

## 4.URDF model

---

### 1. Program function description

After the program is started, the URDF model file of the vehicle model will be displayed in rviz. There will also be a GUI debugging tool window. By sliding the progress bar, the model file can be moved.

### 2. Program code reference path

The source code of this function is located at:

```
/root/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_description/launch/display
.launch.py                                #docker container
/home/yahboom/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_description/launch
/display.launch.py                        #virtual machine side
```

Among them, the path of the URDF model file is:

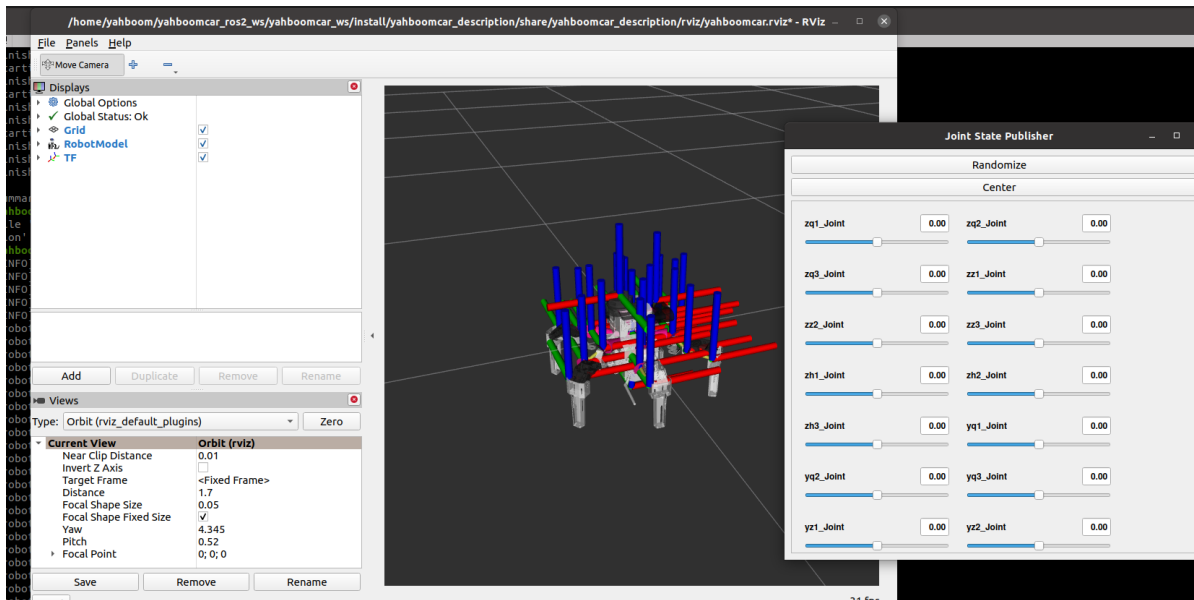
```
/root/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_description/urdf/Muto.urdf
#docker container
/home/yahboom/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_description/urdf/M
uto.urdf                                #virtual machine side
```

### 3. Program startup

After entering the docker container, according to the actual car model, enter the terminal:

```
ros2 launch yahboomcar_description display.launch.py
```

You can also choose to start it on the virtual machine side. The command is the same as above.

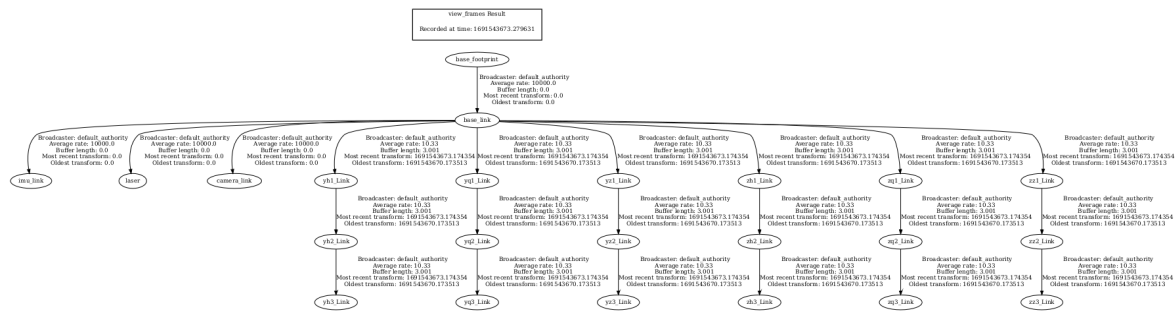


You can view the TF tree and docker terminal input,

```
ros2 run tf2_tools view_frames.py
```

A frame.pdf file will be generated in the terminal directory, and then we can open it and view it with the following command:

```
evince frames.pdf
```



## 4. URDF introduction

URDF, the full name is Unified Robot Description Format, which is translated into Chinese as Unified Robot Description Format. It is a kind of

Robot model file described in xml format, similar to D-H parameters.

```
<?xml version="1.0" encoding="utf-8"?>
<robot name="yahboomcar">
</robot>
```

The first line is a required item for XML and describes the version information of XML.

The second line describes the current robot name; all information about the current robot is contained in the [robot] tag.

## 4.1. Components

- link, connecting rod, can be imagined as a human arm
- joint, joint, can be imagined as a human elbow joint

The relationship between link and joint: Two links are connected through joints. Imagine that an arm has a forearm (link) and an upper arm (link) connected through an elbow joint (joint).

### 4.1.1. link

#### 1. Introduction

In the URDF descriptive language, link is used to describe physical properties.

- Describe the visual display, `tag`.
- Describe collision properties, `tag`.
- Describing physical inertia, the `tag` is not commonly used.

Links can also describe the size, color, shape, inertial matrix, collision parameters, etc. Each Link will become a coordinate system.

#### 2. Sample code

```
<link
  name="base_link">
  <inertial>
    <origin
      xyz="-0.00848711599566719 0.000523834964811633 -0.00410168576849108"
      rpy="0 0 0" />
    <mass
      value="0.53214037432126" />
    <inertia
      ixx="0.000304665507474336"
      ixy="7.99736134132429E-07"
      ixz="8.57646538118724E-06"
      iyy="0.000837714471641479"
      iyz="-9.52196363004723E-07"
      izz="0.00102369759156651" />
    </inertial>
    <visual>
      <origin
        xyz="0 0 0"
        rpy="0 0 0" />
      <geometry>
        <mesh
          filename="package://yahboomcar_description/meshes/base_link.STL" />
        </geometry>
        <material
          name="">
            <color
              rgba="1 1 1 1" />
          </material>
        </visual>
      <collision>
        <origin
          xyz="0 0 0"
          rpy="0 0 0" />
```

```

    <geometry>
      <mesh
        filename="package://yahboomcar_description/meshes/base_link.STL" />
      </geometry>
    </collision>
  </link>

```

### 3. Label introduction

- origin

It describes the pose information; the xyz attribute describes the coordinate position in the general environment, and the rpy attribute describes its own posture.

- mess

It describes the quality of link.

- inertia

The inertial reference system, due to the symmetry of the rotational inertia matrix, only requires 6 upper triangular elements  $ixx$ ,  $ixy$ ,  $ixz$ ,  $iyy$ ,  $iyz$ ,  $izz$  as attributes.

- geometry

The label describes the shape; the main function of the mesh attribute is to load the texture file, and the filename attribute is the file address of the texture path.

```

<box size="1 2 3"/> #The box body describes the length, width and height of
the box through the size attribute.
<cylinder length="1.6" radius="0.5"/> #The cylinder is cylindrical. The
`length` attribute describes the height of the cylinder, and the `radius`
attribute describes the radius of the cylinder.
<sphere radius="1"/> #sphere is spherical, describing the radius of the
sphere through the `radius` attribute.

```

- material

The label describes the material; the name attribute is **required**, can be empty, and can be repeated. Use the rgba attribute in the [color] tag to describe red, green, blue, and transparency, separated by spaces. The color range is [0-1].

## 4.1.2. joints

### 1. Introduction

Describe the relationship between two joints, motion position and speed limits, kinematic and dynamic properties.

Joint type:

- fixed: Fixed joint. No movement is allowed and it serves as a connection.
- continuous: rotating joint. Can continue to rotate, there is no limit on the rotation angle.
- revolute: rotate joint. Similar to continuous, there is a limit on the rotation angle.
- prismatic: sliding joint. Move along a certain axis, with position restrictions.
- floating: floating joint. It has six degrees of freedom, 3T3R.
- planar: plane joint. Allows translation or rotation orthogonally to the plane.

### 2. Sample code

```

<joint
  name="camera_Joint"
  type="fixed">
  <origin
    xyz="0.0710921136864663 0 0.095735206628551"
    rpy="0 0 0" />
  <parent
    link="base_link" />
  <child
    link="camera_link" />
  <axis
    xyz="0 0 0" />
</joint>

```

In the [joint] tag, the name attribute is **required**, which describes the name of the joint and is unique.

In the type attribute of the [joint] tag, fill in the six joint types correspondingly.

### 3.tag introduction

- origin

The child label refers to the relative position of the rotating joint to the coordinate system where the parent is located.

- parent, child

The parent and child subtags represent two links to be connected; parent is the reference object, and child rotates around parent.

- axis

The child tag indicates which axis (xyz) the link corresponding to the child rotates around and the amount of rotation around the fixed axis.

- limit

Subtags are mainly used to restrict children. The lower attribute and upper attribute limit the arc range of rotation, and the effort attribute limits the force range during rotation. (positive and negative value, unit is Newton or N), velocity attribute limits the speed during rotation, unit is meter/second or m/s.

- mimic

Describe the relationship of this joint to existing joints.

- safety\_controller

Describes safety controller parameters. Protect the movement of robot joints.