1. rospack find [包名称]

返回软件包的路径信息

2. roscd [本地包名称[/子目录]]

直接切换工作目录到某个软件包或者软件包集当中

使用roscd 也可以切换到一个软件包或软件包集的子目录中

3. roscd log

使用roscd log可以切换到ROS保存日记文件的目录下。需要注意的是，如果你没有执行过任何ROS程序，系统会报错说该目录不存在

4. rosls [本地包名称[/子目录]]

直接按软件包的名称而不是绝对路径执行ls命令

节点：

一下操作要在终端里面运行roscore，并且在运行其他节点代码的时候不能关掉roscore

1. rosnode list

列出活跃的节点

2. rosrun 用包名直接运行一个包内的节点(而不需要知道这个包的路径)

用法: rosrun [package\_name] [node\_name]

例子：rosrun turtlesim turtlesim\_node

rosrun turtlesim turtlesim\_node \_\_name:=my\_turtle

更换节点名称成为你想要的名称

话题topic：

节点与节点之间就是用话题来通信的

rosrun turtlesim turtle\_teleop\_key 需要通过键盘来控制turtle的运动

1. rostopic 能够获取有关ROS话题的信息

rostopic -h 查看rostopic的子命令

2. rostopic echo可以显示在某个话题上发布的数据

rostopic echo /turtle1/cmd\_vel

在终端中输入这个之后，接下来我们通过按下方向键使turtle\_teleop\_key节点发布数据，就能看到数据了

3. rostopic list能够列出所有当前订阅和发布的话题

rostopic list -h 输入这个就能看到所有的子命令

在rostopic list中使用verbose选项 ，这会显示出有关所发布和订阅的话题及其类型的详细信息

消息messages

1. rostopic type 命令用来查看所发布话题的消息类型

用法：rostopic type [topic]

结合消息使用rostopic

用法：rostopic pub [topic] [msg\_type] [args]

例子：rostopic pub -1 /turtle1/cmd\_vel geometry\_msgs/Twist -- '[2.0, 0.0, 0.0]' '[0.0, 0.0, 1.8]'

以上命令会发送一条消息给turtlesim，告诉它以2.0大小的线速度和1.8大小的角速度开始移动

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/command\_velocity 这是消息所发布到的话题名称。

turtlesim/Velocity 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。这些参数其实是按照YAML语法格式编写的，这在YAML文档中有更多的描述。

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。

Rostopic hz命令可以用来查看数据发布的频率

用法：rostopic hz [topic]

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

rosrun rqt\_plot rqt\_plot

这会弹出一个新窗口，在窗口左上角的一个文本框里面你可以添加需要绘制的话题。在里面输入/turtle1/pose/x后之前处于禁用状态的加号按钮将会被使能变亮。按一下该按钮，并对/turtle1/pose/y重复相同的过程。现在你会在图形中看到turtle的x-y位置坐标图

ROS服务：是节点与节点之间的另外一种通信方式，节点发出请求，并获得一个响应

使用方法:

rosservice list 输出可用服务的信息

rosservice call 调用带参数的服务

rosservice type 输出服务类型

rosservice find 依据类型寻找服务find services by service type

rosservice uri 输出服务的ROSRPC uri

### 使用rqt\_console和rqt\_logger\_level

rqt\_console属于ROS日志框架(logging framework)的一部分，用来显示节点的输出信息。rqt\_logger\_level允许我们修改节点运行时输出信息的日志等级（logger levels）（包括 DEBUG、WARN、INFO和ERROR）。

现在让我们来看一下turtlesim在rqt\_console中的输出信息，同时在rqt\_logger\_level中修改日志等级。在启动turtlesim之前先在另外两个新终端中运行rqt\_console和rqt\_logger\_level：

$ rosrun rqt\_console rqt\_console

$ rosrun rqt\_logger\_level rqt\_logger\_level

roslaunch可以用来启动定义在launch文件中的多个节点

用法：

$ roslaunch [package] [filename.launch]

roswtf检查ROS系统并尝试发现问题