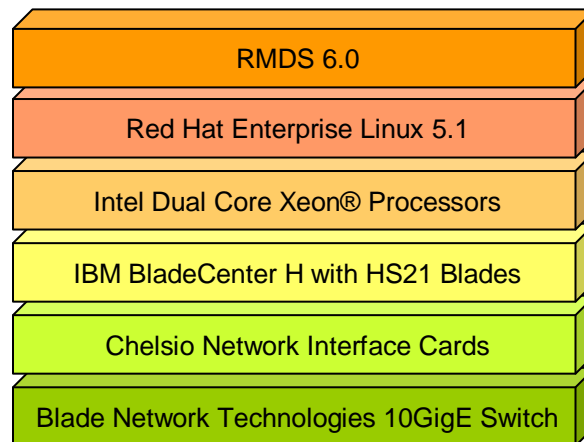


BNT 10GigE Switch and Chelsio NIC with RMDS 6 on Intel[®]-Based IBM BladeCenter[®] Servers

Issue 1.0, 18 March 2008

Technology Stack Under Test



Key Results

- Lowest mean latency ever reported with RMDS
 - Less than 0.9 milliseconds of end-to-end infrastructure latency at up to 600,000 updates per second in the low-latency configuration of RMDS
- Lowest standard deviation of latency ever reported with RMDS
 - Less than 0.5 milliseconds at rates up to 600,000 updates per second.
- Very high output rate in the “Producer 50/50” fanout test of a stacked P2PS
 - 5.8 million updates per second
 - 30% of this due to the TCP/IP Offload Engine (TOE) in the Chelsio NIC

NOTE: Consistent with many previous STAC Reports, this report is based on a vendor’s (Reuters) test methodology. The vendor-neutral STAC Benchmark specifications are currently under development by the STAC Benchmark Council. For more information, see www.STACresearch.com/council.

Disclaimer

The Securities Technology Analysis Center, LLC (STAC[®]) prepared this report at the request of Blade Network Technologies and Chelsio Communications. It is provided for your internal use only and may not be redistributed, retransmitted, or published in any form without the prior written consent of STAC. All trademarks in this document belong to their respective owners.

The test results contained in this report are made available for informational purposes only. STAC does not guarantee similar performance results. All information contained herein is provided on an "AS-IS" BASIS WITHOUT WARRANTY OF ANY KIND. STAC has made commercially reasonable efforts to adhere to Reuters' published test procedures and otherwise ensure the accuracy of the contents of this document, but the document may contain errors. STAC explicitly disclaims any liability whatsoever for any errors or otherwise.

The evaluations described in this document were conducted under controlled laboratory conditions. Obtaining repeatable, measurable performance results requires a controlled environment with specific hardware, software, network, and configuration in an isolated system. Adjusting any single element may yield different results. Additionally, test results at the component level may not be indicative of system level performance, or vice versa. Each organization has unique requirements and therefore may find this information insufficient for its needs.

Customers interested in a custom analysis for their environment are encouraged to contact STAC.

Contents

- 1. Background..... 5
- 2. Description of Tests..... 6
 - 2.1 Methodology..... 6
 - 2.1.1 Throughput testing..... 6
 - 2.1.2 Latency testing 7
 - 2.1.3 Limitations 8
 - 2.2 System Specifications 8
 - 2.2.1 Servers..... 8
 - 2.2.2 Networking 9
 - 2.2.3 Network Interface Configurations 9
 - 2.2.4 Operating System 9
 - 2.2.5 TCP and UDP Buffers – key parameters 9
 - 2.2.6 RMDS Software..... 10
 - 2.2.7 RMDS Configuration..... 10
 - 2.2.8 RMDS affinities and priority settings..... 11
- 3. Results..... 12
 - 3.1 Throughput..... 12
 - 3.2 Latency 13
- About STAC 15

Summary

The rapid growth of data traffic in the capital markets industry continues to be a major concern for technologists, as they attempt to deal with the dual requirements of higher throughput and lower latency.

Market data technologists are now looking at high-bandwidth technologies such as 10 Gigabit Ethernet (10GigE). But such advances in bandwidth have outpaced the gains in CPU processing speed, leading to I/O bottlenecks when the CPU is unable to process the packets that the network can consume or deliver. This is a major concern to those considering the deployment of 10GigE technology.

Client-side acceleration technologies reduce the load on the CPU. An example is TCP/IP Offload Engines (TOE), in which the network controller takes over processing of the TCP/IP stack.

Blade Network Technologies (BNT), which provides 10GigE switches, and Chelsio Communications, which provides interface cards that implement TOE, asked STAC to measure the performance of their products in a market data environment using RMDS. The three goals were:

- Find the maximum P2PS Producer 50/50 throughput with TOE enabled
- Determine the increase in throughput provided by TOE
- Measure RMDS latency with TOE enabled

To summarize, we found:

- Lowest mean latency ever reported with RMDS
 - Less than 0.9 milliseconds of end-to-end infrastructure latency at up to 600,000 updates per second in the low-latency configuration of RMDS
- Lowest standard deviation of latency ever reported with RMDS
 - Less than 0.5 milliseconds at rates up to 600,000 updates per second.
- Very high output rate in the “Producer 50/50” fanout test of a stacked P2PS
 - 5.8 million updates per second
 - 30% of this due to the TCP/IP Offload Engine (TOE) in the Chelsio NIC

1. Background

Market data latency has a huge impact on the overall speed with which a trading firm can execute a transaction in response to new information. In some markets, firms can profit from as little as one millisecond of advantage over competitors, which drives them to find sub-millisecond optimizations of the systems fueling their trades. The latency obsession has resulted from the spread of automated trading to nearly every geography and asset class, and the resulting imperative to exploit—or defend against—new latency arbitrage opportunities.

Another consequence of automated trading is a ballooning of market data traffic volumes, which complicates the latency race, thanks to a well-established tradeoff between throughput and latency. Update-rate increases of 2 to 6 times in a single year are not uncommon for today's exchanges. Automated trading drives this traffic by both increasing transaction volumes and increasing the ratio of quotes and cancellations to actual trades. While North American venues still produce the most traffic, many observers expect the Markets in Financial Instruments Directive (MiFID) to trigger a sharp increase in European traffic as the number of trade-reporting venues proliferates. On top of this, large sell-side institutions often generate enormous amounts of real-time data internally, which they pump onto their internal market data system. The traffic from internal content sometimes exceeds that of information coming in from external sources.

This combination of forces keeps market data technologists on the lookout for new technologies that can shift the performance tradeoffs in the right direction. One layer of the technology stack that receives ongoing scrutiny is the network (switches, interface cards, drivers, etc.). Market data systems are extremely network-I/O intensive. Most clients with heavy market data demands currently use 1-gigabit Ethernet (GigE). But requirements for some of the most intense automated trading systems are beginning to exceed a gigabit. Hence, one of the new networking technologies receiving a great deal of attention is 10-gigabit Ethernet (10GigE). With 10 times the bandwidth of GigE, 10GigE offers considerable headroom for growth.

Blade Network Technologies (BLADE) designed their Nortel Layer 2/3 10GigE Switch Module for IBM BladeCenter for the demands of low-latency market data feeds as well as traditional storage and I/O applications. This Blade switch has twenty 10GigE ports, and according to the company, can operate at line rate (zero loss) performance on all ports and at all packet sizes.

The efficiency with which servers can handle network traffic is also coming under increasing focus, in both automated trading and broader market data distribution. Servers now routinely sport 4, 8, or even 16 processing cores; yet firms' ability to exploit this processing potential is limited by their ability to get data into and out of the server.

Chelsio's 10Gb Ethernet TOE cards (S320EM-BCH) are aimed squarely at this problem. According to the company, the card's TCP Offload Engine (TOE) increases Ethernet bandwidth and reduces Ethernet latency and latency-jitter for high-performance/low-latency computing environments, while reducing power consumption and freeing up CPU cycles for greater use by applications. Chelsio claims that its cards are ideal in market data environments, where traffic consists of high message rates and small message sizes.

In theory, this should reduce the amount of network-related processing required of the CPUs and lead to improved server performance. We put this theory to the test by benchmarking two market data system configurations: the first emulating point-to-point distribution of data to hundreds of users, and the second following a configuration for low-latency distribution to heavy-duty applications.

In both cases, we ran the Reuters Market Data System (RMDS), using Reuters standard benchmarking procedures. We chose these procedures in order to enable easy comparison to other tests we have run with RMDS and because the emerging industry-standard STAC Benchmarks for market data middleware (STAC-M2) are still in development.

2. Description of Tests

2.1 Methodology

The tests followed the procedures set forth by Reuters for hardware vendors and used the test data supplied by Reuters.

2.1.1 Throughput testing

The P2PS “Producer 50/50” test is an extreme test of the fanout capability of a P2PS machine. It is oriented toward environments in which many users are connected to the P2PS and users have a high degree of commonality in their watchlists (meaning that for most of the updates that the P2PS receives from the backbone, it must forward each update to many users).

The `sink_driven_src` utility was used to generate update traffic from the sample files provided with the Source Distributor (`sample.xml`), and the `rmdstestclient` utility was used to consume the updates. The RMDS infrastructure was tuned for maximum throughput as per the Reuters RMDS 6.0 Performance Test Procedures, and the update rate was increased until data was lost, the system failed, or throughput lagged.

To maximize fanout performance, we chose a multiplex topology, in which two P2PS instances feed 2 client apps per P2PS. Two publishing apps and two source distributors supplied the data. This sort of “stacked” topology effectively co-locates multiple P2PS instances that would otherwise run on separate servers. The test harness is diagrammed in Figure 2-1.

As each blade had two Chelsio 10Gb/s interface ports, the TCP traffic between the publishing apps and source distributors and between the P2PS instances and client apps used one network, while the UDP multicast traffic between the source distributor box and P2PS box used a separate interface port and network.

We were interested in determining whether the Maximum Transmission Unit (MTU) size had any effect on performance. Prior to testing, we ran the configuration with the MTU set to 1500 bytes, and then again at 8000 bytes. We observed that there was no significant difference between these settings, so we used the default value of 1500 bytes.

We ran this test with and without TOE enabled.

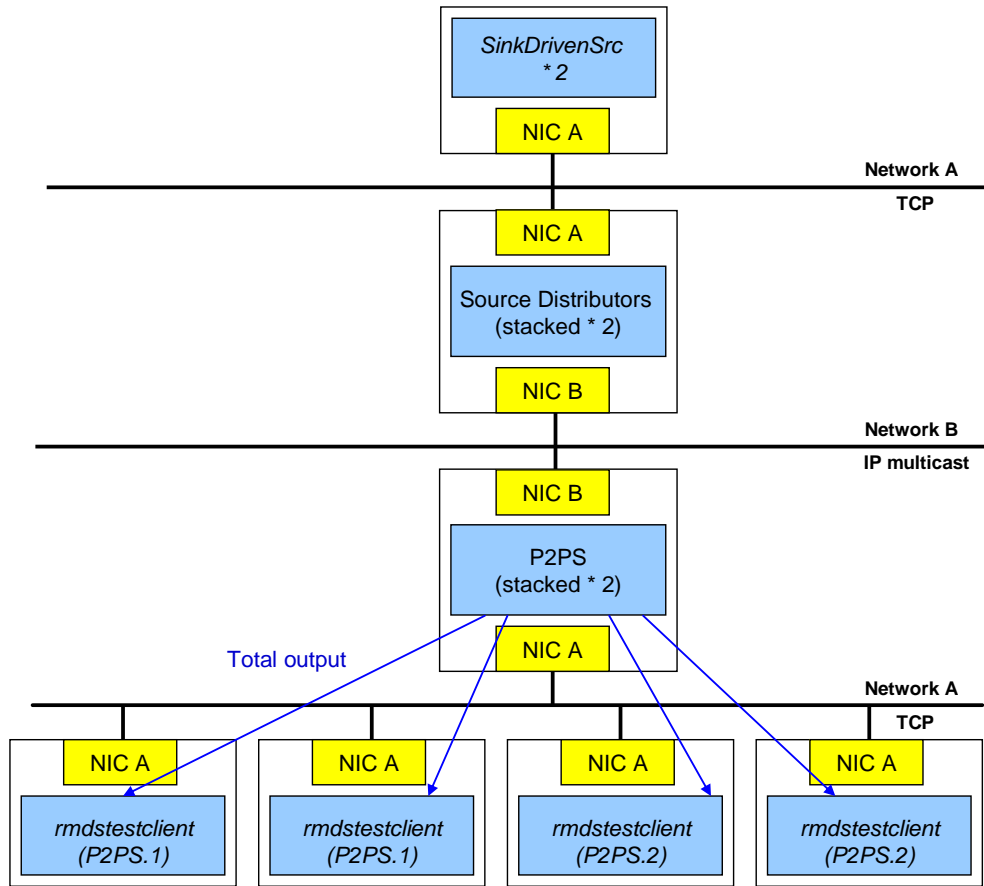


Figure 2-1

2.1.2 Latency testing

As specified by the Reuters test procedures, the embedded timestamp approach was used to calculate end-to-end latency for level 1 data, as shown in Figure 2-2. Latency was measured using **sink_driven_src** as the publisher and **rmdstestclient** as the subscriber. In the embedded timestamp approach, the publisher embeds timestamps into selected updates which the subscriber uses for latency calculations. In this scenario, the publisher and subscriber must be running on the same node for accurate timestamps. Latency tests were run on infrastructure tuned for low latency as per the Reuters RMDS 6.0 Performance Test Procedures.

Again, as each blade had two Chelsio 10Gb/s interface ports, the TCP traffic between the publishing apps and source distributors and between the P2PS instances and client apps used one network (Network A) and the UDP multicast traffic between the source distributor box and P2PS box used a separate interface port and network (Network B).

End-to-end Infrastructure Latency Test

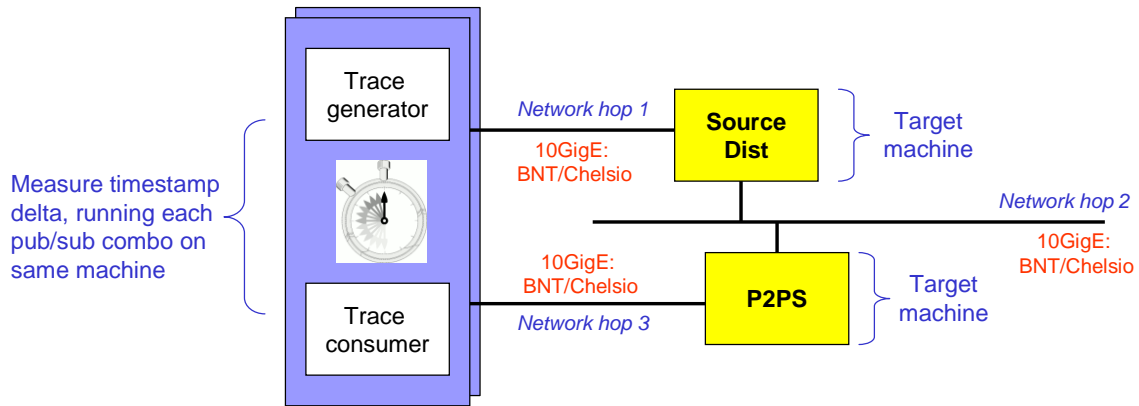


Figure 2-2

2.1.3 Limitations

The Reuters test methodology uses sampling to determine latency statistics for an interval, which reduces the accuracy of distribution-related statistics such as max and standard deviation.

2.2 System Specifications

2.2.1 Servers

The IBM BladeCenter used in these tests was:

Vendor Model	IBM BladeCenter H Chassis
Blade Bays	7 for throughput and 3 for latency
Rack Units	9U
Power Supplies	2

Each of the servers in the test harness had the following specifications:

Vendor Model	IBM eServer BladeCenter HS21
Processors	2
Processor type	Dual-Core Intel Xeon 5160 @ 3.00 GHz
Cache	4MB Integrated L2 Cache split between 2 cores
Bus speed	1.333 MHz
Memory	4 GB (2x2048 MB) DDR DIMMS
Disk	73 GB SAS

2.2.2 Networking

Switch	Blade Network Technologies' Nortel 10Gb Ethernet Switch Module for IBM BladeCenter H, Software version 1.0.3
NIC	Chelsio NIC S320EM-BCH
NIC driver	cxgb3, version 1.0.129a
NIC firmware	T 5.0.0 TP 1.1.0
NIC BIOS	BCE1.08

2.2.3 Network Interface Configurations

Any settings changed from the defaults are noted below

The following values were set on each Ethernet interface (eth0, eth1, ... <ethX>) used for RMDS traffic:	Command
Txqueuelen	ifconfig <ethX> txqueuelen 10000
TOE on/off	sysctl -w toe.toe0_tom.activated=0 (off) sysctl -w toe.toe0_tom.activated=1 (on)

2.2.4 Operating System

Version	Red Hat Enterprise Linux 5.1 beta, 32-bit Kernel 2.6.18-36.el5
OS services	All OS daemons were stopped with the exception of : init, udevd, auditd, audispd, syslogd, klogd, sshd, smartd, mingetty

2.2.5 TCP and UDP Buffers – key parameters

	Values were those specified by the Reuters guidelines. The following lines were entered into the System File (/etc/sysctl.conf):	System File
Setup-specific changes noted	Net.core.wmem_max = 16777216	/etc/sysctl.conf
	Net.core.wmem_default = 8388608	
	Net.core.rmem_max = 16777216	
	Net.core.rmem_default = 8388608	
	Net.ipv4.tcp_rmem = 4096 262144 16777216	
	Latency tests: latency: net.ipv4.tcp_wmem = 4096 262144 16777216	
	Throughput: net.ipv4.tcp_wmem = 4096 1048576 8388608	
	Net.ipv4.tcp_mem = 4096 8388608 16777216	
Net.ipv4.ip_local_port_range = 34800 65535		

2.2.6 RMDS Software

RMDS Binaries	src_dist ver. mdh6.0.2.L2 p2ps ver. p2ps6.0.2.L2 rrcp as included in p2ps6.0.2.L2
RMDS Test Tools	sink_driven_src (from mdh6.0.0.L4) rmdstestclient (from p2ps6.0.0.L4)

2.2.7 RMDS Configuration

	Ensure the following settings in <i>rmds.cnf</i> :
Common to all tests	*p2ps*rsslMsgPacking : True
	*p2ps*hashTableSize = 200000
	*usePointToPointData = False
	*RRCP*maxPktPoolSize : 80000
	*RRCP*pktPoolLimitHigh : 70000
	*RRCP*pktPoolLimitLow : 60000
	*RRCP*userQLimit : 32768
	*RRCP*udpRecvBufSize : 4096
	*RRCP*udpSendBufSize : 4096
	*<serviceName>*cacheLocation : srcApp
*p2ps*enableCache : False	
Throughput test	*p2ps*timedWrites : True
	*p2ps*flushInterval : 20
	*p2ps*tcpNoDelay : False
	*<serviceName>*rrmpFlushInterval : 20
	*p2ps*tcpSendBufSize : [commented out]
	*p2ps*guaranteedOutputBuffers : 800
	*p2ps*maxOutputBuffers : 5000
*p2ps*poolSize : 32000	
Latency test	*p2ps*timedWrites : False
	*p2ps*flushInterval : 0
	*p2ps*tcpNoDelay : True
	*<serviceName>*rrmpFlushInterval : 0
	*p2ps*tcpSendBufSize : 64240
	*p2ps*guaranteedOutputBuffers : 200
	*p2ps*maxOutputBuffers : 400
	*p2ps*poolSize : 16000
	*src_dist*route*numIpcInputBuffers : 10
*src_dist*route*numIpcOutputBuffers : 100	

	*src_dist*server*ipc*transmissionBus*guaranteedOutputBuffers : 200
	*src_dist*server*ipc*transmissionBus*numInputBuffers : 3
	*src_dist*server*ipc*transmissionBus*poolSize : 1600

2.2.8 RMDS affinities and priority settings

p2ps	<ul style="list-style-type: none">• each instance bound to its own set of cores• set to a nice value of -18
rrcpd	<ul style="list-style-type: none">• each instance bound to its own set of cores• On the p2ps machine, one rrcpd bound to the core processing the interrupts from the Chelsio NIC
rmdstestclient	<ul style="list-style-type: none">• bound to a single core using taskset, same core that was processing the interrupts from the Chelsio NIC• set to SCHED_FIFO class with priority 50
SinkDrivenSrc	<ul style="list-style-type: none">• set to SCHED_FIFO class with priority 50• bound to a single core using taskset

3. Results

3.1 Throughput

Table 1 shows the aggregate updates/second of P2PS output in the Producer 50/50 test, with and without TOE enabled. The data show an improvement of 42% when TOE was enabled (put otherwise, 30% of the max performance was attributable to TOE).

Mode	Throughput rate (updates/sec)
Without TOE	4.04 million
With TOE	5.76 million

Table 1

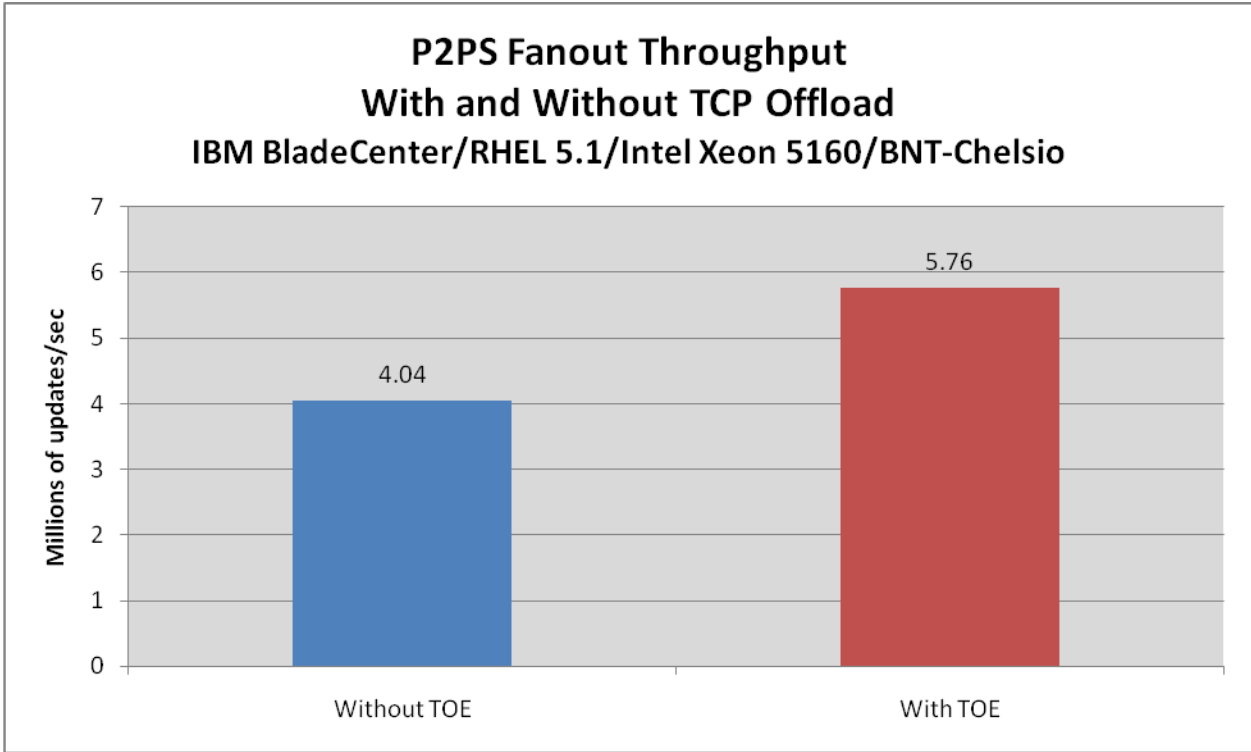


Figure 3-1

3.2 Latency

“End-to-end” RMDS latency is defined as the delta between the time an update is posted by the publisher application to its API and the time the same update is received by the consuming application from its API, i.e. it includes the latency contribution from both the API and the core infrastructure components. Table 2 records the latency statistics for the RMDS configuration. The mean latencies are presented in Figure 3-2 below, while the standard deviations are presented in Figure 3-3.

The system demonstrated the lowest mean and standard deviation of latency STAC has ever reported with RMDS. Mean latency was less than 0.9 ms all the way to 600kups. At higher rates, the system failed.

Update Rate [74-byte RWF messages/sec]	Mean Latency (milliseconds)	Std Deviation (milliseconds)	Maximum Latency (milliseconds)	Minimum Latency (milliseconds)	Number of Latency Points
1,000	0.133	0.005	0.160	0.126	3000
5,000	0.173	0.006	0.201	0.159	3000
10,000	0.221	0.010	0.300	0.198	3000
20,000	0.264	0.027	0.342	0.207	3000
30,000	0.285	0.037	0.380	0.212	3000
40,000	0.311	0.050	0.783	0.219	3000
50,000	0.314	0.041	0.874	0.198	3000
60,000	0.326	0.053	0.935	0.202	3000
70,000	0.337	0.065	1.025	0.198	3000
80,000	0.351	0.084	1.351	0.200	3000
90,000	0.361	0.090	1.348	0.204	3000
100,000	0.372	0.107	1.592	0.203	3000
150,000	0.419	0.092	1.019	0.207	3000
200,000	0.473	0.118	1.142	0.206	3000
250,000	0.509	0.139	1.537	0.206	3000
300,000	0.562	0.171	1.737	0.203	3000
350,000	0.605	0.210	2.139	0.204	3000
400,000	0.673	0.248	2.441	0.199	3000
450,000	0.707	0.287	2.764	0.214	3000
500,000	0.772	0.337	3.008	0.215	3000
550,000	0.800	0.368	3.412	0.215	3000
600,000	0.863	0.443	3.639	0.211	3000

Table 2

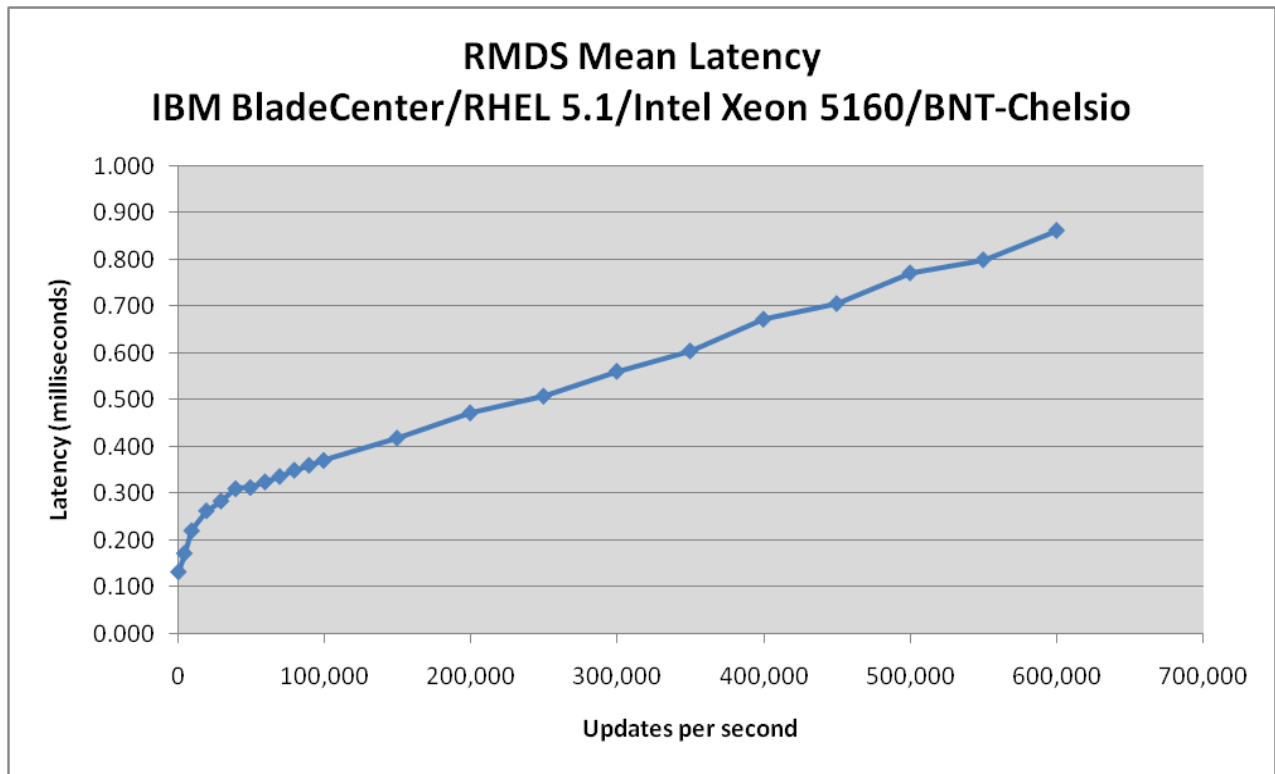


Figure 3-2: Mean Latency

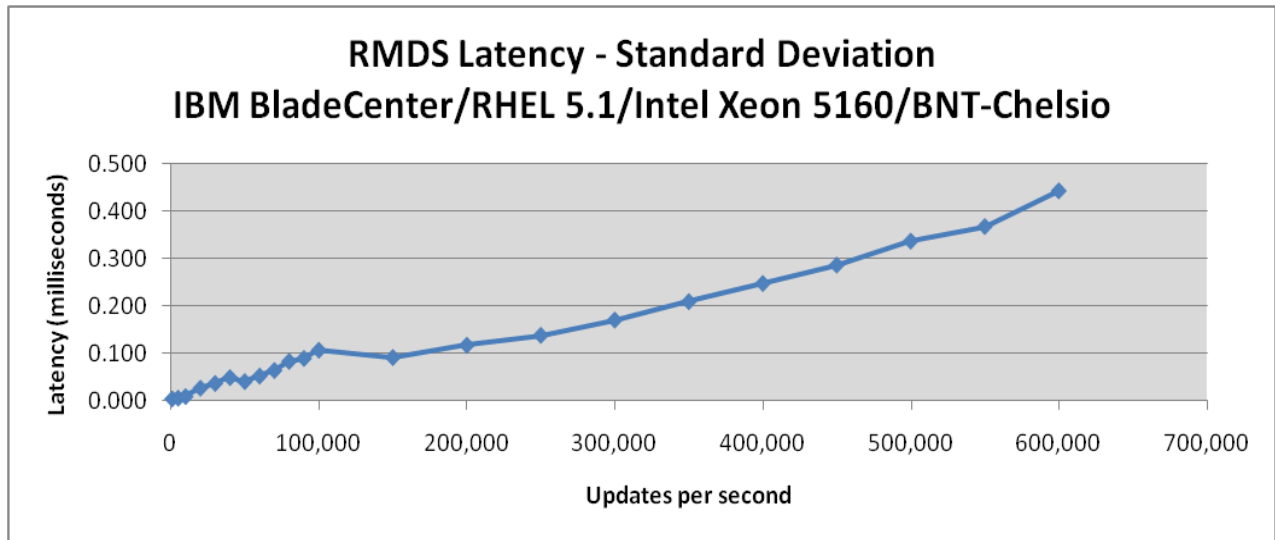


Figure 3-3: Latency Standard Deviation

About STAC



The Securities Technology Analysis Center, or STAC, conducts private and public hands-on research into the latest technology stacks for capital markets firms and their vendors. STAC provides optimization expertise, advanced tools, and simulated trading environments in STAC Labs. Public STAC Reports, available for free at www.STACresearch.com, document the capability of specific software and hardware to handle key trading workloads such as real-time market data, analytics, and order execution.

STAC also facilitates the STAC Benchmark Council, an organization of leading trading firms and vendors that specify standard ways to measure the performance of trading solutions (see www.STACresearch.com/council).

To be notified when new STAC Reports like this one are issued, or to learn more about STAC, see our web site at www.STACresearch.com.