Getting the most out of your NVMe SSD

Charles Rose

Lakshmi Narayanan Durairajan

Red Hat Summit

June 2015

Agenda

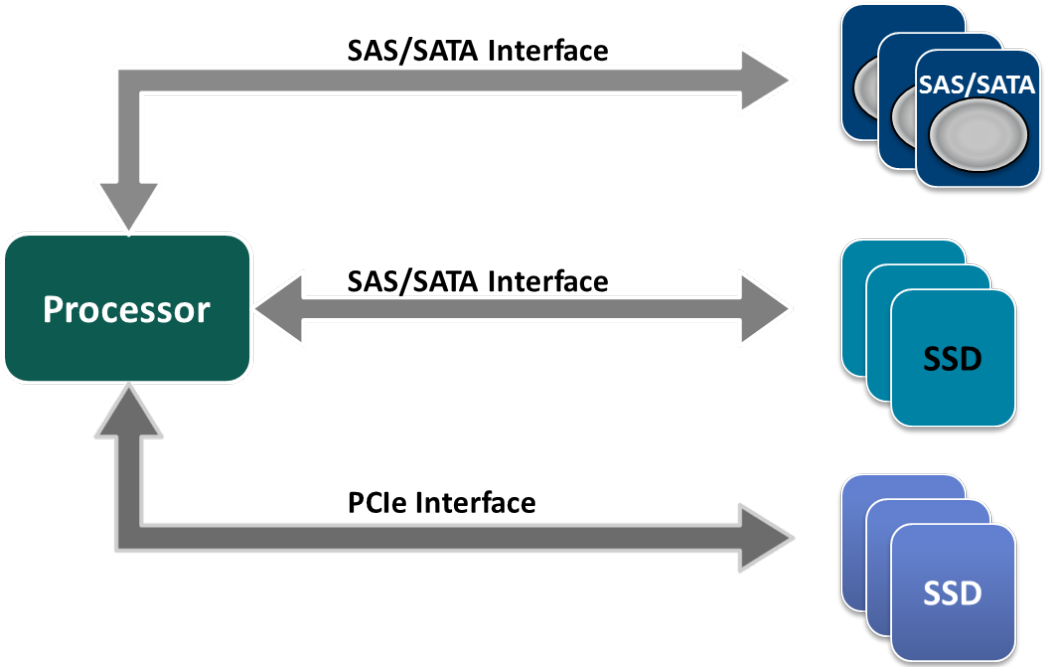
- NVMe – A Brief Introduction
  - What is NVMe
  - Why NVMe
- Overview of NVMe Components
- NVMe SSD vs. SAS SSD
- NVMe Usage Scenarios and Performance
  - File Systems on NVMe
  - dm-cache
  - Mariadb

NVMe – A Brief Introduction

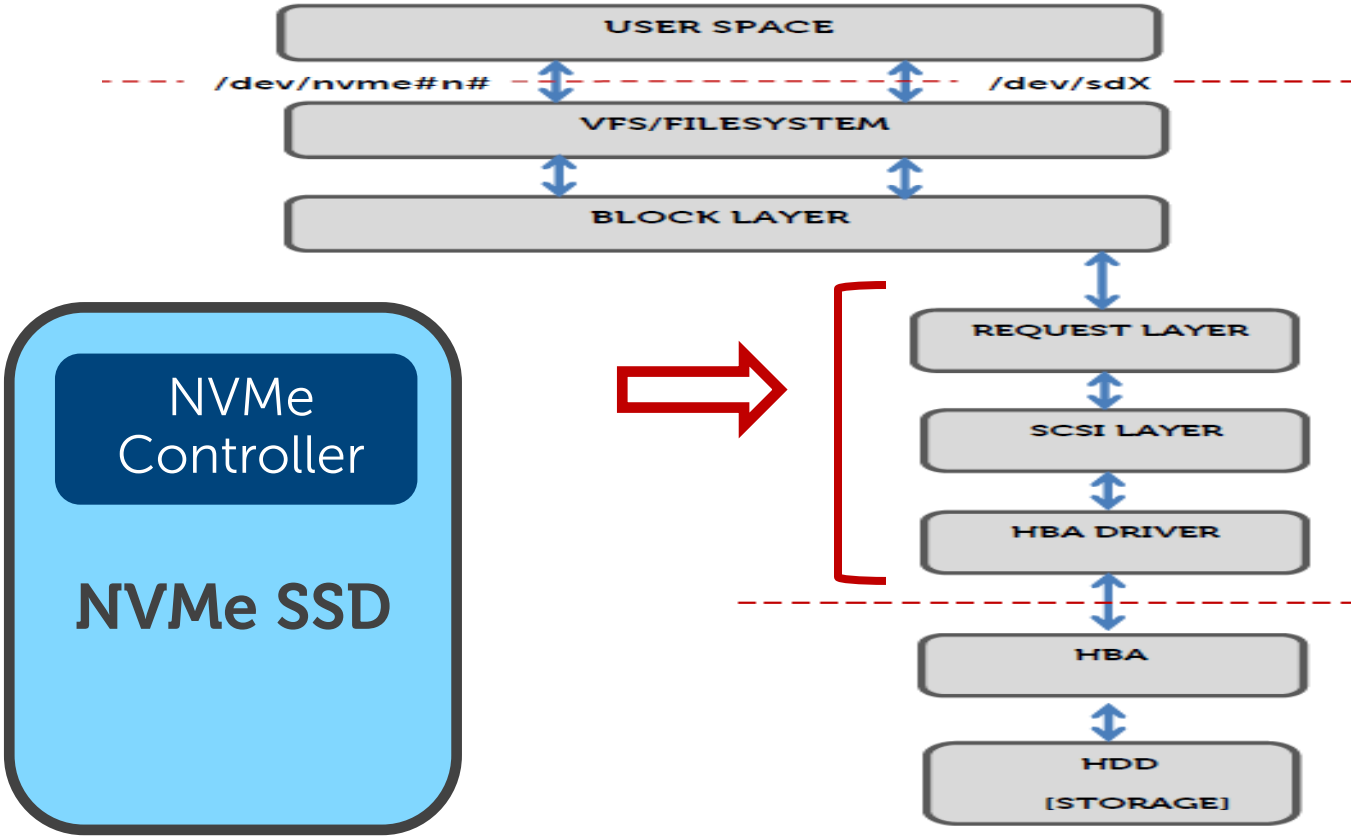
NVMe (Non-Volatile Memory Express)

- A **specification** for accessing solid-state drives (**SSDs**) attached through the PCI Express (**PCIe**) bus.
- Specification defines an optimized **register interface, command set** and **feature set**.
- PCIe SSD devices designed based on the NVMe specification are called **NVMe-based PCIe SSD's**
- Provides a scalable host controller interface for devices in various form-factors, from client to enterprise servers.
- Focus is to **standardize PCIe SSD Interface** and **maximize performance** of PCIe SSDs.

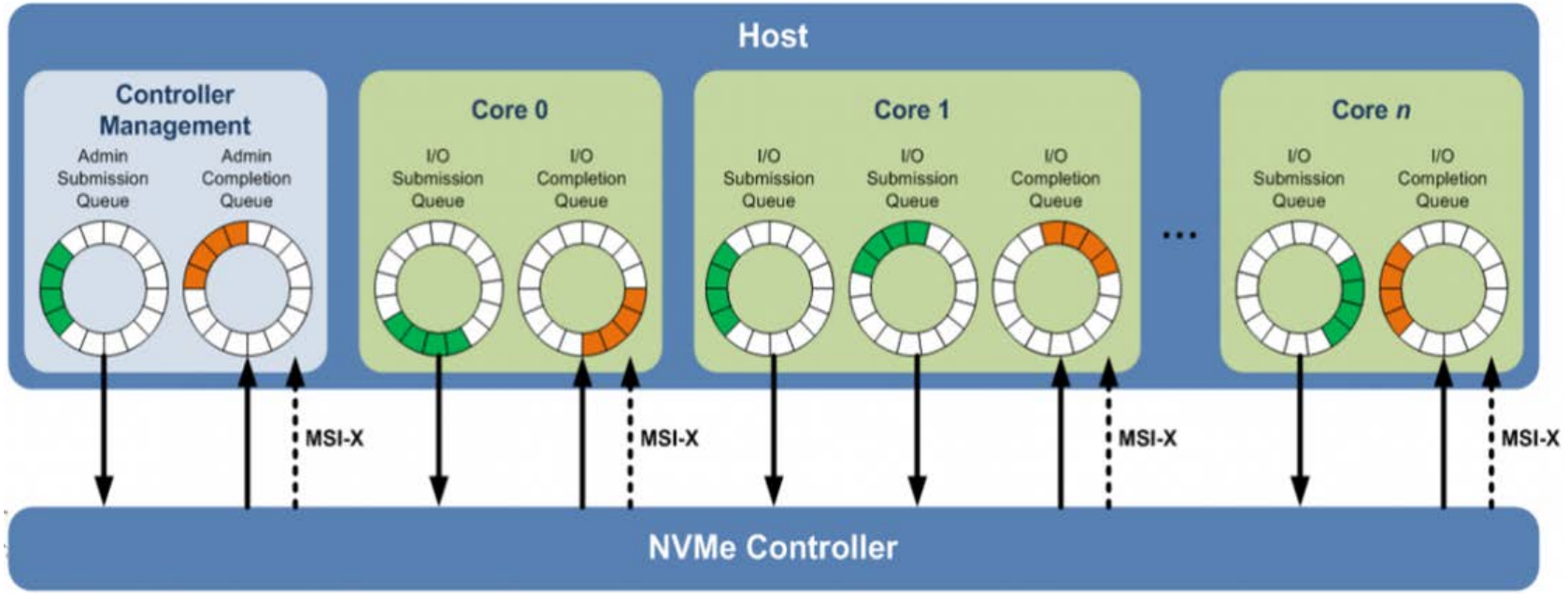
Why NVMe – Evolution



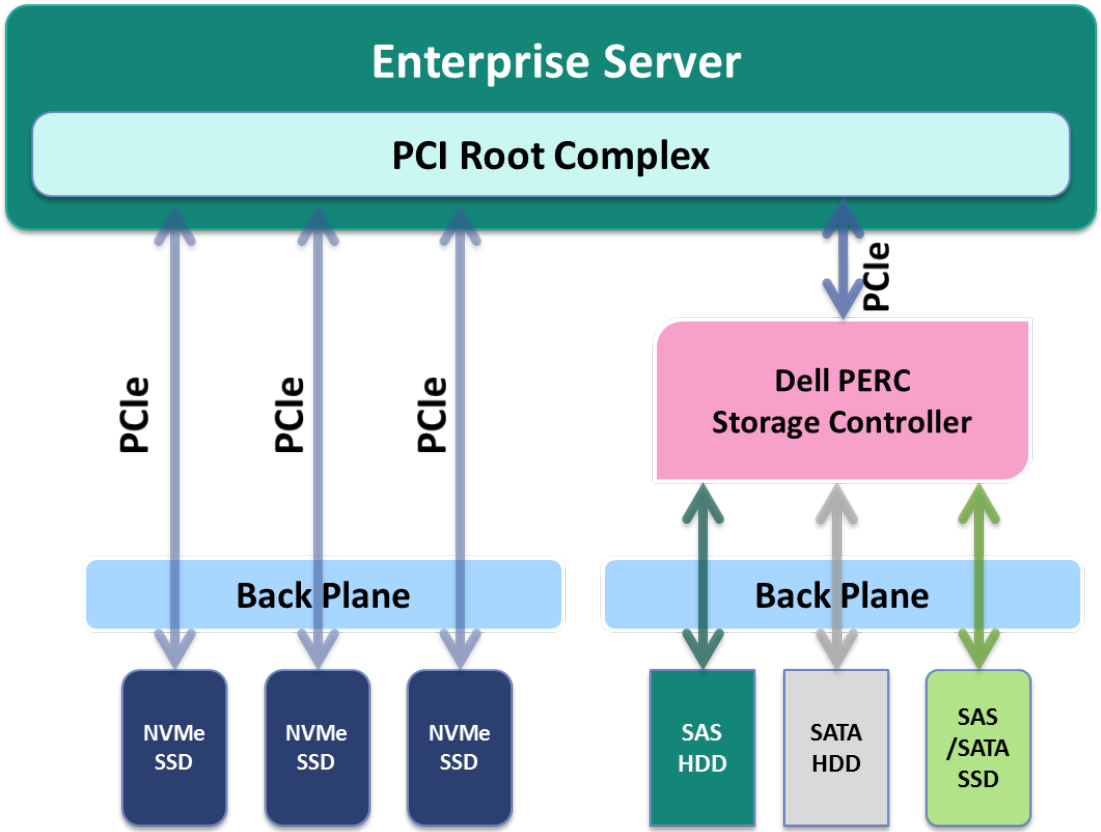
Linux Driver Stack – Problem



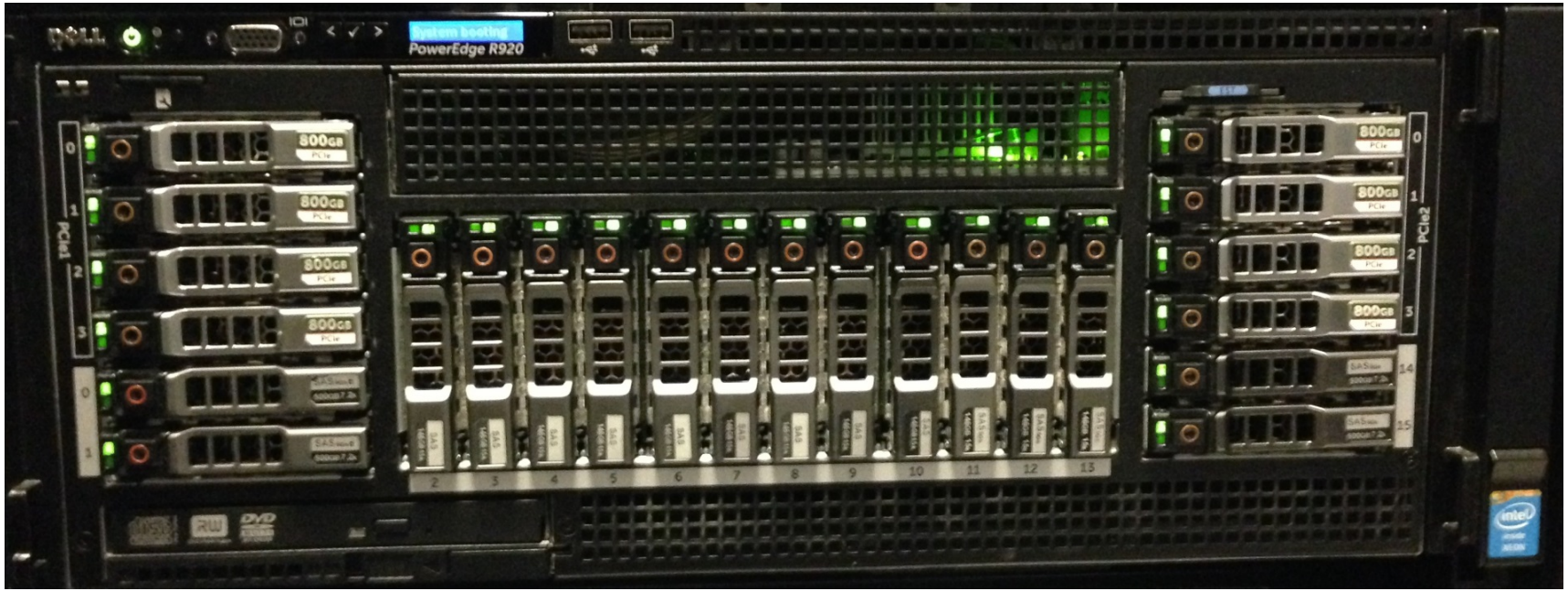
Overview of NVMe Components



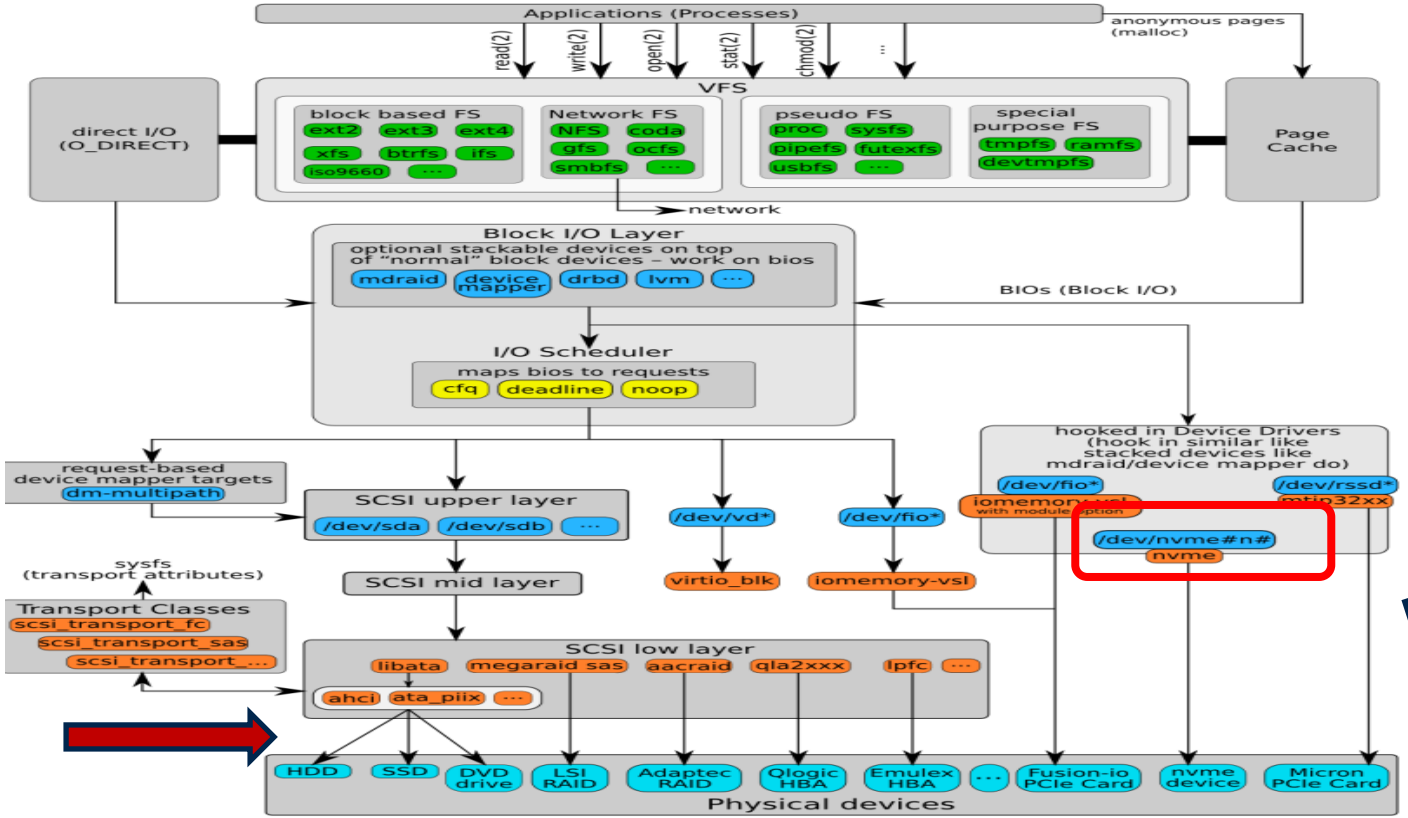
Dell PowerEdge Storage Layout



NVMe Hotplug Drives on Dell PowerEdge

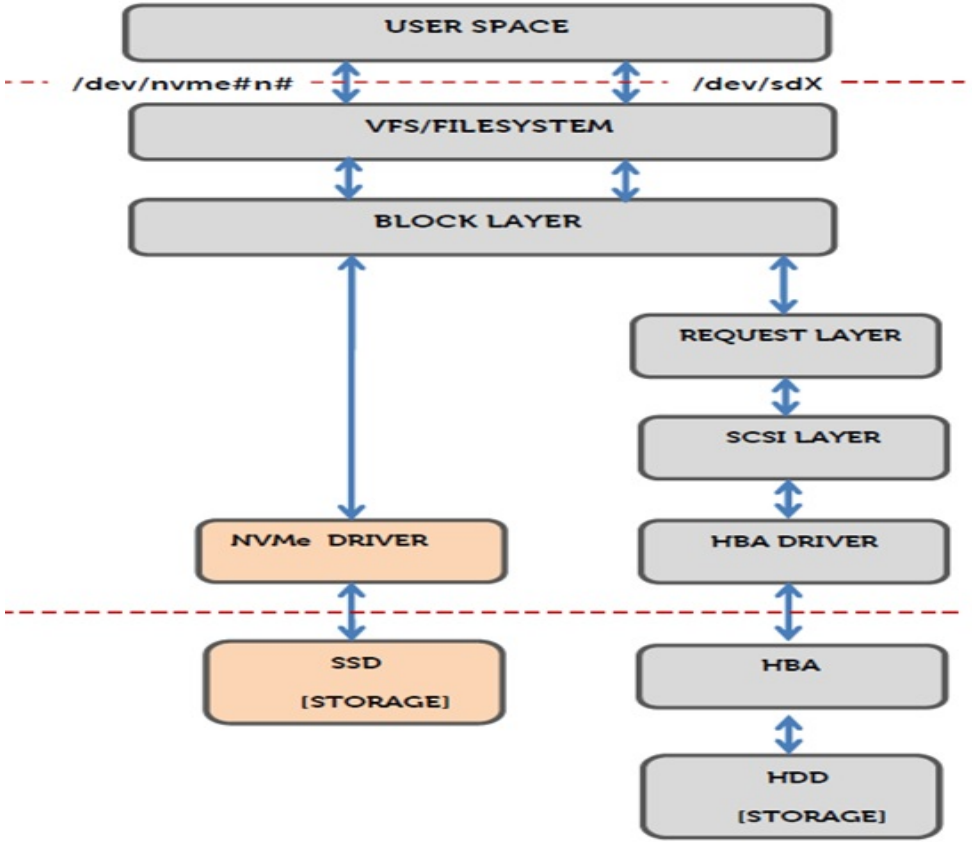


Linux IO Path Comparison

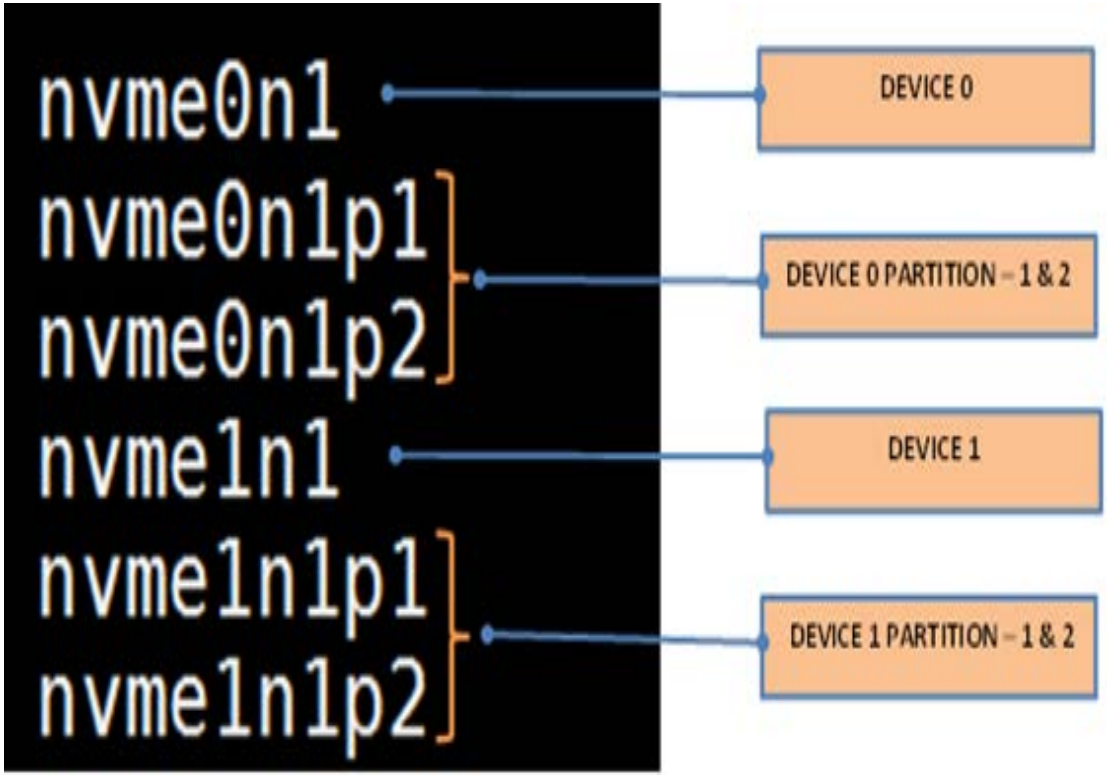


The Linux I/O Stack Diagram (version 0.1, 2012-03-06)
http://www.thomas-krenn.com/en/oss/linux-io-stack-diagram.html
Created by Werner Fischer and Georg Schönberger
License: CC-BY-SA 3.0, see http://creativecommons.org/licenses/by-sa/3.0/

NVMe Driver in Linux Stack



NVMe Device Node



NVMe Performance Scenarios

Block Storage Device for File System

Caching

Storage Device for Data base

File Systems on NVMe

File System Benchmark: Setup

- System
  - PowerEdge R730xd, 64G
  - Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz
  - PERC H730P Mini (Embedded), Seagate 300G 6Gbps SAS
  - Dell Express Flash NVMe XS1715 SSD 400GB
  - RHEL 7.1
- Tunables
  - # blockdev --getra | --setra
  - Set /sys/block/XXX/queue/rq_affinity to "1"

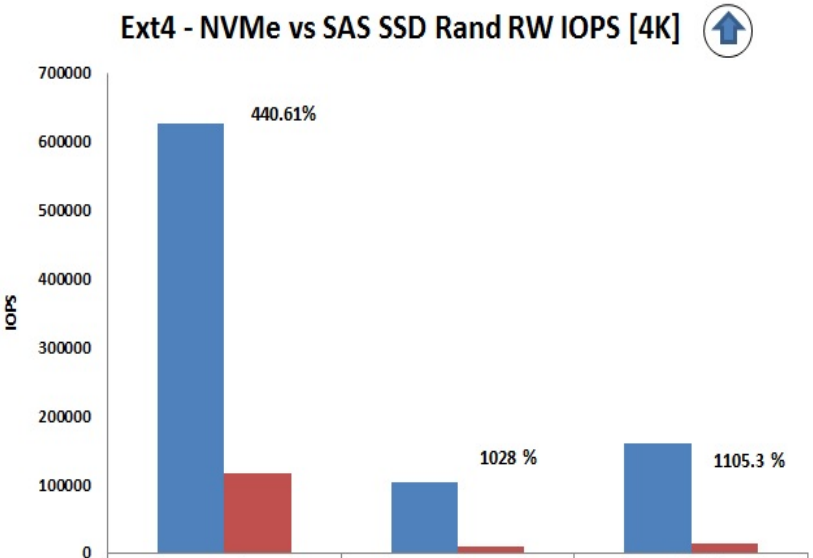
`iozone –a –g 4G –b file.xls`

- a = Auto mode
- g = Set maximum file size (KB)
- b = Filename Create Excel worksheet file

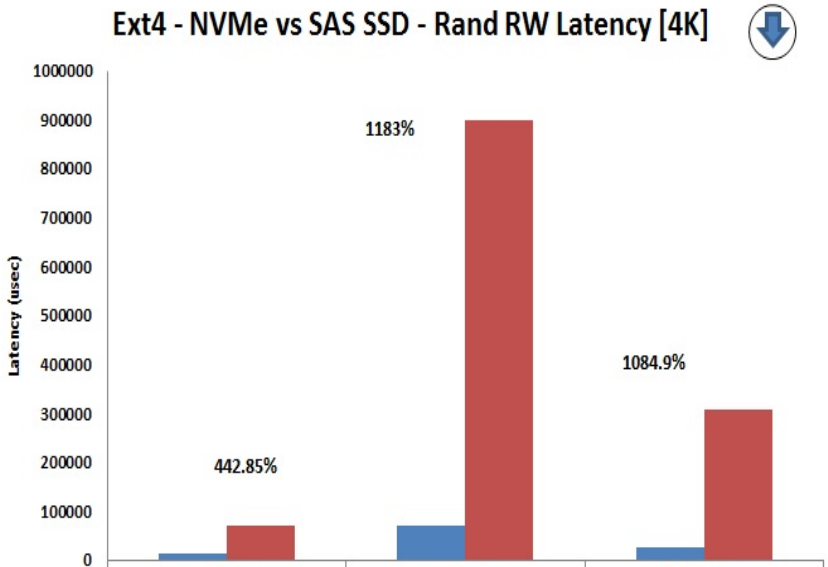
`nvme_4k_read.fio:`

```
[global]
bs=4k
ioengine=libaio
iodepth=512
size=4g
direct=1
directory=/mnt/nvme
name=nvme_4k_read
numjobs=16
group_reporting
rw=randread
```

NVMe vs SAS SSD: ext4

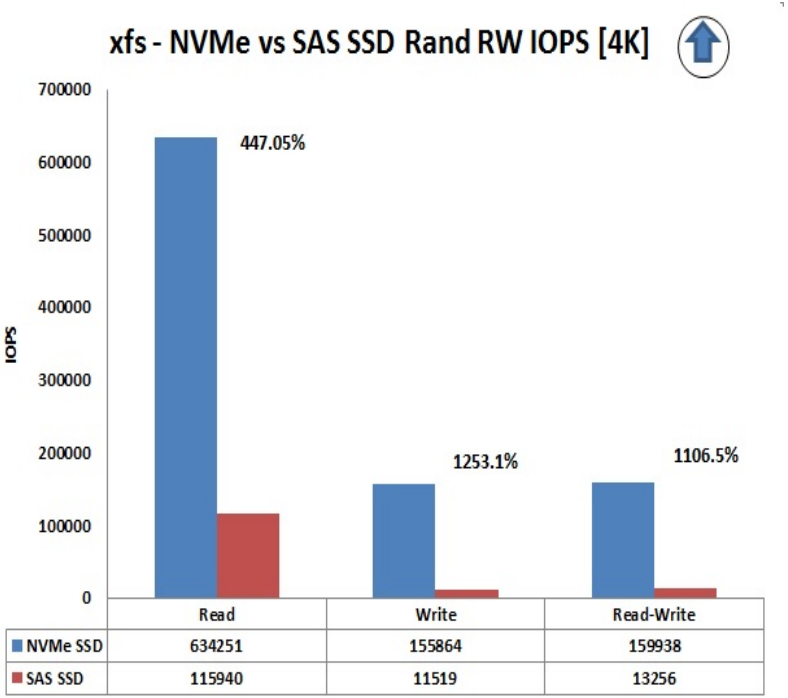


	Read	Write	Read-Write
NVMe SSD	627044	102965	159737
SAS SSD	115988	9121	13252

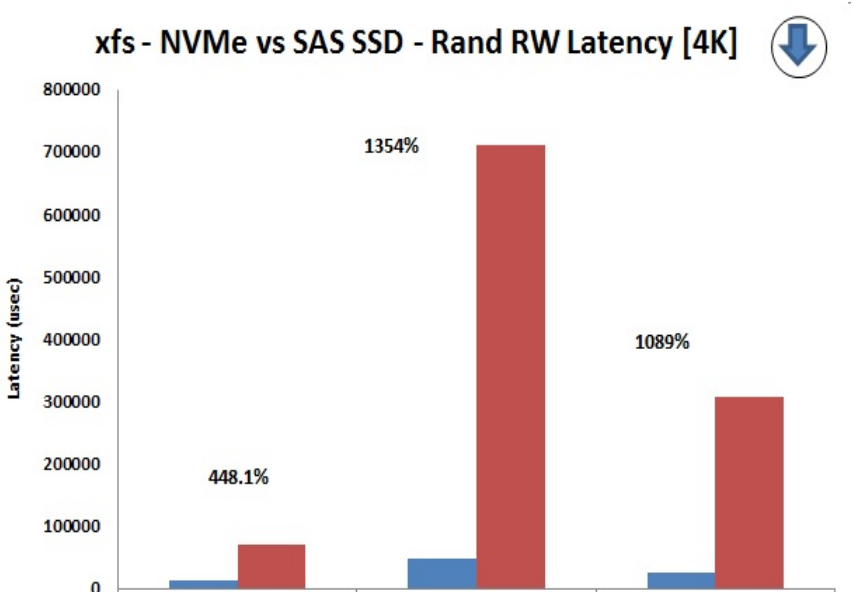


	Read	Write	Read-Write
NVMe SSD	12970.3	69964.43	25872.23
SAS SSD	70410	897650	306580

NVMe vs SAS SSD: xfs

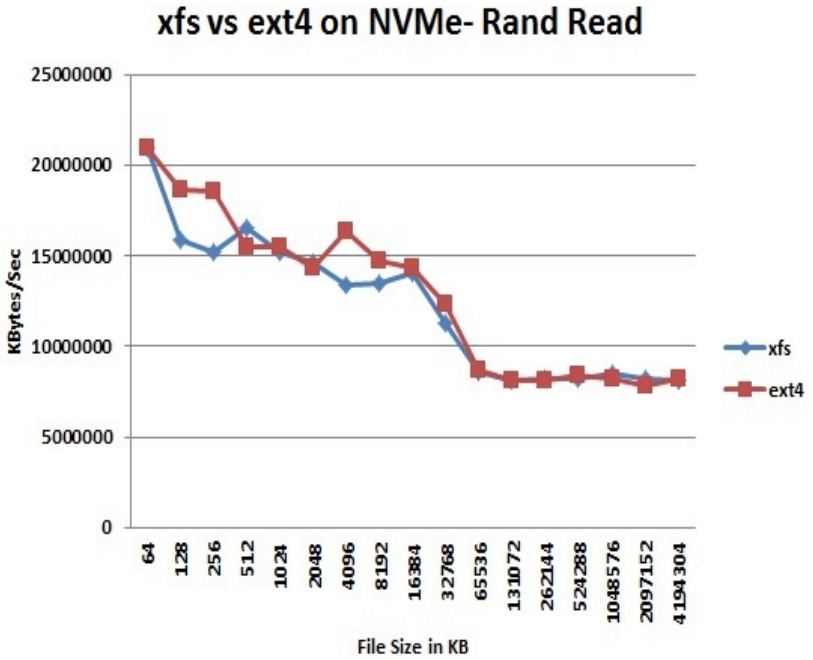


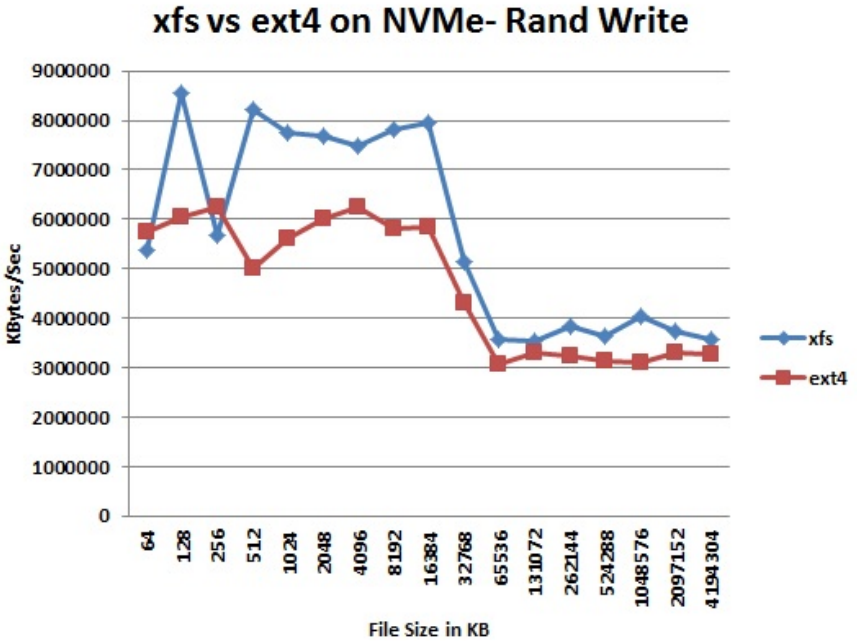
	Read	Write	Read-Write
NVMe SSD	634251	155864	159938
SAS SSD	115940	11519	13256



	Read	Write	Read-Write
NVMe SSD	12865.93	48812.1	25781.15
SAS SSD	70520	709760	306610

NVMe: xfs vs ext4 (IOzone)



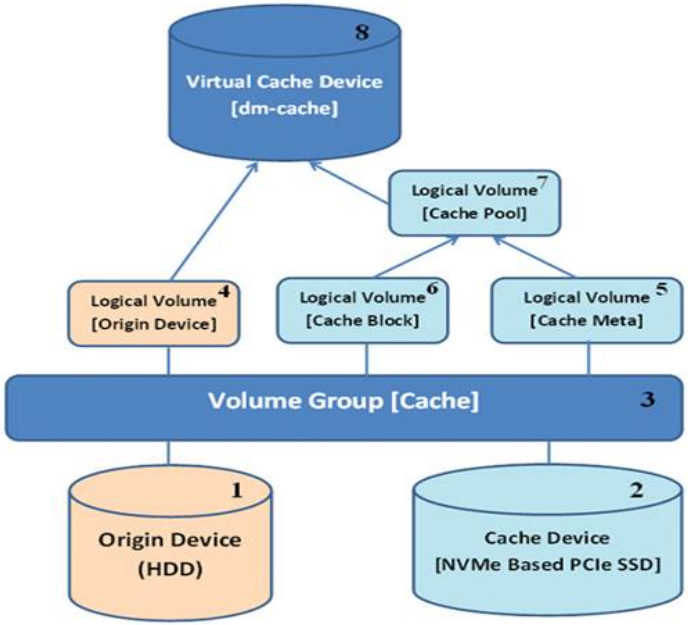


dm-cache

dm-cache: Overview

- Device Mapper Target.
- Tech Preview in RHEL 7.0; Supported in RHEL 7.1
- Setup/Teardown with “lvconvert” command line.
- Cache Pool: SSD split for Metadata volume and Cache Data volume.
- Chunk Size Fixed: Unit of cache promotion/demotion with Origin Device.
  - Between 32k and 1G.
  - Try with different sizes to address need.
- Operating Modes
  - Writeback
  - Writethrough
  - Passthrough

dm-cache: Setup Process



1. Create a physical volume on the hard disk [Origin Device]
2. Create a physical volume on the NVMe PCIe SSD disk [Cache Device]
3. Create a volume group [cache] with the hard disk [Origin Device] and NVMe PCIe SSD disk [Cache Device]
4. Create a logical volume [origin_device] on the hard disk.
5. Create a logical volume [cache_meta] on the NVMe SSD device.
6. Create a logical volume [cache_block] on the NVMe SSD device.
7. Create a cache pool with the cache_block and cache_meta volumes .
8. Enabling cache pool to cache origin_device.
9. Format the Virtual cache device with a file system and use it

dm-cache: Setup Cache

dmcache with NVMe SSD

```
# pvcreate /dev/sde
# pvcreate /dev/nvme1n1
# vgcreate cache /dev/sde /dev/nvme1n1
# lvcreate -L 200G -n origin_device cache/dev/sde
# lvcreate -L 60G -n cache_block cache /dev/nvme1n1
# lvcreate -L 2G -n cache_meta cache /dev/nvme1n1
```

dmcache with SAS SSD

```
# pvcreate /dev/sde
# pvcreate /dev/sdb
# vgcreate cache /dev/sde /dev/sdb
# lvcreate -L 200G -n origin_device cache /dev/sde
# lvcreate -L 60G -n cache_block cache /dev/sdb
# lvcreate -L 2G -n cache_meta cache /dev/sdb
```

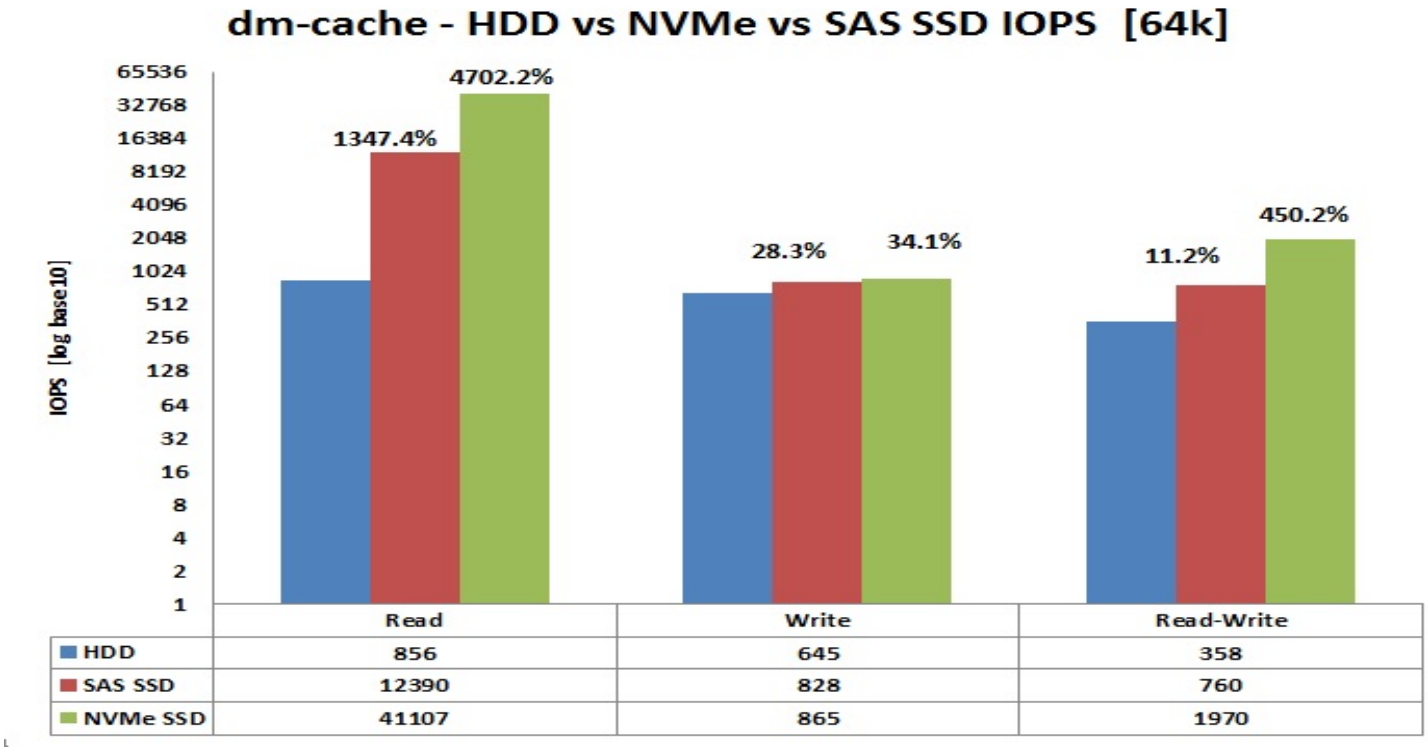
```
# lvconvert --type cache-pool /dev/cache/cache_block --poolmetadata /dev/cache/cache_meta
# lvconvert --type cache /dev/cache/origin_device --cachepool /dev/cache/cache_block
# mkfs –t xfs /dev/cache/origin_device
```

dm-cache: Verify Cache Setup

```
login as: root
root@10.9.166.106's password:
Last login: Wed Jun 24 07:57:03 2015 from 10.132.103.211
[root@dhcp-166-106 ~]# lvs --all
  LV                VG              Attr       LSize   Pool        Origin           Data%  Meta%  Move Log Cpy%Sync Convert
  [cachedisk]       cache           Cwi---C---  70.00g                              0.45   0.22             0.00
  [cachedisk_cdata] cache           Cwi-ao----  70.00g
  [cachedisk_cmeta] cache           ewi-ao----   2.00g
  dbcache           cache           -wi-a----- 300.00g
  [lvol0_pmspare]   cache           ewi-------   2.00g
  slowdisk          cache           Cwi-aoC--- 270.00g [cachedisk] [slowdisk_corig] 0.45   0.22             0.00
  [slowdisk_corig]  cache           owi-aoC--- 270.00g
```

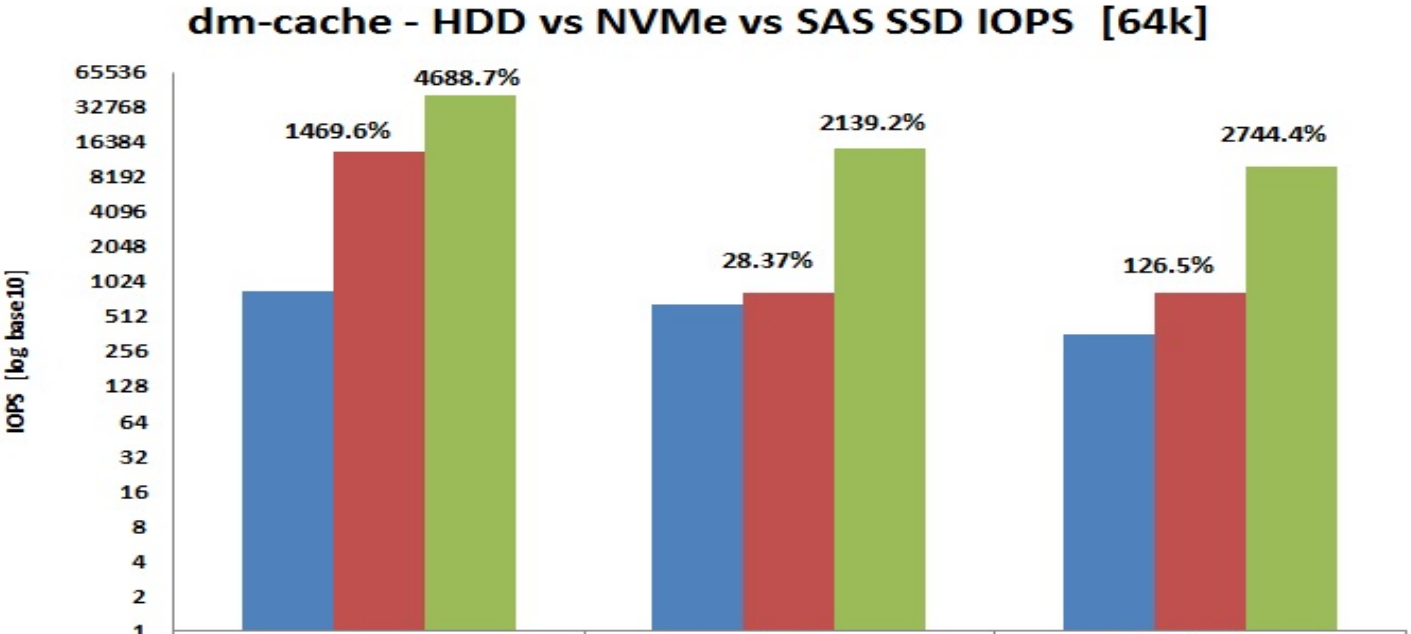
```
[root@dhcp-166-106 dm]# lsblk /dev/sdb /dev/nvme0n1
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sdb                       8:16   0 279.4G  0 disk
└─cache-slowdisk_corig  253:4    0   270G  0 lvm
  └─cache-slowdisk      253:6    0   270G  0 lvm  /perf1
nvme0n1                 259:0    0 372.6G  0 disk
├─cache-cachedisk_cdata 253:2    0    70G  0 lvm
│ └─cache-slowdisk      253:6    0   270G  0 lvm  /perf1
├─cache-cachedisk_cmeta 253:3    0     2G  0 lvm
│ └─cache-slowdisk      253:6    0   270G  0 lvm  /perf1
└─cache-dbcache         253:7    0   300G  0 lvm
```

dm-cache: Writethrough, XFS



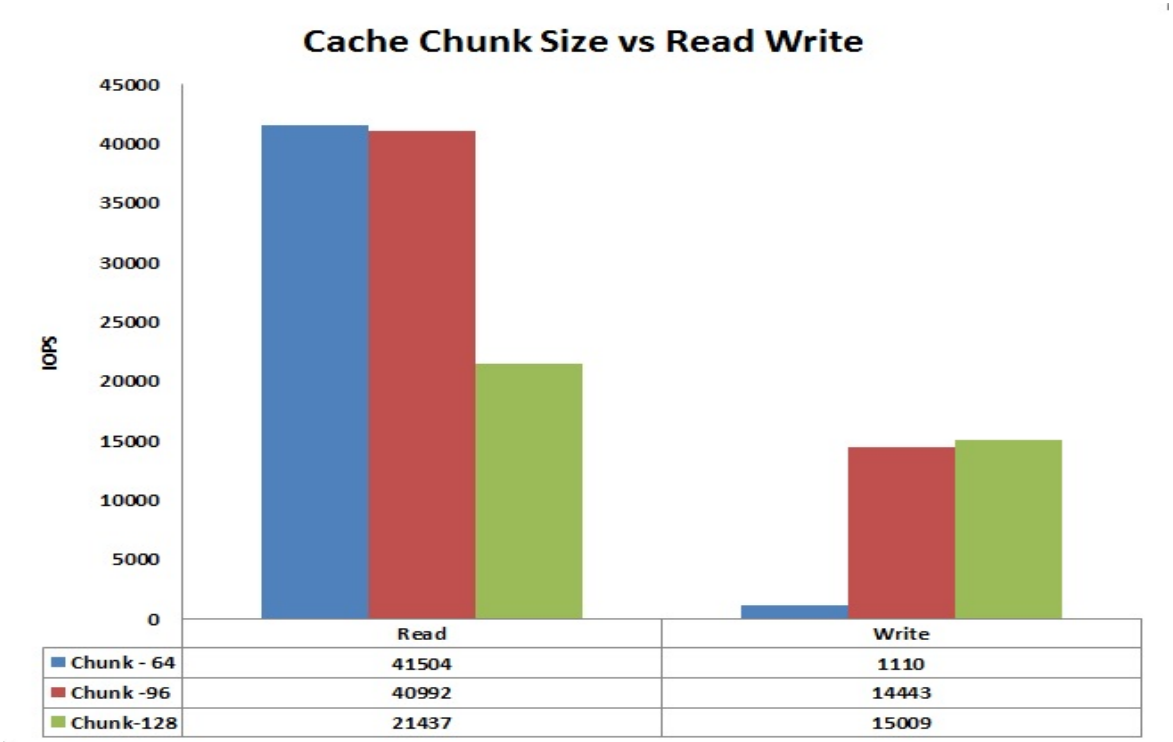
	Read	Write	Read-Write
HDD	856	645	358
SAS SSD	12390	828	760
NVMe SSD	41107	865	1970

dm-cache: Writeback, XFS



	Read	Write	Read-Write
HDD	856	645	358
SAS SSD	13436	828	811
NVMe SSD	40992	14443	10183

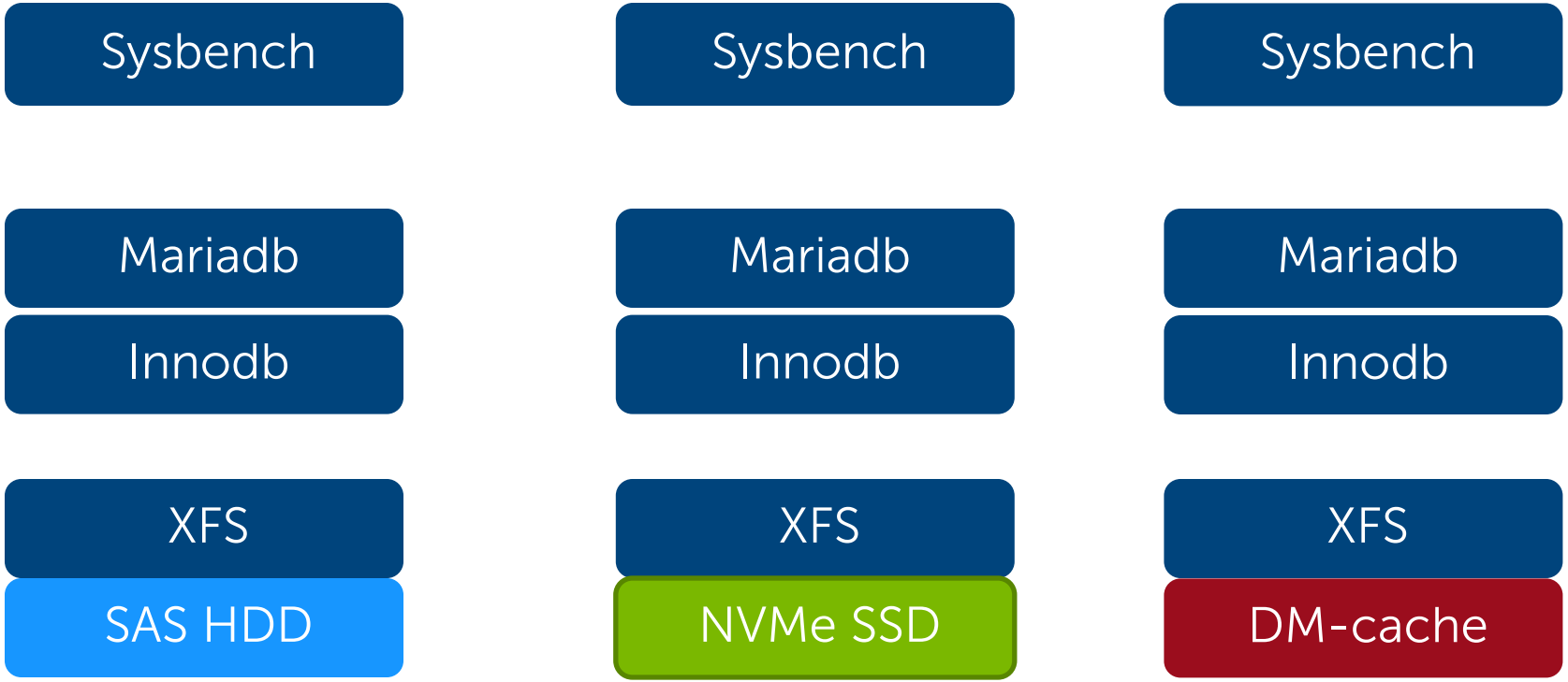
dm-cache: Chunk Size, XFS



	Read	Write
Chunk - 64	41504	1110
Chunk -96	40992	14443
Chunk-128	21437	15009

MariaDB

MariaDB: Test Setup



MariaDB: Configuration

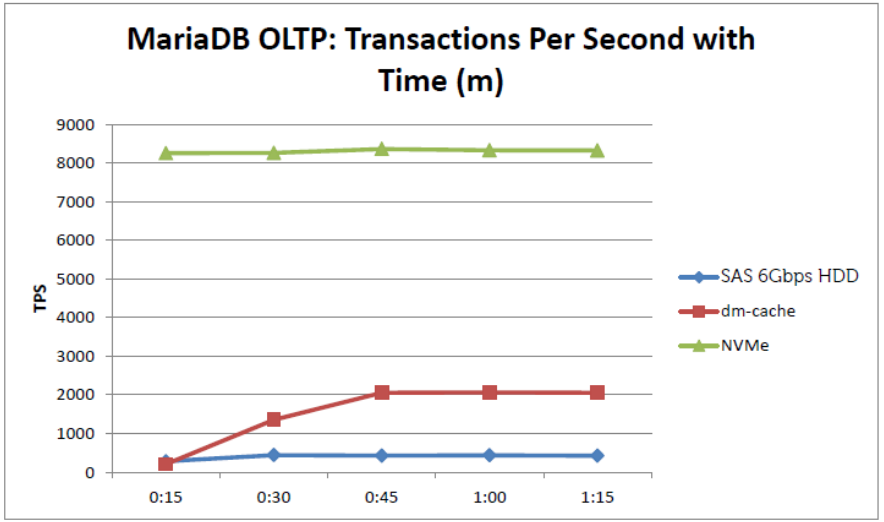
- mariadb-5.5.41-2.el7_0
- Configuration*
  - `innodb_data_file_path = ibdata1:128M:autoextend`
  - `innodb_file_per_table = OFF`
  - `innodb_buffer_pool_size = 8192M`
  - `innodb_additional_mem_pool_size = 32M`
  - `innodb_log_file_size = 32G`
  - `innodb_log_buffer_size = 16M`
  - `innodb_flush_log_at_trx_commit = 0`
  - `innodb_lock_wait_timeout = 50`
  - `innodb_doublewrite = 0`
  - `innodb_flush_method = O_DIRECT`
  - `innodb_thread_concurrency = 1000`
  - `innodb_max_dirty_pages_pct = 80`

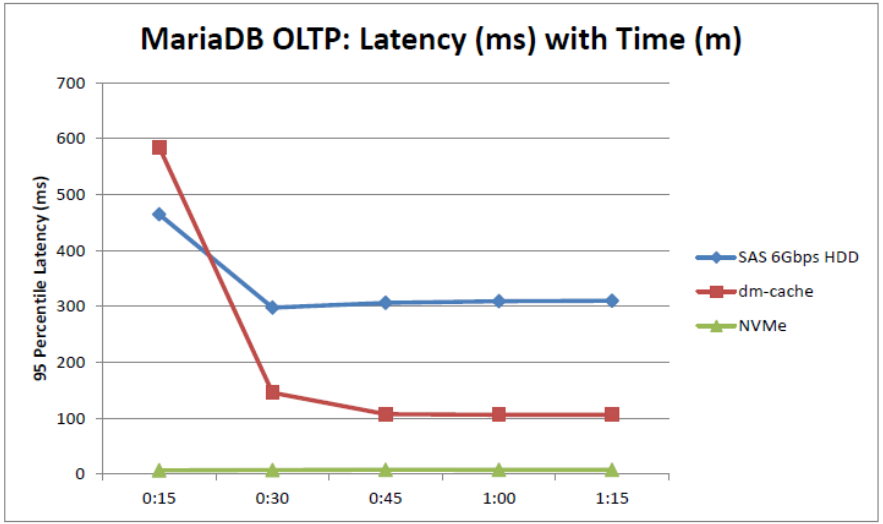
- System
  - Dell PowerEdge R630, 128G
  - PERC H330 Mini, Seagate 300G 6Gbps SAS
  - Dell Express Flash NVMe XS1715 SSD 400GB
  - RHEL 7.1
- Sysbench*

```
sysbench \
--test=oltp \
--mysql-table-engine=innodb \
--mysql-user=sysbench \
--mysql-password=password \
--mysql-db=sysbench \
--max-time=900 \
--num-threads=32 \
--oltp-table-size=100000000 \
--max-requests=100000000 \
```

*Thanks to Sanjay Rao (Red Hat)

MariaDB: Performance





Key Takeaways

- Main goal of NVMe is to scale performance and standardize the PCIe SSD Interface
- NVMe can be used as local storage or as cache for slower storage devices
- Nvme performance:
  - File system: when compared to SAS SSD by 400%
  - Cache device: when compared to SAS 12Gpbs HDD by 450% (Read/Write) to 4702 % (Read)
  - OLTP workload on NVMe: Improves by 18 times
  - OLTP workload on dm-cache: Improves by 4 times; Opportunity to tweak dm-cache tunables further.

Reference/Credits

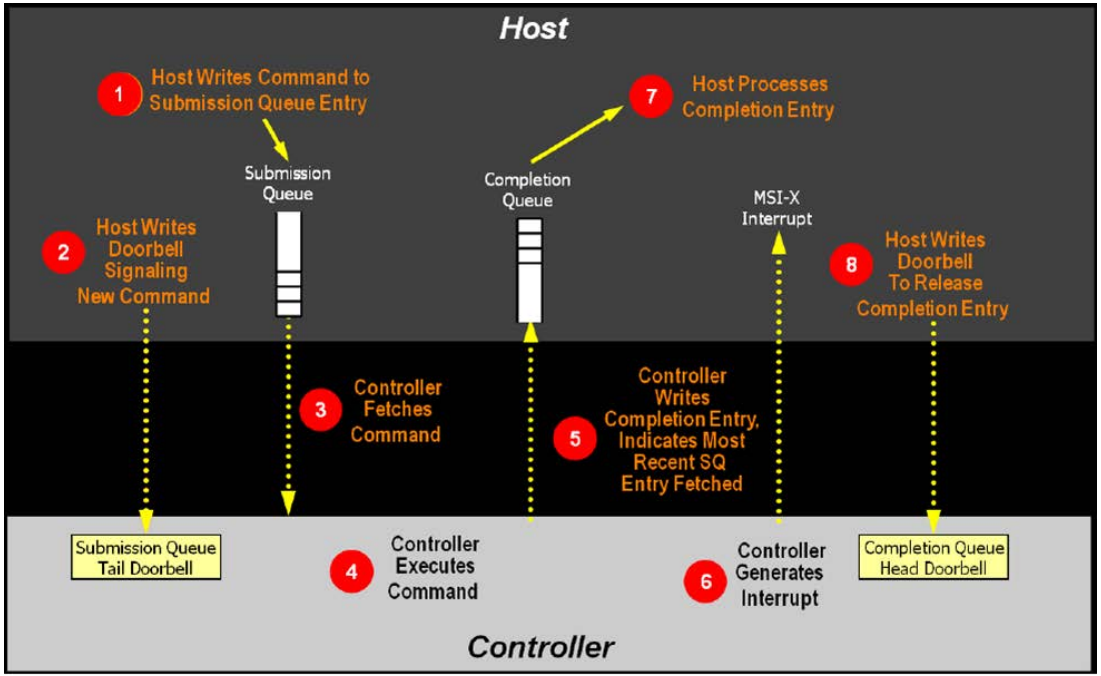
- NVMe Specification
  - http://www.nvmexpress.org/
- Sysbench
  - https://dev.mysql.com/downloads/benchmarks.html
- IOzone
  - http://www.iozone.org/

- Credits:
  - Sanjay Rao, Red Hat
  - Ben England, Red Hat

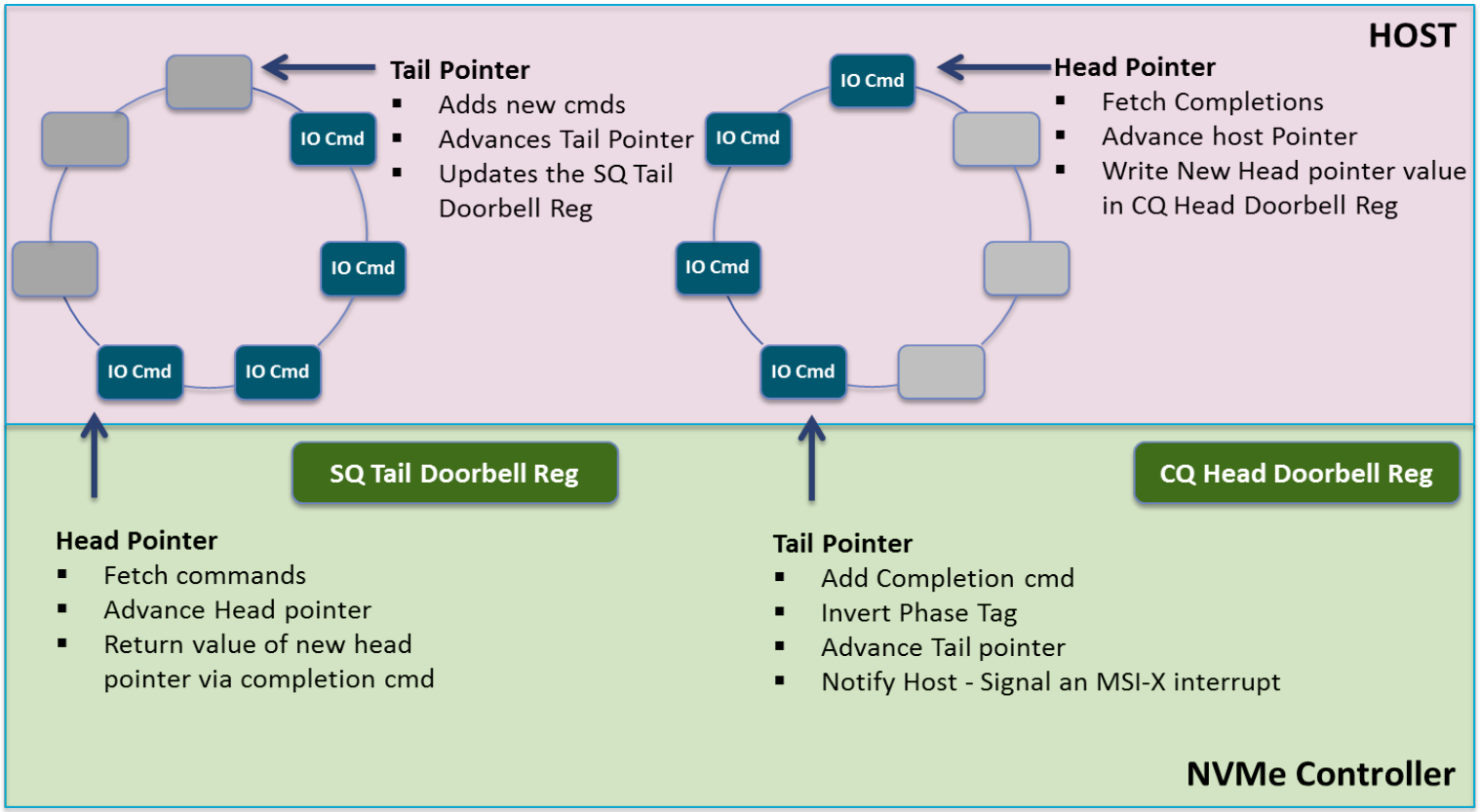
Thank You

Backup

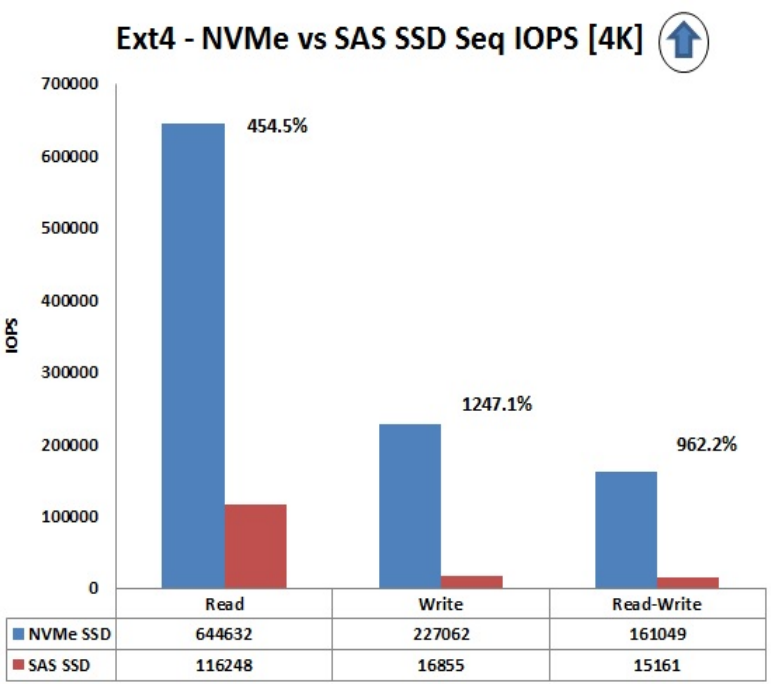
NVMe Command Execution Flow

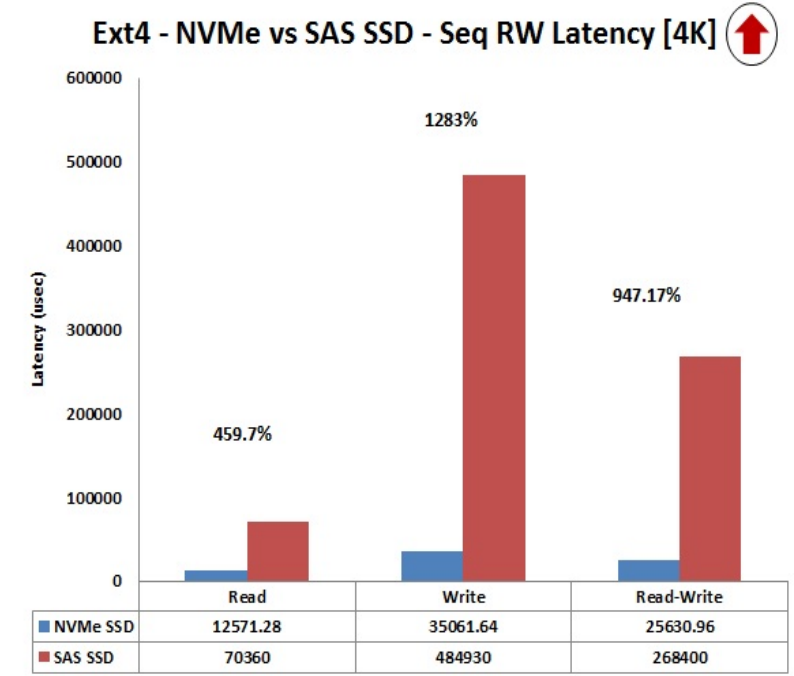


NVMe I/O Queue Pair

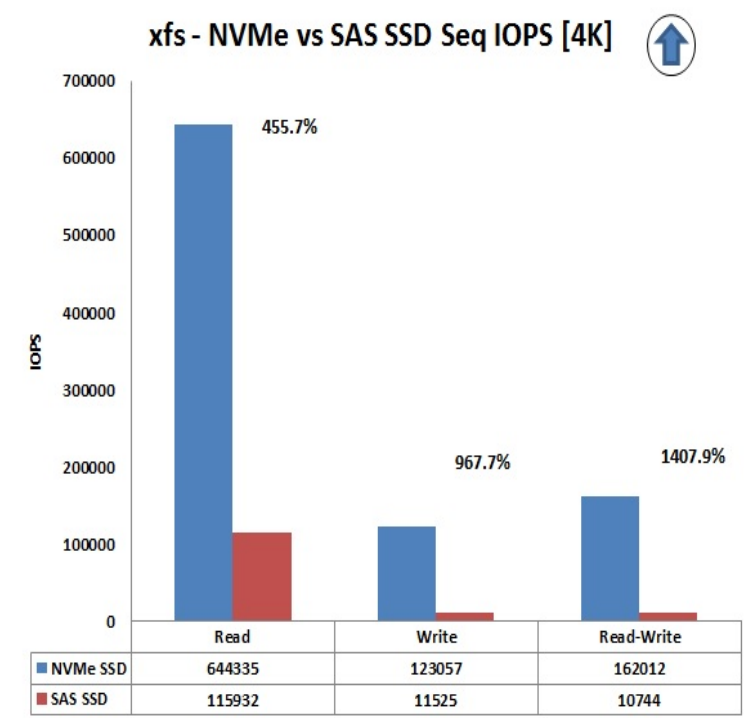


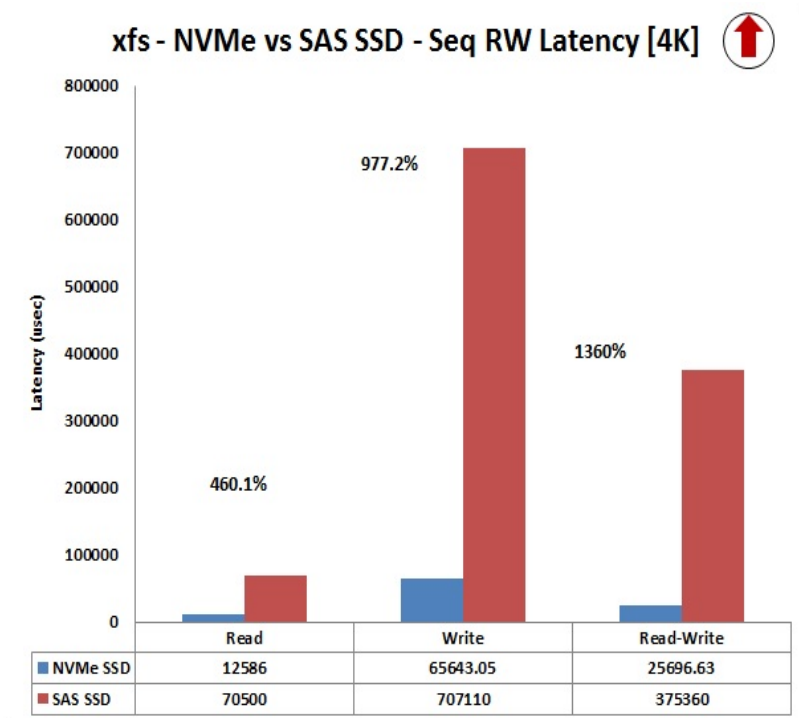
NVMe vs SAS – ext4



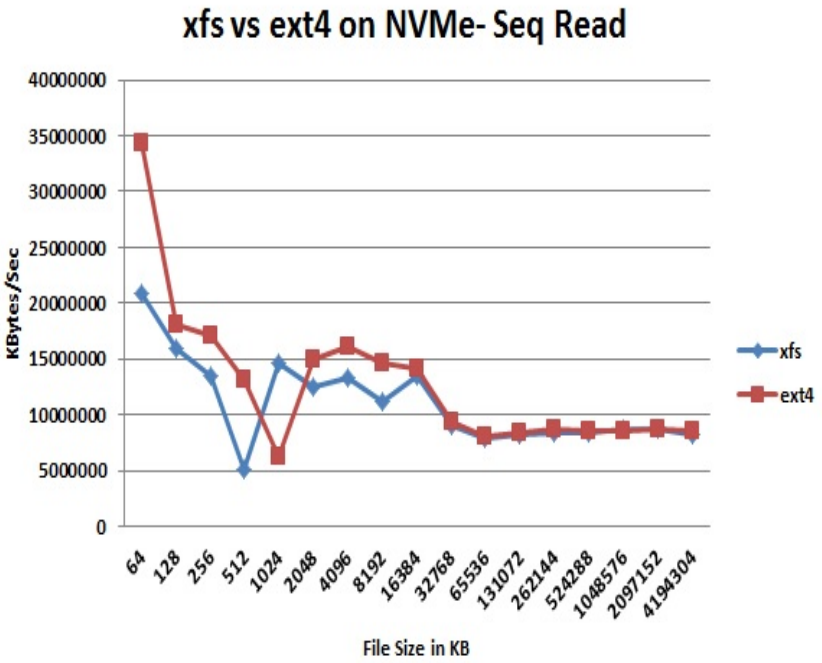


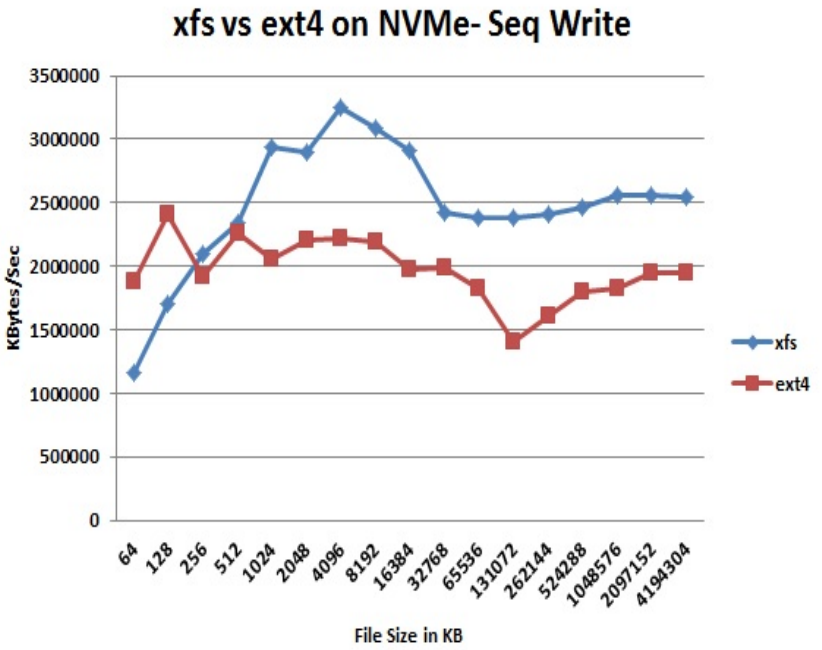
NVMe vs SAS – xfs



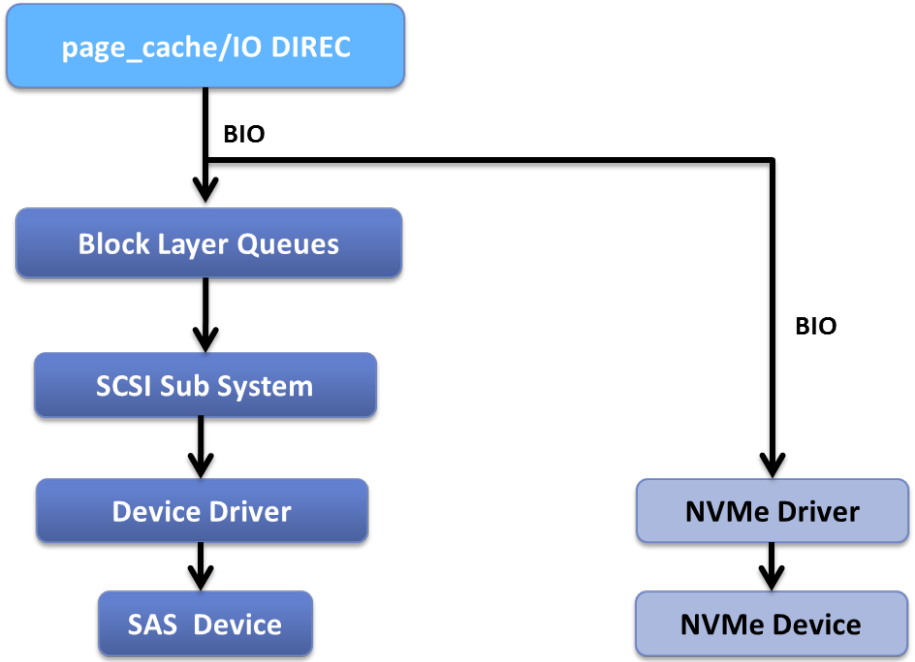


NVMe – xfs vs ext4





NVMe Driver in Linux Stack



The power to do more