

HS MSG C Source Library v1.0.1

Message Oriented Middleware Library

Test Report

16 February 2012
Rev 1.1

Table of Contents

1 Introduction.....	2
2 Test Cases.....	2
2.1 Functional Tests.....	2

1 Introduction

HS MSG is a C messaging library (message oriented middleware) for communication between threads of the same process and between different processes on the same or on different computers. HS MSG C messaging SDK runs on Windows, Linux, MAC OSX, iPhone iOS.

This document is a test report of the testing carried out on version 1.0.0.

2 Test Cases

To call HS MSG API functions include the following header file:

2.1 Basic Functionality

Pre-conditions:

- 1) Initialize: HsMsgInit
- 2) Create inbox: HsMsgCreateInbox
- 3) Add queue: HsMsgAddQueue

#	Test Description	Expected result	Platform	Pass / Fail
1	Send short text message without waiting for result: HsMsgPostMessage	Message received by remote peer	Windows Linux Mac OS X	pass pass pass
2	Receive message from remote peer: HsMsgReadQueue	Message sent from remote peer is received	Windows Linux Mac OS X	pass pass pass
3	Tests 1 and 2 can be executed multiple times in succession in any order	Messages sent and received	Windows Linux Mac OS X	pass pass pass
4	Send a large block of data (16384 bytes) using HsMsgSendMessage (blocks until message delivered)	Message received by remote peer	Windows Linux Mac OS X	pass pass pass
5	Receive message from remote peer: HsMsgReadQueue containing 16384 bytes	Message sent from remote peer is received	Windows Linux Mac OS X	pass pass pass
6	Continually send messages containing 16384 blocks of data – send a file to remote peer using HsMsgSendMessage	File is received by remote peer	Windows Linux Mac OS X	pass pass pass
7	Continually receive message from remote peer: HsMsgReadQueue containing 16384 bytes, save to disk file	File sent from remote peer is received and saved to disk	Windows Linux Mac OS X	pass pass pass

2.2 API Tests

#	Test Description	Expected result	Platform	Pass / Fail
1	HsMsgInit	Library initialized	Windows Linux Mac OS X	pass pass pass
2	HsMsgCleanUp	Library de-initialized	Windows Linux Mac OS X	pass pass pass
3	1) HsMsgInit 2) HsMsgCreateInbox(0)	Library initialized Inbox created using auto-selected IP port, IP port number is set to the actual port number used	Windows Linux Mac OS X	pass pass pass
4	1) HsMsgInit 2) HsMsgCreateInbox(specific port)	Library initialized Inbox created using specified IP port number	Windows Linux Mac OS X	pass pass pass
5	HsMsgCreateInbox(0) (HsMsgInit has not been called)	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
6	1) HsMsgInit 2) HsMsgCreateInbox(NULL)	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
7	1) 64 inboxes already created 2) HsMsgCreateInbox(x)	HSMSG_RC_INBOX_ALLOC	Windows Linux Mac OS X	pass pass pass
8	1) HsMsgInit 2) HsMsgCreateInbox(x), port x is already bound	HSMSG_RC_SOCKET_BIND	Windows Linux Mac OS X	pass pass pass
9	HsMsgAddQueue while library not initialized	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
10	HsMsgAddQueue with NULL queue name	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
11	HsMsgAddQueue with inbox_id out of allowable range	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
12	HsMsgAddQueue with inbox id of inbox that is not allocated	HSMSG_RC_INBOX_NOTALLOC	Windows Linux Mac OS X	pass pass pass
13	HsMsgAddQueue with a queue name that has already been added to the inbox (another queue already exists with the same name on the same inbox)	HSMSG_RC_QNAME_EXISTS	Windows Linux Mac OS X	pass pass pass
14	HsMsgAddQueue while maximum number of queues per inbox already allocated	HSMSG_RC_Q_ALLOC	Windows Linux Mac OS X	pass pass pass
15	HsMsgTick		Windows Linux Mac OS X	pass pass pass
16	HsMsgReadQueue while library not initialized	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
17	HsMsgReadQueue will NULL message pointer	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
18	HsMsgReadQueue specifying unallocated inbox	HSMSG_RC_INBOX_NOTALLOC	Windows Linux	pass pass

			Mac OS X	pass
19	HsMsgReadQueue with queue_id out of range	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
20	HsMsgReadQueue with queue_id referring to an unallocated queue	HSMSG_RC_QNOTALLOC	Windows Linux Mac OS X	pass pass pass
21	HsMsgReadQueue, queue is empty	HSMSG_RC_QEMPTY	Windows Linux Mac OS X	pass pass pass
22	HsMsgReadQueue, queue non empty	HSMSG_RC_OK	Windows Linux Mac OS X	pass pass pass
23	HsMsgPostMessage while library not initialized	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
24	HsMsgPostMessage destination address NULL	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
25	HsMsgPostMessage destination name NULL	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
26	HsMsgPostMessage pMsg NULL	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
27	HsMsgPostMessage Destination address is in invalid format	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
28	HsMsgPostMessage, inbox_id refers to an unallocated inbox	HSMSG_RC_INBOX_NOTALLOC	Windows Linux Mac OS X	pass pass pass
29	HsMsgPostMessage too many messages sessions in progress	HSMSG_RC_NO_CTX	Windows Linux Mac OS X	pass pass pass
30	HsMsgPostMessage Destination address refers to an invalid IP address or IP port	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
31	HsMsgPostMessage destination_queue_name is over 32 bytes	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
32	HsMsgPostMessage, sending a message data block too large, memory allocation failure	HSMSG_RC_NO_MEM	Windows Linux Mac OS X	pass pass pass
33	HsMsgSendMessage while library not initialized	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
34	HsMsgSendMessage destination address NULL	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
35	HsMsgSendMessage destination name NULL	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
36	HsMsgSendMessage pMsg NULL	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
37	HsMsgSendMessage Destination address is in invalid format	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
38	HsMsgSendMessage, inbox_id refers to	HSMSG_RC_INBOX_NOTALLOC	Windows	pass

	an unallocated inbox		Linux Mac OS X	pass pass
39	HsMsgSendMessage too many messages sessions in progress	HSMSG_RC_NO_CTX	Windows Linux Mac OS X	pass pass pass
40	HsMsgSendMessage Destination address refers to an invalid IP address or IP port	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
41	HsMsgSendMessage destination_queue_name is over 32 bytes	HSMSG_RC_INV_PAR	Windows Linux Mac OS X	pass pass pass
42	HsMsgSendMessage, sending a message data block too large, memory allocation failure	HSMSG_RC_NO_MEM	Windows Linux Mac OS X	pass pass pass
43	HsMsgSendMessage, time out waiting for responses from remote peer	HSMSG_RC_TIMEOUT	Windows Linux Mac OS X	pass pass pass
44	HsMsgSendMessage, remote peer has not enough memory to receive the message	HSMSG_RC_NO_MEM	Windows Linux Mac OS X	pass pass pass
45	HsMsgSendMessage, destination queue does not exist on remote peer	HSMSG_RC_QNOTFOUND	Windows Linux Mac OS X	pass pass pass
46	HsMsgSendMessage, destination queue is full on remote peer	HSMSG_RC_QFULL	Windows Linux Mac OS X	pass pass pass
47	HsMsgRemoveQueue while library not initialized	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
48	HsMsgRemoveQueue inbox_id refers to an unallocated inbox	HSMSG_RC_INBOX_NOTALLOC	Windows Linux Mac OS X	pass pass pass
49	HsMsgRemoveQueue queue_id refers to an unallocated queue	HSMSG_RC_QNOTALLOC	Windows Linux Mac OS X	pass pass pass
50	HsMsgDestroyInbox while library not initialized	HSMSG_RC_NOT_INIT	Windows Linux Mac OS X	pass pass pass
51	HsMsgDestroyInbox, inbox_id refers to an unallocated inbox	HSMSG_RC_INBOX_NOTALLOC	Windows Linux Mac OS X	pass pass pass