White Paper

Dell EMC Ready Solutions for HPC BeeGFS High Capacity Storage

Technical White Paper

Abstract

This white paper describes the architecture, tuning guidelines, and performance of a high capacity, high-throughput, scalable BeeGFS file system solution.

June 2020

Revisions

Date	Description
July 2020	Initial release

Acknowledgements

Author: Nirmala Sundararajan

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.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Table of contents

Executive summary

In high-performance computing (HPC), designing a well-balanced storage system to achieve optimal performance presents significant challenges. A typical storage system consists of a variety of considerations, including file system choices, file system tuning, and disk drive, storage controller, IO cards, network card, and switch strategies. Configuring these components for best performance, manageability, and future scalability requires a great deal of planning and organization.

The Dell EMC Ready Solution for HPC BeeGFS High Capacity Storage is a fully supported, easy-to-use, high-throughput, scale-out, parallel file system storage solution with well-described performance characteristics. This solution is offered with deployment services and full hardware and software support from Dell Technologies.

The solution scales to greater than 5 PB of raw storage and uses Dell EMC PowerEdge servers and the Dell EMC PowerVault storage arrays. This white paper describes the solution's architecture, scalability, tuning best practices, and flexible configuration options and their performance characteristics.

Dell Technologies and the author of this document welcome your feedback on the solution and the solution documentation. Contact the Dell Technologies Solutions team by email or provide your comments by completing our documentation survey.

1 BeeGFS High Capacity Storage solution overview

The Dell EMC Ready Solutions for HPC BeeGFS Storage is available in three base configurations, small, medium and large. These base configurations can be used as building blocks to create additional flexible configurations to meet different capacity and performance goals as illustrated in Figure 1. Contact a Dell EMC sales representative to discuss which offering works best in your environment, and how to order.

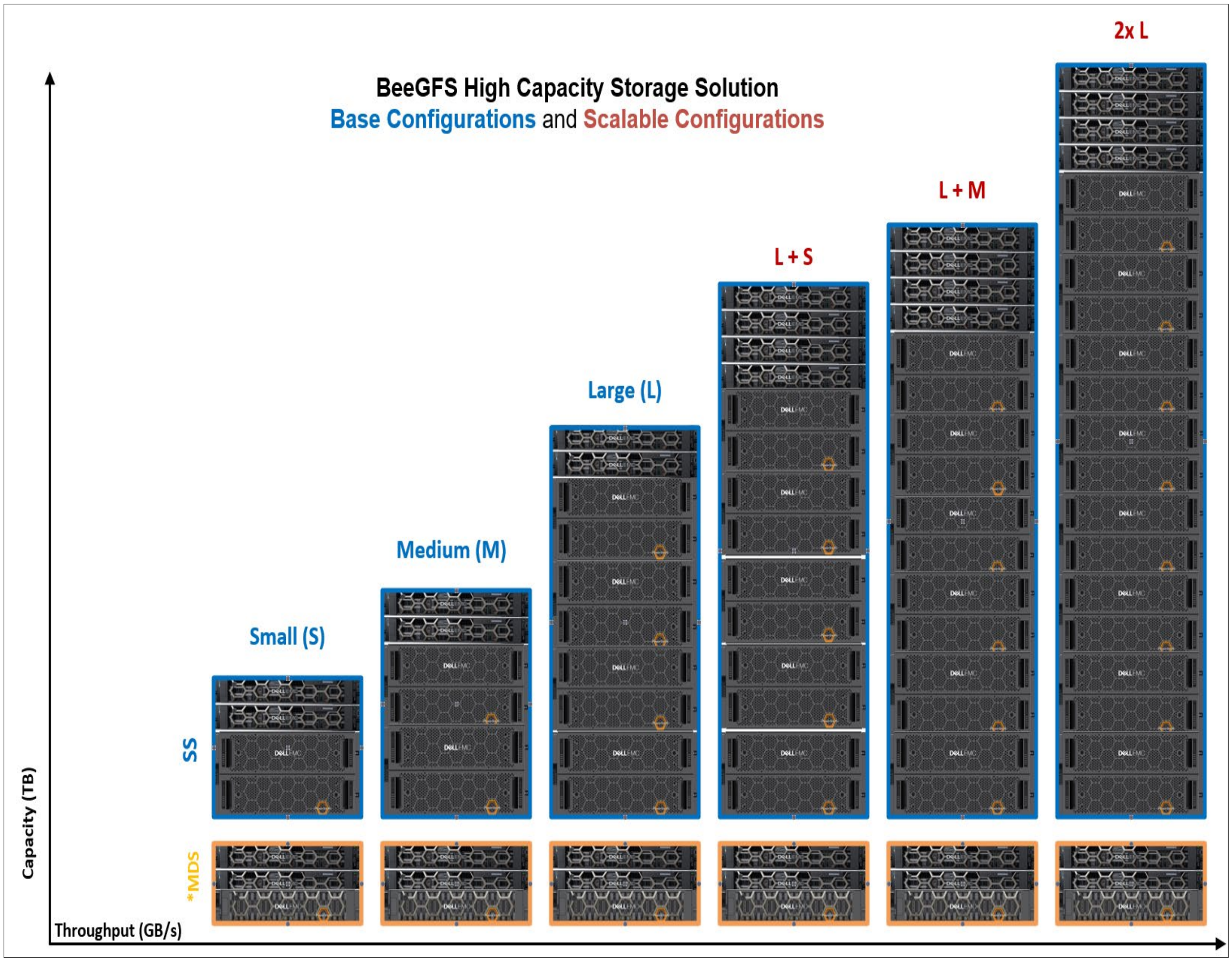

BeeGFS High Capacity Storage Solution
Base Configurations and Scalable Configurations
Capacity (TB)
Throughput (GB/s)
SS
*MDS
Small (S)
Medium (M)
Large (L)
L + S
L + M
2x L

Figure 1 BeeGFS High Capacity Storage solution—base and scalable configurations

Available capacity and performance options are described in detail in the scalable configuration section of this white paper.

The metadata component of the solution that includes a pair of metadata servers (MDS) and a metadata target storage array remains the same across all the configurations as shown in Figure 1. The storage component of the solution includes a pair of storage servers (SS) and a single storage array for the small configuration, while a medium configuration uses two storage arrays and a large configuration uses four storage arrays. The PowerVault ME4024 storage array is used as the metadata storage, and PowerVault ME4084 arrays are used as the data storage.

To scale beyond the large configuration, an additional pair of storage arrays is needed. The additional SS pair can have either one, two, or four storage arrays as noted in the flexible configurations. This pattern for sizing a configuration allows further scaling. The performance of the Large base configuration is described in the performance evaluation section of this white paper. The performance of Scalable configurations built from base configurations is derived from these measured results as explained in the scalable configurations subsection of this white paper.

2 BeeGFS file system

This storage solution is based on BeeGFS, an opensource parallel file system, which offers flexibility and easy scalability. The general architecture of BeeGFS consists of four main services: management, metadata, storage, and client. The server components are user space daemons. The client is a patchless kernel module. An additional monitoring service is also available.

The key elements of the BeeGFS file system are as follows:

- **Metadata targets (MDTs)**

  Stores all the metadata for the file system including filenames, permissions, time stamps, and the location of stripes of data.

- **Management target (MGMTD)**

  Stores management data such as configuration and registry.

- **Metadata server (MDS)**

  Manages the filesystem namespace (files and directories) and file layout allocation, and determines where the files are saved on the storage devices. Manages the MDTs, providing BeeGFS clients with access to files.

- **Storage targets (STs)**

  Stores the data stripes or extents of the files on a file system.

- **Storage server (SS)**

  Manages the STs, providing BeeGFS clients with access to the data.

- **Clients**
  The BeeGFS client kernel module is installed on the clients to allow access to data on the BeeGFS file system. To the clients, the file system appears as a single namespace that can be mounted for access.

For more information on BeeGFS file system architecture, see Introduction to BeeGFS.

3 BeeGFS High Capacity Storage solution architecture

Figure 2 shows the large configuration architecture with four PowerVault ME4084 storage arrays.

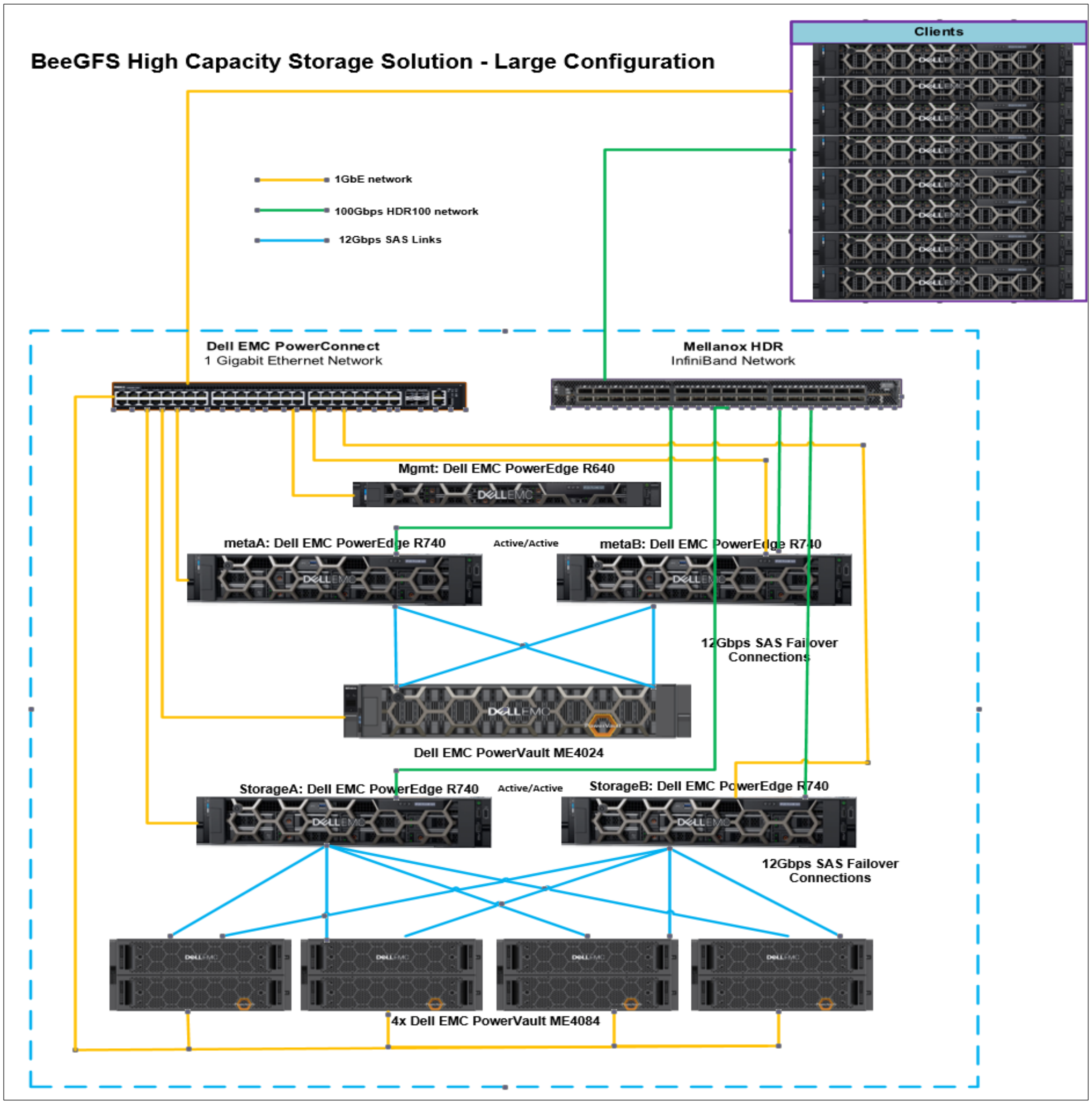

BeeGFS High Capacity Storage Solution - Large Configuration
Clients
1GbE network
100Gbps HDR100 network
12Gbps SAS Links
Dell EMC PowerConnect
1 Gigabit Ethernet Network
Mellanox HDR
InfiniBand Network
Mgmt: Dell EMC PowerEdge R640
metaA: Dell EMC PowerEdge R740
Active/Active
metaB: Dell EMC PowerEdge R740
12Gbps SAS Failover Connections
Dell EMC PowerVault ME4024
StorageA: Dell EMC PowerEdge R740
Active/Active
StorageB: Dell EMC PowerEdge R740
12Gbps SAS Failover Connections
4x Dell EMC PowerVault ME4084

Figure 2 Solution reference architecture—large configuration

In Figure 2, the management server (the topmost server) is a PowerEdge R640. The MDS function is provided by two PowerEdge R740 servers. The MDS pair is attached to a PowerVault ME4024 through 12 Gb/s SAS links. The PowerVault M4024 hosts the metadata targets.

The SSs are a pair of PowerEdge R740 servers. The SS pair is attached to four fully populated PowerVault ME4084 storage arrays through 12 Gb/s SAS links. The four PowerVault ME4084 arrays host the storage targets.

The solution uses Mellanox HDR100 InfiniBand as the data network connecting the storage solution to the compute clients. Gigabit Ethernet is used for management operations.

3.1 Management server

The single management server is connected to the MDS pair and SS pair through an internal 1 GbE network.

BeeGFS provides a tool called `beegfs-mon` that collects use and performance data from the BeeGFS services and stores the data in a timeseries database called InfluxDB. `beegfs-mon` provides predefined Grafana panels that can be used out of the box to extract and visualize this data. Both Grafana and InfluxDB are installed on the management server. After starting and enabling InfluxDB and Grafana server services, Grafana dashboard can be accessed using the url http://<IP of mgmt.>:3000. The default credentials are admin/admin.

3.2 Metadata servers

Each MDS in the MDS pair is equipped with two dual-port 12 Gb/s SAS host bus adapters and one Mellanox InfiniBand HDR100 adapter. Figure 3 shows the recommended slot assignments for the SAS HBAs as slots 1 and 4. This allows each SAS HBA to be mapped to one processor in the server for load balancing. The Mellanox InfiniBand HDR100 HCA is installed in slot 8, which is a PCIe x16 slot.

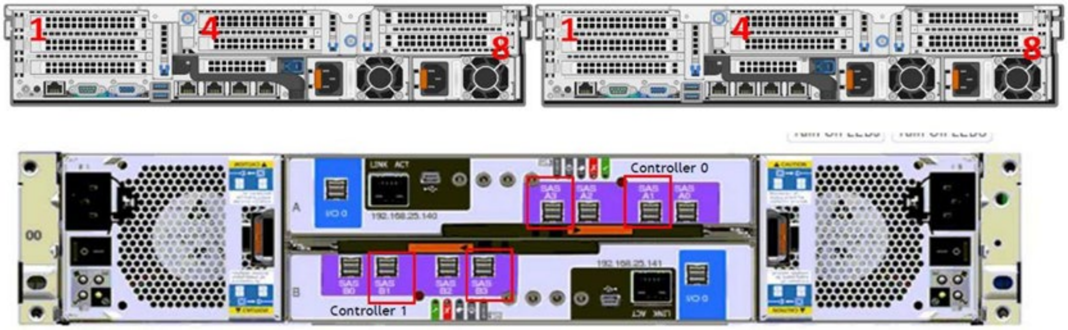


Figure 3 MDS slot priority and ME4024 SAS ports

The `beegfs-mgmtd` service in the BeeGFS high capacity solution is not actually running on the management server. It is managed by Pacemaker to run on either metaA or metaB. Its storage is on the first meta partition on metaA server and will move together with `beegfs-mgmtd` service in case of a service failure. The `beegfs-mgmtd` service is set up on both metadata servers and is started on the metaA server. The `beegfs mgmtd` store is initialized on the directory mgmtd on the metadata target 1 as follows:

```
/opt/beegfs/sbin/beegfs-setup-mgmtd -p /beegfs/metaA-numa0-1/mgmtd -S beegfs-
mgmt
```

The two PowerEdge R740 servers that are used as the MDS are directly attached to the PowerVault ME4024 storage array housing the BeeGFS MDTs and MGMTD. Refer to Appendix A Storage array cabling for details on how the metadata servers are cabled to the ME4024 storage array.

3.3 Metadata targets

The ME4024 array is fully populated with 24 x 960 GB SAS SSDs. An optimal way to configure the 24 drives for metadata is to configure twelve MDTs. Each MDT is a RAID 1 disk group of two drives each. Figure 4 shows how the MDTs are configured.

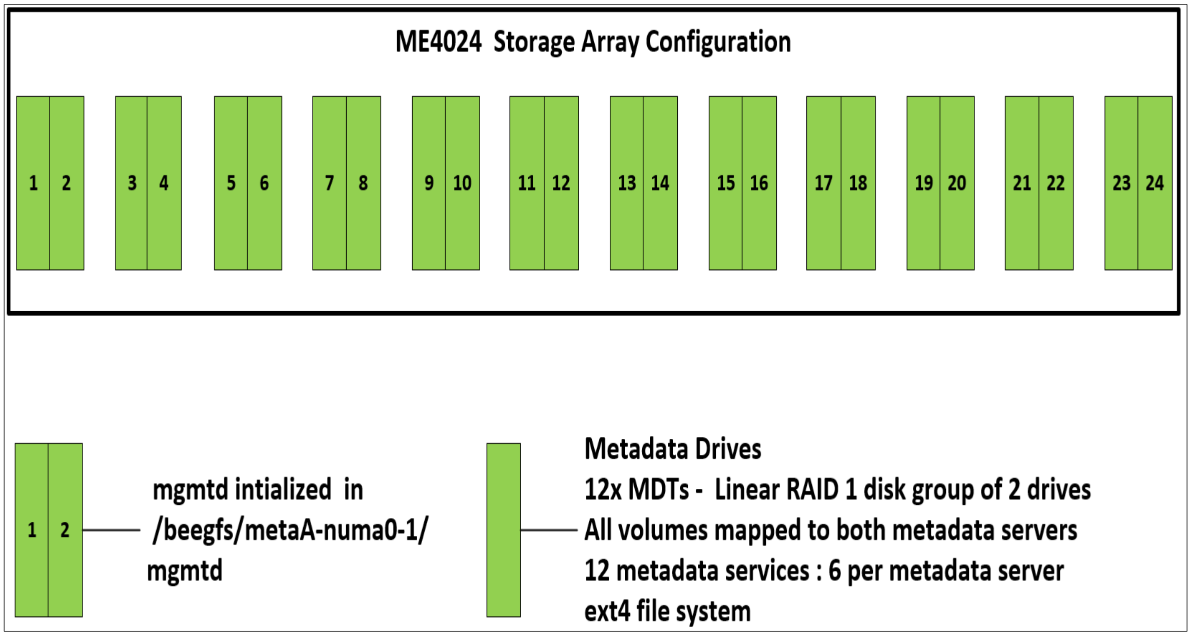

ME4024 Storage Array Configuration
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
1 2
mgmtd intialized in /beegfs/metaA-numa0-1/ mgmtd
Metadata Drives
12x MDTs - Linear RAID 1 disk group of 2 drives
All volumes mapped to both metadata servers
12 metadata services : 6 per metadata server
ext4 file system

Figure 4 Configuration of metadata Targets in the ME4024 storage array

The metadata target is formatted with ext4 file system because ext4 performs well with small files and small file operations. Additionally, BeeGFS stores information as extended attributes and directly in the inodes of the filesystem. Both of these work well with ext4 file system.

3.4 Storage servers

Each SS in the SS pair is equipped with four dual-port 12 Gb/s SAS host bus adapters and one Mellanox InfiniBand HDR100 adapter to handle storage requests. Figure 5 shows the recommended slot assignments for the SAS HBAs as slots 1, 2, 4 a and 5. This allows the SAS HBAs to be evenly distributed across the two processors for load balancing. The Mellanox InfiniBand HDR100 HCA is installed in slot 8, which is a PCIe x16 slot.

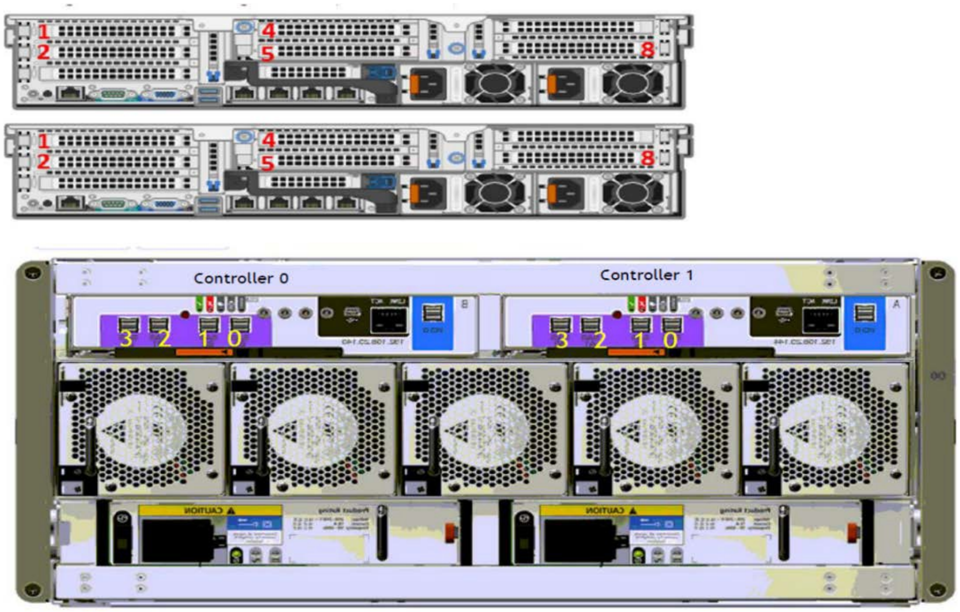


Figure 5 SS slot priorities and ME4084 SAS ports

Note: To clearly display the SAS ports, Figure 5 shows one ME4084. This configuration has 4 x ME4084 arrays, as shown in Figure 2.

With four dual-port 12 Gb/s SAS controllers in each PowerEdge R740, the two servers are redundantly connected to each of the four PowerVault ME4084 high-density storage arrays, with a choice of 4 TB, 8 TB, 12 TB or 16 TB of NL SAS 7.2 K RPM hard disk drives (HDDs). Refer to Appendix A Storage array cabling for details on the storage cabling for the various base configurations of the solution.

3.5 Storage targets

Figure 6 illustrates how each storage array is divided into eight linear RAID 6 disk groups, with eight data and two parity disks per virtual disk.

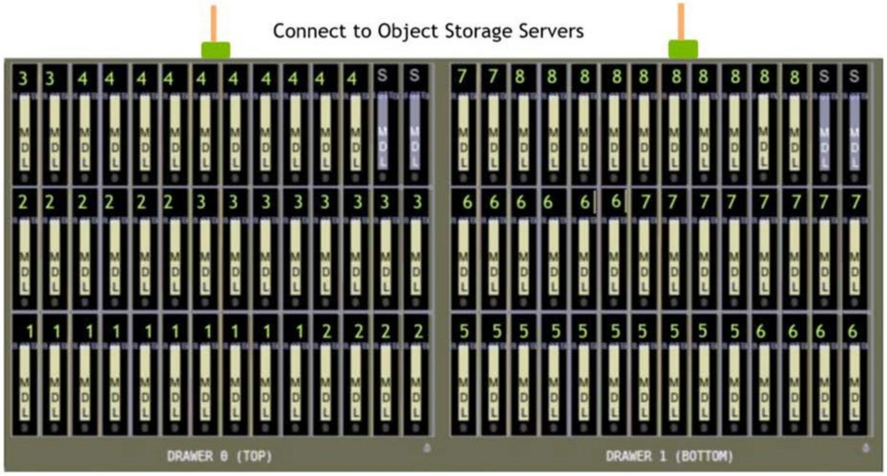


Figure 6 RAID 6 (8+2) LUNs layout on one ME4084

Each OST provides about 64 TB of formatted object storage space when populated with 8 TB HDDs. Since each array has 84 drives, after creating eight RAID-6 disk groups, we have 4 spare drives per array, 2 per tray, which can be configured as global hot spares across the 8 disk groups in the array. For every disk group, a single volume using all the space is created. As a result, a large base configuration as shown in Figure 2 has a total of 32 linear RAID 6 volumes across four ME4084 storage arrays. Each of these RAID 6 volumes are configured as an OST for the BeeGFS file system, resulting in a total of 32 STs across the file system in the base configuration.

The STs are exposed to clients via Mellanox InfiniBand HDR100 connection. From any compute node that is equipped with the BeeGFS client, the entire namespace can be viewed and managed like any other file system.

3.6 Hardware and software configuration

The following table describes the hardware and software details of the solution.

Table 1 Solution hardware and software specifications:

Component	**Specification**
Management server	1 x Dell EMC PowerEdge R640
MDS	2 x Dell EMC PowerEdge R740
Storage servers	2 x Dell EMC PowerEdge R740
Processors	Management server: Dual Intel Xeon Gold 5218 MDS and SS servers: Dual Intel Xeon Gold 6230
Memory	Management server: 12 x 8 GB 2666 MT/s DDR4 RDIMMs MDS and SS servers: 12 x 32 GB 2933 MT/s DDR4 RDIMMs

Component	Specification
Local disks and RAID controller	Management server: PERC H740P Integrated RAID, 8GB NV cache, 6x 300GB 15K SAS hard drives (HDDs) configured in RAID10 MDS and SS servers: PERC H330+ Integrated RAID, 2x 300GB 15K SAS HDDs configured in RAID1 for OS
InfiniBand HCA	Mellanox ConnectX-6 HDR100 InfiniBand adapter
External storage controllers	On each MDS: 2 x Dell 12 Gb/s SAS HBAs On each SS: 4 x Dell 12 Gb/s SAS HBAs
Object storage enclosures	4 x Dell EMC PowerVault ME4084 fully populated with a total of 336 drives
Metadata storage enclosure	1 x Dell EMC PowerVault ME4024 with 24 SSDs
RAID controllers	Duplex RAID controllers in the ME4084 and ME4024 enclosures
HDDs	On each ME4084 Enclosure: 84 x 8 TB 3.5 in. 7.2 K RPM NL SAS3 ME4024 Enclosure: 24 x 960 GB SAS3 SSDs
Operating system	CentOS Linux release 8.1.1911 (Core)
Kernel version	4.18.0-147.5.1.el8_1.x86_64
Mellanox OFED version	4.7-3.2.9.0
BeeGFS file system version	7.2 (beta2)

4 Performance evaluation

Our performance studies of the solution uses Mellanox HDR100 data networks. Performance testing objectives were to quantify the capabilities of the solution, identify performance peaks, and determine the most appropriate methods for scaling. We ran multiple performance studies, stressed the configuration with different types of workloads to determine the limitations of performance, and defined the sustainability of that performance.

We generally try to maintain a standard and consistent testing environment and methodology. In some areas we purposely optimized server or storage configurations and took measures to limit caching effects

4.1 Large base configuration

We performed the tests on the solution configuration described in Table 1. The following table details the client test bed that we used to provide the I/O workload:

Table 2 Client configuration

Component	**Specification**
Operating system	Red Hat Enterprise Linux Server release 7.6 (Maipo)
Kernel version	3.10.0-957.el7.x86_64
Servers	8x Dell EMC PowerEdge R840
BIOS version	2.4.7
Mellanox OFED version	4.7-3.2.9.0
BeeGFS file system version	7.2 (beta2)
Number of physical nodes	8
Processors	4 x Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz, 24 cores
Memory	24 x 16GB DDR4 2933MT/s DIMMs - 384GB

Our performance analysis focused on these key performance characteristics:

- Throughput, data sequentially transferred in GB/s
- I/O operations per second (IOPS)
- Metadata operations per second (OP/s)

The goal was a broad but accurate overview of the capabilities of the solution using the Mellanox InfiniBand HDR100. We used the IOzone, IOR and MDtest benchmarks. IOzone uses an N-to-N file-access method. N-to-N load was tested, where every thread of the benchmark (N clients) writes to a different file (N files) on the storage system. For examples of the commands that we used to run these benchmarks, see Appendix B Benchmark command reference.

We ran each set of tests on a range of clients to test the scalability of the solution. The number of simultaneous physical clients involved in each test ranged from a single client to eight clients. The number of threads per node corresponds to the number of physical compute nodes, up to eight. The total number of threads above eight were simulated by increasing the number of threads per client across all clients. For instance, for 128 threads, each of the eight clients ran 16 threads.

To prevent inflated results due to caching effects, we ran the tests with a cold cache. Before each test started, the BeeGFS file system under test was remounted. A sync was performed, and the kernel was instructed to drop caches on all the clients and BeeGFS servers (MDS and SS) with the following commands:

```
sync && echo 3 > /proc/sys/vm/drop_caches
```

In measuring the solution performance, we performed all tests with similar initial conditions. The file system was configured to be fully functional and the targets tested were emptied of files and directories before each test.

4.1.1 IOzone sequential N-N reads and writes

To evaluate sequential reads and writes, we used IOzone benchmark version 3.487 in the sequential read and write mode. We conducted the tests on multiple thread counts, starting at one thread and increasing in powers of two to 1,024 threads. Because this test works on one file per thread, at each thread count, the number of files equal to the thread count were generated. The threads were distributed across eight physical client nodes in a round-robin fashion.

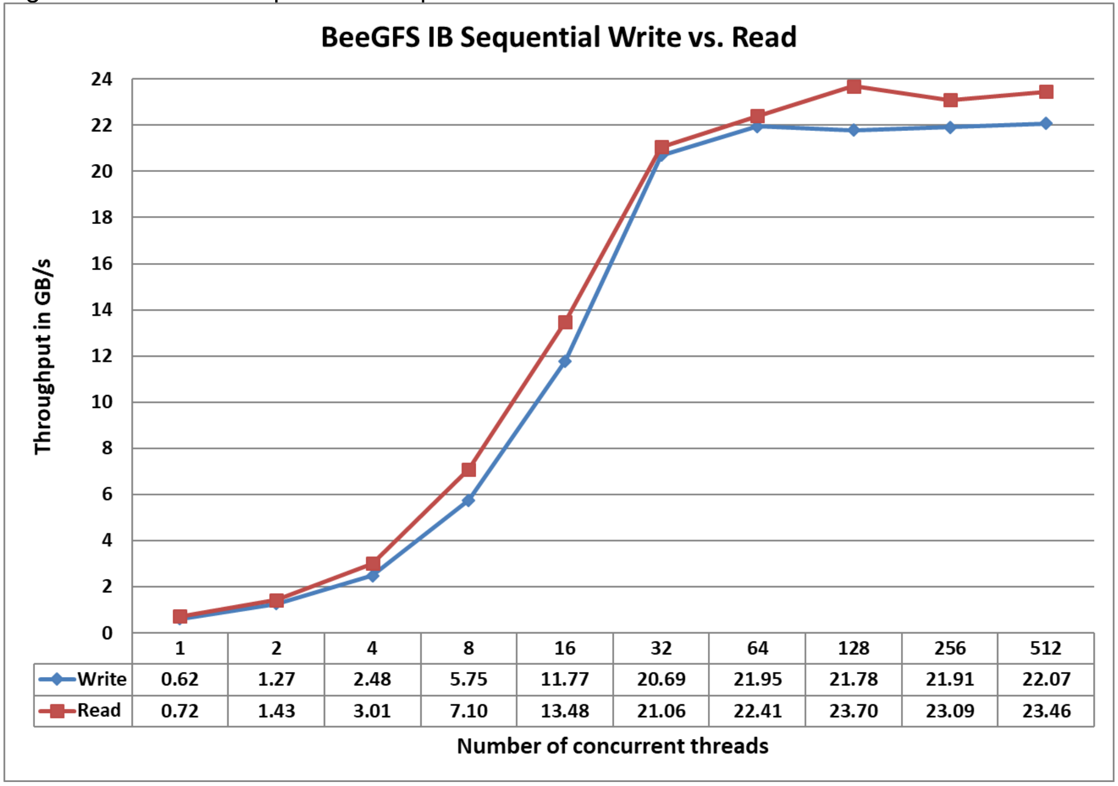
We converted throughput results to GB/s from the KB/s metrics that were provided by the tool. For thread counts 16 and above, an aggregate file size of 8 TB was chosen to minimize the effects of caching from the servers as well as from BeeGFS clients. For thread counts below 16, the file size is 768 GB per thread (i.e. 1.5 TB for two threads, 3 TB for four threads and 6 TB for eight threads). Within any given test, the aggregate file size used was equally divided among the number of threads. A record size of 1 MB was used for all runs. Operating system caches were also dropped or cleaned on the client nodes between tests and iterations and between writes and reads.

The files that were written were distributed evenly across the STs (round-robin) to prevent uneven I/O loads on any single SAS connection or ST, in the same way that a user would expect to balance a workload.

The default stripe count for BeeGFS is four. However, the chunk size and the number of targets per file (stripe count) can be configured on a per-directory or per-file basis. For all these tests, BeeGFS stripe size was set to 1 MB and stripe count was set to 1 as shown below:

```
$beegfs-ctl --setpattern --numtargets=1 --chunksize=1m /mnt/beegfs/benchmark
$beegfs-ctl --getentryinfo --mount=/mnt/beegfs/ /mnt/beegfs/benchmark/ --verbose
   Entry type: directory
   EntryID: 1-5E72FAD3-1
   ParentID: root
   Metadata node: metaA-numa0-1 [ID: 1]
   Stripe pattern details:
   + Type: RAID0
   + Chunksize: 1M
   + Number of storage targets: desired: 1
   + Storage Pool: 1 (Default)
Inode hash path: 61/4C/1-5E72FAD3-1
```

Figure 7 shows the sequential N-N performance of the solution:



	1	2	4	8	16	32	64	128	256	512
Write	0.62	1.27	2.48	5.75	11.77	20.69	21.95	21.78	21.91	22.07
Read	0.72	1.43	3.01	7.10	13.48	21.06	22.41	23.70	23.09	23.46

Figure 7 Sequential N-N read and write

As the figure shows, the peak read throughput of 23.70 GB/s was attained at 128 threads. The peak write was 22.07 GB/s at 512 threads. The single thread write performance was 623 MB/s and read performance was 717 MB/s. The read and write performance scale linearly with the increase in the number of threads until the system attained its peak. After this, we see that reads and writes saturate as we scale. This brings us to understand that the overall sustained performance of this configuration for reads is ≈ 23GB/s and that for the writes is ≈ 22 GB/s with the peaks as mentioned above. The reads are very close to or slightly higher than the writes, independent of the number of the threads used.

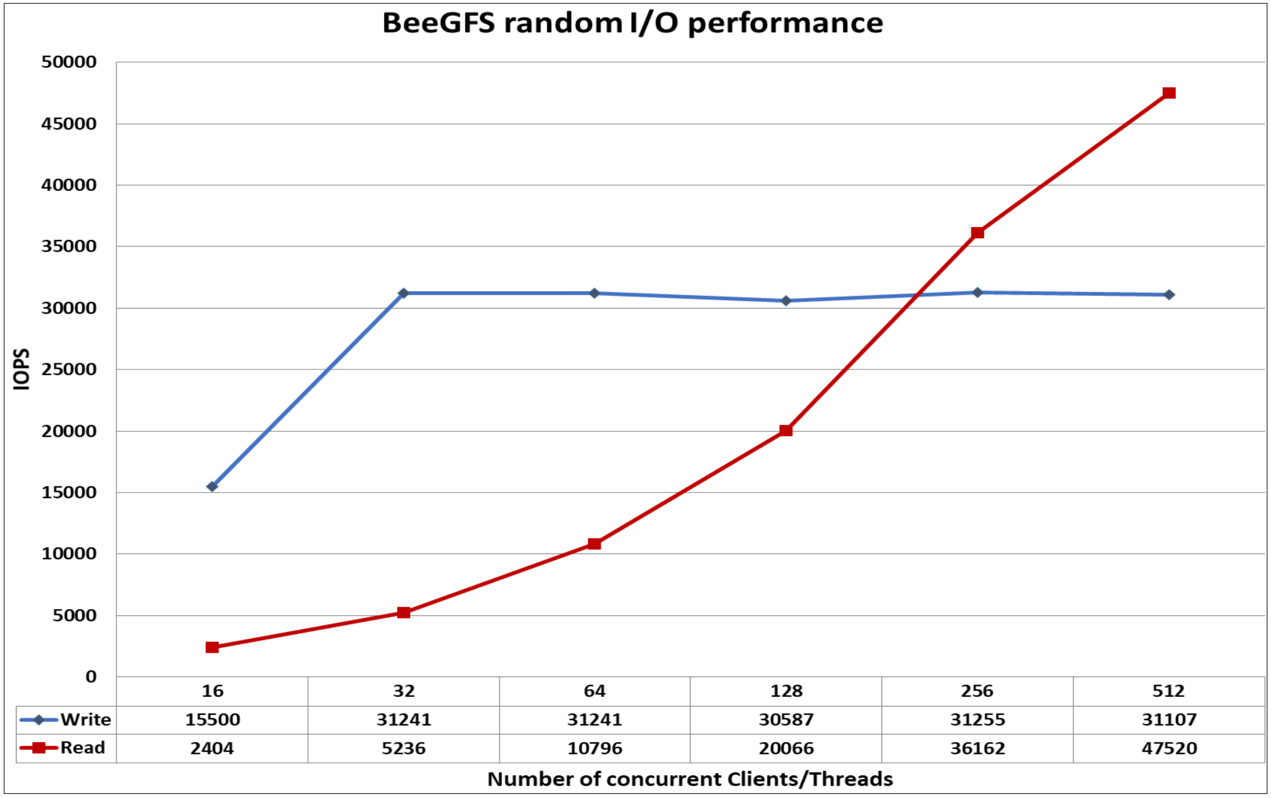
4.1.2 Random reads and writes

To evaluate random I/O performance, we used IOzone version 3.487 in random mode. Tests were conducted on thread counts from 16 to 512 threads. Direct IO option (-I) was used to run IOzone so that all operations bypassed the buffer cache and went directly to the disk.

As described in the IOzone sequential N-N reads and writes, stripe count of 1 and chunk size of 1 MB was used. The files that written were distributed evenly across the STs (round-robin) to prevent uneven I/O loads on any single SAS connection or ST in the same way that a user would expect to balance a workload.

The request size was set to 4KiB. Performance was measured in I/O operations per second (IOPS). The operating system caches were dropped between the runs on the BeeGFS servers. The file system was unmounted and remounted on clients between iterations of the test.

The following figure shows the random read and write performance.



	16	32	64	128	256	512
Write	15500	31241	31241	30587	31255	31107
Read	2404	5236	10796	20066	36162	47520

Figure 8 Random N-N reads and writes

As the figure shows, the write performance reaches around 31K IOPS and remains stable from 32 threads to 512 threads. In contrast, the read performance increases with the increase in the number of IO requests with a maximum performance of around 47K IOPS at 512 threads, which is the maximum number of threads tested for the solution. ME4 requires a higher queue depth to reach the maximum read performance and the graph indicates that the performance may continue to increase with more than 512 threads. However, as the tests were run only with 8 clients, we did not have enough cores to run more than 512 threads.

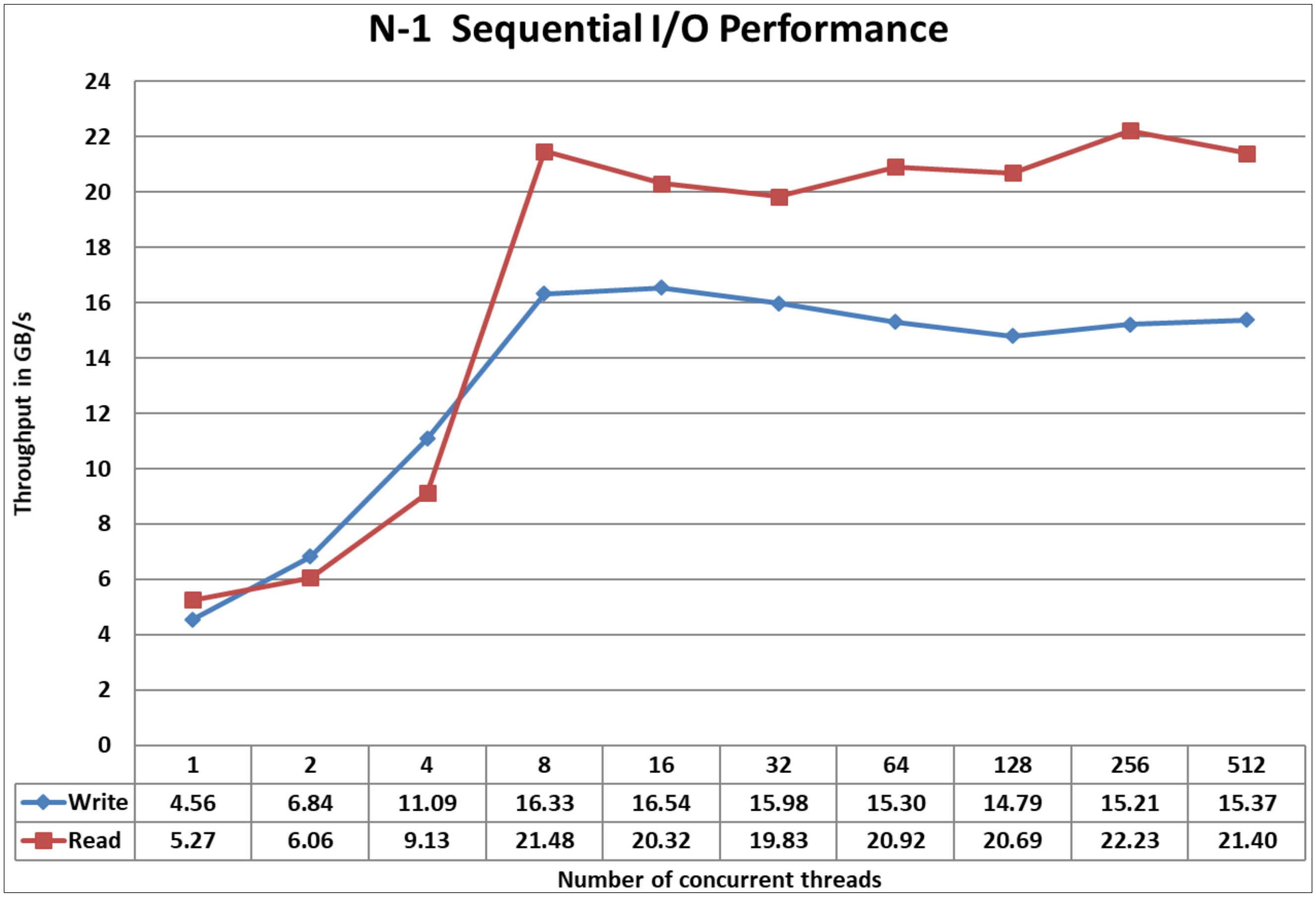
4.1.3 IOR N-1

The performance of sequential reads and writes with N threads to a single shared file was measured with IOR version 3.3.0+dev, assisted by OpenMPI-4.0.2rc3. The benchmark was run over eight compute nodes. Tests executed varied from single thread up to 512 threads.

We converted throughput results to GB/s from the MiB/s metrics that were provided by the tool. For thread counts eight and above an aggregate file size of 6 TiB was chosen to minimize the effects of caching. For thread counts below eight, the file size is 768 GiB per thread (i.e. 1.5 TiB for two threads and 3 TiB for four threads). Within any given test, the aggregate file size used was equally divided among the number of threads. A stripe count of 32 and transfer size of 8 MB was used.

The following commands were used to execute the benchmark for writes and reads, where `$threads` (the variable for the number of threads used) was incremented in powers of two. The transfer size is 8 M and each thread wrote to or read 128 G from a single file. Three iterations of each test have been run and the mean value has been recorded. Figure 9 shows the N to 1 sequential I/O performance. Use the following command to run the test:

```
mpirun --allow-run-as-root -machinefile $hostlist --map-by node -np $threads
~/bin/ior -w -r -i 3 -t 8m -b $file_size -g -d 3 -e -E -k -o
$working_dir/ior.out -s 1
```



	1	2	4	8	16	32	64	128	256	512
Write	4.56	6.84	11.09	16.33	16.54	15.98	15.30	14.79	15.21	15.37
Read	5.27	6.06	9.13	21.48	20.32	19.83	20.92	20.69	22.23	21.40

Figure 9 N-1 Sequential Performance

From the results we can observe that performance rises with the number of clients used and then reaches a plateau that is semi-stable for reads and writes all the way to the maximum number of threads used on this test. Therefore, large, single-shared file sequential performance is stable even for 512 concurrent clients. The maximum read performance was 22.23 GB/s at 256 threads. The maximum write performance of 16.54 was reached at 16 threads.

4.1.4 Metadata performance

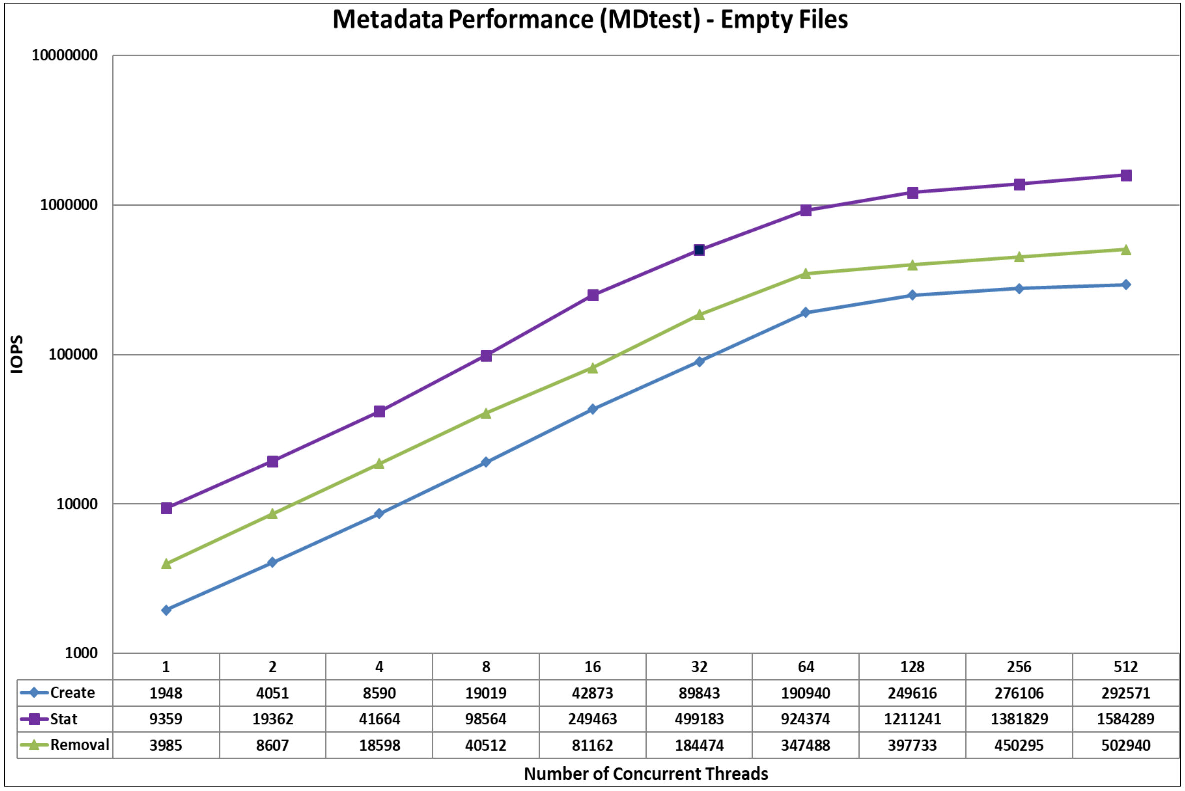
Metadata testing measures the time to complete certain file or directory operations that return attributes. MDtest is an MPI-coordinated benchmark that performs create, stat, and remove operations on files or directories. The metric reported by MDtest is the rate of completion in terms of OP/s. MDtest can be configured to compare metadata performance for directories and files. The benchmark was used for files only (no directories metadata), reporting the number of creates, stats, and removes the solution can handle. MDtest 3.3.0+dev with OpenMPI version 4.0.2rc3 was used to run the benchmark across the 8 BeeGFS clients that are described in Table 2.

To understand how well the system scales and to compare the different thread cases on similar ground, we tested from a single-thread case up to a 512-thread case with a consistent 2,097,152 file count for each case. The following table details the number of files per directory and the number of directories per thread for every thread count. We ran three iterations of each test and recorded the mean values.

Table 3 MDtest files and directory distribution across threads:

# of threads	# of files per directory	# of directories per thread	Total number of files
1	1024	2048	2,097,152
2	1024	1024	2,097,152
4	1024	512	2,097,152
8	1024	256	2,097,152
16	1024	128	2,097,152
32	1024	64	2,097,152
64	1024	32	2,097,152
128	1024	16	2,097,152
256	1024	8	2,097,152
512	1024	4	2,097,152

Figure 10 shows file metadata statistics for empty files:



	1	2	4	8	16	32	64	128	256	512
Create	1948	4051	8590	19019	42873	89843	190940	249616	276106	292571
Stat	9359	19362	41664	98564	249463	499183	924374	1211241	1381829	1584289
Removal	3985	8607	18598	40512	81162	184474	347488	397733	450295	502940

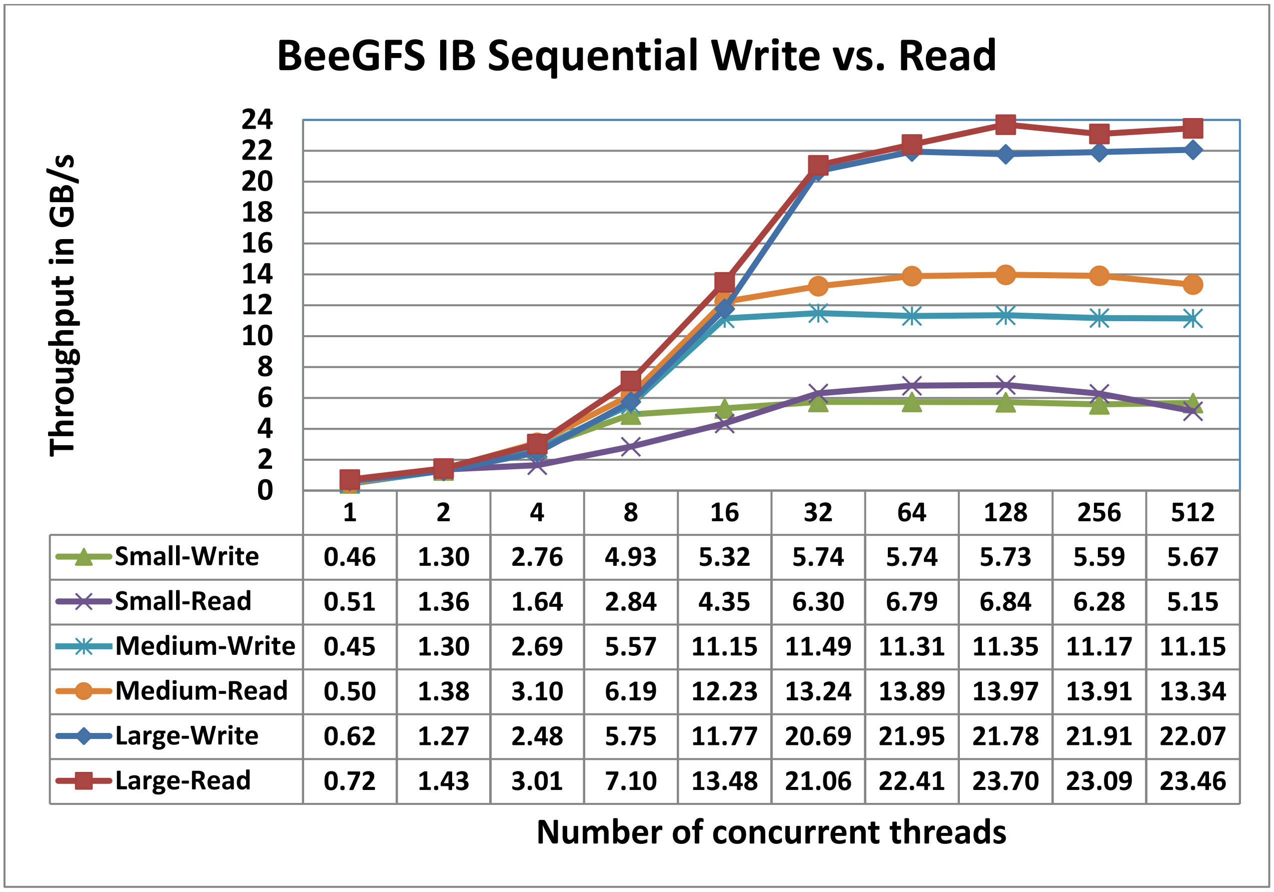
Figure 10 Metadata performance

First, notice that the scale chosen was logarithmic with base 10 to allow comparing operations that have differences several orders of magnitude. Otherwise some of the operations would look like a flat line close to 0 on a linear scale.

The system gets very good results with file stat operations reaching their peak value at 512 threads with 1.58M OP/s. The peak file create was approximately 292K OP/s and file removal at 503K OP/s.

4.2 Base configurations

Figure 11 shows the measured sequential read and write performance of the Small, Medium and Large configurations (base configurations) of the Dell EMC Ready Solution for HPC BeeGFS High Capacity Storage.



	1	2	4	8	16	32	64	128	256	512
▲ Small-Write	0.46	1.30	2.76	4.93	5.32	5.74	5.74	5.73	5.59	5.67
× Small-Read	0.51	1.36	1.64	2.84	4.35	6.30	6.79	6.84	6.28	5.15
✳ Medium-Write	0.45	1.30	2.69	5.57	11.15	11.49	11.31	11.35	11.17	11.15
● Medium-Read	0.50	1.38	3.10	6.19	12.23	13.24	13.89	13.97	13.91	13.34
◆ Large-Write	0.62	1.27	2.48	5.75	11.77	20.69	21.95	21.78	21.91	22.07
■ Large-Read	0.72	1.43	3.01	7.10	13.48	21.06	22.41	23.70	23.09	23.46

Figure 11 Measured IOzone sequential performance of base configurations

The small configuration using a single ME4084 array shows a peak read performance of 6.84 GB/s and peak write performance of 5.74 GB/s. We observe a linear increase in performance as we add ME4084 arrays from Small to Medium to Large configurations.

4.3 Scalable configurations

For the rest of the configurations, the performance numbers shown in Table 4 are estimates or extrapolations, since scaling up is linear with addition of ME4084 arrays and scaling down by removing arrays is assumed to be linear as well.

Table 4 Performance and usable space of base and scalable configurations

Configuration		**Small(S)**	**Medium(M)**	**Large(L)**	**L + S**	**L + M**	**2x L**
Total U (MDS+SS)		18U	23U	33U	42U	47U	58U
# of ME4084		1	2	4	5	6	8
Estimated usable space	4TB	231 TiB	461 TiB	922 TiB	1153 TiB	1383 TiB	1844 TiB
	8TB	461 TiB	922 TiB	1844 TiB	2305 TiB	2766 TiB	3688 TiB
	12TB	691 TiB	1383 TiB	2766 TiB	3458 TiB	4149 TiB	5532 TiB
	16TB	922 TiB	1844 TiB	3688 TiB	4610 TiB	5532 TiB	7376 TiB
Peak sequential read		≈ 6.8 GB/s	≈ 13.9 GB/s	≈ 23.7 GB/s	≈ 30.5 GB/s	≈ 37.6 GB/s	≈ 47.4 GB/s
Peak sequential		≈ 5.7 GB/s	≈ 11.5 GB/s	≈ 22.1 GB/s	≈ 27.8 GB/s	≈ 33.6 GB/s	≈ 44.2 GB/s
Sustained performance		≈ 5 GB/s	≈ 11 GB/s	≈ 22 GB/s	≈ 27 GB/s	≈ 33 GB/s	≈ 44 GB/s

The sustained performance is the steady state performance of the solution stack over a longer period or for more thread counts after the saturation has been attained. The more conservative approach would be to consider the sustained performance to size a system rather than occasional peaks. But the peak performance of the system as such shows the maximum extent to which the system could be pushed in terms of performance or in other words the point where the performance hits the bottleneck and cannot grow any further. All these numbers shown assume minimized caching effects since they are based off actual measurements from the base configurations which were performed by minimizing caching effects.

BeeGFS usable space was calculated using the formula:

```
Estimate of BeeGFS Usable Space in TiB = 0.99 * Number of ME4084 arrays * 80
RAID HDDs per array * 0.8 * HDD size in TB * 10^12/2^40
```

Table 4 uses this formula to estimate the usable space of every configuration for each supported capacity of the 7.2K RPM NL SAS HDDs.

The usable space is calculated in TiB. 0.99 is the factor considering the 1% overhead from the file system, a conservation assumption. 80 is the number of HDDs per ME4084 excluding the hot spares. 0.8 is 80% of the RAID 6 (8 + 2) HDDs being the data drives. The remaining 20% in the RAID volume are parity drives and are not taken into consideration for usable space. The last factor in the fomula 10^12/2^40 is to convert the usable space from TB to TiB.

4.4 Performance tuning

Multiple parameters can be configured to achieve optimal system performance depending on intended workload patterns. This section shows the tuning parameters that we configured on the BeeGFS testbed system in the Dell HPC and AI Innovation lab.

- Set the number of processes for the superuser to 50000 in order to improve performance.
- Tuned the IO scheduler settings for the storage block devices on the storage servers by adding the following lines to /etc/rc.local and make /etc/rc.local executable afterwards:

```
for mdev in /dev/mapper/storage* ; do
dev=$(basename $(readlink -f "$mdev"))
echo "$dev"
echo deadline > /sys/block/${dev}/queue/scheduler
echo 2048 > /sys/block/${dev}/queue/nr_requests
echo 4096 > /sys/block/${dev}/queue/read_ahead_kb
echo 256 > /sys/block/${dev}/queue/max_sectors_kb
done
$ chmod +x /etc/rc.local
```

- Tuned the IO scheduler settings for the metadata block devices on the metadata servers by adding the following lines to /etc/rc.local and make /etc/rc.local executable afterwards:

```
for mdev in /dev/mapper/storage* ; do
dev=$(basename $(readlink -f "$mdev"))
echo "$dev"
echo deadline > /sys/block/${dev}/queue/scheduler
echo 128 > /sys/block/${dev}/queue/nr_requests
echo 128 > /sys/block/${dev}/queue/read_ahead_kb
echo 256 > /sys/block/${dev}/queue/max_sectors_kb
done
$ chmod +x /etc/rc.local
```

- Disabled transparent huge pages by creating `/etc/tmpfiles.d/90-beegfs-hugepages.conf` file with the following content:

```
# Recommended configuration for BeeGFS servers
# Disable transparent hugepages
# Type  Path                            Mode    UID     GID     Age
Argument w   /sys/kernel/mm/transparent_hugepage/khugepaged/defrag  -  -
-  -       0
w   /sys/kernel/mm/transparent_hugepage/defrag -   -   -       -
never
w   /sys/kernel/mm/transparent_hugepage/enabled  -     -   -   -
never
```

- Tuned virtual memory settings by adding the following lines to `/etc/sysctl.d/90-beegfs.conf`:

```
# VM ratios recommended for BeeGFS
vm.dirty_background_ratio = 5
vm.dirty_ratio = 20
vm.min_free_kbytes = 262144x
vm.vfs_cache_pressure = 50
```

- The following BeeGFS specific tuning parameters were used in the metadata, storage, and client configuration files:

```
beegfs-meta.conf
   connMaxInternodeNum = 64
   tuneNumWorkers = 12
   tuneUsePerUserMsgQueues = true # Optional
   tuneTargetChooser = roundrobin (benchmarking)
beegfs-storage.conf
   connMaxInternodeNum = 64
   tuneNumWorkers = 12
   tuneUsePerTargetWorkers = true
   tuneUsePerUserMsgQueues = true # Optional
   tuneBindToNumaZone = 0
   tuneFileReadAheadSize = 2m
beegfs-client.conf
   connMaxInternodeNum = 24
   connBufSize = 720896
```

Note: The `tuneTargetChooser` parameter was set to `roundrobin` for the purpose of benchmarking so that the targets are chosen in a deterministic, round-robin fashion. However, in a production system, it is recommended to use the “`randomized`” algorithm which chooses the targets in a random fashion.

5 Conclusion

The Dell EMC Ready Solution for HPC BeeGFS High Capacity Storage is a high-performance clustered file system solution that is easy to manage, fully supported, and capable of scaling both throughput and capacity. The solution includes the PowerEdge server platform, PowerVault ME4 storage products, and BeeGFS technology, the leading open-source solution for a parallel file system. The large size solution stack with 2.69 PB of raw storage space has shown to sustain a sequential throughput of approximately 22 GB/s, which is consistent with the needs of HPC environments. HDR100 has also been vetted as a network interconnect for the solution.

6 References

The following Dell EMC documentation provides additional and relevant information. Access to these documents depends on your login credentials. If you do not have access to a document, contact your Dell EMC representative.

- Dell EMC ME4 Series Storage System Administrator’s Guide
- Dell EMC Ready Solution for HPC BeeGFS High Capacity Storage

A Appendix A Storage array cabling

This section presents how the metadata and storage servers are cabled to the PowerVault ME storage array.

A.1 ME4024 cabling

Table 5 shows the 12 Gb/s SAS cable connections between the MDS pair and one ME4024 that hosts the MDTs.

Table 5 Cabling metadata servers to the ME4024 array

Server	SAS PCI slot	SAS port	ME4024 controller	ME4024 controller port
metaA	Slot 1	Port 0	Controller 0	Port 3
metaA	Slot 4	Port 0	Controller 1	Port 1
metaB	Slot 1	Port 0	Controller 0	Port 1
metaB	Slot 4	Port 0	Controller 1	Port 3

Figure 12 illustrates the storage cabling with one port on each SAS controller connected to one storage array RAID controller. This provides redundancy for the SAS controllers as well as at the RAID controllers

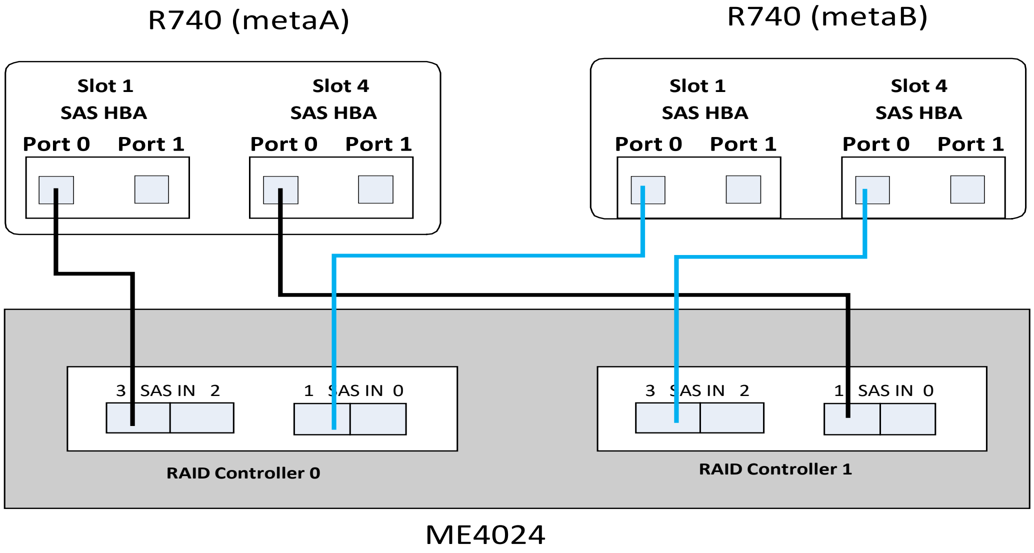


Figure 12 Cabling metadata servers to the ME4024 storage array

A.1.1 ME4084 cabling

This section shows the storage cabling for the three base configurations, small, medium and large.

Small configuration

Figure 13 shows how the storage servers are cabled in the small configuration with a single SS pair (a pair of Dell EMC PowerEdge R740s), attached to a single fully-populated ME4084 array.

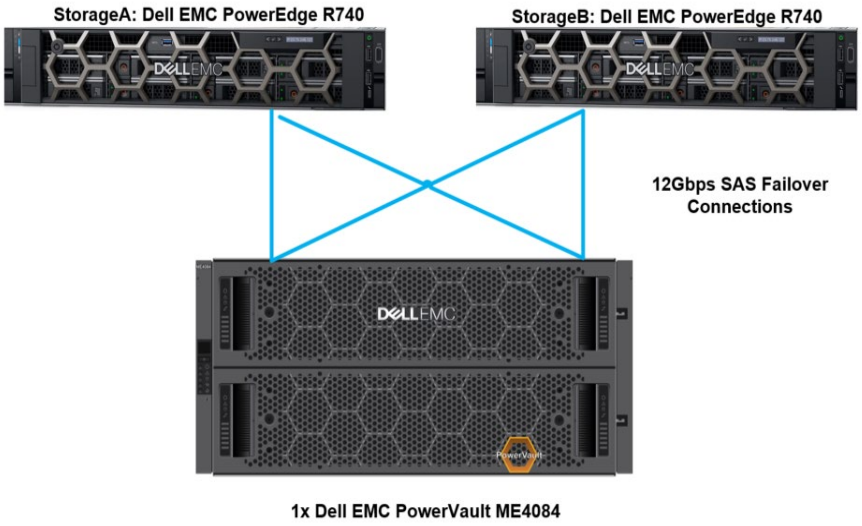

StorageA: Dell EMC PowerEdge R740
StorageB: Dell EMC PowerEdge R740
12Gbps SAS Failover Connections
1x Dell EMC PowerVault ME4084

Figure 13 Storage Cabling of Small Configuration with 1x ME4084 array

Medium configuration

The next size up from the small configuration is the medium configuration which uses a pair of storage servers (a pair of R740s) attached to two fully populated ME4084 arrays as shown in Figure 14.

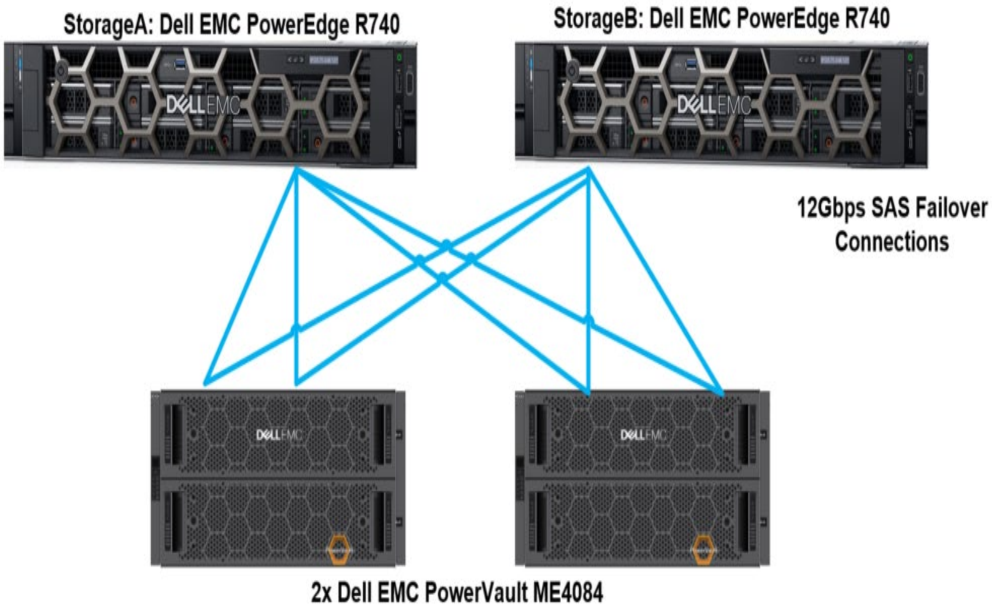

StorageA: Dell EMC PowerEdge R740
StorageB: Dell EMC PowerEdge R740
12Gbps SAS Failover Connections
2x Dell EMC PowerVault ME4084

Figure 14 Storage Cabling of Medium Configuration with 2x ME4084 array

Large configuration

The following table details the 12 Gb/s SAS cable connections between the SS pair and four ME4084 arrays:

Table 6 Cabling of storage servers to the ME4084 arrays

Server	SAS PCI Slot	SAS port	ME4084 array	ME4084 Controller	ME4084 Controller Port
StorageA	Slot 1	Port 0	ME4084 #1	Controller 0	Port 3
StorageA	Slot 1	Port 1	ME4084 #2	Controller 0	Port 3
StorageA	Slot 2	Port 0	ME4084 #2	Controller 1	Port 3
StorageA	Slot 2	Port 1	ME4084 #1	Controller 1	Port 3
StorageB	Slot 1	Port 0	ME4084 #1	Controller 0	Port 1
StorageB	Slot 1	Port 1	ME4084 #2	Controller 0	Port 1
StorageB	Slot 2	Port 0	ME4084 #2	Controller 1	Port 1
StorageB	Slot 2	Port 1	ME4084 #1	Controller 1	Port 1
StorageA	Slot 4	Port 0	ME4084 #3	Controller 0	Port 3
StorageA	Slot 4	Port 1	ME4084 #4	Controller 0	Port 3
StorageA	Slot 5	Port 0	ME4084 #4	Controller 1	Port 3
StorageA	Slot 5	Port 1	ME4084 #3	Controller 1	Port 3
StorageB	Slot 4	Port 0	ME4084 #3	Controller 0	Port 1
StorageB	Slot 4	Port 1	ME4084 #4	Controller 0	Port 1
StorageB	Slot 5	Port 0	ME4084 #4	Controller 1	Port 1
StorageB	Slot 5	Port 1	ME4084 #3	Controller 1	Port 1

Figure 15 illustrates the storage cabling:

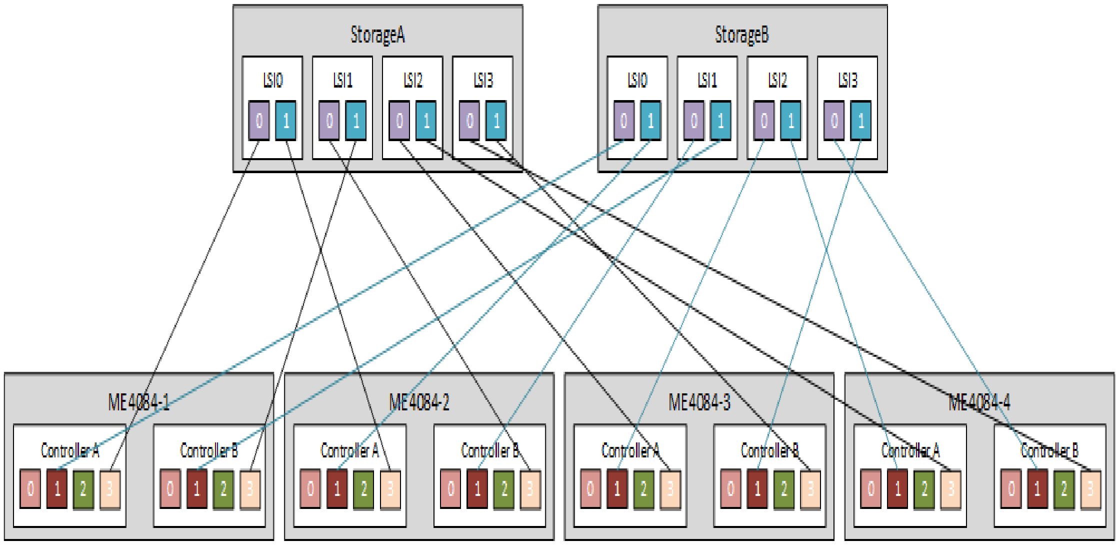


Figure 15 Cabling Storage Servers to the 4x ME4084 Storage Arrays

A simplified version of the storage cabling for the large base configuration is shown in Figure 16:

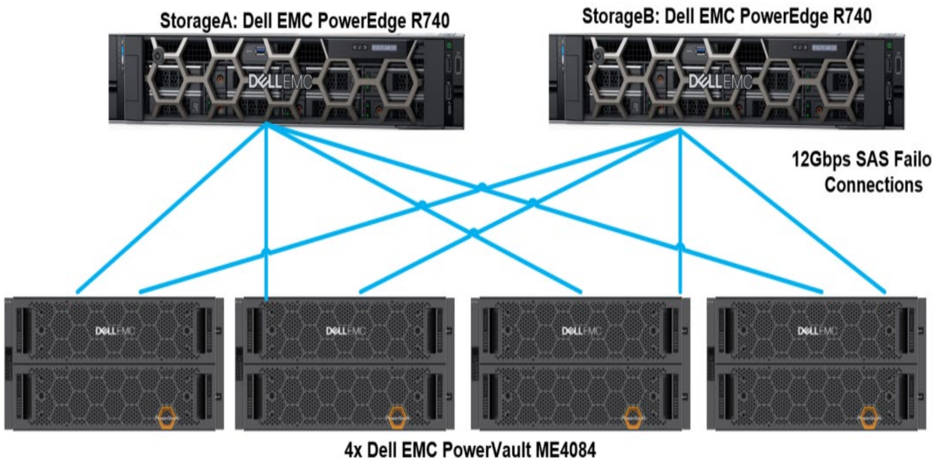

StorageA: Dell EMC PowerEdge R740
StorageB: Dell EMC PowerEdge R740
12Gbps SAS Failover Connections
4x Dell EMC PowerVault ME4084

Figure 16 Simplified depiction of cabling Storage Servers to the 4x ME4084 Storage Arrays

B Appendix B Benchmark command reference

This section describes the commands that were used to benchmark the Dell HPC BeeGFS Storage solution.

B.1 IOzone N-1 sequential and random IO

We used the following commands to run sequential and random IOzone tests, the results of which are recorded in the performance evaluation section of this paper.

For sequential writes:

```
iozone -i 0 -c -e -w -r 1024K -s $Size -t $Thread -+n -+m /path/to/threadlist
```

For sequential reads:

```
iozone -i 1 -c -e -w -r 1024K -s $Size -t $Thread -+n -+m  /path/to/threadlist
```

For IOPS random reads/writes:

```
iozone -i 2 -w -c -O -I -r 4K -s $Size -t $Thread -+n -+m /path/to/threadlist
```

The following table describes the IOzone command line options. The O_Direct command line option, `-I`, enables us to bypass the cache on the compute nodes where the IOzone threads are running.

Table 7 IOzone Command Line Options

Option	Description
-i 0	Write test
-I 1	Read test
-I 2	Random IOPS test
-+n	No retest
-c	Includes close in the timing calculations
-e	Includes flush in the timing calculations
-r	Records size
-s	File size
-+m	Location of clients to run IOzone on when in clustered mode
-I	Use O_Direct
-w	Does not unlink (delete) temporary file
-O	Return results in OPS

B.2 MDtest: Metadata file operations

We used the following command to run metadata tests, the results of which are recorded in the performance evaluation section of this paper.

```
mpirun --allow-run-as-root -machinefile $hostlist --map-by node -np $threads
$mdtest -v -d $working_dir -i ${repetitions} -b $nd -z 1 -L -I $nf -y -u -t -F
```

The following table describes the MDtest command line options:

Table 8 MDtest command line options

Option	Description
-d	Directory in which the tests will run
-v	Verbosity (each instance of option increments by one)
-i	Number of iterations that the test will run
-b	Branching factor, how many directories to create, used in conjunction with -z
-z	Depth of hierarchical directory structure
-L	Files only at leaf level of tree
-I	Number of files to create per directory
-y	Sync file after writing
-u	Unique working directory for each task
-t	Time unique working directory overhead
-F	Perform test on files only (no directories)

B.3 IOR N-1 xequential IO

We used the following command to run the IOR N-1 performance tests, which are recorded in the performance evaluation section of this paper:

```
mpirun --allow-run-as-root -machinefile $hostlist --map-by node -np $threads
~/bin/ior -w -r -i 3 -t 8m -b $file_size -g -d 3 -e -E -k -o
$working_dir/ior.out -s 1
```

The following table describes the IOR command line options:

Table 9 IOR command line options

Component	Specification
-g	intraTestBarriers—use barriers between open, write/read, and close
-d	interTestDelay—delay between reps in seconds

Component	Specification
-e	fsync—perform fsync upon POSIX write close
-E	useExistingTestFile—do not remove test file before write access
-k	keepFile—don’t remove the test file(s) on program exit
-o	testFile—full name for test
-s	segmentCount—number of segments