**Installation:**

Paho, Vpython and backend packages if needed

//words in boldface are the procedures to run the simulation//

**Subscribe:**

1. Think of how many robots you want to show in the canvas
2. Look at “subscribe.py”, create robots in function on\_message and in main like this:

In main:

drone1 = Drone()

drone1.f.pos = (0,0,0) #initialize a drone

car1 = create\_robot()

car1.pos = (0,0,0) #initializa a car

In on\_message create the category by topics and move the positions:

elif msg.topic=="robot1":

vectors = str(msg.payload).split()

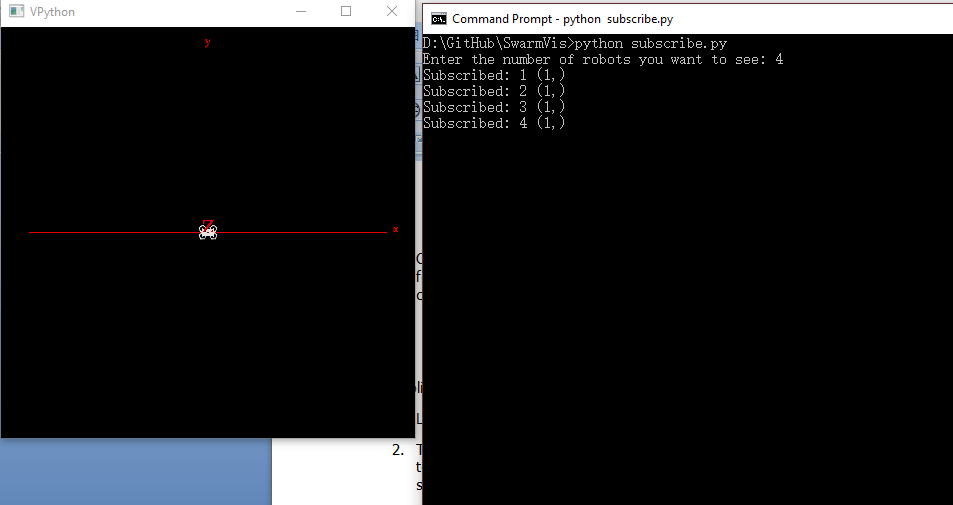
drone1.f.pos = (float(vectors[0]),float(vectors[1]),float(vectors[2]))

elif msg.topic=="robot2":

vectors = str(msg.payload).split()

car1.pos = (float(vectors[0]),float(vectors[1]),float(vectors[2]))

1. **Open cmd, and run “subscribe.py”, enter the number of robots you want to see; for this simulation, enter 4 (you can add more by last step, make sure you have correspondent publish data), the output should look like this:**



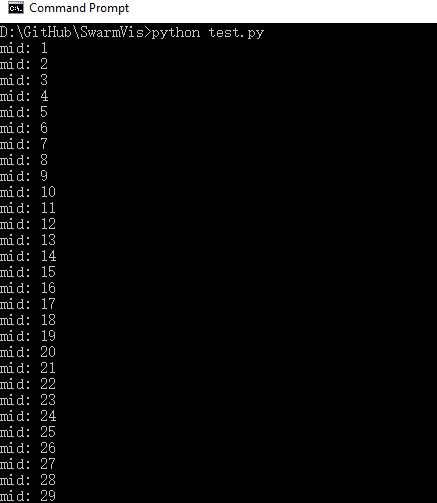
**Don’t close this!**

**Publish:**

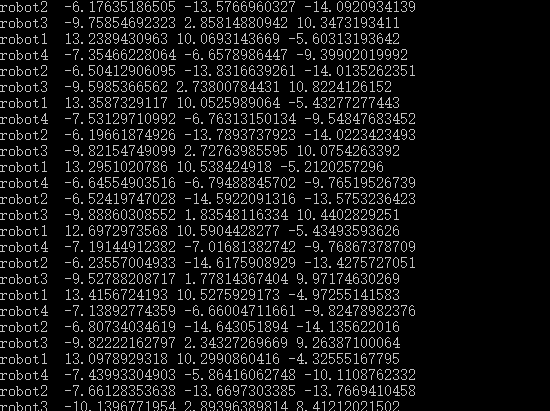
1. Look into “publish.py” and choose the method of publishing you want to use
2. Then edit the “test.py” to utilize the methods in “publish.py”; each robot publish to an unique topic(robot1, robot2, ….); we use threads to let them publish at the same time
3. For the simulation, we are using rbtpublishseperate function; for later real experiments, probably simplepublish function will be used
4. **Open a new cmd other than the one for subscribe and draw; Run “test.py” in cmd, note that if the Internet is not stable or the server of neptune.usc.edu crashed, it may not publish; when encountered this problem, try again with a new cmd later or change ports in “subscribe.py” and “publish.py” into local ports to simulate**
5. If you want to increase the number of robots, create a new thread in “test.py” like this : thread.start\_new\_thread( rbtpublishseperate, ("car1", "robot2", ) )

Car1 is the name of thread, robot2 is the topic of publish, each robot should have its own threadName and topic

1. When there are four robots, the output should look like this for publish cmd:



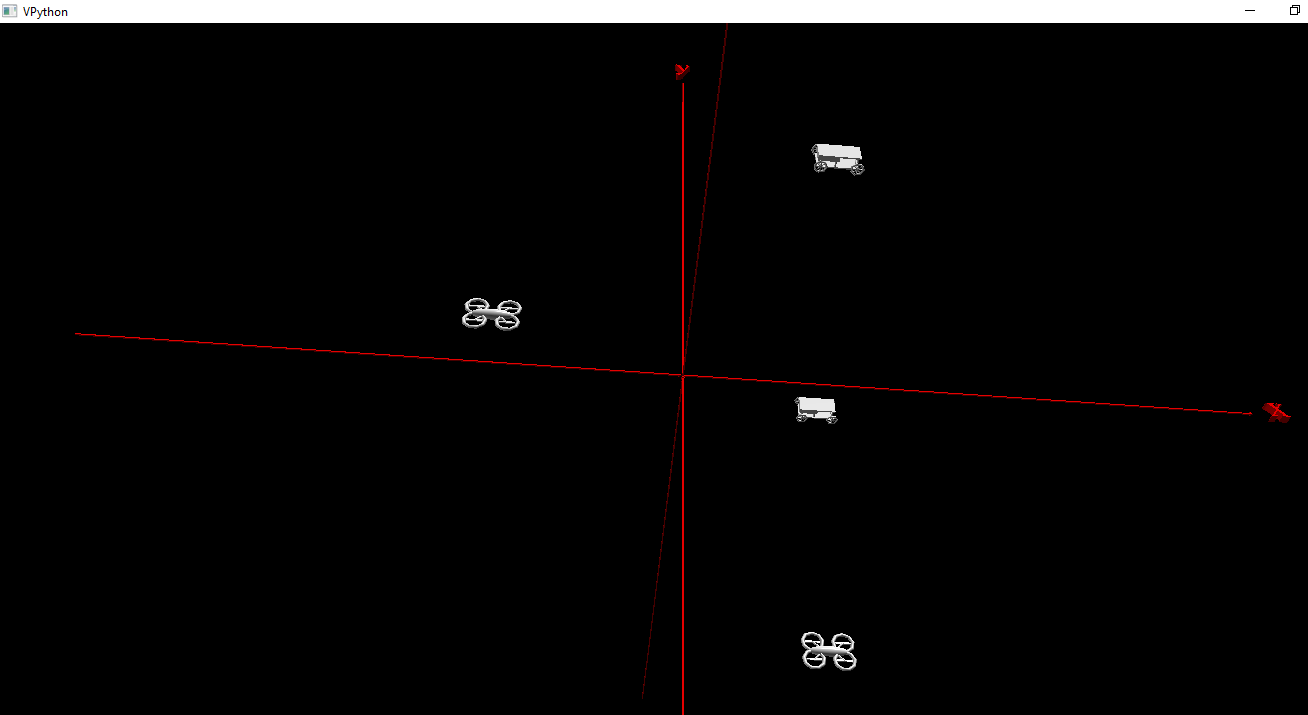
The output should look like this for subscribe cmd:



And now the drones and cars should be moving in the canvas now. to change the rate of the movements, change the sleep time in “publish.py”

Canvas:

1. Now the robots are moving, you can move the visual angles and can zoom in or out: Right button drag or Ctrl-drag to rotate "camera" to view scene. Middle button or Alt-drag to drag up or down to zoom in or out. On a two-button mouse, middle is left + right
2. To change the look of canvas or the models of robots, look into “construct.py”, in which has a drone class to create drone, a create\_robot function to create car, a create\_coordinate function to create xyz axis
3. The simulation should look like this:



1. You can add trails by set trail = True in “construct.py”, and you can add texts showing the current xyz location of each robot; but that may cause the simulation to be slower