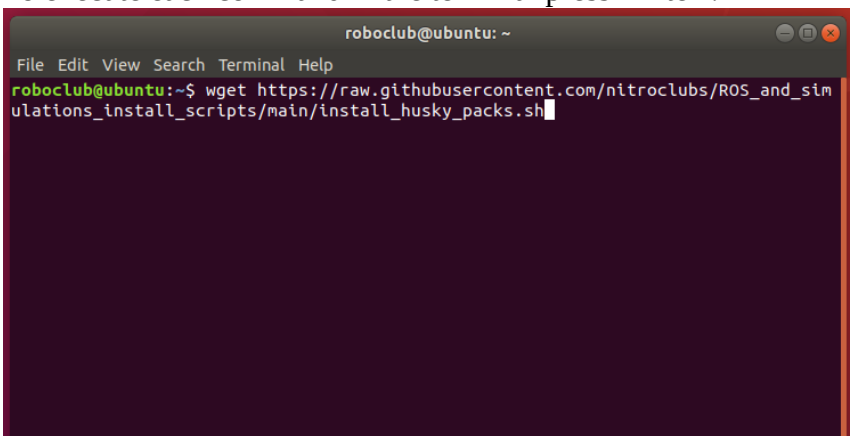
# Install and operate with Husky robot

## Installation

Open new terminal and complete the following commands (copy/paste them in the terminal):

- Download the installation script: `wget https://raw.githubusercontent.com/nitroclubs/ROS_and_simulations_install_scripts/main/install_husky_packs.sh`
- Give permissions to the file: `chmod 775 install_husky_packs.sh`
- Run the script: `bash ./install_husky_packs.sh`
- When the process is complete you will see the massage `<Complete!!!>` at the end of the terminal. Close the terminal!

To execute each command in the terminal press ‘Enter’.

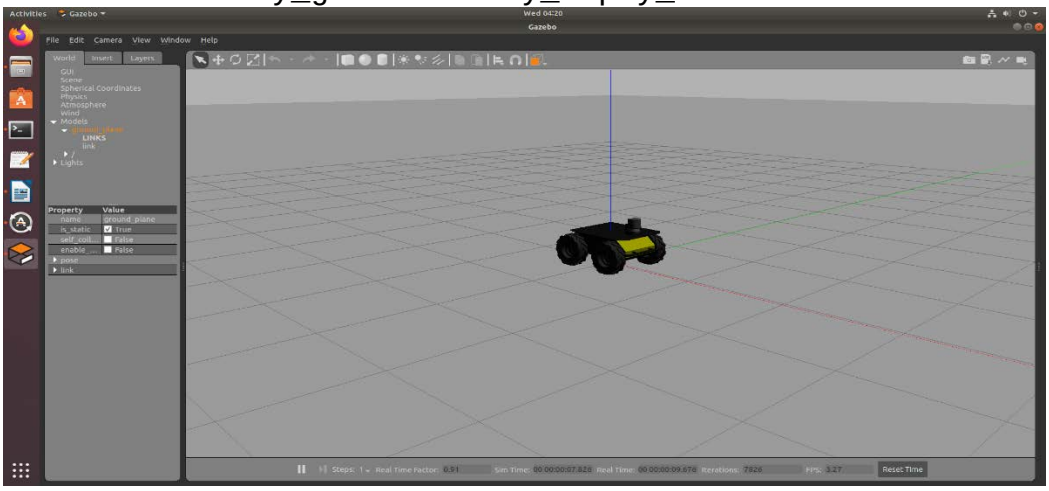



## Starting Husky robot in Gazebo simulation

Open new terminal and enter the command:
Run one of the two provided simulation environments:

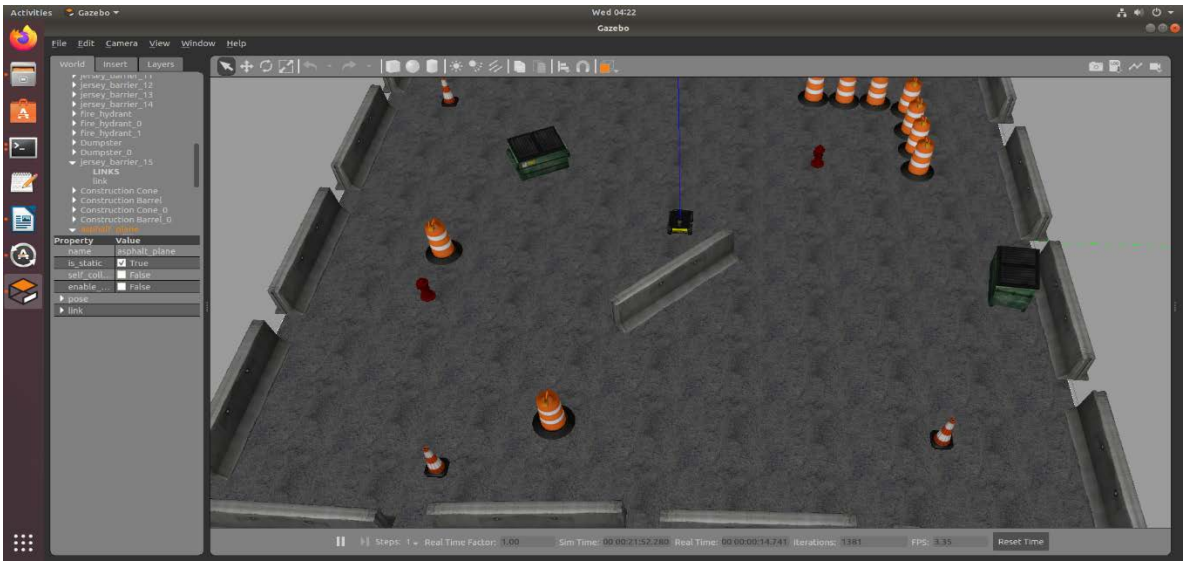
- Simulate Husky in an empty world. You can add new objects to this world using the Gazebo controls.

`roslaunch husky_gazebo husky_empty_world.launch`



- Simulate Husky in a Clearpath designed world. This is the base environment for the navigation tutorials. It will take some time to start, as the simulator will need to download resources from the Gazebo servers.

```
roslaunch husky_gazebo husky_playpen.launch
```

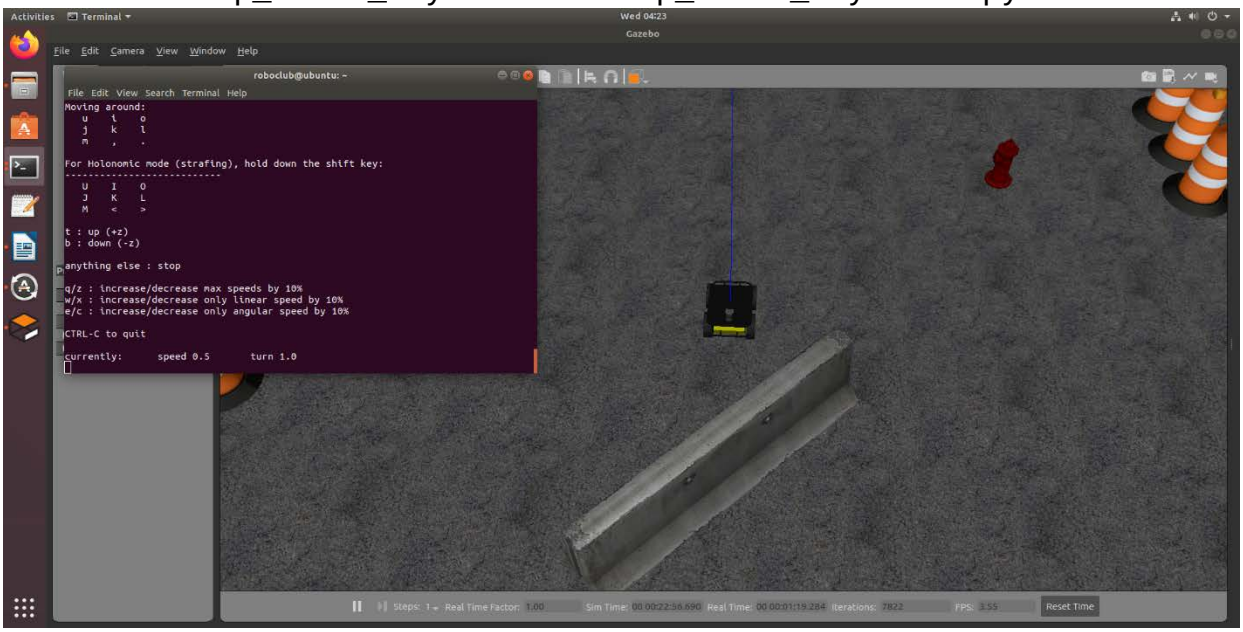



It will take some time to start, as the simulator will need to download resources from the Gazebo servers.

**Move the robot in the simulation**

In order to teleoperate the Husky robot with the keyboard, launch the teleoperation node with below command in a new terminal window. This terminal must be active if you want to send commands.

```
rosrun teleop_twist_keyboard teleop_twist_keyboard.py
```



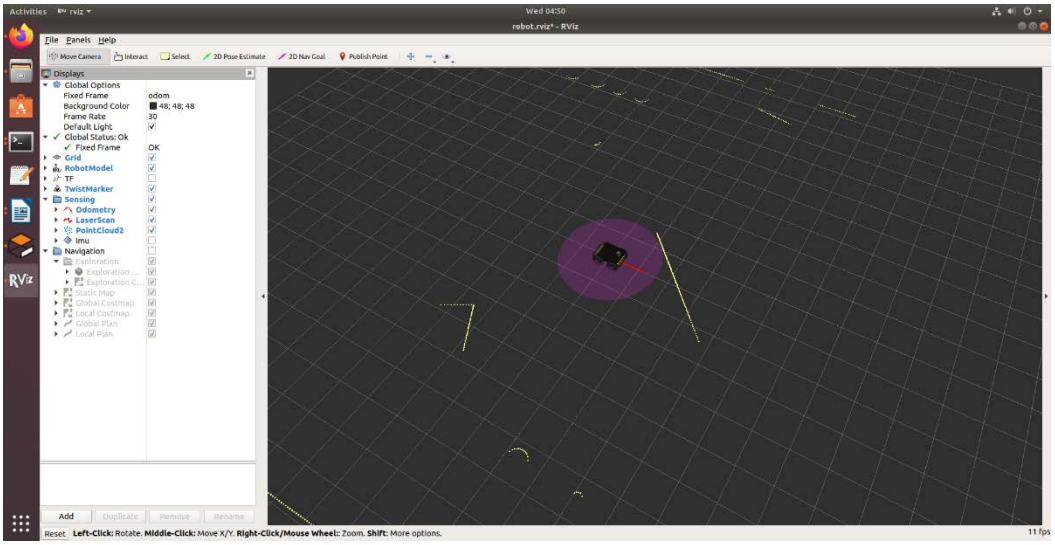

To terminate the teleop node enter the **`Ctrl+C`** in the terminal that runs that node.

Husky Gmapping demo:
http://wiki.ros.org/husky_navigation/Tutorials/Husky%20Gmapping%20Demo
In three separate terminal windows:
Start the Clearpath-configured Husky simulation environment:

```
roslaunch husky_gazebo husky_playpen.launch
```

Start the Clearpath-configured rviz visualizer:

```
roslaunch husky_viz view_robot.launch
```

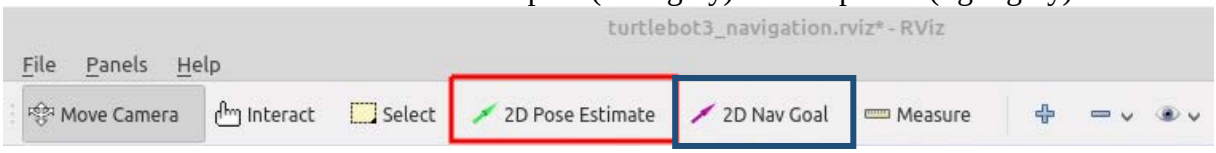


Start the gmapping demo:

```
roslaunch husky_navigation gmapping_demo.launch
```

In the Rviz visualizer, make sure the visualizers in the Navigation group are enabled.

Use the 2D Nav Goal (in the blue rectangle) tool in the top toolbar to select a movement goal in the visualizer. Make sure to select an unoccupied (dark grey) or unexplored (light grey) location.




As the robot moves, you should see the grey static map (map topic) grow. Occasionally, the gmapping algorithm will relocalize the robot, causing a discrete jump in the map->odom transform.

To save the generated map, you can run the map_saver utility in new terminal:

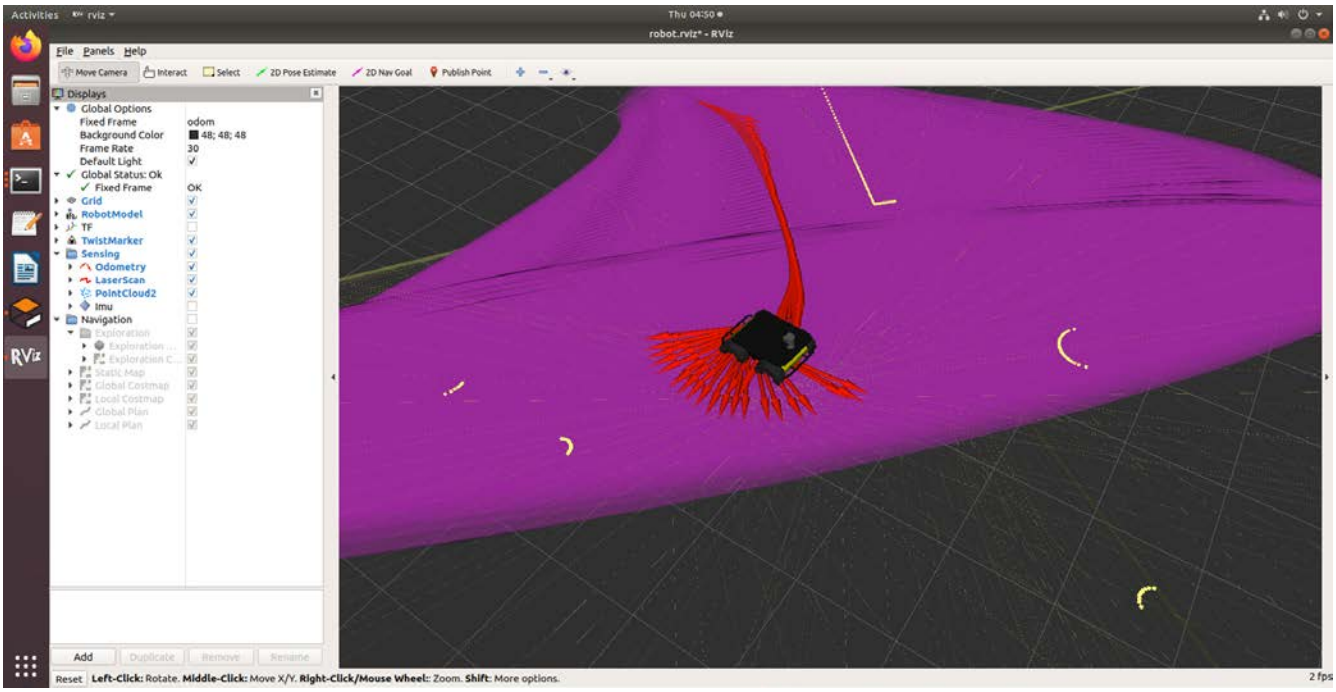
```
rosrun map_server map_saver -f <filename>
```

Husky Frontier Exploration demo
In three separate terminal windows:
Start the Clearpath-configured Husky simulation environment:

```
roslaunch husky_gazebo husky_playpen.launch
```

Start the Clearpath-configured rviz visualizer:
`roslaunch husky_viz view_robot.launch`

Start the frontier_exploration demo:
`roslaunch husky_navigation exploration_demo.launch`




In the Rviz visualizer, make sure the visualizers in the Navigation group are enabled.

Use the Point tool in the top toolbar to draw a closed polygon on the map that the Husky should explore. Watch the terminal window for instructions.

As the robot moves, you should see the grey static map (map topic) grow. Occasionally, the gmapping algorithm will relocalize the robot, causing a discrete jump in the map->odom transform.

When the exploration goal is complete, you will see a feedback message in the terminal window. You can now issue a new exploration goal if you wish.
To save the generated map, you can run the map_saver utility:

`rosrun map_server map_saver -f <filename>`