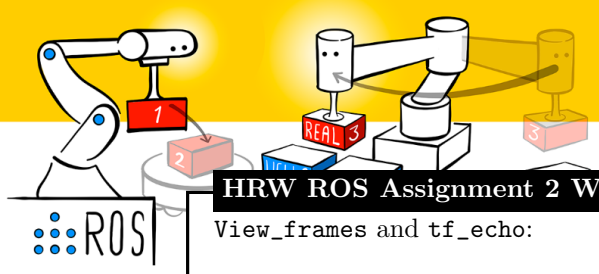



## HRW ROS Assignment 2 Week 5 Part 1

`View_frames` and `tf_echo`:

1. Start the now completed factory simulation, using the launcher for assignment 1: (if you have not completed assignment 1 this will not work)

   ```
   $ roslaunch hrwros_week5 week5_assignment1.launch
   ```

2. In a new CCS, execute the command:

   ```
   $ rosrun tf view_frames
   ```

3. Open the `frames.pdf` file.

   Verify that the TF frames `logical_camera_1_frame` and `logical_camera_2_frame` are now a part of the TF tree and the entire TF tree is connected without any breaks.

4. Go back to the CCS where you executed the `view_frames` command and execute the command

   ```
   $ rosrun tf tf_echo logical_camera_2_frame camera_rgb_frame
   ```

   You will first see some warnings and eventually the `tf_echo` command will start printing the TF information between the above-mentioned frames. After three updates with valid transform information, press `Ctrl+C`.

*Continued on the next page*

## HRW ROS Assignment 2 Week 5 Part 2

`Static_transform_publisher` and RViz Visualization

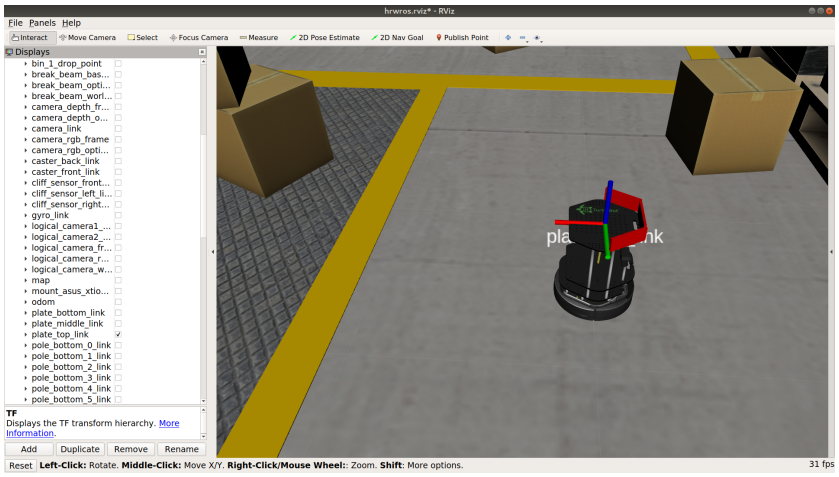
In this part, we will use the `static_transform_publisher` to publish a static transform between the `plate_top_link` and a new frame called `turtlebot_object_top` which will be located 20cm on top of the TurtleBot.

1. First terminate the factory simulation and restart it such that the Gazebo gui is NOT started and RViz is started.

   ```
   $ roslaunch hrwros_gazebo hrwros_environment.launch gui:=false rviz:=true
   ```

2. Then add an additional Robot Model display and modify the `robot_description` parameter of this display, such that the TurtleBot is visible in RViz.

3. Now, enable the TF visualization such that only the `plate_top_link` frame is shown, like in the screenshot below.

   This the most important part of the assignment: Publish a new static transformation.

4. On a new CCS, Publish a static transform from the `plate_top_link` to a new child frame named `turtlebot_object_top`.

   The the `turtlebot_object_top` frame should be 0.2m above the `plate_top_link` with the same orientation

   Recall that a static transform can be published as follows:

   ```
   $ rosrun tf2_ros static_transform_publisher <translation x y z> ...
         <rotation roll pitch yaw> <parent frame> <child frame>
   ```

   We need to visualize that everything is OK.

5. Go back to RViz and enable the newly published frame `turtlebot_object_top`. It should show up exactly on top of the reference frame for `plate_top_link`.

6. In a new CCS, start the turtlebot teleoperation with:

   ```
   $ roslaunch turtlebot_teleop keyboard_teleop.launch
   ```

   Move the TurtleBot around with the keyboard to a different location of your choice, you should notice that the newly published TF frame `turtlebot_object_top` moves along with the TurtleBot.

   Upload an RViz screenshot showing the two TF frames, `plate_top_link` and `turtlebot_object_top` with the TurtleBot at a location different to where it showed up initially.

*This completes HRW ROS Assignment 2 Week 5*