

Table of Contents

Testing of Persistence Configurations Logbook.....	1
Using Store and MSG publisher/MSG consumer python scripts.....	1
Test3 - using bulk instead.....	1
Test4 : Using JDBC in addition to the activemq store.....	3
Test5 : Using Store, messages sent from JMS Java Producer.....	3
Test6 : Network of Brokers : lxb6117 & lxb6118.....	5
Test7 : Multiple Channels, more Ram.....	6
Notes.....	6

Testing of Persistence Configurations Logbook.

Using Store and MSG publisher/MSG consumer python scripts.

<http://activemq.apache.org/amq-message-store.html>

Test Scenarios:

-- Send messages in loop for 2 minutes. Pause 10 minutes. Send random few messages for 10 minutes, pause 10 minutes. Duration: 1 hour. Message Size: random 1,2,5k.

- First observations:
 - ◆ A lot of messages were lost when running the stress cycle. - Actual problem was the limit for open file descriptors: Each script would open a new connection. Rate of closing the connections by the server was lower than the creation of new connections. After ulimit was increased, this problem was stabilized.
 - ◆ Persistence seems to work as long as the server processes the messages into the message store (no messages are lost do the connection failure previously described).



- Increasing the first loop to 10 minutes, a big degradation occurs with the increase on the number of open connections.
- Sent messages 30051 - Received messages: 29449 Lost: ~2%
- Error occurred for a few messages: " ERROR RecoveryListenerAdapter - Message id ID:lxb6118.cern.ch-51583-1204887427969-4:52708:-1:1:1 could not be recovered from the data store! "
- From ActiveMQ: <https://issues.apache.org/activemq/browse/AMQ-1445> Fix on 5.1.0. Perhaps we should consider moving there :S

Test3 - using bulk instead.

Running 2 producers, 1 consumer, messages in bulks of 1000 x 1K, For 3hours. Loop: Send maximum messages for 1 hour, sleep 10 min, send few messages for 3min, sleep 5 min, repeat. producer2 kicks in 40 minutes after the first producer.

A few messages failed: The only traceable errors were

2008-03-11 18:23:00,139 [138.5.237:33191] ERROR Service

- Async error occur

GridMessagingSystemPerformanceTestLogbook < LCG < TWiki

```

2008-03-11 18:23:02,840 [138.5.237:33191] ERROR DataManagerImpl - Looking for key
2008-03-11 18:23:02,840 [138.5.237:33191] ERROR MapContainerImpl - Failed to get va
2008-03-11 18:23:02,941 [138.5.237:33191] ERROR TopicStorePrefetch - Failed to fill b
2008-03-11 18:23:02,941 [138.5.237:33191] ERROR Service - Async error occu
2008-03-11 18:23:09,512 [42.131.89:33644] ERROR DataManagerImpl - Looking for key
2008-03-11 18:23:09,512 [42.131.89:33644] ERROR MapContainerImpl - Failed to get va
2008-03-11 18:23:09,614 [42.131.89:33644] ERROR TopicStorePrefetch - Failed to fill b
2008-03-11 18:23:09,617 [42.131.89:33644] ERROR StoreDurableSubscriberCursor - Failed to get cu
  
```

Already sent a message to activemq users mailing list to see if someone knows if it is an issue. I will try to reproduce it in the meantime. First messages lost on the producer Plxplus225.cern.ch-570 was {179945,179946}:

```

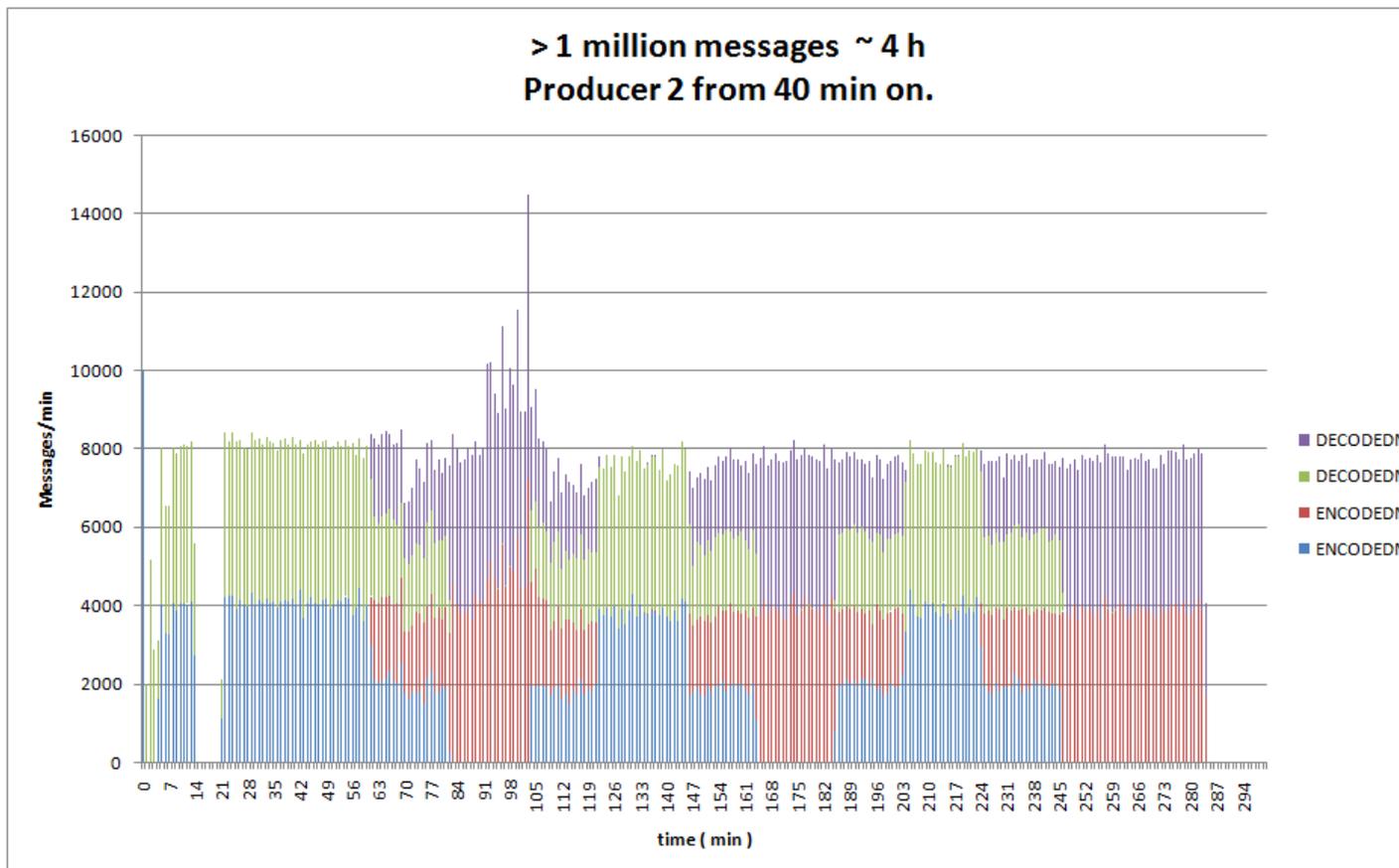
179944 Plxplus225.cern.ch-570 20.5527989 1205252069.339927 1205252089.8927259
179947 Plxplus225.cern.ch-570 20.5527989 1205252069.339927 1205252089.8927259
  
```

in total, 523037 messages were sent, 520206 received. (0,54% lost)

On producer Plxplus236-570 519037 were sent, 516946 received.(0,40% lost) First messages lost: {17543;17544}

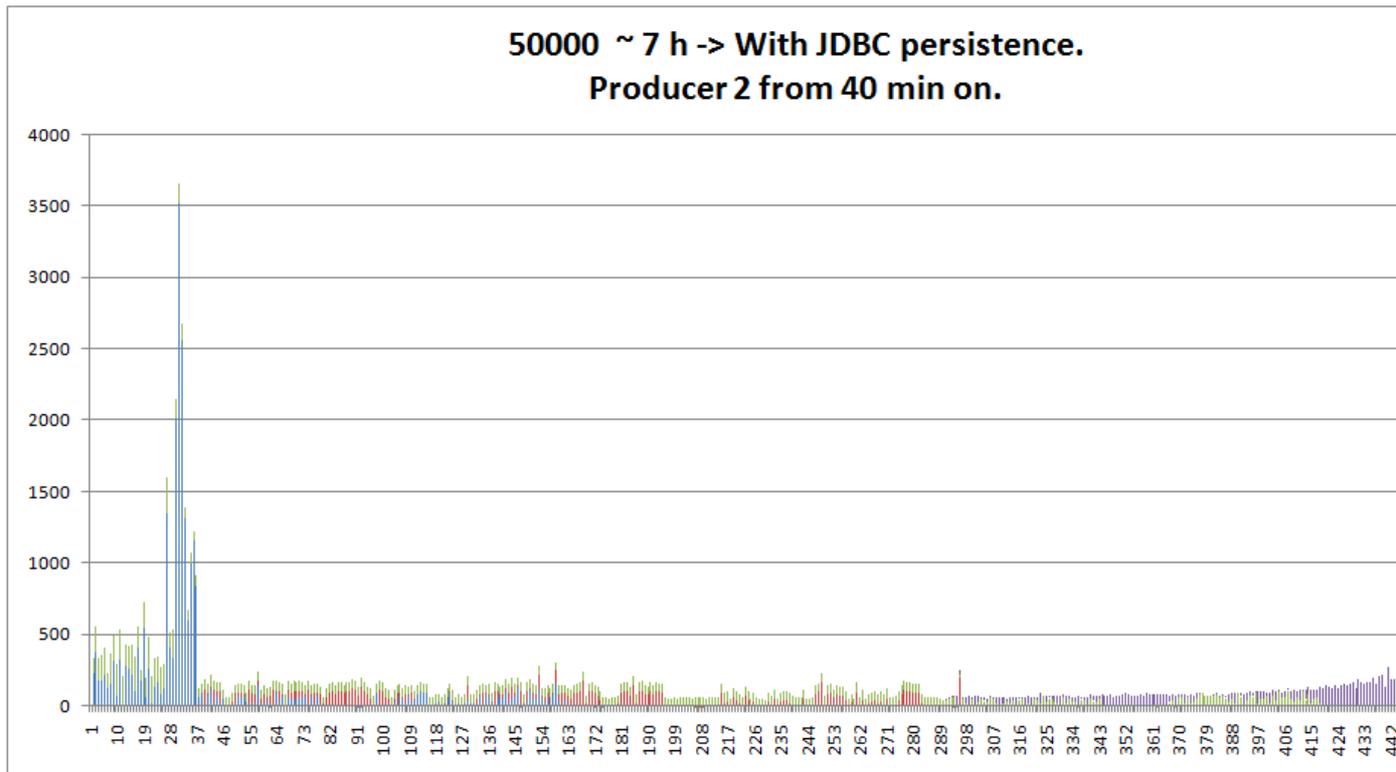
```

17541 Plxplus236.cern.ch-570 20.5527989 1205252069.339927 1205252089.8927259
17542 Plxplus236.cern.ch-570 20.5527989 1205252069.339927 1205252089.8927259
  
```



Test4 : Using JDBC in addition to the activemq store

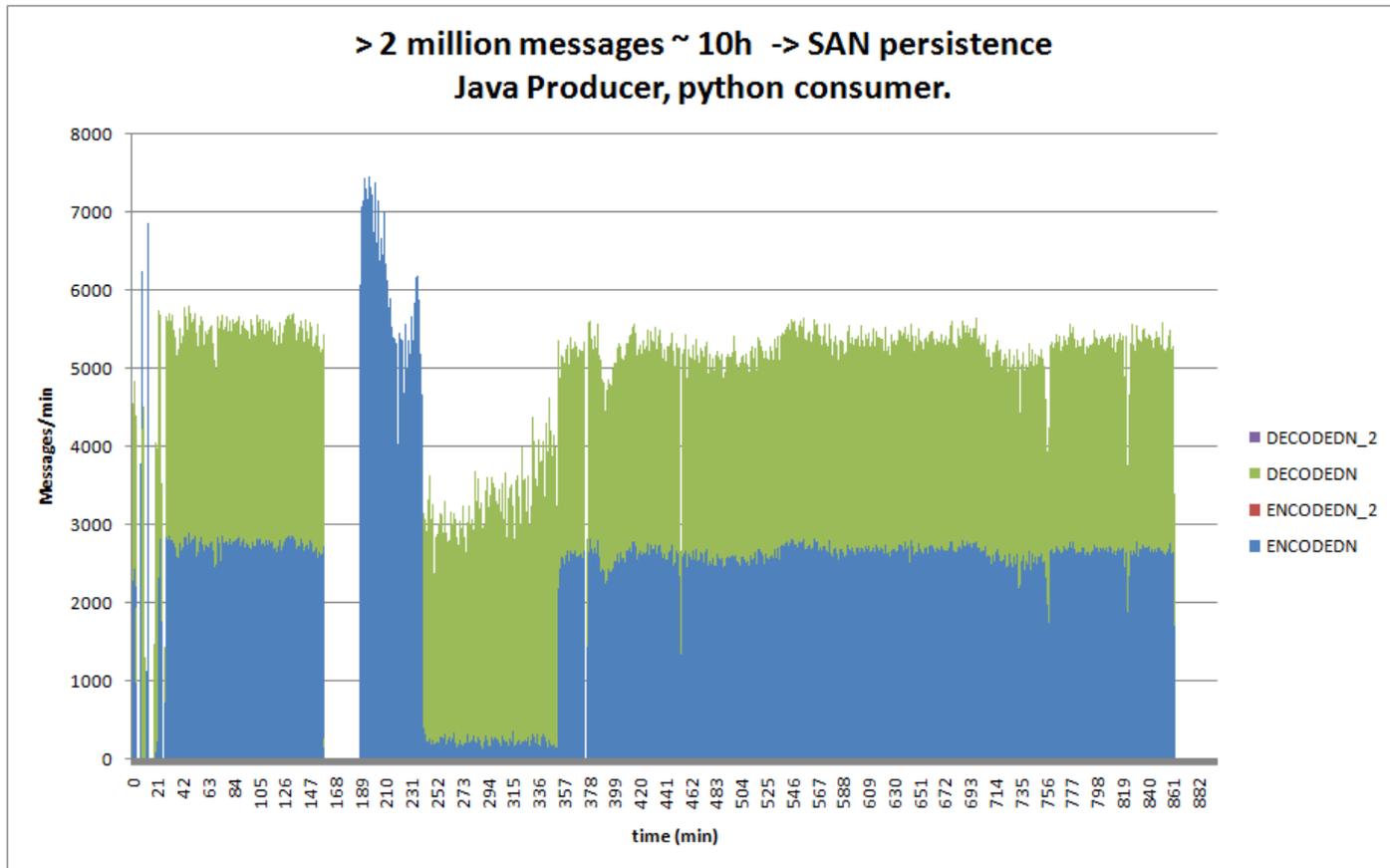
Awfully slow 😞



Test5 : Using Store, messages sent from JMS Java Producer

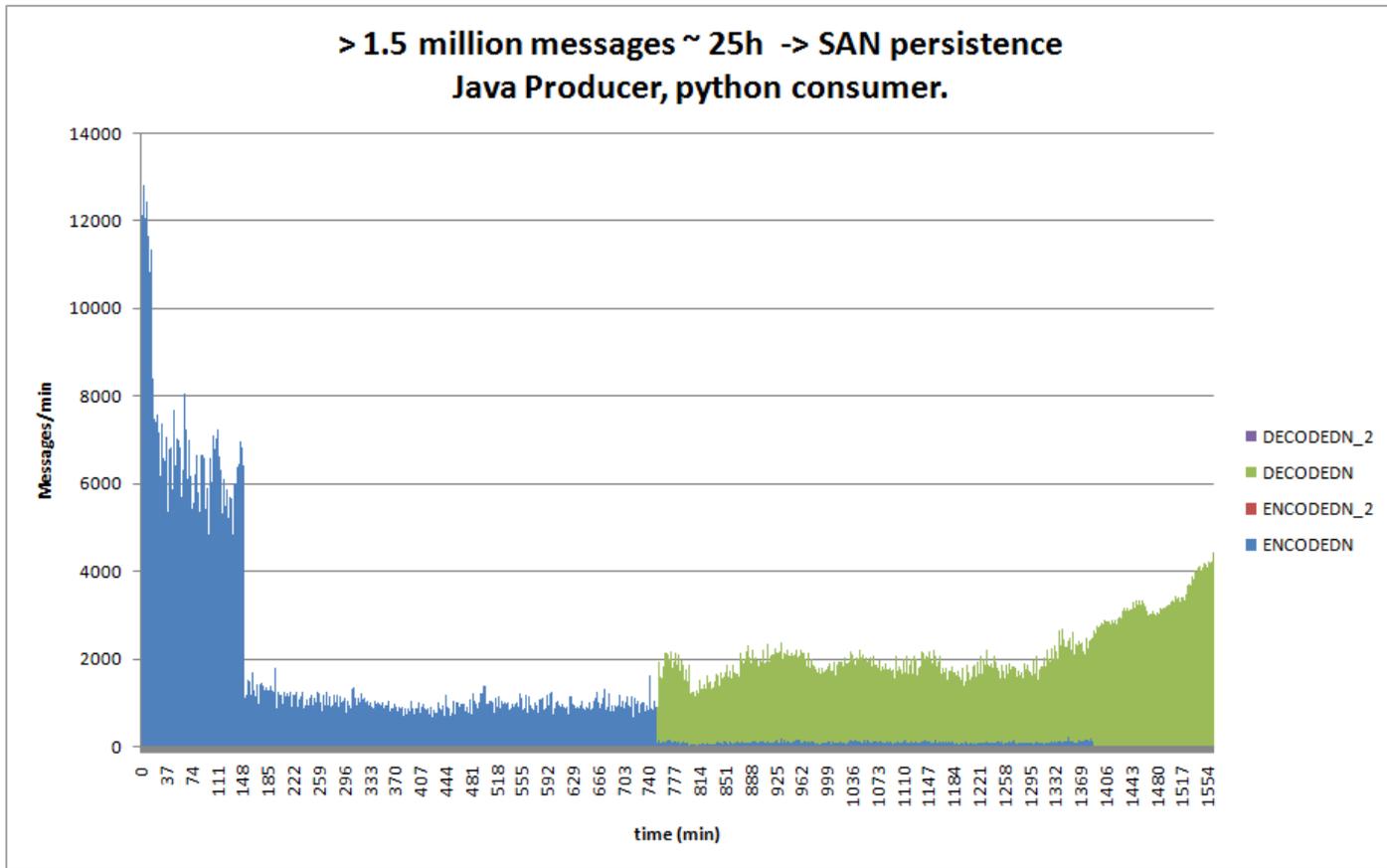
This test aimed at 1) testing if messages sent using different protocols would still be seamless integrated; 2) Try very long runs without consumer (worst case scenario and performance degradation)

First Run, we had 2 hours with consumer and producer active, then a longer run in which for the first hour we produced messages without having consumer, and starting the consumer afterwards. We see the publishing will be limited when we are consuming from the Message Store (up to minute 351). From then on, the system proceeds to its usual behaviour, load balancing consumer and producer.



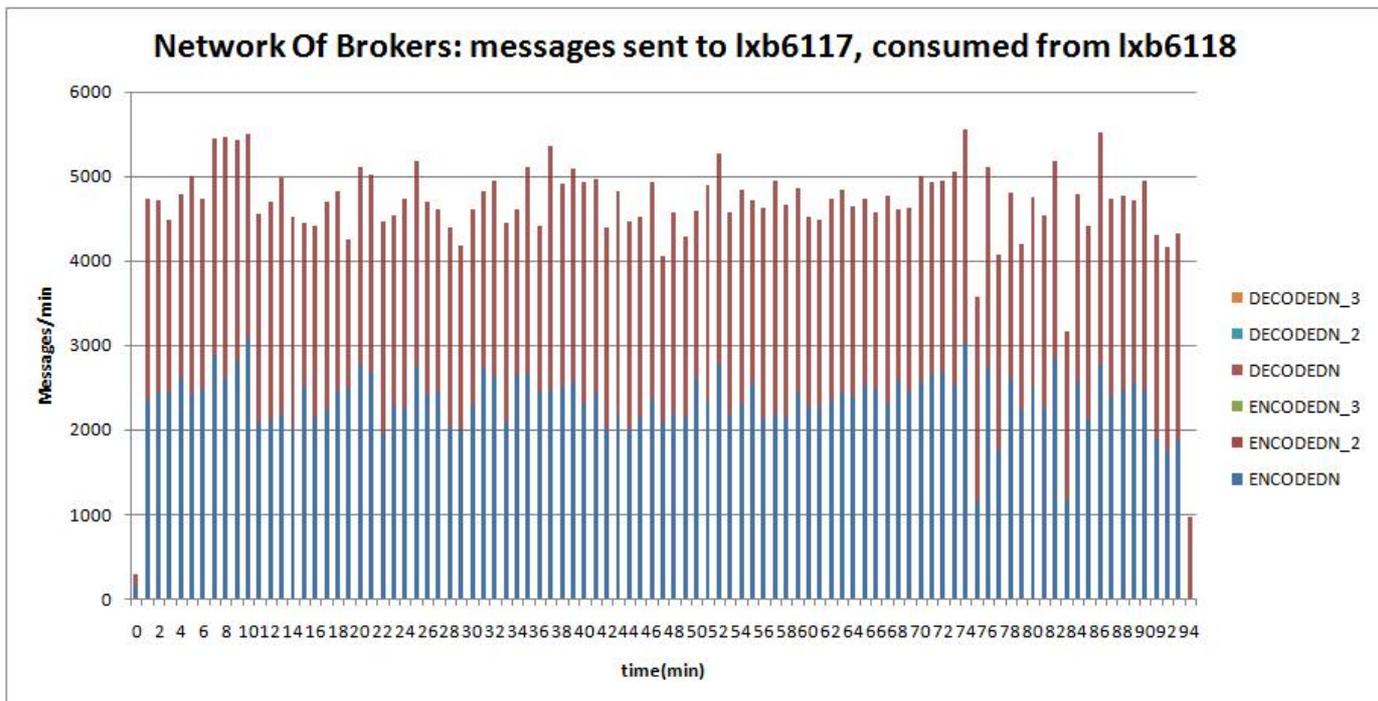
For test run 2 we tried to put even more stress, allowing the producer to accumulate messages up to 12 hours. After having 1 Gb of information on disk (1 million messages, writing performance was greatly reduced, probably due to the configured limitations on message store indexing.)

After 12hours, the consumer was started, and started consuming messages with the same pattern as the previous test run. Unfortunately, producer died with a internal error while there were still ~0.5 million messages indexed. However without the need for load-balancing, we could see the consuming rate increasing along with the reduction of persisted messages in the Message Store.



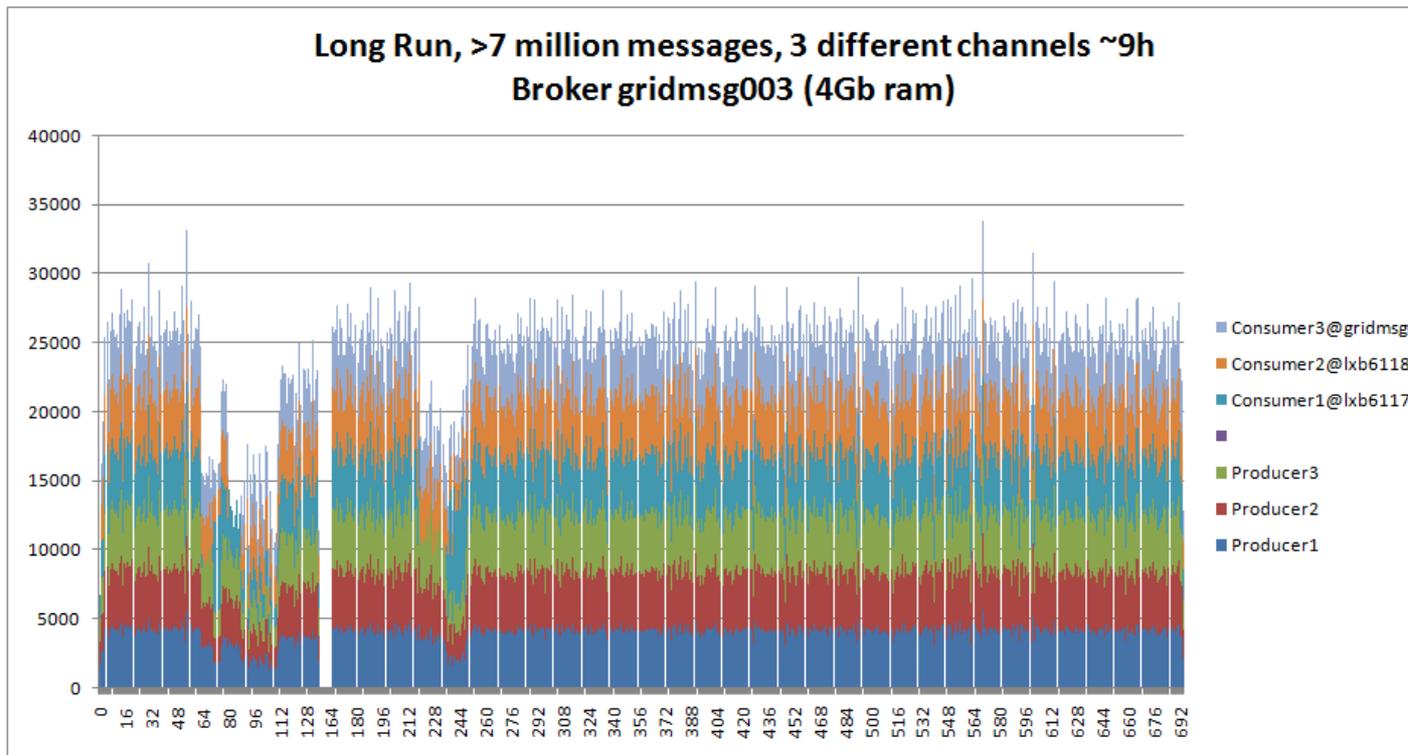
Test6 : Network of Brokers : lxb6117 & lxb6118

A test running a configured Network of Brokers.



Test7 : Multiple Channels, more Ram

It was observed no degradation on using more channels. The duplication of RAM had the greatest impact, allowing a greater throughput without saturation nor the need to resort to file based persistency. Test Run already on version 5.1 Stable release of ActiveMQ.



Notes

http://www.sonicsoftware.com/products/sonicmq/performance_benchmarking/index.ssp

-- DanielRodrigues - 29 Apr 2008

This topic: LCG > GridMessagingSystemPerformanceTestLogbook

Topic revision: r8 - 2008-06-05 - DanielRodrigues



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback