Note of ROS Tutorial

Lectures by Dr. Pilwon Hur

August 17-18, 2016

1 Introduction to ROS

1.1 What is ROS?

This tutorial introduces a software platform called **Robot Operating System**, or **ROS**. The official description of ROS is as follows:

> *ROS is an open-source, meta-operating system for your robot. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. It also provides tools and libraries for obtaining, building, writing, and running code across multiple computers*

In general, a robot system consists of following components:

$$\text{Hardware} + \text{OS} + \text{Apps}$$

For users who want to focus on software (Apps) development, taking care of all the different types of communication between different components can be challenging. ROS, a flexible framework and also 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, provides the need for various types of communication as a platform.

1.1.1 Brief history

ROS was originally developed in 2007 under the name switchyard by the Stanford Artificial Intelligence Laboratory. From 2008 until 2013, development was performed primarily at Willow Garage, a robotics research institute/incubator. In February 2013, ROS stewardship transitioned to the Open Source Robotics Foundation. Table 1 shows the list of ROS versions developed so far. ROS is officially supported on Ubuntu. The current suggested version combinations for ROS and Ubuntu are Indigo + Ubuntu 14.04.5 LTS and Kinetic Kame + Ubuntu 16.04.1 LTS. We use the former option as the version choice through this tutorial.

Table 1: ROS versions

Released Date	Version
August 2, 2010	**C** Turtle
March 2, 2011	**D**iamondback
⋮	⋮
July 22, 2014	**I**ndigo
May 23, 2015,	**J**ade
May 23, 2016,	**K**inetic Kame

1.2 Be familiar with Ubuntu

1.2.1 Recommended version of Ubuntu and installation info

1.2.2 Recommended tools for programming in Ubuntu

1.2.3 Frequently used Linux commands

1.3 ROS installation and related setup

1.4 Turtlesim

2 Details of Programming in ROS

2.1 ROS system structure overview

2.2 First program with ROS: Hello World!

Let's create a catkin workspace:

```
$ mkdir -p  /catkin_ws/src   % create a directory
$ cd  /catkin_ws/src
$ catkin_init_workspace
```

2.3 Program demonstration with Kinect