

---

**Interface Specifications**  
**HKEX China Connect Central Gateway Platform**  
**FIX Drop Copy Protocol**

---

Version 1.1  
5 January 2018

## Table of Contents

<b>1. Document Control .....</b>	<b>4</b>
1.1 Change History .....	4
1.2 Abbreviations .....	4
<b>2. Introduction.....</b>	<b>5</b>
<b>3. Connectivity.....</b>	<b>6</b>
3.1 Session.....	6
3.2 Comp ID.....	6
3.3 IP Address and Port Numbers.....	6
3.4 Encryption .....	6
3.5 Password .....	7
3.6 Change Password .....	7
3.7 Failure and Recovery .....	7
<b>4. Session Management.....</b>	<b>8</b>
4.1 Establishing a FIX Session .....	8
4.2 Message Sequence Numbers .....	8
4.3 Heart Beat and Test Request.....	9
4.4 Terminating a FIX Session .....	9
4.5 Re-establishing a FIX Session .....	9
4.6 Sequence Reset .....	10
4.6.1 During a Session .....	10
4.6.2 When starting a new Session .....	10
4.6.2.1 Reset Initiated by the Client .....	10
4.6.2.2 Reset Initiated by the CCCG.....	10
4.7 Fault Tolerance .....	10
4.8 Checksum Validation .....	10
<b>5. Recovery.....</b>	<b>12</b>
5.1 General Message Recovery .....	12
5.2 Resend Request.....	12
5.3 Logon Message Processing – Next Expected Message Sequence .....	13
5.4 Possible Duplicates .....	13
5.5 Possible Resends .....	13
5.5.1 Client Initiated Messages.....	13
5.5.2 CCCG Initiated Messages .....	13
5.6 Gap Fills .....	14
5.7 Transmission of Missed Messages.....	14
<b>6. Service Description .....</b>	<b>15</b>
6.1 Data Types .....	15
6.2 Security Identification .....	15
6.3 Party Identification .....	15
6.4 Client Order ID.....	15
6.5 Order Handling .....	15
6.6 Order ID .....	15
6.7 Trade ID .....	15
6.8 Execution ID.....	16
6.9 Execution Reports.....	16
6.10 Message Rejection.....	16
6.11 Copy Message Indicator .....	16

---

6.12 Session Contents .....	16
<b>7. Message Definitions.....</b>	<b>18</b>
7.1 Supported Message Types.....	18
7.2 In-bound Messages .....	18
7.3 Out-bound Messages.....	18
7.4 Message Header .....	20
7.5 Message Trailer .....	20
7.6 Administrative Messages.....	21
7.6.1 Logon (A) .....	21
7.6.2 Logout (5) .....	21
7.6.3 Heartbeat (0).....	22
7.6.4 Test Request (1) .....	22
7.6.5 Resend Request (2) .....	22
7.6.6 Reject (3) .....	23
7.6.7 Sequence Reset (4).....	23
7.7 Business Messages – Execution Reports (8) .....	25
7.7.1 Order Registered .....	25
7.7.2 Order Accepted .....	26
7.7.2.1 Order Rejected .....	28
7.7.3 Order Cancelled .....	30
7.7.4 Order Cancelled – Unsolicited.....	32
7.7.5 Trade (Order Executed).....	34
7.8 Infrastructure .....	37
7.8.1 Business Message Reject (j) .....	37
<b>8. Data Dictionary .....</b>	<b>38</b>
8.1 Session Level.....	38
8.2 Business Level.....	42
<b>Appendices .....</b>	<b>46</b>
<b>A. Password Policy .....</b>	<b>47</b>

## 1. Document Control

### 1.1 Change History

Version Number	Issue Date	Section Number	Status
1.0	1 December 2016	All	First Version
1.1	5 January 2018	Section 8.2 – update Data Dictionary	Updated Version

### 1.2 Abbreviations

Terms and abbreviations referred to in this document are tabulated as follows:

BSS	Broker Supplied System
CCASS	Central Clearing and Settlement System
CCCG	China Connect Central Gateway
CSC	China Stock Connect System
EP	HKEX Cash Trading Exchange Participant
FIX	Financial Information eXchange
OCG	Orion Central Gateway - the market access platform for HKEX Securities Trading
SPSA	Special Segregated Account
SSE	Shanghai Stock Exchange
SZSE	Shenzhen Stock Exchange

---

## 2. Introduction

This document describes the FIX interface protocol of the HKEX China Connect Central Gateway (“CCCG”), the market access platform for the China Connect trading service.

The CCCG provides an application level interface to HKEX China Stock Connect System (“CSC”) for drop copy purpose.

The CCCG mostly follows the interface and event model used in Orion Central Gateway (“OCG”), the market access platform for Hong Kong cash market.

The terminology used, message format, message flow and event models described throughout this document are as per FIX 5.0 SP2 protocol specifications with HKEX specific extension packs.

---

HKEX and/or its subsidiaries endeavour to ensure the accuracy and reliability of the information provided, but do not guarantee its accuracy and reliability and accept no liability (whether in tort or contract or otherwise) for any loss or damage arising from any inaccuracy or omission or from any decision, action or non-action based on or in reliance upon information contained in this document.

No part of this document may be copied, distributed, transmitted, transcribed, stored in a retrieval system, translated into any human or computer language, or disclosed to third parties without written permission from HKEX.

HKEX reserves the right to amend any details in this document at any time, without notice.

## 3. Connectivity

### 3.1 Session

Exchange Participants connect their broker supplied systems (“BSS” or “Client”) to the CCCG via subscription to one or more CCCG “Session”. This connection is a standard TCP/IP point-to-point connection.

EPs are expected to pre-register at least one IP address using which a client from their end would establish a Trading Session with the CCCG. EPs can optionally pre-register up to 3 additional IP addresses for each Session, for backup purposes.

A session can be established from only one of the pre-registered IP addresses for that session. EPs can pre-register any given IP address for more than one session such that the same BSS can be used to connect to CCCG through one or more sessions.

### 3.2 Comp ID

The client should use the Comp ID (a unique session identifier) provided by HKEX for each session in order to connect to the CCCG. A single client may have multiple connections to the CCCG i.e., multiple FIX sessions, each with its own Comp ID.

The messages sent to the CCCG should contain the Comp ID assigned to the client in the field Comp ID in the header section.

CompID of the CCCG, as set for TargetCompID of the client initiated messages & SenderCompID of the CCCG initiated messages is “**HKEXCCCC**”.

### 3.3 IP Address and Port Numbers

A FIX client is expected to hunt for a connection point (IP Address and Port Number) within a pool of connection points provided by HKEX for the following CCCG service.

At the time of the first logon for the day, the client is expected to attempt to connect to the first connection point within the pool. If this first attempt is not succeeded or if the connection is accepted but abruptly dropped by the CCCG without any message before the Logon message can be exchanged, then the client is expected to continue attempting the connection points in the pool one by one in sequence until a successful connection and a FIX session can be established. If a session could not be established having attempted all the connection points in the pool, the client is expected to cycle back to the first connection point then and retry.

During a reconnection within the same day the client can attempt to connect to the identified connection point (IP and Port) through the connection hunting process in order to re-establish the session, before moving to the other connection point.

HKEX will provide this pool of IP address and Port numbers through a separate medium.

### 3.4 Encryption

The CCCG interface protocol expects password and new password be encrypted when they are sent in the Logon message from the client to the CCCG.

To encrypt the password, the client is expected to use a 2048-bit RSA ([http://en.wikipedia.org/wiki/RSA\\_\(algorithm\)](http://en.wikipedia.org/wiki/RSA_(algorithm))) public key circulated (through a different medium) by HKEX. The binary output of the RSA encryption must be represented in Big Endian (Padding scheme is PKCS #1) and then converted to an alphanumeric value by means of standard base-64 encoding (<http://en.wikipedia.org/wiki/Base64>) when communicating with the CCCG.

---

HKEX may periodically renew the public key used by the client and after a public key renewal; a client may continue to use the old key for a limited grace period. Both keys may be used during this time.

### 3.5 Password

The client should specify their password in the EncryptedPassword (1402) field of the Logon (35=A) message. This password must be in encrypted form.

The status of the password (i.e. whether it is accepted or rejected) will be specified in the SessionStatus (1409) field of the Logon (35=A) sent by the CCCG to confirm the establishment of a FIX connection.

Repeated failures in password validation may force HKEX to lock the client; the EP is expected to contact HKEX to unlock the client and reset the password.

### 3.6 Change Password

Each new Comp ID will be assigned a password on registration. The client is expected to change the password upon first logon whenever a password is (re)issued by HKEX.

Password change request can be made together with Logon (35=A) request. The client should specify the encrypted new password in the EncryptedNewPassword (1404) field and the current encrypted password in the EncryptedPassword (1402) field.

The new password must comply with HKEX password policy (refer to [Appendix A](#)). The status of the new password (i.e. whether it is accepted or rejected) will be specified in the SessionStatus (1409) field of the Logon sent by the CCCG to confirm the establishment of a FIX connection. The new password will, if accepted, be effective for subsequent logins.

The client is required to change the password periodically. HKEX will set expiry duration for the password without exemption; a reminder will be sent indicating that the password is to about to expire, through the Text (58) field in the Logon (35=A) response. Once the password has expired for a client, that client will not be allowed to logon, and the EP is required to contact HKEX to unlock and reset the client password.

### 3.7 Failure and Recovery

The system has been designed with fault tolerance and disaster recovery technology that ensures that trading should continue in the unlikely event of a process or server outage.

If the client is unexpectedly disconnected, it should attempt to re-connect to the CCCG before attempting to follow the connection mechanism described in [Section 3.3](#).

---

## 4. Session Management

### 4.1 Establishing a FIX Session

Each client will use one of the IP address and port sets provided to establish a TCP/IP connection with the CCCG. The client will initiate a FIX session at the start of each trading day by sending the Logon (35=A) message.

A client must identify itself by providing its Comp ID in SenderCompID (49) field. The CCCG will validate this SenderCompID, password and IP address of the client.

Once the client is authenticated, the CCCG will respond with a Logon (35=A) message with SessionStatus (1409) set to 0 = Session Active. If the client's Logon message included the field NewEncryptedPassword and the client is authenticated, the CCCG will respond with a Logon message with SessionStatus (1409) set to 1 = Session Password Changed.

The client must wait for the Logon from the CCCG before sending additional messages. If additional messages are received from the client before the exchange of Logon messages, the TCP/IP connection with the client will be disconnected.

If a logon attempt fails for the following reasons, the CCCG will send a Logout (35=5) or a Reject (35=3) and then terminate the session:

- Password failure
- Comp ID is locked
- Logon is not permitted during this time

For all other reasons, including the following, the CCCG will terminate the session without sending a Logout or Reject:

- Invalid Comp ID or IP address

If during a logon of a Comp ID, the CCCG receives a second connection attempt while a valid FIX session is already underway for that same Comp ID, the CCCG will terminate both connections without sending a Logout or Reject message.

Inbound message sequence number will not be incremented if the connection is abruptly terminated due to the logon failure.

If a session level failure occurs due to a message sent by the client that contains a sequence number that is less than what is expected and the PossDup (43) is not set to Y = Yes, then the CCCG will send a Logout message and terminate the FIX connection. In this scenario the inbound sequence number will not be incremented.

If the CCCG does not respond to the session initiation (client initiated Logon message), the client is expected to wait for a time period of **60 seconds** prior to terminating the connection. The client is expected to retry session initiation after an elapsed time period of **60 seconds**.

If a client is disconnected abruptly or via a Logout message (35=5) from the CCCG, the client is expected to wait for a time period of **10 seconds** prior to reconnecting to the CCCG.

### 4.2 Message Sequence Numbers

As outlined in the FIX protocol, the client and CCCG will each maintain a separate and independent set of incoming and outgoing message sequence numbers. Sequence numbers should be initialized to 1 (one) at the start of the day and be incremented throughout the session. Either side of a FIX session will track the:



- NextExpectedMsgSeqNum (789) (starting at 1)
- Next To Be Sent Message Sequence number (starting at 1); with respect to the contra-party.

Monitoring sequence numbers will enable parties to identify and react to missed messages and to gracefully synchronize applications when reconnecting during a FIX session.

Any message sent by either side of a FIX session will increment the sequence number unless explicitly specified for a given message type.

If any message sent by one side of a FIX session contains a sequence number that is LESS than the NextExpectedMsgSeqNum (789) then the other side of this session is expected to send a Logout message and terminate the FIX connection immediately, unless the PossDup flag is set to Y = Yes

A FIX session will not be continued to the next trading day. Both sides are expected to initialize (reset to 1) the sequence numbers at the start of each day. At the start of each trading day if the client starts with a sequence number greater than 1 then the CCCG will terminate the session immediately without any further exchange of messages.

### 4.3 Heart Beat and Test Request

The client and the CCCG will use the Heartbeat (35=0) message to monitor the communication line during periods of inactivity and to verify that the interfaces at each end are available.

The heartbeat interval as in HeartBtInt (108) is recommended to be set as **20 Seconds**.

The CCCG will send a Heartbeat anytime it has not transmitted a message for the heartbeat interval. The client is expected to employ the same logic.

If the CCCG detects inactivity for a period longer than **3 heartbeat intervals**, it will send a Test Request message to force a Heartbeat from the client. If a response to the Test Request (35=1) is not received within a reasonable transmission time (recommended being an elapsed time equivalent to 3 Heartbeat intervals), the CCCG will send a Logout (35=5) and break the TCP/IP connection with the client. The client is expected to employ similar logic if inactivity is detected on the part of the CCCG.

### 4.4 Terminating a FIX Session

Session termination can be initiated by either the CCCG or the client by sending a Logout message. Upon receiving the Logout request, the contra party will respond with a Logout message signifying a Logout reply. Upon receiving the Logout reply, the receiving party will terminate the connection.

If the contra-party does not reply with either a Resend Request or a Logout reply, the Logout initiator should wait for **60 seconds** prior to terminating the connection.

The client is expected to terminate each FIX connection at the end of each trading day before the CCCG service is shut down. However, all open FIX connections will be terminated (a Logout message will be sent) by the CCCG when its service is shut down. Under exceptional circumstances the CCCG may initiate the termination of a connection during the trading day by sending the Logout message.

If, during the exchange of Logout messages, the client or the CCCG detects a sequence gap, it should send a Resend Request.

### 4.5 Re-establishing a FIX Session

If a FIX connection is terminated during the trading day it may be re-established via an exchange of Logon messages.

Once the FIX session is re-established, the message sequence numbers will continue from the last message successfully transmitted prior to the termination.

## 4.6 Sequence Reset

The sequence reset could be done in two modes;

### 1. Gap-fill mode:

Gap-fill mode is expected to be used by one side when skipping session level messages which can be ignored by the other side.

### 2. Reset mode:

Reset mode is used only in exceptional scenarios to indicate a reset in the session's starting sequence number. This mode can ONLY be used by the CCCG. Client initiated resets would be rejected by the CCCG.

Following scenarios exist:

### 4.6.1 During a Session

The CCCG and the client may use the Sequence Reset (35=4) message in Gap Fill mode if either side wishes to increase the expected incoming sequence number of the other party.

The CCCG may also use the Sequence Reset message in Sequence Reset mode if it wishes to increase the expected incoming sequence number of the other party. The Sequence Reset mode should only be used to recover from an emergency situation. It should not be relied upon as a regular practice.

### 4.6.2 When starting a new Session

#### 4.6.2.1 Reset Initiated by the Client

Reset sequence (reset to 1) through the Logon Message will not be facilitated by the CCCG. In order to reset the sequence (reset to 1), the client should manually inform the HKEX Operations Desk.

#### 4.6.2.2 Reset Initiated by the CCCG

The system has been designed with fault tolerance and disaster recovery technology that should ensure that the CCCG retains its incoming and outgoing message sequence numbers for each client in the unlikely event of an outage. However, the client is required to support a manual request by HKEX to initialize sequence numbers prior to the next login attempt.

## 4.7 Fault Tolerance

After a failure on client side or on CCCG side, the client is expected to be able to continue the same session.

In case of a catastrophic scenario, the FIX gateway will restart from a higher sequence number considering the previous session or may start from sequence number 1.

The client and the CCCG are expected to negotiate on the NextExpectedMsgSeqNum (789) and Next To Be Received Sequence number through an alternate medium prior to initiating the new session (Manually setting the sequence number for both ends after having a direct communication with the client).

## 4.8 Checksum Validation

The CCCG performs a checksum validation on all incoming messages. Incoming messages that fail the checksum validation will be rejected and the connection will be dropped by the CCCG without sending a logout.

Conversely, in case of a checksum validation failure, the client is expected to drop the connection and take any appropriate action before reconnecting.

Messages that fail the checksum validation should not be processed.

## 5. Recovery

### 5.1 General Message Recovery

- A gap is identified when an incoming message sequence number is found to be greater than NextExpectedMsgSeqNum (789).
- The Resend Request will indicate the BeginSeqNo (7) and EndSeqNo (16) of the message gap identified and when replying to a Resend Request, the messages are expected to be sent strictly honouring the sequence.
- If messages are received outside of the BeginSeqNo and EndSeqNo, then the recovering party is expected to queue those messages until the gap is recovered.

During the message recovery process, the recovering party will increment the *Next Expected Sequence* number accordingly based on the messages received. If messages applicable to the message gap are received out of sequence then the recovering party will drop these messages.

- The party requesting the Resend Request can specify "0" in the EndSeqNo to indicate that they expect the sender to send ALL messages starting from the BeginSeqNo.

In this scenario, if the recovering party receives messages with a sequence greater than the *BeginSeqNo*, out of sequence, the message will be ignored.

- Administrative messages such as Sequence Reset, Heartbeat and Test Request which can be considered irrelevant for a retransmission could be skipped using the Sequence Reset message in gap-fill mode.

Note that the CCCG expects the client to skip Sequence Reset messages when replying to a Resend Request at all times.

- When resending messages, the CCCG would use either PossDup or PossResend flag to indicate whether the messages were retransmitted earlier.

If PossDup flag is set Y = Yes, it indicates that the same message with the given sequence number with the same business content may have been transmitted earlier.

In the case where PossResend flag is set to Y = Yes, it indicates that the same business content may have been transmitted previously but under the different message sequence number. In this case business contents needs to be processed to identify the resend. For example, in Execution Reports the ExecID (17) may be used for this purpose.

### 5.2 Resend Request

The client may use the Resend Request message to recover any lost messages. This message may be used in one of three modes:

- (i) To request a single message. The BeginSeqNo and EndSeqNo should be the same.
- (ii) To request a specific range of messages. The BeginSeqNo should be the first message of the range and the EndSeqNo should be the last of the range.
- (iii) To request all messages after a particular message. The BeginSeqNo should be the sequence number immediately after that of the last processed message and the EndSeqNo should be zero (0).

### 5.3 Logon Message Processing – Next Expected Message Sequence

The session initiator should supply the NextExpectedMsgSeqNum (789) the value next expected from the session acceptor in MsgSeqNum (34). The session acceptor should validate the logon request including that NextExpectedMsgSeqNum (789) does not represent a gap. It then constructs its logon response with NextExpectedMsgSeqNum (789) containing the value next expected from the session initiator in MsgSeqNum (34) having incremented the number above the logon request if that was the sequence expected.

The session initiator must wait until the logon response is received in order to submit application messages. Once the logon response is received, the initiator must validate that NextExpectedMsgSeqNum (789) does not represent a gap.

In case of gap detection from either party (lower than the next to be assigned sequence) recover all messages<sup>4</sup> from the last message delivered prior to the logon through the specified NextExpectedMsgSeqNum (789) sending them in order, then gap fill over the sequence number used in logon and proceed sending newly queued messages with a sequence number one higher than the original logon.

Neither side should generate a resend request based on MsgSeqNum (34) of the incoming Logon message but should expect any gaps to be filled automatically by following the Next Expected Sequence processing described above.

Note that indicating the NextExpectedMsgSeqNum (789) in the Logon (35=A) is mandatory.

### 5.4 Possible Duplicates

The CCCG handles possible duplicates according to the FIX protocol. The client and the CCCG use the PossDupFlag (43) field to indicate that a message may have been previously transmitted with the same MsgSeqNum (34).

### 5.5 Possible Resends

#### 5.5.1 Client Initiated Messages

The CCCG does not handle possible resends for the client-initiated messages and the message will be processed without considering the value in the PossResend (97) field.

#### 5.5.2 CCCG Initiated Messages

The CCCG may use the PossResend (97) field to indicate that an application message may have already been sent under a different MsgSeqNum (34). The client should validate the contents (e.g., ExecID (17)) of such a message against those of messages already received during the current trading day to determine whether the new message should be ignored or processed.

<sup>1</sup> During the period where the CCCG is resending messages to the client, the CCCG does not allow another Resend Request from the client. If a new Resend Request is received during this time, the CCCG will terminate the session immediately without sending the Logout message.

---

## 5.6 Gap Fills

The following messages are expected to be skipped using gap-fills when being retransmitted:

1. Logon
2. Logout
3. Heartbeat
4. Test Request
5. Resent Request
6. Sequence Reset

All other messages are expected to be replayed within a retransmission.

## 5.7 Transmission of Missed Messages

The Execution Report & Trade Capture Reports messages generated during a period when a client is disconnected from the CCCG will be sent to the client when it next reconnects. In the unlikely event the disconnection was due to an outage of the CCCG, all such messages will include a PossResend (97) set to Y = Yes.

## 6. Service Description

### 6.1 Data Types

Data Types used are based on the published standard FIX specifications.

### 6.2 Security Identification

Instruments will be identified using the SecurityID (48) field. It is required to specify SecurityIDSource (22) as Exchange Symbol (8) and SecurityExchange (207) as either the following two ISO MICs:

- **XSSC** = ISO Market Segment Code for Stocks listed in Shanghai Stock Exchange under Shanghai – Hong Kong Stock Connect
- **XSEC** = ISO Market Segment Code for Stocks listed in Shenzhen Stock Exchange under Shenzhen – Hong Kong Stock Connect

Security ID (48) will carry the Security Code of an instrument as assigned by the home market.

### 6.3 Party Identification

Party Identification is defined as follows:

ID	Description	FIX Tag and Value
Broker ID	Identifier of the member the interest is submitted under	PartyID (448) PartyRole (452)=1 PartyIDSource=D
Contra Broker ID	Identifier of the Contra member the interest is submitted under	PartyID (448) PartyRole (452)=17 PartyIDSource=D
BS User ID	The location ID of the member as referred to by Broker ID	PartyID (448) PartyRole (452)=75 PartyIDSource=D
SPSA Investor ID	Investor ID as assigned by CCASS for a Special Segregated Account (SPSA)	PartyID (448) PartyRole (452)=5 PartyIDSource=D

### 6.4 Client Order ID

In a drop copy message ClOrdID (11) field will be provided only if this field was present in original message submitted through the Trading interface. The same rule is applicable to OrigClOrdID (41).

### 6.5 Order Handling

Order book updates such as new orders, order cancels, resultant trades etc., will be published to the drop copy clients using Execution Report (35=8) message.

### 6.6 Order ID

The CCCG will provide the OrderID (37) field to identify an order in all the drop copy messages associated with order handling.

### 6.7 Trade ID

The CCCG will use the TrdMatchID (880) field to identify a trade.

TrdMatchID (880) field value is derived based on the corresponding value provided by the home market.

## 6.8 Execution ID

The CCCG will use the ExecID (17) field to affix a unique identifier for each Execution Report. ExecID value will be unique per trading day.

## 6.9 Execution Reports

The Execution Report (35=8) message is used to communicate many different events to the drop copy client. The events are differentiated by the value in the ExecType (150) field as outlined below:

ExecType (150)	Description	OrdStatus (39)
A = Pending New	<b>Order Pending to be accepted by the Home Market</b> Indicates that a new order has been accepted by CSC and forwarded to the home market.	<ul style="list-style-type: none"> <li>A = Pending New</li> </ul>
0 = New	<b>Order Accepted</b> Indicates that a new order has been accepted by the home market.	<ul style="list-style-type: none"> <li>0 = New</li> </ul>
8 = Rejected	<b>Order Rejected</b> Indicates that an order has been rejected (by the home market). The reason for the rejection is specified in the field OrdRejReason (103).	<ul style="list-style-type: none"> <li>8 = Rejected</li> </ul>
F = Trade	<b>Order Executed (Trade)</b> Indicates that an order has been partially or fully filled. The execution details (e.g., price and quantity) are specified.	<ul style="list-style-type: none"> <li>1 = Partially Filled</li> <li>2 = Filled</li> </ul>
4 = Cancelled	<b>Order Cancelled</b> Indicates that an Order Cancel Request has been accepted and successfully processed.  This message can also be sent unsolicited in which case the Execution Report may include the ExecRestatementReason field to indicate the reason for cancellation; OrigCLOrdID field will not be provided.	<ul style="list-style-type: none"> <li>4 = Cancelled</li> </ul>

## 6.10 Message Rejection

If an incoming message violates any message level validations such as data type mismatches or message structure mismatches, the messages are expected to be rejected back to the sender using a Reject (35=3) message (applicable to both the client and the CCCG) and Business Message Reject (35=j) from the CCCG.

## 6.11 Copy Message Indicator

CopyMsgIndicator (797) field will be set to Y = Yes for each message sent through the drop copy sessions in order to indicate that the given message is a drop copy of another message.

## 6.12 Session Contents



A drop copy connection is configured to receive a copy of all the Execution Reports (35=8) generated for the Broker ID(s) assigned to the drop copy session, irrespective of the access platform (e.g., CCOG Session, CCCG Trading service) used by the given Broker ID for trading. However, some of the rejection messages related to order and order cancel will not be part of the drop copy.

The drop copy client may have the privilege to subscribe to one of the following drop copy options:

- **Trades Only**

Under 'Trades only' option, the client will have access to Execution Reports related to trades.

- **Orders & Trades**

Under 'Orders & Trades' option, the client will have access to all order related Execution Reports.

A drop copy client must choose the required option when subscribing to the drop copy service with HKEX market operations. This option can't be changed intra-day.

The following table describes the contents for the two subscription options:

<b>Business Message Type</b>	<b>Order and Trades Option</b>	<b>Trades Only Option</b>
Execution Report (35=8)		
Exec Type = Pending New	√	X
Exec Type = New	√	X
Exec Type = Reject <i>(i.e., the New Order request is rejected before forwarding to home market)</i>	X	X
Exec Type = Reject <i>(i.e., the New Order request is rejected by the home market)</i>	√	X
Exec Type = Trade	√	√
Exec Type = Pending Cancel	X	X
Exec Type = Cancel	√	X
Exec Type = Cancel Reject	X	X

## 7. Message Definitions

### 7.1 Supported Message Types

All supported message types initiated by the client or the CCCG:

#	Message	Message Type	Usage
1.	Heartbeat	0	Allows the client and the CCCG to exercise the communication line during periods of inactivity and verify that the interfaces at each end are available.
2.	Test Request	1	Allows the client or the CCCG to request a response from the other party if inactivity is detected.
3.	Resend Request	2	Allows for the recovery of messages lost during a malfunction of the communications layers.
4.	Reject	3	Used to reject a message that does not comply with session level validations.
5.	Sequence Reset	4	Allows the client or the CCCG to increase the expected incoming sequence number of the other party.
6.	Logon	A	Allows the client and the CCCG to establish a FIX session.
7.	Logout	5	Allows the client and the CCCG to terminate a FIX session.
8.	Business Message Reject	j	Indicates that an application message could not be processed
9.	Execution Report	8	Indicates one of the following: <ul style="list-style-type: none"> <li>▪ Order Registered</li> <li>▪ Order Accepted</li> <li>▪ Order Rejected</li> <li>▪ Order Cancelled</li> <li>▪ Trade</li> </ul>

**Note:** HKEX may add new data fields and/or new messages to serve new business requirements in the future and BSS should be designed to adopt these interface changes in a flexible approach.

### 7.2 In-bound Messages

Only Session-level messages are expected.

### 7.3 Out-bound Messages

In addition to the Session-level messages, the following lists the messages that are sent by the CCCG to the client:

1. Execution Report (35=8)

## 7.4 Message Header

All messages exchanged between the client and the CCCG have the following standard message header:

FIX Tag	Field Name	Required?	Description
8	BeginString	Y	Always FIXT.1.1
9	BodyLength	Y	Number of bytes after this field up to and including the delimiter immediately preceding the CheckSum.
35	MsgType	Y	Message type.
49	SenderCompID	Y	Comp ID of the party sending the message.
56	TargetCompID	Y	Comp ID of the party the message is sent to.
34	MsgSeqNum	Y	Sequence number of the message.
43	PossDupFlag	N	Indicates whether the message was previously transmitted under the same MsgSeqNum (34). Absence of this field is interpreted as original transmission (N).
97	PossResend	N	Indicates whether the message was previously transmitted under a different MsgSeqNum (34). Absence of this field is interpreted as original transmission (N).
52	SendingTime	Y	Time the message was transmitted.
122	OrigSendingTime	N	Time the message was originally transmitted. If the original time is not available, this should be the same value as SendingTime (52). Required if PossDupFlag (43) is Possible Duplicate (Y).
1128	ApplVerID	N	Version of FIX used in the message: <ul style="list-style-type: none"> <li>▪ 9 = FIX50SP2</li> </ul> Required if the message is generated by the CCCG.

## 7.5 Message Trailer

All messages exchanged between the client and the CCCG have the following standard message trailer:

FIX Tag	Field Name	Required?	Description
10	Checksum	Y	Standard check sum described by FIX protocol. Always last field in the message; i.e. serves, with the trailing <SOH>, as the end-of-message delimiter. Always defined as three characters.

## 7.6 Administrative Messages

### 7.6.1 Logon (A)

This message is initiated by the client and the CCCG may respond with the same message.

FIX Tag	Field Name	Required?	Description
<b>Message Header</b>			
35	MsgType	Y	A = Logon
<b>Message Body</b>			
98	EncryptMethod	Y	Method of encryption. <ul style="list-style-type: none"> <li>0 = None/Other</li> </ul>
108	HeartBtInt	Y	Indicates the heartbeat interval in seconds.
789	NextExpectedMsgSeqNum	Y	Next expected MsgSeqNum value to be received.
1400	EncryptedPasswordMethod	N	Enumeration defining the encryption method used to encrypt password fields: <ul style="list-style-type: none"> <li>101 = RSA</li> </ul>
1402	EncryptedPassword	N	Encrypted password – encrypted via the method specified in EncryptedPasswordMethod (1400)
1404	EncryptedNewPassword	N	Encrypted new password – encrypted via the method specified in EncryptedPasswordMethod (1400)
1409	SessionStatus	N	Status of the FIX session. Required if the message is generated by the CCCG.
1137	DefaultApplVerID	Y	Default version of FIX messages used in this session: <ul style="list-style-type: none"> <li>9 = FIX50SP2</li> </ul>
464	TestMessageIndicator	N	Can be used to specify that this FIX session will be sending and receiving "test" vs. "production" messages Only applicable for Logon (35=A) message from the CCCG to the client.
58	Text	N	Text field will be used to convey the number of days to password expiry when the CCCG replies with a Logon (35=A) message upon a successful logon attempt.
<b>Message Trailer</b>			

Password (1402) must be present in the Logon message (35=A) initiated by the client.

### 7.6.2 Logout (5)

This message can be initiated by both client and the CCCG.

FIX Tag	Field Name	Required?	Description
<b>Message Header</b>			
35	MsgType	Y	5 = Logout
<b>Message Body</b>			

1409	SessionStatus	N	Status of the FIX session. Required if the message is generated by the CCCG.
58	Text	N	Textual reason for the logout.
<a href="#">Message Trailer</a>			

### 7.6.3 Heartbeat (0)

This message can be initiated by both client and CCCG.

FIX Tag	Field Name	Required?	Description
<a href="#">Message</a> Header			
35	MsgType	Y	0= Heartbeat
<b>Message Body</b>			
112	TestReqID	N	Required if the heartbeat is a response to a TestRequest (35=1). The value in this field should echo the TestReqID (112) received in the TestRequest .
<a href="#">Message Trailer</a>			

### 7.6.4 Test Request (1)

This message can be initiated by both the client and the CCCG.

FIX Tag	Field Name	Required?	Description
<a href="#">Message</a> Header			
35	MsgType	Y	1= Test Request
<b>Message Body</b>			
112	TestReqID	Y	Identifier included in Test Request message to be returned in resulting Heartbeat.
<a href="#">Message Trailer</a>			

### 7.6.5 Resend Request (2)

This message can be initiated by both client and the CCCG.

FIX Tag	Field Name	Required?	Description
<a href="#">Message</a> Header			
35	MsgType	Y	2= Resend Request
<b>Message Body</b>			
7	BeginSeqNo	Y	Sequence number of the first message expected to be resent.

16	EndSeqNo	Y	Sequence number of the last message expected to be resent. This may be set to 0 to request the sender to transmit ALL messages starting from BeginSeqNo (7).
<a href="#">Message Trailer</a>			

### 7.6.6 Reject (3)

This message is initiated by the CCCG.

FIX Tag	Field Name	Required?	Description
<a href="#">Message Header</a>			
35	MsgType	Y	3= Reject
<b>Message Body</b>			
45	RefSeqNum	Y	Sequence number of the message which caused the rejection
371	RefTagID	N	If a message is rejected due to an issue with a particular field its tag number will be indicated.
372	RefMsgType	N	Message type of the rejected message.
373	SessionRejectReason	N	Code specifying the reason for the reject: <ul style="list-style-type: none"> <li>▪ 0 = Invalid Tag Number</li> <li>▪ 1 = Required Tag Missing</li> <li>▪ 2 = Tag not defined for this message</li> <li>▪ 3 = Undefined tag</li> <li>▪ 4 = Tag specified without a value</li> <li>▪ 5 = Value is incorrect (out of range) for this tag</li> <li>▪ 6 = Incorrect data format for value</li> <li>▪ 9 = ComplID problem</li> <li>▪ 10 = Sending Time Accuracy problem</li> <li>▪ 11 = Invalid Msg Type</li> <li>▪ 13 = Tag appears more than once</li> <li>▪ 15 = Repeating group fields out of order</li> <li>▪ 16 = Incorrect NumInGroup count for repeating group</li> <li>▪ 18 = Invalid/Unsupported Application Version</li> <li>▪ 99 = Other</li> </ul>
58	Text	N	Text specifying the reason for the rejection.
<a href="#">Message Trailer</a>			

### 7.6.7 Sequence Reset (4)

This message can be initiated by both client and the CCCG.

FIX Tag	Field Name	Required?	Description
<a href="#">Message Header</a>			
35	MsgType	Y	4= Sequence Reset

<b>Message Body</b>			
36	NewSeqNo	Y	Sequence number of the next message to be transmitted.
123	GapFillFlag	N	Mode in which the message is being used. Absence of this field is interpreted as Sequence Reset (N).
<a href="#"><u>Message Trailer</u></a>			



## 7.7 Business Messages – Execution Reports (8)

### 7.7.1 Order Registered

The CCG will send this execution report once the new order is registered in CSC and forwarded to the home market for processing.

FIX Tag	Field Name	Required?	Description
<b>Message Header</b>			
35	MsgType	Y	8 = Execution Report
<b>Message Body</b>			
11	ClOrdID	Y	Client specified identifier of the order.
37	OrderID	Y	Order ID as assigned by CSC.
17	ExecID	Y	Unique Execution ID assigned for each Execution Report generated.
<b>Component Block &lt;Parties&gt;</b>			
453	NoPartyIDs	Y	Number of party identifiers. The value in this field can be 1 or 2 or 3.
→	448	PartyID	ID of the Broker client submitting the order: Broker ID/ SPSA Investor ID/BS User ID.
→	447	PartyIDSource	Defines the naming convention used in specifying the PartyID (448): <ul style="list-style-type: none"> <li>▪ D = Proprietary/Custom Code</li> </ul>
→	452	PartyRole	Role of the specified PartyID (448): <ul style="list-style-type: none"> <li>▪ 1 = Executing Firm (Broker ID that submitted the order)</li> <li>▪ 5 = Investor ID (SPSA Investor ID)</li> <li>▪ 75 = Location ID (BS User ID)</li> </ul>
<b>End Component Block</b>			
<b>Component Block &lt;Instrument&gt;</b>			
48	SecurityID	Y	Instrument identifier
22	SecurityIDSource	Y	Identifies the source of the SecurityID (48). Required if: SecurityID (48) is specified
207	SecurityExchange	N	The market which is used to identify the security. Required if: SecurityIDSource (22) = 8 (Exchange Symbol).
<b>End Component Block</b>			
40	OrdType	N	Order type applicable to the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>
59	TimeInForce	N	Time qualifier of the order: <ul style="list-style-type: none"> <li>▪ 0 = Day (Default)</li> </ul> Absence of this field is interpreted as 0 = Day.

54	Side	Y	Side of the order <ul style="list-style-type: none"> <li>▪ 1 = Buy</li> <li>▪ 2 = Sell</li> <li>▪ 5 = Sell Short</li> </ul>
38	OrderQty	N	Total order quantity of the order
44	Price	N	Price of order. Required if: OrdType (40) = 2 = Limit.
60	TransactTime	Y	The time at which the particular message was generated.
528	OrderCapacity	N	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ A = Agency</li> <li>▪ P = Principal</li> </ul>
1090	MaxPriceLevels	N	Maximum number of price levels to trade through. Applicable if: Order Type (40) = 2 (Limit). If present, this should be set as 1.
77	PositionEffect	N	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>▪ C = Close</li> </ul> Applicable only if: Side (54) = 1 (Buy) to indicate covering a short sell.
39	OrdStatus	Y	Order Status after applying the transaction that is being communicated: <ul style="list-style-type: none"> <li>▪ A = Pending New</li> </ul>
150	ExecType	Y	Execution Type that indicates the reason for the generation of the Execution Report: <ul style="list-style-type: none"> <li>▪ A = Pending New</li> </ul>
14	CumQty	Y	Cumulative execution size
151	LeavesQty	Y	Open order quantity
58	Text	N	Free Text
1093	LotType	N	Lot Type of the order: <ul style="list-style-type: none"> <li>▪ 2 = Round Lot</li> </ul>
797	CopyMsgIndicator	Y	Indicates whether or not this message is a drop copy of another message. Always set to Y.

[Message Trailer](#)

Notes:

1. If LotType (1093) is absent then this execution report carries a board (i.e., round) lot order.

### 7.7.2 Order Accepted

The CCG will send this execution report once a new order, which has already been registered (refer to [Section 7.7.1](#)) is accepted by the home market.

FIX Tag	Field Name	Required?	Description
<b>Message Header</b>			
35	MsgType	Y	8 = Execution Report
<b>Message Body</b>			
11	ClOrdID	N	Client specified identifier of the order.
37	OrderID	Y	Order ID as assigned by the exchange.
17	ExecID	Y	Unique Execution ID assigned for each Execution Report generated.
<b>Component Block &lt;Parties&gt;</b>			
453	NoPartyIDs	Y	Number of party identifiers. The value in this field can be 1 or 2 or 3.
→	448	PartyID	ID of the Broker client submitting the order: Broker ID/SPSA Investor ID/BS User ID.
→	447	PartyIDSource	Defines the naming convention used in specifying the PartyID (448): <ul style="list-style-type: none"> <li>▪ D = Proprietary/Custom Code</li> </ul>
→	452	PartyRole	Role of the specified PartyID (448): <ul style="list-style-type: none"> <li>▪ 1 = Executing Firm (Broker ID that submitted the order)</li> <li>▪ 5 = Investor ID (SPSA Investor ID)</li> <li>▪ 75 = Location ID (BS User ID)</li> </ul>
<b>End Component Block</b>			
<b>Component Block &lt;Instrument&gt;</b>			
48	SecurityID	Y	Instrument identifier
22	SecurityIDSource	Y	Identifies the source of the SecurityID (48): <ul style="list-style-type: none"> <li>▪ 8 = Exchange Symbol</li> </ul> Required if: SecurityID (48) is specified
207	SecurityExchange	N	The market which is used to identify the security: Required if: SecurityIDSource (22) = 8 (Exchange Symbol).
<b>End Component Block</b>			
40	OrdType	N	Order type applicable to the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>
59	TimeInForce	N	Time qualifier of the order: <ul style="list-style-type: none"> <li>▪ 0 = Day (Default)</li> </ul> Absence of this field is interpreted as 0 = Day.
54	Side	Y	Side of the order <ul style="list-style-type: none"> <li>▪ 1 = Buy</li> <li>▪ 2 = Sell</li> <li>▪ 5 = Sell Short</li> </ul>
38	OrderQty	N	Total order quantity of the order

44	Price	N	Price of order. Required if: OrdType (40) = 2 (Limit).
60	TransactTime	Y	The time at which the particular message was generated.
528	OrderCapacity	N	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ A = Agency</li> <li>▪ P = Principal</li> </ul>
1090	MaxPriceLevels	N	Maximum number of price levels to trade through. Applicable if: Order Type (40) = 2 (Limit). If present, this should be set as 1.
77	PositionEffect	N	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>▪ C = Close</li> </ul> Applicable only if: Side (54) = 1 (Buy) to indicate covering a short sell.
39	OrdStatus	Y	Order Status after applying the transaction that is being communicated: <ul style="list-style-type: none"> <li>▪ 0 = New</li> </ul>
150	ExecType	Y	Execution Type that indicates the reason for the generation of the Execution Report: <ul style="list-style-type: none"> <li>▪ 0 = New</li> </ul>
14	CumQty	Y	Cumulative execution size
151	LeavesQty	Y	Open order quantity
58	Text	N	Free Text
1093	LotType	N	Lot Type of the order: <ul style="list-style-type: none"> <li>▪ 2 = Round Lot</li> </ul> Absence of this field indicates a Round (i.e., Board) Lot order.
797	CopyMsgIndicator	Y	Indicates whether or not this message is a drop copy of another message. Always set to Y.

[Message Trailer](#)

### 7.7.2.1 Order Rejected

The CCCG will send this execution report once the new order, which has already been registered (refer to [Section 7.7.1](#)) is rejected by the home market.

FIX Tag	Field Name	Required?	Description
<a href="#">Message</a> Header			
35	MsgType	Y	8 = Execution Report

Message Body			
11	ClOrdID	Y	Client specified identifier of the order.
37	OrderID	Y	Order ID as assigned by CSC.
17	ExecID	Y	Unique Execution ID assigned for each Execution Report generated.
Component Block <Parties>			
453	NoPartyIDs	Y	Number of party identifiers. The value in this field can be 1 or 2 or 3.
→	448	PartyID	ID of the Broker client submitting the order: Broker ID/SPSA Investor ID/BS User ID.
→	447	PartyIDSource	Defines the naming convention used in specifying the PartyID (448): <ul style="list-style-type: none"> <li>▪ D = Proprietary/Custom Code</li> </ul>
→	452	PartyRole	Role of the specified PartyID (448): <ul style="list-style-type: none"> <li>▪ 1 = Executing Firm (Broker ID that submitted the order)</li> <li>▪ 5 = Investor ID (SPSA Investor ID)</li> <li>▪ 75 = Location ID (BS User ID)</li> </ul>
End Component Block			
Component Block <Instrument>			
48	SecurityID	Y	Instrument identifier
22	SecurityIDSource	Y	Identifies the source of the SecurityID (48). Required if: SecurityID (48) is specified
207	SecurityExchange	N	The market which is used to identify the security. Required if: SecurityIDSource (22) = 8 (Exchange Symbol).
End Component Block			
40	OrdType	N	Order type applicable to the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>
59	TimeInForce	N	Time qualifier of the order: <ul style="list-style-type: none"> <li>▪ 0 = Day (Default)</li> </ul> Absence of this field is interpreted as 0 = Day.
54	Side	Y	Side of the order <ul style="list-style-type: none"> <li>▪ 1 = Buy</li> <li>▪ 2 = Sell</li> <li>▪ 5 = Sell Short</li> </ul>
38	OrderQty	N	Total order quantity of the order
44	Price	N	Price of order. Required if: OrdType (40) = 2 = Limit.
60	TransactTime	Y	The time at which the particular message was generated.

528	OrderCapacity	N	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ A = Agency</li> <li>▪ P = Principal</li> </ul>
1090	MaxPriceLevels	N	Maximum number of price levels to trade through. Applicable if: <p style="text-align: center;">OrdType (40) = 2 (Limit).</p> If present, this should be set as 1.
77	PositionEffect	N	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>▪ C = Close</li> </ul> Applicable only if: <p style="text-align: center;">Side (54) = 1 (Buy) to indicate covering a short sell.</p>
39	OrdStatus	Y	Order Status after applying the transaction that is being communicated: <ul style="list-style-type: none"> <li>▪ 8 = Rejected</li> </ul>
150	ExecType	Y	Execution Type that indicates the reason for the generation of the Execution Report: <ul style="list-style-type: none"> <li>▪ 8 = Reject</li> </ul>
14	CumQty	Y	Cumulative execution size
151	LeavesQty	Y	Open order quantity
103	OrdRejReason	N	Reject code indicating the reason for the order reject: <ul style="list-style-type: none"> <li>▪ 3 = Order exceeds Limit</li> <li>▪ 6 = Duplicate order (e.g., dupe ClOrdID)</li> <li>▪ 13 = Incorrect quantity</li> <li>▪ 22 = Short sell not permitted</li> <li>▪ 99 = Other</li> <li>▪ 101 = Price exceeds current price band (override not allowed)</li> </ul>
1328	RejectText	N	Textual description of the rejection that is being communicated through this execution report
58	Text	N	Free Text
797	CopyMsgIndicator	Y	Indicates whether or not this message is a drop copy of another message. Always set to Y.
<a href="#">Message Trailer</a>			

### 7.7.3 Order Cancelled

The CCCG sends this execution report once the request to cancel an order is accepted by the home market.

FIX Tag	Field Name	Required?	Description
<a href="#">Message</a> Header			
35	MsgType	Y	8 = Execution Report

Message Body			
11	ClOrdID	N	Client specified identifier of the order cancel request.
41	OrigClOrdID	N	Original client specified identifier for the incoming cancel request
37	OrderID	Y	Order ID as assigned by the exchange.
17	ExecID	Y	Unique Execution ID assigned for each Execution Report generated.
Component Block <Parties>			
453	NoPartyIDs	Y	Number of party identifiers. The value in this field can be 1 or 2.
→	448	PartyID	ID of the Broker client submitting the order: Broker ID/BS User ID.
→	447	PartyIDSource	Defines the naming convention used in specifying the PartyID (448): <ul style="list-style-type: none"> <li>▪ D = Proprietary/Custom Code</li> </ul>
→	452	PartyRole	Role of the specified PartyID (448): <ul style="list-style-type: none"> <li>▪ 1 = Executing Firm (Broker ID that submitted the order)</li> <li>▪ 75 = Location ID (BS User ID)</li> </ul>
End Component Block			
Component Block <Instrument>			
48	SecurityID	Y	Instrument identifier
22	SecurityIDSource	Y	Identifies the source of the SecurityID (48): <ul style="list-style-type: none"> <li>▪ 8 = Exchange Symbol</li> </ul> Required if: SecurityID (48) is specified
207	SecurityExchange	N	The market which is used to identify the security: Required if: SecurityIDSource (22) = 8 (Exchange Symbol).
End Component Block			
40	OrdType	N	Order type applicable to the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>
59	TimeInForce	N	Time qualifier of the order: <ul style="list-style-type: none"> <li>▪ 0 = Day (Default)</li> </ul> Absence of this field is interpreted as 0 = Day.
54	Side	Y	Side of the order <ul style="list-style-type: none"> <li>▪ 1 = Buy</li> <li>▪ 2 = Sell</li> <li>▪ 5 = Sell Short</li> </ul>
38	OrderQty	N	Total order quantity of the order
44	Price	N	Price of order. Required if: OrdType (40) = 2 (Limit).

60	TransactTime	Y	The time at which the particular message was generated.
528	OrderCapacity	N	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ A = Agency</li> <li>▪ P = Principal</li> </ul>
1090	MaxPriceLevels	N	Maximum number of price levels to trade through. Applicable if: OrdType (40) = 2 (Limit). If present, this should be set as 1.
77	PositionEffect	N	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>▪ C = Close</li> </ul> Applicable only if: Side (54) = 1 (Buy) to indicate covering a short sell.
39	OrdStatus	Y	Order Status after applying the transaction that is being communicated: <ul style="list-style-type: none"> <li>▪ 4 = Cancelled</li> </ul>
150	ExecType	Y	Execution Type that indicates the reason for the generation of the Execution Report: <ul style="list-style-type: none"> <li>▪ 4 = Cancelled</li> </ul>
14	CumQty	Y	Cumulative execution size
151	LeavesQty	Y	Open order quantity
58	Text	N	Free Text
1093	LotType	N	Lot Type of the order: <ul style="list-style-type: none"> <li>▪ 2 = Round Lot</li> </ul> Absence of this field indicates a Round (i.e., Board) Lot order.
797	CopyMsgIndicator	Y	Indicates whether or not this message is a drop copy of another message. Always set to Y.

#### [Message Trailer](#)

### 7.7.4 Order Cancelled – Unsolicited

The CCG will send this execution report for an unsolicited cancellation of an order.

FIX Tag	Field Name	Required?	Description
<b>Message Header</b>			
35	MsgType	Y	8 = Execution Report
<b>Message Body</b>			
11	ClOrdID	N	Client specified identifier of the order.
37	OrderID	Y	Order ID as assigned by the exchange.



17	ExecID	Y	Unique Execution ID assigned for each Execution Report generated.
Component Block <Parties>			
453	NoPartyIDs	Y	Number of party identifiers. The value in this field can be 1 or 2.
→	448	PartyID	ID of the Broker client submitting the order: Broker ID/BS User ID.
→	447	PartyIDSource	Defines the naming convention used in specifying the PartyID (448): <ul style="list-style-type: none"> <li>▪ D = Proprietary/Custom Code</li> </ul>
→	452	PartyRole	Role of the specified PartyID (448): <ul style="list-style-type: none"> <li>▪ 1 = Executing Firm (Broker ID that submitted the order)</li> <li>▪ 75 = Location ID (BS User ID)</li> </ul>
End Component Block			
Component Block <Instrument>			
48	SecurityID	Y	Instrument identifier
22	SecurityIDSource	Y	Identifies the source of the SecurityID (48): <ul style="list-style-type: none"> <li>▪ 8 = Exchange Symbol</li> </ul> Required if: SecurityID (48) is specified
207	SecurityExchange	N	The market which is used to identify the security: Required if: SecurityIDSource (22) = 8 (Exchange Symbol).
End Component Block			
40	OrdType	N	Order type applicable to the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>
59	TimeInForce	N	Time qualifier of the order: <ul style="list-style-type: none"> <li>▪ 0 = Day (Default)</li> </ul> Absence of this field is interpreted as 0 = Day.
54	Side	Y	Side of the order <ul style="list-style-type: none"> <li>▪ 1 = Buy</li> <li>▪ 2 = Sell</li> <li>▪ 5 = Sell Short</li> </ul>
38	OrderQty	N	Total order quantity of the order
44	Price	N	Price of order. Required if: OrdType (40) = 2 (Limit).
60	TransactTime	Y	The time at which the particular message was generated.
528	OrderCapacity	N	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ A = Agency</li> <li>▪ P = Principal</li> </ul>

1090	MaxPriceLevels	N	Maximum number of price levels to trade through. Applicable if: OrdType (40) = 2 (Limit). If present, this should be set as 1.
77	PositionEffect	N	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>C = Close</li> </ul> Applicable only if: Side (54) = 1 (Buy) to indicate covering a short sell.
39	OrdStatus	Y	Order Status after applying the transaction that is being communicated: <ul style="list-style-type: none"> <li>4 = Cancelled</li> </ul>
150	ExecType	Y	Execution Type that indicates the reason for the generation of the Execution Report: <ul style="list-style-type: none"> <li>4 = Cancelled</li> </ul>
14	CumQty	Y	Cumulative execution size
151	LeavesQty	Y	Open order quantity
1328	RejectText	N	Textual description of the reason for unsolicited cancel
58	Text	N	Free Text
1093	LotType	N	Lot Type of the order: <ul style="list-style-type: none"> <li>2 = Round Lot</li> </ul> Absence of this field indicates a Round (i.e., Board) Lot order.
378	ExecRestatementReason	N	Code to identify the reason for an Execution Report message with Exec Type= 4 (Cancel): <ul style="list-style-type: none"> <li>8 = Market Operation</li> <li>99 = Other</li> <li>105 = Cancel due to Broker suspension</li> <li>106 = Cancel due to Exchange Participant suspension</li> <li>107 = System Cancel</li> <li>108 = Cancelled by the home market</li> </ul>
797	CopyMsgIndicator	Y	Indicates whether or not this message is a drop copy of another message. Always set to Y.

[Message Trailer](#)

### 7.7.5 Trade (Order Executed)

The CCCG sends this execution report for a trade concluded at the home market.

FIX Tag	Field Name	Required?	Description
<a href="#">Message Header</a>			
35	MsgType	Y	8 = Execution Report

Message Body			
11	ClOrdID	N	Client specified identifier of the order.
37	OrderID	Y	Order ID as assigned by the exchange.
17	ExecID	Y	Unique Execution ID assigned for each Execution Report generated.
880	TrdMatchID	N	Identifier assigned to a trade by the matching system
Component Block <Parties>			
453	NoPartyIDs	Y	Number of party identifiers. The value in this field can be 1 or 2 or 3.
→	448	PartyID	ID of the Broker client submitting the order: Broker ID/ Counterparty Broker ID/BS User ID.
→	447	PartyIDSource	Defines the naming convention used in specifying the PartyID (448): <ul style="list-style-type: none"> <li>▪ D = Proprietary/Custom Code</li> </ul>
→	452	PartyRole	Role of the specified PartyID (448): <ul style="list-style-type: none"> <li>▪ 1 = Executing Firm (Broker ID that submitted the order)</li> <li>▪ 17 = Contra Firm (Counterparty Broker ID for this trade).</li> <li>▪ 75 = Location ID (BS User ID)</li> </ul>
End Component Block			
Component Block <Instrument>			
48	SecurityID	Y	Instrument identifier
22	SecurityIDSource	Y	Identifies the source of the SecurityID (48): <ul style="list-style-type: none"> <li>▪ 8 = Exchange Symbol</li> </ul> Required if: SecurityID (48) is specified
207	SecurityExchange	N	The market which is used to identify the security: Required if: SecurityIDSource (22) = 8 (Exchange Symbol).
End Component Block			
40	OrdType	N	Order type applicable to the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>
59	TimeInForce	N	Time qualifier of the order: <ul style="list-style-type: none"> <li>▪ 0 = Day (Default)</li> </ul> Absence of this field is interpreted as 0 = Day.
54	Side	Y	Side of the order <ul style="list-style-type: none"> <li>▪ 1 = Buy</li> <li>▪ 2 = Sell</li> <li>▪ 5 = Sell Short</li> </ul>
38	OrderQty	N	Total order quantity of the order
44	Price	N	Price of order. Required if: OrdType (40) = 2 (Limit).

60	TransactTime	Y	The time at which the particular message was generated.
528	OrderCapacity	N	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ A = Agency</li> <li>▪ P = Principal</li> </ul>
1090	MaxPriceLevels	N	Maximum number of price levels to trade through. Applicable if: OrdType (40) = 2 (Limit). If present, this should be set as 1.
77	PositionEffect	N	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>▪ C = Close</li> </ul> Applicable only if: Side (54) = 1 (Buy) to indicate covering a short sell.
39	OrdStatus	Y	Order Status after applying the transaction that is being communicated: <ul style="list-style-type: none"> <li>▪ 1 = Partially Filled</li> <li>▪ 2 = Filled</li> </ul>
150	ExecType	Y	Execution Type that indicates the reason for the generation of the Execution Report: <ul style="list-style-type: none"> <li>▪ F = Trade</li> </ul>
31	LastPx	Y	Execution Price
32	LastQty	Y	Execution Size
14	CumQty	Y	Cumulative execution size
151	LeavesQty	Y	Open order quantity
574	MatchType	N	The point in the matching process at which the trade was matched: <ul style="list-style-type: none"> <li>▪ 4 = Auto match</li> </ul>
58	Text	N	Free Text
1093	LotType	N	Lot Type of the order: <ul style="list-style-type: none"> <li>▪ 2 = Round Lot</li> </ul> Absence of this field indicates a Round (i.e., Board) Lot order
797	CopyMsgIndicator	Y	Indicates whether or not this message is a drop copy of another message. Always set to Y.
75	TradeDate	N	Indicates date of trading day. Absence of this field indicates current day.

#### [Message Trailer](#)

#### Notes:

1. Counterparty Broker ID (PartyID with PartyRole=17) may or may not always be present in the Parties block.
2. Trade Date, if present, may carry a date different from current date, in "Test Mode".

## 7.8 Infrastructure

### 7.8.1 Business Message Reject (j)

This message can be initiated by the CCG.

FIX Tag	Field Name	Required?	Description
<b>Message Header</b>			
35	MsgType	Y	j= Business Message Reject
<b>Message Body</b>			
45	RefSeqNum	N	Sequence number of the message which caused the rejection
372	RefMsgType	Y	Message type of the rejected message.
379	BusinessRejectRefID	N	The value of the business-level "ID" field on the message being referenced.
380	BusinessRejectReason	Y	Code specifying the reason for the rejection of the message: <ul style="list-style-type: none"> <li>▪ 0 = Other</li> <li>▪ 3 = Unspecified Message Type</li> <li>▪ 4 = Application not available</li> </ul>
58	Text	N	Where possible, message to explain reason for rejection
<b>Message Trailer</b>			

## 8. Data Dictionary

### 8.1 Session Level

Tag	Field Name	Data Type	Description	Comments
7	BeginSeqNo	SeqNum	Sequence number of first message in range.	
8	BeginString	String	Identifies beginning of new message and protocol version Valid values: FIXT.1.1	
9	BodyLength	Length	Number of characters after this field up to and including the delimiter immediately preceding the CheckSum.	
10	Checksum	String	Three byte, simple checksum. Always the last field in message	
16	EndSeqNo	SeqNum	Sequence number of last message in range	
34	MsgSeqNum	SeqNum	Sequence number of the message.	
35	MsgType	String	<ul style="list-style-type: none"> <li>▪ 0 = Heartbeat</li> <li>▪ 1 = Test Request</li> <li>▪ 2 = Resend Request</li> <li>▪ 3 = Reject</li> <li>▪ 4 = Sequence Reset</li> <li>▪ 5 = Logout</li> <li>▪ 8 = Execution Report</li> <li>▪ j = Business Message Reject</li> <li>▪ A = Logon</li> </ul>	
36	NewSeqNo	SeqNum	Sequence number of the next message to be transmitted.	
43	PossDupFlag	Boolean	Whether the message was previously transmitted under the same MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N).	
45	RefSeqNum	SeqNum	Sequence number of the rejected message.	

49	SenderCompID	String	Comp ID of the party sending the message.	Limited to maximum 11 characters
52	SendingTime	UTCTimestamp	Time the message was transmitted. Format: YYYYMMDD-HH:MM:SS.sss	
56	TargetCompID	String	Comp ID of the party the message is sent to	Limited to maximum 11 characters
97	PossResend	Boolean	Whether the message was previously transmitted under a different MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N).	
98	EncryptMethod	Int	Method of encryption	
108	HeartBtInt	Int	Indicates the heartbeat interval in seconds.	
112	TestReqID	String	Required if the heartbeat is a response to a Test Request. The value in this field should echo the TestReqID (112) received in the Test Request. Identifier for the request	Limited to maximum 20 characters
122	OrigSendingTime	UTCTimestamp	Time the message was originally transmitted. If the original time is not available, this should be the same value as SendingTime (52). Required if PossDupFlag (43) is Possible Duplicate (Y). Format: YYYYMMDD-HH:MM:SS.sss	
123	GapFillFlag	Boolean	Mode in which the message is being used. Absence of this field is interpreted as Sequence Reset (N).	
371	RefTagID	Int	If a message is rejected due to an issue with a particular field its tag number will be indicated.	
372	RefMsgType	String	Message type of the rejected message.	Limited to maximum 3 characters

373	SessionRejectReason	Int	Code specifying the reason for the reject: <ul style="list-style-type: none"> <li>▪ 0 = Invalid Tag Number</li> <li>▪ 1 = Required Tag Missing</li> <li>▪ 2 = Tag not defined for this message</li> <li>▪ 3 = Undefined tag</li> <li>▪ 4 = Tag specified without a value</li> <li>▪ 5 = Value is incorrect (out of range) for this tag</li> <li>▪ 6 = Incorrect data format for value</li> <li>▪ 9 = CompID problem</li> <li>▪ 10 = Sending Time Accuracy problem</li> <li>▪ 11 = Invalid Msg Type</li> <li>▪ 13 = Tag appears more than once</li> <li>▪ 15 = Repeating group fields out of order</li> <li>▪ 16 = Incorrect NumInGroup count for repeating group</li> <li>▪ 18 = Invalid/Unsupported Application Version</li> <li>▪ 99 = Other</li> </ul>	
379	BusinessRejectRefID	String	The value of the business-level "ID" field on the message being referenced.	Limited to maximum 20 characters
380	BusinessRejectReason	Int	Code specifying the reason for the rejection of the business message <ul style="list-style-type: none"> <li>▪ 0 = Other</li> <li>▪ 3= Unspecified Message Type</li> <li>▪ 4= Application not available</li> </ul>	
464	TestMessageIndicator	Boolean	Used to specify that this FIX session will be sending and receiving "test" vs. "production" messages Only applicable for Logon (35=A) message from the CCCG to the client	
554	Password	String	Password assigned to the Comp ID. Required if the message is generated by the client.	Limited to maximum 24 characters
789	NextExpectedMsgSeqNum	SeqNum	Next expected MsgSeqNum value to be received	



925	NewPassword	String	New password for the Comp ID.	Limited to maximum 24 characters
1128	AppVerID	String	Version of FIX used in the message. Required if the message is generated by the CCCG.	Limited to maximum 4 characters
1137	DefaultAppVerID	String	Default version of FIX messages used in this session.	Limited to maximum 4 characters
1400	EncryptedPasswordMethod	Int	Enumeration defining the encryption method used to encrypt password fields: <ul style="list-style-type: none"> <li>▪ 101 = RSA</li> </ul>	
1402	EncryptedPassword	Data	Encrypted password – encrypted via the method specified in EncryptedPasswordMethod (1400)	Limited to maximum 449 characters
1404	EncryptedNewPassword	Data	Encrypted new password – encrypted via the method specified in EncryptedPasswordMethod (1400)	Limited to maximum 449 characters
1409	SessionStatus	Int	Status of the FIX session. Required if the message is generated by the server. <ul style="list-style-type: none"> <li>▪ 0 = Session active</li> <li>▪ 1 = Session password changed</li> <li>▪ 2 = Session password due to expire</li> <li>▪ 3 = New session password does not comply with policy</li> <li>▪ 4 = Session logout complete</li> <li>▪ 5 = Invalid username or password</li> <li>▪ 6 = Account locked</li> <li>▪ 7 = Logons are not allowed at this time</li> <li>▪ 8 = Password expired</li> <li>▪ 100 = Password change is required</li> <li>▪ 101 = Other</li> </ul>	

## 8.2 Business Level

Tag	Field Name	Data Type	Description	Comments
11	ClOrdID	String	Unique identifier of the order as assigned by the client	Limited to maximum 20 characters
14	CumQty	Qty	Total quantity filled	
17	ExecID	String	Unique Execution ID assigned by the system for each Execution Report generated	Limited to maximum 20 characters
22	SecurityIDSource	String	Identifier of the source of the SecurityID (48) value.	
31	LastPx	Price	Price of this fill. Execution price	
32	LastQty	Qty	Quantity executed in this fill. Execution size	
37	OrderID	String	Unique identifier for Order as assigned by the server	Limited to maximum 20 characters
38	OrderQty	Qty	Total order quantity.	
39	OrdStatus	Char	Current status of the order: <ul style="list-style-type: none"> <li>▪ 0 = New</li> <li>▪ 1 = Partially filled</li> <li>▪ 2 = Filled</li> <li>▪ 4 = Cancelled</li> <li>▪ 8 = Rejected</li> <li>▪ A = Pending New</li> </ul>	
40	OrdType	Char	Type of the order: <ul style="list-style-type: none"> <li>▪ 2 = Limit</li> </ul>	
41	OrigClOrdID	String	ClOrdID (11) of the previous order (NOT the initial order of the day) as assigned by the client, used to identify the previous order in cancel and cancel/replace requests	Limited to maximum 20 characters
44	Price	Price	Limit price of the order	

48	SecurityID	String	Identifier of the instrument.	Limited to maximum 20 characters
54	Side	Char	Side of the order: <ul style="list-style-type: none"> <li>1 = Buy</li> <li>2 = Sell</li> <li>5 = Sell short</li> </ul>	
58	Text	String	Free format text string	Limited to maximum 90 characters
59	TimeInForce	Char	Specifies how long the order remains in effect: <ul style="list-style-type: none"> <li>0 = Day</li> </ul> Absent = Day	
60	TransactTime	UTCTimestamp	Time qualifier of the order. Format: YYYYMMDD-HH:MM:SS.sss	
75	TradeDate	LocalMktDate	Indicates date of trading day. Absence of this field indicates current day. If present, may carry a date different from current date, in <b>"Test Message Indicator in Logon message = Test Mode"</b>	
77	PositionEffect	Char	Indicates whether the resulting position after a trade should be an opening position or closing position: <ul style="list-style-type: none"> <li>C = Close</li> </ul>	
103	OrdRejReason	Int	Code to identify reason for order rejection <ul style="list-style-type: none"> <li>3 = Order exceed limit</li> <li>6 = Duplicate order</li> <li>13 = Incorrect quantity</li> <li>22 = Short sell not permitted</li> <li>99 = Other</li> <li>101 = Price exceeds current price band (override not allowed)</li> </ul>	

150	ExecType	Char	Describes the specific Execution report <ul style="list-style-type: none"> <li>▪ 0 = New</li> <li>▪ 4 = Cancelled</li> <li>▪ 8 = Rejected</li> <li>▪ A = Pending New</li> <li>▪ F = Trade</li> </ul>	
151	LeavesQty	Qty	Quantity open for further execution	
207	SecurityExchange	Exchange	Market used to help identify a security. Valid values: <ul style="list-style-type: none"> <li>▪ XSSC = Stocks listed in Shanghai Stock Exchange under Shanghai – Hong Kong Stock Connect</li> <li>▪ XSEC = Stocks listed in Shenzhen Stock Exchange under Shenzhen – Hong Kong Stock Connect</li> </ul> Required if: SecurityIDSource (22) = 8 (Exchange Symbol).	
378	ExecRestatementReason	Int	Code to identify reason for an Execution Report message sent when communicating an unsolicited cancel: <ul style="list-style-type: none"> <li>▪ 8 = Market (Exchange) Option</li> <li>▪ 99 = Other</li> <li>▪ 105 = Cancel due to Broker suspension</li> <li>▪ 106 = Cancel due to Exchange Participant suspension</li> <li>▪ 107 = System Cancel</li> <li>▪ 108 = Cancelled by the home market</li> </ul>	
447	PartyIDSource	Char	Identifies class or source of the PartyID (448) value.	
448	PartyID	String	Party identifier/code	Limited to maximum 11 characters
452	Party Role	Int	Identifies the type or role of the PartyID (448) specified	
453	NoPartyIDs	NumInGroup	Number of PartyID	

528	OrderCapacity	Char	Designates the capacity of the firm placing the order: <ul style="list-style-type: none"> <li>▪ 1 = Agency</li> <li>▪ 2 = Principal</li> </ul>	
574	MatchType	String	The point in the matching process at which this trade was matched: <ul style="list-style-type: none"> <li>▪ 4 = Auto Match</li> </ul>	Limited to maximum 5 characters
797	CopyMsgIndicator	Boolean	Indicates whether or not the message is a drop copy of another message.	
880	TrdMatchID	String	Identifier assigned to a trade by the matching system	Limited to maximum 24 characters
1090	MaxPriceLevels	Int	Allows an order to specify a maximum number of price levels to trade through	
1093	LotType	Char	Lot Type of the order: <ul style="list-style-type: none"> <li>▪ 2 = Round Lot</li> </ul> Absence of this field indicates a Round (i.e., Board) Lot order.	
1328	RejectText	String	Rejection text	Limited to maximum 99 characters

---

# Appendices

---

## A. Password Policy

- Length is 8 characters.
- Must comprise of a mix of alphabets (A-Z and a-z) and digits (0-9)
- Must be changed on first-time logon or first logon after reset from HKEX market operations.
- New password can't be one of the previous 5 passwords.
- Can't be changed more than once per day.
- Session will be locked after 3 consecutive invalid passwords
- Expires every 90 days.