

ROS Tutorials

The goal of this tutorial is to make a mobile robot of “two-wheel differential drive” run autonomously.

This tutorial consists of the following four parts.

1. ROS Wiki Introduction

(English , Japanese, partialy translated to Vietnamese)

2. ROS Wiki Tutorials - Beginner Level

(English , Japanese, partialy translated to Vietnamese)

3. Understanding ROS Navigation Stack with GAZEBO

(Sorry, Japanese-language only)

4. Understanding ROS Navigation Stack with mobile robot

(May be Japanese.) **This tutorial and the mobile robot is currently prepared.**

2018/11/14

K.Ishibashi

Yokoki Engineering Co.,Ltd.



1 ROS Wiki Introduction



Please refer to URLs bellow.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

2 ROS Wiki Tutorials - Beginner Level



1. [Installing and Configuring Your ROS Environment](#)

This tutorial walks you through installing ROS and setting up the ROS environment on your computer. We use “Ubuntu 16.06” and “ROS kinetic” distros.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

2. [Navigating the ROS Filesystem](#)

This tutorial introduces ROS filesystem concepts, and covers using the `roscd`, `rosls`, and `rospack` command line tools.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

3. [Creating a ROS Package](#)

This tutorial covers using `roscmake-pkg` or `catkin` to create a new package, and `rospack` to list package dependencies.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

4. [Building a ROS Package](#)

This tutorial covers the toolchain to build a package.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

5. [Understanding ROS Nodes](#)

This tutorial introduces ROS graph concepts and discusses the use of `roscore`, `roslaunch`, and `rosls` command line tools.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

6. [Understanding ROS Topics](#)

This tutorial introduces ROS topics as well as using the `rostopic` and `rqt_plot` command line tools.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

7. [Understanding ROS Services and Parameters](#)

This tutorial introduces ROS services, and parameters as well as using the `rosservice` and `rosparam` command line tools.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

8. [Using rqt console and roslaunch](#)

This tutorial introduces ROS using `rqt_console` and `rqt_logger_level` for debugging and `roslaunch` for starting many nodes at once. If you use ROS fuerte or earlier distros where `rqt` isn't fully available, please see this page with [this page](#) that uses old rx based tools.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

9. [Using rosed to edit files in ROS](#)

This tutorial shows how to use `rosed` to make editing easier.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

10. [Creating a ROS msg and srv](#)

This tutorial covers how to create and build msg and srv files as well as the [rosmmsg](#), rossrv and roscp command line tools.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

11. [Writing a Simple Publisher and Subscriber \(C++\)](#)

This tutorial covers how to write a publisher and subscriber node in C++.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

12. [Writing a Simple Publisher and Subscriber \(Python\)](#)

This tutorial covers how to write a publisher and subscriber node in python.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

13. [Examining the Simple Publisher and Subscriber](#)

This tutorial examines running the simple publisher and subscriber.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

14. [Writing a Simple Service and Client \(C++\)](#)

This tutorial covers how to write a service and client node in C++.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

15. [Writing a Simple Service and Client \(Python\)](#)

This tutorial covers how to write a service and client node in python.

([English](#)) ([Japanese](#)) ([Vietnamese](#))

16. [Examining the Simple Service and Client](#)

This tutorial examines running the simple service and client.

([English](#)) ([Japanese](#))

17. [Recording and playing back data](#)

This tutorial will teach you how to record data from a running ROS system into a .bag file, and then to play back the data to produce similar behavior in a running system.

([English](#)) ([Japanese](#))

18. [Getting started with roswtf](#)

Basic introduction to the [roswtf](#) tool.

([English](#)) ([Japanese](#))

19. [Navigating the ROS wiki](#)

This tutorial discusses the layout of the ROS wiki ([ros.org](#)) and talks about how to find what you want to know.

([English](#)) ([Japanese](#))

<http://wiki.ros.org/ja/ROS/Tutorials/NavigatingTheWiki>

20. [Where Next?](#)

This tutorial discusses options for getting to know more about using ROS on real or simulated robots.

([English](#)) ([Japanese](#))

3 Understanding ROS Navigation Stack with GAZEBO

<https://qiita.com/MoriKen/items/0b75ab291ab0d95c37c2>

1. 導入
2. move_base (ナビゲーション)
 1. ROS で遊んでみる
 2. ソフトウェア構成をみる
3. amcl (自己位置推定)
 1. ROS で遊んでみる
 2. ソフトウェア構成をみる
 3. 原理をみる (準備編)
 4. 原理をみる (応用編)
4. gmapping (地図生成)
 1. ROS で遊んでみる
 2. ソフトウェア構成をみる
 3. 原理をみる(応用編)



4 Understanding ROS Navigation Stack with mobile robot

1. Creating a new catkin workspace
 1. Creating a new catkin workspace
 2. Installing the navigation package
 3. Installing the gmapping package
2. Installing the LiDAR package
 1. Understanding the LiDAR topic
 2. Installing the LiDAR package
 3. Checking the LiDAR topic with *RVIZ*
3. Installing the Joystick package
 1. Understanding the Joystick topic
 2. Installing the Joystick package
 3. Checking the Joystick topic with *rostopic*
4. Creating a DC motor driver
 1. Understanding the wheel velocity topic
 2. Understanding PWM (Pulse Width Modulation)
 3. Installing the *Arduino IDE*
 4. Understanding the *Arduino IDE*
 5. Writing a motor driver Arduino sketch
 6. Check the rotation of the motor
5. Creating a wheel odometry localization program
 1. Understanding the differential wheeled robot kinematics
 2. Calculating a position and orientation of the robot
 3. Writing an odometry calculation program
6. Integrating the Navigation Stack
 1. Writing a launch file
 2. Creating a map with *gmapping*
 3. Moving the robot with *move_base*
 4. Playing with ROS !!

