Communication with Marvelmind devices using ROS (Robot Operating System).

Version 2019.08.17

Marvelmind supplies ROS package, which is able to communicate with mobile beacon and provide location data received from it. We suppose you are using ROS Kinetic installed under Ubuntu Linux. The ROS package is available in repository by link:
https://bitbucket.org/marvelmind_robotics/ros_marvelmind_package
To install the package in the ROS system, at first execute the step 1 of ROS tutorial (beginner level):
http://wiki.ros.org/ROS/Tutorials
Then create the directory for the package by executing commands in the terminal:

```
$ cd ~/catkin_ws/src
$ mkdir marvelmind_nav
```

Then copy downloaded sources from the repository into the created directory.
Before running the software from the package, you should execute following command from the 'catkin_ws' directory:

```
$ source devel/setup.bash
```

After this, you can build and install Marvelmind ROS package: execute from ‘catkin_ws’ directory:

```
$ catkin_make
$ catkin_make install
```

Now we are ready to run the software, but before it we need to prepare the marvelmind system.
Use another PC with dashboard software to build map as described in the operating manual:
http://marvelmind.com/pics/marvelmind_navigation_system_manual.pdf
Then connect the mobile beacon (hedgehog) by the USB cable to the USB port of your ROS machine.
Execute command in terminal to find the virtual serial port used by the hedgehog:

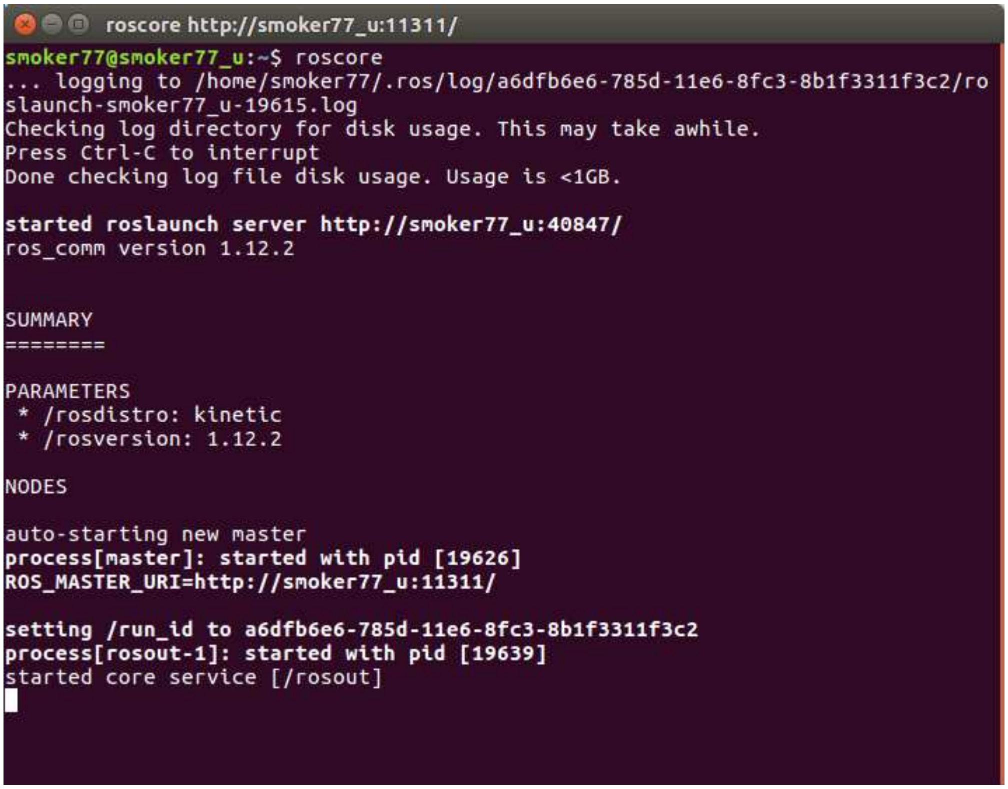
```
$ ls /dev/ttyACM*
```

In most cases, the hedgehog connects to "/dev/ttyACM0", this port is used by default in the Marvelmind ROS software. If no ports found by this command, try another one:

```
$ ls /dev/ttyUSB*
```

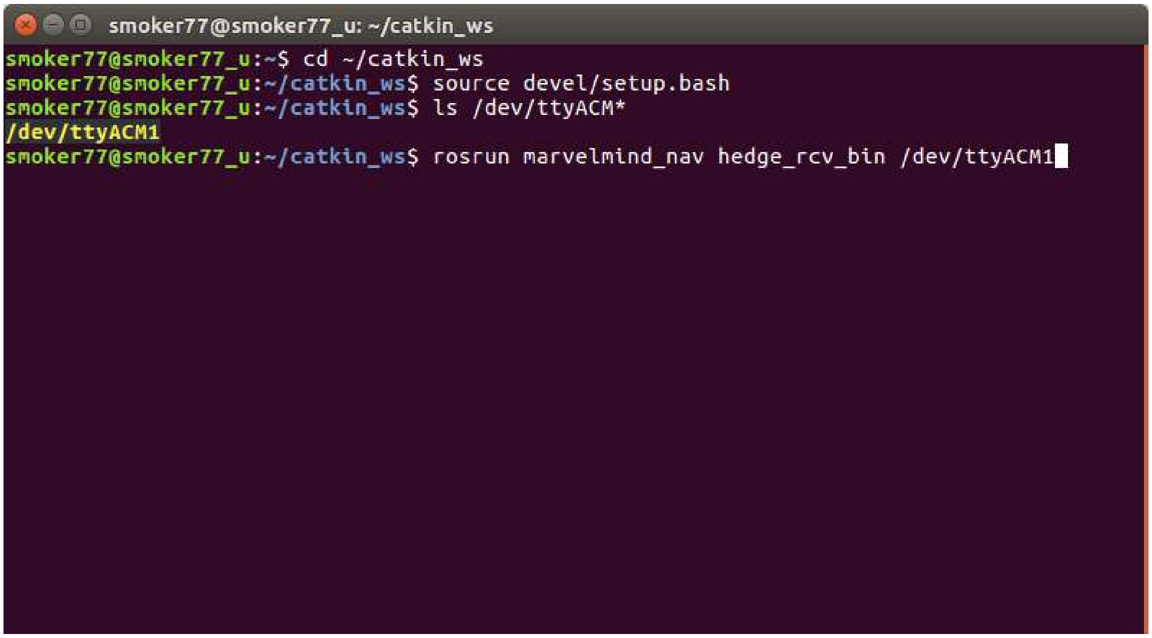
Before running Marvelmind ROS software, run the ROS server in separate terminal:

$ roscore



```
smoker77@smoker77_u:~$ roscore
... logging to /home/smoker77/.ros/log/a6dfb6e6-785d-11e6-8fc3-8b1f3311f3c2/ro
slaunch-smoker77_u-19615.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://smoker77_u:40847/
ros_comm version 1.12.2


SUMMARY
========

PARAMETERS
 * /rosdistro: kinetic
 * /rosversion: 1.12.2

NODES

auto-starting new master
process[master]: started with pid [19626]
ROS_MASTER_URI=http://smoker77_u:11311/

setting /run_id to a6dfb6e6-785d-11e6-8fc3-8b1f3311f3c2
process[rosout-1]: started with pid [19639]
started core service [/rosout]
```

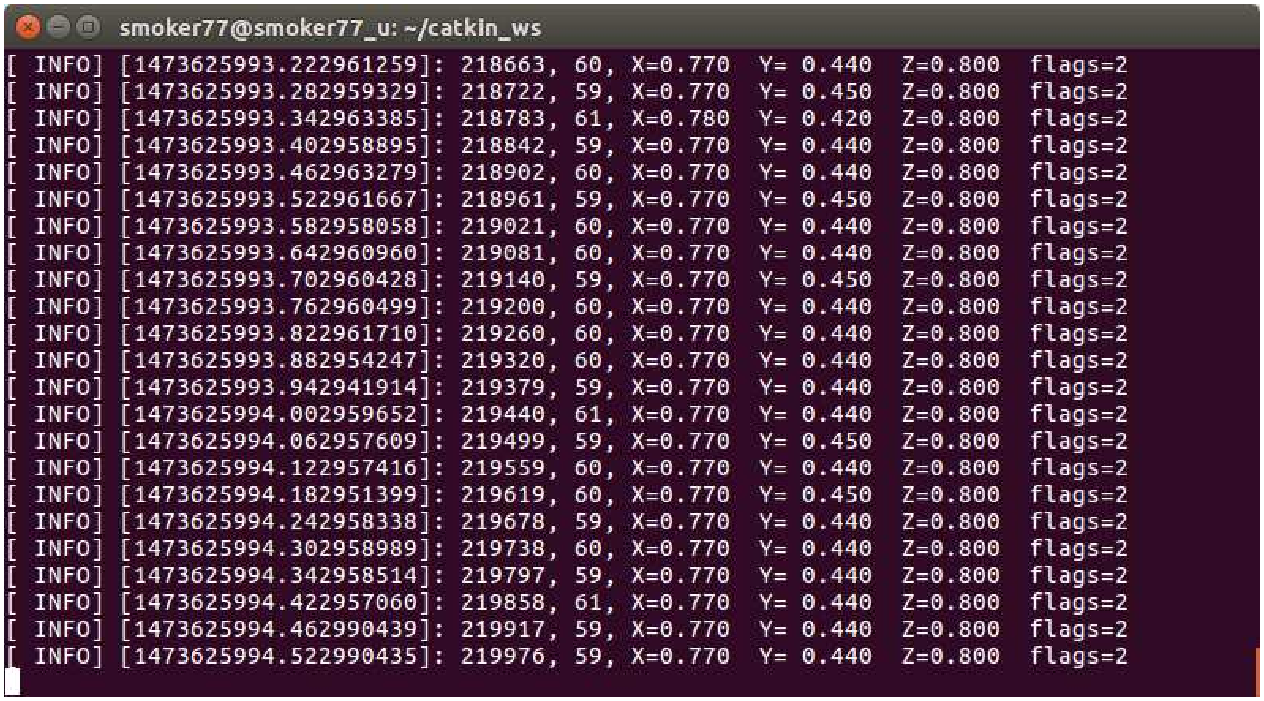
Then run the node **'hedge_rcv_bin'** for receiving data from hedgehog as shown on following screenshot. Note the parameter of the running program is '/dev/ttyACM1', the name of virtual serial port detected by previous command. If the port is '/dev/ttyACM0', this parameter can be skipped.



```
smoker77@smoker77_u:~$ cd ~/catkin_ws
smoker77@smoker77_u:~/catkin_ws$ source devel/setup.bash
smoker77@smoker77_u:~/catkin_ws$ ls /dev/ttyACM*
/dev/ttyACM1
smoker77@smoker77_u:~/catkin_ws$ rosrun marvelmind_nav hedge_rcv_bin /dev/ttyACM1
```

You can get message like ‘unable to open serial connection’ even if the serial port is present. This may mean you have no permissions to access this port. You can get all permissions by command ‘**sudo chmod 0777 /dev/ttyACM0**’. But you will lose the permissions after next reboot. For permanent permissions you can add user to **dialout** group as described here: https://askubuntu.com/questions/58119/changing-permissions-on-serial-port

If the node successfully receives data from hedgehog, it outputs the location data to the terminal as shown on screenshot:

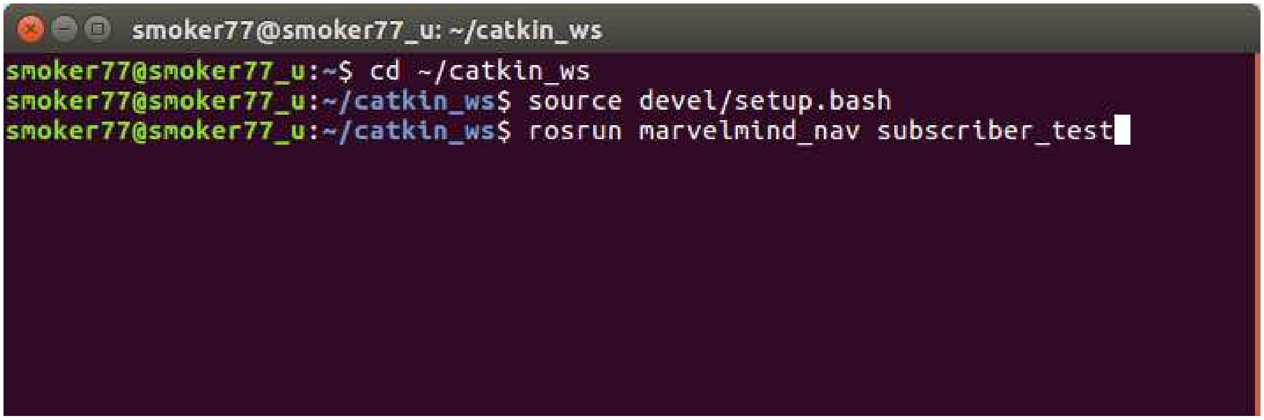


```
smoker77@smoker77_u: ~/catkin_ws
[ INFO] [1473625993.222961259]: 218663, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.282959329]: 218722, 59, X=0.770  Y= 0.450  Z=0.800  flags=2
[ INFO] [1473625993.342963385]: 218783, 61, X=0.780  Y= 0.420  Z=0.800  flags=2
[ INFO] [1473625993.402958895]: 218842, 59, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.462963279]: 218902, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.522961667]: 218961, 59, X=0.770  Y= 0.450  Z=0.800  flags=2
[ INFO] [1473625993.582958058]: 219021, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.642960960]: 219081, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.702960428]: 219140, 59, X=0.770  Y= 0.450  Z=0.800  flags=2
[ INFO] [1473625993.762960499]: 219200, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.822961710]: 219260, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.882954247]: 219320, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625993.942941914]: 219379, 59, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.002959652]: 219440, 61, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.062957609]: 219499, 59, X=0.770  Y= 0.450  Z=0.800  flags=2
[ INFO] [1473625994.122957416]: 219559, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.182951399]: 219619, 60, X=0.770  Y= 0.450  Z=0.800  flags=2
[ INFO] [1473625994.242958338]: 219678, 59, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.302958989]: 219738, 60, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.342958514]: 219797, 59, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.422957060]: 219858, 61, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.462990439]: 219917, 59, X=0.770  Y= 0.440  Z=0.800  flags=2
[ INFO] [1473625994.522990435]: 219976, 59, X=0.770  Y= 0.440  Z=0.800  flags=2
```

First value in square brackets is a ROS timestamp, then hedgehog timestamp in milliseconds, time (in milliseconds) between position samples, coordinates X,Y,Z in meters, and byte of flags.

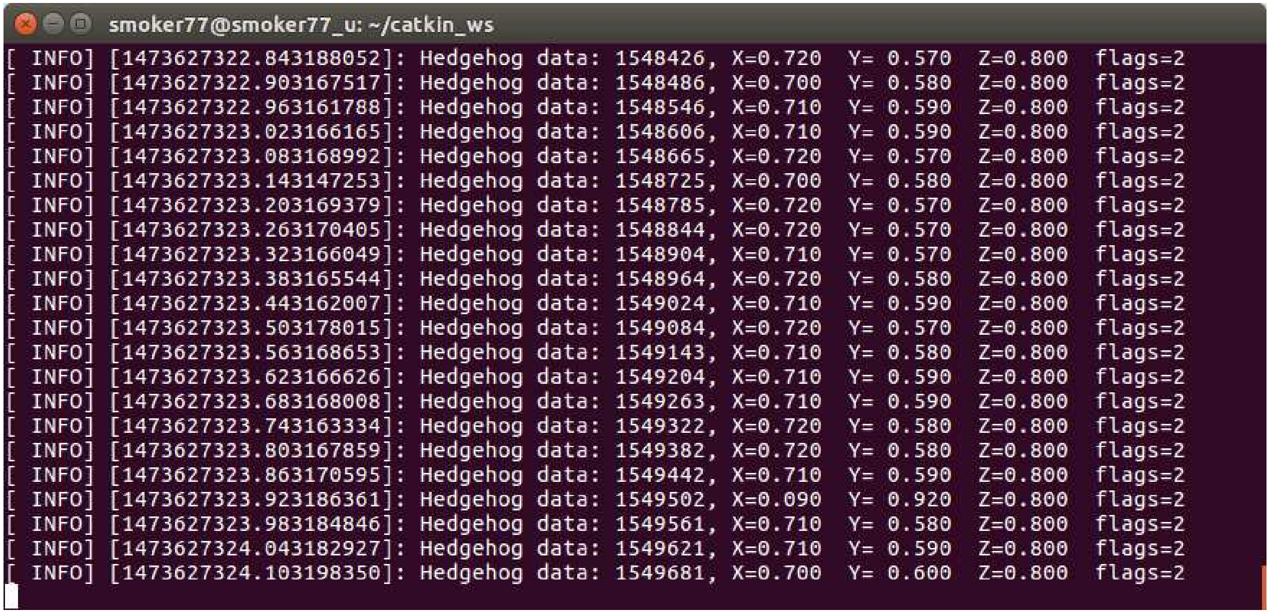
The node **'hedge_rcv_bin'** works as ROS publisher, it sends the message with location data to the topic named **'/hedge_pos'**. The package contains another node **'subscriber_test'**, which is working as ROS subscriber and receiving data from the topic. This node can be used for test purposes and as basis for user software.

Run the **'subscriber_test'** node in separate terminal as shown on following screenshot**:**



```
smoker77@smoker77_u: ~/catkin_ws
smoker77@smoker77_u:~$ cd ~/catkin_ws
smoker77@smoker77_u:~/catkin_ws$ source devel/setup.bash
smoker77@smoker77_u:~/catkin_ws$ rosrun marvelmind_nav subscriber_test
```

The running **'subscriber_test'** node outputs to the terminal received location data from the topic **'/hedge_pos'** as shown on the next screenshot:

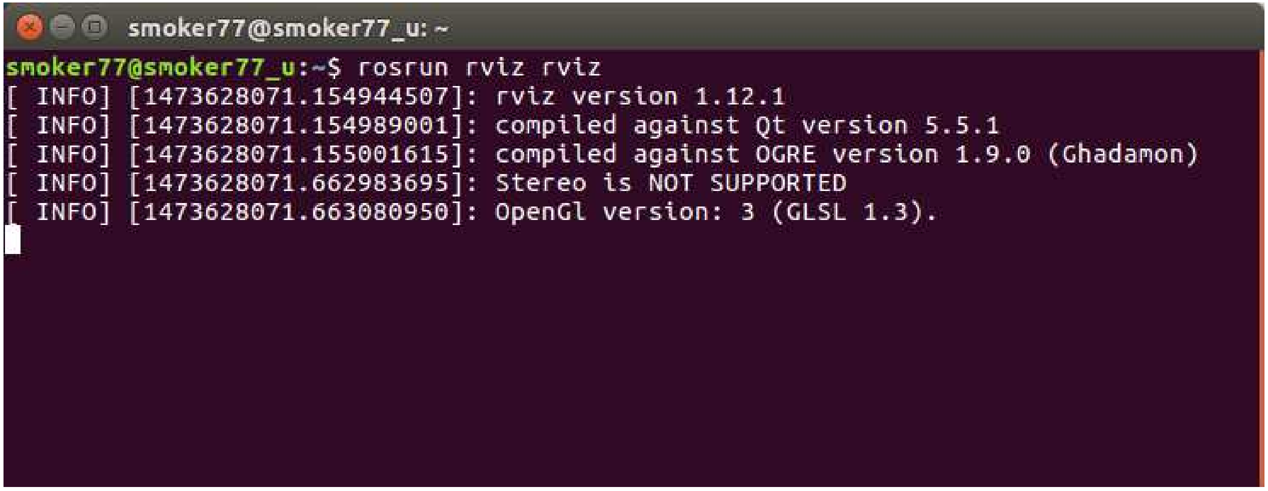


smoker77@smoker77_u: ~/catkin_ws

```
[ INFO] [1473627322.843188052]: Hedgehog data: 1548426, X=0.720  Y= 0.570  Z=0.800  flags=2
[ INFO] [1473627322.903167517]: Hedgehog data: 1548486, X=0.700  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627322.963161788]: Hedgehog data: 1548546, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627323.023166165]: Hedgehog data: 1548606, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627323.083168992]: Hedgehog data: 1548665, X=0.720  Y= 0.570  Z=0.800  flags=2
[ INFO] [1473627323.143147253]: Hedgehog data: 1548725, X=0.700  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627323.203169379]: Hedgehog data: 1548785, X=0.720  Y= 0.570  Z=0.800  flags=2
[ INFO] [1473627323.263170405]: Hedgehog data: 1548844, X=0.720  Y= 0.570  Z=0.800  flags=2
[ INFO] [1473627323.323166049]: Hedgehog data: 1548904, X=0.710  Y= 0.570  Z=0.800  flags=2
[ INFO] [1473627323.383165544]: Hedgehog data: 1548964, X=0.720  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627323.443162007]: Hedgehog data: 1549024, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627323.503178015]: Hedgehog data: 1549084, X=0.720  Y= 0.570  Z=0.800  flags=2
[ INFO] [1473627323.563168653]: Hedgehog data: 1549143, X=0.710  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627323.623166626]: Hedgehog data: 1549204, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627323.683168008]: Hedgehog data: 1549263, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627323.743163334]: Hedgehog data: 1549322, X=0.720  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627323.803167859]: Hedgehog data: 1549382, X=0.720  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627323.863170595]: Hedgehog data: 1549442, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627323.923186361]: Hedgehog data: 1549502, X=0.090  Y= 0.920  Z=0.800  flags=2
[ INFO] [1473627323.983184846]: Hedgehog data: 1549561, X=0.710  Y= 0.580  Z=0.800  flags=2
[ INFO] [1473627324.043182927]: Hedgehog data: 1549621, X=0.710  Y= 0.590  Z=0.800  flags=2
[ INFO] [1473627324.103198350]: Hedgehog data: 1549681, X=0.700  Y= 0.600  Z=0.800  flags=2
```

In addition this node works as publisher and sends data to topic "**visualization_marker**". This allows to view the position in the standard ROS software '**rviz**'.

Run the 'rviz' in separate terminal as shown on screenshot:

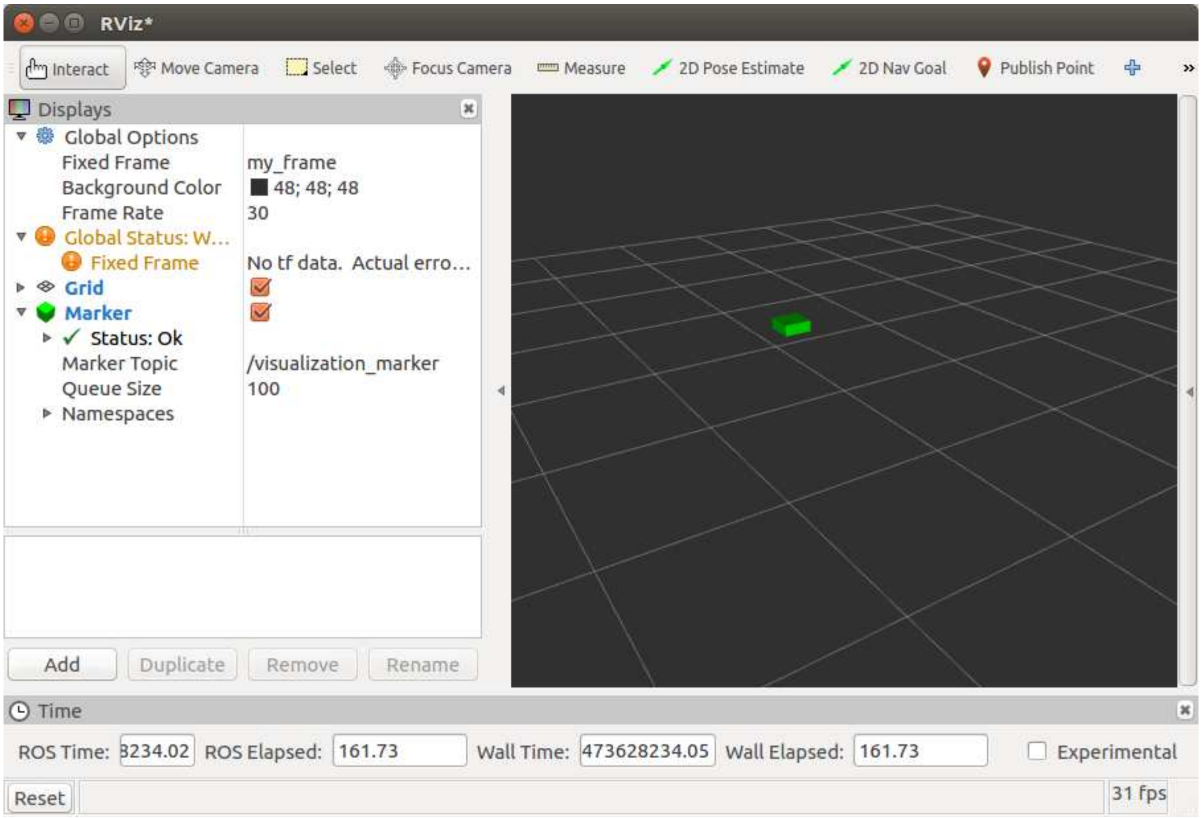


smoker77@smoker77_u: ~

```
smoker77@smoker77_u:~$ rosrun rviz rviz
[ INFO] [1473628071.154944507]: rviz version 1.12.1
[ INFO] [1473628071.154989001]: compiled against Qt version 5.5.1
[ INFO] [1473628071.155001615]: compiled against OGRE version 1.9.0 (Ghadamon)
[ INFO] [1473628071.662983695]: Stereo is NOT SUPPORTED
[ INFO] [1473628071.663080950]: OpenGl version: 3 (GLSL 1.3).
```

The GUI window, shown on the next screenshot, should appear.

To see the hedgehog, make sure the 'Fixed frame' parameter has value 'my_frame', and the marker 'visualization marker' is connected.

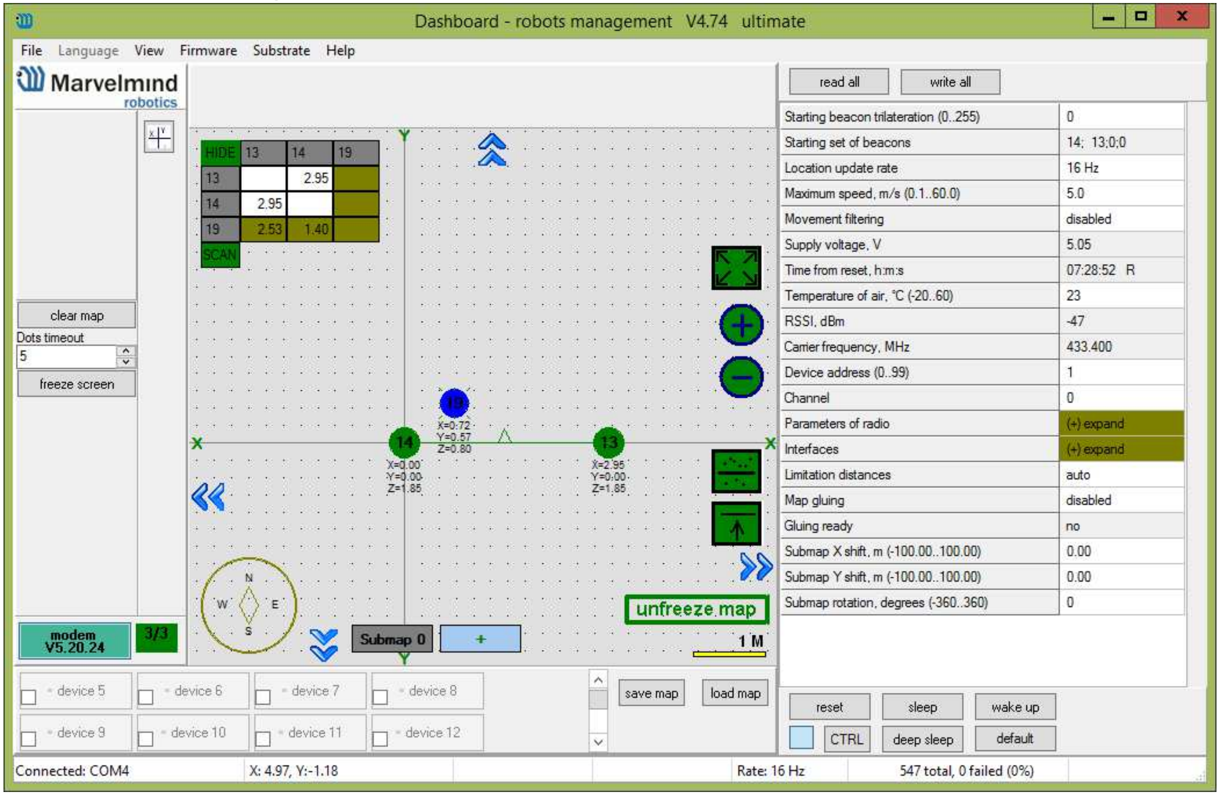


- Global Options
  - Fixed Frame: my_frame
  - Background Color: 48; 48; 48
  - Frame Rate: 30
- Global Status: W…
  - Fixed Frame: No tf data. Actual erro…
- Grid: ☑
- Marker: ☑
  - Status: Ok
  - Marker Topic: /visualization_marker
  - Queue Size: 100
  - Namespaces

ROS Time: 8234.02 ROS Elapsed: 161.73 Wall Time: 473628234.05 Wall Elapsed: 161.73 □ Experimental

31 fps

The next screenshot shows the dashboard window on another computer with system, used for the described above testing of ROS.



Dots timeout: 5

HIDE	13	14	19
13		2.95	
14	2.95		
19	2.53	1.40	

14: X=0.00 Y=0.00 Z=1.85

10: X=0.72 Y=0.57 Z=0.80

13: X=2.95 Y=0.00 Z=1.85

Starting beacon trilateration (0..255)	0
Starting set of beacons	14; 13;0;0
Location update rate	16 Hz
Maximum speed, m/s (0.1..60.0)	5.0
Movement filtering	disabled
Supply voltage, V	5.05
Time from reset, h:m:s	07:28:52 R
Temperature of air, °C (-20..60)	23
RSSI, dBm	-47
Carrier frequency, MHz	433.400
Device address (0..99)	1
Channel	0
Parameters of radio	(+) expand
Interfaces	(+) expand
Limitation distances	auto
Map gluing	disabled
Gluing ready	no
Submap X shift, m (-100.00..100.00)	0.00
Submap Y shift, m (-100.00..100.00)	0.00
Submap rotation, degrees (-360..360)	0

Connected: COM4 X: 4.97, Y:-1.18 Rate: 16 Hz 547 total, 0 failed (0%)