Todo list

Add document to version control

PERFORMANCE OF HETEROGENOUS NETWORKS OF AMQP1.0 COMPLIANT MESSAGE-ORIENTATED MIDDLEWARE (MOM) IMPLEMENTATIONS

By

David Joe Wade

A Dissertation Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

Major Subject: COMPUTER SCIENCE

Examining Committee:

____

Dr. Eugene Eberbach, Dissertation Adviser

Rensselaer Polytechnic Institute
Troy, New York

April 2014
(For Graduation May 2014)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

Special thanks to my wife Anne, my son Jack and my daughter Erin - sorry for spending so many weekends away putting this together. Thanks to my parents, Brian and Catherine.

ABSTRACT

The Advanced Message Queuing Protocol (AMQP) provides an open standard application layer protocol for message orientated middleware (MOM). There are many MOM implementations that meet the AMQP standard making it possible to integrate disassociated systems without having to standardize MOM implementations across different teams. This paper provides benchmarking information about performance impact of integrating different AMQP implementations.

reword

1. INTRODUCTION

1.1 AMQP History

1.1.1 Message Oriented Middleware (MOM)

Message-orientated middleware (MOM) is software that facilitates the development of distributed systems by providing an infrastructure for asynchronously sending and receiving data over a network [?]. There are many implementations of such messaging systems, but until recently most of them have been closed source/proprietary systems. Lacking any standard for communications protocols software, developers and systems engineers were often forced to choose (or have chosen) a single MOM implementation and become stuck with its features and limitations.

1.1.2 Java Messaging Service (JMS)

The Java EE JMS (Java Messaging Service) was the starting point for attempting to standardize MOM. Unfortunately, JMS only specifies an application programming interface (API) and does not specify a format for exchanged messages so it does not create interoperability. To achieve this desired interoperability other standards such as AMQP (Advanced Message Queuing Protocol) built upon the JMS standards).

1.1.3 AMQP founder

The development of AMQP was started in 2003 by John O'Hara and others at JPMorgan. JPMorgan was looking for a messaging solution with high durability that supported a very high number of small message transactions (in the range of 500,000 messages/second) [?]. The AMQP standard also defines message orientation, queuing, routing, reliability, and security. AMQP also describes the wire-level format of the data so that any nodes that are AMQP compliant can handle the message stream.

Standardization lead to the rapid development of many MOM systems such as: Apache Qpid, RabbitMQ, StormMQ, ActiveMQ, Apache Apollo and Microsofts

Azure Service Bus. MOM implementations that support the AMQP version 1.0 specification (Released October 29th 2012) [?] are interoperable and are available on many platforms. Because these implementations were designed around the message traffic of the financial services sector they excel at passing small messages and focus on message throughput. There are performance differences between each implementation and generic benchmarks for each implementation are available [?][?].

2. Methodology

Marsh, Sampat, Potluri, Panda in [3] developed broker network architectures and benchmarked to support large federated broker networks and benchmarked the messaging passing performance. They experimented with multi-level broker architectures and multi-consumer fanout patterns to find the optimal broker design patterns. All of their work was performed with Apache Qpid implementation [3].

Subramoni, Marsh, Narravula, Lai and Panda in [1] designed and developed benchmarks for MOM broker networks building using the Message Passing Interface (MPI) benchmarks developed by Ohio State University Network-Based Computing Laboratorys [1] and the Standard Performance Evaluation Corporation (SPEC) MOM benchmarks [1]. Benchmarks such as the Direct Exchange - Single Publisher Single Consumer (DE-SPSC) and Direct Exchange - Multiple Publishers Multiple Consumers (DE-MPMC) benchmarks [1] will be implemented for each AMQP implementation

This project will combine the methodologies used in these experiments and add an additional variable, different broker implementations to determine what the performance impact of a heterogenous broker network on messaging performance.

In addition to these benchmarks, additional information will be collected in order to compare the performance of the broker implementations.

A lab environment will be setup to isolate the tests from outside message traffic. High-end workstation, class laptop computers and business class network equipment will be used. The computers will use the latest Long-Term Support version of the Ubuntu Server operating system (12.04LTS) and will be configured to minimize background daemons and processes. The times of all systems will be synchronized using a local NTP server. A complete listing of software versions used will be published with the results of this research.

Missing figure

something

3. THE NEXT CHAPTER

And so on, for more chapters. Another citation for the bibliography:[2]

This is a sentence to take up space and look like text. This is a sentence to take up space and look like text. Please refer to Figure 3.1.

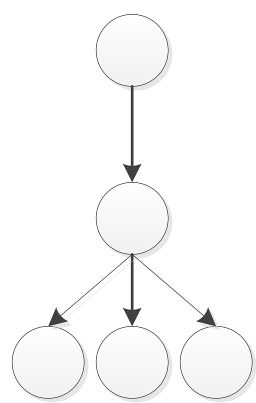

Figure 3.1: This is the Caption for Figure 1 make it long to illustrate how it looks when wrapped around to the next line

Table 3.1: This is the Caption for Table 2

Here's	another	example
of	something	useful
`table`	environment	command.

LITERATURE CITED

[1] This is the first item in the Bibliography. Let's make it very long so it takes more than one line. Let's make it very long so it takes more than one line.

[2] The second item in the Bibliography.

[3] B. Snyder, D. Bosanac, R. Davies, "ActiveMQ in Action", Manning, 2011 pp 1 326.

[4] G. Marsh, A. Sampat, S. Potluri, D. Panda, "Scaling Advanced Message Queueing Protocol (AMQP) Architecture with Broker Federation and InfiniBand", Department of Computer Science and Engineering, The Ohio State University

[5] H Chirino, "STOMP Messaging Benchmarks: ActiveMQ vs Apollo vs HornetQ vs RabbitMQ", http://hiramchirino.com/blog/2011/12/stomp-messaging-benchmarks-activemq-vs-apollo-vs-hornetq-vs-rabbitmq/

[6] K. Sachs, K. Samuel, and S. Appel, "Benchmarking of Message-Oriented Middleware", ACM, pp. 12, Sep. 2009.

[7] H. Subramoni, G. Marsh, S. Narravula, P. Lai, D. Panda, "Design and Evaluation for Financial Applications using Advanced Message Queuing Protcol (AMQP) over InfiniBand", Department of Computer Science and Engineering, The Ohio State University.

[8] P. Maheshwari and M. Pang, Benchmarking Message-Oriented Middleware TIB/RV vs. SonicMQ, Journal Concurrency and Computation: Practice and Experience - Foundations of Middleware Technologies, vol. 17, no. 12, pp. 15071526, Oct. 2005.

NOTE: Printed copies of all sources are available upon request.

Not sure if I really want to advertise this - how on Earth am I going to actually sup-

APPENDIX A
THIS IS AN APPENDIX

Note the numbering of the chapter heading is changed. This is a sentence to take up space and look like text.

A.1 A Section Heading

This is how equations are numbered in an appendix:

$$x^2 + y^2 = z^2 \tag{A.1}$$

APPENDIX B
THIS IS ANOTHER APPENDIX

This is a sentence to take up space and look like text.