

Generic eXecution Accelerator GXA Lite

User Guide

About This Document

This document describes the GXA Lite user guide.

Copyright Information

Celoxica and the Celoxica logo are trademarks of Celoxica Limited.

All other products or services mentioned herein may be trademarks of their respective owners.

Neither the whole nor any part of the information contained in, or the product described in, this document may be adapted or reproduced in any material form except with the prior written permission of the copyright holder.

The product described in this document is subject to continuous development and improvement. All particulars of the product and its use contained in this document are given by Celoxica Limited in good faith. However, all warranties implied or express, including but not limited to implied warranties of merchantability, or fitness for purpose, are excluded.

This document is intended only to assist the reader in the use of the product. Celoxica Limited shall not be liable for any loss or damage arising from the use of any information in this document, or any incorrect use of the product. The information contained herein is subject to change without notice and is for general guidance only.

Copyright © 1991 - 2013 Celoxica Limited. All rights reserved.

Sales sales@celoxica.com
Customer Support support@celoxica.com
Website <http://www.celoxica.com>

UK Head Office

Celoxica Limited
34 Porchester Road
London
W2 6ES, UK
Phone: +44 (0) 20 7313 3180

US Head Office

Celoxica Inc.
275 Madison Avenue, Suite 404
New York, NY
10016, USA
Phone: +1 (0) 212 880 2075

US Chicago Office

Celoxica Inc.
141 W Jackson Blvd, Suite 2350
Chicago, IL
60604, USA
Phone: +1 (0) 312 893 1204

Content

1.	Functional Description	7
1.1	Recommended Reading	7
1.2	Overview	7
1.3	Order Recovery	7
1.4	Configuration	8
1.5	User API	8
1.5.1	Standard Usage	8
1.5.2	Order Recovery	10
1.6	Use Case	10
2.	Technical Architecture.....	11
2.1	GXA Lite Daemon Monitoring	11
3.	Configuration	12
3.1	Configuration File Schema	12
3.2	General Configuration Settings.....	16
3.2.1	General Configurations.....	16
3.2.2	Logger	16
3.2.3	Heart-beats.....	17
3.2.4	Scheduler	18
3.2.5	Libraries.....	19
3.2.6	Loops.....	20
3.2.7	Market	22
3.2.8	Input Queue.....	26
3.2.9	Output Queue.....	27
4.	User API.....	29
4.1	Functions	29
4.1.1	GXAGetError().....	29
4.1.2	GXACreateNewContext()	29
4.1.3	GXAInitContext()	29
4.1.4	GXADestroyContext().....	29

4.1.5	GXAReserveCommand()	29
4.1.6	GXAPushCommandContextStopOnDisconnect()	29
4.1.7	GXAPushCommandContextAbortOnDisconnect()	30
4.1.8	GXAPushCommandContextCloseMarketOnDisconnect()	30
4.1.9	GXAPushMarketConnect().....	30
4.1.10	GXAPushMarketDisconnect()	30
4.1.11	GXAPushNewOrder()	31
4.1.12	GXAPushModifyOrder()	31
4.1.13	GXAPushCancelOrder()	31
4.1.14	GXAGetInfo()	31
4.1.15	GXAReleaseInfo()	32
4.1.16	GXAPushRecoverActiveOrders().....	32
4.2	Format.....	32
4.2.1	Variables	32
4.2.2	Enumerations.....	32
4.2.3	Structures.....	37
4.2.4	Argument Types.....	41
5.	Market Specifics.....	42
5.1	ArcaDirect	42
5.1.1	Source	42
5.1.2	Configuration Settings.....	42
5.1.3	Symbology	45
5.1.4	New Order Message Mapping	45
5.1.5	Replace Order Message Mapping	48
5.1.6	Cancel Order Message Mapping	50
5.1.7	Inbound Data Mapping.....	51
5.2	CBOE CMi2.....	52
5.2.1	Source	52
5.2.2	Configuration Settings.....	52
5.2.3	Symbology	55
5.2.4	New Order Message Mapping	55

5.2.5	Replace Order Message Mapping	57	5.6.7	Inbound Data Mapping.....	100
5.2.6	Cancel Order Message Mapping	59	5.7	PHLX.....	101
5.2.7	Inbound Data Mapping	60	5.7.1	Source	101
5.3	CME.....	61	5.7.2	Configuration Settings.....	101
5.3.1	Source	61	5.7.3	Symbology	104
5.3.2	Configuration Settings	61	5.7.4	New Order Message Mapping	104
5.3.3	Symbology.....	66	5.7.5	Replace Order Message Mapping	106
5.3.4	New Order Message Mapping.....	66	5.7.6	Cancel Order Message Mapping	107
5.3.5	Replace Order Message Mapping.....	68	5.7.7	Inbound Data Mapping.....	109
5.3.6	Cancel Order Message Mapping	70			
5.3.7	Inbound Data Mapping	71			
5.3.8	Regulatory Audit Trail.....	71			
5.4	ICE.....	72			
5.4.1	Source	72			
5.4.2	Configuration Settings	72			
5.4.3	Symbology.....	79			
5.4.4	New Order Message Mapping.....	79			
5.4.5	Replace Order Message Mapping.....	81			
5.4.6	Cancel Order Message Mapping	82			
5.4.7	Inbound Data Mapping	83			
5.5	LIFFE.....	84			
5.5.1	Source	84			
5.5.2	Configuration Settings	84			
5.5.3	Symbology.....	89			
5.5.4	New Order Message Mapping.....	89			
5.5.5	Replace Order Message Mapping	91			
5.5.6	Cancel Order Message Mapping	92			
5.5.7	Inbound Data Mapping	93			
5.6	OUCH.....	94			
5.6.1	Source	94			
5.6.2	Configuration Settings	94			
5.6.3	Symbology.....	96			
5.6.4	New Order Message Mapping.....	96			
5.6.5	Replace Order Message Mapping	98			
5.6.6	Cancel Order Message Mapping	99			

Table of Figures

Figure 1 – Multiple user queues.....7

Figure 2 – Use case10

Figure 3 – GXA Lite daemon11

Revisions

Revision	Date	Description of Changes
R2013-1.0	28 DEC 2012	Release R2013-1.0 - Added LIFFE market specifics - Added <fixtrc-file> parameter for CME under config node config.gxa_loops.IO.market.parameters
R2012-8.1	20 DEC 2012	Release R2012-8.1 - Added CME market specifics - Added ICE market specifics - GXA Lite supports an order recovery facility - Added optional <checkalive-timeout-s> parameter for CBOE CMi2 under config node config.gxa_loops.IO.market.parameters
R2012-8.0	22 NOV 2012	Release R2012-8.0 - Added PHLX market specifics - Added TOE configuration parameters under config node config.gxa_loops.IO.market.ade - Added optional <recv-size>, <send-size> parameters under config node config.gxa_loops.IO.market.parameters for each market - GXAIInfo.ReasonCode is now populated in case of GXA Lite internal reject
R2012-7.0	10 OCT 2012	Release R2012-7.0 - Added field PositionEffect to structure GXANewCommand - Added GXA_ORDER_TYPE_MARKET_MAKER to enumeration GXAOderType - Added logger configuration parameter config.logger.trace-file - Added OUCH market specifics - Added optional <login-timeout>, <connect-timeout> parameters for CBOE under config node config.gxa_loops.IO.market.parameters - Added optional <checkalive-timeout-s> parameter for ArcaDirect under config node config.gxa_loops.IO.market.parameters - Updated ArcaDirect market specifics to support options
R2012-GXA2-alpha	17 SEP 2012	Initial release

1. Functional Description

1.1 Recommended Reading

This document should be read in conjunction with:

- The Celoxica Manager User Guide
- The PCI Express Card Hardware Installation Guide
- The PCI Express Card Software Installation Guide

1.2 Overview

Celoxica's Generic eXecution Accelerator GXA Lite is an FPGA-based solution providing ultra-low-latency order entry to exchanges and high-performance communication with a client software application.

User applications integrated with GXA Lite are able to leverage its functionality, which hides the complexities of working with multiple market access formats and facilitates order management.

GXA Lite ships as a set of dynamic libraries, which the user code can interface to via the single and simple user API it exposes.

GXA Lite enables multi-threading integration into the user applications code.

GXA Lite communicates with the exchanges using the execution venue native protocol.

GXA Lite can be configured to target several markets; it can also be configured to send commands e.g. order entry, order replace or order cancel via multiple shared memory input queues and receive market reports via multiple shared memory output queues. The user can assign priorities to the queues: for instance such assignments allow the user to give priority to new orders over amendments or cancellations.

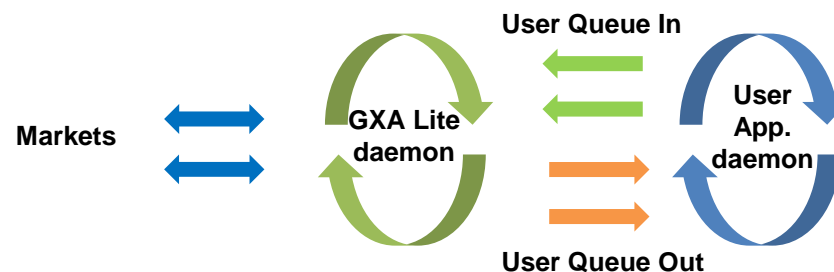


Figure 1 – Multiple user queues

The market session connection and disconnection can either be managed by GXA Lite or by the user application thanks to the configuration node `config.gxa_loops.IO.market.auto-connect` set in the configuration file.

GXA Lite provides a heart-beat service so that the user application continuously knows if GXA Lite daemon is running or not.

Exchanges often require the client order identifier to be unique per trading session. GXA Lite is able to generate and maintain client order identifiers on behalf of the user application.

GXA Lite does not maintain the order state: GXA Lite only makes available to the user the order state as provided by the exchange if the exchange provides order states.

1.3 Order Recovery

GXA Lite provides an order recovery facility for the following use cases:

1. The user application is restarted intra-day.

The user can retrieve all active orders from GXA Lite thanks to an API call (see User API section *Order Recovery* for more details).

2. GXA Lite is restarted intra-day.

GXA Lite recovers all active orders thanks to the GXA Lite order persistence feature. A configuration tag `config.gxa_loops.IO.market.persistence.recover` can enable or disable this functionality per market. When enabled, GXA Lite loads all

active orders thanks to previously submitted commands and received exchange-events automatically stored in internal files.

Note:

It is recommended to activate the persistence feature on FIX markets where the cancel on disconnect facility is not provided by the exchange.

The user can then retrieve the active orders from GXA Lite thanks to an API call (see User API section *Order Recovery* for more details).

1.4 Configuration

GXA Lite is configured using an XML-style '.cfg' file. When the GXA Lite initializes, it loads this configuration file. See section 3.1 for more details.

Example configuration files are shipped with your software. These files were created using reliable, fully tested settings. Celoxica highly recommends creating backup copies of any configuration files before editing.

One configuration file must be defined per GXA Lite instance; it allows the user to configure:

- The logger service for the GXA Lite daemon to log relevant information. See section 3.2.2 for more details.
- The heart-beat service for the user to ensure that GXA Lite daemon is running. See section 3.2.3 for more details.
- The scheduler service for the user to set a pool of threads for asynchronous tasks. These scheduled asynchronous tasks are GXA Lite daemon tasks. For instance they consist in checking if the socket is open, in sending test request ... See section 3.2.4 for more details.
- The libraries and their locations to be used. See section 3.2.5 for more details.
- The GXA Lite threads: the user can define as many GXA Lite threads as required. See section 3.2.6 for more details.

- The markets to be targeted. The user can define as many market sockets as required and configures one GXA Lite thread per market socket. See section 3.2.7 for more details.

The user can set the TOE TCP Offload Engine configurations.

The user can also set the order persistence configurations.

Some markets require specific configuration parameters. See section 5 for more details.

- The input queues: these shared memory queues are used by GXA Lite to send commands to the markets such as order creation, order amendment, order cancellation.

The user can define as many input queues as required and assigns one GXA Lite thread per input queue. See section 3.2.8 for more details.

- The output queues: these shared memory queues are used by GXA Lite to receive events from the markets such as acknowledgements, execution reports, rejection reports or non-order related events.

The user can define as many output queues as required and assigns one GXA Lite thread per output queue. See section 3.2.9 for more details.

1.5 User API

1.5.1 Standard Usage

Firstly the user has to initialize one or more GXA contexts. A GXA context mainly consists in associating one input queue with one output queue. Initialization of a GXA context is done as follows:

1. Firstly a GXA context is created.
Call `GXACreateNewContext()`.
2. Secondly the GXA context is initialized.
Call `GXAInitContext()`.

3. Before leaving GXA Lite, it is recommended to cancel existing GXA context.

Call `GXADestroyContext()`.

If the configuration node `config.gxa_loops.IO.market.auto-connect` is set, GXA Lite automatically initiates the market session. GXA Lite tries to maintain a constant connection to the exchange. If the connection is dropped GXA Lite logs the disconnection and tries to reconnect automatically until the connection is re-established.

If the configuration node `config.gxa_loops.IO.market.auto-connect` is not set, the user has to initiate the market session: call `GXAPushMarketConnect()` to push a market connection.

The user can terminate a market session at any time: call `GXAPushMarketDisconnect()` to push a market disconnection.

In order to anticipate any user application daemon failure, the user can configure GXA Lite to either:

1. Stop upon user disconnection

Call `GXAPushCommandContextStopOnDisconnect()`.

2. Or abort upon user disconnection

Call `GXAPushCommandContextAbortOnDisconnect()`.

3. Or properly close the market sessions upon user disconnection

Call `GXAPushCommandContextCloseMarketOnDisconnect()`.

Then the user can start sending commands (order creation, order amendment and order cancellation) to the markets via the shared memory input queues using a GXA context previously initialized. Sending commands is done as follows:

1. Firstly the user has to reserve a command of type:

- a. `GXA_NEW_ORDER` for an order entry
- b. `GXA_MODIFY_ORDER` for an order amendment

- c. `GXA_CANCEL_ORDER` for an order cancellation

This allows the user to specify the market he wants to target and the order characteristics.

Note:

The client order ID is an optional characteristic: it is recommended to let GXA Lite handle the client order IDs on behalf of the user.

Call `GXAReserveCommand()`.

2. Secondly the user sends the command to the market.

Call `GXAPushNewOrder()` to push an order creation to the market.

Call `GXAPushModifyOrder()` to push an order amendment to the market.

Call `GXAPushCancelOrder()` to push an order cancellation to the market.

A GXA ID is automatically assigned to the command upon its successful push. A new GXA ID is assigned to each successful command push even if the commands pertain to the same order.

If a client order ID was not assigned to the command upon its reservation, GXA Lite will automatically compute and assign a client order ID on behalf of the user according to the market specific client order ID rules.

Note:

The client order ID is not notified back to the user upon its computation.

The market feedbacks are received on the shared memory output queues specified by the GXA context. Note that the user is not notified upon receipt of market reports. The user is responsible for reading the output queue. Reading output queues is done as follows:

1. Firstly the user gets the information received from the market.

Call `GXAGetInfo()`.

If the market event is related to a command previously sent, the user will get the GXA ID of the command at the origin of the event for reconciliation purpose.

If the market event is not related to a command previously sent (exchange session is disconnected or exchange reports received after a client recovery following a crash), the data passed to the user will handle a GXA ID set to 0.

The data passed to the user bear the client order ID as computed by GXA Lite when the command was pushed.

2. Secondly the user releases the received information.

Call `GXAReleaseInfo()`.

1.5.2 Order Recovery

At any time the user can recover all active orders.

1. Firstly, the user requires GXA Lite to provide all active orders.

Call `GXAPushRecoverActiveOrders()`.

2. Secondly, the user gets the active order information.

Call `GXAGetInfo()`.

All active orders are forwarded to the user with an event `GXA_EVENT_ORDER_RECOVERED`. These events bear the GXA ID of the command at the origin of the order creation.

When all active orders have been processed, a final milestone is forwarded to the user with an event `GXA_EVENT_ALL_ORDERS_RECOVERED`. This last event bears the GXA ID returned by `GXAPushRecoverActiveOrders()`.

3. It is then strongly recommended to cancel the recovered orders.

1.6 Use Case

The GXA Lite daemon can be configured to target several markets, send commands via multiple shared memory input queues and receive market reports via multiple shared memory output queues.

The following figure illustrates a two-input queue and single output queue GXA Lite daemon configuration with two markets being targeted.

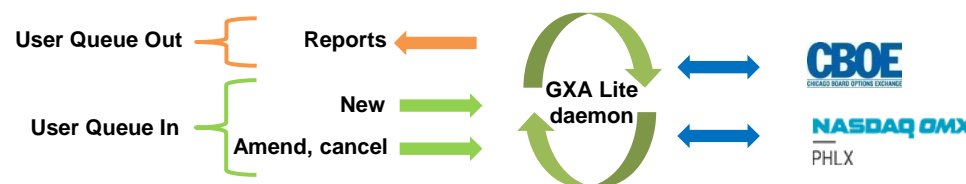


Figure 2 – Use case

2. Technical Architecture

This section should be read in conjunction with the Celoxica Manager User Guide.

2.1 GXA Lite Daemon Monitoring

GXA Lite is monitored by the Master Daemon: this daemon starts and stops processes, acts as a watchdog and sends alerts in case of issues.

GXA Lite uses the Logger Daemon: this daemon reads the log messages written to shared memory by GXA Lite, formats them and writes the formatted log messages to files.

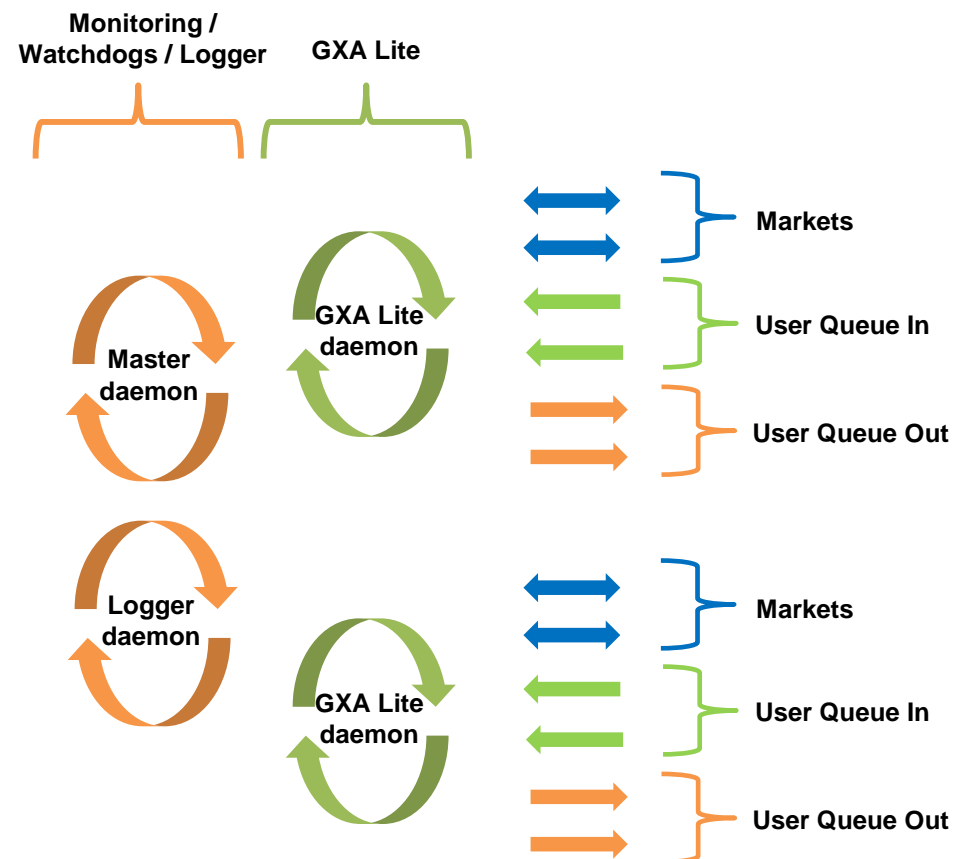


Figure 3 – GXA Lite daemon

3. Configuration

This section describes the GXA Lite configuration file and all available configuration options.

GXA Lite is configured using an XML-style '.cfg' file. When GXA Lite initializes, it loads this configuration file. See section 3.1 for more details.

Example configuration files are shipped with your software. These files were created using reliable, fully tested settings. Celoxica highly recommends creating backup copies of any configuration files before editing.

One configuration file must be defined per GXA Lite instance; it allows the user to configure:

- Some general settings; see section 3.2 for more details;
- Some market specific settings; as an example see section 5.2.2 for the CBOE CMi2 configuration settings.

3.1 Configuration File Schema

The basic structure of a GXA Lite configuration file is shown in the following schema. Tags have to be defined in the order as shown. This schema shows tags but not values.

```
<config>
  <logger>
    <key>
    <file>
    <size>
    <app-name>
    <level>
    <trace-file>
  </logger>

  <heart-beats>
    <host>
    <affinity>
  </heart-beats>

  <scheduler>
    <threads>
    <affinity>
  </scheduler>

  <libraries>
    <paths>
      <path>
    </paths>

    <library>
      <name>
      <entry>
      <file>
    </library>
  </libraries>

  <gxa_loops>
    <counters_key>

    <loops>
      <loop>
        <name>
        <strategy>
        <spin_on_read>
```

```
        <affinity>
    </loop>
</loops>

<IO>
    <market>
        <name>
        <id>
        <loop_name>
        <type>
        <host>
        <info_queue_id>
        <new_orders_pool_size>
        <mod_orders_pool_size>
        <auto_connect>

        <ade>
            <interface>
            <plugin>
        </ade>

        <persistence>
            <commands-file>
            <info-file>
            <market-specific-file>
            <recover>
        </persistence>

        <parameters>
            <!-- Market specific parameters -->
        </parameters>
    </market>

    <input_queue>
        <name>
        <id>
        <loop_name>
        <key>
        <size>
    </input_queue>
```

```
        <output_queue>
            <name>
            <id>
            <loop_name>
            <key>
            <size>
        </output_queue>
    </IO>
</gxa_loops>
</config>
```

3.2 General Configuration Settings

The first settings in a configuration file are general configurations (i.e. they are not market-dependent). This section describes each of these settings.

3.2.1 General Configurations

The general configuration allows the setting of parameters for GXA Lite.

3.2.1.1 config

Description This tag initiates a body for GXA Lite configurations to be made in a file. This tag is at the top of the file.

Usage `<config>`

3.2.2 Logger

The logger configuration allows the setting of parameters for GXA Lite to use the logger daemon.

3.2.2.1 config.logger

Description This tag initiates a body for the logger configurations to be made.

Usage `<logger>`

3.2.2.2 config.logger.key

Description Sets the logger daemon shared memory key.

Usage `<key>Key</key>`

Parameters Key Integer

Example `<key>11</key>`

3.2.2.3 config.logger.file

Description Sets the default log file if the logger daemon is not present.

Usage `<file>File</file>`

Parameters File File path

Example `<file>/tmp/log.log</file>`

3.2.2.4 config.logger.size

Description Sets the size of the shared memory if not created.

Usage `<size>Size</size>`

Parameters Size Log size expressed in bytes

Example `<size>5242880</size>`

3.2.2.5 config.logger.app-name

Description Sets the application name which will appear in the log files.

Usage `<app-name>Name</app-name>`

Parameters Name Application name

Example `<app-name>GXALite_cboe</app-name>`

3.2.2.6 config.logger.log-level

Description Sets the logging level.

Usage `<level>Level</level>`

Parameters Level Possible values are:

- NOTICE Log notice – normal, but significant, condition
- INFO Log info – informational message
- DEBUG Debug – debug-level message

Example `<level>INFO</level>`

3.2.2.7 config.logger.trace-file

Description Sets the prefix of the file in which traces will be written.
The date will be automatically appended to the prefix according to the following format prefix-year-month-day.
This parameter is optional: if it is not set and if GXA Lite is daemonized (the

process is running on the background), all traces will be lost.

Usage `<trace-file>Prefix-file</trace-file>`

Parameters Prefix-file File path

Example `<trace-file>/tmp/cboe-trace</trace-file>`
On 2012/10/05, the generated trace file is cboe-trace-2012-10-05

3.2.3 Heart-beats

The heart-beats configuration allows the setting of parameters for the user to ensure that GXA Lite is alive.

3.2.3.1 config.heart-beats

Description This tag initiates a body for the heart-beat service configurations to be made.

Usage `<heart-beats>`

3.2.3.2 config.heart-beats.host

Description Sets the socket UNIX or socket IP for the heart-beat service.

Usage `<host>Host</host>`

Parameters Host Can be defined as follows:

- Socket UNIX unix/UNIX file:Port
- Socket IP ip/IP address:Port

Example

```
<host>ip/*:51617</host>
<host>ip/10.20.100.200:51617</host>
<host>unix/UNIX-file:51617</host>
```

3.2.3.3 config.heart-beats.affinity

Description Sets the CPU on which the heart-beat service is authorized.
Optional tag.

Usage <affinity>Affinity</affinity>

Parameters Affinity Integer.
Can be defined as follows:

- i,...,j CPU i, CPU ... and CPU j are used
- i-j CPUs i to j are used
- i CPU i is used

Example

```
<affinity>0,4</affinity>
CPU 0 and CPU 4 are used
<affinity>0-4</affinity>
CPU 0, CPU 1, CPU 2, CPU 3 and CPU 4 are used
```

3.2.4 Scheduler

The scheduler configuration allows the setting of parameters for the user to define a pool of threads assigned for asynchronous tasks.

3.2.4.1 config.scheduler

Description This tag initiates a body for the scheduler configurations to be made.

Usage <scheduler>

3.2.4.2 config.scheduler.threads

Description Sets the number of threads.

Usage <threads>Threads</threads>

Parameters Threads Integer

Example <threads>3</threads>

3.2.4.3 config.scheduler.affinity

Description Sets the CPUs on which the scheduler are authorized.

Note:

A single affinity must be configured when the TOE parameters are set using *config.gxa_loops.IO.market.ade*.

Usage <affinity>Affinity</affinity>

Parameters	Affinity	Integer. Can be defined as follows:
		<ul style="list-style-type: none"> • i,...,j CPU i, CPU ... and CPU j are used • i-j CPUs i to j are used • i CPU i is used
Example	<pre><affinity>0,4</affinity></pre> <p>CPU 0 and CPU 4 are used</p> <pre><affinity>0-4</affinity></pre> <p>CPU 0, CPU 1, CPU 2, CPU 3 and CPU 4 are used</p>	

3.2.5 Libraries

The libraries configuration allows the setting of parameters for the user to define the library to be used.

At least one library must be defined.

3.2.5.1 config.libraries

Description This tag initiates a body for the libraries configurations to be made.

Usage `<libraries>`

3.2.5.2 config.libraries.paths

Description This tag initiates a body for the libraries paths configurations to be made.
The user can define as many libraries as required.

At least one library path must be defined.

Usage `<paths>`

3.2.5.3 config.libraries.paths.path

Description Sets the library path.
Several paths can be defined. In that case, the first path found in the configuration file is the path that will be checked in last.

Usage `<path>Path</path>`

Parameters Path Library path

Example

```
<paths>
```

```
  <path>/usr/lib/celoxica</path>
```

```
  <path>/usr/share/lib/celoxica</path>
```

```
</paths>
```

Path /usr/share/lib/celoxica is checked in first.
If the library is not found, path /usr/lib/celoxica is checked in last.

3.2.5.4 config.libraries.library

Description This tag initiates a body for the library configurations to be made.

Usage `<library>`

3.2.5.5 config.libraries.library.name

Description Sets the library name.
Must be unique across all libraries.

Note:

The library “ADEWrapper” must be configured when the TOE parameters are set using *config.gxa_loops.IO.market.ade*.

Usage `<name>Name</name>`

Parameters Name Library name

Example `<name>Arca</name>`

3.2.5.6 config.libraries.library.entry

Description Sets the initialization library.
See section 5 for details on the settings to use for each market.

Usage `<entry>Entry</entry>`

Parameters Entry Library entry name

Example `<entry>InitArca</entry>`

3.2.5.7 config.libraries.library.file

Description Sets the library file.

Usage `<file>File</file>`

Parameters File Library file

Example `<file>libgxa_arca.so</file>`

3.2.6 Loops

The loops configuration allows the setting of parameters for the user to define the threads to be used.

The user can define as many threads as required.

At least one thread must be defined.

3.2.6.1 config.gxa_loops

Description This tag initiates a body for the GXA Lite threads configurations to be made.

Usage `<gxa-loops>`

3.2.6.2 config.gxa_loops.counters_key

Description Sets the shared memory key used to initialize the GXA context using API functions

`GXACreateNewContext()` and `GXAInitContext()`.

Usage `<counters_key>Counters</counters_key>`

Parameters Counters Integer

Example `<counters_key>1519</counters_key>`

3.2.6.3 config.gxa_loops.loop

Description This tag initiates a body for the loops configurations to be made.

Usage `<loop>`

3.2.6.4 config.gxa_loops.loop.name

Description Sets the loop name.
Must be unique across all loops.

Usage `<name>Name</name>`

Parameters Name Loop name

Example `<name>loop1</name>`

3.2.6.5 config.gxa_loops.loop.strategy

Description Sets the loop threading strategy.

Usage `<strategy>Strategy</strategy>`

Parameters Strategy Threading strategy.
Can be defined as follows:

- “sequential”

Example `<strategy>sequential</strategy>`

3.2.6.6 config.gxa_loops.loop.spin_on_read

Description Sets if the thread is continuously read or not.

Usage `<spin_on_read>Spin</spin_on_read>`

Parameters Spin Can be defined as follows:

- “true” the thread is always read

Example `<spin_on_read>true</spin_on_read>`

3.2.6.7 config.gxa_loops.loop.affinity

Description Sets the CPUs on which the loops are authorized.

Note:

A single affinity must be configured when the TOE parameters are set using *config.gxa_loops.IO.market.adc*.

Usage `<affinity>Affinity</affinity>`

Parameters Affinity Integer.

Can be defined as follows:

- i,...,j CPU i, CPU ... and CPU j are used
- i-j CPUs i to j are used
- i CPU i is used

Example `<affinity>0,4</affinity>`

CPU 0 and CPU 4 are used

`<affinity>0-4</affinity>`

CPU 0, CPU 1, CPU 2, CPU 3 and CPU 4 are used

3.2.7 Market

The market configuration allows the setting of parameters for the user to define the markets to be targeted.

The user can define as many markets as required.

At least one market must be defined.

Some markets require specific configuration, see section 5.

3.2.7.1 config.gxa_loops.IO.market

Description This tag initiates a body for the market socket configurations to be made.

At least one market socket must be defined.

Usage `<market>`

3.2.7.2 config.gxa_loops.IO.market.name

Description Sets the market socket name.

Usage `<name>Name</name>`

Parameters Name Socket name

Example `<name>socket Arca</name>`

3.2.7.3 config.gxa_loops.IO.market.id

Description Sets the market socket ID.

Must be unique across all defined input queue ID, output queue ID and market socket ID.

Usage `<id>Id</id>`

Parameters Id Integer

Example `<id>3</id>`

3.2.7.4 config.gxa_loops.IO.market.loop_name

Description Sets the loop to be used.

The loop must be defined using *config.gxa_loops.loop.name*.

Usage `<loop_name>Loop</loop_name>`

Parameters Loop Loop name

Example <loop_name>loop1</loop_name>

3.2.7.5 config.gxa_loops.IO.market.type

Description Sets the market socket type defined in the library.

Must be equal to *config.libraries.library.name* except for ArcaDirect (see market specifics config node *config.gxa_loops.IO.market.type*).

Usage <type>Type</type>

Parameters Type Market socket type

Example <type>Arca</type>

3.2.7.6 config.gxa_loops.IO.market.host

Description Sets the market host.

Usage <host>Host</host>

Parameters Host Market host

Example <host>tcp/localhost:60003</host>

3.2.7.7 config.gxa_loops.IO.market.info_queue_id

Description Sets the default output queue ID if several output queues are defined.

The default output queue ID must be defined using *config.gxa_loops.IO.output_queue.id*.

Usage <info_queue_id>Id</info_queue_id>

Parameters Id Queue ID

Example <info_queue_id>7</info_queue_id>

3.2.7.8 config.gxa_loops.IO.market.new_orders_pool_size

Description Sets the size of the pool for new orders.

Usage <new_orders_pool_size>New</new_orders_pool_size>

Parameters New Integer

Example <new_orders_pool_size>500000</new_orders_pool_size>

3.2.7.9 config.gxa_loops.IO.market.mod_orders_pool_size

Description Sets the size of the pool for amend and cancel orders.

Usage <mod_orders_pool_size>Mod</mod_orders_pool_size>

Parameters	Mod	Integer
-------------------	-----	---------

Example `<mod orders pool size>100000</mod orders pool size>`

3.2.7.10 config.gxa_loops.IO.market.auto-connect

Description	<p>When enabled (value yes) the session connection to the market is managed by GXA Lite.</p> <p>When disabled (value no) the session connection to the market is managed by the user application.</p>
--------------------	---

Usage `<auto-connect>AutoConnect</auto-connect>`

Parameters	AutoConnect	Boolean.
		Mandatory.

Example `<auto-connect>yes</auto-connect>`

3.2.7.11 config.gxa_loops.IO.market.ade

Description	This tag initiates a body for the TCP Offload Engine TOE parameters configurations to be made.
--------------------	--

Note #1:

The library “ADEWrapper” must be configured using *config.libraries.library* when the TOE parameters are set.

Note #2:

A single affinity must be configured using `config.scheduler.affinity` when the TOE parameters are set.

Note #3:

A single affinity must be configured using `config.gxa_loops.loop.affinity` when the TOE parameters are set.

Usage `<ade>`

3.2.7.12 config.gxa_loops.IO.market.ade.interface

Description	Sets the physical interface on the accelerator card used for the TCP Offload Engine TOE.
--------------------	--

Usage `<interface>Interface</interface>`

Parameters	Interface	Interface name
------------	-----------	----------------

Example `<interface>ac2</interface>`

3.2.7.13 config.gxa_loops.IO.market.ade.plugin

Description	Sets the plugin, set to <code>toe</code> when using the TCP Offload Engine TOE on the accelerator card.
--------------------	---

Usage `<plugin>Plugin</plugin>`

Parameters Plugin "toe"

Example `<plugin>toe</plugin>`

3.2.7.14 config.gxa_loops.IO.market.parameters

Description This tag initiates a body for the market parameters configurations to be made.

Usage `<parameters>`

3.2.7.15 config.gxa_loops.IO.market.persistence

Description This tag initiates a body for the market order persistence configurations to be made.

Usage `<persistence>`

3.2.7.16 config.gxa_loops.IO.market.persistence.commands-file

Description Sets the file where all commands GXANewCommand are saved for recovery purpose.

This parameter is mandatory when config node `<persistence>` is set.

Usage `<commands-file>File</commands-file>`

Parameters File File name

Example `<commands-file>/tmp/cme_commands.data</commands-file>`

3.2.7.17 config.gxa_loops.IO.market.persistence.info-file

Description Sest the file where all GXInfo are saved for recovery purpose.
This parameter is mandatory when config node `<persistence>` is set.

Usage `<info-file>File</info-file>`

Parameters File File name

Example `<info-file>/tmp/cme_info.data</info-file>`

3.2.7.18 config.gxa_loops.IO.market.persistence.market-specific-file

Description Sets the file where some specific order life events are saved (for instance the last client Order ID of an amended order) for recovery purpose.
This parameter is mandatory when config node `<persistence>` is set.

Usage `<market-specific-file>File</market-specific-file>`

Parameters File File name

Example `<market-specific-file>/tmp/cme_market_spacific.data</market-specific-file>`

3.2.7.19 config.gxa_loops.IO.market.persistence.recover

Description When enabled and in case of intra-day restart, GXA Lite will load active orders using the three recovery files configured using *config.gxa_loops.IO.market.persistence.commands-file*, *config.gxa_loops.IO.market.persistence.info-file* and *config.gxa_loops.IO.market.persistence.market-specific-file*.

This parameter is mandatory when config node `<persistence>` is set.

Allowable values are:

- “yes”
- “no”

Usage `<recover>Recover</recover>`

Parameters Recover Boolean

Example `<recover>yes</recover>`

3.2.8 Input Queue

The input queue configuration allows the setting of parameters for the user to define the queues used to send commands to the market.

The user can define as many input queues as required.

At least one input queue must be defined.

3.2.8.1 config.gxa_loops.IO.input_queue

Description This tag initiates a body for the input queue configurations to be made.

Usage `<input_queue>`

3.2.8.2 config.gxa_loops.IO.input_queue.name

Description Sets the input queue name.

Usage `<name>Name</name>`

Parameters Name Input queue name

Example `<name>input_queue</name>`

3.2.8.3 config.gxa_loops.IO.input_queue.id

Description Sets the input queue ID.

Must be unique across all defined input queue ID, output queue ID and market socket ID.

An input queue with a lower ID has a higher priority than an input queue with a greater ID.

Usage `<id>Id</id>`

Parameters Id Strictly positive integer

Example `<id>1</id>`

3.2.8.4 config.gxa_loops.IO.input_queue.loop_name

Description Sets the loop to be used.

The loop must be defined using *config.gxa_loops.loop.name*.

Usage `<loop_name>Loop</loop_name>`

Parameters Loop Loop name

Example `<loop_name>loop1</loop_name>`

3.2.8.5 config.gxa_loops.IO.input_queue.key

Description Sets the shared memory key used by the input queue.

Usage `<key>Key</key>`

Parameters Key Integer

Example `<key>55612</key>`

3.2.8.6 config.gxa_loops.IO.input_queue.size

Description Sets the shared memory size for the input queue.

Usage `<size>Size</size>`

Parameters Size Integer

Example `<size>5242880</size>`

3.2.9 Output Queue

The output queue configuration allows the setting of parameters for the user to define the queues used to receive events from the market.

The user can define as many output queues as required.

At least one output queue must be defined.

3.2.9.1 config.gxa_loops.IO.output_queue

Description This tag initiates a body for the output queue configurations to be made.

Usage `<output_queue>`

3.2.9.2 config.gxa_loops.IO.output_queue.name

Description Sets the output queue name.

Usage `<name>Name</name>`

Parameters Name Output queue name

Example `<name>output queue</name>`

3.2.9.3 config.gxa_loops.IO.output_queue.id

Description Sets the output queue ID.

Must be unique across all defined input queue ID, output queue ID and market socket ID.

Usage `<id>Id</id>`

Parameters `Id` Strictly positive integer

Example `<id>7</id>`

3.2.9.4 `config.gxa_loops.IO.output_queue.loop_name`

Description Sets the loop to be used.
The loop must be defined using `config.gxa_loops.loop.name`.

Usage `<loop_name>Loop</loop_name>`

Parameters `Loop` Loop name

Example `<loop_name>loop 1</loop_name>`

3.2.9.5 `config.gxa_loops.IO.output_queue.key`

Description Sets the shared memory key used by the output queue.

Usage `<key>Key</key>`

Parameters `Key` Integer

Example `<key>55614</key>`

3.2.9.6 `config.gxa_loops.IO.output_queue.size`

Description Sets the shared memory size for the output queue.

Usage `<size>Size</size>`

Parameters `Size` Integer

Example `<size>5242880</size>`

4. User API

4.1 Functions

4.1.1 GXAGetError()

Description Gets the alphanumeric error description of a GXA error.

Prototype `char* GXAGetError(GXAErrors)`

Arguments `GXAErrors` GXA error

4.1.2 GXACreateNewContext()

Description Creates a GXA context: a pointer is returned.

Prototype `GXAContext* GXACreateNewContext()`

4.1.3 GXAINitContext()

Description Initializes the GXA context.

Prototype `GXAErrors GXAINitContext(GXAContext* ctx)`

Arguments `ctx` GXA context

4.1.4 GXADestroyContext()

Description Destroys the GXA context.

This function should be used before leaving GXA Lite.

Prototype `GXAErrors GXADestroyContext(GXAContext* ctx)`

Arguments `ctx` GXA context

4.1.5 GXAReserveCommand()

Description Reserves a command.

This function must be called before pushing the command.

A new command cannot be reserved if the previously reserved command has not been pushed.

Prototype `GXAErrors GXAReserveCommand(GXAContext* ctx, GXANewCommand** retCmd)`

Arguments `ctx` GXA context
`retCmd` Command

4.1.6 GXAPushCommandContextStopOnDisconnect()

Description This function asks GXA Lite to properly close all market connections if the user application dies. GXA Lite daemon will be killed. All orders will therefore be lost.

It returns a command ID for the command.

Prototype `GXAEErrors
GXAPushComandContextStopOnDisconnect (GXAContext*
ctx, uint64_t* commandId)`

Arguments `ctx` GXA context
`commandID` Command ID

4.1.7 GXAPushCommandContextAbortOnDisconnect()

Description This function asks GXA Lite to immediately abort. GXA Lite daemon will be killed. All orders will therefore be lost.

It returns a command ID for the command.

Prototype `GXAEErrors
GXAPushCommandContextAbortOnDisconnect (GXAContext*
ctx, uint64_t* commandId)`

Arguments `ctx` GXA context
`commandID` Command ID

4.1.8 GXAPushCommandContextCloseMarketOnDisconnect()

Description This function asks GXA Lite to properly close a market connection if the user application dies. GXA Lite daemon remains alive.

It returns a command ID for the command.

Prototype `GXAEErrors
GXAPushCommandContextCloseMarketOnDisconnect (GXAC
ontext* ctx, GXAMarketID target, uint64_t*`

`commandId)`

Arguments `ctx` GXA context
`target` Market identifier
`commandID` Command ID

4.1.9 GXAPushMarketConnect()

Description This function pushes a market connection to the market.

The function must be used to initiate a session connection when configuration node *config.gxa_loops.IO.market.auto-connect* is not set.

It returns a GXA ID for the command. The command is of type `GXA_CONNECT_MARKET` and has been automatically reserved by GXA Lite on behalf of the user.

Prototype `GXAEErrors GXAPushMarketConnect (GXAContext* ctx,
GXAMarketID target, uint64_t* retGXAID)`

Arguments `ctx` GXA context
`target` Market identifier
`retGXAID` GXA ID

4.1.10 GXAPushMarketDisconnect()

Description This function pushes a market disconnection to the market.

It returns a GXA ID for the command. The command is of type `GXA_DISCONNECT_MARKET` and has been automatically reserved by GXA Lite on behalf of the user.

Prototype `GXAEErrors GXAPushMarketDisconnect(GXAContext* ctx, GXAMarketID target, uint64_t* retGXAIID)`

Arguments

<code>ctx</code>	GXA context
<code>target</code>	Market identifier
<code>retGXAIID</code>	GXA ID

4.1.11 GXAPushNewOrder()

Description This function pushes an order creation to the market.

It returns a GXA ID for the command which must have been previously reserved using *GXAReserveCommand()*.

Prototype `GXAEErrors GXAPushNewOrder(GXAContext* ctx, GXANewCommand** cmd, uint64_t* retGXAIID)`

Arguments

<code>ctx</code>	GXA context
<code>cmd</code>	Command
<code>GXAID</code>	GXA ID

4.1.12 GXAPushModifyOrder()

Description This function pushes an order amendment to the market.

It returns a GXA ID for the command which must have been previously reserved using *GXAReserveCommand()*.

Prototype `GXAEErrors GXAPushModifyOrder(GXAContext* ctx, GXANewCommand** cmd, uint64_t* retGXAIID)`

Arguments

<code>ctx</code>	GXA context
<code>cmd</code>	Command
<code>GXAID</code>	GXA ID

4.1.13 GXAPushCancelOrder()

Description This function pushes and order cancellation to the market.

It returns a GXA ID for the command which must have been previously reserved using *GXAReserveCommand()*.

Prototype `GXAEErrors GXAPushCancelOrder(GXAContext* ctx, GXANewCommand** cmd, uint64_t* retGXAIID)`

Arguments

<code>ctx</code>	GXA context
<code>cmd</code>	Command
<code>GXAID</code>	GXA ID

4.1.14 GXAGetInfo()

Description This function provides the events received from the market. It also provides active orders when the recovery facility is called.

If no event is available, the function will return a null pointer.

Prototype `GXAEErrors GXAGetInfo(GXAContext* ctx, GXAINfo** info)`

Arguments

<code>ctx</code>	GXA context
<code>info</code>	Market event

4.1.15 GXAReleaseInfo()

Description This function releases the events received from the market once provided by `GXAGetInfo()`.

There is no need to call the function if `GXAGetInfo()` has returned a null pointer.

Prototype `GXAErrors GXAReleaseInfo(GXAContext* ctx, GXAINfo** info)`

Arguments

<code>ctx</code>	GXA context
<code>info</code>	Market event

4.1.16 GXAPushRecoverActiveOrders()

Description This function pushes a command to recover all active orders.

It returns a GXA ID for the command. The command is of type `GXA_RECOVER_ACTIVE_ORDERS` and has been automatically reserved by GXA Lite on behalf of the user.

Prototype `GXAErrors GXAPushRecoverActiveOrders(GXAContext* ctx, GXAMarketID target, uint64_t* retGXAID)`

Arguments

<code>ctx</code>	GXA context
<code>target</code>	Market identifier
<code>retGXAID</code>	GXA ID

4.2 Format

Prices (both inbound and outbound) are coded with fixed point decimal places. For example, to place an order with price 12.34, the API needs to take number 1234000000.

See the header files for more details.

4.2.1 Variables

The following table provides the values of some GXA variables:

Variable	Value	Description
<code>GXA_SYMBOL_LENGTH</code>	105	Maximum symbol length
<code>GXA_CLIENT_ORDERID_LENGTH</code>	32	Maximum client order ID length
<code>GXA_EXCHANGE_ORDERID_LENGTH</code>	20	Maximum exchange order ID length
<code>GXA_EXECUTIONID_LENGTH</code>	32	Maximum execution ID length
<code>GXA_DECIMAL_PLACE</code>	8	Decimal places
<code>GXA_ACCOUNT_LENGTH</code>	12	Maximum order account length
<code>GXA_SENDERSUBID_LENGTH</code>	12	Maximum sender sub ID length
<code>GXA_OPAQUE_STR_LENGTH</code>	16	Maximum opaque string length
<code>GXA_MAX_REJECT_REASON_LENGTH</code>	150	Maximum reject reason length
<code>GXA_MAX_TOKEN_LENGTH</code>	32	Maximum client order ID length
<code>GXA_TRADERID_LENGTH</code>	32	Maximum trader ID length
<code>GXA_COMMAND_RESULT_REJECT_SIZE</code>	128	Maximum command reject reason length

4.2.2 Enumerations

This section provides enumerations.

4.2.2.1 Order Side

Description Order side list.

Enumeration GXAOrderSide

Values	Description
GXA_SIDE_BUY	Buy
GXA_SIDE_SELL	Sell
GXA_SIDE_SELL_SHORT	Sell short
GXA_SIDE_SELL_SHORT_EXEMPT	Sell short exempt
GXA_SIDE_CROSS	Cross (<u>not supported yet</u>)
GXA_SIDE_CROSS_SHORT	Cross short (<u>not supported yet</u>)
GXA_SIDE_CROSS_SHORT_EXEMPT	Cross short exempt (<u>not supported yet</u>)

4.2.2.2 Order TIF

Description Order TIF list.

Enumeration GXAOrderTIF

Values	Description
GXA_TIF_IOC	Immediate or cancel (default value)
GXA_TIF_GTC	Good till cancel
GXA_TIF_GTX	Good till crossing
GXA_TIF_DAY	Day
GXA_TIF_FOK	Fill or kill
GXA_TIF_GTD	Good till date (reserved for future use)

4.2.2.3 Order Type

Description Order type list.

Enumeration GXAOrderType

Values	Description
GXA_ORDER_TYPE_MARKET	Market order
GXA_ORDER_TYPE_LIMIT	Limit order
GXA_ORDER_TYPE_CABINET	Cabinet order
GXA_ORDER_TYPE_LIMIT_ON_OPEN	Limit on open order
GXA_ORDER_TYPE_LIMIT_ON_CLOSE	Limit on close order
GXA_ORDER_TYPE_MARKET_ON_OPEN	Market on open order
GXA_ORDER_TYPE_MARKET_ON_CLOSE	Market on close order
GXA_ORDER_TYPE_INSIDE_LIMIT	Inside limit order
GXA_ORDER_TYPE_POST_ONLY	Post only order
GXA_ORDER_TYPE_STOP_LOSS	Stop loss order
GXA_ORDER_TYPE_STOP_LIMIT	Stop limit order
GXA_ORDER_TYPE_PEGGED	Pegged order
GXA_ORDER_TYPE_PNP_BLIND	Post no preference order
GXA_ORDER_TYPE_ICEBERG	Iceberg order
GXA_ORDER_TYPE_MARKET_MAKER	Market maker order
GXA_ORDER_TYPE_INVALID	Invalid type
GXA_ORDER_TYPE_NOTPRESENT	Missing order type

4.2.2.4 Liquidity Indicator

Description Liquidity indicator list.

Enumeration GXALiquidityIndicator

Values	Description
GXA_LIQUIDITY_ADDED	The order added liquidity
GXA_LIQUIDITY_REDUCED	The order reduced liquidity
GXA_LIQUIDITY_NEUTRAL	The order was liquidity neutral
GXA_LIQUIDITY_ROUTED	The order was routed
GXA_LIQUIDITY_OTHER	The value for the liquidity flag is not handled by GXA yet

4.2.2.5 Market State

Description Market state list.

Enumeration GXAMarketState

Values	Description
GXA_MARKET_UNINITIALIZED	The connection to the market is not established yet or the market state is unknown
GXA_MARKET_OPEN	The market is open for trading
GXA_MARKET_STOPPED	The market has stopped and won't reopen for the day
GXA_MARKET_SUSPENDED	The market is suspended and should reopen
GXA_MARKET_CLOSED	The market is closed

Values	Description
GXA_MARKET_CONNECTION_LOST	The connection to the market has been lost

4.2.2.6 Order State

Description Order state list.

Enumeration GXAOOrderState

Values	Description
GXA_ORDER_ACTIVE	The order was acknowledged
GXA_ORDER_INACTIVE	The order was cancelled or has been fully filled

4.2.2.7 Order Trading Session

Description Order trading session list.

Enumeration GXAOOrderTradingSession

Values	Description
GXA_TS_MARKETHOURS	All day during the market hours
GXA_TS_SYSTEMHOURS	All day but during the system hours
GXA_TS_INVALID	Invalid trading session

4.2.2.8 Peg Type

Description Peg type list.

Enumeration GXAPegType

Values	Description
GXA_PEG_TYPE_MARKET	Peg to the opposite side
GXA_PEG_TYPE_PRIMARY	Peg to the same side
GXA_PEG_TYPE_MID	Peg to the midpoint (default)
GXA_PEG_TYPE_INVALID	Invalid peg type
GXA_PEG_TYPE_ROUTED	Peg eligible to execution against routable orders

4.2.2.9 Context Event

Description Context event list.

Enumeration GXAContextEvent

Values	Description
EVENT_CNX_UP	
EVENT_CNX_CANNOT_CONNECT	
EVENT_CNX_DOWN	
EVENT_CNX_LEAVING	
EVENT_CNX_NO_HEART_BEAT	
EVENT_CTX_COMMAND	
EVENT_CTX_BAD_RECV	

4.2.2.10 Errors

Description Error list.

Enumeration GXAErrors

Values	Description
GXA_NO_ERROR	No error
GXA_ERR_CTX_ALREADY_INITIALIZED	Context is already initialized
GXA_ERR_CTX_INVALID_SHARED_COUNTER	Context shared counter key is not valid
GXA_ERR_CTX_OPEN_SHARED_COUNTER	Error while opening the context shared counter
GXA_ERR_CTX_NOT_API	Context has not been created via the API
GXA_ERR_CTX_INVALID_MISSING_QUEUES	No key has been given to input and output queue
GXA_ERR_CTX_INVALID_MISSING_CALLBACK	No callback has been given
GXA_ERR_CTX_OPEN_INPUT_QUEUE	Error while opening the context input queue
GXA_ERR_CTX_OPEN_OUTPUT_QUEUE	Error while opening the context output queue
GXA_ERR_INVALID_CTX	Invalid context
GXA_ERR_INVALID_COMMAND	Invalid command
GXA_ERR_CTX_INPUT_OUT_OF_SPACE	There is no available space in the input queue
GXA_ERR_CTX_INPUT_ITEM	The command has already been reserved
GXA_ERR_CTX_INPUT_NO_ITEM	The command has not been reserved
GXA_ERR_CTX_OUTPUT_ITEM	Only one GXAIInfo can be used at once
GXA_SIZE_MISMATCH	expected size mismatches with queue object size
GXA_ERR_CTX_CNX	Error while trying to connect, typically GXA Lite daemon has not been started
GXA_ERR_CTX_CNX_CLOSED	Internal error, GXA Lite daemon has failed
GXA_ERR_CTX_CNX_SEND	GXA Lite daemon has failed

Values	Description
GXA_ERR_NULL_PARAMETER	One function parameter is null
GXA_ERR_UNKNOWN	Unknown error
GXA_ERR_CTX_DEAD	The context is dead
GXA_ERR_CTX_ALREADY_INITIALIZING	the context is already initialized
GXA_ERR_NO_INPUT_QUEUE	Invalid input queue parameter
GXA_ERR_NO_OUTPUT_QUEUE	Invalid output queue parameter

4.2.2.11 Command Type

Description Command type list.

Enumeration GXACommandType

Values	Description
GXA_NEW_ORDER	Order creation
GXA_MODIFY_ORDER	Order amendment
GXA_CANCEL_ORDER	Order cancellation
GXA_CONNECT_MARKET	Market session connection
GXA_DISCONNECT_MARKET	Market session disconnection
GXA_RECOVER_ACTIVE_ORDERS	Active orders recovery

4.2.2.12 Event

Description Event list.

Enumeration GXAEvent

Values	Description
GXA_EVENT_INSERT_ACK	The order inserted has been acknowledged
GXA_EVENT_INSERT_NACK	The order insert has been rejected
GXA_EVENT_CANCEL_ACK	The order cancel has been acknowledged
GXA_EVENT_CANCEL_NACK	The order cancel has been rejected
GXA_EVENT_AMEND_ACK	The order amend has been acknowledged
GXA_EVENT_AMEND_NACK	The order amend has been rejected
GXA_EVENT_UNSOLICITED_ORDER_CHANGE	An unsolicited order change occurred
GXA_EVENT_UNSOLICITED_ORDER_CANCEL	An unsolicited order cancellation occurred
GXA_EVENT_TRADE	An order was traded
GXA_EVENT_TRADE_BUSTED	The trade was busted by the exchange
GXA_EVENT_TRADE_CORRECTED	The trade was corrected
GXA_EVENT_ORDER_RECOVERED	The order was received after startup
GXA_EVENT_TASK_REJECTED	The task was rejected by GXA due to invalid state or parameters
GXA_EVENT_CONNECTION_EXISTS	Connection to the exchange already exists
GXA_EVENT_CONNECTION_ESTABLISHED	Connection to the exchange was established
GXA_EVENT_CONNECTION_ERROR	An error occurred while connecting to the exchange
GXA_EVENT_CONNECTION_LOST	Connection to the exchange has been lost
GXA_EVENT_MARKET_STATE_CHANGE	The state of the market changed

Values	Description
GXA_EVENT_MARKET_START_OF_DAY	The market is starting for the day
GXA_EVENT_ORDER_RECOVERED	The order has been recovered
GXA_EVENT_ALL_ORDERS_RECOVERED	All orders have been recovered

4.2.2.13 Heart-beat-Connection Mode

Description Heart-beat connection mode list.

Enumeration HeartBeatsCnxMode

Values	Description
HB_MODE_IP	GXA Lite heart-beats use a socket IP
HB_MODE_UNIX	GXA Lite heart-beats use a socket UNIX

4.2.2.14 Command Status

Description Command status list.

Enumeration GXAContextCommandStatus

Values	Description
CTX_COMMAND_REGISTERED	The command is registered
CTX_COMMAND_REJECTED	The command is rejected

4.2.2.15 Position Effect

Description Position effect list.

Enumeration GXAPositionEffect

Values	Description
GXA_OPENED_POSITION	Open position
GXA_CLOSED_POSITION	Closed position

4.2.3 Structures

The presence map is a bit combination indicating the presence or the absence of a field in the message. If the presence map bit is set, it indicates that the field is sent within the message and if not, it indicates that the field is omitted.

Fields indexed with “PM” e.g. **Account^{PM}** are subject to the presence map: for these fields, check the presence map bit to check whether the field is present or not in the message.

4.2.3.1 GXInfo

Description The structure handles:

- The events received from the market i.e. acknowledgements, rejections, executions and non-order related events
- The events related to the recovery facility.

Structure GXInfo

The structure description is as follows:

Field	Type	Description
Event	GXAEvent	Market event

Field	Type	Description
Market	GXAMarketID	Market ID
GXAID	GXAIDType	Command GXA ID at the origin of the event. Sets to 0 if the event is not related to a command pushed to the market.
The client order ID is defined either as a string or as a binary		
ClientOrderIDString^{PM}	GXAClientOrderID	Client Order ID defined as a string
ClientOrderIDUInt64^{PM}	uint64_t	Client Order ID defined as a binary
The order ID assigned by the exchange is defined either as a string or as a binary		
ExchangeOrderIDString^{P_M}	char	Order ID assigned by the exchange and defined as a string. Maximum length defined by GXA_EXCHANGE_ORDERID_LENGTH
ExchangeOrderIDUInt64^{PM}	uint64_t	Order ID assigned by the exchange and defined as a binary.
The trade matched ID assigned by the exchange is defined either as a string or as a binary		
ExecutionIDString^{PM}	GXAExecutionID	Matching ID for this particular trade defined as a string
ExecutionIDUInt64^{PM}	uint64_t	Matching ID for this particular trade defined as a binary
LiquidityIndicator^{PM}	GXALiquidityIndicator	Liquidity indicator
MarketState^{PM}	GXAMarketState	Market state
OrderState^{PM}	GXAOrderState	Order state
Price^{PM}	int64_t	Order price
Quantity^{PM}	uint32_t	Order quantity
Reason^{PM}	char	Indication of the reason for a command to be rejected or an order to be cancelled.

Field	Type	Description
		Maximum length defined by GXA_MAX_REJECT_REASON_LENGTH
ReasonCode^{PM}	uint32_t	Populated in the event of a GXA Lite internal reject. <u>Possible values:</u> 0 No problem (no rejection) 1 The OrderState does not allow this operation (typically the order has already been cancelled) 2 Invalid input command (typically mismatch between server and client headers) 3 Unmanaged input message 4 The market is not available 5 Error returned by the plugin 6 Transmission error
Side^{PM}	GXAOrderSide	Order side
The symbol is defined either as a string or as a binary		
Symbol^{PM}	GXAGlobalName	Symbol name defined as a string Maximum length defined by GXA_SYMBOL_LENGTH
SymbolUInt64^{PM}	uint64_t	Symbol ID defined as a binary
TotalTradedVolume^{PM}	uint32_t	Total amount of shares traded for this order
TradedPrice	int64_t	Price at which this part of the order was executed
TradedVolume^{PM}	uint32_t	The volume executed in this trade
Type^{PM}	GXAOrderType	Order type
OpaqueStr1^{PM}	char	Opaque string for specific market functionality. Maximum length defined by GXA_OPAQUE_STR_LENGTH
OpaqueChar1^{PM}	char	Opaque char for specific market

Field	Type	Description
		functionality
OpaqueInt1^{PM}	int64_t	Opaque integer for specific market functionality
StrategyId^{PM}	uint32_t	Strategy identifier in case of strategy creation message
MappableID^{PM}	int64_t	Mappable identifier associated with the order
ExchangeTimestamp^{PM}	uint64_t	Exchange timestamp in microseconds since epoch

4.2.3.2 GXANewCommand

Description The structure handles the command to be pushed to the market.

Structure GXANewCommand

The structure description is as follows:

Field	Type	Description
The symbol is defined either as a string or as a binary		
Name^{PM}	GXAGlobalName	<p>Symbol name defined as a string.</p> <p>For options only: when NameOSI^{PM} is set in the presence map, Name follows an extended Option Symbology Initiative defined as a string:</p> <ul style="list-style-type: none"> Name on 9 bytes right padded with spaces Expiry year on 2 bytes Expiry month on 2 bytes Expiry day on 2 bytes Call/Put on 1 byte Strike price on 5 bytes Strike price decimal on 3 bytes

Field	Type	Description
NameUInt64^{PM}	uint64_t	Symbol ID defined a binary
Price	int64_t	Order price
Size	uint32_t	Order size
TimeInForce	GXAOrderTIF	Order TIF
CommandType	GXACommandType	Command type
Side	GXAOrderSide	Order side
Type	GXAOrderType	Order type
TradingSession	GXAOrderTradingSession	Order trading times
GXAID	GXAIDType	GXA ID assigned to the command when pushed to the market
GXAOriginalID^{PM}	GXAIDType	Original GXA ID assigned to the command to be modified or cancelled
AltPrice^{PM}	int64_t	Alternate Price. Either stop price for stop orders or peg difference for pegged orders.
MinQuantity^{PM}	uint32_t	Minimum quantity
MaxFloor^{PM}	uint32_t	Displayed quantity
PegType^{PM}	GXAPegType	Peg type
Account^{PM}	char	Order account Maximum length defined by GXA_ACCOUNT_LENGTH
Rule80A^{PM}	char	Rule80A to override the default value
ISO^{PM}	char	ISO flag to override the default value <u>Possible values:</u> 'Y' ISO 'N' Not ISO
SenderSubID^{PM}	char	SenderSubId to be used for this order.

Field	Type	Description
		If not specified, the default SenderSubID specified in the configuration file will be used. Maximum length defined by GXA_SENDERSUBID_LENGTH
TraderID^{PM}	char	Trader ID to be used for this order (if supported on this market) Maximum length defined by GXA_TRADERID_LENGTH
OpaqueStr1^{PM}	char	Opaque string for specific market functionality. Maximum length defined by GXA_OPAQUE_STR_LENGTH
OpaqueStr2^{PM}	char	Opaque string for specific market functionality. Maximum length defined by GXA_OPAQUE_STR_LENGTH
OpaqueInt1^{PM}	int64_t	Opaque integer for specific market functionality
OpaqueInt2^{PM}	int64_t	Opaque integer for specific market functionality
OpaqueChar1^{PM}	char	Opaque char for specific market functionality
OpaqueChar2^{PM}	char	Opaque char for specific market functionality
MappableID^{PM}	int64_t	Mappable identifier associated with the order
PositionEffect^{PM}	GXAPositionEffect	Position effect (open or close)
The client order ID is defined either as a string or as a binary		
ClientOrderIDString^{PM}	GXAClientOrderID	Client Order ID defined as a string
ClientOrderIDUInt64^{PM}	uint64_t	Client Order ID defined as a binary
The order ID assigned by the exchange is defined either as a string or as a binary		

Field	Type	Description
ExchangeOrderIDString^{PM}	GXAExchangeOrderID	Order ID assigned by the exchange and defined as a string. Maximum length defined by GXA_EXCHANGE_ORDERID_LENGTH
ExchangeOrderIDUInt64^{PM}	uint64_t	Order ID assigned by the exchange and defined as a binary.
TotalTradedVolume^{PM}	uint32_t	Total amount of shares traded for this order (for GTC recovery).
TargetMarketId	GXAMarketID	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseId	uint16_t	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

4.2.3.3 GXAContext

Description The structure handles the GXA context to be initialized.

Structure GXAContext

The structure description is as follows:

Field	Type	Description
SharedCounters	key_t	Shared memory key set using configuration tag <i>config.gxa_loops.counters_key</i>
Input	key_t	Input queue ID set using configuration tag <i>config.gxa_loops.IO.input_queue_id</i>
Output	key_t	Output queue ID set using configuration tag <i>config.gxa_loops.IO.output_queue_id</i>

Field	Type	Description
Host	char [256]	UNIX file if the heart-beat service mode is socket UNIX
		IP address if the heart-beat service mode is socket IP
Port	unsigned short	Port associated to the host
Mode	HeartBeatsCnxMode	GXA context heart-beat mode
Callback	GXAContextCallback	Call back function
void* PrivateData		User's private data, optional

4.2.3.4 GXACommandResult

Description The structure handles the command status.

Structure GXACommandResult

The structure description is as follows:

Field	Type	Description
CmdID	uint64_t	Command ID
CmdStatus	GXAContextCommandStatus	Command status
RejectReason	char	Reject reason.
		Maximum length defined by GXA_COMMAND_RESULT_REJECT_SIZE

4.2.4 Argument Types

The following table provides the argument types description:

Enumeration	Description
GXAMarketID	uint8_t
GXAClientOrderID	char GXAClientOrderID [GXA_CLIENT_ORDERID_LENGTH]
GXAGlobalName	char GXAGlobalName [GXA_SYMBOL_LENGTH]
GXAExchangeOrderID	char GXAExchangeOrderID [GXA_EXCHANGE_ORDERID_LENGTH]
GXAExecutionID	char GXAExecutionID [GXA_EXECUTIONID_LENGTH]
GXAContextCallback	void *GXAContextCallback (struct GXAContext*, GXAContextEvent, GXACommandResult*)

5. Market Specifics

This section describes market specifics for the supported plugins. Refer to the market documentation as provided in the source sections to get the actual list of available values for some fields.

Boolean variables in the configuration files can have any of the following values:

on	off
yes	no

5.1 ArcaDirect

5.1.1 Source

Documentation	ArcaDirect API Specification
Version	4.0w
Date	June 15, 2011

5.1.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
ARCA	ArcaDirect	ARCA Equities
ARCA OX	ArcaDirect	ARCA Option
AMEX OX	ArcaDirect	AMEX Option

The basic structure of a GXA Lite market specific parameters configuration for ArcaDirect is shown in the following schema:

```
<parameters>
  <recv-size>
  <send-size>
  <checkalive-timeout-s>
  <logonid>
  <groupid>
  <account>
  <start-seq>
  <self-trade-prevention>
  <trading-session-code>
  <position-effect>
</parameters>
```

5.1.2.1 config.gxa_loops.IO.market.type

Description Sets the market socket type.

Usage `<type>Type</type>`

Parameters Type Must be set to “**Arca_Options**” (valid for both stock and option markets)

Example `<type>Arca_Options</type>`

5.1.2.2 config.libraries.library.entry

Description Sets the initialization library for ArcaDirect.

Usage `<entry>Entry</entry>`

Parameters Entry Must be set to “initArca” (valid for both stock and option markets)

Example <entry>initArca</entry>

5.1.2.3 config.gxa_loops.IO.market.parameters.recv-size

Description Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite).

Used only when the TCP Offload Engine TOE is not used.

Usage <recv-size>RecvSize</recv-size>

Parameters RecvSize Default is 65536 bytes

Example <recv-size>65536</recv-size>

5.1.2.4 config.gxa_loops.IO.market.parameters.send-size

Description Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).

Used only when the TCP Offload Engine TOE is not used.

Usage <send-size>SendSize</send-size>

Parameters SendSize Default is 65536 bytes

Example <send-size>65536</send-size>

5.1.2.5 config.gxa_loops.IO.market.parameters.checkalive-timeout-s

Description Sets the time in seconds to wait without receiving a heartbeat before disconnection.

Usage <checkalive-timeout-s>Timeout</checkalive-timeout-s>

Parameters Timeout Default is 30 seconds

Example <checkalive-timeout-s>20</checkalive-timeout-s>

5.1.2.6 config.gxa_loops.IO.market.parameters.logonid

Description ArcaDirect logon id.
Used to populate AppName field of the Logon message.

Usage <logonid>AppName</logonid>

Parameters AppName Logon id

Example <logonid>id</logonid>

5.1.2.7 config.gxa_loops.IO.market.parameters.groupid

Description The company ID for the firm entering the order.
Used to populate the CompanyGroupId field of the New Order message.

Usage <groupid>CompanyId</groupid>

Parameters CompanyGroupId Company ID

Example <groupid>id</groupid>

5.1.2.8 config.gxa_loops.IO.market.parameters.account

Description Default executing account.
Used to populate field Account if the account has not been set in *GXANewCommand*.

Usage <account>Account</account>

Parameters Account Account

Example <account>Account123</account>

5.1.2.9 config.gxa_loops.IO.market.parameters.start-seq

Description Default starting sequence number.
If not set, the ARCA logon message will be sent with last sequence number set to -1 (do not replay any message but continue from the last known

transmitted sequence plus one).

Usage <start-seq>StartSeq</start-seq>

Parameters StartSeq Sequence number

Example <start-seq>1000</start-seq>

5.1.2.10 config.gxa_loops.IO.market.parameters.self-trade-prevention

Description Used to populate NoSelfTrade.
If set to "yes", NoSelfTrade will be populated with 'C' (Cancel both) for all orders.
Used for equities only.
Allowable values are:

- "yes" (default value)
- "no"

Usage <self-trade-prevention>NoSelfTrade</self-trade-prevention>

Parameters NoSelfTrade NoSelfTrade

Example <self-trade-prevention>no</self-trade-prevention>

5.1.2.11 config.gxa_loops.IO.market.parameters.trading-session-code

Description Used to populate ExDestination.

Allowable values are:

- 102 ARCA Equities
- 103 ARCA Options
- 104 AMEX Options

Usage `<trading-session-code>ExDestination</trading-session-code>`

Parameters ExDestination ExDestination

Example `<trading-session-code>103</trading-session-code>`

5.1.2.12 config.gxa_loops.IO.market.parameters.position-effect

Description Default position effect.

Used to populate Open or Close.

Used for options only.

Allowable values are:

- 'O' Open
- 'C' Close

Usage `<position-effect>PositionEffect</position-effect>`

Parameters PositionEffect Open or Close

Example `<position-effect>C</position-effect>`

5.1.3 Symbology

Instruments are identified using the following symbology:

Type	Format	Exchange Field
Equity	String(8)	Symbol
Options	NameOSI ¹	Symbol, Strike Price, Put or Call, Strike Date

5.1.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd Equity	Req'd Option	Description
Name	Y	Y	Used to set Symbol for equities. Used to set Symbol, Strike Price, Put or Call, Strike Date for options (NameOSI must be set in the presence map).
NameUInt64	N	N	Not used
Price	Y	Y	Used to set Price
Size	Y	Y	Used to set OrderQty
TimeInForce	Y	Y	Used to set TimeInForce. <u>Allowable TIFs:</u> GXA_TIF_DAY GXA_TIF_IOC GXA_TIF_FOK GXA_TIF_GTX If TimeInForce = GXA_TIF_GTX: <ul style="list-style-type: none"> • TimeInForce forced to GXA_TIF_DAY

¹ See section 4.2.3.2 for more details

Field	Req'd Equity	Req'd Option	Description
			<ul style="list-style-type: none"> TradingSession = GXA_TS_SYSTEMHOURS
CommandType	Y	Y	GXA_NEW_ORDER to send message type D, variant 2
Side	Y	Y	Used to set Side. Allowable sides: GXA_SIDE_BUY GXA_SIDE_SELL GXA_SIDE_SELL_SHORT
Type	Y	Y	Used to set OrderType. <u>Allowable order types for equities:</u> GXA_ORDER_TYPE_MARKET <ul style="list-style-type: none"> Order Type = 1 (Market) GXA_ORDER_TYPE_LIMIT <ul style="list-style-type: none"> Order Type = 2 (Limit) GXA_ORDER_TYPE_INSIDE_LIMIT <ul style="list-style-type: none"> Order Type = 7 (Inside limit) GXA_ORDER_TYPE_POST_ONLY: <ul style="list-style-type: none"> Order Type = 2 (Limit) Execution Instructions = 6 Extended Execution Instructions 1 = A (Add Liquidity Only) GXA_ORDER_TYPE_MARKET_ON_OPEN <ul style="list-style-type: none"> Order Type = 1 (Market) Time In Force = 2 Trading Sessions = 2 GXA_ORDER_TYPE_LIMIT_ON_CLOSE <ul style="list-style-type: none"> Order Type = 2 (Limit) Time In Force = 7 (Closing auction only) GXA_ORDER_TYPE_MARKET_ON_CL

Field	Req'd Equity	Req'd Option	Description
			OSE <ul style="list-style-type: none"> Order Type = 1 (Market) Time In Force = 7 (Closing auction only) GXA_ORDER_TYPE_PNP_BLIND <ul style="list-style-type: none"> Order Type = 2 (Limit) Execution Instructions = 6 Extended PNP = B (Extended PNP Blind) GXA_ORDER_TYPE_MARKET_MAKER Order Type = q (Market maker) <u>Allowable order types for options:</u> GXA_ORDER_TYPE_MARKET <ul style="list-style-type: none"> Order Type = 1 (Market) GXA_ORDER_TYPE_LIMIT <ul style="list-style-type: none"> Order Type = 2 (Limit) GXA_ORDER_TYPE_PNP_BLIND <ul style="list-style-type: none"> Order Type = 2 (Limit) Execution Instructions = 6 Extended PNP = B (Extended PNP Blind) GXA_ORDER_TYPE_MARKET_ON_OPEN <ul style="list-style-type: none"> Order Type = 1 (Market) Time In Force = 2 Trading Sessions = 2
TradingSession	Y	Y	Used to set TradingSessionID
GXAOriginalID	N	N	Not used
AltPrice	N	N	Not used
MinQuantity	N	N	Not used
MaxFloor	N	N	Not used

Field	Req'd Equity	Req'd Option	Description
PegType	N	N	Not used
Account	O	O	Used to set Account. If not specified, the default Account <i>config.gxa_loops.IO.market.parameters.account</i> will be used.
Rule80A	Y	N	Used to set Rule80A
ISO	Y	Y	Used to set ISO. <u>Allowable flags:</u> 'Y' ISO 'N' Not ISO
SenderSubID	O	O	Used to set SenderSubID
TraderID	N	N	Not used
OpaqueStr1	N	O	Used to set Clearing Firm
OpaqueStr2	N	O	Used to set Clearing Account
OpaqueInt1	N	O	Used to set LocalOrAway Mandatory only when CustomerOrFirm = 3 (market maker).
OpaqueInt2	N	N	Not used
OpaqueChar1	N	O	Used to set CustomerOrFirm
OpaqueChar2	N	N	Not used
MappableID	O	O	A mappable ID associated with an order
PositionEffect	N	O	Used to set Open or Close. If not specified for options, the default PositionEffect <i>config.gxa_loops.IO.market.parameters.position-effect</i> will be used.
ClientOrderIDString	N	N	Not used

Field	Req'd Equity	Req'd Option	Description
ClientOrderIDUInt64	O	O	Used to set Client Order ID. Managed by GXA Lite when not provided
ExchangeOrderIDString	N	N	Not used
ExchangeOrderIDUInt64	N	N	Not used
TotalTradedVolume	N	N	Not used
TargetMarketID	Y	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Link ID	0
ExDestination	Set using <i>config.gxa_loops.IO.market.parameters.trading-session-code</i>
DeliverToCompID	""
Price Scale	4
Extended PNP	Set to 'B' if Type = GXA_ORDER_TYPE_PNP_BLIND Otherwise set to 0x00
Execution Instructions	Not used
NoSelfTrade	Set using <i>config.gxa_loops.IO.market.parameters.self-trade-prevention</i> for equities

Field	Value
Underlying Quantity	0 for options, otherwise not set
Corporate Action	'0' for options, otherwise not set
Extended Execution Instructions 1	Set to 'A' if Type = GXA_ORDER_TYPE_POST_ONLY for equities Otherwise set to 0x00

5.1.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd Equity	Req'd Option	Description
Name	Y	Y	Used to set Symbol for equities. Used to set Symbol, Strike Price, Put or Call, Strike Date for options (NameOSI must be set in the presence map).
NameUInt64	N	N	Not used
Price	Y	Y	Used to set Price
Size	Y	Y	Used to set Order Quantity
TimeInForce	Y	Y	Used to set TimeInForce. Allowable TIFs: GXA_TIF_DAY GXA_TIF_IOC GXA_TIF_FOK GXA_TIF_GTX If TimeInForce = GXA_TIF_GTX: <ul style="list-style-type: none"> TimeInForce forced to GXA_TIF_DAY TradingSession = GXA_TS_SYSTEMHOURS
CommandType	Y	Y	GXA_MODIFY_ORDER to send message type G, variant 1

Field	Req'd Equity	Req'd Option	Description
Side	Y	Y	Used to set Side. Allowable sides: GXA_SIDE_BUY GXA_SIDE_SELL GXA_SIDE_SELL_SHORT
Type	Y	Y	Used to set OrderType. Allowable order types for equities: GXA_ORDER_TYPE_MARKET <ul style="list-style-type: none"> Order Type = 1 (Market) GXA_ORDER_TYPE_LIMIT <ul style="list-style-type: none"> Order Type = 2 (Limit) GXA_ORDER_TYPE_INSIDE_LIMIT <ul style="list-style-type: none"> Order Type = 7 (Inside limit) GXA_ORDER_TYPE_POST_ONLY: <ul style="list-style-type: none"> Order Type = 2 (Limit) Execution Instructions = 6 Extended Execution Instructions 1 = A (Add Liquidity Only) GXA_ORDER_TYPE_MARKET_ON_OPEN <ul style="list-style-type: none"> Order Type = 1 (Market) Time In Force = 2 Trading Sessions = 2 GXA_ORDER_TYPE_LIMIT_ON_CLOSE <ul style="list-style-type: none"> Order Type = 2 (Limit) Time In Force = 7 (Closing auction only) GXA_ORDER_TYPE_MARKET_ON_CLOSE <ul style="list-style-type: none"> Order Type = 1 (Market) Time In Force = 7 (Closing auction only) GXA_ORDER_TYPE_PNP_BLIND

Field	Req'd Equity	Req'd Option	Description
			<ul style="list-style-type: none"> Order Type = 2 (Limit) Execution Instructions = 6 Extended PNP = B (Extended PNP Blind) <p>GXA_ORDER_TYPE_MARKET_MAKER Order Type = q (Market maker)</p> <p><u>Allowable order types for options:</u> GXA_ORDER_TYPE_MARKET</p> <ul style="list-style-type: none"> Order Type = 1 (Market) <p>GXA_ORDER_TYPE_LIMIT</p> <ul style="list-style-type: none"> Order Type = 2 (Limit) <p>GXA_ORDER_TYPE_PNP_BLIND</p> <ul style="list-style-type: none"> Order Type = 2 (Limit) Execution Instructions = 6 Extended PNP = B (Extended PNP Blind) <p>GXA_ORDER_TYPE_MARKET_ON_OPEN</p> <ul style="list-style-type: none"> Order Type = 1 (Market) Time In Force = 2 Trading Sessions = 2
TradingSession	Y	Y	Used to set TradingSessionID
GXAOriginalID	Y	Y	GXAID to be modified
AltPrice	N	N	Not used
MinQuantity	N	N	Not used
MaxFloor	N	N	Not used
PegType	N	N	Not used
Account	O	O	Used to set Account. If not specified, the default Account <i>config.gxa_loops.IO.market.parameters.a</i>

Field	Req'd Equity	Req'd Option	Description
			<i>ccount</i> will be used.
Rule80A	Y	N	Used to set Rule80A
ISO	Y	Y	Used to set ISO. <u>Allowable flags:</u> 'Y' ISO 'N' Not ISO
SenderSubID	N	N	Not used
TraderID	N	N	Not used
OpaqueStr1	N	N	Not used
OpaqueStr2	N	N	Not used
OpaqueInt1	N	N	Not used
OpaqueInt2	N	N	Not used
OpaqueChar1	N	N	Not used
OpaqueChar2	N	N	Not used
MappableID	N	N	Not used
PositionEffect	N	O	Used to set Open or Close. If not specified for options, the default PositionEffect <i>config.gxa_loops.IO.market.parameters.p</i> <i>osition-effect</i> will be used.
ClientOrderIDString	N	N	Not used
ClientOrderIDUInt64	N	N	Not used
ExchangeOrderIDString	Y	Y	Used to set OrderID
ExchangeOrderIDUInt64	N	N	Not used
TotalTradedVolume	N	N	Not used
TargetMarketID	Y	Y	Market socket ID set using configuration tag

Field	Req'd Equity	Req'd Option	Description
			<i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Original ClOrdID	Set by GXA Lite
Display Range	Not set
ExDestination	Set using <i>config.gxa_loops.IO.market.parameters.trading-session-code</i>
DeliverToCompID	""
Price Scale	4
Corporate Action	'0' for options, otherwise not set
Under Qty	0 for options, otherwise not set
Execution Instructions	Set by GXA Lite

5.1.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd Equity	Req'd Option	Description
Name	Y	Y	Used to set Symbol for equities.

Field	Req'd Equity	Req'd Option	Description
			Used to set Symbol, Strike Price, Put or Call, Strike Date for options (NameOSI must be set in the presence map).
NameUInt64	N	N	Not used
Price	N	N	Not used
Size	N	N	Not used
TimeInForce	N	N	Not used
CommandType	Y	Y	GXA_CANCEL_ORDER to send message type F, variant 1
Side	N	N	Not used
Type	N	N	Not used
TradingSession	N	N	Not used
GXAOriginalID	Y	Y	GXAID to be cancelled
AltPrice	N	N	Not used
MinQuantity	N	N	Not used
MaxFloor	N	N	Not used
PegType	N	N	Not used
Account	N	N	Not used
Rule80A	N	N	Not used
ISO	N	N	Not used
SenderSubID	N	N	Not used
TraderID	N	N	Not used
OpaqueStr1	N	N	Not used
OpaqueStr2	N	N	Not used
OpaqueInt1	N	N	Not used

Field	Req'd Equity	Req'd Option	Description
OpaqueInt2	N	N	Not used
OpaqueChar1	N	N	Not used
OpaqueChar2	N	N	Not used
MappableID	N	N	Not used
PositionEffect	N	N	Not used
ClientOrderIDString	N	N	Not used
ClientOrderIDUInt64	N	N	Not used
ExchangeOrderIDString	N	N	Not used
ExchangeOrderIDUInt64	N	N	Not used
TotalTradedVolume	N	N	Not used
TargetMarketID	Y	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the `GXANewCommand` structure:

Field	Value
Original ClOrdID	Set by GXA Lite
ExDestination	Set using <i>config.gxa_loops.IO.market.parameters.trading-session-code</i>
Bulk Cancel	0

Field	Value
DeliverToCompID	Not set
Under Qty	0
Corporate Action	'0'
Open or Close	Set by GXA Lite
OrderID	Set by GXA Lite

5.1.7 Inbound Data Mapping

There are no market specifics.

5.2 CBOE CMi2

5.2.1 Source

Documentation CMi2 programmer's Guide API Specification

Version 1.04

Date June 15, 2012

5.2.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
CBOE	CMi2	Chicago Board Options Exchange
C2	CMi2	CBOE C2
One Chicago	CMi2	CBOE One Chicago
CFE	CMi2	CBOE Futures Exchange
CBSX	CMi2	CBOE Stock Exchange

The basic structure of a GXA Lite market specific parameters configuration for CBOE is shown in the following schema:

```
<parameters>
  <market-counters-shm-key>
  <recv-size>
  <send-size>
  <checkalive-timeout-s>
  <logonid>
  <password>
  <login-mode>
```

```
    <executing-give-up-firm>
    <referential-mode>
    <referential-file>
    <trading-session-code>
    <login-timeout>
    <connect-timeout>
  </parameters>
```

5.2.2.1 config.libraries.library.entry

Description Sets the initialization library for CBOE.

Usage <entry>Entry</entry>

Parameters Entry Must be set to "initCboe"

Example <entry>initCboe</entry>

5.2.2.2 config.gxa_loops.IO.market.parameters.market-counters-shm-key

Description Shared memory key for market specifics counters.

Usage <market-counters-shm-key>Key</market-counters-shm-key>

Parameters Key Integer

Example <market-counters-shm-key>2512</market-counters-shm-key>

5.2.2.3 config.gxa_loops.IO.market.parameters.recv-size

Description Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite).

Used only when the TCP Offload Engine TOE is not used.

Usage `<recv-size>RecvSize</recv-size>`

Parameters RecvSize Default is 65536 bytes

Example `<recv-size>65536</recv-size>`

5.2.2.4 config.gxa_loops.IO.market.parameters.send-size

Description Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).

Used only when the TCP Offload Engine TOE is not used.

Usage `<send-size>SendSize</send-size>`

Parameters SendSize Default is 65536 bytes

Example `<send-size>65536</send-size>`

5.2.2.5 config.gxa_loops.IO.market.parameters.checkalive-timeout-s

Description Sets the time in seconds to wait without receiving a heartbeat before

disconnection.

Usage `<checkalive-timeout-s>Timeout</checkalive-timeout-s>`

Parameters Timeout Default is 30 seconds

Example `<checkalive-timeout-s>20</checkalive-timeout-s>`

5.2.2.6 config.gxa_loops.IO.market.parameters.logonid

Description Used to populate User Id field of the Logon message.

Usage `<logonid>UserId</logonid>`

Parameters UserId Logon id

Example `<logonid>id</logonid>`

5.2.2.7 config.gxa_loops.IO.market.parameters.password

Description Used to populate the Password field of the Logon message.

Usage `<password>Password</password>`

Parameters Password Password

Example `<password>password</password>`

5.2.2.8 `config.gxa_loops.IO.market.parameters.login-mode`

Description Used to populate the Login Mode of the Logon message.

Allowable values are:

- 'STAND_ALONE_TEST'
- 'NETWORK_TEST'
- 'PRODUCTION'

Usage `<login-mode>LoginMode</login-mode>`

Parameters LoginMode Login Mode

Example `<login-mode>PRODUCTION</login-mode>`

5.2.2.9 `config.gxa_loops.IO.market.parameters.executing-give-up-firm`

Description Used to populate Executing Give Up Firm associated with the order.

Usage `<executing-give-up-firm>Firm</executing-give-up-firm>`

Parameters Firm Firm identifier

Example `<executing-give-up-firm>501</executing-give-up-firm>`

5.2.2.10 `config.gxa_loops.IO.market.parameters.referential-mode`

Description enables (value yes) or disables (value no) the referential request.

Usage `<referential-mode>RefMode</referential-mode>`

Parameters RefMode Boolean

Example `<referential-mode>yes</referential-mode>`

5.2.2.11 `config.gxa_loops.IO.market.parameters.referential-file`

Description Sets the file where the referential is dumped.

Usage `<referential-file>File</referential-file>`

Parameters File File name

Example `<referential-file>/tmp/cboe_ref.txt</referential-file>`

5.2.2.12 `config.gxa_loops.IO.market.parameters.trading-session-code`

Description Sets the trading session code.

Allowable values are:

- 'W_MAIN'
- 'CFE_MAIN'

- 'ONE_MAIN'
- 'W_STOCK'
- 'C2_MAIN'
- 'COF_MAIN'
- 'FLEX_MAIN'

Usage `<trading-session-code>Session</trading-session-code>`

Parameters Session Trading session code

Example `<trading-session-code>W_MAIN</trading-session-code>`

5.2.2.13 config.gxa_loops.IO.market.parameters.login-timeout

Description Sets the maximum time in seconds allowed by the daemon, between login request sent and login response received.

Usage `<login-timeout>Timeout</login-timeout>`

Parameters Timeout Default is 10 seconds

Example `<login-timeout>120</login-timeout>`

5.2.2.14 config.gxa_loops.IO.market.parameters.connect-timeout

Description Sets the maximum time in seconds allowed to establish a TCP connection

with the market.

Usage `<connect-timeout>Timeout</connect-timeout>`

Parameters Timeout Default is 5 seconds

Example `<connect-timeout>10</connect-timeout>`

5.2.3 Symbology

Instruments are identified using the following symbology:

Format	Exchange Field
UInt64	Product Key

5.2.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameUInt64	Y	Used to set Product Key
Price	Y	Used to set Price
Size	Y	Used to set Original Quantity
TimeInForce	Y	Used to set Time In Force. <u>Allowable TIFs:</u> GXA_TIF_DAY GXA_TIF_GTC
CommandType	Y	GXA_NEW_ORDER to send a message Order Entry, format 1

Field	Req'd	Description
Side	Y	Used to set Side. Allowable sides: GXA_SIDE_BUY GXA_SIDE_SELL GXA_SIDE_SELL_SHORT GXA_SIDE_SELL_SHORT_EXEMPT
Type	Y	Used to set Price Type. Allowable order types: GXA_ORDER_TYPE_MARKET GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_CABINET
TradingSession	N	Not used
GXAOriginalID	N	Not used
AltPrice	N	Not used
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	Y	Used to set Account
Rule80A	Y	Used to set Origin Type
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used

Field	Req'd	Description
OpaqueChar2	N	Not used
MappableID	O	A mappable ID associated with an order
PositionEffect	O	Used to set Position Effect
ClientOrderIDString	N	Not used
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Branch	Set by GXA Lite
Order Date	Current date
Branch Sequence Number	Set by GXA Lite
Trading Session Id	Set using <i>config.gxa_loops.IO.market.parameters.trading-session-code</i>
Coverage	'C' (covered)
NBBO Protection	2 (full)
Executing give Up Firm Number	Set using <i>config.gxa_loops.IO.market.parameters.executing-give-up-firm</i>

Field	Value
Correspondent Firm	Not set
Subaccount	Not set
CMTA Firm Number	Not set
Extensions	Not set
Optional Data	Not set
User Assigned Id	Not set
Originator Acronym	Not set
Contingency Type	Not set
Contingency Price Type	Not set
Contingency Price	Not set
Contingency Volume	Not set
Preferred	Not set
Auction Id High	Not set
Auction Id Low	Not set
Clearing Info	Not set

5.2.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameUInt64	Y	Used to set Product Key
Price	Y	Used to set Price
Size	Y	Used to set Quantity

Field	Req'd	Description
TimeInForce	Y	Used to set Time In Force. Allowable TIFs: GXA_TIF_DAY GXA_TIF_GTC
CommandType	Y	GXA_MODIFY_ORDER to send a message Order Cancel / Replace, format 1
Side	Y	Used to set Side. Allowable sides: GXA_SIDE_BUY GXA_SIDE_SELL GXA_SIDE_SELL_SHORT GXA_SIDE_SELL_SHORT_EXEMPT
Type	Y	Used to set Price Type. Allowable order types: GXA_ORDER_TYPE_MARKET GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_CABINET
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be modified
AltPrice	N	Not used
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	Y	Used to set Account
Rule80A	Y	Used to set Origin Type
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used

Field	Req'd	Description
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	O	Used to set Position Effect
ClientOrderIDString	N	Not used
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Branch	Set by GXA Lite
Original Order Branch	Set by GXA Lite
Branch Sequence Number	Set by GXA Lite

Field	Value
Original Order Branch Seq Number	Set by GXA Lite
Order Date	Current date
Original Order Date	Set by GXA Lite
Trading Session Id	Set using <i>config.gxa_loops.IO.market.parameters.trading-session-code</i>
Coverage	'C' (covered)
NBBO Protection	2 (full)
Executing give Up Firm Number	Set using <i>config.gxa_loops.IO.market.parameters.executing-give-up-firm</i>
Original Order Executing give Up Firm Number	Set using <i>config.gxa_loops.IO.market.parameters.executing-give-up-firm</i>
Correspondent Firm	Not set
Original Order Correspondent Firm	Not set
Subaccount	Not set
CMTA Firm Number	Not set
Extensions	Not set
Optional Data	Not set
User Assigned Id	Not set
Original User Assigned Id	Not set
Originator Acronym	Not set
Contingency Type	Not set
Contingency Price Type	Not set
Contingency Price	Not set
Contingency Volume	Not set

Field	Value
Preferred	Not set
Original Order Id High	Set by GXA Lite
Original Order Id Low	Set by GXA Lite
Clearing Info	Not set

5.2.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameUInt64	Y	Used to set Product Key
Price	N	Not used
Size	N	Not used. Quantity To Cancel is set by GXA Lite with the remaining quantity. Therefore the cancellation is requested for the whole remaining quantity.
TimeInForce	N	Not used
CommandType	Y	GXA_CANCEL_ORDER to send a message Order Cancel, format 1
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be cancelled
AltPrice	N	Not used
MinQuantity	N	Not used
MaxFloor	N	Not used

Field	Req'd	Description
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	N	Not used
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <code>config.gxa_loops.IO.market.id</code>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <code>config.gxa_loops.IO.market.info_queue_id</code> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Branch	Set by GXA Lite
Order Date	Set by GXA Lite
Branch Sequence Number	Set by GXA Lite
Executing give Up Firm Number	Set using config.gxa_loops.IO.market.parameters.executing-give-up-firm
User Assigned id	Not set

5.2.7 Inbound Data Mapping

There are no market specifics.

5.3 CME

5.3.1 Source

Documentation	CME wiki: http://www.cmegroup.com/confluence/display/EPICSANDBOX/iLink+Order+Entry
Version	N/A
Date	N/A

5.3.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
CME	iLink	CME
CBOT	iLink	CBOT
COMEX	iLink	COMEX
NYMEX	iLink	NYMEX

The basic structure of a GXA Lite market specific parameters configuration for CME is shown in the following schema:

```
<parameters>
  <market-counters-shm-key>
  <recv-size>
  <send-size>
  <objects-pool-size>
  <sender-comp-id>
  <sender-sub-id>
  <heartbeat-interval>
```

```
<checkalive-timeout-s>
<customer-or-firm>
<sender-location-id>
<application-system-name>
<application-system-version>
<application-system-vendor>
<password>
<account>
<cti-code>
<in-flight-mitigation>
<reset-sequence>
<forward-session-reject>
<compute-fix-time>
<fixtrc-file>
</parameters>
```

5.3.2.1 config.libraries.library.entry

Description Sets the initialization library for CME.

Usage <entry>Entry</entry>

Parameters Entry Must be set to “initCme”

Example <entry>initCme</entry>

5.3.2.2 config.gxa_loops.IO.market.parameters.market-counters-shm-key

Description Shared memory key for market specifics counters.

Usage <market-counters-shm-key>Key</market-counters-shm-key>

Parameters Key Integer

Example `<market-counters-shm-key>2512</market-counters-shm-key>`

5.3.2.3 config.gxa_loops.IO.market.parameters.recv-size

Description Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite).

Used only when the TCP Offload Engine TOE is not used.

Usage `<recv-size>RecvSize</recv-size>`

Parameters RecvSize Default is 65536 bytes

Example `<recv-size>65536</recv-size>`

5.3.2.4 config.gxa_loops.IO.market.parameters.send-size

Description Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).

Used only when the TCP Offload Engine TOE is not used.

Usage `<send-size>SendSize</send-size>`

Parameters SendSize Default is 65536 bytes

Example `<send-size>65536</send-size>`

5.3.2.5 config.gxa_loops.IO.market.parameters.objects-pool-size

Description Number of pre-allocated objects in memory.

Each FIX message is defined as a class: the objects created for each class can be pre-allocated in memory to speed up the FIX protocol.

GXA Lite asynchronously cleans the pre-allocated objects every second.

Usage `<objects-pool-size>ObjectsPoolSize</objects-pool-size>`

Parameters ObjectsPoolSize Integer

Example `<objects-pool-size>1000</objects-pool-size>`

5.3.2.6 config.gxa_loops.IO.market.parameters.sender-comp-id

Description Used to populate SenderCompId [49] in the standard message header.

Usage `<sender-comp-id>SenderCompId</sender-comp-id>`

Parameters SenderCompId Tag 49

Example `<sender-comp-id>CompId</sender-comp-id>`

5.3.2.7 config.gxa_loops.IO.market.parameters.sender-sub-id

Description	Default SenderSubId [50].
Usage	<code><sender-sub-id>SenderSubId</sender-sub-id></code>
Parameters	SenderSubId Tag 50
Example	<code><sender-sub-id>SubId</sender-sub-id></code>

5.3.2.8 config.gxa_loops.IO.market.parameters.heartbeat-interval

Description	Sets then interval, in seconds, for sending or expecting heartbeats.
Usage	<code><heartbeat-interval>Interval</heartbeat-interval></code>
Parameters	Interval Integer
Example	<code><heartbeat-interval>30</heartbeat-interval></code>

5.3.2.9 config.gxa_loops.IO.market.parameters.checkalive-timeout-s

Description	Sets the time in seconds to wait without receiving a heartbeat before disconnection.
Usage	<code><checkalive-timeout-s>Timeout</checkalive-timeout-s></code>

Parameters Timeout Default is 30 seconds

Example `<checkalive-timeout-s>20</checkalive-timeout-s>`

5.3.2.10 config.gxa_loops.IO.market.parameters.customer-or-firm

Description	Used to specify if the order is for a customer or the firm placing it. Used to populate CustomerOrFirm [204]. Allowable values are: <ul style="list-style-type: none"> • 0 Customer • 1 Firm
Usage	<code><customer-or-firm>CustomerOrFirm</customer-or-firm></code>
Parameters	CustomerOrFirm Tag 204
Example	<code><customer-or-firm>0</customer-or-firm></code>

5.3.2.11 config.gxa_loops.IO.market.parameters.sender-location-id

Description	Identifies the name or location of the device from which the order is being entered. Used to populate SenderLocationId [142] in the standard message header.
Usage	<code><sender-location-id>SenderLocationID</sender-location-id></code>
Parameters	SenderLocationID Tag 142

Example `<sender-location-id>SenderLocationID</sender-location-id>`

5.3.2.12 **config.gxa_loops.IO.market.parameters.application-system-name**

Description System generating the message.
Used to populate ApplicationSystemName [1603] the Logon message.

Usage `<application-system-name>ApplicationSystemName</application-system-name>`

Parameters ApplicationSystemName Tag 1603

Example `<application-system-name>MyTradingApp</application-system-name>`

5.3.2.13 **config.gxa_loops.IO.market.parameters.application-system-version**

Description Version of the system generating the message.
Used to populate ApplicationSystemVersion [1604] in the Logon message.

Usage `<application-system-version>ApplicationSystemVersion</application-system-version>`

Parameters ApplicationSystemVersion Tag 1604

Example `<application-system-version>v1.0</application-system-version>`

5.3.2.14 **config.gxa_loops.IO.market.parameters.application-system-vendor**

Description Vendor of the system generating the message.
Used to populate ApplicationSystemVendor [1605] in the Logon message.

Usage `<application-system-vendor>ApplicationSystemVendor</application-system-vendor>`

Parameters ApplicationSystemVendor Tag 1605

Example `<application-system-vendor>Celoxica</application-system-vendor>`

5.3.2.15 **config.gxa_loops.IO.market.parameters.password**

Description Password given by the CME.
Used to populate RawData [96].

Usage `<password>Password</password>`

Parameters Password Tag 96

Example `<password>Password</password>`

5.3.2.16 **config.gxa_loops.IO.market.parameters.account**

Description Default executing account.
Used to populate Account [1].

Usage `<account>Account</account>`

Parameters Account Tag 1

Example `<account>Account123</account>`

5.3.2.17 **config.gxa_loops.IO.market.parameters.cti-code**

Description Default type of customer or account requesting the order.
Used to populate CtiCode [9702].
Allowable values are:

- 1 Broker/trader trading for own account
- 2 Broker/trader trading for clearing member firm's house account
- 3 Broker/trader trading for the account of another broker/trader present on the trading floor or controlled by such broker/trader
- 4 Broker/trader trading for any other customer's account or for the account of a broker/trader not present on the floor

Usage `<cti-code>CtiCode</cti-code>`

Parameters CtiCode Tag 9702

Example `<cti-code>1</cti-code>`

5.3.2.18 **config.gxa_loops.IO.market.parameters.in-flight-mitigation**

Description Default OFM Override.
Indicates whether the cancel/replace supports in-flight mitigation.
Used to populate OFMOverride [9768].

Usage `<in-flight-mitigation>OFMOverride</in-flight-mitigation>`

Parameters OFMOverride Tag 9768.
Default if "no".

Example `<in-flight-mitigation>N</in-flight-mitigation>`

5.3.2.19 **config.gxa_loops.IO.market.parameters.reset-sequence**

Description CME has sequence numbers that persist through the week. If a recovery file is deleted, there could be a long recovery period. To skip this period `<reset-sequence>` can be set to "yes", knowing this may skip order status events or trade events.

Note:

This parameter is forced to "yes" by GXA Lite if the recovery is enabled and the recovery file is not present. This prevents the user receiving

unwanted updates from previous sessions.

Usage `<reset-sequence>ResetSequence</reset-sequence>`

Parameters ResetSequence Boolean.
Default is "no".

Example `<reset-sequence>yes</reset-sequence>`

5.3.2.20 [config.gxa_loops.IO.market.parameters.forward-session-reject](#)

Description Enables or disables the session level rejects (non-order related rejects) to be forwarded to the client.

Usage `<forward-session-reject>ForwardSessionReject</forward-session-reject>`

Parameters ForwardSessionReject Boolean.
Default is "no".

Example `<forward-session-reject>yes</forward-session-reject>`

5.3.2.21 [config.gxa_loops.IO.market.parameters.compute-fix-time](#)

Description Enables or disables the computation of time-based tags.
When disabled, the time-based tags are set to 01/01/1900.

Usage `<compute-fix-time>ComputeFixTime</compute-fix-time>`

Parameters ComputeFixTime Boolean.
Default is "no".

Example `<compute-fix-time>yes</compute-fix-time>`

5.3.2.22 [config.gxa_loops.IO.market.parameters.fixtrc-file](#)

Description Path to the log file that will be used to store the FIX strings.

Usage `<fixtrc-file>FixLogName</fixtrc-file>`

Parameters FixLogName File name

Example `<fixtrc-file>/tmp/cme-fix.txt</fixtrc-file>`

5.3.3 Symbology

Instruments are identified using the following symbology:

Format	Exchange Field
String	SecurityDesc [107]

5.3.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
-------	-------	-------------

Field	Req'd	Description
Name	Y	Used to set SecurityDesc [107]
NameUInt64	N	Not used
Price	O	Used to set Price [44]
Size	Y	Used to set OrderQty [38]
TimeInForce	Y	Used to set TimeInForce [59]. Allowable TIFs: GXA_TIF_DAY GXA_TIF_IOC GXA_TIF_FOK GXA_TIF_GTC
CommandType	Y	GXA_NEW_ORDER to send a message New Order Single 'D'
Side	Y	Used to set Side [54]. Allowable sides: GXA_SIDE_BUY GXA_SIDE_SELL
Type	Y	Used to set OrdType [40]. Allowable order types: GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_STOP_LOSS GXA_ORDER_TYPE_STOP_LIMIT GXA_ORDER_TYPE_MARKET
TradingSession	N	Not used
GXAOriginalID	N	Not used
AltPrice	O	Used to set StopPx [99]. Mandatory if Type = GXA_ORDER_TYPE_STOP_LIMIT or GXA_ORDER_TYPE_STOP_LOSS
MinQuantity	O	Used to set MinQty [110]
MaxFloor	O	Used to set MaxShow [210]

Field	Req'd	Description
PegType	N	Not used
Account	O	Used to set Account [1]. If not specified, the default account <i>config.gxa_loops.IO.market.parameters.account</i> will be used.
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	Y	Used to set CustomerOrFirm [204]. If not specified, the default CustomerOrFirm indicator <i>config.gxa_loops.IO.market.parameters.customer-or-firm</i> will be used.
OpaqueChar2	N	
MappableID	O	A mappable ID associated with an order
PositionEffect	O	Used to set OpenClose [77]
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used

Field	Req'd	Description
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
HandlInst [21]	1
Symbol [55]	Not provided
Text [58]	Not provided
TransactTime [60]	Populated with the message sending time
NoAllocs [78]	Not provided
AllocAccount [79]	Not provided
SecurityType [167]	Not provided
ExpireDate [432]	Not provided
ManualOrderIndicator [1028]	'N' (Automated)
CustOrderHandlingInst [1031]	Not provided
CtiCode [9702]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.cti-code</i>
GiveUpFirm [9707]	Not provided
CmtaGiveupCD [9708]	Not provided
CorrelationClOrdID [9717]	Set by GXA Lite

5.3.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set SecurityDesc [107]
NameUInt64	N	Not used
Price	O	Used to set Price [44]
Size	Y	Used to set OrderQty [38]
TimeInForce	N	Set by GXA Lite from the original order
CommandType	Y	GXA_MODIFY_ORDER to send a message Order Cancel Replace Request 'G'
Side	N	Set by GXA Lite from the original order
Type	N	Set by GXA Lite from the original order
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be modified
AltPrice	O	Used to set StopPx [99]. Mandatory if Type = GXA_ORDER_TYPE_STOP_LIMIT or GXA_ORDER_TYPE_STOP_LOSS
MinQuantity	N	Not used
MaxFloor	O	Used to set MaxShow [210]
PegType	N	Not used
Account	O	Used to set Account [1] If not specified, the account from the original order will be used. If there was no account in the original order, the default CustomerOrFirm indicator <i>config.gxa_loops.IO.market.parameters.account</i> would be used.

Field	Req'd	Description
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	Y	Used to set CustomerOrFirm [204]. If not specified, the CustomerOrFirm indicator from the original order will be used. If there was no CustomerOrFirm indicator in the original order, the default CustomerOrFirm indicator <i>config.gxa_loops.IO.market.parameters.customer-or-firm</i> would be used.
OpaqueChar2	O	Used to set OFMOverride [9768]. If not specified, the default OFMOverride indicator <i>config.gxa_loops.IO.market.parameters.in-flight-mitigation</i> will be used.
MappableID	N	Not used
PositionEffect	O	Used to set OpenClose [77]
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used

Field	Req'd	Description
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
HandlInst [21]	1
OrigClOrdID [41]	Populated by GXA Lite
Symbol [55]	Not provided
Text [58]	Not provided
TransactTime [60]	Populated with the message sending time
NoAllocs [78]	Not provided
AllocAccount [79]	Not provided
SecurityType [167]	Not provided
ExpireDate [432]	Not provided
ManualOrderIndicator [1028]	'N' (Automated)
CustOrderHandlingInst [1031]	Not provided
CtiCode [9702]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.cti-code</i>
GiveUpFirm [9707]	Not provided
CmtaGiveupCD [9708]	Not provided
CorrelationClOrdID [9717]	Set by GXA Lite

5.3.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameOSI	N	Not used
NameUInt64	N	Not used
Price	N	Not used
Size	N	Not used
TimeInForce	N	Not used
CommandType	Y	GXA_CANCEL_ORDER to send a message Order Cancel Request 'F'
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be cancelled
AltPrice	N	Not used
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	O	Used to set Account [1] If not specified, the account from the original order will be used. If there was no account in the original order, the default account <i>config.gxa_loops.IO.market.parameters.account</i> would be used.

Field	Req'd	Description
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set ClOrdID [11] Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Account [1]	Populated by GXA Lite
HandlInst [21]	1
OrderID [37]	Populated by GXA Lite
OrigClOrdID [41]	Populated by GXA Lite
Side [54]	Populated by GXA Lite
Symbol [55]	Populated by GXA Lite
TransactTime [60]	Populated with the message sending time
SecurityDesc [107]	Populated by GXA Lite
SecurityType [167]	Populated by GXA Lite
ManualOrderIndicator [1028]	'N' (Automated)
CorrelationClOrdID [9717]	Set by GXA Lite

5.3.7 Inbound Data Mapping

The following table lists the specific process of inbound information received from the exchange.

GXAInfo Field	Value
ReasonCode	<p>Populated with ExecRestatementReason [378] from the execution reports for order eliminations (MsgType [35] = 8 and OrdStatus [39] = 4).</p> <p>ExecRestatementReason [378] is used to identify the originator of the order elimination and is not sent on eliminations triggered by a cancel request.</p>

5.3.8 Regulatory Audit Trail

CME, CBOT, NYMEX and COMEX Rule 536.B. ("Globex Order Entry") require all systems that access the Globex platform through the CME iLink® gateway to create an audit trail which contains a complete record of all activity through the connection.

In order to address this regulatory requirement, the CME plugin is able to generate a file containing FIX strings for a GXA Lite session using the configuration node *config.gxa_loops.IO.market.parameters.fixtrc-file* which indicates the path to the log file; the log file is used to generate a CME-compliant audit trace.

5.4 ICE

5.4.1 Source

Documentation ICE FIX OS Reference Manual

Version 3.1.7

Date August 17, 2011

5.4.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
ICE Futures US	FIX 3.1.3	ICE Futures US
ICE Futures Europe	FIX 3.1.3	ICE Futures Europe
ICE Futures Canada	FIX 3.1.3	ICE Futures Canada
ICE OTC	FIX 3.1.3	ICE OTC

The basic structure of a GXA Lite market specific parameters configuration for ICE is shown in the following schema:

```
<parameters>
  <market-counters-shm-key>
  <recv-size>
  <send-size>
  <objects-pool-size>
  <sender-comp-id>
  <sender-sub-id>
  <heartbeat-interval>
  <checkalive-timeout-s>
  <connect-timeout>
```

```
<login-timeout>
<logonid>
<password>
<account>
<cti-code>
<reset-sequence>
<onbehalfof-comp-id>
<onbehalfof-sub-id>
<onbehalfof-sub-id>
<clearing-firm>
<clearing-account>
<traders>
  <!-- List of traders>
  <trader>
    <!-- Trader logon and password>
    <trader-name>
    <trader-password>
  </trader>
</traders>
<security-id>
<security-type>
<referential-file>
</parameters>
```

5.4.2.1 config.libraries.library.entry

Description Sets the initialization library for ICE.

Usage <entry>Entry</entry>

Parameters Entry Must be set to "initICE"

Example <entry>initICE</entry>

5.4.2.2 config.gxa_loops.IO.market.parameters.market-counters-shm-key

Description	Shared memory key for market specifics counters.
Usage	<code><market-counters-shm-key>Key</market-counters-shm-key></code>
Parameters	Key Integer
Example	<code><market-counters-shm-key>2512</market-counters-shm-key></code>

5.4.2.3 config.gxa_loops.IO.market.parameters.recv-size

Description	Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite). Used only when the TCP Offload Engine TOE is not used.
Usage	<code><recv-size>RecvSize</recv-size></code>
Parameters	RecvSize Default is 65536 bytes
Example	<code><recv-size>65536</recv-size></code>

5.4.2.4 config.gxa_loops.IO.market.parameters.send-size

Description	Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).
--------------------	---

Used only when the TCP Offload Engine TOE is not used.

Usage	<code><send-size>SendSize</send-size></code>
Parameters	SendSize Default is 65536 bytes
Example	<code><send-size>65536</send-size></code>

5.4.2.5 config.gxa_loops.IO.market.parameters.objects-pool-size

Description	Number of pre-allocated objects in memory. Each FIX message is defined as a class: the objects created for each class can be pre-allocated in memory to speed up the FIX protocol. GXA Lite asynchronously cleans the pre-allocated objects every second.
Usage	<code><objects-pool-size>ObjectsPoolSize</objects-pool-size></code>
Parameters	ObjectsPoolSize Integer
Example	<code><objects-pool-size>1000</objects-pool-size></code>

5.4.2.6 config.gxa_loops.IO.market.parameters.sender-comp-id

Description	Used to populate SenderCompId [49] in the standard message header.
Usage	<code><sender-comp-id>SenderCompId</sender-comp-id></code>

Parameters SenderCompId Tag 49

Example <sender-comp-id>CompId</sender-comp-id>

5.4.2.7 config.gxa_loops.IO.market.parameters.sender-sub-id

Description Default SenderSubId [50].

Usage <sender-sub-id>SenderSubId</sender-sub-id>

Parameters SenderSubId Tag 50

Example <sender-sub-id>SubId</sender-sub-id>

5.4.2.8 config.gxa_loops.IO.market.parameters.heartbeat-interval

Description Sets then interval, in seconds, for sending or expecting heartbeats.

Usage <heartbeat-interval>Interval</heartbeat-interval>

Parameters Interval Integer

Example <heartbeat-interval>30</heartbeat-interval>

5.4.2.9 config.gxa_loops.IO.market.parameters.checkalive-timeout-s

Description Sets the time in seconds to wait without receiving a heartbeat before

disconnection.

Usage <checkalive-timeout-s>Timeout</checkalive-timeout-s>

Parameters Timeout Default is 30 seconds

Example <checkalive-timeout-s>20</checkalive-timeout-s>

5.4.2.10 config.gxa_loops.IO.market.parameters.connect-timeout

Description Sets the maximum time in seconds allowed to establish a TCP connection with the market.

Usage <connect-timeout>Timeout</connect-timeout>

Parameters Timeout Default is 5 seconds

Example <connect-timeout>10</connect-timeout>

5.4.2.11 config.gxa_loops.IO.market.parameters.login-timeout

Description Sets the maximum time in seconds allowed by the daemon, between login request sent and login response received.

Usage <login-timeout>Timeout</login-timeout>

Parameters Timeout Default is 20 seconds

Example `<login-timeout>10</login-timeout>`

5.4.2.12 **config.gxa_loops.IO.market.parameters.logonid**

Description Username.
Used to populate UserName [553] in message Logon Request 'A'.

Usage `<logonid>Username</logonid>`

Parameters Username Tag 553

Example `<logonid>ABCuser</logonid>`

5.4.2.13 **config.gxa_loops.IO.market.parameters.password**

Description Password given by the CME.
Used to populate RawData [96] in message Logon Request 'A'.

Usage `<password>Password</password>`

Parameters Password Tag 96

Example `<password>Password</password>`

5.4.2.14 **config.gxa_loops.IO.market.parameters.account**

Description Default account.
Used to populate AccountCode [9195].

Usage `<account>Account</account>`

Parameters Account Tag 9195

Example `<account>Account123</account>`

5.4.2.15 **config.gxa_loops.IO.market.parameters.cti-code**

Description Default type of customer or account requesting the order.
Used to populate CtiCode [9208].
Allowable values are:

- 1 Broker/trader trading for own account
- 2 Broker/trader trading for house or prop account
- 3 Broker/trader trading for the account of another broker/trader
- 4 Broker/trader trading for any other customer account that doesn't have direct trading access

Usage `<cti-code>CtiCode</cti-code>`

Parameters CtiCode Tag 9208

Example `<cti-code>1</cti-code>`

5.4.2.16 **config.gxa_loops.IO.market.parameters.reset-sequence**

Description If a recovery file is deleted, there could be a long recovery period. To skip this period `<reset-sequence>` can be set to "yes", knowing this may skip order status events or trade events.

Note:

This parameter is forced to "yes" by GXA Lite if the recovery is enabled and the recovery file is not present. This prevents the user receiving unwanted updates from previous sessions.

Usage `<reset-sequence>ResetSequence</reset-sequence>`

Parameters ResetSequence Boolean.
Default is "no".

Example `<reset-sequence>yes</reset-sequence>`

5.4.2.17 **config.gxa_loops.IO.market.parameters.onbehalf-of-comp-id**

Description Assigned value used to identify firm originating message.
Used to populate OnBehalfOfCompID [115].

Usage `<onbehalf-of-comp-id>OnBehalfOfCompID</onbehalf-of-comp-id>`

Parameters OnBehalfOfCompID Tag 115

Example `<onbehalf-of-comp-id>OnBehalfOfCompID</onbehalf-of-comp-id>`

5.4.2.18 **config.gxa_loops.IO.market.parameters.onbehalf-of-location-id**

Description Assigned value used to identify specific message originator's location.
Used to populate OnBehalfOfLocationID [144].

Usage `<onbehalf-of-location-id>OnBehalfOfLocationID</onbehalf-of-location-id>`

Parameters OnBehalfOfLocationID Tag 144

Example `<onbehalf-of-location-id>OnBehalfOfLocationID</onbehalf-of-location-id>`

5.4.2.19 **config.gxa_loops.IO.market.parameters.onbehalf-of-sub-id**

Description Default specific message originator.
Used to populate OnBehalfOfSubID [116].

Usage `<onbehalf-of-sub-id>OnBehalfOfSubID</onbehalf-of-sub-id>`

Parameters OnBehalfOfSubID Tag 116

Example `<onbehalf-of-sub-id>OnBehalfOfSubID</onbehalf-of-sub-id>`

5.4.2.20 **config.gxa_loops.IO.market.parameters.clearing-firm**

Description Default clearing firm.
Used to populate ClearingFirm [439].

Usage `<clearing-firm>ClearingFirm</clearing-firm>`

Parameters ClearingFirm Tag 439

Example `<clearing-firm>ClearingFirm</clearing-firm>`

5.4.2.21 **config.gxa_loops.IO.market.parameters.clearing-account**

Description Default clearing account.
Used to populate ClearingAccount [440].

Usage `<clearing-account>ClearingAccount</clearing-account>`

Parameters ClearingAccount Tag 440

Example `<clearing-account>ClearingAccount</clearing-account>`

5.4.2.22 **config.gxa_loops.IO.market.parameters.traders**

Description This tag initiates a body for the traders configurations to be made.

Usage `<traders>`

5.4.2.23 **config.gxa_loops.IO.market.parameters.traders.trader**

Description This tag initiates a body for one trader configurations to be made.

Usage `<trader>`

5.4.2.24 **config.gxa_loops.IO.market.parameters.traders.trader.trader-name**

Description Trader name.
Up to 20 traders can be set.
Used to populate UserName [553] in message Trader Login.

Usage `<trader-name>UserName</trader-name>`

Parameters UserName Tag 553

Example `<trader-name>TraderName</trader-name>`

5.4.2.25 **config.gxa_loops.IO.market.parameters.traders.trader.trader-password**

Description Trader password.
Up to 20 traders can be set.
Used to populate RawData [96] in message Trader Login.

Usage `<trader-password>Rawdata</trader-password>`

Parameters Rawdata Tag 96

Example `<trader-password>TraderPassword</trader-password>`

5.4.2.26 config.gxa_loops.IO.market.parameters.security-id

Description Security ID of the product for the security definition request.

on For a list of valid values, please refer to https://www.theice.com/publicdocs/technology/Supported_Market_Types_on_ICE_API.pdf.

A security definition request message will be sent if this configuration tag is present.

Note:

If the configuration parameters security-id and security-type are filled, the ICE plug-in will send a security definition request message for this market. The security definition response is not yet accessible through the API but can be seen in the plugin log.

Usage `<security-id>SecurityID</security-id>`

Parameters SecurityID Tag 48

Example `<security-id>Cocoa</security-id>`

5.4.2.27 config.gxa_loops.IO.market.parameters.security-type

Description Security Type for the security definition request.

Allowable values are:

- **FUT** Futures: one single security definition request will be sent with tag 167 set to 'FUT'
- **OPT** Options: one single security definition request will be sent with tag 167 set to 'OPT'
- **BOTH** Futures and Options: one security definition request will be sent with tag 167 set to 'FUT' and one single security definition request will be sent with tag 167 set to 'OPT'

This parameter is only used for security definition requests.

Usage `<security-type>SecurityType</security-type>`

Parameters SecurityType Tag 167

Example `<security-type>FUT</security-type>`

5.4.2.28 config.gxa_loops.IO.market.parameters.referential-file

Description Specifies the path/filename where the symbol/product descriptions should be stored.

This parameter is only used to store security responses received following security definition requests.

Usage `<referential-file>File</referential-file>`

Parameters File Filename

Example <referential-file>/tmp/ice_ref.txt</referential-file>

5.4.3 Symbology

Instruments are identified using the following symbology:

Format	Exchange Field
String	Symbol [55]

5.4.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set Symbol [55]
NameUInt64	N	Not used
Price	O	Used to set Price [44]
Size	Y	Used to set OrderQty [38]
TimeInForce	Y	Used to set TimeInForce [59]. Allowable TIFs: GXA_TIF_DAY GXA_TIF_IOC GXA_TIF_FOK GXA_TIF_GTC
CommandType	Y	GXA_NEW_ORDER to send a message New Order Single 'D'
Side	Y	Used to set Side [54]. Allowable sides:

Field	Req'd	Description
		GXA_SIDE_BUY GXA_SIDE_SELL
Type	Y	Used to set OrdType [40]. Allowable order types: GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_STOP_LOSS GXA_ORDER_TYPE_STOP_LIMIT GXA_ORDER_TYPE_MARKET
TradingSession	N	Not used
GXAOriginalID	N	Not used
AltPrice	O	Used to set StopPx [99]. Mandatory if Type = GXA_ORDER_TYPE_STOP_LIMIT or GXA_ORDER_TYPE_STOP_LOSS
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	O	Used to set AccountCode [9195]. If not specified, the default account <i>config.gxa_loops.IO.market.parameters.account</i> will be used.
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	O	Used to set OriginatorUserId [9139]. Mandatory if more than one trader is defined in the configuration file using <i>config.gxa_loops.IO.market.parameters.traders</i> . Optional if only one trader is defined in the configuration file using

Field	Req'd	Description
		<i>config.gxa_loops.IO.market.parameters.traders.</i>
OpaqueStr1	O	Used to set OnBehalfOfSubID [116] If not specified, the default OnBehalfOfSubID <i>config.gxa_loops.IO.market.parameters.onbehalfof-sub-id</i> will be used
OpaqueStr2	O	Used to set ClearingAccount [440]. If not specified, the default clearing account <i>config.gxa_loops.IO.market.parameters.clearing-account</i> will be used.
OpaqueInt1	O	Used to set ClearingFirm [439]. If not specified, the default clearing firm <i>config.gxa_loops.IO.market.parameters.clearing-firm</i> will be used.
OpaqueInt2	N	Not used
OpaqueChar1	O	Used to set SecurityType [167]. <u>Allowable values:</u> 'O' Options, SecurityType [167] = 'OPT' 'F' Futures, SecurityType [167] = 'FUT'
OpaqueChar2	N	Not used
MappableID	O	A mappable ID associated with an order
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>

Field	Req'd	Description
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
HandlInst [21]	1
TransactTime [60]	Populated with the message sending time
NoAllocs [78]	Not provided
AllocAccount [79]	Not provided
AllocQty [80]	Not provided
ClientID [109]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.sender-comp-id</i>
OnBehalfOfCompID [115]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.onbehalfof-comp-id</i>
OnBehalfOfLocationID [144]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.onbehalfof-location-id</i>
PutOrCall [201]	Not provided
StrikePrice [202]	Not provided
AllocHandlInst [209]	Not provided
MaxShow [210]	Not provided
MemoField [9121]	Not provided
GiveUpClearingFirm [9196]	Not provided
CustomerAccountRefId [9207]	Not provided

Field	Value
CtiCode [9208]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.cti-code</i>

5.4.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Set by GXA Lite from the original order
NameUInt64	N	Not used
Price	O	Used to set Price [44]
Size	Y	Used to set OrderQty [38]
TimeInForce	N	Set by GXA Lite from the original order
CommandType	Y	GXA_MODIFY_ORDER to send a message Order Cancel Replace Request 'G'
Side	N	Set by GXA Lite from the original order
Type	N	Set by GXA Lite from the original order
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be modified
AltPrice	O	Used to set StopPx [99]. If not specified, the AltPrice from the original order will be used.
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	O	Used to set AccountCode [9195]. If not specified, the default account <i>config.gxa_loops.IO.market.parameters.account</i> will

Field	Req'd	Description
		be used.
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Set by GXA Lite from the original order
OpaqueStr1	N	Set by GXA Lite from the original order
OpaqueStr2	N	Set by GXA Lite from the original order
OpaqueInt1	N	Set by GXA Lite from the original order
OpaqueInt2	N	Not used
OpaqueChar1	N	Set by GXA Lite from the original order
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
HandlInst [21]	1
OrigClOrdID [41]	Populated by GXA Lite
TransactTime [60]	Populated with the message sending time
ListID [66]	Not provided
ClientID [109]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.sender-comp-id</i>
OnBehalfOfCompID [115]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.onbehalfof-comp-id</i>
OnBehalfOfSubID [116]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.onbehalfof-sub-id</i>
OnBehalfOfLocationID [144]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.onbehalfof-location-id</i>
PutOrCall [201]	Not provided
StrikePrice [202]	Not provided
MaxShow [210]	Not provided
MemoField [9121]	Not provided
CustomerAccountRefId [9207]	Not provided
CtiCode [9208]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.cti-code</i>

5.4.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used

Field	Req'd	Description
NameOSI	N	Not used
NameUInt64	N	Not used
Price	N	Not used
Size	N	Not used
TimeInForce	N	Not used
CommandType	Y	GXA_CANCEL_ORDER to send a message Order Cancel Request 'F'
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be cancelled
AltPrice	N	Not used
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Set by GXA Lite from the original order
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used

Field	Req'd	Description
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
OrigClOrdID [41]	Populated by GXA Lite
Symbol [55]	Populated by GXA Lite
TransactTime [60]	Populated with the message sending time
ListID [66]	Not provided
ClientID [109]	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.sender-comp-id</i>

Field	Value
SecurityType [167]	Populated by GXA Lite
PutOrCall [201]	Not provided
StrikePrice [202]	Not provided

5.4.7 Inbound Data Mapping

There are no market specifics.

5.5 LIFFE

5.5.1 Source

Documentation	NYSE Liffe Common Customer Gateway CCG Interface Specification – Binary protocol
Version	1.7a
Date	December 13, 2010

5.5.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
Liffe	CCG Binary	NYSE Euronext Liffe CCG Binary

The basic structure of a GXA Lite market specific parameters configuration for Euronext Liffe is shown in the following schema:

```
<parameters>
  <counter-shm-key>
  <recv-size>
  <send-size>
  <sender-comp-id>
  <sender-sub-id>
  <heartbeat-interval>
  <checkalive-timeout-s>
  <connect-timeout>
  <login-timeout>
  <account>
  <account-code>
  <client-info>
  <order-capacity>
  <clearing-instruction>
```

```
<trade-input-source>
<risk-id>
<start-seq>
<cancel-on-disconnect>
<scale-factor>
</parameters>
```

5.5.2.1 config.libraries.library.entry

Description Sets the initialization library for LIFFE.

Usage <entry>Entry</entry>

Parameters Entry Must be set to “initLiffe”

Example <entry>initLiffe</entry>

5.5.2.2 config.gxa_loops.IO.market.parameters.market-counters-shm-key

Description Shared memory key for market specifics counters.

Usage <market-counters-shm-key>Key</market-counters-shm-key>

Parameters Key Integer

Example <market-counters-shm-key>2512</market-counters-shm-key>

5.5.2.3 config.gxa_loops.IO.market.parameters.recv-size

Description Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite).

Used only when the TCP Offload Engine TOE is not used.

Usage `<recv-size>RecvSize</recv-size>`

Parameters RecvSize Default is 65536 bytes

Example `<recv-size>65536</recv-size>`

5.5.2.4 config.gxa_loops.IO.market.parameters.send-size

Description Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).

Used only when the TCP Offload Engine TOE is not used.

Usage `<send-size>SendSize</send-size>`

Parameters SendSize Default is 65536 bytes

Example `<send-size>65536</send-size>`

5.5.2.5 config.gxa_loops.IO.market.parameters.sender-comp-id

Description Exchange provided sender comp id.

Usage `<sender-comp-id>SenderCompID</sender-comp-id>`

Parameters SenderCompID Sender comp id

Example `<sender-comp-id>SenderCompID</sender-comp-id>`

5.5.2.6 config.gxa_loops.IO.market.parameters.sender-sub-id

Description Exchange provided sender sub id.

Usage `<sender-sub-id>SenderSubID</sender-sub-id>`

Parameters SenderSubID Sender sub id

Example `<sender-sub-id>SenderSubID</sender-sub-id>`

5.5.2.7 config.gxa_loops.IO.market.parameters.heartbeat-interval

Description Heartbeat interval.

Usage `<heartbeat-interval>Interval</heartbeat-interval>`

Parameters Interval Heartbeat interval expressed in seconds.
Default is 10 seconds.

Example `<heartbeat-interval>30</heartbeat-interval>`

5.5.2.8 config.gxa_loops.IO.market.parameters.checkalive-timeout-s

Description The time to wait without receiving a heartbeat before disconnection.

Usage `<checkalive-timeout-s>Timeout</checkalive-timeout-s>`

Parameters Timeout Timeout expressed in seconds.
Default is 20 seconds.

Example `<checkalive-timeout-s>10</checkalive-timeout-s>`

5.5.2.9 config.gxa_loops.IO.market.parameters.connect-timeout

Description Sets the maximum time in seconds allowed to establish a TCP connection with the market.

Usage `<connect-timeout>Timeout</connect-timeout>`

Parameters Timeout Default is 5 seconds

Example `<connect-timeout>10</connect-timeout>`

5.5.2.10 config.gxa_loops.IO.market.parameters.login-timeout

Description Sets the maximum time in seconds allowed by the daemon, between login request sent and login response received.

Usage `<login-timeout>Timeout</login-timeout>`

Parameters Timeout Default is 10 seconds

Example `<login-timeout>10</login-timeout>`

5.5.2.11 config.gxa_loops.IO.market.parameters.account

Description Default Account.
Used to populate Account.

Usage `<account>Account</account>`

Parameters Account Account

Example `<account>Account123</account>`

5.5.2.12 config.gxa_loops.IO.market.parameters.account-code

Description Default type of account.
Used to populate AccountCode.

Allowable values are:

- 'A' Give-up to multiple firms
- 'C' Client
- 'G' Give-up to single firm
- 'H' House

- 'M' Market Maker
- 'N' Non Segregated
- 'S' Segregated
- 'U' Unknown

Usage `<account-code>AccountCode</account-code>`

Parameters AccountCode Account code

Example `<account-code>C</account-code>`

5.5.2.13 [config.gxa_loops.IO.market.parameters.client-info](#)

Description Optional tag, trader's free text.
Used to populate ClientInfo.

Usage `<client-info>ClientInfo</client-info>`

Parameters ClientInfo Free text

Example `<client-info>MyInfo</client-info>`

5.5.2.14 [config.gxa_loops.IO.market.parameters.order-capacity](#)

Description Type of customer trading.
Used to populate CustOrderCapacity.

Allowable values are:

- 1 For own account
- 2 For clearing members house account
- 3 For account of another member present
- 4 For any other customer account

Usage `<order-capacity>CustOrderCapacity</order-capacity>`

Parameters CustOrderCapacity Default is 1 (for own account).

Example `<order-capacity>1</order-capacity>`

5.5.2.15 [config.gxa_loops.IO.market.parameters.clearing-instruction](#)

Description Optional tag, posting code for clearing.
Used to populate ClearingInstruction.

Allowable values are:

- 0 Undefined
- 8 Manual
- 9 Automatic
- 4008 Automatic and Account Authorisation
- 4009 Manual and Account Authorisation
- 4010 Give-up to single firm

Usage `<clearing-instruction>ClearingInstruction</clearing-instruction>`

Parameters ClearingInstruction Default is 9 (Automatic)

Example `<clearing-instruction>9</clearing-instruction>`

5.5.2.16 `config.gxa_loops.IO.market.parameters.trade-input-source`

Description Optional tag, automatic order injection indicator.

Used to populate TradeInputSource.

Allowable values are:

- 'U' Undefined
- 'M' Manual
- 'A' Automated
- 'G' Generated

Usage `<trade-input-source>TradeInputSource</trade-input-source>`

Parameters TradeInputSource Default is A (Automated)

Example `<trade-input-source>A</trade-input-source>`

5.5.2.17 `config.gxa_loops.IO.market.parameters.risk-id`

Description Optional tag, the risk manager's ITM.

Used to populate RiskID.

Usage `<risk-id>RiskID</risk-id>`

Parameters RiskID Risk managers' ITM

Example `<risk-id>Myriskmanager</risk-id>`

5.5.2.18 `config.gxa_loops.IO.market.parameters.start-seq`

Description Default starting sequence number.

Allowable values are:

- -1 To restart from last sent message
- 0 To restart from the beginning of the day or any other number

This takes priority over recovery so even after recovery, it will always restart with the sequence given as parameter. If misused, this could result in missed messages.

Usage `<start-seq>StartSeq</start-seq>`

Parameters StartSeq Sequence number.
Default is -1.

Example `<start-seq>1000</start-seq>`

5.5.2.19 `config.gxa_loops.IO.market.parameters.cancel-on-disconnect`

Description Specifies whether the exchange will cancel all active orders on disconnection or not.

Note:

This feature is not supported by LIFFE yet.

Usage `<cancel-on-disconnect>CancelOnDisconnect</cancel-on-disconnect>`

Parameters `CancelOnDisconnect` Boolean.
Default is "yes".

Example `<cancel-on-disconnect>no</cancel-on-disconnect>`

5.5.2.20 config.gxa_loops.IO.market.parameters.scale-factor

Description When used, the real price (multiplied by 10^8) has to be sent to GXA Lite. GXA Lite will then translate the given prices into ticks before sending them to the exchange.

Usage `<scale-factor>ScaleFactor</scale-factor>`

Parameters `ScaleFactor` Default is 1

Example `<scale-factor>100000000</scale-factor>`

5.5.3 Symbology

Instruments are identified using the following symbology:

Format	Exchange Field
String(15)	SecurityID (AMR)

5.5.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set SecurityID
NameUInt64	N	Not used
Price	Y	Used to set Price
Size	Y	Used to set OrderQty
TimeInForce	Y	Used to set TimeInForce. <u>Allowable TIFs:</u> GXA_TIF_DAY GXA_TIF_FOK GXA_TIF_GTC GXA_TIF_IOC
CommandType	Y	GXA_NEW_ORDER to send a message New Order Single 'D'
Side	Y	Used to set Side. <u>Allowable sides:</u> GXA_SIDE_BUY GXA_SIDE_SELL
Type	Y	Used to set Price Type. <u>Allowable order types:</u> GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_STOP_LOSS GXA_ORDER_TYPE_STOP_LIMIT GXA_ORDER_TYPE_MARKET GXA_ORDER_TYPE_MARKET_ON_OPEN
TradingSession	N	Not used

Field	Req'd	Description
GXAOriginalID	N	Not used
AltPrice	O	Used to set StopPx. Mandatory when Type = GXA_ORDER_TYPE_STOP_LOSS or GXA_ORDER_TYPE_STOP_LIMIT
MinQuantity	O	Used to set MinQty
MaxFloor	N	Not used
PegType	N	Not used
Account	O	Used to set Account. If not specified, the default account <i>config.gxa_loops.IO.market.parameters.account</i> will be used.
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	O	Used to set SecondaryCLOrdID
OpaqueStr2	O	Used to set ClientInfo
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	O	A mappable ID associated with an order
PositionEffect	N	Not used
ClientOrderIDString	N	Not used

Field	Req'd	Description
ClientOrderIDUInt64	O	Used to set CLOrdID. Managed by GXA Lite when not provided
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
ExpireDate	0
TradingSessionID	0
SecurityIDSource	'8' (AMR)
TradeInputSource	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.trade-input-source</i>
OrderOrigin	' ' (space)
ClearingInstruction	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.clearing-instruction</i>
CustOrderCapacity	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.order-capacity</i>
PartyID	Not provided
AccountCode	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.account-code</i>
ClientInfo	Populated with configuration tag

Field	Value
	<i>config.gxa_loops.IO.market.parameters.client-info</i>
PartyRole	Not provided
PostingAction	Not provided

5.5.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Set by GXA Lite from the original order
NameUInt64	N	Not used
Price	Y	Used to set Price
Size	Y	Used to set OrderQty
TimeInForce	N	Not used
CommandType	Y	GXA_MODIFY_ORDER to send a message Order Revision Request 'G'
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be modified
AltPrice	O	Used to set StopPx If not specified, the AltPrice from the original order will be used.
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used

Field	Req'd	Description
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	N	Not used
ClientOrderIDUInt64	O	Used to set ClOrdID. Managed by GXA Lite when not provided
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
OrigClOrdID	Populated by GXA Lite
OrderID	Not provided
ExpireDate	0
SecurityIDSource	'P' (Security Group)

5.5.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameUInt64	N	Not used
Price	N	Not used
Size	N	Not used
TimeInForce	N	Not used
CommandType	Y	GXA_CANCEL_ORDER to send a message Order Cancel Request 'F'
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be cancelled
AltPrice	N	Not used
MinQuantity	N	Not used

Field	Req'd	Description
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	N	Not used
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> .

Field	Req'd	Description
		Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
CIOrdID	Set by GXA Lite
OrigCIOrdID	Set by GXA Lite
OrderID	Not provided
SecurityIDSource	'P' (Security Group)
RiskID	Populated with configuration tag <i>config.gxa_loops.IO.market.parameters.risk-id</i>

5.5.7 Inbound Data Mapping

The following table lists the specific process of inbound information received from the exchange.

GXAInfo Field	Value
OpaqueStr1	Populated with SecondaryCIOrdID from the Execution Report '8'
OpaqueChar1	Populated with OrderCapacity from the Execution Report '8'

5.6 OUCH

5.6.1 Source

Documentation	OUCH 4.1
Version	N/A
Date	April 19, 2012

5.6.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
NASDAQ	OUCH 4.1	NASDAQ
PSX	OUCH 4.1	NASDAQ OMX PSX
BX	OUCH 4.1	NASDAQ OMX BX

The basic structure of a GXA Lite market specific parameters configuration for OUCH is shown in the following schema:

```
<parameters>
  <recv-size>
  <send-size>
  <groupid>
  <logonid>
  <password>
  <display>
  <capacity>
  <minimum-quantity>
  <intermarket-sweep-eligibility>
</parameters>
```

5.6.2.1 config.libraries.library.entry

Description Sets the initialization library for OUCH.

Usage `<entry>Entry</entry>`

Parameters `Entry` Must be set to “initOuch”

Example `<entry>initOuch</entry>`

5.6.2.2 config.gxa_loops.IO.market.parameters.recv-size

Description Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite).
Used only when the TCP Offload Engine TOE is not used.

Usage `<recv-size>RecvSize</recv-size>`

Parameters `RecvSize` Default is 65536 bytes

Example `<recv-size>65536</recv-size>`

5.6.2.3 config.gxa_loops.IO.market.parameters.send-size

Description Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).
Used only when the TCP Offload Engine TOE is not used.

Usage `<send-size>SendSize</send-size>`

Parameters SendSize Default is 65536 bytes

Example <send-size>65536</send-size>

5.6.2.4 config.gxa_loops.IO.market.parameters.groupid

Description Firm.

Usage <groupid>Firm</groupid>

Parameters Firm Firm

Example <groupid>ABCFirm</groupid>

5.6.2.5 config.gxa_loops.IO.market.parameters.logonid

Description The SoupBinTCP username.

Usage <logonid>UserId</logonid>

Parameters UserId Logon id

Example <logonid>id</logonid>

5.6.2.6 config.gxa_loops.IO.market.parameters.password

Description The SoupBinTCP password.

Usage <password>Password</password>

Parameters Password Password

Example <password>password</password>

5.6.2.7 config.gxa_loops.IO.market.parameters.display

Description Specifies the default Display field for the new order and replace order messages.

Allowable values are:

- 'A' Attributable – price to display
- 'Y' Anonymous – price to comply (default value)
- 'N' Hidden
- 'P' post only
- 'I' Imbalance only
- 'M' Mid-point peg
- 'W' Mid-point peg post only
- 'L' post only and attributable – price to display

Usage <display>Display</display>

Parameters Display Display field

Example `<display>Y</display>`

5.6.2.8 config.gxa_loops.IO.market.parameters.capacity

Description Default order capacity.

Allowable values are:

- 'A' Agency (default value)
- 'P' Principal
- 'R' Riskless

Usage `<capacity>Capacity</capacity>`

Parameters Capacity Order capacity

Example `<capacity>P</capacity>`

5.6.2.9 config.gxa_loops.IO.market.parameters.minimum-quantity

Description The default minimum quantity.

Usage `<minimum-quantity>MinQty</minimum-quantity>`

Parameters MinQty Minimum quantity

Example `<minimum-quantity>10</minimum-quantity>`

5.6.2.10 config.gxa_loops.IO.market.parameters.intermarket-sweep-eligibility

Description The default intermarket sweep eligibility for orders.

Allowable values are:

- 'Y' ISO Eligible
- 'N' Not ISO eligible (default value)

Usage `<intermarket-sweep-eligibility>ISO</intermarket-sweep-eligibility>`

Parameters ISO Intermarket sweep eligibility

Example `<intermarket-sweep-eligibility>Y</intermarket-sweep-eligibility>`

5.6.3 Symbology

Instruments are identified using the following symbology:

Format	Exchange Field
String(8)	Stock

5.6.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set Stock
NameUInt64	N	Not used

Field	Req'd	Description
Price	Y	Used to set Price
Size	Y	Used to set Shares
TimeInForce	Y	Used to set Time In Force. <u>Allowable TIFs:</u> GXA_TIF_DAY GXA_TIF_IOC GXA_TIF_FOK
CommandType	Y	GXA_NEW_ORDER to send a message Order Entry 'O'
Side	Y	Used to set Side. <u>Allowable sides:</u> GXA_SIDE_BUY GXA_SIDE_SELL GXA_SIDE_SELL_SHORT GXA_SIDE_SELL_SHORT_EXEMPT
Type	Y	Used to set Price Type. <u>Allowable order types:</u> GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_POST_ONLY GXA_ORDER_TYPE_PEGGED
TradingSession	N	Not used
GXAOriginalID	N	Not used
AltPrice	N	Not used
MinQuantity	O	Used to set Minimum Quantity. If not specified, the default Minimum Quantity <i>config.gxa_loops.IO.market.parameters.minimum-quantity</i> will be used.
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used

Field	Req'd	Description
Rule80A	O	Used to set Capacity. If not specified, the default Capacity <i>config.gxa_loops.IO.market.parameters.capacity</i> will be used.
ISO	O	Used to set Intermarket Sweep Eligibility. If not specified, the default ISO flag <i>config.gxa_loops.IO.market.parameters.intermarket-sweep-eligibility</i> will be used.
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	O	Used to set Display
OpaqueChar2	N	Not used
MappableID	O	A mappable ID associated with an order
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set Order Token. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive

Field	Req'd	Description
		the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Firm	Set using <i>config.gxa_loops.IO.market.parameters.groupid</i>
Display	Set using <i>config.gxa_loops.IO.market.parameters.display</i> if not set by OpaqueChar1 'P' if Type = GXA_ORDER_TYPE_POST_ONLY regardless the OpaqueChar1 value
Cross Type	'N' (no cross)
Minimum Quantity	Set using <i>config.gxa_loops.IO.market.parameters.minimum-quantity</i> if not set by MinQuantity
Capacity	Set using <i>config.gxa_loops.IO.market.parameters.capacity</i> if not set by Rule80A

5.6.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set Stock
NameUInt64	N	Not used
Price	Y	Used to set Price
Size	Y	Used to set Shares
TimeInForce	Y	Used to set Time In Force.

Field	Req'd	Description
		<u>Allowable TIFs:</u> GXA_TIF_DAY GXA_TIF_IOC GXA_TIF_FOK
CommandType	Y	GXA_MODIFY_ORDER to send a message Order Replace 'U'
Side	Y	Used to set Side. <u>Allowable sides:</u> GXA_SIDE_BUY GXA_SIDE_SELL GXA_SIDE_SELL_SHORT GXA_SIDE_SELL_SHORT_EXEMPT
Type	Y	Used to set Price Type. <u>Allowable order types:</u> GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_POST_ONLY GXA_ORDER_TYPE_PEGGED
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be modified
AltPrice	N	Not used
MinQuantity	O	Used to set Minimum Quantity. If not specified, the default Minimum Quantity <i>config.gxa_loops.IO.market.parameters.minimum-quantity</i> will be used.
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	O	Used to set Intermarket Sweep Eligibility. If not specified, the default ISO flag <i>config.gxa_loops.IO.market.parameters.intermarket-sweep-eligibility</i> will be used.

Field	Req'd	Description
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set Replacement Order Token. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Existing Order Token	Set by GXA Lite
Firm	Set using <i>config.gxa_loops.IO.market.parameters.groupid</i>
Display	Set using <i>config.gxa_loops.IO.market.parameters.display</i> if not set by OpaqueChar1 'P' if Type = GXA_ORDER_TYPE_POST_ONLY regardless the OpaqueChar1 value
Minimum Quantity	Set using <i>config.gxa_loops.IO.market.parameters.minimum-quantity</i> if not set by MinQuantity

5.6.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameUInt64	N	Not used
Price	N	Not used
Size	N	Not used
TimeInForce	N	Not used
CommandType	Y	GXA_CANCEL_ORDER to send a message Order Cancel 'X'
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be cancelled
AltPrice	N	Not used
MinQuantity	N	Not used
MaxFloor	N	Not used

Field	Req'd	Description
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	N	Not used
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	N	Not used
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> .

Field	Req'd	Description
		Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
Order Token	Set by GXA Lite
Shares	0: partial cancel is not supported

5.6.7 Inbound Data Mapping

There are no market specifics.

5.7 PHLX

5.7.1 Source

Documentation PHLX XL Firm FIX Engine

Version 2.5b

Date April 04, 2012

5.7.2 Configuration Settings

This section describes the configuration specific settings for the following markets:

Market	Protocol	Description
PHLX	PHLX XL Firm Fix	NASDAQ OMX PHLX

The basic structure of a GXA Lite market specific parameters configuration for PHLX is shown in the following schema:

```
<parameters>
  <market-counters-shm-key>
  <recv-size>
  <send-size>
  <objects-pool-size>
  <sender-comp-id>
  <sender-sub-id>
  <start-seq>
  <heartbeat-interval>
  <checkalive-timeout-s>
  <position-effect>
  <customer-or-firm>
</parameters>
```

5.7.2.1 config.libraries.library.entry

Description Sets the initialization library for PHLX.

Usage `<entry>Entry</entry>`

Parameters Entry Must be set to "initPhlx"

Example `<entry>initPhlx</entry>`

5.7.2.2 config.gxa_loops.IO.market.parameters.market-counters-shm-key

Description Shared memory key for market specifics counters.

Usage `<market-counters-shm-key>Key</market-counters-shm-key>`

Parameters Key Integer

Example `<market-counters-shm-key>2512</market-counters-shm-key>`

5.7.2.3 config.gxa_loops.IO.market.parameters.recv-size

Description Sets the TCP window size in bytes for the inbound communication (from the exchange to GXA Lite).

Used only when the TCP Offload Engine TOE is not used.

Usage `<recv-size>RecvSize</recv-size>`

Parameters `RecvSize` Default is 65536 bytes

Example `<recv-size>65536</recv-size>`

5.7.2.4 `config.gxa_loops.IO.market.parameters.send-size`

Description Sets the TCP window size in bytes for the outbound communication (from GXA Lite to the exchange).

Used only when the TCP Offload Engine TOE is not used.

Usage `<send-size>SendSize</send-size>`

Parameters `SendSize` Default is 65536 bytes

Example `<send-size>65536</send-size>`

5.7.2.5 `config.gxa_loops.IO.market.parameters.objects-pool-size`

Description Number of pre-allocated objects in memory.

Each FIX message is defined as a class: the objects created for each class can be pre-allocated in memory to speed up the FIX protocol.

GXA Lite asynchronously cleans the pre-allocated objects every second.

Usage `<objects-pool-size>ObjectsPoolSize</objects-pool-size>`

Parameters `ObjectsPoolSize` Integer

Example `<objects-pool-size>1000</objects-pool-size>`

5.7.2.6 `config.gxa_loops.IO.market.parameters.sender-comp-id`

Description Used to populate SenderCompId [49] in the standard message header.

Usage `<sender-comp-id>SenderCompId</sender-comp-id>`

Parameters `SenderCompId` Tag 49

Example `<sender-comp-id>CompId</sender-comp-id>`

5.7.2.7 `config.gxa_loops.IO.market.parameters.sender-sub-id`

Description Used to populate SenderSubId [50] in the standard message header.

Usage `<sender-sub-id>SenderSubId</sender-sub-id>`

Parameters `SenderSubId` Tag 50

Example `<sender-sub-id>SubId</sender-sub-id>`

5.7.2.8 `config.gxa_loops.IO.market.parameters.heartbeat-interval`

Description Sets then interval, in seconds, for sending or expecting heartbeats.

Usage `<heartbeat-interval>Interval</heartbeat-interval>`

Parameters Interval Integer

Example `<heartbeat-interval>30</heartbeat-interval>`

5.7.2.9 [config.gxa_loops.IO.market.parameters.checkalive-timeout-s](#)

Description Sets the time in seconds to wait without receiving a heartbeat before disconnection.

Usage `<checkalive-timeout-s>Timeout</checkalive-timeout-s>`

Parameters Timeout Default is 30 seconds

Example `<checkalive-timeout-s>20</checkalive-timeout-s>`

5.7.2.10 [config.gxa_loops.IO.market.parameters.position-effect](#)

Description Default position effect.
Used to populate OpenClose [77].
Allowable values are:

- 'O' Open
- 'C' Close

Usage `<position-effect>PositionEffect</position-effect>`

Parameters PositionEffect Tag 77

Example `<position-effect>C</position-effect>`

5.7.2.11 [config.gxa_loops.IO.market.parameters.customer-or-firm](#)

Description Default customer or firm indicator.
Used to populate CustomerOrFirm [204].
Allowable values are:

- '0' Customer
- '1' Firm
- '2' Broker / Dealer
- '4' Non-PHLX Registered Market Maker
- '5' On-Floor Specialist, SQT or ROT
- '6' Remote Specialist or RSQT
- '8' Professional Customer

Usage `<customer-or-firm>CustomerOrFirm</customer-or-firm>`

Parameters CustomerOrFirm Tag 204

Example `<customer-or-firm>0</customer-or-firm>`

5.7.2.12 [config.gxa_loops.IO.market.parameters.start-seq](#)

Description Default starting sequence number.

Note:

The PHLX recovery mechanism is not supported yet.

Usage <start-seq>StartSeq</start-seq>

Parameters StartSeq Sequence number

Example <start-seq>100000</start-seq>

5.7.3 Symbology

Instruments are identified using the following symbology:

Format	Exchange Field
NameOSI ²	Symbol [55], MaturityMonthYear [200], MaturityDay [205], MaturityDate [541], PutOrCall [201], StrikePrice [202]

5.7.4 New Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set Symbol [55], MaturityMonthYear [200], MaturityDay [205], MaturityDate [541], PutOrCall [201], StrikePrice [202] NameOSI must be set in the presence map.
NameUInt64	N	Not used

² See section 4.2.3.2 for more details

Field	Req'd	Description
Price	O	Used to set Price [44]
Size	Y	Used to set OrderQty [38]
TimeInForce	Y	Used to set TimeInForce [59]. <u>Allowable TIFs:</u> GXA_TIF_DAY (to be used for at the opening orders) GXA_TIF_IOC GXA_TIF_GTC
CommandType	Y	GXA_NEW_ORDER to send a message New Order Single 'D'
Side	Y	Used to set Side [54]. <u>Allowable sides:</u> GXA_SIDE_BUY GXA_SIDE_SELL
Type	Y	Used to set OrdType [40]. <u>Allowable order types:</u> GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_MARKET GXA_ORDER_TYPE_STOP_LIMIT GXA_ORDER_TYPE_STOP_LOSS GXA_ORDER_TYPE_LIMIT_ON_OPEN (for future use) <ul style="list-style-type: none"> • OrdType [40] = 2 (Limit) • TimeInForce [59] = 2 (At the opening) GXA_ORDER_TYPE_MARKET_ON_OPEN <ul style="list-style-type: none"> • OrdType [40] = 1 (Market) • TimeInForce [59] = 2 (At the opening)
TradingSession	N	Not used
GXAOriginalID	N	Not used
AltPrice	O	Used to set StopPx [99]. Mandatory if Type = GXA_ORDER_TYPE_STOP_LIMIT or GXA_ORDER_TYPE_STOP_LOSS
MinQuantity	N	Not used

Field	Req'd	Description
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	O	Used to set SecurityExchange [207]
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	O	Used to set ExecInst [18]
OpaqueChar2	O	Used to set CustomerOrFirm [204]. If not specified, the default CustomerOrFirm indicator <i>config.gxa_loops.IO.market.parameters.customer-or-firm</i> will be used.
MappableID	O	A mappable ID associated with an order
PositionEffect	Y	Used to set OpenClose [77]. If not specified, the default OpenClose indicator <i>config.gxa_loops.IO.market.parameters.position-effect</i> will be used.
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used

Field	Req'd	Description
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
SecurityType [167]	'OPT'
TransactTime [60]	Not set
HandlInst [21]	2
ExecBroker [76]	Not set
CoveredOrUncovered [203]	Not set
ClearingFirm [439]	Not set
ClearingAccount [440]	Not set
ClientID [109]	Not set
AllocAccount [79]	Not set
Text [58]	Not set
AuctionType [9373]	Not set
AuctionID [9370]	Not set

5.7.5 Replace Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	Y	Used to set Symbol [55], MaturityMonthYear [200], MaturityDay [205], MaturityDate [541], PutOrCall [201], StrikePrice [202] NameOSI must be set in the presence map.
NameUInt64	N	Not used
Price	O	Used to set Price [44]
Size	Y	Used to set OrderQty [38]
TimeInForce	Y	Used to set Time In Force [59]. <u>Allowable TIFs:</u> GXA_TIF_DAY (to be used for at the opening orders) GXA_TIF_IOC GXA_TIF_GTC
CommandType	Y	GXA_MODIFY_ORDER to send a message Cancel / Replace Request 'G'
Side	Y	Used to set Side [54]. <u>Allowable sides:</u> GXA_SIDE_BUY GXA_SIDE_SELL
Type	Y	Used to set OrdType [40]. <u>Allowable order types:</u> GXA_ORDER_TYPE_LIMIT GXA_ORDER_TYPE_MARKET GXA_ORDER_TYPE_STOP_LIMIT GXA_ORDER_TYPE_STOP_LOSS GXA_ORDER_TYPE_LIMIT_ON_OPEN (for future use) <ul style="list-style-type: none"> OrdType [40] = 2 (Limit) TimeInForce [59] = 2 (At the opening) GXA_ORDER_TYPE_MARKET_ON_OPEN

Field	Req'd	Description
		<ul style="list-style-type: none"> OrdType [40] = 1 (Market) TimeInForce [59] = 2 (At the opening)
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be modified
AltPrice	O	Used to set StopPx [99]. Mandatory if Type = GXA_ORDER_TYPE_STOP_LIMIT or GXA_ORDER_TYPE_STOP_LOSS
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	O	Used to set SecurityExchange [207]
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	O	Used to set ExecInst [18]
OpaqueChar2	O	Used to set CustomerOrFirm [204]. If not specified, the default CustomerOrFirm indicator <i>config.gxa_loops.IO.market.parameters.customer-or-firm</i> will be used.
MappableID	N	Not used

Field	Req'd	Description
PositionEffect	Y	Used to set OpenClose [77]. If not specified, the default OpenClose indicator <i>config.gxa_loops.IO.market.parameters.position-effect</i> will be used.
ClientOrderIDString	O	Used to set ClOrdID [11]. Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
OrigClOrdID [41]	Set by GXA Lite
SecurityType [167]	'OPT'
TransactTime [60]	Not set
HandlInst [21]	2
ExecBroker [76]	Not set
CoveredOrUncovered [203]	Not set

Field	Value
ClearingFirm [439]	Not set
ClearingAccount [440]	Not set
ClientID [109]	Not set
AllocAccount [79]	Not set
Text [58]	Not set
NWT OrdType [9371]	Not set
NWT Price [9372]	Not set

5.7.6 Cancel Order Message Mapping

The GXANewCommand structure to be pushed to the market is as follows:

Field	Req'd	Description
Name	N	Not used
NameOSI	N	Not used
NameUInt64	N	Not used
Price	N	Not used
Size	N	Not used
TimeInForce	N	Not used
CommandType	Y	GXA_CANCEL_ORDER to send a message Order Cancel Request 'F'
Side	N	Not used
Type	N	Not used
TradingSession	N	Not used
GXAOriginalID	Y	GXAID to be cancelled
AltPrice	N	Not used

Field	Req'd	Description
MinQuantity	N	Not used
MaxFloor	N	Not used
PegType	N	Not used
Account	N	Not used
Rule80A	N	Not used
ISO	N	Not used
SenderSubID	N	Not used
TraderID	N	Not used
OpaqueStr1	N	Not used
OpaqueStr2	N	Not used
OpaqueInt1	N	Not used
OpaqueInt2	N	Not used
OpaqueChar1	N	Not used
OpaqueChar2	O	Used to set CustomerOrFirm [204]. If not specified, the default CustomerOrFirm indicator <i>config.gxa_loops.IO.market.parameters.customer-or-firm</i> will be used.
MappableID	N	Not used
PositionEffect	N	Not used
ClientOrderIDString	O	Used to set ClOrdID [11] Managed by GXA Lite when not provided
ClientOrderIDUInt64	N	Not used
ExchangeOrderIDString	N	Not used
ExchangeOrderIDUInt64	N	Not used

Field	Req'd	Description
TotalTradedVolume	N	Not used
TargetMarketID	Y	Market socket ID set using configuration tag <i>config.gxa_loops.IO.market.id</i>
TargetResponseID	Y	Output queue ID if the user does not want to receive the market response in the default output queue set using configuration tag <i>config.gxa_loops.IO.market.info_queue_id</i> . Otherwise set to 0.

The following table lists the exchange fields that are not provided using the GXANewCommand structure:

Field	Value
OrigClOrdID [41]	Set by GXA Lite
OrderQty [38]	Set by GXA Lite
MaturityDate [541]	Not set
CustomerOrFirm [204]	Not set
Side [54]	Set by GXA Lite
Symbol [55]	Set by GXA Lite
MaturityMonthYear [200]	Set by GXA Lite
MaturityDay [205]	Set by GXA Lite
PutOrCall [201]	Set by GXA Lite
StrikePrice [202]	Set by GXA Lite
TransactTime [60]	Not set
SecurityType [167]	'OPT'
NoLegs [555]	Not set
LegRefID [654]	Not set

Field	Value
LegSymbol [600]	Not set
LegCFIcode [608]	Not set
LegMaturityMonthYear [601]	Not set
LegMaturityDate [611]	Not set
LegStrikePrice [612]	Not set
LegSide [624]	Not set
LegRatioQty [623]	Not set
LegPositionEffect [564]	Not set

5.7.7 Inbound Data Mapping

There are no market specifics.