CMPE 565

Introduction to ROS and V-REP

Robot Operating System

ROS stands for Robot Operating System.

- It is not an operating system but a framework, a collection of many robot-related programs.
- It provides:
  - A means of code reuse. Commonly used software is already implemented within ROS.
  - Package management. The structure that ROS uses to maintain the software is through packages. ROS allows us to manage them by installing, updating, removing; just like Unix distro.
  - A set of nice interfaces among several programming domains. You can create a Python package and a C++ package that work together by making them talk ROS.

Robot Operating System

- ROS supports:
  - Python
  - C++
  - Java
  - Lisp
  - Lua
- It has a set of Arduino libraries so that you can talk to your software that runs on the Arduino’s microprocessor.
- Most importantly, it provides distribution.
  - Several programs can run simultaneously and collaboratively via the message passing functionality ROS provides.

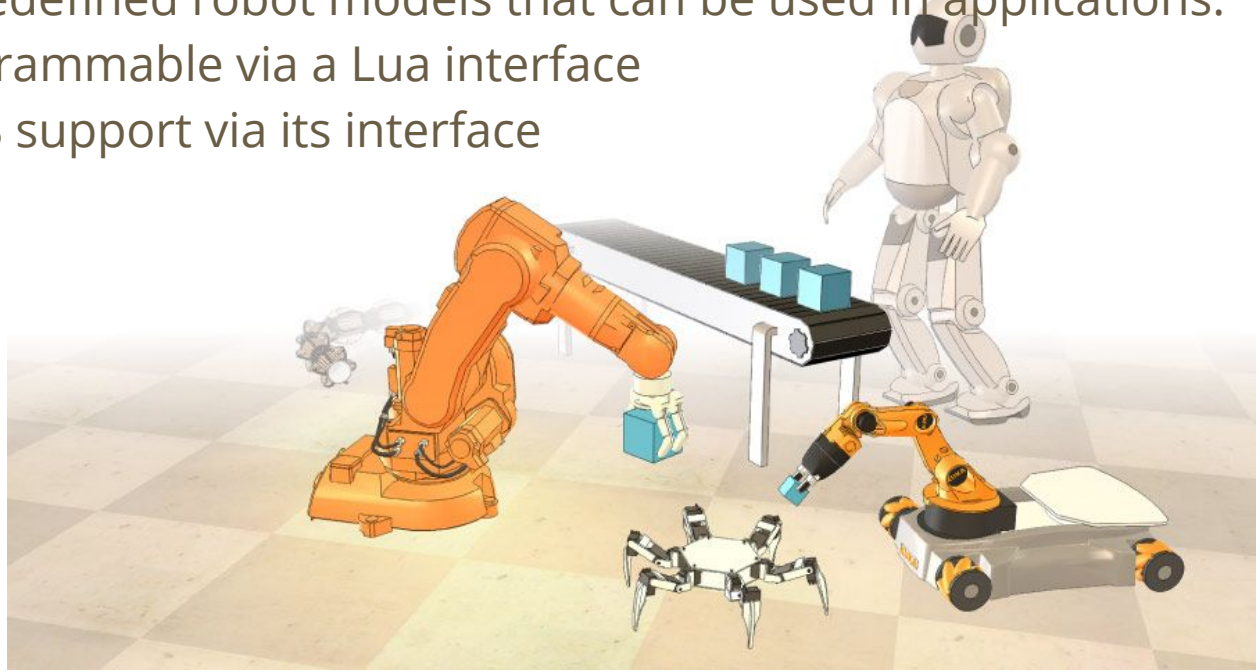
Robot Operating System

- ROS is developed for Unix-based systems, but Mac version not stable yet.
- ROS has an active community. You can find answers to your problems from answers.ros.org
- Its core code and packages are very well maintained and constantly revised.
    - Due to active development, there may be inconsistencies between versions.

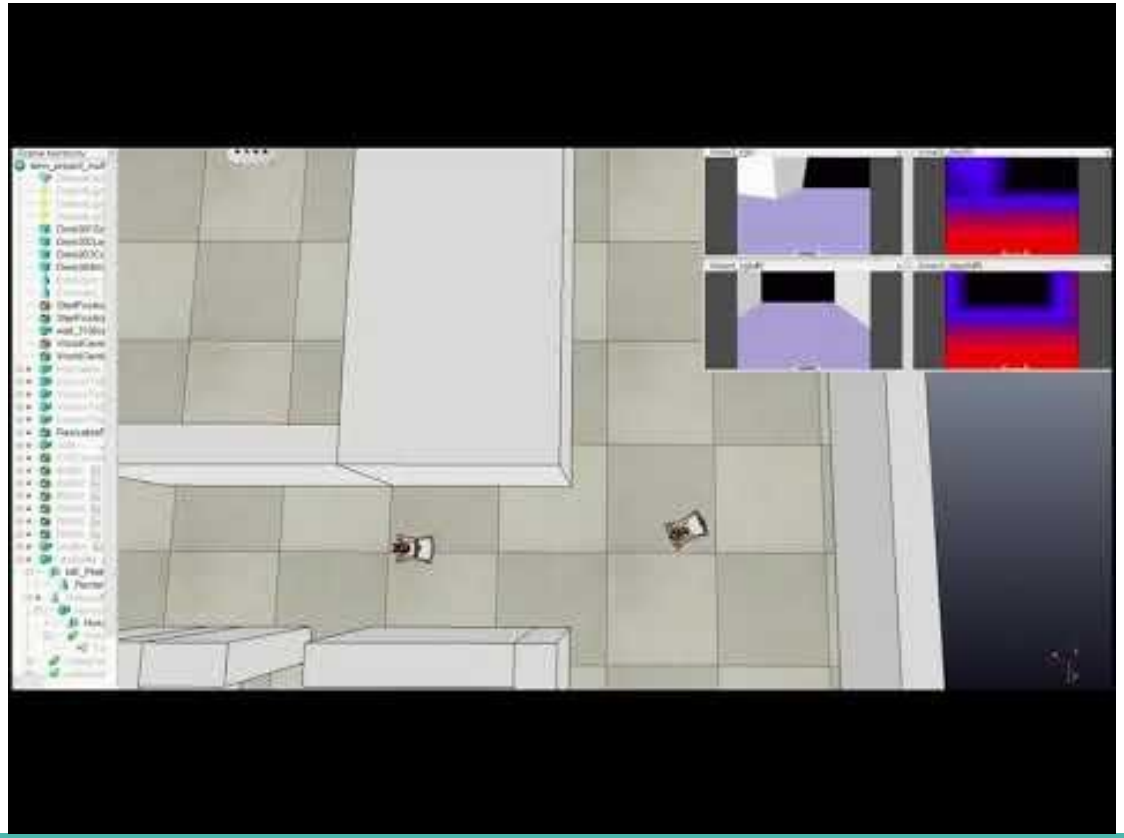
- We are going to use the latest stable LTS distribution: Kinetic Kame

V-REP

- V-REP is a robot simulator with 3 different physics engines.
- It has many predefined robot models that can be used in applications.
- It is easily programmable via a Lua interface
- It provides ROS support via its interface



Sample Term Project



First Assignment

- Install:
  a. Ubuntu 16.04.3 LTS, https://www.ubuntu.com/download/alternative-downloads
  b. ROS Kinetic Kame full-desktop version, http://wiki.ros.org/kinetic/Installation
  c. V-REP 3.4.0 Educational version http://www.coppeliarobotics.com/previousversions.html
- Clone our repos.
- Run the package that we provide to control the robot and run *rviz* to visualize the information flow.