

股票交易系统

股票中央交易子系统

The Central Exchange Subsystem of the Stock Trading System

需求说明书 Requirement Specification



作者 陈玮烨 孙克染 张梓欣 杨清杰

2019 年 3 月 22 日

Table of Contents 目录

- 引言..... 1
 - 目的 1
 - 背景 1
- 概述..... 2
 - 指令 2
 - 指令的状态 2
 - 撮合 3
 - 撮合生成的过程 4
 - 成交 5
 - 开盘和封盘 5
 - 涨跌停 5
 - 收盘价和开盘价..... 6
 - 收盘价 6
 - 开盘价 6
- 功能场景 7
 - 开盘和收盘 7
 - 交易发生..... 8
 - 涨跌停限制 9
- 类图..... 10
- CRC Card 11
- 数据流图 13
- 状态图 14

验收标准 15

 功能需求..... 15

 性能需求..... 15

 储存需求..... 15

 后期运维..... 16

引言

中央交易子系统是股票交易系统的一个子系统，其主要目的是实现股票场内交易相关业务逻辑。该子系统能对用户发出的购买和出售股票的指令进行保存，匹配同一个股票的相关指令，将价格合适的两个指令进行撮合，并将交易结果放入不同的证券账户进行保存，然后修改已经匹配指令的信息。

目的

本需求说明是在系统开发前期用于详细分析系统的需求，描述系统所能提供的功能和系统的性能，确定系统的根本参数和属性。本需求说明的预期读者包括需求提出方、系统架构师、软件工程师、测试工程师和美工设计师等，同时也面向系统后期的运维工程师。为了提供软件设计的基础，系统的不同需求会被详细分析，同时所提供的相应功能也将被阐明。

背景

本软件是作为浙江大学软件工程课程实践项目“股票交易系统”的一个子系统被提出，任务提出者为该课程任课教师王章野，具体开发者为软件工程 D 小组，即陈玮烨、孙克染、张梓欣和杨清杰四位学生。软件面向的用户包括股票市场投资者与政策制定者/系统管理员两个主要人群。

概述

中央交易系统是整个股票交易过程的核心部分。所有投资者发出的买卖股票指令都在这个自动撮合系统参加集合竞价或连续竞价，交易系统根据时间优先及价格优先的原则，对符合条件的指令予以成交，这个过程就是撮合成交。股票成交后，中央交易系统随后将成交记录反馈到相关的股票帐户和证券经纪商的资金帐户。

本系统仅支持连续竞价机制，暂时不引入部分证券交易市场中在某些情况下采用的集合竞价原则。

指令 Market Orders

指令是一种投资者发出的买卖股票的请求，通常具有时间、股票代码、交易股数和价格等要素，并在一定的时间内过期，在投资者的要求下未执行的指令亦可以撤回。

指令被分为买指令(Bid)和卖指令(Ask)。买卖指令在中央交易系统内根据价格优先、时间优先的原则进行撮合。由于买卖指令的交易的股数可能不一致，一条指令可能会与多个指令相结合，其成交价为所有股份撮合价的加权平均值。

指令规定的买卖价格是可以越过涨跌停限制的，然而成交价格却不能越过涨跌停限制。我们考虑撮合价，当价格越过涨跌停限制时，最终撮合价应当以涨跌停限制为准。一个卖指令和一个买指令相撮合的部分应当具有相同的撮合价。

在现实的股票交易系统中，每次交易都应当缴纳一定的手续费。然而目前，我们的系统暂不考虑手续费等。

指令的状态

指令应当具有待撮合、被撤回、已成交等状态。

当指令满足以下条件时，其状态属于**待撮合**：

1. 指令尚处于当日交易时段。
2. 指令中仍有股份待撮合。（即指令尚未成交）
3. 指令未被撤回。
4. 其他政策规则下允许继续交易的条件。

当指令满足以下条件时，其状态属于**被撤回**：

- 用户命令撤回该指令。
- 该指令中不存在已经被撮合的股份。
- 撤回时，该指令尚处于当日的交易时段。

当指令满足以下条件时，其状态处于**已成交**：

- 该指令中所有的股份都被撮合

指令另外还具有存档属性，被存档的指令不能继续交易。当指令满足以下条件时，将指令设置为已存档：

- 指令的生成日期与系统当前的日期不一致。

撮合 Matches

在中央交易系统中，撮合是股票交易系统对买卖指令的结合。在我们的中央交易子系统中，为了方便描述，我们将一个买指令和一个卖指令对应的交易关系称为**撮合(Match)**，而把一个卖指令或买指令对应的所有的撮合的结合统计称为一个**成交(Deal)**。

一个撮合对象通常包含以下属性：撮合编号、买指令编号、卖指令编号、撮合股份数量、撮合单价、撮合时间。

撮合应当是对用户隐藏的过程，因为用户没有必要了解其发出的指令和哪些对应的指令发生了撮合，而用户仅需要了解成交的情况。

撮合生成的过程

撮合关系中的记录由关于指令表的触发器生成，即每当在指令表中插入指令记录，中央交易系统都将执行查询，判定是否可以生成撮合。

具体生成撮合的步骤如下

- 每插入一个待撮合的买指令时
 1. 查询待撮合的卖指令：对应股票中，选择所有小于等于买指令价格的卖指令，按价格由高到低排序，再按时间由前至后排序，最后按卖指令中未撮合的股票数量由大到小排序
 2. 将买指令和上述查询得到的待撮合指令一一排序撮合，每次撮合在对应的买卖指令关系中，减少“未撮合股数”对应的数字
 3. 直到该买指令的未撮合的股数为 0 或者是第一步中查询所得的卖指令全部撮合完毕为止
- 每插入一个待撮合的卖指令时
 1. 查询待撮合的买指令：对应股票中，选择所有大于等于卖指令价格的买指令，按价格由低到高排序，再按时间由前至后排序，最后按买指令中未撮合的股票数量由大到小排序。
 2. 将买指令和上述查询得到的待撮合指令一一排序撮合 每次撮合在对应的买卖指令关系中，减少 “未撮合股数”对应的数字
 3. 直到该卖指令的未撮合股数为 0 或者是第一步中查询四平的的买指令全部撮合完毕为止
- 其他撮合规则
 - 每次价格为

$$\text{撮合价格} = \frac{\text{买指令价格} + \text{卖指令价格}}{2}$$

- 价格应当不超过涨跌停限制。如果越过了这些限制，应当以交易价格为准。

成交 Deals

在中央交易系统子系统中，我们把一个指令对应的所有的撮合的统计集合称为一个成交(Deal)对象。

成交对象主要需要包含投资者发出的指令的编号、股票代码、交易数量、成交价等。

指令生成成交有两种条件，满足其一即可生成一个成交：

1. 指令规定的所有的股份都被撮合。
2. 指令规定的部分股份被撮合，且在交易日结束后未被全部撮合，则将被撮合的部分组成为一个成交。（需要在交易日结束后，处理所有的未存档待交易的指令）。

成交完成后，向前段发出通知，告知对应的指令在对应的股票代码下以多少钱交易了多少股份。

开盘和封盘

系统管理员应当根据政策等按照规定进行开盘和封盘操作。开盘后，系统处于可交易的状态，并且可以执行撮合、成交等环节。封盘后，系统中的交易停止，此时部分被撮合的指令保存为成交，并被存档。

涨跌停

涨跌停板制度源于国外早期证券市场，是证券市场中为了防止交易价格的暴涨暴跌，抑制过度投机现象，对每只证券当天价格的涨跌幅度予以适当限制的一种交易制度，规定了交易价格在一个交易日中的最大波动幅度为前一交易日收盘价上下百分之几。即规定当日交易最高价格和最低价格。

在本系统中，根据股票的性质（常规和KT股），我们对该股票的交易价格在一天内设置上下限。涨跌停的参照为前日的收盘价（收盘价的定义见下一小节）。

收盘价和开盘价

收盘价

以沪市为例，收盘价为当日该证券最后一笔交易前一分钟所有交易的成交量加权平均价（含最后一笔交易）。当日无成交的，以前收盘价为当日收盘价。

开盘价

当日的开盘价取决于第一笔成交的成交价。注意，昨日的收盘价和今日的开盘价并不一定相等，所以存在开高（今日开盘价在昨日收盘价之上）、开平（今日开盘价在昨日收盘价持平）、开低（今日开盘价在昨日收盘价之下）。

功能场景

我们的系统主要需要支持交易的顺利进行，并维护交易的规则和秩序。所以，我们将分为三个功能模块介绍我们的功能场景，并在最后描述我们用于运营维护所需的日志功能。

开盘和收盘

项目	内容
主要参与者	系统管理员
目标	开启或者关闭股票交易进程
前提	管理员的权限得到认证
场景	<ol style="list-style-type: none">1. 管理员发出请求开启/关闭股票交易的指令2. 系统审核通过该指令3. 系统向最高权限申请开启/关闭股票交易指令4. 得到许可开启/关闭股票交易
异常	<ol style="list-style-type: none">1. 管理员无权限2. 最高权限无法获取
优先级	重要
使用频率	每天开盘和收盘

交易发生

项目	内容
主要参与者	指令由投资者生成，而撮合由系统管理员规定程序
目标	实现买卖指令的撮合与成交
前提和触发条件	买家在资金允许的情况下以及政策等其他环境因素允许下发出买指令 或 卖家在持股数量足够的情况下以及政策等其他环境因素允许下发出卖指令
场景	<ol style="list-style-type: none">1. 买家发出买指令后，该指令被挂在买的位置上，卖家发出卖指令后，该指令挂在卖的地方上2. 系统检索到同时存在着买卖指令，将两个指令撮合3. 系统调整股票的价格4. 数据库后台更新数据
异常	<ol style="list-style-type: none">1. 大盘交易封停2. 无该股票信息3. 买家或者卖家的指令已经过期（被交易过了）4. 撮合价格超过/低于昨日收盘价 10%5. 买家最高价低于卖家最低价无法撮合
优先级	重要
使用频率	几乎每时每刻

涨停跌停限制

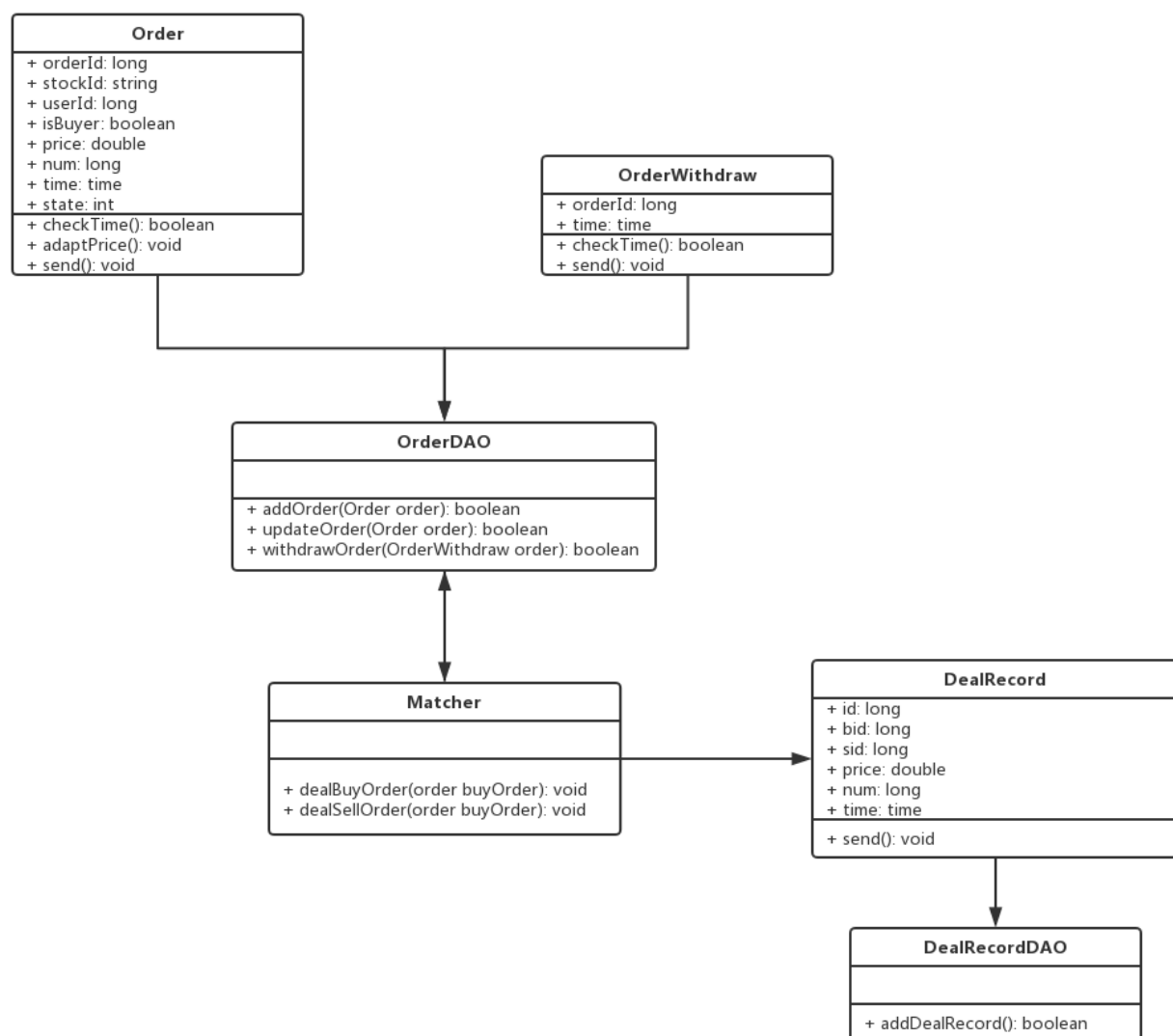
项目	内容
主要参与者	系统管理员
目标	实现涨停/跌停
前提	股票可以正常交易
触发条件	股票撮合价格达到了涨停跌停的价格（一般 10%，ST 股 5%）
场景	1. 在系统撮合成交的过程中，价格不断变化到 10%的涨停跌停限制； 2. 系统判读股票价格不能继续上升或者下降； 3. 之后的撮合价格如果高于涨停或者低于跌停价格，则拉回涨停或跌停价格； 4. 数据库进行更新
异常	1. 封盘； 2. 无该股票信息 3. 买卖指令过期；
优先级	重要
使用频率	非常频繁

系统日志

为了保证系统的正常运行，并便于后期维护，我们需要维护一个系统日志。日志应当记录以下内容：

- 产生的指令记录。
- 产生的成交信息记录。
- 产生的查询和撤回等命令记录。
- 系统异常记录。

类图



CRC Card

Order

Responsibilities	Collaborators
本类包含了一个交易申请的各项属性。在判断是否在交易时间内，并进行价格改正（涨跌停限制）后，把 Order 实例提交给数据操作类 OrderDAO 。	OrderDAO

OrderWithdraw

Responsibilities	Collaborators
本类是一个交易撤销的申请。首先判断是否在交易时间内，然后将该实例提交给数据操作类 OrderDAO 。	OrderDAO

OrderDAO

Responsibilities	Collaborators
本类是一个数据操作类，负责交易信息的插入、更新和撤销。	Order 、 OrderWithdraw 、 Matcher

Matcher

Responsibilities	Collaborators
本类负责交易的撮合，包含两个方法，分别适用于购买申请和卖出申请。一笔交易完成以后，就会生成一个 DealRecord 实例。	OrderDAO、DealRecord

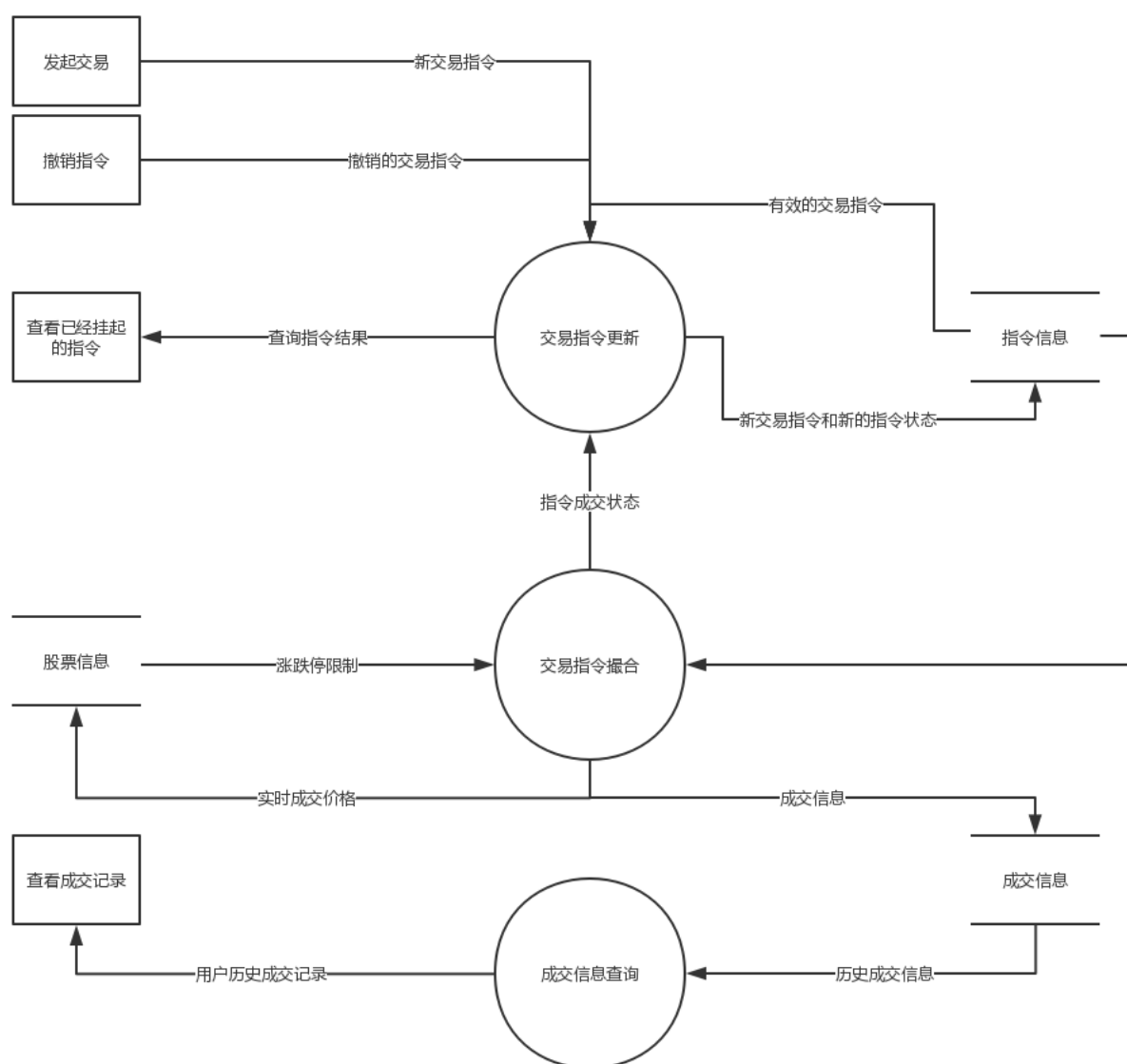
DealRecord

Responsibilities	Collaborators
本类包含了一个成交记录的各项属性。	Matcher、DealRecordDAO

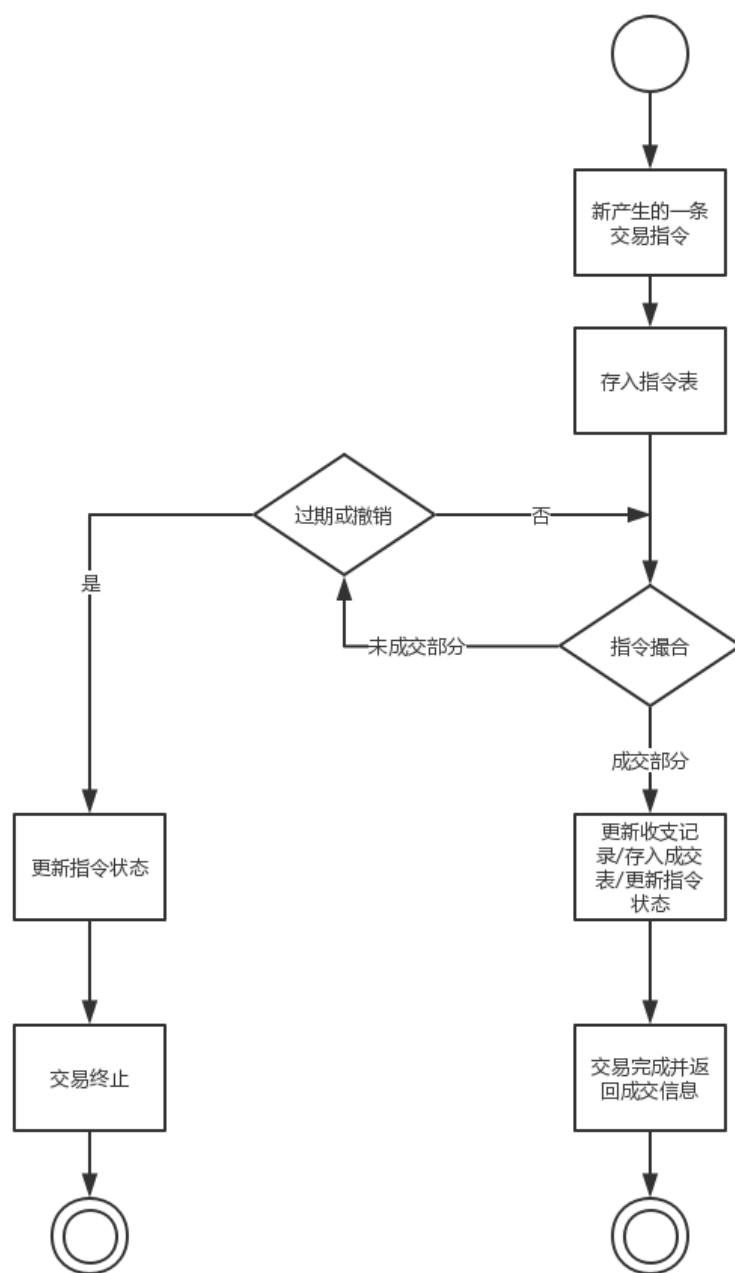
DealRecordDAO

Responsibilities	Collaborators
本类是一个数据操作类，负责成交记录的插入。	DealRecord

数据流图



状态图



?

验收标准

功能需求

- 实现从用户在前端发出的指令中读取买卖指令信息和查询、撤回命令。
 - 判断交易发生的条件，包括用户是否有足够的资金、持股以及权限等。
 - 实现撮合指令形成成交。
 - 实现股票价格涨停的控制。
 - 对接前端管理员操作，实现股票交易市场的开盘和收盘。
 - 支持获取股票的实时价格、价格变动趋势，并支持形成开盘价、收盘价、最高价、最低价等的历史查询。
 - 产生服务器系统日志。
-

性能需求

- 系统能够并行处理事务，能够控制多用户并发操作而不出错。
 - 系统能够应对频发的交易指令，具有较大的数据吞吐能力和并发处理能力。
 - 为应对较大的计算量，需要运行快速的处理算法。
 - 能够处理异常情况，保证较高的系统容错率。
-

储存需求

- 需要较大的储存空间，用于存放系统运行过程中产生的各种数据。
- 数据库系统需要支持足够大的规模。

- 储存设施和网络设备应当有较大的吞吐能力。
-

后期运维

- 保存日志文件，保障系统对于撮合操作的正确性，已经应对突发情况的处理。
- 模块化系统化的设计，建立适合维护的应用系统，并方便日后扩展功能。
- 开发过程中，在 **GitHub** 上建立源代码版本控制。
- 定期维护数据库系统，备份数据，防止数据丢失。