PolyVerification Suite setup and Configuration

This document contains the information of Setup and Installation process for PolyVerification suite in Linux/Unix. We have modified some of the open source packages which are added in repository.

Below are list of component and dependencies need to be installed for before start the PolyVerif suite.

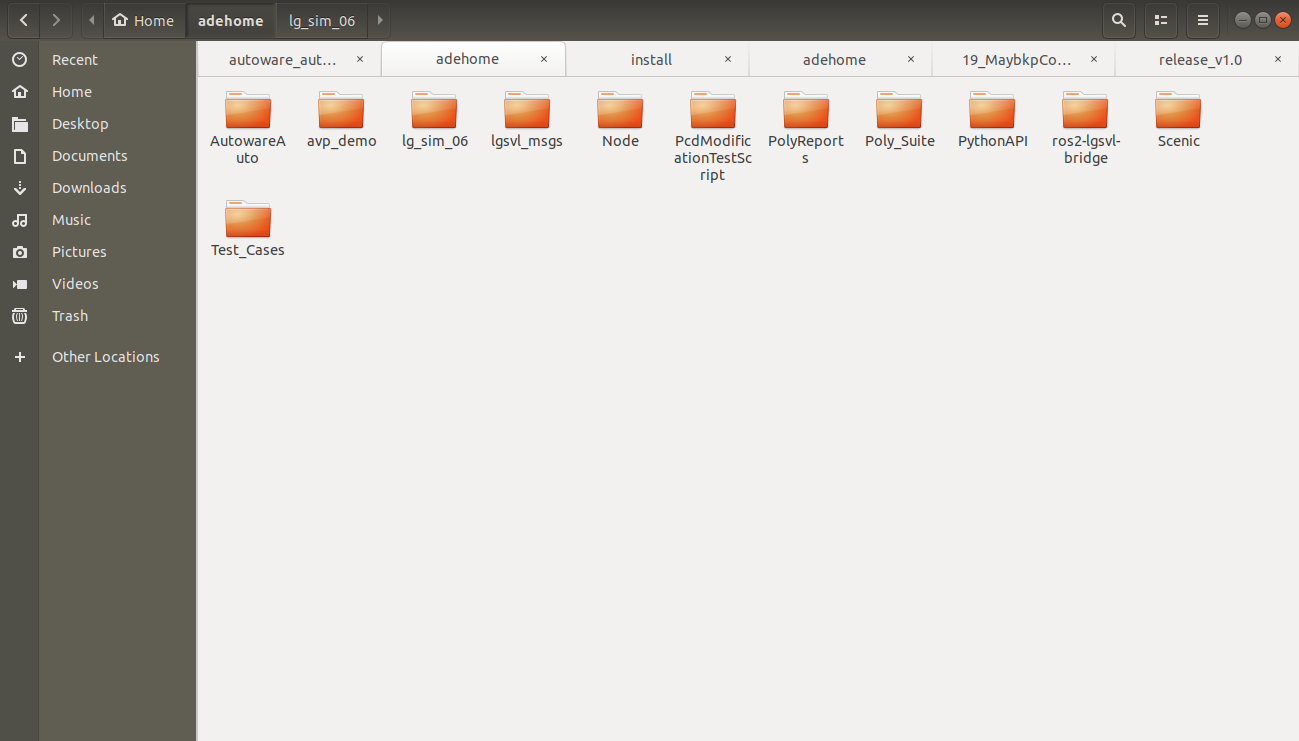
* AutowareAuto
* Lgsvl\_msgs
* Ros2-lgsvl-bridge
* PolyVerif repository
  + PythonAPI
  + Scenic SDL
  + Lgsvl Simulator 2020.06
  + Avp\_demo
  + Perception Validation Node
  + Control Validation Node
  + Test\_Cases
  + PolyVerif Suite

Above package and module need to be install and configured in the same sequence

And directory structure should look like this –

* **adehome**
  + **AutowareAuto**
  + **lgsvl\_msgs**
  + **ros2-lgsvl-bridge**
  + **lg\_sim\_06**
    - **lgsvlsimulator-linux64-2020.06**
    - **m\_lgsvlsimulator-linux64-2020.06**
  + **PythonAPI**
  + **Scenic**
  + **Avp\_demo**
  + **Node**
    - **Node\_Perception\_Validation**
    - **Node\_Control\_Validation**
    - **Node\_Localization\_Validation**
    - **Node\_Path\_Planner\_Validation**
  + **Test\_Cases**
  + **PolyReports**
  + **Poly\_Suite**

Below is the screenshot of the mention structure –



## Dependencies install in Ubuntu-

**Python3.8 –**

**$ sudo apt update**

**$ sudo apt-get install python3-gi**

**Watchdog –**

**$ sudo apt update**

**$ pip install watchdog**

**Pandas –**

**$ sudo apt update**

**$ pip install pandas**

## Setup Docker and Install AutowareAuto

The requirement for ADE is to install docker, please follow below steps –

**Installation*:***

**$ cd ${HOME}**

**$ mkdir adehome**

**$ cd adehome**

**$ wget** <https://gitlab.com/ApexAI/ade-cli/uploads/85a5af81339fe55555ee412f9a3a734b/ade+x86_64>

**$ mv ade+x86\_64 ade**

**$ chmod +x ade**

**$ mv ade ~/.local/bin and /usr/local/bin**

**$ which ade**

**$ ade update-cli**

**$ touch .adehome**

**$ git clone** <https://gitlab.com/autowarefoundation/autoware.auto/AutowareAuto.git>

**$ cd AutowareAuto/**

**$ ade start --update --enter # It will update and start the ade**

Now if you should see the following in your prompt then it successfully installs the autoware:

**Testing:**

**<your\_username>@ade:~$**

Check the distro of autoware auto by running the below command-

**@ade:~$ ls /opt**

**Expected Output: -**

**AutowareAuto # image: binary-foxy:master**

**lgsvl # image: ade-lgsvl/foxy:2020.06**

**ros # image: ade-foxy:master**

If you face any issue during the process please go through the below link –https://autowarefoundation.gitlab.io/autoware.auto/AutowareAuto/installation-ade.html

## Install and Setup lgsvl\_msgs package

**Install ROS2 LGSVL Messages:** lgsvl\_msgs is a ROS / ROS2 hybrid package that provides AD stack agnostic message definitions for interfacing with the LGSVL Simulator.

Once the AutowareAuto install successfully, please follow the below steps to install the lgsvl\_msgs package

**Installation*:***

**$ cd AutowareAuto/**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ sudo apt update**

**$ sudo apt install ros-$ROS\_DISTRO-lgsvl\_msgs**

**$ git clone https://github.com/lgsvl/lgsvl\_msgs.git**

**$ cd lgsvl\_msgs**

**$ colcon build**

**$ source install/setup.bash**

Please check below link If you face any issue while installing the lgsvl\_msgs package-

https://www.svlsimulator.com/docs/archive/2020.06/lgsvl-msgs/

## Ros2-lgsvl-bridge

The SVL Simulator can publish and subscribe to ROS 2 messages by connecting to the [ROS2 LGSVL Bridge](https://github.com/lgsvl/ros2-lgsvl-bridge).

It requires some package to build, follow the below steps to setup-

**Installation*:***

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ sudo apt update**

**$ sudo apt install python3-colcon-common-extensions**

**$ sudo apt install libboost-all-dev**

**$ sudo apt update**

**$ sudo apt install ros-$ROS\_DISTRO-lgsvl-bridge**

**$ git clone** <https://github.com/lgsvl/ros2-lgsvl-bridge.git>

**$ cd ros2-lgsvl-bridge**

**$ git checkout ${ROS\_DISTRO}-devel**

**$ colcon build --cmake-args '-DCMAKE\_BUILD\_TYPE=Release'**

**Testing:**

**$ source install/setup.bash**

**$ lgsvl\_bridge**

Please check the below link if you face any issue-

https://www.svlsimulator.com/docs/system-under-test/ros2-bridge/

## Install dependencies in ade docker-

Python3.8 –

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ sudo apt update**

**$ sudo apt-get install python3-gi**

Watchdog –

**$ sudo apt update**

**$ pip install watchdog**

Pandas –

**$ sudo apt update**

**$ pip install pandas**

## Download/Clone PolyVerif package from the repository

Below is the list of all the packages of repository required-

* **Lgsvl simulator**
* **PythonAPI**
* **Scenic SDL library**
* **Node**
  + **Perception Validation Node**
  + **Control Validation Node**
* **PolyVerif Suite Binary**
* **PolyReports**
* **Test\_Cases**
* **Documentation**

You can download or cloned the updated code (release-v1.1) from the git repository and placed in the adehome path as mention above.

Below is the link -

**$ git clone https://github.com/MaheshM99/PolyVerif.git**

## LGSVL Simulator

Please follow the **Lgsvl\_Simulator\_Setup.docx** document for install and setup the lgsvl simulator.

There are two version of lgsvl simulator-

* Lgsvl simulator from the svl website (follow the document to install and replace in **adehome/lg\_sim\_06** directory)
* Modified lgsvl simulator (added in the repository)

## PythonAPI

Open terminal and go to the Python API folder and enter the below command to install the Python files and necessary dependencies. This is a modified python api’s for use.

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd PythonAPI/**

**$ pip3 install --user -e .**

Run the following example to see the API in action:

**$ python3 ./quickstart/<any\_test\_script>.py**

**Note-** Need to start lgsvl simulator in API\_Only mode.

## Scenic SDL library

Scenic is a domain-specific probabilistic programming language for modelling the environments in simulation for autonomous cars. This is a modified scenic package with AutowareAuto support.

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd Scenic/**

**$ pip3 install --user -e .**

Please follow the below link if you face any issue-

[**https://scenic-lang.readthedocs.io/en/latest/**](https://scenic-lang.readthedocs.io/en/latest/)

## Perception/Detection Validation package

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd Node/Node\_perception\_validation\_ws**

**$ colcon build**

**$ source install/setup.bash**

**Testing:** Run the below command to the package is running or not

**$ ros2 run perception\_validation perception\_subscriber**

## Control Validation package

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd Node/Node\_control\_validation\_ws**

**$ colcon build**

**$ source install/setup.bash**

**Testing:** Run the below command to the package is running or not

**$ ros2 run control\_validation control\_subscriber**

## Localization Validation package

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd Node/Node\_localization\_validation\_ws**

**$ colcon build**

**$ source install/setup.bash**

**Testing:** Run the below command to the package is running or not

**$ ros2 run localizatation\_validation localizatation\_node**

## Mission/Path Planning Validation package

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd Node/Node\_path\_planner\_validation\_ws**

**$ colcon build**

**$ source install/setup.bash**

**Testing:** Run the below command to the package is running or not

**$ ros2 run path\_planner\_validation path\_planner\_node**

## Avp\_Demo

**$ cd adehome**

**$ cd AutowareAuto**

**$ ade start**

**$ ade enter**

**$ source /opt/AutowareAuto/setup.bash**

**$ cd avp\_demo/**

**$ colcon build**

**$ source install/setup.bash**

## PolyVerif Installation and Use

Please follow the **PolyVerification\_Suite\_UserGuide.docx** document for install and setup.