



RAIDShield: Characterizing, Monitoring, and Proactively Protecting Against Disk Failures

Ao Ma, Fred Douglass, Guanlin Lu, and Darren Sawyer, *EMC Corporation*;
Surendar Chandra and Windsor Hsu, *Datrium, Inc.*

<https://www.usenix.org/conference/fast15/technical-sessions/presentation/ma>

**This paper is included in the Proceedings of the
13th USENIX Conference on
File and Storage Technologies (FAST '15).**

February 16–19, 2015 • Santa Clara, CA, USA

ISBN 978-1-931971-201

**Open access to the Proceedings of the
13th USENIX Conference on
File and Storage Technologies
is sponsored by USENIX**

RAIDShield: Characterizing, Monitoring, and Proactively Protecting Against Disk Failures

Ao Ma¹, Fred Douglis¹, Guanlin Lu¹, Darren Sawyer¹, Surendar Chandra², Windsor Hsu²
¹EMC Corporation, ²Datrium, Inc.

Abstract

Modern storage systems orchestrate a group of disks to achieve their performance and reliability goals. Even though such systems are designed to withstand the failure of individual disks, failure of multiple disks poses a unique set of challenges. We empirically investigate disk failure data from a large number of production systems, specifically focusing on the impact of disk failures on RAID storage systems. Our data covers about one million SATA disks from 6 disk models for periods up to 5 years. We show how observed disk failures weaken the protection provided by RAID. The count of *reallocated sectors* correlates strongly with impending failures.

With these findings we designed RAIDSHIELD, which consists of two components. First, we have built and evaluated an active defense mechanism that monitors the health of each disk and replaces those that are predicted to fail imminently. This proactive protection has been incorporated into our product and is observed to eliminate 88% of triple disk errors, which are 80% of all RAID failures. Second, we have designed and simulated a method of using the joint failure probability to quantify and predict how likely a RAID group is to face multiple simultaneous disk failures, which can identify disks that collectively represent a risk of failure even when no individual disk is flagged in isolation. We find in simulation that RAID-level analysis can effectively identify most vulnerable RAID-6 systems, improving the coverage to 98% of triple errors.

1 Introduction

Storage systems have relied for decades on redundancy mechanisms such as RAID to tolerate disk failures, assuming an ideal world with independent and instantaneous failures as well as exponential distributions of the time to failure [3, 11, 18, 36]. However, some assumptions no longer hold given the fault model presented by modern disk drives. Schroeder and Gibson [42] analyzed 100,000 disks and rejected the hypothesis of the time between disk replacements following an exponential distribution. Further, in addition to whole-disk failures that

make an entire disk unusable, modern drives can exhibit latent sector errors in which a block or set of blocks become inaccessible [6, 29]. Such sector faults in otherwise working disks further weaken the RAID reconstruction capability. Not only were sector errors previously ignored in the early RAID reliability model, these errors may worsen over time due to increasing drive complexity [4] and the common use of less reliable disks in storage systems [6, 17, 38]. In short, RAID protection is no longer enough; however, given its prevalence in the storage industry, a mechanism to shield RAID systems from unreliable disks would have a wide audience.

System designers have realized the new threats caused by these disk faults and built additional mechanisms to improve data reliability. While the original RAID system would protect against the loss of data from one disk (either an unavailable sector or the failure of the entire disk), the trend has been to use additional redundancy to guard against related data loss on multiple disks. For example, some storage arrays incorporate extra levels of parity, such as RAID-6, which can tolerate two simultaneous whole or partial disk failures [2, 12, 13, 19, 22, 23]; others add redundancy with CPU-intensive erasure coding [14, 25]. Throughout this paper we focus on “triple-disk failures,” or “triple failures” for short, which refer to any combination of losing related data from three disks simultaneously, due to bad sectors or an entire disk. If a RAID-6 system encounters a triple failure it will lose data, but additional layers of redundancy (such as replication) can further protect against catastrophic data loss.

Many storage systems apply disk scrubbing to proactively detect latent sector errors; i.e., they read data from disk specifically to check for media errors, rather than because an application has requested the data [28, 43]. File systems also incorporate techniques such as replication and parity to improve data availability [10, 37, 41]; replication is critical because the failure of a disk group (DG) can be rectified, at high overhead, with a separate replica accessible via a LAN or WAN. Finally, even when primary storage systems are backed up onto separate dedicated backup systems, those backup systems can them-

selves be replicated [27].

Unfortunately, improvements to the basic RAID architecture are still based on certain assumptions given the limited understanding of disk fault modes. For example, empirical observations show both the sector error rate and the whole-disk failure rate grow over time [6, 42], causing RAID availability to continuously degrade. It is possible for multiple disks in the same RAID DG to fail simultaneously while other working disks have developed a number of latent sector errors [16]. Such multiple combined faults can overcome RAID protection and affect data availability. Unfortunately, little data is publicly available that quantifies such correlated faults.

To address this knowledge gap with respect to storage system reliability, we collected and analyzed disk error logs from EMC Data Domain backup systems. The data cover periods up to 60 months and include about 1 million SATA disks from deployed systems at customer and internal sites. To our knowledge, this is the first study of this magnitude to focus on analyzing disk faults (e.g., whole-disk failures and sector errors) that influence data reliability. The logs report when a disk failure is detected, at which point a system can automatically initiate data recovery onto a spare drive using available data from within that system. They also report larger-scale outages, when too many drives fail simultaneously for data to be accessible. We define a *recovery-related incident* as a failure that requires the retrieval of data from another system, such as a backup or disk replica.

Our analysis reveals that many disks fail at a similar age and the frequency of sector errors keeps increasing on working disks. Ensuring data reliability in the worst case requires adding considerable extra redundancy, making the traditional passive approach of RAID protection unattractive from a cost perspective. By studying numerous types of disk error, we also observe that the accumulation of sector errors contributes to whole-disk failures, causing disk reliability to deteriorate continuously. Specifically, a large number of *reallocated sectors* (RS¹) indicates a high probability of imminent whole-disk failure or, at a minimum, a burst of sector errors.

With these findings we designed RAIDSHIELD, a monitoring mechanism, which proactively identifies and preempts impending failures and vulnerable RAID groups. RAIDSHIELD consists of two components, PLATE+ARMOR. First, we have built and evaluated *Predict Loss Accumulating in The Enterprise* (PLATE), an active defense mechanism that monitors the health of each disk by tracking the number of reallocated sectors, proactively detecting unstable disks and replacing them in advance. PLATE has been deployed in production systems for nearly a year. Second, we have de-

¹RS is also sometimes referred to as RAS in disk statistics, but we prefer to avoid the confusion with other uses of RAS in the CS literature.

signed and simulated *Assure Redundant Media Or Replace* (ARMOR), which uses the joint failure probability of a DG to quantify the likelihood of multiple simultaneous disk failures. ARMOR has the potential to identify sets of disks that collectively represent a risk of failure even when no individual disk is flagged in isolation. Given this assessment, unstable disks can then be replaced in advance or the redundancy of a DG can be increased; either approach can improve overall RAID availability.

Simulation results for PLATE, the single-disk proactive protection, show it can capture up to 65% of impending whole-disk failures with up to 2.5% false alarms. After incorporating it into our product, we find its effect on RAID failures is disproportionate: it has been observed to eliminate 70% of the recovery-related incidents caused by RAID failures and 88% of the RAID failures due to triple disk failures. Its benefits are somewhat limited by the types of errors that it cannot predict: about 20% of DG failures are caused by user errors, hardware faults, and other unknown reasons. Simulation results indicate that ARMOR, the cross-disk proactive protection, can effectively identify 80% of vulnerable RAID-6 systems in a test of 5500 DGs. We find that it can predict most of the triple failures not prevented by PLATE, leading to total coverage of 98% of triple failures.

The rest of this paper is organized as follows. We first provide background on partial disk failures and describe our storage system architecture, including an overview of RAIDSHIELD (§2). §3 presents our study on the relation between whole-disk failure and sector errors, and it characterizes reallocated sectors, which are found to be highly correlated with whole-disk failures. §4 describes and evaluates PLATE, demonstrating the substantial reduction in RAID failures after deploying single-disk predictive replacement. §5 describes the design and evaluation, via simulation, of ARMOR: using joint probabilities to assess the failure risk to a DG as a whole. §6 discusses related work and §7 concludes.

2 Background and Motivation

In this section we define disk partial failures, providing the background to understand our subsequent failure analysis. We then present an overview of our storage system architecture and describe the two aspects of RAIDSHIELD.

2.1 Disk Failures

Disks do not fail in a simple fail-stop fashion. Hence, there is no consensus definition of what constitutes a disk failure [5, 8, 45]. The production systems we studied define a whole-disk failure as:

- The system loses its connection to the disk,
- An operation exceeds the timeout threshold, or
- A write operation fails.

These criteria serve as the bottom line to replace disks that cannot function properly. However, in addition to whole-disk failures, disk drives can experience various partial failures while they still otherwise function. Sector-related issues are the major partial failures that endanger data safety [7, 31, 41]. Disk drives therefore provide a variety of proprietary and complicated mechanisms to rectify some failures and extend drive lifespans. In this subsection, we briefly describe disk technology, focusing on detection and error handling mechanisms for sector errors; refer elsewhere for more detailed descriptions [6, 38]. Failure detection and recovery mechanisms vary by manufacturer, production model, interface and capacity; the mechanisms introduced here cover common SATA disk internal mechanisms.

Sector errors can be categorized into different specific types based on how they are detected, as shown in Figure 1. Operations to the disk can be initiated by file system `read()` and `write()` calls as well as by an internal *scan* process, which systematically checks sector reliability and accessibility in the background. (These are shown in Figure 1 in blue, magenta, and green respectively.)

Media error: This error occurs when a particular disk sector cannot be read, whether during a normal read or a background disk scan. Any data previously stored in the sector is lost. The disk interface reports the status code upon detecting a sector error, specifying the reason why the read command failed.

Pending and Uncorrectable sector: Unstable sectors detected in the background process will be marked as pending sectors, and disk drives can try rectifying these errors through internal protection mechanisms, such as built-in Error Correcting Codes and Refreshment. These techniques rewrite the sector with the data read from that track to recover the faded data. Any sectors that are not successfully recovered will be marked as uncorrectable sectors.

Reallocated sector: After a number of unsuccessful retries, disk drives automatically re-map a failed write to a spare sector; its logical block address (LBA) remains unchanged. Modern disk drives usually reserve a few thousand spare sectors, which are not initially mapped to particular LBAs. Reallocation only occurs on detected write errors.

We also observe that changes to disk technology tend to increase the frequency of sector errors, a major fraction of partial disk failures. First, the number of sectors on a disk keeps increasing: while the capacity of individual disks may not be increasing at the rate once predicted by Kryder [33, 47], they still increase. Thus, if sector errors occur at the current rate, there would be more sector

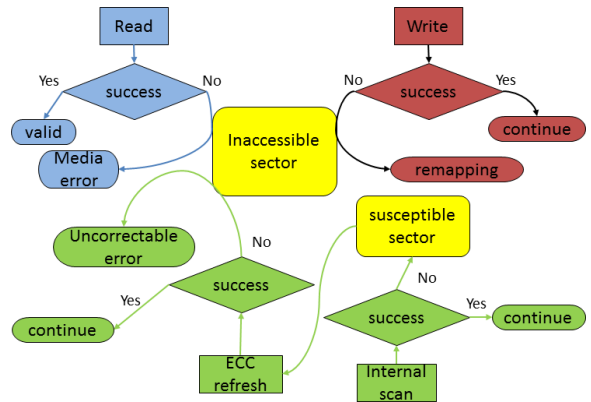


Figure 1: Sector error transition. This figure depicts different responses to sector errors. A read (shown in blue) will report a media error if target sector is unreadable. A write (magenta) will attempt to remap a bad sector. An internal scan (green) will try to identify and rectify unstable sectors.

errors per disk. Second, the disk capacity increase comes from packing more sectors per track, rather than adding more physical platters. Sectors become increasingly vulnerable to media scratches and side-track erasures [15].

2.2 Storage System Environment

We now briefly describe the context of our storage system with a focus on sector error detection and handling. At a high level, the storage system is composed of three layers, including a typical file system, the RAID layer, and the storage layer. The file system processes client requests by sending read and write operations to the RAID layer. The RAID layer transforms the file system requests into disk logical block requests and passes them to the storage layer, which accesses the physical disks. Our RAID layer adopts the RAID-6 algorithm, which can tolerate two simultaneous failures.

In addition to reporting latent sector errors captured in ordinary I/Os, our storage systems scrub all disks periodically as a proactive measure to detect latent sector errors and data corruption errors. Specifically, this scan process checks the accessibility of “live” sectors (those storing data accessible through the file system), verifies the checksums, and notifies the RAID layer on failures.

Sector error handling depends on the type of disk request. A failed write is re-directed to a spare sector through the automatic disk remapping process, without reporting the error to the storage layer. If a read fails, the RAID layer reconstructs data on the inaccessible sector and passes it to the storage layer for rewriting. Writing to the failed sector will trigger the disk internal mapping process. Note that given the process of RAID reconstruction and re-issued write, the failed sector detected

through read (media error) will eventually lead to an RS. Therefore, the RS count is actually the number of inaccessible sectors detected in either reads or writes.

Finally, the systems evaluated in this paper are backup systems, which are known to have write-heavy workloads with fewer random I/Os than primary storage [46]; this workload may change the way in which disk faults are detected, as write errors may be relatively more common than read errors. The general conclusions should hold for other types of use.

2.3 RAIDSHIELD Motivation

Despite the expectation that RAID-6 systems should be resilient to disk failures, given a large enough population of DGs there will be errors leading to potential data loss [3]. Indeed, our systems encounter RAID-level errors, but thankfully these are extremely rare.² These systems usually rely on extra layers of redundancy such as (possibly off-site) replication to guard against catastrophic failures, but there is a strong incentive to decrease the rate at which RAID failures occur.

As we see in §3, disks that are installed together are somewhat likely to fail together, and disks that have partial (media) errors will rapidly accumulate errors until they are deemed to have failed completely. Our goal for RAIDSHIELD is to identify and replace failing disks before they completely fail, within reason. In the extreme case, one could use a single disk error as a warning signal and replace any disk as soon as it reported the slightest problem. However, the cost in time and expense would be prohibitive, especially for large-scale installations like cloud providers. With RAIDSHIELD, we take two tacks in this regard. The first is to use statistical information to discriminate between those disks that are likely to fail soon and those that are not. In the next section we consider a number of disk statistics that might be used for this purpose, finding that the *reallocated sectors* (RS) metric is an excellent predictor of impending failures. We show in §4 that after deploying PLATE proactive disk replacement, looking at each disk in isolation, our RAID failures dropped dramatically.

Can we do better with ARMOR, our second tack? We hypothesize that by using the joint failure probability across a DG we can find some additional instances where no single disk is close enough to failure to justify replacing it using the criteria for PLATE, but enough disks are symptomatic that the DG as a whole is in jeopardy. In §5 we present the probability analysis and some simulation results to justify this approach. In addition, we speculate that in some environments, it will be undesirable to

²We are unable to release specific error rates for DGs or disk models.

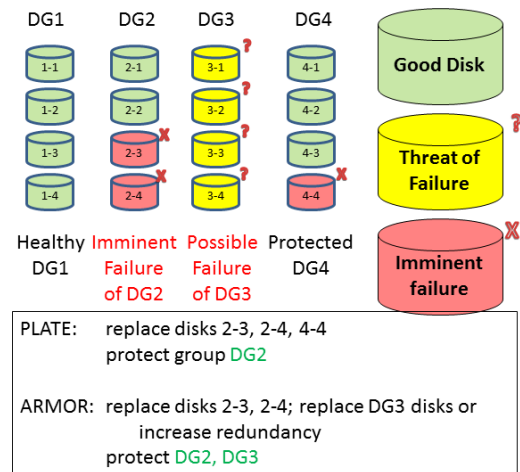


Figure 2: **Example of RAIDSHIELD.** Four DGs are shown, each with four disks. Green disks are healthy, yellow disks are at risk, and red disks are likely to fail imminently. DG2 and DG3 are at risk of failure.

proactively replace every disk that is showing the possibility of failure; instead, it may be important to prioritize among DGs and first replace disks in the most vulnerable groups. A single soon-to-fail disk in an otherwise healthy DG is a lower risk than a DG with many disks that have moderate probability of failure.

Figure 2 provides an example of the difference between PLATE and ARMOR. There are four disk groups; DG2, with two failing disks, is at high risk, while DG3 has a moderate risk due to the large number of partly-failing disks. With PLATE, we would replace the red disks, protecting vulnerable DG2 and improving the protection of DG4, but DG4 is already protected by three healthy disks. With ARMOR, we replace the two failing disks in DG2 but also recognize the vulnerability of DG3 given the large number of at-risk disks.

3 Disk Failure Analysis

Understanding the nature of whole-disk failures and partial failures is essential for improving storage system reliability and availability. This section presents the results of our analysis of about 1 million SATA disks. First, we describe how we collected the disk data studied in this work. Second, we present our observations of the new disk failure modes (e.g., simultaneous disk failures and sector errors) which endanger RAID availability. Third, we analyze the correlation between these two failure modes. Finally, we analyze characteristics and properties of reallocated sectors, the specific sector error type that is found to predict drive failures.

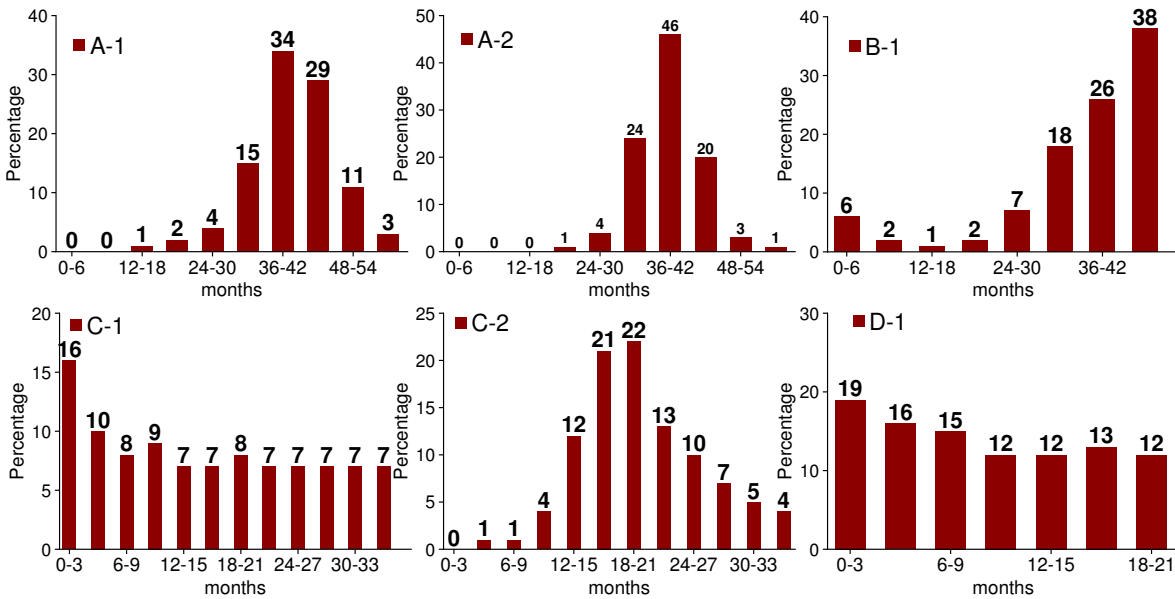


Figure 3: **Distribution of lifetimes of failed drives.** These graphs show that many disks fail at a similar age. Note that the number of buckets, i.e. total age since deployment, and time length of each bucket varies by drive.

Disk Model	Population (Thousands)	First Deployment	Log Length (Months)
A-1	34	06/2008	60
A-2	165	11/2008	60
B-1	100	06/2008	48
C-1	93	10/2010	36
C-2	253	12/2010	36
D-1	384	09/2011	21

Table 1: **Disk population.** Population, earliest deployment date and log length of disk models used in this study.

3.1 Data Collection

Our storage system has a built-in mechanism to log system status, which can optionally send important events back to a central repository each day [46]. These messages record a variety of system events including disk errors and failures. The data studied here are collected from these reports over a period of 5 years starting in June, 2008.

Similar to previous work [6], we anonymize disk information to make it possible to compare across disks from a single manufacturer but not across disk families. We denote each disk drive model as $\langle \text{family-capacity} \rangle$. Family is a single letter representing the disk family and capacity is a single number representing the disk's particular capacity. Although capacities are anonymized as a single number, relative sizes within a family are ordered by the number representing the capacity. That is, A-2 and C-2 are larger than A-1 and C-1 respectively.

Our entire sample of 1 million disks includes 6 disk models, each of which has a population of at least 30,000. They have been shipped in our storage systems since June, 2008, giving us a sufficient observation window to study various errors over the full lifespans of many drives. Details of the drives studied are presented in Table 1. Note that the recorded period of each disk model varies: the studied data range from 60-month logs of A-1 and A-2 down to 21 months for D-1.

3.2 New Disk Failure Modes

We observe two new disk failure modes that are not predicted by the early RAID reliability model and degrade RAID reliability and availability.

Drives fail at similar ages: We analyze all failed drives and categorize them into different buckets based on their lifetime. Figure 3 shows that a large fraction of failed drives are found at a similar age. For example, 63% of A-1 failed drives, 66% of A-2 failed drives and 64% of B-1 failed drives are found in their fourth year. This failure peak is also observed in the second year of the C-2 model, with 68% of failed drives found in this period. Given a large population of drives, some drives will fail not only in the same month but occasionally the same week or day, resulting in vulnerable systems. If a third error (a defective sector or a failed drive) should also occur before drives can be replaced and data reconstructed, the DG will be unavailable.

The lifetime distributions of C-1 and D-1 failed drives are comparatively uniform. However, these drives are

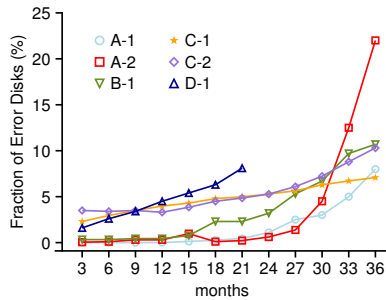


Figure 4: **Percentage of disks developing sector errors.** As disks age, the number with at least one error increases, and the rate of increase is higher the older the disk is. Note that D-1 has only a 21-month record.

relatively young compared to the drives with long observation intervals, so it is difficult to draw specific conclusions from this uniformity. We note a degree of “infant mortality” with these drives, with peaks of failures in the first three months.

Sector errors exacerbate risk: Figure 4 presents the fraction of disks affected by sector errors as a function of the disk age. Disks from all models show sector errors by the time they have been in use for 2–3 years, but some have significant errors much earlier. In addition, the rate at which errors appear increases with the age of the disks: for example, about 5% of A-2 disks get sector errors in the first 30 months, but it only takes an additional 6 months for 10% more to develop sector errors. Similar trends can be observed with A-1, B-1, and C-2.

To demonstrate the rate of error increase, we select 1000 disks randomly from each disk model, which developed at least one sector in a one-month observation window. We collect the count of their sector errors one year later. Figure 5 shows the average number of sector errors in the first and second years. For all drives with at least one sector error, the number of sector errors for the second year increases considerably, ranging from 25% for the C-2 model to about 300% for A-2.

These new disk failure modes reveal that the traditional RAID mechanism has become inadequate. The observation that many disks fail at a similar age means RAID systems face a higher risk of multiple whole-disk failures than anticipated. The increasing frequency of sector errors in working disks means RAID systems face a correspondingly higher risk of reconstruction failures: a disk that has not completely failed may be unable to provide specific sectors needed for the reconstruction. The disk technology trends introduced in §2.1 exacerbate these risks.

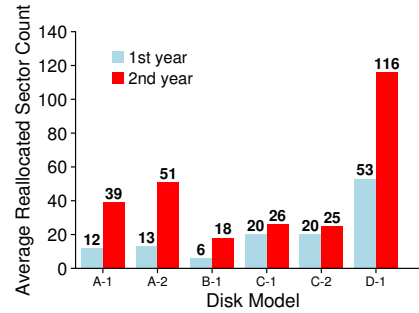


Figure 5: **Error counts year over year.** Among disks with sector errors, for each model the number of errors increased significantly in the second year.

3.3 Correlating Full and Partial Errors

Since both whole-disk failures and sector errors affect data availability, exploring how they are correlated helps us to understand the challenges of RAID reliability. Here we introduce the statistical methodology used to analyze the data, then we evaluate the correlation between whole-disk failures and sector errors.

3.3.1 Statistical Methods

Our objective is to compare the sector errors in working disks and failed ones, and to use a measure to reflect their discrimination. We use quantile distributions to quantitatively evaluate the correlation degree between disk failures and sector errors. Specifically, we collect the number of sector errors on working and failed disks, summarizing each data set value using *deciles* of the cumulative distribution (i.e., we divide the sorted data set into ten equal-sized subsets; we normally display only the first nine deciles to avoid the skew of outliers). Such quantiles are more robust than other statistical techniques, such as mean and cumulative distribution function, to outliers and noise in depicting the value distribution and have been used to analyze performance crises in data centers [9].

3.3.2 Identifying Correlation

As introduced in §2.1, sector errors can be categorized into specific types based on how they are detected. For example, a sector error detected in a read is regarded as a media error while a sector error captured in a write is counted as an RS. Those error counts can be collected through the disk SMART interface [1] and are included in our logs.

Figures 6-7 compare the deciles of disk errors built on

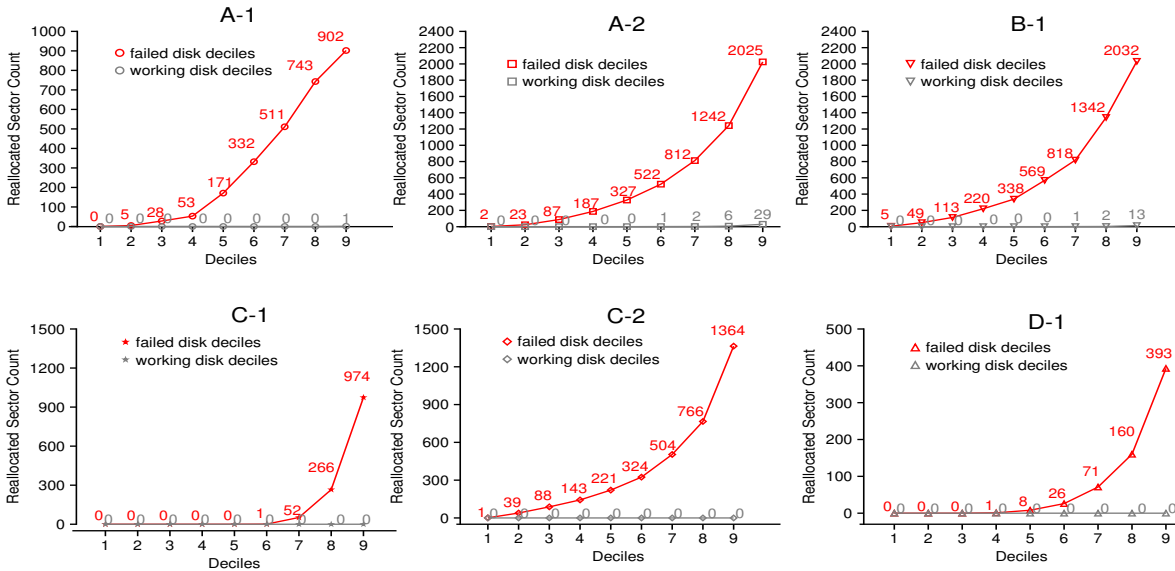


Figure 6: **Reallocated sector comparison.** Failed drives have more RS across all disk models. Many disks fail before they exhaust their spare sectors. Failed drives with bigger capacity have more RS. Y-axis scales vary.

the working and failed disk sets. The x-axis represents the Kth deciles, with the error counts on the y-axis.

Reallocated sector: Figure 6 presents the number of RS on failed and working drives. We observe that the majority of failed drives developed a large number of RS while most that are working have only a few. For example, 80% of A-2 failed drives have more than 23 RS but 90% of working drives have less than 29 of this error. Every disk model demonstrates a similar pattern; the only difference is how large the discrimination is. Failed disks have different RS counts, implying that many disks fail before they use up all spare sectors. We also find that failed drives with bigger capacity tend to have more RS, though the numbers depend more on the maximum number of reallocations permitted than the total size. For example, the median count of RS on A-2 failed drives is 327, compared to 171 for A-1; A-2 has both twice the capacity and twice the maximum number of reallocations, so this difference is expected. On the other hand, C-2 has twice the capacity as C-1 but the same maximum number of RS (2048), and its 9th decile of RS is only 40% higher than C-1. (Note that the median RS count for C-1 is zero, implying that many C-1 disks fail for reasons other than reallocated sectors; this is consistent with the large infant mortality shown in Figure 4 and bears further investigation. D-1 has similar characteristics.)

Media error: Due to the limitation of the logging messages we have on hand, we can analyze this error type only on the A-2 disk model. The result is presented in Figure 7. Though failed disks have more media errors than working ones, the discrimination is not that signif-

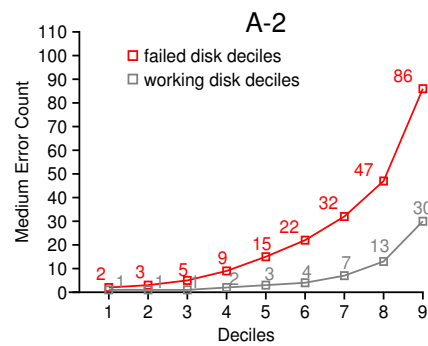


Figure 7: **Media error comparison.** There is only moderate discrimination. Shown only for A-2.

icant compared to RS. For example, 50% of failed disks have fewer than 15 media errors, and 50% of working ones developed more than 3 errors. There is a large overlap between them, perhaps because only sector errors detected in read operations are reported as media errors. Sector errors detected in writes will trigger the reallocation process directly without notifying the upper layer. Since the RAID layer will re-write the reconstructed data upon a detected media error, which causes the reallocation process, every media error will lead to an RS eventually: the media error count is thus a subset of RS. More details can be found in §2.2.

Pending and Uncorrectable sectors: As introduced in §2.1, sector errors discovered through the disk internal scan will be marked as pending sectors or uncorrectable sectors. The results for pending sectors are presented

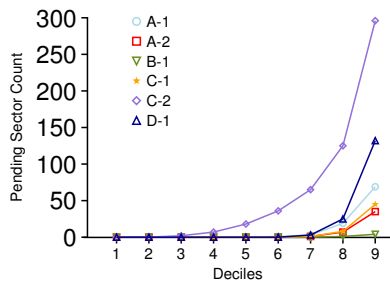


Figure 8: **Pending sector comparison.** *There is a large variation among different models.*

in Figure 8; the figure for uncorrectable sectors is similar and is omitted for space considerations. Through the comparison we find that for some disk models (such as A-1, A-2, and B-1), a certain fraction of failed disks (usually 30%) develop a similar amount of pending and uncorrectable sectors. Failed drives of the other disk models, including C-1, C-2, D-1 develop pending sector errors but none of them have uncorrectable sector errors, implying most pending errors have been addressed with drives' internal protection mechanisms. No working disks show these two types of sector errors, revealing that once disks develop these two types of error, they are very probable to fail.

3.3.3 Summary

These experiments characterize the correlation between whole-disk failures and various sector-related errors. We observe that most failed disks tend to have a larger number of RS than do working disks. Thus RS are strongly correlated with whole-disk failures. We infer that reallocation is the last resort to tolerate a defective sector after all other recovery mechanisms have failed; therefore, it avoids the influence of temporary errors which also appear on working disks. Further, given the process of RAID reconstruction and re-issued writes, inaccessible sectors detected through read and write will both eventually lead to RS. Therefore, the number of RS represents all inaccessible sectors and is a good indication of the extent to which a disk is wearing out.

3.4 Characterization of RS

The previous subsection revealed that RS appear more frequently in a large population of failed disks than working disks. Thus the number of RS is highly correlated with whole-disk failures across all disk models studied. This subsection studies characteristics of RS.

The best data set to study the properties of RS over the disk lifetime is disk model A-2. The reason is that this disk model was deployed for a long enough time

period (more than 5 years) with a robust population of failed drives and detailed logging. Therefore, we use disk model A-2 as an illustration to explain our findings in the following sections.

All disks fail eventually, so we define an *impending disk failure* in our study as the disk failing within a 60-day observation window. A two-month window gives enough time for the disk to expose latent problems, since disk failure is not a simple fail-stop process. If a disk does not fail during this observation period, it is regarded as a qualified working disk.

We first evaluate how RS counts relate to disk failure rates. We analyze the percentage of disk failures after they exceed different thresholds of RS. The results are presented in Figure 9. The X-axis represents the RS count and the Y-axis depicts the failure percentage.

As found by Pinheiro, et al., the failure rate jumps dramatically once the disk starts to develop RS [38]. This rate grows steadily as the count of RS increases; for example, the failure rate of disks without any RS is merely 1.7%, while more than 50% of disks fail after this count exceeds 40. If the count grows to the range of 500 and 600, the failure rate increases to nearly 95%. We conclude that the more RS the disk has, the higher probability the disk will fail.

Second, we study the failed drives by analyzing the period between the time the disk RS count exceeds a certain value and the time a disk failure happens. We collect all time-to-fail (TTF) values and summarize the data set with the box-and-whisker plot in Figure 10, showing the 10-25-50-75-90 percentiles. All values for the time margin shrink as the number of RS grows. For example, one of every two failed disks would have more than seven days TTF when it exceeds 40 RS. But when the count of RS grows beyond 200, 50% of those disks that will soon fail are found to fail within just two days. However, the prediction is not guaranteed: the 90th percentile of failures is measured in weeks rather than days. We conclude that a larger number of RS indicates a disk will fail more quickly, in most cases just a few days.

Third, we analyze working drives, which have developed a certain number of RS, and categorize them into different buckets based on their RS counts. Figure 11 groups disks into buckets, randomly selecting 1000 disks with 0-100 RS, 1000 disks with 101-200 reallocations, and so on. We track how many sector errors they have accumulated 30 days later: for each bucket, the first (blue) bar shows the mean RS of the 1000 disks within that bucket as of the first month, and the second (magenta) bar shows the mean reallocations as of the second month.

The data shows that drives with less than 100 RS developed another 6 RS on average, while drives with RS in the range of 100 and 200 developed 100 more on average, well more than the aforementioned set. A similar

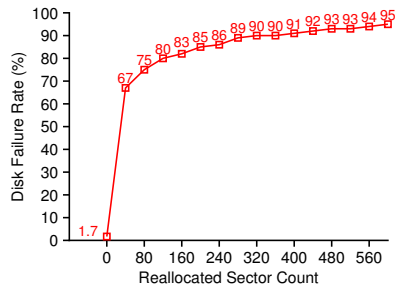


Figure 9: **Disk failure rate given different reallocated sector count.** *The failure probability increases quickly with more reallocated sectors. Shown for A-2.*

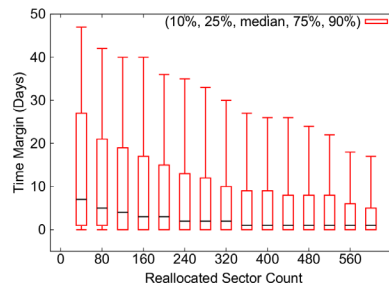


Figure 10: **Disk failure time given different reallocated sector count.** *The time margin decreases rapidly with more reallocated sectors. Shown for A-2.*

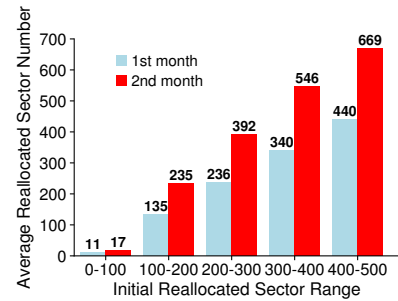


Figure 11: **Month-over-month comparison of reallocated sectors, grouped by first month's count.** *The mean reallocations in each bucket increase 50–75% across months. Shown for A-2.*

trend has been observed in other drive sets. In general, no matter how many existing sectors the disks have, the number of RS grows consistently.

Our analysis of other disk models is trending in the same direction of all the observations of model A-2. A slight difference is that the latest disk models can survive for a longer time with a certain number of RS. Therefore, the latest disk drives have a greater time margin as the number of RS grows.

From these experiments, we conclude that the accumulation of sector errors contributes to the whole-disk failure, causing disk reliability to deteriorate continuously. The more RS errors the drive has, the higher the probability to fail shortly or suffer a larger burst of sector errors. Therefore, the number of RS is a good criteria to reflect the disk survivability and sector reliability.

4 PLATE: Individual Failures

Much of the previous research on RAID has focused on improving redundancy schemes to tolerate more simultaneous failures [13, 30, 32, 39, 40]. However, our data analysis reveals that the likelihood of simultaneous whole-disk failures increases considerably with older disks. Further, the accumulation of sector errors contributes to whole-disk failures, causing the disk reliability to deteriorate continuously. Hence, ensuring data reliability in the worst case requires adding considerable extra redundancy, making the traditional passive approach of RAID protection unattractive from a cost perspective.

Meanwhile, the RS count has been observed to be a good criteria to quantify and predict the degree of deterioration of disk reliability. Therefore, we can upgrade the passive RAID protection into a proactive defense: PLATE monitors disk health (§4.1), identifies unreliable disks (§4.2), and replaces unstable disks in advance to

prevent failures. Since unreliable disks are detected and removed promptly, the likelihood of simultaneous failures also decreases (§4.3).

4.1 Monitor Disk Status

Our previous analysis reveals that the number of RS is a good criteria to identify unstable disks. This expands the role of disk scrubbing: originally, scrubbing aimed to verify data accessibility and proactively detect lost data on failed sectors which could be recovered through RAID redundancy; thus, it only scans “live” sectors (those storing data accessible through the file system). The new findings show that recognizing *all* latent sector errors in a timely fashion is invaluable for monitoring the status of a DG, so our scrubbing is being updated to periodically check even unused disk sectors. We then monitor the status of each disk via daily system logs, and when a disk’s RS count exceeds a threshold, its replacement is automatically triggered.

4.2 Proactively Identify Unreliable Disks

We see that the accumulation of sector errors contributes to whole-disk failures, causing disk reliability to deteriorate continuously. Hence, using the RS count can predict impending disk failures in advance. Such proactive protection provides administrators the chance to replace disks before whole-disk failures happen, improving RAID availability. We evaluate the methodology of the proactive protection through simulations based on historical disk information. We provide the result of deployment in production systems in §4.3.

If the RS count exceeds the given failure threshold T , the disk is considered to be unreliable. We evaluate the result using two curves that represent the trade-off between the fraction of failures successfully predicted

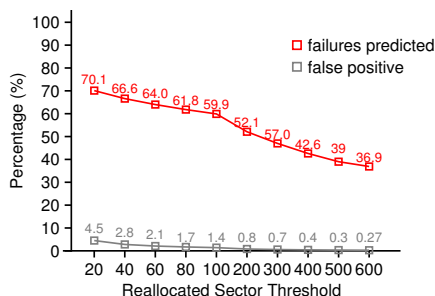


Figure 12: **Failure captured rate given different reallocated sector count.** Both the predicted failure and false positive rates decrease as the threshold increases. Shown for A-2.

(i.e., the *recall* of the prediction), and the false positive amount, which includes qualified working disks identified incorrectly. The impending whole-disk failure is defined as the disk failing within a 60-day observation window. If a disk that has more RS than the threshold is still functioning properly after the observation window, it is regarded as a false positive. Similarly, if a failed disk reports at least the given minimum number of RS within 60 days prior to the failure, the failure is successfully predicted. By comparing these two curves over the whole range of the identification threshold, we take into account all possible cost-based scenarios in terms of the trade-off between missing impending disk failures versus failing working ones incorrectly.

We measure the proactive protection on a population of 100,000 A-2 disks as reported by autosupport logs, and present the result in Figure 12. It shows that both the successful prediction rate and the false positive rate decrease smoothly as the RS threshold grows from 20 to 600. When the threshold is less than 200, it captures nearly 52–70% impending whole-disk failures, with 0.8–4.5% false positive rates. The majority of the unpredicted failures are caused by hardware faults, user error and other unknown reasons, which are unpredictable from a software perspective; these prediction rates are consistent with the curve for A-2 in Figure 6, which depicted the fraction of failed disks that had encountered a given number of RS. Other disk models demonstrate similar trends in our experiments.

System administrators can decide the appropriate threshold to fail disks based on their expectation of captured rate, tolerance of replacing disks prematurely, and the time required to replace disks.

4.3 Deployment Result

PLATE, the single-disk proactive protection using remapped sector count, has been incorporated into some

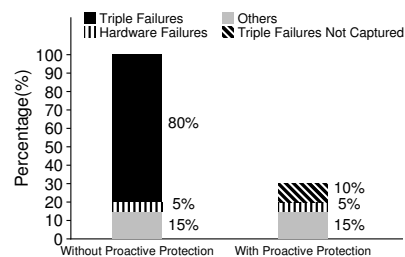


Figure 13: **Causes of recovery incidents.** The distribution of causes of RAID failures, before and after proactive protection was deployed, normalized to the case without protection. Single disk proactive protection reduces about 70% of RAID failures and avoids 88% of the triple-disk failures previously encountered.

production systems. In our initial deployment, affecting disks A-1, A-2, and B-1, we set the threshold for predicting failure at 200 RS. This threshold was based on the “training set” of our analysis prior to deployment and was selected for two reasons: first, replacing disks in production systems may take up to 3 days in the worst case, and second, the median time to failure drops to less than 3 days when the count of RS grows beyond 200. In other words, setting the threshold less than 200 provides enough time to fix 50% of those impending failures proactively. In addition, the cost of replacing a working disk by mistake requires us to strive for a false positive rate less than 1% (i.e., < 1% unnecessarily added costs from incorrectly replacing working drives), resulting in a replacement threshold of at least 200.

Figure 13 compares the recovery incidents caused by RAID failures before and after proactive protection was added to our systems. The graphs are normalized to the average number of RAID failures per month before the deployment, which are dominated by triple failures (80%), the results of some combination of whole-disk failures and sector errors. Another 5% are due to other hardware faults (for example, errors in host bus adapters, cables and shelves), while the remaining 15% are caused by factors such as user error and other unknown reasons.

While it is a challenge to reduce failures due to hardware faults and other errors, single-disk proactive protection detects unstable drives before their reliability is further deteriorated and triggers the DG reconstruction promptly, reducing the likelihood of multiple simultaneous failures. We find this eliminates about 88% of recovery incidents caused by triple failures, equivalent to about 70% of all disk-related incidents. This disproportionate reduction in DG errors (relative to the fraction of individual disk failures we can predict) is because we only need to avoid one out of the three disk failures

that would disable a RAID-6 DG.³ The remaining 12% of triple failures are due to sudden failures or multiple “somewhat unreliable” disks, all of which have a number of RS but none of which exceeds the failure threshold; we address these in §5. All the proactively replaced disks subsequently undergo rigorous testing by our company upon their return; the specialists analyzing these disks have not seen a noticeable number of false positives upon replacement.

5 ARMOR: Multiple Failures

Single-disk proactive protection (PLATE) identifies and fails unreliable disks in advance, which can prevent potential data loss by reducing the likelihood of multiple simultaneous failures. But PLATE will wait patiently for one disk to exceed a threshold before sounding an alarm. Disks can fail quickly after exceeding that threshold and will sometimes fail before it is even reached. If several disks are close to being declared near imminent failure, they may collectively put the DG at high enough risk to take action. At the same time, simply replacing all unreliable disks is not the most efficient approach, because not every impending disk failure will lead to a RAID failure. If disk failures are within the tolerance of RAID redundancy, repair efforts may be better directed elsewhere: i.e., administrators might triage to prioritize another DG at higher risk. (Refer to the example in §2.3.)

The next subsection (§5.1) introduces how we quantify the degree of RAID reliability and identify a vulnerable RAID, which is likely to lose redundancy in the face of multiple unreliable disks. §5.2 presents some simulation results using the ARMOR technique, and §5.3 discusses ongoing work.

5.1 Identifying Vulnerabilities

The accumulation of sector errors contributes to whole-disk failures, causing the RAID reliability to deteriorate continuously. Therefore, we can quantify and predict the single disk reliability with its number of existing RS, which can be further used to evaluate the degree of RAID reliability deterioration through joint probability. There are two steps in this process.

Calculate the probability of single whole-disk failure: Our previous analysis reveals that the RS count reflects the likelihood of whole-disk failure. This probability is calculated as follows. We define:

- $P(\text{fail})$ as the probability of disk failure
- N_{RS} as the observed number of reallocated sectors

³It may also arise from differences in the rate of failures over time, something that is difficult to assess.

- $P(N_{RS})$ as the probability that a disk has a reallocated sector count larger than N_{RS}
- $P(\text{fail}|N_{RS})$ as the probability of a whole-disk failure given at least N_{RS} reallocated sectors
- $P(N_{RS}|\text{fail})$ as the probability that a failed disk has a reallocated sector count larger than N_{RS}

$$\begin{aligned}
 P(\text{fail}|N_{RS}) &= \frac{P(N_{RS}|\text{fail}) \times P(\text{fail})}{P(N_{RS})} \\
 &= \frac{\frac{\text{num. of failed disks with } N_{RS}}{\text{num. of failed disks}} \times \frac{\text{num. of failed disks}}{\text{num. of disks}}}{\frac{\text{num. of all disks with } N_{RS}}{\text{num. of disks}}} \\
 &= \frac{\text{num. of failed disks with } N_{RS}}{\text{num. of all disks with } N_{RS}}
 \end{aligned}$$

Figure 14: **Formula of calculating the probability of whole-disk failure given a certain number of reallocated sectors.**

Ultimately we want to compute $P(\text{fail}|N_{RS})$, which can be calculated according to Bayes’s Theorem (the first line of Figure 14).

Calculate the probability of a vulnerable RAID: Our storage system uses RAID-6, which can tolerate two simultaneous failures. We define *RAID vulnerability* as the probability of a RAID system having more than one disk failure. Specifically, we use the formula introduced in Figure 14 to calculate the failure probability of each disk given its reallocated sector count. The combination of these single disk probabilities allows us to compute RAID vulnerability using the formula shown in Figure 15. A similar methodology can be applied to other redundant disk systems (e.g., RAID-5).

5.2 Simulation Result

We evaluate our methodology of identifying vulnerable RAID DGs. Specifically, we analyze historical disk failures recorded in our logs and categorize their corresponding RAID DGs into two subsets: “good” RAID DGs with no disk failures (subset G) and “bad” RAID DGs with more than one disk failure (subset B). We use their reallocated sector counts (one or more days prior to a failure, in the case of subset B) as an input to compute the probability of RAID vulnerability through our measurement. If our approach can effectively identify vulnerable RAID DGs, the calculated probability of most DGs in subset B should be considerably larger than that of the majority of DGs in subset G.

We use one-year disk historical data to build the statistical model and collect 5000 DGs for G and 500 DGs for B respectively from other years. Deciles are used to summarize the distribution of vulnerable probability of these

$$\begin{aligned}
P(\text{vulnerable RAID} | RS_1, RS_2, \dots, RS_N) &= P(\geq 2 \text{ disks fail} | RS_1, RS_2, \dots, RS_N) \\
&= 1 - P(0 \text{ disk fail} | RS_1, RS_2, \dots, RS_N) - P(1 \text{ disk fails} | RS_1, RS_2, \dots, RS_N) \\
P(0 \text{ disk fail} | RS_1, RS_2, \dots, RS_N) &= \prod_{i=1}^N (1 - P(i_{th} \text{ disk fails} | RS_i)) \\
P(1 \text{ disk fails} | RS_1, RS_2, \dots, RS_N) &= \sum_{j=1}^N P(j_{th} \text{ disk fails} | RS_j) \prod_{i=1, i \neq j}^N (1 - P(i_{th} \text{ disk fails} | RS_i))
\end{aligned}$$

N is the number of disks in a RAID DG, RS_i represents the reallocated sector count of disk i . $P(i_{th} \text{ disk fails} | RS_i)$ represents the failure probability of i_{th} disk given RS_i reallocated sector count.

Figure 15: Formula of calculating the probability of a vulnerable RAID DG.

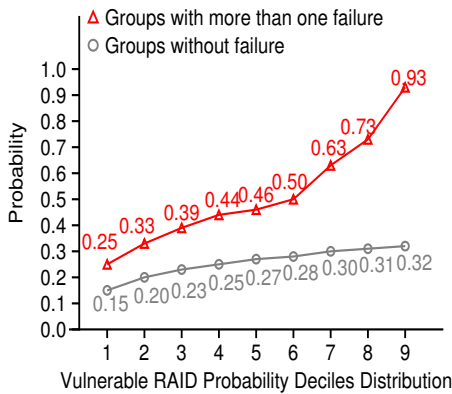


Figure 16: Deciles Comparison of Vulnerable RAID probability.

two subsets. The result is presented in Figure 16, which shows that probabilities of 90% of DGs in subset G are less than 0.32, while probabilities of most DGs in subset B are between 0.25 and 0.93. This probability discrimination between subset G and B show the methodology of identifying vulnerable RAID is effective to recognize endangered DGs, which are likely to have more than one disk failures. For example, when its probability grows to more than 0.32, we can regard this DG as a vulnerable one with high confidence. This threshold can capture more than 80% of vulnerable RAID DGs. Administrators can rely on this monitoring mechanism to keep track of disk statuses, recognize endangered RAID DGs, and trigger the appropriate proactive protection mechanism.

We examined the 12% of triple-disk failures that were not prevented by PLATE, looking at logs reported 3–7 days prior to the failure. (The duration varies depending on when the reports were made.) In 80% of the cases, ARMOR computed a failure probability of 80–95% despite no single disk being above the 200 RS threshold; this indicates that between PLATE and ARMOR, we could potentially prevent 98% of triple failures. Although the results of this analysis are based on a small sample, we are encouraged by the possibility of elim-

inating nearly failures resulting from triple-disk errors. However, greater attention will be needed for the 20% of RAID recovery incidents due to other causes.

5.3 Ongoing and Future Work

Incorporating the RAID monitoring mechanism into production systems has some operational considerations. We are upgrading our monitoring and logging mechanisms to recognize and record the reason for disk failure, as well as quantifying the variance of parameters of the statistical model, so we can activate the DG monitoring mechanism in our production systems.

We are considering methods to put a potentially failing disk “on probation” to test whether it is truly failing. This would be especially useful in cases where individual disks are not above a threshold for replacement but the availability of the DG as a whole is in doubt. Spare disks could be brought on-line while suspect disks get scrubbed thoroughly.

It would be interesting to extend ARMOR to other configurations, such as erasure coding, or to consider replicated data. A disk group or erasure coding system might be vulnerable when a given number of disks fail, but the data stored on it would be recoverable from another replica at a high cost. What if the replica is also vulnerable? The joint probability of multiple replicas failing simultaneously should by necessity be comparatively low, but it should be quantified.

Finally, it will be important to gain more operational experience with both PLATE and ARMOR on a greater variety of disk models over a greater period of time. Tuning the thresholds for the characteristics of each system will be important; even within a model, we would like to test different thresholds on a limited set of disks to determine the accuracy of our chosen threshold. Sensitivity to application workloads may also prove interesting: a backup storage system sees different read-write workloads than a primary system [46].

6 Related Work

Early work by Gibson, et al. [18, 36] and Chen, et al. [11] evaluates RAID reliability assuming an ideal world with independent failures, exponential lifetimes, and instantaneous failures. Unfortunately, the fault model presented by modern disk drives is more complex. Schroeder and Gibson [42] analyze 100,000 disks to reject the hypothesis that time between disk failure/replacement follows an exponential distribution. Bairavasundaram et al. [6] reveal the potential risk of sector errors during RAID reconstruction, which is not predicted in the early RAID reliability model. Researchers have since noted that the original RAID reliability model has outlived its useful life and built new models to depict RAID reliability [16, 21].

Given the presence of these new disk failure modes, many mechanisms have been built to improve system reliability. Schwarz, et al. [43] propose disk scrubbing to proactively detect latent sector errors. Many new storage arrays adopt extra levels of redundancy to tolerate more failures [12, 19]. File systems also detect and handle disk faults through checksums and replication. For example, in addition to using RAID techniques, ZFS employs checksums to detect block corruption and keep replicas of certain "important" on-disk blocks to tolerate disk faults [10]. The IRON file system applies similar techniques to improve robustness of commodity file systems [41]. Another related approach is to tolerate disk faults at the application-level [44, 17].

Unfortunately, our previous analysis reveals that whole-disk failure and sector errors are strongly correlated. Further, the likelihood of such simultaneous failures is not consistent over time. Ensuring data reliability in the worst case requires adding considerable extra redundancy, which adds unnecessary costs.

Alternatively, a number of previous approaches seek indicators of impending failures. In particular, Pinheiro et al [38] study the failure characteristics of consumer-grade disk drives used in Google's services. They find that most SMART error metrics such as reallocated sectors strongly suggest an impending failure, but they also determine that half of failed disks show no such errors. We find that some disks (such as C-1 and D-1) frequently do not report errors before failing, but several models correlate well. These differences are presumably due to differences in disk models as well as workloads: since our systems rewrite data upon error, we may trigger remappings in ways their systems would not.

Goldszmidt [20] seeks to predict whole-disk failures with a performance signal, particularly the average maximum latency. Murray et al. [26, 34, 35] and Hamerly et al. [24] also attempt to improve whole-disk failure prediction by applying various advanced data mining al-

gorithms on SMART [1] analytic data. In comparison, our work is on a much larger population of production disks with a focus on the correlation between whole-disk failures and sector errors, both of which affect the data safety. We quantitatively evaluate their correlation, and reveal that the RS count is a good criteria to reflect disk survivability and the sector reliability, which is then used to proactively recognize unstable disks and vulnerable RAID DGs.

7 Conclusion

In this paper, we present and analyze disk failure data from a large number of backup systems, including some of the world's largest enterprises. Our analysis reveals that the accumulation of *reallocated sectors*, a specific type of sector error, causes the disk reliability to deteriorate continuously. Therefore, the RS count can be used as an indicator to quantify and predict the degree of deterioration in disk reliability.

With these findings we designed RAIDSHIELD, consisting of PLATE and ARMOR. PLATE monitors disk health by tracking the number of RS and proactively detecting unstable disks; the deployment of single-disk proactive protection has eliminated 70% of RAID failures in production systems. With ARMOR, we aim to quantify the deterioration of RAID reliability and detect vulnerable RAID DGs in advance, even when individual disks have not degraded sufficiently to trigger alarms. Initial results with ARMOR suggest that it can eliminate most of the remaining triple-disk errors not identified by PLATE.

While we expect that the techniques presented here apply to all storage systems, the specific analyses were performed on backup systems with particular I/O patterns [46]. Extending the analysis and evaluating these techniques in other environments are promising future work.

Acknowledgments

We thank Andrea Arpaci-Dusseau, Remzi Arpaci-Dusseau, Stephen Manley, Philip Shilane, Grant Wallace, and members of the EMC Data Protection CTO office for comments and feedback on earlier drafts and discussions. We especially thank our shepherd, Haryadi Gunawi, and the anonymous reviewers; their feedback and guidance have been especially helpful.

In addition, we thank the Data Domain RAID, I/O, and customer support teams for feedback on the disk analyses and deployment. We thank Yanpei Chen, Rean Griffith, and Cheng Li for insightful comments on data science.

References

- [1] B. Allen. Monitoring hard disks with S.M.A.R.T. *Linux Journal*, 2004.
- [2] G. A. Alvarez, W. A. Burkhard, and F. Cristian. Tolerating multiple failures in RAID architectures with optimal storage and uniform declustering. In *Proceedings of the 24th International Symposium on Computer Architecture (ISCA'97)*, pages 62–72, Denver, CO, USA, 1997.
- [3] A. Amer, D. D. Long, and S. Thomas Schwarz. Reliability challenges for storing exabytes. In *Computing, Networking and Communications (ICNC), 2014 International Conference on*, pages 907–913. IEEE, 2014.
- [4] D. Anderson, J. Dykes, and E. Riedel. More Than an Interface: SCSI vs. ATA. In *Proceedings of the 2nd USENIX Conference on File and Storage Technologies (FAST'03)*, pages 245–257, San Francisco, CA, USA, Apr. 2003.
- [5] R. H. Arpaci-Dusseau and A. C. Arpaci-Dusseau. Fail-Stutter Fault Tolerance. In *The Eighth Workshop on Hot Topics in Operating Systems (HotOS VIII)*, pages 33–38, Schloss Elmau, Germany, May 2001.
- [6] L. N. Bairavasundaram, G. R. Goodson, S. Pasupathy, and J. Schindler. An analysis of latent sector errors in disk drives. In *Proceedings of the 2007 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '07)*, pages 289–300, San Diego, CA, USA, June 2007.
- [7] L. N. Bairavasundaram, G. R. Goodson, B. Schroeder, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau. An Analysis of Data Corruption in the Storage Stack. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies (FAST '08)*, San Jose, California, February 2008.
- [8] W. Bartlett and L. Spainhower. Commercial fault tolerance: A tale of two systems. *IEEE Trans. Dependable Secur. Comput.*, 1(1):87–96, Jan. 2004.
- [9] P. Bodik, M. Goldszmidt, A. Fox, D. B. Woodard, and H. Andersen. Fingerprinting the datacenter: Automated classification of performance crises. In *Proceedings of the 2010 EuroSys Conference (EuroSys '10)*, pages 111–124, Paris, France, Apr. 2010.
- [10] J. Bonwick and B. Moore. Zfs: The last world in file systems. In *SNIA Software Developers's Conference*, Santa Clara, CA, Sept. 2008.
- [11] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson. RAID: High-performance, reliable secondary storage. *ACM Comput. Surv.*, 26(2):145–185, June 1994.
- [12] P. Corbett, B. English, A. Goel, T. Grcanac, S. Kleiman, J. Leong, and S. Sankar. Row-diagonal parity for double disk failure correction. In *Proceedings of the 3th USENIX Conference on File and Storage Technologies (FAST '04)*, page 14, San Francisco, CA, Apr. 2004.
- [13] A. Dholakia, E. Eleftheriou, X.-Y. Hu, I. Iliadis, J. Menon, and K. Rao. A new intra-disk redundancy scheme for high-reliability raid storage systems in the presence of unrecoverable errors. *ACM Transactions on Storage*, 4(1):1:1–1:42, May 2008.
- [14] C. Dubnicki et al. HYDRAsstor: A scalable secondary storage. In *Proceedings of the 7th USENIX Conference on File and Storage Technologies (FAST'09)*, 2009.
- [15] J. Elerath. Hard-disk drives: the good, the bad, and the ugly. *Commun. ACM*, 52(6):38–45, June 2009.
- [16] J. G. Elerath and J. Schindler. Beyond MTDDL: A closed-form RAID 6 reliability equation. *ACM Trans. Storage*, 10(2):7:1–7:21, Mar. 2014.
- [17] S. Ghemawat, H. Gobioff, and S.-T. Leung. The Google file system. In *Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP'03)*, pages 29–43, Bolton Landing, NY, USA, Oct. 2003.
- [18] G. Gibson. *Redundant disk arrays: Reliable, parallel secondary storage*. PhD thesis, University of California, Berkeley, CA, USA, 1992.
- [19] A. Goel and P. Corbett. RAID triple parity. *ACM SIGOPS Oper. Syst. Rev.*, 46(3):41–49, Dec. 2012.
- [20] M. Goldszmidt. Finding soon-to-fail disks in a haystack. In *USENIX HotStorage'12*, Boston, MA, USA, June 2012.
- [21] K. M. Greenan, J. S. Plank, and J. J. Wylie. Mean time to meaningless: MTDDL, markov models, and storage system reliability. In *USENIX HotStorage'10*, Boston, MA, Oct. 2010.
- [22] J. L. Hafner. Weaver codes: Highly fault tolerant erasure codes for storage systems. In *Proceedings*

- of the 4th Conference on USENIX Conference on File and Storage Technologies (FAST'05), 2005.
- [23] J. L. Hafner, V. Deenadhayalan, K. K. Rao, and J. A. Tomlin. Matrix methods for lost data reconstruction in erasure codes. In *Proceedings of the 4th Conference on USENIX Conference on File and Storage Technologies (FAST'05)*, 2005.
- [24] G. Hamerly and C. Elkan. Bayesian approaches to failure prediction for disk drives. In *ICML'01*, pages 202–209, Williamstown, MA, USA, June 2001.
- [25] C. Huang et al. Erasure coding in windows azure storage. In *USENIX Annual Technical Conference*, 2012.
- [26] G. F. Hughes, J. F. Murray, K. Kreutz-Delgado, and C. Elkan. Improved disk-drive failure warnings. *IEEE Transactions on Reliability*, 51(3):350–357, Sept. 2002.
- [27] N. Jain, M. Dahlin, and R. Tewari. TAPER: tiered approach for eliminating redundancy in replica synchronization. In *Proceedings of the 4th Conference on USENIX Conference on File and Storage Technologies (FAST'05)*, 2005.
- [28] H. Kari, H. Saikkonen, and F. Lombardi. Detection of defective media in disks. In *IEEE Workshop on Defect and Fault Tolerance in VLSI Systems*, pages 49–55, Venice, Italy, Oct. 1993.
- [29] H. H. Kari. *Latent Sector Faults and Reliability of Disk Arrays*. PhD thesis, Helsinki University of Technology, Espoo, Finland, May 1997.
- [30] O. Khan, R. Burns, J. S. Plank, W. Pierce, and C. Huang. Rethinking erasure codes for cloud file systems: Minimizing I/O for recovery and degraded reads. In *Proceedings of the 10th USENIX Conference on File and Storage Technologies (FAST'12)*, San Jose, February 2012.
- [31] A. Krioukov, L. N. Bairavasundaram, G. R. Goodson, K. Srinivasan, R. Thelen, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau. Parity Lost and Parity Regained. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies (FAST'08)*, San Jose, California, February 2008.
- [32] M. Li, J. Shu, and W. Zheng. Grid codes: Strip-based erasure codes with high fault tolerance for storage systems. *ACM Transactions on Storage*, 4(4):15:1–15:22, Feb. 2009.
- [33] C. Mellor. Kryder's law craps out: Race to UBER-CHEAP STORAGE is OVER. *The A Register*, 2014. http://www.theregister.co.uk/2014/11/10/kryders_law_of_ever_cheaper_storage_disproven.
- [34] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado. Hard drive failure prediction using non-parametric statistical methods. In *ICANN/ICONIP*, Istanbul, Turkey, June 2003.
- [35] J. F. Murray, G. F. Hughes, and D. Schuurmans. Machine learning methods for predicting failures in hard drives: A multiple-instance application. In *Journal of Machine Learning research*, volume 6, pages 783–816, May 2005.
- [36] D. Patterson, G. Gibson, and R. Katz. A Case for Redundant Arrays of Inexpensive Disks (RAID). In *Proceedings of the 1988 ACM SIGMOD Conference on Management of Data (SIGMOD'88)*, pages 109–116, 1988.
- [37] H. Patterson, S. Manley, M. Federwisch, D. Hitz, S. Kleiman, and S. Owara. SnapMirror: file system based asynchronous mirroring for disaster recovery. In *Proceedings of the 1st USENIX Conference on File and Storage Technologies*, 2002.
- [38] E. Pinheiro, W.-D. Weber, and L. A. Barroso. Failure trends in a large disk drive population. In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST '07)*, pages 17–28, San Jose, CA, USA, Feb. 2007.
- [39] J. S. Plank and M. Blaum. Sector-disk (SD) erasure codes for mixed failure modes in RAID systems. *ACM Transactions on Storage*, 10(1), January 2014.
- [40] J. S. Plank, M. Blaum, and J. L. Hafner. SD codes: Erasure codes designed for how storage systems really fail. In *Proceedings of the 11th USENIX Conference on File and Storage Technologies (FAST'13)*, San Jose, February 2013.
- [41] V. Prabhakaran, L. N. Bairavasundaram, N. Agrawal, H. S. Gunawi, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau. IRON File Systems. In *Proceedings of the 20th ACM Symposium on Operating Systems Principles (SOSP'05)*, pages 206–220, Brighton, United Kingdom, Oct. 2005.
- [42] B. Schroeder and A. G. Gibson. Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you? In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST '07)*, San Jose, CA, Feb. 2007.

- [43] T. J. Schwarz, Q. Xin, E. L. Miller, D. D. E. Long, A. Hospodor, and S. Ng. Disk scrubbing in large archival storage systems. In *IEEE MASCOTS'04*, pages 409–418, Volendam, The Netherlands, Oct. 2004. IEEE Computer Society.
- [44] K. Shvachko, H. Kuang, S. Radia, and R. Chansler. The hadoop distributed file system. In *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, pages 1–10. IEEE, 2010.
- [45] N. Talagala and D. Patterson. An Analysis of Error Behaviour in a Large Storage System. In *The IEEE Workshop on Fault Tolerance in Parallel and Distributed Systems*, San Juan, Puerto Rico, Apr. 1999.
- [46] G. Wallace, F. Douglis, H. Qian, P. Shilane, S. Smaldone, M. Chamness, and W. Hsu. Characteristics of Backup Workloads in Production Systems. In *Proceedings of the 10th Conference on File and Storage Technologies (FAST'12)*, 2012.
- [47] C. Walter. Kryder's law. *Scientific American*, 293(2):32–33, 2005.