

# Understanding NVMe Zoned Namespace (ZNS) Devices

Nick Tehrany and Animesh Trivedi  
Contact: [N.Tehrany@student.tudelft.nl](mailto:N.Tehrany@student.tudelft.nl)  
Homepage: <https://nicktehrany.github.io/>  
June 8-10  
Compsys'22



# Flash Storage

SSD



- Multiple GB/s Bandwidth
- Millions IOPS

# Flash Storage

SSD



- Multiple GB/s Bandwidth
- Millions IOPS

Flash Chip



- $\mu$ -second r/w latency
- **cannot** overwrite  
must erase

# Flash Storage

SSD



- Multiple GB/s Bandwidth
- Millions IOPS



- Package flash chips together
- Erase multiple at once

# Flash Storage

SSD



**FAST**

- Multiple GB/s Bandwidth
- Millions IOPS

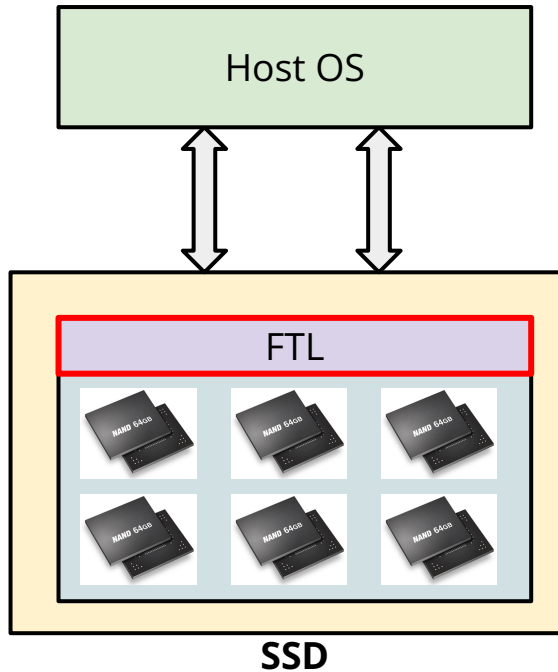


- Package flash chips together
- Erase multiple at once

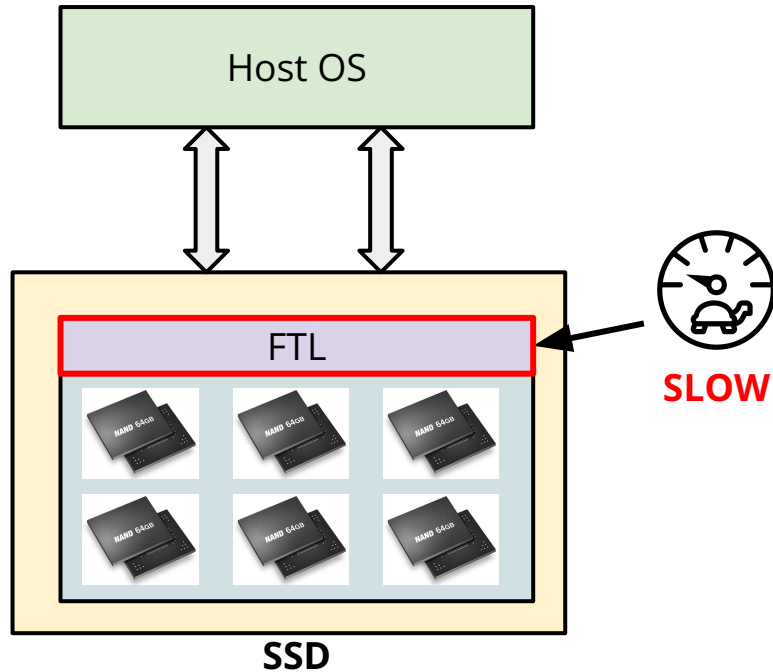


**GC**

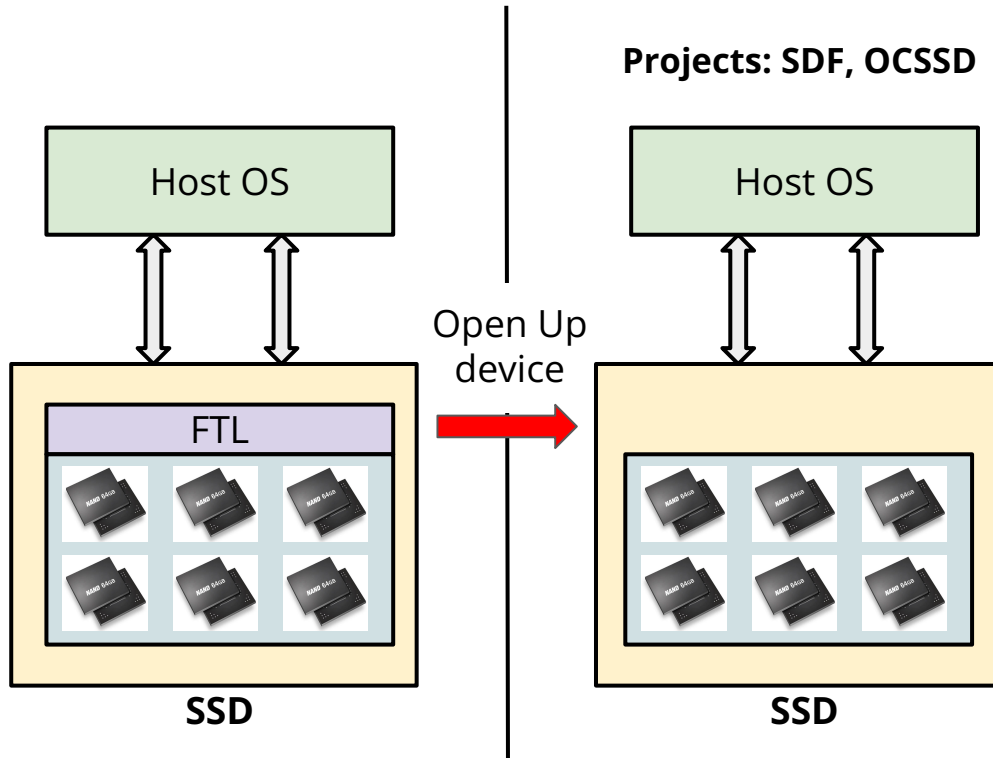
# Flash Storage Integration



# Flash Storage Integration



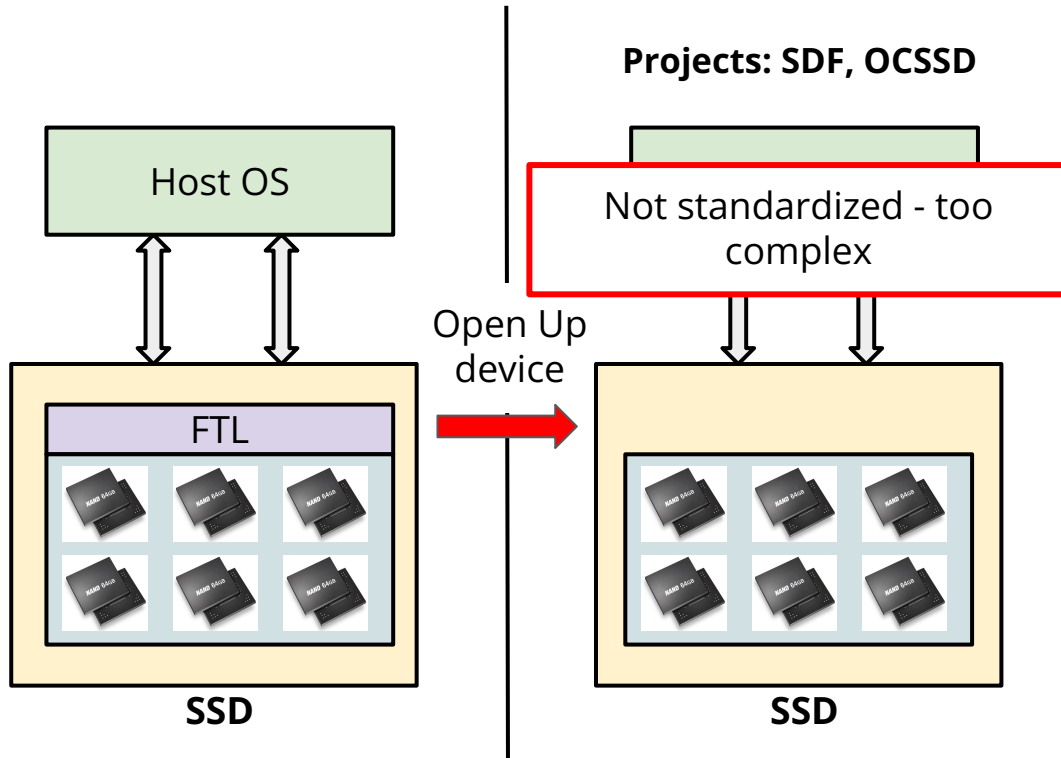
# Flash Storage Integration



1. Ouyang, Jian, et al. "SDF: Software-defined flash for web-scale internet storage systems." Proceedings of the 19th international conference on Architectural support for programming languages and operating systems. 2014.
2. Picoli, Ivan Luiz, et al. "Open-Channel SSD (What is it Good For)." CIDR. 2020.

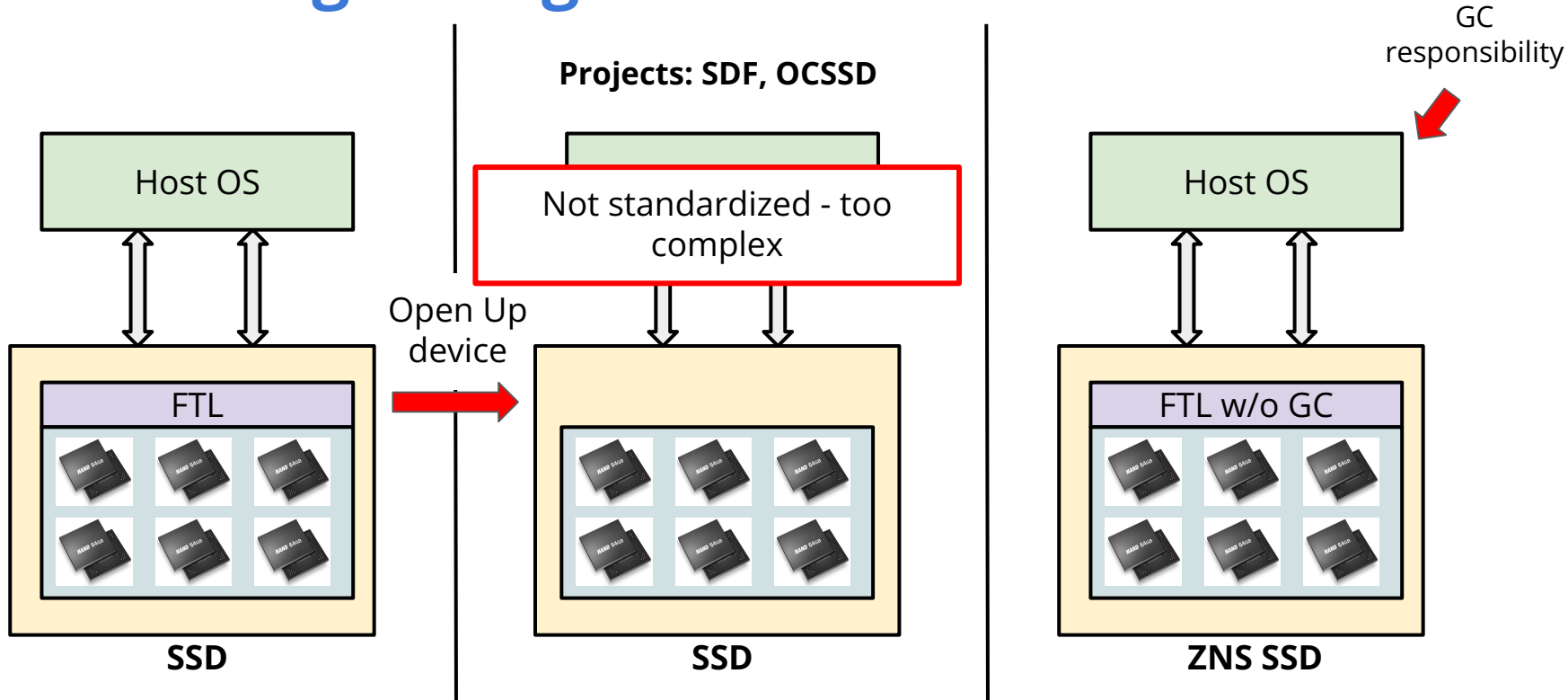


# Flash Storage Integration



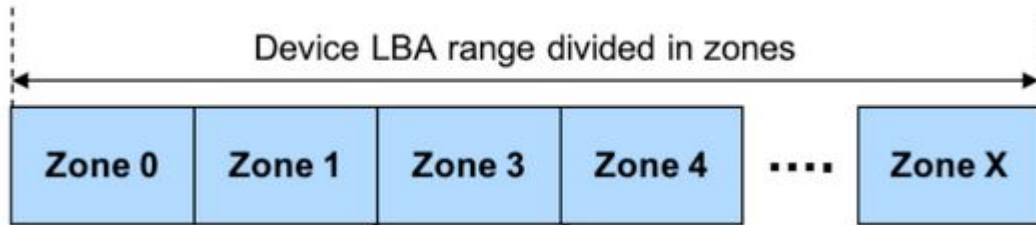
1. Ouyang, Jian, et al. "SDF: Software-defined flash for web-scale internet storage systems." Proceedings of the 19th international conference on Architectural support for programming languages and operating systems. 2014.
2. Picoli, Ivan Luiz, et al. "Open-Channel SSD (What is it Good For)." CIDR. 2020.

# Flash Storage Integration

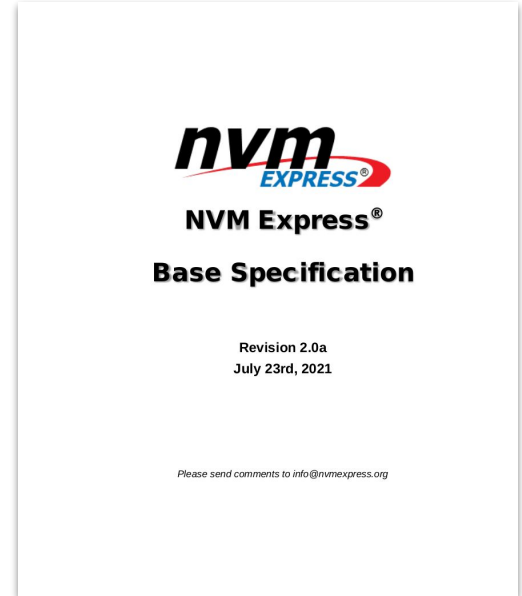


1. Ouyang, Jian, et al. "SDF: Software-defined flash for web-scale internet storage systems." Proceedings of the 19th international conference on Architectural support for programming languages and operating systems. 2014.
2. Picoli, Ivan Luiz, et al. "Open-Channel SSD (What is it Good For)." CIDR. 2020.

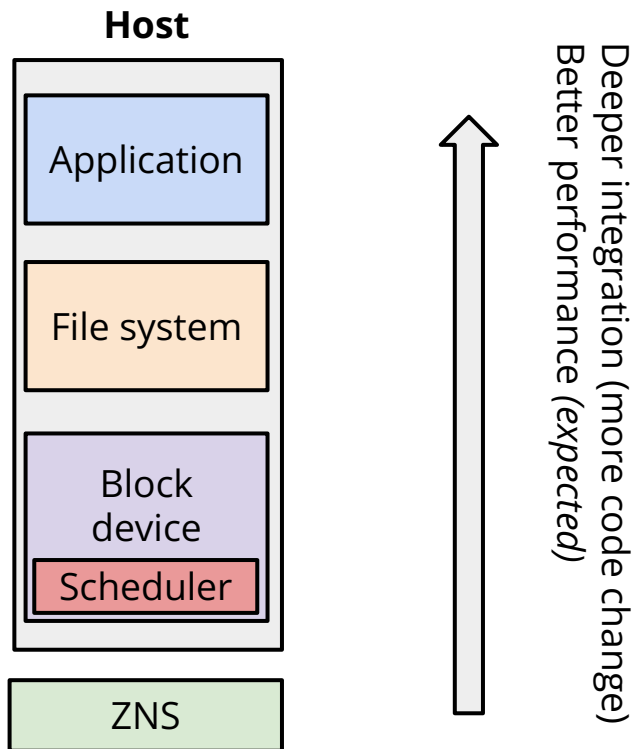
# Zoned Namespace (ZNS) Devices



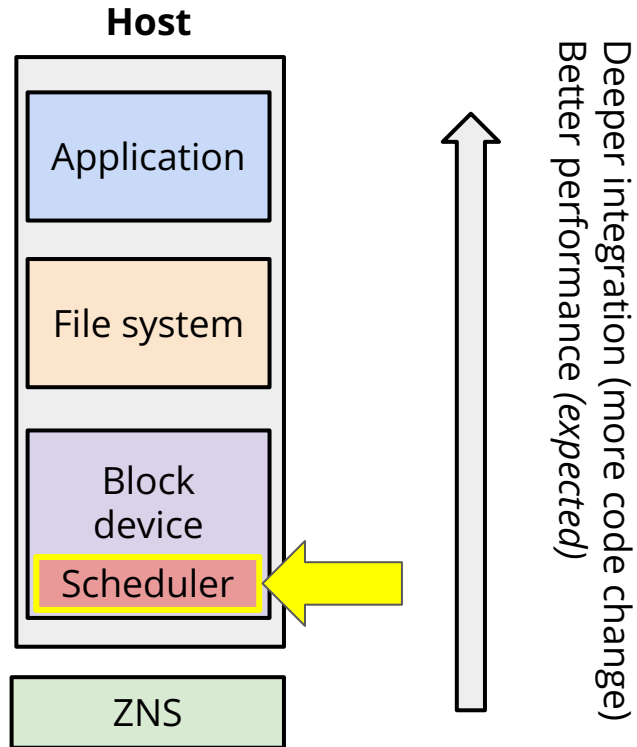
- Append-only sequential writes
- Zone erase before overwrite
- 1 outstanding write per zone



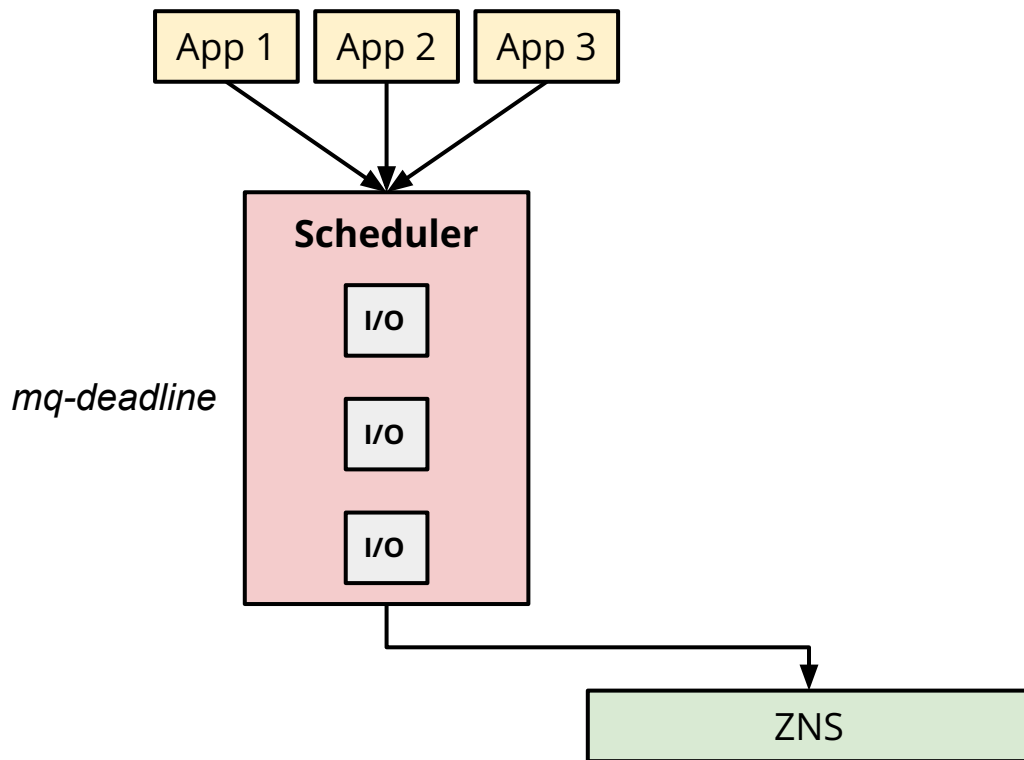
# ZNS Integration - Tradeoff



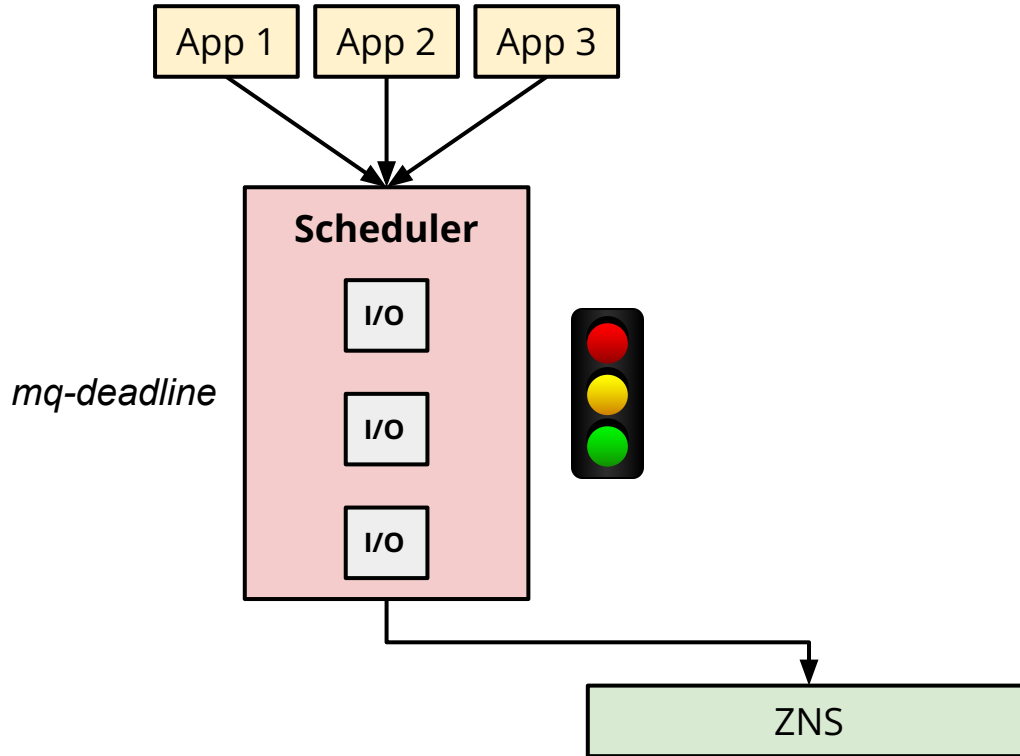
# ZNS Integration - Tradeoff



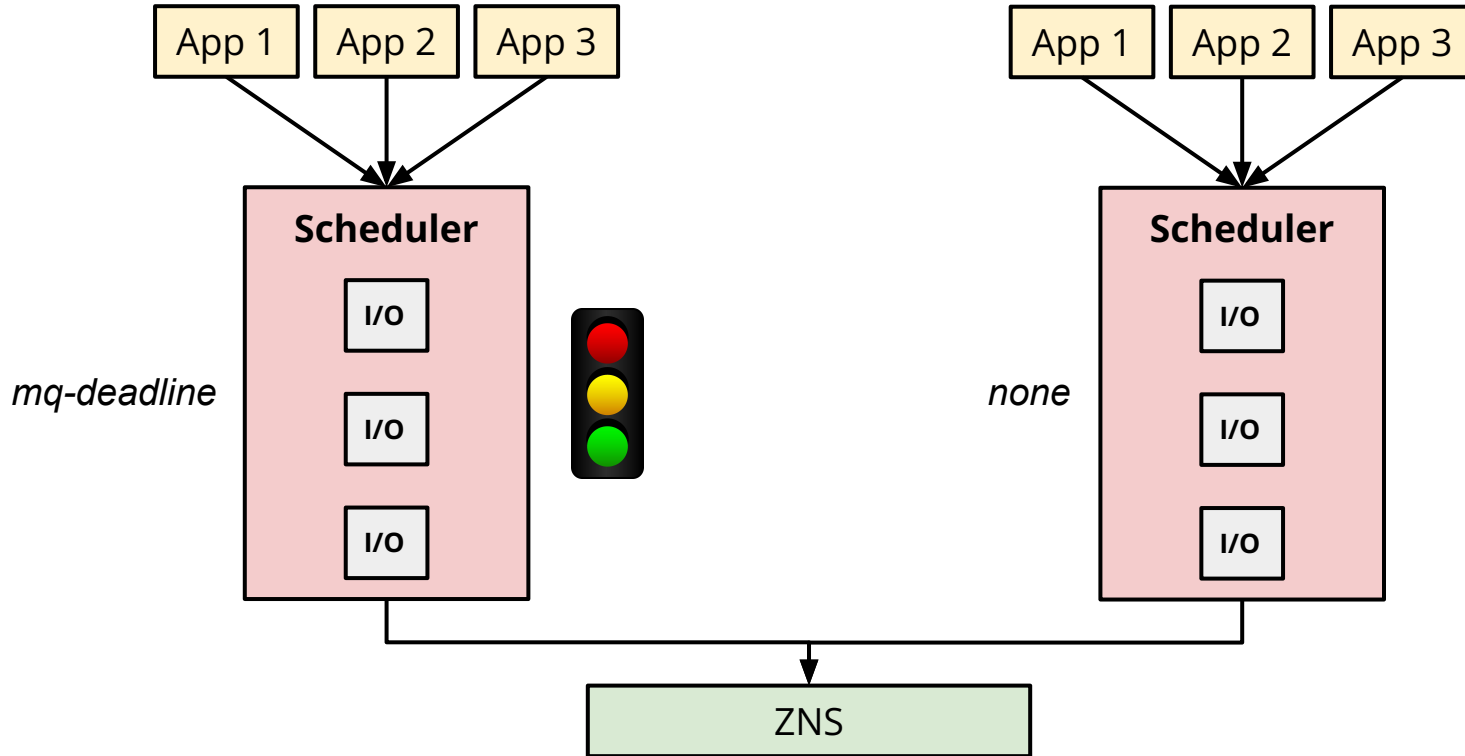
# ZNS I/O Scheduling



# ZNS I/O Scheduling

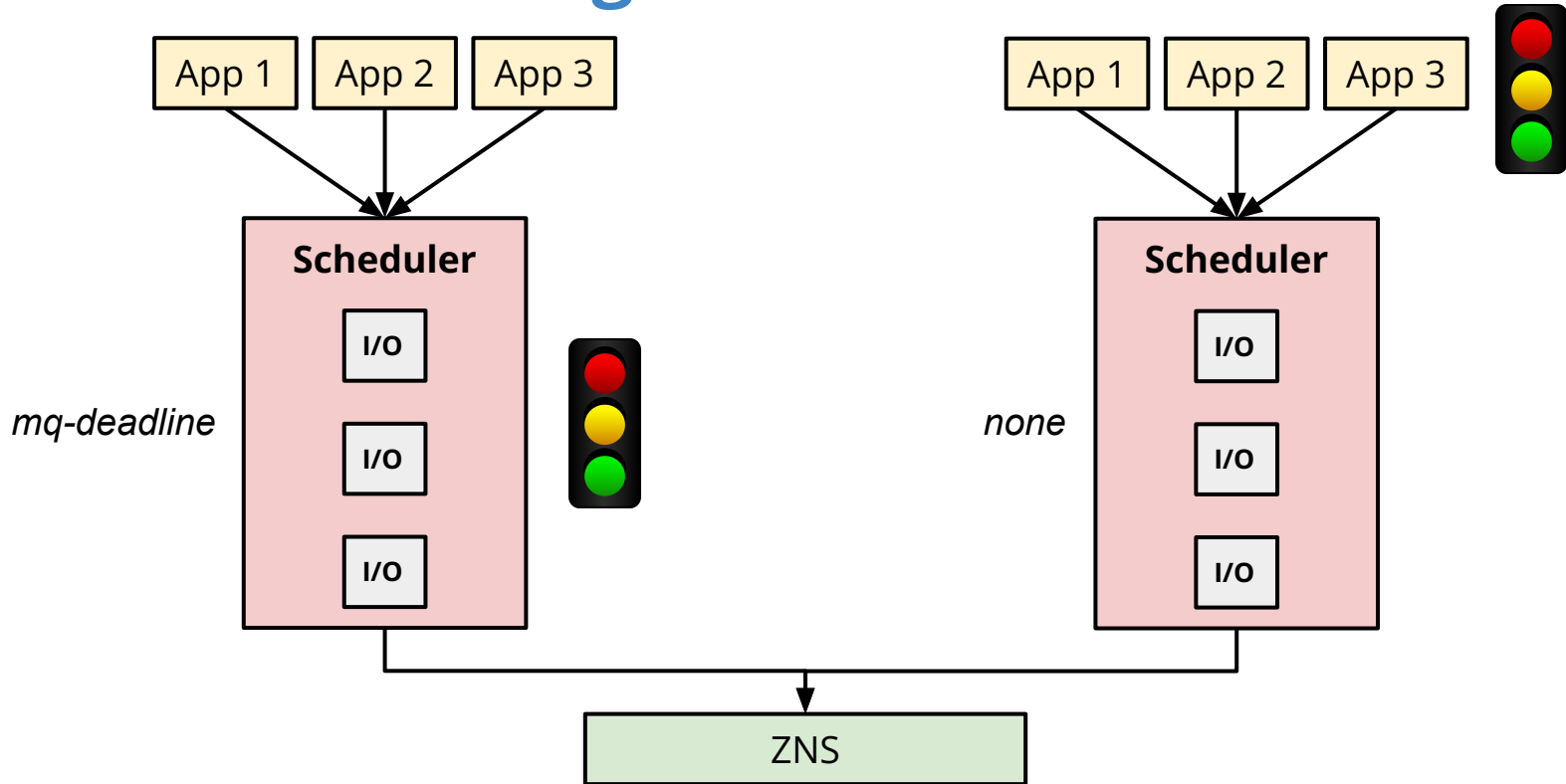


# ZNS I/O Scheduling



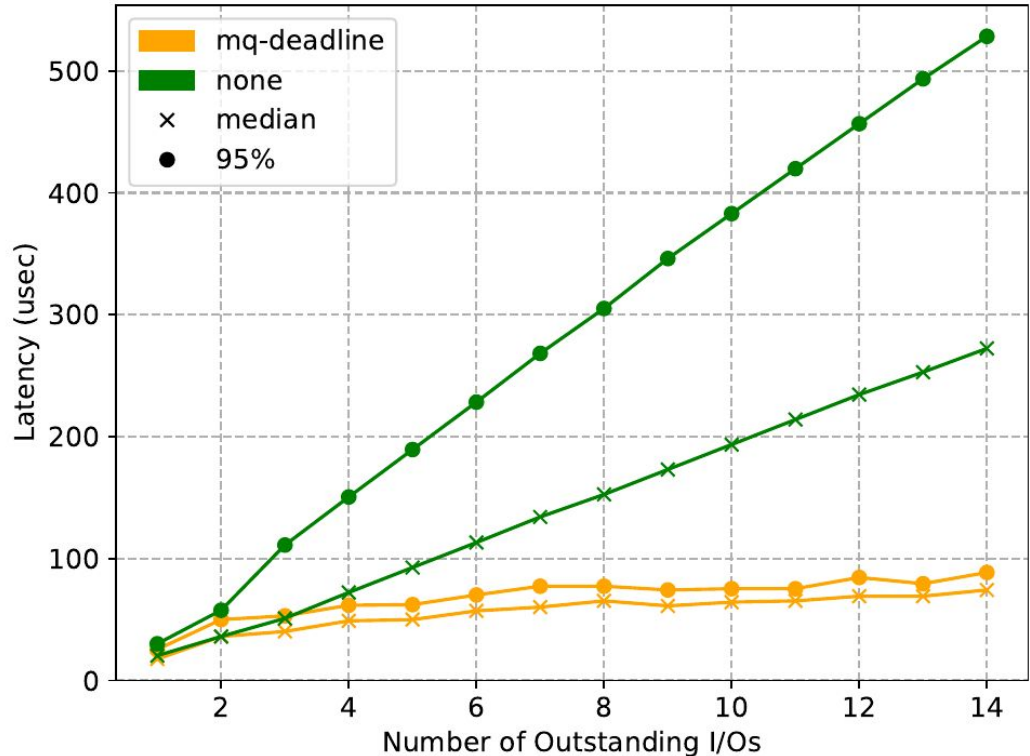


# ZNS I/O Scheduling

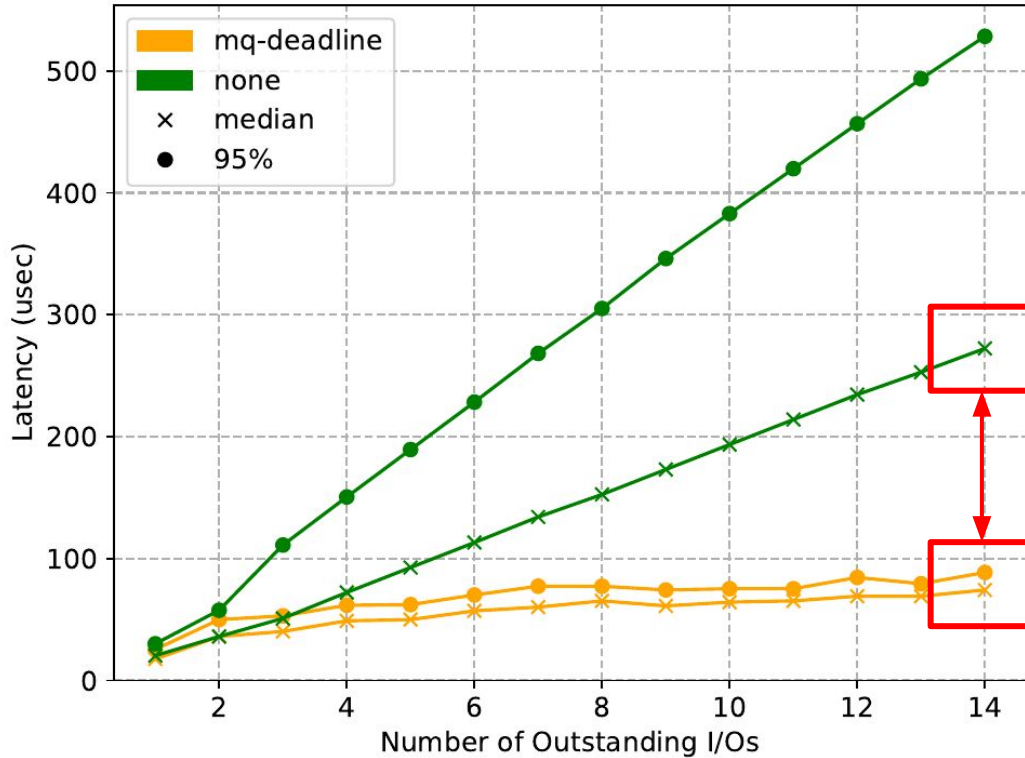


# Initial Results

- Write performance of both scheduling configurations:
  - *mq-deadline* - increasing number of outstanding I/Os in a single zone
  - *none* - single I/O per zone and increasing concurrent zones

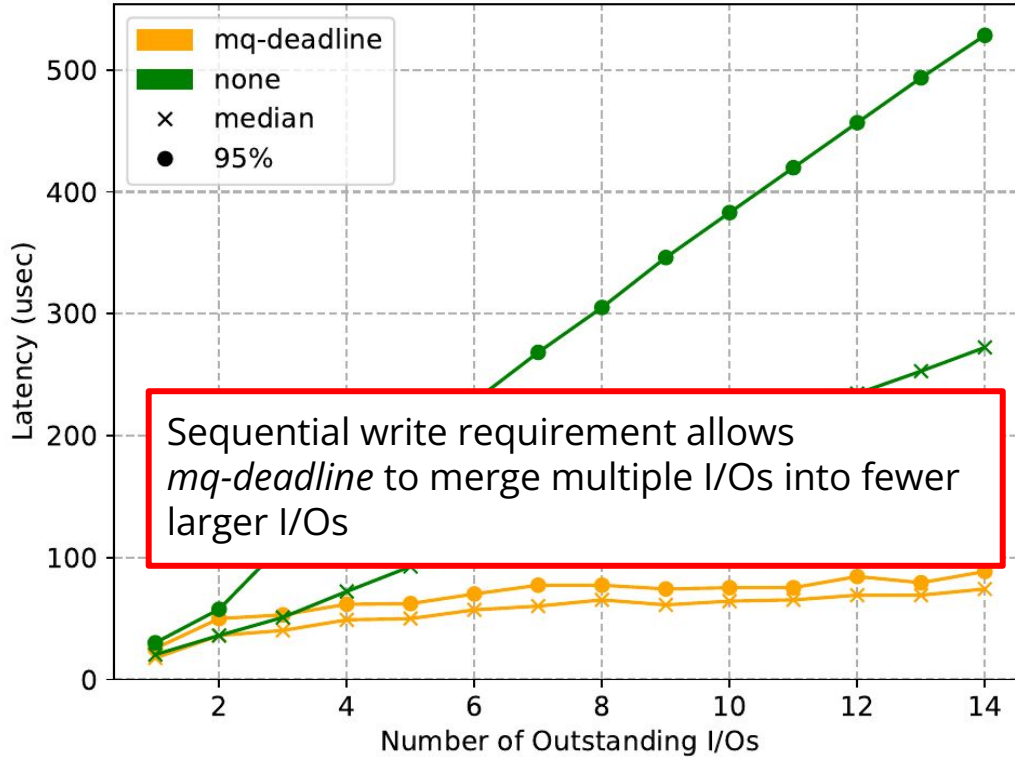


# Initial Results



73% lower median latency with *mq-deadline*

# Initial Results



# Ongoing Work

- Performance at the varying levels of integration
- GC implications at the different levels
- Concurrent instances of different levels

# Summary

- ZNS present unique addition to host storage stack
- Gaining significant attention in research community

The screenshot shows the arXiv preprint page for the paper "Understanding NVMe Zoned Namespace (ZNS) Flash SSD Storage Devices" by Nick Tehrani and Animesh Trivedi. The page includes the arXiv logo, the title, authors, and a detailed abstract. The abstract discusses the standardization of NVMe Zoned Namespaces (ZNS) in the NVMe 2.0 specification and the challenges of integrating ZNS devices into a host system. It mentions that ZNS devices push certain operations regarding data placement and garbage collection out from the device to the host, which allows for more optimal collection overheads and lower device write amplification. The authors report on their findings, including the need for larger I/O sizes to saturate the ZNS device bandwidth and the importance of configuration for I/O schedulers. The page also features a submission history section and a "Code & Data" tab.

<https://arxiv.org/abs/2206.01547>

The screenshot shows the GitHub repository page for "ZNS-Study". The repository is owned by "nicktehrany" and is currently on the "master" branch. It has 75 commits and 1 branch. The repository contains several files and folders, including "IO\_Performance", "figures", "plot", "gitignore", "LICENSE", "README.md", "get\_zoned\_device\_info", "plot.py", and "run\_benchs". The "README.md" file is selected, showing the repository's purpose and requirements. The requirements include Linux Kernel 5.9+ (for ZNS support), libzmq, nvme-cli, blkzone, libzbd, fio compiled against libzbd, RocksDB, ZenFS, and CTS. The "Setup" section is also visible.

<https://github.com/nicktehrany/ZNS-Study>

# Acknowledgements

- This work is generously supported by Western Digital Donations
- Matias Bjørling, Hans Holmberg, and ZNS team at Western Digital provided helpful comments and feedback