KIOXIA

Technical Brief

Software-Enabled Flash™ Technology:

As Easy as 1-2-3 for Accommodating Workload Changes, Deploying New Flash Hardware and Optimizing Capabilities in a Software-Defined Platform

Software-Enabled Flash technology delivers a new approach for hyperscale data center storage by unlocking the full power and capabilities of flash memory through an open source, software-defined, multi-protocol-enabled solution. The technology represents an evolution in design where flash storage hardware and open source software work together with dynamic and purpose-built solutions to deliver advanced levels of flash storage functionality. Software-Enabled Flash technology can accommodate varied and changing workloads with new multi-purpose flash storage devices, and has the ability to adapt and change storage protocols in software. These are capabilities desired by cloud applications developers, and for hyperscale and data center developers as their applications will change over time and new workloads will start, increase or decrease in activity, creating a variety of ever changing storage requirements.

Software-Enabled Flash technology can dynamically manage the needs of multiple workloads with software drivers that support various software interface protocols and standard drive form factors (Figure 1) using hardware designed specifically for this technology. This ability to dynamically manage multiple workloads eliminates the need to change, upgrade or purchase new hardware, and is a total cost of ownership (TCO) game changer. Software-Enabled Flash technology can simplify the deployment of data center flash storage hardware up to the very large hyperscale level, enhancing TCO while simplifying inventory management.

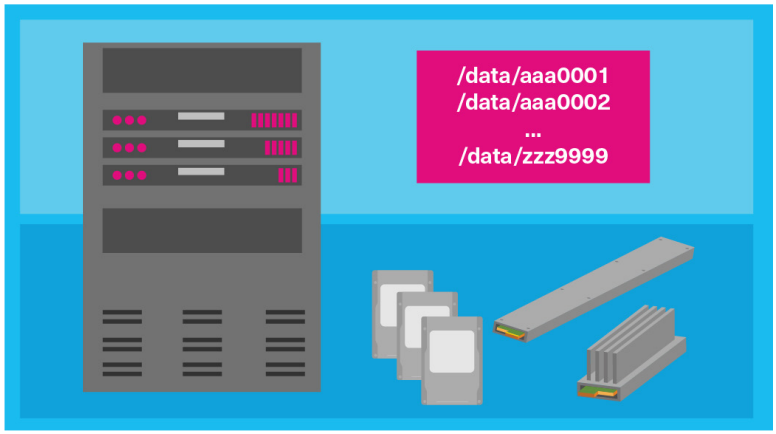

Figure 1: Software-Enabled Flash technology supports multiple software interface protocols and physical drive form factors in large data centers

This tech brief showcases how Software-Enabled Flash technology can easily accommodate workload changes, deploy new flash hardware and optimize flash-specific capabilities in a software-defined platform.

The Powerful Capabilities of Software-Enabled Flash Technology

To begin to understand the powerful capabilities of Software-Enabled Flash technology, developers must first identify the flash storage types within the data center that need to be supported. The technology provides support for all of the standard flash memory types[1] (i.e., SLC, MLC, TLC and QLC) by abstracting those differences from the open source Application Programming Interface (API), as well as future proofing for new generations of flash media. Support for flash memory types enables fast deployments from the data center host side, as well as fast time-to-market as new flash capabilities can be easily deployed with existing software applications.

Software-Enabled Flash technology allows developers to use their own supported test cases via the Software Developers Kit (SDK) or a larger deployment, and quickly verify the technology's adaptability and expandability. Depending on the eventual storage capacity requirements, the compatible Software-Enabled Flash hardware, in combination with open source software available from The Linux Foundation®, can be deployed on the largest hyperscale data centers.

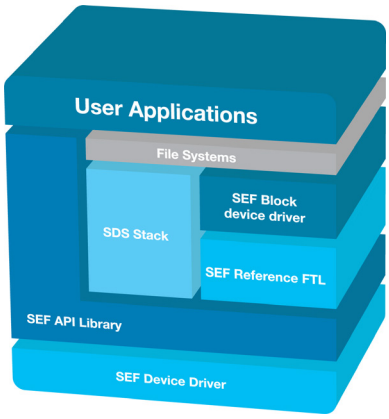

Figure 2: Software-Enabled Flash technology API

Instead of using a lot of different storage devices to match the wide variety of application requirements, which is also represented by many part numbers, developers only need to replicate one hardware part number as many times as needed for any application. If more capacity is required for an application or specific workload, Software-Enabled Flash technology enables similar storage hardware devices to be added to scale up or down as needed. The open source API (Figure 2) enables developers to customize, adapt and change the 'personality' of the hardware through the use of software defined protocols that meet the various application requirements at any given time.

As data center requirements change, Software-Enabled Flash technology avoids the need to replace existing hardware. Instead, developers can simply change the supported software-defined drivers as needed, enabling the existing hardware to perform many different tasks. As such, developers can adapt their flash storage hardware to the workload requirements at hand and improve storage hardware TCO.

Software-Enabled Flash technology abstracts low-level details of flash memory which are vendor and generation specific so existing code can take advantage of the latest flash generations without modifying existing applications. With the Software-Enabled Flash SDK, developers can easily evaluate and test the technology with existing applications without writing new application code.

Run the API to Enable Flash-specific Features

After compatible hardware is installed, the open source API can be started. Developers can choose which features to enable, can run test cases to verify which applications can benefit from software-defined flash storage, and can run the supported drivers ranging from the simple legacy hard drive-based block interfaces to a wide spectrum of options including Zoned Namespaces[2] (ZNS) and others. From low-level flash die control to abstraction, the hardware personalities can be tuned to the needs at hand.

Application and host requirements can be optimized through the Software-Enabled Flash API. It enables the host or the developer with full control over latency outcomes, data placement to manage locality and tenancy, wear-leveling to extend storage media life, and other controls that address performance and Quality of Service (QoS) requirements. The combination of the Software-Enabled Flash API with the SDK offer developers a fast time-to-market with deployment of the right flash memory generation. The software API can be application-specific and modifiable as application needs change enabling flash memory to reach its full potential.

Final Thoughts

Data centers are built to run a variety of workloads. Some applications may require high performance random access - others may need to store data on log-based data files - some may need to implement custom storage layers for specific applications - while others may focus on effective cold data storage. These types of changes are handled at the software API and driver levels via Software-Enabled Flash technology. As applications change or other storage priorities arise, the requirements can be met in real time enabling the compatible Software-Enabled Flash hardware to remain in place and untouched. Therefore, every Software-Enabled flash compatible device in the data center can be improved through the software API, positively impacting TCO and inventory management.

Get Involved

Software-Enabled Flash technology is an open source community project that seeks to unify different flash capabilities by providing an application-centric method of interfacing with storage. Developers interested in this Linux Foundation community project can review and request details about membership for contribution to a collaborative solution at the Software-Enabled Flash technology home page.

https://softwareenabledflash.org

KIOXIA, member of The Linux Foundation, released an API definition and specification document to the open source software (OSS) community that is downloadable from the KIOXIA repositories on the GitHub® site. The site provides solutions that can help developers, architects and end users solve their unique flash storage challenges. KIOXIA also provides updates to the Software-Enabled Flash technology home page as a Linux Foundation individual supporter.

NOTES:

[1] Flash memory types include Single-Level Cell (SLC) which stores 1-bit of data per flash memory cell; Multi-Level Cell (MLC) which stores 2-bits of data per flash memory cell; Triple-Level Cell (TLC) which stores 3-bits of data per flash memory cell; and Quad-Level Cell (QLC) which stores 4-bits of data per flash memory cell.

[2] Zoned Namespaces (ZNS) are a new command set in the NVMe® specification that exposes a zoned block storage interface between the host and the SSD, enabling the SSD to perfectly align the data to its media.

TRADEMARKS:

DISCLAIMERS: