

FIX Session Layer Test Cases

Version 1.0 – June 2020

THIS DOCUMENT IS THE FINAL VERSION OF A FIX TECHNICAL STANDARD. THIS VERSION HAS BEEN APPROVED BY THE GLOBAL TECHNICAL COMMITTEE AS THE FINAL STEP IN CREATING A NEW FIX TECHNICAL STANDARD OR A NEW VERSION OF AN EXISTING FIX TECHNICAL STANDARD. POTENTIAL ADOPTERS ARE STRONGLY ENCOURAGED TO USE ONLY THE FINAL VERSION. EXISTING ADOPTERS ARE STRONGLY ENCOURAGED TO UPGRADE TO THE FINAL VERSION.

Table of Contents

1	Scope	4
2	Normative references	5
3	Terms and definitions	6
4	FIX session layer test cases	7
4.1	Applicability	7
4.2	Test cases	7
4.3	Buy-side-oriented (session initiator) Logon and session initiation test case	7
4.3.1	Scenario 1B Connect and Send Logon message	7
4.4	Sell-side-oriented (session acceptor) Logon and session initiation test case	7
4.4.1	Scenario 1S Receive Logon message	7
4.4.2	Scenario 2S. Receive any message other than a Logon message	8
4.5	Test cases applicable to all FIX systems	8
4.5.1	Scenario 2 Receive Message Standard Header	8
4.5.2	Scenario 3 Receive Message Standard Trailer	11
4.5.3	Scenario 4 Send Heartbeat message	11
4.5.4	Scenario 5 Receive Heartbeat message	12
4.5.5	Scenario 6 Send Test Request	12
4.5.6	Scenario 7 Receive Reject message	12
4.5.7	Scenario 8 Receive Resend Request message	12
4.5.8	Scenario 9 Synchronize sequence numbers	12
4.5.9	Scenario 10 Receive Sequence Reset (Gap Fill)	13
4.5.10	Scenario 11 Receive Sequence Reset (Reset)	14
4.5.11	Scenario 12 Initiate logout process	14
4.5.12	Scenario 13 Receive Logout message	14
4.5.13	Scenario 14 Receive application or session layer message	14
4.5.14	Scenario 15 Send application or session layer messages to test normal and abnormal behavior/response	16
4.5.15	Scenario 16 Queue outgoing messages	17
4.5.16	Scenario 17 Support encryption	17
4.5.17	Scenario 18 Support third party addressing	19
4.5.18	Scenario 19 Test PossResend handling	19
4.5.19	Scenario 20 Simultaneous Resend request test	19

DISCLAIMER

THE INFORMATION CONTAINED HEREIN AND THE FINANCIAL INFORMATION EXCHANGE PROTOCOL (COLLECTIVELY, THE “FIX PROTOCOL”) ARE PROVIDED “AS IS” AND NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL MAKES ANY REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, AS TO THE FIX PROTOCOL (OR THE RESULTS TO BE OBTAINED BY THE USE THEREOF) OR ANY OTHER MATTER AND EACH SUCH PERSON AND ENTITY SPECIFICALLY DISCLAIMS ANY WARRANTY OF ORIGINALITY, ACCURACY, COMPLETENESS, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SUCH PERSONS AND ENTITIES DO NOT WARRANT THAT THE FIX PROTOCOL WILL CONFORM TO ANY DESCRIPTION THEREOF OR BE FREE OF ERRORS. THE ENTIRE RISK OF ANY USE OF THE FIX PROTOCOL IS ASSUMED BY THE USER.

NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL SHALL HAVE ANY LIABILITY FOR DAMAGES OF ANY KIND ARISING IN ANY MANNER OUT OF OR IN CONNECTION WITH ANY USER’S USE OF (OR ANY INABILITY TO USE) THE FIX PROTOCOL, WHETHER DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL (INCLUDING, WITHOUT LIMITATION, LOSS OF DATA, LOSS OF USE, CLAIMS OF THIRD PARTIES OR LOST PROFITS OR REVENUES OR OTHER ECONOMIC LOSS), WHETHER IN TORT (INCLUDING NEGLIGENCE AND STRICT LIABILITY), CONTRACT OR OTHERWISE, WHETHER OR NOT ANY SUCH PERSON OR ENTITY HAS BEEN ADVISED OF, OR OTHERWISE MIGHT HAVE ANTICIPATED THE POSSIBILITY OF, SUCH DAMAGES.

DRAFT OR NOT RATIFIED PROPOSALS (REFER TO PROPOSAL STATUS AND/OR SUBMISSION STATUS ON COVER PAGE) ARE PROVIDED “AS IS” TO INTERESTED PARTIES FOR DISCUSSION ONLY. PARTIES THAT CHOOSE TO IMPLEMENT THIS DRAFT PROPOSAL DO SO AT THEIR OWN RISK. IT IS A DRAFT DOCUMENT AND MAY BE UPDATED, REPLACED, OR MADE OBSOLETE BY OTHER DOCUMENTS AT ANY TIME. THE FIX GLOBAL TECHNICAL COMMITTEE WILL NOT ALLOW EARLY IMPLEMENTATION TO CONSTRAIN ITS ABILITY TO MAKE CHANGES TO THIS SPECIFICATION PRIOR TO FINAL RELEASE. IT IS INAPPROPRIATE TO USE FIX WORKING DRAFTS AS REFERENCE MATERIAL OR TO CITE THEM AS OTHER THAN “WORKS IN PROGRESS”. THE FIX GLOBAL TECHNICAL COMMITTEE WILL ISSUE, UPON COMPLETION OF REVIEW AND RATIFICATION, AN OFFICIAL STATUS (“APPROVED”) OF/FOR THE PROPOSAL AND A RELEASE NUMBER.

No proprietary or ownership interest of any kind is granted with respect to the FIX Protocol (or any rights therein), except as expressly set out in FIX Protocol Limited’s Copyright and Acceptable Use Policy.

© Copyright 2003-2020 FIX Protocol Limited, all rights reserved



FIX Technical Standard Specifications by [FIX Protocol Ltd.](#) are licensed under a [Creative Commons Attribution-NonDerivatives 4.0 International License](#). Based on a work at <https://github.com/FIXTradingCommunity/>.

1 Scope

This document provides a set of mandatory and optional compliance tests applicable to all versions of the FIX session layer standard.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

- Financial Information eXchange – FIX TagValue Encoding Technical Specification
- Financial Information eXchange – FIX Session Layer Technical Specification
- FIX 4.2 Specification with 20010501 Errata <https://www.fixtrading.org/standards/fix-4-2/>
- FIX 4.4 Specification with 20030618 Errata <https://www.fixtrading.org/standards/fix-4-4/>

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 11404:2007 *Information technology – General-Purpose Datatypes (GPD)* and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <https://www.iso.org/obp>

4 FIX session layer test cases

4.1 Applicability

This document was last revised September 20, 2002 at which time FIX version 4.3 with Errata 20020930 was the latest version of the FIX Protocol. Note that future amendments to this document may be found on the FIX website and any version of this document published on a later date takes precedence over this version of the document. This document is applicable to all supported versions of the FIX session layer (4.2, 4.4, FIXT) except where explicitly indicated.

4.2 Test cases

These test cases are from the perspective of the FIX system being tested. The FIX system receives the “Condition / Stimulus” and is expected to take the appropriate action as defined by “Expected Behavior”.

4.3 Buyside-oriented (session initiator) Logon and session initiation test case

4.3.1 Scenario 1B Connect and Send Logon message

Mandatory

Condition/Stimulus	Expected Behavior
a. Establish transport layer connection.	Successfully established transport layer connection with peer.
b. Send Logon(35=A) request message.	Logon(35=A) acknowledgement message sent by peer.
c. Valid Logon(35=A) acknowledgement message received.	If MsgSeqNum(34) is too high, send ResendRequest(35=2).
d. Invalid Logon(35=A) acknowledgement message received.	<ol style="list-style-type: none"> 1. Generate an error condition in test output. 2. (Optional) Send Reject(35=3) message with RefSeqNum(45) identifying Logon(35=A) message’s MsgSeqNum(34) and Text(58) referencing error condition. 3. Send Logout(35=5) message with Text(58) referencing error condition. 4. Disconnect.
e. Receive any message other than a Logon(35=A) message.	<ol style="list-style-type: none"> 1. Log an error “first message not a logon” 2. (Optional) Send Reject(35=3) message with RefSeqNum(45) identifying message’s MsgSeqNum(34) and Text(58) referencing error condition. 3. (Optional) Send Logout(35=5) message with Text(58) referenceing error condition. 4. Disconnect.

4.4 Sellside-oriented (session acceptor) Logon and session initiation test case

4.4.1 Scenario 1S Receive Logon message

Mandatory

Condition/Stimulus	Expected Behavior
a. Valid Logon(35=A) request message received.	<ol style="list-style-type: none"> Respond with Logon(35=A) acknowledgement message. If MsgSeqNum(34) > NextNumIn send ResendRequest(35=2).
b. Logon(35=A) message received with duplicate identity (e.g. same IP, port, SenderCompID(49), TargetCompID(56), etc. as existing connection).	<ol style="list-style-type: none"> Generate an error condition in test output. Disconnect without sending a message (Note: sending a Reject or Logout(35=5) would consume a MsgSeqNum(34)).
c. Logon(35=A) message received with unauthenticated/non-configured identity (e.g. invalid SenderCompID(49), invalid TargetCompID(56), invalid source IP address, etc. vs. system configuration).	<ol style="list-style-type: none"> Generate an error condition in test output. Disconnect without sending a message (Note: sending a Reject or Logout(35=5) would consume a MsgSeqNum(34)).
d. Invalid Logon(35=A) message.	<ol style="list-style-type: none"> Generate an error condition in test output. (Optional) Send Reject(35=3) message with RefSeqNum(45) identifying Logon(35=A) message's MsgSeqNum(34) with Text(58) referencing error condition. Send Logout(35=5) message with Text(58) referencing error condition. Disconnect.

4.4.2 Scenario 2S. Receive any message other than a Logon message

Mandatory

Condition/Stimulus	Expected Behavior
First message received is not a Logon(35=A) message.	<ol style="list-style-type: none"> Log an error "First message not a logon". Disconnect.

4.5 Test cases applicable to all FIX systems

4.5.1 Scenario 2 Receive Message Standard Header

Mandatory

Condition/Stimulus	Expected Behavior
a. MsgSeqNum(34) received as expected.	Accept MsgSeqNum(34) for the message.
b. MsgSeqNum(34) higher than expected.	Respond with ResendRequest(35=2) message.
c. MsgSeqNum(34) lower than expected without PossDupFlag(43) set to Y. Exception: SequenceReset(35=4).	<ol style="list-style-type: none"> Whenever possible it is recommended that FIX engine attempt to send a Logout(35=5) message with a text message of "MsgSeqNum too low, expecting X but received Y". (Optional) Wait for Logout(35=5) message response (Note: likely will have inaccurate MsgSeqNum(34)) or

Condition/Stimulus	Expected Behavior
	<p>wait 2 seconds, whichever comes first.</p> <ol style="list-style-type: none"> 3. Disconnect. 4. Generate an error condition in test output.
<p>d. Garbled message received.</p>	<ol style="list-style-type: none"> 1. Consider garbled and ignore message (do not increment NextNumIn) and continue accepting messages. 2. Generate a warning condition in test output.
<p>e. PossDupFlag(43) set to Y; OrigSendingTime(122) specified is less than or equal to SendingTime(52) and MsgSeqNum(34) lower than expected.</p> <p>Note: OrigSendingTime(122) should be earlier than SendingTime(52) unless the message is being resent within the same second during which it was sent.</p>	<ol style="list-style-type: none"> 1. Check to see if MsgSeqNum(34) has already been received. 2. If already received then ignore the message, otherwise accept and process the message.
<p>f. PossDupFlag(43) set to Y; OrigSendingTime(122) specified is greater than SendingTime(52) and MsgSeqNum(34) as expected.</p> <p>Note: OrigSendingTime(122) should be earlier than SendingTime(52) unless the message is being resent within the same second during which it was sent.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 10 (SendingTime accuracy problem). 2. Increment NextNumIn. 3. Optional: <ul style="list-style-type: none"> • Send Logout(35=5) message referencing inaccurate SendingTime(52) value. • (Optional) Wait for Logout(35=5) message response (Note: likely will have inaccurate SendingTime(52)) or wait 2 seconds, whichever comes first. • Disconnect. • Generate an error condition in test output.
<p>g. PossDupFlag(43) set to Y and OrigSendingTime(122) not specified.</p> <p>Note: Always set OrigSendingTime(122) to the time when the message was originally sent-not the present SendingTime(52) and set PossDupFlag(43)=Y when responding to a ResendRequest(35=2).</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 1 (Required tag missing). 2. Increment NextNumIn.
<p>h. BeginString(8) value received as expected and specified in testing profile and matches BeginString(8) on outbound messages.</p>	<p>Accept BeginString(8) for the message.</p>
<p>i. BeginString(8) value received did not match value expected and specified in testing profile or does not match BeginString(8) on outbound messages.</p>	<ol style="list-style-type: none"> 1. Send Logout(35=5) message referencing incorrect BeginString(8) value. 2. (Optional) Wait for Logout(35=5) message response (Note: likely will have incorrect BeginString(8)) or wait <i>LogoutAckThreshold</i> seconds, whichever comes first. 3. Disconnect. 4. Generate an error condition in test output.

Condition/Stimulus	Expected Behavior
j. SenderCompID(49) and TargetCompID(56) values received as expected and specified in testing profile.	Accept SenderCompID(49) and TargetCompID(56) for the message.
k. SenderCompID(49) and TargetCompID(56) values received did not match values expected and specified in testing profile.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with RefTagID(371) to identify field with mismatched value, and SessionRejectReason(373) set to 9 (CompID problem). 2. Increment NextNumIn. 3. Send Logout(35=5) message referencing incorrect SenderCompID(49) or TargetCompID(56) value. 4. (Optional) Wait for Logout(35=5) message response (Note: likely will have incorrect SenderCompID(49) or TargetCompID(56)) or wait 2 seconds, whichever comes first. 5. Disconnect. 6. Generate an error condition in test output.
l. BodyLength(9) value received is correct.	Accept BodyLength(9) for the message.
m. BodyLength(9) value received is not correct.	<ol style="list-style-type: none"> 1. Consider garbled and ignore message (do not increment NextNumIn) and continue accepting messages. 2. Generate a warning condition in test output.
n. SendingTime(52) value received is specified in UTC (Universal Time Coordinated) also known as GMT) or is not within <i>SendingTimeThreshold</i> seconds of a synchronized time source.	Accept SendingTime(52) for the message.
<p>o. SendingTime(52) value received is either not specified in UTC (Universal Time Coordinated) or is not GMT) or is not within a within <i>SendingTimeThreshold</i> seconds of a synchronized time source.</p> <p>Rationale: Verify system clocks on both sides are in sync and that SendingTime(52) is current time.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 10 (SendingTime accuracy problem). 2. Increment NextNumIn. 3. Send Logout(35=5) message referencing inaccurate SendingTime(52) value. 4. (Optional) Wait for Logout(35=5) message response (Note: likely will have inaccurate SendingTime(52)) or wait 2 seconds, whichever comes first. 5. Disconnect. 6. Generate an error condition in test output.
p. MsgType(35) value received is valid (defined in spec or classified as user-defined).	Accept MsgType(35) for the message.
q. MsgType(35) value received is not valid (defined in spec or classified as user-defined).	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 11 (Invalid MsgType). 2. Increment NextNumIn. 3. Generate a warning condition in test output.
r. MsgType(35) value received is valid (defined in spec or	1. Send BusinessMessageReject(35=j) with

Condition/Stimulus	Expected Behavior
classified as user-defined) but not supported or registered in testing profile.	BusinessRejectReason(380) set to 3 (Unsupported Message Type). 2. Increment NextNumIn. 3. Generate a warning condition in test output.
s. BeginString(8), BodyLength(9), and MsgType(35) are first three fields of message.	Accept the message.
t. BeginString(8), BodyLength(9), and MsgType(35) are not the first three fields of message.	1. Consider garbled and ignore message. 2. Do not increment NextNumIn. 3. Continue accepting messages. 4. Generate a warning condition in test output.

4.5.2 Scenario 3 Receive Message Standard Trailer

Mandatory

Condition/Stimulus	Expected Behavior
a. Valid CheckSum(10).	Accept message
b. Invalid CheckSum(10).	1. Consider garbled and ignore message. 2. Do not increment NextNumIn. 3. Continue accepting messages. 4. Generate a warning condition in test output.
c. Garbled message.	1. Consider garbled and ignore message. 2. Do not increment NextNumIn. 3. Continue accepting messages. 4. Generate a warning condition in test output.
d. CheckSum(10) is last field of message, value has length of 3, and is delimited by <SOH>.	Accept message.
e. CheckSum(10) is not the last field of message, value does not have length of 3, or is not delimited by <SOH>.	1. Consider garbled and ignore message. 2. Do not increment NextNumIn. 3. Continue accepting messages. 4. Generate a warning condition in test output.

4.5.3 Scenario 4 Send Heartbeat message

Mandatory

Condition/Stimulus	Expected Behavior
a. No data sent during preset heartbeat interval (HeartBtInt(108) field).	Send Heartbeat(35=0) message.
b. TestRequest(35=1) message received.	Send Heartbeat(35=0) message with TestRequest(35=1) message's TestReqID(112).

4.5.4 Scenario 5 Receive Heartbeat message

Mandatory

Condition/Stimulus	Expected Behavior
Valid Heartbeat(35=0) message.	Accept Heartbeat(35=0) message.

4.5.5 Scenario 6 Send Test Request

Mandatory

Condition/Stimulus	Expected Behavior
No data received during preset heartbeat interval (HeartBtInt(108)) + "some reasonable period of time" (use 20% of HeartBtInt(108)).	<ol style="list-style-type: none"> 1. Send TestRequest(35=1) message. 2. Track and verify that a Heartbeat(35=0) message with the same TestReqID(112) is received (may not be the next message received).

4.5.6 Scenario 7 Receive Reject message

Mandatory

Condition/Stimulus	Expected Behavior
Valid Reject(35=3) message.	<ol style="list-style-type: none"> 1. Increment NextNumIn. 2. Continue accepting messages.

4.5.7 Scenario 8 Receive Resend Request message

Mandatory

Condition/Stimulus	Expected Behavior
Valid ResendRequest(35=2)	Respond with application layer messages and SequenceReset(35=4) with GapFillFlag(123)=Y for session layer messages in request range according to message recovery rules.

4.5.8 Scenario 9 Synchronize sequence numbers

Optional

Condition/Stimulus	Expected Behavior
Application failure that caused loss of session state.	<ol style="list-style-type: none"> 1. Establish new FIX connection. 2. Reset NextNumOut to an arbitrarily large value that is greater than the last known NextNumOut value. 3. Reset NextNumIn value to 1. 4. Send SequenceReset(35=4) with GapFillFlag(123)=N and NewSeqNo(36) set to a number arbitrarily larger than the last known NextNumOut value.

Condition/Stimulus	Expected Behavior
	5. Process retransmitted messages and gap fills from peer. OR 1. Both peers reset their NextNumIn and NextNumOut values to 1. 2. Start new FIX session

4.5.9 Scenario 10 Receive Sequence Reset (Gap Fill)

Mandatory

Condition/Stimulus	Expected Behavior
a. Receive SequenceReset(35=4) with GapFillFlag(123)=Y message with NewSeqNo(36) > MsgSeqNum(34) and MsgSeqNum(34) > NextNumIn	Issue ResendRequest(35=2) to fill gap between last expected MsgSeqNum(34) & received MsgSeqNum(34).
b. Receive SequenceReset(35=4) with GapFillFlag(123)=Y message with NewSeqNo(36) > MsgSeqNum(34) and MsgSeqNum(34) = NextNumIn	Set next expected sequence number = NewSeqNo(36).
c. Receive SequenceReset(35=4) with GapFillFlag(123)=Y message with NewSeqNo(36) > MsgSeqNum(34) and MsgSeqNum(34) < NextNumIn and PossDupFlag(43)=Y.	Ignore message.
d. Receive SequenceReset(35=4) with GapFillFlag(123)=Y message with NewSeqNo(36) > MsgSeqNum(34) and MsgSeqNum(34) < NextNumIn and without PossDupFlag(43)=Y.	1. If possible, send a Logout(35=5) message with text of "MsgSeqNum too low, expecting X received Y," prior to disconnecting FIX session. 2. (Optional) Wait for Logout(35=5) message response (Note: likely will have inaccurate MsgSeqNum(34)) or wait 2 seconds, whichever comes first. 3. Disconnect. 4. Generate an error condition in test output.
e. Receive SequenceReset(35=4) with GapFillFlag(123)=Y message with NewSeqNo(36) <= MsgSeqNum(34) and MsgSeqNum(34) = NextNumIn	Send Reject(35=3) message with message "attempt to lower sequence number, invalid value NewSeqNo(36)=<x>".

4.5.10 Scenario 11 Receive Sequence Reset (Reset)

Mandatory

Condition/Stimulus	Expected Behavior
a. Receive SequenceReset(35=4) with GapFillFlag(123)=N message with NewSeqNo(36) > NextNumIn	<ol style="list-style-type: none"> 1. Accept the SequenceReset(35=4) message without regard to its MsgSeqNum(34). 2. Set NextNumIn to NewSeqNo(36).
b. Receive SequenceReset(35=4) with GapFillFlag(123)=N message with NewSeqNo(36) = NextNumIn	<ol style="list-style-type: none"> 1. Accept the SequenceReset(35=4) message without regard to its MsgSeqNum(34). 2. Generate a warning condition in test output.
c. Receive SequenceReset(35=4) with GapFillFlag(123)=N message with NewSeqNo(36) < NextNumIn	<ol style="list-style-type: none"> 1. Accept the SequenceReset(35=4) message without regard to its MsgSeqNum(34). 2. Send Reject(35=3) message with SessionRejectReason(373) set to 5 (Value is incorrect (out of range) for this tag). 3. Do NOT change the value of NextNumIn. 4. Generate an error condition in test output.

4.5.11 Scenario 12 Initiate logout process

Mandatory

Condition/Stimulus	Expected Behavior
Initiate logout.	<ol style="list-style-type: none"> 1. Send Logout(35=5) message. 2. Wait for counterparty to respond with Logout(35=5) message <i>LogoutAckThreshold</i> seconds (Note: may not be received if communications problem exists). If not received, generate a warning condition in test output. 3. Disconnect.

4.5.12 Scenario 13 Receive Logout message

Mandatory

Condition/Stimulus	Expected Behavior
a. Receive valid Logout(35=5) acknowledgement in response to a Logout(35=5) request.	Disconnect without sending a message.
b. Receive valid Logout(35=5) request message unsolicited.	<ol style="list-style-type: none"> 1. Send Logout(35=5) acknowledgement message. 2. Wait for counterparty to disconnect up to 10 seconds. If max exceeded, disconnect and generate an error condition in test output.

4.5.13 Scenario 14 Receive application or session layer message

Mandatory

Condition/Stimulus	Expected Behavior
a. Receive field identifier (tag number) not defined in	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with

Condition/Stimulus	Expected Behavior
<p>specification.</p> <p>Exception: undefined tag used is specified in testing profile as user-defined.</p>	<p>SessionRejectReason(373) set to 0 (Invalid tag number).</p> <ol style="list-style-type: none"> 2. Increment NextNumIn. 3. Generate an error condition in test output.
<p>b. Receive message with a required field identifier (tag number) missing.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 1 (Required tag missing). 2. Increment NextNumIn. 3. Generate an error condition in test output.
<p>c. Receive message with field identifier (tag number) which is defined in the specification but not defined for this message type.</p> <p>Exception: undefined tag used is specified in testing profile as user-defined for this message type.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 2 (Tag not defined for this message type). 2. Increment NextNumIn. 3. Generate an error condition in test output.
<p>d. Receive message with field identifier (tag number) specified but no value (e.g. "55=<SOH>" vs. "55=IBM<SOH>").</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 4 (Tag specified without a value). 2. Increment NextNumIn. 3. Generate an error condition in test output.
<p>e. Receive message with incorrect value (out of range or not part of valid list of enumerated values) for a particular field identifier (tag number).</p> <p>Exception: undefined enumeration values used are specified in testing profile as user-defined.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 5 (Value is incorrect (out of range) for this tag). 2. Increment inbound MsgSeqNum(34). 3. Generate an error condition in test output.
<p>f. Receive message with a value in an incorrect data format (syntax) for a particular field identifier (tag number).</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 6 (Incorrect data format for value). 2. Increment NextNumIn. 3. Generate an error condition in test output.
<p>g. Receive a message in which the following is not true: Standard Header fields appear before Body fields which appear before Standard Trailer fields.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 14 (Tag specified out of required order). 2. Increment NextNumIn. 3. Generate an error condition in test output.
<p>h. Receive a message in which a field identifier (tag number) which is not part of a repeating group is specified more than once.</p>	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 13 (Tag appears more than once). 2. Increment NextNumIn. 3. Generate an error condition in test output.

Condition/Stimulus	Expected Behavior
i. Receive a message with repeating groups in which the “count” field value for a repeating group is incorrect.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 16 (Incorrect NumInGroup count for repeating group). 2. Increment NextNumIn. 3. Generate an error condition in test output.
j. Receive a message with repeating groups in which the order of repeating group fields does not match the specification.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 15 (Repeating group fields out of order). 2. Increment NextNumIn 3. Generate an error condition in test output.
k. Receive a message with a field of a data type other than “data” which contains one or more embedded <SOH> values.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) (Non “data” value includes field delimiter (<SOH> character)). 2. Increment NextNumIn 3. Generate an error condition in test output. <p>OR</p> <ol style="list-style-type: none"> 1. Consider garbled and ignore message. 2. Generate an error condition in test output
l. Receive a message when application layer processing or system is not available (optional).	<ol style="list-style-type: none"> 1. Send BusinessMessageReject(35=j) with BusinessRejectReason(380) set to 4 (Application not available). 2. Increment NextNumIn. 3. Generate a warning condition in test output.
m. Receive a message in which a conditionally required field is missing.	<ol style="list-style-type: none"> 1. Send BusinessMessageReject(35=j) with BusinessRejectReason(380) set to 5 (Conditionally Required Field Missing). 2. Increment NextNumIn. 3. Generate an error condition in test output.
n. Receive a message in which a field identifier (tag number) appears in both cleartext and encrypted section but has different values.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 7 (Decryption problem). 2. Increment NextNumIn. 3. Generate an error condition in test output.

4.5.14 Scenario 15 Send application or session layer messages to test normal and abnormal behavior/response

Optional

Condition/Stimulus	Expected Behavior
Send more than one message of the same type with header and body fields ordered differently to verify acceptance. (Exclude those which have	Message accepted and subsequent messages’ MsgSeqNum(34) are

Condition/Stimulus	Expected Behavior
restrictions regarding order.)	accepted.

4.5.15 Scenario 16 Queue outgoing messages

Mandatory

Condition/Stimulus	Expected Behavior
a. Message to send/queue while disconnected.	<p>Queue outgoing messages. Note there are two valid approaches:</p> <ol style="list-style-type: none"> 1. Queue without regard to MsgSeqNum(34). <ol style="list-style-type: none"> a. Store data for messages 2. Queue each message with the next MsgSeqNum(34) value. <ol style="list-style-type: none"> b. Store data for messages in such a manner as to use and “consume” the next MsgSeqNum(34). <p>Note: SendingTime(52) must contain the time the message is sent, not the time the message was queued.</p>
b. Reconnect with queued messages.	<ol style="list-style-type: none"> 1. Complete logon process (establish transport layer connection and exchange Logon(35=A) messages). 2. Synchronize the FIX session, if required (inbound Logon(35=A) message MsgSeqNum(34) > NextNumIn). 3. Recommended short delay, or use TestRequest(35=1) or Heartbeat(35=0) message to verify FIX session synchronization completed. 4. Note there are two valid queuing approaches. <ol style="list-style-type: none"> a. Queue without regard to MsgSeqNum(34): <ol style="list-style-type: none"> i. Send queued messages with new MsgSeqNum(34) values (greater than Logon(35=A) message’s MsgSeqNum(34)). b. Queue each message with the next MsgSeqNum(34) value: <ol style="list-style-type: none"> ii. (Note: Logon(35=A) message’s MsgSeqNum(34) will be greater than the queued messages’ MsgSeqNum(34)). iii. Counterparty will issue ResendRequest(35=2) requesting the range of missed messages. iv. Resend each queued message with PossDupFlag(43) set to Y. <p>Note: SendingTime(52) must contain the time the message is sent, not the time the message was queued.</p>

4.5.16 Scenario 17 Support encryption

Optional

Condition/Stimulus	Expected Behavior
a. Receive Logon(35=A) request message with valid, supported EncryptMethod(98).	<ol style="list-style-type: none"> 1. Accept the message. 2. Perform the appropriate decryption and encryption method readiness. 3. Respond with Logon(35=A) acknowledgement message with the same EncryptMethod(98).

Condition/Stimulus	Expected Behavior
b. Receive Logon(35=A) message with invalid or unsupported EncryptMethod(98).	<ol style="list-style-type: none"> 2. Send Reject(35=3) message with SessionRejectReason(373) set to 7 (Decryption problem). 3. Increment NextNumIn. 4. Send Logout(35=5) request message referencing invalid or unsupported EncryptMethod(98) value. 5. (Optional) Wait for Logout(35=5) acknowledgement message response (Note: could have decrypt problems) or wait <i>LogoutAckThreshold</i> seconds, whichever comes first. 6. Disconnect. 7. Generate an error condition in test output.
c. Receive message with valid SignatureLength(93) and Signature(89) values.	Accept the message.
d. Receive message with invalid SignatureLength(93) value.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 8 (Signature problem). 2. Increment inbound MsgSeqNum(34). 3. Generate an error condition in test output.
e. Receive message with invalid Signature(89) value.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 8 (Signature problem). 2. Increment inbound MsgSeqNum(34). 3. Generate an error condition in test output. <p>Or consider decryption error or message out of order, ignore message (do not increment inbound MsgSeqNum(34)) and continue accepting messages.</p>
f. Receive message with a valid SecureDataLen(90) value and a SecureData(91) value that can be decrypted into valid, parsable cleartext.	Accept the message.
g. Receive message with invalid SecureDataLen(90) value.	<ol style="list-style-type: none"> 1. Consider garbled and ignore message. 2. Generate an error condition in test output
h. Receive message with a SecureData(91) value that cannot be decrypted into valid, parsable cleartext.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 7 (Decryption problem). 2. Increment NextNumIn. 3. Generate an error condition in test output.
i. Receive message with one or more fields not present in the unencrypted portion of the message that “must be unencrypted” according to the spec.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 7 (Decryption problem). 2. Increment NextNumIn.

Condition/Stimulus	Expected Behavior
	3. Generate an error condition in test output.
j. Receive message with incorrect handling of “leftover” characters (e.g. when length of cleartext prior to encryption is not a multiple of 8) according to the specified EncryptMethod(98). ¹	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with SessionRejectReason(373) set to 7 (Decryption problem). 2. Increment NextNumIn. 3. Generate an error condition in test output.

4.5.17 Scenario 18 Support third party addressing

Optional

Condition/Stimulus	Expected Behavior
a. Receive messages with OnBehalfOfCompID(115) and DeliverToCompID(128) values expected as specified in testing profile and with correct usage.	Accept messages.
b. Receive messages with OnBehalfOfCompID(115) or DeliverToCompID(128) values not specified in testing profile or incorrect usage.	<ol style="list-style-type: none"> 1. Send Reject(35=3) message with RefTagID(371) to identify field with incorrect values, and SessionRejectReason(373) set to 9 (CompID problem). 2. Increment NextNumIn. 3. Generate an error condition in test output.

4.5.18 Scenario 19 Test PossResend handling

Mandatory

Condition/Stimulus	Expected Behavior
a. Receive message with PossResend(97)=Y and application layer check of message-specific ID indicates that it has already been seen on this session.	<ol style="list-style-type: none"> 1. Ignore the message. 2. Generate a warning condition in test output.
b. Receive message with PossResend(97)=Y and application layer check of message-specific ID indicates that it has NOT yet been seen on this session.	Accept and process the message normally.

4.5.19 Scenario 20 Simultaneous Resend request test

Mandatory

Condition/Stimulus	Expected Behavior
Receive a ResendRequest(35=2) message while having sent and awaiting complete set of responses to a ResendRequest(35=2) message.	<ol style="list-style-type: none"> 1. Perform resend of requested messages. 2. Send ResendRequest(35=2) message to request missed messages if gap still exists.

¹ For block ciphers which transform a fixed-size block of data (usually 64 bits) into another fixed-size block (possibly 64 bits long again) using a function selected by the key.