

Related Documents:	2
Other Information:	2
This Document:	2
Revision:	2
Similar MCOOrders Products:	2
1. Overview:	3
1.0 Program Operation:	3
1.1 ASXOrders:	4
1.2 Order Cancellation:	4
1.3 SQL Database:	4
1.4 Web Client:	4
2. Daily Cycle:	4
3. Installation/Configuration:	4
3.1 Installation:	5
3.1.1 ASXOrders:	5
3.1.2 MCclickASX:	5
3.2 ASXOrders Configuration:	6
3.2.1 MCclickASX Connection Parameters:	6
3.2.2 ASX Trade Logon Parameters:	6
3.2.3 Broker List Parameter:	7
3.2.4 Order Types to Query:	7
3.2.5 Trades Data:	7
3.2.6 TCP/IP Feed Parameters:	8
3.2.7 Command Client Parameters – Order Cancellation:	8
3.2.8 SQL Database Parameters	8
3.2.9 Logging Parameters:	9
3.2.10 Daily Cycle Parameters:	9
3.2.11 Other parameters:	10
3.3 MCclickASX Configuration:	10
3.3.1 Client (ASXOrders) Connection Parameters:	10
3.3.2 ASX Trade Connection Parameters:	10
3.3.3 ASX Trade Connection Options:	11
3.3.4 Logging Options:	11
3.3.5 Other Configuration Options:	12
4. Password Changing:	12
5. Recovery Strategy:	13
6. Command Clients – Order Cancellation:	14
6.1 Order Cancellation Overview:	14
7. SQL Database:	14
7.1 SQL Database Tables:	14
7.1.1 Table - system	14
7.1.2 Table – system_state	14
7.1.3 Table – asx_orders	14
7.2 SQL Database Parameters:	15
7.3 npgsql files:	15
7.4 SQL Script files:	16
8. PHP – Web Client:	16
9. Fields Mappings ASXOrders <-> ASX Trade:	16

Related Documents:

RJE:-

- ASXOrders Design.pdf
- ASXOrders Technical Overview.pdf

ASX:-

- ASX Trade Open Interface Manual - asx_IO_.4.pdf
- OMexExternal_API_ASX_va522.pdf

These are the most relevant ASX documents but there are others.

Other Information:

Please consult the RJE Web Site www.rje.com.au for the latest information on RJE products.

This Document:

ASXOrders – ASX Trade Orders Capture – details all issues relating to installation and operation of ASXOrders.

Revision:

27/02/2013 – Initial Document – ASXOrders Rev 01_01.

Similar MCOOrders Products:

Similar MCOOrders products exist for other exchanges:- ASXTrade, ASX24, Chix Aust, SGX, TSE, OSE, SBI, Chix Japan contact RJE for more details.

1. Overview:

1.0 Program Operation:

There is a separate MCclickASX component, similar to MCTradesAT.

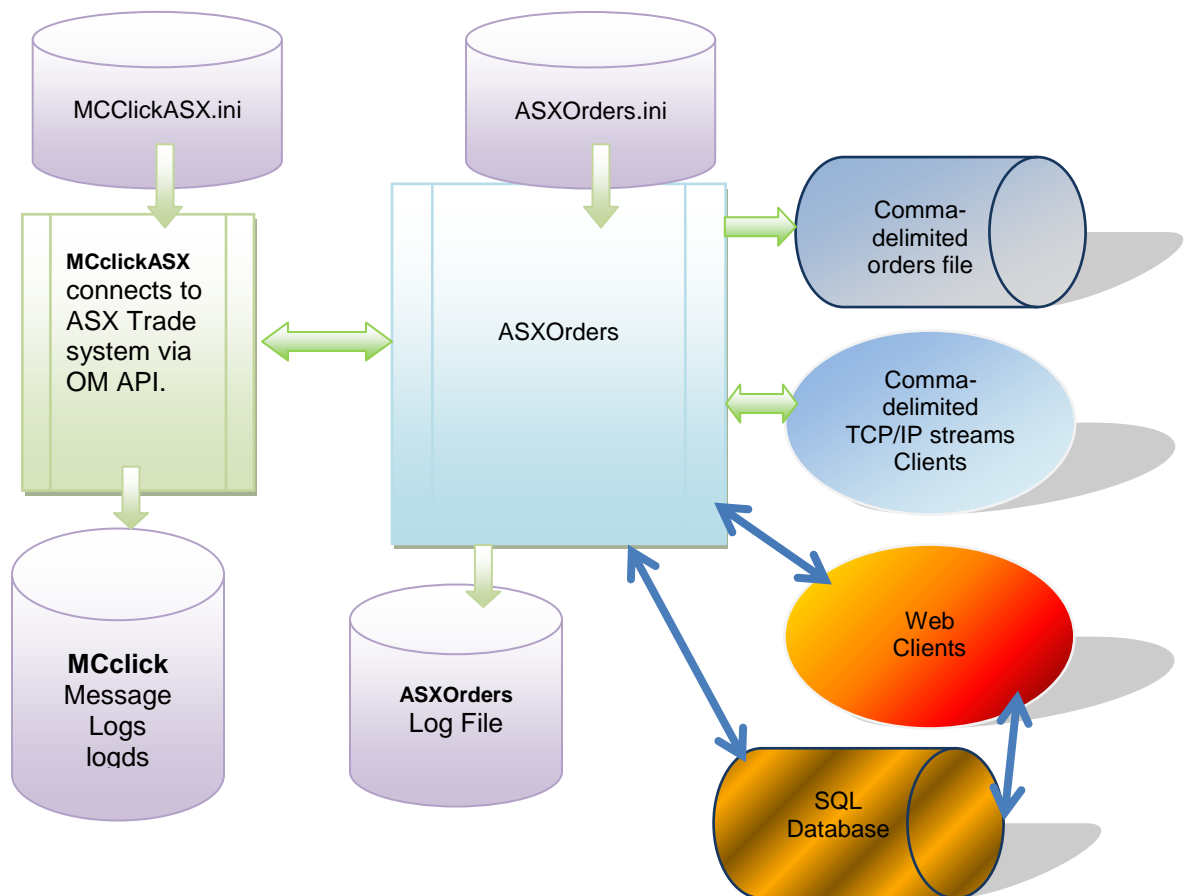


Figure 1. The ASXOrders Production System

1.1 ASXOrders:

The ASXOrders Program:-

- Connects to the ASX Trade system via MCclickASX
- Updates the SQL database with details of current orders.
- Provides TCP/IP stream feeds of orders and transactions.

The configuration settings for the system are found in ASXOrders.ini

1.2 Order Cancellation:

ASXOrders does not support cancellation of orders.

1.3 SQL Database:

An optional SQL database can be configured to store orders and related information. More details can be found in :- [7. SQL Database:](#)

1.4 Web Client:

A PHP based web client application can be used to view orders.

See :- [8. PHP – Web Client:](#)

2. Daily Cycle:

ASXOrders can be run for multiple days; it shuts down and wakes up at a certain scheduled time each day.

Refer:- [3.2.6 Daily Cycle Parameters:](#)

WAKE_TIME = time when program wakes up each morning, SHUT_TIME = time when the program shutdown (hibernation) occurs. This area functions as per existing RJE products.

3. Installation/Configuration:

3.1 Installation:

3.1.1 ASXOrders:

Simply install ASXOrders as follows :-

<Install Directory> :- ASXOrders.exe, ASXOrders.ini
<Install Directory>:-au.com.mcx.dll, Npgsql.dll,
<Install Directory>/logs :- make a subdirectory for log files.

To run the program simply run ASXOrders.exe, provided the configuration in the .ini file is correct no other information is needed.

You must set the following parameters correctly:-

- 3.2.1 MCclickASX Connection Parameters
- 3.2.2 ASX Trade Logon Parameters
- 3.2.3 Broker List Parameter.

Note: If you wish to run the program without a GUI refer:-[3.2.11 Other parameters:](#)

3.1.2 MCclickASX:

Typically, this would be in a separate directory but could be on the same machine.

Simply copy all files as follows;

<Install Directory> :- MCclickASX.exe, MCclickASX.ini
<Install Directory>:- libeay32.dll, oapimtdll.dll, ssleay32.dll, zlib1.dll
<Install Directory>/logs :- make a subdirectory for log files.
<Install Directory>:- plogASX.exe – utility for converting log file extracts to text.

3.2 ASXOrders Configuration:

All configuration parameters are stored in ASXOrders.ini

3.2.1 MCclickASX Connection Parameters:

MC_SERVER_HOST=myhost
MC_SERVER_PORT=7001

MC_SERVER_HOST = Name of Server running MCclickASX.
e.g. FIX_SERVER_HOST= myhost

MC_SERVER_PORT = Port to connect for MCclickASX client connections.
e.g. FIX_SERVER_PORT=7001

This corresponds to a port setting in MCclickASX.ini:-
CLIENTS_PORT =7001

Note: ASXOrders always connects to the ASX Trade system via MCclickASX.

3.2.2 ASX Trade Logon Parameters:

ASX_USER_ID= ASX Trade Logon user supplied by the ASX.
e.g. ASX_USER_ID=Z1389A1

ASX_PASSWORD= ASX Trade user password initially supplied by the ASX.
e.g. ASX_PASSWORD=ABCDEF1205G463HIJKLM

ASX Trade user passwords can expire and currently must be changed every 90 days. ASXOrders will detect that a password has expired and will automatically change the password to one that is compliant with the ASX rules. The .ini file is updated with the new password; this is a full automated process and should not need manual intervention.

The following parameters can be set for automatic password handling:-

AUTO_PASSWORD_BASE=ABCDEF

CHANGE_PASSWORD_DAYS=10

CHANGE_PASSWORD_DAYS=n – change the password ‘n days’ before expiry.

3.2.3 Broker List Parameter:

You must define a list of broker numbers to be monitored by the ASXOrders application in this parameter. This setting ensures the ASXOrders program does MQn orders data queries for each broker in the list.

As well, the ASX Trade system must be configured to correctly route orders and trades data for those brokers to the Drop Copy user id.

This should always be tested, both querying existing orders via Broker List at start up and receiving details for any new orders via BO5 broadcasts.

```
*****  
* BROKER_LIST Data Processed for brokers in this list*  
*****  
BROKER_LIST=AB389,XX123
```

3.2.4 Order Types to Query:

* (Normal ACTIVE orders always processed)

Normal Active orders are always queried, it is possible to configure the system so it does not query other types of orders on start up. However, generally you should get all types of orders data as per the settings below.

```
INACTIVE_ORDERS=YES
```

3.2.5 Trades Data:

Some clients are already getting trade data from the ASXClear environment via MCTradesAT.

Initially, we recommend turning off the gathering of trade data as per the setting below:-
TRADES=NO

This is because there will be an issue with the dated .trades file if the program is stopped/restarted between 12pm and 2am ASX time. (We will address this in the next version.)

Some more development and testing may be required to adequately ensure we can robustly handle all aspects of 24x7 operation while gathering trades.

Once this is resolved ASXOrders should be able to provide a reliable feed of T+1 trades data.

Some consideration was given to matching orders and trades to determine if an order has traded out while the program is offline. However, this issue is more complex than is first apparent, detailed consideration is required before attempting this type of matching.

3.2.6 TCP/IP Feed Parameters:

This is the TCP/IP port that applications can connect to receive a feed of orders data.

ORDERS_PORT = TCP/IP port for all Orders.
e.g. ORDERS_PORT=12008

This is the primary feed produced by this application, but it also produces a feed of BO5 transactions and trades, this can be configured as follows:-

e.g. ORD_TRANS_PORT=12012
e.g. TRADES_PORT=12014

3.2.7 Command Client Parameters – Order Cancellation.

e.g. COMMAND_PORT= TCP port command clients must connect to issue cancel requests.
COMMAND_PORT=12010

3.2.8 SQL Database Parameters

The use of an SQL Database is optional but the Web Client facility will not work properly if it is not enabled.

All SQL Database access occurs in a separate thread it should not affect the performance of the rest of the ASXOrders application.

If we detect that database updating cannot keep up with the rate the ASX is sending data, the program can be enhanced with multiple database updating threads.

SQL_DATABASE_NAME=Name of the database to access. The presence of this parameter turns on database processing. All tables and functions mentioned in SQL_DATABASE_NAME=webdb

SQL_DATABASE_SERVER=The machine which is the PostgreSQL database server.
SQL_DATABASE_SERVER=rjlinuxlap

SQL_DATABASE_PORT=Port for the PostgreSQL database.
SQL_DATABASE_PORT=5432

SQL_USER_ID=PostgreSQL database user.
SQL_USER_ID=sfe

SQL_PASSWORD= PostgreSQL database user password*
SQL_PASSWORD=rjexxxxx

* There may be a better way of controlling access to the PostgreSQL database. We have chosen the user/password model to simplify the initial development.

3.2.9 Logging Parameters:

APP_LOG_FILE = file base for application log, a new log is taken each run; the application log includes the current date and time.

e.g. APP_LOG_FILE= ASXOrders
filename= ASXOrders_20080429_150113.log.

APP_LOG_DIRECTORY=Directory where the application log is stored.
APP_LOG_DIRECTORY=logs

LOGGING_LEVEL= Set the level of application message logging; can turn on additional diagnostic messages.
LOGGING_LEVEL=9

3.2.10 Daily Cycle Parameters:

Refer:- [2. Daily Cycle:](#)

WAKE_TIME = time when program wakes up each morning (hour:min), default 07:00.
e.g. WAKE_TIME=07:30

SHUT_TIME = time when the program shutdown (hibernation) occurs (hour:min) default 23:30.

e.g. SHUT_TIME=21:00

3.2.11 Other parameters:

NO_GUI=YES – Specify this value to run without a GUI, e.g. as a Windows NT service.

3.3 MCclickASX Configuration:

These are similar to parameter settings for similar components e.g. MCclickASX.ini, MC SecurASX.ini.

3.3.1 Client (ASXOrders) Connection Parameters:

As mentioned earlier ASXOrders must make a TCP/IP connection to MCclickASX which in turn talks to the ASX Trade system via OM API over a TCP/IP transport.

The corresponding settings for ASXOrders are:- [3.2.1 MCclickASX Connection Parameters:](#)

CLIENTS_PORT = TCP/IP port that clients (in this case ASXOrders) connect to.
CLIENTS_PORT =7001

3.3.2 ASX Trade Connection Parameters:

ASX_CLICK_GATEWAY = ASX Trade Gateway to connect to – supplied by the ASX.

e.g. ASX_CLICK_GATEWAY =vpnETE

ASX_CLICK_PORT =Port to use on ASX Trade Gateway – supplied by the ASX.

ASX_CLICK_PORT =21024

*questDT=10.37.253.177

3.3.3 ASX Trade Connection Options:

These control the use of encryption or compression on the OM API link.

The ASX will advise if Compression or Encryption is to be used and our settings must match theirs.

e.g. OMNIAPI_COMPRESS =NO (YES)

e.g. OMNIAPI_ENCRYPT =NO (YES)

3.3.4 Logging Options:

These control the amount of information being logged.

DIAGNOSTIC_LEVEL =1 - controls how much information is logged in text diagnostics messages. Higher number mean more information is logged. Just use the default value unless otherwise instructed by RJE support personnel.

LOG_MESSAGES =A - controls amount of info logged

A=All, C=Client, X=Exchange, T=Text, W=Warning, E=Error, N=None

Can specify a single type or multiples (e.g. C+T)

A=C+X+T

T -> All Text messages includes warnings & errors.

W -> Warnings includes errors

Lowest setting is E -> Error messages only.

Error messages are always logged if logging is enabled.

N -> Turns logging off

Log files can get big quickly but logged info gives us the ability to diagnose problems.

LOG_FILTER - further control on amount of info logged for Exchange & MC API messages

D = log deals

5 = log BO5's

When the filter is set no other query response/broadcast message types are logged.

By default the filter is not set and all message types are logged.

Example – the recommended settings for MCTrades are:-

- LOG_MESSAGES =C+W
- LOG_FILTER =D5

3.3.5 Other Configuration Options:

TCPIP_CONNECTIONS =n Allow 'n' concurrent TCP/IP connect attempts (backlog), Default = 5.

BCAST_POLL_RATE=n - Broadcast Poll Rate - Polls per second (default = 10)
The ASX may instruct users to set the poll rate to a particular value.

BCAST_HBEAT_POLL=n – Special poll rate for order entry apps not subscribed to any broadcasts (default = 1).

QUIT_DELAY= 'n' milliseconds - time to wait before closing client socket after sending quit response. Default = 200 M/S. (You should not need to use this.)

Performance Statistics:

BCAST_STATS=n - Output Broadcast Stats every 'n' seconds - zero default = no stats

4. Password Changing:

ASX Trade user passwords can expire and currently must be changed every 60 days. ASXOrders will detect that a password has expired and will automatically change the password to one that is compliant with the ASX rules. The .ini file is updated with the new password; this is a full automated process and should not need manual intervention.

See [3.2.2 ASX Trade Logon Parameters:](#) for more details.

5. Recovery Strategy:

Each order transaction has a unique key:- OrderID+Symbol+Side

This allows for OMX block order transactions, although these are not in current use in the ASX Trade system. There are block transactions for Market Maker quotes, but quotes are currently not stored in the SQL database.

When the link to ASX Trade drops out, ASXOrders performs the following recovery sequence:-

After some consideration the easiest way for us to handle this is:-

1. Wipe all order data in memory.
2. Close all client connections (kicking off, connected applications).
3. Market all order in the database as not current.
4. Open a new.orders file.
5. Recover the link then re-download all order data from ASX Trade.

So each session starts with an orders snapshot which is then updated via BO5 broadcasts.

In other words the only difference between a between the ASX Trade link dropping out and a clean restart is that the program keeps running and attempts to recover.

The other advantage of this scenario is that each restart is a clean restart if things seem to be going wrong you can just restart.

If ASXOrders detects any serious problem, it will just stop running.

6. Command Clients – Order Cancellation:

6.1 Order Cancellation Overview:

ASXOrders does not support the cancellation of orders.

7. SQL Database:

The database is common to a number of order pulling applications which gather orders information for use by the web client application.

The ASXOrders program uses the “npgsql” .net data provider for PostgreSQL.

The ASXOrders program typically calls PostgreSQL Functions (Stored Procedures) for database access and updating. This approach was chosen as we believe it will deliver the best performance.

7.1 SQL Database Tables:

7.1.1 Table - system

Standard Web Client table – used by ASXOrders.

7.1.2 Table – system_state

Standard Web Client table – used by ASXOrders.

7.1.3 Table – asx_orders

This is the main table of interest to Web Clients and other application which wish to display data.

When the field order_active='Y' the order is an active order which is a candidate for cancellation.

When the field order_active='I' the order is an inactive order which is a candidate for cancellation.

As orders trade out or are cancelled order_active is set to 'N'.

On a restart/recovery order_active is set to 'U'. After recovery any orders which remains as order_active='U' must have been deleted or traded out while the system was offline.

We are keeping orders information as it may be useful, if we continue with this approach are archiving process must be developed.

Function :- func_asx_update_order() – Updates the orders table for each execution report transaction. Note: It is likely that this function will be specific to the SFE other systems will use the same database table but may have a slightly different update function.

Function :- func_asx_wipe_current_orders() – Sets order_active to 'U' on restart/recovery. Can be changed to delete orders if required,

```
CREATE TABLE asx_orders
(
  id bigserial NOT NULL,
  system_id uuid NOT NULL,
  order_id character varying(50) NOT NULL,
  message_no integer,
  order_active character(1),
  order_status character varying(4),
  order_type character varying(4),
  order_book_class character varying(4),
  ord_change_reason character varying(4),
  order_ref character varying(50),
  ....
```

[7.2 SQL Database Parameters:](#)

See [3.2.8 SQL Database Parameters](#)

[7.3 npgsql files:](#)

The following files should reside in the same directory as SFEOrders.exe:-

07/07/2007	12:09 AM	282,624	Mono.Security.dll
28/09/2011	08:55 PM	365,568	Npgsql.dll

These files are the “npgsql” .net data provider for PostgreSQL.

7.4 SQL Script files:

The following files create database tables:-

```
28/05/2012  11:59 AM      2,058  asx_create_orders.sql
03/05/2012  10:13 AM      999    asx_create_order_cancel_result.sql
```

The following files create database functions:-

```
28/05/2012  12:02 PM      8,636  func_asx_update_order.sql
03/05/2012  10:14 AM      4,715  func_asx_update_order_cancel.sql
03/05/2012  03:04 PM      675    func_asx_wipe_current_orders.sql
```

8. PHP – Web Client:

This is documented elsewhere.

9. Fields Mappings ASXOrders <-> ASX Trade:

ASX Orders Field Name	ASX Trade Field Name
firm_id	trading_code.ex_customer_s
crader_id	trading_code.user_id_s
crder_id	order_number_u
cl_order_id	exchange_info_s
exec_id	(last trade.trade_number)
exec_trans_type	<No Value>
order_status	order_state_u
order_bos_pos	ob_position_u
account	ex_client_s
exchange_code	<Always “ASX”>
symbol	Series Name – (series.ins_id_s)
Side	bid_or_ask_c
order_qty	mp_quantity_i
Price	premium_i
last_shares	(last trade.quantity)

cum_quantity	<No Value>
transact_time	<No Value>
Text	customer_info_s
order_type	order_type_c
expire_time	time_validity_n.duration
commodity	<No Value>
Month	<No Value>
Year	<No Value>
ob_class	<RJE Derived field>
block	block_n
change_reason	change_reason_c
combo_mark	combo_mark_c
display_qty	display_quantity_i
exch_ord_type	exch_order_type_n
ex_state	ext_t_state_c
give_up_member	give_up_member
oc_request	open_close_request_c
order_no_bin	order_number_u
Party	Party
sequence_no	sequence_number_u
Series	Series
stop_condition	stop_condition_c
stop_series	Stop Orders – stop_series
stop_series_name	Stop Orders -(stop_series.ins_id_s)
time_val_type	time_validity_n.type
total_volume	total_volume_i
trading_code	trading_code
transaction_no	Transaction_number_n
user_code	ex_user_code
trans_ack	(Txstat from last BO5 transaction)
active	<RJE Derived field>
unique_key	<RJE Derived field>