EURONEXT

PREFACE

DOCUMENT HISTORY

The following table provides a description of all changes to this document.

For historical document history details please see Appendix F. The Current Version change history is provided in the following section.

VERSION NO.	DATE	CHANGE DESCRIPTION
3.3.0	May 2016	- Modification of reference Data – 553 Message to provide more information in existing fillers, without impact on the message length and structure. The following new fields have been added: * **GuaranteeIndicator** * **ParValue** * **ParValueScaleCode** * **MaturityDate** * **SettlementDelay** * **DarkEligibility** * **DarkMinimumQuantity** * **DarkLISThreshold** - Modification of Trade Full Information – 240 Message to add the following new fields with a change to the length of the message: * **TradingDate** * **TradingTime** * **TradingTimeMicroSecs** * **TransparencyInd** - Modification of Price Update – 241 Message to include a new value for the existing field **TypeOfPrice** - Modification of Order Update – 230 Message: definition of value **'M'** updated (field **ActionType**) - Modification of 'Message Type' sections 4.4, 4.5, 5.4, and 5.5: Value **'EW'** removed from channel 109, 209, 2 and 12
3.4.0	July 2016	- Clarification of the definition of the following fields: * **GuaranteeIndicator** * **DarkEligibility** * **DarkMinimumQuantity** * **DarkLISThreshold**
3.4.1	August 2016	- Addition of value '00030' for the **DepositaryList** field of reference Data – 553 Message.

For further information in relation to these modifications, please visit:

https://www.euronext.com/en/exchange-publications/cash/info-flashes

CONTACT INFORMATION

For technical support please contact the Customer Technical Support Group at CTSG@euronext.com /
+33 1 8514 8588.

FURTHER INFORMATION

For additional product information, visit: https://www.euronext.com/market-data

For details of IP addresses, visit our IP address pages at: https://www.euronext.com/en/it-documentation/market-data (Technical Documentation tab).

CONTENTS

1. EURONEXT CASH MARKETS PROCESSING INFORMATION

1.1 EXCHANGE DATA PUBLISHER OVERVIEW

The Exchange Data Publisher (XDP) provides high-speed, real-time market data for Euronext European Cash markets.

The data feed has the following high-level features:

- Multicast technology

- High Availability

- Ultra-low latency

- Reliable network solution

- High level of scalability

- Access to wide range of European market data sets

This chapter provides detailed information about the features of the feed, to support the development of client applications by Members, Independent Software Vendors and Quote Vendors.

The other chapters of this document provide details that are specific to each of the Euronext market data sets, including formats for each message type.

1.2 ACCESS TO MARKET DATA

Clients access market data as follows:

- **Real time service**   Clients connect to multicast channels to receive market data messages

- **Retransmission service**   Clients connects to a TCP/IP server for packet retransmissions

- **Refresh service**   Clients connect to a TCP/IP server for the initial request–response transaction and the data is then sent over multicast channels

Figure 1 Access to Market Data
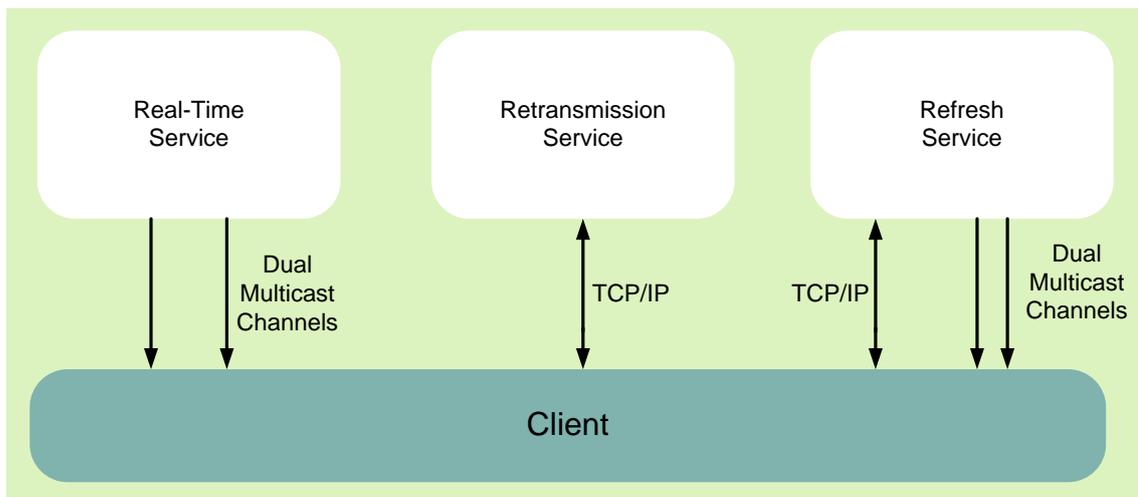
1.2.1   Real Time Market Data

Real-time market data is message-based over the UDP IP protocol with fixed length binary and ASCII fields.

It uses the push-based publishing model. This means that data will be published based on its availability. Once an update is available, it will be published to the appropriate multicast group.

For capacity reasons, market data is split across a number of multicast groups organized into predefined data sets (channels).

Each multicast group will deliver a set of data for a certain market segment.

The client application will be responsible for issuing multicast subscriptions to one or more of the multicast groups assigned to each product.

See Production Feed Configuration and External User Acceptance Feed Configuration for detailed content of Production and External User Acceptance multicast groups.

The process of subscribing to a multicast group ID is also known as 'joining' a multicast group. Upon session termination, the client's host system should issue an 'unjoin' message. This will terminate delivery of data to that host's local network. If a client application terminates without issuing an 'unjoin' message, the network will eventually issue a 'timeout' for the multicast group subscription that will automatically terminate delivery of the multicast packets to the host's local network.

The 'join' and 'unjoin' processes are standard functions. No specific instructions are provided here, as they are specific to the user's operating system and programming language.

1.2.2   Retransmission Functionality

The retransmission functionality is designed to allow the user to recapture a small number of missed packets.

Note: The Packet Sequence Reset packet (packet number one - see Packet Sequence Number Reset) is *not* available on the Retransmission Server.

It is not intended that clients use the retransmission functionality to recover data after long outages or on late start up. Accordingly, the number of packets that the user can request is limited. The number of retransmission requests permitted per user is also limited per day.

The client makes a TCP/IP connection with the Retransmission Server, and receives the requested messages also via the TCP/IP channel.

Figure 2 Retransmission Request shows the sequence of messages and the transport protocols employed when making a retransmission request.

Figure 2 Retransmission Request
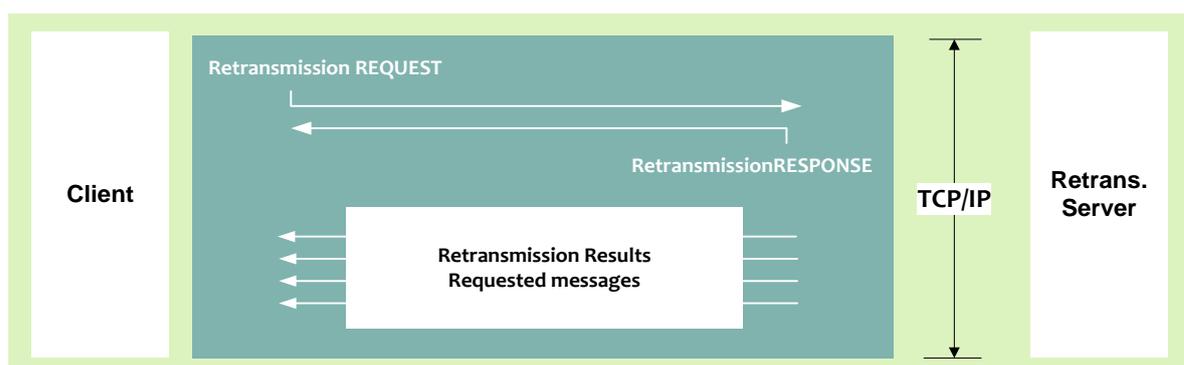
The retransmission request will include a Source ID (username) which will be validated by the Retransmission Server. It is important to note that only one Source ID can be used per application session.

The retransmission request may be rejected for any of the following reasons:

■ Invalid Source ID

■ Invalid Packet Sequence Number (PSN)

■ PSN no longer in cache, or is not yet available due to an incorrect request

■ Total number of packets requested in the current day exceeds the predefined system limit

■ Number of retransmission requests in the current day exceeds the predefined system limit

■ Retransmission request message incorrectly formatted

In the case of such a failure, the user will receive a retransmission response with a rejection code to advise of the reason for failure.

If an application connects with a Source ID that is currently in use, the application will be disconnected and no retransmission response will be sent.

Applications which send 200 invalid requests (for any of the reasons above) will no longer be serviced by the Retransmission Server, for the current trading day. The offending applications will not be disconnected and will continue to receive heartbeats, however any requests, valid or otherwise, will be ignored and not responded to by the Retransmission Server. In this situation clients may use a separate Source ID, or recover via the refresh functionality.

1.2.3   Refresh Functionality

The Refresh Server supplies (on demand) a snapshot including reference data, last trade price, high, low and the order book.

The refresh messages are compressed using the zlib compression format, which delivers a significant reduction in network latency.

The Refresh Server is designed to allow the user to update the market state within their applications before restarting in real-time, following a data outage or late start.

The client makes a TCP/IP connection to the Refresh Server for requesting the refresh, whilst also joining the refresh multicast channels for receiving the refresh messages.

The Refresh Server will respond to a request with a Refresh Response message to indicate whether the request was accepted or rejected.

The refresh messages will then be sent on the multicast channels. This will be preceded by a Start of Refresh message and followed by an End of Refresh message. No dedicated retransmission service is available for the refresh; if packet loss is detected, clients should submit another refresh request.

The refresh request will include a Source ID (username) which will be validated by the exchange system. It is important to note that only one Source ID can be used per application session.

A pair of refresh multicast channels will be provided for each corresponding real-time service. The contents of the refresh and message formats will correspond to the contents and message formats contained in the appropriate real-time service.

Figure 3 Refresh Request shows the sequence of messages and the transport protocols employed when making a refresh request.

Figure 3 Refresh Request
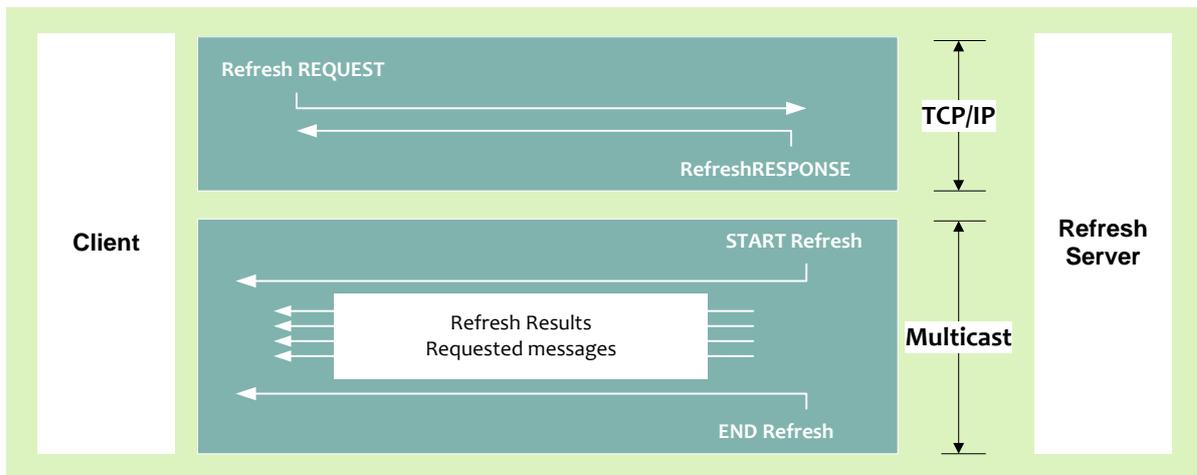
The refresh request may be rejected for any of the following reasons:

- Invalid Source ID

- Invalid Service ID

- Incorrectly formatted message sent

- Incorrect packet type sent

- Total number of refreshes requested in the current day exceeds the predefined system limit

- Rejected due to unavailability of refresh data

In the case of such a failure, the user will receive a rejection message to advise of the reason for failure.

1.3   PROCESSING GUIDELINES

1.3.1   General Processing Notes

The following processing notes apply to all messages:

■   All fields will be sent for every message

■   Only field values will appear in the published messages (for example, no names or 'tags' will appear in the message)

■   The field names that appear in the message format documents are for reference purposes only

■   All the fields are contiguous, with reserved fields for alignment issues

■   All field sizes are fixed and constant

■   Binary fields are provided in network byte order (big-endian format)

■   ASCII string fields are left aligned and null padded

■   Segmentation of messages across packets will not be supported. This means that a message will never straddle a packet boundary.

1.3.2   Packet Structure

In the feed, messages are sent contained within packets. All packets of data sent on the XDP feed will have a common packet header followed by one or more messages (with the exception of some technical messages that do not contain any messages). Messages within a packet are laid out sequentially, one after another without any spaces between them. Here is a graphic that illustrates message sequencing within a packet:

PACKET HEADER	MESSAGE 1	MESSAGE 2	...	MESSAGE N
16 bytes	n1 bytes	n2 bytes	...	nN bytes

The beginning of the *nth* message is determined by:

start_of_packet + 16 + accumulated_size_of_the_nth_previous_messages

Client applications should check if the total length of messages inside the current packet + 16 (packet header length) corresponds to the PacketLength provided in the packet header. If not, the packet should be considered corrupted. This rule applies only to market data packets; packet types 1, 2, 10 and 20 do not contain any market data messages.

The packet header format is the same for all packets, and contains packet length, number of messages within the packet, Packet Sequence Number, and so forth.

The format of each message in the packet depends on message type, however each message will start with message size (MsgSize) and message type (MsgType).

The maximum length of a packet is 1400 bytes.

A packet will only ever contain complete messages. A single message will never straddle multiple packets.

The message size will never exceed the maximum packet length (less the packet header size).

The packet header provides information including the total packet length, a Packet Sequence Number, the number of messages within the packet and a send timestamp. The format is described in detail in Packet Header Format.

The format of each message within a packet will vary according to message type.

However, regardless of the message type (with the exception of technical messages), each message will start with a two-byte message size (MsgSize) followed by a two-byte message type (MsgType). These are described in **Table 1 MsgSize and MsgType Fields**.

Table 1 MsgSize and MsgType Fields

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	-	2	Binary Int.	Size of the message in bytes (excluding these two bytes).
MsgType	-	2	Binary Int.	Type of message. See Real-Time Message Specifications for details.

1.3.3   Packet Sequence Numbers

All messages conform to the line level sequencing. Each channel has its own Packet Sequence Number (PSN). Clients can use PSNs to determine the following:

■ Missing (gapped) packets

■ Unordered packets

■ Duplicate packets

1.3.4   Detecting and Recovering Missed Data

UDP is an 'unreliable' protocol and therefore may drop packets from line A and line B.

The XDP feed provides three different mechanisms for recovering missed data:

■ **Line arbitration**   Using dual multicast channels

■ **Retransmission Server**   Recovery of a limited number of packets

■ **Refresh Server**   Snapshot of current market state

These mechanisms should be used as described in **Table 2 Recovery Mechanisms**.

Table 2 Recovery Mechanisms

EVENT	ACTION
Packet lost on one of the two lines	Try to recover data from the other line with a configurable timeout.
Dropped packet(s) on both line A and line B	Recover dropped packet(s) from the Retransmission Server.
Late start up or extended intraday outage	Request a refresh of the current market state and then continue with real time messages

1.3.4.1   Gap Detection

Each packet has a PSN that starts at one (1) and increases one-by-one and without gaps with each subsequent packet. Users should use the PSN to detect gaps in the transmission of packets.

Figure 4 Gap Detection using the Packet Sequence Number illustrates how the PSN should be used to detect gaps in the feed.

Figure 4 Gap Detection using the Packet Sequence Number
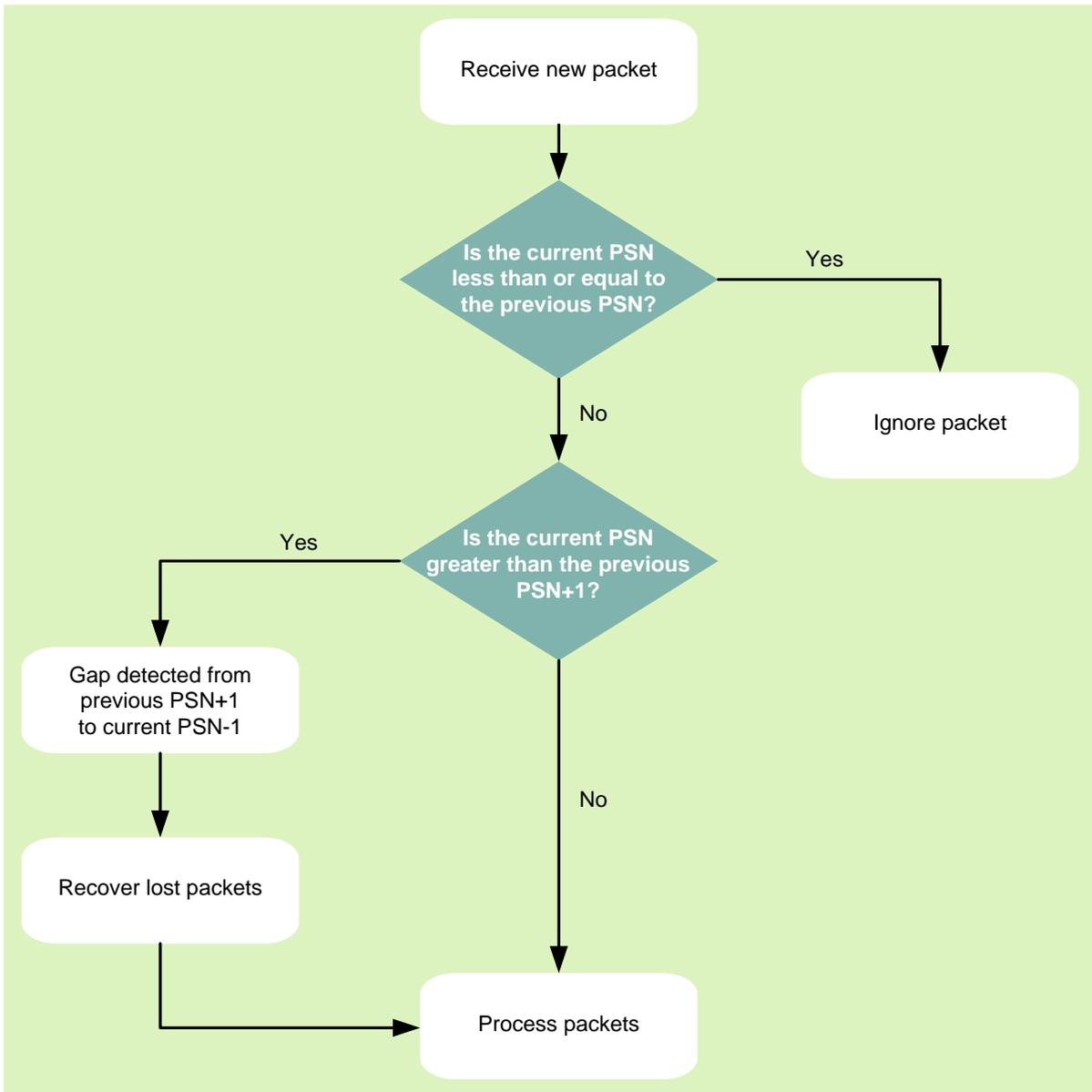
1.3.4.2 Line Arbitration

Client applications should check the PSN for every packet received. PSNs are unique and increase monotonically for each service.

Line A and line B are identical in terms of:

- Packet contents
- PSNs
- Sequence in which packets are sent

Client applications should listen to both channels in real-time. Clients should look at packets coming from both lines and process the ones that arrive first, regardless of whether they came from line A or line B. It is advisable to apply the 'first come – first served' rule.

See **Figure 5** for a schematic illustrating how missed packets are detected.

Figure 5 Detecting Missed Packets
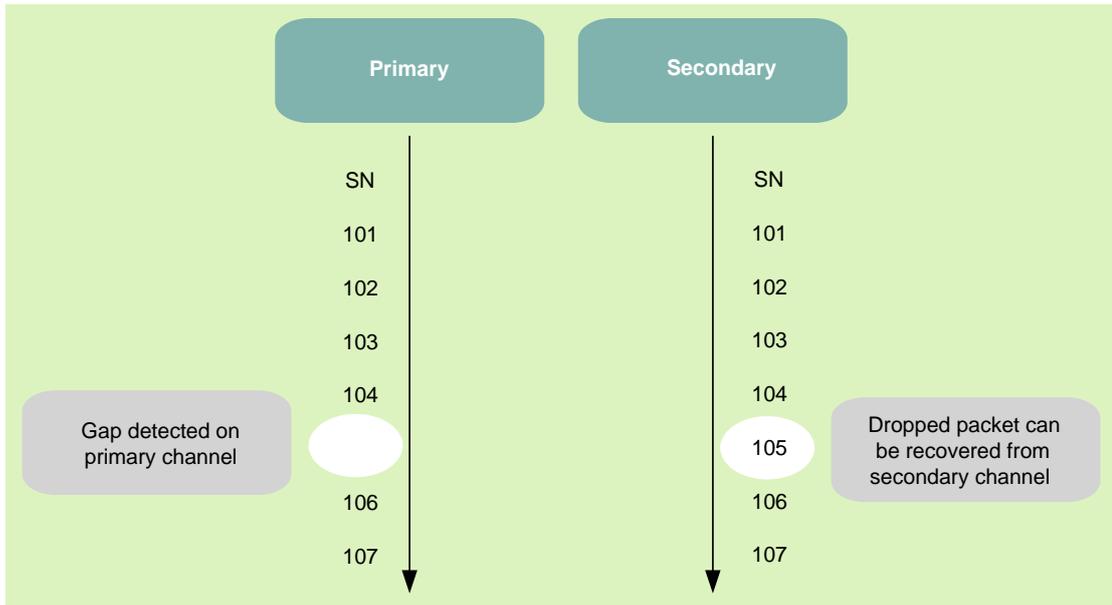
1.3.4.3   Retransmission Server

If a packet is lost from both line A and line B, clients then make a TCP/IP request to have the packets resent. Packets are resent from the Retransmission Server.

Note: The Packet Sequence Reset packet (packet number one – see Packet Sequence Number Packet Sequence Number Reset) is *not* available on the Retransmission Server.

After a client establishes a TCP/IP connection, the Retransmission Server will periodically send heartbeat request messages to the client. Clients must respond to this request with a heartbeat response within a specific timeframe – otherwise, the Retransmission Server will close the connection.

The client makes a TCP/IP connection to the Retransmission Server for both requesting and receiving retransmitted packets.

Retransmission requests should contain a Start PSN, an End PSN and a Source ID. The Source ID identifies the client application, and will be supplied by the exchange. The request can be rejected for a number of reasons as defined in Retransmission Response.

The number of retransmissions allowed per client per day is limited and the length of each retransmission is limited to a pre-defined number of packets. See Production Feed Configuration and External User Acceptance Feed Configuration for detailed descriptions of Production and External User Acceptance Retransmission Server limitations.

Figure 6 Requesting Dropped Packets illustrates the process of requesting dropped packets from the Retransmission Server.

Figure 6 Requesting Dropped Packets

```
Initiate a TCP/IP connection
with the
Retransmission Server
              │
              ▼
Send a retransmission
request
Begin + End PSN
Source ID
              │
              ▼
      Is the retransmission          No
      request accepted?  ──────────────────┐
              │                            │
             Yes                           │
              │                            ▼
      Do the received      No      Examine RejectReason
      packets fill the gap? ───┐            │
              │                │            ▼
             Yes               │     Can the request be      Yes
              │                │     reformulated based on ──────
              ▼                │     reject reason?
Close the TCP/IP connection    │            │
to the Retransmission Server   │           No
or maintain TCP/IP             │            │
connection to the Retransmission│           ▼
Server and respond to heartbeats│    Contact the Service Desk
as required
```

1.3.4.4   Refresh Server

If a client starts their application late, or experiences an outage, the Refresh Server should be used to provide the means to get back in synchronization with the real-time market. Clients send and receive the refresh request and response messages over a TCP/IP connection, and join the refresh multicast groups to receive the content of the refresh.

After a client establishes a TCP/IP connection, the Refresh Server will periodically send heartbeat request messages to the client. Clients must respond to this request with a heartbeat response within a specific timeframe – otherwise, the Refresh Server will close the connection.

The client makes a TCP/IP connection to the Refresh Server to request a refresh of packets, and at the same time joins the multicast refresh groups. Refresh requests should contain a Service ID and a Source ID. The Service ID will define for which multicast channel a refresh is required. The Source ID identifies the client application, and will be supplied by the exchange. This will be identical to the Source IDs allocated for the Retransmission Server. The request can be rejected for a number of reasons as defined in the refresh response message (see Refresh Response).

Once a successful request has been received by the server, a refresh response will be sent to the client. Clients should then process the refresh content from the refresh multicast groups and synchronize this with the real-time data.

The number of refreshes that are allowed per client per day is limited (see Refresh Request Limitations).

Note the following about the refresh service:

■ It is normal for the LastSeqNum in the 580 (Start Refresh) and 581 (End Refresh) messages to be lower than the first–spooled packet from the live feed.

   The Refresh Server receives the packets in roughly the same time as the client applications. From the moment the refresh request is made, there is a 10–second timeout until the dissemination of the refresh. This is to allow other clients to potentially ask for the same refresh.

   The client application should start spooling the live packets before sending a refresh request. This way it is very unlikely that there will be a gap between the lowest–spooled packet sequence number + 1 and the LastSeqNum. If, due to some unforeseen problems, the LastSeqNum is greater than the lowest–spooled sequence number + 1, this event should be treated as a gap and trigger a TCP retransmission request.

   If a client application started to spool packets before sending the refresh request and is able to recover potential gaps as described above, it will be in–sync with the current market state after processing the refresh and the spooled packets.

■ It is possible for a gap to occur between the LastSeqNum in the 580/581 message and the current live sequence number in EUA, with the LastSeqNum being lower.

   The bandwidth is limited in EUA, therefore if a lot of people are testing the refresh it is possible that the refresh cycles will be delayed because of throttling. Therefore the refresh received after a certain request may be a refresh requested in the past by someone else.

   Also in EUA, the refresh of the SmartPool channel provides the reference data only. Therefore only the reference data messages will update the LastSeqNum in the 580/581 message. For this reason, if there are many messages other than 553 on this channel, they will not affect the LastSeqNum but will affect the live sequence number. This situation is not possible in production.

■ You cannot request a retransmission of packets from the refresh channels over TCP. Client applications must monitor for packet loss on the refresh channels and request another refresh if there is a gap.

■ If you join the refresh channel intraday, the first sequence number you receive will be greater than 1. The probability of losing the first packet of the refresh is very low, however one possible method you could use to discover if the first packet was missed from the refresh is to have a timeout before sending a refresh request and receiving the 580 message (the timeout must be greater than 10 seconds). After this timeout, the application should re-request the refresh.

■ The refresh of the full order books does not explicitly tell your application to clear each order book. You have to clear every full depth book on the channel that is being refreshed prior to applying any full order book refresh updates.

Figure 7 Request Process from the Refresh Server illustrates the process of requests from the Refresh Server.

Figure 7 Request Process from the Refresh Server
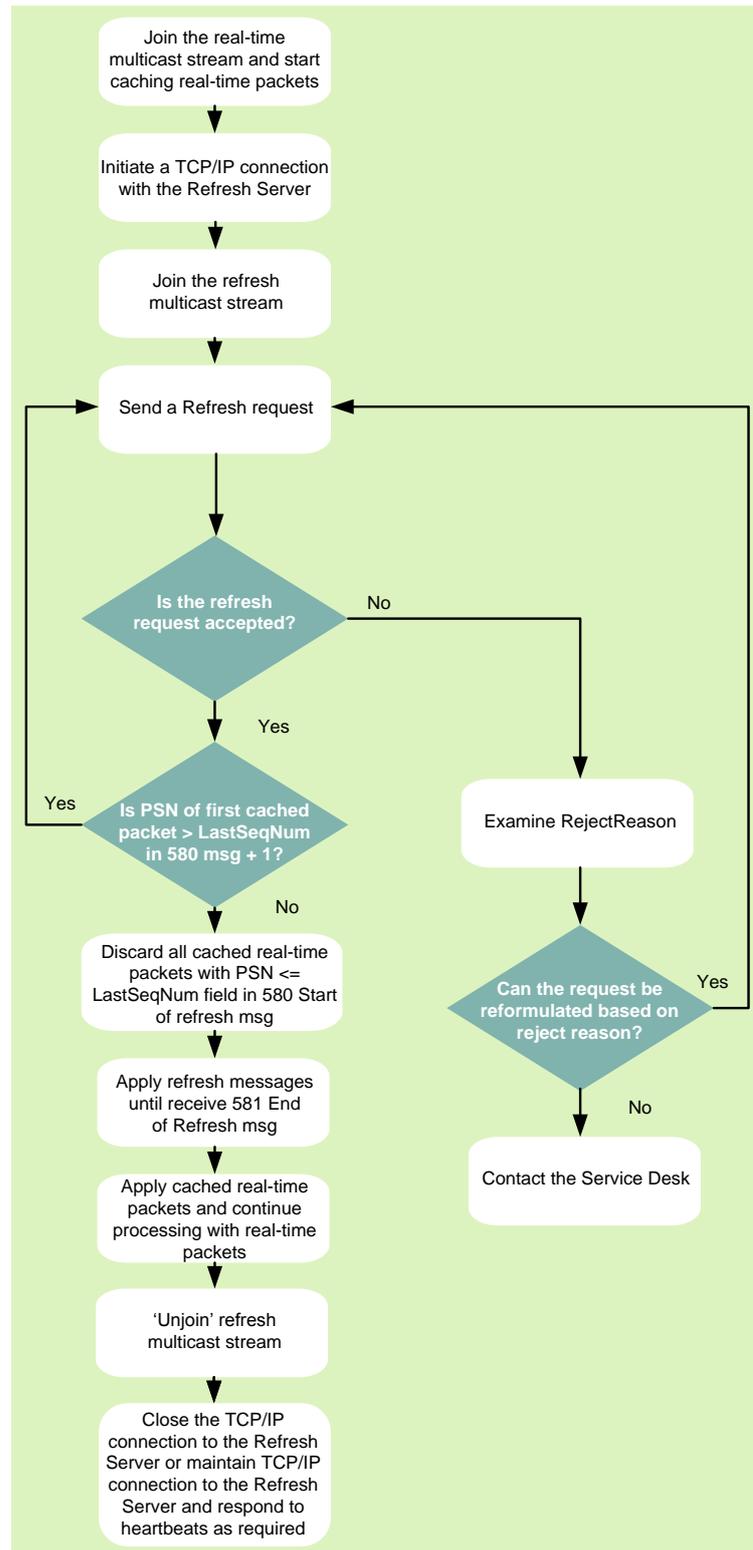
1.4   OPERATIONAL INFORMATION

Measures are in place to safeguard against unexpected system failures.

1.4.1   Exchange System Failure

1.4.1.1   Dual Multicast Lines

Under normal operating conditions, the exchange system will send real-time messages to two unique multicast addresses. This provides clients with two redundant data feeds. The client application should be designed to handle the loss of one of the two multicast channels without any interruption to service.

1.4.1.2   High Availability

The High Availability (HA) functionality of the market data publisher is set up to ensure that there is no loss of service for clients if there is any kind of outage in the exchange on the primary publisher, for example a hardware failure. The failover to the secondary publisher will occur without any gap in market data Packet Sequence Numbers. The HA failover has been designed to be as transparent as possible for clients, as the connectivity in terms of multicast groups and ports will not change. However, clients should note that there are specific technical details that should be considered.

For details of retransmissions and refresh behaviour that should be included as part of application logic, see the high availability Retransmission and Refresh Behaviour sections in Production Feed Configuration and External User Acceptance Feed Configuration.

1.4.1.3   Disaster Recovery Site

In order to mitigate any serious outage in the primary data centre, a secondary data centre is online in standby mode, in case of a serious incident.

Clients should ensure that all configuration surrounding the secondary data centre is included, as described in Production Feed Configuration.

1.4.2   Client System Failure

Real-time market data will be made available on two different multicast groups. This offers clients the possibility to set up more than one receiving system processing the same data. In the event of a client system failure, the backup client system should continue to process the real-time data sent on the second multicast group.

1.4.3   Gap Detection

The XDP feed provides a unique, sequential packet sequence number for each multicast channel. This will allow recipients to identify 'gaps' in the message sequence and, if appropriate, reconcile them 'locally' with an alternate channel or request retransmission of the missing/corrupted data packet. See Gap Detection for more details.

1.5   GENERAL PROCESSING NOTES

1.5.1   Environments

This specification provides information on how to develop applications to subscribe to Euronext Cash market data. A number of environments are available for clients, all of which have various data products separated into different multicast channels. The environments available, and the different multicast groups available in each (known as Service IDs) are defined in **Table 3 Production Environment** and

Table 4 External User Acceptance Environment.

Table 3 Production Environment

SERVICE ID	MULTICAST CONTENT
'101'	Euronext Equities – Reference Data
'102'	Euronext Equities – Trades
'103'	Euronext Equities – Quotes
'104'	Euronext Equities – Orders
'105'	Euronext Warrants – Trades
'106'	Euronext Warrants – Quotes
'107'	Euronext Indices –Values
'108'	European Stocks (Off Exchange Trade Reporting) – Reference Data
'109'	European Stocks (Off Exchange Trade Reporting) – Trade Reporting
'110'	Luxembourg Stock Exchange
'301'	Service A (Orders and Trades)
'302'	Service B (Orders and Trades)
'303'	Service C (Orders and Trades)

Table 4 External User Acceptance Environment

SERVICE ID	MULTICAST CONTENT
'1'	Euronext Cash Markets – Equities Orders (Production 104)
'2'	Euronext Cash Markets – Reference Data + Equities / Indices / Luxembourg (Production 101, 102, 103, 107, 108, 109, 110)
'3'	Euronext Cash Markets – Warrants (Production 105, 106)
'501'	Service A (Orders and Trades)
'502'	Service B (Orders and Trades)
'503'	Service C (Orders and Trades)

Channel definitions for each of the above can be found in Production Feed Configuration and External User Acceptance Feed Configuration.

1.5.2 Multicast Streams

Dual multicast streams are made available for the distribution of real-time and refresh data.

A multicast stream refers to the data available from a given IP address and port number. Users should refer to Production Feed Configuration and External User Acceptance Feed Configuration for information on what data is carried in each multicast group.

Clients should connect to multicast stream(s) for which they require data. In order to complete line arbitrage, clients should connect to both multicast streams with the same data.

1.5.3   TCP/IP Channels

TCP/IP channels are made available for retransmission and refresh requests and responses.

The user can choose to disconnect/reconnect in between requests. However if choosing to remain connected, the user will need to respond to heartbeat requests from the exchange.

1.5.4   Date and Time Conventions

Dates and Times use UTC (Universal Time, Coordinated).

The base for timestamps in packet headers is the number of milliseconds since the previous Sunday 00:00:00.000 UTC (so in the night from Saturday to Sunday).

The base for timestamps in Message bodies is the number of milliseconds since previous midnight 00:00:00.000 UTC.

For example Wednesday 15:30:00.000 UTC is indicated as 315000000 in a packet header or 55800000 in a message body.

1.5.5   Sequence Numbers

The feed contains two sequence numbers:

■ The Packet Sequence Number is part of the packet header, and should be used for retransmission requests. It is unique per service and common across a pair of dual multicast streams. Note that the Packet Sequence Number is only unique for market data packets; heartbeats use the PSN of the last packet.

■ The source sequence number is assigned by the source system to this message. Whilst this sequence number increases serially, it does not increase one by one.

1.5.6   Price Formats

Prices in the feed are represented by two fields, an integer value and a scale code. The scale code is represented in the PriceScaleCode field.

The value should be calculated using the following formula:

$$Value = \frac{Integer}{10^{ScaleCode}}$$

For example, a price of 27.56 is represented by an Integer of 2756 and a PriceScaleCode of 2.

1.5.7   Data Types

All "Binary Int." formatted fields are numeric unsigned binary. All "Binary Int. (signed)" formatted fields are signed binary integer. Binary data is in network byte order (Big-Endian).

All "ASCII Str." and "ASCII Ch." fields are alphanumeric, left aligned and null padded.

1.5.8   Instrument Identifiers

An instrument is identified by its SymbolIndex, across all Service IDs that relate to that instrument. The SymbolIndex is arbitrarily assigned by the feed, and will not change for the lifetime of the instrument.

The SymbolIndex can take a different value for the same instrument depending on the environment (Production or Test).

Standard security identifiers (for example ISIN, Euronext Trading Code) can be found in the 553 Reference Data message (see Reference Data – 553 Message). The symbol mnemonic, ISIN code, and so forth, are carried *only* in the 553 Reference Data message, so the client application must associate the SymbolIndex with the instrument's characteristics during the reference data processing.

1.5.9   How to Determine a Closing Price

The last not–cancelled trade price becomes the closing price, when the state of the instrument's trading group is in the 'end of day inquiry / closed' state.

If no trades took place during the day, the Last Adjusted Closing Price can be used as the closing price. Last Adjusted Closing Price is supplied every morning as part of the reference data. It is the previous day's last trade price, adjusted for corporate events (if applicable).

2.   TECHNICAL MESSAGE SPECIFICATIONS

2.1   INTRODUCTION

There are two types of messages transmitted as part of this protocol: technical and market data. Technical messages do not contain data, they allow conversing parties to exchange session-specific information (for example, 'reset Packet Sequence Number'). Market data messages are product specific.

2.2   PACKET HEADER FORMAT

All messages will contain a common packet header. **Table 5 Packet Header Format** describes the header fields. The design is intended to minimize the development burden on behalf of clients. This means that all clients may implement line-level protocol processing once, and then only need develop parsing algorithms for their choice of message.

Table 5 Packet Header Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-byte packet header.
PacketType	2	2	Binary Int.	Identifier for the type of data contained in the packet: '501' – Market Data Packet '1' – Packet Sequence Number Reset '2' – Heartbeat Packet '10' – Retransmission Response Packet '20' – Retransmission Request Packet '22' – Refresh Request Packet '23' – Refresh Response Packet '24' – Heartbeat Response Packet
PacketSeqNum	4	4	Binary Int.	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and is reset to 1 at the beginning of each trading day. The PacketSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00:00.000 UTC.

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream. Possible values are described in Production Feed Configuration and External User Acceptance Feed Configuration.
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method: ■ '0' – Real-Time message (uncompressed) ■ '2' – Retransmission message (uncompressed) ■ '17' – Refresh message (zlib compressed)
NumberMsgEntries	15	1	Binary Int.	The number of messages that are contained within the packet.

2.3   PACKET SEQUENCE NUMBER RESET

This packet is sent to 'reset' the Packet Sequence Number at start of day, in response to failures, and so forth. Note that this packet will contain a valid sequence number. The message format is shown in **Table 6 Packet Sequence Number Reset Message Format**.

Table 6 Packet Sequence Number Reset Message Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
NextSeqNumber	0	4	Binary Int.	Contains the Packet Sequence Number value that the client should expect in the immediately succeeding data packet. Note that this packet will contain its own valid Packet Sequence Number in the header portion of the message.

2.3.1   Packet Sequence Number Reset Processing Notes

Packet Sequence Numbers normally begin at one (1) and increase one-by-one with each subsequent packet. There are two scenarios where the Packet Sequence Number is reset (besides the start of day):

■ If the value should exceed the maximum value that the SeqNum field may contain, it will be reset to one (1).

■ If the system fails and it recovers, it sends a Packet Sequence Number reset message. The PacketSeqNum field of that packet will be set to one (1) and the NextSeqNumber field will be set to two (2).

Note that the Packet Sequence Number reset is always sent as the first message of the day.

2.4   HEARTBEAT

Heartbeat packets are sent in the multicast streams as well as in the active TCP/IP retransmission and refresh sessions.

2.4.1   General Heartbeat Processing Notes (TCP and Multicast)

■ This applies to the TCP channels for retransmissions and refresh, and also the multicast channels for real-time and refresh data.

■ Heartbeat packets will contain only the packet header (with PacketType = '2'). The packet will not contain a message body.

■ Heartbeats will be sent only on the multicast channels when there is no market data. Heartbeat frequency since the last packet, is:

– 2 seconds in the multicast streams
– 30 seconds in the active TCP/IP retransmission and refresh sessions

2.4.2   Retransmission and Refresh Heartbeat Processing Notes (TCP)

■ Clients may receive a heartbeat packet if they have an active TCP/IP session with the Retransmission or Refresh Server.

■ To determine the health of the user connection on the TCP/IP channel, the Retransmission or Refresh Server will send regular heartbeat packets to the user. The heartbeat frequency is 30 seconds The time out for this heartbeat response packet is set at 5 seconds. If no response is received by the server within this timeframe, the TCP/IP session will be disconnected.

■ Clients that choose to establish and remain connected to the Retransmission or Refresh Server intraday must respond to a heartbeat packet with a heartbeat response packet. Clients can choose to either disconnect following each retransmission or refresh request, or remain connected to the Retransmission or Refresh Server.

Figure 8 Retransmission Server Heartbeat Message

Figure 9 Refresh Server Heartbeat Response

2.5   HEARTBEAT RESPONSE

Clients that choose to establish and remain connected to the Retransmission Server intraday must respond to a heartbeat message with a heartbeat response message.

The fields in the packet header should be filled as described in **Table 7 Heartbeat Response Message Header Format**:

Table 7 Heartbeat Response Message Header Format

FIELD	DESCRIPTION
PacketLength	36
PacketType	24
PacketSeqNum	Optional
SendTime	Optional
ServiceID	Optional
DeliveryFlag	0
NumberMsgEntries	1 (only 1 heartbeat response message should be sent per packet)

The fields in the message body are described in **Table 8 Heartbeat Response Message Body Format**:

Table 8 Heartbeat Response Message Body Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
SourceID	0	20	ASCII Str.	This field represents the Identifier of the source (client) requesting retransmission. Field is null padded, left aligned.

2.6   RETRANSMISSION REQUEST

This packet is sent by clients requesting missing packets identified by a sequence number gap. Upon receipt of a valid retransmission request packet, the requested packet(s) will be sent. The requested packets(s) have the same packet format and content as the original sent by the system.

Note: The Packet Sequence Reset packet (packet number one – see Packet Sequence Number Reset) is *not* available on the Retransmission Server.

The fields in the packet header should be filled as described in **Table 9 Retransmission Request Message Header Format**:

Table 9 Retransmission Request Message Header Format

FIELD	DESCRIPTION
PacketLength	44
PacketType	20
PacketSeqNum	Optional
SendTime	Optional
ServiceID	Service ID of the broadcast stream corresponding to the request, in other words the stream for which messages need to be recovered by the client.
DeliveryFlag	0
NumberMsgEntries	1 (only 1 retransmission request should be sent per packet)

The fields in the message body are described in **Table 10 Retransmission Request Message Body Format**:

Table 10 Retransmission Request Message Body Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
BeginSeqNum	0	4	Binary Int.	Begin Sequence Number of the requested range of packets to be retransmitted. Note that the Sequence Number refers to the PacketSeqNum in the header. The broadcast stream from which a retransmission is requested has to be stated in the field ServiceID in the Packet header of the RetransmissionRequest packet.
EndSeqNum	4	4	Binary Int.	End Sequence Number of the requested range of packets to be retransmitted. Note that the Sequence Number refers to the PacketSeqNum in the header. The broadcast stream from which a retransmission is requested has to be stated in the field ServiceID in the Packet header of the RetransmissionRequest packet.
SourceID	8	20	ASCII Str.	This field represents the Identifier of the source (client) requesting retransmission. Source-ID is pre-set by the Exchange and subject to validation. Field is null padded, left aligned.

2.7   RETRANSMISSION RESPONSE

This packet will be sent immediately via TCP/IP in response to the client's request for retransmission packets.

The fields in the packet header are described in **Table 11 Retransmission Response Message Header Format**:

Table 11 Retransmission Response Message Header Format

FIELD	DESCRIPTION
PacketLength	44
PacketType	10
PacketSeqNum	Contains the Packet Sequence Number if sent in the retransmission request
SendTime	Ignored in the response
ServiceID	This will be the same value as in the initial request
DeliveryFlag	Ignored in the response
NumberMsgEntries	0

The fields in the message body are described in **Table 12 Retransmission Response Message Body Format**:

Table 12 Retransmission Response Message Body Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
SourceSeqNum	0	4	Binary Int.	This field contains the request Packet Sequence Number assigned by the client. It is used by the client to couple the request with the response packet.
SourceID	4	20	ASCII Str.	This field represents the Identifier of the source (client) requesting retransmission. Field is null padded, left aligned.
Status	24	1	ASCII Str.	Indicates whether the retransmission request was accepted or rejected. Valid values are: ■ 'A' – Accepted ■ 'R' – Rejected

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
RejectReason	25	1	Binary Int.	Indicates the reason for the rejection. Valid values are: ■ '0' – Retransmission Request Packet was accepted ■ '1' – Rejected due to permissions (the ServiceID is not granted for the SourceID or an incorrect Source ID has been used) ■ '2' – Rejected due to invalid sequence range ■ '3' – Rejected due to max sequence range reached (> thresholds) ■ '4' – Rejected due to max request reached in a day (> thresholds) ■ '5' – Rejected – Requested packets are no longer available ■ '6' – Rejected – Retransmission request incorrectly formatted ■ '7' – Rejected – Due to incorrect ServiceID
Filler	26	2	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

2.8   REFRESH REQUEST

This packet is sent by clients requesting a refresh. The system will provide the appropriate packet(s) in response.

The fields in the standard packet header should be filled as described in **Table 13 Refresh Request Message Header Format**:

Table 13 Refresh Request Message Header Format

FIELD	DESCRIPTION
PacketLength	36
PacketType	22
PacketSeqNum	Optional
SendTime	Optional
ServiceID	Service ID of the broadcast stream corresponding to the request (the stream for which the refresh will be applied)
DeliveryFlag	Ignored in the response
NumberMsgEntries	1

The fields in the message body are as described in **Table 14 Refresh Request Message Body Format**:

Table 14 Refresh Request Message Body Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
SourceID	0	20	ASCII Str.	Identifier of the Source ID requesting the refresh.

Refresh Response

This packet will be sent immediately via TCP/IP in response to the client's request for a refresh.

Table 15 Refresh Response Message Header Format

FIELD	DESCRIPTION
PacketLength	44
PacketType	23
PacketSeqNum	Contains the Packet Sequence Number if sent in the refresh request
SendTime	Ignored in the response
ServiceID	Service ID of the broadcast stream corresponding to the request (the stream for which the refresh will be applied). This will be the same value as in the initial request.
DeliveryFlag	Ignored in the response
NumberMsgEntries	1

The fields in the message body are described in **Table 16 Refresh Response Message Body Format**:

Table 16 Refresh Response Message Body Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
SourceSeqNum	0	4	Binary Int	This field contains the request Packet Sequence Number assigned by the client. It is used by the client to couple the request with the response packet.
SourceID	4	20	ASCII Str.	Identifier of the source requesting the refresh.
Status	24	1	ASCII Str.	Indicates if the Refresh request has been accepted. Valid values are: ■ 'A' – Accepted ■ 'R' – Rejected
RejectReason	25	1	Binary Int.	Indicates the reason for rejection. Valid values are: ■ '0' – Refresh Request Packet was accepted

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ '1' – Request rejected due to permissions – an incorrect Source ID has been used
				■ '2' – Request rejected due to incorrect ServiceID
				■ '3' – Refresh request incorrectly formatted
				■ '4' – Request rejected due to incorrect packet type sent
				■ '5' – Rejected due to max request reached in a day (> thresholds)
				■ '6' – Request rejected due to unavailability of refresh data, for example:
				Requesting reference data refresh before the real-time reference data has been fully broadcast
				Requesting order book refresh before the morning market sheet retransmission
				■ '7' – Refresh request rejected as sent to incorrect server (secondary instead of primary)
Filler	26	2	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

2.9   REFRESH MESSAGES

2.9.1   Refresh Compression

The refresh messages disseminated over the refresh multicast channels are compressed using the zlib compression format. As mentioned before, this method of compression considerably reduces network latency. The complete specifications for the zlib compression format can be found here: http://www.zlib.net/zlib_docs.html.

Clients should use the DeliveryFlag field in the packet header to determine if a packet is zlib compressed or not.

All messages delivered over the refresh multicast channels will be compressed. The messages delivered over the real-time multicast channels will not be compressed.

2.9.2   Refresh Packet Type

Table 17 Refresh Packet Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-byte packet header.
PacketType	2	2	Binary	Identifier for the type of data contained in the

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	packet. '501' – Market Data Packet
PacketSeqNum	4	4	Binary Int.	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and monotonically and is reset to 1 at the beginning of each trading day. The PackSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00 UTC.
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream. Possible values are described in Production Feed Configuration and External User Acceptance Feed Configuration.
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method: ■ '0' – Real-Time message (uncompressed) ■ '2' – Retransmission message (uncompressed) ■ '17' – Refresh message (zlib compressed)
NumberMsgEntries	15	1	Binary Int.	The number of messages that are contained within the packet.

2.9.3   Start Refresh – 580 Message

A refresh cycle begins with a Start Refresh message and ends with an End Refresh message on the multicast channels.

Table 18 Start Refresh Message Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'580' – Start Refresh Message
LastSeqNum	4	4	Binary Int.	Contains the last cached PacketSeqNum that the Refresh is valid to.

2.9.4   End Refresh – 581 Message

A refresh cycle begins with a Start Refresh message and ends with an End Refresh message on the multicast channels.

Table 19 End Refresh Message Format

FIELD	OFFSET (OFFSET)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'581' – End Refresh Message
LastSeqNum	4	4	Binary Int.	Contains the last cached PacketSeqNum that the Refresh is valid to.

3. REAL-TIME MESSAGE SPECIFICATIONS

3.1 MARKET INFORMATION

3.1.1 Overview

The Market Information message types are:

- 505 – Stock state change message

- 513 – Euro & Interbank rates

- 516 – Class state change

- 523 – Mail

- 530 – Indicative Matching Price

- 534 – Authorized out of session limits

- 535 – TCS (Trade Confirmation System) State change

- 537 – Collars

- 539 – Session Timetable

- 550 – Start Reference Data

- 551 – End Reference Data

- 553 – Reference Data

3.1.2 Packet Header Format

All messages are preceded by a common packet header format. **Table 20 Packet Header Format** describes the header fields of a Market Information message.

Table 20 Packet Header Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-byte packet header.
PacketType	2	2	Binary Int.	Identifier for the type of data contained in the packet. '501' – Market Data Packet
PacketSeqNum	4	4	Binary Int.	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and monotonically and is reset to 1 at the beginning of each trading day. The PackSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00 UTC.
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream. Possible values are described in Feed Configuration descriptions.
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method: ■ '0' – Real-Time message (Uncompressed) ■ '2' – Retransmission message (Uncompressed) ■ '17' – Refresh message (zlib Compressed)
NumberMsgEntries	15	1	Binary Int.	The number of messages that are contained within the packet.

3.1.3   Stock State Change – 505 Message

3.1.3.1   Message Overview

A 'Stock State Change' message is generated by the trading engines to announce a change in the state of an instrument.

3.1.3.2   Message Sending Rules

This is sent for tradable instruments:

■ When an instrument has been halted

■ When an instrument has been unhalted

■ When an opening or an opening auction is programmed

■ When a programmed opening or opening auction is cancelled

■ Prior to the trading session to carry over instruments halted the day before

Interaction between 505 and 516 messages

There are three rules that should be considered when processing the 505 message (Stock State Change) in combination with the 516 message (Class State Change). These are as follows:

1. When a 516 message is received, the state indicated by the "ClassState" field should be applied to every instrument in the instrument group represented by the message ("InstrumentGroupCode" field).

2. When a 505 message is received, the states indicated and the order entry permissions should be applied only to the instrument represented by the Symbol Index in the message. For example, if instrument XYZ is in continuous trading (COCO) but is halted by a 505 message, only that instrument is halted and the rest of the instruments in the same trading group remain unaffected.

3. When processing any combination of 505 and 516 messages, the most restrictive behaviour should take precedence. For example, if there is conflict between a 516 message with state TAL and a 505 message with a Halted State and OrderEntryRejection=Y, the 505 message should take precedence.

Consistency between class phases and instrument status

The following two tables (**Table 21** and **Table 22**) show, for each class market phase, whether order entry is allowed, the order entry state and the processing type occurring (processing types are further explained in the third table below (**Table 23 Processing Types**)). The tables show the various state combinations that an instrument can have. In general, an instrument inherits the behaviour of its class unless it is overridden with a Stock State Change message. Note the following:

■ "Class Market Phase" refers to the phase of the market that relates to classes or groups of instruments. There are different actions and messages that relate to each of the groups and individual stocks. For example, an entire group can be put into the Call phase, but an individual stock can only be halted, inherit from group and so forth.

■ "Other Class Status" refers to states that both stocks and classes/groups can have. See **Table 24 Class State Change Message Format**, in particular the field "Class State", to see a list of market phases that apply to entire instrument groups. Contrast this with **Table 25 Stock State Change Message Format,** particularly the fields "InstrumentTradingStatus" and "InstrumentState".

■ The class states link to **Table 25 Class States**; the "Phase" column in that table relates to the three main phases of the trading day:

Early – The phase before any market interaction occurs

Core – The phase during which market interaction occurs

Late – The phase after market interaction occurs

■ In the following tables, a hyphen (-) indicates that the two states cannot coexist. For example, there will never be an Auction during monitoring periods.

For more information about market phases, refer to the Euronext Instruction n° 4-01 Universal Trading Platform Trading Manual and the Appendix to Instruction n° 4-01 the Universal Trading Platform Trading Manual.

Table 21 Instrument State Details (Part 1)

INSTRUMENT STATE	CLASS MARKET PHASE							
	CALL		AUCTION		CONTINUOUS TRADING		TRADING AT LAST (TAL)	
	ORDER ENTRY STATE	PROCESSING TYPE	ORDER ENTRY STATE	PROCESSING TYPE	ORDER ENTRY STATE	PROCESSING TYPE	ORDER ENTRY STATE	PROCESSING TYPE
Inherited	Yes	IP	Queued	Selective	Yes	Match	Yes	Match*
Halted	Yes	IP	Queued	Selective	Yes	IP	Yes	IP
Auction	-	-	-	-	Queued	Selective	-	-

* Matching is against the closing price of the closing auction.

Table 22 Instrument State Details (Part 2)

OTHER CLASS STATUS			
MONITORING		HALTED	
ORDER ENTRY STATE	PROCESSING TYPE	ORDER ENTRY STATE	PROCESSING TYPE

OTHER CLASS STATUS			
MONITORING		**HALTED**	
ORDER ENTRY STATE	**PROCESSING TYPE**	**ORDER ENTRY STATE**	**PROCESSING TYPE**
No	-	Yes	IP
No	-	Yes	IP
-	-	-	-

Table 23 Processing Types explains each of the order processing types.

Table 23 Processing Types

PROCESSING TYPE	DESCRIPTION
IP Processing	Call Type (order accumulation with IP (indicative price) calculation and without any matching)
Match Processing	Standard continuous price time priority matching
Selective Processing	Fixed price matching
–	Not Applicable

Table 24 POSSIBLE VALUES FOR WARRANTS AND CERTIFICATES FOR TRADING AFTER KNOCK OUT (TAKO) AND ENHANCEMENT OF KNOCK-IN and KNOCK-OUT MANAGEMENT.

The below table details possible events and the expected values:

EVENT ON THE INSTRUMENT	Instrument State	Instrument Trading Status	Halt Reason	Action Affecting State
Knock-out by Market Operations	H	S	K	E
Knock-out by Issuer (KOBI)	H	S	K	I
Beginning of a TAKO phase (triggered by LP during the day, happens only once)	H	S	R	T
LP present in TAKO phase	*space*	*space*	0	O
UTP initialization, for instrument which are in TAKO phase	*current instrument state*	*current instrument state*	0	B

3.1.3.3

3.1.3.4   Message Structure

Table 25 Stock State Change Message Format describes the body fields of an Equities Market Information message, MsgType = '505' Stock State Change.

Table 25 Stock State Change Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'505' – Stock State Change
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SystemID	16	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	20	2	Binary Int.	Number of microseconds to be combined with SourceTime.
Filler	22	2	Binary Int.	Filler to retain the structure of the message – possibly for future use.
StartDateHalting	24	8	ASCII Str.	Date that suspension of trading started for an instrument. Valid values are: ■ YYYYMMDD ■ 00000000 if not provided
StartTimeHalting	32	6	ASCII Str.	Time that suspension of trading started for an instrument. Valid values are: ■ HHMMSS ■ 000000 if not provided
ProgOpeningTime	38	6	ASCII Str.	Instrument opening time that has been programmed by the Cash Market Operations. Valid values are: ■ HHMMSS ■ 000000 if not provided.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
OrderEntryRejection	44	1	ASCII CH.	Indicates whether order entry is allowed or forbidden. Valid values are: ■ 'N' – Order entry allowed ■ 'Y' – Order entry forbidden
InstrumentState	45	1	ASCII Ch.	Indicates the state of the instrument. Valid values are: ■ '0' – Not applicable ■ 'A' – Auction ■ 'H' – Halted ■ Null (or space) – Inherited (following the state of the class the instrument belongs to) Remark: There is a relation between **ClassState (Msg 516)** and InstrumentState: in case the state of the Group/Class to which the instrument belongs differs from the state of that individual instrument, the most restrictive value supersedes: If the Stock state shows "tradable" but its class/group is declared "not tradable" (for example, Interrupted, Forbidden), effectively trading is not allowed in this instrument.
InstrumentTradingStatus	46	1	ASCII Ch.	Indicates whether trading on an instrument is suspended, halted or resumed. Valid values are: ■ Null – Not provided ■ 'S' – Suspended For Indices: ■ 'S' – Calculation Suspended ■ 'R' – Calculation Resumed
HaltReason	47	1	ASCII Ch.	Indicates the origin of halting for an instrument. Valid values are: ■ '0' – Not applicable ■ 'R' – Halted. No liquidity provider ■ 'C' – Opening or trade price outside

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				dynamic collars
				■ 'M' – Manual halting by Market Operations
				■ 'K' – Knock-out
				■ Null or space – Instrument not halted or information not available
ActionAffectingState	48	1	ASCII Ch.	Code indicating the event that caused a change in the stock state. Valid values are:
				■ '0' – Not applicable
				■ 'B' – Trading after knock-out period (beginning of the trading day)
				■ 'C' – Trading on the instrument at the opening (sent before O)
				■ 'D' – Cancelled programmed opening
				■ 'M' – Instrument manually halted by Market Operations
				■ 'E' – Instrument halted by Market Operations on knock-out
				■ 'I' – Instrument halted by Issuer/LP on knock-out
				■ 'N' – Instrument is being initialized (beginning of the trading day)
				■ 'O' – Instrument opened
				■ 'P' – Deferred programmed opening
				■ 'R' – Automatic halting at the class auction'Y' – Beginning of a one side only period
				■ 'T' – Beginning of a trading after knock-out period
				■ Z' – End of a one side only period
				■ Null or space – Indicates that the 'Order Entry' flag on an instrument has just changed.
InstrumentStateTCS	49	1	ASCII Ch.	Indicates if trade declaration/reporting is authorized in TCS. Valid values are:
				■ '0' – Not applicable
				■ 'S' – Instrument is forbidden

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ 'A' – Instrument is authorized ■ Null – Not significant
PeriodSide	50	1	ASCII Ch.	This indicator indicates the side phase for an instrument being in a Bid only situation or Offer only situation. Valid values are: ■ '0' – Not applicable ■ 'A' – Bid only period ■ 'V' – Offer only period ■ Null – Not significant
Filler	51	1	Binary Int.	Filler to retain the structure of the message – possibly for future use.

3.1.4   Class State Change – 516 Message

3.1.4.1   Message Overview

The Class State Change message indicates a change in the state of a class / instrument group.

There is a relationship between the Class State (516 message) and the Stock State (505 message). In case the state of the Class to which the instrument belongs differs from the state of that individual instrument, the most restrictive value supersedes the other. If the Stock State shows "tradable" but its Class is declared "not tradable" (for example, Interrupted, Forbidden), effectively trading is not allowed in this instrument.

3.1.4.2   Message Sending Rules

This message is sent each time that a class / instrument group changes state during the trading day.

3.1.4.3   Message Structure

Table 24 Class State Change Message Format describes the body fields of an Equities Market Information message, MsgType = '516' Class State Change.

Table 24 Class State Change Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'516' – Class State Change.
SourceSeqNum	4	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	8	4	Binary	This field specifies the message generation time.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SystemID	12	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	16	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
InstrumentGroupCode	18	3	ASCII Str.	Instrument Group / Class Identification.
Filler	21	1	ASCII Ch.	Filler to retain the structure of the message – possibly for future use.
SessionType	22	1	ASCII Ch.	Market Session. Valid values are: ■ 'E' – Early ■ 'C' – Core Session ■ 'L' – Late
Filler	23	1	Binary Int.	Filler to retain the structure of the message – possibly for future use.
ClassState	24	4	ASCII CH.	Class status representing the current trading market phase for instruments belonging to that class and whose status is inherited. Valid values are: ■ 'EAMO' – Early Monitoring ■ 'COCA' – Core Call ■ 'COAU' – Core Auction ■ 'COCO' – Core Continuous ■ 'CLCA' – Closing Call ■ 'CLAU' – Closing Auction ■ 'TAL' – Trading At Last ■ 'COMO' – Core Monitoring ■ 'LAMO' – Late Monitoring ■ 'CLSD' – Closed ■ 'HALT' – Halted

Table 25 Class States provides details of each of the Class States, including whether order entry is allowed and if order matching occurs.

Note: In the following table, a hyphen (-) indicates that the combination does not exist. The market phase dictates what interaction is allowed; for example there is an "OrderEntryRejection" field in the Stock State Change 505 Message (see **Table**) that indicates if order entry is allowed during the period when the stock is halted. The following table shows the *default* behaviours however these can be overridden, in which case the most restrictive action takes precedence.

Table 25 Class States

PHASE	STATE	ORDER ENTRY	ORDER MATCHING	CLASS STATE
Early	Monitoring	N	N	EAMO
Core	Call	Y	N	COCA
	Auction	N	Y	COAU
	Continuous	Y	Y	COCO
	Call	Y	N	CLCA
	Auction	N	Y	CLAU
	Trading at last	Y	Y	TAL
	Monitoring	-	N	COMO
Late	Monitoring	-	N	LAMO

Note: There are two additional class states:

- HALT – This state can occur at any of the other phases, as per the Euronext Instruction n° 4-01 Universal Trading Platform Trading Manual and the Appendix to Instruction n° 4-01 the Universal Trading Platform Trading Manual.
- CLSD – This state indicates that the entire trading environment is terminating.

3.1.5   Mail – 523 Message

3.1.5.1   Message Overview

This message supplies a part or a complete electronic message (E-mail), in order to inform members and data vendors about trading conditions on a given listed instrument or a set of listed instruments or otherwise about a technical problem.

3.1.5.2   Message Sending Rules

This message is sent manually by Cash Market Operations to inform member firms about events of general interest that occurred in the market (suspension of securities, deletions of order books, new listings of securities, various technical messages, and so forth).

Note: A long mail message can be split into several pages. Information in the mail header enables users to rebuild the entire mail message.

3.1.5.3   Message Structure

Table 26 Mail Message Format describes the body fields of a Equities Market Information message, MsgType = '523' Mail.

Table 26 Mail Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'523' – Mail
SourceSeqNum	4	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	8	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day.\n\nFor example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170
SystemID	12	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	16	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
EMailMsgNumber	18	2	Binary Int.	Mail number. Helps in rebuilding a mail message that was sent in several messages.
NumberOfMessage	20	2	Binary Int.	Number of messages making up the mail message. Helps in rebuilding a mail message that was sent in several transmission (between 1 to 20).
SequenceNumber	22	2	Binary Int.	Current sequence number message.
InstrumentGroupCode	24	3	ASCII Str.	Instrument Group of Identification.
Filler	27	1	ASCII Str	Filler to retain the structure of the message – possibly for future use.
PriorityIndicator	28	1	ASCII Ch.	Urgency of the mail. Valid values are:\n\n■ 'O' – Urgent message\n\n■ 'N' – Non urgent message

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
TypeOfMessage	29	1	ASCII Ch.	General contents of the message. Valid values are: ■ 'B' – Market Information contents ■ 'T' – Technical Information ■ 'R' – Market and Technical information
AddressType	30	2	ASCII Str.	Indicates the users to whom the message is addressed.
Title	32	80	ASCII Str	Title of the mail.
Text	112	854	ASCII Str.	Text, line break (@), page break (#).
Filler	966	2	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

3.1.6   Request For Size – 524 message

3.1.6.1   Message overview

The Request For Size message is sent upon validation of an analoguous member Request For Size message by UTP. Once Liquidity Providers receive the Request For Size notification message from XDP, they are free to respond by sending new orders to the trading engine.

3.1.6.2   Message Sending Rules

When a member sends a Request For Size message, UTP carries out several consistency and functional checks on the incoming message. If the member message passes all the consistency and functional checks, UTP attributes an RFS ID to it and the 524 message is diffused by XDP.

3.1.6.3   Message Structure

Table 27 Request For Size Message format describes the body fields of an Equities Market Information message, MsgType = '524' Request For Size message.

Table 28 Request For Size Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'524' – Request For Size
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SystemID	16	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
RFSID	20	4	Binary Int.	Unique identifier generated by UTP upon validation of the member RFS message
Volume	24	4	Binary Int.	Quantity requested in message
SourceTimeMicroSecs	28	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
Side	30	1	ASCII Ch	Order side: 0 for None (Binary Zero) '1' for Buy '2' for Sell
Filler	31	1	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

3.1.7   Indicative Matching Price – 530 Message

3.1.7.1   Message Overview

The Indicative Matching Price message indicatesthe instrument's theoretical opening conditions which consists of:

- the Indicative Matching Price (IMP): price at which the instrument would trade if it opened at the moment the price is calculated

- the Indicative Matching Volume (IMV): quantity that would trade at the IMP if the instrument opened at the moment the price is calculated

- the indicative imbalance volume: remaining unmatched quantity at the IMP

- The indicative imbalance volume side: side of the indicative imbalance volume

An Indicative Matching Price message is sent if at least one of the instrument's theoretical opening conditions changes: (indicative matching price or indicative matching volume or imbalance volume or imbalance volume side varies).

■ If the Indicative Matching Price remains undetermined, but the reason for this indetermination changes, then an Indicative Matching Price message is sent with zero values.

3.1.7.2   Message Sending Rules

■ On the UTP trading engine, the Indicative Matching Price is calculated in a preset frequency (set at 50 msec) and sent only when at least one of the opening conditions is modified.

3.1.7.3   Message Structure

Table 29 Indicative Matching Price Message Format describes the body fields of an Equities Market Information message, MsgType = '530' Indicative Matching Price.

Table 29 Indicative Matching Price Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'530' – Indicative Matching Price.
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
IMPrice	16	4	Binary Int.	Indicative Matching Price: Price at which transaction on a security would occur if Opening occurred at this moment (to be calculated with the PriceScaleCode).
Variation	20	4	Binary Int. (signed)	Percentage variation of the last indicative matching price against the previous day's reference price for the stock concerned, i.e. the last known price can be the last adjusted closing price or the last price indication (if an indication was entered after the last traded price). (To be calculated with the VariationScaleCode).
SystemID	24	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	28	2	Binary Int.	Number of microseconds. To be combined with SourceTime.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PriceScaleCode	30	1	Binary Int.	To be combined with IMPrice.
VariationScaleCode	31	1	Binary Int.	To be combined with Variation.
IMVolume	32	4	Binary Int.	Indicative Matching Volume: Volume that would be exchanged if Auction occurred at this moment. This volume includes hidden quantities.
ImbalanceVolumeSide	36	1	ASCII Ch.	Side of the imbalance volume if the Auction occurred at this moment. '1' for Buy '2' for Sell Blank if no imbalance
Filler	37	3	ASCII Str.	Filler for data alignment.
ImbalanceVolume	40	4	Binary Int.	Imbalance volume quantities if Auction occurred at this moment. This volume includes hidden quantities.

3.1.8   Authorized Out of Session Limits – 534 Message

3.1.8.1   Message Overview

The message Authorized Out of Session Limits (outside the Trading Session) carries the authorized limits for trades executed outside the trading session for a given instrument.

These limits are calculated by the Trade Confirmation System (TCS) reporting engine according to the last price and possibly the Weighted Average Spread for the instrument.

This message is *not* sent for the following types of instruments:

■ Instruments listed on the OTC market

■ Instruments that have been delisted from the regulated markets

■ Instruments listed on the Alternext market (for trades outside the central order book on these instruments, neither the price nor the traded amount is checked by TCS)

■ Instruments belonging to the Dutch Fund category

■ Exchange-Traded Funds (ETFs) – also called "trackers" in Euronext terminology

3.1.8.2   Message Sending Rules

This message is sent:

■ Every morning for all securities submitted to a price control using the TCS operation type "off-market"

■ If the security is traded on a continuous basis, the message is sent when the security's class group enters the Late Monitoring Phase.

■ If the security is traded by auction, the message is sent at each auction for that security

■ By manual intervention by Cash Market Operations

3.1.8.3   Message Structure

Table 30 Authorized Out of Session Limits Message Format describes the body fields of an Equities Market Information message, MsgType = '534' Authorized Out Of Session Limits.

Table 30 Authorized Out of Session Limits Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'534' – Authorized Out Of Session Limits.
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
MinPriceTrades	16	4	Binary Int.	Minimum price for normal out of session trades (to be calculated with the TradeScaleCode).
MaxPriceTrades	20	4	Binary Int.	Maximum price for normal out of session trades (to be calculated with the TradeScaleCode).
MinPriceBlockTrades	24	4	Binary Int.	Minimum price for out of session block trades (to be calculated with the BlockTradeScaleCode).
MaxPriceBlockTrades	28	4	Binary Int.	Maximum price for out of session block trades (to be calculated with the BlockTradeScaleCode).
SystemID	32	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	36	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
TradeScaleCode	38	1	Binary Int.	To be combined with MinPriceTrades and MaxPriceTrades
BlockTradeScaleCode	39	1	Binary Int.	To be combined with MinPriceBlockTrades and MaxPriceBlockTrades

3.1.9   TCS State Change – 535 Message

3.1.9.1   Message Overview

A TCS state change is sent by the TCS reporting engine to announce the next trading day. It makes it possible to determine the trading date and the related indicative Net Asset Value (iNAV) for valuing the Dutch Funds trades.

3.1.9.2   Message Sending Rules

This message is sent:

■ For a daily auction, once a day for an Investment fund instrument group only in order to indicate the change of a trading day.

■ For a weekly, monthly or quarterly execution cycle, only once during the concerned cycle for an Investment funds instrument group in order to indicate the change of a trading cycle.

3.1.9.3   Message Structure

Table 31 TCS State Change Message Format describes the body fields of an Equities Market Information message, MsgType = '535' TCS State Change.

Table 31 TCS State Change Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'535' – TCS State Change.
SourceSeqNum	4	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	8	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SystemID	12	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	16	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
InstrumentGroupCode	18	3	ASCII Str.	Instrument Group concerned by the change of trading cycle.
ChangeTradingCycle	21	1	ASCII Ch.	Indicates the cut-off for an investment fund group in order to take into account the trading cycle change. Valid values are: ■ 'C' – Change trading cycle
Filler	22	2	ASCII Str	Filler to retain the structure of the message – possibly for future use.

3.1.10   Collars – 537 Message

3.1.10.1   Message Overview

A Collars message (also referred as Dynamic Thresholds) informs clients of modifications in authorized price intervals for an instrument. This message is subjected to selective transmission according to the class of which the instrument belongs.

3.1.10.2 Message Sending Rules

This message is sent:

- At the start-up of the UTP trading system in the morning

- Each time Cash Market Operations changes a collar for an instrument

- After the first auction of the given instrument

For instruments traded in the new Warrants Market Model:

- By default the 537 message is not sent for instruments traded on the new Warrants market model. The exception to this rule is that the 537 message will be sent when the instrument is:

In a bid-only period (505 message PeriodSide=A)

During the period, each time that the bid price of the instrument changes the high and low collar will be sent. The collar on the bid side will correspond to the price quoted by the Liquidity Provider. The price on the offer side will correspond to a virtual offer price calculated by the trading engine.

3.1.10.3 Message Structure

Table 32 Collars Message Structure describes the body fields of an Equities Market Information message, MsgType = '537' Collars.

Table 32 Collars Message Structure

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'537 – Collars
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
HighCollar	16	4	Binary Int.	Higher bound of the collar price range (to be calculated with the HighScaleCode).
LowCollar	20	4	Binary Int.	Lower bound of the collar price range (to be calculated with the LowScaleCode).
SystemID	24	4	Binary	The ID of the originating Exchange/System of the

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	message. See System ID.
SourceTimeMicroSecs	28	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
HighScaleCode	30	1	Binary Int.	To be combined with HighCollar.
LowScaleCode	31	1	Binary Int.	To be combined with LowCollar.

3.1.11 Session Timetable – 539 Message

3.1.11.1 Message Overview

The session timetable message indicates the timetable for changes in the state of a Class / Instrument group for the current trading day.

3.1.11.2 Message Sending Rules

This message is sent:

■ Automatically for each Class / Instrument group code, before the pre-opening to indicate the times at which the market session will change from one phase to another

■ On an exceptional basis, it may be sent during the trading day in cases where normal hours have changed or there are multiple openings during the day

3.1.11.3 Message Structure

Table 33 Session Timetable Message Format describes the body fields of an Equities Market Information message, MsgType = '539' Session timetable.

Table 33 Session Timetable Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'539 – Session timetable.
SourceSeqNum	4	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	8	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				47576170
SystemID	12	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	16	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
InstrumentGroupCode	18	3	ASCII Str	Indicates for which Class / Instrument Group code the different times are provided.
SessionType	21	1	ASCII Ch	Market Session. Valid values are: ■ 'E' – Early session ■ 'C' – Core session ■ 'L' – Late session
TimePreOpening1	22	6	ASCII Str.	Indicates the pre-opening time of the Class / Instrument Group for the session.
TimeOpening1	28	6	ASCII Str.	Indicates the opening time of the Class / Instrument Group for the session.
TimeClosing1	34	6	ASCII Str.	Indicates the closing time of the Class / Instrument Group for the session.
TimePreOpening2	40	6	ASCII Str.	Indicates, if applicable, the pre-opening time of second auction.
TimeOpening2	46	6	ASCII Str.	Indicates, if applicable, the opening time of second auction.
TimeClosing2	52	6	ASCII Str.	Indicates, if applicable, the closing time of second auction.
TimePreOpening3	58	6	ASCII Str.	Indicates, if applicable, the pre-opening time of third auction.
TimeOpening3	64	6	ASCII Str.	Indicates, if applicable, the opening time of third auction.
TimeClosing3	70	6	ASCII Str.	Indicates, if applicable, the closing time of third auction.
EodTime	76	6	ASCII Str.	Time of end of day (EOD) inquiry.
Filler	82	2	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

3.1.12 Start Reference Data – Message 550

3.1.12.1 Message Overview

The Start Reference Data message is sent to signal the start of transmission of the Instrument Reference Data Flow. It is sent in the morning and in the evening.

3.1.12.2 Message Sending Rules

This message is sent:

■ Each time Reference Data messages (553) are sent, prior to the first 553 message

■ There is a 550 message per multicast group containing 553 messages

3.1.12.3 Message Structure

Table 34 Start Reference Data Message Format describes the body fields of an Equities Market Information message, MsgType = '550' Start Reference Data.

Table 34 Start Reference Data Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'550' – Start Reference Data
Indicator	4	1	ASCII Ch.	This field indicates the start of the Instrument characteristic flow. Always takes the value 'S'.
Filler	5	3	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

3.1.13 End Reference Data – 551 Message

3.1.13.1 Message Overview

The End Reference Data message is sent to signal the end of transmission of the Instrument Reference Data Flow. It is sent in the morning and in the evening.

3.1.13.2 Message Sending Rules

This message is sent:

■ Each time Reference Data messages (553) are sent, after the last one

■ There is a 551 message per multicast group containing 553 messages

3.1.13.3 Message Structure

Table 35 End Reference Data Message Format describes the body fields of an Equities Market Information message, MsgType = '551' End Reference Data.

Table 35 End Reference Data Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'551' – End Reference Data
Indicator	4	1	ASCII Ch.	This field indicates the end of the Instrument characteristic flow. Always takes the value 'E'.
Filler	5	3	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

3.1.14   Reference Data – 553 Message

3.1.14.1 Message Overview

The Reference Data message provides the main characteristics of a listed instrument:

■   Characteristics of the instrument itself

■   Trading characteristics of the instrument

■   Previous trading day closing price and amount of capital traded

For morning messages, the characteristics are valid for the trading day at the start of which it was transmitted.

For evening messages, the characteristics are valid for the next business date (the trading day D+1 following the Day D of its transmission).

Multi-Listing – Market of Reference – Single Order Book

An instrument can be listed on more than one of the exchanges operated by Euronext. Such a Multi-Listing can be identified in the feed by using the field MICLIST in the 553 message.

If an instrument is Multi-Listed, a Market of Reference (MoR) is designated and Euronext's European Single Order Book will consolidate liquidity in such instrument by ensuring that all order flow in that instrument is concentrated on a single trading line in the designated MoR. Companies can decide to be Multi-Listed on more than one of Euronext markets to benefit from increased visibility and exposure.

For example, ING Groep (NL0000303600) (headquartered in the Netherlands) is listed on two of Euronext European exchanges, Euronext Amsterdam (being its Market of Reference) and Euronext Brussels. Even though order flow in ING Groep is concentrated on the single trading line in the designated Market of Reference (being Euronext Amsterdam), ING Groep is still considered a listed company in the Netherlands and Belgium.

The MICLIST will show an instrument being listed on more than one of Euronext's European exchanges and it always starts with the MIC of the MoR.

The Euronext website should be used as Reference for correct display of Multi-Listed instruments; each display of a Multi-Listed instrument should (i) include the relevant exchanges on which the instrument is

listed and (ii) show the real-time quotes of the relevant instrument (based on the single trading line in the designated Market of Reference).

3.1.14.2 Message Sending Rules

This message is sent:

- Every morning – There is a 553 message per instrument, tradable or broadcast the current day

- Every evening – There is a 553 message per instrument, tradable or broadcast the next trading day

- 553 messages will be sent in between the start and end reference data messages (550 and 551)

3.1.14.3 Message Structure

Table 36 Reference Data Message Format describes the body fields of an Equities Market Information message, MsgType = '553' Reference Data.

Table 36 Reference Data Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	553 – Instrument Characteristic
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
LastAdjPrice	16	4	Binary Int.	Last traded price of the previous trading day after application of the adjustment coefficient (to be calculated with the LastAdjPriceScaleCode). Provided only in the message sent in the morning. Not provided for European instruments.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
SystemID	20	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
PrevVolumeTraded	24	4	Binary Int.	Number of shares traded on this security during trading day identified by 'DateOfLastTrade'. Provided only in the message sent in the morning. Not provided for European instruments.
FixPriceTick	28	4	Binary Int.	Indicates the amount of the fixed tick size (according to FixPriceTickScaleCode). Provided only for tradable instruments.
SourceTimeMicroSecs	32	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
StockExchangeCode	34	2	Binary Int.	Indicates the Market Place. For a description of which exchange each of the possible codes refers to, see Stock Exchange Code.
TypeOfInstrument	36	2	Binary Int.	Type of instrument. For a description of what instrument each of the possible codes mean, see Stock Type.
EventDate	38	8	ASCII Str.	Date of the last modification of the characteristics of the instrument except the following ones (which are modified every day): ■ Previous day's adjusted closing price (LastAdjPrice) ■ Previous day capital traded (PrevDayCapitalTraded) ■ Number of shares for this security traded on previous day (PrevVolumeTraded) ■ Date security last traded (DateOfLastTrade)
InstrumentName	46	18	ASCII Str.	Instrument Name

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PeriodIndicator	64	1	ASCII Ch.	Indicates for which day the characteristics of the instrument will be effective. Valid values are: ■ 'M' – Morning, effective for the current day ■ 'E' – Evening, effective for the next trading day
TypeOfMarketAdmission	65	1	ASCII Ch.	Indicates the type of market to which a security has been listed. Valid values are: ■ 'A' – Instruments traded on the primary market ■ 'B' – Instruments traded on the secondary market ■ 'C' – Instruments traded on the New Market ■ 'D' – Non regulated market / instruments traded on the free market ('Marché Libre') ■ 'E' – Non regulated market / Alternext ■ 'F' – Non listed ■ 'G' – Regulated Market / Non equities ■ 'H' – Regulated Market / Equities / Segment A ■ 'I' – Regulated Market / Equities / Segment B ■ 'J' – Regulated Market / Equities / Segment C ■ 'K' – Regulated Market / All securities / Special Segment ■ 'L' – Regulated Market / Equities / Other instruments ■ 'S' – OPCVM, SICOMI non listed (French Investment Funds) ■ '6' – Off Market ■ '7' – Gold, Currencies, and Indices of

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				Euronext ■ '9' – Foreign
IssuingCountryCode	66	3	ASCII Str.	Country code of location for the corporate headquarters of the company that issued the instrument. (ISO 3166-3A)
TradingCurrency	69	3	ASCII Str.	Code of the currency in which the instrument is traded. (ISO 4217-3A)
InstrumentGroupCode	72	3	ASCII Str.	Designates the group of instruments to which the security belongs.
InstrumentCategory	75	1	ASCII Ch.	Indicates to which category, the security belongs. Valid values are: ■ 'A' – Stock ■ 'O' – Bond ■ 'W' – Warrant and certificate ■ 'T' – Tracker ■ 'D' – Miscellaneous
InstrumentCode	76	12	ASCII Str.	ISIN Code, International Securities Identification Number, according to the ISO-6166.
DateOfLastTrade	88	8	ASCII Str.	Date of the last trading day during which a trade was executed for the instrument. ('YYYYMMDD'). Provided only in the message sent in the morning. Not provided for European instruments.
UnderlyingRepoISINCode	96	12	ASCII Str.	Underlying security (security used in the loan quotation system) for loan contracts on centralized lending market.
RepoExpiryDate	108	8	ASCII Str.	Represents the inclusive date until which a lending/borrowing contract can be traded.
FirstSettlementDate	116	8	ASCII Str.	Represents the first possible settlement date for a given instrument with the instrument's depository. When this date is not provided, it means that the

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				first possible settlement date is the same as the instrument's flotation date. This item is provided solely for Amsterdam-listed instruments of the type As If and When Issued. In other words, it is provided solely for new issues for which the first settlement date is a considerable length of time in the future, or is still not known even though it is already possible to trade the instrument. As long as the date remains unknown, this is a fictitious date that must be modified as soon as the real date is known. In terms of instrument types, the instrument can be either a bond or a warrant. This item is determined as follows. If the marketplace = 038 (Amsterdam), then: ■ If the instrument is a bond, the first possible settlement date is the settlement date for the issue price if this item is not set to zero. ■ If the instrument is a warrant, the first possible settlement date is the settlement date for the issue price (taken from the Warrant Characteristics message) if this item is not set to zero. In all other cases, this item is not provided. Used by the clearing house in the rule for determining the theoretical settlement date for a trade. Valid values are: ■ Nulls – If not provided ■ 'YYYYMMDD' format ■ '20111111' is the date used for Dutch warrants for which the settlement date is unknown at the time the instrument is floated

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
TypeOfDerivatives	124	1	ASCII Ch.	Type of derivative instrument associated to the security. Valid values are: ■ '1' – INAV (Indicative Net Asset Value for ETF-Exchange Traded Funds) ■ '4' – Supports short terms options, ■ '6' – Supports long term options ■ '9' – No
BICDepositary	125	11	ASCII Str.	Identifies the depository organization for the shares of an instrument, or when the same organization manages several systems, this item identifies the relevant settlement system for settling trades on a given listed security. The ID is allocated by SWIFT, according to the BIC standard (ISO 9362). Used by the Clearing 21® system to determine the relevant system for settling trades in a Brussels-listed instrument (the Bank of Belgium's X/N system or VIF system), and thus, the time at which the transactions must be sent during the operating day, and the format in which they must be sent.
ICB	136	4	ASCII Str.	Identifies for a listed instrument, the economic subsector of the issuing company in the ICB (Industry Classification Benchmark) classification.
MIC	140	4	ASCII Str.	Identifies the market to which an instrument y belongs by its MIC (Market Identification Code), according to ISO 10383. Euronext owns the following MICs: ■ 'ALXA' – ALTERNEXT AMSTERDAM ■ 'ALXB' – ALTERNEXT BRUSSELS ■ 'ALXP' – ALTERNEXT PARIS ■ 'EMTF' – EURO MTF ■ 'ENXB' – EASYNEXT BRUSSELS

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ 'ENXL' – EASYNEXT LISBON
				■ 'MLXB' - MARCHE LIBRE BRUSSELS
				■ 'TNLA' – TRADED BUT NOT LISTED AMSTERDAM
				■ 'TNLB' – EURONEXT – TRADING FACILITY BRUSSELS
				■ 'VPXB' - EURONEXT - VENTES PUBLIQUES BRUSSELS
				■ 'WQXL' – MARKET WITHOUT QUOTATIONS LISBON
				■ 'XAMS' – EURONEXT AMSTERDAM
				■ 'XBRU' – EURONEXT BRUSSELS
				■ 'XLIS' – EURONEXT LISBON
				■ 'XLDN' – EURONEXT LONDON
				■ 'XLUX' – LUXEMBOURG STOCK EXCHANGE
				■ 'XMLI' – MARCHE LIBRE PARIS
				■ 'XPAR' – EURONEXT PARIS
UnderlyingWISINCode	144	12	ASCII Str.	Gives the trading code of the underlying listed security of a warrant.
DepositaryList	156	25	ASCII Str.	Identifies the possible main depository organizations (maximum five) for the shares or bonds for an instrument. Used the clearing house to determine the relevant system for settling trades. Valid values are: ■ '00001' – Euroclear France ■ '00002' – CIK (Belgium) ■ '00003' – NECIGEF (the Netherlands) ■ '00004' – X/N (BoB's system) ■ '00005' – VIF (non-fungible Belgian instruments) ■ '00006' – Euroclear Bank ■ '00007' – NIEC

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ '00008' – Physical ■ '00009' – Euronext Paris non Euroclear France ■ '00010' – Interbolsa ■ '00030 – Euroclear UK / Ireland Limited ■ '00000' – No depository organization ■ Nulls – Not significant
MainDepositary	181	5	ASCII Str.	Identifies the default (or main) depository organization of the instrument (between the possible 5 depositaries registered) used by priority for the settlement (for example: multi-listed instruments which have several depositories). This data has to be treated in consideration of the data DepositoryList. Used by the clearing house to determine the relevant system for settling trades. Valid values are the same as for "DepositoryList".
TypeOfCorporateEvent	186	2	ASCII Str.	Indicates the last type of corporate event that has occurred on an instrument, such as detachment of rights, or of coupons. The data item is automatically calculated by the adjustment application but in case of problem or error, the data item value could be modified manually, particularly for purging the order book in case of absence of corporate event. This data has to be treated in consideration of the date of the event included into the header of the message. Valid values are: ■ '00' – No specific event ■ '01' – Dividend payment in cash or in stocks

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ '02' – Interest payment (bonds for which the price is not expressed in % of the nominal, only)
				■ '04' – Split
				■ '05' – Bonus (i.e. attribution)
				■ '06' – Subscription
				■ '07' – Share allocation
				■ '08' – Share swap
				■ '09' – Reverse split
				■ '10' – Merger
				■ '11' – Final bond redemption
				■ '12' – Capital amortization
				■ '13' – Draw announcement (Belgian bonds only)
				■ '14' – Block trade of controlling interest
				■ '15' – Optional corporate events(dividend option)
				■ '16' – Complex corporate event
				■ '17' – Purge of the order book (purge is initiated manually in the absence of a corporate event, for example, following the modification of the variable tick of the listed security)
				■ '22' Bourse de Luxembourg corporate event
TimeLagEuronextUTC	188	5	ASCII Str.	Effective difference time between CET (Euronext time) and UTC. To be interpreted in conjunction with the time difference between MiFID regulators and UTC. Valid values are: ■ 'SHHMN' format (with S = + / − , HH = Hour, MN=Minutes) ■ Always provided (MiFID instrument) ■ '0000' means no difference time

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
TimeLagMiFIDRegUTC	193	5	ASCII Str.	Effective time difference between MiFID regulators and UTC. To be interpreted in conjunction with the time difference between CET (Euronext time) and UTC. Valid values are: ■ 'SHHMM' format (with S = + / − , HH = Hour, MM=Minutes) ■ Always provided (MiFID instrument) ■ '0000' means no difference time
CFI	198	6	ASCII Str.	Classification code of a financial instrument defined by the ISO-10962 standard. The structure of the CFI code: The CFI reflects characteristics that are defined when a financial instrument is issued, and remain unchanged during its entire lifetime. The CFI consists of six alphabetical characters. The first character indicates the highest level of classification (Categories): ■ 'E' – Equities ■ 'D' – Debt instruments ■ 'R' – Entitlements (Rights) ■ 'O' – Options ■ 'F' – Futures ■ 'M' – Others/Miscellaneous The second character indicates specific groups within each category: Groups, for example, for equities: ■ Shares ■ Preferred shares ■ Convertible preferred shares ■ Units, i.e. unit trusts/mutual funds etc. ■ Others The four last characters indicate the most important attributes applicable to

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				each group: whereas voting rights, restrictions, payment status and form are useful information in Equities, these features do not exist for Options, which have other attributes (underlying instruments, type of scheme, delivery, standardized/non-standardized). In Equities, Debt instruments and Entitlements, the sixth (last) character indicates the form of the instrument. If the information is not available or applicable at the time of assignment, the code "X" is to be used for the respective element, i.e. X = not applicable, unknown, not available.
QuantityNotation	204	4	ASCII Str.	Specifies the nature of the amount expression used for negotiating the instrument on the market. Valid values are: ■ 'UNT' – In unit (i.e. number of shares), left padded ■ 'FMT' – In facial amount (i.e. bonds expressed in %), left padded ■ Null – Not applicable
IndexSetOfVarPriceTick	208	2	ASCII Str.	The key for the variable price tick table consists of two data items [the index for a set of variable price ticks, the lowest value in a range of prices]. The index data item refers to a set of lines in that table which make it possible to determine the price tick for an instrument, based on the price range in which a given price for the instrument falls (i.e. a price to be rounded off or a limit to be checked). When a listed security is created, or when the characteristics of an existing listed security are modified, this data item is (re)initialized to the "typical" index value for this listed security (if a value is not provided for the Fixed Price Tick data item).

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				This "typical index" is derived from the following three characteristics: ■ Trading currency (type of unit in which the listed security's price is expressed), ■ Broad instrument category associated with the listed security, ■ Trading group.
MarketFeedCode	210	2	ASCII Str.	The "Market data flow" to which the instrument belongs. Possible values are listed in Market Feed Code.
MICList	212	24	ASCII Str.	Identifies the Euronext markets on which a security is listed by its MIC (Market Identification Code). For a security listed on a single Euronext market, the listing MIC code is the same than "Market Identification Code (MIC) of the listed security" For a security listed on several Euronext Markets: ■ The first MIC is the same than the "Market Identification Code (MIC) of the listed security ■ The others MIC indicate the other listing places
IndustryCode	236	4	ASCII Str.	Not used
Filler	240	4	ASCII Str.	Filler to retain the structure of the message – possibly for future use (note that this filler is used for the PrimaryMIC field in the NYSE Arca Europe and SmartPool XDP feed).
SettlementDelay	244	2	ASCII Str.	Period in number of days between the trade date and the settlement date. Valid values are: ■ [0 – 10] – Possible values ■ 'X' – General value assigned for

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				most of securities (Equities, Bonds) and corresponding to 2 days. ■ 'Z' – Dedicated value for Lending/Borrowing and corresponding to 3 days.
MaturityDate	246	8	ASCII Str.	The date when a bond or other debt instrument becomes due or payable.
GuaranteeIndicator	254	1	ASCII Ch.	Indicator for Guarantee in the Settlement System. Valid values are: ■ '0' – Any trade executed on this instrument will be cleared and Guaranteed by a Clearing House. ■ ' 1' – Any trade executed on this instrument will be cleared but not Guaranteed by a Clearing House. ■ '2' – Any trade executed on this instrument is not clearable by a Clearing House. ■ '8' – Not applicable as corresponding to non-tradable instruments
DarkEligibility	255	1	ASCII Ch.	Eligibility of an instrument for the Dark liquidity. Valid values are: ■ '0': No (Not Eligible / Applicable) ■ '1': Yes (Eligible)
DarkLISThreshold	256	8	Binary Int.	Define the minimum amount of an order to benefit from the LIS (Large In Scale) pre-transparency waiver.
DarkMinimumQuantity	264	4	Binary Int.	Define the minimum quantity required for an order to be eligible for the Dark liquidity. This is solely defined by Euronext. Format: No decimal portion. '0': No minimum required
ParValue	268	4	Binary Int.	Par Value (all called Nominal value) for Instrument. For Bond it represents the par amount

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				to be repaid at maturity (not including interest revenue) .
ParValueScaleCode	272	1	Binary Int.	To be combined with ParValue.
Filler	273	67	ASCII Str.	Filler to retain the structure of the message – possibly for future use
FinancialMarketCode	340	3	ASCII Str.	For details of what market each possible code can refer to, see Financial Market Code.
USIndicators	343	7	ASCII Str.	Not used.
Filler	350	1	ASCII Str.	Filler to retain the structure of the message – possibly for future use.
MaxDecimalPlacesInQuantity	351	1	ASCII Str.	Maximum number of decimal per instrument.
PrevDayCapitalTraded	352	8	Binary Int.	Cumulative capital traded for all previous day trades on a security (to be calculated with the PrevDayCapitalTradedScaleCode). Provided only in the message sent in the morning. Not provided for European instruments.
NomMktPrice	360	8	Binary Int.	Amount of the nominal value of the security (to be calculated with the NomMktPriceScaleCode).
LotSize	368	8	Binary Int.	Amount, expressed in number of shares or in an amount or a volume of the capital, of the lot size. The lot size is a minimum tradable quantity that is set for each instrument by Euronext. The quantity of an order entered by a trading member on the market must be a multiple of the lot size. This number is also called the "Quotité de Marché" (Minimum market tradable quantity). For bonds, this data has to be considered with the data "Amount of par value for instrument for calculating trade amount".

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				This item is calculated in the following way:
				For Brussels-listed bonds that are quoted in %:
				■ If 1 ≤ market par value ≤99 999 999, then the Instrument Lot Size is the integer part of the market nominal (and, moreover, the lot size and the par value for trade amount are set to 1).
				For Amsterdam-listed bonds quoted in %:
				■ If 1≤ initial par value ≤ 99 999 999, then the Instrument Lot Size is the integer part of the initial nominal (and, moreover, the lot size and the par value for trade amount are set to 1).
				For Lisbon-listed bonds quoted in %:
				■ If 1≤ market par value ≤ 99 999 999, then the Instrument Lot Size is the integer part of the market nominal (and, moreover, the lot size and the par value for trade amount are set to 1).
				■ If the market par value is not an integer, until 4 decimals (over, the number is rounded to 4 decimals) the Instrument Lot Size is set to an integer multiple of the market par value (and, moreover, the par value for trade amount is set to 1).
				In all other cases, the Instrument Lot Size is the item Instrument Lot Size that is provided by PGD-OST (this is a whole number of shares or bonds).
				Note 1: Only positive values are possible. If not provided IFT = space and QMT = 000000000000.
				Note 2: Only integer values that are equal to or greater than one are

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				accepted until Euronext systems have been adapted for using quantities expressed as a par value amount (Decimalization project). According to LotSizeScaleCode.
NumberInstrumentCirc	376	8	Binary Int.	Number of shares issued/bonds outstanding after payments.
SharesOut	384	8	Binary Int.	Not used.
AuthShares	392	8	Binary Int.	Not used.
Filler	400	3	ASCII Ch.	Filler to retain the structure of the message – possibly for future use.
RepoIndicator	403	1	ASCII Ch.	Indicates whether the security listed underlies any loan contracts, meaning it has been admitted to the Deferred Settlement system and/or to the lending market. Valid values are: ■ '0' – Instrument neither eligible for the SRD, nor eligible for the Loan and Lending Market ■ '1' – Instrument eligible for the SRD and for the Loan and Lending Market ■ '2' – Instrument eligible for the SRD long only ■ '3' – Instrument eligible for the Loan and Lending Market and for the SRD long only ■ '4' – Easy-to-borrow Instrument eligible for the SRD and the for Loan and Lending Market ■ '5' – Instrument eligible for the Loan and Lending Market ■ '8' – Non significant
LastAdjPriceScaleCode	404	1	Binary Int.	To be combined with LastAdjPrice.
TypeOfUnitExp	405	1	Binary	Unit in which the security is quoted.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	Valid values are: ■ '1' – In units ■ '2' – As a % of nominal (excluding accrued interest - Clean) ■ '5' – As a % of nominal (including accrued interest - Dirty) ■ '8' – In kilograms ■ '9' – In ounces
MarketIndicator	406	1	Binary Int.	Indicates the market regulations governing the market on which the security is traded. Valid values are: ■ 0 – Not used ■ 1 – Cash ■ 9 – Not defined
PrevDayCapitalTradedScaleCode	407	1	Binary Int.	To be combined with PrevDayCapitalTraded.
TaxCode	408	1	Binary Int.	Tax deduction code to which the security belongs. Valid values are: ■ '0' – Not eligible to PEA ■ '3' – Eligible to PEA ■ '9' – Not provided
NomMkdtPriceScaleCode	409	1	Binary Int.	To be combined with NomMkdtPrice.
LotSizeScaleCode	410	1	Binary Int.	To be combined with LotSize.
FixPriceTickScaleCode	411	1	Binary Int.	To be combined with FixPriceTick.
Mnemo	412	5	ASCII Str.	Mnemonic code of the instrument. This field is not populated for every instrument.
TradingCode	417	12	ASCII Str.	Trading code of the instrument. This is the only instrument identifier that is unique in the feed in addition to the symbol index.
Filler	429	3	ASCII Str	Filler to retain the structure of the message – possibly for future use.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
StrikePrice	432	4	Binary Int.	The specified price of an option contract at which the contract may be exercised, whereby a call option buyer can buy the underlying or a put option buyer can sell the underlying. The buyer's profit from exercising the option is the amount by which the strike price exceeds the cash instrument price (in the case of a call), or the amount by which the cash instrument price exceeds the strike price (in the case of a put). In general, the smaller the difference between spot (cash instrument price) and strike price, the higher the option premium. Also called exercise price. According to StrikeScaleCode. Only provided for warrants or other derivative instruments.
StrikeCurrency	436	3	ASCII Str	Code for the currency of the strike price applied for an instrument whose maturity is reached (ISO 4217-3A). Only provided for warrants or other derivative instruments.
StrikeScaleCode	439	1	Binary Int.	To be combined with StrikePrice.
CurrencyCoef	440	4	Binary Int.	Change ratio coefficient of currency applied to instrument. Used in conjunction with one of the change rate indicators in order to apply this coefficient to a currency among two available currencies defined for the instrument: ■ Trading currency ■ Currency code of strike price for derivative Instrument The currency to which this coefficient will be applied depends on just one of the two values set to the related indicators (defined below in the integrity constraints).

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				This coefficient is used when the currency is not compliant with the ISO 4217 (3A) standard as pence (GBP) or cent (USD) expression of an official currency. In this case the formula to apply in order to retrieve the price expressed in an official currency is: Real price in Trading currency = Traded price (i.e. 1565 pence) x Change ratio coefficient value (0.01).
CurrencyCoefScaleCode	444	1	Binary Int.	To be combined with CurrencyCoef.
TradingCurrencyIndicator	445	1	Binary Int.	Change rate indicator for currency of Instrument Traded Price. Indicates if the change rate will be applied to the currency defined for traded prices of the instrument. Valid values are: ■ '– 0' – Change rate not applied to the traded price ■ '– 1' – Change rate applied to the traded price ■ Null – Not applicable
StrikeCurrencyIndicator	446	1	Binary Int.	Change rate indicator for currency of the strike price. Indicates if the change rate will be applied to the currency defined for strike prices of the instrument. Valid values are: ■ '– 0' – Change rate not applied to the strike price ■ '– 1' – Change rate applied to the strike price ■ Null – Not applicable
Filler	447	9	ASCII Str.	Filler to retain the structure of the message – possibly for future use.

3.2 TRADES

3.2.1 Overview

The Trades message types are:

- 221 – Trade Cancel
- 240 – Trade Full Information
- 241 – Trade Price Update
- 242 – TCS Trade
- 243 – Trade Publication
- 244 – Settlement Price
- 245 – Auction Summary

3.2.2 Packet Header Format

All messages are preceded by a common packet header format**. Table 37 Packet Header Format** describes the header fields of a Trade messages.

Table 37 Packet Header Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-byte packet header.
PacketType	2	2	Binary Int.	Identifier for the type of data contained in the packet. '501' – Market Data Packet
PacketSeqNum	4	4	Binary Int.	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and monotonically and is reset to 1 at the beginning of each trading day. The PackSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00 UTC.
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream. Possible values are described in Feed Configuration descriptions

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method: ■ '0' – Real-Time message (Uncompressed) ■ '2' – Retransmission message (Uncompressed) ■ '17' – Refresh message (zlib Compressed)
NumberMsgEntries	15	1	Binary Int.	The number of messages that are contained within the packet.

3.2.3   Trade Cancel – 221 Message

3.2.3.1   Message Overview

A Trade cancel message informs that a trade which was already executed has been cancelled.

3.2.3.2   Message Sending Rules

This message is sent each time a trade executed on the trading engine is cancelled.

3.2.3.3   Message Structure

Table 38 Trade Cancel Message Format describes the body fields of an Equities Trade message, MsgType = '221' Trade cancel.

Table 38 Trade Cancel Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'221' – Trade Cancel
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the Trade generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
OriginalTradeIDNumber	16	4	Binary Int.	This field refers to the initial TradeIDNumber of the trade concerned (as reported in a 240 message) in this cancellation message.
SystemID	20	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	24	2	Binary Int.	Number of microseconds. To be combined with SourceTime field.
Filler	26	2	Binary Int.	Filler to retain the structure of the message – possibly for future use.

3.2.4   Trade Full Information – 240 Message

3.2.4.1   Message Overview

A Trade Full Information message is sent each time a trade has occurred and a trade message is sent. The specific message provides the detailed information related to the trade.

3.2.4.2   Message Sending Rules

This message is sent each time a trade is taking place on the trading engine. As for each trade a msg 240 is sent, clients should not process both messages as "reporting of a new trade".

■   This message is sent each time a trade takes place on the trading engine.

■   If TradeCond3 = '4' this represents a valuation trade. A valuation trade is disseminated to provide a price indication for an instrument that has not been traded during the last trading session. This price indication is based on Liquidity Provider activity.

■   If TradeCond3 = '7' this represents the cumulated volume data of the day. In this case, when the 240 message is sent, the 'CumulativeQuantity' field represents the sum of TCS Cumulative Quantity and UTP Cumulative Quantity (in addition, TradCond2=1, OpeningTradeIndicator=S, and the following fields are set to 0: TradeIDNumber, Price, Volume, HighestPrice, LowestPrice, VariationLastPrice, TickDirection).

3.2.4.3   Message Structure

Table 39 Trade Full Information Structure describes the body fields of an Equities Trade message, MsgType = '240' Trade Full Information.

Table 39 Trade Full Information Structure

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'240' – Full Information
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the Trade generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
TradeIDNumber	12	4	Binary Int.	Unique numeric and increasing Identifier of the trade, set by the trading engine.
QuoteLinkID	16	4	Binary Int.	Not used.
SourceSeqNum	20	4	Binary	This field specifies the sequence number assigned by the source system to this message. Please

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	note that while the sequence number increases serially, it does not increase monotonically.
Price	24	4	Binary Int.	Price of the trade (to be calculated with the PriceScaleCode).
Volume	28	4	Binary Int.	Number of instruments transacted in this trade.
CumulativeQuantity	32	4	Binary Int.	Cumulative number of instruments traded since the start of the current trading session.
HighestPrice	36	4	Binary Int.	Highest Price traded during the day (to be calculated with the PriceScaleCode).
LowestPrice	40	4	Binary Int.	Lowest Price traded during the day (to be calculated with the PriceScaleCode).
VariationLastPrice	44	4	Binary Int. (signed)	Percentage variation of today's price/last reference price (to be calculated with the VariationScaleCode).
SystemID	48	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	52	2	Binary Int.	Number of microseconds. To be combined with SourceTime field.
Filler	54	2	Binary Int.	Filler to retain the structure of the message – possibly for future use.
TradCond1	56	1	ASCII Ch.	Not used.
TradCond2	57	1	ASCII Ch.	Last trade at same price indicator. Valid values are: ■ '0' – Not last of a series of trade at the same price ■ '1' – Last of a series of trade at the same price
TradCond3	58	1	ASCII Ch.	Cross Trade Indicator. Valid values are: ■ '0' – Trade does not stem from a Cross Order ■ '1' – Trade stems from a Cross Order ■ '4' – Valuation trade ■ '7' – Indicates the cumulated volume data of the day, either traded on UTP or received from TCS (CumulativeQuantity=Cumulative Quantity of TCS + CumulativeQuantity of UTP)

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ 'N' – For future use
				■ 'R' – Retail Matching Facility Trade
TradCond4	59	1	ASCII Ch.	Not used.
TickDirection	60	1	ASCII Ch.	Symbol of the variation of the price / previous prices. Valid values are: ■ '+' – positive ■ '-' – negative ■ '0' – no variation or data field not significant ■ Null – not significant
OpeningTradeIndicator	61	1	ASCII Ch.	If the trade took place during the opening auction or during the core session. Valid values are: ■ 'O' – Opening ■ 'S' – Core Session **Note**: The first trade of the day will always have a value of 'O' – even if it occurs during the core session.
VariationScaleCode	62	1	Binary Int.	To be combined with VariationLastPrice.
PriceScaleCode	63	1	Binary Int.	Applicable to all prices in the message.
TradingDate	64	4	Binary Int.	Date at which the trade was executed. (YYYYMMDD)
TradingTime	68	4	Binary Int.	Time at which the trade was executed. (HHMMSSsss)
TradingTimeMicroSecs	72	2	Binary Int.	Number of microseconds. To be combined with TradingTime field. (µµµ)
TransparencyInd	74	1	Binary Int.	Indicates if the trade has benefited from the Dark liquidity and if the publication has been deferred. ■ 0: Lit / Regular trade ■ 1: Dark trade and immediate publication ■ 2: Dark trade and deferred publication
Filler	75	1	Binary Int.	Filler to retain the structure of the message – possibly for future use.

3.2.5   Price Update – 241 Message

3.2.5.1   Message Overview

The Price update message contains a bid price or an ask price, or the modification of the last reference price on a given instrument. This message informs of modifications operated on instrument prices.

3.2.5.2   Message Sending Rules

This message is sent:

■   When the market operation changes the last adjusted price of the previous day (type of price 34).

■   When a security cannot quote despite the orders present in the book. In this case, the market operation enters a bid or an ask price on a security depending on the sense of the imbalance (type of price: 02, 03).

■   When the iNAV fixing time for the Investment Funds groups traded on TCS is reached. The message is only sent for Investment Funds instruments.

■   When Euronext Cash Market Operations enters an iNAV on behalf of the fund manager agent after iNAV fixing time for the Investment Funds groups traded on TCS.

■   When the market operation resends Key Market Data (via Market Operation monitoring tool).

3.2.5.3   Message Structure

Table 40 Price Update Message Format describes the body fields of an Equities Trade message, MsgType = '241' Price Update.

Table 40 Price Update Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'241' – Price Update
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the Trade generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
Price	16	4	Binary Int.	Last Price of the trading day (to be calculated with the ScaleCode).

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
HighestPrice	20	4	Binary Int.	Highest Price traded during the day (to be calculated with the PriceScaleCode).
LowestPrice	24	4	Binary Int.	Lowest Price traded during the day (to be calculated with the PriceScaleCode).
VariationLastPrice	28	4	Binary Int. (signed)	Percentage variation of today's price/last reference price (to be calculated with the VariationScaleCode).
SystemID	32	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	36	2	Binary Int.	Number of micro seconds. To be combined with SourceTime field.
TypeOfPrice	38	2	Binary Int.	European Market Code identifying the type of price that is updated. Valid values are: '04' - Net Asset Value (+/-) for the instruments eligible to the Euronext Fund Service '27' – Net Asset Value (NAV) '28' – Subscription price for instruments eligible to Euronext Fund service '29' – Redemption price for instruments eligible to Euronext Fund service' '33' – New last price '34' – New previous day's closing price '43' – Subscription Price '51' – Indicative price
Filler	40	2	Binary Int.	Filler to retain the structure of the message – possibly for future use.
PriceScaleCode	42	1	Binary Int.	Applicable to all prices in the message.
VariationScaleCode	43	1	Binary Int.	Available for the price variation in the message.

3.2.6   TCS Trade – 242 Message

3.2.6.1   Message Overview

A TCS trade message indicates either:

■   The automatic matching of two trade declarations outside the central order book (the buyer's and the seller's), or the reception of a single valid declaration made in the names of the buyer and the seller

■   The cancellation of a TCS trade

3.2.6.2   Message Sending Rules

This message is sent:

- When TCS receives a single valid declaration or two declarations which can be matched automatically (the publication of a trade can be delayed)

- A TCS trade is cancelled

- A digest on TCS activity for the day, built on the basis of TCS statements

3.2.6.3   Message Structure

Table 41 TCS Trade Message Format describes the body fields of a Equities Trade message, MsgType = '242' TCS Trade.

Table 41 TCS Trade Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'242' – TCS Trade.
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the Trade generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
Price	16	4	Binary Int.	Price at which the security was negotiated (to be calculated with the PriceScaleCode).
Filler	20	4	ASCII Str.	Filler to retain the structure of the message – possibly for future use.
Volume	24	8	Binary Int.	Number of negotiated securities.
FirstPrice	32	4	Binary Int.	Summary Activity (TCSSubtypeCode='R') – First price (to be calculated with the PriceScaleCode).

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
LastPrice	36	4	Binary Int.	Summary Activity (TCSSubtypeCode='R') – Last price (to be calculated with the PriceScaleCode).
HighestPrice	40	4	Binary Int.	Summary Activity (TCSSubtypeCode='R') – High price (to be calculated with the PriceScaleCode).
LowestPrice	44	4	Binary Int.	Summary Activity (TCSSubtypeCode='R') – Low price (to be calculated with the PriceScaleCode).
CumulativeCapital	48	4	Binary Int.	Summary Activity (TCSSubtypeCode='R') – Cumulative amount of capital exchanged for a given trading day and instrument (K Euro).
Filler	52	4	ASCII Str.	Filler to retain the structure of the message – possibly for future use.
CumulativeNumberSecurities	56	8	Binary Int.	Summary Activity(TCSSubtypeCode='R') – Cumulative quantity for a given trading day and instrument.
SystemID	64	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	68	2	Binary Int.	Number of micro seconds. To be combined with SourceTime field.
TCSTradeID	70	16	ASCII Str.	Unique Identifier of the Trade (European Market).
TradingVenue	86	4	ASCII Str.	Identifies the Euronext market on which a security is traded by its Market Identification Code). Valid values are: ■ Initial MIC of instrument if TCS Operation Type = D, E,H, I ■ Other Euronext MIC but different of the initial MIC of instrument if TCS Operation Type = R
DateOriginalDecl	90	8	ASCII Str	Date declarations were matched.
TimeOriginalDecl	98	6	ASCII Str	Time declarations were matched.
StartTimeVwap	104	6	ASCII Str.	Start time for the Volume Weight Average price computation period.
EndTimeVwap	110	6	ASCII	End time for the Volume Weight Average

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Str.	price computation period.
EffectiveDateIndicator	116	1	ASCII Ch.	Declarations making up the trade were introduced on the day. Valid values are: ■ '0' – date that the seller's declaration was received ■ '1' -Trading day preceding day seller's declaration was received
BlockTradeCode	117	1	ASCII Ch.	Trades relates to a block or a negotiated deal following MiFID rules. Valid values are: ■ 'B' – Block Trade ■ 'N' – Regular trade or Negotiated deal ■ Null – undefined
TradeCancelIndicator	118	1	ASCII Ch.	Indicates if the trade specified was cancelled. Valid values are: ■ '0' – Valid trade ■ '1' – Cancelled trade ■ Null – Summary message
TradeType	119	1	ASCII Ch.	Type of Operation. Valid values are: ■ 'D' – Delta Neutral Liffe Connect ■ 'E' – Market VWap operation ■ 'H' – Out of Market ■ 'I' – Euronext Fund Service ■ 'R' – Secondary listing place ■ 'Y' – Exchange for Physical
FinancialMarketCode	120	1	ASCII Ch.	Financial Market as defined the Investment Service Directive. Valid values are: ■ 'S' – Paris ■ 'B' – Brussels ■ 'A' – Amsterdam ■ 'P' – Lisbon
TCSSubtypeCode	121	1	ASCII Ch.	The subtype code used by the Trade Confirmation System, as follows: ■ 'R' – Complete summary (first price, number of trades, and so forth)

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ 'S' – Trade without summary (no summary information – just the basic trade)
PriceScaleCode	122	1	Binary Int.	Applicable to all prices in the message.
VolumeScaleCode	123	1	Binary Int.	Applicable to all quantities in the message.
Transaction Type Indicator	124	1	ASCII Ch.	Transaction Type Indicator possible values: ■ 'P' – Plain Vanilla Trade ■ 'D' – Dark Trade ■ 'B' – Benchmark Trade ■ 'T' – Technical Trade ■ 'G' – Give-up/Give-in Trade ■ 'E' – Ex/Cum dividend Trade ■ 'F' – Trade with condition ■ ' ' (space) – Not defined
Filler	125	3		

3.2.7 Trade Publication – 243 Message

3.2.7.1 Message Overview

A Trade Publication message is sent by TCS whenever an OTC trade under the MiFID regulation is sent to TCS. The trade is executed either outside the regulated market by trading members that are MiFID-compliant either issued by the matching of two trade declarations applied on OTC operations.

Each new trade is stored on TCS. A trade publication message is sent through the Market Data dissemination flows if and at the moment that it is defined for publication.

This message is either sent for a trade creation or cancellation.

3.2.7.2 Message Sending Rules

This message is sent:

■ When TCS receives a single valid declaration or two declarations which can be matched automatically (the publication of a trade can be delayed)

■ A Trade Publication is cancelled

3.2.7.3 Message Structure

Table 42 Trade Publication Message describes the body fields of an Equities Trade message, MsgType = '243' Trade Publication.

Table 42 Trade Publication Message

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'243' – Trade Publication
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the Trade generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
Price	16	4	Binary Int.	Price at which the instrument was traded, reported to Euronext (to be calculated with the PriceScaleCode).
Quantity	20	4	Binary Int.	Number of traded units.
PriceMultiplier	24	4	Binary Int.	Number of units of the financial instrument that are contained in a trading lot (to be calculated with the MultiplierScaleCode).
SystemID	28	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	32	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
Transaction Type Indicator	34	1	ASCII Ch.	Transaction Type Indicator possible values: ■ 'P' – Plain Vanilla Trade ■ 'D' – Dark Trade ■ 'B' – Benchmark Trade ■ 'T' – Technical Trade ■ 'G' – Give-up/Give-in Trade ■ 'E' – Ex/Cum dividend Trade ■ 'F' – Trade with condition

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ ' ' (space) – Not defined
Filler	35	1	Binary Int.	Filler to retain the structure of the message – possibly for future use.
TradingDate	36	8	ASCII Str.	Date at which the trade was executed.
TradingTime	44	6	ASCII Str.	Time at which the trade was executed.
ReportingDate	50	8	ASCII Str.	Date at which the trade was reported to Euronext Cash markets.
ReportingTime	58	6	ASCII Str.	Time at which the trade was reported to Euronext Cash markets.
ISIN	64	12	ASCII Str.	ISIN code of the Instrument.
QuantityNotation	76	4	ASCII Str.	Indication to whether the quantity is a number of units, the nominal value of bonds or a number of contract.
TradeReference	80	30	ASCII Str.	Reference of the trade reported to Euronext.
TCSTradeId	110	16	ASCII Str.	Unique reference of the trade provided by Euronext.
TradeActionIndicator	126	1	ASCII Ch.	Type of update for the trade. Valid values are: ■ '1' – Insertion, ■ '0' – Deletion
PriceNotation	127	3	ASCII Str.	Code of the currency (ISO 4217-3A).
Venue	130	11	ASCII Str.	Indicates the venue where the trade took place. Valid values are: ■ BIC Code (ISO 9362) ■ OTC (Over the counter) ■ SI (Systematic Internaliser) ■ XSMP (SmartPool)
DelayedIndicator	141	1	ASCII Ch.	Subject to a deferred publication or not. ■ '0' – Not Delayed ■ '1' – Deferred ■ Null – Not available

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PriceScaleCode	142	1	Binary Int.	Applicable to all prices in the message.
MultiplierScaleCode	143	1	Binary Int.	To be combined with PriceMultiplier.

3.2.8   Settlement Price – 244 Message

3.2.8.1   Message Structure

The settlement price message defines the daily clearing price of the underlying instrument for a security lending contract (that means the interest rate of the Lending Borrowing contract that is dealt the day that the message is sent).

3.2.8.2   Message Sending Rules

This message is sent every morning before the beginning of the trading session for all the securities eligible to the lending/borrowing lending/borrowing and SRD market.

3.2.8.3   Message Structure

Table 43 Settlement Price Message Format describes the body fields of an Equities Trade message, MsgType = '244' Settlement Price.

Table 43 Settlement Price Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'244' – Settlement Price
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the Trade generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
Price	16	4	Binary Int.	Settlement price (to be calculated with the ScaleCode).
SystemID	20	4	Binary	The ID of the originating Exchange/System of the

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	message. See System ID.
SourceTimeMicroSecs	24	2	Binary Int.	Number of micro seconds. To be combined with SourceTime.
PriceScaleCode	26	1	Binary Int.	Applicable to all prices in the message.
Filler	27	1	Binary Int.	Filler to retain the structure of the message – possibly for future use.

3.2.9   Auction Summary – 245 Message

3.2.9.1   Message Overview

An Auction Summary message summarizes an instrument's opening trades. The Opening Summary message is sent by the trading engine after an instrument opening (fixing) that has been traded to summarize the opening (fixing), or if the first trade(s) occurred during continuous trading.

3.2.9.2   Message Sending Rules

This message indicates, for each security, the cumulative quantity traded during an auction.

3.2.9.3   Message Structure

Table 44 Auction Summary Message describes the body fields of an Equities Trade message, MsgType = '245' Auction Summary.

Table 44 Auction Summary Message

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'245' – Auction Summary
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	Specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				monotonically.
FirstPrice	16	4	Binary Int.	First traded price for the current trading day (to be calculated with the PriceScaleCode).
LastPrice	20	4	Binary Int.	Most recent traded price for the current trading day (to be calculated with the PriceScaleCode).
HighestPrice	24	4	Binary Int.	Highest traded price for the current trading day (to be calculated with the PriceScaleCode).
LowestPrice	28	4	Binary Int.	Lowest traded price for the current trading day (to be calculated with the PriceScaleCode).
CumulativeQuantity	32	4	Binary Int.	Cumulative quantity traded for the current trading day.
Variation	36	4	Binary Int. (signed)	Percentage variation of the last trade price against the previous day's reference price for the stock concerned, i.e. the last known price or the last price indication (if an indication was entered after the last traded price). (To be calculated with the VariationScaleCode).
SystemID	40	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	44	2	Binary Int.	Number of micro seconds. To be combined with SourceTime.
TypeOfLastPrice	46	2	Binary Int.	Indicates the message trade type. Valid values are: ■ '4' – 1st traded price ■ '7' – nth traded price (resume after reservation)
TickDirection	48	1	ASCII Ch.	Indicates how the last price has moved as compared to the preceding last traded price. Valid values are: ■ '+' – Rising ■ '-' – Falling ■ '0' – Unchanged
InstrumentValuationPrice	49	1	ASCII Ch.	Indicates whether the data item "last price" is a valuation price. Valid values are: ■ '0' – Not a valuation price

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				■ '1' – Valuation price
PriceScaleCode	50	1	Binary Int.	Applicable to all prices in the message.
VariationScaleCode	51	1	Binary Int.	Available for the price variation in the message.

3.3 QUOTES AND BBO10

3.3.1 Overview

List of Quotes message types:

■ 140 – Quotes

■ 146 – Retail Matching Facility (RMF) Quotes

■ 141 – WAS

3.3.2 Packet Header Format

All messages are preceded by a standard header format.

Table 45 Packet Header Format describes the header fields of a Quotes message.

Table 45 Packet Header Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-byte packet header.
PacketType	2	2	Binary Int.	Identifier for the type of data contained in the packet. '501' – Market Data Packet
PacketSeqNum	4	4	Binary Int..	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and monotonically and is reset to 1 at the beginning of each trading day. The PackSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00 UTC.

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream. Possible values are described in Production Feed Configuration and External User Acceptance Feed Configuration.
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method: ■ '0' – Real-Time message (Uncompressed) ■ '2' – Retransmission message (Uncompressed) ■ '17' – Refresh message (zlib Compressed)
NumberMsgEntries	15	1	Binary Int.	The number of messages that are contained within the packet.

3.3.3   Quotes – 140 Message

3.3.3.1   Message Overview

A Quotes message indicates during the trading session any modification of one or more of the ten best limits for an instrument. In Pre-open mode (Class in Pre-Opening phase or Instrument halted), it is also used to indicate the modification of the "Market Summary" for the instrument.

The 'Market Summary' is the summary of the orders that would be executed if the auction took place at the moment that this message was sent. It consists of the Indicative Matching Price and the Indicative Matching Volume (invluding hidden quantities).

3.3.3.2   Message Sending Rules

This message is sent:

■ During the trading session (Pre-opening or Continuous Trading phases) each time one or more of the ten best limits for an instrument is modified.

■ When the instrument is in Pre-open mode to publish the "market summary". The market summary has no meaning except when the Indicative Matching Price has been determined during the Call / Pre-Opening phase. The market summary is not provided if it is equal to the first limit.

3.3.3.3   BBO Display Guidelines

Euronext advises customers to always use the value in level 1 of the 'Quotes' message (Msg Type=140, QuoteNumber=1) as the best bid and offer.

Customers should not use the Indicative Matching Price or Market Summary Quote messages as the basis to publish the best bid and offer.

3.3.3.4   Message Structure

Important: There can be '140' messages associated with more than one order book event for the same instrument in the same packet. An order book event can be one or more events and an event is an order creation, modification or deletion. For each of these events we send out individual 230 Order Update messages with a SourceSeqNum and System ID. The '140' messages associated with the same order book

event for example an order deletion immediately followed by an order creation, will have the same SourceSeqNum and System ID as the last event, the order creation and no separate '140' messages will be generated for the order deletion. It is important to distinguish between particular events within the same packet, otherwise it will lead to missed ticks and, with certain implementations, to a crossed book.

140	140	140	140	140	140	140
Event 1			Event 2+3			Event 4

Table 46 Quotes Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int..	'140' – Quotes
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the Quote generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
QuoteLinkID	16	4	Binary Int.	Not used.
AskPrice	20	4	Binary Int.	Ask Price for Quote (to be calculated with the ScaleCode) Specific values are: ■ 'FF FF FF FE' – Market Order or Market To Limit Order ■ 'FF FF FF FD' – At Opening Order
AskSize	24	4	Binary Int.	Total number of instruments requested in sell orders at the ask price.
BidPrice	28	4	Binary Int.	Bid Price for Quote (to be calculated with the ScaleCode) Specific values are: ■ 'FF FF FF FE' – Market Order or Market To Limit

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				Order ■ 'FF FF FF FD' – At Opening Order
BidSize	32	4	Binary Int.	Total number of instruments requested in buy orders at the bid price.
SystemID	36	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
NumberAskOrders	40	2	Binary Int.	Number of sell orders at the ask price.
NumberBidOrders	42	2	Binary Int.	Number of buy orders at the bid price.
SourceTimeMicroSecs	44	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
TypeOfAskPrice	46	1	Binary Int.	Valid values are: ■ '0' – Limit order ■ '1' – Market order
TypeOfBidPrice	47	1	Binary Int.	Valid values are: ■ '0' – Limit order ■ '1' – Market order
QuoteCondition	48	1	ASCII Ch.	Liquidity provider presence indicator: Valid values are: ■ '0' – No Liquidity provider (LP) ■ '1' – LP only on Ask side ■ '2' – LP only on Bid side ■ '3' – LP on Ask and Bid sides ■ Null – Not provided/Not applicable
QuoteNumber	49	1	Binary Int.	Indicates the level in the order book for the given aggregate quote, derived from its price value. **Note**: Level '0' is dedicated to the market summary (the market summary for an instrument is the summary of the orders that would be executed if the opening of an instrument took place at the moment that this message is sent.) If this field has a value of 0, it should not be processed for order book maintenance.
PriceScaleCode	50	1	Binary	Applicable to all prices in the message.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	
Filler	51	1	ASCII Str	Filler to retain the structure of the message – possibly for future use.

3.3.4   Retail Matching Facility (RMF) Quotes – 146 Message

3.3.4.1   Message Overview

As part of the RMF service, Retail Liquidity Provider (RLP) quotes are published through an RLP Quotes message '146' (structure based on the Quotes message '140'). This RLP Quotes message is broadcast to RMF participants only using a dedicated XDP channel see Production Feed Configuration.

A Retail Matching Facility Quotes message indicates the modification of one or more of the ten best RLP limits for an instrument during the Pre-Opening or Continuous Trading phases. There is no Market Summary information calculated and provided through this message.

3.3.4.2   Message Sending Rules

This message is sent:

■   Each time a Retail Liquidity Provider Order (RLO) is added, modified or pulled.

3.3.4.3   Message Structure

Important: Like for the '140' Quotes messages, there can be '146' messages associated with more than one order book event for the same instrument in the same packet:

146	146	146	146	146	146	146
Event 1			Event 2+3			Event 4

The 146 messages associated with the same event, will have the same SourceSeqNum and SystemID. It is important to distinguish particular events within the same packet, otherwise it will lead to missed ticks and, with certain implementations, to a crossed book.

Table 47 describes the body fields of a MsgType = '146' Retail Matching Facility Quotes.

Table 47 Euronext Retail Matching Facility Quotes Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'146' – Retail Matching Facility Quotes
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially,

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the Quote generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
QuoteLinkID	16	4	Binary Int.	Not used.
AskPrice	20	4	Binary Int.	Ask Price for Quote (to be calculated with the ScaleCode) Valid values are: ■ 'FF FF FF FE' – Market Order or Market To Limit Order ■ 'FF FF FF FD' – At Opening Order
AskSize	24	4	Binary Int.	Total number of instruments requested in sell orders at the ask price.
BidPrice	28	4	Binary Int.	Bid Price for Quote (to be calculated with the ScaleCode) Valid values are: ■ 'FF FF FF FE' – Market Order or Market To Limit Order ■ 'FF FF FF FD' – At Opening Order
BidSize	32	4	Binary Int.	Total number of instruments requested in buy orders at the bid price.
SystemID	36	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
NumberAskOrders	40	2	Binary Int.	Number of sell orders at the ask price.
NumberBidOrders	42	2	Binary Int.	Number of buy orders at the bid price.
SourceTimeMicroSecs	44	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
TypeOfAskPrice	46	1	Binary Int.	Valid values are: • '0' – Limit order
TypeOfBidPrice	47	1	Binary Int.	Valid values are: • '0' – Limit order

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
QuoteCondition	48	1	ASCII Ch.	Liquidity Provider (LP) presence indicator, only provided for instruments traded on a LP quote-driven book. Valid values are: ■ '0' – No Liquidity Provider ■ '1' – LP only on Ask side ■ '2' – LP only on Bid side ■ '3' – LP on Ask and Bid sides ■ Null – Not provided/Not applicable
QuoteNumber	49	1	Binary Int.	Indicates the level in the order book for the given aggregate quote, derived from its price value. Note: there is no market summary.
PriceScaleCode	50	1	Binary Int.	Applicable to all prices in the message.
Filler	51	1	ASCII Str	Filler to retain the structure of the message – possibly for future use.

3.3.5 Weighted Average Spread (WAS) – 141 Message

3.3.5.1 Message Overview

The Weighted Average Spread is an evaluation of the buying and selling average prices of an instrument, for a predetermined amount, given by the instrument's attributed Money Amount for calculation of the WAS.

This message is mainly provided for institutional investors interested in the Euronext's main instruments to be traded in blocks.

The Weighted Average Spread is also used by the TCS system to check that the declarations of trades outside the central order book comply with Euronext's rule book.

3.3.5.2 Message Sending Rules

This message is sent:

■ For opening of an instrument with its group: if (the instrument's Indicative Matching Price could be determined and belongs to the spread defined by the freezing thresholds) and (the Normal Money Amount of Block attribute is not zero), then a Weighted Average Spread message is sent.

■ For opening of an instrument: Same than for Opening of an Instrument with its Group

■ For entering and processing an order in continuous market phase:

If the instrument's attribute Normal Money Amount of Block is not zero, and if the Weighted Average Spread has fluctuated once or more during the last 30 seconds (due to either the incoming order and / or one the previous orders), a Weighted Average Spread message is sent. (N.B.: the Weighted Average Spread broadcasting delay is a parameter).

if the instrument is reserved, if its attribute Normal Money Amount of Block is not zero, a Weighted Average Spread message is sent.

3.3.5.3   Message Structure

Table 48 Weighted Average Spread Message Format describes the body fields of an Equities Quotes WAS message, MsgType = '141' Weighted average spread.

Table 48 Weighted Average Spread Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 byte MsgSize field.
MsgType	2	2	Binary Int.	'141' – Weighted Average Spread
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
SourceTime	12	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
BuyingPrice	16	4	Binary Int.	Buying price of the Weighted Average Spread (to be calculated with the ScaleCode).
SellingPrice	20	4	Binary Int.	Selling price of the Weighted Average Spread (to be calculated with the ScaleCode).
MoneyAmount	24	4	Binary Int.	Money amount for calculation of the WAS (to be calculated with the MoneyScaleCode).
SystemID	28	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	32	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
PriceScaleCode	34	1	Binary Int.	Applicable to all prices in the message.
MoneyScaleCode	35	1	Binary Int.	To be combined with MoneyAmount.

3.4   ORDER BOOK

3.4.1   Overview

List of Orders message types:

- 230 – Order update

- 231 – Order book retransmission delimiter

3.4.2   Packet Header Format

All messages are preceded by a common header format. **Table 49 Packet Header Format** describes the header fields of an Order book message.

Table 49 Packet Header Format

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-bytes packet header.
PacketType	2	2	Binary Int.	Identifier for the type of data contained in the packet. '501' – Market Data Packet
PacketSeqNum	4	4	Binary Int.	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and monotonically and is reset to 1 at the beginning of each trading day. The PackSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00 UTC.
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream. Possible values are described in Production Feed Configuration and External User Acceptance Feed Configuration.
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method: - '0' – Real-Time message (Uncompressed) - '2' – Retransmission message (Uncompressed) - '17' – Refresh message (zlib Compressed)
NumberMsgEntries	15	1	Binary	The number of messages that are contained within

FIELD	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	the packet.

3.4.3   Order Update / Market Sheet – 230 Message

3.4.3.1   Message Overview

The Order update message generated by the trading engines, indicates the creation or modification of a line in the market sheet of normal orders for an instrument. It is also used when the Market Sheet of normal orders is rebroadcast. The deletion of an order from the Market Sheet of normal orders is indicated via a Delete N lines from the Market Sheet message.

This message takes into account any order type, except Stop Loss and Stop Limit (Stop orders).

Stop orders are not broadcasted to the market participants until they are triggered.

3.4.3.2   Message Sending Rules

This message is sent:

■   In the morning, when the trading engine is initialized, to retransmit orders remaining in the book from previous days (taking into account expired orders and order book purges). This is known as the 'order book retransmission' or 'market sheet retransmission' – action type – Y.

■   During the day, each time an order introduced, modified or deleted by a member firm modifies the market sheet (for example, when creating modifying or changing the priority of an order).

■   During the day, in case of order book retransmission. This is a failsafe in circumstances intraday where the order book may need to be resynchronized.

To be noted:

■   SymbolIndex + OrderDate + OrderID uniquely identifies an order

■   In case of retransmission of the order book, at trading engine start up or intraday, an order update message is sent for all the instruments in order to clean the order book prior to the retransmission.

■   Market sheet sequencing:

Orders must be arranged according to:

Order type (OrderType): Priorities are first Market orders, second Market to limit and Opening orders and third Limit and Peg orders

Order price (Price)

Order priority (OrderPriorityDate+OrderPriorityTime+OrderPriorityMicroSecs)

Client applications should be able to do the following:

■   Determine the action type (add, modify, delete, and so forth)

■   Determine the priority of an order based on OrderType, Price, and the OrderPriorityTime stamp (consisting of OrderPriorityDate, OrderPriorityTime and OrderPriorityMicroSecs). The priority of orders of the same type and price depends on their order priority time stamp. The earlier the time stamp, the

higher the priority. Bid orders with higher price have higher priority, ask orders with lower price have higher priority.

■ Uniquely identify the order (OrderID + OrderDate make up a unique order for an instrument, in other words OrderID is unique during the date for an instrument).

■ Determine the price and size of an order.

3.4.3.3 Message Structure

Table 50 Order Update/Market Sheet Message Format describes the body fields of an Equities Order Book message, MsgType = '230' Order Update.

Table 50 Order Update/Market Sheet Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 bytes MsgSize field.
MsgType	2	2	Binary Int.	'230' – Order Book
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the time when the Order Update is generated. Note, when an order is added, the SourceTime represents the order entry time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	12	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
Price	16	4	Binary Int.	Price (to be calculated with the PriceScaleCode) *Provided when ActionType = A, M and Y*
AggregatedVolume	20	4	Binary Int.	Total interest quantity at a price point. *Provided when ActionType = A, M, D and Y*
Volume	24	4	Binary Int.	Remaining displayed quantity of the order. *Provided when ActionType = A, M and Y*
LinkID	28	4	Binary Int.	Not used.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
OrderID	32	4	Binary Int.	Identifies the uniqueness of the order, when combined with OrderDate. *Provided when ActionType = A, M, D and Y*
SystemID	36	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	40	2	Binary Int.	Number of micro seconds. To be combined with SourceTime.
NumberOrders	42	2	Binary Int.	Number of order at the current price point. *Provided when ActionType = A, M, D and Y*
Side	44	1	ASCII Ch.	Indicates the side of the order. Valid values are: ■ 'B' – Buy ■ 'S' – Sell ■ Null – Not provided
OrderType	45	1	ASCII Ch.	Type of Order. Valid values are: ■ '1' – Market order ■ '2' – Limit order ■ 'K' – Market to limit order ■ 'P' – Peg order
ActionType	46	1	ASCII Ch.	This field identifies why the volume (size) at the price point was modified. Values reserved for future use (US markets): 'O', 'C', 'E', 'X', 'Z'. Valid values are: ■ 'A' – New order ■ 'D' – Deletion of order identified by OrderID ■ 'F' – Deletion of all orders for the given instrument (depending on the side. If side is not provided, it means both) ■ 'M' – Modification of existing order without loss of priority ■ 'Y' – Retransmission of all orders for the given instrument
PriceScaleCode	47	1	Binary Int.	Applicable to all prices in the message. *Provided when ActionType = A, M and Y*

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
OrderDate	48	4	Binary Int.	Date of order (YYYYMMDD). To be combined with OrderId. *Provided when ActionType = A, M, D and Y*
OrderPriorityDate	52	4	Binary Int.	Date giving the priority of the order (YYYYMMDD). *Provided when ActionType = A, M, D and Y*
OrderPriorityTime	56	4	Binary Int.	Time giving the priority of the order (HHMMSSsss). *Provided when ActionType = A, M, D and Y*
OrderPriorityMicroSecs	60	2	Binary Int.	Number of microseconds in the current millisecond. *Provided when ActionType = A, M, D and Y*
Filler	62	2	Binary Int.	Filler to retain the structure of the message – possibly for future use.

3.4.4   Order Book Retransmission Delimiter – 231 Message

3.4.4.1   Message Overview

An Order Book Retransmission Delimiter message is set to clean and update the market sheet with the orders remaining in the order books from previous days.

3.4.4.2   Message Sending Rules

This message is sent at the beginning of the trading session to indicate the beginning and the end of the order book retransmission. A message is sent per instance of the trading engine (this is identified by the field InstanceID).

3.4.4.3   Message Structure

Table 51 Order Book Retransmission Delimiter Message Format describes the body fields of an Equities Order Book message, MsgType = '231' Order Book Retransmission Delimiter.

Table 51 Order Book Retransmission Delimiter Message Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body, excluding the 2 bytes MsgSize field.
MsgType	2	2	Binary Int.	'231' – Order book retransmission delimiter Message
SourceTime	4	4	Binary Int.	This field specifies the time when the Order Update is generated. Note, when an order is added, the SourceTime represents the order

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				entry time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
SourceSeqNum	8	4	Binary Int.	This field specifies the sequence number assigned by the source system to this message. Please note that while the sequence number increases serially, it does not increase monotonically.
TradingEngineID	12	2	ASCII Str.	Code identifying the trading engine. Valid values are: ■ 'C1' – UTP for Equities
InstanceID	14	1	Binary Int.	Indicates the number of instances for a given trading engine rebroadcasting order books.
RetransmissionIndicator	15	1	ASCII Ch.	Indicates the status of the retransmission for the instance of the trading engine: ■ 'B' – Beginning of the retransmission ■ 'E' – End of the retransmission

3.5   INDICES

3.5.1   Overview

List of Indices message types:

■ 542 – Real-Time Index

■ 543 – Index Summary

3.5.2   Packet Header Format

All messages are preceded by a common packet header format. **Table 52 Packet Header Format** describes the header fields of Indices messages.

Table 52 Packet Header Format

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
PacketLength	0	2	Binary Int.	Length of the packet including the 16-byte packet header.
PacketType	2	2	Binary Int.	Identifier for the type of data contained in the packet.

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				'501' – Market Data Packet
PacketSeqNum	4	4	Binary Int.	This field contains the Packet Sequence Number. It is unique for each broadcast stream (multicast group) and is used for gap detection. It increases serially and monotonically and is reset to 1 at the beginning of each trading day.
				The PackSeqNum is unique for packets containing market data only. Heartbeats inherit their sequence number from the last market data packet or Packet Sequence Number reset packet.
SendTime	8	4	Binary Int.	Timestamp in milliseconds indicating the packet broadcast time. The number represents the number of milliseconds since midnight of the last Sunday 00:00 UTC.
ServiceID	12	2	Binary Int.	Numeric value identifying the broadcast stream.
				Possible values are described in in Production Feed Configuration and External User Acceptance Feed Configuration.
DeliveryFlag	14	1	Binary Int.	Indicates the delivery method:
				■ '0' – Real-Time message (Uncompressed)
				■ '2' – Retransmission message (Uncompressed)
				■ '17' – Refresh message (zlib Compressed)
NumberMsgEntries	15	1	Binary Int.	The number of messages that are contained within the packet.

3.5.3 Real-Time Index – 542 Message

3.5.3.1 Message Overview

A Real-Time Index message handles the real-time characteristics of an index: the level of the index, type of index level (opening index level, real-time, indicative level), and various indicators for the instruments that make up the index. This message is sent for:

■ Stock Indices

■ Strategy Indices

■ Volatility Indices

■ Indicative Net Asset Value (iNAV) of an ETF

3.5.3.2 Message Sending Rules

Sending of these messages for a given index is conditioned by a flag configured at the index level. These conditions and the nature of these messages that are sent for each index are dependent on two factors:

■ The publication mode of the index; there are three publication modes for message 542:

Continuous – Calculated index levels are published periodically, at a frequency that can be configured for each index. Currently an index that is published continuously can either be published every 15 seconds or every 30 seconds.

Discontinuous – A single Closing level (level 5) before the provisional closing phase, occurring at a time (a 'fixed time') that can be configured for each index

At closing only – No broadcast before the provisional closing phase

■ The current calculation phase of the index

The following sections provide an overview of the different conditions at which an index level can be sent.

3.5.3.3   CAC 40 Index (Financial Market Code = 25)

At System Startup

The Closing level of the index of the previous trading day (level 5) is sent at the start of each trading day.

During the Trading Session

Opening Kinematics

Upon reception of the first trade price of any instrument that is part of the composition of the CAC 40, the index moves into the Session phase. If at this point 65% or more of the market cap of the index has traded, the Official Opening level (level A) is calculated and published. The Official Opening level is based on the last trade prices or the last-adjusted closing price if a last traded price is not available. Subsequently, real-time Session levels (level 2) are calculated and published every 15 seconds.

If, at the opening of the index, less than 65% of the market cap of the index has traded, an Automatic Indicative level (level 3) is published every 15 seconds following the opening of the market until at least 65% of the market cap of the index has traded. Once this threshold of 65% has been breached (and the index is not in the 'Indicative' phase), the Official Opening level (level A) is calculated and published. Subsequently, real-time Session levels (level 2) are calculated and published every 15 seconds.

For most other French indices, there are two thresholds that need to be breached for the index to send an official opening level:

At the opening of the market, at least 65% of the market cap needs to have traded. If at the opening of the market this threshold of 65% has not been reached, an Automatic indicative level (level 3) is sent every 15 seconds.

The Automatic indicative level continues to be sent every 15 seconds until a second threshold has been reached. For most French indices this second threshold is configured at 95% of the market cap. Once this second threshold has been reached, the Official Opening level (level A) is calculated and published. Subsequently, real-time Session levels (level 2) are calculated and published every 15 seconds.

If the index remains in an Automatic Indicative state the entire day, the last Automatic Indicative index level (level 3) is considered to be the official close. (There is no official opening level in this case.)

As soon as 100% of the market cap of the non regulated-halted constituents of the index has traded (and the index is not in 'indicative' state), the Reference level (level B) is calculated and broadcast. This level is calculated using only the opening (first trade) prices of its constituents.

Following the Opening

Once the Official Opening level (level A) has been published, the real-time Session levels (level 2) are calculated and published every 15 seconds.

In the case of an 'Indicative' Phase

The compiler can decide, following the opening of the index, to change the status of the index. This decision can be made if it is believed that circumstances prevent the proper calculation of the index. In this case, instead of the real-time Session level (level 2), an indicative level (level 0) is sent every 15 seconds. This level 0 is calculated by using the last-traded price or the last-adjusted closing price if a last traded price is not available.

The index levels that are calculated during the 'Indicative' status of an index are not taken into account to update the highest and the lowest levels of the index.

Once the compiler is sure that the index level is representative again, the real-time index levels (level 2) are calculated and published again every 15 seconds.

Options Liquidation Index (Level 7) for CAC 40

Definition / Purpose

- The liquidation index is used as a basis for the automatic exercise of options that are within the price range on their expiration date, as well as for the calculation of resulting payments.

- It is the average of the index level calculated every 15 seconds between 15:40 (CET) and 16:00 (CET). The result of the calculation is published every 15 seconds during the same time interval.

- This average is sent at each expiry date.

At the End of the Trading Day

After all instruments that are part of the composition of the index are closed, the index moves into the Temporary Closing phase. On a normal trading day, this occurs around 18:00:00 (CET). During this phase, the first Closing level (level 5), the first confirmation of the Reference level (level C) and the first Daily Summary message (message 543) are published. During the Temporary Closing phase, Euronext can make any necessary adjustments to the index if deemed necessary. The Temporary Closing phase currently lasts 5 minutes.

At the end of the Closing delay, the index moves into the Final Closing phase. The second Closing level, the second confirmation of the Reference level (level C) and the second Daily summary message (message 543) are published. Any adjustments that are made during the Temporary Closing Phase are taken into account in the second Closing level and the Daily Summary message.

The first and second Closing levels (level 5) are calculated based on the last trades of the instruments that take part of the composition of the index. This level represents the official Closing level of the CAC 40.

The confirmation of the Reference level (level C) is calculated using only the opening (first trade) prices of its constituents not taking into account any cancellation of opening trades.

3.5.3.4   AEX Index, BEL 20 Index and PSI 20 Index

At System Startup

The Closing level of the index of the previous trading day (level 5) is sent at the start of each trading day.

During the Trading Session

Opening Kinematics

Upon reception of the first trade price of any instrument that is part of the composition of the index, the index moves into the Session phase. If at this point 100% or more of the market cap of the index has traded, the Official Opening level (level A) is calculated and published. The Official Opening level is based on the last trade prices, including previous day, adjusted closing prices. Subsequently, real-time Session levels (level 2) are calculated and published every 15 seconds.

If by 9:05 the threshold of 100% is still not met, the threshold is dropped to 80% (second threshold). As soon as 80% of the market cap is available any time after 09:05, the Official Opening level (level A) is calculated and published followed by real-time Session levels (level 2).

From the opening of the index up until the first or second threshold is met, an Automatic Indicative level (level 3) is published every 15 seconds.

If the index remains in an Automatic Indicative state the entire day, the last Automatic Indicative index level (level 3) is considered to be the official close. (There is no official opening level in this case.)

Following the Opening

Once the Official Opening level (level A) has been published, the real-time Session levels (level 2) are calculated and published every 15 seconds.

In the case of an 'Indicative' Phase

The compiler can decide, following the opening of the index, to change the status of the index. This decision can be made if it is believed that circumstances prevent the proper calculation of the index. In this case, instead of the real-time Session level (level 2), an indicative level (level 0) is sent every 15 seconds. This level 0 is calculated by using the last-traded price or the last-adjusted closing price if a last-traded price is not available.

The index levels that are calculated during the 'Indicative' status of an index are not taken into account to update the highest and the lowest levels of the index.

Once the compiler is sure that the index level is representative again, the real-time index levels (level 2) are calculated and published again every 15 seconds.

Options Liquidation Index (Level 7) for AEX Index

Definition / Purpose

- The liquidation index is used as a basis for the automatic exercise of options that are within the price range on their expiration date, as well as for the calculation of resulting payments.

- It is the average of the index level calculated every 15 seconds between 15:30 (CET) and 16:00 (CET). The result of the calculation is published every minute during the same time interval.

At the End of the Trading Day

After all instruments that are part of the composition of the index are closed, the index moves into the Temporary Closing phase. On a normal trading day this occurs around 18:00:00 (CET). During this phase, the first Closing level (level 5) and the first Daily Summary message (message 543) are published. During the Temporary Closing phase, Euronext can make any necessary adjustments to the index if deemed necessary. The temporary Closing phase currently lasts 5 minutes.

At the end of the Closing delay, the index moves into the Final Closing phase. The second Closing level (level 5) and the second Daily Summary message (message 543) are published. Any adjustments that are made

during the Temporary Closing Phase are taken into account in the second closing level and the Daily Summary message.

The first and second Closing levels (level 5) are calculated based on the last trades of the instruments that take part of the composition of the index. This level represents the official Closing level of the index.

3.5.3.5   Message Structure

Table 53 Real-Time Index Message describes the body fields of an Equities Indices message, MsgType = '542' Real-Time Index.

Table 53 Real-Time Index Message

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body. MsgSize excluded.
MsgType	2	2	Binary Int.	'542' – Real-Time Index
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
IndexLevel	12	4	Binary Int.	The value of the last level for the index that is the subject of this message (to be calculated with the LevelScaleCode).
ForerunnerLevel	16	4	Binary Int. (signed)	No longer used.
SessionHigh	20	4	Binary Int.	Highest level of the day (to be calculated with the LevelScaleCode).
SessionLow	24	4	Binary Int.	Lowest level of the day (to be calculated with the LevelScaleCode).
PercentageOfCapitalization	28	4	Binary Int.	Percentage of capitalization for the active instruments in the index (to be calculated with the PercentageScaleCode).
VariationFromPreviousDaysPrice	32	4	Binary Int.	Percentage of variation for day's index / Previous day's reference (to be

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			(signed)	calculated with the VariationScaleCode).
SystemID	36	4	Binary Int.	The ID of the originating Exchange/System of the message. See System ID.
SourceTimeMicroSecs	40	2	Binary Int.	Number of micro seconds. To be combined with SourceTime.
NumOfSecuritiesQuoted	42	2	Binary Int.	Number of traded instruments in the index.
IndexLevelCode	44	1	ASCII Ch.	Index Level Code. Valid values are: ■ '0' – Indicative index ■ '2' – Real-Time index ■ '3' – Automatic indicative index ■ '5' – Closing Reference index ■ '7' – Options liquidation index ■ 'A' – Official Opening index ■ 'B' – (Preliminary) Reference index ■ 'C' – (Confirmed) Reference index
TypeOfLevel	45	1	ASCII Ch	■ '1' – Price level
LevelScaleCode	46	1	Binary Int.	To be combined with IndexLevel, SessionHigh, SessionLow.
PercentageScaleCode	47	1	Binary Int.	To be combined with PercentageOfCapitalization.
VariationScaleCode	48	1	Binary Int.	To be combined with VariationFromPreviousDaysPrice.
RebroadcastIndicator	49	1	Binary Int.	For most indices, the closing level of the previous trading day is sent at the beginning of the trading day. This indicator should be used to validate if this message applies to the previous day's or current day's index level. Valid values are: ■ '0' – Current Day ■ '1' – Previous Day
Filler	50	2	ASCII Str	Filler to retain the structure of the

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
				message – possibly for future use.

3.5.4   Index Summary – 543 Message

3.5.4.1   Message Overview

The index summary message provides a summary of the data calculated in a stock index for a given day – the opening level, the closing level, the high and the low that were published for the index.

3.5.4.2   Message Sending Rules

This message is sent:

- Every trading day, for each index of the type 'stock index' two types of index summary messages are sent (this rules out iNAVs):

The first is sent when the index enters the provisional closing phase

The second is sent when the index enters the final closing phase

3.5.4.3   Message Structure

Table 54 Index Summary describes the body fields of an Equities Indices message, MsgType = '543' Index summary.

Table 54 Index Summary

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
MsgSize	0	2	Binary Int.	Length of the message body. MsgSize excluded.
MsgType	2	2	Binary Int.	'543' – Index Summary
SymbolIndex	4	4	Binary Int.	XDP proprietary identification of the instrument.
SourceTime	8	4	Binary Int.	This field specifies the message generation time. The number in this field represents the number of milliseconds since midnight of the same day. For example, if SourceTime = 13:12:56 secs, 170ms and 30 microsecs, this field will contain 47576170.
PreliminaryOpeningLevel	12	4	Binary Int.	This field will no longer be filled.
PreliminaryOpeningTime	16	4	Binary Int.	This field will no longer be filled.
OpeningLevel	20	4	Binary	Official Opening Index Level. This level

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	corresponds to the IndexLevelCode A of the message 542 of the corresponding index (to be calculated with the LevelScaleCode).
OpeningTime	24	4	Binary Int.	Time of Official Opening level. Number of milliseconds since midnight.
ConfirmedReferenceLevel	28	4	Binary Int..	(Confirmed) Reference level. This level corresponds to the IndexLevelCode C of the message 542 of the corresponding index (to be calculated with the LevelScaleCode).
ConfirmedReferenceTime	32	4	Binary Int.	Time of (Confirmed) Reference level. Number of milliseconds since midnight.
ClosingReferenceLevel	36	4	Binary Int.	Reference closing index level. This level corresponds to the IndexLevelCode 5 of the message 542 of the corresponding index (to be calculated with the LevelScaleCode).
ClosingReferenceTime	40	4	Binary Int.	Time of provisional closing reference index level. Number of milliseconds since midnight.
PercentVariationPrevClos	44	4	Binary Int. (signed)	Percentage of variation for last index level versus. previous day's closing (to be calculated with the VariationScaleCode).
HighLevel	48	4	Binary Int.	Highest index level (to be calculated with the LevelScaleCode).
HighTime	52	4	Binary Int.	Time of provisional highest index level. Number of milliseconds since midnight.
LowLevel	56	4	Binary Int.	Lowest index level (to be calculated with the LevelScaleCode).
LowTime	60	4	Binary Int.	Time of provisional lowest index level. Number of milliseconds since midnight.
ClearingLevel	64	4	Binary Int.	This field will no longer be filled.
ClearingTime	68	4	Binary Int.	This field will no longer be filled.
LiquidationLevel	72	4	Binary Int.	Reference at-expiration settlement index level (to be calculated with the LevelScaleCode).
LiquidationTime	76	4	Binary Int.	Time of provisional expiation settlement index level. Number of milliseconds since midnight.
SystemID	80	4	Binary	The ID of the originating Exchange/System of

FIELD NAME	OFFSET (BYTES)	SIZE (BYTES)	FORMAT	DESCRIPTION
			Int.	the message. See System ID.
SourceTimeMicroSecs	84	2	Binary Int.	Number of microseconds. To be combined with SourceTime.
TypeOfLevel	86	1	ASCII Ch	■ '1' -Price level
LevelScaleCode	87	1	Binary Int.	To be combined with all level fields.
VariationScaleCode	88	1	Binary Int.	To be combined with PercentVariationPrevClose.
Filler	89	3	ASCII Str	Filler to retain the structure of the message – possibly for future use.

4. PRODUCTION FEED CONFIGURATION

4.1 INTRODUCTION

4.1.1 Data Content

The information supplied in this chapter applies to the XDP feed for Euronext European Cash Markets services only.

4.1.2 Data Delivery

XDP is available via the SFTI network.

4.2 RENDEZVOUS POINT (ALL CONNECTIVITY)

Table 55 Production Platform Rendezvous Point

RP1 PLATFORM LINE A	RP2 LINE B
S: 156.48.121.2 Associated to: G: 224.0.52.x	S: 156.48.121.55 Associated to: G: 224.0.53.x

4.3 PRODUCTION FEED CONFIGURATION

Configuration details are provided in the *Euronext European Cash Markets Data Feed Configuration* document. This document includes primary and secondary data centre source IPs, and multicast group/port details for the real-time and refresh feeds.

4.4 MESSAGE TYPE, BANDWIDTH AND MARKET PLACES PER SERVICE ID

Table 56 Production Message Type, Bandwidth and Market Places Per Service ID describes the message types and bandwidths in the production environment.

Table 56 Production Message Type, Bandwidth and Market Places Per Service ID

MESSAGE TYPE	ENXT EQUITIES REF.	ENXT EQUITIES TRADES	ENXT EQUITIES QUOTES	EXNT EQUITIES ORDERS	ENXT WARRANTS TRADES	ENXT WARRANTS QUOTES	INDICES	EUROPEAN STOCKS REF.	EUROPEAN STOCKS TRADES	LUXEMBOURG SE	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	101	102	103	104	105	106	107	108	109	110	119	301	302	303
140 – Quotes			•			•				•				
146 – RMF Quotes											•			
141 – WAS			•											
221 – Trade – Cancel		•			•					•		•	•	•
230 – Order Update				•						•		•	•	•
231 – Order Retransmission				•						•		•	•	•
240 – Trade – Full information		•			•					•		•	•	•
241 – Trade – Price update		•			•					•		•	•	•
242 – Trade – TCS trade		•			•							•	•	•
243 – Trade – Trade publication									•					

MESSAGE TYPE	ENXT EQUITIES REF.	ENXT EQUITIES TRADES	ENXT EQUITIES QUOTES	EXNT EQUITIES ORDERS	ENXT WARRANTS TRADES	ENXT WARRANTS QUOTES	INDICES	EUROPEAN STOCKS REF.	EUROPEAN STOCKS TRADES	LUXEMBOURG SE	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	101	102	103	104	105	106	107	108	109	110	119	301	302	303
244 – Trade – Settlement price		•										•	•	•
245 – Trade – Auction summary		•			•					•		•	•	•
505 – Stock state change		•			•					•		•	•	•
513 – Euro & Interbank rates		•								•		•	•	•
516 – Class State Change		•			•					•		•	•	•
523 – Mail		•			•					•		•	•	•
524 – Request for Size		•										•	•	•
530 – Indicative Matching Price		•			•					•		•	•	•
534 – Authorized out of session limits		•			•					•		•	•	•
535 – TCS State change		•										•	•	•
537 – Threshold		•			•					•		•	•	•
539 – Session timetable		•			•					•		•	•	•

MESSAGE TYPE	ENXT EQUITIES REF. 101	ENXT EQUITIES TRADES 102	ENXT EQUITIES QUOTES 103	EXNT EQUITIES ORDERS 104	ENXT WARRANTS TRADES 105	ENXT WARRANTS QUOTES 106	INDICES 107	EUROPEAN STOCKS REF. 108	EUROPEAN STOCKS TRADES 109	LUXEMBOURG SE 110	RETAIL MATCHING FACILITY QUOTES 119	SERVICE A (ORDERS AND TRADES) 301	SERVICE B (ORDERS AND TRADES) 302	SERVICE C (ORDERS AND TRADES) 303
542 – Real-Time index							•			•				
543 – Index Summary							•			•				
550 – Start Reference Data	•				•		•	•		•				
551 – End Reference Data	•				•		•	•		•				
553 – Reference Data	•				•		•	•		•				
Bandwidth Size (Mb)	1	6	24	44	3	10	2	0.25	0.25	1	6	75	50	40
Population *	CAC	CAC	CAC	CAC	EW	EW	EI	ES	ES	LU	CAC	CAC	BC	EE
	BC	BC	BC	BC					CAC		BC			
	EE	EE	EE	EE					BC		EE			
									EE					

* Population acronym:

■ CAC: CAC40

■ BC: Blue Chips

- EE: Euronext Equities

- EW: Euronext Warrants

- EI: Euronext Indices

- ES: European Stocks

- LU: Luxembourg instruments

Bandwidth size (KB) explanation: each XDP multicast channel capacity has been sized / optimized to cope with the expected amount of data to be disseminated on this channel and avoid any pick, overrun, bottleneck or packet loss.

4.5　REFRESH MESSAGE TYPE, BANDWIDTH AND MARKET PLACES PER SERVICE ID

Table 57 Production Refresh Message Type, Bandwidth and Markets Places Per Service ID describes the Service ID where the refresh functionality will be available in the production environment.

Table 57 Production Refresh Message Type, Bandwidth and Markets Places Per Service ID

MESSAGE TYPE	ENXT EQUITIES REF.	ENXT EQUITIES TRADES	ENXT EQUITIES QUOTES	EXNT EQUITIES ORDERS	ENXT WARRANTS TRADES	ENXT WARRANTS QUOTES	INDICES	EUROPEAN STOCKS REF.	EUROPEAN STOCKS TRADES	LUXEMBOURG SE	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	201	202	203	204	205	206	207	208	209	210	219	401	402	403
140 – Quotes			•			•				•				
146 – RMF Quotes											•			
141 – WAS														

MESSAGE TYPE	ENXT EQUITIES REF.	ENXT EQUITIES TRADES	ENXT EQUITIES QUOTES	EXNT EQUITIES ORDERS	ENXT WARRANTS TRADES	ENXT WARRANTS QUOTES	INDICES	EUROPEAN STOCKS REF.	EUROPEAN STOCKS TRADES	LUXEMBOURG SE	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	201	202	203	204	205	206	207	208	209	210	219	401	402	403
221 – Trade – Cancel		●			●					●		●	●	●
230 – Order Update				●						●		●	●	●
231 – Order Retransmission										●				
240 – Trade – Full information		●			●					●		●	●	●
241 – Trade – Price update		●			●					●		●	●	●
242 – Trade – TCS trade		●			●							●	●	●
243 – Trade – Trade publication									●					
244 – Trade – Settlement price		●										●	●	●
245 – Trade – Auction summary		●			●					●		●	●	●
505 – Stock state change message		●			●					●		●	●	●
513 – Euro & Interbank rates		●								●		●	●	●
516 – Class State Change		●			●					●		●	●	●

MESSAGE TYPE	ENXT EQUITIES REF.	ENXT EQUITIES TRADES	ENXT EQUITIES QUOTES	EXNT EQUITIES ORDERS	ENXT WARRANTS TRADES	ENXT WARRANTS QUOTES	INDICES	EUROPEAN STOCKS REF.	EUROPEAN STOCKS TRADES	LUXEMBOURG SE	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	201	202	203	204	205	206	207	208	209	210	219	401	402	403
530 – Indicative Matching Price		•			•					•		•	•	•
534 – Authorized Out of Session Limits		•			•					•		•	•	•
535 – TCS State Change		•								•		•	•	•
537 – Threshold		•			•					•		•	•	•
539 – Session timetable		•			•					•		•	•	•
542 – Real-Time index							•			•				
543 – Index Summary							•			•				
550 – Start Reference Data	•				•		•	•		•				
551 – End Reference Data	•				•		•	•		•				
553 – Reference Data	•				•		•	•		•				
Bandwidth Size (Mb)	0.5	1	1	3	1	1	0.25	0.25	0.125	0.5	512 Kb	3	2	2
Population	CAC	CAC	CAC	CAC	EW	EW	EI	ES	ES	LU	CAC	CAC	BC	EE

MESSAGE TYPE	ENXT EQUITIES REF.	ENXT EQUITIES TRADES	ENXT EQUITIES QUOTES	EXNT EQUITIES ORDERS	ENXT WARRANTS TRADES	ENXT WARRANTS QUOTES	INDICES	EUROPEAN STOCKS REF.	EUROPEAN STOCKS TRADES	LUXEMBOURG SE	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	201	202	203	204	205	206	207	208	209	210	219	401	402	403
	BC	BC	BC	BC					CAC		BC			
	EE	EE	EE	EE					BC		EE			
									EE					

4.6   PRODUCTION TIMETABLE

Table 58 Production Timetable is an overview of the events during a trading day which impact market data activity. Customers should also reference the trading manual / 516 Class State message for details of all trading phases.

Table 58 Production Timetable

EVENT	TIME (CET)	COMMENT
Application start-up	~ 06:10	
Reference Data Sent	~ 06:11	
Order Book Retransmission	~ 06:20	Retransmission of outstanding orders and associated messages from previous day.
Open (Structured Products)	~ 08:00	Applies to Warrants, Certificates and Structured Notes.
Open (all European markets)	~ 09:00	
Close (Paris, Amsterdam, Brussels equities/warrants)	17:40	To include closing auction and TAL phases.
Close (Luxembourg)	17:40	
Close (Structured Products)	18:30	Applies to Warrants, Certificates and Structured Notes.
Close (all indices)	~ 20:00	Manual action
Reference Data Sent	Between 20:00 and 21:00	Manual action
Application close down	23:00	**It should be noted that the close down of the application depends on the dissemination of the end of day Reference Data. The end of day Reference Data is mandatory, therefore If it is delayed for any reason, the application close down will be delayed until it is disseminated to customers.** Customers can use the 551 End Reference Data message to know there will be no more market data disseminated for a trading day.

4.7   RETRANSMISSION AND REFRESH CONFIGURATION

TCP/IP configuration details are provided in the *Euronext European Cash Markets Data Feed Configuration* document. This document includes primary and secondary data centre IP address/port details for the retransmission and refresh services.

4.7.1   High Availability Retransmission Behaviour

In production there are two redundant Retransmission Servers in the exchange. Clients should monitor the connection to the Retransmission Server to determine if there is an outage, and have the ability to switch to connect to the 'secondary' Retransmission Server in the event of a failover. The secondary Retransmission Server will maintain the same cache of packets as the primary therefore providing redundancy. Clients should monitor the availability of the primary / secondary Retransmissions Server by the following means:

■   For clients remaining connected to the Retransmission Server throughout the day, a disconnection from the Retransmission Server should trigger a failover to the secondary Retransmissions Server.

■   Clients may choose to only connect to the Retransmissions Server if the application requires packets to be serviced. In this instance, clients should fail over to the secondary Retransmission Server if they cannot establish a connection with the primary.

4.7.2   High Availability Refresh Behaviour

In production there are two redundant Refresh Servers in the exchange. Clients should monitor the TCP/IP connection to the Refresh Server to determine if there is an outage, and have the ability to switch to connect to the 'secondary' Refresh Server in the event of a failover. The secondary Refresh Server will maintain the same cache of packets as the primary therefore providing redundancy. Clients should monitor the availability of the primary / secondary Refresh Server by the following means:

■   For clients remaining connected to the Refresh Server throughout the day, a disconnection from the Refresh Server should trigger a failover to the secondary server.

It should be noted that if a disconnection is caused for any other reason than by an outage with the Refresh Server itself, a client sending a request to the secondary will be rejected with rejection code 7. For example, this can be due to a client side network disconnection. In this instance, clients should reconnect to the primary.

■   Clients may choose to only connect to the Refresh Server if the application requires packets to be serviced. In this instance, clients should fail over to the secondary Refresh Server if they cannot establish a connection with the primary.

Clients should note that unlike the secondary Retransmission Server, the secondary Refresh Server will not service requests while it is in secondary mode. Clients sending a request to the secondary Refresh Server when there is no outage and the primary server is available, will receive a response with rejection code 7 – Refresh request rejected as sent to incorrect server (secondary instead of primary).

4.7.3   Source ID

The Source ID allows clients to perform retransmission and refresh requests. Please note that the Source IDs for retransmissions and refresh are identical. Euronext will provide a default of four Source IDs for Production. If more Source IDs are required please contact the European Service Desk:

Email: CAS@euronext.com

Tel: +33 (0) 1 8514 8589

Clients should clearly state for which environment (EUA or Production) a Source ID is requested.

Each Source ID may only be logged on to a server once at a given time.

4.8　RETRANSMISSION REQUEST LIMITATIONS

The recommendations below apply to production and EUA.

4.8.1　Heartbeat Mechanism

The heartbeat message frequency is set to 30 seconds.

A heartbeat message response has to be sent within 5 seconds to stay connected to the server

4.8.2　Number of Source IDs

Clients are provided with four Source IDs by default. In the event more Source IDs are required, please contact the Service Desk.

4.8.3　Parallel Sessions

Clients may file several concurrent requests on the server at the same time with the same Source ID; there is no need to wait for the active retransmission to be closed to ask for another one.

Responses to these requests are sent in the same order as the initial requests.

4.8.4　Maximum Number of Requests

Maximum number of Retransmission Requests per Source ID per day has been set at 10,000.

A defensive feature of the Retransmission Server is that it will not respond to clients' requests after 200 invalid requests are sent per Source ID, per day. An invalid request is defined as any of the descriptions 1-6 in the RejectReason field of the retransmission response message. Client applications will remain connected and continue to receive heartbeats, however requests for retransmission will receive no response (accepted or rejected). This defensive feature is designed to stop badly-behaved client applications from impacting the performance of the service for other participants.

4.8.5　Maximum Number of Packets per Request

Maximum number of packets that can be requested in one Retransmission Request has been set at 1,000.

4.8.6　Maximum Number of Packets Stored in the Retransmission Cache

The maximum number of packets that are cached for retransmission request has been set at 500,000. The following table shows the number of packets per Service ID. A packet out of cache cannot be retransmitted (a dedicated response "Rejected – requested packets are not available" will be sent in this case).

Table 59 Retransmission Cache per Service ID

SERVICE ID	RETRANSMISSION CACHE - PACKETS
101	50,000
102	200,000
103	500,000
104	500,000
105	100,000
106	500,000
107	20,000
108	5,000

SERVICE ID	RETRANSMISSION CACHE - PACKETS
109	5,000
110	200,000
119	5,000
301	300,000
302	300,000
303	300,000

4.9 REFRESH REQUEST LIMITATIONS

The recommendations below apply to production and EUA.

4.9.1 Heartbeat Mechanism

The heartbeat messages frequency is set to 30 seconds.

A heartbeat message response has to be sent within 5 seconds to stay connected to the server.

4.9.2 Number of Source IDs

The Source IDs for the refresh are identical to the Retransmissions Server. They should be reused with the Refresh Server.

Clients are provided with four Source IDs by default. In the event more Source IDs are required, please contact the Service Desk.

4.9.3 Maximum Number of Requests

Maximum number of Refresh Requests per Source ID per day has been set at 1,000.

A defensive feature of the Refresh Server is that it will not respond to clients requests after 100 invalid requests are sent per Source ID, per day. An invalid request is defined as any of the descriptions 1-7 in the RejectReason field of the refresh response message. Client applications will remain connected and continue to receive heartbeats, however requests for refresh will receive no response (accepted or rejected). This defensive feature is designed to stop badly behaved client applications from impacting the performance of the service for other participants.

5. EXTERNAL USER ACCEPTANCE FEED CONFIGURATION

5.1 INTRODUCTION

5.1.1 Data Content

The information supplied in this chapter applies to the XDP feed for Euronext European Cash Markets External User Acceptance (EUA) service only.

5.1.2 Data Delivery

XDP EUA is available via the SFTI network.

5.2 RENDEZVOUS POINT (ALL CONNECTIVITY)

Table 60 Rendezvous Point External User Acceptance Platform

RP1 PLATFORM LINE A	RP2 LINE B
S: 156.48.121.2 **Associated to: G: 224.0.52.x**	S: 156.48.121.55 Associated to: G: 224.0.53.x

5.3 SFTI FEED CONFIGURATION

Configuration details are provided in the *Euronext European Cash Markets Data Feed Configuration* document. This document includes primary and secondary data centre source IPs, and multicast group/port details for the real-time and refresh feeds.

5.4 MESSAGE TYPE, BANDWIDTH AND MARKET PLACES PER SERVICE ID

Table 61 EUA Message Type, Bandwidth and Markets Places Per Service ID describes the message types and bandwidths in the EUA environment.

Table 61 EUA Message Type, Bandwidth and Markets Places Per Service ID

MESSAGE TYPE	EXNT EQUITIES ORDERS	ENXT EQUITIES	ENXT WARRANTS	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	1	2	3	10	501	502	503
140 – Quotes message		●	●				
146 – RMF Quotes				●			
141 – WAS message		●					
221 – Trade message – Cancel		●	●		●	●	●
230 – Order Update	●				●	●	●
231 – Order Retransmission	●				●	●	●

MESSAGE TYPE	EXNT EQUITIES ORDERS	ENXT EQUITIES	ENXT WARRANTS	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	1	2	3	10	501	502	503
240 – Trade Message – Full information		●	●		●	●	●
241 – Trade Message – Price update		●	●		●	●	●
242 – Trade Message – TCS trade		●	●				
243 – Trade Message – Trade publication		●					
244 – Trade Message – Settlement price		●			●	●	●
245 – Trade Message – Auction summary		●	●		●	●	●
505 – Stock state change message		●	●		●	●	●
513 – Euro & Interbank rates		●			●	●	●
516 – Class State Change		●	●		●	●	●
523 – Mail		●	●		●	●	●
524 – Request for Size		●			●	●	●
530 – Indicative Matching Price		●	●		●	●	●
534 – Authorized Out of Session Limits		●	●		●	●	●
535 – TCS State Change		●			●	●	●
537 – Threshold		●	●		●	●	●
539 – Session timetable		●	●		●	●	●
542 – Real-Time index		●					
543 – Index Summary		●					
550 – Start Reference Data		●	●				
551 – End Reference Data		●	●				
553 – Reference Data		●	●				
Bandwidth Size (Mb)	0.25	0.75	0.125	512 Kb	512 Kb	512 Kb	512 Kb
Population	CAC	CAC	EW	CAC	CAC	BC	EE
	BC	BC		BC			

MESSAGE TYPE	EXNT EQUITIES ORDERS	ENXT EQUITIES	ENXT WARRANTS	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	1	2	3	10	501	502	503
	EE	EE		EE			
	LU	EI					
		ES					
		LU					

* Population acronym:

■ CAC: CAC40

■ BC: Blue Chips

■ EE: Euronext Equities

■ EW: Euronext Warrants

■ EI: Euronext Indices

■ ES: European Stocks

■ LU: Luxembourg Instruments

Bandwidth size (KB) explanation: each XDP multicast channel capacity has been sized / optimized to cope with the expected amount of data to be disseminated on this channel and avoid any pick, overrun, bottleneck or packet loss.

5.5   REFRESH MESSAGE TYPE, BANDWIDTH AND MARKETS PLACES PER SERVICE ID

Table 62 EUA Refresh Message Type, Bandwidth and Markets Places Per Service ID describes the Service ID where the refresh functionality will be available in the EUA environment.

Table 62 EUA Refresh Message Type, Bandwidth and Markets Places Per Service ID

MESSAGE TYPE	EXNT EQUITIES ORDERS	ENXT EQUITIES	ENXT WARRANTS	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	11	12	13	20	601	602	603
140 – Quotes message		•	•				

MESSAGE TYPE	EXNT EQUITIES ORDERS	ENXT EQUITIES	ENXT WARRANTS	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	11	12	13	20	601	602	603
146 – RMF Quotes				•			
141 – WAS message							
221 – Trade message – Cancel		•	•		•	•	•
230 – Order Update	•				•	•	•
231 – Order Retransmission							
240 – Trade Message – Full information		•	•		•	•	•
241 – Trade Message – Price update		•	•		•	•	•
242 – Trade Message – TCS trade		•	•		•	•	•
243 – Trade Message – Trade publication		•					
244 – Trade Message – Settlement price		•			•	•	•
245 – Trade Message – Auction summary		•	•		•	•	•
505 – Stock state change message		•	•		•	•	•
513 – Euro & Interbank rates		•			•	•	•
516 – Class State Change		•	•		•	•	•
530 – Indicative Matching Price		•	•		•	•	•
534 – Authorized Out of Session Limits		•	•		•	•	•
535 – TCS State Change		•			•	•	•
537 – Threshold		•	•		•	•	•
539 – Session timetable		•	•		•	•	•
542 – Real-Time index		•					
543 – Index Summary		•					
550 – Start Reference Data		•	•				
551 – End Reference Data		•	•				
553 – Reference Data		•	•				
Bandwidth Size (Mb)	0.2	0.7	0.125	256 Kb	512 Kb	512 Kb	512 Kb

MESSAGE TYPE	EXNT EQUITIES ORDERS	ENXT EQUITIES	ENXT WARRANTS	RETAIL MATCHING FACILITY QUOTES	SERVICE A (ORDERS AND TRADES)	SERVICE B (ORDERS AND TRADES)	SERVICE C (ORDERS AND TRADES)
	11	12	13	20	601	602	603
Population	CAC	CAC	EW	CAC	CAC	BC	EE
	BC	BC		BC			
	EE	EE		EE			
	LU	EI					
		ES					
		LU					

5.6   EUA TIMETABLE

The starting time for the XDP EUA feed is ~ 10 minutes delayed in comparison to the production feed.

The closing time for the XDP EUA feed varies daily depending on which testing is scheduled (afternoon upgrades, after hours testing, and so forth).

5.7   RETRANSMISSION AND REFRESH CONFIGURATION

TCP/IP configuration details are provided in the *Euronext European Cash Markets Data Feed Configuration* document. This document includes primary and secondary data centre IP address/port details for the retransmission and refresh services.

5.7.1   High Availability Retransmission Behaviour

In EUA there are two redundant Retransmission Servers in the exchange. Clients should monitor the connection to the Retransmission Server to determine if there is an outage and have the ability to switch to connect to the 'secondary' Retransmission Server in the event of a failover. The secondary Retransmission Server will maintain the same cache of packets as the primary, therefore providing redundancy. Clients should monitor the availability of the primary and secondary Retransmission Servers by the following means:

■   For clients remaining connected to the Retransmission Server throughout the day, a disconnection from the Retransmission Server should trigger a failover to the secondary Retransmission Server.

■   Clients may choose to connect only to the Retransmission Server if the application requires packets to be serviced. In this instance, clients should fail over to the secondary Retransmission Server if they cannot establish a connection with the primary.

5.7.2   High Availability Refresh Behaviour

In EUA there are two redundant Refresh Servers in the exchange. Clients should monitor the TCP/IP connection to the Refresh Server to determine if there is an outage and have the ability to switch to connect to the 'secondary' Refresh Server in the event of a failover. The secondary Refresh Server will

maintain the same cache of packets as the primary therefore providing redundancy. Clients should monitor the availability of the primary and secondary Refresh Server by the following means:

■   For clients remaining connected to the Refresh Server throughout the day, a disconnection from the Refresh Server should trigger a failover to the secondary server.

It should be noted that if a disconnection is caused for any other reason than by an outage with the Refresh Server itself, a client sending a request to the secondary will be rejected with rejection code 7. For example, this can be due to a client-side network disconnection. In this instance, clients should reconnect to the primary.

■   Clients may choose to only connect to the Refresh Server if the application requires packets to be serviced. In this instance, clients should fail over to the secondary Refresh Server if they cannot establish a connection with the primary.

Clients should note that unlike the secondary Retransmission Server, the secondary Refresh Server will not service requests while it is in secondary mode. Clients sending a request to the secondary Refresh Server when there is no outage and the primary server is available, will receive a response with rejection code 7 – Refresh request rejected as sent to incorrect server (secondary instead of primary).

5.7.3   Source ID

The Source ID allows clients to perform retransmission and refresh requests. Please note that the Source IDs for retransmissions and refresh are identical. Euronext will provide a default of four Source IDs for EUA. If more Source IDs are required please contact the European Service Desk:

Email: CAS@euronext.com

Tel: +33 (0) 1 8514 8589

Clients should clearly state for which environment (EUA or Production) a Source ID is requested.

Each Source ID may only be logged on to a server once at a given time.

APPENDIX A: MARKET FEED CODE

MARKET	CODE	CONTENT
	00	None Applicable
	CC	Technical message
	61	Volatility Indices
	62	Strategic indices
	63	Net Return Indices
	64	Gross Return Indices
	66	Commodities Indices
	01	Equity Indices
Euronext Paris	02	CAC40 stocks that do not support traded options.
	03	Non-CAC40 stocks that support traded options.
	04	Non-CAC40 stocks that do not support traded options.
	05	Non-CAC40 stocks that support traded options (same as 03)
	06	Non-CAC40 stocks that do not support traded options.
	07	Bonds.
	10	Free market ("Marché libre")
	11	Securities in process of being introduced (SHIVA).
	12	Gold.
	13	Foreign exchange.
	15	Centralized lending market
	16	Commercial paper
	60	iNAV (Indicative Net Asset Value of ETFs)
	20	Stock listed in Brussels
Euronext Brussels	21	Bonds listed in Brussels
	22	Brussels Index constituents
	23	Warrants listed in Brussels
	30	Stocks listed in Amsterdam
Euronext Amsterdam	31	Bonds listed in Amsterdam
	32	Amsterdam Index constituents
	33	Warrants listed in Amsterdam

MARKET	CODE	CONTENT
	40	Equities, Rights and Investment Funds listed in Lisbon
Euronext Lisbon	41	Bonds and Certificates listed in Lisbon
	42	PSI-20 equities listed in Lisbon
	43	Warrants listed in Lisbon
	18	Equities and GDR
Euronext London	19	Bonds and Bonds convertible / exchangeable
	14	Warrants
	50	Stocks listed in BDL
Luxembourg	51	Bonds listed in BDL
	52	Corporate Bonds listed in BDL
	53	Warrants listed in BDL (bilateral clearing / settlement)
	54	Indices listed in BDL
	55	OPCs listed in BDL (bilateral clearing / settlement)
	56	Bonds listed in BDL (bilateral clearing / settlement)
	57	Stocks listed in BDL (European regulated market)
European Instruments	65	European stocks (Trade publication purpose)

APPENDIX B: FINANCIAL MARKET CODE

CODE	CONTENT
000	None applicable
025	Paris
277	Centralized lending market
278	Brussels
279	Amsterdam
290	Lisbon
292	London
295	Luxembourg
299	Europe
300	Commercial paper

APPENDIX C: STOCK EXCHANGE CODE

CODE	CONTENT
000	Not applicable
006	BRUXELLES
025	PARIS
027	LYON
028	MARSEILLE
029	NANCY
030	BORDEAUX
031	NANTES
032	LILLE
038	AMSTERDAM
047	LUXEMBOURG
051	LISBON
726	LONDON
991	EUROPEAN STOCK

APPENDIX D: STOCK TYPE

CODE	STOCK TYPE
008	Participating bond-Cum-warrant
009	Participating bond-Ex-warrant
010	Participating share
011	Subordinated bond
013	Interest strip
014	Principal strip
015	Perpetual
017	Bunny bond
018	ORT (France)
019	OAT Fungible government bond
021	Convertible bond
022	Exchangeable bond
023	Participating bond
024	Indexed bond
025	Ordinary bond or note
026	Lottery bond
027	Savings bond
028	Indemnity bond
030	Bond warrant
032	Bond Cum-warrant
033	Bond Ex-warrant
035	Right to indemnity security
038	Bond option
039	Emprunt notionnel (France)
040	Founder's share
041	Ordinary share
042	Bonus share
043	Preferred share
044	Saving share

CODE	STOCK TYPE
045	Certificate
046	AFV share (Belgium)
047	Accumulating right
048	Allotment right
049	Subscription right
050	Preferred stock
051	VVPR share (Belgium)
052	Bons de caisse
053	Cash note
054	Allocation right
055	Option right
056	Share-Cum warrant
057	Share-Ex warrant
058	Preference share
059	Preference
060	Gold
061	Unit of international investment trust
062	Unit of unit trust
063	Mortgage warrant
064	Bank note
065	Devise commerciale (France)
066	Devise titre (France)
067	Commodities
068	Index
069	Unit
070	Investment fund share
071	Miscellaneous products-Warrant
072	Share warrant
075	Miscellaneous
076	Listed call option

CODE	STOCK TYPE
077	Listed put option
078	Call money, average BD rate
080	Founder's stock
081	Partnership interest
082	Part de réserve (Belgium)
084	Deferred share
085	Regional development company share
086	Venture capital company
087	Real estate company share
090	Mortgage bond
100	Participation certificate
105	Index warrant
106	Currency warrant
108	Warrant of a warrant
110	Participating share-Warrant
111	Subordinated convertible bond
121	Convertible bond-Cum warrant
123	Convertible participating bond
139	Matif 90 days treasury bond
140	Part de réserve AFV (Belgium)
141	Convertible ordinary share
142	Dividend right certificate
144	Convertible saving share
145	Investment certificate-Warrant
150	Convertible preference share
162	MBO share
200	Participative certificate-Warrant
221	Convertible bond-Ex warrant
240	AFV company's share (Belgium)
241	VVPR's company's share (Belgium)

CODE	STOCK TYPE
242	Accumulation fund share
243	Distribution fund share
244	Unit-Futures and Options market investment fund
245	Accumulation fund share
246	Distribution fund share
247	Certificate of guaranteed value
248	Share warrant
249	BTAN (France)
250	OAT (France)-Interest certificate
251	OAT (France)-Principal certificate
252	Indexed OAT (France)
253	Indexed OAT (France)-Principal certificate
254	Indexed OAT (France)-Interest certificate
255	Euro medium term note-EMTN
256	BTF (France)
257	Indexed certificate
258	BMTN (France)
259	Real estate bond
260	Convertible EMTN
261	Indexed EMTN
262	Indexed certificate-EMTN
263	Exchange Traded fund-ETF
264	Venture Cap. mutual fund share
265	Mutual fund for innovation share
266	Medium Term Note
267	Medium Term Note-Floating rate
268	Accumulating share
269	Distribution share
270	Ordinary bond-Interest certificate
271	Ordinary bond-Principal certificate

CODE	STOCK TYPE
272	'Beneficial interest' share
273	Redeemable cumulative preferred share
274	Convertible redeemable cumulative preferred share
275	Convertible cumulative preferred share
276	Cumulative preferred share
277	Redeemable preferred share
278	Foreign treasury note
279	Subordinated Euro medium term note
280	Strip VVPR
281	Mortgage Bonds
303	Exotic warrant
304	Certificate of deposit
305	Commercial paper
306	ETF – Closed ended Fund
307	ETF – Open ended Fund
308	SICAV Action
309	Preferred securities

APPENDIX E: SYSTEM ID

ID	APPLICATION
1	System ID not provided
11 to 19	UTP for Equities – Regulated markets
31 to 39	UTP for Warrants

APPENDIX F: CHANGE HISTORY – PREVIOUS VERSIONS

F.1   VERSION 2.0

F.1.1   Document Style and Structure

The most significant change of the version 2.0 specification was the act of merging all prior 'appendices' and configuration documents into one specification document. The table below shows the one-to-one mapping between the old documents and the chapters within version 2.0.

Please note that chapter 2 in the version 2.0 specification applies to all sections of chapter 3. This was included in separate appendices before, but is now only documented once.

OLD DOCUMENT NAME	CHAPTER IN NEW SPECIFICATION
Generic Customer Specifications 1.3	1 Euronext Cash Markets Processing Information
Trade Appendix 1.4	3.2 Trades
Quotes & BBO10 Appendix 1.4	3.3 Quotes and BBO10
Order Book Appendix 1.4	3.4 Order Book
Indices Appendix 1.4	3.5 Indices
Market Information Appendix 1.4	3.1 Market Information
Production Feed Configuration 1.6	4 Production Feed Configuration
External User Acceptance Feed Configuration 1.6	5 External User Acceptance Feed Configuration
XDP Production Services Content	4.4.1 – SFTI
XDP EUA Services Content	5.3.2.1 – SFTI
XDP Notice Of Interest (SmartPool Members)	3.1.11 Notice Of Interest – 246 Message

The change history from the previous 1.x documentation has been removed. If this is required, please contact the Service Desk. Certain textual changes have been made in order to improve the descriptions in sections of the document. Where this did not impact anything other than the cosmetics of the specifications, it has not been highlighted.

F.1.2   Refresh Functionality

The following chapters have been added or edited to define the XDP refresh functionality.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
1.2.3	Refresh Functionality (Key Principles)	EUA: 28th September 2009 PRD: 26th October 2009
1.3.2, 2.2, 2.10.1, 3.1.2, 3.2.2, 3.3.2, 3.4.2, 3.5.2	Inclusion of DeliveryFlag 16 for Real-Time ZliB Compressed packets	EUA: 28th September 2009 PRD: 26th October 2009

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
2.4.1 / 2.4.2	Refresh Heartbeat Processing Notes	EUA: 28th September 2009 PRD: 26th October 2009
2.8	Refresh Request	EUA: 28th September 2009 PRD: 26th October 2009
2.9	Refresh Response	EUA: 28th September 2009 PRD: 26th October 2009
2.10.1	Refresh Compression	EUA: 28th September 2009 PRD: 26th October 2009
2.10.2	Refresh Packet Type	EUA: 28th September 2009 PRD: 26th October 2009
2.10.3	Start Refresh Message	EUA: 28th September 2009 PRD: 26th October 2009
2.10.4	End Refresh Message	EUA: 28th September 2009 PRD: 26th October 2009
4.3.1.2	Production Refresh Feed Channel Definitions	EUA: 28th September 2009 PRD: 26th October 2009
4.5	Refresh Contents (production)	EUA: 28th September 2009 PRD: 26th October 2009
4.7.2	Production Refresh TCP/IP Settings	EUA: 28th September 2009 PRD: 26th October 2009
4.7.4	Production High Availability Refresh Behaviour	EUA: 28th September 2009 PRD: 26th October 2009
4.7.5 / 5.7.5	Refresh Source ID (Production and EUA) – Same as retransmissions Source ID.	EUA: 28th September 2009 PRD: 26th October 2009
4.9	Refresh Request Limitations	EUA: 28th September 2009 PRD: 26th October 2009
5.3.1.2	EUA Refresh Feed Channel Definitions	EUA: 28th September 2009 PRD: 26th October 2009
5.7.2	EUA Refresh TCP/IP Settings	EUA: 28th September 2009 PRD: 26th October 2009
5.7.4	EUA High Availability Refresh Behaviour	EUA: 28th September 2009

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
		PRD: 26th October 2009
5.5	Refresh Contents (EUA)	EUA: 28th September 2009 PRD: 26th October 2009

A release schedule for refresh will be defined imminently.

F.1.3 Channel Definitions

Channel definitions have been added to the specification for the following.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
4.3.1	Production Disaster Recovery Source IP's (note 'SS' in the definition table)	EUA: 28th September 2009 PRD: 26th October 2009
4.3.1	Production Refresh Feed Channel Definitions (as per above note)	EUA: 28th September 2009 PRD: 26th October 2009
5.3.1	EUA Refresh Feed Channel Definitions (as per above note)	EUA: 28th September 2009 PRD: 26th October 2009
4.7.1	Disaster Recovery Retransmissions IP details	Already live in production. New information provided.

F.1.4 Message Changes

The following changes have been made to specific messages throughout the specification. All of these changes are already live on the data feed.

DOCUMENT CHAPTER	MESSAGE NUMBER	DETAILS OF CHANGE	IMPACT / LIVE DATE DETAILS
1.5.2, 2.2, 3.1.2, 3.2.2, 3.3.2, 3.4.2, 3.5.2	Packet Header	The FAST compression delivery flags 8 and 10 have been removed from these sections	Removal of Information (N / A for live date)
3.1.3.3	505	Additional values 0 and null added to the InstrumentState field of the 505 message	Already live in production
3.1.3.3	505	Additional value null added to the InstrumentTradingStatus field of the 505 message	Already live in production
3.1.3.3	505	In the InstrumentTradingStatus field, values S and R added for Indices	Already live in production
3.1.5.3	516	Changed the name of the field Phase to	Cosmetic Change

DOCUMENT CHAPTER	MESSAGE NUMBER	DETAILS OF CHANGE	IMPACT / LIVE DATE DETAILS
		SessionType	
3.1.12.3	539	Changed the name of the field Phase to SessionType	Cosmetic Change
3.1.17.3	553	Additional value of 9 added to the TaxCode field. Values 1, 2 and 4 removed.	Already live in production
3.1.17.3	553	The StrikeScaleCode field has been defined in position 455 of the 553 message. This was previously defined as a filler which was an error in the previous specifications.	Already live in production
3.2.3.3	220	Changed the name of the field Phase to OpeningTradeIndicator	Cosmetic Change
3.2.5.3	240	Changed the name of the field Phase to OpeningTradeIndicator	Cosmetic Change
3.4.4.3	231	Removal of the value VE from the TradingEngineID field	Already live in production
3.4.4.3	231	Addition of the value CO to the TradingEngineID field	Already live in production
3.5.3.3	542	The values in the TypeOfLevel field for levels 2 and 3 were inverted in the previous specifications. This has now been corrected	Already live in production
3.5.4.3	543	The values in the TypeOfLevel field for levels 2 and 3 were inverted in the previous specifications. This has now been corrected	Already live in production

F.1.5   Miscellaneous Changes

The following miscellaneous changes have been made throughout the specification.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
1.3.4.3	Improved diagram to describe retransmission behaviour	For information purposes.
1.4.1.2	New section added for the High Availability System Failure	Already live in production. New information provided.
1.4.1.3	New section added for Disaster Recovery System Failure	Already live in production. New information provided.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
4.6	New section added to describe the production feed timetable	For information purposes.
5.6	New section added to describe the EUA feed timetable	For information purposes.
Appendix D	Addition of two new ETF Stock Types	Already live in production. New information provided.
Appendix E	New appendix added to describe the SystemID. The SystemID field in every message now refers to this appendix	For information purposes.

F.1.6   Message Overview and Message Sending Rules

To improve our documentation the 'Message Overview' and 'Message Sending Rules' sections have been added to each of the defined message types. This should be reviewed for added value and information.

F.1.7   Messages Not Live – For Information

The following messages Will no longer be delivered.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
–	Message 531 'Market Imbalance'	
–	Message 541 'Daily Summary'	

 Version 2.1

F.1.8   Refresh Server Updates

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
1.3.2, 2.2, 2.10.1, 3.1.2, 3.2.2, 3.3.2, 3.4.2, 3.5.2	Change to DeliveryFlag packet header values of refresh based on client feedback.	EUA: 28th September 2009 PRD: 26th October 2009
2.9	New refresh response added (7) to cater for clients sending refresh requests to the secondary server.	EUA: 28th September 2009 PRD: 26th October 2009
4.3.1.2 / 5.3.1.2	Added production and EUA multicast channel definitions for the Refresh Server	EUA: 28th September 2009 PRD: 26th October 2009
4.7.2 / 5.7.2	Added production and EUA TCP/IP channel definitions for Refresh Server	EUA: 28th September 2009 PRD: 26th October 2009

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
4.3.1.2 / 5.3.1.2	Added Luxembourg to the production and EUA multicast channel definitions for the Refresh Server	EUA: 28th September 2009 PRD: 26th October 2009
4.5 / 5.5	Added Luxembourg to production and EUA refresh contents table Addition of 516 messages in channel 102 refresh	EUA: 28th September 2009 PRD: 26th October 2009
4.7.4 / 5.7.4	Added Refresh Server High Availability behaviour	EUA: 28th September 2009 PRD: 26th October 2009
5.3.1.2	Added NAE to EUA refresh configuration, and updated EUA refresh service IDs.	EUA: 28th September 2009 PRD: 26th October 2009

F.1.9   Miscellaneous Changes

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
3.1.5.3	Addition of value 'CLSD' to the ClassState field of the 0 Class State Change message	PRD: 21st September 2009.
3.4.3.2	Additional information added to the message sending rules of the 230 message	Additional information provided.
3.5.3.3	Addition of a new field in the 542 message 'RebroadcastIndicator'. This field DOES NOT affect the previous message structure and uses one of three characters previously occupied by a filler. The default value of this field is currently 0. The date this field will be activated will be provided via the XDP Issue List.	The date this field will be activated will be provided via the XDP Issue List.
5.3.1.1	IP range details provided for EUA instead of dedicated IP addresses. This is consistent with production.	Already live. New information provided.
5.8	Removed the EUA Internet VPN details as this service has been discontinued.	EUA VPN service has already been terminated.

F.1.10 Warrants and SmartPool XDP Migrations

The following changes have been made to the specification in anticipation of both Warrants and SmartPool being migrated to XDP.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
3.1.3.3	Addition of new value 'R' to the HaltReason field in the 505 message Stock State Change.	XDP Warrants live on 22nd February 2010
3.1.3.3	Addition of new values 'Y' and 'Z' to the ActionAffectingState field in the 505 message Stock State Change.	XDP Warrants live on 22nd February 2010
Appendix E	Addition of XDP for SmartPool system IDs.	XDP Warrants live on 22nd February 2010

F.1.11 Removal of NYSE Arca Europe

In light of NYSE Arca Europe migrating to the London data centre, and an independent market data publisher, a separate specification has been created for this now fully separate market. It should be noted that NYSE Arca Europe EUA is still accessed via the Paris Euronext Cash EUA environment. For that reason NYSE Arca Europe configuration details for EUA have remained in this specification.

The following has been removed from this document.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
1.5.1	Removed NAE channel 114 from the environments section	Cosmetic documentation update
3.1.17.3	Removed reference to NAE in the following fields in the reference data where this information was noted as not being relevant to NAE: 'LastAdjPrice' / 'PrevVolumeTraded' / 'DateOfLastTrade'	Cosmetic documentation update
3.1.17.3	Removal of 'XHFT NYSE ARCA EUROPE' from the 'MIC' field in the reference data	Field value no longer provided
3.4.4.3	Removal of value C0 for UTP NAE in the 'TradingEngineID' field	Field value no longer provided
4.3.1.1	Removal of channel 114 for NAE from production channel definitions	This channel is now defined in the new NAE specification
4.4.1 / 4.4.2	Removal of NAE channel 114 for SFTI production	This channel is now defined in the new NAE specification
4.5	Removed reference to NAE in the production refresh contents	This channel is now defined in the new NAE specification

F.2 VERSION 2.2

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
4.4.1 , 4.4.2	2010 Bandwidth Increases.	PROD : 18 January 2010
3.1.17	New value '4' added to the RepoIndicator field, and a change of meaning for the value '2'.	PROD : 22 February 2010 (to coincide with Warrants migration to XDP)
3.1.11.2	Addition of sending rules for the 537 Collars message relating to instruments traded in the new Warrants Market Model.	EUA: Currently available PROD: 22 January 2010 (to coincide with Warrants migration to XDP)
2.2 , 2.10.2 , 3.1.2 , 3.2.2 , 3.3.2 , 3.4.2, 3.5.2	The concept of multiple market data packet types has been removed from the feed, in the PacketType field of the packet header. This has been completed in order to increase performance. A single PacketType of 501 will be used for all market data packets. Technical packets (as defined in PacketType field in chapter 2.2) will continue to have unique PacketType values.	EUA: 24 February 2010 PROD: 10 May 2010
3.5.3.3	Addition of a new field in the 542 message 'RebroadcastIndicator'. This field DOES NOT affect the previous message structure and uses one of three characters previously occupied by a filler. The default value of this field is currently 0. The date this field will be activated will be provided at a later date.	EUA: 24 February2010 PROD: 29 March 2010
3.3.3.3	Addition of BBO display guidelines to clarify for members, ISVs and data vendors the correct approach using the XDP feed.	Additional Information Provided.
1.2.2 , 4.7.4 , 4.8.4 , 4.9.3 , 5.7.4	Additional information provided to describe the retransmission and refresh services.	Additional information Provided.
3.4.3.3	Removal of ActionType 'P' – this value is no longer used. Additional information provided regarding field content.	No impact for clients
3.1.17.3	Conversion of IndustryText field into a filler for future use. This field has never been used.	No impact for clients

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
3.1.6	Due to client demand the 523 'Mail' message will continue to be supported in the feed (this was planned to be removed previously.	No impact for clients
1.5.1 , 5.3.1.1	Removal of NYSE Arca Europe EUA details from this specification. This detail is documented in the dedicated NYSE Arca Europe XDP specification.	Cosmetic change to documentation only
3.2.3 , 3.2.5 , 3.4.3	QuoteLinkID field definitions made consistent as 'Not Used'.	Cosmetic change to documentation only

F.3 VERSION 2.3

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
2.2 , 2.10.2 , 3.1.2 , 3.2.2 , 3.3.2 , 3.4.2, 3.5.2	The Market Data Packet Type has been carried over as a change from document version 2.2 to version 2.3, as this has not yet gone live in Production in the time between the two released specification versions. The concept of multiple market data packet types has been removed from the feed, in the PacketType field of the packet header. This has been completed in order to increase performance. A single PacketType of 501 will be used for all market data packets. Technical packets (as defined in PacketType field in chapter 2.2) will continue to have unique PacketType values.	EUA: 24 February 2010 PROD: 10 May 2010
4 – Production Feed Configuration, 5 – External User Acceptance Feed Configuration	Network changes for the Data Centre migration of the Euronext Cash Markets to Basildon, London.	The final connectivity details for the Basildon migration are given in the Basildon Data Centre Migration Addendum. Multicast Source IP Addresses will change along with TCP connectivity details.

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
3.5.3.1, Appendix A	542 Real-Time Index messages now sent for Volatility Indices on the XDP feed. New Market Feed Code added for Volatility Indices	EUA: 19 April 2010 PROD: 17 May 2010
3.1.14.3, Appendix A, Appendix B, Appendix C	Addition of Euronext London technical details. New Market Feed Codes, Financial Markets Code and Stock Exchange code added, and new MIC Code defined in the Reference Data.	EUA: Currently Available PROD: Production live dates will be communicated in a dedicated Info Flash regarding the Euronext London project.
4.3.1.2, 4.5, 5.3.1.2, 5.5	Refresh services completed to include ALL channels.	EUA: 14 April 2010 PROD: 7 June 2010
5.5	Bandwidth Adjustments as per the new SFTI order form release.	PROD: 7 June 2010
3.1.3.3, 3.1.5.3, 3.1.10.3, 3.2.3.3, 3.2.5.3, 3.4.3.3	Removal of NSC values for fields which prior to the complete XDP migration, could contain either an NSC value or a XDP value. These details have been displayed in red with a strike through to demonstrate their removal. They will be removed from all future documentation.	XDP for Equities and Warrants is now live in Production. These values are no longer sent in the feed.
Multiple Chapters Affected	Removal of SmartPool from this specification. SmartPool now has its own dedicated market data specification. Please refer to the dedicated SmartPool specification for details of specific SmartPool changes.	Removal of Information. Customers using SmartPool should refer to the new SmartPool specification.
3.1.13	Removal of the Display Bid and Ask Message. This message is no longer sent since the discontinuation of the NSC trading engine.	Message is no longer sent.

F.4   VERSION 2.4

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
4 Production Feed Configuration, section 4.4	Update bandwidth size to increase service channels 103 & 104 (+4Mb each), and to decrease service channel 106 (-4Mb)	PROD: 6 September 2010

F.5   VERSION 2.5

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
4 Production Feed Configuration, section 4.4	Update bandwidth size to increase service channels 103 & 104 (+3Mb each)	PROD: 6 December 2010

F.6   VERSION 2.6

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 1	– Additions to Section 1.3.2 (Packet Structure) – Additions to Section 1.3.4.4 (Refresh Server) – Additions to Section 1.5.8 (Instrument identifiers) – Addition of Section 1.5.9 (How to Determine a Closing Price)	PROD: 28 March 2011
Chapter 2, Section 2.7	Added Table 11 (Retransmission Response Message Header Format)	
Chapter 3, Section 3.1.3	Added Tables 21, 22 and 23 to Stock State Change – 505 Message	
Chapter 3, Section 3.1.5	Added Table 27 (Class State Change – 516 Message)	
Chapter 3, Section 3.3.3	– Added notes on message structure (Quotes – 140 Message) – Additional information added to QuoteNumber field description in Table 47 (Quotes Message Format)	
Chapter 3, Section 3.4.3	Additional information added to Order Update / Market Sheet – 230 Message	
Chapter 4 and 5	– Moved IP address information from Production and EUA Feed Configuration chapters to separate document – Updated the 'Bandwidth Size (Mb)' field from 19 to 24 in Table 56 (Production Message Type, Bandwidth and Market Places Per Service ID)	
Appendix D, Stock Type	Updated with codes 279 and 308	

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Throughout	Format changes and minor edits made throughout and artwork updated	
Chapter 3, Section 3.1.14	Removed the 'OfficialQuotationList' field and replaced with a 'Filler' field in Table 36 (Reference Data Message Format)	PROD: 16 May 2011

F.7   VERSION 2.7

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3	Added values of 'N' and '7' to the 'TradeCond3' field of the 240 message (Trade Full Information)	PROD: 27 June 2011

F.8   VERSION 2.8

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3	– Updated the description of the 'Opening TradeIndicator' field in Table 38 Trade Creation Message Structure and Table 40 Trade Full Information Structure. – Clarified that a value of Null (or space) is valid for the ActionAffectingState field in Table 24 Stock State Change Message Format. – Removed obsolete value of 2 (Opening order) from the 'TypeOfAskPrice' and 'TypeOfBidPrice' fields in Table 47 Quotes Message Format.	N/A (changes already reflected in the feed)
Chapter 4	Increased bandwidth size from 1 to 2 for Index Service ID channel 107 in Table 57 Production Message Type, Bandwidth and Market Places Per Service ID.	PROD: 11 July, 2011

F.9   VERSION 2.9

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS

CHAPTER ADDED / EDITED	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Throughout	Replaced references to "UTP-MD" with "XDP"	PROD: November, 2011
Chapters 1 & 2	Added clarifications that the Packet Sequence Reset packet is not available on the Retransmission Server.	
Chapter 3	– Added values of 'XLUX' and 'EMTF' to the 'MIC' field in Table 36 Reference Data Message Format – Updated description of the 'HaltReason' value 'C' field in the Stock State Change message (505) – Added description of "Behaviour when an instrument is halted" to Stock State Change message (505) section (3.1.3)	
Chapter 4	– Updated Table 58 Production Refresh Message Type, Bandwidth and Markets Places Per Service ID with message types 542, 543 and 544 checked for the Luxembourg SE. – Added "Close (Luxembourg) 17.40" to Table 59 Production Timetable – Updated bandwidth size for msg 104 in Table 57 Production Message Type, Bandwidth and Market Places Per Service ID – Updated the maximum number of Retransmission and Refresh Requests per Source ID per day to 10,000 and 1000 respectively	
Appendix A	Added Luxembourg to the table of Market Feed Codes	
Appendix D	Added code 309 (Preferred securities)	

F.10   DOCUMENT HISTORY SUMMARY - VERSIONS 2.0-2.9

VERSION NO.	DATE	CHANGE DESCRIPTION
2.0	Aug 2008	Initial distribution

VERSION NO.	DATE	CHANGE DESCRIPTION
2.1	Sep 2009	– Refresh server updates – Warrants and SmartPool XDP changes – Removal of NYSE Arca Europe
2.2	Jan 2010	– New specification template – Addition of the 'Market Data Packet Type' – Addition of the RebroadcastIndicator to the 542 message – Addition of the BBO Guidelines
2.3	Apr 2010	– Addition of the 'Market Data Packet Type' – Volatility Indices added to the feed – Addition of refresh channels to support Warrants, Indices and Euro Stocks – Other changes including the London Gateway, removal of NSC values and the removal of SmartPool from the specifications.
2.4	Jul 2010	Update to reflect bandwidth size increase, see Table 57 Production Message Type, Bandwidth and Market Places Per Service ID.
2.5	Nov 2010	Further update to Table 57 Production Message Type, Bandwidth and Market Places Per Service ID.
2.6	Mar 2011	Multiple sections updated: for full details see Version 2.6.
2.7	Apr 2011	Added values of 'N' and '7' to the 'TradeCond3' field in Table 40 Trade Full Information Structure.
2.8	July 2011	– Updated the description of the 'Opening TradeIndicator' field in Table 38 Trade Creation Message Structure and Table 40 Trade Full Information Structure. – Clarified that a value of Null (or space) is valid for the ActionAffectingState field in Table 24 Stock State Change Message Format. – Removed obsolete value of 2 (Opening order) from the 'TypeOfAskPrice' and 'TypeOfBidPrice' fields in Table 47 Quotes Message Format. Increased bandwidth size from 1 to 2 for Index Service ID channel 107 in Table 57 Production Message Type, Bandwidth and Market Places Per Service ID.
2.9	Nov 2011	Replaced references to "UTP-MD" with "XDP" Added clarifications that the Packet Sequence Reset packet is not available on the Retransmission Server. Added values of 'XLUX' and 'EMTF' to the 'MIC' field in Table 36 Reference Data Message Format

VERSION NO.	DATE	CHANGE DESCRIPTION
		Updated description of the 'HaltReason' value 'C' field in the Stock State Change message (505)
		Added description of "Behaviour when an instrument is halted" to Stock State Change message (505) section (3.1.3)
		Updated Table 58 Production Refresh Message Type, Bandwidth and Markets Places Per Service ID with message types 542, 543 and 544 checked for the Luxembourg SE.
		Added "Close (Luxembourg) 17.40" to Table 59 Production Timetable
		Updated bandwidth size for msg 104 in Table 57 Production Message Type, Bandwidth and Market Places Per Service ID
		Updated the maximum number of Retransmission and Refresh Requests per Source ID per day to 10,000 and 1000 respectively
		Added Luxembourg to the table of Market Feed Codes
		Added code 309 (Preferred securities)

F.11   VERSION 3.0

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Dec 2011)	- Updated the 'Message Sending Rules' sub-section of Section 3.5.3 (Real-Time Index – 542 Message) - Updated the 'Message Sending Rules' sub-section of Section 3.5.4 (Index Summary – 543 Message) - Updated the 'Message Sending Rules' sub-section of Section 3.5.5 (Index Composition – 544 Message) - Updated the descriptions of the following fields in Table 53 (Real-Time Index Message): ForerunnerLevel, IndexLevelCode, TypeOfLevel, LevelScaleCode, RebroadcasterIndicator - Updated the descriptions of the following field in Table 54 (Index Summary): TypeOfLevel - Updated Appendix A (Market Feed Code) with codes 63, 64 and 66; also updated the	EUA: Available now PROD: March 2012

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
	description of codes 01 and 60	

F.12   VERSION 3.1

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Jan 2012)	Updated Section 3.5.3.3 (CAC 40 Index) – Updated Table 53 (Real-Time Index Message) – Updated Table 54 (Index Summary)	PROD: March 2012

F.13   VERSION 3.2

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Apr 2012)	Updated Section 3.1.3.2 (Stock State Change – 505 Message) – Updated description before and after Table 27 (Class States) and the table itself. – Removed obsolete values from the TypeOfPrice field in Table 41 (Price Update Message Format) – Updated artwork throughout	Available now
Chapter 4 (Apr 2012)	Increased Channel 104 bandwidth to 44 Mbps.	16 April

F.14   VERSION 3.3

CHAPTER ADDED / EDITED (MODIFICATION DATE)	Description	Impact / Live Date Details
Chapter 2 (Apr 2012)	Added value of '7' to the RejectReason field in Table 12 (Retransmission Response Message)	EUA: Available now PROD: TBC
Chapter 3 (Apr 2012)	Corrected the note in the description of the OpeningTradeIndicator field in messages 220 (Trade Creation) and 240 (Trade Full Information) to read "O" instead of "0"	Available now

CHAPTER ADDED / EDITED (MODIFICATION DATE)	Description	Impact / Live Date Details
	(zero).	
	Replaced the value of '3' with the value of '5' in the description of the 'TypeOfUnitExp' field in the 553 Reference Data message.	EUA: Available now PROD (End of Day Referential) : May 25

F.15   VERSION 3.4

CHAPTER ADDED / EDITED (MODIFICATION DATE)	Description	Impact / Live Date Details
Chapter 3 (Jul 2012)	– Added value of '43' to the TypeOfPrice field in the Price Update 241 Message – Added 'Transaction Type Indicator' field to TCS Trade (242 Message) and Trade Publication (243 Message)	TBC (expected Q4 2012)
Throughout (Jul 2012)	Rebranded in new colour scheme.	Available now

F.16   VERSION 3.5

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Sep 2012)	Corrected description of Collars message in Section 3.1.10.1	Available now
	Removed Trade Creation message 220	

F.17   VERSION 3.6

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Sep 2012)	Added value of 'R' to 'TradeCond3' field in the Trade Full Information message (240) and added 146 message (Retail Matching Facility Quotes)	Available now
Chapters 4 & 5 (Sep 2012)	Added details IP configuration details for 146 message (Retail Matching Facility Quotes)	

F.18   VERSION 3.7

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Oct 2012)	Changed the description of value '3' in the RepoIndicator field in Table 36 (Reference Data Message Format).	Available now
Appendix A (Oct 2012)	Added Luxembourg code 54 (Indices)	

F.19   VERSION 3.8

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Jan 2013)	Added the value '5' – Instrument eligible for the Loan and Lending Market only in the RepoIndicator field in Table 36 (Reference Data Message Format).	Live on EUA testing on 18th January 2013, Production Live Effective date TBC

F.20   VERSION 3.9

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapters 1, 4 & 5 (Feb 2013)	- Added Service A, Service B and Service C and packet limits in the Retransmission Cache. - Added Service A, Service B and Service C to tables 3, 4, 57, 58, 62 and 63 and updated population details. Added table 60 to Section 4.8.6.	Available Q2 2013

F.21   VERSION 3.10

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 and 4 (Mar 2013)	3.2.8.2 added SRD, 3.3.3.4 corrected typo 4.4 Table 57 Bandwidth increased for Retail Matching Facility	

F.22   VERSION 3.11

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Mar 2013)	Updated Transaction Type Indicator in 3.2.6.1 (Msg Type 242) and 3.2.7.3 (Msg Type 243) to add ' ' (space) – Not defined and format for the field changed from Binary Int to ASCII Ch.	

F.23   VERSION 3.12

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Apr 2013)	Updated Section 3.1.14 Reference Data-553 Message Added a description on Multi-Listed instruments to provide clarification on how those are identified in the feed.	

F.24   VERSION 3.13

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
(Apr 2013)	Removal of message type 513	

F.25   VERSION 3.14

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Jul 2013)	Addition of Type 22 in message 553 In section 3.1.3.2 message types 505 and 516 changed around to correct order.	

F.26   VERSION 3.15

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3 (Nov 2013)	MSG Type 241 Added the following value in the TypeOfPrice field: '04' - Net Asset Value (+/-) for the instruments eligible to the NAV Trading Facility.	

F.27   VERSION 3.16

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.5.5 (Dec 2013)	Chapter 3.5.5 Index Composition – 544 Message removed	3rd March 2014
Chapter 3.1.3.3 (Dec 2013)	Added 'K' as new value for 'Halt Reason' Added the values 'B', 'E', 'I' and 'T' as new values for 'Halt Reason' Added Table 24 for the possible values for warrants and certificates for trading after knock out (TAKO) and for the Enhancement of Knock-In and Knock-out Management.	Q2 2014

F.28   VERSION 3.17

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 4.4, Section 3.2.6.3 and Section 3.2.5.2 (Apr 2014)	– Increased Channel 106 bandwidth to 10 Mbps – Increased Channel 301 bandwidth to 75 Mbps – Increased Channel 302 bandwidth to 50 Mbps In section: 3.2.6.3: changed description for Trade Type: 'I' from Investment Fund to NAV Trading Facility In Section: 3.2.5.2 removed obselete type of price from Message Sending Rules: 05, 06, 30, 31, 32, 41, 42	7th April 2014

F.29   VERSION 3.18

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3, Section 3.1.3.4. (Mar 2014)	Amended the InstrumentState field with correct wording.	14th March 2014

F.30   VERSION 3.19

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.2.6 (Apr 2014)	Addition of TradeType=Y (Exchange for Physical) to the TCS Trade – 242 Message.	Q1 2015

F.31   VERSION 3.20

CHAPTER ADDED / EDITED (MODIFICATION DATE)	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.6 (Apr 2014)	Addition of the new message Request For Size Message – 524 message	Q1 2015

F.32   VERSION 3.21

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Global	Euronext Technologies	Rebranded to Euronext Template	

F.33   VERSION 3.22

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.14 (Jul 2014)	Euronext Technologies	Addition of MIC Code "VPXB" with description - NYSE EURONEXT - VENTES PUBLIQUES BRUSSELS to the field MIC in the Reference	

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
		Data – 553 Message.	

F.34   VERSION 3.23

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.6 (Oct 2014)	Euronext Technologies	- Modification of Request For Size – 524 message. 3 new fields added (System ID, SourcetimemicroSecs and filler). - Addition of new message type 524 and its corresponding channels to sections Production Feed Configuration and External User Acceptance Feed Configuration. - Message type 524 will not be included in the Refresh service.	

F.35   VERSION 3.24

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Appendix D (Apr 2015)	Euronext Technologies	Modification of APPENDIX D: Stock Type value 52 - Bons de caisse	

F.36   VERSION 3.25

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.3, 3.1.7, 3.3.3, 3.3.4, 3.1.14 (May 2015)	Euronext Technologies	- Correction of possible values of field OrderEntryRejection to only keep Y and N as possible values in Stock State Change – 505	

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
		Message. - Modification of Indicative Matching Price – 530 Message to add to new fields **ImbalanceVolumeSide** and **ImbalanceVolume**. - Clarification of Market Summary information in Quotes – 140 Message and Retail Matching Facility (RMF) Quotes – 146 Message. - Field format and value updates to Reference Data – 553 Message (**TypeOfUnitExp** and **MarketIndicator** fields).	

F.37   VERSION 3.26

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.7 (Jul 2015)	Euronext Technologies	- Modification of Indicative Matching Price – 530 Message to add a filler between **ImbalanceVolumeSide** and **ImbalanceVolume**.	

F.38   VERSION 3.27A

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.14, 3.2.5, 3.2.6, 4.5, 5.5 (Mar 2016)	BSA & TCH BA Team - Euronext Technologies	- Modification of reference Data – 553 Message to: * add a field **MaxDecimalPlacesInQuantity** added * update the format of the field **TypeOfUnitExp** to Binary Int. - Modification of Price Update – 241 Message to: * Add 2 new possible values to the field **TypeOfPrice** * Update the definition of the value '04' of the	Q1 2016

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
		field **TypeOfPrice** ("NAV Trading Facility" replaced by "Euronext Fund service") - Modification of TCS Trade – 242 Message to: * Update the definition of the value I in the field **TradeType** ("NAV Trading * Facility replaced by "Euronext Fund Service") * Update the size of the field **Volume** * Update the size of the field **CumulativeNumberSecurities** * Add a new field **VolumeScaleCode** * Add a filler at the end of the message * Update the general offset positions - Modification of Price Update – 241 Message sending rules: * No Price Update message is sent when the market operation cancels a trade which modifies the last price * A Price Update message is sent when the market operation resends Key Market Data - Removing of Mail – 523 Message from the Refresh channels in: * Table 57 Production Refresh Message Type, Bandwidth and Markets Places Per Service ID * Table 62 EUA Refresh Message Type, Bandwidth and Markets Places Per Service ID - Modification of TCS Trade – 242 Message to: * Add a filler (size = 4) between **Price** and **Volume** * Add a filler (size = 4) between **CumulativeCapital** and **CumulativeNumberSecurities** * Update the general offset position based on the previous mentioned modifications.	

F.39   VERSION 3.3.0

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.14, 3.2.4, 3.2.5, 4.4, 4.5, 5.4, and 5.5	BSA BA Team - Euronext Technologies	- Modification of reference Data – 553 Message to provide more information in existing fillers, without impact on the message length and structure. The following new fields have been added: * **GuaranteeIndicator** * **ParValue** * **ParValueScaleCode** * **MaturityDate** * **SettlementDelay** * **DarkEligibility** * **DarkMinimumQuantity** * **DarkLISThreshold** - Modification of Trade Full Information – 240 Message to add the following new fields with a change to the length of the message: * **TradingDate** * **TradingTime** * **TradingTimeMicroSecs** * **TransparencyInd** - Modification of Price Update – 241 Message to include a new value for the existing field **TypeOfPrice** - Modification of Order Update – 230 Message: definition of value **'M'** updated (field **ActionType**) - Modification of 'Message Type' sections 4.4, 4.5, 5.4, and 5.5: Value **'EW'** removed from channel 109, 209, 2 and 12	Q4 2016

F.40   VERSION 3.4.0

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.14	BSA BA Team - Euronext Technologies	- Clarification of the definition of the following fields: * **GuaranteeIndicator** * **DarkEligibility** * **DarkMinimumQuantity** * **DarkLISThreshold**	Q4 2016

F.41   VERSION 3.4.1

CHAPTER ADDED / EDITED (MODIFICATION DATE)	AUTHOR	DESCRIPTION	IMPACT / LIVE DATE DETAILS
Chapter 3.1.14	TCH BA Team - Euronext Technologies	- Addition of value '00030' for the **DepositaryList** field of reference Data – 553 Message.	