THOMSON REUTERS MATCHING API

THOMSON REUTERS MATCHING CERTIFICATION GUIDE

Document Version 1.6
Date of issue: 3 February 2012

THOMSON REUTERS

Thomson Reuters, by publishing this document, does not guarantee that any information contained herein is and will remain accurate or that use of the information will ensure correct and faultless operation of the relevant service or equipment. Thomson Reuters, its agents and employees, shall not be held liable to or through any user for any loss or damage whatsoever resulting from reliance on the information contained herein.

This document contains information proprietary to Thomson Reuters and may not be reproduced, disclosed, or used in whole or part without the express written permission of Thomson Reuters.

Any Software, including but not limited to, the code, screen, structure, sequence, and organization thereof, and Documentation are protected by national copyright laws and international treaty provisions. This manual is subject to U.S. and other national export regulations.

Nothing in this document is intended, nor does it, alter the legal obligations, responsibilities or relationship between yourself and Thomson Reuters as set out in the contract existing between us.

Contents

About this document

In this document

This document provides a guide to the certification of a client FIX/RFA implementation and describes the objectives of each test, and the message format(s) required for each trade workflow.

What's new in this document?

- Version 1.6 new test cases (27 and 28) are added to test drop copy functionality. Certification tiers (Chapter 3.5) updated for drop copy users. This version has also been updated for self certification scenario 7, 8 and 9.
- Version 1.5 a new test case was added: Ability to handle Logout Message.
- Version 1.4 a new test case was added: Ability to handle Heartbeat messages, and a new self certification scenario: #6 for test request message.
- Version 1.3 a section was added Certification Tiers and applicable test cases.
- Version 1.2 a new test case was added: Ability to handle unacknowledged matches.

References

Reference #	Document Name	Date	Version
1	Thomson Reuters Matching Interface User Guide		
2	Thomson Reuters Matching Data Feed Direct User Guide		

Terms and acronyms

Term	Description
P2PS	Point to Point Server
AOR	Automatic Order Replacement
STU	Special Ticket User
PB	Prime Broker
PBC	Prime Broker Client
IDEAL	Single Order with Both BID and OFFER sides
BMR	Business Message Reject
MFG	Matching FIX Gateway
RFA	Reuters Foundation API

Term	Description
FIX	Financial Information Exchange
TCR	FIX Trade Capture Report
ROE	Rules of Engagement
AQ	AutoQuote
CFG	Client FIX Gateway
MDFD	Market Data Feed Direct
HTG	Host To Go (Matching Test system)
MAPI	Matching API (FIX/RFA)
DCU	Drop Copy User

Feedback

If you have any comments on this document please contact the Thomson Reuters TPG Documentation team.

.

Chapter 1 Introduction

As part of Thomson Reuters going commitment towards providing a world-leading FX Matching service we have made a significant investment into its New Matching Host Program (NMP).The Thomson Reuters New Matching Host program delivers an enhanced Matching system that is highly performant, scalable and extensible. The first phase of the NMP successfully upgraded the core Matching engine. The second phase delivers a completely new set of Matching API solutions. Later phases include a new spot Matching user interface along with several other functional enhancements such as the Depth of Book (DOB).

The Matching API offers two separate interfaces, one dedicated for price discovery and another dedicated for order management. The Reuters Foundation API (RFA) is used for price discovery and an industry standard FIX interface has been introduced for order management and post trade processing. The Matching API solution discussed herein will ultimately replace our current Auto Quote API.

The Matching FIX Gateway (MFG) supports FIX version 5.0 SP2 EP100 for the purpose of order management and post trade processing. Every FIX session is associated with a Matching Site id (aka Credit Code). The FIX client application is permissioned to create one or more FIX session with the MFG. It is only possible to establish a user session over an authorized FIX session. Thomson Reuters Matching requires all users to be first authenticated by Matching Host before they can connect to their Market Data Feed Direct Feed (MDFD). Once authenticated via FIX, the requesting user is provided an authentication token, which then can be used for login to the MDFD via RFA connection. The authorization token is renewed by the Matching FIX Gateway each time the user logs into the FIX interface however the token is valid for a trading session.

Chapter 2 Certification Purpose

The purpose of Matching API certification process is to ensure that your applications have correctly implemented key trading workflows as outlined in Matching API User Guides. This document presents a number of test scenarios which are designed to ensure that FIX and RFA applications are correctly implemented for usage against the Thomson Reuters Matching service. The successful completion of the certification process helps protect the integrity of Thomson Reuters Matching service and minimizes the risk of any unexpected production level trading issues.

The test scenarios will demonstrate that your applications have the ability to send/receive session and application level messages to and from the Matching service. The testing also assures that the messages sent by your FIX and RFA applications can be successfully processed by Matching FIX Gateway and Point to Point Servers (P2PS).

All Matching API clients are required to complete the test scenarios outlined in Chapter 5 prior to going live on Matching service. Tests pertain to both new client applications and for the recertification of existing implementations of the Matching API. The successful completion of a conformance test is required for any release of a client application, whether new or revised.

Matching API clients are also required to successfully complete the self certification section outlined in Chapter 6. User application also need to be aware of the notification messages outlined in chapter 7 as these messages can impact your ability to successfully trade on Matching.

Chapter 3 Certification Procedure

This section describes the rules, resources and infrastructure required to perform the certification tests.

3.1 Rules

- Participant is required to provide separate log files for each test case for review by MAPI On-boarding team. These log files should have names starting with "Test Case #". Participant is required to open a BizBuzz report for conformance and upload the zipped log files to this report. Feedback on the conformance results will be provided via the report.
- Participant is required to provide test trade details in "Matching API certification.xls" spreadsheet provided along with this document. If a client is not attempting a specific test case then he needs to indicate so in the certification spreadsheet.
- Chapter 6 outlines self certification tests which client is required to attempt on its own and update the spreadsheet with his results.
- Chapter 7 describes user notifications from Matching Host and Client is required to update certification spreadsheet with the workflow that their application has implemented to handle these user notification message.
- Participant is required to liaise with MAPI on-boarding team to test the session level test cases.
- A Participant's application version used during certification procedure is the one known as being "certified" and any major changes applied after the certification testing should be discussed with MAPI on-boarding team so they can advise about the need for a recertification.
- MAPI on-boarding team may ask a participant to reattempt failed test cases.
- **Participants are required to follow the input specifications as given in each test case (for ex. currency, amount and side).**

3.2 Resources

Chapter 5 of this document details the required test cases to pass the conformance. Client will have to liaise with MAPI on-boarding team to complete user session level test cases.

3.3 Certification Environment

MAPI Certification testing is required to be performed in the Matching Host to Go (HTG) environment which is functionally equivalent to the Thomson Reuters Matching Production environment. Pricing is provided to the Matching test environment from an auto price injector which allows for the testing of both screened and unscreened prices.

3.4 Check list

- The Matching HTG is running and is in 'Trade' mode.
- Client should have acquired a SenderCompID, SenderLocationID, FIX session login, FIX Trader login, TargetCompID and targetSubID for testing. Client should have connection details (IP/Port for FIX and RFA) to connect to the HTG environment.
- Client should have acquired a certificate for session level SSL authentication, if needed.

3.5 Certification Tiers and applicable test cases

Below summarizes the certification testing requirement for various user types:

PBC Users: PBC Users are required to attempt all session level test cases (1-6, 25, 26). Client can choose the required testcases from Test cases (8-20) as appropriate to their order entry workflow. Test case 24 is mandatory and if for some reason client doesn't support the usd/xxx instrument (as per the test requirement), then they are required to update the certification spreadsheet with the workflow their application has to handle this scenario. Client gateway should also have correct handling of self certification scenario #6. Client applications are also required to have CancelAll ability so they should complete test case #17 or #18 as appropriate to their implementation.

PB Users: PB users are required to attempt all session level test cases (1-6, 25, 26). PB users can liaise with on-boarding team to assist them with some order executions so they can verify receiving and correct handling of TCR in their back office application. Client gateway should also have correct handling of self certification scenario #6.

Interbank Users: Interbank users are required to attempt all session level test cases (1-6, 25, 26). Client can choose the required testcases from Test cases (8-20) as appropriate to their order entry workflow. Test case 24 is mandatory and if for some reason client doesn't support usd/xxx instrument (as per the test requirement), then they are required to update the certification spreadsheet with the workflow their application has to handle this scenario. Client gateway should also have correct handling of self certification scenario #6. Client applications are also required to have CancelAll ability so they should complete test case #17 or #18 as appropriate to their implementation.

MDFD Users: MDFD Users are required to attempt all session level test cases (1-7, 25, 26) and self certification scenarios as given in Chapter 6.

Drop Copy Users: Drop copy allows Matching subscribers to optionally route their trade tickets to risk management and position keeping systems while at the same time providing the corresponding trade details back to the order originator. Drop copy users are required to complete test cases (1, 3-6, 24-28). Drop copy users will also receive below user notification messages (35=CB) as detailed in Chapter 7.

- Host State change
- Start of Trading Session Countdown
- End of Trading Session Countdown

Chapter 4 Test Case Structure

Below is a test case template and structure:

Test Case#:	**Purpose:**
Description:	
Prerequisites:	
Test Steps:	
Input Specification:	**Expected Results:**
Sequence Diagram:	

Test Case #:	Unique identifier for each test case.
Purpose:	Purpose of this test.
Description:	Brief description.
Prerequisite:	Indicates specific actions that need to be taken before executing the test. This also sets any preconditions required before starting the test.
Test Steps:	The steps required to take the test.
Input Specifications:	Required input parameters
Expected Results:	Expected outcome of the test for Success and Failure criterion. In order for client to pass a test case, one of the **SUCCESS** scenarios should have been observed. Client may observe a different behaviour then explained in the SUCCESS scenario which could indicate a failure of test. **FAILURE** scenario is outlined to help clients troubleshoot a failure case.
Sequence Diagram:	To indicate sequence of events between Client application and Matching FIX Gateway. Blue dotted lines indicate multiple messages with same message type.

Chapter 5 Certification Test Cases

5.1 Reset password

Test Case 1:	**Purpose:** To verify that the user is able to reset password and login to FIX session correctly.

Description:

This test case will verify that client application has ability to reset password as it will be required for client application to do initial password reset during first time Production login.

Prerequisites:

- NMH is running and in Trading Mode.
- MFG is running.
- Both client and Matching test sessions should have their initial message sequence numbers as 1/1 to indicate start of trading week scenario.
- Client is aware of his original password.

Test Steps:

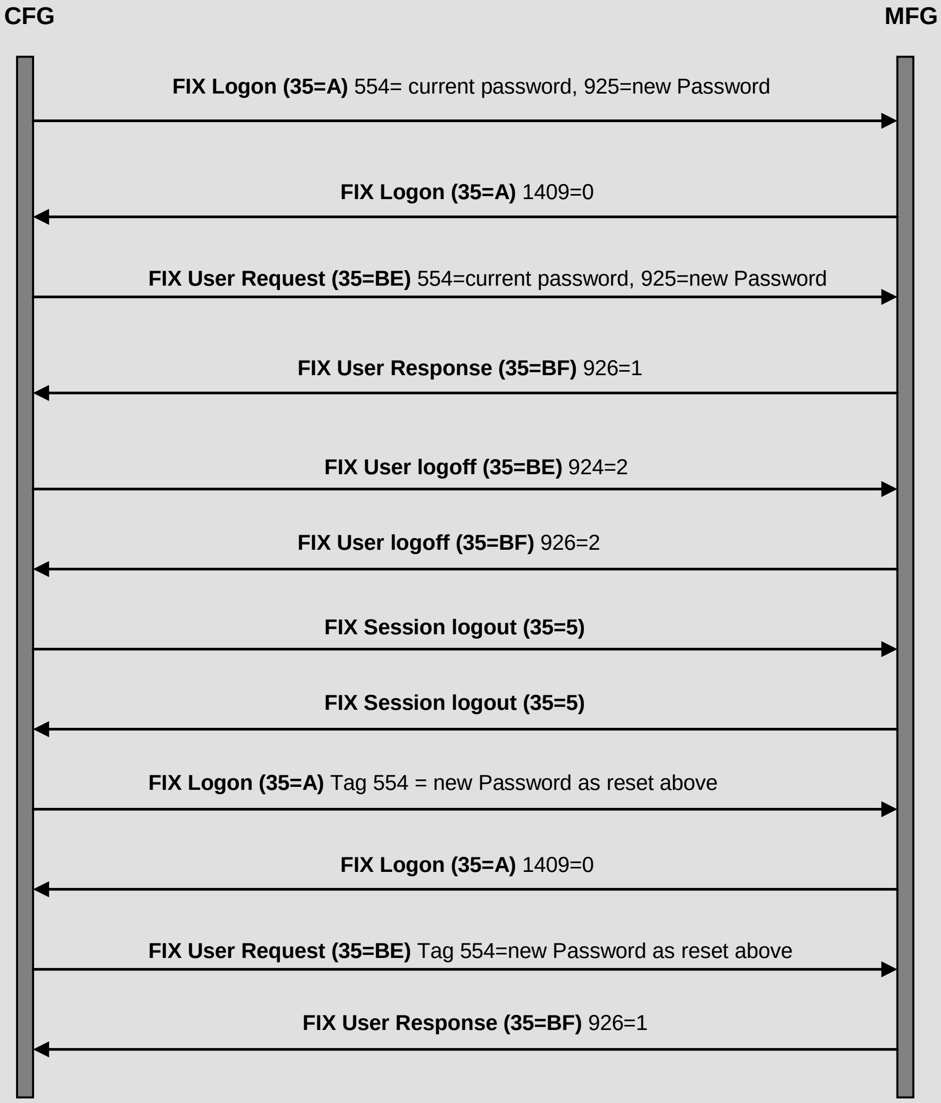
1. Client application will send a Session FIX Login message (35=A) to MFG with Tag 925 populated with New Password and Tag 554 with default/existing password.
2. Client should receive a success FIX Logon (35=A), Tag 1409=0 from MFG.
3. Client application will send a FIX User request (35=BE) message with Tag 925 populated with New Password and Tag 554 with existing password.
4. Client should receive a success FIX User response message (35=BF) with Tag 926=1.
5. Client should see stream of FIX Heartbeat messages (35=0) every 4 secs.
6. Next step, client is required to log off from FIX user and session logins.
7. Client should send FIX Session Login (35=A) and FIX User request (35=BE) with changed password and see if he is able to login to MFG correctly.

Input Specifications:	**Expected Results:**
FIX Logon Message (35=A) as defined in Matching Interface User Guide. Expected Incoming/Outgoing sequence numbers should be 1/1 for this test case as the required sequence numbers at the start of trading session. Below input params required for password roll in (35=A) message 553=*Username* *554= default password*	**SUCCESS:** 1. Matching API user should see a successful FIX Session Logon Message (35=A) with Tag 1409=0. Client should also see a success FIX User response message (35=BF) with 926=1 in both cases (session and user login before password reset and after password reset). MFG should receive heart beat messages from Matching API user at every 4 sec interval.

925=New Password Below input params required for password roll in (35=BE) message 924=1 553=*Username* *554= default password* *925=New Password*	**FAILURE:** 1. Client application receives disconnect without a FIX Logon message indicating an issue either with invalid password or network connectivity. In case of an invalid password in FIX User request (35=BE), MFG will respond with FIX User response message (35=BF) with invalid password error (926=4).

SUCCESS SCENARIO

User resets password successfully and able to relogin with the new reset password


CFG
MFG
FIX Logon (35=A) 554= current password, 925=new Password
FIX Logon (35=A) 1409=0
FIX User Request (35=BE) 554=current password, 925=new Password
FIX User Response (35=BF) 926=1
FIX User logoff (35=BE) 924=2
FIX User logoff (35=BF) 926=2
FIX Session logout (35=5)
FIX Session logout (35=5)
FIX Logon (35=A) Tag 554 = new Password as reset above
FIX Logon (35=A) 1409=0
FIX User Request (35=BE) Tag 554=new Password as reset above
FIX User Response (35=BF) 926=1

5.2 User login and subsequent notification messages

Test Case 2:

Purpose:

To verify that Matching user is able to correctly send a FIX User Request Message (35=BE) and able to handle subsequent response and notification messages.

Description:

This test case will verify that user is able to send a FIX User request message and able to handle the user response and other user notification messages from Matching FIX Gateway in response to the User request message.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.

Test Steps :

1. Send a FIX Session login message and receive a successful logon message from MFG.
2. Send a FIX User Request message (35=BE). User login should use a trader’s long name for SenderSubID field.
3. Receive a FIX User response (35=BF) with Tag 926=1/2.
4. If Tag 926=1 in user response (BF) message then user should receive a user notification message (35=CB) with the Host trading status (MTM:0) or (MTM:1).
5. A PBC user should also receive (35=CB) with the STU status message. Tag 58=”STU:0” will indicate STU is not logged in and Tag 58=”STU:1” will indicate STU is logged in. User can only submit trade messages if STU is logged in. User should escalate STU not logged in message to their PB bank.

Input Specification:

FIX User Request message (35=BE) as defined in Matching Interface User Guide.

Tag 553=Username

Tag 554=Password

Tag 50=Trader’s long name

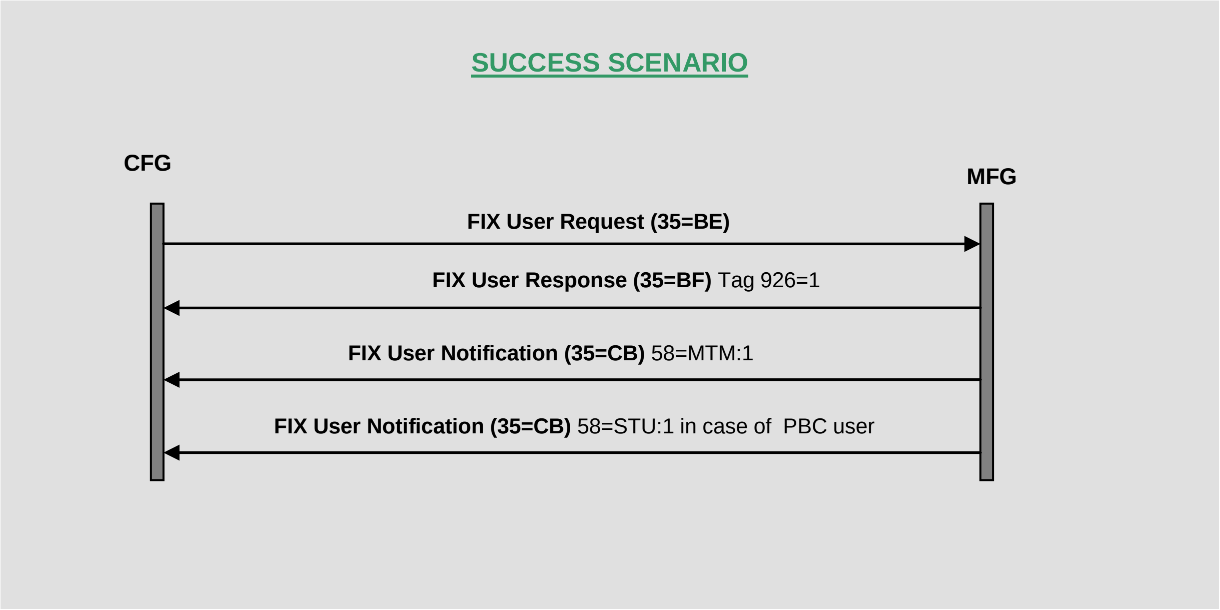
Expected Results:

SUCCESS:

1. Matching API user receives a User response message (35=BF) with Tag 926=1 (Success).
2. Matching API user receives subsequent User notification messages (35=CB) with Tag 58=’MTM:1’ (trading enabled) and 58=’STU:1’ (STU enabled message when the user is PBC user).

FAILURE:

1. Matching API user receives a User response message (35=BF) with Tag 926=2(FAILURE) indicating an issue with user login/password etc.


SUCCESS SCENARIO
CFG
MFG
FIX User Request (35=BE)
FIX User Response (35=BF) Tag 926=1
FIX User Notification (35=CB) 58=MTM:1
FIX User Notification (35=CB) 58=STU:1 in case of PBC user

Note: Clients who are looking to have multi user logins via same session should be validated for correct user login and order processing via atleast 2 trader user logins.

5.3 Client gateway sending Tag34 lower then expected by MFG

Test Case 3:	**Purpose:** To verify that the user is able to re-establish and resynchronize sequence number if a FIX session disconnect happens during trading hours and client’s outgoing sequence number (Tag 34) is lower then expected by MFG.

Description:

This test case will verify that client application is able to handle message sequencing errors when client is sending a lower outgoing Message sequence number (Tag 34) than expected and able to re-establish connection with MFG.

Prerequisites:

- Client has test system connectivity and session login details.
- NMH is running and in Trading Mode.
- MFG is running.
- Client should liaise with MAPI on-boarding team to manually reset sequence numbers from MFG and confirm successful completion of this test.
- Both client and Matching test sessions should have their initial message sequence numbers as 1/1.

Test Steps:

1. Client application will establish session with Matching FIX Gateway and send 5 New Order Single (35=D) messages. MDFD only session users need not send order messages, they can skip to line 3.
2. Client application will send a FIX Session Logout (35=5) message. This will trigger MFG to cancel all the Pending orders, reporting each cancellation with a FIX Execution Report followed by a Logout. This should end client session.
3. Increase incoming sequence number by 5 from MFG to simulate the expected behaviour. For ex. if the sequence numbers at MFG were 9 (incoming) and 16 (outgoing) then after session Logout these values can be set as 14(incoming) and 16 (outgoing) at MFG.
4. Client will send a FIX Logon message (ex. Tag 34=9 and Tag 789=16 from above step) as per their last communication with MFG.
5. Client will receive a FIX Logout message with Tag 58 populated with message sequencing error and expected sequence number.
6. Client application will send a FIX Session login with correct sequence number as expected by MFG and receive a success login message back from MFG.

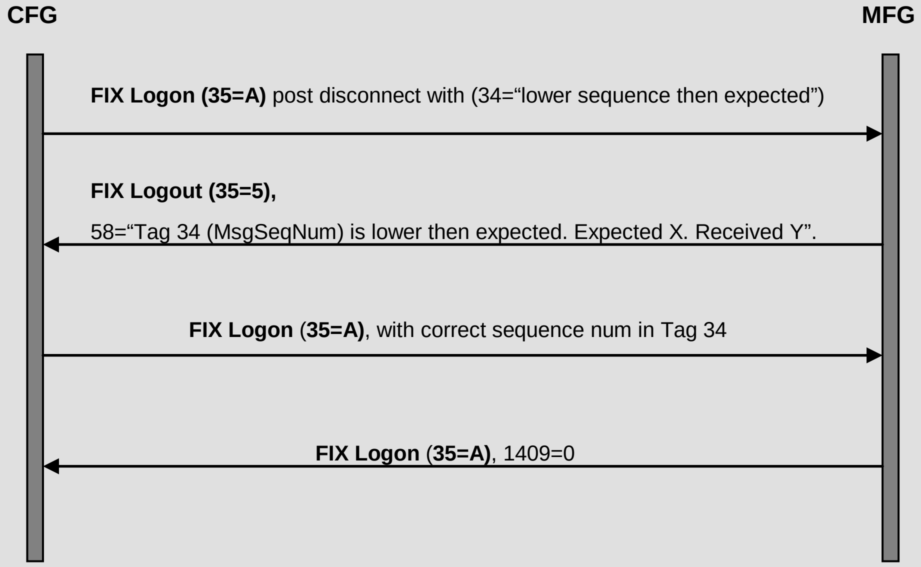
Input Specifications:	**Expected Results:**
FIX Logon Message (35=A) as defined in Matching Interface User Guide.	**SUCCESS:** 1. Client receives a Logout message from Matching FIX Gateway. Client resets its message sequence number and resends Logon message. Client receives a Logon Success message from Matching FIX

<table>
<tr><td></td><td>Gateway.

FAILURE:

1. MFG waits for sometime but Client application is unable to reset sequence number in next Logon message or next 20 minutes.</td></tr>
<tr><td colspan="2">

SUCCESS SCENARIO


CFG
MFG
FIX Logon (35=A) post disconnect with (34="lower sequence then expected")
FIX Logout (35=5),
58="Tag 34 (MsgSeqNum) is lower then expected. Expected X. Received Y".
FIX Logon (35=A), with correct sequence num in Tag 34
FIX Logon (35=A), 1409=0

</td></tr>
</table>

5.4 Client gateway sending Tag 34 higher then expected by MFG

Test Case 4:	**Purpose:** To verify that the user is able to re-establish and resynchronize sequence number if a FIX session disconnect happens during trading hours and client’s outgoing sequence number (Tag 34) is higher then expected by MFG.

Description:

This test case will verify that client application is able to handle message sequencing errors when Client is sending a higher Message sequence number (Tag 34) than expected by MFG.

Prerequisites:

- Client has test connection and session login details.
- NMH is running and in Trading Mode.
- MFG is running.
- Client should liaise with MAPI on-boarding team to manually reset sequence numbers from MFG and confirm successful completion of this test.
- Both client and Matching test sessions should have their initial message sequence numbers as 1/1.

Test Steps:

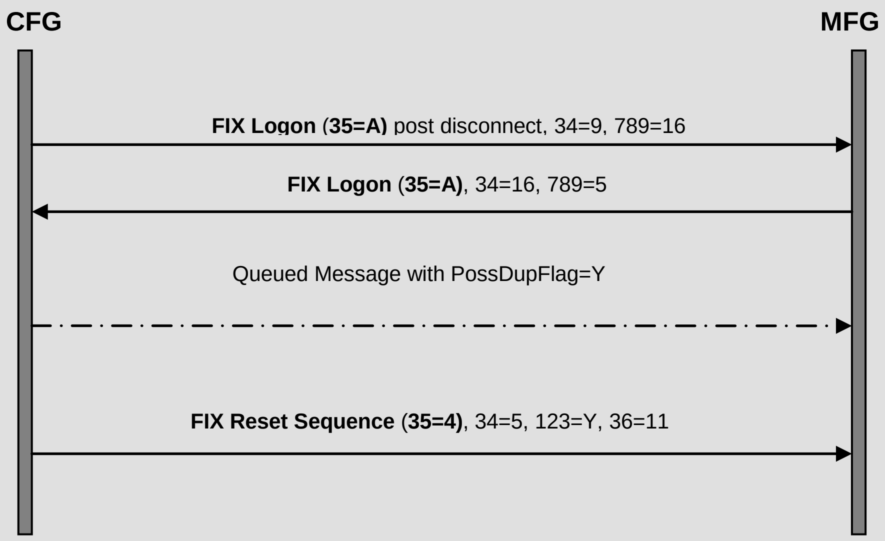
1. Client application will establish session with Matching FIX Gateway.
2. After sending 5 New Order Single (35=D) messages, Client application will send a FIX Session Logout (35=5) message. This will trigger MFG to cancel all the Pending orders, reporting each cancellation with a FIX Execution Report followed by a Logout. This should end client session. MDFD only session users need not send order messages, they can skip to line 3.
3. This test requires manually reducing the incoming sequence number from MFG. At this point it is required to make note of sequence numbers at MFG and reset the session which would cause sequence numbers at MFG to be 1/1. Only sequence numbers at MFG should be reset to 1/1, no changes required at CFG. Now set the Outgoing sequence number at MFG to be same as prior to session reset and set Incoming sequence number at MFG to be a lower value then what it was prior to session reset. For ex. If the sequence numbers at MFG were 9 (incoming) and 16 (outgoing) then after session reset these values can be set as 5 (incoming) and 16 (outgoing).
4. At relogin, Client will send a Login message with Tag34=9 and Tag 789=16 (or whatever the numbers were prior to session disconnect) since Client gateway hasn’t changed sequence numbers on its end. But MFG would be expecting Tag34=5 and Tag 789=16 from CFG since the sequence numbers were manually reset at MFG (as per step 3 above).
5. Client will receive a FIX Logon message from MFG with Tag 789 set to the MFG’s next expected sequence number (From above steps the next expected incoming sequence number at MFG is 5).
6. Client application is recommended to do Tag 789 validation and automatically resend missing messages to MFG (with PossDupFlag=Y) followed by FIX Reset request message (35=4). In this case since the incoming sequence number on MFG side was manually decreased to simulate the behaviour so client will have no recovery messages to send; client will send FIX Reset sequence message (35=4) with Tag 36 populated with new sequence number and Tag 34 populated with the desired incoming sequence number from MFG. Presence of Tag 36 indicates that the recipient should allow a gap in sequence numbers starting from the value in Tag 34 and ending with the value in Tag 36. In the event that client FIX Gateway is not doing tag 789 validation, MFG will send a resend request (35=2) and client FIX Gateway should send the missed messages (with PossDupFlag=Y) in response to that followed by a reset sequence message (35=4). If client FIX Gateway sends any message to MFG prior to recovery process, especially without the PossDupFlag=Y, then such messages will be ignored by

MFG.

7. The recipient of a FIX Sequence Reset message containing tag 123 = N should ignore the value of tag 34/MsgSeqNum in the message and expect the next message received to have a sequence number equal to the value sent in tag 36/NewSeqNum.

Input Specifications:	**Expected Results:**
FIX Logon Message (35=A) as defined in Matching Interface User Guide.	**SUCCESS:** 1. Client receives a Logon message from Matching FIX Gateway with Tag 789 lower than expected by client Gateway. Client will receive resend request from MFG and Client is required to retransmit all the missing messages with PossDupFlag=Y followed by ResetRequest message. **FAILURE:** 1. Matching FIX Gateway wait for sometime but Client application is unable to reset sequence number in next Logon message or next 20 minutes.

SUCCESS SCENARIO


CFG
MFG
FIX Logon (35=A) post disconnect, 34=9, 789=16
FIX Logon (35=A), 34=16, 789=5
Queued Message with PossDupFlag=Y
FIX Reset Sequence (35=4), 34=5, 123=Y, 36=11

Note: Sequence numbers above are for example only, during testing these values may differ. With the proper implementation of tag 789 resend requests are not used.

5.5 Client gateway sending Tag 789 lower then expected by MFG

Test Case 5:

Purpose:

To verify that the user is able to re-establish and resynchronize sequence number if a FIX session disconnect happens during trading hours and Client is sending Tag 789 lower then expected by MFG.

Description:

This test case will verify that client application has the ability to relogin and handle the recovery messages when client is sending Tag 789 lower then expected by MFG.

Prerequisites:

- Client has test connection and session login details.
- NMH is running and in Trading Mode.
- MFG is running.
- Client and Matching FIX Gateway will reset their incoming/outgoing sequence number to 1/1.

Test Steps:

1. Client application will establish session with Matching FIX Gateway and send 5 FIX New Order Single (35=D) messages. MDFD only session users and PB users need not send order messages, Outgoing sequence number at MFG will be manually increased to create this scenario.
2. Client will disconnect by dropping the socket connection which will trigger Cancellation of pending orders at Host.
3. Since MFG couldn't send cancelled Execution Reports to Client so the outgoing sequence number at MFG should be higher.
4. Client will reinitiate FIX Session login (35=A) and should receive FIX Logon message from MFG immediately followed by 5 Cancelled Execution Reports and Sequence Reset message (35=4) with NewSeqNo (Tag 36) populated with new reset sequence number.
5. The recipient of a FIX Sequence Reset message containing tag 123 = N should ignore the value of tag 34/MsgSeqNum in the message and expect the next message received to have a sequence number equal to the value sent in tag 36/NewSeqNum.

Input Specifications:

FIX Logon Message (35=A) as defined in Matching Interface User Guide.

Expected Results:

SUCCESS:

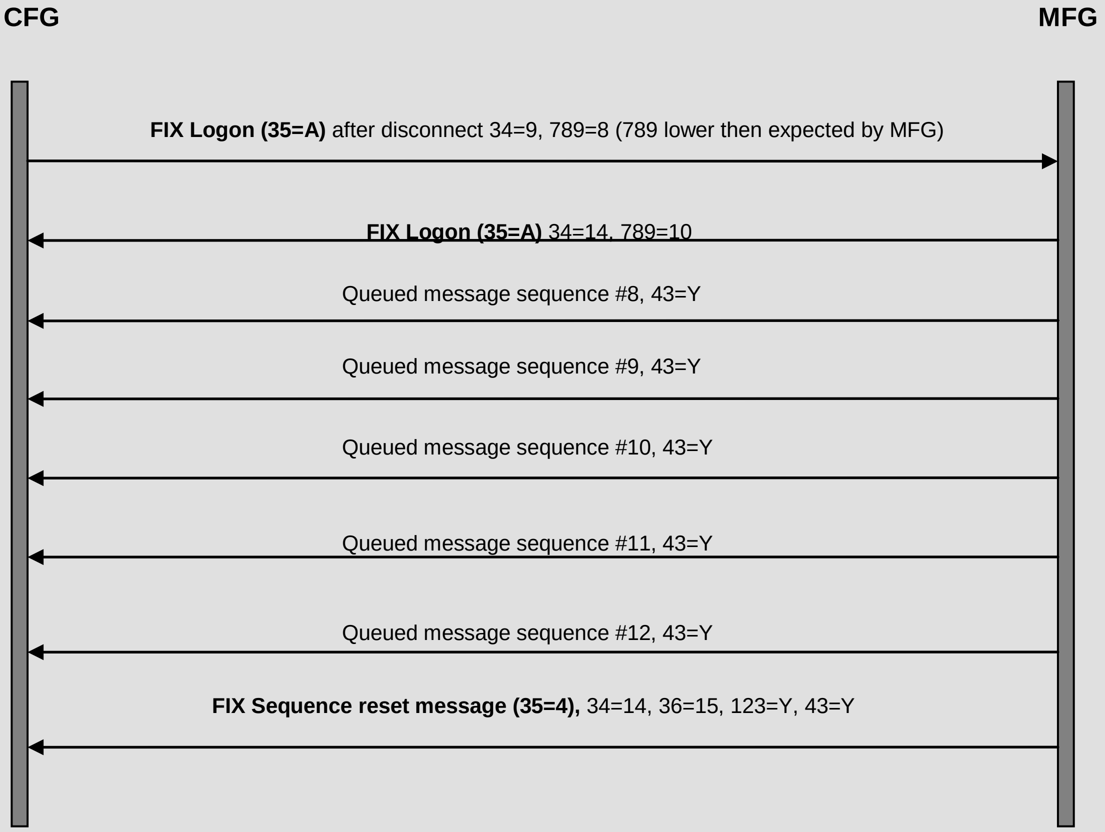
1. Client application is able to process all the recovered messages and successfully able to process sequence reset message for gap fill.

FAILURE:

1. Client application fails to validate Tag 789

<table>
<tr><td></td><td>2. Client application fails to process the recovered messages and responds with a Logout.
3. Client application fails to reset Tag 789 and continues to use the invalid sequence number.</td></tr>
<tr><td colspan="2">

SUCCESS SCENARIO


CFG
MFG
FIX Logon (35=A) after disconnect 34=9, 789=8 (789 lower then expected by MFG)
FIX Logon (35=A) 34=14, 789=10
Queued message sequence #8, 43=Y
Queued message sequence #9, 43=Y
Queued message sequence #10, 43=Y
Queued message sequence #11, 43=Y
Queued message sequence #12, 43=Y
FIX Sequence reset message (35=4), 34=14, 36=15, 123=Y, 43=Y

</td></tr>
</table>

Note: Sequence numbers above are for example only. During testing these values may differ.

5.6 Sequence reset prior to new trading session

Test Case 6:

Purpose:

To verify that the client can manually reset his sequence numbers at the beginning of the trading week in the event that the client sequence numbers were not reset from the previous trading session.

Description:

This test case will verify that the client can manually reset his sequence numbers at the beginning of the trading week in the event that the client sequence numbers were not reset from the previous trading session (e.g. over the weekend).

Prerequisites:

- Client has test connection and session login details.
- NMH is running and in Trading Mode.
- MFG is running.
- Client and Matching FIX Gateway will reset their incoming/outgoing sequence number to 1/1.

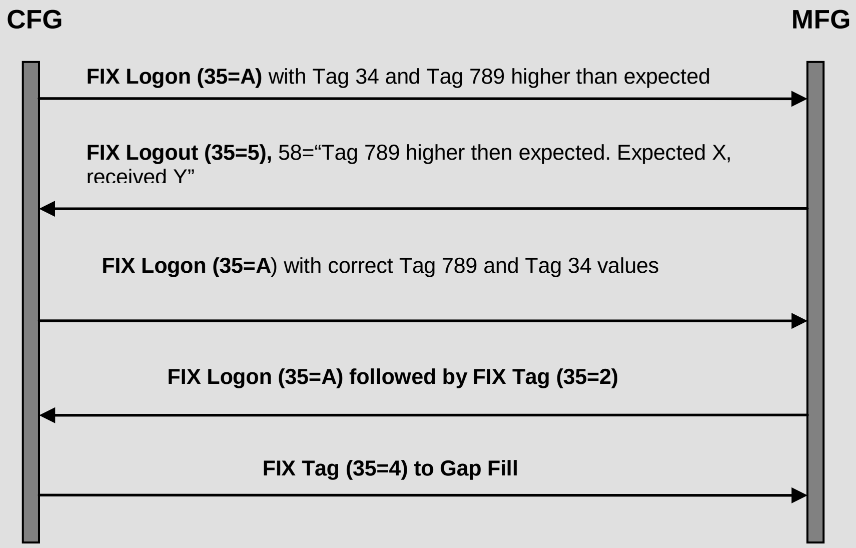
Test Steps:

1. Client application will establish session with Matching FIX Gateway.
2. Client will leave the session running till the incoming and outgoing sequence numbers reach 100.
3. Client application will logoff from the session.
4. This test requires manually resetting the session from MFG which will leave MFG with sequence numbers 1/1 (incoming/outgoing).
6. At relogin, Client will send a Login message with Tag34=100 and Tag 789=100 (or whatever the numbers were prior to session disconnect) since Client gateway hasn’t changed sequence numbers on its end.
5. MFG will respond with a logout message with tag 58 error, “Tag 789 (NextExpectedSeqNum) is higher then expected. Expected 1. Received 100”.
6. Client will reset Tag 789 to value 2 (as the next expected sequence number by MFG) and resend login.
7. Since client’s outgoing sequence number (FIX Tag 34) is still higher then what MFG is expecting so MFG will send a resend request (FIX Tag 35= 2) to fill the gap. At this point client application can choose to resend any missed messages with possDup flag set to “Yes” or send a Gap Fill (FIX Tag 35=4).
8. **For best practices, client should make sure that their session is reset (incoming and outgoing sequence numbers as 1) over the weekend, prior to initiating any connection to MFG on Sunday.**

Input Specifications:	**Expected Results:**

FIX Logon Message (35=A) as defined in Matching Interface User Guide.	**SUCCESS:** 1. Client application should be able to login to MFG after resetting Tag 789 and Tag 34. **FAILURE:** 1. Client application unable to reset sequence numbers correctly and in-turn unable to login to MFG.

SUCCESS SCENARIO


CFG
MFG
FIX Logon (35=A) with Tag 34 and Tag 789 higher than expected
FIX Logout (35=5), 58="Tag 789 higher then expected. Expected X, received Y"
FIX Logon (35=A) with correct Tag 789 and Tag 34 values
FIX Logon (35=A) followed by FIX Tag (35=2)
FIX Tag (35=4) to Gap Fill

5.7 Connection to Matching P2PS via RFA

Test Case 7:	**Purpose:** To verify that the user is able to connect to Matching P2PS via RFA.

Description:

This test case will verify that client application is able to connect to Matching P2PS via RFA and receive event message for successful login.

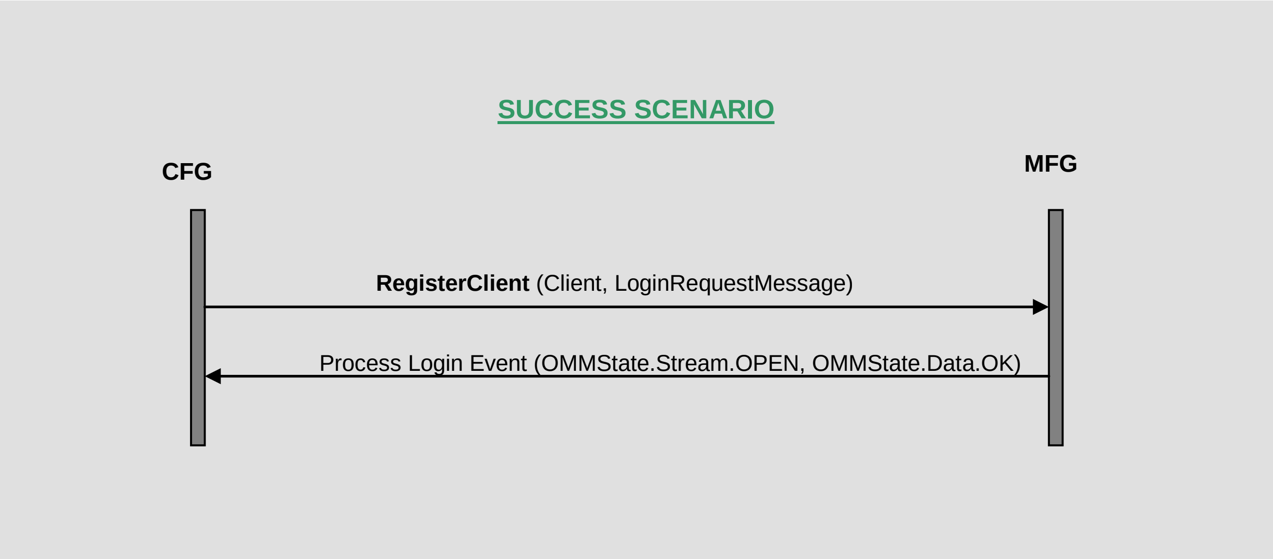
Prerequisites:

- Client should have an established session with Matching FIX Gateway.

Test Steps:

- Client application will establish session with Matching FIX Gateway.
- Client application will acquire the RFA token from the FIX Session user login response (FIX Tag 58) and attempt to login to P2PS Server.
- At successful login, Client application will receive OMMLoginItemEventMsg with OMMState=OPEN and OMMState.data=OK.

Input Specifications:	Expected Results:
• OMMMsg.MsgType=”STREAMING_REQ” • OMMMsg.MsgModelType=”LOGIN”	**SUCCESS:** 1. Client should be able to login to Matching P2PS via RFA with Login Event message having OMMState=OPEN and OMMState.data=OK. **FAILURE:** 1. Client application unable to login to Matching P2PS via RFA. OMMLoginItemEventMsg.getState().getText() populated with Login failure reason in it.


SUCCESS SCENARIO
CFG
MFG
RegisterClient (Client, LoginRequestMessage)
Process Login Event (OMMState.Stream.OPEN, OMMState.Data.OK)

5.8 IOC (YOURS/MINE) order submission

Test Case 8:

Purpose:

To verify that user can submit an IOC (YOURS/MINE) order, and process the fill messages.

Description:

This test case will verify that Matching API user is able to build and send an IOC order correctly for a specific currency, side and price and process the FILL messages.

Prerequisites:

- Client application should have an established session with Matching FIX Gateway.

Test Steps:

Client application will do following steps:

1. Connect to P2PS via RFA using the RFA token from user login response message (35=BF).
2. Subscribe for EUR symbol via MARKET_PRICE or MARKET_BY_PRICE Message Model Type.
3. Parse the received screened price event.
4. Build a FIX New Order Single IOC order (35=D) for EUR BUY (MINE) based on the screened price from RFA, with OrdType=2 and TimeInForce=3.
5. Receive Fill/Partial Fill or Cancelled FIX Execution Report messages (35=8) successively.
6. Receive and process corresponding FIX Trade Capture Report (35=AE) messages.

Input Specification:

FIX New Order Single (35=D) as defined in Matching Interface User Guide.

OrdType (Tag 40=2)

TimeInForce (Tag 59=3)

Currency (Tag 15=EUR)

Symbol (Tag 55=eur/usd)

Side (Tag 54=1)

OrderQty (Tag 38=5)

Expected Results:

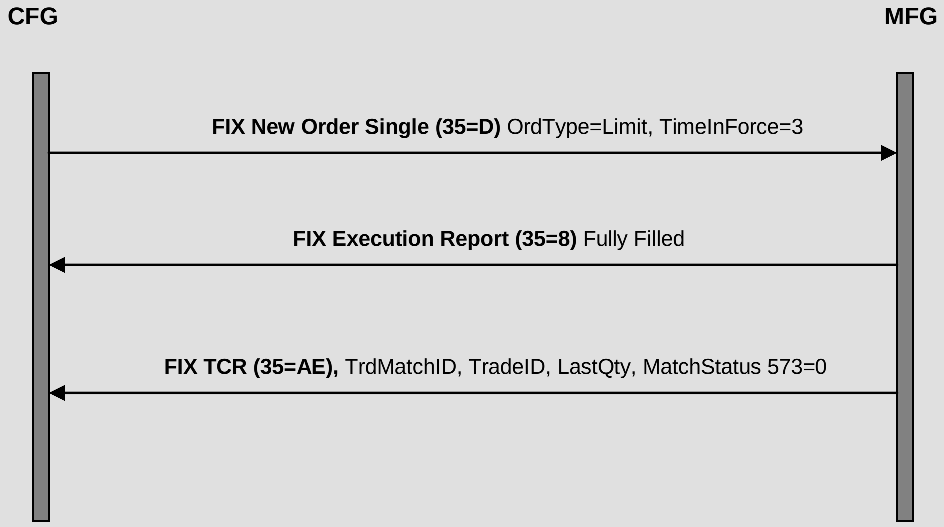
SUCCESS: (below are all success scenarios)

1. Client receives a Filled ExecutionReport (35=8) with ExecType (150=F) and OrdStatus (39=2) followed by a TCR. Client application should verify Tag 573=0/1 for trade confirmation status. TrdMatchID (Tag 880) in TCR should match with what's received in ExecutionReport. TradeID (Tag 1003) in TCR should match the SecondaryExecID (Tag 527) in the ExecutionReport. *A Tag value (573=1) in TradeCaptureReport message indicates that the Match is unconfirmed and Client application should raise alert. In Production system the client can call their Prime Bank or login to Dealing Xtra to obtain an interim DCS to assure that the trade is confirmed with your counterparty.*
2. Client receives multiple partial fill ExecutionReport's with last report being Fully

	Filled (ExecType=F, OrdStatus=2). In fully filled ExecutionReport (35=8) (OrdStatus =2) should have the complete amount filled with OrderQty same as the CumQty, LeavesQty=0, and LastQty indicating the amount filled in the last ExecutionReport notification. 3. Client receives a Partially Filled ExecutionReport (35=8) with ExecType (150=F) and cancelled OrdStatus (39=4) indicating that no more fills will be received. CumQty should be same as LastQty with LeavesQty=0. Any residual Qty is cancelled. 4. Client receives multiple partial fill ExecutionReport's with last ExecutionReport having a cancelled OrdStatus with Partial Fill (ExecType=F, OrdStatus=4). In last ExecutionReport received, CumQty should indicate the total amount filled so far with LastQty indicating the amount filled in the cancelled ExecutionReport notification and LeavesQty=0. 5. Client receives a cancel ExecutionReport (35=8) with ExecType (150=4) and OrdStatus (39=4). **FAILURE:** 1. Message submitted by Matching user has permission or validation error which may result in BusinessMessageReject (j) from Matching FIX Gateway. 2. Various reasons for New Order Single reject are explained in Matching Interface guide Appendix A.

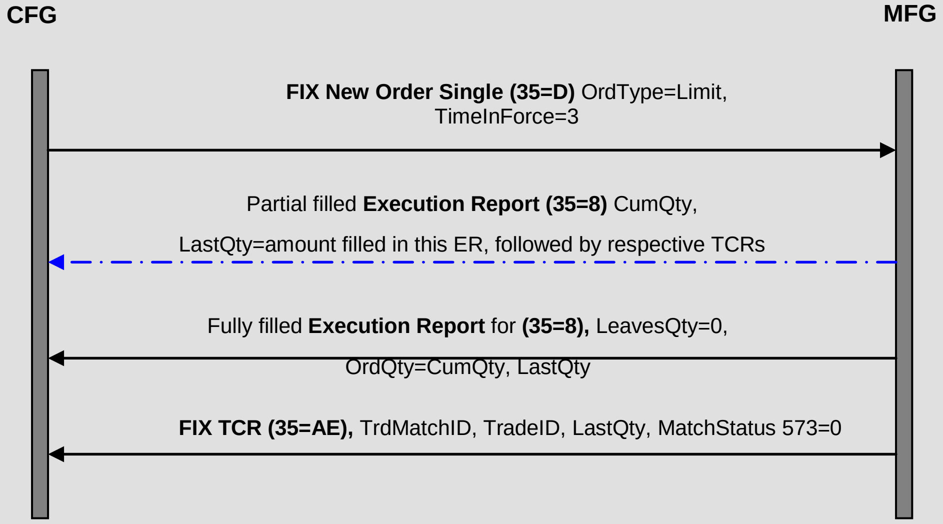
SUCCESS SCENARIO #1

Fully Filled Order Matched against one standing order


CFG
MFG
FIX New Order Single (35=D) OrdType=Limit, TimeInForce=3
FIX Execution Report (35=8) Fully Filled
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

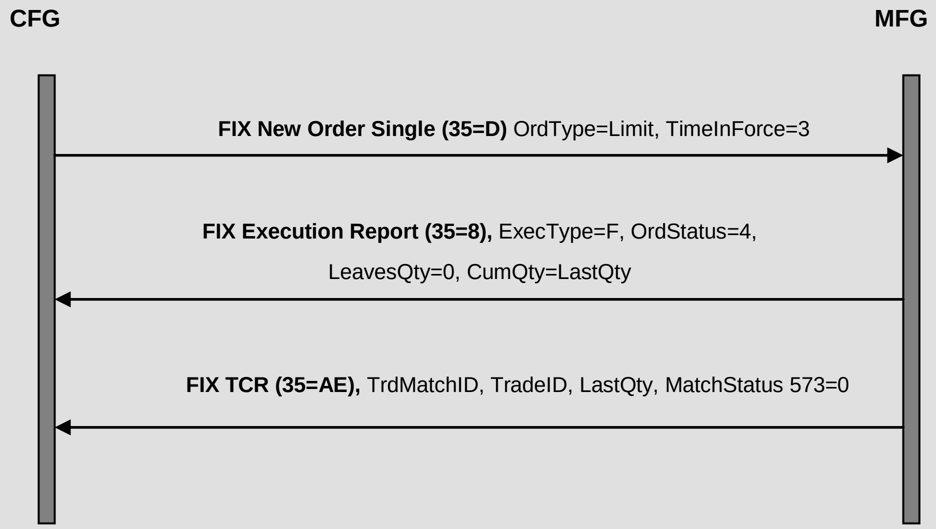
SUCCESS SCENARIO #2

Multi Fills – User can expect to see multiple partial Fill ExecutionReports with order fully filled in last ExecutionReport


CFG
MFG
FIX New Order Single (35=D) OrdType=Limit, TimeInForce=3
Partial filled **Execution Report (35=8)** CumQty,
LastQty=amount filled in this ER, followed by respective TCRs
Fully filled **Execution Report** for **(35=8),** LeavesQty=0, OrdQty=CumQty, LastQty
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

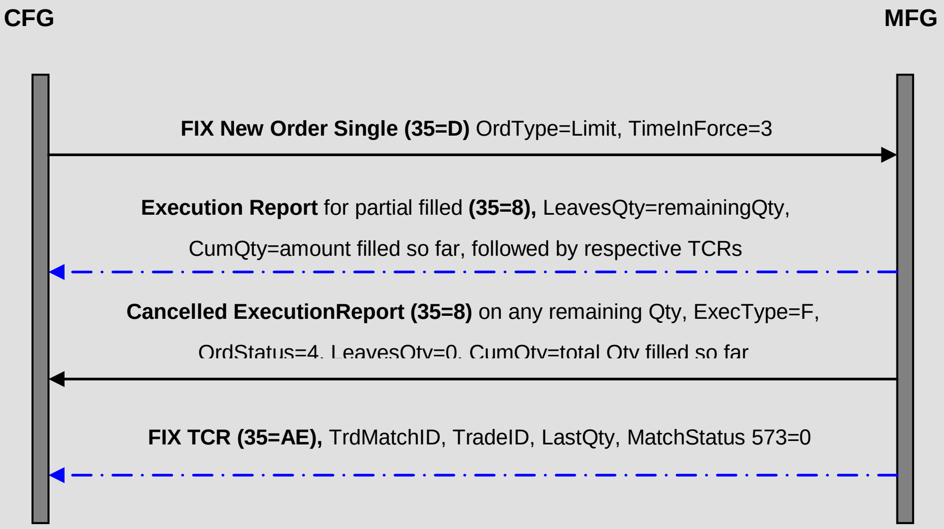
SUCCESS SCENARIO #3

Order Partially filled on entry with one standing order and any remaining quantity is cancelled


CFG
MFG
FIX New Order Single (35=D) OrdType=Limit, TimeInForce=3
FIX Execution Report (35=8), ExecType=F, OrdStatus=4, LeavesQty=0, CumQty=LastQty
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

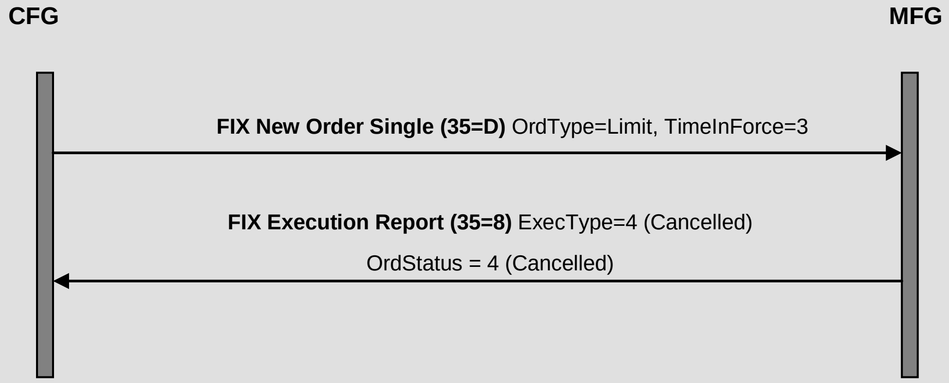
SUCCESS SCENARIO #4

Order multi-filled with any remaining quantity cancelled


CFG
MFG
FIX New Order Single (35=D) OrdType=Limit, TimeInForce=3
Execution Report for partial filled **(35=8),** LeavesQty=remainingQty, CumQty=amount filled so far, followed by respective TCRs
Cancelled ExecutionReport (35=8) on any remaining Qty, ExecType=F, OrdStatus=4, LeavesQty=0, CumQty=total Qty filled so far
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

SUCCESS SCENARIO #5

Order cancelled due to Price removal from Host


CFG
MFG
FIX New Order Single (35=D) OrdType=Limit, TimeInForce=3
FIX Execution Report (35=8) ExecType=4 (Cancelled)
OrdStatus = 4 (Cancelled)

5.9 Standard order (BID/OFFER) submission

Test Case 9:	**Purpose:** To verify that the Matching API user is able to send a standard order (BID/OFFER) and process the received fill messages.

Description:

This test case will verify that Matching API user is able to send a standard order and process the FILL Execution Reports received from Matching FIX Gateway.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

Test Steps:

Client application will do following steps:

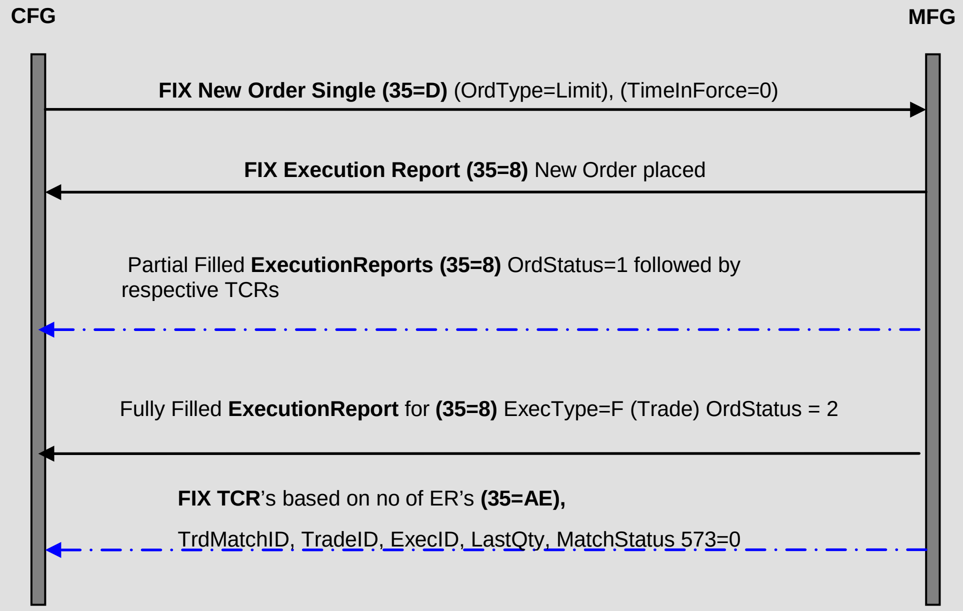
1. Build a FIX New Order Single (35=D) for GBP SELL with OrdType=2 and TimeInForce=0.
2. Receive an Execution Report for New order submitted with OrderID (Tag 37), ExecType ‘New’ (Tag 150=0) and OrdStatus ‘New’ (Tag 39=0). This indicates Order is placed in book.
3. Receive Fill/Partial Fill or Cancelled FIX Execution Report ((35=8) messages based on the no. of matches against this order.
4. Process received FIX Trade Capture Report (35=AE) messages.

Input Specification:	**Expected Results:**
FIX New Order Single (35=D) as defined in Matching Interface User Guide. OrdType (Tag 40=2) TimeInForce (Tag 59=0) Side (Tag 54=2) Currency (Tag 15=GBP) Symbol (Tag 55=gbp/usd) Price (Tag 44) OrderQty (Tag 38=20) to fill an amount of 20	**SUCCESS: (below are all success scenarios)** 1. Client receives multiple Partial Fill ExecutionReport (35=8) followed by a Fully Filled ExecutionReport with ExecType (150=F) and OrdStatus (39=2). Client application should also receive a TCR (35=AE) corresponding to every Filled/Partial Filled ExecutionReport with Tag 573=0/1. TrdMatchID (Tag 880) in TCR should match with what’s received thru ExecutionReport. TradeID (Tag 1003) in TCR should match the SecondaryExecID (Tag 527) in the ExecutionReport. 2. Client receives multiple partial Fill ExecutionReport’s with OrdStatus=1, followed by an unsolicited cancel ExecutionReport (OrdStatus=4) due to user disconnect or abrupt end of trading session. Process any received TCRs. 3. Client receives one Fill ExecutionReport

	(OrdStatus=2) indicating that the order was fully filled upon hitting the book. Process any received TCRs. 4. Client receives ExecutionReport with partial fill (OrdStatus=1) indicating the remaining quantity still staying in the book for further possible matches. 5. Client receives a reject ExecutionReport (OrdStatus=8, ExecType=8). **FAILURE:** 1. Message submitted by Matching user has permission error which may result in BusinessMessageReject (j) from Matching FIX Gateway. 2. Various reasons for New Order Single reject are explained in Matching Interface guide Appendix A.

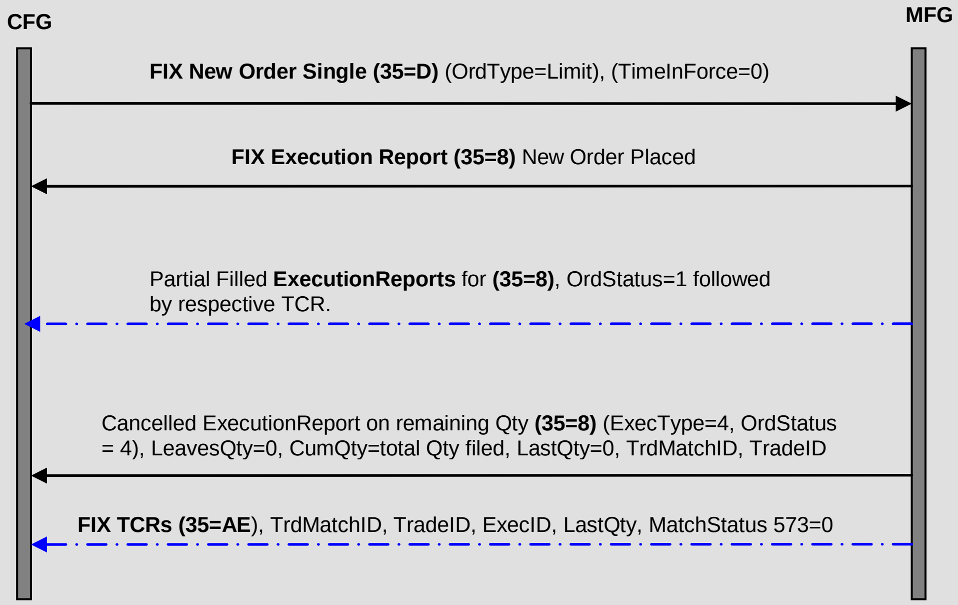
SUCCESS SCENARIO #1

User receives multiple partial fills followed by a Fully Filled Execution Report


CFG
MFG
FIX New Order Single (35=D) (OrdType=Limit), (TimeInForce=0)
FIX Execution Report (35=8) New Order placed
Partial Filled **ExecutionReports (35=8)** OrdStatus=1 followed by respective TCRs
Fully Filled **ExecutionReport** for **(35=8)** ExecType=F (Trade) OrdStatus = 2
FIX TCR's based on no of ER's **(35=AE),**
TrdMatchID, TradeID, ExecID, LastQty, MatchStatus 573=0

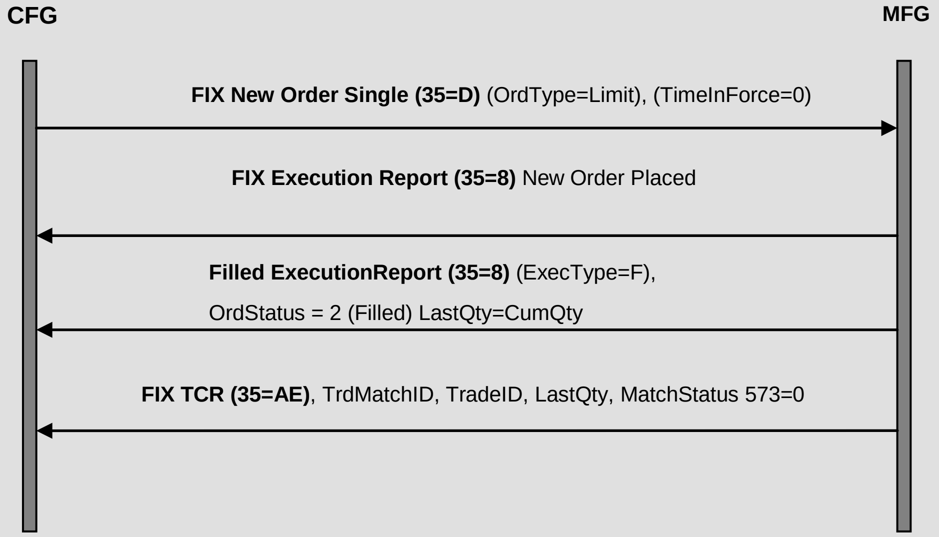
SUCCESS SCENARIO #2

User receives multiple partial fills followed by an unsolicited cancel


CFG
MFG
FIX New Order Single (35=D) (OrdType=Limit), (TimeInForce=0)
FIX Execution Report (35=8) New Order Placed
Partial Filled **ExecutionReports** for **(35=8)**, OrdStatus=1 followed by respective TCR.
Cancelled ExecutionReport on remaining Qty **(35=8)** (ExecType=4, OrdStatus = 4), LeavesQty=0, CumQty=total Qty filed, LastQty=0, TrdMatchID, TradeID
FIX TCRs (35=AE), TrdMatchID, TradeID, ExecID, LastQty, MatchStatus 573=0

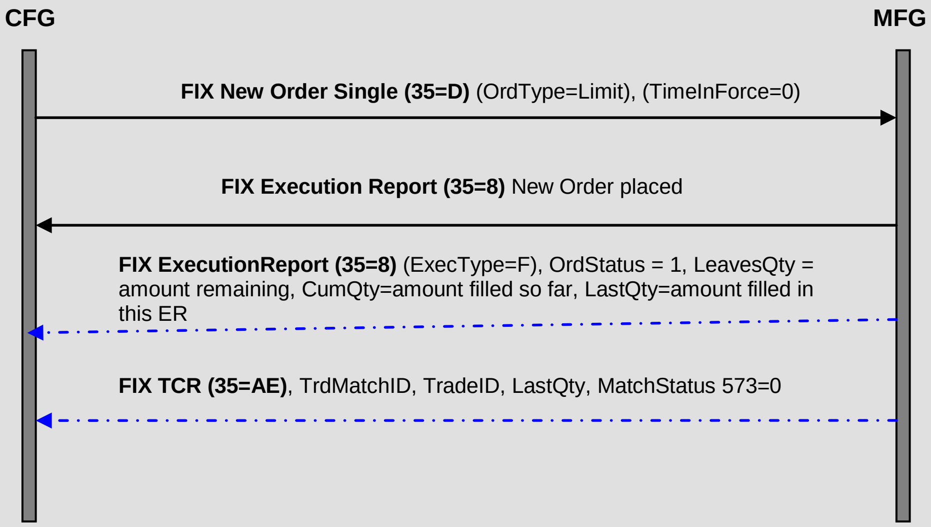
SUCCESS SCENARIO #3

Order fully filled upon hitting the book


CFG
MFG
FIX New Order Single (35=D) (OrdType=Limit), (TimeInForce=0)
FIX Execution Report (35=8) New Order Placed
Filled ExecutionReport (35=8) (ExecType=F),
OrdStatus = 2 (Filled) LastQty=CumQty
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

SUCCESS SCENARIO #4

Order partially filled and remains at host for further fills



FIX New Order Single (35=D) (OrdType=Limit), (TimeInForce=0)

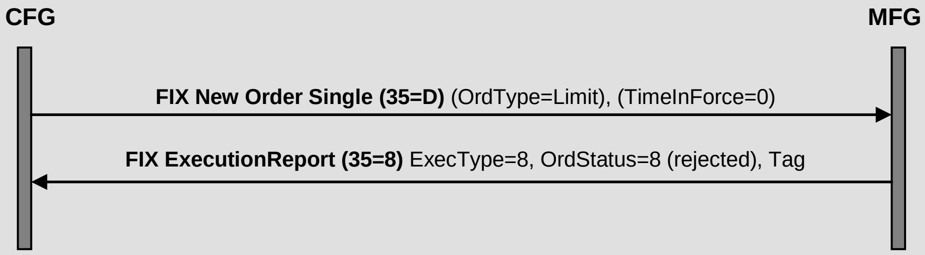
FIX Execution Report (35=8) New Order placed

FIX ExecutionReport (35=8) (ExecType=F), OrdStatus = 1, LeavesQty = amount remaining, CumQty=amount filled so far, LastQty=amount filled in this ER

FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

SUCCESS SCENARIO #5

Order is cancelled by host as it enters



FIX New Order Single (35=D) (OrdType=Limit), (TimeInForce=0)

FIX ExecutionReport (35=8) ExecType=8, OrdStatus=8 (rejected), Tag

5.10 Standard order with More Quantity

Test Case 10:

Purpose:

To verify that the user is able to send a standard order with More Quantity and process any received fill messages.

Description:

This test case will verify that Matching API user is able to send a standard order with More Quantity (Iceberg order) and process any filled execution reports received from Matching FIX Gateway.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login from their MAPI on-boarding team so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

Test Steps:

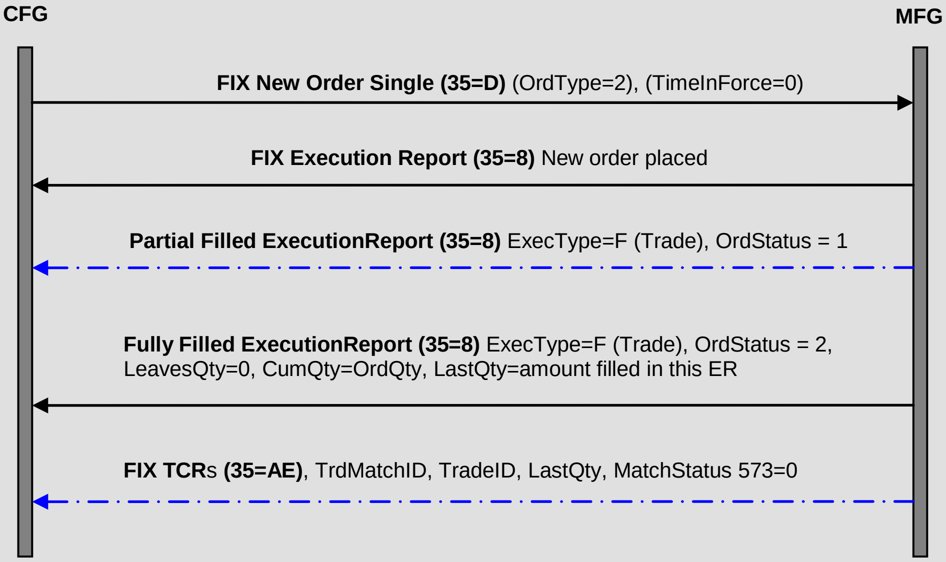
1. Build a FIX New Order Single message (35=D) for EUR BUY order with OrdType=2 and TimeInForce=0.
2. Provide OrderQty (Tag 38=20) with DisplayQty (Tag 1138=10) to indicate a More Quantity of 10.
3. Receive a FIX Execution Report for New order submitted with OrderID (Tag 37), ExecType ‘New’ (Tag 150=0) and OrdStatus ‘New’ (Tag 39=0). This indicates Order is placed in book.
4. Receive Fill/Partial Fill or Cancelled FIX Execution Reports (35=8).
5. Process received FIX Trade Capture Report (35=AE) messages.

Input Specification:	**Expected Results:**
FIX New Order Single (35=D) as defined in Matching Interface User Guide. OrdType (Tag 40=2) TimeInForce (Tag 59=0) OrderQty (Tag 38=20) Currency (Tag 15=EUR) Symbol (Tag 55=eur/usd) DisplayQty (Tag 1138=10)	**SUCCESS: (below are all success scenarios)** 1. Client gets all his Order quantity filled. Client possibly receives multiple Partial Fill ExecutionReports (35=8) followed by a Fully Filled ExecutionReport with ExecType (150=F) and OrdStatus (39=2). Client application should also receive a TCR (35=AE) corresponding to every Filled/Partially Filled ExecutionReport with Tag 573=0/1. TrdMatchID (Tag 880) in TCR should match with what’s received thru ExecutionReport. TradeID (Tag 1003) in TCR should match the SecondaryExecID (Tag 527) in the ExecutionReport. 2. In multi-way matches, client may get either all of the OrderQty filled or little more then the DisplayQty filled, with any remaining MoreQuantity cancelled. In last ExecutionReport report received, a client should expect LeavesQty=0, LastQty

	indicating amount filled in this ExecutionReport along with CumQty indicating total amount filled so far. **FAILURE:** 1. Various reasons for New Order Single reject are explained in Matching Interface User guide Appendix A.

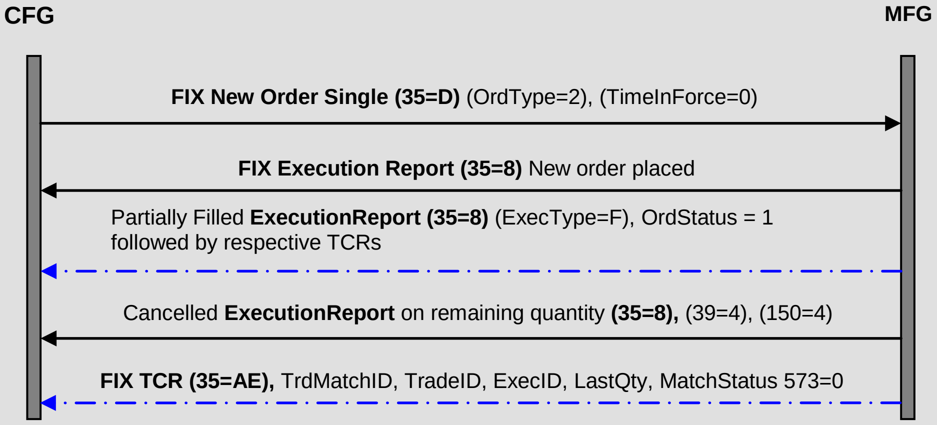
SUCCESS SCENARIO #1

Client gets all his OrderQty filled including the More Quantity


CFG
MFG
FIX New Order Single (35=D) (OrdType=2), (TimeInForce=0)
FIX Execution Report (35=8) New order placed
Partial Filled ExecutionReport (35=8) ExecType=F (Trade), OrdStatus = 1
Fully Filled ExecutionReport (35=8) ExecType=F (Trade), OrdStatus = 2, LeavesQty=0, CumQty=OrdQty, LastQty=amount filled in this ER
FIX TCRs **(35=AE)**, TrdMatchID, TradeID, LastQty, MatchStatus 573=0

SUCCESS SCENARIO #2

Client gets partial fills followed by a cancellation on remaining quantity


CFG
MFG
FIX New Order Single (35=D) (OrdType=2), (TimeInForce=0)
FIX Execution Report (35=8) New order placed
Partially Filled **ExecutionReport (35=8)** (ExecType=F), OrdStatus = 1 followed by respective TCRs
Cancelled **ExecutionReport** on remaining quantity **(35=8),** (39=4), (150=4)
FIX TCR (35=AE), TrdMatchID, TradeID, ExecID, LastQty, MatchStatus 573=0

5.11 IDEAL Spot Order submission

Test Case 11:	**Purpose:** To verify that the user can submit an IDEAL Spot Order and process received matches.

Description:

This test case will verify that Matching API user is able to send an IDEAL Spot Order message and process the received Match notifications.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

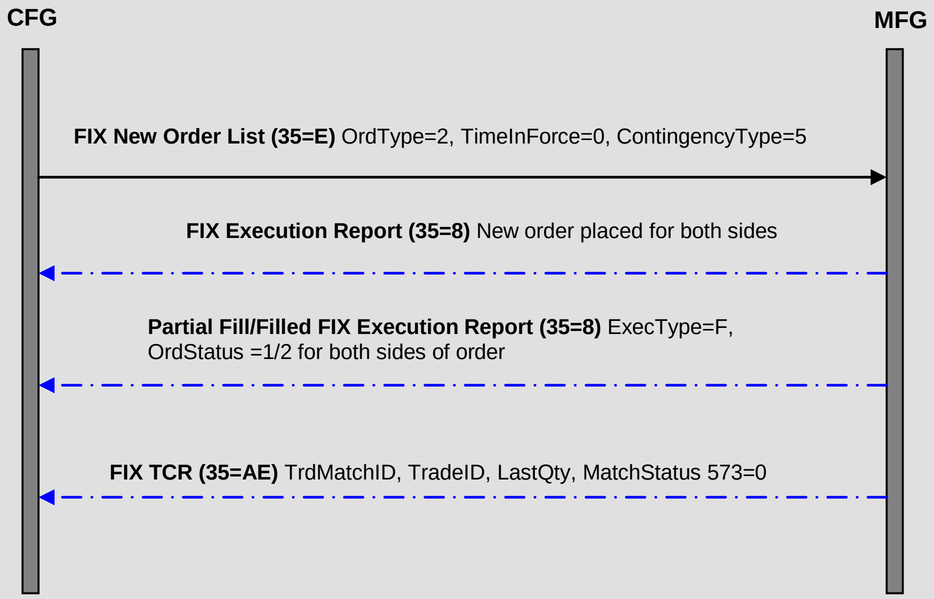
Test Steps:

1. Build an IDEAL order, FIX New Order List (35=E), with both bid and offer price populated for the same currency with Contingency Type (Tag 1385 = 5).
2. Matching Host will assign a unique OrderID (Tag 37) to each side of the order in the new Execution Report message (35=8) with ExecType (150=0). Client application will receive 2 Execution Reports for each side of the List Order to indicate 2 orders placed at Host.
3. MFG will send (Partial-Fill/Filled/Cancelled) ExecutionReports for each side (BID and OFFER) of the order as and when match happens.
4. Process FIX Trade Capture Report (35=AE) corresponding to every Filled/Partial Filled ExecutionReport.

Input Specification:	**Expected Results:**
FIX New Order List Message (35=E) as defined in Matching Interface User Guide. ContingencyType (Tag 1385=5) Account (Tag 1 = ACCT) - *Account is mandatory tag* TimeInForce (Tag 59 = 0) - *Absence of TimeInForce will indicate a value 0, DAY trade* OrdQty (Tag 38) = 10 Symbol (Tag 55=eur/usd) should be same on both sides Currency (Tag 15=EUR) should be same on both sides	**SUCCESS:** **1.** Matching API user will receive FIX ExecutionReports (Filled/Partial Filled) on orders as they get matched at Host. Client application should also receive FIX Trade Capture Reports (35=AE) corresponding to every Filled/Partial Filled ExecutionReport with Tag 573=0/1. TrdMatchID (Tag 880) in TCR should match with what's received in ExecutionReport. TradeID (Tag 1003) in TCR should match the SecondaryExecID (Tag 527) in the ExecutionReport. **FAILURE:** 1. Message validation fails at Matching FIX Gateway and may result in user receiving a FIX ListStatus (35=N) message with (431=7), (1386=11 or 99), (444=Error text).

	2. Various reasons for New Order List reject are explained in Matching Interface User guide Appendix A.

SUCCESS SCENARIO


CFG
MFG
FIX New Order List (35=E) OrdType=2, TimeInForce=0, ContingencyType=5
FIX Execution Report (35=8) New order placed for both sides
Partial Fill/Filled FIX Execution Report (35=8) ExecType=F, OrdStatus =1/2 for both sides of order
FIX TCR (35=AE) TrdMatchID, TradeID, LastQty, MatchStatus 573=0

5.12 IDEAL Spot order with OCO (One Cancel Other)

Test Case 12:	**Purpose:** To verify that the user can submit an IDEAL Spot order with OCO (One Cancel Other) condition and verify receiving matches on submitted order.

Description:

This test case will verify that Matching API user is able to send an IDEAL OCO order where if one side is fully filled then other side is cancelled by Matching Host.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

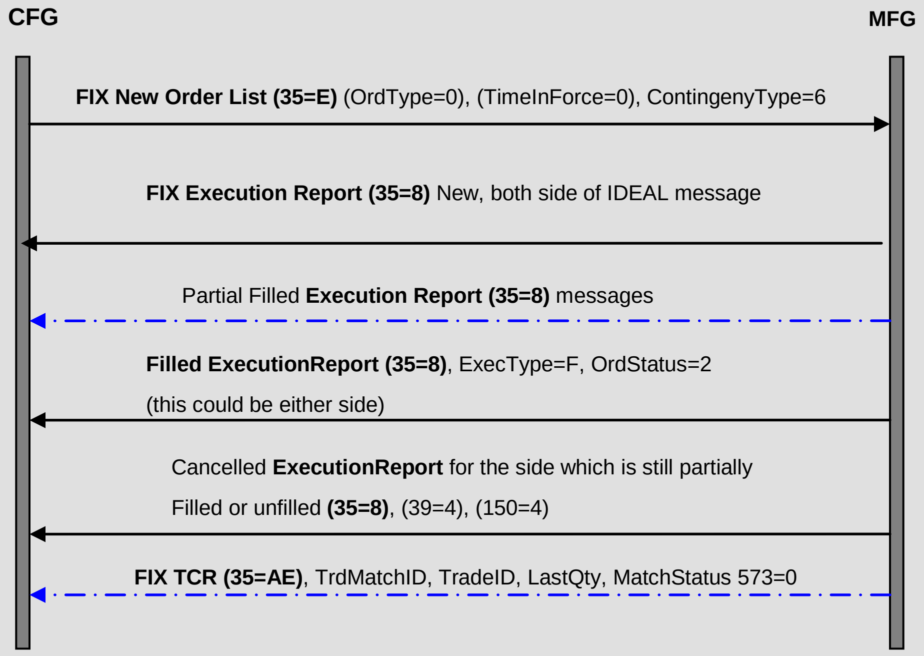
Test Steps:

1. Build an IDEAL order, FIX New Order List (35=E), with both bid and offer price populated for the same currency with Contingency Type (Tag 1385 = 6). This will ensure that once one side of the order is fully filled then other side is automatically cancelled by Matching Host.
2. Client will receive FIX Execution Report (35=8) with ExecType (150=0) and OrderID as assigned by Host for each side of the List Order.
3. Receive Partial Fill/Filled/Cancelled FIX ExecutionReports for each side (BID and OFFER) of the order once the order is placed at Host.
4. Receive FIX Trade Capture Report (35=AE) corresponding to every Filled/Partial Filled ExecutionReport.
5. Matching user will receive a cancellation on the other side of the List order when one side is fully traded.

Input Specification:	**Expected Results:**
New Order List Message (35=E) as defined in Matching Interface User Guide. Contingency Type (Tag 1385=6) Account (Tag 1 = ACCT) - *Account is mandatory tag* TimeInForce (Tag 59 = 0) - *Absence of TimeInForce will indicate a value 0, DAY trade* OrdQty (Tag 38) = 10 Symbol (Tag 55=eur/usd) should be same on both sides Currency (Tag 15=EUR) should be same on both sides	**SUCCESS:** 1. Matching API user will receive Filled/Partial Filled Execution Reports on orders as they get matched at Host. Client application should also receive a FIX Trade Capture Report (35=AE) corresponding to every Filled/Partial Filled Execution Report with Tag 573=0/1. TrdMatchID (Tag 880) in TCR should match with what’s received thru Execution Report. TradeID (Tag 1003) should match the SecondaryExecID (Tag 527) in the Execution Report. User will receive a cancel Execution Report on the other side of the order when one side is fully traded.

	FAILURE: 1. Matching user may receive a FIX ListStatus (35=N) message with (431=7), (1386=11 or 99), (444=Error text) if the order fails validation. 2. Various reasons for New Order List reject are explained in Matching Interface User guide Appendix A.

SUCCESS SCENARIO


CFG
MFG
FIX New Order List (35=E) (OrdType=0), (TimeInForce=0), ContingenyType=6
FIX Execution Report (35=8) New, both side of IDEAL message
Partial Filled **Execution Report (35=8)** messages
Filled ExecutionReport (35=8), ExecType=F, OrdStatus=2
(this could be either side)
Cancelled **ExecutionReport** for the side which is still partially
Filled or unfilled **(35=8)**, (39=4), (150=4)
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

5.13 Standard order with ‘Jobbing Mode’ enabled

Test Case 13:	Purpose: To verify that the user can submit a New Order Single message (35=D) with ‘Jobbing Mode’ enabled and verify receiving cancellations on all orders for a given symbol and side.

Description:

This test case will verify that Matching API user is able to send a FIX New Order Single message (35=D) with ‘Jobbing Mode’ enabled which will cancel any previously submitted order for a given symbol and side.

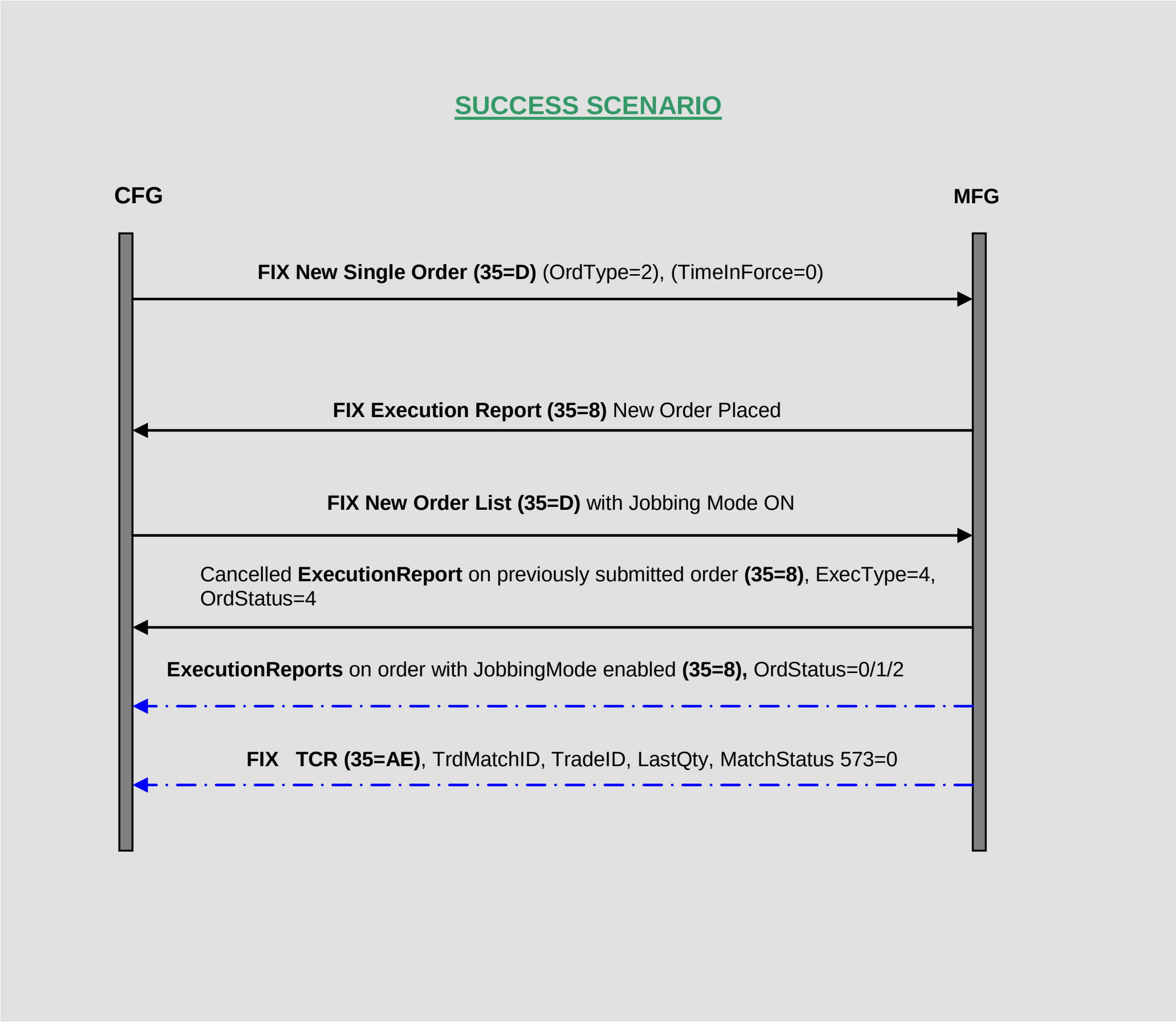
Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

Test Steps:

1. Build a FIX New Order Single (35=D) message with BID price populated for the same symbol and side.
2. Build another FIX New Order Single (35=D) message with a new BID price populated for the same symbol and side with TriggerType=5, TriggerAction=3 and TriggerScope=4 to enable ‘Jobbing Mode’.
3. Receive Cancelled Execution Reports on previously submitted order.
4. Receive Execution Report New (ExecType=0) for ‘Jobbing Mode’ enabled order.
5. Process any Execution Reports received on Partial/Fully Filled messages.
6. Process received Trade Capture Report (35=AE) messages.

Input Specification:	Expected Results:
New Order List Message (35=D) as defined in Matching Interface User Guide. TriggerType (Tag 1100=5), TriggerAction (1101=3) TriggerScope(1628=4) Symbol (55=eur/usd) Currency (15=EUR) Side (54=1) OrderQty (Tag 38) = 10	**SUCCESS:** 1. Once a new order is submitted with JobbingMode enabled, Matching API User will receive Cancelled ExecutionReport on previously submitted order with same symbol and side. Matching API User will receive TradeCaptureReports on any filled trades. **FAILURE:** 1. If Message fails validation at Matching FIX Gateway then this may result in user receiving a FIX ListStatus (35=N) message with (431=7), (1386=11 or 99), (444=Error text). 2. Various reasons for Order Reject are explained in Matching Interface user guide Appendix A.


SUCCESS SCENARIO
CFG
MFG
FIX New Single Order (35=D) (OrdType=2), (TimeInForce=0)
FIX Execution Report (35=8) New Order Placed
FIX New Order List (35=D) with Jobbing Mode ON
Cancelled ExecutionReport on previously submitted order (35=8), ExecType=4, OrdStatus=4
ExecutionReports on order with JobbingMode enabled (35=8), OrdStatus=0/1/2
FIX TCR (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

5.14 IDEAL order with ‘AOR’ enabled

Test Case 14:	**Purpose:** To verify that the user can submit an IDEAL order with ‘AOR’ enabled and verify receiving cancellation on originally submitted orders in given symbol, price and side.

Description:

This test case will verify that Matching API user is able to send an IDEAL Spot order with ‘AOR’ enabled which will cancel any previously submitted orders in given symbol, price and side. With ‘AOR’ disabled a client can enter multiple orders at same price level.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

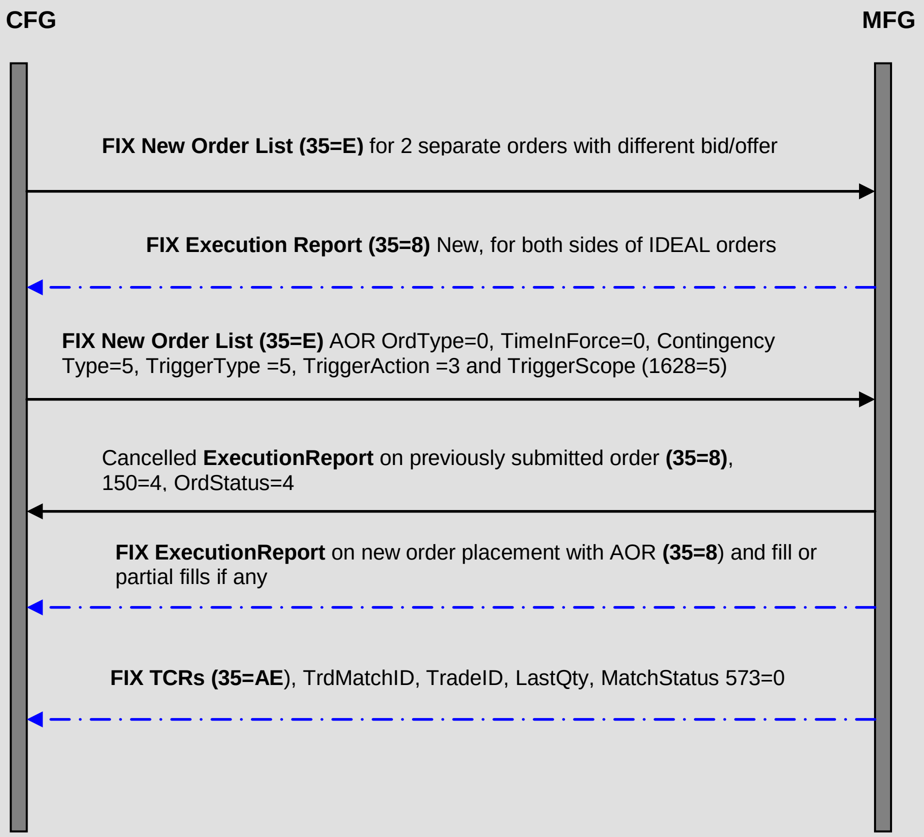
Test Steps:

1. Build a New List Order (35=E) message with both bid and offer price populated for the same currency (eur/usd) with Contingency Type (Tag 1385 = 5).
2. Build another List Order (35=E) message with a different bid and offer price populated for the same currency (eur/usd).
3. Build another New Order List (35=E) message for same currency (eur/usd) and price as in step 1 above with Contingency Type (Tag 1385 = 5). The order should have same price, side and symbol as the order submitted in step 1 above with TriggerType=5, TriggerAction=3 and TriggerScope=5 to enable ‘AOR’.
4. Receive cancelled Execution Reports on both sides of the order submitted in step 1.
5. Receive FIX Execution Reports on new order placement for the order sent with AOR enabled (in step3).
6. Process any Partial Fills/Filled FIX Execution Reports received on messages.
7. Process received Trade Capture Reports (35=AE).

Input Specification:	**Expected Results:**
New Order List Message (35=E) as defined in Matching Interface User Guide. Contingency Type (1385=5) TriggerType (Tag 1100=5) TriggerAction (1101=3) TriggerScope(1628=5) Price (44= same as the order that needs replacing) Side (54=1) Symbol (55=eur/usd)	**SUCCESS:** 1. Matching API user will receive Cancelled FIX ExecutionReport on previously submitted order with same symbol, price and side. Client application should also receive FIX Trade Capture Reports (35=AE) corresponding to every Filled/Partial Filled ExecutionReport with Tag 573=0/1. There is a possibility for client to have one side of the order fully filled before the AOR enabled order is sent in which case only the side that is not filled will be cancelled.

Currency(15=EUR) OrdQty (Tag 38) = 10	**FAILURE:** 1. If Message fails validation at Matching FIX Gateway then this may result in user receiving a ListStatus (35=N) message with (431=7), (1386=11 or 99), (444=Error text). 2. Various reasons for New Order List reject are explained in Matching Interface guide Appendix A.

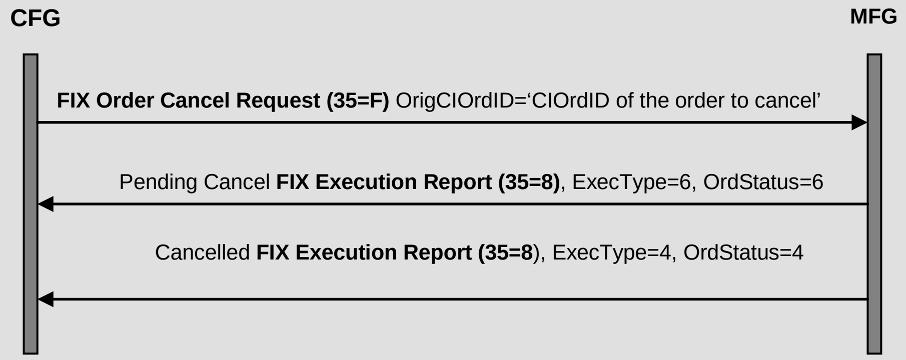
SUCCESS SCENARIO


CFG
MFG
FIX New Order List (35=E) for 2 separate orders with different bid/offer
FIX Execution Report (35=8) New, for both sides of IDEAL orders
FIX New Order List (35=E) AOR OrdType=0, TimeInForce=0, Contingency Type=5, TriggerType =5, TriggerAction =3 and TriggerScope (1628=5)
Cancelled **ExecutionReport** on previously submitted order **(35=8)**, 150=4, OrdStatus=4
FIX ExecutionReport on new order placement with AOR **(35=8**) and fill or partial fills if any
FIX TCRs (35=AE), TrdMatchID, TradeID, LastQty, MatchStatus 573=0

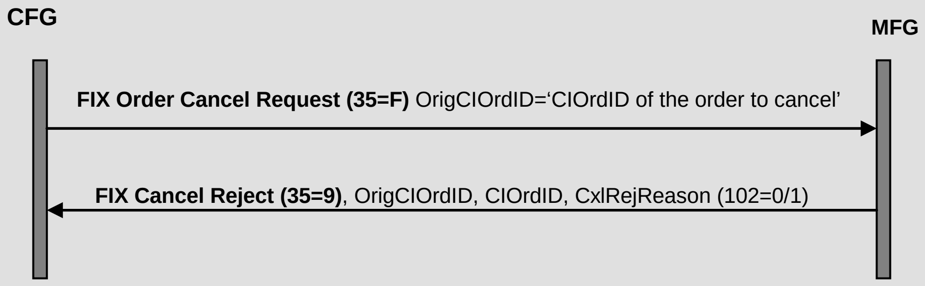
5.15 Cancel Request message to cancel a specific order

Test Case 15:	**Purpose:** To verify that the Matching API user can submit a Cancel Request Message to cancel a previously submitted message by providing ClOrdID.
Description: This test case will verify that the Matching API user can submit a Cancel Request Message to cancel a previously submitted message by providing ClOrdID.	
Prerequisites: • Client should have an established session with Matching FIX Gateway.	
Test Steps: 1. Send a EURUSD FIX New Order Single (35=D) message. 2. Build and send a FIX Cancel Request (35=F) with OrigClOrdID (Tag 41) = "ClOrdID of the message to cancel". 3. Receive and process Cancel Pending and cancelled ExecutionReport (35=8) OR FIX Cancel Reject (35=9) in case order gets filled before the cancel request was received by Host.	
Input Specification:	**Expected Results:**
FIX Cancel Request Message (35=F) as defined in Matching Interface User Guide. OrigClOrdID (Tag 41="ClOrdID of submitted SPOT order")	**SUCCESS: (below are all success scenarios)** 1. Client application will receive Cancel ExecutionReport (35=8) with ExecType =6 (Pending Cancel), OrdStatus=6. Followed by Cancelled ExecutionReport (35=8) with ExecType=4(Cancelled), OrdStatus=4(Cancelled). 2. Optionally, an ExecutionReport can be received on a previously submitted order under which circumstance if the order is fully filled then a CancelReject (35=9) can be received by client app, OR if the order is partially filled then Cancel Execution will be received. If the Cancel request is rejected due to cancel request being too late then CxlRejReason (102=0) will be received by client otherwise a CxlRejReason 102=1 (unknown) will be received. **FAILURE:** 1. Various reasons for CancelReject are defined in Matching Interface user guide Appendix A.

SUCCESS SCENARIO #1


CFG
MFG
FIX Order Cancel Request (35=F) OrigClOrdID='ClOrdID of the order to cancel'
Pending Cancel **FIX Execution Report (35=8)**, ExecType=6, OrdStatus=6
Cancelled **FIX Execution Report (35=8**), ExecType=4, OrdStatus=4

SUCCESS SCENARIO #2


CFG
MFG
FIX Order Cancel Request (35=F) OrigClOrdID='ClOrdID of the order to cancel'
FIX Cancel Reject (35=9), OrigClOrdID, ClOrdID, CxlRejReason (102=0/1)

5.16 Mass Action Cancel Request to cancel previously submitted messages for a specific SecurityType and Symbol

Test Case 16:

Purpose:

To verify that the Matching API user can submit a Mass Action Cancel Request (35=CA) to cancel previously submitted messages for a specific SecurityType and Symbol.

Description:

This test case will verify that the Matching API user can submit Mass Cancel Request Message (35=CA) to cancel previously submitted messages on a specific Security and Symbol.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.

Test Steps:

1. Send three FIX New Order Single (35=D) messages with eur/usd and two with gbp/usd.
2. Build and send Mass Cancel Requests (35=CA) with Tag 1373=3, Tag 1374=1 and Tag 55=eur/usd.
3. Receive and process Cancel Pending and cancelled ExecutionReports (35=8) on all eur/usd orders.

Input Specification:

FIX MassActionRequest (35=CA) as defined in Matching Interface User Guide.

MassActionType (1373=3)

MassActionScope (1374=1)

Symbol (55=eur/usd)

SecurityType (167=FXSPOT)

Expected Results:

SUCCESS: (below are all success scenarios)

1. User will receive FIX MassActionReport (35=BZ) ACK status (1375=1) followed by cancelled FIX ExecutionReports. Once all the cancelled ExecutionReports have been delivered, user will receive FIX MassActionReport (35=BZ) to acknowledge completion of the report (1375=1) and indicate TotalAffectedOrders (Tag 533) which should be 3 as per the test.
2. User may optionally receive a FIX Mass Action Reject Report (BZ) with 1375=0 and reason (1376) = 99 *(99= other -- Used for when there is nothing to cancel)*.
3. There is also a possibility that orders were in cache at the time of the request but were filled prior to the Matching Host receiving the cancel request. This will result in FIX MassActionReport (BZ) Ack (1375=1) followed by FIX MassActionReport (BZ) complete status (1375=1) with TotalAffectedOrders=0.

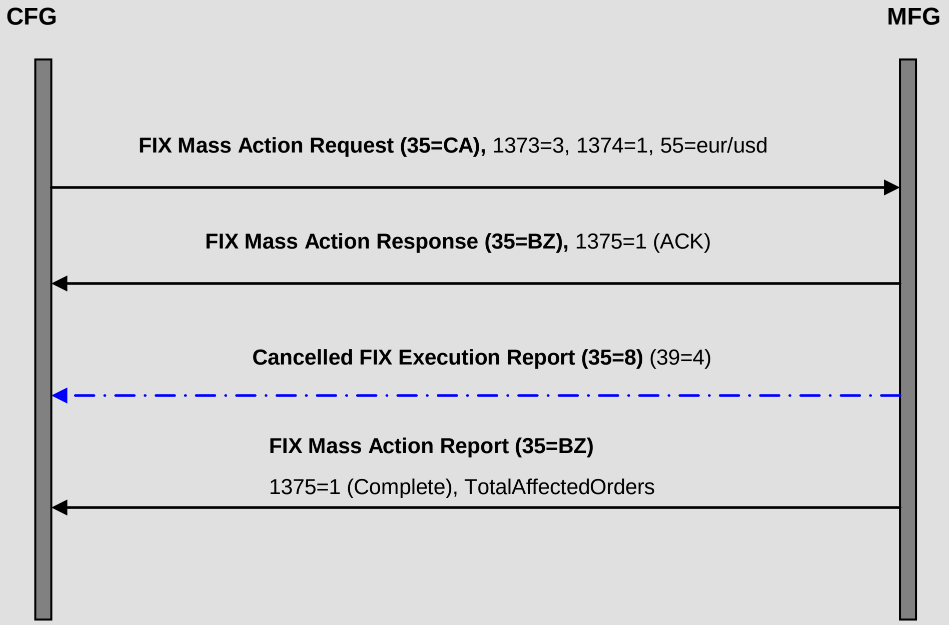
FAILURE:

1. User may optionally receive a FIX Mass

	Action Reject Report (BZ) with reason (1376) = 1 – Invalid or unknown security, or 5 – Invalid or unknown SecurityType. 2. Various reasons for Mass Action Cancel are defined in Matching Interface user guide Appendix A.

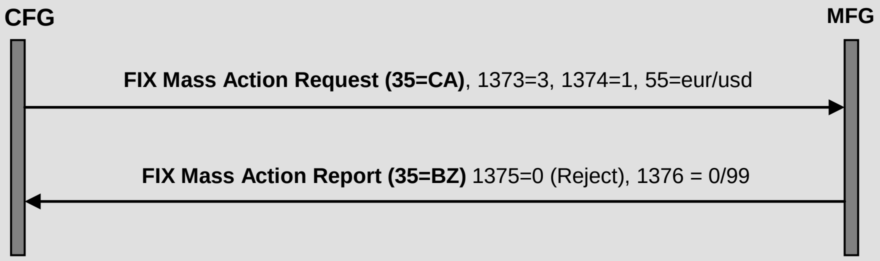
SUCCESS SCENARIO #1

User will receive Cancelled ExecutionReports as a result of MassActionRequest


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=1, 55=eur/usd
FIX Mass Action Response (35=BZ), 1375=1 (ACK)
Cancelled FIX Execution Report (35=8) (39=4)
FIX Mass Action Report (35=BZ)
1375=1 (Complete), TotalAffectedOrders

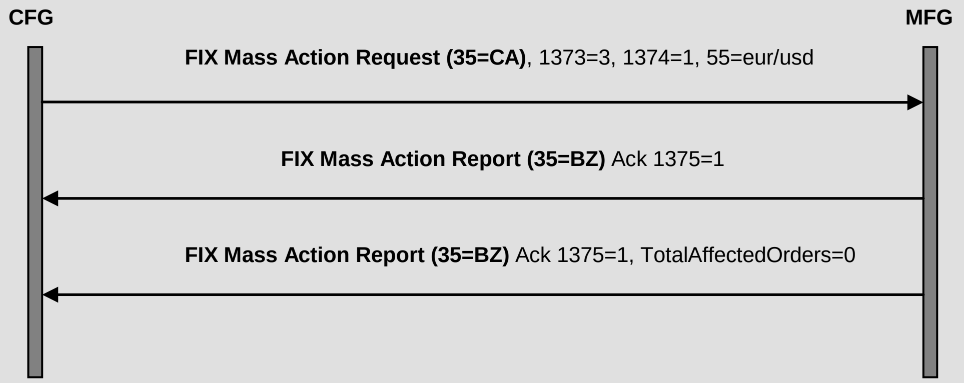
SUCCESS SCENARIO #2

User may receive MassActionReport with rejection if there is nothing to cancel


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=1, 55=eur/usd
FIX Mass Action Report (35=BZ) 1375=0 (Reject), 1376 = 0/99

SUCCESS SCENARIO #3

User may receive a MassActionReport with 0 affected orders if the orders were cancelled or filled before the MassActionReport was received by Matching Host


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=1, 55=eur/usd
FIX Mass Action Report (35=BZ) Ack 1375=1
FIX Mass Action Report (35=BZ) Ack 1375=1, TotalAffectedOrders=0

5.17 Mass Action Cancel Request to cancel previously submitted messages for a specific SecurityType

Test Case 17:	**Purpose:** To verify that the Matching API user can submit a Mass Action Cancel Request Message to cancel previously submitted messages for a specific SecurityType.

Description:

This test case will verify that the Matching API user can submit a Mass Action Cancel Request Message to cancel previously submitted messages on a specific SecurityType.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Client should have access to a negotiation user so he can create and enter a SWAP order. Also the negotiation user has to be logged in at the time of test.
- Users who may not be doing both spot and forward workflow can skip this test case.

Test Steps:

1. Send three EURUSD Spot Orders (35=D) and two SWAP (FXSWAP) Orders in GBPUSD.
2. Receive 5 ExecutionReports with OrdStatus=0(New) on Order placement.
3. Build and send FIX Mass Cancel Requests (35=CA) with (Tag 1373=3, Tag 1374=5, Tag 167=FXSPOT) this will cancel only the Spot orders.
4. Receive and process Cancel Pending ExecutionReports (35=8) on all three SPOT orders.

Input Specification:	**Expected Results:**

FIX MassActionRequest (35=CA) as defined in Matching Interface User Guide.
MassActionType (1373=3)
MassActionScope(1374=5)
SecurityType(167=FXSPOT)
	SUCCESS: (below are all success scenarios)
	1. User should receive FIX Mass Action Report (35=BZ) ACK status (1375=1) followed by Execution Reports on all the cancelled Spot orders. Once all the Execution Reports have been delivered, user will receive FIX Mass Action Report (35=BZ) with complete status (1375=1) and TotalAffectedOrders for ex. Tag 553= 3 in this case.
	2. User may optionally receive a FIX Mass Action Reject Report (BZ) with reason (1376) =99 *(Other -- Used for when there is nothing to cancel).*
	3. There is also a possibility that orders were in cache at the time of the request but were removed prior to the Matching Host receiving the cancel request. This will result

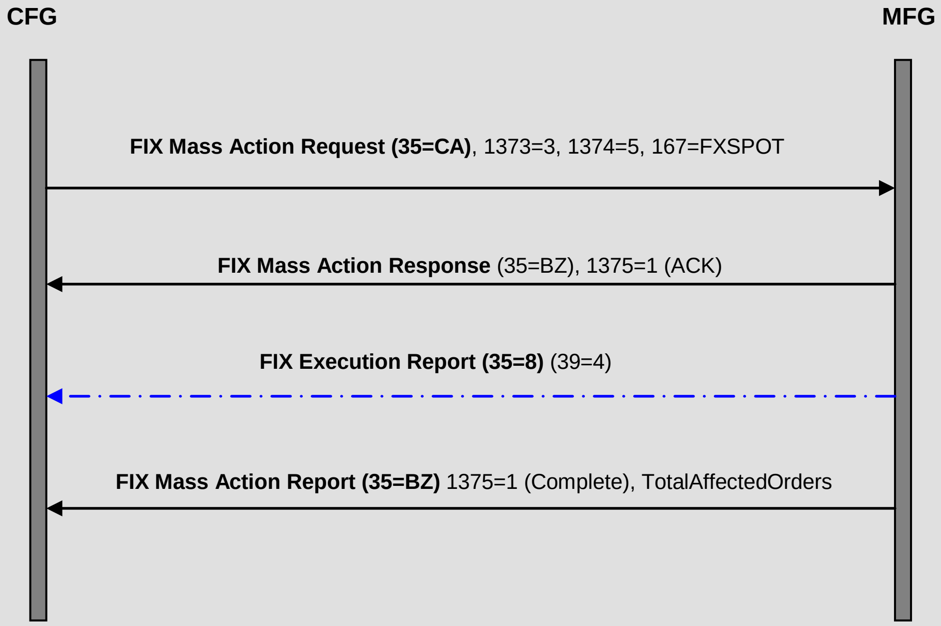
in FIX Mass Action Report (35=BZ) Ack followed by FIX Mass Action Report (35=BZ) complete message with TotalAffectedOrders=0.

FAILURE:

1. User may optionally receive a Mass Action Reject Report (BZ) with reason (1376) = 1 – Invalid or unknown security, or 5 – Invalid or unknown SecurityType.
3. Various reasons for Mass Action Cancel are defined in Matching Interface user guide Appendix A.

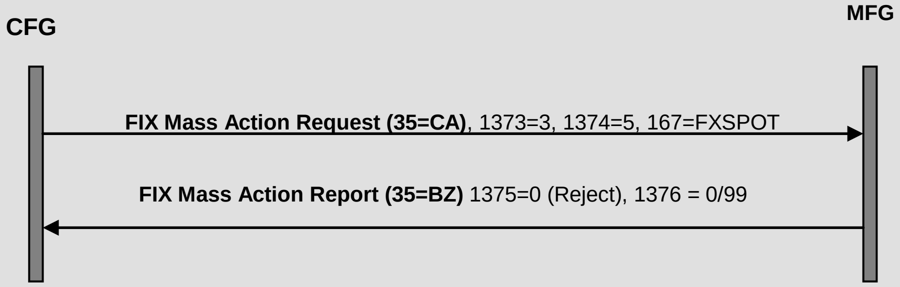
SUCCESS SCENARIO #1

User will receive Cancelled ExecutionReports as a result of MassActionRequest


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=5, 167=FXSPOT
FIX Mass Action Response (35=BZ), 1375=1 (ACK)
FIX Execution Report (35=8) (39=4)
FIX Mass Action Report (35=BZ) 1375=1 (Complete), TotalAffectedOrders

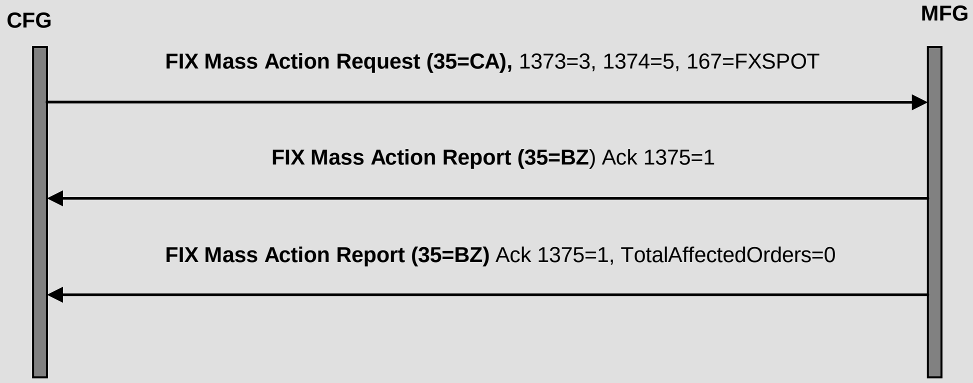
SUCCESS SCENARIO #2

User may receive MassActionReport with rejection if there is nothing to cancel


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=5, 167=FXSPOT
FIX Mass Action Report (35=BZ) 1375=0 (Reject), 1376 = 0/99

SUCCESS SCEANRIO #3

User may receive a MassActionReport with 0 affected orders if the orders were cancelled or filled before the MassActionReport was received by Matching Host


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=5, 167=FXSPOT
FIX Mass Action Report (35=BZ) Ack 1375=1
FIX Mass Action Report (35=BZ) Ack 1375=1, TotalAffectedOrders=0

5.18 Mass Action Cancel Request to cancel all of previously submitted messages for a specific user

Test Case 18:	**Purpose:** To verify that the Matching API user can submit a FIX Mass Action Cancel Request Message to cancel all of previously submitted messages for a specific user.

Description:

This test case will verify that the Matching API user can submit Cancel Request Message to cancel previously submitted messages on a specific instrument type.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Client should have additional user logins through same session to attempt this test.

Test Steps:

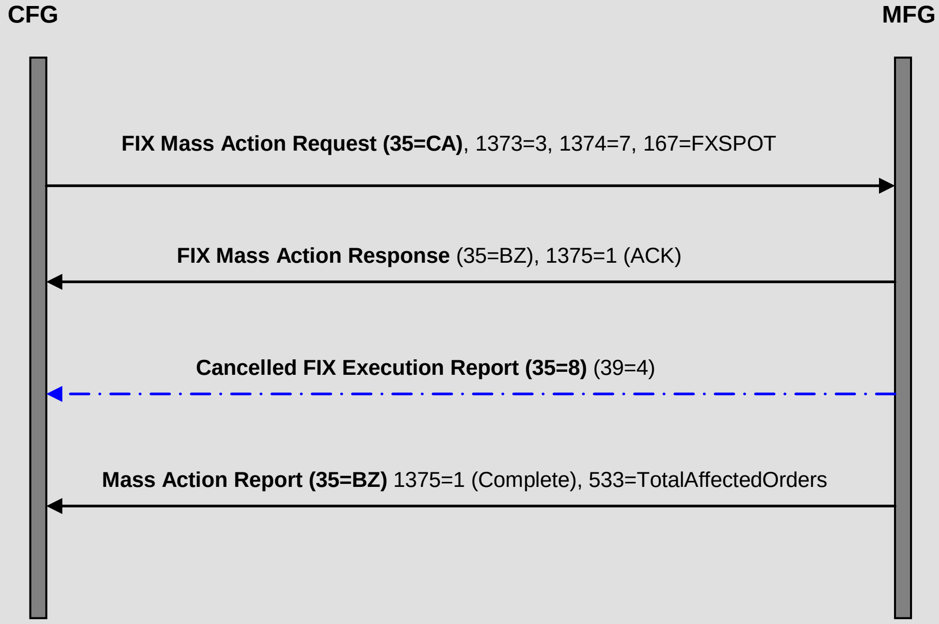
1. Send three EURUSD Spot Orders (35=D) and two GBPUSD Orders from User1.
2. Receive 5 ExecutionReports with OrdStatus=0 (New) on Order placement.
3. Send three EURUSD Spot Orders (35=D) from User2.
4. Build and send Mass Action Cancel Requests (35=CA) with (Tag 1373=3, Tag 1374=7) from User1 (Same as SenderSubID value).
5. Receive and process Cancel Pending ExecutionReports (35=8) on all five submitted orders from User1.

Input Specification:	**Expected Results:**
MassActionRequest (35=CA) as defined in Matching Interface User Guide. MassActionType (1373=3) MassActionScope(1374=7)	**SUCCESS: (below are all success scenarios)** 1. User should receive FIX Mass Action Report (35=BZ) ACK status (1375=1) followed by cancelled Execution Reports on orders submitted by User1. Once all the Execution Reports have been delivered, user will receive FIX Mass Action Report (35=BZ) to acknowledge completion of the report indicating TotalAffectedOrders (Tag 553=5 in this case). 2. User may optionally receive a FIX Mass Action Reject Report (35=BZ) with reason (1376) = 99 (Other -- Used for when there is nothing to cancel). 3. There is also a possibility that orders were in cache at the time of the request but were removed prior to the Matching Host receiving the cancel request. This will result in FIX Mass Action Report (35=BZ) Ack

	(1375=1) followed by FIX Mass Action Report (35=BZ) complete status (1375=1) with TotalAffectedOrders Tag 553=0. **FAILURE:** 1. Various reasons for MassActionRequest (CA) reject are defined in Matching Interface user guide Appendix A.

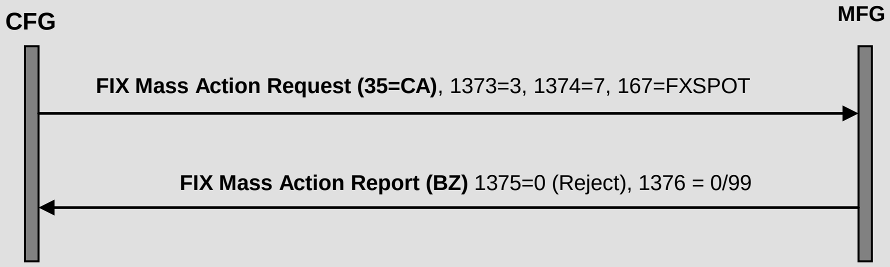
SUCCESS SCENARIO #1

User receives Cancelled ExecutionReports as a result of MassActionRequest


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=7, 167=FXSPOT
FIX Mass Action Response (35=BZ), 1375=1 (ACK)
Cancelled FIX Execution Report (35=8) (39=4)
Mass Action Report (35=BZ) 1375=1 (Complete), 533=TotalAffectedOrders

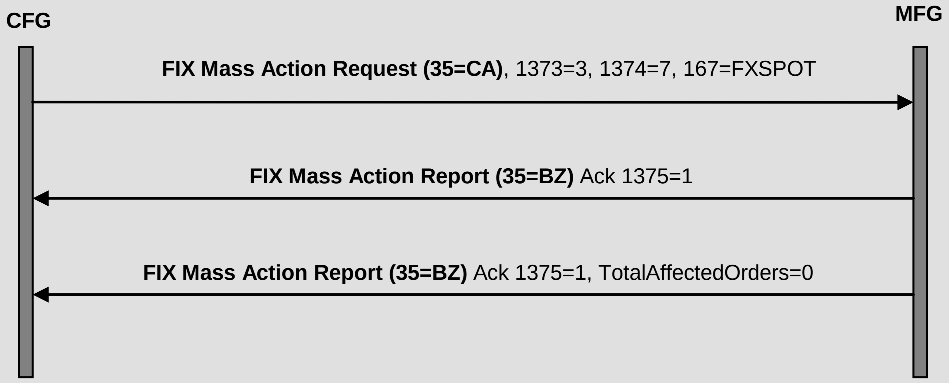
SUCCESS SCENARIO #2

User will receive MassActionReport with rejection if there is nothing to cancel


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=7, 167=FXSPOT
FIX Mass Action Report (BZ) 1375=0 (Reject), 1376 = 0/99

SUCCESS SCEANRIO #3

User may receive a MassActionReport with 0 affected orders if the orders were cancelled or filled before the MassActionReport was received by Matching Host


CFG
MFG
FIX Mass Action Request (35=CA), 1373=3, 1374=7, 167=FXSPOT
FIX Mass Action Report (35=BZ) Ack 1375=1
FIX Mass Action Report (35=BZ) Ack 1375=1, TotalAffectedOrders=0

5.19 List Cancel Request Message to cancel a previously submitted IDEAL order

Test Case 19:	Purpose:
	To verify that the Matching API user can submit a List Cancel Request Message to cancel a previously submitted IDEAL order.

Description:

List cancel request (35=K) should only be used to cancel both sides of an IDEAL order. Cancellation of a single side can be achieved via an OrderCancelRequest (35=F) message.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.

Test Steps:

1. Send FIX New Order List (35=E) message.
2. Receive FIX ExecutionReports (35=8) on the new Orders being positioned at Host.
3. Send a FIX List Cancel Request (35=K) with ListID as in the original IDEAL order.
4. Receive Cancelled Execution Reports followed by FIX List Status (35=N) response message to indicate completion of the List Cancel request (431=6). There is also a possibility that one or both sides of users order get filled before cancel is received by Host in which case user will receive an Execution Report (Filled) message.

Input Specification:	Expected Results:
FIX New Order List (35=E) as defined in Matching Interface User Guide. **List Cancel Request (35=K)** as defined in Matching Interface User Guide section 4.5.3.3 with ListID (Tag 66="ListID from the IDEAL order submitted previously")	**SUCCESS: (below are all success scenarios)** 1. Matching API user will receive a ListStatus (35=N) Ack message (429=1, 83=1, 431=4) with OrdStatus=6 to indicate Pending Cancellation status followed by any ExecutionReports which are cancelled due to the ListCancelRequest. Finally, ListStatus response (35=N, 429=2, 83=2, 431=6) is received to indicate completion of the ListCancelRequest. ListStatus response will have status of both cancelled orders (68=2, 11=2). 2. Alternatively, Matching API user may receive a ListStatus (35=N) Ack message (429=1, 83=1, 431=4) followed by any ExecutionReports while the ListCancelRequest is in flight. A ListStatus response (35=N, 429=2, 83=1) is received with ListOrderStatus (431=7) Reject to indicate that no Orders were cancelled as a direct result of the ListCancelRequest.

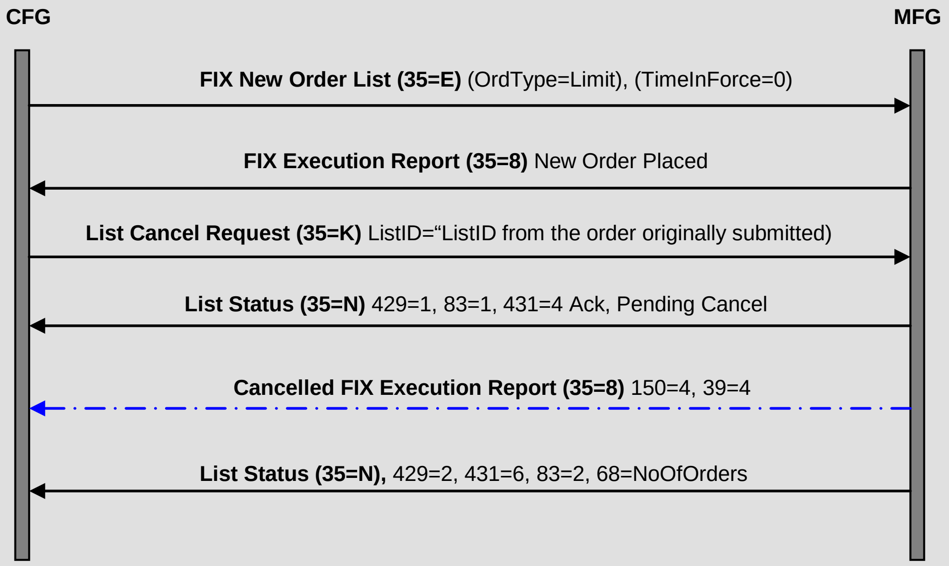
3. Also there is the possibility that by the time FIX Gateway receives a ListCancel request, one side of the IDEAL order is filled already; under such circumstance only remaining side of the order will get cancelled. Matching API user will receive a ListStatus (35=N) Ack message (429=1, 83=1, 431=4) followed by a Cancelled ExecutionReport. A ListStatus response (35=N, 429=2, 83=1) is received with ListOrderStatus (431=6) and status for the one order that’s cancelled due to this ListCancelRequest.
4. There is also the possibility that by the time FIX Gateway receives a ListCancel request one side or the other side is already being matched or cancelled (by other means). Under such circumstance, Matching API user will receive a ListStatus (35=N) Ack message (429=1, 83=1, 431=4) followed by a ListStatus response reject (35=N, 429=2, 83=1, 431=7) .

FAILURE:

1. Various reasons for ListCancelRequest reject are defined in Matching Interface user guide Appendix A.

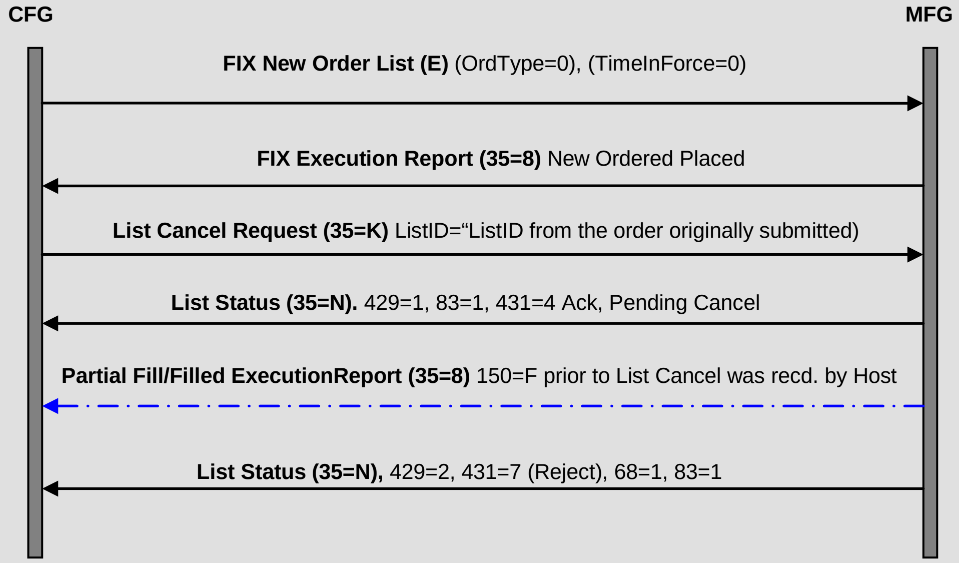
SUCCESS SCENARIO #1

User receives cancelled ExecutionReports due to ListCancelRequest for IDEAL order


CFG
MFG
FIX New Order List (35=E) (OrdType=Limit), (TimeInForce=0)
FIX Execution Report (35=8) New Order Placed
List Cancel Request (35=K) ListID=“ListID from the order originally submitted)
List Status (35=N) 429=1, 83=1, 431=4 Ack, Pending Cancel
Cancelled FIX Execution Report (35=8) 150=4, 39=4
List Status (35=N), 429=2, 431=6, 83=2, 68=NoOfOrders

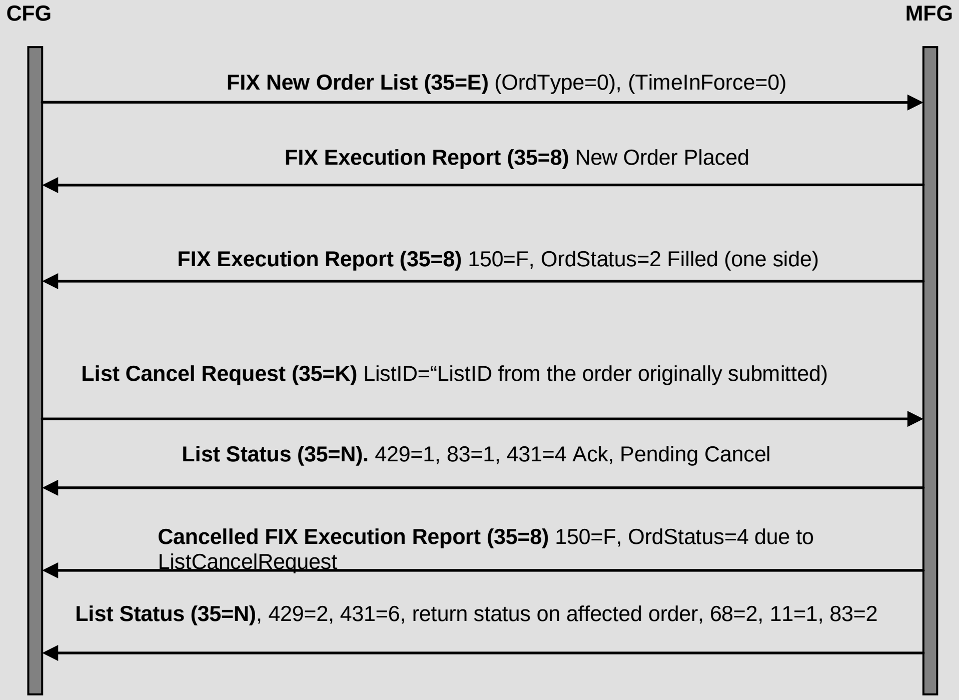
SUCCESS SCENARIO #2

User receives Filled ExecutionReports and reject ListStatus as there are no orders to cancel


CFG
MFG
FIX New Order List (E) (OrdType=0), (TimeInForce=0)
FIX Execution Report (35=8) New Ordered Placed
List Cancel Request (35=K) ListID="ListID from the order originally submitted)
List Status (35=N). 429=1, 83=1, 431=4 Ack, Pending Cancel
Partial Fill/Filled ExecutionReport (35=8) 150=F prior to List Cancel was recd. by Host
List Status (35=N), 429=2, 431=7 (Reject), 68=1, 83=1

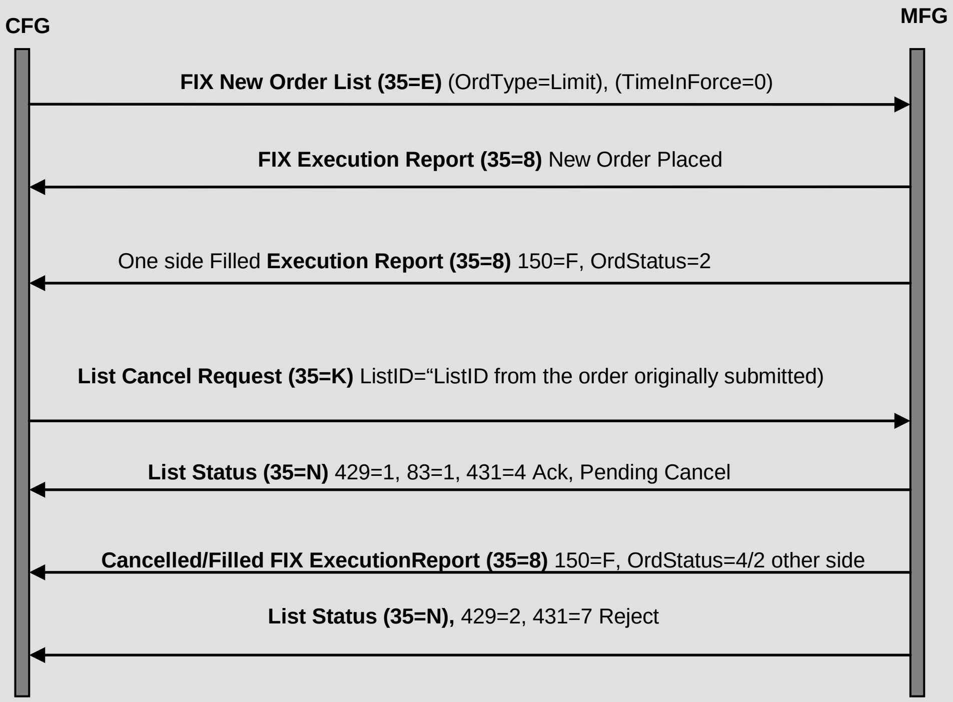
SUCCESS SCENARIO #3

User receives a Filled Execution report on one side of the order while the other side gets cancelled as a result of ListCancelRequest


CFG
MFG
FIX New Order List (35=E) (OrdType=0), (TimeInForce=0)
FIX Execution Report (35=8) New Order Placed
FIX Execution Report (35=8) 150=F, OrdStatus=2 Filled (one side)
List Cancel Request (35=K) ListID="ListID from the order originally submitted)
List Status (35=N). 429=1, 83=1, 431=4 Ack, Pending Cancel
Cancelled FIX Execution Report (35=8) 150=F, OrdStatus=4 due to ListCancelRequest
List Status (35=N), 429=2, 431=6, return status on affected order, 68=2, 11=1, 83=2

SUCCESS SCENARIO #4

User receives Filled/Cancelled ExecutionReports by other means and a ListStatus reject is sent to Client as there are no orders to cancel any more


CFG
MFG
FIX New Order List (35=E) (OrdType=Limit), (TimeInForce=0)
FIX Execution Report (35=8) New Order Placed
One side Filled **Execution Report (35=8)** 150=F, OrdStatus=2
List Cancel Request (35=K) ListID=“ListID from the order originally submitted)
List Status (35=N) 429=1, 83=1, 431=4 Ack, Pending Cancel
Cancelled/Filled FIX ExecutionReport (35=8) 150=F, OrdStatus=4/2 other side
List Status (35=N), 429=2, 431=7 Reject

5.20 Hold and Release

Test Case 20:

Purpose:

To verify that the user can Hold and Release all his submitted orders.

Description:

This test case will verify that a FIX User can successfully put all his orders on hold, followed by a release of those orders.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.

Test Steps:

1. Submit 4 FIX New Order Single (35=D) messages.
2. Receive 4 New order placed Execution Reports (35=8) with ExecType=0.
3. Put them on hold by sending FIX Mass Action Request (35=CA) with (Tag 1373=1) and (Tag 1374=7) for all orders from this user (this test can be done for Tag 1374=1 for all orders of a specific symbol, or 1374=5 for all orders for a security type). If putting all messages with same Symbol on hold then provide the Tag 55 value.
4. Wait few secs and then release these orders by sending (35=CA), Tag 1373=2. If only a specific Symbol orders were put on hold then specify Tag 55 value.
5. Client is required to indicate in the certification spreadsheet the exact scenario they attempted.

Input Specification:

To put orders on Hold:

FIX Order Mass Action Request (35=CA) as defined in Matching Interface User Guide.

MassActionType (1373=1)

MassActionScope (1374=7), all orders for a given user

To release orders:

FIX Order Mass Action Request (35=CA) as defined in Matching Interface User Guide.

MassActionType (1373=2)

MassActionScope (1374=7), all orders for a user

Expected Results:

SUCCESS:

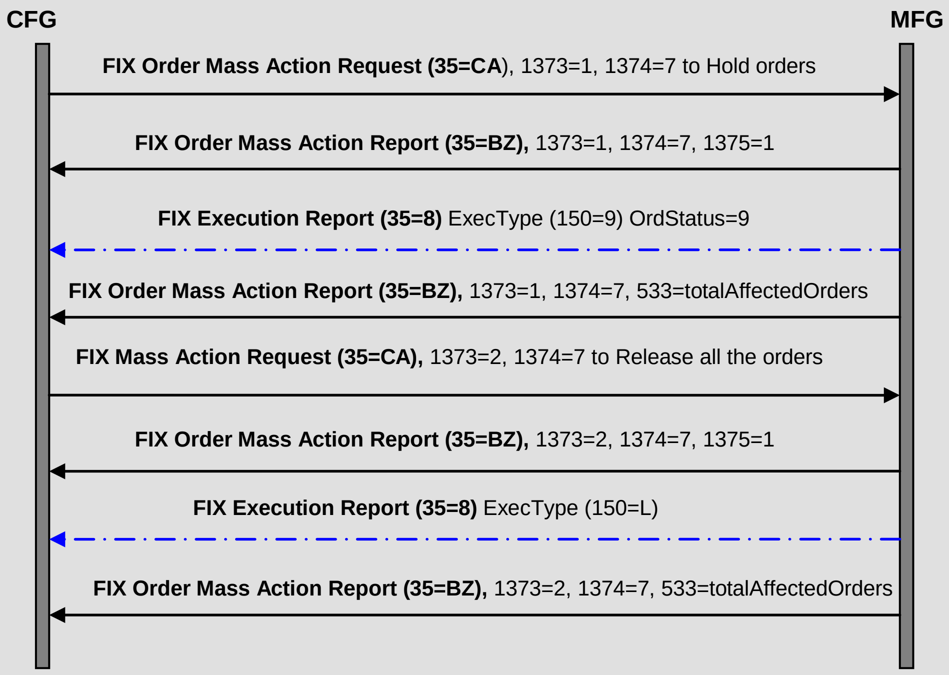
1. Matching API user will receive an Order Mass Action Report (35=BZ) followed by Execution Report (35=8) with ExecType=9 (Hold) in response to Order Mass Action Request 1373=1 (Suspend). Matching API user will receive ExecutionReport (35=8) with ExecType=L when orders are released in response to Order Mass Action Request 1373=2 (Release).
2. Alternatively, a Matching user can also receive a Mass Action Report (BZ) reject message (1375=0 with 1376 with reject reason) if there is nothing to cancel or reject.

FAILURE:

1. Various reasons for MassActionRequest reject are defined in Matching Interface user guide Appendix A.

SUCCESS SCENARIO

User successfully puts all the orders in Hold and then releases them


CFG
MFG
FIX Order Mass Action Request (35=CA), 1373=1, 1374=7 to Hold orders
FIX Order Mass Action Report (35=BZ), 1373=1, 1374=7, 1375=1
FIX Execution Report (35=8) ExecType (150=9) OrdStatus=9
FIX Order Mass Action Report (35=BZ), 1373=1, 1374=7, 533=totalAffectedOrders
FIX Mass Action Request (35=CA), 1373=2, 1374=7 to Release all the orders
FIX Order Mass Action Report (35=BZ), 1373=2, 1374=7, 1375=1
FIX Execution Report (35=8) ExecType (150=L)
FIX Order Mass Action Report (35=BZ), 1373=2, 1374=7, 533=totalAffectedOrders

5.21 Forward Swap order workflow

Test Case 21:

Purpose:

To verify that the user can submit a Standard Forward Swap order and process subsequent Execution Report messages from Matching FIX Gateway. This test case is only applicable to users who are trading or permissioned for Forward, not applicable to PBCs.

Description:

This test case will verify that a Matching user can submit a Standard Swap order correctly and process all the ExecutionReports received thereon.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user should have access to a manual keystation which should be logged in as Negotiation user to pickup the escalated proposal for manually completing the trade.

Test Steps:

1. Matching API user will submit a FIX New Order Single Order (35=D) with SecurityType=FXSWAP, TimeInForce=0 (DAY) and Negotiation user specified in the PartyID (453=1) of the Order.
2. Matching API user will receive FIX Execution Report (New) followed by Part Filled/Filled or Cancelled ExecutionReports.
3. Negotiation user will see the escalated trade on his keystation and should accept or reject the deal.
4. Once the trade is escalated to Negotiation user the API will not receive any notifications of any nature.

Input Specification:

FIX New Order single (35=D) as defined in Matching Interface User Guide.

SecurityType (167=FXSWAP)

TimeInForce=0 (Day)

NoOfPartyIDs (453=1)

- 448=PartyID (for ex. AUTA, negotiation user)
- 447=C
- 452=53

Symbol (55=eur/usd 1M)

Expected Results:

SUCCESS:

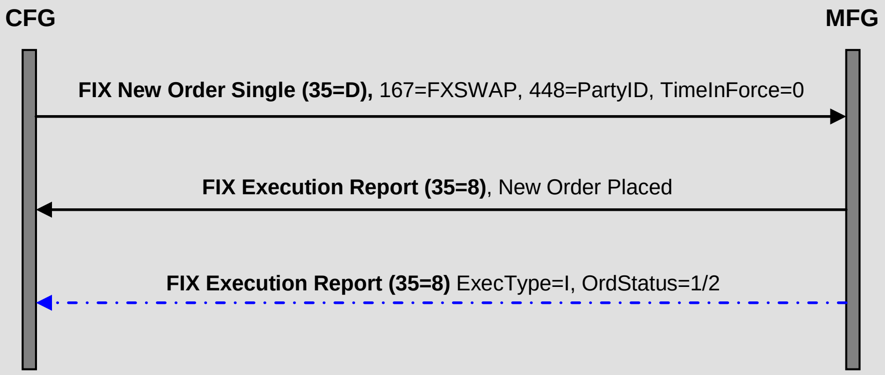
1. A Forward Swap order could be partial/fully filled after resting in the book. Matching API user will receive ExecutionReport ExecType=I on Partial/Fully filled Matches.
2. Order is rejected upon entry due to negotiation user not logged in.
3. Order is partial matched and cancelled due to end of Trading or TimeInForce expiration or because negotiation user log off or disconnects.

FAILURE:

1. Matching API user will receiver OrdStatus (rejected) if the trade is rejected due to Negotiation user being invalid.

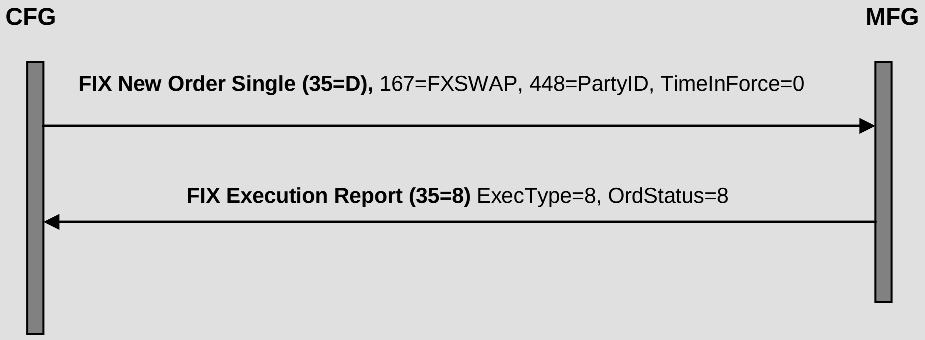
SUCCESS SCENARIO #1

User receives fills/partial fills on submitted order


CFG
MFG
FIX New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
FIX Execution Report (35=8), New Order Placed
FIX Execution Report (35=8) ExecType=I, OrdStatus=1/2

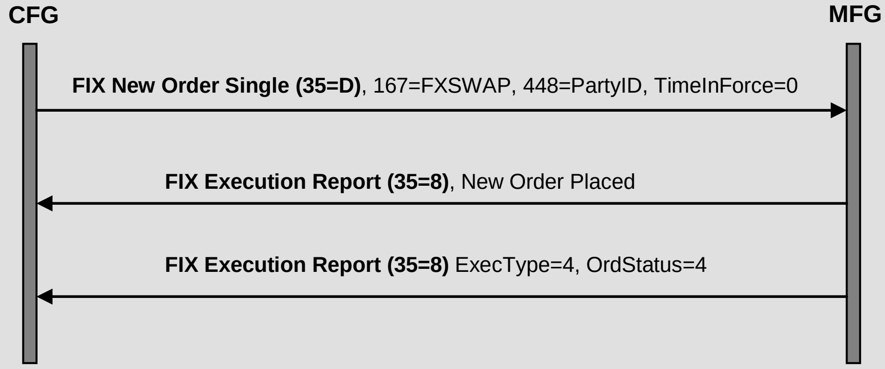
SUCCESS SCENARIO #2

User receives Rejected order status because Transfer user isn’t logged in


CFG
MFG
FIX New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
FIX Execution Report (35=8) ExecType=8, OrdStatus=8

SUCCESS SCENARIO #3

User receives Cancelled order status because Transfer user logs off or disconnects after successful Order entry


CFG
MFG
FIX New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
FIX Execution Report (35=8), New Order Placed
FIX Execution Report (35=8) ExecType=4, OrdStatus=4

5.22 Aggressor (Mine/Yours) Forward Swap order submission

Test Case 22:	**Purpose:** To verify that the user can submit an aggressor (Mine/Yours) Forward Swap order and process subsequent ExecutionReport messages from Matching FIX Gateway.

Description:

This test case will verify that a Matching user can submit an aggressor (MINE/YOURS) Forward Swap order correctly and process all the ExecutionReports received thereon.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user should have access to a manual keystation which should be logged in as Negotiation user to pickup the escalated Proposal for manually completing the trade.

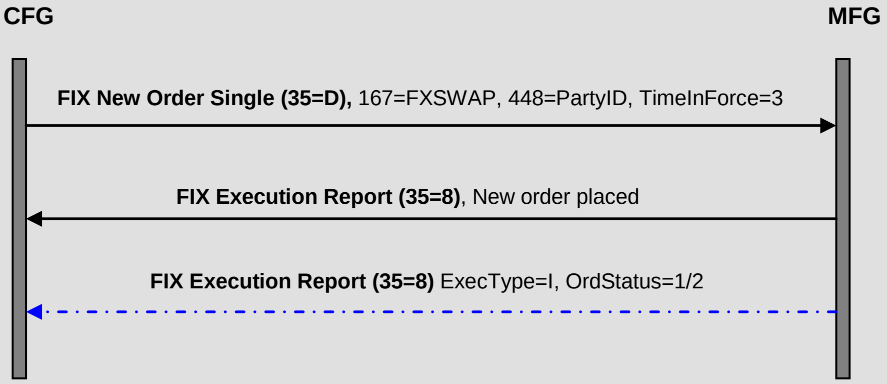
Test Steps:

1. Matching API user will submit a Swap order (35=D) with SecurityType=FXSWAP, TimeInForce=3 and Negotiation user specified in the PartyID (453=1) of the Order.
2. Matching API user will receive FIX Execution Report (New) for order placement in the book once a Match happens.
3. Negotiation user will see the escalated trade on his keystation and should accept or reject the deal.
4. Once the trade is escalated to Negotiation user the API will not receive any notifications of any nature.

Input Specification:	**Expected Results:**
FIX New Order Single (35=D) as defined in Matching Interface User Guide. SecurityType (167=FXSWAP) NoOfPartyIDs (453=1) o 448=PartyID (for ex. AUTA, negotiation user) o 447=C o 452=53 Symbol (55=eur/usd 1M) TimeInForce=3	**SUCCESS:** 1. A Forward Swap order could be partial filled after resting in the book. Matching API user will receive Execution Report ExecType=I on Partial/Fully filled Matches. 2. Order is rejected upon entry due to negotiation user not logged in. 3. Order has OrdStatus=7(Stopped) due to more then 10 Proposals being open on negotiation user's workspace. **FAILURE:** 1. Matching API user will receiver Ordstatus (rejected) if the trade is rejected due to Negotiation user being invalid.

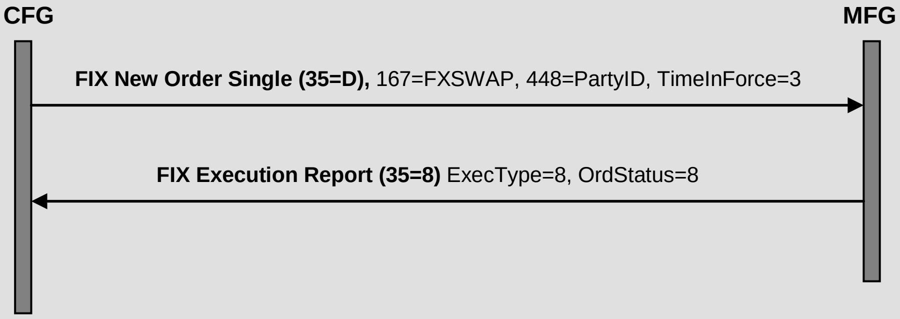
SUCCESS SCENARIO #1

User receives fills/partial fills on submitted order


CFG
MFG
FIX New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=3
FIX Execution Report (35=8), New order placed
FIX Execution Report (35=8) ExecType=I, OrdStatus=1/2

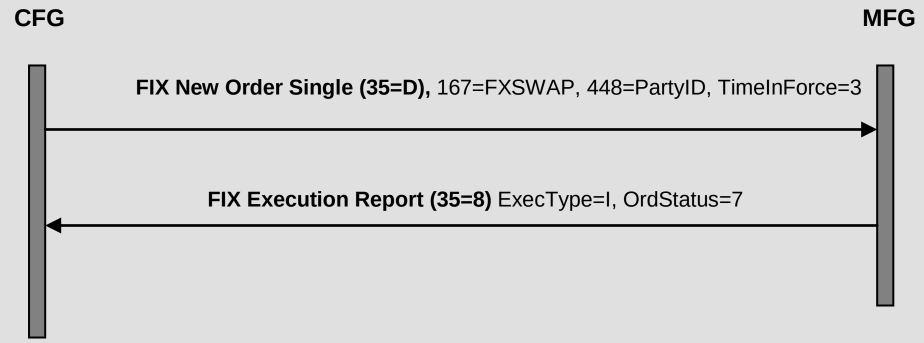
SUCCESS SCENARIO #2

User receives Rejected order status because Transfer user isn't valid


CFG
MFG
FIX New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=3
FIX Execution Report (35=8) ExecType=8, OrdStatus=8

SUCCESS SCENARIO #3

User receives Stopped status since Negotiation user has more then 10 Proposals open in his workspace


CFG
MFG
FIX New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=3
FIX Execution Report (35=8) ExecType=I, OrdStatus=7

5.23 Cancellation of a Forward Swap order

Test Case 23:

Purpose:

To verify that the user can submit a Forward Swap order and Cancel it after partial fills.

Description:

This test case will verify that a Matching user can submit a Standard Forward Swap order correctly and can Cancel it after receiving partial fills on it.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user should have access to a manual keystation which should be logged in as Negotiation user to pickup the open Proposal for manually completing the trade.
- Client should liaise with MAPI on-boarding team so they can aggress on the client order to simulate required behaviour.

Test Steps:

1. Matching API user will submit a Forward Swap order (35=D) with SecurityType=FXSWAP, TimeInForce=0, OrdQty=25 and Negotiation user specified in the PartyID (453=1) of the Order.
2. Matching API user will receive FIX Execution Report (New) ExecType=0 for order placement in the book.
3. A Counterparty will Match partial order (Matched amount 10, leaving 15 in the book).
4. Matching API user will receive FIX Execution Report on partial fills.
5. Matching API user will cancel any remaining quantity by sending Cancel Request on pending order.
6. Matching API user will receive FIX Execution Reports with Pending Cancel and Cancelled Order.

Input Specification:

Swap order (35=D) as defined in Matching Interface User Guide.

SecurityType (167=FXSWAP)

NoOfPartyIDs (453=1)

- 448=PartyID (for ex. AUTA, negotiation user)
- 447=C
- 452=53

Symbol (55=EUR/USD 1M)

TimeInForce=0

Cancel Request (35=F) as defined in Matching Interface User Guide.

OrigClOrdID=ClOrdID of the previously submitted order.

Expected Results:

SUCCESS:

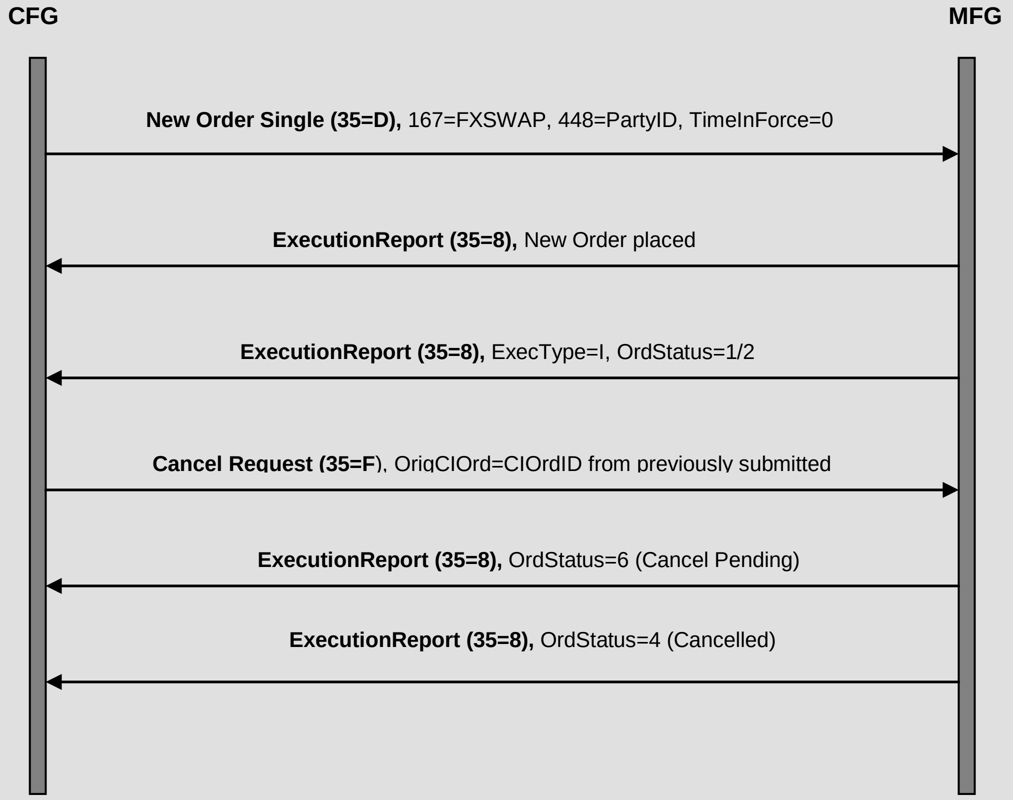
1. Matching API user should receive Cancelled Execution Reports on pending order after sending successful cancellation on outstanding Swap limit order.
2. Matching API user Swap order gets fully filled upon entry.

FAILURE:

1. Matching API user will receiver Ordstatus (rejected) if the trade is rejected due to Negotiation user being invalid.

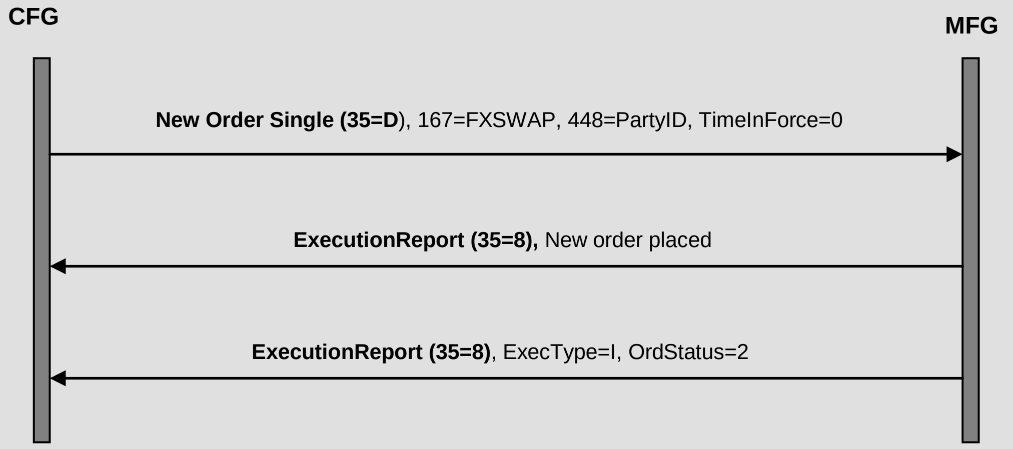
SUCCESS SCENARIO #1

User receives Fill/partial Fills on submitted order and receive successful cancellations on Cancelled Swap order


CFG
MFG
New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
ExecutionReport (35=8), New Order placed
ExecutionReport (35=8), ExecType=I, OrdStatus=1/2
Cancel Request (35=F), OrigClOrd=ClOrdID from previously submitted
ExecutionReport (35=8), OrdStatus=6 (Cancel Pending)
ExecutionReport (35=8), OrdStatus=4 (Cancelled)

SUCCESS SCENARIO #2

User's order gets fully filled upon entry


CFG
MFG
New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
ExecutionReport (35=8), New order placed
ExecutionReport (35=8), ExecType=I, OrdStatus=2

5.24 Ability to handle unacknowledged matches

Test Case 24:	**Purpose:** To verify that the Matching API user is able to handle unacknowledged trade confirmation messages.

Description:

Under certain circumstances a trade may go unacknowledged, for ex. counterparty or PB STU disconnects from Matching service as the Match is in progress, and this can cause a trade to go unconfirmed at Host. This test case will verify that Matching API user application is able to handle unacknowledged Trade Capture Reports and take the required action to reconcile their trade positions.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user has additional session/subscriber login so they can aggress on their own prices to test this scenario or ask for testing assistance from MAPI on-boarding team.

Test Steps:

Client application will do following steps:

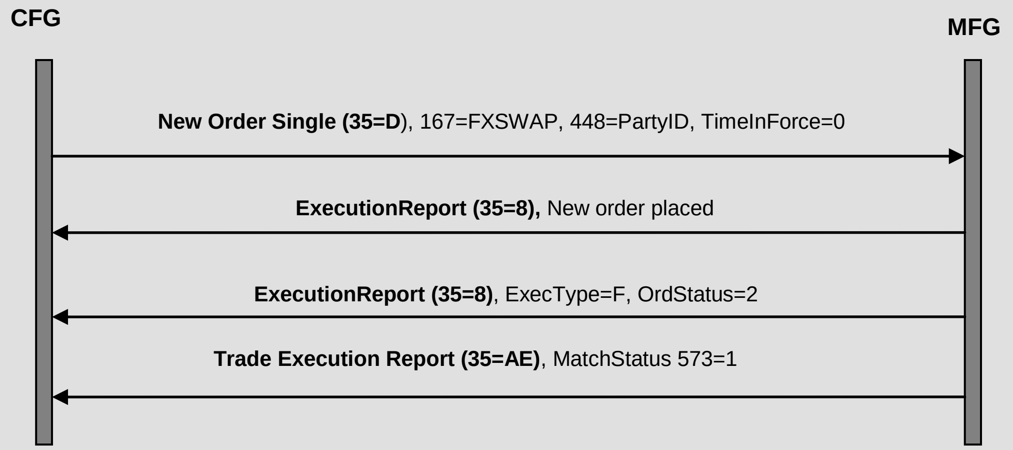
1. Build a FIX New Order Single (35=D) for Sell USD on usd/xxx Symbol at gbp/usd price. usd/xxx pair is built to test unacknowledged scenarios and it has same pricing as gbp/usd in test HTG system.
2. Receive an Execution Report for New order submitted with OrderID (Tag 37), ExecType 'New' (Tag 150=0) and OrdStatus 'New' (Tag 39=0). This indicates Order is placed in book.
3. Receive Fill/Partial Fill or Cancelled FIX Execution Report ((35=8) messages based on the no. of matches against this order.
4. Process received FIX Trade Capture Report (35=AE) message with Match Status (573=1) indicating an unconfirmed trade.
5. Since user application receives an unconfirmed trade so it is expected to raise an alert to indicate support escalation. Client is expected to obtain their Daily Confirmation Statement via Dealing Xtra (or if not available then obtain from Thomson Reuters Help Desk) and check the deal. If the deal appears on your Daily Confirmation Statement whatever the status, then the deal is done. If the deal does not appear on the Daily Confirmation Statement then the deal is not done. Alternatively, client can also confirm with their PB or their counterparty.

Input Specification:	**Expected Results:**
FIX New Order Single (35=D) as defined in Matching Interface User Guide. OrdType (Tag 40=2) TimeInForce (Tag 59=0) Side (Tag 54=2) Currency (Tag 15=USD) Symbol (Tag 55=usd/xxx) Price (Tag 44) (should be gbp/usd price for sell)	**SUCCESS: (below are all success scenarios)** 1. Client receives multiple Partial Fill ExecutionReport (35=8) followed by a Fully Filled ExecutionReport with ExecType (150=F) and OrdStatus (39=2). Client application should also receive a TCR (35=AE) corresponding to every Filled/Partial Filled ExecutionReport with Tag 573=1. TrdMatchID (Tag 880) in TCR should match with what's received thru ExecutionReport. TradeID (Tag 1003) in TCR

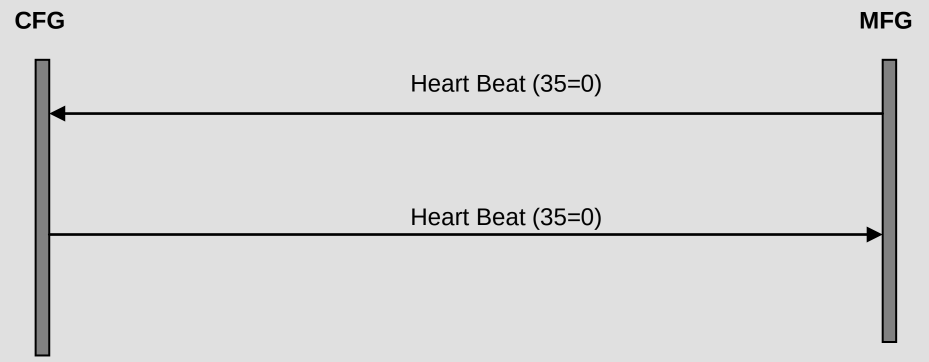
OrderQty (Tag 38=20) to fill an amount of 20	should match the SecondaryExecID (Tag 527) in the ExecutionReport. 2. Client receives multiple partial Fill ExecutionReport's with OrdStatus=1, followed by an unsolicited cancel ExecutionReport (OrdStatus=4) due to user disconnect or abrupt end of trading session. Process unconfirmed TCRs with 573=1. 3. Client receives one Fill ExecutionReport (OrdStatus=2) indicating that the order was fully filled upon hitting the book. Process unconfirmed TCRs with 573=1. 4. Client receives ExecutionReport with partial fill (OrdStatus=1) indicating the remaining quantity still staying in the book for further possible matches. Process unconfirmed TCRs with 573=1. 5. Client receives a reject ExecutionReport (OrdStatus=8, ExecType=8). Process unconfirmed TCRs with 573=1. **FAILURE:** 1. Message submitted by Matching user has permission error which may result in BusinessMessageReject (j) from Matching FIX Gateway. 2. Various reasons for New Order Single reject are explained in Matching Interface guide Appendix A.

SUCCESS SCENARIO

User receives Fully Filled order on submitted order and receives an unconfirmed trade status


CFG
MFG
New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
ExecutionReport (35=8), New order placed
ExecutionReport (35=8), ExecType=F, OrdStatus=2
Trade Execution Report (35=AE), MatchStatus 573=1

5.25 Ability to handle Heart Beat Messages

<table>
<tr><td><b>Test Case 25:</b></td><td><b>Purpose:</b>
To verify that the user is able to send Heart Beat messages at required interval.</td></tr>
<tr><td colspan="2"><b>Description:</b>
The Matching FIX Gateway uses a Heartbeat interval of 4 secs. In the absence of any outgoing traffic for a period of 4 seconds both ends of the FIX link will be expected to generate an outgoing heartbeat.</td></tr>
<tr><td colspan="2"><b>Prerequisites:</b>
• Client should have an established session with Matching FIX Gateway.</td></tr>
<tr><td colspan="2"><b>Test Steps:</b>
1. Client application will send Heart Beat messages for 2 mins at 4 sec interval.
2. Client application will send 2 order messages and process any execution and match/fill notifications as a result.
3. Client application will send Heart Beat messages again for 2 mins at 4 sec interval.</td></tr>
<tr><td><b>Input Specifications:</b></td><td><b>Expected Results:</b></td></tr>
<tr><td><b>Heartbeat Message (35=0)</b> as defined in Matching Interface User Guide.
SendingTime (Tag 52)</td><td><b>SUCCESS:</b>
• Client is able to send Heartbeat messages correctly at 4 sec interval.
<b>FAILURE:</b>
• Client fails to send Heartbeat message at 4 sec interval at times.</td></tr>
<tr><td colspan="2"><b>SUCCESS SCENARIO</b>
 </td></tr>
</table>

5.26 Ability to handle Logout Message

Test Case 26:	**Purpose:** To verify that the user is able to send Logout message in order to do a graceful disconnect from Matching FIX Gateway.

Description:

The Matching FIX Gateway has an alerting mechanism where if a Client application disconnects abruptly then it sends an auto alert to support team. To avoid MFG generating unnecessary alerts it is required for Client applications to send a graceful Logout when they wish to disconnect from Matching FIX Gateway.

Prerequisites:

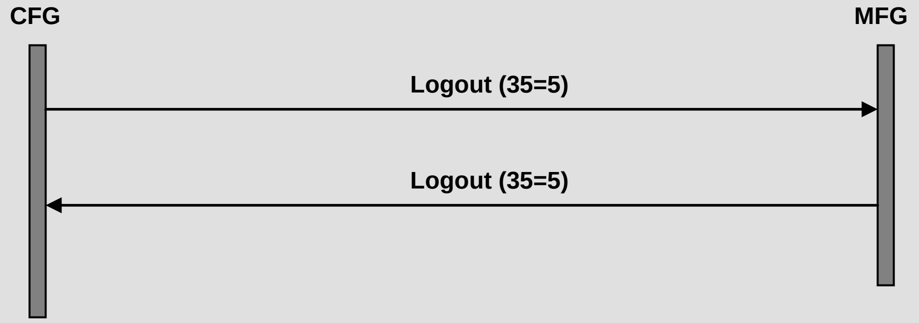
- Client should have an established session with Matching FIX Gateway.

Test Steps:

1. Client application will send 2 order messages and process any execution and match/fill notifications as a result.
2. Client application will send a Logout message to indicate a graceful disconnect from MFG.

Input Specifications:	**Expected Results:**
FIX New Order Single (35=D) as defined in Matching Interface User Guide. **FIX Logout message (35=5)**	**SUCCESS:** • Client receives a successful Logout from MFG and any cancelled orders as a result of Client initiated Logout. **FAILURE:** • Client fails to send Logout or process Logout from MFG.

SUCCESS SCENARIO



5.27 Drop Copy workflow

Test Case 27:

Purpose:

To verify that the Matching API user is able to handle Drop Copy messages.

Description:

This test case verifies that client application is correctly able to receive and process Trade Capture Reports through a Drop Copy Session.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user should have connection to a trading session so they can execute trades.

Test Steps:

Client application will do following steps:

1. Connect to the trading session via trader user (USRE or USRF).
2. Connect to drop copy session with a drop copy user; only FIX session login is required for receiving drop copy.
3. Build a FIX New Order Single (35=D) for 1 M Sell EUR on eur/usd.
4. Receive an Execution Report for New order submitted with OrderID (Tag 37), ExecType ‘New’ (Tag 150=0) and OrdStatus ‘New’ (Tag 39=0). This indicates Order is placed in book.
5. Receive Fill or Cancelled FIX Execution Report ((35=8) message. Receive TCR on trading session.
6. Receive and process Trade Capture Reports on Drop Copy session provided the order wasn’t cancelled or rejected.

Input Specification:

FIX New Order Single (35=D) as defined in Matching Interface User Guide.

OrdType (Tag 40=2)

TimeInForce (Tag 59=0)

Side (Tag 54=2)

Currency (Tag 15=EUR)

Symbol (Tag 55=eur/usd)

Price (Tag 44)

OrderQty (Tag 38=20)

Expected Results:

SUCCESS: (below are all success scenarios)

1. Client receives Fill ExecutionReport (35=8) with ExecType (150=F) and OrdStatus (39=2) on trading session. Client application should also receive a TCR (35=AE) confirming the Match. TrdMatchID (Tag 880) in TCR should match with what’s received thru ExecutionReport. TradeID (Tag 1003) in TCR should match the SecondaryExecID (Tag 527) in the ExecutionReport. Client also receives same TCR on Drop Copy session.
2. Client receives a reject ExecutionReport (OrdStatus=8, ExecType=8).

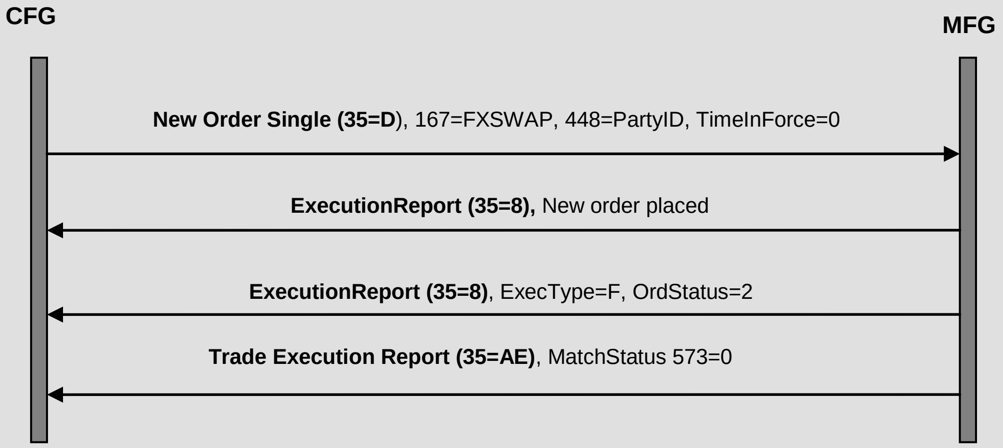
FAILURE:

1. Message submitted by Matching user has permission error which may result in

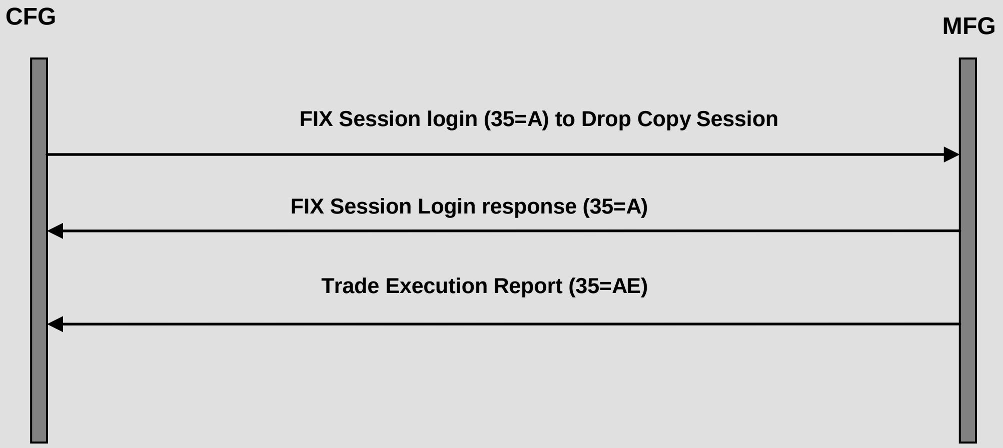
<table>
<tr><td></td><td>BusinessMessageReject (j) from Matching FIX Gateway.
2. Various reasons for New Order Single reject are explained in Matching Interface guide Appendix A.</td></tr>
<tr><td colspan="2">

SUCCESS SCENARIO

User receives Execution Report and TCR on Trading Session


CFG
MFG
New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
ExecutionReport (35=8), New order placed
ExecutionReport (35=8), ExecType=F, OrdStatus=2
Trade Execution Report (35=AE), MatchStatus 573=0

User receives TCR on Drop Copy Session


CFG
MFG
FIX Session login (35=A) to Drop Copy Session
FIX Session Login response (35=A)
Trade Execution Report (35=AE)

</td></tr>
</table>

5.28 Drop Copy session disconnect and recovery workflow

Test Case 28:	**Purpose:** To verify that the Matching API user is able to handle session disconnect and recovery of pending messages.

Description:

This test case verifies that client application is correctly able to receive and process Trade Capture Reports through a Drop Copy Session after it reconnects to MFG.

Prerequisites:

- Client should have an established session with Matching FIX Gateway.
- Matching API user should have connection to a trading session so they can execute trades.

Test Steps:

Client application will do following steps:

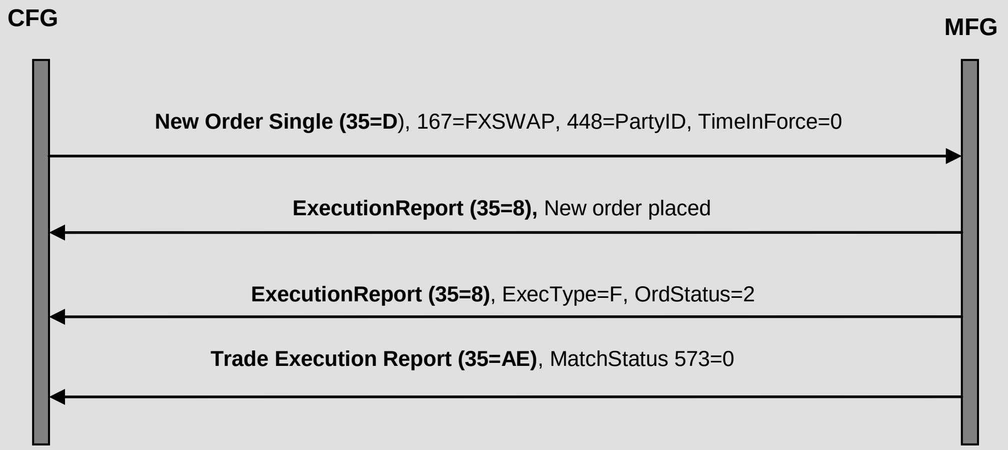
1. Connect to the trading session via trader user (USRE or USRF).
2. Connect to drop copy session with a drop copy user; only FIX session login is required for receiving drop copy.
3. Build a FIX New Order Single (35=D) for Sell EUR on eur/usd for 1M.
4. Receive an Execution Report for New order submitted with OrderID (Tag 37), ExecType ‘New’ (Tag 150=0) and OrdStatus ‘New’ (Tag 39=0). This indicates Order is placed in book.
5. Receive Fill or Cancelled FIX Execution Report ((35=8) message. Receive TCR on trading session.
6. Receive and process Trade Capture Report on Drop Copy session provided the order wasn’t cancelled or rejected.
7. Disconnect Drop copy session.
8. Send 10 trades and receive filled execution reports and TCRs.
9. Reconnect Drop copy session to MFG and confirm receiving 10 TCR’s which were pending since last disconnect. TrdMatchID on the TCRs received from trading session and Drop copy session should match. Also, the TargetSubID on the Drop Copy session will be same as the TargetSubID in the trading session.

Input Specification:	**Expected Results:**
FIX New Order Single (35=D) as defined in Matching Interface User Guide. OrdType (Tag 40=2) TimeInForce (Tag 59=0) Side (Tag 54=2) Currency (Tag 15=EUR) Symbol (Tag 55=eur/usd)	**SUCCESS: (below are all success scenarios)** 1. Client receives Fill ExecutionReport (35=8) with ExecType (150=F) and OrdStatus (39=2) on trading session. Client application should also receive a TCR (35=AE) confirming the Match. TrdMatchID (Tag 880) in TCR should match with what’s received thru ExecutionReport. TradeID (Tag 1003) in TCR should match the SecondaryExecID (Tag 527) in the

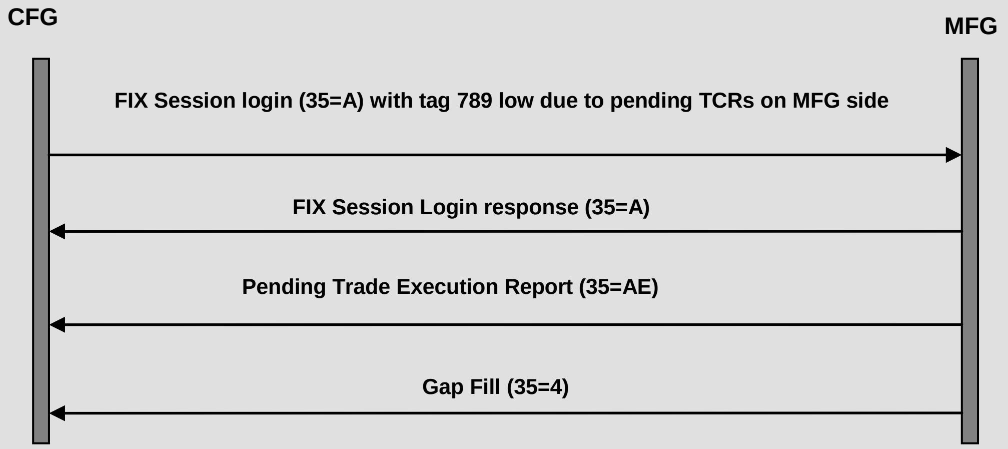
Price (Tag 44) OrderQty (Tag 38=20)	ExecutionReport. Client also receives same TCR on Drop Copy session post reconnect. **FAILURE:** 1. Message submitted by Matching user has permission error which may result in BusinessMessageReject (j) from Matching FIX Gateway. 2. Various reasons for New Order Single reject are explained in Matching Interface guide Appendix A.

SUCCESS SCENARIO

User receives Execution Report and TCR on Trading Session


CFG
MFG
New Order Single (35=D), 167=FXSWAP, 448=PartyID, TimeInForce=0
ExecutionReport (35=8), New order placed
ExecutionReport (35=8), ExecType=F, OrdStatus=2
Trade Execution Report (35=AE), MatchStatus 573=0

User receives TCR on Drop Copy Session post reconnect


CFG
MFG
FIX Session login (35=A) with tag 789 low due to pending TCRs on MFG side
FIX Session Login response (35=A)
Pending Trade Execution Report (35=AE)
Gap Fill (35=4)

Chapter 6 Self-Certification Scenario

Scenario #1: Matching user can correctly subscribe for screened prices via MARKET_PRICE Model type.

Description:

A Matching API user should be able to correctly subscribe for screened prices from RFA using MARKET_PRICE model type. Screened prices can only be retrieved by passing the correct 4 char TCID mnemonic code, for ex. subscription for GBDUSD would be GBP=NNAG where NNAG is the PBC code assigned to client.

Self certification criterion:

Matching API user should connect to P2PS in test environment and subscribe for screened prices via MARKET_PRICE model type. Tester can verify correct processing of prices by subscribing for screened prices from RFA sample tool, provided by Thomson Reuters, and comparing the prices received from both sources. Thomson Reuters can provide additional MDFD users for logging via sample tool for this testing. User should also verify receiving Last traded price though this model type.

Scenario #2: Matching user can correctly subscribe for unscreened prices via MARKET_PRICE Model type.

Description:

A Matching API user should be able to correctly subscribe for unscreened prices from RFA using MARKET_PRICE model type. User need not send the 4 letter TCID while subscribing for unscreened prices.

Self certification criterion:

Matching API user should connect to P2PS in test environment and subscribe for unscreened prices via MARKET_PRICE model type. Tester can verify correct processing of prices by subscribing for unscreened prices from RFA sample tool, provided by Thomson Reuters, and comparing the prices received from both sources. User should also verify receiving Last traded price though this model type.

Scenario #3: Matching user can correctly subscribe for screened prices via MARKET_BY_PRICE Model type.

Description:

A Matching API user should be able to correctly subscribe for screened prices from RFA using MARKET_BY_PRICE model type. Screened prices can only be retrieved by passing the correct 4 char TCID mnemonic code.

Self certification criterion:

Matching API user should connect to P2PS in test environment and subscribe for screened prices via MARKET_BY_PRICE model type. Tester can verify correct processing of prices by subscribing for screened prices via RFA sample tool provided by Thomson Reuters, and comparing the prices received from both sources. User should also verify receiving aggregate prices and Last traded price though this model type.

Scenario #4: Matching user can correctly subscribe for unscreened prices via MARKET_BY_PRICE Model type.

Description:

A Matching API user should be able to correctly subscribe for unscreened prices from RFA using MARKET_BY_PRICE model type.

Self certification criterion:

Matching API user should connect to P2PS in test environment using his application and subscribe for unscreened prices via MARKET_BY_PRICE model type. Tester can verify correct processing of prices by subscribing for unscreened prices via RFA sample tool provided by Thomson Reuters, and comparing the prices received from both sources. User should also verify receiving Last traded price though this model type.

Scenario #5: Matching user can successfully reconnect to RFA by acquiring the RFA Token from active FIX session in the event of a FIX and RFA disconnect.

Description:

Though an RFA Token is valid for the duration of a trading session (which is from start of trading week to the end of trading week) but in the event of a FIX Session disconnect the user is provided with a new RFA Token through new FIX login. As long as the RFA connection is active RFA Token stays valid however if the Client loses RFA Connection then he is required to gain access to the RFA Token from active FIX Session other wise client's RFA application will have connectivity issues due the use of RFA Token from an expired session.

Self certification criterion:

Matching API user should connect to P2PS in test environment using his application and subscribe for unscreened prices via MARKET_BY_PRICE model type. User should disconnect his FIX session without disconnecting RFA Connection. RFA connection should continue to receive price updates even though the FIX session is disconnected. FIX session should relogin after 3 minutes or so. Tester should not observe any change in RFA feed while FIX session is disconnected or FIX users relogin. User should be able to subscribe for new instrument updates without a problem. RFA Client should disconnect his client application without dropping the FIX session and try relogin. RFA Client application will require RFA token from the new FIX session in order to establish connectivity with P2PS at this point since the old RFA token expires after both FIX and RFA sessions lose connectivity.

Scenario #6: Matching user should have ability to handle Test Request message.

Description:

The Matching FIX Gateway uses a Heartbeat interval of 4 secs. In the absence of any outgoing traffic for a period of 4 seconds both ends of the FIX link will be expected to generate an outgoing heartbeat. MFG will issue a Test Request message if a no message is received from client gateway in any given 6 secs interval, Client application is required to respond to this request with a Heart Beat message.

Self certification criterion:

It's not possible to manually create a Test Request message to test this scenario so client will have to make sure that their application is able to send a Heart Beat message if they receive a Test Request message from MFG.

Scenario #7: Matching user should have ability to handle Forced Logoff.

Description:

Matching administration has the ability to invoke a user logoff at any point in time. This is referred to as a forced logoff. A client session will receive a logout message (35=5) with tag 58 clearly indicating “Forced Log off by exchange”. In the event that MFG was disconnected from Matching Host while Match took place then a trade confirmation message will not be delivered to MFG and no TCR will be generated. Client will receive a call from Reuters Help Desk making them aware of unacknowledged Match and client should contact the counterparty to manual confirm the match accept the Match as per the Rule Book.

Self certification criterion:

It is not possible to create this scenario in test environment but client application should be able to handle “Forced logoff” message from Exchange.

Scenario #8: Matching user should have ability to handle MTM (trade/non trade mode) messages

Description:

Client application may receive a non trade (MTM:0) message during trading session if Matching is suffering from a technical difficulty or any other such reason.

Self certification criterion:

It is not possible to change Matching Host state in test environment however if client wants they can test (MTM:0) message on Friday 22 GMT when Host moves to non Trade mode.

Scenario #9: Credit Threshold alerts

Description:

Client application can receive credit threshold alert (<CTA>) user notification messages in the event that their credit utilization has gone down to 25% or lower.

Self certification criterion:

Client can request On-boarding team to reduce their credit limit so they can test receiving credit threshold alerts. These are very important notifications and in the event that client has exhausted all of their credit in Live environment then they will still be able to submit orders but these orders will not be Matched due to insufficient credit.

Chapter 7 User Notifications

User notifications are explained in detail in section 4.7.3 of Thomson Reuters Matching Interface User Guide. It is important for MAPI users to make sure that their applications are able to take appropriate action based on the nature of the notification from MFG. Client is required to update certification spreadsheet with the workflow that their application has to handle below notifications.

User Notification (35=CB)	Brief Description
User Credit Threshold notification	User receives credit threshold alert with Tag 58= “CTA:<TCID>:<9999> indicating user is running low on credit. User should escalate this to their Prime Bank.
STU Logout notice	User receives Tag 58=”STU:0” indicating that Special Ticket User is not logged in and new order requests will be rejected. This notification is only received by PBC users. User should raise this as an alert to their Prime Bank or call Thomson Reuters Help Desk.
STU Login notice	User receives Tag 58=”STU:1” indicating that Special Ticket User is logged in now and user can submit trade messages. This notification is only received by PBC users.
Matching Service Trade Status	User receives Tag 58=”MTM:0” to indicate Matching is in non Trade mode, and Tag 58=”MTM:1” to indicate Matching is in Trading mode
Start of Trading week countdown	User receives Tag 58=”SOTW:X” where X indicating how many minutes before trading starts.
End of Trading week countdown	User receives Tag 58=”EOTW:X” where X indicating how many minutes before end of trading.
Throttle Override imposed	User receives Tag 58=”GTRC1:60” indicating that global throttle rate has changed and user should adjust to new throttle limit.
Throttle Override removed	User receives Tag 58=”GTRC:0:0” indicating that global throttle limit has been removed.

Chapter 8 Completion and review Criterion

Upon completion of the certification test, MAPI on-boarding team will review the results as submitted by the client in “Matching API Certification” spreadsheet. MAPI on-boarding team will inform client via BizBuzz website whether they have passed conformance or they need to reattempt any failed test cases.

For more information

Send us a sales enquiry at
financial.thomsonreuters.com/sales

Read more about our products at
financial.thomsonreuters.com

Find out how to contact your local office
financial.thomsonreuters.com/locations

Document Version 1.6
Date of issue: 3 February 2012