交易后台QUEEN（v0.0.1）

# 目录

[目录 1](#_Toc40045590)

[修订（开发）记录 3](#_Toc40045591)

[一．QUEEN整体设计 4](#_Toc40045592)

[二．QUEEN周边系统 5](#_Toc40045593)

[三．APMS功能设计 6](#_Toc40045594)

[3.1 初始化/重启服务 6](#_Toc40045595)

[3.2 冻结资金/冻结股份—买单/卖单委托 7](#_Toc40045596)

[3.3 解冻资金/解冻股份—卖单/卖单撤销委托 8](#_Toc40045597)

[3.4 扣减（冻结中）资金并增加股份—买单成交 8](#_Toc40045598)

[3.5 增加资金并扣减（冻结中）股份—卖单成交 9](#_Toc40045599)

[3.6 增加资金/减少资金 9](#_Toc40045600)

[3.7 修改/新增账户信息 10](#_Toc40045601)

[3.8 冻结/解冻账户 11](#_Toc40045602)

[3.9 异步持久化更新数据库 11](#_Toc40045603)

[四．RDS功能设计 12](#_Toc40045604)

[4.1 交易日RDS 12](#_Toc40045605)

[4.2 证券信息RDS（Stock/Fund/Future/Repo/Bond/Index） 12](#_Toc40045606)

[4.3 定时失效更新 12](#_Toc40045607)

[4.4 强制更新（全量or指定证券id） 12](#_Toc40045608)

[五．APMS&RDS对外暴露接口 13](#_Toc40045609)

[六．数据库设计 14](#_Toc40045610)

[6.1 APMS表 14](#_Toc40045611)

[6.1.1 交易账户信息表（queen\_account） 14](#_Toc40045612)

[6.1.2 持仓信息表(queen\_position) 14](#_Toc40045613)

[6.1.3 资金信息表(queen\_cash) 14](#_Toc40045614)

[6.1.4 持仓详细流水表（queen\_position\_statement） 14](#_Toc40045615)

[6.1.5 资金详细流水表（queen\_cash\_statement） 14](#_Toc40045616)

[6.2 RDS表 14](#_Toc40045617)

[6.2.1 交易日信息表（queen\_rds\_holiday） 14](#_Toc40045618)

[6.2.2 股票静态信息表（queen\_rds\_stock） 14](#_Toc40045619)

[6.2.3 债券静态信息表（queen\_rds\_bond） 14](#_Toc40045620)

[6.2.4 基金静态信息表（queen\_rds\_fund） 14](#_Toc40045621)

[6.2.5 回购静态信息表（queen\_rds\_repo） 14](#_Toc40045622)

[6.2.6 期货静态信息表（queen\_rds\_future） 14](#_Toc40045623)

[6.2.7 指数静态信息表（queen\_rds\_index） 14](#_Toc40045624)

# 修订（开发）记录

| 日期 | 版本 | 修订说明 | 修订者 |
| --- | --- | --- | --- |
| 2020.05.10 | v0.0.1 | 1. QUEEN详细设计起草 | 姜瀚 |
|  |  |  |  |

# 一．QUEEN整体设计

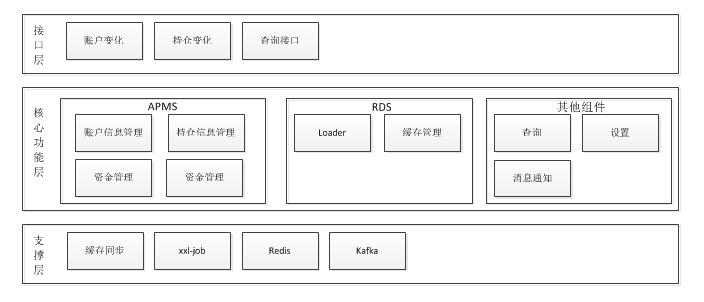


图1.1 QUEEN整体设计

交易支持QUEEN用于管理与业务非强相关的信息系统，目前设计包括APMS账户&持仓管理服务（Account Postion Manager Service）和RDS证券参考信息服务（Reference Data Service）：

***APMS*** : 负责维护FAMS中交易账号的持仓管理、资金管理，以RPC的方式对上层（主要是ACE-OMS、KING-OrderExecutor和UpStream）提供服务。

***RDS*** : 负责维护FAMS中所有的静态证券信息、交易日信息等静态较稳定的数据。

# 二．QUEEN周边系统

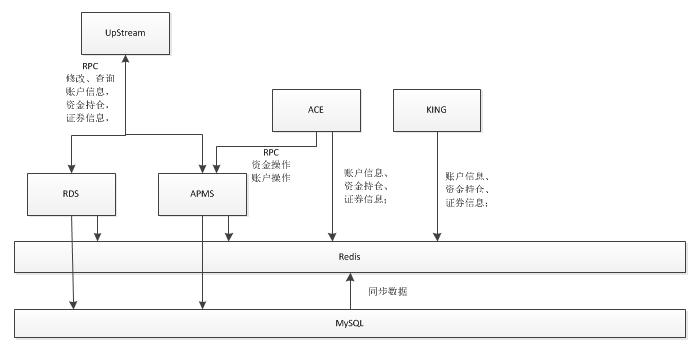


图2.1 QUEEN周边系统

RDS周边系统较为简单，因为所有数据在RDS启动时，都从数据库放入redis，而其他服务可以直接从redis中获取数据。RDS需要定时的从数据库捞取数据来更新redis

APMS周围系统包括直接上游OMS和KING，所有接口均以RPC方式提供给上游。APMS是以redis为核心的服务，需要定时线程来将redis数据同步给数据库。为了redis加锁的效率，对资金表、持仓表进行分表。即根据key的hash值，划分到不同的分表中：

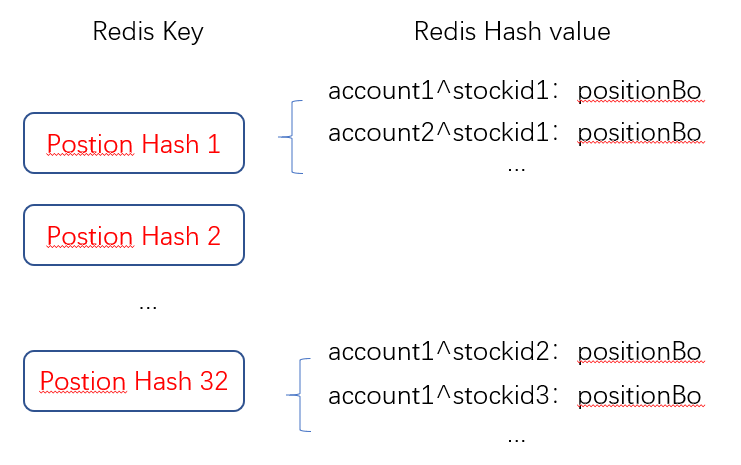


图2.2 redis分表

如上图的持仓表，根据key（account^stockid）的hash值再%32,从而将持仓分散到32个子表中。在冻结操作中，从而降低redis加锁带来的性能下降。对应的资金表的key是account^currency

# 三．APMS功能设计

## 3.1 初始化/重启服务

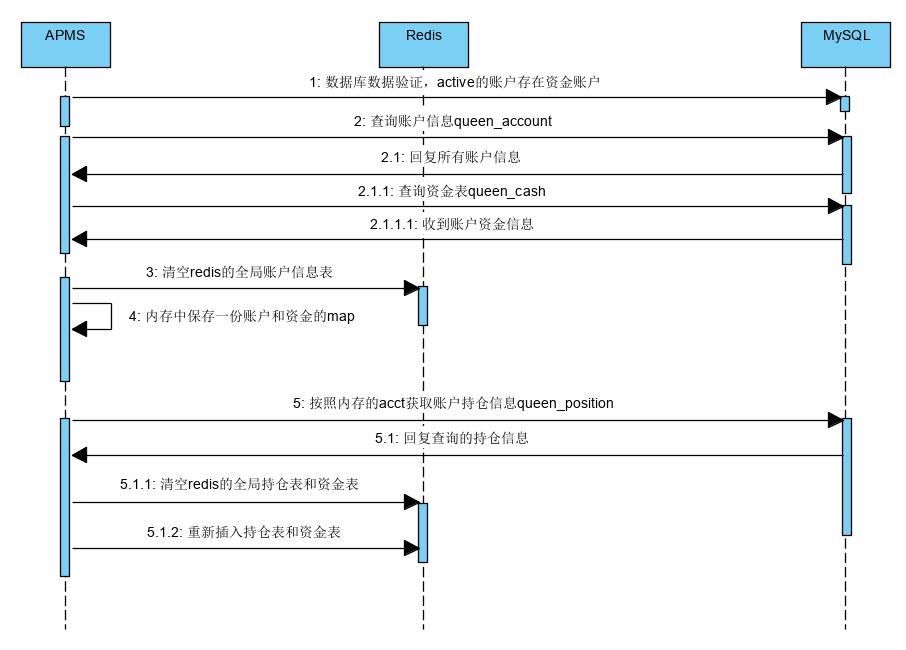


图3.1 APMS节点初始化

场景：当一个APMS实例启动时：

1. 首先检查redis的在线服务表，查看当前是否有存活的APMS节点，如果有，只需要启动后等待即可；如果没有存活中的APMS节点，并做以下事情：
2. 查询物理数据库，关联查询active中的账户和资金，确认每个账户至少在资金表（cash）中存在一个资金账户，如果存在异常，需要报警。
3. 对redis的账户表、资金表和持仓表加锁；查询交易账户表queen\_account，并写入redis；查询资金表queen\_cash，并存入redis
4. 关联账户表的accout去数据库查询每个账户的持仓，写入持仓表position
5. 释放redis中的三个表的锁，完成redis的初始化

## 3.2 冻结资金/冻结股份—买单/卖单委托

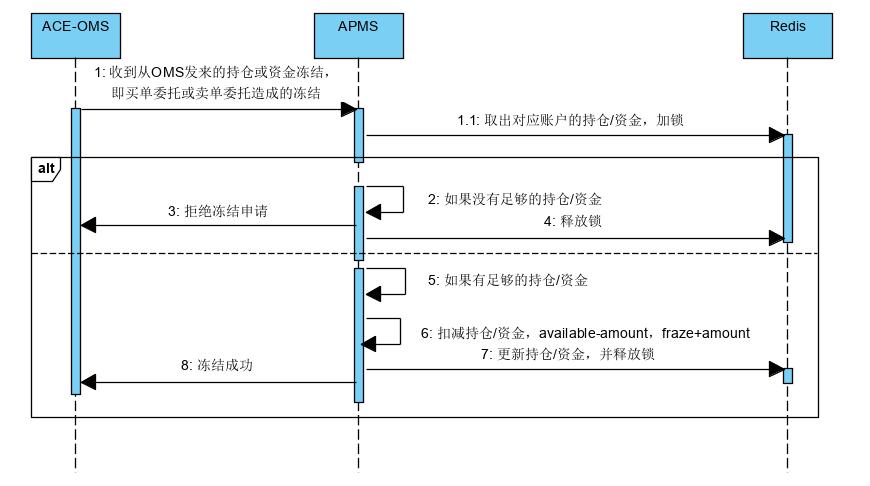


图3.2 APMS的冻结资金/冻结股份接口

场景：APMS收到OMS的持仓或资金冻结请求：

1. 从redis中取出对应账户的持仓或资金，如果没有对应的持仓和资金，直接返回失败；否则，并对该子表加锁。
2. 冻结持仓：可用持仓 = 可用持仓–请求数量；冻结持仓 = 冻结持仓+请求数量

冻结资金：可用资金 =可用资金-请求金额；冻结资金 = 冻结资金+请求金额

1. 上述过程如无异常，更新redis并释放锁，并返回OMS结果
2. 如有异常，释放锁，返回OMS冻结失败

## 3.3 解冻资金/解冻股份—卖单/卖单撤销委托

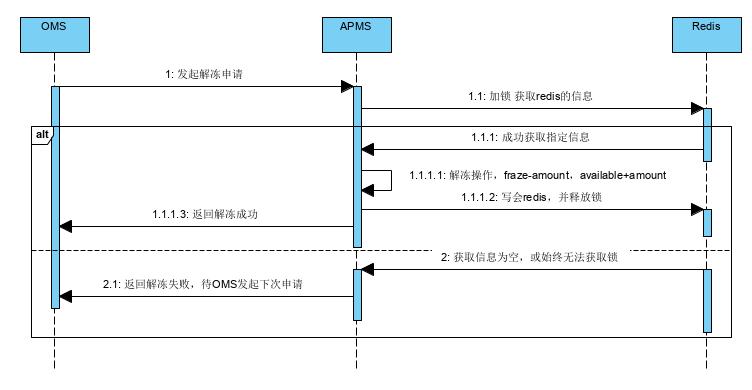


图3.3 解冻资金和解冻持仓

场景：收到OMS的撤销委托时

解冻资金和解冻股份过程类似上述过程。只是公式变为：

解冻持仓：可用持仓= 可用持仓+请求数量；冻结持仓=冻结持仓-请求数量

解冻资金：可用资金=可用资金+请求金额；冻结资金=冻结资金-请求金额

## 3.4 扣减（冻结中）资金并增加股份—买单成交

场景：OMS处理买单成交时

扣减（冻结中）资金并增加股份类似上述过程。只是公式变为：

持仓：冻结持仓=冻结持仓-请求数量；

T+1类：T1持仓= T1持仓+请求数量 T+0类：可用持仓=可用持仓+请求数量

资金：冻结资金=冻结资金-请求金额

## 3.5 增加资金并扣减（冻结中）股份—卖单成交

场景：OMS处理卖单成交时

增加资金并扣减（冻结中）股份类似上述过程。只是公式变为：

持仓：冻结持仓=冻结持仓-请求数量；

资金：可用资金=可用资金+请求金额；

## 3.6 增加资金/减少资金

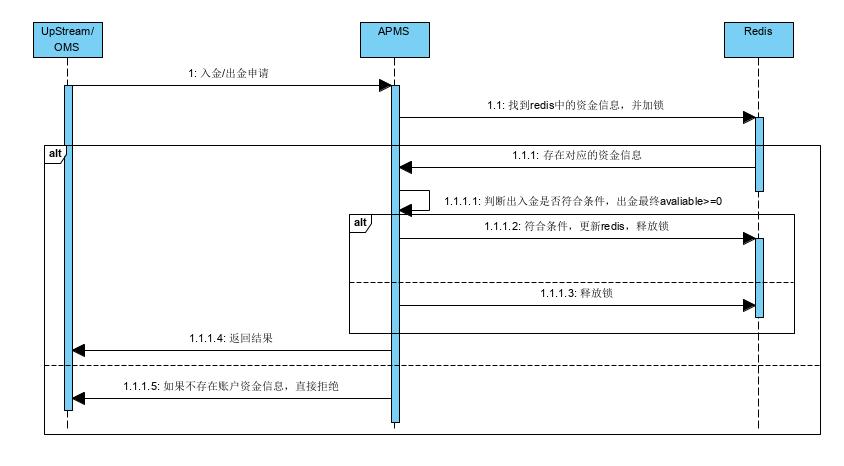


图3.4 入金/出金情景

场景：交易员从对交易账号的资金进行操作（交易账户和对公账户之间的交割）

1. 收到入金出金请求，检查redis的账户资金信息，并加锁
2. 如果存在对应的信息，如果是出金，需要判断资产是否足够；如果是入金直接在可用资金上增加。最后更新回redis，并释放锁
3. 如果不存在资金信息，拒绝申请

## 3.7 修改/新增账户信息

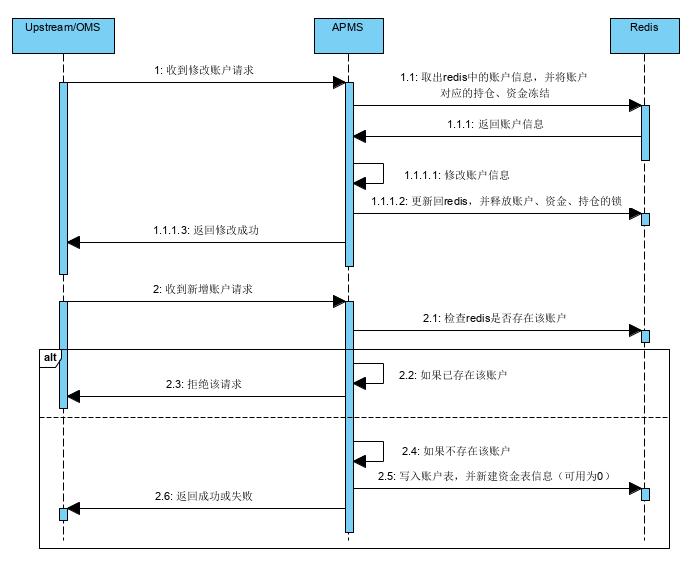


图3.5 修改/新增交易账户

场景：前台Upstream发起的修改交易账户或新增账户请求

修改：

1. 取出redis中的账户信息，同时将该账户的资金、持仓进行加锁（这两把锁不好加，因为持仓信息分布在多个子表中）
2. 修改账户信息后，更新回redis
3. 返回前台结果

新增：

1. 检查是否有重复账户
2. 若没有，新建账户对应的资金信息，新建账户信息，更新入redis
3. 返回前台结果

## 3.8 冻结/解冻账户

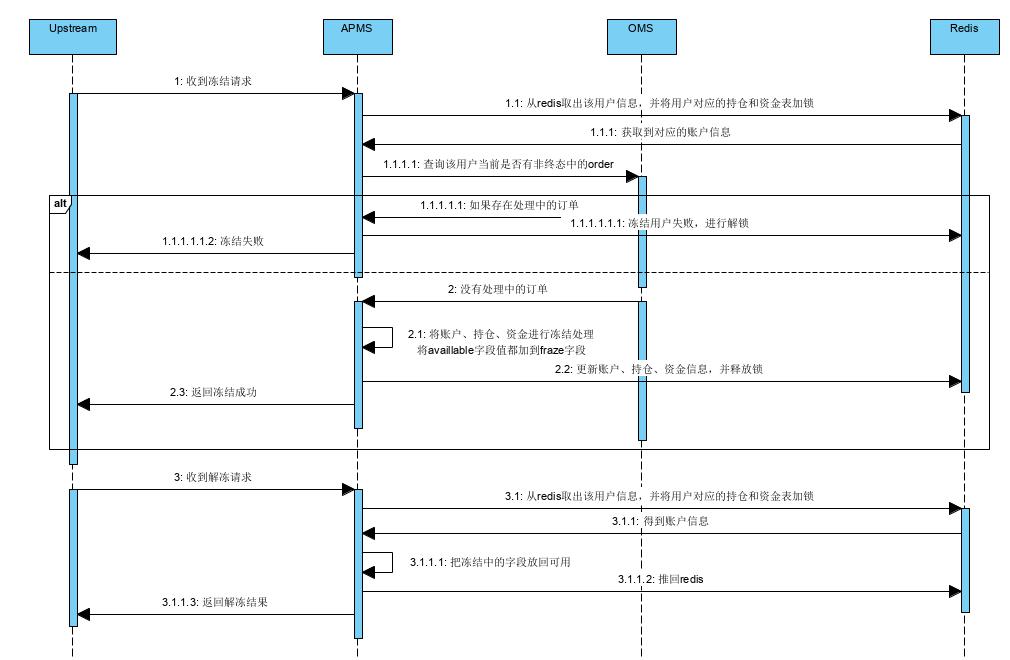


图3.6 冻结/解冻交易账户

场景：前台发起的冻结解冻交易账户申请

冻结：

1. 从redis中取出账户信息，并对账户对应的持仓和资金表加锁（也不好加）
2. 向OMS发起查询，查询当前账户是否有处于非终态的委托，如果有，则返回前台冻结失败，并解锁redis
3. 如果所有委托都已完成，将账户信息的status标志设置为FRAZE，更新回redis并解锁

## 3.9 异步持久化更新数据库

见OMS，可考虑设计成公用服务，放入framework

# 四．RDS功能设计

## 4.1 交易日RDS

## 4.2 证券信息RDS（Stock/Fund/Future/Repo/Bond/Index）

## 4.3 定时失效更新

## 4.4 强制更新（全量or指定证券id）

# 五．APMS&RDS对外暴露接口

# 六．数据库设计

## 6.1 APMS表

### 6.1.1 交易账户信息表（queen\_account）

### 6.1.2 持仓信息表(queen\_position)

### 6.1.3 资金信息表(queen\_cash)

### 6.1.4 持仓详细流水表（queen\_position\_statement）

### 6.1.5 资金详细流水表（queen\_cash\_statement）

## 6.2 RDS表

### 6.2.1 交易日信息表（queen\_rds\_holiday）

### 6.2.2 股票静态信息表（queen\_rds\_stock）

### 6.2.3 债券静态信息表（queen\_rds\_bond）

### 6.2.4 基金静态信息表（queen\_rds\_fund）

### 6.2.5 回购静态信息表（queen\_rds\_repo）

### 6.2.6 期货静态信息表（queen\_rds\_future）

### 6.2.7 指数静态信息表（queen\_rds\_index）