Trading system dynamics

MEFF SMART

12 May 2023

Sensitivity: C1 Public

Contents

1 Introduction 3
1.1 The renewed MEFF Trading System 3

2 Components of the trading system 5
2.1 Core Trading System 5
2.2 Middleware 5
2.3 Logical architecture 6
2.4 Working modes 7

3 Some features of the different types of interfaces 8
3.1 UDP Market Data Server 8
3.2 Order Entry Server 8
3.3 FIX Protocol gateways 9

4 Deployment 10
4.1 Hardware and Operating System 10
4.2 Network equipment 10
4.3 Connectivity to UDP Market Data servers 11
4.4 Connectivity to the Order Entry Servers 11

1 Introduction

1.1 The renewed MEFF Trading System

After completion of the migration to be performed from 23 November 2020 to 28 February 2021, the MEFF Trading System for the Financial Derivatives segment will have a completely new structure that is worth to know in order to better understand the behaviour of orders, especially for those applications related to High Frequency Trading or Market Making.

The overall project has been conducted with the following main goals in mind:

- Improved response time, both in terms of average response time and its variability.
- Improved scalability and peak performance.
- New protocol interface more suitable for HFT applications, while still supporting the FIX protocol interface best suited for general-purpose applications.

The main strategies used to achieve these objectives are the following:

- Use UDP instead of TCP in internal communications so that a single write operation allows the transmission of one message to all recipients. Therefore:
  - There is an optimization in the message sending processes, that require less write operations.
  - The internal network handles less traffic.
  - All recipients are in equal conditions to receive and handle the message.
  - The number of Gateways within 1 hop from the Central Trading System can be scaled up without impacting the performance of the publishing processes.
- Reduce the number of messages published by the matching engine (TRADING process):
  - Several messages are now compacted into a single one, so that the common information is not written twice by the TRADING process. For instance, in the case of a trade, a single binary message is produced by the TRADING process with all the information about buyer and seller.
  - Every Gateway is able to take these messages and produce new shorter ones taking only the fields that need to be forwarded to the corresponding party. For instance, in the case of a trade between two different members, a public message without any information on buyer or seller is produced by the UDP Market Data Server, whereas an Order Entry Server will produce messages with just one of the legs.
- Provide a binary protocol which allows for:
  - Shorter message length.
  - Avoiding type conversion when formatting or parsing messages.

- Split some functionalities among several processes or process threads (e.g. input vs output message sequencing, message handling vs message persistence, …).
- Usage, where appropriate, of continuous polling vs an interrupt-driven approach.

The following sections go into further details about some of the components of the system, their features and the actual deployment.

2 Components of the trading system

2.1 Core Trading System

The core Trading System consists of several types of processes

Process name	Main functions
TREALBD	Maintain reference data
TRADING	Matching engine Order and quote management
CTRLTR	Control session and session status in each trading mode Scheduled transactions
OPEPRECON	Cross trades RFQs (Request for quotes)
GESSTD	Session Statistics
EXTFEED	Connection to external market data feeds for underlying price information

The MEFF Financial Derivatives segment has 10 different TRADING processes. Each one handles a specific partition of the Derivative contracts. Currently, TRADING 0 handles IBEX futures, TRADING 1 is for IBEX options, TRADING 2 is for the rest of futures and TRADING 3-9 handle stock options.

All contracts in the same partition can be recognized since they share the most significant byte of the field SecurityCode (see message Security List in the Binary Interface Specifications). They also have an independent numbering for SecondaryOrderID or for TrdMatchID, so that they do not overlap with each other.

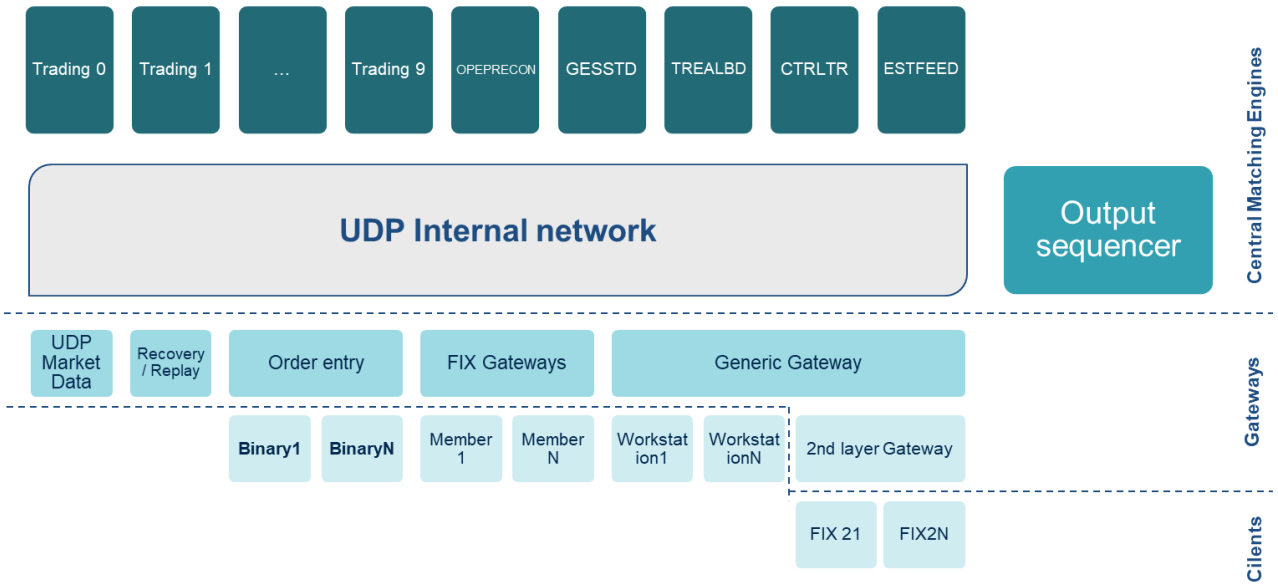
2.2 Middleware

The system middleware main processes are the following:

Process name	Main functions
SEQUENCER	Output sequencer of the messages coming from all TRADING processes and the rest of Core Trading System processes
UDPMARKETSERVER	Market Data feed producer
TCPREPLAYSERVER	Replay Server (to request missed messages from the Market Data feeds)

Process name	Main functions
TCPRECOVERYSERVER	Recovery Server (to request a snapshot of the Market Data feeds)
ORDERENTRYSERVER	Order Entry Server for binary orders
FASERVER	FIX protocol Gateway (both for market data and order entry in FIX protocol format)
GWSERVER	Generic Gateway (for Workstation and 2nd layer server connections)
MWAHCLIENT	Allows the connection of 2nd layer GWSERVER or FASERVER processes

2.3 Logical architecture



The previous diagram shows the different layers of processes. Client applications establish connections to one of the Gateways. For instance, a client application "Binary 1" can establish a connection with the Order Entry Server.

All the Gateways in the main Data Center have separate network interfaces to connect with member applications and to connect with the equipment where Core Trading System processes run. In the internal network that communicates the Core Trading System processes, the Output Sequencer and the Gateways, messages are transmitted using the UDP protocol, so that whenever there is a message with several recipients there is only one write operation.

For instance, an order on an IBEX Future contract sent by a client application to the Order Entry Server is forwarded to the appropriate TRADING process. The result of this order (e.g. an order confirmation) is published by the TRADING process and received by the Output Sequencer, which produces a single ordered feed with all the messages published by all TRADING processes and the rest of Core System processes and delivers it to all the Gateways (again with a single write operation). The different types of Gateways take the received message or parts of it and deliver the appropriate information to their clients. The Order Entry Server will send the order confirmation to the client that entered the order, but the UDP Market Data Server will take only a subset of the message fields and will produce new shorter messages to be distributed through the three different Market Data Feeds.

There is also the possibility for Member applications to use Gateways outside the main Data Processing Center. In this case, the 2nd layer Gateways establish a TCP connection with one of the Gateways in the main Data Processing Center.

2.4 Working modes

The system can be configured so it works either in Continuous Polling mode or as an Interrupt-Driven system. Continuous Polling mode implies more resource consumption and better response time, whereas Interrupt-Driven uses only the required amount of CPU time, but has a slightly slower response time.

Test environments are configured in Interrupt-Driven mode. The Production environment for Financial Derivatives is configured in Continuous Polling mode.

3 Some features of the different types of interfaces

3.1 UDP Market Data Server

The UDP Market Data Server processes the System output produced by the Output Sequencer and transforms it into 3 types of Multicast feeds:

- Full Depth:
  - Full depth during open market. Top-of-book during auctions.
  - Order book to be built by receiver based on orders and trades.
- Top of Book (depth 5)
- Top of Book (depth 1)

These feeds are produced by the same single process and in this order. Therefore, whenever there is an event to be published, it is first of all published always through the Full Depth feed, afterwards through the Top of Book Depth 5 feed, and finally through the Top of Book Depth 1 feed.

3.2 Order Entry Server

The Order Entry Server process handles all Member connections, and has two main threads: one for the messages coming from the Members to be sent to the Core Trading System and another one for the output messages received from the Output Sequencer.

Messages received from the Output Sequencer are read and, if seller and/or buyer are users of the Order Entry Server, they are transformed and sent to the appropriate destination. In case the write operation into the TCP socket that communicates with the Member cannot be immediately done, due to full TCP write buffers, the connection is logged off, so that network congestion with one counterparty doesn't affect the rest of connections. The Member is expected to reconnect once the application has recovered the corresponding delay.

Messages received from members are handled by a single thread that reads the messages coming from all Member connections and forwards them to the appropriate TRADING process.

New connections are handled in additional temporary threads until the moment they are fully synchronized. This approach allows that late connections don't interfere with currently active ones.

The message format received by the Order Entry Server is forwarded without transformation to the corresponding TRADING process. This fact, together with the shorter length of messages compared with FIX protocol messages (which are transformed into an internal message format) results in better response time for binary orders compared with FIX protocol orders.

Since Order Entry Servers and UDP Market Data Servers are independent processes, applications processing both cannot rely on any specific order of arrival of information regarding the same transaction.

A throttle mechanism is implemented regarding the maximum number of messages per second allowed. This limit is set up on a per connection basis. The control is implemented so that the number of messages sent by the Member application in the past 10 time slices of 0.1s is counted. If the number

of messages sent in this period is over the maximum configured, all new messages will be rejected until at least the next time slice of 0.1 seconds.

3.3 FIX Protocol gateways

The FIX Protocol gateways have one single-threaded FASERVER process per Member connection. The process typically is waiting in a read from the Member socket. When a FIX protocol message sent by the Member application is read by the FASERVER process, it is parsed, translated into an ASCII proprietary internal message format and forwarded to the appropriate internal process, typically a TRADING process.

This read operation can be interrupted if a new message coming from the host has arrived. In that case, if the message must be sent to the current connection user, it is translated into FIX Protocol and sent. In this case, network congestion may arise, and the FASERVER application would block in the write operation until the buffer is available.

Apart from the FASERVER processes, all FIX Protocol gateways have a single MWAHCLIENT process reading messages coming from the Sequencer. Whenever a new message is received from the Sequencer, the message is stored into an intermediate message file and a POSIX signal is sent to all FASERVER processes, so that they can interrupt the read socket operation and handle the newly arrived message.

A throttle mechanism is implemented regarding the maximum number of messages per second allowed. This limit is set up on a per connection basis. The control is implemented so that the number of messages sent by the Member application in the past 10 time slices of 0.1s is counted. If the number of messages sent in this period is over the maximum configured, all new messages will be stored in a buffer whose size is 50% of the limit + 1, and they will not be forwarded to the host until at least the next time slice of 0.1 seconds. Please note that this may result in the Member application noticing some response times apparently higher than 1 second. Messages that don’t fit into the buffer are rejected. It is strongly recommended that the application controls the message flow into the system.

4 Deployment

4.1 Hardware and Operating System

The hardware where the system is deployed are HP Server models DL560 or DL380 depending on the number of processors and cores required for each environment. The number of cores has been established so that processes that are running in continuous polling mode can devote one full core to each of the corresponding processes.

Some servers have been optimized in terms of speed (processor clock 3.6GHz) whereas others have been optimized in terms of capacity (processor clock 2.8GHz).

Server type	Hardware	Memory	Frequency
Matching Engines	DL560 4P 8C LD	192G	3.6GHz
Sequencer	DL380 2P 8C LD	96G	3.6GHz
UDP Market Data Server	DL380 2P 8C LD	96G	3.6GHz
Replay/Recovery	DL380 2P 16C LD	96G	2.8GHz
Order Entry Server	DL560 4P 16C LD	192G	2.8GHz
FIX Gateway	DL560 4P 16C LD	192G	2.8GHz
Generic Gateways	DL380 2P 16C LD	96G	2.8GHz

In the Production environment currently, all matching engines run in the same server, but the system is ready to have them split among different pieces of equipment if required for scalability reasons.

Each process has a hot-standby process running in another server at the same Data Processing Center, which is ready to automatically take the control in a matter of seconds if necessary. In the unlikely event of having to switch to a different process, the messages sent to the corresponding matching engine during the switching period would be ignored by the system.

Switching the operation to another Data Processing Center would only be done in the case of a major disaster affecting the main premises, and would always imply human intervention in order to activate the processes in the contingency site.

Test environments have the same type of equipment, but with a lower number of servers: several segments share the same hardware and several gateways also share the same infrastructure. This, together with the fact that test environments are configured as Interrupt-driven, results in better response time in Production than in Test environments.

The Operating System used is RedHat Linux version 7.7.

4.2 Network equipment

The servers are equipped with network cards Solarflare XtremeScale SFN8542 Dual Port 40Gb. The middleware takes advantage of the low-level libraries provided with these network cards for UDP connections, in order to achieve fast response times.

The internal network uses a couple of CISCO Nexus 3548XL so that there are two available paths to transmit information from one piece of equipment to another.

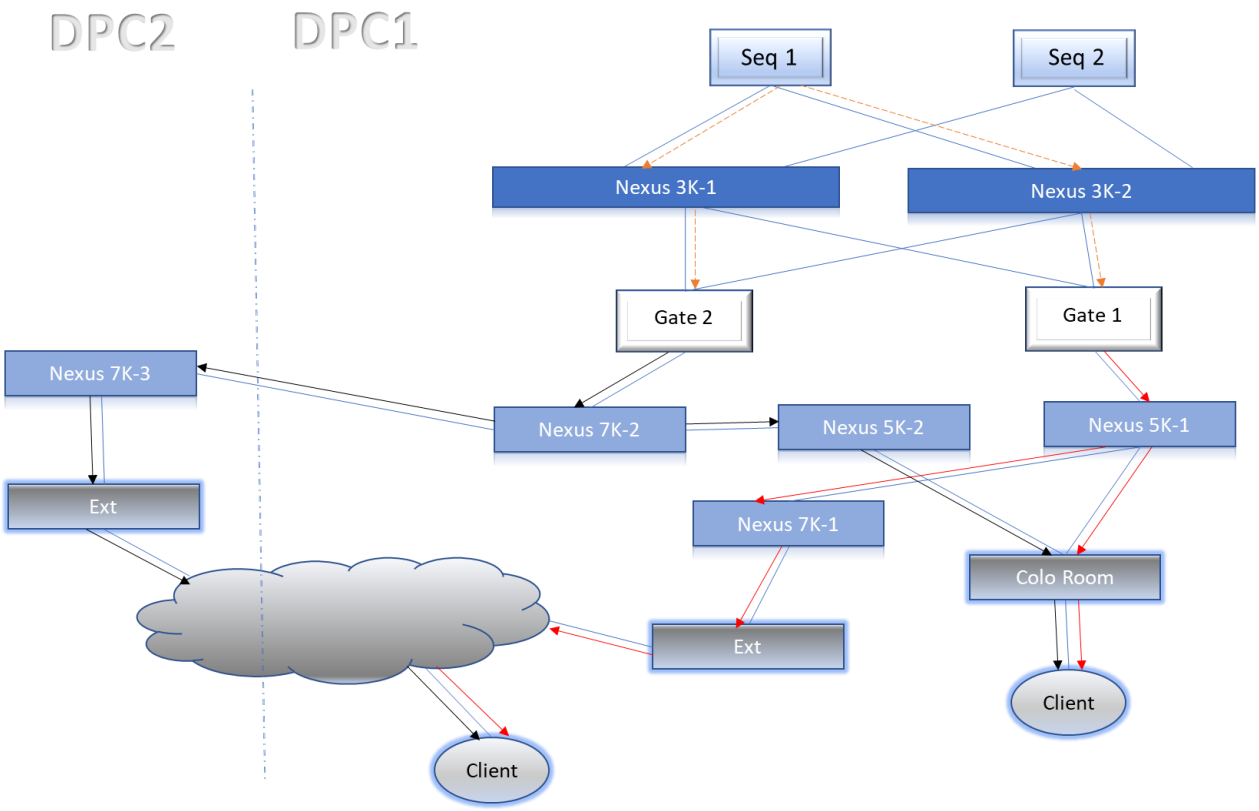
Members using the Co-location facilities are connected through a couple of CISCO Nexus 5K switches. The rest of members connect through routers located in two of the BME data centers, always providing a network without single points of failure.

4.3 Connectivity to UDP Market Data servers

The following diagram represents how the two UDP Market servers (Gateway Public A and Gateway Public B) are physically connected to the different network infrastructure elements. Both servers publish all the Market Data Feeds.

For Members with a co-location connection and under normal working circumstances, feed A is expected to be received earlier than feed B, since there is just one intermediate switch in the case of feed A compared with two intermediate switches for feed B.

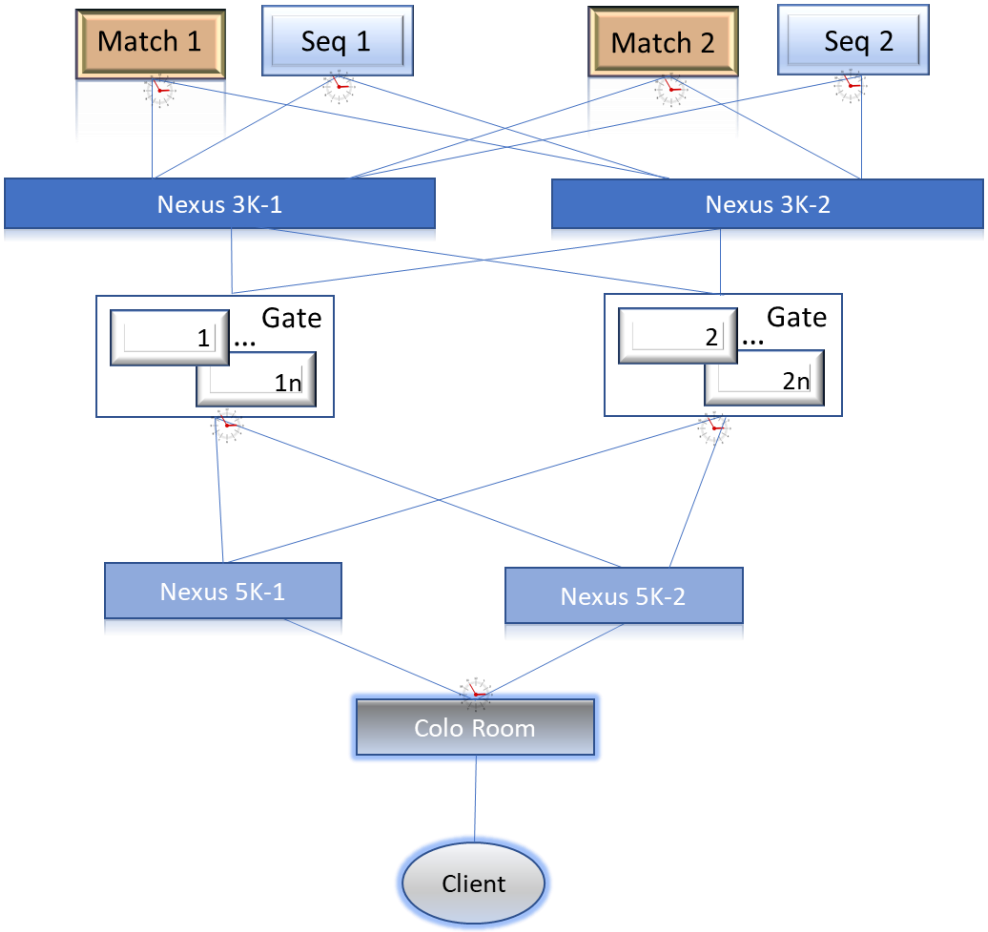
In the case of the rest of Members the number of intermediate steps is the same. The flow for feed A is distributed from the main Data Processing Center, whereas the flow for feed B is distributed from the alternate Data Processing Center (interconnected through two 24.6 km 10Gb communications line, which imply 122µs one-way latency). Depending on the Member location and the lines installed between each of the Data Centers and the Member premises, this can result in either one of the feeds to be received earlier than the other under normal circumstances.



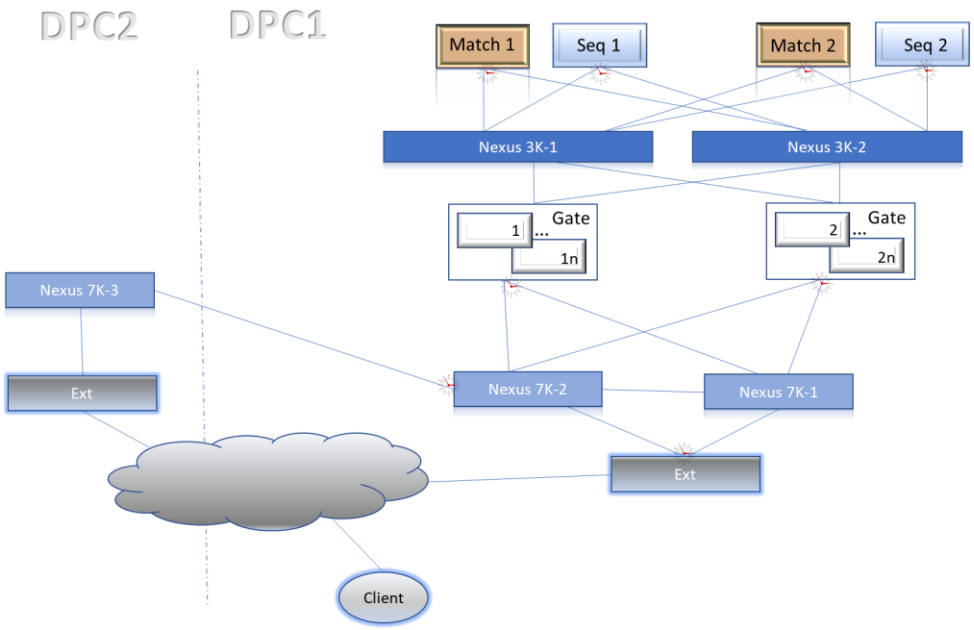
4.4 Connectivity to the Order Entry Servers

To take real advantage of the low-latency capabilities of the new binary Order Entry Servers, the connection from the Co-location service is highly recommended.

The Order Entry Server (and the rest of FIX Gateways and Generic Gateways as well) have a similar setup, so that the connection from the co-location site or from the member premises can be done through a path that implies minimum number of hops.



The connection to the Order Entry Server can also be established from the Member premises through the BME network, as shown in the following diagram.


DPC2
DPC1
Match 1
Seq 1
Match 2
Seq 2
Nexus 3K-1
Nexus 3K-2
1
Gate
1n
2
Gate
2n
Nexus 7K-3
Ext
Nexus 7K-2
Nexus 7K-1
Ext
Client

This material has been prepared by Bolsas y Mercados Españoles, Sociedad Holding de Mercados y Sistemas Financieros S. A. (BME), its subsidiaries, affiliates and/or their branches (together, "BME") for the exclusive use of the persons to whom BME delivers this material. This material or any of its content is not to be construed as a binding agreement, recommendation, investment advice, solicitation, invitation or offer to buy or sell financial information, products, solutions or services. The information does not reflect the firm positions (proprietary or third party) of the entities involved in the Spanish Securities Market. BME is under no obligation to update, revise or keep current the content of this material, and is subject to change without notice at any time. No representation, warranty, guarantee or undertaking – express or implied – is or will be given by BME as to the accuracy, completeness, sufficiency, suitability or reliability of the content of this material.

The opinions presented are theoretical and, therefore, the content hereof is intended for informational purposes only and should not be used for portfolio or asset valuations, or as the basis for any investment recommendations. Neither contributing Entities, nor Bolsas y Mercados Españoles, Sociedad Holding de Mercados y Sistemas Financieros S.A.(BME) nor any of its subsidiaries, accept responsibility for any financial loss or decision made based on the information contained in this material. In general, neither Bolsas y Mercados Españoles, Sociedad Holding de Mercados y Sistemas Financieros S. A. (BME) nor any of its subsidiaries, nor the contributing Entities, their directors, representatives, associates, subsidiaries, managers, partners, employees or advisors accept any responsibility for this information or unauthorised use of the same.

This material is property of BME and may not be printed, copied, reproduced, published, passed on, disclosed or distributed in any form without the express prior written consent of BME.

2023 Bolsas y Mercados Españoles, Sociedad Holding de Mercados y Sistemas Financieros S. A. All rights reserved.

BME
Plaza de la Lealtad,1
Palacio de la Bolsa
28014 Madrid

www.bolsasymercados.es

Sensitivity: C1 Public