TAMPERE UNIVERSITY OF TECHNOLOGY

FUNDAMENTALS OF MOBILE ROBOTS

Exercise session 1

September 5, 2018

Tuomas Välimäki

tuomas.s.valimaki@tut.fi

Laboratory of Automation and Hydraulics
Tampere University of Technology

EXERCISE STRUCTURE

Week 36	Week 37–40	Week 41, 43–45	Week 46–49
Introduction	Exercise 1	Exercise 2	Exercise 3
ROS & Robotic System Toolbox	Briefing ex1	Briefing ex2	Briefing ex3
	QA session 1	QA session 1	QA session 1
	QA session 2	QA session 2	QA session 2
	Feedback	Feedback	Feedback

PRACTICALITIES

- → no office hours
- → you can post questions for the QA sessions at all times to Moodle
- → new exercises will be made available after the previous feedback session, but won’t be briefed until next week

Course grading

- → 50% exercises
- → 30% exam
- → 20% commitment

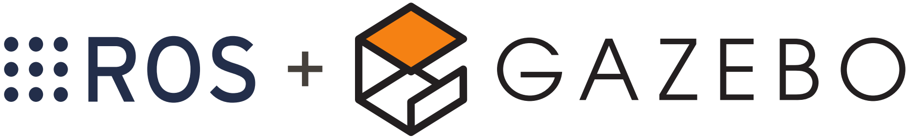
EXERCISE GROUPS

- → new 3–4 person group for each exercise
- → self organized group for 1st exercise
- → randomized groups for others

!

If you’re not allocated to any group but wish to complete the course, you must notify Risto or Reza before next weeks exercises.

TOOLS



ENVIRONMENT

- → Matlab on Windows + virtual machine running Linux
- → available in PC rooms SB202, TB205, and K1???

- → own laptops can be used
- → code needs to run in provided environment
- → **no support**

C++/ PYTHON

Can be used with certain **limitations**:

1. you will get no base code and will have to implement everything yourselves
2. the exercises require not only implementation but also analysis of the results, including plots etc. If your language of choice does not support this (C++) you will have to record the data and do the analysis using other languages
3. no matter the language, the code has to be clear and well documented

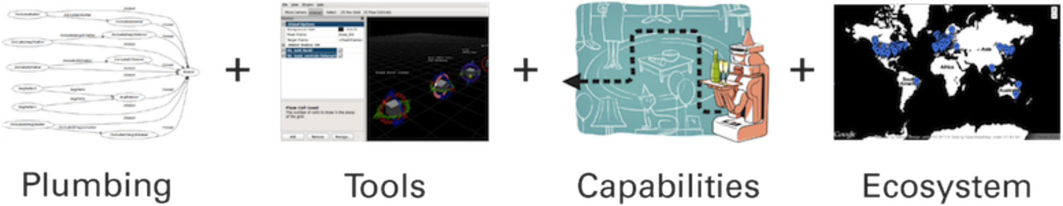
ROS — Robot Operating System

For background information on ROS, self study:

🌐 ETH Zürich. *Programming for Robotics*. URL: **http://www.rsl.ethz.ch/education-students/lectures/ros.html**.

→ ROS architecture & philosophy
→ ROS master, nodes, and topics
→ Console commands
→ Launch-files
→ ROS subscribers and publishers

WHAT IS ROS?



Plumbing

- process management
- communication
- device drivers

Tools

- simulation
- visualization
- logging

Capabilities

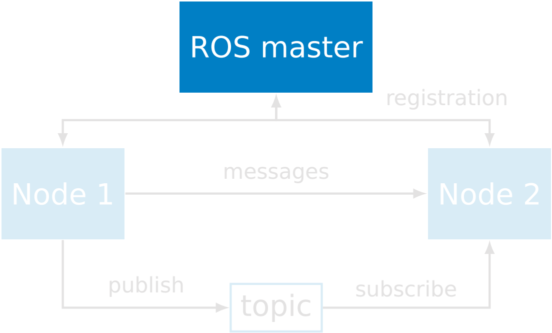
- control
- planning
- mapping

Ecosystem

- software distribution
- tutorials
- documentation

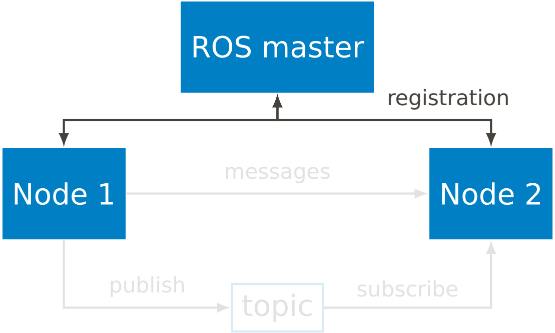
ROS MASTER

- → provides naming and registration to ROS system
- → manages communication between nodes
- → provides a parameter server



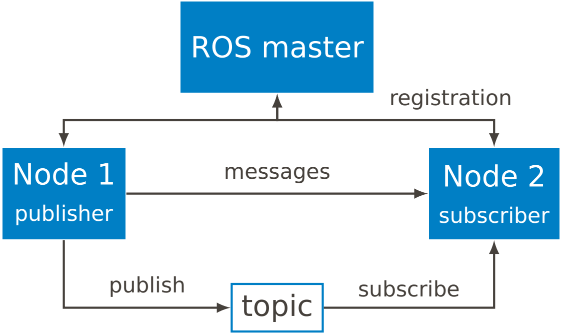
ROS NODES

- → single-purpose process to perform computation
- → multiple nodes combined form a graph
- → a control system will typically comprise multiple nodes
- → organized in packages



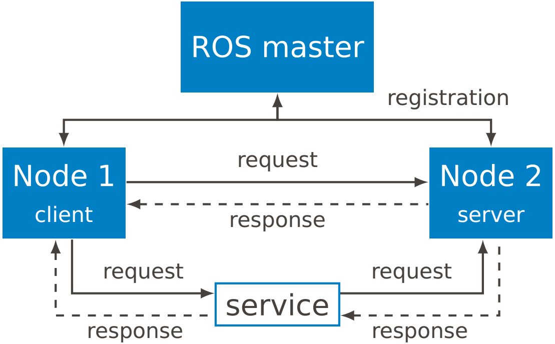
ROS MESSAGES

- nodes communicate over topics
  - nodes can either publish or subscribe
  - typically 1 publisher and *n* subscribers
- topic is a name for a stream of messages
- messages comprise of a nested structure of datatypes and objects
  - defined in *.msg files



ROS SERVICES

- → allows request/response communication
  - → service server advertises the service
  - → service client accesses the service
- → similar structure to messages
  - → defined in *.srv files



GAZEBO

- → simulation software with realistic physics
- → dynamics, sensors, etc.
- → provides a ROS interface

FURTHER READING

- Mathworks. *Robotics System Toolbox*. URL: **https://se.mathworks.com/products/robotics.html**.
- Open Source Robotics Foundation. *ROS Wiki*. URL: **http://wiki.ros.org**.
- Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. The MIT Press, 2005. ISBN: 978-0-262-20162-9.