INTERNATIONAL STANDARD

ISO 3531-2

First edition
2022-04

Financial services — Financial information eXchange session layer —

Part 2:
FIX session layer

Reference number
ISO 3531-2:2022(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Table of Contents

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by FIX Trading Community (as FIX Session Layer Technical Specification) and drafted in accordance with its editorial rules. It was assigned to Technical Committee ISO/TC 68, *Financial services*, Subcommittee SC 9, *Information exchange for financial services* and adopted under the “fast-track procedure”.

A list of all parts in the ISO 3531 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

FIX session protocol was written to be independent of any specific communications protocol (e.g. X.25, async, TCP/IP) or physical medium (e.g. copper, fibre, satellite) chosen for electronic data delivery. It offers a reliable stream where a message is delivered once and in order. The FIX session layer is designed to survive and resume operation in the event of the loss of transport level connections caused by any type of failure, including network outage, application failure or computer hardware failures.

The session layer is concerned with the ordered delivery of data while the application level defines business-related data content. This document focuses on the ordered delivery of data using the “FIX session protocol”.

The FIX session protocol is implemented using the FIX tagvalue encoding syntax for the standard header, standard trailer and the session level messages which make up the FIX session protocol. It is possible to send messages using other encodings, such as the other FIX-defined encodings (e.g. FIXML, SBE, JSON, GPB, ASN.1) or non-FIX-defined encodings (e.g. XML, FpML, ISO 20022 XML, JSON).

The Financial Information eXchange session layer is used to provide reliable and recoverable messaging for electronic trading. The protocol is intended for use by asset managers, trading firms, brokerages, trading venues, clearing houses, custodians, depositories, asset servicers and others involved in the trading life cycle activities of a wide range of financial instruments. The FIX session layer functionality is a realization of the ISO/IEC 7498-1 Open System Interconnection basic reference model level 5 session layer.

Financial services — Financial information eXchange session layer —

Part 2: FIX session layer

1 Scope

This document provides the normative specification of the FIX session layer standard and its session profiles.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

3.1
session layer message
message carried over the FIX session that is integral to the operation of the FIX session

3.2
application message
message that is carried over a FIX session to accomplish some business purpose

Note 1 to entry: Examples of a business purpose include an order to buy or sell a financial instrument, reporting market data or reporting an execution of a trade.

3.3
message type
identifier (code) that defines the type of message being sent

Note 1 to entry: The message type for the FIX session is a case-sensitive string encoded in the MsgType(35) field.

3.4
valid FIX message
session or application message that is a tagvalue encoded string of octets that is properly formed according to the FIX tagvalue encoding specification

3.5
FIX session processor
combination of computer hardware, firmware and software that implements the FIX session layer

Note 1 to entry: Commonly referred to as a FIX Engine.

3.6
initiator
FIX session processor that establishes the transport layer connection and initiates the session via transmission of the initial Logon(35=A) message

3.7
acceptor
FIX session processor that is able to establish a transport layer connection and receive Logon(35=A) requests from FIX session initiators to start or resume a FIX session

3.8
rules of engagement
specification, usually provided in document form, that describes a specific use of FIX

Note 1 to entry: Often referred to as a FIX service offering. FIX Orchestra is a standard that may be used to specify machine-readable rules of engagement.

3.9
peer
FIX session processor being communicated with over a FIX session

Note 1 to entry: The peer of the initiator is the acceptor. The peer of the acceptor is the initiator. The initiator and acceptor are peers.

3.10
counterparty
firm, legal entity or individual agreeing to use the FIX session layer to conduct some form of business endeavour

3.11
NextNumIn
Each FIX session processor must keep track of the next message sequence number it is expecting to receive from its peer to guarantee ordered delivery of messages over the life of a FIX session that may span multiple FIX connections. The next expected incoming sequence number (NextNumIn) is compared to the value in the MsgSeqNum(34) field in each message received from the peer.

3.12
NextNumOut
Each FIX session processor must keep track of the next outbound sequence number (NextNumOut) it will send to its peer over a FIX connection.

3.13
retransmission
resending of a message that was previously sent in order to resynchronize the FIX session

Note 1 to entry: A retransmitted message uses the original MsgSeqNum(34) value with PossDupFlag(43)=Y.

3.14
resend
application message being resent by an application layer because it has not received an application layer acknowledgement for the message

Note 1 to entry: A resend may only be initiated by the application layer, not the session layer. The determination of whether the message was previously received is the responsibility of the application layer, not the session layer.

3.15
gap fill
process to resolve gaps in message sequence numbers within a FIX session

3.16
application version
The FIX session layer provides fields to communicate versions of the application layer messages. The FIX Trading Community defines standard application versions.

3.17
extension pack
approved addition to the FIX standard

Note 1 to entry: The granularity of an extension pack may vary widely from a single enumeration value addition to the definition of entire new categories of messages. Extension packs are identified by a sequential integer number and must be applied in order. An extension pack is considered available for use if it has been approved and published by the FIX Global Technical Committee. Extension packs are created on an as-needed basis and are generally driven by community requests. An extension pack is cumulative, in that the artefacts (Orchestra file, FIXimate) include all previous extension packs. When an extension pack is published, it becomes the *FIX Latest* version of FIX.

3.18
session profile
extension or restriction on the use of the standard session layer messages that can be used to represent context of usage

Note 1 to entry: FIX4, FIXT, and LFIXT are the current FIX session profiles.

3.19
CompID
alphanumeric identifier for the entity associated with a FIX session

Note 1 to entry: As this is likely a financial markets participant company, the name was viewed colloquially as a company identifier abbreviated as a *CompID*.

3.20
SubID
subidentifier optionally available to identify a subentity within a CompID

Note 1 to entry: The SubID may be used to identify a specific trader or a subunit of a business entity. The use of SubIDs is at the discretion of the counterparties. Certain regulatory regimes globally require the use of SubID to identify specific traders.

3.21
LocationID
location identifier providing additional location information either geographical or within a trading desk on a trading floor

Note 1 to entry: The use of LocationIDs is at the discretion of the counterparties.

3.22
transport layer connection
A FIX session relies on a transport layer to provide for ordered delivery of messages and message recovery during the life of the transport layer connection. The FIX session does not require a specific transport layer, although TCP/IP is widely used and is a de facto standard transport layer for FIX sessions. TCP/IP provides an ordered reliable delivery of a stream of bytes during the life of the TCP connection. The FIX session processor reads this stream identifying FIX message boundaries.

3.23
in-band communication

transmission of control information or metadata about the application layer or session layer in application and session FIX message types over the FIX session

Note 1 to entry: Sending the HeartBtInt(108) field in the Logon(35=A) message is an example of in-band transmission of control information. Providing version information about a FIX service in the CstmApplVerID(1129) on the Logon(35=A) message is another example.

3.24
out-of-band communication

exchange of control information or metadata about a FIX session via a separate communication mechanism than the FIX session

Note 1 to entry: Providing the TestRequestThreshold in a rules of engagement document is an example of exchanging information out-of-band. Providing version information about a FIX service on a website or in a rules of engagement document is another example of out-of-band communication. Providing access to a specification online via a website or web service would be another example of out-of-band communication. Even though this is electronic communication, it is not being done over the FIX session to which that information applies.

3.25
TestRequestThreshold

amount of time expressed as a multiplier of the heart beat interval before a TestRequest(35=1) message is sent to the peer when the heartbeat interval has been exceeded without receiving a message from the peer

Note 1 to entry: This value may be specified out-of-band in a rules of engagement.

3.26
SendingTimeThreshold

amount of time expressed in seconds in which the SendingTime(52) value in an inbound message differs from the system time available to the receiving FIX session processor

Note 1 to entry: This value may be specified out-of-band in a rules of engagement.

3.27
LogoutAckThreshold

amount of time expressed in seconds that a FIX session processor that has transmitted a Logout(35=5) request will wait for the Logout(35=5) acknowledgement before terminating the transport layer connection

Note 1 to entry: This value may be specified out-of-band in a rules of engagement.

4 FIX session

4.1 General

A **FIX session** is a bidirectional stream of ordered messages between two peers within a continuous sequence number series beginning with 1. A single FIX session can exist across multiple sequential (not concurrent) FIX connections, which means that peers may intentionally or unintentionally connect and disconnect multiple times while maintaining a single FIX session. The FIX session can be thought of as a bidirectional durable session sharing characteristics of the guaranteed delivery and durable subscriber enterprise integration architecture patterns.

A **FIX connection** consists of three parts: logon process, message exchange (inclusive of resynchronization of state) and possible logout process over a transport layer connection. A FIX connection may be concluded by the unrecoverable loss of the transport layer, a system failure or an application failure.

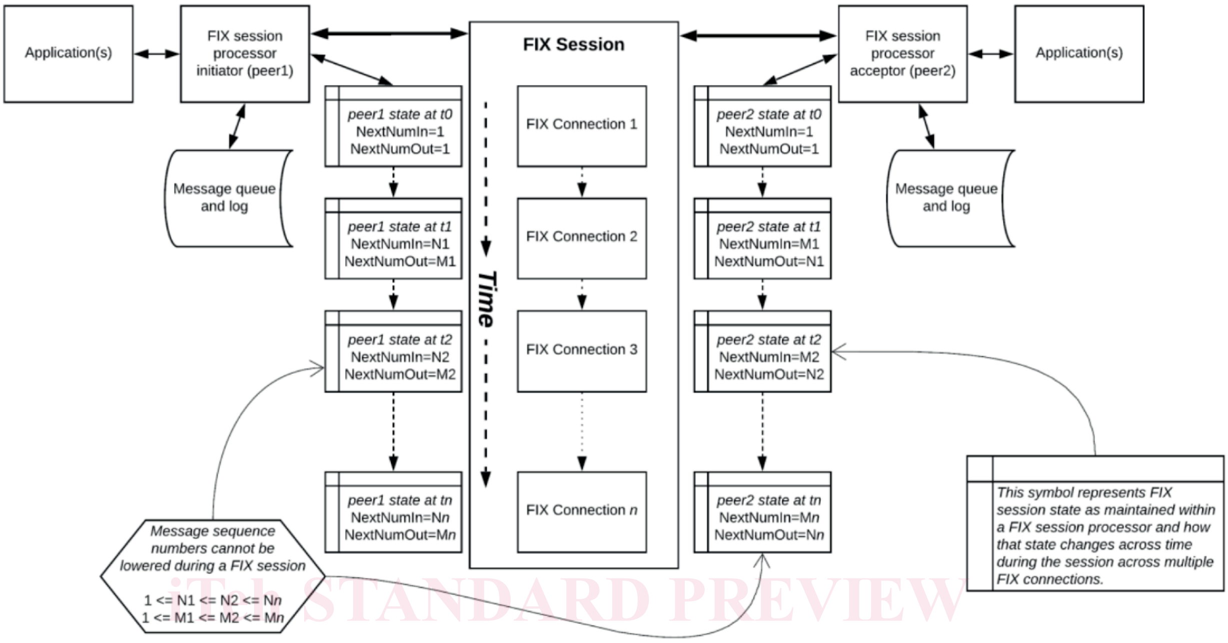

Figure 1 — Conceptual view of FIX session layer

Connecting parties shall bilaterally agree upon the time the FIX session is started and the duration of a FIX session.[1] A FIX session is often configured to correspond to a certain period of time, such as a trading day, calendar day or a trading session. A FIX session may extend beyond multiple periods by counterparty agreement.

The FIX session is based on an optimistic model. Normal delivery of data are assumed (i.e. no session layer acknowledgement of individual messages) with errors in delivery identified by message sequence number gaps.

4.2 Sequence numbers

All messages sent over a FIX session shall be identified by a unique sequence number in the MsgSeqNum(34) field.

A FIX session shall start with a next expected outgoing sequence number (NextNumOut) of 1 and a next expected incoming sequence number (NextNumIn) of 1.

A FIX session processor must maintain the NextNumOut and NextNumIn for the entire FIX session.

A FIX session processor shall persist the NextNumOut and NextNumIn in order to support FIX session recovery across multiple FIX connections.

Resetting NextNumOut to 1 and NextNumIn to 1, for whatever reason, shall constitute the beginning of a new FIX session.

FIX session layer and application layer messages shall share the same sequence number space.

[1] Firms may document the session layer configuration using the FIX Orchestra interface specification.

Each FIX session layer and application layer message sent consumes the next outbound sequence number, incrementing NextNumOut by 1.

Each FIX session layer and application layer message received consumes the next inbound sequence number, incrementing NextNumIn by 1.

A FIX session must always have peer1.NextNumIn < = peer2.NextNumOut and peer2.NextNumIn < = peer1.NextNumOut to be considered in a valid and recoverable state.

4.3 Identifying the FIX session

4.3.1 General

A FIX session is identified by the unique combination of BeginString(8) + initiator CompID + acceptor CompID.

4.3.2 The FIX session profile

BeginString(8) shall be used to identify the FIX session profile or version of FIX.

These FIX session profiles are defined within this specification.

Table 1 — The FIX session profiles

FIX session profile	BeginString(8)	Description
FIX.4.2	FIX.4.2	The FIX session profile for use with the FIX.4.2 application layer.
FIX4	FIX.4.4	The FIX session profile backward compatible with FIX.4.4 recommended when counterparties will only be using a single application version during the FIX session, such as *FIX Latest*.
FIXT	FIXT.1.1	The FIX session profile that must be used when mixing multiple application versions over the same FIX session. May be used with a single application version of FIX such as *FIX Latest*.
LFIXT	FIXT.1.1	Lightweight FIXT restricted session layer message recovery[2] to simplify the protocol while maintaining compatibility with FIXT when using LFIXT compatible model of operation.

FIX session acceptor may be configured to support multiple FIX session profiles (or *FIX versions* prior to FIX.4.4) over the same transport layer with only one FIX session profile (or *FIX version* prior to FIX.4.4) being used per FIX session.

The FIX session acceptor should use the incoming Logon(35 = A) request BeginString(8) to identify the FIX session profile or FIX version being requested by the initiator.

4.3.3 Identification of FIX session peers

FIX relies on alphanumeric strings known as CompIDs to identify the initiator and the acceptor of FIX messages.

Counterparties must agree to their respective CompID values, which act as an identifier for the peer.

Each message sent over a FIX session layer must contain the sending entity in SenderCompID(49) and the receiving (destination) entity in TargetCompID(56). This requirement applies to both session layer and application layer messages.

[2] The LFIXT session profile uses the same BeginString(8) value as the FIXT session profile. The use of LFIXT and its mode of operation must be agreed upon out-of-band by counterparty agreement.

FIX service processors must validate the SenderCompID(49) and TargetCompID(56) of each message and ensure that it is identical to the values present in the FIX connection Logon(35 = A) message. A discrepancy in the SenderCompID(49)+TargetCompID(56) pair should result in the termination of the FIX connection by sending a Logout(35 = A) message using the Text(58) field to indicate the error, followed by termination of the transport layer connection.

4.3.4 Validation of SendingTime(52)

SendingTime(52) must be set to the time the message is transmitted by the FIX session processor, not the time the message was queued for transmission.

SendingTime(52) must be in UTC (Universal Time Coordinated).

SendingTime(52) should be within the reasonable time of a synchronized time source of the receiving FIX session processor. This *SendingTimeThreshold* is highly dependent upon the type of application using the FIX session layer. The *SendingTimeThreshold* may be as low as seconds for high volume, low latency applications up to several minutes for certain order routing applications.

The *SendingTimeThreshold* may be defined and communicated to counterparties out-of-band within the rules of engagement.

The receiving peer shall transmit a Reject(35 = 3) message with SessionRejectReason(373) set to 10 (SendingTime Accuracy Problem) followed by a Logout(35 = 5) request when the SendingTime(52) is not within a specified tolerance of a synchronized clock.

4.3.5 Additional fields available for peer identification

The FIX session layer provides additional fields which may be used to further identify recipients within the respective counterparties.

Each counterparty may provide a subidentifier (SenderSubID(50) and TargetSubID(57)) known as SubID.

Each counterparty may provide a location identifier (SenderLocationID(142) and TargetLocationID(143)) known as LocationID.

The SubID and LocationID may vary across messages across the same FIX session.

The counterparties must agree upon the context and semantics of the SubID and LocationID if used.

4.4 Establishing a FIX connection

Establishing a FIX connection involves three distinct operations: creation of a transport layer connection, acceptance with optional authentication of the initiator by the acceptor and message synchronization (initialization).

The initiator shall establish a transport layer connection with the acceptor to establish a FIX connection.

4.4.1 Transport layer requirements

FIX session layer requires that the transport layer provides ordered and lossless message delivery and full duplex operation for the life of the transport layer connection.

FIX implementations that use the TCP/IP protocol as the transport layer must use the FIX-over-TLS (FIXS) specification which specifies the use of Transport Layer Security (TLS) with the FIX session layer to provide transport layer encryption.

The initiator shall send a Logon(35 = A) request message to initiate a FIX connection.

The Logon(35 = A) request message must always be the first message transmitted over a FIX connection. If the acceptor receives anything other than a valid Logon(35 = A) request message, an error should be logged and the transport layer connection terminated without Logout(35 = 5) processing.