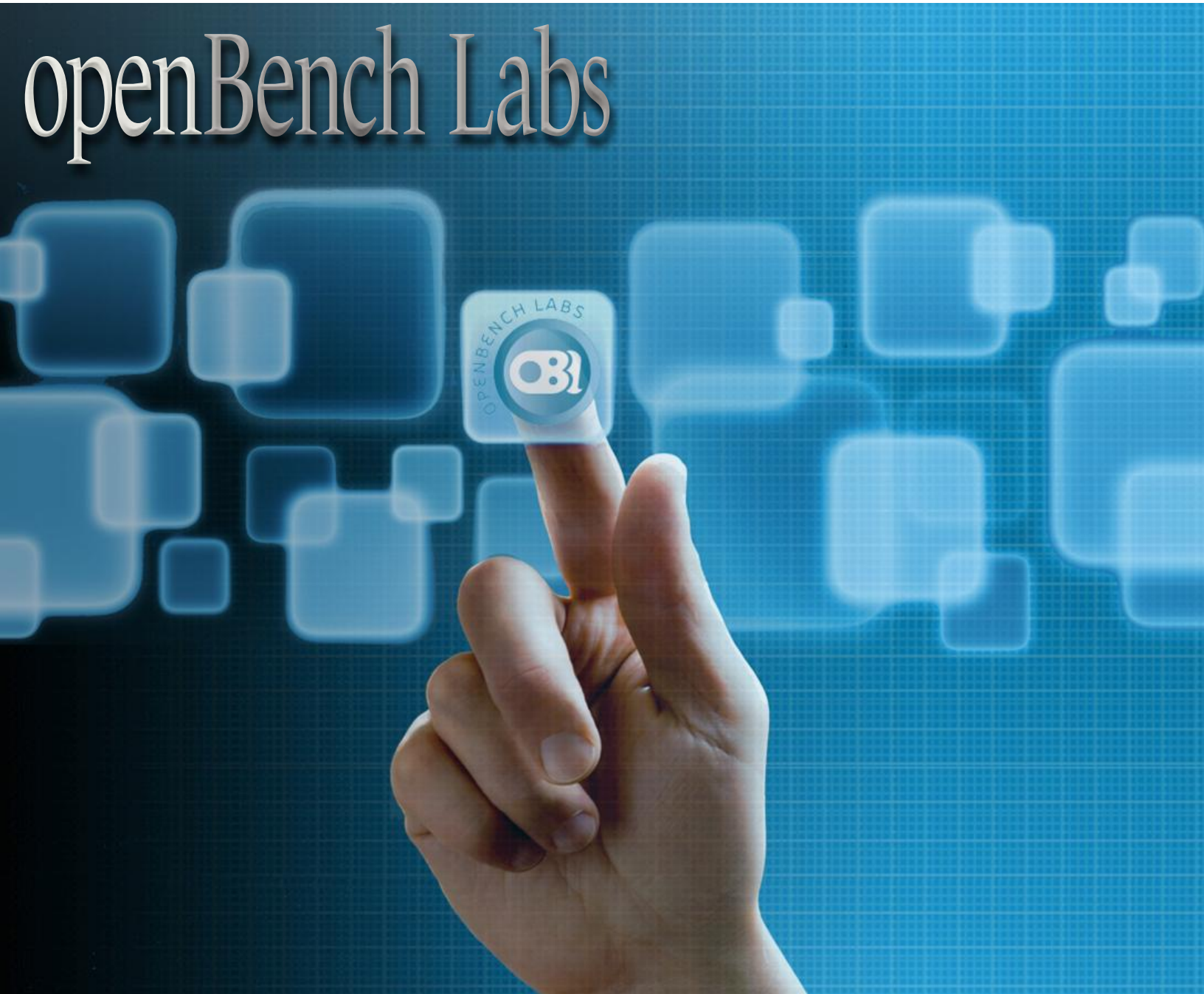
May 20, 2013

Condusiv's V-locity Server Boosts Performance of SQL Server 2012 by 55%

Optimizing I/O for Increased Throughput and Reduced Latency on Physical Servers



Executive Overview

"V-locity Server increased SQL Server 2012 transaction processing rate by an average of 55% and improved response time by 33%."

Why Read This Document?

For this briefing, openBench Labs tested the ability of V-locity® Server to optimize I/O in a SQL Server environment. Using SQL Server 2012, we tested a mix of a high volume of light-weight SQL select transaction processing (TP) queries, combined with heavy-weight background update queries.

For the SQL Server benchmark testing, openBench simulated 1 to 32 daemon processes (1 daemon generates the equivalent of 70 normally-queued user processes) issuing queries non-stop. When a real application user interacts with SQL Server, there is lag between queries issued. In our test scenario, however, the daemon process issued queries without lag—that is, no think-time, type-time, or pause-time between query activity.

In a series of tests, openBench Labs measured the ability of V-locity Server's IntelliMemory™ to offload I/O on read operations through dynamic caching, in order to boost throughput and reduce latency. In addition we examined the ability of IntelliWrite® technology to prevent unnecessary split I/Os, using its intelligence to extend current database files and create new log files as single contiguous collections of logical blocks.

In a test of SQL Server query processing, openBench Labs benchmark findings revealed that V-locity, on a server running SQL Server, enabled higher transaction per second (TPS) rates and improved response time by reducing I/O processing on storage devices. What's more, in a SAN- or NAS-based storage environment, V-locity Server reduced I/O stress on multiple systems sharing storage resources. Overall, V-locity Server can improve scalability by reducing average response time and enabling SQL Server to support more users.

Snapshot of Findings

1) **In SQL Server 2012 processing tests involving 1 to 32 daemon processes, V-locity Server improved transaction performance by an average of 55%, before the I/O subsystem of our server began to physically limit the disk IOPS rate.**
2) **By improving the average response time of transactions by 33%, V-locity Server enabled SQL Server to scale enough to handle the transactions of 62% more daemon processes than the database could sustain without V-locity Server.**
3) **V-locity Server prevented 96% log file fragments from being created during a typical series of stress tests, contributing to the increase in transaction processing rate and improved response time.**

Executive Briefing: Condusiv's V-locity Server Boosts Performance of SQL Server 2012 by 55%

Jack Fegreus
May 20, 2013

Optimizing I/O for Increased Throughput and Reduced Latency on Physical Servers

With the release of SQL Server® 2012, Microsoft® is positioning SQL Server as a database for use with both large structured and unstructured (big data) applications. Whether with traditional data warehouse scenarios, or as a link between unstructured data platforms such as Hadoop, sites with large databases with large numbers of rows are now a key target, which makes SQL Server 2012 a good way for CIOs to initiate big data pilot projects.

Under Test: Server I/O Acceleration
Condusiv Technologies' V-locity Server

1) **V-locity's read optimization (IntelliMemory™)** technology reduces disk I/O requests by predictively caching active data within the server to increase local IOPS performance and reduces overhead on shared storage devices by reducing physical read I/O requests.
2) **V-locity's write optimization (IntelliWrite®)** technology provides continuous I/O write optimization using dynamic intelligence when creating or extending files to behave sequentially, eliminating split and other unnecessary I/Os for greater sequential throughput performance on both writes and subsequent reads.
3) **By optimizing I/O processing** on a server running V-locity, less I/O traffic is directed at an underlying SAN- or NAS-based storage system. Moreover, with IntelliWrite and IntelliMemory, both TP throughput and average response time were more predictable and scalable as the TP rate increased 55% and response time was improved by 33% over all test loads of 1 to 32 daemon processes on SQL Server 2012.
4) **V-locity's advanced I/O optimization** technology is compatible with all advanced storage features, such as replication, de-duplication, thin provisioning, and snapshots.

To provide a test case and example of a large mission-critical application that could not be fully cached by SQL Server 2012, and would be highly dependent on disk access rates for essential performance, openBench labs created a 30GB instance of the ANSI AS3AP benchmark database and populated it with 32.5 million rows of data.

For IT operations, optimizing I/O of transaction processing applications is complicated by the difference in the rate that CPU and memory performance have advanced versus the rate that disk access times have improved. Attempts to deal with these issues often create CPU bottlenecks as processing stalls waiting for the delivery of data.

With I/O access time highly dependent on disk drive mechanics, IT frequently turns to costly hardware solutions involving solid-state drives (SSDs). Such hardware solutions, however, suffer from dependence on a hardware platform, under-utilization of resources, lack of flexibility with changing workloads, and high costs.

In this analysis, openBench Labs examines the ability of Condusiv Technologies' V-locity Server to maximize I/O performance in a high-TP SQL Server environment. By efficiently optimizing the way data is read from and written to disk for Windows systems, V-locity Server optimized the processing of database transactions sent from another application that was used to generate multiple SQL Server client requests.

Running a specific collection of SQL transactions for up to 32 daemon processes (1 daemon generating the equivalent of 70 normally-queued user processes) with randomly-generated values, such as indices and row numbers, we initiated transaction loads for varyingly-sized groups of users. Over multiple load tests, we measured an average increase in SQL Server transactions of 55% with V-locity Server.

Removing Barriers to I/O Performance

To optimize I/O, V-locity Server has been designed to resolve two very important issues:

1) **V-locity eliminates nearly all unnecessary I/O operations at the source when writing a file, which eliminates all unnecessary I/O operations on subsequent reads.**
2) **V-locity caches frequently-accessed data to keep read requests from traveling the full distance to storage and back.**

> *"Using sophisticated pattern analysis to determine which blocks will be placed in cache, IntelliMemory offloads a significant portion of I/O read operations from logical disks."*

IntelliWrite Technology

V-locity Server solves the important issue of unnecessary I/O generation with IntelliWrite technology. IntelliWrite prevents the Windows OS from storing files as disjointed block sets in its logical block space representation of a logical storage volume.

In a SAN- or NAS-based environment, the problem of unnecessary I/O is compounded as multiple systems share the same set of storage resources, which results in highly random I/O behavior pushed down to the disk subsystem. The generation of unnecessary I/Os not only slows the speed of the originating system, it also slows other systems sharing the same storage arrays.

To solve this, IntelliWrite adds more intelligence to the way the Windows OS preallocates file space in order to continuously restructure writes in a coherent manner, store files as contiguous sets of blocks, and prevent performance penalties. Moreover, when a file is accessed and modified in the future, IntelliWrite will automatically restructure that file for optimal I/O performance.

IntelliMemory Technology

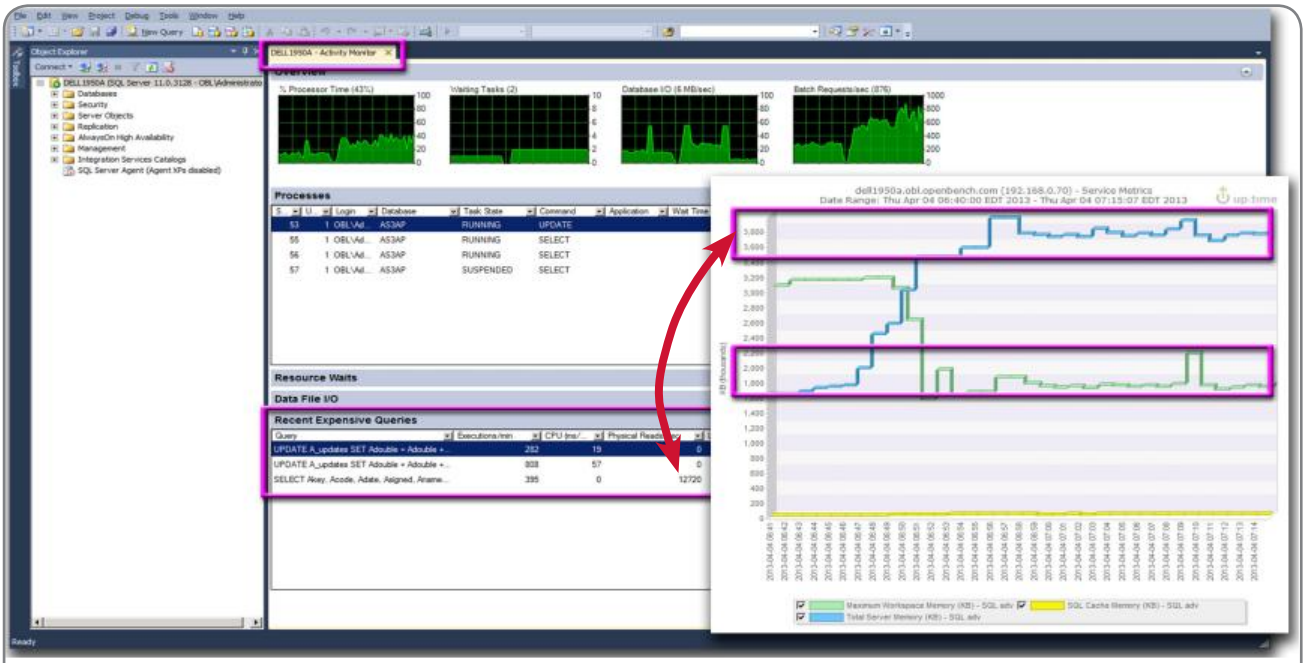
Optimizing writes and eliminating unnecessary I/O operations does not resolve all important data access issues, especially when reading data. To provide a full spectrum of I/O optimization, V-locity Server implements IntelliMemory, a highly efficient data caching solution that leverages available server memory to provide faster access to data and dramatically improved throughput. Using sophisticated pattern analysis to determine which blocks will be placed in cache, IntelliMemory offloads a significant portion of I/O read operations from logical disks. More importantly, by offloading physical I/O, V-locity Server improves performance on all systems sharing storage resources via a SAN or NAS.

The Test

To assess the performance capabilities of V-locity Server, we configured a test scenario using three servers. The following is an overview of the test environment and process:

1) **A Dell 1950 PowerEdge server, with a quad-core CPU and 8GB RAM, was used to run Windows Server 2012, V-locity Server and SQL Server 2012.**
2) **We set up an independent logical disk for the ANSI AS3AP benchmark database.**

SQL Stress Test



We configured a 30GB instance of the ANSI AS3AP benchmark database, which had 32.5 million rows in data tables. With 8GB of RAM on the server, the entire database could not fit within the SQL Server cache. More importantly, SQL Server claimed 3.5GB of RAM; however it utilized less than 2GB as a workspace. As a result, a significant amount of I/O was directed at the disk subsystem.

For our tests, we configured the database with 32.5 million rows, which translated into a 30GB database.

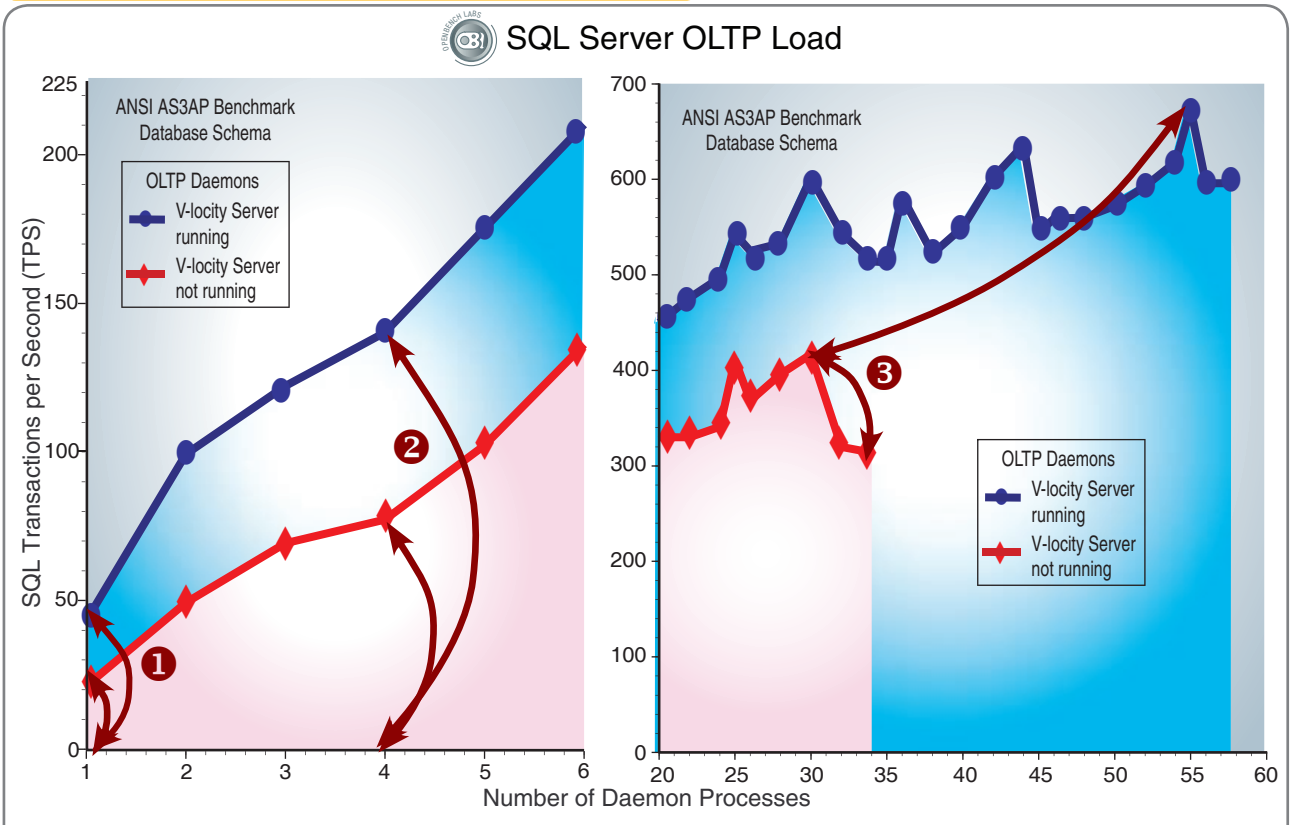
3) **We configured SQL Server to use a maximum of 3MB of server memory.**
4) **To generate both a batch and an interactive transaction processing load of SQL queries, we configured two servers with 2 CPUs running Windows Server 2008 R2 and a SQL load generator, which used a random number generator to modify scripted shell transactions directed at the target database server.**
5) **We generated queries for each daemon user and each batch process by inserting random data into fixed templates such that each TP query accessed one row and each batch processing task accessed at least 100 rows at a time.**
6) **We launched all batch transactions in stress mode without pausing for user think time or keyboard typing time.**

The Results

Our goal was to assess the ability of V-locity Server to create an environment in which our SQL Server 2012 benchmark would be able to reach maximum performance and scalability. To achieve our goal, we first created a set of jobs that represented a fixed level of batch production processing. In particular, three simultaneous batch jobs independently accessed and updated a minimum of 100 database rows at a time as rapidly as possible.

To our base level of production processing, we added a sequence of TP daemon processes to represent interactive users, who would typically be creating and placing online orders. To maximize application stress on our system, we introduced no pauses for queuing time and forced each TP daemon to continuously launch simple queries that accessed one row of data per transaction.

V-locity Server TP Performance Improvement



V-locity Server did not change the internal scalability of our SQL benchmark. Instead, V-locity created a more responsive environment in which the benchmark could leverage storage and memory resources as it ran. In particular, We measured a 73% lower initial response time—22 ms vs. 38 ms—for our benchmark and a base transaction rate❶ of 44 TPS, which was twice that without V-locity Server. We added daemons to increase the transaction rate to 200 TPS, which is typical of high-volume TP applications. More importantly, as we increased the TPS rate❷ to 200 TPS, daemon load TPS rates averaged 89% higher with V-locity Server running.

At the other end of the load spectrum, our 100 ms limit on response time provided a large enough window to reach the maximum TPS rate that our physical configuration could sustain either with or without V-locity Server. When we reached those limits❸—417 TPS running 30 daemons without V-locity and 677 TPS running 55 users with V-locity—we could add fewer than five users without exceeding the average response time limit, at which point the TPS rate plummeted.

To test scalability, we added five TP daemons at a time, until the average query response time for each daemon exceeded 100 ms. As a result, the number of queries processed and the average query response time measured on each daemon load cycle provided a measure of the ability of V-locity Server to create an optimal environment in which to run an application driven by SQL Server 2012.

We began by running our stress tests on SQL Server 2012 with the default memory configuration, which in theory allowed SQL Server to utilize all of the memory provisioned in our Dell PowerEdge server. On this test, we measured the amount of memory claimed by SQL Server 2012 and the amount of memory that SQL Server 2012 utilized. In all of these tests, SQL Server 2012 conservatively claimed about 3GB of memory and utilized less than 2GB of the 8GB that were physically installed.

Next, we reset the maximum memory for SQL Server from its default value to 3GB. When we made this change, overall TP throughput improved between 5% to 10% on nearly all of our test loads. As a result, we imposed a 3GB memory limit for all further testing.

The Results: A Closer Look

The following table highlights the test results:

V-locity Server Performance: SQL Server Stress
1 to 58 Daemon Loads Applied to a 30GB Database

I/O Activity	TPS Rate With V-locity	TPS Rate Without V-locity	TPS Rate Delta	Response Time With V-locity	Response Time Without V-locity	Response Time Delta
1 daemon*	44*	20	22	22 ms.	38 ms.	(16 ms.)
5 daemons	176	103	73	28 ms.	43 ms.	(15 ms.)
10 daemons	281	188	93	37 ms.	53 ms.	(16 ms.)
15 daemons	383	315	82	42 ms.	64 ms.	(22 ms.)
20 daemons	450	330	120	45 ms.	65 ms.	(20 ms.)
25 daemons	543	411	132	50 ms.	72 ms.	(22 ms.)
30 daemons	602	417	185	50 ms.	82 ms.	(21 ms.)
32 daemons	544	320**	224	61 ms.	96** ms.	(35 ms.)
35 daemons	513			67 ms.		
40 daemons	532			70 ms.		
45 daemons	544			80 ms.		
50 daemons	571			89 ms.		
55 daemons	677			97 ms.		
58 daemons	601***			98*** ms.		
V-locity Server Database I/O Caching (IntelliMemory)						
Disk Reads	60,466 - 58%	N.A.				
Cache Reads	31,568 - 42%	N.A.				
I/O Latency	1.2 ms.	6 ms.				
V-locity Server Database Log File Fragment Prevention (IntelliWrite)						
Log File Fragments	768 Prevented	803 Fragmented				

*One daemon generates the equivalent of 70 normally-queued user processes
**Without V-locity Server running, we reached the benchmark's maximum response time at 32 daemon processes
***With V-locity Server running, we were able to reach 58 daemon processes

When we ran SQL Server with V-locity Server, IntelliMemory came into play for reads not cached by SQL Server. In particular, IntelliMemory cached 42% of the physical reads associated with running SQL Server. By lowering the number of physical I/Os, V-locity improved the overall IOPS performance of the server's entire I/O subsystem, which included faster writes to log files which in turn reduced log file wait time.

For workloads of 1 to 32 daemons (approximately 70 normally-queued users per daemon), V-locity Server improved SQL Server TP performance by an average of 55%. Similarly, V-locity Server improved transaction response time by an average of 33%. With the average transaction response time on the initial single-user test reduced by more than 100% in a V-locity Server environment, we were able to complete tests on which the average query response time was less than 100 ms for workloads that included up to 58 daemons—without V-locity the system maxed out at 32 daemons.

What's more, we were able to scale workloads and process close to the maximum number of transactions that our physical server was capable of handling while keeping the average transaction response time under 100 ms.

Bottom Line

On a server running SQL Server 2012, V-locity Server created an environment that enabled up to 55% higher TPS rates, improved transaction response time by 33%, and enabled SQL to process 62% more transactions at peak transaction rates.

As a result, IT has a powerful tool to maximize the ROI associated with any business application initiative driven by SQL Server at the back end.