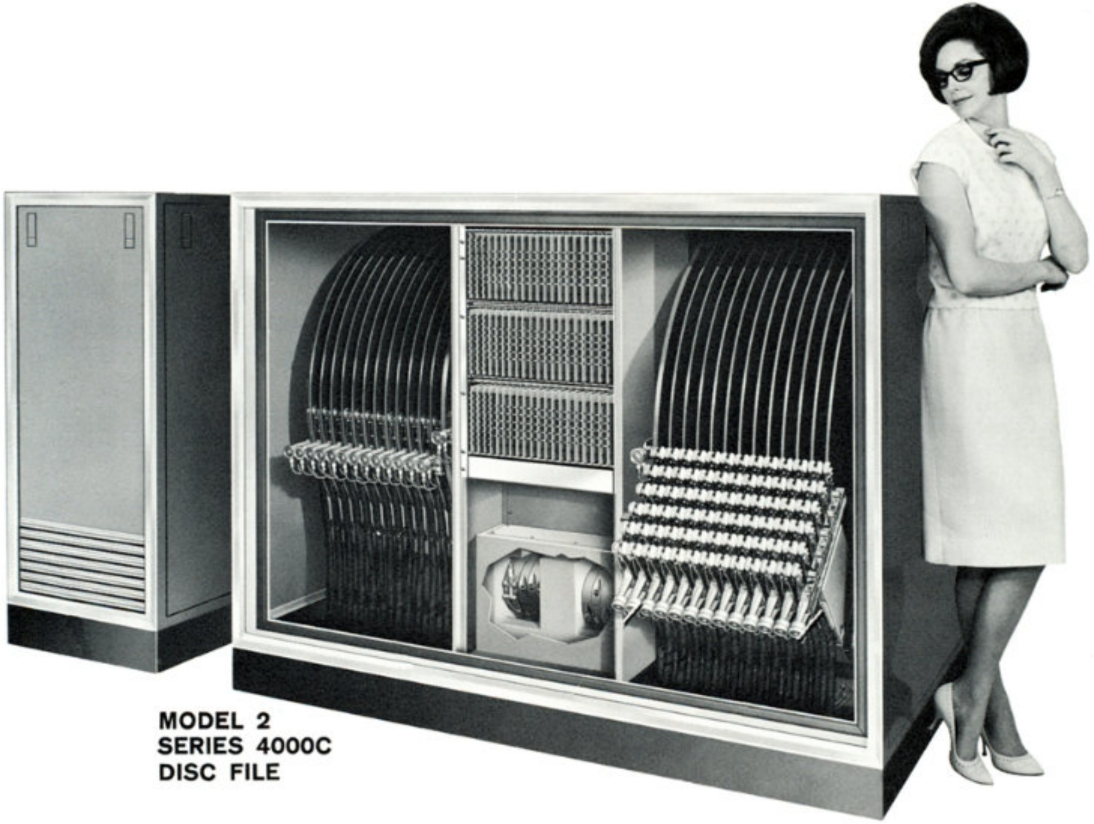
Accelerating Spectrum Scale with a Intelligent IO Manager

Ray Coetzee
Pre-Sales Architect
Seagate Systems Group, HPC



ClusterStor: Lustre, Spectrum Scale and Object

Vertically Integrated: From the RAW media, to the fastest systems in the world

L300/N Lustre System
›Up to 360 GB/s per rack
›Lustre 2.5 / 2.7

lustre

Secure Lustre
›Up to 60 GB/s per rack
›Lustre 2.5 on **SE-Linux**

lustre

G200,G300/N with ISS
›Up to 360 GB/s per rack
›IBM SS 4.2

IBM **Spectrum Storage**

A200 - Object Store
›Tiered Archive
›More than 5 PB per rack

CP-3584
›Up to 84 x 8 TB drives
›Dual Controllers

SP-3224
›24 x 2.5' drives or SSDs
›Dual Controllers

SP-3224
›24 x 8 TB drives
›Dual Controllers

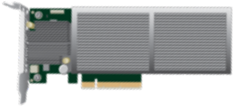
SAS
›NL SAS
›10 TB
›7.2K RPM

›HPC Drive
›4TB
›10K RPM

SSD
›SAS SSD
›3.2 TB

›NVMe
›1.3 TB

Flash accelerators



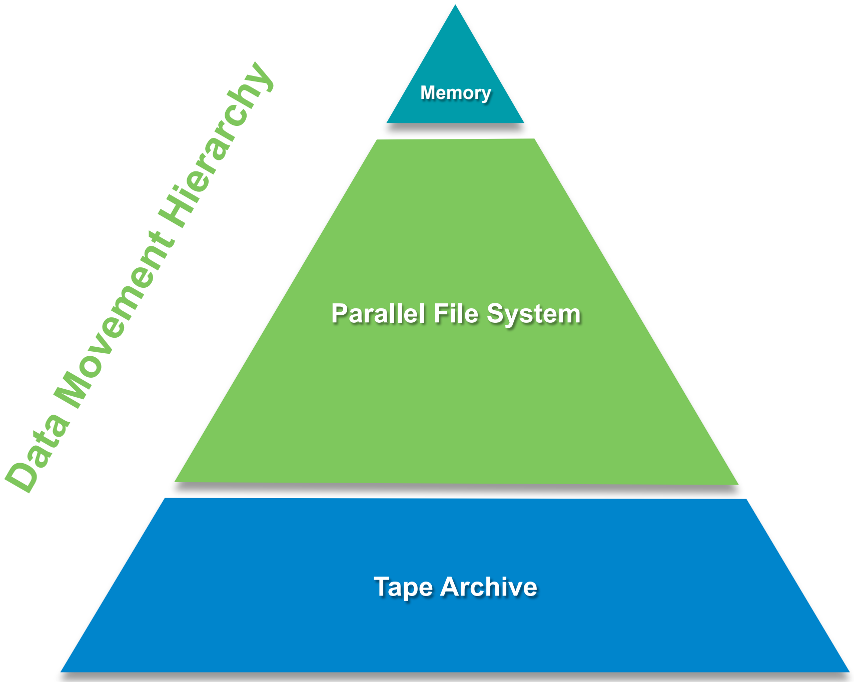
›PCIe x 16
›NVMe
›10 GB/s

SATA

›SMR Drive
›8TB

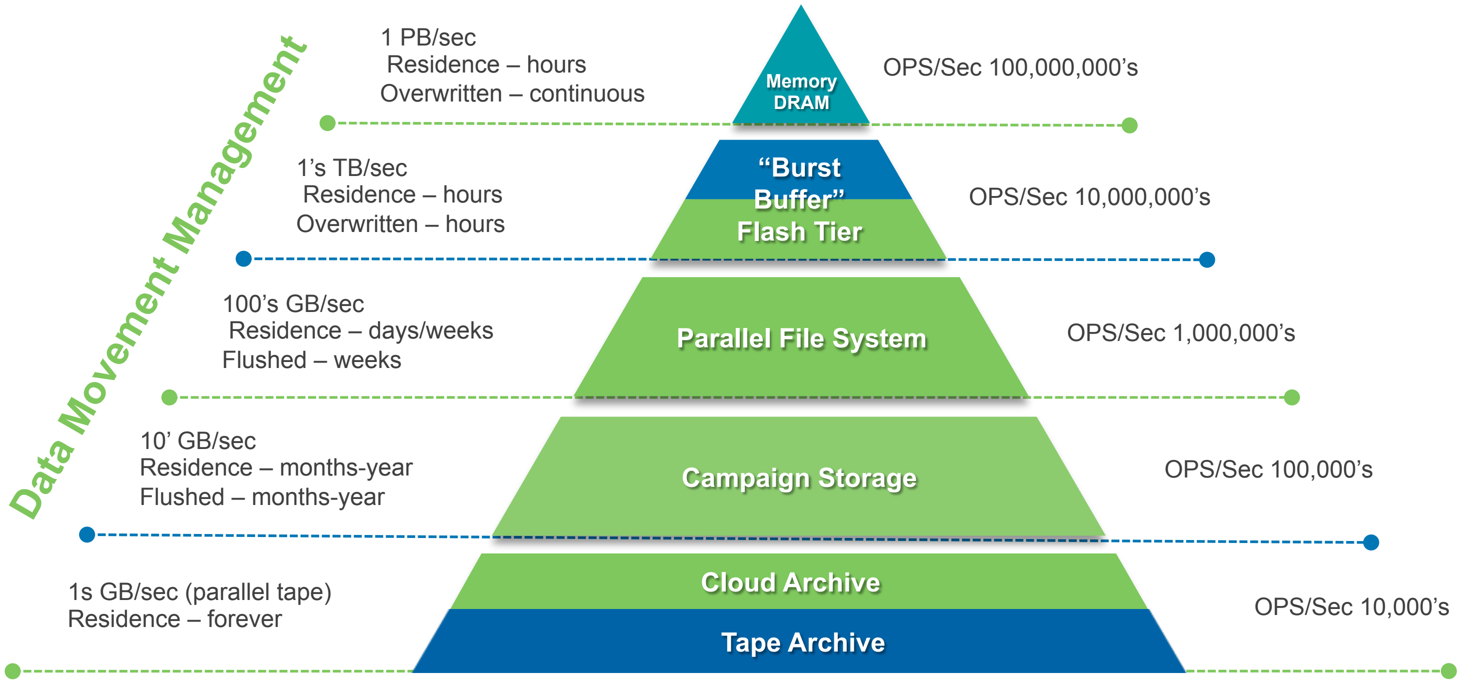
* File system performance (GB/s) per [HDD, RU, Enclosure, Rack ….]

HPC I/O Storage Stack is Transitioning



HPC I/O Storage Stack is Transitioning

Adoption of Flash and Object Storage Expands The I/O stack



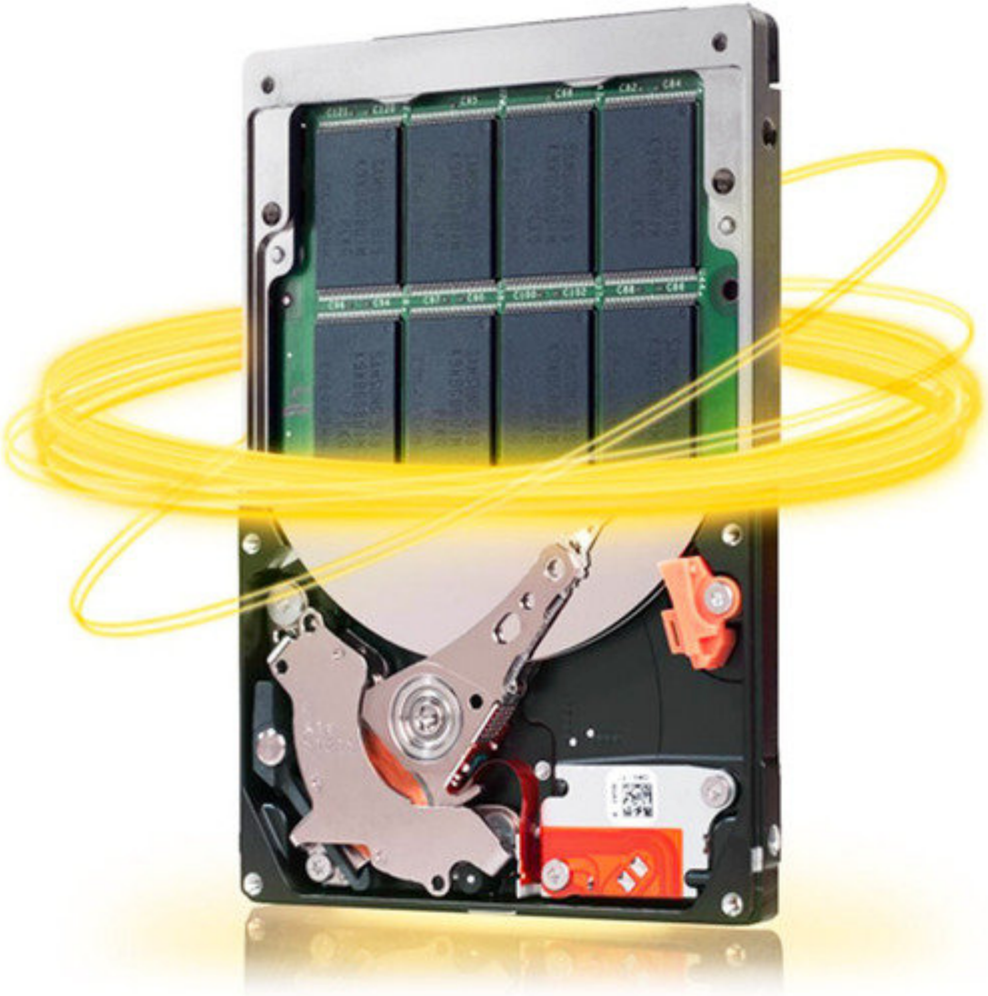
Flash Tiers come in many shapes …

Flash Acceleration	Options	Examples
Server side	Memory like	3DXpoint
	AIC / SSDs	NVMe / Nytro / Data Warp / LROC
Network attached	Tiered flash	IME, All Flash Array file system
Enhanced Storage	In File System	AFA, SSD pools in FS, HAWC
	Flash accelerated HDD tier	Seagate NytroXD, SSD based read cache

- All alternatives have advantages and drawbacks
- Most solutions have significant cost implications
- Actual value depends heavily on application capabilities
- **Bottom line, the jury is still out …..**

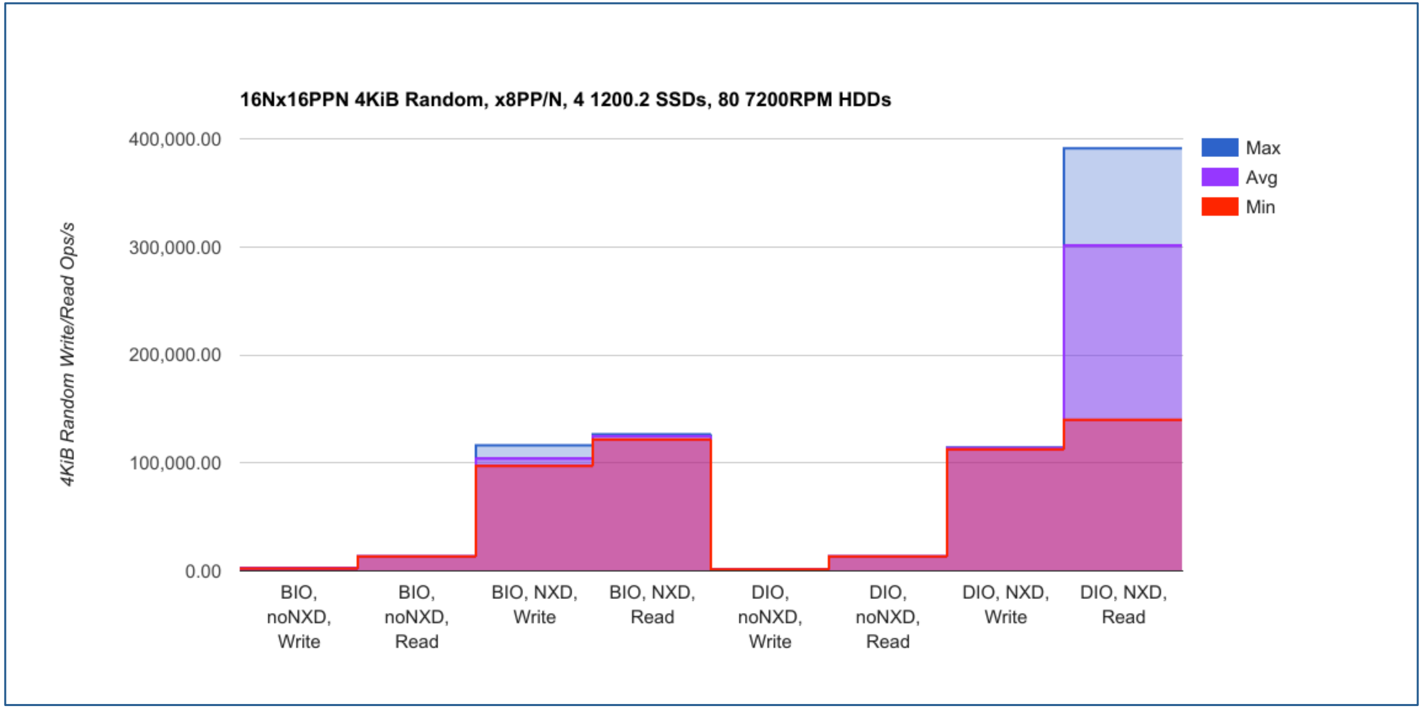
Storage side Flash Acceleration

Seagate style



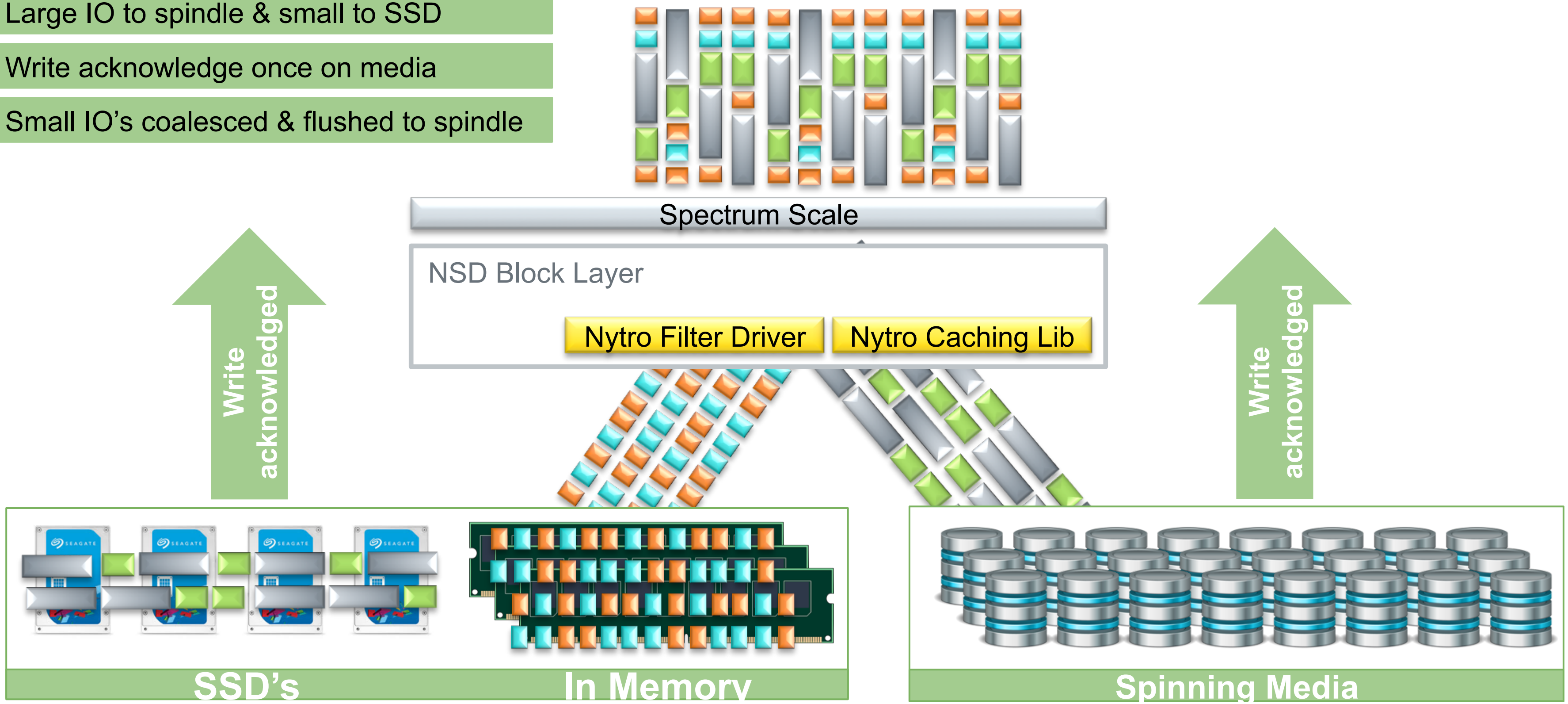
What's Possible With Just A Little "Invisible" Flash

Pre-Alpha Results



Basic Nytro IO Manager Data Path

1. Incoming IO are profiled & filtered
2. Large IO to spindle & small to SSD
3. Write acknowledge once on media
4. Small IO's coalesced & flushed to spindle



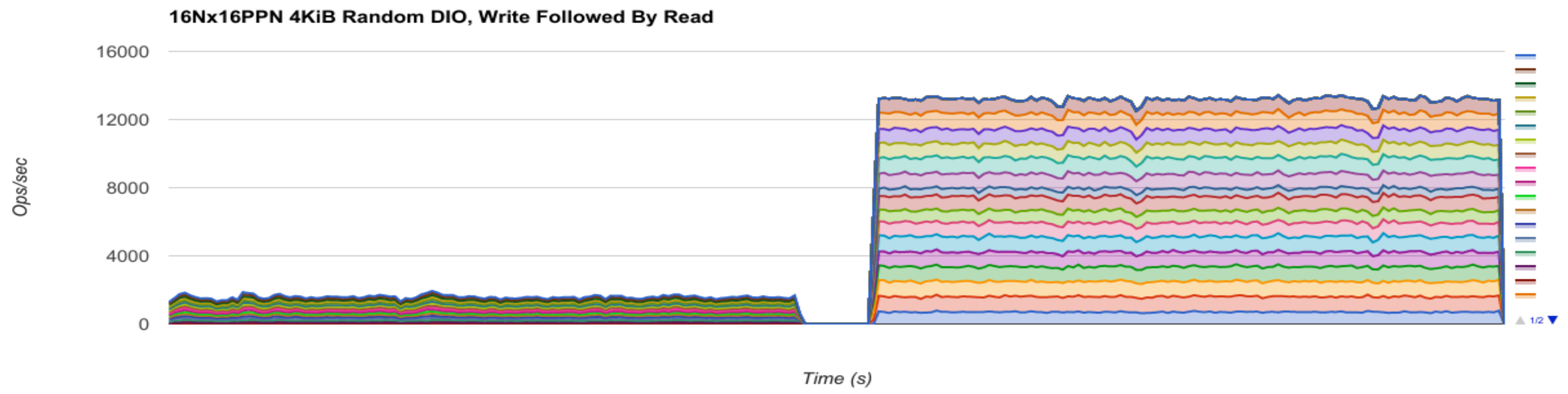
The test bench

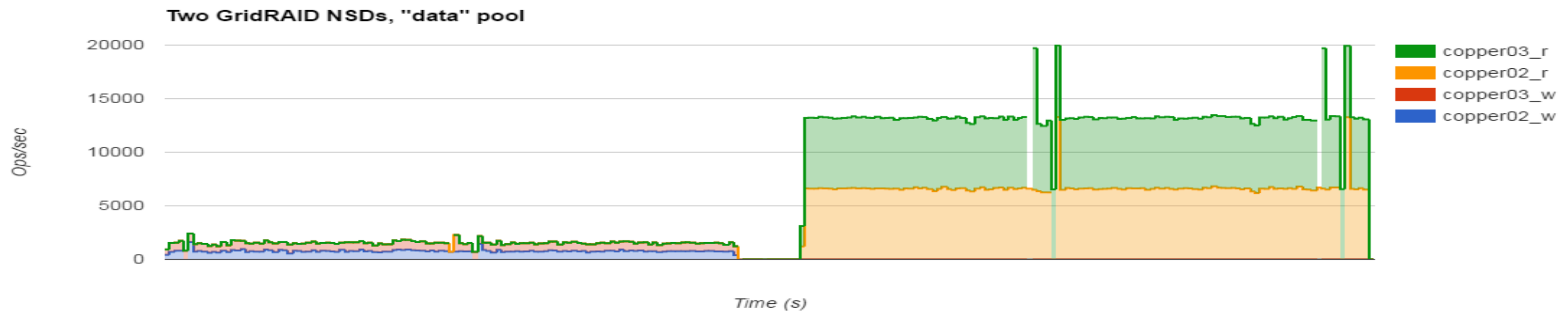
- All Spectrum Scale 4.2.3
  - Use 1GiB Pagepool
  - prefetchAggressiveness=0
- 16x dual socket E5-2630 v3 @ 2.40GHz w/ 64GB (8x8GB) DDR4
- Dataset sizes x1,x2,x3,x4 PP @ 3 iterations unless noted
- 2x GridRAID arrays, 40 NLSAS disks each
- All G300N Disk Drives have Write Cache Disabled
- GridRAID/HDD based NSDs use 32MB stripe cache
- 4x 1.6TB SSDs are RAID10, partitioned 50% system pool
- SSD’s used:
  - Random Read, 4KiB, QD32=200,000 IOPS <= 5us
  - Random Write, 4KiB, QD32=80,000 IOPS <= 12.5us
  - DWPD=10

How Many IOPS?

- First, define it, agree how to measure it
  - Ask many people, get many different answers
- Second, what is actually needed?
  - So many layers obscure the answer
    - What does the file system client see from the application?
      - Direct IO, vs. system call buffered IO (write), vs. buffered IO library (fwrite)
    - What does the RAID device and individual disks see?
      - Mixed, simultaneous use cases
      - “Advanced” read-ahead mechanics
      - DirectIO w/ HAWC
    - These considerations ALL also apply when designing an IOPS test
  - Answer changes with every new researcher, product, application, application version

16Nx16PPN 4KiB - No NXD



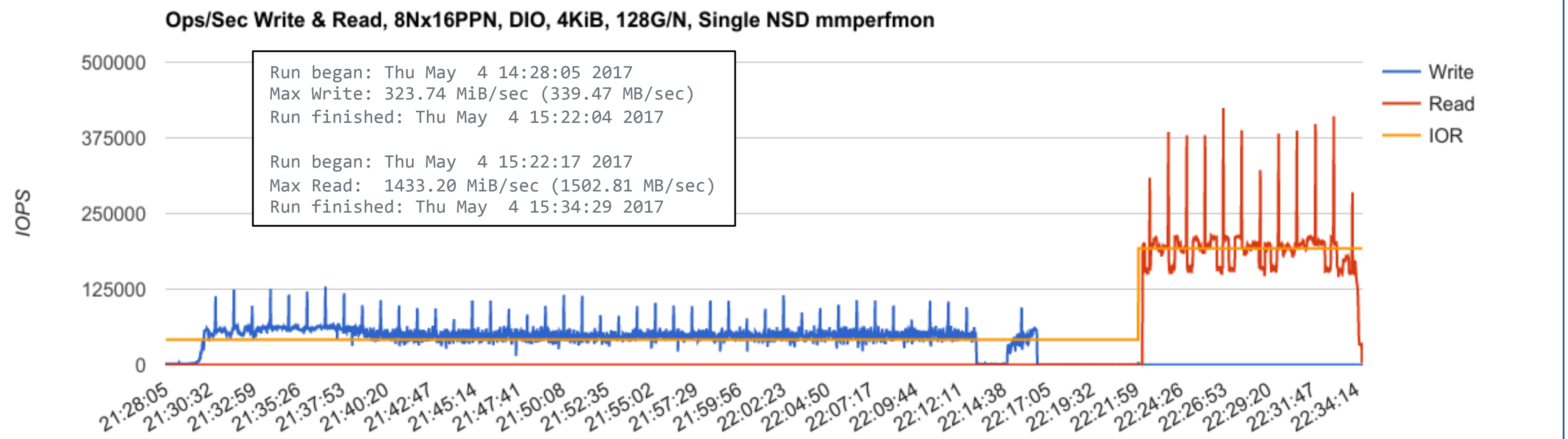


SEAGATE

8Nx16PPN, DIO, Random 4KiB xfer, 128GB/N (x2 client memory!!!) w/ NXD

```
Summary:
        api                 = POSIX
        test filename       = /mnt/copperfs//scratch/1493933271.0/out
        access              = file-per-process
        pattern             = strided (2097152 segments)
        ordering in a file = random offsets
        ordering inter file= no tasks offsets
        clients             = 128 (16 per node)
        repetitions         = 1
        xfersize            = 4096 bytes
        blocksize           = 4096 bytes
        aggregate filesize = 1024 GiB
```

	Start	After Write	Delta After W	After Read	Delta After R	
Cache Size in use	0	549755813888	549755813888	549755813888	0	
Total number of IOs	2645424241	2784385958	138961717	2914353110	129967152	
Number of reads	1507463413	1511715076	4251663	1641682228	129967152	
Number of writes	1137960828	1272670882	134710054	1272670882	0	
Total number of bypass IOs	655710496	656236614	526118	656236677	63	
Number of bypass reads	235763879	419947641	184183762	419947704	63	
Number of bypass writes	235763879	236288973	525094	236288973	0	
Number of Cache Hits	1716621799	1720871402	4249603	1850806759	129935357	99.98%
Number of Cache Misses	273091946	407277942	134185996	407309674	31732	
Number of dirty CWs	0	0	0	0	0	
Total Cache Blocks flushed	1017817720	1152002680	134184960	1152002680	0	



mmperfmon gpfsNumberOperations

SEAGATE

Sequential IO, RandomIO, Simultaneous, noNXD

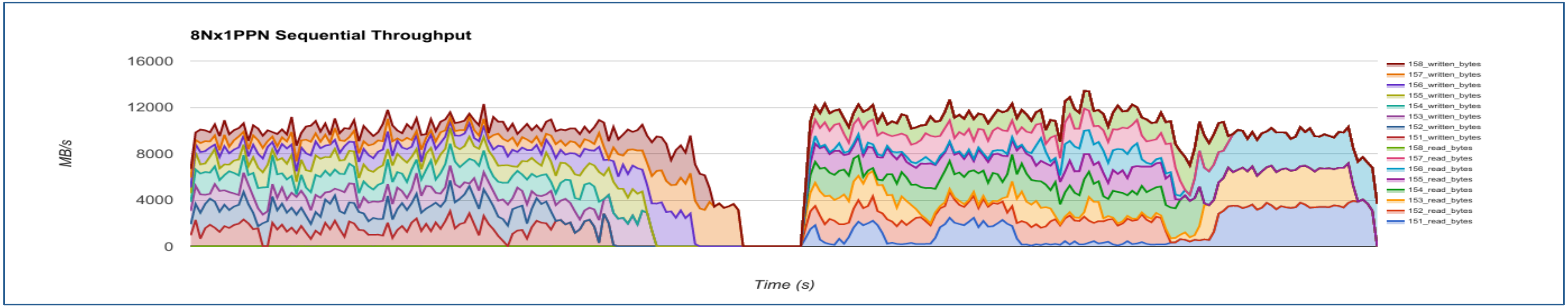
No NXD 2

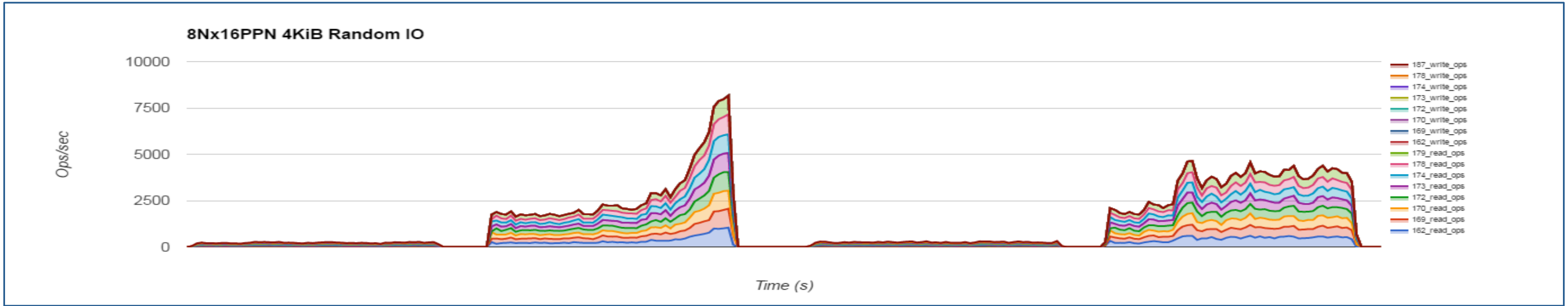
14:15:09 | 14:17:04 | 14:17:17 | 14:19:17

Sequential Write 8x1 1MiB 9590.25 MB/sec | Sequential Read 8x1 1MiB 9169.70 MB/sec

Fill 8x16 8GiB/N | Rand 8x16 W 0.85 MB/sec | Rand 8x16 R 12.10 MB/sec | Rand 8x16 W 0.95 MB/sec | Rand 8x16 R 14.02 MB/sec

14:15:10 | 14:16:02 | 14:16:12 | 14:17:02 | 14:17:18 | 14:18:10 | 14:18:20 | 14:19:11



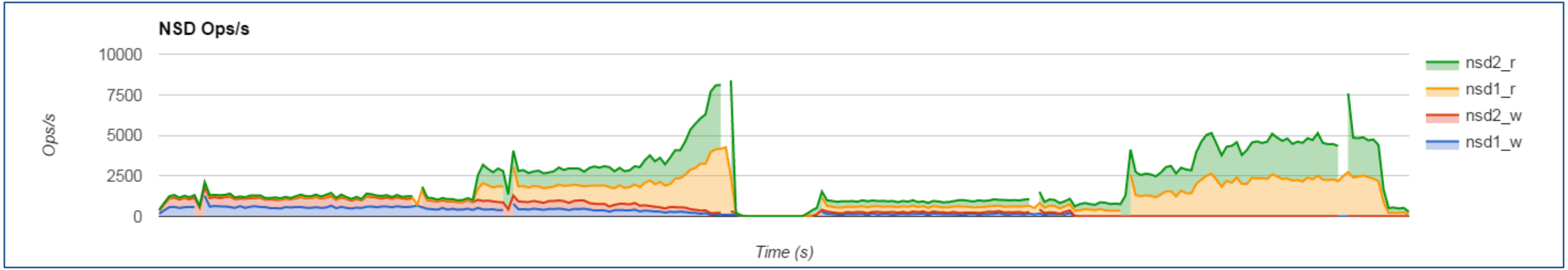


mmperfmon gpfsNumberOperations

Sequential IO, RandomIO, Simultaneous, noNXD, cont.

No NXD 2

	14:15:09	14:17:04	14:17:17	14:19:17
	Sequential Write 8x1 1MiB 9590.25 MB/sec		Sequential Read 8x1 1MiB 9169.70 MB/sec	
Fill 8x16 8GiB/N	Rand 8x16 W 0.85 MB/sec	Rand 8x16 R 12.10 MB/sec	Rand 8x16 W 0.95 MB/sec	Rand 8x16 R 14.02 MB/sec
	14:15:10 – 14:16:02	14:16:12 – 14:17:02	14:17:18 – 14:18:10	14:18:20 – 14:19:11



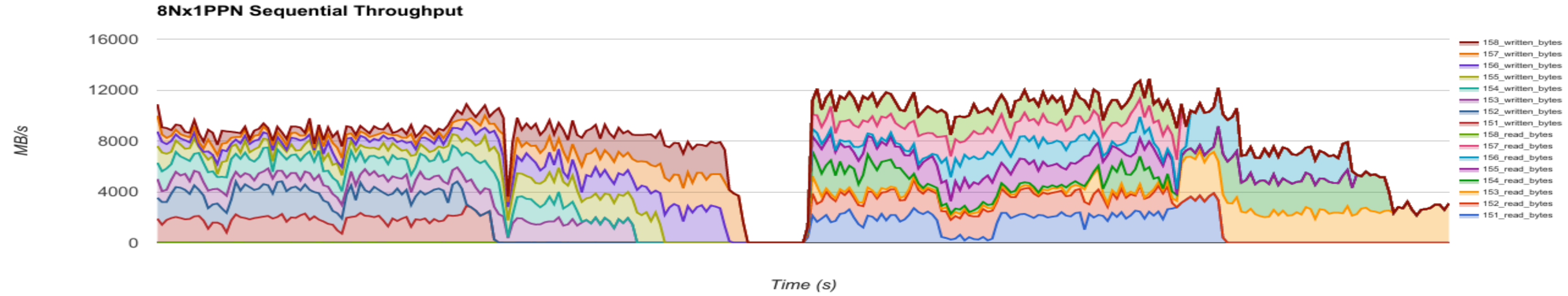
Sequential IO, RandomIO, Simultaneous, NXD

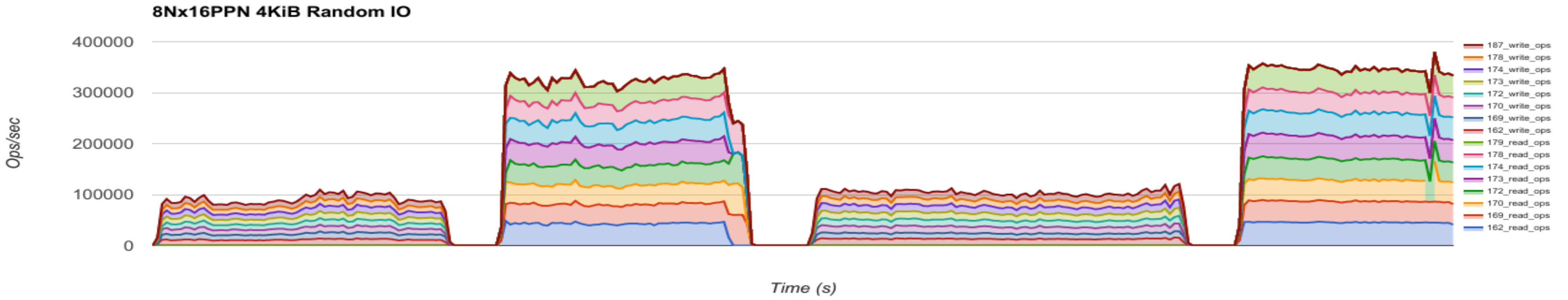
NXD Test 5

08:09:53 | 08:12:00 | 08:12:14 | 08:14:34

	Sequential Write 8x1 1MiB 8611.76 MB/sec		Sequential Read 8x1 1MiB 7877.00 MB/sec
Fill 8x16 8GiB/N	Rand 8x16 W 90,375 ops/s	Rand 8x16 R 315,439 ops/s	Rand 8x16 W 103,767 ops/s	Rand 8x16 R 332,587 ops/s

08:09:54 | 08:10:57 | 08:11:08 | 08:12:02 | 08:12:15 | 08:13:35 | 08:13:48 | 08:14:38





mmperfmon gpfsNumberOperations

Sequential IO, RandomIO, Simultaneous, NXD, cont...

NXD Test 5

08:09:53		08:12:00	08:12:14		08:14:34

Sequential Write 8x1 1MiB 8611.76 MB/sec

Sequential Read 8x1 1MiB 7877.00 MB/sec

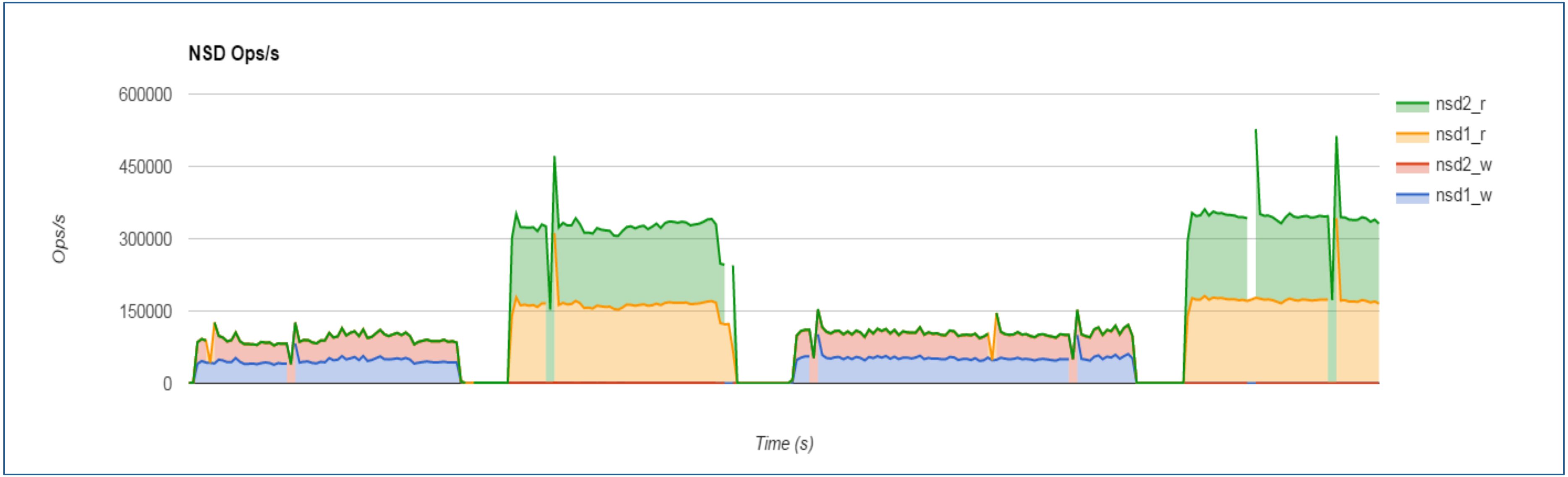
Fill 8x16 8GiB/N

Rand 8x16 W 90,375 ops/s (08:09:54 – 08:10:57)

Rand 8x16 R 315,439 ops/s (08:11:08 – 08:12:02)

Rand 8x16 W 103,767 ops/s (08:12:15 – 08:13:35)

Rand 8x16 R 332,587 ops/s (08:13:48 – 08:14:38)



mmperfmon gpfs_nsdfs_write_ops,gpfs_nsdfs_read_ops

Benefits of Nytro

The benefit of this method is

1. The penalties of read-modify-writes are greatly reduced.
2. It reduces the need for ILM to move data between pools,
3. IO gets served by the most appropriate media type
4. Operates on much larger IO sizes than HAWC (<=1MB)
5. Can address larger SSD pools than LROC (~76TB).
6. Can be enabled/disabled and tuned without changing the filesystem.
7. Mitigates the financial impact of going all flash.

Monitoring Lessons Learnt

- Is it actually random/small/fast?
  - Nytro Histogram (profile IO size only)
  - mmpmon (not granular enough)
  - mmperfmon (good but cant confirm IO type)
  - --iohist On NSD Server (prone to dropping metrics)

```
Nytro Histogram :
=========
Num Reads < 4K              = 0
Num Reads 4K                = 161393
Num Reads 4K+1 - 8K         = 1
Num Reads 8K+1 - 16K        = 0
Num Reads 16K+1 - 32K       = 0
Num Reads 32K+1 - 64K       = 0
Num Reads 64K+1 - 128K      = 0
Num Reads 128K+1 - 256K     = 0
Num Reads 256K+1 - 512K     = 0
Num Reads 512K+1 - 1M       = 604492
Num Reads 1M+1 - 2M         = 0
Num Reads 2M+1 - 4M         = 0
Num Reads 4M+1 - 8M         = 0
Num Reads 8M+1 - 16M        = 0
Num Reads 16M+1 - 32M       = 0
Num Writes < 4K             = 0
Num Writes 4K               = 11309
Num Writes 4K+1 - 8K        = 0
Num Writes 8K+1 - 16K       = 0
Num Writes 16K+1 - 32K      = 0
Num Writes 32K+1 - 64K      = 0
Num Writes 64K+1 - 128K     = 0
Num Writes 128K+1 - 256K    = 0
Num Writes 256K+1 - 512K    = 0
Num Writes 512K+1 - 1M      = 525490
Num Writes 1M+1 - 2M        = 0
Num Writes 2M+1 - 4M        = 0
Num Writes 4M+1 - 8M        = 0
Num Writes 8M+1 - 16M       = 0
Num Writes 16M+1 - 32M      = 0
```

```
Device [md300]
 Zero Sector Seeks: 0 Non-zero Sector Seeks: 64976 Percent Zero Sector Seeks: 0.00
  Device [md300] == [data] == [W]
    Total Operations: 64976 Avg Duration (ms): 127.41 Avg Size: 8.00 Percent W: 100.00 Rate: 0.03 MiB/s
      === Num Sectors Histogram ===
      Count: 64976  Range:  8.000 - 16.000; Mean:  8.000; Median:  8.000; Stddev:  0.044
      Percentiles:  90th:  8.000; 95th:  8.000; 99th:  8.000
       8.000 -    8.591: 64974 #####################################################
       8.591 -   16.000:     2 |
```

Seagate is HPC Storage

Unmatched speed and efficiency from the
Trusted Leader in HPC storage

SEAGATE