

# 理解 ROS 话题

AI 航 团队

## 1. 开始

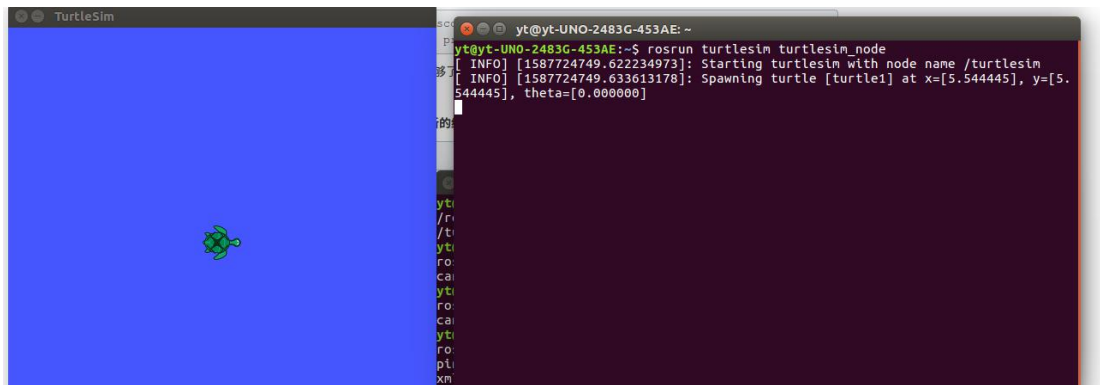
### 1.1 Roscore

首先确保 roscore 已经运行, 打开一个新的终端:

```
$ roscore
```

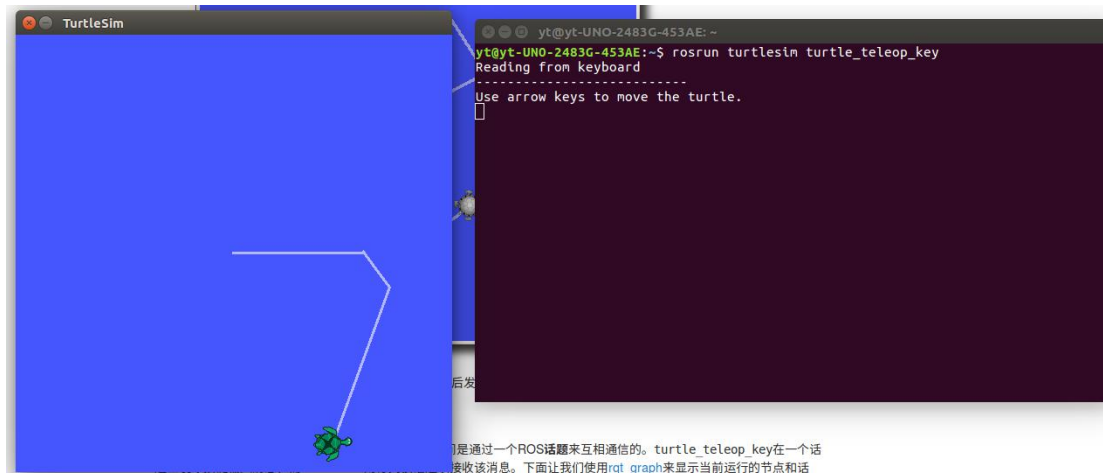
### 1.2 Turtlesim

在本教程中我们也会使用到 turtlesim, 请在一个新的终端中运行:



### 1.3 通过键盘远程控制 turtle

我们也需要通过键盘来控制 `turtle` 的运动，请在一个新的终端中运行：现在你可以使用键盘上的方向键来控制 `turtle` 运动了。如果不能控制，请选中 `turtle_teleop_key` 所在的终端窗口以确保你的按键输入能够被捕获。



## 2. Ros Topics

`turtlesim_node` 节点和 `turtle_teleop_key` 节点之间是通过一个 ROS 话题来互相通信的。`turtle_teleop_key` 在一个话题上发布按键输入消息，而 `turtlesim` 则订阅该话题以接收该消息。下面让我们使用 `rqt_graph` 来显示当前运行的节点和话题。

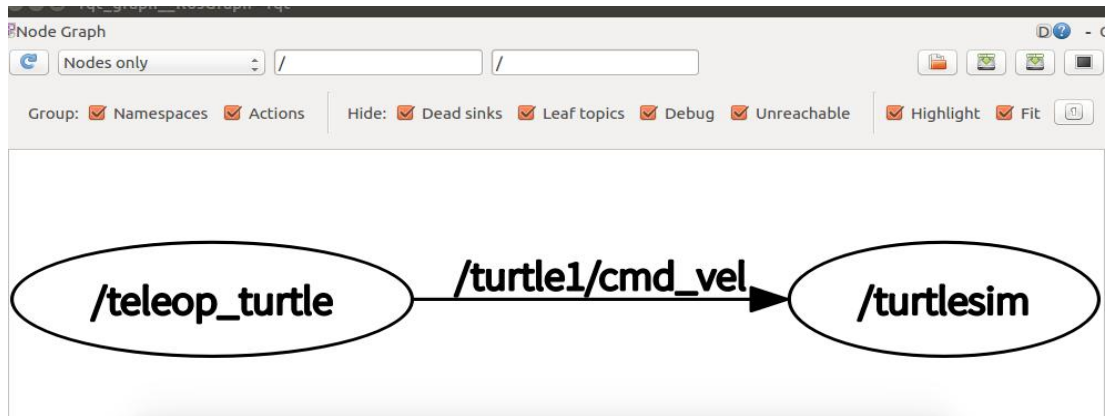
注意：如果你使用的是 `electric` 或更早期的版本，那么 `rqt` 是不可用的，请使用 `rxgraph` 代替。

### 2.1 使用 `rqt_graph`

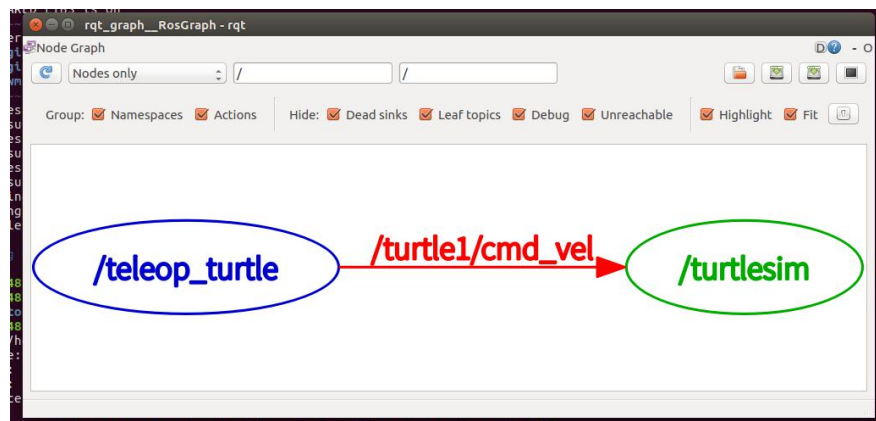
在一个新终端中运行:

```
$roslaunch rqt_graph rqt_graph
```

你会看到类似下图所示的图形:



如果你将鼠标放在/turtle1/command\_velocity上方, 相应的 ROS 节点 (蓝色和绿色) 和话题 (红色) 就会高亮显示。正如你所看到的, turtlesim\_node 和 turtle\_teleop\_key 节点正通过一个名为 /turtle1/command\_velocity 的话题来互相通信。



## 2.2 rostopic 介绍

rostopic 命令工具能让你获取有关 ROS 话题的信息。

你可以使用帮助选项查看 rostopic 的子命令:

```
$ rostopic -h
```

## 2.3 使用 rostopic echo

让我们在一个新终端中看一下 `turtle_teleop_key` 节点在 `/turtle1/command_velocity` 话题（非 hydro 版）上发布的数据。

```
$ rostopic -h
```

如果你是用 ROS Hydro 及其之后的版本（下同），请运行：

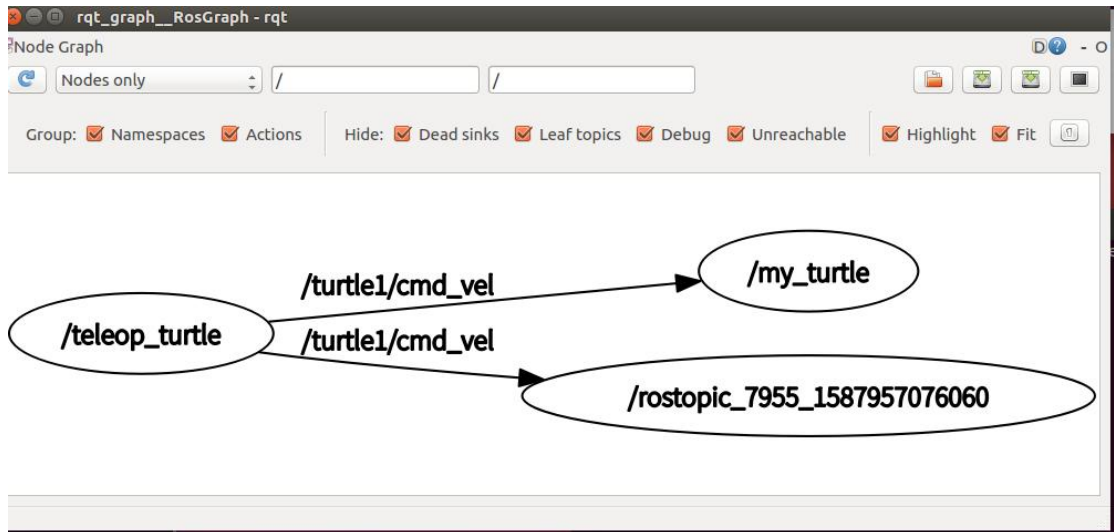
```
$ rostopic echo /turtle1/cmd_vel
```

你可能看不到任何东西因为现在还没有数据发布到该话题上。接下来我们通过按下方向键使 `turtle_teleop_key` 节点发布数据。记住如果 `turtle` 没有动起来的话就需要你重新选中 `turtle_teleop_key` 节点运行时所在的终端窗口。

现在当你按下向上方向键时应该会看到下面的信息：

```
yt@yt-UN0-2483G-453AE:~$ 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: -2.0
  y: 0.0
  z: 0.0
angular:
  x: 0.0
  y: 0.0
  z: 0.0
---
linear:
  x: -2.0
  y: 0.0
  z: 0.0
angular:
```

现在让我们再看一下 `rqt_graph`（你可能需要刷新一下 ROS graph）。正如你所看到的，`rostopic echo`(红色显示部分) 现在也订阅了 `/turtle1/command_velocity` 话题。



## 2.4 使用 rostopic list

rostopic list 能够列出所有当前订阅和发布的话题。在一个新终端中运行

```
$ rostopic -h
```

```
yt@yt-UN0-2483G-453AE:~$ 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
yt@yt-UN0-2483G-453AE:~$
```

## 3. ROS Messages

### 3.1 使用 rostopic type

rostopic type 命令用来查看所发布话题的消息类型。

hydro 版请运行：

```
$ rostopic type /turtle1/cmd_vel
```

```
yt@yt-UNO-2483G-453AE:~$ rostopic type /turtle1/cmd_vel
geometry_msgs/Twist
yt@yt-UNO-2483G-453AE:~$
```

我们可以使用 rosmmsg 命令来查看消息的详细情况

```
$rosmmsg show geometry_msgs/Twist
```

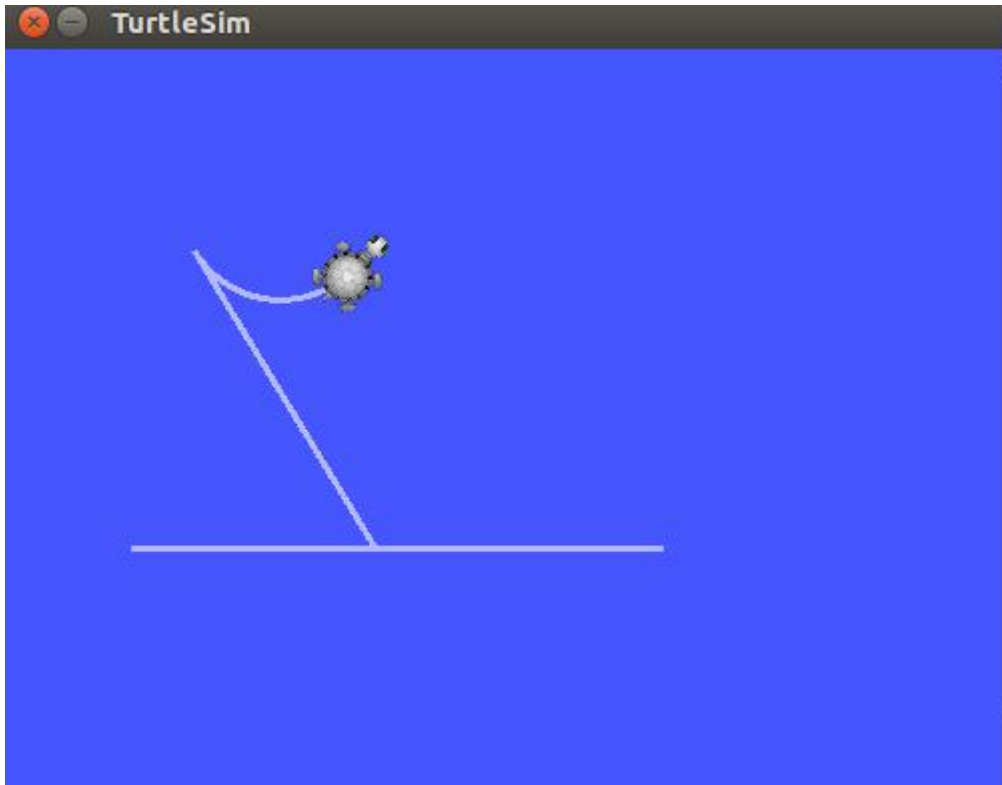
```
yt@yt-UNO-2483G-453AE: ~
yt@yt-UNO-2483G-453AE:~$ rostopic type /turtle1/cmd_vel |rosmmsg show
geometry_msgs/Vector3 linear
  float64 x
  float64 y
  float64 z
geometry_msgs/Vector3 angular
  float64 x
  float64 y
  float64 z
yt@yt-UNO-2483G-453AE:~$
```

## 4. 继续学习 ROS topics

### 4.1 使用 rostopic pub

rostopic pub 可以把数据发布到当前某个正在广播的话题上。

```
rostopic pub -1 /turtle1/cmd_vel geometry_msgs/Twist -- '[2.0, 0.0, 0.0]' '[0.0, 0.0, 1.8]'
```



这是一个非常复杂的例子，因此让我们来详细分析一下其中的每一个参数。

rostopic pub -----这条命令将会发布消息到某个给定的话题。

-1-----（单个破折号）这个参数选项使 rostopic 发布一条消息后马上退出。

/turtle1/cmd\_vel -----这是消息所发布到的话题名称。

geometry\_msgs/Twist-----这是所发布消息的类型。

—— -----（双破折号）这会告诉命令选项解析器接下来的参数部分都不是命令选项。这在参数里面包含有破折号-（比如负号）时是必须要添加的。

2.0 1.8-----正如之前提到的，在一个 turtlesim/Velocity 消息里面包含有两个浮点型元素：linear 和 angular。在本例中，2.0 是 linear 的值，1.8 是 angular 的值。

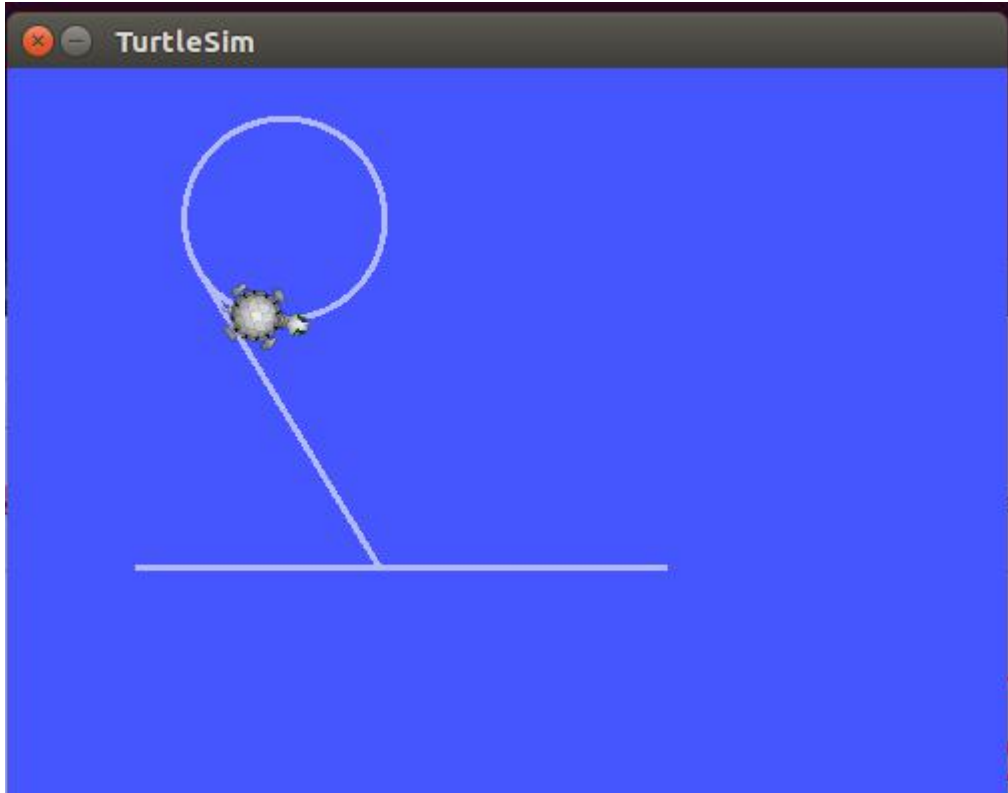
接下来你可能已经注意到 turtle 已经停止移动了。这是因为 turtle 需要一个稳定的频率为 1Hz 的命令流来保持移动状态。我们可以使用 rostopic pub -r 命令来发布一个稳定的命令



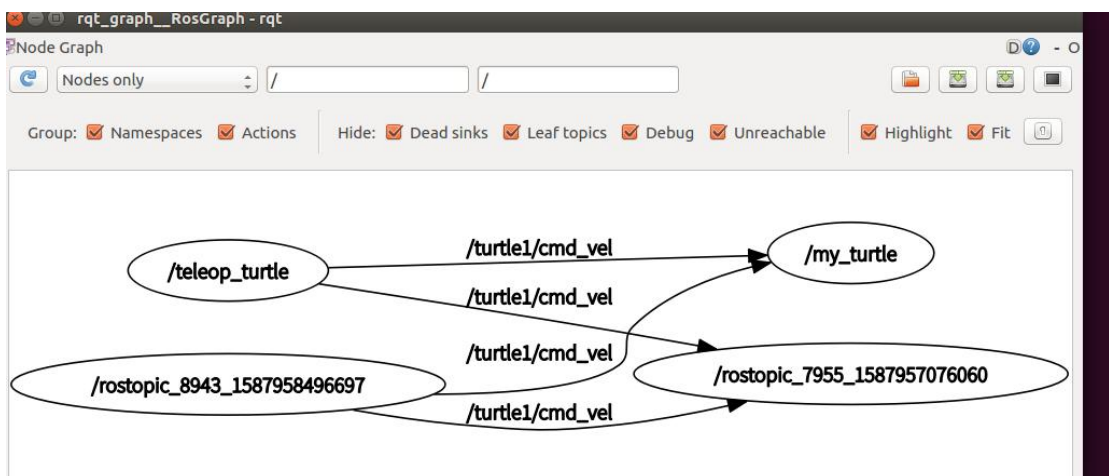
流

```
$ rostopic pub /turtle1/cmd_vel geometry_msgs/Twist -r 1 -- '[2.0, 0.0, 0.0]' '[0.0, 0.0, 1.8]'
```

这条命令以 1Hz 的频率发布速度命令到速度话题上。



我们也可以看一下 rqt\_graph 中的情形，可以看到 rostopic 发布者节点（红色）正在与 rostopic echo 节点（绿色）进行通信：



## 4.2 使用 rostopic hz

rostopic hz 命令可以用来查看数据发布的频率。

我们看一下 turtlesim\_node 发布/turtle/pose 时有多快：

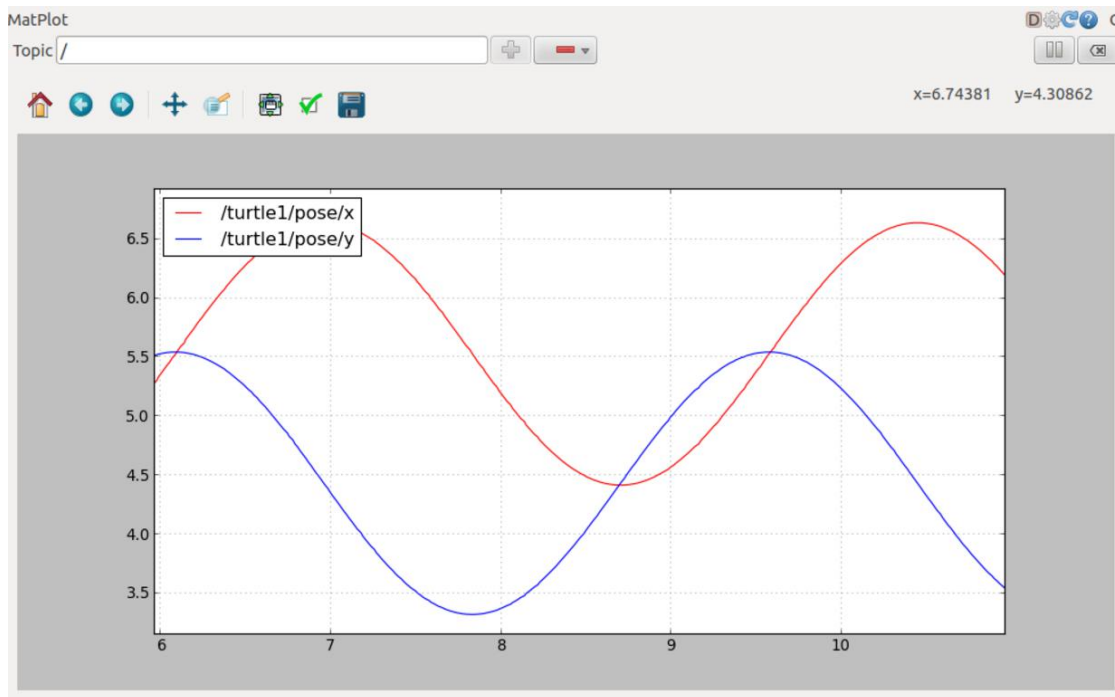
```
$ rostopic hz/turtle/pose
```

## 5. 使用 rqt\_plot

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。这里我们将使用 rqt\_plot 命令来绘制正在发布到/turtle1/pose 话题上的数据变化图形。首先，在一个新终端中运行 rqt\_plot 命令：

```
rostopic pub /turtle1/cmd_vel geometry_msgs/Twist -r 1 -- '[2.0, 0.0, 0.0]' '[0.0, 0.0, 0.0]'
```

```
<launch> <group ns="turtlesim1"> <node pkg="turtlesim" name="sim"
type="turtlesim_node"/> </group> <group ns="turtlesim2"> <node pkg="turtlesim"
name="sim" type="turtlesim_node"/> </group> <node pkg="turtlesim" name="mimic"
type="mimic"> <remap from="input" to="turtlesim1/turtle1"/> <remap from="output"
to="turtlesim2/turtle1"/> </node></launch>
```



按下减号按钮会显示一组菜单让你隐藏图形中指定的话题。现在隐藏掉你刚才添加的话题并添加

`/turtle1/pose/theta`，你会看到如下图所示的图形：

