# Evaluation Report: Improving Database Performance with Log Partitions on Microsemi Flashtec NV1616 NVRAM Drives

***Evaluation report prepared under contract with Microsemi***

## Executive Summary

High-performance enterprise-class storage is an expensive but essential part of a robust database platform. Businesses trying to meet demanding service levels already know that legacy spinning hard disk drives can no longer sustain the bandwidth, response times, and transaction levels needed to support modern application requirements. Businesses are increasingly deploying flash as primary storage. Application performance expectations are adjusting to exploit those improvements. However, SAS and SATA SSDs, and even new NVMe devices, still cannot deliver I/O as quickly as modern processors are able to request it. Critical applications benefit from any boost possible. An ideal solution for maximizing database performance is an in-memory database, but these come with unique challenges, not the least of which are scalability and preserving data integrity in the event of an outage. Microsemi offers another option.

Databases, as well as some filesystems and many applications, perform update request logging before executing changes to data. This is essential for rollback or recovery in the case of disaster. However, logging is only as fast as the speed of the storage media, and it must be done before committing database writes. With DRAM as the fastest memory tier in a server, logging to DRAM would save precious time that can be returned to the storage device, but that is risky. An outage that corrupts the log can leave a database unrecoverable. Microsemi's Flashtec NV1616 NVRAM Drive delivers DRAM speed, but with data persistence, making it a viable option as a database log device.

Microsemi commissioned Demartek to evaluate the Flashtec NV1616 NVRAM PCIe card as a log device for an Oracle database executing a write-heavy OLTP workload. Demartek deployed Oracle 12c on Oracle Enterprise Linux to support a transactional workload with a relatively high read-to-write ratio. Performance, measured by the average number of database transactions per minute, with redo logs deployed on SATA SSDs was compared with logging to Flashtec NVRAM. A 15% improvement in the number of transactions was recorded by placing logs on just two mirrored Flashtec NV1616 cards.

## Microsemi Flashtec NV1616 16 GB NVRAM PCIe Drive

The Microsemi Flashtec NV1616 is a 16 GB NVRAM drive. By combining DRAM performance with the persistence of flash memory, the Flashtec NV1616 drive delivers extremely low I/O latency. Data in DRAM is backed up to flash, with super capacitor protection in case of power failure.

The Flashtec NV1616 comes standard with the following features.

- x8 lane, PCI Express® 3.0 host interface
- Low profile MD2 PCIe®, Add-In-Card form factor
- Both NVMe and Direct Memory Modes
- 16 GB memory capacity
- Flash module backup store
- Tethered super capacitor module backup power supply
- 5 years operational lifetime
- < 30 sec backup & restore times

Figure 1 - Flashtec NV1616

Ideal applications include any scenario where a small amount of DRAM can be harnessed for measurable performance gain. Potential deployment options may include, but are not limited to the following list.

- Write Cache for Low-Latency Response Time
- 64bit Addressable Persistent Metadata Memory Region
- Persistent Shared Memory for Scale-Out Clustered Systems
- High Performance Journaling or Write Ahead Logging
- Flash module backup store
- Persistent Cache for Fast Cache Rebuild

Demartek deployed the Microsemi Flashtec NV1616 NVRAM drive as a redo log device for a transactional database. The hypothesis was that logging transactions that update a database on low latency NVRAM would drive up performance of the entire database, returning precious microseconds of I/O response time to the more traditional storage for writing the actual data. A cascade effect was anticipated where faster writes would move queued I/Os through the system more quickly or potentially deliver new data immediately to be read by other application processes.

## The Transactional Workload

Logging records all database update activities before they occur. It's a critical tool for rolling back transactions to return a database to a prior state or to restore data integrity in case an interruption in service occurs that disrupts database updates. Since logging is only applicable to transactions that make changes to the database, a workload that is fairly heavy in write transactions was preferred for this evaluation. We chose to deploy the open source HammerDB tool and a workload modelled on an online retailer taking and delivering orders. Roughly 30% of this workload's transactions are write transactions.

We deployed the workload on a Dell R920 server installed with Oracle Enterprise Linux 7 and Oracle 12c Enterprise Edition. The database was configured with a single application tablespace of 800 GB and three redo logs of 4 GB each. A 24 drive all-flash array was the storage target. This array was configured with RAID 5 and a single 1 TB volume was mapped to the host as a storage target for the database data partition, control files, and temp space for all test scenarios. Three different types of storage were provisioned for the redo logs, each tested separately. Redo logs were placed on a 16 GB logical volume in all scenarios.

The first test, the baseline, placed redo logs on the same 24 drive RAID 5 array as the data volume. For the second configuration, logs were moved to a volume created from two on-host mirrored NVMe PCIe SSDs[1]. The last test scenario saw the logs migrated to a volume built from two on-host mirrored Microsemi Flashtec NV1616 NVRAM PCIe drives. Linux MD RAID was used to create the mirrored volumes from the on-host PCIe drives. In all cases, a 16 GB partition was created for database redo logs.

[1] These NVMe SSDs were 1.6 TB in size, resulting in a great deal of wasted drive space. A particularly active write-heavy database might suggest a need for significant over-provisioning if write endurance limits are a concern, but in most cases this type of deployment would likely not see a good return on investment.

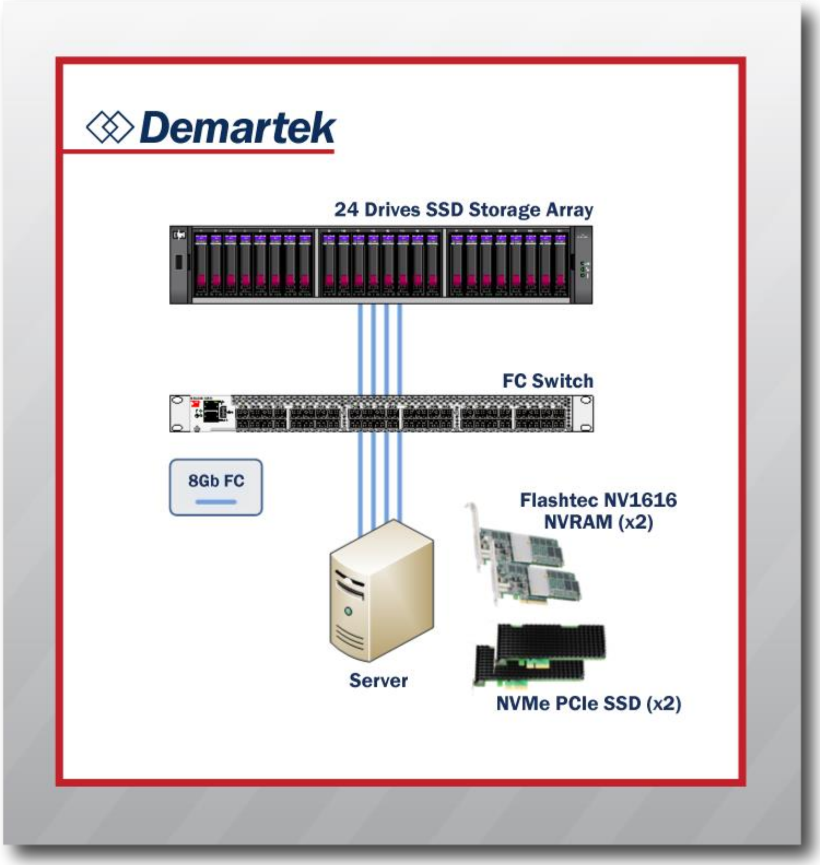

Figure 2 - Testbed diagram

To make certain that I/O was primarily served by the storage devices and not cached into system memory, database SGA and PGA were limited to a total of 15 GB. Testing was performed with database and log partitions deployed as ext2 filesystems. The workload was allowed a ten minute warmup interval followed by ten minutes of data collection. Transaction counts were averaged every minute.

## Results and Analysis

Database transactions are real work performed by the database, and as such, seemed a reasonable metric to evaluate whether faster redo logs would improve database performance. We recorded a baseline of 507,000 transactions per minute with redo logs on the same RAID 5 SSD array as the data partition. When moved to a PCIe NVMe SSD, that value increased by 9%, to 554,600 transactions per minute. Logging to Microsemi Flashtec NV1616 NVRAM drives improved database performance 15% over the SSD baseline to 595,100 transactions every minute.

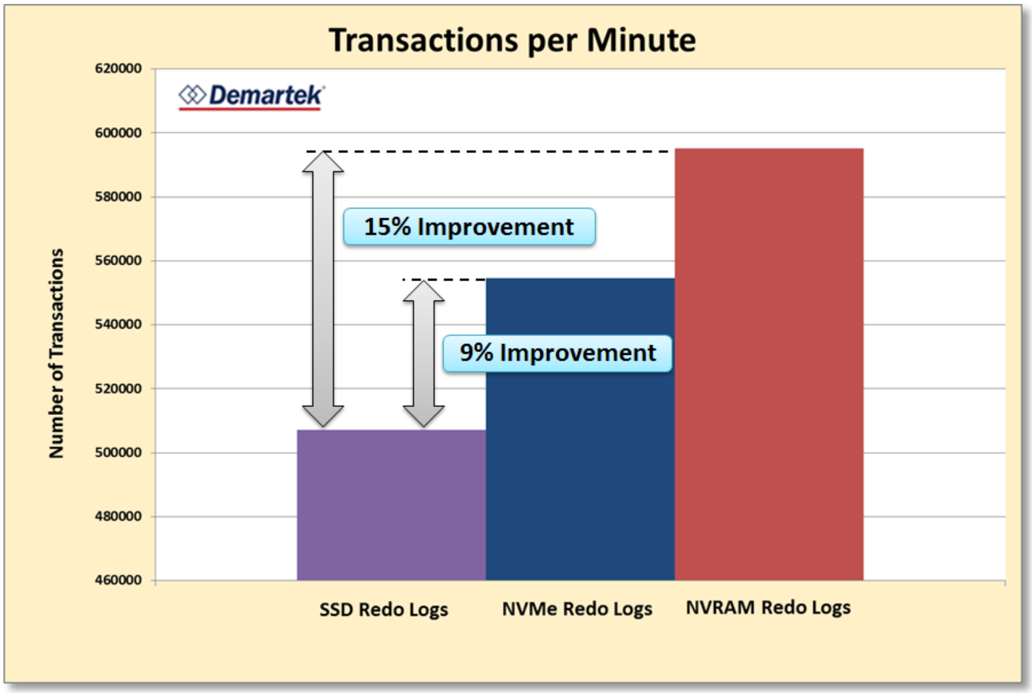

Figure 3 - Database transactions per minute by redo log device

These performance improvements are a direct result of moving time-consuming logging activities to faster storage tiers, effectively returning IOPS and tens-to-hundreds of microseconds back to the SSD array to do application work.

Logging to an NVMe tier did result in significant gains, but it comes with a cost. Well-designed applications will not typically need hundreds of gigabytes or terabytes of log space. Squandering the majority of an NVMe drive's capacity is not likely to be a good return on investment. On the other hand, the 16 GB of NVRAM provided by the Flashtec NV1616 drive (two drives mirrored in this evaluation), accommodated the space requirements of this workload with no waste, while delivering greater performance than logging to NVMe.

The reason the Flashtec NV1616 can deliver this type of performance boost is latency, or rather, the lack thereof. Figure 4 displays the response times of log writes for each media. The SATA SSDs within the array delivered writes at an average of 400 microseconds, where log writes on NVMe were serviced in 130 microseconds. These are reasonable response times for SATA and NVMe flash drives. However, the Microsemi Flashtec NV1616 NVRAM response time for log writes is 6.5 times faster than NVMe and 20 times faster than SATA SSD at an average of 20 microseconds per log write.

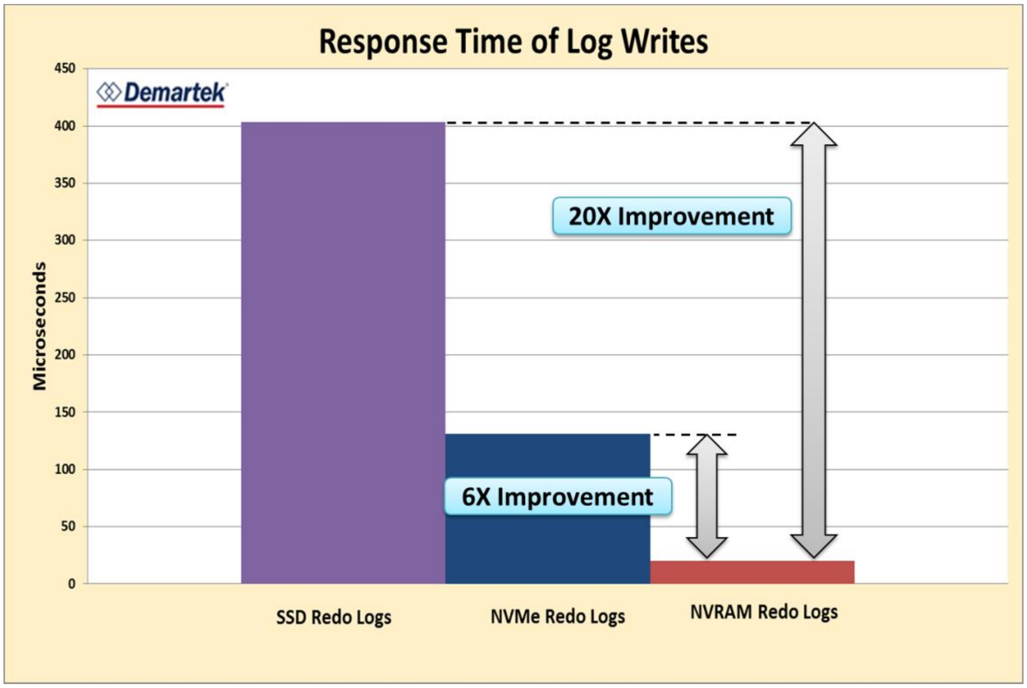

Figure 4 - Response times of redo log writes by drive type

Placing log write I/O on the fastest storage tier available effectually gives back I/O potential to the array for servicing application processing, increasing transactional I/O totals as the data demonstrates.

## Summary and Conclusion

Logging update requests is a necessary evil, but it does rob storage systems of valuable time and IOPS that could otherwise be put to use by applications. Newer storage technologies are emerging; NVMe drives are making news with record performance numbers, but NAND flash still lags behind DRAM's response time. Offloading logging to DRAM can have a substantial impact on application performance, as demonstrated through this evaluation. Giving DRAM persistence through a super capacitor backup and flash caching removes the risk of critical data loss from power interruptions. As a final benefit, NVRAM has the unlimited endurance of DRAM, unlike flash which eventually wears out through repeated writes/erase cycles.

A small amount of NVRAM can go a long way. The 850 GB database used for this evaluation required less than 15 GB of redo logs. This ratio of data to logs will vary by application (and quality of application coding), but in many cases, a small quantity of NVRAM can be deployed compared to database size. This may provide a much greater return on investment over the purchase and upgrade of new storage systems and extend the operational life of existing systems. With a write-heavy application, a small log partition on flash may also have the adverse effect of reducing the life span of NAND flash cells by concentrating log writes to a limited number of cells. Instead of overprovisioning log partitions to compensate for wear, the unlimited write endurance of NVRAM achieves a better ROI by properly sizing logs for the actual work being performed.

Business objectives should drive investment in new technology. Deploying Microsemi Flashtec NV1616 NVRAM for redo logging can boost the performance of write-heavy Oracle database applications. We would suggest that businesses looking to improve database performance consider the benefits of adding NVRAM to their infrastructure as an affordable option to wholesale replacements of storage technologies.

The most current version of this report is available at http://www.demartek.com/Demartek_Microsemi_Flashtec_NV1616_NVRAM_Database_Performance_2016-06.html on the Demartek website.