2020年7月6日星期一

老师，您好。第3集我实在有点儿没看懂，虽然第8集有补充，但是和第3集开头还是有点儿接不上，麻烦您再讲解一下吧：

1. 第3集一出来就是左右两侧各打开了一个项目，左侧是要跟着一起写的，右侧是示例，但之前的操作是怎样的不清楚。根据第1集的教学内容，我自己想象了一下之前的操作，您看是不是这样：1）在visual studio里面建一个c++动态链接的新项目，可以命名为vnctp，目录放在c盘根目录下；2）在vnctp目录下新建include目录，将pybind11和ctp头文件夹放入其中；3）在vnctp目录下新建libs目录，将ctp静态链接库放入其中；把generator文件夹也拷进来；4）在visual中按照第1集的内容对vnctp项目属性就行配置。

答：这里不需要把generator文件夹拷贝进来。generator是基于pyscript规则，通过对include里面头文件的解析，生成一系列h和cpp文件，用来复制粘贴到如vnctpmd.h和vnctpmd.cpp的，然后debug一下，可以编译到pyd文件。

2. 第3集一出来就是说要删掉virtual和大括号啥的，但是上面的内容是怎样的，也没有看到，麻烦您再具体讲讲，那个位置到底是要干什么？

答：这个问题，应该是在如vnctpmd.h头文件里面，去掉{}，应该是在回调函数上，我们要异步处理，把C++结构体的数据，放入队列中，然后解析到python的dict。

3. 如果前面是按照上述操作进行的，如何把vnctpmd和vnctptd放在同一个项目中？

答：先在vs2017中创建一个【空项目】，右键解决方案，点击添加，新建dll，如vnctpmd，vnctptd，这样，就实现2个dll项目都放在同一个解决方案上了。

4. 另外，请问一下vnctp.h是自动生成的，vnctpmd.h和vnctpmd.cpp是需要我们来写的，是吗？

答：vnctp.h几乎是通用模板，只要修改一下函数，如getInt -》getInt64，和task结构体，可以兼容其他接口，如xtp，tap，oes等等。vnctpmd.h和vnctpmd.cpp的内容，是基于generator的文件，我们一个个复制粘贴进去。当然 某些复杂的函数，还得人工手写，如createapi，sendorder之类的，然后 debug一下啊，可以点击【生成】来生成pyd文件，pyd类似python可读的dll。

就是说vnctptd.h、vnctpmd.h、vnctpmd.cpp这三个文件其实都是需要我们手动新建的文件，只是vnctp.h这个文件是通用模板，只需要复制过来即可，vnctpmd.h、vnctpmd.cpp的内容是通过从generator解析出来的文件中复制得来的，然后加上少量人工手写的函数。

5. vnctp.h、vnctpmd.h、vnctpmd.cpp这三个文件的逻辑关系是怎样的，就是哪些函数要写进这三个函数中的哪一个是怎么来分的？

答：. h是头文件，定义类和函数长什么样子。. cpp是源码文件，里面写具体的函数代码逻辑。几乎每个要封装的函数，都要同时进入到后面两个文件里。vnctp. h是定义了一些特殊的函数，不用改其中的东西。

vnctp.h是模板，vnctpmd.h要继承vnctp.h，调用里面某些函数。vnctpmd.h定义了主动函数（如reqUserLgin）和回调函数（如OnRspUserlogin）。这些函数 直接被python的ctpgateway调用，但是 vnctpmd.h的函数如何实现，其具体实现方法在vnctpmd.cpp里面找到。

2020年7月7日星期二

老师，您好。我的电脑是64位的，是不是vnctp.h中的void getInt(const dict &d, const char \*key, int \*value)，要改成void getInt64？vnctpmd.h中的//系统 #ifdef WIN32也要改成WIN64?

答：都不用

2020年7月8日星期三

老师，请问vnctpmd.h中以下三个主动函数都不要是吧？

virtual void RegisterNameServer(char\* pszNsAddress) = 0;

virtual void RegisterFensUserInfo(CThostFtdcFensUserInfoField\* pFensUserInfo) = 0;

virtual void RegisterSpi(CThostFtdcMdSpi\* pSpi) = 0;

答：是的。

2020年7月9日星期三

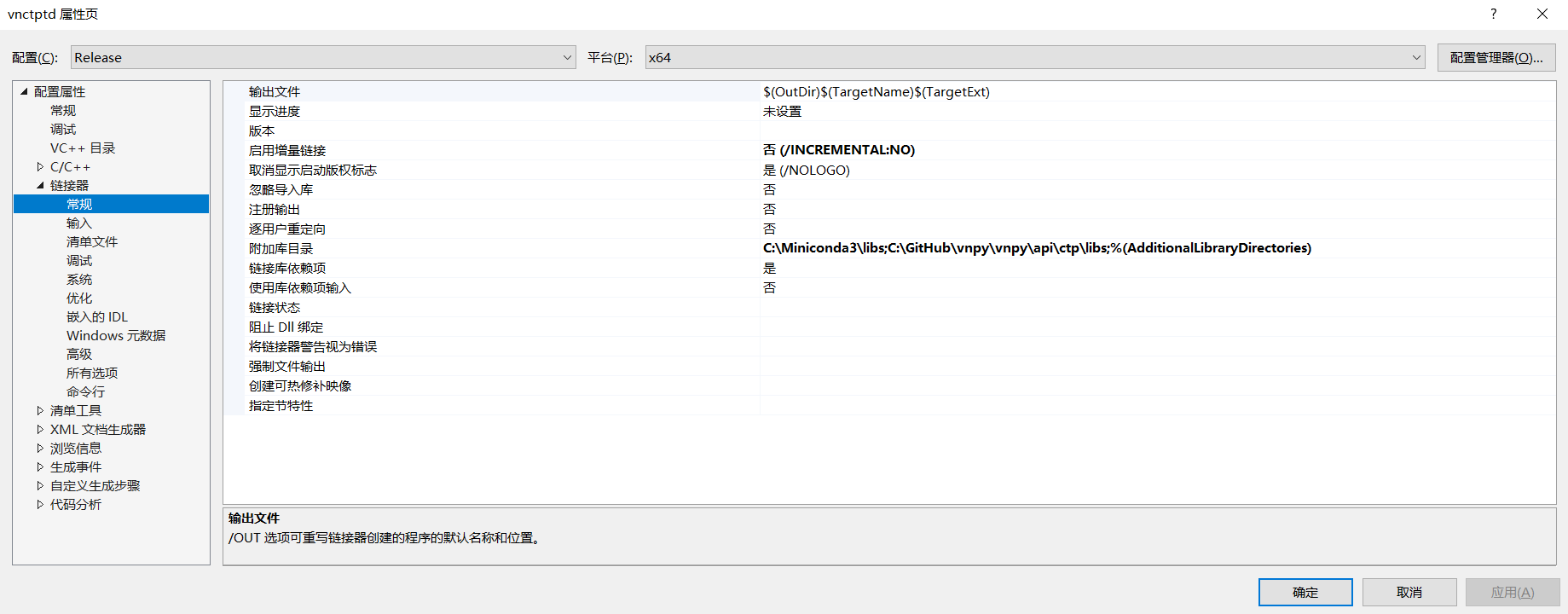
老师，您好。我写好了vnctpmd，生成的时候报错：

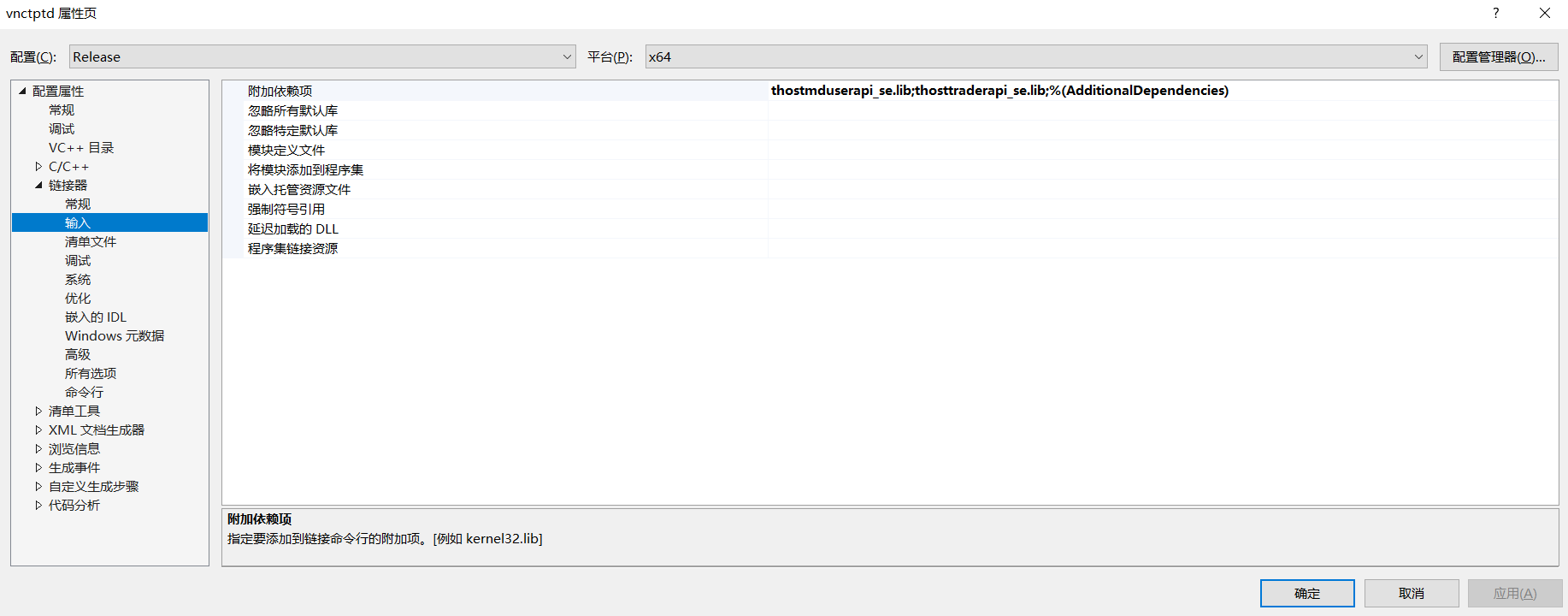
1>vnctpmd.obj : error LNK2001: 无法解析的外部符号 "public: static class CThostFtdcMdApi \* \_\_cdecl CThostFtdcMdApi::CreateFtdcMdApi(char const \*,bool,bool)" (?CreateFtdcMdApi@CThostFtdcMdApi@@SAPEAV1@PEBD\_N1@Z)

1>C:\ctp-HN\vnctp\x64\Release\vnctpmd.dll : fatal error LNK1120: 1 个无法解析的外部命令

麻烦您看看这个应该怎么办？

答：应该是少配置了目录，看看连接器的配置目录





2020年8月10日星期一

老师，您好。麻烦您请教两个比较小白的问题：

1. 根据您的经验，在vnpy框架下，不对其中任何部分进行C++封装改造，vnpy能支持的最高交易频率是多少？以tick数据进行交易是可行的吗？

答：vn.py开源版本的话，一秒下个200-500单没问题，取决于你的CPU主频和网络。

2. 您开发过的纯python交易策略中，是不是有能每天持续盈利的策略，是不是能持续盈利的策略大多是以高频为基础的？或者说，能高效盈利的策略都是以高频为基础的？

答：每天盈利的说实话没有，但是每个月盈利还是可以做到的，CTA类策略的组合，不同品种、周期、信号。特别高频的策略，对个人来说没意义，TOWER/JUMP这些公司，一年中国分部的IT预算就是2000万以上，设备、人员、网络、通道。

国外的数字货币市场、期货市场没有开平的概念，因此交易命令只会用到buy和sell吗？

实盘交易中，可能出现希望平掉的仓位没有平掉，但同时反向开仓又成交了，这时整体仓位是不对的，该如何处理？

2020年10月8日星期四

陈老师，不好意思，打搅一下。感觉您最近讲的复杂交易算法课程非常有用，但是关于self.pos值到底如何在策略运行过程中更新,还是没有太弄清楚，所以想请教一下：

（1）这次课程中有两次断掉了，部分内容没有听到，所以想确认一下，课程的关键内容就是讲清楚CTP先推送on\_order，后推送on\_trade，而self.pos是只有在收到on\_trade之后才会更新，在断掉的内容里没有讲到其他关键性内容了吧？

答：没啥了。

（2）如果只有在收到on\_trade推送之后，self.pos才会更新，那么on\_tick、on\_bar等其他回调函数下的self.pos是不是都不准确？

答：准确来说，要满足两个条件才是准确的：1）外部没有活动委托；2）在on\_trade下收到最新的成交回报。但对于分钟级别的CTA策略来说，这在99.9%情况下就是on\_bar中获取self.pos时的状态。

（3）self.pos到底是怎么计算的？是来自class PositionHolding的update\_trade函数吗？我自己找了半天没有看到self.pos的计算逻辑在哪。

答：在CtaEngine的process\_trade\_event中可以看到，位于vnpy/app/cta\_strategy/engine.py中，基于成交回报来计算的。PositionHolding是vn.py引擎层组件中的账户级持仓，和这里应用层的策略持仓无关。

2020年10月20日星期二

陈老师，打搅一下。我还是想请教一下关于rqdata数据是如何进入到回测引擎的，自己研究了很多天还是没怎么搞明白。我倒推数据来源的路径是这样的：

1）BacktestingEngine 中有个def load\_bar\_data；

2）然后跳到def load\_bar\_data 中有个return database\_manager.load\_bar\_data；

3）然后跳到database\_manager: "BaseDatabaseManager" = init(settings=settings)；

4）然后跳到def init(settings: dict) -> BaseDatabaseManager

接下来，我就不知道如何继续了，因为我没有学习过关于数据库的命令。所以麻烦请教一下：

1. 前面列的寻找rqdata如何导入回测引擎的倒推路径对吗？

2. 如果是对的，后面的关键点到底在哪，能否指点一下？

3. 为了彻底弄懂这个问题，我是不是应该去学习一下关于数据库的知识？单纯就我问的这个问题而言，是不是应该从sqlite开始？

非常感谢！

1.实盘CTP不显示行情数据

2.price\_tick 和 一般的超价买卖的区别