WHITE PAPER

Oracle NoSQL Database Cluster YCSB Testing with Fusion ioMemory™ Storage

ORACLE Gold Partner

SanDisk a Western Digital brand
Western Digital Technologies, Inc.
951 SanDisk Drive, Milpitas, CA 95035

Table of Contents

Executive Summary....3
- Fusion ioMemory PCIe Application Accelerators 3
- Oracle NoSQL Database 3

Test Configuration Details 4

Oracle NoSQL Database Configuration Setup 5
- Oracle NoSQL Database Shard Planning 5
- Oracle NoSQL Database Two-Shard Deployment 5
- Oracle NoSQL Database Cluster Setup 6
  - Bootstrapping Storage Nodes to Host the NoSQL Instance 6
  - Configuring Key-Value Stores 7
  - NoSQL Cluster Verification 8
- Oracle NoSQL Database Admin Console Report 8

YCSB Test Configuration Setup 10

YCSB Test Results and Analysis 12
  - Data Ingestion and Deletion 12
  - Workload A: 50% Read / 50% Write 13
  - Workload B: 95% Read / 5% Write 15
  - Workload C: 100% Read / 0% Write 17

Scalability Tests 19

Conclusion 20

Executive Summary

Highly distributed systems with large data stores in the form of NoSQL databases are becoming increasingly important to enterprises, not just to hyperscale organizations. NoSQL databases are being deployed for capturing patient sensors data in health care, smart meter analysis in utilities, customer sentiment analysis in retail, and various other use cases in different industries. NoSQL database systems help organizations store, manage, and analyze huge amounts of data on distributed system architecture. The sheer volume of data and the distributed system design needed to manage this large data at a reasonable cost necessitated a different category of database systems, leading to NoSQL databases. Oracle NoSQL Database is part of the NoSQL database family and is based on a distributed, key-value architecture.

This technical white paper describes a three-node Oracle NoSQL Database Cluster deployment procedure on Fusion ioMemory™ storage. The following points are emphasized:

- Highlights performance and scalability advantages compared to traditional spinning disks.
- Because enterprises evaluate and assess new technologies for enterprise-wide adaptability, Yahoo Cloud Serving Benchmark (YCSB) is the standard benchmark tool employed for testing and is the same tool used in this paper to evaluate Oracle NoSQL Database for YCSB Benchmark Testing.
- Analysis and discussion are provided for throughput and latency testing results with YCSB.

Fusion ioMemory PCIe Application Accelerators

Performance tests were conducted using a single Fusion ioMemory PCIe device with 1.6 TB capacity per server, on a three-node Oracle NoSQL Database cluster setup. The latest Fusion ioMemory SX350 PCIe series is the latest generation of the world-renowned Fusion ioMemory platform. It has an intelligent controller that supports all major operating systems, including Microsoft Windows, UNIX, Linux, VMware ESXi, and Microsoft Hyper-V. The Fusion ioMemory SX350 PCIe series is available in sizes ranging from 1.25 to 6.4 TB of addressable, persistent flash storage. It provides a random read/write performance up to 345K/585K IOPS, with 79-microsecond read latency and 15-microsecond write latency.

Additional details on Fusion ioMemory devices can be found in this data sheet:

https://www.sandisk.com/content/dam/sandisk-main/en_us/assets/resources/enterprise/data-sheets/fusion-iomemory-sx350-pcie-application-accelerators-datasheet.pdf

Oracle NoSQL Database

Oracle NoSQL Database uses a simple data model, with a paradigm of key-value pairs of major and minor keys. Each key-value pair is hashed into multiple buckets, called partitions. These partitions are mapped to a replication group, similar to logical grouping for the data subset. A set of replication nodes provides both high availability and read scalability for each replication group. A storage node runs these replication nodes, which have one master and two replica nodes on each storage node. Each application or client is intelligently connected to the correct replica node by the NoSQL database driver. Oracle NoSQL Database also provides the flexibility to match to the customer requirements, emphasizing consistency and durability.

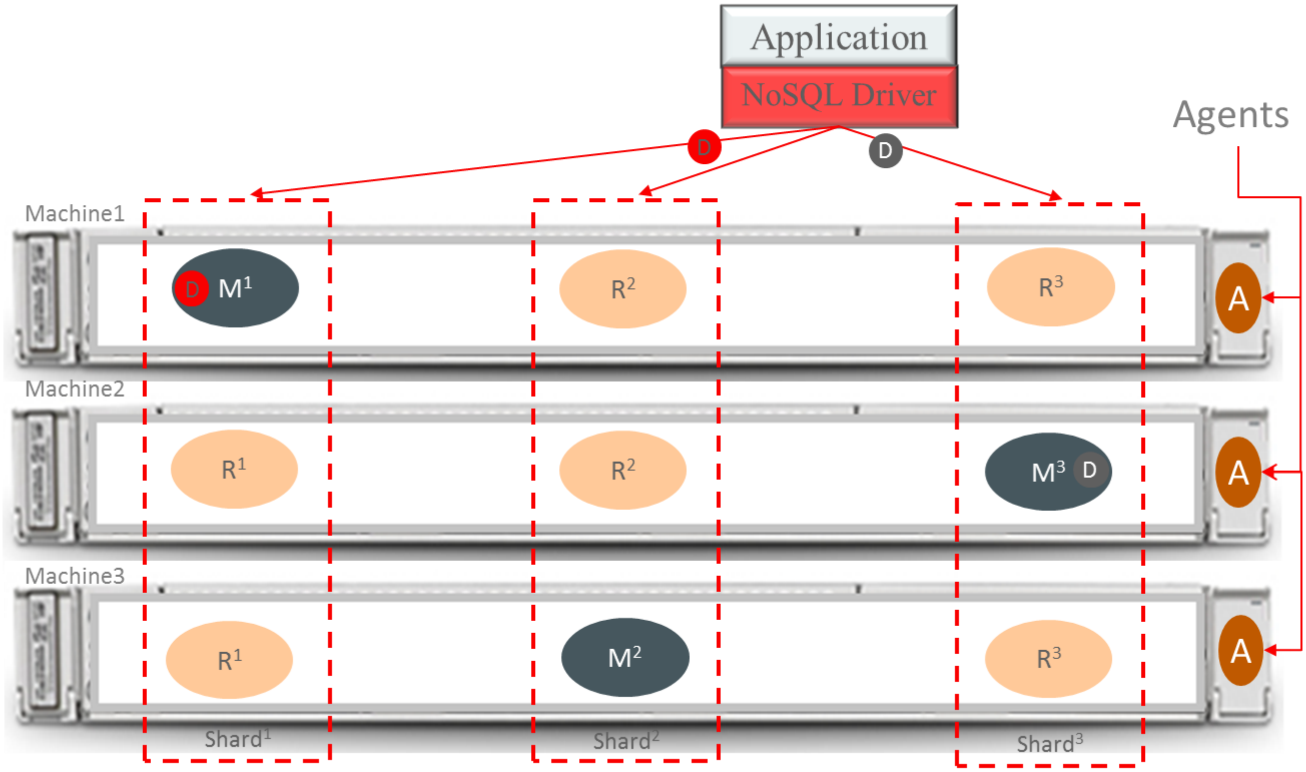

Figure 1: Oracle NoSQL Database architecture (D=Data, M=Master Node, R=Replication Node)

Test Configuration Details

The benchmark test environment consists of a three-node Oracle cluster setup on three Lenovo System x3650 servers. Each of these servers has 16 Intel Xeon E5-2690 cores with 128 GB RAM. A 10 GbE network interconnect is used for intra-node communication in the Oracle NoSQL Database cluster and YCSB client.

Storage for the large Oracle NoSQL Database data store is initially configured with traditional spinning disks and later switched to Fusion ioMemory. An XFS file system is created for hosting the Oracle NoSQL Database data store.

The table below provides the complete hardware and software components used for this test environment.

Component	Hardware	Software if applicable	Purpose	Quantity
Server	Lenovo System x3650, 2-Socket, 16 Cores, Intel Xeon CPU E5-2690 V2 @2.9 GHz, 128 GB memory	Red Hat Linux - 7.1 Oracle NoSQL Database 3.2.5	Server	3
Client	Lenovo System x3650, 2-Socket, 16 Cores, Intel Xeon CPU E5-2690 V2 @2.9 GHz, 64 GB memory	Red Hat Linux 7.1 - YCSB 0.2.0	Client	1
Network	Network	10 GbE network switch	Management Network	1
Traditional Storage	6 TB HDD (RAID 0)	RAID 0 - XFS File System	Storage Node	3
SanDisk® Flash – Fusion ioMemory devices	Fusion ioMemory with 1.6 TB storage	XFS File System	Storage Node	3

Table 1: Test configuration details

Oracle NoSQL Database Configuration Setup

Oracle NoSQL Database Shard Planning

The following configuration settings will derive the total shards required for the 3 node Oracle NoSQL Database cluster deployment.

- Total storage nodes available = 3
- Data distribution on each storage node = 2 x capacity
- Replication factor = 3
- Total shards derived = $\frac{\text{2 x capacity on each machine * 3 machines}}{\text{3 replication factor}} = \text{2 shards}$

Oracle NoSQL Database Two-Shard Deployment

Each replication node is organized into shards with a replication factor of 3, as shown in the figure below.

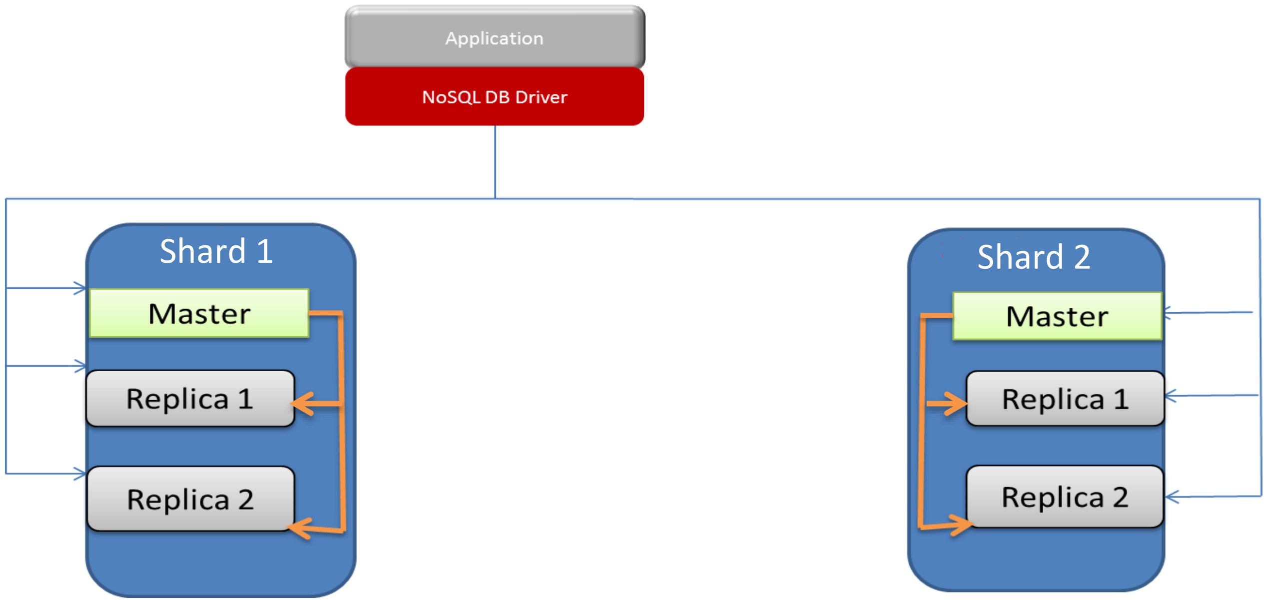

Figure 2: Two-shard cluster topology deployed with replication factor = 3

Oracle NoSQL Database Cluster Setup

1. Download the Oracle NoSQL Database software from the Oracle website.

   http://www.oracle.com/technetwork/database/database-technologies/nosqldb/downloads/index.html

2. Extract the Oracle NoSQL Database binaries to a location and define it as the KVHOME directory.
3. Create KVROOT and Oracle NoSQL Database data directories for storing configuration files and key value data for YCSB testing.
4. Follow the bootstrapping, key-value configuration, and cluster verification procedures shown below.
5. Deploy the NoSQL cluster.

Bootstrapping Storage Nodes to Host the NoSQL Instance

Before hosting Oracle NoSQL Database instances, you must bootstrap the services. As discussed previously, capacity was defined as 2, which stores the Oracle NoSQL Database key values in two separate storage directories (data1 and data2) on the storage node. A Storage Node Agent (SNA) process runs on each storage node to facilitate communications between other SNA processes and the client applications. The SNA listens to the registry port specified with the `-port` parameter, as well as `-admin` and `-harange` ports for administration and replication processes.

1. Run the following commands on each server in the cluster:

```
# java -Xmx256m -Xms256m -jar $KVHOME/lib/kvstore.jar  \
makebootconfig -root /sandisk/data/nosqldb/data/ \
 -port 5000 \
-admin 5001\
```

```
-host <server1_ip>\
-harange 5010, 5020 \
-store-security none \
-capacity 2 \
-storagedir /sandisk/data/nosqldb/data/data1 \
-storagedir /sandisk/data/nosqldb/data/data2/
```

2. Start `kvstore` on each server as a background process:

```
# nohup java -jar $KVHOME/lib/kvstore.jar start -root /sandisk/data/nosqldb/data/
&
```

Configuring Key-Value Stores

When all the storage nodes have been created and started, the storage nodes need to be configured to work as one distributed cluster of the key-value store. This key-value store is built based on the initial planning of replication nodes and number of shards participating in the cluster.

1. Name the KVSTORE as "kvstore".
2. Create and deploy the datacenter as "California" with a replication factor of 3.
3. Deploy a storage node and create a Storage Node Pool, such as "Mypool"
4. Join "Mypool" to the storage node.
5. Repeat the above steps on each server.
6. Join storage nodes from other nodes with "Mypool" created on the first server.
7. Create a topology with all storage nodes connected to Mypool and with 100 partitions.

The following command accomplishes these steps:

```
# java -jar $KVHOME/kvstore.jar runadmin –port 5000 –host <IP_Address_Node1>
```

These commands provide connectivity to the storage node:

```
kv> configure -name "kvstore"
kv> plan deploy-datacenter –name "California" -name MyZone -rf 3 -wait
kv> plan deploy-sn -znname MyZone -host <IP_Address_Node1> -port 5000 -wait
kv> plan deploy-admin –sn sn1 -port 5001 -wait
kv> pool create -name MyPool
kv> pool join -name MyPool -sn sn1
kv> plan deploy-sn -znname MyZone -host <IP_Address_Node2> -port 5000 -wait
kv> pool join -name MyPool -sn sn2
kv> plan deploy-sn -znname MyZone -host <IP_Address_Node3> -port 5000 -wait
kv> pool join -name MyPool -sn sn3
kv> topology create -name MyStoreLayout -pool MyPool -partitions 100
kv> topology preview -name MyStoreLayout
kv> plan deploy-topology -name MyStoreLayout -wait
kv> show plans
```

```
kv> show topology
kv> verify
```

NoSQL Cluster Verification

Oracle NoSQL Database provides both a command line and an admin console for cluster deployment verification and management. The following commands provide cluster verification for both options.

```
# java  -jar $KVHOME /lib/kvstore.jar runadmin -port 5000 -host server1
kv -> verify
```

```
kv-> verify
The command:

        verify [-silent]

is deprecated and has been replaced by:

        verify configuration [-silent]

Verify: starting verification of kvstore based upon topology sequence #112
100 partitions and 3 storage nodes. Version: 12.1.3.2.5 Time: 2015-05-22 09:16:20 UTC
See 192.168.0.11:/sandisk/data/nosqldb/data/kvstore/log/kvstore_{0..N}.log for progress messages
Verify: == checking storage node sn1 ==
Verify: Storage Node [sn1] on 192.168.0.11:5000    Zone: [name=MyZone id=zn1 type=PRIMARY]    Status: RUNNING   Ver: 12cR1.3.2.5 2014-12-05 01:49:22
 UTC  Build id: 7ab4544136f5
Verify:         Admin [admin1]          Status: RUNNING
Verify:         Rep Node [rg2-rn1]      Status: RUNNING,MASTER at sequence number: 267,763,483 haPort: 5012
Verify:         Rep Node [rg1-rn1]      Status: RUNNING,REPLICA at sequence number: 267,853,507 haPort: 5011
Verify: == checking storage node sn2 ==
Verify: Storage Node [sn2] on 192.168.0.13:5000    Zone: [name=MyZone id=zn1 type=PRIMARY]    Status: RUNNING   Ver: 12cR1.3.2.5 2014-12-05 01:49:22
 UTC  Build id: 7ab4544136f5
Verify:         Rep Node [rg2-rn2]      Status: RUNNING,REPLICA at sequence number: 267,763,483 haPort: 5011
Verify:         Rep Node [rg1-rn2]      Status: RUNNING,MASTER at sequence number: 267,853,507 haPort: 5010
Verify: == checking storage node sn3 ==
Verify: Storage Node [sn3] on 192.168.0.14:5000    Zone: [name=MyZone id=zn1 type=PRIMARY]    Status: RUNNING   Ver: 12cR1.3.2.5 2014-12-05 01:49:22
 UTC  Build id: 7ab4544136f5
Verify:         Rep Node [rg2-rn3]      Status: RUNNING,REPLICA at sequence number: 267,763,483 haPort: 5011
Verify:         Rep Node [rg1-rn3]      Status: RUNNING,REPLICA at sequence number: 267,853,507 haPort: 5010

Verification complete, no violations.
```

Figure 3: Two-shard cluster topology deployed with replication factor = 3, on three storage nodes

Oracle NoSQL Database Admin Console Report

Oracle NoSQL Database Admin Console is a web-based interface that can be accessed from the admin node and default port 5501. It is accessed at http://<adminnode: port>

Figure 4 shows the Oracle Database Admin Console and it is useful for

- Becoming familiar with NoSQL database administration
- Checking the Event Monitor and Performance Logs
- Starting and stopping replication nodes
- Viewing and validating the topology

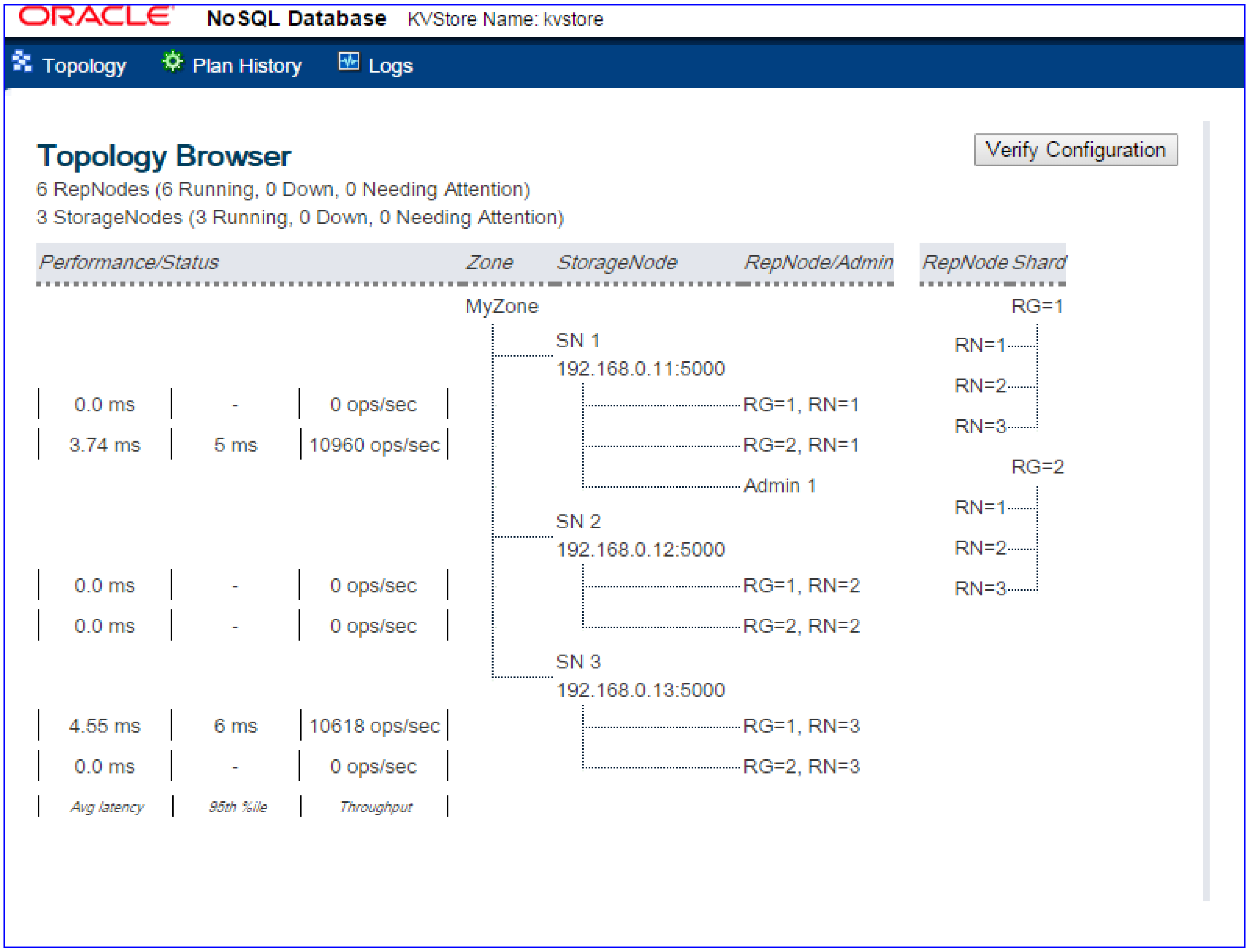

Figure 4: Oracle NoSQL Database Admin Console Topology Browser results

Figure shows the topology verification output completed with no cluster deployment violations. It also shows the Oracle NoSQL cluster topology deployed on 3 storage nodes with 100 partitions.

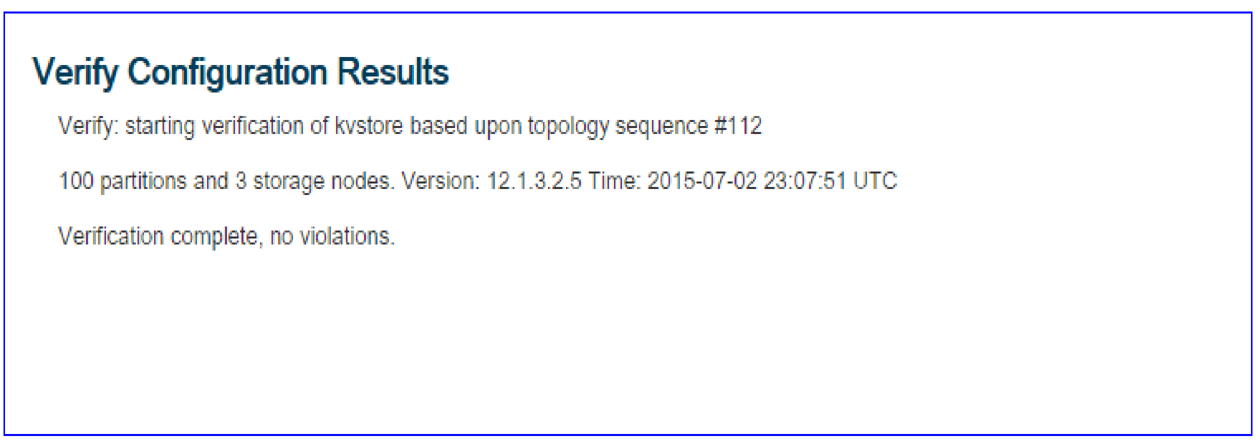

Figure 5: Verify configuration results

The figure below shows the two shards deployed with a configuration of three storage nodes.

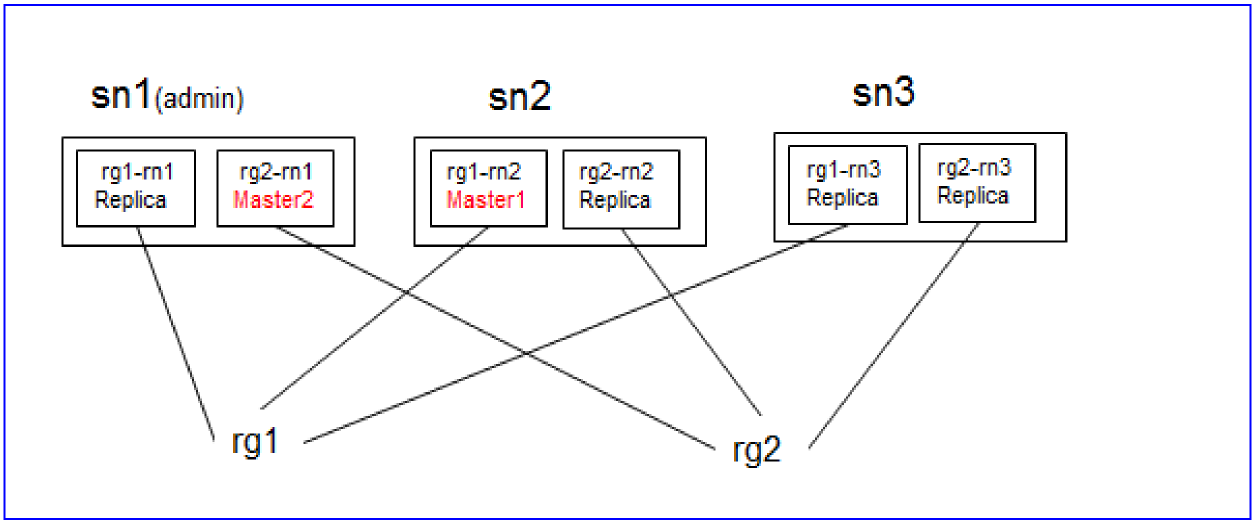

Figure 6: Two-shard cluster topology deployed with replication factor = 3, on three storage nodes

YCSB Test Configuration Setup

The figure below shows the YCSB configuration setup for the three-node Oracle NoSQL Database cluster benchmark. YCSB consists of a workload parameter file, which defines the type of workload to be executed, a read-write percentage and a dataset size to be used. YCSB also provides command-line parameters that can be passed during execution, such as the type of NoSQL to be used, YCSB client threads, etc.

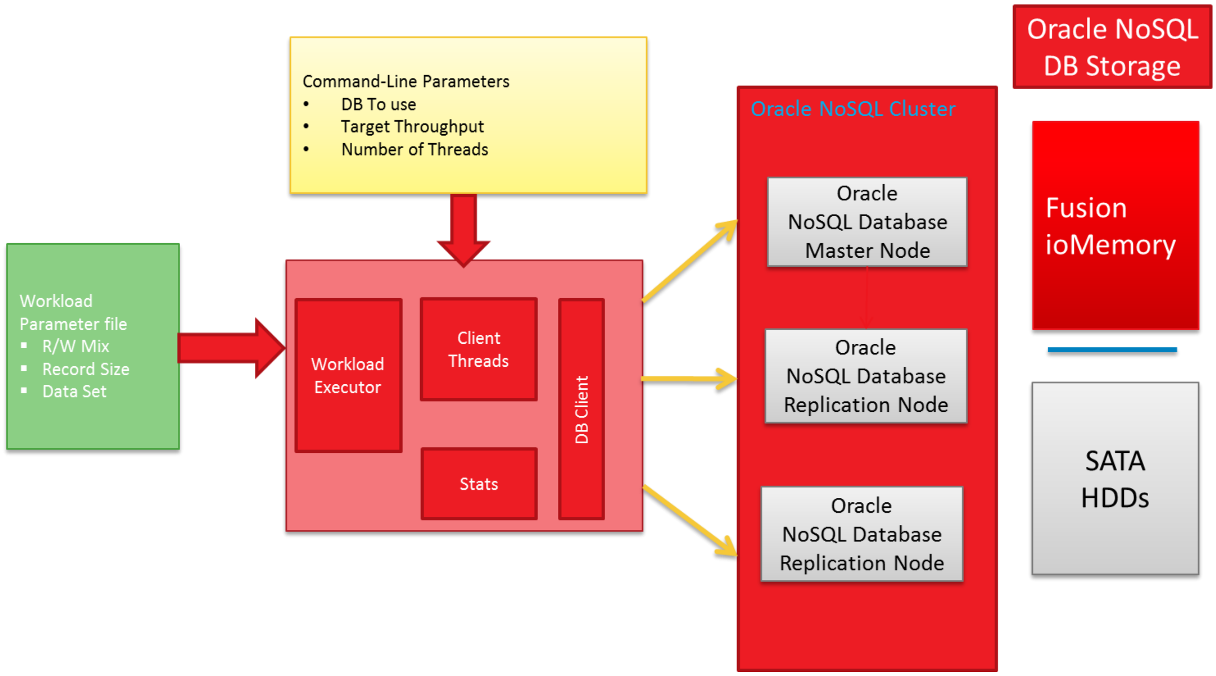

Figure 7: YCSB test configuration setup for Oracle NoSQL Database Cluster benchmark with Fusion ioMemory devices

The YCSB benchmark loads the dataset into the NoSQL cluster first and then executes the workloads. The following workload types have been used in testing:

- Workload A: Write-heavy, 50% write / 50% read
- Workload B: Read-heavy, 5% write / 95% read
- Workload C: Read-only, 100% read

Each of the above workloads can be executed using the following distribution types:

- Uniform: All database records are uniformly accessed.
- Zipfian: A few records in the database are accessed more often than other records.

The YCSB default data size is 1 KB records (1 field, 1000 bytes each, plus key).

The testing configuration loads the Oracle NoSQL Database cluster with dataset sizes of 32 GB, 128 GB, 256 GB, and 1 TB. Then workloads A, B, and C are executed for both Uniform and Zipfian distribution. To evaluate Fusion ioMemory benefits for Oracle NoSQL Database – including small to large datasets, heavy writes to heavy reads, and uniform workload distribution to Zipfian distribution – its performance was compared with that of standard spinning disks.

All workload combinations resulted in 24 different tests for this benchmark. This performance evaluation helps organizations provide sufficient data points to meet their performance evaluation criteria, based on application workload demands.

YCSB Test Results and Analysis

This section analyzes the YCSB benchmarking with an Oracle NoSQL Database cluster using Fusion ioMemory storage.

Data Ingestion and Deletion

1 Billion Records Operation Type	Storage Type	Ops/Sec	Latency (ms)	Total time (ms)
Insert	HDD	71,083	6	14,405
	Fusion ioMemory Devices	84,004	5	9,650
Delete	HDD	85,603	9	11,962
	Fusion ioMemory Devices	104,942	8	9,757

Table 2: Data loading and latency results summary for a 1 TB dataset

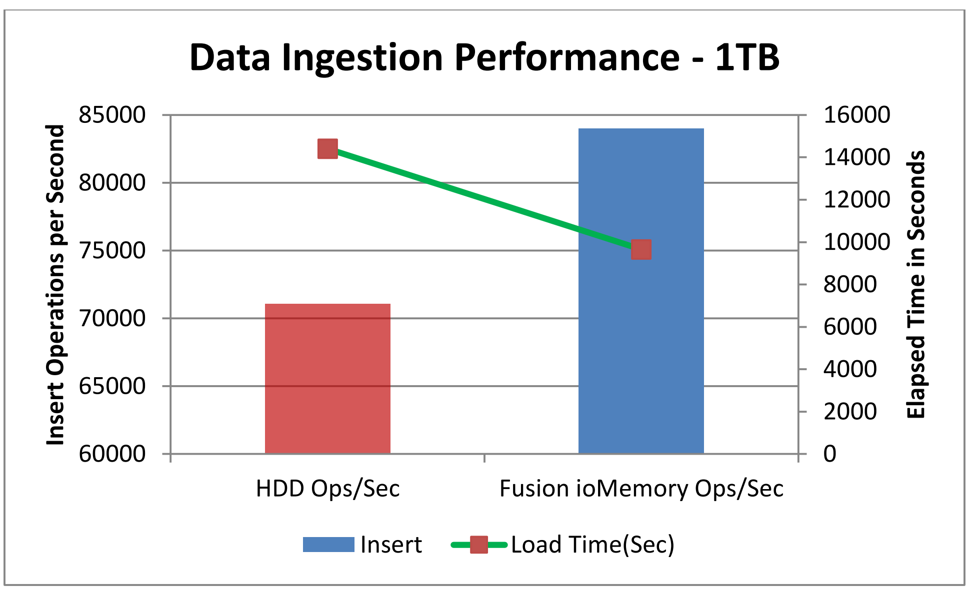

Chart 1: Data loading and latency results summary for a 1 TB dataset

The first benchmarking step was to load the dataset to an Oracle NoSQL Database cluster, which was done using the YCSB load routine for all full datasets. The largest dataset loaded to the Oracle NoSQL Database cluster for both hard disk and Fusion ioMemory devices is 1 TB, and its test results are captured for analysis in Table 1 and Chart 1.

As shown in the chart above, a 1 TB dataset of 1 billion records on spinning disk is ingested in 14,405 seconds, with 71,083 Ops/sec and 6 ms latency. For the same 1 TB dataset on Fusion ioMemory devices, data ingestion completes in 9,650 seconds. This is 33% faster, with 84,004 Ops/sec at 5 ms latency. As shown in Table 4, delete operations are completed 18% faster. This result is important for key-value data stores, which capture session data for web applications and data from devices such as patient sensors and load the data into the Oracle NoSQL Database datastore.

Workload A: 50% Read / 50% Write

This test evaluates the performance benefits of Fusion ioMemory devices for Oracle NoSQL Database cluster, with balanced workloads. Throughput test results for Fusion ioMemory vs. HDDs are shown in Table 3 and plotted in Chart 2, and latency results are shown in Table 4 and Chart 3. The 32 GB dataset test involves in-memory operations and hence both HDD and Fusion ioMemory results are in the same range. Fusion ioMemory provides only a marginal 1.4x performance benefit over hard disk drives for the 32 GB test case.

Data (Keys) required for this mixed workload operation are served from the server memory with minimal I/O operations from the disk.

Storage	Workload Distribution Type	Throughput Ops/Sec			
Type		32 GB	256 GB	512 GB	1 TB
HDD	Uniform	90,600	13,073	10,468	8,009
HDD	Zipfian	80,798	26,359	20,765	16,598
Fusion ioMemory	Uniform	127,125	96,045	94,470	88,118
Fusion ioMemory	Zipfian	90,341	92,829	88,376	84,760

Table 3: Throughput results summary for Workload A

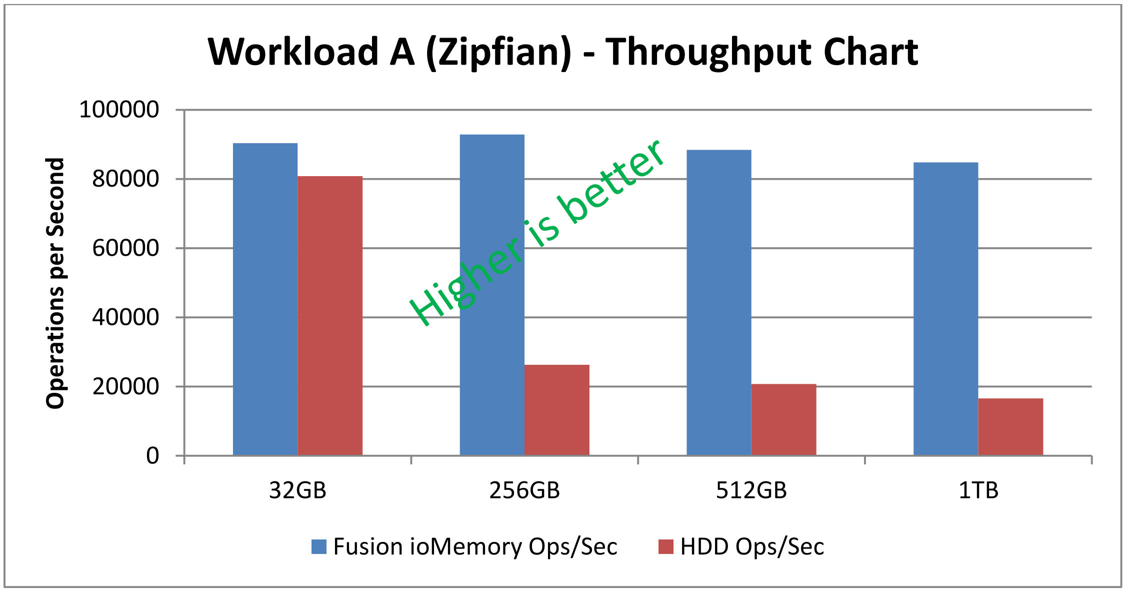

Chart 2: Workload A (Zipfian) throughput test results

As the dataset increases from 32 GB to 256 GB, the 512 GB to 1 TB Fusion ioMemory storage provides a significant performance benefit, with a minimal drop in throughput and a marginal increase in latency.

Storage Type	Workload Distribution Type	Latency in Milliseconds			
		32 GB	256 GB	512 GB	1 TB
HDD	Uniform	9	229	243	288
HDD	Zipfian	6	159	186	194
Fusion ioMemory	Uniform	9	14	14	14
Fusion ioMemory	Zipfian	9	8	8	10

Table 4: Latency results summary for Workload A

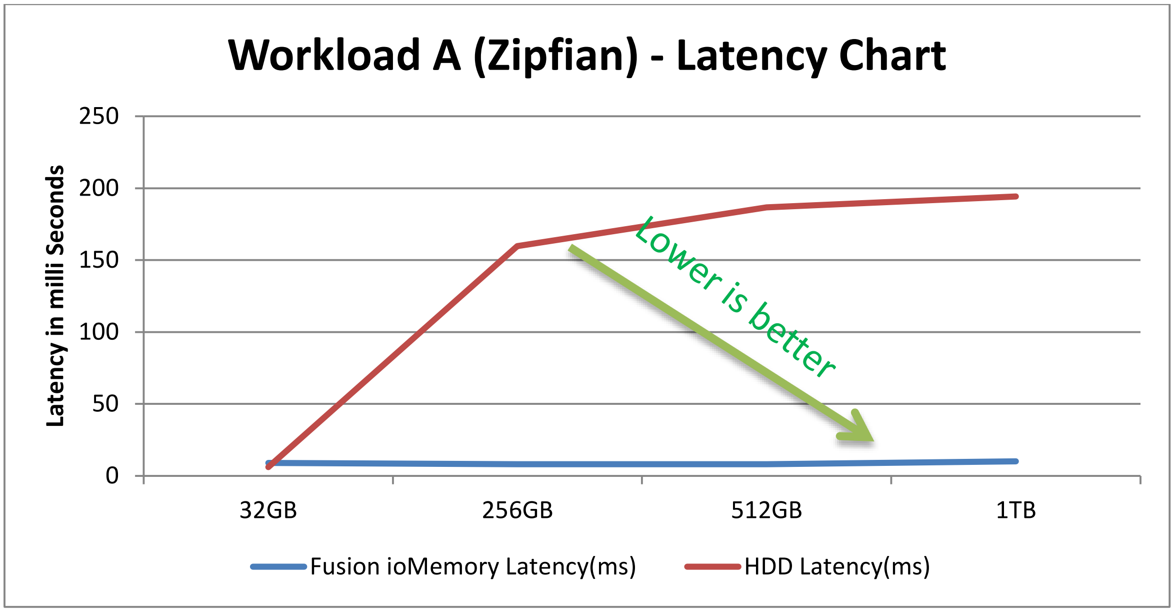

Chart 3: Workload A (Zipfian) latency test results

Fusion ioMemory storage provides up to a 3x to 11x throughput benefit, with a 16x to 23x latency reduction compared to HDD operations. For larger datasets, HDD performance drops considerably: a 32 GB dataset on HDD provided 80,798 ops/sec with 6 ms latency, but a 1 TB dataset provided only 16,598 IOPS with 194 ms latency. That is nearly a 5x drop in performance and 32x increase in latency. The performance advantage of Fusion ioMemory storage is useful for web applications using Oracle NoSQL Database, such as in immediate capture of user session actions.

Workload B: 95% Read / 5% Write

Workload B is a read-intensive mixed workload with 95% read and 5% write operations. Throughput test results for Fusion ioMemory vs. HDDs are shown in Table 5 and plotted in Chart 4, and latency results are shown in Table 6 and Chart 5. In this workload, Fusion ioMemory storage provides up to a 19x to 48x improvement in throughput operations and a 16 to 45x reduction in latency, compared to hard disks, for datasets ranging from 256 GB to 1 TB. (For a 32 GB dataset completely residing in memory, the throughput and latency metrics remain the same for both Fusion ioMemory and HDD storage.) This improved performance is beneficial for applications such as photo tagging, where adding a tag is a write operation, but all the other operations are reads.

Storage types	Workload Distribution Type	Throughput Ops/Sec			
		32 GB	256 GB	512 GB	1 TB
HDD	Uniform	313,733	6,714	4,895	4,695
HDD	Zipfian	304,496	7,466	5,656	9,090
Fusion ioMemory	Uniform	318,600	131,234	140,801	176,354
Fusion ioMemory	Zipfian	312,345	255,217	275,292	283,243

Table 5: Throughput results summary for Workload B

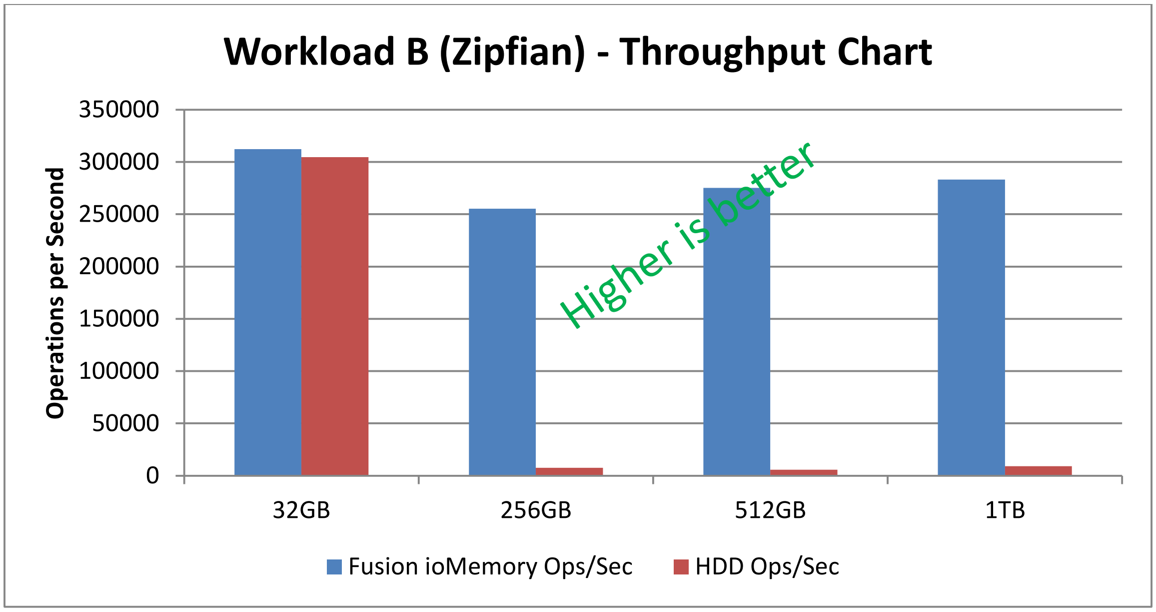

Chart 4: Workload B (Zipfian) throughput test results

Storage Type	Workload Distribution Type	Latency in Milliseconds			
		32 GB	256 GB	512 GB	1 TB
HDD	Uniform	9	238	229	243
HDD	Zipfian	6	165	159	186
Fusion ioMemory	Uniform	9	10	14	14
Fusion ioMemory	Zipfian	9	9	8	8

Table 6: Latency results summary for Workload B

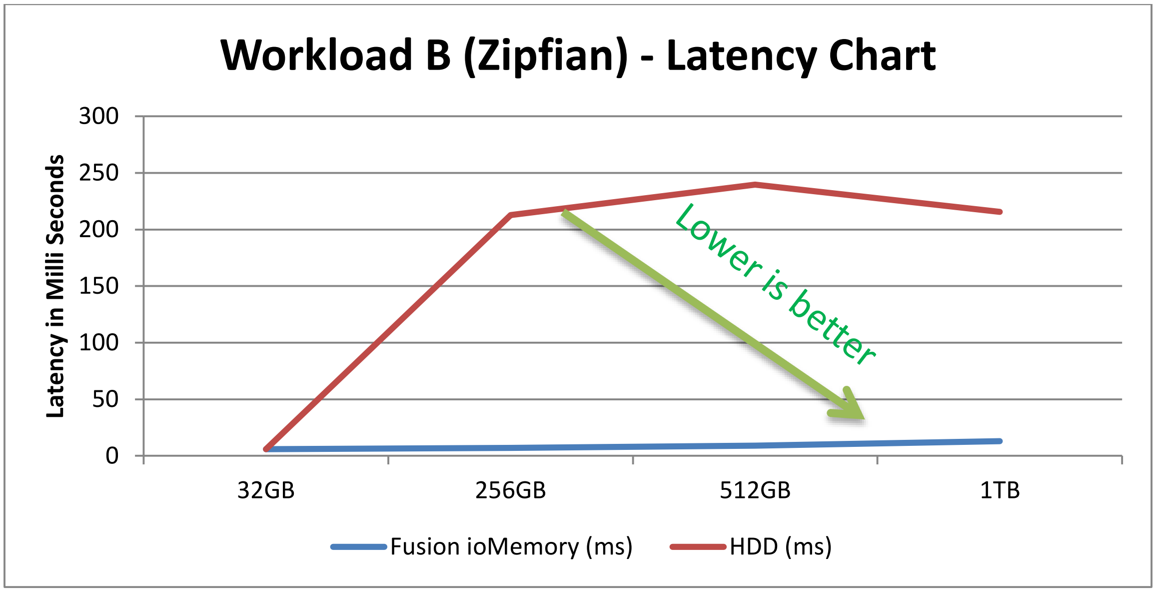

Chart 5: Workload B (Zipfian) latency test results

Workload C: 100% Read / 0% Write

Workload C is a read-only workload with 100% read operations. Generally this kind of workload is used for caching user profiles in web applications. Table 7 and Chart 6 provide the throughput performance numbers, and Table 8 and Chart 7 capture the latency metrics. For larger datasets from 256 GB to 1 TB, Fusion ioMemory devices provided a 17x to 48x performance advantage, with latency improvements from 30x to 100x, compared to the same operations on HDDs.

Storage	Workload types	Throughput Ops/Sec			
types		32 GB	256 GB	512 GB	1 TB
HDD	Uniform	377,187	7,291	5,360	4,397
HDD	Zipfian	378,075	14,723	10,080	8,286
Fusion ioMemory	Uniform	378,889	131,948	157,830	215,180
Fusion ioMemory	Zipfian	379,408	251,450	295,594	357,685

Table 7: Throughput results summary for Workload C

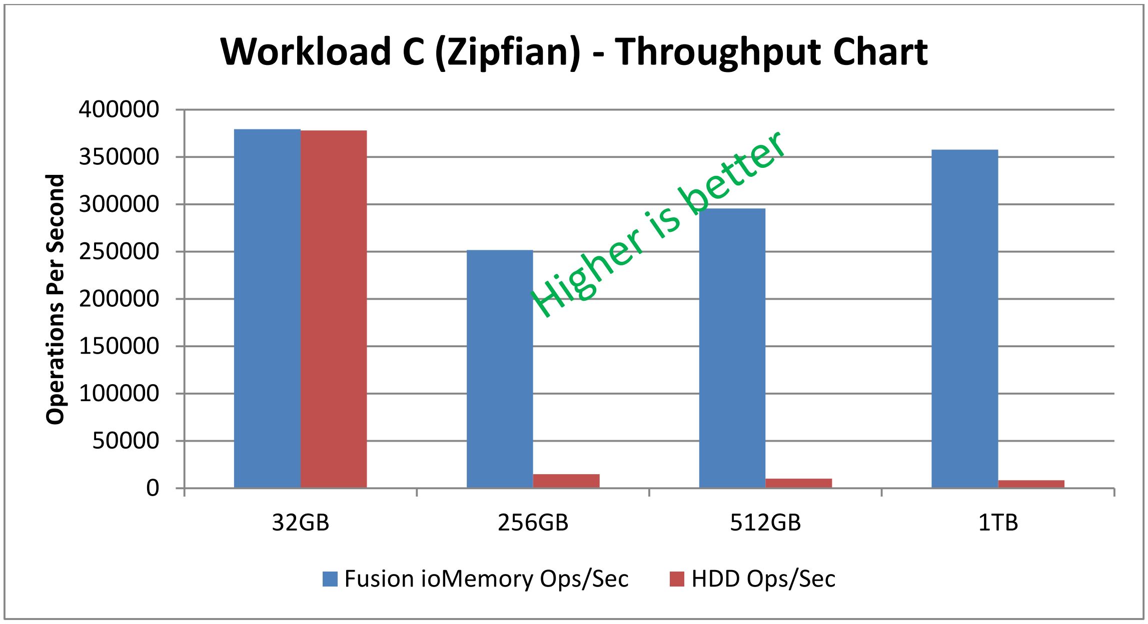

Chart 6: Workload B (Zipfian) Latency test results

Storage Type	Workload types	Latency in Milliseconds			
		32 GB	256 GB	512 GB	1 TB
HDD	Uniform	1	216	229	243
HDD	Zipfian	1	190	159	186
Fusion ioMemory	Uniform	1	7	7	5
Fusion ioMemory	Zipfian	1	6	6	4

Table 8: Latency results summary of Workload C

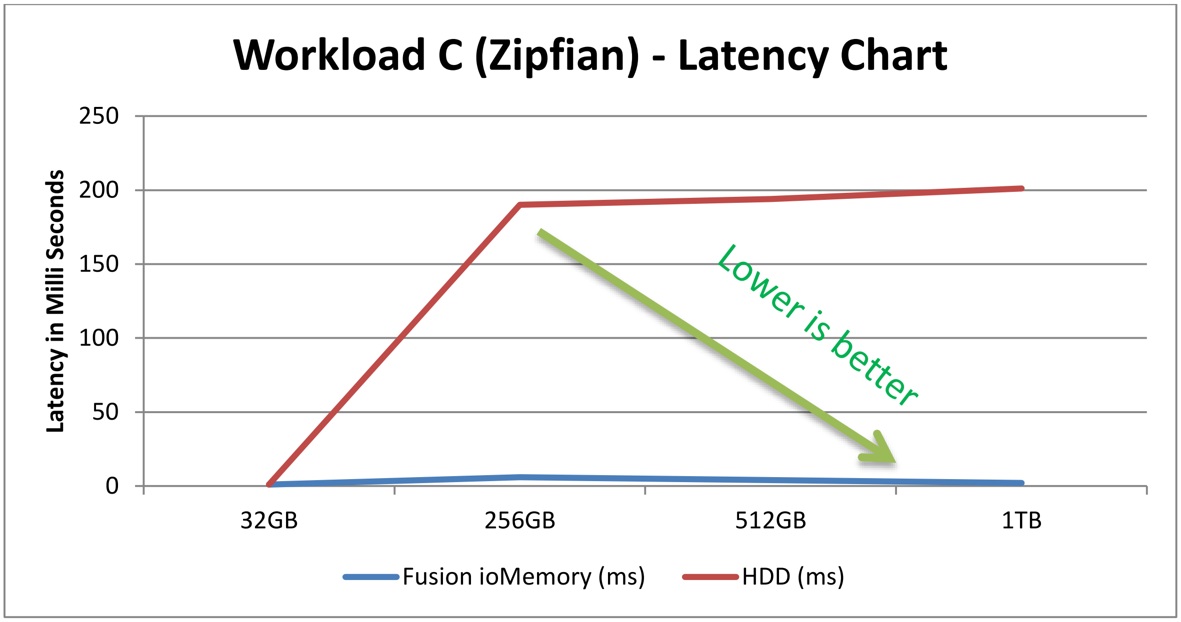

Chart 7: Workload B (Zipfian) latency test results

Scalability Tests

The maximum performance limit of this 3-node Oracle NoSQL Database cluster setup was tested with Fusion ioMemory storage by increasing from 2 shards to 3 shards. This configuration resulted in each server having one shard with one master, and two replicas of other shard nodes, as shown in the figure below.

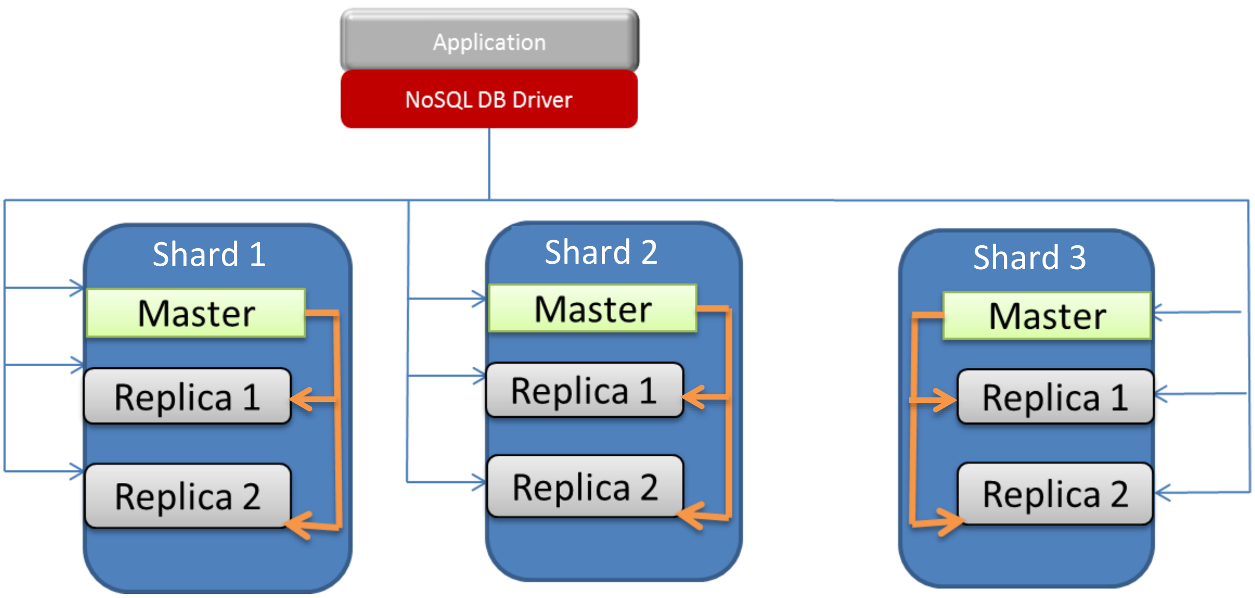

Figure 7: Oracle NoSQL Database cluster with 3 shards and replication factor = 3, on 3 storage nodes

Shards	KVS Size (Record Count)	YCSB Threads	Throughput (ops/sec)	Read Latency ms (95%)	Write Latency ms (95%)
2	252m(2x3)	90(360)	131234	8	0
3	252m(3x3)	90(360)	234636	4	1
2	512m(2x3)	90(360)	140801	6	0
3	512m(3x3)	90(360)	233716	3	1
2	1000m(2x3)	90(360)	176354	3	1
3	1000m(3x3)	90(360)	256688	2	2

Table 9: Oracle NoSQL Database, 2- and 3-shard scalability results for Workload B

With this setup a mixed workload was run; Workload B (95% read / 5% write) results were shown previously. As seen in Chart 8 below, 3 shards provide a 45% to 79% improvement in throughput IOPS and a 35% latency reduction for dataset sizes of 256 GB to 1 TB. Additional shards participating in the mixed workload increase the performance and reduce latency.

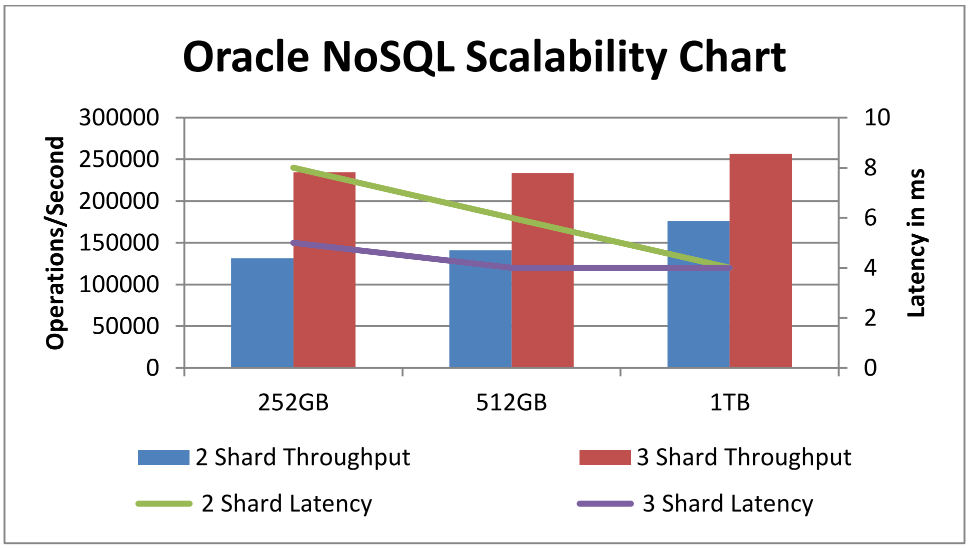

Chart 8: Oracle NoSQL Database, 2- and 3-shard scalability results

Conclusion

Enterprises are increasingly adopting NoSQL databases in various industry verticals such as healthcare (patient sensors), automotive (auto sensors), utilities (smart meter analysis), etc. The YCSB benchmark is well known and used across the industry for evaluating distributed NoSQL databases. Oracle NoSQL Database is an excellent choice; it's easy to deploy, with a simple key-value data model. It provides flexibility for consistency and durability, based on organization requirements. The performance analysis in this white paper demonstrates that Oracle NoSQL Database with Fusion ioMemory devices is a perfect combination to deliver both high performance and scalability benefits. For large dataset operations with mixed, write-intensive, or read-intensive workloads, Fusion ioMemory products deliver up to 11x to 48x improvement in performance. This also translates to fewer servers with higher performance storage for optimized NoSQL cluster deployment.

Acknowledgments

Special thanks to Ashok Joshi and Michael A. Schulman of Oracle for their expemplary support in reviewing and validating the solution during the design and development.

Prasad Venkatachar, Technical Marketing Manager, and Patric Chang, Senior Director, SanDisk were instrumental in the design, development and delivery of this solution.

Resources

Customers can find more information about SanDisk, Oracle NoSQL database, and Lenovo servers used at the links listed below:

SanDisk

SanDisk ESS Products: http://www.sandisk.com/enterprise/

Fusion ioMemory: https://www.sandisk.com/business/datacenter/products/flash-devices/pcie-flash/sx350

Oracle

Oracle NoSQL Database: http://www.oracle.com/us/products/database/nosql/overview/index.html

Oracle NoSQL Blogs: https://blogs.oracle.com/nosql/

Lenovo

System X3650 Server: https://lenovopress.com/tips0850-system-x3650-m4-e5-2600-v2