

## ArduPilot SITL and Gazebo setup and linking

The task is to install and setup Ardupilot SITL and Gazebo (we will be using Gazebo Harmonic aka Gazebo 8) on Ubuntu 24.

The issue we faced:

We installed a different version of gazebo(gazebo 9 aka gazebo ionic) but the dependencies we installed from the ardupilot documentation was for gazebo 8 (Harmonic). Hence they worked individually but failed to link to each other.

Note: For those who have attempted installation before and faced issues, just do a clean installation again by uninstalling gazebo. The ardupilot cloned directory can be used as it is unless you made some changes. For others trying installation for the first time, just follow the steps below and you are good to go.

We started with following the documentation on the official ardupilot SITL website:

<https://ardupilot.org/dev/docs/building-setup-linux.html#building-setup-linux>

STEPS:

1. Make sure you have git installed, if not, install git first by following these commands.  
sudo apt-get update  
sudo apt-get install git  
sudo apt-get install gitk git-gui
2. Clone the ardupilot github repo using (Make sure you are in the desired folder of your choice before cloning, home directory also works) [Note: run the commands one by one instead of copy-pasting them all at once]  
git clone --recurse-submodules <https://github.com/ArduPilot/ardupilot>  
cd ardupilot  
Tools/environment\_install/install-prereqs-ubuntu.sh -y #(we are just installing some pre-reqs, recommended)  
. ~/.profile
3. Add some directories to the search path(helps later)  
open your home directory and press ctrl+H to show hidden files.  
open .bashrc and add the following 2 lines to the end of the file.  
export PATH=\$PATH:\$HOME/ardupilot/Tools/autotest  
export PATH=/usr/lib/ccache:\$PATH  
Reload using: . ~/.bashrc

These steps are enough for our use

Next install a supported Gazebo version. We are using Gazebo Harmonic.

Perk: latest version of gazebo compatible with ardupilot

The steps were followed from the official gazebo website:

[https://gazebo.org/docs/harmonic/install\\_ubuntu/](https://gazebo.org/docs/harmonic/install_ubuntu/)

It has some basic steps:(just follow the commands)

1. sudo apt-get update
2. sudo apt-get install curl lsb-release gnupg
3. sudo curl https://packages.osrfoundation.org/gazebo.gpg --output /usr/share/keyrings/pkgs-osrf-archive-keyring.gpg

4. `echo "deb [arch=$(dpkg --print-architecture)  
signed-by=/usr/share/keyrings/pkgs-osrf-archive-keyring.gpg]  
https://packages.osrfoundation.org/gazebo/ubuntu-stable $(lsb_release -cs) main" | sudo tee  
/etc/apt/sources.list.d/gazebo-stable.list > /dev/null`
5. `sudo apt-get update`
6. `sudo apt-get install gz-harmonic`
7. That's it, you have gazebo harmonic installed.

The next part is where the problems arise(atleast for us they did). This part is where we link the ardupilot SITL and gazebo using ardupilot\_gazebo repo

Follow these steps:

1. Check if gazebo is working using: `gz sim -v4 -r shapes.sdf`
2. Install additional dependencies for gazebo:
  - A. `sudo apt update`
  - B. `sudo apt install libgz-sim8-dev rapidjson-dev`
  - C. `sudo apt install libopencv-dev libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev  
gstreamer1.0-plugins-bad gstreamer1.0-libav gstreamer1.0-gl`
3. Clone the ardupilot\_gazebo repo:
  - A. `mkdir ws_name/src && cd ws_name/src` #(you can replace ws\_name with your desired workspace name)
  - B. `git clone https://github.com/ArduPilot/ardupilot\_gazebo`
4. Paste the following line in the .bashrc file as earlier:  
`export GZ_VERSION=harmonic`
5. Then run: `source ~/.bashrc`
6. `cd ardupilot_gazebo`
7. `mkdir build && cd build`
8. `cmake .. -DCMAKE_BUILD_TYPE=RelWithDebInfo`
9. `make -j4`
10. Paste these 2 lines in .bashrc:(Make sure to change ws\_name to whatever name you gave to your workspace above)
  - A. `export  
GZ_SIM_SYSTEM_PLUGIN_PATH=$HOME/ws_name/src/ardupilot_gazebo/build:$GZ_SIM_SY  
STEM_PLUGIN_PATH`
  - B. `export  
GZ_SIM_RESOURCE_PATH=$HOME/ws_name/src/ardupilot_gazebo/models:$HOME/ws_name  
/src/ardupilot_gazebo/worlds:$GZ_SIM_RESOURCE_PATH`

This is it for the setup, now it's turn to test them.

1. Run Gazebo using: (can be run in the home directory)  
`gz sim -v4 -r iris_runway.sdf`
2. Run SITL using: `sim_vehicle.py -v ArduCopter -f gazebo-iris --model JSON --map --console`  
Can be run in the home directory because of the paths we added in the .bashrc file
3. Arm and takeoff:  
In the SITL terminal you will see 'STABILIZE>' run the following commands one by one to test basic drone working:
 

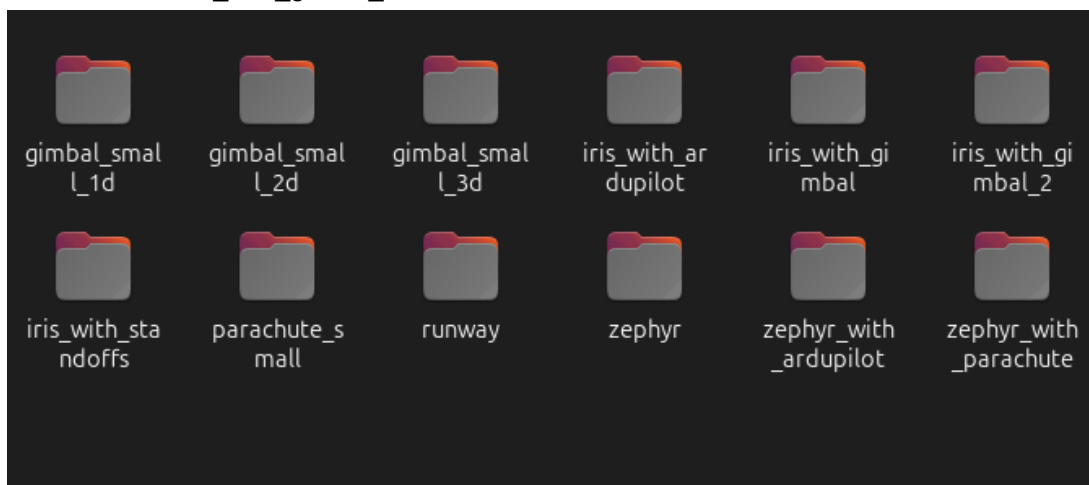
```
STABILIZE> mode guided
GUIDED> arm throttle
GUIDED> takeoff 2
```

4. You should see the drone takeoff in Gazebo, and voila, we are ready for the simulation.
5. You can now play around with the drone, trying to rotate it mid flight(in gazebo), or moving it up and down(in gazebo) or check out other commands on drone control on ardupilot website.
6. Most important of all.  
Make sure to contact Parinay Bhaiya(+917000617506), whether you face any issues or not and get some tips on aura farming.

## Two Drones Simulation:

Now that we are done with simulating one drone successfully, let us simulate two drones on Gazebo together.

1. For this we need to make some changes in the .sdf files in ardupilot\_gazebo folder. Open ardupilot\_gazebo>models. Here you can see the folder iris\_with\_gimbal. Copy the entire folder and rename it 'iris\_with\_gimbal\_2'.



2. Open iris\_with\_gimbal\_2>model.sdf and change the 3rd line to <model name="iris\_with\_gimbal\_2"> and the 191st line to <fdm\_port\_in>9012</fdm\_port\_in>
  3. Now go to ardupilot\_gazebo>worlds. There is a file named 'iris\_runway.sdf'.
  4. Open 'iris\_runway.sdf' and make the following changes. (You can alternately make a new sdf file and paste the iris\_runway.sdf with the following changes. But make sure to run the gazebo using that name)
- A. Replace this part of the code(towards the end of the iris\_runway.sdf)

```
<include>
  <uri>model://runway</uri>
  <pose degrees="true">-29 545 0 0 0 363</pose>
</include>

<include>
  <uri>model://iris_with_gimbal</uri>
  <pose degrees="true">0 0 0.195 0 0 90</pose>
</include>
```

with

```
113     <include>
114     |   <uri>model://runway</uri>
115     |   <pose degrees="true">-29 545 0 0 0 363</pose>
116     </include>
117
118     <include>
119     |   <uri>model://iris_with_gimbal</uri>
120     |   <pose degrees="true">0 0 0.195 0 0 90</pose>
121     </include>
122
123     <include>
124     |   <uri>model://iris_with_gimbal_2</uri>
125     |   <pose degrees="true">2 2 0.195 0 0 90</pose>
126     </include>
```

B. This is the copy-paste ready code for you(only the replacement part)

```
<include>
  <uri>model://runway</uri>
  <pose degrees="true">-29 545 0 0 0 363</pose>
</include>

<include>
  <uri>model://iris_with_gimbal</uri>
  <pose degrees="true">0 0 0.195 0 0 90</pose>
</include>

<include>
  <uri>model://iris_with_gimbal_2</uri>
  <pose degrees="true">2 2 0.195 0 0 90</pose>
</include>
```

C. Save the file and close.

D. Now run gazebo using: `gz sim -v4 -r iris_runway.sdf` (or name of your sdf file if you did make a new file)

E. Run 1st instance of ardupilot sitl using: `sim_vehicle.py -v ArduCopter -f gazebo-iris --model JSON --map --console -l0 --out udp:127.0.0.1:14550`

F. Run another instance using: `sim_vehicle.py -v ArduCopter -f gazebo-iris --model JSON --map --console -l1 --out udp:127.0.0.1:14551`

G. Run each command in a different terminal.

H. And.. You are ready to go! You should now be able to see gazebo with two drones and two terminals with MAV controls. Now you can control the drones independently using the commands just like you did for one drone.

You can contact the Rushikesh bhaiya(9351571049) for any doubts or issues.