

上海证券交易所技术文档



IS120 上海证券交易所行情网关
BINARY 数据接口规范
0.63 版



上海证券交易所

二〇二六年九月

修订记录

修订日期	版本号	修订内容
<u>2026-09-18</u>	<u>0.63</u>	<u>1、市场状态消息 (MsgType=M101)，增加在综合业务市场无支持开盘集合竞价的产品、无发行期内国债预发行产品的情况下，字段无意义说明。</u> <u>2、对于盘后固定价格交易业务，明确行情快照字段取值。</u>
2026-07-03	0.62	1、下线竞价撮合系统行情（股票、基金、指数及债券分销）中 IOPV 行情，相关字段标记为“无意义”。 2、在 2.1.3 建立行情会话章节中补充登录时间要求，在 2.4.2 重传消息章节中调整数据缓存要求相关描述。
2026-05-11	0.61	1、原 2.1 会话机制交互图拆分为业务消息和重传消息。 2、原应用信息章节抽取逐笔行情和快照行情通用规则组成 2.5.1 应用信息通用约定章节。 3、添加了固收迁移互联网平台后新增行情信息（含债券非匹配成交快照行情及各类逐笔行情等）。 4、移除原固收文件行情接口包括：se015cjhqYYMMDD001.txt（固收成交行情文件）、se015cjmxxYYMMDD001.txt（固收成交明细文件）、se015qdbjYYMMDD001.txt（固收确定报价文件）和 se015zqxxYYMMDD001.txt（固收证券信息文件）。 5、新增接收 products_yyyymmdd.xml（证券产品基础信息文件）、bondmbrs_yyyymmdd.xml（债券交易参与人信息文件）和

		bondtrdrs_yyyymmdd.xml (债券交易员信息文件)。
2026-03-20	0.60	配合 txt 版 ETF 定义文件下线，删除相关描述。
2025-10-17	0.59	支持 ETF 公告文件 xml 版，调整 2.4.2.2 集中竞价类行情快照扩展字段章节 IOPV 行情与 ETF 公告文件关联关系描述。
2025-08-01	0.58	在市场状态消息 (MsgType=M101)、行情快照消息 (MsgType=M102) 中增加独立 IOPV 行情描述。
2023-08-14	0.57	1、明确 PreCloseIOPV (基金 T-1 日收盘时刻 IOPV)、IOPV (基金 IOPV) 字段适用范围以及和 ETF 公告文件的关联关系。 2、增加基金通行情接收相关描述。
2022-04-13	0.56	增加期权基础信息第二版文件接收相关描述。
2022-03-21	0.55	增加 B 转 H 行情文件接收相关描述。
2022-01-17	0.54	调整市场状态消息中股票、基金、指数及债券分销 (SecurityType = 01) 和债券市场行情 (SecurityType = 12) 描述
2022-01-11	0.53	1、调整行情接口中债券指数、债券和回购的数量单位； 2、调整 STEP 行情接口中“成交金额”字段长度，与 BINARY 接口保持一致； 3、移除市场行情状态 (SecurityType = 01) 中债券质押回购行情结束标志信息
2021-08-06	0.52	调整行情接口中债券相关的行情数据说明
2021-03-24	0.51	调整股票 (含指数) 全市场行情状态中债券质押回购行情结束标志的相关描述

2021-03-05	0.50	调整债券产品为单独证券类型
2021-01-19	0.42	行情快照消息增加公募 REITs 相关描述
2020-05-20	0.41	变更市场状态消息、行情快照消息的消息头中 SendingTime 为交易所时间
2019-12-05	0.40	原内容移入第二章,增加章节描述通过行情网关接收的文件及外部转发数据
2019-01-25	0.32	1、增加盘后固定价格交易的行情接口说明,调整国债预发行接口字段取值 2、调整盘后固定价格行情的产品状态取值
2019-01-10	0.31	增加债券回购延长对市场状态消息字段的说明
2018-07-11	0.30	1、根据反馈意见调整部分说明、调整价格精度、增加成交笔数及期权虚拟匹配数量 2、TradingPhaseCode 闭市集合竞价相关调整
2018-03-25	0.20	据原有文件接口进行字段及内容调整
2018-03-09	0.10	文档创建

目 录

1	引言	1
1.1	名词释义	1
2	BINARY 实时行情	2
2.1	会话机制	2
2.1.1	消息序号	3
2.1.2	会话安全	3
2.1.3	建立行情会话	3
2.1.4	行情数据发布	4
2.1.5	关闭行情会话	4
2.1.6	心跳	4
2.1.7	行情网关主动关闭行情会话的情况	5
2.2	协议介绍	5
2.2.1	字段说明	5
2.2.2	BINARY 消息头	6
2.2.3	BINARY 消息尾	6
2.3	会话消息	8
2.3.1	登录消息 (MsgType=S001)	8
2.3.2	注销消息 (MsgType=S002)	8
2.3.3	心跳消息 (MsgType=S003)	9
2.4	公共消息	10

2.4.1	频道心跳 (MsgType=P001)	10
2.4.2	重传消息 (MsgType=P002)	10
2.5	应用消息	12
2.5.1	应用消息通用约定	12
2.5.2	市场状态消息 (MsgType=M101)	14
2.5.3	行情快照消息 (MsgType=M102)	17
2.5.4	逐笔行情消息 (MsgType=M201)	25
3	文件接收	31
4	转发行情	33
	后 记	34

1 引言

上海证券交易所行情网关数据接口规范包括 BINARY 与 STEP 两卷,本卷主要介绍 BINARY 数据接口规范(第二章),并对行情网关可接收文件(第三章)和转发行情(第四章)进行一定的说明,请市场参与者结合《上海证券交易所行情网关技术指引及接口开发指南》一并使用。

文档采用的术语及消息内容与 STEP 数据接口规范具有对应关系,可以互为参考。

1.1 名词释义

名词	含义
MDGW	Market Data GateWay 行情网关
VSS	Vendor Supplied System 用户行情系统
STEP	Securities Trading Exchange Protocol 证券交易数据交换协议

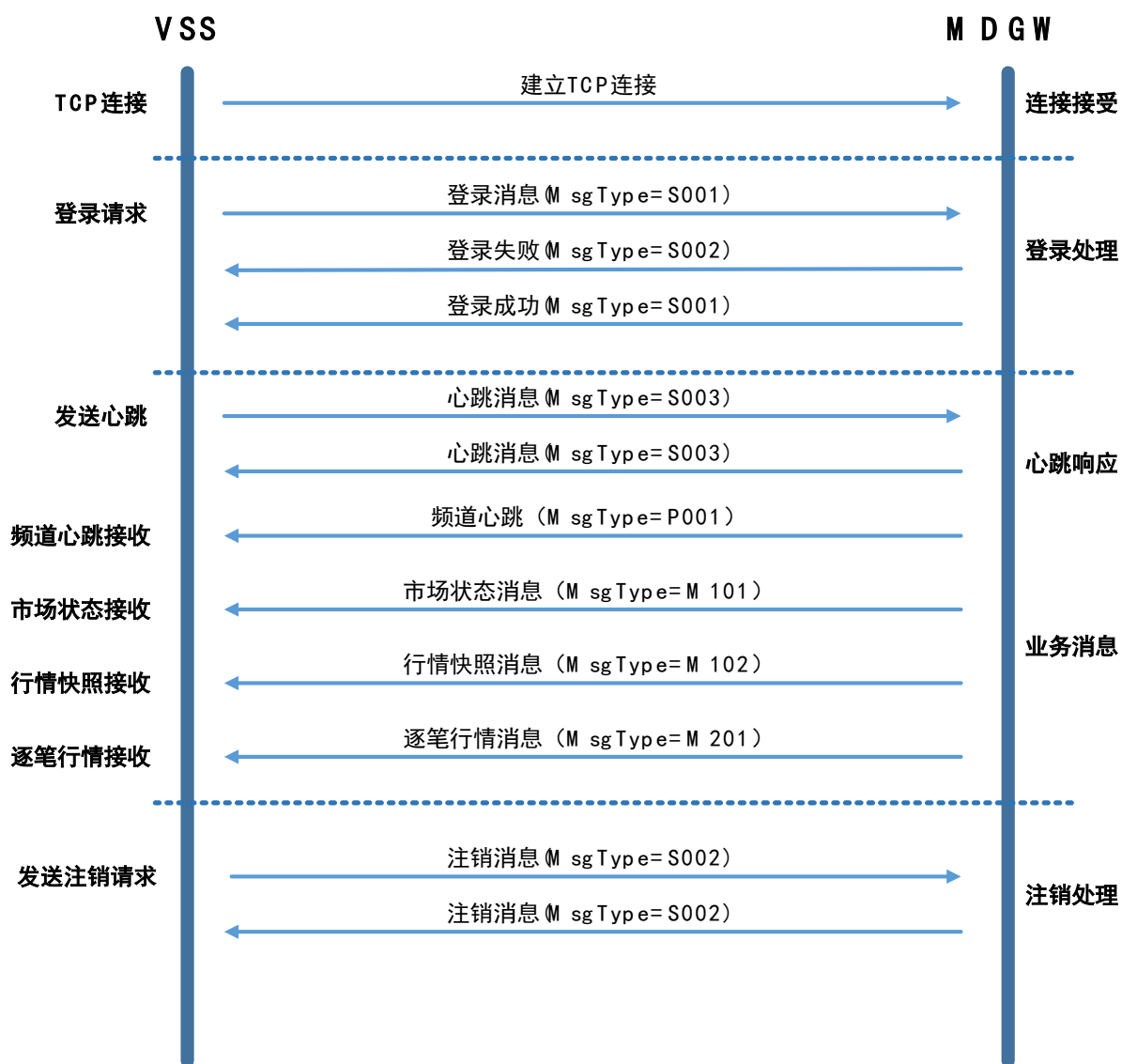
2 BINARY 实时行情

本章描述市场参与者与信息服务商等用户行情系统，通过 BINARY 协议接入上海证券交易所行情网关 MDGW 进行行情数据传输的机制和相关数据交换格式。

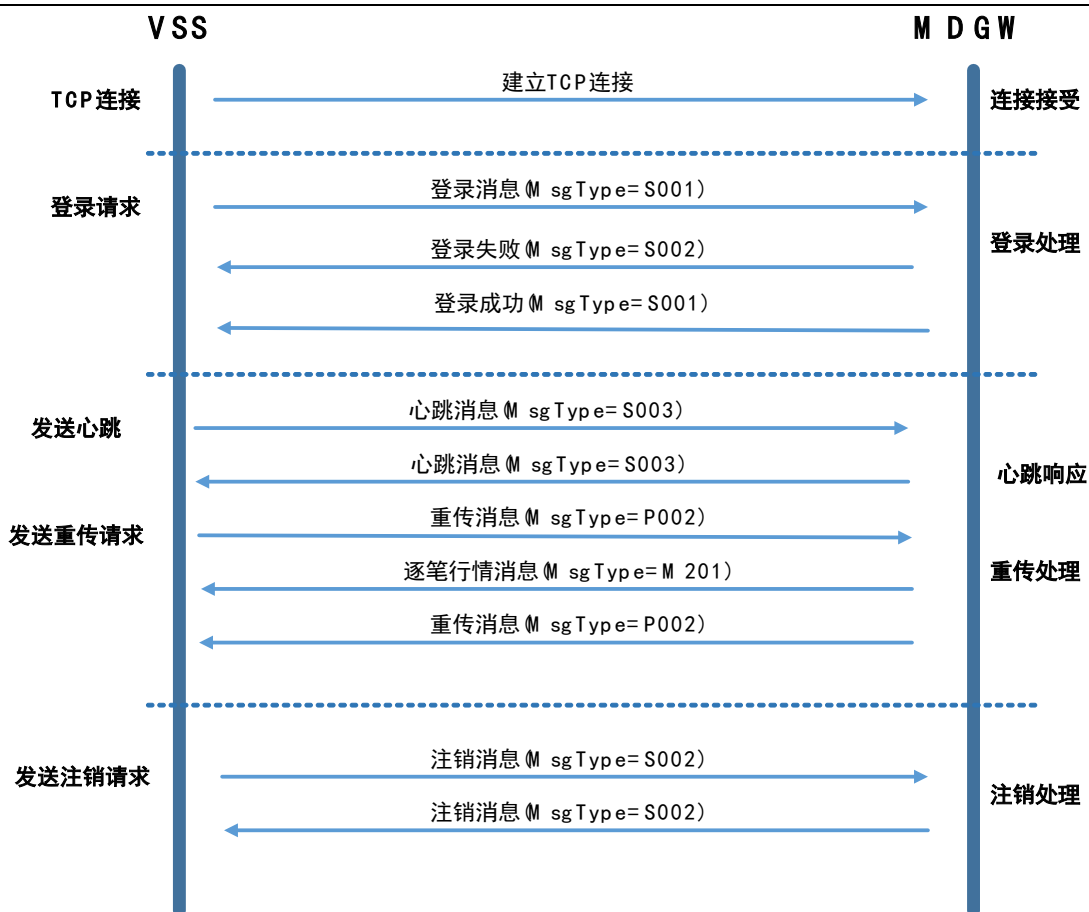
2.1 会话机制

用户行情系统（VSS）通过 TCP 方式接收行情网关（MDGW）发布的流式行情。

交互示意图如下：



BINARY 协议交互图（业务消息）



BINARY 协议交互图（重传消息）

2.1.1 消息序号

会话双方收发的每条消息都被分配有一个消息序号 `MsgSeqNum` 来唯一标识。参与通信的每一端都需要维护一对序号（`NxtIn`, `NxtOut`），`NxtIn` 表示下一个期望的入向消息序号，`NxtOut` 表示下一个出向消息序号。消息序号一般在每次会话过程中从 1 开始，在整个会话过程中连续递增，直到该会话过程全部结束。

2.1.2 会话安全

MDGW 与 VSS 之间传输的数据是非加密的，数据传输安全由接入用户网络保证。

2.1.3 建立行情会话

每个行情会话都是完全独立的，同一 VSS 在一个交易日内的多次登录被视为不同的行情会

话。

建立行情会话包含两个步骤：建立 TCP 连接、登录。具体步骤如下：

1、TCP 连接

VSS 与 MDGW 建立 TCP 通讯连接，其中 MDGW 是服务端，VSS 负责发起连接。

2、登录

VSS 发送的第一个消息必须是登录消息，如果 VSS 登录成功，则 MDGW 发送一个登录消息作为应答。如果 VSS 登录失败，MDGW 则在发送一个含失败说明的注销消息后由 VSS 主动关闭连接。VSS 必须在收到 MDGW 的登录消息之后才允许发送其他消息。

涉及接收逐笔行情的各市场参与者需在 8:30 前完成行情网关登录，避免影响行情接收和正常交易。

2.1.4 行情数据发布

在完成建立行情会话之后，MDGW 将向 VSS 发送行情数据消息。行情数据消息格式将在“应用消息”中详细叙述。MDGW 目前发布的行情快照消息不支持重传，逐笔行情消息支持重传。

2.1.5 关闭行情会话

行情会话的正常关闭是通过连接双方互相发送注销消息完成的。MDGW 和 VSS 均可以主动发送注销消息，接收方需要回传注销消息作为应答。如果超过预定时间（一般为 5 秒）没有收到对方回传的注销消息，任何一方均有权主动关闭连接。

2.1.6 心跳

连接双方在数据发送的空闲期间应主动发送心跳消息，通过心跳消息可以监控行情会话的状态。心跳最小间隔由登录消息中的 HeartBtInt 域确定。

连接双方在发送任何消息后,应立即重新设置心跳间隔计时器。如果 VSS 超过 2 个 HeartBtIntl 指定周期没有收到 MDGW 发送的任何消息,则行情会话被视为可能存在异常,VSS 需要重新建立行情会话。

2.1.7 行情网关主动关闭行情会话的情况

在以下几种场景(包括但不限于)下,MDGW 会主动关闭行情会话:

- 1、VSS 与 MDGW 建立 TCP 连接之后,超过预定时间(一般为 5 秒)未发送消息完成登录;
- 2、VSS 在登录失败之后,长时间不关闭 TCP 连接;
- 3、VSS 在注销之后,长时间不关闭 TCP 连接;
- 4、MDGW 与交易所行情主机连接异常;

5、VSS 如未及时处理 MDGW 发送的数据,导致 MDGW 内积压的待发送消息超过特定阈值。

2.2 协议介绍

每条 BINARY 消息由消息头、消息体和消息尾组成,消息最大长度为 8K 字节。

2.2.1 字段说明

所有数值型字段采用高字节序(BIG_ENDIAN)编码。字段类型说明如下:

类型	说明
char[x]	代表该字段为字符串,x 代表该字符串的最大字节数,x 为大于零的数字,例如 char[5]代表最大长度为 5 字节的字符串;当最大长度大于实际长度时,右补空格。字符串使用 GBK 编码。
int、uint	代表该字段为整型数值,如 uint32 表示 32 位无符号整数,int64 表示 64 位有符

	号整数
Nx、Nx(y)	与 int、uint 一并使用，用于给出该整型数值实际表示的业务字段的长度（精度）： Nx 代表最大长度为 x 位数字的整数；Nx(y)代表最大长度为 x 位数字，其中最末 y 位数字为小数部分。

2.2.2 BINARY 消息头

每一个会话或应用消息都有一个消息头，该消息头指明消息类型、消息体长度、消息序号和发送时间等信息。

消息头格式如下：

域名	说明	类型
MsgType	消息类型	char[4]
SendingTime	发送时间，格式：YYYYMMDDHHmmSSsss	uint64
MsgSeqNum	消息序号	uint64
BodyLength	消息体长度	uint32

说明：

1. 当消息为市场状态消息或行情快照消息或逐笔行情消息时，SendingTime 为交易所时间。

2.2.3 BINARY 消息尾

每一个会话或应用消息都有一个消息尾，并以此终止。

消息尾包含一个 CheckSum 字段，其值是计算从消息头开始（包括消息头）到消息体结束的校验和。计算校验和的代码段可参考附录一“计算校验和”。

消息尾格式如下：

域名	说明	类型
----	----	----

Checksum	校验和	uint32
----------	-----	--------

2.3 会话消息

会话消息格式将在以下各节中予以介绍，并定义会话消息格式。

连接双方均可生成会话消息。

2.3.1 登录消息 (MsgType=S001)

登录消息应是在行情会话开始时 VSS 发送的第一个消息。MDGW 只作为登录的接受方，不会作为登录的发起方。

HeartBtInt 域用来指定心跳消息的发送时间间隔，必须设置为大于 0 的整数。VSS 需要在登录消息中填入预期的心跳时间间隔，MDGW 在回传登录消息时返回的 HeartBtInt 域作为协商后的心跳时间间隔。

登录请求消息格式如下：

域名	说明	类型
标准消息头	<i>MsgType = S001</i>	
SenderCompID	发送方代码	char[32]
TargetCompID	接收方代码	char[32]
HeartBtInt	心跳间隔，单位为秒	uint16
ApplVerID	协议版本，格式为 mm.nn，其中 mm 为主版本号，nn 为子版本号，如 0.30，12.22	char[8]
标准消息尾		

2.3.2 注销消息 (MsgType=S002)

注销消息是发起或确认行情会话终止的消息。未经注销消息交换而断开连接，一律视为非正

常的断开。

连接双方在发送注销消息之后不应发送任何消息。

注销消息格式如下：

域名	说明	类型
标准消息头	<i>MsgType = S002</i>	
SessionStatus	Logout 时的会话状态 0 正常注销 【1-999】 一般情况注销，重连可以恢复 【1000-9999】 严重情况注销，建议切换服务器	uint32
Text	文本	char[256]
标准消息尾		

2.3.3 心跳消息 (MsgType=S003)

心跳消息用于监控通信连接的状况。

当连接的任何一方在心跳时间间隔（由 HeartBtInt 域指定）时间内没有接收或发送任何数据的时候，需要产生一个心跳消息并发送出去。如果接收方在 2 倍心跳时间间隔内都没有收到任何消息的时候，那么可认为行情会话出现异常，可以立即关闭 TCP 连接。

心跳消息格式如下：

域名	说明	类型
标准消息头	<i>MsgType = S003, BodyLength = 0</i>	
标准消息尾		

2.4 公共消息

2.4.1 频道心跳 (MsgType=P001)

逐笔行情支持在数据发送的空闲期间每 3 秒发送一次心跳，如数据一直处于发送忙状态，则可能出现频道心跳超过 3s。

域名	说明	类型
标准消息头	<i>MsgType=P001</i>	
SecurityType	证券类型 13 = 债券（非匹配成交）	uint8
ChannelNO	频道号 100 = 非匹配成交逐笔行情	uint16, N4
ApplLastSeqNum	该频道已发布的最大消息记录号	uint64
EndOfChannel	频道结束标志 Y = 行情更新已结束 N = 行情更新未结束	char[1]
标准消息尾		

2.4.2 重传消息 (MsgType=P002)

重传消息用于请求可重传频道下的行情数据。

针对可重传的应用类消息(各类逐笔行情消息)，若应用类消息缺失，用户可向上发送重传消息，代表重传请求。上游以“请求-应答”的方式处理重传请求，根据重传消息中指定的起始、

结束序号返回需要重传的数据，并在重传完成后返回一个重传消息，告知重传完成；若重传失败，则返回一个重传消息，告知重传失败。上游系统仅支持一个缺口消息正在重传，当上游收到多个重传请求时，其他请求消息会返回重传失败。

对于逐笔行情数据可通过频道代码和消息记录号判断是否有消息丢失，当收到的消息记录号 $\leq$ 本频道已经收到的最大消息记录号时，说明已经收到过该消息，此时应忽略该消息。当收到的消息记录号 $>$ 已经收到的最大消息记录号+1（如已收的最大消息记录号=100，新的消息记录号=102）说明发生了消息丢失，此时应通过发送重传请求恢复丢失的数据。

域名	说明	类型
标准消息头	<i>MsgType=P002</i>	
ChannelNO	频道号 100 = 非匹配成交逐笔行情	uint16, N4
ResendType	重传类别 1 = 逐笔行情	uint8
ApplBegSeqNum	重传起始消息记录号 取值大于 0	uint64
ApplEndSeqNum	重传结束消息记录号 当取值为 0 时，交易所会将此字段设置为该频道已发布的最大消息记录号；应答时取值为实际传输的最大消息记录号。	uint64
ResendStatus	仅在重传应答消息中有效，取值如下： 1 = 全部完成 2 = 部分完成（有部分请求的数据未返回）	uint8

	3 = 无权限 4 = 数据不可用 5 = 重传请求正在处理中 99 = 其他	
RejectText	仅在重传应答消息中有效。 填写重传拒绝的原因（如有）。	char[32]
标准消息尾		

注：

1、VSS 应主动缓存逐笔消息，避免频繁发送重传请求，禁止定时或不必要的反复进行重传数据行为。当检测到消息缺失时，建议继续接收、展示及处理后续行情数据，避免因局部消息缺口影响整体数据服务，并合理控制重传频次。

2、建议每次请求回补的数据不大于 200 条。

3、当收到 ResendStatus=2 的重传应答时，建议对于未传输数据重新发起重传请求。

4、对于无效字段，字符类型字段填充格，数值类型字段填 0。

2.5 应用消息

2.5.1 应用消息通用约定

1. 字段无意义或无该字段行情数据时，字符填充格，数值填 0；数值字段取值超过约定格式最大值时，取最大值，如 N10 取 9999999999；
2. 行情类别(MDStreamID)及其扩展字段如下：

SecurityType	MDStreamID	说明
1	MD001	指数类产品
	MD002	股票（A、B 股）类产品
	MD003	债券分销类产品
	MD004	基金类产品（含公募 REITs）

2	MD301	期权类产品
3	MD101	国债预发行产品
	MD102	盘后固定价格
12	MD201	债券匹配成交快照行情 ¹
13	MD210	债券非匹配成交快照行情 ²
	MD211	确定报价逐笔委托行情
	MD212	待定报价逐笔委托行情
	MD213	现券意向申报逐笔委托行情
	MD214	协议回购意向申报逐笔委托行情
	MD215	三方回购意向申报逐笔委托行情
	MD216	竞买逐笔委托行情
	MD217	债券非匹配成交逐笔成交行情
14	MDE01	IOPV

3. 对数量单位说明如下：

- a) 产品价格、金额单位，除 B 股为美元外，其他为人民币元。特别地，对于协议回购或三方回购，产品价格字段表示回购利率，单位为%。对于 SecurityType 为 13 的各类行情，其价格表示净价还是全价则以证券产品基础信息文件（products_yyyyymmdd.xml）中该产品的计价方式为准。
- b) 指数的成交数量(TradeVolume)为参与计算相应指数的交易数量，股票指数交易数量单位是 100 股，基金指数的交易数量单位是 100 份，债券指数的交易数量单位是千元面额；指数成交金额(TotalValueTraded)为参与计算相应指数的成交金额；
- c) 各类产品数量与成交数量单位，股票为股，基金、公募 REITs 为份，债券与回购为千元面额，期权合约的数量单位为张；
- d) 对于债券分销，成交金额为每笔成交的“价格*数量*10”的总和；对于债券现券，成交金额为每笔成交的“价格*数量*10”的总和；对于通用质押式回购，成交金额为 100*成交数量*10。

¹ 包含债券现券匹配成交及质押式回购类产品。

² 包含债券现券非匹配成交、协议回购、三方回购产品。

2.5.2 市场状态消息 (MsgType=M101)

市场状态消息用于交易所发布市场状态和产品数量等信息,会周期性发布以及在市场状态变化时发布。

市场状态消息的格式如下:

域名	说明	类型
标准消息头	<i>MsgType=M101</i>	
SecurityType	证券类型 1 = 股票、基金、指数及债券分销 2 = 衍生品 3 = 综合业务 12 = 债券 (匹配成交) 13 = 债券 (非匹配成交) 14 = 外部源行情	uint8
TradSesMode	交易盘交易模式 1 = 系统测试 2 = 模拟交易 3 = 产品 (正常交易)	uint8
TradingSessionID	市场行情状态: 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。	char[8]
TotNoRelatedSym	最大产品数目 (包括指数)	uint32
标准消息尾		

说明：

1. 衍生品指期权业务；

2. 综合业务指国债预发行、盘后固定价格交易业务；

3. 市场行情状态(TradingSessionID)为 8 位字符串，左起每位表示特定的含义，无定义则填充空格。该字段具体含义在不同证券类型时说明如下：

SecurityType = 1 ³	
第 1 位	‘S’表示全市场启动期间（开市前） ‘T’表示全市场处于交易期间（含中间休市） ‘E’表示全市场处于闭市期间
第 2 位	‘1’表示开盘集合竞价结束标志，未结束取‘0’
第 3 位	‘1’表示全市场行情结束标志，未结束取‘0’
第 4 位	‘1’表示上海市场行情结束标志，未结束取‘0’
SecurityType = 2	
第 1 位	‘S’表示期权市场启动期间（开市前） ‘T’表示期权市场处于交易期间（含中间休市） ‘E’表示期权市场处于闭市期间
第 2 位	‘1’表示期权市场开盘集合竞价结束标志，未结束取‘0’
第 3 位	‘1’表示期权市场行情闭市标志，未闭市取‘0’
SecurityType = 3	
第 1 位	‘S’表示综合业务市场启动期间（开市前） ‘T’表示综合业务市场处于交易期间（含中间休市） ‘E’表示综合业务市场处于闭市期间
第 2 位	‘1’表示综合业务市场开盘集合竞价结束标志，未结束取‘0’。 <u>当综合业务市场无支持开盘集合竞价的产品时，无意义</u>
第 3 位	‘1’表示综合业务市场收盘集合竞价结束标志，未结束取‘0’， <u>暂不启用</u>

³ SecurityType = 1 时，上海市场包括股票、基金、债券分销。全市场包括上海市场及指数。

第 4 位	‘1’表示国债预发行市场行情闭市标志，未闭市取‘0’。 <u>当无发行期内国债预发行产品时，无意义</u>
第 5 位	‘1’表示盘后固定价格交易行情闭市标志，未闭市取‘0’
SecurityType = 12 ⁴	
第 1 位	‘S’表示债券市场启动期间（开市前） ‘T’表示债券市场处于交易期间（含中间休市） ‘E’表示债券市场处于闭市期间
第 2 位	‘1’表示债券市场开盘集合竞价结束标志，未结束取‘0’
第 3 位	‘1’表示债券市场行情结束标志，未结束取‘0’
第 4 位	‘1’表示债券现券（可转债及新标准券）行情结束标志，未结束取‘0’
第 5 位	‘1’表示债券质押式回购、债券现券（除可转债及新标准券）行情结束标志，未结束取‘0’
SecurityType = 13 ⁵	
第 1 位	‘S’表示债券市场启动期间（开市前） ‘T’表示债券市场处于交易期间（含中间休市） ‘E’表示债券市场处于闭市期间
第 2 位	无意义
第 3 位	‘1’表示债券市场行情结束标志，未结束取‘0’
第 4 位	‘1’表示债券现券（可转债）行情结束标志，未结束取‘0’
第 5 位	‘1’表示协议回购、三方回购、债券现券（除可转债）等行情结束标志，未结束取‘0’。
SecurityType = 14	
第 1 位-第 8 位	全空格

⁴ SecurityType = 12 时，债券市场（匹配成交）包含债券现券及质押式回购。

⁵ SecurityType = 13 时，债券市场（非匹配成交）包含债券现券、协议回购、三方回购等。

2.5.3 行情快照消息 (MsgType=M102)

行情快照消息用于发布证券产品行情,交易所会周期性发布全量快照消息以及在行情变化时发送增量快照消息。多条快照消息之间无数据依赖性, VSS 无需区分全量/增量消息,可直接替换当前数据。

消息格式如下:

域名	说明	类型
标准消息头	<i>MsgType=M102</i>	
SecurityType	证券类型 1 = 股票、基金、指数及债券分销 2 = 衍生品 3 = 综合业务 12 = 债券 (匹配成交) 13 = 债券 (非匹配成交) 14 = 外部源行情	uint8
TradSesMode	交易盘交易模式 1 = 系统测试 2 = 模拟交易 3 = 产品 (正常交易)	uint8
TradeDate	交易日期 YYYYMMDD	uint32, N8
LastUpdateTime	最新更新时间 HHMMSSsss	uint32, N9
MDStreamID	行情类别	char[5]

SecurityID	产品代码	char[8]
Symbol	产品简称	char[8]
PreClosePx	昨收盘	uint64, N13(5)
TotalVolumeTraded	成交数量	uint64, N16
NumTrades	成交笔数	uint64, N16
TotalValueTraded	成交金额	uint64, N16(2)
TradingPhaseCode	实时阶段及标志	char[8]
ExtendFields	扩展字段	
标准消息尾		

说明：

1. 当 SecurityType = 14 且 MDStreamID = MDE01 时，PreClosePx 昨收盘、TotalVolumeTraded 成交数量、NumTrades 成交笔数、TotalValueTraded 成交金额字段取值为 0，TradingPhaseCode 实时阶段及标志取值为空。
2. 实时阶段及标志(TradingPhaseCode)为 8 位字符串，左起每位表示特定的含义，无定义则填充格。该字段具体含义在不同行情类别时说明如下：

SecurityType = 1, MDStreamID= MD001
全为空格（预留）

SecurityType = 1, MDStreamID= MD002, MD003, MD004	
第 1 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 2 位	‘0’表示此产品不可正常交易 ‘1’表示此产品可正常交易。 在产品进入开盘集合竞价、连续交易、收盘集合竞价、熔断(盘中集合竞价)状态时值为‘1’，在产品进入停牌、熔断（暂停交易至闭市）状态时值为‘0’，且闭市后保持该产品闭市前的是否可正常交易状态。
第 3 位	‘0’表示未上市 ‘1’表示已上市
第 4 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。 仅在交易时段有效，在非交易时段无效。

SecurityType = 2, MDStreamID= MD301	
第 1 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）
第 2 位	‘0’表示未连续停牌 ‘1’表示连续停牌 （预留，暂填空格）
第 3 位	‘0’表示不限制开仓 ‘1’表示限制备兑开仓 ‘2’表示限制卖出开仓 ‘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
第 4 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。 仅在交易时段有效，在非交易时段无效

SecurityType = 3, MDStreamID= MD101	
第 1 位	‘S’表示启动（开市前）时段 ‘C’表示 <u>开盘</u> 集合竞价时段 ‘T’表示连续交易时段 ‘E’表示闭市时段 ‘P’表示停牌
SecurityType = 3, MDStreamID= MD102	
第 1 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
SecurityType = 12, MDStreamID= MD201	
第 1 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
第 2 位	‘0’表示此产品不可正常交易 ‘1’表示此产品可正常交易 在产品进入开盘集合竞价、连续交易状态时值为‘1’，在产品进入停牌状态时值为‘0’，且闭市后保持该产品闭市前的是否可正常交易状态。
第 3 位	‘0’表示未上市 ‘1’表示已上市
第 4 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 仅在交易时段有效，在非交易时段无效。

SecurityType = 13, MDStreamID= MD210	
第 1 位	无意义
第 2 位	‘0’表示当前处于停牌状态 ‘1’表示当前未处于停牌状态
SecurityType = 14, MDStreamID= MDE01	
第 1 位-第 8 位	全空格

3. 盘后固定价格交易 (SecurityType = 3, MDStreamID= MD102), PreClosePx 无意义。

2.5.3.1 指数行情快照扩展字段

域名		说明	类型
NoMDEntries		行情条目个数	uint16, N5
→	MDEntryType	行情条目类别 3 = 最新指数 4 = 今开盘指数 5 = 今收盘指数 7 = 最高指数 8 = 最低指数	char[2]
→	MDEntryPx	指数点位	uint64, N13(5)

说明：

1. 适用于 MDStreamID=MD001 时。

2.5.3.2 行情快照扩展字段

域名		说明	类型
NoMDEntries		行情条目个数	uint16, N5
→	MDEntryType	行情条目类别 0=买入 1=卖出 2=最新成交价 4=今开盘价 5=今收盘价 6=今结算价 7=当日最高成交价 8=当日最低成交价 9=当日成交均价 v=IOPV w=前一日 IOPV x=动态参考价格及虚拟匹配数量 z1=昨日结算价 z2=总持仓量 z3=昨日成交均价	char[2]
→	MDEntryPx	价格	uint64, N13(5)
→	MDEntrySize	数量	uint64, N12
→	MDEntryPositionNo	买卖盘序号, 对应五档买入卖出价量的档位信息	uint8, N2

说明:

1. 适用于 MDStreamID=MD002、MD003、MD004、MD101、MD102、MD301、MD201、MDE01、MD210

时；

2. 行情条目类别为 0 或 1 时，表示该条目为买卖盘档位，此时 MDEntryPx、MDEntrySize、MDEntryPositionNo 分别表示买卖价格、数量和档位序号，MDEntryPositionNo 从 0 开始计数；
3. 行情条目类别为 w 时，MDEntryPx 表示基金 T-1 日收盘时刻 IOPV，数据内容无意义；行情条目类别为 v 时，MDEntryPx 表示基金 IOPV⁶，此条目仅当 MDStreamID= MDE01 时有意义，该字段取 0 时无意义；
4. 行情条目类别为 x 时，MDEntryPx 表示动态参考价格、MDEntrySize 表示虚拟匹配数量，仅当 MDStreamID=MD301 时有意义；
5. 行情条目类别为 z2 时，MDEntrySize 表示期权合约总持仓量，单位是(张)，仅当 MDStreamID=MD301 时有意义；
6. 行情条目类别为 9 时，MDEntryPx 表示债券非匹配成交的当日成交均价；行情条目类别为 z3 时，MDEntryPx 表示债券非匹配成交的昨日成交均价；
7. 其他行情条目类别，将仅通过 MDEntryPx 表示相应值，其他字段无意义；
8. 在集合竞价时段内，当前买入价和当前卖出价中同时为虚拟开盘参考价格，即根据集合竞价算法计算得出的虚拟撮合价格。同时申买量一和申卖量一为行情发布时刻的虚拟匹配量。申买量二为行情发布时刻的买方虚拟未匹配量。申卖量二为行情发布时刻的卖方虚拟未匹配量；
9. 对期权产品（MDStreamID=MD301），今日结算价通过期权收盘价格文件（clpr03MMDD.txt）揭示。
10. 对盘后固定价格行情（MDStreamID=MD102），仅今日收盘价、买一数量、卖一数量有意义。其中，买一数量表示当前未成交的买入申报总股数，卖一数量表示当前未成交的卖出申报总股数；
 - a) 启动（开市前）时段/集中撮合时段（TradingPhaseCode 第 1 位为 ‘I’ 或 ‘A’），循环组包含 MDEntryType 为 0=买入、1=卖出两个条目，价格 MDEntryPx、数量 MDEntrySize 均取值为 0；
 - b) 连续交易/闭市时段（TradingPhaseCode 第 1 位为 ‘H’ 或 ‘D’），循环组包含 1-3 条记录，包含 MDEntryType 为 0=买入（买方当前未成交的申报总股数为 0 时，无该条目）、MDEntryType 为 1=卖出（卖方当前未成交的申报总股数为 0 时，无该条目）、5=今收盘价条目。
 - c) 特殊的，停牌状态（TradingPhaseCode 第 1 位为 ‘F’）产品循环组包含 2-3 条记录，包含 MDEntryType 为 0=买入、1=卖出两个条目（价格 MDEntryPx、数量 MDEntrySize 均取值为 0）及 5=今收盘

⁶ 仅对 ETF 产品，此字段有效。该字段取 0 时，无意义。当 ETF 公告文件 PublishIOPVFlag 字段（是否需要发布 IOPV）取值为 1 时，此字段为对应产品的 IOPV 值；当 ETF 公告文件 PublishIOPVFlag 字段取值为 0 时，此字段固定取 0，无意义；

价条目（仅盘后固定价格交易时间⁷及闭市后存在该条目）。

11. 对债券非匹配成交快照行情（MDStreamID=MD210），扩展字段中仅最新成交价、今开盘价、今收盘价、当日最高成交价、当日最低成交价、当日成交均价和昨日成交均价有意义，均仅计入债券非匹配成交（含点击成交、询价成交、竞买成交和协商成交）的交易；基础字段中前收盘价对于双边挂牌产品填写其匹配成交收盘价，单边挂牌产品填写其非匹配成交参考价。

对于公募可转债和公募 REITs，无当日成交均价、昨日成交均价和今收盘价信息，其今收盘价参考对应匹配成交行情或盘后文件。

2.5.4 逐笔行情消息（MsgType=M201）

逐笔行情包括逐笔委托行情和逐笔成交行情，两者按照发生的顺序在一个数据流中发送。

同一个频道内的逐笔委托和逐笔成交消息的消息记录号（ApplSeqNum）统一连续编号，VSS 需要通过逐笔序号的连续性判断是否丢包，并通过 TCP 连接向上发起重传请求。

消息格式如下：

域名	说明	类型
标准消息头	<i>MsgType=M201</i>	
SecurityType	证券类型 13 = 债券（非匹配成交）	uint8
ChannelNO	频道号	unit16, N4
ApplSeqNum	消息记录号，从 1 开始计数	uint64, N18
MDStreamID	行情类别	char[5]
SecurityID	产品代码	char[8]
TransactTime	委托或成交的行情生成时间：HHMMSSsss	uint32, N9

⁷ “盘后固定价格交易时间”请参照《上海证券交易所交易规则》。

ExecType	类型： 对于逐笔委托： 0 = 新增委托订单 4 = 删除委托订单 对于逐笔成交： F = 成交	char[1]
Price	逐笔委托时表示委托价格，如意向申报时未 填写价格或删除委托订单时，则为 0； 逐笔成交时表示成交价格	uint64,N13(5)
OrderQty	逐笔委托时表示委托数量，如意向申报时未 填写数量或删除委托订单时，则为 0； 逐笔成交时表示成交数量	uint64,N15(3)
ExtendFields	扩展字段	
标准消息尾		

2.5.4.1 逐笔委托行情扩展字段

域名	说明	类型
Side	买卖方向 1 = 买 2 = 卖	char[1]

QuoteID	订单编号，删除委托时表示被删除订单的 订单编号	char[18]
MemberID	报价方交易参与人机构代码 如发起方选择匿名，填空	char[12]
TraderCode	报价交易员一债通账户 如发起方选择匿名，填空	char[12]
FullAmountTrade	是否全额成交 1 = 是 2 = 否	uint8, N1
SettlType	结算方式 1 = 净额结算 2 = RTGS	uint8, N1
ContractMultiplier	折算比例（%），如发起方未填写此字段， 填 0	uint32, N5(2)
ExpirationDays	期限（天），如发起方未填写此字段，填 0	uint16, N3
TotalValueTraded	成交金额，如发起方未填写此字段，填 0	uint64, N16(2)
BasketID	篮子信息 左起顺序代表第 1 号至第 N 号篮子。例如 指定 1, 2, 5 号篮子，填 “1100100000000000”	char[16]

InvestorName	投资者名称	char[32]
BidTransType	竞买申报类别 1 = 竞买预约 2 = 竞买申报 3 = 单一主体中标应价申报	uint8, N1
BidExecInstType	竞买申报方式 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	uint8, N1
ExpireDate	竞买日期: YYYYMMDD	uint32, N8
SecondaryOrderID	应价申报对应的订单编号(也即竞买申报编号)	char[18]
MinQty	最低成交数量, 预留	uint64, N12

注:新增委托订单时扩展字段与业务的对照关系参见下表,删除委托订单时仅 Side、QuoteID、

MemberID 和 TraderCode 有效(其他字段均无意义)

MDStreamID	MD211	MD212	MD213	MD214	MD215	MD216 ⁸		
域名	确定 报价	待定 报价	现券 意向	协议回 购意向	三方回 购意向	竞买 预约	竞买 申报	应价 申报 ⁹
Side	●	●	●	●	●	●	●	●
QuoteID	●	●	●	●	●	●	●	●

⁸ 不同竞买委托方式可根据 BidTransType 字段进行区分。

⁹ 仅适用于竞买单一主体中标应价申报。

MemberID	●	●	●	●	●	●	●	●
TraderCode	●	●	●	●	●	●	●	●
FullAmountTrade	●	●						
SettlType	●	●				●	●	●
ExpireDate						●		
ContractMultiplier				●				
ExpirationDays				●	●			
TotalValueTraded				●	●			
BasketID					●			
InvestorName					●			
BidTransType						●	●	●
BidExecInstType						●	●	
SecondaryOrderID								●
MinQty ¹⁰						●	●	

2.5.4.2 逐笔成交行情扩展字段

域名	说明	类型
BidApplSeqNum	买方订单编号 (QuoteID) 无对应委托时填空	char[18]

¹⁰ 仅适用于竞买多主体中标 (含单一价格或多重价格中标)。

OfferApplSeqNum	卖方订单编号 (QuoteID) 无对应委托时填空	char[18]
TotalValueTraded	成交金额	uint64,N16(2)
TradeMethod	成交方式 0 = 匹配成交 (预留) 1 = 确定报价点击成交 2 = 待定报价点击成交 3 = 询价成交 6 = 协商成交 (含合并申报) 7 = 竞买单一主体成交 10 = 竞买多主体单一价格成交 11 = 竞买多主体多重价格成交	uint8,N2
MarginPrice	边际价格 仅适用于竞买多主体成交	uint64,N13(5)

3 文件接收

用户可通过行情网关接收的行情及相关参考文件主要包括：

文件名	描述
csiYYYYMMDD.txt	指数通行情文件
reff03MMDD.txt	期权基础信息文件
reff0302YYYYMMDD.xml	期权基础信息第二版文件
trdses04.txt	港股通交易盘实时状态
mktdt04.txt	港股行情文件
mktdth.txt	B 转 H 行情文件
mktdt06.txt	基金通行情文件
cpxx0201MMDD.txt	产品信息文件第二版本第一批次
cpxx0202MMDD.txt	产品信息文件第二版本第二批次
fjyYYYYMMDD.txt	产品非交易基础数据
tpxxxhmmss.txt	盘中停牌公告文件
ztxxxhmmss.txt	债券盘中停牌公告文件
products_yyyymmdd.xml	证券产品基础信息文件
bondmbrs_yyyymmdd.xml	债券交易参与人信息文件
bondtrdrs_yyyymmdd.xml	债券交易员信息文件
reff04MMDD.txt	港股通基础信息文件
extra04MMDD.txt	港股通参考汇率文件

zxjcMMDD.txt	港股通最小价差文件
--------------	-----------

具体文件格式及相关描述详见本所各平台市场参与者接口规格说明书。

4 转发行情

目前，用户可通过本所行情网关转发任务获取的外部行情数据包括：

- 深交所 V5 行情

具体数据接口规范请参考《深圳证券交易所 Binary 行情数据接口规范》，对转发任务的网络及带宽限制要求请参考《上海证券交易所行情网关用户网络接入指引》。

后 记

上海证券交易所对本文档享有知识产权，未经上海证券交易所书面许可，任何单位和个人不得将本文档用于其他商业目的。

本文档编写过程中，深受证券业界信息技术同仁讨论启发，特此致谢。对本文档有任何批评指正意见，可通过电子邮件或服务电话进行反馈。

电子邮件：tech_support@sse.com.cn

服务电话：4008888400-2

网站地址：<https://www.sse.com.cn/> - 交易技术支持专区

通讯地址：上海市浦东新区浦东南路 528 号 上交所技术公司技术开发总部

附录一、计算校验和（资料性附录）

以下为计算校验和的代码段：

```
uint32 CalcChecksum(const char* buffer, uint32 len)
{
    uint8 checksum = 0;

    uint32 i = 0;
    for (i = 0; i < len; i++)
    {
        checksum += (uint8)buffer[i];
    }

    return (uint32)checksum;
}
```