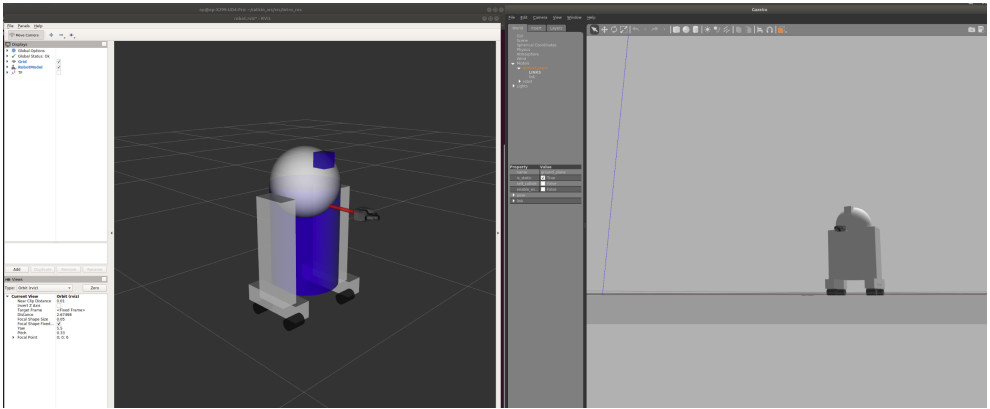
Introduction to ROS: Basics, Motion, and Vision

INNOPOLIS UNIVERSITY

R2-D2

R2-D2

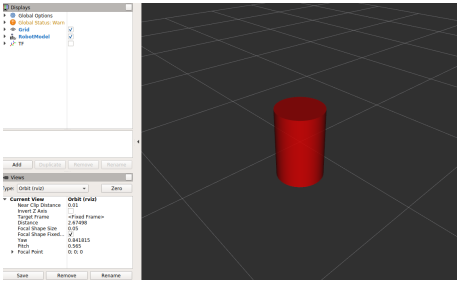


Step 01

1. Draw a cylinder 0.6 meter long and with a 0.2 meter radius
2. Where to place the center of the cylinder?
3. Can we add frame to the center of cylinder?
4. How to publish joints , i.e., sensor_msgs/JointState, information?
5. How can we load the parameters that define in the step_01.urdf?

```
roslaunch ros_urdf display_robot.launch model:='$(find ros_urdf)/urdf/step_01.urdf'
```

Step 01

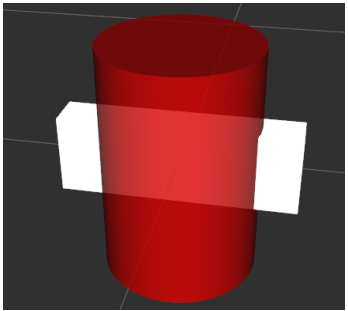

Displays
Global Options
Global Status: Warn
Grid
RobotModel
TF
Add
Duplicate
Remove
Rename
Views
Type: Orbit (rviz)
Zero
Current View
Orbit (rviz)
Near Clip Distance
0.01
Invert Z Axis
Target Frame
<Fixed Frame>
Distance
1.9598
Focal Shape Size
0.05
Focal Shape Fixed...
Yaw
0.841815
Pitch
0.565
Focal Point
0; 0; 0
Save
Remove
Rename

Step 02

1. How can we add more than one shape (or adding multiple shapes/links)?
2. If the first link is connected to the second link, how can we joint them?
3. Add a 0.6m $\times$ 0.6m $\times$ 0.6m box
4. If box is a child of cylinder how can specify it in the urdf?

```
roslaunch ros_urdf display_robot.launch model:='$(find ros_urdf)/urdf/step_02.urdf'
```

Step 02

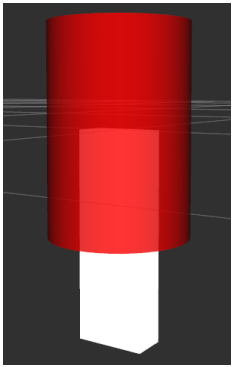


Step 03

1. How can we change the origin of the child with respect to parent?
2. How to change the position (xyz) and orientation (rpy (roll pitch yaw))of the box?

```
roslaunch ros_urdf display_robot.launch model:='$(find ros_urdf)/urdf/step_03.urdf'
```

Step 03

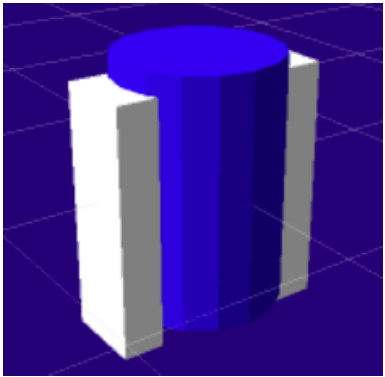


Step 04

1. How to change the color of the joints you have added?

```
roslaunch ros_urdf display_robot.launch model:=’$(find ros_urdf)/urdf/step_04.urdf’
```

Step 04

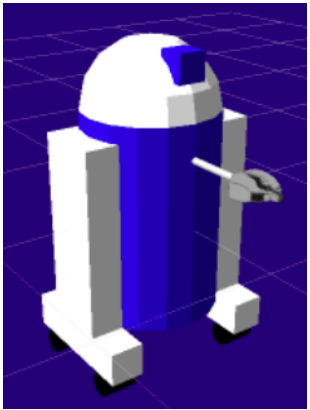


Step 05

1. Finish off the rest of the model

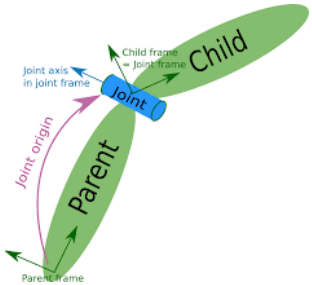
```
roslaunch ros_urdf display_robot.launch model:='$(find ros_urdf)/urdf/step_05.urdf'
```

Step 05



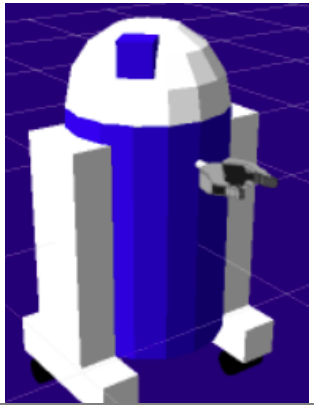
Step 06

1. How to define movable joints in URDF?
2. Continuous joint: can take on any angle from negative infinity to positive infinity on a specified axis, e.g., z axis by specifying 0 0 1
3. Revolute joints: rotate in the same way that the continuous joints do, but these type of joints have strict limits
4. Prismatic joints: moves along an axis, not around it



Step 06

roslaunch ros_urdf display_robot.launch model:='$(find ros_urdf)/urdf/step_06.urdf'



Step 07

1. Xacro, how to reduce repetitive things in the urdf
2. With Xacro, you can define constants, do simple math operations, and define macros

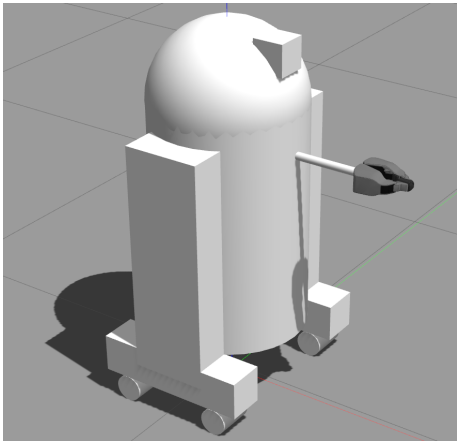
```
roslaunch ros_urdf display_robot.launch model:='$(find ros_urdf)/urdf/step_07.urdf'
```

Step 08

1. Try to simulate robot behaviour in Gazabo

```
roslaunch ros_urdf gazebo.launch model:='$(find ros_urdf)/urdf/step_08.urdf'
```

Step 08



Step 09

1. How to add controllers to control specified joints?

```
roslaunch ros_urdf gazebo.launch model:='$(find ros_urdf)/urdf/step_09.urdf'
```

Step 10

1. Let's try to go around?

```
roslaunch ros_urdf diffdrive.launch model:='$(find ros_urdf)/urdf/step_12.urdf'
```