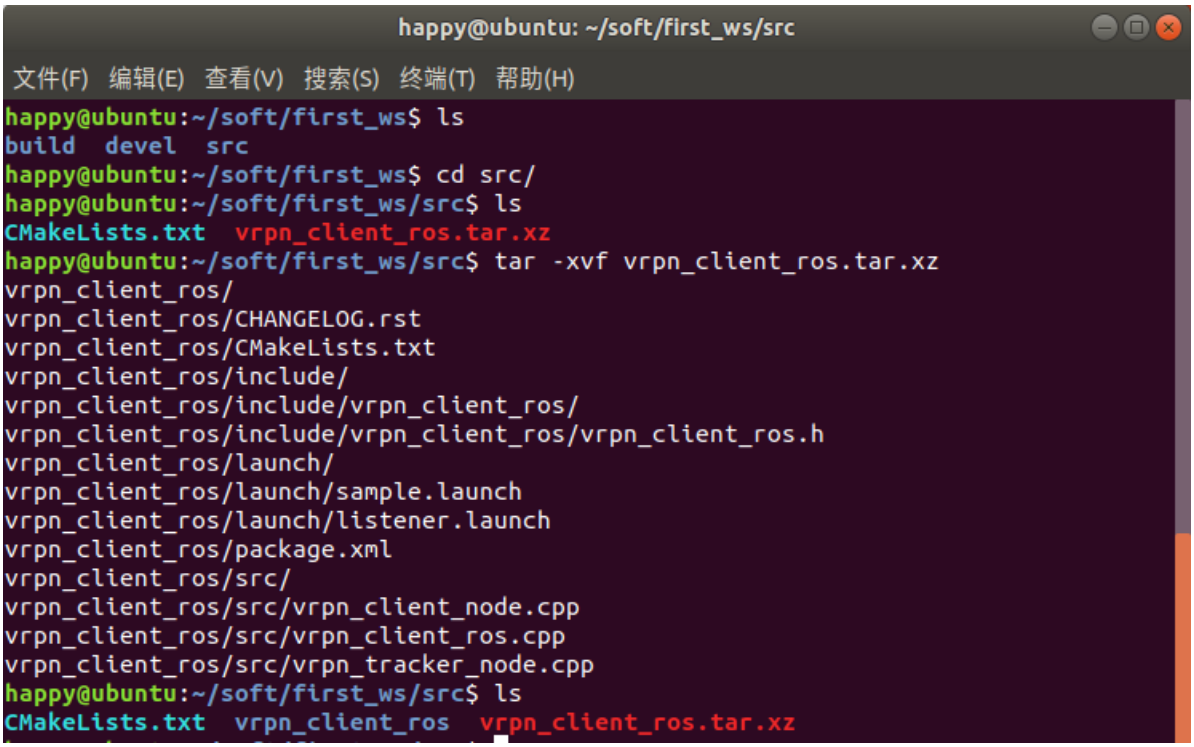
Ros环境下安装vrpn-client

环境要求

1.支持ROS Melodic Morenia以及ROS Kinetic Kame（测试环境是Melodic。Kinetic之前的版本可能不兼容）

安装步骤

1.拷贝文件至ros工作路径下解压



2.安装vrpn_client_ros

> kinetic下安装

```
sudo apt-get install ros-kinetic-vrpn-client-ros
```

> melodic下安装

```
sudo apt-get install ros-melodic-vrpn-client-ros
```

```
happy@ubuntu:~$ sudo apt-get install ros-melodic-vrpn-client-ros
[sudo] happy 的密码：
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
正在读取状态信息... 完成
下列软件包将被升级：
  ros-melodic-vrpn-client-ros
升级了 1 个软件包，新安装了 0 个软件包，要卸载 0 个软件包，有 191 个软件包未被升级。
需要下载 101 kB 的归档。
解压缩后会消耗 0 B 的额外空间。
获取:1 http://mirrors.ustc.edu.cn/ros/ubuntu bionic/main amd64 ros-melodic-vrpn-client-ros amd64 0.2.2-0bionic.20200930.060827 [101 kB]
已下载 101 kB，耗时 0秒 (808 kB/s)
(正在读取数据库 ... 系统当前共安装有 251081 个文件和目录。)
正准备解包 .../ros-melodic-vrpn-client-ros_0.2.2-0bionic.20200930.060827_amd64.deb  ...
正在将 ros-melodic-vrpn-client-ros (0.2.2-0bionic.20200930.060827) 解包到 (0.2.2-0bionic.20200821.180444) 上 ...
正在设置 ros-melodic-vrpn-client-ros (0.2.2-0bionic.20200930.060827) ...
```

3.返回工作空间编译

```
happy@ubuntu:~/soft/first_ws/src$ cd ..
happy@ubuntu:~/soft/first_ws$ pwd
/home/happy/soft/first_ws
happy@ubuntu:~/soft/first_ws$ catkin_make
```

```
happy@ubuntu: ~/soft/first_ws
文件(F)  编辑(E)  查看(V)  搜索(S)  终端(T)  帮助(H)
happy@ubuntu:~/soft/first_ws/src$ cd ..
happy@ubuntu:~/soft/first_ws$ pwd
/home/happy/soft/first_ws
happy@ubuntu:~/soft/first_ws$ catkin_make
Base path: /home/happy/soft/first_ws
Source space: /home/happy/soft/first_ws/src
Build space: /home/happy/soft/first_ws/build
Devel space: /home/happy/soft/first_ws/devel
Install space: /home/happy/soft/first_ws/install
####
#### Running command: "cmake /home/happy/soft/first_ws/src -DCATKIN_DEVEL_PREFIX=/home/happy/soft/first_ws/devel -DCMAKE_INSTALL_PREFIX=/home/happy/soft/first_ws/install -G Unix Makefiles" in "/home/happy/soft/first_ws/build"
####
-- Using CATKIN_DEVEL_PREFIX: /home/happy/soft/first_ws/devel
-- Using CMAKE_PREFIX_PATH: /opt/ros/melodic
-- This workspace overlays: /opt/ros/melodic
-- Found PythonInterp: /usr/bin/python2 (found suitable version "2.7.17", minimum required is "2")
-- Using PYTHON_EXECUTABLE: /usr/bin/python2
-- Using Debian Python package layout
-- Using empy: /usr/bin/empy
-- Using CATKIN_ENABLE_TESTING: ON
```

4.编译完成后，需要更新.bash

```
source devel/setup.bash
```

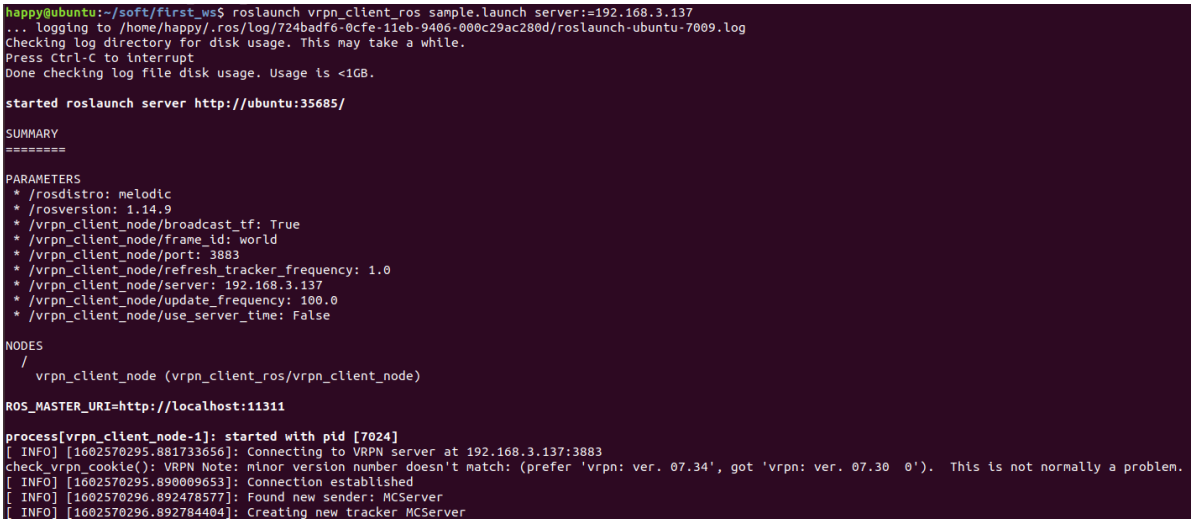
5.运行代码

> 运行程序程序前，请确认ros的工作路径。建议每次启动前在准备使用的工程目录下执行步骤4。

方式一

```
roslaunch vrpn_client_ros sample.launch server:=192.168.3.137
```

> roslaunch vrpn_client_ros [launch文件下的launch文件] server:=192.168.3.137
>
> 默认的sample.launch，运行后会自行检索对应的vrpn,成功匹配到以后会出现下图内容
>
> [INFO] [1602570295.890009653]: Connection established
> [INFO] [1602570296.892478577]: Found new sender: MCServer
> [INFO] [1602570296.892784404]: Creating new tracker MCServer



方式二

> 先修改`vrpn_client_ros/launch`文件,如下图

```
sensor_id:true//启用物体id标识
```

```
<launch>

  <arg name="server" default="localhost"/>

  <node pkg="vrpn_client_ros" type="vrpn_client_node" name="vrpn_client_node" output="screen">
    <rosparam subst_value="true">
      server: $(arg server)
      port: 3883

      update_frequency: 100.0
      frame_id: world

      # Use the VRPN server's time, or the client's ROS time.
      use_server_time: false
      broadcast_tf: true

      # Must either specify refresh frequency > 0.0, or a list of trackers to create
      refresh_tracker_frequency: 1.0
      #trackers:
      #- FirstTracker
      #- SecondTracker
    </rosparam>
  </node>

</launch>
```

替换为服务端的ip

可写可不写，如要修改，请修改为与服务端服务名一致：MCServer（可以通过服务端配置修改）

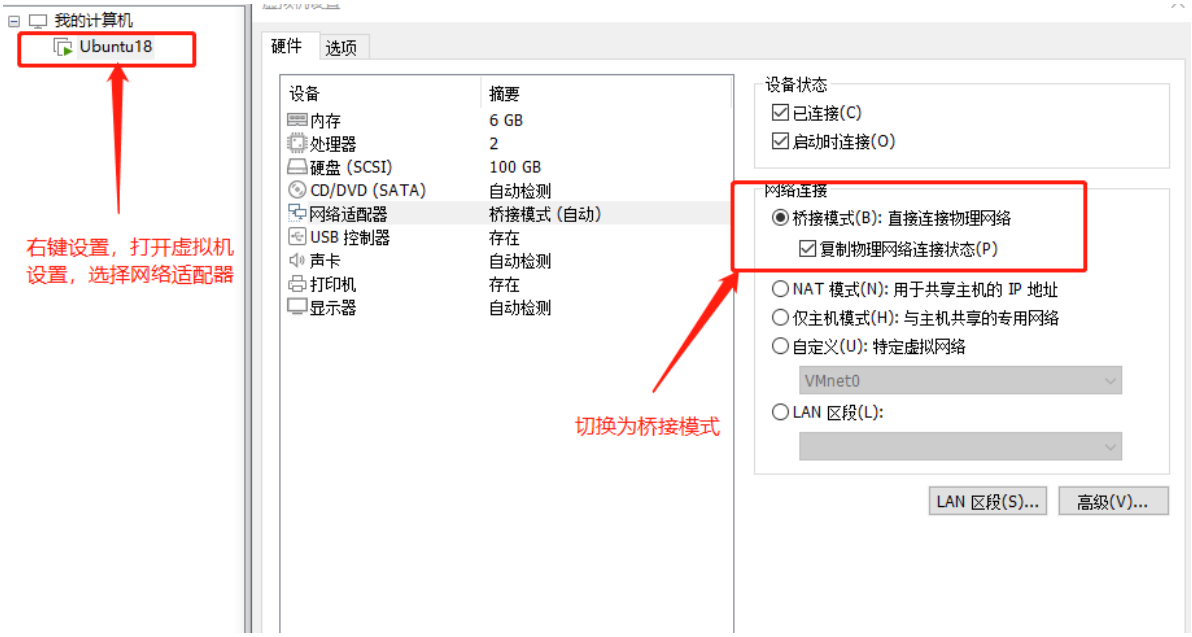
```
roslaunch vrpn_client_ros sample.launch
```

```
happy@ubuntu:~/soft/first_ws$ clear
happy@ubuntu:~/soft/first_ws$ source devel/setup.bash
happy@ubuntu:~/soft/first_ws$ roslaunch vrpn_client_ros listener.launch
... logging to /home/happy/.ros/log/467c1e4a-0d35-11eb-83e3-000c29ac280d/roslaunch-ubuntu-3011.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://ubuntu:42901/

SUMMARY
========

PARAMETERS
 * /rosdistro: melodic
 * /rosversion: 1.14.9
 * /vrpn_client_node/broadcast_tf: True
 * /vrpn_client_node/frame_id: world
 * /vrpn_client_node/port: 3883
 * /vrpn_client_node/refresh_tracker_frequency: 1.0
 * /vrpn_client_node/server: 192.168.3.137
 * /vrpn_client_node/update_frequency: 100.0
 * /vrpn_client_node/use_server_time: False

NODES
  /
    vrpn_client_node (vrpn_client_ros/vrpn_client_node)

ROS_MASTER_URI=http://localhost:11311

process[vrpn_client_node-1]: started with pid [3026]
[ INFO] [1602582953.770675685]: Connecting to VRPN server at 192.168.3.137:3883
[ INFO] [1602582953.777319967]: Connection established
check_vrpn_cookie(): VRPN Note: minor version number doesn't match: (prefer 'vrpn: ver. 07.34', got 'vrpn: ver. 07.30  0').  This is not
normally a problem.
[ INFO] [1602582954.781262153]: Found new sender: MCServer
[ INFO] [1602582954.781712719]: Creating new tracker MCServer
```

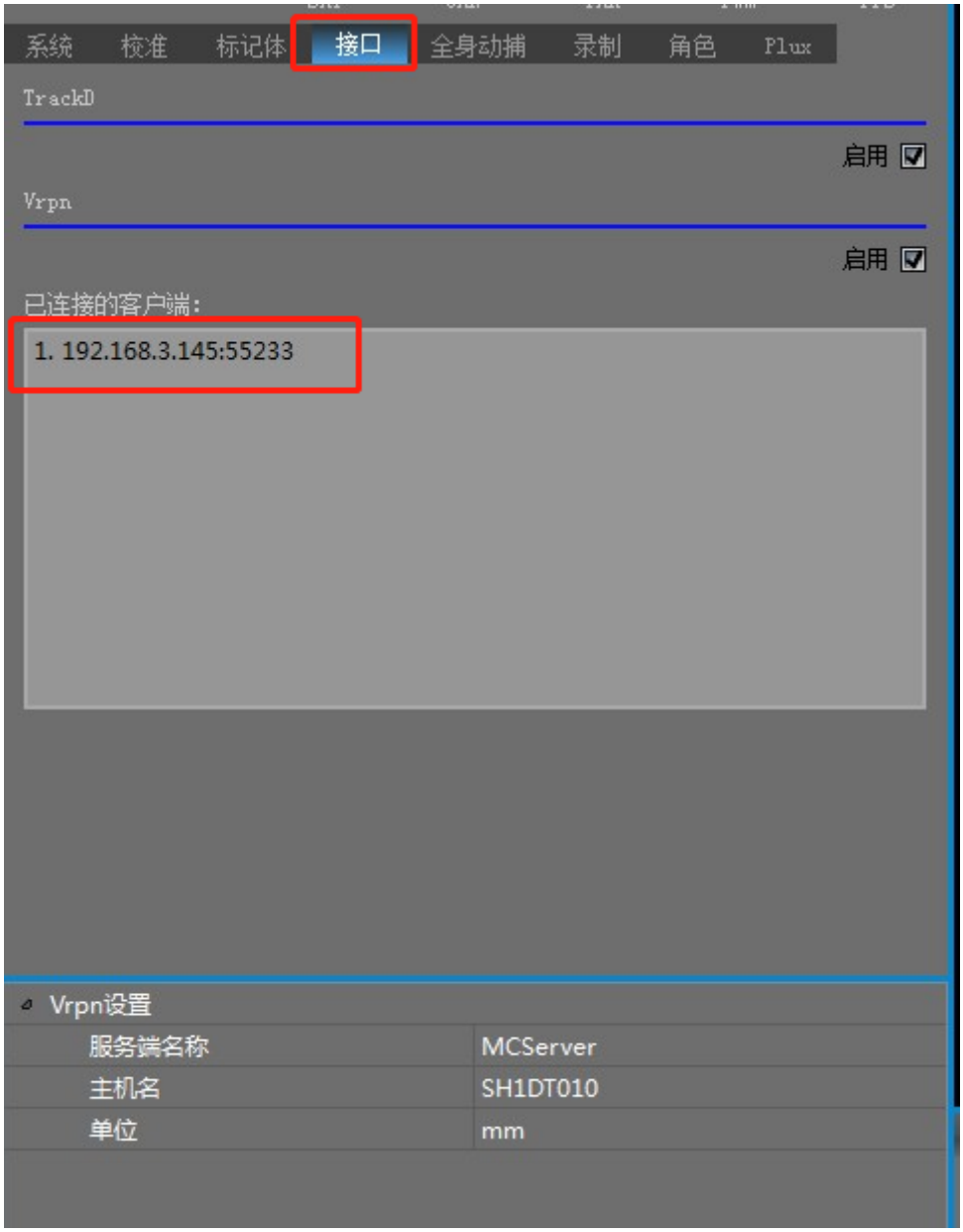
注意事项

> 如果是虚拟机下运行ros，可能会出现找不到发送的者的情况仅出现
>
> [INFO] [1602582346.683577053]: Connection established
>
> 而没有后面两条
>
> [INFO] [1602582347.703744913]: Found new sender: MCServer
> [INFO] [1602582347.704082242]: Creating new tracker MCServer

> 这种情况下，请将虚拟机的网络连接
>
> 方式切换为桥接，VMware如下

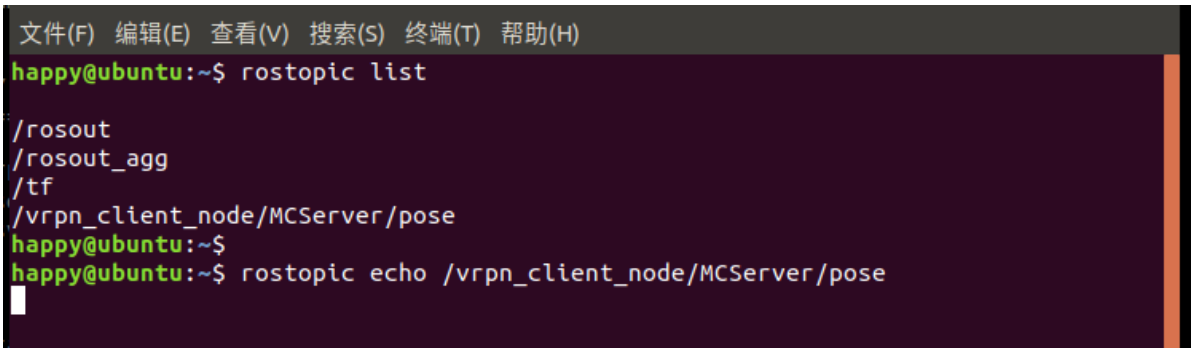


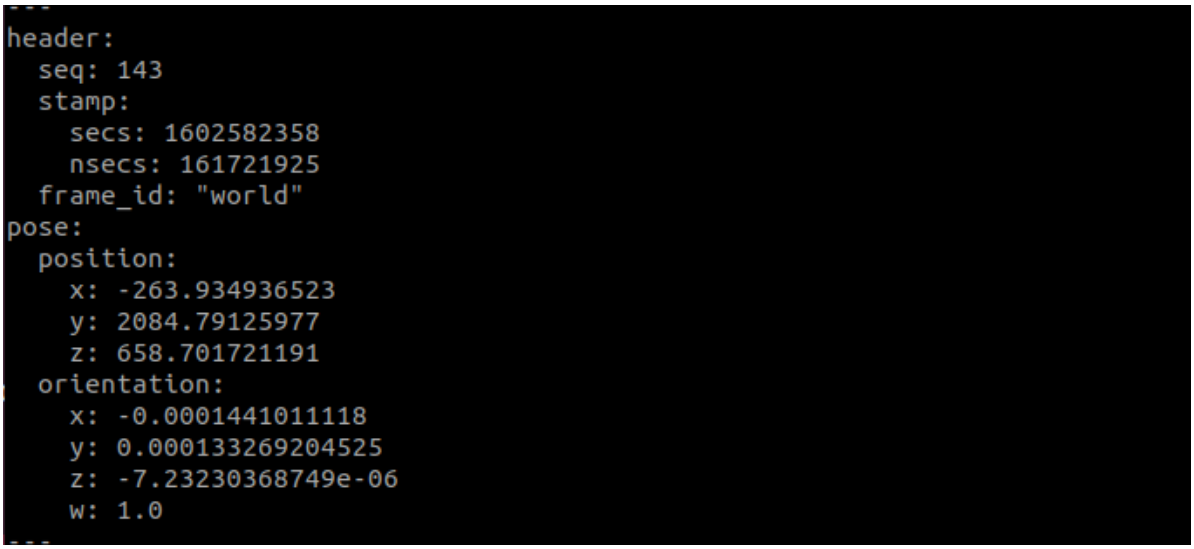
> 再次确认已经连接上服务端,在TrackerClient上确认已经和ros中的vrpnNode连接，如下图显示ros环境的ip地址以及端口号



6.查看接收到的数据

```
rostopic list
rostopic echo /vrpn_client_node/MCServer/pose
```





提示信息

> 可以用ros提供可视化工具，更方便的查看数据内容。

```
rosrun rqt_console rqt_console
rosrun rqt_logger_level rqt_logger_level
```

异常情况

1.没有数据信息

> 如图1.1情况，排除安装步骤5下的注意事项的原因，可能原因是服务端没有数据发送，请检查CMtracker clinet上是否已经添加标记体（刚体或者人物模型等）参考图1.2。

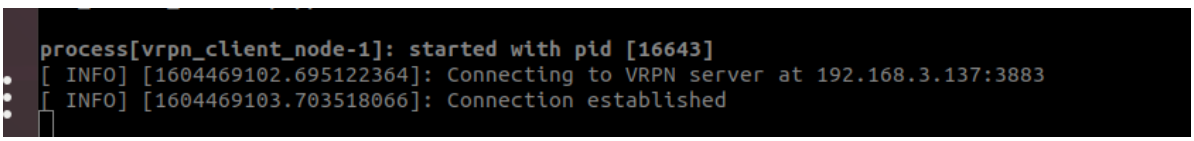


图1.1

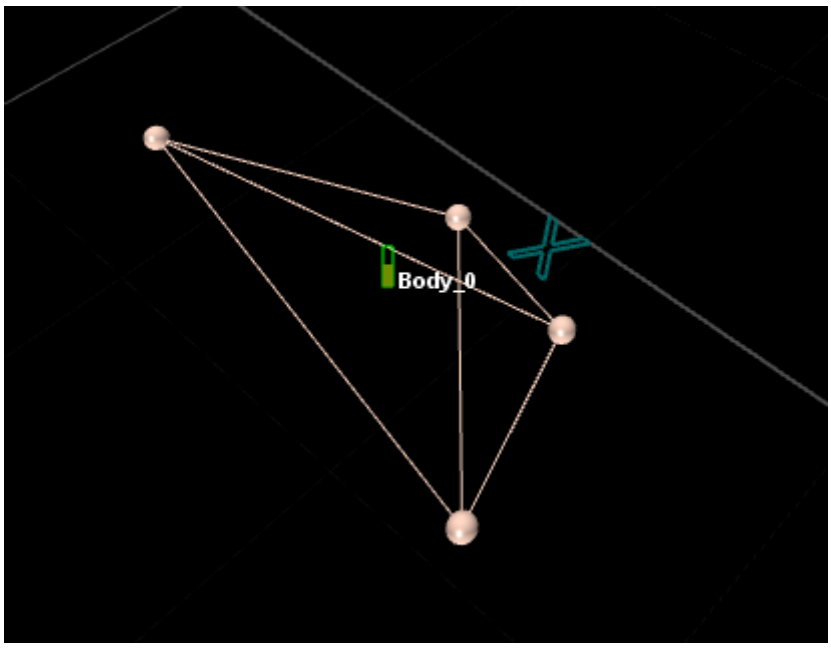


图1.2

示例代码

CPP

```cpp
#include "ros/ros.h"
#include <iostream>
#include "geometry_msgs/PoseStamped.h"

using namespace std;

void topicCallback(const geometry_msgs::PoseStamped & transform_stamped) {
    ROS_INFO_STREAM("Received a piece of data:" <<
transform_stamped.header);
    //ROS_INFO_STREAM("Received a piece of data:" <<
transform_stamped.pose);
}

int main(int argc, char **argv) {
    string nodeName = "cppsubscriber";
    string topicName = "/vrpn_client_node/MCServer/pose";

    //初始化节点
    ros::init(argc, argv, nodeName);
    ros::NodeHandle node;
    //创建订阅者
    const ros::Subscriber &subscriber = node.subscribe(topicName, 1000,
topicCallback);
    // 阻塞线程
    ros::spin();
    return 0;
}
```

联系方式

如果您安装的过程中遇到了其他疑问，欢迎致电青瞳视觉科技有限公司。

电话：021-56471565

技术支持：zhanghaidong@chingmu.com