CSAL: the Next-Gen Local Disk for the Cloud

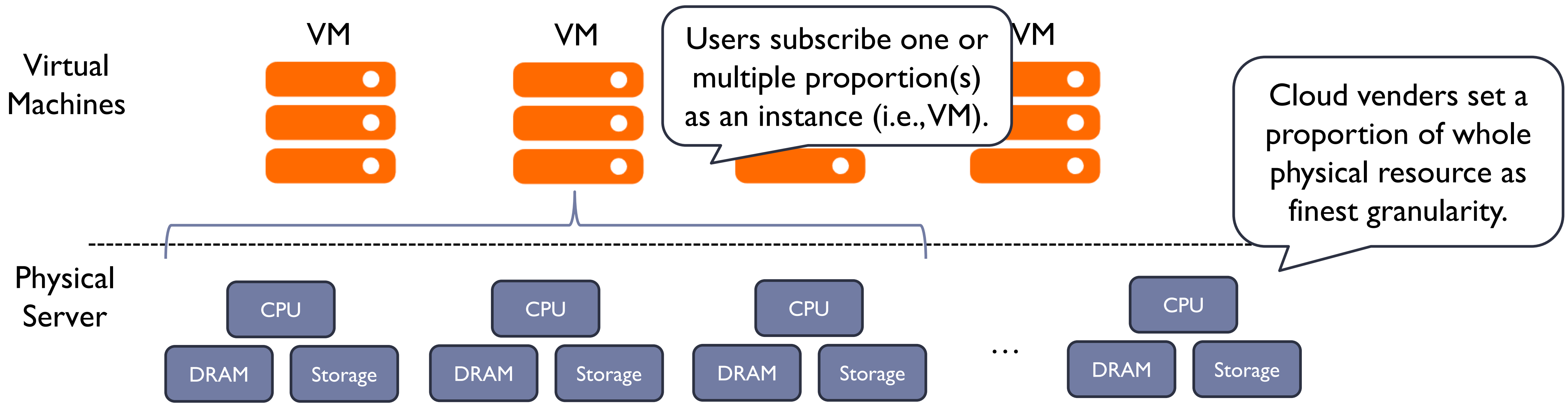
Yanbo Zhou, Erci Xu, Li Zhang, Kapil Karkra [†], Mariusz Barczak [†], Wayne Gao [†], Wojciech Malikowski [†], Mateusz Kozlowski [†], Łukasz Łasek [†], Ruiming Lu, Feng Yang, Lilong Huang, Xiaolu Zhang, Keqiang Niu, Jiaji Zhu and Jiesheng Wu

Alibaba Group
[†]Solidigm

Outline

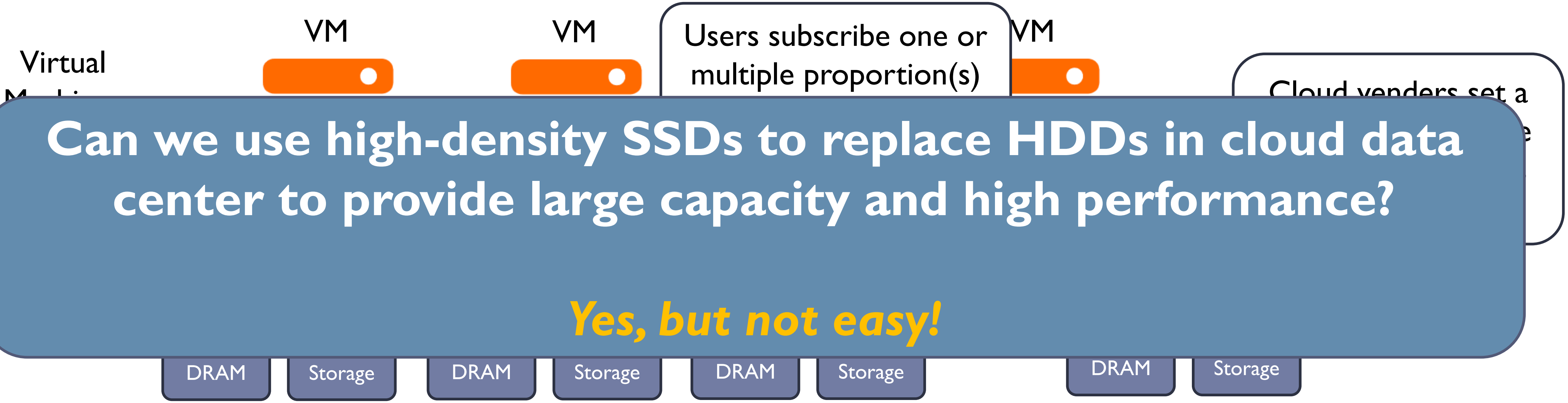
- Background
- Motivation
- Design
- Evaluation
- Conclusion

Cloud Local Disks and Characteristics



- CPU tend to have more cores and increase per-core efficiency
- Cloud venders scale up storage capacity and performance to meet CPU trends
  - HDD? Large capacity (e.g., 22TB HDD) but bad performance per TB.
  - SSD (MLC/TLC)? High performance but limited capacity and high costs.

Cloud Local Disks and Characteristics



Can we use high-density SSDs to replace HDDs in cloud data center to provide large capacity and high performance?

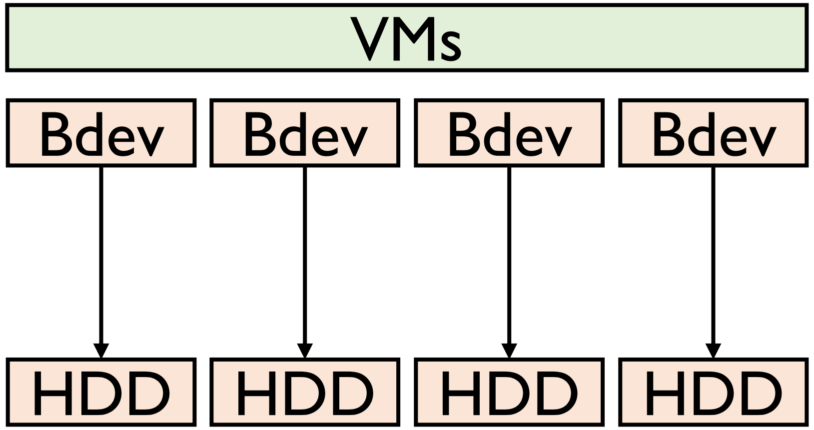
Yes, but not easy!

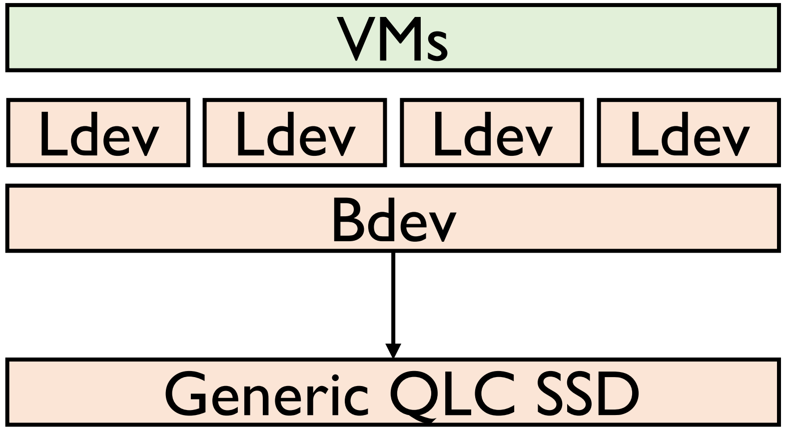
- CPU tend to have more cores and increase per-core efficiency
- Cloud venders scale up storage capacity and performance to meet CPU trends
  - HDD? Large capacity (e.g., 22TB HDD) but bad performance per TB.
  - SSD (MLC/TLC)? High performance but limited capacity and high costs.

Outline

- Background
- Motivation
- Design
- Evaluation
- Conclusion

Attempt 1: QLC as a Drop-in Replacement





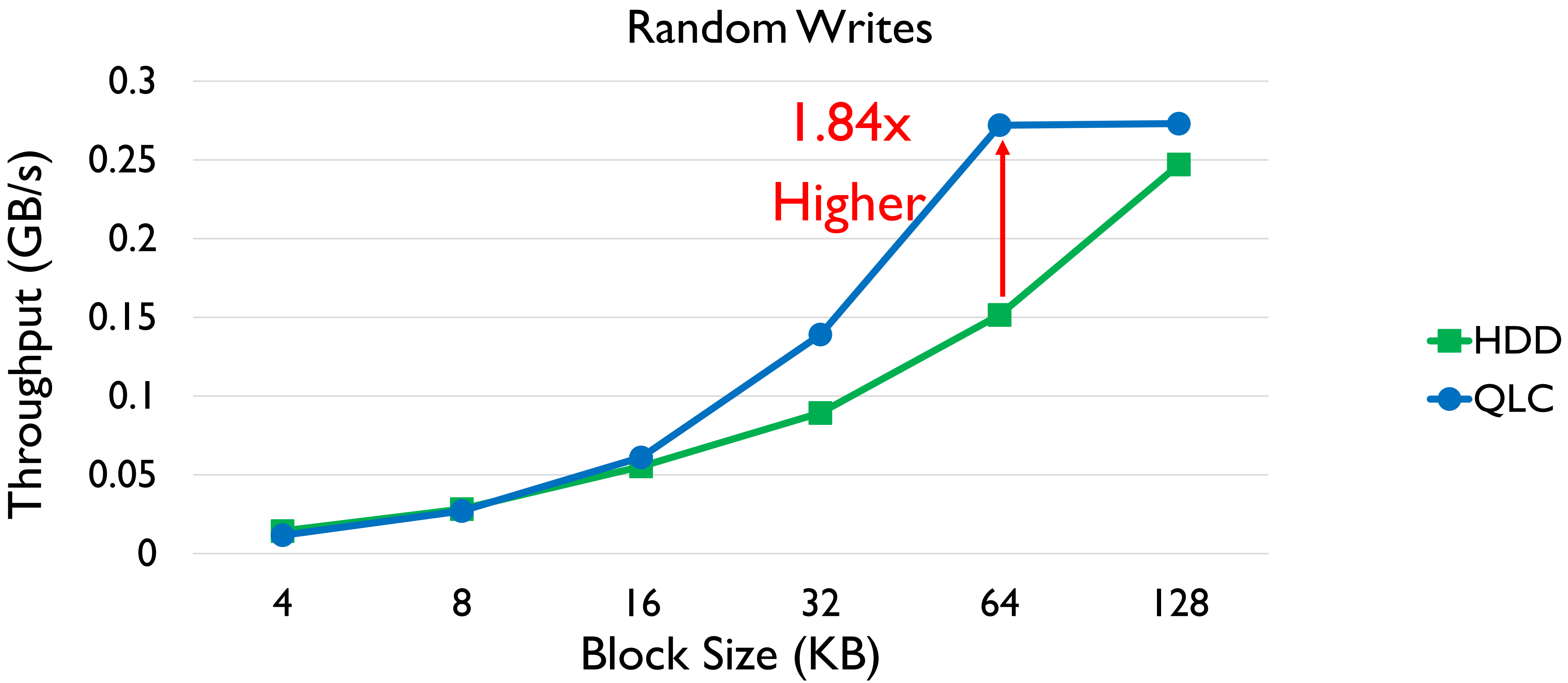
Legacy approach with HDD

- Total 16TB Storage
- 8x logical devices on 8x 2TB HDDs
- 8 VMs (each with one 2TB device)

QLC as a drop in replacement

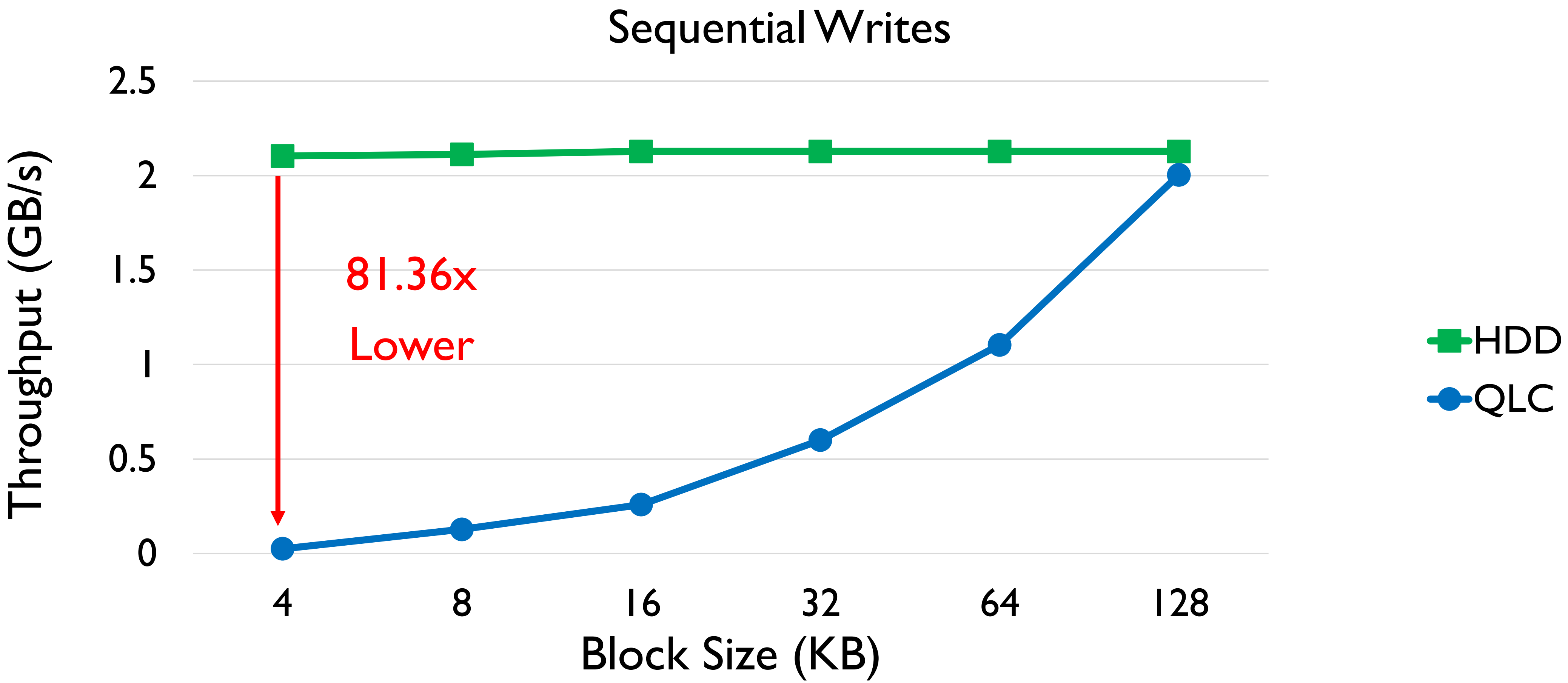
- Total 16TB Storage
- 8x logical devices on1x 16QLC SSD
- 8 VMs (each with one 2TB device)

Performance Analysis



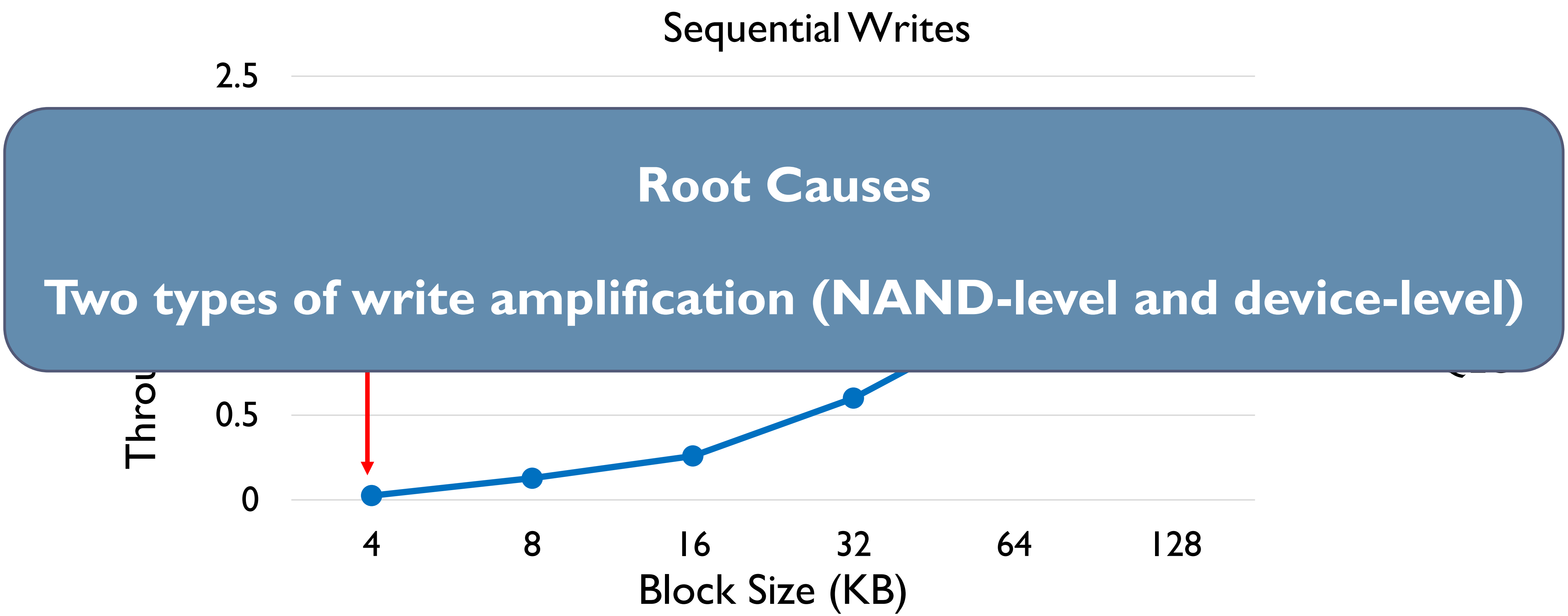
- Random write performance is better than HDDs

Performance Analysis



- Sequential writes performance is worse than HDDs (especially for small I/Os)

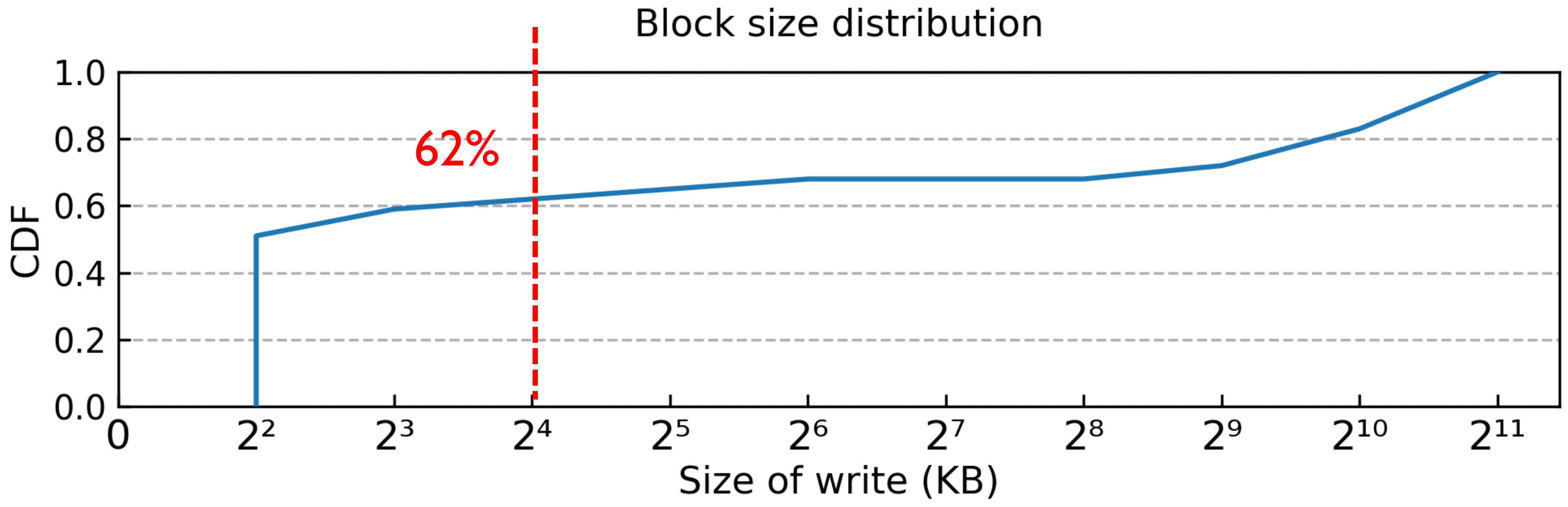
Performance Analysis



- Sequential writes performance is worse than HDDs (especially for small I/Os)

NAND-level Write Amplification

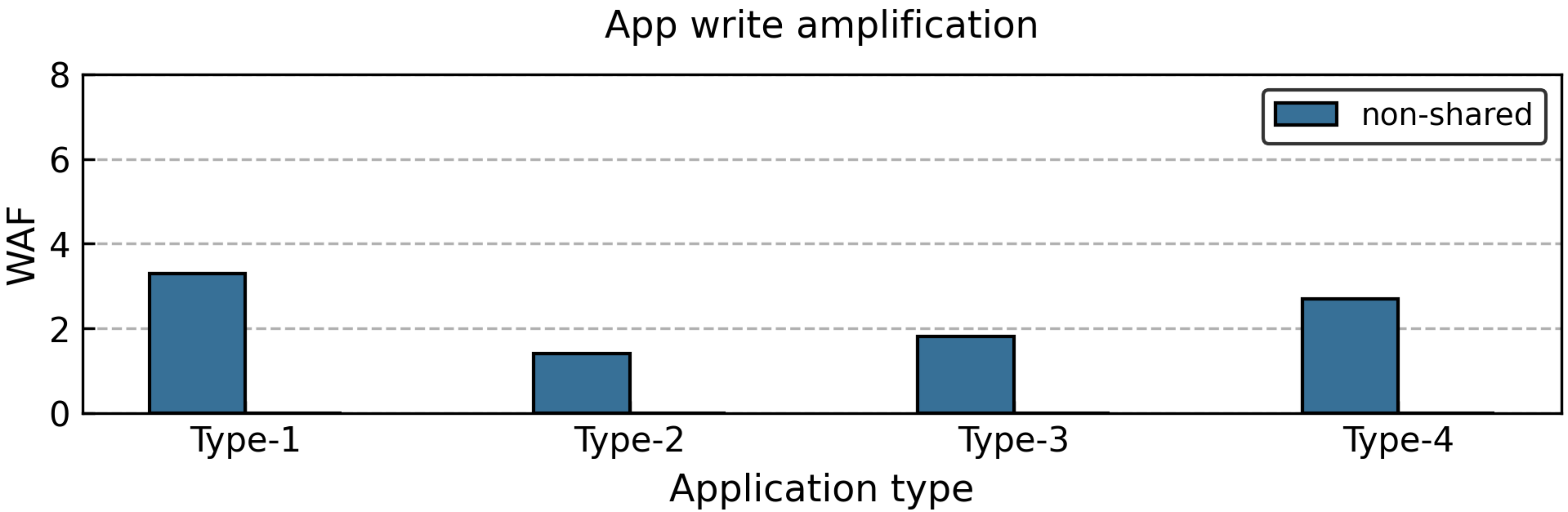
Large capacity SSDs tend to use larger super block so that frequent small writes lead to a significant increase in NAND-level write amplification (WA).



- Small block (4KB-16KB) writes account for more than 60% in real workloads

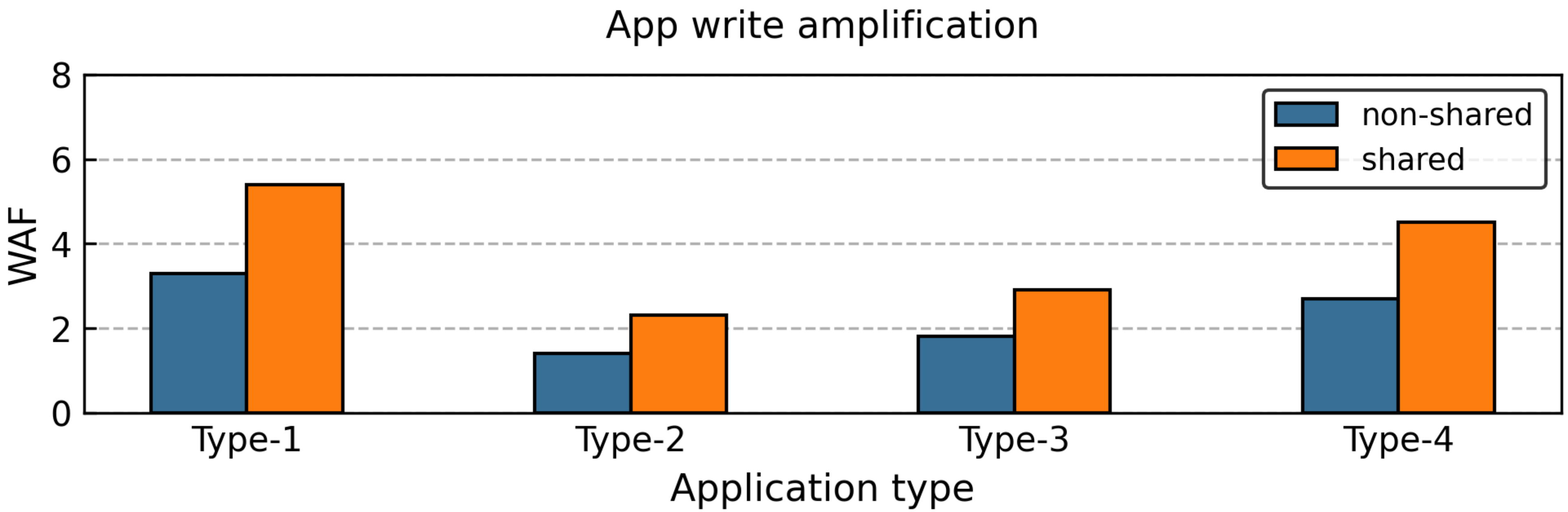
NAND-level Write Amplification

Large capacity SSDs tend to use larger super block so that frequent small writes lead to a significant increase in NAND-level write amplification (WA).



NAND-level Write Amplification

Large capacity SSDs tend to use larger super block so that frequent small writes lead to a significant increase in NAND-level write amplification (WA).



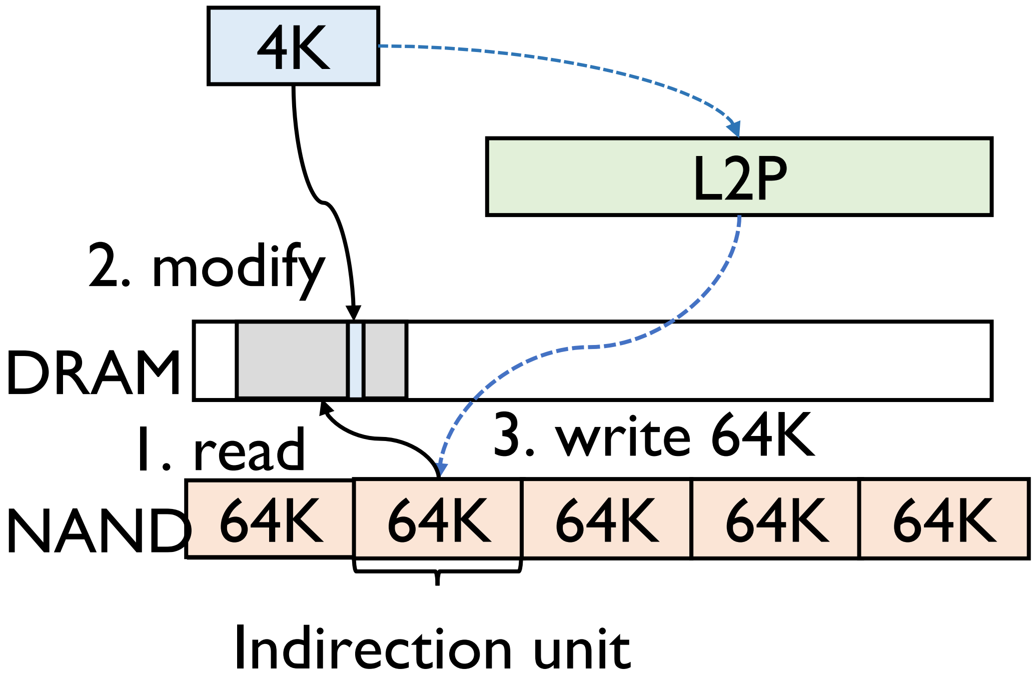
- Sharing QLC under diverse applications lead to higher NAND-level WA

Device-level Write Amplification

Large capacity SSDs tend to use larger indirection unit (e.g., 64K) so that non-optimal writes lead to device-level write amplification (WA).

- User updates 4KB data
- SSD reads 64KB data from NAND
- Updates 4KB of 64KB
- Writes whole 64KB back to NAND

16X write amplification



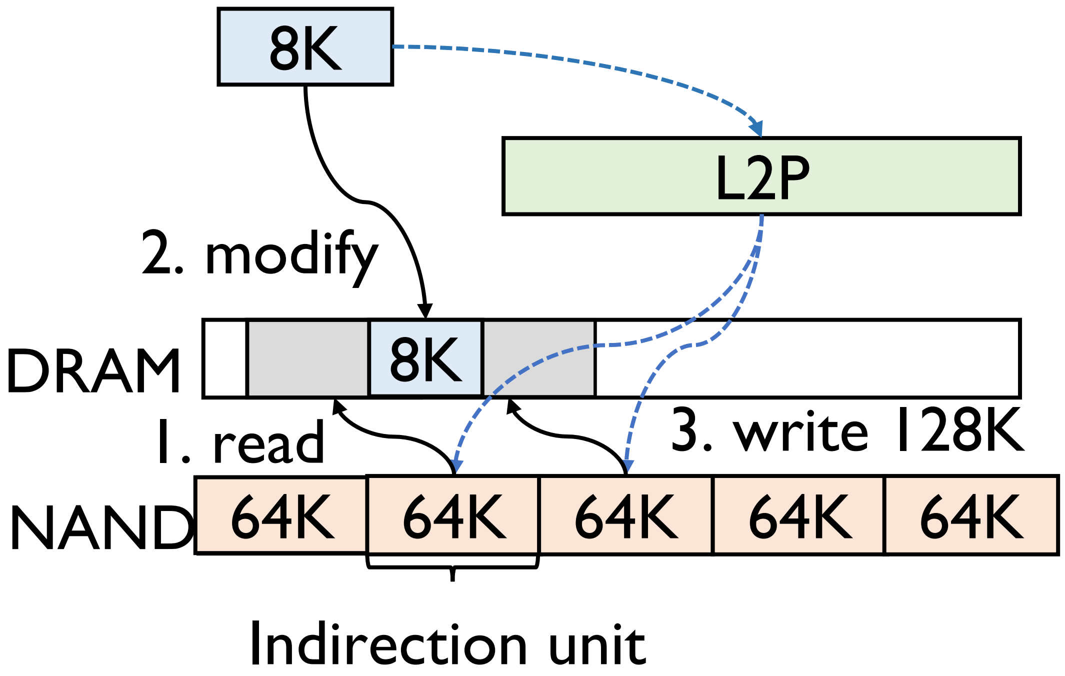
Case 1: Missized write

Device-level Write Amplification

Large capacity SSDs tend to use larger indirection unit (e.g., 64K) so that non-optimal writes lead to device-level write amplification (WA).

- User updates 8KB data
- SSD reads two 64KB data from NAND
- Updates 4KB of each 64KB
- Writes two 64KB back to NAND

16X write amplification



Case 2: Misaligned write

Endurance Analysis

Estimated NAND writes calculated via logical writes from real workloads, NAND-level write amplification, device-level write amplification with real block size distribution

	Logical Writes (TB)	NAND Writes (TB)
p50	1.23	25.07
p75	1.42	28.94
p90	1.58	32.20
p99	2.20	44.84
p999	2.54	51.76
p100	2.94	59.92

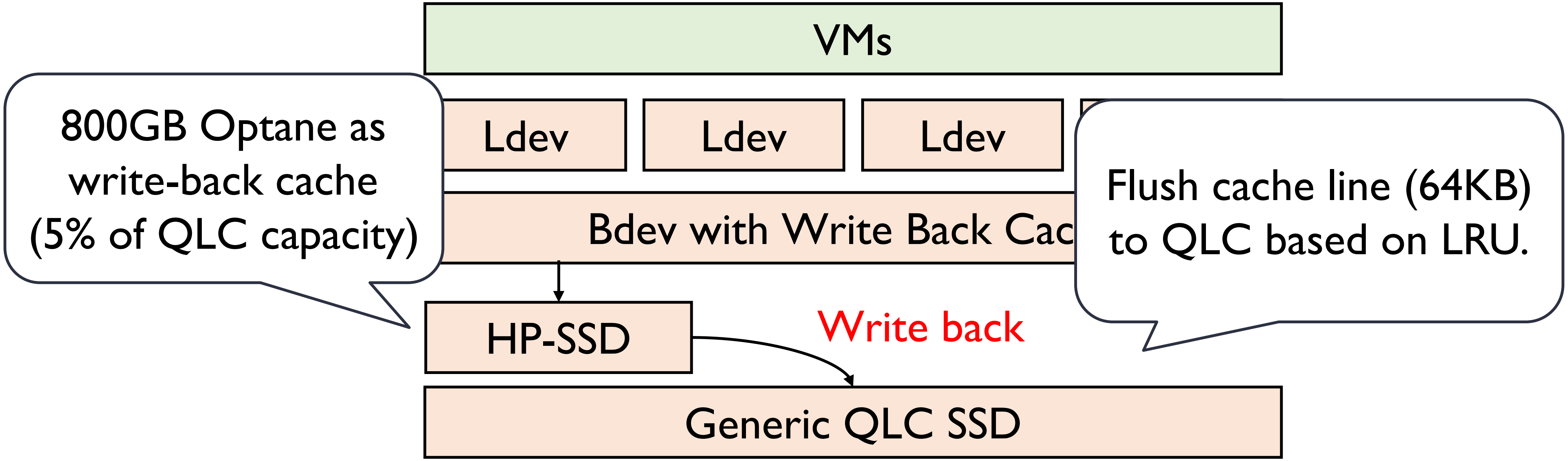
> Drive Write Per Day (DWPD) of QLC

User writes per day

Estimated writes to NAND per day

Attempt 2: QLC with Write-Back Cache

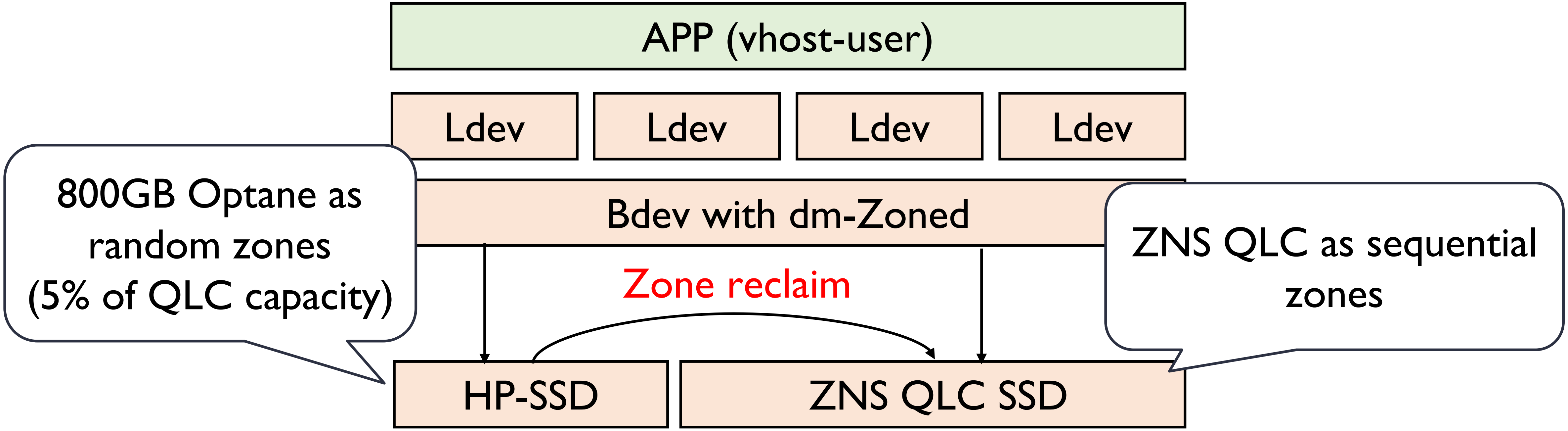
Fast SSDs, such as Optane and SLC SSD, provide higher performance and endurance.
Write-back cache can merge data in cache line granularity.



QLC with Open-CAS (Write-Back Cache)

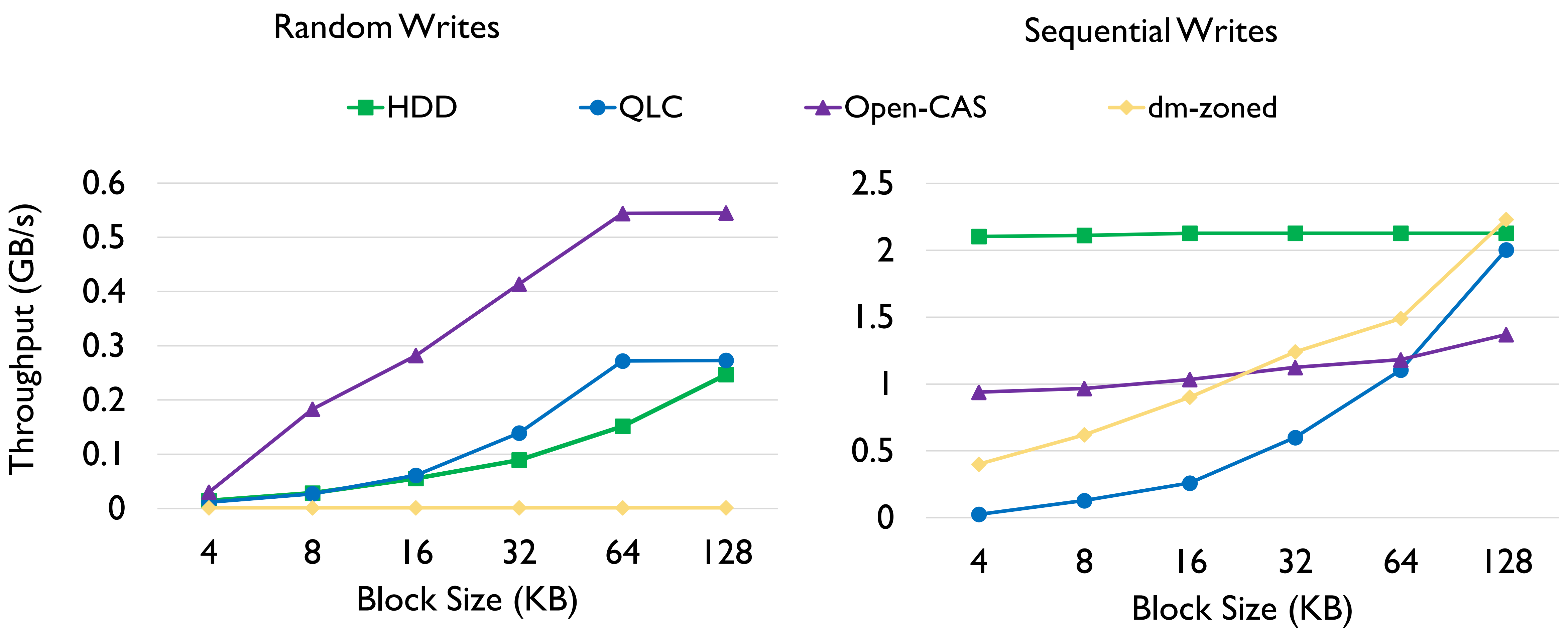
Attempt 3: QLC with ZNS and dm-zoned

Zoned Namespace SSDs remove all indirection units (no device-level write amplification) inside SSDs and let host to manage data/block mapping.



QLC with ZNS and dm-zoned

Performance Comparison

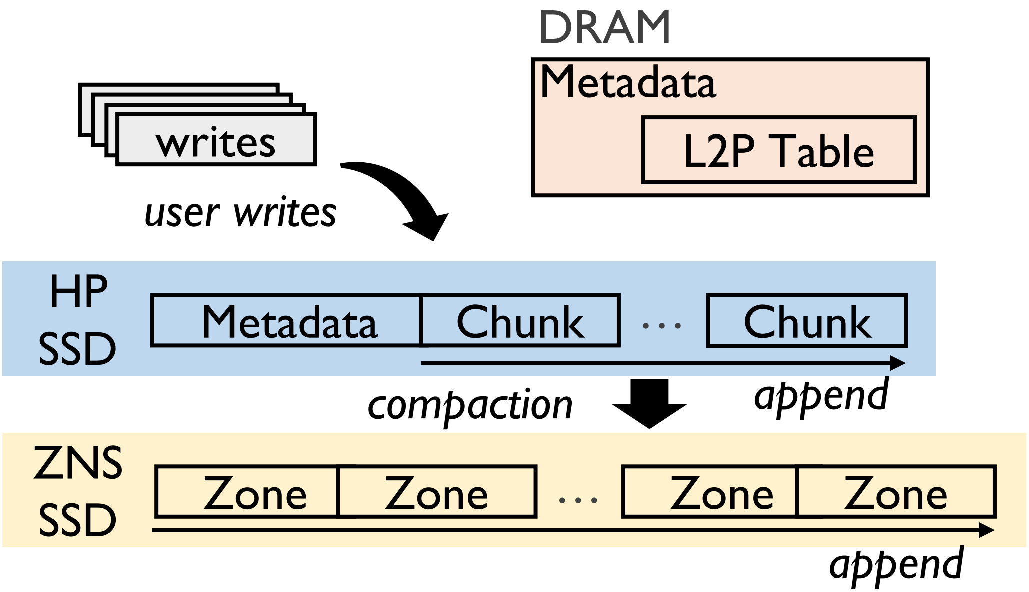


- Open-CAS can not aggregate all missized writes due to limited cache capacity.
- Dm-zoned suffers performance loss because of the zone granularity mapping.

Outline

- Background
- Motivation
- Design
- Evaluation
- Conclusion

CSAL: Cloud Storage Acceleration Layer



CSAL Overview

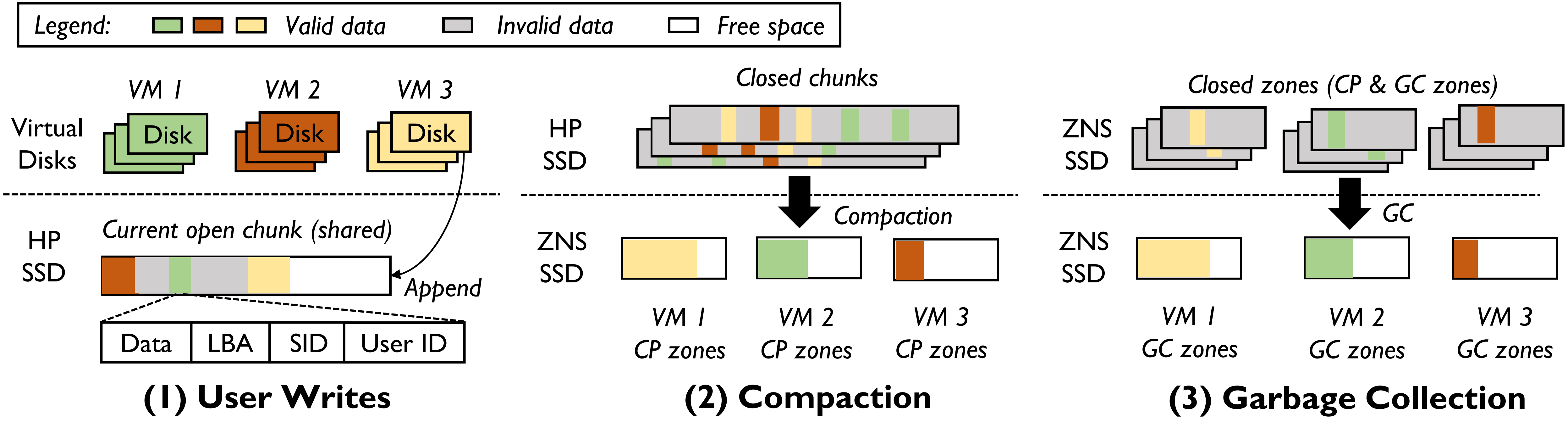
Key Ideas

- Two-level L2P page table for fine-grained logical to physical address mapping on ZNS.
- Use fast and highly endurable SSD as a long-structured write cache to aggregate data and flush to underlying ZNS QLC SSDs.

Benefits

- Mapping page with 4KB granularity (with minimal DRAM) alleviates the device-level WA.
- CSAL groups data with similar lifespans to QLC SSDs, reducing NAND-level WA.

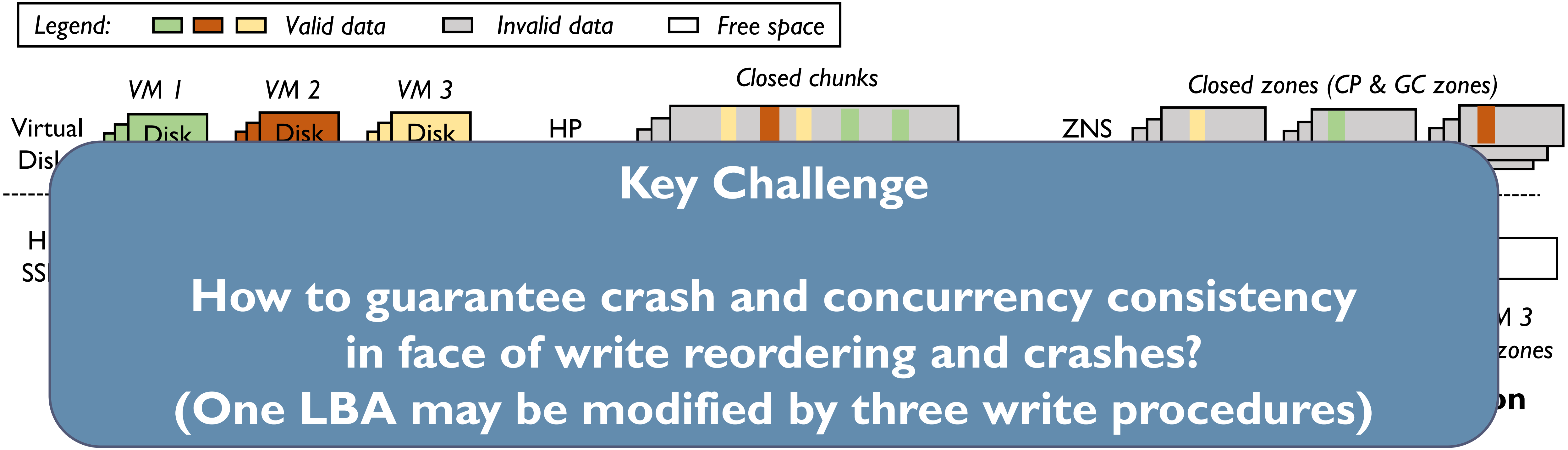
CSAL Data Flow



Three types of writes:

- User writes: append to current open chunk of log-structured write cache.
- Compaction: aggregate valid data by VMs and then flush to isolated zones of QLC.
- Garbage collection: reclaim zone spaces by VMs.

CSAL Data Flow



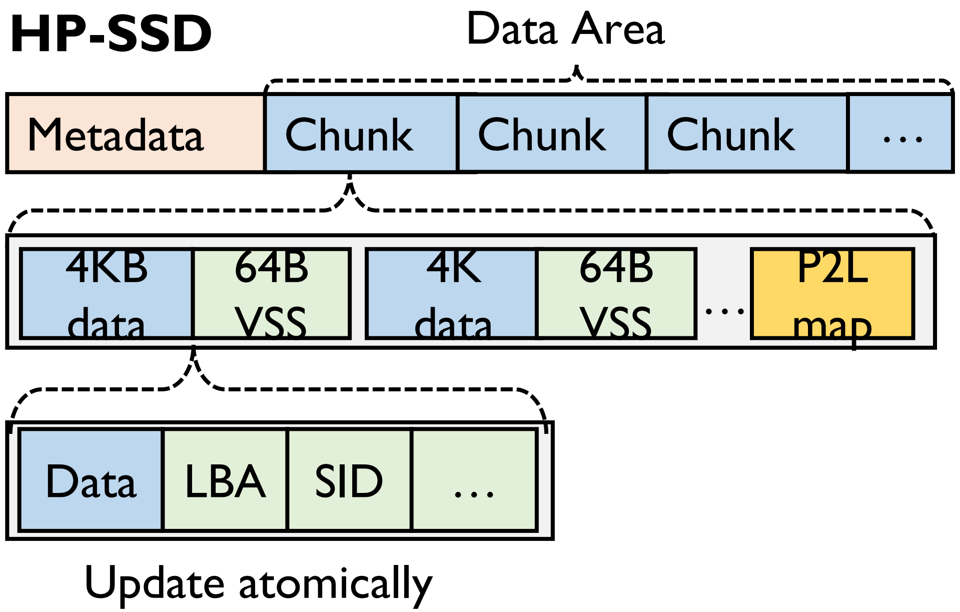
Key Challenge

How to guarantee crash and concurrency consistency in face of write reordering and crashes? (One LBA may be modified by three write procedures)

Three types of writes:

- User writes: append to current open chunk of log-structured write cache.
- Compaction: aggregate valid data by VMs and then flush to isolated zones of QLC.
- Garbage collection: reclaim zone spaces by VMs.

Crash and Concurrency Consistency



On-disk Layout

On-disk data and metadata

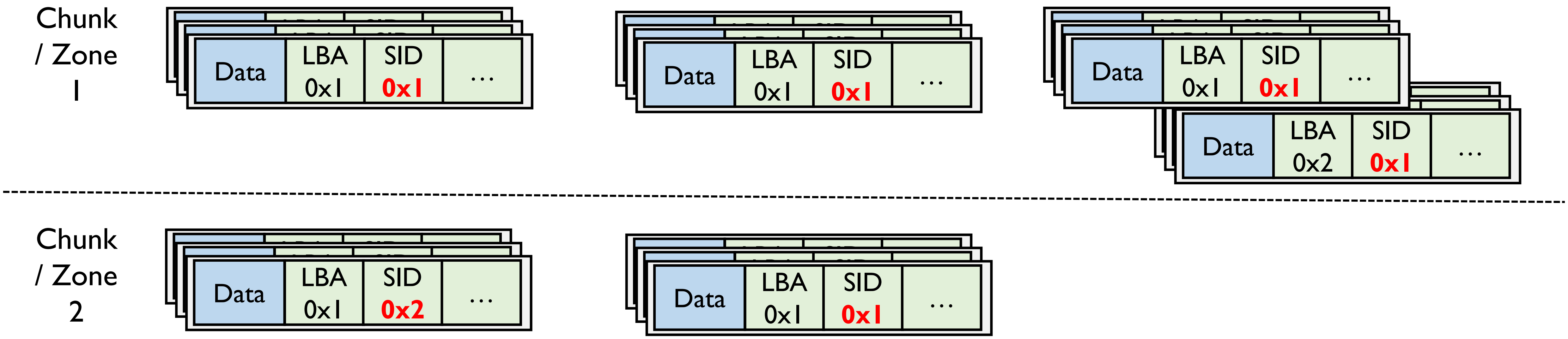
- VSS region: LBA, SID, etc.
- Update data with LBA in VSS with atomically.

Naïve solution

- After crashes, restore L2P table by scanning the whole data regions.

Crash and Concurrency Consistency

Resolving LBA conflicts (more than one PBAs pointing to an LBA)



Case 1: different SIDs and chunks/zones

Trust the one with highest SID

Case 2: same SIDs and different chunks/zones

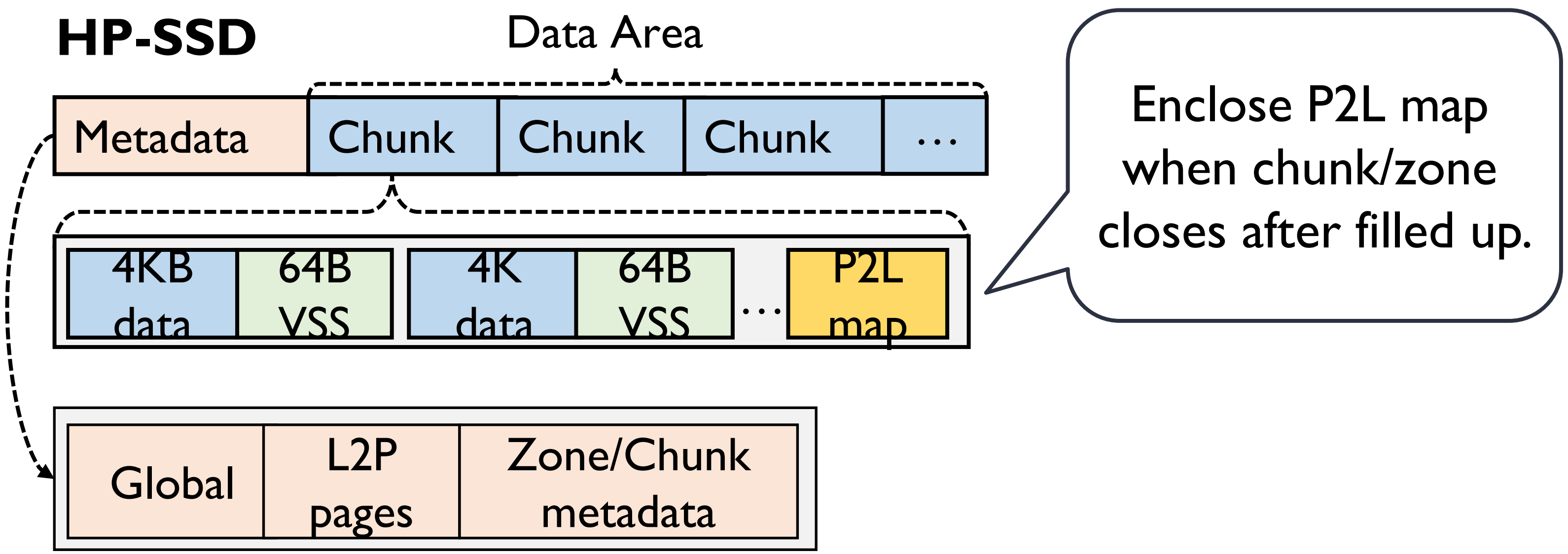
Either option is trustworthy

Case 3: same SIDs and same chunks/zones

Trust the one with largest PBA

Crash and Concurrency Consistency

Scanning whole data region (16TB) takes long time (impossible in real deployment)

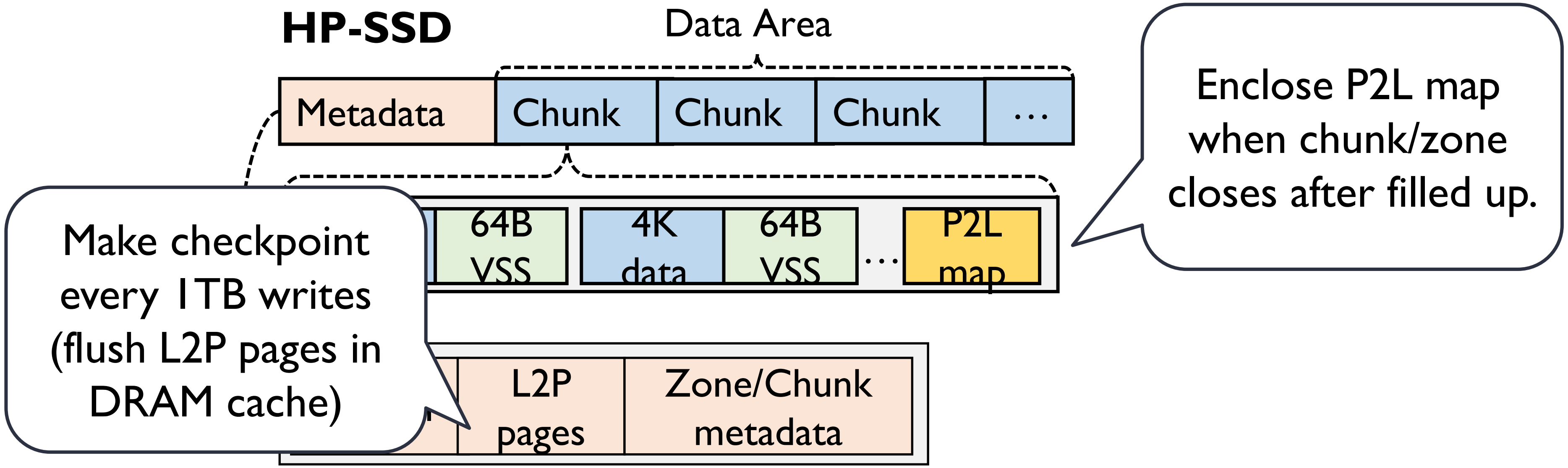


Optimization 1: adding P2L Table

- After crashes, only scan the tail of each closed chunks/zones and three open ones (one open chunk, two open zones for compaction and GC)
- Scan 32GB (P2L) + 3GB (open chunks/zones)

Crash and Concurrency Consistency

Scanning whole data region (16TB) takes long time (impossible in real deployment)



Optimization 2: adding checkpoint

- After crashes, load the L2P pages and check entries. For entries with a higher-than-checkpoint SID, read P2L map from recent 1TB writes and three open chunks/zones.
- Scan 16GB L2P table + 1GB (PL2 table) + 3GB (open chunks/zones)

Outline

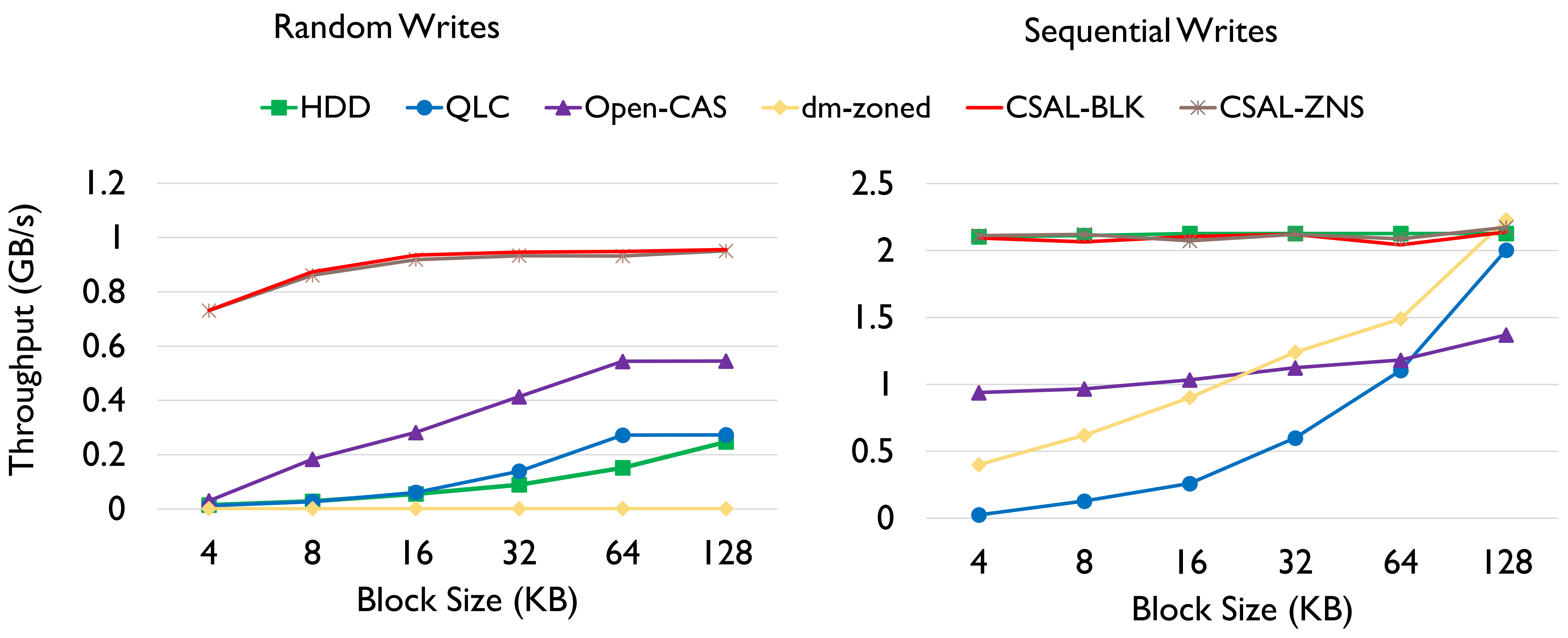
- Background
- Motivation
- Design
- Evaluation
- Conclusion

Experimental Setup

- Hardware
  - Cache: 800GB Optane P5800X SSD (800GB SLC SSD also measured)
  - QLC:
    - Standard QLC: 1x Solidigm QLC SSD P5316 16TB
    - ZNS QLC: 4x WD TLC SSD ZN540 4TB (emulating ZNS QLC via throttling)
  - CSAL-BLK (Optane + Standard QLC)
  - CSAL-ZNS (Optane + ZNS QLC)

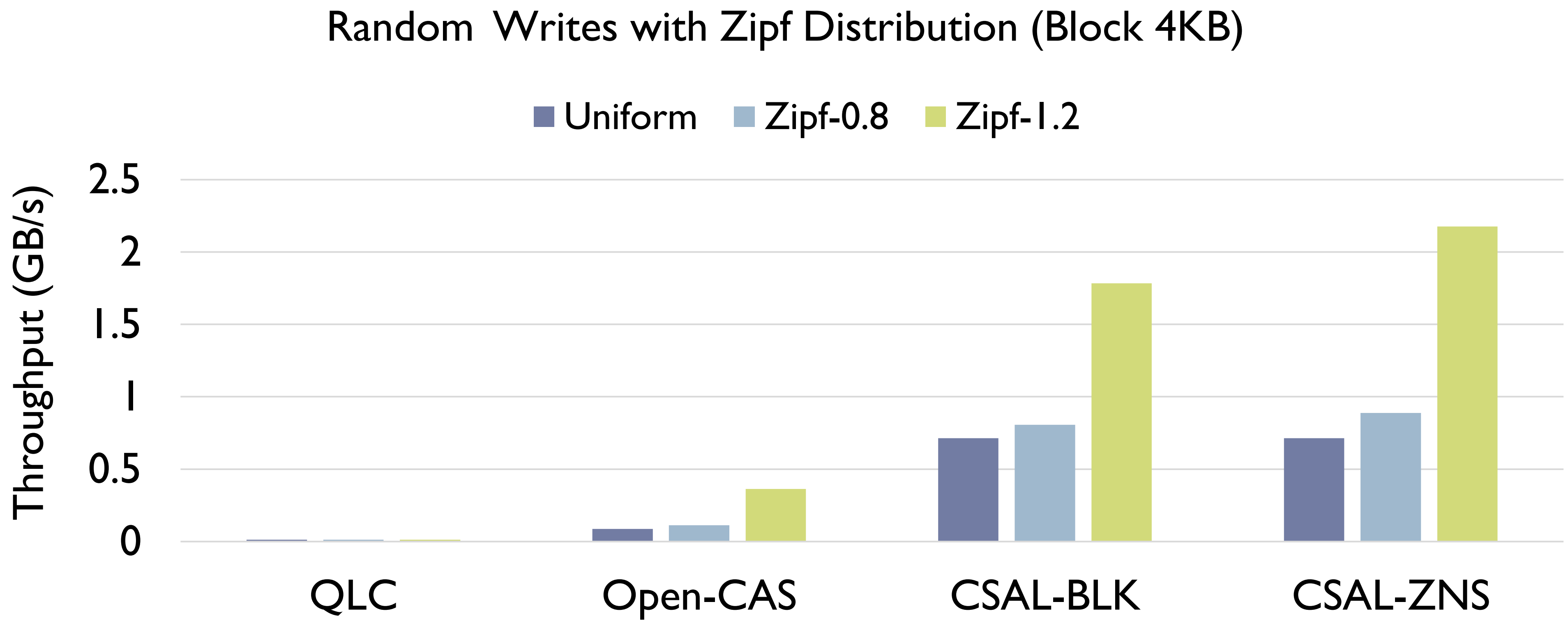
- Software
  - 8x VMs (each owns one 2TB virtual device) share one 16TB QLC
  - Hypervisor: QEMU + Vhost-NVMe
  - FIO as micro-benchmarks

Performance under Uniform Writes



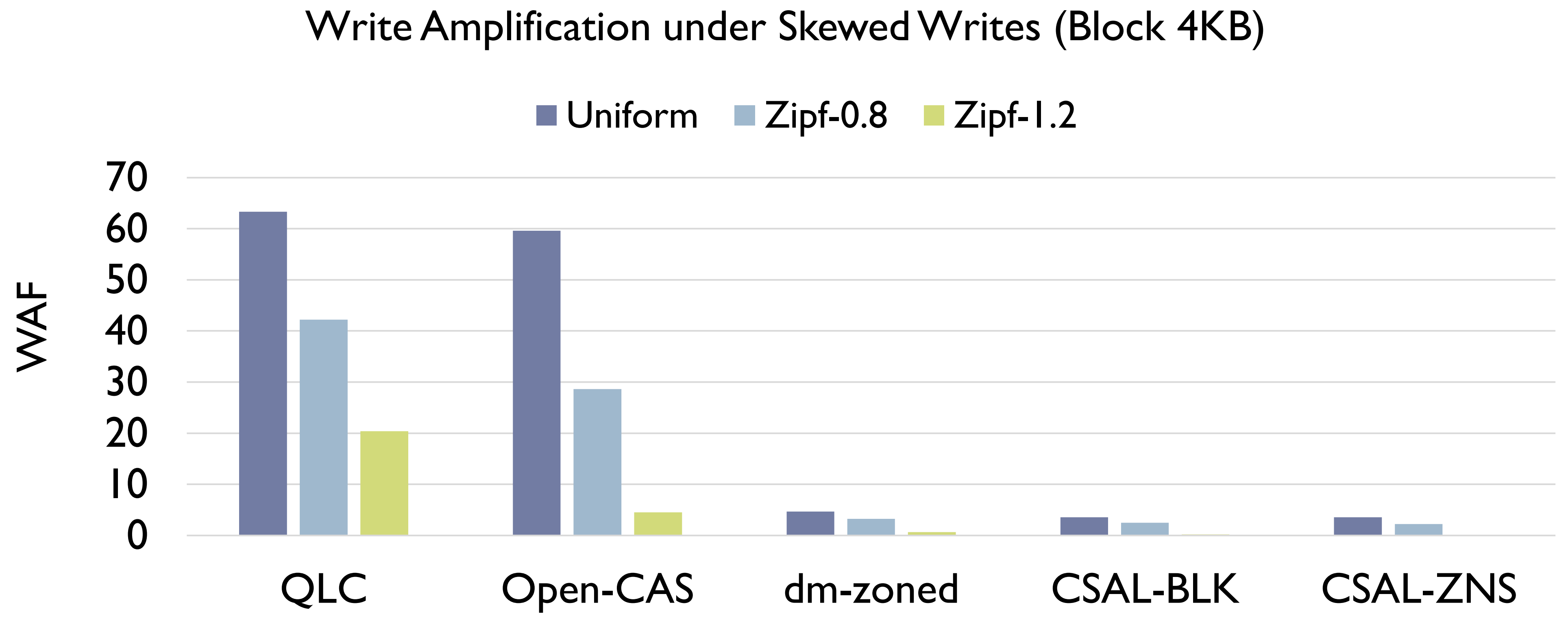
- Higher performance under random writes than all candidates.
- Comparable performance as HDDs under sequential writes.

Performance under Skewed Writes



- Up to 6x higher performance compared to Open-CAS under Zipf 1.2 (heavy skewed).
- Open-CAS suffers performance loss due to large granularity of indirection units (64K).

Write Amplification Comparison



- More skewed distribution leads to less data flushed to underlying QLC.
- Raw QLC and Open-CAS are bounded by 64K indirection units (device-level WA).

Conclusion

- Deploying high-density (QLC) SSDs to replace HDDs in cloud local disks is non-trivial.
- We identified the performance and endurance challenges due to two write amplifications.
- We proposed CSAL, a log-structured cache designed for high-density (QLC) SSDs.
- With CSAL, we can reduce two levels of write amplifications by a large margin.

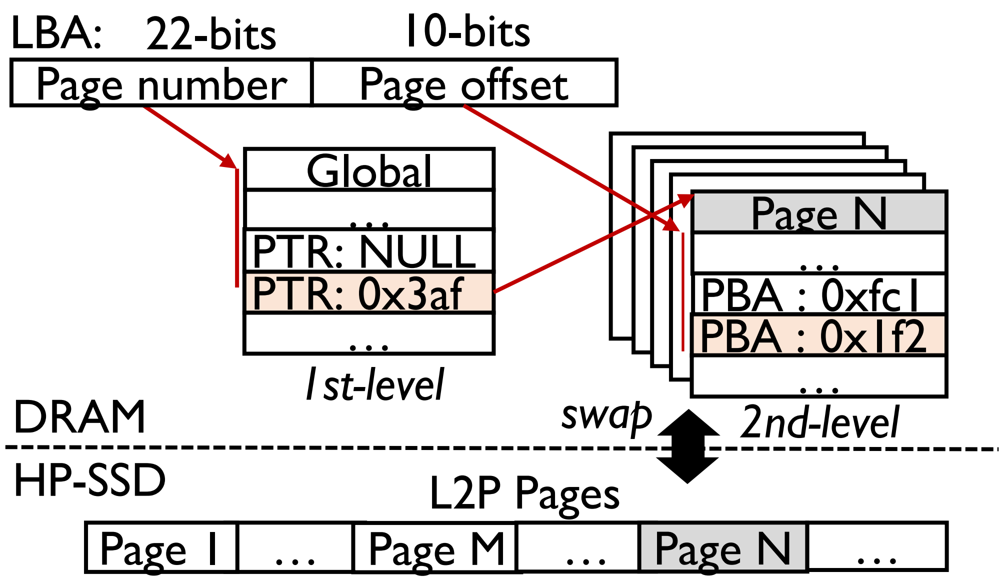
Thank You!

Q & A

CSAL is available at SPDK:
https://spdk.io/doc/ftl.html

Backup Slides

CSAL Metadata – L2P Table



Two-level L2P Table Hierarchy

L2P Table

- LBA (32bits) to PBA mapping
- Page number and offset to get PBA (32bits)
- All pages are in fast SSD.
- DRAM as page cache based on LRU

- In deployment, we use 2GB DRAM as page cache to manage 16TB storage (1x QLC)
- Totally, we use 16GB DRAM for 128TB storage (8x QLC) in a physical server