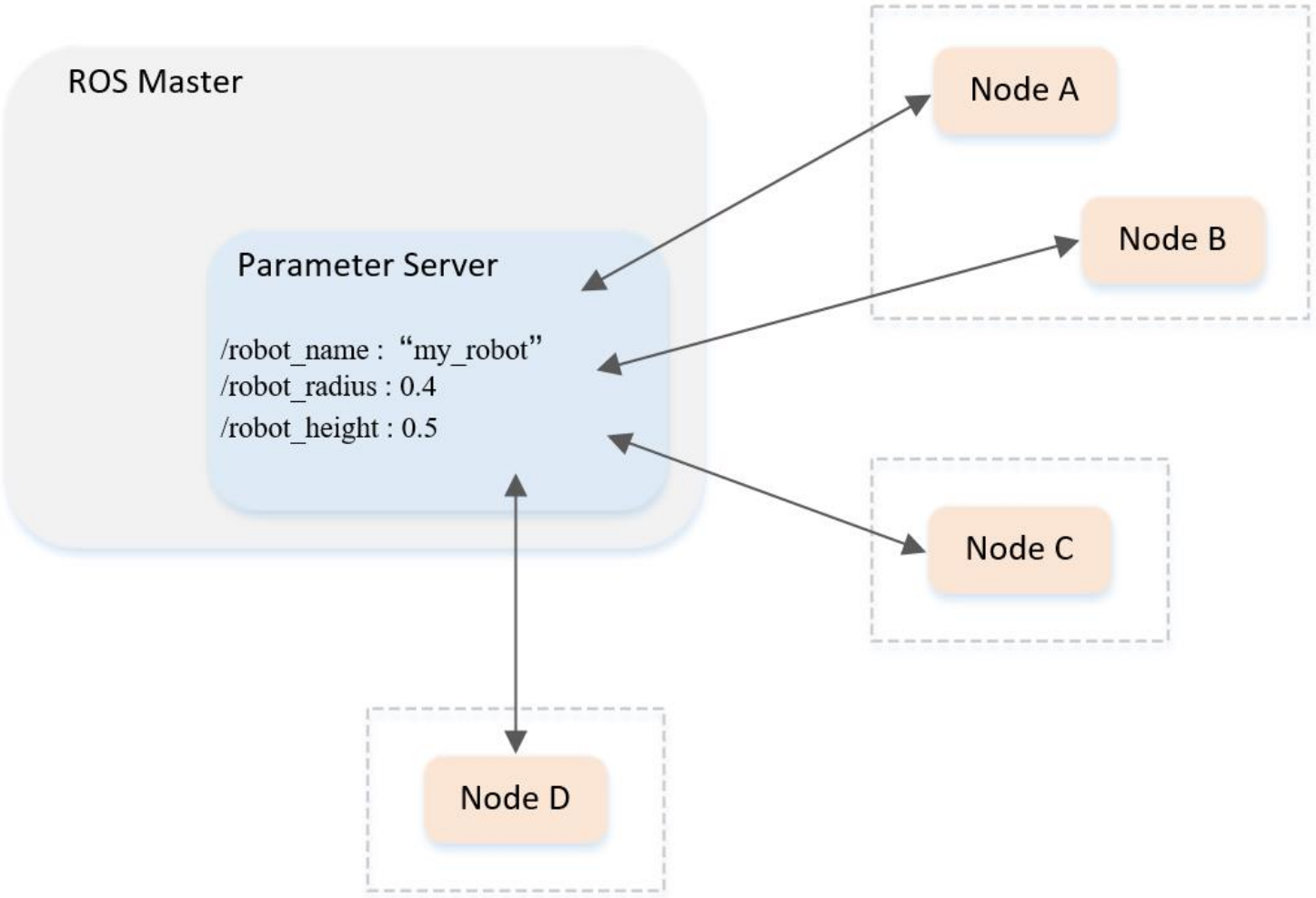
古月居 GYH.AI ROS

ROS入门
21讲

16.参数的使用与编程方法

主讲人：古月

参数模型

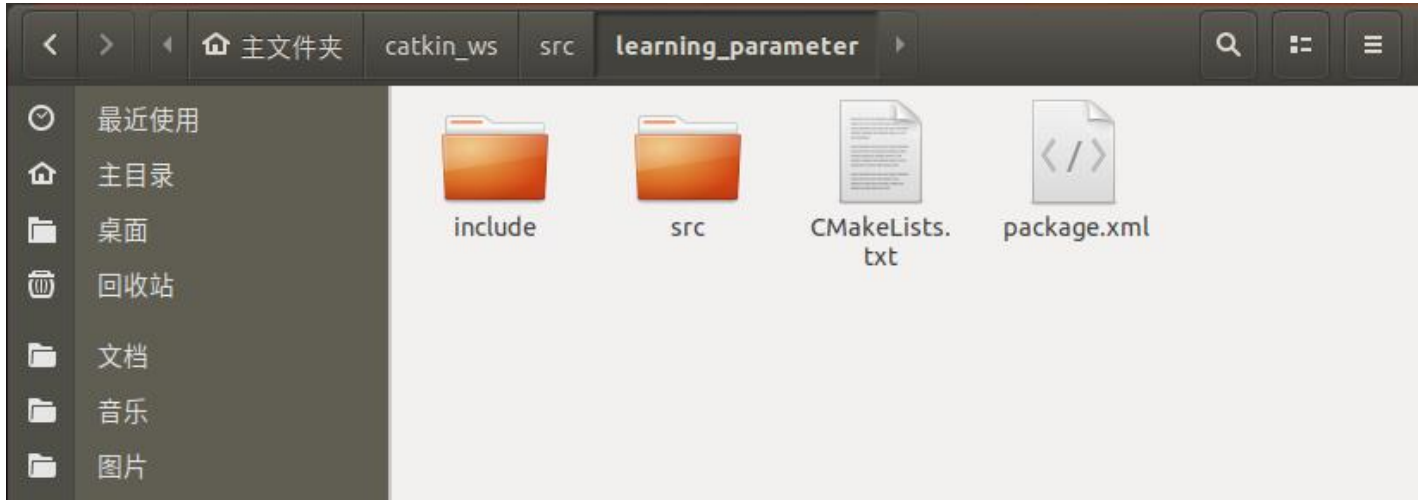


参数模型（全局字典）

*参考链接：http://wiki.ros.org/Parameter%20Server

创建功能包

```
$ cd ~/catkin_ws/src
$ catkin_create_pkg learning_parameter roscpp rospy std_srvs
```



参数命令行使用

YAML参数文件

```
background_b: 255
background_g: 86
background_r: 69
rosdistro: 'melodic'
roslaunch:
  uris: {host_hcx_vpc__43763: 'http://hcx-vpc:43763/'}
rosversion: '1.14.3'
run_id: 077058de-a38b-11e9-818b-000c29d22e4d
```

rosparam

- 列出当前多有参数

$ rosparam list

- 显示某个参数值

$ rosparam get *param_key*

- 设置某个参数值

$ rosparam set *param_key param_value*

- 保存参数到文件

$ rosparam dump *file_name*

- 从文件读取参数

$ rosparam load *file_name*

- 删除参数

$ rosparam delete *param_key*

编程方法（C++）

```cpp
/**
 * 该例程设置/读取海龟例程中的参数
 */
#include <string>
#include <ros/ros.h>
#include <std_srvs/Empty.h>

int main(int argc, char **argv)
{
    int red, green, blue;

    // ROS节点初始化
    ros::init(argc, argv, "parameter_config");

    // 创建节点句柄
    ros::NodeHandle node;

    // 读取背景颜色参数
    ros::param::get("/background_r", red);
    ros::param::get("/background_g", green);
    ros::param::get("/background_b", blue);

    ROS_INFO("Get Backgroud Color[%d, %d, %d]", red, green, blue);

    // 设置背景颜色参数
    ros::param::set("/background_r", 255);
    ros::param::set("/background_g", 255);
    ros::param::set("/background_b", 255);

    ROS_INFO("Set Backgroud Color[255, 255, 255]");

    // 读取背景颜色参数
    ros::param::get("/background_r", red);
    ros::param::get("/background_g", green);
    ros::param::get("/background_b", blue);

    ROS_INFO("Re-get Backgroud Color[%d, %d, %d]", red, green, blue);

    // 调用服务，刷新背景颜色
    ros::service::waitForService("/clear");
    ros::ServiceClient clear_background = node.serviceClient<std_srvs::Empty>("/clear");
    std_srvs::Empty srv;
    clear_background.call(srv);

    sleep(1);

    return 0;
}
```

parameter_config.cpp

如何获取/设置参数

- 初始化ROS节点；
- get函数获取参数；
- set函数设置参数；

配置代码编译规则

```
## Declare a C++ executable
## With catkin_make all packages are built within a single CMake context
## The recommended prefix ensures that target names across packages don't collide
# add_executable(${PROJECT_NAME}_node src/learning_parameter_node.cpp)

## Specify libraries to link a library or executable target against
# target_link_libraries(${PROJECT_NAME}_node
#   ${catkin_LIBRARIES}
# )

add_executable(parameter_config src/parameter_config.cpp)
target_link_libraries(parameter_config ${catkin_LIBRARIES})
```

如何配置CMakeLists.txt中的编译规则

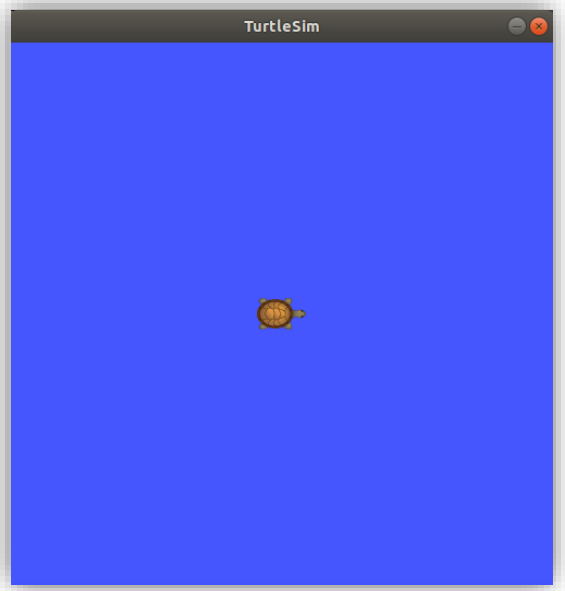
- 设置需要编译的代码和生成的可执行文件；
- 设置链接库；

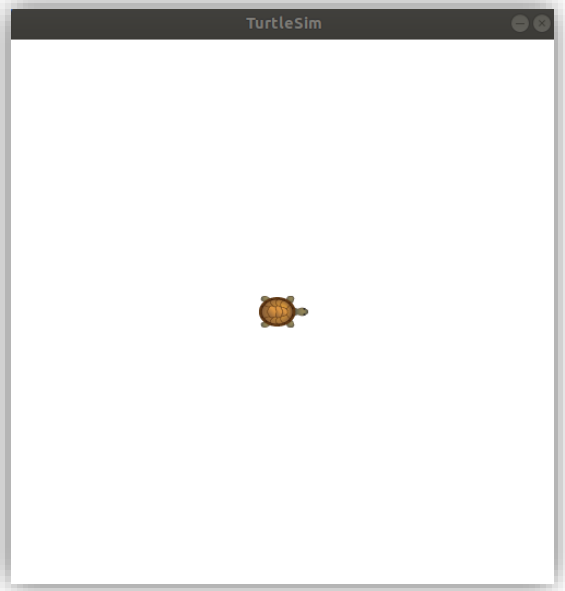
add_executable(parameter_config src/parameter_config.cpp)
target_link_libraries(parameter_config ${catkin_LIBRARIES})

CMakeLists.txt

编译并运行发布者

```
$ cd ~/catkin_ws
$ catkin_make
$ source devel/setup.bash
$ roscore
$ rosrun turtlesim turtlesim_node
$ rosrun learning_parameter parameter_config
```






```
hcx@hcx-vpc:~$ rosrun learning_parameter parameter_config
[ INFO] [1562816961.090709151]: Get Backgroud Color[69, 86, 255]
[ INFO] [1562816961.104068283]: Set Backgroud Color[255, 255, 255]
[ INFO] [1562816961.110197845]: Re-get Backgroud Color[255, 255, 255]
```

编程方法（Python）

```python
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# 该例程设置/读取海龟例程中的参数

import sys
import rospy
from std_srvs.srv import Empty

def parameter_config():
    # ROS节点初始化
    rospy.init_node('parameter_config', anonymous=True)

    # 读取背景颜色参数
    red   = rospy.get_param('/background_r')
    green = rospy.get_param('/background_g')
    blue  = rospy.get_param('/background_b')

    rospy.loginfo("Get Backgroud Color[%d, %d, %d]", red, green, blue)

    # 设置背景颜色参数
    rospy.set_param("/background_r", 255);
    rospy.set_param("/background_g", 255);
    rospy.set_param("/background_b", 255);

    rospy.loginfo("Set Backgroud Color[255, 255, 255]");

    # 读取背景颜色参数
    red   = rospy.get_param('/background_r')
    green = rospy.get_param('/background_g')
    blue  = rospy.get_param('/background_b')

    rospy.loginfo("Get Backgroud Color[%d, %d, %d]", red, green, blue)

    # 发现/spawn服务后，创建一个服务客户端，连接名为/spawn的service
    rospy.wait_for_service('/clear')
    try:
        clear_background = rospy.ServiceProxy('/clear', Empty)

        # 请求服务调用，输入请求数据
        response = clear_background()
        return response
    except rospy.ServiceException, e:
        print "Service call failed: %s"%e

if __name__ == "__main__":
    parameter_config()
```

parameter_config.py

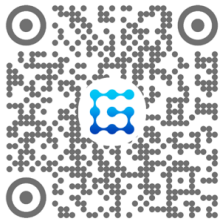
如何获取/设置参数

- 初始化ROS节点；
- get函数获取参数；
- set函数设置参数；

感谢观看

怕什么真理无穷，进一寸有一寸的欢喜

更多精彩，欢迎关注



古月居

古月学院

古月居GYH