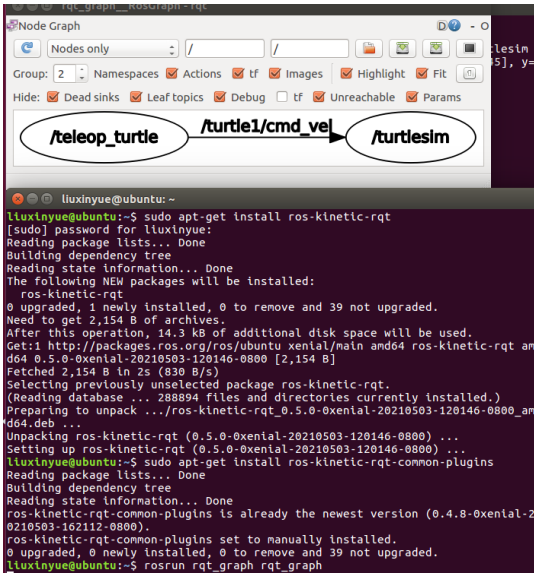


课程实验报告

学号：202100130052	姓名：刘欣月	班级：人工智能班
实验题目：机器人操作系统 ROS ROS 话题通信实验		
实验学时：2	实验日期：20210510	
实验目的：机器人操作系统 ROS 话题通信实验，在小海龟的基础上，学习 ROS 话题通信的相关内容。		
实验环境：Ubuntu 16 ROS		
<p>实验步骤：</p> <p>实验三：ROS 话题同行实验</p> <p>rqt_graph 安装与运行，首先运行小海龟程序，再打开输入 rosrund rpq_graph rqt_graph，结果如下：两个节点和一个话题</p>  <p>rostopic 介绍，rostopic 命令工具能获取有关 ROS 话题的信息，运行 rostopic -h 命令，结果如下</p>		

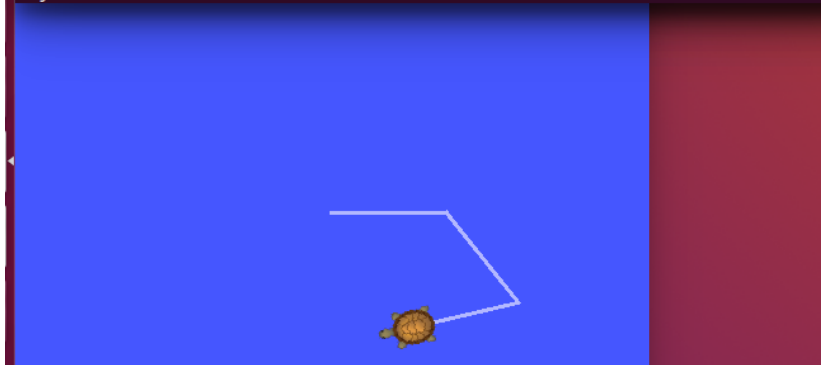
```
bash: /home/catkin_ws/devel/setup.bash: No such file or directory
liuxinyue@ubuntu:~$ rostopic -h
rostopic is a command-line tool for printing information about ROS Topics.

Commands:
  rostopic bw      display bandwidth used by topic
  rostopic delay   display delay of topic from timestamp in header
  rostopic echo    print messages to screen
  rostopic find    find topics by type
  rostopic hz      display publishing rate of topic
  rostopic info    print information about active topic
  rostopic list    list active topics
  rostopic pub     publish data to topic
  rostopic type    print topic or field type

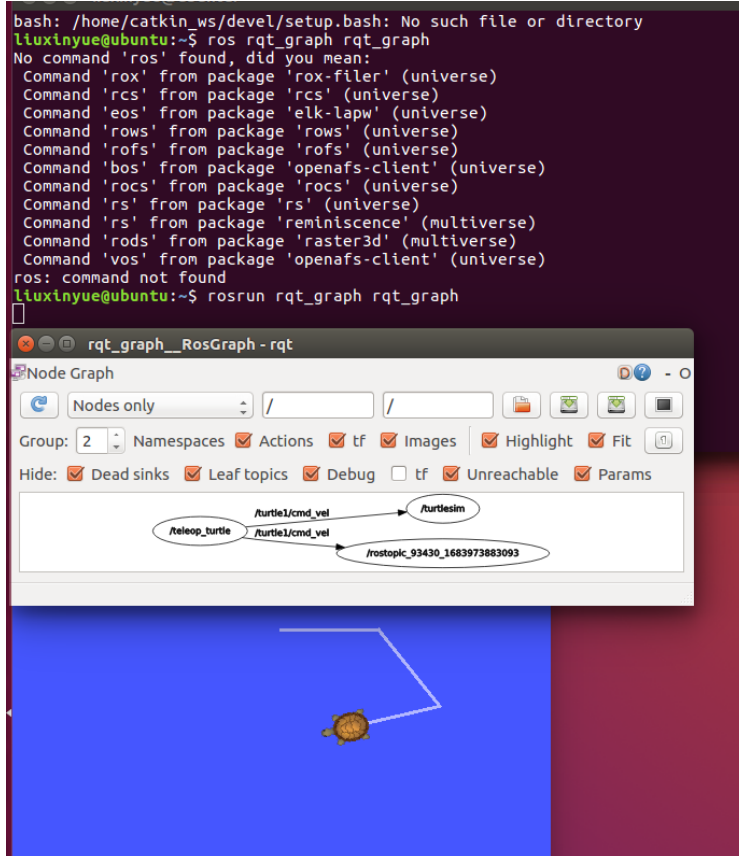
Type rostopic <command> -h for more detailed usage, e.g. 'rostopic echo -h'
liuxinyue@ubuntu:~$
```

Rostopic echo 可以显示在某个话题上发布的数据，在终端输入 `rostopic echo /turtle1/cmd_vel`，由于还没有数据发布到话题上就看不到任何东西，接下来通过按方向键让小海龟动起来，`turtle_teleop_key` 节点会发送数据，如下所示：

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



刷新 ROS graph rostopic echo 发现订阅了 `turtle1/cmd_vel` 话题



Rostopic list 能列出当前所有订阅和发布的话题

Rostopic list -h 结果和 rostopic list -v 结果如下所示：

```
bash: /home/catkin_ws/devel/setup.bash: No such file or directory
liuxinyue@ubuntu:~$ rostopic list -h
Usage: rostopic list [/namespace]

Options:
  -h, --help                show this help message and exit
  -b BAGFILE, --bag=BAGFILE
                           list topics in .bag file
  -v, --verbose              list full details about each topic
  -p                         list only publishers
  -s                         list only subscribers
  --host                     group by host name
liuxinyue@ubuntu:~$ rostopic list -v

Published topics:
* /turtle1/color_sensor [turtlesim/Color] 1 publisher
* /turtle1/cmd_vel [geometry_msgs/Twist] 1 publisher
* /rosout [rosgraph_msgs/Log] 4 publishers
* /rosout_agg [rosgraph_msgs/Log] 1 publisher
* /turtle1/pose [turtlesim/Pose] 1 publisher

Subscribed topics:
* /turtle1/cmd_vel [geometry_msgs/Twist] 2 subscribers
* /rosout [rosgraph_msgs/Log] 1 subscriber
* /statistics [rosgraph_msgs/TopicStatistics] 1 subscriber
```

ROS Message

话题之间的通信时通过在节点之家发送 ROS 消息实现的, 对于发布者 和订阅者之间的通信, 发布者和订阅者之间必须发送和接受相同

类型的消息,这就意味着话题的类型是由发布在他上面的消息类型决定的,使用 `rostopic type` 来查看发布在某个话题上的消息类型,用 `rosmmsg show geometry_msgs/Twist` 来查看消息的详细情况。

```
bash: /home/catkin_ws/devel/setup.bash: No such file or directory
liuxinyue@ubuntu:~$ rostopic type /turtle1/cmd_vel
geometry_msgs/Twist
liuxinyue@ubuntu:~$ rosmmsg show geometry_msgs/Twist
geometry_msgs/Vector3 linear
  float64 x
  float64 y
  float64 z
geometry_msgs/Vector3 angular
  float64 x
  float64 y
  float64 z
```

使用 `rostopic pub` 可以把数据发布到当前某个正在广播的话题上,如下所示,发布消息到某个给定的话题, `-l` 这个给参数选项是 `rostopic` 发布一条消息后马上退出

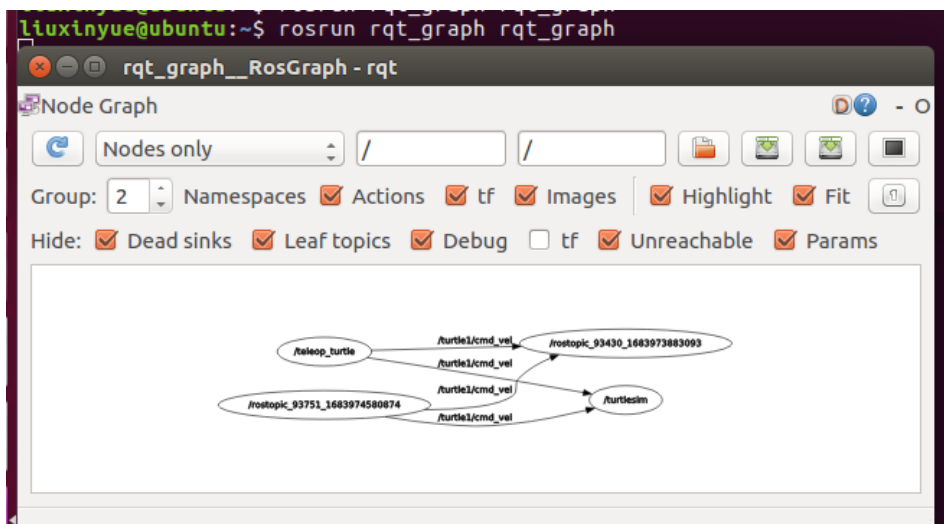
然后的参数是话题名称,然后是发布消息的类型,接下来是告诉命令解释器参数部分,分贝是对应的线速度和角速度。

```
liuxinyue@ubuntu:~$ rostopic pub -l /turtle1/cmd_vel geometry_msgs/Twist -- '[2.0,0.0,0.0]' '[0.0,0.0,1.8]'
publishing and latching message for 3.0 seconds
liuxinyue@ubuntu:~$
```



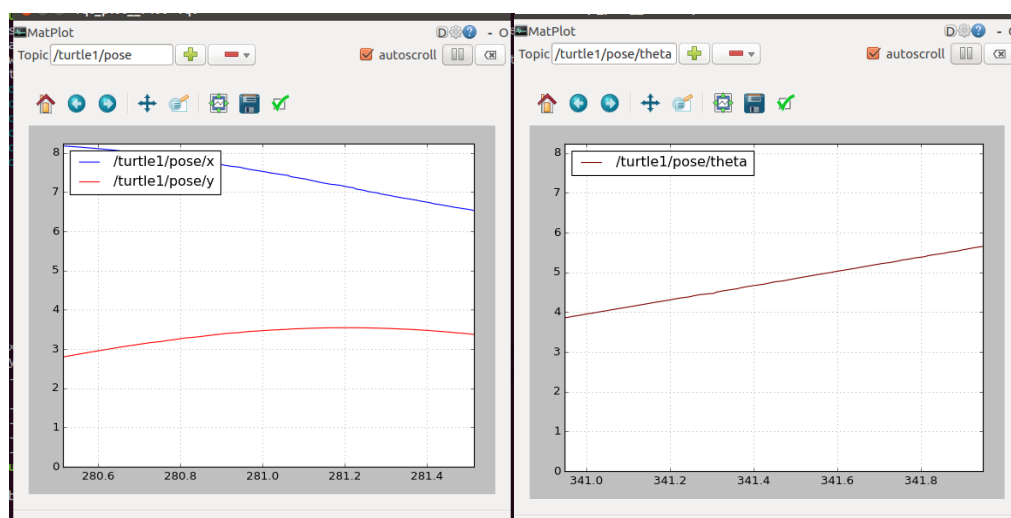
```
liuxinyue@ubuntu:~$ rostopic pub /turtle1/cmd_vel geometry_msgs/Twist -r 1 -- '[2.0,0.0,0.0]' '[0.0,0.0,1.8]'
```





```
liuxinyue@ubuntu:~$ rostopic echo /turtle1/cmd_vel
linear:
  x: 2.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 1.8
---
linear:
  x: 2.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
```

Rqt+plot 命令 可以实时显示一个发布到某个话题上的数据变化图形，这里使用 rpt_plot 明亮来绘制正在发布到/turtle/pose 话题上的数据变化图形。



--