**ROS入门指南**

**版本0.1**

**内容**

## 一 ROS介绍

ROS-Robot Operating System，是一个适用于机器人的开源的元操作系统。它提供了操作系统应有的服务，包括硬件抽象，底层设备控制，常用函数的实现，进程间消息传递，以及包管理。它也提供用于获取、编译、编写、和跨计算机运行代码所需的工具和库函数。

ROS 的主要目标是为机器人研究和开发提供代码复用的支持。ROS是一个分布式的进程（也就是“节点”）框架，这些进程被封装在易于被分享和发布的程序包和功能包中。ROS也支持一种类似于代码储存库的联合系统，这个系统也可以实现工程的协作及发布。这个设计可以使一个工程的开发和实现从文件系统到用户接口完全独立决策（不受ROS限制）。同时，所有的工程都可以被ROS的基础工具整合在一起。

## 二 ROS安装

### **步骤1**

在安装好ubuntu系统之后，首先配置安装ROS需要的key。

Step 1 ：用ctrl+alt+t打开一个终端，在终端中键入

sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb\_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'

Step 2：再在终端中键入

sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116

注意事项：此时可能会遇到无法连接到服务器的问题，可以把代码中的hkp://ha.pool.sks-keyservers.net:80网址换成hkp://pgp.mit.edu:80 或 hkp://keyserver.ubuntu.com:80

### **步骤2**

进行ROS安装

Step 1： 在终端中键入

sudo apt update

注意事项：此时可能会报错如下

Could not get lock/var/lib/dpkg/lock – open

这是因为apt还在运行，解决方法为在终端中键入

sudo rm /var/cache/apt/archives/lock

sudo rm /var/lib/dpkg/lock

Step 2：在终端中键入

sudo apt install ros-melodic-desktop-full

注意事项：此操作完成时间较长

### **步骤3**

对ROS进行初始化和环境设定

Step 1：在终端中键入

sudo rosdep init

rosdep update

Step 2：在终端中键入

echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc

source ~/.bashrc

### **步骤4**

安装一个常用包

Step 1：在终端中键入

sudo apt install python-rosinstall python-rosinstall-generator python-wstool build-essential

## 三 ROS基础使用

### 步骤一

创建一个工作空间

Step 1：在终端中键入

mkdir –p ~/catkin\_ws/src

cd ~/catkin\_ws/src

cd ~/catkin\_ws/

catkin\_make

注意事项：此时我们创建了一个catkin\_ws工作空间，该空间下有一个src文件夹，一个build文件夹和一个devel文件夹。在这个工作空间中，catkin包可以被编译。

### 步骤2

创建一个catkin包

Step 1： 在终端中键入

cd ~/catkin\_ws/src

catkin\_creat\_pkg beginner\_tutorials std\_msgs rospy roscpp

注意事项：一个catkin的包必须包括一个package.xml文件，一个CMakeLists.txt文件，且每个文件夹下只有一包，包之间不允许嵌套。这是一个名为begineer\_tutorials的包。这样我们的第一个包就创建好了。

### 步骤3

ROS有以下几个概念需要理解

* Nodes:节点,一个节点即为一个可执行文件，它可以通过ROS与其它节点进行通信。
* Messages:消息，消息是一种ROS数据类型，用于订阅或发布到一个话题。
* Topics:话题,节点可以发布消息到话题，也可以订阅话题以接收消息。
* Master:节点管理器，ROS名称服务 (比如帮助节点找到彼此)。
* rosout: ROS中相当于stdout/stderr。
* roscore: 主机+ rosout + 参数服务器

我们在运行所有ros程序前需要最先运行roscore。

Step 1：:在终端中键入

roscore

Step 2：使用rosnode可以查看现在活跃的节点。打开一个新的终端，键入

rosnode list

注意事项：它会列出所以当前活跃的节点。此时应该只显示一个rosout，因为这个节点用于收集和记录节点调试输出信息，总是在运行。

Step 3：ROS可以允许使用包名直接运行一个包内的节点。其格式如下

rosrun [package\_name] [node\_name]

Step 4：当节点之间有了通讯时，我们可以查看在某个话题上发布的数据，其格式如下

rostopic echo [topic]

Step 4：我们可以查看所有发布和订阅的话题及其类型的详细信息。命令如下

rostopic list -v

Step 4：使用rostopic pub可以把数据发送到某个正在运行的topic上，其格式如下

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

## 四 使用ROS构建一个简单的控制程序

### 步骤1

建立一个ROS主机

Step 1：打开一个终端，在其中键入

roscore

你会看到类似下面的输出信息

... logging to ~/.ros/log/9cf88ce4b14d11df8a7500251148e8cf/roslaunch-machine\_

name13039.log

Checking log directory for disk usage. This may take awhile.

Press Ctrl-C to interrupt

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

started roslaunch server http://machine\_name:33919/

ros\_comm version 1.4.7

SUMMARY

========

PARAMETERS

\* /rosversion

\* /rosdistro

NODES

auto-starting new master

process[master]: started with pid [13054]

ROS\_MASTER\_URI=http://machine\_name:11311/

setting /run\_id to 9cf88ce4b14d11df8a7500251148e8cf

process[rosout1]: started with pid [13067]

started core service [/rosout]

### 步骤2

查看现在正在工作的ROS节点信息

Step 1：打开一个新终端，在其中输入

rosnode list

你会看到

/rosout

这表明现在只有一个节点在运行：rosout。我们之前说过rosout无论何时都在运行。我们现在可以把这个终端关闭。

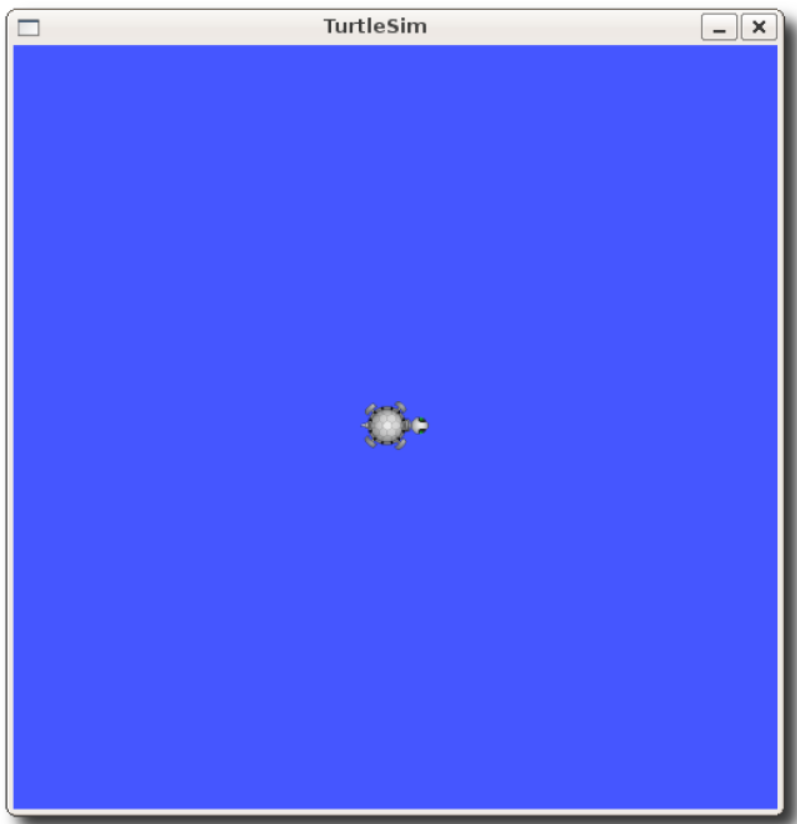
### 步骤3

我们运行一个节点turtlesim节点。ROS允许我们直接使用包名运行一个包内的节点而不需要知道这个包的完整路径。

Step 1：打开一个新的终端。在其中键入

rosrun turtlesim turtlesim\_node

运行之后，我们会看到一个如下图的界面出现。



每个人的乌龟样子或许会不同，这并不需要担心，因为这个乌龟的样子是随机生成的。

Step 2：我们再次检查运行的节点。打开一个新的在终端，在其中键入

rosnode list

你会看到

/rosout

/turtlesim

这表明我们现在有两个节点在运行。

### 步骤4

使用键盘来控制这个乌龟

Step 1：打开一个新的终端，在其中运行

rosrun turtlesim turtle\_teleop\_key

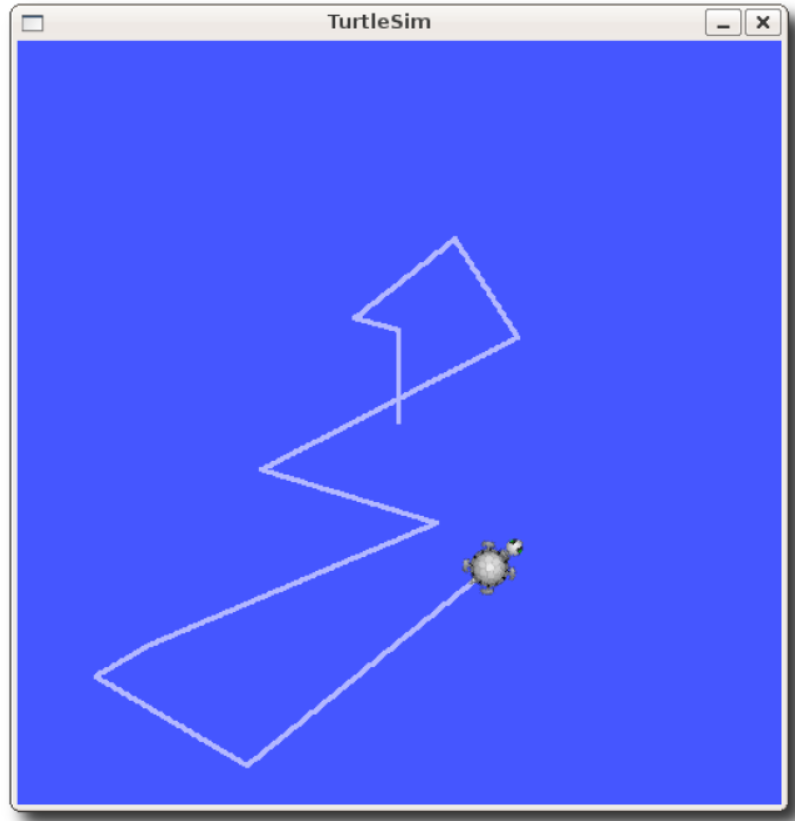
你可以看到返回如下

[ INFO] 1254264546.878445000: Started node [/teleop\_turtle], pid [5528], bound on [aqy], xmlrpc port [43918], tcpros port [55936], logging to [~/ros/ros/log/teleop\_turtle\_5528.log], using [real] time

Reading from keyboard

Use arrow keys to move the turtle.

此时你可以使用键盘上的方向键控制刚才显示的乌龟了。当按动方向键时，乌龟会在图像中移动并且留下轨迹，如下图。



## 五 附录

<http://wiki.ros.org/ROS/Tutorials>

<http://www.ros.org/>

<http://wiki.ros.org/melodic/Installation/Ubuntu>