

ROS Summary 3/28/2016

TURTLESIM PACKAGE Be sure that the turtlesim package is loaded on your system. Open a terminal window and try the following commands:

```
$ rospack find turtlesim
  opt/ros/indigo/share/turtlesim
```

```
$ rosln turtlesim          (List Files - Note no need to type full path)
  cmake images msg package.xml srv
  Read the package.xml, look at the images. Srv directory has services
```

```
$ pwd          (Show working directory – not at turtlesim)
```

```
$ roscd turtlesim
  /opt/ros/indigo/share/turtlesim
```

rospack List Packages

Two of the ROS packages we are using in this report are **std_msgs** and **turtlesim**. The command **rospack list** lists the packages and their directories on the workstation of which std_msgs and turtlesim are only two of many.

```
tlharmanphd@D125-43873:~$ rospack list
.
.
std_msgs /opt/ros/indigo/share/std_msgs
.
turtlesim /opt/ros/indigo/share/turtlesim
.
.
```

Note that the distribution of ROS is Indigo.

To clear the screen of the long list:
tlharmanphd@D125-43873:~\$ **clear**

rospack help

tlharmanphd@D125-43873:/\$ rospack help

USAGE: rospack <command> [options] [package]

Allowed commands:

help
cflags-only-I [--deps-only] [package]
cflags-only-other [--deps-only] [package]
depends [package] (alias: deps)
depends-indent [package] (alias: deps-indent)
depends-manifests [package] (alias: deps-manifests)
depends-msgsrv [package] (alias: deps-msgsrv)
depends-on [package]
depends-on1 [package]
depends-why --target=<target> [package] (alias: deps-why)
depends1 [package] (alias: deps1)
export [--deps-only] --lang=<lang> --attrib=<attrib> [package]
find [package]
langs
libs-only-L [--deps-only] [package]
libs-only-l [--deps-only] [package]
libs-only-other [--deps-only] [package]
list
list-duplicates
list-names
plugins --attrib=<attrib> [--top=<toppkg>] [package]
profile [--length=<length>] [--zombie-only]
rosdep [package] (alias: rosdeps)
rosdep0 [package] (alias: rosdeps0)
vcs [package]
vcs0 [package]
Extra options:
-q Quiets error reports.

If [package] is omitted, the current working directory is used (if it contains a manifest.xml).

tlharmanphd@D125-43873:~\$ rospack depends turtlesim

cpp_common

roetime
roscpp_traits
roscpp_serialization
genmsg
genpy
message_runtime
rosconsole
std_msgs
rosgraph_msgs
xmlrpcpp
roscpp
catkin
rospack
roslib
std_srvs

```
tlharmanphd@D125-43873:~$ rostopic list
/roscout
/roscout_agg
```

Create a Topic /hello and A String Message with rostopic pub

```
tlharmanphd@D125-43873:~$ rostopic pub /hello std_msgs/String "Hello User"
publishing and latching message. Press ctrl-C to terminate
```

New Topic /hello from rostopic list. In a new window, note that the topic **/hello** is present using **rostopic list**.

```
tlharmanphd@D125-43873:~$ rostopic list
/hello
/roscout
/roscout_agg
```

Data for Topic /hello- message

```
tlharmanphd@D125-43873:~$ rostopic echo /hello
data: Hello User
```

New Node caused by rostopic pub

```
tlharmanphd@D125-43873:~$ rosnode list
/roscout
/rostopic_4375_1424715269456
```

Thus far, we have a new topic **/hello** and a new node **/rostopic_4375_1424715269456**. The topic number will change with each run since it indicates time as well as other information.

```
tlharmanphd@D125-43873:~$ rosnode info /rostopic_4375_1424715269456
```

```
-----
Node [/rostopic_4375_1424715269456]
Publications:
* /hello [std_msgs/String]

Subscriptions: None
Services:
* /rostopic_4375_1424715269456/set_logger_level
* /rostopic_4375_1424715269456/get_loggers
```

```
contacting node http://D125-43873:36024/ ...
```

```
Pid: 4375
```

```
Connections:
```

```
* topic: /hello
* to: /rostopic_4504_1424715395163
* direction: outbound
```

```
⑩ transport: TCPROS
```

This created the **topic /hello** with **topic type std_msgs/String**. **Hello User** is the data.

```
tlharmanphd@D125-43873:~$ roslaunch turtlesim turtlesim_node
[ INFO] [1427212356.117628994]: Starting turtlesim with node name /turtlesim
[ INFO] [1427212356.121407419]: Spawning turtle [turtle1] at x=[5.544445],
y=[5.544445], theta=[0.000000]
```

Messages

```
tlharmanphd@D125-43873:/$ rosmmsg help
rosmmsg is a command-line tool for displaying information about ROS Message types.
```

```
Commands:
  rosmmsg show      Show message description
  rosmmsg list      List all messages
  rosmmsg md5       Display message md5sum
  rosmmsg package   List messages in a package
  rosmmsg packages  List packages that contain messages
```

```
Type rosmmsg <command> -h for more detailed usage
```

If a topic publishes a message, we can determine the message type and read the message. This is shown in the example to determine the color of the background for the turtle.

```
tlharmanphd@D125-43873:~$ rostopic type /turtle1/color_sensor
turtlesim/Color
```

```
turtlebot-0877@Turtlebot-0877:~$ rosmmsg package turtlesim
turtlesim/Color
turtlesim/Pose
```

```
turtlebot-0877@Turtlebot-0877:~$ rosmmsg package turtlesim
turtlesim/Color
turtlesim/Pose
```

```
tlharmanphd@D125-43873:~$ rostopic echo /turtle1/color_sensor
```

```
r: 69
g: 86
b: 255
```

```
---
```

```
r: 69
g: 86
b: 255
```

```
.
```

```
.
```

rosparam help

```
tlharmanphd@D125-43873:/$ rosparam help
```

rosparam is a command-line tool for getting, setting, and deleting parameters from the ROS Parameter Server.

Commands:

```
rosparam set      set parameter
rosparam get      get parameter
rosparam load     load parameters from file
rosparam dump     dump parameters to file
rosparam delete   delete parameter
rosparam list     list parameter names
```

```
tlharmanphd@D125-43873:~$ roscore
```

(Start ROS Master)

```
tlharmanphd@D125-43873:~$ rosparam list
```

```
/rostdistro
/roslaunch/uris/host_d125_43873__39549
/rosversion
/run_id
```

```
tlharmanphd@D125-43873:~$ rosparam get rosversion
```

```
'1.11.10 (1.11.16 2/08/2016)
```

```
tlharmanphd@D125-43873:~$ rosparam get rostdistro
```

```
'indigo
```

As the node publishes, the color of the background for example, it is possible to change the parameters. The command format is

rosparam list for turtlesim node

To list the parameters for the turtlesim node:

```
tlharmanphd@D125-43873:/$ rosparam list
```

```
/background_r
/background_g
/background_b
/rostdistro
/roslaunch/uris/host_d125_43873__51759
/rosversion
```

rosparam get

```
tlharmanphd@D125-43873:~$ rosparam get /
```

```
background_b: 255
background_g: 86
background_r: 69
rostdistro: 'indigo
roslaunch:
  uris: {host_d125_43873__60512: 'http://D125-43873:60512/'}
rosversion: '1.11.10
run_id: 2429b792-d23c-11e4-b9ee-3417ebbca982
```

rosparam set

Change the colors:

```
tlharmanphd@D125-43873:/$ rosparam set background_b 0
```

```
tlharmanphd@D125-43873:/$ rosparam set background_g 0
```

```
tlharmanphd@D125-43873:/$ rosparam set background_r 255
```

```
tlharmanphd@D125-43873:/$ rosservice call /clear
```

tlharmanphd@D125-43873:/\$ rosservice help

Commands:

rosservice args	print service arguments
rosservice call	call the service with the provided arguments
rosservice find	find services by service type
rosservice info	print information about service
rosservice list	list active services
rosservice type	print service type
rosservice uri	print service ROSRPC uri

Type `rosservice <command> -h` for more detailed usage, e.g. `'rosservice call -h'`

Use the **\$rosservice list** command to see the services for the active node.

TURTLEBOT2 Guide I Motion ROS Indigo 2/19/2016

```
$ roslaunch turtlebot_bringup minimal.launch
```

```
$ roslaunch turtlebot_teleop keyboard_teleop.launch
```

```
$ roslaunch turtlebot_teleop xbox360_teleop.launch
```

```
tlharmanphd@D125-43873:~$ rospack list turtle <Tab> <Tab>
```

turtle_actionlib	turtlebot_navigation
turtlebot_actions	turtlebot_panorama
turtlebot_bringup	turtlebot_rapps
turtlebot_calibration	turtlebot_rviz_launchers
turtlebot_capabilities	turtlebot_stage
turtlebot_dashboard	turtlebot_stdr
turtlebot_description	turtlebot_teleop
turtlebot_follower	turtlesim

```
turtlebot_gazebo          turtle_tf
```

```
turtlebot_interactive_markers turtle_tf2
```

```
turtlebot_msgs
```

```
tlharmanphd@D125-43873:~$ roscnode list
```

```
/app_manager  
/bumper2pointcloud  
/capability_server  
/capability_server_nodelet_manager  
/cmd_vel_mux  
/diagnostic_aggregator  
/interactions  
/master  
/mobile_base  
/mobile_base_nodelet_manager  
/robot_state_publisher  
/rosout  
/turtlebot_laptop_battery  
/zeroconf/zeroconf
```

TOPICS for TurtleBot

```
tlharmanphd@D125-43873:~$ rostopic list
```

```
/capability_server/bonds  
/capability_server/events  
/cmd_vel_mux/active  
/cmd_vel_mux/input/navi  
/cmd_vel_mux/input/safety_controller  
/cmd_vel_mux/input/teleop  
/cmd_vel_mux/parameter_descriptions  
/cmd_vel_mux/parameter_updates  
.  
  
/joint_states  
/laptop_charge  
/mobile_base/commands/controller_info  
/mobile_base/commands/digital_output  
/mobile_base/commands/external_power  
/mobile_base/commands/led1  
/mobile_base/commands/led2  
/mobile_base/commands/motor_power  
/mobile_base/commands/reset_odometry  
/mobile_base/commands/sound  
/mobile_base/commands/velocity  
/mobile_base/controller_info  
/mobile_base/debug/raw_control_command  
/mobile_base/debug/raw_data_command  
/mobile_base/debug/raw_data_stream  
/mobile_base/events/bumper  
/mobile_base/events/button  
/mobile_base/events/cliff
```


Messages that involve the Kobuki base:

```
tlharmanphd@D125-43873:~$ rosmmsg list | grep kobuki
```

```
kobuki_msgs/AutoDockingAction  
kobuki_msgs/AutoDockingActionFeedback  
kobuki_msgs/AutoDockingActionGoal  
kobuki_msgs/AutoDockingActionResult  
kobuki_msgs/AutoDockingFeedback  
kobuki_msgs/AutoDockingGoal  
kobuki_msgs/AutoDockingResult  
kobuki_msgs/BumperEvent  
kobuki_msgs/ButtonEvent  
kobuki_msgs/CliffEvent  
kobuki_msgs/ControllerInfo  
kobuki_msgs/DigitalInputEvent  
kobuki_msgs/DigitalOutput  
kobuki_msgs/DockInfraRed  
kobuki_msgs/ExternalPower  
kobuki_msgs/KeyboardInput  
kobuki_msgs/Led  
kobuki_msgs/MotorPower  
kobuki_msgs/PowerSystemEvent  
kobuki_msgs/RobotStateEvent  
kobuki_msgs/ScanAngle  
kobuki_msgs/SensorState  
kobuki_msgs/Sound  
kobuki_msgs/VersionInfo  
kobuki_msgs/WheelDropEvent  
tlharmanphd@D125-43873:~$
```

Messages for position, orientation, etc. Commands and responses

```
tlharmanphd@D125-43873:~$ rosmmsg list | grep geometry
```

```
geometry_msgs/Accel  
geometry_msgs/AccelStamped  
geometry_msgs/AccelWithCovariance  
geometry_msgs/AccelWithCovarianceStamped  
geometry_msgs/Inertia  
geometry_msgs/InertiaStamped  
geometry_msgs/Point  
geometry_msgs/Point32  
geometry_msgs/PointStamped  
geometry_msgs/Polygon  
geometry_msgs/PolygonStamped  
geometry_msgs/Pose  
geometry_msgs/Pose2D  
geometry_msgs/PoseArray  
geometry_msgs/PoseStamped  
geometry_msgs/PoseWithCovariance  
geometry_msgs/PoseWithCovarianceStamped  
geometry_msgs/Quaternion  
geometry_msgs/QuaternionStamped  
geometry_msgs/Transform
```

geometry_msgs/TransformStamped
geometry_msgs/Twist
geometry_msgs/TwistStamped
geometry_msgs/TwistWithCovariance
geometry_msgs/TwistWithCovarianceStamped
geometry_msgs/Vector3
geometry_msgs/Vector3Stamped
geometry_msgs/Wrench
geometry_msgs/WrenchStamped

tlharmanphd@D125-43873:~\$ **rosmmsg list | grep turtle**

turtle_actionlib/ShapeAction
turtle_actionlib/ShapeActionFeedback
turtle_actionlib/ShapeActionGoal
turtle_actionlib/ShapeActionResult
turtle_actionlib/ShapeFeedback
turtle_actionlib/ShapeGoal
turtle_actionlib/ShapeResult
turtle_actionlib/Velocity
turtlebot_actions/FindFiducialAction
turtlebot_actions/FindFiducialActionFeedback
turtlebot_actions/FindFiducialActionGoal
turtlebot_actions/FindFiducialActionResult
turtlebot_actions/FindFiducialFeedback
turtlebot_actions/FindFiducialGoal
turtlebot_actions/FindFiducialResult
turtlebot_actions/TurtlebotMoveAction
turtlebot_actions/TurtlebotMoveActionFeedback
turtlebot_actions/TurtlebotMoveActionGoal
turtlebot_actions/TurtlebotMoveActionResult
turtlebot_actions/TurtlebotMoveFeedback
turtlebot_actions/TurtlebotMoveGoal
turtlebot_actions/TurtlebotMoveResult
turtlebot_calibration/ScanAngle
turtlebot_msgs/PanoramaImg
turtlesim/Color
turtlesim/Pose

tlharmanphd@D125-43873:~\$ **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
```

Rosmmsg show geometry_msgs/Pose

tlharmanphd@D125-43873:~\$ **rosmmsg show geometry_msgs/Pose**

```
geometry_msgs/Point position
float64 x
float64 y
float64 z
geometry_msgs/Quaternion orientation
float64 x
float64 y
float64 z
float64 w
```

<http://wiki.ros.org/msg>

Rosmmsg show nav_msgs/Odometry

tlharmanphd@D125-43873:~\$ **rosmmsg show nav_msgs/Odometry**

```
std_msgs/Header header
uint32 seq
time stamp
string frame_id
string child_frame_id
geometry_msgs/PoseWithCovariance pose
geometry_msgs/Pose pose
geometry_msgs/Point position
float64 x
float64 y
float64 z
geometry_msgs/Quaternion orientation
float64 x
float64 y
float64 z
float64 w
float64[36] covariance
geometry_msgs/TwistWithCovariance twist
geometry_msgs/Twist twist
geometry_msgs/Vector3 linear
float64 x
float64 y
float64 z
geometry_msgs/Vector3 angular
float64 x
float64 y
float64 z
float64[36] covariance
```

RE-START MINIMAL LAUNCH SET X=0, Y=0. OUTPUT TO TEXT FILE

```
tlharmanphd@D125-43873:~$ rostopic echo /odom/pose/pose >> tb_pose_test1.txt
```

```
position:  
x: 0.0  
y: 0.0  
z: 0.0  
orientation:  
x: 0.0  
y: 0.0  
z: 0.0  
w: 1.0
```

FINAL – MOVE IN STRAIGHT LINE ABOUT 1.2 METERS

```
position:  
x: 1.22930107254  
y: -0.0141608381814  
z: 0.0  
orientation:  
x: 0.0  
y: 0.0  
z: -0.00741758129944  
w: 0.999972489365
```

GO BACK

```
pose:  
position:  
x: 0.0111620718896  
y: -0.053471637895  
z: 0.0  
orientation:  
x: 0.0  
y: 0.0  
z: 0.0312363117969  
w: 0.999512027354  
covariance: [0.1, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.1, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.7976931348623157e+308, 0.0,  
0.0, 0.0, 0.0, 0.0, 0.0, 1.7976931348623157e+308, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.7976931348623157e+308,  
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.05]
```

```

#!/usr/bin/env python          python_GoInCircle

# A very basic TurtleBot script that moves TurtleBot InCircle
indefinitely. Press CTRL + C to stop.  To run:
# On TurtleBot:
# $ roslaunch turtlebot_bringup minimal.launch
# On work station: $ python python_GoInCircle

import rospy
from geometry_msgs.msg import Twist

class GoInCircle():
    def __init__(self):
        # initiliaze
        rospy.init_node('GoInCircle', anonymous=False) # NODE

# tell user how to stop TurtleBot
        rospy.loginfo("To stop TurtleBot CTRL + C")

# What function to call when you ctrl + c
        rospy.on_shutdown(self.shutdown)

# Create a publisher which can "talk" to TurtleBot and tell it to move
# Tip: You may need to change cmd_vel_mux/input/navi to /cmd_vel if
you're not using TurtleBot2
        self.cmd_vel = rospy.Publisher('cmd_vel_mux/input/navi',
Twist, queue_size=10) # Topic

#TurtleBot will stop if we don't keep telling it to move.  How often
should we tell it to move? 10 HZ
        r = rospy.Rate(10);

# Twist is a datatype for velocity
        move_cmd = Twist()
# let's go forward at 0.2 m/s
        move_cmd.linear.x = 0.2
# let's turn at 1.0 radians/s About 6 seconds to complete circle
        move_cmd.angular.z = 1.0

# as long as you haven't ctrl + c keeping doing...
        while not rospy.is_shutdown():
            # publish the velocity
            self.cmd_vel.publish(move_cmd)
            # wait for 0.1 seconds (10 HZ) and publish again
            r.sleep()

    def shutdown(self):
        # stop turtlebot
        rospy.loginfo("Stop TurtleBot")

```

```
# a default Twist has linear.x of 0 and angular.z of 0. So it'll stop
TurtleBot
    self.cmd_vel.publish(Twist())
# sleep just makes sure TurtleBot receives the stop command prior to
shutting down the script
    rospy.sleep(1)

if __name__ == '__main__':
    try:
        GoInCircle()
    except:
        rospy.loginfo("GoInCircle node terminated.")
```

See the new node **/GoInCircle**

tlharmanphd@D125-43873:~\$. ./turtlebot2

tlharmanphd@D125-43873:~\$ **rostopic list**

```
/GoInCircle
/app_manager
/bumper2pointcloud
/capability_server
/capability_server_nodelet_manager
/cmd_vel_mux
/diagnostic_aggregator
/interactions
/master
/mobile_base
```