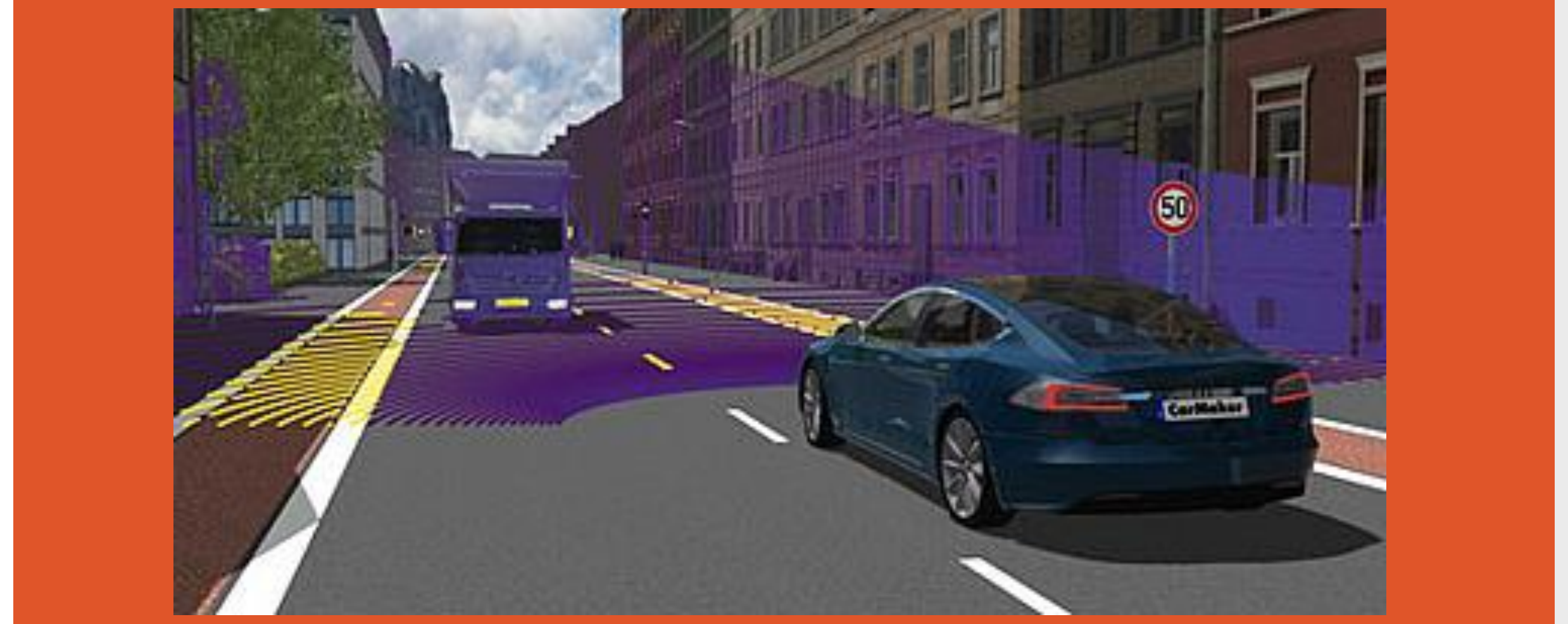
COLLEGE OF ENGINEERING

Electrical Engineering and Computer Science

SIMULATION AND TOOLS

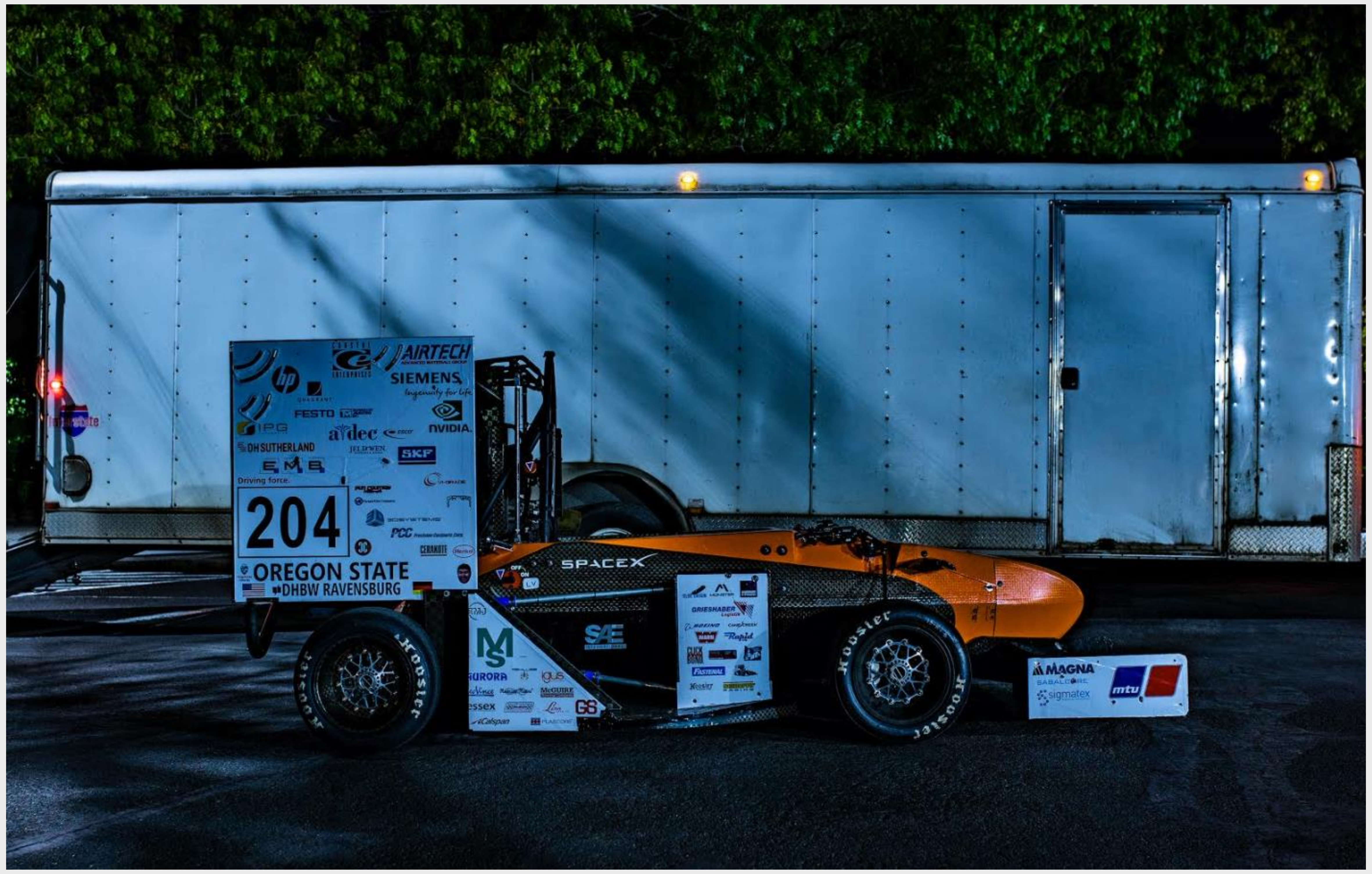
Team Mission: To provide a full simulation and test environment for the autonomous race car to provide necessary testing of the car for competitions.



SIMULATOR

- Our simulator needed to have several things in order to be useful: accurate physics, lidar, camera, and the ability to run tests within the system.
- The simulator we chose is called CarMaker. It is a simulator developed by IPG, and is an industry standard in the automotive industry. We chose this because it has the ability to be heavily edited, basically creating a whole new simulator once we were finished editing it. We had to create a new vehicle model, maps of racetracks with cones, and we had to change the motion of the car to detect cones and run the calculations that we need.
- In order to properly test the environment models, we need to have the ability to benchmark data that has been outputted by the simulator. So we created a way to allow CarMaker to accept test parameters such as: tracks, number of cones, and also changes to any sensor.
- The simulator is set up to have the sim output a bag that stores all the data that was transferred through the entire time of the simulation. These bags can then be visualized to make sure everything is running smoothly, and to also check single points of data to understand the final output of the tests.
- This simulator has the goal of providing an easy way to configure and test the system that is being prepared for the car itself and allow us to test parameters to make sure that the car that is being built accurately.
- This simulator hooks directly up to our testing platform, which allows it to automatically build and test anything that was made that directly works with the simulator.

Oregon State University



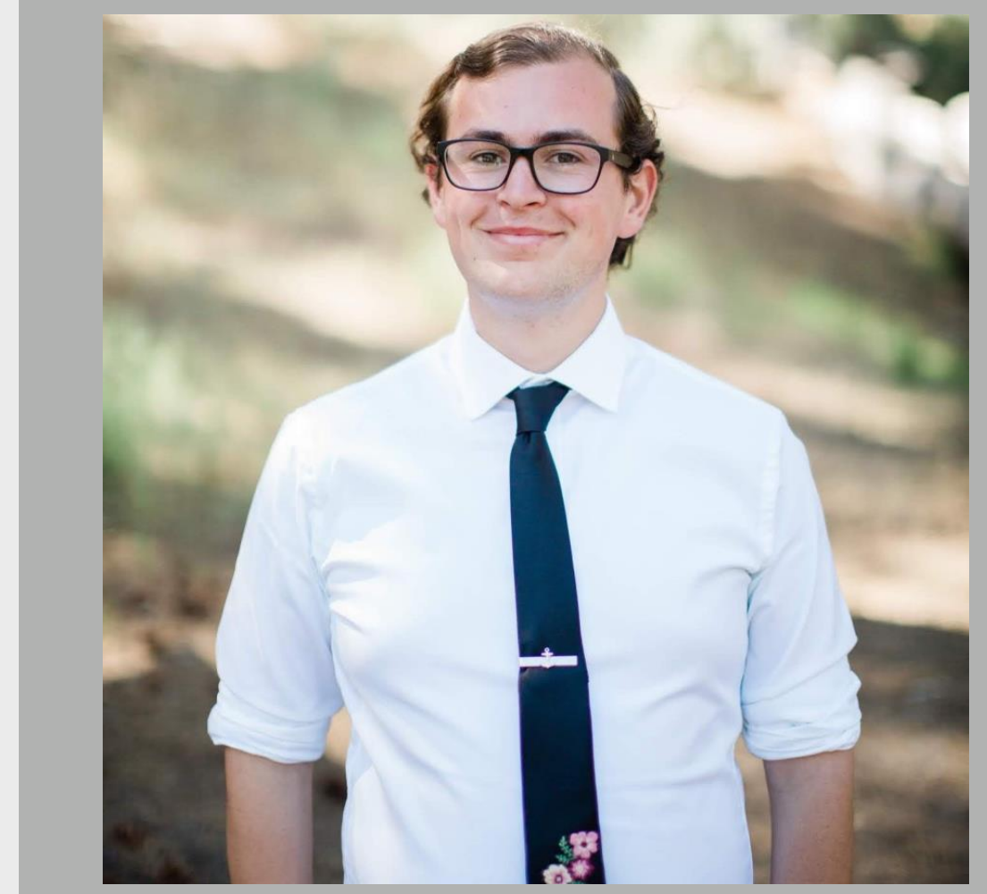
TESTING

- The output of the simulator will provide sufficient data to developers to either change how the autonomous system is associated with the environment.
- Each simulator output has built-in nodes that acts as a middle client to store and reinject desired or most efficient output as a benchmark. Noise will then be added to the output for a realistic measurement of the real environment.
- The simulator will provide the ground truth of all the reading required for the built-in autonomous system to read track lines, which used cones as its boundaries and the car position on the track.
- The ground truth will be the benchmark for the autonomous system accuracy and added noise will be an added precaution for the simulator if the system fails to recognize the real position of a cone.

DEVOPS

- Validation through testing has been one of our teams guiding principles.
- In order to better validate and develop the cars automated systems we need a stable code base and the ability to run as many tests as possible.
- Achieving this goal requires a continuous integration server to manage builds and code validation. Additionally, due to the complex environment required to run the simulator a virtualization agent is needed to consistently reproduce the runtime conditions.
- For our continuous integration tool we chose to use Jenkins, an open source automation server.
- Jenkins allowed us to interface with our existing code base and create a variety of environments.
- To produce sufficient test data we needed to virtualize the simulators runtime environment. Allowing us to automate and distribute the generation of test data.

The Team



GLOBAL FORMULA RACING
GFR
DHBW RAVENSBURG / OREGON STATE UNIVERSITY

Ryan Cryar – cryarr@oregonstate.edu

Arin Reinsch - reinscha@oregonstate.edu

Enrico Sundjojo - sundjoje@oregonstate.edu