Central Trading System
— subsystem of Stock Trading System

Software Requirements Specification

Team leader：陈徐希
Author：　　　陈徐希、管铮、
　　　　　　　刘世林、赖剑、
　　　　　　　吴玉书
Version：　　 1.0
Date：　　　　9.27.2007

TABLE CONTENTS

1. Introduction

1.1 Goals and objectives

The central trading system (CTS) is to complete the trading of stock. It analyses the instructions that enter the central trading system and divides them into several kinds of instructions. System will make a match between them under specifically prescripts. Also central trading system provides some interfaces to send messages to other modules.

1.2 System Context

The CTS is just a subsystem of the whole stock trading system (STS) system. There are mainly six subsystems involved, stock account operation, financing account operation, trade client serve, network message promulgating, centeral trading system and trading system manage. Communications exist among these subsystems which complicates the relationships between subsystems.

The CTS receive instructions from trade client serve. Then the CTS produce new messages which to be sent to network message promulgating after deal with the instructions from trade client serve. Also the system manager is authorization to access the information in CTS.

1.3 Potential Users' Specification

There are mainly two kinds of users. The CTS provides the terminal users easy Operations that are confined to a series of mouse clicks and keyboard, which compared to other systems are really much simplified. However, for another user, things are different .the maintainers of this CTS must be familiar with java programming and the socket. When the system crashes down, they can find the cause and fix it. When new requirements should be supported, they can modify the program to make it fit.

As the instruction is frequently operated in CTS such as fetch, deal with, repeal and so on, the CTS is heavily transferred. The program must take this into consideration. The maintainers of CTS should have good strategies to overcome crash of the system when overhead exceeds the capacity of it.

2. User Scenario

2.1 User Profiles

There are mainly five kinds of actors, which here consist of other related sub-systems, involved with our Central Trading System. They are Transaction User Interface, Security

Account Management, Trading Management System, Trading Information Release.

2.2   User Scenarios Specification

2.2.1   Personal Opinion

Often, user-scenario is considered the same to use-case. They both can help our team understand how system functions and features will be used by different classes of end-users (actors). However, I prefer to understand it like this: a user scenario is a particular environment within a specified use case. So, I will specify user-scenarios going along with developing use cases in the next session.

2.2.2   Develop Use-Cases within which User-Scenarios are specified

Use-case	Buy stock
Primary actor	Transaction User Interface
Goal in context	To fulfill the buy transaction
Preconditions	Customer has successfully logged into UI and submit the buy information
Trigger	The 'buy' button on UI been clicked
Scenario	1. transaction user interface: give the buy instruction 2. central trading system: save the buy instruction 3. central trading system: match the instructions with the same stock id 4. central trading system: make a trade by matching 5. central trading system: modify the information of matched instructions
Exception	1. all the operations have been suspended 2. some trade exceptions come up 3. no matched stock with the buy instruction
Priority	Essential, must be implemented
When available	First increment
Frequency of use	Frequent
Channel to actor	Via transaction user interface
Secondary actors	Trading System Management
Channels to secondary actors	Via central trading system interface

Open issues	1. Should the trade information return to Transaction User Interface immediately when the deal is done? 2. Should the failure of trade return to Transaction User Interface the next day? 3. If exception happens, is exception log needed? 4. security issue

Use-case	Sell stock
Primary actor	Transaction User Interface
Goal in context	To fulfill the sell transaction
Preconditions	Customer has successfully logged into UI and submit the sell information
Trigger	The 'sell' button on UI been clicked
Scenario	1. transaction user interface: give the sell instruction 2. central trading system: save the sell instruction 3. central trading system: match the instructions with the same stock id 4. central trading system: make a trade by matching 5. central trading system: modify the information of matched instructions
Exception	1. all the operations have been suspended 2. some deal exceptions come up 3. no matched stock with the buy instruction
Priority	Essential, must be implemented
When available	First increment
Frequency of use	Frequent
Channel to actor	Via transaction user interface
Secondary actors	Trading System Management
Channels to secondary actors	Via central trading system interface
Open issues	1. Should the trade information return to Transaction User Interface immediately when the deal is done? 2. Should the failure of trade return to Transaction User Interface the next day? 3. If exception happens, is exception log needed? 4. security issue

Use-case	Cancel trading instruction
Primary actor	Transaction User Interface
Goal in context	To fulfill the cancel transaction
Preconditions	Customer has successfully logged into UI and submit the cancel information
Trigger	The 'cancel' button on UI been clicked
Scenario	1. transaction user interface: give the cancel instruction 2. central trading system: save the cancel instruction 3. central trading system: cancel the correlative instruction
Exception	1. all the operations have been suspended 2. the instruction concerned has been implemented 3. no matched instruction to be cancelled
Priority	Moderate, to be implemented after basic functions
When available	Second increment
Frequency of use	Many times per day
Channel to actor	Via transaction user interface
Secondary actors	Trading System Management
Channels to secondary actors	Via central trading system interface
Open issues	1. Is it necessary to return the status of cancel transaction? 2. If exception happens, is exception log needed? 3. security issue

Use-case	Save trade information
Primary actor	Security Account Management
Goal in context	To save all successful trade information
Preconditions	The correlative trade transactions have been successfully conducted
Trigger	Any trade transaction been conducted
Scenario	1. central trading system: give out the successful trade information 2. security account management: save the trade information

Exception	1. all the operations have been suspended 2. no matched stock with the buy instruction
Priority	Essential, must be implemented
When available	First increment
Frequency of use	Frequent
Channel to actor	Via central trading system interface
Secondary actors	Trading System Management
Channels to secondary actors	Via central trading system interface
Open issues	1. If exception happens, is exception log needed? 2. Will we develop a media (ie. Web page) where any customer can look up the latest trade records which have been saved? 3. security issue

Use-case	Query trade information
Primary actor	Trading Information Release
Goal in context	To query the trading information which needed to be statistically analyzed and released on a web site
Preconditions	Central Trading System deal transactions well
Trigger	Trading Information Release System send a query
Scenario	1. trading information release system: send a query 2. central trading system: implement the query 3. central trading system: structuralize the queried data 4. central trading system: send the data to release
Exception	1. all the operations have been suspended 2. the query is invalid 3. no matched data
Priority	Essential, must be implemented
When available	First increment
Frequency of use	Frequent
Channel to actor	Via central trading system interface
Secondary actors	Trading System Management

Channels to secondary actors	Via central trading system interface
Open issues	1. If exception happens, is exception log needed? 2. How frequent the release system is to send a query? 3. Do we need a warning if any query failed? 4. security issue

2.2.3 Use-case diagram for Central Trading System function
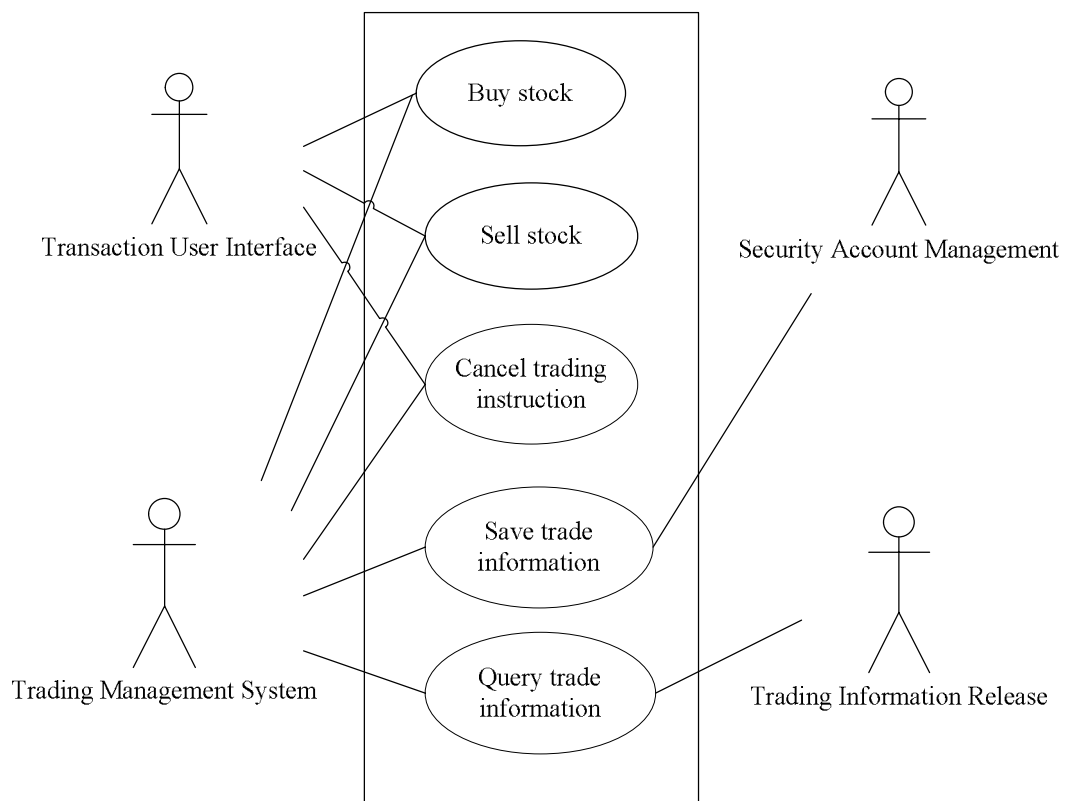
3 Data Flow Diagram

3.1 Context-level DFD for the stock trade system
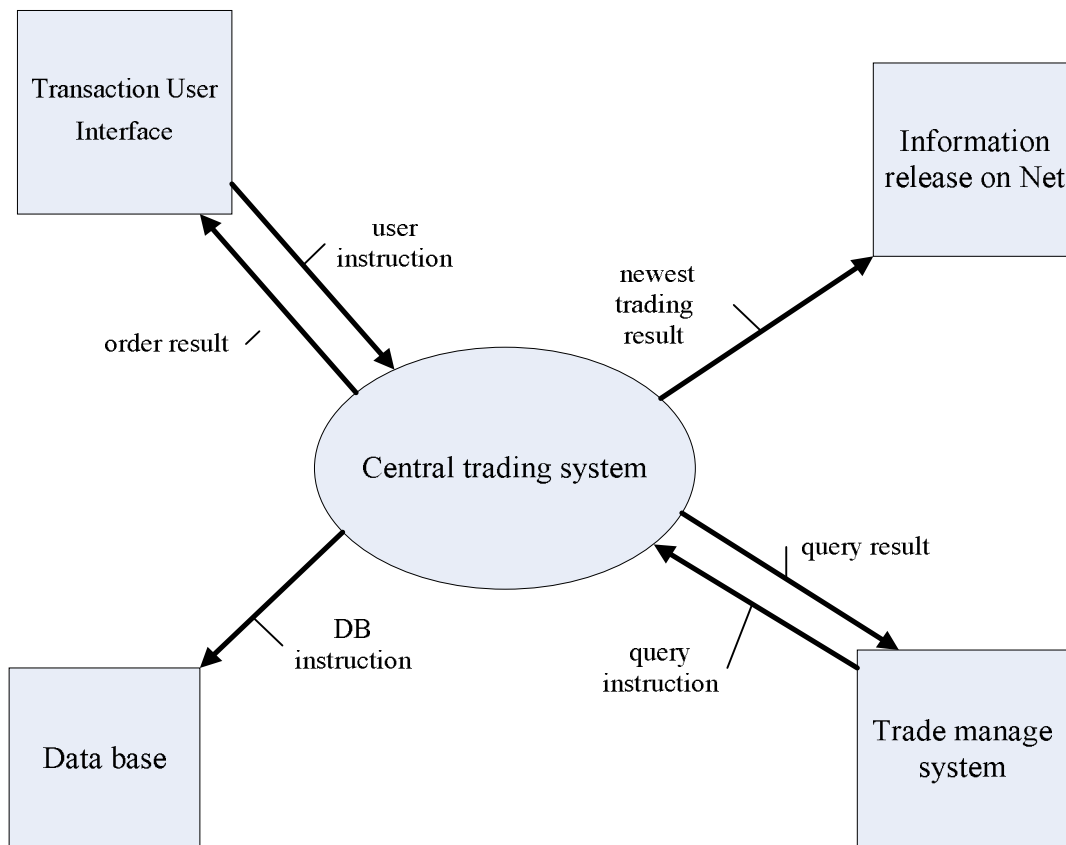
Hint: we only indicate the flow that involved the central trading system.

3.2 Level-1 DFD for the central trade system
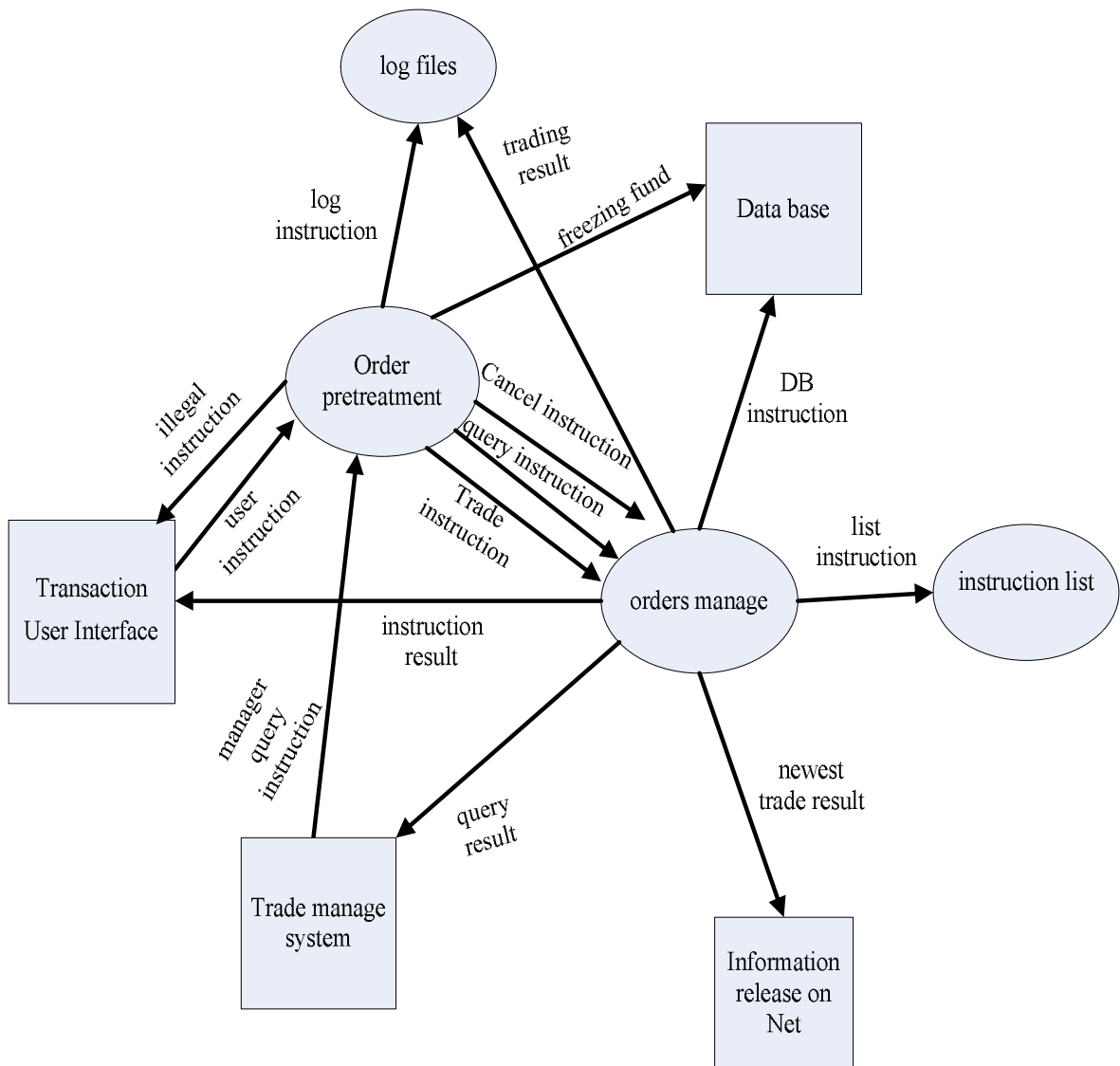
The instructions first arrive at the instruction pretreatment modular which will judge the validity of the instruction and freeze the fund and write the log files. Then the instructions will go to the instructions manager modular. This modular will deal with all the three kinds of instructions. Then it will send the results to the Trading client serve and information releasing modular as well as keep a log file.

3.3 Level 2 DFD for the instruction pretreatment
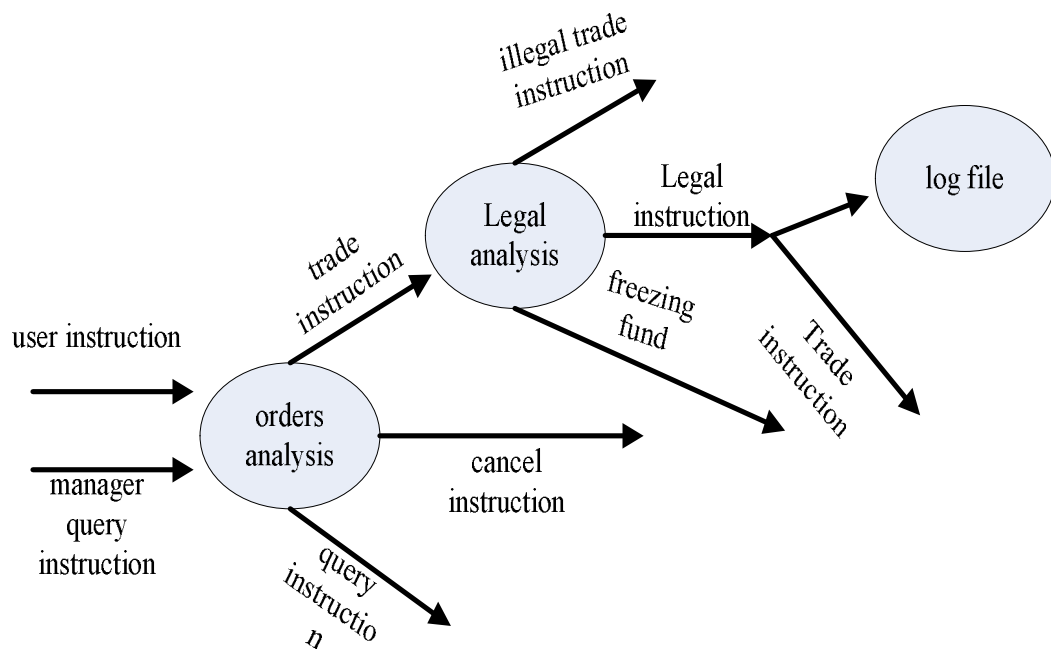
The instructions analysis will identify the kind of the instructions. Then it will deal with the illegal instructions like the raising limit and freezing the fund in the Data Base. In addition, it will keep log file about the instructions.

3.4 Level-2 DFD of the instructions manager
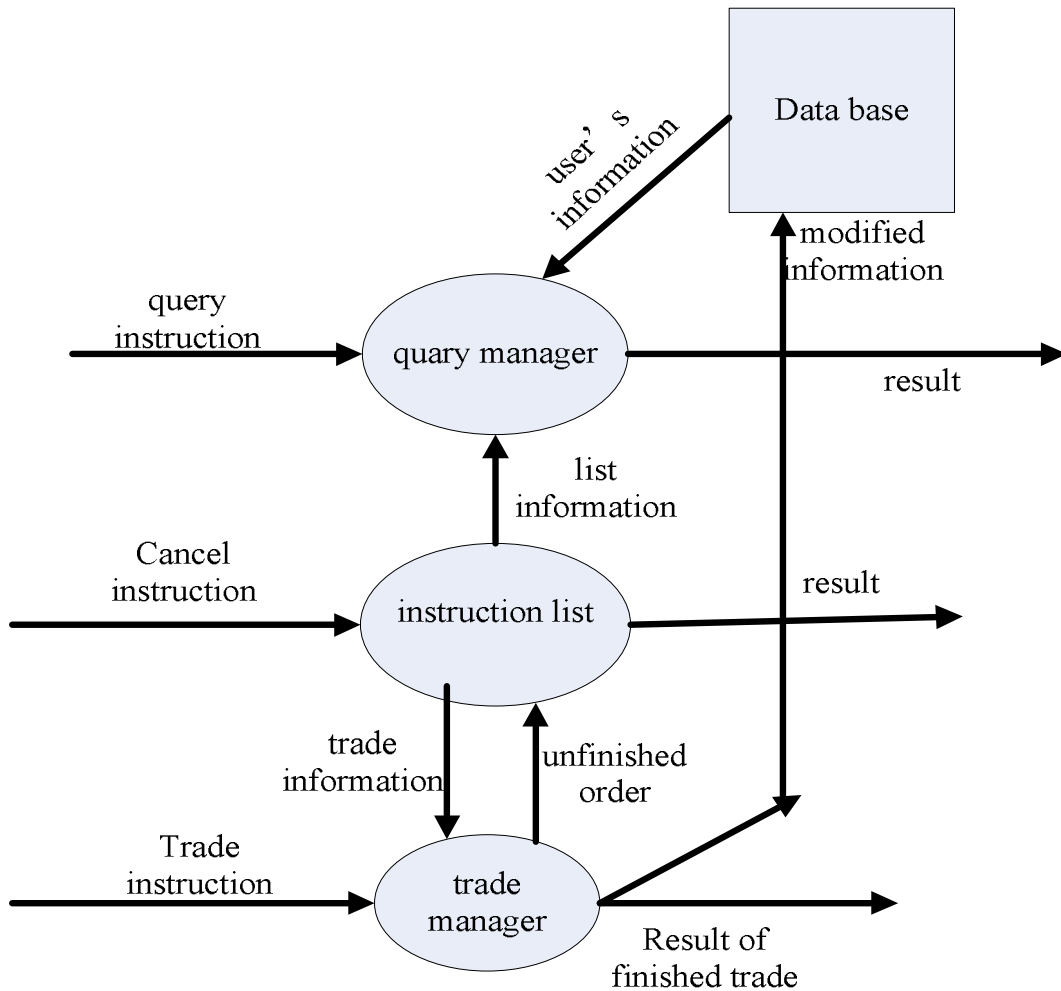
In this level the three kinds of instructions will go to three different modules. The query instruction will refer to the Data Base or the instructions list. The cancel instruction will delete the instruction the instruction in the instructions list. And the trade instruction will go to the trade manager to make a match in the instructions list.

4   Data Dictionary

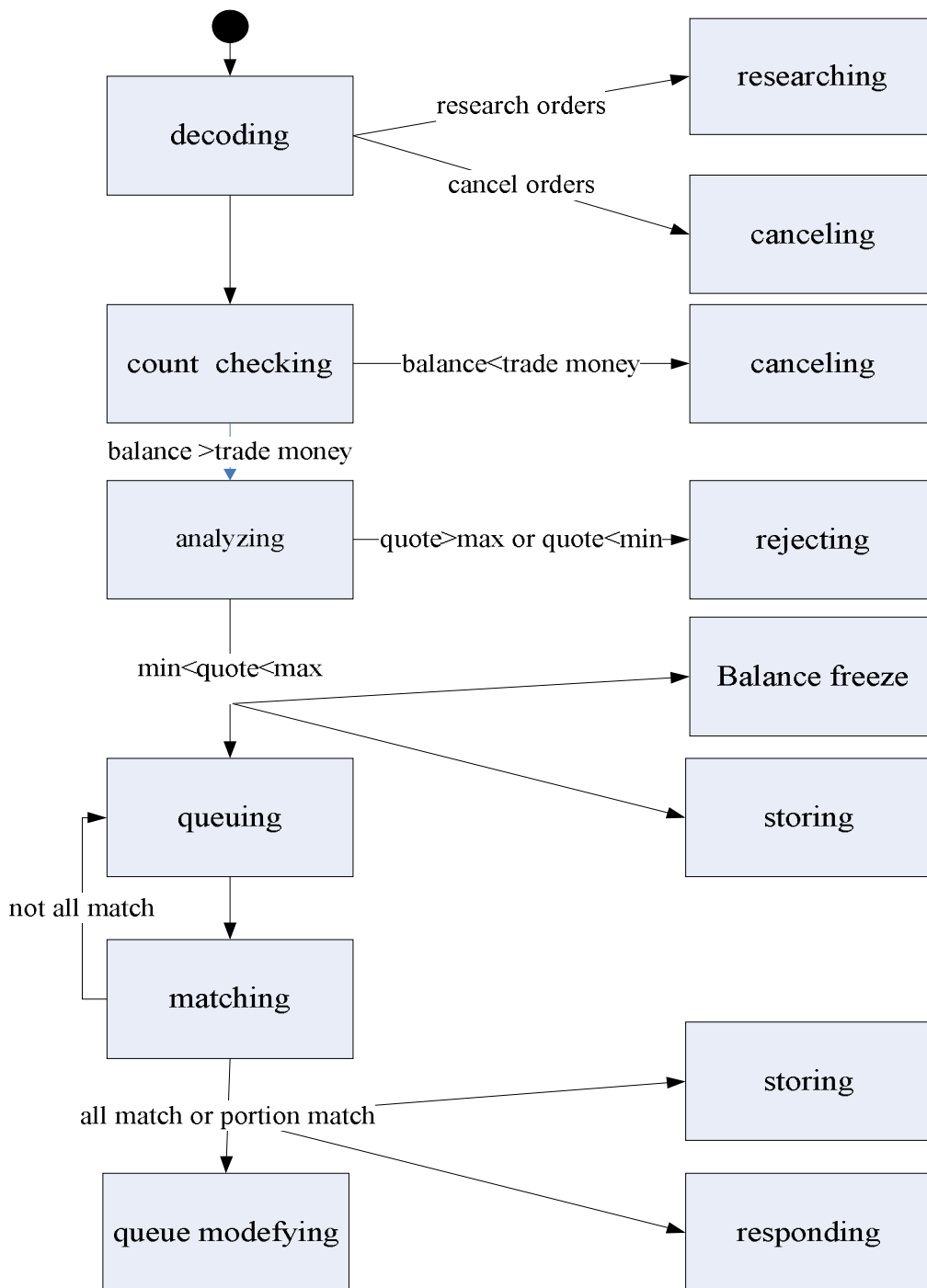| Name	Alias	When /how to use	Description
central trading system	CTS	receive the instructions and return the result.	to accomplish the matching work of the stock trading system as well as the query.
user instruction		trading client serve(output) CTS(input)	Indicate the users' instruction (buy、sell、query), also include the trading quantity, and the time stamp.
instruction result		CTS(output)	Indicate the handled result and return to the other three sub system of the whole stock trading system.
DB instruction		Data Base(input)	Include the query and writing and modifying instructions to the Data Base
newest trading result		CTS(output) Information releasing module(input)	Send the newest trading result and status to the information releasing module.
Instruction pretreatment		deal the instruction for first step	Judge the validity of the instruction Write the log file Freeze the fund of user's account
instruction manager		deal the instruction in detail.	has the instruction input and deal with it.
log instruction		trading client serve (output) CTS (input)	Indicate the users' instruction (buy、sell、query), also include the trading quantity, and the time stamp.
freezing fund		instruction pretreatment(output) DB (input)	Freeze the user's corresponding account accord to the trade instruction.
list instruction		instruction manager(output) instruction list(input)	The instruction that involve the several operations of the list as well as the making match of the instructions.

5   State Diagram

```
      ●
      │
      ▼
  ┌──────────┐  research orders    ┌──────────────┐
  │ decoding │─────────────────────│  researching │
  │          │                     └──────────────┘
  └──────────┘    cancel orders    ┌──────────────┐
      │         ─────────────────── │   canceling  │
      │                            └──────────────┘
      ▼
  ┌───────────────┐  balance<trade money  ┌──────────────┐
  │ count checking│──────────────────────▶│   canceling  │
  └───────────────┘                       └──────────────┘
      │
 balance >trade money
      ▼
  ┌──────────┐  quote>max or quote<min  ┌──────────────┐
  │ analyzing│─────────────────────────▶│   rejecting  │
  └──────────┘                          └──────────────┘
      │
 min<quote<max                          ┌──────────────┐
      │                                 │ Balance freeze│
      ▼                                 └──────────────┘
  ┌──────────┐                          ┌──────────────┐
  │ queuing  │                          │   storing    │
  └──────────┘                          └──────────────┘
      │
 not all match
      ▼
  ┌──────────┐
  │ matching │
  └──────────┘
      │                                 ┌──────────────┐
 all match or portion match             │   storing    │
      │                                 └──────────────┘
      ▼                                 ┌──────────────┐
  ┌────────────────┐                    │  responding  │
  │ queue modefying│                    └──────────────┘
  └────────────────┘
```

Here is the state diagram. There are several states in the diagram, which are decoding, researching, canceling, analyzing, rejecting, queuing, storing, matching, queue modifying, responding. The flow between the adjacent states goes their way according to specific conditions which are displayed on the flow line.

6 CRC index cards

Class:	PretreatmentOfInstruction
Description:	
After a new instruction come, we must get key information from instruction and solve some simple problems by necessary constraints and some key rules	
Responsibility:	Collaborator:
Determination of prices'increments and decrements constraints of Instruction	
Freeze account of buyers	ManagementOfDatabase
Prededuct commission charge and tax	ManagementOfDatabase
Log instruction	
Send to Management of instruction	ManagementOfInstruction

Class:	ManagementOfInstruction
Description:	
Manage and maintain the instructions of buyers and sellers	
Responsibility:	Collaborator:
Add instructions	
Cancel instructions	
Search instructions	
Sort instructions	
Class: ManagementOfInstruction	

Class:	ManagementOfDealing
Description:	
Deal business following the key rules, and some operations are executed after the deals	
Responsibility:	Collaborator:

Deal the business	
Log the results of business	
Store some key data to database	ManagementOfDatabase
Determine the commission charge and tax	

Class: ManagementOfDatabase	
Description:	
Support the basic operations interface of database for other modules	
Responsibility:	Collaborator:
Operations of account, insert, delete, update	
Operations of stock, insert, delete, update	

7 Validation Criteria

7.1 Interface criteria

The central trading system has relation with three modules that are trading client serve、trading manager system、information releasing module. These modules have contact with CTS using the interface supplied by CTS.

7.1.1 Transaction User Interface

The Transaction User Interface has to input the instructions. The input instruction is divided into three kinds: buy instruction, sell instruction and query instruction.

a) *buy instruction*
   This instruction should have five parameters: user ID, stock ID, quantity, respected price, timestamp.

b) *sell instruction*
   This instruction should have five parameters: user ID, stock ID, quantity, respected price, timestamp.

c) *query instruction*
   This instruction can be divided into two kinds: user query instruction and the stock query instruction. The user query instruction should have three

parameters: user ID, query content, some restrict parameters. The stock query instruction should has three parameters: stock ID，query content, some restrict parameter。

7.1.2　Information Releasing Module

The information releasing module use the interface to find the price of the stocks. The input should has two parameters including the stock name and the restrict parameter.

7.1.3　Trading manager system

The trading manager system uses the interface to find the user's trading instruction. The input should has two parameters including the trading instruction type(buy or sell),and the restrict parameter.

7.2　Function Criteria

The CTS deals with trading instruction, query instruction and some cancel instruction. The functions are as follows:

1)　instruction matching
When the client serve get a trading instruction, it will send the instruction to the CTS to trade with other trading instructions. The process of the trading includes the following two main principles: price first principle and the time first principle. If the trading fails when these two principles have been applied, we should refer to another principle. If the lowest buy price is higher than the highest sell price, then the CTS will make a match of this trading.

2)　rising and falling limit
If the trading price is higher(lower) than the rising limit (falling limit), the CTS will reject this trading instruction.

3)　return the result
The CTS will return the result to the client serve for any trading instructions than go into the CTS.
The trading states are divided into two kinds: totally finished and the partially finished.

4)　outdated instruction
If a instruction haven't finished it's trading in one day, then it will be removed form the CTS for out of date.