服务器端第一阶段接口

一. 用户查询接口

用户通过 websocket 发送一个 JSON 格式的请求，服务器返回一个 JSON 响应。

1. 请求

{

"type" : "type of request"

"data" : [parameters about "exCode", "proType", "conCode" or "kType"]

}

请求类型 type:

HET：心跳包。客户端必须定期向服务器传心跳包，否则会被服务器踢掉，具体心跳时间待定。

CON: 查询合约

MAR: 查询行情

EMR: 根据交易所查询行情

KLE: 查询 K 线

SUB: 订阅合约

产品类型 proType**:**

比如原油，国企指数等(在第一次合约请求中会返回给客户端)

K 线类型 kType:

"ONE","THR","FIV","TEN","HAF","SIT","FOH","DAY","WEK","MON";

分别代表 1 3 5 10 30 60 4hours 1day 1week 1month

交易所代码 exCode：

例如“CME”

“ALL”表示所有交易所

合约代码 conCode：

例如“CL1712”

“ALL”表示所有合约

具体的请求格式和返回格式如下(第一阶段采用轮询式访问)

原则: type 类型必须有, data 字段根据 type 的请求视情况传递：

a. **心跳包请求**

{

“type”：“HET”，

“data”: []

}

返回:

{

"state" : 1, //1 表示请求成功，0 表示请求失败

"errorCode" : error_code,

"errorMsg" : "error_massage",

}

b. "type"=="CON"(合约代码查询)

{

“type”：“CON”，

“data”:[]

}

返回数据:

{

"state" : 1, //1 表示请求成功，0 表示请求失败

"errorCode" : error_code,

"errorMsg" : "error_massage",

"data" : {

"交易市场" : {

"交易品种" : [“合约名称”,"品种合约","品种合约",…],

"交易品种" : [“合约名称”,"品种合约","品种合约",…],

……

},

"交易市场" : {

"交易品种" : [“合约名称”,"品种合约","品种合约",…],

"交易品种" : [“合约名称”,"品种合约","品种合约",…],

……

},

……

}

}

c. "type"=="MAR"（市场行情查询 可以查询多个合约）

请求数据

```
{
  “type”：“MAR”，
   "data":[
       [ "exCode1", "conCode1"],
       ["exCode2", "conCode2"],
         ……
     ]
}
```

返回数据：(标红的表示 api 中无法获取到的数据 暂时没管)

```
{
    "state" : 1, //1 表示请求成功，0 表示请求失败
    "errorCode" : error_code,
    "errorMsg" : "error_massage",
    "data" : [
        [交易日, 合约代码, 交易所代码, 最新价, 现手, 持仓量, 上次结算价,
        本次结算价, 涨跌, 涨跌幅, 当日均价, 买量, 卖量, 申买价一, 申买量
        一, 申买价二, 申买量二, 申买价三, 申买量三, 申买价四, 申买量四,
        申买价五, 申买量五, 申卖价一, 申卖量一, 申卖价二, 申卖量二, 申卖
        价三, 申卖量三, 申卖价四, 申卖量四, 申卖价五, 申卖量五],
        [交易日, 合约代码, 交易所代码, 合约在交易所的代码, 最新价, 现手,
        持仓量, 上次结算价, 本次结算价, 涨跌, 涨跌幅, 当日均价, 买量, 卖
        量, 申买价一, 申买量一, 申买价二, 申买量二, 申买价三, 申买量三,
```

申买价四, 申买量四, 申买价五, 申买量五, 申卖价一, 申卖量一, 申卖价二, 申卖量二, 申卖价三, 申卖量三, 申卖价四, 申卖量四, 申卖价五, 申卖量五],

……

]

}

d. “type”=”EMR” (合约的查询 查询涨跌和持有量 新增接口 根据交易所查询该交易所下所有合约的行情)

请求数据

{

“type”: “EMR”,

"data":[“exCode”]

}

返回数据

{

"state" : 1, //1 表示请求成功，0 表示请求失败

"errorCode" : error_code,

"errorMsg" : "error_massage",

"data" :[

[合约代码，最新价，涨跌，涨跌幅，成交量，持仓量]

[合约代码，最新价，涨跌，涨跌幅，成交量，持仓量]

……

]

}

e. “type”=“SUB”（查询订阅合约的信息）

请求数据

{

“type”：“SUB”，

"data":[

["exCode1", "conCode1"],

["exCode2", "conCode2"],

……

]

}

返回数据

{

"state" : 1, //1 表示请求成功，0 表示请求失败

"errorCode" : error_code,

"errorMsg" : "error_massage",

"data" :[

[合约代码，最新价，涨跌，涨跌幅，成交量，持仓量]

[合约代码，最新价，涨跌，涨跌幅，成交量，持仓量]

……

]

}

f. "type"="KLE"(单个合约的 K 线)

请求数据

{

“type”：“KLE”，

"data":[“exCode”,”conCode”,”kType”]

}

返回数据

{

"state" : 1, //1 表示请求成功，0 表示请求失败

"errorCode" : error_code,

"errorMsg" : "error_massage",

"data" : [//30 个点

[时间, 开盘, 最高, 最低, 收盘, 涨跌, 涨跌幅, 成交量]

……

]

}