



Market Access Control User Manual

Version 2.4

Updated April 22, 2020

Japannext Co., Ltd. believes that the information in this publication is accurate as of its publication date.

This document may include typographical errors, omissions or technical inaccuracies. Japannext Co., Ltd. reserves the right to revise the document and to make changes without notice. Japannext Co., Ltd. acknowledges the proprietary rights in the trademarks and product names of other companies mentioned in this document.

Contents

1. Introduction.....	3
2. Overview.....	3
3. Service Configuration	3
4. Web Based Interface	4
4.1. Groups	5
4.1.1. Activate/Suspend groups	5
4.2. All sessions.....	6
4.3. Individual sessions.....	8
4.3.1. Activate/Suspend session.....	8
4.3.2. Cancel orders.....	8
5. Programmable API.....	9
5.1. Session status	9
5.2. Suspend all sessions	9
5.3. Suspend individual session.....	10
5.4. Activate individual session	10
5.5. Suspend group.....	10
5.6. Activate group	11
5.7. Cancel Orders	11
Revision History	12

1. Introduction

This document explains access to the Market Access Control (MAC) service of Japannext PTS. It describes the service's configuration and specifies the external programmable interface. For further information and inquiries regarding MAC service, please contact Japannext PTS Technical Support via email to ito@japannext.co.jp.

2. Overview

The purpose of MAC service is to allow users to suspend trading activity on their order entry ports in emergency situations. Upon suspension, all open orders will be automatically cancelled and the trading system will start to reject new orders. Access to the service is only available via Japannext PTS private network. Users can choose between a web-based interface or programmable API.

3. Service Configuration

On request, a specific instance of MAC will be created for each individual client of the service. The full list of order entry ports covered by the MAC instance must be specified in the request. Japannext PTS will provide a unique URL, which must be used to access the service.

4. Web Based Interface

The simplest way to use MAC is to access its web-based interface. To do so, please open the URL provided by Japannext PTS in your web browser.

Japannext Market Access Control

PLEASE READ IMPORTANT RISK WARNING BELOW

You are going to use Market Access Control service which allows you to manage status of the trader accounts belonging to your firm. Upon trader account suspension all of its open orders will be automatically eliminated by the exchange. No more orders will be accepted until further account re-activation.

Japannext assumes no obligation or responsibility whatsoever for any damages resulting from the use of Market Access Control service. Please check the box below, for accepting the terms and conditions.

☐ I understand all the risks and accept the terms and conditions stated above.

Market Access Control for: Japannext Co., Ltd. (11226)

Groups:

GROUP	STATUS	SET_STATE
JNXA	active	Suspend
JNXB	active	Suspend
JNXC	active	Suspend

Suspend all displayed sessions: Suspend All Sessions

SESSION	STATUS	LOGON_STATE	SET_STATE
QJNX11	active	loggedOffBySelf	Suspend Cancel Orders
QJNX12	active	loggedOffBySelf	Suspend Cancel Orders
QJNX13	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1A	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1B	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1C	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1D	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1E	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1M	active	loggedOffBySelf	Suspend Cancel Orders
QJNX1N	active	loggedOffBySelf	Suspend Cancel Orders
TD11226A	active	loggedOffBySelf	Suspend Cancel Orders
TD11226E	active	loggedOffBySelf	Suspend Cancel Orders
TD11226F	active	loggedOffBySelf	Suspend Cancel Orders
TD11226G	active	loggedOffBySelf	Suspend Cancel Orders

© 2013-2020 Japannext Co., Ltd. All Rights Reserved.

Figure 1

Figure 1 shows the MAC web interface. The interface has two sections: Groups and Individual sessions. Every action requires doing the following steps:

1. Tick legal notice acceptance check box (Figure 2)
2. Click the desired action button
3. Confirm the action in the confirmation dialog box. "OK" to proceed with the action, "Cancel" to cancel the action

☐ I understand all the risks and accept the terms and conditions stated above.

Figure 2

Upon a successful action, you will receive an “OK” confirmation message as shown in Figure 3



Figure 3

4.1. Groups

This section describes the actions available as a group (Figure 4)

Groups:		
GROUP	STATUS	SET_STATE
JNXA	intermediate	Suspend
JNXB	active	Suspend
JNXC	suspended	Activate

Figure 4

There are three possible values in the **STATUS** column:

- **Active:** All sessions subscribed to this group are active
- **Suspended:** All sessions subscribed to this group are suspended
- **Intermediate:** Some sessions subscribed to this group are active, and some are suspended

4.1.1. Activate/Suspend groups

When clicking on the “Activate” or “Suspend” button in the groups section, **ALL the sessions subscribed to the respective group will change their status to the one on the selected button.** Refer to Figure 5 and Figure 6 for the before and after of test group JNXA as an example.

Market Access Control for: Japannext Co., Ltd. (11226)			
Groups:			
GROUP	STATUS	SET_STATE	
JNXA	active	Suspend	
JNXB	active	Suspend	
JNXC	active	Suspend	
Suspend all displayed sessions: Suspend All Sessions			
SESSION	STATUS	LOGON_STATE	SET_STATE
QJNX01	active	loggedOnBySelf	Suspend Cancel Orders
QJNX02	active	loggedOnBySelf	Suspend Cancel Orders
QJNX03	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0A	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0B	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0C	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0D	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0E	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0M	active	loggedOnBySelf	Suspend Cancel Orders
QJNX0N	active	loggedOnBySelf	Suspend Cancel Orders
TD11226A	active	loggedOnBySelf	Suspend Cancel Orders
TD11226E	active	loggedOnBySelf	Suspend Cancel Orders
TD11226F	active	loggedOnBySelf	Suspend Cancel Orders
TD11226G	active	loggedOnBySelf	Suspend Cancel Orders

Figure 5

Groups:

GROUP	STATUS	SET_STATE
JNXA	suspended	Activate
JNXB	active	Suspend
JNXC	active	Suspend

Suspend all displayed sessions: Suspend All Sessions

SESSION	STATUS	LOGON_STATE	SET_STATE
QJNX01	suspended	loggedOffBySelf	Activate Cancel Orders
QJNX02	suspended	loggedOffBySelf	Activate Cancel Orders
QJNX03	suspended	loggedOffBySelf	Activate Cancel Orders
QJNX0A	active	loggedOffBySelf	Suspend Cancel Orders
QJNX0B	active	loggedOffBySelf	Suspend Cancel Orders
QJNX0C	active	loggedOffBySelf	Suspend Cancel Orders
QJNX0D	active	loggedOffBySelf	Suspend Cancel Orders
QJNX0E	active	loggedOffBySelf	Suspend Cancel Orders
QJNX0M	active	loggedOffBySelf	Suspend Cancel Orders
QJNX0N	active	loggedOffBySelf	Suspend Cancel Orders
TD11226A	suspended	loggedOffBySelf	Activate Cancel Orders
TD11226E	active	loggedOffBySelf	Suspend Cancel Orders
TD11226F	active	loggedOffBySelf	Suspend Cancel Orders
TD11226G	active	loggedOffBySelf	Suspend Cancel Orders

Figure 6

4.2. All sessions

The only action available to apply to all sessions is “Suspend All Sessions” (Figure 7). **Note that there is no “Activate All Sessions” feature**, suspended sessions can only be activated by group or on an individual session basis.

Suspend all displayed sessions: Suspend All Sessions

Figure 7

Note that when you use this ‘Suspend All Sessions’ functionality, the system will try to suspend the sessions listed one-by-one. If there is a session that is already suspended, the message “Error: Transaction Failed” for that particular session will be displayed.

In the following example, Figure 8 shows the state before suspending all sessions. Figure 9 and Figure 10 show the action confirmation for session states, and the subsequent status screen after clicking the “Suspend All Sessions” button respectively.

Groups:

GROUP	STATUS	SET_STATE
JNXA	intermediate	Suspend
JNXB	active	Suspend
JNXC	active	Suspend

Suspend all displayed sessions: Suspend All Sessions

SESSION	STATUS	LOGON_STATE	SET_STATE
QJNXt1	active	loggedOffBySelf	Suspend Cancel Orders
QJNXt2	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXt3	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXtA	active	loggedOffBySelf	Suspend Cancel Orders
QJNXtB	active	loggedOffBySelf	Suspend Cancel Orders
QJNXtC	active	loggedOffBySelf	Suspend Cancel Orders
QJNXtD	active	loggedOffBySelf	Suspend Cancel Orders
QJNXtE	active	loggedOffBySelf	Suspend Cancel Orders
QJNXtM	active	loggedOffBySelf	Suspend Cancel Orders
QJNXtN	active	loggedOffBySelf	Suspend Cancel Orders
TD11226A	active	loggedOffBySelf	Suspend Cancel Orders
TD11226E	active	loggedOffBySelf	Suspend Cancel Orders
TD11226F	active	loggedOffBySelf	Suspend Cancel Orders
TD11226G	active	loggedOffBySelf	Suspend Cancel Orders

Figure 8

```

Success: Transaction OK :: (firmId=11226,sessionId=TD11226A,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXt1,status=S)
Error: Transaction failed :: (firmId=11226,sessionId=QJNXt2,status=S)
Error: Transaction failed :: (firmId=11226,sessionId=QJNXt3,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtA,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtB,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtC,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtD,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtE,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=TD11226E,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=TD11226F,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=TD11226G,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtM,status=S)
Success: Transaction OK :: (firmId=11226,sessionId=QJNXtN,status=S)

```

Figure 9

Groups:			
GROUP	STATUS	SET_STATE	
JNXA	suspended	Activate	
JNXB	suspended	Activate	
JNXC	suspended	Activate	
Suspend all displayed sessions: Suspend All Sessions			
SESSION	STATUS	LOGON_STATE	SET_STATE
QJNXI1	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXI2	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXI3	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXIA	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXIB	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXIC	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXID	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXIE	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXIM	suspended	loggedOffBySelf	Activate Cancel Orders
QJNXIN	suspended	loggedOffBySelf	Activate Cancel Orders
TD11226A	suspended	loggedOffBySelf	Activate Cancel Orders
TD11226E	suspended	loggedOffBySelf	Activate Cancel Orders
TD11226F	suspended	loggedOffBySelf	Activate Cancel Orders
TD11226G	suspended	loggedOffBySelf	Activate Cancel Orders

Figure 10

4.3. Individual sessions

Individual sessions have two different actions available:

- Set state (Activate/Suspend)
- Cancel orders

4.3.1. Activate/Suspend session

To set the state of an individual session, click the “Activate” or “Suspend” button. Upon a successful request, an “OK” message will be displayed, as shown in Figure 3.

4.3.2. Cancel orders

This action will cancel all open orders for the selected session without suspending the session.

Note that if there are no open orders, a “Transaction Failed” message will be displayed (Figure11). This action is only available for active sessions.



Figure 11

IMPORTANT NOTE

ONLY SESSIONS WITH "CANCEL ON DISCONNECT" FUNCTIONALITY ENABLED CAN TRIGGER THE MASS ORDER CANCELLATION WHEN THE SESSION IS SET TO "SUSPENDED".

ORDERS ENTERED VIA SESSIONS WITHOUT "CANCEL ON DISCONNECT" FUNCTIONALITY ENABLED WILL NOT BE CANCELLED WHEN THE "SUSPEND ALL SESSIONS" FEATURE IS ACTIVATED

CONTACT JAPANNEXT PTS FOR SUPPORT RELATING TO THE "CANCEL ON DISCONNECT" FUNCTION.

5. Programmable API

The primary MAC functionality can also be used via API.

A HTTP/1.1 POST request is sent to the MAC service (e.g. via the curl command) with the variable "mode" set to "api" and other optional parameters.

The reply will be in JSON format. An empty value returned indicates that the MAC service is not currently available. Examples of use cases of the available commands are as follows:

5.1. Session status

To query the status of all configured sessions simply send an empty request to the service:

```
curl -X POST -k -d "mode=api" http://X.X.X.X/
```

Sample reply:

```
[{"session": "QJNXn1", "status": "active"},
{"session": "QJNXn2", "status": "active"},
{"session": "QJNXn3", "status": "active"},
{"session": "QJNXn4", "status": "active"},
{"session": "TN11226A", "status": "active"},
{"session": "TN11226B", "status": "active"},
{"group": "JNXA", "status": "active"},
{"group": "JNXB", "status": "active"},
{"group": "JNXC", "status": "active"}]
```

5.2. Suspend all sessions

```
$ curl -X POST -k -d "suspend_all=S&mode=api" http://X.X.X.X/
```

Sample reply:

```
[{"session": "QJNXn1", "status": "suspended"},
{"session": "QJNXn2", "status": "suspended"},
{"session": "QJNXn3", "status": "suspended"}]
```

```
{"session": "QJNXn4", "status": "suspended"},  
{"session": "TN11226A", "status": "suspended"},  
{"session": "TN11226B", "status": "suspended"},  
{"group": "JNXA", "status": "suspended"},  
{"group": "JNXB", "status": "suspended"},  
{"group": "JNXC", "status": "suspended"}]
```

5.3. Suspend individual session

To suspend an individual session, add the “set_state=S” and “session_id=<session-name>” parameters to the request:

```
$ curl -X POST -k -d "set_state=S&session_id=QJNXn1&mode=api" http://X.X.X.X/
```

Sample reply:

```
[{"session": "QJNXn1", "status": "suspended"},  
{"session": "QJNXn2", "status": "active"},  
{"session": "QJNXn3", "status": "active"},  
{"session": "QJNXn4", "status": "active"},  
{"session": "TN11226A", "status": "active"},  
{"session": "TN11226B", "status": "active"},  
{"group": "JNXA", "status": "intermediate"},  
{"group": "JNXB", "status": "active"},  
{"group": "JNXC", "status": "active"}]
```

5.4. Activate individual session

To activate an individual session, add the “set_state=A” and “session_id=<session-name>” parameters to the request:

```
$ curl -X POST -k -d "set_state=A&session_id=QJNXn4&mode=api" http://X.X.X.X/
```

Sample reply:

```
[{"session": "QJNXn1", "status": "suspended"},  
{"session": "QJNXn2", "status": "suspended"},  
{"session": "QJNXn3", "status": "suspended"},  
{"session": "QJNXn4", "status": "active"},  
{"session": "TN11226A", "status": "suspended"},  
{"session": "TN11226B", "status": "suspended"},  
{"group": "JNXA", "status": "suspended"},  
{"group": "JNXB", "status": "suspended"},  
{"group": "JNXC", "status": "intermediate"}]
```

5.5. Suspend group

To suspend all sessions in a group, add the “set_groupstate=S” and “group=<group-name>” parameters to the request:

```
$ curl -X POST -k -d "set_groupstate=S&group=JNXB&mode=api" http://X.X.X.X/
```

Sample reply:

```
[{"session": "QJNXn1", "status": "suspended"},
{"session": "QJNXn2", "status": "suspended"},
{"session": "QJNXn3", "status": "suspended"},
{"session": "QJNXn4", "status": "active"},
{"session": "TN11226A", "status": "suspended"},
{"session": "TN11226B", "status": "active"},
{"group": "JNXA", "status": "suspended"},
{"group": "JNXB", "status": "suspended"},
{"group": "JNXC", "status": "active"}]
```

5.6. Activate group

To re-activate all sessions in a group, add the “set_groupstate=S” and “group=<group-name>” parameters to the request:

```
$ curl -X POST -k -d "set_groupstate=A&group=JNXC&mode=api" http://X.X.X.X/
```

Sample reply:

```
[{"session": "QJNXn1", "status": "suspended"},
{"session": "QJNXn2", "status": "suspended"},
{"session": "QJNXn3", "status": "suspended"},
{"session": "QJNXn4", "status": "active"},
{"session": "TN11226A", "status": "suspended"},
{"session": "TN11226B", "status": "active"},
{"group": "JNXA", "status": "suspended"},
{"group": "JNXB", "status": "suspended"},
{"group": "JNXC", "status": "active"}]
```

5.7. Cancel Orders

All open orders for a session will be canceled. Note that “0” is returned when there are no orders available to cancel.

```
$ curl -X POST -k -d "cxl_session_orders=X&session_id=QJNXn4&mode=api"
http://X.X.X.X
```

Sample reply (success):

```
[result: 1]
```

Sample reply (No open orders/fail):

```
[result: 0]
```

Revision History

Date	Version	Description
2013-02-15	1.0	Initial revision
2013-02-25	1.1	Update purpose section Added activate screen case
2014-01-09	1.2	Added API mode section
2014-08-11	2.0	Changed service name to Market Access Control
2014-10-22	2.1	Added group suspend and activate feature to GUI and API
2018-02-01	2.2	Added Cancel Orders options in GUI and API
2020-04-01	2.3	Change company name
2020-04-22	2.4	Add additional GUI screenshots Add caption to all images Re-organize GUI samples