Technical White Paper

Dell EMC Isilon: Using Isilon F810 with SAS Analytics for Financial Services

Abstract

This document demonstrates how the Dell EMC™ Isilon™ F810 all-flash scale-out NAS can be used to accelerate SAS® Grid workloads. Runtime results for various SAS jobs are also included.

June 2020

H18334

Revisions

Date	Description
June 2020	Initial release

Acknowledgments

Authors:

- Boni Bruno, Dell Technologies
- Tom Keefer, D4t4 Solutions

This document may contain certain words that are not consistent with Dell's current language guidelines. Dell plans to update the document over subsequent future releases to revise these words accordingly.

This document may contain language from third party content that is not under Dell's control and is not consistent with Dell's current guidelines for Dell's own content. When such third party content is updated by the relevant third parties, this document will be revised accordingly.

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Table of contents

Executive summary

This document details how the Dell EMC™ Isilon™ F810 all-flash scale-out NAS system is tested for performance and scalability for SAS® Grid workloads. The Isilon F810 system provides enterprise-level storage performance and data compression for SAS. This document provides performance test results produced by Dell Technologies™ and D4t4 Solutions. The results show that the F810 system is an ideal storage solution to store and serve SAS data.

Audience

This document is intended for organizations that are interested in simplifying and accelerating SAS Grid workloads with advanced computing and scale-out data-management solutions. The target audience includes solution architects, system administrators, and other readers within those organizations.

1 Introduction

In the financial services industry, volumes of data continue to grow in support of analytics. It is a strategic advantage and a regulatory requirement to run extensive analysis of large data volumes in support of day-to-day business operations and reporting. Due to these data volumes, it is a requirement to have a dynamic, shareable, scalable, high-performance, and cost-effective storage system. This paper evaluates using Dell EMC Isilon F810 storage with SAS analytics, a common tool used in financial services for analytics and reporting.

1.1 SAS analytics in the financial industry

SAS software is a commonly used analytics tool in financial services to analyze, forecast, mine, model, and report on customer and market-condition data. Despite increases in processing power and a move to analyze more data in memory, data storage is still a key component of analytics. Financial statisticians and modelers are running more iterations of programs against growing data volumes. A typical SAS financial analytics environment supports hundreds to thousands of users running simultaneous programs on a collection of servers called a grid. To support these increasing requirements in large SAS environments, it is critical that storage products are highly scalable and support growing data volumes without a massive price increase. This paper evaluates the results of a SAS financial workload run against Isilon F810 storage, highlights the solution benefits, and provides best practices for optimal use.

1.2 Isilon for SAS analytics

Isilon network-attached storage (NAS) systems deliver breakthrough ease-of-scale, ease-of-administration, and scalable NAS performance using a clustered architecture. Powered by the Isilon OneFS™ operating system, Isilon clusters provide multiprotocol access to large volumes of data over NFS, SMB, HTTP, FTP, and HDFS protocols. OneFS is developed to meet big-data market demands for easy-to-deploy scale-out capacity, performance, and operational simplicity. Some key markets include media and entertainment, healthcare, life sciences, high performance computing (HPC), and the burgeoning data analytics market. OneFS continues to extend its reach into numerous new markets thanks to continuous refinements to Isilon hardware and OneFS software and a growing set of enterprise data-management features.

2 Isilon storage for SAS GRID

The Isilon F800 series represents the sixth generation of hardware built to run the proven and massively scalable OneFS operating system. Each F810 chassis, shown in Figure 1, contains four storage nodes, 60 high-performance solid-state drives (SSDs), and eight 40 Gb Ethernet network connections. OneFS combines up to 252 nodes in 63 chassis into a single high-performance file system that is designed to handle the most intense SAS I/O workloads. As performance and capacity demands increase, both can be scaled out nondisruptively, allowing applications and users to continue working.

Figure 1 Isilon F810 chassis containing four internal storage nodes

The Isilon F810 system has the following features:

- Low latency, high throughput, and massively parallel I/O for SAS GRID.
  - Up to 250,000 file IOPS per chassis, and up to 15.75 million IOPS per cluster
  - Up to 15 GB/s throughput per chassis, and up to 945 GB/s per cluster
  - 96 TB to 924 TB raw flash capacity per chassis, and up to 58 PB per cluster (all-flash)
- Integrated enterprise-grade features.
  - Enterprise data protection and resiliency
  - Robust security options

  This enables organizations to manage SAS data life cycle with minimal cost and risk, while protecting data and meeting regulatory requirements.
- Extreme scale:
  - Seamless tiering between all flash, hybrid, and archive nodes through SmartPools
  - Grow-as-you-go scalability with up to 58 PB flash capacity per cluster
  - Easy addition of new nodes to a cluster by connecting power, back-end Ethernet, and front-end Ethernet
  - Ability to grow storage capacity, throughput, IOPS, cache, and CPU as new nodes are added
  - Support for connecting up to 63 chassis (252 nodes) to form a single cluster with a single namespace and a single coherent cache
  - Up to 85% storage efficiency to reduce costs
  - Data deduplication and compression enabling up to a 3:1 data reduction

Organizations can achieve analytics at scale in a cost-effective manner, enabling them to handle multi-petabyte datasets with high-resolution content without rearchitecture or performance degradation.

There are several key features of Isilon OneFS that make it an excellent storage system for SAS workloads that require performance, concurrency, and scale. These features are detailed in the following subsections.

2.1 Storage tiering

Isilon SmartPools software enables multiple levels of performance, protection, and storage density to co-exist within the same file system. It unlocks the ability to aggregate and consolidate a wide range of applications within a single extensible, ubiquitous storage resource pool. This helps provide granular performance optimization, workflow isolation, higher utilization, and independent scalability—all with a single point of management.

SmartPools allows you to define the value of the data within your workflows based on policies and automatically aligns data to the appropriate price-to-performance tier over time. Data movement is seamless, and with file-level granularity and control through automated policies, manual control, or API interface, you can tune performance and layout, storage-tier alignment, and protection settings—with minimal impact to end users.

Storage tiering places data according to its business value and aligns it with the appropriate class of storage and levels of performance and protection. Information life-cycle management techniques have been around for several years but have typically suffered from the following inefficiencies: complex to install and manage, involves changes to the file system, requires the use of stub files, and other limitations.

Isilon SmartPools is a next-generation approach to tiering that facilitates the management of heterogeneous clusters. The SmartPools capability is native to the Isilon OneFS scale-out file system, which allows for unprecedented flexibility, granularity, and ease of management. To achieve this, SmartPools uses many of the components and attributes of OneFS, including data layout and mobility, protection, performance, scheduling, and impact management.

A typical Isilon cluster stores multiple datasets with different performance, protection, and price requirements. Generally, files that were recently created and accessed are stored in a hot tier, while files that have not been accessed recently are stored in a cold (or colder) tier. Because Isilon supports tiering based on file access time, this can be performed automatically. For storage administrators that want more control, complex rules can be defined to set the storage tier based on file path, size, or other attributes.

All files on Isilon storage are always immediately accessible (read and write) regardless of their storage tier and even while being moved between tiers. The file-system path to a file is not changed by tiering. Storage tiering policies are applied, and files are moved by the Isilon SmartPools job, which runs daily at 22:00 by default.

For more details, see the document Storage Tiering with Dell EMC Isilon SmartPools.

2.2 OneFS caching

The OneFS caching infrastructure design is predicated on aggregating the cache present on each node in a cluster into one globally accessible pool of memory. This allows all the memory cache in a node to be available to every node in the cluster. Remote memory is accessed over an internal interconnect and has much lower latency than accessing hard disk drives.

The OneFS caching subsystem is coherent across the cluster. This means that if the same content exists in the private caches of multiple nodes, this cached data is consistent across all instances.

OneFS uses up to three levels of read cache, plus an NVRAM-backed write cache, or coalescer. These levels and their interaction with each other are illustrated at a high level in Figure 2.

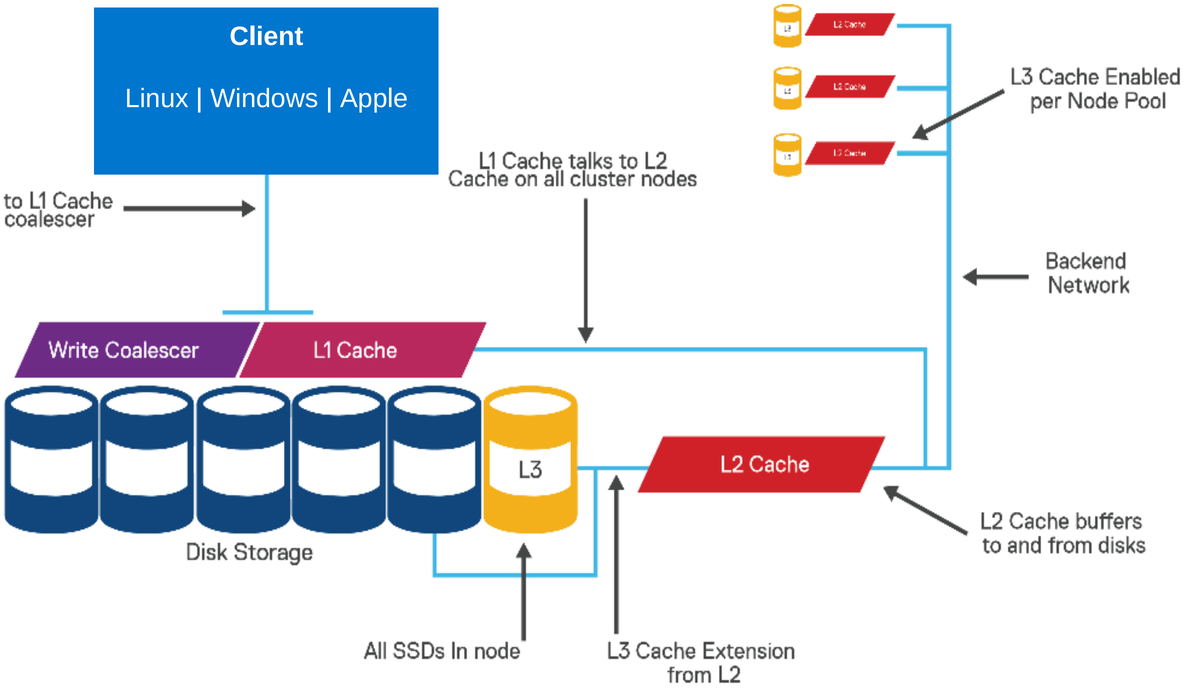

Client
Linux | Windows | Apple
Clients talk to L1 Cache and write coalescer
L1 Cache talks to L2 Cache on all cluster nodes
L3
L2 Cache
L3
L2 Cache
L3
L2 Cache
L3 Cache Enabled per Node Pool
Backend Network
Write Coalescer
L1 Cache
L3
L2 Cache
Disk Storage
L2 Cache buffers to and from disks
All SSDs In node
L3 Cache Extension from L2

Figure 2 OneFS caching architecture

2.3 File reads

For files marked with an access pattern of concurrent or streaming, OneFS can take advantage of data prefetching that is based on heuristics used by the Isilon SmartRead component. SmartRead can create a data pipeline from the L2 cache, prefetching into a local L1 cache on the captain node. This greatly improves sequential-read performance across all protocols and means that reads come directly from RAM within milliseconds. For high-sequential cases, SmartRead can aggressively prefetch ahead, allowing reads of individual files at high data rates.

Intelligent caching provided by SmartRead allows for high read performance with high levels of concurrent access. It is faster for node 1 to get file data from the cache of node 2 (over the low-latency cluster interconnect) than to access its own local disk. The SmartRead algorithms control how aggressive the prefetching is (disabling prefetch for random-access cases) and how long data stays in the cache, and it optimizes where data is cached. This optimized file read logic is visualized in Figure 3.

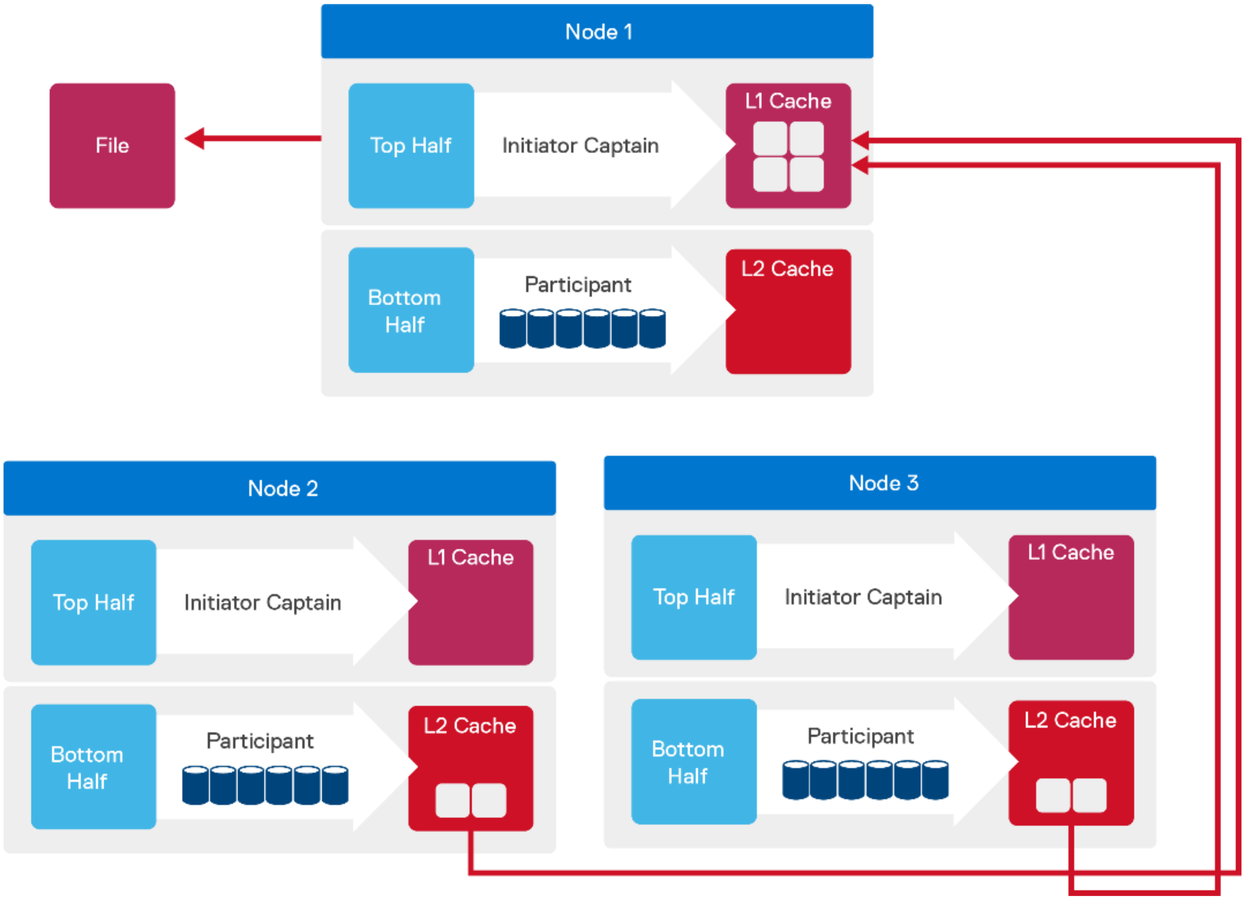


Figure 3 A file read operation on a three-node Isilon cluster

2.4 Locks and concurrency

OneFS has a fully distributed lock manager that coordinates locks on data across all nodes in a storage cluster. The lock manager is highly extensible and allows for multiple lock personalities. It supports both file-system locks and cluster-coherent protocol-level locks such as SMB share mode locks or NFS advisory-mode locks. OneFS also supports delegated locks such as CIFS oplocks and NFSv4 delegations. Every node in a cluster is a coordinator for locking resources, and a coordinator is assigned to lockable resources based on an advanced hashing algorithm. For more details, see the OneFS Technical Overview.

Efficient locking is critical to support the efficient parallel I/O profile that is demanded by many iterative SAS workloads enabling up to millions of concurrent file reads.

3 SAS Grid architecture

SAS Grid computing enables organizations to create a managed, shared environment to process large volumes of data and analytic programs more efficiently. In the SAS Grid environment, the SAS computing tasks are distributed among multiple nodes on a network, and jobs are processed in a distributed manner. Figure 4 shows a typical SAS Grid architecture. SAS resource management is used to support many users and programs while using shared resources like servers and storage.

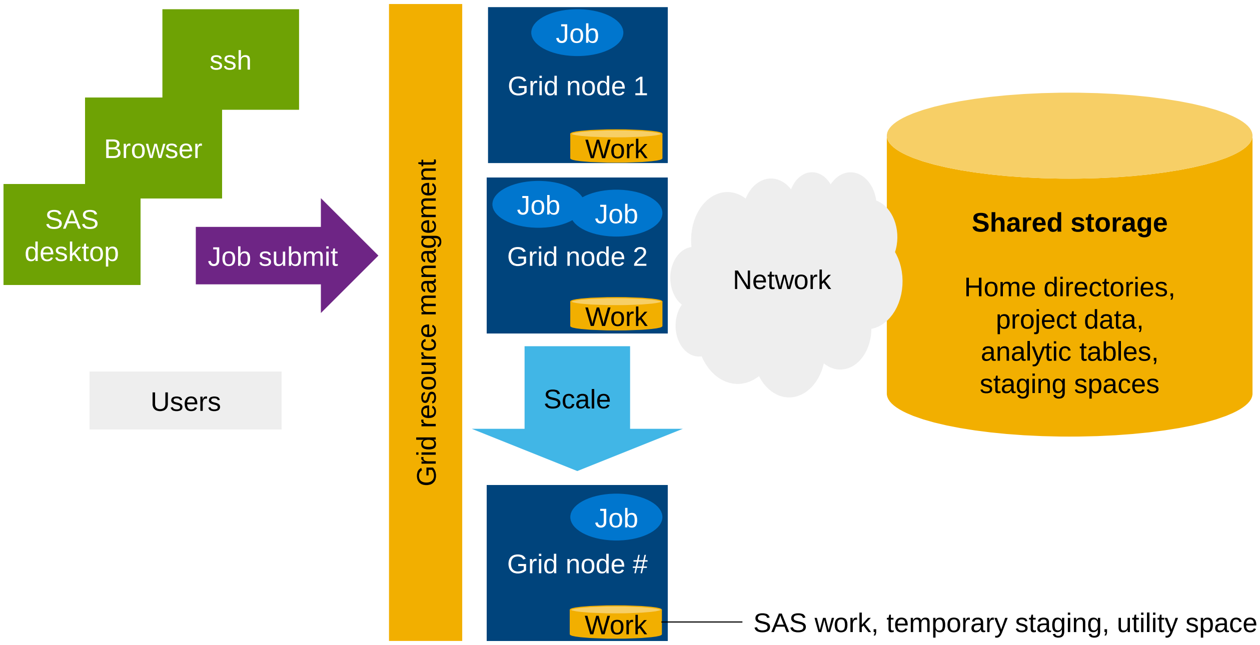


Figure 4 Typical SAS Grid architecture

4 How SAS and other analytic software tools use storage

SAS products, like most analytic solutions, use storage for archiving, temporary processing, caching, and general day-to-day project storage. Newer SAS products use storage for staging and lifting to memory for processing. However, most day-to-day analytics in the financial industry are still based on SAS 9, which is typically run on a SAS Grid environment that uses shared storage systems. Analytic systems process large amounts of data during transformation and processing. Keeping a sustained stream of data to and from CPUs is the most critical capability an analytic system needs to provide.

Besides providing performance, an analytic storage system must also provide scalability, compatibility with many analytic tools, security, high compression rates to reduce cost, and ease of management. This document focuses on the testing for performance, scalability, and compression benefits.

SAS analytics read and write data from many data sources. The three most heavily used I/O devices for SAS Grid are as follows:

- SASWORK: High-speed disk storage (temporary processing space)
- Network: RDBMS or other storage spaces like Apache® Hadoop® and S3
- Permanent file storage: SAS datasets, CSV files, and other file types (file systems, shared or local)
- Memory: Data being read or written to memory-based storage systems

Of the three potential I/O sources, SAS file systems require a shared file system like the one provided by the Isilon F810 storage array.

Note: NFS-based storage systems should never be used in production for SASWORK due to the significant throughput requirements (GBps for a single server).

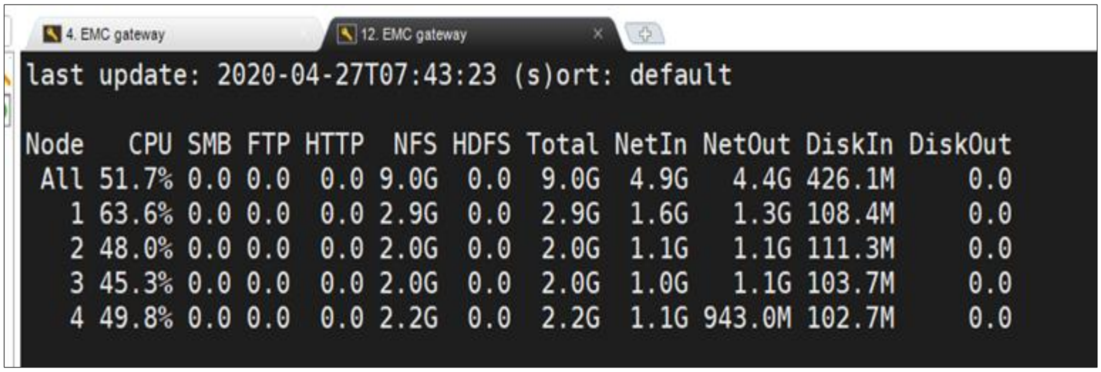
As SAS programs begin to run, they typically read data from a network (RDBMS) or storage location on disk (typically a project folder) to start processing. Combinations of reading and writing from any of the data devices or locations listed above is common in any SAS job. On average, it is common to have 40–60% of the overall I/O reads and writes go to or from SASWORK (temporary storage). SASWORK is a focal point for SAS processing because it is the common location where data is merged, processed, and assembled in support of analytics. The remaining I/O is spread across the other devices, but most of it typically goes to the SAS file systems (shared or permanent storage).

SAS has done extensive research around storage and the I/O throughput required to feed SAS. A commonly referenced paper from these efforts is the document Best Practices for Configuring Your I/O Subsystem for SAS9 Application*.* In the paper, SAS states the overall I/O of an analytic system must able to sustain 100–150 MBps I/O (mixed read/write) per CPU to properly feed CPU cores running SAS analytics.

Based on SAS requirements and the trend that 40–50% of I/O goes to and from shared storage, that storage system needs to provide 60–75 or more MBps I/O (read and write) per CPU core throughput to support a SAS Grid environment. To put this in perspective, a typical NFS supported grid node (single server in a larger multinode SAS Grid) has 8 to 16 CPU cores. With the 40–50% requirement, 8- to 16-core grid nodes would need to read and write at a sustained rate of 480–900 MBps to shared (NFS) storage. This requirement assumes that the other 50–60% of the I/O would be read or written from SASWORK and other data sources (RDBMS, Hadoop, or memory).

In summary, a shared storage system supporting SAS needs to sustain an average of 60–75 MBps for every CPU core or there may be I/O performance issues such as high I/O wait times.

As a precursor to testing the Isilon F810 system, we ran a simple device-to-device (DD) copy test with a set of scripts that help simulate I/O like SAS (reads and writes to storage with 128k or 256k block sizes). During a test that used all our SAS servers, we could drive the storage to a sustained I/O rate of 9 GBps while still having a large amount of Isilon CPU available. See Figure 5. In this image, the array is sustaining 9 GBps across the four F810 nodes in the four-node chassis used in the test. With a sustained rate of 9 GBps and an I/O requirement from NFS of 60–75 GBps per CPU core, it means we could support 122–153 CPU cores for SAS. Our target large-scale test is to simulate 144 cores (or 12 servers with 12 cores each).



```
last update: 2020-04-27T07:43:23 (s)ort: default

Node   CPU SMB FTP HTTP  NFS HDFS Total NetIn NetOut DiskIn DiskOut
 All 51.7% 0.0 0.0  0.0 9.0G  0.0  9.0G  4.9G   4.4G 426.1M     0.0
   1 63.6% 0.0 0.0  0.0 2.9G  0.0  2.9G  1.6G   1.3G 108.4M     0.0
   2 48.0% 0.0 0.0  0.0 2.0G  0.0  2.0G  1.1G   1.1G 111.3M     0.0
   3 45.3% 0.0 0.0  0.0 2.0G  0.0  2.0G  1.0G   1.1G 103.7M     0.0
   4 49.8% 0.0 0.0  0.0 2.2G  0.0  2.2G  1.1G 943.0M 102.7M     0.0
```

Figure 5 Statistics on the Isilon array showing total input and output throughput through NFS

5 Multiuser analytic workload

There are many different tools to test I/O performance of storage subsystems. However, due to the dynamic nature of analytic systems, it makes better sense to test with a tool that can replicate the patterns and data that are typical in a financial analytics system.

5.1 Description of test scripts

The multiuser analytic workload was written to run a workload (programs and data) like that found in a financial services SAS Grid. It is built to stress the system of a typical SAS financial services customer (such as a large bank with many modelers, statisticians, and reporting users). This test is built as a joint effort between former SAS employees and financial services analytics team members.

The multiuser workload can be run on a single SMP system or a multinode SAS Grid environment. It can be modified quickly to ramp the workload up and down to stress a system's CPU, RAM, and I/O capability based on its performance potential (size). SAS I/O, being the most critical component of any customer's SAS environment, is one of the prime focuses of the scenario.

SAS programs in the workload include data and functions that simulate the following SAS financial user personas:

- SAS studio or report user with an interactive report or a coding user (sleep periods are added to create the feel of real users working on the system at random periods)
- SAS modeler that runs complex analytics like logistic or regression
- SAS dataset construction in support of modeling or analytics (building analytics datasets)
- ETL workflow simulation reading from remote source and populating tables (includes index creation, merge, where, sorts)
- Advanced analytics user with larger datasets with more advanced analytics and data manipulation

The above jobs are simultaneous executions of jobs that are launched in a timed launch sequence to simulate users coming and going from the grid.

Examples of the SAS PROCEDURES and methods used in the programs in the test scenario are as follows:

- Data Step
- PRINT
- MEANS
- CONTENTS
- SQL
- HPLOGISTIC
- SORT
- REG
- GLM
- DELETE
- DATASETS

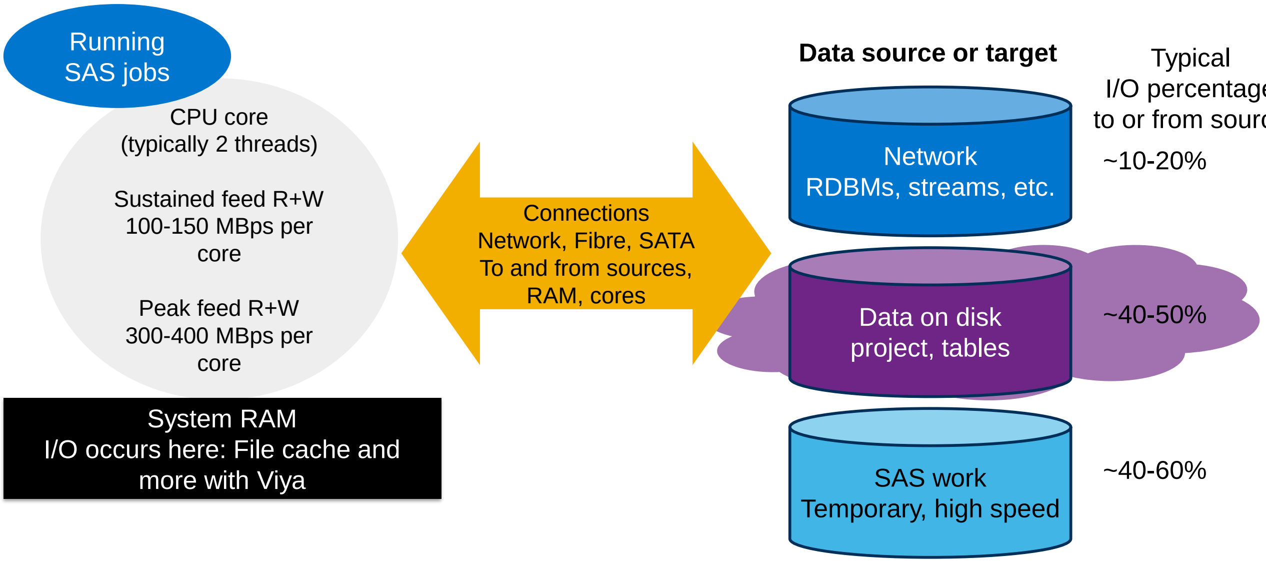
5.2 Test execution philosophy

It is common to run this test scenario with different mixes of users (SAS jobs) to resemble a customer's environment. The workload is not designed as a publishable benchmark like TPC or SPEC where the results are always the same and the test is run in exactly the prescribed fashion for later comparison. It is primarily meant to stress the system, especially related to its I/O capability, to confirm its ability to achieve the recommended SAS requirements. The test is tuned up and down to ensure that under a multiuser workload throughput can be maintained across the server or servers. In financial institutions, data volumes can be significantly larger than those used in the test scenario. Since runtimes for larger data volumes can be several hours in some cases, a compromise between file size and run time is used. Sizes of data are chosen to ensure file cache and I/O levels match those seen in banking environments but without the extended runtimes, allowing many more test runs during this evaluation.

5.3 Primary goal: Meet or exceed SAS I/O requirements

As mentioned earlier in the paper, SAS requires a system to be able to sustain an I/O rate of 150 MBps per CPU core for every core in the SAS Grid node as shown in Figure 6. This means that the total I/O (read and write) to temporary (SASWORK) and storage devices or locations like RDBMS, SAN, or NAS storage devices must be able to sustain 150 MBps per CPU core at any time. Fifty percent or more of the SAS Grid I/O throughput is typically to SASWORK, while the other 40–50% comes from permanent stores like NFS. Therefore, if we use the F810 as the other datastore, NFS would need to maintain a throughput of 60–75 MBps per CPU core. The larger the SAS compute server is, the more I/O you need to provide. In a SAS Grid environment, each Grid node needs to have an I/O throughput capability to support its CPUs while the other Grid nodes are also accessing the shared storage. This means any shared file system must be able to support the sum of the potential I/O requirement across **all** the grid nodes.

Figure 6 provides a high-level overview of where SAS I/O comes and goes to during execution of SAS programs on a server.


Running SAS jobs
CPU core (typically 2 threads)
Sustained feed R+W 100-150 MBps per core
Peak feed R+W 300-400 MBps per core
System RAM
I/O occurs here: File cache and more with Viya
Connections
Network, Fibre, SATA
To and from sources, RAM, cores
Data source or target
Typical I/O percentage to or from source
Network
RDBMs, streams, etc.
~10-20%
Data on disk
project, tables
~40-50%
SAS work
Temporary, high speed
~40-60%

SAS rule: Sustain I/O throughput of around 150 MBps Total (combined R+W) per core. Cores range in speed and performance, but this is a good target throughput.

Figure 6 SAS I/O requirements and data flow

5.4 Test execution details

The multiuser analytic workload consists of 33 SAS analytics jobs that simulate common activities found in a SAS financial services environment. The launch scripts used on each Grid node run a single batch of the same 33 jobs in a controlled time launch sequence on each server used in the test. Data for the jobs is pregenerated (SAS compressed or uncompressed) and duplicated on all the machines and placed on the shared file system. Each batch has its own input and output destinations on the Isilon storage.

In this test scenario, input data is placed on NFS (shared storage with Isilon F810 systems). A SASWORK local file system is created to handle 50–60+% of the I/O workload (local XFS file system on each grid node). An output data directory is also placed on the NFS file system because it is common to typically have more than one input/output directory per server. Read and write I/O operations are carried out on both NFS file system mounts (input and output folders on NFS). Figure 7 shows how the batches of the test workload are run in the lab environment.

After data generation, scripts are launched on each grid node that is participating in the test scenario. No data is shared between Grid nodes for this test, and each node operates on its own set of the data. During execution, it is typical to see 16 or more simultaneous SAS jobs running on each grid node during the test at a time. The number of jobs used and the configuration of the test is designed to stress a typical 8–16 core SAS Grid node with two 10 GbE connections to NAS/NFS.

Figure 7 shows how batches of the test workload are run on the lab hardware (Grid).

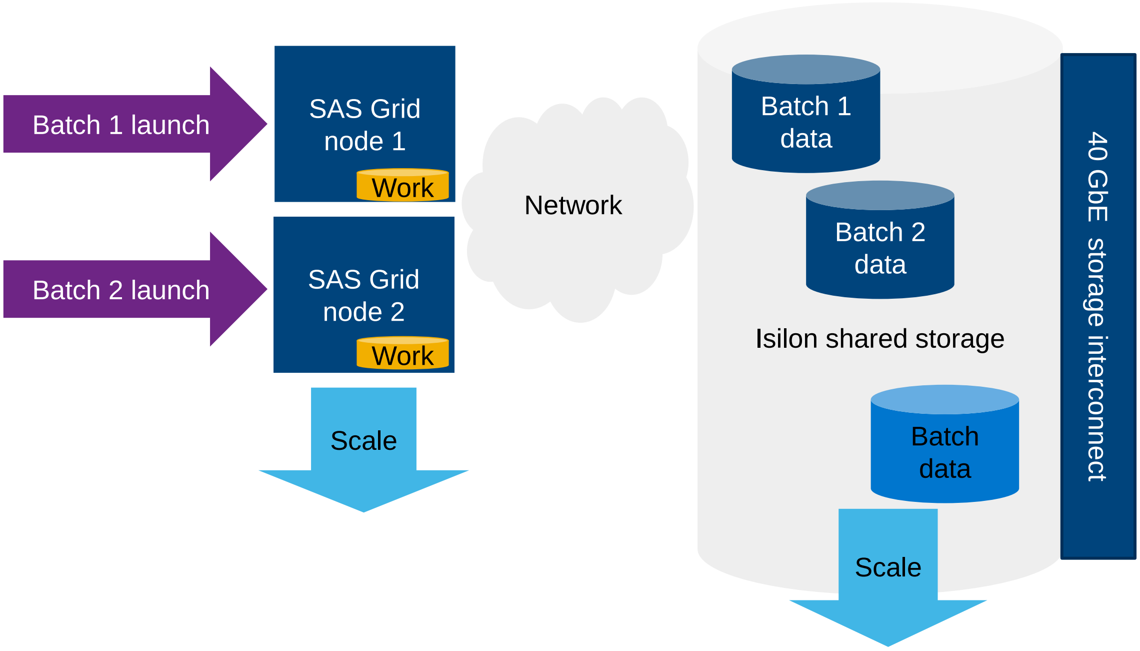


Figure 7 Multiuser analytics workload

6 Hardware configuration of test lab

To run the test workload, we use a set of 12 Linux® grid nodes to drive the SAS test scenarios against a four-node Isilon F800 chassis and a four-node Isilon F810 chassis. Each grid node is set up in the same way that a SAS Grid node would be configured for networking access to shared storage and local SASWORK. The nodes all have dual 10GbE network connections configured in a bonded mode to increase throughput for NFS. The Isilon nodes all have a single 40 GbE connection to the Ethernet where the grid nodes are also connected. Each grid node also uses a RAID-0 stripe composed of 22 internal disks for SASWORK. Throughput of SASWORK must exceed the overall I/O capability of the dual 10 GbE access to the NFS file systems for testing since SASWORK typically has a larger I/O throughput requirement for SAS systems. In this case, the system is able to achieve over two times the I/O throughput of the NFS access, which is suitable for testing efforts because it must meet or exceed SASWORK in a typical grid node. The workload is designed to simulate only 12 of the CPU cores in the server, even though all grid nodes have more cores than required to drive the workload to the network/NFS. The goal of this test is not to stress the grid machines. The goal is to produce enough workload to simulate the I/O load of a standard 12-core grid server using dual 10 GbE connections to NFS storage (a common configuration for SAS Grid environments using NFS).

Details of test configuration:

- 12 Dell EMC PowerEdge™ R730 servers:
  - Centos 7.7
  - SASWORK configuration: 22 disks, RAID 0, internal disks
  - Intel® Xeon® CPU, 2.2 GHz (2 x 20 cores, 80 threads)
  - 256 GB RAM
  - 2 x 10 GbE with LACP (bonded configuration)
- NFS mounts to Isilon (3 for each Grid node):
  - /multiuser: Logs, code, scripts
  - /multiuser/sas7bdat: Input and output project folder for data
  - /multiuser/output: Mostly output directory

  All mounts were carefully spread across all the interfaces of the array.
- NFS data mount options:
  - nfsvers=3,vers=3,tcp,rw,hard,inrt,retrains=2,nosuid,noatime,nodiratime
- Network settings:
  - Jumbo Frames
  - MTU 9000
- Other system settings used in /etc/sysctl.conf
  - fs.file-max=2000000
  - vm.swappiness=0
  - vm.dirty_background_ratio=20
  - vm.dirty_ratio=50
  - net.core.rmem_max = 67108864
  - net.core.wmem_max = 67108864

  - net.ipv4.tcp_rmem = 4096 87380 33554432
  - net.ipv4.tcp_wmem = 4096 65536 33554432
  - net.ipv4.tcp_congestion_control=htcp
  - net.core.default_qdisc = fq
  - vm.dirty_writeback_centisecs=500
  - net.core.netdev_max_backlog=1000
  - net.core.rmem_default=212992
  - net.core.wmem_default=212992
  - net.core.somaxconn=65530
  - net.ipv4.tcp_fin_timeout=60
  - sunrpc.tcp_slot_table_entries=128
  - net.ipv4.tcp_timestamps=0

- Isilon models:
  - Isilon F810-4U-Single-256GB-1x1GE-2x40GE SFP+-24TB SSD, OneFS version 8.1.3
  - Isilon F800-4U-Single-256GB-1x1GE-2x40GE SFP+-24TB SSD, OneFS version 8.2.0
- Volumes of data used in each batch to support the 33 SAS jobs:
  - Input data /multiuser/sas7bdat for a single Grid node: 28 input files
    - Uncompress datasets = 1433 GB
    - SAS compressed datasets = 503 GB
    - SAS compression and F810 hardware compression = 149 GB

7 Test results and benefits of using Isilon F810 with SAS

7.1 Performance of the array

The mixed workload was run against the older Isilon F800 system and the newer Isilon F810 system with the same set of SAS Grid nodes during the test. The first set of tests ran a single SAS Grid server with a single batch of the workload and compared the runtime results of the 33 jobs. 0 compares the runtimes of the individual jobs with the average runtime and the sum of all runtimes for the 33 jobs. The table shows a single server running one workload batch against both array types, with and without SAS compression.

Table 1 Job runtime comparison

SAS job name in test suite	F800, SAS compression = none	F800, SAS compression = binary	F810, SAS compression = binary and hardware compression
bank1_1	0:53:26	0:53:44	0:41:02
bank1_3	0:53:12	0:53:26	0:41:13
bank2_1	2:17:14	1:24:45	0:49:28
bank2_3	2:17:03	1:23:58	0:49:26
comp_glm_1a	0:00:39	0:00:42	0:00:37
comp_glm_4a	0:00:45	0:00:53	0:00:44
comp_glm_4b	0:00:43	0:00:51	0:00:46
etl_inbound_1	0:05:02	0:43:29	0:12:12
etl_inbound_4	0:07:41	0:40:07	0:12:37
fscheck_a	0:00:01	0:00:02	0:00:02
fscheck_c	0:00:00	0:00:01	0:00:01
fscheck_f	0:00:00	0:00:02	0:00:01
fscheck_i	0:00:01	0:00:00	0:00:00
fscheck_l	0:00:00	0:00:00	0:00:00
fscheck_m	0:00:01	0:00:05	0:00:04
hplogistic_1	0:20:30	0:09:44	0:12:25
hplogistic_2	0:17:08	0:10:23	0:12:04
rtumble_1	0:36:21	0:07:41	0:07:47
rwrw_1	0:18:25	0:54:42	0:34:05
rwrw_2	0:17:29	0:51:10	0:32:12
rwtumble_1	0:36:51	0:10:16	0:10:25
smallnoise_11b	0:01:05	0:01:04	0:00:59
smallnoise_17	0:01:09	0:01:04	0:00:59

SAS job name in test suite	**F800, SAS compression = none**	**F800, SAS compression = binary**	**F810, SAS compression = binary and hardware compression**
smallnoise_18	0:01:13	0:01:09	0:00:59
smallnoise_5	0:01:18	0:01:01	0:00:59
smallnoise_6a	0:01:08	0:01:01	0:00:59
smallnoise_6	0:01:16	0:01:02	0:00:59
smallnoise_9	0:01:04	0:01:00	0:00:59
sort_1	0:20:07	0:27:55	0:03:41
where_test_1	0:10:24	0:24:30	0:02:19
wr_junk_10	1:21:08	0:52:13	0:36:34
wr_junk_1	1:25:18	0:56:18	0:37:16
wr_junk_3	1:25:16	0:56:22	0:37:19
Sum of ALL Jobs Runtimes	13:52:58	12:10:40	7:21:13
Average individual Job Runtime	25:14	22:08	13:22

The results show that with the F810 system using hardware compression running with the same network, the SAS jobs and system settings were significantly faster than the other tests on the older F800 system.

During the single-server testing on the F810 system, we captured the I/O throughput in MBps of both SASWORK and the shared file system (NFS) during the test (Figure 8). The graph shows that the I/O rates peak at 2.5 GBps for short periods and sustains I/O rates of 600–1200 MBps for total I/O. The graph also shows extended periods of NFS I/O rates of 500–600 MBps early in the test scenario. This matches the typical sustained NFS I/O rate experienced by a SAS financial customer with forty 12-core SAS Grid nodes, which is the design goal for the test.

Before running the test scenarios or batches, the file system cache (memory caching of all I/O reads to storage) was flushed. All systems started with a cold file cache. In production NFS environments, it is encouraged to build SAS Grid servers with a high ratio of RAM-to-CPU cores to augment I/O performance. A common statement can be applied here, which says “The fastest I/O read is the one you never had to read from disk.” When you write data, and re-read it often, it stays resident in memory and takes the pressure of the NFS file system by reducing unnecessary reads.

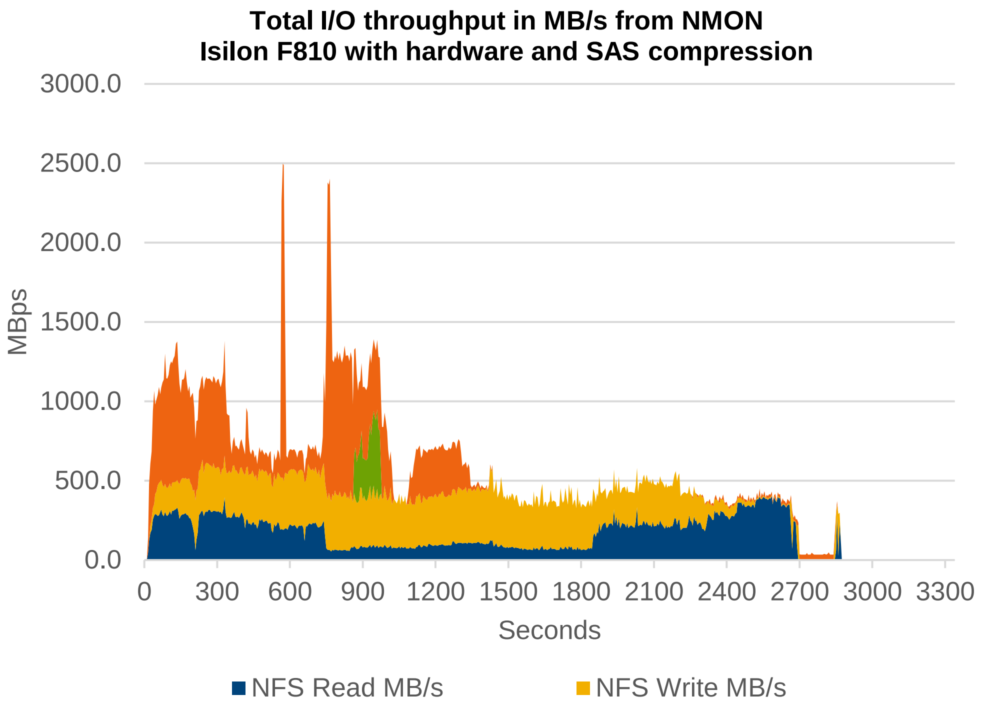


Figure 8 I/O throughput of all I/O channels for a single server during a single batch of the workload

7.2 Scalability

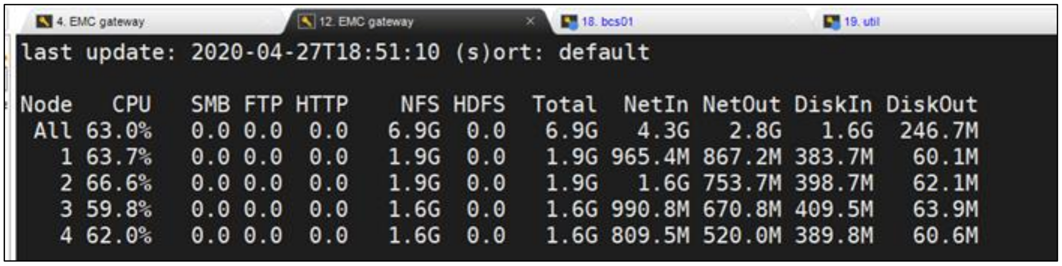
Each server during the test ran its own copy of the workload (batch). Tests were run with 1, 2, 4, 8, and 12 simultaneous batch executions across an equal number of SAS Grid nodes. During each increase of work (batches) and grid nodes, no additional Isilon resources for the Isilon F810 were added to the environment.

Table 2 shows the average job runtime, maximum job runtime, standard deviation of the runtimes, and the average sustained I/O throughput from all the SAS Grid servers used in each test scenario. Runtimes are listed because they are critical to the user experience. Being able to provide a sustained and predictable runtime for users is also critical to user satisfaction. Having the ability to add more SAS Grid nodes and maintain the runtime consistency up to 12 Grid nodes demonstrates the scalability of the architecture. Figure 9 and Figure 10 show the Isilon I/O statistics and show is significant available CPU capacity during the 12-node testing.

Table 2 Scalability of throughput from 1 to 12 SAS Grid nodes

Test scenario	SAS programs run	SAS Grid nodes	Average job runtime (MM:ss)	Maximum job runtime (HH.MM:ss)	Standard deviation in job runtime comparing all jobs	Sustained throughput at peak times on Isilon (R+W)
1	33	1	13:12	49:28	16:58	650 to 750 MBps
2	66	2	12:51	47:18	16:12	1 to 1.4 GBps
3	132	4	13:11	49:20	16:42	2 to 2.5 GBps
4	264	8	13:02	49:57	16:28	4.5 to 5 GBps
5	396	12	12:28	49:30	15:47	6.5 to 7 GBps

Figure 9 shows a snapshot on the Isilon F810 system during the 12-node run. There is still significant CPU capacity available on the Isilon system, which shows potential for more headroom to support additional NFS clients.

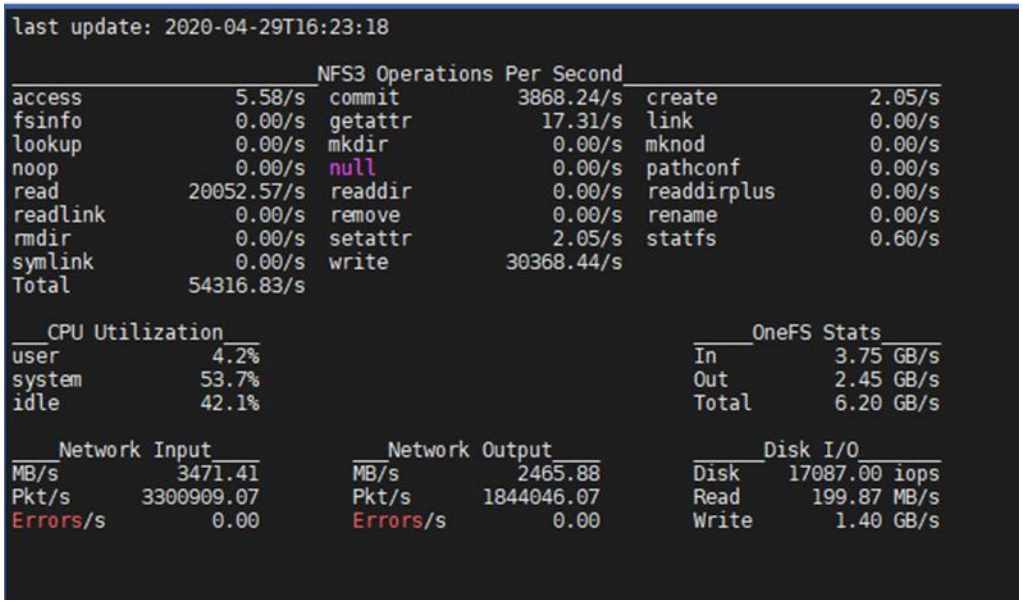


```
last update: 2020-04-27T18:51:10 (s)ort: default

Node   CPU   SMB FTP HTTP    NFS HDFS  Total  NetIn NetOut DiskIn DiskOut
 All 63.0%   0.0 0.0  0.0   6.9G  0.0   6.9G   4.3G   2.8G   1.6G  246.7M
   1 63.7%   0.0 0.0  0.0   1.9G  0.0   1.9G 965.4M 867.2M 383.7M   60.1M
   2 66.6%   0.0 0.0  0.0   1.9G  0.0   1.9G   1.6G 753.7M 398.7M   62.1M
   3 59.8%   0.0 0.0  0.0   1.6G  0.0   1.6G 990.8M 670.8M 409.5M   63.9M
   4 62.0%   0.0 0.0  0.0   1.6G  0.0   1.6G 809.5M 520.0M 389.8M   60.6M
```

Figure 9 Snapshot on Isilon F810 system during 12-node run

Figure 10 shows a snapshot on the Isilon F810 system during the 12-node test scenario. Figure 11 shows more detail about the I/O workload across the cluster. The graph shows the total I/O on a Grid node during the 12-node test scenario. Its runtime and pattern matches the single node run showed in Figure 8.



```
last update: 2020-04-29T16:23:18

__________________________NFS3 Operations Per Second___________________________
access             5.58/s  commit          3868.24/s  create             2.05/s
fsinfo             0.00/s  getattr           17.31/s  link               0.00/s
lookup             0.00/s  mkdir              0.00/s  mknod              0.00/s
noop               0.00/s  null               0.00/s  pathconf           0.00/s
read           20052.57/s  readdir            0.00/s  readdirplus        0.00/s
readlink           0.00/s  remove             0.00/s  rename             0.00/s
rmdir              0.00/s  setattr            2.05/s  statfs             0.60/s
symlink            0.00/s  write          30368.44/s
Total          54316.83/s

___CPU Utilization___                                     _____OneFS Stats_____
user             4.2%                                     In          3.75 GB/s
system          53.7%                                     Out         2.45 GB/s
idle            42.1%                                     Total       6.20 GB/s

____Network Input____        ___Network Output____        ______Disk I/O_______
MB/s          3471.41        MB/s          2465.88        Disk    17087.00 iops
Pkt/s      3300909.07        Pkt/s      1844046.07        Read      199.87 MB/s
Errors/s         0.00        Errors/s         0.00        Write       1.40 GB/s
```

Figure 10 Snapshot on Isilon F810 system during 12-node test scenario

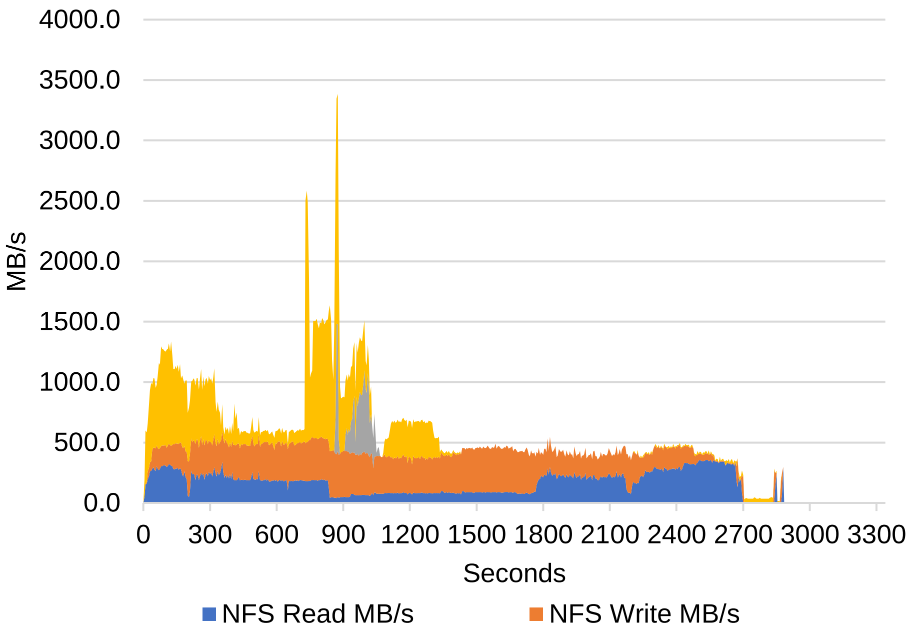


Figure 11 Graph of total I/O on Grid node during 12-node test scenario

It is important to look at individual job behavior during the test scenarios. One of the longer running tests, bank2, is a typical data-manipulation job that is used to help analyze and create an analytic input in preparation for doing advanced analytics. This test job reads and manipulates a 150-million-row table with 126 variables. It runs several SAS PROCEDURES, runs basic statistics, calculates new values, runs basic statistics, and even generates an index on the file for later use. In both the F800 and F810 test scenarios, as the number of Grid nodes increased, the runtime of the job remained the same. However, the F810 system was able to run the job faster. Again, the servers and network used in this test were the same, and the only difference is the storage. The 12-node test was not run on the F800 due to lack of availability of the 12-nodes during that testing cycle. Table 3 shows the runtimes for the bank job.

Table 3 Comparison of running bank job 2 on both array types as the workload scales up

Grid nodes used in test	F800 (HH:mm)	F810 (HH:mm)
1	1:24	0:49
2	1:25, 1:22	0:48, 0:49
4	1:26, 1:21, 1:25, 1:22	0:48, 0:49, 0:49, 0:47
8	1:25, 1:22, 1:25, 1:21, 1:24, 1:25, 1:18, 1:23	0:45, 0:48, 0:49, 0:45, 0:48, 0:47, 0:45, 0:46
12	Not run	0:49, 0:47, 0:45, 0:49, 0:46, 0:44, 0:50, 0:48, 0:50, 0:49

7.3 Compression

One of the most exciting features of the F810 system is the addition of the hardware compression to the array. As shown in Table 4, SAS compression provides a large reduction in storage-space requirements for input data. The addition of the F810 hardware compression reduces the storage requirement by almost 3 to 1. The F810 provides compression enhancement with no reduction in performance (see previous section of the paper on performance).

Table 4 Sized of compressed and uncompressed tables used in the test scenarios for a single batch of the test workload

Input filename	F800 no SAS compression	F800 with SAS compression	F810 with SAS compression
bank1input_1.sas7bdat	63 GB	22 GB	2.8 GB
bank1input_2.sas7bdat	63 GB	22 GB	2.8 GB
bank1input_3.sas7bdat	63 GB	22 GB	2.8 GB
bank1input_4.sas7bdat	63 GB	22 GB	2.8 GB
bank2input_1.sas7bdat	184 GB	57 GB	7.2 GB
bank2input_2.sas7bdat	185 GB	57 GB	7.2 GB
bank2input_3.sas7bdat	185 GB	57 GB	7.2 GB

Input filename	F800 no SAS compression	F800 with SAS compression	F810 with SAS compression
bank2input_4.sas7bdat	185 GB	57 GB	7.2 GB
glminput_1.sas7bdat	4.6 MB	6.3 MB	2.8 MB
glminput_2.sas7bdat	4.8 MB	6.6 MB	2.8 MB
multiuser_1.sas7bdat	22 GB	17 GB	14 GB
multiuser_2.sas7bdat	22 GB	17 GB	14 GB
multiuser_3.sas7bdat	22 GB	17 GB	14 GB
multiuser_4.sas7bdat	22 GB	17 GB	14 GB
ranrw_medium_1.sas7bdat	13 GB	825 MB	103 MB
ranrw_medium_2.sas7bdat	13 GB	825 MB	103 MB
ranrw_skinny_1.sas7bdat	1.6 GB	480 MB	78 MB
ranrw_skinny_2.sas7bdat	1.6 GB	480 MB	78 MB
ranrw_small_1.sas7bdat	544 KB	544 KB	64 KB
ranrw_small_2.sas7bdat	544 KB	544 KB	64 KB
ranrw_wide_1.sas7bdat	51 GB	1.7 GB	210 MB
ranrw_wide_2.sas7bdat	51 GB	1.7 GB	210 MB
simdata_1.sas7bdat	40 GB	55 GB	19 GB
simdata_2.sas7bdat	16 GB	22 GB	7.3 GB
simdata_tnk_1.sas7bdat	12 GB	9.6 GB	8.8 GB
simdata_tnk_2.sas7bdat	12 GB	9.6 GB	8.8 GB
sortinput_1.sas7bdat	25 GB	5.2 GB	1.7 GB
sortinput_2.sas7bdat	99 GB	21 GB	6.6 GB
GB total	**1433.6**	**503**	**149**
Storage savings ratio		**3.3:1**	

There are occasions that SAS compression inhibits the performance of some SAS jobs. In testing, one of the jobs that uses a 10-million-row table with 112 variables showed decreased performance when comparing SAS compression to an uncompressed dataset when running on the older F800 array. However, the addition of the F810 hardware compression without the SAS compression ran faster than the original F800 with or without the compression. SAS compression can be turned off for any portion of a SAS job or the entire job.

Some jobs have issues with SAS compression. Table 5 shows the benefit that is gained by turning off SAS compression while still using hardware compression of the F810 system.

Table 5 Turning off SAS compression while using hardware compression of the F810 system

Isilon model	SAS compression = binary	Isilon hardware compression	File size: du -sh	Runtime to create file (MM:ss)	Data step, copy file from NFS to NFS lib (MM:ss)	All steps, total SAS job (MM:ss)
F800	-	-	12 GB	1:40	3:35	18:25
F800	Yes	-	9.6 GB	6:08	30:17	54:42
F810	-	Yes	8 GB	1:10	8:24	14:00
F810	Yes	Yes	8.8 GB	8:53	7:35	34:05

7.4 Deduplication

Another feature of the Isilon F810 system is deduplication of on-disk data. Much of the data in an analytics Grid is data that is pulled from other data sources and combined to create analytic tables for processing. It is common that different modelers and coworkers share or create similar if not duplicate tables (sometimes by accident). This feature requires further research by SAS, Dell Technologies™, and its partners, but the initial tests run with this feature show that it can further enhance the storage-space reduction achieved by the F810 compression capability.

The following results show running deduplication on the F810c system after running the 12-node test scenario. The total data before the run includes compressed SAS datasets, scripts, and code that are stored on the F810 array and are already compressed by the integrated hardware compression of the Isilon system. The deduplication process shows that roughly 40% of the data on disk were duplicated blocks, were cataloged, and the duplicates were deleted.

Test scenario data before and after deduplication:

- Before: 4.6 TB, 6% use
- After: 2.7 TB, 4% use

Deduplication job run results:

```
Job Report Details
Time: 2020-04-02 03:32:08
Event ID: 3.13534
Job ID: 1207
Job Type: Dedupe
Phase: 1
Report:
```

```
Dedupe job report:{
  Start time = 2020-Apr-02:02:34:40
  End time = 2020-Apr-02:06:32:08
  Iteration count = 3
  Scanned blocks = 1182629476
  Sampled blocks = 45504643
  Deduped blocks = 528351533
  Dedupe percent = 44.676
  Created dedupe requests = 34065196
  Successful dedupe requests = 33986741
  Unsuccessful dedupe requests = 78455
  Skipped files = 1195
  Previously assessed files = 455
  Index entries = 10387523
  Index lookup attempts = 7479509
  Index lookup hits = 1164297
}
Elapsed time: 14248 seconds
Aborts: 0
Errors: 0
Scanned files: 317
Directories: 179
1 path:
/ifs/f810c
CPU usage: max 194% (dev 4), min 0% (dev 1), avg 121%
Virtual memory size: max 539432K (dev 1), min 441384K (dev 3), avg 504675K
Resident memory size: max 89376K (dev 1), min 22352K (dev 2), avg 55837K
Read: 113141338 ops, 926853840896 bytes (883916.7M)
Write: 175404067 ops, 1436910116864 bytes (1370344.3M)
Other jobs read: 15 ops, 122880 bytes (0.1M)
Other jobs write: 493183 ops, 4040155136 bytes (3853.0M)
Non-JE read: 1043 ops, 8544256 bytes (8.1M)
```

8 Conclusion

After testing the F810 system using the multiuser analytic workload, it is easy to see that the enhanced performance of the F810 array compared to the older hardware and scalability justify transitioning to the latest Isilon model. With older models of Isilon systems, it is common to use a ratio of 1.5 Grid nodes per Isilon storage node when building or designing a SAS Grid. With the enhanced performance of the newer model, it is possible to run a ratio of 3 Grid nodes per Isilon storage node.

Beyond the performance improvements of the F810 system over the older array, the compression capability and deduplication potential can provide significant savings for financial services institutions that are seeing increasing pressure to make more data available with the same storage budget.

In future testing, we may be able to share results for SAS Grid nodes with dual 25 GbE connections. Based on our experience, being able to increase I/O throughput potential to the Grid nodes is important to support SAS Grid nodes larger than 12 cores. The limiting factor of dual 10 GbE connections has been a common deployment in customer sites. However, we are seeing a move to faster networking as customers upgrade their servers.

NFS for SAS Grid is also a viable storage option. However, it is only successful with careful planning and following strict adherence to SAS guidelines on requirements like those for SASWORK (which should never be placed on NFS).

A Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

Storage technical documents and videos provide expertise that helps to ensure customer success on Dell EMC storage platforms.