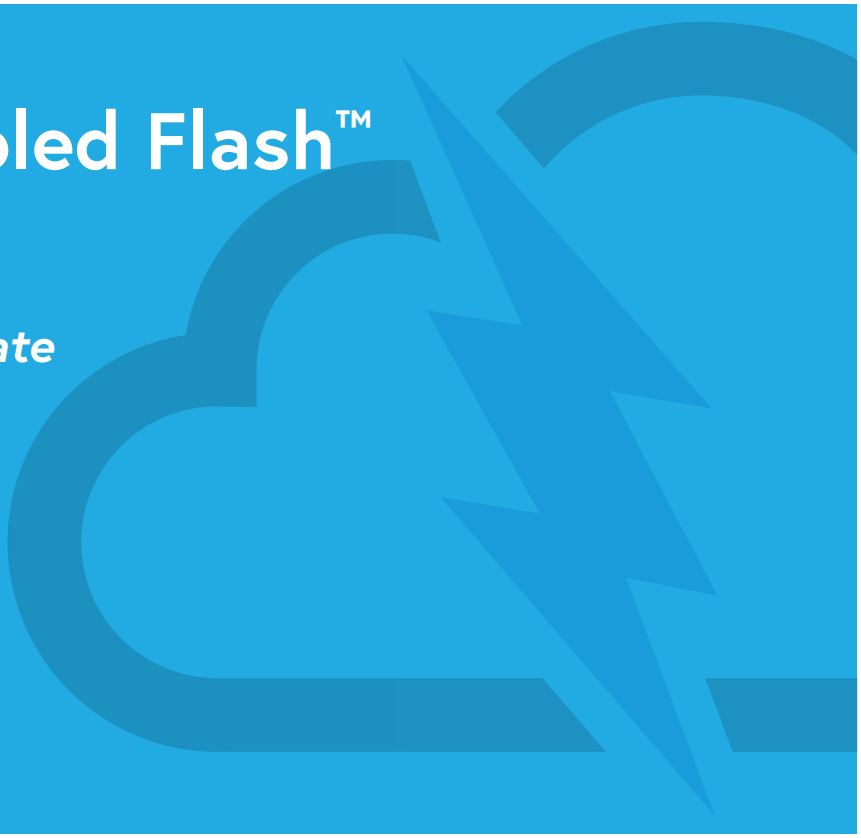
SEF

SOFTWARE-ENABLED FLASH™

Software-Enabled Flash™ Technology:

'How-to' Guide to Evaluate SEF in a Data Center



TECHNICAL BRIEF

Software-Enabled Flash (SEF) technology is a new and powerful way of deploying and using flash memory in a data center. It combines purpose-built hardware with an open source software layer that runs on the host. Targeted to storage, data center and hyperscale developers, SEF provides hardware and software tools that can turn flash memory into a software-defined data center resource. Developers are afforded low-level access to flash memory, as well as direct control over flash storage device management tasks. As such, SEF can deliver data center improvements in Quality of Service (QoS), latency performance and flash memory value, to name a few. The technology can maximize flash memory value in a variety of use cases, both at the single server level and across cloud infrastructures, with advanced capabilities not available from traditional SSD storage.

As an example, background flash memory operations, such as data placement and garbage collection (GC), can be optimized by developers through the SEF software-defined API. It manages how and when tasks start, run and stop, and abstracts low-level background functions.

This technical brief presents further examples on how developers can program, use and evaluate the effectiveness of SEF technology firsthand, to bring key flash capabilities to their data centers.

SEF Capabilities

Through the SEF Software Development Kit (SDK), scheduled for availability in 2023, the capabilities addressed through host control, with working source code examples, includes:

1. *Hardware Efficiency and Die Time Control*
2. *Hardware Isolation in Virtual Devices*
3. *Software Isolation in Quality of Service Domains*
4. *Control over Latency Outcomes*

SEF provides flash storage control at scale and under control of the developer at the host layer. It also works with commercially-available test tools to evaluate these capabilities from a performance perspective. A ported Flexible I/O[1] (FIO) utility is included in the SEF SDK that can run performance benchmarks of specific SEF capabilities. For this tech brief, FIO test tools were used to show the desired performance results for the four bulleted SEF capabilities above. When FIO is used with SEF, developers can control, extract, demonstrate and evaluate valuable and advanced flash memory capabilities.

This technical brief is a 'how-to' guide to best showcase important and powerful SEF capabilities. Additionally, source code for SEF is hosted on this GitHub® site - https://github.com/softwareenabledflash.

Hardware Efficiency and Die Time Control

One effective way in which SEF delivers hardware efficiencies in large-scale data center storage is through its very powerful die time weighted fair queueing capability. Die time control enables individual Input/Output (I/O) weights to be finely-tuned so they can account for their on-die time. By using the advanced die time weighted fair queueing available to applications, individual I/Os from multiple applications can be prioritized (Figure 1). Since this queueing mode is based on die time (the amount of time an individual I/O operation requires on a flash die), it can provide high priority applications with data quickly while avoiding I/O starvation for lower priority tasks.

Figure 1: SEF features die time weighted fair queues that help high priority I/O avoid bottlenecks
(Used with permission from KIOXIA America, Inc.)

Performance Measuring Using FIO Tools

When FIO tools are used to run performance tests, the following results are desired:

Test Condition	Desired Result
When only read operations are turned on:	*The SEF drive is expected to approach a theoretical maximum of both the read bandwidth and I/O Operations Per Second (IOPS).*
When only write operations are turned on:	*The SEF drive is expected to approach a theoretical maximum of both the write bandwidth and IOPS.*
When both read and write operations are turned on:	*The performance of both should proportionally match the theoretical read/write bandwidth and IOPS where the read/write die time options have been set to*

The read/write die time options enable the read and write die time priorities to be modified in real time. The options include a 100% read intensive workload with 0% write operations, a 100% write intensive workload with 0% read operations, and any variation in between. With FIO, the throughput will update as the read/write options change. A common option is 70%, representing a 70% read / 30% write mixed workload.

Test Setup within SEF

The setup and interactive controls for running the hardware efficiency and die time control includes:

Hardware Efficiency and Die Time Control	Description
Setup	*Use one Virtual Device that spans across all dies* *Use one Quality of Service Domain for the Virtual Device.*
Interactive Controls	*Use 'Read on/off,' 'Write on/off,' and the Virtual Device read/write die time options.* *The 'Read on/off' command controls an FIO job with an unbounded sequential read workload of a given read size and queue depth.* *The 'Write on/off' controls an FIO job with an unbounded sequential write workload of a given write size and queue depth.* *The I/O size and queue depths of these jobs can be selected by developers to achieve optimal performance.* *Use the read/write die time options to enable results from various workload ratios.*

In addition to die time weighted fair queuing, SEF includes priority queuing and round robin queuing (Figure 2).

Priority queuing orders workloads strictly by importance, enabling higher priority workloads (i.e., mission-critical databases) to effectively monopolize the SEF drive. The SEF drive will allow lower priority workloads to access it only when it is not busy.

Round robin queuing enables all workloads to effectively time-slice access to the SEF drive. In this model, every workload is allowed access to the drive in turn.

Once an operation is transmitted to the SEF drive, the read and write operations are split up and processed in separate, parallel paths. Separating read and write operations per flash die enables SEF drives to minimize the impact of head-of-queue blocking. This occurs when a lower priority, long running operation, such as a flash write or erase, blocks other higher priority operations from being processed.

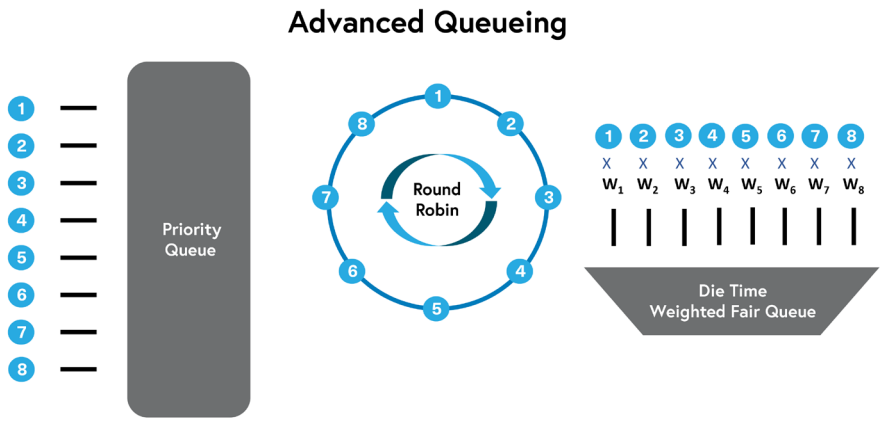


Figure 2: SEF includes three choices for data queuing
(Used with permission from KIOXIA America, Inc.)

Hardware Isolation in Virtual Devices

In a multi-tenant server when tenants share space on the flash die, a large read or write operation from one tenant could block other tenants from progressing with their respective operations, and a potential hit to latency performance. Developers can avoid this latency concern, as well as other potential cross-tenant interferences, by using Virtual Devices. These devices enable sets of flash dies to be assigned to specific workloads, effectively isolating them from other workloads and associated interferences (Figure 3).

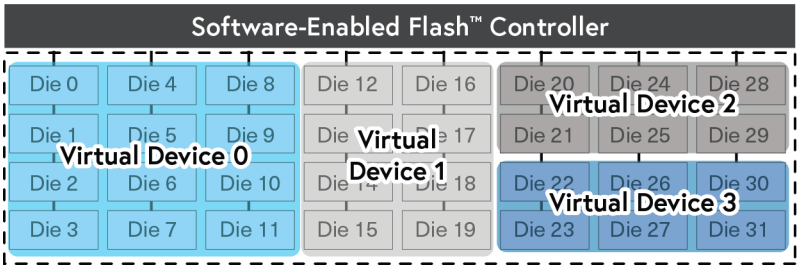


Figure 3: Image representation of the Virtual Device capability in SEF
(Used with permission from KIOXIA America, Inc.)

To demonstrate one approach in evaluating hardware isolation, two Virtual Devices are required – one with a small number of dies and one with a large number of dies. For each Virtual Device, one SEF Quality of Service Domain is required. Though Virtual Devices are a powerful way to isolate workloads, a finer grain of isolation control can be achieved through Quality of Service Domains that further subdivide Virtual Devices and impose a secondary level of isolation between different workloads (Figure 4). While workloads in different Quality of Service Domains may utilize the same flash dies, their data is never intermingled within a flash super block[2]. Many Quality of Service Domains can be generated within a single Virtual Device simultaneously, allowing for many more simultaneous workloads than flash dies in a system.

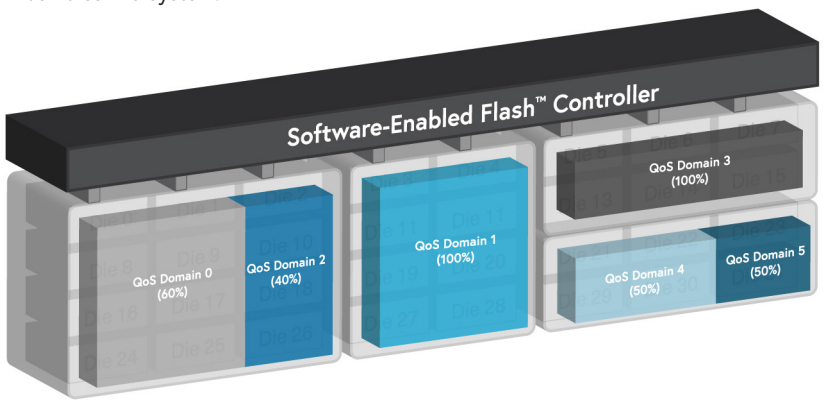


Figure 4: Image representation of the Quality of Service Domain capability in SEF technology
(Used with permission from KIOXIA America, Inc.)

Performance Measuring Using FIO Tools

When FIO tools are used to run performance tests, the following results are desired:

Desired Results
The performance of a Quality of Service Domain should be proportional to the number of flash dies in the respective Virtual Device.
The performance of each Quality of Service Domain should not be affected whether other domains are turned on or off.
When the read/write die time options are moved for each Quality of Service Domain, only that domain should be affected.
The read and write performance for each Quality of Service Domain should proportionally match the theoretical read/write bandwidth and IOPS where the read/write die time options have been set to. One suggested way to achieve this is to initially set the read/write die time options to 50% as 50% read / 50% write, representing a very common mixed workload.

Test Setup within SEF

The setup and interactive controls for running hardware isolation in Virtual Devices includes:

Hardware Isolation in Virtual Devices	Description
Setup	*Use two Virtual Devices - one with a small number of flash dies – one with a large number of flash dies.* *Use one Quality of Service Domain for each Virtual Device.*
Interactive Controls	*Use 'Domain 1 on/off,' 'Domain 2 on/off,' Domain 1 read/write die time options, and Domain 2 read/write die time options.* *The 'Domain 1 on/off' command controls an FIO job with both an unbounded sequential read and write workload of a given read and write size and queue depth.* *The 'Domain 2 on/off' command controls an FIO job with both an unbounded sequential read and write workload of a given read and write size and queue depth.* *The I/O size and queue depths of these jobs can be selected by developers to achieve optimal performance.* *Use the read/write die time options to enable results from various workload ratios.*

Software Isolation in Quality of Service Domains

As discussed in the Hardware Efficiency and Die Time Control section, SEF features an advanced queueing design that isolates workload requests as they are received by a SEF drive. Some of this data queueing is under application or orchestration software control via modifiable I/O operation weights, and some is managed directly by the internal architecture of the SEF drive. Multiple parallel queues within the SEF drive separate workload operations down to the flash die level. Once an operation is transmitted to the SEF drive via standard NVMe® protocol submission queues, the following occurs:

1. *Read and write operations are split up and processed in separate, parallel paths;*
2. *The software-defined weights and queueing models are applied independently, and in parallel;*
3. *The separated read and write streams are assigned to per-flash-die queues.*

By separating the read and write operations per flash die, the technology enables SEF drives to minimize the impact of head-of-queue blocking previously discussed. For this example, one Virtual Device is required with a large number of dies, as well as two Quality of Service Domains for the Virtual Device.

Performance Measuring Using FIO Tools

When FIO tools are used to run performance tests, the following results are desired:

Desired Results
If only one Quality of Service Domain is enabled within the Virtual Device, its performance should be proportional to the maximum theoretical read/write bandwidth and IOPS where the read/write die time options have been set to.
If both Quality of Service Domains are enabled, their respective performances should show 50% of the maximum theoretical read/write bandwidth and IOPS where the read/write die time options have been set to.
Changes to the read/write die time options can affect both Quality of Service Domains while changes to the domain priority options should be proportional to the read/write bandwidth and IOPS performance of the individual Quality of Service Domains.
The read and write performance for each Quality of Service Domain should proportionally match the theoretical read/write bandwidth and IOPS where the read/write die time options have been set to. One suggested way to achieve this is to initially set the read/write die time options to 50% as 50% read / 50% write, representing a very common mixed workload.

Test Setup within SEF

The setup and interactive controls for running the software isolation in Quality of Service Domains includes:

Hardware Isolation in Virtual Devices	Description
Setup	*Use one Virtual Device with a large number of flash dies.* *Use two Quality of Service Domains for the Virtual Device.*
Interactive Controls	*Use 'Domain 1 on/off,' 'Domain 2 on/off,' Virtual Device read/write die time options, and domain priority options.* *The 'Domain 1 on/off' command controls an FIO job with both an unbounded sequential read and write workload of a given read and write size and queue depth.* *The 'Domain 2 on/off' command controls an FIO job with both an unbounded sequential read and write workload of a given read and write size and queue depth.* *The I/O size and queue depths of these jobs can be selected by developers to achieve optimal performance.* *Use the read/write die time options to enable results from various workload ratios.*

Control over Latency Outcomes

SEF can provide developers with control over latency outcomes which is particularly useful to hyperscalers for determining that their QoS application objectives are being met. Some applications benefit best from the lowest possible latency outcomes, dismissing any long latency outliers. Other applications may prefer a smaller, more predictable distribution of latency outcomes over raw speed. As needs change, developers can use SEF to dynamically modify the latency parameters to best suit the application requirements at that time.

Cloud providers can also benefit from latency control by delivering different service tiers. Performance-minded customers can be configured to receive lower latency I/O operations, while budget-conscious customers can be assigned a lower QoS level.

For this assessment, one Virtual Device is required with a large number of dies, as well as one Quality of Service Domain for the Virtual Device.

Performance Measuring Using FIO Tools

When FIO tools are used to run performance tests, the following results are desired:

Desired Results
With write operations disabled, a performance baseline can be established for read latency that accounts for die collision due to random access operations.
Increasing or decreasing latency can be controlled and performance test tools (such as FIO) will show the changes.
With write operations enabled, the SUSPEND/RESUME controls can be used to shape latency response times for a given setting.
The priorities associated with read and write operations can be controlled/changed for desired latency outcomes.

Test Setup within SEF

The interactive controls for controlling latency outcomes in Virtual Devices includes:

Control over Latency Outcomes	Description
Setup	*Use one Virtual Device with a large number of flash dies.* *Use one Quality of Service Domain for the Virtual Device.*
Interactive Controls	*Use 'Read on/off,' 'Write on/off,' Virtual Device read/write die time options, and appropriate controls for SUSPEND/RESUME.* *The 'Read on/off' command controls an FIO job with both an unbounded random read workload of a given read size and queue depth.* *The 'Write on/off' command controls an FIO job with both an unbounded random write workload of a given write size and queue depth.* *The I/O size and queue depths of these jobs can be selected by developers to achieve optimal performance.* *Use the read/write die time options to enable results from various workload ratios.* *The SUSPEND/RESUME controls should be initially set to disable SUSPEND.*

Summary

Software-Enabled Flash technology is a powerful new tool for developers to access low level flash details and extract maximum value from flash storage. The SDK, when available in 2023, will show ways that developers can demonstrate, evaluate and experience SEF while showcasing the improved flash storage control that is delivered at scale repeatedly. Using common test tools, such as FIO software, developers can control and extract additional performance value from their data center flash storage. As an open source project under The Linux Foundation®, SEF is designed for ALL developers to cooperatively work together and produce best-in-class flash storage technology and data management.

This technical brief presents ways to showcase the value and effectiveness for such SEF capabilities as hardware efficiency and die time control, hardware isolation in Virtual Devices, software isolation in a Quality of Service Domain, and control over latency outcomes. Through the SEF SDK, the ecosystem of storage developers, industry companies, and flash memory/SSD manufacturers can better understand the many new and important capabilities such as hardware/software workload isolation, data placement control for optimal layouts, modifiable Quality of Service Domains for ever-changing workloads and multiple data queueing models.

For more information on Software-Enabled Flash technology and the SDK, go to https://softwareenabledflash.org where whitepapers, videos and infographics are available.

Notes:

[1] Flexible I/O (FIO) is a free and open source disk I/O tool used both for benchmark and stress/hardware verification. The software displays a variety of I/O performance results, including complete I/O latencies and percentiles.

[2] A flash super block is a group of flash blocks from flash dies written in parallel for performance reasons.