SEF

SOFTWARE-ENABLED FLASH™

Technical Steering Committee Meeting

December 14, 2023

Agenda

- Welcome and quick review of the previous TSC meeting
  - SEF Command Set release
- SEF (Re-)introduction
- SEF SDK release to GitHub
  - Software
  - Patches
  - **User Guide**
- Member PoC1 hardware with SEF support
- Q & A discussions

Software-Enabled Flash (Re-)Introduction

The next evolution of flash is Software-Defined

- Fine-grained data placement
- Workload isolation
- Write amplification reduction
- Latency outcome control
- Advanced queueing methods
- Die-Time I/O prioritization
- Customized protocols
- Open source API and SDK

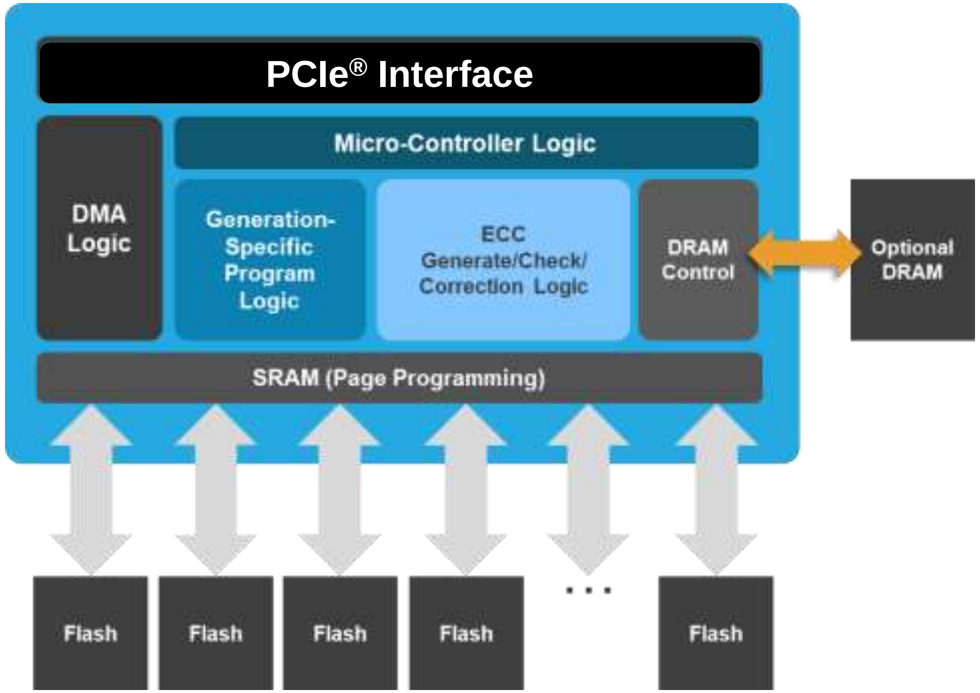
Control to the Storage Developer

- Complete physical isolation **control**
  - Software isolation layered on top
- Data placement **control**
  - Including instant-reclaim
- Write Amplification Factor (WAF) **control**
- Latency outcome **control**
- Housekeeping acceleration and **control**
- Software-defined protocol **control**
  - Block, FDP, ZNS, etc.

Software-Enabled Flash (Re-)Introduction

Vendor-Neutral Hardware Features

Vendor-Agnostic Hardware Design



Graphic used with permission of KIOXIA

Vendor Configurable

- Flash Technology
- Flash Interface Protocol
- DRAM Buffer vs. Host Buffer vs. Mixed
- Form Factor
- Power Limits

PCIe is a registered trademark of PCI-SIG.

Targeted Hardware Features for SEF Support

Advanced queueing control

Control latencies at the flash operation level

Flash abstraction & management

Simplify porting between flash generations, vendors, and technologies

Low-level hardware partitioning & isolation

Maximum performance decoupling between critical workloads

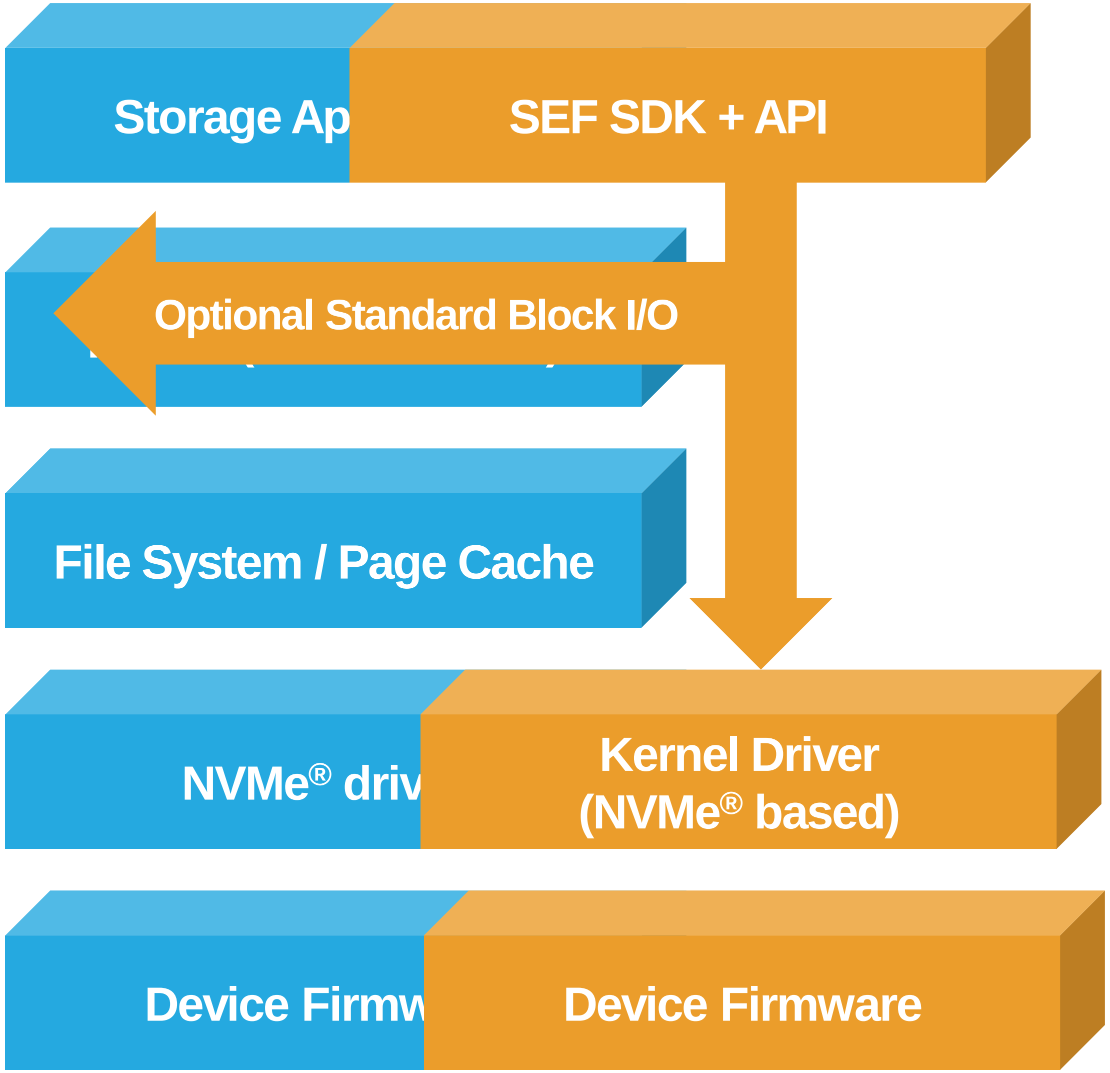
Advanced on-board copy offload

Minimize CPU and bus management for data movement operations

Software-Enabled Flash (Re-)Introduction

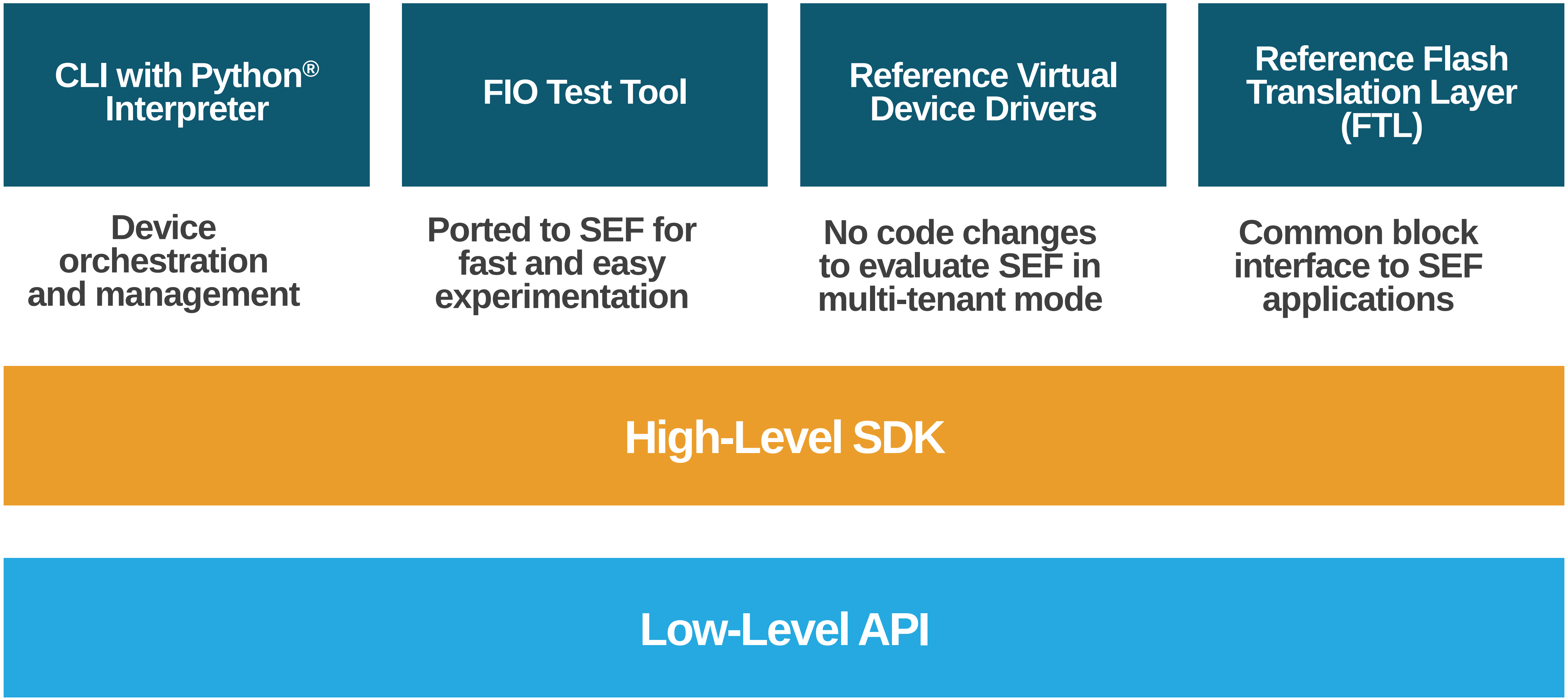
Open Source Software Stack

Standard Block vs. SEF Application



* NVMe is a registered trademark and/or service mark of the PCI-SIG

Open Source API and SDK



CLI with Python® Interpreter

Device orchestration and management

FIO Test Tool

Ported to SEF for fast and easy experimentation

Reference Virtual Device Drivers

No code changes to evaluate SEF in multi-tenant mode

Reference Flash Translation Layer (FTL)

Common block interface to SEF applications

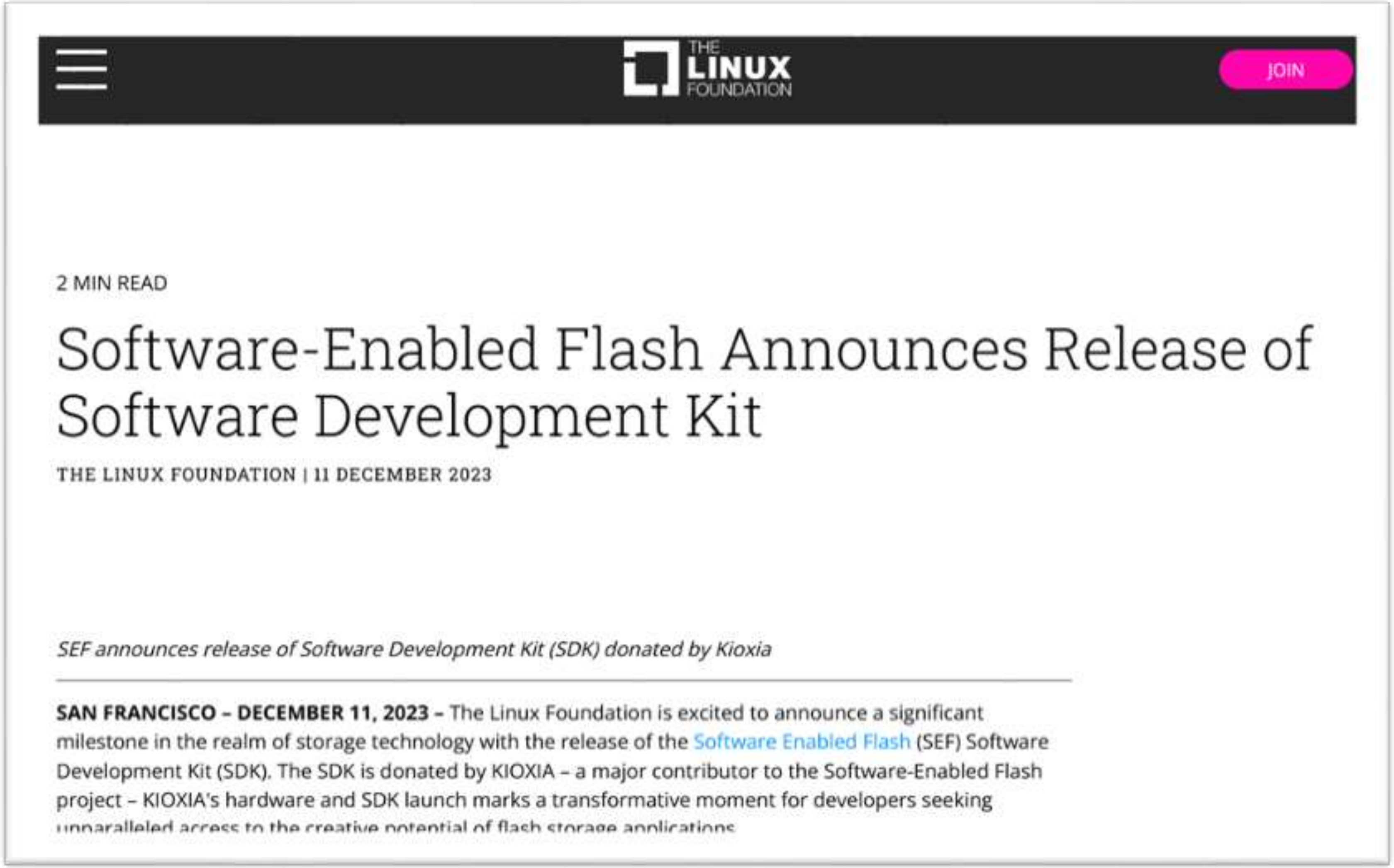
High-Level SDK

Low-Level API

* Python is a registered trademark of the Python Software Foundation.

SDK Release

Software Development Kit (SDK) donated by KIOXIA



2 MIN READ

Software-Enabled Flash Announces Release of Software Development Kit

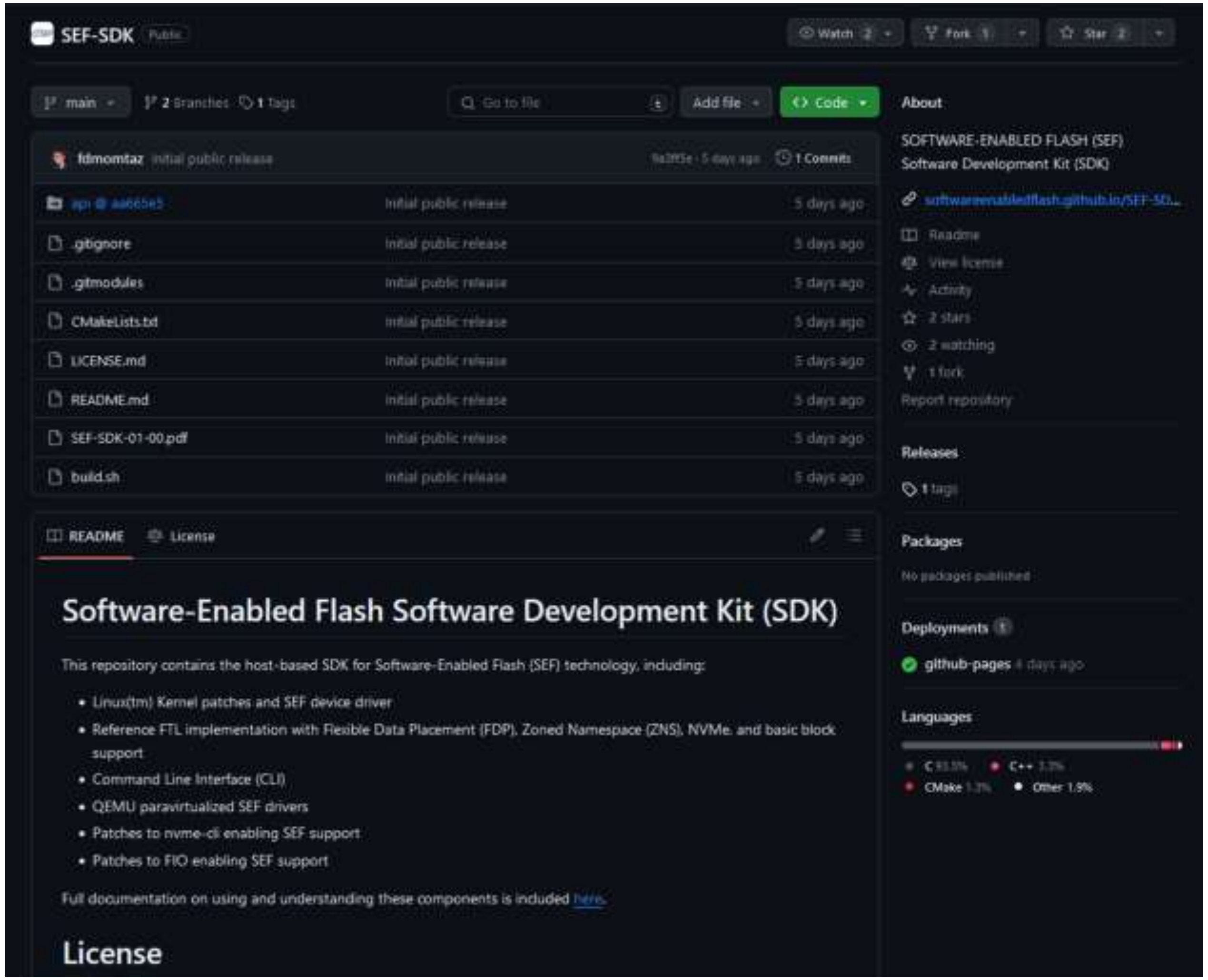
THE LINUX FOUNDATION | 11 DECEMBER 2023

SEF announces release of Software Development Kit (SDK) donated by Kioxia

SAN FRANCISCO – DECEMBER 11, 2023 – The Linux Foundation is excited to announce a significant milestone in the realm of storage technology with the release of the Software Enabled Flash (SEF) Software Development Kit (SDK). The SDK is donated by KIOXIA – a major contributor to the Software-Enabled Flash project – KIOXIA's hardware and SDK launch marks a transformative moment for developers seeking unparalleled access to the creative potential of flash storage applications

https://www.linuxfoundation.org/press/software-enabled-flash-announces-software-development-kit-sdk

Getting the SDK - https://github.com/SoftwareEnabledFlash/SEF-SDK



SEF-SDK Public

fdmomtaz initial public release 9a2ff5e · 5 days ago 1 Commits

api @ aa66be5	Initial public release	5 days ago
.gitignore	Initial public release	5 days ago
.gitmodules	Initial public release	5 days ago
CMakeLists.txt	Initial public release	5 days ago
LICENSE.md	Initial public release	5 days ago
README.md	Initial public release	5 days ago
SEF-SDK-01-00.pdf	Initial public release	5 days ago
build.sh	Initial public release	5 days ago

Software-Enabled Flash Software Development Kit (SDK)

This repository contains the host-based SDK for Software-Enabled Flash (SEF) technology, including:

- Linux(tm) Kernel patches and SEF device driver
- Reference FTL implementation with Flexible Data Placement (FDP), Zoned Namespace (ZNS), NVMe, and basic block support
- Command Line Interface (CLI)
- QEMU paravirtualized SEF drivers
- Patches to nvme-cli enabling SEF support
- Patches to FIO enabling SEF support

Full documentation on using and understanding these components is included here.

License

About

SOFTWARE-ENABLED FLASH (SEF) Software Development Kit (SDK)

softwareenabledflash.github.io/SEF-SD...

- BSD 3-clause licensed
- Includes updates to API and Command Set
- Command Set Spec now CSL
- Most important file: **SEF-SDK-01-00.PDF**

Software-Enabled Flash (Re-)Introduction

Member Hardware Update

Q & A Discussion