Document title

STREAM WEBSOCKET

Document type or subject

TECHNICAL SPECIFICATION

Version Number
2.1.0

Date
20 May 2026

Number of pages
39

PREFACE

WHAT'S NEW?

The following lists only the most recent modification made to this revision/version.

Version Number	Date	Change Description
1.0.0	06/11/25	Creation of document.
1.1.0	13/01/26	Addition of WebSocket response example.

TABLE OF CONTENTS

1. INTRODUCTION

Modern financial markets operate at lightning speed, and timely access to accurate data is critical for decision-making, trading strategies, and risk management. Our Euronext Stream WebSocket Service for Real-Time Market Data is designed to deliver high-frequency updates with optimized latency, ensuring that clients receive the most current market information as it happens.

Unlike traditional polling or REST-based APIs, WebSocket technology enables a persistent, bidirectional connection between your application and our servers. This allows for instantaneous data delivery, reducing overhead and improving efficiency for applications that require continuous updates—such as portfolio monitoring tools, risk management and analytics dashboards.
This manual provides a comprehensive guide to integrating and using the streaming service, including:

- Connection setup and authentication.
- Subscription management for instruments and data types.
- Message formats and schema details.

By following this guide, you will be able to seamlessly incorporate real-time market data into your workflows, enabling faster insights and more informed decisions.

2. ACCESS TO THE SERVICE

The WebSocket endpoint is available through the public Internet:

- wss://euronext-md.eu.diffusion.cloud

To establish connection and use the service, clients are provided with a username and a password by Euronext.

3. USE OF THE SERVICE

3.1 ORGANIZATION OF DATA

Data is modelled inside the service as "topics", with a well-defined hierarchy. Using the SDK, clients are able to select one or more topics by "path" within the hierarchy, then receive the current value followed by streaming updates.

The path to a topic is a / separated string (similar to a filesystem path). The paths for data related to a symbol have the following structure:

```
view / <topic> / <Market> / <MIC> / <SymbolIndex> / <EMM> / ...
```

Where:

- Market: name of the market the clients is willing to retrieve data from
- MIC: MIC code associated to the chosen market
- SymbolIndex: A numerical symbol index
- EMM: Exchange Market Mechanism, such as Cash_and_Derivative_Central_Order_Book

There are further topics (level of data) beneath this path, including:

```
... / BBO_and_Limits
... / BBO
... / FullTradeInformation
... / StandingData
```

The system performs an automated daily clearing of the book after each trading session. This scheduled reset, executed between one hour after the close of trading and midnight ensures that the book starts each business day in a clean and consistent state.

3.2 SELECTING DATA TO RECEIVE

3.2.1.1 Subscribe to one topic

To subscribe to a topic, client must first specify a "topic selector" that describes the path to it. To target a specific topic, client should enter the full path prefixed with a “>”

```
>view/FullTradeInformation/equity_italy/instrument/EXGM/3316280/Cash_and_Deriva
tive_Central_Order_Book/FullTradeInformation
```

3.2.1.2 Subscribe to multiple topics

To subscribe to multiple topics, clients can use a "?" instead of a ">" at the beginning of a topic selector:

```
?view/BBO/equity_italy/.*/3.*
```

- This command will retrieve all BBO topics (BestBid and BestOffer) across all MICs for all symbols beginning with number "3".

Clients can also subscribe to all descendants of a topic by using a "//" suffix to the topic selector:

```
? view/BBO/equity_italy/EXGM//
```

- This command will retrieve all BBOs for all Symbols of MIC="EXGM".

Clients can also retrieve data without entering the MIC associated to the instrument:

```
?view/BBO/equity_italy/.*/3316253
```

Clients can also retrieve all data for a specific market by adding "//" after the Market parameter:

```
?view/FullTradeInformation/equity_italy//
```

- This command will retrieve all the data for the equity_italy market

Clients can also retreive the standing data for all symbols that belong to a specific MIC :

```
?view/StandingData/equity_italy/instrument/EXGM/.*/StandingData
```

For this type of queries, clients are encouraged to ensure that their bandwidth is sufficient to receive and process large amount of data.

An important volume of subscriptions can cause disconnections on client side. In such cases, clients are encouraged to stagger the number of subscriptions they launch at any one time.

For all the data for the same symbol, you may ask for:

```
*view/.*/2269194//
```

4. PYTHON SDK

Clients can find below guidelines to use the Python SDK for this service. More information can be found at https://docs.diffusiondata.com/docs/latest/python

The easiest way to install the Python SDK is:

```
pip install diffusion
```

If you need to install a specific version of the SDK, you can do this like so:

```
pip install "diffusion==6.12.1"
```

The steps to receive data are:

1. Make a connection to the server
2. Register a `ValueStreamHandler` to receive updates, with appropriate callbacks.
3. Request topics from the server.

4.1 MAKING A CONNECTION TO THE SERVER

We must first establish a connection to the server using the URL, username and password that you have been provided with. If successful, the returned `session` object becomes our interface to the rest of the API.

```
async with diffusion.Session(
                "<provided_url>",
                "<provided_username>",
                diffusion.Credentials("<provided_password>")
) as session:
```

- URL of service, starts with `wss://`

4.2 REGISTERING A HANDLER FOR RECEIVED DATA

When updates start to flow from the server to the client, you will need to dispatch them to your code for processing. We do this via a `ValueStreamHandler`:

```
session.topics.add_value_stream("?view/BBO/equity_norway//", norway_stream)
```

The first argument is a topic selector, as previously discussed, and the second is the `ValueStreamHandler`, as described below.

Note that the topic selector is not used here to select topics; it is only used to dispatch updates that match the selector to an appropriate handler. It is perfectly reasonable to define multiple handlers:

```
session.topics.add_value_stream("?view/BBO/equity_norway//", norway_stream)
session.topics.add_value_stream("?view/BBO/equity_portugal//", portugal_stream)
session.topics.add_value_stream("?view/BBO//", all_bbo_data_stream)
```

The `ValueStreamHandler` itself is defined like so:

```
json_stream = topics.streams.ValueStreamHandler(
        data_type   = diffusion.datatypes.JSON,
        update      = on_update,
        subscribe   = on_subscription,
        unsubscribe = on_unsubscription
)
```

- Type of data in the topic is JSON; mandatory parameter.
- Callback when a value is received from a topic.
- Optional callbacks to notify when a topic has been successfully subscribed to, or unsubscribed from.

A skeleton for these callbacks is provided below:

```
def on_subscription(*, topic_path, **kwargs):
        print(f"Subscribed to {topic_path}")

def on_unsubscription(*, topic_path, reason, **kwargs):
        print(f"Unsubscribed from {topic_path}")

def on_update(*, old_value, topic_path, topic_value, **kwargs):
        j = json.dumps(topic_value, indent=4)
        print(f"Value for {topic_path}: {j}")
```

- `topic_path` is the path to the topic that this callback is acting on
- `reason` is an enum indicating why an unsubscription happened.
- `old_value` contains the previous value for the topic, or `None` on receipt of the first value.
- `topic_value` is the current value of the topic as received from the server.

4.3 REQUEST TOPICS FROM THE SERVER

So far we have only described how we want the client to handle updates as they are received; we have not requested them from the server yet. To do this, we issue a request to `subscribe` to a topic selector:

```
await session.topics.subscribe("?view/BBO/equity_norway/instrument/XOSL//")
```

4.4 FULL EXAMPLE

```
import asyncio
import json
import diffusion
import diffusion.features.topics as topics
import sys

#
# Callbacks for value stream handler
#
def on_subscription(*, topic_path, **kwargs):
        print(f"Subscribed to {topic_path}")

def on_unsubscription(*, topic_path, reason, **kwargs):
        print(f"Unsubscribed from {topic_path}")

def on_update(*, old_value, topic_path, topic_value, **kwargs):
        j = json.dumps(topic_value, indent=4)
        print(f"Value for {topic_path}: {j}")

#
# Main application code
#
async def main():
    # Make a connection to the Diffusion WebSocket
    async with diffusion.Session(
                    "<provided_url>",
                    "<provided_username>",
                    diffusion.Credentials("<provided_password>")
    ) as session:
            print(f"Connected!")

            # Specify callbacks for updates to topics
            json_stream = topics.streams.ValueStreamHandler(
                    data_type   = diffusion.datatypes.JSON,
                    update      = on_update,
                    subscribe   = on_subscription,
                    unsubscribe = on_unsubscription
            )
            # Set up a stream to handle any updates received from topics that
            # match the given topic selector
            session.topics.add_value_stream("?view/BBO/equity_norway//",
json_stream)

            # Subscribe to topics matching the given topic selector
            await
session.topics.subscribe("?view/BBO/equity_norway/instrument/XOAS//")

            # Run for 5 minutes
            await asyncio.sleep(300)

            # Cleanup
            await session.topics.remove_stream(json_stream)
```

```
            await session.close()

#
# Entry point
#
if __name__ == "__main__":
        try:
                asyncio.run(main())
        except KeyboardInterrupt:
                print("Interrupted")
                sys.exit(0)
```

5. Example of WebSocket response:

5.1 CASH AND DERIVATIVE CENTRAL ORDER BOOK

5.1.1 BBO and Limits

5.1.1.1 BBO

Path	Json Response
view/BBO_and_Limits/equity_italy/EXGM/3316280/BBO	{ "symbolIndex": 3316280, "bestBid": { "price": 45800000, "numberOfOrders": 2, "quantity": 850, "oBSeqNum": 30, "mDSeqNum": 23823, "rebroadcastIndicator": 0, "eventTime": 1779087606974574374, "mDContext": { "mDDecodeSeqNum": 166576, "mDDecodeTime": 1779087607054, "packetTime": 1779087606974641312, "packetSeqNum": 24050, "packetFlags": 0, "channelId": 11077 } }, "bestOffer": {l "price": 46800000, "numberOfOrders": 1, "quantity": 8023, "oBSeqNum": 35, "mDSeqNum": 1324061, "rebroadcastIndicator": 0, "eventTime": 1779092352661672536, "mDContext": { "mDDecodeSeqNum": 3457059, "mDDecodeTime": 1779092352721, "packetTime": 1779092352661715243, "packetSeqNum": 577874, "packetFlags": 0, "channelId": 11077

	} }, "oBSeqNum": 35, "lastClearSeqNum": 1, "lastClearTime": 1779077126436 }

Definition of attributes:

Field	Short Description
symbolIndex	Exchange identification code of the instrument/contract
bestBid	Best bid price level
bestOffer	Best offer price level
price	Price per unit of quantity (to be calculated with the Price/Index Level Decimals)
numberOfOrders	Number of orders at the current price limit
quantity	Number of traded or ordered units (to be calculated with Quantity Decimals)
oBSeqNum	Technical field to track message updates
mDSeqNum	Assigned by MDG for each message. Each channel has its own Market Data Sequence Number sequence
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'
eventTime	Time when an event has been processed
mDContext	Metadata about the market data packet (sequence numbers, timestamps, channel) that carried the message through the network
mDDecodeSeqNum	Technical field to track message updates
mDDecodeTime	Time when an event has been processed
packetTime	Time when the packet is pushed to the clients (Time in number of nanoseconds since 01/01/1970 UTC)
packetSeqNum	Each channel has its own PSN sequence. Starting from 1 at every MDG start and increasing by step of 1
packetFlags	Used to flag information (Little-Endian): Bit 0 Compression, Bit 1-3 restart counter, Bit 4-6 PSN overflow bits, Bit 7-9 snapshot/health status flags, Bit 10-15 for future use
channelId	Identifies the channel. First figure defines if it is Real-Time or Snapshot feed. Second figure identifies the MDG partition. Last 3 figures are the channel identifier
oBSeqNum	-
lastClearSeqNum	Technical field to track message updates
lastClearTime	Time when an event has been processed

5.1.1.2 Limits

Path	Json Response

view/BBO_and_Limits/equity_italy/EXGM/3316280/Limits	{
	"symbolIndex": 3316280,
	"bids": [
	{
	"price": 45800000,
	"numberOfOrders": 2,
	"quantity": 850,
	"oBSeqNum": 20,
	"mDSeqNum": 4449,
	"rebroadcastIndicator": 0,
	"eventTime": 1779083203839951190,
	"mDContext": {
	"mDDecodeSeqNum": 92074,
	"mDDecodeTime": 1779083203892,
	"packetTime": 1779083203839971593,
	"packetSeqNum": 11958,
	"packetFlags": 0,
	"channelId": 11077
	}
	},
	{
	"price": 45400000,
	"numberOfOrders": 2,
	"quantity": 2274,
	"oBSeqNum": 34,
	"mDSeqNum": 23829,
	"rebroadcastIndicator": 0,
	"eventTime": 1779087606975452835,
	"mDContext": {
	"mDDecodeSeqNum": 166577,
	"mDDecodeTime": 1779087607054,
	"packetTime": 1779087606975469732,
	"packetSeqNum": 24051,
	"packetFlags": 0,
	"channelId": 11077
	}
	},
	{
	"price": 45200000,
	"numberOfOrders": 1,

```
  "quantity": 300,
  "oBSeqNum": 28,
  "mDSeqNum": 11730,
  "rebroadcastIndicator": 0,
  "eventTime": 1779087510494566308,
  "mDContext": {
   "mDDecodeSeqNum": 134393,
   "mDDecodeTime": 1779087510522,
   "packetTime": 1779087510494582717,
   "packetSeqNum": 18186,
   "packetFlags": 0,
   "channelId": 11077
  }
 },
 {
  "price": 45000000,
  "numberOfOrders": 2,
  "quantity": 1500,
  "oBSeqNum": 26,
  "mDSeqNum": 7894,
  "rebroadcastIndicator": 0,
  "eventTime": 1779086626510627083,
  "mDContext": {
   "mDDecodeSeqNum": 116503,
   "mDDecodeTime": 1779086626538,
   "packetTime": 1779086626510650536,
   "packetSeqNum": 15565,
   "packetFlags": 0,
   "channelId": 11077
  }
 },
 {
  "price": 44600000,
  "numberOfOrders": 1,
  "quantity": 750,
  "oBSeqNum": 5,
  "mDSeqNum": 929,
  "rebroadcastIndicator": 1,
  "eventTime": 1779068686504815045,
  "mDContext": {
   "mDDecodeSeqNum": 32197,
   "mDDecodeTime": 1779077126435,
   "packetTime": 1779068686708494707,
   "packetSeqNum": 2987,
```

```
      "packetFlags": 0,
      "channelId": 11077
    }
  }
],
"offers": [
  {
    "price": 46800000,
    "numberOfOrders": 1,
    "quantity": 8023,
    "oBSeqNum": 36,
    "mDSeqNum": 1324061,
    "rebroadcastIndicator": 0,
    "eventTime": 1779092352661672536,
    "mDContext": {
      "mDDecodeSeqNum": 3457059,
      "mDDecodeTime": 1779092352721,
      "packetTime": 1779092352661715243,
      "packetSeqNum": 577874,
      "packetFlags": 0,
      "channelId": 11077
    }
  },
  {
    "price": 47600000,
    "numberOfOrders": 1,
    "quantity": 240,
    "oBSeqNum": 10,
    "mDSeqNum": 929,
    "rebroadcastIndicator": 1,
    "eventTime": 1779068686504815045,
    "mDContext": {
      "mDDecodeSeqNum": 32197,
      "mDDecodeTime": 1779077126435,
      "packetTime": 1779068686708494707,
      "packetSeqNum": 2987,
      "packetFlags": 0,
      "channelId": 11077
    }
  },
  {
    "price": 47800000,
    "numberOfOrders": 3,
    "quantity": 1073,
```

```
    "oBSeqNum": 23,
    "mDSeqNum": 4451,
    "rebroadcastIndicator": 0,
    "eventTime": 1779083203840280348,
    "mDContext": {
      "mDDecodeSeqNum": 92075,
      "mDDecodeTime": 1779083203892,
      "packetTime": 1779083203840298074,
      "packetSeqNum": 11959,
      "packetFlags": 0,
      "channelId": 11077
    }
  },
  {
    "price": 48000000,
    "numberOfOrders": 1,
    "quantity": 1000,
    "oBSeqNum": 12,
    "mDSeqNum": 929,
    "rebroadcastIndicator": 1,
    "eventTime": 1779068686504815045,
    "mDContext": {
      "mDDecodeSeqNum": 32197,
      "mDDecodeTime": 1779077126435,
      "packetTime": 1779068686708494707,
      "packetSeqNum": 2987,
      "packetFlags": 0,
      "channelId": 11077
    }
  },
  {
    "price": 48600000,
    "numberOfOrders": 1,
    "quantity": 1000,
    "oBSeqNum": 13,
    "mDSeqNum": 929,
    "rebroadcastIndicator": 1,
    "eventTime": 1779068686504815045,
    "mDContext": {
      "mDDecodeSeqNum": 32197,
      "mDDecodeTime": 1779077126435,
      "packetTime": 1779068686708494707,
      "packetSeqNum": 2987,
      "packetFlags": 0,
```

```
      "channelId": 11077
    }
   }
  ],
  "currentBidPrices": 10,
  "currentOfferPrices": 7,
  "oBSeqNum": 36,
  "lastClearSeqNum": 1,
  "lastClearTime": 1779077126436
}
```


Definition of attributes:

FIELD	SHORT DESCRIPTION
symbolIndex	Exchange identification code of the instrument/contract
bids	List of the five best Bids
price	Price per unit of quantity (to be calculated with the Price/Index Level Decimals).
numberOfOrders	Number of orders at the current price limit.
quantity	Number of traded or ordered units (to be calculated with Quantity Decimals).
oBSeqNum	Technical field to track message updates
mDSeqNum	Assigned by MDG for each message. Each channel has its own Market Data Sequence Number sequence
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'
eventTime	Time when an event has been processed
mDContext	Metadata about the market data packet (sequence numbers, timestamps, channel) that carried the message through the network
mDDecodeSeqNum	Technical field to track message updates
mDDecodeTime	Time when an event has been processed
packetTime	Time when the packet is pushed to the clients (Time in number of nanoseconds since 01/01/1970 UTC)
packetSeqNum	Each channel has its own PSN sequence. Starting from 1 at every MDG start and increasing by step of 1 (...)
packetFlags	Used to flag information (Little-Endian): Bit 0 Compression, Bit 1-3 restart counter, Bit 4-6 PSN overflow bits, Bit 7-9 snapshot/health status flags, Bit 10-15 for future use
channelId	Identifies the channel. First figure defines if it is Real-Time or Snapshot feed. Second figure

	identifies the MDG partition. Last 3 figures are the channel identifier
offers	List of the five best Offers
currentBidPrices	Number of current bid price levels available
currentOfferPrices	Number of current offer price levels available
oBSeqNum	Technical field to track message updates
lastClearSeqNum	Technical field to track message updates
lastClearTime	Time when an event has been processed

5.1.2 BBO

Path	Json Response

view/BBO/equity_italy/EXGM/3305367	{
	"symbolIndex": 3305367,
	"bestBid": null,
	"bestOffer": {
	"price": 40000000,
	"numberOfOrders": 2,
	"quantity": 31500,
	"oBSeqNum": 4,
	"mDSeqNum": 5777067,
	"rebroadcastIndicator": 0,
	"eventTime": 1779102048815736237,
	"mDContext": {
	"mDDecodeSeqNum": 6928100,
	"mDDecodeTime": 1779102048847,
	"packetTime": 1779102048815759129,
	"packetSeqNum": 2457771,
	"packetFlags": 0,
	"channelId": 12077
	}
	},
	"oBSeqNum": 4,
	"lastClearSeqNum": 1,
	"lastClearTime": 1779077123545
	}

Definition of attributes:

FIELD	SHORT DESCRIPTION
symbolIndex	Exchange identification code of the instrument/contract
bestBid	Best bid price level (null if no bid available)
bestOffer	Best offer price level

price	Price per unit of quantity (to be calculated with the Price/Index Level Decimals). NB: price will appear as null when the best price available in the order book is derived from a market order.
numberOfOrders	Number of orders at the current price limit
quantity	Number of traded or ordered units (to be calculated with Quantity Decimals)
oBSeqNum	Technical field to track message updates
mDSeqNum	Identifies the channel.
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'.
eventTime	Time when the change is effective
mDContext	Metadata about the market data packet (sequence numbers, timestamps, channel) that carried the message through the network
mDDecodeSeqNum	Technical field to track message updates
mDDecodeTime	Time when an event has been processed
packetTime	Time when the packet is pushed to the clients (Time in number of nanoseconds since 01/01/1970 UTC)
packetSeqNum	-
packetFlags	-
channelId	Identifies the channel.
oBSeqNum	Technical field to track message updates
lastClearSeqNum	Technical field to track message updates
lastClearTime	Time when an event has been processed

5.1.3 Full Trade Information

Path	Json Response
view/FullTradeInformation/equity_italy/instrument/EXGM/3316280/Cash_and_Derivative_Central_Order_Book/FullTradeInformation	{ "mMTContributiontoPrice": "-", "tradeReference": "", "mMTSpecialDividendIndicator": "-", "tradingDateTime": "2026-05-18T08:19:12.661673Z", "rebroadcastIndicator": 0, "mMTPublicationModeSizeSpecific": "-", "messagePriceNotation": "null", "postTradeDeferralFlags": "", "mDSeqNum": 1324062, "publicationDateTime": "2026-05-18T08:19:12.661704Z", "endTimeVwap": null, "mifidQuantityMeasurementUnit": "",

```
  "mMTBenchmarkIndicator": "-",
  "repoSettlementDate": null,
  "mMTDuplicativeIndicator": "",
  "priceMultiplierDecimals": null,
  "mMTPublicationModeIlliquid": "-",
  "miFIDQtyinMsrmtUnitNotation": "",
  "mMTPreTradeTransparencyWaiverRelatedtoSizeandScale":   "-
",
  "tradeQualifier": {
    "UncrossingTrade": false,
    "FirstTradePrice": false,
    "PassiveOrder": false,
    "AggressiveOrder": true,
    "TradeCreationByMarketOperations": false,
    "NAVTradeExpressedInBps": false,
    "NAVTradeExpressedInPriceCurrency": false,
    "DeferredPublication": false
  },
  "mMTMarketMechanism": "Central_Limit_Order_Book",
  "miFIDEmissionAllowanceType": "",
  "settlementDate": null,
  "miFIDClearingFlag": "-",
  "mMTTransactionCategory": "-",
  "marketOfReferenceMIC": "",
  "mifidInstrumentID": "IT0004125677",
  "venueofPublication": "EXGM",
  "mMTPublicationMode": "-",
  "symbolIndex": 3316280,
  "mMTModificationIndicator": "-",
  "mifidPriceNotation": "MONE",
  "notionalCurrency": "EUR",
  "venue": "EXGM",
  "transparencyIndicator": "LitRegular_Trade",
  "eMM": "Cash_and_Derivative_Central_Order_Book",
  "currencyCoefficient": null,
  "blockTradeCode": "Regular_trade_or_Negotiated_deal",
  "mMTTradingMode": "Continuous_Trading",
  "priceMultiplier": null,
  "mMTPortfolioTransactionIndicator": "-",
  "mifidCurrency": "EUR",
  "evaluatedPrice": null,
  "missingPrice": "",
  "eventTime": 1779092352661672536,
  "mifidPrice": "4.6800000",
```

```
  "mMTAlgorithmicIndicator": "-",
  "mDContext": {
    "mDDecodeSeqNum": 3457062,
    "mDDecodeTime": 1779092352722,
    "packetTime": 1779092352661745514,
    "packetSeqNum": 25903,
    "packetFlags": 1,
    "channelId": 11075
  },
  "mMTNegotiationIndicator": "-",
  "startTimeVwap": null,
  "tradeType": "Conventional_Trade",
  "NotUsedGroup1": [],
  "tradeUniqueIdentifier": "1OIABL0KX",
  "effectiveDateIndicator":
"Seller_declaration_is_received_on_the_cur_trading_session_da
y",
  "mifidExecutionID": "00033162800010000000513",
  "mifidInstrumentIdType": "ISIN",
  "miFIDNotionalAmount": "",
  "originalReportTimestamp": null,
  "mMTPostTradeDeferral": "-",
  "transactionType": "Plain_Vanilla_Trade",
  "mMTOffBookAutomatedIndicator":
"Unspecified_or_does_not_apply",
  "mifidQuantity": "500",
  "mMTAgencyCrossTradeIndicator": "-",
  "mMTContingentTransactionIndicator": "-"
}
```


Definition of attributes:

FIELD	SHORT DESCRIPTION
mDContext	Metadata about the market data packet (sequence numbers, timestamps, channel) that carried the trade message through the network.
mDSeqNum	Assigned by MDG for each message. Each channel has its own Market Data Sequence Number sequence.
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'.
eMM	Defines the Exchange Market Mechanism applied on each platform.
eventTime	Time when an event has been processed

symbolIndex	Exchange identification code of the instrument/contract.
tradingDateTime	Date and time when the transaction was executed
publicationDateTime	Date and time when the transaction was published by a trading venue or Approved Publication Arrangement (APA).
tradeType	Type of trade.
mifidInstrumentIdType	Code type used to identify the financial instrument
mifidInstrumentID	Code used to identify the financial instrument. This code has to be processed with the MiFID Instrument ID Type.
mifidExecutionID	MiFID Transaction Identification Code is a unique ID of the Execution per instrument, day and EMM. The value in the field is a concatenation of the Execution ID (10 char) and an identifier of the instruments [ISIN code (12 char) or Symbol Index (10 char)]. In most cases the identifier of the instrument is the 12 character ISIN code. For derivatives, in cases when the trade occurs on an instrument for which the ISIN code is not populated (e.g. Strategies) the Symbol Index of the instrument is used instead.
mifidPrice	Traded price of the transaction excluding, where applicable, commission and accrued interest.
mifidQuantity	Number of units of the financial instrument. The nominal or monetary value of the financial instrument.
mifidPriceNotation	Indication as to whether the price is expressed in monetary value, in percentage or in yield.
mifidCurrency	Currency in which the price is expressed (applicable if the price is expressed as monetary value) following ISO 4217 standard
miFIDQtyinMsrmtUnitNotation	Indication of measurement units in which the quantity in measurement unit is expressed.
mifidQuantityMeasurementUnit	The equivalent amount of commodity or emission allowance traded expressed in measurement unit
miFIDNotionalAmount	Nominal amount or notional amount.
notionalCurrency	Currency in which the notional is denominated following ISO 4217 standard.
miFIDClearingFlag	Code to identify whether the transaction will be cleared.
mMTMarketMechanism	Defines the fundamental functional market mechanism that has facilitated the trade following MMT level 1. This field is technically

	optional for backward compatibility but is functionally mandatory.
mMTTradingMode	Differentiates transactions by defining the trading mode under which the trade was executed following MMT level 2. This field is technically optional for backward compatibility but is functionally mandatory.
mMTTransactionCategory	Defines the transaction category following MMT level 3.1. This field is technically optional for backward compatibility but is functionally mandatory
mMTNegotiationIndicator	Defines the negotiation indicator or pre-trade transparency waiver following MMT level 3.2. This field is technically optional for backward compatibility but is functionally mandatory.
mMTAgencyCrossTradeIndicator	Defines the agency cross trade indicator following MMT level 3.3.
mMTModificationIndicator	Defines the modification indicator following MMT level 3.4. This field is technically optional for backward compatibility but is functionally mandatory.
mMTBenchmarkIndicator	Defines the benchmark indicator or the reference price indicator following MMT level 3.5. This field is technically optional for backward compatibility but is functionally mandatory
mMTSpecialDividendIndicator	Defines the special dividend indicator following MMT level 3.6. This field is technically optional for backward compatibility but is functionally mandatory.
mMTOffBookAutomatedIndicator	Defines the off book automated indicator following MMT level 3.7. This field is technically optional for backward compatibility but is functionally mandatory.
mMTContributiontoPrice	Defines the contribution to price or the price discovery process following MMT level 3.8. This field is technically optional for backward compatibility but is functionally mandatory.
mMTAlgorithmicIndicator	Defines the algorithmic indicator following MMT level 3.9. This field is technically optional for backward compatibility but is functionally mandatory.
mMTPublicationMode	Defines the publication mode or post-trade deferral reason following MMT level 4.1.
mMTPostTradeDeferral	Defines the post trade deferral or enrichment type following MMT level 4.2. This field is technically optional for backward compatibility but is functionally mandatory.
mMTDuplicativeIndicator	Defines the duplicative indicator following MMT level 5.

tradeQualifier	Trade Qualifier. Values specified, in the list of possible values, indicate the bit positions that should be used to set zero (0) or one (1) values. A single field contains multiple values provided in different positions.
UncrossingTrade	Indicates whether the trade occurred during an Uncrossing, or not.
FirstTradePrice	Indicates whether the price of the trade is the first trade price of the day, or not
PassiveOrder	Indicates whether the corresponding order was passive, or not. (
AggressiveOrder	Indicates whether the corresponding order was aggressive, or not
TradeCreationByMarketOperations	Indicates whether the trade results from a creation by Market Operations, or not
NAVTradeExpressedInBps	Indicates whether the trade results from a NAV trade expressed in basis point on the ETF Access platform.
NAVTradeExpressedInPriceCurrency	Indicates whether the trade is a NAV trade expressed in price currency. This trade is always an update from a previous NAV trade expressed in basis point on the ETF Access platform.
DeferredPublication	indicates whether the trade publication is deferred or immediate.
transactionType	Transaction type or publication type.
effectiveDateIndicator	Indicates if the trade is introduced on the trading session day or earlier.
blockTradeCode	Indicates if trade relates to a block or a negotiated deal following EU and UK MiFID II rules.
tradeReference	Reference of the trade reported to the Exchange.
originalReportTimestamp	Timestamp of trade reporting to the Exchange
transparencyIndicator	Used to define the transparency of the trade.
currencyCoefficient	When an actual price is displayed in a different 'price expression' than the official instrument trading currency, the Currency Coefficient represents the ratio 'price expression' divided by 'official currency' (To be calculated with Ratio / Multiplier Decimals).
priceMultiplier	Number of units of the financial instrument that are contained in a trading lot. Price multiplier coefficient for instrument unit price.
priceMultiplierDecimals	Number of decimals for the field Price Multiplier.

venue	Identification of the venue where the transaction was executed using the ISO 10383 segment MIC for transactions executed on a trading venue.
startTimeVwap	Start time for the Volume Weight Average price computation period
endTimeVwap	End time for the Volume Weight Average price computation period
miFIDEmissionAllowanceType	This field is only applicable for emission allowances.
marketOfReferenceMIC	Indicates the instrument Exchange of Reference by its MIC (Market Identification Code according to ISO 10383) (For Future Use)
evaluatedPrice	This field is used for Total Return Future and Market On Close Future trading. The Price is calculated by ME in index point notation. It can contain the provisional price calculated in intraday or the final price calculated at the end of the day. The Trade Type value will enable user to know if the field contains provisional price or final price.
messagePriceNotation	This field provides the type of price notation used per message. For TRF and MOC products the value "Price" is used for TAM trading mode, the values "Spread in basis points" and "Spread" are used for TAIC trading mode.
settlementDate	Date when a trade is final, and the buyer must make payment to the seller while the seller delivers the assets to the buyer
repoSettlementDate	Date when the RepoSeller gets the assets back from the RepoBuyer. Used in case of a Repo (Repurchase Agreement) trade.
tradeUniqueIdentifier	Alphanumerical code unique, consistent and persistent per ISO10383 segment MIC and per trading day assigned by the trading venue to a transaction. Trade Unique Identifier (TUI) is a field aimed at identifying an individual transaction and used as Trading Venue Transaction Identification Code (TVTIC). This is a unique transaction identification code generated by trading venues and disseminated to both the buying and the selling parties, in accordance with Article 12 of the Commission Delegated Regulation (EU) 2017/580 Regulatory Technical Standards (RTS)
missingPrice	Defines the reason why Price is not available
mMTPreTradeTransparencyWaiverRelatedtoSizeandScale	Defines the pre-trade transparency waiver related to the size following MMT level 3.10.

	This field is technically optional for backward compatibility but is functionally mandatory
mMTPortfolioTransactionIndicator	Defines if the transaction is part of a portfolio trade following MMT level 3.11. This field is technically optional for backward compatibility but is functionally mandatory.
mMTContingentTransactionIndicator	Defines a transaction where all the components of the trade are meant to be executed as a single lot following MMT level 3.12. This field is technically optional for backward compatibility but is functionally mandatory.
mMTPublicationModeIlliquid	Defines if the post-trade deferral is due to liquidity reason following MMT level 4.3. This field is technically optional for backward compatibility but is functionally mandatory.
mMTPublicationModeSizeSpecific	Defines if the post-trade deferral is due to size specific reason following MMT level 4.4. This field is technically optional for backward compatibility but is functionally mandatory.
venueofPublication	Identification of the venue where the transaction was published using the ISO 10383 segment MIC for transactions executed on a trading venue.
postTradeDeferralFlags	Indicates which deferral type was applied for transaction on Bonds instrument
NotUsedGroup1	List

5.2 STANDING DATA

Path	Json Response
view/StandingData/equity_norway/instrument/XOAS/2272761/StandingData	{ "instrumentEventDate": 20557, "strikeCurrencyIndicator": "null", "poolFactor": null, "maxOrderQuantityCall": 999999999, "lastAdjustedClosingPrice": 33800000, "iCB": "0000000000000000", "rebroadcastIndicator": 0, "maxOrderAmountCall": 4337999890000000, "iCBCode": "", "quantityDecimals": 0, "underlyingMIC": "", "mDSeqNum": 5, "tradingCurrencyIndicator": "null",

	"instUnitExp": "Units", "maturityDate": "", "tradingCurrency": "NOK", "issuePriceDecimals": 3, "instrumentGroupCode": "OD", "priceDecimals": 7, "mnemonic": "ALNG", "typeOfMarketAdmission": "Regulated_Market_Equities_Segment_C", "maximumDecimalsInQuantity": null, "maxOrderQuantityContinuous": 999999999, "strikeCurrency": "", "liquidInstrumentIndicator": 0, "dateOfLastTrade": 20588, "depositaryList": "00050", "countryOfExchange": "NOR", "taxCode": "Eligible_to_PEA", "marketOfReferenceMIC": "", "numberInstrumentCirculating": 209860609, "instrumentName": "AWILCO LNG", "firstSettlementDate": null, "thresholdLISPostTrade60mn": 888230000000, "darkLISThreshold": 1776450000000, "longMnemonic": "ALNG", "optiqSegment": "Equities", "symbolIndex": 2272761, "amountDecimals": 7, "thresholdLISPostTradeEOD": null, "typeOfCorporateEvent": "00", "nominalCurrency": "NOK", "strikePriceDecimals": null, "thresholdLISPostTrade120mn": 1776450000000, "strikePrice": null, "guaranteeIndicator": "This_instrument_is_guaranteed", "iSINCode": "NO0010607971", "fullInstrumentName": "AWILCO LNG", "instrumentTradingCode": "NO0010607971", "issuePrice": 1075, "partitionID": 12, "currencyCoefficient": null, "mIC": "XOAS", "ratioDecimals": 7, "underlyingISINCode": "", "darkMinQuantity": 0,

```
  "mDContext": {
    "mDDecodeSeqNum": 33177,
    "mDDecodeTime": 1779077123349,
    "packetTime": 1779068585676739893,
    "packetSeqNum": 2858,
    "packetFlags": 1,
    "channelId": 12085
  },
  "parValue": 1000000,
  "lotSize": 1,
  "cFI": "ESVUFR",
  "mICList": "XOAS",
  "mainDepositary": "00050",
  "repoIndicator":
"Instr_neither_eligible_for_SRD_or_Loan_and_Lending_Mkt"
,
  "EMMPatternRep": [
    {
      "eMM": "Cash_and_Derivative_Central_Order_Book",
      "patternID": 1320,
      "tickSizeIndexID": 21,
      "marketModel": "Order_Driven",
      "lotSize": 1,
      "instUnitExp": "Units",
      "anonymous": "Yes"
    },
    {
      "eMM": "Cash_On_Exchange_Off_book",
      "patternID": null,
      "tickSizeIndexID": 88,
      "marketModel": "Order_Driven",
      "lotSize": 1,
      "instUnitExp": "Units",
      "anonymous": "null"
    }
  ],
  "settlementDelay": "02",
  "darkEligibility": 1,
  "issuingCountry": "NOR",
  "quantityNotation": "UNT",
  "maxOrderAmountContinuous": 4337999890000000
}
```


Definition of attributes:

FIELD	SHORT DESCRIPTION
mDContext	Metadata about the market data packet (sequence numbers, timestamps, channel) that carried the instrument message through the network.
mDSeqNum	Assigned by MDG for each message. Each channel has its own Market Data Sequence Number sequence
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'.
symbolIndex	Exchange identification code of the instrument/contract.
optiqSegment	An Optiq segment is a universe of instruments sharing common trading properties.
partitionID	Identifies uniquely an Optiq partition across all the Exchange partitions
fullInstrumentName	Full Instrument Name.
instrumentName	Instrument Name
instrumentTradingCode	Cash: Trading code is a 12-character string, the only instrument identifier that is unique in the feed in addition to the symbol index.
instrumentGroupCode	Instrument Group / Class Identifier.
iSINCode	Instrument ISIN following ISO 6166.
priceDecimals	Indicates the number of decimals for each Price related to this Symbol Index
quantityDecimals	Indicates the number of decimals for each Quantity related to this Symbol Index
amountDecimals	Indicates the number of decimals for each Amount related to this Symbol Index
ratioDecimals	Indicates the number of decimals for each Ratio related to this Symbol Index
cFI	Classification code of a financial instrument defined by the ISO-10962:2015 standard.
instrumentEventDate	Date of the last instrument characteristic modification(s) except for some exceptions.
strikePrice	The strike price of an option/warrant is the specified price at which the underlying can be bought (in the case of a call/right to buy) or sold (in case of a put/right to sell) by the holder (buyer) of the option/warrant contract, at the moment he exercises his right against a writer (seller) of the option/warrant.
darkEligibility	Indicates the Eligibility to dark. 0 is not eligible, 1 is eligible.
darkLISThreshold	Defines the minimum amount of an order to benefit from the LIS (Large In Scale) pre-transparency waiver (to be calculated with the Amount Decimals).

darkMinQuantity	Defines the minimum quantity required for an order to be filled in the Dark liquidity. 0 indicates that no minimum amount is required.
dateOfLastTrade	Date of the Last Price for the Instrument
depositaryList	Identifies the possible main depository organizations (maximum four) for shares or fixed income.
mainDepositary	Identifies the default (or main) depository organization of the instrument (between the possible 4 depositaries registered) used by priority for the settlement (for example: multi-listed instruments which have several depositories).
firstSettlementDate	Represents the first possible settlement date for a given instrument.
guaranteeIndicator	Indicates if the trade is guaranteed or not (for clearing purpose)
iCB	Identifies for a listed instrument, the economic subsector of the issuing company in the ICB (Industry Classification Benchmark) classification. This field is not relevant and is currently filled with '00000000'.
issuingCountry	Provides the ISO 3166 (Alpha 3) code for the country of headquarter company that issued the instrument.
lastAdjustedClosingPrice	Last traded price of the previous trading day after application of the adjustment coefficient (to be calculated with the Price/Index Level Decimals). Not provided for European instruments.
lotSize	For Cash and Derivatives, it defines a multiple of the tradable quantity.
maturityDate	Maturity Date of the instrument (text formatted as YYYYMMDD). For contracts with one expiry per month the day component may be "00" (text formatted as YYYYMMDD). For repo (repurchase agreement) it represents the inclusive date until which a lending/borrowing contract can be traded.
maximumDecimalsInQuantity	Maximum Decimals In Quantity was introduced for Euronext Fund Services Paris and indicates the maximum of relevant decimal number for trading.
mIC	Appendix
mICList	Identifies the Euronext markets on which an instrument is listed by its MIC (Market Identification Code). For an instrument listed on a single Euronext market, the listing MIC code is the same than "Market Identification Code (MIC) of the listed instrument" For an instrument listed on several Euronext Markets: - The first MIC is the same than the "Market Identification Code (MIC) of the listed instrument - The others MIC indicate the other listing places

countryOfExchange	Country of exchange is the Country associated to the MIC following ISO 3166 Alpha-3.
mnemonic	Mnemonic code of the instrument. This field is not populated for every instrument
underlyingMIC	Identifies the market to which an instrument' underlying belongs by its MIC (Market Identification Code), according to ISO 10383. Refer to MIC field to have all the authorized values.
underlyingISINCode	Underlying ISIN.
tradingCurrency	Code of the currency (ISO 4217-3A).
currencyCoefficient	When an actual price is displayed in a different 'price expression' than the official instrument trading currency, the Currency Coefficient represents the ratio 'price expression' divided by 'official currency' (To be calculated with Ratio / Multiplier Decimals).
tradingCurrencyIndicator	Indicates whether the 'price expression' is in the Currency or in a ratio of this Currency. Use Currency Coefficient field to identify the ratio to apply.
strikeCurrencyIndicator	Indicates whether the 'price expression' is in the Currency or in a ratio of this Currency. Use Currency Coefficient field to identify the ratio to apply.
numberInstrumentCirculating	For stocks: this is the total number of shares issued by the company. For Fix Income: this is the number of Fix Income still to be repaid.
parValue	Par Value (also called Nominal value) for Instrument. For Fixed Income it represents the par amount to be repaid at maturity (not including interest revenue) (to be calculated with the Amount Decimals).
quantityNotation	Indication of the type of measurement (e.g. number of units, nominal, monetary value, etc.) in which the transaction is expressed.
instUnitExp	Unit in which the instrument is quoted.
settlementDelay	Gives the number of trading days that represents the period between the trade date and the settlement date (delivery and payment) for an instrument to be cleared and settled. This is generally a standard period for Euronext Cash markets. Permitted Values - From 0 to 30 (Standard values) - X: This value is assigned for a lot of products and internal management rules shared by Euronext and LCH-Clearnet (D+2). - Z: This value is assigned for Lending/Borrowing instruments. This value is especially interpreted to manage the associated management rules (D+3).
strikeCurrency	Code of the strike currency (ISO 4217-3A)

taxCode	Tax deduction code to which the instrument belongs.
typeOfCorporateEvent	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 "02" – Interest payment (Fix Income 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 Fix Income redemption "12" – Capital amortization "13" – Draw announcement (Belgian Fix Income 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 instrument)
typeOfMarketAdmission	Indicates the type of market to which an instrument has been listed.
repoIndicator	Indicates whether the instrument listed underlies any loan contracts, meaning it has been admitted to the Deferred Settlement system and/or to the lending market.
issuePrice	Issuing price of the instrument
nominalCurrency	Code of the nominal currency (ISO 4217-3A).

issuePriceDecimals	Indicates the number of decimals for Issue Price related to this Symbol Index
strikePriceDecimals	Indicates the number of decimals for Strike Price related to this Symbol Index
liquidInstrumentIndicator	Indicates whether the instrument is liquid or not, as defined per EU and UK MiFID II. (0 = Illiquid; 1 = Liquid)
marketOfReferenceMIC	Indicates the instrument Exchange of Reference by its MIC (Market Identification Code according to ISO 10383) (For Future Use).
iCBCode	Identifies for a listed instrument, the economic subsector of the issuing company in the ICB (Industry Classification Benchmark) classification. This field is not relevant and is currently filled with '00000000'.
thresholdLISPostTrade60mn	Defines the amount of an order to benefit from the LIS Trade Deferred publication to 60 min (to be calculated with the Amount Decimals).
thresholdLISPostTrade120mn	Defines the amount of an order to benefit from the LIS Trade Deferred publication to 120 min (to be calculated with the Amount Decimals).
thresholdLISPostTradeEOD	Defines the amount of an order to benefit from the LIS Trade Deferred publication to EOD (to be calculated with the Amount Decimals).
longMnemonic	Mnemonic code of the instrument. This field is not populated for every instrument
maxOrderAmountCall	Maximum order amount allowed at order entry during a call phase (to be combined with Amount Decimals).
maxOrderAmountContinuous	Maximum order amount allowed at order entry during a continuous phase (to be combined with Amount Decimals).
maxOrderQuantityCall	Maximum order quantity allowed at order entry during a call phase (to be combined with Quantity Decimals).
maxOrderQuantityContinuous	Maximum order quantity allowed at order entry during a continuous phase (to be combined with Quantity Decimals).
poolFactor	Allows to calculate how much of the original loans have yet to be repaid
EMMPatternRep	List
eMM	Defines the Exchange Market Mechanism applied on each platform.
patternID	Numerical Pattern identifier available as a characteristic of an instrument in Standing Data file and message, and used in the MDG timetable message. Cash Markets only.
tickSizeIndexID	ID of the tick size table available in the Tick Table file.
marketModel	Market Model identifier.

lotSize	For Cash and Derivatives, it defines a multiple of the tradable quantity.
instUnitExp	Unit in which the instrument is quoted.
anonymous	Indicates if the Market Data notifications on the instrument are anonymous or not. (0: Non Anonymous – Member Firm ID published; 1: Anonymous – Member Firm ID not published).

5.3 STATISTICS

Path	Json Response

json/equity_norway/instrument/ MERK/3036242/Statistics	{
	"mDSeqNum": 702864,
	"rebroadcastIndicator": 0,
	"symbolIndex": 3036242,
	"NewStats": [
	{
	"statsUpdateType": "Daily_Low",
	"statsUpdateValue": 12700000
	},
	{
	"statsUpdateType": "Variation_Last_Price",
	"statsUpdateValue": -23076923
	},
	{
	"statsUpdateType": "Trade_Count",
	"statsUpdateValue": 2
	},
	{
	"statsUpdateType": "Last_Traded_Price",
	"statsUpdateValue": 12700000
	},
	{
	"statsUpdateType": "Percent_Variation_Previous_Close",
	"statsUpdateValue": 24193548
	},
	{
	"statsUpdateType": "On_Book_Continuous_Cumul_Qty",
	"statsUpdateValue": 463
	},
	{
	"statsUpdateType": "On_and_Off_Book_Cumul_Qty",
	"statsUpdateValue": 544
	}

	] }

Definition of attributes:

FIELD	SHORT DESCRIPTION
mDSeqNum	Assigned by MDG for each message. Each channel has its own Market Data Sequence Number sequence
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'.
symbolIndex	Exchange identification code of the instrument/contract.
NewStats	List
statsUpdateType	Indicates the type of published statistics update
statsUpdateValue	Indicates the value of the published statistics update. This field has to be calculated with a scale code field depending on the "Stats Update Type" as follow: - Price / Index Level Decimals for "Stats Update Type": "5 - Daily High", "6 - Daily Low", "7 - Yearly High", "8 - Yearly Low", "9 - Lifetime High", "10 - Lifetime Low", "15 - Open Price" and "17 - Last Trade Price" - Quantity Decimals for "Stats Update Type": "19 - Off Book Cumulative Quantity", "21 - On Book Auction Cumulative Quantity", "22 - On book Continuous Cumulative Quantity" and "23 - On and Off Book Cumulative Quantity" - Ratio / Multiplier Decimals for "Stats Update Type": "14 - Variation Last Price" and "18 - Percent Variation Previous Close" "16 - Trade Count" has no scale code.

5.4 TECHNICAL NOTIFICATION

Path	Json Response

json/equity_norway/instrument/MERK/2269194/TechnicalNotification	{
	"mDSeqNum": 453,
	"technicalNotificationType": "Instrument_Book_Retransmission_End",
	"rebroadcastIndicator": 0,
	"retransmissionStartTime": -1,
	"retransmissionEndTime": -1,
	"symbolIndex": 2269194
	}

Definition of attributes:

FIELD	SHORT DESCRIPTION

mDSeqNum	Assigned by MDG for each message. Each channel has its own Market Data Sequence Number sequence
technicalNotificationType	Indicates the technical notification sent.
rebroadcastIndicator	Indicates if this message is resent or new (1 if resent, 0 otherwise). For a snapshot, this field will always be set to '1'.
retransmissionStartTime	Indicates when the retransmission starts. For trade retransmission, all the trades previously received by the clients that have an "Event time" strictly lower than this field are valid (Time in number of nanoseconds since 01/01/1970 UTC).
retransmissionEndTime	Indicates when the retransmission ends. For trade retransmission, all the trades previously received by the clients that have an "Event time" strictly higher than this field are valid (Time in number of nanoseconds since 01/01/1970 UTC)
symbolIndex	Exchange identification code of the instrument/contract.

6. APPENDIX A – LIST OF MARKETS AVAILABLE

Optiq Segment	Market Name	MIC(s) Available on Market
Cash	**Funds ETFs classic**	XAMS/XAMC/XBRU/XLIS/XPAR/XPMC/XOSL/XMSM
	Funds ETF Access (XMLI MTF)	XMLI
	Funds ETF Italy (IT)	ETFP
	BONDs FXI	XPAR/XBRU/XAMS/ALXB/XOSL/ALXP/XMLI/XLIS
	Fixed Income Italy (MOT/extraMOT)	MOTX/XMOT
	Fixed Income ETLX (Bond-X)	ETLX
	BONDs Norway - Nordic ABM	XOAM (Saturn)
	Warrants & Certificates ETLX (Cert-X)	ETLX
	EQUITIES BElgium	XBRU
	EQUITIES PorTugal	XLIS
	EQUITIES IrEland	XMSM/XESM
	EQUITIES NOrway	XOSL/MERK/XOAS
	EQUITIES ITaly	MTAA /MIVX/ EXGM
	Equities ETLX	ETLX
	GEM	BGEM
	TRADING AFTER HOURS ITALY	XTAH
Derivatives	**Commodities power**	XEUC