Table of Contents

1. ROS Introduction

1.1. An introduction

As a requirement from the product owner, **ROS** is used as framework on Willy. ROS, the Robot Operating System, is a flexible software **framework** for use in robots. It consists of a collection of libraries, tools and conventions that provide basic infrastructure to communicate between different parts of the robot.

In the case of Willy, ROS is especially handy because Willy is made with a modular design. All modules can be removed without disrupting the other functionalities of Willy. For example, when the web interface is removed, Willy is still able to drive, but with another module as for example the keyboard controller. Or the removal of the motor driver makes Willy still able to interact with public.

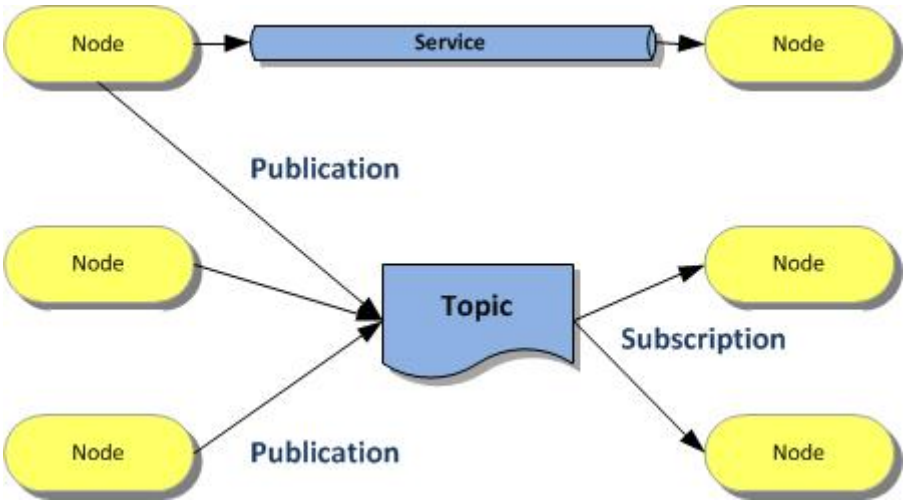
1.2. Nodes

A **node** in ROS can be seen as a module. It is an executable that communicates through ROS to other nodes to send and receive data. A node can be for example a c++ application, or a piece of Python code, or even an Arduino connected with USB running code. A piece of information a node receives or sends is called a **message**.

More information can be found at http://wiki.ros.org/ROS/Tutorials/UnderstandingNodes

1.3. Topics

A topic is a bus over which nodes can exchange data messages. A topic always has a name, so all topics can be identified.



More information can be found at http://wiki.ros.org/Topics

To interact with a topic, two methods are used, subscribing and publishing.

1.3.1. Subscribing

Subscribing is getting data from a topic. Everytime the data in a topic is updated, a message will be passed to all subscribing nodes. This way a node can use this information.

1.3.2. Publishing

Publishing is sending data to a topic. When a node has new information, a message will automatically be sent to the linked topic, so this data is updated.