Reflecting on the Past 17 Years of Shingled Magnetic Recording for Insights Into Future Disk Transitions: A Survey

BABAR KHAN, Embedded Systems and Applications Group, Computer Science, Technical University of Darmstadt, Germany
ANDREAS KOCH, Embedded Systems and Applications Group, Computer Science, Technical University of Darmstadt, Germany

Shingled magnetic recording (SMR) is a data storage recording technology used in modern hard disk drives (HDDs) to increase the areal density capacity (ADC) of underlying media. The research on SMR drives began around 2008, with the first SMR disk entering the market in 2013. We have performed an extensive survey on SMR research, encompassing over 100 scientific research articles spanning nearly 17 years. Our survey offers an in-depth analysis of the evolution of SMR disks, examining the different types of SMR architectures and the inherent performance challenges in existing SMR disks. We have also explored how SMR technology integrates with data storage solutions like RAID and Deduplication, including an examination of real-world use cases where hyperscalers have successfully leveraged SMR for large-scale data management. Furthermore, as storage demands continue to escalate, there is a notable shift from various HDD technologies toward Heat-Assisted Magnetic Recording (HAMR) disks, offering potential for increased storage densities beyond 1.5 $Tbit/in^2$. To this end, our survey also briefly discusses the research contributing to two specific HAMR variants such as Shingled-HAMR and Heat Interlaced Magnetic Recording (HIMR).

CCS Concepts: • **Information systems**; • **Hardware** → **External storage**; • **General and reference** → **Surveys and overviews**;

Additional Key Words and Phrases: Taxonomy, shingle translation layer, RAID, deduplication, hyperscalers, HAMR, HDMR

ACM Reference Format:
Babar Khan and Andreas Koch. 2025. Reflecting on the Past 17 Years of Shingled Magnetic Recording for Insights Into Future Disk Transitions: A Survey. *ACM Trans. Storage* 21, 3, Article 22 (June 2025), 50 pages. https://doi.org/10.1145/3731453

This work was supported by the German Federal Ministry for Education and Research (BMBF) for project SODDAS with the funding ID 01 IS 19018 B.
Authors' Contact Information: Babar Khan (Corresponding author), Embedded Systems and Applications Group, Computer Science, Technical University of Darmstadt, Darmstadt, Hessen, Germany; e-mail: khan@esa.tu-darmstadt.de; Andreas Koch, Embedded Systems and Applications Group, Computer Science, Technical University of Darmstadt, Darmstadt, Hessen, Germany; e-mail: koch@esa.tu-darmstadt.de.

ACM 1553-3077/2025/06-ART22
https://doi.org/10.1145/3731453

1 Introduction

The magnetic recording field has continuously developed over several decades, tracing back to the pioneering work of Oberlin Smith, who conceptualized a magnetic recording apparatus in 1878 [47, 62]. Fast forward, in 1947, researchers at Manchester University invented the first high-speed, cathode ray tube based fully electronic memory. In 1950, the U.S. company **Engineering Research Associates** (**ERA**) developed magnetic drum memory. In 1951, the first tape storage device, known as UNISERVO, was invented. By 1953, researchers had developed magnetic core memory. The first hard disk drive (hereafter HDD), RAMAC (random access method of accounting and control), was developed for the IBM 350 computer in 1957. Since then, HDDs have dominated the storage market for several decades and continue to do so [19, 74, 119] because of its low *cost-per-gigabyte* and *low bit-error-rate* [38, 75, 153, 231]. By 2024, due to their low cost per terabyte, HDDs are expected to dominate over half of the world's data storage. Based on some estimates, in 2023, approximately 330 million terabytes of data were created each day. It is even more evident considering that the amount of data generated worldwide continues to increase progressively and is estimated to reach 175 ZB in 2025, according to IDC [48]. The research involved in building an HDD is interdisciplinary, including computer science, signal processing and coding, electro-magnetics, materials science and engineering, magnetism, microfabrication, electronics, control systems, heat transfer, and so on [248]. Consequently, the systems research in computer science on nearly every new HDD has also been influenced by this diversification in manufacturing processes. Among various metrics in manufacturing processes, *areal density* has remained one of the crucial metrics that has steered the research in systems research regarding HDD. Areal density measures the quantity of information bits that can be stored on a given length of track, area of surface, or in a given volume of a computer storage medium.

Figure 1 shows the areal density versus time since the original IBM RAMAC. The vertical axis corresponds to the areal density in *Terabit per square inch* ($Tbit/in^2$), and the horizontal axis represents the levels of technology readiness in terms of *Years*. The graph illustrates a selection of HDDs that employed conventional recording technology from 1960 to 2000. From 2000 onward, the graph depicts the emergence of various magnetic recording technologies diverging from conventional recording.

The storage density of an area of a disk surface in a RAMAC was only $2kbit/in^2$. And in the year 2003, it was already 60 $Gbit/in^2$, a 30 million-fold increase over the past 46 years [248]. In 2003, a 3.5″ HDD had a capacity of 160 GB, capable of storing nearly a thousand times more data than an HDD of the same size just ten years ago [248]. Notably, the increase in disk density in recent years has largely stemmed from advancements in track density ***tracks per inch*** (**TPI**) rather than just linear density improvements. Estimates suggest that track density improvements have contributed about 60% to 80% of the increases in areal density in recent HDD generations. A technical report from 2012 [194] noted that, despite significant efforts to miniaturize components, the minimum width of a track appeared to stabilize at approximately 50 nm [72]. However, some industry sources reported in 2011 having observed track widths as low as 30 nm [64].

As shown in Figure 1, the sharp trajectory from 1990 onward became feasible with the advent of **Very Large-Scale Integration** (**VLSI**), enabling the attainment of high-performance data channels, servo channels, and attachment electronics within a compact form factor [58, 199]. A crucial aspect of this advancement was the introduction of lithographic manufacturing techniques for disk heads. These integrated circuit-like methods allowed for the production of heads that are only a few tens of nanometers wide, enhancing the precision and efficiency of data storage.

While gate density has increased for integrated circuits, bit density has also increased at a similar rate or even more. Hence, as depicted in Figure 1, the HDD has followed its own version of Moore's Law, known as *Kryder's Law* [110, 209], for decades.

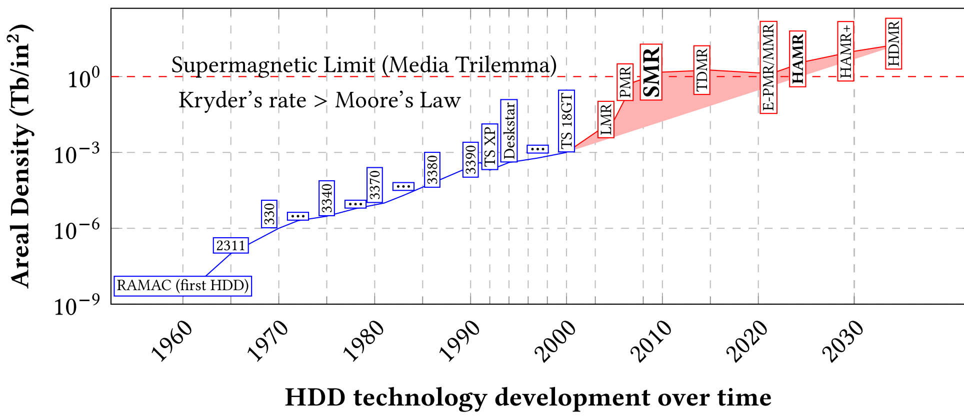

Fig. 1. Graph illustrating progression of HDD Areal Density over the years, approaching 1 Tb/in^2 and beyond.

Kryder's Law has been a critical concept in understanding the evolution of HDD technology. Named after Mark Kryder, it posits that the density of information stored on magnetic disks has been increasing exponentially, doubling approximately every 18 months. This trend parallels Moore's Law in semiconductor technology but is uniquely focused on magnetic storage systems. Kryder observed that since the introduction of the disk drive in 1956, the density of information it can record has swelled from a paltry 2,000 bits to 100 billion bits (gigabits), all crowded in the small space of square inch. That represents a 50-million-fold increase. Not even a Moore's silicon chip can boost that kind of progress.

Around 2000, it was already predicted that the physical scaling processes utilized thus far would soon be unable to provide substantial additional capacity gains [199]. This suggested that **conventional magnetic recording (CMR)**[1] disks have already attained the upper limit of their storage density. And the rate of improvement in areal density (and consequently, cost per bit) will level off over the next decade. This was termed *superparamagnetic limit* or ***superparamagnetic effect (SPE)*** (hereafter SPE) [199, 213]. Therefore, around the year 2000, the SPE began to become a significant concern for new HDDs as data densities increased and the size of magnetic grains decreased. The SPE is fundamentally a tradeoff between *three* competing parameters: the media SNR, the write-ability of the media, and the thermal stability of the media. This tradeoff was first coined as the *media trilemma* in [23]. As manufacturers attempted to increase areal density by narrowing track widths, they had to reduce the width of the write head. This reduction decreased the strength of the magnetic field generated by the write head, necessitating the use of *softer* (lower coercivity) magnetic media, which could be magnetized by this weaker field.

Around the year 2006, the disk drives stored 400 $Gbit/in^2$, and were again reported to be rapidly approaching the density limit imposed by the SPE for perpendicular recording, estimated to be about 1 $Tbit/in^2$ [176]. In the year 2010, areal densities of over 800Gb/in^2 have been demonstrated in the lab using perpendicular recording, and the highest areal density shipping in a product was 739Gb/in^2 [55]. Around the same time, i.e., 2009–2010, it was mentioned that with hybrid disk technology may be the goal of 1 $Tbit/in^2$ [152] can be increased to 4Tb/sq [71, 110, 127]. Here, the term *hybrid* meant a new technology with minimal changes to current CMR disks. The rationale behind the proposal was that while the SPE limit poses a significant challenge, the areal densities of drives, around 500 $Gbit/in^2$, are still distant from fundamental limits. This new

[1]CMR also stands for Connectional Magnetic Recording.

technology was termed as **Shingled Magnetic Recording** (**SMR**). And the first SMR disk, also termed as **Shingled Write Disk** (**SWD**) was released in year 2013.

The Figure 1 illustrates that HDDs are presently undergoing a transition from **Two-Dimensional Magnetic Recording** (**TDMR**) to Enhanced-PMR, **Microwave Assisted Magnetic Recording** (**MAMR**), **Heat Assisted Magnetic Recording** (**HAMR**), and **Heated Dot Magnetic Recording** (**HDMR**) [12, 44, 60, 109, 111, 136, 178]. Among these newly emerging disk technologies, there will be variations of SMR in one form or another. Therefore, it is crucial to thoroughly investigate the microarchitectural details of SMR disks from a systems research perspective. Furthermore, to the best of our knowledge, there has been no prior work to date that has examined the significant advancements in the SMR domain while integrating various important research contributions that have contributed to SMR research over the last 17 years. We have performed an extensive study in the area of SMR research encompassing spanning nearly 17 years. This work offers a much needed analysis of the evolution of SMR disks, encompassing both the different types of SMR architectures and the inherent performance challenges in existing SMR disks. Furthermore, as storage demands continue to escalate, there is a notable shift from various HDD technologies toward HAMR disks, offering the potential for increased storage densities beyond 1.5 $Tbit/in^2$. To this end, our survey will also briefly discuss the research that contributes to two specific HAMR variants i.e., Shingled-HAMR and **Heat-Assisted Interlaced Magnetic Recording** (**HIMR**). Next, we will briefly describe the main SMR architecture and elaborate further on the structure of our survey.

The remainder of this article is organized as follows: Section 2 provides an overview of SMR architecture, detailing its fundamental principles and key components. Section 3 reviews foundational research into SMR, focusing on seminal approaches to band organization and garbage collection methodologies. Section 4, Section 5, and Section 6 explore the three primary types of SMR: **Drive-Managed SMR** (**DM-SMR**), **Host-Managed SMR** (**HM-SMR**), and **Host-Aware SMR** (**HA-SMR**), examining their respective design strategies and operational characteristics. Section 8 discusses the integration of SMR with RAID, data deduplication techniques, and its practical applications in real-world storage systems. Section 9 synthesizes lessons learned from SMR deployments and investigates how the technology can be extended to future advancements. Finally, Section 10 outlines the prospects for SMR technology and presents concluding remarks.

2 Shingled Magnetic Recording (SMR) Architecture

Figure 2 illustrates the microarchitecture of a SMR disk compared to a CMR, also known as PMR, disk. Additionally, Figure 2 also shows the *three* main types of SMR drives: DM-SMR, HM-SMR and HA-SMR (discussed in detail in later sections: Section 4, Section 6, and Section 5). A command specification has been established for SMR hard disks implementing both the HM-SMR and HA-SMR drives [52, 95]. These standardized command interfaces have been formulated by the INCITS T10 [52] committee for SCSI hard disks and by the INCITS T13 [95] committee for ATA hard disks. However, no industry standard has been established for the Drive-Managed model, as it prioritizes backward compatibility and transparency to hosts.

In CMR, data is written onto magnetic tracks that are arranged side-by-side with small gaps between them, allowing each track to be independently rewritten without affecting adjacent tracks, thus enabling efficient random write operations. In contrast, SMR takes advantage of the difference in magnetic field strengths required for reading and writing. Since the magnetic field needed for reading is smaller than that needed for writing, the read track can be narrower than the write track. Specifically, the magnetic write head is typically 3–8 times larger than the read head [130, 143]. This configuration allows each data track to be written by partially overlapping the preceding track, while leaving sufficient space for a narrower read head to access the previous track's data.

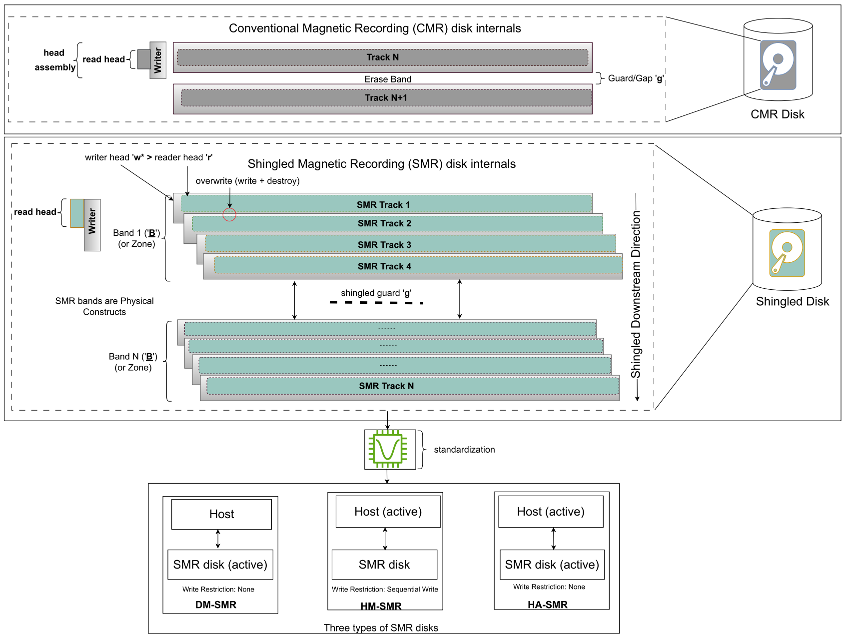

Fig. 2. The internal architecture of SMR compared to CMR disks, and the three types of SMR drives.

Physically, SMR disks organize these overlapping tracks into fixed-size ***bands*** or ***zones*** (shown in Figure 2), resembling the layered arrangement of *shingles* on a roof or *clapboards* on a wall. This shingle layout increases TPI and thereby enhances the areal density of magnetic tracks on the disk. Common band sizes in SMR disks vary; for example, 8 TB SMR disks often use a zone size of 256 MiB, while other configurations have band sizes ranging from 13–36 MiB or 15–40 MiB [4, 219]. Within each band, contiguous tracks are written, with individual track sizes typically ranging from 100 kB to 1 MB [87]. Another key parameter emphasized in earlier research was the *number of tracks* shingled together. With a shingle spacing set at 10% of the write width, 90% of the last shingle remained unused. To limit capacity overhead to 10%, each band contained around 100 shingles [65].

While SMR disks support random read operations, writing is more complex due to the overlapping track structure. Modifying any part of a band in SMR, a process known as **random modify-write** (**RMW**), requires rewriting the entire band. This approach leads to write amplification, as multiple tracks–typically 4 to 8–are affected [87, 168] during a single write operation. As a result, SMR disks are best suited for workloads that prioritize high data density and tolerate infrequent modifications to stored data.

If all tracks on an SMR device were fully shingled, the disk would essentially function as a *write-once* medium. Once data is written, reusing the tracks would be impossible without impacting adjacent, non-overwritten sectors. To achieve maximum capacity, an SMR drive could be written sequentially from beginning to end, filling all available tracks. To mitigate this limitation, ***guard regions*** (or *guard gaps*) (shown in Figure 2) are introduced between zones. These small gaps prevent data from being unintentionally overwritten when new data is written.

Guard regions ensure that the wide write head used in SMR does not interfere with the initial tracks of a zone while writing to the final track of the preceding zone, which is critical for maintaining data integrity during write operations. In one of the earlier studies, Gibson et al. [64] noted that

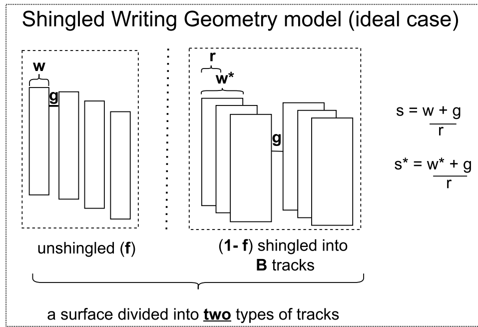

Fig. 3. Shingled Writing Geometry model for an ideal case, Gibson et al. [64].

conventional recording technology uses a track width of 25 nanometers (nm) with guard gaps of 5 nanometers (nm) between tracks. In contrast, shingled writing employs wider tracks measuring 70 nanometers (nm), with sharper edges allowing adjacent down-band tracks to be placed as close as 10 to 20 nanometers (nm) apart. This arrangement optimizes areal density while preserving track accessibility.

2.1 Inherent Challenges of SMR Technology: Balancing Increased Areal Density and Write Amplification

It is also important to understand the dual challenges presented by SMR technology, particularly regarding the significant increase in areal density and the corresponding rise in write amplification. We will discuss particular examples from various research studies that have quantified these two factors, aiming at providing a clearer understanding of the tradeoffs involved in optimizing SMR technology for modern storage demands.

2.1.1 Increase Factor of Areal Density. Since the motivation behind SMR technology was fundamentally rooted in the need for increased data storage density and capacity, the study in [64], motivated by the work in [15], introduced a *Shingling Writing Geometrical Model*. This model initially provided valuable insights into how track geometry influences areal density, highlighting the underlying mechanisms responsible for the enhanced storage capacity in SMR disks. It can be summarized as follows:

The SMR disk space is organized by multiple *bands/zones* "**B**", and a band consists of several data tracks followed by a *guard* region "**g**". In a non-shingled region of the disk, tracks of width, denoted as "**w**", are written with guard gaps, represented also by "**g**", between them. Shingled writing, characterized by wider tracks, denoted as "**w***", facilitates a more defined edge, allowing the adjacent tracks in the *down-band* to be positioned as closely as "**r**". A surface "**S**" may be divided into two types of tracks, i.e., $(1 - f)$ *shingled* and (f) *unshingled*. To this end, the *Areal Density* increase factor *(A)* can be estimated as follows:

$$\text{Increase Factor}(A) = S\left(\frac{f}{s^*} + \frac{(1-f)B}{s^* + B - 1}\right) \text{ where } s = \left(\frac{w+g}{r}\right) \text{ and } s^* = \left(\frac{w'+g}{r}\right).$$

Given it, the increase factor in areal density can be up to 2.25 times [64], showcasing the potential of shingled writing for higher storage capacity. The work in [15] reports that the combination of smaller grain size and increased track density leads to an areal density increase by a factor of at least 2.5× [195] with potential gains of 3 to 5 times, according to industry sources. Greaves et al. modeled shingled writing and identified a maximum density of 3 $Tbit/in^2$ [71], representing a threefold increase over the superparamagnetic limit of 1 $Tbit/in^2$. The work in [168] mentions that among the new technologies being explored, SMR targets an areal density increase of approximately 2.3×.

2.1.2 Quantification of Write Amplification. While increasing areal density was a primary objective, quantifying write amplification emerged as equally critical. Suresh et al. [194] investigated this relationship, particularly regarding band sizes in SMR technology. For instance, in an SMR device with a 64 MB band size, updating merely 4 KB of data necessitates reading the entire band into memory, modifying the 4 KB, and then writing back the whole band. This sequential write process effectively converts a 4 KB update into a 64 MB write, resulting in substantial write amplification. To mitigate this issue, SMR technology divides the disk surface into smaller segments called bands, which consist of consecutive tracks separated by a gap known as the *band gap*. This gap is specifically designed to prevent interference between writes to adjacent bands. Although this approach reduces write amplification to the size of the band in worst-case scenarios, it remains far from ideal. As a concrete example, even with a modest band size of 64 MB, an update of just 4 KB leads to a staggering write amplification factor of 1,638,400%, as the operation involves writing the entire 64 MB. The problem becomes even more pronounced with larger band sizes, highlighting the significant challenges in effectively managing write amplification in SMR technology.

2.2 Literature Analysis

We have performed an extensive study in the area of SMR research encompassing over 100 scientific research articles spanning nearly 17 years, sourced from leading systems and storage conferences.

Table 1 provides a comprehensive overview of SMR related research studies. Check-marks (✓) are used to indicate alignment with specific categories while the sign (✗) is used to indicate otherwise. The table is primarily classified based on the types of SMR drives: DM-SMR, HM-SMR, and HA-SMR. Additionally, we also included a relatively new SMR type called **Hybrid-SMR (H-SMR)** discussed in later sections (refer to Section 7). For brevity, the table Table 1 encompasses research studies related *only* to SMR. It excludes the work on future technologies covered in later sections (Section 9). However, a detailed bibliography is provided to ensure that readers can access a broader range of relevant literature. Since most of the SMR research studies have further categorized the DM-SMR drives into *three* subcategories (explained later in (Section 4.1, Section 4.2, and Section 4.3), our Table 1 follows the same pattern. Although the majority of the work cited in Table 1 is peer-reviewed, it's important to note that SMR research over the years also includes significant contributions from technical reports and similar sources (pioneering patents). Hence, the "Misc" category indicates work that falls into this classification. Furthermore, some of the work has been open-source, contributing immensely to SMR research. We have explicitly mentioned such contributions where applicable under the category "Code".

3 Seminal Approaches to Shingled Magnetic Recording Foundational Research: Bands and Garbage Collection Methodologies

From 2009 to 2016, extensive academic research [14, 15, 116, 169, 170, 173]–alongside critical discussions among industry practitioners [1, 9–11, 37, 45, 46, 63, 83, 83, 90, 102, 140, 156, 161, 167, 174, 201, 203], significantly influenced overall trajectory of SMR research. Figure 4 illustrates the progression toward the standardization of SMR technology, mapping its development through two primary pathways i.e., academic research and collaboration with the stakeholders in the industry. This early stage research had profound implications for practical aspects such as manufacturing processes, cost considerations, and compatibility with existing file systems. A key challenge during

Table 1. Classification of SMRResearch in Systems

Ref	DM-SMR			HM-SMR	HA-SMR	H-SMR	Source		Evaluation		Code
	O-SMR	I-SMR	hybrid				Peer	Misc	Shingled	Sim/CMR	
[31, 65, 81, 113, 128, 139, 145, 164, 189, 235] [24, 32, 33, 100, 101, 130, 144, 207, 236] [25, 36, 41, 116, 132, 137, 142, 143, 147, 148] [14, 15, 21, 77, 124, 146, 170, 181, 196, 210, 212]	✓	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗
[88, 115]	✗	✓	✗	✗	✗	✗	✓	✗	✗	✗	✗
[87] [187]	✗	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗
[3–5, 188, 192]	✓	✗	✗	✗	✗	✗	✓		✓	✗	✓
[193]	✓	✗	✗	✓	✗	✗	✓		✓		✓
[211, 227, 246]	✓						✓		✓		✗
[76, 79]	✗	✗	✗	✓	✓		✓		✓	✓	✓
[220, 224, 230, 233, 234, 240] [78, 228, 229, 232, 242]	✗	✗	✗	✗	✓	✗	✓	✗	✓	✗	✗
[29, 125, 131, 133–135]	✗	✗	✗	✗	✓	✗	✓	✗	✗	✓	✗
[85, 98, 117, 129, 171] [27, 103, 138, 217, 218, 237, 238]	✗ ✗ ✗	✗ ✗ ✗	✗ ✗ ✗	✗ ✗ ✓	✗ ✗ ✗	✗ ✗ ✗	✗ ✗ ✓	✗ ✗ ✗	✗ ✗ ✗	✗ ✗ ✓	✗
[30, 149, 150, 155]	✗	✗	✗	✓	✗	✗	✓	✗	✓	✗	✗
[239, 245]	✗	✗	✗	✓	✗	✗	✓	✗	✓	✗	✓
[64, 168]	✓	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗
[6, 50]	✓	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗
[194]	✗	✗	✗	✓	✗	✗	✗	✓	✗	✓	✗
[107]	✗	✗	✗	✓	✗	✗	✗	✓	✓	✗	✗
[221, 222]	✗	✗	✗	✗	✗	✓	✓	✗	✗	✓	✗
[184]	✗	✗	✗	✗	✗	✓	✗	✓	✗	✓	✗

Abbreviations: DM-SMR - Drive-Managed SMR; O-SMR- Out-of-place update SMR; I-SMR- In-place update SMR; HM-SMR - Host-Managed SMR; HA-SMR - Host-Aware SMR; H-SMR - Hybrid SMR; Peer - Peer Reviewed Article; Misc - Miscellaneous (Patent/Report); Eval - Evaluation; Shingled - Evaluated on real Shingled Disk; Other - CMR or Simulator; Code - source Code available;

this period was ensuring backward compatibility with legacy systems that predominantly utilized CMR technology. To address these challenges, researchers rigorously evaluated multiple proposals based on theoretical models, simulations, and empirical testing, focusing on understanding write amplification in SMR, the challenges posed by random writes in SMR, the significance of SMR band structures, and the role of garbage collection in affecting SMR disk performance. We categorize the early research on SMR into three main categories: (**1**) design space exploration of *SMR bands*, (**2**) *Garbage Collection* algorithms, and (**3**) earlier research on potential SMR *filesystems*. These research efforts led to the release of first consumer-grade DM-SMR in 2013, HM-SMR in 2014, and Linux mainline support for SMR that same year, followed by the release of HA-SMR drive in 2016.

3.1 Design Space Exploration of SMR Bands: Zones, Layout, and Workload

A **Log-structured File System** (**LFS**), first introduced in the 1990s, is designed to perform all writes to disk sequentially, structuring data in a continuous, log-like (circular buffer) manner [18, 163, 179, 186]. This design optimizes high write throughput by aggregating small write operations into larger, contiguous blocks, reducing the performance penalties associated with random writes. Building on these principles, log-structured designs offered a compelling approach for SMR disks, where sequential, log-like writes help manage SMR's inherent constraints. SMR disks increase data density by overlapping, or *shingling* disk tracks, which requires that writes proceed in a strictly sequential manner within a given area to avoid overwriting neighboring data. This sequential structure aligns closely with LFS principles, where data is written in contiguous blocks, making LFS-based designs naturally suitable for managing SMR constraints.

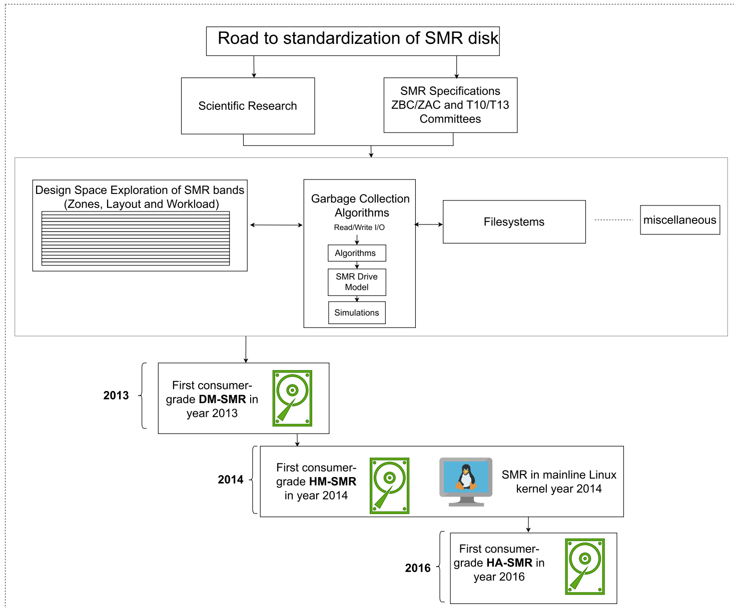

Fig. 4. Earlier Research and Road to Standardization. Key areas of focus included the design space exploration of SMR **bands**, new **garbage collection** algorithms, and the design of potential SMR-specific **filesystems.**

One of the earliest contributions to this area, presented in the form of a patent by Kasiraj et al. [105], organized the disk into bands, with each band storing a single file, such as a large multimedia file. Bands were separated by a gap of tracks to prevent a write to the last track in a band from overwriting data in the first track of the subsequent band. LFS segments and SMR bands are similar in concept, and band compaction in an SMR disk parallels segment cleaning in an LFS [101].

In two related studies conducted in 2009 and 2011, Gibson et al. [64, 65] explored the operational implications of shingled writing and suggested that shingled-write disks be managed as collections of *append-only* logs, echoing the *log-based* approach inherent in LFS. The study suggested that increase in disk density, although advantageous, introduces complexity in SMR, as altering a disk sector risks corrupting other data on the overlapped tracks, necessitating copying to prevent data loss. Instead of imposing this task on the host (or application), the SMR disk should maintain compatibility with existing CMR drives by incorporating a functionality inside the firmware that emulates the block interface. To this end, the authors mentioned that an internal software in the shingled disk controller can mask the changes imposed by **shingled-writing** (**SW**), similar to how the **Flash Translation Layer** (**FTL**) in NAND-based SSDs [8, 42, 59, 86, 118, 172] masks the need to erase a block before writing any part of it. Additionally, the work mentions the potential use of SSD technology for embedded disk cache technology, enabling hybrid solutions that combine the capacity of shingled-write with the small random read and write performance of SSDs. In case of SMR disk, the *software* is called a **Shingled Translation Layer** (hereafter **STL**).

3.1.1 Design of Bands and Zones on SMR. As illustrated in Figure 2 and Figure 3, the disk surface is divided into bands. While most of these bands are shingled, SMR technology allows for certain bands to remain unshingled, enabling random writes without the risk of overwriting adjacent data or incurring a read-modify-write penalty. To this end, early SMR research was based on the work that significantly influenced the strategies for partitioning bands on an SMR drive. We will discuss *two* specific studies that explored the partitioning of shingled and unshingled bands on SMR drives.

In their exploration of system-level design parameters for shingled write disks, Amer *et al.*[14, 15, 116, 173] delved into a spectrum of possibilities, considering alternative interfaces like object-based stores or file system-based (log-structured file systems) approaches to adapt to the new disk behavior. They discussed a partitioning scheme for shingled write disks, dividing them into *two* distinct regions: **Log Access Zones** (**LAZ**) and **Random Access Zones** (**RAZs**). A RAZ was defined by maintaining $k1$ unused tracks between each pair of tracks. The remainder of the disk, designated as the LAZ (comprising shingled tracks), remained organized into bands. The RAZ could be viewed as consisting of bands that contained a single track or as thicker bands, specifically $k + 1$ times the thickness of LAZ tracks, with the last track in the LAZ matching the width of the RAZ. RAZ can be interspersed with LAZ, allowing for the placement of metadata in a RAZ close to the data in the nearby LAZ. Additionally, using cylinders instead of tracks for forming RAZ might prove advantageous.

In addition to this, the authors in a study [194] also explored the same concept under a section titled *differences in disk geometry*. The authors discussed that another area of difference might be the location of the unshingled band on the device concerning the shingled bands. Since metadata updates are written to the unshingled partition while all file data is written to the shingled bands, the disk arm may need to move between these two types of bands more frequently than in traditional HDDs. However, optimal placement of the unshingled band could mitigate this issue. For example, positioning the unshingled band in the center of the disk surface, with shingled bands on either side, would equalize the overall displacement of the disk arm. Furthermore, coalescing multiple metadata updates into a single write on the host could further alleviate this problem.

3.1.2 Band Layout. The subsequent discussion was on band layout. The work in [15] mentioned that there are *two* basic layouts that a SMR disk may use. One option is to keep moving live data at the tail of the log to the head of the log and then considering the cleansed part free. A second option is to clean bands *atomically* in this approach, we either write complete bands or bands or append to them until they are filled. In the first case, the circular log would need an additional buffer between its head and tail of at least k tracks. We now discuss the tradeoffs involved in determining the best band usage, the best number of logs, and the reclamation of space. (**1**) *band usage*, and (**2**) *number of logs*

In case of band usage, four approaches were proposed to enhance band management: (**1**) *complete band writes*, (**2**) *append-only writes*, (**3**) *circular log*, and (**4**) *flexible band sizes*. The first approach involved writing entire bands containing segments of the LFS, which results in highly effective writes. The second approach appends data to bands without overwriting them. The third approach introduced a circular log in each band, requiring band sizes to be at least doubled compared to designs that clean entire bands atomically. To avoid overwriting active data at the head with incoming writes to the tail, an additional gap (k-track intra-band gap) is often needed between the two ends of the log. To reclaim free space, a cleaning process moves active data from the tail to the head, freeing up space by not copying it forward. However, if the log starts within k tracks of the band's beginning, this gap becomes unnecessary. This circular log design reflects a shift from Rosenblum's segmented LFS back toward Ousterhout and Douglis's original proposal, which did not partition the log into segments.

In case of number of logs, the authors mentioned that while a primarily write-by-append device might use a LFS with a single log, there are benefits to employing multiple logs. These included separating metadata from data due to differences in update frequency and optimizing read access for scenarios such as media streaming. They also discussed the use of a hierarchical log system to

Table 2. List of GC Frameworks in Earlier Stages of SMR Research

Year	Garbage Collection Frameworks in SMR
2010	*set-associative* and *S-blocks* [21]
2012	*intraband GC*, *normal GC*, and *forced GC* [128][2]
2012	*short block (E-region and I-region)* [81, 82][3]
2014	*Round Robin mapping schemes:* R(4123), 124R(3), 14R(23) [88]
2015	*empty weight (greedy)* and *cold weight* [101]

Each framework consisted of a number of GC algorithms.

improve write efficiency by managing hot and cold data, thereby reducing the need for frequent cleaning. However, the authors noted that using multiple logs comes with the drawback of increased head movement during writes, which could negatively impact performance under heavy write loads.

3.2 Compaction and Space Reclamation in SMR: Garbage Collection Algorithms

To reclaim space in SMR disks, band compaction is employed to consolidate overwritten data, much like the cleaning process in LFS [18, 101, 163, 179, 186]. This process involves identifying and relocating valid data within SMR bands to create contiguous free space for future writes. Factors such as band size and configuration can significantly affect the efficiency of compaction, as larger bands may require more extensive data relocation, increasing resource demands and compaction duration. Frequent compaction not only consumes resources but can also delay I/O operations, creating a potential bottleneck that impacts overall system responsiveness. Therefore, optimized compaction strategies are critical to minimize resource consumption and maintain system performance. Table 2 lists some of the **garbage collection** (**GC**) frameworks designed to address these challenges. A range of STL approaches is found in the patent literature [56, 57, 82, 97, 99, 179, 186].

3.2.1 Two Indirection GC Algorithms: Set-associative and S-BLOCKS (2010). Cassuto *et al.* [21] from **Hitachi Global Storage Technologies** (**HGSTs**) were among the first to address the random-write restrictions inherent in SMR disks. Their work introduced *two* indirection systems, or translation layers, comprising data structures and algorithms that map **logical block addresses** (**LBAs**) to **physical block addresses** (**PBAs**). The first (**1**) system proposed a shingled *set-associative* disk-cache-based architecture designed to enable unrestricted writes, while the second (**2**), more intricate system utilized circular buffers integrated with *S-blocks* to alleviate access limitations and enhance performance.

The disk-cache based architecture provides unrestricted access with a manageable drop in performance. It utilizes a disk cache to mask access restrictions and optimize performance in shingled recording drives. At the same time, the S-block architecture utilizes a new storage unit called S-block. It is designed to allow good sustained random-write performance through a dynamic system dividing the disk into circular write buffers with head and tail pointers. Furthermore, the work utilized a discrete-event simulation to implement and simulate the two proposed indirection systems for SMR drives. The simulator for evaluating shingled-recording architectures included a core indirection module for LBAs to PBAs and a physical drive model to provide access-time information. The outputs of the simulator include logs of PBA access, the time taken for accesses, average read/write throughput, and internal statistics on the indirection system. The results of the study provided insights into the tradeoff between capacity and performance in SMR drives. They highlighted the need for better algorithms at the firmware level to optimize performance based

[2]Most consumer grade SMR drives till to date have E-region STL

on workload conditions. Considering the impact of their work, the current DM-SMRs can also be termed *Cassuto* drives.

3.2.2 Hot Data Identification-based Shingled Write Disk (H-SWD) (2012). Although the work by Cassuto *et al.* was instrumental, like any research work, it raised some further open questions. One such question was how to mitigate the overhead of garbage collection. Further research works [81, 82, 128] addressed the performance pitfalls related to garbage collection. In this article, we propose both a novel SWD design (called H-SWD) to considerably reduce the garbage collection overhead by adopting our hot data identification mechanism and an address mapping scheme for our SWD. The H-SWD had *three* basic GC policies: **(1)** *intraband GC*, **(2)** *normal GC*, and **(3)** *forced GC*. These fundamental polices were exploited by both hot bands and cold bands. The *intra-band* GC reads a block in the tail position of a band and checks if it is valid (i.e., live) or invalid (i.e., outdated). If it is a valid block, it is moved to the head of the band. Otherwise, it is freed. The *normal* GC is a fundamental GC policy in the H-SWD design. This includes the aforementioned intra-band GC and adds one more policy: it tries to reclaim valid data blocks. The *forced* GC is a more aggressive GC policy, which is invoked only when the SMR disk cannot prepare a specified amount of free space even after the normal GC process.

3.2.3 Short Block (Large I-region and Small E-region) (2012). Hall *et al.* [81, 82] first explored an alternative strategy where SMR drive constraints are enforced autonomously, without significant modifications to the host system. They developed a data handling algorithm that achieves excellent *short-block* random write performance under SMR constraints, while still asymptotically maintaining expected random read and sequential performance. In the second part, they address the development of optimal autonomous algorithms for longer block random writes.

The authors organized the majority of HDD data into host-sequential shingled regions called **I-regions**. These **I-regions** mimic the layout of a conventional HDD but with key differences: they are optimized for efficient refresh operations under SMR constraints and include additional tracks serving as guard and refresh buffers. Although the **I-regions** store host data sequentially, the physical location of this data gradually shifts around the region as it is refreshed. An indirection system within the HDD dynamically translates host addresses to the updated physical locations. **I-regions** derive their name from this indirection mechanism. Additionally, a separate shingled region, termed the **E-region** (on-media cache), functions as a circular write log, accepting new writes without regard to logical addressing. A number of STL mechanisms have been proposed to date; however based on measurements of shipping drives, the dominant ones may be termed E-Region STLs [187].

3.2.4 Mapping Schemes (2014). In 2014, He *et al.* [88] proposed several static LBA to PBA address mapping schemes to minimize re-writes for in-place updates, assuming the file system could compact data into the lower end of the block address range, with further enhancements incorporating a track-based translation layer. The first (**1**) scheme, "R(4123),"distributes LBAs across the tracks of all bands in a *round-robin* manner: the initial 25% of LBAs map to the 4th tracks, the next 25% to the 1st tracks, followed by the 2nd and 3rd tracks, evenly distributing updates but increasing seek overhead. The second scheme (**2**), "124R(3),"maps 75% of LBAs sequentially to the first, second, and fourth tracks and the remaining 25% to the third tracks again in a *round-robin* manner, offering better spatial locality and reduced seek overhead but risking write amplification under 50% usage, highlighting a tradeoff between amplification and seek efficiency. A third variant (**3**) maps 50% of LBAs sequentially to the first and fourth tracks, the next 25% in a round-robin manner to the second tracks, and the last 25% similarly to the third tracks.

3.2.5 Empty and Cold Weight GC Algorithms. Jones *et al.* [101] highlighted that the longer a shingled disk spends in band compaction, the more resources it wastes, potentially causing I/O operations to block and reducing system responsiveness. To address this, the authors proposed developing band compaction strategies that minimize system costs. They suggested that by separating frequently updated blocks (*hot*) from those that are infrequently updated (*cold*), they can reduce the number of blocks moved over time. The authors developed a methodology that used *write heat*, defined by the frequency of writes to a logical block address (LBA), to guide band compaction. By classifying data blocks as (*hot*) or (*cold*), the methodology aims at reducing the likelihood of mixed bands, which are more expensive to compact. Hot bands are compacted less often, while cold bands, being more stable, can be compacted more frequently without leaving many holes. This classification helps decrease the number of bands read and the overall amount of data copied during the compaction process, ultimately improving efficiency in SMR media usage. To this end, they developed two algorithms: (**1**) *empty (greedy naive)* and (**2**) *cold-weight*.

The naive *empty (greedy naive)* algorithm always attempts to find an empty band for compaction. If no empty bands are available, it selects bands with the least amount of live data, compacting only the necessary number of bands based on the compaction requirement. In contrast, the *cold-weight* algorithm considers both the free space in a band and the distribution of hot and cold blocks. It selects bands based on a formula that incorporates the fraction of free space, the fraction of cold data, and the fraction of hot data. This approach helps optimize band compaction by prioritizing bands with more stable data while reducing the overall data movement required during the process.

3.3 FileSystems and Workload

Furthermore, the early-stage discussions were also aimed at developing specialized file systems and optimizing workload management to address the unique characteristics of overlapping tracks inherent to this storage technology. This section reviews two key works that have significantly contributed to the advancement of SMR file systems and workload optimization, providing insights into the challenges and innovations in this domain.

3.3.1 Filesystem (2012): In 2012, the authors [194] developed an SMR device emulator and an SMR file system. The emulator implements a subset of the Banded Device Command Set to facilitate an API for effective data management by the SMR file system. This setup enables the intelligent placement of data on the emulated device, mitigating the challenges posed by the sequential write nature of SMR. The authors explored the fundamentals of SMR, including its architectural features and performance implications. Evaluation workloads were crucial for assessing the effectiveness of the SMR infrastructure. For this purpose, the authors selected Hadoop/HDFS, a widely used workload generator.

3.3.2 Workload (2012): Le *et al.* [116, 173] evaluated the impact of diverse workloads, derived from real-world traces, on a shingled write disk model. They highlighted how different workloads influenced the strategies of remapping and restoring blocks, demonstrating that write-heavy workloads could provide a performance advantage for shingled-write disks. To ensure robust analysis, the authors developed flexible models and gathered universal performance metrics, focusing on the functional characteristics of shingled drives rather than specific physical assumptions about future disks. *Three* Key metrics included (**1**) logical block distance, which measured block movements based on the difference between sequential logical addresses, as well as (**2**) track and (**3**) band movements, reflecting the impact of head transitions on data transfer tasks.

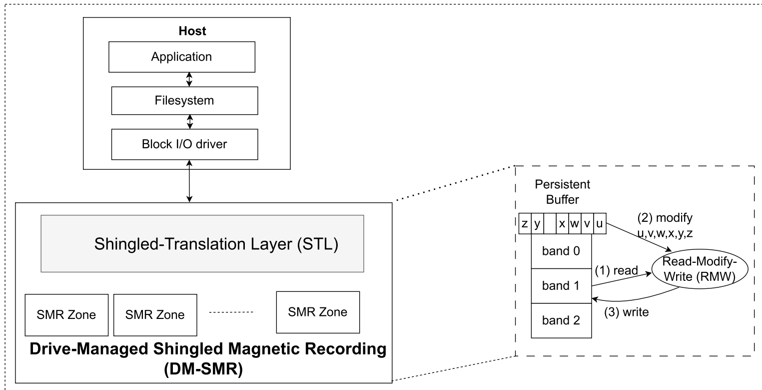

Fig. 5. DM-SMR main architecture.

4 Drive-managed Shingled Magnetic Recording (DM-SMR)

Figure 5 illustrates the fundamental architecture of **D**rive-**M**anaged SMR (hereafter DM-SMR) disk. DM-SMR drives were among the earliest types of SMR disks. A substantial portion of the earlier research discussed in the previous section culminated in the release of the first consumer-grade SMR drive, known as DM-SMR, in 2013 [182] (as also shown in the Figure 4). In DM-SMR, the drive's firmware manages the shingling process, optimizing data placement. In some research studies, DM-SMR drives have been referred to by alternative names. They are sometimes called *autonomous* disks because they shield all management complexities from the host system (and applications), independently handling all I/O operations. Additionally, DM-SMR has been labeled as *Type-1* SMR [194]. Due to its backward compatibility and transparency to hosts, the DM-SMR model does not have an established industry standard like other two types of SMR disks (discussed in Section 5, and Section 6).

As shown in Figure 5, the STL is a critical functionality of DM-SMR drives responsible for managing the mapping between logical addresses (LBA or Logical Block Address) used by the host system and physical addresses (PBA or Physical Block Addresses) on the disk platters. Since DM-SMR drives use shingled tracks where data is written in overlapping bands, the STL plays a crucial role in tracking the mapping of data within these shingled bands. It ensures that incoming data writes are efficiently organized into the appropriate shingled tracks and manages the complexities associated with shingled data placement. The STL also handles data movement and garbage collection processes to optimize storage utilization and performance. In a (DM-SMR) drive, RMW operations typically occur within the drive itself, specifically within the STL and associated firmware. Given the critical role of the STL in the overall performance of DM-SMR disks, a significant portion of DM-SMR research and seminal work has focused on experimenting with various data mapping schemes and STL algorithms.

To the best of our knowledge, there have been approximately 60 to 70 research publications focused solely on DM-SMR drives. Given the extensive nature of this research, it is impractical to discuss every study in detail. Therefore, this section is organized to highlight the most significant contributions to the development and understanding of DM-SMR drives, presenting the main ideas and observations of each work to offer a comprehensive overview. To this end, we have structured our discussion on DM-SMR drives into *six* sections as following:

The first three sections will explain the three fundamental types of DM-SMR drives. Based on how *random* writes/updates or modifications to existing data are managed within the DM-SMR

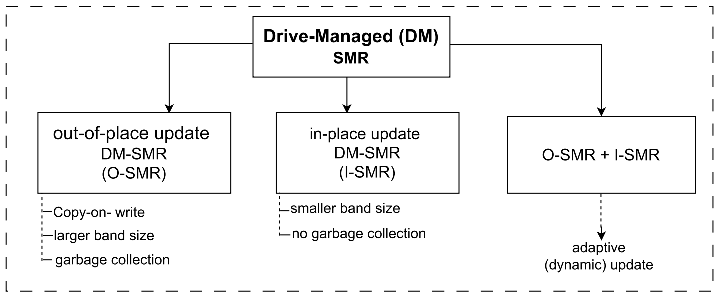

Fig. 6. Types of DM-SMR disks. These *three* terminologies for different types of DM- SMR were coined by Lin et al. [128] and He et al. [87].

drive, DM-SMR drives can be further classified into *three* [3] categories (as illustrated in Figure 6): First, the most fundamental category is ***out-of-place* update SMR (O-SMR)** drives, which are explained in Section 4.1. Second, we have ***in-place* update SMR (I-SMR)** drives, detailed in Section 4.2. Lastly, the third category is a hybrid combination of both, that is, a hybrid of **I**-SMR and **O**-SMR, explained in Section 4.3. It is important to note that these *three* terminologies for different types of DM-SMR drives were not coined by us; they were introduced by Lin et al. [128] and He et al. [87].

Furthermore, in the fourth section, Section 4.4, we will discuss the seminal research framework that reverse-engineered the internals of DM-SMR drives and laid the groundwork for various subsequent frameworks in the study of SMR drives. In the next two sections, we will provide a brief overview of the research studies related to *on-disk persistent cache* in Section 4.5 and *hybrid drives: SSDs and DM-SMR* in Section 4.6. These sections will be followed by a final discussion in Section 4.7 regarding the performance pitfalls present in existing DM-SMRs.

4.1 Out-of-place Update SMR (O-SMR)

An *out-of-place* update SMR (hereafter **O**-SMR) disk is a type of SMR device where data updates are managed by writing the updated data to a new location rather than directly overwriting the existing data. To mitigate potential performance degradation caused by random writes and updates, most SMR designs typically utilize out-of-place updates. This technique is akin to the **copy-on-write** (**COW**) method, where instead of modifying the original data, a copy is created at a new location, preserving the integrity of the existing data until the new write is confirmed. COW. Specifically, when an update is issued, the storage device invalidates the old data blocks and writes the updated data to clean (i.e., free) blocks. While this approach helps avoid performance issues associated with in-place updates, it necessitates an address mapping table to maintain the relationship between PBAs and LBAs. Additionally, the out-of-place update method results in numerous invalid data blocks on the storage device, leading to a gradual reduction in the number of available clean blocks. Consequently, a specialized reclamation mechanism, commonly referred to as garbage collection, is required to reclaim and reuse these invalid blocks. O-SMR drives provide much more space gain by using larger bands or zones. Therefore, only a negligible amount of space is used for safety gaps. This approach means that most of the earlier research we have discussed in Section 3 falls into this category [14, 15, 41, 81, 82, 87, 88, 116, 124, 128, 131, 173].

[3] these three types do not mean three types of consumer-grade drives

4.2 In-place Update SMR (I-SMR)

As mentioned in Section 4, **I**-SMR drives are slightly different from **O**-SMR drives. To execute an update operation on a previously written track in an **I**-SMR drive, it is necessary to first securely read out the data on the subsequent tracks and then rewrite them back to their original positions after updating the data on the targeted track. Notably, static LBA-to-PBA mappings are feasible in I-SMR drives, eliminating the need for mapping tables and garbage collection operations. To be more precise, no garbage collection mechanism at all.

This method stands in contrast to **O**-SMR drives, wherein updated data is written to a new location, preserving the old data until it is eventually overwritten. The work in [187] has also termed **I**-SMR drives as drives that have *track mapping STLs*, where translation is performed after mapping an LBA to a track/head/sector location. In-place SMR can present challenges in managing data updates efficiently due to the overlapping nature of data tracks on SMR disks, which resemble shingles on a roof. However, it may still be used in certain scenarios depending on the specific requirements and capabilities of the storage system. The work by He *et al.* [88] is the first work that has incorporated the **I**-SMR drive. The work has presented several new address mapping schemes for in-place update in SMR disk. By appropriately changing the order of space allocation, the new mapping schemes were implemented improve the write amplification overhead significantly. The work by Shafaei *et al.* [187] has also presented an in-place methodology. The work implements a static mapping translation layer designed for SMR drives at the track level. It facilitates in-place data writing by caching data from the next track in the shingling direction, thereby enabling direct overwriting of sectors in the target track.

4.3 In-place Update SMR (I-SMR) + Out-of-place Update SMR (O-SMR)

In a hybrid approach, combining **I**-SMR and **O**-SMR, the DM-SMR can adjust its data management scheme according to the drive's utilization changes. To the best of our knowledge there is only one research work that have implemented this hybrid design. In work [87] He *et al.* introduced an methodology termed as *SMaRT* tailored for drive-managed SMR drives, leveraging two distinctive properties inherent in SMR drives. SMaRT employs *copy-on-write* updates exclusively when in-place updates are not possible. By integrating *track-level* mapping with a novel space management scheme, the system can automatically identify and filter out cold data to minimize data migration overhead for GC operations.

4.4 Reverse Engineering and Modeling DM-SMR

The framework *Skylight* in research article by Aghayev *et al.* [3, 4] remains one of the most influential contributions. It laid a critical foundation for subsequent studies, providing a comprehensive understanding of DM-SMR disk internals. As a result, it has widely served as a benchmark for the development and evaluation of other SMR research frameworks. The framework's implementation included detailed microarchitectural modeling of the Seagate 8TB DM-SMR disk, utilizing practical tools such as a high-speed camera and slow-motion replay to meticulously examine the head assembly movements. Additionally, they developed an SMR emulator to explore the disk's behavior further. Their observations revealed an on-disk persistent cache with a *lazy* write-back policy and provided insights into the cache size and band size characteristics. The practical implications of this work have significantly advanced the understanding of SMR drives, guiding their application and development across various domains. Furthermore, in related research [5], Aghayev *et al.* presented optimizations for the ext4 file system specifically tailored to DM-SMR disks. Their study detailed how ext4's data layout mechanism operates on these disks and demonstrated the effective use of a generic kernel layer to facilitate journaling.

The researchers in two other studies [220, 234] compared *Skylight* reverse-engineered work with more modern SMR drives, such as the Seagate ST6000NM0034, focusing particularly on the performance characteristics associated with the newly introduced zoned block APIs. For instance, they analyzed the impact of zone sequential and non-sequential writes based on the number of zones in the active writing set, addressing issues related to open zones and non-sequential access patterns. Additionally, since the efficiency of *media cache* cleaning is crucial for non-sequential write performance, they examined how different workloads influence media cache cleaning efficiency.

The work by Shafei *et al.* [188] further investigated early advancements in DM-SMR disk modeling, a task complicated by the proprietary nature of the algorithms that drive their internal operations. The initial model demonstrated the capability to predict mean latency with high accuracy, maintaining an RMS cumulative latency error of 25% or less across a wide range of workloads. Despite these variations, which are non-trivial, the model's behavior was deemed realistic, falling within the variability range seen among ostensibly identical drives. This work underlined the necessity of addressing drive-specific performance intricacies, similar to conventional hard drives, where tools like DiskSim had been instrumental. It also set the stage for the creation of a validated library of SMR disk models, essential for facilitating robust system performance simulations across diverse SMR devices.

4.5 Frameworks based on Persistent Cache in DM-SMR

The following subsections describe various cache management schemes that have been proposed to efficiently handle data in DM-SMR drives, each aiming at improving performance. For the sake of brevity, we have selected five related frameworks, which are discussed in the following subsections. A more detailed understanding of various other approaches is provided in Table 1.

4.5.1 Dual-buffer [145]. The work in [145] introduced a cache management scheme called Dual-buffer to optimize the management of SMR disk space by partitioning the persistent cache into a persistent buffer and a filter buffer. Dual-buffer employs a two-level buffer cache architecture to defer frequent write-back operations of hot blocks. Leveraging this architecture, a prediction-based dynamic configuration mechanism is proposed to partition the persistent cache efficiently, utilizing a fine-grained prediction model to maximize space utilization. Additionally, a dual-buffer-aware space management and allocation strategy is presented to enhance the efficiency of data handling within the persistent cache. Experimental results indicate that this scheme can significantly reduce unnecessary read-modify-write operations and achieve improved access latency.

4.5.2 SWC^2 [31]. This article proposes SWC^2 management with two critical components: an adaptive cache allocation mechanism, which dynamically allocates cache space for frequently updated file data, and an overhead-aware cleaning method that passively reclaims cache space as needed. Additionally, it separates cached data based on their hotness to minimize cache space reclaim overhead. The SWC^2 management approach was implemented within the Linux system, emulating DM-SMR drive behavior using the device mapper layer. Experimental results demonstrated a reduction in both overall write latency and cache cleaning latency, validating the efficacy of the proposed solution.

4.5.3 KFR [146]. DM-SMR disks are known to experience high **performance recovery time** (**PRT**) due to the challenge of "SMR space reclamation." This article introduces an optimal cache management strategy called **K-Framed Reclamation** (**KFR**) to address and minimize PRT in DM-SMR disks. The motivation, design, evaluations, and analysis of KFR are thoroughly discussed. To mitigate the issue of prolonged blocking time caused by SMR reclamation processes, KFR decomposes the large-scale reclamation task into more manageable, fine-grained K-Framed reclamation processes. By employing the K-Framed Write policy, KFR significantly delays the initiation of reclamation processes, and under ideal circumstances, it can even eliminate the need for reclamation. Experimental results demonstrate that KFR effectively reduces performance recovery time by several orders of magnitude.

4.5.4 Tiler [144]. The authors in [144] introduced an approach called Tiler, which manages SMR disks by segmenting the physical space into small **autonomous regions** (**ARs**). Each AR independently handles space allocation, address mapping, and cleaning operations. By employing a log-structured management method for these ARs, Tiler effectively avoids RMWs and simplifies data updates, particularly when adjacent tracks do not contain valid data. Additionally, Tiler partitions large-scale cleaning tasks into smaller, self-contained cleaning processes, ensuring that data relocation is confined within individual ARs and thereby reducing performance overhead. Experimental results indicate that Tiler successfully minimizes overall system response time and decreases cleaning duration.

4.5.5 SAC [192]. The authors in [192] developed **SMR-Aware Co-design cache algorithm** (**SAC**). SAC employs a hardware/software co-design strategy to address the specific characteristics of SMR disks, focusing on optimizing RMW operations. It balances conflicting factors, such as data popularity versus SMR write amplification and the distribution of clean versus dirty cache space, to enhance performance. Additionally, the authors introduced the **Clean/Dirty Comparator** (**CDC**) module, a standalone cache balancer designed for SMR-based high-density disks. This module can be used alongside other cache algorithms to dynamically optimize eviction policies based on real-time workload characteristics, further boosting performance. In practical evaluations using real-world traces, SAC demonstrated significant improvements in mixed read-write scenarios compared to **Least Recently Used** (**LRU**).

4.6 SSD (NAND) Caching in DM-SMR Drives

In this section, we will discuss four studies that have focused on NAND SSD caching for SMR drives.

4.6.1 HWSR. The work in [138, 210] proposes a **hybrid wave-like shingled recording** (**HWSR**) disk system aimed at enhancing both the performance and capacity of shingled recording disks. HWSR integrates three different storage media: memory, SSD, and hard disk. The memory, designed to be a small capacity component, such as 100 MB in the proposed setup, serves as a buffer for hot writes to minimize overall cost. The SSD functions as a disk cache to boost random read performance, complementing the overall architecture and contributing to the system's efficiency. Additionally, a new LRU algorithm based on minimizing write amplification is designed to optimize overall I/O performance. HWSR has been prototyped as a stand-alone kernel module in the Linux kernel version 2.6.35.6.

4.6.2 Apas. This article [132] discusses how the general persistent cache in current SMR disk designs typically serves as a first-level cache to buffer all incoming writes across the entire SMR storage system in order to mitigate the adverse effects of RMW operations. Instead, a better approach termed as *Apas: an Application Aware Hybrid Storage System*, is proposed, where the cache functionality is redefined. Specifically, incoming data are categorized based on their write-back behavior, and data that would trigger RMWs are stored in a built-in NAND flash cache called **RMW-free Cache** (**RMW-F**) to prevent RMW operations. Furthermore, RMW-F enhances cleaning efficiency through a model that considers both write-back cost and data popularity. The experimental evaluations demonstrated that RMW-F significantly improved both system performance and cleaning efficiency.

4.6.3 PORE. The article [211] introduces a novel hardware/software co-design framework for SMR-oriented caching, termed **Partially Open Region for Eviction** (**PORE**). This framework strategically limits the LBA range of evicted data from SSD caches to enhance the overall performance of the hybrid storage system. By optimizing the tradeoff between the popularity of cached data and the write amplification associated with SMR, PORE effectively addresses the challenges inherent in hybrid storage architectures. Experimental evaluations conducted using real-world traces demonstrate the efficacy of the proposed framework.

4.6.4 Flash-cache (FC). The proposed approach in [147, 148] addresses the challenge of minimizing unnecessary RMW operations in **embedded flash with SMR** (**EF-SMR**) disk storage systems. The authors developed novel a translation layernamed MU-RMW, specifically designed for EF-SMR disks. By implementing concentrated address mapping and a lazy RMW reclamation strategy, MU-RMW significantly reduces the frequency of unnecessary RMW operations.

4.7 Performance Pitfalls in DM-SMR: Unpredictability and Black Box

4.7.1 Unpredictability in DM-SMRs. Considering that DM-SMR drives have been the first implementation of SMR technology, incorporating STL to address the complexities of rewriting data on overlapping tracks, our first analysis will predominantly center on performance dynamics, particularly regarding the difference between STL and FTL (NAND flash). Because synergy expected between DM-SMR's STL and FTL has yielded notable performance implications. The work in [41] has very aptly discussed it. A notable distinction between flash memory and SMR drives is their treatment of seek latency during read or write operations. Flash memory does not incur seek latency during read or write operations, whereas the seek latency factor significantly impacts the performance of the STL in SMR drives. Moreover, the constraints that govern the FTL and STL vary significantly. While SMR drives necessitate sequential-write constraints to prevent overwriting and eliminate the need for erase operations, the FTL must adhere to erase-before-write constraints and consider asymmetric program/erase units. Furthermore, the work in [77, 79] has also discussed the performance implication due to cleaning operations in STL layer. Cleaning operations in SMR disks involve significantly larger units compared to flash memory (256 MB zones as compared to 2-16 MB erase units), equating to 1-2 seconds of transfer time for writing and an equivalent time for reading. These cleaning operations may necessitate reading or writing a 256 MiB zone of data three or more times, leading to potentially lengthy delays.

4.7.2 DM-SMR Black Box Model. The DM-SMR drive operates with a *black-box model*, lacking the standardized command [52, 95] set found in other SMR drives. This discrepancy poses a significant challenge for system engineers aiming at optimizing the performance of DM-SMR drives within storage systems. The issue becomes particularly noticeable in scenarios where the DM-SMR drive struggles to differentiate between metadata and user data. Given that metadata undergoes more frequent updates than user data, it is crucial for storage systems to handle metadata with care to prevent write amplification problems. Unlike SSDs, which can store metadata in per-page out-of-band regions, modern disks adhere to a fixed sector size of 4 KB, restricting per-sector metadata access to low-level drive firmware. Furthermore, recent research, as illustrated in [108], emphasizes the variability among disks, even those of the same make and model. This underscores the transition from uniform sets of disks to the inherent heterogeneity of multi-disk systems. Manufacturers are increasingly adopting adaptive zoning techniques, leveraging manufacturing variations to configure per-head zone arrangements and resulting in distinct performance characteristics for each disk. The observed heterogeneity in disk performance has notable implications for DM-SMR drives, necessitating tailored approaches to address their unique operational challenges.

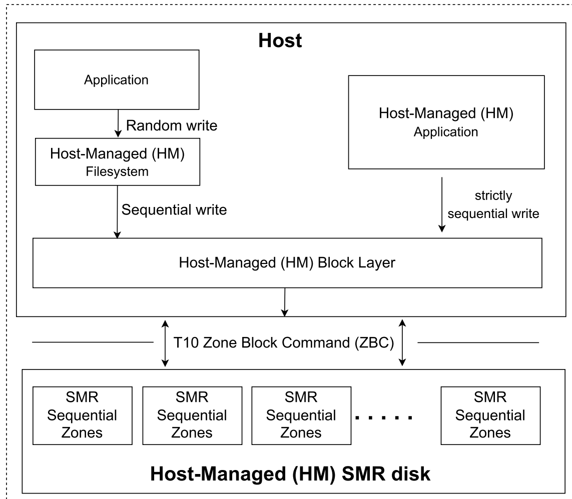

Fig. 7. HM-SMR main architecture.

5 Pushing the Intelligence up: Host-managed Shingled Magnetic Recording (HM-SMR)

Research into Host-Managed Shingled Magnetic Recording (hereafter HM-SMR) disks began around 2012 onwards [30, 85, 103, 129, 149, 150, 155, 171, 218, 237–239]. The first consumer-grade HM-SMR disk was indeed released by HGST (a Western Digital subsidiary, formerly Hitachi Global Storage Technologies) in 2014 [191]. The implementation of HM-SMR disks likely gained traction shortly after the introduction of SMR technology, as it became evident that host-managed approaches could offer advantages in terms of flexibility and performance optimization for specific use cases compared to DM-SMRs.

The technical report [194] hinted at performance bottlenecks with GC-based DM-SMR and proposed an alternative drive termed the *Type-II* SMR disk. A proposal followed this in [96, 200] that categorized HM-SMR disks into two categories: *Strictly Append* SMR and *Exposed SMR (Caveat) Scriptor* SMR. One of the main performance improvements expected was the *predictable* performance by HM-SMR to mitigate the *unpredictability* of DM-SMRs.

As depicted in Figure 7, unlike DM-SMRs, HM-SMRs do not autonomously manage data, placing the responsibility on the host system. Consequently, interface modifications may be necessary to expose data layout specifics to the host. Such a drive model will necessitate either changes in current filesystem implementations or the development of new filesystems. Furthermore, HM-SMR disks use T10 Zoned Block Commands (hereafter ZBC) and Zoned Device ATA Command Set (hereafter ZAC) in the context of **Serial ATA** (**SATA**) or **Small Computer System Interface** (**SCSI**) standards [52, 95].

The primary purpose of ZBC, ANSI INCITS 536, is to facilitate the management of zones within a storage device. Each zone in a zoned block device must be written sequentially, meaning that data cannot be randomly overwritten; instead, a write pointer tracks the next available position for writing data within each zone. Similar to ZBC, ZAC is defined by ANSI INCITS 537 and is designed for ATA devices, including SATA drives. The objectives and functionalities of ZAC closely mirror those of ZBC, providing commands tailored for managing zones in zoned ATA devices.

These standards provide a framework for implementing HM-SMR functionality and enabling host systems to interact with SMR drives effectively. As shown in Figure 8, we have categorized

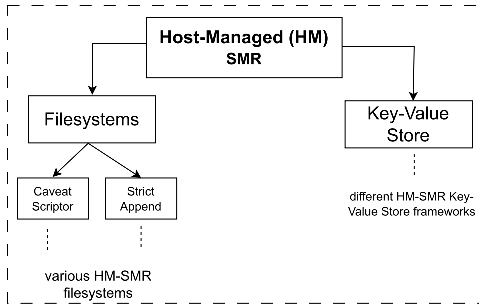

Fig. 8. Overview of HM-SMR research.

the research work related to HM-SMR into *two* categories.[4] First, the work is based on filesystem research. Second, the work is implementing Key-Value stores in HM-SMRs.

5.1 HM-SMR Based Filesystems and Block I/O Layer

According to the work in [200], mirroring the *three* main types of SMR drives (DM-SMR, HM-SMR, and HA-SMR), there exist three corresponding categories of **Shingle-Aware File Systems** (**SAFS**) [240]. Specifically, for DM-SMR drives, the filesystem is referred to as ***Device-side*** **SAFS** (**DSAFS**). For HM-SMRs, it is denoted as ***Host-side*** **SAFS** (**HSAFS**). Finally, for HA-SMR drives, it is designated as ***Cooperative*** **SAFS** (**CSAFS**).

Before explaining the work related to HM-SMR-based filesystems, it is important to explain the *two* types of HM-SMRs, namely *Strictly Append (or Strict Append)* HM-SMR and *Exposed (or Caveat Scriptor)* HM-SMR. Strict-Append HM-SMR limits host writes, confining them exclusively to the write cursors of bands. These cursors signify the next sector available for writing within a band. Writes implicitly move the write cursor forward; the only write operation is appended. During band cleaning, prompted by inter-track interference, the write cursor reverts to the initial sector of the band. It cannot be relocated elsewhere within the band apart from its beginning [103, 200]. In simple terms, this is similar to LFS [179]. Most SMR research mentioned in previous DM-SMR (Section 4.1) have implemented Strict-Append variants of LFS. In contrast, Exposed SMR (hereafter: Caveat Scriptor) represents a distinct category within HM-SMR, characterized by the host's awareness of the disk's layout and mapping. The host gains insight into the precise location of bands through interface specifications. Static mapping is typically adopted, wherein each band constitutes a consecutive set of LBAs. Given this knowledge, the host understands that writing to an LBA nullifies the data retention of all subsequent LBAs until the end of the band. Unlike other SMR types, Caveat Scriptor does not impose restrictions on host accesses; instead, it transfers the responsibility of the data retention model to the host. This shift carries implications for defect management and other reliability considerations [103, 200].

5.1.1 ShingledFS (Strict Append + Caveat-Scriptor) SMRfs. The work in [194] is one of the earliest works on HM-SMR specific filesystems. It introduces two software components: an SMR Emulator and ShingledFS, a second-generation SMR-aware FUSE-based file system. The tools offer a platform for assessing the suitability of SMR devices for Big Data workloads within the Hadoop/HDFS framework. The SMR device emulator incorporates a subset of the Banded Device Command Set, proposed to the T13 Technical Committee, to provide an API to the SMR-aware file system. This API organizes data on the emulated device to obscure its sequential

[4]This classification does not mean there are two types of consumer-grade HM-SMR drives

write characteristics. It supports Strict-Append and Caveat-Scriptor modes and operates atop a traditional raw disk partition [103].

5.1.2 Shingled File System (SFS). The work in [117] introduced the **Shingled File System** (**SFS**), offering a host-side HM-SMR specific solution to random updates in SMR disks. Unlike firmware-based solutions like DM-SMR, SFS leverages its comprehensive understanding of the usage status of all sectors to manage track shingling and minimize the burden of random updates. It was mainly designed for video servers and assumed 64 MB bands and the presence of both random and sequential shingled zones on the same disk [103].

5.1.3 HiSMRfs (Append Only). Similarly, the work in [98] developed a POSIX interface filesystem called HiSMRfs, an abbreviation for High-Performance File System for Shingled Storage Array. It supports *append-only* semantics, relies on unshingled partitions, and adds a RAID module to support striping across multiple shingled drives. It integrated an SSD to store file system metadata, aiming at reducing random writes to the SMR disk, as file system metadata accesses constitute a significant portion of random writes.

5.1.4 HM-SMR Specific Block I/O Layer. In [149], the authors developed an HM-SMR framework on the block layer within the Linux kernel 3.13 as an independent kernel module, requiring no alterations to the Linux kernel itself. The underlying Host-Managed **Software-Defined** (**SWD**) setup comprises two partitions: a conventional zone and a sequential-write-required zone. In their HMSS implementation, they utilized an SSD to mimic the behavior of the conventional zone, while a traditional HDD emulated the sequential-write-required zone. They employed a dynamic address mapping table to ensure that data was sequentially appended to the bands requiring sequential writes.

5.1.5 DM-zoned Device Mapper. Western Digital introduced the dm-zoned device mapper target as an open-source solution for managing HM-SMR devices [43, 49, 84, 159, 244]. Designed to expose zoned block devices compliant with ZBC and ZAC standards as regular block devices, dm-zoned allows these devices to operate without the write pattern constraints typical of host-managed zones. This essentially emulates a drive-managed zoned block device that abstracts away the sequential write restrictions of host-managed devices, enabling applications or filesystems accessing the block device to bypass the limitations otherwise imposed by these constraints.

The dm-zoned algorithm categorizes the device's zones into *two*types: **metadata zones** and **data zones**. Metadata zones are conventional zones designated solely for storing metadata and are not included in the device's reported usable capacity. Data zones comprise the remaining zones, with most configured as sequential zones dedicated exclusively to user data storage. Conventional zones within this category can also serve as temporary buffers for random writes, where data initially placed in these conventional zones may later be migrated to sequential zones. This migration allows conventional zones to be reused for buffering new random write requests, optimizing storage efficiency and performance.

5.2 Key-value Store on HM-SMR Drives

Research on Key-Value stores on HM-SMR disks focused on exploring innovative data storage and management approaches tailored to the unique characteristics of HM-SMR technology. For the sake of brevity, here we will focus on some of the research works that have investigated key-based access methods within the context of HM-SMR environments [30, 140, 171, 237–239].

5.2.1 SMDB. The work in [171] termed as *SMRDB*, a key-value data store for SMR disks, utilizes the proclivity of SMR disks for sequential writes. The performance of SMRDB is evaluated against a state-of-the-art LSM-tree-based key-value database engine, LevelDB, on conventional disks. The article discusses previous research related to alternate data management approaches for SMR disks, including the use of log-structured layouts, STL, and block-based indirection systems. It highlights the challenges and limitations of these approaches and emphasizes the need for SMR-aware data management solutions.

5.2.2 SMORE. The authors in [150] have developed an object storage system, namely *SMORE*, for storing large, infrequently changing data objects on host-managed SMR disks. In particular, SMORE limits the amount of SSD storage it uses, storing only a single copy of the metadata index on flash and using log records embedded in zone sets to provide failure recovery. It uses a separate garbage collection thread to migrate live data out of empty zones for space reclamation. Overall, SMORE demonstrates the effectiveness of its design and implementation for storing large, seldom-changing data objects on HM-SMR drives.

5.2.3 Ton et al. Ton *et al.* represent some of the most significant advancements in KV implementation on SMR drives. Their initial work [237, 238], known as *SEALDB*, introduces a **Log-Structured Merge tree** (**LSM-tree**) based key-value store optimized specifically for SMR drives. It effectively addresses issues related to poor random writes and severe I/O amplification. SEALDB employs several strategies to enhance performance on SMR drives. Firstly, it concatenates the SSTables of each compaction and organizes them into sets, improving compaction efficiency by reducing random I/Os. Secondly, it introduces dynamic bands on HM-SMR drives, accommodating the storage of sets and eliminating auxiliary write amplification associated with SMR drives. Their subsequent work in [239], known as *GearDB*, presents a GC-free KV store tailored for HM-SMR drives. It proposes three new techniques: a new on-disk data layout, compaction windows, and a novel gear compaction algorithm.

5.3 Performance Pitfalls in HM-SMR Drives: Uphill Battle with Linux Mainline Kernel

Until now, research on HM-SMR has indicated that the effort required to overhaul the I/O stack, particularly filesystems, for HM-SMR workloads is substantial. There have been efforts to initiate this revamp, but the challenges remain significant. Under these circumstances, the most suitable applications for HM-SMR drives are data backup or archival purposes. Another significant performance pitfall of HM-SMR disks was the inherent difficulty in ensuring sequential writes, as most I/O software stacks were designed to re-order accesses, leading to out-of-order writes and resulting in I/O errors. For instance, the Linux kernel community observed such errors when using the CFQ scheduler [51] with host-managed SMR disks [160]. As a temporary workaround, the deadline I/O scheduler [175], which was SMR-aware, was recommended, since other schedulers almost always resulted in out-of-order writes to zones, causing errors. The kernel community had currently forced the use of the deadline scheduler for HM-SMR devices, but the long-term goal was to implement a dispatch layer that could work with all I/O schedulers, removing this limitation [160].

6 Host-aware Shingled Magnetic Recording (HA-SMR)

The research studies on **Host-Aware** (**HA**) Shingled Magnetic Recording (hereafter HA-SMR) began around the year 2013. The first consumer-grade HA-SMR disk was released by Seagate in 2016 [183, 190]. Figure 9 shows the fundamental architecture of HA-SMR. [90]. The methodology for HA-SMR was initially introduced in the [200] under the nomenclature *Cooperatively Managed SMR Proposal* (hereafter referred to as COOP) [200]. Subsequently, technical discussions were presented at leading storage conferences starting from 2013 onwards [1, 37, 45, 46, 83, 90, 102, 156, 174, 201, 219]. To the best of our knowledge, academic research articles focusing on systems research related to

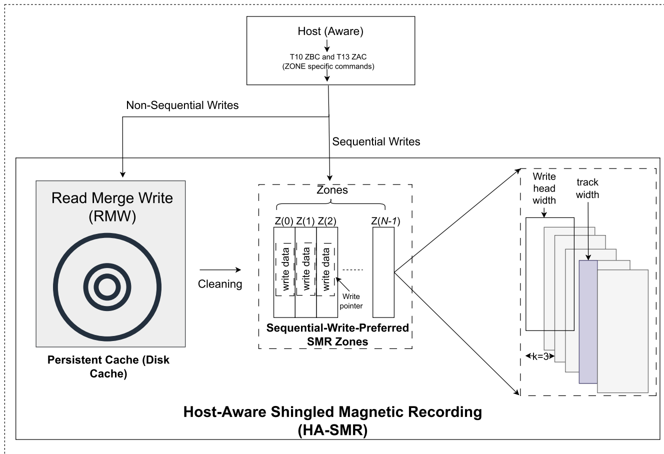

Fig. 9. Host-aware Shingled Magnetic Recording Main Architecture.

HA-SMR were first published from 2017 onwards [78, 117, 125, 131, 134, 135, 220, 224, 228, 230, 232–234, 240, 242]. The initial evaluation of a real HA-SMR drive was presented in [224].

In the initial proposal [200], the core idea behind HA-SMR was to integrate the data retention model of DM-SMR with the performance traits of HM-SMR, especially under a constrained band reuse model enforced by the host. HA-SMR is tailored for applications primarily characterized by sequential writes across large LBA extents, optionally accompanied by a small subset of randomly written extents.

Moreover, HA-SMR is designed to accommodate applications where occasional deviations from sequential write behavior may occur, even outside the randomly written extents. To achieve this goal, the primary distinction of HA-SMR lies in its intelligent *host policies*. Compared to the HM-SMR model, HA-SMR offers better backward compatibility, enabling applications to derive benefits without necessitating a substantial overhaul of the Linux block I/O stack. Moreover, HA-SMR provides essential functionalities to overcome the inherent SMR sequential-write constraint observed in HM-SMR. Similarly, in contrast to the DM-SMR model, HA-SMR offers the potential for more consistent performance by leveraging intelligent host policies through host APIs. Unlike DM-SMR drives, HA-SMR drives do not present a unified, logically consecutive space for upper-layer applications. Overall, the HA-SMR Drive represents a superset of DM-SMR and HM-SMR, equipped with disk firmware capable of managing random and sequential writes.

We have categorized the HA-SMR sections into *three* sub-sections. First, we will provide a brief overview of the cache and zone structures in HA-SMR. In the second section, we will briefly explain how the combination of T10 ZBC and T13 ZAC host APIs and disk firmware in HA-SMR disk handles non-sequential writes, thus mitigating the limitations of HM-SMR. Similarly, within this section, we will highlight where exactly HA-SMR addresses the specific limitations of DM-SMR. In the last section of HA-SMR, we will explain whether HA-SMR drives had a significant impact, namely, whether it has improved performance compared to DM-SMR and HM-SMR.

6.1 Cache and Zones in HA-SMR

The **cache** in HA-SMR drives operates transparently to the host, meaning that the host does not need to be aware of its existence or management. This cache is utilized to temporarily store non-sequential writes, which are then later organized and written sequentially to the appropriate SMR zones. This process helps mitigate performance issues associated with non-sequential data writes, which can introduce significant latency in traditional storage systems. When a non-sequential write occurs, the HA-SMR drive redirects this data to the persistent cache. A flag is set for the corresponding SMR zone, indicating that subsequent writes will also be directed to the cache until a cleaning process is performed. This cleaning process involves migrating the buffered data back to its original SMR zone, thus maintaining data integrity and optimizing performance.

In HA-SMR drives, **SMR zones** are segments of the disk optimized for sequential data writing, with each zone typically ranging from tens to hundreds of megabytes in size. These zones are classified as *sequential-write-preferred*, allowing for efficient data storage while accommodating some non-sequential writes through a hidden (or virtual) persistent cache. When a non-sequential write occurs, the data is temporarily redirected to this cache, and the corresponding zone is marked as *non-sequential.* Later, during a cleaning process, the cached data is migrated back to its original zone using a Read-Modify-Write operation, restoring the zone's sequential state. Zones can be **open** or **closed**, with open zones available for new writes and closed zones filled to capacity.

6.2 Non-sequential Writes in HA-SMR Drive

The work in [234] has provided a detailed explanation of the operations of HA-SMR. For the sake of brevity, we will focus solely on the core operations. As illustrated in Figure 9, HA-SMR consists of a significant quantity of *sequential-write-preferred* SMR Zones designated for storing user data. Each SMR Zone typically spans tens to hundreds of **megabytes** (**MBs**), and an HA-SMR drive typically comprises several thousand such SMR Zones. All of the sequential-write-preferred SMR zones form a linear and contiguous storage space, where the smallest LBA is mapped to the beginning of the SMR Zone Z_0, and the largest LBA is mapped to the end of the last SMR Zone Z_{N-1}.

To write data into SMR Zones, the host will send *"sequential writes"* to populate an SMR zone sequentially, appending data to the location specified by the zone's corresponding *write pointer*. The figure shows that the *write data* can be appended to SMR Zone Z_2. Once an SMR Zone has been sequentially filled, the host system can also issue the special *RESET _WRITE_PTR()* ZBC command to reset the write pointer for this zone, enabling it to accommodate new writes. However, before resetting an SMR Zone by resetting its corresponding write pointer, the host must ensure that there is no valid data remaining in this zone. This condition is essential to ensure that data is not lost. Next, the true distinguishing factor emerges within the data model as compared to HM-SMR. When the host sends "non-sequential writes" to store non-sequential data at a location not indicated by the write pointer of any SMR zone, instead of reporting an *ILLEGAL_REQUEST* as the HM-SMR drive does, the HA-SMR drive will automatically redirect all non-sequential writes to a *"Persistent Cache"* to avoid violating the SMR sequential-write constraint. For example, in Figure 9, *write data* in the first zone is not written to the current write pointer position of SMR Zone Z_1, so the HA-SMR drive will automatically redirect $writedata$ to the cache. Finally, at certain intervals, the HA-SMR drive will transfer these temporarily stored data from the Cache back to their respective SMR Zones using the RMW operation.

We have taken another concrete example from [228] to explicitly explain the write operation in 6 steps as depicted in Figure 10. ❶ Initially, incoming write requests. ❷ Sequential write is written into the sequential HA-SMR zone. ❸ Conversely, non-sequential writes are rerouted to the media cache, awaiting further processing. ❹ Even a write aimed at the write pointer of a non-sequential HA-SMR zone falls under the range of non-sequential writes. ❺ Notably, a non-sequential write poised to yield obsolete data during the subsequent cleaning process underscores the complexities inherent in SMR technology. ❻ To address this, the media cache undergoes clearance through the RMW operation, ensuring data integrity and optimal storage utilization.

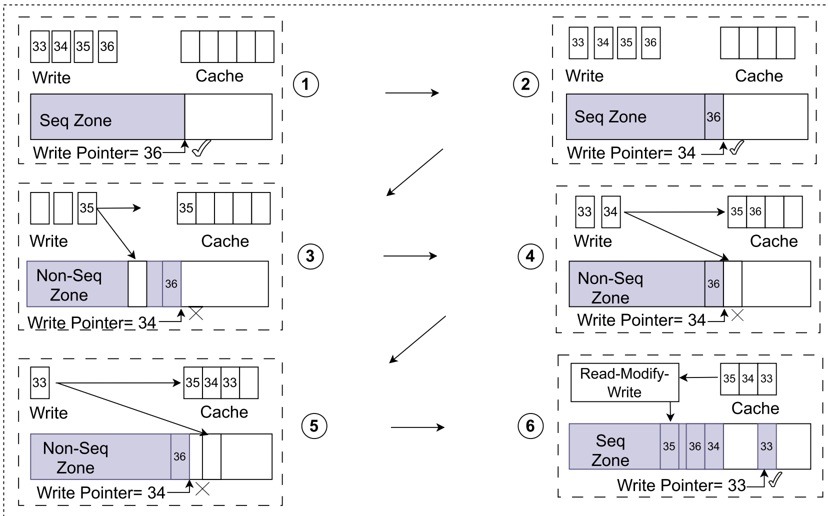

Fig. 10. Write Request from initial zone state to cache cleaning in HA-SMR [228].

6.3 DM-SMR and HM-SMR Bottleneck Mitigations in HA-SMR

Given that the HA-SMR drive aims at addressing the performance limitations of DM-SMR and HM-SMR, exploring some of the research efforts in this area is essential. Tail latency, characterized by prolonged response times due to blocking cleaning in STL, emerged as a primary performance bottleneck in DM-SMR drives, as discussed in preceding sections (Section 4.7). Consequently, much of the research in HA-SMR has also focused on mitigating tail latency (unpredictability). This bottleneck is particularly significant considering that tail latency in DM-SMR drives has previously been measured in seconds even for a single I/O request [3, 4]. Additionally, we will explore some of the research dedicated to filesystems specific to HA-SMR drives, offering a fair comparison to HM-SMR implementations.

6.3.1 Wu et al. *on Evaluation of Real HA-SMR Drive.* Yang *et al.* [220, 224] evaluated the performance of HA-SMR drives, focusing on features like the open zone issue and media cache cleaning efficiency. This was the first work that has evaluated HA-SMR comprehensively. It proposes a host-controlled indirection buffer to enhance the drive's I/O performance. The open zone issue refers to performance degradation when the recommended maximum number of open zones in HA-SMR drives is exceeded. HA-SMR drives have specific open zones, areas where data can be written without affecting adjacent tracks. When the number of open zones exceeds the recommended limit, the performance of sequential writes can significantly decrease. The work creates sustained non-sequential write workloads to evaluate the performance of HA-SMR drives in handling a large number of zones. The testing program varies the update ratio, IO request number, and IO request size to analyze their impact on the average band cleaning time of HA-SMR drives.

6.3.2 HA-SMR Mitigating the Long Latency. The work introduced in [233, 234] presents a comprehensive Virtual Persistent Cache design aimed at addressing the prolonged latency challenges observed in HA-SMR drives. While the work focuses on HA-SMR drives, its implications suggest that it could contribute to mitigating the prolonged latency bottleneck prevalent in DM-SMR drives as well. Similarly, the work in [242] also targets to mitigate the long latency in HA-SMR drives. Inspired by the varying performance impacts of idle and blocking cleanings, the work explores a potential strategy for mitigating the substantial tail response time resulting from blocking cleanings through **Artificially Triggered Idle Cleanings** (**AT-IC**). AT-IC aims at alleviating the performance degradation caused by blocking cleanings by simulating idle periods during workload execution. During these artificially induced idle durations, I/O requests are delayed and processed in subsequent execution duration. This allows for the invocation of idle cleanings, ensuring that cleaning operations proceed uninterrupted for a predefined idle duration.

6.3.3 HA-SMR Filesystem CosaFS. To the best of our knowledge, the work [240] termed as *CosaFS* (cooperative shingle-aware file system) is the first work that has presented an HA-SMR based filesystem. The basic idea of CosaFS is to classify objects as hot or cold, where hot objects are served by SSDs and cold objects are stored on SMR drives. The authors conducted evaluations of CosaFS in comparison to filesystems based on DM-SMR and HM-SMR technologies. The findings of this study demonstrate improvements in throughput and reductions in latency when compared to DM-SMR and HM-SMR baseline performances. While the authors highlight the separation of metadata from file data as a primary distinguishing feature of CosaFS, aiming at optimizing SMR drive bandwidth utilization, our research suggests that similar decoupling of metadata is also feasible in various HM-SMR-based file systems.

7 Dynamic Hybrid-Shingled Magnetic Recording (H-SMR)

Dynamic H-SMR [202, 206, 221, 222] is a drive technology that enhances the flexibility and efficiency of **hard disk drives** (**HDDs**) by enabling dynamic reconfiguration between *two* recording zones (or regions): CMR zone and SMR zone. The CMR zones are typically positioned in the *middle* of the disk, situated halfway between the inner and outer tracks. This strategic placement is advantageous because the head, fixed to the head assembly, maintains a perfectly tangential position to the disk at this point, thereby minimizing data access errors.

The concept involves distributing *hot* data storage across a group of disk drives by allocating CMR regions for hot data on each drive instead of dedicating the entire disk surface to hot data. This strategy aims at enhancing data access speed. Meanwhile, the remaining portion of an H-SMR drive is designated for storing *cold* data within an SMR region. The original concept for H-SMR was introduced under the title *"Flex Dynamic Recording"* in [202]. Simultaneously, other hard disk vendors also presented their initial set of requirements [17, 19, 112, 151, 184, 202, 214]. The alternative names to the H-SMR were *Flex* and *Realms*.

Importantly, the adoption of H-SMR facilitates SKU reduction, allowing manufacturers to ship a single drive model that can be reprogrammed to accommodate different customer preferences for CMR zone allocation, based on their unique use cases and engineering requirements [206].

We stick to H-SMR as it simplifies the understanding of its microarchitectural details. To our knowledge, only three research studies have specifically addressed H-SMR technology [202, 221, 222]. Moreover, in [78], the term "re-writable" zones is introduced, which shares a similar concept. This work presents an SMR translation layer employing re-writable zones to diminish garbage collection overhead and enhance performance.

As depicted in the Figure 11, the H-SMR physical media can be organized into either CMR zones (occupying the lower address space) or SMR zones (occupying the higher address space). Here, a zone refers to a contiguous LBA space measuring 256 MiB in size. When there's a combination of SMR and CMR regions on the disk, it's possible to assign frequently updated data to the CMR zone while reserving the SMR zone for less frequently accessed or cold data. The host has the capability to convert any desired number of zones from one format to another through H-SMR APIs. These APIs enable the host to take a consecutive set of zones from one format offline and bring the corresponding zones in the other format online. In this context, *"online"* refers to the zone being supported by the physical disk area, while *"offline"* indicates the opposite. Because CMR and SMR have different areal densities, the number of zones typically changes after conversion compared

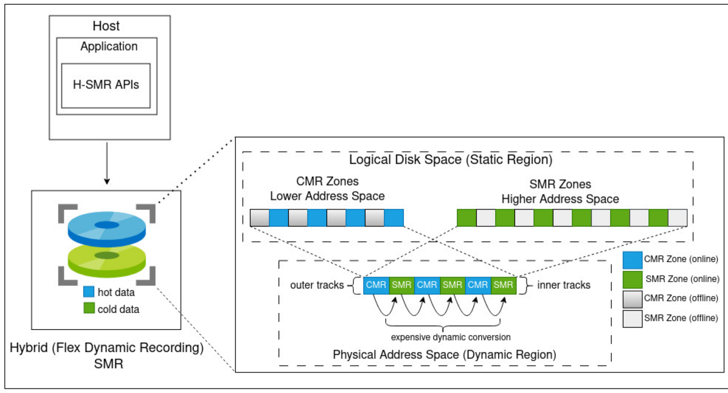

Fig. 11. H-SMR.

ALGORITHM 1: Hybrid-Shingled Magnetic Recording (H-SMR) Stroke principle

$h_s, h_e, t, p, s_p, h_p, n_v, v_p \leftarrow$ HotDataLBAStart, HotDataLBAEnd, TotalLBA, StrokePercentage, ShortStrokePercentage, HotDataPercentage, NumVolumes, VolumePercentage
procedure SS(h_s, h_e, t, p)
    $s_p \leftarrow \frac{h_e - h_s}{t} \times 100$
    return s_p
end procedure
procedure PD(t, n_v)
    $v_p \leftarrow \frac{100}{n_v}$
    for $i \leftarrow 1$ **to** n_v **do**
        Allocate Volume i with v_p% of Stroke
    end for
end procedure
procedure AD(d)
    if d within HotDataRange **then**
        Use ShortStroking
    else
        Access Data using Regular Stroke
    end if
end procedure

to before. Implementing such a data allocation methodology can decrease the overhead associated with SMR updates mentioned in previous sections. This is because the application can directly update data in the CMR area in place, thus avoiding the SMR update overhead.

Building upon the core concept described in [202], the precise microarchitectural aspects are further explained in Algorithm 1. The term *"stroke"* in disk terminology refers to the range of disk tracks accessed by an application, correlating with the movement range of the heads. When an application accesses the entire LBA space, it utilizes 100% of the stroke. Partitioning a disk into ten volumes means each volume utilizes roughly 10% of the stroke. This correlation between logical addresses and physical radius assumes the conventional mapping of the logical-to-physical disk, where lower LBAs are situated on the outer tracks. With this partitioning scheme, if the disk workload is confined to a single partition, then the application utilizes approximately 10% of the stroke. As access time is influenced by disk seek distance, this restricted workload operates significantly faster compared to a workload spanning 100% of the stroke.

In practice, speeding up accesses to hot data (CMR Zone) can be achieved by confining it to a restricted range of LBAs, a technique commonly referred to as *short stroking*. Likewise, cold data, in contrast to hot data, can typically be sequentially written using the SMR zone. This not only boosts the (IOPS) but also significantly enhances the performance density (IOPS per TB) as the denominator of that metric reduces. However, adapting to dynamic workloads still remains a hard problem. Overall, the exploration and implementation of H-SMR architecture, particularly in the context of the first commercial HAMR disk, presents an interesting avenue for research within the storage community. By combining the advantages of both SMR and HAMR technologies, H-SMR offers the potential to significantly enhance storage density while maintaining acceptable performance levels.

8 RAID, Deduplication, and Real-world Applications of SMR Technology

With a comprehensive and in-depth understanding of the technical evolution of SMR technology in previous sections, this section provides a comprehensive overview of how SMR technology integrates with contemporary data storage solutions. RAID [16, 26, 141, 165, 197, 198] systems exploit the high capacity and cost-effectiveness of SMR drives, enabling enhanced storage performance and data redundancy. In conjunction with deduplication techniques [54, 154, 158, 162, 166, 208, 226], SMR storage solutions facilitate significant reductions in data redundancy, thereby optimizing storage efficiency and resource utilization. Additionally, various industrial applications [89, 177, 247, 250] utilized the advantages of SMR technology, which is particularly well-suited for environments characterized by large-scale data storage needs. These applications span diverse sectors, including cloud computing, big data analytics, and archival storage, showcasing the effectiveness of SMR in meeting modern storage challenges. The next three sub-sections (Section 8.1, Section 8.2, and Section 8.3) will provide a brief and concise discussion of RAID, deduplication, and industrial use cases of SMR technology.

8.1 RAID

Redundant Array of **Independent Disks** (**RAID**) [16, 26, 165, 197, 198], once termed as a *black art* [215], is a robust storage architecture that aggregates multiple physical disks to enhance redundancy, performance, and data accessibility. With the adoption of SMR drives, researchers revisited traditional RAID techniques, focusing specifically on RAID 4 and RAID 5 to address SMR's unique properties [98, 113–115, 129, 130, 139]. One of the motivations for this exploration arose from the increasing demand for cloud storage solutions, which resulted in the accumulation of massive data within their mass storage systems. These systems needed to scale extensively while maintaining low costs, particularly as cloud storage was predominantly utilized for data archiving, characterized by frequent writes, limited reads, and virtually no updates to archived data. This presented an optimal scenario for the implementation of SMR disks, given their increased areal density and cost efficiency, thereby offering significant advantages for such workloads.

8.1.1 HiSMRfs: HiSMRfs [98] is a file system designed specifically for SMR drives, featuring a standard POSIX interface. HiSMRfs incorporated a file/object-based RAID 5 module tailored for SMR/HDD arrays, which computed parity for individual files or objects, ensuring that both data and parity writing occurred sequentially and fully within a stripe. This file system was also compatible with hybrid storage systems that integrated conventional HDDs and SSDs.

8.1.2 RAID 4S:. Le *et al.* [113, 115] proposed a novel application of SMR disks in conjunction with conventional disks, termed RAID 4S, which had the potential to enhance the performance of storage systems. Their evaluation demonstrated that RAID 4S effectively showcased the feasibility of utilizing SMR disks within a RAID 4 array, outperforming the use of SMR disks in standard RAID 4 configurations with in-place updates.

8.1.3 DVS:. The authors [139] proposed a dynamic variable-width striping RAID specifically designed for SWDs to reduce the costs associated with parity updating. Their contributions were summarized as follows: First, they introduced DVS-RAID for SWDs, which dynamically generated new full or partial stripes and appended new data to the tail of the existing data. Second, they designed a write cache management system that took into account the unique properties of SWDs for DVS-RAID. Finally, they implemented a DVS-RAID simulator and evaluated its performance under various workloads. The results indicated that DVS-RAID outperformed traditional HDD-based RAID systems, particularly under read-dominant and sequential write-dominant workloads.

8.1.4 HSMR-RAID:. Lin *et al.* [129] proposed an SMR-friendly RAID 5 storage system, named HSMR-RAID, which was developed within a pure SMR environment without the use of additional storage drives. HSMR-RAID was based on a host-managed SMR architecture. The fundamental principle of HSMR-RAID was to leverage data information from the host system to group data blocks that would be invalidated simultaneously within the same stripe. This approach effectively reduced the amount of data movement and minimized the need to recalculate parity data, thereby lessening garbage collection overhead. Experimental results demonstrated that the proposed scheme could enhance garbage collection performance.

8.2 Deduplication

In data storage systems, deduplication has been a critical technique for reducing storage demands and optimizing capacity [20, 54, 154, 158, 166, 208, 226]. By identifying and eliminating redundant data blocks, deduplication minimizes the amount of data stored, thereby enhancing storage efficiency and cost-effectiveness. This process is particularly valuable in environments with rapidly growing data volumes, where redundant data can consume significant storage resources and impact system performance. When applied to SMR drives, deduplication offers the potential for both space savings and increased storage density, addressing the rising data demands in various applications.

8.2.1 SMR-aware Fingerprint Store. The first work that applied deduplication techniques to SMR was in 2018. Wu et al. discussed that naively applying deduplication techniques on SMR drives may downgrade the runtime performance of data storage services, because of the time-consuming SMR space reclamation processes. Therefore, the authors advocated a vertical integration solution by jointly managing the HM-SMR drives with a deduplication system, in order to essentially relieve the time-consuming SMR space reclamation problem. To effectively improve the runtime performance, the work leveraged the new design flexibility provided by the HM-SMR drives to develop an *SMR aware fingerprint* store to jointly manage the HM-SMR drives with the deduplication system. In particular, the proposed design exploited the concept of *chunk lifetime* to effectively reduce the runtime performance delay of SMR zone reclamation processes. The emulated SMR drive was developed based on the Skylight emulator [4], to emulate the characteristics of a host-managed SMR drive. Moreover, the authors used block I/O storage tool *fio* (version 2.6) to replay the evaluated traces and to precisely measure the performance results. To the best of our knowledge, there are four research publications that apply deduplication techniques to SMR drives. Later in year 2022 the authors extended their existing work [217].

8.2.2 SMRTS. [20] The authors designed and implemented SMRTS, a performance optimized file system for SSD **SMR** **T**iered **S**torage. In SMRTS, SSDs and SMR drives were used as two storage tiers, called the SSD-tier and SMR-tier, respectively. First, metadata, the deduplication index, and frequently accessed files (hot files) were stored in the SSD-tier. Periodically, the data of rarely accessed files (cold files) were migrated to the SMR-tier to reclaim space in the SSD-tier. During the migration to the SMR-tier, files were deduplicated, and only unique data chunks were accumulatively stored in a container buffer, which was written to the SMR-tier sequentially once full. When these deduplicated files were requested, they were restored to the SSD-tier as the original files. Additionally, a fine-grained read/write logic was designed.

8.2.3 LaDy. The researchers in a recent publication [24] proposed a framework, **L**ocality **A**ware **D**eduplication technolog**y**, named LaDy, to address both the overhead of writing duplicate data and the impact on data locality. The framework determined the necessity of writing duplicate data to optimize performance. The researchers integrated LaDy with DiskSim, an open-source simulation tool, and modified it to model an SMR-based drive. Experimental results demonstrated that the framework significantly reduced response times in the best-case scenario compared with the well-known deduplication method CAFTL on an SMR drive. LaDy achieved this by selectively writing duplicate data, thereby preserving data locality and improving read performance.

8.3 Industrial Use cases

8.3.1 Dropbox. Dropbox [53] collaborated with Western Digital to incorporate Host-Managed SMR disks, revolutionizing large-scale cloud storage with purpose-built solutions optimized for sequential data zones. This integration allowed Dropbox to scale efficiently, leveraging advanced storage architecture to handle exabyte-scale data. Western Digital supported this endeavor by providing essential open-source tools and contributing to zoned storage standards, which were crucial for re-architecting storage management practices at Dropbox. The deployment of 14 TB Ultrastar® DC HC620 SMR drives enabled Dropbox to achieve up to 25% more capacity compared to the prior 12 TB models, significantly enhancing storage density and optimizing data storage for modern cloud applications.

8.3.2 Alibaba's SMRSTORE. The researchers from Alibaba [13] developed a framework, SMRSTORE [247], which is a high-performance user-space storage engine designed to leverage HM-SMR drives for standard-class cloud object storage. The problem statement highlighted that existing HM-SMR solutions are unsuitable for standard-class Alibaba Cloud **Object Storage Service** (**OSS**) due to significant throughput drops. Furthermore, although both archival-class and standard-class OSS share the same data abstraction (object), their differing **Service Level Objectives** (**SLOs**) imply that designs effective for archival storage may not adequately meet the performance requirements for standard-class needs. To tackle these performance challenges, Alibaba engineered *SMRSTORE* with a fully log-structured design, direct chunk store interfaces over SMR drives, and strategic data placement to minimize GC impact. Evaluations showed that SMRSTORE achieved performance comparable to Ext4 on CMR drives and outperformed F2FS on SMR drives by up to 2.16 times.

8.3.3 IBM Zone Slice Storage (ZSS) and SoftwAre Log-Structured Array (SALSA). To the best of our knowledge, there are *two* notable works by IBM that have leveraged SMR drives. First, IBM **Cloud Object Storage** (**COS**) research and development team developed a specialized storage engine called **Zone Slice Storage** (**ZSS**) to optimize data handling on SMR drives, which effectively utilized the drives' sequential write characteristics to minimize performance degradation, a common issue in traditional garbage collection schemes. Second, IBM introduced the **SoftwAre Log-Structured Array** (**SALSA**), a unified software stack that manages low-cost Flash-based SSDs as well as SMR hard disks. SALSA enhances the performance and endurance of low-cost, commodity

Flash SSDs to meet the requirements of data centers for specific classes of workloads. Additionally, SALSA enables SMR hard disks to be accessed over the traditional block interface with increased performance, allowing them to serve as drop-in replacements for conventional hard disks [89, 107, 177, 250].

8.3.4 Ceph's Bluestore and SMR. Aghayev *et al.* [2] integrated SMR drives into the Ceph [7, 22] distributed storage system. Modifications were made to BlueFS, a user-space file system, and RocksDB to operate directly on SMR drives, bypassing the typical translation layer and thereby reducing the associated overhead. The adaptation of BlueStore, Ceph's purpose-built storage backend, to zoned devices demonstrated how Ceph could leverage the additional storage capacity of SMR drives, achieving better throughput while minimizing tail latency.

8.3.5 Google's Advocacy for Caveat Scriptor. Presented at the 14th USENIX Conference on File and Storage Technologies in 2016, the whitepaper *Disks for Data Centers* highlighted the critical role of disks in cloud-based storage systems [19]. In this context, authors mentioned that SMR emerges as a significant technological advancement, allowing hard disks to achieve a higher areal density, which is crucial given the exponential demand for storage in data centers. However, SMR's write restrictions can complicate data management, necessitating innovative solutions to optimize performance. One such solution is through host-managed SMR abstractions, such as the *Caveat Scriptor* proposal [112]. This approach not only reduces GC overhead but also enhances overall disk performance by facilitating dynamic resizability of SMR zones and supporting diverse write patterns.

8.3.6 Huawei. To the best of our knowledge, only two notable studies have utilized SMR disks. The first [140], published in 2015, explored the integration of SMR technology into object storage systems, emphasizing its advantages for managing large-scale data. This work detailed the architecture and performance benefits of employing a key-value store model, which enhances data retrieval efficiency and scalability, while also discussing practical applications and deployment strategies for cloud storage solutions. The second study, conducted by Huawei [157], focused on integrating 20 TB SMR drives into its OceanStor Pacific storage array. This mass data storage solution is optimized for the demands of the 5G era, featuring a scale-out architecture capable of supporting configurations from three to 4,096 clustered nodes, thereby ensuring reliable and scalable data storage.

9 Lessons Learned from Current SMR Technology and Extending SMR to Future Technologies in Systems Research

As explained in previous sections, the evolution of SMR disks has seen significant advancements in its architecture over a relatively short period of around 17 years. Initially, SMR drives operated under a *black-box* model known as DM-SMR, where the drive autonomously managed data placement without transparency to the host system. Subsequently, a more predictable model emerged with HM-SMR, enabling control by the host for improved performance. Concurrently, HA-SMR combines aspects of both DM-SMR and HM-SMR architectures, offering a balance between autonomy and host control. Despite these advancements, performance challenges highlighted in previous sections have somewhat hindered the wide adoption of SMR disks in performance-critical ecosystems. In brief, the performance gains initially anticipated from SMR drives are still not achievable.

However, it's crucial to acknowledge that *shingled* magnetic technology remains relevant and will undergo a transitional phase toward future disk media. Given this, exploring future disk media became imperative. The two most recent roadmaps discussed in [12, 178], as illustrated in Figure 1, which also integrates the roadmap from the **Advanced Storage Research Consortium** (**ASRC**),

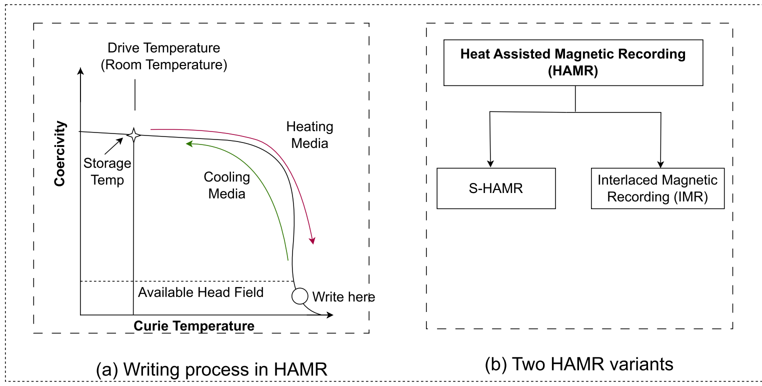

Fig. 12. (a) A schematic figure of writing in HAMR, (b) two specific HAMR-variants discussed in our work.

have projected advancements in HDD areal density. These advancements are envisioned through the adoption of *energy*-assisted magnetic recording and the reduction of magnetic media grain size.

TDMR [23] and MAMR [249] represent two key advancements in magnetic recording technologies. TDMR improves areal density by using multiple read heads to enhance signal quality and reduce interference between tracks, but it does not completely eliminate the inherent limitations of SMR. On the other hand, MAMR is an energy-assisted magnetic recording technology that uses microwave energy to assist in the recording process. This allows for higher data density without compromising write performance and is viewed as a promising step toward more efficient and scalable disk storage solutions.

Additionally, technologies such as bit patterned media (briefly discussed in Section 10) are expected to be implemented, which define separate islands of magnetic media. One significant approach in energy-assisted magnetic recording is **H**eat **A**ssisted **M**agnetic **R**ecording (hereafter HAMR).

HAMR has emerged as the leading candidate for the next generation of magnetic recording technology, anticipated to surpass densities of 1.5 $Tbit/in^2$ and potentially reaching densities of up to 5 $Tbit/in^2$. Moreover, substantial research in HAMR is evident from the filing of over 400 patents for HAMR drives [39]. Although research on HAMR spans several decades, for brevity, the pivotal work in [109, 180, 185] laid the groundwork for the practical realization of HAMR technology. HAMR drives incorporate advancements in physics and magnetics, fields beyond our expertise as system researchers. Therefore, our discussion will focus solely on the most fundamental aspects of HAMR drives. HAMR is a magnetic storage technology for increasing the amount of data that can be stored on a magnetic device such as a HDD by temporarily heating the disk material during writing, which makes it much more receptive to magnetic effects and allows writing to much smaller regions (and much higher levels of data on a disk).

A sketch illustrating the HAMR writing process (or recording mechanism of HAMR) is shown in part (a) of Figure 12. As system researchers, without delving into the magnetics, material science and physical intricacies of the writing process in HAMR drives, it's crucial to grasp the underlying principle. Essentially, temporarily elevating (or laser heating) the temperature at the writing spot facilitates the establishment of the required magnetic field and enables writing to a smaller region. This process effectively boosts areal density, enhancing storage capacity. In simpler terms, HAMR circumvents the performance drawbacks linked with drives based on SMR or CMR by preheating the drive before writing data to it. As previously acknowledged, existing scientific literature predominantly delves into the physical aspects, such as material science and magnetics, of HAMR drives. However, there is a noticeable lack of emphasis on systems-level considerations regarding HAMR drives.

In the following sections, we will explore the current state-of-the-art in systems-level aspects related to HAMR drives. The current research clearly indicates that *shingled* technology will continue to play a role as one of the variants of HAMR drives. This integration signifies a hybrid approach, leveraging the benefits of both disk media to further enhance storage capacity and performance. Specifically, we will delve into the current research on **Shingled Heat Assisted Magnetic Recording** (**S-HAMR**) as depicted in part (b) of Figure 12 as depicted in part (b) of Figure 12. Furthermore, as sketched in Figure 12, we will discuss another variant of HAMR media known as **Interlaced Magnetic Recording** (**IMR**). Both of these technologies are currently the subject of active investigation within the storage research.

9.1 Shingled Heat Assisted Magnetic Recording (S-HAMR)

Shingled **H**eat **A**ssisted **M**agnetic **R**ecording (hereafter S-HAMR) represents a significant advancement in data storage technology by combining the benefits of SMR with the precision and high-density capabilities of HAMR. Hence, it is essential to briefly discuss some of the relevant research that explains the *shingling* variant in HAMR [35, 67–69, 73, 91, 92, 106, 216].

Again, the primary emphasis of the discussed research has been on the physical aspects of the S-HAMR drive. The work in [73] discusses the results of simulations of **Shingled recording with Thermally Assisted Magnetic Recording** (**STAMR**) with an areal density target of 8 $Tbit/in^2$. The results hinted that an areal density of 8 $Tbit/in^2$ could be achieved with a shingled track width if demanding conditions on the heat spot and recording medium parameters were met. The authors of work [35] propose to use a wide **near field transducer** (**NFT**) [216] to write narrower HAMR tracks by writing tracks sequentially overlapping with previous tracks (i.e., shingling) to increase track density and areal density.

It introduces the concept of SH-HAMR. Then, the work performs the experimental verification of Shingled HAMR. The concept of SH-HAMR shows the proposed HAMR writing process. Recorded HAMR tracks are sequentially overlapped with previous tracks with overlapped thermal spots (/write bubbles) for both bits and tracks. The research presented in [67] investigated the potential areal density capabilities of HAMR CMR, HAMR SMR, and HAMR IMR combined. The study reported that the track layout architecture of HAMR IMR allows for a further increase in HAMR areal density. Additionally, the gains observed in HAMR SMR primarily stem from reductions in track pitch. The work in [68] compared the areal density of HAMR-CMR and HAMR-SMR at various reader clearances with integrated HAMR readers and tested narrow high SNR readers.

In the context of HAMR drives, reader technology usually includes components such as read heads or read sensors, which are designed to detect and interpret the magnetic signals recorded on the disk surface. In a more recent study by the same authors [69], they experimentally compared the dependence of writer current on HAMR areal density for both CMR-HAMR and SMR-HAMR configurations. This comparison was conducted on a spinstand with 6× adjacent track writes for CMR and 1× adjacent track writes for SMR. In HAMR technology, the write head typically comprises a magnetic writer and a laser diode. The writer's current is utilized to generate a magnetic field, aligning the magnetic domains in the recording medium to encode the data.

The work in [91] is based on a comparison of **Adjacent Track Erasure** (**ATE**) in C-HAMR and SH-HAMR. ATE is a phenomenon in disk recording where data written on one track of a disk medium unintentionally affects data on adjacent tracks. This interference typically occurs due to the magnetic field generated during the writing process, which can extend beyond the intended track and affect neighboring tracks. In brief, while comparing the CMR-HAMR and SH-HAMR,

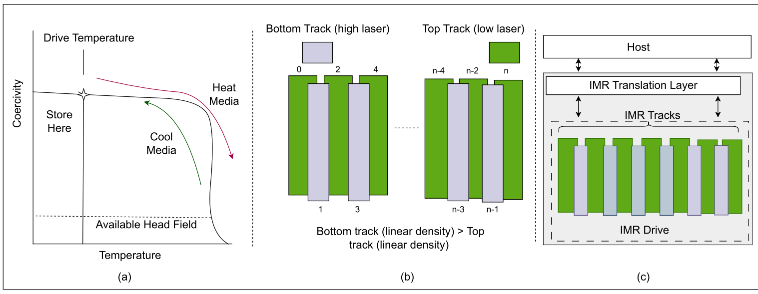

Fig. 13. IMR uses both heat-assisted recording and track overlap.

the work [91] is based on improving the areal density of SH-HAMR, focusing on further mitigating the ATE-related limitations in SH-HAMR. 3D HAMR, as also mentioned in recent roadmap [178], refers to an advancement in HAMR technology that involves stacking multiple layers of magnetic recording media on top of each other within a single HDD [204]. The comparison between SH-HAMR and 3D HAMR is an important requirement to determine the advantages of 3D HAMR, especially in terms of recording density and interference management. The authors in [106] compared 4 Tbpsi SH-HAMR and 2 Tbpsi/layer 3D HAMR.

Considering the relevance of multi-layer HAMR drives, in a recent article, authors demonstrated two-layer heat-assisted magnetic recording [205]. The research team combined experimental measurements and proof of concept of multi-level recording to show that this approach to magnetic recording could accomplish up to 4 Tbpsi areal density. According to an article [40], while SMR upgrades of a CMR drive typically yield a 16.66% capacity increase, HAMR drives demonstrated a smaller increase of around 6.66%. However, in the same article, it was mentioned that product offerings may not fully reflect the overall gains between SMR and CMR technologies at a fundamental level, as capacity gains with SMR on HAMR drives may vary and are not fixed.

9.2 Heat Interlaced Magnetic Recording (HIMR)

Heat **I**nterlaced **M**agnetic **R**ecording (hereafter HIMR) is an alternative recording architecture where tracks are written in an interlaced order with different linear densities [28, 34, 61, 66, 67, 70, 80, 93, 94, 104, 120–123, 126, 223, 225, 241, 243]. IMR is different from SMR, which relies on track overlap alongside conventional room-temperature recording techniques. It is a relatively new drive technology, with the initial descriptions emerging in 2016 [94].

The basic architecture of HIMR is illustrated in Figure 13. In the first part (a) of Figure 13, HIMR adopts principles from HAMR, where a precise laser heats the recording medium to its Curie point, allowing the magnetic write head to modify the material's magnetic orientation. By integrating heat assistance with the interlaced track layout, HIMR aims at achieving higher data densities and improved write performance compared to conventional magnetic recording methods. Part (b) of Figure 13 illustrates the interlaced writing process, where tracks are written in a staggered manner. Initially, a "bottom" layer of tracks is written, followed by the writing of a "top" layer positioned between and partially overlapping these bottom tracks. Finally, part (c) in Figure 13 presents a generic system-level architecture of an IMR drive.

In an IMR drive, the bottom tracks are designed to support a high linear density by being written wide and using a high laser power, while the top tracks are narrower and require lower laser power. However, writing directly to a bottom track can lead to the loss of data blocks already written on neighboring top tracks. To prevent this potential data loss, the data on the affected top tracks must be read before updating the bottom track. After the bottom track is updated, the preserved data from the top tracks is rewritten back to their original positions, constituting the RMW operation. Although this approach introduces constraints on writing, they are less stringent compared to those imposed by SMR disks. Since only the top tracks are prone to encroachment, the laser current applied to the top tracks defines the track pitch of the entire HIMR system. It's important to note that the terms *bottom* and *top* are conceptual and do not imply the existence of distinct physical layers on the IMR disk.

To the best of our knowledge, research on system-level implications involving IMR began to emerge in 2018. Specifically, we have identified 12 research studies that explicitly focus on the system-level aspects of IMR disks [28, 61, 80, 104, 121–123, 126, 223, 225, 241, 243]. Initial research studies have delved into *data placement* algorithms tailored specifically for IMR-based drives. Additionally, investigations have been conducted into the development of *filesystem* designs optimized for IMR drives, such as the interlacing of magnetic tracks.

Furthermore, studies have concentrated on evaluating the impact of IMR on various storage *workloads* and identifying performance pitfalls to leverage IMR's capabilities to enhance overall system performance. Collectively, these research efforts contribute to advancing our comprehension of IMR technology and its integration into modern storage systems, laying the groundwork for extending IMR to HAMR-based drives. Instead of merely narrating the work, we will specifically focus on the incremental changes brought about by each research study in improving IMR performance. Therefore, we have categorized IMR system-level research into *three* categories: first, the work that contributed to IMR translation layer algorithms; second, the work that simulated or emulated the IMR drive; and third, the work that contributed to IMR-specific file systems.

9.2.1 IMR Translation Layer. As previously mentioned, track rewriting poses a significant performance challenge in IMR drives. Addressing this issue, one of the earliest research studies aimed at mitigating the inherent track rewriting problem in IMR. Gao *et al.* [61, 104] proposed a *three-phase* write management method. This approach involves allocating disk space based on three phases of space usage. Initially, when space usage is below the total space of bottom tracks, data is exclusively written to the bottom tracks without incurring any track rewriting overhead. Once all bottom tracks are utilized, the *second* phase commences, wherein data is stored in every other top track, while the remaining top tracks are reserved for data accommodation in the *third* phase. Subsequently, Wu *et al.* [223, 225] have also developed data management techniques for IMR translation layers. Their first work [225] proposed a *three-phase* baseline that differs from the earlier three-phase implementation in [61, 104] by reversing the second phase allocation direction to preserve data locality.

Furthermore, whereas the three-phase implementation in [61, 104] maps data blocks statically, their DM-IMR can adapt to the access pattern of dynamic workloads by buffering updates to the unallocated top tracks (Top-Buffer) and swapping hot bottom track data with the cold top track data (Block-Swap). The work in [80] performed one of the initial performance analysis of the RMW approach IMR drives. In addition the authors proposed three new algorithms for IMR track-based translation layer. Considering the microarchitecture of tracks in IMR, *crash consistency* is still a significant topic in IMR. As explained by [80], to avoid data loss, host writes to the affected top tracks in IMR must be blocked during the RMW operation, as they would be overwritten when data is copied back to the tracks. Therefore, the work in [121] argues that the typical RMW update strategy ensures the crash consistency of the track rewrite process at the expense of further-deteriorated write performance. In contrast, the authors presented crash-consistent update strategy, namely ***move-on-modify*** (**MOM**), to completely substitute the typical RMW by moving to-be-rewritten data to a proper free top track without taking extra data migrations.

9.2.2 Simulation and Emulation. In this category, we have examined two research studies [122, 241]. The study in [241] introduces an open-source IMR disk simulator implemented as a block device driver in the Linux kernel. This simulator replicates the interlaced tracks and integrates fundamental IMR-specific data placement algorithms such as the *two-stage* allocation strategy and the *three-phase* allocation strategy. It undergoes testing against CMR-based HDDs using real workloads to assess the I/O performance of IMR-based HDDs. Conversely, the research in [122] proposes a *drive-level* solution that mimics an IMR-based HDD on an actual CMR-based HDD for conducting realistic evaluations. The authors utilize the widely recognized benchmark tool *fio* to replay genuine I/O workloads collected by **Microsoft Research** (**MSR**) Cambridge on both the emulated IMR-based HDD and the genuine CMR-based HDD, subsequently gathering their performance outcomes.

9.2.3 Filesystems. Three studies have made significant contributions to IMR-specific file systems [28, 123, 126]. The study in [123] introduces a middleware termed *KVIMR*, which is an LSM-tree based key-value store designed for IMR-based HDDs. It introduces a Compaction-aware Track Allocation scheme and a Merged RMW approach aimed at enhancing efficiency and crash consistency. Evaluation entails emulating an IMR-based HDD using a real CMR-based HDD, configuring tracks in an interlaced layout, and implementing various track allocation schemes. The study presented in [126] introduces a file-system-aware data management approach termed as *FSIMR*, aimed at enhancing the write performance of IMR-based HDDs. This methodology involves logically partitioning the HDD into zones based on data directories, thereby reducing seek time and eliminating RMW operations. By considering both spatial locality and data access frequency, the approach improves write performance by organizing data according to file system characteristics. Specifically, it segregates frequently accessed hot data to upper tracks and less frequently accessed cold data to lower tracks, optimizing data placement to enhance overall performance. In the study outlined in [28], the authors tackle the issue of secure data deletion within IMR drives. Their focus is on addressing the inefficiencies associated with overwritten-based secure deletion techniques, which are exacerbated by the interlaced track layout of IMR drives. This layout significantly amplifies internal write traffic and diminishes deletion efficiency. To counter these challenges, the article proposes a strategy that leverages the track interference constraints inherent in IMR drives, coupled with a no-update journaling data stream.

10 Future and Conclusion

Looking ahead, the landscape of HDD and disk media technologies is set for significant advancement. One promising avenue for research is definitely HAMR with bit-patterned media, known as HDMR [111], which holds potential to enhance storage density and performance beyond current limits. This innovative approach, along with emerging technologies like HAMR, represents the next phase in the evolution of high-capacity and efficient storage solutions. In conclusion, our work represents a comprehensive exploration of SMR systems-specific research over the past 17 years. By examining the microarchitecture of various SMR disks, analyzing relevant research contributions to the field, and discussing the performance pitfalls inherent in SMR drives, we have provided a study that fills a significant gap in the storage community.

11 Main Glossary

The following terms, which are part of established scientific literature and have been previously published, were not coined by us; acronyms follow the commonly used in the existing articles.

(1) Advanced Storage Research Consortium - ASRC
(2) HDD - Hard disk drive

(3) CMR - Conventional Magnetic Recording
(4) SMR - Shingled Magnetic Recording
(5) STL - Shingled Translation Layer
(6) SPE - superparamagnetic effect
(7) ADC - areal density capacity
(8) DM-SMR - Drive-Managed SMR
(9) O-SMR - Out-of-place update SMR
(10) I-SMR - In-place update SMR
(11) HM-SMR - Host-Managed SMR
(12) HA-SMR - Host-Aware SMR
(13) HAMR - Heat Assisted Magnetic Recording
(14) H-SMR - Hybrid Shingled Magnetic Recording
(15) S-HAMR - Shingled Heat Assisted Magnetic Recording
(16) HIMR - Heat Interlaced Magnetic Recording
(17) HDMR - Heated Dot Magnetic Recording
(18) ZBC - Zoned Block Commands
(19) ZAC - Zoned Device ATA Command
(20) SFS - Shingled Filesystem
(21) TDMR - Two-dimensional Magnetic Recording

Acknowledgment

The authors would like to thank the reviewers for the meticulous reviews and insightful suggestions which have significantly improved the quality of this manuscript. The authors also extend their gratitude to the guest editors for the invitation to contribute this article.

References

[1] Adrian Palmer. 2015. FS Design Around SMR Seagate's Journey with EXT4. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC15_presentations/smr/AdrianPalmer_FS_Design_Around_SMR-nc.pdf. (2015).

[2] Abutalib Aghayev. 2020. *Adopting Zoned Storage in Distributed Storage Systems*. Ph.D. Dissertation. Carnegie Mellon University.

[3] Abutalib Aghayev and Peter Desnoyers. 2015. Skylight—A window on shingled disk operation. In *Proceedings of the 13th USENIX Conference on File and Storage Technologies*. USENIX Association, Santa Clara, CA, 135–149. Retrieved from https://www.usenix.org/conference/fast15/technical-sessions/presentation/aghayev

[4] Abutalib Aghayev, Mansour Shafaei, and Peter Desnoyers. 2015. Skylight–a window on shingled disk operation. *ACM Transactions on Storage* 11, 4 (2015), 28 pages. DOI: https://doi.org/10.1145/2821511

[5] Abutalib Aghayev, Theodore Ts'o, Garth Gibson, and Peter Desnoyers. 2017. Evolving ext4 for shingled disks. In *Proceedings of the 15th USENIX Conference on File and Storage Technologies*. USENIX Association, Santa Clara, CA, 105–120. Retrieved from https://www.usenix.org/conference/fast17/technical-sessions/presentation/aghayev

[6] Abutalib Aghayev, Theodore Ts'o, Garth A. Gibson, and Peter Desnoyers. 2016. *Ext4-zcj: An Evolved Journal Optimized for Drive-Managed Shingled Magnetic Recording Disks*. Technical Report. Carnegie Mellon University, Google Inc., Northeastern University, USA. Retrieved from https://api.semanticscholar.org/CorpusID:20083434. Technical Report.

[7] Abutalib Aghayev, Sage Weil, Michael Kuchnik, Mark Nelson, Gregory R. Ganger, and George Amvrosiadis. 2020. The case for custom storage backends in distributed storage systems. *ACM Transactions on Storage* 16, 2 (2020), 31 pages. DOI: https://doi.org/10.1145/3386362

[8] Nitin Agrawal, Vijayan Prabhakaran, Ted Wobber, John D. Davis, Mark Manasse, and Rina Panigrahy. 2008. Design tradeoffs for SSD performance. In *Proceedings of the USENIX 2008 Annual Technical Conference*. USENIX Association, USA, 57–70.

[9] Alan Chen. 2015. Integrating Cooperative Flash Management with SMR Technology for Optimized Tiering in Hybrid Systems. Retrieved January 01, 2024 from https://www.snia.org/educational-library/integrating-cooperative-flash-management-smr-technology-optimized-tiering-hybrid. (2015).

[10] Albert Chen. 2020. Improve Distributed Storage System Total Cost of Ownership with Host-Managed S M R HDDs. Retrieved January 01, 2024 from https://www.snia.org/educational-library/improve-distributed-storage-system-tco-host-managed-smr-hdds-2020. (2020).

[11] Albert Chen,Jim Malina,TK Kato. 2015. Host Managed SMR. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC15_presentations/smr/AlbertChen_JimMalina_Host_Managed_SMR_revision5.pdf. (2015).

[12] Goncalo Albuquerque, Stephanie Hernandez, Mark T. Kief, Daniele Mauri, and Lei Wang. 2022. HDD reader technology roadmap to an areal density of 4 tbpsi and beyond. *IEEE Transactions on Magnetics* 58, 2 (2022), 1–10. DOI: https://doi.org/10.1109/TMAG.2021.3081042

[13] Alibaba Group Holding Limited. 2024. Alibaba. Retrieved October 25, 2024 from https://www.alibabagroup.com. (2024).

[14] Ahmed Amer, JoAnne Holliday, Darrell D. E. Long, Ethan L. Miller, Jehan-François Paris, and Thomas Schwarz. 2011. Data management and layout for shingled magnetic recording. *IEEE Transactions on Magnetics* 47, 10 (2011), 3691–3697. DOI: https://doi.org/10.1109/TMAG.2011.2157115

[15] Ahmed Amer, Darrell D. E. Long, Ethan L. Miller, Jehan-François Pâris, and S. J. Thomas Schwarz. 2010. Design issues for a shingled write disk system. In *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies*. IEEE, New York, NY, USA, 1–12. DOI: https://doi.org/10.1109/MSST.2010.5496991

[16] Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, Lakshmi N. Bairavasundaram, Timothy E. Denehy, Florentina I. Popovici, Vijayan Prabhakaran, and Muthian Sivathanu. 2006. Semantically-smart disk systems: Past, present, and future. *ACM SIGMETRICS Performance Evaluation Review* 33, 4 (2006), 29–35. DOI: https://doi.org/10.1145/1138085.1138093

[17] Bill Boyle and Curtis E. Stevens Western Digital. 2017. Realms API. Retrieved from Retrieved January 01, 2024 from http://www.t10.org/cgi-bin/ac.pl?t=d&f=17-158r1.pdf. (2017).

[18] Trevor Blackwell, Jeffrey Harris, and Margo Seltzer. 1995. Heuristic cleaning algorithms in log-structured file systems. In *Proceedings of the USENIX 1995 Technical Conference Proceedings*. USENIX Association, USA, 23.

[19] Eric Brewer, Lawrence Ying, Lawrence Greenfield, Robert Cypher, and Theodore T.so. 2016. Disks for data centers. *White paper, Version 1.1, Google Inc., presented at FAST'16*. http://research.google.com/pubs/pub44830.html. Revised February 29, 2016.

[20] Zhichao Cao, Hao Wen, Fenggang Wu, and David H.C. Du. 2023. SMRTS: A performance and cost-effectiveness optimized SSD-SMR tiered file system with data deduplication. In *Proceedings of the 2023 IEEE 41st International Conference on Computer Design*. IEEE, USA, 275–282. DOI: https://doi.org/10.1109/ICCD58817.2023.00050

[21] Yuval Cassuto, Marco A. A. Sanvido, Cyril Guyot, David R. Hall, and Zvonimir Z. Bandic. 2010. Indirection systems for shingled-recording disk drives. In *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies*. IEEE, USA, 1–14. DOI: https://doi.org/10.1109/MSST.2010.5496971

[22] Ceph Developers. 2024. Zoned Storage in Ceph. (2024). Retrieved November 04, 2024 from https://docs.ceph.com/en/reef/dev/zoned-storage/

[23] Kheong Sann Chan, Rathnakumar Radhakrishnan, Kwaku Eason, Moulay Rachid Elidrissi, Jim J. Miles, Bane Vasic, and Anantha Raman Krishnan. 2010. Channel models and detectors for two-dimensional magnetic recording. *IEEE Transactions on Magnetics* 46, 3 (2010), 804–811. DOI: https://doi.org/10.1109/TMAG.2009.2035635

[24] Jung-Hsiu Chang, Tzu-Yu Chang, Yi-Chao Shih, and Tseng-Yi Chen. 2023. LaDy: Enabling locality-aware deduplication technology on shingled magnetic recording drives. *ACM Transactions on Embedded Computing Systems* 22, 5s (2023), 25 pages. DOI: https://doi.org/10.1145/3607921

[25] Chih-Hsuan Chen, Shuo-Han Chen, Yu-Pei Liang, Tseng-Yi Chen, Tsan sheng Hsu, Hsin-Wen Wei, and Wei-Kuan Shih. 2021. Facilitating external sorting on SMR-based large-scale storage systems. *Future Generation Computer Systems* 116, May (2021), 333–348. https://doi.org/10.1016/j.future.2020.10.032

[26] Peter M. Chen, Edward K. Lee, Garth A. Gibson, Randy H. Katz, and David A. Patterson. 1994. RAID: High-performance, reliable secondary storage. *ACM Computing Surveys* 26, 2 (1994), 145–185. DOI: https://doi.org/10.1145/176979.176981

[27] Ping-Xiang Chen, Shuo-Han Chen, Yuan-Hao Chang, Yu-Pei Liang, and Wei-Kuan Shih. 2021. Facilitating the efficiency of secure file data and metadata deletion on SMR-based Ext4 file system. In *Proceedings of the 26th Asia and South Pacific Design Automation Conference*. Association for Computing Machinery, New York, NY, USA, 728–733. DOI: https://doi.org/10.1145/3394885.3431517

[28] Shuo-Han Chen and Kuo-Hao Huang. 2023. Leveraging journaling file system for prompt secure deletion on interlaced recording drives. *IEEE Transactions on Emerging Topics in Computing* 11, 3 (2023), 619–634. DOI: https://doi.org/10.1109/TETC.2022.3226620

[29] Shuo-Han Chen, Wei-Shin Li, Min-Hong Shen, Yi-Han Lien, Tseng-Yi Chen, Tsan-sheng Hsu, Hsin-Wen Wei, and Wei-Kuan Shih. 2017. An update-overhead-aware caching policy for write-optimized file systems on SMR disks. In *Proceedings of the 2017 IEEE 36th International Performance Computing and Communications Conference*. IEEE, USA, 1–2. DOI: https://doi.org/10.1109/PCCC.2017.8280486

[30] Shuo-Han Chen, Yuhong Liang, and Ming-Chang Yang. 2022. KVSTL: An application support to LSM-tree based key-value store via shingled translation layer data management. *IEEE Transactions on Computers* 71, 7 (2022), 1598–1611. DOI: https://doi.org/10.1109/TC.2021.3098961

[31] Shuo-Han Chen, Yong-Ching Lin, Yuan-Hao Chang, Ming-Chang Yang, Tseng-Yi Chen, Hsin-Wen Wei, and Wei-Kuan Shih. 2019. A new sequential-write-constrained cache management to mitigate write amplification for SMR drives. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing.* Association for Computing Machinery, New York, NY, USA, 599–606. DOI: https://doi.org/10.1145/3297280.3297336

[32] Shuo-Han Chen, Chun-Feng Wu, Ming-Chang Yang, and Yuan-Hao Chang. 2022. A file-oriented fast secure deletion strategy for shingled magnetic recording drives. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 41, 8 (2022), 2463–2476. DOI: https://doi.org/10.1109/TCAD.2021.3112640

[33] Shuo-Han Chen, Ming-Chang Yang, Yuan-Hao Chang, and Chun-Feng Wu. 2019. Enabling file-oriented fast secure deletion on shingled magnetic recording drives. In *Proceedings of the 2019 56th ACM/IEEE Design Automation Conference.* IEEE, USA, 1–6.

[34] Yufeng Chen, Yao Wang, Yushu Xu, and Lei Chen. 2022. Detection scheme with joint intertrack interference and media noise mitigations for heat assisted interlaced magnetic recording. *IEEE Access* 10 (2022), 103963–103974. DOI: https://doi.org/10.1109/ACCESS.2022.3210330

[35] Y. J. Chen, S. H. Leong, B. X. Xu, and H. Z. Yang. 2014. Challenges and implementation approaches of shingled heat assisted magnetic recording (HAMR) for 10 Tb/In2 data storage. In *Proceedings of the Magnetics Symposium 2014 - Celebrating 50th Anniversary of IEEE Magnetics Society.* IEEE, USA, 1–2. DOI: https://doi.org/10.1109/MSSC.2014.6947937

[36] Zhengang Chen, Guohui Wang, Zhiping Shi, Yong Guan, and Tianyu Wang. 2023. Region-based flash caching with joint latency and lifetime optimization in hybrid SMR storage systems. In *Proceedings of the 2023 Design, Automation and Test in Europe Conference and Exhibition.* IEEE, Europe, 1–6. DOI: https://doi.org/10.23919/DATE56975.2023.10137148

[37] Chi-Young Ku and Stephen P. Morgan. 2015. An SMR-aware Append-only File System. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC15_presentations/smr/StephenMorgan-An_SMR-Aware_Append_Only-FiveSystem.pdf. (2015).

[38] S. N. Piramanayagam; Tow C. Chong. 2012. *Developments in Data Storage: Materials Perspective.* Wiley-IEEE Press, IEEE Press 445 Hoes LanePiscataway, NJ 08854, 352.

[39] Chris Mellor. December 3, 2021. Western Digital: Oh HAMR, how I do love thee. Retrieved January 01, 2024 from https://blocksandfiles.com/2021/12/03/western-digital-oh-hamr-how-i-do-love-thee/. (December 3, 2021).

[40] Chris Mellor. Jan 2024. 30TB HAMR disk drives get less shingled capacity increase than you'd think. Retrieved April 04, 2024 from https://blocksandfiles.com/2024/01/23/higher-disk-capacities-lower-shingling-capacity-addition/. (Jan 2024).

[41] Yi-Jing Chuang, Shuo-Han Chen, Yuan-Hao Chang, Yu-Pei Liang, Hsin-Wen Wei, and Wei-Kuan Shih. 2020. DSTL: A demand-based shingled translation layer for enabling adaptive address mapping on SMR drives. *ACM Transactions on Embedded Computing Systems* 19, 4 (2020), 21 pages. DOI: https://doi.org/10.1145/3391892

[42] Tae-Sun Chung, Dong-Joo Park, Sangwon Park, Dong-Ho Lee, Sang-Won Lee, and Ha-Joo Song. 2009. A survey of flash translation layer. *Journal of Systems Architecture* 55, 5–6 (2009), 332–343. DOI: https://doi.org/10.1016/j.sysarc.2009.03.005

[43] Western Digital Corporation. 2023. dm-zoned-tools. (2023). Retrieved October 31, 2024 from https://github.com/westerndigitalcorporation/dm-zoned-tools

[44] Tom Coughlin, Roger Hoyt, and Tom Coughlin. 2024. IEEE roadmap outlines development of mass digital storage technology. *Computer* 57, 9 (2024), 111–116. DOI: https://doi.org/10.1109/MC.2024.3416230

[45] Damien Le Moal. 2016. ZBC/ZAC Support in Linux. Retrieved January 01, 2024 from https://www.snia.org/educational-library/zbczac-support-linux-2016. (2016).

[46] Damien Le Moal. 2017. Linux SMR Support Status. Retrieved January 01, 2024 from https://events.static.linuxfound.org/sites/events/files/slides/lemoal-Linux-SMR-vault-2017.pdf. (2017).

[47] Eric D. Daniel, Mark H. Clark, and C. Denis Mee (Eds.). 1998. *Magnetic Recording: The First 100 Years* (1st ed.). Wiley-IEEE Press, New York, NY. Paperback, 370 pages.

[48] John Rydning David Reinsel, John Gantz. 2018. The digitization of the world: From edge to core. Data age 2025. Retrieved March 01, 2024 from https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf. (2018).

[49] Linux Kernel Developers. 2024. Device Mapper: Zoned Block Devices. (2024). Retrieved October 31, 2024 from https://docs.kernel.org/admin-guide/device-mapper/dm-zoned.html

[50] Benixon Arul Dhas, Erez Zadok, James Borden, and Jim Malina. 2014. *Evaluation of Nilfs2 for Shingled Magnetic Recording (SMR) Disks.* Technical Report FSL-14-03. Computer Science Department, Stony Brook University, Stony

Brook, NY, USA. https://api.semanticscholar.org/CorpusID:17254999. Technical Report FSL-14-03, jointly with Western Digital Corporation (Last accessed on 2024-01-01).

[51] Linux Kernel Documentation. 2018. CFQ I/O Scheduler. Retrieved October 24, 2024 from https://www.kernel.org/doc/Documentation/block/cfq-iosched.txt. (2018). The Linux Foundation.

[52] Draft Standard T10/BSR INCITS 536, American National Standards Institute, Inc. 2014. INCITS T10 Technical Committee. Information technology - Zoned Block Commands (ZBC). Retrieved January 01, 2024 from https://www.t10.org/ftp/zbcr01.pdf. (2014).

[53] Dropbox, Inc. 2024. Dropbox. Retrieved October 10, 2024 from https://www.dropbox.com. (2024).

[54] Ahmed El-Shimi, Ran Kalach, Ankit Kumar, Adi Ottean, Jin Li, and Sudipta Sengupta. 2012. Primary data deduplication—large scale study and system design. In *Proceedings of the 2012 USENIX Annual Technical Conference* USENIX Association, Boston, MA, 285–296. Retrieved from https://www.usenix.org/conference/atc12/technical-sessions/presentation/el-shimi

[55] Evangelos Eleftheriou, Robert Haas, Jens Jelitto, Mark Lantz, and Haralampos Pozidis. 2010. Trends in storage technologies. *IEEE Data Engineering Bulletin* 33, 4 (2010), 4–13.

[56] Robert M. Fallone and William B. Boyle. 2013. Data storage device employing a run-length mapping table and a single address mapping Table. U.S. Patent 8,443,167. https://patents.google.com/patent/US8443167

[57] Timothy Richard Feldman. 2012. Dynamic storage regions. U.S. Patent Application US20120210092A1. https://patents.google.com/patent/US20120210092A1

[58] R. E. Fontana, G. M. Decad, and S. R. Hetzler. 2013. The impact of areal density and millions of square inches (MSI) of produced memory on petabyte shipments of TAPE, NAND flash, and HDD storage class memories. In *Proceedings of the 2013 IEEE 29th Symposium on Mass Storage Systems and Technologies*. IEEE, USA, 1–8. DOI: https://doi.org/10.1109/MSST.2013.6558421

[59] Eran Gal and Sivan Toledo. 2005. Algorithms and data structures for flash memories. *ACM Computing Surveys* 37, 2 (2005), 138–163. DOI: https://doi.org/10.1145/1089733.1089735

[60] K. Gao. 2018. Data temperature and future data storage architecture. In *Proceedings of the 2018 IEEE International Magnetics Conference*. IEEE, USA, 1–2. DOI: https://doi.org/10.1109/INTMAG.2018.8508525

[61] Kaizhong Gao, Wenzhong Zhu, and Edward Gage. 2016. Write Management for Interlaced Magnetic Recording Devices. U.S. Patent US9508362B2. https://patents.google.com/patent/US9508362B2

[62] Shayan Garani and Bane Vasić. 2023. Channels engineering in magnetic recording: from theory to practice. *IEEE BITS the Information Theory Magazine* 3, 3 (2023), 6–49. https://doi.org/10.1109/MBITS.2023.3336213

[63] Garth Gibson. 2012. Storage Systems for Shingled Disks. Retrieved January 01, 2024 from https://www.snia.org/sites/default/orig/SDC2012/presentations/Gen_Sessions/GarthGibson_Storage_System_for_Shingled_Disks.pdf. (2012).

[64] Garth Gibson and Greg Ganger. 2011. Principles of Operation for Shingled Disk Devices. https://www.usenix.org/conference/hotstorage11/principles-operation-shingled-disk-devices (Last accessed on 2024-01-01).

[65] Garth A. Gibson and Milo Polte. 2009. Directions for Shingled-Write and Two-Dimensional Magnetic Recording System Architectures: Synergies with Solid-State Disks. Technical Report CMU-PDL-09-104. Carnegie Mellon University, Parallel Data Lab, USA. https://api.semanticscholar.org/CorpusID:59915322 (Last accessed on 2024-01-01).

[66] Steven Granz, Michael Conover, Javier Guzman, William Cross, Pete Harllee, and Tim Rausch. 2019. Perpendicular interlaced magnetic recording. *IEEE Transactions on Magnetics* 55, 12 (2019), 1–5. DOI: https://doi.org/10.1109/TMAG.2019.2936658

[67] Steven Granz, Jason Jury, Chris Rea, Ganping Ju, Jan-Ulrich Thiele, Tim Rausch, and Edward C. Gage. 2019. Areal density comparison between conventional, shingled, and interlaced heat-assisted magnetic recording with multiple sensor magnetic recording. *IEEE Transactions on Magnetics* 55, 3 (2019), 1–3. DOI: https://doi.org/10.1109/TMAG.2018.2869046

[68] Steven Granz, Chris Rea, Cathy Jackson, Mark Gubbins, Ganping Ju, Jan-Ulrich Thiele, and Tim Rausch. 2020. Heat assisted magnetic recording dependence on reader for conventional and shingled magnetic recording. In *Proceedings of the 2020 IEEE 31st Magnetic Recording Conference*. IEEE, USA, 1–2. DOI: https://doi.org/10.1109/TMRC49521.2020.9366715

[69] Steven Granz, Chris Rea, Ganping Ju, Peter Czoschke, and Stephanie Hernandez. 2023. Heat assisted magnetic recording areal density dependence on writer current for conventional and shingled magnetic recording. In *Proceedings of the 2023 IEEE 34th Magnetic Recording Conference*. IEEE, USA, 1–2. DOI: https://doi.org/10.1109/TMRC59626.2023.10264023

[70] Steven Granz, Wenzhong Zhu, Edmun Chian Song Seng, Utt Heng Kan, Chris Rea, Ganping Ju, Jan-Ulrich Thiele, Tim Rausch, and Edward C. Gage. 2018. Heat-assisted interlaced magnetic recording. *IEEE Transactions on Magnetics* 54, 2 (2018), 1–4. DOI: https://doi.org/10.1109/TMAG.2017.2748719

[71] Simon Greaves, Yasushi Kanai, and Hiroaki Muraoka. 2009. Shingled recording for 2–3 tbit/in^2. *IEEE Transactions on Magnetics* 45, 10 (2009), 3823–3829. DOI: https://doi.org/10.1109/TMAG.2009.2021663

[72] S.J. Greaves, H. Muraoka, and Y. Kanai. 2008. Simulations of recording media for 1Tb/in2. *Journal of Magnetism and Magnetic Materials* 320, 22 (2008), 2889–2893. DOI: https://doi.org/10.1016/j.jmmm.2008.08.094. Eighth Perpendicular Magnetic Recording Conference.

[73] Simon John Greaves, Yasushi Kanai, and Hiroaki Muraoka. 2014. Shingled thermally assisted magnetic recording for 8 Tbit/in ^{2}. *IEEE Transactions on Magnetics* 50, 11 (2014), 1–4. DOI: https://doi.org/10.1109/TMAG.2014.2318327

[74] E. Grochowski and D.A. Thompson. 1994. Outlook for maintaining areal density growth in magnetic recording. *IEEE Transactions on Magnetics* 30, 6 (1994), 3797–3800. DOI: https://doi.org/10.1109/20.333908

[75] Edward G. Grochowski. 1999. Emerging Trends in Data Storage on Magnetic Hard Disk Drives. https://api.semanticscholar.org/CorpusID:17582673 (Last accessed on 2024-01-01).

[76] Mohammad Hossein Hajkazemi. 2021. *High-Performance Translation Layers for Cloud Immutable Storage*. Ph.D. Dissertation. Northeastern University, USA. Advisor(s) Peter, Desnoyers. AAI28717244.

[77] Mohammad Hossein Hajkazemi, Mania Abdi, and Peter Desnoyers. 2018. Minimizing read seeks for SMR disk. In *Proceedings of the 2018 IEEE International Symposium on Workload Characterization (IISWC)*. IEEE, USA, 146–155. DOI: https://doi.org/10.1109/IISWC.2018.8573498

[78] Mohammad Hossein Hajkazemi, Mania Abdi, and Peter Desnoyers. 2020. uCache: A mutable cache for SMR translation layer. In *Proceedings of the 2020 28th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, USA, 1–8. DOI: https://doi.org/10.1109/MASCOTS50786.2020.9285939

[79] Mohammad Hossein Hajkazemi, Mania Abdi, Mansour Shafaei, and Peter Desnoyers. 2018. FSTL: A framework to design and explore shingled magnetic recording translation layers. In *Proceedings of the 2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, USA, 40–52. DOI: https://doi.org/10.1109/MASCOTS.2018.00012

[80] Mohammad Hossein Hajkazemi, Ajay Narayan Kulkarni, Peter Desnoyers, and Timothy R Feldman. 2019. Track-based translation layers for interlaced magnetic recording. In *Proceedings of the 2019 USENIX Annual Technical Conference*. USENIX Association, Renton, WA, 821–832. Retrieved from http://www.usenix.org/conference/atc19/presentation/hajkazemi

[81] David Hall, John H. Marcos, and Jonathan D. Coker. 2012. Data handling algorithms for autonomous shingled magnetic recording HDDs. *IEEE Transactions on Magnetics* 48, 5 (2012), 1777–1781. DOI: https://doi.org/10.1109/TMAG.2011.2179528

[82] David Robison Hall. 2014. Shingle-written magnetic recording (SMR) device with hybrid E-region. (2014). US Patent 8,687,303.

[83] Hannes Reinecke. 2015. Strategies for running unmodified filesystems on SMR drives. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC15_presentations/smr/HannesReinecke_Strategies_for_running_unmodified_FS_SMR.pdf. (2015).

[84] Hannes Reinecke. 2020. High-performance SMR Drives with dm-zoned. Retrieved January 01, 2024 from https://www.snia.org/educational-library/high-performance-smr-drives-dm-zoned-caching-2020. (2020).

[85] Jingpeng Hao, Xubin Chen, Yifan Qiao, Yuyang Zhang, and Tong Zhang. 2020. On the design of SMR HDD block device driver. In *Proceedings of the 2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems*. IEEE, USA, 1291–1299. DOI: https://doi.org/10.1109/HPCC-SmartCity-DSS50907.2020.00166

[86] Jun He, Sudarsun Kannan, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. 2017. The unwritten contract of solid state drives. In *Proceedings of the 12th European Conference on Computer Systems*. Association for Computing Machinery, New York, NY, USA, 127–144. DOI: https://doi.org/10.1145/3064176.3064187

[87] Weiping He and David H.C. Du. 2017. SMaRT: An approach to shingled magnetic recording translation. In *Proceedings of the 15th USENIX Conference on File and Storage Technologies*. USENIX Association, Santa Clara, CA, 121–134. Retrieved from https://www.usenix.org/conference/fast17/technical-sessions/presentation/he

[88] Weiping He and David H. C. Du. 2014. Novel address mappings for shingled write disks. In *Proceedings of the 6th USENIX Conference on Hot Topics in Storage and File Systems*. USENIX Association, USA, 5.

[89] Eric Herzog. 2020. IBM Storage builds your journey to AI. (2020). Retrieved October 20, 2024 from https://www.linkedin.com/pulse/ibm-storage-builds-your-journey-ai-eric-herzog

[90] HGST, Seagate, Toshiba, Western Digital T10 SMR Study Group. 2013. SMR Standardization. Retrieved January 01, 2024 from https://www.digitalpreservation.gov/meetings/documents/storage13/DaveAnderson_Standards.pdf. (2013).

[91] Wei-Heng Hsu and R. Victora. 2021. Rotated read head design for high-density heat-assisted shingled magnetic recording. *Applied Physics Letters* 118, 7 (02 2021), 072406. https://doi.org/10.1063/5.0044042

[92] Wei-Heng Hsu and R.H. Victora. 2022. Heat-assisted magnetic recording – micromagnetic modeling of recording media and areal density: A review. *Journal of Magnetism and Magnetic Materials* 563 (2022), 169973. DOI: https://doi.org/10.1016/j.jmmm.2022.169973

[93] Euiseok Hwang. 2017. Asynchronous inter-track interference cancellation (A-ITIC) for interlaced magnetic recording (IMR). In *Proceedings of the GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. IEEE, USA, 1–6. DOI: https://doi.org/10.1109/GLOCOM.2017.8255007

[94] Euiseok Hwang, Jongseung Park, Richard Rauschmayer, and Bruce Wilson. 2017. Interlaced magnetic recording. *IEEE Transactions on Magnetics* 53, 4 (2017), 1–7. DOI: https://doi.org/10.1109/TMAG.2016.2638809

[95] INCITS T13 Technical Committee. 2015. Information technology - Zoned Device ATA Command Set (ZAC). Draft Standard T13/BSR INCITS 537. Retrieved January 01, 2024 from http://www.t13.org/Documents/UploadedDocuments/docs2015/di53. (2015).

[96] J. Campello. 2013. Shingled Magnetic Recording (SMR) Intro- duction. Retrieved January 01, 2024 from http://www.t10.org/cgi-bin/ac.pl?t=d&f=13-148r0.pdf. (2013).

[97] Bruce Jacob, Spencer Ng, and David Wang. 2008. *Memory Systems: Cache, DRAM, Disk.* Morgan Kaufmann Publishers Inc., 340 Pine Street, Sixth FloorSan FranciscoCAUnited States.

[98] Chao Jin, Wei-Ya Xi, Zhi-Yong Ching, Feng Huo, and Chun-Teck Lim. 2014. HiSMRfs: A high performance file system for shingled storage array. In *Proceedings of the 2014 30th Symposium on Mass Storage Systems and Technologies* . IEEE, Santa Clara, CA, USA, 1–6. DOI: https://doi.org/10.1109/MSST.2014.6855539

[99] David Robison Hall Jonathan Darrel Coker. 2013. Indirection memory architecture with reduced memory requirements for shingled magnetic recording devices. (2013). US Patent 8,578,122.

[100] Stephanie N. Jones, Ahmed Amer, Ethan L. Miller, Darrell D. E. Long, Rekha Pitchumani, and Christina R. Strong. 2015. Classifying data to reduce long term data movement in shingled write disks. In *Proceedings of the 2015 31st Symposium on Mass Storage Systems and Technologies*. IEEE, New York, NY, USA, 1–9. DOI: https://doi.org/10.1109/MSST.2015.7208285

[101] Stephanie N. Jones, Ahmed Amer, Ethan L. Miller, Darrell D. E. Long, Rekha Pitchumani, and Christina R. Strong. 2016. Classifying data to reduce long-term data movement in shingled write disks. *ACM Transactions on Storage* 12, 1 (2016), 17 pages. DOI: https://doi.org/10.1145/2851505

[102] Jorge Campello. 2015. SMR:The Next Generation of Storage Technology. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC15_presentations/smr/JorgeCampello_SMR_Next_Generation_Storage_Technology.pdf. (2015).

[103] Saurabh Kadekodi, Swapnil Pimpale, and Garth A. Gibson. 2015. Caveat-scriptor: Write anywhere shingled disks. In *Proceedings of the 7th USENIX Conference on Hot Topics in Storage and File Systems.* USENIX Association, USA, 16.

[104] Kaizhong Gao, Wenzhong Zhu, and Edward Gage. 2017. Interlaced Magnetic Recording, US Patent 9,728,206. (2017).

[105] P. Kasiraj, R. New, J. de Souza, and M. Williams. 2009. System and method for writing data to dedicated bands of a hard disk drive. (2009). United States Patent 7,490,212.

[106] T. Kobayashi, Y. Nakatani, and Y. Fujiwara. 2023. Comparison of shingled heat-assisted magnetic recording and three-dimensional heat-assisted magnetic recording. *Journal of the Magnetics Society of Japan* 47, 1 (2023), 1–14. DOI: https://doi.org/10.3379/msjmag.2301R001

[107] Ioannis Koltsidas, Nikolas Ioannou, Kornilios Kourtis, and Thomas Weigold. 2015. *SoftwAre Log-Structured Array (SALSA) – A unified stack for SSDs and SMR disks.* RZ3892. IBM Research Technical Paper Archive.

[108] Elie Krevat, Joseph Tucek, and Gregory R. Ganger. 2011. Disks are like snowflakes: no two are alike. In *Proceedings of the 13th USENIX Conference on Hot Topics in Operating Systems.* USENIX Association, USA, 14.

[109] Mark H. Kryder, Edward C. Gage, Terry W. McDaniel, William A. Challener, Robert E. Rottmayer, Ganping Ju, Yiao-Tee Hsia, and M. Fatih Erden. 2008. Heat assisted magnetic recording. *Proc. IEEE* 96, 11 (2008), 1810–1835. DOI: https://doi.org/10.1109/JPROC.2008.2004315

[110] Mark H. Kryder and Chang Soo Kim. 2009. After hard drives–what comes next? *IEEE Transactions on Magnetics* 45, 10 (2009), 3406–3413. DOI: https://doi.org/10.1109/TMAG.2009.2024163

[111] David S. Kuo, Kim Y. Lee, XiaoMin Yang, Shuaigang Xiao, Yautzong Hsu, Zhaoning Yu, Michael Feldbaum, Tim Klemmer, Yukiko Kubota, Jan-Ulrich Thiele, Philip Steiner, Koichi Wago, Stefano Dallorto, and Deirdre Olynick. 2016. Heated dot magnetic recording media - path to 10 TDOTS/in2. In *Proceedings of the 2016 International Conference of Asian Union of Magnetics Societies.* IEEE, USA, 1–1. DOI: https://doi.org/10.1109/ICAUMS.2016.8479899

[112] Lawrence Ying,Ted Ts'o. 2017. Dynamic Hybrid-SMR: an OCP proposal to improve data center disk drives. Retrieved January 01, 2024 fromhttps://blog.google/products/google-cloud/dynamic-hybrid-smr-ocp-proposal-improve-data-center-disk-drives/. (2017).

[113] Quoc Le, Ahmed Amer, and JoAnne Holliday. 2019. RAID 4SMR: RAID array with shingled magnetic recording disk for mass storage systems. *Journal of Computer Science and Technology* 34, 4 (2019), 854–868. DOI: https://doi.org/10.1007/s11390-019-1946-4

[114] Quoc Minh Le. 2019. *Shingled Magnetic Recording Disks for Mass Storage Systems.* Ph.D. Dissertation. Santa Clara University. https://scholarcommons.scu.edu/eng_phd_theses/21 (Last accessed on 2024-01-01).

[115] Quoc Minh Le, Ahmed Amer, and JoAnne Holliday. 2015. SMR disks for mass storage systems. In *Proceedings of the 2015 IEEE 23rd International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, New York, NY, USA, 228–231. DOI: https://doi.org/10.1109/MASCOTS.2015.42

[116] Quoc M. Le, Kumar SathyanarayanaRaju, Ahmed Amer, and JoAnne Holliday. 2011. Workload impact on shingled write disks: All-writes can be alright. In *Proceedings of the 2011 IEEE 19th Annual International Symposium on Modelling, Analysis, and Simulation of Computer and Telecommunication Systems*. IEEE, New York, NY, USA, 444–446. DOI: https://doi.org/10.1109/MASCOTS.2011.58

[117] Damien Le Moal, Zvonimir Bandic, and Cyril Guyot. 2012. Shingled file system host-side management of shingled magnetic recording disks. In *Proceedings of the 2012 IEEE International Conference on Consumer Electronics*. IEEE, New York, NY, USA, 425–426. DOI: https://doi.org/10.1109/ICCE.2012.6161799

[118] Sang-Won Lee, Bongki Moon, Chanik Park, Jae-Myung Kim, and Sang-Woo Kim. 2008. A case for flash memory ssd in enterprise database applications. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. Association for Computing Machinery, New York, NY, USA, 1075–1086. DOI: https://doi.org/10.1145/1376616.1376723

[119] Yin Li, Xubin Chen, Ning Zheng, Jingpeng Hao, and Tong Zhang. 2019. An exploratory study on software-defined data center hard disk drives. *ACM Transactions on Storage* 15, 3 (2019), 22 pages. DOI: https://doi.org/10.1145/3319405

[120] Yuan Li, Yao Wang, Yushu Xu, Lei Chen, Yumei Wen, and Ping Li. 2021. Multitrack detection with 2-d iterative soft estimate aided neural network equalizer for heat-assisted interlaced magnetic recording. *IEEE Transactions on Magnetics* 57, 3 (2021), 1–8. DOI: https://doi.org/10.1109/TMAG.2020.3027304

[121] Yuhong Liang and Ming-Chang Yang. 2021. Move-on-modify: An efficient yet crash-consistent update strategy for interlaced magnetic recording. In *Proceedings of the 2021 58th ACM/IEEE Design Automation Conference (DAC)*. IEEE, New York, NY, USA, 97–102. DOI: https://doi.org/10.1109/DAC18074.2021.9586091

[122] Yuhong Liang, Ming-Chang Yang, and Shuo-Han Chen. 2022. MAGIC: Making IMR-based HDD perform like CMR-based HDD. *IEEE Transactions on Computers* 71, 3 (2022), 643–657. DOI: https://doi.org/10.1109/TC.2021.3059770

[123] Yuhong Liang, Tsun-Yu Yang, and Ming-Chang Yang. 2021. KVIMR: Key-value store aware data management middleware for interlaced magnetic recording based hard disk drive. In *Proceedings of the 2021 USENIX Annual Technical Conference*. USENIX Association, Berkeley, CA, 657–671. Retrieved from https://www.usenix.org/conference/atc21/presentation/liang

[124] Yu-Pei Liang, Shuo-Han Chen, Yuan-Hao Chang, Yong-Chin Lin, Hsin-Wen Wei, and Wei-Kuan Shih. 2019. Mitigating write amplification issue of SMR drives via the design of sequential-write-constrained cache. *Journal of Systems Architecture* 99, C (2019), 11. DOI: https://doi.org/10.1016/j.sysarc.2019.101634

[125] Yu-Pei Liang, Tseng-Yi Chen, Yuan-Hao Chang, Shuo-Han Chen, Kam-Yiu Lam, Wei-Hsin Li, and Wei-Kuan Shih. 2019. Enabling sequential-write-constrained B+-tree index scheme to upgrade shingled magnetic recording storage performance. *ACM Transactions on Embedded Computing Systems* 18, 5s (2019), 20 pages. DOI: https://doi.org/10.1145/3358201

[126] Yi-Han Lien, Yen-Ting Chen, Yuan-Hao Chang, Yu-Pei Liang, and Wei-Kuan Shih. 2023. FSIMR: File-system-aware data management for interlaced magnetic recording. *ACM Transactions on Embedded Computing Systems* 22, 5s (2023), 18 pages. DOI: https://doi.org/10.1145/3607922

[127] Fabian Lim, Bruce Wilson, and Roger Wood. 2010. Analysis of shingle-write readback using magnetic-force microscopy. *IEEE Transactions on Magnetics* 46, 6 (2010), 1548–1551. DOI: https://doi.org/10.1109/TMAG.2009.2038281

[128] Chung-I Lin, Dongchul Park, Weiping He, and David H.C. Du. 2012. H-SWD: Incorporating hot data identification into shingled write disks. In *Proceedings of the 2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*. IEEE, New York, NY, USA, 321–330. DOI: https://doi.org/10.1109/MASCOTS.2012.44

[129] Ting-Yu Lin and Tseng-Yi Chen. 2023. HSMR-RAID: Enabling a low overhead RAID-5 system over a host-managed shingled magnetic recording disk array. In *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing*. Association for Computing Machinery, New York, NY, USA, 294–296. DOI: https://doi.org/10.1145/3555776.3577820

[130] Wenguo Liu, Dan Feng, Lingfang Zeng, and Jianxi Chen. 2014. Understanding the SWD-based RAID system. In *Proceedings of the 2014 International Conference on Cloud Computing and Big Data*. IEEE, New York, NY, USA, 175–181. DOI: https://doi.org/10.1109/CCBD.2014.17

[131] Wenguo Liu, Hui Li, and Lingfang Zeng. 2021. STAR: A zone translation scheme to improve the performance of host-aware SMR. In *Proceedings of the 2021 IEEE 6th International Conference on Big Data Analytics*. IEEE, New York, NY, USA, 267–271. DOI: https://doi.org/10.1109/ICBDA51983.2021.9402952

[132] Wenguo Liu, Lingfang Zeng, and Dan Feng. 2016. Apas: An application aware hybrid storage system combining SSDs and SWDs. In *Proceedings of the 2016 IEEE International Conference on Networking, Architecture and Storage*. IEEE, USA, 1–4.

[133] Wenguo Liu, Lingfang Zeng, and Dan Feng. 2018. CASS: A cooperative hybrid storage system consisting of an SSD and a SMR drive. In *Proceedings of the 2018 6th International Conference on Advanced Cloud and Big Data*. IEEE, USA, 24–29. DOI: https://doi.org/10.1109/CBD.2018.00014

[134] Wenguo Liu, Lingfang Zeng, and Dan Feng. 2018. CLDM: A cache cleaning algorithm for host aware SMR drives. In *Proceedings of the Algorithms and Architectures for Parallel Processing*, Jaideep Vaidya and Jin Li (Eds.). Springer International Publishing, Cham, 608–620.

[135] Wen-Guo Liu, Ling-Fang Zeng, Dan Feng, and Kenneth Kent. 2019. ROCO: Using a solid state drive cache to improve the performance of a host-aware shingled magnetic recording drive. *Journal of Computer Science and Technology* 34 (2019), 61–76. DOI: https://doi.org/10.1007/s11390-019-1899-7

[136] Lockwood, GK Hazen, D Koziol, Q. 2020. Storage 2020: A Vision for the Future of HPC Storage. Retrieved January 01, 2024 from https://escholarship.org/content/qt744479dp/qt744479dp.pdf. (2020).

[137] Zheng-Wu Lu and Gang Zhou. 2016. Design and implementation of hybrid shingled recording RAID system. In *Proceedings of the 2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech)*. IEEE, USA, 937–942. DOI: https://doi.org/10.1109/DASC-PICom-DataCom-CyberSciTec.2016.161

[138] Dan Luo, Jiguang Wan, Yifeng Zhu, Nannan Zhao, Feng Li, and Changsheng Xie. 2016. Design and implementation of a hybrid shingled write disk system. *IEEE Transactions on Parallel and Distributed Systems* 27, 4 (2016), 1017–1029. DOI: https://doi.org/10.1109/TPDS.2015.2425402

[139] Dan Luo, Ting Yao, Xiaoyang Qu, Jiguang Wan, and Changsheng Xie. 2016. DVS: Dynamic variable-width striping RAID for shingled write disks. In *Proceedings of the 2016 IEEE International Conference on Networking, Architecture and Storage*. IEEE, USA, 1–10. DOI: https://doi.org/10.1109/NAS.2016.7549398

[140] luo qing chao,thunder zhang. 2015. Implement Object Storage with SMR based Key-Value Store. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC15_presentations/smr/QingchaoLuo_Implement_Object_Storage_SMR_Key-Value_Store.pdf. (2015).

[141] Chenlin Ma. 2019. *READ-MODIFY-WRITE Optimization for Shingled Magnetic Recording Storage Systems*. Ph.D. Dissertation. The Hong Kong Polytechnic University.

[142] Chenlin Ma, Zhaoyan Shen, Lei Han, and Zili Shao. 2019. FC: Built-in flash-cache with fast cleaning for SMR storage. In *Proceedings of the 2019 IEEE International Conference on Embedded Software and Systems*. IEEE, USA, 1–7. DOI: https://doi.org/10.1109/ICESS.2019.8782438

[143] Chenlin Ma, Zhaoyan Shen, Lei Han, and Zili Shao. 2019. RMW-F: A design of RMW-free cache using built-in NAND-flash for SMR storage. *ACM Transactions on Embedded Computing Systems* 18, 5s (2019), 18 pages. DOI: https://doi.org/10.1145/3358210

[144] Chenlin Ma, Zhaoyan Shen, Jihe Wang, Yi Wang, Renhai Chen, Yong Guan, and Zili Shao. 2021. Tiler: An autonomous region-based scheme for SMR storage. *IEEE Transactions on Computers* 70, 2 (2021), 291–304. DOI: https://doi.org/10.1109/TC.2020.2988004

[145] Chenlin Ma, Zhaoyan Shen, Yi Wang, and Zili Shao. 2019. Alleviating hot data write back effect for shingled magnetic recording storage systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 38, 12 (2019), 2243–2254. DOI: https://doi.org/10.1109/TCAD.2018.2878190

[146] Chenlin Ma, Yi Wang, Zhaoyan Shen, and Zili Shao. 2020. KFR: Optimal cache management with k-framed reclamation for drive-managed SMR disks. In *Proceedings of the 2020 57th ACM/IEEE Design Automation Conference*. ACM/IEEE, USA, 1–6. DOI: https://doi.org/10.1109/DAC18072.2020.9218636

[147] Chenlin Ma, Zhuokai Zhou, Yingping Wang, Yi Wang, and Rui Mao. 2022. MAID-Q: Minimizing tail latency in embedded flash with SMR disk via -learning model. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 41, 11 (2022), 3709–3720. DOI: https://doi.org/10.1109/TCAD.2022.3197498

[148] Chenlin Ma, Zhuokai Zhou, Yingping Wang, Yi Wang, and Rui Mao. 2022. MU-RMW: Minimizing unnecessary RMW operations in the embedded flash with SMR disk. In *Proceedings of the 2022 Design, Automation and Test in Europe Conference and Exhibition*. IEEE, USA, 490–495. DOI: https://doi.org/10.23919/DATE54114.2022.9774638

[149] Liuying Ma and Lu Xu. 2016. HMSS: A high performance host-managed shingled storage system based on awareness of SMR on block layer. In *Proceedings of the 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems*. IEEE, USA, 570–577. DOI: https://doi.org/10.1109/HPCC-SmartCity-DSS.2016.0086

[150] Peter Macko, Xiongzi Ge, John Haskins Jr., James Kelley, David Slik, Keith A. Smith, and Maxim G. Smith. 2017. SMORE: A cold data object store for SMR drives. *Short Paper in 33rd International Conference on Massive Storage Systems and Technology (MSST)*. https://msstconference.org/MSST-history/2017/Papers/ColdDataObjectStore.pdf. Accessed on 2024-01-01.

[151] Magic Pocket Hardware Engineering Teams. 2018. Extending Magic Pocket Innovation with the first petabyte scale SMR drive deployment. Retrieved January 01, 2024 from https://dropbox.tech/infrastructure/extending-magic-pocket-innovation-with-the-first-petabyte-scale-smr-drive-deployment. (2018).
[152] M. Mallary, A. Torabi, and M. Benakli. 2002. One terabit per square inch perpendicular recording conceptual design. *IEEE Transactions on Magnetics* 38, 4 (2002), 1719–1724. DOI: https://doi.org/10.1109/TMAG.2002.1017762
[153] Abdullah Mamun, Guoxiao Guo, and Chao Bi. 2006. *Block Trace Analysis and Storage System Optimization A Practical Approach with MATLAB/Python Tools.* CRC Press; 1st edition (November 2, 2006), 6000 Broken Sound Parkway NW, Suite 300 Boca Raton, FL 33487-2742, 398.
[154] Nagapramod Mandagere, Pin Zhou, Mark A. Smith, and Sandeep Uttamchandani. 2008. Demystifying data deduplication. In *Proceedings of the ACM/IFIP/USENIX Middleware '08 Conference Companion.* Association for Computing Machinery, New York, NY, USA, 12–17. DOI: https://doi.org/10.1145/1462735.1462739
[155] Adam Manzanares, Noah Watkins, Cyril Guyot, Damien LeMoal, Carlos Maltzahn, and Zvonimir Bandic. 2016. ZEA, a data management approach for SMR. In *Proceedings of the 8th USENIX Conference on Hot Topics in Storage and File Systems.* USENIX Association, USA, 26–30.
[156] Mary Dunn and Timothy Feldman. 2014. Shingled Magnetic Recording Models, Standardization, and Applications. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/Dunn-Feldman_SNIA_Tutorial_Shingled_Magnetic_Recording-r7_Final.pdf. (2014).
[157] Chris Mellor. 2020. Huawei packs OceanStor Pacific array with 20TB SMR drives. Retrieved November 04, 2024 from https://blocksandfiles.com/2020/06/01/huawei-0ceanstor-pacific-array20tb-smr-drive/. (2020).
[158] Dutch T. Meyer and William J. Bolosky. 2012. A study of practical deduplication. *ACM Transactions on Storage* 7, 4 (2012), 20 pages. DOI: https://doi.org/10.1145/2078861.2078864
[159] Damien Le Moal. 2017. dm-zoned: Zoned block device target. https://lwn.net/Articles/714387/ (Last accessed on 2024-01-01).
[160] Jeff Moyer. 2018. block: default to deadline for host-managed SMR devices. https://patchwork.kernel.org/project/linux-block/patch/20180525211432.20359-3-jmoyer@redhat.com/. Patchwork Message ID: 20180525211432.20359-3-jmoyer@redhat.com (Last accessed on 2024-01-01).
[161] Muhammad Ahmad,Gregory Frazier Seagate Technology. 2016. Iometer for Zoned Block Devices Microbenchmarking SMR HDDs. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/SDC/2016/presentations/etc/Muhammad_Ahmad_Iometer_for_Zoned_Block_Devices.pdf. (2016).
[162] Junpeng Niu, Jun Xu, and Lihua Xie. 2018. Hybrid storage systems: A survey of architectures and algorithms. *IEEE Access* 6 (2018), 13385–13406. DOI: https://doi.org/10.1109/ACCESS.2018.2803302
[163] John Ousterhout and Fred Douglis. 1989. Beating the I/O bottleneck: A case for log-structured file systems. *SIGOPS Operating Systems Review* 23, 1 (Jan. 1989), 11–28. https://doi.org/10.1145/65762.65765
[164] Yungang Pan, Zhiping Jia, Zhaoyan Shen, Bingzhe Li, Wanli Chang, and Zili Shao. 2021. Reinforcement learning-assisted cache cleaning to mitigate long-tail latency in DM-SMR. In *Proceedings of the 2021 58th ACM/IEEE Design Automation Conference.* IEEE, USA, 103–108. DOI: https://doi.org/10.1109/DAC18074.2021.9586084
[165] David A. Patterson, Garth Gibson, and Randy H. Katz. 1988. A case for redundant arrays of inexpensive disks (RAID). *SIGMOD Record* 17, 3 (1988), 109–116. DOI: https://doi.org/10.1145/971701.50214
[166] João Paulo and José Pereira. 2014. A survey and classification of storage deduplication systems. *ACM Computing Surveys* 47, 1 (2014), 30 pages. DOI: https://doi.org/10.1145/2611778
[167] Piotr Modrzyk. 2023. Bridging the Gap Between Host Managed SMR Drives and Software-Defined Storage. Retrieved January 01, 2024 from https://www.kernel.org/doc/html/latest/block/bfq-iosched.html. (2023).
[168] Rekha Pitchumani. 2012. *Shingled Write Disk Emulator.* Technical Report UCSC-SSRC-12-05. University of California, Santa Cruz.
[169] R. Pitchumani. 2015. Data Management For Shingled Magnetic Recording Disks. Ph.D. Thesis. University of California, Santa Cruz. https://escholarship.org/uc/item/65s546pq. ProQuest ID: Pitchumani_ucsc_0036E_10944, Merritt ID: ark:/13030/m5sf6cxx (Last accessed on 2024-01-01).
[170] Rekha Pitchumani, Andy Hospodor, Ahmed Amer, Yangwook Kang, Ethan L. Miller, and Darrell D.E. Long. 2012. Emulating a shingled write disk. In *Proceedings of the 2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems.* IEEE, USA, 339–346. DOI: https://doi.org/10.1109/MASCOTS.2012.46
[171] Rekha Pitchumani, James Hughes, and Ethan L. Miller. 2015. SMRDB: Key-value data store for shingled magnetic recording disks. In *Proceedings of the 8th ACM International Systems and Storage Conference.* Association for Computing Machinery, New York, NY, USA, 11 pages. DOI: https://doi.org/10.1145/2757667.2757680
[172] Milo Polte, Jiri Simsa, and Garth Gibson. 2008. Comparing performance of solid state devices and mechanical disks. In *Proceedings of the 2008 3rd Petascale Data Storage Workshop.* IEEE, USA, 1–7. DOI: https://doi.org/10.1109/PDSW.2008.4811886

[173] Quoc M. Le JoAnne Holliday Ahmed Amer. 2012. The Peril and Promise of Shingled Disk Arrays. Retrieved January 01, 2024 from https://www.usenix.org/legacy/events/fast12/poster_descriptions/Ledescription.pdf. (2012).

[174] Raj Das. 2017. ReFS Support For SMR Drives. RetrievedJanuary01,2024from{https://www.snia.org/sites/default/files/SDC/2017/presentations/smr/Das_Rajsekhar_ReFS_Support_For_Shingled_Magnetic_Recording_Drives.pdf}. (2017).

[175] Zebin Ren, Krijn Doekemeijer, Nick Tehrany, and Animesh Trivedi. 2024. BFQ, multiqueue-deadline, or kyber? Performance characterization of linux storage schedulers in the NVMe era. In *Proceedings of the 15th ACM/SPEC International Conference on Performance Engineering.* Association for Computing Machinery, New York, NY, USA, 154–165. DOI: https://doi.org/10.1145/3629526.3645053

[176] H.J. Richter, A.Y. Dobin, O. Heinonen, K.Z. Gao, R.J.M. v.d. Veerdonk, R.T. Lynch, J. Xue, D. Weller, P. Asselin, M.F. Erden, and R.M. Brockie. 2006. Recording on bit-patterned media at densities of 1 Tb/in^{2}and beyond. *IEEE Transactions on Magnetics* 42, 10 (2006), 2255–2260. DOI: https://doi.org/10.1109/TMAG.2006.878392

[177] Daniel Robinson. 2020. IBM details storage portfolio for AI infrastructure. (2020). Retrieved October 20, 2024 from https://blocksandfiles.com/2020/07/10/ibm-ai-data-storage-portfolio/

[178] Eric Roddick, Mark Kief, and Akihiko Takeo. 2022. A new advanced storage research consortium HDD technology roadmap. In *Proceedings of the 2022 IEEE 33rd Magnetic Recording Conference*. IEEE, USA, 1–2. DOI: https://doi.org/10.1109/TMRC56419.2022.9918580

[179] Mendel Rosenblum and John K. Ousterhout. 1992. The design and implementation of a log-structured file system. *ACM Transactions on Computer Systems* 10, 1 (1992), 26–52. DOI: https://doi.org/10.1145/146941.146943

[180] R.E. Rottmayer, S. Batra, D. Buechel, W.A. Challener, J. Hohlfeld, Y. Kubota, L. Li, B. Lu, C. Mihalcea, K. Mountfield, K. Pelhos, C. Peng, T. Rausch, M.A. Seigler, D. Weller, and X.-M. Yang. 2006. Heat-assisted magnetic recording. *IEEE Transactions on Magnetics* 42, 10 (2006), 2417–2421. DOI: https://doi.org/10.1109/TMAG.2006.879572

[181] Soror Sahri and S.J. Schwarz, Thomas. 2014. Design issues of shingled write disk for database table implementation. *Journal of Computers* 9, 10 (2014), 2247–2257.

[182] Seagate. 2013. Seagate Delivers On Technology Milestone: First To Ship Hard Drives Using Next-Generation Shingled Magnetic Recording. (2013). Retrieved October 31, 2024 from https://investors.seagate.com/news/news-details/2013/Seagate-Delivers-On-Technology-Milestone-First-To-Ship-Hard-Drives-Using-Next-Generation-Shingled-Magnetic-Recording/default.aspx

[183] Seagate. 2016. NAS HDD v2 SATA Product Manual ST8000VN000. (2016). Retrieved October 31, 2024 from https://www.seagate.com/www-content/product-content/nas-fam/nas-hdd/en-us/docs/100795783a.pdf

[184] Seagate Technology. 2018. New flex dynamic record ing method redefines the data center hard drive. Retrieved January 01, 2024 from https://blog.seagate.com/intelligent/new-flex-dynamic-recording-method-redefines-data-center-hard-drive/. (2018).

[185] Michael A. Seigler, William A. Challener, Edward Gage, Nils Gokemeijer, Ganping Ju, Bin Lu, Kalman Pelhos, Chubing Peng, Robert E. Rottmayer, Xiaomin Yang, Hua Zhou, and Tim Rausch. 2008. Integrated heat assisted magnetic recording head: Design and recording demonstration. *IEEE Transactions on Magnetics* 44, 1 (2008), 119–124. DOI: https://doi.org/10.1109/TMAG.2007.911029

[186] Margo Seltzer, Keith Bostic, Marshall Kirk Mckusick, and Carl Staelin. 1993. An implementation of a log-structured file system for UNIX. In *Proceedings of the USENIX Winter 1993 Conference Proceedings on USENIX Winter 1993 Conference Proceedings.* USENIX Association, USA, 3.

[187] Mansour Shafaei and Peter Desnoyers. 2017. Virtual guard: A track-based translation layer for shingled disks. In *Proceedings of the 9th USENIX Conference on Hot Topics in Storage and File Systems.* USENIX Association, USA, 18.

[188] Mansour Shafaei, Mohammad Hossein Hajkazemi, Peter Desnoyers, and Abutalib Aghayev. 2017. Modeling drive-managed SMR performance. *ACM Transactions on Storage* 13, 4 (2017), 22 pages. DOI: https://doi.org/10.1145/3139242

[189] Zhaoyan Shen, Yuhan Yang, Yungang Pan, Yuhao Zhang, Zhiping Jia, Xiaojun Cai, Bingzhe Li, and Zili Shao. 2023. A multiagent reinforcement learning-assisted cache cleaning scheme for DM-SMR. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 42, 8 (2023), 2500–2513. DOI: https://doi.org/10.1109/TCAD.2022.3222670

[190] Anton Shilov. 2016. Seagate Introduces First 8 TB Hard Disk Drive for Consumer NAS Applications. (2016).Retrieved October 31, 2024 from https://www.anandtech.com/show/9946/seagate-introduces-8-tb-hard-disk-drive-for-nas

[191] Scot Strong. 2014. HGST Announces World's First Enterprise-Class 10TB HDD. (2014). Retrieved October 31, 2024 from https://www.thessdreview.com/raid-enterprise/hgst-announces-worlds-first-enterprise-class-10tb-hdd/. Accessed: 2024-10-31.

[192] Diansen Sun and Yunpeng Chai. 2020. SAC: A co-design cache algorithm for emerging SMR-based high-density disks. In *Proceedings of the 25th International Conference on Architectural Support for Programming Languages and Operating Systems.* Association for Computing Machinery, New York, NY, USA, 1047–1061. DOI: https://doi.org/10.1145/3373376.3378474

[193] Diansen Sun, Ruixiong Tan, and Yunpeng Chai. 2023. A universal SMR-aware cache framework with deep optimization for DM-SMR and HM-SMR disks. *ACM Transactions on Storage* 19, 3, Article 26 (jun 2023), 35 pages. DOI: https://doi.org/10.1145/3588442

[194] Anand Suresh, Garth A. Gibson, and Gregory R. Ganger. 2012. Shingled magnetic recording for big data applications. https://api.semanticscholar.org/CorpusID:18715030 (Last accessed on 2024-01-01).

[195] I. Tagawa and M. Williams. 2009. High density data-storage using shingle-write. https://intermagconference.com/intermag2009/src/Program1.pdf

[196] Sophia Tan, Weiya Xi, Zhi Yong Ching, Chao Jin, and Chun Teck Lim. 2012. Simulation for a shingled magnetic recording disk. In *Proceedings of the 2012 Digest APMRC*. IEEE, USA, 1–7.

[197] Alexander Thomasian and Mario Blaum. 2009. Higher reliability redundant disk arrays: Organization, operation, and coding. *ACM Transactions on Storage* 5, 3 (2009), 59 pages. DOI: https://doi.org/10.1145/1629075.1629076

[198] A. Thomasian and Jai Menon. 1997. RAID5 performance with distributed sparing. *IEEE Transactions on Parallel and Distributed Systems* 8, 6 (1997), 640–657. DOI: https://doi.org/10.1109/71.595583

[199] D. A. Thompson and J. S. Best. 2000. The future of magnetic data storage techology. *IBM Journal of Research and Development* 44, 3 (2000), 311–322. DOI: https://doi.org/10.1147/rd.443.0311

[200] TIM FELDMAN AND GARTH GIBSON. 2013. Shingled Magnetic Recording Areal Density Increase Requires New Data Management. Retrieved January 01, 2024 from https://www.usenix.org/system/files/login/articles/05_feldman_022-030_final.pdf. (2013).

[201] Timothy Feldman. 2014. Host Aware SMR Timothy Feldman OpenZFS Dev Summit. Retrieved January 01, 2024 from https://openzfs.org/w/images/2/2a/Host-Aware_SMR-Tim_Feldman.pdf. (2014).

[202] Timothy Feldman. 2018. Flex Dynamic Recording. Retrieved January 01, 2024 from https://www.usenix.org/system/files/login/articles/login_spring18_09_feldman.pdf. (2018).

[203] Tom Coughlin. 2016. Shingled Magnetic Recording (SMR) Panel: Data Management Techniques Examined. Retrieved January 01, 2024 from https://www.snia.org/sites/default/files/DSI/2016/presentations/etc/TomCoughlin-Shingled_Magnetic_Recording_Panel.pdf. (2016).

[204] Tom Coughlin. April, 2024. Dual Layer Recording Could Enable 120+TB Hard Disk Drives. Retrieved April 08, 2024 from https://www.forbes.com/sites/tomcoughlin/2024/04/06/dual-layer-recording-could-enable-120tb-hard-disk-drives/. (April, 2024).

[205] P. Tozman, S. Isogami, I. Suzuki, A. Bolyachkin, H. Sepehri-Amin, S. J. Greaves, H. Suto, Y. Sasaki, T. Y. Chang, Y. Kubota, P. Steiner, P. W. Huang, K. Hono, and Y. K. Takahashi. 2024. Dual-layer FePt-C granular media for multi-level heatassisted magnetic recording. *Acta Materialia* 271 (1 June 2024), 119869. https://doi.org/10.1016/j.actamat.2024.119869

[206] Theodore Ts'o. 2018. Hybrid SMR Disk Drives: Who, What, Why, When, and Where? (2018). Retrieved October 31, 2024 from https://www.opencompute.org/files/OCP18-Hybrid-SMR-Product-Requirements.pdf

[207] Kalyana Sundaram Venkataraman, Tong Zhang, Wenzhe Zhao, Hongbin Sun, and Nanning Zheng. 2013. Scheduling algorithms for handling updates in shingled magnetic recording. In *Proceedings of the 2013 IEEE 8th International Conference on Networking, Architecture and Storage*. IEEE, USA, 205–214. DOI: https://doi.org/10.1109/NAS.2013.33

[208] Grant Wallace, Fred Douglis, Hangwei Qian, Philip Shilane, Stephen Smaldone, Mark Chamness, and Windsor Hsu. 2012. Characteristics of backup workloads in production systems. In *Proceedings of the 10th USENIX Conference on File and Storage Technologies*. USENIX Association, USA, 4.

[209] Chip Walter. 2005. *Kryder's Law. Scientific American* 293, 2 (August 2005), 32–33. https://doi.org/10.1038/scientificamerican0805-32

[210] Jiguang Wan, Nannan Zhao, Yifeng Zhu, Jibin Wang, Yu Mao, Peng Chen, and Changsheng Xie. 2012. High performance and high capacity hybrid shingled-recording disk system. In *Proceedings of the 2012 IEEE International Conference on Cluster Computing*. IEEE, New York, NY, USA, 173–181.

[211] Chunling Wang, Dandan Wang, Yupeng Chai, Chuanwen Wang, and Diansen Sun. 2017. Larger, cheaper, but faster: SSD-SMR hybrid storage boosted by a new SMR-oriented cache framework. In *Proceedings of the 33rd International Conference on Massive Storage Systems and Technology*. IEEE, USA, 1–16. Retrieved from https://api.semanticscholar.org/CorpusID:203576871

[212] Jibin Wang, Xiao ke Su, Yuchen Yang, Jiguang Wan, and Xiang Chen. 2012. An optimization model for shingled disk via hybrid mapping and multiple garbage collection strategies. In *Proceedings of the International Conference on Automatic Control and Artificial Intelligence*. IEEE, USA, 473–476. DOI: https://doi.org/10.1049/cp.2012.1019

[213] D. Weller and A. Moser. 1999. Thermal effect limits in ultrahigh-density magnetic recording. *IEEE Transactions on Magnetics* 35, 6 (1999), 4423–4439. DOI: https://doi.org/10.1109/20.809134

[214] Western Digital, Dynamic Hybrid SMR. 2017. Dynamic Hybrid SMR. Retrieved January 01, 2024 from https://blog.westerndigital.com/dynamic-hybrid-smr/. (2017).

[215] John Wilkes, Richard Golding, Carl Staelin, and Tim Sullivan. 1996. The HP AutoRAID hierarchical storage system. *ACM Transactions on Computer Systems* 14, 1 (1996), 108–136. DOI: https://doi.org/10.1145/225535.225539

[216] Alexander Q. Wu, Yukiko Kubota, Timothy Klemmer, Tim Rausch, Chubing Peng, Yingguo Peng, Darren Karns, Xiaobin Zhu, Yinfeng Ding, Eric K. C. Chang, Yongjun Zhao, Hua Zhou, Kaizhong Gao, Jan-Ulrich Thiele, Mike Seigler, Ganping Ju, and Edward Gage. 2013. HAMR areal density demonstration of 1+ tbpsi on spinstand. *IEEE Transactions on Magnetics* 49, 2 (2013), 779–782. DOI: https://doi.org/10.1109/TMAG.2012.2219513

[217] Chun-Feng Wu, Martin Kuo, Ming-Chang Yang, and Yuan-Hao Chang. 2022. Performance enhancement of SMR-based deduplication systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 41, 9 (2022), 2835–2848. DOI: https://doi.org/10.1109/TCAD.2021.3120072

[218] Chun-Feng Wu, Ming-Chang Yang, and Yuan-Hao Chang. 2018. Improving runtime performance of deduplication system with host-managed SMR storage drives. In *Proceedings of the 2018 55th ACM/ESDA/IEEE Design Automation Conference*. IEEE, USA, 1–6. DOI: https://doi.org/10.1109/DAC.2018.8465941

[219] Fenggang Wu. 2020. *Efficient Data and Space Management in Modern Storage Devices and Systems*. Ph.D. Dissertation. University of Minnesota. Retrieved from https://hdl.handle.net/11299/216400. Retrieved from the University Digital Conservancy.

[220] Fenggang Wu, Ziqi Fan, Ming-Chang Yang, Baoquan Zhang, Xiongzi Ge, and David H.C. Du. 2017. Performance evaluation of host aware shingled magnetic recording (HA-SMR) drives. *IEEE Transactions on Computers* 66, 11 (2017), 1932–1945. DOI: https://doi.org/10.1109/TC.2017.2713360

[221] Fenggang Wu, Bingzhe Li, Zhichao Cao, Baoquan Zhang, Ming-Hong Yang, Hao Wen, and David H. C. Du. 2019. ZoneAlloy: Elastic data and space management for hybrid SMR drives. In *Proceedings of the 11th USENIX Conference on Hot Topics in Storage and File Systems.* USENIX Association, USA, 2.

[222] Fenggang Wu, Bingzhe Li, and David H. C. Du. 2021. FluidSMR: Adaptive management for hybrid SMR drives. *ACM Transactions on Storage* 17, 4 (2021), 30 pages. DOI: https://doi.org/10.1145/3465404

[223] Fenggang Wu, Bingzhe Li, Baoquan Zhang, Zhichao Cao, Jim Diehl, Hao Wen, and David H.C. Du. 2021. TrackLace: Data management for interlaced magnetic recording. *IEEE Transactions on Computers* 70, 3 (2021), 347–358. DOI: https://doi.org/10.1109/TC.2020.2988257

[224] Fenggang Wu, Ming-Chang Yang, Ziqi Fan, Baoquan Zhang, Xiongzi Ge, and David H. C. Du. 2016. Evaluating host aware SMR drives. In *Proceedings of the 8th USENIX Conference on Hot Topics in Storage and File Systems.* USENIX Association, USA, 31–35.

[225] Fenggang Wu, Baoquan Zhang, Zhichao Cao, Hao Wen, Bingzhe Li, Jim Diehl, Guohua Wang, and David H. C. Du. 2018. Data management design for interlaced magnetic recording. In *Proceedings of the 10th USENIX Conference on Hot Topics in Storage and File Systems.* USENIX Association, USA, 14.

[226] Wen Xia, Hong Jiang, Dan Feng, Fred Douglis, Philip Shilane, Yu Hua, Min Fu, Yucheng Zhang, and Yukun Zhou. 2016. A comprehensive study of the past, present, and future of data deduplication. *Proc. IEEE* 104, 9 (2016), 1681–1710. DOI: https://doi.org/10.1109/JPROC.2016.2571298

[227] Wenjian Xiao, Huanqing Dong, Liuying Ma, Zhenjun Liu, and Qiang Zhang. 2016. HS-BAS: A hybrid storage system based on band awareness of shingled write disk. In *Proceedings of the 2016 IEEE 34th International Conference on Computer Design*. IEEE, USA, 64–71. DOI: https://doi.org/10.1109/ICCD.2016.7753262

[228] Xuchao Xie, Liquan Xiao, and David H.C. Du. 2019. ZoneTier: A zone-based storage tiering and caching co-design to integrate SSDs with SMR drives. *ACM Transactions on Storage* 15, 3 (2019), 25 pages. DOI: https://doi.org/10.1145/3335548

[229] Xuchao Xie, Liquan Xiao, Xiongzi Ge, and Qiong Li. 2018. SMRC: An endurable SSD cache for host-aware shingled magnetic recording drives. *IEEE Access* 6 (2018), 20916–20928. DOI: https://doi.org/10.1109/ACCESS.2018.2825109

[230] Xuchao Xie, Tianye Yang, Qiong Li, Dengping Wei, and Liquan Xiao. 2018. Duchy: Achieving both SSD durability and controllable SMR cleaning overhead in hybrid storage systems. In *Proceedings of the 47th International Conference on Parallel Processing.* Association for Computing Machinery, New York, NY, USA, Article 81, 9 pages. DOI: https://doi.org/10.1145/3225058.3225067

[231] Jun Xu. 2018. *Block Trace Analysis and Storage System Optimization A Practical Approach with MATLAB/Python Tools*. Apress Berkeley, CA, 233 Spring Street, 6th Floor, New York, NY 10013, 271.

[232] Peng Xu, Jiguang Wan, Ping Huang, Bihua Shu, Chenlei Tang, and Changsheng Xie. 2019. An active method to mitigate the long latencies for host-aware shingle magnetic recording drives. In *Proceedings of the 2019 IEEE 25th International Conference on Parallel and Distributed Systems.* IEEE, USA, 17–26. DOI: https://doi.org/10.1109/ICPADS47876.2019.00012

[233] Ming-Chang Yang, Yuan-Hao Chang, Fenggang Wu, Tei-Wei Kuo, and David H.C. Du. 2017. Virtual persistent cache: Remedy the long latency behavior of host-aware shingled magnetic recording drives. In *Proceedings of the 2017 IEEE/ACM International Conference on Computer-Aided Design.* IEEE, Irvine California, USA, 17–24. DOI: https://doi.org/10.1109/ICCAD.2017.8203755

[234] Ming-Chang Yang, Yuan-Hao Chang, Fenggang Wu, Tei-Wei Kuo, and David H. C. Du. 2019. On improving the write responsiveness for host-aware SMR drives. *IEEE Transactions on Computers* 68, 1 (2019), 111–124. DOI: https://doi.org/10.1109/TC.2018.2845383

[235] Tianming Yang, Haitao Wu, Ping Huang, and Fei Zhang. 2017. A shingle-aware persistent cache management scheme for DM-SMR disks. In *Proceedings of the 2017 IEEE International Conference on Computer Design* IEEE, Boston Marriott Newton Boston Area, Massachusetts, USA, 81–88. DOI: https://doi.org/10.1109/ICCD.2017.21

[236] Ting Yao, Ji guang Wan, Ping Huang, Xubin He, Qingxin Gui, Fei Wu, and Changsheng Xie. 2017. A light-weight compaction tree to reduce I / O amplification toward efficient key-value stores. In *Proceedings of the 2017 IEEE 33rd International Conference on Massive Storage Systems and Technology.* IEEE, USA, 1–13. Retrieved from https://api.semanticscholar.org/CorpusID:13577976

[237] Ting Yao, Zhihu Tan, Jiguang Wan, Ping Huang, Yiwen Zhang, Changsheng Xie, and Xubin He. 2018. A set-aware key-value store on shingled magnetic recording drives with dynamic band. In *Proceedings of the 2018 IEEE International Parallel and Distributed Processing Symposium.* IEEE, Vancouver, British Columbia CANADA, 306–315. DOI: https://doi.org/10.1109/IPDPS.2018.00040

[238] Ting Yao, Zhihu Tan, Jiguang Wan, Ping Huang, Yiwen Zhang, Changsheng Xie, and Xubin He. 2019. SEALDB: An efficient LSM-tree based KV store on smr drives with sets and dynamic bands. *IEEE Transactions on Parallel and Distributed Systems* 30, 11 (2019), 2595–2607. DOI: https://doi.org/10.1109/TPDS.2019.2918219

[239] Ting Yao, Jiguang Wan, Ping Huang, Yiwen Zhang, Changsheng Xie, and Xubin He. 2023. GearDB: A GC-free key-value store on HM-SMR drives with gear compaction. In *Proceedings of the ACM Turing Award Celebration Conference - China 2023.* Association for Computing Machinery, New York, NY, USA, 51–52. DOI: https://doi.org/10.1145/3603165.3607392

[240] Lingfang Zeng, Zehao Zhang, Yang Wang, Dan Feng, and Kenneth B. Kent. 2017. CosaFS: A cooperative shingle-aware file system. *ACM Transactions on Storage* 13, 4 (2017), 23 pages. DOI: https://doi.org/10.1145/3149482

[241] Zhimin Zeng, Xinyu Chen, Laurence T. Yang, and Jinhua Cui. 2022. IMRSim: A disk simulator for interlaced magnetic recording technology. In *Proceedings of the Network and Parallel Computing*, Shaoshan Liu and Xiaohui Wei (Eds.). Springer Nature Switzerland, Cham, 267–273.

[242] Baoquan Zhang, Ming-Hong Yang, Xuchao Xie, and David H.C. Du. 2020. Idler : I/O workload controlling for better responsiveness on host-aware shingled magnetic recording drives. *IEEE Transactions on Computers* 69, 6 (2020), 777–788. DOI: https://doi.org/10.1109/TC.2020.2966194

[243] Chi Zhang, Song Liu, Fangxing Yu, Menghan Li, Wei Tang, Fei Liu, and Weiguo Wu. 2023. Balloon: An elastic data management strategy for interlaced magnetic recording. *Applied Sciences* 13, 17 (2023), 17. DOI: https://doi.org/10.3390/app13179767

[244] Chi Zhang, Fangxing Yu, Shiqiang Nie, Wei Tang, Fei Liu, Song Liu, and Weiguo Wu. 2024. Amphisbaena: A novel persistent buffer management strategy to improve SMR disk performance. *Applied Sciences* 14, 2 (2024), 630. https://doi.org/10.3390/app14020630

[245] Yiwen Zhang, Ting Yao, Jiguang Wan, and Changsheng Xie. 2022. Building GC-free key-value store on HM-SMR drives with ZoneFS. *ACM Transactions on Storage* 18, 3 (2022), 23 pages. DOI: https://doi.org/10.1145/3502846

[246] Xuda Zheng, Chi Zhang, Keqiang Duan, Weiguo Wu, and Jie Yan. 2021. SLA: A cache algorithm for SSD-SMR storage system with minimum RMWs. In *Algorithms and Architectures for Parallel Processing: 21st International Conference, ICA3PP 2021, Virtual Event, December 3–5, 2021, Proceedings, Part III.* Springer-Verlag, Berlin, 587–601. DOI: https://doi.org/10.1007/978-3-030-95391-1_37

[247] Su Zhou, Erci Xu, Hao Wu, Yu Du, Jiacheng Cui, Wanyu Fu, Chang Liu, Yingni Wang, Wenbo Wang, Shouqu Sun, Xianfei Wang, Bo Feng, Biyun Zhu, Xin Tong, Weikang Kong, Linyan Liu, Zhongjie Wu, Jinbo Wu, Qingchao Luo, and Jiesheng Wu. 2023. SMRSTORE: A storage engine for cloud object storage on HM-SMR drives. In *Proceedings of the 21st USENIX Conference on File and Storage Technologies.* USENIX Association, Santa Clara, CA, 395–408. Retrieved from https://www.usenix.org/conference/fast23/presentation/zhou

[248] Jian-Gang (Jimmy) Zhu. 2003. New heights for hard disk drives. *Materials Today* 6, 7 (2003), 22–31. https://doi.org/10.1016/S1369-7021(03)00729-6

[249] Jian-Gang Zhu, Xiaochun Zhu, and Yuhui Tang. 2008. Microwave assisted magnetic recording. *IEEE Transactions on Magnetics* 44, 1 (2008), 125–131. DOI: https://doi.org/10.1109/TMAG.2007.911031

[250] IBM Zurich. 2024. IBM Cloud storage: Reliable and secure data availability. https://www.zurich.ibm.com/cloudstorage/ (Last accessed: 2024-10-20).

Received 15 November 2023; revised 7 November 2024; accepted 2 April 2025