Fiorano
Enabling change at the speed of thought
www.fiorano.com

JMS Performance Comparison

Performance Comparison for Publish Subscribe Messaging

AMERICA'S
Fiorano Software, Inc.
230 S. California Avenue,
Suite 103, Palo Alto, CA 94306
USA
Tel:+1 650 326 1136
Fax: +1 646 607 5875
Toll-Free: +1 800 663 3621
Email: info@fiorano.com

EMEA
Fiorano Software Ltd.
3000 Hillswood Drive
Hillswood Business Park
Chertsey Surrey KT16 0RS UK
Tel: +44 (0) 1932 895005
Fax: +44 (0) 1932 325413
Email: info_uk@fiorano.com

APAC
Fiorano Software Pte. Ltd.
Level 42, Suntec Tower Three
8 Temasek Boulevard
Singapore 038988
Tel: +65 6829 2234
Fax: +65 6826 4015
Email: info_asiapac@fiorano.com

Contents

Executive Summary

This paper presents a performance analysis of publish/subscribe messaging throughput of FioranoMQ® 10.0.0, Tibco EMS v8.0, ActiveMQ 5.9.0, HornetQ 2.4.0, OpenMQ5.0.0, RabbitMQ 3.2.0, and IBM WebSphere MQ 7.5. This analysis provides a head-to-head comparison of these products designed to illustrate the products' relative performance characteristics for several messaging scenarios.

The test scenarios represent stress level conditions for real world applications. The tests examine performance under load, where a single message broker is required to support many publishers and subscribers.

The testing tool used for these performance tests is highly configurable and can be used to test any JMS broker. Also, this tool allows running and measurement of a wide range of test definitions.

Do note that the different configurations or performance tuning of any JMS broker may potentially yield throughput gains (or losses) for any of these tests. Changes to the test definitions will produce different throughput rates and this should be considered when attempting to map these results to expected performance of any particular JMS application.

All the JMS brokers were configured with out-of-the-box default values and no performance specific product tuning was carried out for any of them. It's observed from the detailed results that the relative performance of the message brokers varies under various conditions. While performance analysis should always be conducted for a particular messaging environment, the results of these tests suggest that FioranoMQ will deliver messages more efficiently in demanding messaging environments in today's real-time enterprises.

1. Test Methodology

All the tests described in this section were carried out using a highly configurable testing tool. This tool allows running and measurement of a wide range of test definitions.

This section begins with a brief description of test conditions which are created to test the JMS server. This is followed by a section that describes the tests and their results. The final section provides a brief description of the hardware and software configurations.

1.1 Test Conditions

All the tests were conducted under the following conditions:

- Each client runs on a separate JMS connection.
- All test results are recorded after the client connections have been established and publishers/subscribers and other objects had been created.
- All tests were run multiple times to assure repeatability.

- Performance was measured under maximum load by publishing as many messages as possible using default settings of the server.
- During the test, no other applications were running and using resources on the system under test.
- Dups_ok was used by all consumers.
- All servers were tested in the default mode - which meant running IBM MQ, Tibco EMS in "Evaluation" (non-HA) mode, ActiveMQ 5.9 (default configuration mode), FioranoMQ and others in normal production ready (non-HA) mode.

1.2 Test Scenario's

The tests were conducted for the most popular messaging models employed using Topics in JMS.

Non-Persistent Publishers & Non-Durable Subscribers

This model is typically used by applications which are exchanging high volume of messages and have a requirement of minimum latency.

Persistent Publishers & Durable Subscribers

This model is typically employed by applications which need maximum level of redundancy and need once and only once guarantee of message delivery irrespective of the client or server failure.

The following tests were conducted based on typical customer use-cases:

a. **Topic Scalability Tests**: These tests observe the performance characteristics of JMS server with varying # of Pub/Sub clients on a fixed number of topics. The results illustrate the scalability of JMS server as more clients (all working on same JMS Topic) are employed.

b. **Server Scalability Tests**: These tests observe the performance characteristics of JMS server with varying # of Topics with fixed # of Pub/Sub clients per topic. The results illustrate the scalability of JMS server as more clients (each working on independent JMS Topics) are employed.

c. **Persistent Producer, Multiple Durable Consumers**: These tests observe the performance characteristics of JMS server when a single persistent publisher is used to publish messages to multiple durable subscribers.

d. **Non-Persistent Producer, Multiple Non-Durable Consumers**: These tests observe the performance characteristics of JMS server when a single non-persistent publisher is used to publish messages to multiple non-durable subscribers.

In order to generate the highest amount of message load, no processing time is introduced at either side of the client message exchanges. Allowing publishers to send messages as fast as possible in this manner enables these tests to expose the maximum message throughput rates. The test message size was chosen to reflect use cases observed in typical customer proof of concept scenarios.

1.3 Test Duration

All test scenarios were executed for a total of five minutes. Each test execution comprised of five, sixty-second intervals. The first two and last intervals were considered **ramp-up** and **ramp-down** intervals, respectively.

Ramp-up intervals are times during which the systems are increasing their message handling capacities, typically via resource allocation in response to the newly introduced client load.

Ramp-down intervals are times in which the systems are decreasing their capacity in response to decreased client loads that result from test completion. The remaining five intervals were considered **measurement** intervals during which steady-state performance was achieved.

Steady-state is the condition in which message rates exhibit negligible change.

1.4 Environment Setup

All client connections, publishers and subscribers were established before any testing ramp-up periods were started.

Each product's message store, log files, queues, and topics were deleted and recreated therefore the broker stopped and restarted between each test.

1.5 Measurement

Performance data was collected during the five-minute measurement intervals only. No data was collected during ramp-up and ramp-down intervals. Tests were run twice, and measurements were averaged to obtain final results.

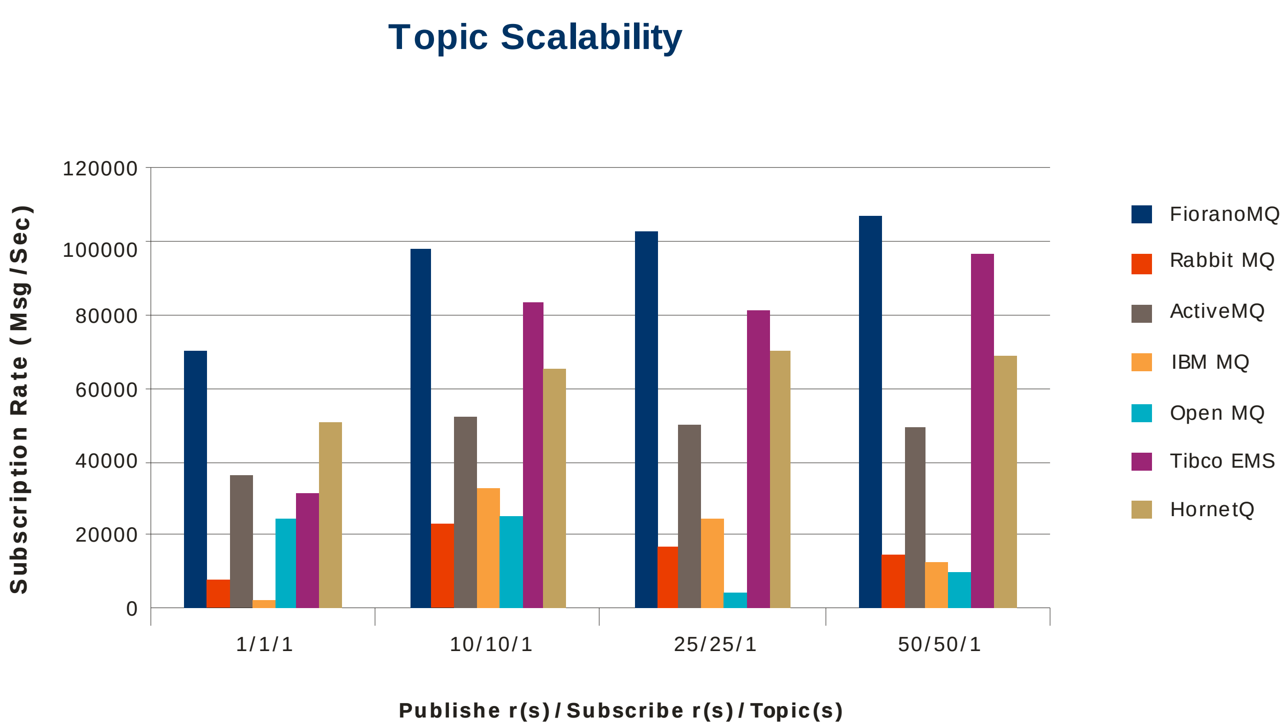
1.6 Topology

The topology contains two machines: One for running the clients and the other for running the server. The system configurations are detailed later in this document. These systems having 1GB NIC cards were interconnected using a 1 GBPS peer to peer connection.

2. Performance Results

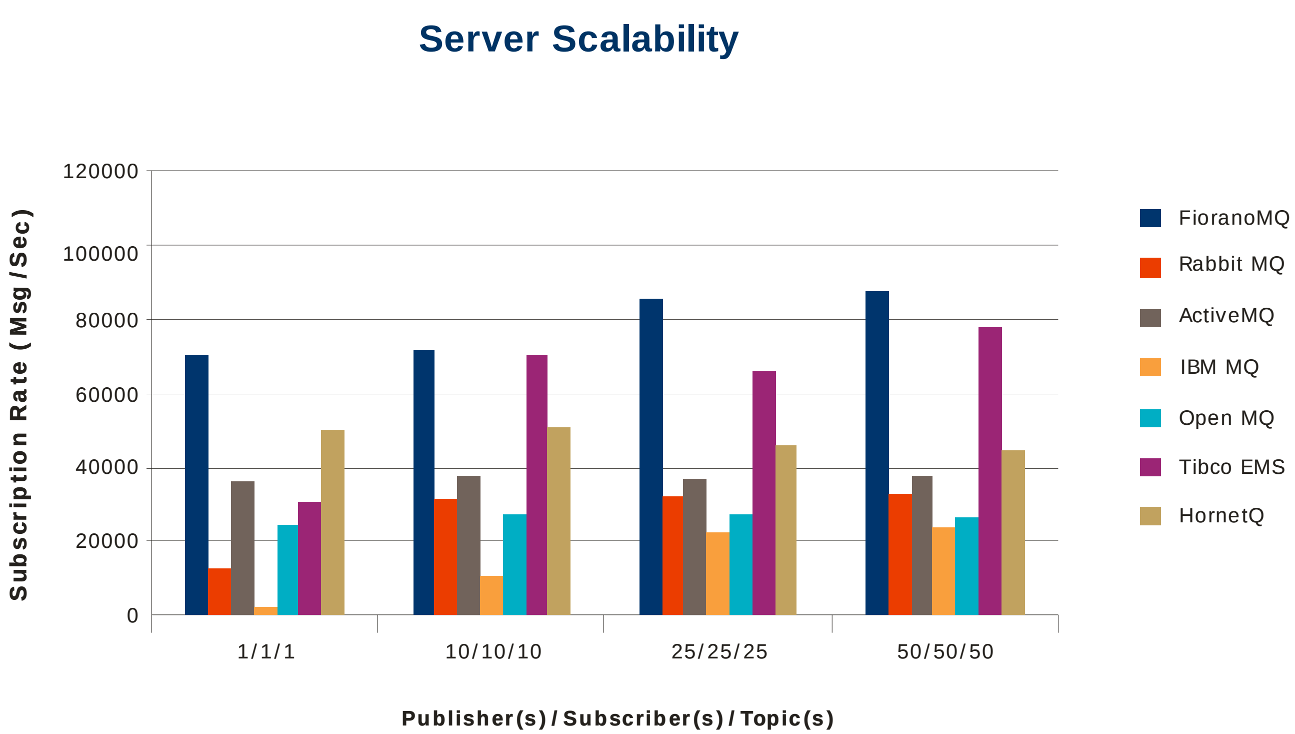
2.1 Topic Scalability

P/S/T	Message Type	Subscriber Type	Message Size (bytes)	Subscription Rate (messages / sec)						
				Fiorano MQ 10	Rabbit MQ 3.2.0	Active MQ 5.9.0	IBM MQ 7.5.0_2	Open MQ 5.0.0	Tibco EMS 8.0	HornetQ 2.4.0
1/1/1	Non-Persistent	Non-Durable	1024	67876	14298	39137	3176	23476	31905	52581
10/10/1	Non-Persistent	Non-Durable	1024	99726	21034	53406	37771	23592	81720	66381
25/25/1	Non-Persistent	Non-Durable	1024	103094	18609	51212	22488	3930	80813	65784
50/50/1	Non-Persistent	Non-Durable	1024	103660	16635	49440	16111	9146	98191	69741



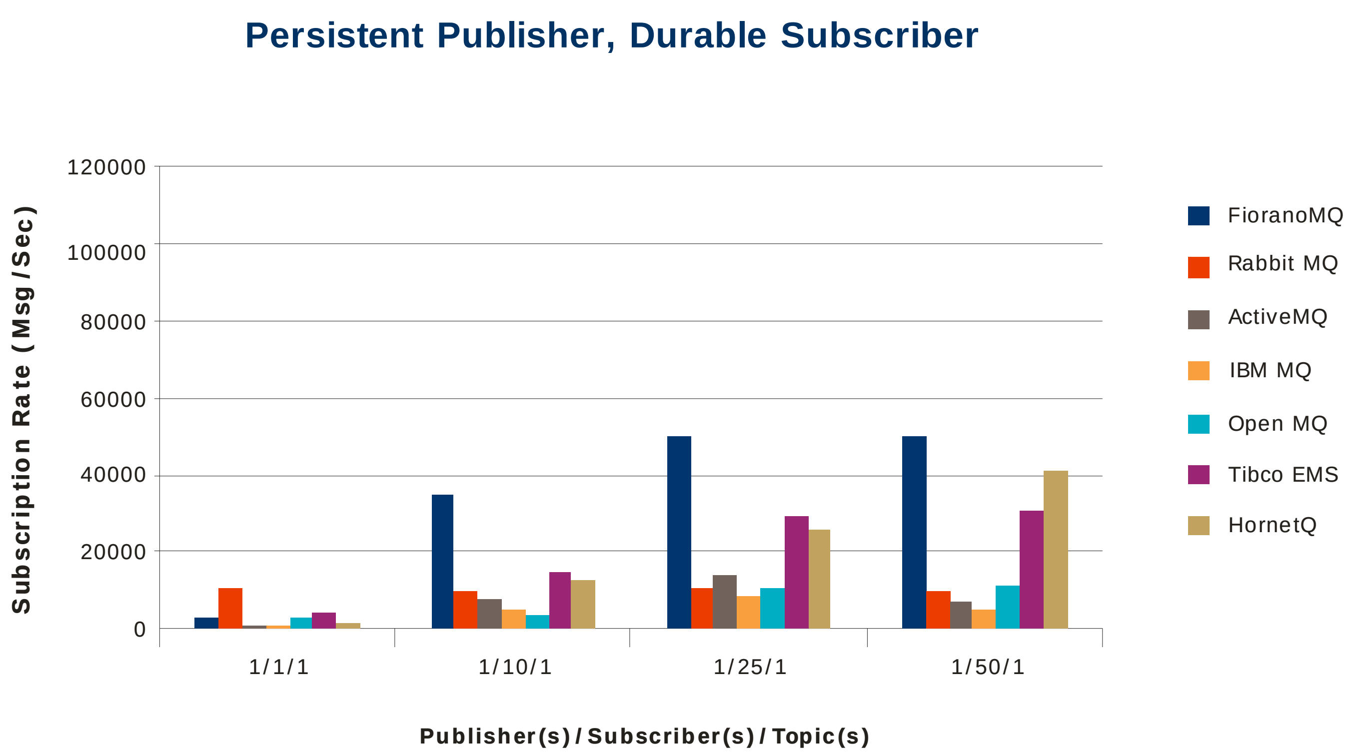
2.2 Server Scalability

P/S/T	Message Type	Subscriber Type	Message Size (bytes)	Subscription Rate (messages / sec)						
				Fiorano MQ 10	Rabbit MQ 3.2.0	Active MQ 5.9.0	IBM MQ 7.5.0_2	Open MQ 5.0.0	Tibco EMS 8.0	HornetQ 2.4.0
1/1/1	Non-Persistent	Non-Durable	1024	67876	14298	39137	3176	23476	31905	52581
10/10/10	Non-Persistent	Non-Durable	1024	76298	37148	39298	12165	26721	67035	49459
25/25/25	Non-Persistent	Non-Durable	1024	83270	35528	38062	21740	26923	64825	45805
50/50/50	Non-Persistent	Non-Durable	1024	84330	35780	39179	22827	24911	79165	43033



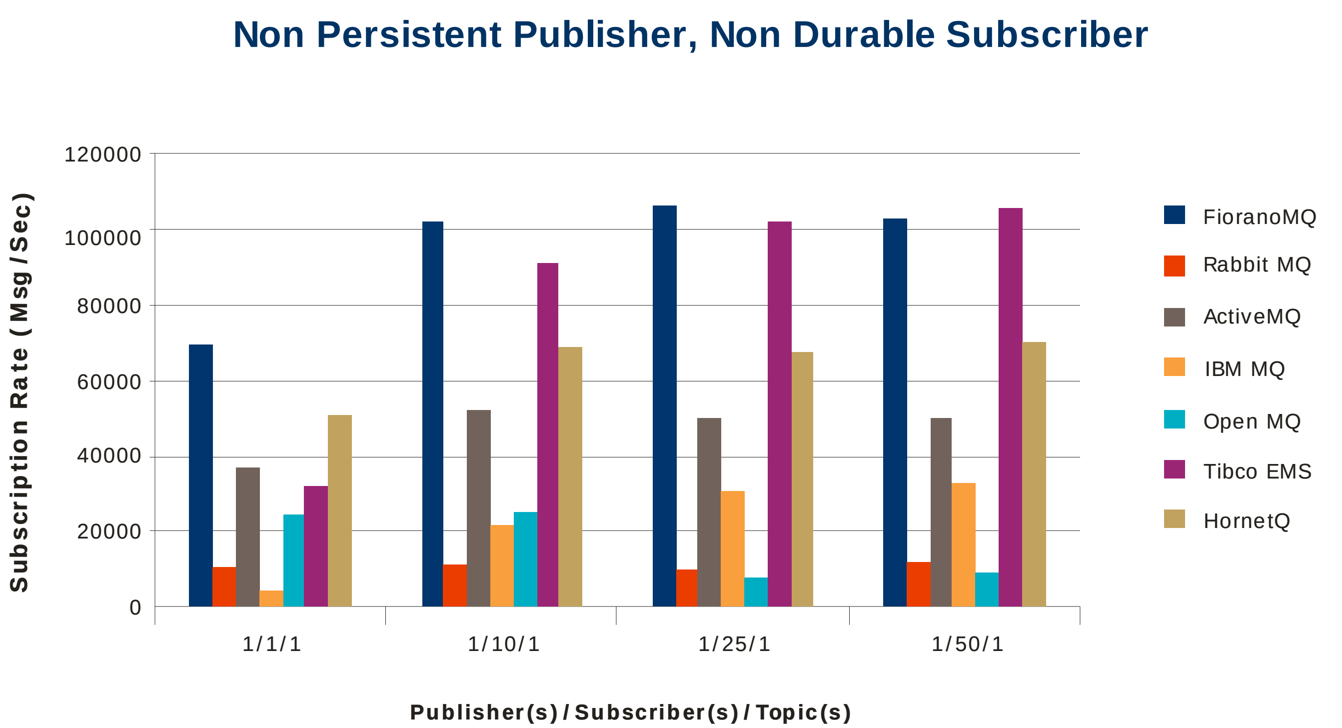
2.3 Persistent Publisher, Durable Subscribers

P/S/T	Message Type	Subscriber Type	Message Size (bytes)	Subscription Rate (messages / sec)						
				Fiorano MQ 10	Rabbit MQ 3.2.0	Active MQ 5.9.0	IBM MQ 7.5.0_2	Open MQ 5.0.0	Tibco EMS 8.0	HornetQ 2.4.0
1/1/1	Persistent	Durable	1024	3818	9152	1192	1097	3375	4195	1767
1/10/1	Persistent	Durable	1024	33894	9831	8206	6025	6005	16785	14571
1/25/1	Persistent	Durable	1024	47448	9774	13383	8341	9482	28003	26941
1/50/1	Persistent	Durable	1024	50519	9734	7416	5609	9071	33804	40180



2.4 Non-Persistent Publisher, Non-Durable Subscribers

P/S/T	Message Type	Subscriber Type	Message Size (bytes)	Subscription Rate (messages / sec)						
				Fiorano MQ 10	Rabbit MQ 3.2.0	Active MQ 5.9.0	IBM MQ 7.5.0_2	Open MQ 5.0.0	Tibco EMS 8.0	HornetQ 2.4.0
1/1/1	Non Persistent	Non Durable	1024	67876	14298	39137	3176	23476	31905	52581
1/10/1	Non Persistent	Non Durable	1024	101096	13676	52868	20059	23307	93083	68741
1/25/1	Non Persistent	Non Durable	1024	104637	13578	52148	29941	6103	101425	66163
1/50/1	Non Persistent	Non Durable	1024	101853	13491	49046	35905	9071	102432	71002



3. System Configuration

3.1 Hardware Configuration

Server System

- Linux CentOS 2.6.18-92.el5 (x64)
- 2 Quad Core Intel(R) Xeon(R) CPU 5405 @ 2.00GHz
- 64 bit 16 GB RAM

Client System

- Linux CentOS 2.6.18-92.el5 (x64)
- Quad Core Intel(R) Xeon(R) CPU 5405 @ 2.00GHz
- 64 bit 16 GB RAM

Network Settings

- Client and Server were on the same network
- Network Speed: 1GBPS

3.2 Software Configuration

- Java Runtime Environment, Standard Edition (build 1.7.0_45-b18)
- FioranoMQ v 10.0.0
- RabbitMQ 3.2.0
- Tibco EMS v 8.0 (In persistent tests, the TIBCO topics were set to failsafe to ensure persistence to disk)
- ActiveMQ v 5.9.0
- HornetQ 2.4.0
- OpenMQ 5.0.0
- IBM WebSphere 7.5

Fiorano
Enabling change at the speed of thought

About Fiorano Software

Founded in 1995, Silicon Valley based Fiorano is a California Corporation with proven leadership in enterprise middleware and peer-to-peer distributed systems. Fiorano's innovative event-driven, dataflow SOA platform integrates applications and complex technologies into an enterprise nervous system, increases business process performance, yields higher message throughput and enhances availability through agent-based visual composition that bridges the capability gap between business models and their implementation – the model is the application, ready to run.

Global leaders including ABN AMRO, Boeing, British Telecom, Capgemini Telecom, Chicago Mercantile Exchange Group, McKesson, NASA, POSCO Steel, Qwest Communications, Rabobank, Schlumberger, Lockheed Martin, United States Coast Guard and Vodafone have deployed Fiorano to drive innovation through open, standards-based, dataflow SOA applications built in just days, yielding unprecedented productivity.

The Fiorano SOA Platform built on the Fiorano Enterprise Service Bus (ESB) and Fiorano Message Queue (MQ), together deliver the industry fastest, lowest latency, highest throughput real-time messaging (asynchronous and synchronous) to power high performance, highly available, and collaborative workflow applications whose application services are distributed throughout the IT landscape. Fiorano's distributed, peer-to-peer agents abstract complexity of developing and deploying services to unlock value in a customer's enterprise architecture framework.

