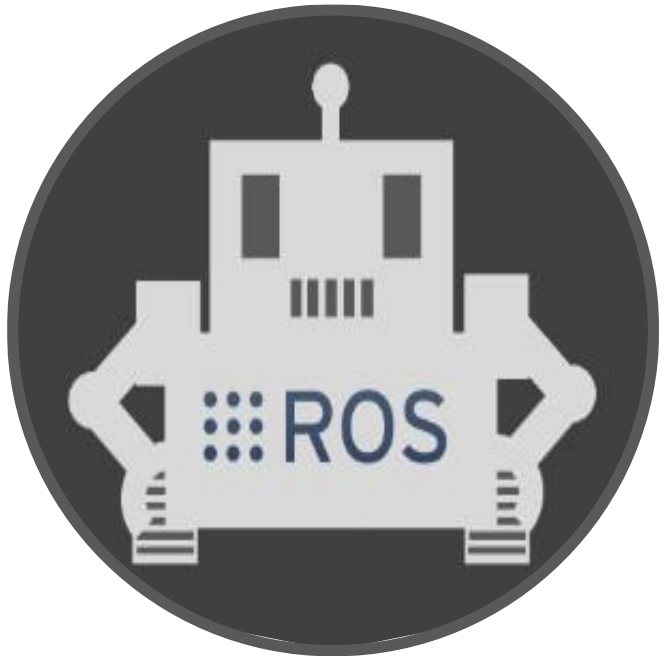
ROS 기초 강의

Chapter 9. ROS TF

구선생 로보틱스



Update 20240620

강의 자료 다운로드

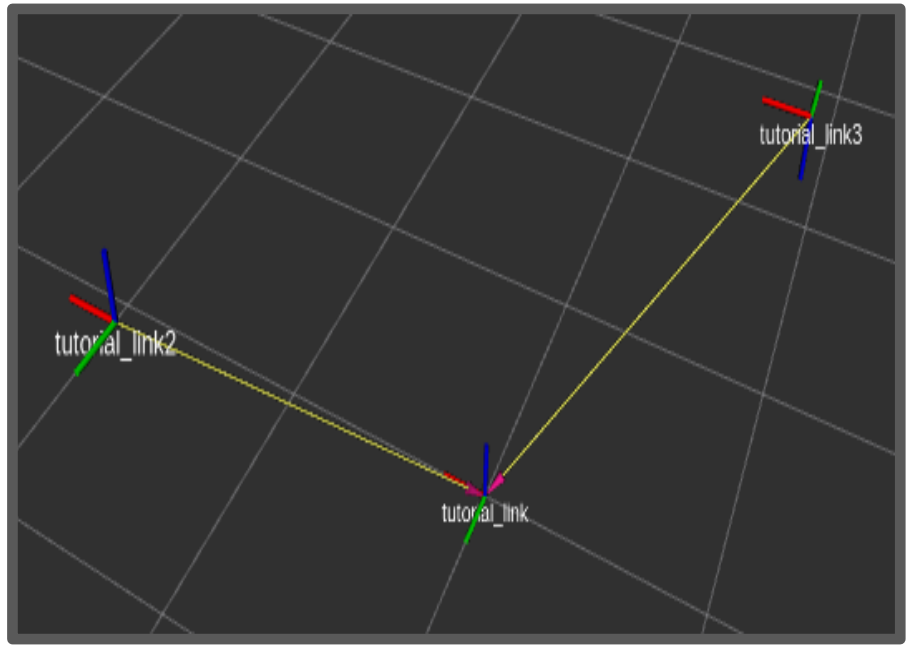
ROS 기초 강의 강의노트

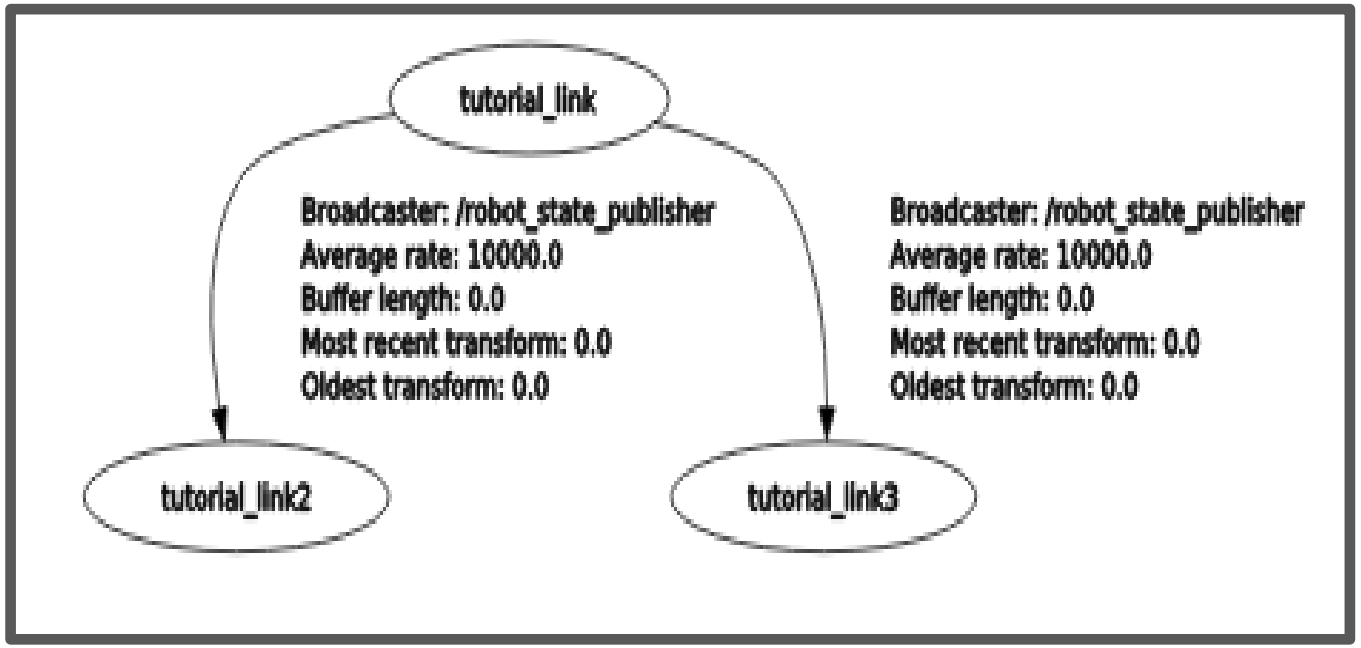
https://drive.google.com/drive/folders/1rRwS2j98HNyj5Is_yVXEGj30ILvMPtrz?usp=drive_link

1. ROS TF
2. ROS URDF

ROS TF

ROS TF란?





동영상 강의 – ROS TF 기초
https://youtu.be/mbAKwCXb9vs?si=1SM5a8ZnwunGxhht

ROS에서 좌표계끼리의 관계를 정의하는 도구

ROS TF

TF 생성

1) 패키지 생성

동영상 강의 - ROS TF 생성
https://youtu.be/2V2730XV0o0?si=v4yt92yxZckddyfM

```
$ catkin_create_pkg basic_tf_tutorial roscpp
```

```
ubuntu@ubuntu:~/catkin_ws/src$ catkin_create_pkg basic_tf_tutorial roscpp
Created file basic_tf_tutorial/package.xml
Created file basic_tf_tutorial/CMakeLists.txt
Created folder basic_tf_tutorial/include/basic_tf_tutorial
Created folder basic_tf_tutorial/src
Successfully created files in /home/ubuntu/catkin_ws/src/basic_tf_tutorial. Please adjust the values in package.xml.
```

소스코드 - basic_tf_tutorial
https://github.com/PigeonSensei/pigeon_ros_tutorial/tree/master/basic/basic_tf_tutorial

ROS TF

TF 생성

2) 소스코드 파일 생성

```
$ nano basic_tf.cpp
```

```
ubuntu@ubuntu:~/catkin_ws/src$ cd basic_tf_tutorial/
ubuntu@ubuntu:~/catkin_ws/src/basic_tf_tutorial$ cd src
ubuntu@ubuntu:~/catkin_ws/src/basic_tf_tutorial/src$ nano basic_tf.cpp
```

∵ 소스코드는 패키지의 src 경로에서 생성되어야 한다.

ROS TF

TF 생성

3) 소스코드 작성

```
#include <ros/ros.h>
#include "tutorial_srvs/TutorialSrv.h"

bool TutorialCommandServiceCallback(tutorial_srvs::TutorialSrv::Request &req,
                                    tutorial_srvs::TutorialSrv::Response &res)
{
  if(req.command == "tutorial 1")
  {
      ROS_INFO("Receive Service call tutorial command : tutorial 1");
      res.message = "Receive success";
      res.result = "true";
  }
  else if(req.command == "tutorial 2")
  {
      ROS_INFO("Receive Service call tutorial command : tutorial 2");
      res.message = "Receive success";
      res.result = "true";
  }

  else
  {
    ROS_INFO("Receive Service call tutorial command : non");
    res.message = "Receive fall";
    res.result = "false";
  }

  return true;

}

```

ROS TF

TF 생성

4) CMakeLists.txt 편집

```
$ nano CMakeLists.txt
```

```
ubuntu@ubuntu:~/catkin_ws/src/basic_tf_tutorial/src$ cd ..
ubuntu@ubuntu:~/catkin_ws/src/basic_tf_tutorial$ ls
CMakeLists.txt  include  package.xml  src
ubuntu@ubuntu:~/catkin_ws/src/basic_tf_tutorial$ nano CMakeLists.txt
```

∴ CMakeLists.txt는 패키지 경로에 있다.

ROS TF

TF 생성

5) CMakeLists.txt의 find_package 내용 추가

```
10  find_package(catkin REQUIRED COMPONENTS
11    roscpp
12    tf
13  )
```

6) CMakeLists.txt add_executable, target_link_librariese, add_dependencies 추가

```
122  add_executable(basic_tf_node src/basic_tf.cpp)
123  target_link_libraries(basic_tf_node ${catkin_LIBRARIES})
```

ROS TF

TF 생성

7) CMakelists.txt의 find_packag에 추가 된 내용을 package.xml에 추가

```
13    <build_depend>roscpp</build_depend>
14    <build_depend>tf</build_depend>
15
16    <build_export_depend>roscpp</build_export_depend>
17    <build_export_depend>tf</build_export_depend>
18
19    <exec_depend>roscpp</exec_depend>
20    <exec_depend>tf</exec_depend>
21
```

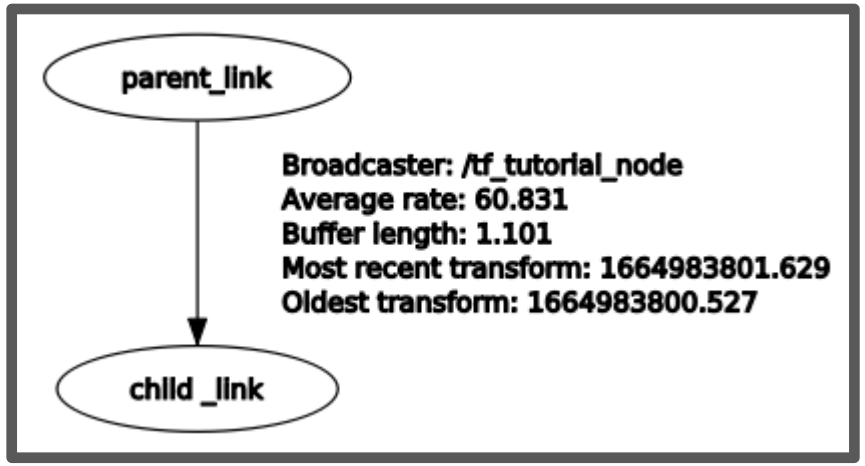
ROS TF

TF 생성

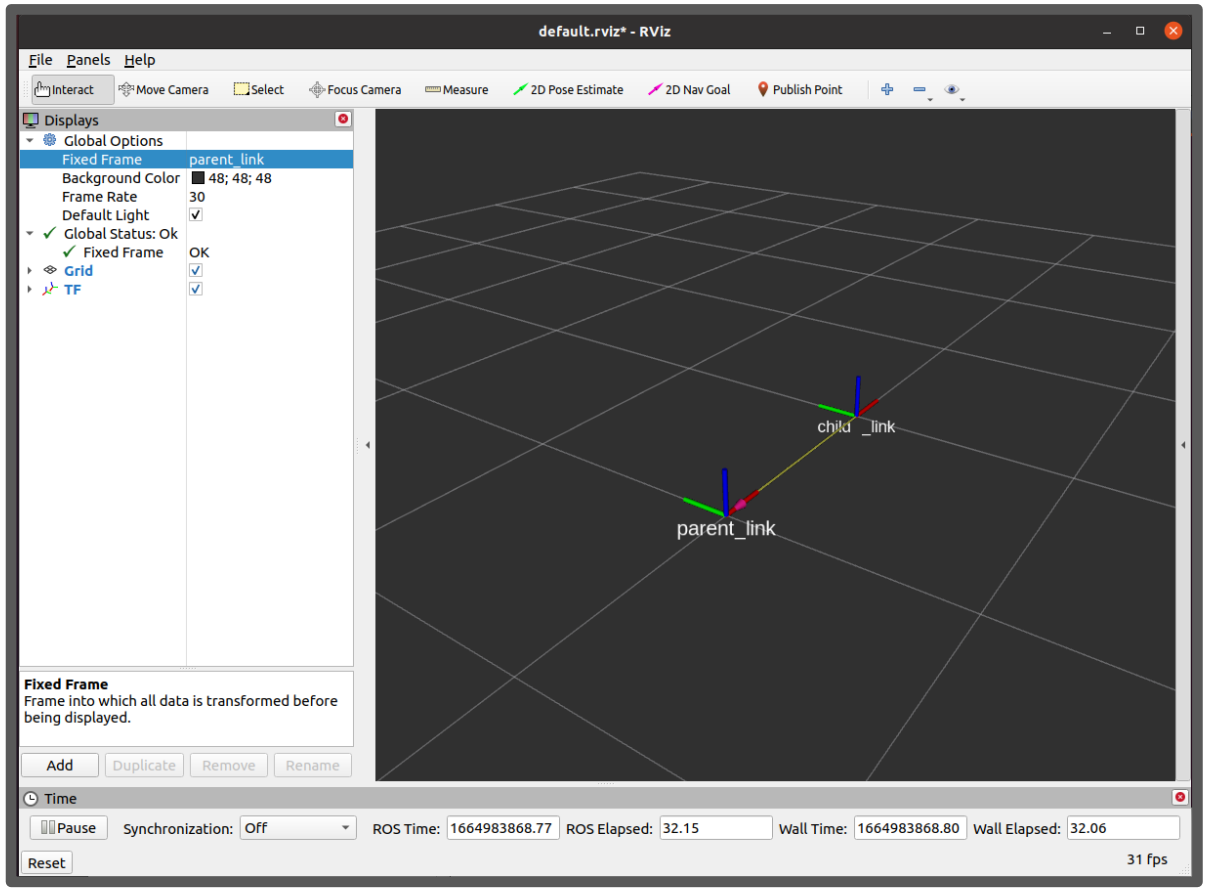
- TF Node실행 결과

```
ubuntu@ubuntu:~/catkin_ws$ rosrun basic_tf_tutorial basic_tf_node
[ INFO] [1691574814.569268119]: basic_tf_node Open
```

- rqt_tf_tree



- Rviz



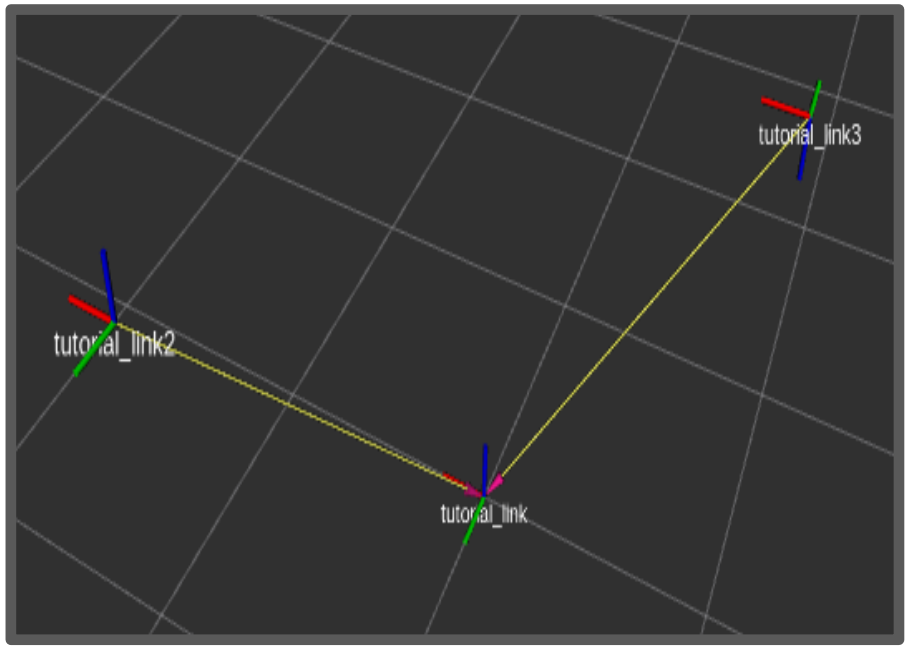
1. ROS TF
2. ROS URDF

ROS URDF

URDF란?

```xml
<?xml version="1.0"?>

<robot name="Tutorial">

  <link name="tutorial_link"/>
  <link name="tutorial_link2"/>
  <link name="tutorial_link3"/>

  <joint name="joint1" type="fixed">
    <parent link="tutorial_link" />
    <child link="tutorial_link2" />
    <origin rpy="0 0 0" xyz="1.0 1.0 1.0" />
  </joint>

  <joint name="joint2" type="fixed">
    <parent link="tutorial_link" />
    <child link="tutorial_link3" />
   <origin rpy="3.141592 0 0" xyz="-1.0 -1.0 1.0" />
  </joint>

</robot>
```

동영상 강의 - ROS URDF 기초
https://youtu.be/XnuML7rr4Yc?si=l4d_MvNEQgdhahYI



"Unified Robot Description Format"의 줄임말로
로봇모델에 대한 정보들을 명세해놓은 규격서

ROS URDF

URDF 생성

동영상 강의 – ROS URDF 생성
https://youtu.be/r5OyG4Nybyo?si=LteFiDTpnAOFgbd5

1) 패키지 생성

```
$ catkin_create_pkg basic_urdf_tutorial roscpp
```

```
ubuntu@ubuntu:~/catkin_ws/src$ catkin_create_pkg basic_urdf_tutorial roscpp
Created file basic_urdf_tutorial/package.xml
Created file basic_urdf_tutorial/CMakeLists.txt
Created folder basic_urdf_tutorial/include/basic_urdf_tutorial
Created folder basic_urdf_tutorial/src
Successfully created files in /home/ubuntu/catkin_ws/src/basic_urdf_tutorial. Please adjust the values in package.xml.
```

소스코드 – basic_urdf_tutorial
https://github.com/PigeonSensei/pigeon_ros_tutorial/tree/master/basic/basic_urdf_tutorial

ROS URDF

URDF 생성

2) urdf 파일 생성

```
$ nano tutorial.urdf
```

```
ubuntu@ubuntu:~/catkin_ws/src$ cd basic_urdf_tutorial/
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial$ mkdir urdf
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial$ ls
CMakeLists.txt  include  package.xml  src  urdf
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial$ cd urdf/
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial/urdf$ nano tutorial.urdf
```

∴ urdf 파일은 패키지의 urdf 경로에서 생성되어야 한다.
urdf 경로가 없다면 생성한다.

ROS URDF

URDF 생성

3) urdf 작성

```
1  <?xml version="1.0"?>
2
3  <robot name="Tutorial">
4
5    <link name="tutorial_link"/>
6    <link name="tutorial_link2"/>
7    <link name="tutorial_link3"/>
8
9    <joint name="joint1" type="fixed">
10     <parent link="tutorial_link" />
11     <child link="tutorial_link2" />
12     <origin rpy="0 0 0" xyz="1.0 1.0 1.0" />
13   </joint>
14
15   <joint name="joint2" type="fixed">
16     <parent link="tutorial_link" />
17     <child link="tutorial_link3" />
18     <origin rpy="3.141592 0 0" xyz="-1.0 -1.0 1.0" />
19   </joint>
20
21 </robot>
```

ROS URDF

URDF 생성

4) launch 파일 생성

```
$ nano urdf_tutorial.launch
```

```
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial/urdf$ cd ..
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial$ mkdir launch
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial$ ls
CMakeLists.txt  include  launch  package.xml  src  urdf
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial$ cd launch/
ubuntu@ubuntu:~/catkin_ws/src/basic_urdf_tutorial/launch$ nano urdf_tutorial.launch
```

∵ launch 파일은 패키지의 launch 경로에서 생성되어야 한다.
launch 경로가 없다면 생성한다.

ROS URDF

URDF 생성

5) launch 파일 작성

```
1  <?xml version="1.0"?>
2
3  <launch>
4
5  <!-- ========================== URDF  ================================== -->
6
7    <param name="robot_description"
8      textfile="$(find basic_urdf_tutorial)/urdf/tutorial.urdf" />
9
10   <node name="robot_state_publisher" pkg="robot_state_publisher"
11     type="robot_state_publisher" />
12
13 </launch>
```

ROS URDF

URDF 생성

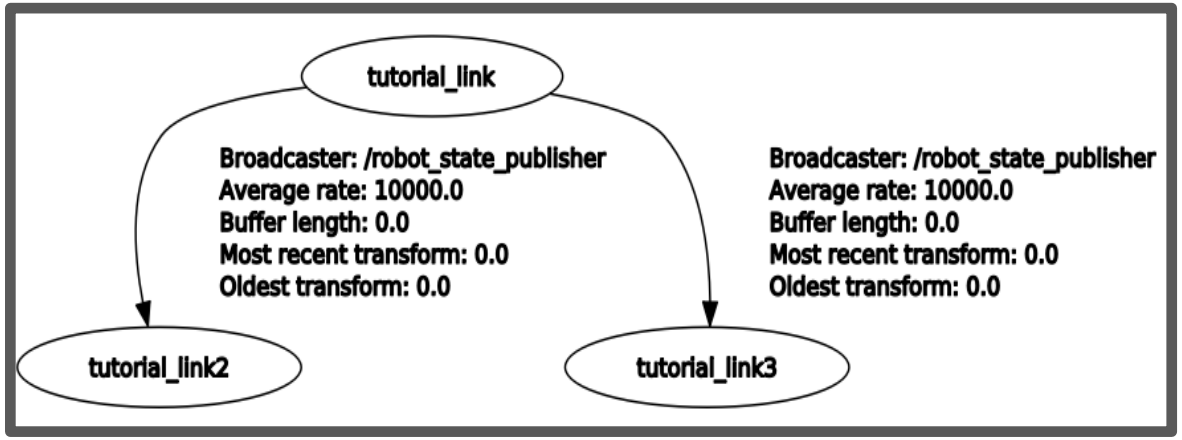
- urdf_tutorial.launch 실행 결과

```
ubuntu@ubuntu:~/catkin_ws$ roslaunch basic_urdf_tutorial urdf_tutorial.launch
... logging to /home/ubuntu/.ros/log/eb1083f8-369d-11ee-9f8e-ebe4e0527a1e/roslaunch-ubuntu-6005.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://192.168.209.128:42839/

SUMMARY
========

PARAMETERS
 * /robot_description: <?xml version="1....
 * /rosdistro: noetic
 * /rosversion: 1.16.0

NODES
  /
    robot_state_publisher (robot_state_publisher/robot_state_publisher)

ROS_MASTER_URI=http://192.168.209.128:11311

process[robot_state_publisher-1]: started with pid [6019]
```

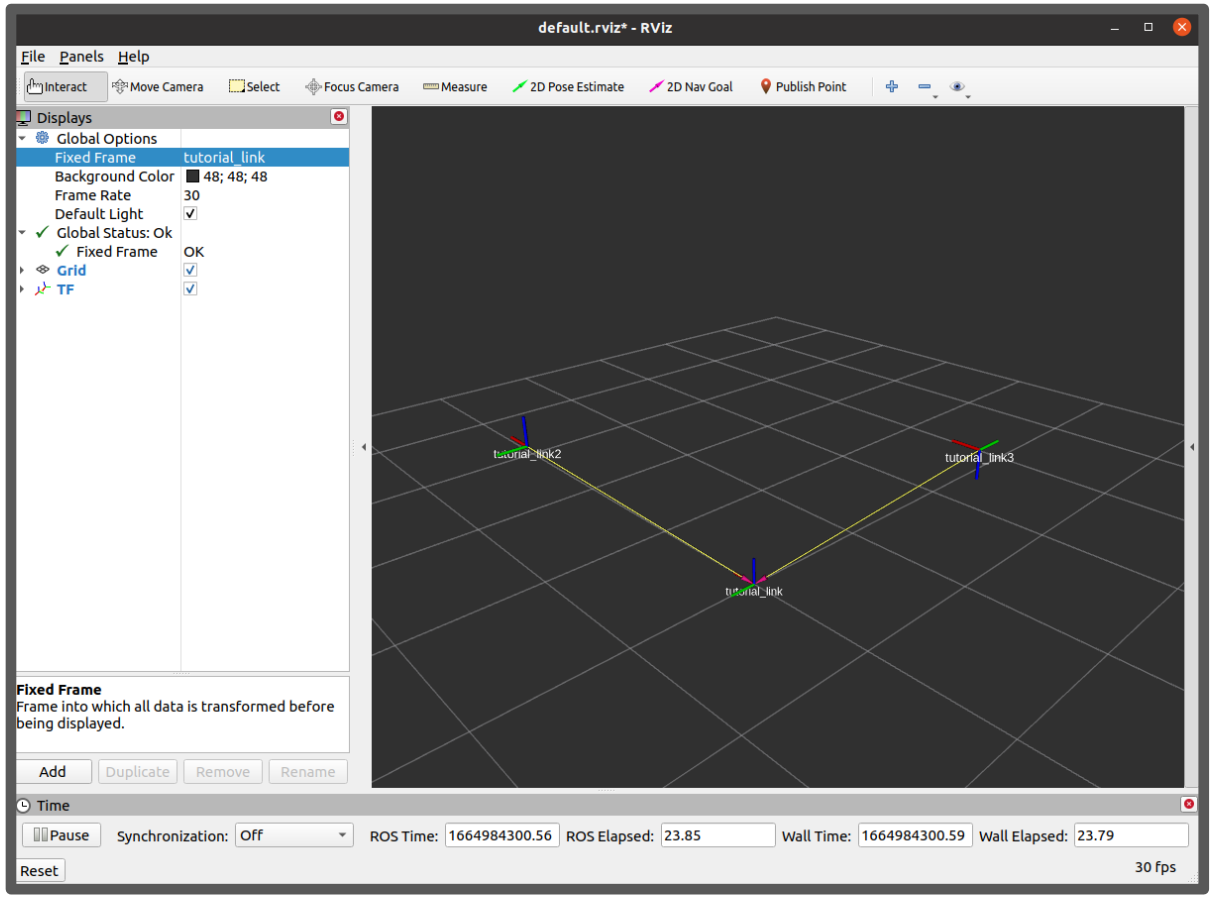
ROS URDF

URDF 생성

- rqt_tf_tree



- Rviz



감사합니다

구선생 로보틱스