MDP 3.0 - Conflated TCP Market Data Group Processing

This page outlines conflated TCP market data group processing.

> The Conflated TCP Market Data Group schemas utilize appended template extension. With template extension, a template is extended as an appended extension, client systems can choose to continue process the prior template version or process the new data with the new schema version.

Session Attributes

The conflated TCP Market Data Group uses the FIX and SBE protocol to establish and manage bi-directional sessions. A MDP TCP session is defined as a bi-directional message stream between two parties. A session will be maintained by default on a weekly basis and represented with a Universally Unique ID (UUID), which is recommended to the current timestamp. A client system can Negotiate and Terminate multiple times intra-day or at the end of each day, with the same UUID, or Negotiate the new session with a new UUID.

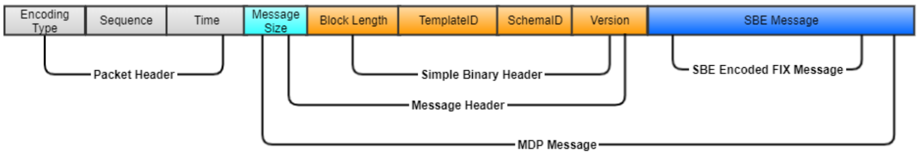
BrokerTec Session Access Model

The same MDP TCP Sessions can be used for U.S. markets, U.S. Treasury Actives, and U.S. Repo. The same MDP TCP Sessions can be used for EU markets, EU Repo and EGBs. The following table outlines the session access model.

Example Session ID	Session Entitlement	US Actives	US Repo	EU Repo	EGBs
AB	US	Yes	Yes	No	No
XY	EU	No	No	Yes	Yes

Packet Structure

The encoded FIX transmission for the CME MDP Conflated TCP market data group is sent in a packet structured. The following diagram shows the structure of an SBE-encoded message for the conflated TCP market data group:



The following table outlines the packet concepts in the diagram above:

Packet Header (14 bytes)	Message Header (10 bytes)	SBE Message (Variable Length)
Encoding Type: 2 bytes - CME SBE Version 1.0 Little-endian: 0xCAFE - Identifier of the encoding used in the message payload - Only available for the CME MDP Conflated TCP market data group.	**Message Size: 2 bytes** - Overall message size including headers to support framing	**Customer to CME Globex Messages** Examples: - Negotiation Message - Termination Message - Market Data Request Message

Sequence: 4 bytes - Packet sequence number. - A unique sequence number given to each packet sent. - Each channel will have its own separate set of sequence numbers that will increment sequentially with each packet and reset weekly.	**Block Length: 2 bytes** - The total space reserved for the root level of the message	**CME Globex to Customer Messages** Examples: - Negotiation Response - Negotiation Reject - Incremental Refresh Book Update (35=X) - Security Definition (35=d)
Sending Time: 8 bytes - UTC Time of message transmission by the Gateway. - UTC Timestamps are sent in number of nanoseconds since Unix epoch synced to a master clock to microsecond accuracy.	**TemplateID: 2 bytes** - Message SBE Template Identifier	
	SchemaID: 2 bytes - Identifier of the SBE Message Schema that contains the Template	
	Version: 2 bytes - Version of the SBE Message Schema in which the message is defined	

Conflated MDP TCP - Initialization and Unbinding

Conflated MDP TCP uses the FIX protocol via Simple Binary Encoding (SBE) to establish and manage bi-directional sessions. A session is defined as a bi-directional stream of ordered messages between two parties. Conflated MDP TCP accepts one connection per session ID.

> Conflated MDP TCP does not support session layer re-transmit request functionality.

TCP MDP Session Messages

TCP MDP session messages are summarized as follows.

Stage	Message Name	Template Name	From | To	Purpose
Initialization	Negotiate	Negotiate200	Client System to CME Globex	Initiates connection
	Negotiation Response	NegotiationResponse202	CME Globex to Client System	Accepts connection
	Negotiation Reject	NegotiationReject201	CME Globex to Client System	Rejects connection
Unbinding	Terminate	Terminate203	Client System to CME Globex	Disconnect connection
			CME Globex to Client System	

Universally Unique Identifier (UUID)

Each session established at the start of the week, intra-day, or at the beginning of each trading day is represented with a unique UUID value set by the customer as a 64-bit value. CME Group recommends using the system timestamp, which represents the number of microseconds since the epoch (Jan 1, 1970) as the timestamp.

Session Security

The customer will authenticate a TCP MDP session by digitally signing the Negotiate message using a hashed message authentication code (HMAC). In general, the customer interaction with CME will be as follows:

1. Customer logs into the CME Self Service Portal using 2-factor authentication over a secure channel (HTTPS).

2. CME generates a pair of cryptographically random keys—an Access Key and a Secret Key—for the customer. The keys are provided to the customer and also securely stored in a secure vault with CME.
3. Customer connects to MDP TCP and sends a digitally signed Negotiate message. The digital signature is generated using a hashed message authentication code (HMAC) based on the contents of the message and the Secret Key and is included in the Negotiate message sent to CME.
4. CME processes the Negotiate message and calculates the digital signature using the same HMAC-based process, and compares the calculated signature with the signature provided on the customer's Negotiate message. If the signatures match, the Negotiate message is authentic and validates that the sender has the Secret Key.

The details of calculating the digital signature are described below.

> CME MDP Conflated TCP uses the same HMAC methodology as iLink 3.

Steps to Sign a Login Request to MDP TCP

1. Create the canonical Negotiate message
2. Create Signature Using the Secret Key and Canonical FIX Message
3. Populate Signature in the Credentials Field

Step 1 - Create the Canonical FIX Message

Concatenate these fields into a single String delimited by the newline character (ie. '\n');

The required fields in Negotiate message are:

- RequestTimestamp
- UUID
- SessionID
- FirmID

Step 2 - Create Signature Using the Secret Key and Canonical FIX Message

The signature is a hash (digest) of the canonical message created in Step 1 using the Secret Key provided by CME.

The Secret Key downloaded from Request Center is **Base64 URL Encoded**. Customers must decode the secret key first.

Example of Creating Signature Using HMAC SHA256 in Java 8+

The HMAC signature is populated in a fixed length string field (i.e. String32Req) since SHA-256 signature is always 32 bytes.

> While real logic provides two options to encode this field—as a string and as a byte array—the byte array option should be used.

Encoding options for SHA-256 (256 bits long):

- Base64URL encoding FIX ASCII: 6 bits per char = CHAR(44)including padding character
- Hex: 4 bits per char = CHAR(64)
  - HEX example 617633EFEF2F98DCA32DD3BD8407122B9B375CF4BC076930F24C1F7A0F930094
- Binary: 8 bits per byte = BINARY(32)
  - Binary example byte array:[97, 118, 51, -17, -17, 47, -104, -36, -93, 45, -45, -67, -124, 7, 18, 43, -101, 55, 92, -12, -68, 7, 105, 48, -14, 76, 31, 122, 15, -109, 0, -108]

> The signature should be encoded as a byte array.

Signature calculation in Java:

```
public byte[] calculateHMAC(final String canonicalRequest, final String userKey) {
    byte[] hash = null;

    try
    {
      // Init HMAC instance
      Mac sha256HMAC;
      sha256HMAC = Mac.getInstance("HmacSHA256");

      // Initialize HMAC instance with the key
      // Decode the key first, since it is base64url encoded

      byte[] decodedUserKey = Base64.getUrlDecoder().decode(userKey);
      SecretKeySpec secretKey = new SecretKeySpec(decodedUserKey, "HmacSHA256");
      sha256HMAC.init(secretKey);


      // Calculate HMAC
      hash = sha256HMAC.doFinal(canonicalRequest.getBytes("UTF-8"));

    } catch (NoSuchAlgorithmException | InvalidKeyException | IllegalStateException |
UnsupportedEncodingException e)
    {
      e.printStackTrace();
    }

    return hash;
  }


}
```

Step 3 - Populate Signature in the Credentials Field

Populate the Signature from Step 2 into the HMACSignature field.

Session Negotiation

An MDP TCP session connection is initiated by the customer using a Negotiation message. The UUID must be sent in the Negotiation message and that ID is used for the lifetime of the session. Negotiation is provided as a mechanism used for the customer to declare what ID they will be using. There is no concept of resetting a session. Instead of starting over a session, a new session is negotiated with a new UUID.

The following required string fields cannot contain empty bytes:

- HMACSignature
- AccessKeyID
- Session
- Firm
- UUID

The following field will be checked for boundary conditions:

- RequestTimestamp

The customer should send a Negotiate message to the exchange and await acknowledgement.

- When Negotiate message from customer is accepted by CME, then CME will return a Negotiation Response.
- When Negotiate message from customer is rejected by CME, then CME will return a Negotiation Reject.
- If no response is forthcoming from the exchange after two 30 second heartbeat intervals have lapsed then Negotiation can be retried or out-of-band inquiry may be made to determine application readiness.
- If a client system sends a message before a connection has been authenticated and an acknowledgement has sent, the client system will receive a Terminate message from the gateway.

Heartbeat

The client system must send a heartbeat within the heartbeat interval of 30 seconds via template SubscriberHeartbeat210. The Conflated MDP TCP Gateway will send a heartbeat via template AdminHeartbeat12 if there is no market activity. Client systems not sending a heartbeat after two heartbeat intervals will be disconnected.

Unbinding

Termination

The Terminate message is used to signal that the sender is going to disconnect the TCP socket connection. The Terminate message initiates the termination of a FIX session. Once a FIX connection has been terminated, Subscription requests and book states are reset.

Conflated MDP TCP - Request Messages

The following section outlines request message market data functionality provided by CME Globex. Once the client system has established a FIX session, client systems may send request messages.   Subscription requests only apply to their respective channels.

TCP MDP Request Messages and Response Messages

Conflated MDP TCP payload messages are summarized as follows.

Message Name	FIX Tags	Template Name	From | To	Purpose
Security List Request Message	35-MsgType=x	SecurityListRequest208	Client System to CME Globex	Security List Request Message allows late joiners to recover all instruments or to recover any new / updates to instruments that may have been missed in the event of a lost connection.
Security Status Request Message	35-MsgType=g	SecurityStatusRequest209	Client System to CME Globex	Security Status Request Message allows joiners to recover market status messages or to recover any new / updates to market status that may have been missed in case of the lost connection.
Market Data Request Message	35-MsgType=V	MarketDataRequest205	Client System to CME Globex	Request to recover current state via the Market Data Snapshot Recovery Message (35=W) and receive all subsequent message type updates for the subscribed instruments or security groups.
Request Acknowledgment	35-MsgType=V	RequestAck206	CME Globex to Client System	Globex acknowledgment to denote if a client system request is fully or partially acknowledged.
Request Reject	tag 35-MsgType=Y	RequestReject207	CME Globex to Client System	Message sent to client systems as a reply to any type of request that is fully rejected.

Client System to CME Globex Request Messages

Request messages allow Conflated TCP client systems to recover and to subscribe to instruments or security groups. Specific request message ordering is not required. Client systems are required to be in Negotiated status before any requests are processed, otherwise the connection is Terminated. Subscription requests and book states are reset at session disconnection. Therefore, subscriptions and books must be rebuilt must be reestablished after TCP reconnection and successful Negotiation. Message responses to request messages maintain the original tag 60-TransactTime of when the message was initially sent.

Security List Request Message (tag 35-MsgType=x)

The Security List Request message (tag 35-MsgType=x) is a request message that allows client systems to recover all security definition (tag 35-MsgType=d) messages and to subscribe to any subsequent updates to security definition messages. All Security List Request messages must include a unique ID in tag 262-MDReqID. The MDReqID is used to identify the response message.   On Conflated TCP market data groups, expired instruments will be removed after the market close.

The Subscription Request Type (tag 263-SubscriptionReqType) indicates the type of response expected:

- **Snapshot (tag 263-SubscriptionReqType=0)** - Provides a current list of security definition (tag 35-MsgType=d) messages offered on the channel
- **Snapshot and updates (tag 263-SubscriptionReqType=1)** - Provides a current list of security definition (tag 35-MsgType=d) messages offered on the respective channel and a subscription to subsequent security definition updates
- **Unsubscribe (tag 263-SubscriptionReqType=2)** - Unsubscribes previous subscription.

Requests can be made at the security group (tag 1151-SecurityGroup) level or the instrument (tag 48-SecurityID) level. The security repeating group (tag 37022-NoSecurityGroups) should be equal to zero when a subscription is requested for all groups on the segment or individual Security IDs. The instrument repeating group (tag 146-NoRelatedSym) is the number of instruments requested. When NoSecurityGroups is greater than zero, the NoRelatedSym should be zero. If both NoSecurityGroup and NoRelatedSym are equal to zero then the request will be for all entitled groups.

After sending a Snapshot (tag 263-SubscriptionReqType=0) or Snapshot and updates (tag 263-SubscriptionReqType=1) message, the gateway will mark the End of Event (bit 7 equal to 1 via tag 5799-MatchEventIndicator) on the last message. After the End of Event is received, client systems may expect to receive heartbeats or further messaging related to the subscription. For Security List Requests, the original 60-LastUpdateTime denotes when the instrument added or an update was last sent.

Example Security List Request Message (tag 35-MsgType=x)

Tag	FIX Name	Tag Value
262	MDReqID	679
263	SubscriptionReqType	1 (Snapshot and updates)

Repeating Group 1 (NoSecurityGroups=0)		
Repeating Group 2 (NoRelatedSym=4)		
48	SecurityID	30027
48	SecurityID	141645
48	SecurityID	228219
48	SecurityID	467123

Security Status Request Message (tag 35-MsgType=g)

The Security Status Request message (tag 35-MsgType=g) allows client systems to recover all security status (tag 35-MsgType=f) messages and to subscribe to any subsequent updates to security status messages. All Security Status Request messages must include a unique ID in tag 262-MDReqID. The MDReqID is used to identify the response message.

The Subscription Request Type (tag 263-SubscriptionReqType) indicates the type of response expected:

- **Snapshot (tag 263-SubscriptionReqType=0)** - Provides a current list of security status messages (tag 35-MsgType=f) offered on the channel
- **Snapshot and updates (tag 263-SubscriptionReqType=1)** - Provides a current list of security status messages (tag 35-MsgType=f) offered on the channel and a subscription to subsequent security status updates
- **Unsubscribe (tag 263-SubscriptionReqType=2)** - Unsubscribes previous subscription.

Requests can be made at the security group (tag 1151-SecurityGroup) level or the instrument (tag 48-SecurityID) level. The security repeating group (tag 37022-NoSecurityGroups) should be equal to zero when a subscription is requested for all groups on the segment or individual Security IDs. The instrument repeating group (tag 146-NoRelatedSym) is the number of security status messages requested. When NoSecurityGroups is greater than zero, the NoRelatedSym should be zero. If both NoSecurityGroup and NoRelatedSym are equal to zero then the request will be for all entitled groups.

After sending a Snapshot (tag 263-SubscriptionReqType=0) or Snapshot and updates (tag 263-SubscriptionReqType=1) message, the gateway will mark the End of Event (bit 7 equal to 1 via tag 5799-MatchEventIndicator) on the last message. After the End of Event is received, client systems may expect to receive heartbeats or further messaging related to the subscription.

Example Security Status Request Message (tag 35-MsgType=g)

Tag	FIX Name	Tag Value
262	MDReqID	8
263	SubscriptionReqType	0 (Snapshot)
Repeating Group 1 (NoSecurityGroups=0)		
Repeating Group 2 (NoRelatedSym=3)		
48	SecurityID	9201
48	SecurityID	368213
48	SecurityID	577121

Market Data Request Message (tag 35-MsgType=V)

The Market Data Request message (tag 35-MsgType=V) is a request message that allows client systems to recover the current state of Market By Price (MBP) books in the Snapshot Full Refresh message (tag 35-MsgType=W) and subscribe to **all** subsequent messages applicable to the session. The Snapshot Refresh Message is sent via SBE template SnapshotRefreshTopOrders59.

All Market Data Request messages must include a unique ID in tag 262-MDReqID. The MDReqID is used to identify the response message.

The Subscription Request Type (tag 263-SubscriptionReqType) indicates the type of response expected:

- **Snapshot (tag 263-SubscriptionReqType=0)** - Provides a current list of SnapshotFullRefresh (tag 35-MsgType=W) messages for instruments offered on the respective channel.
- **Snapshot and updates (tag 263-SubscriptionReqType=1)** - Provides a current list of SnapshotFullRefresh (tag 35-MsgType=W) messages for instruments offered on the respective channel and a subscription to all subsequent messages applicable to the session.
- **Unsubscribe (tag 263-SubscriptionReqType=2)** - Unsubscribes previous subscription.

Requests can be made at the security group (tag 1151-SecurityGroup) level or the instrument (tag 48-SecurityID) level. The security repeating group (tag 37022-NoSecurityGroups) should be equal to zero when subscription is requested for all groups on the segment or individual Security IDs.
The instrument repeating group (tag 146-NoRelatedSym) is the number of security status messages requested. When NoSecurityGroups is greater than zero, the NoRelatedSym should be zero. If both NoSecurityGroup and NoRelatedSym are equal to zero then the request will be for all entitled groups.

Snapshot Full Refresh (35=W) messages will not be sent for the instruments that have only price bands or thresholds, but have had no book activity since the beginning of the week.

Example Market Data Request (tag 35-MsgType=V) Message

Tag	FIX Name	Tag Value
262	MDReqID	7
263	SubscriptionReqType	2 (Unsubscribe)
Repeating Group 1 (NoSecurityGroups=2)		
1151	SecurityGroup	ST25
1151	SecurityGroup	UGC
Repeating Group 2 (NoRelatedSym=0)		

CME Globex to Client System Response Messages

The CME Globex response messages outlined below are sent in response to Client System Request Messages.

Request Acknowledgment (tag 35-MsgType=V) Message

The Request Acknowledgment (tag 35-MsgType=V) message is sent from CME Group to the client system as a reply to any type of Request that is Fully or Partially Acknowledged. The status of the request is denoted on the Market Data Status ID (tag 37720-MDReqIDStatus):

- **FullAck (tag 37720-MDReqIDStatus=0)** - Requested subscription scope is fully acknowledged.
- **PartialAck (tag 37720-MDReqIDStatus=1)** - Requested subscription scope is partially acknowledged. Partial Acks are due to the client system session not being entitled to the requested instrument or group.

For Full Acks, the security group repeating group (tag 37022-NoSecurityGroups) and the instrument repeating group (tag 146-NoRelatedSym) will be zero. If there is a Partial Ack, the security group repeating group or the instrument repeating group will be populated with the subscribed instruments or groups.

Example Request Acknowledgment (tag 35-MsgType=V) Message

Tag	FIX Name	Tag Value
262	MDReqID	11
263	SubscriptionReqType	0 (Snapshot)
37720	MDReqIDStatus	1 (PartialAck)
Repeating Group 1 (NoSecurityGroups=0)		
Repeating Group 2 (NoRelatedSym=4)		
48	SecurityID	9297
48	SecurityID	24103
48	SecurityID	78157
48	SecurityID	100250

Request Reject (tag 35-MsgType=Y) Message

The Request Reject (tag 35-MsgType=Y) message is sent from CME Group to the client system as a reply to any type of request that is fully rejected. The field Market Data Reject Reason (tag 281-MDReqRejReason) provides the reject reason:

- **Unknown Security (tag 281-MDReqRejReason=0)** - All securities requested are unknown
- **Unknown Message (tag 281-MDReqRejReason=1)** - Unknown or invalid message sent
- **Unsupported Scope (tag 281-MDReqRejReason=2)** - Unsupported scope
- **Other (tag 281-MDReqRejReason=3)** - Other

The text field (tag 58-Text) provides the reason for the rejection.

Example Request Acknowledgment Message (tag 35-MsgType=V)

Tag	FIX Name	Tag Value

262	MDReqID	3
281	MDReqRejReason	0
58	Text	Entitlement not found for requested scope

TCP MDP Request Message Rules

This section summarizes request message rules for Security List Request, Security Status Request, and Market Data Request messages.

- Requests (security, status or data) can be made at the security group (tag 1151-SecurityGroup) or the instrument (tag 48-SecurityID) level.
    - A maximum of 254 instruments is allowed per request
- If a client is entitled to security group(s) and requests all groups (NoSecurtiyGroups=0, NoRelatedSym=0), RequestIDStatus=FullAck message will be sent.
- If a client system specifies both SecurityGroup and SecurityID lists in the same message, the MDP gateway will reply only with the SecurityGroup subscriptions and MDReqIDStatus= Partial.
- If a request is made and some (but not all) of the security group or instruments are not entitled, then the MDP gateway will reply with a partial ack (MDReqIDStatus= Partial).
- Tag 262-MDReqIDStatus should be unique for the week.
- If a client system with existing entitlements makes a request for only security groups or instruments to which the customer is not entitled, then a request rejection will be sent
- If a request is made on a session with NO entitlements, the request is rejected and the session is terminated
- If a client system is subscribed to a group and a new instrument is added to the same group mid-week, client systems will begin to receive real-time updates to that instrument.
- If entitlements for new security group/instruments are added mid-week, the following occurs:
    - Additional security group/instruments will be available, however the session will not send the newly entitled security group/instruments updates unless a request is sent.
- If entitlements for new security group/instruments are reduced mid-week, then the following occurs:
    - If all entitlements are removed, the gateway will terminate the connection.
    - If the entitlements are partially removed, the gateway will stop sending data for instruments/groups that have been removed.
- Session subscriptions will be removed either via an unsubscribe (tag 263-SubscriptionReqType=2) request message or disconnection.
- The conflated TCP gateway maintains a single subscription scope per connection and any new requested scope will be merged into already acknowledged subscription scope for the connection. A Security Group subscription will maintain priority over individual instrument subscription requests.

Conflated MDP TCP - Mid-Week Join and Recovery

For a client system mid-week join, or in the event of a CME Group Conflated MDP TCP component failure, the client system must reestablish a TCP connection, negotiate, and re-subscribe via request messages. Subscription requests and book states are reset at session disconnection and must be reestablished upon re-connection. To ensure there are no in-flight data issues during a mid-week join or component recovery, CME recommends client systems send request messages in the following order with tag 263-SubscriptionReqType=1 (Snapshot and Updates):

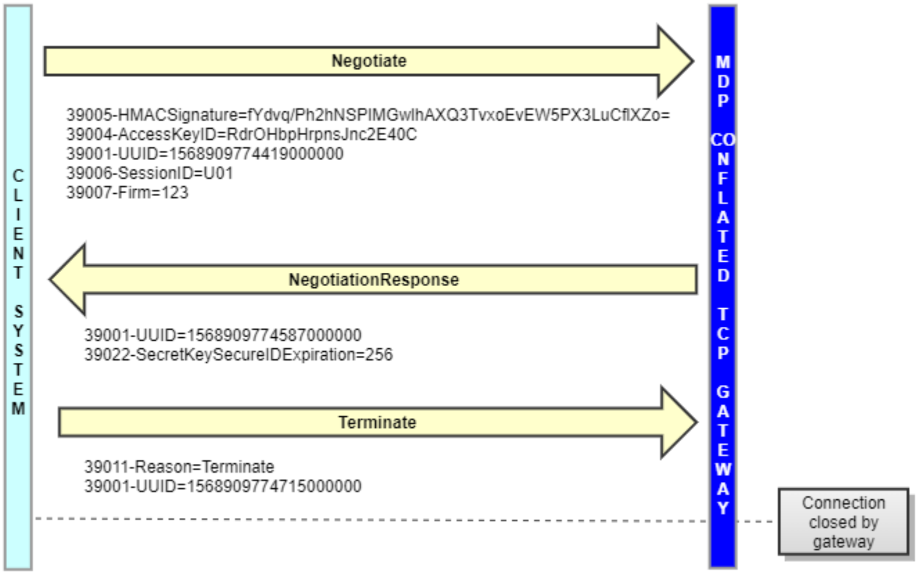
1. Security List Request Message (tag 35-MsgType=x)
2. Security Status Request Message (tag 35-MsgType=g)
3. Market Data Request Message (tag 35-MsgType=V)

Conflated TCP Messaging Examples

The following section provides Conflated MDP TCP messaging examples.

Example 1 - Successful Negotiation and Client System Initiated Termination

In this example a client system successfully logs on and logs off.



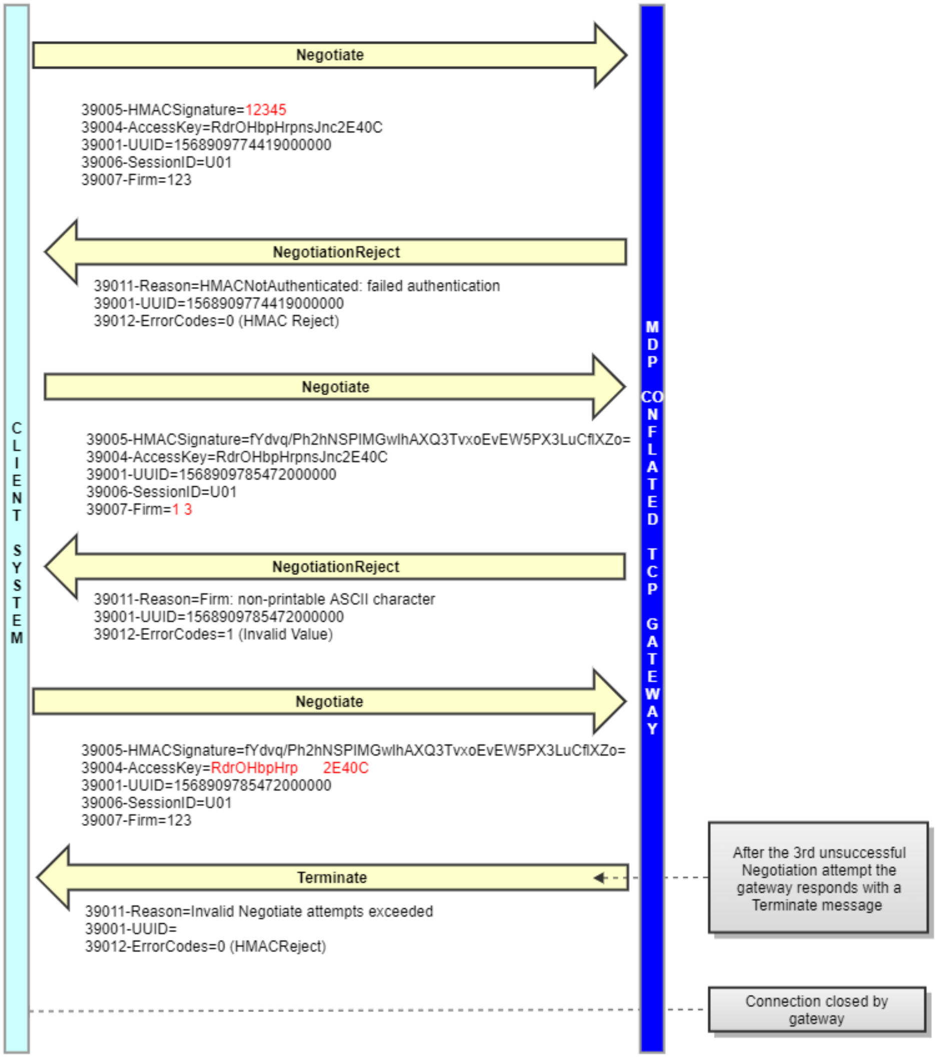
39005-HMACSignature=fYdvq/Ph2hNSPIMGwlhAXQ3TvxoEvEW5PX3LuCflXZo=
39004-AccessKeyID=RdrOHbpHrpnsJnc2E40C
39001-UUID=1568909774419000000
39006-SessionID=U01
39007-Firm=123

39001-UUID=1568909774587000000
39022-SecretKeySecureIDExpiration=256

39011-Reason=Terminate
39001-UUID=1568909774715000000

Example 2 - Negotiation Reject and CME Initiated Termination

In this is example a client system attempts to log on three times before a CME Group-initiated terminate is sent due to three unsuccessful negotiation attempts. After the third invalid invalid negotiate message, the MDP Conflated TCP Gateway will not send a only respond with a terminate message. In the example, the invalid values are denoted in red.



```
39005-HMACSignature=12345
39004-AccessKey=RdrOHbpHrpnsJnc2E40C
39001-UUID=1568909774419000000
39006-SessionID=U01
39007-Firm=123
```

```
39011-Reason=HMACNotAuthenticated: failed authentication
39001-UUID=1568909774419000000
39012-ErrorCodes=0 (HMAC Reject)
```

```
39005-HMACSignature=fYdvq/Ph2hNSPIMGwlhAXQ3TvxoEvEW5PX3LuCflXZo=
39004-AccessKey=RdrOHbpHrpnsJnc2E40C
39001-UUID=1568909785472000000
39006-SessionID=U01
39007-Firm=1 3
```

```
39011-Reason=Firm: non-printable ASCII character
39001-UUID=1568909785472000000
39012-ErrorCodes=1 (Invalid Value)
```

```
39005-HMACSignature=fYdvq/Ph2hNSPIMGwlhAXQ3TvxoEvEW5PX3LuCflXZo=
39004-AccessKey=RdrOHbpHrp     2E40C
39001-UUID=1568909785472000000
39006-SessionID=U01
39007-Firm=123
```

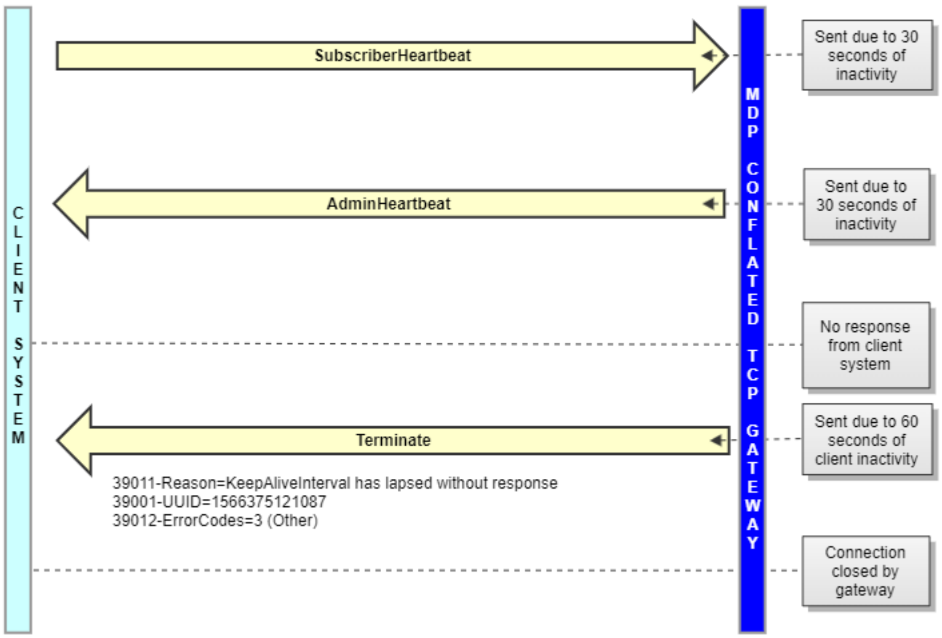
```
39011-Reason=Invalid Negotiate attempts exceeded
39001-UUID=
39012-ErrorCodes=0 (HMACReject)
```

After the 3rd unsuccessful Negotiation attempt the gateway responds with a Terminate message

Connection closed by gateway

Example 3 - Subscriber Heartbeat

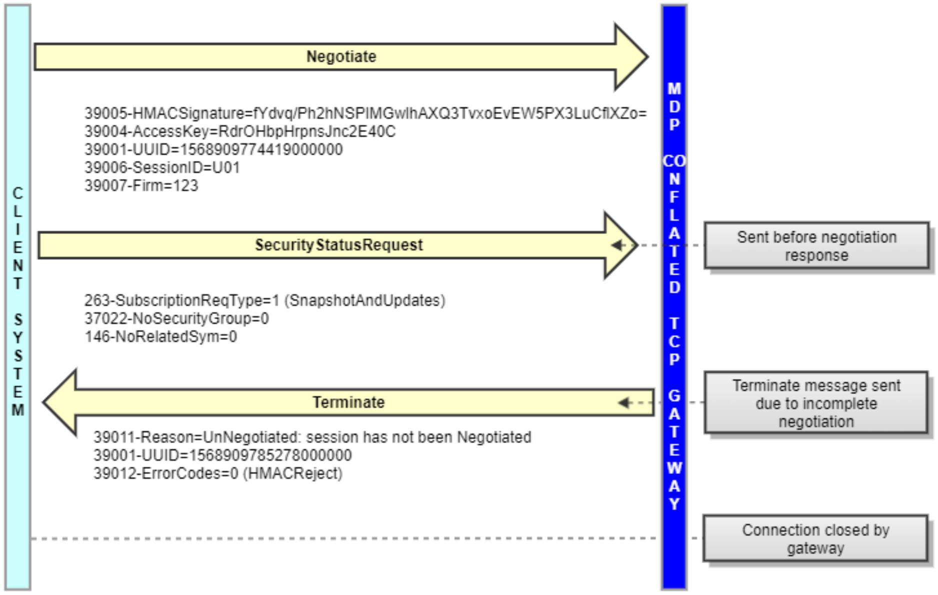
In this example, the client system sends a Subscriber Heartbeat (via template SubscriberHeartbeat210) due to 30 seconds of inactivity. Then, following 60 seconds of inactivity from the client system, the session is terminated.


CLIENT SYSTEM
MDP CONFLATED TCP GATEWAY
SubscriberHeartbeat
Sent due to 30 seconds of inactivity
AdminHeartbeat
Sent due to 30 seconds of inactivity
No response from client system
Terminate
Sent due to 60 seconds of client inactivity
Connection closed by gateway

39011-Reason=KeepAliveInterval has lapsed without response
39001-UUID=1566375121087
39012-ErrorCodes=3 (Other)

Example 4 - Incomplete Negotiation and Termination

In this example, after a client system Negotiate message is sent, the client system sends an additional message before the connection is authenticated and the gateway responds with a NegotiationResponse. Consequently, the gateway sends a Terminate message and closes the connection.


CLIENT SYSTEM
MDP CONFLATED TCP GATEWAY
Negotiate
SecurityStatusRequest
Sent before negotiation response
Terminate
Terminate message sent due to incomplete negotiation
Connection closed by gateway

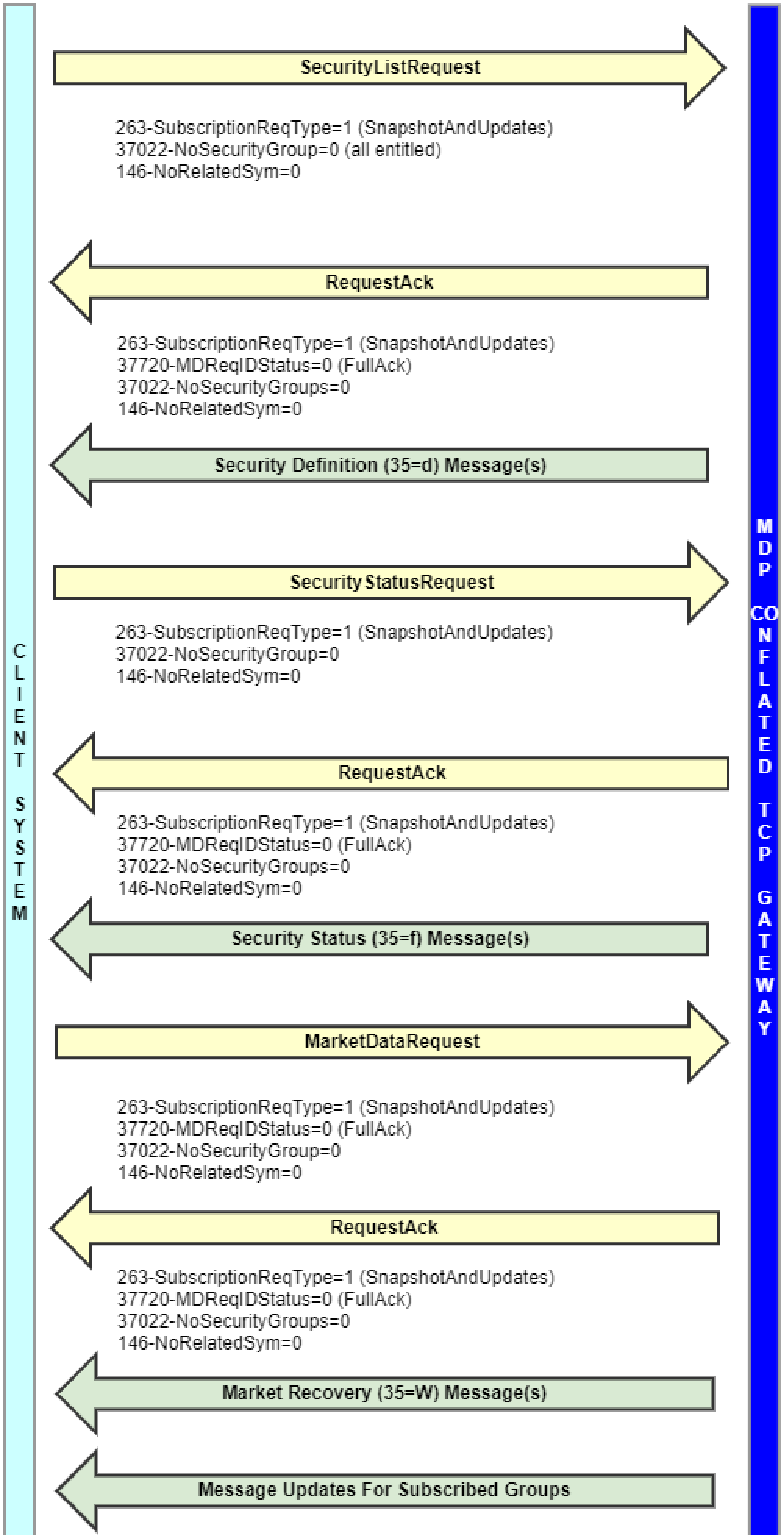
39005-HMACSignature=fYdvq/Ph2hNSPIMGwlhAXQ3TvxoEvEW5PX3LuCflXZo=
39004-AccessKey=RdrOHbpHrpnsJnc2E40C
39001-UUID=1568909774419000000
39006-SessionID=U01
39007-Firm=123

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0
146-NoRelatedSym=0

39011-Reason=UnNegotiated: session has not been Negotiated
39001-UUID=1568909785278000000
39012-ErrorCodes=0 (HMACReject)

Example 5 - Request Messages

The following example provides a message workflow for all 3 request types (Security List, Security Status and Market Data Request) during a TCP mid-week startup. All requests are for Snapshot & Updates (tag 263-SubscriptionReqType=1) and are fully acknowledged (tag 37720-MDReqIDStatus=0). Subscription requests only apply to their respective channels.



SecurityListRequest

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0 (all entitled)
146-NoRelatedSym=0

RequestAck

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroups=0
146-NoRelatedSym=0

Security Definition (35=d) Message(s)

SecurityStatusRequest

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0
146-NoRelatedSym=0

RequestAck

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroups=0
146-NoRelatedSym=0

Security Status (35=f) Message(s)

MarketDataRequest

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroup=0
146-NoRelatedSym=0

RequestAck

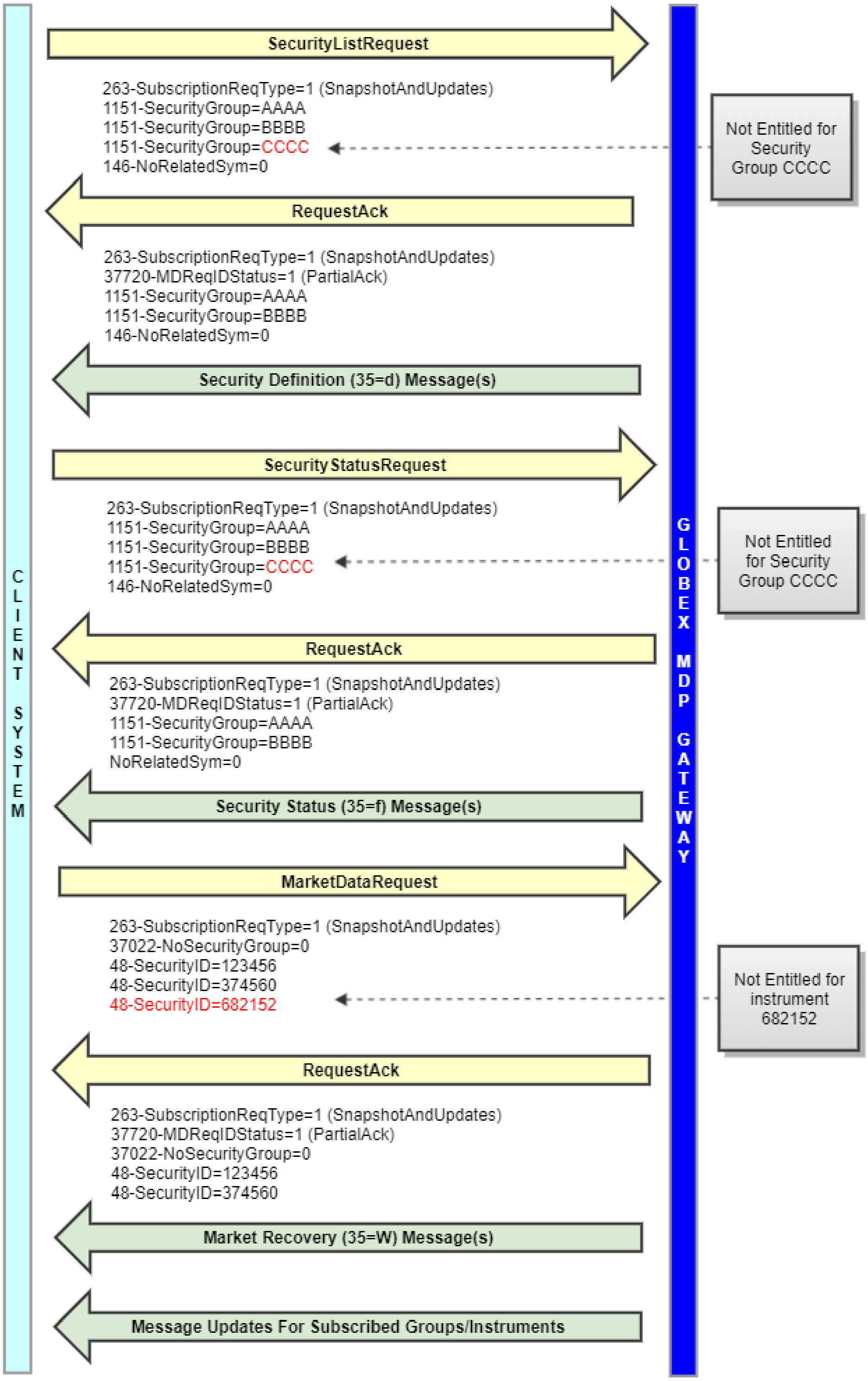
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroups=0
146-NoRelatedSym=0

Market Recovery (35=W) Message(s)

Message Updates For Subscribed Groups

Example 6 - Message Requests with Partial Acks

The following example provides a message workflow for all 3 request types (Security List, Security Status, and Market Data Request) with invalid tag 1151-SecurityGroup values. All requests are for Snapshot & Updates (tag 263-SubscriptionReqType=1) and are partially acknowledged (tag 37720-MDReqIDStatus=0). In the example, the invalid values are denoted in red.


CLIENT SYSTEM
GLOBEX MDP GATEWAY
SecurityListRequest
Not Entitled for Security Group CCCC
RequestAck
Security Definition (35=d) Message(s)
Security StatusRequest
Not Entitled for Security Group CCCC
RequestAck
Security Status (35=f) Message(s)
MarketDataRequest
Not Entitled for instrument 682152
RequestAck
Market Recovery (35=W) Message(s)
Message Updates For Subscribed Groups/Instruments

SecurityListRequest

263-SubscriptionReqType=1 (SnapshotAndUpdates)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
1151-SecurityGroup=CCCC
146-NoRelatedSym=0

RequestAck

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=1 (PartialAck)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
146-NoRelatedSym=0

SecurityStatusRequest

263-SubscriptionReqType=1 (SnapshotAndUpdates)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
1151-SecurityGroup=CCCC
146-NoRelatedSym=0

RequestAck

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=1 (PartialAck)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
NoRelatedSym=0

MarketDataRequest

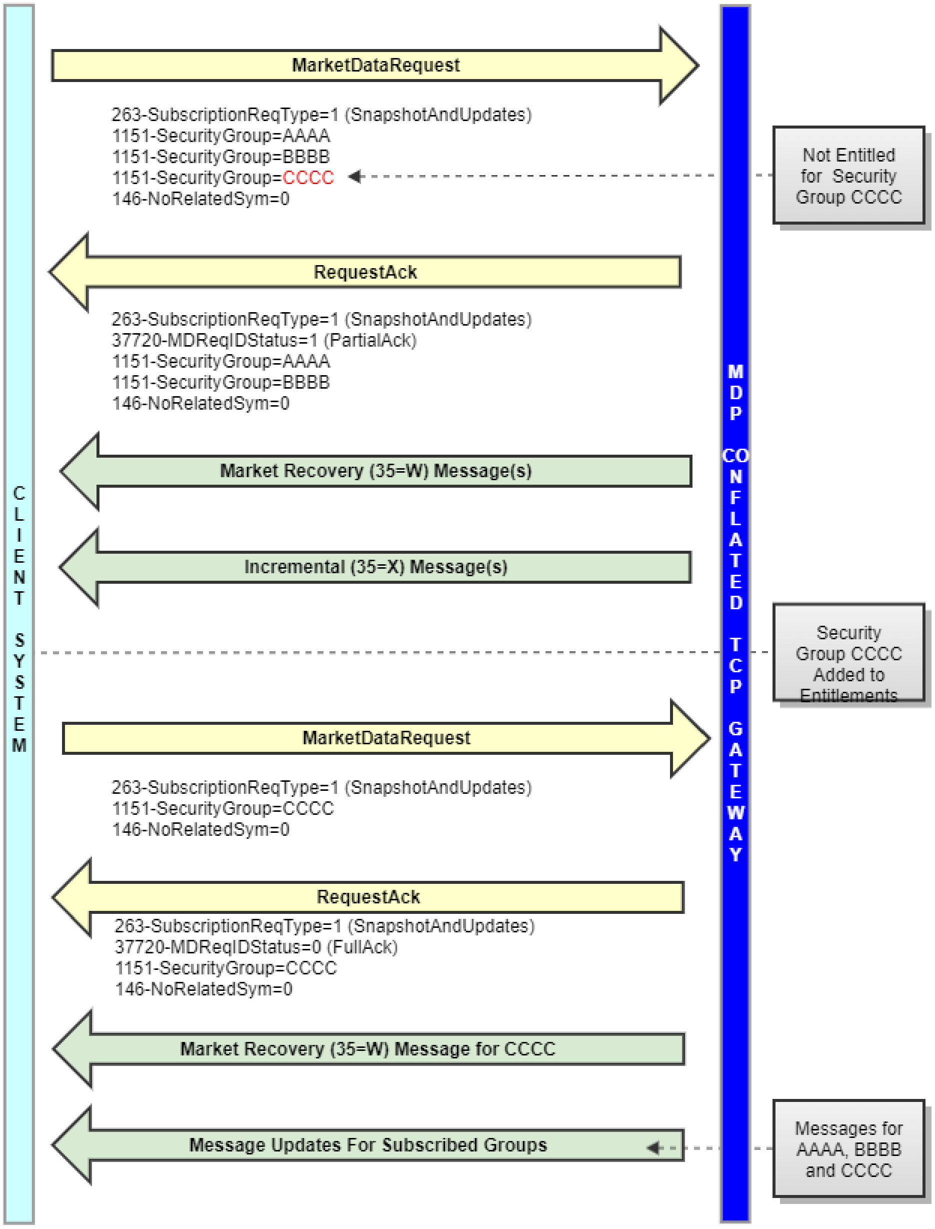
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0
48-SecurityID=123456
48-SecurityID=374560
48-SecurityID=682152

RequestAck

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=1 (PartialAck)
37022-NoSecurityGroup=0
48-SecurityID=123456
48-SecurityID=374560

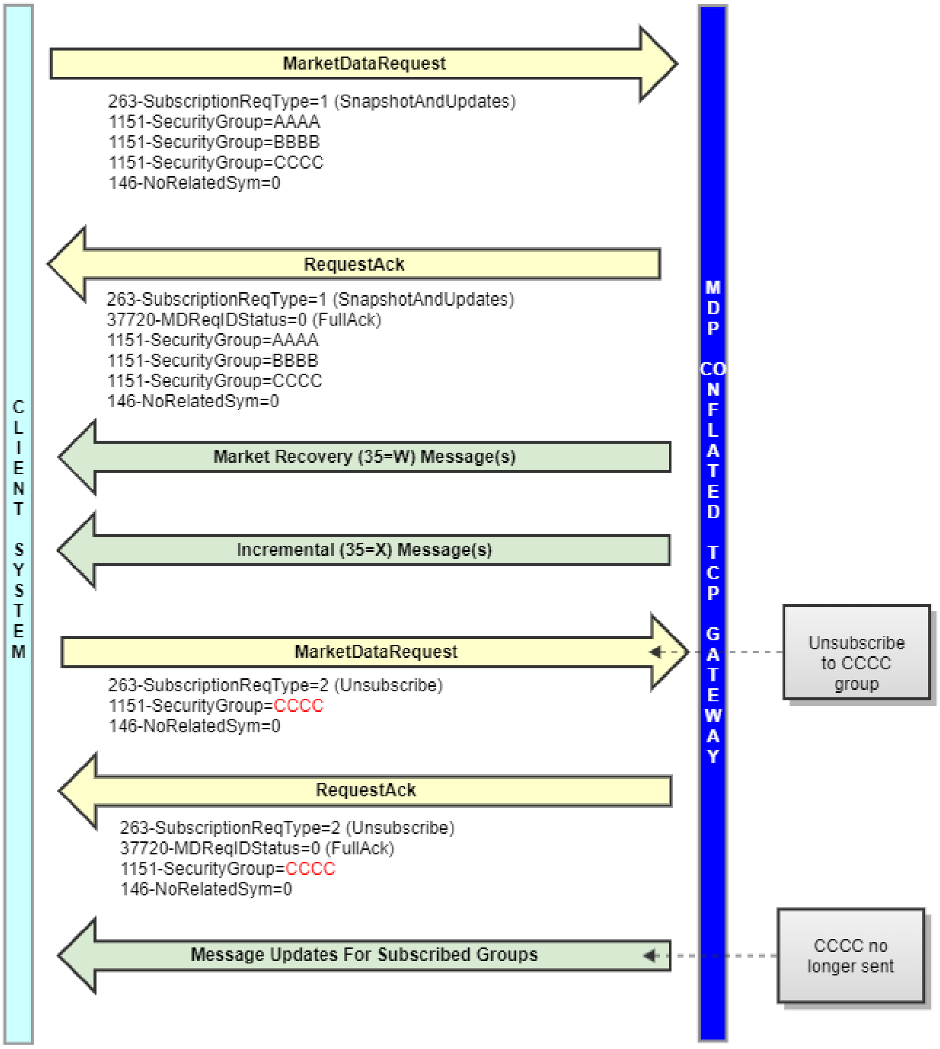
Example 7 - Mid-Week Entitlement Change

In this example, a firm's entitlements are updated mid-week. The session will not receive updates for the newly entitled group unless a request message is sent by the client system. Consequently, the client system sends a new request message to update the subscription and begins receiving data for the newly entitled group. In the example, the new group is denoted in red.


CLIENT SYSTEM
MDP CONFLATED TCP GATEWAY
MarketDataRequest
263-SubscriptionReqType=1 (SnapshotAndUpdates)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
Not Entitled for Security Group CCCC
RequestAck
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=1 (PartialAck)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
146-NoRelatedSym=0
Market Recovery (35=W) Message(s)
Incremental (35=X) Message(s)
Security Group CCCC Added to Entitlements
MarketDataRequest
263-SubscriptionReqType=1 (SnapshotAndUpdates)
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
RequestAck
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
Market Recovery (35=W) Message for CCCC
Message Updates For Subscribed Groups
Messages for AAAA, BBBB and CCCC

Example 8 - Unsubscribe Request

In this example a client system unsubscribes from a group. The unsubscribed group is denoted in red.

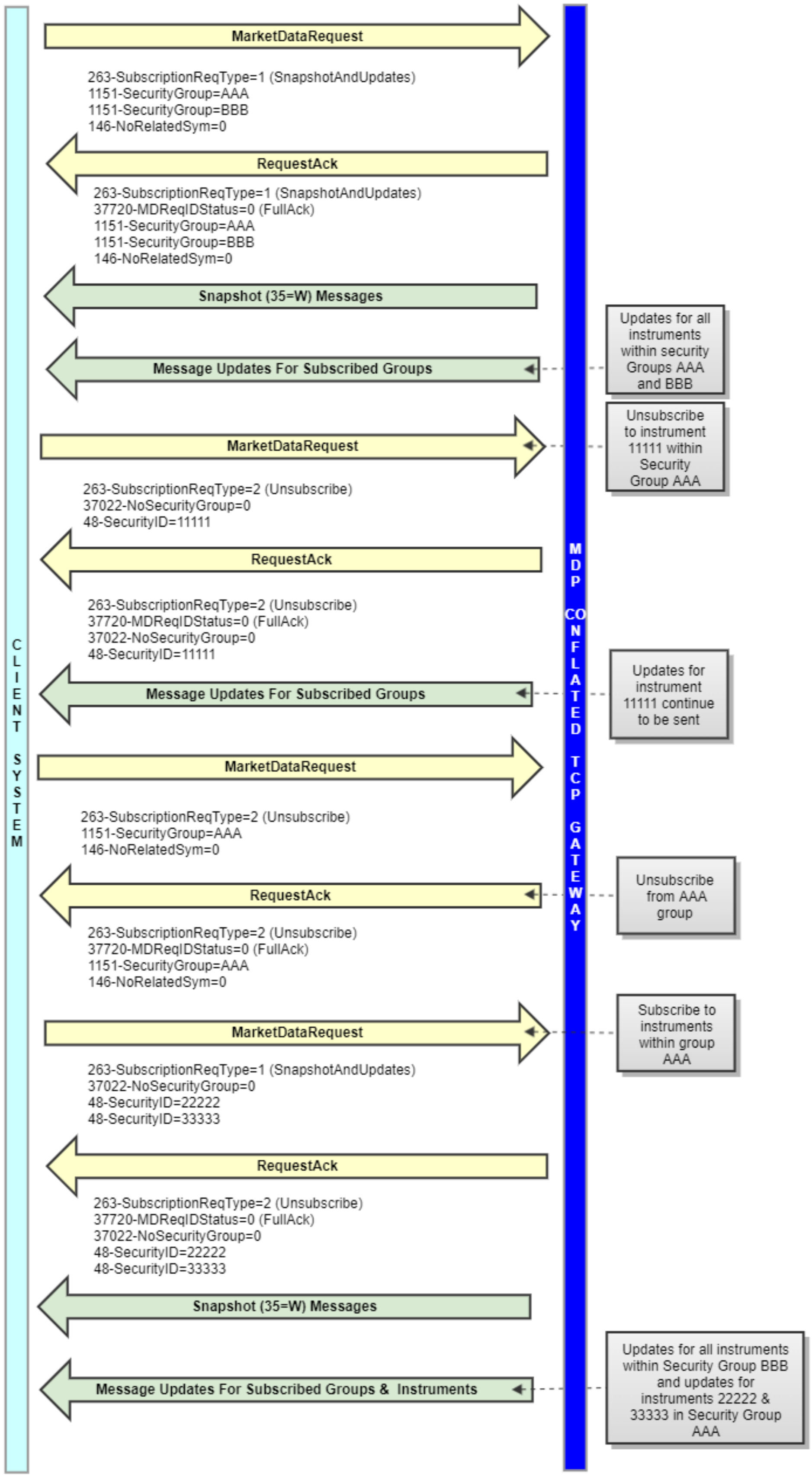

CLIENT SYSTEM
MDP CONFLATED TCP GATEWAY
MarketDataRequest
263-SubscriptionReqType=1 (SnapshotAndUpdates)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
RequestAck
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
1151-SecurityGroup=AAAA
1151-SecurityGroup=BBBB
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
Market Recovery (35=W) Message(s)
Incremental (35=X) Message(s)
MarketDataRequest
263-SubscriptionReqType=2 (Unsubscribe)
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
Unsubscribe to CCCC group
RequestAck
263-SubscriptionReqType=2 (Unsubscribe)
37720-MDReqIDStatus=0 (FullAck)
1151-SecurityGroup=CCCC
146-NoRelatedSym=0
Message Updates For Subscribed Groups
CCCC no longer sent

Example 9 - Subscription Example

The following example provides a message workflow of a request subscription. Assume a session is entitled to subscribe to the following security groups and instruments:

Security Group	Instrument
(Tag 1151)	(Tag 48)
Security Group 1	
AAA	11111
AAA	22222
AAA	33333
Security Group 2	
BBB	44444
BBB	55555
BBB	66666

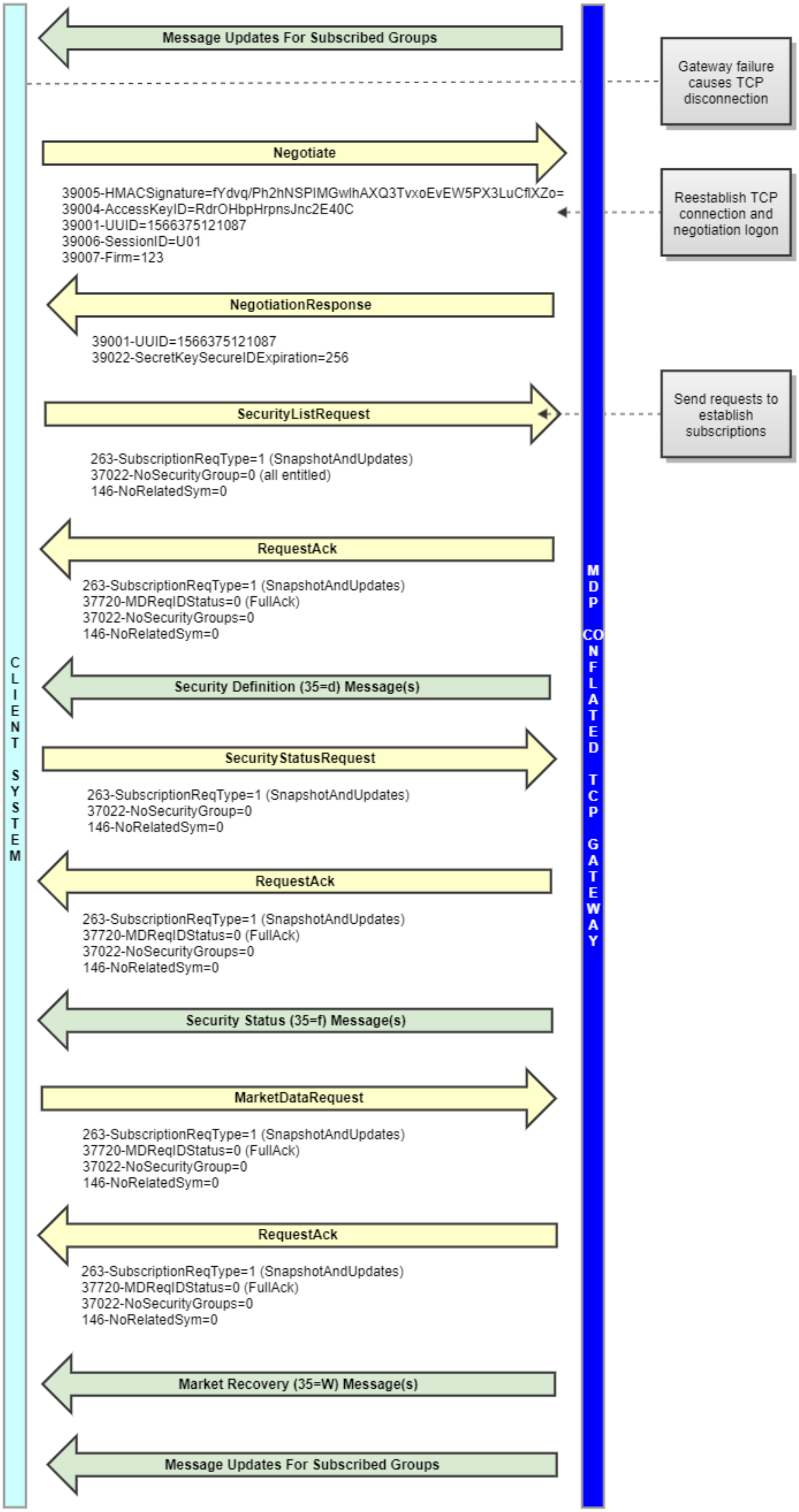
In the scenario below, a client system sends a market data request for both security groups (AAA and BBB). Next, the client system session unsubscribes to instrument 11111 in security group AAA which does not supersede the previous group request. Consequently, the client system session must unsubscribe to the security group AAA first and then individually subscribe to instruments 22222 and 33333 within security group AAA.


CLIENT SYSTEM
MDP CONFLATED TCP GATEWAY
MarketDataRequest
263-SubscriptionReqType=1 (SnapshotAndUpdates)
1151-SecurityGroup=AAA
1151-SecurityGroup=BBB
146-NoRelatedSym=0
RequestAck
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
1151-SecurityGroup=AAA
1151-SecurityGroup=BBB
146-NoRelatedSym=0
Snapshot (35=W) Messages
Message Updates For Subscribed Groups
Updates for all instruments within security Groups AAA and BBB
MarketDataRequest
Unsubscribe to instrument 11111 within Security Group AAA
263-SubscriptionReqType=2 (Unsubscribe)
37022-NoSecurityGroup=0
48-SecurityID=11111
RequestAck
263-SubscriptionReqType=2 (Unsubscribe)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroup=0
48-SecurityID=11111
Message Updates For Subscribed Groups
Updates for instrument 11111 continue to be sent
MarketDataRequest
263-SubscriptionReqType=2 (Unsubscribe)
1151-SecurityGroup=AAA
146-NoRelatedSym=0
RequestAck
Unsubscribe from AAA group
263-SubscriptionReqType=2 (Unsubscribe)
37720-MDReqIDStatus=0 (FullAck)
1151-SecurityGroup=AAA
146-NoRelatedSym=0
MarketDataRequest
Subscribe to instruments within group AAA
263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0
48-SecurityID=22222
48-SecurityID=33333
RequestAck
263-SubscriptionReqType=2 (Unsubscribe)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroup=0
48-SecurityID=22222
48-SecurityID=33333
Snapshot (35=W) Messages
Message Updates For Subscribed Groups & Instruments
Updates for all instruments within Security Group BBB and updates for instruments 22222 & 33333 in Security Group AAA

Example 10 - Recovery

In the unlikely event of a CME Group Conflated MDP TCP Gateway failure, client systems must reestablish a TCP connection, negotiate and re-subscribe via request messages. In this example, once a FIX connection has been terminated, Subscription requests and book states are reset. To ensure data there are no in-flight data issues, during a recovery, CME recommends sending request messages in the following order with tag 263-SubscriptionReqType=1 (Snapshot and Updates):

1. Security List Request Message (tag 35-MsgType=x)
2. Security Status Request Message (tag 35-MsgType=g)
3. Market Data Request Message (tag 35-MsgType=V)


CLIENT SYSTEM
MDP CONFLATED TCP GATEWAY
Message Updates For Subscribed Groups
Gateway failure causes TCP disconnection
Negotiate
Reestablish TCP connection and negotiation logon

39005-HMACSignature=fYdvq/Ph2hNSPIMGwlhAXQ3TvxoEvEW5PX3LuCflXZo=
39004-AccessKeyID=RdrOHbpHrpnsJnc2E40C
39001-UUID=1566375121087
39006-SessionID=U01
39007-Firm=123

39001-UUID=1566375121087
39022-SecretKeySecureIDExpiration=256

SecurityListRequest
Send requests to establish subscriptions

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0 (all entitled)
146-NoRelatedSym=0

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroups=0
146-NoRelatedSym=0

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37022-NoSecurityGroup=0
146-NoRelatedSym=0

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroups=0
146-NoRelatedSym=0

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroup=0
146-NoRelatedSym=0

263-SubscriptionReqType=1 (SnapshotAndUpdates)
37720-MDReqIDStatus=0 (FullAck)
37022-NoSecurityGroups=0
146-NoRelatedSym=0

Market Recovery (35=W) Message(s)
Message Updates For Subscribed Groups