Module 1 - ROS Overview

ME4140 - ROS Workshop

Mechanical Engineering

Tennessee Technological University

Lecture 1 - ROS Overview

Lecture 1 - ROS Overview

- What is ROS?
- Benefits of ROS
- How does ROS work?
- Supported Hardware and Software
- Tutorial 1 - Virtualize Ubuntu

What is ROS?

- *The Robot Operating System (ROS) is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.* - ROS WIKI
- Open Source (Creative Commons) Software Framework for Robotics Development
- not what you may think when you hear *operating system*

What is ROS?

Where did ROS come from?

- Developed at Stanford (mid 2000s)
- Continued by Willow Garage (2007)

Where is ROS now?

- Maintained by an international community of developers (present)
- *The ROS ecosystem now consists of tens of thousands of users worldwide, working in domains ranging from tabletop hobby projects to large industrial automation systems.* - ROS WIKI
- We are going to use ROS Noetic in Ubuntu 20.04(timeline)

Benefits of ROS

ROS allows us to build upon the knowledge and capabilities of the current state of robotics technology and theory.

- Hardware/Software Compatibility
- Pre-Compiled Algorithms for Robotics
- Multi-threading, Parallel Processing, Distributed Computing
- Open Source Community (Creative Commons)

Benefits of ROS

Who is using ROS?

- Researchers
- Students
- Hobbyists
- Industry - *start ups* and *big business (?)*

Who can use ROS? You!

Benefits of ROS

Thought exercise: Think about designing a robot to move boxes from one location to another in a large room.

- Where do you begin?
- What does the robot look like?
- What major elements or components are required for the robot?

How does ROS work?

ROS is organized in a system of connected *nodes* which each node represents a different element or component in a robotic system.

- Laser
- Drive Kinematics
- Navigation
- Manipulator
- etc.

Supported Hardware and Software

Each node can have corresponding source code, executables, data files, and more. Different software languages are available.

- C++ (instructor support in ME4140)
- Python (you are on your own)
- *markup languages* such as XML and YAML (we may use some)

Pre-built software is avaiable for interfacing with different Robots, sensors, actuators, and other components. Also, ROS can run on small board computers with limited resources.

- Robots! (Adept, Clearpath, UR)
- LIDAR (SICK, RPLidar, etc.) and Cameras (webcam, Kinect, Opti-track)
- Motor Drivers (Roboteq, ROSARIA)

Tutorial 1 - Virtualize Ubuntu

- **Overview:** ROS runs on Linux! Your first exercise is to setup your computer so that you can begin learning ROS.
- **Assignment:** Complete the tutorial tutorial1_virtualize_ubuntu. After completion your new system should be able to access the internet.
- **Deliverable:** Write a one to two paragraph summary of what you accomplished and what you struggled with the most. Include an image of your Ubuntu desktop.
- **Next Week:** After completion of Module 1, you will be ready to install the ROS Noetic software package in Ubuntu 20.04. This is described in detail in Module 2.