



**THE FOUNDATION
FOR SECURE
MARKETS**

ENCORE Data Distribution Services Guide
Overview and Implementation Reference

Version 1.18
June 2020

Reasonable measures are taken by OCC to ensure the accuracy of the information it distributes in its DDS program. This information is produced from data received from a number of different sources, which are believed to be reliable. However, due to the number of sources for such data, the possibility of human error, and the risks inherent in electronic distribution, there may be omissions or inaccuracies in such information and delays or interruptions in providing it. Accordingly, OCC disclaims all express or implied warranties with respect to the information distributed in its DDS program, including any warranty of merchantability or fitness for a particular purpose. Further, information sent on a real time basis should not be considered final until OCC issues an end of day message advising no additional transmissions will be made on a particular business day.

Contents

DOCUMENT ORGANIZATION	4
GLOSSARY OF TERMS	4
DDS SUBSCRIPTION CONCEPTS	6
SUBSCRIBERS, RECIPIENTS AND PACKAGES.....	6
REAL-TIME END OF DAY MESSAGES	7
DATA DELIVERY PROCESSING	8
BATCH FLOW.....	8
REAL-TIME FLOW.....	10
SYSTEM COMPONENTS / REQUIREMENTS	11
HARDWARE	11
SOFTWARE FOR PUSH FILE DELIVERY.....	11
SOFTWARE FOR PULL FILE DELIVERY	12
SOFTWARE FOR REAL-TIME MESSAGING.....	13
FIXML SCHEMA CONCEPTS	14
USAGE	14
FIXML AND FIXML EXTENSIONS VERSION IDENTIFICATION	15
FIX CONCEPTS	17
CFI CODE	17
MARKET IDENTIFIER CODE (MIC)	20
PARTY COMPONENT BLOCK	21
INSTRUMENT COMPONENT BLOCK	22
FIXML DATA TYPES (AS USED BY OCC)	23
UTC TIMESTAMP	25
IMPLEMENTATION DETAIL ON EMPTY VALUES	25
FIX REFERENCE MATERIALS	26
APPENDIX	27
REVISION HISTORY	27

Document Organization

This document is one of a set of three intended to provide a detailed description of all aspects of the new Data Distribution Services system:

Part One: Data Distribution Services Overview. This section is intended for a reader that needs to understand the Data Distribution Services (DDS) system. This part of the guide contains the information one would need to read in order to understand the design concepts and the data delivery services being developed and implemented.

Part Two: Developer Reference – Non-Proprietary Transmissions. This section is intended for use as a Non-Proprietary Transmission mapping reference for FIXML developers. This part of the guide includes FIXML elements, transmission layouts, message structures, and samples.

Part Three: Developer Reference – Proprietary Transmissions. This section is intended for use as a Proprietary Transmission mapping reference for FIXML developers. This part of the guide includes FIXML elements, transmission layouts, message structures, and samples for each transmission.

Glossary of Terms

You should be familiar with the following terms prior to reading this guide.

Batch – In a computer, a batch job is a program that is assigned to the computer to run without further user interaction. In larger commercial computers or servers, batch jobs are usually initiated by a system user. Some are defined to run automatically at a certain time.

DDS (Data Distribution Services) — DDS supports both batch and real-time data delivery and will utilize the FIXML data formatting standard.

ENCORE – The clearing system at OCC.

Event Driven Processing – A business event is a meaningful change in the state of the enterprise, such as the opening of a new customer account, clearing a trade, or the matching of a transfer. Event driven processing is system behavior that is initiated by these business events rather than system events—such as time based scheduling. Event driven systems possess the following attributes: 1) Individual treatment of transactions; 2) Push delivery systems; and 3) Electronic notification.

FIXML (Financial Information eXchange Markup Language) – The XML derived grammar of the FIX protocol. A FIXML implementation will have message format validation, cleaner, more expressive structure, and leverage existing standards. The initial goal is to provide the ability to embed FIXML messages within traditional FIX header and trailers to minimize the impact on existing implementations.

Messaging – There are two major messaging server models: the point-to-point model and the publish/subscribe model. Messaging allows programs to share common message-handling code, to isolate resources and interdependencies, and to easily handle an increase in message volume. Messaging also makes it easier for programs to communicate across different programming environments (languages, compilers, and operating systems) since the only thing that each environment needs to understand is the common messaging format and protocol.

Package – A Package is a collection of DDS transmissions that are grouped together based on selections made when the subscription was created.

Pull Delivery Model – In this information delivery model, the observer—or client—requests information from the information owner. An example of this model is the download of a document from a web page.

Push Delivery Model – In this information delivery model, the information owner distributes the data to the observer as it deems appropriate. An example of this model is the sending and delivery of an email message.

Real-Time – A level of computer responsiveness that a user senses as sufficiently immediate or that enables the computer to keep up with some external process (for example, to present trade data as trades are executed and cleared). *Real-time* is an adjective pertaining to computers or processes that operate in real time. Real time describes a human rather than a machine sense of time.

Recipient – The entity (Clearing Member Organization, Exchange, Regulatory Agency or Service Bureau) that owns the systems where DDS delivers data for processing or retransmission.

STP (Straight-Through-Processing) – The seamless integration of systems and processes to automate the trade process from end-to-end--trade execution, confirmation and settlement--without the need for manual intervention or the re-keying of data.

Subscriber – The entity (a Clearing Member Organization, Exchange, or Regulatory Agency) that requests a package of transmissions and owns the data that is transmitted to recipients.

XML (eXtensible Markup Language) – A simple and flexible text format derived from SGML (ISO 8879). Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere. Special purpose XML languages and standards are commonly developed with several hundred already adopted since XML 1.0 was released in February 1998.

DDS Subscription Concepts

DDS creates a flexible facility within ENCORE to maintain the profiles of the organizations that subscribe and/or receive data service, series, or prices from OCC. The DDS subscription system provides a centralized view of all of these profiles from a master file setup and audit perspective.

The following are the main entities that make up the DDS subscription system:

- Subscribers
- Recipients
- Packages

Subscribers, Recipients and Packages

A subscriber represents the entity (clearing member, exchange, regulatory agency) that is the final beneficiary of DDS data.

A recipient represents the entity (clearing member, exchange, regulatory agency or service bureau) that owns the systems where DDS data will be delivered for processing or retransmission.

Flexibility in the setup of subscriber and recipient profiles is achieved through the various setup scenarios that are available:

- An entity can act as a subscriber and recipient at the same time.
- A subscriber can have its data distributed to one or more recipients.
- A recipient can receive data for multiple subscribers.

For organizations that want to receive proprietary data from the ENCORE DDS module, a minimum of one subscription needs to be defined. For Clearing Members that require the separation of data files between groups of accounts, a separate subscription will be created for each group of accounts. Security will follow the same data-level security protocols that are applied to accounts in ENCORE.

Once a subscription has been created, the DDS transmissions that will be received for that subscription need to be defined. Organizations have the ability to bundle one or more transmissions as a package (e.g., Exercises and Assignment transmissions in an E&A Package, Prices, Security List and Security Definition transmissions in a Master File Package).

Each of the packages set up are then assigned one or more recipient destinations. A recipient could be the same organization that subscribes to the data service or a different entity (e.g., a service bureau). A recipient destination can be a file to be pushed to the recipient's data center, a file to be pulled by recipient's data center, or a message queue (MQ) where messages are delivered in a real-time mode. Although all transmission types are available in batch mode, not all are available in a real-time mode.

Packages to be delivered in batch mode will become available in their entirety when processing of the last transmission in the package is complete. Messages for transmissions included in packages to be delivered in real time will be sent out as soon as the individual messages are

generated within the ENCORE system.

The maintenance within ENCORE of the subscribers, recipients and packages will be performed by Member Services at the request of organizations subscribing or receiving DDS transmissions.

Real-Time End Of Day Messages

For all transmissions that can be delivered in a real-time mode (via MQ), an end of day (EOD) message will be transmitted. An EOD message will be sent for all messages associated with a specific transmission when the transmission is complete. The EOD message will:

- Indicate that no more messages will be sent for the transmission associated with the EOD message;
- Provide a total count of messages associated with the EOD message for the given cycle;

EOD messages will be delivered only to those organizations that subscribe to the data in real-time mode.

A sample message is provided below:

```
<FIXML ... >
<DDSEODMessage
  BizDt="2005-01-09"
  MsgTypeCode="TRADE"
  SchemaVer="FIX 4.4"
  TransType="TRADES"
  TransSubType="MATCHED"
  TransProductSet="OPTN"
  FinalizationCycle="ENCORE Equity Index Finalization"
  NoMessagesSent="177966" />
</FIXML>
```

The combination of MsgTypeCode, SchemaVer, TransType, TransSubType, TransProdSet and FinalizationCycle uniquely identifies the transmission associated with the EOD message. The NoMessageSent field represents the number of messages sent as part of the transmission associated with the EOD message.

The DDS EOD message will have a proprietary content and format and will not be proposed as an addition to the FIX Protocol Standard due to its unique functionality for our purposes. The detailed content for these messages is provided as part of each transmission's section.

Data Delivery Processing

Batch Flow

Within the ENCORE processing environment, OCC will offer both a data publishing and a data delivery service. For those recipients that choose **not** to have OCC push data out to the recipient's system, the Pull Delivery Model illustrated in Figure 1 will be employed. In this scenario, OCC will publish the recipient's data to a secure data storage device and it will be incumbent upon the client system to initiate the data transfer. Assuming the data files being requested have been published, OCC will service the request; if the publishing of the file has not yet occurred, a reply notifying the requestor that the file is not available will be sent. The hardware and software requirements for the client system in this scenario are discussed in the subsequent section.

Interactions / Confirmations

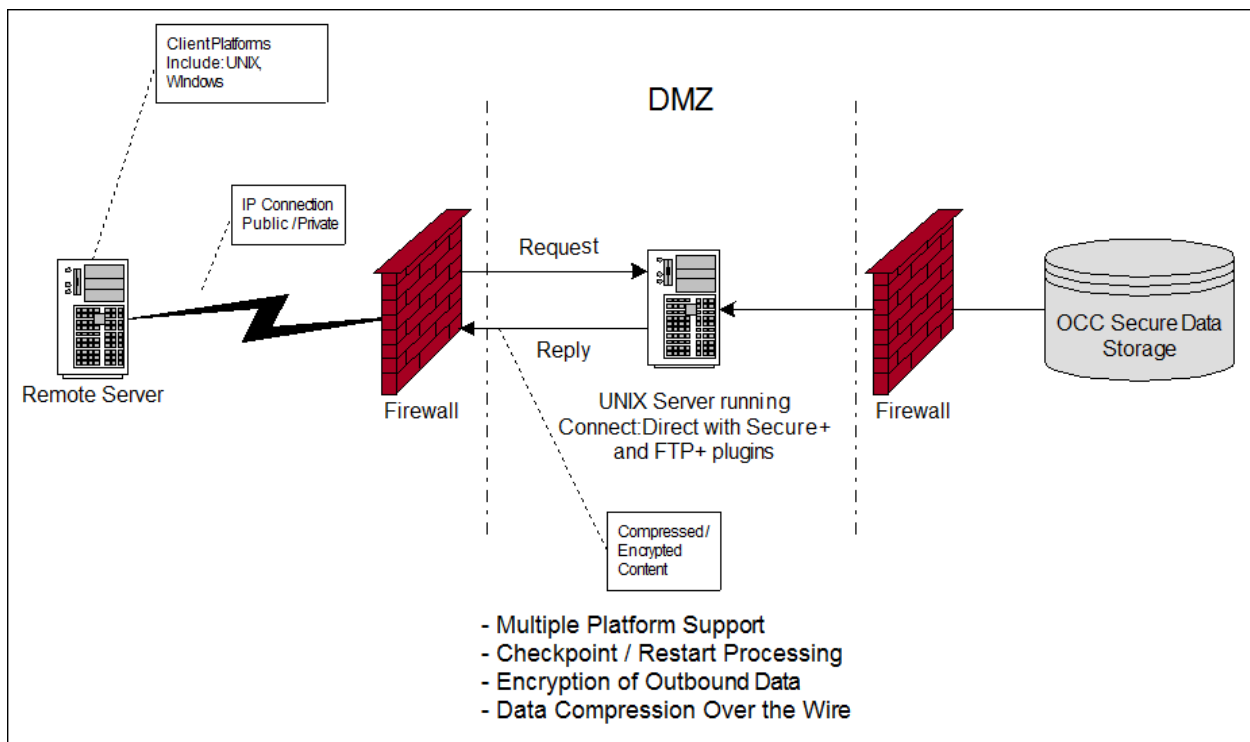


Figure 1 – ENCORE Pull Delivery Model

For those recipients that choose to have OCC push data out to the recipients' system, the Push Delivery Model illustrated in Figure 2 will be employed. In this scenario, OCC will have access to a remote server in the recipient's data center and will initiate the transfer of that recipient's data files as they become available. The hardware and software requirements for the client system in this scenario are discussed in the subsequent section.

Interactions / Confirmations

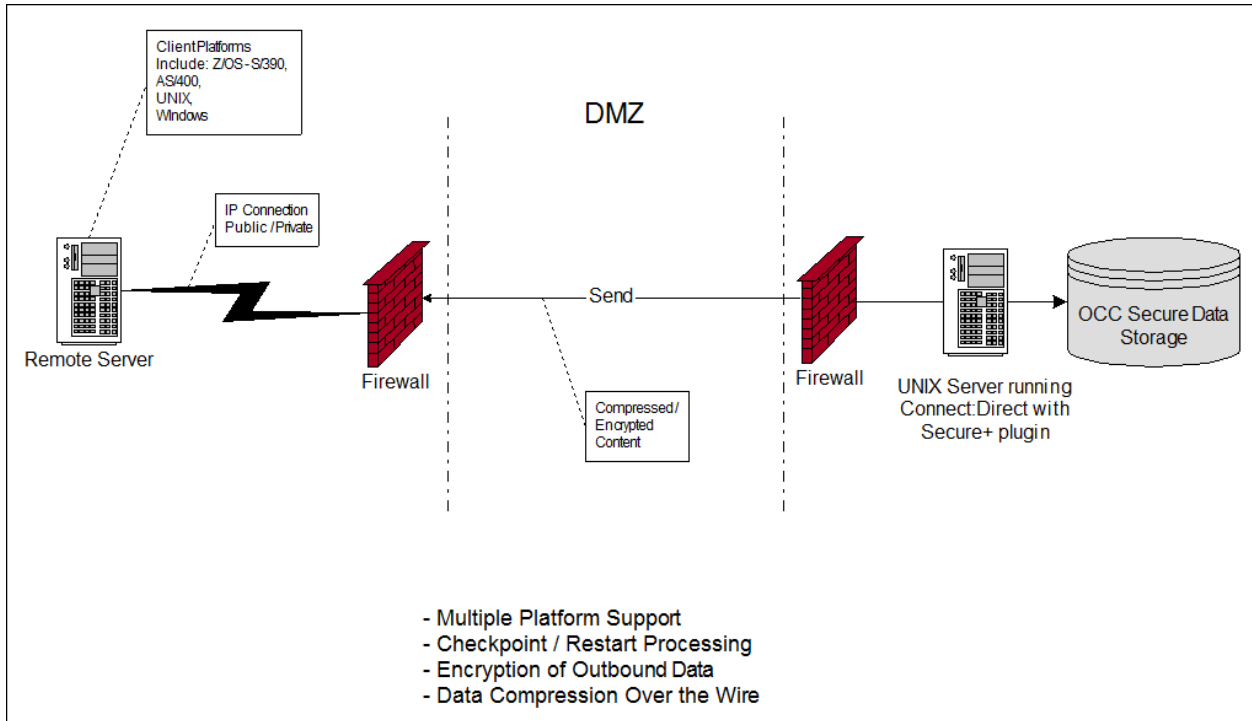


Figure 2 – ENCORE Push Delivery Model

Real-Time Flow

One of the new features of ENCORE DDS is the availability of transaction-based records — trades, post trades, etc. — in a real-time environment. For example, as a real-time trade is validated, the DDS system will be notified and will generate an output message to be sent to real-time DDS subscribers that will state the terms of the trade and the status of the trade — Accepted, Rejected, or Busted.

To enable the delivery of these real-time messages, OCC uses infrastructure which is proven and offers native data security, guaranteed delivery, and high-throughput capacity. The hardware and software requirements for the client system in this scenario are discussed in the subsequent section.

Interactions / Confirmations

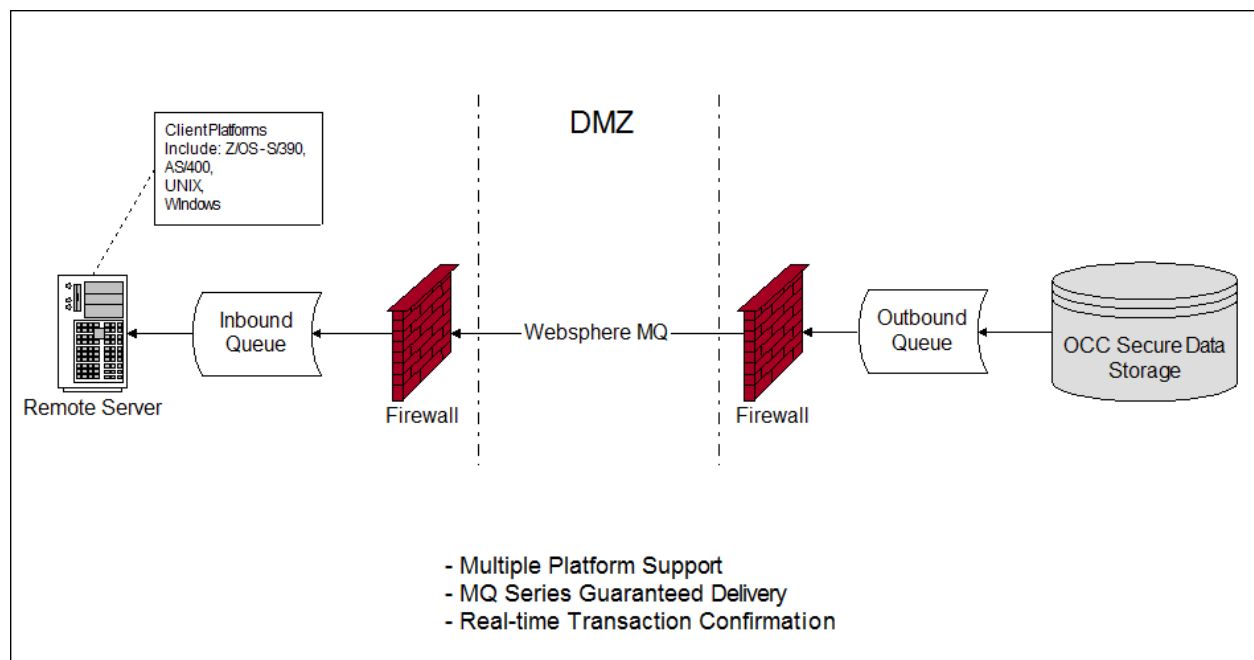


Figure 3 – ENCORE Real-Time Data Delivery Model

System Components / Requirements

Hardware

There should be no additional hardware requirements resulting directly from implementing the DDS system. For those organizations that do not currently have an IP channel established with OCC, additional network hardware may be required — in those cases, OCC will coordinate the channel setup.

OCC will continue to support several different IP connection options for data delivery. For private line communications, OCC will support T1 broadband connections as well as ISDN dial-up connections. In addition to private line access, OCC will now be supporting public line transmissions as a separate but equal alternative. With the ENCORE implementation of DDS, OCC has the ability to both encrypt and compress data, making the Internet option more viable and secure for its membership.

Software for Push File Delivery

For those organizations that currently communicate with OCC via Connect:Direct (NDM) or that are, for other reasons, licensed Connect:Direct users, the change required to migrate to the new delivery structure is minimal. The change consists of node configuration changes and confirmation of an IP channel to OCC.

Organizations that do not have a Connect:Direct license and intend to receive data from OCC through a Push mechanisms will need to acquire the software for the platform of their choice.

As part of the DDS implementation, OCC will also be able to secure its NDM transmissions by leveraging the Secure + plug-in for Connect:Direct. As such, organizations that wish to secure their NDM transmissions from OCC can choose this option, as long as Secure + is part of their Connect:Direct installation and the proper configurations are setup.

Compression over the wire (a feature provided as part of the Connect:Direct products) will be more widely used to accommodate the higher volume of data.

So far as platform support is concerned, the following installations are available and have been validated:

- NT/2000/2003: Connect:Direct Java for Windows
- UNIX (most versions): Connect:Direct for UNIX
- Z/OS Connect:Direct for OS/390

Software for Pull File Delivery

OCC plans to standardize the software used by the organizations to retrieve data service files from its servers. But at the same time, the OCC will also make sure to offer flexibility with regard to the supported platforms and the types of communication. The drivers for this change originate from the need to better accommodate the data delivery requirements for DDS with a particular focus on data volume and encryption.

Compression over the wire can be used as a viable alternative to decrease the bandwidth usage. Additionally, the ability to provide data encryption becomes mandatory in order to support secure transmissions over various types of communication lines, including the public internet.

The product selected by OCC that meets these requirements is SFTP. SFTP is the SSH File Transfer Protocol. SFTP allows for secure, encrypted, compressable file transfers over any reliable network.

OCC's SFTP server supports any SSH v3 compliant SFTP client on any UNIX or Windows platform. Most modern UNIX and Linux operating systems come pre-installed with the OpenSSH packages which include the SFTP client. Some common Windows clients include WinSCP, WS_FTP, Putty, FileZilla, and CuteFTP. Customers are responsible for acquiring the SFTP client of their choice. An SFTP client setup guide for common SFTP clients will be provided for convenience.

In order to authenticate to OCC's SFTP server, each client firm will need to provide an ssh public key that is paired to an ssh private key to be used for the connection. A user ID and password will also be issued during the setup process.

Software for Real-Time Messaging

The real-time messaging solution used by OCC is based on the WebsphereMQ product suite. The use of this solution is a natural extension of the existing real-time architecture implemented by OCC as part of ENCORE.

For those organizations that already possess MQ or for those organizations that already communicate with OCC via MQ, the only work required to prepare for real-time messages, from an infrastructure perspective, is the definition and configuration of a new channel (that is, no new software is required).

For those organizations that do not currently possess an MQ license, WebsphereMQ will need to be purchased and a new MQ IP channel with OCC will need to be configured. Should an organization choose not to purchase MQ, all requested data will still be available in the form of file-based transmissions at the end of the processing day, as all transmissions that are available in real-time are also available in batch mode.

FIXML Schema Concepts

FIXML Schema is the data standard for OCC's Data Distribution Services.

Usage

There are many files included in the OCC FIXML package. For all parsing and validation, start with the file **fixml-occ-4-4.xsd**. All other files that are used are included from this base file.

To read DDS FIXML messages, OCC recommends using an XML parser that adheres to the W3C* 1.0 and 1.1 XML recommendations and not the byte by byte method typically used for "flat file" parsing. In order to support new future business needs, OCC reserves the right to add at any time previously unused tags, which are already part of the FIXML schema, to the DDS FIXML messages. If the parsing mechanism recommended above will be used, the addition of new tags will have no impact on the programs that read in the DDS FIXML messages.

Following the XML standard, DDS messages will not include elements or attributes that do not contain a value. This would include both NULL values and empty string values.

The FIXML Schema imposes an order to the message but this order only applies to the component blocks included in the message. This means that within a given component block the tags (attributes) can be present in any order. In addition, there is no sort order imposed on the data content of the message. For example, this means that the SecurityList messages (the DDS equivalent of the Series file) will not be sorted per symbol or per any other tag. On a more general level it should also be noted that if a DDS Recipient receives a batch file containing more than one message type (e.g. Positions + Trades), this batch file will not be sorted per message type. Position messages and trade messages may be commingled throughout the file depending on how the particular file was built.

* - The World Wide Web Consortium (W3C) is an international consortium where Member organizations, a full-time staff, and the public work together to develop Web standards.

FIXML and FIXML Extensions Version Identification

FIXML versions are identified explicitly in the schema file names and also with constant attribute values defined in the fixml-component-base schema file.

FIXML Schema File Versioning

FIXML Schema employed the file naming convention developed for FpML. The major and minor version numbers of the FIX version represented by the schema are appended to all FIXML schema file names. This approach was taken to explicitly force users to recognize when counterparties have changed their version of the schema.

FIXML Message Versioning

The FIXML root element <FIXML> contains five attributes that define the version of the message. The FIXML root element is defined in the **fixml-components-base** schema file.

Attribute Description Format Example

Attribute	Description	Format	Example
v	FIX Version	N.N	4.4
r	FIX Version release date (used to designate errata releases between FIX versions)	YYYYMMDD	20030618
s	Schema Release (used to designate schema releases between errata releases)	YYYYMMDD	20040109
xv	FIXML Extensions Version	N.N	1
xr	FIXML Extensions Originator	String	FIA

```
<FIXML r="20030618" s="20040109" v="4.4" xr="FIA" xv="1"
xmlns="http://www.fixprotocol.org/FIXML-4-4">
  ... Message ...
</FIXML>
```

For On-Demand Positions, the following FIXML root element will be used.

```
<FIXML r="20030618" s="20040109" v="4.4" xr="FIA" xv="1.1"
xmlns="http://www.fixprotocol.org/FIXML-4-4"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.fixprotocol.org/FIXML-4-4
https://www.theocc.com/getmedia/11f77dc2-98cf-4d31-898a-a95eb91abc9b/fixml-
main-4-4-FIA-1-1.xsd">
```

Batch Files vs. Real-Time MQ Transmission

Real-time transmissions of FIXML messages via MQ will treat each message as a separate event and therefore each message transmitted via MQ will include the above referenced FIXML tag.

Batch files follow a different implementation. A batch file will include an additional tag to communicate to the XML parser that multiple messages are contained in the file as follows:

```
<FIXML r="20030618" s="20040109" v="4.4" xr="FIA" xv="1"
xmlns="http://www.fixprotocol.org/FIXML-4-4">
  <Batch>
    ... Message ...
    ... Message ...
    ... Message ...
  </Batch>
</FIXML>
```

FIX Concepts

The FIX protocol has developed several concepts that are repeated in many of the message types. This section will provide specific detail on these concepts.

CFI Code

As of FIX 4.3, the CFI Code field was added to the FIX Protocol in an attempt to provide a standards-based source of security type values by using values defined in the ISO 10962 standard: Classification of Financial Instruments (CFI code). This code will appear in every transmission that contains the Instrument Block (product/series/contract information).

A subset of ISO 10962 values applicable to FIX usage is identified below. The official standard and set of possible values are maintained by the ISO 10962 standard and any discrepancies below should be considered typographical errors using the ISO 10962 standard as the correct set of values. To obtain the ISO 10962 standard, please contact the ISO 10962 secretariat and/or visit the ISO web site at <http://www.iso.ch>.

The ISO 10962 standard defines a six (6) character code in which each character's position value carries a special significance (attribute) and set of values. Note that "X" represents an unspecified or unknown attribute, thus it is not always necessary to specify every attribute (character position value).

Note: The corresponding FIX 4.2 SecurityType field value is identified within brackets [] in the list below.

The following is a high-level subset of possible values applicable to FIX usage:

EXXXXX = Equity Shares (various)
DXXXXX = Debt (various)
FXXXXX = Future [FUT]
MRCXXX = Misc., Referential Instrument, Currency [FOR]
MRIXXX = Misc., Referential Instrument, Index [n/a]
OCXXXX = Option - Call [OPT]
OPXXXX = Option - Put [OPT]

Note: "X" represents an unspecified or unknown attribute and many of the values above containing "X" can be further defined according to the CFI standard (e.g., Voting rights are the third character of Equity Common Shares).

Detailed subset of possible values applicable to FIX usage:

Definition for Options (code defined by character position)

Char 1 Category	Char 2 Group	Char 3 Scheme	Char 4 Underlying Asset	Char 5 Delivery	Char 6 Stdized/Non- Std
O =Options	C =Call P =Put M =Other X =Unknown (n/a)	A =American E =European X =Unknown (n/a)	B =Basket S =Stock- Equities D =Interest rate/notional debt sec T =Commodities C =Currencies I =Indices O =Options F =Futures W =Swaps M =Other X =Unknown (n/a)	P =Physical C =Cash X =Unknown (n/a)	S =Standardized terms (maturity date, strike price, contract size) N =Non- standardized terms X =Unknown (n/a)

Examples

OCXXXS	Standardized Call Option
OPXXXS	Standardized Put Option
OCXFXS	Standardized Call Option on a Future
OPXFXS	Standardized Put Option on a Future
OCEFCN	Nonstandard call option on future with European style expiration and cash delivery
OPASPN	Nonstandard put option on stock with American style expiration and physical delivery
OCEICN	Nonstandard call option on an index with European style expiration and cash delivery

Non-Standard designation will use OCC's definition of this term: Non-Standard terms of settlement or multiple deliverables.

Definition for Futures (code defined by character position)

Char 1 Category	Char 2 Group	Char 3 Underlying Asset	Char 4 Delivery	Char 5 Stdized/Non- Std	Char 6 N/A Undefined
F =Futures	F =Financial Futures C =Commodity Futures M =Others X =Unknown (n/a)	A =Agriculture, forestry, and fishing B =Basket S =Stock-Equities (for financial future) or Services (for commodities futures) D =Interest rate/notional debt sec C =Currencies I =Indices (for financial futures) or Industrial Products (for commodities futures) O =Options F =Futures W =Swaps M =Other X =Unknown (n/a)	P =Physical C =Cash X =Unknown (n/a)	S =Standardized terms (maturity date, strike price, contract size) N =Non-standardized terms X =Unknown (n/a)	X =Not applicable / undefined

Examples

FXXXS	Standardized Future
FFICN	Nonstandard Financial Future on an index with cash delivery
FFSPSX	Standard Future on an equity with physical delivery
FXXXN	Nonstandard future – contract type specified in symbology – not provided in CFI Code

Non-Standard designation will use OCC’s definition of this term: Non-Standard terms of settlement or multiple deliverables.

Market Identifier Code (MIC)

As of FIX 4.3, Exchange Codes used in FIX are those defined in the ISO 10383 standard: Market Identifier Code (MIC). A MIC will be used whenever exchange information is included in a message. The official standard and set of values are maintained by the ISO 10383 standard and any discrepancies below should be considered typographical errors. Always refer to the ISO 10383 standard for the correct set of values. These values are maintained by ISO 10383 secretariats and as of the time of this publication the web site link to view current list of MIC values is: <http://www.iso15022.org/MIC/homepageMIC.htm>

Note: Refer to the current ISO 10383 standard for the complete list. The following list is a subset of the complete list and is designed primarily to support exchanges that interact with OCC.

Exchange Acronym	MIC
AMEX	XASE
ARCA	XPSE
BATS	BATO
BOX	XBOX
C2	C2OX
CBOE	XCBO
CFE	XCBF
CME	XCME
EDGX	EDGO
EMLD	EMLD
GEM	GMNI
ISE	XISX
MCRY	MCRY
MIAX	XMIO
MPRL	MPRL
NOBO	XBXO
NSDQ	XNDQ
ONE	XOCH
PHLX	XPHO
SML	SMFE

Party Component Block

The FIX Party Component Block will be used in all applicable FIX Reports to represent OCC Account Information. Below is a sample of this block and the corresponding translations:

<Pty ID="OCC" R="21"/>	➔ Clearing Group
<Pty ID="00352" R="4">	➔ Clearing Member Number
<Sub ID="M" R="26"/>	➔ Account Type
</Pty>	
<Pty ID="XYZ" R="38"/>	➔ Account ID

Occasionally, additional information will be listed in the Party Component Block when applicable. For example, in the Trade Capture Report for a CMTA trade, the Give-Up information will be listed in the block. In this case, the block will look like the following example:

<Pty ID="OCC" R="21"/>	➔ Clearing Group
<Pty ID="00551" R="4"/>	➔ Clearing Member Number
<Sub ID="M" R="26">	➔ Account Type
</Pty>	
<Pty ID="XYZ" R="38"/>	➔ Account ID
<Pty ID="00792" R="14"/>	➔ Give Up Clearing Firm

Instrument Component Block

The Instrument Component Block will be used in all applicable messages to describe OCC cleared products. Below are sample(s) of this block and the corresponding translations.

OPTION EXAMPLE

```
<Instrmt
  Sym="AOL"           → Product Symbol
  CFI="OCASPS"       → See CFI Code description on page 18
  MMY="20031122"     → Series/Contract Date
  MatDt="2003-11-22" → Expiration Date
  StrkPx="47.5"      → Strike Price
  StrkCcy="USD"      → Strike Currency
  StrkMult="1"       → Strike Multiplier
  StrkValu="1"       → Strike Value
  Mult="100"         → Contract Multiplier
/>
```

FUTURES EXAMPLE

```
<Instrmt
  Sym="AOL1C"        → Product Symbol
  ID=" AOL1C"        → Product Symbol
  Src="8"            → FIX enumeration for Exchange Symbol
  CFI="FFSPSX"       → See CFI Code description on page 19
  MMY="20031122"     → Series/Contract Date
  MatDt="2003-11-22" → Expiration Date
  Mult="100"         → Contract Multiplier
/>
```

In some messages, such as the Security Definition Report and Security Update Report, additional fields will be included in the Instrument Block to further describe the option or futures product.

The StrkMult and Mult fields have been provided in the Instrument Component Block because they are often used by OCC to calculate settlement values and in-the-money values.

FIXML Data Types (as used by OCC)

FIX Data Type	FIX Definition	OCC Definition	Example
Integer	Sequence of digits without commas or decimals and optional sign character. Integer values may contain leading zeros.	Leading zeros will be removed.	723
Float	Sequence of digits with optional decimal point and sign character. Float values may contain leading and trailing zeros.	Leading and trailing zeros will be removed. The number of decimal points will be limited to six.	245.3967
Qty	Float field capable of storing either a whole number of “shares” or a decimal value containing decimal places for non-share quantity asset classes	Whole numbers only	25
Price	Float field representing a price. The number of decimal places can vary and prices may be negative values.	The number of decimal places will be limited to six.	3.12
Amt	Float field typically representing a Price times a Qty.	The number of decimal places will be limited to six.	392785.23
Percentage	Float field typically representing a percentage. The number of decimal places can vary.	The number of decimal places will be limited to six.	0.95
Char	Single character value that can include any alphanumeric character or punctuation except the delimiter. All character fields are case sensitive.		Y
String	Alpha-numeric free format strings that can include any character or punctuation except the space delimiter. All character fields are case sensitive.		GUI

MultipleValue String	Alpha-numeric free format strings that can include any character or punctuation. Can contain one or more space-delimited values. All character fields are case sensitive.		A 1 A 2
Currency	String field representing a currency type using ISO 4217 Currency code (3 character) values.	Three character Currency code will be used	USD
Exchange	String field representing a market or exchange using ISO 10383 Market Identifier Code (MIC).	Four character MIC will be used	XASE
Month-Year	String field representing a month of a year. An optional day of the month can be appended or an optional week code. Valid formats: YYYYMM, YYYYMMDD, YYYYMMWW	Only valid format is YYYYMMDD	20031122
UTCTimestamp	Time/date combination represented by local time and its offset from UTC (also known as GMT). The format is YYYY-MM-DDThh:mm:ss-hh:mm	Only valid format is YYYY-MM-DDThh:mm:ss-hh:mm	2003-11-25T03:45:23-05:00
UTCTimeOnly	Time-only represented in UTC (also known as GMT) is HH:MM:SS-hh:mm	Only valid format is HH:MM:SS-hh:mm	03:45:23-05:00
UTCDateOnly	Date represented in UTC (also known as GMT) in YYYY-MM-DD format		2003-11-25
LocalMktDate	Date of local market (vs. UTC) in YYYY-MM-DD format. This is the "normal" date field used by the FIX protocol.		2003-11-25

UTC Timestamp

All FIX Reports that include transaction times, creation times, update times, etc., will be reported in a time/date combination that includes local time and the offset from UTC (Coordinated Universal Time, also known as GMT). The UTC Timestamp is represented by the format, YYYY-MM-DDThh:mm:ss-hh:mm. To indicate the time zone (i.e., the difference between the local time and UTC) the time is immediately followed by a plus or minus sign (+ or -) followed by the difference from UTC represented as hh:mm. Adjustments are made for Daylight Savings Time.

For example:

To indicate 3:45:23 am on November 25, 2003 for Eastern Standard Time which is five (5) hours behind Coordinated Universal Time (UTC), one would write:

2003-11-25T03:45:23-05:00.

Implementation Detail on Empty Values

Any attributes which are omitted from a FIXML message should be considered empty or as having no value. The example below shows this behavior in FIXML for an Option instrument:

```
<Instrmt
  Sym="AOL"           → Product Symbol
  CFI="OCASPS"       → See CFI Code description on page 18
  MMY="20031122"    → Series/Contract Date
  MatDt="2003-11-22" → Expiration Date
  StrkPx="47.5"      → Strike Price (Decimal format)
  StrkCcy="USD"      → Strike Currency
  StrkMult="1.0"     → Strike Multiplier
  Mult="100"         → Contract Multiplier
/>
```

Futures instruments have no Strike attributes. Therefore, whenever possible the attributes are omitted and should be considered to have no value.

```
<Instrmt
  Sym="IBM1C"        → Product Symbol
  CFI="FFSPSX"      → See CFI Code description on page 19
  MMY="20031122"    → Series/Contract Date
  MatDt="2003-11-22" → Expiration Date
  Mult="100"         → Contract Multiplier
/>
```

FIX Reference Materials

Information on the current FIX 4.4 specification can be found at:

http://www.fixprotocol.org/specification/fix-44_w_Errata_20030618.zip (ZIP of MS Word)

http://www.fixprotocol.org/specification/fix-44_w_Errata_20030618_PDF.zip (PDF of MS Word)

In each message layout for a given transmission, FIX tag numbers have been included for ease of reference to FIX documentation. This data will not be included or referenced in the FIXML schema provided by OCC.

The OCC's DDS Reference guides are available at

If you have additional questions or comments, please contact your Member Services representative or The Options Clearing Corporation Help Desk at one of the following telephone numbers:

1.800.621.6072 (U.S.)

1.800.424.7320 (Canada)

Or, you can get DDS help by e-mail at the following address: ddshelp@theocc.com

Appendix

Revision History

Version	Date	Author	Version Updates
1.10	6/30/2014	OCC	Removed references to NYL. Added Revision History. Updated Cover Page logo.
1.11	9/2/2015	OCC	Added EDGX to exchange list.
1.12	12/15/2015	OCC	Added MCRY to exchange list.
1.13	12/6/2016	OCC	Added MPRL to exchange list.
1.14	1/26/2017	OCC	Changed http://optionsclearing.com URL to https://www.theocc.com
1.15	8/24/2018	OCC	<ul style="list-style-type: none">• Applied rebranding updates.• Removed ELX exchange.
1.16	11/12/2018	OCC	Added EMLD to exchange list.
1.17	8/30/2019	OCC	Added SML to exchange list
1.18	6/28/2020	OCC	<ul style="list-style-type: none">• Revised the www.TheOCC.com web site URLs for FIXML schema to reflect new web site organization.• Removed NFX from exchange list.