Git Repo

代码托管于GitHub上：[git@github.com:cangyun212/stock.git](mailto:git@github.com:cangyun212/stock.git)

继续开发请自行fork. BCGPControlBar 是一个收费的第三方mfc控件库，由于没有授权所以并没有放到repo内，而是本地自留一份拷贝。

Architecture

UI: Views and Dialogs

BCGPControlBar

MFC

Control: Trade Control

Domain: Trade Model Trade Order

Class

CStockApp: 负责应用程序初始化，持有CStockAppData

CStockAppData: 保存应用程序配置文件信息，持有CStockLocateData & CTradeSettingsData

CStockLocateData: 保存目标程序窗口控件位置及其他信息，并在程序加载时获取窗口句柄

CTradeSettingsData: 保存交易设置信息

CStockLocatePropPage: 属性对话框，用于设置或更新CStockLocateData数据

CStockTradeSettingsPropPage: 属性对话框，用于设置或更新CTradeSettingsData数据

CStockMainFrame: 主窗口，用于创建，销毁及管理CTradeView，负责接收ST\_MODEL\_UPDATE\_MSG并通知CTradeControl刷新相关view

CTradeWnd: 交易窗口，负责显示数据和响应用户操作

CTradeView: CTradeWnd父窗口，负责窗口停靠及响应主要UI逻辑

CTradeControl: UI 和 Domain之间的中间层，负责分发UI请求，通知数据更新及刷新UI显示

CTradeModel/CTradeModelManager: 持有股票交易信息，负责与wind交互并刷新交易数据

CTradeOrder/CTraderOrderManager: 持有订单信息，负责下单撤单

Utils: ApiHelper.h/Utils.h/TradeTime.h包含所有的工具类和函数

StockConfig.h/StockGlobal.h: 包含全局类型，宏的配置和定义

Known Issues & TODO

Loading 界面右下角状态指示标签在显示不同信息时会有重叠现象，应与控件背景被设为透明有关。

CTradeView 在关闭时会有四种不同状态，并以四种不同形式通知main frame, 目前3中状态下关闭都已被正确处理，但是当view 处于 floating 状态并与其他view tab在一起组成multi mini frame window时窗口没有被正确关闭, 请在CStockMainFrame::OnCloseMiniFrame 函数加断点跟踪调试。这个问题不会影响程序功能，但是阻止无用trade model的释放，程序仍会刷新这些数据，影响效率。

CTradeView在Dock状态时被关闭，所在区域没有重绘。

CTradeModel中仍然包含订阅模式获取wind数据的代码，如果订阅模式不可用请去除这些代码, 如需继续使用，需要自行添加thread safe代码

程序中一些设置被hard code为全局配置，如果需要可以将它们移到CTradeSetting中，交由用户自行配置。