CSCI 473/573
Human-Centered Robotics
Let’s design a simple robot
Let’s design a simple robot
Before implementing intelligence, we need to consider:

- Encoder
- Motors
- Control Pendant
- Robot Controller

Motor controllers, Inverse Kinematics, GUI, Collision Detection
Trajectory Planner, PID Controller, Error Monitoring
Example: Manipulation
ROS: Robot Operating System

What a mess!

How can we deal with all program components?
ROS: Robot Operating System

ROS is an open-source, meta-operating system
ROS: Robot Operating System

ROS is an open-source, meta-operating system

MoveIt!

ROS

OpenCV

pointcloudlibrary
ROS: Robot Operating System

- Modular design
- Reusable robotics components!
- 62 Robotic platforms officially support ROS on 9/18/2015, 121 on 1/30/2017, and 150+ on 1/22/2019: [http://wiki.ros.org/Ros/Robots](http://wiki.ros.org/Ros/Robots)

Now: (unofficially) almost all open source robots are using certain components of ROS...

- Hundreds of ready to use algorithms
- Efficient, so it can be used for actual products, not just prototyping
- Runs on Ubuntu, also many embedded processors
- Parallelization and networking made easy, can use multiple machines simultaneously
Current Robotics Job Ads

A relevant degree is required, for instance in Computer Science or Engineering. A background in Robotics/Computer Vision is desirable, while knowledge of the Robot Operating System (ROS), the Point Cloud Library (PCL), or the Open Source Computer Vision Library (OpenCV) is a big plus.

Goal of this PhD is to study, design and build novel industry-level software based on ROS or ROS-Industry which is modular, reconfigurable, adaptive, easy to use to integrate and control various robotic systems.

The candidate should be knowledgeable about C/C++, ROS and matlab/simulink. Scientific curiosity, large autonomy and ability to work independently are also expected.

The candidate must be a proficient user of C/C++ and ROS and any relevant computer vision library (e.g., ViSP, OpenCV, PCL). Scientific curiosity, large autonomy and ability to work independently are also expected.

* Nice To Haves: *
- Experience with Robot Operating System (ROS).
- Experience with OpenCV and/or PCL.
- Strong background in machine learning.

Required Skills
* MSc in Engineering / Computer Science or equivalent.
* Experience with Robotics
* Knowledge about ROS (Robot Operating System) and CV.
* Advanced experience with C++ and soft real-time programming.
* Team spirit and ability to work independently.
* Excellent communication skills, flexibility and creativity.
ROS: Robot Operating System
ROS: Robot Operating System

Celebrating 5 years of ROS
ROS: Robot Operating System

CELEBRATING EIGHT YEARS
ROS: Robot Operating System

Presented in 2009 by Willow Garage is a meta-operating system for robotics with a rich ecosystem of tools and programs.
ROS: Robot Operating System

**Plumbing**: ROS provides messaging infrastructure designed to support the quick and easy construction of distributed computing systems.
Nodes

Nodes are processes that perform computation, “executables”

- Camera
- Motion Planner
- Mapping
Topics

Topics are streams of data with publish / subscribe semantics. They are uniquely identifiable by its name.
Messages

A message is simply a data structure, comprising typed fields. Language agnostic data representation. C++ can talk to Python.

File: pos.msg

- string robotName
- uint32 posX
- uint32 posY
- uint32 goalX
- uint32 goalY
Services

Request / reply is done via services, which are defined by a pair of message structures: one for the request and one for the reply.
ROS Master

The ROS Master provides name registration and lookup to nodes. Without the Master, nodes would not be able to find each other, exchange messages, or invoke services.
ROS: Robot Operating System

Computer 1

ROS Master

Registration

Registration

ROS Node 1

Messages

Messages

ROS Node 2

ROS Node n

Messages
ROS: Robot Operating System

Computer on the Robot

- ROS Master
  - Registration
  - Camera Node
    - Data
  - Image Processing Node

Laptop

- Image Display Node
  - Registration
ROS: Robot Operating System

Computer on the Robot

ROS Master

Registration

Camera Node

Request

/image_data Message

Reply

Registration

Image Processing Node

Reply

Data

Camera
Tools: ROS provides an extensive set of tools for configuring, starting, introspecting, debugging, visualizing, logging, testing, and stopping distributed computing systems.
ROS: Robot Operating System

System Visualization: rqt_graph
ROS: Robot Operating System

System Visualization: rqt_graph
ROS: Robot Operating System

Live Plotting: rqt_plot
ROS: Robot Operating System
Logging and Visualization Sensor Data: rosbag and rqt_bag
ROS: Robot Operating System

Logging and Visualization Sensor Data: rosbag and rqt_bag
ROS: Robot Operating System
3D Visualization: RVIZ
ROS: Robot Operating System
2D and 3D Simulation: Player-Stage and Gazebo
ROS: Robot Operating System
2D and 3D Simulation: Player-Stage and Gazebo
ROS: Robot Operating System

2D and 3D Simulation: Player-Stage and Gazebo
**Capabilities:** ROS provides a broad collection of libraries that implement useful robot functionality, with a focus on mobility, manipulation, and perception.
Capabilities: Manipulation
Capabilities: Mobility

Lots Of Nice Steps
Capabilities: Mapping
ROS: Robot Operating System

**Ecosystem**: ROS is supported and improved by a large community, with a strong focus on integration and documentation. ros.org is a one-stop-shop for finding and learning about the thousands of ROS packages that are available from developers around the world.
ROS: Robot Operating System

Worldwide User Base (ros.org)
ROS: Robot Operating System

<table>
<thead>
<tr>
<th>Distro</th>
<th>Release date</th>
<th>Poster</th>
<th>Turtle, turtle in tutorial</th>
<th>EOL date</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROS Melodic Morenia</td>
<td>May, 2018</td>
<td>TBD</td>
<td>TBD</td>
<td>May, 2023</td>
</tr>
<tr>
<td>ROS Lunar Loggerhead</td>
<td>May 23rd, 2017</td>
<td>TBD</td>
<td>TBD</td>
<td>May, 2019</td>
</tr>
<tr>
<td>ROS Kinetic Kame</td>
<td>May 23rd, 2016</td>
<td>TBD</td>
<td>TBD</td>
<td>April, 2021 (Xenial EOL)</td>
</tr>
<tr>
<td>ROS Jade Turtle</td>
<td>May 23rd, 2015</td>
<td>TBD</td>
<td>TBD</td>
<td>May, 2017</td>
</tr>
<tr>
<td>ROS Indigo Iguana</td>
<td>July 22nd, 2014</td>
<td>TBD</td>
<td>TBD</td>
<td>April, 2019 (Trusty EOL)</td>
</tr>
<tr>
<td>ROS Hydro Medusa</td>
<td>September 4th, 2013</td>
<td>TBD</td>
<td>TBD</td>
<td>May, 2015</td>
</tr>
<tr>
<td>ROS Fuerte Turtle</td>
<td>April 23, 2012</td>
<td>TBD</td>
<td>TBD</td>
<td>–</td>
</tr>
<tr>
<td>ROS Electric Emys</td>
<td>August 30, 2011</td>
<td>TBD</td>
<td>TBD</td>
<td>–</td>
</tr>
<tr>
<td>ROS Diamondback</td>
<td>March 2, 2011</td>
<td>TBD</td>
<td>TBD</td>
<td>–</td>
</tr>
<tr>
<td>ROS C Turtle</td>
<td>August 2, 2010</td>
<td>TBD</td>
<td>TBD</td>
<td>–</td>
</tr>
<tr>
<td>ROS Box Turtle</td>
<td>March 2, 2010</td>
<td>TBD</td>
<td>TBD</td>
<td>–</td>
</tr>
</tbody>
</table>
ROS: Robot Operating System

- ROS is a meta-operating system for robotics
- Provides basic (and many!) algorithms for robotics
- Modular approach allows easy adaptation to hardware changes and both hw and sw updates
- Effective visualisation and simulation tools
- World-wide spread in research and commercial use
- BSD license - open source, free to use!
- Over 150 robot platforms support ROS, and growing!
- Easy to start
- Linux based, best works on Ubuntu
- Easy to parallelise, nodes based approach communicate over TCP and can be synchronised using timestamps for messages

Plumbing + Tools + Capabilities + Ecosystem
ROS: Robot Operating System

SpotMini: https://www.youtube.com/watch?v=tf7IEVTDjng