

AUTONOMOUS SYSTEM MODEL DRIVEN DEVELOPMENT PROJECT

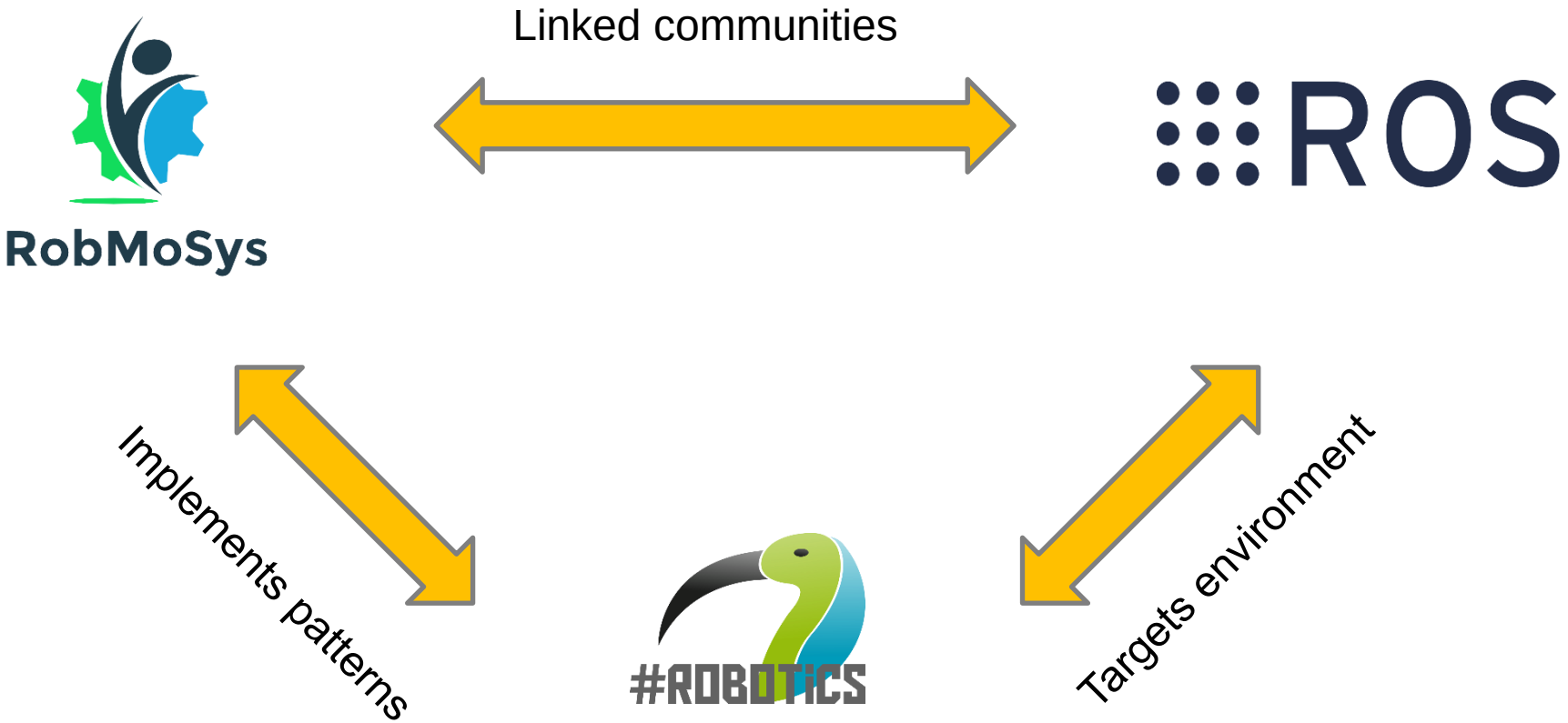
FROM RESEARCH TO INDUSTRY
cea tech

Shuai Li, CEA List

Robotics development context

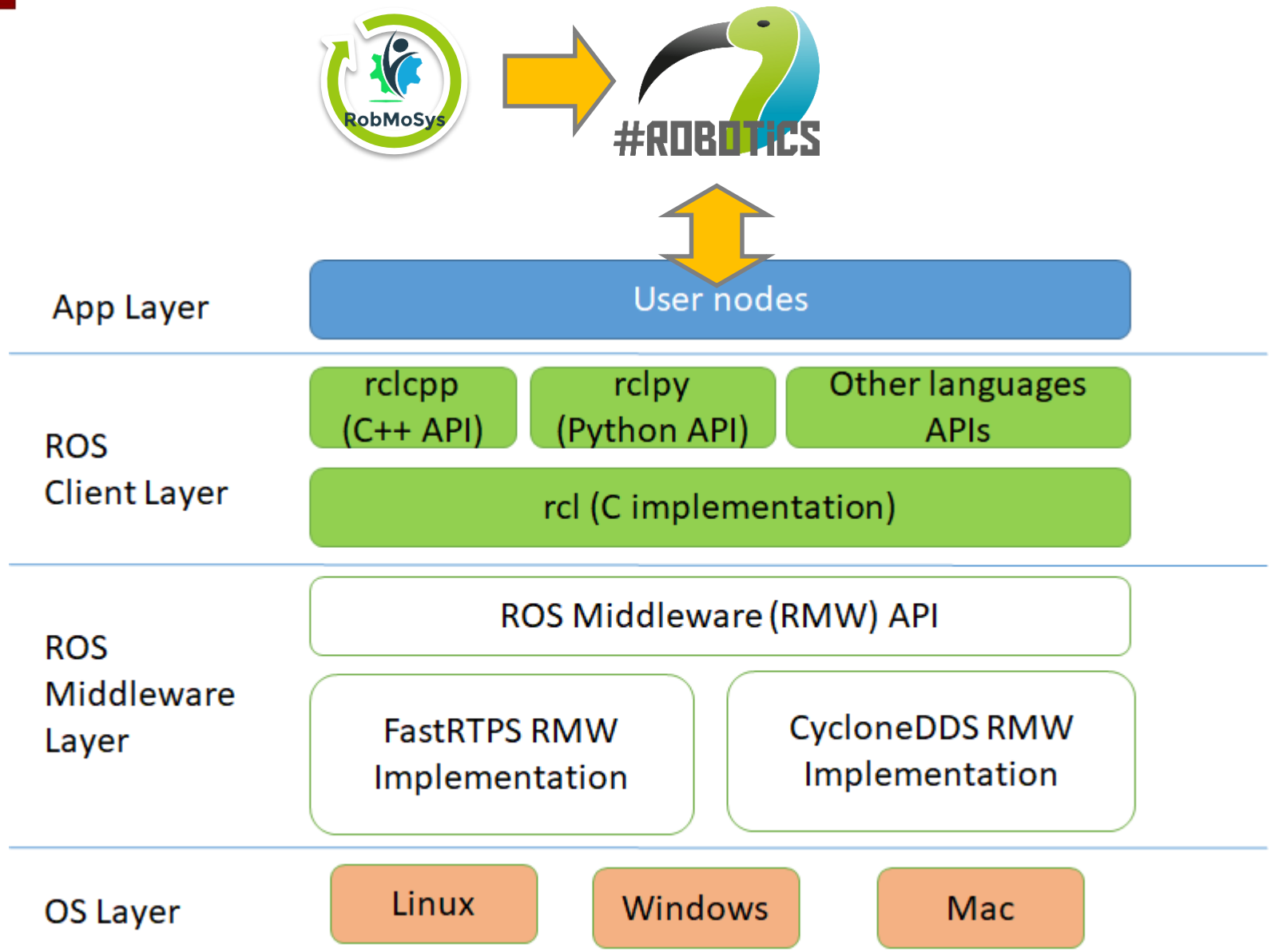
Robotics architectural and methodological patterns

Communication middleware and libraries for robot development



MBSE tools for robot development:
Papyrus 4 Robotics (P4R)

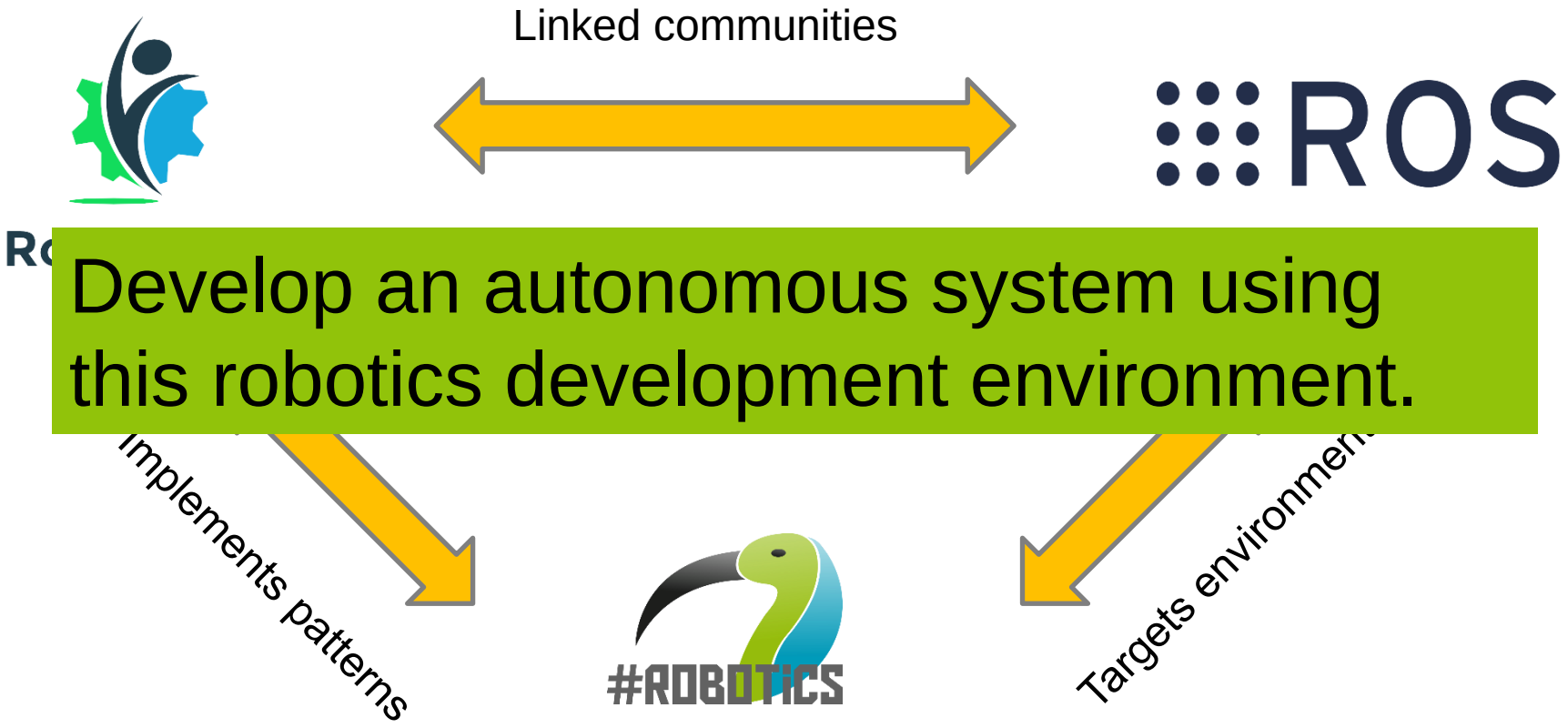
Layered architecture



Project technical objective

Robotics architectural and methodological patterns

Communication middleware and libraries for robot development



MBSE tools for robot development

Case-study

- Advanced Driver-Assistance System (ADAS)
- Sensors: stereo cameras, odometer, lidar
- Detections: speed, lane, pedestrian, road, traffic sign, depth, obstacle, data fusion
- Actuations: speed control
- Supervision: information display
- Demonstrator shown at CES 2019 to showcase future vehicle E/E architecture and SDK developed for the Renault-Nissan-Mitsubishi Alliance.

RENAULT NISSAN MITSUBISHI

Roles

P4R tool provider

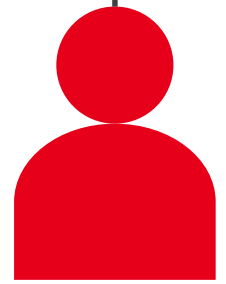
Methodology and development tools

Public knowledge

search

ADAS Project

Domain knowledge



Domain experts (Students)

query

System architect (CEA expert)

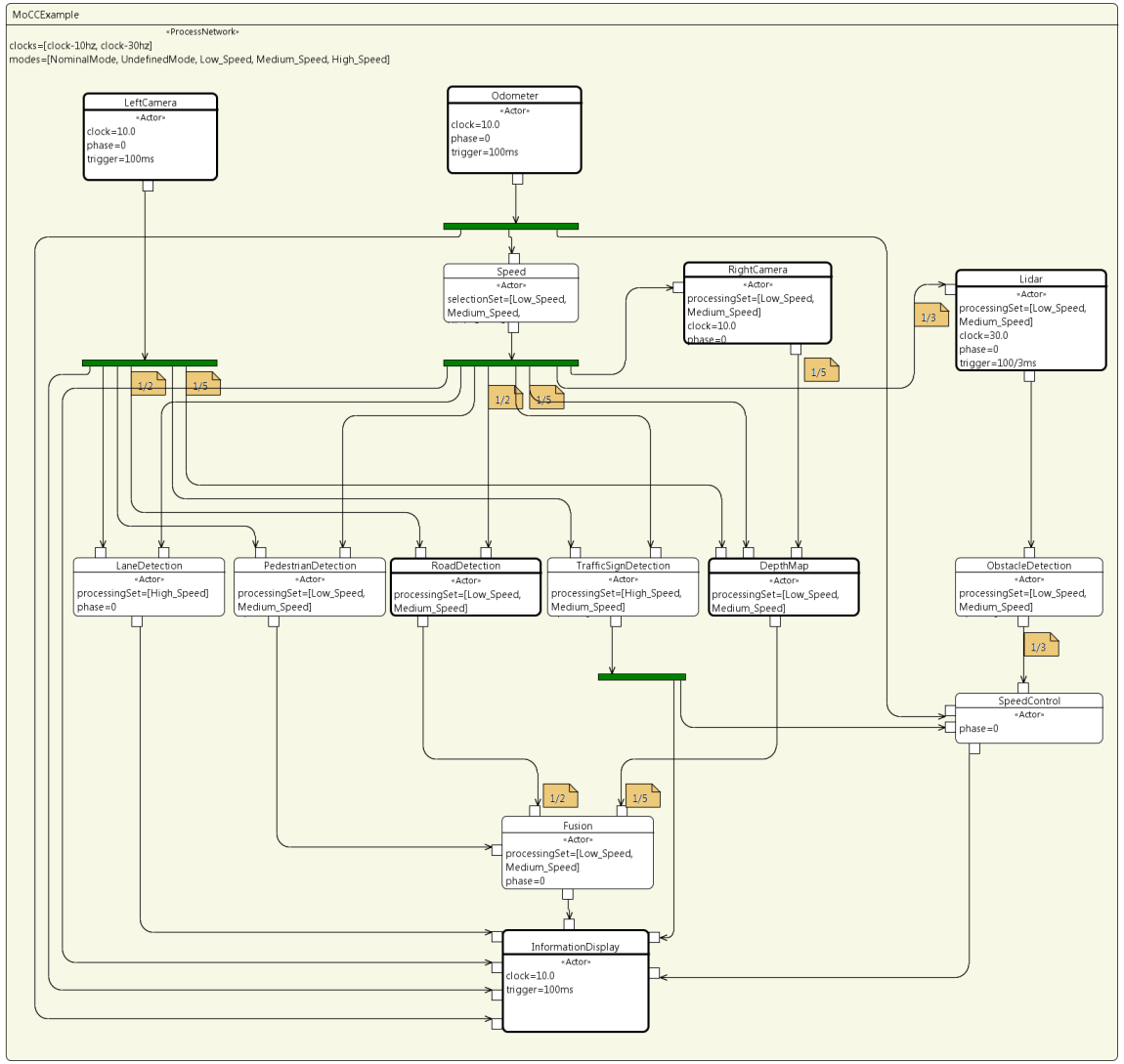
MBSE expertise

search

ROS2 & Gazebo knowledge

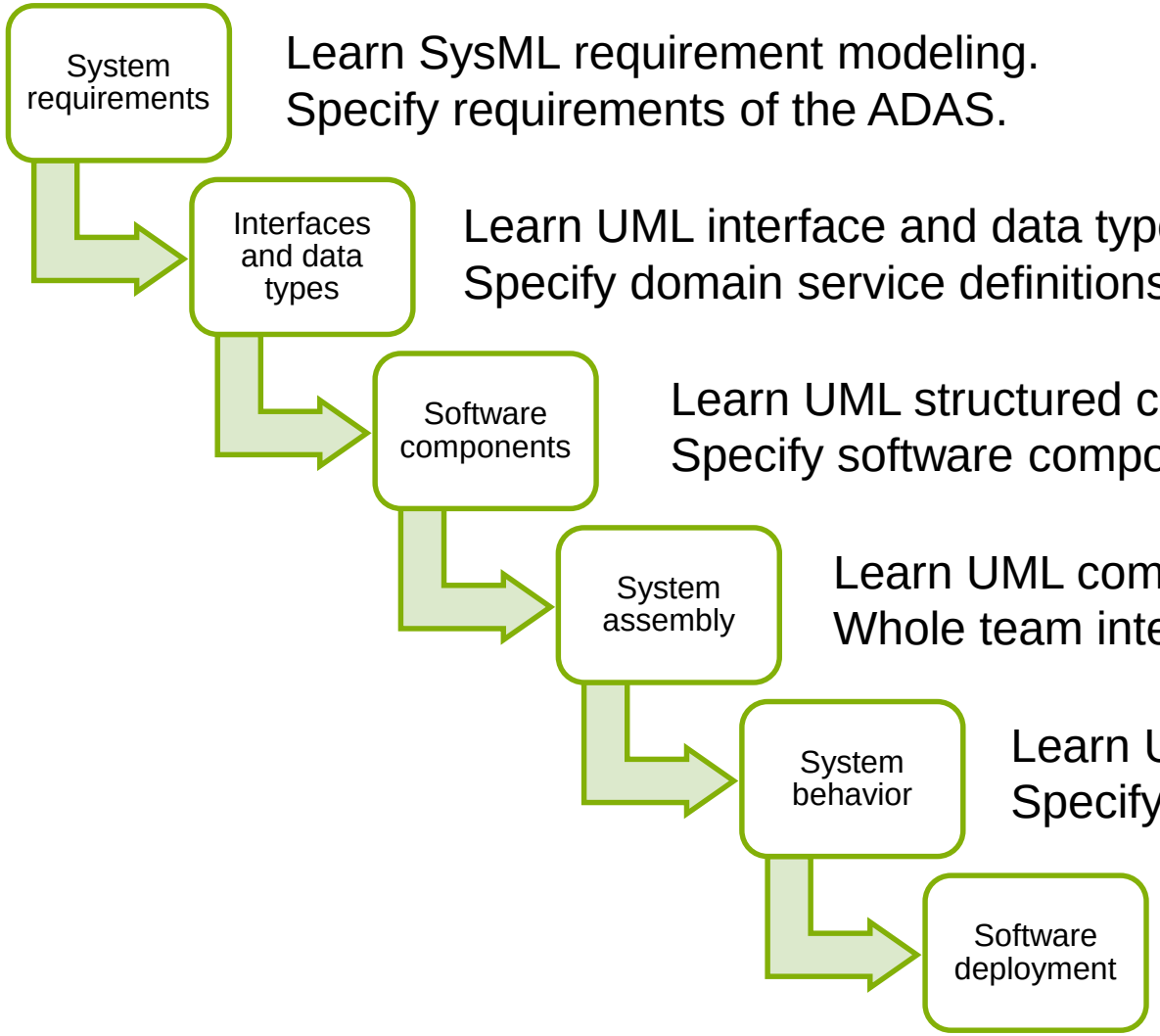
search

Teams



- **Each team will develop one set of components**
- **Components are assembled into a system.**
- **Teams will challenge each other's design when integration problems occur.**
- **Team 1:**
  - LeftCamera, LaneDetection, PedestrianDetection
- **Team 2:**
  - Odometer, Speed, RoadDetection, TrafficSignDetection
- **Team 3:**
  - RightCamera, DepthMap, Fusion
- **Team 4:**
  - Lidar, ObstacleDetection, SpeedControl, InformationDisplay
- **Personal re-allocation is permitted during the project**

Timeline


System requirements
Learn SysML requirement modeling.
Specify requirements of the ADAS.
Interfaces and data types
Learn UML interface and data type modeling.
Specify domain service definitions and skills.
Software components
Learn UML structured class modeling.
Specify software components by each individual team.
System assembly
Learn UML composite structure modeling.
Whole team integrates all individual software components.
System behavior
Learn UML behavior modeling.
Specify AI of system as behavior tree.
Software deployment
Learn real-time software analysis and implementation.
Analyze schedulability and generate code to complete.

Module pedagogical objectives

1. **Gain general knowledge of Model-Based System Engineering (MBSE) and apply them in a real project that uses MBSE concepts**
   - SysML concepts
   - UML concepts
   - Architecture paradigms: Service Oriented Architecture (SoA), Component-Based Modeling (CBM), Object-Oriented Programming (OOP)
2. **Develop domain knowledge and practical skills in a project-oriented environment:**
   - Eclipse Integrated Development Environment (IDE)
   - Eclipse Modeling Framwork environments (including Papyrus)
   - ROS2 concepts and runtime setup
   - Gazebo simulator setup and usage
   - Auto-didactical gain of domain knowledge related to information flow and algorithms in autonomous systems
   - …
3. **Evaluate a MBSE robotics development environment**
   - Evaluation of RobMoSys methodological and architectural patterns
   - Evaluation of Papyrus 4 Robotics development tools

- **Scoring: 60% common score based on project success. 40% individual team score based on defense at the exam.**

One typical session

- **Agenda**
  - Short lecture on MBSE topic
  - Tutorial with screen share
  - Workshops of individual teams
  - Whole team discussions, peer help, and cross-reviews
- **Remote collaboration tools in the Covid-19 context**
  - Skype for lecture audio and screen sharing
  - Discord for persistent information, questions, and virtual meeting rooms.
    - https://discord.gg/ny62RdKEra
    - This is your comm. tool. Use it however your like!
- **Artefact sharing**
  - https://github.com/sli88/adas-p4r
  - No more model and code sharing by email please!