



## Speed up demanding workloads, save admin hassle, and maximize storage capacity with a Dell EMC Isilon All-Flash Storage System

Frametest reported more throughput and frames per second (FPS) on the Dell EMC Isilon F800 than on a Vendor A array, for faster project completion times

For organizations with demanding video editing or machine learning workloads, the storage solution you choose can help you succeed, or it can threaten failure. Either it can speed tasks and shorten completion times, reduce the management burden, and make the most of the storage you paid for—or it can hinder your operations.

In the Principled Technologies data center, we compared high-throughput media and entertainment workload performance on the Dell EMC™ Isilon® F800 All-Flash NAS Storage System to that of a similar all-flash storage platform from a competitor (hereby referred to as "Vendor A"). The Isilon F800 storage system delivered significantly better read (video playback) and write (video editing) performance, which means videographers, media library managers, video archivists, and other media-focused users can complete tasks faster and improve productivity. In addition, we tested machine learning workloads on the updated Dell EMC Isilon F810, which adds compression. We found it delivered comparable performance to the Vendor A array while offering more usable storage capacity, which lets you use the storage you paid for. Further, the Dell EMC Isilon F810 was easier to configure and provision than the Vendor A array.

For your video projects, machine learning, and other demanding workloads, we proved that the Dell EMC Isilon All-Flash F800 and F810 storage systems can deliver comparable or faster performance versus the Vendor A platform. The Isilon F810 can also improve ease of management while offering more usable storage space through compression.

*We completed Frametest testing on a Dell EMC Isilon F800, but completed all other testing on an updated Dell EMC Isilon F810.*

**Up to 55% more MB/s and FPS while reading data\***

Consistent video playback on the Dell EMC Isilon F800



**Up to 47% more MB/s and FPS while writing data\***

Faster video editing on the Dell EMC Isilon F800



**Get 32.3% more effective usable capacity with compression\*\***

on the Dell EMC Isilon F810



**Configure and provision Isilon storage more quickly and easily**

per testing on the Dell EMC Isilon F810

\* compared to the array of Vendor A

\*\*with an efficiency rate of 1.96:1

## About Frametest

To test read/write throughput, we used the Frametest utility, which simulates writes and reads at a user-specified number of individual frames at a certain resolution. This emulates raw still frames or frames generated by post-processing or 3D rendering software. To download Frametest, visit <https://support.dvsus.com/hc/en-us/articles/212925466-How-to-use-frametest>.

## Big video files need fast storage

Videographers and other high-quality digital media users require many tools to perform daily work. These tools demand far greater performance from hard disk drive (HDD) or solid-state drive (SSD) storage than the standard productivity software many businesses use. Each day, these professionals spend hours editing their footage to create the perfect video—and a large 4K video project requires far more storage capacity and performance than the average corporate workstation or laptop can provide. Video assets live safely in the company data center, where colleagues can connect to access footage and collaborate to meet client goals.

Today's 4K, high-resolution video looks great but presents a challenge: the amount of data that must be available to users is simply immense, requiring ultra-fast storage with high throughput to let editors scrub through video assets quickly, make changes without waiting for the storage to catch up, and play back the finished results without hitching. Otherwise, low frame rates and frustrating pauses could create unnecessary hurdles to completing work on time.

In our data center, we used Frametest, a synthetic benchmark utility that can evaluate storage performance specifically for video editing usage. We tested at various storage node counts, up to two full chassis, with varying numbers of clients (employees) accessing the storage.

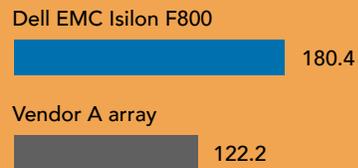
Through most of those tests, the Dell EMC Isilon F800 offered more FPS and greater MB/s in throughput compared to the Vendor A mid-range market storage array we tested against. During write testing, which offered an I/O pattern similar to that of video capture or ingestion, the Isilon F800 supported up to 47.7 percent more FPS and throughput compared to the Vendor A mid-range market array. During read testing, which used an I/O pattern similar to that of video playback, the Isilon F800 supported up to 55.1 percent more FPS and throughput. We observed both disparities when our Dell EMC Isilon F800 solution had two full chassis containing eight storage nodes accessed by 12 clients.

### What's the Dell EMC Isilon Storage System?

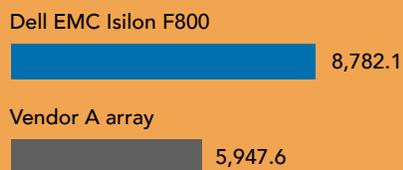
The Dell EMC Isilon platform is a scale-out network-attached storage (NAS) array, powered by the OneFS operating system, that is intended to house unstructured data. Isilon storage systems come from one of three product lines that focus on different storage technologies: all-flash SSD technology, a hybrid of SAS and SATA HDDs, and archive arrays that use SATA HDDs.

We tested the Isilon F800 and F810 All-Flash scale-out NAS storage systems. According to Dell EMC, the F800 aims to combine "extreme performance and scalability with the massive efficiency and enterprise grade capabilities." Each Isilon F800 chassis can support up to 60 SSDs for up to 924 TB. Dell EMC lists potential uses for the F800 as digital media (broadcasting, real-time streaming, or post-production), electronic design automation, and genomics DNA and RNA sequencing.<sup>1</sup> You can learn more about the Dell EMC Isilon platform at [DellEMC.com/Isilon](https://DellEMC.com/Isilon).

### Write testing FPS\*



### Write testing throughput (MB/s)\*



By letting users access and make changes to large video files faster, the Dell EMC Isilon F800 could improve the productivity of employees working with these large files. For more results at other node and client counts, see Appendix A. To see the technical details of our two solutions, see Appendix B, and to see how we tested, see Appendix C.

### Read testing FPS\*



### Read testing throughput (MB/s)\*



\*Higher is better

## Assessing performance through machine learning and overall ease of use

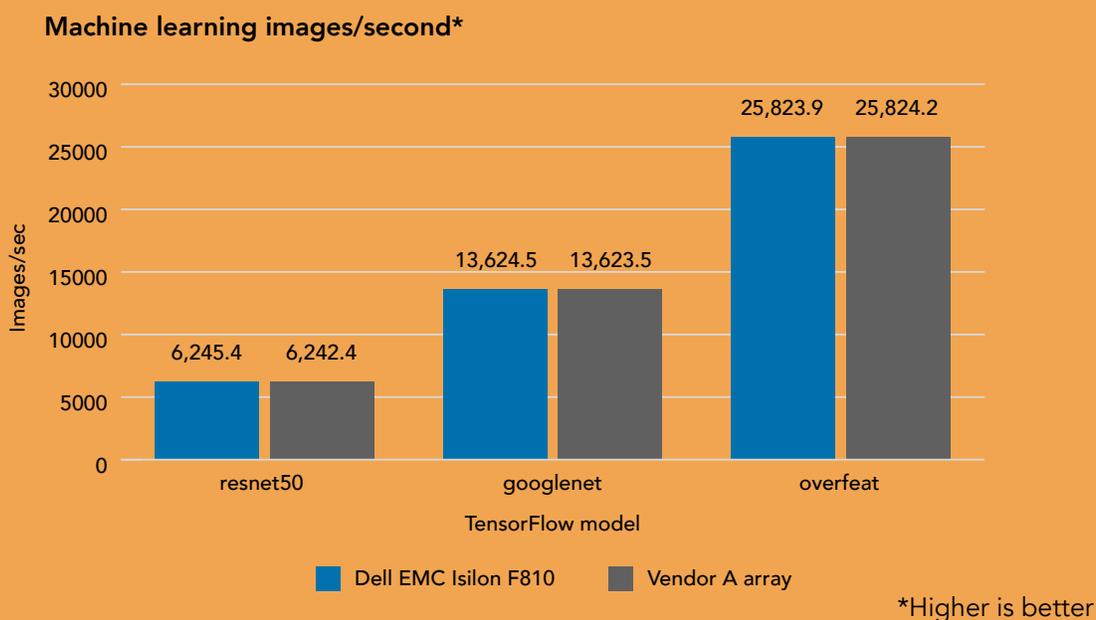
Machine learning is a form of artificial intelligence that helps organizations analyze large amounts of data and turn it into something they can use. These demanding workloads require robust compute and storage resources, so testing with popular TensorFlow machine learning models can further validate the performance of a storage solution.

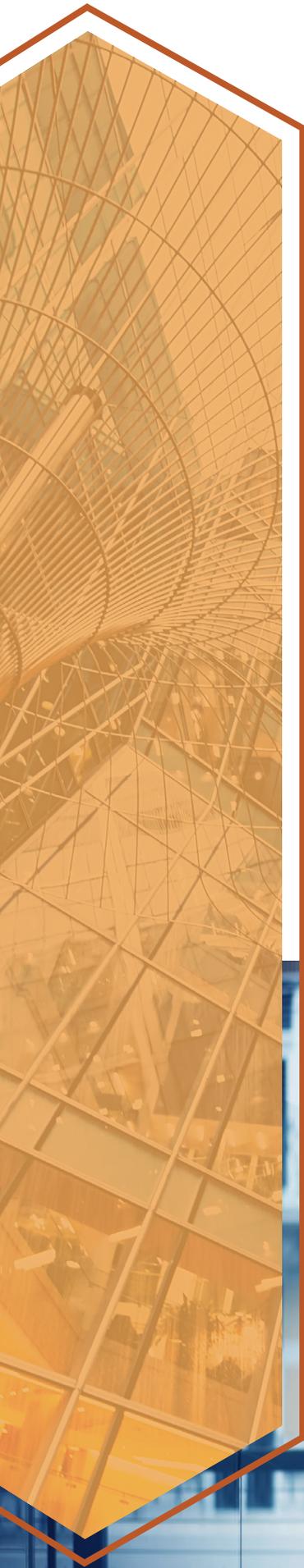
We ran three TensorFlow models (resnet50, googlenet, and overfeat), and found that both the Dell EMC Isilon F810 and Vendor A array handled the machine learning tasks we tested.

For complex tasks like machine learning, the Dell EMC Isilon F810 and Vendor A array differ when it comes to ease of use: we found that it was easier to configure and provision the Dell EMC Isilon F810 storage system and had no problems making policy changes.

While our experiences setting up the arrays was similar, the Dell EMC Isilon F810 storage system delivered an overall simpler, more streamlined experience based on four key advantages over the Vendor A array:

- Isilon has a clearer, easier-to-use UI that allows admins to configure and provision storage as needed.
- Isilon organizes and pools storage more intuitively than the Vendor A array. The Isilon array we tested used all 60 drives to create a single pool, which was immediately available as usable file storage. Our admin could then apply storage policies to files and folders or globally, as needed. Admins could treat this storage as a single pool to tweak, adjust, and manipulate as necessary.
- Isilon management uses more automation. For example, Isilon can apply policy changes automatically in the background (though admins can adjust the settings as needed).
- Admins have more flexibility when changing policies that affect redundancy (node and drive), retention (snapshots), efficiency (compression/deduplication), performance (random vs. sequential), and security (permissions). Admins can change these policies to files, folders, or globally after they've set them, in a non-destructive way.





Admins can begin setting up and using Isilon with only a general plan; admins can configure and provision storage on the fly to adapt to changing needs. This allows for flexibility—admins don't need to feel as if they're experts before managing Isilon.

In contrast, making policy changes or changes to the Vendor A array required much more thought and attention to detail from an admin. This means that admins must seriously plan at the start of deployment as later changes present significant challenges. In our hands-on testing, we had to perform these additional steps on the Vendor A array:

- The vendor-preferred method for provisioning requires assigning drives or parts of drives to an individual pool on a single node. The Vendor A array then aggregates these pools together to create a larger virtual pool. Admins cannot resize the individual pools easily or change their redundancy after they have been allocated. If an admin resizes the pools or adds additional pools, the storage may be unbalanced, which could affect its performance.
- Create a container that includes assigned and allocated volumes. Admins can expand these volumes but not shrink them. At this point, an admin must determine what type of network file server to use. The admin assigns that server to a volume, assigns networking to that server, and applies all permissions. An admin must complete all the previous steps before using the array.
- After assigning network file servers to volumes, admins must configure retention and efficiency on a per-volume basis. The Vendor A array applies additional efficiency tuning at the pool level. An admin must schedule retention and efficiency changes, which could cause storage performance to take a hit.
- If an admin decides they don't want volumes to be in the pooled storage, the admin would have to move data off the current volumes and redo nearly everything from the beginning.





On the Vendor A array, admins can change only permissions to files and they must apply efficiency, retention, performance, and redundancy at the volume or pool level. For example, if your admin wanted to change efficiency and retention on a specific folder, they would have to do so at the volume level. This would apply to all folders on that volume regardless of whether those folders needed the changes.

Because it's easier to use, the Dell EMC Isilon F810 array could be managed by any number of less experienced admins in your existing IT staff. To manage the Vendor A array, your organization would likely need a set of dedicated admins with detailed knowledge and experience with the Vendor A platform. Choosing Isilon could reduce the time an admin spends managing the solution, the amount of storage admin training your organization needs to conduct or fund, and the need to hire specialized admins.

## Get the storage capacity you paid for with Dell EMC Isilon F810

You need a powerful external storage solution for a reason: If you create and store large files such as videos, you need the room to keep them. That's why it can be frustrating when the usable capacity of an array isn't what you anticipated.

We found that the Dell EMC Isilon F810 All-Flash Storage System offers more usable capacity than the array from Vendor A, 22.2 percent more capacity from more efficient data protection alone, or 32.3 percent more capacity when using compression (this assumes an efficiency rate of 1.96:1). Please note that these capacity numbers are for the Dell EMC Isilon F810, which added a new compression feature that enables greater capacity.

### Effective usable (TB)\*

Dell EMC Isilon F810



Vendor A array



\*Higher is better

## How we determined usable storage capacity

We configured the array from Vendor A at maximum disk capacity with 48 disks for approximately 42 TB of raw storage. Once we provisioned the storage, the total usable storage capacity dropped to 27.48 TB, or 65 percent of the total raw capacity. This means customers pay for 35 percent storage overhead that they can't use.

We configured the Isilon array with 60 disks. However, in a scenario where the Isilon F810 array is configured to match the same number of disks (48 disks) and capacity as array from Vendor A, the Isilon array should have a larger usable capacity of 33.6TB, which is 22.2 percent more usable capacity than the array from Vendor A. This lower storage overhead and increase in usable capacity is due to a more efficient use of erasure coding and file-level protection instead of the drive-level protection on RAID configuration that the Vendor A array employs. We calculated the usable storage figures for Isilon using the 80% raw-to-usable ratio advertised for Isilon clusters of five or more nodes (See: <https://www.emc.com/collateral/hardware/white-papers/h10719-isilon-onefs-technical-overview-wp.pdf> for details.)

We also performed an efficiency test. Not all storage arrays report data efficiency the same way. The Vendor A array presents a data reduction ratio that compares the effective-usable capacity to the reported usable. This ratio automatically takes into account the raw capacity sacrificed for redundancy by comparing to the capacity that is left after the volumes are created. The Isilon reports an efficiency ratio that compares effective-usable capacity to raw capacity. Because Isilon lets users set redundancy ratios at the file level instead of volume level, every user will have different capacity lost to redundancy. We found that when we presented a similar dataset to both arrays, the array from Vendor A delivered a 2.28:1 data reduction rate vs. a 1.96:1 efficiency rate on Isilon. Due to the lower usable capacity on the array from Vendor A, Isilon is, in fact, capable of storing more data even if the rate appears lower. This is because the Isilon efficiency rate takes into the account the overhead for data protection and is relative to the total available raw storage, whereas the Vendor A data reduction ratio is based on the much smaller storage capacity available after drive level protection is applied. (Note: The efficiency rate is highly dependent on the dataset presented to the arrays and the amount of data set to high redundancy on the Isilon array.)

The table below shows the capacity for the two arrays configured with 100 TB of raw storage.

	Raw (TB)	Usable (TB)	Effective usable (TB)	Data reduction ratio	Efficiency ratio
Vendor A	100	65	148	2.28	1.48
Isilon	100	80	196	2.45	1.96

# Effective-usable (TB) for Vendor A is Usable (TB)\* Data Reduction Ratio. Efficiency Ratio is effective-usable (TB)/ Raw

## Effective-usable (TB) for Isilon is Raw (TB)\* Efficiency Ratio. Data reduction Ratio is Effective-usable (TB)/ Usable (TB)

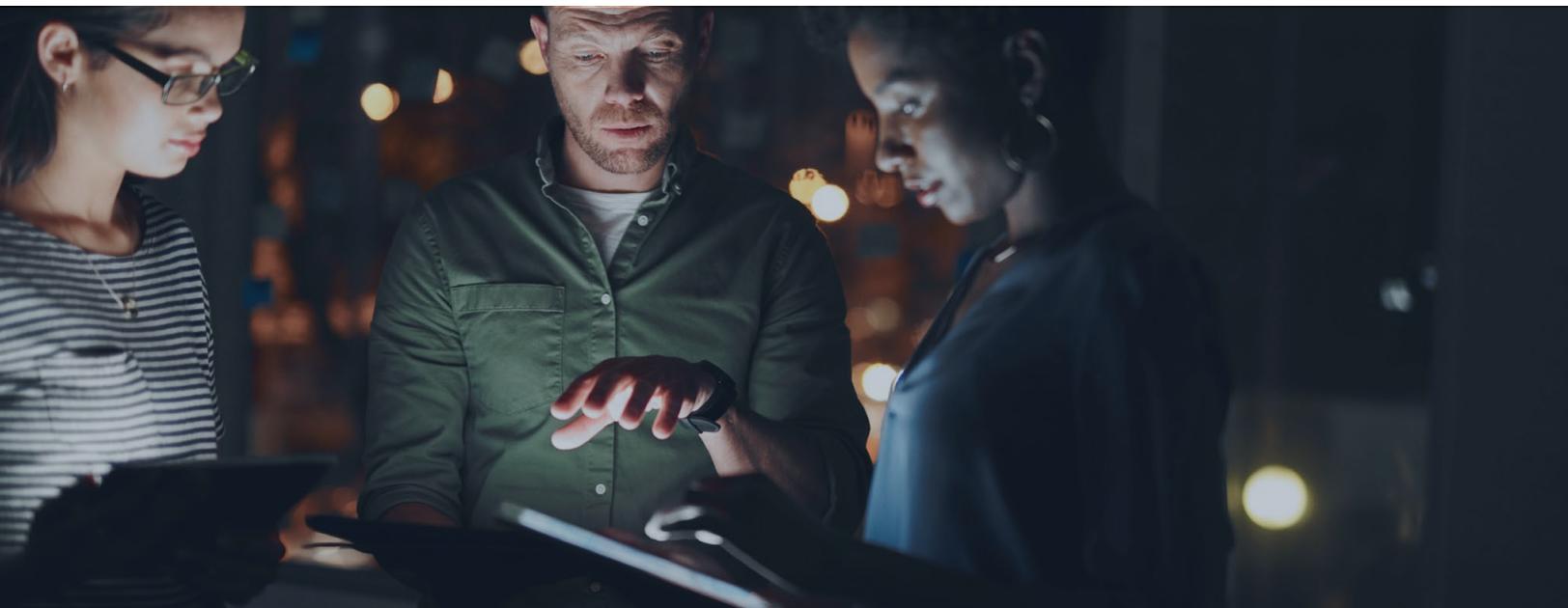


## Conclusion

If your organization works routinely with applications requiring high throughput, top-of-the-line storage is essential to give workers quick and easy access to data so they can finish projects on time. In our tests showcasing 4K video editing and playback performance, we found that the Dell EMC Isilon F800 All-Flash Storage System performed better in both read and write testing compared to a storage array from Vendor A. Greater throughput and FPS mean that the editing and viewing experience is more responsive, so workers can do their jobs better and faster. The related Dell EMC Isilon F810 also performed well, handling demanding machine learning workloads on par with Vendor A, but with a significantly better management experience. With benefits such as better throughput performance, improved ease of use, and a larger usable storage capacity, choosing a Dell EMC Isilon All-Flash Storage System can help your organization meet data challenges head on.

To learn more about Dell EMC Isilon, visit [DellEMC.com/Isilon](https://DellEMC.com/Isilon).

- 1 "Dell EMC Isilon Scale-out NAS product family," accessed January 3, 2019, <https://www.dell EMC.com/resources/en-us/asset/offering-overview-documents/products/storage/h10541-ds-isilon-platform.pdf>.
- 2 "Dell EMC Isilon OneFS: A technical overview," accessed December 20, 2018, <https://www.emc.com/collateral/hardware/white-papers/h10719-isilon-onefs-technical-overview-wp.pdf>.



We began testing with the Dell EMC Isilon F800 All-Flash Storage System. While we were testing, Dell EMC introduced the Isilon F810, which adds the ability to compress data. To test this new feature, Dell EMC upgraded our F800 to an F810 by replacing the HBAs and updating the OneFS version to 8.1.3. We completed Frametest testing for this report on the F800, and did all other testing on the F810.

We concluded our hands-on testing on the Dell EMC Isilon F800 Storage System on December 11, 2018. During testing, we determined the appropriate hardware and software configurations and applied updates as they became available. The Frametest results in this report reflect configurations that we finalized on November 14, 2018 or earlier.

We concluded our hands-on testing on the Dell EMC Isilon F810 Storage System on June 7, 2019. During testing, we determined the appropriate hardware and software configurations and applied updates as they became available. The TensorFlow, ease-of-use, and storage capacity results in this report reflect configurations that we finalized on June 7, 2019 or earlier.

Unavoidably, these configurations may not represent the latest versions available when this report appears.

## Appendix A: Our results

### Throughput and frames per second testing

Test mode write summary - Dell EMC Isilon F800						
# of clients	# of nodes	Total frame rate (FPS)	Total bandwidth (MB/s)	Total BW (Mb/s)	% client max	
1	1	48.40	2,356.61	18,852.88	94.26%	
2	1	41.69	2,029.96	16,239.68	40.60%	
3	1	45.56	2,218.40	17,747.20	29.58%	
4	1	47.18	2,297.19	18,377.52	22.97%	
2	2	96.72	4,709.02	37,672.16	94.18%	
4	2	76.66	3,732.30	29,858.40	37.32%	
6	2	82.41	4,011.83	32,094.64	26.75%	
3	3	120.47	5,865.22	46,921.76	78.20%	
6	3	109.86	5,348.32	42,786.56	35.66%	
9	3	118.06	5,747.85	45,982.80	25.55%	
4	4	116.74	5,683.94	45,471.52	56.84%	
8	4	138.40	6,738.58	53,908.64	33.69%	
12	4	148.02	7,207.30	57,658.40	24.02%	
8	8	166.88	8,124.29	64,994.32	40.62%	
12	8	180.38	8,782.14	70,257.12	29.27%	

Test mode read summary - Dell EMC Isilon F800						
# of clients	# of nodes	Total frame rate (fps)	Total bandwidth (MB/s)	Total BW (Mb/s)	% client max	
1	1	45.37	2,209.10	17,672.80	88.36%	
2	1	91.44	4,451.99	35,615.92	89.04%	
3	1	93.69	4,561.55	36,492.40	60.82%	
4	1	94.31	4,591.79	36,734.32	45.92%	
2	2	90.80	4,420.93	35,367.44	88.42%	
4	2	172.94	8,419.69	67,357.52	84.20%	
6	2	176.57	8,596.43	68,771.44	57.31%	
3	3	136.93	6,666.66	53,333.28	88.89%	
6	3	244.37	11,897.90	95,183.20	79.32%	
9	3	251.12	12,226.23	97,809.84	54.34%	
4	4	181.76	8,849.20	70,793.60	88.49%	
8	4	309.20	15,054.07	120,432.56	75.27%	
12	4	321.55	15,655.53	125,244.24	52.19%	
8	8	365.81	17,810.53	142,484.24	89.05%	
12	8	488.81	23,799.27	190,394.16	79.33%	

Test mode write summary - Vendor A array						
# of clients	# of nodes	Total frame rate (fps)	Total bandwidth (MB/s)	Total BW (Mb/s)	% client max	
1	1	23.54	1,146.24	9,169.92	45.85%	
2	1	34.39	1,674.46	13,395.68	33.49%	
3	1	47.21	2,298.07	18,384.56	30.64%	
4	1	57.35	2,792.09	22,336.72	27.92%	
2	2	33.19	1,615.98	12,927.84	32.32%	
4	2	60.39	2,939.64	23,517.12	29.40%	
6	2	79.99	3,893.91	31,151.28	25.96%	
3	3	51.26	2,495.55	19,964.40	33.27%	
6	3	82.38	4,010.48	32,083.84	26.74%	
9	3	107.97	5,256.60	42,052.80	23.36%	
4	4	59.30	2,886.78	23,094.24	28.87%	
8	4	103.74	5,050.37	40,402.96	25.25%	
12	4	123.02	5,989.87	47,918.96	19.97%	
8	8	101.24	4,929.16	39,433.28	24.65%	
12	8	122.15	5,947.55	47,580.40	19.83%	

Test mode read summary - Vendor A array						
# of clients	# of nodes	Total frame rate (fps)	Total bandwidth (MB/s)	Total BW (Mb/s)	% client max	
1	1	35.15	1,711.16	13,689.28	68.45%	
2	1	74.87	3,645.29	29,162.32	72.91%	
3	1	115.10	5,603.97	44,831.76	74.72%	
4	1	137.37	6,688.15	53,505.20	66.88%	
2	2	75.59	3,680.10	29,440.80	73.60%	
4	2	136.87	6,663.96	53,311.68	66.64%	
6	2	162.12	7,892.97	63,143.76	52.62%	
3	3	105.46	5,134.42	41,075.36	68.46%	
6	3	168.78	8,217.83	65,742.64	54.79%	
9	3	184.90	9,002.49	72,019.92	40.01%	
4	4	132.48	6,450.24	51,601.92	64.50%	
8	4	185.39	9,026.17	72,209.36	45.13%	
12	4	216.03	10,516.86	84,134.88	35.06%	
8	8	243.22	11,841.75	94,734.00	59.21%	
12	8	315.20	15,346.81	122,774.48	51.16%	

## Appendix B: System configuration information

The tables below present detailed information on the systems we tested. Both storage systems were high-end, all-flash platforms. We completed TensorFlow, ease-of-use, and storage capacity testing on an upgraded Isilon F810 array. To upgrade to the F810, Dell EMC replaced the HBAs in our F800 array and updated the OneFS version to 8.1.3, which includes compression.

Storage configuration information	Dell EMC Isilon F800	Vendor A array
Operating system	OneFS version: 8.1.2.0	9.4
Number of storage shelves	2	2
Number of drives per shelf	30	24
Drive vendor and model number	Hitachi H4SMR321CLAR1600	[Vendor A] X371_S163A960ATE
Drive size	1.6 TB	960 GB

Server configuration information	12 x Dell EMC PowerEdge FX2 FC630 servers
BIOS name and version	2.8.0
Operating system name and version/build number	Microsoft® Windows Server® 2016
Date of last OS updates/patches applied	12/1/2018
Power management policy	Performance
Processor	
Number of processors	2
Vendor and model	Intel® Xeon® E5-2698 v4
Core count (per processor)	20
Core frequency (GHz)	2.20
Memory module(s)	
Total memory in system (GB)	128
Number of memory modules	4
Vendor and model	Samsung® M393A4K40BB1-CRC
Size (GB)	32
Type	PC4-19200
Speed (MHz)	2,400
Speed running in the server (MHz)	2,400
Storage controller	
Vendor and model	Dell EMC PERC S130
Firmware version	4.3.0-0002
Local storage	
Number of drives	2
Drive vendor and model	Intel SSDSA2CW600G3
Drive size (GB)	600
Drive information (speed, interface, type)	3Gb SATA, SSD

Server configuration information		12 x Dell EMC PowerEdge FX2 FC630 servers
Network adapter		
Vendor and model	QLogic BCM57810 10 Gigabit Ethernet (NDIS VBD Client)	
Number and type of ports	2 x 10GbE	
Driver version	7.13.104.0	
Enclosure		
Blade enclosure	3x PowerEdge FX2s	
Cooling fans		
Number of cooling fans	8	
Power supplies #1		
Vendor and model	Dell EMC	
Number of power supplies	2	
Wattage of each (W)	1,600	
Power supplies #2		
Vendor and model	Dell EMC DD1100E-S0	
Number of power supplies	2	
Wattage of each (W)	1,100	

The following table details the servers we used in our TensorFlow testing.

Server configuration information		4 x Dell PowerEdge R740
BIOS name and version	1.6.13	
Operating system name and version/build number	Ubuntu 18.04 LTS	
Date of last OS updates/patches applied	04/25/19	
Power management policy	Performance	
Processor		
Number of processors	2	
Vendor and model	Intel Xeon Platinum 8168	
Core count (per processor)	24	
Core frequency (GHz)	2.7	
Stepping	4	
Memory module(s)		
Total memory in system (GB)	256	
Number of memory modules	8	
Vendor and model	Samsung M393A4K40CB2-CTD	
Size (GB)	32	
Type	DDR4 PC4-21300	
Speed (MHz)	2,666	

Server configuration information	4 x Dell PowerEdge R740
Graphics Processing Unit (GPU)	
Vendor and model	NVIDIA Tesla V100-PCIE-16GB
Number of GPUs	2
Driver version	418.40.04
CUDA version	10.1
Storage controller	
Vendor and model	Dell PERC H740P
Cache size	8GB
Firmware version	50.5.0-1750
Local storage	
Number of drives	2
Drive vendor and model	Toshiba THNSF8120CCSE
Drive size (GB)	120
Drive information (speed, interface, type)	6Gbps, SSD, SATA
Network adapter	
Vendor and model	Mellanox ConnectX-3 (CX324A)
Number and type of ports	2x 40GbE
Firmware	02.42.50.00
Driver version	OFED 4.5-1.0.1.0
Cooling fans	
Vendor and model	Dell 04VXP3
Number of cooling fans	6
Power supplies	
Vendor and model	Dell 450-AEBL
Number of power supplies	2
Wattage of each (W)	1,100

## Appendix C: How we tested

### Configuring the Dell EMC Isilon F800 cluster for video performance

Following the guidelines in the Isilon filename-based prefetch [white paper](#), we configured Isilon for optimum sequential read-write performance of uncompressed video files.

1. Log into Isilon as root.
2. Enter the command: `sysctl -e isi.access.streaming > /tmp/custom_access.txt`
3. Edit `custom_access.txt`, and change the values as follows:

```
isi.access.custom1.read_batch=0
isi.access.custom1.prefetch.read_cluster_limit=256
isi.access.custom1.prefetch.aread=0
isi.access.custom1.prefetch.offload=1
isi.access.custom1.prefetch.adaptive=0
isi.access.custom1.prefetch.l1_window_blocks=2048
isi.access.custom1.prefetch.l1_cluster_blocks=256
isi.access.custom1.prefetch.l2_window_blocks=32768
isi.access.custom1.prefetch.l2_cluster_blocks=8192
isi.access.custom1.prefetch.l2_ssd_disable=0
isi.access.custom1.prefetch.l2_by_run=1
isi.access.custom1.prefetch.l2_unconditional=1
isi.access.custom1.prefetch.l2_batch=1
isi.access.custom1.prefetch.prefetch_by_cluster=1
isi.access.custom1.prefetch.meta_pgs=8
isi.access.custom1.coalescer.cregion_min_size=2097152
isi.access.custom1.coalescer.cregion_max_forced=16
isi.access.custom1.coalescer.coal_falloc=1
isi.access.custom1.fnprefetch.fn_enabled=1
isi.access.custom1.fnprefetch.lookahead=9
isi.access.custom1.fnprefetch.l1_all=1
isi.access.custom1.fnprefetch.size_limit=67108864
isi.access.custom1.write.realloc_blocks=0
isi.access.custom1.write.writealloc_blocks=0
isi.access.custom1.write.prealloc=1
```
4. Run `cat /tmp/custom_access.txt >> /etc/mcp/override/sysctl.conf` to copy the file to the settings override directory.
5. Run `sysctl isi.access.custom1` to confirm the settings are active.
6. Run `isi set -R -l streaming /ifs/<path to test data>` to apply the streaming layout template.
7. Run `isi set -R -a custom1 /ifs/<path to test data>` to apply the streaming access pattern template.
8. Run `isi get /ifs/<path to test data>` to confirm the settings have been applied.

### Acquiring performance data with Frametest

Frametest is a cross-platform utility for evaluating a filesystem's ability to handle streaming data, such as a sequence of frames that would be generated by video capture or editing software. The software can be configured to test for a variety of frame sizes, durations, and read/write patterns: <https://support.dvsus.com/hc/en-us/articles/212925466-How-to-use-frametest>.

1. Install and update Windows Server 2016 on 12 hosts.
2. Configure two 10Gb NICs per host, each with a static IP on the storage network.
3. Enable jumbo frames/MTU 9000 on all NICs.
4. Copy `frametest.exe` to each host's `C:\VidTest` directory.
5. Create a test matrix to check both performance per-node, as well as the entire storage cluster. Map network drives to allow each host to communicate with an individual IP on the storage node, instead of the load-balanced IP, to control how the load is distributed.
6. On Isilon, run `isi flush` on each node. On storage systems without a built-in flush command, use `iometer` to target nodes individually with each host, and write repeating data for 15 minutes to level the cache before the next test.
7. On each host, simultaneously run the following command to write 4K frames to the storage under test.

```
frametest.exe -w4k -n6000 -t20 -p15 -x<filename>.csv <destination file path>
```
8. Once this test is complete, connect to the storage and flush the caches.
9. On each host, simultaneously run the following command to read the 4K frames that were written to the storage under test.

```
frametest.exe -r -n6000 -t20 -p15 -x<filename>.csv <destination file path>
```
10. Repeat steps 6 through 9 for each data point in the test matrix, and tabulate the results.

## Running machine learning tests on the F810 with TensorFlow

For the Tensorflow testing, we configured four Dell EMC PowerEdge R740 servers each with two NVIDIA Tesla V100 GPUs. We installed Ubuntu 18.04 on each server, and installed NVIDIA Docker containers on each server to run the workload. We created a 137GB dataset and copied it out 100 times to get ~13.7TB. We ran three of the most popular Tensorflow models against the data set—resnet50, googlenet, and overfeat—and recorded the images/second when the test finished.

### Updating and configuring the operating system

After setting the BIOS settings to Max Performance, we installed Ubuntu 18.04 LTS onto each server. We updated the operating system and installed prerequisites for the NVIDIA Docker container. We performed the following steps on each server:

1. Log into the operating system as the root user.
2. Update the operating system, and reboot:

```
sudo apt update
sudo apt upgrade -y
sudo reboot
Install the NVIDIA driver:
sudo apt-get install -y apt-transport-https curl
cat <<EOF | sudo tee /etc/apt/sources.list.d/cuda.list > /dev/null
deb https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 /
EOF
curl -s \
https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64/7fa2af80.pub \
| sudo apt-key add -

cat <<EOF | sudo tee /etc/apt/preferences.d/cuda > /dev/null
Package: *
Pin: origin developer.download.nvidia.com
Pin-Priority: 600
EOF
sudo apt-get update && sudo apt-get install -y --no-install-recommends cuda-drivers
sudo reboot
```

3. Verify that the NVIDIA drivers installed correctly:

```
nvidia-smi
```

### Installing the NVIDIA Docker container

1. Install Docker:

```
sudo apt-get install -y ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release
-csstable)"
```

2. Install NVIDIA Docker:

```
curl -s -L https://nvidia.github.io/nvidia-docker/gpgkey | \
sudo apt-key add -
distribution=$(. /etc/os-release;echo $ID$VERSION_ID)
curl -s -L https://nvidia.github.io/nvidia-docker/$distribution/nvidia-docker.list | \
sudo tee /etc/apt/sources.list.d/nvidia-docker.list
sudo apt-get update
sudo apt-get install -y nvidia-docker2
sudo usermod -aG docker $USER
sudo reboot
```

3. Pull and run the TensorFlow image:

```
sudo nvidia-docker create --name=${CONTAINER_NAME} --shm-size=1g --ulimit memlock=-1 --privileged -v
${DATA_DIR}:${DATA_DIR} -v ${TRAIN_DIR}:${TRAIN_DIR} -i nvcr.io/
nvidia/tensorflow:19.03-py3
```

## Installing and running the benchmark

1. We pulled the benchmark `tf_cnn_benchmark` from the TensorFlow Github repo. We adjusted the run length and sample size to ensure steady-state GPU temperature. To make these changes, run the following commands inside the NVIDIA Docker container on each server:

2. Pull `tf_cnn_benchmark`:

```
cd /tensorflow
git clone https://github.com/tensorflow/benchmarks.git -b cnn_tf_v1.13_compatible --single-branch
```

3. Initialize Tensorflow benchmarks:

```
nvidia-docker exec ${CONTAINER_NAME} ${PYTHON} -u /tensorflow/benchmarks/scripts/tf_cnn_benchmarks/
tf_cnn_benchmarks.py --model=resnet50 --data_name=imagenet
nvidia-docker exec ${CONTAINER_NAME} ${PYTHON} -u /tensorflow/benchmarks/scripts/tf_cnn_benchmarks/
tf_cnn_benchmarks.py --model=resnet56 --data_name=cifar10
```

4. Run the benchmarks:

- a. All tests:

```
nvidia-docker exec ${CONTAINER_NAME} ${AFFINITY_CMD} ${PYTHON} -u /tensorflow/benchmarks/scripts/tf_
cnn_benchmarks/tf_cnn_benchmarks.py ${TS_PARAMS}
```

- b. Switching between FP16 and FP32:

```
FP16: --use_fp16=True
```

```
FP32: --use_fp16=False
```

## Mounting NFS storage and preparing imagenet data

After preparing the benchmark, we mounted the NFS storage and prepared the imagenet dataset. Once we created the initial dataset, we copied and renamed it 100 times to increase the total dataset size from 137GB to ~13.7TB.

1. Add the following to the `/etc/fstab` file to the NFS storage (adjust the IP address and mount name as needed for your network and storage NAS setup):

```
192.168.42.21:/ifs/tf/data /tf/data nfs rw,relatime,vers=3,rsize=1048576,wsiz=1048576,
namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,mountvers=3,mountpo
rt=300,mountproto=tcp,local_lock=none,_netdev 0 0
```

2. Download and build the imagenet dataset as a single shard (this could take several hours to complete):

```
NAME=tf_test
PYTHON=python3
DATA_DIR=/tf/data
MODELS_BRANCH=r1.13.0
IMAGENET_DIR=${DATA_DIR}/imagenet
TF_USERNAME=
TF_PASSWORD=
TRAIN_SHARDS=1
VALIDATION_SHARDS=1
NUM_THREADS=1
nvidia-docker stop ${NAME}
echo "cd ${DATA_DIR} && rm -rf models && git clone https://github.com/tensorflow/models.git -b
${MODELS_BRANCH} --single-branch" | nvidia-docker start -i ${NAME}
}

echo "cd ${DATA_DIR}/models/research/inception && sed -i 's/wget "/wget -nd -c "/" inception/data/
download_imagenet.sh && sed -i '/\${BUILD_SCRIPT}/a --train_sh
ards=${TRAIN_SHARDS} --validation_shards=${VALIDATION_SHARDS} --num_threads=${NUM_THREADS} \\\\'
inception/data/download_and_preprocess_imagenet.sh && bazel buil
d //inception:download_and_preprocess_imagenet && IMAGENET_USERNAME=${TF_USERNAME} IMAGENET_ACCESS_
KEY=${TF_PASSWORD} bazel-bin/inception/download_and_preproces
s_imagenet ${IMAGENET_DIR}" | nvidia-docker start -i ${NAME}
```

3. Duplicate the dataset shard 100 times and rename accordingly. Example:

```
cp imagenet_1x/train-00000-of-00001 imagenet_100x/train-00000-of-00099
cp imagenet_1x/train-00000-of-00001 imagenet_100x/train-00001-of-00099
```

## Server model tuning:

For each model, we varied `batch_size`, `variable_update`, and `all_reduce_spec`. We ran the varied settings against a fixed `num_batches` count for each model that varied from model to model. We added the sample syntax of the test execution below. Where we noted with `<variable>`, the requisite parameter for that configuration needs to be passed.

```
--num_gpus=2 --data_format=NCHW --model=<variable> --data_name=imagenet --data_dir=/tf/data/
imagenet_100x --use_fp16=True --batch_size=<variable> --num_batches=<variable> --op
timizer=sgd --distortions=False --variable_update=<variable> --local_parameter_device=<variable>
--all_reduce_spec=<variable>
```

Example:

```
--num_gpus=2 --data_format=NCHW --model=alexnet --data_name=imagenet --data_dir=/tf/data/
imagenet_100x --use_fp16=True --batch_size=2048 --num_batches=3000 --op
timizer=sgd --distortions=False --variable_update=replicated --local_parameter_device=gpu --all_
reduce_spec=nccl
```

## Storage efficiency testing on the Dell EMC Isilon F810

We used eight datastores (one per node on Isilon and two per node on array from competitor A) with default storage settings across both storage arrays. We used the open-source benchmarking tool Vdbench v5.04.07 as a disk I/O workload generator and employed VMware HCIBench v2.0 to deploy test VMs, coordinate workloads, and aggregate test results. We configured eight Linux test VMs with 8 x 200GB VMDKs per array.

### Deploying the HCIBench controller VM

We downloaded HCIBench v2.0 from <https://labs.vmware.com/flings/hcibench> and deployed it onto our infrastructure server. We also downloaded Vdbench 5.04.07 from <https://www.oracle.com/technetwork/server-storage/vdbench-downloads-1901681.html>.

1. Log into vCenter server.
2. Right-click Datacenter.
3. Click Deploy OVF Template.
4. Select Local file, and click Browse.
5. Select HCIBench\_2.0.ova, and click Open.
6. Click Next.
7. At the Select name and location screen, click Next.
8. Select the infrastructure host, and click Next.
9. At the Review details screen, click Next.
10. At the Accept license agreement screen, click Accept twice, and click Next.
11. At the Select storage screen, choose a datastore to host the appliance, and click Next.
12. At the Select network screen, select the VM and Management networks, and click Next.
13. At the Customize template screen, enter Network Settings and Root credentials, and click Next.
14. At the Ready to complete screen, click Finish.

### Configuring HCIBench

1. Navigate to `http://HCIBench_IP:8443/` and log in using root credentials.
2. Provide the following vSphere environment information:
  - vCenter hostname or IP
  - vCenter username and password
  - Datacenter name
  - Cluster name
  - Network name
  - Datastore name
  - Hosts
  - Host username and password
3. Under Upload the Vdbench File, click Choose File, and select the `vdbench50407.zip` file.
4. Select Open, and click Upload Vdbench.
5. Click Save Configuration.

## Deploying Vdbench guest VMs and running the test

We deployed eight VMs with eight data disks each.

1. Navigate to [http://HCIBench\\_IP:8443/](http://HCIBench_IP:8443/) and log in using root credentials.
2. Provide the following guest VM specifications:
  - VM name prefix = TestVM
  - Number of VMs = 8
  - Number of data disks = 8
  - Size of data disks = 200GB
3. Under Upload a Vdbench parameter file, click Choose File.
4. Select the appropriate parameter file, and click Open.
5. Click Upload Parameter File.
6. Under Select a Vdbench parameter file, click Refresh.
7. Select the appropriate parameter file, and click Save Configuration.
8. Click Test.

## Prefilling VMDKs data

We used 256KB sequential writes to fill the VMDKs with data.

```
compratio=4
dedupunit=8k
dedupratio=3
dedupsets=5%
sd=sd1,lun=/dev/sda,openflags=o_direct
sd=sd2,lun=/dev/sdb,openflags=o_direct
sd=sd3,lun=/dev/sdc,openflags=o_direct
sd=sd4,lun=/dev/sdd,openflags=o_direct
sd=sd5,lun=/dev/sde,openflags=o_direct
sd=sd6,lun=/dev/sdf,openflags=o_direct
sd=sd7,lun=/dev/sdg,openflags=o_direct
sd=sd8,lun=/dev/sdh,openflags=o_direct
wd=wd_prefill,sd=*,xfersize=256k,rdpct=0,seekpct=eof
rd=prefill_1,wd=wd_prefill,iorate=max,interval=30,elapsed=100h,threads=1
```

## Testing data reduction performance

For these tests we performed the data reduction tests immediately after the data reduction prefill completed.

```
compratio=4
dedupunit=8k
dedupratio=3
dedupsets=5%
sd=sd1,lun=/dev/sda,openflags=o_direct
sd=sd2,lun=/dev/sdb,openflags=o_direct
sd=sd3,lun=/dev/sdc,openflags=o_direct
sd=sd4,lun=/dev/sdd,openflags=o_direct
sd=sd5,lun=/dev/sde,openflags=o_direct
sd=sd6,lun=/dev/sdf,openflags=o_direct
sd=sd7,lun=/dev/sdg,openflags=o_direct
sd=sd8,lun=/dev/sdh,openflags=o_direct
wd=s1w,sd=(sd1),xfersize=128k,seekpct=100,rdpct=0
rd=rand_w_1,wd=s1w,iorate=max,interval=10,elapsed=1h,warmup=60,threads=8
```

This project was commissioned by Dell EMC.



Facts matter.®

Principled Technologies is a registered trademark of Principled Technologies, Inc.  
All other product names are the trademarks of their respective owners.

**DISCLAIMER OF WARRANTIES; LIMITATION OF LIABILITY:**

Principled Technologies, Inc. has made reasonable efforts to ensure the accuracy and validity of its testing, however, Principled Technologies, Inc. specifically disclaims any warranty, expressed or implied, relating to the test results and analysis, their accuracy, completeness or quality, including any implied warranty of fitness for any particular purpose. All persons or entities relying on the results of any testing do so at their own risk, and agree that Principled Technologies, Inc., its employees and its subcontractors shall have no liability whatsoever from any claim of loss or damage on account of any alleged error or defect in any testing procedure or result.

In no event shall Principled Technologies, Inc. be liable for indirect, special, incidental, or consequential damages in connection with its testing, even if advised of the possibility of such damages. In no event shall Principled Technologies, Inc.'s liability, including for direct damages, exceed the amounts paid in connection with Principled Technologies, Inc.'s testing. Customer's sole and exclusive remedies are as set forth herein.