编译 ROS 程序包

AI 航 团队

1. 编译程序包

一旦安装了所需的系统依赖项，我们就可以开始编译刚才创建的程序包了。

记得事先 source 你的环境配置(setup)文件，在 Ubuntu 中的操作指令如下：

```
$ source /opt/ros/groovy/setup.bash
```

1.1 使用 catkin_make

catkin_make 是一个命令行工具，它简化了 catkin 的标准工作流程。你可以认为 catkin_make 是在 CMake 标准工作流程中依次调用了 cmake 和 make。

使用方法:

```
# 在 catkin 工作空间下
$ catkin_make [make_targets] [-DCMAKE_VARIABLES=...]
```

CMake 标准工作流程主要可以分为以下几个步骤：

```
# 在一个 CMake 项目里
$ mkdir build
$ cd build
$ cmake ..
$ make
$ make install  # (可选)
```

每个 CMake 工程在编译时都会执行这个操作过程。相反，多个 catkin 项目可以放

```
# In a catkin workspace
$ catkin_make
$ catkin_make install  # (可选)
```

在工作空间中一起编译，工作流程如下：

上述命令会编译 src 文件夹下的所有 catkin 工程。想更深入了解请参考 REP128。

如果你的源代码不在默认工作空间中（~/catkin_ws/src),比如说存放在了 my_src 中，那

么你可以这样来使用 catkin_make:

```
beginner_tutorials/  CMakeLists.txt@
```

1.2 开始编译你的程序包

按照之前的创建一个 ROS 程序包教程，你应该已经创建好了一个 catkin 工作空间 和一个名为 beginner_tutorials 的 catkin 程序包。现在切换到 catkin workspace 并查看 src 文件夹：

```
$ cd ~/catkin_ws/
$ ls src
```

你可以看到一个名为 beginner_tutorials 的文件夹，这就是你在之前的 catkin_create_pkg 教程里创建的。现在我们可以使用 catkin_make 来编译它了:

```
$ catkin_make
```

```
yt@yt-UNO-2483G-453AE:~/catkin_ws$ catkin_make
Base path: /home/yt/catkin_ws
Source space: /home/yt/catkin_ws/src
Build space: /home/yt/catkin_ws/build
Devel space: /home/yt/catkin_ws/devel
Install space: /home/yt/catkin_ws/install
####
#### Running command: "cmake /home/yt/catkin_ws/src -DCATKIN_DEVEL_PREFIX=/home/
####
-- The C compiler identification is GNU 5.4.0
-- The CXX compiler identification is GNU 5.4.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Using CATKIN_DEVEL_PREFIX: /home/yt/catkin_ws/devel
-- Using CMAKE_PREFIX_PATH: /opt/ros/kinetic
-- This workspace overlays: /opt/ros/kinetic
-- Found PythonInterp: /usr/bin/python (found version "2.7.12")
-- Using PYTHON_EXECUTABLE: /usr/bin/python
-- Using Debian Python package layout
-- Using empy: /usr/bin/empy
-- Using CATKIN_ENABLE_TESTING: ON
-- Call enable_testing()
-- Using CATKIN_TEST_RESULTS_DIR: /home/yt/catkin_ws/build/test_results
-- Found gmock sources under '/usr/src/gmock': gmock will be built
-- Looking for pthread.h
-- Looking for pthread.h - found
-- Looking for pthread_create
-- Looking for pthread_create - not found
-- Looking for pthread_create in pthreads
-- Looking for pthread_create in pthreads - not found
-- Looking for pthread_create in pthread
-- Looking for pthread_create in pthread - found
-- Found Threads: TRUE
-- Found gtest sources under '/usr/src/gmock': gtests will be built
-- Using Python nosetests: /usr/bin/nosetests-2.7
-- catkin 0.7.14
-- BUILD_SHARED_LIBS is on
-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
-- ~~  traversing 3 packages in topological order:
-- ~~  - beginner_tutorials
-- ~~  - beginner_tutorials1
-- ~~  - newmsg
-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
-- +++ processing catkin package: 'beginner_tutorials'
-- ==> add_subdirectory(beginner_tutorials)
-- +++ processing catkin package: 'beginner_tutorials1'
-- ==> add_subdirectory(beginner_tutorials1)
-- +++ processing catkin package: 'newmsg'
-- ==> add_subdirectory(newmsg)
-- Configuring done
-- Generating done
```

catkin_make 首先输出它所使用到的每个空间所在的路径。更多关于空间的信息，请参考 REP128 和 catkin/workspaces。需要注意的是由于这些空间存在默认配置的原因，有几个文件夹已经在 catkin 工作空间自动生成了，使用 ls 查看:

```
$ ls
```