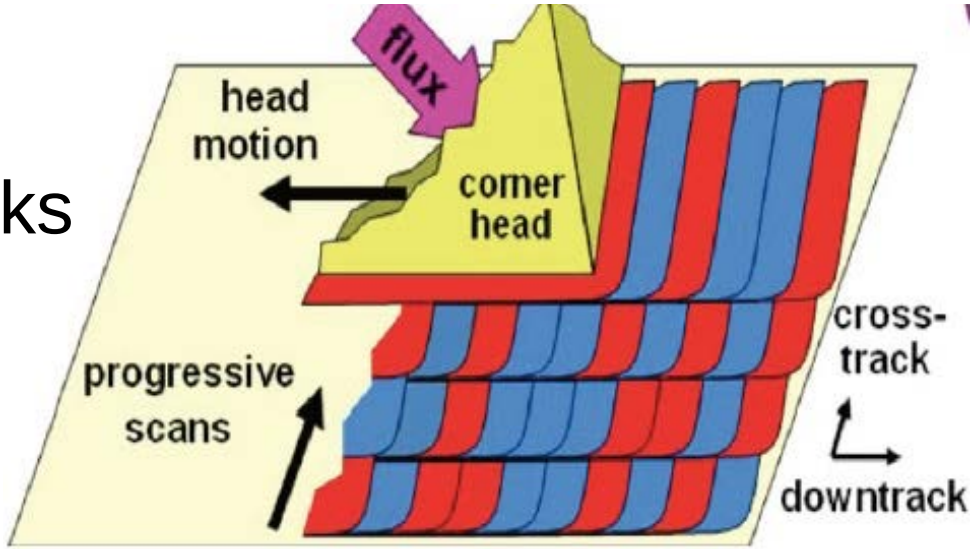
SDC
STORAGE DEVELOPER CONFERENCE
SNIA SANTA CLARA, 2016

Shingled Magnetic Recording (SMR) Panel: Data Management Techniques Examined

Tom Coughlin
Coughlin Associates

Introduction

- SMR partially overwrites written tracks with new tracks
- Creates an “erase” process in HDDs
- Increase in Areal Density but increase in performance overhead too



Panel Participants

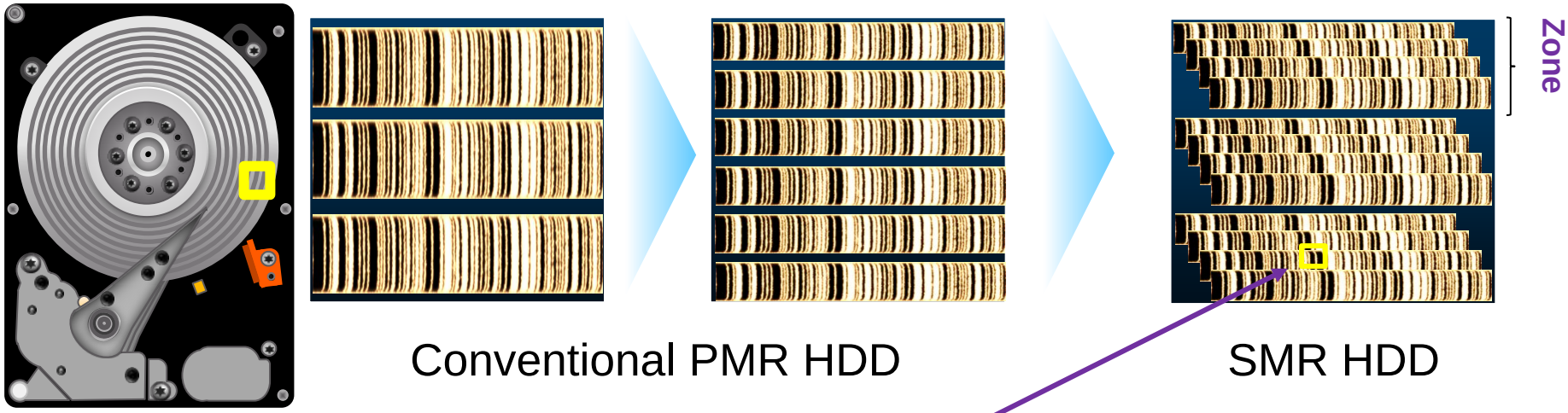
- Damien Le Moal, Ph.D.
  - Sr. Manager, System Software Group, HGST Research, Western Digital
- Mark Carlson
  - Principal Engineer, Industry Standards, Toshiba
  - Chair, SNIA Technical Council
- James Borden
  - Principal Cloud Architect, Seagate Technology

SDC
STORAGE DEVELOPER CONFERENCE
SNIA SANTA CLARA, 2016

How SMR is Implemented on HDDs

Damien Le Moal
Western Digital

What is Shingled Magnetic Recording?



Conventional PMR HDD

Data in Discrete Tracks

SMR HDD

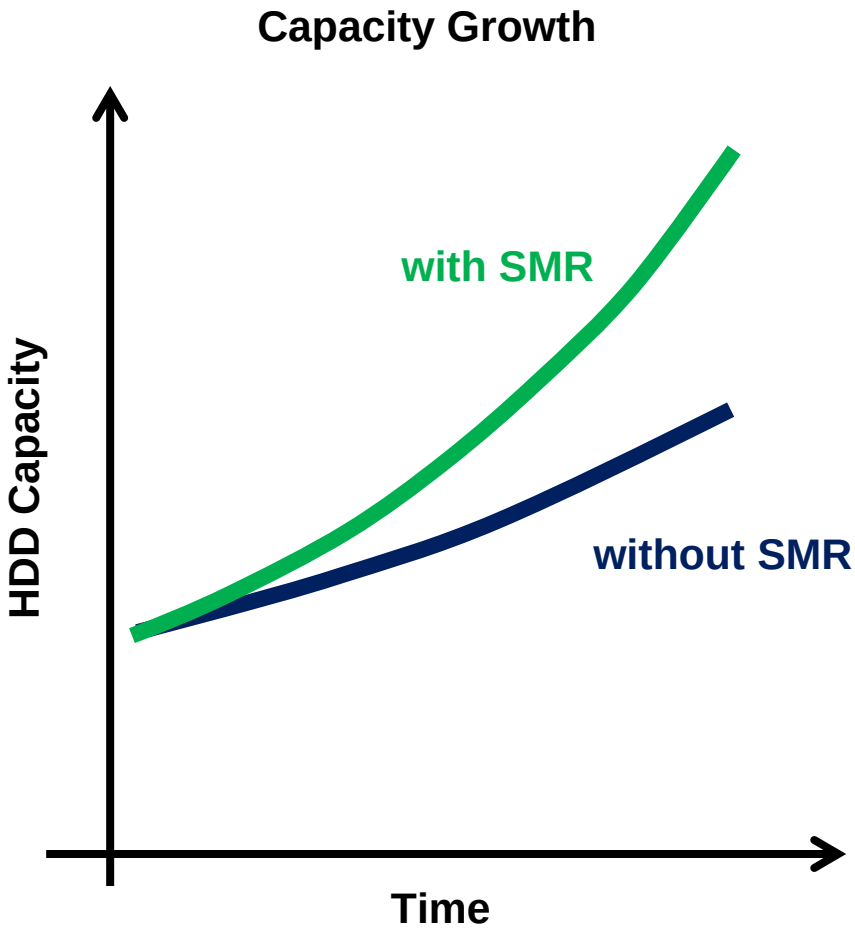
Data in Zones of Overlapped Tracks

While Zones are independent, we can't change sectors independently within a Zone.

SMR Standards

- T10: ZBC
- T13: ZAC

Why SMR?



SMR accelerates areal density growth

Some Architectural Constructs

- Caching
  - Stage writes to sequentialize the IOs.
  - This can be done both on the media or on Solid State Storage.
- Indirection system
  - Not a fixed mapping from LBA to physical location
- Over provisioning
  - Need extra space for internal bookkeeping
- Garbage Collection
  - Need background process to fix up the data-structures.
- Indirection system storage
  - Need special mechanism to maintain the indirection system.
- Solid State NV Storage
  - Emergency storage for indirection system

Drive Managed Model

- Sequential Read
  - Similar to PMR

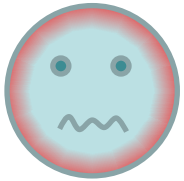
- Random Read
  - Similar to PMR

- Sequential Write
  - Similar to PMR

- Random Write
  - YMMV

?

Drive Managed Model: Random Write

	Small Block	Large Block	Huge Block
High Duty Cycle	Performance dominated by seek time. Caching writes on media and moving later has good performance. 	Seek time no longer dominates. Writing twice has overhead. High duty cycle fills up cache quickly and doesn't allow time for recovery. 	Behaves close to sequential writes. 
Low Duty Cycle	Performance dominated by seek time. Caching writes on media and moving later has good performance. 	Seek time no longer dominates. Writing twice has overhead. Low duty cycle allows drive to hide overhead. 	Behaves close to sequential writes. 

SMR Management Models and Standards

Mark Carlson
Toshiba
SNIA Technical Council

SMR Management Models

- Hide the complexity of SMR from host software
  - Drive Managed model – performance impact
- Allow the host software to manage the SMR complexity
  - Host Managed Model – best performance, but all new software
- Something in between
  - Host Aware Model

SMR Standards

- For Serial ATA (SATA)
  - ZAC – Zoned ATA Commands
- For Serial Attached SCSI (SAS)
  - ZBC – Zoned Block Commands
- Two primary commands
  - **Report Zones** – discover zone configuration and write pointers
  - **Reset Write Pointer** – reset the write pointer the beginning of zone (destructive to zone contents)

Writing to a Zoned Device

- Additional commands:
  - **Open Zone** – nail down resources for a zone
  - **Close Zone** – free up those resources
  - **Finish Zone** – fill out the remaining space
- Proposed simplification
  - Allow **Report Zones** even in a Drive Managed model

SMR Standards Future

- T13 working on ZAC-2
- T10 working on ZBC-2
- Infrastructure software changes are now being put in place
  - Hyperscalers building into their infrastructure
  - OS and Hypervisor software being updated
- Developers still needed to implement changes

SDC
STORAGE DEVELOPER CONFERENCE
SNIA SANTA CLARA, 2016

Host Implementation Considerations with SMR

Seagate Technology

Primary Host Issue: Non-Sequential Writes

- File modifications via appends are a primary example (write in place) - does not conform to ZAC/ZBC
- Host workloads would need to become copy-on-write for modifications and discard/trim old data
- Several Implementation approaches:

Full Stack Solution

- For customers who can develop/adopt ZAC/ZBC compliant stacks…
  - Strict latency sensitivity: Host-managed SMR and fully compliant stack.
  - Less strict latency sensitivity and/or not fully compliant stack: SMR Host-aware

Emulation / Shim: Sequentializer (STL)

- Translation layer akin to FTL that sits below the file system
- Manages LBA remapping and supports metadata storage and searches, as well as garbage collection
- Possible workload dependent performance implications, but majority of the I/O stack will not need to change
- Open Source prototype example for Linux
  - https://github.com/Seagate/ZDM-Device-Mapper

Emulation / Shim: Caching

- Use part of the drive (conventional space) for a “random” cache to clean later
- Garbage collection and metadata tracking/searching required, similar to sequentializer
- Different performance tradeoffs than sequentializer

SMR Market Considerations

- SMR is and will be an enabler for higher HDD areal density (with TDMR, HAMR, etc.)
- Drive-managed SMR is a good solution in low latency implementations (e.g., backup, video surveillance)
- Host-aware and Host-managed implementations are currently most applicable and active in Enterprise and Cloud Segments (greatest ability to drive stack changes)
- Appropriate use-case implementation of SMR should be able to deliver performance similar to CMR HDDs
- Market adoption is positioned for a consistent, steady ramp, with initial growth in hyperscale followed by enterprise customers