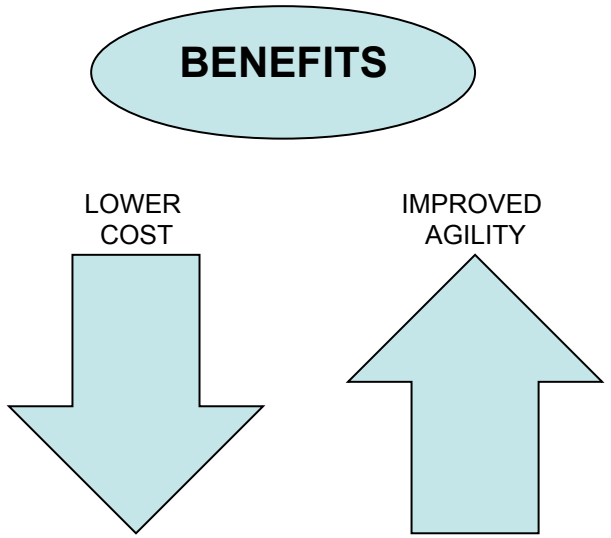
Flash in a Software Defined Storage World

Priyadarshi Prasad (@Priyadarshi_Pd)
Sr. Director of Product Mgmt., Atlantis Computing

Software Defined Storage

- Storage - Persistent, Reliable, Available
- Software Defined - Intelligence in software, allowing to use commodity, off-the-shelf hardware components



Flash

- SLC
- eMLC
- MLC
- cMLC
- TLC
- QLC

LOWER PERFORMANCE

LOWER COST

SDS SWEETSPOT

Using Flash Intelligently

- Good for Reads
- Treat Writes carefully
  - Manage OP, GC, Partial Stripes
- Keep it close to the server (avoid protocol latency)

*Intelligent SDS design should be able to use the "cheapest, cr**piest" flash*

Using Flash Intelligently - Example 1

- In-Memory Deduplication
- Especially suited for write intensive workloads (e.g. VDI)

PROBLEM:
Write-intensive workloads requiring sub-ms performance

CHALLENGE:
Keep solution price low

SOLUTION:
cMLC Flash + Intelligent SDS

ARCHITECTURAL APPROACH:
Inline deduplication - only unique writes make it to underlying media (flash)

Using Flash Intelligently - Example 2

- Flash for Reads only
- Enables using Fry's Flash in Enterprise datacenters

PROBLEM:
Read-intensive workloads requiring sub-ms performance

CHALLENGE:
Keep solution price low

SOLUTION:
cMLC Flash + HDDs + Intelligent SDS

ARCHITECTURAL APPROACH:
Separate performance from storage, keeping performance in hosts and storage outside.

Using Flash Intelligently - Example 3

- Flash for both Reads and Writes
- Enables widest application of Flash

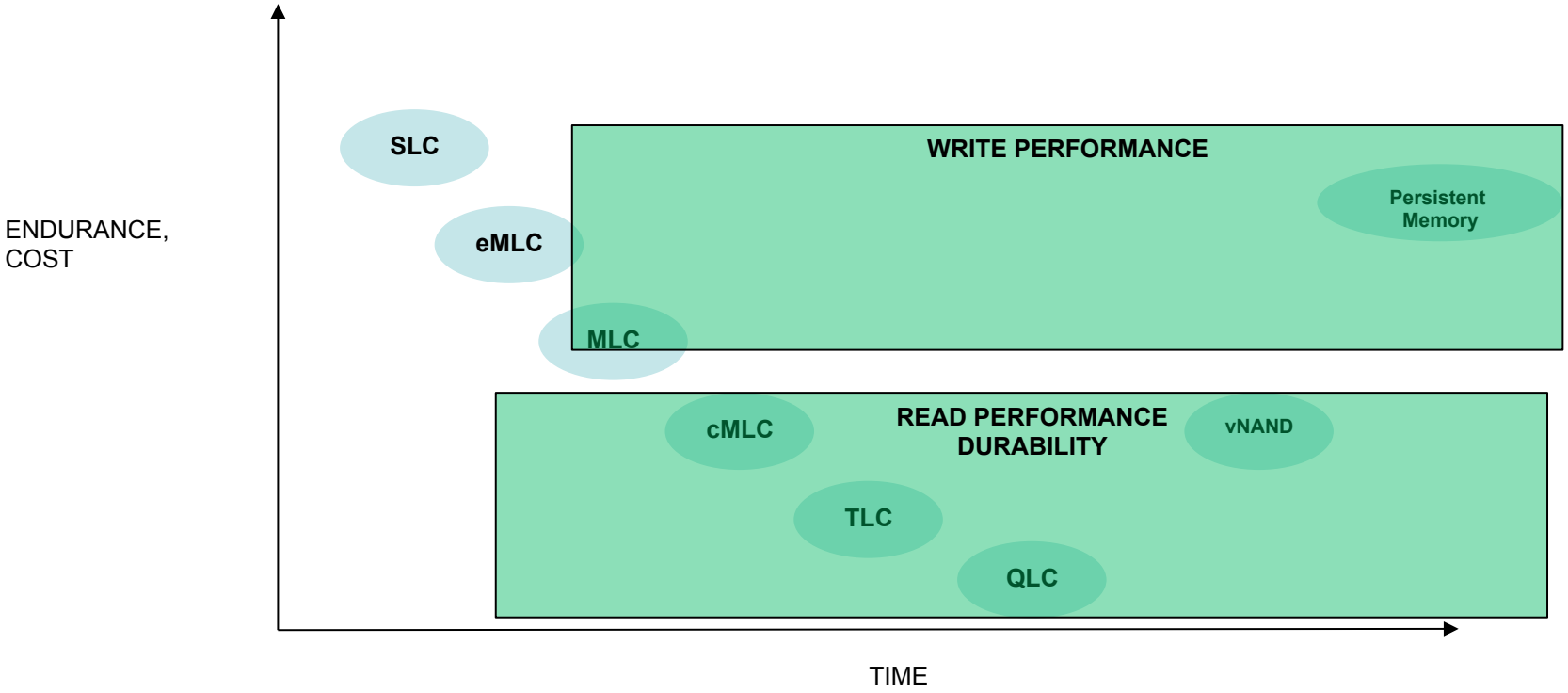
PROBLEM:
Balanced (R/W) workloads requiring sub-ms performance

CHALLENGE:
Solution price, Flash Endurance, Performance

SOLUTION:
eMLC + cMLC/HDD + Intelligent SDS

ARCHITECTURAL APPROACH:
Tiered solution, often using eMLC (high DWPD) to absorb writes. Data locality in nodes for reads.

Looking Forward



Takeaways

- Understand your SDS architecture - is it *flash optimized for your use case*
- “FLASH” is too generic a term - choose flash per your application needs
- How will the SDS architecture leverage Persistent Memory

Thanks

@Priyadarshi_Pd