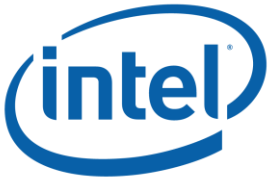

Report: High Performance Trading - FIX Messaging Testing for Low Latency

Abstract:

FIX is the de-facto standard protocol used extensively for electronic communication between buy and sell-side and execution venues, where the performance requirements of algorithmic and high frequency trading are extreme and or the benefits of STP (straight through processing) are sought from electronic connectivity.

January 2012

WITH THANKS TO THE TEAM AT INTEL FASTERLAB UK

*Contact can be made to **FIX@onx.com**.*

TABLE OF CONTENTS

1. Summary

This briefing paper reports on the activity of a consortium of leading IT vendors that have joined forces to create demonstrable high performance solution stacks to address common business requirements in financial trading. The initial focus of the consortium is on a reference-able technology stack of products and services to support FIX protocol communication functions. The paper describes the test environment, documents a set of benchmark tests performed on both commercial and open source FIX engine offerings, and details and interprets the representative latency and throughput figures achieved.

The objective is to create transparency in and capability around comparing performance statistics for key functions along the trading life cycle. The tests used business workloads and were deliberately aligned to reflect the market's current interest in the measurement of interparty latency across the trade life cycle – using FIX formatted messages for defined legs.

An on-going objective is to provide the market with useful data in order to support decisions in technology investment. Therefore, a range of technologies and application software has been addressed. Approaches were made to a number of application vendors with the ultimate agreement to test FIX engines covering both C++ and Java implementations from EPAM Systems' B2BITS unit and Rapid Addition, respectively. As a datum point for comparison, the open source QuickFIX, in both its C++ and Java variants was used.

OnX Enterprise Solutions Ltd is leading a consortium whose charter members include Intel, Dell, Arista Networks and Solarflare Communications, with additional services provided by Edge Technology Group, GreySpark Partners and Equinix. The foundation objective is to create transparent comparative performance statistics for key functions along the trading life cycles using business workloads – FIX being used on a number of legs of the typical trade life cycle.

A series of tests were undertaken that demonstrate the value of commercial software (versus open source) and use of specialist technologies in a low latency infrastructure. The consortium approach recognises the reality that the creation of high performance solutions requires the interaction of many leading edge technologies and the integration of components from several vendors. These parties must work together in order to specify correct parts and then to tune them together such that a complete and reliable solution is available through a collective single channel.

Results for the tests showed that both B2BITS and Rapid Addition's commercial FIX engines out performed the open source QuickFIX offerings (C++ and Java) in a range of tests, being between 4 and 16 times faster in generating messages during a standardised simulated trade. The average latency for the commercial engines was 11 to 12 microseconds, whereas the open source engines were between 45 and 180 microseconds. The variation in results was equally stark, since the frequency distribution results from the commercial engines were bell curved but the open source results had a long fat tail. This indicated the commercial solutions significantly reduced the effect of network jitter and with it the undesired variance of performance.

B2BITS FIX Antenna engine was a C++ version; Rapid Addition's Cheetah engine was Java. Both demonstrated similar performance characteristics over a range of tests and workloads. The similarity of results between the commercial C++ and Java engines stood in contrast to the open source equivalents, demonstrating that Java can perform as well as C++ code when implemented in an optimised fashion.

2. Introduction

In the online and Co-Lo based financial trading markets, performance, both in terms of latency and throughput is paramount. It is the difference between a firm being 'in the market' or not. Complete trading systems are built from many complex elements, including market data capture, trading algorithms, trade execution, and in-flow risk analysis. These elements run on critical infrastructure components, hardware, software, network and connectivity all of which must interoperate with each other.

Today, there is a lack of industry-recognised benchmarks for designers, which can demonstrate solutions have 'high performance' characteristics. To achieve performance and agility, with low up-front and on-going operating costs, trade infrastructure implementation teams need to source the best available components from different innovative specialist vendors, integrate them and tune their interoperability.

FIX is the de-facto standard protocol used extensively for electronic communication between buy and sell-side and execution venues, where the performance requirements of algorithmic and high frequency trading are extreme and or the benefits of STP (straight through processing) are sought from electronic connectivity.

2.1 Purpose

FIX message generation is an increasingly important leg in automated trading and can be a source of significant latency and jitter which can adversely impact the success of business and trading strategies. As trading strategies require access to a greater diversity of execution venues, communication over the standard FIX protocol is more cost effective than accessing markets via diverse proprietary protocols at the various venues.

Infrastructure deployment teams have to select appropriate components, integrating them, commissioning them, deploying them for maximum performance, which can be an extreme challenge, It requires a combination of knowledge, skills, experience and deployment ability that is today scarce and expensive in the market.

The testing undertaken by OnX in the Intel lab with support from the consortium was to investigate these assertions:

1. Using commercial FIX engines would achieve lower latency and less jitter.

2. Using specialist low latency network techniques would have a significant impact on latency.

Full results for each environment and latency improvement are available on request.

2.2 Roles and Responsibilities

OnX Enterprise Solutions, as a "solution facilitator" led a collaborative approach through the creation of a consortium of IT vendors focused on the creation of high performance infrastructure designs specifically for financial trading systems. OnX consultants provided input into the hardware selection, conduct of the tests and post-test analysis.

The benchmarks were conducted at Intel's fasterLAB in the UK. Intel engineers screened hardware and software performance for optimization. Intel engineers performed the tests, recorded the results and provided post test process to produce average tables and graph outputs.

Software suppliers Rapid Addition and B2BITS EPAM Systems provided their FIX engines. The test harness was designed by Rapid Addition and the open source implementations in Java and C++ were supplied by Rapid Addition and B2BITS respectively.

2.2.1 Consortium Members

A number of technology and services providers have invested as charter members of the consortium. However, the initiative is open, and further participant members may be added in the future. Between them, these members provide a complete infrastructure capability and created the reference architecture, each drawing on specific expertise while OnX provided the integration and build capability.

The charter group members directly involved in building the initial technology stack and in the performance benchmark testing comprise:

Lead:
OnX Enterprise Solutions – Product procurement and architecture design

Infrastructure component providers:
Arista Networks – Network Switch
Dell – X86 Servers
Intel – Intel® Xeon® processors, and lab environment
Solarflare Communications – Network Interface Card

Implementation and deployment Services:
Edge Technology Group – Buy-side solutions
Equinix – Trading ecosystem hosting
GreySpark Partners – Capital Markets business, management and Technology consulting services

Applications under test:
Rapid Addition – FIX engine
B2BITS EPAM – FIX engine
QuickFIX – Open Source FIX engine

2.3 Conduct and Presentation

Tests were performed by Intel engineers and preliminary results shared with the software suppliers who were then given an opportunity to optimize their code. A second round of testing was then conducted, the results of which were used in the preparation of this paper.

The software houses had access to the test harness prior to testing, in order to agree and finalize the methodology – but no access or amendment was allowed during the test runs. All results were captured by Intel and shared with OnX. Only results from Rapid Addition were shared with Rapid Addition and likewise the results from B2BITS EPAM were shared only with B2BITS EPAM.

3. Method

3.1 Test Harness – Software Design

The test harness used to perform the benchmarks was designed by Rapid Addition (audited by B2BITS EPAM) with implementations for C++ and Java written by B2BITS EPAM and Rapid Addition, respectively.

The test software was implemented across the two servers. One ran simulators for a market data (MD) feed and an execution venue (EV), the other represented a typical software implementation of a real life algorithmic trading system – with all applications run on a single server to minimise latency, and through the 'in-process' linkage of the algorithmic application logic and the FIX engine under test.

The tests measured recognisable legs in the trading life cycle, mapping to real life workflow scenarios and researching current industry interest in the measurement of interparty latency over discrete legs of a trading cycle.

The very simple logic of the simulated algorithmic trading component minimises latency and jitter, so allowing the focus of the benchmark to be on the FIX engines themselves.

The benchmarking on the FIX engines focused on their ability (a) to process both FIX-formatted market data and (b) order processing messages for 2 defined stages in the trade cycle at different throughput rates, over both burst and prolonged periods.

The companies under scrutiny were given controlled access to the test rig with the ability to run tests, analyse results, tune and re-test. This activity was supported by skilled Intel engineers, who were also available to assist the companies optimising their code for the target hardware stack.

The diagram below illustrates the test harness with its simulated market data and execution venue.

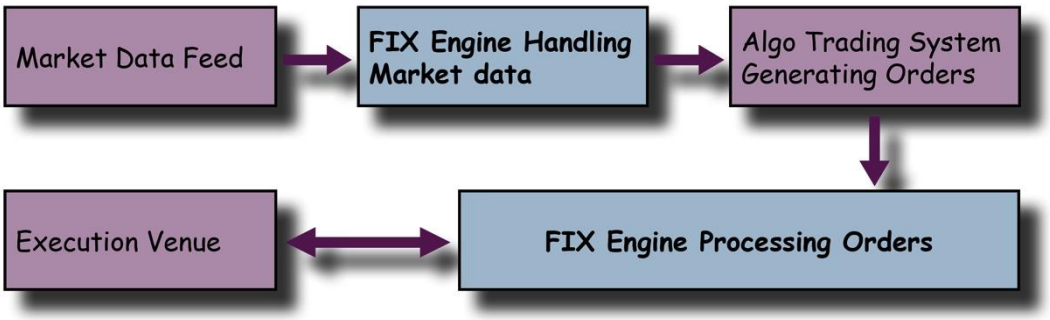

Figure 1: Test Harness Overview

3.2 Test Harness - Hardware Design

In order to conduct benchmark tests on the FIX engines, the reference architecture was specified and built by OnX at the fasterLAB in the UK. OnX also analysed and interpreted the benchmarks, and provided an independent audit of the test activities by the FIX engine vendors. These vendors accessed the test rig via remote access under pre-approved and agreed conditions. The main components of the reference architecture are shown below:

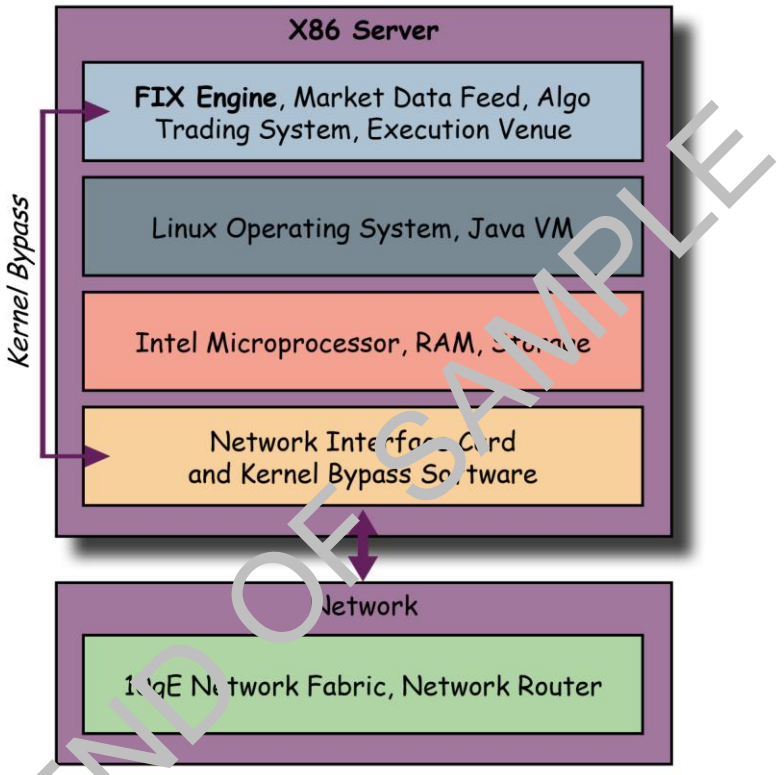

Figure 2: Reference Architecture Components

3.2.1 CPU and Servers

The test harness server was a Dell PowerEdge R710 server. This occupies 2U of rack space and incorporates energy efficient technologies to reduce power consumption and cooling. These are typically deployed in co-location environments, where space and power can be limited.

The market data simulator and execution venue server included 2 x Intel® Xeon® processor X5677 , each with 4 cores, at 3.47GHz and 16GB of RAM; running Microsoft Windows Server 2008. This configuration was sufficient for the test harness task of generating a suitable trading workload.

Dell also provided the monitoring server that hosted the network monitoring service. This comprised an Endace network monitor, timings were uploaded to the operating system only for post test processing.