Boost Oracle Data Warehouse Performance Using SanDisk® Solid State Drives (SSDs)

Prasad Venkatachar (prasad.venkatachar@sandisk.com)
September 2014

www.SanDisk.com

Table of Contents

Executive Summary

In recent years, the IT industry has witnessed the adoption of high-performance, low-latency storage, deployed in servers and storage arrays, to accelerate database workloads.

With its Oracle Database 12c release, Oracle introduced an exciting set of new features for Cloud and Data Warehouse environments. Driven by emerging user demands for faster processing, these new features are increasingly being considered as Oracle customers evaluate their data warehouse environment for new deployments—or for upgrades to their current environments. Database I/O throughput and fast response times for database queries are the two driving factors to improve the performance of these data warehouse environments.

To address these performance concerns, database storage media is, in many cases, getting replaced by flash storage for an increasing number of deployment scenarios. For many customers, the use of traditional hard disk drives (HDDs) as a storage media in these data warehouse environments does not meet some, or even most, of the performance challenges in today's data centers. Often, these customers spend precious months looking for alternative technology approaches that would help them to cope with this data growth, now and the near future.

The objective of this technical paper is to highlight the performance benefits of using SanDisk Optimus SAS Solid State Drives (SSDs) over HDDs. This paper describes an optimized way of using flash storage for Oracle databases, and it shows the advantages of using SanDisk Optimus SAS SSDs for data warehouse deployments.

SanDisk Optimus SAS SSDs

SanDisk is a leader in flash storage solutions. SanDisk's portfolio of solid-state drives (SSDs) support the megatrends in the industry that are driving new deployments, including Cloud, Big Data/Analytics, Mobility and Social Media. Databases such as Oracle provide storage for many of these new workloads, and these applications must deliver optimized performance to support timely business results.

The SanDisk Optimus SAS SSDs address a wide range of enterprise storage and server applications. Optimus SAS SSDs, support high performance for applications and databases, while providing consistent quality of service (QoS). Their wide range of storage capacities, spanning from 200GB to 4TB (Optimus MAX™ SSD) in the Optimus SAS SSD product portfolio, enables more flexibility for IT deployments.

It's worth noting that the Mean Time Between Failures (MTBF) for the SanDisk Optimus SAS SSDs is high—2.5 Million Hours MTBF—with a warranty period of five years. Optimus SAS SSDs are protected by SanDisk's Guardian™ Technology platform, which provides FlashGuard™, DataGuard™ and EverGuard™ capabilities that increase the durability, recoverability and prevention of data loss and corruption. Specifically:

- FlashGuard technology reliably extracts significantly more usable life from MLC flash than would otherwise be provided by the standard specifications for this commercial-grade NAND.
- DataGuard technology provides full data-path protection, ensuring that data will be safe throughout the entire data path. DataGuard provides the ability to recover data from failed page and NAND blocks.
- EverGuard technology prevents the loss of user data during unexpected power interruptions.

For additional details, refer to the Guardian Technology data sheet on   www.sandisk.com.

Oracle Database 12c

This version of Oracle, which was introduced in June, 2013, has introduced many new features, including expanded support for cloud-based workloads. For example, Oracle Multitenant introduces a new architecture that supports a multitenant container database that holds many pluggable databases. This enables better support for both Cloud and Big Data/Analytics workloads in its Oracle Database 12c Release, so that customers can consolidate multiple databases into a single-container database for easier management and IT efficiency.

This approach reduces server sprawl (for virtual servers, or VMs, that support databases), which is a chief concern of first-generation adopters of virtualization in database environments. The benefits of SSD technology build on the enhancements to Oracle Database 12c, allowing customers to achieve a better consolidation ratio than they would be able to do with HDDs.

Overall, Oracle Database 12c has hundreds of new features, including the support for storage tiering heat-maps for data warehouse environments—and other enhancements for performance, security and data protection. (More details can be found on the Oracle website at www.oracle.com). Together, they provide a broad range of new capabilities for the worldwide database marketplace that leverages Oracle database products.

This paper is focused mainly on SSD and HDD performance in data warehouse workloads. For this reason, the rest of this paper will provide more details about the specific testing that was done to demonstrate the advantages of using SSDs in Oracle database deployments.

Flash Advantage

The Oracle Database 12c engine makes good use of a server's multi-core architecture for its parallel execution activities. So, before going ahead and looking into SanDisk SSD performance advantages on Oracle Database 12c, let's look at the operating speed for processors (CPUs) and storage devices (e.g., RAM, SSD, and HDD). The CPU operates at nanosecond speeds, and it has a storage cache in the form of its L1, L2, and L3 caches for storing intermediate computational data on its way into, and out of, the microprocessor cores. These on-chip caches are usually in the MB range.

Next, we considered the role of DRAM memory in accelerating data warehouse performance. Usually, mid-range servers are configured with DRAM that is in the gigabytes (GB) range. It's important to understand that DRAM access operates at microsecond speeds—rather than millisecond speeds that are more typical of HDD drives. In terms of Oracle Database 12c, the Oracle database intelligently stores frequently accessed data in the DRAM memory—and the data read from the memory is referred to as logical I/O, as shown in Figure 1. In data warehouse environments, only a partial query dataset is cached in the DRAM memory, due to the sheer volume of data size and bulk of the datasets that are fetched from system drives.

Next, we looked at the HDDs in the system, where the bulk of the data for the user queries will be stored—and from which it will be fetched for further processing. As shown in Figure 1, the I/O operations from disk are referred to as Physical I/O. HDD data access speeds are in the milliseconds range (which amounts to thousands of microseconds), and the data storage capacity in these drives is in the Terabytes (TB) in range.

The difference between the operating speed of memory and that of the HDDs is, therefore, several orders of magnitude. That explains why database performance involving physical I/O results in a slowdown of overall database operations every time that data has to be fetched from HDDs. However, this performance challenge can be better addressed by solid state drives (SSDs), which operate with a latency of a 100 microseconds or less. And while SSDs and HDDs can both address multiple terabytes of data with one drive, that range of latency is much closer to DRAM operating speeds. So organizations that deploy SSDs gain an immediate performance benefit for their data warehouse workloads.
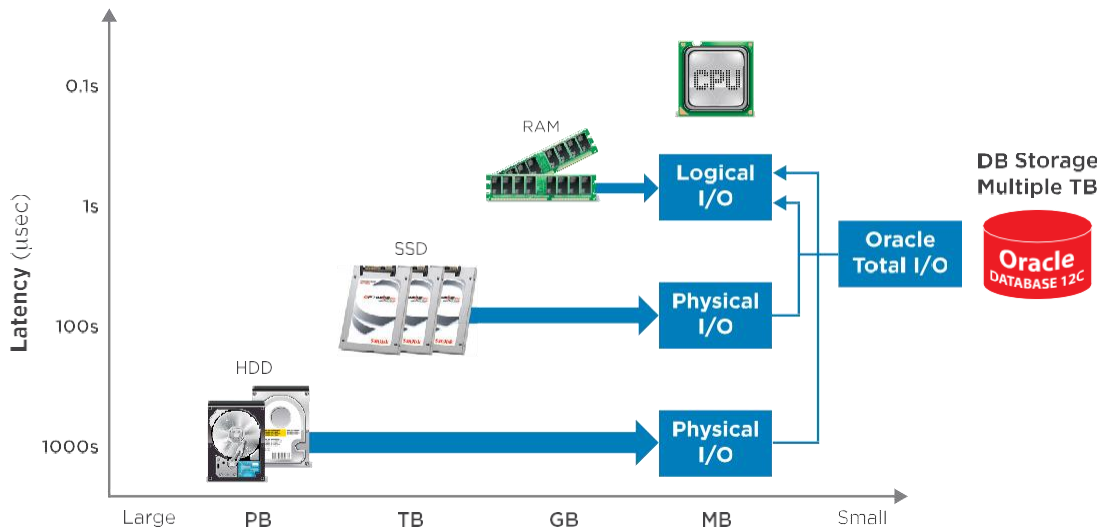

Figure 1: Compute and Storage for Oracle

Test Methodology

Our test consisted of using HammerDB, an open-source testing tool for Oracle data warehouse workloads, comparing the performance data points for SanDisk SSDs and HDDs. The size of the Oracle Database 12c databases, as used in this test comparing SSD and HDD support for data warehouse workloads, varied from 100GB to 300GB to 1TB. For more information on HammerDB, refer to hammerora.sourceforge.net.
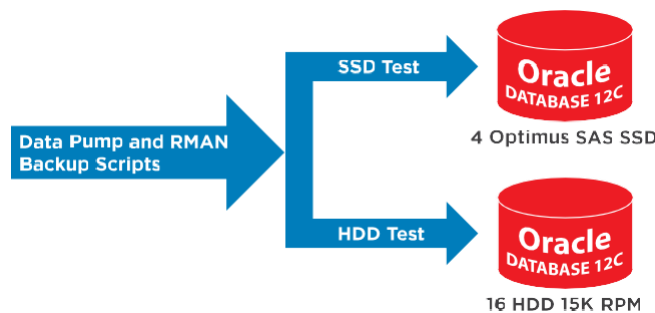
Test Topology

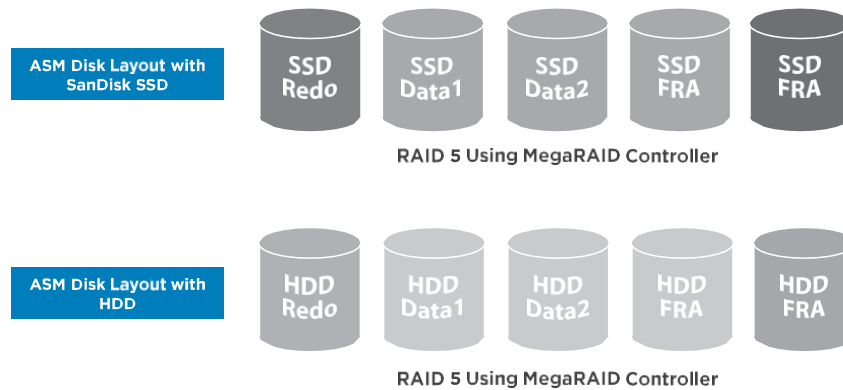**Figure 2: SSD and HDD Testing Using   HammerDB**

Figure 3: Storage Layout with SanDisk SSD and HDD

Component	Details
Server and Client	Supermicro® 2-socket 16-core 2027R-E1R24N server with Intel® Xeon® E5-2690 processor, 64GB RAM memory
Storage SSD Configuration HDD Configuration	8 Optimus SAS SSDs from SanDisk, each with 800GB 16 SAS 15K RPM HDD, each with 400GB
Operating System	Red Hat® Enterprise Linux (RHEL) 6.5
Database	Oracle® Database 12c (12.1.0.1)

Table 1: Hardware and Software Configuration

Test Setup

For our tests, we deployed a Supermicro server with a fully populated set of 24 drive bays. The internal drive was used for setting up the Red Hat Enterprise Linux 6.5 operating system and the Oracle Database 12c software. The Oracle database was created on SanDisk Optimus SAS SSDs—and then the same database was switched between SSDs to HDDs during testing to compare the test results.

Server Optimization

The following server configuration settings were enabled on the server:

1) Intel processor hyper-threading was turned off. For data warehouse workloads, we found that results were better when the hyper-threading option was turned off because of workload characteristics.
2) The configuration was NUMA-enabled to support non-uniform memory architecture.
3) The configuration was Intel Turbo Boost-enabled.

Flash Optimization

The following settings were enabled to take advantage of the presence of solid-state drives (SSDs):

• The LSI MegaRAID Storage Manager was used to create the RAID5 configuration for both the SSD and HDDs being tested. The following settings were changed:

    a)   Write-back policy enabled: The SanDisk SSDs had battery backup to ensure data durability and this mode provided superior performance compared to the write-through policy.

    b)   Read-ahead policy enabled for drive read operations.

• NOOP Linux Scheduler: NOOP scheduler provides better performance for SSD drives.

    — echo noop > /sys/block/sda/queue/scheduler

• Queue depth increased to 1024: Higher queue depth shown better performance   results.

    — echo 1024 > /sys/block/sda/queue/nr_requests

Operating System Configuration

The best practices of Oracle configuration on the Red Hat Enterprise Linux 6.5 operating system incorporated the following:

Kernel Configuration

```
### Virtual Memory Settings ###
vm.swappiness = 0
vm.dirty_background_ratio = 3
vm.dirty_ratio = 80
vm.dirty_expire_centisecs = 500
vm.dirty_writeback_centisecs = 100

### Shared memory Settings ####
kernel.shmmax = 4398046511104
kernel.shmall = 1073741824
kernel.shmmni = 4096

### Network Port Range ###
net.ipv4.ip_local_port_range = 9000 65500

### Optimum Network settings ##
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576

## Asynchronous I/O Request Settings

fs.aio-max-nr = 1048576
## File Handle Max limit
fs.file-max = 6815744
```

SELinux was turned off

Shell Limits Settings

```
oracle soft nproc 16384
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
oracle soft stack 10240
oracle hard stack 32768
grid soft nproc 16384
grid hard nproc 16384
grid soft nofile 1024
grid hard nofile 65536
grid soft stack 10240
grid hard stack 32768
```

Red Hat Enterprise Linux 6.5 provides a "tuned" package for optimizing database storage using Oracle ASM. The "tuned" package automatically tunes the system for different workloads, leading to the improved performance benefit in using this package. The following steps were taken:

- Installed the "tuned" package.
- Set up the enterprise-storage profile settings.
- Started the "tuned" service with enterprise-storage profile settings.

Oracle Automatic Storage Management (ASM):

The following steps were taken:

- Downloaded, installed and configured the Oracle ASM packages—which manage how data is stored within the data warehouse.
- SSDs were configured with RAID 5 using the LSI MegaRAID Controller.
- The ASM drive groups relied on the RAID controller for redundancy.
- Three ASM data drive groups were created based on a scale factor of 100GB, 300GB, and 1000GB sizes.
- The Oracle ASM allocation unit size was kept at 4MB, based on Oracle's recommended best practices.

Database Configuration

- An Oracle single-instance database was created with Database Configuration Assistant (DBCA)
- Data tablespaces were created with a 32K block size, which is a typical setup in data warehouse environments, to increase I/O throughput for each I/O cycle

Test Execution and Monitoring

- HammerDB, an open-source software tool, was used for populating the 100/300/1000GB databases.
- Oracle Enterprise Manager (OEM) was used for monitoring Oracle performance and Linux scripts for system monitoring.
- An Automatic Workload Repository (AWR) report was created for each workload test.
- An initial test was executed on SSDs and, later, the database was migrated to HDDs to verify the difference in performance between the SSD and HDD deployments.
- HammerDB executed 22 complex data warehousing queries, and it generated an output file to measure query execution time for each of these 22 queries.
- HammerDB also provided an option to invoke parallel threads for each of the complex queries, so that the parallel execution timings could be evaluated.

Oracle Acceleration using SanDisk SSD

This section focuses on providing SanDisk SSD performance benefits, compared with HDDs, for the following three scenarios:

    a)   Data warehouse complex query execution timings.

    b)   Data warehouse parallelism benefits.

    c)   Data warehouse complex query execution timings, across various database sizes, being tested.

Data Warehouse Performance Analysis

Figure 4 below shows the elapsed time for the 22 data warehouse queries using SanDisk Optimus SAS SSDs, compared with other vendors' HDDs. For these queries, the Oracle database running on eight SanDisk SSDs performed 2.5X times faster than the same database running on 16 HDDs. It took twice as many HDD devices to run the same workload as was running on the eight SanDisk SSDs. Query 18 showed the most SSD benefit, with a 5X performance improvement over HDDs. It's important to note that Oracle Database 12c made good use of the low-latency SanDisk SSDs to provide improved query performance. The Oracle Database 12c internal technical aspects involved in the performance benefits will be discussed in the next section.

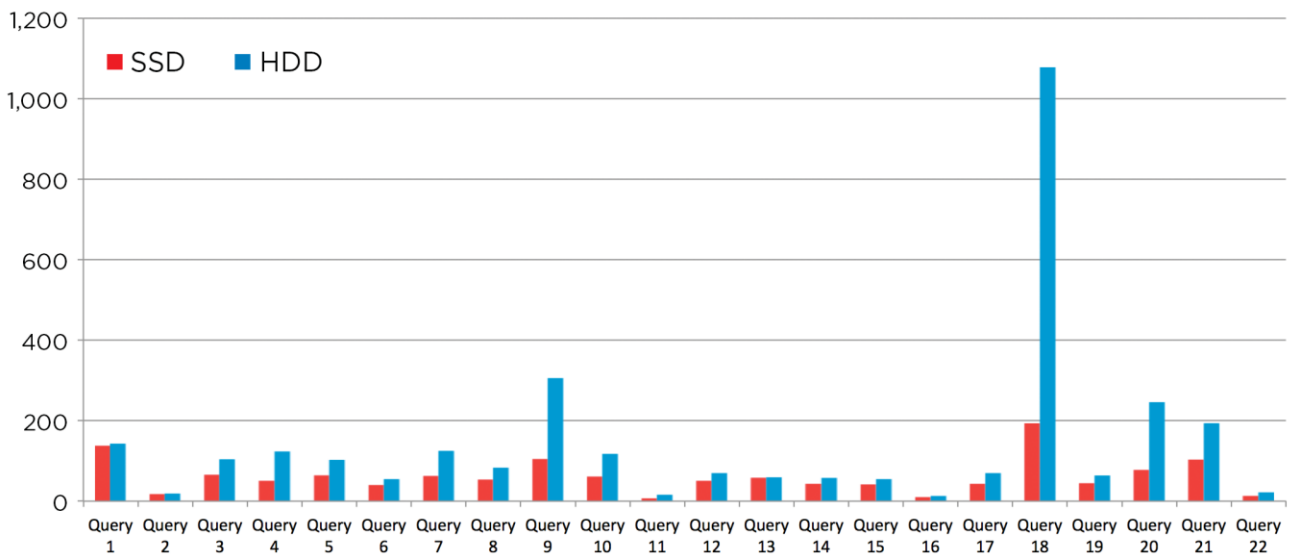## Data Warehouse Queries Performance

Figure 4: SSD VS HDD Elapsed Time

DB time is another key metric that indicates how well the server computing resources are being utilized for Oracle Database 12c performance. DB time is derived by two parameters: DB CPU, which is the time spent by the Oracle database performing actual work, and WAIT time, which is the amount of time spent by the Oracle database in wait events, such as I/O wait time needed to obtain the data from the drive.

This equation shows how these key metrics are calculated: DB Time $\begin{array}{l} = \text{DB CPU} + \text{WAIT TIME} \\ = \text{WORK Time} + \text{WAIT Time} \end{array}$

DB CPU or Work Event Analysis: DB CPU event indicates the amount of CPU resource usage in terms of the percentage consumed by user queries. Shorter query response times resulted from configurations that offered a higher percentage of DB CPU time. Based on the HDD AWR performance report that is shown in Figure 5A, the DB CPU is only 33.4%. In contrast, for SanDisk SSDs, the DB CPU is 75.2%, as shown by the AWR report in Figure 6A. This difference, as observed in our testing comparing SSDs and HDDs, demonstrates a clear benefit for Oracle customers who use SSDs. These SSD deployments resulted in faster query execution and faster business results.

WAIT Event Analysis: The HDD AWR performance report shows that the majority of TOP10 wait events are waiting for user I/O operations from HDDs to be completed. For HDDs, the TOP10wait event direct path reads contributed to 40% of DB time, with an average wait time of 7 milliseconds for a direct path read operation resulting in a total of 603K waits. This adds up to 4,172 seconds,

Direct Path Reads is the #1 Wait Event for HDDs
4,172 Seconds = 7 millisecond x 603, 973 Waits*

Direct Path Reads is the #2 Wait Event for SSDs
700 Seconds = 1 millisecond x 700,047 Waits*

* Total Wait Time = Average Wait x Number of Waits

which is the amount of time spent in waiting for user I/O. In contrast, the SanDisk SSD direct path reads is the second highest wait event, contributing to just 24.6% of DB time. With an average wait time of just 1 millisecond, the total wait time for this entire event is 703 seconds—a fraction of the 4,000+ seconds seen for the HDD-based configuration.

Event	Waits	Total Wait Time (sec)	Average Wait (ms)	% DB Time	Wait Class
Direct Path Read	603,973	4,172.6	7	40.4	User I/O
DB CPU		3,447.5		33.4	
Direct Path Write Temp	132,595	1,027.1	8	9.9	User I/O
DB File Sequential Read	203,042	669.3	3	6.5	User I/O
Direct Path Read Temp	199,340	579.7	3	5.6	User I/O
Local Write Wait	25,929	148.2	6	1.4	User I/O
DB File Parallel Read	25,422	84.9	3	.8	User I/O
DB File Scattered Read	5,596	15.2	3	.1	User I/O
Buffer Busy Waits	1,335	14.9	11	.1	Concurrency
CRS Call Completion	166	5.9	35	.1	Other

Figure 5A: HDD AWR Top 10 Foreground Events by Total Wait Time

Event	Waits	Total Wait Time (sec)	Average Wait (ms)	% DB Time	Average Active Sessions
User I/O	1,198,401	6,702	6	64.9	2.1
DB CPU		3,448		33.4	1.1
System I/O	79,943	195	2	1.9	0.1
Other	18,211	26	1	.3	0
Concurrency	2,665	16	6	.2	0
Scheduler	582	5	9	0	0
Commit	67	3	47	0	0
Application	17	1	66	0	0
Network	167,770	0	0	0	0
Configuration	4	0	5	0	0

Figure 5B: HDD AWR Wait Classes by Total Wait Time

Event	Waits	Total Wait Time (sec)	Average Wait (ms)	% DB Time	Wait Class
DB CPU		2,148.9		75.0	
Direct Path Read	700,047	703.9	1	24.6	User I/O
Direct Path Read Temp	53,905	32.2	1	1.1	User I/O
Direct Path Write Temp	11,242	3.3	0	.1	User I/O
PX Deq: Slave Session Stats	176	.7	4	0	Other
DB File Scattered Read	945	.6	1	0	User I/O
PX Deq: Signal ACK EXT	88	.5	6	0	Other
Enq: BF Allocation Contention	32	.3	10	0	Other
SGA: Allocation Forcing Component Growth	6	.2	33	0	Other
DB File Sequential Read	422	.2	0	0	User I/O

Figure 6A: SanDisk SSD AWR Top 10 Foreground Events by Total Wait Time

Event	Waits	Total Wait Time (sec)	Average Wait (ms)	% DB Time	Average Active Sessions
DB CPU		2,149		75.0	1.2
User I/O	767,488	740	1	25.8	.4
Other	16,511	3	0	.1	0
System I/O	8,433	3	0	.1	0
Network	379,334	0	0	0	0
Concurrency	116	0	0	0	0
Commit	20	0	0	0	0
Application	4	0	1	0	0

Figure 6B: SanDisk SSD AWR Wait Classes by Total Wait Time

Wait Classes Analysis

The AWR reports provide another important metric to gauge the level of Oracle database performance, which is calculated by multiplying the Wait Classes by Total Wait Time. The HDD AWR report as shown in Figure 5B shows that 64.9% of the time is spent in user I/O operations—and that user I/O operations for the SanDisk SSDs as shown in Figure 6B for the SanDisk SSDs were just 25.8% of DB time.

> **Summary:** The DB CPU metric, as discussed in the section above, shows only 33.4% utilization for Oracle running on HDDs, and 75% utilization for Oracle running on SanDisk SSDs.

Oracle Enterprise Manager (OEM) Performance Metrics

The Oracle Enterprise Manager (OEM) performance report provides testing data regarding database throughput and latency metrics. Based on this report, the Oracle instance running on HDDs was generating 1 GBps throughput, and had latency that was measured at 7 milliseconds, which matches with the Oracle AWR report. The same test, when conducted on Oracle instances that were running on

SanDisk Optimus SAS SSDs provided nearly 2GBps (twice the HDD-based throughput), with a maximum spike latency of 1.6 millisecond—a fraction of the seven milliseconds latency seen with the HDDs.
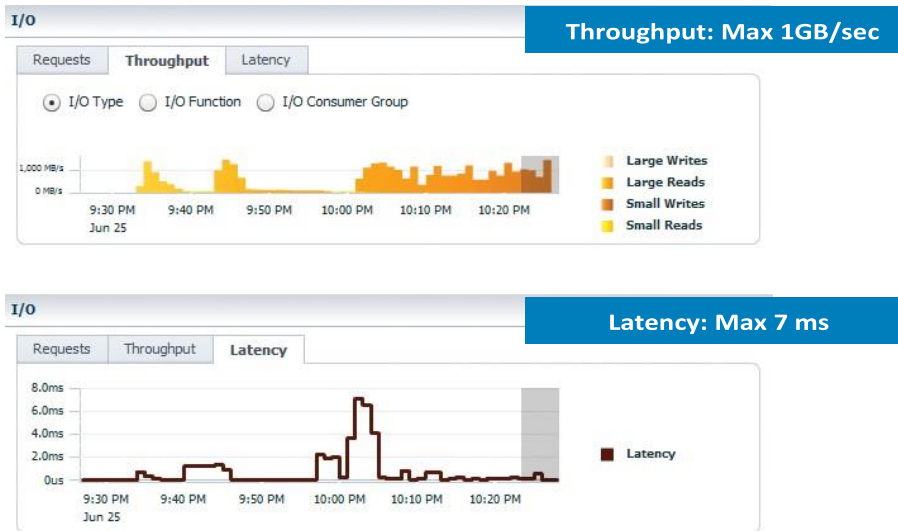

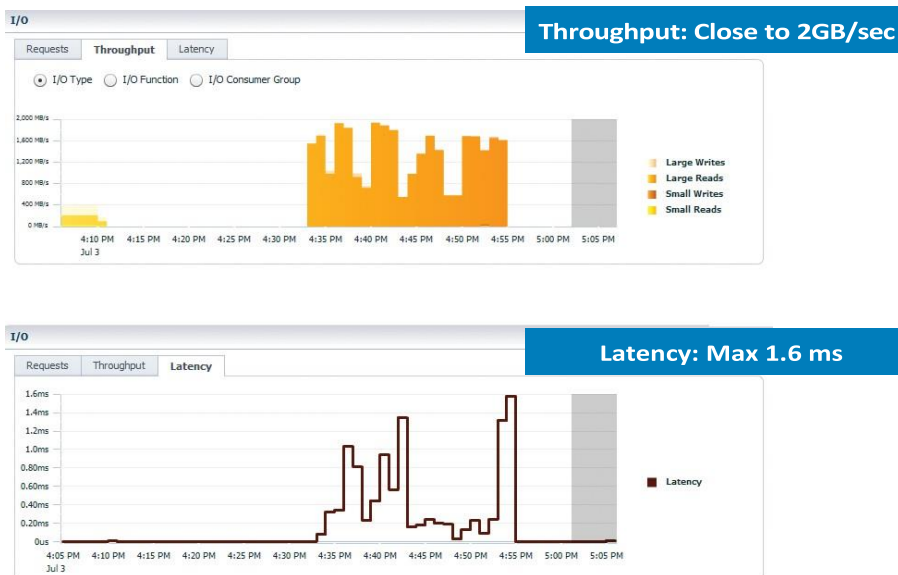**Figure 5: HDD Throughput and Latency Report from Oracle Enterprise   Manager**

Figure 6: SanDisk SSD Throughput and Latency Report from Oracle Enterprise Manager

Oracle Parallelism for Data Warehouse

Oracle Database uses query parallelism to fetch and process large amounts of data with parallel threads, which is a substantial benefit for data warehouse workloads. Oracle accomplishes this parallelism by splitting the main task into parallel query threads, with each of them executing a subset of the main task. This test also highlighted how Oracle Parallelism improved Oracle Data Warehouse performance using the SanDisk Optimus SAS SSDs.

Figure 7 shows the impact of the degree of parallelism (DOP) on total query elapsed time for the 22 data warehouse queries. The tests varied DOP from number 2 to 32, and observed that SSD performance scaled from 2 to 20. We did not see any benefit beyond a DOP value of 20, because the test server had only 16 cores. The key observation from our testing is that SanDisk SSDs consistently provide a 50% performance benefit over HDDs for all of the data warehouse parallel workloads.

Data Warehouse Parallel Query Performance
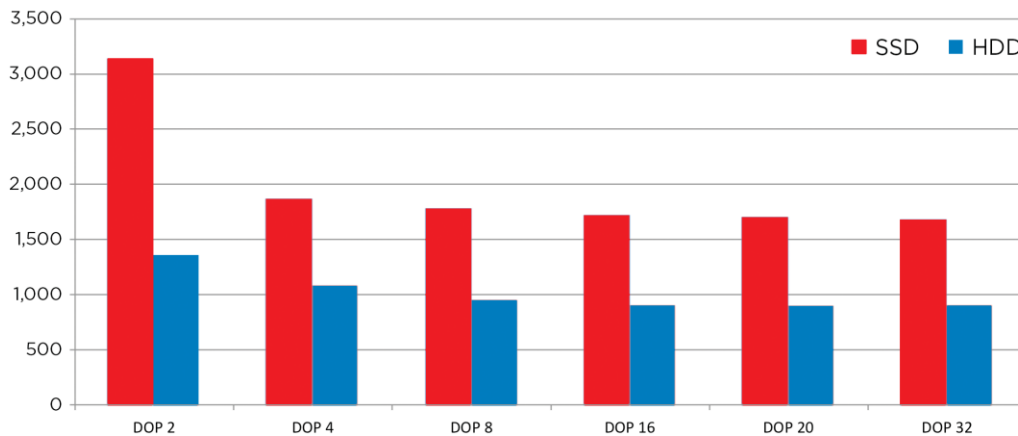
Figure 7: Parallel Execution: SSD VS HDD Elapsed Time

Database Scale Factor

The testing also compared HDD to SSD performance for different database sizes—100GB, 300GB and 1000GB. Figure 8 below provides the execution timings of SSDs and HDDs for the above database sizes. It is evident from Figure 8 that SanDisk SSDs provide a consistent performance benefit for a range of database sizes—from smaller databases to large-size databases.

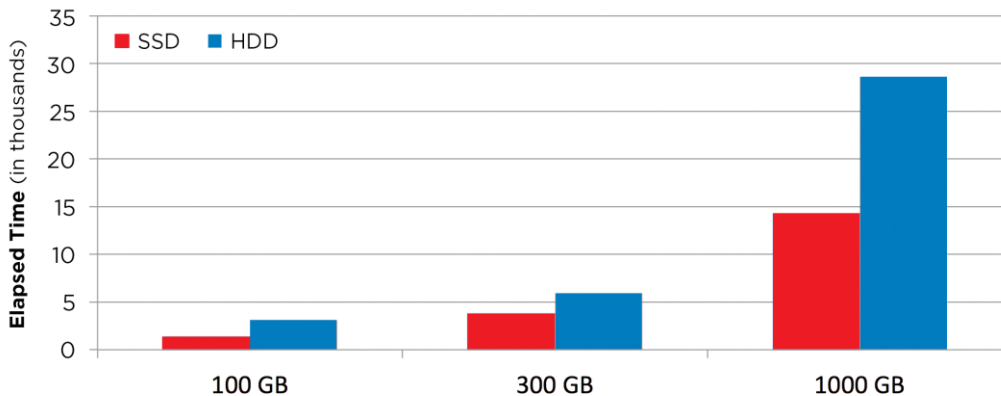**Data Warehouse Queries Performance**

Figure 8: SSD VS HDD Elapsed Time for DB Scale factor of 100/300/1000GB

Conclusions

The performance studies and analysis highlighted within this white paper show that SanDisk Optimus SAS SSDs provide consistent performance and scalability benefits for complex data warehouse workloads.

Based on these tests, SanDisk Optimus SAS SSDs:

1. Delivered over 2X performance benefit over HDDs running complex data warehouse queries for small, medium and large databases.
2. Provide a 50% faster parallel query performance.
3. Show a consistent 50% performance benefit across database sizes from 100GB to 1TB.
4. Deliver results at 1 millisecond latency—or less—making them very well-suited for the most demanding, low-latency Oracle database workloads.

The amount of data that is stored in the data warehouses has increased exponentially in recent years. Further, data warehouse support for Big Data and Analytics has led to large, multi-terabyte (TB) data-stores being housed in enterprise data centers—created another pressing demand for fast storage systems with low latency. However, the size of the database should not, by itself, introduce performance bottlenecks into database processing. That is why the strategic use of SSDs in large data warehouse workloads will address the performance challenges that are caused by the rapid growth of Big Data in the enterprise. Given the growth of Analytics, often leveraged in support of Big Data and data warehouses to find the "patterns in the data," solid-state drives (SSDs) will have a strong role in accelerating the performance of data warehouses.

About the Author

Prasad Venkatachar is a Technical Marketing Manager at SanDisk. He has more than 10 years of experience in architecting, design and implementations of Oracle database solutions for various customers in retail, banking, and insurance industries. He is certified on Oracle database releases, ranging from Oracle Database 8X to Oracle Database 11X, and is also certified on the IBM DB2 relational database, ITIL V3 and VMware virtualization products. His other areas of expertise include his work on SAP HANA and NoSQL databases.

References

SanDisk website: www.sandisk.com

Oracle Automatic Workload and Data Warehouse Concepts: oracle.com

Red Hat Optimization for Oracle : redhat.com

HammerDB Testing Tool documentation: hammerora.sourceforge.net/document.html

Western Digital Technologies, Inc. is the seller of record and licensee in the Americas of SanDisk® products.