Ai 4.0 Developer's Guide

EBS®™ Spot Ai

XML 3.0

Table of Contents

Table of Figures

1 Introduction

1.1 Document Layout

This document is arranged in sections, which are aligned to the Uses Cases, from the client's perspective. A text version of the XML schema and comprehensive list of Error messages can be found in the Appendices.

Each of the sections defines the details of one or more message exchanges (Use Cases), and includes the following subsections:

Configuration Parameters: If there are parameters that will affect the functionality of individual messages, they are included in a Configuration Parameters table. If no parameters exist, the subsection is not included.

Request Message: This subsection defines the format of the Request message that will be sent by the Client application. It includes a table showing all elements that apply, whether they are mandatory or optional, along with explanations of what values each element or attribute must contain.

Response Message: Once the client sends a Request message, the Ai Server will send a Response message. This subsection details the format of the Response message, and all the elements and attributes the client can expect to find in it, along with explanations.

Event Messages: Some sections describe Event messages which are sent by the Ai Server due to changes in the market, deals, or session event. These sections do not include Request and Response message subsections, but instead define the format of the Event message.

Error Messages: If there are problems of any kind with the Request message received from the client, the AI Server responds with various Error messages. This subsection includes the most common Error messages, along with descriptions. A complete list of Error messages can be found in the Appendices.

Diagram: Most sections also include a high-level Activity Diagram to illustrate (or make a little clearer) the functionality of the message exchange for that section. Where the message exchange is simple or straight forward, a diagram is not necessary.

1.2 Message Types

Communicating with the Ai Server revolves entirely around the messaging structures (Ai XML 3.0), which is divided into three types of messages:

- Request messages
- Response messages
- Event messages

Request/Response Messages – Request messages are sent by the Client application to the Ai Server, which will then send a Response message. Heartbeat messages are unique in that they are Request messages that can be sent by either the Ai Server, or the Client application, to which the other side must respond.

Event Messages – Event messages are triggered by a status change in an Order or Deal, by changes in market prices, session changes or system errors, and are initiated by the Ai Server.

1.3 Useful Reference Material

Introduction to EBS Spot
Guide to ICAP Pair Parameters
ICAP Spot Ai 4.0 What's New
ICAP Spot Ai 4.0 Technical Overview
ICAP Spot Ai 4.0 Protocol Schema
ICAP Spot Ai 4.0 Lab Guide

1.4 Client Functionality

In order to communicate with the Ai Server, the client Application must properly implement the following functionality:

- Connect
- Heartbeat
- Login
- Change Password
- Cancel Other Session
- Subscribe & Market View
- Unsubscribe
- Order Submit
- Order Interrupt
- Order Event
- Deal Event
- Interrupt All
- Deal Query
- Session Event
- Logoff

Each of the functions listed above involves XML message exchanges, either Request/Response or Event messages, and are described in detail in the following sections.

The diagram below shows the Use Cases associated with the messaging functions from the client's perspective.

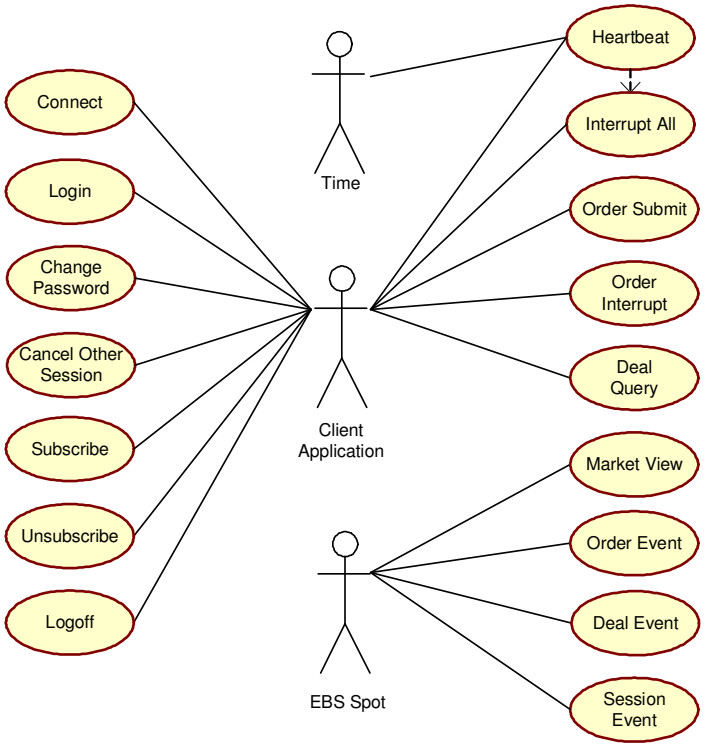

Figure 1 - Communications Use Cases

2 XML Messages

2.1 Message Format

Prior to any messages being exchanged between the Ai Server and the Client application, a socket connection must be established by the Client application to the Ai Server. The only requirement is that the connection is made on the configured port.

All Ai messages are contained within the main element Ati.

Element Name	Attribute or Value	Description
Ati mandatory		Contains the timestamp attribute and all other elements for all message types
	timestamp=*DateTime* mandatory	The Date and the Time should be formatted as: yyyy-mm-ddThh:mm:ss.zzzZ The Ai Server does not check the formatting of the timestamp. It is only used for logging and troubleshooting when necessary.

Messages are of variable length. The length of the XML document must be written to the stream immediately before the document itself. Likewise, the message length must be read from the socket in order to read the correct number of bytes from the stream for the XML document. The message length is defined as a four-byte unsigned integer, which is followed by the XML document of the specified size. Response messages, as well as Event messages from the Ai Server will be in the same format.

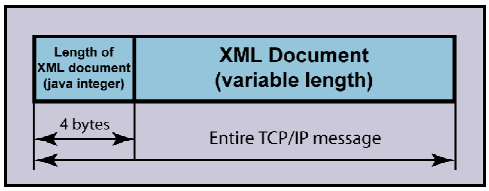

Figure 2 - Ai message definition

The length field contains the length of the XML document only. It should not include itself. Network restrictions prohibit a message length exceeding 16,384 bytes (16KB).

2.2 Connect

2.2.1 Client Socket

Prior to any message exchanges between the Ai Server and the client, a socket connection must be created and opened, using the following parameters:

Host: The **IP address** of your Ai Server.

Port: The port number for XML messages is **4000**. For clients with multiple Ai Servers, the table below shows the port number for each Ai Server instance.

Ai Server	XML Port
1st instance	4000
2nd instance	4010
3rd instance	4020
4th instance	4030
5th instance	4040
6th instance	4050
7th instance	4060
8th instance	4070
9th instance	4080
10th instance	4090

Note: Client must ensure the correct port is used for the particular Ai Server.

2.3 Heartbeat

Once Logged in, the Ai Server sends Heartbeat messages to monitor the status of the connection. If the Client application does not respond to Heartbeat messages for more than 3 seconds, the connection is considered stale, and the Ai Server will Interrupt all client's Orders. If the client does not respond for more than 8 seconds, the Ai Server terminates the session and disconnects the client.

Note: If communication is re-established before the dead interval (8seconds) elapses and before the client is disconnected, the session will continue.

2.3.1 Heartbeat Request

Heartbeat Request Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
HeartbeatRequest	sequence mandatory	number	The sequence number as assigned by the initiator of the message. The recipient must respond with the same sequence number.

Heartbeat Request Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <HeartbeatRequest sequence="304" />
</Ati>
```

2.3.2 Heartbeat Response

Heartbeat Response Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Heartbeat	sequence	number	The sequence number as assigned by the initiator of the message. The recipient must respond with the same sequence number.

Heartbeat Response Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
   <Heartbeat sequence="304" />
</Ati>
```

2.3.3 Heartbeat Error Messages

The only Heartbeat error message is the Protocol Violation message which means that the Request message was formatted incorrectly. This can be due to the four-byte size header specifying a size different than what was received, element names that are capitalized incorrectly or are misspelled, or other XML formatting problems.

Heartbeat Error Messages

Error	Description	Resolution
ATI100901 Protocol Violation	The Request message was formatted incorrectly. The session is terminated and the client is disconnected.	Client must correct the format of the XML message and reconnect and re-login.

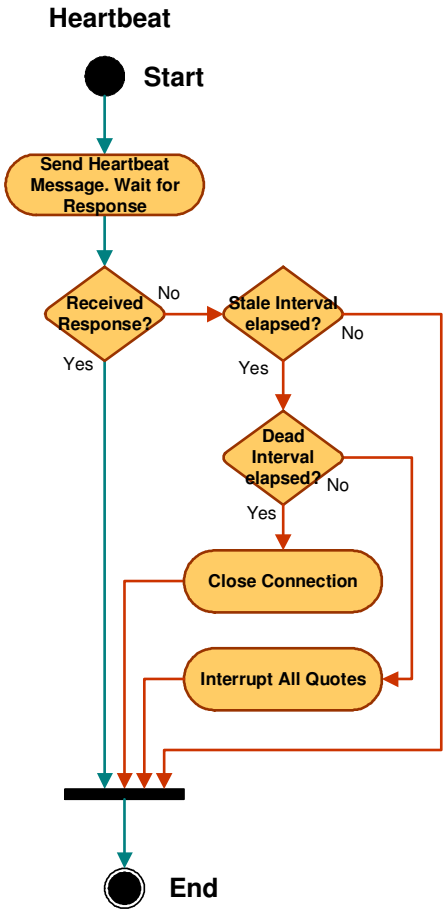

Figure 3 - Heartbeat Message

2.4 Login

After creating a socket connection to the Ai Server, the Client application can then Login to begin a Trading Session. The Login process also allows the client to configure certain parameters that will be in effect for the duration of the session.

2.4.1 Login Request

The table below represents the elements and formatting of the Login Request message. The **Param** element is used to hold all Configurable parameters which can be set by the client, and are valid for the duration of the session. In order to change any of these parameters the client must close the existing session and start a new session.

Login Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Login	username mandatory	Trader ID	Client's 3-character Trader ID
	password mandatory	password	Associated password for the Trader ID used by the Ai Server. 8—16 characters. Not Case sensitive. When first issued, it is trading floor default password.
	aiProtocolVersion mandatory	protocol version (3.0)	The Ai version of the XML protocol that will be used for this session. This value must agree with the Ai version value to log in successfully (e.g., "3.0").
Param	optional		The **Param** element is repeated for each individual parameter. If no parameters are included, the element is not included.
	name="AutoCancelDuplSession" optional	value="true" or "false"	If True, when the Login Request is sent, any existing trading session (using the same Trader ID) will be terminated. If False, no action taken until client Requests session to be terminated. The default setting is "False". If this parameter is not included in the message, it will be interpreted as "false".
	name="PriceCheck" optional	value="true" or "false"	When submitting a bid/offer, a check is performed to determine if the price is acceptable and not inverted. In the ICAP FX Spot trading system, this check is known as "Bid Greater than Offer," but it applies to both sides of an order. (Refer to the *EBS FX Spot Dealing Rules* for more details.) Default is **true**.

Element Name	Attribute	Value	Description
	name=”LargeDifferenceCheck” optional	value=“true” or “false”	If LargeDifferenceCheck is enabled, Ai server will reject a bid price that varies (higher or lower) from the current Dealable Best Bid by more than the large difference pips for the currency pair. Also rejected is an offer price that varies (higher or lower) from the current Dealable Best Offer by more than the large difference pips for the currency pair. If Large Difference check is disabled, no restriction in effect and order accepted. Default is **true**.
	name=”WideSpreadCheck” optional	value=“true” or “false”	If WideSpreadCheck is enabled, the Ai server ensures that a bid price less than the current Dealable Best Offer by more than the wide spread pips is rejected, and an offer price that exceeds the current Dealable Best Bid by more than the wide spread pips is rejected. If Wide Spread is disabled, no restriction is in effect and the order accepted. Default is **true**.
	name=”HideMyPrices” optional	value=“true” or “false”	This feature is only in effect if the TFA has already allowed any trader on the floor to select this option. If **HideMyPrices** is enabled, the trader ID will not appear to other traders on the floor. If another trader on the same floor has this parameter enabled, his Trader ID will not appear in market view updates. Default is **false**.
	name=”LocalPriceDisplay” optional	value=“true” or “false”	When “LocalPriceDisplay” is enabled, local bid and offer prices are displayed in Market Views for the Ai client. If disabled, local bid and offer prices are suppressed. The local price and trader ID are replaced with “NoPrice”. This parameter only affects the information shown in the market update to the client. Default is **true**.
	name=”SendConfirmedDeals” optional	value=“true” or “false”	An Ai client may request to receive or prevent the notification of a deal when confirmed. If this parameter is enabled, a client receives a “pending,” “confirmed” and “done” sequence of messages for deal processing. If disabled, the “confirmed” message is not sent to the client. Default is **false**.

Simple Login example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Login username="TR3" password="123456" aiProtocolVersion="3.0" />
</Ati>
```

Example with AutoCancelDuplSession, and WideSpreadCheck parameters

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Login username="TR3" password="123456" aiProtocolVersion="3.0">
      <Param name="AutoCancelDuplSession" value="true" />
      <Param name="WideSpreadCheck" value="false" />
    </Login>
</Ati>
```

2.4.2 Login Response

The Ai Server responds to Login Request messages with a Login Response message. The table below represents the elements and formatting of the Login Response message. The **Param** element is used to hold information about the session that the client may want to log, but cannot be changed.

Login Response Message

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
LoginOk	sessionId=”1234”	ID	Session ID assigned by the system when login successfully starts an active trading session. Client can use the Session ID when contacting Customer Support and for logging to identify the session when and if a problem occurs.
Param			The **Param** element is repeated for each of the parameters listed below.
	name=”TotalNumberOfOrders”	value=”20” default	The maximum number of Orders the client can have active in the market at any one time.
	name=“NumberOfOrders”	value=”10” default	The maximum number of Orders that can be submitted within the time specified by the **NumberOfOrdersTimeInterval** parameter (below).
	name=”NumberOfOrdersTimeInterval”	value=”5000” default	Used with **NumberOfOrders**. In this example it means the client cannot submit more than 20 Orders in any five second interval.
	name=”AiHostName”	value=*hostname*	The Host name of the Ai Server.
	name=“AiPort”	value=”4000” default	The Ai server port number. If there are multiple Ai Servers installed on one machine, the port number is used to identify each Ai

Element Name	Attribute	Value	Description
			instance.
Instrument			The **Instrument** element appears for each of the Currency Pairs to which the client's Trading Floor is entitled, with the attributes show below.
	InstrumentId	ID	The ID for the particular Currency Pair. This number must be used when subscribing to Market Views or when placing Orders. The Ai Server does not use the Currency Pair directly.
	base	Currency	The Base Currency. Use the base and local currency to create a correlation between the Instrument ID and the Currency Pair. Ai Server only uses and recognizes the Instrument ID.
	local	Currency	The Local Currency. Used along with base, as described above.
	regSize	amount	The Regular Order Amount Size. Refer to Guide to ICAP Pair Parameters for more information.
	minSize	amount	The minimum Order size for this Instrument ID (Currency Pair)
	maxSize	amount	The maximum Order size for this Instrument ID (Currency Pair)
	sizeIncrement	amount	From the Minimum size Order size can be increased up to the maximum in increments equal to this amount.
	priceIncrement	price	Order prices must be specified in increments equal to this amount.
	outsidePriceSpread	price	The difference in pips between the best price and the outside price.
	outsidePriceAmount	amount	The depth of liquidity at the Outside price will be shown up to this amount. If there is 150000000 in the market at the Outside price, and the **outsidePriceAmount** is 50000000, the **outsidePriceAmount** is shown. If depth of liquidity is less than the **outsidePriceAmount**, then the actual amount is shown.
	bestPriceAmount	amount	The depth of liquidity at the Dealable Best price will be shown up to this amount. If the Dealable Best price is less than this amount, then the actual amount is shown.
	bestPricePlusAmount	amount	The depth of liquidity at the Dealable Best Price Plus will be shown up to this amount. If the Dealable Best Price Plus is less than this amount, then the actual amount is shown.
	wideSpread	price	The value of the **WideSpreadCheck** parameter.
	largeDifference	price	The value of the **LargeDifferenceCheck**

Element Name	Attribute	Value	Description
			parameter.
	xPips	price	The amount an Order price can deviate (when inverted) from the Best Price before it is rejected. Applies if the **PriceCheck** parameter is disabled. If **PriceCheck** is enabled, all Orders with inverted prices are rejected.
	valueDate	Date	The **Settlement Date** for SPOT Currency Pairs. For NDFs, it is the Date when the near value is established.
	tradeDate	Date	The Date when the Trade was executed.
	type	NDF	Specifies that the Currency Pair is NDF. Only for NDF Instruments.
	tenor	xW, xM, xY	Specifies the Tenor of the NDF in Weeks, Months or Years, where x denotes the number value. Ex: 3M = three months 2Y= two years Only for NDF Instruments
	fixingDate	Date	The Date when the forward value of the NDF is established. Only for NDF Instruments
	settlementDate	Date	The Settlement Date for NDFs. Only for NDF Instruments

Successful Login Response example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
  <LoginOk sessionId="fb56">
    <Param name="TotalActiveOrders" value="20"/>
    <Param name="NumberOfOrders" value="10"/>
    <Param name="NumberOfOrdersTimeInterval" value="5000"/>
    <Param name="AiHostName" value="USNJAFORAYD1"/>
    <Param name="AiPort" value="4000"/>
    <Instrument instrumentId="7442" base="USD" local="JPY" regSize="15000000"
        xPips="0.10000000" wideSpread="0.30000000" priceIncrement="1" largeDiff="0.20000000"
        minSize="1000000" maxSize="30000000" sizeIncrement="1000000" valueDate="2007-08-
        28" tradeDate="2007-08-24"/>
    <Instrument instrumentId="13085" base="XPD" local="USD" regSize="3000"
        xPips="0.50000000" wideSpread="0.50000000" priceIncrement="5" largeDiff="0.50000000"
        minSize="500" maxSize="10000" sizeIncrement="500" valueDate="2007-08-28"
        tradeDate="2007-08-24"/>
    <Instrument instrumentId="7471" base="USD" local="KRW" regSize="10000000"
        xPips="0.20000000" wideSpread="0.60000000" priceIncrement="1" largeDiff="0.40000000"
        minSize="1000000" maxSize="50000000" sizeIncrement="1000000" valueDate="2007-08-
        28" tradeDate="2007-08-24" type="NDF" tenor="1M" fixingDate="2007-09-21"
        settlementDate="2007-09-28"/>
    <Instrument instrumentId="8717" base="EUR" local="GBP" regSize="10000000"
        xPips="0.0010000000" wideSpread="0.0030000000" priceIncrement="5"
        largeDiff="0.0020000000" minSize="1000000" maxSize="30000000"
        sizeIncrement="1000000" valueDate="2007-08-29" tradeDate="2007-08-24"/>
    <Instrument instrumentId="10013" base="XAG" local="USD" regSize="100000"
        xPips="0.10000000" wideSpread="0.30000000" priceIncrement="25"
        largeDiff="0.20000000" minSize="50000" maxSize="300000" sizeIncrement="50000"
        valueDate="2007-08-29" tradeDate="2007-08-24"/>
    <Instrument instrumentId="7488" base="USD" local="RUB" regSize="10000000"
        xPips="0.10000000" wideSpread="0.0030000000" priceIncrement="1"
        largeDiff="0.0020000000" minSize="1000000" maxSize="30000000"
        sizeIncrement="1000000" valueDate="2007-08-27" tradeDate="2007-08-24" type="NDF"
        tenor="1M" fixingDate="2007-09-26" settlementDate="2007-09-27"/>
    <Instrument instrumentId="9757" base="XAU" local="USD" regSize="5000"
        xPips="0.10000000" wideSpread="3.0000000" priceIncrement="1" largeDiff="2.0000000"
        minSize="1000" maxSize="30000" sizeIncrement="1000" valueDate="2007-08-29"
        tradeDate="2007-08-24"/>
    <Instrument instrumentId="3357" base="GBP" local="USD" regSize="5000000"
        xPips="0.0015000000" wideSpread="0.0030000000" priceIncrement="1"
        largeDiff="0.0020000000" minSize="1000000" maxSize="30000000"
        sizeIncrement="1000000" valueDate="2007-08-29" tradeDate="2007-08-24"/>
      <Instrument instrumentId="8733" base="EUR" local="USD" regSize="15000000"
        xPips="0.0010000000" wideSpread="0.0030000000" priceIncrement="1"
        largeDiff="0.0020000000" minSize="1000000" maxSize="999000000"
        sizeIncrement="1000000" valueDate="2007-08-28" tradeDate="2007-08-24"/>
  </LoginOk>
</Ati>
```

Note: Actual Parameter Values may be different than in the example above. Refer to the Guide to ICAP Pair Parameters for more information.

2.4.3 Login Error Message

Several type of Error messages exist associated with the Login process, as shown in the table below.

Login Error Messages

Error	Description	Resolution
ATI300201 Already Logged In	The Trader ID is being used in another session.	The client must send a Cancel Other Session message to close the existing session and open a new one. Although the Login fails, it remains in pending state and will continue after the client sends a **Cancel Other Session** message, as described in the Cancel Other Session section. An alternative solution is to disconnect and Login again with the **AutoCancelDuplSession** parameter set to true.
ATI300202 Change Password	Default password used for Login, password change required.	Send a Password Change message with a new Password. The Login process remains in pending state and will continue after a successful password change. Refer to the Change Password section for more information.
ATI300009	Bad Credentials (Invalid password).	Verify that the password used for this account is the correct one. Retry the Login.
ATI300009	No such user XXX (Invalid Trader ID).	The Trader ID used in the Login does not exist on the system. Verify the Trader ID and retry the Login.
ATI300009	Too many signon attempts. Password suspended and must be reset before signing on again.	Contact the Trade Floor Administrator to verify and reset the password. Retry the Login.
ATI100901 Protocol Violation	The Request message was formatted incorrectly, or the client is using the wrong version protocol. The client is disconnected.	Client must correct the format of the XML message and reconnect and re-login.

Note: See the Appendices for a list of all Error Codes

Login Failed Response Message

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
LoginFailed	msgId	messageID	The specific error message ID.
	msg	text	Text describing the Error (e.g. “Bad Credentials”, see table above).

Failed Login Response example (Invalid userID)

```
<Ati timestamp="2007-07-17T17:26:37.873Z">
    <LoginFailed msgId="ATI300009" msg="No such user AI1"/>
</Ati>
```

Failed Login Response example (Invalid Password)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <LoginFailed msgId="ATI300009" msg="Bad Credentials" />
</Ati>
```

Failed Login Response example (Already Signed In)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <AlreadyLoggedIn />
</Ati>
```

Note: The failed Login due to Protocol Violation happens if the there is a protocol version mismatch, but it can also happen if the Client application sends a message that is incorrectly formatted, contains the wrong elements, or the size specified at the beginning of the message does not match the actual message size.

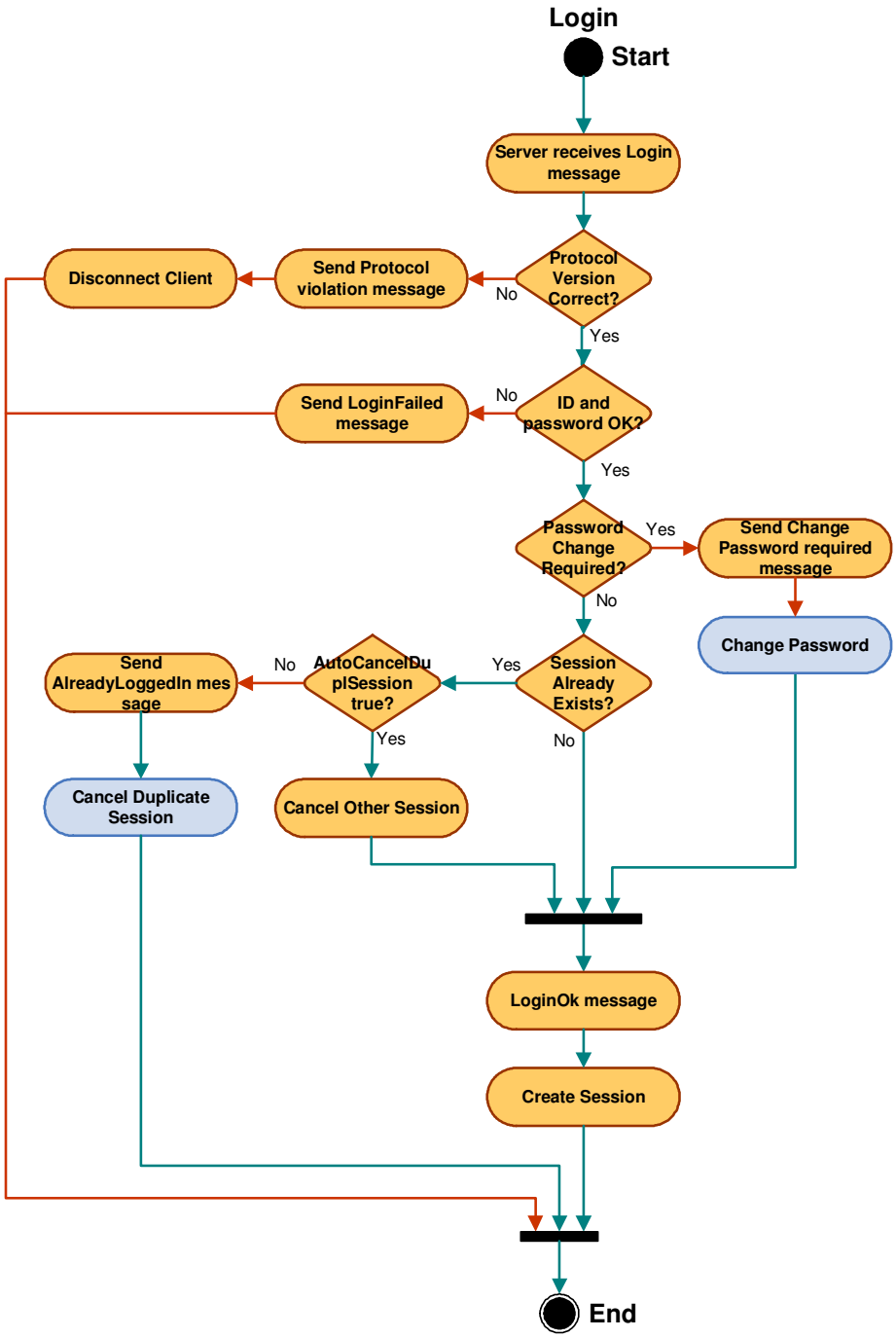

Figure 4 - Login Process Flow

2.5 Change Password

When logging into the Ai Server for the first time, the trading floor default password must be changed to a user selected password. This is also required if a password needs to be reset or it expires. The Login is placed in a pending state until the client changes the password. Upon successful password change, the Login process continues.

The client can also send a Change Password Request to change the password (after the initial change from the default password) at any time during an active session for any reason. The password must meet security requirements to be accepted.

2.5.1 Change Password Request

Change Password Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
ChangePassword	oldPassword mandatory	password	Default Password as issued by the Trading Floor Administrator (TFA), or the previous user selected password. This must be the same as the password used in the Login message; otherwise a Bad Credential response will be received. When changing the password during the Login process (when changing the default password), the oldPassword field is optional. It is mandatory only when changing a use selected password after the Login, during a session.
	newPassword mandatory	password	The new Password must be 8–16 characters. Letters are NOT case sensitive.

Password Change Request example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <ChangePassword oldPassword="123456" newPassword="654321" />
</Ati>
```

2.5.2 Change Password Response

Change Password Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time is the following format: yyyy-mm-ddThh:mm:ss.zzzZ
ChangePasswordOk			The element does not contain any values. Its presence signifies a successful Password Change.

Successful Password Change Response message

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <ChangePasswordOk />
</Ati>
```

2.5.3 Change Password Error Messages

Several types of Error messages associated with the Change Password process exist, as shown in the table below.

Login Error Messages

Error	Description	Resolution
ATI300211 Change Password Failed	Invalid New Password.	The password did not pass minimum security requirements as to length and composition.
ATI300211 Bad Credentials	The Old Password included in the message was not the correct password for the account.	Verify the password and resend the Password Change request.
ATI100901 Protocol Violation	The Request message was formatted incorrectly. The session in terminated and the client is disconnected.	Client must correct the format of the XML message and reconnect and re-login.

Note: See the Appendices for a list of all Error Codes

Change Password Error Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
ChangePasswordFail	msgId	error ID	Numeric error ID assigned by the system.
	msg	text	A description of the Password Change failure.

Invalid New Password Response message

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <ChangePasswordFail msgId="ATI300211" msg="Password was not changed!" />
</Ati>
```

Invalid Old Password Response message

```
<Ati timestamp="2007-07-17T20:47:48.555Z">
      <ChangePasswordFail msgId="ATI300211" msg="Bad Credentials!"/>
</Ati>
```

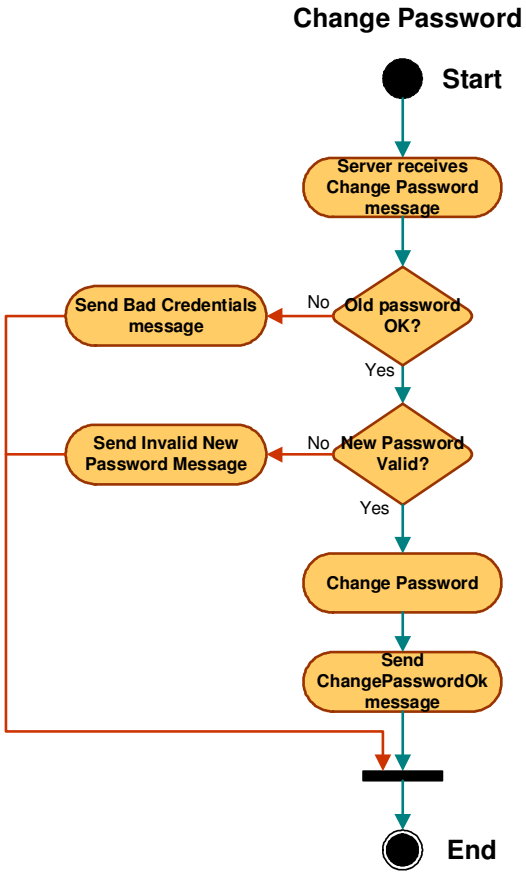

Figure 5 - Password Change Diagram

2.6 Cancel Other Session

A Trading Floor can have many Trader IDs. The Ai Client application can use any of the available IDs; there are no special IDs used only by Ai Clients. Because of this, it is possible that the Trader ID used by the Client application may already be used by another Trader on the floor. Client should contact the Trading Floor Administrator before cancelling an existing session, to ensure the correct Trader ID is being used.

Although an existing session can be cancelled through the Login message, it can also be cancelled during the Login process, after the client receives an **AlreadyLoggedIn** message from the Ai Server.

2.6.1 Cancel Other Session Request

If the Login fails because another session using the same Trader ID is active, the Client application can instruct the Ai Server to close the existing session.

This message can only be sent after the Login returns an **AlreadyLoggedIn** Login failure. The Login remains in a pending state until the Client application sends a Cancel Other Session Request message. When the Ai Server receives the request, it terminates the other session and the Login process continues.

The Client application can also close the connection and re-send the Login Request message, this time with the **AutoCancelDuplSession** parameter set to true.

Cancel Other Session Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
CancelOtherSession	none mandatory	none	The presence of this element instructs the Ai Server to close the other session. The Ai Server closes the other session and continues the Login process.

Cancel Other Session Request Message Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <CancelOtherSession />
</Ati>
```

There is no actual response to this message. The other session is terminated and the Login process continues. The client will receive the **Login Response** message as described in the Login section.

2.7 Subscribe and Market View

Subscribing to Market Views is not required in order to trade.

The Ai Server must be configured to allow the Client application to subscribe to Market View messages, as shown in the table below.

Market View parameters

Parameter Name	Default value	Description
market_view_enabled	true	The default value of **true** will allow the Client application to subscribe and receive Market Views. If this parameter is set to **false**, the Ai Server will reject subscription Requests. This parameter is set by EBS Operations and cannot be changed by the client.

Note: If there are multiple Ai Servers on the Trading Floor it may be necessary to create the Client application such that it connects to two Ai Servers: one which allows only Market View Subscriptions, and one that allows trading.

2.7.1 Subscribe Request

In order to receive Market Views, the client application must subscribe for them. The subscription is based on the Instrument IDs received in the Login Response message. The client must send a Subscribe message for each individual Instrument ID.

Subscribe Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Subscribe			Subscribe messages must be sent individually for each Instrument..
	instrumentId=”xxxx” mandatory	ID	Each Currency Pair has an associated Instrument ID. The ID can be retrieved from the **Login** message response, which includes all Currency Pairs that are enabled for the client’s Trading Floor.

Subscribe Request Message Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Subscribe instrumentId="2143" />
</Ati>
```

No actual response exists for the Subscribe Request. The Ai Server simply starts sending Market Views, one at a time, for each Instrument ID. The first Market View sent is a snapshot message that includes all elements, whether or not a value is currently available.

Subsequent messages are updates, which only include the elements that have been updated. Market Views are generated from new Orders and Deal Information in the market.

Market View Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
MarketVIew	instrumentId	ID	The Instrument ID for the particular Currency Pair.
Data			The Data element is repeated for each of the following parameters.
	name	”EbsBestBid”	The Best Bid regardless of the amount or credit. If a credit arrangement does not exist with the counterparty, a trade is not possible and will be rejected.
		”DealableBestBid”	The Best Bid which can be dealt against by the client. The amount may be lower than the Regular Amount for that Instrument.
		”DealableBestPlusBid”	Part of the Depth of Book feature, The Best Dealable Plus Bid will show the next best price in the book.
		”DealableRegularBid”	A bid that is with the Regular Amount and can be traded.
		”DealableOutsideBid”	Part of the Depth of Book feature, the Outside Bid will show the amount X pips away from the Dealable Best Bid. The number of pips (X pips) depends on the particular currency pair.
		”EbsBestOffer”	The Best Offer regardless of the amount or credit. If a credit arrangement does not exist with the counterparty, a trade is not possible and will be rejected.
		”DealableBestOffer”	The Best Offer that can be dealt against by the client. The amount may be lower than the Regular Amount for that Instrument.
		”DealableBestPlusOffer”	Part of the Depth of Book feature, The Dealable Best Plus Offer will show the next best price in the book.
		”DealableRegularOffer”	An Offer that is with the Regular Amount and can be traded.
		”DealableOutsideOffer”	Part of the Depth of Book feature, the Dealable Outside Offer will show the amount X pips away from the Dealable Best Offer. The number of pips (X pips) depends on the particular currency pair.
		”LocalBid”	The Best Bid from the same Trading Floor. If Intra-Floor Dealing is disabled, these Quotes are not dealable.

Element Name	Attribute	Value	Description
		”LocalOffer”	The Best Offer from the same Trading Floor. If Intra-Floor Dealing is disabled, these Quotes are not dealable.
		”Given”	Given and Paid are the highest paid and lowest given dealt rates during the previous Market View time slice. The deals reflect activity throughout the entire ICAP FX Spot market.
		”Paid”	See above.
	price	price	Note: When a price is available the Data element will include the **price** attribute with the value set to the actual price. When a price is not available, the Data element will include instead the **status** attribute.
	size	amount	Part of the Depth of Book feature, associated with the **DealableBestBid/Offer**, **DealableBestPlusBid/Offer**, and **DealableOusideBid/Offer**, will show the depth of the market at that price up to a certain maximum amount, by Currency Pair. If depth is greater than the maximum amount for that Currency Pair the parameter will also include the plus attribute, as described below. Not included for EbsBestBid/Offer, DealableRegularBid/Offer, andLocalBid/Offer, and Paid/Given.
	status	”NoPrice”	Shown only when there is no price in the market for that InstrumentID.
		“NoData”	Shown only when the price cannot be determined due to an internal error.
	plus	“true”	Part of the Depth of Book feature, If the depth of the market at that price is greater than the threshold amount that can be reported for that Currency Pair, the plus attribute will be included. For example, if the maximum reportable size for a Currency Pair is 50m, and the actual depth size is 35m, then the size attribute will show 35m. If the actual depth size is 75m, then the size attribute will show 50m, and the plus attribute will be included and it will show “true”, which will signify that the depth is greater than 50m. Refer to the Login Response message for the plus amount for each Instrument ID. The plus attribute does not appear if the size attribute is not included.
	timestamp	DateTime	The Trade Date and Time of Paid and Given.

Element Name	Attribute	Value	Description
	traderId	Trader ID	The Trader ID associated with the Quote. Appears only when local Quotes are reported. Will appear only if the local Trader that issued the Quote does not have his HydeMyPrices parameter enabled, otherwise it will show "?".

Subscribe Response Message Example (initial snapshot)

```
<Ati timestamp="2007-07-18T17:09:29.954Z">
    <MarketView instrumentId="8733">
      <Data name="EbsBestBid" price="2.0633"/>
      <Data name="DealableBestBid" status="NoPrice"/>
    <Data name="DealableBestPlusBid" status="NoPrice"/>
      <Data name="DealableRegularBid" status="NoPrice"/>
    <Data name="DealableOutsideBid" status="NoPrice"/>
      <Data name="EbsBestOffer" price="2.0636"/>
      <Data name="DealableBestOffer" status="NoPrice"/>
    <Data name="DealableBestPlusOffer" status="NoPrice"/>
      <Data name="DealableRegularOffer" status="NoPrice"/>
    <Data name="DealableOutsideOffer" status="NoPrice"/>
      <Data name="LocalBid" status="NoPrice"/>
      <Data name="LocalOffer" status="NoPrice"/>
    </MarketView>
</Ati>
```

Market View Update Message Example

```
<Ati timestamp="2007-07-18T17:09:29.954Z">
    <MarketView instrumentId="2143">
      <Data name="EbsBestBid" price="116.34" />
      <Data name="DealableBestBid" size="6000000" price="116.33 />
      <Data name="DealableBestPlusBid" size="3000000" price="116.32 />
      <Data name="DealableRegularBid" price="116.31" />
      <Data name="DealableOutsideBid" size="40000000" plus="true" price="116.30 />
      <Data name="LocalBid" traderId="AF1" price="116.33" />
      <Data name="DealableBestOffer" status="NoPrice" />
      <Data name="EbsBestOffer" price="116.36" />
      <Data name="DealableBestPlusOffer" size="3000000" price="116.32 />
      <Data name="DealableRegularOffer" price="116.33" />
      <Data name="DealableOutsideOffer" size="40000000" plus="true" price="116.40 />
      <Data name="LocalOffer" traderId="JP1" price="116.36" />
    </MarketView>
</Ati>
```

Market View Message Example (Given update)

```
<Ati timestamp="2007-07-18T17:09:29.954Z">
      <MarketView instrumentId="2143">
         <Data name ="Paid" timestamp="2007 03 14T17:16:40.000Z" price="116.34" />
      </MarketView>
</Ati>
```

Market View Message Example (EbsBestOffer update)

```
<Ati timestamp="2007-07-18T17:09:33.626Z">
    <MarketView instrumentId="7442">
        <Data name="EbsBestOffer" price="183.71"/>
    </MarketView>
</Ati>
```

2.7.2 Subscribe Error Message

The only Error message for Subscribe Requests (other than Protocol Violations) is when the Instrument ID is not recognized, because the number is invalid. Only Instrument IDs that were returned in the Login message are valid for subscriptions.

Subscribe Error Messages

Error	Description	
ATI100910 Invalid Currency	The Instrument ID specified does not exist or the client may not be entitled to that Currency Pair.	Verify that the Instrument ID used in this request was included Login response message from the Ai Server. Only Instrument IDs in the Login response are active and can be used. Contact the Trading Floor Administrator to verify the entitlement for that currency pair.
ATI117002 Market View is Disabled	The Ai Server is not configured to send Market Views.	For clients that have multiple Ai servers on the same trading floor, only one is allowed to have Market Views enabled. Contact ICAP Customer Support.
ATI100901 Protocol Violation	The Request message was formatted incorrectly. The session is terminated and the client is disconnected.	Client must correct the format of the XML message and reconnect and re-login.

Note: See the Appendices for a list of all Error Codes

Subscribe Error Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
SubscribeFail	instrumented=”xxxx”	ID	The Instrument ID for the particular Currency Pair.
	msgId=”ATIxxxxx”	errorID	The ID number of the message. Helpful for troubleshooting and logging.
	msg	text	The String containing the actual message along with the incorrect Instrument ID.

Subscribe Error Response Message Example

```
<Ati timestamp=2007-08-24T19:27:49.000Z">
    <SubscribeFail instrumentId="9999" msgId="ATI100910" msg="Invalid currency pair 9999" />
</Ati>
```

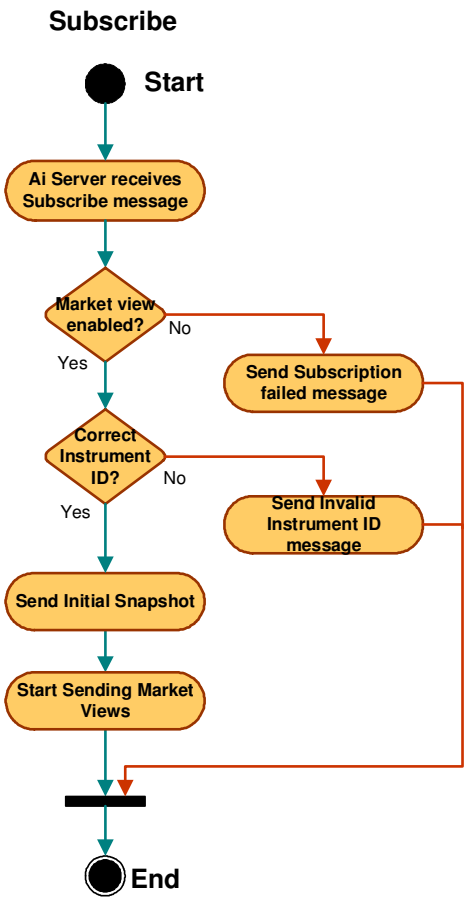

Figure 6 - Subscribe to Market Views

2.8 Unsubscribe

Once logged in and subscribed to Instrument IDs, the Client application will receive Market Views for each of the Instruments subscribed. At times the client may want to stop receiving Market Views for some or all of the Instruments.

2.8.1 Unsubscribe Request

A separate Unsubscribe Request message must be sent for each Instrument.

Unsubscribe Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Unsubscribe	instrumentId mandatory	ID	The Instrument ID that is to be Unsubscribed. Only one Instrument ID is allowed in each Unsubscribe message.

Unsubscribe Request Message Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <Unsubscribe instrumentId="2143" />
</Ati>
```

2.8.2 Unsubscribe Response

There is no Response message to an Unsubscribe Request. The client will simply stop receiving the Market Views for the affected Instrument IDs.

2.8.3 Unsubscribe Error Messages

There are no specific Error messages associated with this message, except the Protocol Violation which is general for all messages. If the Instrument ID is invalid or the Instrument is not subscribed to, the Ai Server ignores the request.

2.9 Order Submit

The Order Submit message allows Client applications to submit Orders to be placed in the market. Orders must have an Order reference ID included and it is recommended that this value be unique. The reference ID does not have to be unique; however, if the reference ID is not unique, when Interrupting Orders, all Orders that have the same reference ID will be Interrupted.

Order Submit parameters

Parameter Name	Default value	Description
TotalActiveOrders	40	The maximum number of Orders client can have in the market at any one time. Additional Orders are rejected. This value is returned by the Logon Response message.
NumberOfOrders	20	The maximum number of Orders client can place in the market during the Throughput Window (see below). Orders that exceed the Throughput are rejected. This value is returned by the Logon Response message.
NumberOfOrdersTimeInterval	5000 (5 sec)	Throughput Window Interval. This value is returned by the Logon Response message.
trading_enabled	true	If Trading is disabled Orders are rejected.
PriceCheck	true	Client configurable. If true and Order does not pass the price check, Order is rejected. Configured by client in the Login message. If enabled, Orders with inverted prices are rejected. If disabled, and the Order deviates form market price by more than the **xPips** amount, the Order is rejected. Refer to the Login Request and Response for more information.
LargeDifferenceCheck	true	Client configurable. If true and Order does not pass the large diff check, Order is rejected. Configured by client in the Login message. If enabled the **largeDifference** parameter value (returned in the Login response) is checked. If the Order price deviates from market price by more than the **largeDifference** amount, the Order is rejected. Refer to the Login Request and Response for more information.
WideSpreadCheck	true	Client configurable. If true and Order does not pass the wide spread check, Order is rejected. Configured by client in the Login message. If enabled the **WideSpread** parameter value (returned in the Login response) is checked. If the Order price deviates from market price by more than the **WideSpread** amount, the Order is rejected. Refer to the Login Request and Response for more information.
minSize	amount	If Order amount is less than the minSize amount for the Instrument, the Order is rejected. Refer to the Login response section for more information.

Parameter Name	Default value	Description
maxSize	amount	If Order amount is greater than the maxSize amount for the Instrument, the Order is rejected. Refer to the Login response section for more information.
sizeIncrement	amount	If the increment of the Order amount is different then the sizeIncrement, the Order is rejected. Refer to the Login response section for more information.
priceIncrement	pips	If the Price increment of the Order is different than the priceIncrement, the Order is rejected. Refer to the Login response section for more information.

2.9.1 Order Submit Request

Each Order must be submitted separately. Inserting more than one OrderSubmit element in a single message is not allowed; it will generate a protocol error, and the Client application will be disconnected.

Order Submit Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
OrderSubmit	instrumentId mandatory	ID	Orders are placed using the Instrument ID for the particular Currency Pair as received in the Login message.
	orderRef mandatory	Reference ID	The client must assign a reference ID to all Orders. Reference IDs should be unique to simplify cancelling Orders and querying Deals, but it is not required. The Reference ID field may be blank or empty (“”). The server rejects all orders with an invalid Order Reference ID attribute or without any Order Reference ID and treats them as protocol violations. Valid characters include the printable ASCII character set (32 ≤ decimal value ≥ 126). The maximum length for an Order Reference ID is 40 characters. The character set exclusions are as follows: ! Exclamation Mark decimal 33 * Asterisk decimal 42 , Comma decimal 44 - Hyphen decimal 45 : Colon decimal 58 =Equals Sign decimal 61 [Left Square Bracket decimal 91] Right Square Bracket decimal 93 _Underscore decimal 95 `Accent decimal 95 ~Tilde decimal 126

Element Name	Attribute	Value	Description
	orderType mandatory	“Bid”, “Offer”, “Buy” or “Sell”	Indicates if the client is placing a “Bid” or an “Offer” Order (Maker Side), or a “Buy” or a “Sell” Order (Taker Side).
	price mandatory	price	The Order price. The price must be formatted as in the Market Views, with the same number of decimal places. Refer to the Guide to ICAP Pair Parameters for more information.
	size mandatory	amount	The amount of the Order. Orders must be greater than or equal to the minSize, less than or equal to the maxSize, and conform to the size increment, or the Order is rejected.

Order Submit Request Message Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <OrderSubmit instrumentId="2143" orderRef="ref123" OrderType="BID" price="121.54"
       size="2000000" />
</Ati>
```

2.9.2 Order Submit Response

If the Order passes all applicable validity tests and is formatted correctly, the Ai Server accepts the Order. The response to the Order Submit message is identical in format to the Order Interrupt Response and the Order Event messages, and is detailed in the Order Event section, along with the Error Messages.

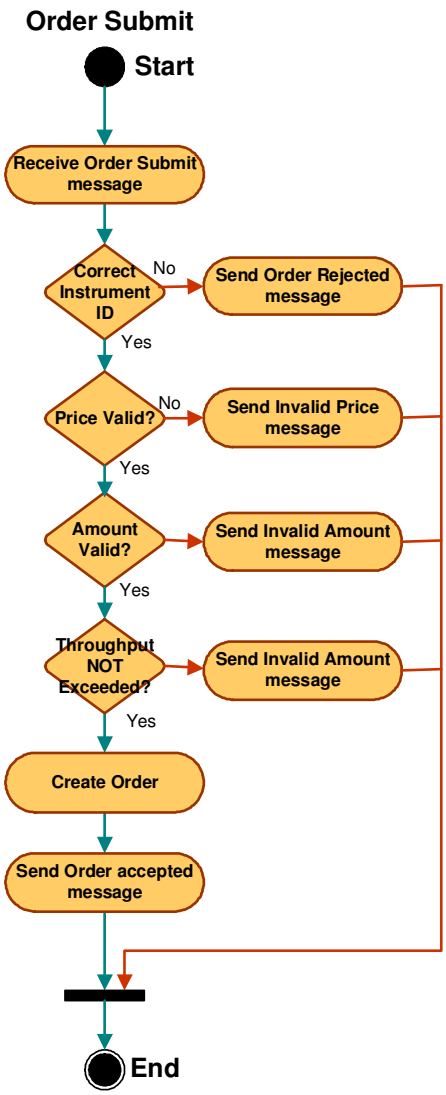

Figure 7 - Submit New Order

2.10 Order Interrupt

To cancel an active Order, the Client application must send an Order Interrupt Request message. Order Interrupts are sent one at a time, for each active Order. Sending multiple Order Interrupts in a single message will result in a protocol violation.

Only Maker Bids and Offers can be Interrupted. Taker Buys or Sells cannot be Interrupted. They are automatically cancelled if the Take misses and cannot be executed.

2.10.1 Order Interrupt Request

Order Interrupt Request messages can specify either the client's Order Reference ID or the Order ID. If using the Order Reference ID and the client has not used unique orderRef IDs when placing Orders, all Orders that have been submitted with that orderRef ID will be cancelled. To Interrupt a specific Order client must then use the Order ID instead of the orderRef ID.

Order Interrupt Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
OrderInterrupt	orderRef optional	Reference ID	The client's Order Reference ID from the Order Submit message. The orderRef ID should be unique. If not unique, all Orders that have that orderRef will be cancelled. If orderRef is used, orderI ID cannot be included in the message. See the Order Interrupt Error Response message.
	orderId optional	Order ID	The Order ID as sent by the Ai Server. If order ID is used, orderRef cannot be included in the message.

Order Interrupt Request Message Example (using orderRef)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <OrderInterrupt orderRef="ref123" />
</Ati>
```

Order Interrupt Request Message Example (using orderId)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <OrderInterrupt orderId="055C-50BF-0000" />
</Ati>
```

2.10.2 Order Interrupt Response

When the Ai Server receives an Order Interrupt message, the Server cancels the Order and responds back to the client with an Order Interrupt Response message. The response to the Order Interrupt message is identical in format to the Order Submit Response and the Order Event messages, and is detailed in the Order Event section.

2.10.3 Order Interrupt Error Messages

The following table shows the error messages that can be received if the Order Interrupt fails.

Order Interrupt Error Messages

Error	Description	Resolution
ATI118001 Order not found.	The Order with the Order ID or Order Ref specified could not be found.	Non-existent Orders cannot be interrupted.
ATI118002 Wrong type for interrupt: %.	Interrupt of a BUY/SELL Order is not allowed.	Only Bid and Offer Orders can be interrupted. Client must change the order type for

		the interrupt.
ATI118003 Order is not active	Interrupt can not be performed for an inactive order.	Only orders in active state can be interrupted. Completed and Cancelled Orders cannot be Interrupted.
ATI100901 Protocol Violation	The Request message was formatted incorrectly. The session in terminated and the client is disconnected. **Note**: All client Orders are Interrupted when the session is terminated.	Client must correct the format of the XML message and reconnect and re-login.

Note: See the Appendices for a list of all Error Codes

Order Interrupt Error Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
InterruptFailed	msgId	ID	The specific Ai Server error ID.
	msg	text	The reason for the Order Submit rejection or Order Interrupt Failure.
	orderRef	Reference ID	The client's Order Reference ID from the Order Submit message.
	orderId	Order ID	The Order ID number specified in the Interrupt Request. Present only if client tried to cancel the Order using the order ID number.

Order Interrupt Error Response Message Example (orderRef)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <InterruptFailed orderRef="ref123" msgId="ATI118001" msg="Order not found." />
</Ati>
```

Order Interrupt Error Response Message Example (order not active)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <InterruptFailed orderRef="ref123" orderId="055C-50BF-0000" msgId="ATI118003"
     msg="Order is not active." />
</Ati>
```

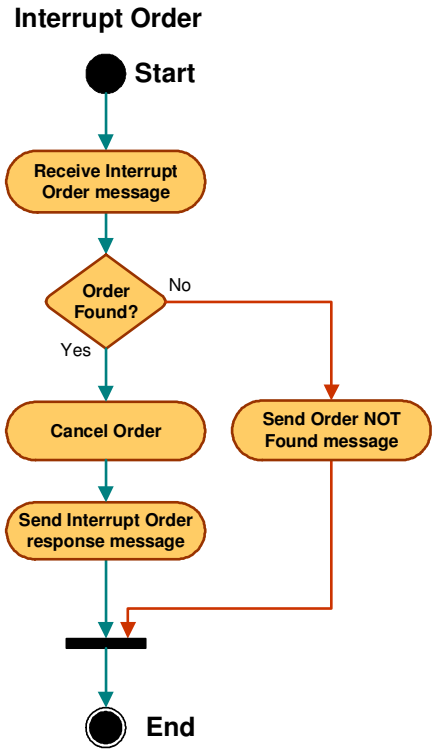

Figure 8 - Order Interrupt Diagram

2.11 Order Event

Order Event is a message sent by the Ai Server in response to: the client sending an Order Submit message, changes in status of a submitted Order, an Interrupt Order request message.

2.11.1 Order Event Message

Order Event Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Order	status	"done" "accepted" or "rejected" "cancelled"	Deal Event messages with status="done" contain the details of the deal or deals that resulted from that Order. See the Deal Section. "accepted" or "rejected" are the results of an Order Submit message. "Cancelled" is the only possible value in the Order Element of the Order Interrupt message. See the Order Interrupt section for more details.
	orderRef	Reference ID	The client's Order Reference ID from the Order Submit message.
	orderId	Order ID	The Order ID as sent by the Ai Server.
	done	amount	The amount of the Order that was executed. This may be all or part of the Order amount. If the amount is zero (0), it means the Order missed (if Taker) and the remaining amount was cancelled. If prior to an Interrupt Request part of the Order was executed, done will show that amount.
	cancelled	amount	Cancelled will show the amount of the Order that was actually cancelled. If this amount is less than the original Order, it means part of the Order was executed. The amount that was cancelled if only part of the Order was executed. If the done amount is zero (0), this will show the original Order amount. The Ai Server will automatically cancel the remaining portion of an Order. If an existing Order for 1m is hit by the client with an Order for 5m, the deal will be for 1m, the remaining 4m will be cancelled.

Element Name	Attribute	Value	Description
	pending	amount	Shown if any part of the Order is in pending state. It is possible that just prior to the client sending the Interrupt Request, another trader may have hit the Quote. In this situation, all or part of the Order may be pending execution. If the pending is for part of the Order, the Ai Server will Interrupt the remaining amount, otherwise the Order cannot be Interrupted. Once the Order is cancelled no other Order Event messages will be received, even if there is a pending amount. If the pending Deal is executed, the client will receive a Deal Event message, either for the amount or for zero.
	msgId	ID	The specific Ai Server message ID.
	msg	text	The actual text message.

Order Event Message Example (done)

```
<Ati timestamp=2007-07-23T18:42:08.697Z">
      <Order status="done" orderRef="ref123" orderId="055C 50BF 0000" done="2000000"
      cancelled="0" />
</Ati>
```

Order Event Message Example (done for zero)

```
<Ati timestamp="2007-07-23T18:42:08.697Z">
    <Order status="done" orderRef="101" orderId="0563-632E-0000" done="0"
    cancelled="5000000"/>
</Ati>
```

Order Submit Response Message Example (accepted)

```
<Ati timestamp="2007-07-23T18:42:08.697Z">
      <Order status="accepted" orderRef="ref123" orderId="055C 50BF 0000 />
</Ati>
```

Order Interrupt Response Message Example

```
<Ati timestamp="2007-07-23T18:42:08.697Z">
      <Order status="cancelled" orderRef="ref123" orderId="055C 50BF 0000" pending="0"
      done="0" cancelled="2000000" msgId="ATI200060" msg="Requested by the client" />
</Ati>
```

2.11.2 Order Submit Error Messages

If the Order fails any of the validation tests, the Order is rejected. Regardless of which test failed, the status of the Order will be “rejected”.

Order Submit Error Messages

Error	Description	Resolution
ATI114001 Order: Throughput violation	The client has exceeded the maximum number of orders allowed. The default limit is 20 orders within five (5) seconds.	The client must reduce the frequency of Order submission. The client can also wait a second or two and retry submitting the rest of the Orders up to the maximum allowed.
ATI114002 Order: Maximum number of active orders exceeded	The maximum number of active Orders allowed has been exceeded. Additional orders are rejected.	Interrupt an active Order or wait for an Order to complete before resubmitting new ones.
ATI115001 Order Id: %. Order rejected due to self-match	The Order submitted by the client matches an active existing Order of the client on the opposite side. The server rejects the second Order.	Verify that the Order information is correct and re-submit.
ATI116001 Reference id is required!	The Order was placed without a Reference ID.	Re-submit the Order with a Reference ID.
ATI117001 Trading is Disabled!	The particular Ai Server is configured for Market Views only and cannot be used for trading.	Log in to an Ai Server enabled for trading, contact ICAP Customer Support if only a single Server exists.
ATI111001 Amount value(%) is invalid!	The amount of the Order in invalid. Amount is less than Minimum, greater than maximum, or is using the wrong increment.	Most currency pairs use an Order amount increment of 1000000. Verify that the amount of the Order adheres.
ATI112002 Price value(%) is invalid!	The supplied price does not conform to the price format rules specified for a currency pair.	Resubmit the Order with the correct price.
ATI113001 Invalid instrument %!	The Instrument ID specified in the Order is not recognized.	Re-submit the Order with the correct Instrument ID.
ATI113003 Invalid Large Difference % for price %!	Bid price is greater than or less than the dealable bid by more than the large difference pips, or Offer price is greater than or less than the dealable Offer by more than the large difference pips.	Resubmit the Order with the correct price or disable the Large Difference Check parameter. (see Login section).
ATI113005 Dynamic Price(%) Check Failed! Dealable = %	Bid is greater than dealable Offer or Offer less than dealable Bid. If price check is not enabled: Bid greater than dealable Offer by more than X-Pips or Offer less than dealable Bid by more than X-Pips	Resubmit the Order with a proper price, i.e., bid prices cannot exceed dealable best Offer and vice-versa, or disable the Price Check parameter (see Login section).

Error	Description	Resolution
ATI113007 Dynamic Price(%) Check: Widespread(%) Check Failed!	Submitted Bid price is less than dealable Offer by more than the Wide Spread pips. Submitted Offer price is greater than the dealable Bid by more than the Wide Spread pips.	Re-submit the Order with the corrected price, or disable the Wide Spread Check parameter (see Login section).
ATI113008 Dynamic Price(%) Check: Localprice(%) Check Failed!	The submitted Order price matches a local price and intra-floor dealing (IFD) is disabled.	Resubmit the Order with a corrected price.
ATI113009 Order price(%) exceeds xPips(%) of Dealable(%)!	Only when the Price Check parameter is disabled – The submitted price varies by more than X-pips from the current best dealable Bid/Offer.	Resubmit the Order with a price closer to the current market.
ATI100901 Protocol Violation	The Request message was formatted incorrectly. The session in terminated and the client is disconnected.	Client must correct the format of the XML message and reconnect and re-login.

Note: See the Appendices for a list of all Error Codes

Order Submit Error Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Order	status	“rejected”	“Rejected” is the only value if an Order is rejected.
	msgId	ID	The specific Ai Server error ID.
	msg	text	The reason for the Order Submit rejection or Order Interrupt Failure.
	orderRef	Reference ID	The client’s Order Reference ID from the Order Submit message.

Order Submit Error Response Message Example (Throughput Violation)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <Order status="rejected" orderRef="ref123" msgId="ATI114001" msg="Order: Throughput
       violation" />
</Ati>
```

2.12 Deal Event

Deal Event messages are not initiated by a Client Request message, but are generated by deals that resulted from the Client's Orders.

2.12.1 Deal Event Message

Different types of Deal Event messages can be received depending of the status of the Deal. The number of attributes present in the Deal Event message will depend on the status of message, but the format will always be as shown in the table.

Deal Event Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Deal	dealId	Deal ID	The Deal ID as sent by the Ai Server, for the particular returned deal info.
	size	amount	The amount of the Deal.
	status	"pending", "confirmed" "done", "unverified", verified, or "unknown"	Pending means that the Quote was hit, confirmed means it was confirmed by the other side. Done means that settlement instructions have been exchanged and the deal is done. Unverified means the deal cannot be verified with the counterparty, the Spot System will try to recover the deal. Unknown means there was an error, and client should contact Customer Support,
	role	"maker" or "taker"	Specifies if the role was "maker" or "taker."
	orderRef	Reference ID	The client's Order Reference ID from the Order Submit message.
	orderId	Order ID	The Order ID as sent by the Ai Server.
	price	price	The price at which the Deal was transacted.
	ticket	ticket ID	The Ticket ID number as assigned by EBS.
	dealTime	DateTime	The Date and Time the deal was transacted.
	valueDate	Date	For SPOT deals, the day the transaction will settle. For NDFs, the day the near value is established. In both cases, it is typically two business days.
	fixingDate	Date	Used in NDFs only, the day the forward value is established, usually two days before the Settlement date.
	settlementDate	Date	Used for NDFs only, the day the transaction settles.
	dealCode	Floor Code	The Floor Code (Bank Code) of the counterparty.
	traderId	Trader ID	The counterparty Trader ID.
	prime	"true" or "false"	Indicates if the counterparty was a prime customer. This element will not be present if "send_prime_deal_indicator" is set to false.

The elements present in the Deal Event message depends on the particular **status** of the Deal Event. The following table details which elements are present, and when.

Status Element	Pending	Confirmed	Done	Unverified	Verified	Unknown
Status	Y	Y	Y	Y	Y	Y
Deal ID	Y	Y	Y	Y	Y	Y
Size	Y	Y	Y	Y	Y	Y
Price	Y	NZ	NZ	NZ	NZ	NZ
Order Ref	Y		NZ		NZ	
Order ID	Y		NZ		NZ	
Role	Y		NZ		NZ	
Ticket		NZ, AV	NZ		NZ	
Deal Code		NZ	NZ	NZ	NZ	NZ
Trader ID			NZ, AV		NZ, AV	
Prime			NZ		NZ	
Deal Time		NZ, AV	NZ		NZ	
Value Date			NZ		NZ	
Fixing Date			NZ, NDF		NZ, NDF	
Settlement Date			NZ, NDF		NZ, NDF	

NZ – Included when the value is not zero. Omitted if deal is not done (size=0)
AV – Included if value is available (not always present)
NDF - Included for NDFs only

Deal Event Message Example (pending)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Deal status="pending" dealId="055C 5196 001E-01" price="116.44" size="2000000"
    role="taker" orderRef="ref123" orderId="055C 50BF 0000" />
</Ati>
```

Deal Event Message Example (confirmed)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Deal status="confirmed" dealId="055C 5196 001E-01" size="1000000" ticket="3334"
    dealTime="2007¬ 03 14T14:32:46Z" dealCode="BRA3" />
</Ati>
```

Deal Event Message Example (done SPOT)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Deal status="done" dealId="055C-5196-001E-01" price="116.44" size="1000000"
      role="taker" ticket="3334" dealTime="2007 03 14T14:32:46Z" valueDate="2007 03 19"
      orderRef="ref123" orderId="055C-50BF-0000" dealCode="BRA3" traderId="AF3"
      prime="false" />
</Ati>
```

Deal Event Message Example (done NDF)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
   <Deal status="done" dealId="055C-5196-002B-01" price="116.44" size="1000000" role="taker"
      ticket="3337" dealTime="2007 03 14T14:32:46Z" valueDate="2007 03 19"
      orderRef="ref123" orderId="055C-50BF-0000" dealCode="BRA3" traderId="AF3"
      prime="false" settlementDate="2007 06 07" fixingDate="2008 03 05" />
</Ati>
```

Deal Event Message Example (unverified)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Deal status="unverified" dealId="055C-5196-001E-01" price="116.44" size="2000000"
    role="taker" dealCode="BRA3" />
</Ati>
```

Deal Event Message Example (verified)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <Deal status="verified" dealId="055C-5196-001E-01" price="116.44" size="1000000"
        role="taker" ticket="3334" dealTime="2007-08-24T18:57:29Z" valueDate="2007-03-19"
        dealCode="BRA3" traderId="AF3" prime="false" />
</Ati>
```

Deal Event Message Example (unknown)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
      <Deal status="unknown" dealId="055C-5196-001E-01" price="116.44" size="2000000"
       role="taker" dealCode="BRA3" />
</Ati>
```

2.13 Interrupt All

To cancel all active Orders, the Client application must send an Interrupt All message request to the Ai Server. The Ai Server will try to cancel all Orders and respond with an Interrupt Response message for each cancelled Order.

The Interrupt All function is also initiated by the Ai Server if the Client application suddenly logs out, disconnects, losses connection or fails to respond to Heartbeats.

2.13.1 Interrupt All Request

Interrupt All Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time mandatory	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
InterruptAll	none	none mandatory	The presence of this element tells the Ai Server that all Orders are to be cancelled.

Interrupt All Request Message Example

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <InterruptAll />
</Ati>
```

One Order Interrupt Response (Order Cancelled) message must be sent for each active Order. The format is identical to the Order Interrupt Response message. Refer to the Order Event section for Response message format details and examples.

Note: If the client lost connectivity or was disconnected, no messages will have been received.

2.13.2 Interrupt All Error Messages

If the client does not have active Orders in the market, the Interrupt All message does not return a response. No other Error messages are associated with the Interrupt All message.

2.14 Deal Query

The Deal Query message can be used to get information about executed deals. Queries can be performed based on Order Reference IDs, Order IDs, or Deal IDs. Queries using a Deal ID will only return that particular deal. Queries using the Order Reference ID or the Order ID will return all deals that resulted from that particular Order.

Deal Query parameters

Parameter Name	Default value	Description
max_query_time	2	Queries can be issued only for deals that were done in the last 2 hours. This is to minimize the impact the query may have on Trading.

2.14.1 Deal Query Request

Deal Query Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
DealQuery	queryRef mandatory	Reference ID	The client must assign a reference ID to all Queries. Reference IDs should be unique, but it is not required. The Reference ID field may be blank or empty (“”). The server rejects all Queries with an invalid Reference ID attribute or without any Reference ID and treats them as protocol violations. Valid characters include the printable ASCII character set (32 ≤ decimal value ≥ 126). The maximum length for a Query Reference ID is 40 characters. The character set exclusions are as follows: ! Exclamation Mark decimal 33 * Asterisk decimal 42 , Comma decimal 44 - Hyphen decimal 45 : Colon decimal 58 =Equals Sign decimal 61 [Left Square Bracket decimal 91] Right Square Bracket decimal 93 _Underscore decimal 95 ` Accent decimal 95 ~Tilde decimal 126
	orderRef optional	Reference ID	The client’s Order Reference ID from the Order Submit message. Used only if client submits query using the orderRef ID.
	orderId optional	Order ID	The Order ID as sent by the Ai Server. . Used only if client submits query using the orderId.
	dealId optional	Deal ID	The Deal ID as sent by the Ai Server. Used only if client submits query using the dealId.

Deal Query Request Message Example (Order Reference)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <DealQuery queryRef="query1" orderRef="ref123" />
</Ati>
```

Deal Query Request Message Example (Order ID)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <DealQuery queryRef="query2" orderId="055C-5196-001E" />
</Ati>
```

Deal Query Request Message Example (Deal ID)

```
<Ati timestamp="2007-08-24T19:27:49.000Z">
    <DealQuery queryRef="query3" dealId="055C-5196-001E-01" />
</Ati>
```

2.14.2 Deal Query Response

The response that the client receives depends on the type of query that was submitted, and can include a single deal or multiple deals.

Deal Query Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
DealQueryResult			The main element that contains the returned deal information. When there are no deals to return, DealQueryResult will contain no deal elements.
	queryRef	Reference ID	The Reference ID assigned to the query by the client.
Deal			The Deal element is repeated for every Deal included in the Query Response.
	dealId	Deal ID	The Deal ID as sent by the Ai Server, for the particular returned deal info.
	dealType	“sold” or “bought”	Specifies if the deal amount was “sold” or “bought.”
	role	“maker” or “taker”	Specifies if the role was “maker” or “taker.”
	status	“done” or “pending”	Primarily this attribute should say “done.” There are chances, depending on timing, that some deals returned may say “pending.” A subsequent query should show that deal as “done.”

Element Name	Attribute	Value	Description
	price	amount	The price at which the deal was executed.
	size	amount	The amount of the Deal.
	instrumentId	ID	The Instrument ID of the particular Currency Pair.
	ticket	ID	The Ticket ID of the deal as assigned by EBS.
	valueDate	Date	For SPOT deals, the day the transaction will settle. For NDFs, the day the near value is established. In both cases it is typically two business days.
	fixingDate	Date	Used in NDFs only, the day the forward value is established, usually two days before the Settlement date.
	settlementDate	Date	Used for NDFs only, the day the transaction settles.
	dealTime	date and time	The date and time when the deal was transacted.
	orderRef	Reference ID	The original client assigned Order Reference ID.
	orderId	Order ID	The original Order ID assigned by EBS.
	dealCode	Floor Code	The Floor Code (Bank Code) of the counterparty.
	traderId	Trader ID	The counterparty Trader ID.

Deal Query Response Message Example (orderRef or OrderId)

```
<Ati timestamp="2007-08-24T20:17:48.062Z">
  <DealQueryResult queryRef="Q-123">
    <Deal dealId="055C-FB86-0001-01" dealType="bought" role="taker" status="done"
        price="118.25" size="1000000" instrumentId="7442" ticket="3472" valueDate="2007-08-28"
        dealTime="2007-08-24T20:32:03Z" orderRef="ref01ac" orderId="055C-FB86-0001"
        dealCode="BRA3" traderId="AF3" prime="false"/>
  </DealQueryResult>
</Ati>
```

Deal Query Response Message Example (NDF dealId)

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
  <DealQueryResult queryRef="Q-123">
    <Deal dealId="055C-FB86-0000-01" dealType="bought" role="taker" status="done"
        price="1335.0" size="1000000" instrumentId="7471" ticket="3470" valueDate="2007-08-28"
        fixingDate="2007-09-21" settlementDate="2007-09-28" dealTime="2007-08-24T20:31:46Z"
        orderRef="ref01ab" orderId="055C-FB86-0000" dealCode="BRA3" traderId="AF3"
        prime="false"/>
  </DealQueryResult>
</Ati>
```

2.14.3 Deal Query Error Messages

Error	Description	
ATI119101 Exception during deal query [%].	Queries cannot be performed using non-unique order reference ID.	Re-submit the Query using the Order ID or the Deal ID.
ATI119111 Query combination is not supported.	Queries should include only one search parameter: Order Reference, Order ID, or Deal ID.	Restructure the query and re-submit.
ATI119114 ID value invalid for query.	The Query Reference ID is invalid.	Re-submit using a valid Query Reference ID.
ATI119115 Requested data exceeds % hour limit	Queries that relate to Deals older than the allowed time limit are rejected. The default is 2 hours.	Submit Queries that relate only to Deals executed within the allowed time limit
ATI119117 Query for non-unique reference is not supported.	Queries using a non-unique Order Reference are rejected.	Re-submit the Query using the Order ID.
ATI119118 Query aborted [%].	The Query was aborted.	
ATI119119 User must be signed on for query.	Queries cannot be performed if the client is not Logged In.	Client must Login.
ATI100901 Protocol Violation	The Request message was formatted incorrectly. The session in terminated and the client is disconnected.	Client must correct the format of the XML message, reconnect and re-login.

Note: See the Appendices for a list of all Error Codes

Deal Query Error Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
DealQueryFailed	queryRef	Reference ID	The reference ID assigned to the query by the client.
	msgId	ID	The specific Query Error message ID.
	msg	text	The reason for the query failure.

Deal Query Error Response Message Example (non-unique Ref number)

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
      <DealQueryFailed queryRef="query1" msgId="ATI119117" msg="Query for non unique
       reference is not supported." />
</Ati>
```

2.15 Session Event

Session Event messages are sent by the Ai Server when there are changes in the status of the session.

2.15.1 Session Event Message

Session Event messages may or may not require action on the client's part. It is up to the client application to determine what action is required for the specific event.

Note: Refer to the Appendices for a complete list of all message codes. Session related messages are grouped under the **Session** heading.

Session Event Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
SessionEvent	type	”EBSdown”, “EBSup”, “ValueDate”, “TradeDate”, “CreditLow”, CreditExh”, CreditAvl”, “unknown”	Refer to the Error and Event messages table in the Appendices for a complete list, under the Session heading.
	msgId	ID	The ID of the particular Event message.
	msg	text message	A plain text explanation of the event.
	instrumentId	ID	The Instrument ID corresponding to the particular Currency Pair being affected by the event.
	tradeDate		If the event was a Trade Date Change, this would include the new Trade Date. Happens for most currencies at 5:00PM New York time.
	valueDate		If the event was a Value Date Change, this would include the new value Date.

Session Event Message Example (Value Date Change)

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
      <SessionEvent type="ValueDate" msgId="ATI300109" msg="Value Date Changed"
   instrumentId ="2135" valueDate="2007-03-07" />
</Ati>
```

Session Event Message Example (Credit Low)

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
      <SessionEvent type="CreditLow" msgId="ATI300105" msg="Credit is low" />
</Ati>
```

Session Event Message Example (EBS Down)

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
      <SessionEvent type="EBSDown" msgId="ATI300102" msg="Trading system is down" />
</Ati>
```

2.16 Logoff

To close the Trading Session and log off from the System, the Client application must send a Logoff Request message.

2.16.1 Logoff Request

The Logoff Request message can be sent any time during the session. The Ai Server will cancel any Orders that the client may have in the market, after which it will terminate the session.

Logoff Request Message Format

Element Name	Attribute	Value	Description
Ati	timestamp mandatory	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Logoff	none mandatory	none	The presence of the Logoff element is required to instruct the Ai Server to close the session.

Logoff Request Message Example

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
    <Logoff />
</Ati>
```

2.16.2 Logoff Response

The Ai Server will respond with a Logoff Response message. If there were active Orders in the market, the AI Server will Interrupt All Orders, send the Interrupt Response message for each Order (as detailed in the Interrupt section), and will terminate the session, and disconnect.

Logoff Response Message Format

Element Name	Attribute	Value	Description
Ati	timestamp	date and time	The current Date and Time in the format: yyyy-mm-ddThh:mm:ss.zzzZ
Logoff	msgId	ID	The ID number of the message.
	msg	text	The text message. For Logoff, it is always “Client Initiated Signoff”.

Logoff Response Message Example

```
<Ati timestamp="2007-08-24T20:17:40.234Z">
      <Logoff msgId="ATI200022" msg="Protocol violation: out of order event" />
</Ati>
```

Note: No specific Error messages are associated with the Logoff message, other than the Protocol Violation message.

Appendix 1 Error and Session Messages

The following table includes all Ai generated Error and Session messages and their IDs. Some Error messages are simply wrappers for Spot system messages, which are too numerous to list, but they will always be delivered using one of the ATI ID numbers.

Message ID	Description	Resolution
Main Processing		
ATI100901	Client protocol error	Client must correct the format of the XML message and reconnect and re-login.
Order Checks		
ATI111001	Amount value(%) is invalid!	Most currency pairs use an Order amount increment of 1000000. Verify that the amount of the Order adheres.
ATI112002	Price value(%) is invalid!	Resubmit the order with the correct price.
ATI113001	Invalid instrument %!	Re-submit the Order with the correct Instrument ID.
ATI113003	Invalid Large Difference % for price %!	Resubmit the order with the correct price or disable the Large Difference Check parameter. (see Login section).
ATI113005	Dynamic Price(%) Check Failed! Dealable = %	Resubmit the order with a proper price, i.e., bid prices cannot exceed dealable best Offer and vice-versa, or disable the Price Check parameter (see Login section).
ATI113007	Dynamic Price(%) Check: Widespread(%) Check Failed!	Re-submit the Order with the corrected price, or disable the Wide Spread Check parameter (see Login section).
ATI113008	Dynamic Price(%) Check: Localprice(%) Check Failed!	Resubmit the order with a corrected price.
ATI113009	Order price(%) exceeds xPips(%) of Dealable(%)!	Resubmit the order with a price closer to the current market.
ATI114001	Order: Throughput violation	The client must reduce the frequency of Order submission. The client can also wait a second or two and retry submitting the rest of the Orders up to the maximum allowed.
ATI114002	Order: Maximum number of active orders exceeded	Interrupt an active order or wait for an Order to complete before resubmitting new ones.
ATI115001	Order Id: %. Order rejected due to self-match	Verify that the Order information is correct and re-submit.
ATI117001	Trading is Disabled!	Log in to an Ai Server enabled for trading, contact ICAP Customer Support if only a single Server exists.
ATI300008	Order cancelled [%]	
Subscription		

Message ID	Description	Resolution
ATI100910	Invalid currency pair %	Verify that the Instrument ID used in this request was included Login response message from the Ai Server. Only Instrument IDs in the Login response are active and can be used. Contact the Trading Floor Administrator to verify the entitlement for that currency pair.
ATI117002	Market View is Disabled	For clients that have multiple Ai servers on the same trading floor, only one is allowed to have Market Views enabled. Contact ICAP Customer Support.
Interrupt Failure		
ATI118001	Order not found.	Non-existent Orders cannot be interrupted.
ATI118002	Wrong type for interrupt: %.	Only Bid and Offer Orders can be interrupted. Client must change the order type for the interrupt.
ATI118003	Order is not active	Only orders in active state can be interrupted. Completed and Cancelled Orders cannot be Interrupted.
Deal Query		
ATI119101	Exception during deal query [%].	Re-submit the Query using the Order ID or the Deal ID.
ATI119111	Query combination is not supported.	Restructure the query and re-submit.
ATI119114	ID value invalid for query.	Re-submit using a valid Query Reference ID.
ATI119115	Requested data exceeds % hour limit.	Submit Queries that relate only to Deals executed within the allowed time limit
ATI119117	Query for non-unique reference is not supported.	Re-submit the Query using the Order ID.
ATI119118	Query aborted [%].	
ATI119119	User must be signed on for query.	Client must Login.
Communication		
ATI200015	Protocol error, event in wrong state: %.	
ATI200020	Socket closed [%].	
ATI200021	Exception on socket [%].	
ATI200050	Stale connection detected on socket [%].	
ATI200051	Dead connection detected on socket	

Message ID	Description	Resolution
	[%].	
ATI200061	Caused by Stale Socket Connection	
ATI200071	Socket connection closed by the client.	
ATI200072	Socket connection closed by the server.	
ATI200073	Socket connection accepted from %.	
Signon		
ATI200016	Protocol error, wrong version: %.	Verify that The correct Ai XML version is used in the Login message. For Ai 4.0 it should be aiProtocolVersion="3.0"
ATI200028	Client Initiated Signoff	Normal response to client Logoff request.
ATI300009	%	Verify that the password used for this account is the correct one. Retry the Login. The Trader ID used in the Login does not exist on the system. Verify the Trader ID and retry the Login.
ATI300201	User already signed on, requesting other session cancellation!	The client must send a Cancel Other Session message to close the existing session and open a new one. Although the Login fails, it remains in pending state and will continue after the client sends a Cancel Other Session message, as described in the Cancel Other Session section. An alternative solution is to disconnect and Login again with the AutoCancelDuplSession parameter set to true.
ATI300202	Default password used to Login, password change required!	Send a Password Change message with a new Password. The Login process remains in pending state and will continue after a successful password change. Refer to the Change Password section for more information.
ATI300203	Password expired, password change required!	Send a Password Change message with a new Password.
ATI300211	%! Invalid New Password. The Old Password included in the message was not the correct password for the account.	The password did not pass minimum security requirements as to length and composition. Verify the password and resend the Password Change request.
Session		
ATI300101	Trading system is up	Client can resume trading.

Message ID	Description	Resolution
ATI300102	Trading system is down	Client must wait for the “Trading System is up” message.
ATI300103	Credit is now available	Client can resume trading.
ATI300104	Credit has been exhausted	Client must inform TFA of condition. Credit is replenished next trading day, or adjusted by TFA.
ATI300105	Credit is low	Client must inform TFA of condition. Credit is replenished next trading day, or adjusted by TFA.
ATI300106	Unknown	Contact Customer Support.
ATI300108	Trade Date Changed A new trading day has begun.	No Action
ATI300109	Value Date Changed Notification that the Value Date for a particular currency pair has changed.	No Action

Appendix 2 XML 3.0 Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
    <xs:element name="Ati">
        <xs:annotation>
            <xs:documentation>Comment describing your root element</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:choice>
                <xs:element name="Login">
                    <xs:complexType>
                        <xs:choice minOccurs="0" maxOccurs="unbounded">
                            <xs:element name="Param" maxOccurs="unbounded">
                                <xs:complexType>
                                    <xs:attribute name="name" use="required"/>
                                    <xs:attribute name="value" use="required"/>
                                </xs:complexType>
                            </xs:element>
                        </xs:choice>
                        <xs:attribute name="aiProtocolVersion" use="required"/>
                        <xs:attribute name="password" use="required"/>
                        <xs:attribute name="username" use="required"/>
                    </xs:complexType>
                </xs:element>
                <xs:element name="LoginFailed">
                    <xs:complexType>
                        <xs:attribute name="msg" use="required"/>
                        <xs:attribute name="msgId" use="required"/>
                        <xs:attribute name="action" use="required"/>
                    </xs:complexType>
                </xs:element>
                <xs:element name="LoginOk">
                    <xs:complexType>
                        <xs:sequence maxOccurs="unbounded">
                            <xs:element name="Param" type="ParamType30" minOccurs="0" maxOccurs="unbounded"/>
                            <xs:element name="Instrument" maxOccurs="unbounded">
                                <xs:complexType>
                                    <xs:attribute name="instrumentId" type="xs:int" use="required"/>
                                    <xs:attribute name="base" type="xs:string" use="required"/>
                                    <xs:attribute name="local" type="xs:string" use="required"/>
                                    <xs:attribute name="regSize" type="xs:integer" use="required"/>
                                    <xs:attribute name="xPips" type="xs:decimal" use="required"/>
                                    <xs:attribute name="wideSpread" type="xs:decimal" use="required"/>
                                    <xs:attribute name="priceIncrement" type="xs:integer" use="required"/>
                                    <xs:attribute name="largeDiff" type="xs:decimal" use="required"/>
                                    <xs:attribute name="minSize" type="xs:integer" use="required"/>
                                    <xs:attribute name="maxSize" type="xs:integer" use="required"/>
                                    <xs:attribute name="sizeIncrement" type="xs:integer" use="required"/>
                                    <xs:attribute name="valueDate" type="xs:date" use="required"/>
                                    <xs:attribute name="tradeDate" type="xs:date" use="required"/>
                                    <xs:attribute name="type" use="optional">
                                        <xs:simpleType>
                                            <xs:restriction base="xs:string">
                                                <xs:enumeration value="Spot"/>
                                                <xs:enumeration value="NDF"/>
                                            </xs:restriction>
                                        </xs:simpleType>
                                    </xs:attribute>
                                    <xs:attribute name="tenor" type="xs:string" use="optional"/>
```

```
                    <xs:attribute name="settlementDate" type="xs:date" use="optional"/>
                    <xs:attribute name="fixingDate" type="xs:date" use="optional"/>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
        <xs:attribute name="sessionId" type="xs:hexBinary" use="required"/>
    </xs:complexType>
</xs:element>
<xs:element name="PasswordExpired"/>
<xs:element name="ChangePassword">
    <xs:complexType>
        <xs:attribute name="newPassword" use="required"/>
        <xs:attribute name="oldPassword" use="required"/>
    </xs:complexType>
</xs:element>
<xs:element name="ChangePasswordOk"/>
<xs:element name="ChangePasswordFail">
    <xs:complexType>
        <xs:attribute name="msg" use="required"/>
        <xs:attribute name="msgId" use="required"/>
    </xs:complexType>
</xs:element>
<xs:element name="AlreadyLoggedIn"/>
<xs:element name="CancelOtherSession"/>
<xs:element name="Heartbeat">
    <xs:complexType>
        <xs:attribute name="sequence" type="xs:integer" use="required"/>
    </xs:complexType>
</xs:element>
<xs:element name="HeartbeatRequest">
    <xs:complexType>
        <xs:attribute name="sequence" type="xs:integer" use="required"/>
    </xs:complexType>
</xs:element>
<xs:element name="SessionEvent">
    <xs:complexType>
        <xs:attribute name="msg" use="required"/>
        <xs:attribute name="type" use="required">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="CreditLow"/>
                    <xs:enumeration value="CreditOut"/>
                    <xs:enumeration value="CreditAvailable"/>
                    <xs:enumeration value="EBSUp"/>
                    <xs:enumeration value="EBSDown"/>
                    <xs:enumeration value="EBSNotification"/>
                    <xs:enumeration value="TradeDate"/>
                    <xs:enumeration value="ValueDate"/>
                    <xs:enumeration value="Unrecognized"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="msgId" use="required"/>
        <xs:attribute name="tradeDate" type="xs:date" use="optional"/>
        <xs:attribute name="instrumentId" use="optional"/>
        <xs:attribute name="valueDate" type="xs:date" use="optional"/>
    </xs:complexType>
</xs:element>
<xs:element name="Logoff">
    <xs:complexType>
        <xs:attribute name="msg" use="optional"/>
        <xs:attribute name="msgId" use="optional"/>
    </xs:complexType>
```

```
</xs:element>
<xs:element name="Subscribe">
   <xs:complexType>
      <xs:attribute name="instrumentId" use="required"/>
   </xs:complexType>
</xs:element>
<xs:element name="Unsubscribe">
   <xs:complexType>
      <xs:attribute name="instrumentId" use="required"/>
   </xs:complexType>
</xs:element>
<xs:element name="SubscribeFail">
   <xs:complexType>
      <xs:attribute name="msgId" use="optional"/>
      <xs:attribute name="msg" use="optional"/>
      <xs:attribute name="instrumentId" use="required"/>
   </xs:complexType>
</xs:element>
<xs:element name="MarketView">
   <xs:complexType>
      <xs:choice maxOccurs="unbounded">
         <xs:element name="Data" maxOccurs="unbounded">
            <xs:complexType>
               <xs:attribute name="name" use="required">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:enumeration value="EbsBestBid"/>
                        <xs:enumeration value="EbsBestOffer"/>
                        <xs:enumeration value="DealableBestBid"/>
                        <xs:enumeration value="DealableBestOffer"/>
                        <xs:enumeration value="DealableRegularBid"/>
                        <xs:enumeration value="DealableRegularOffer"/>
                        <xs:enumeration value="DealableOutsideBid"/>
                        <xs:enumeration value="DealableOutsideOffer"/>
                        <xs:enumeration value="DealableBestPlusBid"/>
                        <xs:enumeration value="DealableBestPlusOffer"/>
                        <xs:enumeration value="LocalBid"/>
                        <xs:enumeration value="LocalOffer"/>
                        <xs:enumeration value="Paid"/>
                        <xs:enumeration value="Given"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:attribute>
               <xs:attribute name="price" type="xs:decimal" use="optional"/>
               <xs:attribute name="size" type="xs:integer" use="optional"/>
               <xs:attribute name="timestamp" type="xs:dateTime" use="optional"/>
               <xs:attribute name="status" use="optional"/>
               <xs:attribute name="traderId" use="optional"/>
               <xs:attribute name="plus" type="xs:boolean" use="optional"/>
            </xs:complexType>
         </xs:element>
      </xs:choice>
      <xs:attribute name="instrumentId" use="required"/>
   </xs:complexType>
</xs:element>
<xs:element name="OrderInterrupt">
   <xs:complexType>
      <xs:attribute name="orderId" use="optional"/>
      <xs:attribute name="orderRef" use="optional"/>
   </xs:complexType>
</xs:element>
<xs:element name="OrderSubmit">
   <xs:complexType>
```

```
            <xs:attribute name="instrumentId" use="required"/>
            <xs:attribute name="size" type="xs:integer" use="required"/>
            <xs:attribute name="price" type="xs:decimal" use="required"/>
            <xs:attribute name="orderType" use="required">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:enumeration value="Bid"/>
                     <xs:enumeration value="Offer"/>
                     <xs:enumeration value="Buy"/>
                     <xs:enumeration value="Sell"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:attribute>
            <xs:attribute name="orderRef" use="required"/>
         </xs:complexType>
      </xs:element>
      <xs:element name="InterruptAll"/>
      <xs:element name="InterruptFailed">
         <xs:complexType>
            <xs:attribute name="msg" use="required"/>
            <xs:attribute name="msgId" use="required"/>
            <xs:attribute name="orderId" use="optional"/>
            <xs:attribute name="orderRef" use="optional"/>
         </xs:complexType>
      </xs:element>
      <xs:element name="Order">
         <xs:complexType>
            <xs:attribute name="status" use="required">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:enumeration value="accepted"/>
                     <xs:enumeration value="rejected"/>
                     <xs:enumeration value="cancelled"/>
                     <xs:enumeration value="done"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:attribute>
            <xs:attribute name="orderRef" use="required"/>
            <xs:attribute name="orderId" use="optional"/>
            <xs:attribute name="pending" type="xs:integer" use="optional"/>
            <xs:attribute name="cancelled" type="xs:integer" use="optional"/>
            <xs:attribute name="done" type="xs:integer" use="optional"/>
            <xs:attribute name="msg" use="optional"/>
            <xs:attribute name="msgId" use="optional"/>
         </xs:complexType>
      </xs:element>
      <xs:element name="OrderPosition">
         <xs:complexType>
            <xs:attribute name="orderId" use="required"/>
            <xs:attribute name="orderRef" use="required"/>
            <xs:attribute name="redQuote" use="required"/>
         </xs:complexType>
      </xs:element>
      <xs:element name="Deal">
         <xs:complexType>
            <xs:attribute name="dealId" use="required"/>
            <xs:attribute name="status" use="required">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:enumeration value="pending"/>
                     <xs:enumeration value="confirmed"/>
                     <xs:enumeration value="done"/>
                     <xs:enumeration value="unverified"/>
```

```
                        <xs:enumeration value="verified"/>
                        <xs:enumeration value="unknown"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:attribute>
            <xs:attribute name="price" type="xs:decimal" use="optional"/>
            <xs:attribute name="size" use="required"/>
            <xs:attribute name="prime" use="optional"/>
            <xs:attribute name="traderId" use="optional"/>
            <xs:attribute name="valueDate" type="xs:date" use="optional"/>
            <xs:attribute name="dealCode" use="optional"/>
            <xs:attribute name="dealTime" type="xs:dateTime" use="optional"/>
            <xs:attribute name="ticket" use="optional"/>
            <xs:attribute name="orderId" use="optional"/>
            <xs:attribute name="orderRef" use="optional"/>
            <xs:attribute name="role" use="optional"/>
            <xs:attribute name="SettlementDate" type="xs:date" use="optional"/>
            <xs:attribute name="fixingDate" type="xs:date" use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="DealQuery">
        <xs:complexType>
            <xs:attribute name="queryRef" use="required"/>
            <xs:attribute name="orderId" use="optional"/>
            <xs:attribute name="orderRef" use="optional"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="DealQueryResult">
        <xs:complexType>
            <xs:choice maxOccurs="unbounded">
                <xs:element name="Deal" maxOccurs="unbounded">
                    <xs:complexType>
                        <xs:attribute name="role" use="required"/>
                        <xs:attribute name="instrumentId" use="required"/>
                        <xs:attribute name="size" type="xs:integer" use="required"/>
                        <xs:attribute name="status" use="required"/>
                        <xs:attribute name="dealId" use="required"/>
                        <xs:attribute name="price" type="xs:decimal" use="required"/>
                        <xs:attribute name="dealType" use="required">
                            <xs:simpleType>
                                <xs:restriction base="xs:string">
                                    <xs:enumeration value="bought"/>
                                    <xs:enumeration value="sold"/>
                                </xs:restriction>
                            </xs:simpleType>
                        </xs:attribute>
                        <xs:attribute name="orderRef" use="optional"/>
                        <xs:attribute name="orderId" use="optional"/>
                        <xs:attribute name="dealCode" use="optional"/>
                        <xs:attribute name="traderId" use="optional"/>
                        <xs:attribute name="valueDate" type="xs:date" use="optional"/>
                        <xs:attribute name="dealTime" type="xs:dateTime" use="optional"/>
                        <xs:attribute name="ticket" use="optional"/>
                        <xs:attribute name="SettlementDate" type="xs:date" use="optional"/>
                        <xs:attribute name="prime" use="optional"/>
                        <xs:attribute name="fixingDate" type="xs:date" use="optional"/>
                    </xs:complexType>
                </xs:element>
            </xs:choice>
            <xs:attribute name="queryRef"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="DealQueryFailed">
```

```
                <xs:complexType>
                    <xs:attribute name="msg" use="required"/>
                    <xs:attribute name="msgId" use="required"/>
                    <xs:attribute name="queryRef" use="required"/>
                </xs:complexType>
            </xs:element>
        </xs:choice>
        <xs:attribute name="timestamp" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="ParamType30">
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>
</xs:schema>
```

3 Contact Information

Document Version: 4.0
August 2007

EBS Dealing Resources, Inc.
Technology & Operations
One Upper Pond Road
Building F, 3rd Floor
Parsippany, New Jersey USA 07054
Tel. (973) 257-6600

For Customer Support:

Country	Freephone	Direct Dial	Country	Freephone	Direct Dial
Australia	1800121312	+ 44 (0) 20 7029 9344	Malaysia	1800802912	+ 44 (0) 20 7029 9373
Austria	0800291749	+ 44 (0) 20 7029 9345	Mexico	0018666391907	+ 44 (0) 20 7029 9374
Bahrain	80000375	+ 44 (0) 20 7029 9346	Monaco	N/A	+ 44 (0) 20 7029 9375
Barbados	18005340082	+ 44 (0) 20 7029 9347	Netherlands	08000224174	+ 44 (0) 20 7029 9376
Belgium	080010469	+ 44 (0) 20 7029 9348	New Zealand	0800444226	+ 44 (0) 20 7029 9377
Bermuda	18006230166	+ 44 (0) 20 7029 9349	Norway	80011816	+ 44 (0) 20 7029 9378
Brazil	0008110051994	+ 44 (0) 20 7029 9350	Panama	0018886608293	+ 44 (0) 20 7029 9379
Canada	18005762534	+ 44 (0) 20 7029 9351	Peru	080050793	+ 44 (0) 20 7029 9380
Chile	12300200651	+ 44 (0) 20 7029 9353	Philippines	180014410011	+ 44 (0) 20 7029 9381
China	108004400023	+ 44 (0) 20 7029 9354	Poland	008004411343	+ 44 (0) 20 7029 9382
Colombia	01800 9122064	+ 44 (0) 20 7029 9355	Portugal	800844130	+ 44 (0) 20 7029 9383
Cyprus	N/A	+ 44 (0) 20 7029 9356	Russia	74955809410	+ 44 (0) 20 7029 9384
Czech Republic	N/A	+ 44 (0) 20 7029 9357	Singapore	800 852 3666	+ 44 (0) 20 7029 9385
Denmark	80017779	+ 44 (0) 20 7029 9358	South Africa	0800991174	+ 44 (0) 20 7029 9386
Dubai	N/A	+ 44 (0) 20 7029 9359	South Korea	00308440046	+ 44 (0) 20 7029 9387
Finland	0800114424	+ 44 (0) 20 7029 9360	Spain	900974434	+ 44 (0) 20 7029 9388
France	0800908284	+ 44 (0) 20 7029 9361	Sweden	020792749	+ 44 (0) 20 7029 9389
Germany	08001810598	+ 44 (0) 20 7029 9362	Switzerland (D)	0800558443	+ 44 (0) 20 7029 9390
Greece	0080044129654	+ 44 (0) 20 7029 9363	Switzerland (F)	0800551368	+ 44 (0) 20 7029 9391
Hong Kong	800968580	+ 44 (0) 20 7029 9364	Switzerland (I)	0800551369	+ 44 (0) 20 7029 9392
Hungary	0680014347	+ 44 (0) 20 7029 9365	Taiwan	00801 444125	+ 44 (0) 20 7029 9393
Indonesia	001803440095	+ 44 (0) 20 7029 9366	Thailand	18004410152	+ 44 (0) 20 7029 9394
Ireland	1800409190	+ 44 (0) 20 7029 9367	Turkey	80044942788	+ 44 (0) 20 7029 9395
Israel	18009437368	+ 44 (0) 20 7029 9368	United Kingdom	0800446633	+ 44 (0) 20 7029 9396
Italy	800780939	+ 44 (0) 20 7029 9369	Uruguay	0004110052240	+ 44 (0) 20 7029 9397
Japan	0120653638	+ 44 (0) 20 7029 9370	USA	18008723373	+ 44 (0) 20 7029 9398
Lichtenstein	0800809417	+ 44 (0) 20 7029 9371	Venezuela	80015802	+ 44 (0) 20 7029 9399
Luxembourg	80022939	+ 44 (0) 20 7029 9372			

All calls to Customer Support are recorded.
Email: customersupport@icap.com