ZIL Accelerator: Random Write Revelation

Christopher George
Founder/CTO
www.ddrdrive.com

OpenStorage Summit 2011
October 26-27, 2011
San Jose, CA USA

Storage Terminology/Nomenclature:

- ZFS (Zettabyte File System)
- ZIL (ZFS Intent Log) Accelerator
  - a.k.a SLOG (Separate LOG) or dedicated log device
- SSD (Solid-State Drive)
  - SSD Types (defined by the I/O media targeted):
    - Flash (NAND) Based
    - DRAM (Dynamic Random Access Memory) Based
  - SSD Form Factors:
    - HDD (Hard Disk Drive) Compatible (2.5”)
    - PCI Express Plug-in Card
- IOPS (Input/Output operations Per Second)

The Filesystem Reinvented.

ZFS Hybrid Storage Pool:

A pool (or collection) of high capacity, low cost, and low RPM HDDs accelerated with integrated support of both read and write optimized SSDs.

The key is both storage devices (HDD/SSD) work together as one to provide the capacity and cost per bit benefits of an HDD with the performance and power benefits of an SSD.

The ZFS Vanguard.

ZIL Accelerator: (ZIL != log device)

One of the two optional accelerators built into ZFS. A ZIL Accelerator is expected to be write optimized as it only captures synchronous writes. Thus, a prospective SSD must have both extremely low latency and high ***sustained*** write IOPS capability to successfully target.

A ZIL Accelerator is critical for accelerating applications bound by synchronous writes (e.g. NFS, iSCSI, CIFS).

A ZIL Accelerator can be created from either type of SSD (DRAM or Flash). Which SSD type to choose?

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

Questions to be answered:

- **What is the ZIL (ZFS Intent Log)?**
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

What is the ZFS Intent Log (ZIL)?

- Logs all file system related system calls as transactions in host memory. If synchronous semantics apply (O_SYNC, fsync()...), transactions are also placed on stable (non-volatile) storage, so in the event of a host failure each can be replayed on the next reboot.
- Satisfies POSIX requirements for synchronous write transactions.
- Default implementation uses the pool for stable “on-disk” format. Optionally, a ZIL Accelerator can be added for increased performance.
- One ZIL per dataset (e.g. file system, volume), with one or more datasets per pool. A ZIL Accelerator is a pool assigned resource and thus shared by **all** datasets (ZILs) contained in that pool.
- Transactions are committed to the pool as a group (txg) and involve reading the ZIL “in-memory” representation and NOT the "on-disk" format. After the txg commits, the relevant ZIL (either pool based or optionally a ZIL Accelerator) blocks are released.

What is a synchronous write transaction?

- Synchronous writes are forced to stable (non-volatile) storage prior to being acknowledged. Commonly initiated by setting O_SYNC, O_DSYNC, or O_RSYNC flag parameters when the target file was opened or by calling fsync().
- Guarantees, upon a host power or hardware failure all writes successfully acknowledged prior are safely stored and unaffected.
- Critical Assumption: All relevant storage devices (including HBA Controller) and associated device drivers must properly implement the SCSI SYNCHRONIZE_CACHE or ATA FLUSH CACHE command by flushing any/all volatile caches to stable (non-volatile) storage.
- **WARNING:** Some storage devices ignore the cache flush command and are unable to correctly implement synchronous write semantics.
- **WARNING:** Do NOT set the system-wide "zfs_nocacheflush" tunable unless every system storage device's volatile cache is power protected.

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- **What are the key characteristics of a ZIL Accelerator?**
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

What are the key characteristics of a ZIL Accelerator?

- The ZIL Accelerator is added to a pool, thus ***shared*** by all datasets (file systems, volumes, clones) contained within this pool.
- Device data integrity is paramount to operational correctness. Unless the ZIL Accelerator is mirrored, no ZFS checksum fallback is available.
- Requires a low latency, high ***sustained*** write IOPS capable device.
- Write IOPS intensive, never read unless at reboot (replay) and import.
- **ZFS does NOT support TRIM**, an issue for Flash SSDs but not DRAM SSDs.
- BONUS: By relocating the default ZIL from the pool, it reduces both pool block fragmentation and pool IO congestion, increasing all IO performance.
- **WARNING**: Device must correctly and consistently implement the SCSI SYNCHRONIZE_CACHE or ATA FLUSH CACHE command for cache flush support.
- **WARNING**: Operational correctness (cache flush support) requires power protection of ALL on-board volatile caches. Most obvious with memory components, but also beware of controller based on-chip volatile caches.

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- **Why is ZIL Accelerator volatile cache power protection so critical?**
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

Why is ZIL Accelerator volatile cache power protection so critical?

A ZIL Accelerator's **"prime directive"** is the stable (non-volatile) storage of all synchronous writes. So in case of a host failure, all log device data, which has already been acknowledged as securely written, can then be replayed (i.e. rewritten to the pool) on the next host reboot. The above behavior is of the highest priority to the mutually agreed contract between ZFS and any application which relies on it, failure to uphold said contract can/will lead to application level corruption and integrity issues. Application level (not pool based) consistency and robustness are both predicated on the ZIL Accelerator's ability to secure ***all*** stored data, even and especially in case of an unexpected SSD power loss. Any SSD which does not power protect on-board volatile caches violates the above "prime directive" and thus sacrifices the very premise and promise of a ZIL Accelerator.

Why is ZIL Accelerator volatile cache power protection so critical?

Data safety features prepare for unexpected power-loss and protect system and user data.

Worried about data loss during unplanned power shutdowns? Waiting for a solution to address system power loss in client environments or inadvertent drive removal in data center applications? Your wait is over – the Intel® Solid-State Drive 320 Series (Intel® SSD 320 Series) introduces enhanced power-loss data protection features that prepare the SSD for unexpected system power loss and protect your data.

Importance of Power-Loss Data Protection

During a "clean" shutdown, most host sytems initiate a command (the STANDBY IMMEDIATE command) to an SSD to give the SSD enough time to prepare for the shutdown. This allows the SSD to save data currently in transition (in temporary buffers) to the non-volatile NAND media.

However, during an unsafe power shutdown, the SSD abruptly loses power before the host system can initiate the STANDBY IMMEDIATE command. This prevents data in the temporary buffers from being saved in the non-volatile NAND.

How Power-Loss Data Protection is Implemented

The SSD then relies on its on-board power-loss protection capacitance to provide enough energy for the SSD firmware to move data from the transfer buffer and other temporary buffers to the NAND.

What Type of Data is Protected

During an unsafe shutdown, firmware routines in the Intel SSD 320 Series respond to power loss interrupt and make sure both user data and system data in the temporary buffers are transferred to the NAND media.

[Excerpts from the "Enhanced power-loss data protection in the Intel SSD 320 Series" Intel Technology Brief.]

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- **Which Intel SSDs have volatile cache protection and which do not?**
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

Intel Flash SSDs which do NOT power protect on-board volatile caches!

No power-loss data protection:

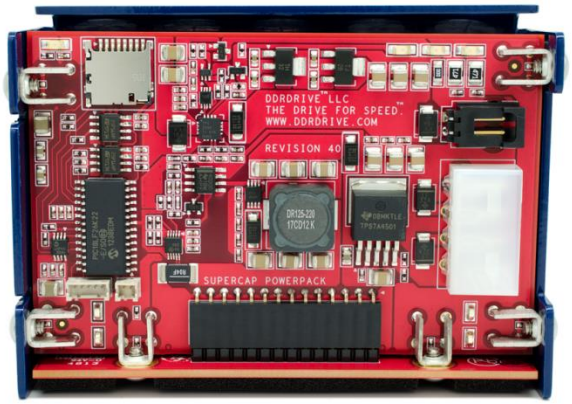
- **Intel 311 Series**
- Intel 520/510 Series
- Intel 310 Series
- **Intel X25-E Series**
- Intel X25-M Series
- Intel X25-V Series

Intel Flash SSDs which do power protect on-board volatile caches:

- **Intel 710 Series**
  - 100GB / 200GB / 300GB
  - **Power-Loss Data Protection**
  - 25nm MLC Flash with HET
  - 2.5” SATA II SSD

- **Intel 320 Series**
  - 40GB/80GB/160GB/300GB/600GB
  - **Power-Loss Data Protection**
  - 25nm MLC Flash
  - 2.5” SATA II SSD

DRAM SSD which power protects ALL on-board volatile memory.



DDRdrive X1:

Guarantees correct and consistent implementation of cache flushes. (SCSI SYNCHRONIZE_CACHE Command)

Guarantees, in conjunction with an internally mounted DDRdrive SuperCap Power Pack or an externally attached UPS, all on-board volatile memory is power protected. During a host failure or power loss an automatic backup occurs transferring all DRAM contents to on-board SLC NAND. Then automatically restores NAND to DRAM when host is next powered on and reboots.

The X1 is singularly designed to perform the unique function of a ZIL Accelerator.

DDRdrive™
The drive for speed.™

OpenStorage Summit 2011

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- **Is the ZIL Accelerator access pattern random and/or sequential?**
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

ZIL Accelerator Access Pattern?

The answer is a key variable in determining which of the SSD types is best suited as a ZIL Accelerator. As a Flash based SSD, unlike a DRAM SSD, has highly variable write IOPS performance depending on IO distribution (sequential, random, and mixed). For a Flash SSD, performance variability is especially pronounced if the workload is random or mixed.

Contrast with a DRAM SSD, in which performance is absolutely consistent regardless of IO distribution.

Is the ZIL Accelerator access pattern random and/or sequential?

iopattern.d - Single IOzone workload targeted at a single file system (ZIL):

DEVICE	%RAN	%SEQ	COUNT	MIN	MAX	AVG	KR	KW
sd5	**6**	94	152	4096	131072	34708	0	5152
sd5	**0**	100	506	4096	131072	7422	0	3668
sd5	**0**	100	830	4096	131072	7446	0	6036
sd5	**2**	98	272	4096	131072	21202	0	5632
sd5	**1**	99	483	4096	131072	8904	0	4200
sd5	**0**	100	606	4096	131072	8502	0	5032
sd5	**1**	99	511	4096	131072	12167	0	6072
sd5	**1**	99	440	4096	131072	10994	0	4724
sd5	**0**	100	601	4096	69632	8444	0	4956
sd5	**1**	99	583	4096	131072	12042	0	6856
sd5	**1**	99	436	4096	131072	10878	0	4632
sd5	**2**	98	148	4096	73728	18293	0	2644
sd5	**0**	100	928	4096	131072	7216	0	6540
sd5	**6**	94	152	4096	131072	34708	0	5152
sd5	**2**	98	544	4096	131072	9118	0	4844
sd5	**0**	100	928	4096	131072	7216	0	6540
sd5	**2**	98	414	4096	131072	16176	0	6540
sd5	**1**	99	267	4096	81920	11060	0	2884
sd5	**0**	100	943	4096	131072	7722	0	7112
sd5	**5**	95	152	4096	131072	34708	0	5152

Is the ZIL Accelerator access pattern random and/or sequential?

seeksize.d - Single IOzone workload targeted at a single file system (ZIL):

```
ZIL Accelerator = sd5 (negative seek offsets)

           value  ------------- Distribution ------------- count
          -32768 |                                         0
          -16384 |                                         35
           -8192 |                                         234
           -4096 |                                         0
           -2048 |                                         70
           -1024 |                                         35
            -512 |                                         0
            -256 |                                         3
            -128 |                                         0
             -64 |                                         0
             -32 |                                         0
             -16 |                                         0
              -8 |                                         0
              -4 |                                         0
              -2 |                                         0
              -1 |                                         0
               0 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ 56701
```

Is the ZIL Accelerator access pattern random and/or sequential?

seeksize.d - Single IOzone workload targeted at a single file system (ZIL):

```
ZIL Accelerator = sd5 (positive seek offsets)

           value  ------------- Distribution ------------- count
               0 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ 56701
               1 |                                         0
               2 |                                         0
               4 |                                         0
               8 |                                         9
              16 |                                         11
              32 |                                         5
              64 |                                         4
             128 |                                         14
             256 |                                         0
             512 |                                         0
            1024 |                                         35
            2048 |                                         0
            4096 |                                         0
            8192 |                                         218
           16384 |                                         1
           32768 |                                         0
```

Is the ZIL Accelerator access pattern random and/or sequential?

iopattern.d - Five IOzone workloads each targeted at separate file systems (ZILs):

DEVICE	%RAN	%SEQ	COUNT	MIN	MAX	AVG	KR	KW
sd5	**71**	29	619	4096	131072	14862	0	8984
sd5	**63**	37	100	4096	131072	59064	0	5768
sd5	**27**	73	1706	4096	131072	5997	0	9992
sd5	**37**	63	717	4096	131072	12419	0	8696
sd5	**38**	62	488	4096	131072	19539	0	9312
sd5	**32**	68	962	4096	131072	10078	0	9468
sd5	**65**	35	820	4096	131072	10464	0	8380
sd5	**22**	78	946	4096	131072	12448	0	11500
sd5	**36**	64	1132	4096	131072	7927	0	8764
sd5	**55**	45	664	4096	131072	16414	0	10644
sd5	**22**	78	490	4096	131072	13642	0	6528
sd5	**30**	70	877	4096	131072	8322	0	7128
sd5	**42**	58	786	4096	131072	11886	0	9124
sd5	**21**	79	675	4096	131072	15316	0	10096
sd5	**33**	67	1628	4096	131072	7024	0	11168
sd5	**43**	57	458	4096	131072	24745	0	11068
sd5	**25**	75	459	4096	131072	14813	0	6640
sd5	**35**	65	1513	4096	131072	7607	0	11240
sd5	**52**	48	282	4096	131072	29441	0	8108
sd5	**28**	72	1677	4096	131072	7361	0	12056

Is the ZIL Accelerator access pattern random and/or sequential?

seeksize.d - Five IOzone workloads each targeted at separate file systems (ZILs):

```
ZIL Accelerator = sd5 (negative seek offsets)

           value  ------------- Distribution ------------- count
          -65536 |                                         12
          -32768 |@                                        9094
          -16384 |@                                        4162
           -8192 |                                         2328
           -4096 |                                         1210
           -2048 |                                         824
           -1024 |                                         695
            -512 |                                         730
            -256 |                                         2076
            -128 |@                                        3498
             -64 |@                                        3548
             -32 |@                                        6635
             -16 |@@                                       12743
              -8 |                                         0
              -4 |                                         0
              -2 |                                         0
              -1 |                                         0
               0 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  158590
```

Is the ZIL Accelerator access pattern random and/or sequential?

seeksize.d - Five IOzone workloads each targeted at separate file systems (ZILs):

```
ZIL Accelerator = sd5 (positive seek offsets)

           value  ------------- Distribution ------------- count
               0 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ 158590
               1 |                                         0
               2 |                                         0
               4 |                                         0
               8 |@@@                                      18452
              16 |@@                                       10456
              32 |@                                        5298
              64 |@                                        3767
             128 |@                                        3641
             256 |                                         2121
             512 |                                         952
            1024 |                                         743
            2048 |                                         852
            4096 |                                         1178
            8192 |                                         2258
           16384 |@                                        4132
           32768 |@                                        9035
           65536 |                                         0
```

Is the ZIL Accelerator access pattern random and/or sequential?

ZIL Accelerator Access Pattern:

A predominately sequential write pattern is found for a pool with only a single file system. But as additional file systems are added to the pool, the resultant (or aggregate) write pattern trends to random access. Almost 50% random with a pool containing just 5 filesystems. This makes intuitive sense knowing each filesystem has a ZIL and ***all*** share the same pool assigned ZIL Accelerator.

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- **How do Flash/DRAM based SSDs random write IOPS compare?**
- How do Flash/DRAM based SSDs IOPS/$ compare?
- Are the finite write limitations of Flash based SSDs a concern?

Inherent disadvantages of a Flash write compared to a DRAM write?

- Can ONLY program a zero (change a 1 to 0), must be erased (set to 1) prior.
- Each write “will” require two separate Flash operations (erase/program).
- Asymmetric Flash operation (erase/program) unit sizes (Block/Page).
- Asymmetric Flash (erase/program) completion times (1.5ms/200us).
- Block/Page asymmetry (64-128X) results in RMW (Read Modify Write).
- RMW results in a write multiplicative effect called write amplification.
- Finite number of writes (erase/program) cycles (1-10K MLC/100K SLC).
- Complicated wear leveling schemes (LBA remapping) for use as an SSD.
- Writes (erase/program) will fail, requiring Bad Block Management.
- Continual performance degradation without TRIM support or Secure Erase.
- **SUMMATION**: Flash has nondeterministic and inferior write performance.

Iometer benchmark devices, settings, procedure, and system:

Log Devices Under Test:

- Intel 710 - 100GB/200GB/300GB (MLC SSD)
- Intel 320 - 40GB/160GB/300GB (MLC SSD)
- DDRdrive X1 (DRAM SSD)

Iometer 1.1.0 rc1 4KB Random Write IOPS Settings:

- Target Raw Devices Directly
- 32 Outstanding I/O's
- Pseudo Random Data Pattern

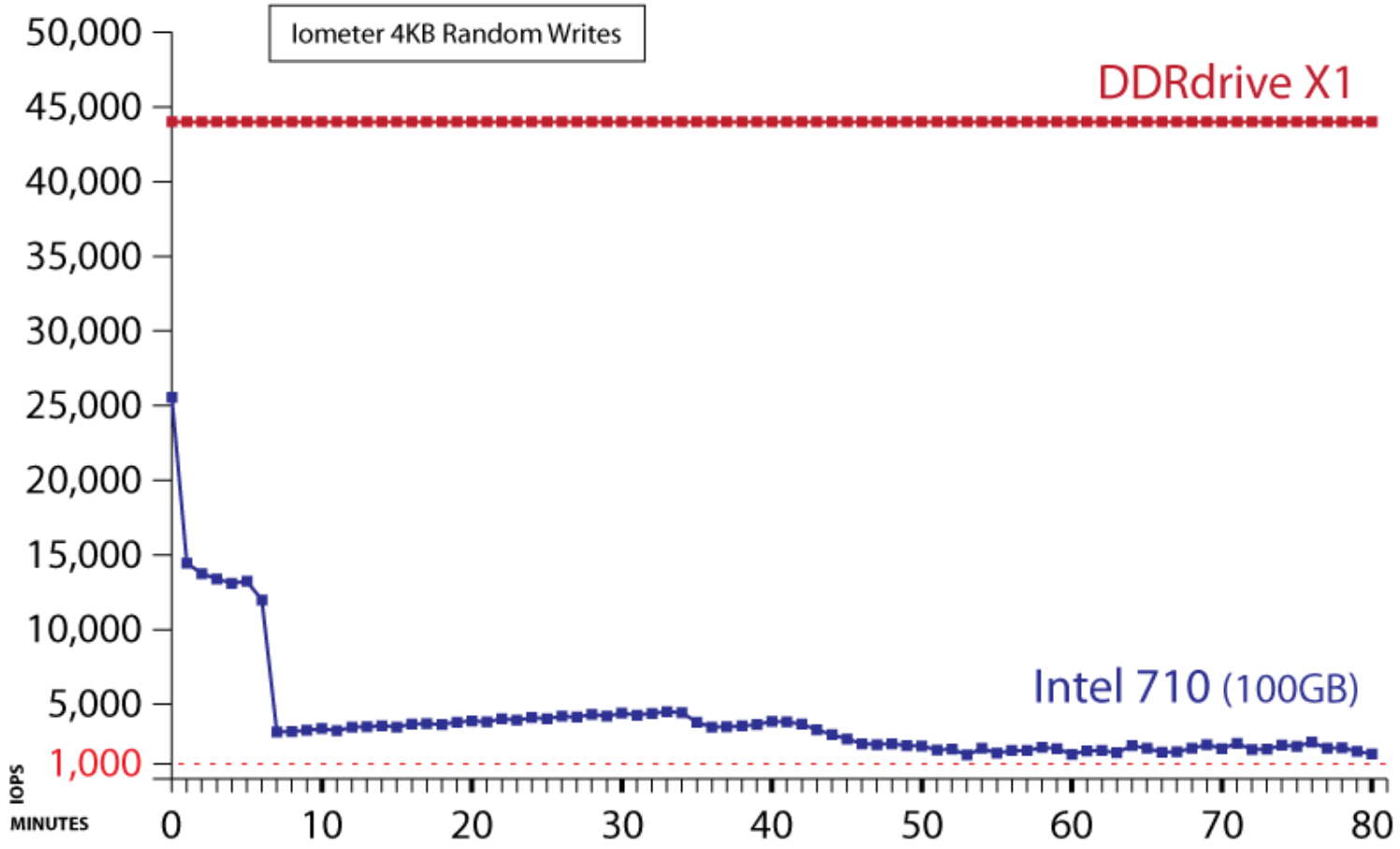
Benchmark Procedure :

- Secure Erase (SE) Flash SSD (no SE with X1).
- Start test, record each 60 second update.
- Run test continuously for 80 minutes.
- Capture last update screenshot, stop test.

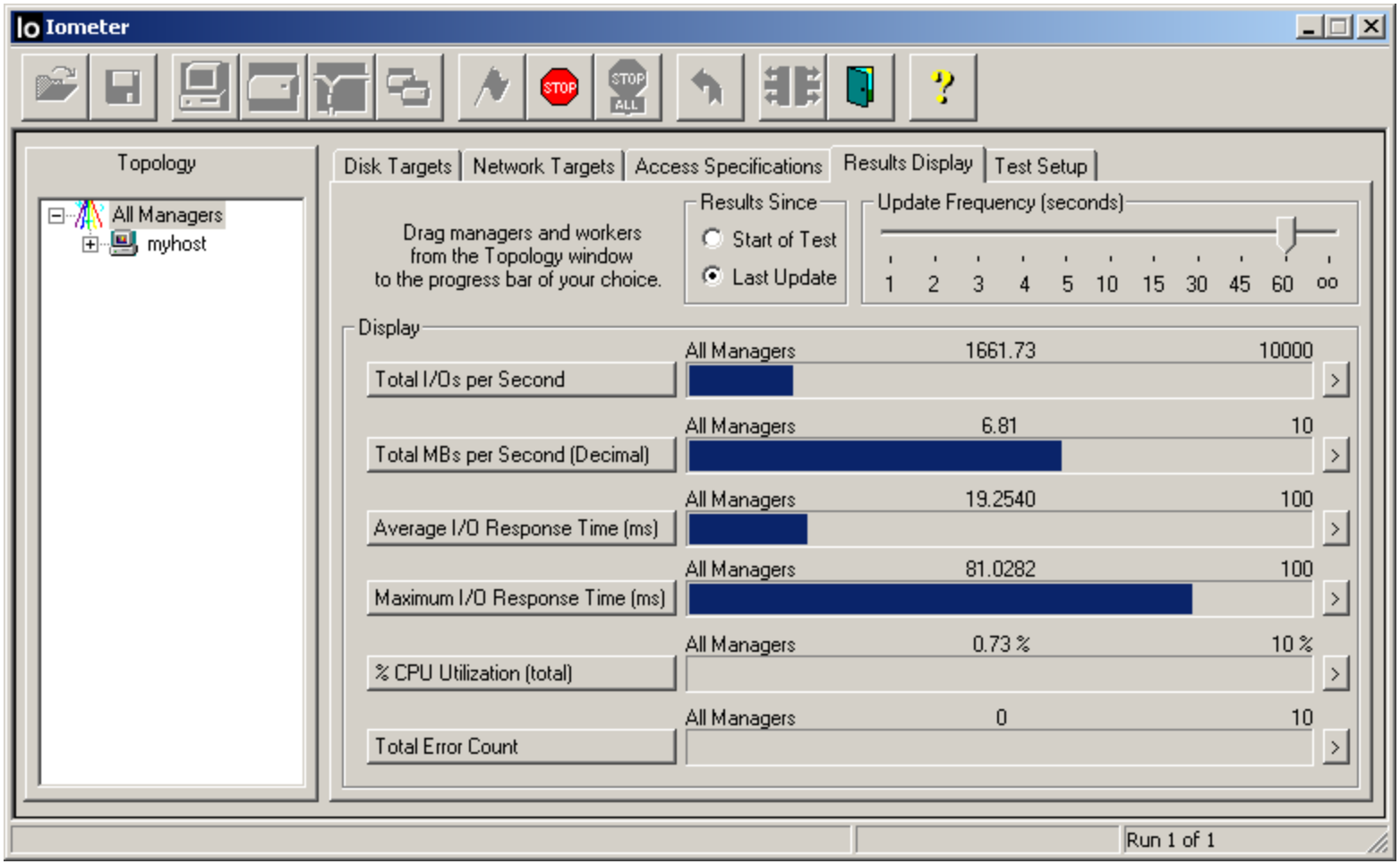
Benchmark Storage Server System:

- Nexenta NexentaStor 3.1.1 Operating System
- SuperMicro X8DTU-F Motherboard / ICH10R
- Dual Quad Core Xeon E5620 / 24GB Memory

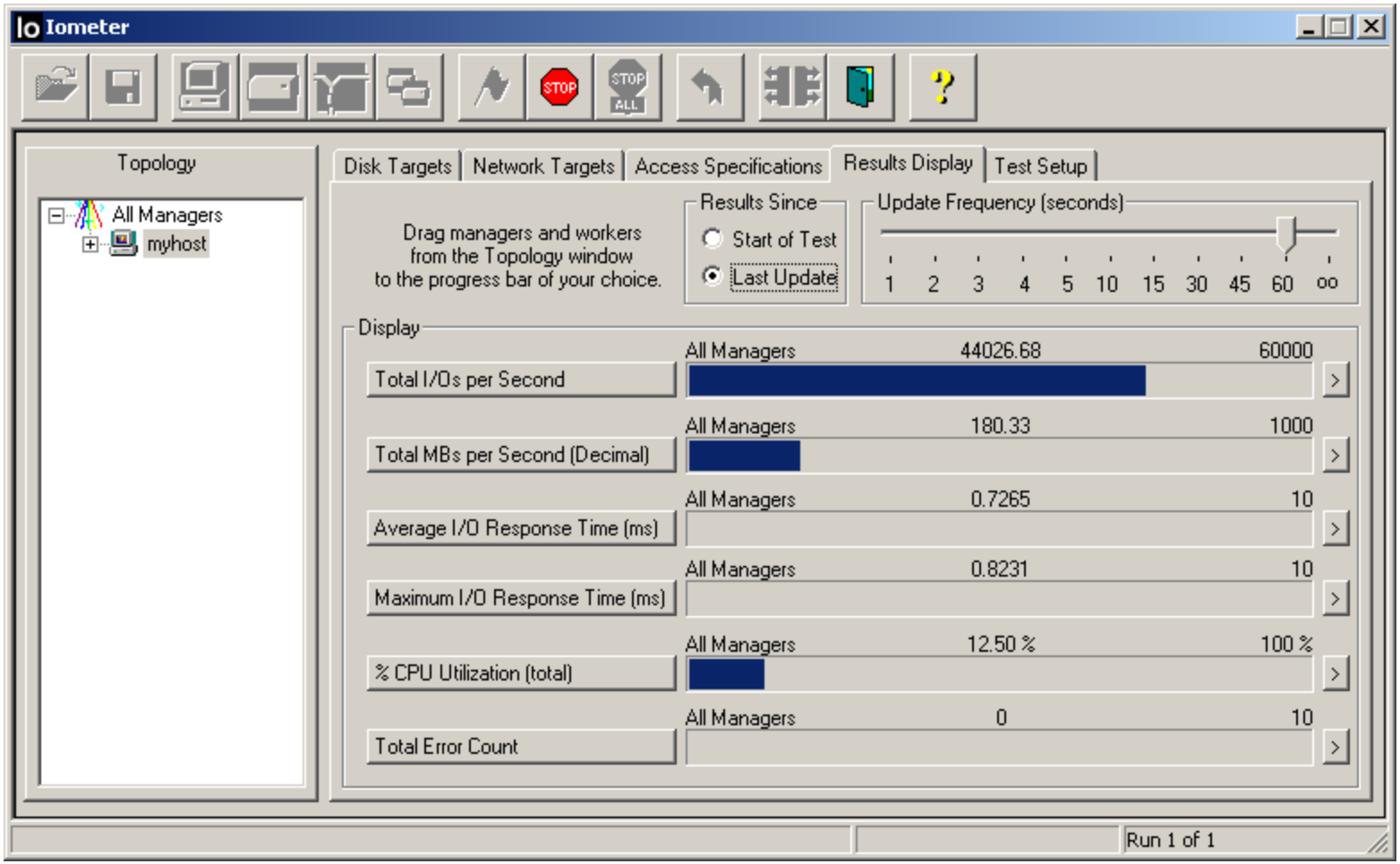
How do Flash/DRAM based SSDs random write IOPS compare?



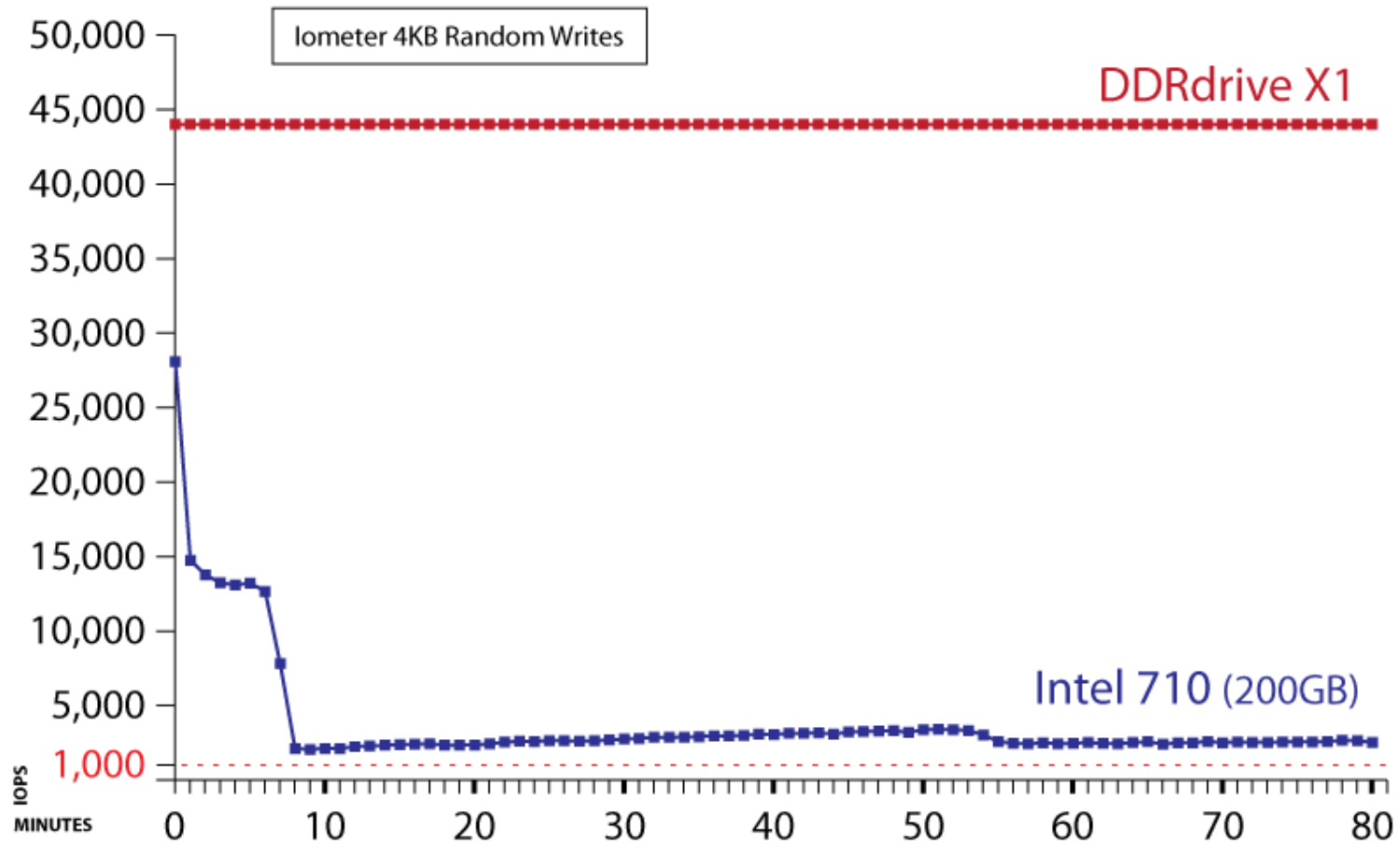
Intel 710 100GB - 4KB Random Write - 80 Minute Test Run:



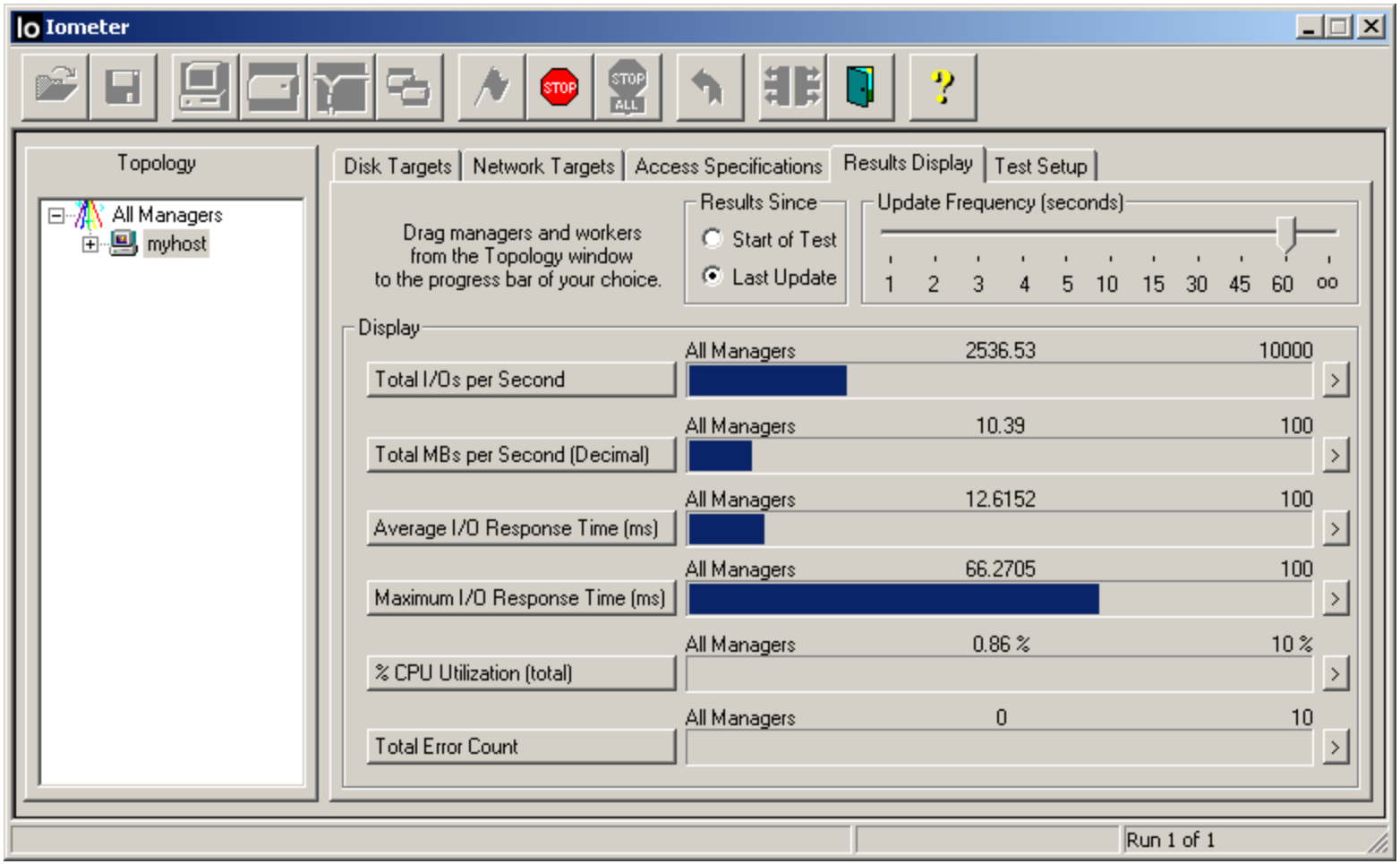
DDRdrive X1 - 4KB Random Write - 80 Minute Test Run:



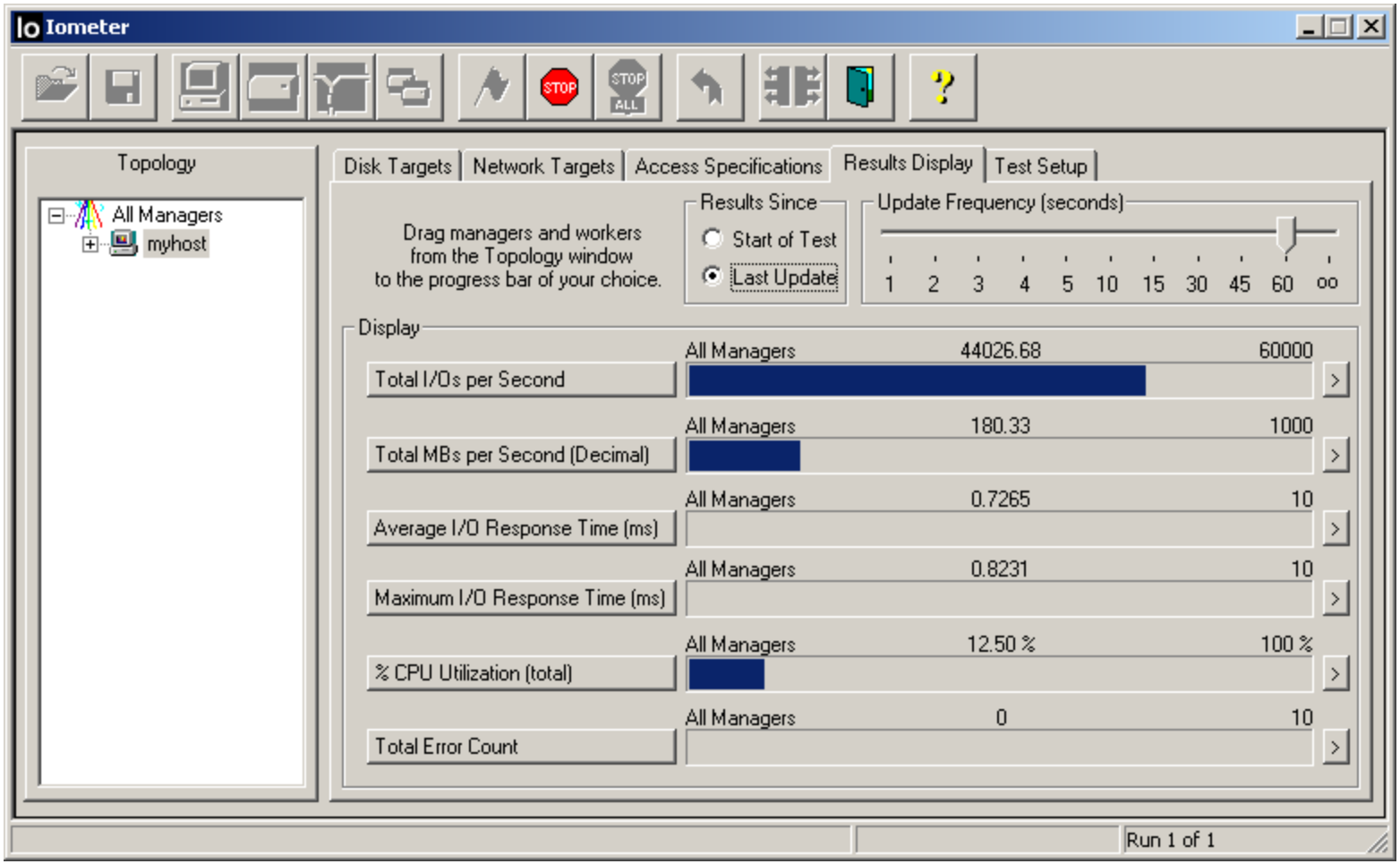
How do Flash/DRAM based SSDs random write IOPS compare?



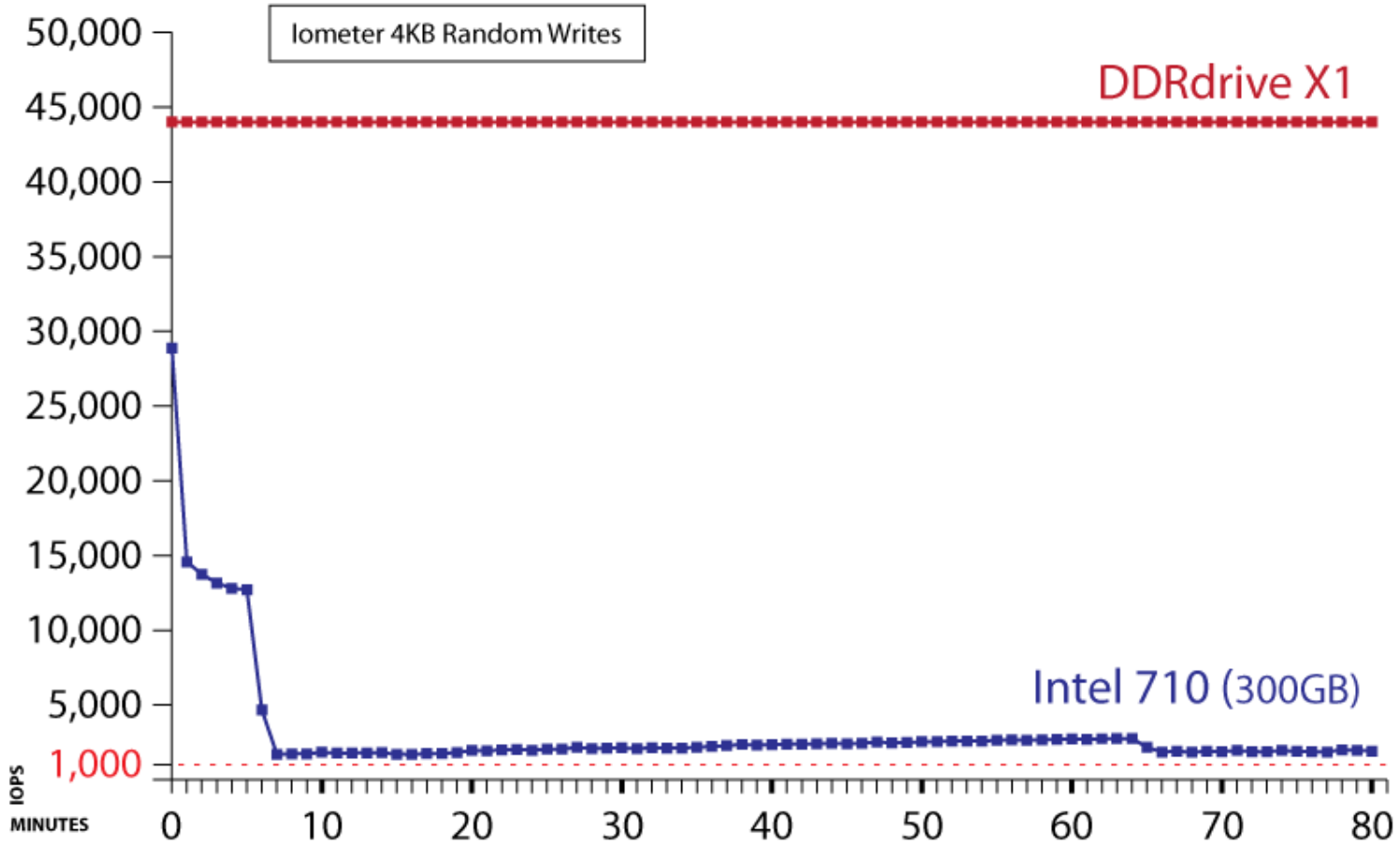
Intel 710 200GB - 4KB Random Write - 80 Minute Test Run:



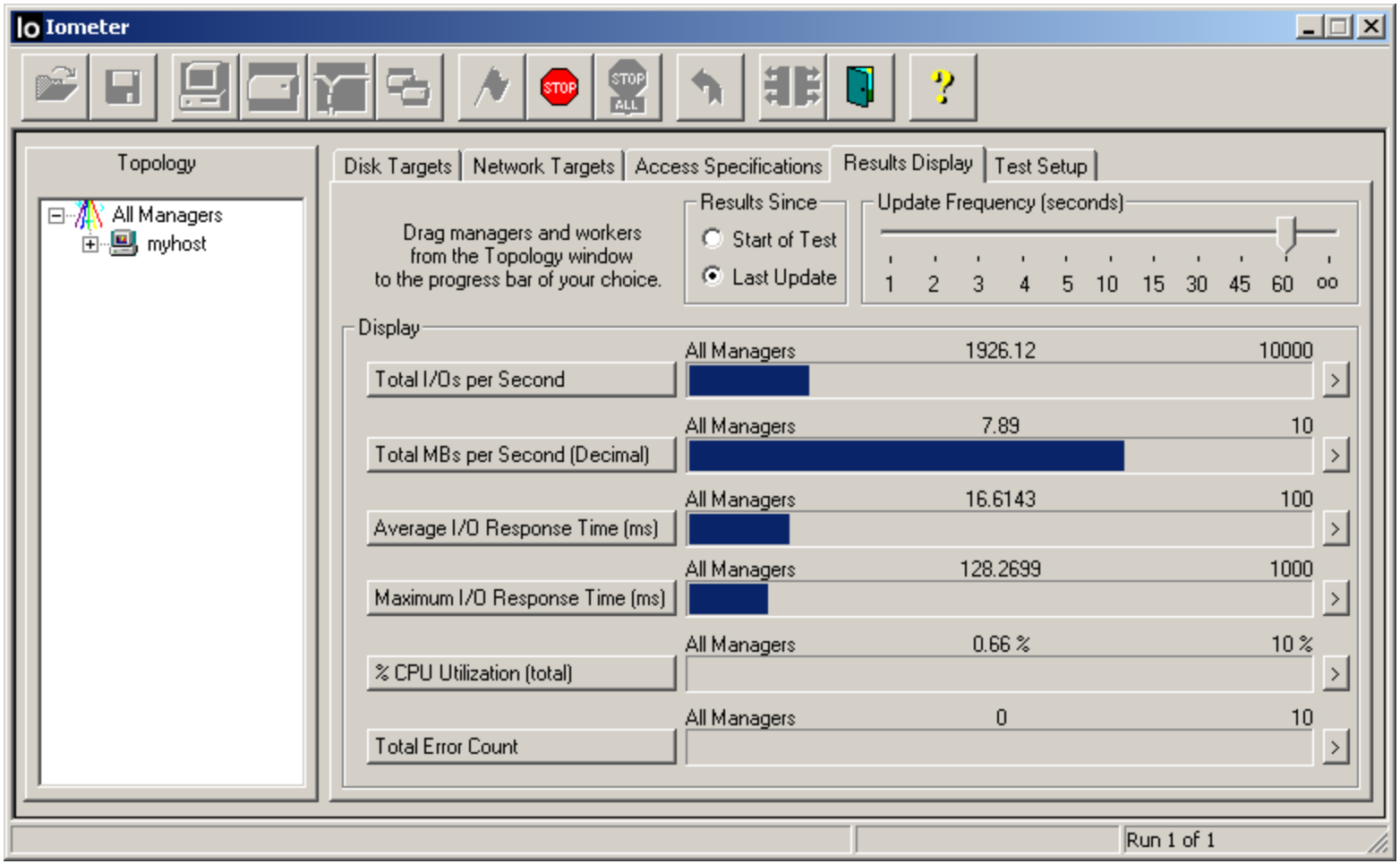
DDRdrive X1 - 4KB Random Write - 80 Minute Test Run:



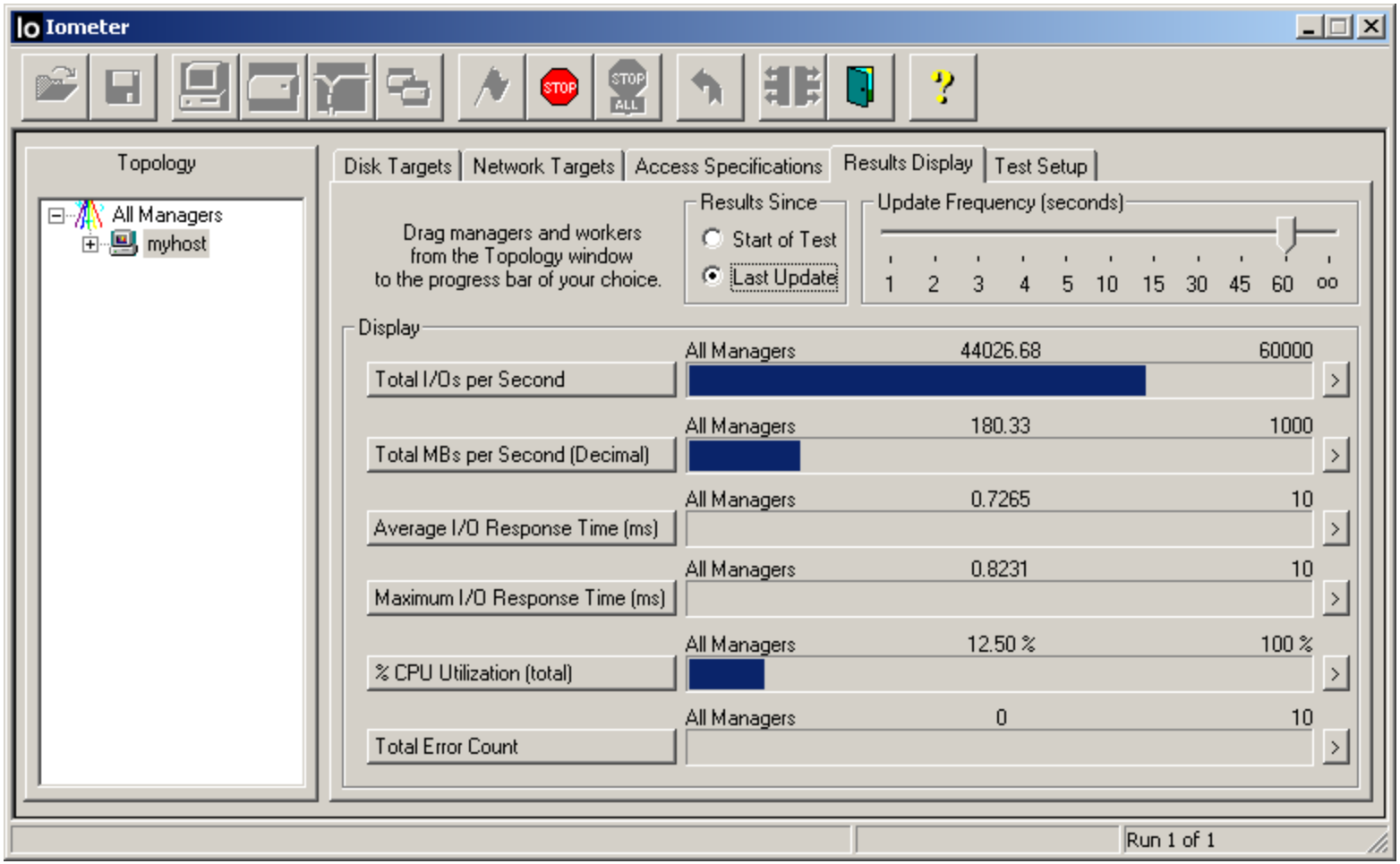
How do Flash/DRAM based SSDs random write IOPS compare?



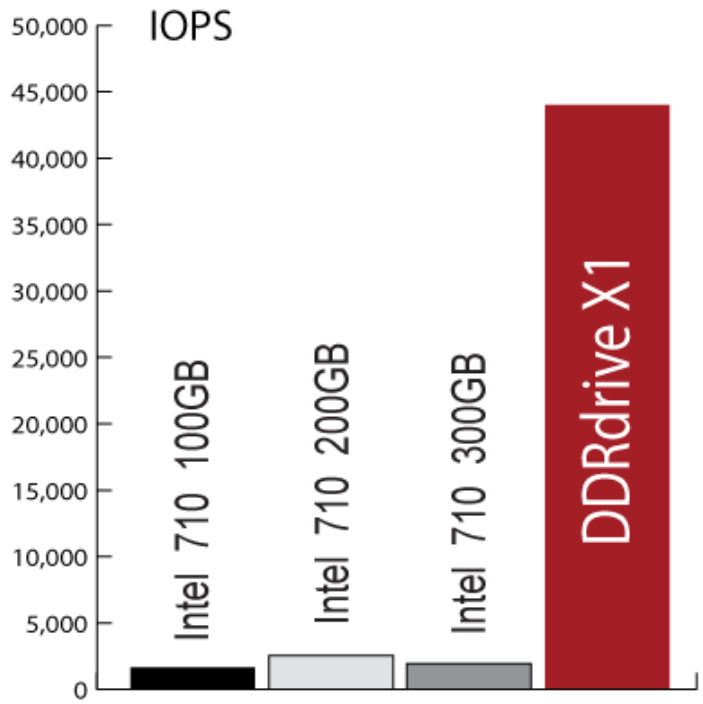
Intel 710 300GB - 4KB Random Write - 80 Minute Test Run:



DDRdrive X1 - 4KB Random Write - 80 Minute Test Run:



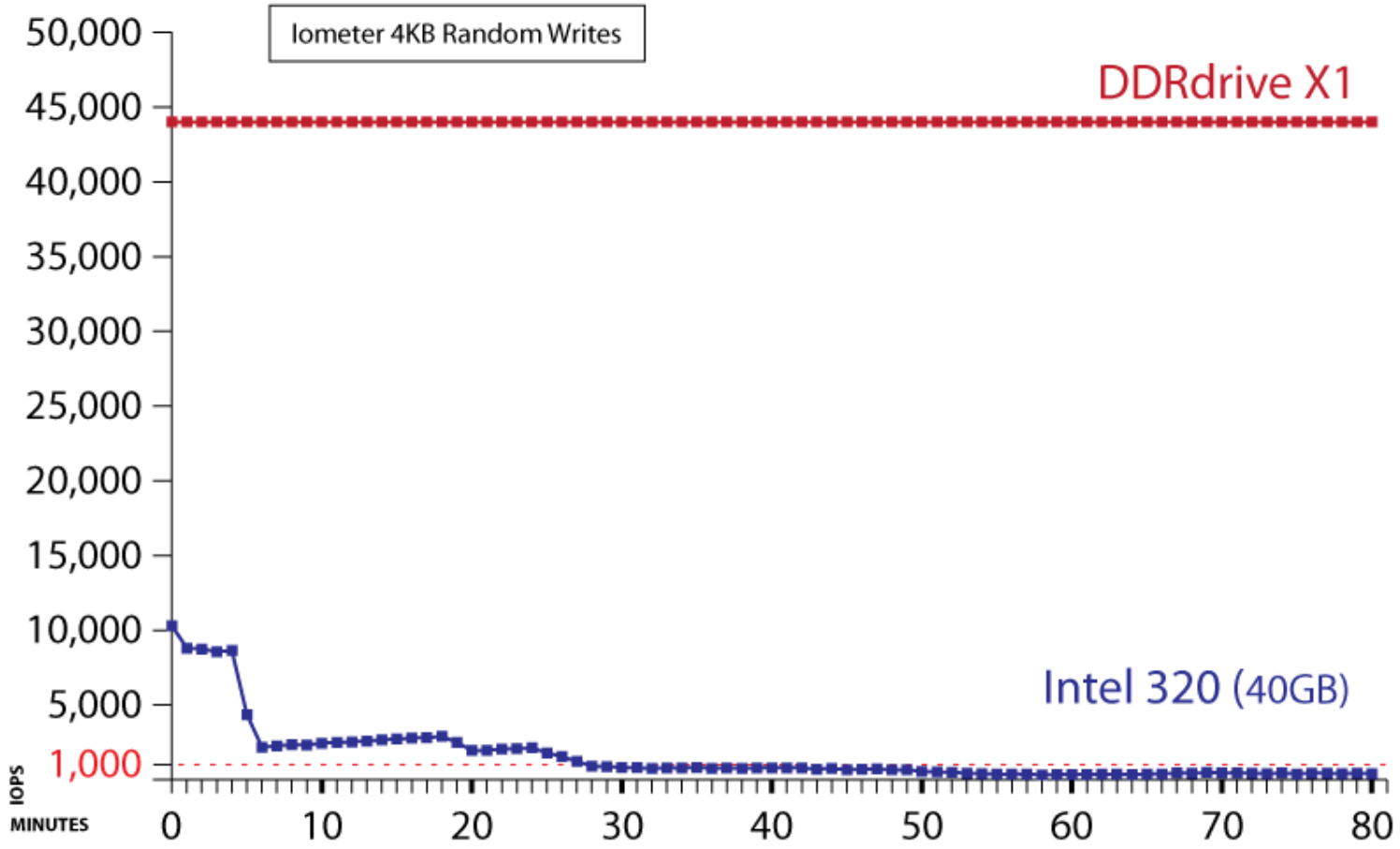
DDRdrive X1 / Intel 710 Series / 80 Minute Test Run Compare:



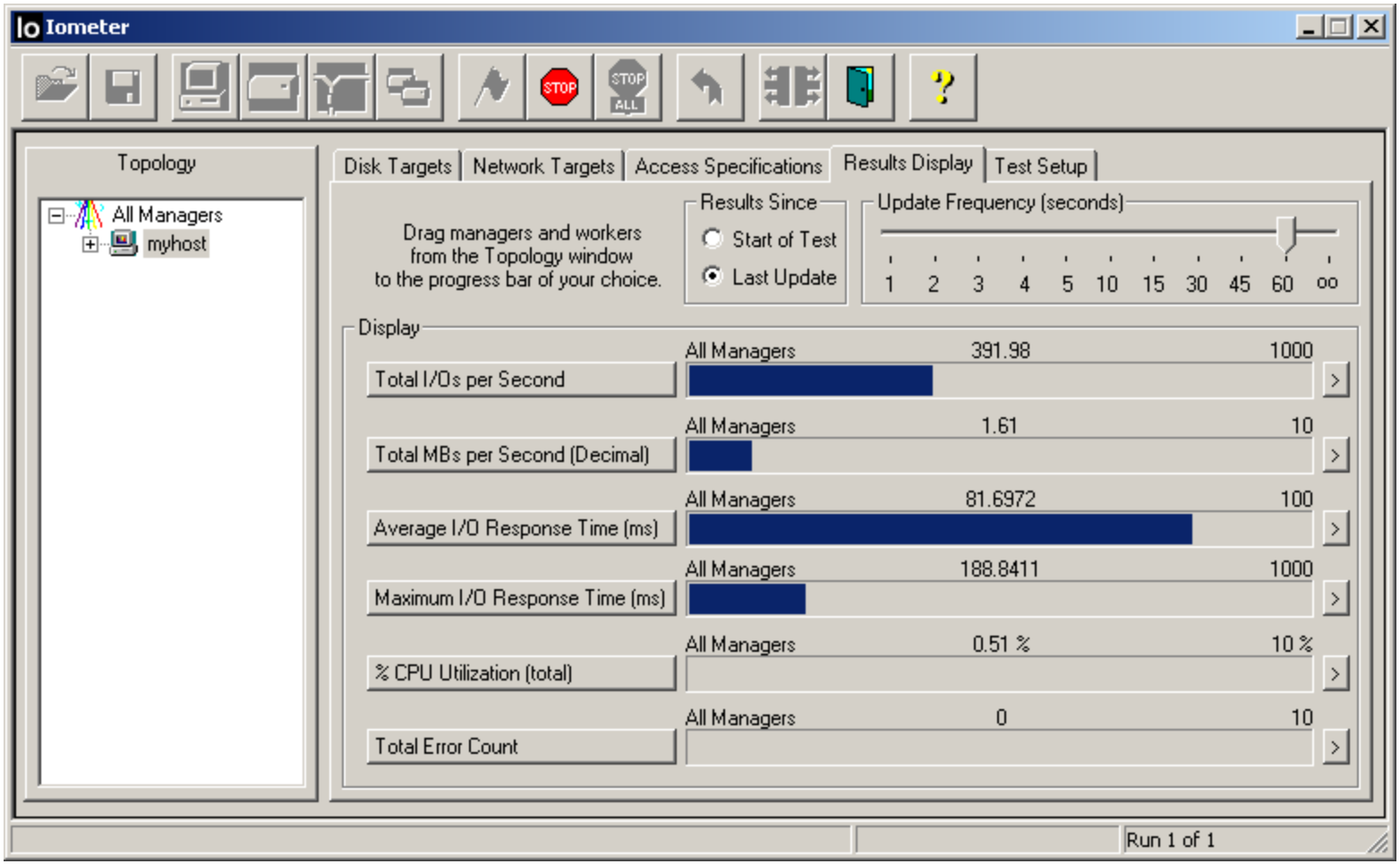
4KB Random Write

Log Device	IOPS
Intel 710 100GB	1,661
Intel 710 200GB	2,536
Intel 710 300GB	1,926
DDRdrive X1	44,026

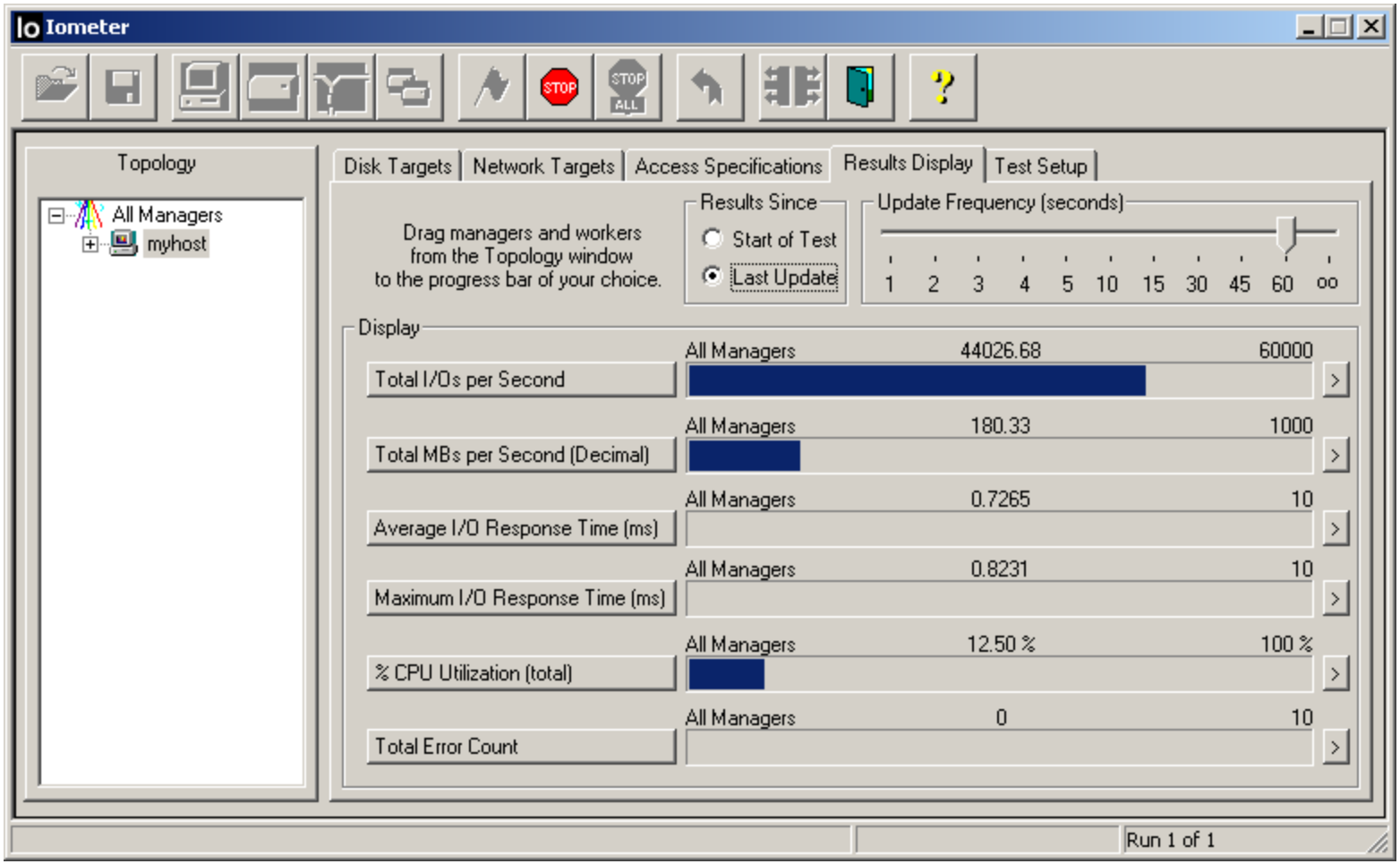
How do Flash/DRAM based SSDs random write IOPS compare?



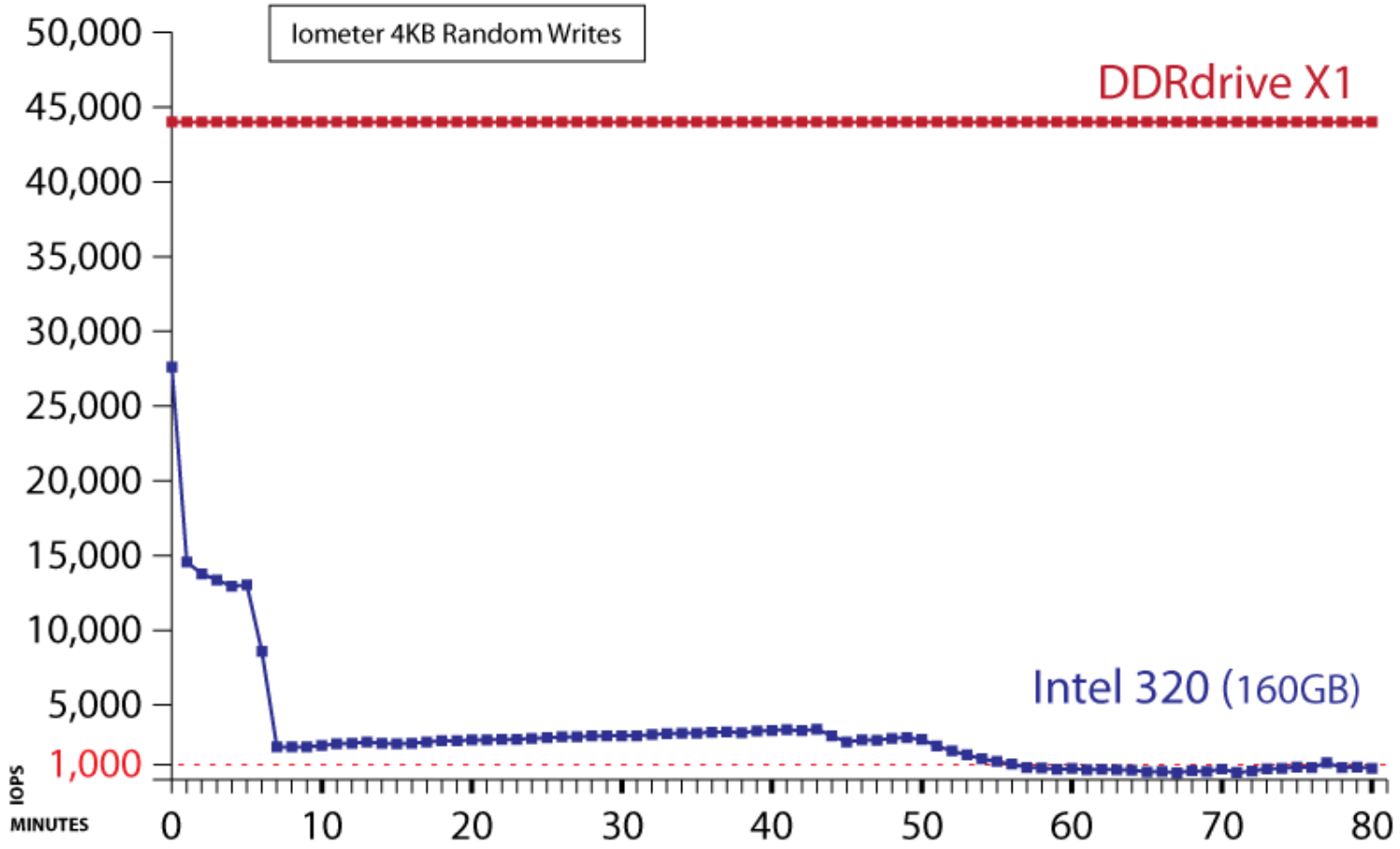
Intel 320 40GB - 4KB Random Write - 80 Minute Test Run:



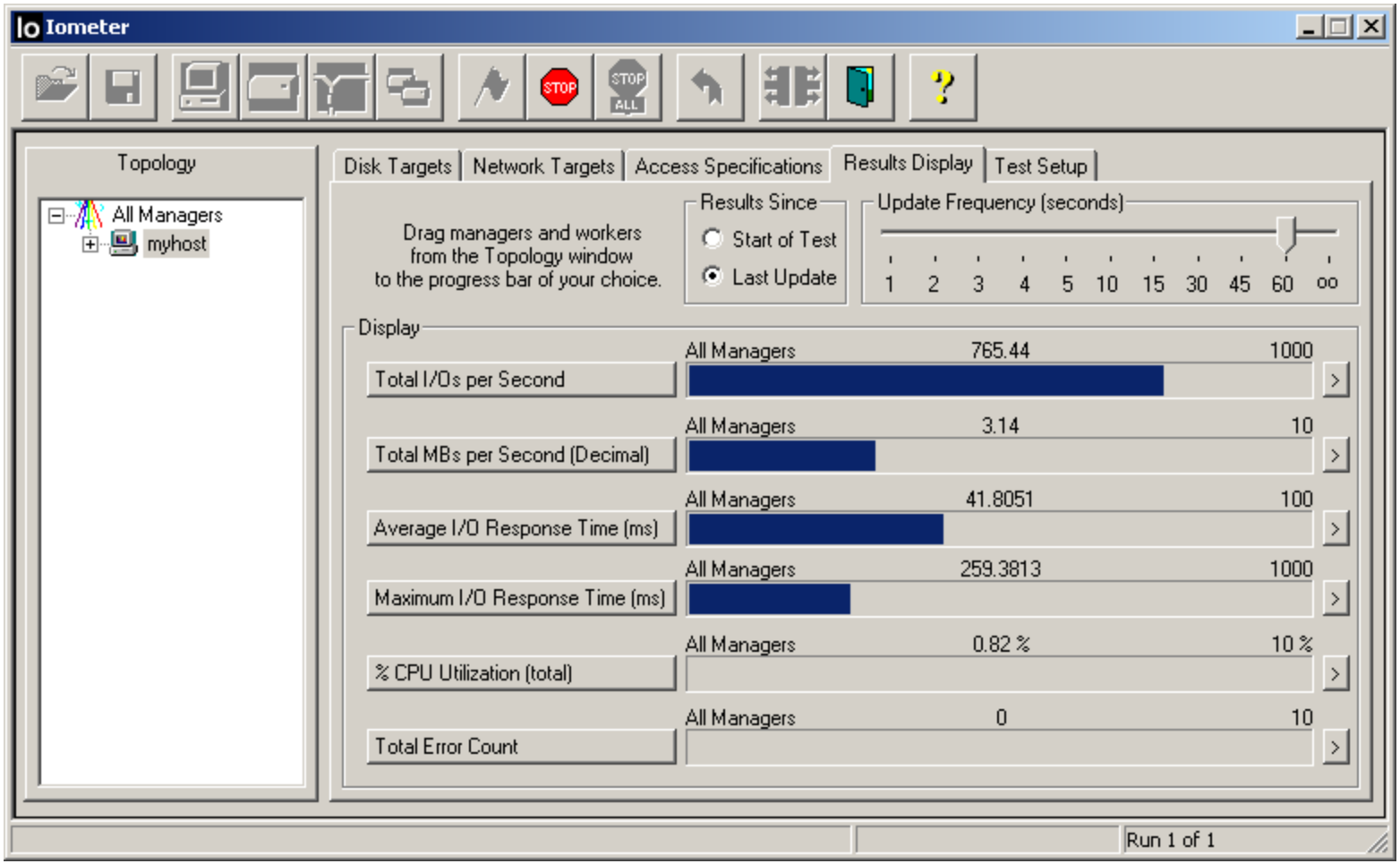
DDRdrive X1 - 4KB Random Write - 80 Minute Test Run:



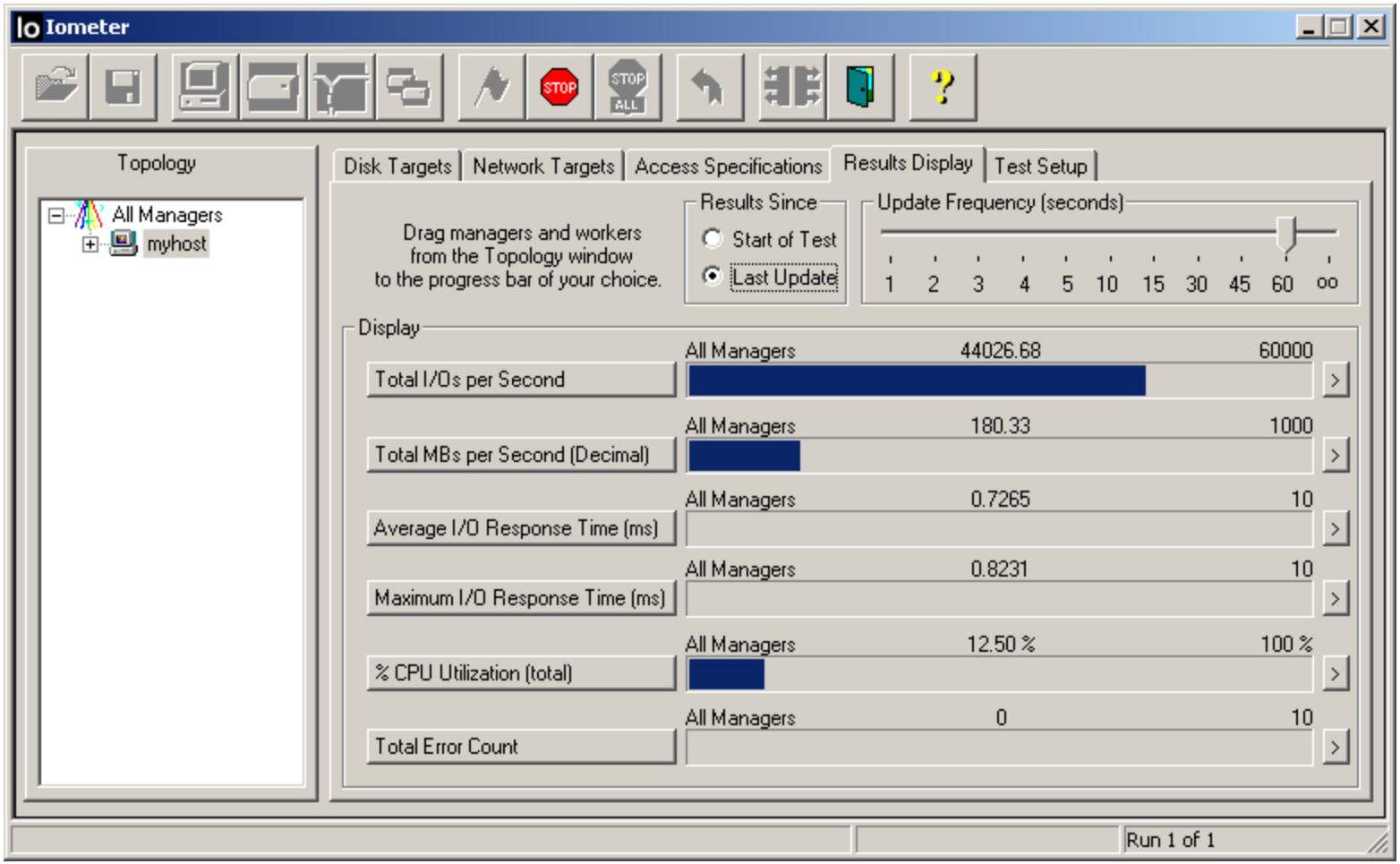
How do Flash/DRAM based SSDs random write IOPS compare?



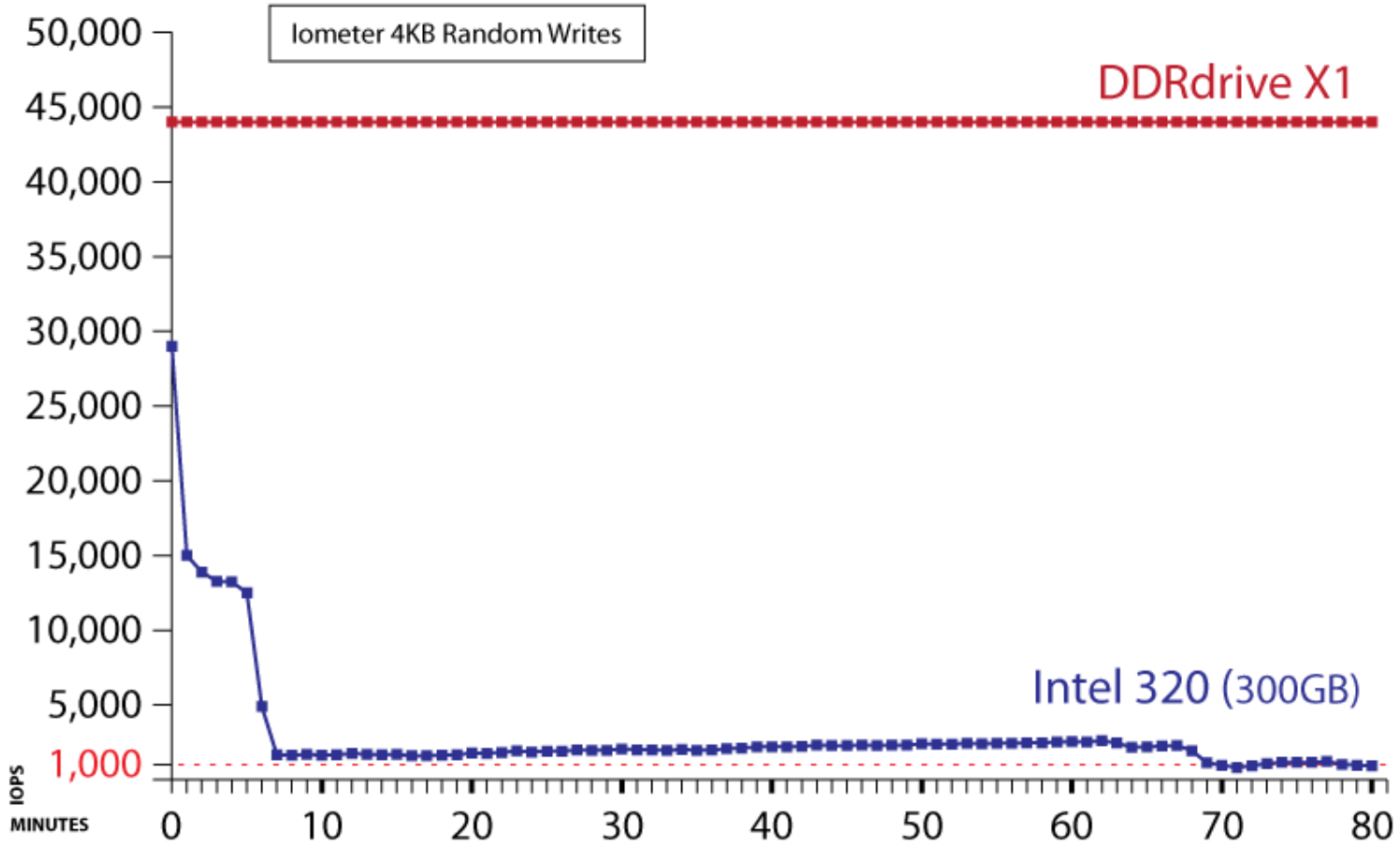
Intel 320 160GB - 4KB Random Write - 80 Minute Test Run:



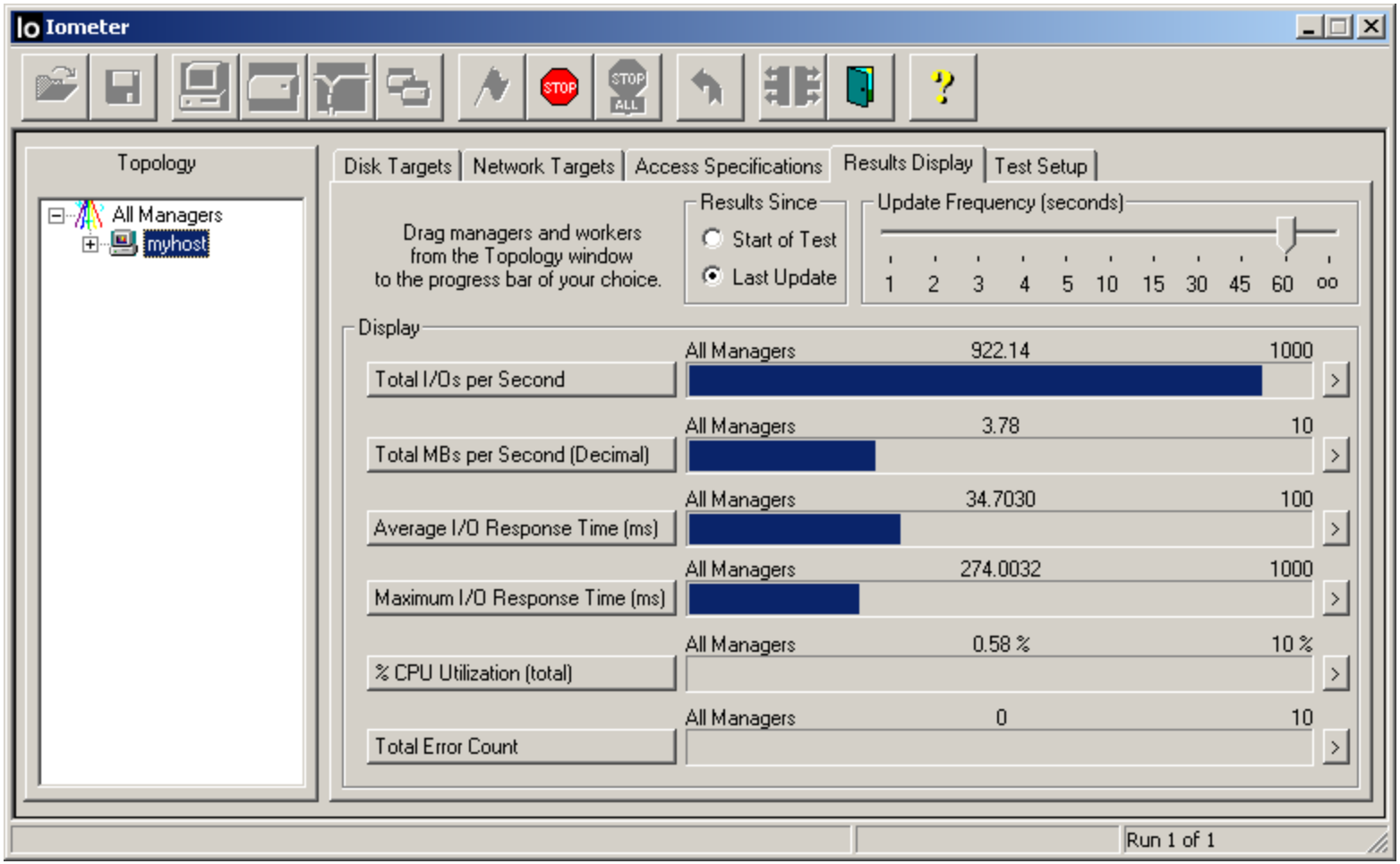
DDRdrive X1 - 4KB Random Write - 80 Minute Test Run:



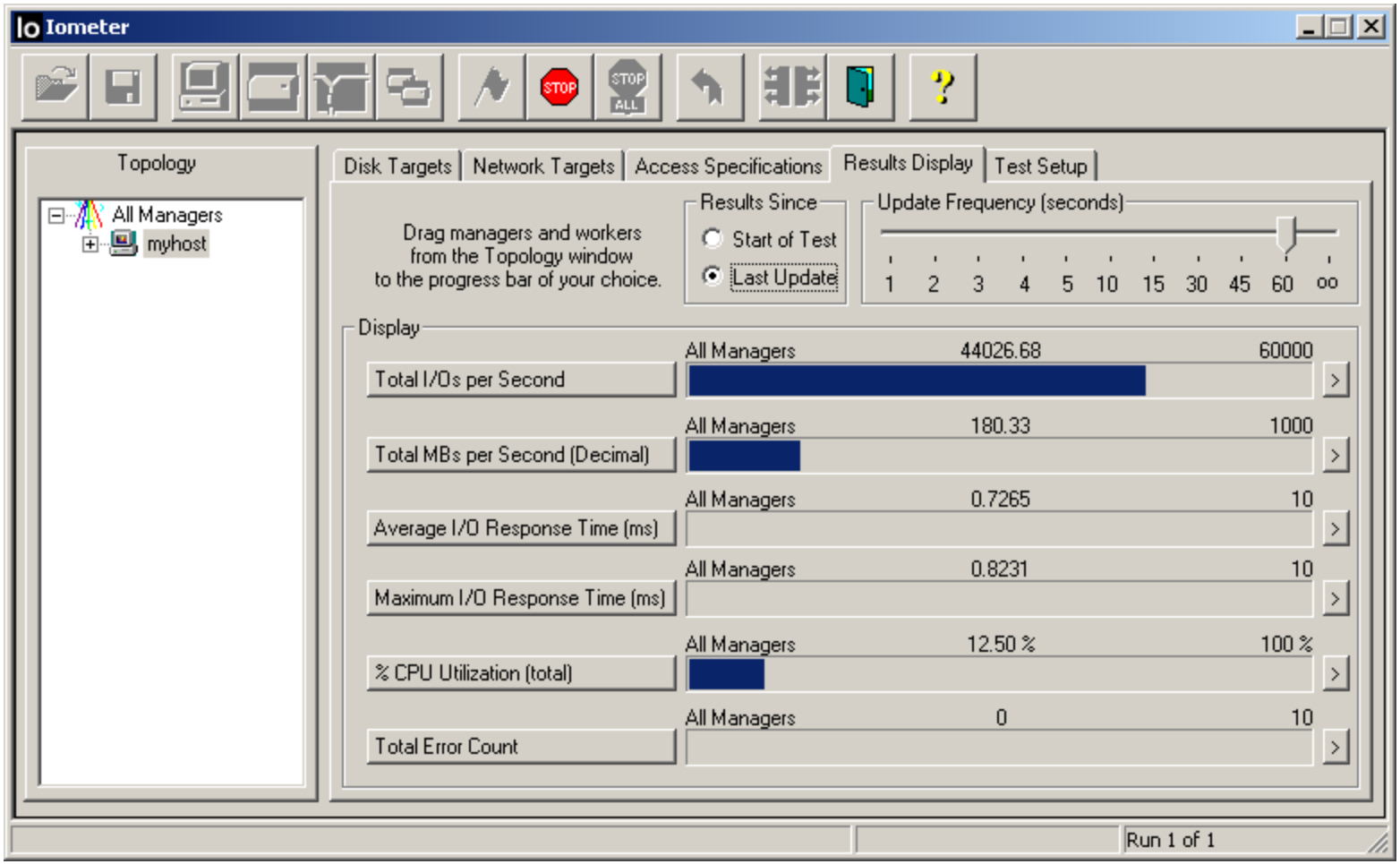
How do Flash/DRAM based SSDs random write IOPS compare?



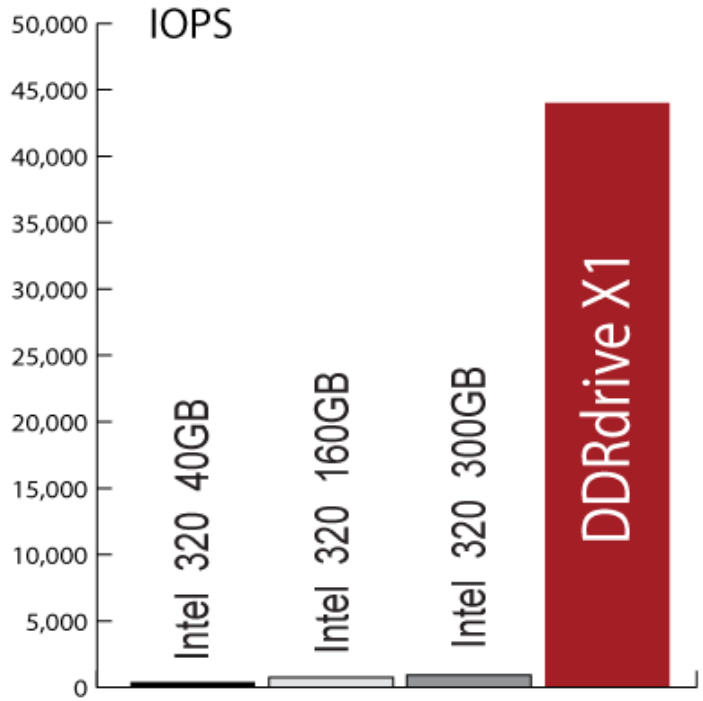
Intel 320 300GB - 4KB Random Write - 80 Minute Test Run:



DDRdrive X1 - 4KB Random Write - 80 Minute Test Run:



DDRdrive X1 / Intel 320 Series / 80 Minute Test Run Compare:



4KB Random Write

Log Device	IOPS
Intel 320 40GB	391
Intel 320 160GB	765
Intel 320 300GB	922
DDRdrive X1	44,026

How do Flash/DRAM based SSDs random write IOPS compare?

Flash based SSD Random Write IOPS:

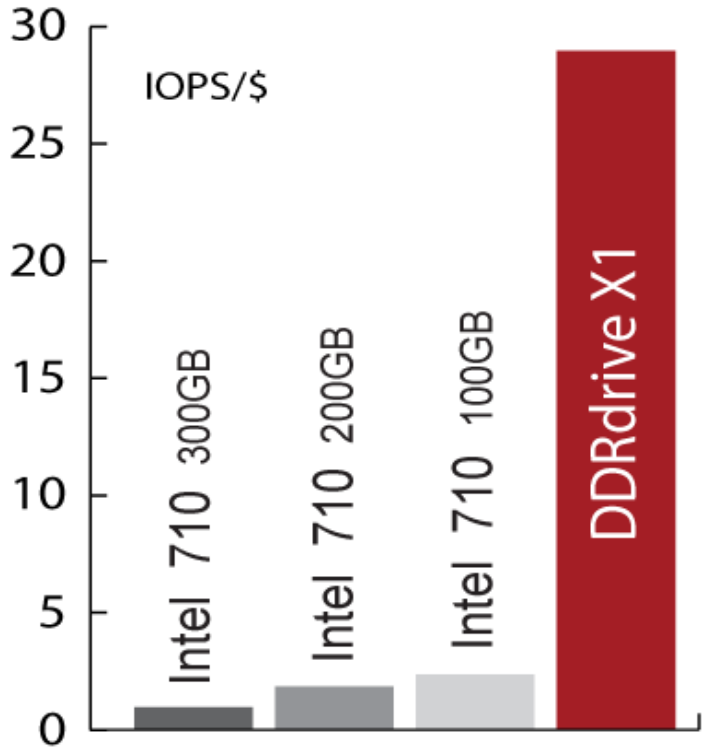
With typical ZIL Accelerator use, Flash based SSDs succumb to dramatic write IOPS ***degradation*** in less than 10 minutes after device is unpackaged or Secure Erased. The overall trend is ***not reversed*** with device inactivity. Contrast with a DRAM SSD (DDRdrive X1) where performance stays constant, not only over the entire product lifetime, but with any and all write IOPS workloads (random, sequential, mixed distributions).

In summary: The ***sustained*** write IOPS usage requirement of the ZIL Accelerator is in direct conflict with the random write IOPS inconsistency of a Flash based SSD.

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- **How do Flash/DRAM based SSDs IOPS/$ compare?**
- Are the finite write limitations of a Flash based SSD a concern?

How do Flash/DRAM based SSDs IOPS/$ compare?

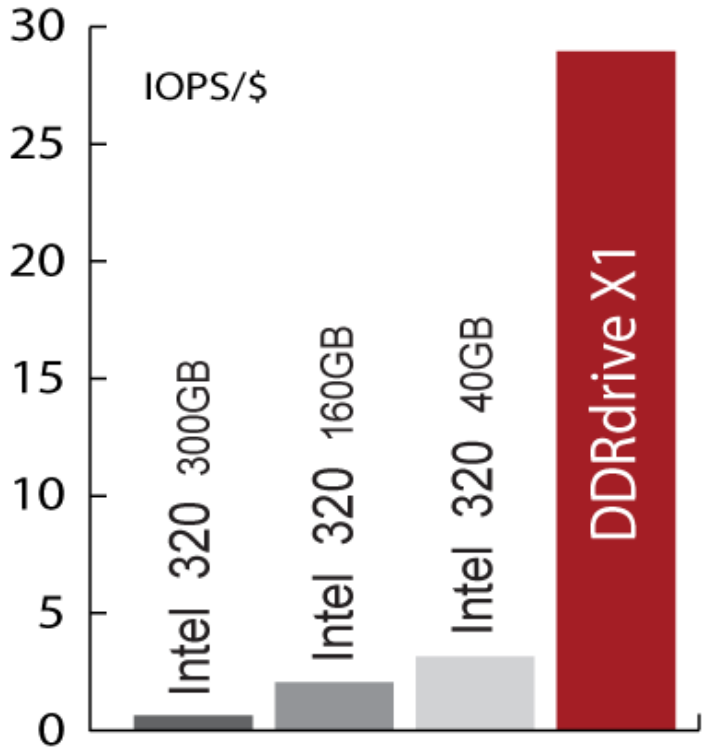


4KB Random Write

Log Device	IOPS/$ [1,2]
Intel 710 100GB	2.4
Intel 710 200GB	1.9
Intel 710 300GB	1.0
DDRdrive X1	29

[1] Amazon.com USD pricing as of 10-20-11.
[2] Iometer results 80 min post Secure Erase.

How do Flash/DRAM based SSDs IOPS/$ compare?



4KB Random Write

Log Device	IOPS/$ [1,2]
Intel 320 40GB	3.2
Intel 320 160GB	2.1
Intel 320 300GB	0.7
DDRdrive X1	29

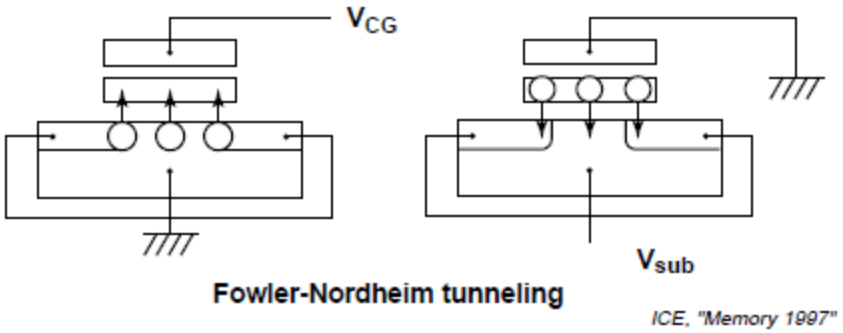
[1] Amazon.com USD pricing as of 10-20-11.
[2] Intel website published specifications.

Questions to be answered:

- What is the ZIL (ZFS Intent Log)?
- What are the key characteristics of a ZIL Accelerator?
- Why is ZIL Accelerator volatile cache power protection so critical?
- Which Intel SSDs have volatile cache protection and which do not?
- Is the ZIL Accelerator access pattern random and/or sequential?
- How do Flash/DRAM based SSDs random write IOPS compare?
- How do Flash/DRAM based SSDs IOPS/$ compare?
- **Are the finite write limitations of Flash based SSDs a concern?**

Thought Experiment: What is the underlying physics of Flash?

- The **ball** and the **table**.
- Quantum Mechanics. (quantum tunneling)
- The **electron** and the **barrier**.
- Fowler-Nordheim. (electron tunneling)
- Underlying process by which Flash writes (erase/program).



Fowler-Nordheim tunneling

ICE, "Memory 1997"

Are the finite write limitations of Flash based SSDs a concern?

[Excerpt from the Intel® SSD 710 Series Product Specification, dated September 2011.[1]]

Endurance Rating The number of bytes that may be written to the SSD such that the SSD meets the requirements according to the JESD218 specification.	**See Section 2.7, "Write Endurance"**

2.7 Write Endurance

Write endurance is measured while running 100% random 4KB (4,096) writes and 8KB (8,192) spanning 100% of the SSD using Iometer.

Table 9. Write Endurance Specifications

Intel SSD 710 Series	4 KB Writes	8 KB Writes
100 GB	Up to 500 TB	Up to 900 TB
200 GB	Up to 1.0 PB	Up to 1.0 PB
300 GB	Up to 1.1 PB	Up to 1.8 PB

[1] Includes table formatting changes and color highights not found in the original document.

Are the finite write limitations of Flash based SSDs a concern?

Intel® Solid-State Drive 710 Series

Limited Warranty with Media Wear-out Indicator

Intel warrants to the purchaser of the Product specified above in its original sealed packaging ("Original Purchaser") and to the purchaser of a computer system built by an Original Purchaser containing the Product ("Original System Customer") as follows: if the Product is properly used and installed, it will be free from defects in material and workmanship, and will substantially conform to Intel's publicly available specifications for the "warranty period", which is THE SHORTER OF: (A) A PERIOD OF THREE (3) YEARS BEGINNING ON THE DATE THE PRODUCT WAS PURCHASED IN ITS ORIGINAL SEALED PACKAGING IN THE CASE OF AN ORIGINAL PURCHASER OR THE DATE OF ORIGINAL PURCHASE OF A COMPUTER SYSTEM CONTAINING THE PRODUCT IN THE CASE OF AN ORIGINAL SYSTEM CUSTOMER; OR (B) THE PERIOD ENDING ON THE DATE WHEN THE USAGE OF THE DRIVE, AS MEASURED BY INTEL'S IMPLEMENTATION OF THE "SMART" ATTRIBUTE (E9) "MEDIA WEAR-OUT INDICATOR", REACHES A "NORMALIZED VALUE" OF "1", AS REPORTED BY THE INTEL® SSD TOOLBOX. The "Media Wear-out Indicator" (E9) is specified in the Intel datasheet for the Product, and can be accessed using the Intel® Solid-State Drive Toolbox software available as a free download from Intel. By using the Intel SSD Toolbox and clicking on the "Check SMART Attributes" button the user will find the E9 or "Media Wear-out Indicator" value. A new, unused drive will show a Media Wear-out Indicator value of "100", while a drive that has reached its write endurance limit will show a Media Wear-out Indicator value of "1". If the Product, which is the subject of this Limited Warranty, fails to conform to the above warranty during the warranty period, Intel, at its option, will:

Are the finite write limitations of Flash based SSDs a concern?

[Excerpt from the Intel® SSD 320 Series Product Specification, dated March 2011.[1]]

Minimum Useful Life / Endurance Rating The SSD will have a minimum of five years of useful life under typical client workloads with up to 20 GB of host writes per day.	5 years

[1] Includes table formatting changes and color highights not found in the original document.

[Excerpt from the Intel® SSD 710 debut Press Release, dated September 14, 2011.]

Referring to the newly released Intel SSD 710 Series, Rob Crooke Intel vice president and general manager of the Intel Non-Volatile Memory Solutions Group said "Our latest SSD product family offers more than 30 times the write endurance of our current MLC SSDs,..."

Can log device mirroring mitigate Flash based SSD write wear out?

Is there a solution to storage server downtime precipitated by a Flash based SSD (log device) failure resulting from Flash's finite write limit?

Mirrored log devices individually see the exact same IO activity and thus will wear out equally. With equal wear one would expect the finite write limit to also be approximately reached at the same time.

Keeping with the ZFS Best Practice of using whole disks instead of slices, one could mirror unequally sized devices. For example, mirror the Intel 710 100GB with either the 710 200GB or 300GB, although this involves both performance and cost tradeoffs.

A DRAM based solution to the finite write limitations of Flash:

DDRdrive X1:

No write IO wear of any kind, irrespective of write workload, thus unlimited writes for the **entire** device lifetime.

Never need to worry about possible storage server downtime resulting from an off-lined log device because of Flash's finite write limitations.

Onboard SLC Flash is *only* used during the automatic backup/restore function and is guaranteed for 100,000+ backups. Provides 27+ years of continuous operation even if the storage server was powered off once per day.

Summary of key differences between ZIL Accelerator SSD types?

* Intel 710 Series 3 Year Limited Warranty is further constrained by a Media Wear-out Indicator.	Intel 710 Series	DDRdrive X1
Sustained/Consistent Random Write IOPS		+
Random Write IOPS Degradation/Variability	×	
Write results regardless of IO Workload.		+
Write results regardless of IO Alignment.		+
Write results regardless of Partition Alignment.		+
Write results regardless of ZFS TRIM Support.		+
Random Write IOPS / Dollar ($)		+
Product Longevity / Durability		+
Product Warranty	3* YEAR	5 YEAR
Requires Separate HBA/Controller	+	

Thank you!

www.ddrdrive.com