Trader => Broker

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Orders | id | futures | sendtime | arrivetime | amount | type | price | status | traderid |
| market |  | input | Input |  | input | input |  |  | input |
| limit |  | input | Input |  | input | input | input |  | input |
| stop |  | input | Input |  | input | input | input |  | input |
| cancel |  | input |  |  | Id | input |  |  | input |

type: 0:default 1:market buy 2:market sell 3:limit buy 4:limit sell 5:stop buy 6.stop sell 7:cancel

1. order Input

Order（futures,sendtime,amount,type,price,traderid）

* 1. type = 1 or 2

Return = 1(success)

* 1. type = 3,4,5,6

Return = order.Id(该挂单在broker自动递增生成的id)

=-1(设定价格不符合要求)

* 1. type = 7（amount字段设置为要取消的orderid用来通讯）

Return = 剩余amount数量

1. 查询order

Input:traderid

Return:List<order> 如果一条order全部成交则不会搜索到这一条，否则返回的order.amount指剩余手数

1. 查询deal

3.1个人

Input:traderid

Return:List<deal> 只显示买房或买方为该trader的deal

3.2全部

Input:traderid

Return:List<deal> 所有deal，但是与trader无关的deal会屏蔽信息

1. 查询marketdepth

Input：。。

Return:List<Marketdepth>

其中marktdepth为对Market的按商品打包

查Orders（futures=input，type =limit）buy前三sell前写入market表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Market | id | futures | amount | price | level |
|  | int | String | int | double | -3/-2/-1/1/2/3 int |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Deals | id | futures | time | amount | type | price | Buyerid | Sellerid | Buy  /sell  Orderid |
|  |  |  |  |  | Int |  |  |  |  |

1对于market buy（market sell 反之）

查Orders（futures,type=limit sell）得到最小值x

* 1. buy.amount<sell amount 添加并返回deals 修改market 的sell 1
  2. buy.amount=sell amount 添加并返回deals 删除sell 1，从order中查补sell3到market
  3. Buy.amount>sell amount 查找比sell price 大的limit sell重复，stop buy触发为market

查对于

2对于 limit buy

查Orders（futures,type=limit sell）得到最小值x

2.1 buy.price>x 返回交易失败

2.2 buy.price=x

Buy.amount<=sell.amount 同market

Buy.amount>sell.amount 多余的作废，添加dell.amount=sell.amount,修改 market

同时向上查询，如果先遇到stop buy就触发为marketbuy

2.3buy.price<x 修改market

3对于 limit sell

查Orders（futures,type=limit sell）得到最小值x

3.1 order.price>x 返回交易失败

3.2 order.price=x

Buy.amount<=sell.amount 同market

Buy.amount>sell.amount 多余的作废，添加dell.amount=sell.amount,修改 market

同时向上查询，如果先遇到stop buy就触发为marketbuy

3.3buy.price<x 修改market表

4对于 stop buy

查Orders（futures,type=limit sell）得到最小值x

4.1 order.price<x 返回交易失败

4.2 order.price=x 当作market buy处理

4.3order.price>x

5对于 stop sell

查Orders（futures,type=limit buy）得到最大值y

2.1 order.price>y 返回交易失败

2.2 order.price=y 当作market sell处理

2.3order.price<y

6 cancel

仅对order中的pending limit order 删除order条目

？（mq）可对market

Trader => Broker

Input：Traderid 查order

Return：order + deal中所有traderid有关的内容

Input：futures

Return：marketdepth（futures,buy123,buy123price,sell123,sell123price,time）