
Article

[Mark Bolinsky](#) · Jul 10, 2018 4m read

Storage Performance Series - NetApp All-Flash FAS (AFF) A300

Often InterSystems technology architect team is asked about recommended storage arrays or storage technologies. To provide this information to a wider audience as reference, a new series is started to provide some of the results we have encountered with various storage technologies. As a general recommendation, all-flash storage is highly recommended with all InterSystems products to provide the lowest latency and predictable IOPS capabilities.

The first in the series was the most recently tested Netapp AFF A300 storage array. This is middle-tier type storage array with several higher models above it. This specific A300 model is capable of supporting a minimal configuration of only a few drives to hundreds of drives per HA pair, and also capable of being clustered with multiple controller pairs for tens of PB's of disk capacity and hundreds of thousands of IOPS or higher.

Storage Configuration and Linux LVM Setup

The configuration tested included the following details:

- dual controllers
- 48 x SAS SSD split evenly between both controllers (2 x 24 disk aggregates)
- 32Gb FC Host ports

The server was a physical server with 24-cores running RHEL 7.5 64-bit Linux with a two dual-port 32Gb FC HBAs (4 ports total).

To maximize IO capabilities of both controllers and all disk devices, multiple LUNs were presented to the host. The following LUNs were created for each of the database components:

- Database LUNs = 8 x 512GB LUNs (4TB total) - - - > `vgirisdb`
- Journal LUNs = 2 x 512GB LUNs (1TB total) - - - > `vgirisjr`
- WIJ LUNs = 2 x 256GB LUNs (512GB total) - - - > `vgiriswij`
- Install and IRISTEMP LUNs = 2 x 128GB LUNs (256GB total) - - - > `vgirisys`

Three separate LVM volume groups were created using the Linux Multipath (`/dev/mapper/*`) devices for each of the four areas above, and all used a 4MB physical extent (PE) size. Details of LVM PE striping setup can be found [here](#) in another community article.

InterSystems Iris Installation and Setup

InterSystems Iris is installed in `vgirisys` volume group in a separate logical volume mounted as `/iris/sys`. The configuration file (`iris.cpf`) was modified to include the parameters to enable async IO for database IO and async WIJ writes. This also bypasses Linux file cache so all IO performed during the testing is from the storage array itself and not hitting OS file cache so an accurate measurement of storage array performance can be observed.

The `vgirisdb` volume is used to create a separate logical volume mounted as `/iris/db`. This contained the 1TB database to be used for the testing. The journal and wij locations were also configured to separate logical volumes

as /iris/wij and /iris/jrn within the vgiriswij and vgirisjrn volume groups respectively.

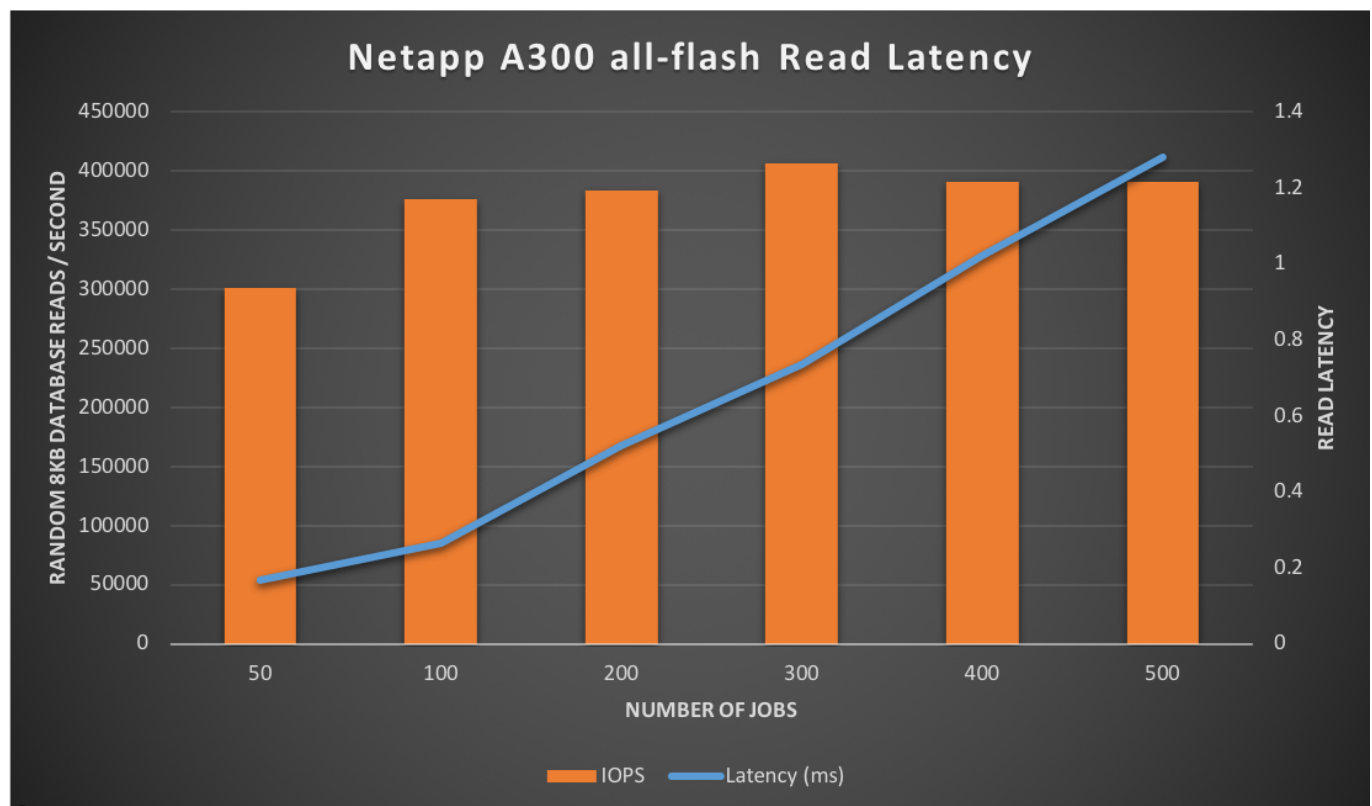
A random read workload from within the database instance is used to generate the IO workload in a progressive number of jobs. The 1TB database is the target of the IO workload. Details of the RANREAD tool used can be found [here](#).

Testing Results

Starting with only 50 jobs, the storage throughput was surprising with a sustained 301,000 IOPS with an impressive 166 microsecond (μ s) latency. Increasing the number of jobs was linear in terms of throughput with minimal increase in latency until reaching a plateau of 390-400k IOPS. Once reaching that point the number of IOPS stayed flat at the same rate and only increased with minimal and predictable latency as the number of jobs increased. One other observation is that total CPU utilization in the server itself was near 100% as well during the test runs with a higher number of concurrent jobs, so that is a factor to be considered as well. This throughput and latency behavior is ideal and shows a predictable response time to utilization curve while maintaining maximum throughput.

Figure-1 illustrates the predictable performance and sustained throughput for the configuration as tested with merely only a total of 48 SSD disk devices. This array supports up to 384 SSD devices. Linear scaling with more disk devices based on these test results are not assumed due to potentially reaching other bottlenecks or bandwidth saturation within the IO sub-system (such as FC paths, controller processors, back-end SAS bus, etc...) or the server running the test itself.

Figure-1: Predictable latency while maintaining maximum throughput



Conclusion

The NetApp AFF A300 provided a very capable storage platform with excellent response times while sustaining throughput. Additional details of the Netapp AFF A-series storage array portfolio can be found [here](#). This array has been added to our Hardware Compatibility List¹ (HCL) for both HealthShare and TrakCare application products. An official HCL for storage arrays is not maintained for our data platforms products including InterSystems Iris, Cache', and Ensemble since the application running on InterSystems data platforms will have varying response

time and throughput requirements based on the application needs. Storage performance requirements need to be determined by the application partner or end user. The HCL only applies to InterSystems HealthShare and TrakCare applications.

[#Deployment](#) [#High Availability](#) [#InterSystems Business Solutions and Architectures](#) [#Performance](#) [#System Administration](#) [#Testing](#) [#InterSystems IRIS](#)

Source URL: <https://community.intersystems.com/post/storage-performance-series-netapp-all-flash-fas-aff-a300>