



Cboe Europe MiFID II Identifier Management Application

VERSION 1.10

31 May 2019

This document has been established for informational purposes only. None of the information concerning the services or products described in this document constitutes advice or a recommendation of any product or service. To the extent that the information provided in this document constitutes a financial promotion as defined in the applicable legislation and regulation, it is only directed at persons who qualify as a "professional client" or "eligible counterparty" as defined in the applicable legislation and regulation. Persons who do not qualify should not act or rely upon it.

Contents

1	Introduction	4
1.1	Background	4
1.2	Functionality Provided	5
1.3	Access	5
1.3.1	Web UI	5
1.3.2	API	5
1.4	Backwards Compatibility	6
2	Register Identifiers	7
2.1	Upload	7
2.1.1	Heading and Data	7
2.1.2	Sample	8
2.2	Upload Results	8
2.2.1	Heading and Data	8
2.2.2	Sample	9
2.3	API Details	10
2.3.1	Command Details	10
2.3.2	Example Request (POST params)	10
2.3.3	Example cURL request	10
2.3.4	Response	10
3	View Missing Identifiers	11
3.1	Heading and Data	11
3.2	Sample	11
3.3	API Details	11
3.3.1	Command details	11
3.3.2	Example Request (GET)	11
4	View Identifiers	12
4.1	Heading and Data	12
4.2	Sample	12
4.3	API Details	12
4.3.1	Command details	12
4.3.2	Example Request (GET)	12
5	Corrections to Order Record Keeping Data Supplied via FIX or BOE	13
5.1	Submitting corrections	13
5.1.1	Heading and Data	13
5.1.2	Sample	14
5.2	Viewing submitted corrections	14
5.2.1	Heading and Data	14
5.2.2	Sample	14

6	Frequently Asked Questions for Registering Identifiers	15
7	Support	17
8	Revision History	18

1 Introduction

This document describes the Cboe MiFID II Identifier Management Application for use by Participants to manage the registration of long codes to short codes.

1.1 Background

In order to comply with RTS 24 - Record Keeping Requirements for Trading Venues, venues need to obtain the following data points from their customers:

- Submitting Member/Participant
- Client Identification Code (either a Legal Entity Identifier (LEI) or a National ID)
- An identifier for the trader or algorithm responsible for the investment decision
- An identifier for the trader or algorithm responsible for the execution
- Any non-executing broker involved (N/A for Cboe)
- Whether the order arose due to Direct Electronic Access (DEA) or not
- Whether the order represents some form of liquidity provision activity

The Cboe FIX and BOE protocols support collecting this information at the point of order entry. However the client identifier, the decision maker for the investment decision and the responsible entity for the execution is not provided at order entry in the form of the raw identifier (e.g. passport numbers, algorithm IDs etc), to help reduce the amount of sensitive data flowing through the trading system, as well as minimising message latency by keeping messages smaller.

On the order entry protocol, Participants supply an integer, known as the 'short code', for the identifiers that are relevant to their order flow. This code must accurately depict the entity involved in this activity and be persistent throughout the day. The short code associated with the order can not be changed.

Using the MiFID II Identifier Management Application outlined in this document, Participants must supply Cboe with a mapping from the 'short code' used on the order to a 'long code' (e.g. passport number, algorithm ID etc), referred to throughout the document as registering a long code to a short code.

The collection of this data is summarised in Figure 1 below.

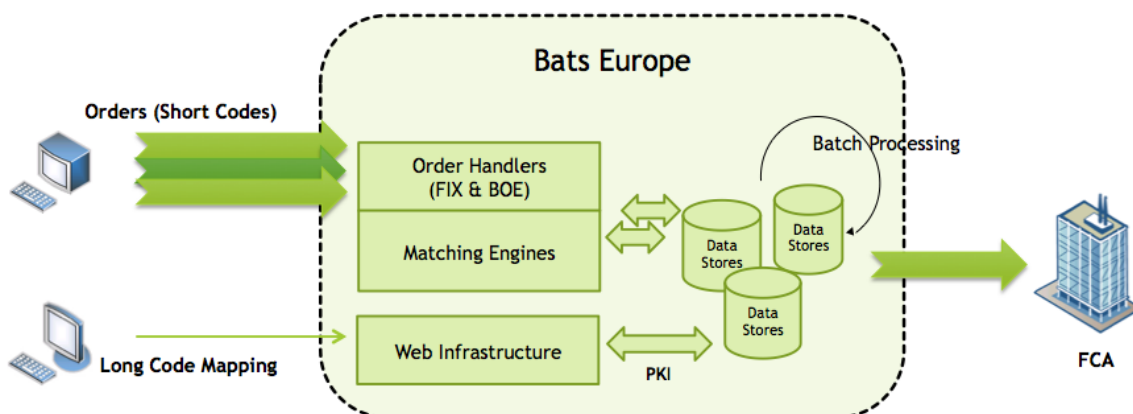


Figure 1: Collection of Record Keeping Identifier Information

With both the short and long form codes stored, Cboe then has the full information required under RTS 24, where the short codes are substituted with the full underlying data when requested by FCA.

1.2 Functionality Provided

The Cboe MiFID II Identifier Management Application for use by Participants provides the following functionality:

- **Register Identifiers:** Register long codes to short code identifiers by uploading a CSV file with the mapping, along with the effective date range and what type of client or a decision maker each identifier represents.
- **View Missing Identifiers:** Download a CSV file containing any short codes that have been supplied on orders via FIX or BOE, but which have not yet had a long code identifier registered against them for that date.
- **View Identifiers:** Download a CSV file containing the mappings between short and long codes. Long code identifiers are encrypted. Long codes are provided back to Participants in a partially masked form for security.
- **Submit Order Corrections:** Upload a CSV file containing corrections to the record keeping data originally supplied in real-time via FIX or BOE for individual orders. Download a CSV file containing the latest corrections for each order.

1.3 Access

1.3.1 Web UI

The MiFID II ID Management App is accessed via the Participant Tools section of the Cboe website for permissioned users using the HTTPS protocol and is served from the www.batstrading.co.uk domain.

Production: CXE, BXE and DXE

Identifiers: https://www.batstrading.co.uk/account/identifiers_management/

Order Corrections: https://www.batstrading.co.uk/account/rts24_corrections/

Certification: CXE, BXE and DXE

Identifiers: https://certification.batstrading.co.uk/account/identifiers_management/

Order Corrections: https://certification.batstrading.co.uk/account/rts24_corrections/

1.3.2 API

You must be able to use the Cboe Secure Web API. General details on how to access this are available in the Specification Document. The provided API allows you to register and view Identifiers. Submitting and viewing corrections must be done through the member portal and cannot be done via the API.

The MiFID II ID Management App is accessed via a URL using the HTTPS protocol and is served from the api.batstrading.co.uk domain.

Production: CXE, BXE and DXE

https://api.batstrading.co.uk/account/identifiers_management/

Certification: CXE, BXE and DXE

https://api.certification.batstrading.co.uk/account/identifiers_management/

Sensitive data is encrypted before storage and can not be decrypted by the web server, although a masked version is available to help in any reconciliation Participants may wish to perform.

1.4 Backwards Compatibility

The CSV files specified in this document are each described in sections 2.1.1, 2.3.4, 3.1, 4.1, 5.1.1 and 5.2.1. Each row in the tables listed in these sections describe the format of a CSV file data column. To aid backward compatibility, columns will never be removed or reordered. New columns, however, may be added. It is highly recommended that parsers be able to ignore new columns which are added over time.

2 Register Identifiers

2.1 Upload

Having provided short codes on the individual orders entered into the exchange, Participants must also, on either a pre- or post-order basis (by the end of the trading day at 18:00), provide the mapping to the long code, which will be either a Legal Entity Identifier (LEI), a National ID or an Algorithm ID. This mapping is performed via the 'Register Identifiers' function provided by the MiFID II Identifier Management Application, by uploading a CSV file to Cboe using the format described in the following sub-sections. Participants can upload CSV files with a mix of different types of client, investment decision maker or execution decision maker identifier, or separate files for each type.

The decision as to whether to pre- or post-register is entirely within the Participant's control. If, for example, you have a limited set of traders, you may choose to register all of them immediately and only perform updates in line with staffing changes. On the other hand, you might prefer to only inform us about a client once that client has had an order sent to Cboe. In any case, data must be provided by the end of the trading day at 18:00, of the day in which a short code was utilised. Previous registrations carry across to future dates; you only need to tell Cboe about changes.

Upon uploading a CSV file to Cboe, a results file will be returned to the user as outlined in section 2.2.

Any change to a previous short code mapping will result in future mappings for that short code to be removed, so a change in a short code mapping that is not the most recent will require you to re-upload all subsequent mappings for that short code. Refer to section 6 for guidance on how to amend the end date of an existing registration or change the long code registered to a short code.

2.1.1 Heading and Data

Column Name	Data Type	Description
Short Code	Integer	Values between 4 to 4,294,967,295 are permitted. Values 0 to 3 are reserved. Attempting to register a long code against these values will result in the registration failing.
Long Code	String	Algorithm ID, Legal Entity Identifier (LEI) or Natural Person Identifier (PI).
Identifier Type	String	Used to specify which of the six unique ranges the short code is for. Any of the following string values are permitted: Client-Person Client-Entity InvestorDecisionMaker-Person InvestorDecisionMaker-Algo ExecutionDecisionMaker-Person ExecutionDecisionMaker-Algo
Effective Date	ISO Date	Date in the format of YYYY-MM-DD for when this registration is effective from. The effective date cannot be more than 180 days prior to the date the upload is being performed on; from July 1st 2018 this will be reduced to 30 days prior.

End Date	ISO Date	Date in the format of YYYY-MM-DD if an end date for the registration is known, otherwise leave null (empty) if an end date for the registration is not known. Refer to section 6 for guidance on how to amend the end date of an existing registration or change the long code registered to a short code. If an end date is supplied, the date cannot be more than 180 days prior to the date the upload is being performed on; from July 1st 2018 this will be reduced to 30 days prior.
----------	----------	--

2.1.2 Sample

Example CSV dataset:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
27,221454659,Client-Person,2017-06-01,2018-01-01
27,13341233423,Client-Person,2018-01-02,
29,335123053,Client-Entity,2017-02-01,
30,ABCx76x,InvestorDecisionMaker-Person,2017-01-01,
31,14241424,ExecutionDecisionMaker-Algo,2017-01-15,
```

CSV files must be UTF-8 encoded. Byte Order Marks are not supported.

2.2 Upload Results

Upon uploading a CSV file to Cboe as per section 2.1 a results file will be returned to the user to indicate whether the registrations were successful.

If the CSV upload file format is invalid then the upload will fail straight away with an indication stating the file has not been processed.

If however the file is correctly formatted then each registration will be processed. Each registration attempted will result in the corresponding row in the upload file being echoed into the results file line of the CSV file, with an additional column indicating whether the registration was successful or not.

2.2.1 Heading and Data

Column Name	Data Type	Description
Short Code	Integer	The short code this result row is describing.
Long Code	String	The long code attempting to be registered against the short code.
Identifier Type	String	The type of identifier: Client-Person Client-Entity InvestorDecisionMaker-Person InvestorDecisionMaker-Algo ExecutionDecisionMaker-Person ExecutionDecisionMaker-Algo
Effective Date	ISO Date	Start date of registration

End Date	ISO Date	End date of registration. Null (empty) if an end date for the registration was not specified upon upload.
Processed	Single Character	Y = Registration successful N = Registration unsuccessful
Message	String	This is to provide Participants with more details on the reason for a submission not being processed.

2.2.2 Sample

Example CSV dataset. In this example the second row was not processed because the effective date could not be processed.

```
Short Code,Long Code,Identifier Type,Effective Date,End Date,Processed,Message
65,*****1356",Client-Person,2017-09-01,,Y,
66,*****4455,Client-Persons,201-01-02,,N,Invalid Identifier Type display name: Client-Persons
67,,InvestorDecisionMaker-Person,2017-01-01,,N,Missing Long Code
68,*****1424,ExecutionDecisionMaker-Algo,2017-01-15,,Y,
```

2.3 API Details

This is the API equivalent of submitting short code mappings through the web UI. All CSV's submitted through the API must match the format outlined in section 2.1.

The request must be made using multipart form submission with the file passed through the 'file' field in the POST request. Currently this is not possible in Wget, so a sample cURL request has been provided.

2.3.1 Command Details

Parameter	Description/Value	Ordinality
command	registerIdentifiers	Mandatory
file	file containing registration data	Mandatory

2.3.2 Example Request (POST params)

```
command = bulkRegisterIdentifiers
file (contents of CSV file) =
Short Code,Long Code,Identifier Type,Effective Date,End Date
4,444444444,Client-Person,2017-01-01,2018-01-01
5,555555555,Client-Person,2018-01-02,
6,666666666,Client-Entity,2017-02-01,
```

2.3.3 Example cURL request

```
curl -u user:pass -F "file=@/path/to/csv.csv"
https://api.batstrading.co.uk/account/identifiers_management/?command=registerIdentifiers
```

2.3.4 Response

Assuming the CSV file isn't corrupt in any way, the CSV file outlined in section will be returned. The file will be named in the following manner: FIRM_MPID_CURRENTDATE_identifier_management_upload_results.csv

3 View Missing Identifiers

Using the 'View Missing Identifiers' function provided by the MiFID II Identifier Management Application, Participants can download a CSV file containing all of the short codes that do not have a long code registered against them for days prior to today.

3.1 Heading and Data

Column Name	Data Type	Description
Short Code	Integer	The short code this row is describing.
Identifier Type	String	The type of identifier: Client-Person Client-Entity InvestorDecisionMaker-Person InvestorDecisionMaker-Algo ExecutionDecisionMaker-Person ExecutionDecisionMaker-Algo
Start Date	ISO Date	Start date of the period where the short code does not have a long code registered against it.
End Date	ISO Date	End date of the period where the short code does not have a long code registered against it.

3.2 Sample

Example CSV dataset:

```
Short Code,Identifier Type,Start Date,End Date  
100,Client-Person,2017-07-01,2017-07-01  
102,InvestorDecisionMaker-Person,2017-06-27,2017-07-01  
110,ExecutionDecisionMaker-Algo,2017-06-26,2017-06-28
```

3.3 API Details

This is the API equivalent of downloading the data using the web UI. The response will be the CSV file outlined above. The file will be named in the format of:

```
FIRM_MPID_CURRENTDATE_identifier_management_missing_registrations.csv.
```

3.3.1 Command details

Parameter	Description/Value	Ordinality
command	viewMissingRegistrations	Mandatory

3.3.2 Example Request (GET)

```
?command=viewMissingRegistrations
```

4 View Identifiers

Using the 'View Identifiers' function provided by the MiFID II Identifier Management Application, Participants can download a CSV file containing all of the short codes that have a long code registered.

Note that long codes are encrypted as soon as Cboe receive them and thus it is not possible to provide unencrypted values. Cboe provides a masked version of the long code for partial identification of long codes.

4.1 Heading and Data

Column Name	Data Type	Description
Short Code	Integer	The short code this row is describing.
Long Code	String	The long code registered against the short code for the period specified by the Effective Date and End Date columns. The long code is partially masked for security.
Identifier Type	String	The type of identifier: Client-Person Client-Entity InvestorDecisionMaker-Person InvestorDecisionMaker-Algo ExecutionDecisionMaker-Person ExecutionDecisionMaker-Algo
Effective Date	ISO Date	Date the long to short code mapping commenced.
End Date	ISO Date	Date the long to short code mapping ended. Null (empty) if mapping is still active.

4.2 Sample

Example CSV dataset:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
27,*****4659",Client-Person,2017-06-01,2018-01-01
27,*****3423,Client-Person,2018-01-02,
29,*****3053,Client-Entity,2017-02-01,
30,***x76x,InvestorDecisionMaker-Person,2017-01-01,
31,***1424,ExecutionDecisionMaker-Algo,2017-01-15,
```

4.3 API Details

This is the API equivalent of downloading the data via the web UI. The response will be the CSV file outlined above. The file returned will be named in the format of:

FIRM.MPID.CURRENTDATE.identifier_management.registered.identifiers.csv.

4.3.1 Command details

Parameter	Description/Value	Ordinality
command	viewIdentifiers	Mandatory

4.3.2 Example Request (GET)

```
?command=viewIdentifiers
```

5 Corrections to Order Record Keeping Data Supplied via FIX or BOE

Corrections to the Order Record Keeping data originally supplied on orders in FIX or BOE messages can be submitted through the RTS 24 Corrections section of the Cboe Europe website for permissioned users. The Corrections section allows you to submit Order Record Keeping data corrections, receive feedback on errors that occurred during your submission and to download a CSV with the latest corrections for each submitted order.

Until July 1st 2018 orders that are less than 180 days old can have corrections submitted. This behavior will change after July 1st 2018 to only allow corrections of orders that are less than 30 days old.

Each uploaded file cannot be larger than 10 megabytes in size. If you want to submit larger files, or orders that are older than this limit, you should contact Cboe Europe Trade Desk directly.

5.1 Submitting corrections

Order Record Keeping data corrections can be made by submitting a CSV file containing a line per order ID. Section 5.1.1 contains a list of columns that the submitted CSV should contain. The Cboe Europe FIX Specification contains a list of allowed values for MiFID II short code fields.

Upon submission of a correctly-formatted CSV file, your submission will be processed asynchronously. The page will refresh periodically until the submission has been completely processed, upon which you will be able to see the result of your submission.

Submitting corrections via this application does not actually alter the original Order Record Keeping data supplied on the order via FIX and BOE, so when downloading the a list of registered long to short code mappings via the View Identifiers section of the application, the original values will still be returned. Similarly, even if a correction to a short code has been supplied, the original incorrect short code still needs to have a long code registered (even if the long code is a placeholder stating the long code is invalid and a correction has been provided), otherwise the original incorrect short code will appear in the Missing Identifiers section of the application.

Any corrections submitted are stored separately from the data originally supplied on the order and can be viewed via the View Corrections section of the application.

5.1.1 Heading and Data

Column Name	Data Type	Description
Order ID	String	A 12-character alphanumeric string that corresponds to the FIX <i>OrderID</i> field. Mandatory.
Order Date	ISO Date	The date of the order in 'YYYY-MM-DD' format corresponding to the FIX <i>TransactTime</i> (60) field. Mandatory. The date of the order should be less than 180 days ago.
Symbol	String	The symbol of the order as specified in the FIX <i>Symbol</i> (55) field. Mandatory.
Order Capacity	String	A corrected value for the FIX <i>OrderCapacity</i> (47) field. Mandatory.
Algorithmic Trade	Boolean	A corrected value for the FIX <i>AlgorithmicTradeIndicator</i> (2667) field. Mandatory.
Liquidity Provision Activity	Boolean	A corrected value for the FIX <i>OrderAttributeTypes</i> (8015) field. A value of True implies that this order should be treated as a Liquidity Provision Activity order. Mandatory.
Client Code	Integer	The corrected short code value for the FIX <i>PartyID</i> (448) field for PartyRole 3 (Client). Empty strings are allowed. If a non-empty string is supplied, <i>Client Qualified Role</i> must also be present.
Client Qualified Role	Integer	The corrected short code value for the FIX <i>PartyRoleQualifier</i> (2376) field for PartyRole 3 (Client). Empty strings are allowed. If a non-empty string is supplied, <i>Client Code</i> must also be present.
Execution Code	Integer	The corrected short code value for the FIX <i>PartyID</i> (448) field for PartyRole 12 (Executing Decision Maker). Empty strings are allowed. If a non-empty string is supplied, <i>Executor Qualified Role</i> must also be present.
Executor Qualified Role	Integer	The corrected short code value for the FIX <i>PartyRoleQualifier</i> (2376) field for PartyRole 12 (Executing Decision Maker). Empty strings are allowed. If a non-empty string is supplied, <i>Execution Code</i> must also be present.

Investment Code	Integer	The corrected short code value for the FIX <i>PartyID</i> (448) field for PartyRole 122 (Investment Decision Maker). Empty strings are allowed. If a non-empty string is supplied, <i>Investor Qualified Role</i> must also be present.
Investor Qualified Role	Integer	The corrected short code value for the FIX <i>PartyRoleQualifier</i> (2376) field for PartyRole 122 (Investment Decision Maker). Empty strings are allowed. If a non-empty string is supplied, <i>Investment Code</i> must also be present.

5.1.2 Sample

Example CSV dataset:

```
Order ID,Order Date,Symbol,Order Capacity,Algorithmic Trade,Liquidity Provision Activity,Client Code,
Client Qualified Role,Execution Code,Executor Qualified Role,Investment Code,Investor Qualified Role
BATS00000001,2017-10-01,VOD1,A,t,f,123,24,123,24,123,24
BATS00000002,2017-11-12,BMWd,P,t,f,123,24,123,24,123,24
```

5.2 Viewing submitted corrections

For each submitted CSV file you will be able to see the number of rows that were imported, the number of rows that had errors, and a link to download a CSV file of all the rows in the submission. This CSV file contains additional information per row about any potential validation errors that occurred.

You can download a CSV file containing the latest successfully submitted corrections per order ID. This file only contains corrections that did not have any validation errors and does not contain any order IDs for which no corrections were submitted.

The format of the per-submission CSV file and the latest corrections CSV file is identical and corresponds to the format specified in section 5.2.1.

5.2.1 Heading and Data

Both the latest CSV file and each individual submission's CSV file contain all of the columns as specified in section 5.1.1. The following additional columns are also present.

Column Name	Data Type	Description
Submission ID	Integer	The sequential numeric ID of the submission belonging to the correction.
Error Message	String	A string containing one or more errors that occurred during the validation of the correction. Each error is separated by the ";" (semicolon) character. A non-empty string indicates a correction that is rejected by Cboe.

5.2.2 Sample

Example CSV dataset of a single submission result:

```
Submission ID,Order ID,Order Date,Symbol,Order Capacity,Algorithmic Trade,Liquidity Provision Activity,Client Code,
Client Qualified Role,Execution Code,Executor Qualified Role,Investment Code,Investor Qualified Role,Error Message
1,BATS00000001,2017-10-01,VOD1,A,t,f,123,24,123,24,123,24,
1,BATS00000002,2017-11-12,BMWd,P,t,f,123,24,123,24,1,24,Order ID not found;Invalid Execution Code
1,BATS00000003,2016-01-12,BMWd,P,t,f,123,24,123,24,123,24,Order must be less than 180 days old
```

Example CSV dataset of the latest corrections for each order:

```
Submission ID,Order ID,Order Date,Symbol,Order Capacity,Algorithmic Trade,Liquidity Provision Activity,Client Code,
Client Qualified Role,Execution Code,Executor Qualified Role,Investment Code,Investor Qualified Role,Error Message
3,BATS00000001,2017-10-01,VOD1,A,t,f,123,24,123,24,123,24,
1,BATS00000002,2017-11-12,BMWd,P,t,f,123,24,123,24,1,24,
```

6 Frequently Asked Questions for Registering Identifiers

i. How do I register a different long code against a short code that already has a long code registered against it without an end date?

In this scenario MYLONGCO14 was initially registered to short code 3000000003 from 2018-01-02 with no end date. To achieve this, the following CSV mapping file was used:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000003,MYLONGCO14,Client-Person,2018-01-02,
```

When downloading the long code registration file to confirm the registrations in place, it would contain the following entry for short code 3000000003:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000003,*****014,Client-Person,2018-01-02,
```

If the Participant then wanted to register a different long code (MYLONGCO15) against short code 3000000003 effective from 2018-01-18, the following CSV mapping file could be used:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000003,MYLONGCO15,Client-Person,2018-01-18,
```

Upon downloading the long code registration file, note that the new mapping for short code 3000000003 has resulted in the end date of the previous registration automatically being set to the date prior to the effective date of the new mapping:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000003,*****015,Client-Person,2018-01-18,
3000000003,*****014,Client-Person,2018-01-02,2018-01-17
```

ii. How do I set an end date for a long code registration that currently does not have an end date?

In this scenario MYLONGCO15 was initially registered to short code 3000000004 from 2018-01-18 with no end date. This is how the registration would be presented in the long code registration file downloaded from the website:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000004,*****015,Client-Person,2018-01-18,
```

If the Participant then wanted to set an end date for this registration, the below CSV mapping file which sets an end date could be used. If the long code registration file was then downloaded, the end date would be set accordingly. If the Participant then wanted to further adjust the end date for this registration, the same CSV mapping entry could be used, but with the end date changed accordingly.

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000004,MYLONGCO15,Client-Person,2018-01-18,2018-01-25
```

iii. One or more long codes are already registered against a single short code. A different long code should have been registered to the short code for a period of time within a date range where a registration already exists; how can this be achieved?

In this scenario, the Participant initially wanted MYLONGCO17 to be registered against short code 3000000006 from 2018-01-02 until 2018-01-11 and then wanted MYLONGCO18 to be registered against the same short code from 2018-01-12 until 2018-01-27. To achieve this, the following CSV mapping file was used:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000006,MYLONGCO17,Client-Person,2018-01-02,2018-01-11
3000000006,MYLONGCO18,Client-Person,2018-01-12,2018-01-27
```

When downloading the long code registration file to confirm the registrations in place, it would contain the following entries for short code 3000000006:

```
Short Code,Long Code,Identifier Type,Effective Date,End Date
3000000006,*****018,Client-Person,2018-01-12,2018-01-27
3000000006,*****017,Client-Person,2018-01-02,2018-01-11
```

At a later date the Participant realises a different long code (MYLONGCO19) should have been registered against short code 3000000006 from 2018-01-02 until 2018-01-10; note that this new registration falls within the date range that MYLONGCO17 is already registered against short code and that MYLONGCO18 is also registered to the short code for a period of time afterwards. To change the long code registered for 2018-01-02 until 2018-01-10, the following CSV mapping file could be used:

```
3000000006,MYLONGCO19,Client-Person,2018-01-02,2018-01-10
```

Upload downloading the long code registrations file, the new registration is in place as illustrated below, but any existing registrations that were **after** the new registration end date have been removed. Any registrations after the end date of the new registration are deemed invalid and are therefore removed. If they are still required, the registrations need to be re-uploaded.

```
Short Code,Long Code,Identifier Type,Effective Date,End Date  
3000000006,*****019,Client-Person,2018-01-02,2018-01-10
```


7 Support

If you have any queries regarding MiFID II Identifier Management Application, please contact your account manager directly. Other useful numbers are set out below.

Sales Team

SalesEurope@cboe.com

Phone: +44.207.012.8906

Participant Services

ParticipantServicesEurope@cboe.com

Phone: +44.207.012.8902

Trade Desk

TradeDeskEurope@cboe.com

Phone: +44.207.012.8901

8 Revision History

Protocol Version	Date	Description
1.0	10/02/2017	Initial Version.
1.1	02/03/2017	Added in API docs.
1.2	20/03/2017	Corrected valid values range for Short Code
1.3	29/11/2017	Clarified time by which long codes need to be registered at the end of the trading day.
1.4	15/12/2017	Added new section on how to submit and view corrections to record keeping data originally supplied via FIX or BOE.
1.5	19/01/2018	Added headings to example CSV files. Clarified wording for order corrections section. Updated Register Identifiers command to limit the Effective and End Dates to be no more than 90 days prior to the date of upload. Added guidance on how to amend the end date of an existing registration or change the long code registered to a short code.
1.6	02/02/2018	Added CSV file encoding requirements to section 2.1.2
1.7	08/03/2018	Added "Message" heading in sections 2.2.1 and 2.2.2
1.8	09/04/2018	Updated Register Identifiers command to limit the Effective and End Dates to be no more than 180 days prior to the date of upload, until July 1st when it will become 30 days prior. Clarified that amending a previous short code mapping will result in future mapping of that short code being deleted.
1.9	12/06/2018	Clarified how corrections to short codes are handled by the corrections application.
1.10	31/05/2019	Added DXE environment.