



上证所Level-2行情接口开发

高昀



主要内容



- Level-2产品介绍
- Level-2系统接入
- Level-2数据解码



一、Level-2 产品介绍



Level-1行情的不足

Level-1行情Show2003. dbf不足之处

内容

- 信息量少，市场深度反映不够——Show2003只有29个有效字段
- 实时信息产品单一——Show2003 为仅有的实时行情产品
- 按同一模式显示不同的交易品种——显示债券、基金信息时出现“市盈率”字段名

技术

- 5秒钟一次的快照，文件交换方式，发布延时明显
- 信息内容可扩展性差——dbf文件是固定字段，只能通过增加数据库字段方式来增加信息内容



从Level-1到Level-2



- 商业模式：“免费”->收费
- 发布路径：券商->信息商
- 行情深度：五档->十挡，新增委托明细和全部成交信息
- 行情频率：5秒->3秒
- 数据格式：dbf->FIX/STEP->FAST
- 交换方式：文件->文本数据流->二进制数据流
- 行情延时：3秒->28毫秒->20毫秒





Level-2内容



- Reference Data (UA101)
- Transaction(UA3101)
- Market Data(UA3102,UA3113)
- Order Queue(UA3111)



上證所信息网络有限公司
SSE INFONET LTD



行情展示

Transaction(UA3111)

14:06:05	5.07	100	↑
2	5.07	10	↑
14:06:06	5.06	5	↓
2	5.06	8	↓
14:06:07	5.07	20	↑
14:06:08	5.07	160	↑
2	5.07	10	↑
14:06:09	5.07	2	↑
2	5.07	67	↑
14:06:13	5.06	12	↓
2	5.06	8	↓
14:06:14	5.07	41	↑
14:06:16	5.07	5	↑
14:06:17	5.06	12	↓
2	5.06	10	↓
14:06:18	5.07	3	↑
14:06:19	5.07	20	↑
14:06:20	5.07	10	↑

均卖价	3.70	总卖量	23.26万
卖⑩	3.65		26079
卖⑨	3.64		11463
卖⑧	3.63		16036
卖⑦	3.62		17518
卖⑥	3.61		14677
卖⑤	3.60		11693
卖④	3.59		9094
卖③	3.58		2065
卖②	3.57		2951
卖①	3.56		146
买①	3.55		14071
买②	3.54		8412
买③	3.53		3551
买④	3.52		2802
买⑤	3.51		2857
买⑥	3.50		8080
买⑦	3.49		166
买⑧	3.48		1359
买⑨	3.47		247
买⑩	3.46		339
均买价	3.50	总买量	50562
总笔数	9637	均笔量	35

卖一 3.56 146手 6笔 24.3手/笔	买一 3.55 14071手 367笔 38.3手/笔
10 10 50 22 15 39	3 21 10 50 10 10 10 20
	20 10 10 10 3 25 5 2
	3 30 18 20 21 10 5 1
	10 5 10 20 5 10 5 1
	10 5 3 10 56 250 6 5
	10 200 9 100 10 150 7 14
	2 10

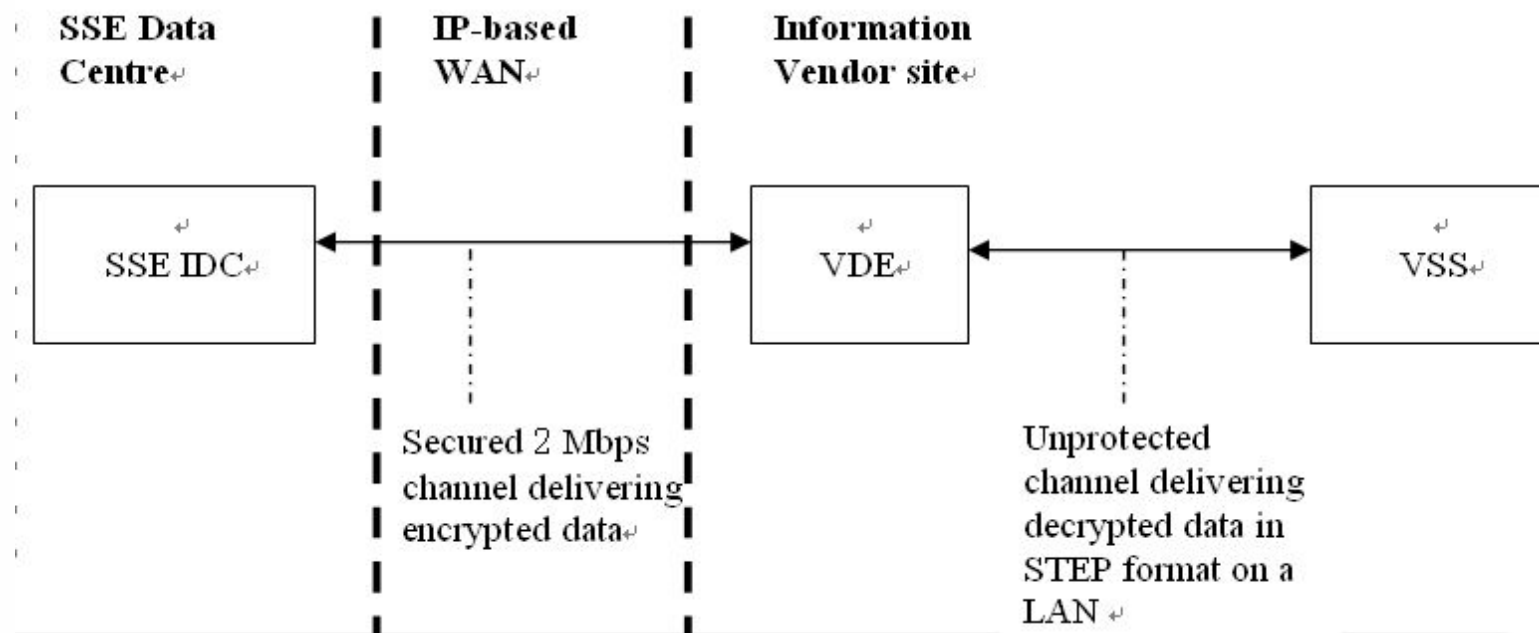
Order Queue(UA3111)

Market Data(UA3102)



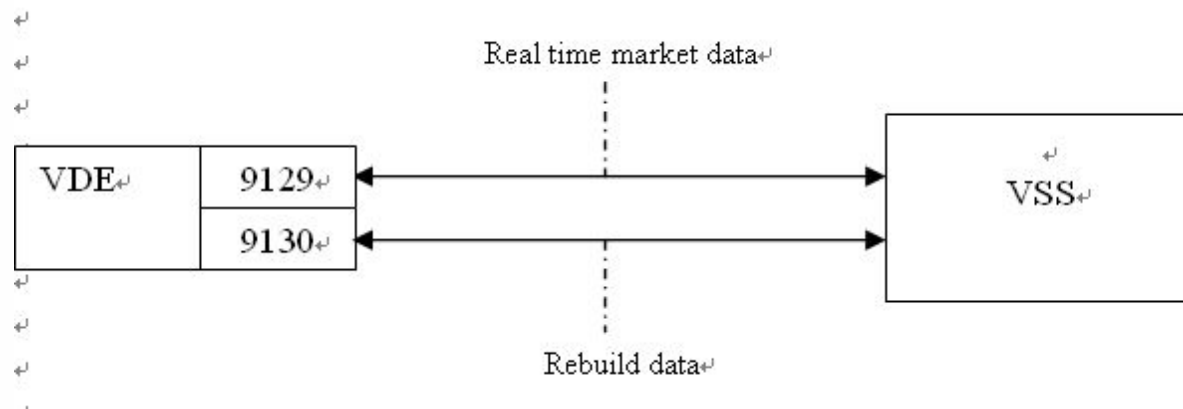
二、Level-2系统接入

SSE to Vendor Connection



- 1.VDE(Vendor Data Engine)
- 2.VSS(Vendor Supplies System)

VDE to VSS Connection



Daily Procedure



Time (GMT+8)	VDE
0830	IDC starts sending market overview with trade date only Reference data ready for download by VSS
0915	SSE virtual auction price data sending
0925	SSE morning trading session open
1130	SSE morning trading session close
1300	SSE afternoon trading session open
1500	SSE afternoon trading session close
1515	IDC resends a complete set of market information to VSS.





三、 Level-2 FAST数据解码



Level-2 FAST的介绍

FIX Aadapted for Sstreaming (FAST)

- 为解决FIX协议传输市场数据存在冗余度高，带宽需求大的问题，芝加哥商品交易所于2003年11月17日向FPL提交了一个隐示标记 (Implicit tags) 方案
- FPL于2004年12月27日宣布成立市场数据优化工作组 (MDOWG, Market Data Optimization Working Group).
- 2005年5月开始MDOWG根据一系列POC的结果，开始进行协议标准制定
- 2006年初完成FAST (FIX Adapted for STreaming) V1.0
- 2006年12月完成FAST V1.1



Level-2 FAST的介绍

- 中国证券市场的特点
 - 1.2亿交易帐户
 - 较高的换手率，上海证券交易所日交易金额峰值超过3000亿，成交笔数超过4000万
- 市场数据的发展趋势和挑战
 - 产品越来越多、挂牌证券数量越来越大
 - 更多的价格档位、更多的市场深度
 - 更少的延时

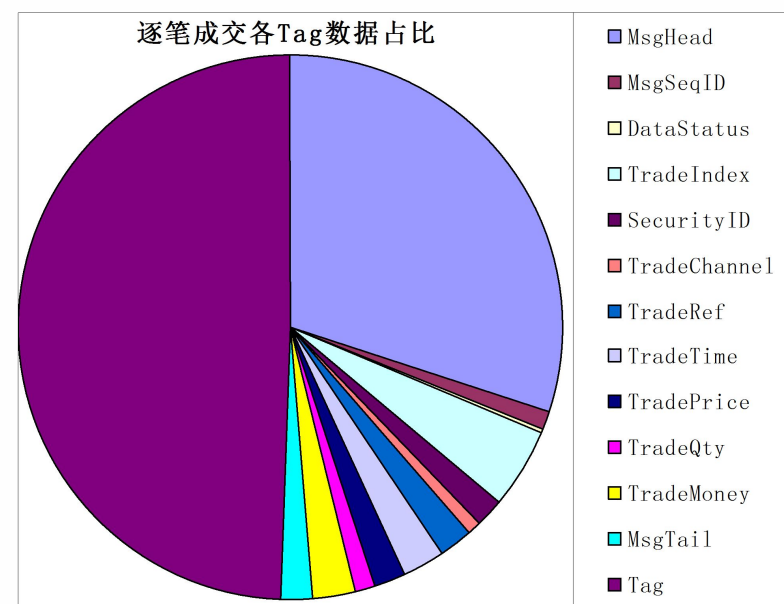
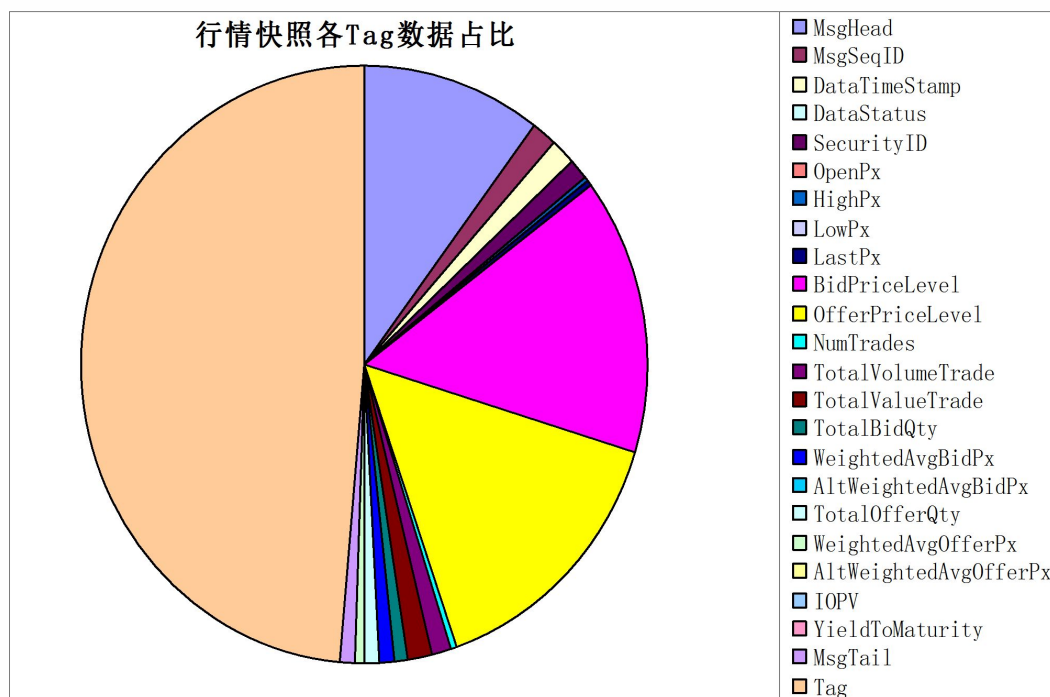


Level-2 FAST的介绍

- STEP协议的优缺点
 - 基于FIX，已成为事实上的证券数据标准
 - 语法简单，定义灵活，易扩展
 - 数据相对冗余
- $8 = STEP. 1. 0. 0 < SOH > 9 = 1 1 2 < SOH > 3 5 =$
 $D < SOH > 4 9 = S e n d e r < SOH > 5 6 = T a$
 $r g e t < SOH > 3 4 = 2 3 5 < SOH > 1 = A 1 2$
 $3 4 5 6 7 8 9 < SOH > 5 5 = 青 岛 啤 酒 < SOH > 48$
 $= 600600 < SOH > 5 4 = B < SOH > 44 = 8. 520 < S$
 $O H > 38 = 1000 < SOH > 6 0 = 20030703-09: 35:$
 $28 < SOH > 11 = 825 < SOH > 10 = 157 < SOH >$



STEP 的冗余



平均每条消息长度为467字节,FAST编码后为127字节

平均每条消息长度为160字节,FAST
编码后为25字节



Level-2 FAST的介绍



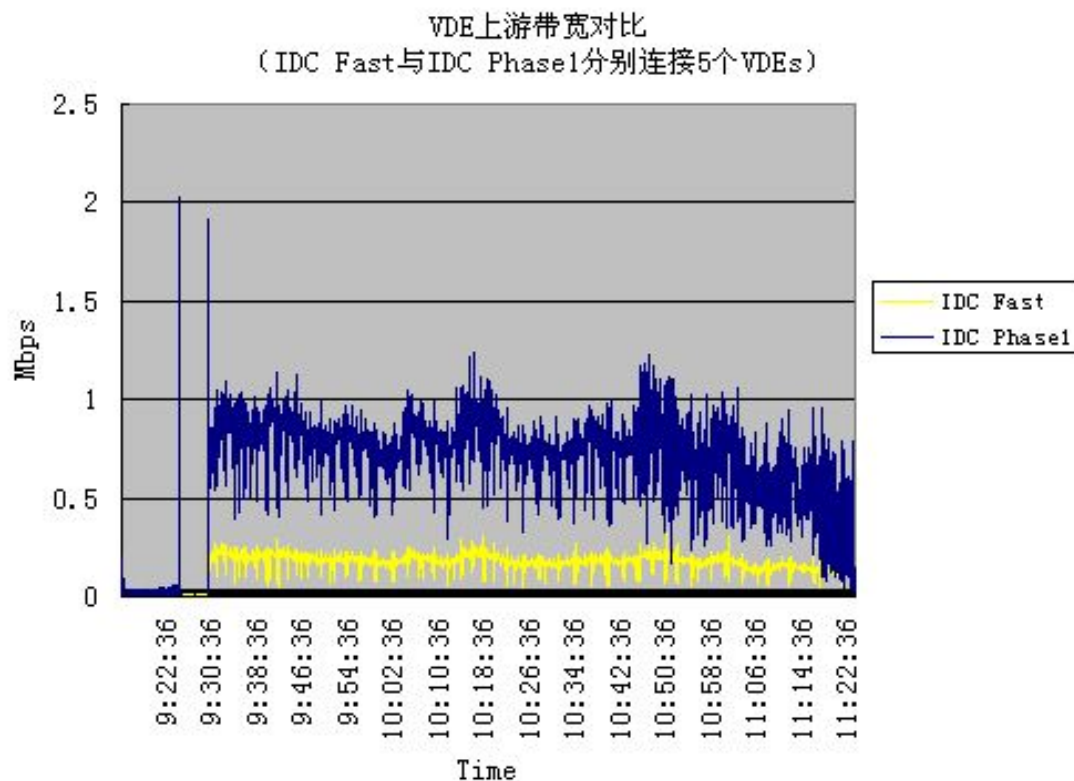
- FAST的技术优势
 - 高压缩比(70-80%)
 - 低资源消耗
 - 算法简单,效率高,每秒百万级别消息的处理能力



FAST系统测试结果



◆带宽利用率与STEP系统对比测试结果比较



测试结果:

FAST版本
最大值为0.679Mbps;
平均值为0.162Mbps

STEP版本
最大值为2.23Mbps;
平均值为0.668Mbps

**FAST版本平均值为STEP版本的
24.2%**



上咨所信息网络有限公司
SSE INFONET LTD



FAST关键技术

- Field Encoding
 - 隐式标签（Implicit Tag）(模版 Template)
 - 提供消息层面的优化
 - 操作符（Operator）
 - 消息层面和tag层面的优化
- Transfer Encoding
 - 二进制数据流
 - Presence Map(PMAP)
 - Stop Bit Encoded Entities





Field Encoding—Implicit Tag

- Implicit tag
 - 将FIX的tag=value格式进行分离，变成消息模板和消息内容。
 - 消息模板在双方通信之前就确定，实际只需要传输消息内容

Field Encoding

- 常用数据类型

Data type	Description
string	String data type
int32	integer data type
int64	long data type
sequence	repeating group



Field Encoding

Field Encoding Operator

Entry	Description
Constant	Constant value
Default	Default value if not specified
Copy	Copy previous value if not specified
Increment	Increment previous value if not specified
Delta	Delta value from previous value
Tail	Tail value from previous value



Field Encoding Operator

- Constant Value

用途：主要用于表示模板中的固定数据

constant value	value in stream	output value
"VALUE"	N/A	"VALUE"



Field Encoding Operator

- Default Value

用途：主要用于表示默认值

default value	value in stream	output value
"VALUE"	N/A	"VALUE"
"VALUE"	"HELLO"	"HELLO"



Field Encoding Operator

- Copy Value

用途：主要用于表达相邻数据相同的情况

previous value	value in stream	output value
"VALUE"	N/A	"VALUE"
"VALUE"	"HELLO"	"HELLO"



Field Encoding Operator

- Increment Value

用途：主要用于表示前后数据有规律的递增的情况

previous value	value in stream	output value
5	N/A	6
5	10	10

Field Encoding

BeginStr	SeqNum	SenderID	SendingTime	Price	Symbol
8=FIX.4.4	34=10000	49=CLIENT1	52=20060126-13:06:58.100	44=1200	55=F001
8=FIX.4.4	34=10001	49=CLIENT1	52=20060126-13:06:58.200	44=1210	55=F001
8=FIX.4.4	34=10002	49=CLIENT1	52=20060126-13:06:58.300	44=1190	55=BAR2

Original size 71 bytes

Constant

Increment

Copy

Tail

Delta

Copy

10000	CLIENT1	20060126-13:06:58.100	1200	F001
		200	10	
		300	-20	BAR2



上證所信息网络有限公司
SSE INFONET LTD



Field Encoding-Template

- 定义和语法，包括：
 - Field（Message Structure）
 - Field Encoding Operator
 - Data Type
- 如何表示：
 - XML语言



Field Encoding-Template

- XML格式的模板示例

Field Operators

Structure {

```
<template name="ExampleOrder">
  <messageRef name="NewOrderSingle"/>
  <string name="BeginStr"> <constant value="FIX.4.4"/> </string>
  <u32 name="SeqNum"> <increment/> </u32>
  <string name="SenderID"> <copy/> </string>
  <string name="SendingTime"> <tail/> </string>
  <decimal name="Price"> <delta/> </decimal>
  <string name="Symbol"> <copy/> </string>
</template>
```

Data Types



上證所信息网络有限公司
SSE INFONET LTD



Transfer Encoding



- 对FIX数据实现二进制Byte/Bit Level优化
- 根据模板生成数据流，并且同步进行压缩的协议
- 本身具有通用性，是一个与具体的应用无关的协议。



上證所信息网络有限公司
SSE INFONET LTD

Transfer Encoding

stream ::= message* | block*
block ::= *BlockSize* message+
message ::= segment
segment ::= *PresenceMap* *TemplateIdentifier*? (field |
segment)*
field ::= integer | string | delta | *ScaledNumber* | *ByteVector*
integer ::= *UnsignedInteger* | *SignedInteger*
string ::= *ASCIIString* | *UnicodeString*
delta ::= *IntegerDelta* | *ScaledNumberDelta* |
ASCIIStringDelta | *ByteVectorDelta*



Transfer Encoding-Stop Bit

- 最高比特位
 - 0: 下一个字节属于同一个实体
 - 1: 下一个字节不属于同一个实体
- 编码举例:

Input Value	Native Hex/Binary	FAST Hex/Binary
0	0x00 0	0x80 10000000
1	0x01 1	0x81 10000001
942755	0x0e 0x62 0xa3 1110 01100010 10100011	0x39 0x45 0xa3 00111001 01000101 10100011





Transfer Encoding-Presence Map

- 在每一条消息的开头处增加标志串(BYTE 数组)
- 每一个Bit标示后面的各个field存在与否，1 存在，0不存在
- 不用传送的field不用进行编码，减少了数据量

Transfer Encoding-Presence Map

	Presence Map Bit in Required	
Operator	Mandatory	Optional
None	No	No
default	Yes	Yes
constant	No	Yes*
copy	Yes	Yes
increment	Yes	Yes



Presence Map and NULL

- NULL一般用来指示这一field没有出现（absent）

Operator	Presence	Occupy	使用NULL
None	Mandatory	no	
	Optional	no	yes
constant	Mandatory	no	
	Optional	yes	
Default,copy, increment	Mandatory	yes	
	Optional	yes	yes

Data Type Example

- uInt32 Example—Optional Number

<uInt32 id="1" presence="optional" name="Value"/>

Input Value	Native Hex/Binary	FAST Hex/Binary
null	n/a	0x80 1 0000000
0	0x00 0	0x81 1 0000001
1	0x01 1	0x82 1 0000010
942755	0x0e 0x62 0xa3 1110 01100010 10100011	0x39 0x45 0xa4 0 0111001 0 1000101 1 0100100
		Increment by 1 since field is optional Stop bits indicated in bold

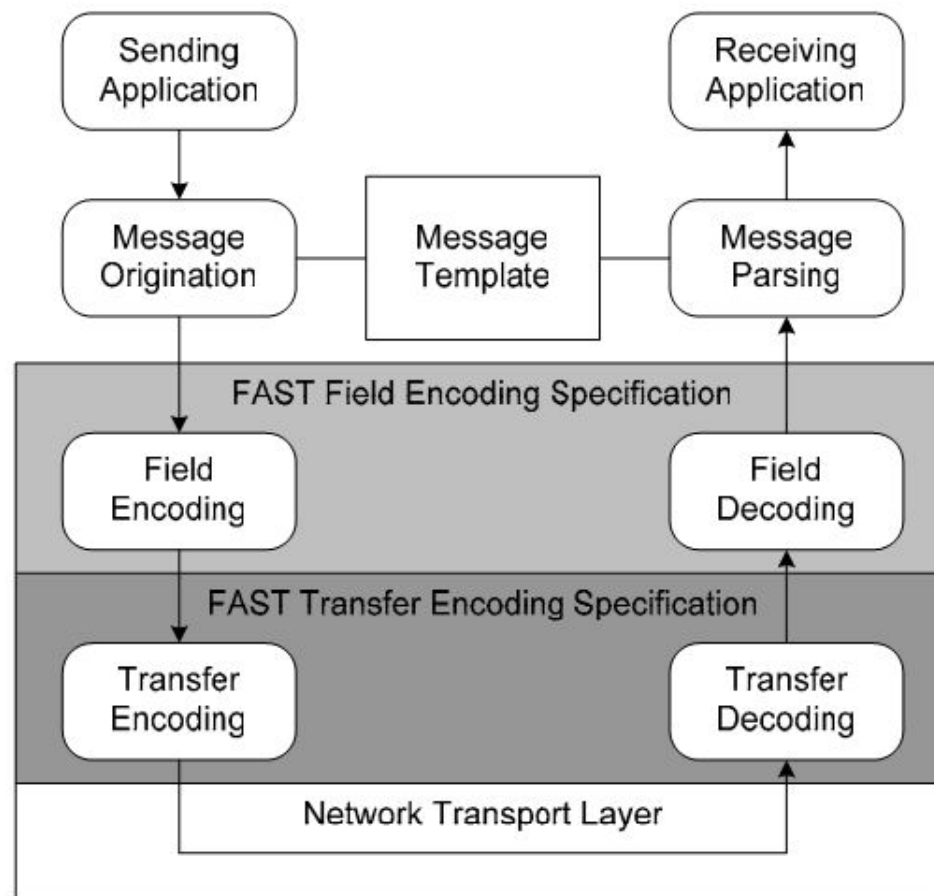


如何解析Level-2 FAST 的消息

- VSS
 - 1 Initial
 - 2 ReadTemplates
 - 3 Connect VDE
 - 4 Logon
 - 5 Receive Data
 - 6 Decode
 - 7 业务处理
 - 8 数据存储和转发



FAST的实现过程



UA3101的解码实例

```

01 28 57 66 80 01 31 30 3D 30 30 34 01 38 3D 53 .(Wf..10=004.8=S
54 45 50 2E 31 2E 30 2E 30 01 39 3D 31 30 35 36 TEP.1.0.0.9=1056
37 01 33 35 3D 55 41 33 31 30 31 01 34 39 3D 56 7.35=UA3101.49=V
44 45 01 35 36 3D 56 44 52 01 33 34 3D 30 01 35 DE.56=VDR.34=0.5
32 3D 32 30 31 30 30 31 32 32 2D 30 39 3A 32 35 2=20100122-09:25
3A 30 37 01 31 30 31 34 32 3D 37 01 31 30 30 37 :07.10142=7.1007
32 3D 30 01 39 35 3D 31 30 34 38 38 01 39 36 3D 2=0.95=10488.96=
5F E0 18 9D 81 86 36 30 31 37 36 B6 04 34 4A 83 .....60176..4J.
29 83 1E 42 A1 00 7D 15 5D A1 E6 18 9D 82 80 80 )..B..}.].....
41 E0 18 9D 29 83 06 0D A1 19 04 2C A1 E6 18 9D A...).....
82 80 80 41 E0 18 9D 29 83 1E 42 A1 00 7D 15 5D ...A...)..B..}.]
A1 E6 18 9D 82 80 80 41 E0 18 9D 29 83 02 18 61 .....A...)..a
C1 08 7F 47 73 C1 E6 18 9D 82 80 80 41 E0 18 9D ...Gs.....A...
29 83 12 27 E1 00 4B 0D 04 E1 E6 18 9D 82 80 80 )...'...K.....
41 E0 18 9D 29 83 1E 42 A1 00 7D 15 5D A1 E6 18 A...)..B..}.]...
9D 82 80 80 41 E0 18 9D 29 83 00 43 11 E1 02 13 ....A...)..C....

```

PMAP 5F E0 1011111 11100000 10111111100000

TID 占位=1 18 9d 11000 10011101 110000011101 3101

<string name="MessageType" id="35"><constant value="UA3101"/></string>

35 不占位 UA3101

<int32 name="DataStatus" id="10121" presence="optional"><default/></int32>

10121 占位=0

<int32 name="TradeIndex" id="10011"><increment/></int32>

10011 占位=1 81 10000001 00000001 1

<int32 name="TradeChannel" id="10115"><copy/></int32>

10115 占位=1 86 10000110 00000110 6



上證所信息网络有限公司
SSE INFONET LTD

UA3101的解码实例

<string name="SecurityID" id="48" presence="optional"><copy/></string>

48 占位=1 36 30 31 37 36 B6 601766

<int32 name="TradeTime" id="10013" presence="optional"><copy/></int32>

10013 占位=1 04 34 4A 83 100 0110100 1001010 0000011 9250051 9250050

<int32 name="TradePrice" id="10014" presence="optional"

decimalPlaces="3"><default/></int32>

10014 占位=1 29 83 101001 0000011 5251 5250 5.250

<int64 name="TradeQty" id="10015" presence="optional"

decimalPlaces="3"><default/></int64>

10015 占位=1 1E 42 A1 11110 1000010 0100001 500001 500000 500.000

<int64 name="TradeMoney" id="10016" presence="optional"

decimalPlaces="5"><default/></int64>

**10016 占位=1 00 7D 15 5D A1 1111101 0010101 1011101 0100001 262500001
262500000 2625.00000**

**35=UA3101|10011=1|10115=6|48=601766|10013=9250050|10014=5.250|10015
=500.000|10016=2625.000|**



上 證 所 信 息 網 絡 有 限 公 司
SSE INFONET LTD



谢 谢