Hyperscale Innovation: Flexible Data Placement Mode (FDP)

Sponsored by NVM Express organization, the owner of NVMe® specifications

Chris Sabol, Google

Ross Stenfort, Meta

Write Amplification Overview

- What is Write Amplification (WA)?
  - When the host sends write data to the device it is additional data that is written to the media.
  - Write Amplification Factor (WAF) = media written data/ host written data

- WAF = 2.5 Example
  - Host writes 1 MB
  - Device writes 2.5 MB to the media
  - Thus Device
    - Media Writes
      - 1 MB Host Data
      - 1.5 MB Garbage Collected Data
    - Extra Media reads to enable host write
      - 1.5+ MB

Why is Write Amplification Undesirable?

- Write Amplification results in additional:
  - Media Reads/ Writes affecting performance/ QOS
  - Flash media writes causing non-host induced media wear
  - Additional power needed to perform the additional reads/writes

Google Datacenter Infrastructure WA Impact

- Example with random 4KiB writes, 28% OP, and greedy GC algorithm, can expect a WAF ~2.5

WAF Reduction from 2.5 to 1.25 Benefits	Benefit
Reduce Over Provisioning (OP)/ Higher Usable Capacity	18% Capex Savings
Enable 2x drive size with the same application write density	7% Capex / 15% Opex Savings
Double effective drive lifetime	Up to 35% Capex Savings
Enable 2x application write rate	Performance

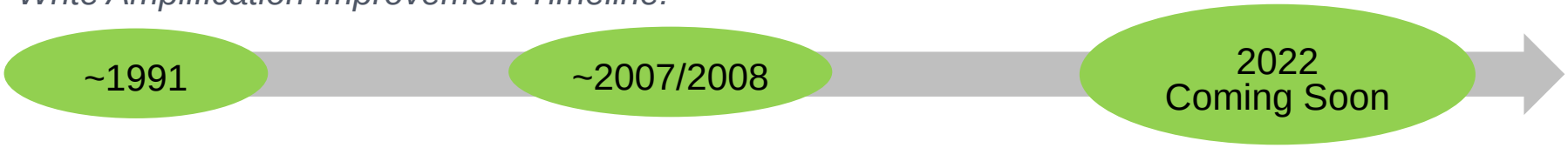
Write Amplification has very significant hyperscale TCO impact

Imagine a World of WAF = ~1

- What would this mean?
  - SSD overprovisioning would significantly decrease, and user capacities would increase
    - 28% OP Devices would go away in a WAF of 1 world
  - Performance
    - Random and Sequential Write would have similar performance
    - No need to precondition
    - Improved QOS for read and write
  - Media wear would be reduced
    - Devices last longer without NAND media changes

History of Write Amplification Improvements

Write Amplification Improvement Timeline:



NAND Based SSDs
Solution #1: Overprovisioning

Host provides SSD LBA Hints
Solution#2 TRIM/Deallocate

Host provided data placement hints
Solution #3 Flexible Data Placement

- How did Flexible Data Placement come about?
  - Google Write Amplification Investigation Result
    - Data placement on media is key
    - SMART FTL Proposal
  - Meta Write Amplification Investigation Result
    - Data placement on media is key
    - Direct Placement Mode Proposal
  - Google & Meta merged their independent learnings into Flexible Data Placement (FDP) merging the best features of each proposal to enable best industry solution

Flexible Direct Placement Overview

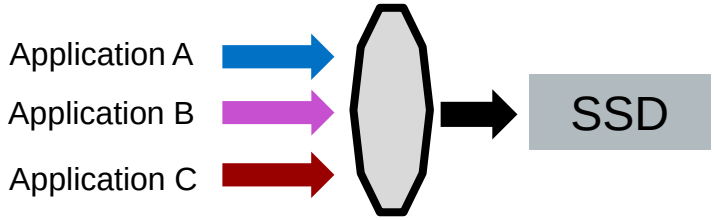
- High Level Goal
  - Host provides write hints for media placement
  - Device reads and other behaviors do not change

FDP is targeted for
- Datacenter SSDs
- Backwards compatible with legacy hosts

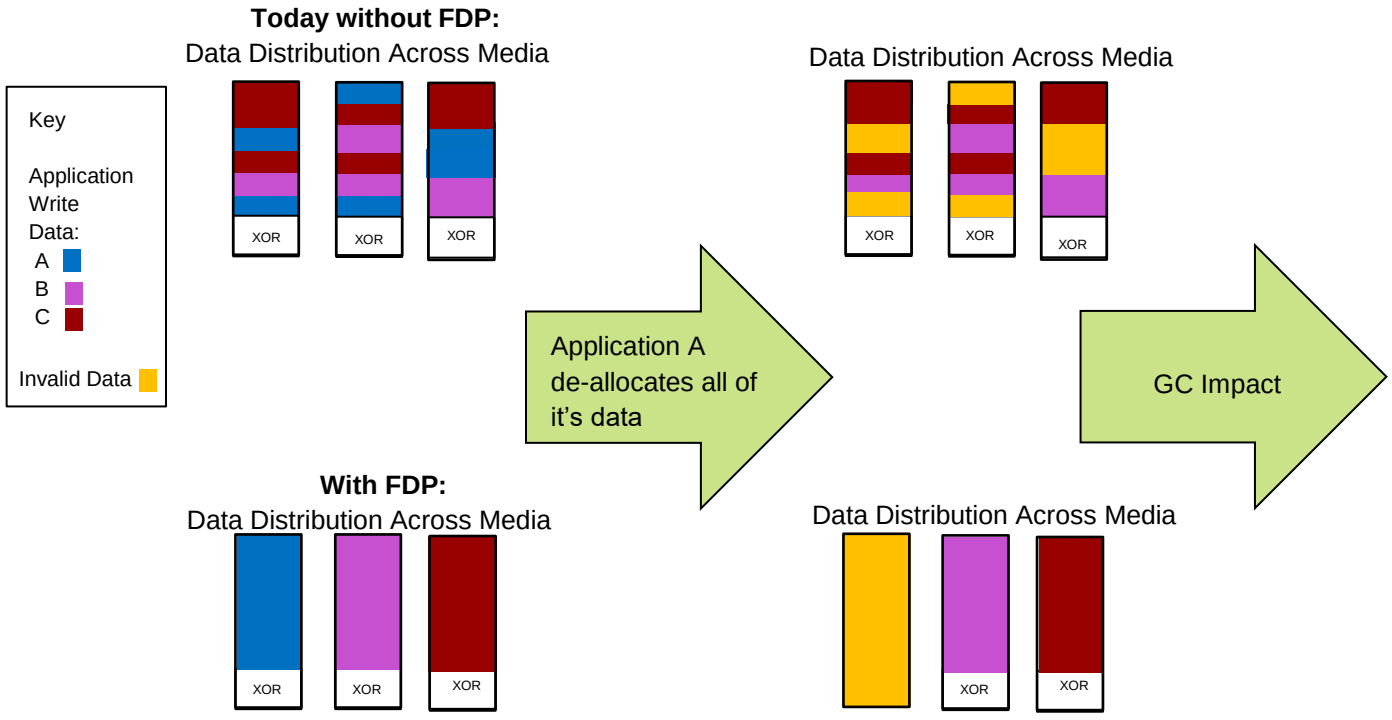
Functionality	FDP Support
Standard Device Feature Enable/Disable	Yes
Host Enable Data / Media Alignment	Yes
Read Operations	No Changes
Enable Erase On Media Boundaries	Yes
LBA Placement Restrictions	No
Media XOR Support	Optional
Multiple Namespace Support	Optional
Backwards Compatible	Yes

Flexible Direct Placement Use Case Challenge

- Multi-user/ Multi-workload/ Disaggregated Storage
- Today's Challenges
  - Application's Data is Mixed
  - Device performance is unstable
    - Never reaches "steady state" due to mixed workloads
  - Overprovisioning is increased until WA is low enough and performance appears stable
  - Workload changes causes process above to repeat



Flexible Direct Placement Solution



Results:

Today's Method:
All Media blocks must be garbage collected resulting in a WAF ~3.

FDP Method:
Only single media block erased resulting in WAF = ~1

Flexible Direct Placement TP4146 Next Steps

- FDP is working through the NVM Express standardization process

Looking forward to a new world, where a WAF =~1 is commonplace.

Thank You

Architected for Performance