CSCI 545 – Introduction to Robotics

Lab1 Report

Team 17 - QuirkyBot

Zishen Wei - Jiashang Cao - Srija Madarapu - Mahya Gheini - Zhen Qin

6 ROS exercises

2. Find Topics using rostopics

(b)

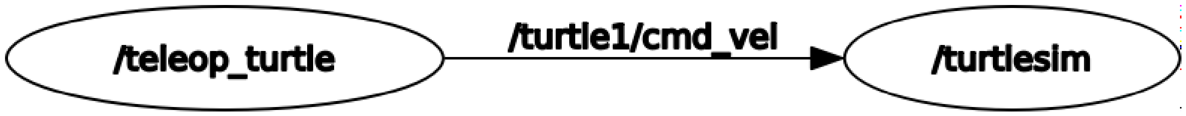


Figure1: Graphical Representation of The Nodes

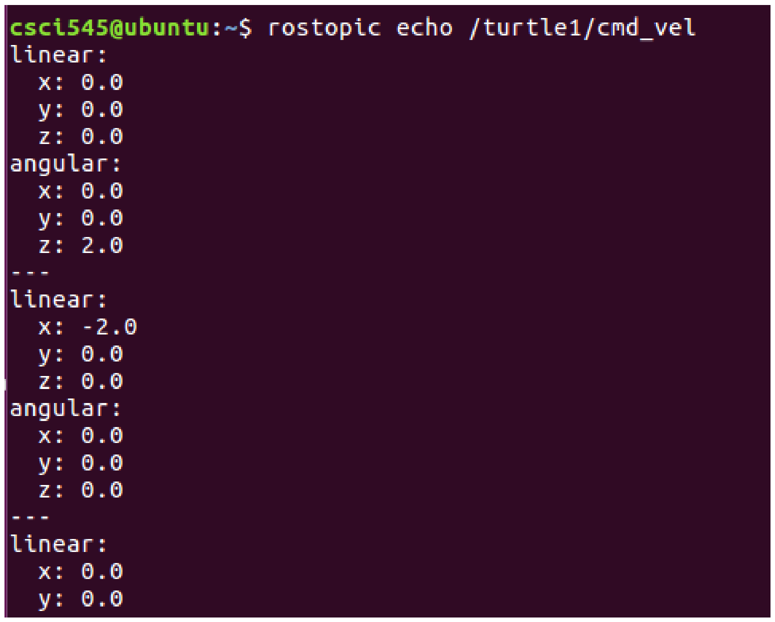
As you can see in Figure1, there are two nodes (***/teleop_turtle*** and ***/turtlesim***) and one topic (***/turtle1/cmd_vel***) in the graph.

Node (***/teleop_turtle***) is the publisher of the topic (***/turtle1/cmd_vel***) , and node (***/turtlesim***) subscribes to the topic (***/turtle1/cmd_vel***).

Node (***/teleop_turtle***) is sending keyboard input messages to node (***/turtlesim***) through the topic (***/turtle1/cmd_vel***).

(c)

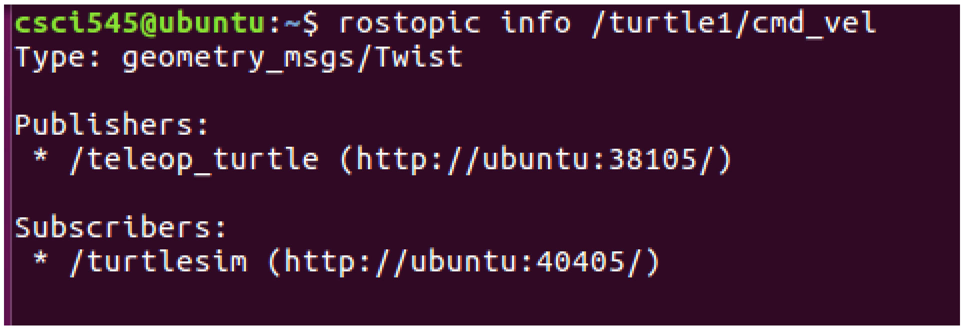
The messages from the topic (***/turtle1/cmd_vel***) showed the linear and angular velocity of the turtle. This information is shown in Figure2 as well.



```
csci545@ubuntu:~$ rostopic echo /turtle1/cmd_vel
linear:
  x: 0.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 2.0
---
linear:
  x: -2.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
linear:
  x: 0.0
  y: 0.0
```

Figure2: Messages Gained From rostopic - Topic ***/turtle1/cmd_vel***

Using rostopic we can also gain detailed information about a topic; such as publisher, subscribers, and message type. Figure3 shows the publishers, subscribers, and its message type of the topic (***/turtle1/cmd_vel***).



```
csci545@ubuntu:~$ rostopic info /turtle1/cmd_vel
Type: geometry_msgs/Twist

Publishers:
 * /teleop_turtle (http://ubuntu:38105/)

Subscribers:
 * /turtlesim (http://ubuntu:40405/)
```

Figure3: Information Gained From rostopic - Topic ***/turtle1/cmd_vel***

5. Difference between **rosservice**, **rostopic**, **rosparams**, and **rosbag**

These are all command-line tools, but for different purposes in ROS framework:

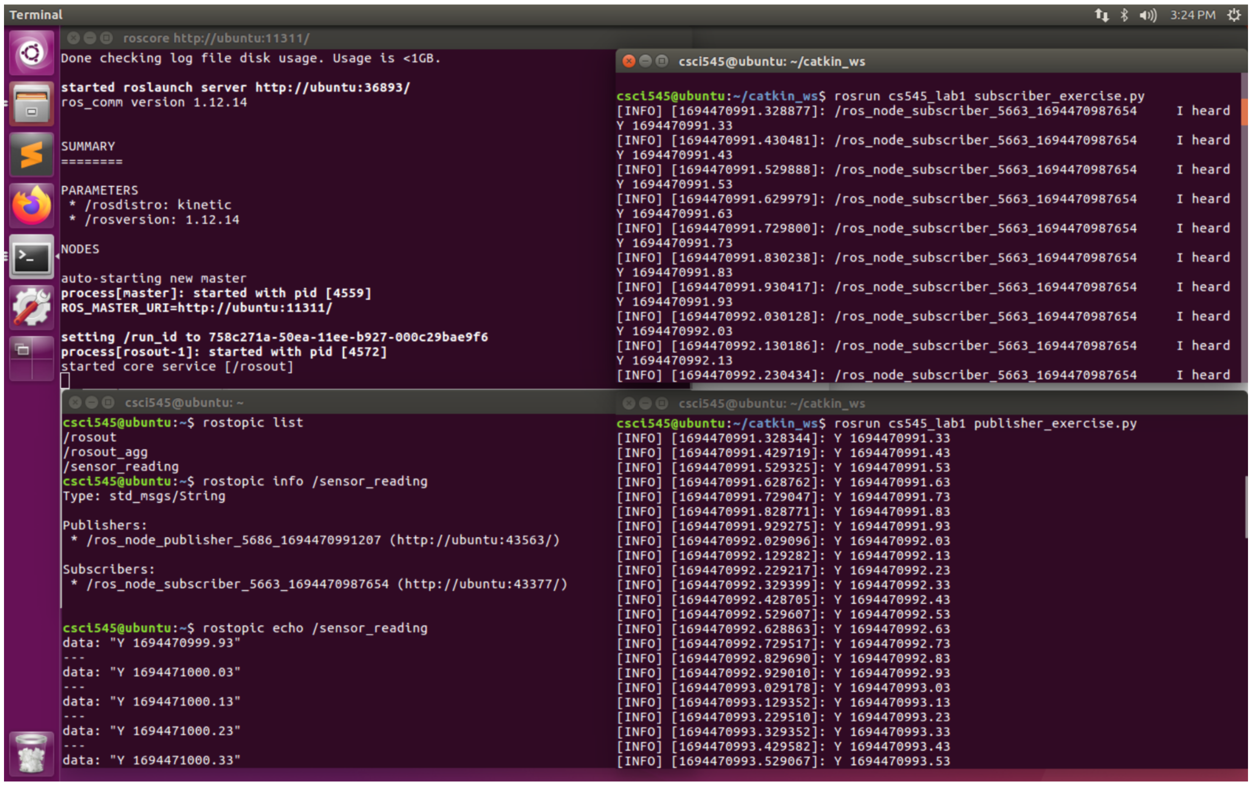
Rosservice for listing and querying ros services and is a quick communication system for remote procedure calls.

Rostopic for displaying debug information about ros topics and ros messages, and is a communication system for continuous data streams, which allows many publishers to write into it, and the subscribers can read these data flows.

Rosparam for getting and setting ros parameters and is a data storage system that enables users to store or manipulate data.

Rosbag for recording from and playing back to ros topics.

7 Publisher and Subscriber Topics


Terminal
3:24 PM
roscore http://ubuntu:11311/
csci545@ubuntu: ~/catkin_ws
csci545@ubuntu: ~
csci545@ubuntu: ~/catkin_ws

```
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://ubuntu:36893/
ros_comm version 1.12.14


SUMMARY
========

PARAMETERS
 * /rosdistro: kinetic
 * /rosversion: 1.12.14

NODES

auto-starting new master
process[master]: started with pid [4559]
ROS_MASTER_URI=http://ubuntu:11311/

setting /run_id to 758c271a-50ea-11ee-b927-000c29bae9f6
process[rosout-1]: started with pid [4572]
started core service [/rosout]
```

```
csci545@ubuntu:~/catkin_ws$ rosrun cs545_lab1 subscriber_exercise.py
[INFO] [1694470991.328877]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.33
[INFO] [1694470991.430481]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.43
[INFO] [1694470991.529888]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.53
[INFO] [1694470991.629979]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.63
[INFO] [1694470991.729800]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.73
[INFO] [1694470991.830238]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.83
[INFO] [1694470991.930417]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470991.93
[INFO] [1694470992.030128]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470992.03
[INFO] [1694470992.130186]: /ros_node_subscriber_5663_1694470987654     I heard
Y 1694470992.13
[INFO] [1694470992.230434]: /ros_node_subscriber_5663_1694470987654     I heard
```

```
csci545@ubuntu:~$ rostopic list
/rosout
/rosout_agg
/sensor_reading
csci545@ubuntu:~$ rostopic info /sensor_reading
Type: std_msgs/String

Publishers:
 * /ros_node_publisher_5686_1694470991207 (http://ubuntu:43563/)

Subscribers:
 * /ros_node_subscriber_5663_1694470987654 (http://ubuntu:43377/)


csci545@ubuntu:~$ rostopic echo /sensor_reading
data: "Y 1694470999.93"
---
data: "Y 1694471000.03"
---
data: "Y 1694471000.13"
---
data: "Y 1694471000.23"
---
data: "Y 1694471000.33"
```

```
csci545@ubuntu:~/catkin_ws$ rosrun cs545_lab1 publisher_exercise.py
[INFO] [1694470991.328344]: Y 1694470991.33
[INFO] [1694470991.429719]: Y 1694470991.43
[INFO] [1694470991.529325]: Y 1694470991.53
[INFO] [1694470991.628762]: Y 1694470991.63
[INFO] [1694470991.729047]: Y 1694470991.73
[INFO] [1694470991.828771]: Y 1694470991.83
[INFO] [1694470991.929275]: Y 1694470991.93
[INFO] [1694470992.029096]: Y 1694470992.03
[INFO] [1694470992.129282]: Y 1694470992.13
[INFO] [1694470992.229217]: Y 1694470992.23
[INFO] [1694470992.329399]: Y 1694470992.33
[INFO] [1694470992.428705]: Y 1694470992.43
[INFO] [1694470992.529607]: Y 1694470992.53
[INFO] [1694470992.628863]: Y 1694470992.63
[INFO] [1694470992.729517]: Y 1694470992.73
[INFO] [1694470992.829690]: Y 1694470992.83
[INFO] [1694470992.929010]: Y 1694470992.93
[INFO] [1694470993.029178]: Y 1694470993.03
[INFO] [1694470993.129352]: Y 1694470993.13
[INFO] [1694470993.229510]: Y 1694470993.23
[INFO] [1694470993.329352]: Y 1694470993.33
[INFO] [1694470993.429582]: Y 1694470993.43
[INFO] [1694470993.529067]: Y 1694470993.53
```

Figure4: Terminal screenshots when roscore, publisher_exercise.py, and subscriber_exercise.py is running

As we can see in Figure4 above, after starting ROS with ***roscore***(terminal upper-left), we can start running the ***subscriber***(terminal upper-right) and the ***publisher***(terminal lower-right). The ***publisher*** keeps generating messages until we kill(Ctrl-C) it. The ***subscriber*** only displays the call back message when the publisher is running. While the **publisher** and **subscriber** are running, we can check the topic detail using ***rostopic***(terminal lower-right):

rostopic list: Get a list of topics so that we can confirm current topic names.

rostopic info /topic_name: Get subscriber & publisher information from the topic.

rostopic echo /topic_name: Prints out the message sent by the publisher.