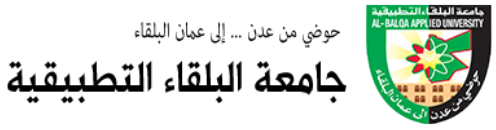


ROS Workshop

Packages, Tools, and Libraries

January 11, 2023

Hassan Umari

1. Packages

2. Gazebo

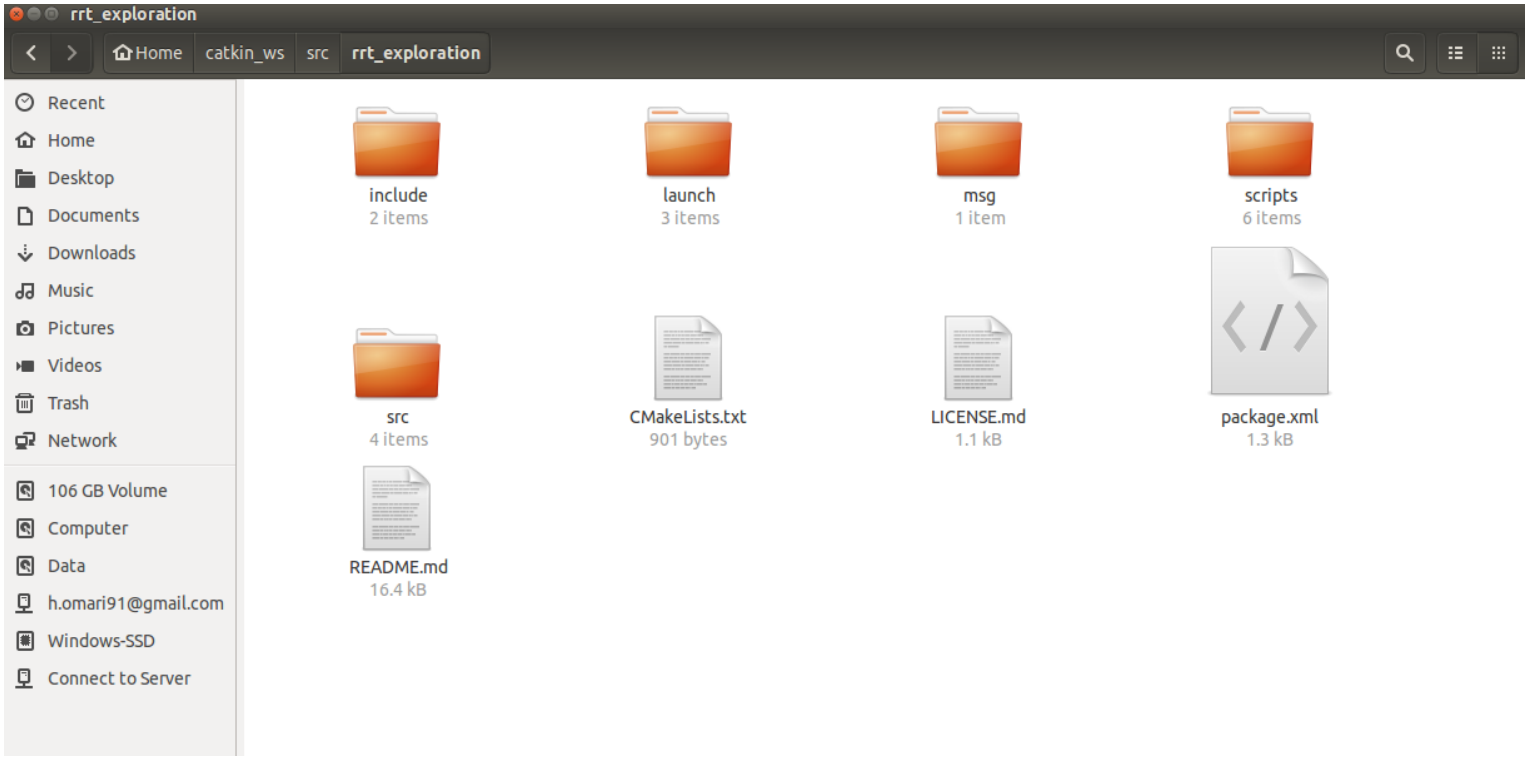
3. RViz

4. Coordinate Frames

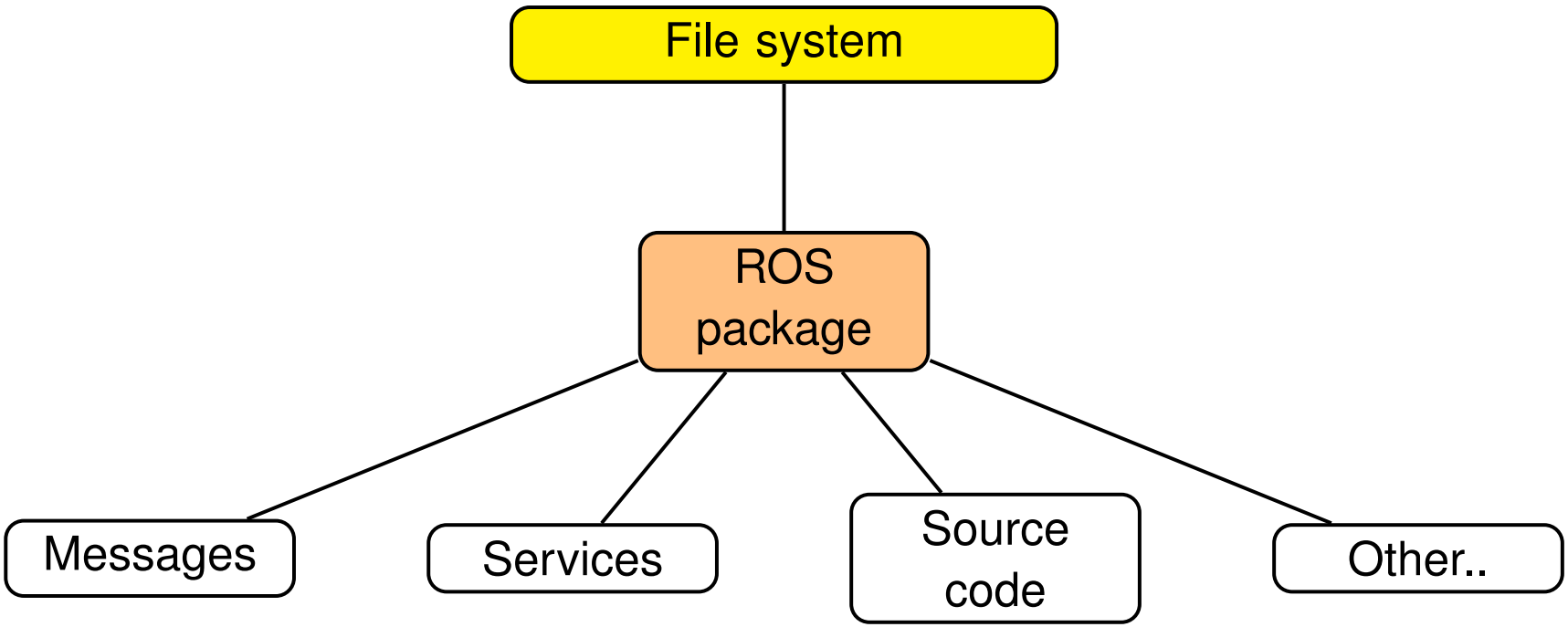
4.1 tf - The Transform Library

ROS Packages

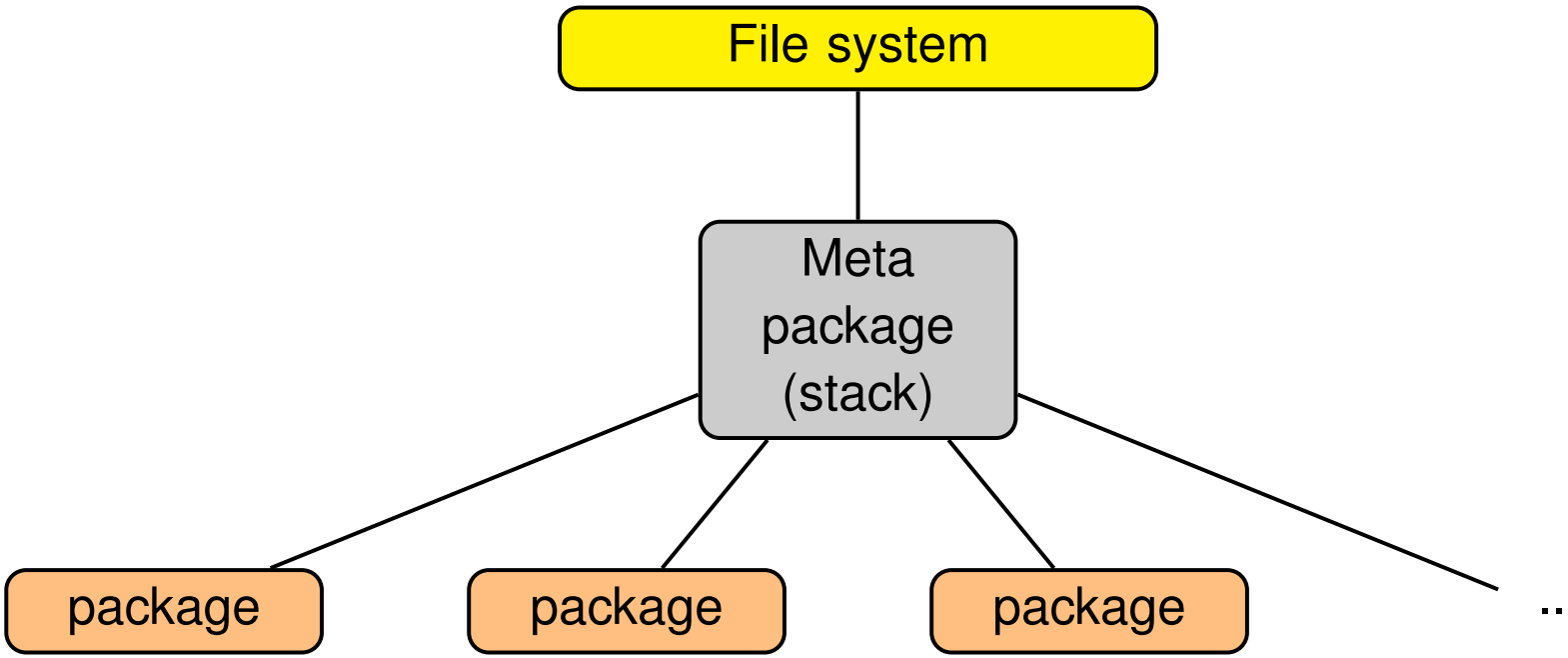
Inside a ROS package:



ROS Packages



ROS Packages



ROS Packages

How to install packages:

- From source (many packages available on GitHub).
- Using **apt**. Example:

```
sudo apt install ros-noetic-navigation
```

Demo

package, rosrun, launch files

1. Packages

2. Gazebo

3. RViz

4. Coordinate Frames

4.1 tf - The Transform Library

Gazebo

- Gazebo is a simulator that is bundled with ROS distributions.
- Gazebo includes a physics engine, 3D rendering (OpenGL), and support for simulating sensors and actuators.
- Simulation environment can be defined in a `.world file`.

Example

1. Packages

2. Gazebo

3. RViz

4. Coordinate Frames

4.1 tf - The Transform Library

RViz

- RViz can be used to **visualize** several commonly used ROS messages
- Examples: laser scans, occupancy gird maps, point clouds, images, robot frames, etc..

RViz

1. Packages

2. Gazebo

3. RViz

4. Coordinate Frames

4.1 tf - The Transform Library

Coordinate Frames

- A robotic system involves several coordinate frames that change over time.
- A common task in robotics, is to find the transformation between frames.
- Visualization of these frames is also very important.

Coordinate Frames

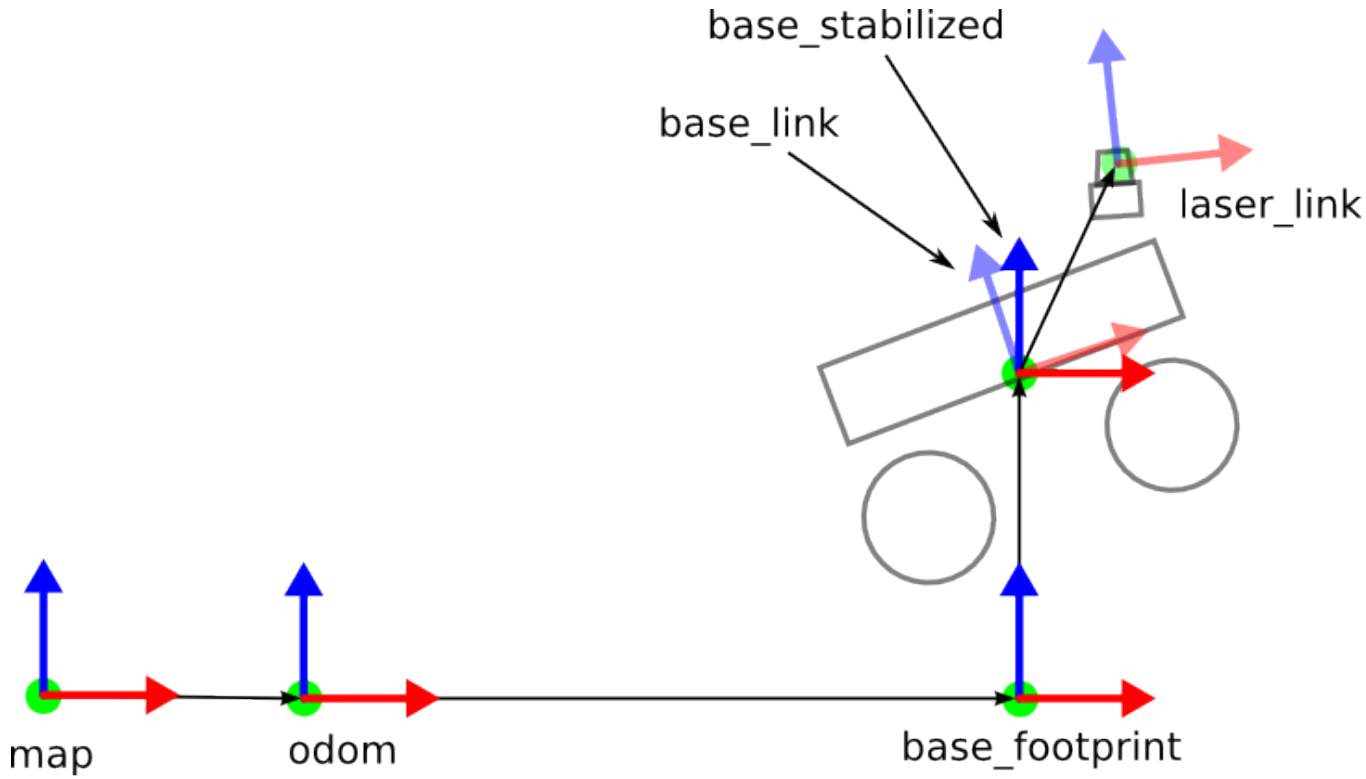
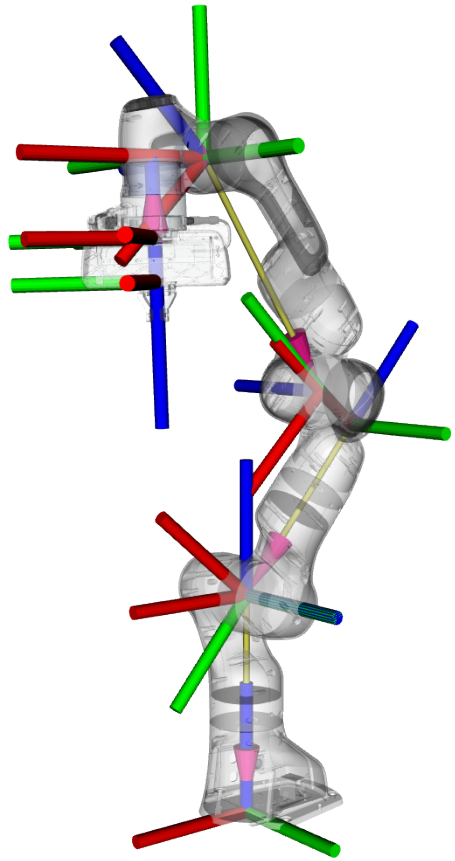


image source: http://wiki.ros.org/hector_slam/Tutorials/SettingUpForYourRobot

Coordinate Frames



Source: https://ros-planning.github.io/moveit_tutorials/doc/robot_model_and_robot_state/robot_model_and_robot_state_tutorial.html

1. Packages

2. Gazebo

3. RViz

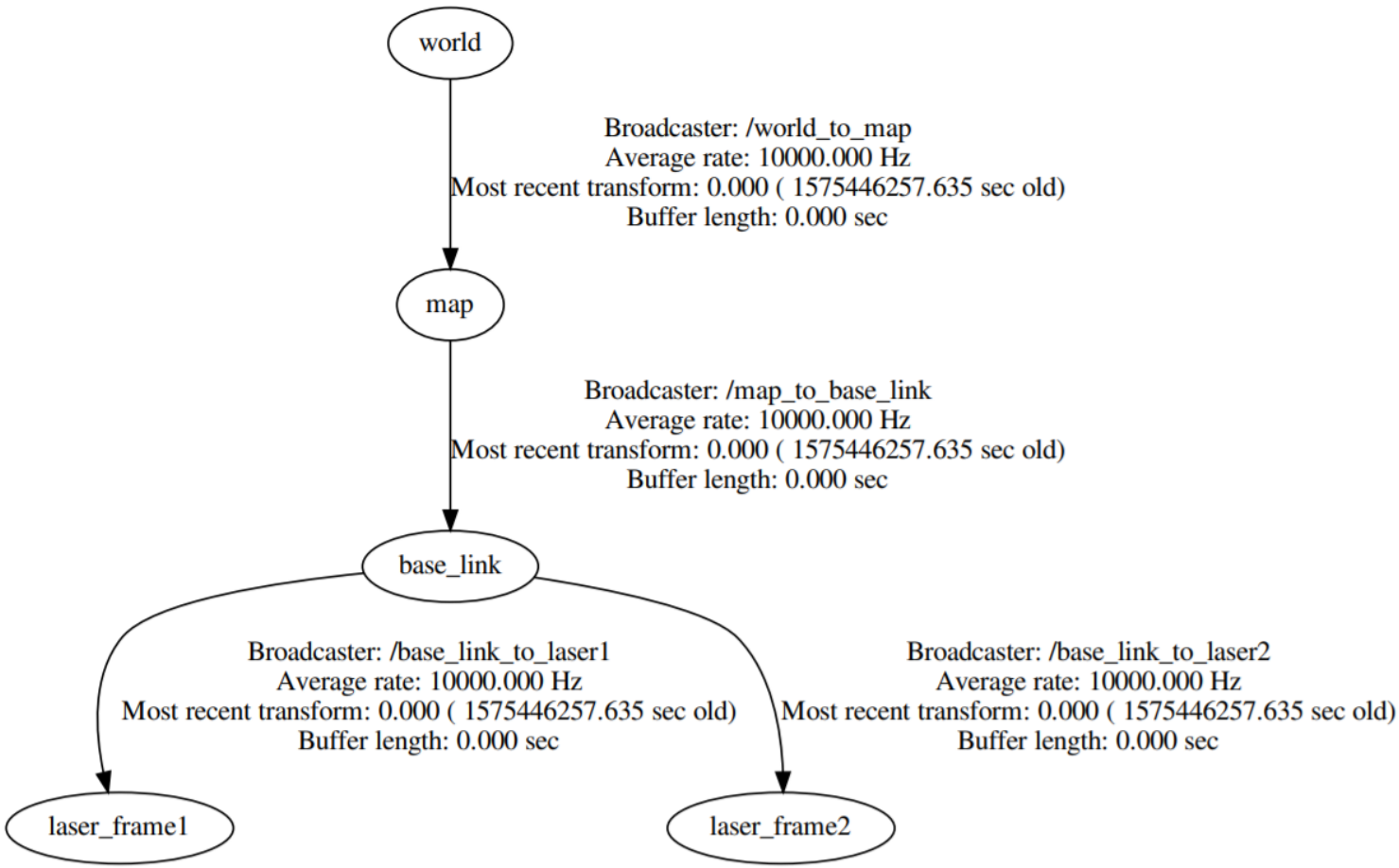
4. Coordinate Frames

4.1 tf - The Transform Library

tf - The Transform Library

- The tf is a library for handling **transformation** between frames.
- The library stores the relationship between frames in a **tree** structure buffered in **time**.
- In ROS, it comes as a package called **tf2**.

tf - The Transform Library



Source: https://spl.hevs.io/spl-docs/tools/ros/tf2.html

tf - The Transform Library

- There is no central server. Each client listens to published transforms and maintains a copy of the tree.
- What a node can do:
  - Broadcast a transformation
  - Listens for all transformations
  - query for a transform at a chosen time instance
- Comes with a tool to quickly publish static transformations.

Demo