Shingled Magnetic Recording Drives

Csci 5980 Spring 2017
SMR using In-place Updates
Background

• Traditional HDDs (perpendicular magnetic recording) are reaching areal data density

• Shingled magnetic recording is a new promising technology
SWD Characteristics

• Sequential write is preferred
• Write/update a block in place may destroy the valid data on the subsequent tracks if any
• General approaches to updates:
  • In-place update:
    • Extra reads/writes
  • Out-of-place update: 1 write = 2 reads + 3 writes
    • Copy-on-write
    • Mapping table and Garbage collection

• Write amplification
  • Update may incur extra read/write operations

![Shingled tracks diagram](image)
Tradeoff Between Space and Performance

- SWD Layout
  - Tracks are organized into bands
  - There are safety gaps between bands

A good candidate: W=2, N=4
SG = 1.6
WAR = 4

\[ SG = W \frac{N}{N + W - 1} \] (1)
\[ WAR = \frac{1}{N} \sum_{i=0}^{N-1} (1 + 2i) = N \] (2)

Write width = W
Band size = N
Objective

• Good balance between capacity and performance
  • Reduce the write amplification overhead
Motivation for New Schemes

- **General rule**
  - Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th>Affected Tracks</th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>25%</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>

Single Band comparison
Motivation for New Schemes

• General rule
  • Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th>Affected Tracks</th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>25% no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>50% Track1</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>

Single Band comparison
Motivation for New Schemes

- General rule
  - Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th>Affected Tracks</th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>25%</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>50%</td>
<td>Track1</td>
<td>no</td>
</tr>
<tr>
<td>75%</td>
<td>Track 1,2</td>
<td>Track1</td>
</tr>
</tbody>
</table>

Single Band comparison
Motivation for New Schemes

• General rule
  • Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th>Affected Tracks</th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>25%</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>50%</td>
<td>Track1</td>
<td>no</td>
</tr>
<tr>
<td>75%</td>
<td>Track 1,2</td>
<td>Track1</td>
</tr>
<tr>
<td>100%</td>
<td>Track 1,2,3</td>
<td>Track 1,2,3</td>
</tr>
</tbody>
</table>

Single Band comparison
Motivation for New Schemes

• General rule
  • Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th></th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affected Tracks</td>
<td>25% Track 1</td>
<td>no</td>
</tr>
</tbody>
</table>

Two Bands comparison

---

1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8
Motivation for New Schemes

• General rule
  • Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th>Affected Tracks</th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>25%</td>
<td>Track 1</td>
<td>no</td>
</tr>
<tr>
<td>50%</td>
<td>Track 1,2,3</td>
<td>no</td>
</tr>
</tbody>
</table>

Two Bands comparison
Motivation for New Schemes

• General rule
  • Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th></th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>25%</td>
<td>Track 1</td>
<td>no</td>
</tr>
<tr>
<td>50%</td>
<td>Track 1,2,3</td>
<td>no</td>
</tr>
<tr>
<td>75%</td>
<td>Track 1,2,3,5</td>
<td>Track1,5</td>
</tr>
</tbody>
</table>

Affected Tracks

Two Bands comparison
Motivation for New Schemes

• General rule
  • Delay the use of track(s) in the middle of the bands, e.g., 3rd tracks

<table>
<thead>
<tr>
<th>Affected Tracks</th>
<th>Conventional</th>
<th>Novel</th>
</tr>
</thead>
<tbody>
<tr>
<td>25% 25%</td>
<td>Track 1</td>
<td>no</td>
</tr>
<tr>
<td>50% 50%</td>
<td>Track 1,2,3</td>
<td>no</td>
</tr>
<tr>
<td>75% 75%</td>
<td>Track 1,2,3,5</td>
<td>Track1,5</td>
</tr>
<tr>
<td>100% 100%</td>
<td>Track 1,2,3,5,6,7</td>
<td>Track 1,2,3,5,6,7</td>
</tr>
</tbody>
</table>

Two Bands comparison
Scheme 1: R(4123)

1\textsuperscript{st} 25\%: 4\textsuperscript{th} tracks
Scheme 1: R(4123)

1\textsuperscript{st} 25\%: 4\textsuperscript{th} tracks

2\textsuperscript{nd} 25\%: 1\textsuperscript{st} tracks
Scheme 1: \( R(4123) \)

1st 25%: 4th tracks

2nd 25%: 1st tracks

3rd 25%: 2nd tracks
Scheme1: R(4123)

1\(^{st}\) 25\%: 4\(^{th}\) tracks
2\(^{nd}\) 25\%: 1\(^{st}\) tracks
3\(^{rd}\) 25\%: 2\(^{nd}\) tracks
4\(^{th}\) 25\%: 3\(^{rd}\) tracks

Similarly, R(1423) can be adopted.
Scheme 2: 14R(23)

1st 50%: 1st and 4th tracks
Scheme 2: 14R(23)

1st 50%: 1st and 4th tracks

The following 25%: 2nd tracks
Scheme 2: 14R(23)

1st 50%: 1st and 4th tracks

The following 25%: 2nd tracks

The last 25%: 3rd tracks
Scheme 3: 124R(3)

1st 75%: 1st, 2nd and 4th tracks
Scheme 3: 124R(3)

1st 75%: 1st, 2nd and 4th tracks

The last 25%: 3rd tracks
Scheme Comparisons

- SWD setup:
  - $W = 2$
  - $N = 4$

<table>
<thead>
<tr>
<th>scheme</th>
<th>Spatial Locality</th>
<th>Write Amplification</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDD</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>R(4123)</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>14R(23)</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>124R(3)</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>1234</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

5 is best, 1 is worst
Evaluation Design (1/2)

- **SWD simulation**
  - Disksim, with hp_c3323a disk model
  - **Address mapper**: translate LBAs to PBAs
  - **Write amplifier**: convert an update into several reads/writes accordingly

- **SWD setup**:
  - 3000 cylinders
  - 1000 blocks per cylinder
  - Band size = 4
  - Write width = 2

<table>
<thead>
<tr>
<th>Trace</th>
<th>Write %</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web_0</td>
<td>0.70123</td>
<td>Update intensive</td>
</tr>
<tr>
<td>Financial_0</td>
<td>0.096978</td>
<td>Update light</td>
</tr>
<tr>
<td>hp_c2247</td>
<td>0.488449</td>
<td>Update moderate</td>
</tr>
<tr>
<td>SYN</td>
<td>1</td>
<td>Sequential write, average size 8 blocks, IAT=(mean 50ms, std. dev 10ms). No update.</td>
</tr>
</tbody>
</table>
Evaluation Design (2/2)

• Performance test points:
  • 25%, 50%, 75%, 100%

• For each test point, “pre-fill” the SWD space to the corresponding percentage to logically convert writes to updates
  • This is done to pass the percentage to the write amplifier
Results
Summary

• Shingled magnetic recording drives
• Write amplification problem
• Achieve good space gain and performance balance with new static address mappings
  • R(4123) or R(1423)
  • 14R(23)
  • 124R(3)
SWD using hybrid updates
Motivation

- In-place update SWDs spent a fair amount of space on safety gaps, as a result of small bands.
  - 20% in the case of 4-tracks band and 2-tracks write head

- Can we improve the space gain?
  - Yes, use larger band!

- Challenges with larger band?
  - In-place update is not acceptable
  - Out-of-place update is required!
    - Mapping table + GC + space allocation
Existing Solutions

• Indirection system.
  • S-block: 2000 contiguous LBAs
  • E-region + I-region
  • GC types: E-region GC, E-to I destage, I-region GC
  • Drawback: It performs bad under constant workloads

• Track refreshment.
  • E-region + I-region
  • No E-region GC
  • Background GC: contiguous track refresh in the I-region
  • Drawback: significant track refresh overhead (e.g., 100X)
  • Unrealistic assumption: all tracks have the same capacity.
Proposed Solution

• T-STL
  • a Track-based Shingled Translation Layer for autonomous Shingled Write Disks (SWDs)
  • Large band (100 tracks or 200 tracks per band)

• T-STL components:
  • Track level mapping scheme: an 8TB SWD creates 64MB mapping table.
  • Space management scheme: track allocation + GC
    • Greedy scheme: simple
    • Smart scheme: automatic cold data progression

• Two special SMR properties are identified and utilized
  • When space utilization is less than 50%, SWD can use alternate tracks only to emulate a regular HDD.
  • When a used track is invalidated, it becomes free and can be reused as long as its next track is free too.
1. RAZ is used for efficient metadata access.
2. Practically, SWD shingled space is divided into several zones and tracks in the same zone has the same size.
3. Tracks are further organized into bands. T-10 standard suggests 256 MB for host-aware/host-managed SWDs.
When Space Usage <= 50% 

• SWD can use a track-level mapping table to keep track of the track allocation (logical track to physical track mapping)

• Used tracks are protected by free tracks and thus can be updated freely

• No write amplification overhead
When Space Usage > 50%

• Aggressive update strategy:
  • if a track whose next track is free, update in place.
  • else: copy-on-write

• Track status:
  • Free track (usable free vs. unusable free)
    • Invalidated tracks naturally become free tracks.
  • Used track

• How to best describe the track information?
  • “Space element” is defined.
  • These track information will be used for space management scheme.
Space Element

- **Free space**: [4, 5], [7, 8, 9], [13], [18, 19], [23, 24]
- **Used space**: [0, 1, 2, 3], [6], [10, 11, 12], [14, 15, 16, 17], [20, 21]

* Element Size = # tracks inside an element

* Last track property:
  - Last track of a free space element cannot be written
  - Last track of a used space element can be updated freely
Track Allocation

**Greedy**
- On update:
  - Search starting from the updated track(s) in a greedy manner to find the nearest free space element with a similar size to the request size
  - If a request is too large, use multiple free space elements
- On new writes:
  - Search starting from the current head position.

**Smart**
- Preallocate a free space pool, essential it is the largest free space element.
- On writes and updates:
  - Write to the track inside the free space pool pointed by the “bookmark pointer”
  - Increment bookmark pointer accordingly.
Garbage Collection

**Greedy**
- Search starting with current head position for the nearest used space element, whose size is smaller than threshold $W$ and append to its nearest neighbor.
- $W$ will be doubled if search fails.
- $W$ is initialized to $W = \left\lfloor \frac{U}{1-U} \right\rfloor$, where $0 < U < 1$. 

**Smart**
- Search in a similar way as the Greedy algorithm.
- However, the preallocated space pool is skipped.
- Victim or selected used space element is always appended in the leftmost free space element with a similar (larger) size.
- Victim elements contain statistically "cold" data.
GC Trigger

• A free space fragmentation ratio (R) is defined to help decide when to trigger a GC operation
  • GC is a foreground operation.

\[ R = \frac{F - N}{F}, \text{where } 1 \leq N \leq F \]

• F: the total number of free tracks in a selected band
• N: the total number of free space elements.
• R=0: very fragmented
• R=max: contiguous free space
• R=0.5 used in our experiments as a trigger.
Experiment Setup

Table 1: Tested Schemes

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Type</th>
<th>Effective Capacity</th>
<th>Band Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDD</td>
<td>HDD</td>
<td>140GB</td>
<td>N/A</td>
</tr>
<tr>
<td>Greedy</td>
<td>O-SWD</td>
<td>140GB</td>
<td>100</td>
</tr>
<tr>
<td>Smart</td>
<td>O-SWD</td>
<td>140GB</td>
<td>100</td>
</tr>
<tr>
<td>R(4123)</td>
<td>I-SWD</td>
<td>112GB</td>
<td>4</td>
</tr>
<tr>
<td>IS</td>
<td>O-SWD</td>
<td>140GB</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 2: Trace Statistics

<table>
<thead>
<tr>
<th>Trace</th>
<th>I.A.T. (ms)</th>
<th>Average B.D.</th>
<th>MAX LBA</th>
<th>Average R.S.</th>
<th>Write Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>mds_0</td>
<td>499.41</td>
<td>968339427.25</td>
<td>36417159175</td>
<td>18</td>
<td>0.8811</td>
</tr>
<tr>
<td>usr_0</td>
<td>270.25</td>
<td>1682254833.62</td>
<td>17077784583</td>
<td>45</td>
<td>0.5958</td>
</tr>
<tr>
<td>stg_0</td>
<td>297.79</td>
<td>1332575418.89</td>
<td>11612643455</td>
<td>23</td>
<td>0.8481</td>
</tr>
<tr>
<td>web_1</td>
<td>3757.39</td>
<td>2627690359.97</td>
<td>72831761927</td>
<td>58</td>
<td>0.4589</td>
</tr>
</tbody>
</table>

- Experiment is trace based simulation.
- Different schemes, traces and space usages are used.
1. Response time, Gross write time breakdown are used as measurement metrics.
2. Another workload (SYN) is used:
   • Emulate backup and archive workload
   • Sequential write.
## Experiment Conclusion

<table>
<thead>
<tr>
<th>Schemes</th>
<th>E-region?</th>
<th>Performance</th>
<th>Sustainability</th>
<th>Metadata Overhead</th>
<th>Space Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDD</td>
<td>no</td>
<td>best</td>
<td>best</td>
<td>no</td>
<td>best</td>
</tr>
<tr>
<td>Smart</td>
<td>no</td>
<td>good</td>
<td>good</td>
<td>low</td>
<td>good</td>
</tr>
<tr>
<td>O-SWD</td>
<td>no</td>
<td>good</td>
<td>good</td>
<td>low</td>
<td>good</td>
</tr>
<tr>
<td>I-SWD</td>
<td>no</td>
<td>good</td>
<td>fair</td>
<td>no</td>
<td>fair</td>
</tr>
<tr>
<td>IS</td>
<td>yes</td>
<td>not good</td>
<td>not good</td>
<td>high</td>
<td>good</td>
</tr>
</tbody>
</table>
Evaluating Host Aware SMR Drives
Scope and Objective

• New storage devices always bring benefit, introduce challenge and call for innovations.
• Our research investigates a new type of disk drive: Host Aware SMR drives.
• Understand the I/O performance feature, and propose I/O solutions for it.
Outline

• Scope & Objective
• Host Aware SMR (HA-SMR) Drive background
• Roadmap
• Testing Results and Lessons Learned
• Host-controlled Indirection Buffer (H-Buffer)
• Related Work
• Summary & Future Work
Shingled Magnetic Recording (SMR)

Shingled Magnetic Recording:
+ enables higher data density by overlapping data tracks. Lower $/GB!
- requires careful data handling when updating old blocks. I/O stack change
SMR Prefers Append-only Writing

Append-only Write

Read-Modify-Write
Guard Tracks

Enables independent Read-Modify-Write
Three SMR Drive Models

- **Drive Managed**
  - Black Box
  - Used as normal non-shingled drive
  - SMR data handling: by the drive

- **Host Managed**
  - White Box
  - Suggest where to write
  - Reject out-of-order write operation

- **Host Aware**
  - White Box
  - Suggest where to write
  - Accept out-of-order write operation

**White box for SMR: zoned block device**

- **zones**
- write pointer (wp)
- wp
- wp
More HA-SMR Drive Terminologies

Conventional Zones (optional)

Write Pointer Zones

outer track

inner track

zones

write pointer (wp)

wp

wp

sequential write

1 2

write pointer (wp)

wp

wp

outer track

inner track
HA-SMR Drive Background

Conventional Zones (optional)  

Media Cache

Write Pointer Zones

write pointer (wp)

inner track

outer track

non-sequential write

1

2

write pointer (wp)

wp

wp

wp

inner track

outer track
We focus on Host Aware SMR (HA-SMR) drives

- Ultimate goal: building low-cost large-scale storage system using HA-SMR drives.
- HA-SMR is the superset of the two other SMR models (Host Managed and Drive Managed) in functionality.

Roadmap:

- Understand intrinsic features of HA-SMR drives by performance characterization;
- Summarize the lesson learned and system implications from the performance testing; and
- Explore a Host-controlled indirection Buffer (H-Buffer) to meet the system design challenge of HA-SMR by exploit the special characteristics of it.
Characteristic for Traditional Disk Drives

- Rotational Speed, Seek time, etc.
- Disk buffer size..
- ...

But we focus on the features that does not exist in the traditional HDDs!
Characterization for the HA-SMR Unique Feature

- **Open zone issue**
  - A zone must be opened, before we write data to it. Recommended maximum #: 128

- **Non-sequential written zone issue**
  - Recommended maximum non-sequentially written zone #: 8

- **Media cache cleaning efficiency**
  - The relative relation between workload intensity and band cleaning time become the key point to understand non-sequential workload performance.
Characterization Goals

- **Open zone issue**
  - A zone must be opened, before we write data to it. Recommended maximum #: 128

- **Under the hood:**
  - Opening a zone will reserve “open zone resource” that will persistent the zone metadata through unexpected power loss.
  - When all the “open zone resource” is used up, opening a new zone will result closing an old zone, incurring expensive disk synchronization operation.
Test Environment Setup

- **trace**: micro-benchmark trace
- **fio**: replay tool
- **libzbc**: user space ZBC library developed by HGST (branch r04)
- **SMR**: Seagate HA-SMR sample drive
  - model: ST8000AS0022, prototype firmware revision ZN03
Open Zone Issue Investigation

Sequential Write

Non-sequential Write
Open Zone Issues Results

• Recommended optimal number: 128.
• 1000 cycles, 4KB iosize.
• Clear performance drop for sequential write (we have proposed a solutions by H-Buffer).
• No significant drop in non-sequential write.
• Sequential write does not always perform better than non-sequential write.
Testing both Large and Small IO Sizes

sequential, iosize=512K

non-seq, iosize=512K

sequential, iosize=4K

non-seq, iosize=4K
Lesson Learned

• Designers should always respect the open zone recommendation for sequential write.
• HA-SMR can do well in light-weight bursty non-sequential workload.
Characterization for the HA-SMR Unique Feature

• Open zone issue
  • A zone must be opened, before we write data to it. Recommended maximum #: 128

• Non-sequentially written zone issue
  • Recommended maximum non-sequentially written zone #: 8

• Media cache cleaning efficiency
  • The relative relation between workload intensity and band cleaning time become the key point to understand non-sequential workload performance.
Non-sequential Zone Issue Investigation

• Recommended optimal number: 8.
• Perform non-sequential write (iosize: 256KB) to a set of zones for 2 hours.
• Blocking media cache cleaning makes the throughput drops three orders of magnitude.
• Larger number of non-sequential zones leads to a longer time for the throughput recover.
Lesson Learned

• Application designers for HA-SMR drives should keep the non-sequential zone as few as possible to avoid this severe performance degradation.

• HA-SMR will fit well in an hierarchical storage architecture as the second tier storage where the first tier – possibly Flash – can filter out most of the non-sequential write requests.
Characterization for the HA-SMR Unique Feature

• Open zone issue
  • A zone must be opened, before we write data to it. Recommended maximum #: 128

• Non-sequential written zone issue
  • Recommended maximum non-sequentially written zone #: 8

• Media cache cleaning efficiency
  • The relative relation between workload intensity and band cleaning time become the key point to understand non-sequential workload performance.
Media Cache Cleaning Efficiency

- We noticed that the average band cleaning time varies widely.
- However an accurate estimation of the band cleaning time is crucial
  - for designers to predict the media cache cleaning efficiency and the non-sequential write performance.
- Therefore in this section we investigate how different factors would affect the average band cleaning time.
- We hypothesize that the band cleaning time depends on workload characteristics such as zone coverage, update ratio, I/O request offset, I/O request number, I/O request size, Zone distribution in Media Cache, etc.
Media Cache Cleaning:
-- Update ratio, I/O Request Offset

Issue shuffled write into the beginning 20% of each zone. And collect the avg cleaning time.

zones

- 20%, 40%, 60%, 80%, 100%
- beginning, end, middle and random
- Empty zone, full zone.

combination
Media Cache Cleaning:
-- Update ratio, I/O Request Offset

• Avg cleaning time is positively correlated to the **update ratio**.
• It also depends on I/O request offset and empty/full zone, actually on how much extra data is being moved.
Media Cache Cleaning: -- I/O request number, I/O request size

Issue shuffled write into the beginning 10%, 20%, 40%, 80% of each zone. And collect the avg cleaning time.

- Increase I/O size
- Increase I/O number
Media Cache Cleaning: -- I/O request number, I/O request size

• Avg cleaning time is positively correlated to the I/O request number and I/O request size.

• I/O request number contributes more to the cleaning time than I/O request size does, even the total data size is the same.
Lesson Learned

• Generally, the cleaning time is positive correlated with the total amount of data to be read/write and number of operations to be migrated.

• Between the two, the number of operations contribute more to the cleaning time.

• This will give us a hint that larger non-sequential I/O size is favorable in a sense that the cleaning time will be shorter.
Host-controlled Indirect Buffer (H-Buffer) Motivation

• Design a software layer above HA-SMR which reorganizes the workload into HA-SMR friendly ones to enhance the I/O performance.

• Exploit HA-SMR model for the design.
Data Handling in HA-SMR Drives

1. Sequential Write: direct path

2. Non-Seq Write: indirect path by drive-controlled media cache

3. Seq & non-Seq Write: indirect path by host-controlled indirection buffer

**Host Managed**
- Only HA-SMR supports such a three-data-path system
Three-data-path System

- H-Buffer embodies a HA-SMR three-data-path system (the mechanism)
- It supports a broad spectrum of workload switching algorithm (the policy)
- We believe such separation of mechanism and policy can eventually lead to various solutions to the performance degradation problem.
Case Study: Open Zone Issue

- workload: 200 concurrent streams
- policy off: write directly
- policy on: write to h-buf, then migrate to write pointer zones

H-Buffer reduces I/O time

We explore a simple case study (open zone issue) to demonstrate the potential of H-Buffer. A complete design for H-Buffer is left as future work.
## Related Work

<table>
<thead>
<tr>
<th></th>
<th>Skylight*</th>
<th>Our Work</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objective</strong></td>
<td>uncover the internal structure of DM-SMR</td>
<td>We focus on the unique features in HA-SMR such as the open-zone issue, and how to meet the system design challenge by exploring an H-Buffer concept.</td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>Software + hardware (high speed camera)</td>
<td>Leverage richer libzbc API to manipulate the drive and more information (non-sequential zone number, wp position, etc.) to aid the evaluation.</td>
</tr>
</tbody>
</table>

*A. Aghayev and P. Desnoyers, FAST’15*
T10 ZBC Standard

• New SCSI command set – ZBC (Zoned Block device Command set)
• Standardized by T10 (the SCSI technical committee)
• Ideal for SMR drives
# T10 SMR Drive Models

<table>
<thead>
<tr>
<th></th>
<th>Drive Managed SMR Drive</th>
<th>Host Aware SMR Drive*</th>
<th>Host Managed SMR Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Conventional zone</strong></td>
<td>Mandatory</td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>Seq. write pref. zone</strong></td>
<td>Not supported</td>
<td>Mandatory</td>
<td>Not supported</td>
</tr>
<tr>
<td><strong>Seq. write req. zone</strong></td>
<td>Not supported</td>
<td>Not supported</td>
<td>Mandatory</td>
</tr>
</tbody>
</table>

*Our investigation has more focus on Host-Aware SMR Drive (highlighted by red square)*
HA-SMR Features

- Media cache is used for non-sequential writes
  - Non-sequential write will go to the media cache
  - Later the non-sequential written data will be migrated to the home zone by media cache cleaning (in a Read-Modify-Write fashion)

- Open zone Issue
  - A zone must be opened, before we write data to it. Recommended maximum #: 128

- Non-sequential written zone issue
  - Recommended maximum #: 8

Characterization Goals

- Open zone Issue
- Non-sequential zone Issue
- Media cache cleaning efficiency
What’s Special for Host Aware (HA) Model?

- HA is Superset of both Host Controlled model and Drive Managed model. HA model is:
  - Equipped with media cache, accepting non-sequential write (like Drive Managed model); and
  - Exposing zone write pointer information, supporting accurate host-control (like Host Managed model).
Media Cache Cleaning: -- Zone distribution in Media Cache
Media Cache Cleaning:
-- Zone distribution in Media Cache

• Interleaving zones in the media cache will have shorter idle cleaning time.
• The reason is the drive starts the cleaning before the disk is idle. However, it is not as intrusive that blocks the I/O.
• There is not significant benefit to have one over the other.