Flexible Data Placement using NVM Express®

Implementation Perspective

EMPOWERING OPEN.

OCP GLOBAL SUMMIT

OCTOBER 18-20, 2022
SAN JOSE, CA

Storage

Flexible Data Placement using NVM Express®
Implementation Perspective

STORAGE

Dave Landsman, Distinguished Engineer, Western Digital
Dan Helmick, NVMe SSD Architect, Samsung
John Rudelic, SSD Architect, Solidigm Technology
David Wang, SSD FW Architect, Silicon Motion Inc.

OPEN
PLATINUM™

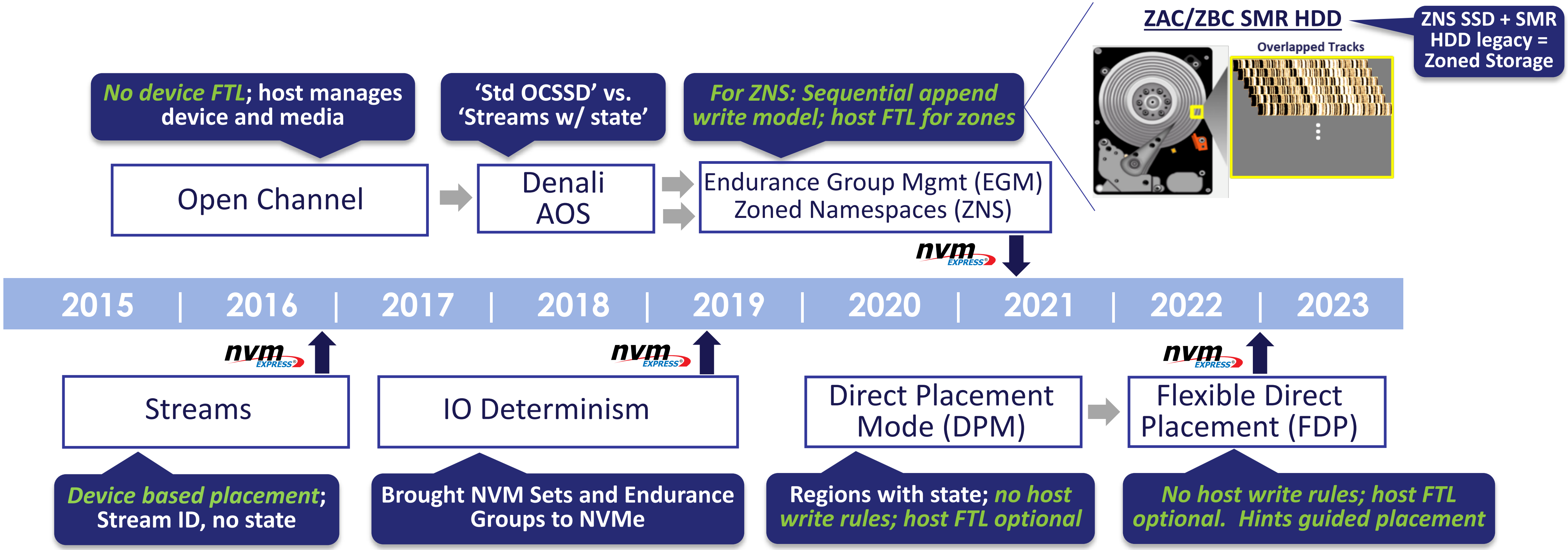
Standards Perspective

Dave Landsman, Distinguished Engineer, Western Digital

OPEN PLATINUM™

Data Placement Discussions In and Around NVMe

Common Goal: Reduce Write Amp ➔ Reduce GC ➔ Better Endurance/Throughput/QoS



ZAC/ZBC SMR HDD

ZNS SSD + SMR HDD legacy = Zoned Storage

No device FTL; host manages device and media

Open Channel

'Std OCSSD' vs. 'Streams w/ state'

Denali AOS

For ZNS: Sequential append write model; host FTL for zones

Endurance Group Mgmt (EGM)
Zoned Namespaces (ZNS)

Streams

Device based placement; Stream ID, no state

IO Determinism

Brought NVM Sets and Endurance Groups to NVMe

Direct Placement Mode (DPM)

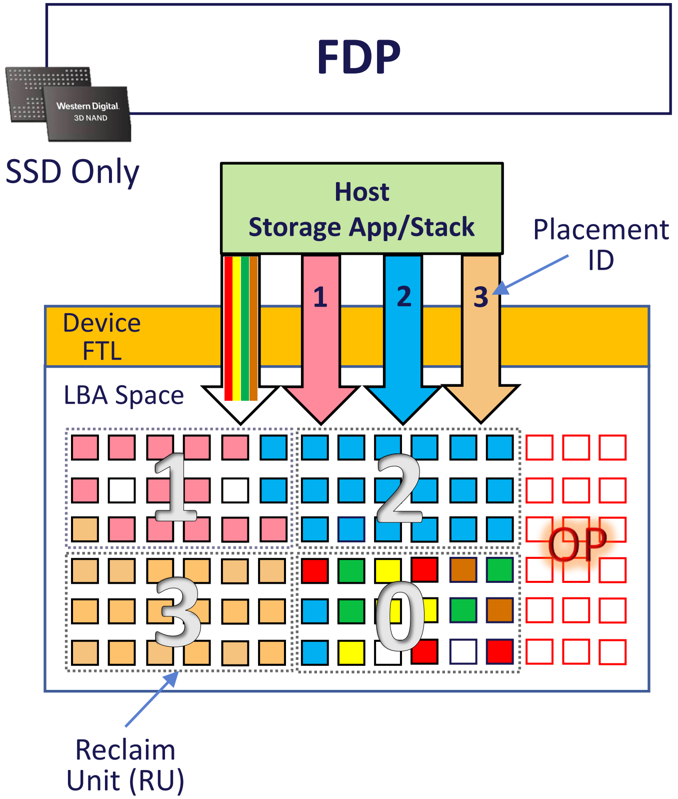
Regions with state; *no host write rules; host FTL optional*

Flexible Direct Placement (FDP)

No host write rules; host FTL optional. Hints guided placement

All before ZNS and FDP abandoned or not widely adopted

FDP and ZNS (Zoned Storage) have their own "Lanes"

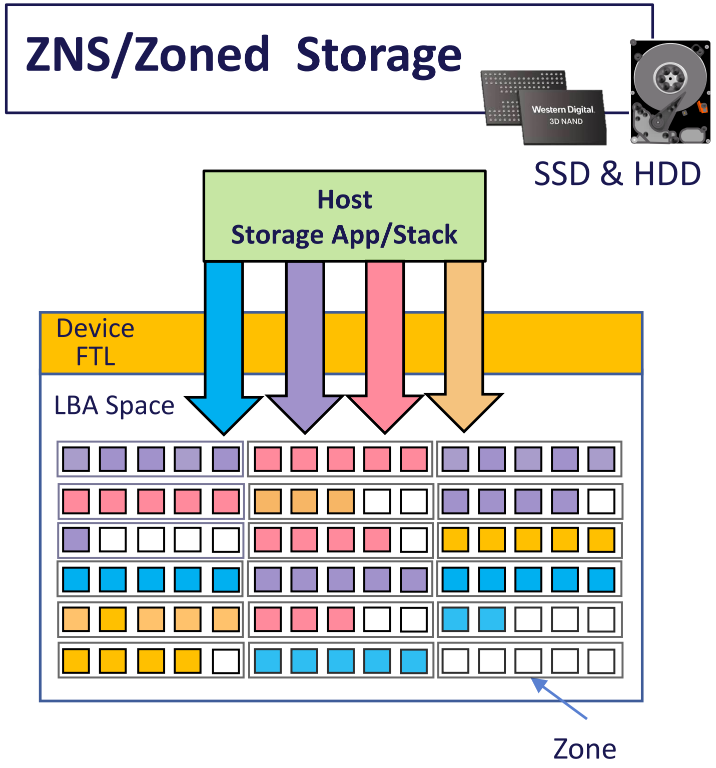


Use Case Contrasts

FDP	ZNS
Compute-centric • Standard Block Device • Mainstream NAND • Std OP%, Std DRAM	Capacity/Cost-centric • Zoned Block Device • Highest capacity NAND • ~0% OP, Reduced DRAM

Protocol Contrasts

FDP	ZNS
Host writes to any LBA in any order	Host writes to append point in Zone
Host guides placement with hints; reduces device GC	Host manages zone GC; sequential writes in zones avoid device GC
Writes never rejected	Non-sequential writes rejected
Host FTL/Optimization optional	Host FTL/Optimization assumed



Let's build out FDP and ZNS; the ecosystem needs stability

Implementation Perspective

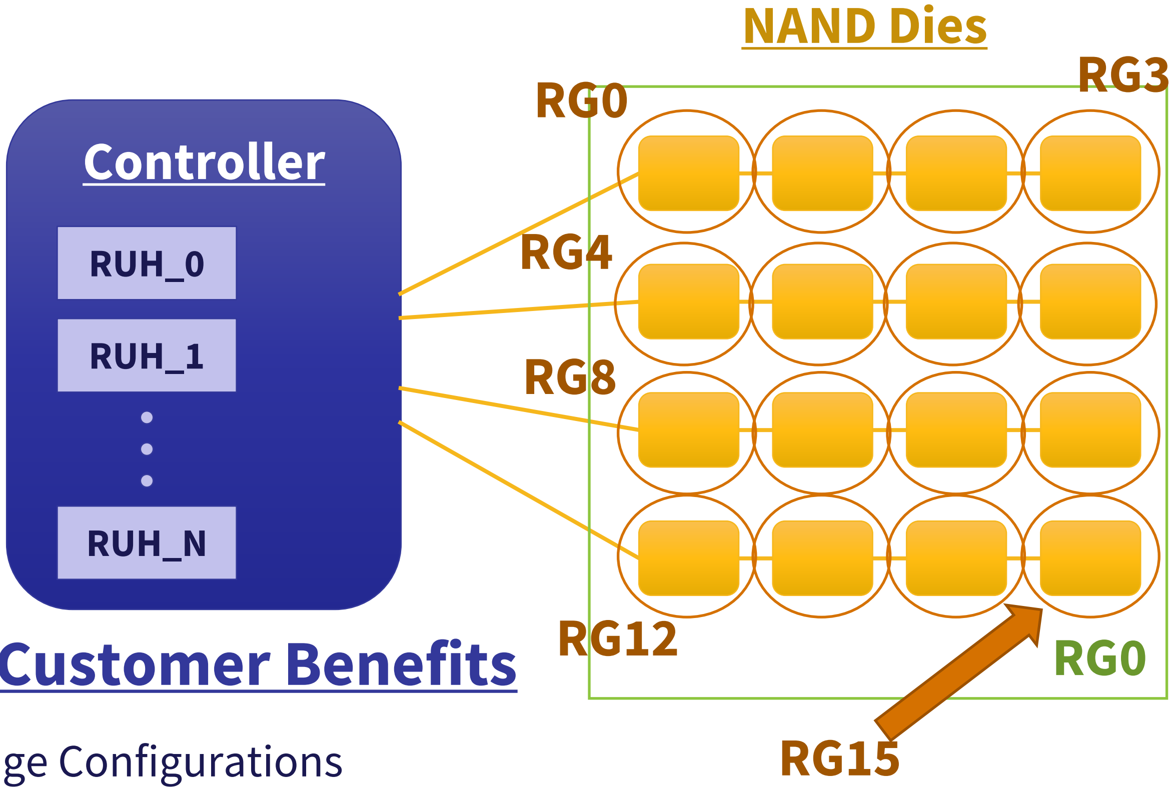
STORAGE

Dan Helmick, NVMe SSD Architect, Samsung

OPEN PLATINUM™

FDP Implementation Externalizes Drive Design Options

- NAND Layout per Configuration
  - Ex: **RG0** vs **RG0-RG15**
  - Ex: RU Formation
- Efficient Controller Resource Addressing and Utilization
  - RUH reporting
  - Power Fail Configurations
- Data Rerouting on Media Exceptions
- Reclaim Group selection for Legacy Users
- Extensible and Scalable for Future Feature Additions



Highlighted Customer Benefits

- Server vs Storage Configurations
- Legacy Interoperability
- Command checking by Drive rather than Host Layers
- Event Logging rather than Error Interrupts

Implementation Perspective

STORAGE

John Rudelic, SSD Architect, Solidigm Technology

OPEN PLATINUM™

Optimization Relies on Partnership

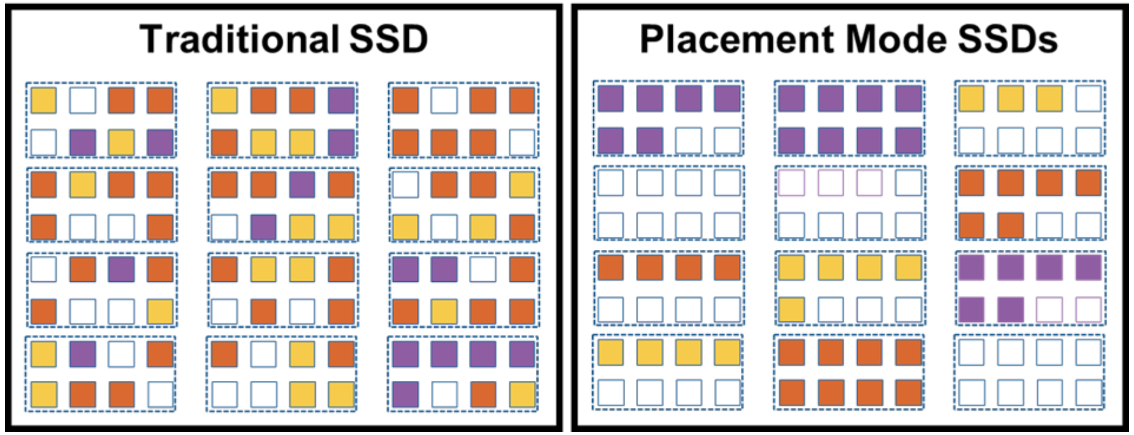
Key Host Side Considerations:

- Host responsibilities with TP-4146
  - Parallel IO scheduling for performance
  - NAND constraints/architecture
  - NAND features and benefits
  - SSD geometry / capacity
  - Garbage collection / housekeeping
  - New commands

- System Benefits – Better SSD Utilization
  - Significant performance (e.g. WA improvement)
  - QoS improvements
  - Write amplification improvement
  - Host placement granularity
  - Flexible use cases – (compute & storage)

Controller Considerations & Industry Alignment on Use Cases:

- Number of supported configuration(s)
- Capability for "Default PID"
- Future NAND features
- NAND Evolution (divergent features)
- IU size with large capacity SSDs



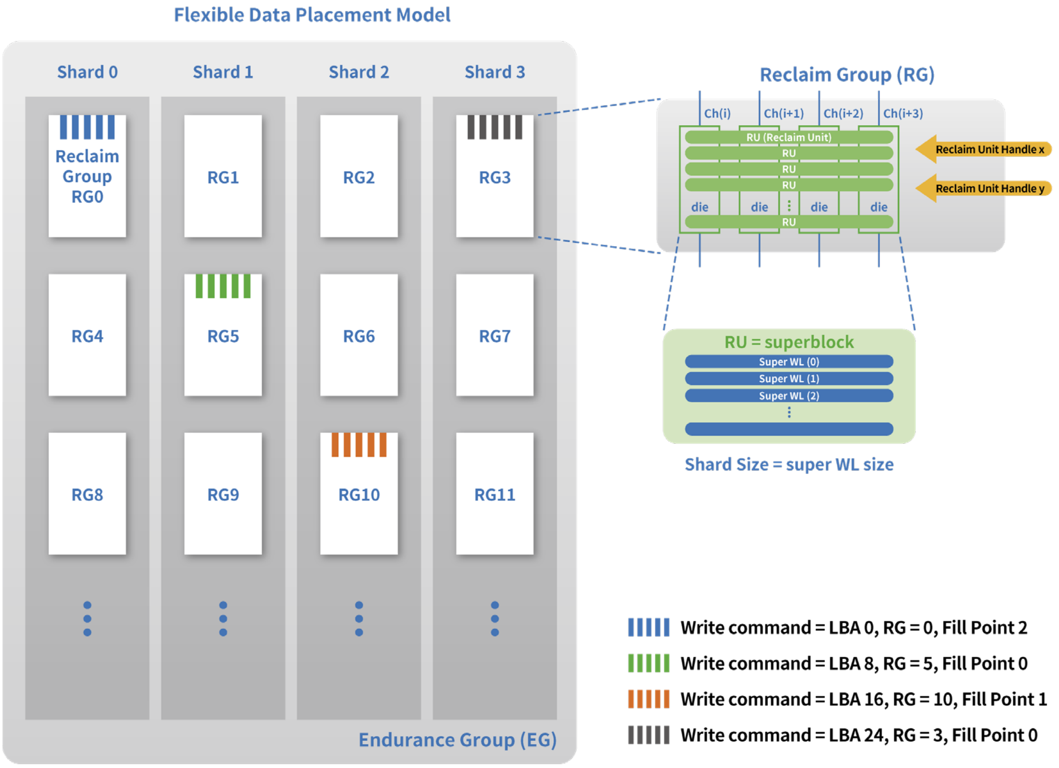
Implementation Perspective

David Wang, SSD FW Architect, Silicon Motion Inc.

FDP Implementation Difficulties

- FDP application usually requires a big number of open blocks. This implies SSD controller provide <u>a large size of high bandwidth write buffers</u>
    - Smaller size of RU (e.g., in 1GB level better for GC efficiency) means more Reclaim Groups (RGs) and open block number (i.e., concurrent write/fill pointers)
    - To deal with write-write collision and write-erase collision, also need the extra write buffer size to buffer host data during waiting time

- Using DRAM as write buffers is one feasible way but there are two challenges
    - <u>DRAM bandwidth</u>
    - <u>Supercap restriction</u>



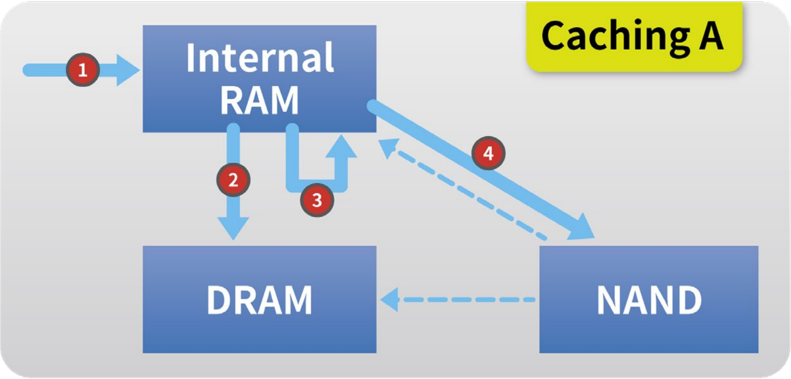
The content of this document including, but not limited to, concepts, ideas, figures and architectures is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Silicon Motion Inc. and its affiliates. Silicon Motion Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this document.

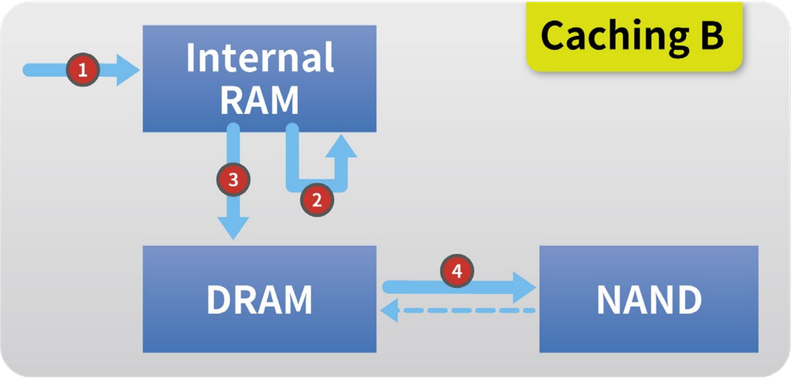
Gen5 NVMe Controller Solution

1. NVMe Controller may support high bandwidth DDR5 DRAM, that can be used as Write Buffers in Write IO path without performance loss
   - Dual Channel DDR5 bandwidth: 4.8GHz*8B*0.7 (DDR efficiency) = ~27GB/s, then max supported NAND Prog throughput is ~13.5GB/s

2. Use Flexible Write Cache to Configurable Open Block Number

Open Block Number	**Buffer** (for Data Operation and NAND Prog)	**Cache** (for backup in case prog failure)	**Mechanism**
Small/medium num (e.g. <=32)	Internal RAM → NAND	DRAM	A
large num (e.g. <=128)	Combined Internal RAM + DRAM → NAND	DRAM Limited by Supercap*	B





* For example, Supercap capacitance 1500uF may only protect ~100MB DRAM data (write buffers and other FTL metadata) during power lose.

Nothing in these materials is an offer to sell any of the components or devices referenced herein.
Silicon Motion Inc. may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document.

Thank you!

EMPOWERING OPEN.

OCP
GLOBAL
SUMMIT

OCTOBER 18-20, 2022
SAN JOSE, CA