

# 华锐 AMD 行情传输平台

## API (AMA) 开发指南

AMD 产品部

2023. 03



# CONTENTS

## 目录

## 目录

一、 引言.....	14
1. 文档目的.....	14
2. 术语和缩略语.....	14
二、 文件说明.....	14
三、 参数及精度说明.....	15
1. 参数影响说明.....	15
1.1. channel_mode 参数说明.....	15
1.2. tcp_compress_mode 参数说明.....	15
1.3. ha_mode 参数说明.....	15
1.4. min_log_level 参数说明.....	15
1.5. is_output_mon_data 参数说明.....	16
1.6. keep_order 与 keep_order_timeout_ms 参数说明.....	16
1.7. is_subscribe_full 参数说明.....	17

1.8. is_thread_safe 参数说明.....	17
1.9. 委托簿参数说明.....	17
2. 精度说明.....	18
3. 产品数量数据单位说明(灰色代表无法确定品种单位).....	18
四、 兼容性说明.....	19
1. AMA 与服务端系统 AMD 的兼容性.....	19
2. AMA 对于操作系统的兼容性.....	19
3. AMA 接口开发语言的支持.....	19
3.1. 支持的语言接口.....	19
3.2. 接口效率对比说明.....	19
五、 各通道说明.....	20
1. TCP 通道模式.....	20
1.1. 数据接入方式.....	20
1.2. 关键参数设置.....	20
1.3. 数据过滤方式.....	20
1.4. 异常情况说明.....	20
2. AMI 通道模式说明.....	21
2.1. 数据接入方式.....	21
2.2. 关键参数设置.....	21
2.3. 数据过滤方式.....	21
3. DAS 抓包通道模式说明.....	21
3.1. 数据接入方式.....	21

3.2. 关键参数设置.....	21
3.3. 数据过滤方式.....	22
4. MDDP 通道模式说明.....	22
4.1. 数据接入方式.....	22
4.2. 关键参数设置.....	22
4.3. 数据过滤方式.....	22
5. FPGA 通道模式说明.....	22
5.1. 数据接入方式.....	22
5.2. 关键参数设置.....	22
5.3. 数据过滤方式.....	22
5.4. 本地网卡选择方式.....	22
6. UDP 通道模式说明.....	23
6.1. 数据接入方式.....	23
6.2. 关键参数设置.....	23
6.3. 数据过滤方式.....	23
六、 C++开发接口.....	23
1. IAMDApi 接口.....	23
1.1. GetVersion 方法.....	23
1.2. Init 方法.....	24
1.3. Join 方法.....	24
1.4. Release 方法.....	24
1.5. FreeMemory 方法.....	25

1.6. SubscribeData 方法.....	25
1.7. GetCodeTableList 方法.....	30
1.8. GetETFCodeTableList 方法.....	30
1.9. SubscribeOrderBookData 方法.....	31
1.10. SubscribeDerivedData 方法.....	32
1.11. GetIMCExchangeRate 方法.....	32
2. IAMDSpi 接口.....	33
2.1. OnLog 方法.....	33
2.2. OnIndicator 方法.....	33
2.3. OnEvent 方法.....	35
2.4. OnMDSnapshot 方法.....	36
2.5. OnMDOptionSnapshot 方法.....	36
2.6. OnMDHKTSnapshot 方法.....	36
2.7. OnMDIndexSnapshot 方法.....	37
2.8. OnMDTickOrder 方法.....	37
2.9. OnMDTickExecution 方法.....	37
2.10. OnMDOOrderQueue 方法.....	38
2.11. OnMDAfterHourFixedPriceSnapshot 方法.....	38
2.12. OnMDAfterHourFixedPriceTickExecution 方法.....	38
2.13. OnMDFutureSnapshot 方法.....	39
2.14. OnMDCSIIIndexSnapshot 方法.....	39
2.15. OnMDIndicatorOfTradingVolumeSnapshot 方法.....	39

2.16. OnMDCnIndexSnapshot 方法.....	40
2.17. OnMDRefinancingTickOrder 方法.....	40
2.18. OnMDRefinancingTickExecution 方法.....	40
2.19. OnMDNegotiableTickOrder 方法.....	41
2.20. OnMDNegotiableTickExecution 方法.....	41
2.21. OnMDHKTRealtimeLimit 方法.....	41
2.22. OnMDHKTProductStatus 方法.....	42
2.23. OnMDHKTVCM 方法.....	42
2.24. OnMDNEEQSnapshot 方法.....	42
2.25. OnMDNEEQSecurityInfo 方法.....	43
2.26. OnMDNEEQNonPublicTransDeclaredInfo 方法.....	43
2.27. OnMDNEEQHierarchicalInfo 方法.....	43
2.28. OnMDHKMarketStatus 方法.....	44
2.29. OnMDNEEQNegotiableDeclaredInfo 方法.....	44
2.30. OnMDNEEQMarketMakerDeclaredInfo 方法.....	44
2.31. OnMDNEEQNonPublicTransferDealInfo 方法.....	45
2.32. OnMDOrderBook 方法.....	45
2.33. OnMDBondSnapshot 方法.....	45
2.34. OnMDBondTickOrder 方法.....	45
2.35. OnMDBondTickExecution 方法.....	46
2.36. OnMDBondQuotedTickOrder 方法.....	46
2.37. OnMDBondQuotedTickExecution 方法.....	46

2.38. OnMDOrderBookSnapshot 方法.....	47
2.39. OnMDFundExpertSnapshot 方法.....	47
2.40. OnMDIOPVSnapshot 方法.....	47
2.41. OnMDZQQDBJSnapshot 方法.....	48
2.42. OnMDZQCJHQSsnapshot 方法.....	48
2.43. OnMDZQCJMXSnapshot 方法.....	48
3. *HKExSpi 接口.....	49
3.1. *OnMDHKEExSnapshot 方法.....	49
3.2. *OnMDHKEExOrderSnapshot 方法.....	49
3.3. *OnMDHKEExOrderBrokerSnapshot 方法.....	49
3.4. *OnMDHKEExTickExecution 方法.....	50
3.5. *OnMDHKEExIndexSnapshot 方法.....	50
3.6. *OnMDHKEExConnectTurnoverSnapshot 方法.....	50
4. *UsaSpi 接口.....	51
4.1. *OnMDUsaStockSnapshot 方法.....	51
4.2. *OnMDUsaOrderSnapshot 方法.....	51
4.3. *OnMDUsaEodSnapshot 方法.....	51
5. 工具函数说明.....	52
5.1. ConvertTradingPhaseCode 方法.....	52
6. 接口线程安全说明.....	52
7. 行情数据结构字典.....	55
7.1. 沪、深、北、港股通、期货数据字典.....	55



7.2. *商业港股数据字典.....	97
7.3. *美股数据字典.....	100
8. 公共数据字典.....	103
8.1. 通道类型(ChannelMode).....	103
8.2. 高可用模式类型(HighAvailableMode).....	103
8.3. 日志输出级别(LogLevel).....	103
8.4. 委托簿构建方式(OrderBookType).....	104
8.5. 事件级别(EventLevel).....	104
8.6. 市场类型(MarketType).....	104
8.7. 品种类型(VarietyCategory).....	105
8.8. 错误码(ErrorCode).....	105
8.9. 订阅类型(SubscribeType).....	105
8.10. 权限订阅数据类型(SubscribeDataType).....	106
8.11. 证券数据类型(SubscribeSecuDataType).....	107
8.12. 证券品种类型(SubscribeCategoryType).....	107
8.13. 事件代码(EventCode).....	107
8.14. 证券子类别(Security_type).....	109
8.15. 币种(Currency).....	111
8.16. 产品状态标志(Security_status).....	111
8.17. 委托簿订阅数据类型(SubscribeOrderBookDataType).....	112
8.18. 行情衍生数据订阅类型(SubscribeDerivedDataType).....	112
8.19. 华锐自定义交易阶段(TradingPhaseCode).....	112



七、 附录.....	113
1. 权限数据类型订阅(灰色标识代表不支持此项数据).....	113
2. 品种数据类型订阅(灰色标识代表不支持此项数据).....	117

修订历史			
日期	版本	AMA 版本号	修订说明
2019.9	0.1		创建
2020.03	0.2		补充精度,出入参说明及缺失的市场行情
2020.06	1.0	3.3.0	基于 V3.3 版本完善该文档
2020.06	1.1	3.4.0	基于 V3.4 版本完善该文档
2020.09	1.2.0	3.5.0	1. MDSnapshot 快照数据结构新增"当前品种交易状态"(instrument_status)、"基金 T-1 日收盘时刻 IOPV" (pre_close_iopv)、"债券加权平均委买价格" (alt_weighted_avg_bid_price)、"债券加权平均委卖价格" (alt_weighted_avg_offer_price)、"ETF 申购笔数" (etf_buy_number)、"ETF 申购数量"(etf_buy_amount)、"ETF 申购金额" (etf_buy_money)、"ETF 赎回笔数"(etf_sell_number)、"ETF 赎回数量" (etf_sell_amount)、"ETF 赎回金额"(etf_sell_money)、"权证执行的总数量" (total_warrant_exec_volume)、"债券质押式回购品种加权" (war_lower_price)、"权证涨停价格"(war_upper_price)、"买入撤单笔数" (withdraw_buy_number)、"买入撤单数量"(withdraw_buy_amount)、"买入撤单金额"(withdraw_buy_money)、"卖出撤单笔数" (withdraw_sell_number)、"卖出撤单数量"(withdraw_sell_amount)、"卖出撤单金额"(withdraw_sell_money)、"买入总笔数"(total_bid_number)、"卖出总笔数"(total_offer_number)、"买入委托成交最大等待时间" (bid_trade_max_duration)、"卖出委托成交最大等待时间" (offer_trade_max_duration)、"买方委托价位数"(num_bid_orders)、"卖方委托价位数"(num_offer_orders)。
2020.10	1.2.1	3.5.1	1. 修改了 MDFutureSnapshot 的"合约代码"(security_code)字段长度,由 16 位改成了 32 位。
2020.11	1.3.0	3.6.0	1. 调整了 MDHKTSnapshot 港股通快照行情结构中的"申买价"(bid_price)、"申买量"(bid_volume)、"申卖价"(offer_price)、"申卖量"(offer_volume)的档位,由原来的 1 档调成了 5 档。 2. 增加了 OnMDHKTVCM 港股通 VCM 数据回调接口以及 MDHKTVCM 数据结构。
2021.01	1.4.0	3.7.0	1. 增加 OnMDNEEQSecurityInfo 股转系统证券信息数据回调接口以及 MDNEEQSecurityInfo 数据结构。 2. 增加 OnMDNEEQNonPublicTransDeclaredInfo 股转系统非公开申报转让信息数据回调接口以及 MDNEEQNonPublicTransDeclaredInfo 数据结构。

			<p>2. 增加 OnMDNEEQHierarchicalInfo 股转系统分层信息数据回调接口以及 MDNEEQHierarchicalInfo 数据结构。</p> <p>3. MDSnapshot 快照数据结构新增"最近成交时间"(last_trade_time)。</p> <p>4. MDOptionSnapshot 期权快照数据结构新增"最近成交时间"(last_trade_time)和"参考价"(ref_price)。</p> <p>5. OnEvent 接口中 EventCode 结构体增加行情数据升降级通知消息。</p>
2021.3	1.5.0	3.8.0	<p>1. 更新了 Cfg 结构体定义: 1)"逐笔保序时间"(keep_order_timeout_ms)的单位调整成为毫秒, 2)"数据压缩标志"tcp_compress_mode 数据压缩标志只有 TCP 模式 3) 删除了"线程工作模式"(polling)和"线程处理工作队列长度"(queue_size)以及各通道的独立扩展配置。</p> <p>2. MDTickOrder 现货逐笔委托数据结构体中新增"原始订单号"(orig_order_no)和"业务序号"(biz_index)。</p> <p>3. MDTickExecution 现货逐笔成交数据结构体中新增"业务序号"(biz_index)。</p>
2021.4	1.6.0	3.9.0	<p>1. 新增按市场、证券数据类型、证券品种类型、代码订阅接口(SubscribeData)。</p> <p>2. 新增获取代码表接口(GetCodeTableList)。</p> <p>3. 期权(MDOptionSnapshot)期货(MDFutureSnapshot)结构体新增字段, 具体参考对应结构体。</p> <p>4. 所有结构体增加品种类型字段(variety_category)。</p> <p>5. 新增委托簿(目前只有 RedHat 系统的 c++和 java 接口支持)。</p>
2021.6	1.6.1	3.9.1	<p>1. 新增上海委托簿支持,同时支持新增 python 版本委托簿支持。</p> <p>2. 新增支持数据结构 MDNEEQNegotiableDeclaredInfo。</p> <p>3. 新增支持数据结构 MDNEEQMarketMakerDeclaredInfo。</p> <p>4. 新增支持数据结构 MDNEEQNonPublicTransferDealInfo。</p> <p>5. 新增支持数据结构 MDHKMarketStatus。</p> <p>6. 数据结构 MDHKTRealtimeLimit 新增字段 mkt_status。</p>
2021.8	1.6.3	3.9.3	<p>1. 新增支持债券快照数据结构 MDBondSnapshot。</p> <p>2. 新增支持债券逐笔委托数据结构 MDBondTickOrder。</p> <p>3. 新增支持债券逐笔成交数据结 MDBondTickExecution。</p> <p>4. 新增支持债券业务报价及大额逐笔委托数据结构 MDBondQuotedTickOrder。</p> <p>5. 新增支持债券业务报价及大额逐笔成交数据结构 MDBondQuotedTickExecution。</p> <p>6. 委托簿新增支持盘中启动。</p>
2021.11	1.6.4	3.9.4	<p>1. 新增获取 ETF 代码表接口(GetETFCodeTableList)。</p> <p>2. 优化了代码表接口(GetCodeTableList),能提供更加丰富的证券信息, 其中证券子类别 security_type 字段的含义进行了重新映射。</p> <p>3. MDBondQuotedTickOrder 债券业务报价及大额逐笔委托数据结构新</p>

			<p>增"secondary_order_id"(竞买场次编号)、"bid_trans_type"(竞买业务类别)、"bid_execinst_type"(竞买成交方式)、"lowlimit_price"(价格下限)、"highlimit_price"(价格上限)、"min_qty"(最低成交数量)、"trade_date"(交易日期)字段。</p> <p>4. MDBondQuotedTickExecution 债券业务报价及大额逐笔成交数据结构新增"secondary_order_id"(竞买场次编号)、"bid_execinst_type"(竞买成交方式)、"margin_price"(达成成交的边际价格)字段。</p> <p>5. 新增释放 ETF 代码表内存的 FreeMemory 方法。</p>
2021.12	1.6.5	3.9.5	<p>1. "产品实时阶段及标志"统一描述为"交易阶段代码"。</p>
2022.01	1.6.6	3.9.6	<p>1. 新增委托簿数据订阅接口 SubscribeOrderBookData。</p> <p>2. 委托簿数据新增四个字段: 基于委托簿演算的成交总笔数(total_num_trades)、基于委托簿演算的成交总量(total_volume_trade)、基于委托簿演算的成交总金额(total_value_trade)、基于委托簿演算的最新价(last_price)。</p> <p>3. 新增委托簿快照数据结构 OnMDOrderBookSnapshot。</p> <p>4. Cfg 配置定义的 enable_order_book 参数由布尔类型转换成 unsigned char 类型。</p>
2022.02	1.6.7	3.9.7	<p>1. 代码表新增上市日期(list_day)、面值(par_value)、总发行量(outstanding_share)、流通股数(public_float_share_quantity)、对回购标准券折算率(contract_multiplier)、对应回购标准券(regular_share)字段。</p> <p>2. 代码表接口新增订阅功能。</p> <p>3. 新增基金通快照行情数据结构 MDFundExpertSnapshot。</p> <p>4. 新增权限订阅基金通快照数据结构 kFundExpertSnapshot。</p>
2022.04	1.6.8	3.9.8	<p>1. 新增 FPGA 通道方式获取数据。</p> <p>2. 新增行情衍生数据订阅接口(SubscribeDerivedData)。</p> <p>3. 新增 IOPV 数据接口(OnMDIOPVSnapshot)及 IOPV 快照数据结构(MDIOPVSnapshot)。</p> <p>4. 新增交易阶段代码转换函数(ConvertTradingPhaseCode)以及对应的华锐自定义阶段代码定义(TradingPhaseCode)。</p> <p>5. 代码表结构新增"interest"(应计利息)、"coupon_rate"(票面年利率)字段。</p>
2022.06	1.7.0	4.0.0	<p>1. 新增商业港股数据回调基类 HKExSpi,支持接收商业港股行情数据,新增商业港股股票快照数据回调(MDHKEExSnapshot),商业港股委托挂单数据回调(MDHKEExOrderSnapshot),商业港股经纪席位数据回调(MDHKEExOrderBrokerSnapshot),商业港股逐笔成交数据回调(MDHKEExTickExecution),商业港股指数行情快照数据回调(MDHKEExIndexSnapshot)。</p> <p>2.代码表中币种(currency)字段补充枚举值类别:新台币(TWD)和其他(Other)。</p>
2022.08	1.7.1	4.0.1	<p>1.新增 GetIMCExchangeRate 接口,用于获取国际市场汇率数据。新增国际市场汇率数据结构(IMCExchangeRate)。</p> <p>2.MDTickOrder 新增 traded_order_volume 字段。</p>

2022.09	1.8.0	4.1.0	<p>1. 新增 NASDAQ 美股行情数据回调基类 UsaSpi,支持接收 NASDAQ 美股行情数据,新增美股股票快照数据回调(OnMDUsaStockSnapshot),美股挂单快照数据回调(OnMDUsaOrderSnapshot),美股 EOD 收盘快照数据回调(OnMDUsaEodSnapshot)。</p> <p>2. 代码表新增支持获取期货代码, 新增期货品种产品代码(product_code)、交割年份(delivery_year)、交割月份(delivery_month)、创建日期(create_date)、开始交割日(start_deliv_date)、结束交割日(end_deliv_date)、持仓类型(position_type)字段。</p>
2022.10	1.8.1	4.1.1	<p>增加支持通过 UDP 组播接入 UDF 转码数据。</p>
2023.03	1.9.1	4.2.0	<p>1. 支持接收上海固定收益行情数据,新增确定报价固收行情数据(MDZQQDBJSnapshot),成交行情固收行情数据(MDZQCJHQSnapshot),成交明细固收行情数据(MDZQCJMXSnapshot)。</p> <p>2.ETF 代码表成分股的成分证券代码长度(security_code)字段扩位至 32 位; 支持启用成份证券所属市场 ID(underlying_security_id)字段。</p> <p>3. 商业港股委托挂单数据结构买单序列(bid_list)、卖单序列(ask_list)字段长度由 10 档增加到 40 档; 新增商业港股港股通成交额快照数据(MDHKExConnectTurnoverSnapshot)。</p>

# 一、引言

## 1. 文档目的

华锐 AMD 高速行情传输平台是华锐技术推出的低时延、高吞吐、高可用的行情分发平台。行情消费者可以通过华锐高速行情转码 API(AMA)对接 AMD,接收转码行情。

本文档是 AMA 的开发指南,包含了对 API 接口的说明以及示例,用于指引行情接收系统的开发人员基于此 AMA 进行行情接收功能的开发。

## 2. 术语和缩略语

术语、缩写	说明
AMD	Archforce Market Data,华锐高速行情平台
AMA	Archforce Market Data API,华锐高速行情转码 API

# 二、文件说明

文件	说明
c++/	C++语言接口目录
c++/include/ama.h	API 头文件
c++/include/ama_datatype.h	API 使用的类型定义
c++/include/ama_struct.h	API 使用的数据结构定义
c++/include/hkex/ama_hkex_spi.h	API 商业港股行情回调基类的头文件定义
c++/include/hkex/ama_hkex_struct.h	API 商业港股行情数据结构定义
c++/lib/*	API 所依赖的动态链接库
c++/ama_demo.cpp	仅供参考的开发样例程序
c++/Makefile	ama_demo 程序编译脚本
c++/test_tool	C++版本测试工具,可以用来调试链路和数据比对,工具使用请参考 test_tool/README.txt

## 三、参数及精度说明

### 1. 参数影响说明

#### 1.1. channel\_mode 参数说明

参数作用:

用来设置接入上游的方式,即和上游建立数据通道的方式。

参数取值:

通道 [ChannelMode](#) 的优先级由 kFPGA 到 kPCAP 优先级依次递减(优先级越高,时延越小,优先级从高到低依次为 kFPGA/kRDMA/kEXA/kMDDP/kAMI/kUDP/kTCP/kPCAP)。

#### 1.2. tcp\_compress\_mode 参数说明

参数作用:

如果是 TCP 方式接入上游,此参数用来设置数据压缩方式,其他接入模式该参数不生效。

参数取值:

0 表示不压缩,1 表示华锐自定义数据压缩方式,2 表示 zstd 数据压缩方式。

#### 1.3. ha\_mode 参数说明

参数作用:

如果希望同时使用两种或两种以上的模式接入上游以达到更高级别的高可用效果(例如:同时使用 AMI 和 TCP 模式接入,若 AMI 模式异常无法使用时仍然可以使用 TCP 模式接入的数据),此参数用来设置多个接入模式之间的高可用切换方式。如果只使用一种模式接入上游,建议设置成 kMasterSlaveA/kMasterSlaveB。

参数取值:

参考 [HignAvailableMode](#)

#### 1.4. min\_log\_level 参数说明

参数作用:



用来设置最小日志级别(接口日志数据通过 [OnLog](#) 回调函数返回)。

参数取值:

取值范围:参考 [LogLevel](#) , 日志严重级别从上到下依次递增。

建议取值:kInfo。

### 1.5. is\_output\_mon\_data 参数说明

参数作用:

监控数据回调返回开关(监控数据通过 [OnIndicator](#) 回调函数返回)。

参数取值:

false: OnIndicator 回调函数不会返回任何数据。

true: OnIndicator 回调函数定时返回监控数据。

### 1.6. keep\_order 与 keep\_order\_timeout\_ms 参数说明

参数作用:

在 enable\_order\_book 设置为 true 的情况下,keep\_order 参数用来设置委托簿逐笔数据的保序开关,对非委托簿构建的逐笔数据不生效,keep\_order\_timeout\_ms 用来设置保序超时时间(ms);

在 enable\_order\_book 设置为 false 的情况下,keep\_order 参数不生效,直接递交给委托簿构建。

参数取值:

keep\_order 设置为 true 时,会对进入委托簿构建模块的逐笔委托和逐笔成交数据进行全局保序,如果上游数据出现乱序或者缺失,会等待 keep\_order\_timeout\_ms 超时时间,同时缓存逐笔数据。如果上游超时时间后,仍旧没有补齐数据,那么默认数据丢失,打印警告日志和增加监控丢失数量统计,直接递交给委托簿构建;

keep\_order 设置为 false 时,不对上游数据进行保序,递交给委托簿构建。

注意事项:

订阅操作会过滤数据,如果只订阅了逐笔数据的部分代码,那么保序操作没有意义,计算出的数据缺失的原因有可能是因为订阅过滤导致序号不连续触发的,因此非全部代码订阅时不能打

开逐笔保序开关。

## 1.7. is\_subscribe\_full 参数说明

参数作用:

初始订阅操作是否为全部订阅开关。

参数取值:

true:初始订阅值为订阅所有数据。

false:初始化订阅值为不订阅任何数据。

## 1.8. is\_thread\_safe 参数说明

参数作用:

业务数据回调接口(不包括 OnIndicator/OnLog 等功能数据回调)的线程安全模式设置。

参数取值:

true: 所有的业务数据接口为接口集线程安全(打开这一开关会导致 api 内部多一层线程合并封装,对时延敏感的系统不建议开启)。

false: 业务接口单接口为线程安全,接口集非线程安全。

## 1.9. 委托簿参数说明

数据类型	配置	说明
uint8_t	enable_order_book	是否启用委托簿,取值参考 <a href="#">OrderBookType</a> 如果启用委托簿,账户也需要有相应的委托簿权限
uint16_t	entry_size	委托簿输出最大档位数量(递交的委托簿数据档位小于等于 entry_size)
uint8_t	thread_num	构建委托簿的线程数量(递交和构建每一个市场分别开启 thread_num 个线程)
uint8_t	order_queue_size	每个档位输出的委托队列揭示(最大设置 50)
uint16_t	order_book_deliver_interval_microsecond	递交的最小时间间隔(单位:微妙, 委托簿递交间隔大于等于 order_book_deliver_interval_microsecond) 设置为 0 时, 委托簿实时下发, 不会等待间隔。

## 2. 精度说明

为兼容多市场,华锐 API 对各行情字段的数值取值统一了表现方式和相对实际值的倍数关系,具体表现如下:

沪、深、北、期货、港股、银行间、美股数据回调接口扩大倍数规则:

数据类型	二进制类型	说明	转码数据相对于实际的扩大倍数
Qty2	int64_t	数量 (如成交量,报买量等)	$10^2$
Qty4	int64_t	数量 (如当日累计成交量等)	$10^4$
Qty10	int64_t	数量 (如增发量,行权量等)	$10^{10}$
Price6	int64_t	价格 (如参考价,报买价等)	$10^6$
Amt5	int64_t	金额 (如拆借金额等)	$10^5$
Amt10	int64_t	面额(如发行价等)	$10^{10}$
Rate5	int64_t	汇率(如对回购标准券折算率等)	$10^5$
Rate6	int64_t	比例 (如最新利率,开盘利率等)	$10^6$
Rate7	int64_t	比例 (如到期收益率等)	$10^7$
Rate8	int64_t	汇率(如应计利息等)	$10^8$
Rate15	int64_t	利息(如每百元应计利息)	$10^{15}$

## 3. 产品数量数据单位说明(灰色代表无法确定品种单位)

品种类别	上海市场品种单位	深圳市场品种单位
股票	1 股	1 股
基金	1 份	1 份
债券	千元面额(10 张)	1 张
期权	1 张	1 张
指数	股票指数为 100 股 基金指数为 100 份 债券指数为 1 手(10 张)	

港股通	1 股	1 股
-----	-----	-----

## 四、兼容性说明

### 1. AMA 与服务端系统 AMD 的兼容性

AMA 不兼容低版本的 AMD,此接口不支持接入 4.2.0 之前版本的 AMD 系统,高版本的 AMD 兼容低版本的 AMA 接口。

### 2. AMA 对于操作系统的兼容性

系统	编译器	说明
RedHat7.6	gcc4.8.5	支持连接上游方式:TCP、AMI、PCAP、EXA、RDMA、MDDP、FPGA 模式。
Windows7 及以上版本(32 和 64 位)	VC++ 2017	支持连接上游方式:TCP、MDDP 模式

### 3. AMA 接口开发语言的支持

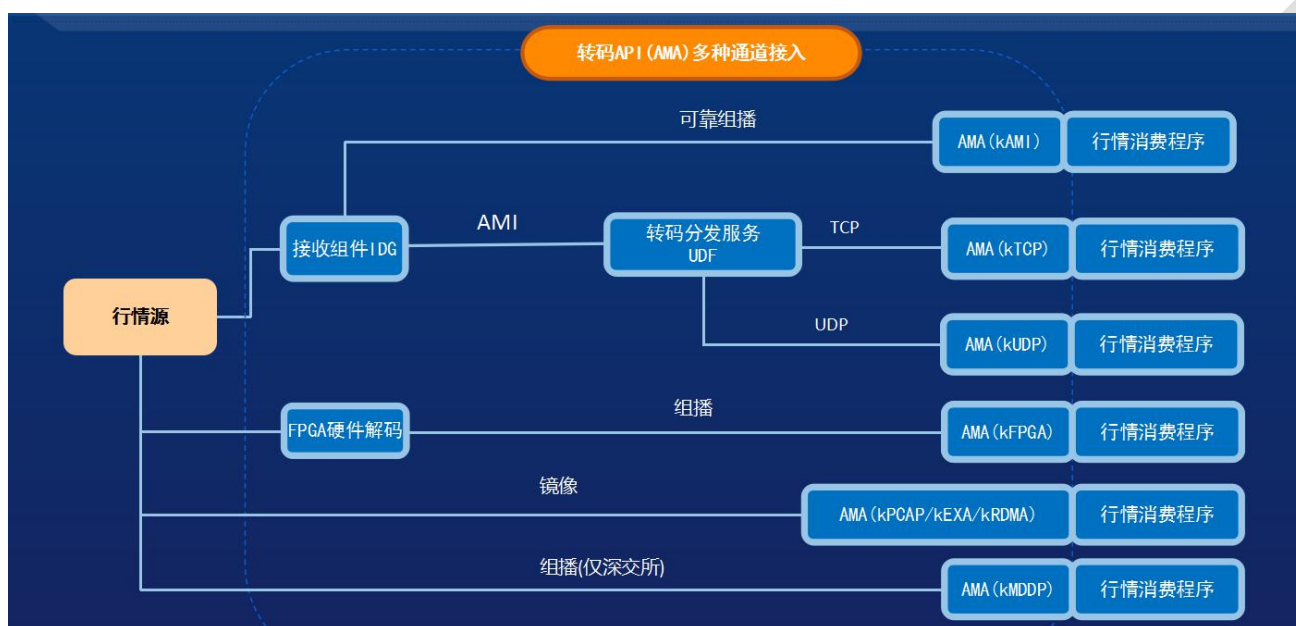
#### 3.1. 支持的语言接口

C++、JAVA、Python(2.7/3.6)

#### 3.2. 接口效率对比说明

接口 C++语言处理效率比 JAVA、Python 要快,如果数据量较大而且机器性能比较差的情况下建议使用 C++ 语言接口开发。

## 五、各通道说明



### 1. TCP 通道模式

#### 1.1. 数据接入方式

通过 TCP 方式和上游保持长连接,上游收到数据后主动推送转码数据给下游。TCP 通道登录成功信号通过 [OnEvent](#) 函数返回,信号值参考公共数据字典事件代码([EventCode](#))中 `kChannelTCPLogonSuccess` 的数值。

#### 1.2. 关键参数设置

`channel_mode:kTCP`

`tcp_compress_mode`:具体参数使用请参考 [tcp\\_compress\\_mode](#) 参数说明。

#### 1.3. 数据过滤方式

TCP 接入上游方式数据过滤全部是由上游组件处理

- 1) 账号权限:只会下发账户权限范围内的数据,权限范围外的数据过滤。
- 2) 订阅信息:只会下发订阅范围内的数据,订阅范围外的数据过滤。

#### 1.4. 异常情况说明

- 1) 网络问题导致下游连接断开,接口会自动发起重连,无需调用特殊接口处理。

2) 网络堵塞或者下游处理速度太慢会导致消息堆积,如果一直没有改善上游会将处理慢的下游连接断开,保证不同的连接之间不会互相影响。

## 2. AMI 通道模式说明

### 2.1. 数据接入方式

组播方式接入上游数据,上游收到数据后主动推送转码数据给下游(需要确保上游和下游组播网络连通,设置此模式前请先用工具测试组播网络的连通性)。AMI 通道登录成功信号通过 [OnEvent](#) 函数返回,信号值参考公共数据字典事件代码([EventCode](#))中 kChannelAMIInitSuccess 的数值。

如果上游也启用 RDMA,AMA 接收端必须要使用 Mellanox 网卡接收。

### 2.2. 关键参数设置

channel\_mode:kAMI

### 2.3. 数据过滤方式

1)账号权限:只会下发账户权限范围内的数据,权限范围外的数据过滤(由上游处理)。

2)订阅信息:下游过滤掉订阅范围以外的数据(由下游处理)。

## 3. DAS 抓包通道模式说明

### 3.1. 数据接入方式

通过抓取网卡数据来获取原始数据,解码后通过回调递交转码数据。

### 3.2. 关键参数设置

channel\_mode:kRDMA/kEXA/kPCAP

通道模式网卡要求说明:

数据接入方式	说明
kRDMA	网卡要求为支持 RDMA 功能的网卡
kEXA	网卡要求为 EXA 低延迟网卡
kPCAP	普通网卡

### 3.3. 数据过滤方式

1) 账号权限: 只会下发账户权限范围内的数据(权限由上游配置), 权限范围外的数据过滤(由下游处理)。

2) 订阅信息: 下游过滤掉订阅范围以外的数据(由下游处理)。

## 4. MDDP 通道模式说明

### 4.1. 数据接入方式

网关组播模式接入深交所网关获取原始数据解码后通过回调递交转码数据。

### 4.2. 关键参数设置

`channel_mode:kMDDP`

### 4.3. 数据过滤方式

1) 账号权限: 只会下发账户权限范围内的数据, 权限范围外的数据过滤(由上游处理)。

2) 订阅信息: 下游过滤掉订阅范围以外的数据(由下游处理), 此模式不支持品种数据类型订阅。

## 5. FPGA 通道模式说明

### 5.1. 数据接入方式

FPGA 模式以组播接收 FPGA 硬解之后的数据, 再通过回调接口递交解码数据。

### 5.2. 关键参数设置

`channel_mode:kFPGA`

### 5.3. 数据过滤方式

1) 账号权限: 只会下发账户权限范围内的数据, 权限范围外的数据过滤(由下游处理)。

2) 订阅信息: 下游过滤掉订阅范围以外的数据(由下游处理), 此模式不支持品种数据类型订阅。

### 5.4. 本地网卡选择方式

1、本地网卡仅支持 Solarflare 网卡和 Mellanox 网卡。



2、Solarfare 网卡 onload 驱动不同版本可能出现不兼容，比如 openonload-201811-u1 和 onload-7.0.0.176 版本不兼容，AMA 已经对此做过兼容处理，不排除后续版本出现其他不兼容的情况，建议使用 onload-7.0.0.176 版本。

## 6. UDP 通道模式说明

### 6.1. 数据接入方式

UDP 模式下游以 TCP 方式发起登陆和数据重传，上游以 UDP 组播的方式推送转码数据给下游，组播层面丢失的逐笔数据通过 TCP 方式发起重传。

### 6.2. 关键参数设置

channel\_mode:kUDP

### 6.3. 数据过滤方式

1)账号权限:只会下发账户权限范围内的数据(权限由上游配置),权限范围外的数据过滤(由下游处理)。

2)订阅信息:下游过滤掉订阅范围以外的数据(由下游处理)。

## 六、C++开发接口

### 1. IAMDApi 接口

AMA 接口操作类。该类不需要创建实例,直接调用类方法即可。如 IAMDApi::GetVersion()。

#### 1.1. GetVersion 方法

获取 AMA 版本信息。

函数原型:

```
static const char* GetVersion();
```

参数: 无

返回值:

版本信息字符串

## 1.2. Init 方法

初始化 AMA。

函数原型:

```
static int32_t Init(const IAMDSpi* pSpi, const Cfg& cfg);
```

参数:

参数	说明
pSpi(in)	IAMDSpi 的派生类实例指针,必须在调用 <b>Release</b> 函数之后才能销毁该实例
cfg (in)	AMA 内部需要的配置参数

返回值:

错误代码,参考公共数据字典错误码([ErrorCode](#))结构定义。

## 1.3. Join 方法

Join AMA 内部工作线程。非必调用的函数。仅当应用将各回调中的首地址直接转移至其他工作线程,而不在当前回调直接调用 `IAMDApi::FreeMemory(snapshots)` 删除数据的情况下,需先调用 `Join` 将 AMA 内部工作线程停止掉,然后再将转移的数据调用 `IAMDApi::FreeMemory(snapshots)` 删除,最后再调用 `Release` 彻底关闭 AMA。

函数原型:

```
static void Join();
```

参数: 无

返回值: 无

## 1.4. Release 方法

释放 AMA。

函数原型:

```
static int32_t Release();
```

返回值:

错误代码,参考公共数据字典错误码([ErrorCode](#))结构定义。

## 1.5. FreeMemory 方法

释放普通数据内存

AMA 行情数据回调数据/主动请求的返回数据,如果数据指针指向内存为 SDK 内部分配(参考具体回调以及接口说明,确认是否为 SDK 内部分配内存),都需要显示的调用 FreeMemory 释放内存,否则会造成内存泄漏。

函数原型:

```
static void FreeMemory(T* data);
```

参数:

参数	说明
T	此函数为重载函数,T 泛指需要被释放的数据类型
data (in)	需要被释放的内存首地址

返回值: 无

## 1.6. SubscribeData 方法

### 1.6.1. 权限订阅类型

根据市场、数据权限类型、股票代码订阅行情数据。各订阅数据类型与数据类型对应关系详见附录[权限数据类型订阅](#)。

函数原型:

```
static int32_t SubscribeData(int32_t subscribe_type,
                             const SubscribeItem* item,int32_t cnt);
```

参数:

参数	说明
subscribe_type(in)	订阅类型,取值参考公共数据字典订阅类型( <a href="#">SubscribeType</a> )定义
item(in)	订阅信息数据项首地址,参考 SubscribeItem 定义

cnt(in)	订阅信息数据项个数
---------	-----------

**SubscribeItem 结构定义如下:**

数据类型	字段名称	说明
int32_t	market	市场类型,参考公共数据字典市场类型 <a href="#">MarketType</a> ,为 0 表示订阅所有支持的市场
uint64_t	flag	各数据类型的集合,为 0 表示订阅所有支持的数据类型,参考权限订阅数据类型( <a href="#">SubscribeDataType</a> )结构体定义
char	security_code[32]	证券代码,为空表示订阅所有代码

**设置订阅示例代码如下:**

```
/*
    订阅信息设置:
    1. 订阅信息分三个维度 market:市场, flag:数据类型(比如快照, 逐笔成交, 逐笔成交等), 证券代码
    2. 订阅操作有三种:
        kSet 设置订阅, 新的订阅会覆盖之前的所有订阅信息
        kAdd 增加订阅, 在前一个基础上增加订阅信息
        kDel 删除订阅, 在前一个基础上删除订阅信息
        kCancelAll 取消所有订阅信息
*/
amd::ama::SubscribeItem sub[3];
memset(sub, 0, sizeof(sub));

/* 订阅深交所全部证券的逐笔委托和逐笔成交数据 */
sub[0].market = amd::ama::MarketType::kSZSE;
sub[0].flag = amd::ama::SubscribeDataType::kTickOrder
            | amd::ama::SubscribeDataType::kTickExecution;
sub[0].security_code[0] = '\0';

/* 订阅上交所 600000、600004 两只证券的全部数据 */
sub[1].market = amd::ama::MarketType::kSSE;
sub[1].flag = amd::ama::SubscribeDataType::kNone;
strcpy(sub[1].security_code, "600000");
sub[2].market = amd::ama::MarketType::kSSE;
sub[2].flag = amd::ama::SubscribeDataType::kNone;
strcpy(sub[2].security_code, "600004");

/* 发起订阅 */
if (amd::ama::IAMDApi::SubscribeData(
```

```
        amd::ama::SubscribeType::kSet, sub, 3)

    != amd::ama::ErrorCode::kSuccess)
{
    std::lock_guard<std::mutex> _(g_mutex);

    std::cout << "Subscribe data failed" << std::endl;

    amd::ama::IAMDApi::Release();

    return -1;
}
```

添加订阅代码示例如下：

```
// 添加订阅信息
amd::ama::SubscribeItem sub;

// 订阅上交所 600005 证券的逐笔委托和逐笔成交数据
sub.market = amd::ama::MarketType::kSSE;
sub.flag = amd::ama::SubscribeDataType::kTickOrder
           | amd::ama::SubscribeDataType::kTickExecution;
strcpy(sub.security_code, "600005");
// 订阅数据
if (amd::ama::IAMDApi::SubscribeData(
    amd::ama::SubscribeType::kAdd, &sub, 1)
    != amd::ama::ErrorCode::kSuccess)
{
    std::lock_guard<std::mutex> _(g_mutex);

    std::cout << "Subscribe data failed" << std::endl;

    return -1;
}
```

删除订阅代码示例如下：

```
// 添加订阅信息
amd::ama::SubscribeItem sub;

// 订阅上交所 600005 证券的逐笔委托和逐笔成交数据
sub.market = amd::ama::MarketType::kSSE;
sub.flag = amd::ama::SubscribeDataType::kTickOrder
           | amd::ama::SubscribeDataType::kTickExecution;
strcpy(sub.security_code, "600005");

// 订阅数据
if (amd::ama::IAMDApi::SubscribeData(
    amd::ama::SubscribeType::kDel, &sub, 1)
    != amd::ama::ErrorCode::kSuccess)
```

```
{
    std::lock_guard<std::mutex> _(g_mutex);
    std::cout << "Subscribe data failed" << std::endl;
    return -1;
}
```

取消所有订阅如下:

```
// 取消所有订阅信息
amd::ama::SubscribeItem sub;
// 订阅数据
if (amd::ama::IAMDApi::SubscribeData(
    amd::ama::SubscribeType::kCancelAll, &sub, 1)
    != amd::ama::ErrorCode::kSuccess)
{
    std::lock_guard<std::mutex> _(g_mutex);
    std::cout << "Subscribe data failed" << std::endl;
    return -1;
}
```

### 1.6.2. 品种订阅类型

根据市场类型、证券数据类型、证券品种类型、股票代码订阅行情数据。各订阅数据类型与数据类型对应关系详见附录[品种数据类型订阅](#)。

函数原型:

```
static int32_t SubscribeData(int32_t subscribe_type,
                             const SubscribeCategoryItem* item, int32_t cnt);
```

参数:

参数	说明
subscribe_type(in)	订阅类型,取值参考公共数据字典订阅类型( <a href="#">SubscribeType</a> )定义
item(in)	订阅信息数据项首地址,参考 SubscribeCategoryItem 定义
cnt(in)	订阅信息数据项个数

**SubscribeCategoryItem 定义:**

数据类型	字段名称	说明
int32_t	market	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> ),为 0 表示订阅所有

		支持的市场
uint64_t	data_type	各数据证券数据类型的集合,为 0 表示订阅所有支持的证券数据类型,参考证券数据类型( <a href="#">SubscribeSecuDataType</a> )定义
uint64_t	category_type	各数据证券品种类型的集合,为 0 表示订阅所有支持的证券品种数据类型,参考证券品种类型( <a href="#">SubscribeCategoryType</a> )定义
char	security_code[32]	证券代码,为空表示订阅所有代码

## 示例:

```
//按数据品种以及数据类型订阅实时行情数据
int32_t SubscribeWithCategory()
{
    /*
    按品种类型订阅信息设置:
    1. 订阅信息分三个维度 market:市场, data_type:证券数据类型, category_type:品种类型, security_code:证券代码
    2. 订阅操作有三种:
        kSet 设置订阅, 以市场为单位覆盖订阅信息
        kAdd 增加订阅, 在前一个基础上增加订阅信息
        kDel 删除订阅, 在前一个基础上删除订阅信息
        kCancelAll 取消所有订阅信息
    */
    amd::ama::SubscribeCategoryItem sub1[2];
    memset(sub1, 0, sizeof(sub1));

    /* 订阅深交所全部证券代码的股票逐笔委托和逐笔成交数据 */
    sub1[0].market = amd::ama::MarketType::kSZSE;
    sub1[0].data_type = amd::ama::SubscribeSecuDataType::kSnapshot
        | amd::ama::SubscribeSecuDataType::kTickExecution;
    sub1[0].category_type = amd::ama::SubscribeCategoryType::kStock;
    sub1[0].security_code[0] = '\0';

    /* 订阅上交所全部证券代码的基金逐笔委托和逐笔成交数据 */
    sub1[1].market = amd::ama::MarketType::kSSE;
    sub1[1].data_type = amd::ama::SubscribeSecuDataType::kSnapshot
        | amd::ama::SubscribeSecuDataType::kTickExecution;
    sub1[1].category_type = amd::ama::SubscribeCategoryType::kFund;
    sub1[1].security_code[0] = '\0';

    /* 发起订阅 */
    return amd::ama::IAMDApi::SubscribeData(amd::ama::SubscribeType::kSet, sub1, 2);
}
```



## 1.7. GetCodeTableList 方法

代码表请求操作

函数原型:

```
static bool GetCodeTableList(CodeTableRecordList& list,
                             const SubCodeTableItem* item, uint32_t cnt);
```

参数:

参数	说明
list (out)	代码表数据结构,具体参考 CodeTableRecordList 定义
item(in)	查询代码表信息数据项,具体参考 SubCodeTableItem 定义
cnt(in)	查询信息数据项个数

CodeTableRecordList 定义:

数据类型	字段名称	说明
uint32_t	list_nums	代码表数量
<a href="#">CodeTableRecord</a> *	records	代码表数据头指针,需要显示调用 FreeMemory 释放内存

SubCodeTableItem 定义:

数据类型	字段名称	说明
int32_t	market	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> ),为 kNone 表示查询所有支持的市场(目前只支持上交所、深交所与北交所)
char	security_code[16]	证券代码,为空表示查询所有代码

## 1.8. GetETFCodeTableList 方法

ETF 代码表请求操作

函数原型:

```
static bool GetETFCodeTableList(ETFCodeTableRecordList& list,
                                 const ETFItem* items, uint32_t cnt);
```

参数:

参数	说明
list (out)	ETF 代码表数据结构,具体参考 ETFCodeTableRecordList 定义(目前只支持上交所与深交所)
item(in)	查询 ETF 代码表信息数据项,具体参考 ETFItem 定义
cnt(in)	查询信息数据项个数

### ETFCodeTableRecordList 定义:

数据类型	字段名称	说明
uint32_t	etf_list_nums	ETF 代码表数量
<a href="#">ETFCodeTableRecord</a> *	etf_records	ETF 代码表数据头指针,需要显示调用 FreeMemory 释放内存

### ETFItem 定义:

数据类型	字段名称	说明
int32_t	market	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> ),为 kNone 表示查询所有支持的市场(目前只支持上交所与深交所)
char	security_code[16]	证券代码,为空表示查询所有代码

## 1.9. SubscribeOrderBookData 方法

委托簿数据及委托簿快照数据订阅操作

函数原型:

```
static int32_t SubscribeOrderBookData(
    int32_t subscribe_type, const SubscribeOrderBookItem* item, uint32_t cnt);
```

参数:

参数	说明
subscribe_type(in)	订阅类型,取值参考公共数据字典订阅类型( <a href="#">SubscribeType</a> )定义
item(in)	订阅信息数据项首地址,参考 SubscribeOrderBookItem 定义
cnt(in)	订阅信息数据项个数

### SubscribeOrderBookItem 结构定义如下:

数据类型	字段名称	说明
int32_t	market	市场类型,参考公共数据字典市场类型 <a href="#">MarketType</a> , 委托簿仅支

		持 kSSE/kSZSE 两个市场
uint64_t	flag	各数据类型的集合,参考 <a href="#">SubscribeOrderBookDataType</a>
char	security_code[16]	证券代码,仅支持单独订阅代码,订阅代码不能为空(服务端委托簿订阅有上限设置,订阅代码总数超过上限会导致订阅失败)

### 1.10. SubscribeDerivedData 方法

行情衍生数据订阅操作

函数原型:

```
static int32_t SubscribeDerivedData(int32_t subscribe_type,
    uint32_t derived_data_type, const SubscribeDerivedDataItem* item, uint32_t cnt);
```

参数:

参数	说明
subscribe_type(in)	订阅类型,取值参考公共数据字典订阅类型 ( <a href="#">SubscribeType</a> )定义
derived_data_type(in)	各行情衍生数据类型的集合,参考 <a href="#">SubscribeDerivedDataType</a>
item(in)	订阅代码以及行情衍生数据类型的数据项,具体参数请参考 SubscribeDerivedDataItem 定义
cnt(in)	订阅信息数据项个数

SubscribeDerivedDataItem 结构定义如下:

数据类型	字段名称	说明
int32_t	market	市场类型,参考 <a href="#">MarketType</a> ,行情衍生数据支持市场范围[kSSE/kSZSE],其余市场暂时不支持
char	security_code[16]	证券代码(不支持代码为空)

### 1.11. GetIMCExchangeRate 方法

获取国际市场汇率数据操作

函数原型:

```
static bool GetIMCExchangeRate(IMCExchangeRateList& list);
```

参数:

参数	说明
----	----

list (out)	代码表数据结构,具体参考 IMCExchangeRateList 定义
------------	-------------------------------------

## IMCExchangeRateList 定义:

数据类型	字段名称	说明
uint32_t	imc_list_nums	国际市场汇率数据数量
<a href="#">IMCExchangeRate</a> *	imc_rate_data	国际市场汇率数据头指针,需要显示调用 FreeMemory 释放内存

## 2. IAMDSpi 接口

AMA 中的接收数据的回调基类。使用 AMA 时需要继承该类,并将派生类的实例传递给 IAMDApi::Init 函数,供 AMA 回调使用。为保证程序正确运行,必须保证该实例生命周期长于 AMA。

### 2.1. OnLog 方法

接收日志数据回调。

函数原型:

```
void OnLog(const int32_t& level, const char* log, uint32_t len);
```

参数:

参数	说明
level(out)	日志数据级别,(参考日志输出级别 <a href="#">LogLevel</a> )
log(out)	日志内容
len(out)	日志内容长度

### 2.2. OnIndicator 方法

接收监控数据回调。

函数原型:

```
void OnIndicator(const char* indicator, uint32_t len);
```

参数:

参数	说明
----	----

indicator(out)	监控数据内容,格式为 JSON 字符串
len(out)	监控内容长度

## 1、json 字符串的内容

Json 字符串主要分为三个部分

### (1)、通道状态信息

通道状态信息主要包含通道状态信息和接收数据统计的数量信息。

### (2)、订阅信息

订阅信息主要包含订阅的市场类型、数据结构类型,品种类型,订阅的代码。

### (3)、逐笔保序信息

逐笔保序信息主要包含市场收到的逐笔数和市场遗失的逐笔数。

## 2、json 字符串示例(TCP 模式接入):

```
"class_name": "AppHandlerImpl",

"class_objects": [

    {

        "object_name": "AMA",

        "TCP": {

            "[TCP|AMA_L2]": {

                "RemoteIp": "",

                "RemotePort": "0",

                "LocalIp": "",

                "LocalPort": "0",

                "Status": "kDisconnect",

                "RecvTotalBytes": "0",

                "RecvTotalPkg": "0",

                .....
            }
        }
    }
]
```

```

        "RecvSnapshotCnt": "0",

        "SuccessSnapshotCnt": "0",

        .....

    },

    "SubscribeInfo": [

    {

        "Market": "SSE",

        "SubscribeDataTypes": [

            {

                "SubDataType": "Snapshot",

                "CategorysType": [

                    {

                        "CategoryType": "Stock",

                        "SubCodeList": "[ 600000 ]"

                    }

                ]

            }

        ],

        "TickSerialize": {

            "SSERecvTicks": "0",

            .....

        }

    ]

}

```

### 2.3. OnEvent 方法

接收事件通知回调,使用者可根据该回调事件做相应的处理。

函数原型:

```
void OnEvent(uint32_t level, uint32_t code, const char* event_msg, uint32_t len);
```

参数:

参数	说明
level(out)	事件级别,参考事件级别( <a href="#">EventLevel</a> )结构定义
code(out)	事件代码,参考公共数据字典事件代码( <a href="#">EventCode</a> )结构体定义
event_msg(out)	事件具体信息
len(out)	事件具体信息长度

## 2.4. OnMDSnapshot 方法

接收现货快照数据回调。

函数原型:

```
void OnMDSnapshot(MDSnapshot* snapshot, uint32_t cnt);
```

参数:

参数	说明
snapshot(out)	现货快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.5. OnMDOptionSnapshot 方法

接收期权快照数据回调。

函数原型:

```
void OnMDOptionSnapshot(MDOptionSnapshot* snapshot, uint32_t cnt);
```

参数:

参数	说明
snapshot(out)	期权快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	期权快照个数

## 2.6. OnMDHKTSnapshot 方法

接收港股通快照数据回调。

函数原型:



```
void OnMDHKTSnapshot(MDHKTSnapshot* snapshot, uint32_t cnt);
```

参数:

参数	说明
snapshot(out)	港股通快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.7. OnMDIndexSnapshot 方法

接收指数快照数据回调。

函数原型:

```
void OnMDIndexSnapshot(MDIndexSnapshot* snapshot,uint32_t cnt);
```

参数:

参数	说明
snapshot(out)	指数快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	指数快照个数

## 2.8. OnMDTickOrder 方法

接收逐笔委托数据回调。

函数原型:

```
void OnMDTickOrder(MDTickOrder* tick, uint32_t cnt);
```

参数:

参数	说明
tick(out)	逐笔委托数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.9. OnMDTickExecution 方法

接收逐笔成交数据回调。

函数原型:

```
void OnMDTickExecution(MDTickExecution* ticks, uint32_t cnt);
```

参数:

参数	说明
ticks(out)	逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.10. OnMDOrderQueue 方法

接收委托队列数据回调。

函数原型:

```
void OnMDOrderQueue(MDOrderQueue* orderqueues,uint32_t cnt);
```

参数:

参数	说明
orderqueues(out)	委托队列数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.11. OnMDAfterHourFixedPriceSnapshot 方法

接收盘后定价交易快照数据回调。

函数原型:

```
void OnMDAfterHourFixedPriceSnapshot(  
MDAfterHourFixedPriceSnapshot* snapshots,  
uint32_t cnt);
```

参数:

参数	说明
snapshots(out)	盘后定价交易快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.12. OnMDAfterHourFixedPriceTickExecution 方法

接收盘后定价交易逐笔成交数据回调。

函数原型:

```
void OnMDAfterHourFixedPriceTickExecution(  
MDAfterHourFixedPriceTickExecution* ticks,
```

uint32\_t cnt);

参数:

参数	说明
ticks (out)	盘后定价交易逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.13. OnMDFutureSnapshot 方法

接收期货快照数据回调。

函数原型:

void OnMDFutureSnapshot([MDFutureSnapshot](#)\* snapshots, uint32\_t cnt) ;

参数:

参数	说明
snapshots (out)	期货快照成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.14. OnMDCSIIIndexSnapshot 方法

接收上交所中证指数快照数据回调。

函数原型:

void OnMDCSIIIndexSnapshot([MDCSIIIndexSnapshot](#)\* snapshots, uint32\_t cnt) ;

参数:

参数	说明
snapshots (out)	中证指数快照成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.15. OnMDIndicatorOfTradingVolumeSnapshot 方法

接收深交所成交量统计指标快照数据回调。

函数原型:

void OnMDIndicatorOfTradingVolumeSnapshot(  
  
[MDIndicatorOfTradingVolumeSnapshot](#)\* snapshots, uint32\_t cnt);

参数:

参数	说明
snapshots(out)	深交所成交量统计快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.16. OnMDCnIndexSnapshot 方法

接收深交所国证指数快照数据回调。

函数原型:

```
void OnMDCnIndexSnapshot (
    MDCnIndexSnapshot* snapshots, uint32_t cnt);
```

参数:

参数	说明
snapshots(out)	深交所国证指数快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.17. OnMDRefinancingTickOrder 方法

接收深交所转融通证券出借逐笔委托数据回调。

函数原型:

```
void OnMDRefinancingTickOrder (MDRefinancingTickOrder* ticks, uint32_t cnt);
```

参数:

参数	说明
ticks (out)	深交所转融通逐笔委托数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.18. OnMDRefinancingTickExecution 方法

接收深交所转融通证券出借逐笔成交数据回调。

函数原型:

```
void OnMDRefinancingTickExecution (MDRefinancingTickExecution* ticks, uint32_t cnt);
```

参数:

参数	说明
ticks (out)	深交所转融通逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.19. OnMDNegotiableTickOrder 方法

接收深交所协议交易逐笔委托数据回调。

函数原型:

```
void OnMDNegotiableTickOrder(MDNegotiableTickOrder* ticks, uint32_t cnt);
```

参数:

参数	说明
ticks (out)	深交所协议交易逐笔委托数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.20. OnMDNegotiableTickExecution 方法

接收深交所协议交易逐笔成交数据回调。

函数原型:

```
void OnMDNegotiableTickExecution(MDNegotiableTickExecution* ticks, uint32_t cnt);
```

参数:

参数	说明
ticks (out)	深交所协议交易逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.21. OnMDHKTRealtimeLimit 方法

接收港股通实时额度数据回调。

函数原型:

```
void OnMDHKTRealtimeLimit(MDHKTRealtimeLimit* limits, uint32_t cnt);
```

参数:

参数	说明
----	----

limits(out)	港股通实时额度数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.22. OnMDHKTProductStatus 方法

接收港股通可接收订单并转发产品状态数据回调。

函数原型:

```
void OnMDHKTProductStatus(MDHKTProductStatus* status, uint32_t cnt);
```

参数:

参数	说明
status (out)	港股通产品状态数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.23. OnMDHKTVCM 方法

接收港股通 VCM 数据回调。

函数原型:

```
void OnMDHKTVCM(MDHKTVCM* vcms, uint32_t cnt);
```

参数:

参数	说明
vcms(out)	vcm 数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.24. OnMDNEEQSnapshot 方法

接收股转系统证券行情信息回调。

函数原型:

```
void OnMDNEEQSnapshot(MDNEEQSnapshot* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	股转系统证券行情信息数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.25. OnMDNEEQSecurityInfo 方法

接收股转系统证券信息回调。

函数原型:

```
void OnMDNEEQSecurityInfo(MDNEEQSecurityInfo* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	股转系统证券信息数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.26. OnMDNEEQNonPublicTransDeclaredInfo 方法

接收股转非公开转让申报信息库数据回调。

函数原型:

```
void OnMDNEEQNonPublicTransDeclaredInfo(  
MDNEEQNonPublicTransDeclaredInfo* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	股转系统非公开申报转让信息数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.27. OnMDNEEQHierarchicalInfo 方法

接收股转系统分层信息库数据回调。

函数原型:

```
void OnMDNEEQHierarchicalInfo(MDNEEQHierarchicalInfo* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	股转系统分层信息数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.28. OnMDHKMarketStatus 方法

接收港股通市场状态数据回调。

函数原型:

```
void OnMDHKMarketStatus (MDHKMarketStatus* status, uint32_t cnt);
```

参数:

参数	说明
status(out)	港股通市场状态数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.29. OnMDNEEQNegotiableDeclaredInfo 方法

接收北交所协议转让申报信息库数据回调。

函数原型:

```
void OnMDNEEQNegotiableDeclaredInfo  
(MDNEEQNegotiableDeclaredInfo* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	北交所协议转让申报信息库数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

## 2.30. OnMDNEEQMarketMakerDeclaredInfo 方法

接收北交所做市业务申报信息库数据回调。

函数原型:

```
void OnMDNEEQMarketMakerDeclaredInfo  
(MDNEEQMarketMakerDeclaredInfo* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	北交所做市业务申报信息库数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数



### 2.31. OnMDNEEQNonPublicTransferDealInfo 方法

接收北交所非公开转让成交信息库数据回调。

函数原型:

```
void OnMDNEEQNonPublicTransferDealInfo
(MDNEEQNonPublicTransferDealInfo* infos, uint32_t cnt);
```

参数:

参数	说明
infos(out)	北交所非公开转让成交信息库数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.32. OnMDOrderBook 方法

接收委托簿数据回调。

函数原型:

```
void OnMDOrderBook(std::vector<amd::ama::MDOrderBook>& order_book);
```

参数:

参数	说明
order_book(out)	委托簿数据结构

### 2.33. OnMDBondSnapshot 方法

接收债券快照数据回调。

函数原型:

```
void OnMDBondSnapshot(MDBondSnapshot* snapshots, uint32_t cnt);
```

参数:

参数	说明
snapshot(out)	债券快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.34. OnMDBondTickOrder 方法

接收债券逐笔委托数据回调。

函数原型:

```
void OnMDBondTickOrder (MDBondTickOrder * tick, uint32_t cnt);
```

参数:

参数	说明
tick(out)	债券逐笔委托数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.35. OnMDBondTickExecution 方法

接收债券逐笔成交数据回调。

函数原型:

```
void OnMDBondTickExecution (MDBondTickExecution * tick, uint32_t cnt);
```

参数:

参数	说明
tick(out)	债券逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.36. OnMDBondQuotedTickOrder 方法

接收深交所债券业务报价及大额逐笔委托数据回调。

函数原型:

```
void OnMDBondQuotedTickOrder (MDBondQuotedTickOrder * tick, uint32_t cnt);
```

参数:

参数	说明
tick(out)	债券业务报价及大额逐笔委托数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.37. OnMDBondQuotedTickExecution 方法

接收深交所债券业务报价及大额逐笔成交数据回调。

函数原型:

```
void OnMDBondQuotedTickExecution (MDBondQuotedTickExecution * tick, uint32_t cnt);
```

参数:

参数	说明
tick(out)	债券业务报价及大额逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.38. OnMDOrderBookSnapshot 方法

接收委托簿快照数据回调。

函数原型:

```
void OnMDOrderBookSnapshot(MDOrderBookSnapshot* order_book_snapshots, uint32_t cnt);
```

参数:

参数	说明
tick(out)	委托簿快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.39. OnMDFundExpertSnapshot 方法

接收基金通快照数据回调。

函数原型:

```
void OnMDFundExpertSnapshot(MDFundExpertSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	基金通快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	数据个数

### 2.40. OnMDIOPVSnapshot 方法

接收 IOPV 快照数据回调

函数原型:

```
void OnMDIOPVSnapshot(MDIOPVSnapshot* iopv_snapshots, uint32_t cnt);
```

参数:

参数	说明
----	----

iopv_snapshots(out)	IOPV 快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收 IOPV 快照数据条数

#### 2.41. OnMDZQQDBJSnapshot 方法

接收确定报价固收行情数据回调

函数原型:

```
void OnMDZQQDBJSnapshot(MDZQQDBJSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	确定报价固收行情数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收确定报价固收行情数据条数

#### 2.42. OnMDZQCJHQSsnapshot 方法

接收成交行情固收行情数据回调

函数原型:

```
void OnMDZQCJHQSsnapshot(MDZQCJHQSsnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	成交行情固收行情数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收成交行情固收行情数据条数

#### 2.43. OnMDZQCJMXSnapshot 方法

接收成交明细固收行情数据回调

函数原型:

```
void OnMDZQCJMXSnapshot(MDZQCJMXSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	成交明细固收行情数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收成交明细固收行情数据条数

### 3. \*HKExSpi 接口

‘\*’指非标准支持类型,需要和提供此服务的券商确认是否支持商业港股。

AMA 中的接收商业港股数据的回调基类。使用 AMA 时无需做特殊处理,IAMDSpi 已继承该类,直接调用 IAMDSpi 类即可。

#### 3.1. \*OnMDHKExSnapshot 方法

接收商业港股股票快照数据回调。

函数原型:

```
void OnMDHKExSnapshot (MDHKExSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	商业港股股票快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收商业港股股票快照数据条数

#### 3.2. \*OnMDHKExOrderSnapshot 方法

接收商业港股委托挂单数据回调。

函数原型:

```
void OnMDHKExOrderSnapshot(MDHKExOrderSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	商业港股委托挂单数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收商业港股委托挂单数据条数

#### 3.3. \*OnMDHKExOrderBrokerSnapshot 方法

接收商业港股经纪席位数据回调。

函数原型:

```
void OnMDHKExOrderBrokerSnapshot(MDHKExOrderBrokerSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	商业港股经纪席位数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收商业港股经纪席位数据条数

### 3.4. \*OnMDHKExTickExecution 方法

接收商业港股逐笔成交数据回调。

函数原型:

```
void OnMDHKExTickExecution(MDHKExTickExecution* ticks, uint32_t cnt);
```

参数:

参数	说明
ticks(out)	商业港股逐笔成交数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收商业港股逐笔成交数据条数

### 3.5. \*OnMDHKExIndexSnapshot 方法

接收商业港股指数行情快照回调。

函数原型:

```
void OnMDHKExIndexSnapshot(MDHKExIndexSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	商业港股指数行情快照首指针,需要显示的调用 FreeMemory
cnt(out)	接收商业港股指数行情快照条数

### 3.6. \*OnMDHKExConnectTurnoverSnapshot 方法

接收商业港股港股通成交额快照回调。

函数原型:

```
void OnMDHKExConnectTurnoverSnapshot(  
    MDHKExConnectTurnoverSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	商业港股港股通成交额快照首指针,需要显示的调用 FreeMemory
cnt(out)	接收港股港股通成交额快照条数

## 4. \*UsaSpi 接口

‘\*’指非标准支持类型,需要和提供此服务的券商确认是否支持美股。

AMA 中的接收美股数据的回调基类。使用 AMA 时无需做特殊处理,IAMDSpi 已继承该类,直接调用 IAMDSpi 类即可。

### 4.1. \*OnMDUsaStockSnapshot 方法

接收美股股票快照数据回调。

函数原型:

```
void OnMDUsaStockSnapshot(MDUsaStockSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	美股股票快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收美股股票快照数据条数

### 4.2. \*OnMDUsaOrderSnapshot 方法

接收美股挂单快照数据回调。

函数原型:

```
void OnMDUsaOrderSnapshot(MDUsaOrderSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	美股挂单快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收美股挂单快照数据条数

### 4.3. \*OnMDUsaEodSnapshot 方法

接收美股 EOD 收盘快照数据回调。

函数原型:

```
void OnMDUaEodSnapshot(MDUaEodSnapshot* snaps, uint32_t cnt);
```

参数:

参数	说明
snaps(out)	美股 EOD 收盘快照数据首指针,需要显示的调用 FreeMemory
cnt(out)	接收美股 EOD 收盘快照数据条数

## 5. 工具函数说明

### 5.1. ConvertTradingPhaseCode 方法

交易阶段代码转换函数,可以使用此函数将沪深交易所披露的交易阶段代码转换成华锐自定义交易阶段代码定义 [TradingPhaseCode](#)。

函数原型:

```
static uint16_t ConvertTradingPhaseCode(const uint32_t& market_type,  
    const uint64_t& orig_time, char* trading_phase_code, const uint32_t& len = 8);
```

参数:

参数	说明
market_type(in)	市场标识(可以参考类型 <a href="#">MarketType</a> )
orig_time(in)	交易时间
trading_phase_code(in)	需要转换的交易阶段代码
len(in)	交易阶段代码字符串长度(不填的话默认为 8)

返回值:

华锐自定义交易阶段定义 [TradingPhaseCode](#)

## 6. 接口线程安全说明

接口方法集	线程安全
GetVersion Init	接口集线程安全



Join	
Release	
FreeMemory	
SubscribeData	
GetCodeTableList	
GetETFCodeTableList	
OnLog	单独线程安全
OnIndicator	单独线程安全
OnEvent	单独线程安全
OnMDSnapshot	受 <code>cfg.is_thread_safe</code> 参影响具体请参考 <code>is_thread_safe</code> 参数说明
OnMDOptionSnapshot	
OnMDHKTSnapshot	
OnMDIndexSnapshot	
OnMDTickOrder	
OnMDTickExecution	
OnMDOOrderQueue	
OnMDAfterHourFixedPriceSnapshot	
OnMDAfterHourFixedPriceTickExecution	
OnMDFutureSnapshot	
OnMDCSIIndexSnapshot	
OnMDIndicatorOfTradingVolumeSnapshot	
OnMDCnIndexSnapshot	
OnMDRefinancingTickOrder	
OnMDRefinancingTickExecution	
OnMDNegotiableTickOrder	
OnMDNegotiableTickExecution	
OnMDHKTRealtimeLimit	
OnMDHKTProductStatus	

OnMDNEEQSnapshot	
OnMDHKTVC	
OnMDNEEQSecurityInfo	
OnMDNEEQNonPublicTransDeclaredInfo	
OnMDNEEQHierarchicalInfo	
OnMDHKMarketStatus	
OnMDNEEQNegotiableDeclaredInfo	
OnMDNEEQMarketMakerDeclaredInfo	
OnMDNEEQNonPublicTransferDealInfo	
OnMDBondSnapshot	
OnMDBondTickOrder	
OnMDBondTickExecution	
OnMDBondQuotedTickOrder	
OnMDBondQuotedTickExecution	
OnMDOrderBookSnapshot	
OnMDFundExpertSnapshot	
OnMDIOPVSnapshot	
OnMDHKExSnapshot	
OnMDHKExOrderSnapshot	
OnMDHKExOrderBrokerSnapshot	
OnMDHKExTickExecution	
OnMDHKExIndexSnapshot	
OnMDHKExConnectTurnoverSnapshot	
OnMDUsaStockSnapshot	
OnMDUsaOrderSnapshot	
OnMDUsaEodSnapshot	
OnMDZQQDBJSnapshot	
OnMDZQCJHQS	
OnMDZQCJMXSnapshot	

OnMDOOrderBook	不受 is_thread_safe 参数影响,线程非安全,深圳市场和上海市场分别由不同线程回调
----------------	---

## 7. 行情数据结构字典

### 7.1. 沪、深、北、港股通、期货数据字典

#### 7.1.1. 配置结构定义(Cfg)

数据类型	字段名称	说明
全局配置信息		
uint64_t	channel_mode	通道模式的集合,请参考 <a href="#">channel_mode</a> 参数说明
uint32_t	ha_mode	高可用工作模式,请参考 <a href="#">ha_mode</a> 参数说明
int32_t	min_log_level	日志最小级别,请参考 <a href="#">min_log_level</a> 参数说明
bool	is_output_mon_data	是否输出监控数据的配置,true-输出监控数据,false-不输出监控数据
bool	is_thread_safe	回调接口是否保证线程安全,true-启用线程安全模式执行回调接口,false-非线程安全模式执行回调接口
bool	keep_order	逐笔保序标志,true-开启保序,false-开启不保序
uint32_t	keep_order_timeout_ms	逐笔保序超时时间(单位:毫秒),keep_order=true 时有效
bool	is_subscribe_full	默认是否订阅全部数据,true-默认订阅全部,false-默认不订阅任何数据
UMS 服务的连接信息		
UMSItem	ums_servers[8]	UMS 的服务信息项,该信息不能超过 8 个
uint32_t	ums_server_cnt	UMS 的服务信息项个数,小于 1 将启动失败
char	username[32]	用户名
char	password[64]	用户密码,明文填入,密文使用
uint32_t	tcp_compress_mode	TCP 模式传输数据压缩标志,0:不压缩 1:自定义压缩 2:zstd 压缩(仅 TCP 模式有效)
委托簿配置		
uint8_t	enable_order_book	启用委托簿
uint16_t	entry_size	委托簿档位

uint8_t	thread_num	委托簿并行计算线程数
uint8_t	order_queue_size	每个价位委托揭示笔数,最高 50
uint32_t	order_book_deliver_interval_microsecond	委托簿递交时间间隔(微秒级)

### 7.1.2. 现货快照结构定义(MDSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
char	trading_phase_code[8]	<p>交易阶段代码</p> <p>上海现货快照交易状态:</p> <p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填空格。</p> <p>第 0 位:'S'表示启动(开市前)时段,'C'表示开盘集合竞价时段,'T'表示连续交易时段,'E'表示闭市时段,'P'表示产品停牌。</p> <p>第 1 位:'0'表示此产品不可正常交易,'1'表示此产品可正常交易。</p> <p>第 2 位:'0'表示未上市,'1'表示已上市。</p> <p>第 3 位:'0'表示此产品在当前时段不接受进行新订单申报,'1'表示此产品在当前时段可接受进行新订单申报。</p> <p>深圳现货快照交易状态:</p> <p>第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断。</p> <p>第 1 位:'0'= 正常状态 '1'= 全天停牌。</p>
Price6	pre_close_price	昨收价
Price6	open_price	开盘价
Price6	high_price	最高价
Price6	low_price	最低价
Price6	last_price	最新价
Price6	close_price	收盘价
Price6	bid_price[10]	申买价

Qty2	bid_volume[10]	申买量
Price6	offer_price[10]	申卖价
Qty2	offer_volume[10]	申卖量
int64_t	num_trades	成交笔数
Qty2	total_volume_trade	成交总量
Amt5	total_value_trade	成交总金额
Qty2	total_bid_volume	委托买入总量
Qty2	total_offer_volume	委托卖出总量
Price6	weighted_avg_bid_price	加权平均为委买价格
Price6	weighted_avg_offer_price	加权平均为委卖价格
Price6	IOPV	IOPV 净值估产
Rate7	yield_to_maturity	到期收益率
Price6	high_limited	涨停价
Price6	low_limited	跌停价
Rate6	price_earning_ratio1	市盈率 1(仅深圳有效)
Rate6	price_earning_ratio2	市盈率 2(仅深圳有效)
Rate6	change1	涨跌 1(对比昨收价,仅深圳有效)
Rate6	change2	涨跌 2(对比上一笔,仅深圳有效)
int32_t	channel_no	频道代码(仅深圳有效)
char	md_stream_id[6]	行情类别
char	instrument_status[8]	当前品种交易状态
Price6	pre_close_iopv	基金 T-1 日收盘时刻 IOPV
Price6	alt_weighted_avg_bid_price	债券加权平均委买价格(仅上海有效)
Price6	alt_weighted_avg_offer_price	债券加权平均委卖价格(仅上海有效)
int64_t	etf_buy_number	ETF 申购笔数(仅上海有效)

Qty2	etf_buy_amount	ETF 申购数量(仅上海有效)
Amt5	etf_buy_money	ETF 申购金额(仅上海有效)
int64_t	etf_sell_number	ETF 赎回笔数(仅上海有效)
Qty2	etf_sell_amount	ETF 赎回数量(仅上海有效)
Amt5	etf_sell_money	ETF 赎回金额(仅上海有效)
Qty2	total_warrant_exec_volume	权证执行的总数量(仅上海有效)
Price6	war_lower_price	债券质押式回购品种加权平均价(仅上海有效)
Price6	war_upper_price	权证涨停价格(仅上海有效)
int64_t	withdraw_buy_number	买入撤单笔数(仅上海有效)
Qty2	withdraw_buy_amount	买入撤单数量(仅上海有效)
Amt5	withdraw_buy_money	买入撤单金额(仅上海有效)
int64_t	withdraw_sell_number	卖出撤单笔数(仅上海有效)
Qty2	withdraw_sell_amount	卖出撤单数量(仅上海有效)
Amt5	withdraw_sell_money	卖出撤单金额(仅上海有效)
int64_t	total_bid_number	买入总笔数(仅上海有效)
int64_t	total_offer_number	卖出总笔数(仅上海有效)
int32_t	bid_trade_max_duration	买入委托成交最大等待时间(仅上海有效)
int32_t	offer_trade_max_duration	卖出委托成交最大等待时间(仅上海有效)
int32_t	num_bid_orders	买方委托价位数(仅上海有效)
int32_t	num_offer_orders	卖方委托价位数(仅上海有效)
int64_t	last_trade_time	最近成交时间(为 YYYYMMDDHHMMSSsss 仅上海 00 文件,LDDS 生效)

uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
---------	------------------	---

### 7.1.3. 期权快照结构定义(MDOptionSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	期权代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
Price6	pre_settle_price	昨结算价(仅上海有效)
Price6	pre_close_price	昨收盘价
Price6	open_price	今开盘价
Price6	auction_price	动态参考价 (波动性中断参考价)
Qty2	auction_volume	虚拟匹配数量
Price6	high_price	最高价
Price6	low_price	最低价
Price6	last_price	最新价
Price6	close_price	收盘价
Price6	high_limited	涨停价
Price6	low_limited	跌停价
Price6	bid_price[5]	申买价
Qty2	bid_volume[5]	申买量
Price6	offer_price[5]	申卖价
Qty2	offer_volume[5]	申卖量
Price6	settle_price	今日结算价
Qty2	total_long_position	总持仓量
Qty2	total_volume_trade	总成交数
Amt5	total_value_trade	总成交额
char	trading_phase_code[8]	交易阶段代码  上海期权交易状态:

		<p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填充空格。</p> <p>第 0 位:'S'表示启动(开市前)时段,'C'表示集合竞价时段,'T'表示连续交易时段,'B'表示休市时段,'E'表示闭市时段,'V'表示波动性中断,'P'表示临时停牌,'U'表示收盘集合竞价 'M'表示可恢复交易的熔断(盘中集合竞价),'N'表示不可恢复交易的熔断(暂停交易至闭市)。</p> <p>第 1 位:'0'表示未连续停牌,'1'表示连续停牌(预留,暂填充格)。</p> <p>第 2 位:'0'表示不限制开仓,'1'表示限制备兑开仓,'2'表示卖出开仓,'3'表示限制卖出开仓、备兑开仓,'4'表示限制买入开仓,'5'表示限制买入开仓、备兑开仓,'6'表示限制买入开仓、卖出开仓,'7'表示限制买入开仓、卖出开仓、备兑开仓。</p> <p>第 3 位:'0'表示此产品在当前时段不接受进行新订单申报,'1'表示此产品在当前时段可接受进行新订单申报。</p> <p>深圳期权交易状态:</p> <p>第 0 位:S= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'= 波动性中断。</p> <p>第 1 位:'0'= 正常状态 '1'= 全天停牌。</p>
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别
int64_t	last_trade_time	最近成交时间(为 YYYYMMDDHHMMSSsss 仅上海 03 文件,LDSS 生效)
Price6	ref_price	参考价(仅深圳有效)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
char	contract_type	合约类别
int32_t	expire_date	到期日
char	underlying_security_code[16]	标的代码
Price6	exercise_price	行权价

#### 7.1.4. 港股通快照结构定义(MDHKTSnapshot)

数据类型	字段名称	说明
------	------	----



int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	港股通代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
Price6	pre_close_price	昨收价
Price6	nominal_price	按盘价
Price6	high_price	最高价
Price6	low_price	最低价
Price6	last_price	最新价
Price6	bid_price[5]	申买价
Qty2	bid_volume[5]	申买量
Price6	offer_price[5]	申卖价
Qty2	offer_volume[5]	申卖量
Qty2	total_volume_trade	总成交数
Amt5	total_value_trade	总成交额
char	trading_phase_code[8]	<p>交易阶段代码</p> <p>上海港股通交易状态:</p> <p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填空格。</p> <p>第 0 位:'0'表示正常,'1'表示暂停交易。</p> <p>深圳港股通交易状态:</p> <p>第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断。</p> <p>第 1 位:'0'= 正常状态 '1'= 全天停牌。</p>
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别
Price6	ref_price	参考价格
Price6	high_limited	涨停价
Price6	low_limited	跌停价

Price6	bid_price_limit_up	买盘上限价
Price6	bid_price_limit_down	买盘下限价
Price6	offer_price_limit_up	卖盘上限价
Price6	offer_price_limit_down	卖盘下限价
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.5. 指数快照结构定义(MDIndexSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
char	trading_phase_code[8]	交易阶段代码(仅深圳有效)  深圳指数快照交易状态  第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断  第 1 位:'0'= 正常状态 '1'= 全天停牌
Price6	pre_close_index	前收盘指数
Price6	open_index	今开盘指数
Price6	high_index	最高指数
Price6	low_index	最低指数
Price6	last_index	最新指数
Price6	close_index	收盘指数
Qty2	total_volume_trade	参与计算相应指数的交易数量
Amt5	total_value_trade	参与计算相应指数的成交总金额
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.6. 逐笔委托结构定义(MDTickOrder)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int32_t	channel_no	频道号
int64_t	appl_seq_num	频道索引
int64_t	order_time	时间(YYYYMMDDHHMMSSsss)
Price6	order_price	委托价格
Qty2	order_volume	深圳市场:委托数量,上海市场:剩余委托数量
uint8_t	side	买卖方向 深圳市场:(1-买 2-卖 G-借入 F-出借) 上海市场:(B:买单,S:卖单)
uint8_t	order_type	订单类别 深圳市场:(1-市价 2-限价 U-本方最优) 上海市场:(A:增加委托,D:删除委托)
char	md_stream_id[6]	行情类别(仅深圳有效)
int64_t	orig_order_no	原始订单号
int64_t	biz_index	业务序号
uint8_t	variety_category	品种类别 (取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
Qty2	traded_order_volume	已成交的委托数量(仅上海市场逐笔委托和逐笔成交合并后有效)

### 7.1.7. 逐笔成交结构定义(MDTickExecution)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	exec_time	时间(YYYYMMDDHHMMSSsss)
int32_t	channel_no	频道号
int64_t	appl_seq_num	频道编号
Price6	exec_price	成交价格

Qty2	exec_volume	成交数量
Amt5	value_trade	成交金额
int64_t	bid_appl_seq_num	买方委托索引
int64_t	offer_appl_seq_num	卖方委托索引
uint8_t	side	买卖方向(仅上海有效 B-外盘,主动买 S-内盘,主动卖 N-未知)
uint8_t	exec_type	成交类型(深圳:4-撤销 F-成交,上海:F-成交)
char	md_stream_id[6]	行情类别(仅深圳有效)
int64_t	biz_index	业务序号
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.8. 委托队列结构定义(MDOrderQueue)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	order_time	委托时间(YYYYMMDDHHMMSSsss)
uint8_t	side	买卖方向(B-买 S-卖)
Price6	order_price	委托价格
Qty2	order_volume	订单数量
int32_t	num_of_orders	总委托笔数
int32_t	items	明细个数
int64_t	volume[50]	订单明细
int32_t	channel_no	频道号
char	md_stream_id[6]	行情类别(仅深圳有效)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.9. 盘后定价交易快照结构定义(MDAfterHourFixedPriceSnapshot)

数据类型	字段名称	说明
------	------	----

int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
char	trading_phase_code[8]	<p>交易阶段代码</p> <p>上海盘后快照交易状态:</p> <p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填空格。</p> <p>第 0 位:'T'表示启动(开市前)时段,'A'表示集中撮合时段,'H'表示连续交易时段,'D'表示闭市时段,'F'表示停牌。</p> <p>深圳盘后快照交易状态:</p> <p>第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断。</p> <p>第 1 位:'0'= 正常状态 '1'= 全天停牌。</p>
Price6	close_price	今日收盘价(仅上海有效)
Price6	bid_price	申买价
Qty2	bid_volume	申买量
Price6	offer_price	申卖价
Qty2	offer_volume	申卖量
Price6	pre_close_price	昨收价
int64_t	num_trades	成交笔数
Qty2	total_volume_trade	成交总量
Amt5	total_value_trade	成交总金额
Qty2	total_bid_volume	委托买入总量(仅上海有效)
Qty2	total_offer_volume	委托卖出总量(仅上海有效)
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.10. 盘后定价交易逐笔成交结构定义(MDAfterHourFixedPriceTickExecution)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	appl_seq_num	消息记录号
char	security_code[16]	证券代码
int64_t	exec_time	成交时间(YYYYMMDDHHMMSSsss)
Price6	exec_price	成交价格
Qty2	exec_volume	成交数量
Amt5	value_trade	成交金额
int64_t	bid_appl_seq_num	买方委托索引
int64_t	offer_appl_seq_num	卖方委托索引
uint8_t	side	买卖方向(B-外盘,主动买 S-内盘,主动卖 N-未知)
uint8_t	exec_type	成交类型
int32_t	channel_no	频道代码
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.11. 期货快照结构定义(MDFutureSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[32]	合约代码
int32_t	action_day	业务日期
int64_t	orig_time	交易日 YYYYMMDDHHMMSSsss(ActionDay + UpdateTime + UpdateMillisec)
char	exchange_inst_id[31]	合约在交易所的代码
Price6	last_price	最新价
Price6	pre_settle_price	上次结算价
Price6	pre_close_price	昨收价
Qty2	pre_open_interest	昨持仓量

Price6	open_price	开盘价
Price6	high_price	最高价
Price6	low_price	最低价
Qty2	total_volume_trade	数量
Amt5	total_value_trade	总成交金额
Qty2	open_interest	持仓量
Price6	close_price	今收盘
Price6	settle_price	本次结算价
Price6	high_limited	涨停板价
Price6	low_limited	跌停板价
Rate6	pre_delta	昨虚实度
Rate6	curr_delta	今虚实度
Price6	bid_price[5]	申买价
Qty2	bid_volume[5]	申买量
Price6	offer_price[5]	申卖价
Qty2	offer_volume[5]	申卖量
Price6	average_price	当日均价
int32_t	trading_day	交易日期
uint8_t	variety_category	品种类别 (取值参照公共数据字典品种类型 ( <a href="#">VarietyCategory</a> ))
char	exchange_inst_groupid[16]	结算组代码
Price6	his_high_price	历史最高价
Price6	his_low_price	历史最低价
Qty2	latest_volume_trade	最新成交量
Qty2	init_volume_trade	初始持仓量
Qty2	change_volume_trade	持仓量变化
Qty2	bid_imply_volume	申买推导量

Qty2	offer_imply_volume	申卖推导量
char	arbi_type	策略类别
char	instrument_id_1[32]	第一腿合约代码
char	instrument_id_2[32]	第二腿合约代码
char	instrument_name[32]	合约名称
Qty2	total_bid_volume_trade	总买入量
Qty2	total_ask_volume_trade	总卖出量

#### 7.1.12. 上交所中证指数快照结构定义(MDCSIIndexSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
Price6	last_index	最新指数
Price6	open_index	今开盘指数
Price6	high_index	最高指数
Price6	low_index	最低指数
Price6	close_index	收盘指数
Price6	pre_close_index	前收盘指数
Rate6	change	涨跌
Rate6	ratio_of_change	涨跌幅
Qty2	total_volume_trade	成交量
Amt5	total_value_trade	总成交金额(单位为万元)
Rate8	exchange_rate	汇率
char	currency_symbol	币种标志(0-人民币 1-港币 2-美元 3-台币 4-日元)
Price6	close_index2	当日收盘 2
Price6	close_index3	当日收盘 3
uint8_t	index_market	指数市场



char	md_stream_id[6]	行情类别 JLLX
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.13. 深交所成交量统计指标快照结构定义(MDIndicatorOfTradingVolumeSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
Qty2	total_volume_trade	总成交数
Amt5	total_value_trade	总成交额
Price6	pre_close_price	昨收价
uint32_t	stock_num	统计量指标样本个数
char	trading_phase_code[8]	交易阶段代码 深圳成交量统计指标快照交易状态 第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价 'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断 第 1 位:'0'= 正常状态 '1'= 全天停牌
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.14. 深交所国证指数快照结构定义(MDCnIndexSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
char	trading_phase_code[8]	交易阶段代码

		<p>深圳成交量统计指标快照交易状态</p> <p>第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断</p> <p>第 1 位:'0'= 正常状态 '1'= 全天停牌</p>
Price6	pre_close_index	前收盘指数
Price6	open_index	今开盘指数
Price6	high_index	最高指数
Price6	low_index	最低指数
Price6	last_index	最新指数
Price6	close_index	收盘指数
Price6	close_index2	收盘指数 2
Price6	close_index3	收盘指数 3
Qty2	total_volume_trade	参与计算相应指数的交易数量
Amt5	total_value_trade	参与计算相应指数的成交总金额
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别 (取值参照公共数据字典品种类型 ( <a href="#">VarietyCategory</a> ))

#### 7.1.15. 深交所转融通证券出借逐笔委托结构定义(MDRefinancingTickOrder)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int32_t	channel_no	频道编号
int64_t	appl_seq_num	消息记录号
int64_t	order_time	委托时间 (YYYYMMDDHHMMSSsss)
Price6	order_price	委托价格
Qty2	order_volume	委托数量
uint8_t	side	买卖方向 (1-买 2-卖 G-借入 F-出借)

uint16_t	expiration_days	期限
uint8_t	expiration_type	期限类型(1-固定期限)
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.16. 深交所转融通证券出借逐笔成交结构定义(MDRefinancingTickExecution)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	exec_time	成交时间 YYYYMMDDHHMMSSsss
int32_t	channel_no	频道编号
int64_t	appl_seq_num	消息记录号
Price6	exec_price	成交价格
Qty2	exec_volume	成交数量
Amt5	value_trade	成交金额
int64_t	bid_appl_seq_num	买方委托索引
int64_t	offer_appl_seq_num	卖方委托索引
uint8_t	side	买卖方向
uint8_t	exec_type	成交类型(仅深圳有效 4-撤销 F-成交)
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.17. 深交所协议交易逐笔委托结构定义(MDNegotiableTickOrder)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int32_t	channel_no	频道编号
int64_t	appl_seq_num	消息记录号

int64_t	order_time	委托时间 (YYYYMMDDHHMMSSsss)
Price6	order_price	委托价格
Qty2	order_volume	委托数量
uint8_t	side	买卖方向(1-买 2-卖 G-借入 F-出借)
char	confirm_id[8]	定价行情约定号,为空表示是意向行情, 否则为定价行情
char	contactor[12]	联系人
char	contact_info[30]	联系方式
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 ( <a href="#">VarietyCategory</a> ))

#### 7.1.18. 深交所协议交易逐笔成交结构定义(MDNegotiableTickExecution)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	exec_time	成交时间 YYYYMMDDHHMMSSsss
int32_t	channel_no	频道编号
int64_t	appl_seq_num	消息记录号
Price6	exec_price	成交价格
Qty2	exec_volume	成交数量
Amt5	value_trade	成交金额
int64_t	bid_appl_seq_num	买方委托索引
int64_t	offer_appl_seq_num	卖方委托索引
uint8_t	side	买卖方向
uint8_t	exec_type	成交类型(仅深圳有效 4-撤销 F-成交)
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 ( <a href="#">VarietyCategory</a> ))

### 7.1.19. 港股通实时额度结构定义(MDHKTRealtimeLimit)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
Amt5	threshold_amount	每日初始额度
Amt5	pos_amt	日中剩余额度
char	amount_status	额度状态 (1-额度用完或其他原因全市场禁止买入 2-额度可用)
int32_t	channel_no	频道代码
char	md_stream_id[6]	行情类别(仅上海有效)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
char	mkt_status[8]	上交所港股通市场状态(上交所独有,来源于上交所文件行情)

### 7.1.20. 港股通可接收订单并转发的产品状态结构定义(MDHKTProductStatus)

数据类型	字段名称	说明
int32_t	market_type;	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16];	证券代码
int64_t	orig_time;	时间(YYYYMMDDHHMMSSsss)
char	trading_status1[8];	<p>证券交易状态(整手订单)</p> <p>港股通整手订单:</p> <p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填空格。</p> <p>第 1 位:'0'表示限制买入,'1'表示正常无此限制。</p> <p>第 2 位:'0'表示限制卖出,'1'表示正常无此限制。</p>
char	trading_status2[8];	<p>证券交易状态(零股订单)</p> <p>港股通零股订单:</p> <p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填空格。</p> <p>第 1 位:'0'表示限制买入,'1'表示正常无此限制。</p>

		第 2 位:'0'表示限制卖出,'1'表示正常无此限制。
int32_t	channel_no;	频道代码
char	md_stream_id[6];	行情类别(仅上海有效)
uint8_t	variety_category;	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.21. 港股通 VCM 结构定义(MDHKTVCMM)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	港股通代码
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
int64_t	start_time	市调机制开始时间
int64_t	end_time	市调机制结束时间
Price6	ref_price	市调机制参考价格
Price6	low_price	市调机制最低价格
Price6	high_price	市调机制最高价格
char	md_stream_id[6]	行情类别
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.22. 股转系统证券行情信息结构定义(MDNEEQSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间 CCYYMMDD + HHMMSS * 1000
Price6	pre_close_price	昨收价
Price6	open_price	开盘价
Price6	last_price	最新价
Qty2	total_volume_trade	成交总量

Amt5	total_value_trade	成交总金额
int64_t	num_trades	成交笔数
Price6	high_price	最高价
Price6	low_price	最低价
Rate6	price_earning_ratio1	市盈率 1
Rate6	price_earning_ratio2	市盈率 2
Rate6	change1	涨跌 1(对比昨收价)
Rate6	change2	涨跌 2(对比上一笔)
Qty2	open_interest	合约持仓量
Price6	bid_price[5]	申买价
Qty2	bid_volume[5]	申买量
Price6	offer_price[5]	申卖价
Qty2	offer_volume[5]	申卖量
Rate6	index_factor	指数因子
char	trading_phase_code[8]	交易阶段代码 北交所证券行情状态。个位数存放收市行情标志(0:非收市行情;1:收市行情;2:盘后行情)十位数存放正式行情与测试行情标志(0:正式行情;1:测试行情)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.23. 股转系统证券信息结构定义(MDNEEQSecurityInfo)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间 CCYYMMDD + HHMMSSss * 10
char	security_abbreviation[32]	证券简称
char	underlying_security[16]	基础证券
char	ISIN[ConstField::16]	ISIN 编码

int32_t	trading_unit	交易单位
char	industry_type[8]	行业种类
char	currency[8]	货币种类 (00-人民币,02-美元)
Price6	par_value	每股面值
int64_t	general_capital	总股本
int64_t	unrestricted_capital	非限售股本
Price6	last_year_earning	上年每股收益
Price6	cur_year_earning	本年每股收益
Rate6	brokerage_rate	经手费率
Rate6	stamp_duty_rate	印花税率
Rate6	transfer_fee_rate	过户费率
char	listing_date[10]	挂牌日期
char	value_date[10]	起息日
char	expiring_date[10]	到期日
Qty2	every_limited	每笔限量
Qty2	buy_amount_unit	买数量单位
Qty2	sell_amount_unit	卖数量单位
Qty2	mini_dec_amount	最小申报数量
int32_t	price_level	价格档位
Price6	first_trade_limit	首笔交易限价参数
Price6	follow_trade_limit	后续交易限价参数
uint8_t	limit_param_nature	限价参数性质
Price6	high_limited	涨停价
Price6	low_limited	跌停价
Price6	block_trade_ceiling	大宗交易价格上限
Price6	block_trade_floor	大宗交易价格下限
char	component_mark	成分股标志
Rate6	conver_ratio	折合比例



char	trade_status	交易状态
char	security_level	证券级别
char	trade_type	交易类型
Qty2	market_maker_num	做市商数量
char	suspen_sign	停牌标志
char	ex_sign	除权除息标志
char	net_vote_sign	网络投票标志
char	other_buss_sign[8]	其他业务标志
char	record_update_time[10]	记录更新时间
uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 ( <a href="#">VarietyCategory</a> ))

#### 7.1.24. 股转非公开转让申报信息库结构定义(MDNEEQNonPublicTransDeclaredInfo)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间 CCYYMMDD + HHMMSS * 1000
char	transaction_unit[8]	交易单元
char	security_category[8]	证券类别
char	declare_category[8]	申报类别
Qty2	declare_volume	申报数量
Price6	declare_price	申报价格
int32_t	deal_agreement_num	成交约定号
char	declare_time[10]	申报时间
char	record_status	记录状态
char	backup_sign	备用标志
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.25. 股转系统分层信息库结构定义(MDNEEQHierarchicalInfo)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
char	trade_date[10]	交易日期 CCYYMMDD
char	security_abbreviation[64]	证券简称
char	layered_sign	分层标志
char	layered_effective_date[10]	分层生效日期 CCYYMMDD
char	backup_sign	备用标志
uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 ( <a href="#">VarietyCategory</a> ))

### 7.1.26. 港股通市场状态结构定义(MDHKMarketStatus)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	orig_time	时间(YYYYMMDDHHMMSSsss)
char	trading_session_sub_id[8]	市场状态
uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 ( <a href="#">VarietyCategory</a> ))

### 7.1.27. 北交所协议转让申报信息库结构定义(MDNEEQNegotiableDeclaredInfo)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间 CCYYMMDD + HHMMSS * 1000
char	transaction_unit[8]	交易单元
char	md_stream_id[6]	业务类别
Qty2	declare_volume	申报数量

Price6	declare_price	申报价格
int32_t	deal_agreement_num	成交约定号
char	declare_time[10]	申报时间
char	record_status	记录状态
char	backup_sign	备用标志
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.28. 北交所做市业务申报信息库结构定义(MDNEEQMarketMakerDeclaredInfo)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间 CCYYMMDD + HHMMSS * 1000
char	md_stream_id[6]	业务类别
Qty2	declare_volume	申报数量
Price6	declare_price	申报价格
char	data_type	数据类型
char	declare_time[10]	申报时间
int64_t	backup_field	备用字段(预留)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.29. 北交所非公开转让成交信息库结构定义(MDNEEQNonPublicTransferDealInfo)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	serial_num	序号
char	security_code[16]	证券代码
int64_t	orig_time	时间 CCYYMMDD + HHMMSS * 1000
char	security_abbreviation[64]	证券简称

char	security_category[8]	证券类别
char	bid_transaction_unit[8]	买入交易单元(预留)
char	bid_transaction_unit_name[128]	买入营业部名称/交易单元名称
char	offer_transaction_unit[8]	卖出交易单元(预留)
char	offer_transaction_unit_name[128]	卖出营业部名称/交易单元名称
Qty2	deal_volume	成交数量
Price6	deal_price	成交价格
char	deal_time[10]	成交时间
char	backup_sign	备用标志
uint8_t	variety_category	品 种 类 别 ( 取值 参 照 公 共 数 据 字 典 品 种 类 型 ( <a href="#">VarietyCategory</a> ))

### 7.1.30. 委托簿结构定义(MDOrderBook)

数据类型	字段名称	说明
int32_t	channel_no	频道号
int32_t	market_type	市场类型, 参考公共数据字典市场类型 ( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	last_tick_time	最新逐笔生成时间
int64_t	last_snapshot_time	最新快照生成时间(固定为 0, 无实际意义)
int64_t	last_tick_seq	最新逐笔序列号
std::vector<MDOrderBookItem>	bid_order_book	买委托簿, 取值参考 MDOrderBookItem 类型定义
std::vector<MDOrderBookItem>	offer_order_book	卖委托簿, 取值参考 MDOrderBookItem 类型定义
int64_t	total_num_trades	基于委托簿演算的成交总笔数
Qty2	total_volume_trade	基于委托簿演算的成交总量
Amt5	total_value_trade	基于委托簿演算的成交总金额
Price6	last_price	基于委托簿演算的最新价

### MDOrderBookItem 数据结构定义:

数据类型	字段名称	说明
Price6	price	价格
Qty2	volume	总数量
int64_t	order_queue_size	委托队列大小
int64_t	order_queue[50]	委托队列数量,最多揭示 50 笔

### 7.1.31. 债券快照结构定义(MDBondSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
char	trading_phase_code[8]	<p>交易阶段代码</p> <p>上海债券快照交易状态:</p> <p>该字段为 8 位字符数组,左起每位表示特定的含义,无定义则填充空格。</p> <p>第 0 位:'S'表示启动(开市前)时段,'C'表示开盘集合竞价时段,'T'表示连续交易时段,'E'表示闭市时段,'P'表示产品停牌。</p> <p>第 1 位:'0'表示此产品不可正常交易,'1'表示此产品可正常交易。</p> <p>第 2 位:'0'表示未上市,'1'表示已上市。</p> <p>第 3 位:'0'表示此产品在当前时段不接受进行新订单申报,'1'表示此产品在当前时段可接受进行新订单申报。</p> <p>深圳债券快照交易状态:</p> <p>第 0 位:'S'= 启动(开市前)'O'= 开盘集合竞价'T'= 连续竞价'B'= 休市'C'= 收盘集合竞价'E'= 已闭市'H'= 临时停牌'A'= 盘后交易'V'=波动性中断。</p> <p>第 1 位:'0'= 正常状态 '1'= 全天停牌。</p>
Price6	pre_close_price	昨收价
Price6	open_price	开盘价
Price6	high_price	最高价

Price6	low_price	最低价
Price6	last_price	最新价
Price6	close_price	收盘价
Price6	bid_price[10]	申买价
Qty2	bid_volume[10]	申买量
Price6	offer_price[10]	申卖价
Qty2	offer_volume[10]	申卖量
int64_t	num_trades	成交笔数
Qty2	total_volume_trade	成交总量
Amt5	total_value_trade	成交总金额
Qty2	total_bid_volume	委托买入总量
Qty2	total_offer_volume	委托卖出总量
Price6	weighted_avg_bid_price	加权平均为委买价格
Price6	weighted_avg_offer_price	加权平均为委卖价格
Price6	high_limited	涨停价
Price6	low_limited	跌停价
Rate6	change1	涨跌 1(对比昨收价)(仅深圳有效)
Rate6	change2	涨跌 2(对比上一笔)(仅深圳有效)
Rate6	weighted_avg_bp	加权平均利率涨跌 BP(债券质押式回购)(仅深圳有效)
Rate6	pre_close_weighted_avg_price	昨收盘加权平均价(债券质押式回购)(仅深圳有效)
Price6	auct_last_price	匹配成交最近价(仅深圳有效)
uint8_t	last_price_trading_type	最近价成交方式(仅深圳有效)
int32_t	channel_no	频道代码(仅深圳有效)
char	md_stream_id[6]	行情类别
char	instrument_status[8]	当前品种交易状态(仅上海有效)  交易状态:  ADD--- 产品未上市; START--- 启动;OCALL--- 开市集合竞价;TRADE--- 连续自动撮合;SUSP--- 停牌;CLOSE--- 闭

		市;ENDTR---交易结束
int64_t	withdraw_buy_number	买入撤单笔数(仅上海有效)
Qty2	withdraw_buy_amount	买入撤单数量(仅上海有效)
Amt5	withdraw_buy_money	买入撤单金额(仅上海有效)
int64_t	withdraw_sell_number	卖出撤单笔数(仅上海有效)
Qty2	withdraw_sell_amount	卖出撤单数量(仅上海有效)
Amt5	withdraw_sell_money	卖出撤单金额(仅上海有效)
int64_t	total_bid_number	买入总笔数(仅上海有效)
int64_t	total_offer_number	卖出总笔数(仅上海有效)
int32_t	bid_trade_max_duration	买入委托成交最大等待时间(仅上海有效)
int32_t	offer_trade_max_duration	卖出委托成交最大等待时间(仅上海有效)
int32_t	num_bid_orders	买方委托价位数(仅上海有效)
int32_t	num_offer_orders	卖方委托价位数(仅上海有效)
int64_t	last_trade_time	最近成交时间(为 YYYYMMDDHHMMSSsss 仅上海有效)
Price6	weighted_avg_price	加权平均价
uint32_t	no_sub_trading_phase_code	细分交易阶段个数(仅深圳有效)
SubTradingPhase	sub_trading_phase[8]	细分交易阶段信息(仅深圳有效)
Qty2	auct_volume_trade	匹配成交成交量(仅深圳有效)
Amt5	auct_value_trade	匹配成交成交金额(仅深圳有效)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### SubTradingPhase 数据结构定义:

数据类型	字段名称	说明
char	sub_trading_phase_code[8]	交易方式所处的交易阶段代码 该字段为 8 位字符数组 第 0 位:S=启动(开市前) O=开盘集合竞价 T=连续竞价 B=休市 E=已闭市 H=临时停牌 V=断路器阶段/停市恢复集合竞价
		交易方式

uint8_t	trading_type	1=匹配成交 2=协商成交 3=点击成交 4=询价成交 5=竞买成交
---------	--------------	------------------------------------

### 7.1.32. 债券逐笔委托结构定义(MDBondTickOrder)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	appl_seq_num	消息记录号
int32_t	channel_no	频道编号
char	security_code[16]	证券代码
int64_t	order_time	委托时间 (YYYYMMDDHHMMSSsss)
Price6	order_price	委托价格
Qty2	order_volume	委托数量
uint8_t	side	买卖方向 深圳市场: 1-买 2-卖 G-借入 F-出借 上海市场: 当 order_type = 'S'时,此字段无意义,取值默认为'\0'. 当 order_type = 'A'   order_type = 'D'时,意义如下: B:买单,S:卖单
uint8_t	order_type	订单类别 深圳市场:(1-市价 2-限价 U-本方最优) 上海市场:(A:增加委托,D:删除委托,S:产品状态订单)
char	md_stream_id[6]	行情类别(仅深圳有效)
char	product_status[8]	产品状态(仅上海有效) 当 order_type = 'A'   order_type = 'D'时,此字段无意义,取值默认为'\0'. 当 order_type = 'S'时,意义如下: ADD--- 产品未上市;START--- 启动;OCALL--- 开市集合竞价; TRADE---连续自动撮合;SUSP---停牌;CLOSE---闭市;ENDTR---交易结束;
int64_t	orig_order_no	原始订单号
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))



### 7.1.33. 债券逐笔成交结构定义(MDBondTickExecution )

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	appl_seq_num	消息记录号
int32_t	channel_no	频道编号
char	security_code[16]	证券代码
int64_t	exec_time	成交时间 YYYYMMDDHHMMSSsss
Price6	exec_price	成交价格
Qty2	exec_volume	成交数量
Amt5	value_trade	成交金额(仅上海有效)
int64_t	bid_appl_seq_num	买方委托索引
int64_t	offer_appl_seq_num	卖方委托索引
uint8_t	side	买卖方向(仅上海有效 B: 买单, S 卖单)
uint8_t	exec_type	成交类型(深圳: 4-撤销 F-成交,上海: F-成交)
char	md_stream_id[6]	行情类别(仅深圳有效)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

### 7.1.34. 深交所债券业务报价及大额逐笔委托结构定义(MDBondQuotedTickOrder )

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	appl_seq_num	消息记录号
int32_t	channel_no	频道编号
char	security_code[16]	证券代码
int64_t	order_time	委托时间(YYYYMMDDHHMMSSsss)
Price6	order_price	委托价格
Qty2	order_volume	委托数量

uint8_t	side	买卖方向(1-买 2-卖 G-借入 F-出借)
char	md_stream_id[6]	行情类别
char	quote_id[10]	报价消息编号
char	member_id[6]	交易商代码
char	investor_type[2]	交易主体类型(01-自营 02-资管 03-机构经济 04-个人经济)
char	investor_id[10]	交易主体代码
char	investor_name[120]	客户名称
char	trader_code[8]	交易员代码
uint8_t	settl_period	结算周期
uint16_t	settl_type	结算方式(103-多边净额 104-逐笔全额)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
char	secondary_order_id[16]	竞买场次编号(仅债券竞买逐笔数据有效)
uint32_t	bid_trans_type	竞买业务类别( 1=竞买预约申报 2=竞买发起申报 3=竞买应价申报 仅债券竞买逐笔数据有效)
uint32_t	bid_execinst_type	竞买成交方式( 1=单一主体中标 2=多主体单一价格中标 3=多主体多重价格中标 仅债券竞买逐笔数据有效)
Price6	lowlimit_price	价格下限(仅债券竞买逐笔数据有效)
Price6	highlimit_price	价格上限( 0=无价格上限 仅债券竞买逐笔数据有效)
Qty2	min_qty	最低成交数量(仅债券竞买逐笔数据有效)
uint32_t	trade_date	交易日期(YYYYMMDD 仅债券竞买逐笔数据有效)

### 7.1.35. 深交所债券业务报价及大额逐笔成交结构定义(MDBondQuotedTickExecution)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	appl_seq_num	消息记录号
int32_t	channel_no	频道编号
char	security_code[16]	证券代码
int64_t	exec_time	成交时间 YYYYMMDDHHMMSSsss

Price6	exec_price	成交价格
Qty2	exec_volume	成交数量
int64_t	bid_appl_seq_num	买方委托索引
int64_t	offer_appl_seq_num	卖方委托索引
uint8_t	exec_type	成交类型(4-撤销 F-成交)
char	md_stream_id[6]	行情类别
uint8_t	settl_period	结算周期
uint16_t	settl_type	结算方式(103-多边净额 104-逐笔全额)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
char	secondary_order_id[16]	竞买场次编号(仅债券竞买逐笔数据有效)
uint32_t	bid_execinst_type	竞买成交方式(1=单一主体中标 2=多主体单一价格中标 3=多主体多重价格中标 仅债券竞买逐笔数据有效)
Price6	margin_price	达成成交的边际价格(竞买成交方式为多主体单一价格中标或多主体多重价格中标时用于揭示竞买成交的边际价格 仅债券竞买逐笔数据有效)

### 7.1.36. 委托簿快照结构定义(MDOrderBookSnapshot)

数据类型	字段名称	说明
uint8_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
uint8_t	variety_category	品种类别,参照公共数据字典品种类型( <a href="#">VarietyCategory</a> )
char	security_code[16]	证券代码
int64_t	last_tick_seq	构建快照的最新逐笔记录号
int32_t	channel_no	构建快照的逐笔原始频道编号
int64_t	orig_time	基于委托簿演算的行情快照时间(最新逐笔记录号对应的逐笔时间)(YYYYMMDDHHMMSSsss)
Price6	last_price	基于委托簿演算的最新价
int64_t	total_num_trades	基于委托簿演算的成交总笔数
Qty2	total_volume_trade	基于委托簿演算的成交总量
Amt5	total_value_trade	基于委托簿演算的成交总金额

Qty2	total_bid_volume	基于委托簿演算的委托买入总量
Qty2	total_offer_volume	基于委托簿演算的委托卖出总量
int64_t	num_bid_orders	基于委托簿演算的买方委托价位数
int64_t	num_offer_orders	基于委托簿演算的卖方委托价位数
Price6	bid_price[10]	申买价
Qty2	bid_volume[10]	申买量
Price6	offer_price[10]	申卖价
Qty2	offer_volume[10]	申卖量

### 7.1.37. 代码表结构定义(CodeTableRecord)

数据类型	字段名称	说明
char	security_code[32]	证券代码
uint8_t	market_type	证券市场,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	symbol[128]	简称
char	english_name[64]	英文名
char	security_type[16]	证券子类别(参考公共数据字典的证券子类别( <a href="#">Security_type</a> ))
char	currency[8]	币种(参考公共数据字典的币种( <a href="#">Currency</a> ))
uint8_t	variety_category	证券类别
Price6	pre_close_price	昨收价
Price6	close_price	收盘价(已弃用)
char	underlying_security_id[16]	标的代码(仅期权/权证/期货期权有效)
char	contract_type[16]	合约类别(仅期权/期货期权有效)
Price6	exercise_price	行权价(仅期权/期货期权有效)
uint32_t	expire_date	到期日(仅期权/期货期权有效)
Price6	high_limited	涨停价
Price6	low_limited	跌停价
char	security_status[16]	产品状态标志(一只代码可能存在多个状态标志,比如说"1,2,3",具体含义参考公共数据字典的产品状态标志( <a href="#">Security_status</a> ))

Price6	price_tick	最小价格变动单位
Qty2	buy_qty_unit	限价买数量单位
Qty2	sell_qty_unit	限价卖数量单位
Qty2	market_buy_qty_unit	市价买数量单位
Qty2	market_sell_qty_unit	市价卖数量单位
Qty2	buy_qty_lower_limit	限价买数量下限
Qty2	buy_qty_upper_limit	限价买数量上限
Qty2	sell_qty_lower_limit	限价卖数量下限
Qty2	sell_qty_upper_limit	限价卖数量上限
Qty2	market_buy_qty_lower_limit	市价买数量下限
Qty2	market_buy_qty_upper_limit	市价买数量上限
Qty2	market_sell_qty_lower_limit	市价卖数量下限
Qty2	market_sell_qty_upper_limit	市价卖数量上限
uint32_t	list_day	上市日期
Price6	par_value	面值
Qty2	outstanding_share	总发行量(上交所不支持)
Qty2	public_float_share_quantity	流通股数(上交所不支持)
Rate5	contract_multiplier	对回购标准券折算率
char	regular_share[9]	对应回购标准券(仅深交所)
Rate8	interest	应计利息
Rate6	coupon_rate	票面年利率
char	product_code	期货品种产品代码(仅期货期权有效)
uint32_t	delivery_year	交割年份(仅期货期权有效)
uint32_t	delivery_month	交割月份(仅期货期权有效)
uint32_t	create_date	创建日期(仅期货期权有效)
uint32_t	start_deliv_date	开始交割日(仅期货期权有效)
uint32_t	end_deliv_date	结束交割日(仅期货期权有效)
uint32_t	position_type	持仓类型(仅期货期权有效)

### 7.1.38. ETF 代码表结构定义(ETFCodeTableRecord)

数据类型	字段名称	说明
char	security_code[16]	证券代码
Qty2	creation_redemption_unit	每个篮子对应的 ETF 份数
Rate6	max_cash_ratio	最大现金替代比例
char	publish	是否发布 IOPV,Y=是,N=否
char	creation	是否允许申购,Y=是,N=否(仅深圳有效)
char	redemption	是否允许赎回,Y=是,N=否(仅深圳有效)
char	creation_redemption_switch	申购赎回切换(仅上海有效,0---不允许申购/赎回, 1---申购和赎回皆允许, 2---仅允许申购, 3---仅允许赎回)
Qty2	record_num	深市成份证券数目
Qty2	total_record_num	所有成份证券数量
Amt5	estimate_cash_component	预估现金差额
int64_t	trading_day	当前交易日(格式:YYYYMMDD)
int64_t	pre_trading_day	前一交易日(格式:YYYYMMDD)
Amt5	cash_component	前一日现金差额
Price6	nav_per_cu	前一日最小申赎单位净值
Price6	nav	前一日基金份额净值
uint8_t	market_type	证券所属市场(参考公共数据字典市场类型( <a href="#">MarketType</a> ))
char	symbol[128]	基金名称(仅深圳有效)
char	fund_management_company[128]	基金公司名称(仅深圳有效)
char	underlying_security_id[16]	拟合指数代码(仅深圳有效)
char	underlying_security_id_source[4]	拟合指数代码源(仅深圳有效)
Amt5	dividend_per_cu	红利金额
Qty2	creation_limit	累计申购总额限制,为 0 表示没有限制(仅深圳有效)
Qty2	redemption_limit	累计赎回总额限制,为 0 表示没有限制(仅深圳有效)
Qty2	creation_limit_per_user	单个账户累计申购总额限制,为 0 表示没有限制(仅深圳有效)

Qty2	redemption_limit_per_user	单个账户累计赎回总额限制,为 0 表示没有限制(仅深圳有效)
Qty2	net_creation_limit	净申购总额限制,为 0 表示没有限制(仅深圳有效)
Qty2	net_redemption_limit	净赎回总额限制,为 0 表示没有限制(仅深圳有效)
Qty2	net_creation_limit_per_user	单个账户净申购总额限制,为 0 表示没有限制(仅深圳有效)
Qty2	net_redemption_limit_per_user	单个账户净赎回总额限制,为 0 表示没有限制(仅深圳有效)
char	all_cash_flag	是否支持全现金申赎(暂时未启用,取值为空)
char	all_cash_amount[12]	全现金替代的总金额(暂时未启用,取值为空)
char	all_cash_premium_rate[7]	全现金替代的申购溢价比例(暂时未启用,取值为空)
char	all_cash_discount_rate[7]	全现金替代的赎回折价比例(暂时未启用,取值为空)
char	rtgs_flag	是否支持 RTGS(暂时未启用,取值为空)
char	reserved[30]	预留字段(暂时未启用,取值为空)
ConstituentStockInfo	constituent_stock_infos	成分股信息

### ConstituentStockInfo 定义:

数据类型	字段名称	说明
char	security_code[32]	成份证券代码
uint8_t	market_type	成份证券所属市场(仅深圳有效,参考公共数据字典市场类型( <a href="#">MarketType</a> ))
char	underlying_symbol[128]	成份证券简称
Qty2	component_share	成份证券数量
char	substitute_flag	现金替代标志: //深圳现金替代标志 0---禁止现金替代(必须有证券),1---可以进行现金替代(先用证券,证券不足时差额部分用现金替代),2---必须用现金替代 //上海现金替代标志: ETF 公告文件 1.0 版格式 0---沪市不可被替代, 1---沪市可以被替代, 2---沪市必须被替代, 3---深市退补现金替代, 4---深市必须现金替代,5---非沪深市场成份证券退补现金替代(不适用于跨沪深港 ETF 产



		品), 6---非沪深市场成份证券必须现金替代(不适用于跨沪深港 ETF 产品)  ETF 公告文件 2.1 版格式  0---沪市不可被替代, 1---沪市可以被替代, 2---沪市必须被替代, 3---深市退补现金替代, 4---深市必须现金替代, 5---非沪深市场成份证券退补现金替代(不适用于跨沪深港 ETF 产品), 6---非沪深市场成份证券必须现金替代(不适用于跨沪深港 ETF 产品) 7---港市退补现金替代(仅适用于跨沪深港 ETF 产品), 8---港市必须现金替代(仅适用于跨沪深港 ETF 产品)
Rate6	premium_ratio	溢价比例
Rate6	discount_ratio	折价比例
Amt5	creation_cash_substitute	申购替代金额(仅深圳有效)
Amt5	redemption_cash_substitute	赎回替代金额(仅深圳有效)
Amt5	substitution_cash_amount	替代总金额(仅上海有效)
char	underlying_security_id[4]	成份证券所属市场 ID(仅对跨市场债券(银行间)ETF 启用)
char	buy_or_sell_to_open	期权期货买入开仓或卖出开仓(暂时未启用,取值为空)
char	Reserved[30]	预留字段(暂时未启用,取值为空)

### 7.1.39. 基金通快照结构定义(MDFundExpertSnapshot)

数据类型	字段名称	说明
uint8_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
uint8_t	variety_category	品 种类别 (取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
int32_t	channel_no	频道代码(仅深交所)
char	md_stream_id[6]	行情类别
char	security_code[16]	证券代码
int64_t	orig_time	行情发送时间(为 YYYYMMDDHHMMSSsss)
char	trading_phase_code[8]	//上海产品实时阶段及标志  该字段为 8 位字符串,左起每位表示特定的含义,无定义则填充格。  第 0 位:'S'表示启动(开市前)时段,'C'表示开盘集合竞价时段,'T'表示连续交易时段,'E'表示闭市时段,'P'表示产品停牌,'M'表示可恢复交易的熔断时段(盘中集合



		<p>竞价),'N'表示不可恢复交易的熔断时段(暂停交易至闭市),'U'表示收盘集合竞价时段。第 1 位:'0'表示此产品不可正常交易,'1'表示此产品可正常交易,无意义填充空格。第 2 位:'0'表示未上市,'1'表示已上市。第 3 位:'0'表示此产品在当前时段不接受订单申报,'1'表示此产品在当前时段可接受订单申报。无意义填充空格。</p> <p>//深圳交易阶段代码</p> <p>第 0 位:'S'= 启动(开市前)'T'= 连续竞价'B'= 休市'E'= 已闭市'H'= 临时停牌。第 1 位:'0'= 正常状态'1'= 全天停牌</p>
char	instrument_status[8]	<p>当前品种交易状态(仅上交所)</p> <p>//上海市场状态</p> <p>0 = 全日收市 1 = 输入买卖盘(开盘集合竞价时段),2 = 对盘(开盘集合竞价时段),3 = 持续交易,4 = 对盘(收盘集合竞价时段),5 = 输入买卖盘(收盘集合竞价时段),7 = 暂停,100 = 未开市,101 = 不可取消(开盘集合竞价时段),102 = ExchangeIntervention,103 = 收市,104 = 取消买卖盘,105 = 参考价定价(收盘集合竞价时段),106 = 不可取消(收盘集合竞价时段),107 = 随机收市(收盘集合竞价时段),108=随机对盘(开盘集合竞价时段)</p>
Price6	pre_close_price	昨收价
Price6	open_price	SH-开盘价(仅上交所)
Price6	high_price	最高价
Price6	low_price	最低价
Price6	last_price	最新价
Price6	per_price	SH-加权平均价(仅上交所)
Price6	close_price	SH-今收盘价(仅上交所)
Price6	bid_price[10]	申买价(做市商)
Qty2	bid_volume[10]	申买量(做市商)
Price6	offer_price[10]	申卖价(投资者)
Qty2	offer_volume[10]	申卖量(投资者)
int64_t	num_trades	成交笔数

Qty2	total_volume_trade	成交总量
Amt5	total_value_trade	成交总金额
Qty2	market_maker_bid_volume	委托买入总量(仅深交所, 做市商买入总量)
Qty2	sh_investor_offer_volume	委托卖出总量(仅深交所, 投资者卖出总量)
Price6	market_maker_bid_price	加权平均委买价格(仅深交所, 做市商加权平均价)
Price6	sh_investor_offer_price	加权平均委卖价格(仅深交所, 投资者加权平均价)
Price6	high_limited	涨停价(仅深交所)
Price6	low_limited	跌停价(仅深交所)
Price6	investor_bid_price	投资者买入均价(仅上交所)
Qty2	investor_bid_volume	投资者买入总量(仅上交所)
Price6	investor_best_bid_price	投资者买入最优价(仅上交所)
Qty2	investor_bid_volume_best_price	投资者买入最优价数量(仅上交所)
Price6	investor_offer_price	投资者卖出均价(仅上交所)
Qty2	investor_offer_volume	投资者卖出总量(仅上交所)
Price6	investor_best_offer_price	投资者卖出最优价(仅上交所)
Qty2	investor_offer_volume_best_price	投资者卖出最优价数量(仅上交所)
int64_t	iopv	基金 IOPV(仅上交所 MDStreamID=MD601 时存在该字段)
int64_t	last_trade_time	最近成交时间(仅上交所)

#### 7.1.40. IOPV 快照数据信息结构定义(MDIOPVSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	最新快照时间
int64_t	last_iopv	最新 IOPV
int64_t	bid_iopv[10]	买档位 IOPV
int64_t	offer_iopv[10]	卖档位 IOPV

### 7.1.41. 国际市场汇率结构定义(IMCExchangeRate)

数据类型	字段名称	说明
char	from_currency[8]	源货币种类
char	to_currency[8]	目标货币种类
Price6	bid_rate	参考汇率买入价
Price6	offer_rate	参考汇率卖出价
Price6	mid_point_rate	参考汇率中间价
int64_t	rate_date	汇率适用日期

### 7.1.42. 确定报价固收行情数据信息结构定义(MDZQQDBJSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
char	security_code[16]	证券代码
Rate6	interest	应计利息(单位: 0.1 厘)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))
int32_t	bid_item_cnt	买入明细个数
MDZQQDBJItem*	bid_item	买入明细
int32_t	offer_item_cnt	卖出明细个数
MDZQQDBJItem*	offer_item	卖出明细

### MDZQQDBJItem 定义:

数据类型	字段名称	说明
char	order_code[11]	订单编号
int32_t	bid_quote_time	报价时间(HHMMSS)
char	bid_quote_id[16]	报价方
Price6	bid_net_price	价格(净价)
Qty2	bid_volume	数量
Price6	bid_overhead_price	全价

Rate6	bid_expire_yield	到期收益率(百分数,4 位小数)
-------	------------------	------------------

#### 7.1.43. 成交行情固收行情数据信息结构定义(ZQCJHQSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
char	security_code[16]	证券代码
int32_t	last_trade_time	最新成交时间(HHMMSS)
Price6	pre_close_price	昨收价
Price6	pre_weighted_avg_price	昨日加权平均价
Price6	open_price	开盘价
Price6	high_price	最高价
Price6	low_price	最低价
Price6	last_price	最新价
Price6	weighted_avg_price	加权平均价
Qty2	total_volume_trade	总成交量
Amt5	total_value_trade	总成交金额
int64_t	total_num_trade	总成交笔数
Rate6	pre_close_yield_rate	昨收收益率(百分数,4 位小数)
Rate6	pre_weighted_avg_yield_rate	昨日加权平均收益率(百分数,4 位小数)
Rate6	open_yield_rate	开盘收益率(百分数,4 位小数)
Rate6	high_yield_rate	最高收益率(百分数,4 位小数)
Rate6	last_yield_rate	最新收益率(百分数,4 位小数)
Rate6	low_yield_rate	最低收益率(百分数,4 位小数)
Rate6	weighted_avg_yield_rate	加权平均收益率(百分数,4 位小数)
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

#### 7.1.44. 成交明细固收行情数据信息结构定义(ZQCJMXSnapshot)

数据类型	字段名称	说明
------	------	----

int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
char	security_code[16]	证券代码
int64_t	trade_time	成交时间(YYYYMMDDHHMMSS)
Price6	trade_net_price	成交净价
Rate6	interest	应计利息
Price6	trade_overhead_price	成交全价
Rate6	yield_to_maturity	到期收益率(百分数,4 位小数)
Qty2	trade_volume	成交量
Amt5	trade_value	成交金额
uint8_t	trade_mode	成交方式
uint8_t	variety_category	品种类别(取值参照公共数据字典品种类型( <a href="#">VarietyCategory</a> ))

## 7.2. \*商业港股数据字典

‘\*’指非标准支持类型,需要和上游服务端系统确认是否支持提供商业港股。

### 7.2.1. \*商业港股股票快照数据信息结构定义(MDHKExSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	行情时间(为 YYYYMMDDHHMMSSsss)
Price6	pre_close_price	昨收价
Price6	last_price	最新价
Price6	open_price	开盘价
Price6	high_price	最高价
Price6	low_price	最低价
Price6	close_price	收市价
Qty2	total_volume_trade	成交量

Amt5	total_value_trade	成交额
Qty2	short_volume_shares	沽空量
Amt5	short_value_turnover	沽空额
uint8_t	variety_category	品种类别 (取值参照公共数据字典品种类型 <a href="#">VarietyCategory</a> )

### 7.2.2. \*商业港股委托挂单数据信息结构定义(MDHKExOrderSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
MDHKExListItem	bid_list[40]	买单序列
MDHKExListItem	ask_list[40]	卖单序列
uint8_t	variety_category	品种类别 (取值参照公共数据字典品种类型 <a href="#">VarietyCategory</a> )

### MDHKExListItem 定义:

数据类型	字段名称	说明
Price6	order_price	挂单价格
Qty2	order_volume	挂单数量
int64_t	num_of_orders	挂单经纪席位数目

### 7.2.3. \*商业港股经纪席位数据信息结构定义(MDHKExOrderBrokerSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
uint8_t	side	买卖方向('B'买,'S'卖)
char	broker_flag	是否还有未展示完的席位排位(Y-是 N-否)
MDHKExDetailItem	Detail[40]	买卖经纪席位明细(所有档位的席位号数量之和最多有 40 个)

uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 <a href="#">VarietyCategory</a> )
---------	------------------	---

#### MDHKEExDetailItem 定义:

数据类型	字段名称	说明
uint8_t	level	档位信息
uint16_t	broker_num	席位号

#### 7.2.4. \*商业港股逐笔成交数据数据信息结构定义(MDHKEExTickExecution)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
int64_t	appl_seq_num	序号
uint8_t	side	买卖方向('B'主动买入,'-'中性盘,'S'主动卖出)
Price6	exec_price	委托价格
Qty2	exec_volume	委托数量
uint8_t	exec_type	成交类型 0-自动对盘 4-开市前成交盘 22-非自动对盘 100-同一证券商自动对盘 101-同一证券商非自动对盘 102-碎股交易 103-竞价交易
char	cancel_flag	撤单标识(Y 是/N 否)
uint8_t	variety_category	品 种 类 别 ( 取 值 参 照 公 共 数 据 字 典 品 种 类 型 <a href="#">VarietyCategory</a> )

#### 7.2.5. \*商业港股指数行情快照数据信息结构定义(MDHKEExIndexSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型( <a href="#">MarketType</a> )
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss)
char	index_status	指数状态
Price6	pre_close_price	昨收价

Price6	open_price	开盘价
Price6	high_price	最高价
Price6	low_price	最低价
Price6	close_price	收市价
Price6	last_price	最新价
Qty2	total_volume_trade	成交量
Amt5	total_value_trade	成交额
Rate6	change	涨跌
Rate6	ratio_of_change	涨跌%
Amt5	eas_value	预估结算值
uint8_t	variety_category	品 种类别 (取值参照公共数据字典品种类型 VarietyCategory)

#### 7.2.6. \*商业港股港股通成交额快照数据信息结构定义(MDHKEConnectTurnoverSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型,参考公共数据字典市场类型(MarketType)
char	market[3]	市场: SH--上海证券交易所, SZ--深圳证券交易所
int64_t	orig_time	时间 (为 YYYYMMDDHHMMSSsss)
char	Direction[3]	交易方向('NB' -- 北向交易, 'SB'-- 南向交易)
Amt5	buy_turnover	买入额
Amt5	sell_turnover	卖出额
Amt5	buy_sell_turnover	买卖总额
uint8_t	variety_category	品 种类别 (取值参照公共数据字典品种类型 VarietyCategory)

#### 7.3. \*美股数据字典

‘\*’指非标准支持类型,需要和上游服务端系统确认是否支持提供美股。



### 7.3.1. \*美股股票快照数据结构定义(MDUsaStockSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型
char	security_code[16]	证券代码
int64_t	orig_time	时间(为 YYYYMMDDHHMMSSsss 美东时间)
Price6	pre_close_price	昨收价(单位:元)
Price6	now_price	现价(区别于最新成交价.是经处理过的有效价格,用于画分时图/计算涨跌)
Price6	open_price	开盘价(单位:元)
Price6	high_price	最高价(仅盘中时段统计,单位:元)
Price6	low_price	最低价(仅盘中时段统计,单位:元)
Price6	close_price	收盘价(单位:元)
Qty2	total_volume_trade	总成交量(Nasdaq 交易所,统计全天盘前盘中盘后三个时段的所有成交,单位:股)
Amt5	total_value_trade	总成交额(全美成交额估算值,统计全天盘前盘中盘后三个时段,单位:元)
Qty2	all_market_vol	全美总成交量(全美交易所累计,统计全天盘前盘中盘后三个时段,单位:股)
Price6	exec_price	最新成交价(单位:元)
Qty2	exec_volume	最新成交量(当为负数,则表示撤单;当为 0,则表示开盘、收盘或 EOD 或者其他,可以过滤掉该笔逐笔,单位:股)
int32_t	regsho	卖空标识(表示消息发布时证券的卖空价格限制状态,可为空) 48 表示卖空限制解除 49 表示卖空限制生效 50 表示卖空限制继续
char	trade_status	交易状态(标识证券的当前交易状态) H 表示停止交易(全美范围) P 表示暂停交易(Nasdaq 上市的证券) Q 表示仅有效期报价(全美范围) T 表示交易中(Nasdaq 市场)

int32_t	trade_session_id	交易时段(标识该笔交易发生的时段)  -3 表示准备 03:00-04:00 -1 表示盘前 04:00-09:30 1 表示盘中 09:30-16:00 -2 表示盘后 16:00-20:00 0 表示无发生交易或不适用
int32_t	trade_market	交易所发生的市场  81 表示 The Nasdaq Stock Market 76 表示 Nasdaq/FINRA Trade Reporting Facility (TRF) Cartaret 50 表示 Nasdaq/FINRA Trade Reporting Facility (TRF) Chicago 66 表示 Nasdaq BX (BX) 88 表示 Nasdaq PSX (PSX)
char	exec_type[5]	成交类型(全美 eod 数据那一笔的成交类型为:eod)
int32_t	side	买卖方向(1 表示主动买入;0 表示中性盘;-1 表示主动卖出)

### 7.3.2. \*美股挂单快照结构定义(MDUsaOrderSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型
char	security_code[16]	证券代码
int64_t	orig_time	行情时间(为 YYYYMMDDHHMMSSsss 美东时间)
Price6	bid_price	买一价(单位:元)
Qty2	bid_volume	买一量(单位:股)
Price6	offer_price	卖一价(单位:元)
Qty2	offer_volume	卖一量(单位:股)

### 7.3.3. \*美股 EOD 收盘快照结构定义(MDUsaEodSnapshot)

数据类型	字段名称	说明
int32_t	market_type	市场类型
char	security_code[16]	证券代码
int64_t	orig_time	行情时间(为 YYYYMMDDHHMMSSsss 美东时间)

Price6	open_price	开盘价(单位:元)
Price6	high_price	最高价(单位:元)
Price6	low_price	最低价(单位:元)
Price6	close_price	收盘价(单位:元)
Qty2	all_market_vol	全美总成交量(单位:股)

## 8. 公共数据字典

### 8.1. 通道类型(ChannelMode)

通道类型	数值	说明
kRDMA	0x00000001	RDMA 抓包方式获取数据
kEXA	0x00000002	EXA 抓包方式获取数据
kAMI	0x00000004	AMI 组播方式获取数据
kTCP	0x00000008	TCP 流方式获取数据
kPCAP	0x00000010	libpcap 抓包方式获取数据
kMDDP	0x00000020	mddp 网关组播方式获取数据
kFPGA	0x00000040	FPGA 方式获取数据
kUDP	0x00000080	UDP 方式获取数据

### 8.2. 高可用模式类型(HighAvailableMode)

高可用模式	数值	说明
kMasterSlaveA	枚举(=0)	高可用模式的 A 模式,该模式下检测到高优先级通道可用时,会立即切换到高优先级通道接收数据
kMasterSlaveB	枚举(=1)	高可用模式的 B 模式,该模式下仅在当前使用通道出现问题需要切换时,才会切换到当前可用的最高优先级通道接收数据
kRegularDataFilter	枚举(=2)	规则数据过滤模式,该模式下上游通道以多活的方式接入,对数据进行滤重后下发

### 8.3. 日志输出级别(LogLevel)

日志输出级别	数值	说明
kTrace	枚举(=0)	跟踪级别日志

kDebug	枚举(=1)	调试级别日志
kInfo	枚举(=2)	普通级别日志
kWarn	枚举(=3)	警告级别日志
kError	枚举(=4)	错误级别日志
kFatal	枚举(=5)	致命级别日志

#### 8.4. 委托簿构建方式(OrderBookType)

委托簿构建方式	数值	说明
kNone	枚举(=0)	不开启委托簿
kLocalOrderBook	枚举(=1)	本地构建向服务端请求同步初始状态
kServerOrderBook	枚举(=2)	服务端远程 TCP 方式推送委托簿数据

#### 8.5. 事件级别(EventLevel)

事件级别	数值	说明
kInfo	枚举(=1)	普通事件
kWarn	枚举(=2)	告警事件
kError	枚举(=3)	错误事件,比较严重,需要介入处理

#### 8.6. 市场类型(MarketType)

市场类型	数值	说明
kNone	枚举(=0)	表示全市场
kNEEQ	枚举(=2)	北交所
kSHFE	枚举(=3)	上期所
kCFFEX	枚举(=4)	中金所
kDCE	枚举(=5)	大商所
kCZCE	枚举(=6)	郑商所
kINE	枚举(=7)	上海国际能源交易中心
kNASDAQ	枚举(=12)	纳交所
kSSE	枚举(=101)	上交所
kSZSE	枚举(=102)	深交所
kHKEx	枚举(=103)	港交所(港股行情数据通过深交所和上交所的港股通获取,市场类型为 kSZSE/kSSE,商业港股行情对应的市场类型为 kHKEx)

kCFETS	枚举(=104)	银行间市场
kMax	枚举(=150)	市场类型最大值

## 8.7. 品种类型(VarietyCategory)

品种类型	数值	说明
kNone	枚举(=0)	全部品种类型
kStock	枚举(=1)	股票
kFund	枚举(=2)	基金
kBond	枚举(=3)	债券
kOption	枚举(=4)	期权
kIndex	枚举(=5)	指数
kHKT	枚举(=6)	港股通
kFutureOption	枚举(=7)	期货/期货期权
kCFETSRMB	枚举(=8)	银行间本币交易产品
kHKEx	枚举(=9)	商业港股
kOthers	枚举(=255)	其他

## 8.8. 错误码(ErrorCode)

错误码	数值	说明
kFailure	枚举(= -100)	失败
kInitd	枚举(= -99)	已初始化
kUnInitd	枚举(= -98)	未初始化
kNullSpi	枚举(= -97)	未设置输出数据类指针
kSuccess	枚举(= 0)	成功

## 8.9. 订阅类型(SubscribeType)

订阅类型	数值	说明
kSet	枚举(=0)	重新设置订阅(先取消所有旧订阅信息, 然后增加新订阅信息)
kAdd	枚举(=1)	添加订阅
kDel	枚举(=2)	删除订阅
kCancelAll	枚举(=3)	取消所有订阅

## 8.10. 权限订阅数据类型(SubscribeDataType)

数据类型	权限订阅数据类型	数值	说明
static const uint64_t	kNone	0x0000000000000	订阅全部数据
static const uint64_t	kSnapshot	0x0000000000001	订阅现货快照数据
static const uint64_t	kTickExecution	0x0000000000002	订阅逐笔成交数据
static const uint64_t	kTickOrder	0x0000000000004	订阅逐笔委托数据
static const uint64_t	kOrderQueue	0x0000000000008	订阅委托队列数据
static const uint64_t	kIndexSnapshot	0x0000000000010	订阅指数快照数据
static const uint64_t	kFutureSnapshot	0x0000000000020	订阅期货快照数据
static const uint64_t	kOptionSnapshot	0x0000000000040	订阅期权快照数据
static const uint64_t	kHKTSnapshot	0x0000000000080	订阅港股通快照数据
static const uint64_t	kAfterHourFixedPriceSnapshot	0x0000000000100	订阅盘后定价快照数据
static const uint64_t	kAfterHourFixedPriceTickExecution	0x0000000000400	订阅上交所盘后定价逐笔成交数据
static const uint64_t	kCSIIIndexSnapshot	0x0000000000800	订阅中证指数快照数据
static const uint64_t	kNEEQSnapshot	0x0000000001000	订阅北交所快照数据
static const uint64_t	kFundExpertSnapshot	0x0000000002000	订阅基金通快照数据
static const uint64_t	kCFETSSnapshot	0x0000000004000	订阅银行间 CFETS 快照
static const uint64_t	kCFETSTickExecution	0x0000000008000	订阅银行间 CFETS 逐笔
static const uint64_t	kHKExSnapshot	0x000000010000	订阅商业港股行情快照
static const uint64_t	kHKExTickExecution	0x000000020000	订阅商业港股行情逐笔
static const uint64_t	kCFETSPrivateMarket	0x000000040000	订阅银行间 CFETS 本币私有行情
static const uint64_t	kUsaSnapshot	0x000000080000	订阅美股快照行情
static const uint64_t	kFixedIncomeSnapshot	0x000000100000	订阅固定收益快照数据
static const uint64_t	kHKExConnectTurnoverSnapshot	0x000000200000	订阅商业港股港股通成交额快照
static const uint64_t	kMDHKTR realtimeLimit	0x000000400000	订阅港股通实时额度数据
static const uint64_t	kMDHKMarketStatus	0x000000800000	订阅港股通市场状态数据

### 8.11. 证券数据类型(SubscribeSecuDataType)

数据类型	证券数据类型	数值	说明
static const uint64_t	kNone	0x0000000000000	订阅全部证券数据类别
static const uint64_t	kSnapshot	0x0000000000001	订阅快照数据类别
static const uint64_t	kTickExecution	0x0000000000002	订阅逐笔成交数据
static const uint64_t	kTickOrder	0x0000000000004	订阅逐笔委托数据
static const uint64_t	kOrderQueue	0x0000000000008	订阅委托队列数据

### 8.12. 证券品种类型(SubscribeCategoryType)

数据类型	证券品种类型	数值	说明
static const uint64_t	kNone	0x0000000000000	订阅全部证券品种类别
static const uint64_t	kStock	0x0000000000001	订阅股票证券品种类别
static const uint64_t	kFund	0x0000000000002	订阅基金证券品种类别
static const uint64_t	kBond	0x0000000000004	订阅债券证券品种类别
static const uint64_t	kIndex	0x0000000000008	订阅指数证券品种类别
static const uint64_t	kHKT	0x0000000000010	订阅港股通证券品种类别
static const uint64_t	kOption	0x0000000000020	订阅期权证券品种类别
static const uint64_t	kFutureOption	0x0000000000040	订阅期货期权证券品种类别
static const uint64_t	kCFETSRMB	0x0000000000080	订阅银行间本币交易产品
static const uint64_t	kHKEx	0x0000000000100	订阅商业港股证券品种类别
static const uint64_t	kOthers	0x1000000000000	订阅其他证券品种类别

### 8.13. 事件代码(EventCode)

事件代码	数值	说明
kUMSConnectSuccess	枚举(=1)	连接 UMS 成功
kUMSConnectFailed	枚举(=2)	连接 UMS 失败
kUMSLogonSuccess	枚举(=3)	登录 UMS 成功
kUMSLogonFailed	枚举(=4)	登录 UMS 失败
kUMSHeartbeatTimeout	枚举(=5)	与 UMS 心跳超时
kChannelRDMAInitSuccess	枚举(=6)	RDMA 通道开启成功



kChannelRDMAInitFailed	枚举(=7)	RDMA 通道开启失败
kChannelEXAInitSuccess	枚举(=8)	EXA 通道开启成功
kChannelEXAInitFailed	枚举(=9)	EXA 通道开启失败
kChannelPCAPInitSuccess	枚举(=10)	PCAP 通道开启成功
kChannelPCAPInitFailed	枚举(=11)	PCAP 通道开启失败
kCTPDASStreamStart	枚举(=12)	CTP 通道数据流开始
kCTPDASStreamFinished	枚举(=13)	CTP 通道数据流结束
kSzseBinaryDASStreamStart	枚举(=14)	SzseBinary 通道数据流开始
kSzseBinaryDASStreamFinished	枚举(=15)	SzseBinary 通道数据流结束
kSseLDDSDASStreamStart	枚举(=16)	LDDS 通道数据流开始
kSseLDDSDASStreamFinished	枚举(=17)	LDDS 通道数据流结束
kChannelAMIInitSuccess	枚举(=18)	AMI 通道开启成功
kChannelAMIInitFailed	枚举(=19)	AMI 通道开启失败
kChannelTCPInitSuccess	枚举(=20)	TCP 通道开启成功
kChannelTCPInitFailed	枚举(=21)	TCP 通道开启失败
kChannelTCPConnectSuccess	枚举(=22)	TCP 通道连接成功
kChannelTCPConnectFailed	枚举(=23)	TCP 通道连接失败
kChannelTCPLogonSuccess	枚举(=24)	TCP 通道登录成功
kChannelTCPLogonFailed	枚举(=25)	TCP 通道登录失败
kChannelTCPSCessionClosed	枚举(=26)	TCP 通道连接断开
kChannelTCPHeartbeatTimeout	枚举(=27)	TCP 通道会话心跳失败
kChannelTCPMarketDataDegrade	枚举(=28)	TCP 通道行情数据降级
kChannelTCPMarketDataUpgrade	枚举(=29)	TCP 通道行情数据升级
kSourceMasterSlaveChanged	枚举(=30)	主备源切换
kSFSEngineInitSuccess	枚举(=31)	SFS 获取代码表模块初始化成功
kSzseOrderBookRejoinSuccess	枚举(=32)	深圳市场委托簿同步成功
kSzseOrderBookRejoinFailed	枚举(=33)	深圳市场委托簿同步失败
kSseOrderBookRejoinSuccess	枚举(=34)	上海市场委托簿同步成功



kSseOrderBookRejoinFailed	枚举(=35)	上海市场委托簿同步失败
kOrderBookRejonConnectFailed	枚举(=36)	连接上游服务端委托簿失败
kIncompleteData	枚举(=37)	上游服务端委托簿 recover 方式重启,数据有可能不完整
kOrderBookLogonSuccess	枚举(=38)	务端委托簿主动推送模式登陆成功(委托簿属于衍生数据,此信号后续会逐步弃用,用kDerivedDataLogonSuccess 信号代替)
kOrderBookLogonFailed	枚举(=39)	服务端委托簿主动推送模式登陆失败(委托簿属于衍生数据,此信号后续会逐步弃用,用kDerivedDataLogonFailed 信号代替)
kDerivedDataLogonSuccess	枚举(=40)	行情衍生数据服务端登陆成功
kDerivedDataLogonFailed	枚举(=41)	行情衍生数据服务端登陆失败
kChannelFPGAInitSuccess	枚举(=42)	FPGA 通道开启成功
kChannelFPGAInitFailed	枚举(=43)	FPGA 通道开启失败

#### 8.14. 证券子类别(Security\_type)

宏定义	类型值	说明
ID_BT_INDEX	"01000"	指数
ID_BT_SHARES_A	"02001"	主板 A 股
ID_BT_SHARES_B	"02002"	主板 B 股
ID_BT_SHARES_G	"02003"	创业板股
ID_BT_SHARES_KCB	"02004"	科创板股
ID_BT_SHARES_LIST	"02005"	股转系统挂牌股
ID_BT_SHARES_PRE	"02006"	优先股
ID_BT_SHARES_PZ_ZB	"02007"	主板存托凭证
ID_BT_SHARES_PZ_CYB	"02008"	创业板存托凭证
ID_BT_SHARES_PZ_KCB	"02009"	科创板存托凭证
ID_BT_SHARES_OTE	"02999"	其他股票
ID_BT_FUND_ET	"03001"	交易型开放式指数基金(ETF)
ID_BT_FUND_LO	"03002"	上市型开放式基金(LOF)

ID_BT_FUND_GR	"03003"	分级子基金
ID_BT_FUND_OPE	"03004"	未上市开放基金(仅申赎)
ID_BT_FUND_CLOS	"03005"	封闭式基金
ID_BT_FUND_REIT	"03006"	基础设施基金(REITs)
ID_BT_FUND_OT	"03999"	其他基金
ID_BT_BOND_CO	"04001"	可转债
ID_BT_BOND_N	"04002"	国债(含地方债)
ID_BT_BOND_COM	"04003"	公司债
ID_BT_BOND_EN	"04004"	企业债
ID_BT_BOND_P	"04005"	私募债
ID_BT_BOND_PP_E	"04006"	可交换私募债
ID_BT_BOND_COM_S	"04007"	证券公司次级债
ID_BT_BOND_S	"04008"	证券公司短期债
ID_BT_BOND_OT	"04999"	其他债券
ID_BT_REPO_ZY	"05001"	质押式回购
ID_BT_REPO_BOND_MD	"05002"	买断式债券回购
ID_BT_REPO_B	"05003"	报价回购
ID_BT_OPTION_C_G	"06001"	个股认购期权
ID_BT_OPTION_P_G	"06002"	个股认沽期权
ID_BT_OPTION_C_ID	"06003"	指数认购期权
ID_BT_OPTION_P_IDX	"06004"	指数认沽期权
ID_BT_OPTION_C_CMD	"06005"	商品认购期权(规划中)
ID_BT_OPTION_P_CM	"06006"	商品认沽期权(规划中)
ID_BT_OPTION_S	"06007"	商品期货(规划中)
ID_BT_OPTION_ID	"06008"	指数期货(规划中)
ID_BT_OPTION_SHARE	"06009"	股票期货(规划中)
ID_BT_OPTION_OT	"06010"	其他期货(规划中)
ID_BT_Q	"07001"	权证

ID_BT_X	"07002"	信托
ID_BT_BB_G	"07003"	港股股本
ID_BT_PLAN_CA	"99001"	集合资产管理计划
ID_BT_AB	"99002"	资产支持证券
ID_BT_C	"99003"	控制指令

### 8.15. 币种(Currency)

币种类型	类型	说明
CNY	字符串	人民币
HKD	字符串	港币
USD	字符串	美元
AUD	字符串	澳币
CAD	字符串	加币
JPY	字符串	日圆
SGD	字符串	新加坡币
GBP	字符串	英镑
EUR	字符串	欧元
TWD	字符串	新台币
Other	字符串	其他

### 8.16. 产品状态标志(Security\_status)

产品状态标志	说明
1	停牌
2	除权
3	除息
4	风险警示
5	退市整理期
6	上市首日
7	公司再融资
8	恢复上市首日

9	网络投票
10	增发股份上市
11	合约调整
12	暂停上市后协议转让
13	实施双转单调整
14	特定债券转让
15	上市初期
16	退市整理期首日

### 8.17. 委托簿订阅数据类型(SubscribeOrderBookDataType)

数据类型	委托簿订阅数据类型	数值	说明
static const uint64_t	kNone	0x000000000000	订阅全部委托簿数据类别
static const uint64_t	kOrderBook	0x000000000001	订阅委托簿数据
static const uint64_t	kOrderBookSnapshot	0x000000000002	订阅委托簿快照数据

### 8.18. 行情衍生数据订阅类型(SubscribeDerivedDataType)

数据类型	行情衍生数据订阅类型	数值	说明
static const uint64_t	kOrderBook	1	订阅委托簿数据
static const uint64_t	kOrderBookSnapshot	2	订阅委托簿快照数据
static const uint64_t	kIOPVSnapshot	3	订阅 IOPV 快照数据

### 8.19. 华锐自定义交易阶段(TradingPhaseCode)

数据类型	自定义代码名称	数值	说明
static const uint16_t	kStart	1	启动(开市前)
static const uint16_t	kOpenAuction915	2	开盘集合竞价[09:15-09:20)
static const uint16_t	kOpenAuction920	3	开盘集合竞价[09:20-09:25)
static const uint16_t	kOpenAuction925	4	开盘集合竞价[09:25-09:30)
static const uint16_t	kContTrading	5	连续竞价
static const uint16_t	kLunchBreak	6	休市
static const uint16_t	kCloseAuction	7	收盘集合竞价

static const uint16_t	kEndOfTrade	8	闭市
static const uint16_t	kVolatility	9	波动性中断
static const uint16_t	kHalt	10	停牌(含全天停牌和临时停牌)
static const uint16_t	kOther	99	其他

## 七、附录

### 1. 权限数据类型订阅(灰色标识代表不支持此项数据)

		市场 Market						
数据类型 SubSecuDataType	数据结构	深圳 SZSE	上海 SSE	全国股转系统 NEEQ	五大期货 交易所	港交所 HKEx	银行间 CFETS	纳交所 NASDAQ
快照 Snapshot	MDSnapshot 快照数据结构							
	MDBondSnapshot 债券快照数据结构							
	MDIndicatorOfTradingVolumeSnapshot 成交量统计指标快照							
指数快照 IndexSnapshot	MDIndexSnapshot 指数数据结构							
	MDCnIndexSnapshot 国证指数快照							
中证指数快照 CSIIndexSnapshot	MDCSIIndexSnapshot 中证指数行情信息							
港股通快照 HKTSnapshot	MDHKTSnapshot 港股通行快照							
	MDHKTVCM 港股通 VCM 数据结构							
	MDHKTRealtimeLimit 港股通实时额度							
	MDHKTProductStatus 港股通产品状态数据							
	MDHKMarketStatus 港股通市场状态							
港股通实时额度 MDHKTRealtimeLimit	MDHKTRealtimeLimit 港股通实时额度							
港股通市场状态 MDHKMarketStatus	MDHKMarketStatus 港股通市场状态							

期权快照 OptionSnapshot	MDOptionSnapshot 期权快照数据结构							
盘后定价快照 AfterHourFixedPriceSnapshot	MDAfterHourFixedPriceSnapshot 盘后定价行情快照							
盘后定价逐笔成交 AfterHourFixedPriceTickExecution	MDAfterHourFixedPriceTickExecution 盘后定价逐笔成交							
逐笔成交 TickExecution	MDTickExecution 逐笔成交数据结构							
	MDNegotiableTickExecution 协议交易逐笔成交							
	MDRefinancingTickExecution 转融通证券出借逐笔成交							
	MDBondTickExecution 债券逐笔成交数据结构							
	MDBondQuotedTickExecution 债券业务报价及大额逐笔成交数据结构							
逐笔委托 TickOrder	MDTickOrder 逐笔委托数据结构							
	MDNegotiableTickOrder 协议交易逐笔委托							
	MDRefinancingTickOrder 转融通证券出借逐笔委托							
	MDBondTickOrder 债券逐笔委托数据结构							
	MDBondQuotedTickOrder 债券业务报价及大额逐笔委托数据结构							
北交所快照 NEEQSnapshot	MDNEEQSecurityInfo 股转系统证券信息							
	MDNEEQNonPublicTransDeclaredInfo 股转系统非公开转让申报信息							
	MDNEEQHierarchicalInfo 股转系统分层信息							
	MDNEEQSnapshot 股转系统证券行情							
	MDNEEQNegotiableDeclaredInfo 股转系统协议转让申报信息							
	MDNEEQMarketMakerDeclaredInfo 股转系统做市业务申报信息							

	MDNEEQNonPublicTransferDealInfo 股转系统非公开转让成交信息							
期货 FutureSnapshot	MDFutureSnapshot 期货快照数据结构							
委托队列 OrderQueue	MDOOrderQueue 现货委托队列数据结构							
商业港股行情快照  kHKExSnapshot	MDHKExSnapshot 商业港股股票快照数据结构							
	MDHKExOrderSnapshot 商业港股委托挂单数据结构							
	MDHKExOrderBrokerSnapshot 商业港股经纪席位数据结构							
	MDHKExIndexSnapshot 商业港股指数行情快照数据结构							
	MDHKExConnectTurnoverSnapshot 商业港股港股通成交额快照数据结构							
商业港股委托挂单快照 kMDHKExConnectTurnoverSnapshot	MDHKExConnectTurnoverSnapshot 商业港股港股通成交额快照数据结构							
商业港股行情逐笔 kHKExTickExecution	MDHKExTickExecution 商业港股逐笔成交数据结构							
基金通快照数据 kFundExpertSnapshot	MDFundExpertSnapshot 基金通快照数据结构							
CFETSSnapshot  银行间 CFETS 快照	MDSpotSummarySnapshot 现券买卖汇总成交行情数据结构							
	MDSpotXBondSnapshot 现券买卖匿名点击成交行情(X-Bond)数据结构							
	MDPledgeRepoSummarySnapshot 质押式回购汇总成交行情数据结构							
	MDPledgeRepoInquirySnapshot 质押式回购询价成交行情数据结构							
	MDPledgeRepoDepositSnapshot 质押式回购存款类机构间成交行情数据结构							
	MDPledgeRepoXRepoSnapshot 质押式回购双边匿名点击(X-Repo)成交行情数据结构							

	MDCeditLoanSnapshot 同业拆借成交行情数据结构							
	MDCeditLoanDepositSnapshot 同业拆借存款类机构间成交行情数据结构							
	MDSwapFixedFloatSnapshot 利率互换固定-浮动成交行情数据结构							
	MDSwapXSwapSnapshot 利率互换匿名点击(X-Swap)成交行情数据结构							
	MDSwapXSwapBestOfferSnapshot 利率互换匿名点击(X-Swap)最优报价行情数据结构							
	MDBondLoanUnderlyingSecurityRateSnapshot 债券借贷标的债券借贷费率行情数据结构							
	MDBondLoanTradingProductRateSnapshot 债券借贷交易品种费率行情数据结构							
CFETSTickExecution 银行间 CFETS 逐笔	MDSpotTicMDExecution 现券买卖逐笔成交数据结构							
	MDPledgeRepoTicMDExecution 质押式回购逐笔成交数据结构							
	MDCeditLoanTickExecution 同业拆借逐笔成交数据结构							
	MDSwapFixedFloatTickExecution 利率互换固定-浮动逐笔成交数据结构							
	MDBondLoanTickExecution 债券借贷逐笔成交数据结构							
CFETSPprivateMarket 银行间本币私有行情	MDSpotMarketMakersQuoteSnapshot 现券买卖做市报价深度行情数据结构							
	MDSpotDirectiveQuoteSnapshot 现券买卖指示性报价深度行情数据结构							
	MDSpotXBondDepthQuoteSnapshot 现券买卖匿名点击(X-Bond)报价深度行情数据结构							
	MDSpotXBondMatchQuoteSnapshot 现券买卖匿名点击(X-Bond)报价集中匹配行情数据结构							
	MDPledgeRepoXRepoDepthQuoteSnapshot 质押式							



	回购匿名点击(X-Repo)报价深度行情数据结构							
	MDPledgeRepoXRepoBestOfferQuoteSnapshot 质押式回购匿名点击(X-Repo)报价最优行情数据结构							
	MDSwapXSwapDepthQuoteSnapshot 利率互换匿名点击(X-Swap)报价深度行情数据结构							
	MDSwapXSwapTickExecution 利率互换匿名点击(X-Swap)逐笔成交数据结构							
UsaSnapshot 美股快照行情	MDUsaStockSnapshot 美股股票快照数据结构							
	MDUsaOrderSnapshot 美股挂单快照结构							
	MDUsaEodSnapshot 美股 EOD 收盘快照数据结构							
固定收益快照行情 FixedIncomeSnapshot	MDZQQDBJSnapshot 确定报价固收行情数据结构							
	MDZQCJHJSnapshot 成交行情固收行情数据结构							
	MDZQCJMXSnapshot 成交明细固收行情数据结构							

## 2. 品种数据类型订阅(灰色标识代表不支持此项数据)

		市场 Market + 品种 CategoryType						
数据类型 SubSecuDataType	数据结构	深圳 SZSE	上海 SSE	全国股转系统 NEEQ	五大期货交易所	港交所 HKEx	银行间 CFETS	纳交所 NASDAQ
快照 Snapshot	MDSnapshot 快照数据结构	股票 Stock 基金 Fund	股票 Stock 基金 Fund					
	MDIndexSnapshot 指数数据结构	指数 Index	指数 Index					
	MDCnIndexSnapshot 国证指数快照							
	MDIndicatorOfTradingVolumeSnapshot 成交量统计指标快照							
	MDOptionSnapshot 期权快照数据结构	期权 Option	期权 Option					
	MDHKTSnapshot 港股通行情快照	港股通						
	MDHKTVCM 港股通 VCM 数据结构		港股通					

MDHKTRealtimeLimit 港股通实时额度	HKT	HKT					
MDHKTProductStatus 港股通产品状态数据							
MDHKMarketStatus 港股通市场状态							
MDAfterHourFixedPriceSnapshot 盘后定价行情快照	股票 Stock 基金 Fund 债券 Bond	股票 Stock					
MDCSIIIndexSnapshot 中证指数行情信息		指数 Index					
MDFutureSnapshot 期货快照数据结构				期货/期货期权 FutureOption			
MDNEEQSecurityInfo 股转系统证券信息							
MDNEEQNonPublicTransDeclaredInfo 股转系统非公开转让申报信息							
MDNEEQHierarchicalInfo 股转系统分层信息							
MDNEEQSnapshot 股转系统证券行情			股票 Stock				
MDNEEQNegotiableDeclaredInfo 股转系统协议转让申报信息							
MDNEEQMarketMakerDeclaredInfo 股转系统做市业务申报信息							
MDNEEQNonPublicTransferDealInfo 股转系统非公开转让成交信息							
MDBondSnapshot 债券快照数据结构	债券 Bond	债券 Bond					
MDFundExpertSnapshot 基金通快照数据结构	基金 Fund	基金 Fund					
MDHKExSnapshot 商业港股股票快照数据结构					商业港股 HKEx		
MDHKExOrderSnapshot 商业港股委托挂单数据结构					商业港股 HKEx		
MDHKExOrderBrokerSnapshot 商业港股经纪席位数据结构					商业港股 HKEx		
MDHKExIndexSnapshot 商业港股指数行情					商业港股		

快照数据结构					HKEx		
MDHKExConnectTurnoverSnapshot 商业港股 股港股通成交额快照数据结构					商业港股 HKEx		
MDSpotSummarySnapshot 现券买卖汇总成交行情数据结构						银行间本币 CFETSRMB	
MDSpotXBondSnapshot 现券买卖匿名点击成交行情(X-Bond)数据结构						银行间本币 CFETSRMB	
MDPledgeRepoSummarySnapshot 质押式回购汇总成交行情数据结构						银行间本币 CFETSRMB	
MDPledgeRepoInquirySnapshot 质押式回购询价成交行情数据结构						银行间本币 CFETSRMB	
MDPledgeRepoDepositSnapshot 质押式回购存款类机构间成交行情数据结构						银行间本币 CFETSRMB	
MDPledgeRepoXRepoSnapshot 质押式回购双边匿名点击(X-Repo)成交行情数据结构						银行间本币 CFETSRMB	
MDCeditLoanSnapshot 同业拆借成交行情数据结构						银行间本币 CFETSRMB	
MDCeditLoanDepositSnapshot 同业拆借存款类机构间成交行情数据结构						银行间本币 CFETSRMB	
MDSwapFixedFloatSnapshot 利率互换固定-浮动成交行情数据结构						银行间本币 CFETSRMB	
MDSwapXSwapSnapshot 利率互换匿名点击(X-Swap)成交行情数据结构						银行间本币 CFETSRMB	
MDSwapXSwapBestOfferSnapshot 利率互换匿名点击(X-Swap)最优报价行情数据结构						银行间本币 CFETSRMB	
MDBondLoanUnderlyingSecurityRateSnapshot 债券借贷标的债券借贷费率行情数据结构						银行间本币 CFETSRMB	
MDBondLoanTradingProductRateSnapshot 债券借贷交易品种费率行情数据结构						银行间本币 CFETSRMB	
MDSpotMarketMakersQuoteSnapshot 现券买卖做市报价深度行情数据结构						银行间本币 CFETSRMB	

	MDSpotDirectiveQuoteSnapshot 现券买卖指示性报价深度行情数据结构						银行间本币 CFETSRMB	
	MDSpotXBondDepthQuoteSnapshot 现券买卖匿名点击(X-Bond)报价深度行情数据结构						银行间本币 CFETSRMB	
	MDSpotXBondMatchQuoteSnapshot 现券买卖匿名点击(X-Bond)报价集中匹配行情数据结构						银行间本币 CFETSRMB	
	MDPledgeRepoXRepoDepthQuoteSnapshot 质押式回购匿名点击(X-Repo)报价深度行情数据结构						银行间本币 CFETSRMB	
	MDPledgeRepoXRepoBestOfferQuoteSnapshot 质押式回购匿名点击(X-Repo)报价最优行情数据结构						银行间本币 CFETSRMB	
	MDSwapXSwapDepthQuoteSnapshot 利率互换匿名点击(X-Swap)报价深度行情数据结构						银行间本币 CFETSRMB	
	MDUsaStockSnapshot 美股股票快照数据结构							股票 Stock
	MDUsaOrderSnapshot 美股挂单快照结构							股票 Stock
	MDUsaEodSnapshot 美股 EOD 收盘快照数据结构							股票 Stock
	MDZQQDBJSnapshot 确定报价固收行情数据结构		其他 Others					
	MDZQCJHQSnapshot 成交行情固收行情数据结构		其他 Others					
	MDZQCJMXSnapshot 成交明细固收行情数据结构		其他 Others					
逐笔成交 TickExecution	MDTickExecution 逐笔成交数据结构	股票 Stock 基金 Fund	股票 Stock 基金 Fund					
	MDNegotiableTickExecution 协议交易逐笔成交	股票 Stock 基金 Fund 债券 Bond						
	MDRefinancingTickExecution 转融通证券	其他 others						

	出借逐笔成交							
	MDAfterHourFixedPriceTickExecution 盘后定价逐笔成交		股票 Stock					
	MDBondTickExecution 债券逐笔成交数据结构	债券 Bond	债券 Bond					
	MDBondQuotedTickExecution 债券业务报价及大额逐笔成交数据结构	债券 Bond						
	MDHKExTickExecution 商业港股逐笔成交数据结构					商业港股 HKEx		
	MDSpotTicMDEExecution 现券买卖逐笔成交数据结构						银行间本币 CFETS RMB	
	MDPledgeRepoTicMDEExecution 质押式回购逐笔成交数据结构						银行间本币 CFETS RMB	
	MDCeditLoanTickExecution 同业拆借逐笔成交数据结构						银行间本币 CFETS RMB	
	MDSwapFixedFloatTickExecution 利率互换固定-浮动逐笔成交数据结构						银行间本币 CFETS RMB	
	MDBondLoanTickExecution 债券借贷逐笔成交数据结构						银行间本币 CFETS RMB	
	MDSwapXSwapTickExecution 利率互换匿名点击(X-Swap)逐笔成交数据结构						银行间本币 CFETS RMB	
逐笔委托 TickOrder	MDTickOrder 逐笔委托数据结构	股票 Stock 基金 Fund	股票 Stock 基金 Fund					
	MDNegotiableTickOrder 协议交易逐笔委托	股票 Stock 基金 Fund 债券 Bond						
	MDRefinancingTickOrder 转融通证券出借逐笔委托	其他 others						
	MDBondTickOrder 债券逐笔委托数据结构	债券 Bond	债券 Bond					
	MDBondQuotedTickOrder 债券业务报价及大额逐笔委托数据结构	债券 Bond						
委托队列 OrderQueue	MDOOrderQueue 现货委托队列数据结构	股票 Stock 基金 Fund	股票 Stock 基金 Fund					

		债券 Bond	债券 Bond					
--	--	---------	---------	--	--	--	--	--