

# Using ROS/Gazebo Simulator with SITL

Christopher Smith

December 3, 2015

## Intro

**DISCLAIMER:** This installation walkthrough and the software is owed to the hard work and efforts of the Ardupilot group and alexbuyval at Github and Bitbucket(full citation below).

These instructions can be found at <http://dev.ardupilot.com/wiki/using-rosgazebo-simulator-with-sitl>

## LINUX INSTALL: Ubuntu 14.04

**install-ros.bash:** Contains the installs for all the dependencies of the simulation.

**setup-ardu-workspace.bash:** Will create a workspace from the current directory for the simulation.

**IF YOU DO NOT WANT TO USE THE SCRIPTS**, you can copy and paste from the files to execute all or the portion of the script needed. The scripts do make life a bit easier.

1. Download the bash scripts found in this folder named *install-ros.bash* and *setup-ardu-workspace.bash*.

2. Make both files executable:

```
$ chmod +x filename
```

3. Run both scripts sequentially.

```
$ ./install-ros.bash
```

```
$ ./setup-ardu-workspace.bash
```

4. Run the following command.

```
$ export PATH=$PATH:$HOME/ardupilot/Tools/autotest
```

Note: Use your own path to ardupilot folder in the line above!

5. You are done with the setup. You can begin using the simulation(Fig. 1).

**Installation Issues** If there are compilation issues using either `catkin_make` or `catkin_build` then more packages will need to be installed. It will usually be missing a config file for a library that will need to be installed.

## Running the Simulation

1. Source the simulation

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

2. Open a terminal and execute the script:

```
$ cd ~/ardupilot/ArduCopter
```

```
$ sim_vehicle.sh -f arducopter_sitl_ros --console
```

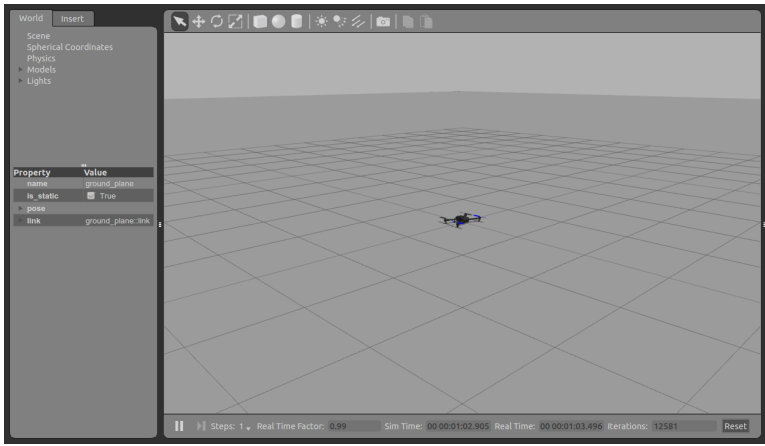


Figure 1: UAV in Gazebo

3. At this point you can control the UAV manually using an XBox(or compatible) controller. We have found that other controllers are capable of controlling the craft, but the controls for steering and motor speed are mapped on the same stick.

As a result of the flight controller being implemented using the ROS framework, the flight controller message traffic can be monitored using ROS tools(Fig. 2), such as using:

```
$ rosrn rqt_graph rqt_graph
```

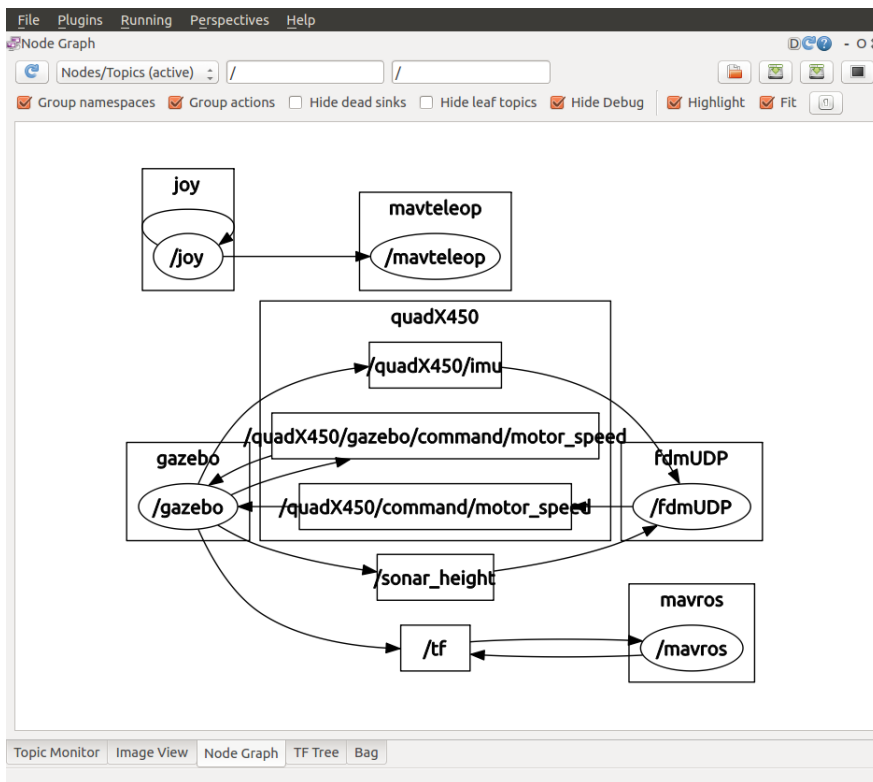


Figure 2: ROS Topic Publishers and Subscribers