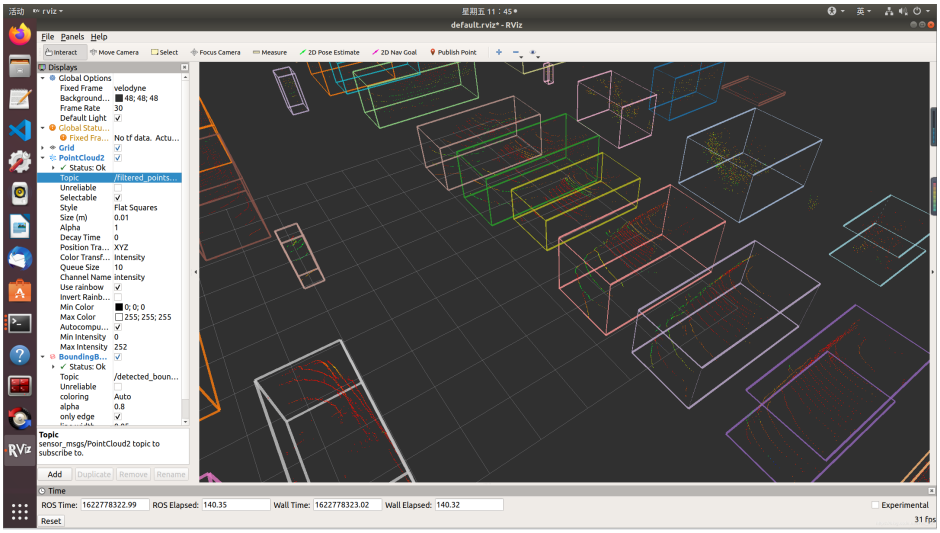
教程所需代码包下载传送门

不太擅长写教程，但是我感觉写的挺详细了，虽然字比较多，但是按照我的流程，耐心点一步步跟着做，我觉得实现最终的效果还是蛮简单的，不懂得也可以直接评论区描述你的问题，关于资料下载，有积分或者会员的直接上边传送门去下载，也感谢支持。没有的也没关系，百度网盘连接给各位大佬安排上：

https://pan.baidu.com/s/1S4sr7lEZkkZ-cbyXSgo8eQ

提取码：w6my

先附上最终效果图：



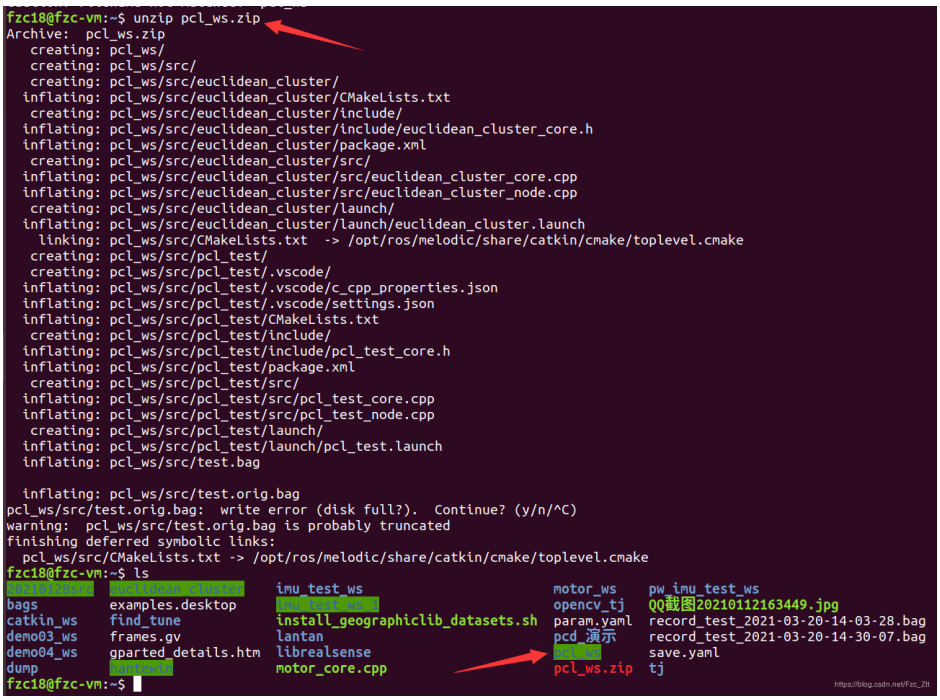
环境介绍：我分别在ubuntu18.04+ros melodic和ubuntu16.04+ros kinetic两种环境搭配下，都是可以跑的通的，其他版本应该问题也不大。

1. 下载代码文件夹后解压，解压后的文件已经是一个标准的ros工作空间，只需在该工作空间下，正常编译就可以，编译如果未通过，大概率是因为缺少编译相对应的ros包，根据实际情况去安装缺少的包就可以了，安装完成后再重新编译，基本上不会有什么问题。

我的文件放在当前用户目录下，即 /home/fzc18

可以在文件夹中找到下载好的压缩包右击选择解压，也可以命令行解压

```
unzip pcl_ws.zip
```

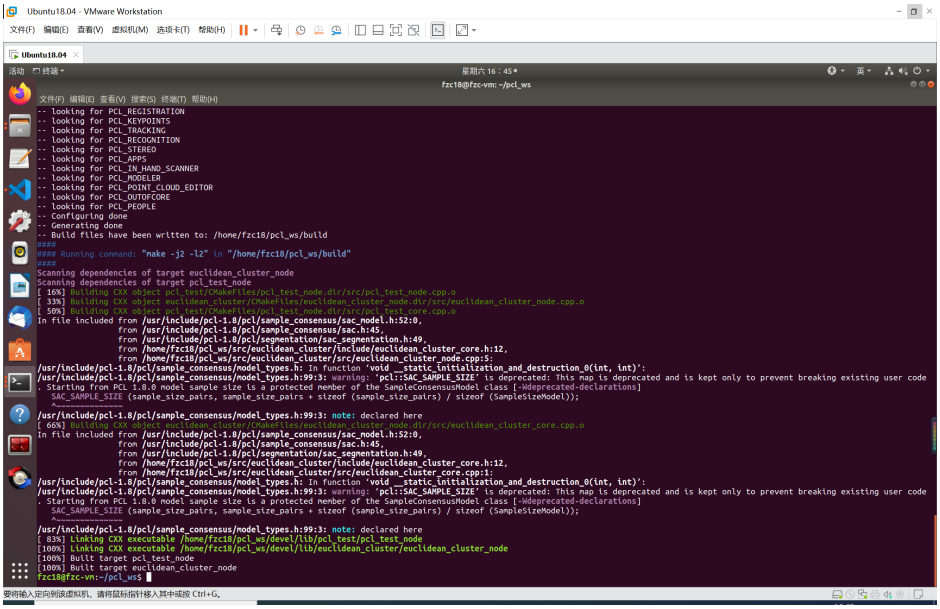


进入解压后的文件夹并编译

```
cd pcl_ws
catKin_make
```

```
fzc18@fzc-vm:~$ cd pcl_ws/
fzc18@fzc-vm:~/pcl_ws$ catkin_make
Base path: /home/fzc18/pcl_ws
Source space: /home/fzc18/pcl_ws/src
Build space: /home/fzc18/pcl_ws/build
Devel space: /home/fzc18/pcl_ws/devel
Install space: /home/fzc18/pcl_ws/install
####
```

应为我之前编译过，该安装的包也都安装了，所以一遍就编译通过了，你们需要根据错误提示，安装相应的包



编译通过后，就可以执行代码开始运行了。
稍等，为了后面bounding box能够顺利进行，还需要安装三个相应的包，这3个包不是编译所必需的，但是是可视化时所依赖的包：

```
sudo apt install ros-melodic-jsk-recognition-msgs ros-melodic-jsk-rqt-plugins ros-melodic-jsk-visualization
//sudo apt install ros-kinetic-jsk-recognition-msgs ros-kinetic-jsk-rqt-plugins ros-kinetic-jsk-visualization
```

注意:如果用的ros版本不同，要将代码中的三个“melodic”替换成“kinetic”或者你所使用的其他ros版本名称（防止小白无法复现，所以说的详细点）。

开始运行（要确保上边的编译通过了，而且改装的包也安装好了，如果文件路径和我的不同，需要在需要修改的地方做一些路径上的调整）

注意： 每一组代码都要打开一个单独的命令窗口，一共需要打开4个窗口。
第一组：

```
cd pcl_ws/
source devel/setup.bash
roslaunch pcl_test pcl_test.launch
```

第二组：

```
cd pcl_ws/
source devel/setup.bash
roslaunch euclidean_cluster euclidean_cluster.launch
```

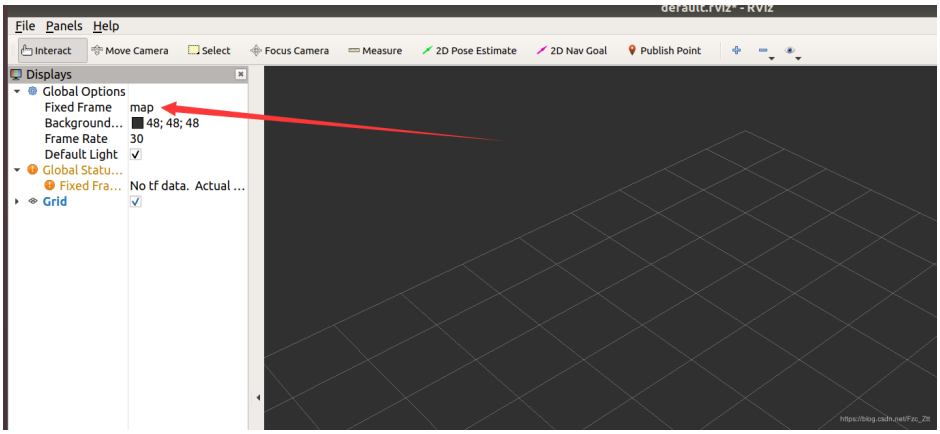
第三组：

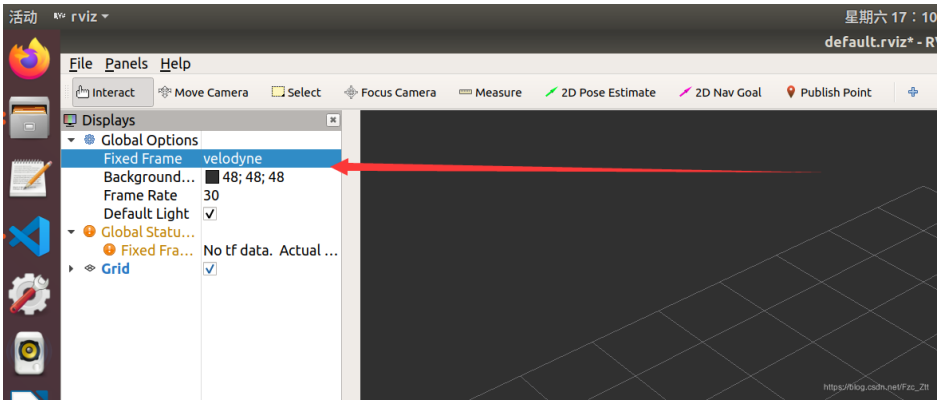
```
cd pcl_ws/src/
rosbag play --clock test.bag
```

第四组：

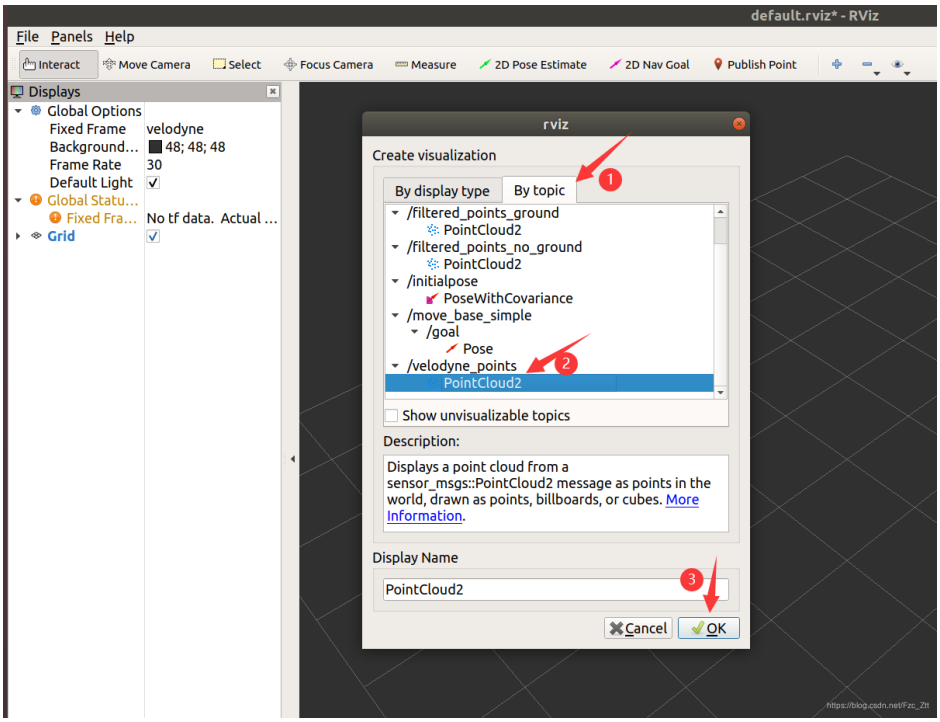
```
rosrun rviz rviz
```

将“map"改为"velodyne",（此步骤为手动输入，因为下拉选项内没有）

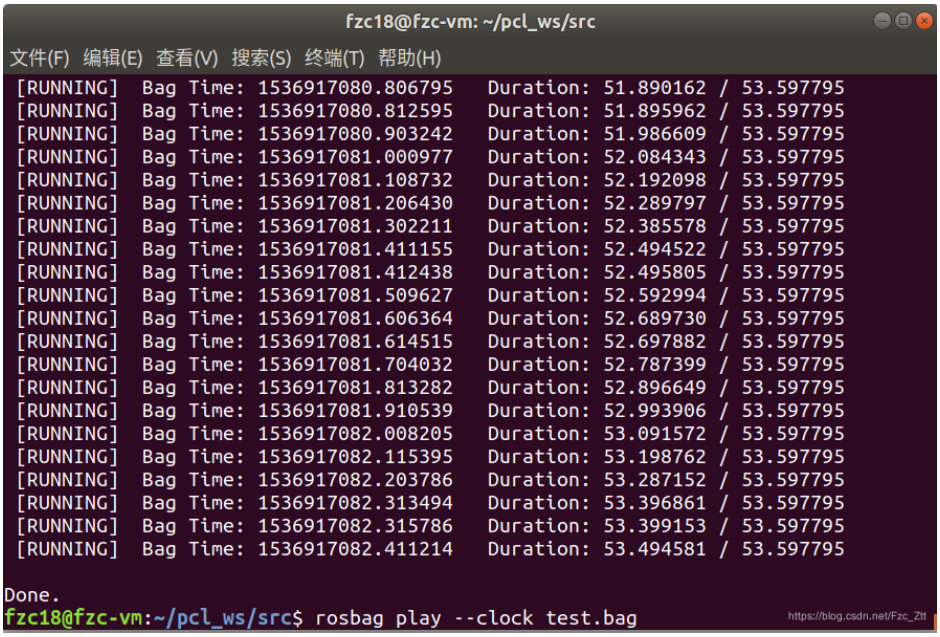




添加话题 Add→By topic→/velodyne_point/PointCloud2→OK

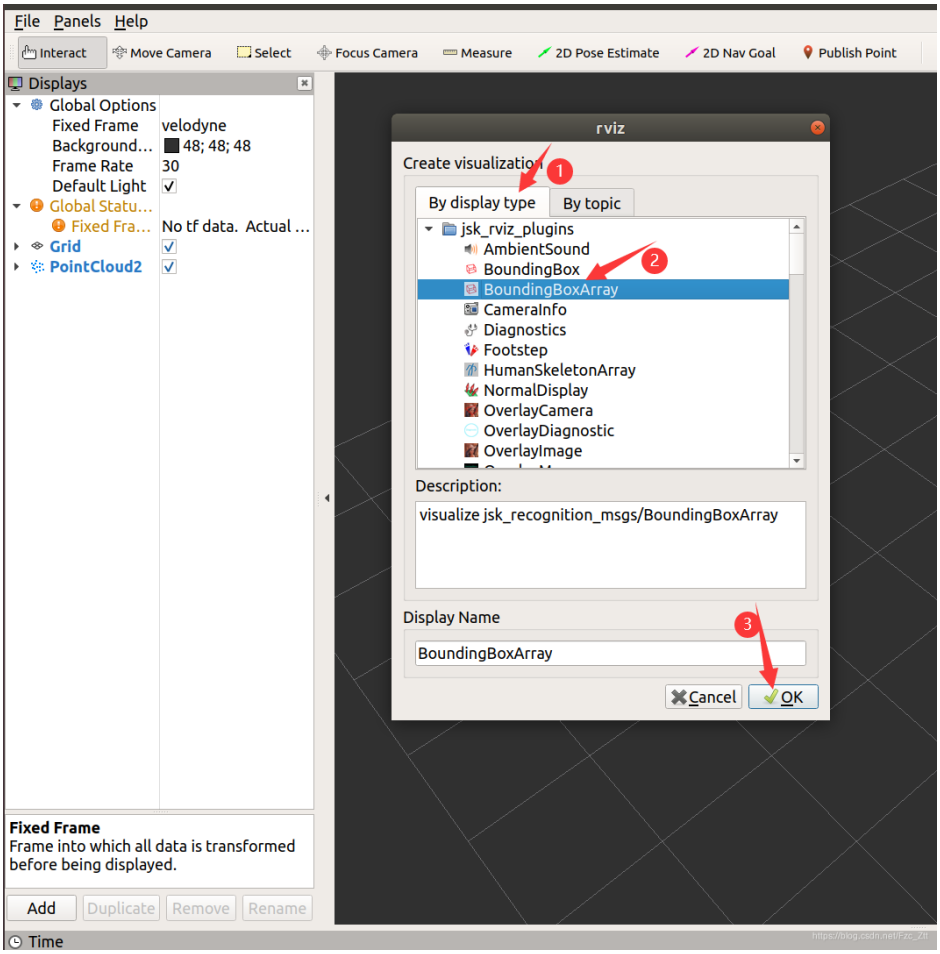


如果你的下拉框中没有/velodyne_points选项，那么肯定是你的第三组代码已经跑完了（如下图所示）



因为这组代码就是遍历并发布bag包中的点云，遍历完所有点程序也就结束了，rviz自然接收不到点云和话题了。重新运行第三组代码后（这个时候是不需要重启rviz的），在重新打开Add窗口，就可以在topic中找到了/velodyne_points了。

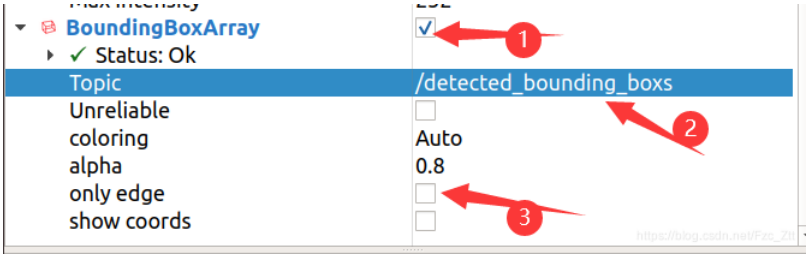
添加展示类型：BoundingBoxArray（这个时用来画目标框的，如果你的选项栏里没有这个选项，说明上边提到的3个包你没安装成功，如果时这种情况，你需要先回过头去安装这三个包，再重新启动rviz（也就是先关掉这个可视化界面，将第四组代码重新运行一下）



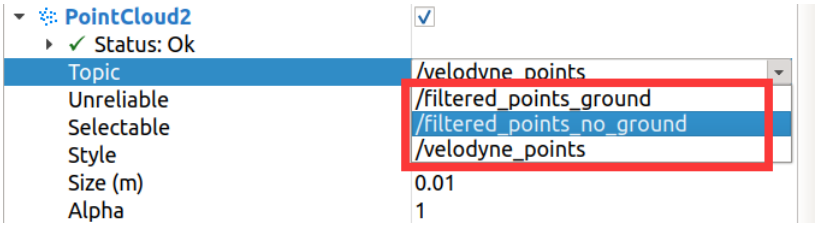
type和topic添加完成后，对相应的模块进行设置；

BoundingBoxArray的Topic手动输入 /detected_bounding_boxs ，或者启动新的命令行窗口运行 rostopic list 从话题列表中直接复制该话题（比手动输入快而且不如出错）

③处的选项为检测框的形式，勾选一下试一试就知道了,不细说了。

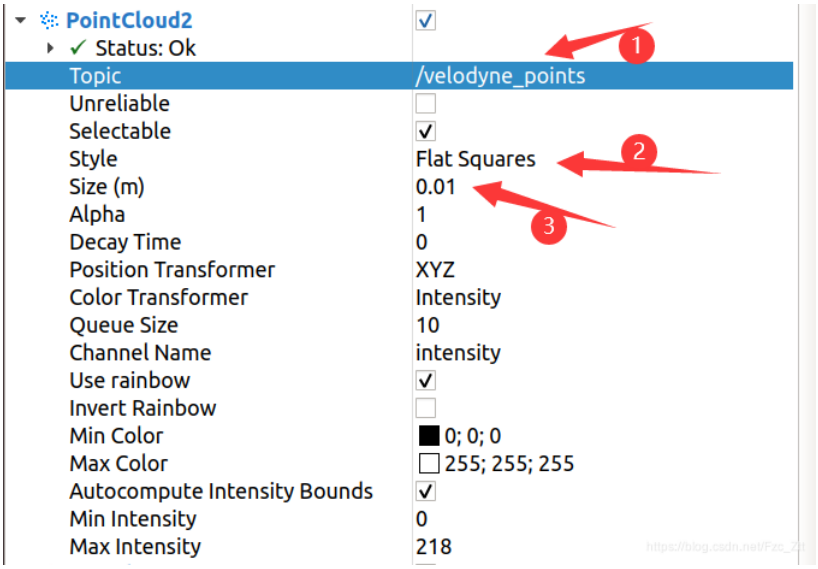


PointCloud2中的Topic下拉框中有三个选项，三个选项的意思也很易懂，不细说，自己尝试观察不同。如果你的下拉框中没有/velodyne_points选项，原因同上文中相同，第三组代码又又又跑完了，依然是重启启动第三组代码，再次点开PointCloud2中的Topic下拉框，就可以找到了。如下图：



尝试修改一些参数使可视化效果更佳：

Style:点云的风格
Size(m)：设置点云中点的大小


PointCloud2
Status: Ok
Topic
/velodyne_points
Unreliable
Selectable
Style
Flat Squares
Size (m)
0.01
Alpha
1
Decay Time
0
Position Transformer
XYZ
Color Transformer
Intensity
Queue Size
10
Channel Name
intensity
Use rainbow
Invert Rainbow
Min Color
0; 0; 0
Max Color
255; 255; 255
Autocompute Intensity Bounds
Min Intensity
0
Max Intensity
218
1
2
3