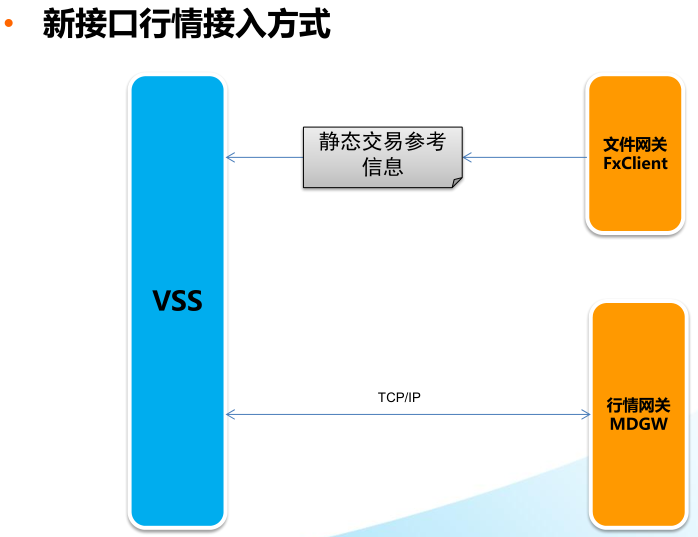
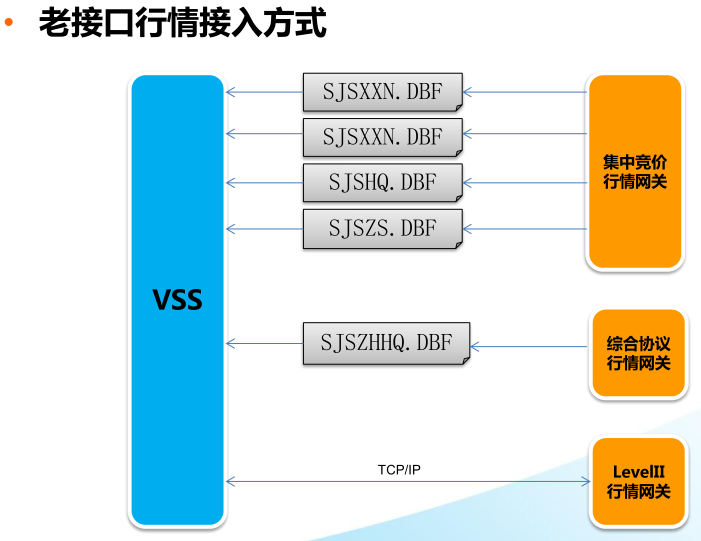
深圳证券交易所第五代交易接口行情改造

|  |  |  |
| --- | --- | --- |
| ***Ver.*** | ***Desc.*** | ***author*** |
| 0.1 | 接口分析、实现思路稿 | 黎伟 |

# 新老行情接入变更方式



# 新老接口文档对应关系

|  |  |
| --- | --- |
| 老接口 | 新接口 |
| 证券信息库SJSXX.DBF | 深圳证券交易所Binary行情数据接口规范1.00α.pdf 深圳证券交易所STEP行情数据接口规范1.00α.pdf 深圳证券交易所数据文件交换接口规范1.00α.pdf |
| 新证券信息库SJSXXN.DBF |
| 行情库SJSHQ.DBF |
| 指数库SJSZS.DBF |
| 综合行情库SJSZHHQ.DBF |
| 信息公告文件XXYYMMDD.NNN |

# 证券信息与新接口字段对应关系

|  |  |  |
| --- | --- | --- |
| **StkInfo属性** | **描述** | **新接口字段** |
| exchId | 市场代码 |  |
| stkId | 证券代码 | 3.1 SecurityID |
| stkName | 证券名称 | 3.1 Symbol |
| stkEnglishAbbr | 英文简称 | 3.1 EnglishName |
| tradeUnit | 交易单位 |  |
| stkIndustryType | 行业种类 | 3.1 IndustryClassification |
| stkParValue | 每股面值 | 3.1 ParValue |
| totalCurrentstkQty | 流通股数 | 3.1 PublicFloatShareQuantity |
| lastYearProfit | 上年利润 | 3.1 PreviousYearProfitPerShare |
| thisYearProfit | 本年利润 | 3.1 CurrentYearProfitPerShare |
| listedDate | 上市日期 | 3.1 ListDate |
| endingDate | 到期/交割日 | 3.1 MaturityDate |
| maxOrderQty | 每笔限量 |  |
| orderPriceUnit | 价格档位 | 3.4 PriceTick |
| aggPriceLimit | 集合竞价限价参数 |  |
| contPriceLimit | 连续竞价限价参数 |  |
| priceLimitFlag | 限价参数性质 |  |
| maxOrderPrice | 涨停价格 | 由动态行情回写 |
| minOrderPrice | 跌停价格 | 由动态行情回写 |
| standardConvertRate | 折合比例 | 3.1 ContractMultiplier注意，APP对此字段有特殊处理。 |
| tradeStatus | 交易状态 | 3.1 Status |
| stkLevel | 证券级别 |  |
| closeFlag | 停牌标志 | 由动态行情回写注意，APP会根据暂停交易标志有特殊处理 |
| stkAllotFlag | 除权除息标志 |  |
| stkIndexFlag | 成分股标志 |  |
| closePrice | 昨日收盘价 | 3.1 PrevClosePx注意，此字段对于深圳基金有特殊处理。 |
| exchTotalKnockQty | 成交数量 | 4.1 TotalVolumeTrade |
| exchTotalKnockAmt | 成交金额 | 4.1 TotalValueTrade |
| accuredInterest | 百元利息额 | 3.1 Interest |
| totalStkQty | 总股本 | 3.1 OutstandingShare |
| stkIdPrefix | 证券简称前缀 | 由动态行情回写 |
| basicStkId | 基础证券 | 3.1 UnderlyingSecurityID |
| ISINCode | ISIN编码 | 3.1 ISIN |
| NAV | 基金份额累计净值 | 3.1 NAV |
| beginInterestDate | 债券起息日 | 3.1 InterestAccrualDate |
| bulkTradingMaxPrice | 大宗交易价格上限 |  |
| bulkTradingMinPrice | 大宗交易价格下限 |  |
| CMOStandardRate | 担保折扣率 |  |
| isCreditCashStk | 融资标的标志 | 3.1 CrdBuyUnderlying |
| isCreditShareStk | 融券标的标志 | 3.1 CrdSellUnderlying |
| marketMarkerFlag | 做市商标志 | 3.4 MarketMakerFlag |
| exchTradeType | 交易类型 |  |
| tradeTimeFlag | 产品交易阶段 | 由动态行情回写 |
| pauseTradeStatus | 暂停交易标志 |  |
| passCreditCashStk | 融资交易状态 | 由动态行情回写 |
| passCreditShareStk | 融券交易状态 | 由动态行情回写 |
| creditShareSellPriceFlag | 融券卖出价格限制 |  |
| netVoteFlag | 网络投票标志 |  |
| otherBusinessMark | 其他业务状态 |  |
| exchMemo | 备用字段 |  |
| newPrice | 当前价 | 由动态行情回写。注意，此字段对于深圳基金有特殊处理。 |
| highPrice | 最高 | 由动态行情回写 |
| lowPrice | 最低 | 由动态行情回写 |

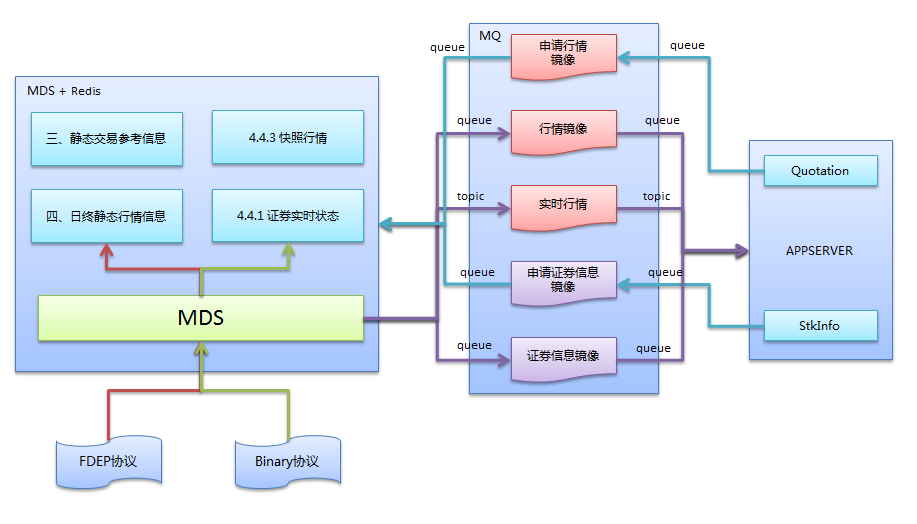
* Note1：参照文档：《深圳证券交易所数据文件交换接口规范1.00α.pdf》
* Note2：APPSERVER证券信息刷新逻辑，参考StockInfoAgent.processRecord(int, byte[], short, TransactionManager)

# 动态行情与新接口字段对应关系

|  |  |  |
| --- | --- | --- |
| **Quotation属性** | **描述** | **新接口字段** |
| exchId | 市场 |  |
| stkId | 证券代码 | 4.4.1 SecurityID |
| stkIdPrefix | 证券简称前缀 | 4.4.1 SecurityPreName |
| stkName | 证券名称 |  |
| levelFlag | 行情标志 |  |
| closeFlag | 停牌标志 | 4.4.1 SecurityPreName 第一位；APP根据暂停交易标志有特殊处理 |
| closePrice | 昨收 | 4.4.3 PrevClosePx；对于深圳基金有特殊处理； |
| openPrice | 今开 | 4.4.3.1 4-开盘价 |
| newPrice | 当前价 | 4.4.3.1 2-最近价；对于深圳基金有特殊处理； |
| highPrice | 最高 | 4.4.3.1 7-最高价 |
| lowPrice | 最低 | 4.4.3.1 8-最低价 |
| knockQty | 成交量 | 4.4.3 TotalVolumeTrade |
| knockMoney | 成交金额 | 4.4.4 TotalValueTrade |
| buyPrice1 | 买1 | 4.4.3.1 0-买+MDPriceLevel+MDEntryPx |
| buyPrice2 | 买2 | 4.4.3.1 0-买+MDPriceLevel+MDEntryPx |
| buyPrice3 | 买3 | 4.4.3.1 0-买+MDPriceLevel+MDEntryPx |
| buyPrice4 | 买4 | 4.4.3.1 0-买+MDPriceLevel+MDEntryPx |
| buyPrice5 | 买5 | 4.4.3.1 0-买+MDPriceLevel+MDEntryPx |
| sellPrice1 | 卖1 | 4.4.3.1 1-卖+MDPriceLevel+MDEntryPx |
| sellPrice2 | 卖2 | 4.4.3.1 1-卖+MDPriceLevel+MDEntryPx |
| sellPrice3 | 卖3 | 4.4.3.1 1-卖+MDPriceLevel+MDEntryPx |
| sellPrice4 | 卖4 | 4.4.3.1 1-卖+MDPriceLevel+MDEntryPx |
| sellPrice5 | 卖5 | 4.4.3.1 1-卖+MDPriceLevel+MDEntryPx |
| buyQty1 | 买量1 | 4.4.3.1 0-买+MDPriceLevel+MDEntrySize |
| buyQty2 | 买量2 | 4.4.3.1 0-买+MDPriceLevel+MDEntrySize |
| buyQty3 | 买量3 | 4.4.3.1 0-买+MDPriceLevel+MDEntrySize |
| buyQty4 | 买量4 | 4.4.3.1 0-买+MDPriceLevel+MDEntrySize |
| buyQty5 | 买量5 | 4.4.3.1 0-买+MDPriceLevel+MDEntrySize |
| sellQty1 | 卖量1 | 4.4.3.1 1-卖+MDPriceLevel+MDEntrySize |
| sellQty2 | 卖量2 | 4.4.3.1 1-卖+MDPriceLevel+MDEntrySize |
| sellQty3 | 卖量3 | 4.4.3.1 1-卖+MDPriceLevel+MDEntrySize |
| sellQty4 | 卖量4 | 4.4.3.1 1-卖+MDPriceLevel+MDEntrySize |
| sellQty5 | 卖量5 | 4.4.3.1 1-卖+MDPriceLevel+MDEntrySize |
| IOPV | 基金净值（市盈率2） | 4.4.3.1 x8-基金实时参考净值；对于债券ETF在APP有特殊处理 |
| maxOrderPrice | 涨停价格 | 4.4.3.1 xe-涨停价格 |
| minOrderPrice | 跌停价格 | 4.4.3.1 xf-跌停价格 |
| tradeTimeFlag | 产品交易阶段 | 4.4.3 TradingPhaseCode |
| pauseTradeStatus | 暂停交易标志 | 4.4.1 SecurityPreName 第一位 H |
| passCreditCashStk | 融资交易状态 | 4.4.1 SecurityPreName 第二位 |
| passCreditShareStk | 融券交易状态 | 4.4.1 SecurityPreName 第二位 |
| otherBusinessMark | 其他业务状态 |  |
| change | 价格变动 | APP根据newPrice和closePrice进行计算 |
| changePecent | 价格变动率 | APP根据newPrice和closePrice进行计算 |
| closeIndex | 昨收指数 | 4.4.3.3 xa |
| openIndex | 今开指数 | 4.4.3.3 xb |
| highIndex | 最高指数 | 4.4.3.3 xc |
| lowIndex | 最低指数 | 4.4.3.3 xd |
| newIndex | 当前指数 | 4.4.3.3 3 |
| changeTime | 行情时间 | 4.4.1 OrigTime，交易所时间 |
| ReceiveTime | MDS收到行情的时间 | 精确到毫秒，考察延时指标时使用 |
| SendTime | MDS将行情推到MQ的时间 | 精确到毫秒，考察延时指标时使用 |

* Note1：参照文档：《深圳证券交易所Binary行情数据接口规范1.00α.pdf》
* Note2：APPSERVER行情信息刷新逻辑，参考Quotation.setMQQuotationData(String[])
* Not3： APPSERVER指数行情刷新逻辑，参考Index.setMQIndexData(String[])
* Note3：实时行情刷新时，部分字段要回写到证券信息内存，参见：
  + StockManager.setStkInfoBySJXXN(IQuotation)
  + StockManager.setStkInfoBySJXX(IQuotation)

# 实现思路



## MQ

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 消息主题 | 主题描述 | 消息类型 | 消息地址 | 消息生产者 | 消息属性 | 消息消费者 |
| SZ5\_REQ\_Quotation | 申请行情镜像 | queue | 1503-ActiveMQURI-3 | APPSERVER | clientId | MDS |
| SZ5\_Quotation\_Image | 行情镜像 | queue | 1503-ActiveMQURI-3 | MDS | clientId | APPSERVER |
| SZ5\_Quotation\_RealTime | 实时行情 | topic | 1503-ActiveMQURI-3 | MDS |  | APPSERVER |
| SZ5\_REQ\_StkInfo | 申请证券信息 | queue | 1503-ActiveMQURI-3 | APPSERVER | clientId | MDS |
| SZ5\_StkInfo\_Image | 证券信息镜像 | queue | 1503-ActiveMQURI-3 | MDS | clientId | APPSERVER |

* Note1：申请行情镜像、申请证券信息，采用持久性消息，避免MQ重启后申请丢失的问题。
* Note2：行情镜像、实时行情、证券信息镜像，采用非永久性消息，过期（2秒）无人消费自动作废。
* Note3：行情镜像、实时行情、证券信息镜像传输格式先暂定json格式（支持压缩）。

## MDS+Redis部分

### 存储

1. MDS根据FDEP协议方式接收深5代静态交易信息；
   1. 参考文档：《深圳证券交易所数据文件交换接口规范1.00α.pdf》
   2. 在Redis中为每个静态交易参考文件构建内存，内存存储的对象属性参考每个文件的接口定义

|  |  |
| --- | --- |
| Securities | 证券信息 |
| IndexInfo | 指数信息 |
| Stat | 统计量指标信息 |
| AuctionParams | 集中竞价交易类业务参考信息 |
| NegotiationParams | 协议交易业务参考信息 |
| AfterhoursParams | 盘后定价大宗交易业务参考信息 |
| SecuritLending | 转融通证券出借业务参考信息 |
| Etfcr | ETF 实时申购赎回业务参考信息（没有提供文件格式） |
| IssueParams | 网上发行认购业务参考信息 |
| BondDistributionParams | 债券分销业务参考信息 |
| EvoteParams | 网络投票业务参考信息 |
| RightsissueParams | 配股业务参考信息 |
| SecurityCloseMD | 证券收盘行情 |

* 1. 文件日期变更时，将当前文件内存异步转移到其他持久化存储归档保存后，重新构建当日文件内存。
  2. MDS重启时，将静态行情从文件重新加载到内存中。

1. MDS根据Binary协议方式获取深5代动态行情信息；
   1. 参考文档：《深圳证券交易所数据文件交换接口规范1.00α.pdf》
   2. 在Redis中为构建动态行情快照内存，内存存储的对象属性参考每个文件的接口定义。

|  |  |
| --- | --- |
| SecurityStatus | 证券实时状态（4.4.1） |
| Quotation | 行情快照（4.4.3） |

Note1：不同业务的MDEntryType是没有重复的，理论上可以将重复组合并成一个条单记录。

* 1. 实时行情变更时，将旧行情内存异步转移到其他持久化存储进行归档；
  2. MDS启动时，从持久化存储中加载最近的一条行情信息；

### 发布

1. MDS启动时与MQ建立连接；订阅以下MQ主题；

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 消息主题 | 主题描述 | 消息类型 | 消息生产者 | 消息属性 | 消息消费者 |
| SZ5\_REQ\_Quotation | 申请行情镜像 | queue | APPSERVER | clientId | **MDS** |
| SZ5\_Quotation\_Image | 行情镜像 | queue | **MDS** | clientId | APPSERVER |
| SZ5\_Quotation\_RealTime | 实时行情 | topic | **MDS** |  | APPSERVER |
| SZ5\_REQ\_StkInfo | 申请证券信息 | queue | APPSERVER | clientId | **MDS** |
| SZ5\_StkInfo\_Image | 证券信息镜像 | queue | **MDS** | clientId | APPSERVER |

1. MDS从SZ5\_REQ\_Quotation接收外部系统的行情镜像请求后，从Redis存储中获取全部深圳市场的行情数据，单笔行情的字段参照“4 动态行情与新接口字段对应关系”进行构建，采用json格式，1000条一个包，进行压缩后从SZ5\_Quotation\_Image主题返回，消息属性填写外部系统请求镜像时发入的clientId，即“谁申请镜像，镜像返回给谁”。
2. 当MDS接收到新的动态行情后，单笔行情的字段参照“4 动态行情与新接口字段对应关系”进行构建，采用json格式，进行压缩后从SZ5\_Quotation\_RealTime主题返回，topic方式，消息有效期2秒，无人消费自动作废。同一时点有多条行情变更时，应尽量合并在一个消息包中进行压缩后发出。
3. MDS从SZ5\_REQ\_StkInfo接收外部系统的证券信息镜像请求后，从Redis存储中获取全部深圳市场的证券信息数据，单笔证券信息字段参照“3证券信息与新接口字段对应关系”进行构建，采用json格式，1000条一个包，进行压缩后从SZ5\_StkInfo\_Image主题返回，消息属性填写外部系统请求镜像时发入的clientId，即“谁申请镜像，镜像返回给谁”。

## APPSERVER

1. APPSERVER启动时与MQ建立连接；订阅以下MQ主题；从StkInfo表构建初始证券信息内存；

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 消息主题 | 主题描述 | 消息类型 | 消息生产者 | 消息属性 | 消息消费者 |
| SZ5\_REQ\_Quotation | 申请行情镜像 | queue | **APPSERVER** | clientId | MDS |
| SZ5\_Quotation\_Image | 行情镜像 | queue | MDS | clientId | **APPSERVER** |
| SZ5\_Quotation\_RealTime | 实时行情 | topic | MDS |  | **APPSERVER** |
| SZ5\_REQ\_StkInfo | 申请证券信息 | queue | **APPSERVER** | clientId | MDS |
| SZ5\_StkInfo\_Image | 证券信息镜像 | queue | MDS | clientId | **APPSERVER** |

1. APPSERVER启动时，向SZ5\_REQ\_Quotation主题申请行情镜像，并将APPSERVER自身的IP\_PORT作为clientId消息属性发送给MDS；需要考虑增加一个手动申请深圳行情镜像的机制，便于故障时重新获取市场行情镜像。
2. APPSERVER同时从SZ5\_Quotation\_Image主题和SZ5\_Quotation\_RealTime主题接收行情镜像和实时行情；可以考虑多线程的方式更新实时行情内存，其中以行情的时间戳为准，如果行情时间戳早于行情内存中的时间戳则直接丢弃。行情处理逻辑参考Quotation.setMQQuotationData(String[])和Index.setMQIndexData(String[])。
3. APPSERVER仍从原有证券信息定时刷新、证券信息手工刷新的出口，向MDS发送SZ5\_REQ\_StkInfo申请证券信息镜像的请求，并将APPSERVER自身的IP\_PORT作为clientId消息属性发送给MDS。
4. APPSERVER监听MQ的SZ5\_StkInfo\_Image主题，将收到的证券信息镜像处理到系统中，原证券信息刷新逻辑参考StockInfoAgent.processRecord(int, byte[], short, transMgr)方法。
5. 多APPSERVER场景下，只有一个APPSERVER具有写入stkinfo的机制。

# 其他

1. MQ版本号5.5.0，C# MQ组件版本号：Apache.NMS.ActiveMQ.dll 1.5.6、Apache.NMS.dll 1.5.1；
2. APPSERVER需要检查一下多域证券信息和动态行情处理中的个性化逻辑，理论上这些逻辑应该移植到MDS上统一处理，这样对于其他客户端直接消费行情即可，不需要再额外加工。