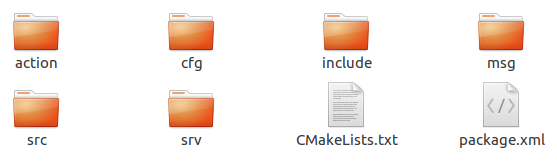
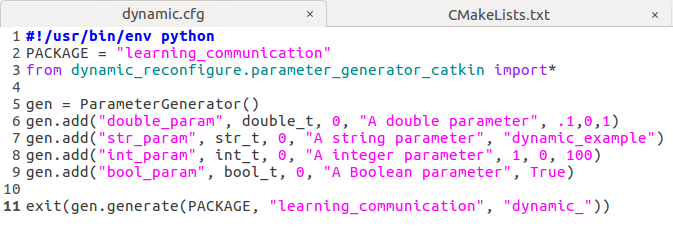
# ROS之动态参数编程

以下演示了动态参数的编程。

1. 在功能包下新建一个cfg文件夹：



新建一个dynamic.cfg文件，内容如下：

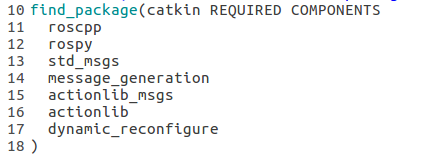


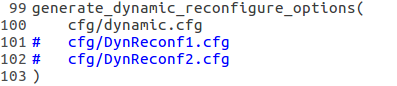
这里要注意的是最后一行代码第3个参数，编译生成的h文件，文件名称由第三个参数后加Config.h，也就是dynamic\_Config.h。

1. 在src目录下编写dynamic.cpp，该程序可视为动态参数的服务器，代码如下：

|  |
| --- |
| #include <ros/ros.h>  #include <dynamic\_reconfigure/server.h>  #include <learning\_communication/dynamic\_Config.h>  void callback**(**learning\_communication**::**dynamic\_Config **&**config**,** uint32\_t level**)**  **{**  ROS\_INFO**(**"Request: %d %f %s %s"**,** config**.**int\_param**,** config**.**double\_param**,** config**.**str\_param**.**c\_str**(),** config**.**bool\_param**?**"True"**:**"False"**);**  **}**  int main**(**int argc**,** char**\*\*** argv**)**  **{**  ros**::**init**(**argc**,** argv**,** "dynamic\_example"**);**  dynamic\_reconfigure**::**Server**<**learning\_communication**::**dynamic\_Config**>** server**;**  dynamic\_reconfigure**::**Server**<**learning\_communication**::**dynamic\_Config**>::**CallbackType f**;**  f **=** boost**::**bind**(&**callback**,** \_1**,** \_2**);**  server**.**setCallback**(**f**);**  ros**::**spin**();**  **return** 0**;**  **}** |

1. 在CMakeLists.txt文件中添加：

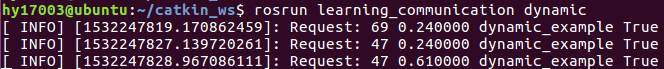






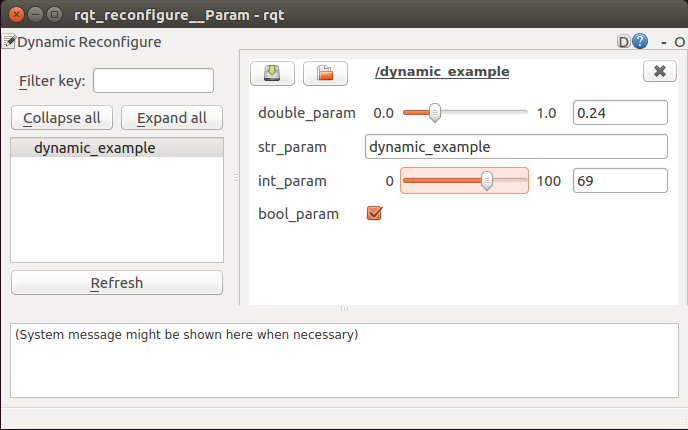
4．编译运行

打开一个终端：



打开另一个终端：





修改界面上的值，可见服务器端打印出了新的值。