

# NAO的C++SDK的安装

## 系统准备

---

Ubuntu16.04，为什么用Ubuntu呢？因为要编译得到能够移植到nao中的分享库的话，只有Ubuntu可以做到（参考官网）

## 安装包准备

---

在官网下载好以下两个包，分别是交叉编译包，一个是nao环境

- ctc-linux64-atom-2.1.4.13
- naoqi-sdk-2.1.4.13-linux64

## 安装qiBuild

---

qiBuild使用python写的命令行工具。。。 （牛逼），但是只支持Python2.7。  
所以用强大的Pypi即可安装qibuild。

```
sudo pip2 install qibuild --user
```

然后把 `$HOME/.local/bin` 路径到你的 `$PATH` 环境变量中  
这一步可以这样做，修改`~/.bashrc`文件，一劳永逸。

```
echo "PATH=$PATH:$HOME/.local/bin" >> ~/.bashrc  
source ~/.bashrc
```

## CMake

---

如果没安装CMake的安装下CMake

```
sudo apt-get install cmake
```

输入以下命令看看cmake的版本是不是在 `2.8.3` 以上。

```
cmake --version
```

# 配置qiBuild

```
qibuild config --wizard
```

- 选择generator，就选Unix Makefiles即可
- 选择IDE，要是用过QtCreator就可以选它，不然就选None即可

## 安装SDK

在 `$HOME` 目录下新建文件夹

```
mkdir -p ~/nao  
qibuild init
```

将ctc-linux64-atom-2.1.4.13和naoqi-sdk-2.1.4.13-linux64解压(复制)至 `~/nao` 文件夹分别命名为 `naoqi_lib` , `naoqi_sdk`

## 编译例程

进入example文件夹下

```
cd ~/nao/naoqi_sdk/doc/dev/cpp/example
```

配置工具链

```
qitoolchain create mytoolchain ~/nao/naoqi_lib/toolchain.xml
```

这样你就会有一个名为mytoolchain的工具链了。

用qitoolchain生成一个工具链之后，你好要告诉qibuild，增加一个工具链，于是执行下面的命令。

```
qibuild add-config mytoolchain --toolchain mytoolchain
```

当然，你可以在上面那行命令后面加上 `--default` 选项，使该工具链编程默认的工具链。下面用新的工具链编译工程。

```
cd core/sayhelloworld  
qibuild configure -c mytoolchain  
qibuild make -c mytoolchain
```

这样，build-mytoolchain/sdk/bin 下就会出现一个可执行文件

至此，NAO的C++SDK安装完成

## TIPS1

---

关于工具链，这个的意思是指，编译成二进制文件的过程中，编译器需要链接一些库，这些库是与系统、环境有关的，因为我们是写机器人程序，所以我们会有两个系统环境：1、自己的环境。2、nao上运行的环境。

而ctc这个文件夹里实际上就是nao里面的各种安装好的依赖库，要是用ctc里的库去编译程序的话，可以直接拷贝到nao上去运行，但是可想而知在自己的电脑上运行不了的，因为自己电脑上并没有对应的依赖库甚至连系统都完全不一样。

要想让程序在自己的电脑上能够运行，必须将编译所需要的依赖库设为自己电脑的，即我们用 `~/nao/naoqi_sdk/toolchain.xml` (之前是用naoqi\_lib里的toolchain.xml生成的)重新create一个工具链，之后编译程序的时候用这个工具链去编译，即可使得程序能够在PC上运行。

## TIPS2

---

现在大家常用的C++标准是C++11和C++14，可惜qibuild最高只支持C++0x :< 于是我按以往经验，在CMakeLists.txt中加入

```
add_definitions(-std=c++0x)
```

写了个简单的例程，一编译，一看能使用auto关键字，大喜 :)

但是一旦例程中包含了NAO的C++sdk的头文件，程序就报错，因为NAO的C++sdk不支持新的C++标准， :<

遂苦恼，一些很有用的STL库比如 `<thread>` , `<unordered_map>` , `<mutex>` , `<tuple>` 用不了了，还有最爱的 `auto` 和范围遍历也用不了 :<

还好nao的交叉编译库包含了大名鼎鼎的 `Boost` 库，以上除了范围遍历用不了之外，其余都能用 `Boost` 库实现!!!