

RoArm-M1 Tutorial VIII: Use of rqt in ROS2

From Waveshare Wiki
Jump to: navigation, search

RoArm-M1 Tutorial Directory

- RoArm-M1 Tutorial I: How To Use (/wiki/RoArm-M1_Tutorial_I:_How_To_Use)
- RoArm-M1 Tutorial II: Secondary Development Tutorial (/wiki/RoArm-M1_Tutorial_II:_Secondary_Development_Tutorial)
- RoArm-M1 Tutorial III: VMware ROS2 Getting Started Tutorial (/wiki/RoArm-M1_Tutorial_III:_VMware_ROS2_Getting_Started_Tutorial)
- RoArm-M1 Tutorial IV: URDF Model Tutorial (/wiki/RoArm-M1_Tutorial_IV:_URDF_Model_Tutorial)
- RoArm-M1 Tutorial V: ROS2 Serial Communication Node (/wiki/RoArm-M1_Tutorial_V:_ROS2_Serial_Communication_Node)
- RoArm-M1 Tutorial VI: How to assemble RoArm-M1 (/wiki/RoArm-M1_Tutorial_VI:_How_to_assemble_RoArm-M1)
- RoArm-M1 Tutorial VII: Assembly Graphics Tutorial (/wiki/RoArm-M1_Tutorial_VII:_Assembly_Graphics_Tutorial)
- RoArm-M1 Tutorial VIII: Use of rqt in ROS2
- RoArm-M1 Tutorial IV: URDF Model Tutorial (/wiki/RoArm-M1_Tutorial_IV:_URDF_Model_Tutorial)
- RoArm-M1 Main Page (/wiki/RoArm-M1)

Use of rqt in ROS2

This tutorial is used to guide the use of rqt in ROS2.

rqt Introduction

In ROS2, rqt is a graphical toolset that allows users to monitor, debug, and visualize various aspects of the ROS2 system through a user interface. rqt is a GUI plugin framework for ROS (Robot Operating System) that enables the loading and utilization of different plugins as needed. rqt offers numerous plugins, including topic monitors, node monitors, service callers, parameter setters, and more.

Install and Use rqt

Open a new terminal and input the following commands to install rqt and its plug-in:

```
sudo apt update
sudo apt install ~nros-humble-rqt*
```

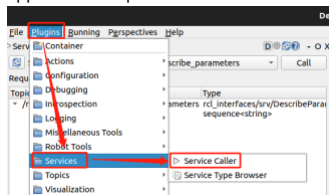
Run rqt:

```
rqt
```

This will open the main interface of rqt. When running rqt for the first time, the window will be blank. You can load and use plugins by following these steps:

Click on the "Plugins" option in the menu bar at the top of the rqt main interface.

Under the "Plugins" menu, you'll see various available plugin options. Select "Services" -> "Service Caller" to call services. Once you've loaded the "Service Caller" plugin, it will appear on the rqt main interface.



(/wiki/File:Rqt_in_ROS2-01.png)

Note:

rqt may take some time to locate all the plugins. If you don't see "Services" or other plugin options after clicking "Plugins", close rqt and enter the command `rqt --force-discover` in the terminal.

Use rqt_graph to View ROS2 Topic

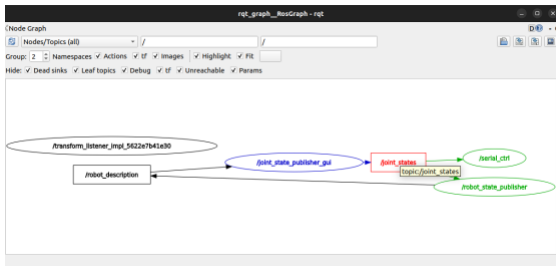
A "topic" is a communication mechanism for message passing between nodes. It can be seen as a publisher-subscriber pattern, where one node publishes (publisher) messages to a topic, and other nodes can subscribe (subscribers) to that topic to receive corresponding messages.

In this section, we'll use rqt_graph to visualize the changes in nodes and topics, as well as their connections. This part requires having all four windows open after running the serial communication node in Tutorial V.

To run rqt_graph, open a new terminal and input the following command:

```
rqt_graph
```

You can also open rqt_graph by opening rqt and selecting "Plugins" -> "Introspection" -> "Node graph" in the menu bar at the top of the main interface, the interface as shown below:



(/wiki/File:Rqt_in_ROS203.png)

You can see the mentioned nodes and topics represented by circular icons for nodes and rectangular icons for topics. When you hover the mouse over the central topic, you'll notice it is highlighted in colors, as shown in the image above.

The highlighted section demonstrates how the /joint_state_publisher_gui node communicates with the /serial_ctrl and /robot_state_publisher nodes through the /joint_states topic. The /joint_state_publisher_gui node publishes data to the /joint_states topic, while the /serial_ctrl and /robot_state_publisher nodes subscribe to this topic to receive the data.

The highlighting feature in rqt_graph is very helpful when examining more complex systems with numerous nodes and various connections among topics.

Using Command Line Tools to View ROS2 Topics

rqt_graph is a graphical tool. Now we will introduce some command line tools for viewing topics.

ros2 topic list

Open a new terminal and run the following command to see all the topics currently active on your system:

```
ros2 topic list
```

```
roarm@roarm-virtual-machine:~$ ros2 topic list
/clicked_point
/clock
/goal_pose
/initialpose
/joint_states
/parameter_events
/robot_description
/rosout
/tf
/tf_static
```

(/wiki/File:Rqt_in_ROS2-07.png)

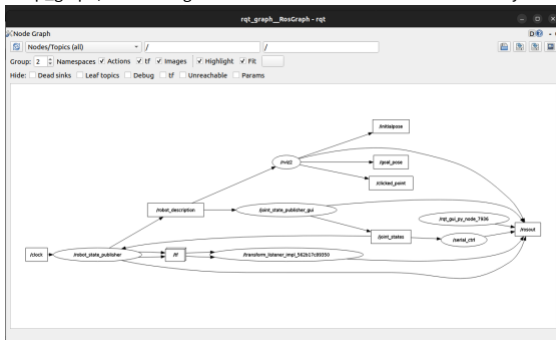
Running the following command displays the same list of topics, but this time with the data types appended in parentheses:

```
ros2 topic list -t
```

```
roarm@roarm-virtual-machine:~$ ros2 topic list -t
/clicked_point [geometry_msgs/msg/PointStamped]
/clock [roscpp_msgs/msg/Clock]
/goal_pose [geometry_msgs/msg/PoseStamped]
/initialpose [geometry_msgs/msg/PoseWithCovarianceStamped]
/joint_states [sensor_msgs/msg/JointState]
/parameter_events [rcdl_interfaces/msg/ParameterEvent]
/robot_description [std_msgs/msg/String]
/rosout [rcdl_interfaces/msg/Log]
/tf [tf2_msgs/msg/TFMessage]
/tf_static [tf2_msgs/msg/TFMessage]
```

(/wiki/File:Rqt_in_ROS2-08.png)

In rqt_graph, unchecking all the checkboxes under "Hide" will allow you to see these topics:



(/wiki/File:Rqt_in_ROS2-09.png)

ros2 topic echo

To view the data published on a topic, please use the following command: `ros2 topic echo <topic_name>`.

Given that we know the /joint_state_publisher_gui node publishes data to the /robot_state_publisher node via the /joint_states topic, let's use echo to inspect that topic:

```
ros2 topic echo /joint_states.
```

Next, use the control panel to control the robotic arm's movement. Meanwhile, observe the terminal running the echo command. You'll see position data being published each time you drag:

```

header:
  stamp:
    sec: 1688456184
    nanosec: 310188902
  frame_id: ''
name:
- base_to_L1
- L1_to_L2
- L2_to_L3
- L3_to_L4
- L4_to_L5_1_A
- L4_to_L5_1_B
- L4_to_L5_2_A
- L4_to_L5_2_B
- L5_1_A_to_L5_3_A
- L5_1_B_to_L5_3_B
position:
- 0.0
- 0.58280256
- -1.8532800000000003
- 0.0
- 0.0
- 0.0
- 0.0
- 0.0
- 0.0
- 0.0
- 0.0
- 0.0
velocity: []
effort: []
---
```

(/wiki/File:Rqt_in_ROS2-10.png)

In this tutorial, you've used both rqt_graph and command-line tools to visualize connections between multiple nodes through topics. By doing so, you should now have a good understanding of how data is transmitted within the ROS2 system.

Retrieved from "https://www.waveshare.com/w/index.php?title=RoArm-M1_Tutorial_VIII:_Use_of_rqt_in_ROS2&oldid=74879" (https://www.waveshare.com/w/index.php?title=RoArm-M1_Tutorial_VIII:_Use_of_rqt_in_ROS2&oldid=74879)"