Software-Enabled Flash Storage™ for Hyperscale Data Center

Sponsored by NVM Express organization, the owner of NVMe® specifications
Speakers

Earle Philhower
Senior Marketing Manager

Sean Stead
Senior Staff

KIOXIA

* The Linux Foundation and Software-Enabled Flash Project logo is ™ the Linux Foundation
Agenda

• Development
• Capabilities
• Open Source Platform
• Hardware
• Software
• Join the Project
Well, how did we get here?

Hyperscale needs drove open source development.
Path to Software-Defined Flash

- **2018**: Talking with Customers
- **2019**: C-language Simulator
- **2020**: FPGA Emulation
- **2021**: API Release
- **2022**: Open Source Project Development Hardware + SDK Release
Software-Defined Data Center

Software-Defined Networking
Software-Defined Silicon
Software-Defined Accelerators
Software-Defined Storage

Apply the same idea to flash…
Hyperscale Drives the Development

Flash needs to be Software-Defined

Software-Enabled Flash™ Technology
Making Flash Storage Software-Defined

Remove HDD-based limitations

Optimize for flash memory’s digital nature

Engineered to meet cloud challenges
Software-Enabled Flash™
A Software-Defined Flash API
Sheds the legacy HDD paradigm

Flash can behave in a much more predictable and uniform manner with:

- Data placement and workload isolation
- Latency control with advanced queuing
- Multi-protocol capabilities

Unlock the full power of flash storage
Software-Enabled Flash™

- Isolated tenant environments
- Deliver tiered service levels
- Better flash economics (TCO)
- Faster time to market (TTM)
- Open Source ecosystem
Join, contribute, lead.

The Project is Live

Membership is open

Join the project contribute to the future of flash becoming “Software-Defined”
Governance Model for Software-Enabled Flash

Governing Board
- oversees business decisions, budgets, outreach, marketing/events, trademarks, etc.

Technical Steering Committee
- leads tooling projects and oversees collaboration with upstream

Outreach Committee
- oversees evangelism, communication, outreach, events, training

Project Communities
- deliver tools and standards

Working on best practices for open source projects
https://softwareenabledflash.org

Vendor-neutral collaboration

* The Linux Foundation and Software-Enabled Flash Project logo is ™ the Linux Foundation
Software-Enabled Flash™ Hardware
Built for optimal flash use and cloud-based workloads
Hardware and workload isolation

Data placement control for optimal layout

Dynamic QoS domains for dynamic workloads

Multiple queueing modes for latency outcome control

Copy offload for bandwidth and CPU savings

Flash abstraction for easier migration
The hardware focuses on media management and host offload functionality…

…allowing software to orchestrate and manage protocols, latencies and data placement
Hardware Based Isolation

Ensure contention-free performance
Workload Based Isolation

Software-Enabled Flash™ Controller

QoS Domain 0 (60%)
QoS Domain 2 (40%)
QoS Domain 1 (100%)
QoS Domain 3 (100%)
QoS Domain 4 (50%)
QoS Domain 5 (50%)

Quality of Service (QoS) Domain

Isolate dynamically, per workload
Data Placement Control

Maximize flash lifetime, performance
Dynamic Control for Multi-Tenant Workloads

Orchestrate per-workload QoS Domains
Per-I/O Queueing Modes

Control multi-tenant latency outcomes
In-drive Copy Offload Capabilities

1 Command

3, 14, 159, 2,
65, 35, 979,
3238, 46, 6,
433, 832, 79,
5028, 841, 971,
6939, 93, 7,
510, 5820,
9749, 4459,
2307, 81, 64,
6286, 20899, …

Software-Enabled Flash Unit

Minimizes CPU, DRAM, PCIe® Bandwidth
Flash Abstraction

Accelerate flash technology transition
Software-Enabled Flash™ Software
API and SDK to speed adoption and maximize developer results
API for low-level access

SDK for rapid development

Multiple software-defined protocols
Low-Level API for Device Control

Low-level control without bare metal code
High-Level SDK for Rapid Development

- Reference Flash Translation Layer
- Reference Virtual Drivers
- Performance Test Tool (FIO)
- CLI

Accelerate adoption, testing, development
Multiple Software-Defined Protocols

Inventory of identical SEF Units

Software-Enabled Flash™ API + SDK

Web App
- Block SW Driver

Streaming App
- Custom SW Driver
- Future protocol SW driver

Database App
- ZNS SW Driver

New Application X

Simplify SKU management
Get involved!

Visit our booth!

Join the project softwareenabledflash.org

Make flash Software-Defined!
Definition of capacity: KIOXIA defines a megabyte (MB) as 1,000,000 bytes, a gigabyte (GB) as 1,000,000,000 bytes and a terabyte (TB) as 1,000,000,000,000 bytes. A computer operating system, however, reports storage capacity using powers of 2 for the definition of 1GB = 2^30 = 1,073,741,824 bytes and therefore shows less storage capacity. Available storage capacity (including examples of various media files) will vary based on file size, formatting, settings, software and operating system, such as Microsoft Operating System and/or pre-installed software applications, or media content. Actual formatted capacity may vary.

All company names, product names and service names may be trademarks of their respective companies.

Images are for illustration purposes only.

© 2022 KIOXIA America, Inc. All rights reserved. Information, including product pricing and specifications, content of services, and contact information is current and believed to be accurate on the date of the announcement, but is subject to change without prior notice. Technical and application information contained here is subject to the most recent applicable KIOXIA product specifications.