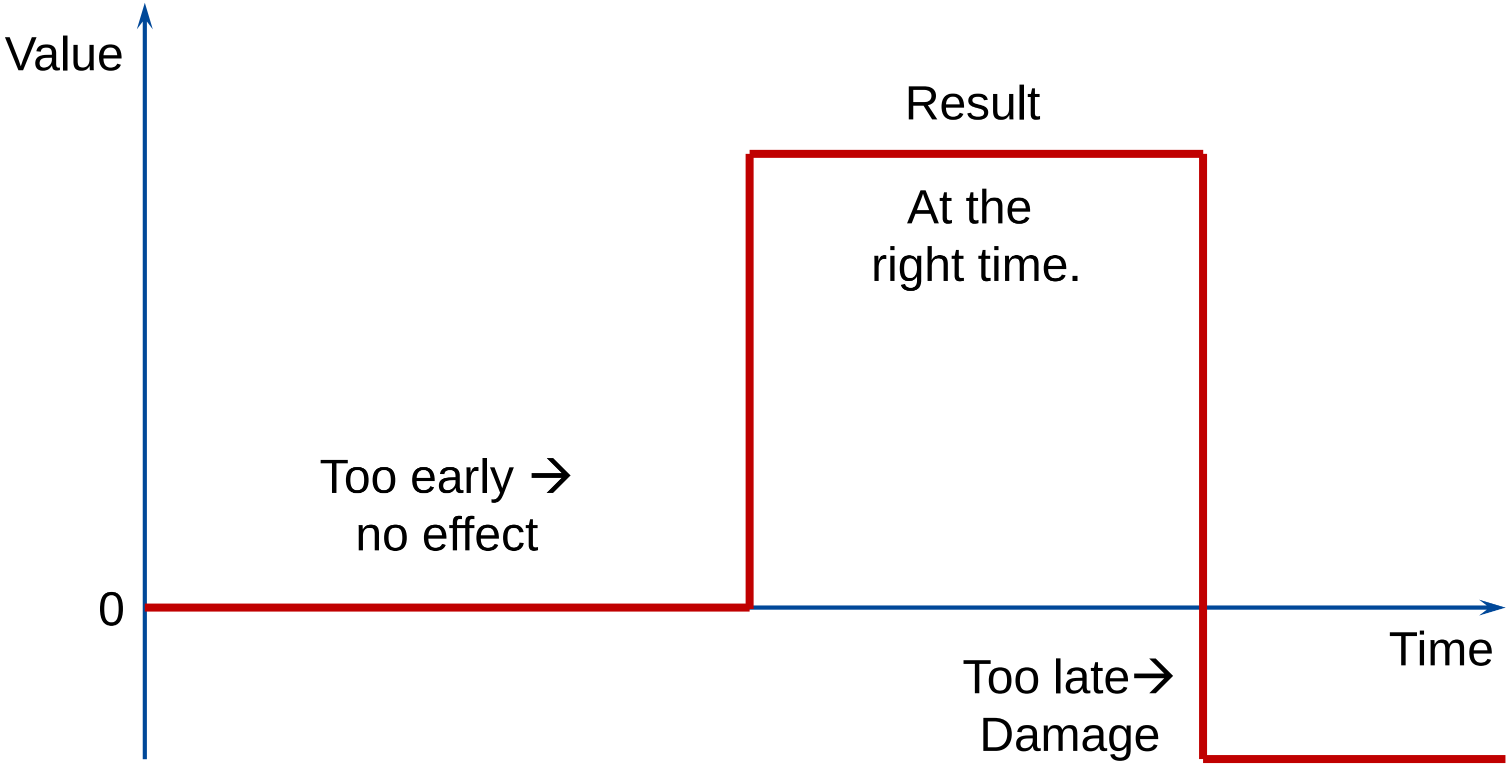
Realtime Publish/Subscribe for Cyber-Physical Systems

NEidI / Projekt CSI

Peter Danielis
Parallele Systeme (ParSys)

Helge Parzyjegla
Architektur von Anwendungssystemen (AVA)

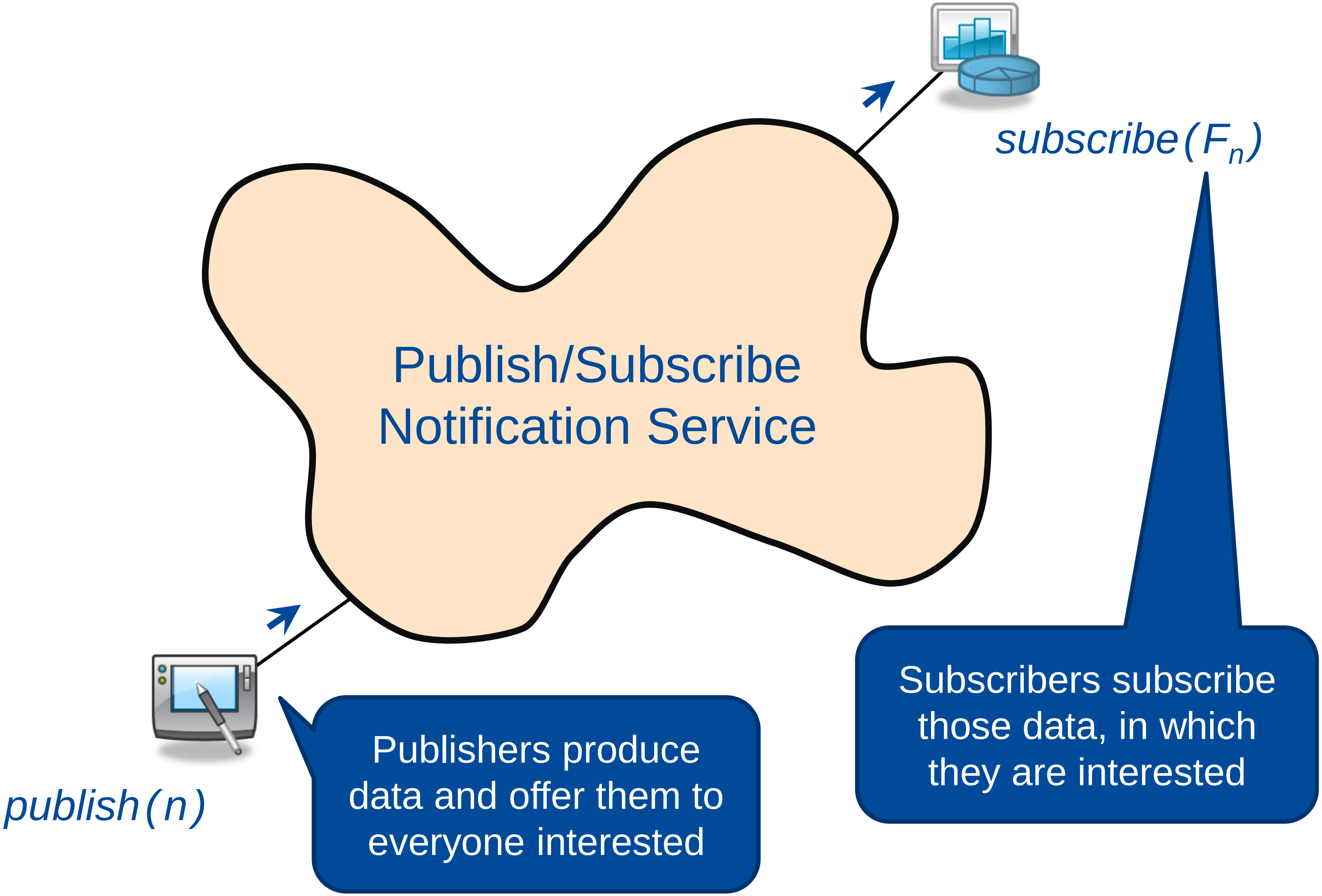
What is Realtime (Echtzeit)?



Not neccessarily fast, but predictable!

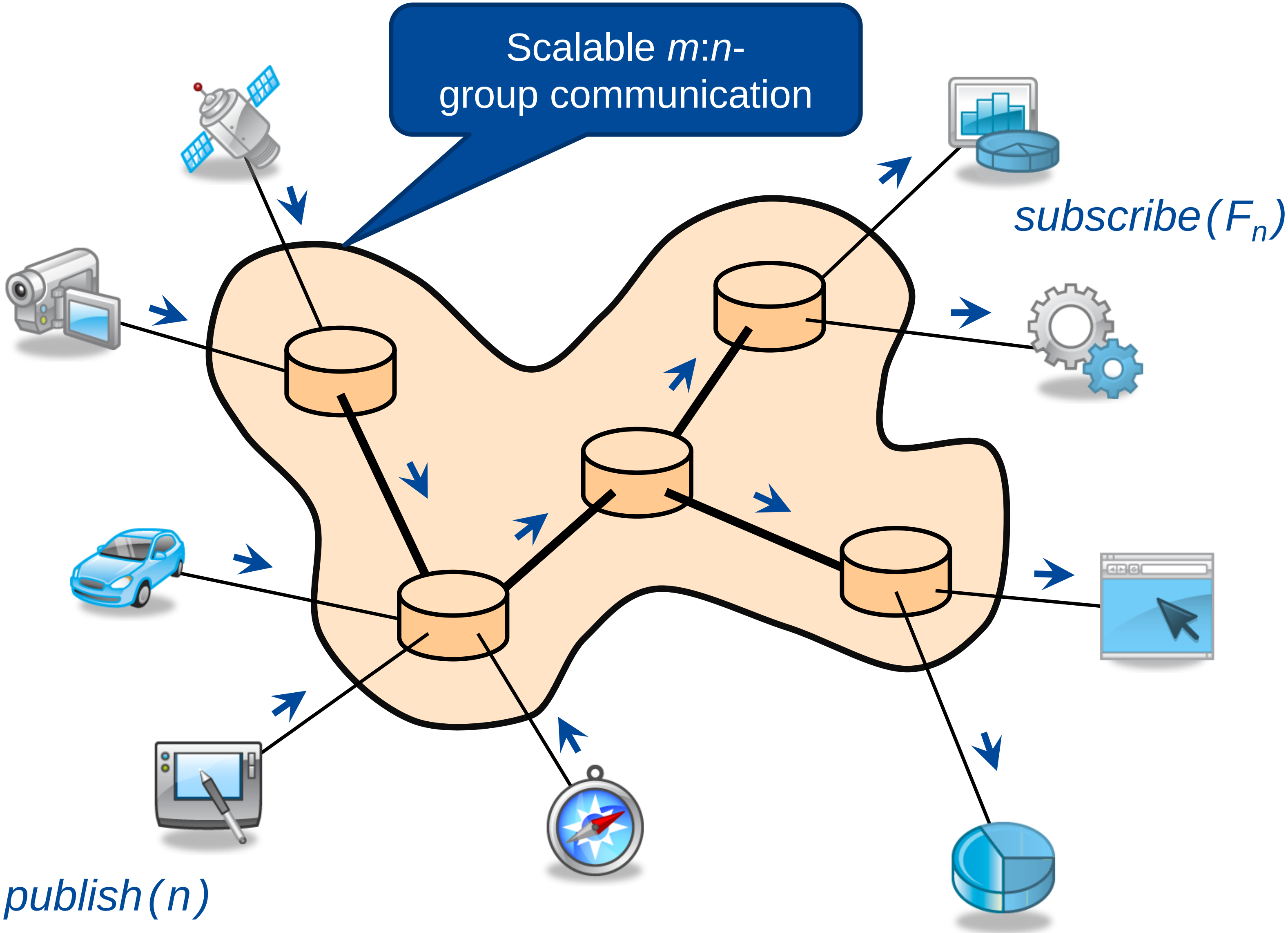
→ Do the right thing at the right time.

What is Publish/Subscribe?



Subscribers subscribe those data, in which they are interested

Publishers produce data and offer them to everyone interested



Scalable *m*:*n*-group communication

subscribe(F_n)

publish(n)

What are Cyber-Physical Systems?

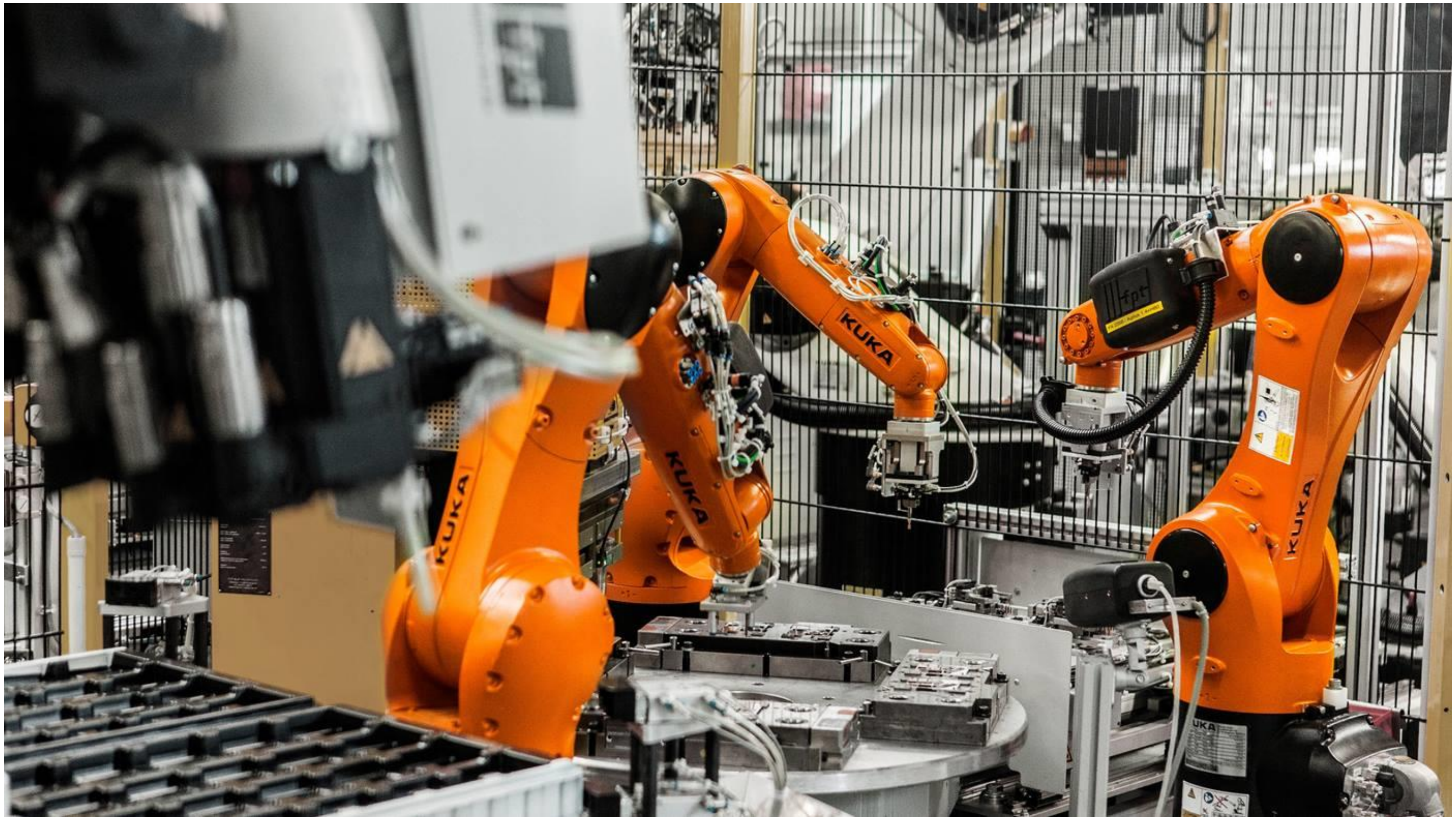
- Systems containing software components and mechanical or electronic parts that are interconnected via network
- Interact with the real, physical world
  → are subject to physical laws
  → have requirements w.r.t (real) time
- Examples
  - Industry robots
    - Production line in the smart factory
    - Reconfigurable production cell of a smart factory
  - Modern (autonomous) vehicles
    - Steer/fly by wire
    - Autopilots of any kind

Industry Robots in a Smart Factory

Industry robots made by Kuka

Time-critical communication when handing over work pieces.

Reconfigurable Production Cell

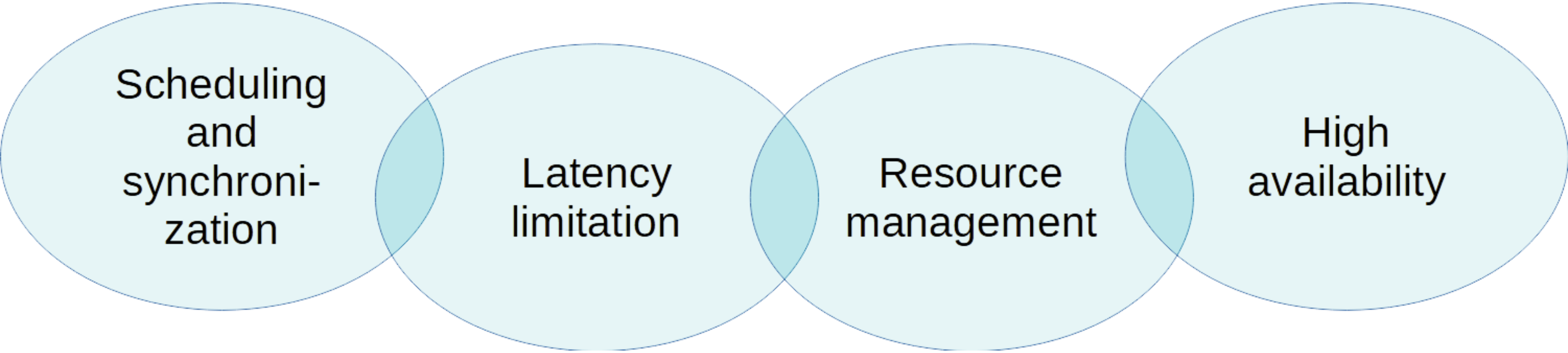


Industry robots made by Kuka

Flexible communication in case of task changes.

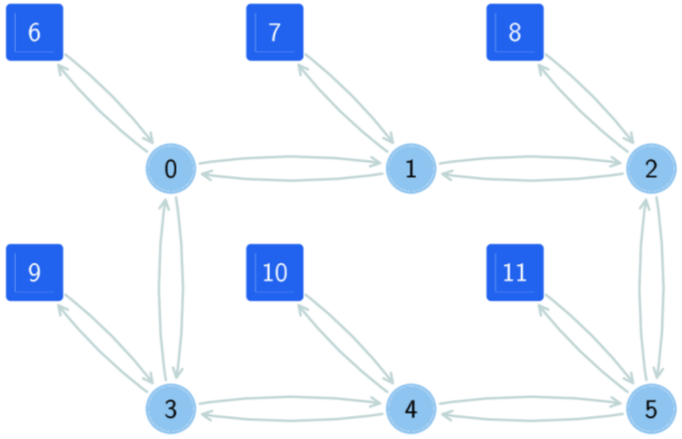
Realtime Ethernet: Time-Sensitive Networking

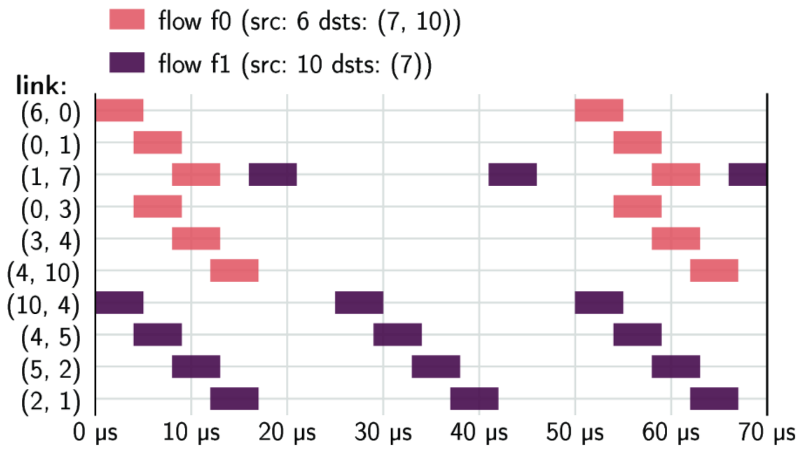
- Wired communication according to IEEE 802.1Q
- Extension of Ethernet
- Adaptation to ISO/OSI layer 2 (data link layer)
- Modular principle



- TSN networks must be configured: we need scheduling for this!

Communication Schedule





- Streams
  - From node 6 to nodes 7 and 10 (multicast)
  - From node 10 to node 7
- Schedule
  - Determines exactly when which packet is sent over which link
  - Has to be always without conflicts → provable correct
  - Needs to be adapted whenever communication pattern changes
  - Additional traffic of lesser importance is possible

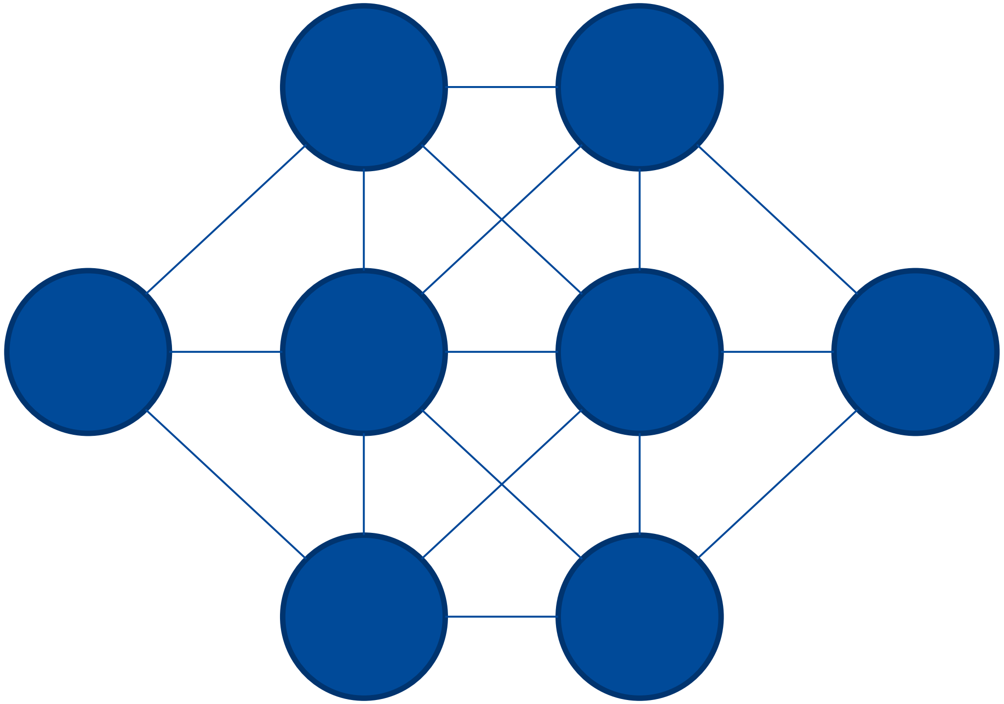
Constraint-based Programming

- Network communication schedules are often calculated using constraint-based programming
    - For example: Integer Linear Programming (ILP)

- User describes the problem
    - Provides condistions to met → Constraints
    - Provides an objective function → (linear) optimization

- Computer/solver solves the problem instance
    - Finding of variable assignemnets that
        - fulfill the constraints
        - optimizes the objective function

Puzzle



Distribute numbers 1 to 8 on the nodes such that adjacent nodes have no consecutive numbers!

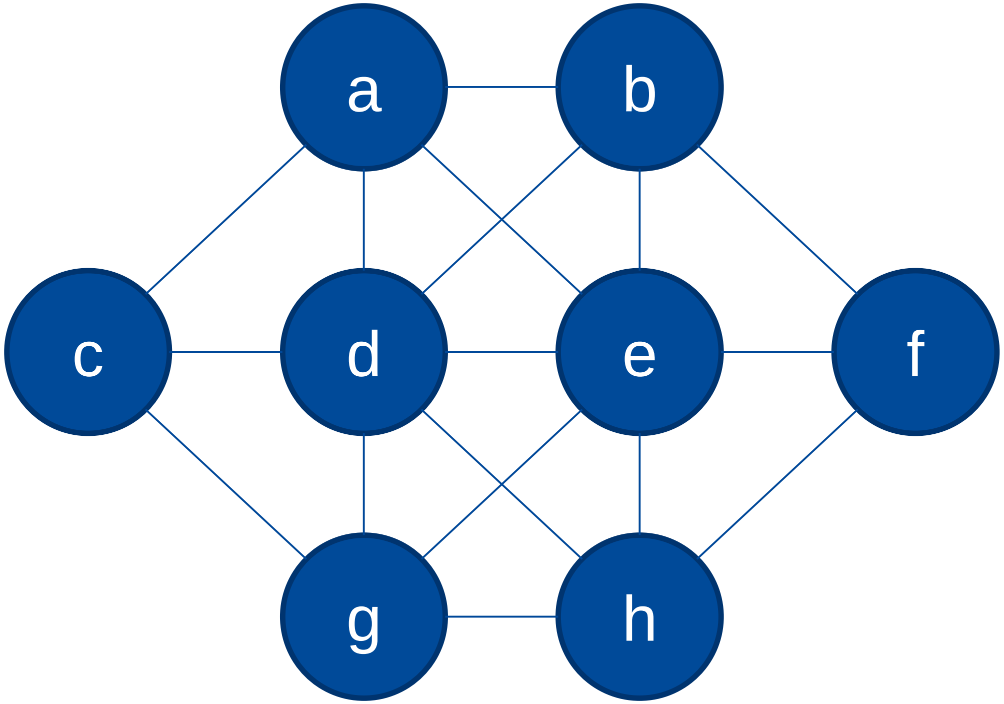
Solution Approaches

- Extensive search
  - Simply test all assignments
  - Computers can test many assignments very fast
  - But search space can be vast even for computers

- Heuristics
  - Identify nodes with the most restrictions
  - Identify numbers with the fewest restrictions
  - Logical inference and propagation

- Constraint-based Programming
  - Modelling of restrictions as constraints
  - Input for a corresponding solver

Modelling with Constraints (i)



Variables and constraints.

Modelling with Constraints (ii)

Different values

- $a \neq b, a \neq c, a \neq d, \ldots$
- $b \neq c, b \neq d, \ldots$
- $\ldots$
- $g \neq h$

Neighboring nodes have no consecutive values

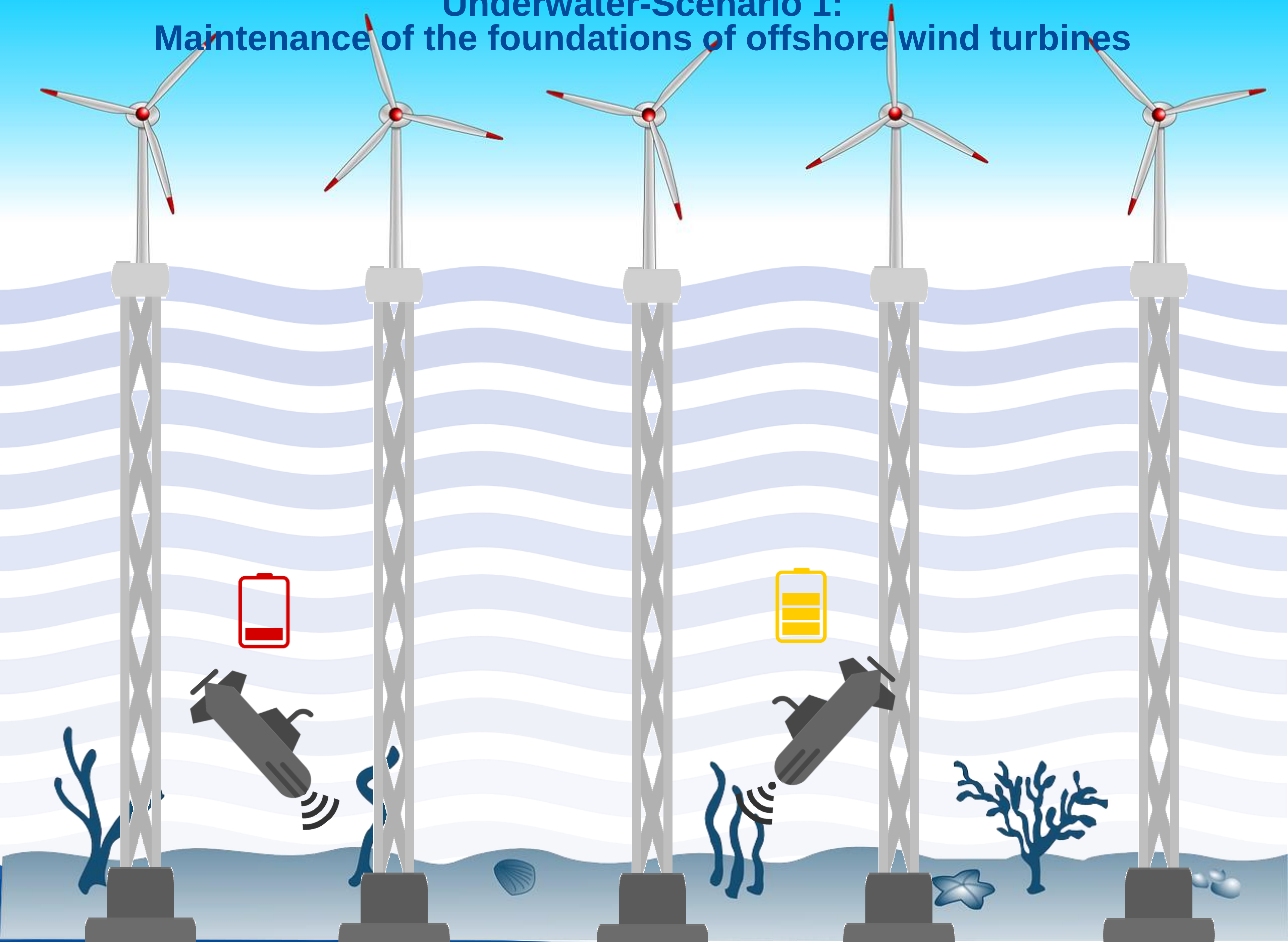
- $|a - b| \geq 2$
- $|a - c| \geq 2$
- $\ldots$

Thereafter simply Solver.solve()!

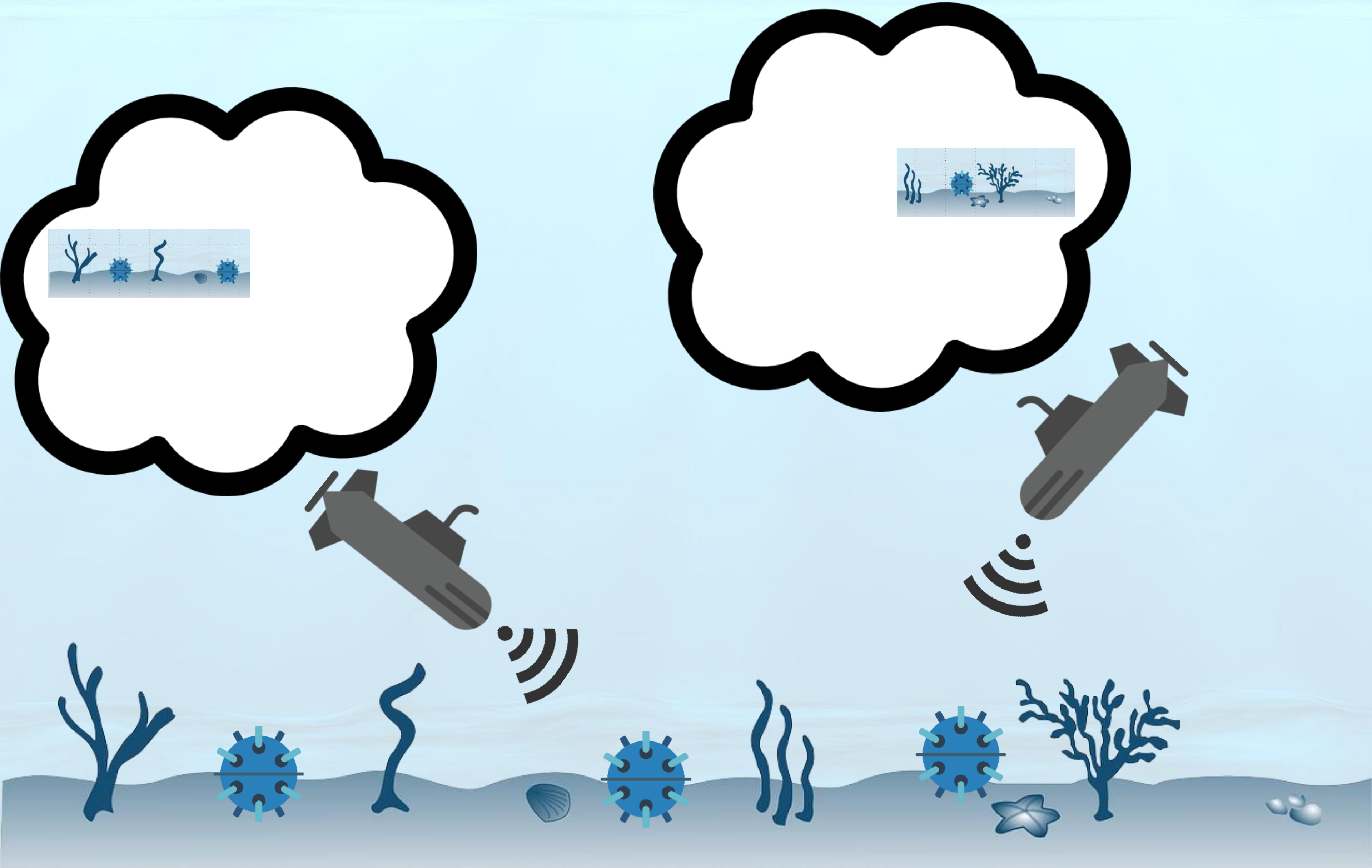
Wireless Communication

- Communication diversity in CPS
  - Combination of wired stability and wireless flexibility to meet various application requirements
- 5G as a key factor
  - Ultra-low-latency and highly reliable wireless communication for mobile and dynamic CPS applications
- Challenges posed by 5G
  - Security risks, interoperability issues and energy efficiency as key topics for integration in CPS

Underwater-Scenario 1:
Maintenance of the foundations of offshore wind turbines



Underwater-Scenario 2:
Clearance of Unexploded Ordnance (UXO) from World War II



Projects and Collaborations

- Realtime publish/subscribe communication
  - Part of a DFG project
  - Planning of flexible communication patterns and reservation of required time slots on communication links
  - Formal models and methods for scheduling
  - Estimation of the worst case runtime for publishing and filtering (content-based if necessary) a notification
  - Application scenario within a smart factory
- Autonomous Underwater Vehicles (AUVs)
  - Cooperation with the Institute for the Protection of Maritime Infrastructures, Resilience Department of Maritime Systems, German Aerospace Center (DLR) Bremerhaven
  - Cooperative navigation of several AUVs
  - Limited Energy restricts movement and usage of sensors
  - Opportunistic communication via acoustic modems

Tasks: TSN Standards in OMNeT++

- OMNeT++ framework INET includes simulation models for TSN standards
- INET offers various TSN showcases to demonstrate the functionality of these standards

- Task TSN.1: Showcases
  - Commissioning of a showcase
  - Reproduction of the showcase in your own use case
  - Integration of missing TSN features into the simulation models if necessary
- Task TSN.2: Tutorial
  - Design a tutorial for a scientific conference
  - Theoretical explanation and practical demonstration of a TSN standard

Tasks: 5G Communication in OMNeT++

- Simu5G: Simulator for 5G NR and LTE/LTE-A networks, integrated in OMNeT++ and INET
- Functions: Simulation of the data plane in 5G RAN and core network with FDD/TDD, heterogeneous gNBs, D2D communication and dual connectivity

- Task 5G.1: Showcases
  - Commissioning of Simu5G
  - Analysis and testing of Simu5G
  - Development of own showcases

Tasks: Autonomous Vehicles in OMNeT++

- Use of an OMNeT++ simulation model for autonomous vehicles (AVs)
- Simulation model has a modular structure and can be easily adapted

- Task AV.1: Energy model
  - Commissioning of the simulation model in OMNeT++
  - Analysis of various energy models and consumption curves
  - Integration of an energy model
- Task AV.2: Cooperative missions
  - Formation of several AUVs
  - Mapping of the seabed
  - Cooperative hunting

Tasks: ILP/Constraint Programming

- Computation of TSN schedules using ILP solvers
  - Modelling as an ILP problem
  - Programming the solver → input, equations, output, …
  - Optimization and evaluation

- Gurobi as ILP solver
  - Industry standard for linear optimization
  - Commercial tool with academic license
  - Possible usage of cloud instances

- Programming in Python
  - Python for preparing/processing input and output, respectively
  - Python as programming language for Gurobi
    → constraints and equations

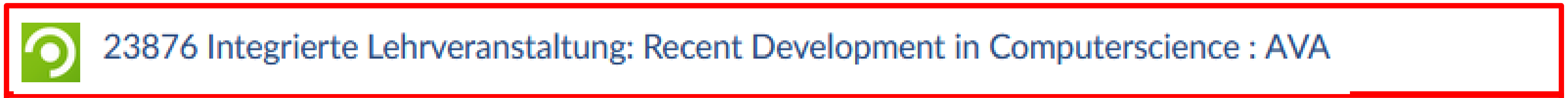
Organizational Matters

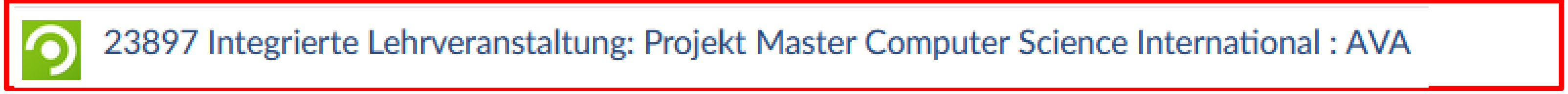
- Weekly meeting on Thursdays at 11:00 am in SR 014 (AE26)
- Up to two teams
  - Team A: TSN, 5G, and AUVs (Peter)
    (probably more fine-grained distribution of tasks)
  - Team B: TSN Linux Host (Helge)
- Design methodology
  - Agile development
  - Three milestones w.r.t. design, implementation, documentation

Type and size/scale of tasks depends on number
and interests of participants!

Registration and Contact

- Enrolement in respective Stud.IP course

1. 
23876 Integrierte Lehrveranstaltung: Recent Development in Computerscience : AVA

2. 
23897 Integrierte Lehrveranstaltung: Projekt Master Computer Science International : AVA

- Questions via email to Peter Danielis and Helge Parzyjegla
  - peter.danielis@uni-rostock.de
  - helge.parzyjegla@uni-rostock.de