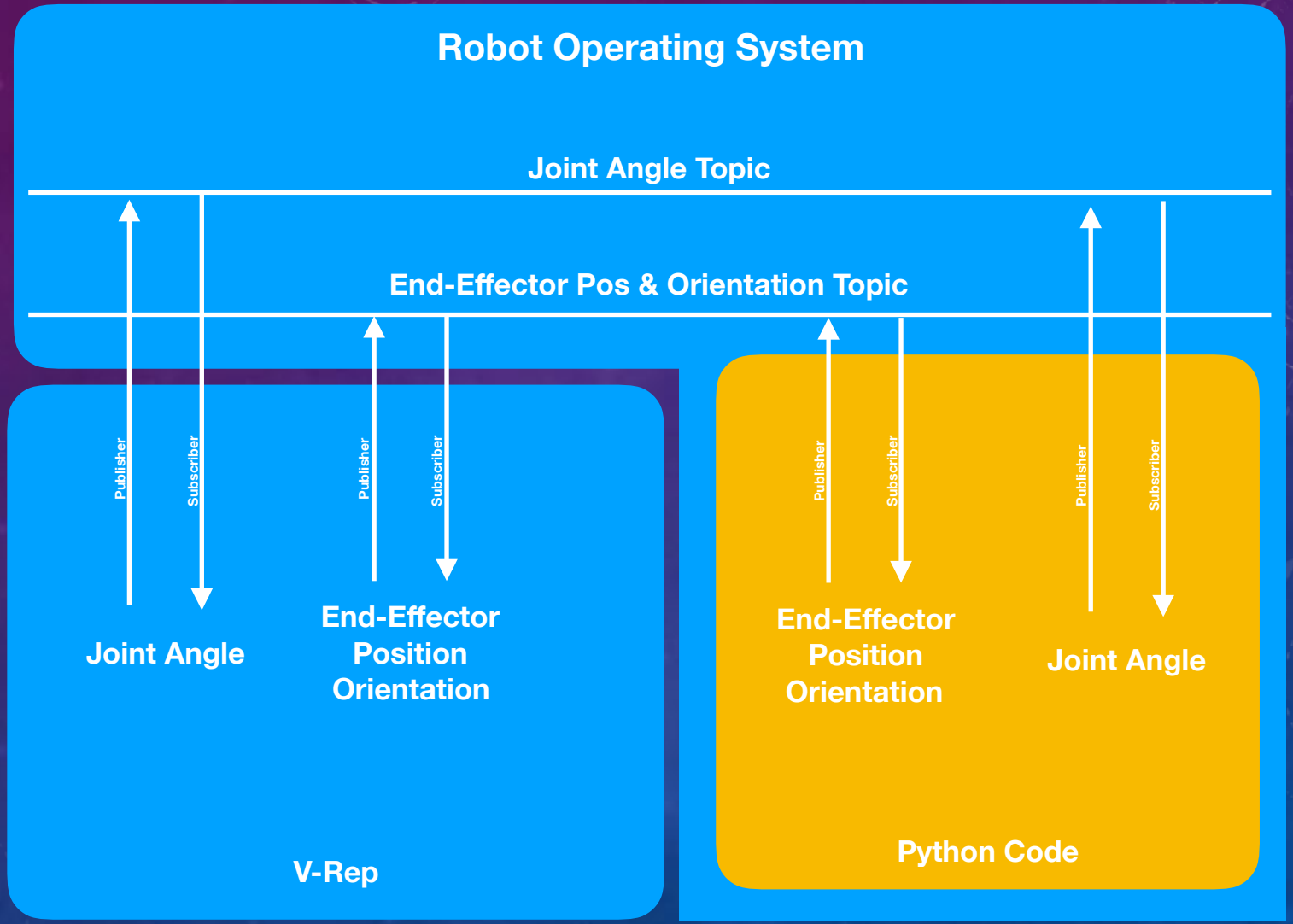
PROCESS REPORT

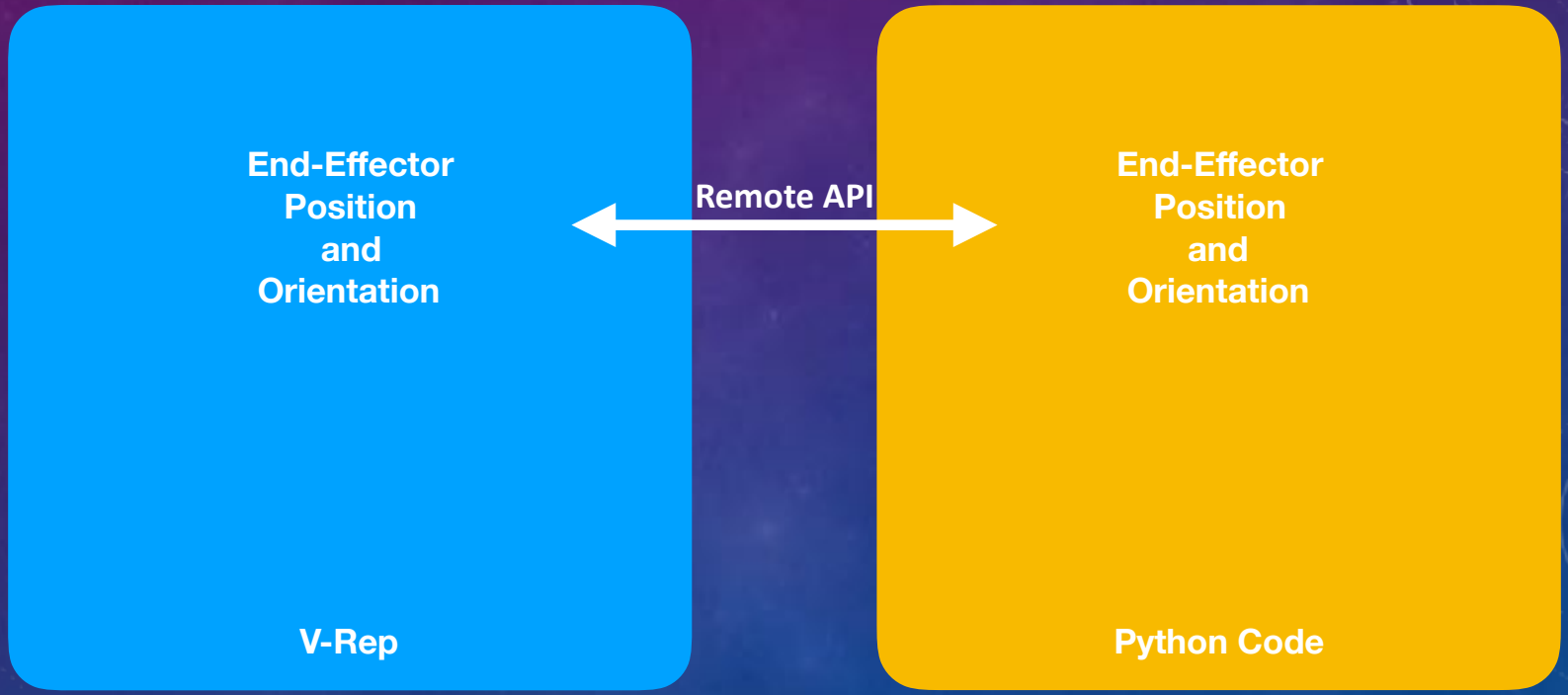
BUFFALO MANUFACTURING WORKS ROS PROJECT

POSSIBLE PATHS TO APPROACH

- V-REP ⟷ ROS ⟷ Python Script
- V-REP ⟷ Python Script
- Gazebo, URSim, or UR3 ⟷ ROS ⟷Python Script

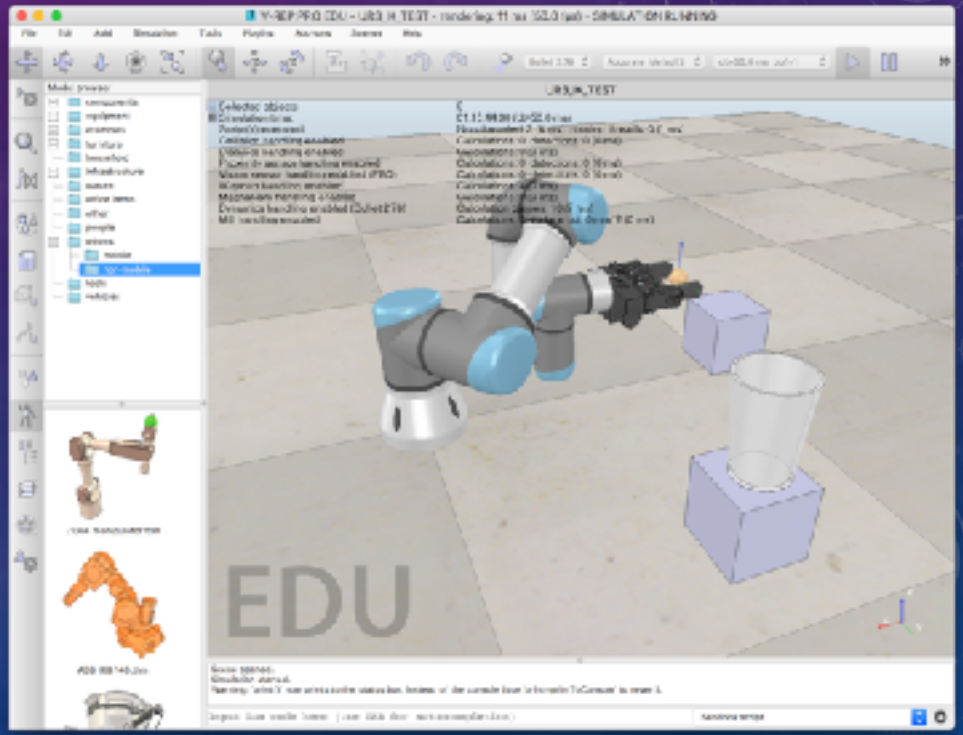

Robot Operating System
Joint Angle Topic
End-Effector Pos & Orientation Topic
Publisher
Subscriber
Publisher
Subscriber
Publisher
Subscriber
Publisher
Subscriber
Joint Angle
End-Effector
Position
Orientation
End-Effector
Position
Orientation
Joint Angle
V-Rep
Python Code

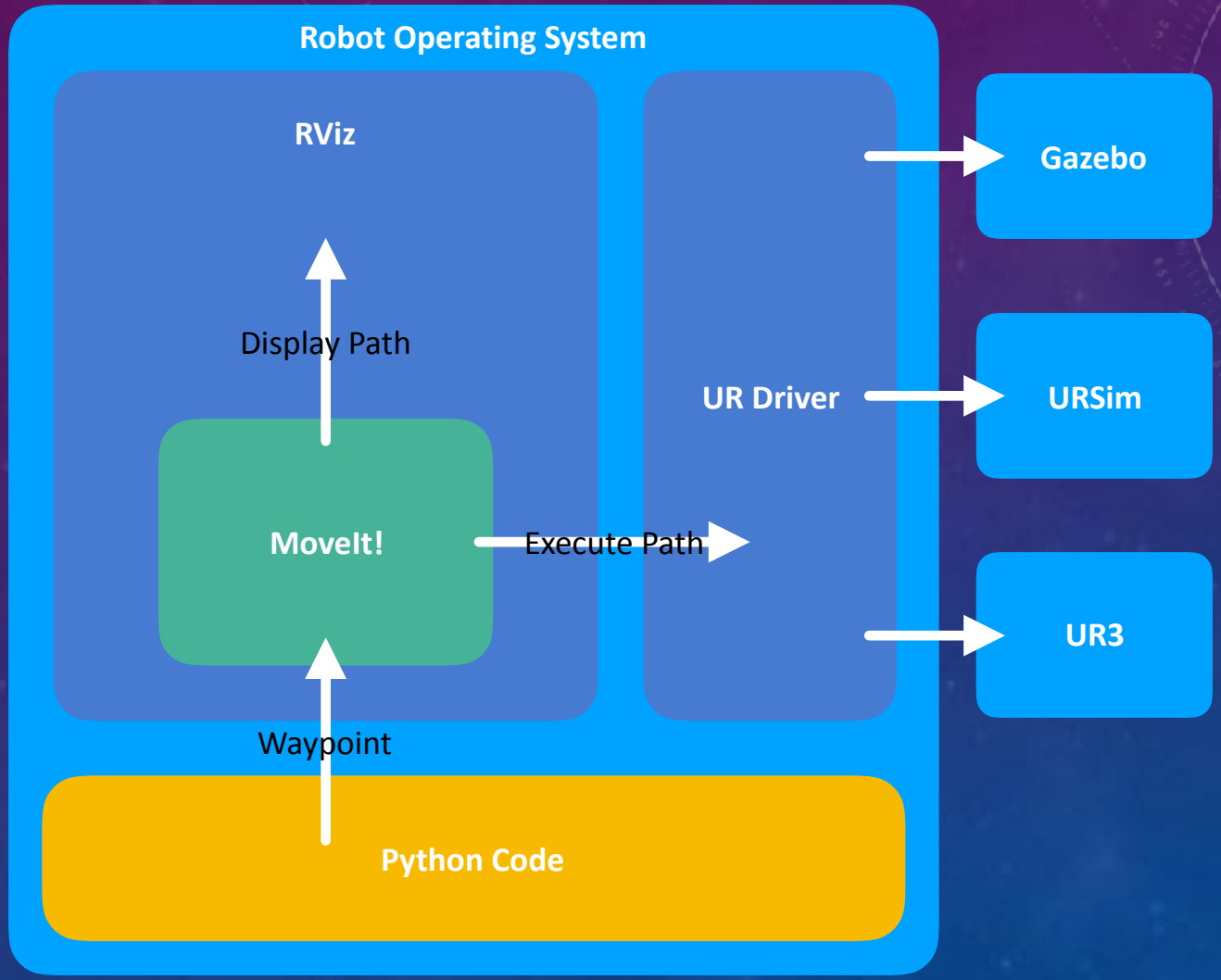
SIMPLIFIED VERSION (NOT USING ROS)



SIMULATION SOFTWARE

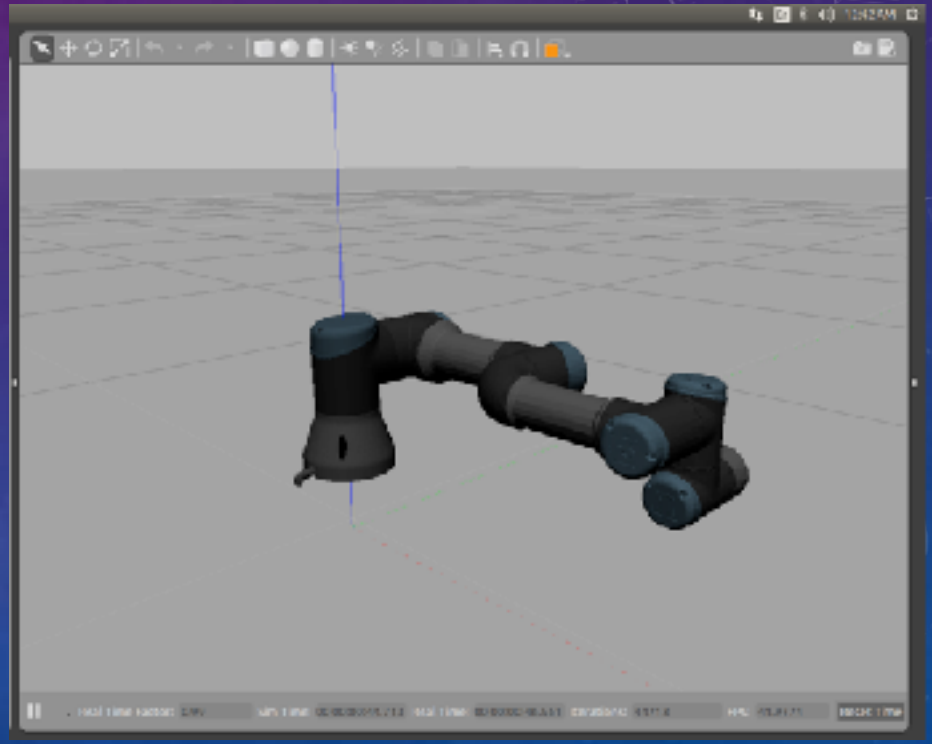
- V-Rep
  - Can easily interface with Python
  - Difficult to interface with ROS
    - Need to build publishers and subscribers from scratch
    - Need to Use LUA script
  - No matter using ROS or not, the simulation code cannot be used when using the real robot
  - TCP positions, orientations, and joint angles data can be collected.




Robot Operating System
RViz
Display Path
MoveIt!
Execute Path
Waypoint
Python Code
UR Driver
Gazebo
URSim
UR3

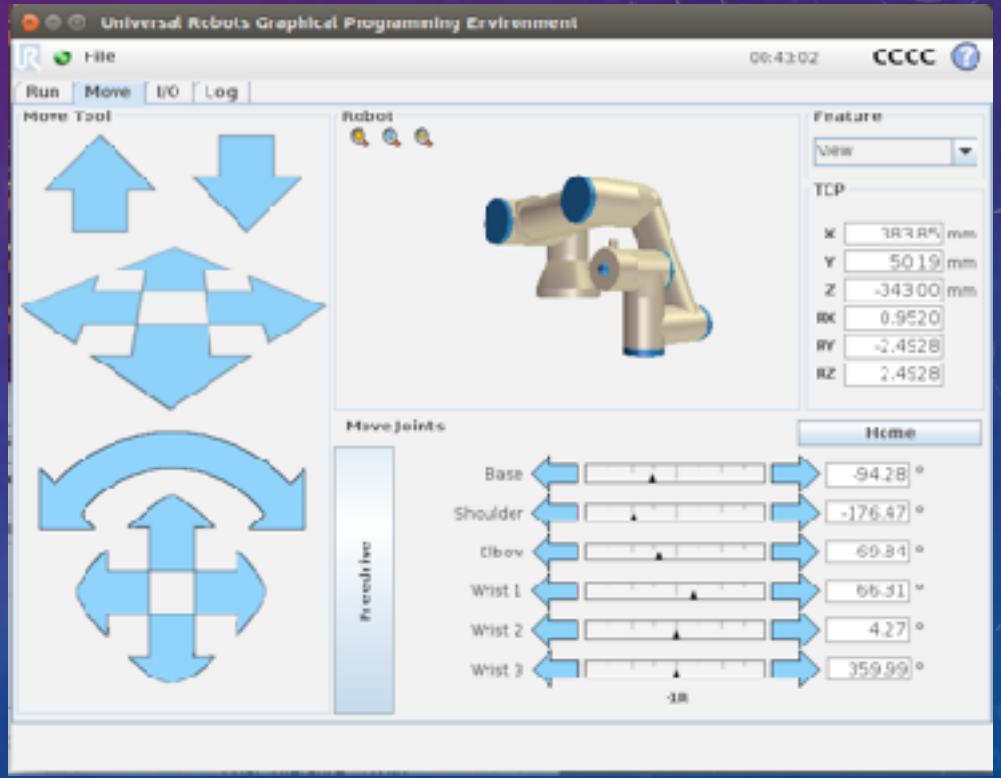
SIMULATION SOFTWARE

- Gazebo
  - Built-in with ROS, relatively easy to setup
  - Behaves as same as the real robot (ROS topics are the same)
  - Easily visualize the robot interacting with the environment



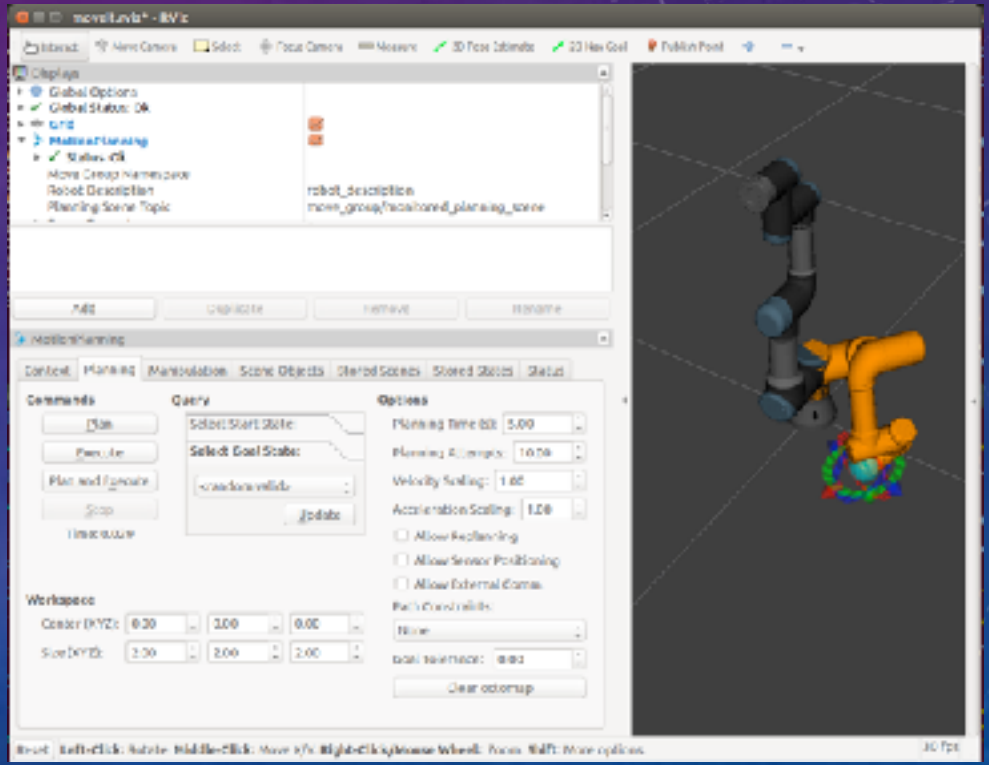
SIMULATION SOFTWARE

- URSim
  - Official simulation software from Universal Robot
  - Behaves exactly same as the real robot (ROS topic are the same)
  - Can show the real time Polyscope interface
  - Cannot visualize the robot in the environment



RVIZ AND MOVEIT

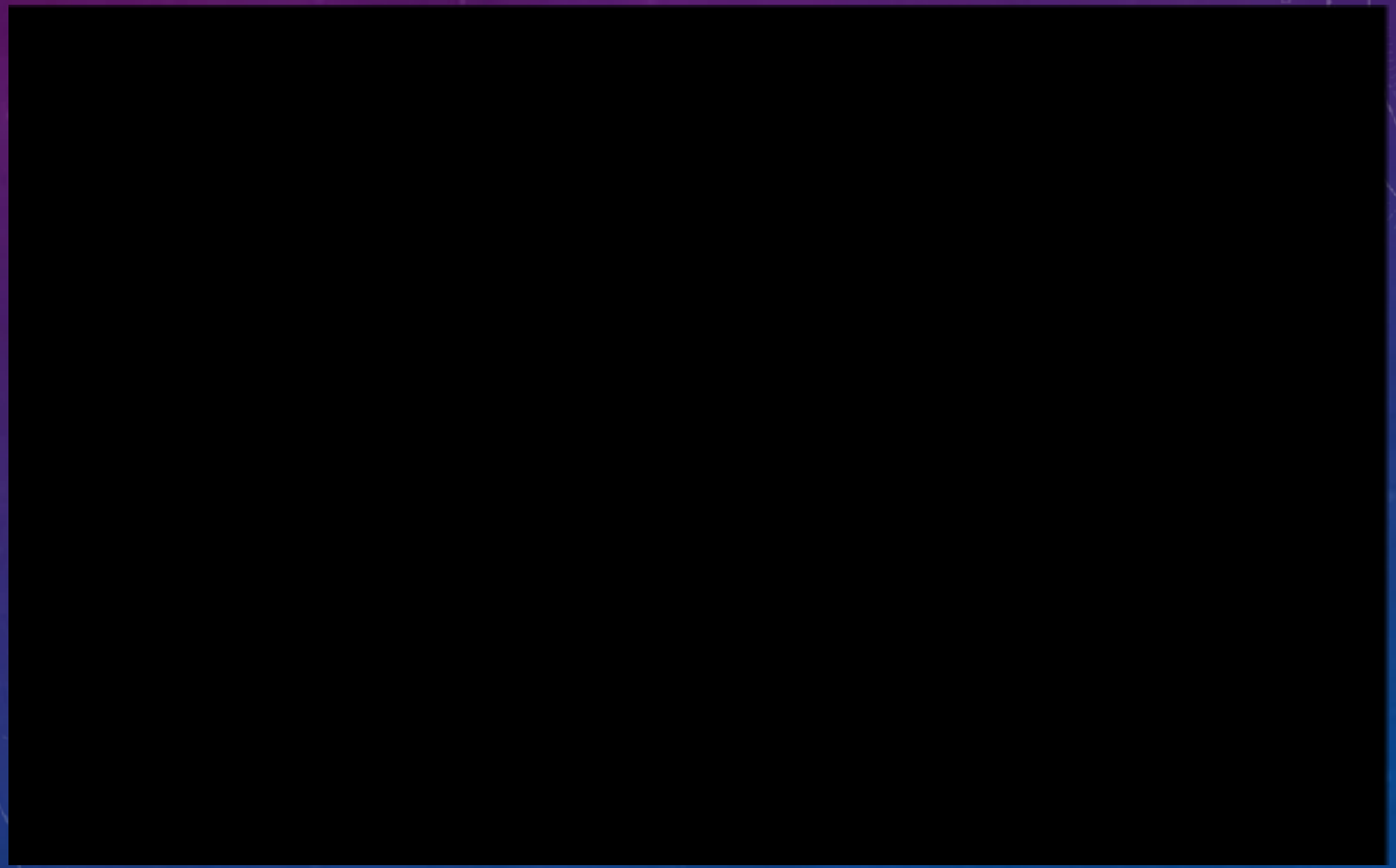
- When using Gazebo/URSim as simulation environment, use RViz with MoveIt plugin for path planning
- MoveIt is a path planning plugin on RViz
- RViz can show the planned path by MoveIt
- Used for IK (Inverse Kinematics) solving
- Python can interact with MoveIt for programmable path planning and path execution
- MoveIt can interact with UR driver which can



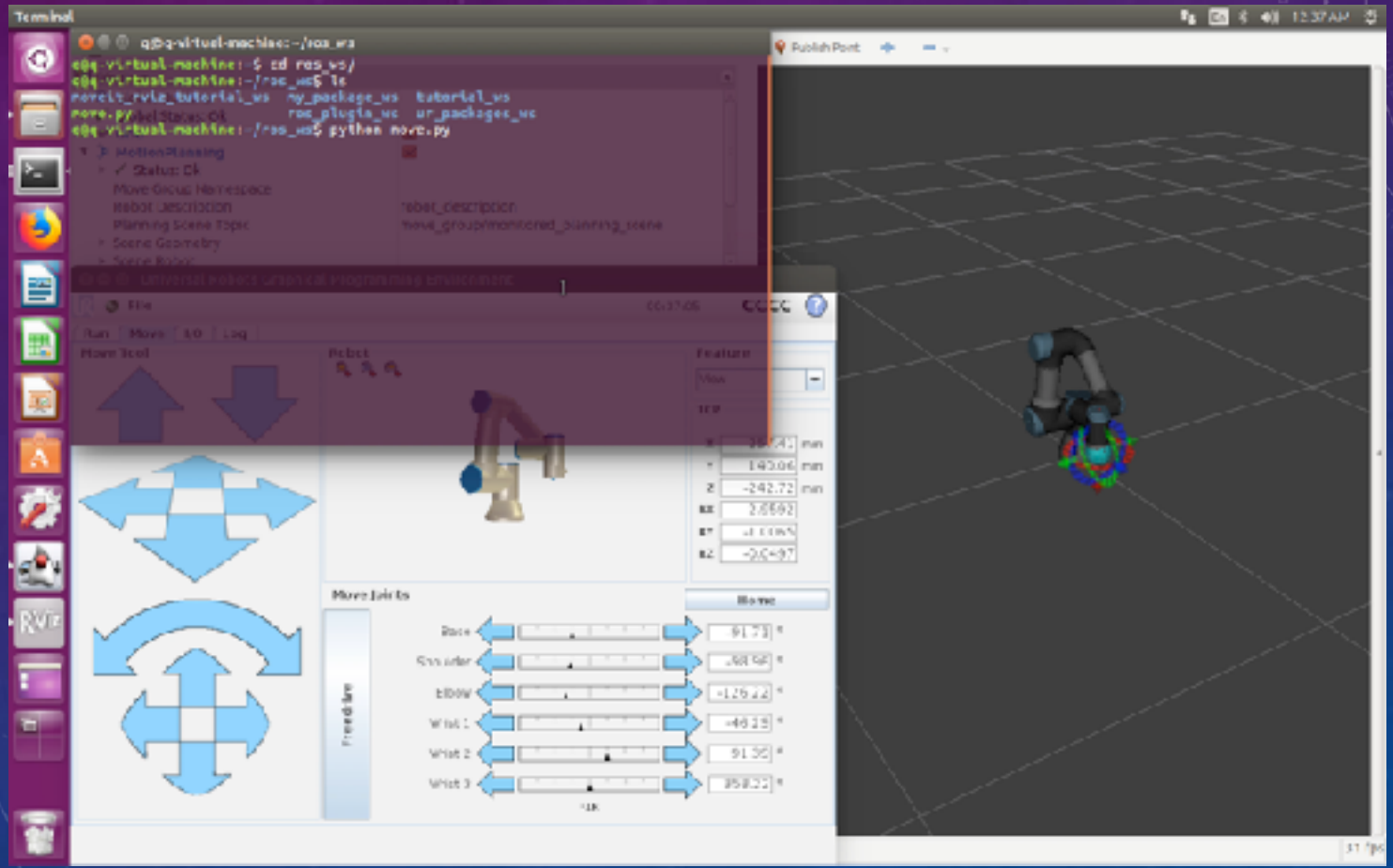
WHAT WE HAVE DONE

- Preliminary pick-and-place simulation using V-Rep and Python Remote API (See next slide)
- Configured ROS environment in a Ubuntu VM
- Installed UR related package (so we are able to control real UR3 using ROS)
- Figured out the relation between ROS, path planning software (RViz, MoveIt), the simulation software (Gazebo, URSim), and the Python script we will write
- Finished Programmable Path Planing using MoveIt

V-REP SIMULATION USING PYTHON REMOTE API



PATH PLANNING USING PYTHON AND MOVEIT



NEXT STEP...

- Integrate the simulation with the gripper and the sensor (micrometer).
  - Do research and see is there any ROS compatible driver package for the micrometer. If not, find a way to develop the driver package.
  - Develop the publisher node (python script) to publish the micrometer data to a ROS topic.
- Learn more about the details about this Quality Assurance process (geometry of the work piece, dimensions that need to be measured, etc.).
- Optimize the path using those information.
  - Modify the path planing code to subscribe the micrometer data topic so that the robot can move according to the data received.
- Finish the simulation, then perform testing on real robot.