Rosbridge

Introduction of RosBridge

Rosbridge provides a JSON API to ROS functionality for non-ROS programs. There are a variety of front ends that interface with rosbridge, including a WebSocket server for web browsers to interact with. **Rosbridge_suite** is a meta-package containing rosbridge, various front end packages for rosbridge like a WebSocket package, and helper packages.

https://wiki.ros.org/rosbridge_suite

https://github.com/RobotWebTools/rosbridge_suite

Basically, if you need to connect external devices or system, you need to use Rosbridge to setup connection.

- All details below suits ros system in linux **melodic** distribution release.

1. Contents

It mainly contains two parts: **protocal and implementation.**

1.1 Protocal

The rosbridge protocol is a specification for sending JSON based commands to ROS (and in theory, any other robot middleware). The specification is programming language and transport agnostic. The idea is that any language or transport that can send JSON can talk the rosbridge protocol and interact with ROS.

💡 JSON is text, written with JavaScript object notation. It can easily be sent to and from a server, and used as a data format by any programming language. JSON tutorial: https://www.w3schools.com/js/js_json_intro.asp

> The protocol covers subscribing and publishing topics, service calls, getting and setting params, and even compressing messages and more.

1.2 Rosbridge Implementation

1.2.1 rosbridge_library

https://wiki.ros.org/rosbridge_library

Rosbridge library is a Python library responsible for **taking JSON strings and converting them to ROS messages**, and vice versa. Rosbridge library is meant to be used as a library for transport layer packages. For example, the rosbridge_server package creates a WebSocket connection and uses the rosbridge library to handle the JSON to ROS conversion.

Any Python package or program can use rosbridge library for direct JSON to ROS communication. For example, a TCP server, a serial bridge, etc.

1.2.2 rosapi

https://wiki.ros.org/rosapi

Rosapi is part of the rosbridge_suite, **exposing ROS functionality like getting and setting** params, topics list, and more via service calls. Rosapi exposes the functionality normally reserved for ROS client libraries to outside programs via rosbridge.

Provides service calls for getting ros meta-information as follows:

- Topics
- Services
- Parameters

1.2.3 rosbridge_server

https://wiki.ros.org/rosbridge_server

Rosbridge server provides a **WebSocket transport layer**, which is bidirectional communication layer between clients (web browsers) and servers. By providing a WebSocket connection, rosbridge server allows **webpages to talk ROS** using the rosbridge protocol.

Rosbridge server creates a WebSocket connection and passes any JSON messages from the WebSocket to rosbridge_library, so rosbridge library can convert the JSON strings into ROS calls. The reverse also happens, with rosbridge library converting any ROS responses into JSON, then passing it to rosbridge server to send over the WebSocket connection.

2. Tutorials

https://wiki.ros.org/rosbridge_suite/Tutorials/RunningRosbridge

3.1 Install

```
sudo apt-get install ros--rosbridge-server

//rosdistro means the distribution of ros (melodic)
```

3.2 Running

- Setup environment

```
source /opt/ros//setup.bash
roslaunch rosbridge_server rosbridge_websocket.launch
```

- Launch rosbridge and specific package (below it is example for websocket launch file)

```
        //Where you need configure port number
```

3.3 Check out roslibjs library that talks to rosbridge server over webSocket.