Technical Report

NetApp AFF A800 Performance with Oracle RAC Database

Joe Carter, Scott Lane, Jeffrey Steiner, Ling Wang - NetApp
March 2019 | TR-4767

Abstract

This report summarizes Oracle database performance with NetApp® AFF A800 storage systems for interested NetApp and partner engineers.

NetApp AFF systems combine the extreme performance capability of flash media with NetApp ONTAP® 9.5 software to provide performance acceleration, operational agility, industry-leading data protection, and business continuity for database deployments.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Introduction

NetApp solutions for Oracle databases are engineered for enterprise workloads with industry-leading performance, superior scalability, continuous data availability, and comprehensive data management. NetApp provides Oracle customers with the next generation of performance and manageability with the industry's first end-to-end, database-to-drive NVMe solution.

Built on ONTAP scale-out architecture, NetApp AFF SAN consistently meets or exceeds the high-performance demands of Oracle databases. Designed specifically for flash, AFF A-Series all-flash systems deliver industry-leading performance, capacity density, scalability, security, and network connectivity in dense form factors. The NetApp AFF A800, the industry's first end-to-end NVMe all-flash system, combines low-latency NVMe solid-state drives (SSDs) and NVMe over Fibre Channel (NVMe/FC) connectivity. It delivers less than 200µs in latency and a massive throughput of up to 204GBps in a 12-node SAN cluster.

NVMe/FC is the simplest, most broadly supported NVMe over Fabrics (NVMe-oF) option. It enables customers to use the same FC fabric for the NVMe protocol and their existing SAN infrastructure. NetApp customers can usually upgrade to NVMe/FC nondisruptively with a simple software upgrade, resulting in unprecedented performance without the need for changes to the network architecture. This upgrade allows customers to run 60% more workloads or to cut application response time by half.

The benefits of adopting a more efficient storage protocol include dramatically simplifying storage management by reducing the number of storage objects required to deliver peak performance. Reducing storage objects translates into reduced CPU demand, further improving database performance and potentially reducing Oracle license requirements. AFF also offers rich data management capabilities, such as integrated data protection, nondisruptive upgrades, and data migration. These features help eliminate performance silos and seamlessly integrate AFF into a shared infrastructure.

ONTAP software delivers enhanced inline deduplication, inline compression, and inline data compaction capabilities that significantly reduce the amount of flash storage required, with no effect on system performance. It also provides industry-leading ecosystem integration with database applications that makes administration of databases and storage systems far more efficient than with other flash storage solutions on the market.

2 Executive Summary

NetApp performed this study for Oracle applications to showcase the storage performance and the benefits of the AFF A800 with NVMe/FC compared to the Fibre Channel Protocol (FCP) for Oracle applications.

With a 100% SELECT workload, NVMe/FC reached a peak IOPS 70% higher than with FCP. Meanwhile, the latency of NVMe/FC at this load point was less than half that of FCP. Host CPU utilization was also reduced by up to 43% with a workload over NVMe/FC.

These results show that customers can run more Oracle workloads by upgrading to NVMe/FC host connectivity using existing hardware. They can even potentially reduce Oracle licensing costs because fewer host CPU cores might be needed.

3 Measuring Storage Performance

NetApp performed the following study to measure the performance of AFF A800 storage systems running ONTAP 9.5. This section describes the methodology and design considerations used to test the AFF storage systems running a standard Oracle workload.

3.1 Test Methodology

For this study, we used the SLOB2 load-generation tool to simulate an online transactional processing (OLTP) workload against the Oracle Database 12c test configuration. Two SLOB workload mixes were run, one with all SQL SELECTs and the other with a SELECT-to-UPDATE ratio of 75:25. In both configurations, a 1.5TB SLOB database was populated before running the SLOB workload.

An AFF A800 high-availability (HA) pair consists of two nodes. For this test, one node was dedicated to NVMe/FC and the other node was dedicated to FCP testing. An OLTP workload called SLOB2 was run individually on each node of an AFF A800 to compare the NVMe/FC and FCP protocols. Although we ran each test on single-storage-node configurations, all of the test results described in this report represent dual-storage-node performance from an active-active AFF A800 HA pair system. Our internal testing confirms linear performance scalability between single-storage-node and dual-storage-node configurations.

We created a 10-node Oracle Real Application Clusters (RAC) environment using SLES 12.3 hosts with a database connected through Fibre Channel to the AFF A800. We conducted the NVMe/FC and FCP testing at different times. However, each test used the same 10 Linux hosts, the same Brocade FC switch, and the same SLOB database size (1.5TB). The SLOB workload driver tool made requests to the Oracle database cluster, which in turn drove I/O to the AFF A800. We controlled the number of virtual users in SLOB to increase the workload intensity. We ran each load point of the workload for 20 minutes with a fixed number of users. We then increased the number of users and ran for another 20 minutes. We used between 3 and 450 SLOB users for each test suite to create about 15 load points for each workload curve.

3.2 Hardware and Software

We configured the Oracle RAC on 10 Fujitsu PRIMERGY RX300 S7 servers. We connected the 10 servers to a Brocade G630 switch with 32Gb FC. The AFF A800 nodes were also connected to this switch through 32Gb FC. The AFF A800 HA pair contained 24 x 1.5TB internal solid-state drives (SSDs).

Tables 1 and 2 list the hardware and software components that we used for the test configuration.

Table 1) Oracle host hardware and software components.

Hardware and Software Components	Details
Oracle Database 12c servers	10 Fujitsu PRIMERGY RX300 S7 servers
Server operating system	SLES 12.3 with 4.4.128-1.1.g286ae20 kernels
Oracle database version	12.2.0.1 (RAC)
Processors per server	Two 6-core Intel Xeon E5-2630 v2, 2.60GHz, 6c/12t
Physical memory per server	128GB
FC network	32Gb FC with multipathing
FC host bus adapter (HBA)	Emulex LPe32002-M2 32Gb
Dedicated public 1GbE ports for cluster management	Two Intel 1350GbE ports

Hardware and Software Components	Details
32Gb FC switch	Brocade G630 128-port switch
10GbE switch	Cisco Nexus 5596 switch

Table 2) NetApp AFF A800 storage system hardware and software.

Component	Details
Storage system	AFF A800 controller, configured as a HA pair
ONTAP version	9.5 RC1 (Tests were also repeated with 9.4 general availability [GA].)
Total number of drives	24
Drive size	1.5TB
Drive type	NVMe-SSD
FC target ports	Eight 32Gb ports (four per node)

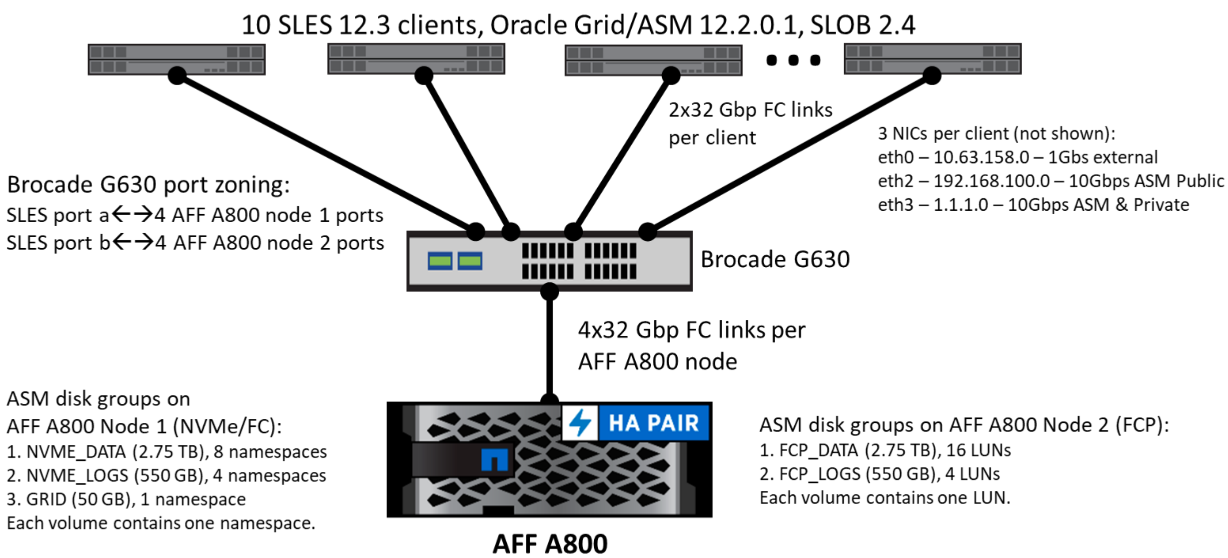
3.3 Network and Database Storage Design

Figure 1 shows the FCP SAN deployed with a Brocade G630 32Gb FCP switch. Each storage node had four target ports connected to the FC switch. Each host had one initiator port connected to the switch with Broadcom LPe32002 FC HBAs.

For Ethernet connectivity, each of the 10 hosts had a 1Gbps link for external access and a 10Gbps link for the Automatic Storage Management (ASM) public network. Each host also had a 10Gbps link for ASM and the private network.

Each of the 10 SLES hosts had one FC port connected to the Brocade switch. Each AFF A800 storage node had four FCP target ports that were also connected to the same switch, for eight total connected target ports. We configured the Brocade switch with port zoning to map the initiator port of each SLES host to all four target ports of each AFF A800 storage node.

Figure 1) Network design.



3.4 Database Layout and Storage Provisioning Design

A namespace is nonvolatile memory storage that is formatted for block access. A namespace is analogous to a logical unit number (LUN), which resides in a volume in the storage array. Both LUNs and namespaces show up as block storage devices at the host. An ASM disk group can be created across a set of LUNs or, similarly, across a set of namespaces.

Figure 1 shows the layout of LUNs, ASM disk groups, and SLOB databases for the NVMe/FC and FCP test configurations. One storage virtual machine (SVM) was created for the FCP configuration. This SVM contained all logical interfaces (LIFs), LUNs, and volumes for FCP. In the NVMe/FC configuration, four

SVMs were used. Each of the NVMe/FC dedicated SVMs contained one LIF and one quarter of the namespaces.

Of the 24 SSDs in the AFF A800, 23 of the drives were used to create a single NetApp RAID DP® aggregate and one was left as a spare drive. An aggregate was created for each of the two AFF A800 nodes.

In the FCP configuration, we created 16 x 176GB data LUNs for a total size of 2.75TB, and 4 x 137GB redo log LUNs for a total size of 550GB. One volume was assigned to each LUN.

In the NVMe/FC configuration, we created 8 x 352GB data namespaces for a total size of 2.75TB, and 4 x 137GB redo log namespaces for a total size of 550GB. One volume was assigned to each namespace.

The Oracle ASM Configuration Assistant was used to create ASM disk groups. In both configurations, an ASM disk group was created for data and redo logs, spanning the associated data and redo volumes.

The SLOB databases were generated and populated using the SLOB toolkit before running the performance workloads. A 1950GB tablespace was created in each data disk group. The SLOB database was populated with data for 300 users and a SLOB SCALE factor of 5,243M. This arrangement resulted in 1.5TB of SLOB data. A 150GB temporary tablespace was also created in each data disk group.

We manually modified the SLES FCP Device Mapper Multipathing (DM-Multipath) devices to use the “deadline” scheduler.

Another disk group was created for the Oracle Grid repository (CRS and Voting), on a single 50GB namespace. The Grid repository was shared on both configurations.

The ASM Configuration Assistant was used to create the ASM disk groups. Table 3 lists the settings used to create disk groups.

Table 3) ASM disk group settings.

Setting	Value
ASM compatibility	12.2.0.1
Database compatibility	12.2.0.1
Sector size	512B for FCP; 4KB for NVMe/FC
Logical sector size	512B for FCP; 4KB for NVMe/FC
Allocation units size	64MB

3.5 Workload Design

In this study, SLOB 2.4.2 was used as an Oracle I/O workload generation tool. SLOB can drive massive scale SQL execution against an Oracle database to simulate an OLTP workload.

A set of SLOB workloads was designed to ramp from 3 to 450 users with approximately 15 intermediate points. Each data point ran a fixed number of users for 20 minutes. This setup allowed us to gather performance metrics at a range of different load points and determine peak performance. Metrics were collected by SLOB in Oracle AWR reports. Each set of data points was run three or more times to generate repeatable results. All sets of workloads were run on two configurations: NVMe/FC and FCP.

Two different workload mixes were run:

- 100% SELECTs (100% reads)
- 75% SELECTs with 25% UPDATEs (an approximately 80:20 read-to-write ratio)

Keep in mind that this test was not designed to have high levels of caching on the 10 Linux hosts in the Oracle cluster. We wanted to demonstrate the capabilities of the AFF storage controller serving I/O in this workload. If we wanted to increase the SLOB throughput even further, additional caching could be configured on the Oracle servers. This setup would service more requests (especially reads) from memory on the Oracle servers, reduce the percentage of requests going to the AFF storage, and increase overall SLOB throughput.

Note: We took care in these test steps to simulate real database and customer workloads. However, we acknowledge that workloads can vary across databases. In addition, these test results were obtained in a closed lab environment with no competing workloads on the same infrastructure. In a typical shared-storage infrastructure, other workloads share resources. Therefore, your results might differ from the results described in this report.

3.6 Performance Test Results

We measured the performance of our Oracle database implementation with the AFF A800, using both the FCP and NVMe/FC protocols, with both ONTAP 9.4 and 9.5. All other hardware and software were configured identically.

Figure 2 shows the results of these tests with a 100% SELECT workload and Figure 3 with a 75% SELECT and 25% UPDATE workload.

In both charts, the x-axis is a sum of the total physical reads and writes per second (IOPS). Note that the IOPS metric is from the perspective of the Oracle database servers. It shows the IOPS that were directed to the AFF storage, not the IOPS that were serviced directly by the cache on the 10 Linux boxes running the Oracle database cluster.

The y-axis represents the read latency. Again, this is from the perspective of the Oracle RAC nodes and includes the FC transport time. Green triangle markers represent FCP protocol tests. Blue circle markers are NVMe/FC. Dark green and dark blue represent ONTAP 9.5. Light green and light blue are ONTAP 9.4.

In Figures 2 and 3, we illustrate the performance of a two-node AFF A800 controller by extrapolating the same workload on both storage nodes. Specifically, we show the performance of a 10-node Oracle RAC database on storage node 1 and a 10-node Oracle RAC database on storage node 2.

Figure 2) Two-node AFF A800 Oracle database performance with 100% SELECT workload.

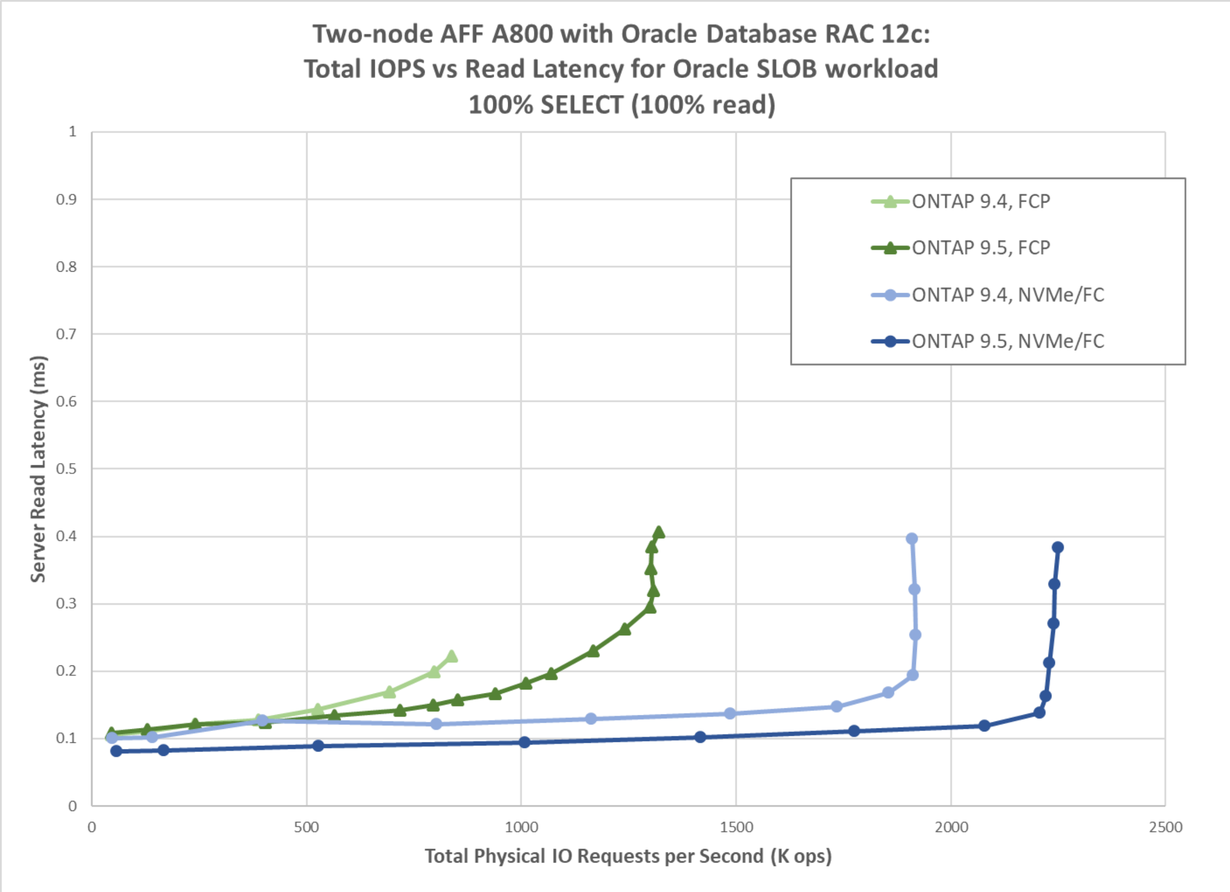
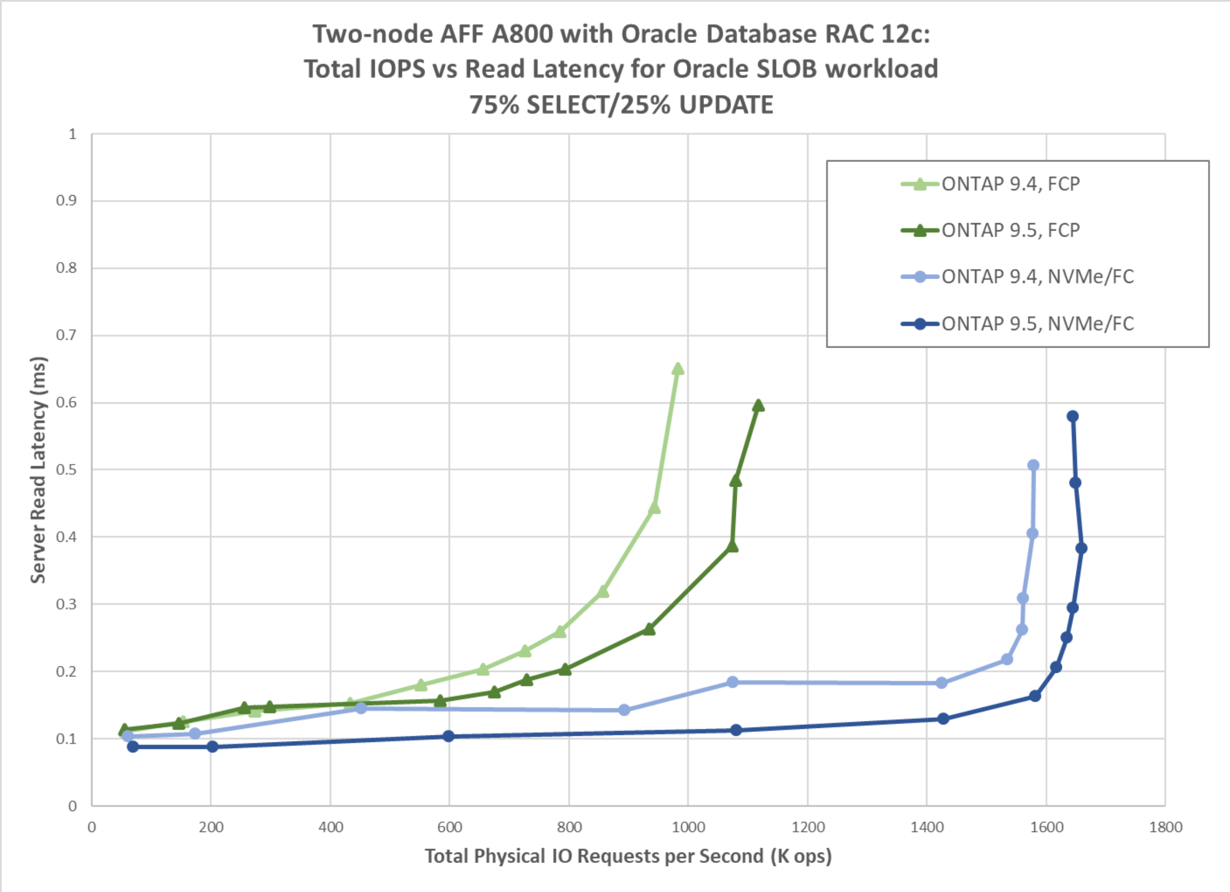


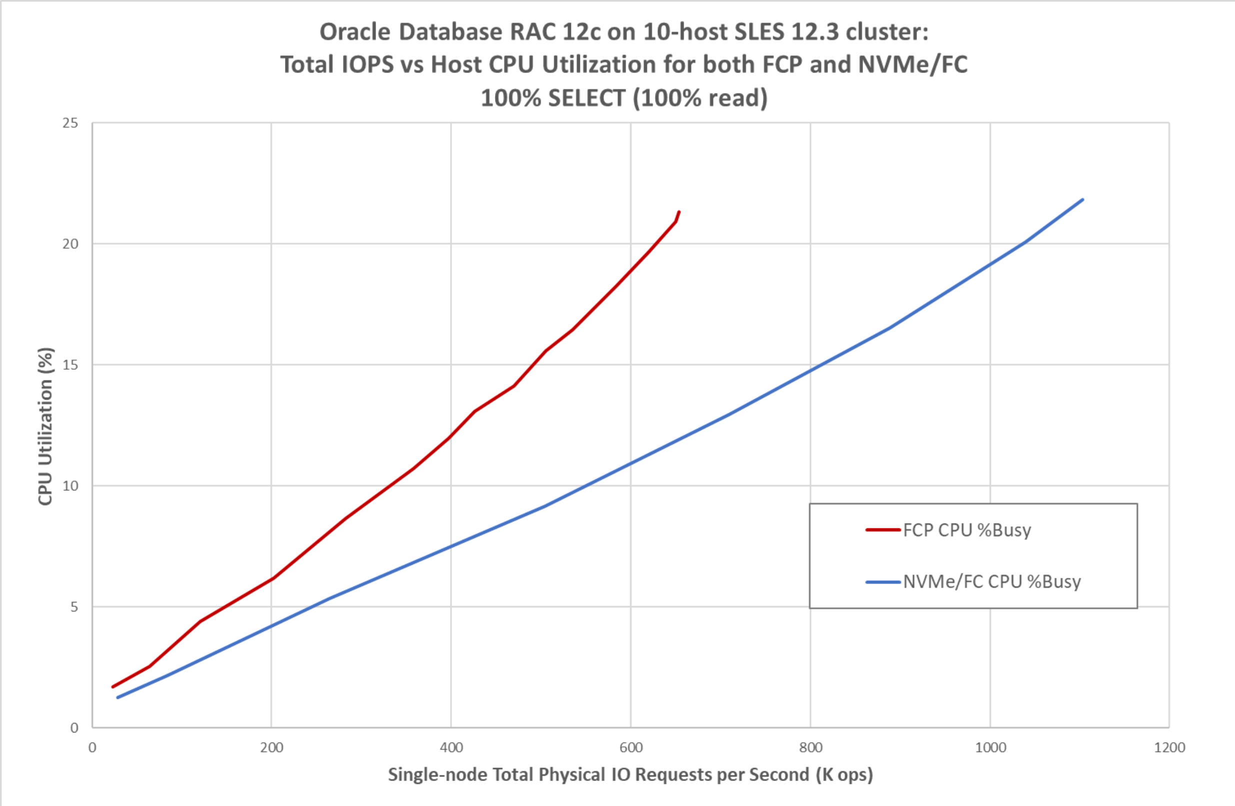
Figure 3) Two-node AFF A800 Oracle database performance with 75% SELECT / 25% UPDATE workload.



In our 10-host Oracle RAC, we monitored CPU utilization of the hosts during FCP and NVMe/FC performance tests. Hosts running the 100% SELECT workload with NVMe/FC showed up to a 43% reduction in CPU utilization when compared to the same IOPS load point with an FCP configuration. Extra CPU cycles on the hosts could be translated into fewer cores needed and potentially reduced software licensing costs.

In Figure 4, we illustrate CPU utilization based on the IOPS from one host from the 10-node Oracle RAC.

Figure 4) Host CPU utilization improvements with NVMe/FC.



Oracle Automatic Workload Repository (AWR) reports were collected during all workloads. The Database Summary section of the AWR shows the elapsed time and the database time for that specific performance point. The Top Timed Events section shows the top 10 events and their respective latencies. The System Statistics - Per Second section shows the number of physical reads and writes per second and the number of redo log operations in kilobytes per second. For details from an example AWR report, see the appendix to this document.

4 Conclusion

We found that the NetApp AFF A800 running ONTAP 9.5 with NVMe/FC generated very high IOPS at consistently low latencies when serving an Oracle Database 12c OLTP workload. ONTAP 9.5 with NVMe/FC achieved up to 70% higher IOPS than FCP while serving data with a 53% lower latency. This workload reached 2,200K IOPS with 0.14ms latency.

Another benefit of NVMe/FC is a reduction in host CPU utilization. This feature frees up CPU cycles on the hosts, which can lead to more compute available for applications or fewer total CPU cores required.

The AFF A800 running ONTAP 9.5 with FCP was able to serve 1,300K IOPS. At the 0.23ms latency point, FCP achieved 39% higher IOPS with ONTAP 9.5 than with ONTAP 9.4.

AFF A800 FCP customers should consider moving from ONTAP 9.4 to ONTAP 9.5 for immediate performance gains. Customers should also investigate how they can move their workloads to NVMe/FC for an even more impressive increase in IOPS and decrease in latency.

Appendix: AWR Report

The following three screenshots show the AWR report that we collected at the 537K IOPS point of the NetApp AFF A800 performance test.

Data points from these single-node measurements do not map directly to the two-node performance results graphed above.

System Statistics - Per Second

I#	Logical Reads/s	Physical Reads/s	Physical Writes/s	Redo Size (k)/s	Block Changes/s	User Calls/s	Execs/s	Parses/s	Logons/s	Txns/s
1	54,938.15	46,983.39	11,325.92	3,612.85	23,027.23	3.21	730.18	6.21	0.70	179.26
2	84,039.18	69,422.62	11,099.57	3,580.02	22,866.15	3.01	774.41	22.43	0.70	176.96
3	43,377.48	38,411.61	10,080.57	3,218.18	20,574.06	3.00	645.77	3.35	0.70	159.67
4	45,281.83	40,553.97	10,640.45	3,396.85	21,701.06	2.99	681.32	3.44	0.70	168.48
5	43,667.33	39,495.53	10,284.31	3,277.62	20,938.93	3.01	654.93	2.64	0.70	162.54
6	42,830.87	38,479.18	10,063.09	3,211.37	20,528.99	3.00	643.31	2.98	0.70	159.28
7	45,328.96	40,274.44	10,608.98	3,393.54	21,683.36	3.00	699.07	7.21	0.70	168.34
8	42,562.08	38,203.47	9,999.56	3,191.01	20,397.96	3.01	641.59	3.24	0.70	158.33
9	47,299.07	44,823.85	11,237.08	3,585.39	22,668.47	3.16	710.60	3.08	0.75	176.04
10	37,663.16	35,619.37	8,918.08	2,851.28	18,035.38	3.00	566.20	3.11	0.70	139.95
Sum	486,988.10	432,267.44	104,257.61	33,318.11	212,421.59	30.39	6,747.38	57.67	7.06	1,648.85
Avg	48,698.81	43,226.74	10,425.76	3,331.81	21,242.16	3.04	674.74	5.77	0.71	164.88
Std	13,167.18	9,783.77	724.77	234.29	1,502.09	0.08	57.76	6.05	0.02	11.69

Total IOPS going to the AFF system are reported by physical reads per second and by physical writes per second in the Oracle AWR report.

WORKLOAD REPOSITORY REPORT (RAC)

Database Summary

Database								Snapshot Ids		Number of Instances		Number of Hosts		Report Total (minutes)	
Id	Name	Unique Name	Role	Edition	RAC	CDB	Block Size	Begin	End	In Report	Total	In Report	Total	DB time	Elapsed time
3558638588	FCPDB	FCPDB	PRIMARY	EE	YES	NO	8192	192	193	10	10	10	10	3,715.41	20.09

Database Instances Included In Report

- Listed in order of instance number, I#

I#	Instance	Host	Startup	Begin Snap Time	End Snap Time	Release	Elapsed Time(min)	DB time(min)	Up Time(hrs)	Avg Active Sessions	Platform
1	FCPDB1	s1	28-Sep-18 21:58	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	391.61	41.68	19.50	Linux x86 64-bit
2	FCPDB2	s2	28-Sep-18 23:24	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.10	435.20	40.24	21.65	Linux x86 64-bit
3	FCPDB3	s3	28-Sep-18 23:25	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	363.56	40.23	18.10	Linux x86 64-bit
4	FCPDB4	s4	28-Sep-18 23:25	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	386.54	40.23	19.25	Linux x86 64-bit
5	FCPDB5	s5	28-Sep-18 23:25	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	384.91	40.22	19.17	Linux x86 64-bit
6	FCPDB6	s6	28-Sep-18 23:25	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	360.19	40.22	17.93	Linux x86 64-bit
7	FCPDB7	s7	28-Sep-18 23:26	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	385.61	40.21	19.20	Linux x86 64-bit
8	FCPDB8	s8	28-Sep-18 23:26	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.10	365.98	40.21	18.21	Linux x86 64-bit
9	FCPDB9	s9	28-Sep-18 23:26	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	360.93	40.20	17.97	Linux x86 64-bit
10	FCPDB10	s10	28-Sep-18 23:27	30-Sep-18 15:18	30-Sep-18 15:38	12.2.0.1.0	20.08	280.87	40.19	13.99	Linux x86 64-bit

Top Timed Events

- Instance '*' - cluster wide summary
- '*' Waits, %Timeouts, Wait Time Total(s) : Cluster-wide total for the wait event
- '*' 'Wait Time Avg' : Cluster-wide average computed as (Wait Time Total / Event Waits)
- '*' Summary 'Avg Wait Time ' : Per-instance 'Wait Time Avg ' used to compute the following statistics
- '*' [Avg/Min/Max/Std Dev] : average/minimum/maximum/standard deviation of per-instance 'Wait Time Avg'
- '*' Cnt : count of instances with wait times for the event

	Wait		Event		Wait Time			Summary Avg Wait Time				
I#	Class	Event	Waits	%Timeouts	Total(s)	Avg Wait	%DB time	Avg	Min	Max	Std Dev	Cnt
*	User I/O	db file sequential read	482,475,845	0.00	186,815.23	387.20us	83.80	387.60us	357.83us	408.24us	17.58us	10
		DB CPU			33,800.32		15.16					10
	Other	latch: parallel query alloc buffer	30,848	0.00	12,961.62	420.18ms	5.81	450.14ms	144.51ms	941.00ms	230.80ms	10
	System I/O	db file parallel write	12,504,066	0.00	3,249.56	259.88us	1.46	267.46us	196.97us	297.97us	38.76us	10
	System I/O	log file parallel write	1,958,702	0.00	1,808.32	923.22us	0.81	924.10us	878.63us	.95ms	21.76us	10
	User I/O	direct path read	176,070	0.00	552.02	3.14ms	0.25	1.81ms	431.50us	3.16ms	1.54ms	10
	Other	buffer exterminate	48,855	84.96	490.66	10.04ms	0.22	10.16ms	8.82ms	11.48ms	818.76us	10
	Other	SGA: allocation forcing component growth	46,986	27.61	428.58	9.12ms	0.19	23.74ms	2.63ms	46.03ms	15.19ms	10
	Other	RMA: IPC0 completion sync	12,600	0.00	244.25	19.38ms	0.11	19.38ms	19.33ms	19.42ms	25.05us	10
	User I/O	db file scattered read	290,105	0.00	228.06	786.12us	0.10	871.46us	676.95us	1.18ms	174.21us	10

Where to Find Additional Information

To learn more about the information that is described in this document, review the following documents and/or websites:

- The Silly Little Oracle Benchmark v2.4.2 (SLOB2)
  http://kevinclosson.net/2012/02/06/introducing-slob-the-silly-little-oracle-benchmark/
- TR-4582: NetApp AFF A700 Performance with Oracle Database
  http://www.netapp.com/us/media/tr-4582.pdf
- NetApp AFF A-Series All Flash Array product webpage
  http://www.netapp.com/us/products/storage-systems/all-flash-array/aff-a-series.aspx

Refer to the [Interoperability Matrix Tool (IMT)](#) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

NetApp