



# 一键将 C/C++ 代码转换为 Python 能调用的代码

韦泽华



## 大家以前都是怎样调用C++代码呢?

1. Cython
2. boost.python/pybind11
3. ctypes



# 先不讨论这些工具的优劣





## 先给大家介绍一下

## 如何使用c2py来调用C++代码

## 在此之前给大家看一下要调用的C++代码源文件

- 我肯定还是会拿上期所的API来作示范：

(看一眼就好了)

Name	Size
 ThostFtdcMdApi.h	6 KB
 ThostFtdcTraderApi.h	36 KB
 ThostFtdcUserApiDataType.h	251 KB
 ThostFtdcUserApiStruct.h	232 KB

# c2py的使用分两步走

- 第一步，生成.pyd (.so) 文件

```
1  c2py generate vnctp \
2      ThostFtdcMdApi.h \
3      ThostFtdcTraderApi.h \
4      ThostFtdcUserApiDataType.h \
5      ThostFtdcUserApiStruct.h \
6      -I vnctp/include/ \
7      --no-callback-pattern ".*Api::.*" \
8      --string-encoding-windows .936 \
9      --string-encoding-linux zh_CN.GB18030 \
10 \
11      --generate-setup . \
12      --setup-lib-dir vnctp/libs/ \
13      --setup-lib thostmduserapi \
14      --setup-lib thosttraderapi \
15      --setup-use-patches \
16 \
17  python ./setup.py build
```

- 第二步，调用代码，并享受完备的type hint。

```
import vnctp

# code with type hint!
vnctp.CT
    CThostFtdcTraderApi vnctp
    CThostFtdcAccountPropertyField vnctp
    CThostFtdcAccountregisterField vnctp
    CThostFtdcAuthenticationInfoField vnctp
    CThostFtdcBatchOrderActionField vnctp
    CThostFtdcBrokerDepositField vnctp
    CThostFtdcBrokerField vnctp
```



**与其他工具的法进行一次比较？**

# 与pybind11进行比较

- c2py: 两个命令

```
1  c2py generate vnctp \
2      ThostFtdcMdApi.h \
3      ThostFtdcTraderApi.h \
4      ThostFtdcUserApiDataType.h \
5      ThostFtdcUserApiStruct.h \
6      -I vnctp/include/ \
7      --no-callback-pattern ".*Api::.*" \
8      --string-encoding-windows .936 \
9      --string-encoding-linux zh_CN.GB18030 \
10 \
11      --generate-setup \
12      --setup-lib-dir vnctp/libs/ \
13      --setup-lib thostmduserapi \
14      --setup-lib thosttraderapi \
15      --setup-use-patches \
16 \
17  python ./setup.py build
```

- pybind11: 1137行代码?

```
1137     .def("onRtnOpenAccountByBank", &T
1137     .def("onRtnCancelAccountByBank",
1137     .def("onRtnChangeAccountByBank",
1137     ;
1137 }
1137
```

- 不, 其实是11,376行代码:

```
11373
11374
11375
11376 }
```



跟pybind11比算个什么事？根本  
不在一个次元！

要比就和不需要写C++代码的工具  
比！



## 与ctypes进行比较

- c2py:调用代码，并享受type hint。

```
import vnctp
```

```
# code with type hint!
```

```
vnctp.CT|
```

CThostFtdcTraderApi	vnctp
CThostFtdcAccountPropertyField	vnctp
CThostFtdcAccountregisterField	vnctp
CThostFtdcAuthenticationInfoField	vnctp
CThostFtdcBatchOrderActionField	vnctp
CThostFtdcBrokerDepositField	vnctp
CThostFtdcBrokerField	vnctp

- ctypes:调用代码？？

```
import ctypes
```

```
dll = ctypes.WinDLL("thosttraderapi.dll")
```

```
fn = "?CreateFtdcTraderApi@CThostFtdcTraderApi@@SAPEAV1@PEBD@Z"
```

```
create = dll.__getattr__(fn)
```

```
obj = create(...)
```

如果把调用C++代码当作吃饭，那么：

工具	科技等级
手撸Cython	手撕猎物；钻木取火
Boost.python/pybind11	用猎枪打猎；在现代化的厨房里做菜
ctypes	在某个便利店里买吃的。（只卖番茄炒蛋）
c2py	外卖app



所以

调用C++，认准

现代化的外卖app: c2py

<https://github.com/nanoric/c2py>



# THANK YOU

微信号:  
nanoric

18807734862

二维码

