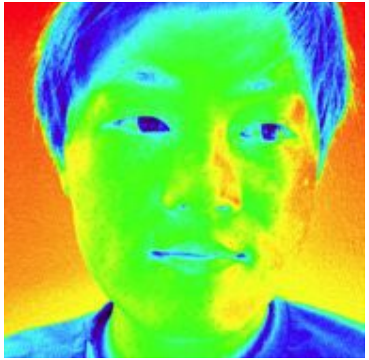
SDC 20
BY Developers FOR Developers

Storage Developer Conference
September 22-23, 2020

Improve Distributed Storage System Total Cost of Ownership with Host-Managed SMR HDDs

Albert Chen
KALISTA IO

Introduction



Albert Chen

CEO of KALISTA IO. Previously, senior engineering and management roles at WDC, MSFT and various startups. Pioneered industry's HM-SMR storage solutions.

hselin@kalista.io
https://linkedin.com/in/alberthchen

Preview: enabling HM-SMR everywhere

Apache Hadoop®

NGINX®

Ceph®

MongoDB®

Kubernetes® vols

Gitlab®

Docker® registry

Media servers

Minio®

and more...

Preview: without friction

No applications changes

No kernel modifications

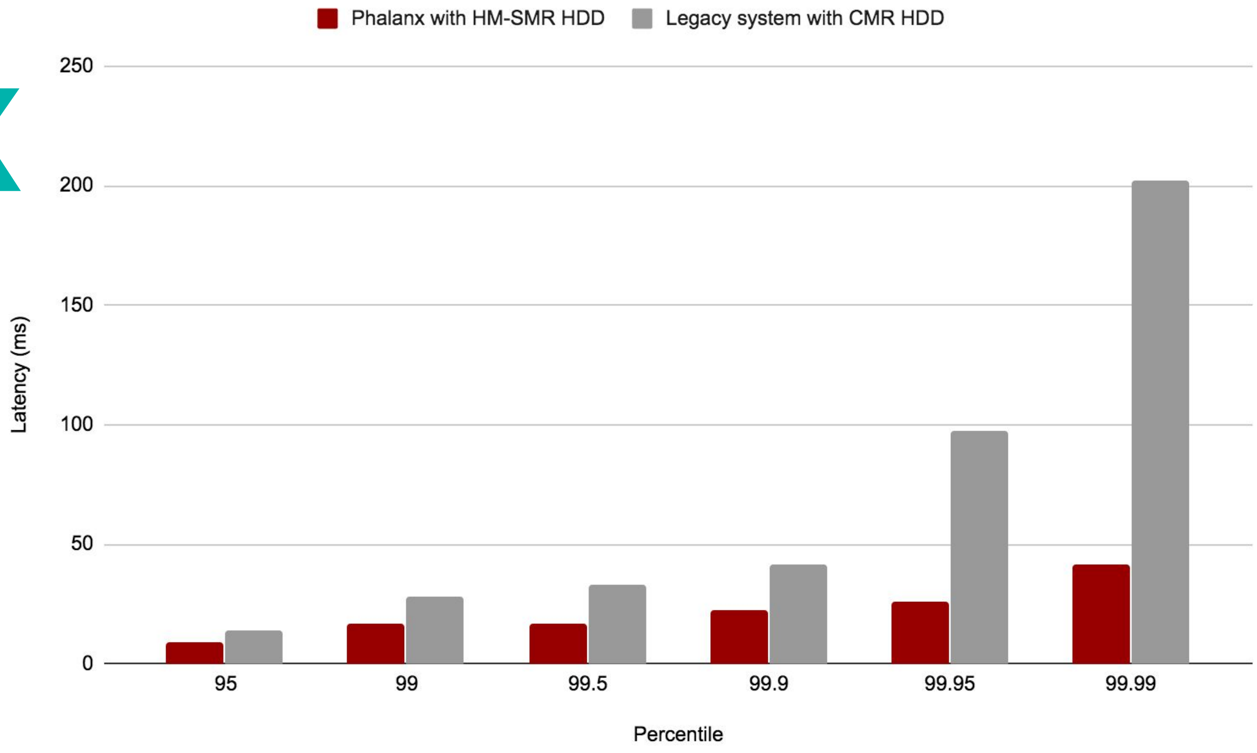
Just works

Preview: consistent performance at scale

4.8x

lower latency
at 99.99th percentile[3][4]

4KB write modifications
600,000 samples



Agenda

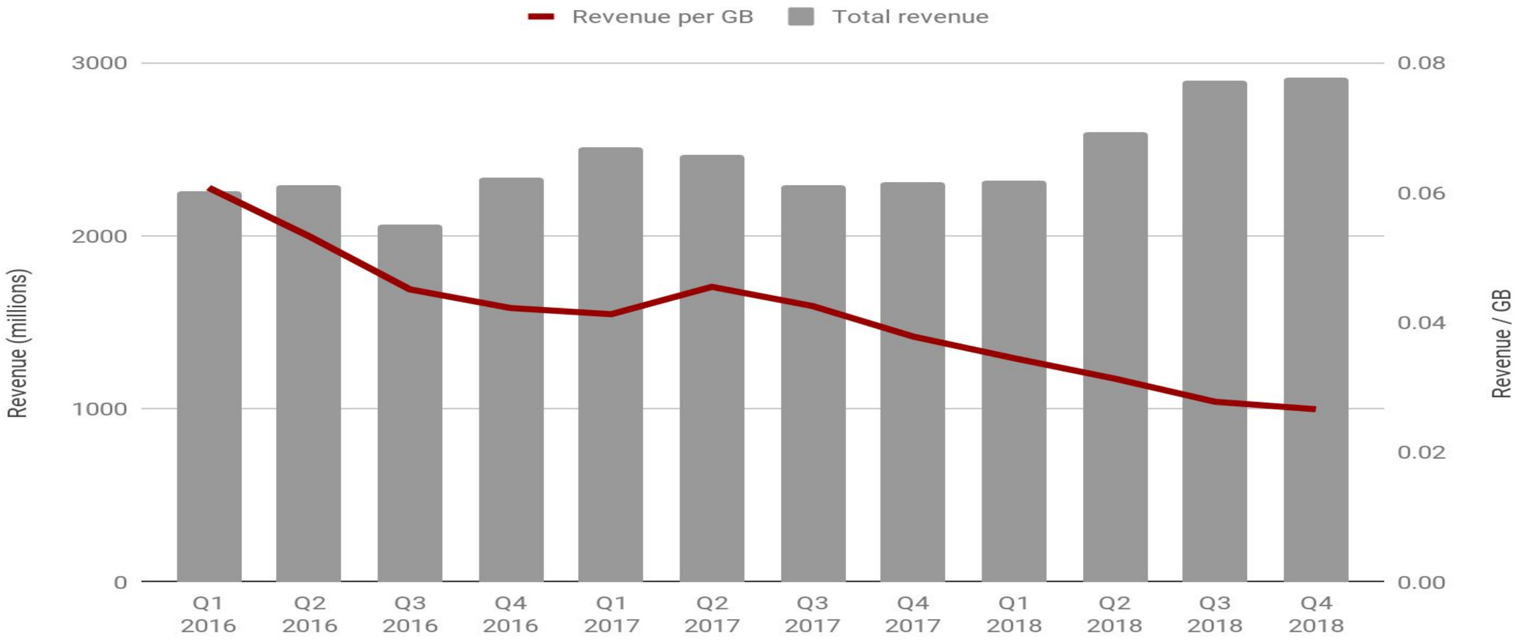
- Trends
  - Problems and opportunities
- Solutions
  - Host-Managed SMR
  - Current implementations and limitations
  - Improvements
- KALISTA Phalanx
  - Performance and simplicity

Trends

Explosive growth of digital data

Amount of data created globally will increase from 32 zettabytes (ZB) last year to over 100 ZB by 2023[1]

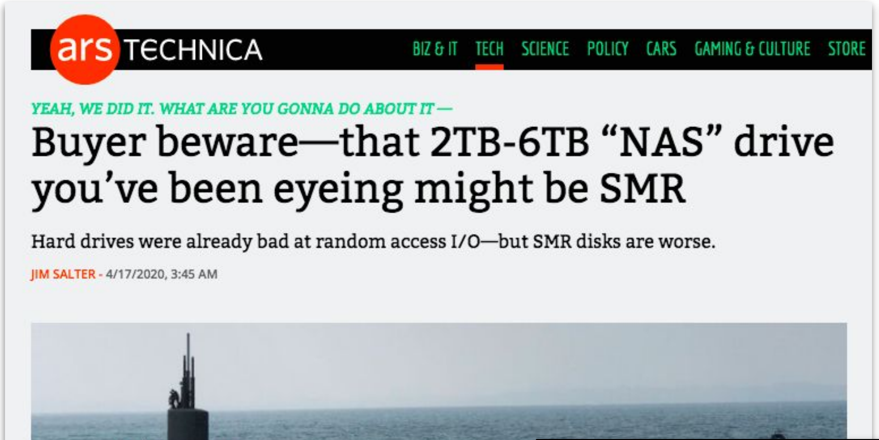
Falling cost ($/GB)[2]



Pushing the limits of device physics

Storage devices are becoming more complex, difficult and costly to use

New and expected usage models



YEAH, WE DID IT. WHAT ARE YOU GONNA DO ABOUT IT —

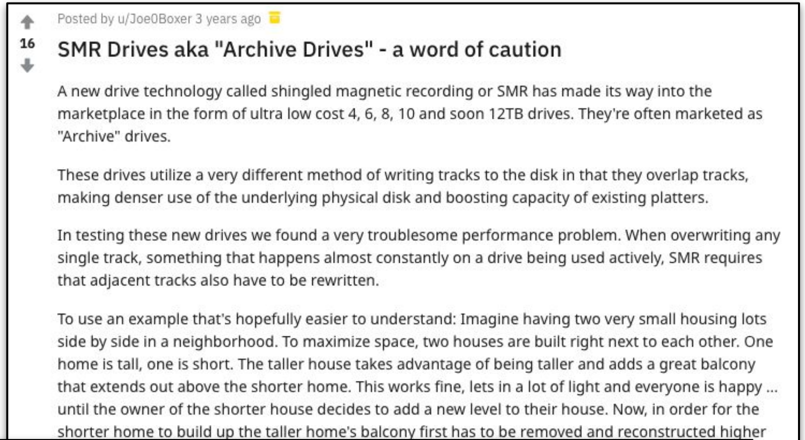
Buyer beware—that 2TB-6TB "NAS" drive you've been eyeing might be SMR

Hard drives were already bad at random access I/O—but SMR disks are worse.

JIM SALTER - 4/17/2020, 3:45 AM

Michael Mellen / ETC Robert Gulini

Enlarge / Shingled Magnetic Recording drives—unlike this *Los Angeles*-class sub vessels.



Posted by u/JoeOBoxer 3 years ago

SMR Drives aka "Archive Drives" - a word of caution

A new drive technology called shingled magnetic recording or SMR has made its way into the marketplace in the form of ultra low cost 4, 6, 8, 10 and soon 12TB drives. They're often marketed as "Archive" drives.

These drives utilize a very different method of writing tracks to the disk in that they overlap tracks, making denser use of the underlying physical disk and boosting capacity of existing platters.

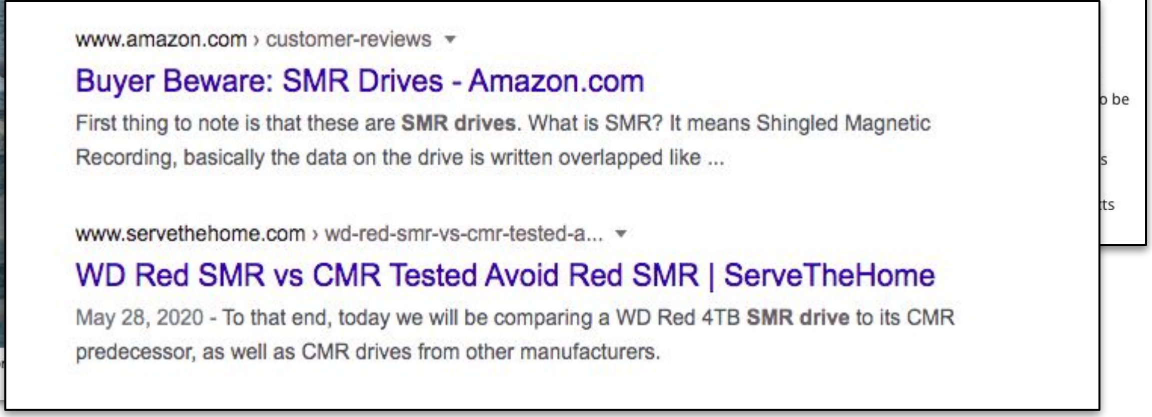
In testing these new drives we found a very troublesome performance problem. When overwriting any single track, something that happens almost constantly on a drive being used actively, SMR requires that adjacent tracks also have to be rewritten.

To use an example that's hopefully easier to understand: Imagine having two very small housing lots side by side in a neighborhood. To maximize space, two houses are built right next to each other. One home is tall, one is short. The taller house takes advantage of being taller and adds a great balcony that extends out above the shorter home. This works fine, lets in a lot of light and everyone is happy ... until the owner of the shorter house decides to add a new level to their house. Now, in order for the shorter home to build up the taller home's balcony first has to be removed and reconstructed higher

o be

s

ts



www.amazon.com › customer-reviews ▾

Buyer Beware: SMR Drives - Amazon.com

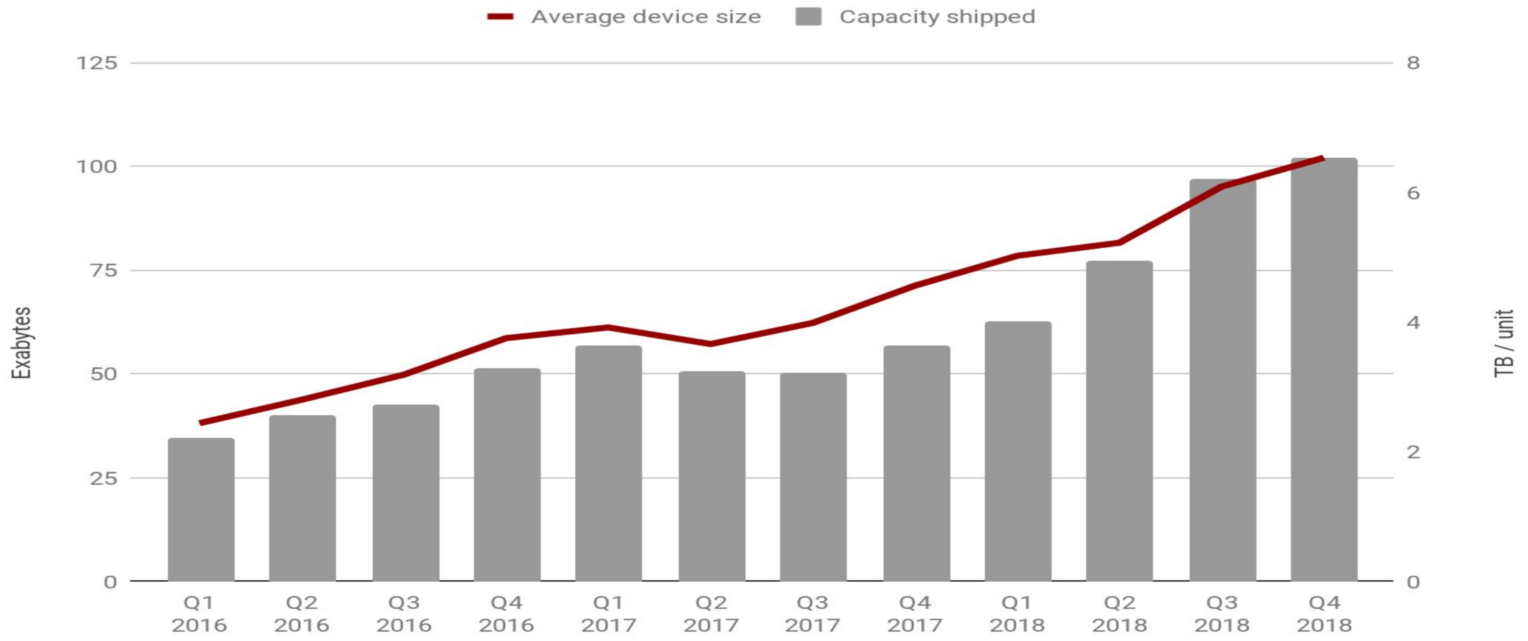
First thing to note is that these are **SMR drives**. What is SMR? It means Shingled Magnetic Recording, basically the data on the drive is written overlapped like ...

www.servethehome.com › wd-red-smr-vs-cmr-tested-a... ▾

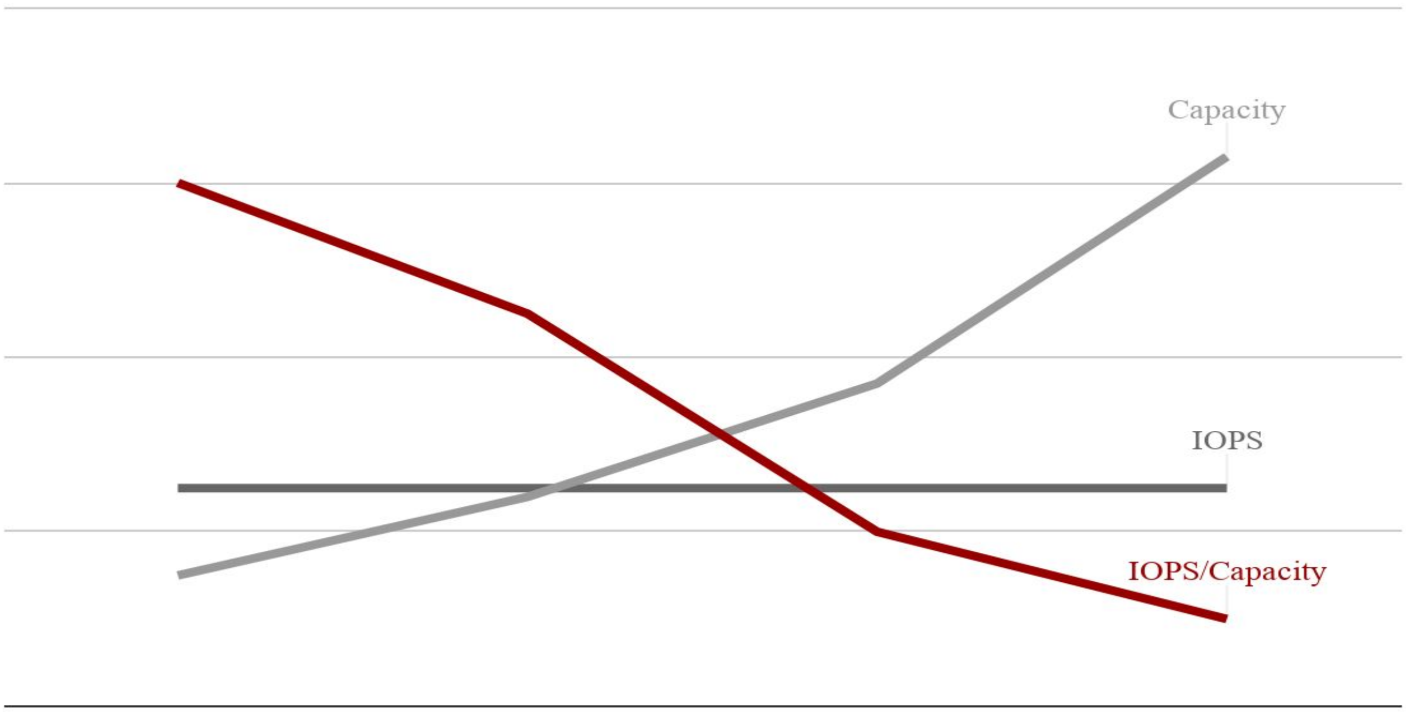
WD Red SMR vs CMR Tested Avoid Red SMR | ServeTheHome

May 28, 2020 - To that end, today we will be comparing a WD Red 4TB **SMR drive** to its CMR predecessor, as well as CMR drives from other manufacturers.

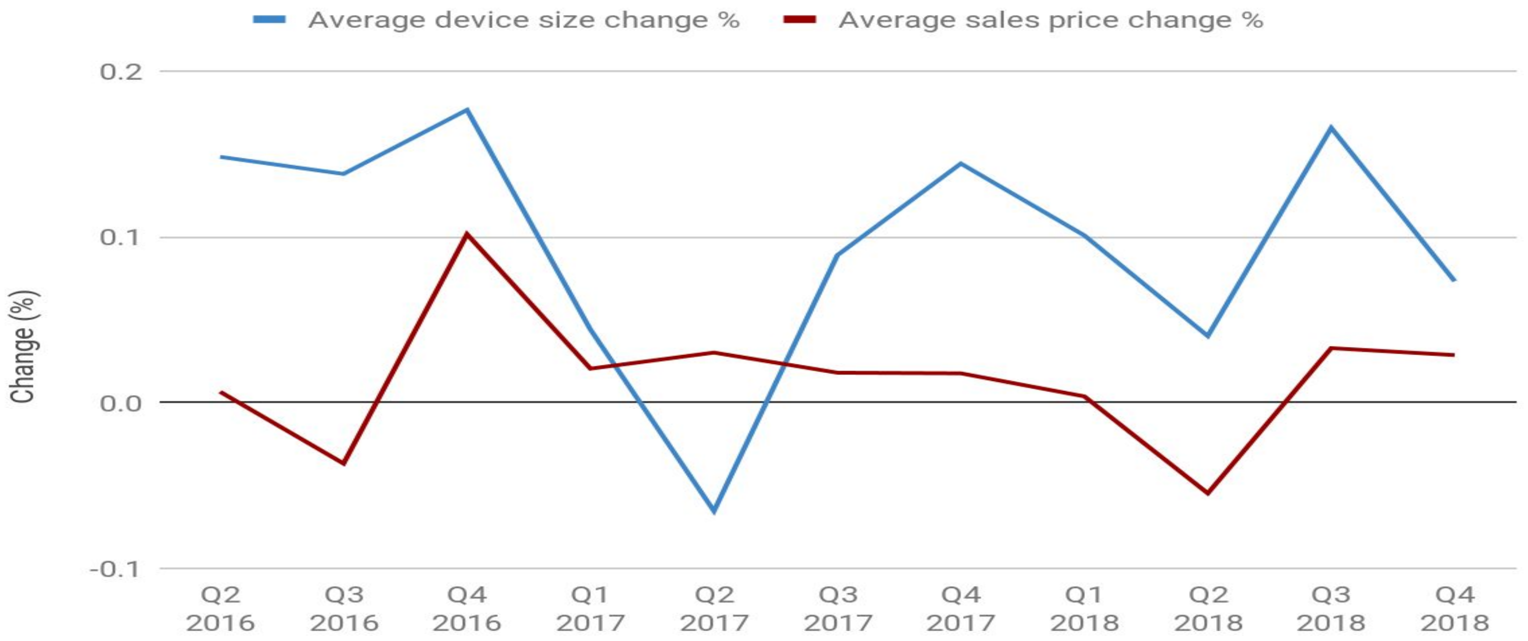
Increasing total capacity & device size[2]



Declining IO density



Limited margin for innovation[2]



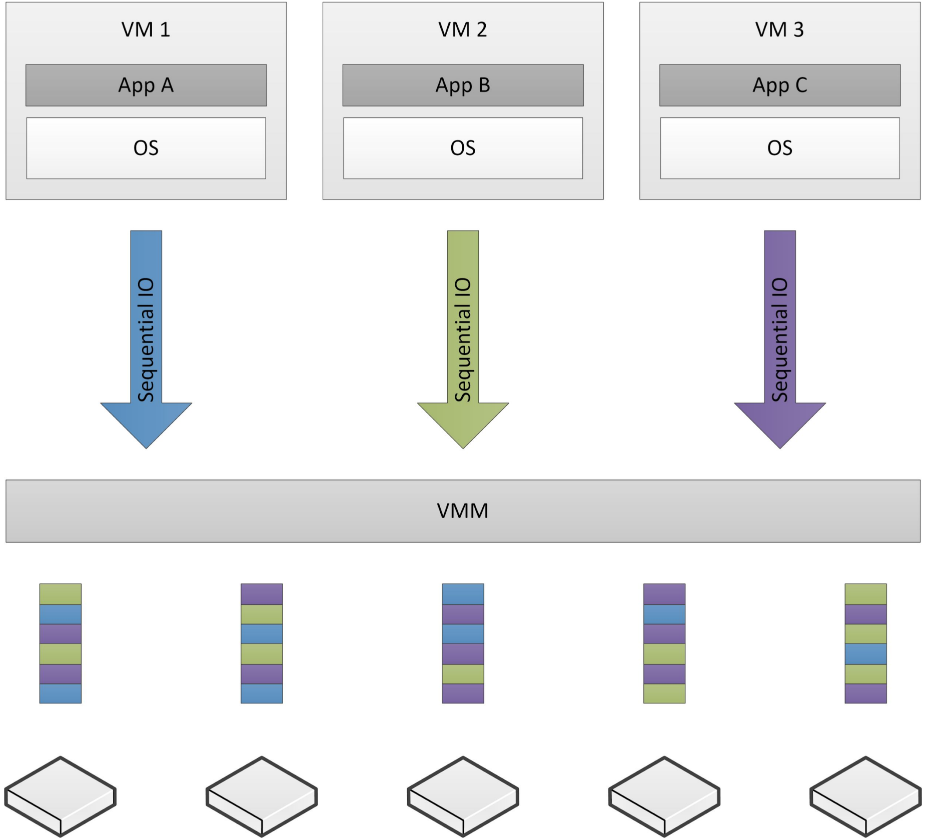
“Hard disk is the worst form of storage device, except for all the others.”

Winston Leonard Spencer-Churchill

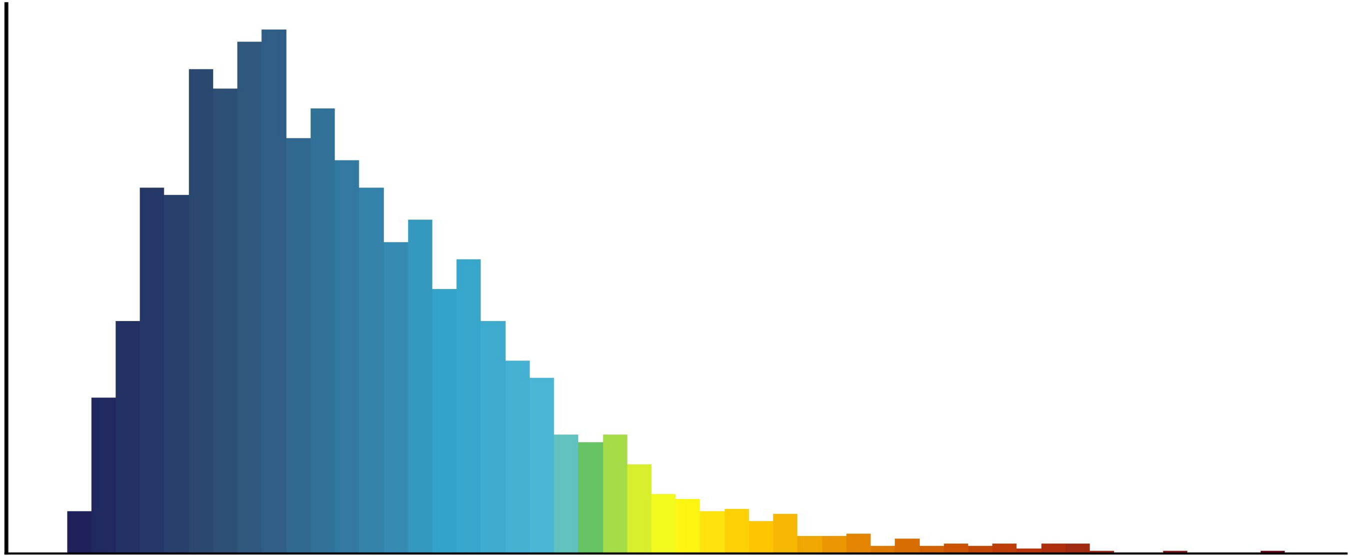
Demand for agility and optimal TCO

New architectures and usage models are growing increasingly incompatible & adverse for next generation storage technologies

IO Blender


VM 1
App A
OS
VM 2
App B
OS
VM 3
App C
OS
Sequential IO
Sequential IO
Sequential IO
VMM

Long tail latency



Total cost of ownership

Current Solutions

Host Managed SMR

Higher capacity

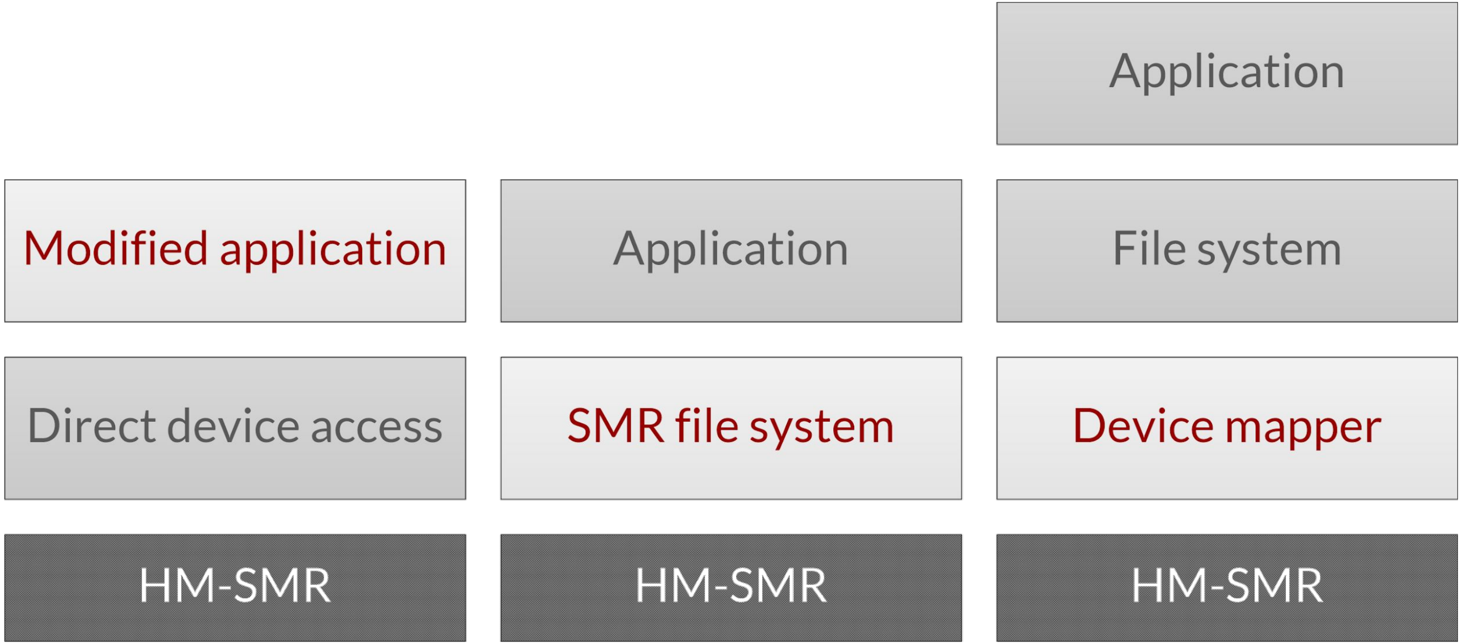
Reduced total cost of ownership

Consistent performance

More restrictive usage model

Investment in storage stack

Layers of indirection



Modified application

Direct device access

HM-SMR

Application

SMR file system

HM-SMR

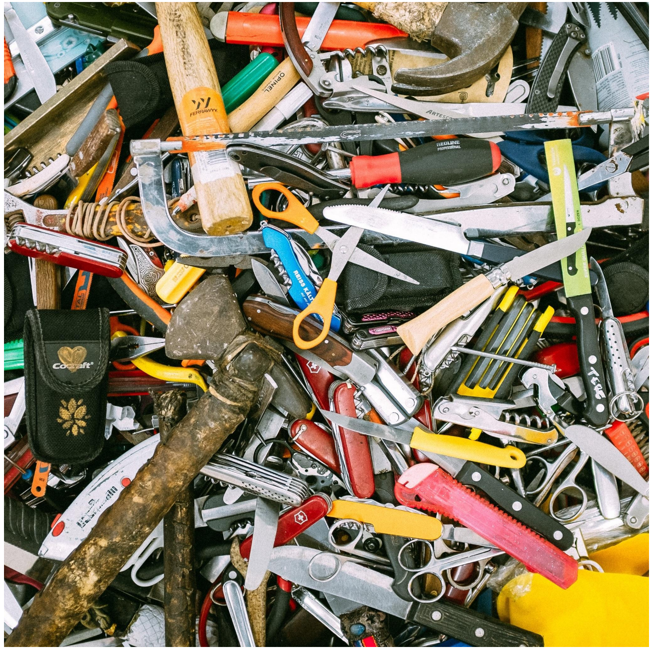
Application

File system

Device mapper

HM-SMR

Available implementations

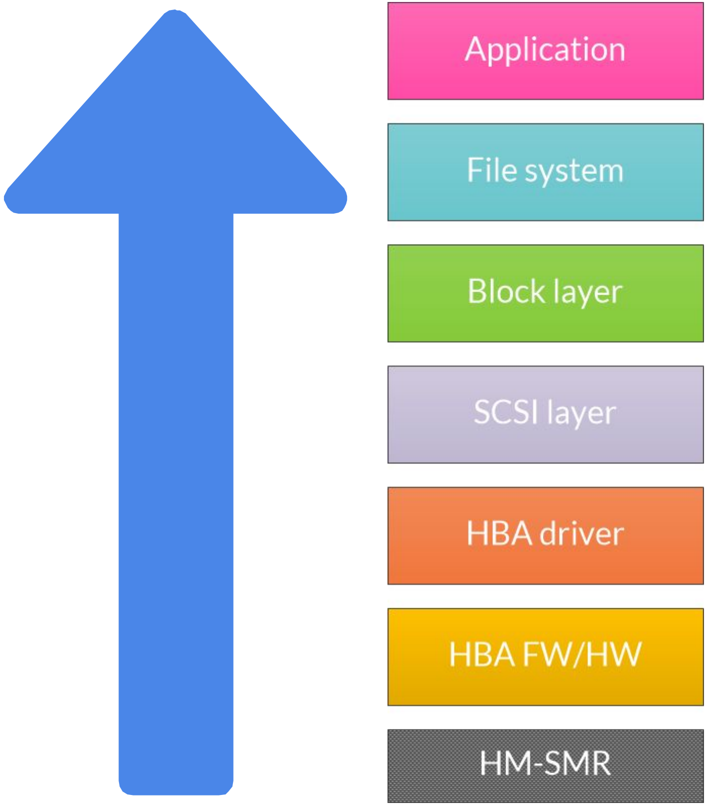


SG_IO	Direct access
libzbc	Direct access library
f2fs	SMR capable file system
dm-zoned	Device mapper target

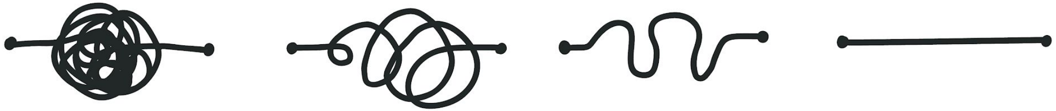
Can we do better?

"Wisdom begins in wonder." — Socrates

Make room for innovation



Improve user experience



Minimize dependency and limitations

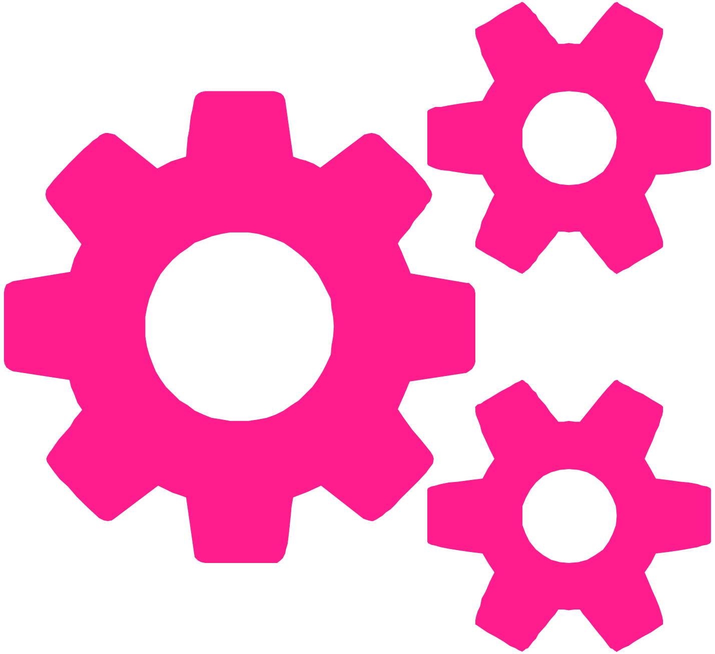
Kernel version

Modules/drivers

Hardware configuration

Protocol support

Leverage existing interfaces



File API

- open(), read(), write()…

Object API

- GET, PUT, DELETE

Block API

- TUR, WRITE, READ

Work for all devices

- Conventional device
  - HDD
  - SSD
- Zoned devices
  - HM/Hybrid-SMR HDD
  - ZNS SSD

Deploy anywhere at anytime

Minimal dependencies

Easy to add & remove capacity

Fits within existing workflows

Works with orchestration fwks

Be device friendly

Minimize seeks

Maximize IO transfer size

Prevent hot spots

Reduce background work

Perform at scale

Reduce contention

Increase IO concurrency

IO prioritization

Trim tail latency

Support new technologies

Multi-actuator

Variable capacity

Large block size

New usage models

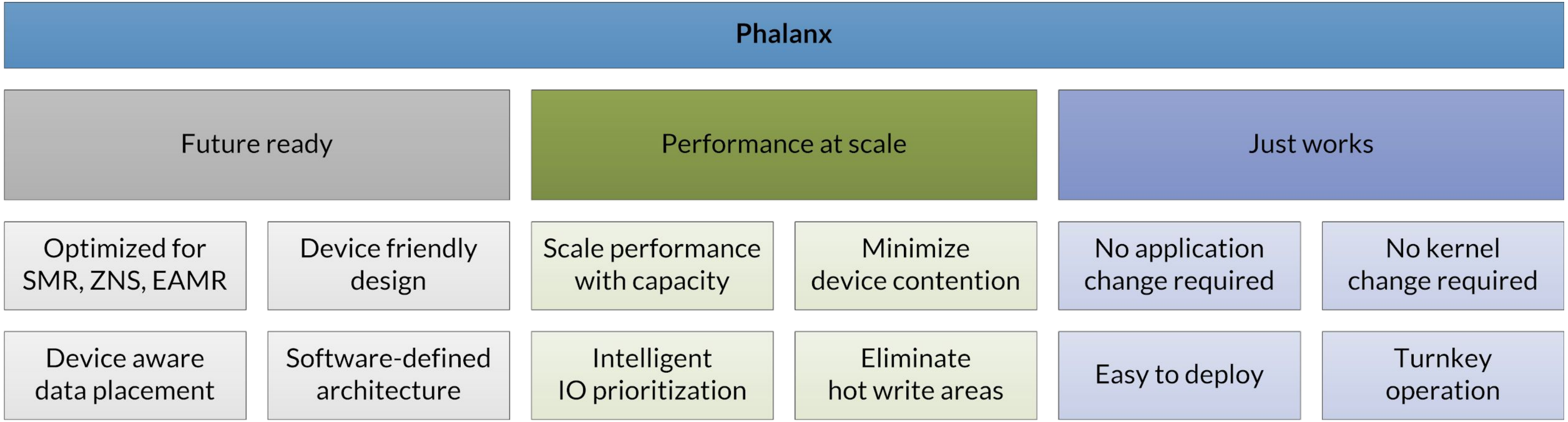
KALISTA IO

Get ready for a storage revolution

PHALANX STORAGE SYSTEM

Adding performance and simplicity

Performance, simplicity and future ready



Phalanx

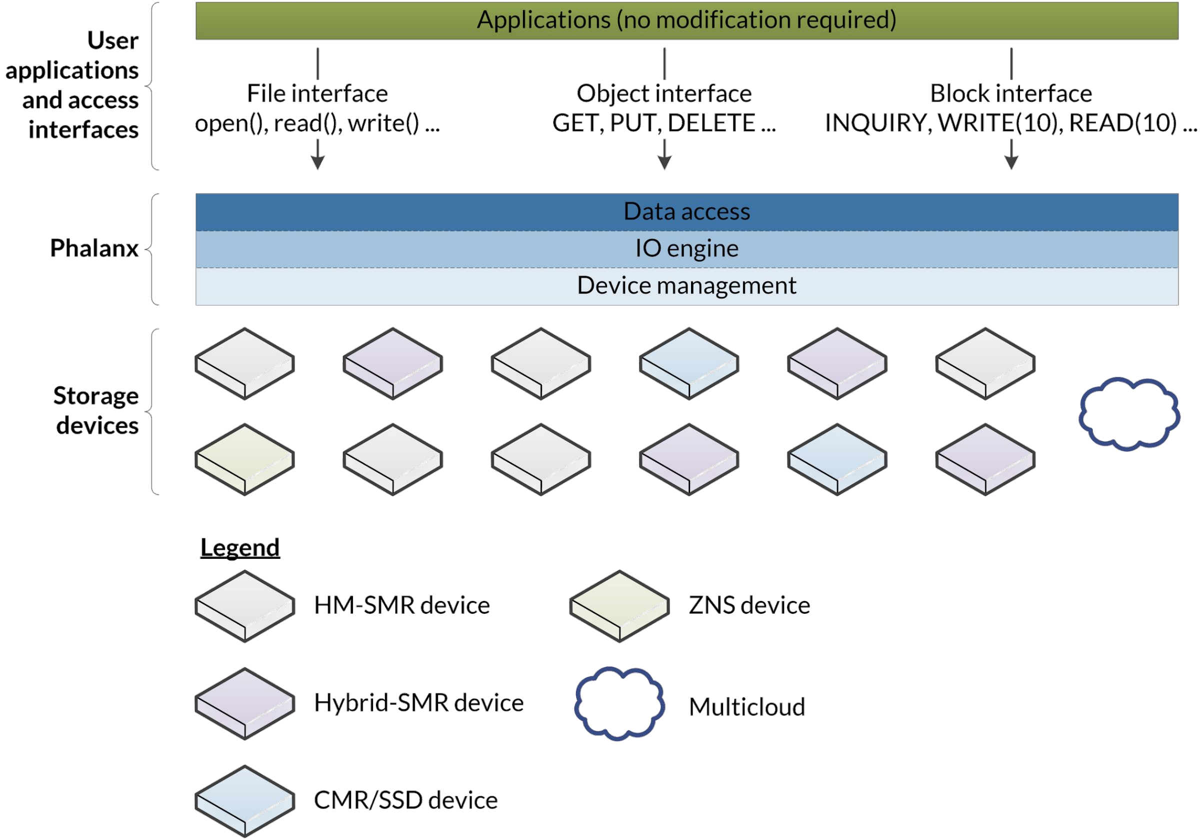
Future ready
- Optimized for SMR, ZNS, EAMR
- Device friendly design
- Device aware data placement
- Software-defined architecture

Performance at scale
- Scale performance with capacity
- Minimize device contention
- Intelligent IO prioritization
- Eliminate hot write areas

Just works
- No application change required
- No kernel change required
- Easy to deploy
- Turnkey operation

Simplifying data access and device management

Support existing interfaces & device types



User applications and access interfaces

Applications (no modification required)

File interface
open(), read(), write() ...

Object interface
GET, PUT, DELETE ...

Block interface
INQUIRY, WRITE(10), READ(10) ...

Phalanx

Data access

IO engine

Device management

Storage devices

Legend

HM-SMR device

ZNS device

Hybrid-SMR device

Multicloud

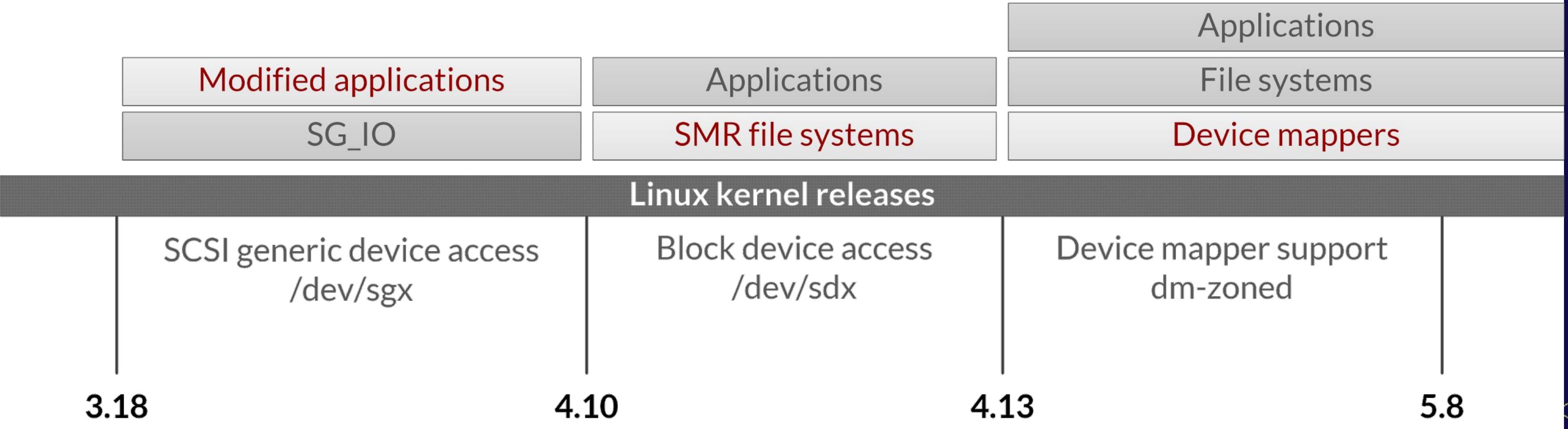
CMR/SSD device

Reducing dependencies and adapting to variations

Engineered to minimize dependency

- User space implementation
  - No kernel modifications
  - No additional modules/drivers
  - Generalized for all kernel versions
- Hardware
  - No zone configuration requirements
  - No device and zone size limitations

Know your dependencies



Modified applications

SG_IO

Applications

SMR file systems

Applications

File systems

Device mappers

Linux kernel releases

SCSI generic device access
/dev/sgx

Block device access
/dev/sdx

Device mapper support
dm-zoned

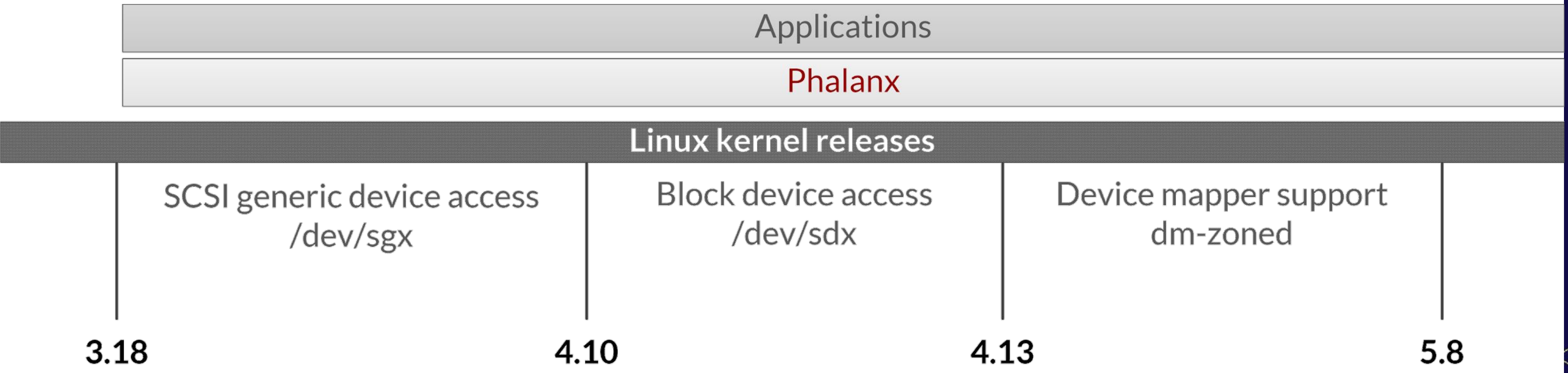
3.18

4.10

4.13

5.8

Declare your independence



Applications

Phalanx

Linux kernel releases

SCSI generic device access
/dev/sgx

Block device access
/dev/sdx

Device mapper support
dm-zoned

3.18

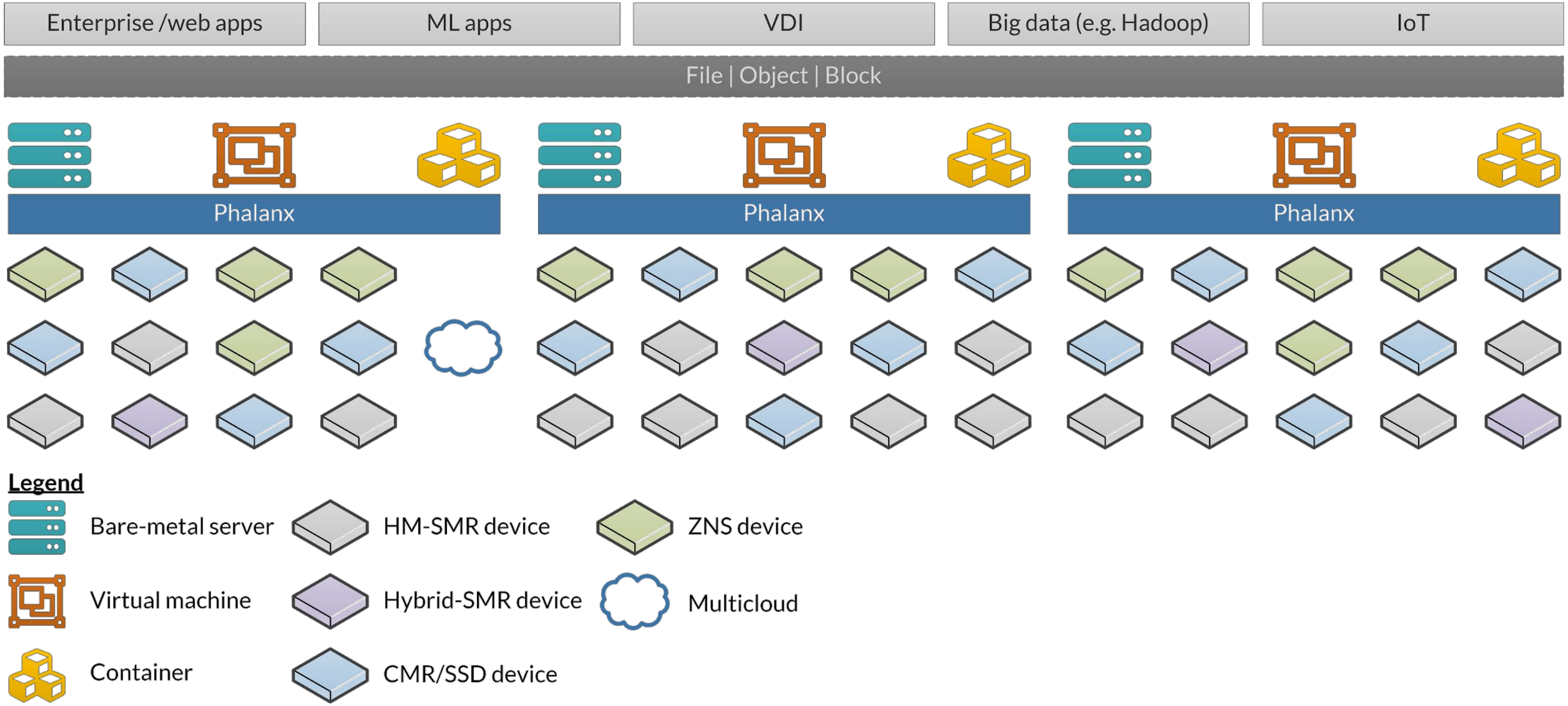
4.10

4.13

5.8

Designing for user experience

Deploy anywhere. Run everywhere.



2020 Storage Developer Conference. © KALISTA IO All Rights Reserved.

Easy to deploy. Simple to operate.

1. Download image

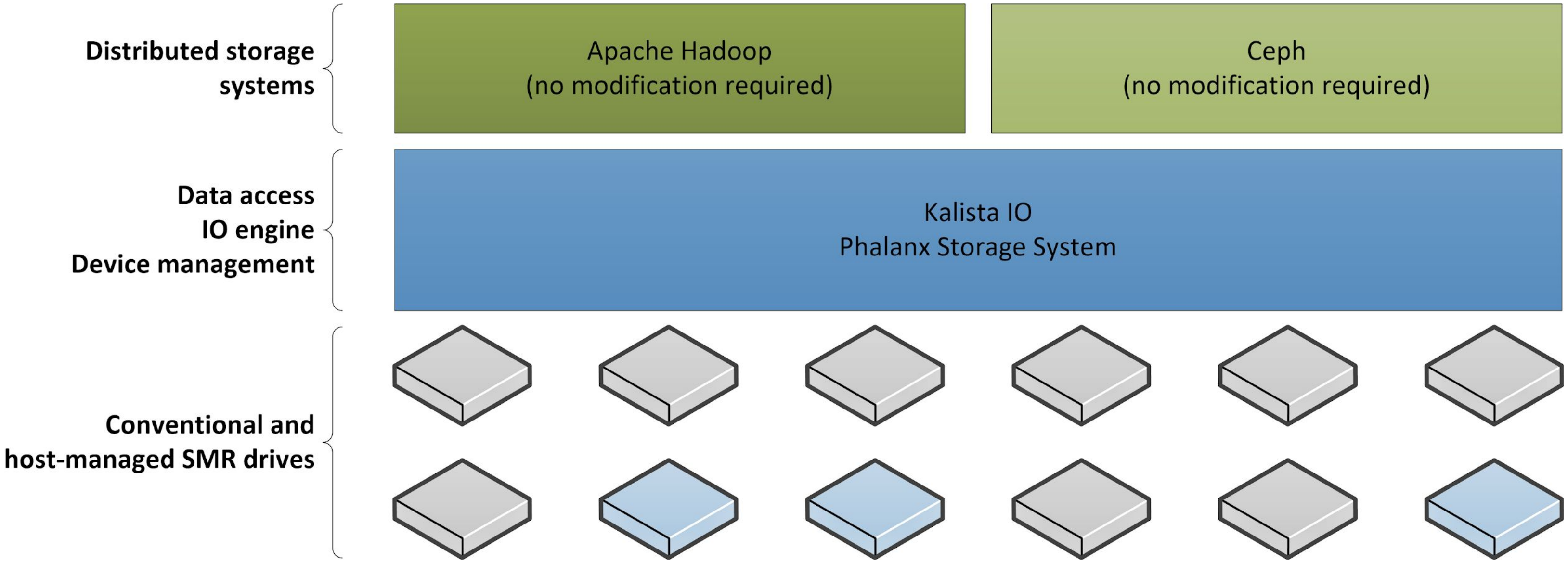
   ```
   docker pull kalistaio/phalanx:release
   ```

2. And start container

   ```
   docker run \
       . . .
       --mount type=bind,src=<mount path>. . . \
       kalistaio/phalanx:release
       . . .
       -d <path to HM-SMR devices> \
       . . .
   ```

What happens when you remove frictions and barriers to HM-SMR

Distributed systems with HM-SMR



Distributed storage systems
Apache Hadoop
(no modification required)
Ceph
(no modification required)
Data access
IO engine
Device management
Kalista IO
Phalanx Storage System
Conventional and host-managed SMR drives

And much more

NGINX®

GitLab®

MongoDB®

OpenStack Swift®

Docker® registry

Kubernetes® volumes

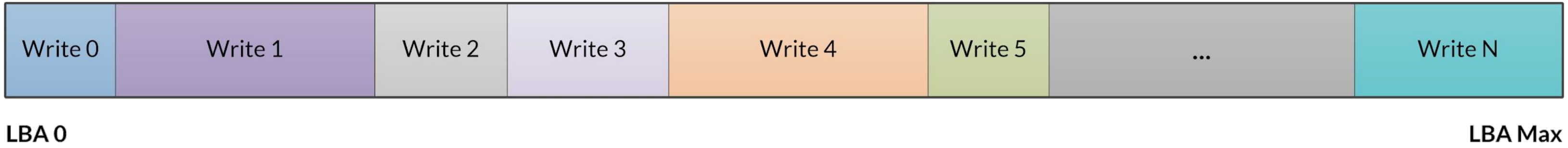
Minio®

Performing at scale

Designed for performance and scalability

- Minimize contention
  - Data/metadata separation
  - Log structured data layout
- Maximize IO concurrency
  - Support multi-actuator disks
  - Distribute workload across devices
- Generate device friendly behavior
  - Prevent hot spots
  - Minimize background work
  - Minimize seeks
- Scale performance with capacity
  - Row and column architecture

Minimize seeks and contention

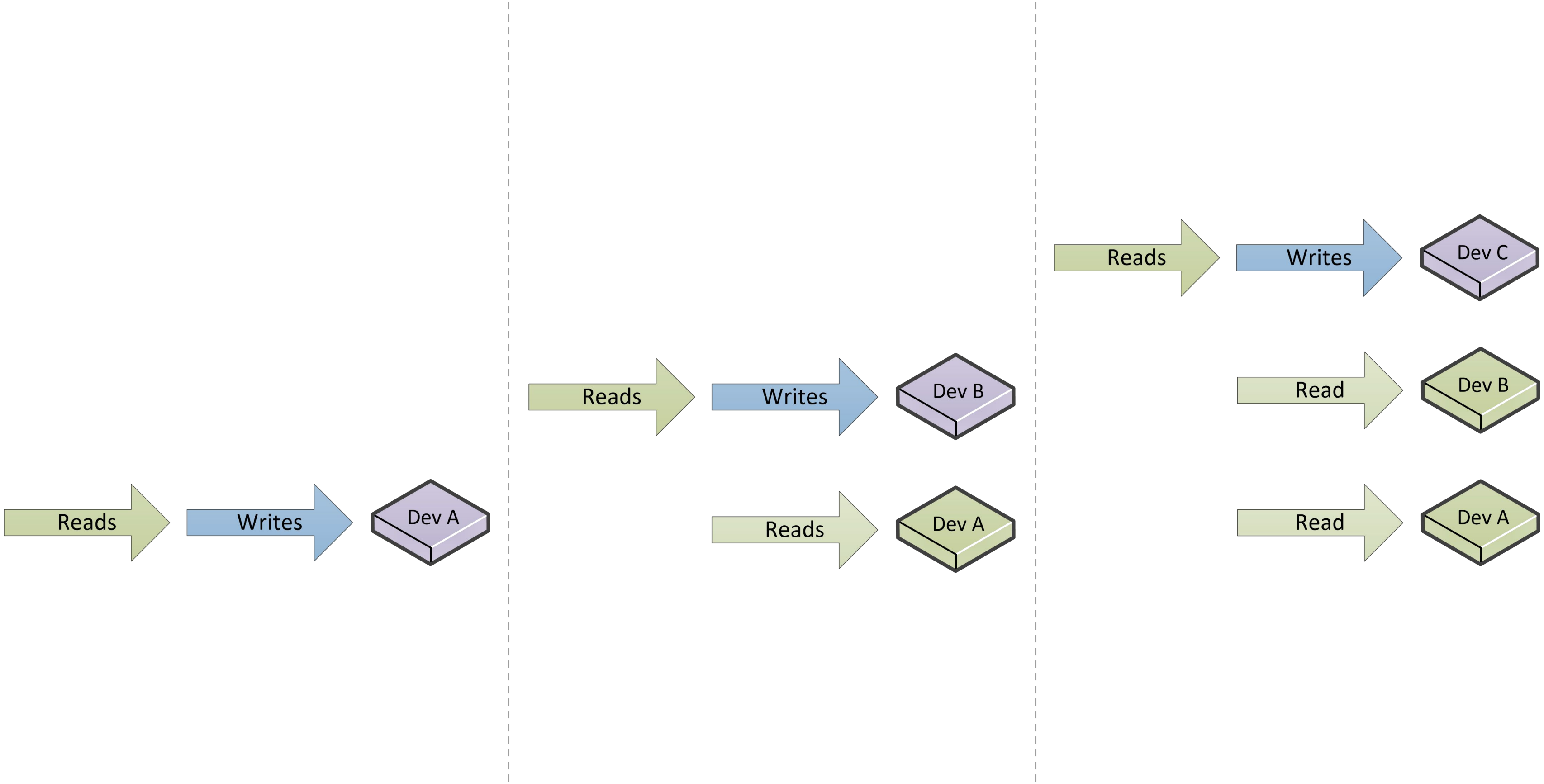


Write 0 | Write 1 | Write 2 | Write 3 | Write 4 | Write 5 | ... | Write N

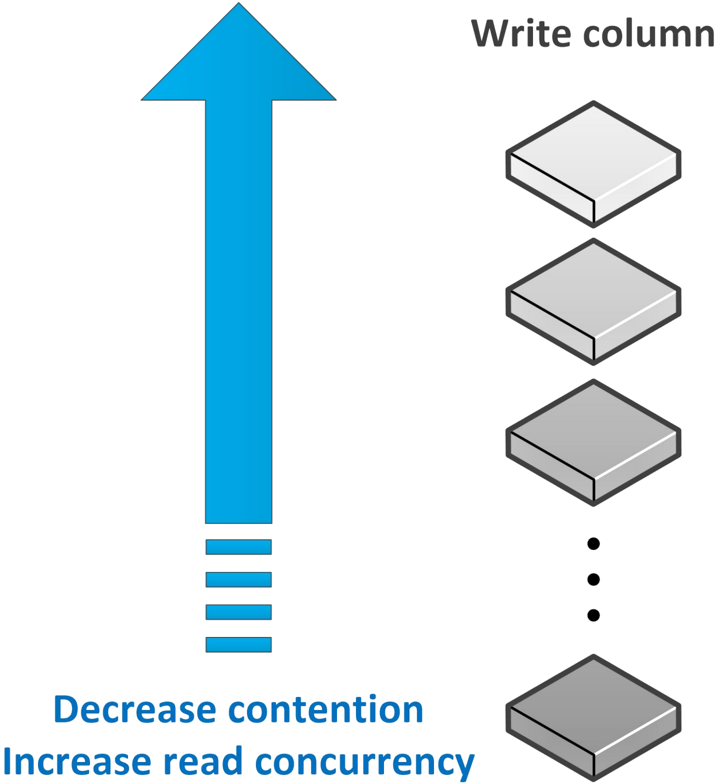
LBA 0

LBA Max

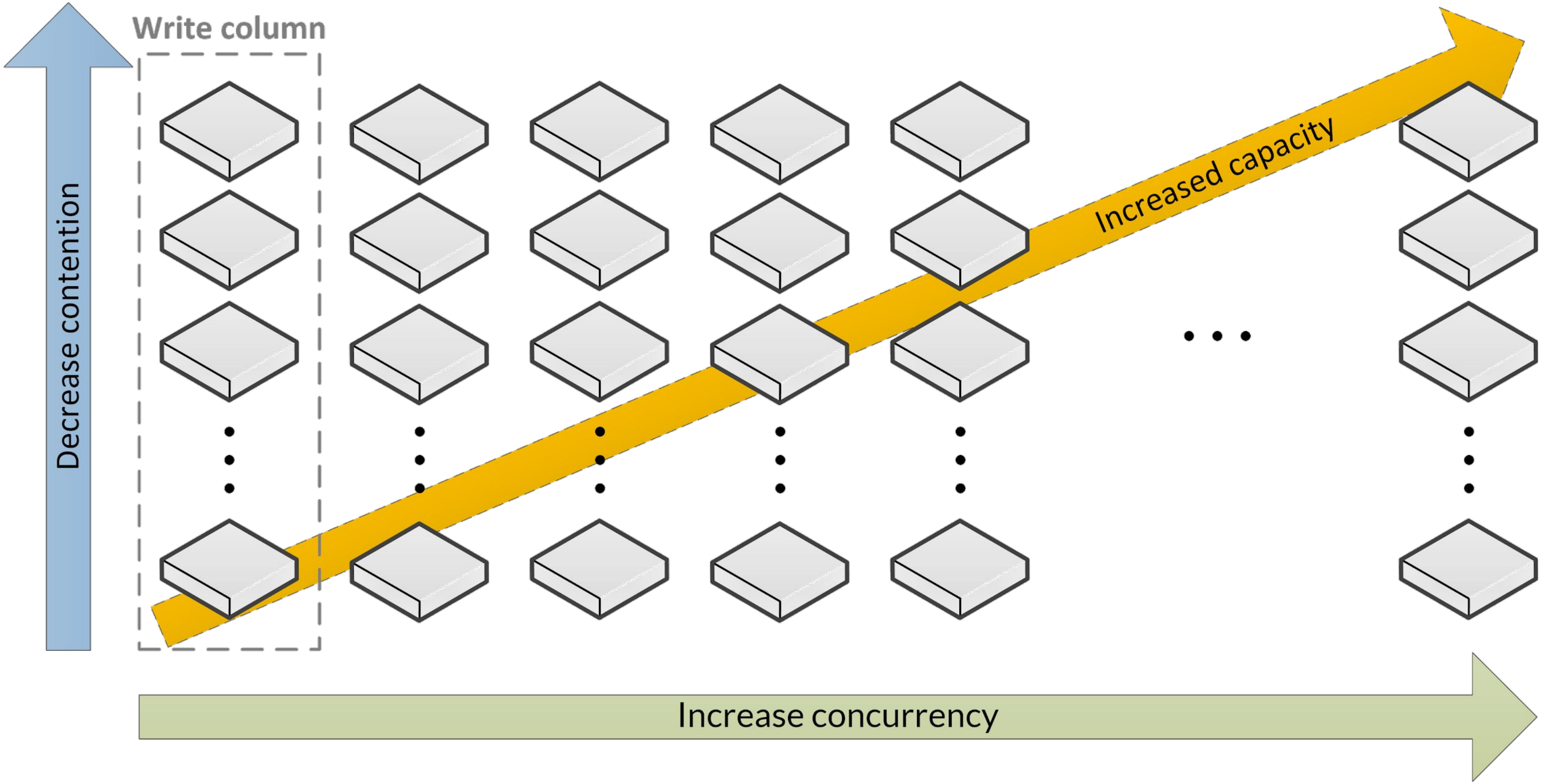
Distribute workload across devices



Decrease contention



Scale performance with capacity



Semantic intelligence

Prioritization

Tiering

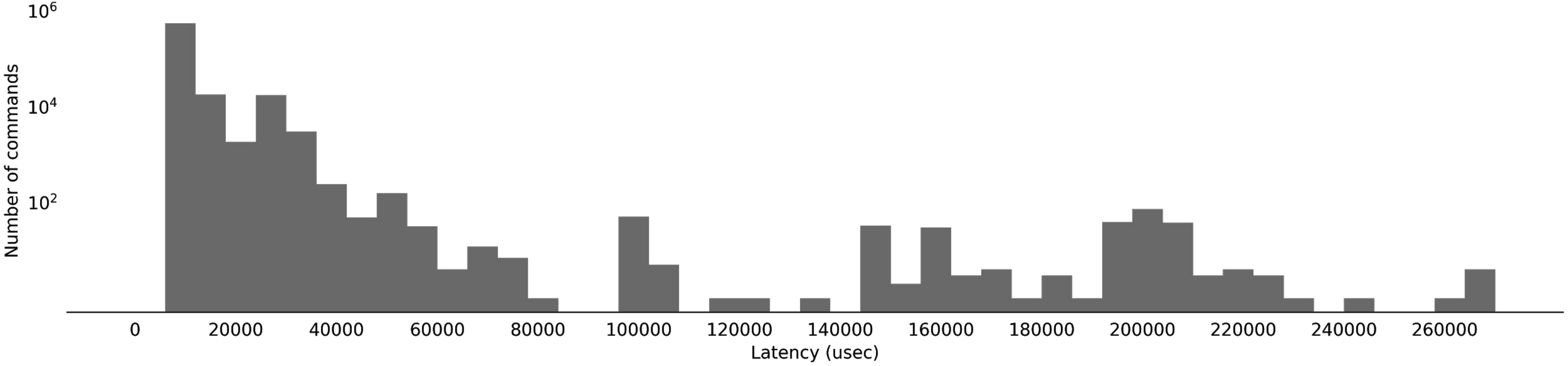
Caching

Predictive optimization

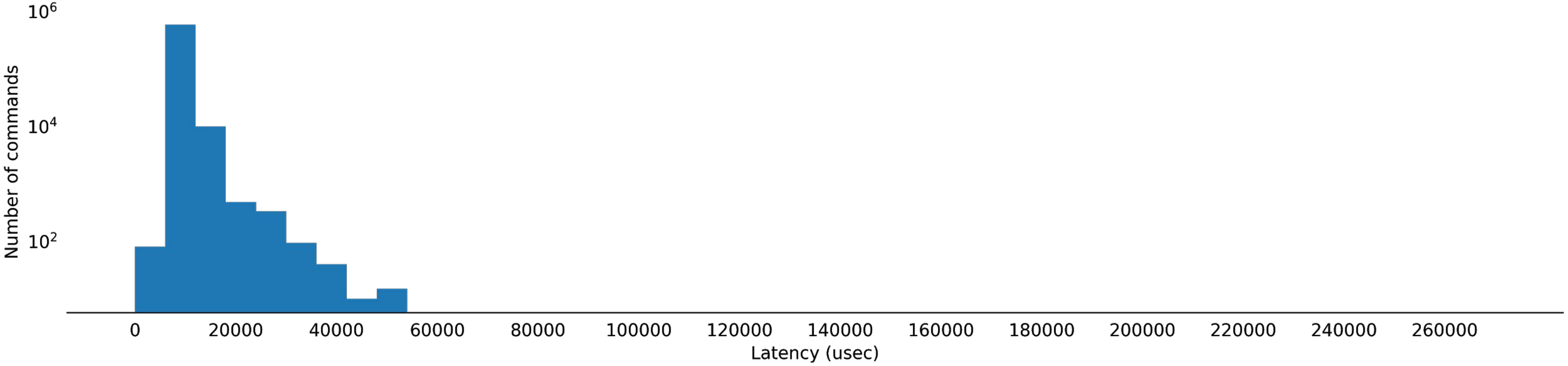
Quality of service (Qos)

What happens when you enable devices to perform at their best

Write tail latencies with legacy system[3]



Curtailed with Phalanx and HM-SMR[4]



Better percentile latencies (us)

	Phalanx with Ultrastar HC620	Legacy stack with Ultrastar HC530
99%	16,924	28,468
99.95%	26,211	97,371
99.99%	41,736	202,227

Benchmark systems configuration

Host-Managed SMR HDD Test System	CMR HDD Test System
Benchmark application (e.g. fio/Hadoop/Ceph)	Benchmark application (e.g. fio/Hadoop/Ceph)
Kalista IO Phalanx storage system	XFS/ext4
Linux 5.0.0-25-generic kernel	Linux 5.0.0-25-generic kernel
Western Digital Ultrastar DC HC620 Host-Managed SMR HDD	Western Digital Ultrastar DC HC530 CMR HDD

Benchmark results

16x

more IOPS
with fio random write[5]

19%

faster throughput
with Hadoop TestDFSIO read[6]

58%

higher IOPS
with Ceph Rados write bench[7]

10x

better performance consistency
with Ceph Rados write bench[7]

Thank you!

Contact

http://www.kalista.io
@kalista.io
hselin@kalista.io

"There is nothing impossible to him who will try." — Alexander

References

References

1. D. Reinsel and J. Rydning, "Worldwide Global DataSphere Forecast, 2019–2023: Consumer Dependence on the Enterprise Widening," IDC, 2019.

2. Source: Seagate Technology LLC and Western Digital Corp quarterly reports

3. Testing conducted by Kalista IO in July 2020 using XFS file system with Linux kernel 5.4.0-42-generic, and Intel® Core™ i7-4771 CPU 3.50GHz with 16GiB DDR3 Synchronous 2400 MHz RAM, and Western Digital Ultrastar DC HC530 CMR drive connected through SATA 3.2, 6.0 Gb/s interface. Write bench created a single 1GB file and executed 600,000 write commands each overwriting the first 64KB region of the file to capture latency values.

4. Testing conducted by Kalista IO in July 2020 using preproduction Olympus (Phalanx) software with Linux kernel 5.4.0-42-generic, and Intel® Core™ i7-4771 CPU 3.50GHz with 16GiB DDR3 Synchronous 2400 MHz RAM, and Western Digital Ultrastar DC HC620 host managed SMR drives connected through SATA 3.2, 6.0 Gb/s interface. Write bench created a single 1GB file and executed 600,000 write commands each overwriting the first 64KB region of the file capture latency values.

References

5. Testing conducted by Kalista IO in August 2019 using preproduction Phalanx software with Linux kernel 4.18.0-25-generic, and Intel Core i7-4771 CPU 3.50GHz with 16GiB DDR3 Synchronous 2400 MHz RAM, and Western Digital Ultrastar DC HC620 host managed SMR and Ultrastar DC HC530 CMR drives connected through SATA 3.2, 6.0 Gb/s interface. Tested with Flexible I/O tester (fio) version 3.14-11-g308a. Random write bench ran for 1800 seconds with 4KB block and 200GB file size, 64 concurrent threads each with queue depth of 1. Executed 3 times to capture average and standard deviation IOPS values.

6. Testing conducted by Kalista IO in August 2019 using preproduction Phalanx software with Linux kernel 5.0.0-25-generic, and Intel® Core™ i7-4771 CPU 3.50GHz with 16GiB DDR3 Synchronous 2400 MHz RAM, and Western Digital Ultrastar DC HC620 host managed SMR and Ultrastar DC HC530 CMR drives connected through SATA 3.2, 6.0 Gb/s interface. Tested with Apache Hadoop version 3.2.0 in single node pseudodistributed mode with single block replica, and TestDFSIO version 1.8 on OpenJDK version 1.8.0_222. TestDFSIO read benchmark ran with 32 files, 16GB each for a 512GB dataset. Executed 3 times to capture average and standard deviation throughput values.

References

7. Testing conducted by Kalista IO in August 2019 using preproduction Phalanx software with Linux kernel 5.0.0-25-generic, and Intel Core i7-4771 CPU 3.50GHz with 16GiB DDR3 Synchronous 2400 MHz RAM, and Western Digital Ultrastar DC HC620 host managed SMR and Ultrastar DC HC530 CMR drives connected through SATA 3.2, 6.0 Gb/s interface. Tested with Ceph version 13.2.6 Mimic in single node mode with single object replica. Rados write bench ran with 4MB object and block (op) size with 16 concurrent operations for 1800 seconds to capture average and standard deviation IOPS values.

Additional information

Additional information

1. **Western Digital Ultrastar DC HC600 SMR Series HDD**
   https://www.westerndigital.com/products/data-center-drives/ultrastar-dc-hc600-series-hdd

2. **KALISTA IO and Western Digital joint solution brief:**
   Distributed Storage System with Host-Managed SMR HDDs
   https://www.kalista.io/resources/joint-solution-briefs/KalistaIO-WDC-Joint-Solution-Brief.pdf

3. **Addressing Shingled Magnetic Recording drives with Linear Tape File System**
   https://www.snia.org/sites/default/files/files2/files2/SDC2013/presentations/Hardware/AlbertChenMalina_Addressing_Shingled_Magnetic_Recording.pdf

4. **Host Managed SMR**
   https://www.snia.org/sites/default/files/SDC15_presentations/smr/AlbertChen_JimMalina_Host_Managed_SMR_revision5.pdf

Additional information

5. **Linux SCSI Generic (sg) driver**
   http://sg.danny.cz/sg/index.html

6. **libzbc**
   https://github.com/hgst/libzbc

7. **dm-zoned**
   https://www.kernel.org/doc/html/latest/admin-guide/device-mapper/dm-zoned.html

8. **Flash-Friendly File System (F2FS)**
   https://www.kernel.org/doc/Documentation/filesystems/f2fs.txt

9. **Zoned storage**
   https://zonedstorage.io

10. **Linux kernel changes**
    https://kernelnewbies.org/LinuxVersions

Additional information

11. **Another Layer of Indirection**
https://www.linkedin.com/pulse/another-layer-indirection-albert-chen/

12. **The Digital Universe of Opportunities: Rich Data and the Increasing Value of the Internet of Things, IDC, April 2014**

13. **Phalanx Flexible I/O tester (fio) benchmarks**
https://www.kalista.io/resources/performance/phalanx-fio-benchmarks.pdf

14. **Phalanx Hadoop TestDFSIO benchmarks**
https://www.kalista.io/resources/performance/phalanx-hadoop-benchmarks.pdf

15. **Phalanx Ceph OSD and Rados benchmarks**
https://www.kalista.io/resources/performance/phalanx-ceph-benchmarks.pdf

Attributions

Attributions

1. Icons from Font Awesome. License available at https://fontawesome.com/license
   No modifications made.

Please take a moment to rate this session.

Your feedback matters to us.