



Introducing Software-Enabled Flash™ (SEF) Technology

Innovative, software-driven storage enabling developers with unprecedented access and control over flash memory, latency and data placement.

Eric Ries Scott Stetzer Scott Harlin

Hyperscale is Redefining Solid-State Drives

Hyperscale applications are major consumers of flash memory for storage today, and it is expected to be the case in the foreseeable future. Cloud service providers are massively expanding the scale of their services to supply an ever-growing customer demand resulting in top tier providers defining their own system architectures in order to scale quickly and efficiently. The single greatest motivating factor behind this movement to define unique system architectures is efficiency at scale. As the name 'hyperscale' implies, cloud providers run their systems at hyper-scale. Given these advanced levels of complexity and scale, small system inefficiencies can undergo a multiplying effect that must be addressed in order to achieve service and business stability.

Critical Requirements for Hyperscale Deployments

The critical factors of cloud storage at this heightened scale include response time, data protection and data resiliency, which can also adversely affect Quality of Service (QoS). For hyperscale cloud service providers, performance predictability as well as generating uniformity in the behavior of their platforms are important characteristics. Solid state storage, due to its digital nature, non-volatility and ability to easily scale-out, is quickly becoming the dominant form of active storage used by hyperscale cloud service providers.

The problem with this model is that solid-state storage (driven by flash memory) and hard drives read and write data in a fundamentally different manner, however, operating systems want to treat them the same. Solid state-drives (SSDs) were invented to make flash memory appear and behave like hard-drives to the system and for easy customer adoption. At a limited scale, SSDs behaving like HDDs may work without noticeable disruptions to system performance or QoS, but at the hyperscale level, the disruptions can be significant.

Redefining SSD Use

To obtain the levels of efficiency demanded by the business, hyperscale cloud-service providers are pursuing methods to redefine how they can and should use solid-state storage, in particular, SSDs. Each provider is scaling their systems in a manner that best takes advantage of their particular strengths, defining their own unique system architectures and approaches that make storage more efficient. These newly defined architectures have become the driving force behind the need for a different type of digital storage – one that is software-driven and enables the host to enforce predictable and uniform outcomes from its flash-based storage.

Once the outcomes are determined, the host must be able to manage behaviors, such as latency and data placement. Traditionally, SSDs do not allow the host to manage or control these behaviors sufficiently. To address this, and other issues, KIOXIA is presenting a new way for hyperscale users to directly control and manage flash devices through a software-driven Application Programming Interface (API). This API enables hyperscale service developers to control the behavior of software-enabled flash™ storage from their host applications.

Enter Software-Enabled Flash™ Technology

Software-Enabled Flash (SEF) technology can provide hyperscale users with the ability to maximize the value and efficiency of their solid-state storage by enabling significantly enhanced host control. SEF solutions can relieve the developer of the need to manage the flash media as well as abstracting generational differences in media behavior.

SEF Features and Benefits

SEF technology enables system optimizations that are not available from traditional storage approaches. It gives hyperscale architects the ability to manage and define the behavior of their storage by providing the following features:

- Latency control over impacting algorithms, such as garbage collection and wear-leveling
- Host control over data placement and isolation
- Host control of workload isolation via system definable virtual devices and QoS domain(s)
- Physical address control over read processing (eliminates latency)
- Abstraction of flash memory generational and vendor differences to simplify adopting each new generation of flash memory

Features related to SEF hardware units can include the following:

- The flexibility in hardware to enable the use, reduction or elimination of device-side DRAM. For those deployments where host memory is preferred over device DRAM, the SEF API offers a Unified Write Buffer (UWB) capability that enables users to:
 - Reduce or completely eliminate DRAM usage in favor of host memory usage
 - Keep the conventional architecture that requires DRAM to be located at the drive side
- A powerful 'nameless write' function that provides a flexible update of the L2P table and system-level optimization, enabling write data to be placed in the most optimized method for flash memory cell health and longevity

Features of SEF technology can deliver the following benefits:

- Host-controlled and software-programmable management of flash storage
- Open-source API enables developers full access to flash memory while maintaining their own code that includes the flash translation layer (FTL)
- Timelier and easier deployment of the latest generations of flash memory where application development is isolated from generational differences in flash memory negating the need to modify applications, software or source code for every new generation
- Tools to design and optimize efficient flash memory utilization
- Application development is isolated from generational differences in flash memory negating the need to modify applications, software or source code for every new generation of flash memory
- Hardware units that support PCIe® standards
- Ability for flash memory to place all necessary flash management techniques into hardware behind the API, enabling hyperscale developers to deliver excellent performance and endurance

SEF Concepts and Technology Overview

SEF technology is an innovative, architectural flash-based storage hardware platform that is driven by software. As software redefines the world, hyperscale and cloud applications are redefining the way every person works, plays and communicates. SEF is being developed for hyperscale developers and innovators who need direct access and host/application-level control over storage, data placement and data access (latency).

The technology utilizes a software API that abstracts flash-based functions and enables host control over such SSD behaviors as latency management, data placement, tenant isolation and garbage collection. The API is open-sourced and designed to enable software developers with the flexibility of adapting, developing, customizing or tuning flash memory to any application or storage need through software programming, and easily accessible to cloud applications, storage and developer communities.

The SEF hardware unit is architected to combine the most recent flash memory generation with a small onboard SoC controller that resides on a PCB module. As an option, the SEF architecture supports an on-device DRAM controller allowing the module to be populated with DRAM, based upon the needs of each hyperscale user. This combination of components comprise a SEF unit that is designed to deliver flash-based storage across a PCIe connection (Figure 1).

Behind the interface, individual SEF units handle all aspects of block and page programming (such as timing, ECC and endurance) for any type or generation of flash memory being used. SEF units also handle low-level read tasks that include error correction, flash memory cell health and life extension algorithms.

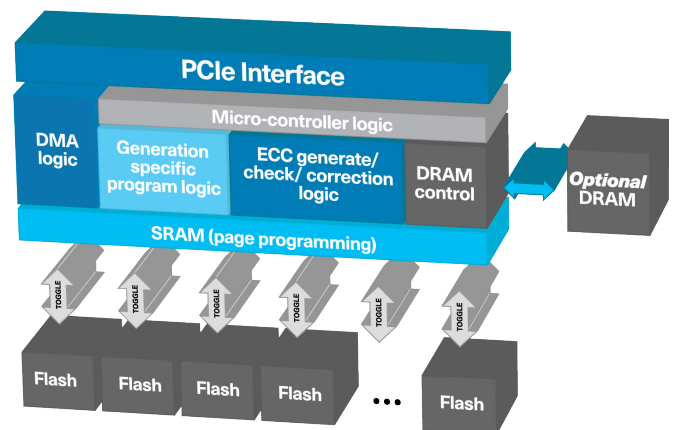


Figure 1: the SEF unit is designed to deliver flash-based storage across a PCIe connection

Components of SEF Technology

A SEF unit is the actual hardware portion for deploying flash memory through software. The hardware implements the API abstraction to the flash memory and is designed to:

- Manage the flash media and abstract generational differences of flash memory
- Control and perform physical page programming inside of the flash memory block and page
- Map around defects, returning actual physical addresses to enable direct read operations
- Fetch data from the on-board controller or host memory via direct memory access to enable page programming
- Manage life time and health of the flash memory
- Implement ECC logic and die interleaving
- Implement QoS domains
- Implement command scheduling across queues

The SEF unit also delivers advanced scheduling functionality that provides developers with a flexible mechanism for implementing separate prioritized queues used for read, write, copy and erase operations. This capability, in combination with die time scheduling features, enables weighted fair queuing (WFQ) and command prioritization in hardware that is accessible from the API.

SEF technology is driven by the API which defines the commands and capabilities of the hardware for use and implementation by host applications. The API (Figure 2) provides abstracted commands for interfacing to flash memory, such as:

- Host-controlled commands for SSD maintenance and housekeeping functions
- ‘Nameless’ write commands provide flexible updates of the L2P table and system level optimization
- Commands for hardware and software-based tenant isolation
- Commands that address all storage and QoS functions

Additional benefits of the API include the ability to:

- Enable flash-aware applications with direct control over flash memory
- Abstract inter-generational flash memory differences
- Simplify the interaction between the host and flash memory for software developers

Capabilities of SEF Technology

The small SEF onboard microcontroller that resides on the PCB module is responsible for managing flash-based media. It abstracts and controls generational differences in flash memory relating to page sizes, endurance control and the way that flash dies are programmed. Through the software API, new generations of flash memory can be deployed quickly, cost-effectively and efficiently, providing developers with full control over data placement, latency, storage management, data recovery, data refreshing and data persistence.

Latency variances are a significant challenge in hyperscale environments relating to QoS. The SEF software approach provides developers with full control over latency management at the host level, enabling them to define and schedule around any tasks that can impact data access. The API provides the host functions that can pause, interrupt or defer an action until latency-critical I/O is processed and completed.

Data placement and workload isolation are major issues facing hyperscale applications that cause limited control and management over data. In this scenario, programmers and architects are provided with SEF tools that define and physically organize the flash memory down to the die level. The API assigns the die and channel configurations into one or more ‘virtual devices’ (Figure 3) that enable applications to define which die to use across one or more specific channels. Users can flexibly set up these virtual devices into zones, creating hardware isolation that eliminates impact of I/O from one domain to another.

QoS domains are defined by capacity inside an isolated virtual device and are deployed as the actual user space within SEF units (Figure 4). This virtual device can have one or more QoS domains defined since they are independently managed by software applications. This also enables the implementation of flash translation layers and garbage collection algorithms that operate independently from other QoS zones nearby, thus creating isolation for different QoS domains at the application level.

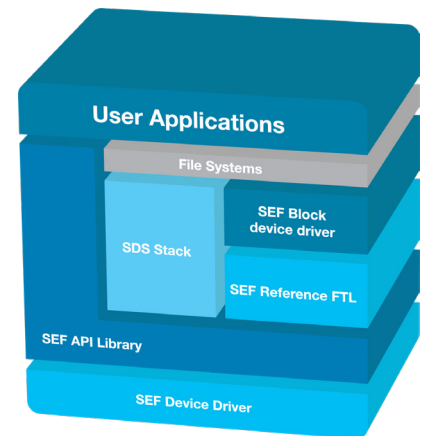


Figure 2: SEF host software block diagram

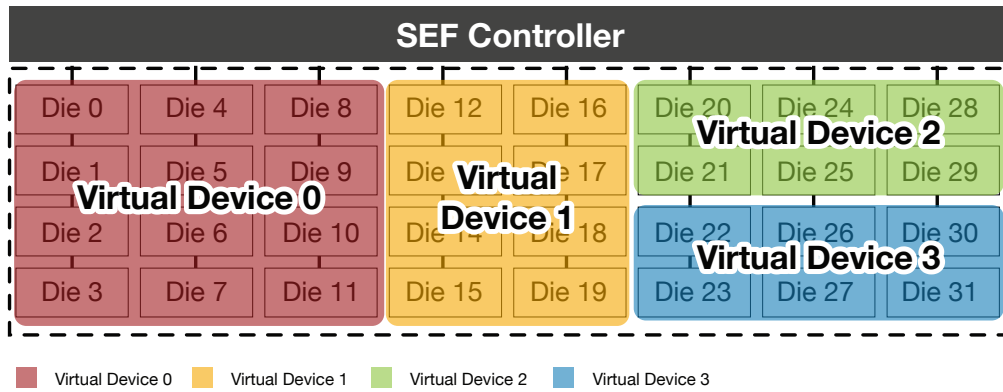


Figure 3: The API assigns dies and channels into virtual devices

Software-enabled host control is the mechanism by which all of these capabilities are exposed and delivered. The main technical benefits of SEF technology are in its ability to deliver a range of capabilities that can be customized to the needs of each application, system or data center, while being able to adapt or modify those capabilities as needed. Thus, the hyperscale cloud service provider can generate greater operational value from the flash memory they consume.

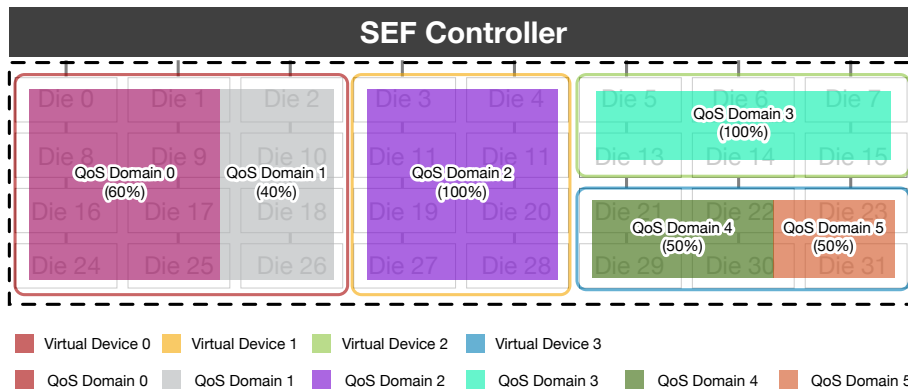


Figure 4: QoS domains are used as user space for SEF units

SEF technology could be accompanied with tools that include sample source code similar to a software development kit (SDK). This source code would demonstrate how flash memory could be deployed via the host, and then redeployed with different personalities. When available, such tools could include a traditional sample block driver that would enable a SEF implementation to appear like an SSD on a system.

Conclusions

Software-enabled flash technology provides a big step forward for hyperscale architects and developers as they progress with their redefinition of solid-state storage. The SEF open-source API will provide full software control over the storage hardware, even down to the flash die level if desired or needed. Media definition and control enables the user to create, define, configure and isolate flash memory dies and channels to meet application performance requirements. Through optimized parallelism, SEF-based storage can scale across thousands of SSDs. Tenant isolation of workloads is fully definable to the programmer and is configurable by system architects, enabling them to achieve or exceed the QoS requirements.

Programmatic control of the flash-based storage media offers the host and application total control over storage latency. This provides hyperscale users the ability to control any aspect of access, enabling them to change or delay housekeeping or firmware-related tasks so they do not interfere with I/O access responses. DRAM can be optionally deployed on the device giving hyperscale users the option of selectively and programmatically offsetting some housekeeping functions to the storage device.

Finally, SEF technology will enable cloud data center developers to adapt each new generation of flash memory rapidly, cost-effectively and efficiently. It will make flash memory easier to manipulate and use, enabling hyperscale users to implement the functions and capabilities they want, with a solution that can look, act and perform like an SSD, but is enabled, defined and managed through a software API.

The future of flash storage is software-enabled.



Notes:

PCIe is a registered trademark of PCI-SIG. All other trademarks or registered trademarks are the property of their respective owners.

Disclaimer:

KIOXIA America, Inc. may make changes to specifications and product descriptions at any time. The information presented in this white paper is for informational purposes only and may contain technical inaccuracies, omissions and typographical errors. Any differences in software or hardware configuration may affect actual performance. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to any changes in product, component and hardware revision changes, new model and/or product releases, software changes, firmware changes, or the like. KIOXIA America assumes no obligation to update or otherwise correct or revise this information.

KIOXIA America makes no representations or warranties with respect to the contents herein and assumes no responsibility for any inaccuracies, errors or omissions that may appear in this information.

KIOXIA America specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. In no event will KIOXIA America be liable to any person for any direct, indirect, special or other consequential damages arising from the use of any information contained herein, even if KIOXIA America is expressly advised of the possibility of such damages.