SNIA COMPUTE + MEMORY + STORAGE SUMMIT

*Architectures, Solutions, and Community*
*VIRTUAL EVENT, APRIL 11-12, 2023*

# A Host-Assisted Computational Storage Architecture


Tao Lu, Ph.D.

Research Manager, DapuStor Corporation


# Outline

- Computational Storage Drive (CSD)
- Host Cores vs. In-device Controllers/Accelerators
- Host-Assisted CSD (HA-CSD) Architecture
- Evaluation
- Conclusion

# Computational Storage Drive: In-storage Processing

Data compression and decompression

Encryption and decryption

Pattern matching and regular expressions

Data analytics and machine learning

# Computational Storage Drive Model and API: Standards

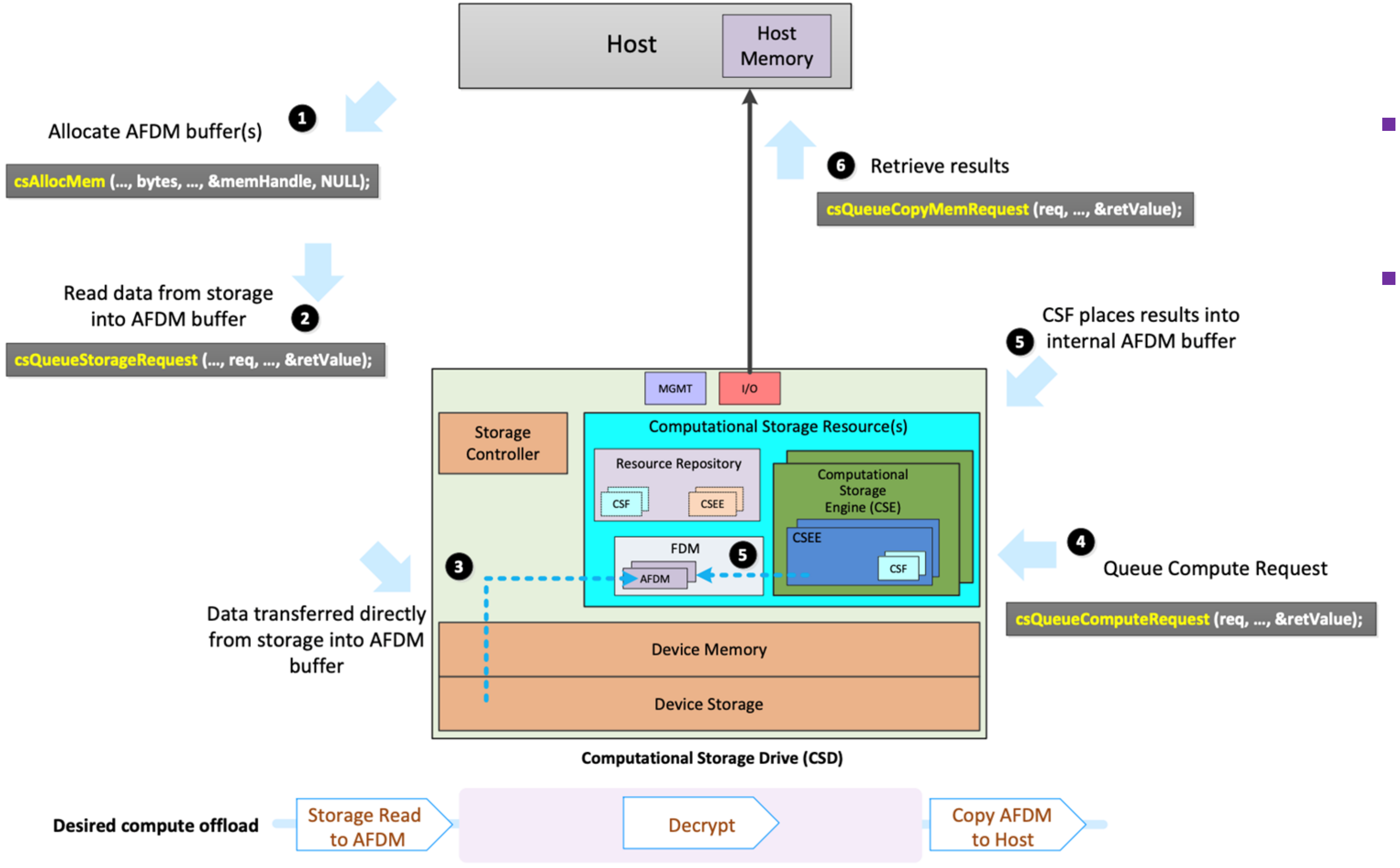



Figure 1: CSD Example API Flow [1]

- Host dictates CSD to run a function on specified data
- No collaboration between the host and device
  - Device compute power
    - Over-provisioned, or
    - Be the bottleneck !
  - **Host compute power wasted!**

**Source:** Computational Storage API Version 0.5 rev 0, https://www.snia.org/sites/default/files/technical-work/computational/draft/SNIA-Computational-Storage-API-v0.5r0-DRAFT.pdf

# Host Cores vs. In-device Controllers / Accelerators

- Host cores
  - Good Performance
  - Shall not be occupied by IO
- In-device
  - Hardware Accelerators
    - High performance
    - Not always deployed
    - Long R&D cycle
  - SSD controllers
    - Low Performance
    - Dedicated for IO

| | Write | | Read | |
|---|---|---|---|---|
| | OPS | CPU Util. | OPS | CPU Util. |
| **No Comp.** | 38,381 | 35.7% | 59,111 | 37.9% |
| **ZSTD** | 16,600 | 31.2% | 47,023 | 66.9% |

Table 1: RocksDB throughput and average host CPU utilization with and without ZSTD compression enabled. OPS indicates operations per second.

Compression Saves Storage Space but Greatly Consumes Host CPU Resources

| | AMD EPYC 7282 | Arm Cortex-A53 |
|---|---|---|
| **Compression** | 200 MB/s | 44 MB/s |
| **Decompression** | 533 MB/s | 137 MB/s |

Table 2: Single-thread compression and decompression throughput (MB/s): host AMD x86 CPU vs. SSD ARM Cortex Core. The tests are conducted with *lzbench* at 4KB compression granularity using the *silesia* dataset.

ARM Cores of the SSD Controller are weak


# FTL-CSD Offline Compression to Avoid Write Bottleneck

# Related Work

- Research Work
  - Park Y, Kim JS. **zFTL: Power-efficient data compression support for NAND flash-based consumer electronics devices.** IEEE transactions on consumer electronics. 2011.
  - Zuck, Aviad, et al. **Compression and SSDs: Where and how?**. INFLOW'14.
  - Zhang, Xuebin, et al. **Reducing solid-state storage device write stress through opportunistic in-place delta compression**. FAST'16
  - Qiao, Yifan, et al. **Closing the B+-tree vs.{LSM-tree} Write Amplification Gap on Modern Storage Hardware with Built-in Transparent Compression**. FAST'22.
- Product
  - Seagate Nytro 1351 and 1551 SSD
  - ScaleFlux CSD 2000/3000

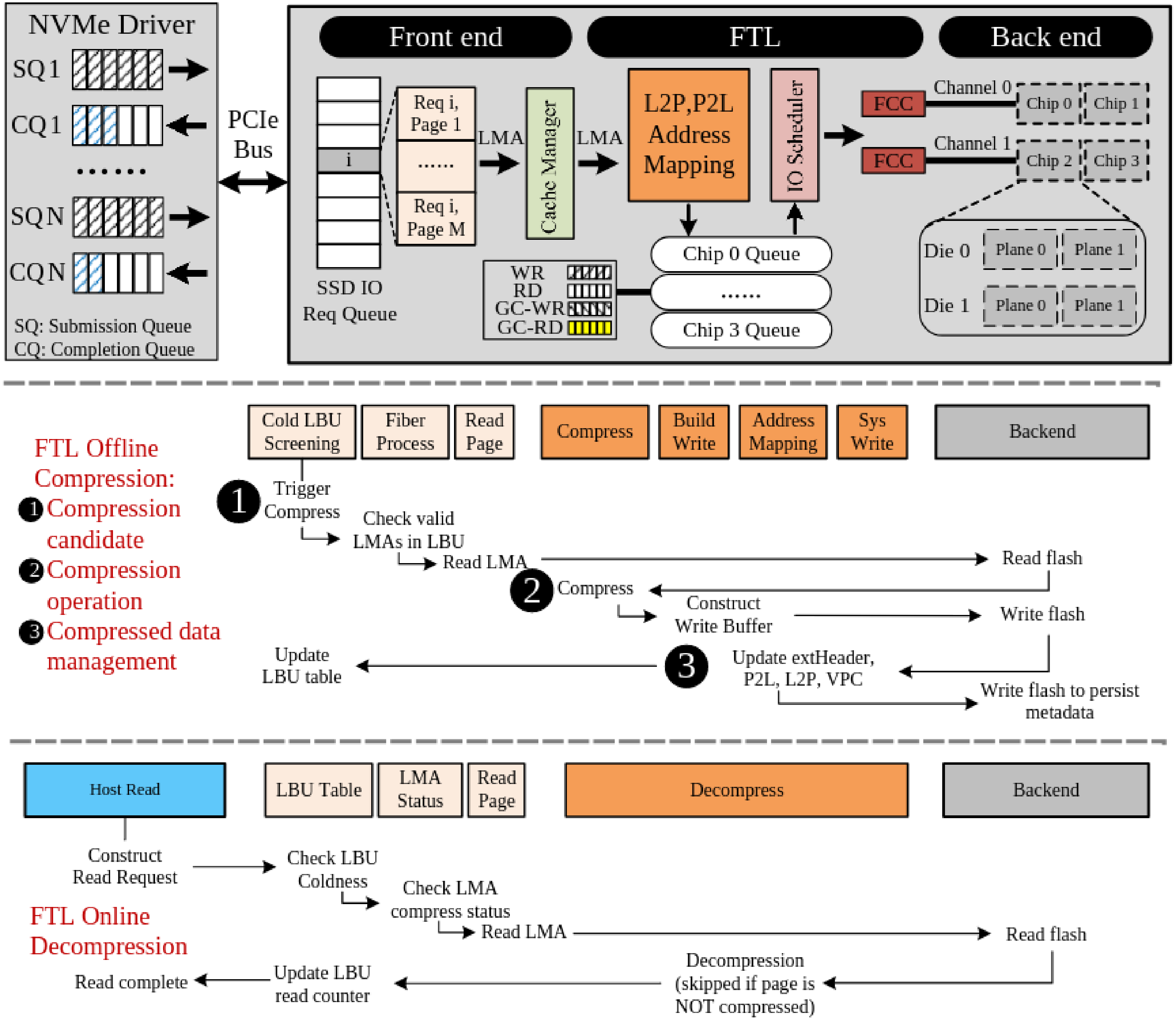
# FTL-CSD Offline Compression: Design Highlights

- Screening Compression Candidates
  - Cold Page Recognition
  - Compressible Page Recognition
- FTL Compression and Decompression
- Address Mapping
- Garbage Collection
- SSD Logical Capacity Expansion

SSD Basic Architectur
e

CSD with FTL Compression (FTL-CSD)



# FTL-CSD Performance

- Write throughput is good, thanks to the offline compression
  - As good as its standard SSD peer
- Read throughput is poor, due to the FTL decompression bottleneck
  - More than one order of magnitude lower than its standard SSD peer

SNIA COMPUTE + MEMORY + STORAGE SUMMIT

Architectures, Solutions, and Community
VIRTUAL EVENT, APRIL 11-12, 2023

# Host-Assisted CSD (HA-CSD) Architecture

# HA-CSD to Mitigate Read Bottleneck

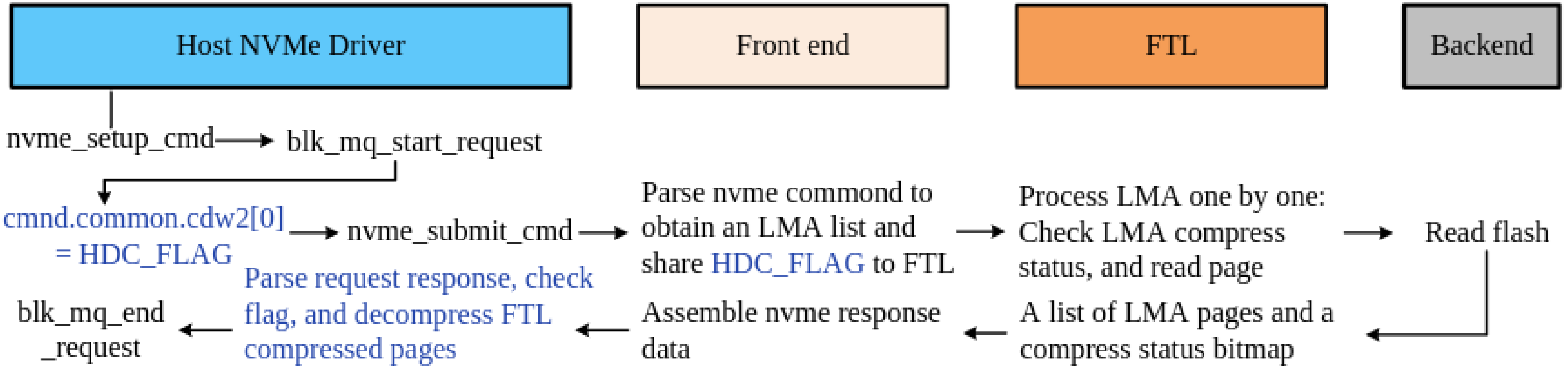



Figure 3: Read IO flow of host-assisted decompression. HDC_Flag indicates the host will decompress the data, therefore the FTL rapidly returns pages without needing to conduct decompression.

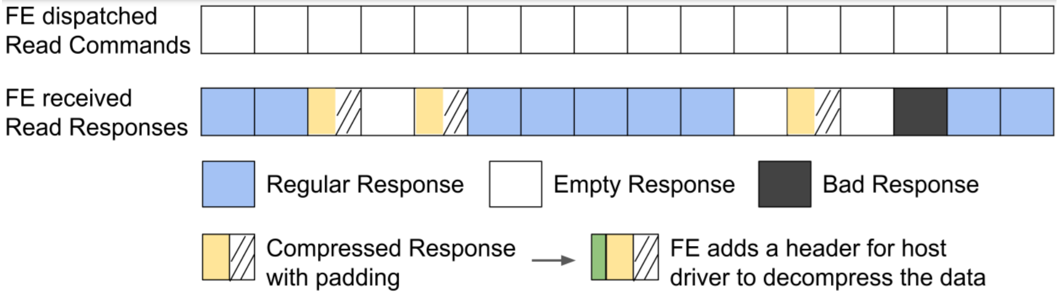



Figure 4: Illustration of firmware frontend (FE) read command and response data layouts.

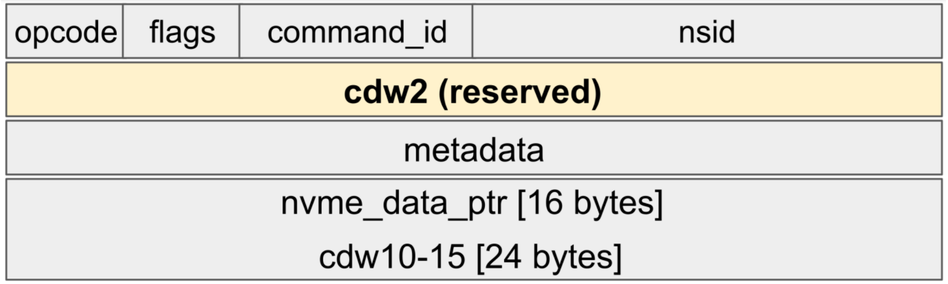



Figure 5: NVMe Base Specification v1.4 Command Format. We utilize the reserved ***cdw2*** (4 bytes) command field to store the ***HDC*** host decompression flag.

# Evaluation

| Device | Description |
|---|---|
| SSD | A regular PCIe Gen4 commercial SSD without in-storage compression |
| FTL-CSD | The PCIe Gen4 SSD with compression and decompr ession integrated in the FTL |
| HA-FTL-CSD | Enable host-assisted decompression for the FTL-CSD |
| FPGA-CSD | SSD with an in-storage FPGA compression module (ScaleFlux CSD2000) |

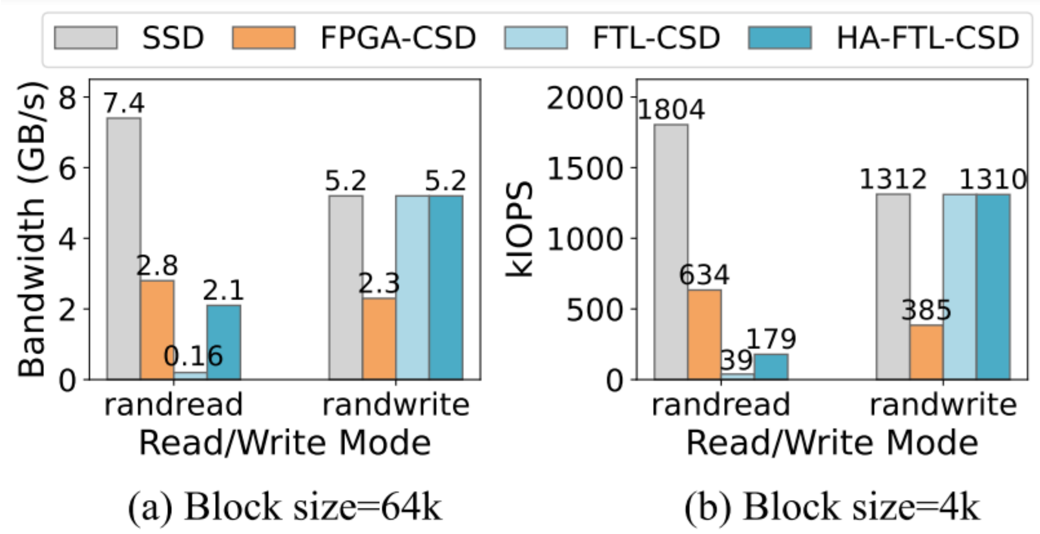



Figure 6: Read and write throughput with fio benchmark

# Evaluation

| Device | Description |
|---|---|
| SSD | A regular PCIe Gen4 commercial SSD without in-storage compression |
| FTL-CSD | The PCIe Gen4 SSD with compression and decompr ession integrated in the FTL |
| HA-FTL-CSD | Enable host-assisted decompression for the FTL-CSD |
| FPGA-CSD | SSD with an in-storage FPGA compression module (ScaleFlux CSD2000) |

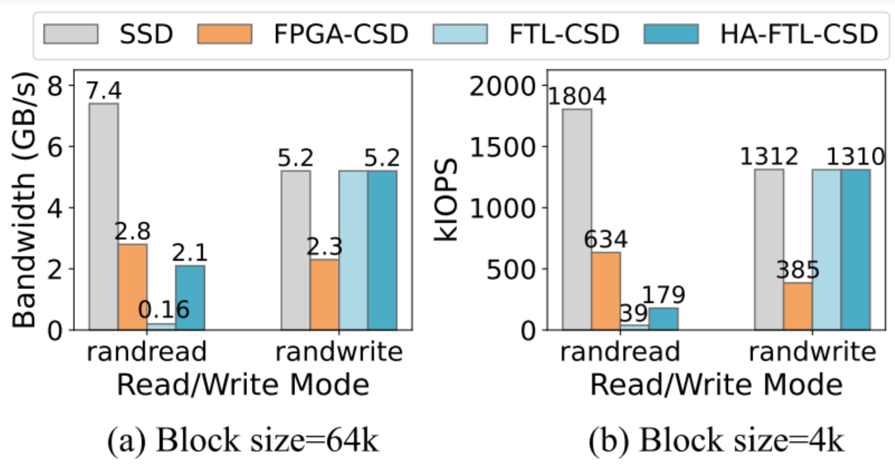
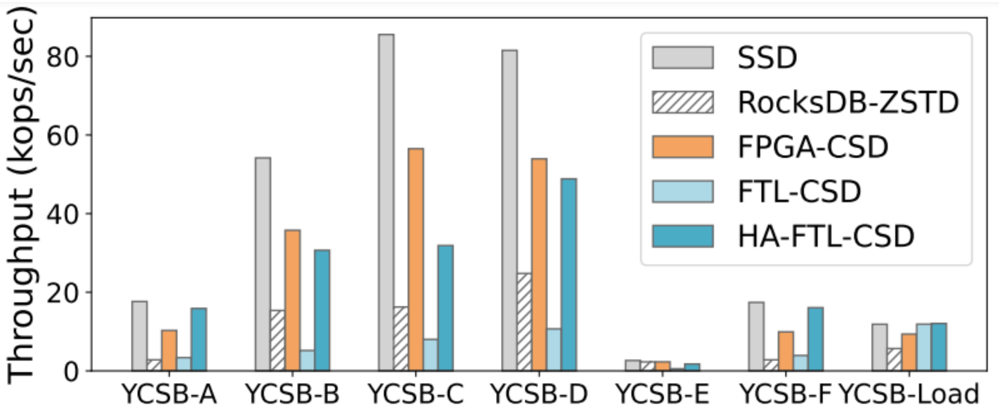
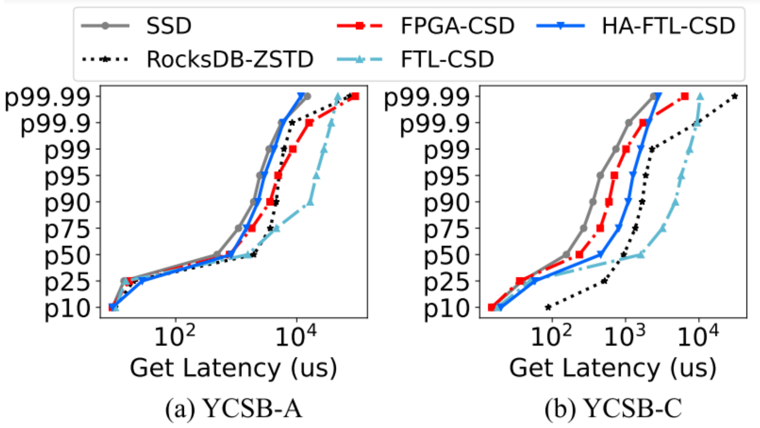



Figure 6: Read and write throughput with fio benchmark



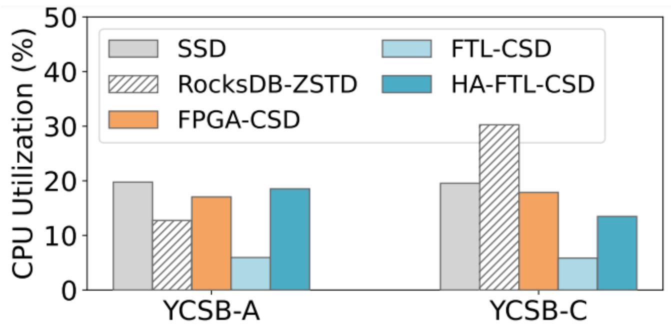




Figure 7: Comprehensive evaluation with YCSB benchmark

# Conclusion

- **In-storage Hardware Accelerator**
  - Pros
    - Performance: Hardware is the King
  - Cons
    - Hardware is costly and not always available; Not that flexible
- **SSD Controller**
  - Pros
    - Always available; Programmable
    - Well handle offline tasks such as compression
  - Cons
    - Wimpy compute power, cannot well handle online tasks (e.g. decompression in critical IO path)
- **HA-CSD**
    - Utilize the host CPU to assist the SSD controller
    - Enable computational functions supported by FTL firmware to achieve near-hardware performance

SNIA COMPUTE + MEMORY + STORAGE SUMMIT

*Architectures, Solutions, and Community*
VIRTUAL EVENT, APRIL 11-12, 2023

# *Thank You!*

## Please take a moment to rate this session.

Your feedback is important to us.