Open-Channel SSDs and Host-Based FTLs

Ronnie Huang, Co-Founder & VP Engineering

CNEX Labs, Inc.

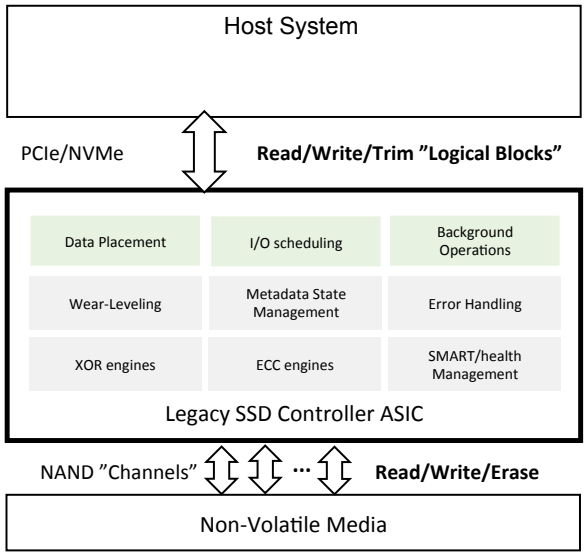
Introduction

- Ronnie Huang,Co-Founder/VP Engineering, CNEX Labs, Inc.

CNEXLABS

- CNEX is a privately held start-up company
- Founded in 2013 by semiconductor industry veterans in Silicon Valley
- Funded by VC and investments from Fortune 500 companies in storage and networking
- Chartered to deliver innovative system solutions in the form of semiconductors and software
- First product is a highly differentiated NVMe SSD controller ASIC
- Currently shipping SDK's; engaged with strategic customers and partners for mass production

SSD Controllers: Terminology and Core Functionality



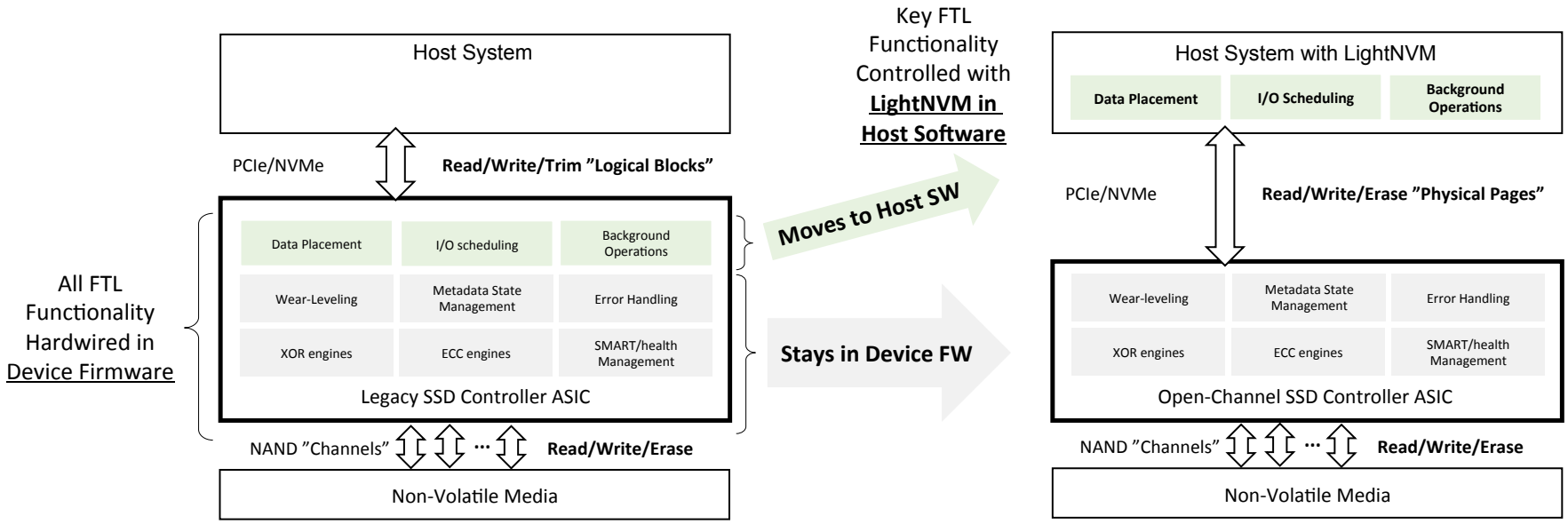
All FTL Functionality Hardwired in <u>Device Firmware</u>

Traditional SSD Controller Hardware and Firmware

Traditional SSD

- Logical Block Addressing (LBA) on Device
- FTL controlled by Device Firmware ("Black-Box")
- Fixed functionality & performance

Open-Channel: Key Concepts



Traditional SSD

- Logical Block Addressing (LBA) on Device
- FTL controlled by Device Firmware ("Black-Box")
- Fixed functionality & performance

Open-Channel SSD

- Physical Page Addressing (PPA) Command Set
- Key FTL functions exposed to LightNVM on Host
- Flexible for application-specific performance

Open-Channel Motivation

- What if every driver coordinated travel with a centralized geo-compute resource (Google Maps)?
  - starting location, destination, priority, car type, etc.?

- Open-Channel SSD:
  - Moves FTL processing from embedded processor in SSD device, to multi-core CPU on Host
  - For efficient/intelligent control over I/O scheduling and data placement
  - To eliminate data congestion

Open-Channel SSDs Offer Many Significant Benefits

- Latency/QoS
  - Low & Predictable latency by efficient timing and execution of garbage collection
  - Reduce/eliminate noisy neighbor effect with deterministic data placement
- Performance
  - Maximize write parallelism and overall read/write bandwidth with physical page addressing
  - Leverage highly efficient host CPU and memory
- Energy Efficiency/Scaling
  - Simplifies SSD controller hardware; reduced CPU/memory requirements on device
- Overprovisioning
  - Flexibility to control (or eliminate) overprovisioning for specific workloads; achieve WAF = 1
- Optimized for Applications
  - Potential to integrate FTL with filesystem or applications, eliminate redundant mappings

Open-Channel Leverages NVMe for Minimal Disruption

- Use existing NVMe Admin and Queuing structure, and NVMe device driver
- Add I/O Commands for "Physical Page Addressing" (PPA)
  - Currently implemented as NVMe "vendor unique" commands;

Open-Channel PPA I/O Commands:

Read PPA: *"Read a PPA, in unit of a sector"*

Write PPA: *"Write to a PPA, in unit of a sector"*

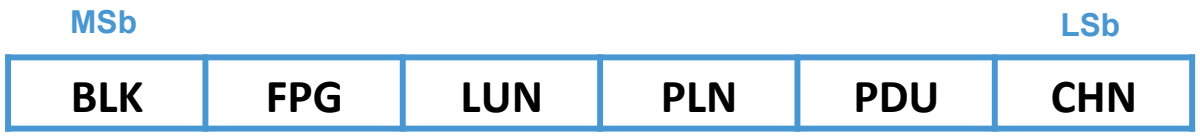
Erase PPA: *"Erase an NVM block"*

Identify Geometry: *"Get geometry of device & media"*

Physical Page Addressing

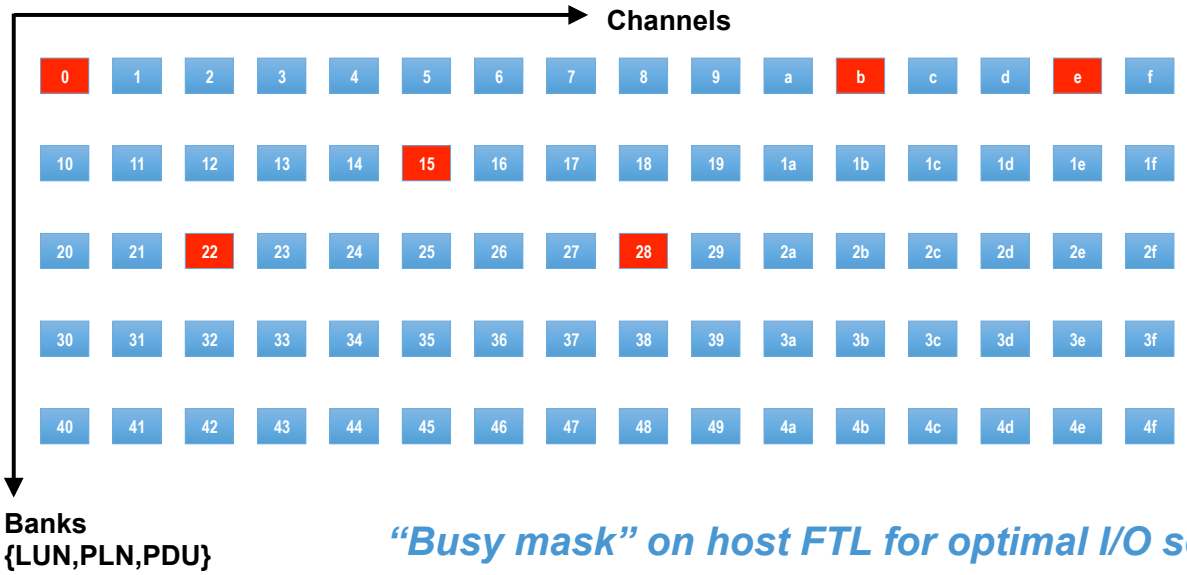
An Example of a Physical Page Address format for NAND Flash:

- CHN: Physical Channel Number (e.g: 4-bits for 16 NAND channels)
- PDU: PPA Data Unit number (2-bits for 4KB ECC protected PPA Page/16KB Flash memory page)
- PLN: Plane number
- LUN: Logical Unit Number (e.g.: 5 for 32 LUNs)
- FPG: Flash Page number
- BLK: Block Number, erasable block number in a plane



“Busy Mask” for Optimal Read/Write Scheduling

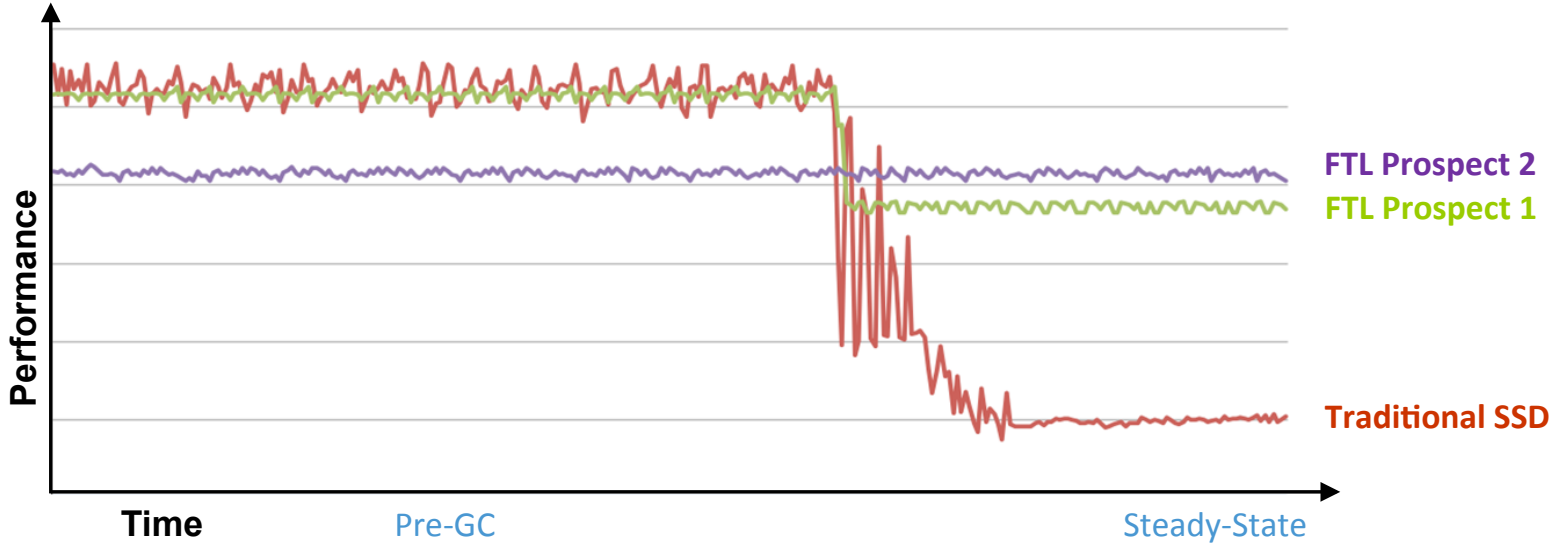
- Intelligent I/O scheduling on host enables read/write efficiency and QoS
  - Host FTL maintains status of busy LUNs
  - Host FTL can schedule writes and reads to avoid physical LUNs that are busy
  - Enables low-latency; deterministic performance



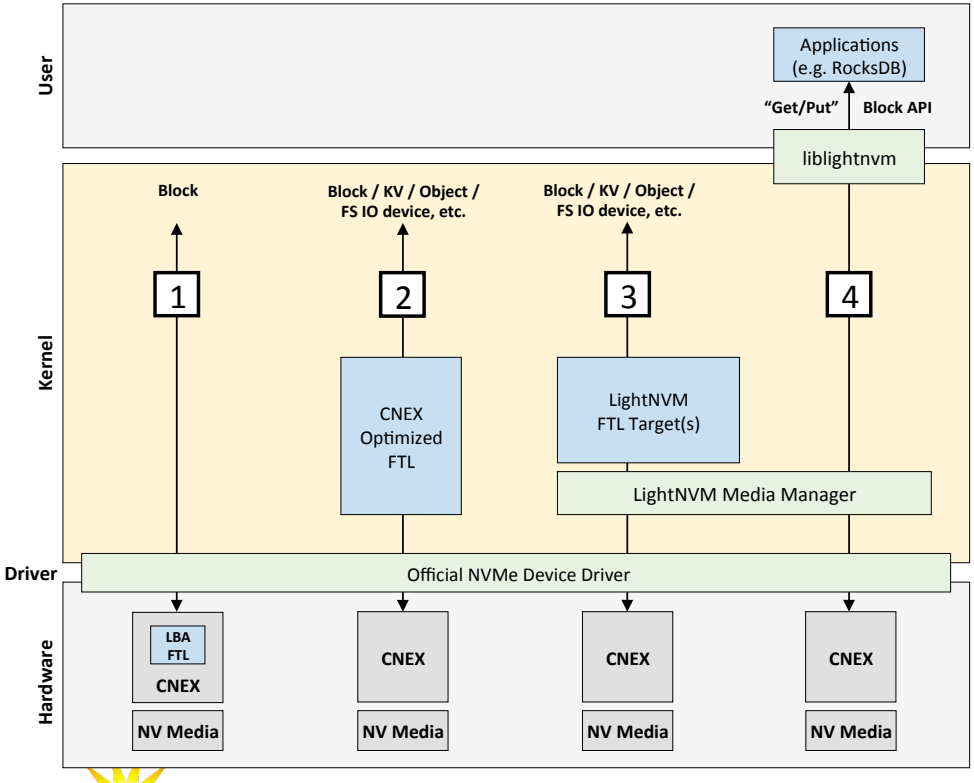
“Busy mask” on host FTL for optimal I/O scheduling

Predictable Performance, Latency

- With Open-Channel SSDs, host FTL software can be tuned for workloads and application types
- Enables data placement by data “type” or “class”, to avoid mixing data within NAND flash blocks
- Reduced overprovisioning, reduced write-amplification, intelligent garbage collection…
- A qualitative example:



CNEX Open-Channel SSD Controller: 4 FTL Options



1 Traditional NVMe
- NVMe Admin, Commands, Queues (LBA mode)

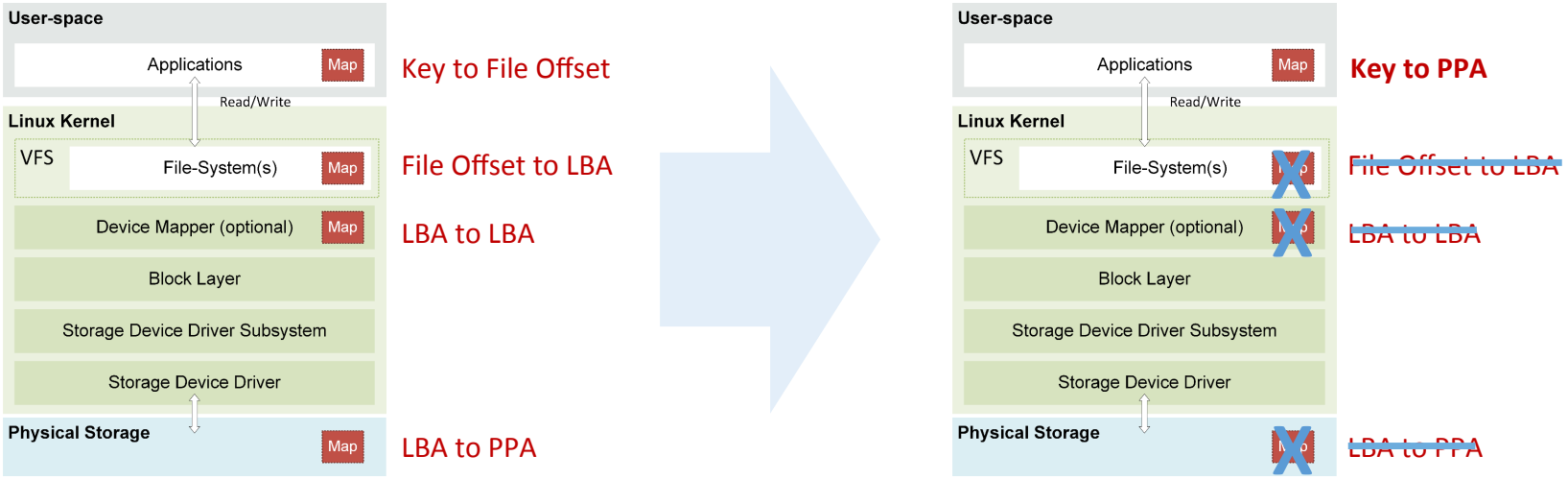
2 CNEX "Optimized" Host FTL
- NVMe Admin, Queues
- NVMe User-defined commands (PPA mode)

3 LightNVM Host FTL
- NVMe Admin, Queues
- NVMe User-defined commands (PPA mode)
- Linux Kernel support

4 LightNVM liblightnvm
- Liblightnvm User Library for "Get/Put" commands
- Linux Kernel support, 4.4+

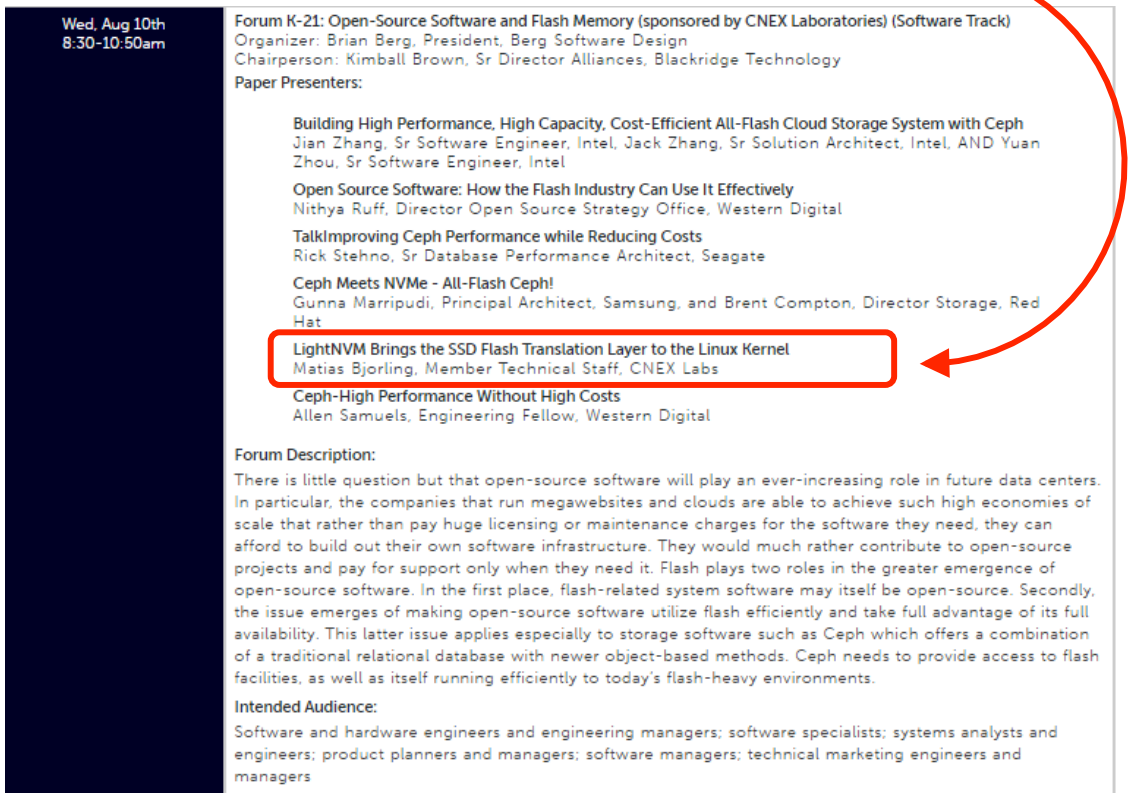
Open-Channel SSD with FTL in User Space

- Potential to collapse multiple layers of redundant mapping in application & filesystem
- Bypass Kernel processing, preserve low-latency characteristics of new/emerging NV Media types



Open-Channel SSD with LightNVM

For more about LightNVM FTL in the Linux Kernel, see:



Wed, Aug 10th
8:30-10:50am

Forum K-21: Open-Source Software and Flash Memory (sponsored by CNEX Laboratories) (Software Track)
Organizer: Brian Berg, President, Berg Software Design
Chairperson: Kimball Brown, Sr Director Alliances, Blackridge Technology
Paper Presenters:

Building High Performance, High Capacity, Cost-Efficient All-Flash Cloud Storage System with Ceph
Jian Zhang, Sr Software Engineer, Intel, Jack Zhang, Sr Solution Architect, Intel, AND Yuan Zhou, Sr Software Engineer, Intel

Open Source Software: How the Flash Industry Can Use It Effectively
Nithya Ruff, Director Open Source Strategy Office, Western Digital

TalkImproving Ceph Performance while Reducing Costs
Rick Stehno, Sr Database Performance Architect, Seagate

Ceph Meets NVMe - All-Flash Ceph!
Gunna Marripudi, Principal Architect, Samsung, and Brent Compton, Director Storage, Red Hat

LightNVM Brings the SSD Flash Translation Layer to the Linux Kernel
Matias Bjorling, Member Technical Staff, CNEX Labs

Ceph-High Performance Without High Costs
Allen Samuels, Engineering Fellow, Western Digital

Forum Description:

There is little question but that open-source software will play an ever-increasing role in future data centers. In particular, the companies that run megawebsites and clouds are able to achieve such high economies of scale that rather than pay huge licensing or maintenance charges for the software they need, they can afford to build out their own software infrastructure. They would much rather contribute to open-source projects and pay for support only when they need it. Flash plays two roles in the greater emergence of open-source software. In the first place, flash-related system software may itself be open-source. Secondly, the issue emerges of making open-source software utilize flash efficiently and take full advantage of its full availability. This latter issue applies especially to storage software such as Ceph which offers a combination of a traditional relational database with newer object-based methods. Ceph needs to provide access to flash facilities, as well as itself running efficiently to today's flash-heavy environments.

Intended Audience:

Software and hardware engineers and engineering managers; software specialists; systems analysts and engineers; product planners and managers; software managers; technical marketing engineers and managers

Flash Memory SUMMIT

Summary

- Significant advantages to Open-Channel SSD with Host FTL
  - Latency, Power, Endurance, RAID, Erasure-Coding, and more…
- Minimal disruption
  - Utilize existing NVMe; add I/O commands for Physical Page Addressing
- OpenChannel SSD and LightNVM FTL is a Growing ecosystem!
  - Participate at: https://github.com/OpenChannelSSD

- **See OpenChannel/LightNVM SSD demos at FMS:**
  - **Liteon: Booth 621**
  - **Micron: Booth 134**
  - **Radian: Booth 615**

Thank-You!

CNEXLABS
www.cnexlabs.com