期货系统数据库，用于资金内转

172.16.130.17:3306/trading root/sptrader

**0、扣款时对可用资金判断**

如果是withdraw，则查询可用资金，并判断收付是否超过可用资金 getDrawableAmount

（1）获取现金余额，并汇率换算

*select* ***cash(上日结余) + today\_cash(今日金额)****, ccy from* ***acc\_bal*** *where acc\_no='9900900001'*

（2）获取其它金额

*select* ***credit\_limit(信用额度)*** *+* ***total\_pl(总盈亏)*** *-* ***imargin(初始保证金)*** *from* ***acc\_mkt*** *where acc\_no='9900085518' and active='1'*

（3）购买力=现金余额+信用额度+总盈亏-初始保证金

（4）取购买力和现金余额的较小值做为可用资金

1. **查询aeid**

FutFundTrans.futureVars1

*select #accNo# as accNo,#ccy# as ccy,#amount# as amount,#remark# as remark,*

*#userid# as userid,#apprid# as apprid,#psw# as psw,#buyorsell# as buyorsell,*

*ae\_id as aeid*

*from* ***acc\_mast*** *where acmast\_no=#accNo#*

1. **获取单号**

FutFundTrans.futureVars2

*select next\_order\_no as nextOrderNo from* *acc\_next\_no where acc\_no=#accNo#*

（1）存在，则更新

FutFundTrans.updateAccNextNo

*update acc\_next\_no set next\_order\_no=next\_order\_no+1 where acc\_no=#accNo#*

（2）不存在，则插入

FutFundTrans.insertAccNextNo

*insert into acc\_next\_no(acc\_no,next\_order\_no,next\_req\_no) values( #accNo#, 1, 1)*

**3、获取资金号 *acc\_bal资金余额表***

FutFundTrans.selectAccBal

*select cash,today\_cash from acc\_bal where acc\_no=#accNo# and ccy=#ccy#*

（1）存在，则更新

FutFundTrans.updateAccBal

*update acc\_bal set today\_cash=today\_cash+#amount2# where acc\_no=#accNo# and ccy=#ccy#*

（2）不存在，则插入

FutFundTrans.insertAccBal

*insert acc\_bal set today\_cash=#amount2# ,acc\_no=#accNo# , ccy=#ccy#*

1. **写流水**

**insertFundTrans1**

*insert API\_ORDER\_REQ set req\_time= unix\_timestamp(now()),action=4,order\_type=0,acc\_no=#accNo#, prod\_code=#ccy#,buy\_sell=#buyorsell#,amount=#amount#,ref=#remark#,user\_id=#userid#,psw=#psw#, req\_status=2,acc\_order\_no=0,order\_no=0x0,time\_stamp=0.000,ret\_code=0,ret\_msg='OK',ret\_msg2='OK',ret\_msg3='OK'*

**insertFundTrans2**

*insert approval\_req set req\_time=unix\_timestamp(),req\_status=2,appr\_sender=#apprid#,appr\_msg='Approved: From web auto approval',req\_sender=#userid#, req\_type='C',ae\_code=#aeid#,acc\_no=#accNo#,acc\_order\_no=#orderNo#,code=#ccy#,amount=#amount2#,req\_msg=#remark#*

**insertFundTrans3**

*insert API\_ORDER\_REQ set req\_time= unix\_timestamp(now()),action=4,order\_type=0,acc\_no=#accNo#, prod\_code=#ccy#,buy\_sell=#buyorsell#,amount=#amount#,ref=#remark#,user\_id=#userid#,psw=#psw#, req\_status=2,acc\_order\_no=0,order\_no=0x0,time\_stamp=0.000,ret\_code=0,ret\_msg='OK',ret\_msg2='OK',ret\_msg3='OK'*

**insertFundTrans4**

*insert acc\_log set log\_time=unix\_timestamp(),action=4,user\_id=#userid#,acc\_no=#accNo#,acc\_order\_no=0,order\_no=0x0,*

*prod\_code=#ccy#,buy\_sell=#buyorsell#,qty=0,price=floor(#amount#),time\_stamp=0.000,order\_type=0,valid\_type=0,*

*spec\_time=0,stop\_price=0,stop\_type='',sender='',cond\_type=0,up\_level=0,up\_price=0,down\_level=0,down\_price=0, sched\_time=0,traded\_qty=0,ref=#remark#,ref2='',gateway\_code='',status=18,return\_code=0,trade\_no=0,deal\_no=0,clear\_no=0,seq\_no=0,acc\_req\_no=0,open\_close=''*

insertFundTrans5

*insert user\_log set log\_time=unix\_timestamp(),user\_id='@API',acc\_no=#accNo#,acc\_order\_no=0,text='API Order Request Completed: RetCode=0(OK)'*

**insertFundTrans6 *voucher资金流水表***

*insert into* *voucher (trade\_date,time\_stamp,user\_id,acc\_no,voucher\_type,prod\_code,*

*qty,price,ccy,amount,ref)*

*select bus\_date,unix\_timestamp(),#userid#,#accNo#,'C','',0,0,#ccy#,#amount2#,#remark# from* *business\_date where bus\_no=0*

**insertFundTrans7**

*insert user\_log set log\_time=unix\_timestamp(),user\_id=#userid#,acc\_no=#accNo#,acc\_order\_no=0,*

*text=concat('Action=Approve, RetCode=0(No Error), ProdCode=',#ccy#,', Cmd=',#cmd#,', Price=',#amount#,', Qty=0, TradedQty=0, Validity=DAY, Status=Traded, Ref=',#remark#);*

**insertFundTrans8**

*insert acc\_log set log\_time=unix\_timestamp(),action=16,user\_id=#userid#,acc\_no=#accNo#,acc\_order\_no=0,order\_no=0x0, prod\_code=#ccy#,buy\_sell=#buyorsell#,qty=0,price=#amount#,time\_stamp=0.000,order\_type=0,valid\_type=0, spec\_time=0,stop\_price=0,stop\_type='',sender='',cond\_type=0,up\_level=0,up\_price=0,down\_level=0, down\_price=0,sched\_time=0,traded\_qty=0,ref=#remark#,ref2='',gateway\_code='',status=9,return\_code=0,*

*trade\_no=0,deal\_no=0,clear\_no=0,seq\_no=0,acc\_req\_no=0,open\_close=''*

**insertFundTrans9**

*insert reload\_req set reload\_action=11,acc\_no=#accNo#,upd\_time=unix\_timestamp(now()),code='',acc\_order\_no=0,trade\_no=0*

gspmg工程

实现类在hk.guosen.wtm.integration.service.impl.FutServiceImpl.depositWithdrawCash

SQL在FutFundTrans-sqlmap.xml

数据库mysql

172.16.130.17:3306/trading

root/sptrader

MySQL中将秒转换为日期函数

select from\_unixtime(1421638751);

汇率换算算法，以一美元为单位维度方便理解，换算金额\*（目标汇率/源汇率）

public double getExchAmount(String fromCcy,String toCcy,BigDecimal amount) throws ServiceException {

BigDecimal fromRate = new BigDecimal(getCcyRate(fromCcy).getRate());

BigDecimal toRate = new BigDecimal(getCcyRate(toCcy).getRate());

BigDecimal midRate = toRate.divide(fromRate, 4, BigDecimal.ROUND\_HALF\_EVEN); //中间汇率

BigDecimal exchAmount = amount.multiply(midRate);

return exchAmount.doubleValue();

}

以美元为单位的汇率表 ccy\_rate

查写入SP流水问题

select (cash+today\_cash)\*1/7.754 from acc\_bal where acc\_no='9900019712';

select credit\_limit+total\_pl-imargin from acc\_mkt where acc\_no='9900019712' and active='1';

获取佣金及汇率

理论上可以从EBROKER获取佣金数据，但是，从EBROKER获取数据运算较复杂，而营销平台已经从EBROKER采集数据并加工，所以，可从营销平台获取相关数据。营销平台测试环境数据库连接如下：

192.168.230.251 sa/123

查佣金

select \* from kdcc30hist.dbo.crm\_qh\_jy\_exchangerate;

查汇率

select \* from kdcc30hist..crm\_fx2\_qh\_khjyhz （缺数据）

select \* from kdcc30hist.dbo.crm\_jy\_Forexrate