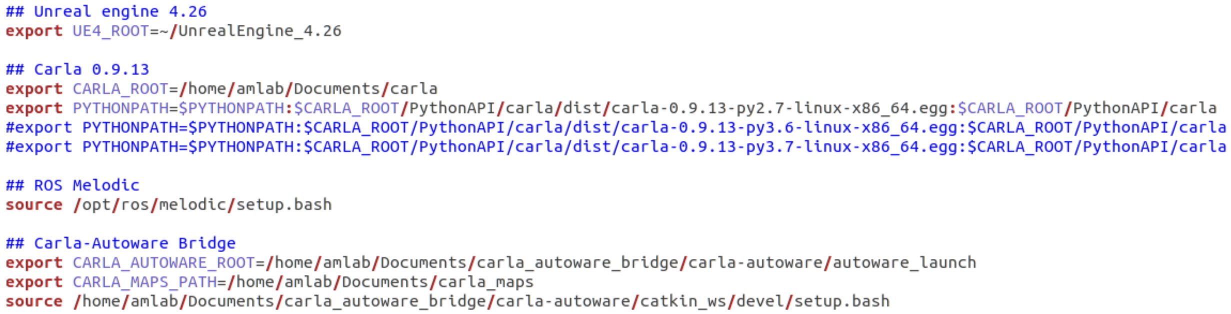
Quick Start Guide for the Carla-Autoware Bridge

This documentation provides step-by-step instructions for the configuration and execution of the Carla simulation on the specified workstation using Autoware.ai.

1. Environment Details:

- The setup utilizes ROS (Robot Operating System) Melodic version.
- Unreal Engine 4.26 is used, and the repository is in the home folder.
- Carla 0.9.13 is used, with the root repository residing in the *~/Documents* directory.
- Autoware.ai 1.13 is employed, and its root repository is in *~/Documents/carla_autoware_bridge/autoware.ai.openplanner*.
- The carla_ros_bridge, carla_autoware_bridge and the ROS launch files are in *~/Documents/carla_autoware_bridge/carla-autoware.*
- All point cloud files for Carla towns are stored in *~/Documents/carla_maps*.

The modified ***"~/.bashrc***" file should contain the subsequent environment variables and sourced files:



```
## Unreal engine 4.26
export UE4_ROOT=~/UnrealEngine_4.26

## Carla 0.9.13
export CARLA_ROOT=/home/amlab/Documents/carla
export PYTHONPATH=$PYTHONPATH:$CARLA_ROOT/PythonAPI/carla/dist/carla-0.9.13-py2.7-linux-x86_64.egg:$CARLA_ROOT/PythonAPI/carla
#export PYTHONPATH=$PYTHONPATH:$CARLA_ROOT/PythonAPI/carla/dist/carla-0.9.13-py3.6-linux-x86_64.egg:$CARLA_ROOT/PythonAPI/carla
#export PYTHONPATH=$PYTHONPATH:$CARLA_ROOT/PythonAPI/carla/dist/carla-0.9.13-py3.7-linux-x86_64.egg:$CARLA_ROOT/PythonAPI/carla

## ROS Melodic
source /opt/ros/melodic/setup.bash

## Carla-Autoware Bridge
export CARLA_AUTOWARE_ROOT=/home/amlab/Documents/carla_autoware_bridge/carla-autoware/autoware_launch
export CARLA_MAPS_PATH=/home/amlab/Documents/carla_maps
source /home/amlab/Documents/carla_autoware_bridge/carla-autoware/catkin_ws/devel/setup.bash
```

2. Executing the Configuration:

Commence the setup process by following these steps:

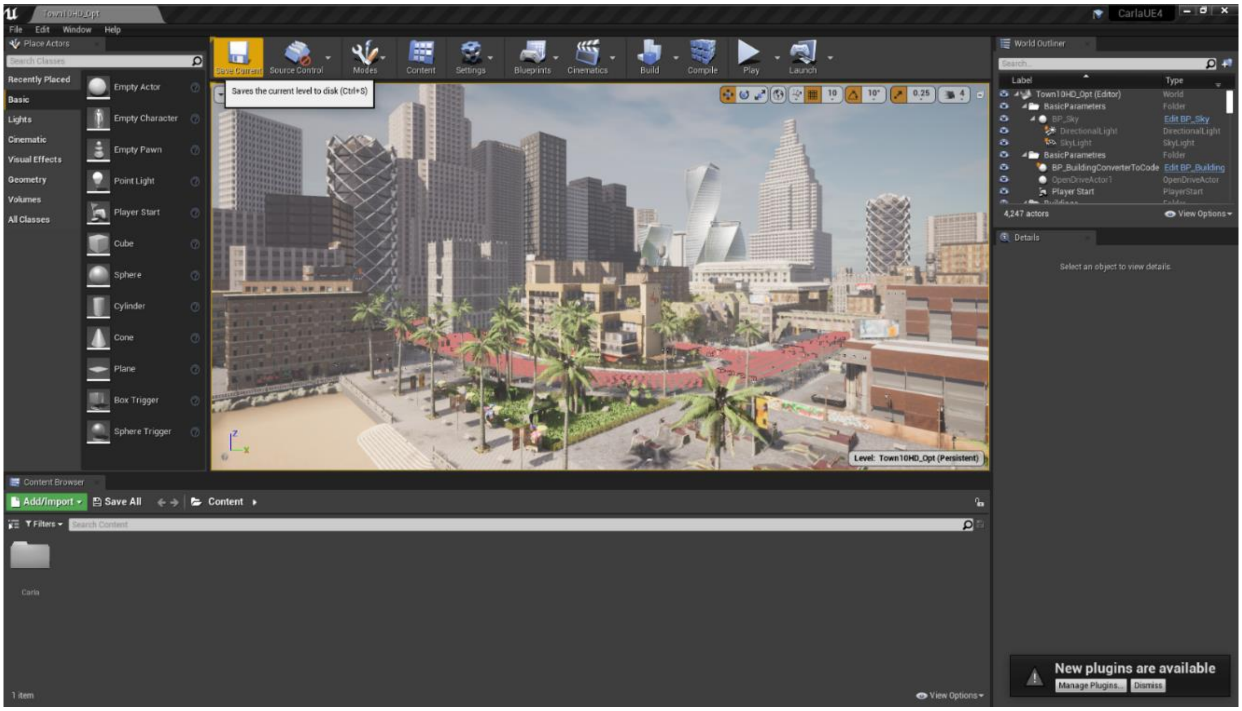
1. To initiate the Carla simulator, launch a new terminal and input the following command:

```
$ cd ~/Documents/carla

$ make launch
```

This command will initiate the Carla simulator, leveraging the Unreal Engine. To terminate the Carla simulator, close this terminal.

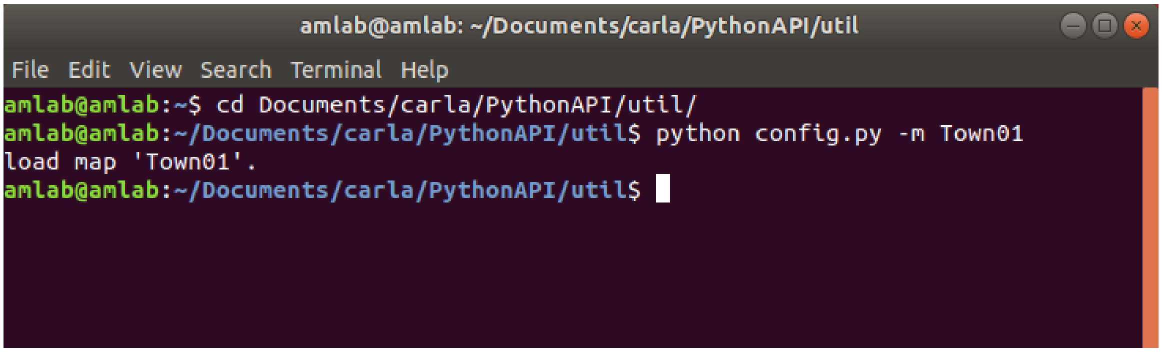
2. After the simulator has completely launched, an interface will be displayed. To run the simulator, click the play button within the Unreal Engine interface.



3. Note that the launcher always starts with Town 10 from the simulator. To switch to Town 01, use the subsequent command:

```
$ cd ~/Documents/carla/PythonAPI/utils/

$ python config.py –map Town01
```



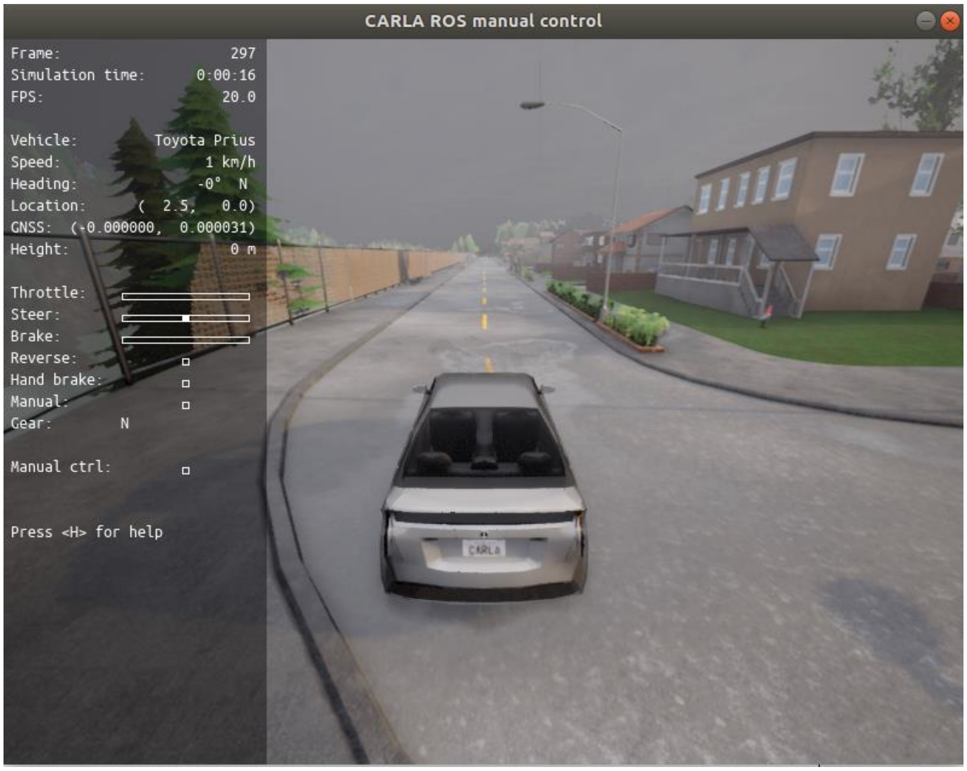
```
amlab@amlab:~$ cd Documents/carla/PythonAPI/util/
amlab@amlab:~/Documents/carla/PythonAPI/util$ python config.py -m Town01
load map 'Town01'.
amlab@amlab:~/Documents/carla/PythonAPI/util$
```

4. Next, launch the carla-autoware bridge. For that, open a new terminal and enter the following command:

```
$ cd ~/Documents/carla_autoware_bridge/carla-autoware/autoware_launch/

$ roslaunch devel.launch
```

This will load the following PyGame window:



```
Frame:                     297
Simulation time:       0:00:16
FPS:                      20.0

Vehicle:           Toyota Prius
Speed:                  1 km/h
Heading:                -0°  N
Location:         (  2.5,  0.0)
GNSS:  (-0.000000,  0.000031)
Height:                    0 m

Throttle:
Steer:
Brake:
Reverse:             □
Hand brake:          □
Manual:              □
Gear:         N

Manual ctrl:         □


Press <H> for help
```

Upon execution, this command will initialize a bridge connecting ROS and Carla, as well as another bridge linking Carla with Autoware through ROS. Use the ***“rostopic list”*** command to access the available ROS topics.

(Note: The launch file can be executed directly without the need to modify the Carla map to Town01. The launch file is designed to handle the map change itself. However, on rare occasions, an error might occur and lead to termination after altering the map. If this occurs, simply rerun the launch file to resolve the issue.)

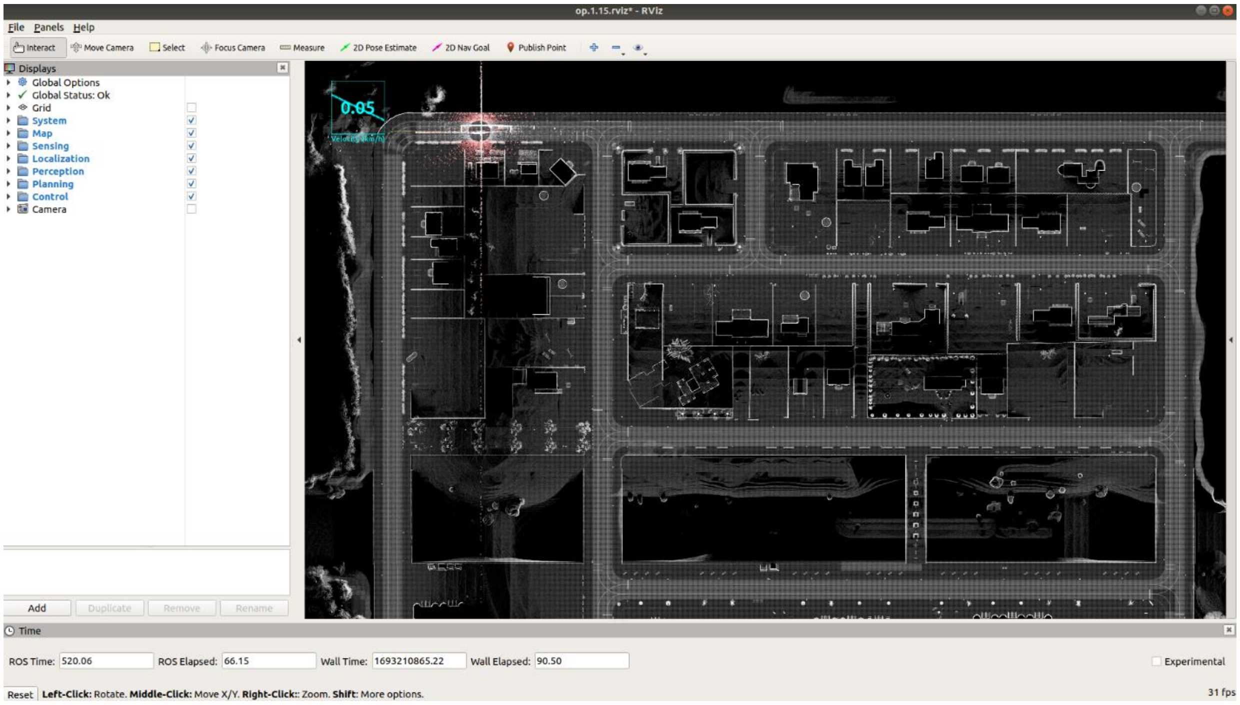
Important: Press **B** to turn off the control commands from Autoware. It will override the control, and you can drive it manually using arrow keys or WASD keys. Press **B** again for Autoware to take control. To check Carla’s autopilot, known as “Traffic Manager”, press **B** first, then press **P**.

5. Run the following command to open the Rviz:

```
$ cd ~/Documents/carla_autoware_bridge/autoware.ai.openplanner/

$ rosrun rviz rviz -d op.1.15.rviz
```

This command will launch Rviz, displaying the car's localization within the town map.

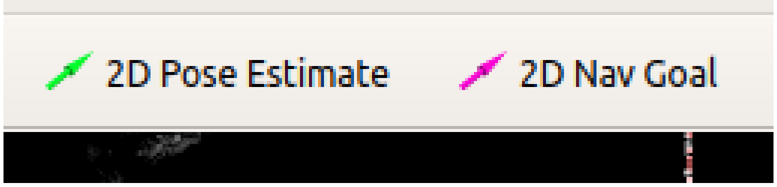


- Global Options
- Global Status: Ok
- Grid □
- System ☑
- Map ☑
- Sensing ☑
- Localization ☑
- Perception ☑
- Planning ☑
- Control ☑
- Camera □

ROS Time: 520.06 ROS Elapsed: 66.15 Wall Time: 1693210865.22 Wall Elapsed: 90.50 □ Experimental

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click:**: Zoom. **Shift**: More options.

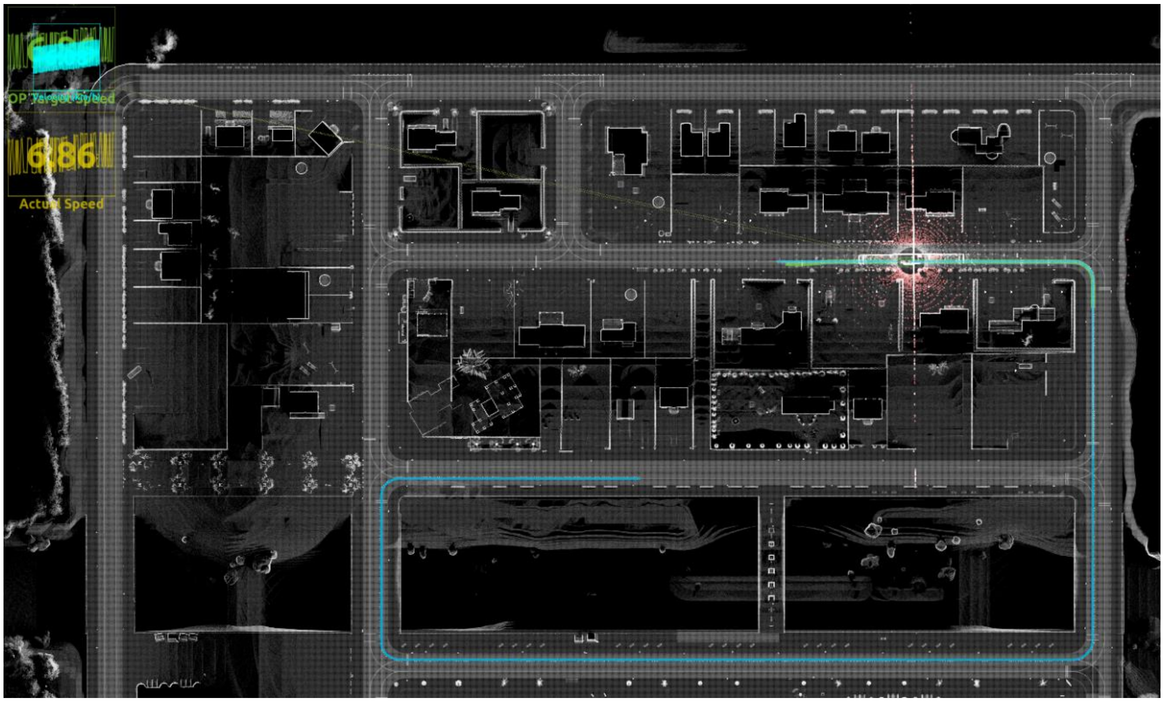
While using Rviz, you'll encounter the following options:



2D Pose Estimate: This option permits you to alter the car's spawn point. By selecting this option and clicking on a location within the road area of the map, the car will respawn at that point.

2D Nav Goal: Use this option to designate a goal point. By selecting this option and clicking on a spot within the road area generates a path from the current location to the chosen goal, prompting the car to navigate until it reaches the goal point.

Upon selecting a goal pose, Rviz will illustrate the generated path, as depicted in the next image.



Note: If you decide to close the Rviz window, a prompt will appear to save any changes. Opt to close without saving unless you've made alterations to the Rviz configuration and wish to retain them.