

FIX TRADING COMMUNITY

France Trading Conference 2024

– Rules of Engagement with Orchestra –

Thursday 21st November 2024

Hanno Klein

FIX Technical Director

GTC EMEA Co-Chair

Senior Standards Advisor, FIXdom



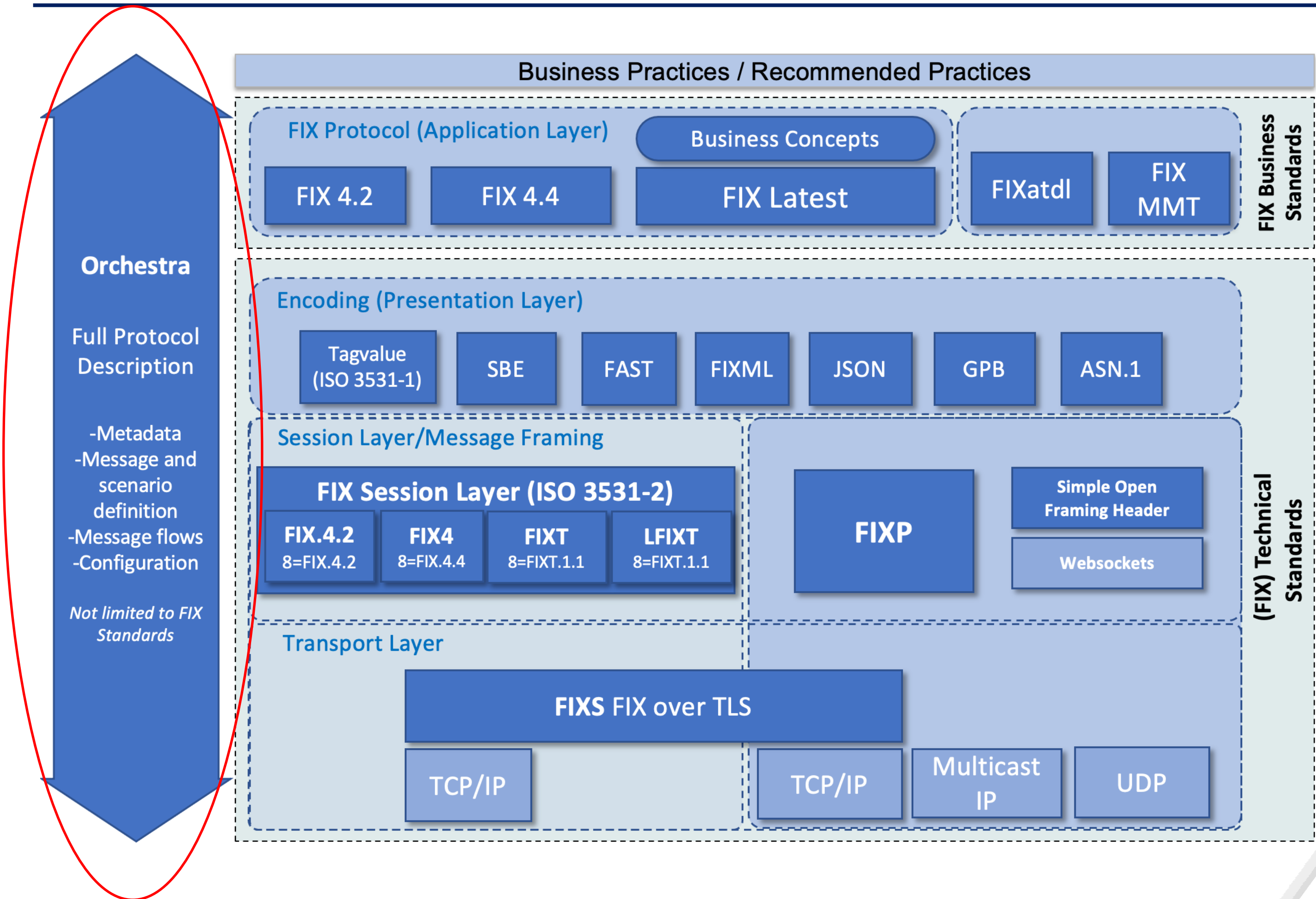
Agenda

- Orchestra Technical Standard
 - Objectives
 - Orchestra in a nutshell
- Orchestra Tools
 - Free tools
 - Open-source tools
- Rules of Engagement with Orchestra Tools
 - Tablature for message design
 - Pandoc for document generation
 - Orchimate for visualization of Orchestra XML files

Orchestra Technical Standard



FIX Standards



Objectives for Orchestra

- Machine-readable standard for meta-data describing the content and behavior of an electronic messaging interface.
- Protocol agnostic to be applicable to FIX and non-FIX interfaces.
 - FIX Protocol (across all versions and flavors, including user-defined elements)
 - Regulatory protocols (e.g. US: SEC-CAT, Europe: ESMA/FCA, Asia: SFC-DS-OL)
 - Industry standard protocols (e.g. ISO 20022, FpML)
 - Proprietary protocols (trading venues, clearinghouses, buy/sell-side, vendors)
- Encoding agnostic to separate the business semantics from the wire format (standard/proprietary, ASCII/binary, with/without meta-data).
- Metadata for technical connectivity (counterparties, connections, sessions, versions, encodings, security,...)

Orchestra in a nutshell (application level)

- Basic features
 - Messages, groups, components, fields, code sets, codes, generic datatypes.
 - Nesting of groups/components inside messages, groups, components.
 - Simple presence rules (e.g. required, optional, ignored) for elements.
 - Unique identification and versioning (a.k.a. pedigree) of all elements.
- Advanced features
 - Conditional rules defined with expressions (Score DSL).
 - Scenarios for most elements to distinguish use cases.
 - Workflows to support request/response models or complex negotiations.
 - Actors and state machines to define transitions.

Orchestra in a nutshell (connection level)

- Basic features

- An interface is a collection of services, protocols, sessions, and the transport exposed by a counterparty.
- A service is an offering of an application (e.g. order entry) and requires the identification of an Orchestra XML file describing the messages etc.
- A protocol relates to a specific layer of the technical stack of the interface, e.g. to the encoding or to the session protocol supported by the interface.
- A session describes the connection with a counterparty (e.g. IP addresses).
- A transport describes the lowest layer of the interface (e.g. TCP, UDP multicast).

- Advanced features

- FIXatdl can be supported as the protocol used for the user interface and requires the identification of an FIXatdl XML file describing the GUI.
- A session may have an effective time (start/end time) to support configuration prior to use.
- A session definition may contain security keys (e.g. certificates, private keys) to be used when exchanging messages.

Orchestra Tools



Free Orchestra Tools

- FIXimate
 - Interactive, browser-based reference for the FIX Protocol Specification, generated from the Orchestra XML file of FIX Latest.
- Playlist*
 - Generates an Orchestra XML file as subset of a reference file in Orchestra XML format (e.g. FIX Latest) loaded into your local browser.
- Log2Orchestra*
 - Generates an Orchestra XML file by using FIX engine logfiles with tag=value encoding and an Orchestra XML reference file as input.
- Orchestra Server* (Esprow)
 - Interactive, browser-based tool to maintain a repository and export to an Orchestra XML file.
- Orchimate (Atomic Wire)
 - Interactive, browser-based tool to upload and display an Orchestra XML file in your local browser.

*requires FIX membership

FIX.Latest_EP292 - English

Find all: [Search](#)
 Regex match: ^=start, \$=end, .=any char
 Match abbreviated name only

Message type: [Lookup](#)

Component: [Lookup](#)

Field tag: [Lookup](#)

Field name: [Lookup](#)

Code set: [Lookup](#)

- About this orchestration
- Message Summary
- Components
- Fields
 - Sorted by Tag Number
 - Sorted by Field Name
 - Sorted by Datatype
 - User Defined Fields
- Code Sets
- Datatypes
- Message Layouts**
- Session
- PreTrade
- Trade
 - SingleGeneralOrderHandling
 - DontKnowTrade
 - ExecutionAck
 - ExecutionReport
 - NewOrderSingle
 - OrderCancelReject
 - OrderCancelReplaceRequest
 - OrderCancelRequest
 - OrderStatusRequest
 - ProgramTrading
 - OrderMassHandling
 - CrossOrders
 - MultilegOrders
- PostTrade
- Infrastructure

NewOrderSingle [type 'D']

<Order>

The new order message type is used by institutions wishing to electronically submit securities and forex orders to a broker for execution.

The New Order message type may also be used by institutions or retail intermediaries wishing to electronically submit Collective Investment Vehicle (CIV) orders to a broker or fund manager for execution.

Pedigree Added FIX.2.7

[Expand Components](#) | [Collapse Components](#)

Field or Component	Field Name	Abbr Name	Req'd	Comments	Pedigree
Component	StandardHeader	Hdr	Y	MsgType = D	Added FIX.2.7
11	ClOrdID	ID	Y	Unique identifier of the order as assigned by institution or by the intermediary (CIV term, not a hub/service bureau) with closest association with the investor.	Added FIX.2.7
2422	OrderRequestID	OrdReqID			Added EP188
526	SecondaryClOrdID	ID2			Added FIX.4.3
583	ClOrdLinkID	LnkID			Added FIX.4.3
2829	DuplicateClOrdIDIndicator	DupClOrdIDInd			Added EP253
Component	Parties	Pty		This is party information related to the submitter of the request.	Added FIX.4.3 Updated EP131

Tag	Field Name	Abbr Name	Data Type	Union Datatype	Description	Pedigree
11	ClOrdID	ClOrdID / ID in SingleGeneralOrderHandling	String		Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID(49) or OnBehalfOfCompID(115) as appropriate). Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the ClOrdID(11) field.	Added FIX.2.7 Updated EP282

Used in messages:

[CollateralAssignment][CollateralInquiry][CollateralInquiryAck][CollateralReport][CollateralRequest][CollateralResponse][Email][ExecutionAck][ExecutionReport][MultilegOrderCancelReplace][NewOrderMultileg][NewOrderSingle][OrderCancelReject][OrderCancelReplaceRequest][OrderCancelRequest][OrderMassActionReport][OrderMassActionRequest][OrderMassCancelReport][OrderMassCancelRequest][OrderStatusRequest][QuoteRequest][QuoteResponse][RegistrationInstructions][RegistrationInstructionsResponse][SettlementInstructions][TradeCaptureReportRequest]

Used in components:

[InstrmtStrkPxGrp][ListOrdGrp][OrdAllocGrp][OrdListStatGrp][OrderAggregationGrp][OrderEntryAckGrp][OrderEntryGrp][SideCrossOrdCxlGrp][SideCrossOrdModGrp][TradeReportOrderDetail]

Orchestra Tools – Playlist v1.2

<https://playlist.fixtrading.org>

FIX Playlist

Creates a subset of an Orchestra file



Input

Source Orchestra file

100%

Drag file to read or

Choose File

FIX Standard

OrchestraFIXLatest.xml

Use Selector File

Clear Input File

Select Your Content

(All elements sorted alphabetically. Greyed out items are deprecated)

+ | MESSAGES

+ | GROUPS

+ | COMPONENTS

- | FIELDS

+ | Tags 1-999

+ | Tags 1000-1999

+ | Tags 2000-2999

+ | Tags 40000-49999

+ | CODESETS

+ | DATATYPES

- | NewOrderSingle(35=D)

Account(1) - Type String

AccountType(581) - Type AccountTypeCodeSet

AcctIDSource(660) - Type AcctIDSourceCodeSet

AffiliatedFirmsTradeIndicator(2525) - Type Boolean

AllocID(70) - Type String

AuctionAllocationPct(1804) - Type Percentage

AuctionInstruction(1805) - Type AuctionInstructionCodeSet

AuctionType(1803) - Type AuctionTypeCodeSet

BookingType(775) - Type BookingTypeCodeSet

BookingUnit(590) - Type BookingUnitCodeSet

CancellationRights(480) - Type CancellationRightsCodeSet

CashMargin(544) - Type CashMarginCodeSet

ClOrdID(11) - Type String

ClOrdLinkID(583) - Type String

ClearingAccountType(1816) - Type ClearingAccountTypeCodeSet

ClearingFeeIndicator(635) - Type ClearingFeeIndicatorCodeSet

CommissionData - Component

CommissionDataGrp - Group

ComplianceID(376) - Type String

ComplianceText(2404) - Type String

ContraOrderOrigination(2882) - Type OrderOriginationCodeSet

ContraRoutingArrangementIndicator(2884) - Type RoutingArrangementIndicatorCodeSet

CopyMsgIndicator(797) - Type Boolean

CoveredOrUncovered(203) - Type CoveredOrUncoveredCodeSet

Currency(15) - Type Currency

Select FIX Standard File

FIX44Session.xml

FIXTSession.xml

OrchestraFIX42.xml

OrchestraFIX44.xml

OrchestraFIXLatest.xml

CANCEL

Output

Orchestra file to create (*.xml)

myorchestra.xml

Create Orchestra file

User Guide

Terms of Service

Version 1.2

© Copyright 2024, FIX Protocol Ltd.

Orchestra Tools – Log2Orchestra

<https://log2orchestra.fixtrading.org>

FIX Log to Orchestra

Creates an Orchestra file from one or more FIX message logs (tag-value encoding)

Input

Reference Orchestra file	FIX message log files	Configuration file for scenarios (optional)
<p>Drag file to read or</p> <p>Choose File</p> <p>FIX Standard</p>	<p>Drag file to read or</p> <p>Choose Files</p>	<p>Drag file to read or</p> <p>Choose File</p>
<p>Clear Input Files</p>		

Output

Orchestra file to create (*.xml)

Append only (removes no scenarios)

[Create Orchestra file](#)

[Terms of Service](#)

[Help](#)

Orchestra Tools – Orchestra Server

<https://orchestra.sbox.espro.com>

< > **NewOrderSingle [D] (base)**

[View Details](#) | [Edit Message](#) | [Duplicate Message](#) | [Delete Message](#)

[+ Add Element](#)
[+ Bulk Add Elements](#)
[Edit Element](#)
[Delete Element\(s\)](#)

[Components](#)
[Groups](#)
[Fields](#)
[Code Sets](#)

<input type="checkbox"/>	Name	Id	Scenario	Presence	Documentation
<input type="checkbox"/>	+ StandardHeader	1024	base	required	MsgType = D
<input type="checkbox"/>	CIOrdID	11	base	required	Unique identifier of the order as assigned by institution or by the intermediary (...)
<input type="checkbox"/>	ExDestination	100	base	optional	
<input type="checkbox"/>	- Instrument	1003	base	required	Insert here the set of "Instrument" (symbology) fields defined in "Common Com...
	Symbol	55	base	optional	Common, "human understood" representation of the security. SecurityID value c...
	SecurityID	48	base	optional	Takes precedence in identifying security to counterparty over SecurityAltID bloc...
	SecurityIDSource <input checked="" type="checkbox"/>	22	base	optional	Conditionally required when SecurityID(48) is specified.
	ISINNumber 4	22004			ISIN
<input type="checkbox"/>	- OrderQtyData	1011	base	required	
	OrderQty	38	base	optional	One of CashOrderQty, OrderQty, or (for CIV only) OrderPercent is required. Not...
<input type="checkbox"/>	OrdType <input checked="" type="checkbox"/>	40	base	required	
	Market 1	40001			Market
	Limit 2	40002			Limit
<input type="checkbox"/>	Price	44	base	optional	Required for limit OrdTypes. For F/X orders, should be the "all-in" rate (spot rate...
<input type="checkbox"/>	+ StandardTrailer	1025	base	required	

Orchestra Tools – Orchimate

<https://orchimate.org>

Orchimate ^{ALPHA} / FIX.Latest (FIX.Latest_EP292)
?

Orchestra Specification

FIX.Latest

ADD LOCAL SPEC

Find All 🔍

Message ▾

Component or Group ▾

Field ▾

Code Set ▾

Datatype ▾

Indexes

☰ Messages

Message

NewOrderSingle (D)

ID	14
Abbr Name	Order

The new order message type is used by institutions wishing to electronically submit securities and forex orders to a broker for execution.

The New Order message type may also be used by institutions or retail intermediaries wishing to electronically submit Collective Investment Vehicle (CIV) orders to a broker or fund manager for execution.

Field	Field Name	Type	Req'd	Comments	Pedigree
> Component	StandardHeader		Y	MsgType = D	Added FIX.2.7
11	CIOrdID	String	Y	Unique identifier of the order as assigned by institution or by the intermediary (CIV term, not a hub/service bureau) with closest association with the investor.	Added FIX.2.7
2422	OrderRequestID	int			Added EP188

Field

CIOrdID ✕

ID (tag)	11
Abbr Name	CIOrdID
Type	String

Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID(49) or OnBehalfOfCompID(115) as appropriate). Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the CIOrdID(11) field.

Used in components: [TradeReportOrderDetail](#)

Used in groups: [InstrmtStrkPxGrp](#), [ListOrdGrp](#), [OrdAllocGrp](#), [OrdListStatGrp](#), [OrderAggregationGrp](#), [OrderEntryAckGrp](#), [OrderEntryGrp](#), [SideCrossOrdCxlGrp](#), [SideCrossOrdModGrp](#)

Used in messages: [CollateralAssignment](#), [CollateralInquiry](#), [CollateralInquiryAck](#), [CollateralReport](#), [CollateralRequest](#), [CollateralResponse](#), [Email](#), [ExecutionAck](#), [ExecutionReport](#), [MultilegOrderCancelReplace](#), [NewOrderMultileg](#), [NewOrderSingle](#), [OrderCancelReject](#), [OrderCancelReplaceRequest](#)



FIX Open-Source Orchestra Tools

- **Tablature (.../tablature)**
 - Simple authoring solution for Orchestra XML files (repository and interfaces).
 - Converts markdown document to an Orchestra XML file (optionally using a reference file in Orchestra XML format, e.g. FIX Latest) and vice versa.
- **XML Diff/Merge (.../xml-diff-merge)**
 - XML Diff compares two Orchestra XML files and generates a third file that represents their differences in IETF RFC 5261 format.
 - XML Merge takes a base Orchestra XML file and a difference file in IETF RFC 5261 format and merges the two to produce an updated Orchestra XML file
- **Orchestra2QuickFIX (.../fix-orchestra-quickfix)**
 - Converts an Orchestra Repository XML file to a QuickFIX dictionary.
- **Orchestra2Avro (.../fix-orchestra-avro)**
 - Utilities for the integration of Orchestra XML files and Apache Avro.

<https://github.com/FIXTradingCommunity/...>

FIX Open-Source Tools – Orchestra Utilities

- Repository Validator
 - Validates that an XML file conforms to the Orchestra Repository schema.
- Interfaces Validator
 - Validates that an XML file conforms to the Orchestra Interfaces schema.
- Repository Compressor
 - Creates extracts from an Orchestra XML file by section or category.
- Unified2Orchestra
 - Populates an Orchestra XML file from a Unified Repository 2010 Edition file.
- Orchestra2Doc
 - Generates documentation for an Orchestra XML file that can be viewed in any web browser.

<https://github.com/FIXTradingCommunity/fix-orchestra>

Rules of Engagement with Orchestra Tools



Rules of Engagement (application level)

■ Task

- Design a FIX Latest compliant interface with an order entry message and a response confirming a) the successful addition to the order book and b) the partial or full execution of the order.
- The order needs to support ticker symbols and ISINs for the security.
- Order attributes are type (market or limit), side, price, quantity, target venue and an optional custom fields “MyUDF1” (string) and “MyUDF2” (enums) for additional information.

■ Approach

- Use spreadsheet to define your messages.
- Write Tablature markdown file with FIX Latest as reference file.
 1. Define standard elements of messages (NewOrderSingle(35=D), ExecutionReport(35=8))
 2. Define standard elements of components (Instrument, OrdQtyData)
 3. Define standard subsets for code sets (SecurityIDSource(22), OrdType(40), Side(54), ExecType(150), OrdStatus(39), MsgType(34), BeginString(8))
 4. Add your RoE text, define your custom fields and add them to messages, components
- Convert markdown to Orchestra XML with Tablature (machine-readable RoE)
- Convert markdown to PDF document with Pandoc (human-readable RoE)

Step 1: Define messages in spreadsheet

Message	Component	Field	Value(s)
NewOrderSingle(35=D)		BeginString(8)	FIXT.1.1
		ClOrdID(11)	
	Instrument	Symbol(55)	
	Instrument	SecurityID(48)	
	Instrument	SecurityIDSource(22)	4=ISIN
		OrdType(40)	1=Market 2=Limit
		Price(44)	
	OrderQtyData	OrderQty(38)	
		Side(54)	1=Buy 2=Sell
		ExDestination(100)	
		MyUDF1(20000)	

Message	Component	Field	Value(s)
ExecutionReport(35=8)		BeginString(8)	FIXT.1.1
		ClOrdID(11)	
		OrderID(37)	
	Instrument	Symbol(55)	
	Instrument	SecurityID(48)	
	Instrument	SecurityIDSource(22)	4=ISIN
		OrdType(40)	1=Market 2=Limit
		Price(44)	
	OrderQtyData	OrderQty(38)	
		Side(54)	1=Buy 2=Sell
		ExDestination(100)	
		ExecID(17)	
		ExecType(150)	0=New F=Trade
		OrdStatus(39)	0=New 1=Partially Filled 2=Filled
		LeavesQty(151)	
		CumQty(14)	
		LastQty(32)	
		LastPx(31)	
		MyUDF1(20000)	
		MyUDF2(20001)	1=MyValue1 2=MyValue2

Step 2: Use Tablature markdown

Orchestra RoE Example Version 1.0

Term	Value
title	My Rules of Engagement
publisher	FIX Trading Community
rights	Copyright © FIX Protocol Ltd.
date	2024-10-16

Messages

Message NewOrderSingle

Message to enter a new order. Note that the presence of ExDestination(100) will not allow the order to be executed anywhere else.

Tag	Name	Presence	Description
comp	StandardHeader	req	MsgType(35)=D
11	ClOrdID	req	
comp	Instrument	req	
40	OrdType	req	
44	Price		
comp	OrderQtyData	req	
54	Side	req	
100	ExDestination		
20000	MyUDF1		
comp	StandardTrailer	req	

...

Components

Component Instrument

Tag	Name	Presence	Description
55	Symbol	req	
48	SecurityID		
22	SecurityIDSource		ISIN only

Fields

Tag	Name	Type	Description
20000	MyUDF1	String	My description of MyUDF1
20001	MyUDF2	MyUDF2CodeSet	My description of MyUDF2

Codesets

Codeset MyUDF2CodeSet type int

Name	Value	Description
MyValue1	1	This describes the first value.
MyValue2	2	This describes the second value.

Codeset ExecTypeCodeSet

Name	Value
New	0
Trade	F

The complete markdown file only has 150 lines.

Step 3: Convert markdown to Orchestra XML

```
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <fixr:repository xmlns:dc="http://purl.org/dc/elements/1.1/"
3   xmlns:dcterms="http://purl.org/dc/terms/"
4   xmlns:fixr="http://fixprotocol.io/2020/orchestra/repository"
5   name="Orchestra RoE Example Version 1.0" version="1.0">
6   <fixr:metadata>
7     <dcterms:title>My Rules of Engagement</dcterms:title>
8     <dcterms:publisher>FIX Trading Community</dcterms:publisher>
9     <dcterms:rights>Copyright © FIX Protocol Ltd.</dcterms:rights>
10    <dcterms:date>2024-10-16</dcterms:date>
11  </fixr:metadata>
12 > <fixr:datatypes>
114 > <fixr:codeSets>
187 > <fixr:fields>
380 > <fixr:components>
410 <fixr:groups/>
411 > <fixr:messages>
482 </fixr:repository>
483
```

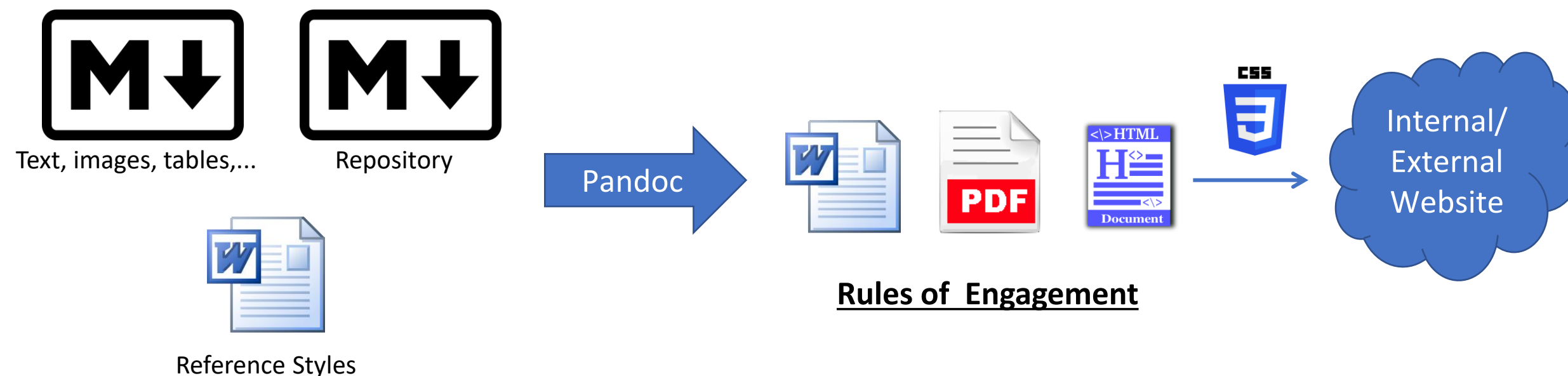
The complete Orchestra XML file for FIX Latest has 163,813 lines as of EP292!

- Standard field definitions and datatypes are automatically retrieved from the FIX Latest reference file.
- Messages, groups, components, and code sets are only subsets of FIX Latest if they are contained in the markdown file.
- Messages, groups, components, fields, and code sets are completely absent if they are not used in the markdown file.
- User-defined fields and code sets need to be fully defined in the markdown file, not in the reference file.

```
<fixr:field type="String" baseCategory="SingleGeneralOrderHandling" baseCategoryAbbrName="ID"
  id="11" name="ClOrdID" abbrName="ClOrdID" added="FIX.2.7" updated="FIX.Latest" updatedEP="282">
  <fixr:annotation>
    <fixr:documentation purpose="SYNOPSIS">
      Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.)
      (identified by SenderCompID(49) or OnBehalfOfCompID(115) as appropriate). Uniqueness must be
      guaranteed within a single trading day. Firms, particularly those which electronically submit
      multi-day orders, trade globally or throughout market close periods, should ensure uniqueness
      across days, for example by embedding a date within the ClOrdID(11) field.</fixr:documentation>
    </fixr:documentation>
  </fixr:annotation>
</fixr:field>
```

Step 4: Convert markdown to PDF document

- Create separate markdown file(s) for chapters preceding or following the chapter defining the repository, i.e. your markdown file used to create an Orchestra XML file.
- Create a Pandoc (<https://www.pandoc.org>, open source) reference style document in Microsoft Word to define layout of your Rules of Engagement document. You only need to create it once and can then use it for all of your specifications.
- Create a YAML file to define the content of your title page.
- Generate your Rules of Engagement document (docx/pdf) from your markdown file(s) with Pandoc using the reference style document.
- Optionally, generate an online version of your Rules of Engagement (html) for your website using your standard CSS file for the look-and-feel.



Pandoc Reference Style Document



Title

[Subtitle]

Author

[Date]

Abstract

[Sample title page](#)

My Company – My Application Version 1.0

October 2024

TOC Heading

1	Heading 1	3
1.1	Heading 2.....	3
1.1.1	Heading 3.....	3
1.1.1.1	Heading 4.....	3

[Sample layout for page header and table of contents](#)

1 Heading 1

Text of Heading 1

1.1 Heading 2

Text of Heading 2

1.1.1 Heading 3

Text of Heading 3

1.1.1.1 Heading 4

Text of Heading 4

Title of Table

Col 1	Col 2	Col 3	Col 4
Row 1			
Row 2			

Table caption

Image Caption

[Sample layout for chapter headings, tables, and captions](#)

```
1 ---
2 title: "My Company Name"
3 subtitle: "My Application Name"
4 author:
5 - "Rules of Engagement"
6 - "Version 1.0"
7 date: "October 16, 2024"
8 abstract: "Simulation Version"
9 ---
10
```

[Sample YAML file](#)

```
pandoc -f markdown -t docx+native_numbering "OrchestraRoEExample.md" -o "OrchestraRoEExample.docx" \
--reference-doc="OrchestraRoE_Style.docx" --metadata-file="OrchestraRoE.yaml" --toc --toc-depth=4
```



Rules of Engagement Document

My Company Name

My Application Name

Rules of Engagement

Version 1.0

October 16, 2024

Simulation Version

My Company – My Application Version 1.0 October 2024

Table of Contents

Table of Contents 2

1 Orchestra RoE Example Version 1.0..... 3

1.1 Messages 3

1.1.1 Message NewOrderSingle 3

1.1.2 Message ExecutionReport 3

1.2 Components 4

1.2.1 Component Instrument 4

1.2.2 Component OrderQtyData 4

1.2.3 Component StandardHeader 4

1.2.4 Component StandardTrailer 4

1.3 Fields 4

1.4 Codesets 5

1.4.1 Codeset BeginStringCodeSet 5

1.4.2 Codeset ExecTypeCodeSet 5

1.4.3 Codeset MyUDF2CodeSet type int 5

1.4.4 Codeset MsgTypeCodeSet 5

1.4.5 Codeset OrdStatusCodeSet 5

1.4.6 Codeset OrdTypeCodeSet 5

1.4.7 Codeset SecurityIDSourceCodeSet 5

1.4.8 Codeset SideCodeSet 6

[Table of contents](#)

1.1 Messages

1.1.1 Message NewOrderSingle

Message to enter a new order. Note that the presence of ExDestination(100) will not allow the order to be executed anywhere else.

Tag	Name	Presence	Description
comp	StandardHeader	req	MsgType(35)=D
11	ClOrdID	req	
comp	Instrument	req	
40	OrdType	req	
44	Price		
comp	OrderQtyData	req	
54	Side	req	
100	ExDestination		
20000	MyUDF1		
comp	StandardTrailer	req	

[Message definition with UDFs](#)

1.3 Fields

Tag	Name	Type	Description
20000	MyUDF1	String	My description of MyUDF1
20001	MyUDF2	MyUDF2CodeSet	My description of MyUDF2

[Custom field definitions](#)

1.4.3 Codeset MyUDF2CodeSet type int

Name	Value	Description
MyValue1	1	This describes the first value.
MyValue2	2	This describes the second value.

[Custom code set definition](#)

Rules of Engagement Visualization

Orchimate ^{ALPHA} / Orchestra RoE Example Version 1.0 (1.0)

Orchestra Specification
Orchestra RoE Example Version ...

ADD LOCAL SPEC

Find All

Message

Component or Group

Field

Code Set

Datatype

Indexes

- Messages
- Components
- Groups
- Fields
- Code Sets
- Datatypes

Orchestra RoE Example Version 1.0

1.0

Title
My Rules of Engagement

Publisher
FIX Trading Community

Rights
Copyright © FIX Protocol Ltd.

Date
2024-10-16

Orchimate Copyright 2024 Atomic Wire Technology Limited
Orchestra Copyright 2024 FIX Protocol Ltd
[Terms of Use](#) | [Privacy Policy](#)

Orchimate ^{ALPHA} / Orchestra RoE Example Version 1.0 (1.0)

Message

ExecutionReport (8)

ID 9

Abbr Name ExecRpt

Field	Field Name	Type	Req'd	Comments
> Component	StandardHeader		Y	MsgType(35)=8
11	ClOrdID	String	Y	
37	OrderID	String	Y	
> Component	Instrument		Y	
40	OrdType	OrdTypeCodeSet	Y	
44	Price	Price		
> Component	OrderQtyData		Y	
54	Side	SideCodeSet	Y	
100	ExDestination	Exchange		
17	ExecID	String	Y	
150	ExecType	ExecTypeCodeSet	Y	
39	OrdStatus	OrdStatusCodeSet	Y	
151	LeavesQty	Qty	Y	
14	CumQty	Qty	Y	
32	LastQty	Qty	Y	
31	LastPx	Price	Y	
20000	MyUDF1	String		
20001	MyUDF2	MyUDF2CodeSet		
> Component	StandardTrailer		Y	

Fields

ID (Tag)	Name	Datatype
20000	MyUDF1	String
20001	MyUDF2	MyUDF2CodeSet

Code Set

MyUDF2CodeSet ↗

ID	35930
Type	int

Name	Value
MyValue1	1
MyValue2	2

Used in messages: [ExecutionReport](#)