

Event Notification Function

Document created on: December 7, 2003

1. Overview

This document describes the specification of event notification function from QSS to Client.

2. Functions

- Notifies Client of the following events in real time.
 - 1) Order status notification: Notifies that the status of order has been updated.
 - 2) Error/attention notification: Notifies error/attention messages encountered on QSS.
- Client can select the types of events it receives.

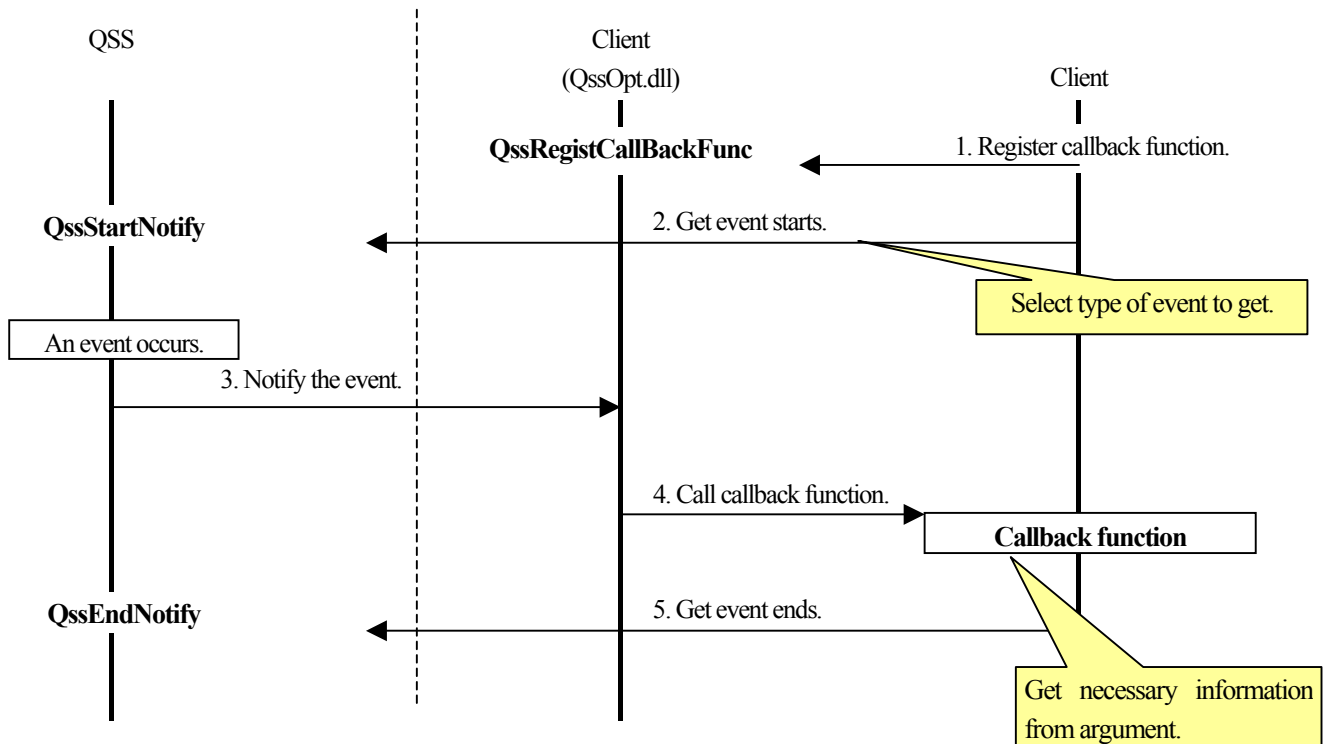
NOTE: It is not notified to Client that the error/attention messages once encountered are resolved. Whether it is resolved or not can be confirmed with QssGetError API.

3. Procedures for Client to be notified of event

When an event takes place, QSS calls Client's callback function to notify Client of the event. It is required for Client to register the callback function by using **QssRegisterCallBackFunc** API included in QssOpt.DLL of NetOrderSDK. Also, Client is required to call **QssStartNotify** API (RPC) to request that QSS notifies Client of event that takes place.

In order to stop event notification, call **QssEndNotify** API (RPC) from Client. It is required to call **QssEndNotify** API when closing Client. Below illustrates the workflow.

Workflow:



NOTE:

- When you reboot QSS, call QssStartNotify again subsequently.

4. Callback function

Callback functions should be of the following type:

```
void CALLBACK CallBack(QSS_PRINTER_INFO* PrinterInfo, QSS_EVENT* Event);
```

QSS_PRINTER_INFO, the first parameter, contains information as to the printer that sends notification, while QSS_EVENT, the second parameter, contains information as to the event to be notified. (Refer to the reference that is described later.) Referring to the contents of these parameters, Client knows what event has taken place on which printer.

5. API reference

5.1. QssRegistCallBackFunc

Registers the callback function to receive error information.

NOTE: This function is provided in QssOpt.DLL.

```
long QssRegistCallBackFunc (
    QSS_CLIENT_INFO*      ClientInfo
    LPQSSCALLBACKFUNC    lpfnCallback
);
```

5.1.1. Parameter

ClientInfo: Specifies Client information structure.
lpfnCallback: Specifies pointer of callback function.

5.1.2. Definition of callback function

```
void CALLBACK CallBack(QSS_PRINTER_INFO* PrinterInfo, QSS_EVENT* Event);
```

When callback function is called, Client can get necessary information from PrinterInfo and Event.

5.1.3. Return value

Returns QSS_SUCCESS(0) when succeeded; else returns QSS_FAIL(1).

5.2. QssStartNotify

Request QSS to notify the Client of event.

```
long QssStartNotify (
    QSS_CLIENT_INFO      ClientInfo
    long                 nKind,
);
```

5.2.1. Parameter

ClientInfo: Specifies Client information structure
nKind: Specifies type of event. You may specify multiple types of events.

NOTIFY_ORDER (0x1): Order status notification
 NOTIFY_ALERT(0x4): Error/Attention notification

5.2.2. Return value

Returns QSS_SUCCESS(0) when succeeded; else returns the following.

QSS_NO_SUPPORT: This printer model does not support the API.
 QSS_ALREADY_REQUEST: Already requested.

5.3. QssEndNotify

Cancels event notification that has been requested with QssStartNotify. No event is notified of from QSS.

```
long QssEndNotify (
    QSS_CLIENT_INFO      ClientInfo
);
```

5.3.1. Parameter

ClientInfo: Specifies Client information structure.

5.3.2. Return value

Returns QSS_SUCCESS(0) when succeeded; else returns the following.

QSS_NOT_SUPPORT: This QSS model does not support the API.
QSS_NOT_REQUEST: Not requested.

5.4. QSS_PRINTER_INFO structure

```
typedef struct _QSS_PRINTER_INFO {
    char                Name[20];
    unsigned long       Version;
    unsigned char       IPAddress[4];
    unsigned char       Reserve[36];
} QSS_PRINTER_INFO;
```

5.4.1. Member

Name: Specifies the QSS model name. This string is NULL terminated.
Version: Specifies the version of NetOrder API. Version number is described in hex.
 e.g. When the NetOrder API is of version 1.2.3, it is described as 0x01020300.
IPAddress: Specifies IP address of the QSS.
Reserve: Reserved (Unused)

5.5. QSS_EVENT structure

```
typedef struct _QSS_EVENT {
    unsigned short      EventNo;
    DWORD               Param1;
    DWORD               Param2;
    wchar_t             Message[256];
    unsigned short      Reserve1;
    unsigned_int64      RefId;
    unsigned char        Reserve[16];
} QSS_EVENT;
```

5.5.1. Member

EventNo: Specifies event number.

Value	Description
QSS_EVENT_ORDER_PRINT(1)	Printing start notification Notifies that printing has started.
QSS_EVENT_ORDER_COMPLETE(2)	Printing complete notification

QSS_EVENT_ORDER_CANCEL(4)

Notifies that printing has been completed.

Order cancel notification

QSS_EVENT_ALERT(7)

Notifies that the order has been canceled.

Error/Attention notification

Notifies that an error/attention message has been arisen on QSS.

Param1, Param2: Specify additional information. Additional information to be specified varies depending on the event to be notified.

	<i>Param1</i>	<i>Param2</i>
Print start notification	Request number	Receipt number
Print complete notification	Request number	Receipt number
Order cancel notification	Request number	Receipt number
Error/attention notification	Error number	Error sub-number *1

*1 Error sub-number: In the upper 16 bit of error sub-number stores display format of the sub-number, while in the lower 16 bit stores error sub-number. Display format is used when displaying error sub-number on the monitor.

For example, when the display format is QSS_ALERT_DISP_HEX4, error sub-number is to be displayed in 4-digit hex. Below is the list of display format types.

Value	Description
QSS_ALERT_DISP_NO (0)	No display of error sub-number.
QSS_ALERT_DISP_DEC (1)	Display error sub-number in decimal number.
QSS_ALERT_DISP_HEX2 (2)	Display error sub-number in 2-digit hex.
QSS_ALERT_DISP_HEX4 (3)	Display error sub-number in 4-digit hex.

Message: Specifies the text of the error and attention messages. The language of the text is consistent with the language specified for QSS. This member is available for error/attention messages only.

RefId: Specifies reference number. Reference number is defined only for notifications of print start, print completion and order delete.

Reserve: Reserved (Unused)

5.6. QSS_CLIENT_INFO structure

```
typedef struct _QSS_CLIENT_INFO {
    char            User[20];
    char            Host[20];
    unsigned char    Address[6];
    unsigned char    IPAddress[4];
    unsigned short    Port;
    unsigned long     Version;
    unsigned short    Level;
    unsigned char     Reserve[38];
} QSS_CLIENT_INFO;
```

5.6.1. Member

User: Specify the user name. The string should be a maximum of 19 characters and NULL terminated.

Host: Specify the host name. The string should be a maximum of 19 characters and NULL terminated.

Address: Specifies MAC address of Client host PC.

IPAddress: Specifies IP address of Client host PC.

Port: Define port number of the socket to receive event notification.

Version: Specifies the version of NetOrderAPI to be used.

Level: Define Client level.

Value	Description
QSS_CLIENT_LEVEL1	Status of orders that the Client has sent to QSS is received in order status notification from QSS.
QSS_CLIENT_LEVEL2	Status of all orders is received in order status notification from QSS.

Reserve: Reserved (Unused)

5.6.2. Remarks

QSS_CLIENT_INFO is used for QSS to control and manage Client.

6. Revision History

<i>Date</i>	<i>Description</i>	<i>Editor</i>
December 7, 2003	Newly created	S. Heya