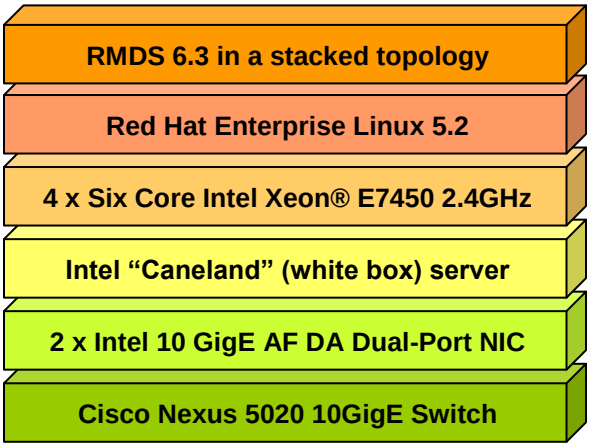
RMDS 6.3 on Intel Dunnington-Based 24-Core Server With Intel® 10 GigE NICs and Cisco Nexus 5020 switch

Issue 1.01, 21 Sep 2008

Technology Stack Under Test



Key Results

- ➔ Highest stacked Point-to-Point Server throughput to date on a single server
- ➔ 15,150,000 updates per second through a single Point-to-Point Server machine with jumbo frames (MTU = 9000 bytes).
- ➔ 11,362,500 updates per second through a single Point-to-Point Server machine with standard frames (MTU = 1500 bytes)

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.

STAC SECURITIES TECHNOLOGY ANALYSIS CENTER

Disclaimer

The Securities Technology Analysis Center, LLC (STAC®) prepared this report at the request of Intel. 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

Summary

The rapid growth of data traffic and volatility in the capital markets industry continues to be a major concern for technologists. They seek to ensure that their systems are capable of handling peak rates and unexpected market events. At the same time, managers are being pressed to reduce power consumption, data center space, and operating costs. Market data technologists are now looking for ways to exploit multi-core servers to satisfy these needs. However, network I/O is often a bottleneck on the performance of these servers. For this reason, trading firms are very interested in high-bandwidth technologies such as 10 Gigabit Ethernet (10GigE).

Intel, which provides multi-core CPUs and 10GigE interface cards, and Cisco, which provides 10 GigE switches, asked STAC to measure the performance of their products in a market data environment using RMDS. The goal of the project was to find the maximum P2PS Producer 50/50 throughput on a single multi-core box.

To summarize, we found:

- Highest Point-to-Point Server throughput to date on a single server:
- 15,150,000 updates per second through a single Point-to-Point Server machine with jumbo frames (MTU = 9000 bytes).
- 11,362,500 updates per second through a single Point-to-Point Server machine with standard frames (MTU = 1500 bytes).

1. Background

The Intel® Xeon® 7400 server platform supports four 6-core Intel Xeon processors (in Intel code names, this is the "Caneland" platform, with "Dunnington" processors) based on 45nm Hi-k process technology. Servers based on this platform have 50% more cores and 2x the cache-memory capacity of the previous generation: 16 MB of shared L3 cache. According to Intel, by increasing the efficiency of cache-to-core data transfers, and maximizing main memory to processor bandwidth, this additional cache reduces latency by storing larger data sets closer to the processor. Dunnington processors are socket compatible with the previous generation of Tigerton processors. Existing Caneland platforms can be upgraded to Dunnington by upgrading the BIOS and CPUs. Intel believes that this platform will be interesting to market data system managers, who are being pressed to reduce power consumption, data center space, and operating costs.

The purpose of this project was to maximize the throughput of RMDS P2PS on a single Intel Caneland server. Previous STAC research has found that RMDS performance on multi-core processors can be significantly increased by changing the topology of RMDS processes on the machine. In these tests, we utilized stacked topologies to exploit as much of the server's 24 cores as possible. RMDS customers, facing rapidly escalating update rates, are increasingly interested in stacked configurations, particularly to enable higher per-machine P2PS throughput.

Because of our prior RMDS testing on Xeon 5160 (Woodcrest)-based servers,[1] we anticipated that at the extremely high update rates used in these tests, networking would become a bottleneck. In the first such STAC Report, we used multiple Gigabit interfaces and surmised in that report that higher-throughput interconnects would be beneficial. We decided to put that assumption to the test using the latest Intel 10 Gigabit NICs and latest Cisco Nexus 5020 10 Gigabit switch.

In the interest of time, we ran a subset of the standard Reuters benchmarks that are focused on throughput. We also limited the tests by running only RRCP as the backbone transport and only OMM/RWF as the message format.

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

[1] See the following reports at www.STACresearch.com:
"BNT 10GigE Switch and Chelsio NIC with RMDS 6," 18 March 2008
"RMDS on Voltaire Infiniband - HP DL380, Dual Core Xeon 3.0 GHz, RHEL 4.4," 11 February 2007
"RMDS on HP DL380 with Dual Core Xeon 3.0 GHz and Stacked RMDS Topology", 3 November 2006.

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 a single RRCP feeds multiple P2PS instances and each of those P2PS feeds multiple client apps. A single publishing app and a single source distributor 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.

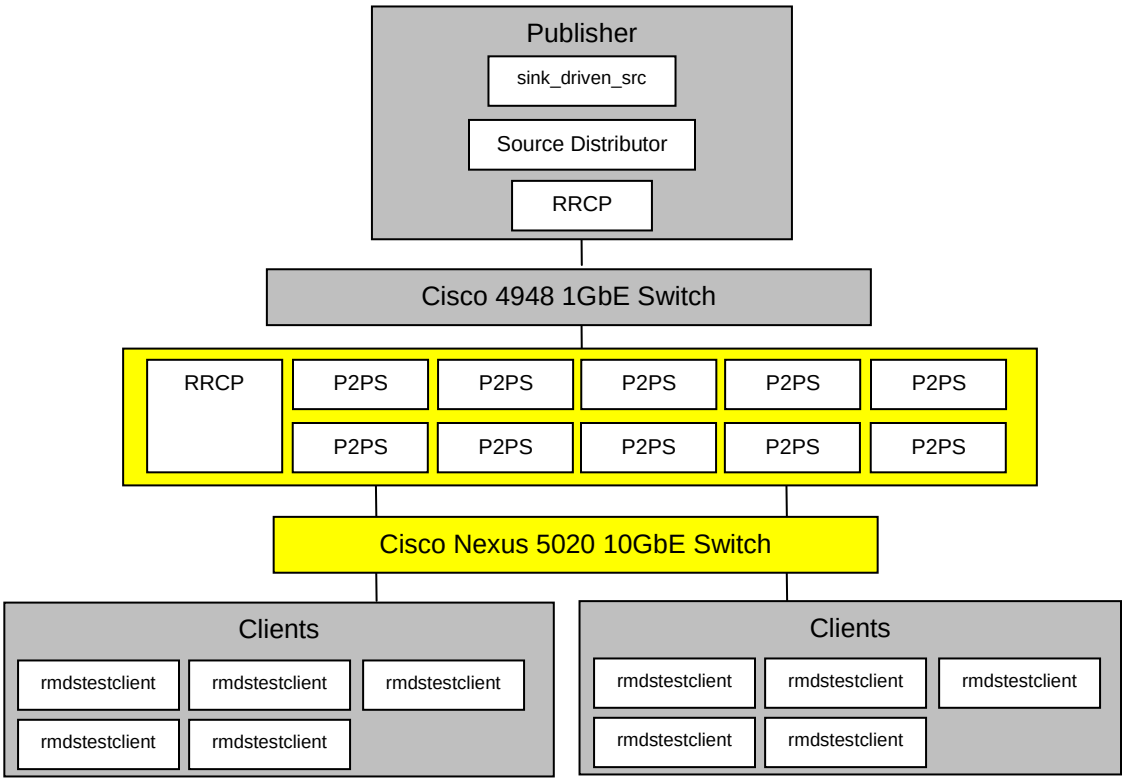


Figure 2-1: Stack Under Test

Each server running consumers on the client LAN had two physical Intel 10Gb/s interface ports however only one of them was used for these tests. The server running the P2PS instances had two Intel 10Gb/s interface cards, each with two ports; however we only used one port per card. The TCP traffic between the P2PS instances and the client applications was split between two switch VLANs (1 per server running consumers). 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. We ran the configuration with the MTU set to its default value of 1500 bytes, and then again

set to 9000 bytes. We observed that there was a significant difference between these settings, so we tested with both the default and larger MTU size. We ran this test without Intel I/O Acceleration Technology (IOAT) enabled.

2.2 System Specifications

2.2.1 Servers

The server used for the P2PS servers had the following specification:

Vendor Model	Intel® chassis code named Deer Harbor with Fox Cove motherboard
Processors	4
Processor Type	Six-Core Intel® Xeon X7450 Processor, 2.40 GHz
Cache	32KB Integrated L1 Cache per core 3 MB Integrated L2 Cache split between 2 cores 12 MB Integrated L3 Cache split between 6 cores
Bus Speed	1.066 GHz
Memory	32 GB (16x2048 MB) FBD 5-5-5 667 MHz
Eth0 & Eth1	2 x 1GbE Intel® NIC codenamed Zoar with I/OAT2 (on I/O riser card)
Eth2 & Eth 3	2 x 1GbE Intel® NIC codenamed Gilgal (on baseboard)
Eth4 – Eth 7	2 x Intel Corporation 82598EB 10 Gigabit AT Dual Port Network Connection (rev 01)
NIC Note	Management traffic was directed at eth2 All RMDS backbone traffic was directed over the interface Eth3 All Client traffic was directed over the interfaces (Eth4 and Eth6)
BIOS	BIOS Information Vendor: Intel Corporation Version: SFC4UR.86B.01.00.S005.060520082030 Release Date: 06/05/2008 BIOS R05 BMC Version 17 HSC 2.09 SDR 14
Disk Controller	Logic MegaRAID SAS 8884E RAID
Disks	2 x 2.5” 73 GB SAS HD
Rack Units	4

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

Vendor Model	Intel® chassis code named Deer Harbor with Fox Cove motherboard
Processors	4
Processor Type	Quad-Core Intel® Xeon X7350 Processor, 2.93 GHz
Cache	32KB Integrated L1 Cache per core 4 MB Integrated L2 Cache split between 2 cores
Bus Speed	1.066 GHz

Memory	32 GB (16x2048 MB) FBD 5-5-5 667 MHz
Eth0 & Eth1	2 x 1GbE Intel® NIC codenamed Gilgal (on baseboard)
Eth2 & Eth3	2 x 1GbE Intel® NIC codenamed Zoar with I/OAT2 (on I/O riser card)
Eth4 – Eth5	Intel Corporation 82598EB 10 Gigabit AT Dual Port Network Connection (rev 01)
NIC Note	Management traffic was directed at eth2 All Client traffic was directed at eth4
BIOS	BIOS Information Vendor: Intel Corporation Version: SFC4UR.86B.01.00.0016.072720071338 Release Date: 07/27/2007 BIOS R16 BMC Version 14 HSC 2.06 SDR 10
Disk Controller	Logic MegaRAID SAS 8884E RAID
Disks	2 x 2.5” 73 GB SAS HD
Rack Units	4

The Publisher server had the following specifications:

Vendor Model	Intel® chassis code named Deer Harbor with Fox Cove motherboard
Processors	4
Processor Type	Quad-Core Intel® Xeon E7340 Processor, 2.4 GHz
Cache	32KB Integrated L1 Cache per core 4 MB Integrated L2 Cache split between 2 cores
Bus Speed	1.066 GHz
Memory	32 GB (16x2048 MB) FBD 5-5-5 667 MHz
Eth0 & Eth1	2 x 1GbE Intel® NIC codenamed Gilgal (on baseboard)
Eth2 & Eth3	2 x 1GbE Intel® NIC codenamed Zoar with I/OAT2 (on I/O riser card)
NIC Note	Management traffic was directed at eth1 All Data traffic was directed at eth2
BIOS	BIOS Information Vendor: Intel Corporation Version: SFC4UR.86B.01.00.0016.072720071338 Release Date: 07/27/2007 BIOS R16 BMC Version 14 HSC 2.06 SDR 10
Disk Controller	Logic MegaRAID SAS 8884E RAID
Disks	2 x 2.5” 73 GB SAS HD
Rack Units	4

2.1.1 Networking

Switch	Cisco Nexus 5020 - 10 Gigabit Ethernet software version 4.0(0)N1(2)
NIC	Intel Corporation 82598EB 10 Gigabit AT Dual Port Network Connection (rev 01)
NIC driver	Ixgbe version 1.3.20.11-lro
NIC firmware	N/A
NIC BIOS	N/A

2.1.2 Network Interface Configurations

Any settings changed from the defaults are noted below

The following values were set on the 10Gb/s interfaces used for RMDS TCP traffic	Command
Txqueuelen	Iconfig <ethX> txqueuelen 10000
Number of Rx queues	“options ixgbe RSS=4,4,4,4” in /etc/modprobe.conf

2.1.3 Operating System

Version	Red Hat Enterprise Linux Server release 5.2 (Tikanga) 2.6.18-92.1.10.el5 x64_64
OS services	All system services were stopped with the exception of : anacron, atd, auditd, crond, haldaemon, microcode_ctl, network, smartd, sshd, syslog

2.1.4 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 8388608 16777216	
	net.ipv4.tcp_wmem = 4096 8388608 16777216	
	Net.ipv4.tcp_mem = 4096 8388608 16777216	
	Net.ipv4.ip_local_port_range = 34800 65535	

2.1.5 RMDS Software

RMDS Binaries	P2PS6.3.1.L2.linux.tis.rrg mdh6.3.0.L4.linux.tis.rrg rrcp as included in mdh6.3.0.L4.linux.tis.rrg
RMDS Test Tools	sink_driven_src (from mdh6.3.0.L4) rmdstestclient (from P2PS6.3.1.L2)

2.1.6 RMDS Configuration

	Ensure the following settings in *rmds.cnf*:
Common to all tests	*P2PS*rsslMsgPacking : True
	*P2PS*hashTableSize = 300000
	*usePointToPointData = True
	*RRCP*maxPktPoolSize : 200000
	*RRCP*pktPoolLimitHigh : 190000
	*RRCP*pktPoolLimitLow : 180000
	*RRCP*userQLimit : 32768
	*RRCP*udpRecvBufSize : 256
	*RRCP*udpSendBufSize : 256
	*<*serviceName*>*cacheLocation : srcApp
	*P2PS*enableCache : False
Throughput test	*P2PS*timedWrites : True
	*P2PS*flushInterval : 20
	*P2PS*tcpNoDelay : False
	*<*serviceName*>*rrmpFlushInterval : 20
	*P2PS*tcpSendBufSize : 64240
	*P2PS*guaranteedOutputBuffers : 1000
	*P2PS*maxOutputBuffers : 16000
	*P2PS*poolSize : 32000

2.1.7 RMDS affinities and priority settings

No process priorities were used for this test given the number of cores available on the machines.

2.1.8 Interrupt Bindings

All numeric interrupts from /proc/interrupts were bound as follows:

P2PS Server (Dunnington)	eth3	Core 1
	eth4 Tx	Core 6
	eth4 Rx	Cores 2 - 5
	eth6 Tx	Core 9
	eth6 Rx	Cores 7,8,10,11
	All other interrupts	Core 0
Client servers (Canelands running rmdstestclients)	eth4 Tx	Core 11
	eth4 Rx	Cores 3,10
	All other interrupts	Core 0

Resulting core bindings

	P2PS Server (Dunnington)	Client servers (Canelands running rmdstestclients)
Core 0	irqs/rrcpd	irqs
Core 1	rrcpd	
Core 2	eth4 Rx	
Core 3	eth4 Rx	eth4 Rx
Core 4	eth4 Rx	
Core 5	eth4 Rx	
Core 6	eth4 Tx	
Core 7	eth6 Rx	
Core 8	eth6 Rx	rmdstestclient1
Core 9	eth6 Tx	rmdstestclient2
Core 10	eth6 Rx	eth4 Rx
Core 11	eth6 Rx	eth4 Tx
Core 12	rrcpd	rmdstestclient3
Core 13	rrcpd	rmdstestclient4
Core 14	p2ps1	rmdstestclient5
Core 15	p2ps2	
Core 16	p2ps3	
Core 17	p2ps4	
Core 18	p2ps6	
Core 19	p2ps8	
Core 20	p2ps10	
Core 21	p2ps5	
Core 22	p2ps7	
Core 23	p2ps9	

These bindings came as a result of applying the following guidelines based upon the Intel architecture:

- Maximize single application use of L2 cache by spreading processes across cores so that if possible one process has more or less exclusive access to the cache i.e. other process instance not bound to the other core sharing the cache if possible
- On the Dunnington architecture, where possible bind NIC interrupts to cores on the same processor as the originating process to take advantage of the shared L3 cache

3. Results

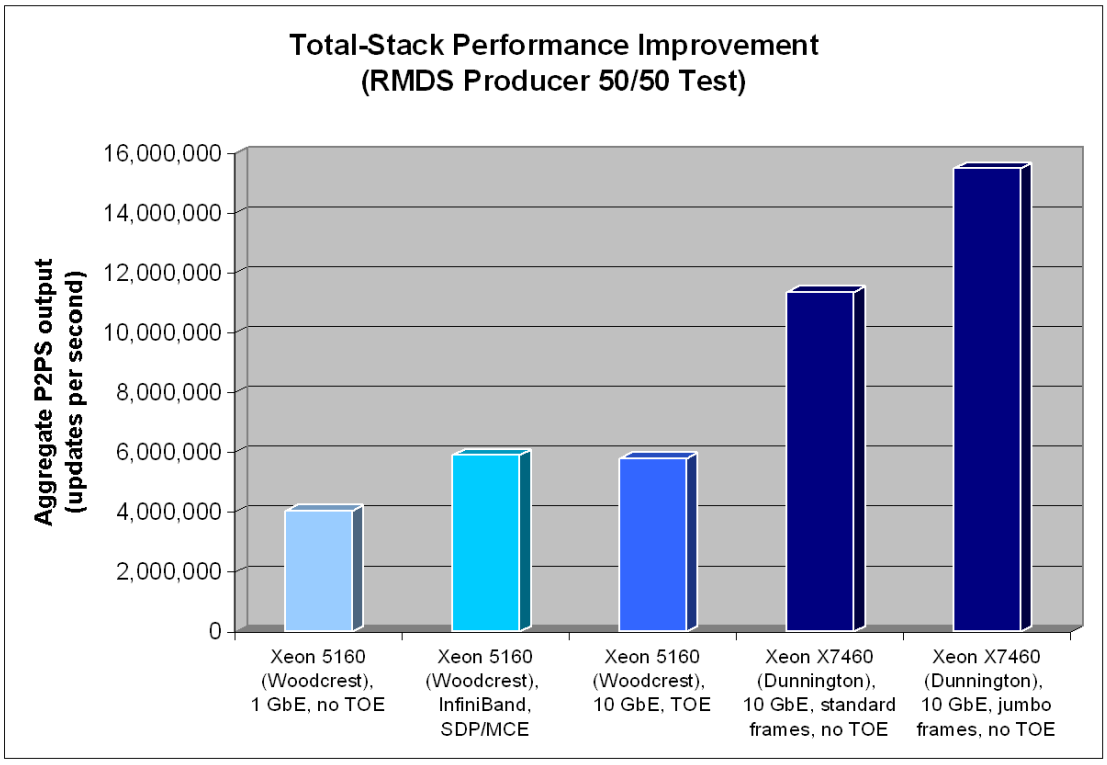
The table below documents the throughput achieved in the configuration explained above.

RMDS Component	Configuration Options	Max Throughput
P2PS	Cache disabled, Producer 50/50, stacked topology, standard frames	225,000 input 11,362,500 output
P2PS	Cache disabled, Producer 50/50, stacked topology, jumbo frames	300,000 input 15,150,000 output

Table 1: P2PS Producer 50/50 Throughput test results

The multi-core, 10 GigE solution used in this benchmark facilitated the highest P2PS server throughput observed on a single server to date. We believe that further optimizations were possible with this system, but time limited our ability to pursue them.

It is interesting to put these results in context. Figure 3-1 plots the results of the P2PS Producer 50/50 test from several STAC Reports using Intel processors and various networking technologies. Many things were different from stack to stack (not all of which are noted in the chart), so it is not possible to draw conclusions about individual components from this comparison. Rather, the point is to show how the achievable performance using state-of-the-art components has increased in the course of less than two years. It has nearly quadrupled.



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 performance measurement services, 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**.