NVMe SSD Failures in the Field: the Fail-Stop and the Fail-Slow

Ruiming Lu, *Shanghai Jiao Tong University;* Erci Xu, *PDL;* Yiming Zhang, *Xiamen University;* Zhaosheng Zhu, Mengtian Wang, and Zongpeng Zhu, *Alibaba Inc.;* Guangtao Xue, *Shanghai Jiao Tong University;* Minglu Li, *Shanghai Jiao Tong University & Zhejiang Normal University;* Jiesheng Wu, *Alibaba Inc.*

https://www.usenix.org/conference/atc22/presentation/lu

This paper is included in the Proceedings of the 2022 USENIX Annual Technical Conference.

July 11–13, 2022 • Carlsbad, CA, USA

978-1-939133-29-8

Open access to the Proceedings of the 2022 USENIX Annual Technical Conference is sponsored by

NVMe SSD Failures in the Field: the Fail-Stop and the Fail-Slow

Ruiming Lu[1]*, Erci Xu[2]*, Yiming Zhang[3]†, Zhaosheng Zhu[4], Mengtian Wang[4], Zongpeng Zhu[4], Guangtao Xue[1]†, Minglu Li[1,5], and Jiesheng Wu[4]

[1]Shanghai Jiao Tong University, [2]PDL, [3]Xiamen University,
[4]Alibaba Inc., and [5]Zhejiang Normal University

Abstract

NVMe SSD has become a staple in modern datacenters thanks to its high throughput and ultra-low latency. Despite its popularity, the reliability of NVMe SSD under mass deployment remains unknown. In this paper, we collect logs from over one million NVMe SSDs deployed at Alibaba, and conduct extensive analysis. From the study, we identify a series of major reliability changes in NVMe SSD. On the good side, NVMe SSD becomes more resilient to early failures and variances of access patterns. On the bad side, NVMe SSD becomes more vulnerable to complicated correlated failures. More importantly, we discover that the ultra-low latency nature makes NVMe SSD much more likely to be impacted by fail-slow failures.

1 Introduction

NVMe SSD is now the new favorite of modern data centers. With a performance specification of up to 6GB/s bandwidth and microsecond-level latency, NVMe SSD serves as a strong performance upgrade to its SATA-based peers [8, 18, 29–31].

Apart from the performance, the reliability of any hardware under mass deployment is of great concern [3, 5–7, 10, 14, 38, 40, 42, 45]. While there is a spate of work covering the failure characteristics of SATA SSDs in the field [34–36, 41, 47], their findings may not be conclusive for NVMe SSD.

First, with a low-latency interface, NVMe SSD can be especially *prone* to fail-slow failure (aka. gray failure [17, 21, 25, 26, 48]). In a nutshell, the NVMe SSD fail-slow failure causes a drive to exhibit abnormal performance slowdown (e.g., high latency under normal traffic). Unlike SATA SSD, where fail-slow failure may be masked by the relatively high latency (>100μs), NVMe SSD can be easily impacted due to its ultra-low latency nature ($\sim$ 10μs) [23, 27, 28].

Moreover, the NVMe SSD is not just the SATA SSD with an interface upgrade. Instead, the internal architecture of NVMe SSD has gone through considerable changes. An outstanding example is the wide adoption of 3D-TLC NAND in NVMe SSD for larger capacity. Compared to MLC, the denser bits per cell (i.e., TLC) shows lower reliability and the vertical stacking (i.e., 3D flash) can exhibit disparate behaviors or even opposite patterns (e.g., lower error rate under higher temperatures [32]). Also, the vendors have integrated a series of techniques to improve the overall reliability in NVMe SSD, such as Redundant Array of Independent NAND (RAIN) or Low-Density Parity-Check code (LDPC) [43, 50]. Unfortunately, with no large-scale NVMe SSD fail-stop study available at the moment, the influences of recent advancements remain unknown.

In this paper, we study the fail-stop and fail-slow failures of NVMe SSDs deployed at Alibaba. Specifically, we collect and analyze device logs (i.e., SMART [11]), runtime logs (i.e., `iostat`), and failure tickets from over one million NVMe SSDs[1]. Throughout the study, we set our analysis into the context of previous studies to help various parties of interest get a clear picture of NVMe SSD reliability, including the improving and deteriorating failure patterns of fail-stop failures and the characteristics regarding the fail-slow failures.

We start our study by plotting and analyzing the baseline statistics (§3) of the NVMe SSDs, including the drive characteristics (e.g., manufacturer and model), usage characteristics (e.g., power-on time), and health metrics (e.g., annual replacement rate). Then, we comb through the dataset against different impact factors such as age and write amplification (§4). Finally, we lay a special focus on the fail-slow failures (§5), where we rigorously identify the fail-slow drives and perform extensive analysis. Altogether, we obtain 10 major findings and we list the highlights as follows:

- Infant mortality (failures occurring soon after deployment), a concerning failure trend in SATA SSD [35], is not outstanding in NVMe SSD. For nearly all of our models, the failure rate in the first three months is equivalent to or even less than that from later periods.
- High Write Amplification Factor (WAF), unlike SATA SSD [36], is no longer closely correlated with failures. Interestingly, NVMe SSD with low WAF (WAF$\leq$1) exhibits 2.19$\times$ higher ARR than high-WAF ones.
- Co-located (i.e., intra-node/rack) NVMe SSD failure becomes more temporally correlated. For example, compared to SATA SSD, NVMe SSD correlated failure increases up

*Equal contribution.
†Corresponding authors.

[1]We release our dataset at https://tianchi.aliyun.com/dataset/dataDetail?dataId=128972.

to 14.69× and 1.78× in intra-node/rack scenarios, respectively.
- The fail-slow failure is a widespread and severe problem for NVMe SSD. On average, 1.41% of NVMe SSDs are infected within four-month monitoring, which is 6.05× that of HDD. Besides, fail-slow NVMe SSD could degrade to SATA SSD or even HDD performance.
- The NVMe SSD fail-slow failure does not correlate with SMART attributes, and rarely (0.22% of the fail-slow drives) transits to fail-stop failures.

We conclude this paper with the limitation of this study (§6), the related work (§7) and a short conclusion (§8).

2 Background

2.1 System Architecture

The NVMe SSDs, in our study, come from multiple IDCs (Internet Data Centers) across the globe. An IDC usually hosts dozens of storage clusters. The clusters are homomorphic with each running an HDFS-like distributed file system (DFS). Each cluster owns several to tens of racks and each rack includes up to 48 nodes. All NVMe SSDs in our study come from the all-flash-configuration nodes that contain 12 NVMe SSDs (not RAIDed) for data storage (not hosting OS).

2.2 Drive Model & Workload

Our candidate SSDs are all enterprise-level. The earliest model was deployed around May 2015, while the latest model is from July 2019. We introduce the details in §3.

The SSD fleet serves a total of 7 services, including block storage, object storage, big data, buffering, log, streaming, and query. Each service spans across several dedicated clusters. For confidentiality, we do not share the numbers of SSDs or their distribution under each service. Still, we study the influences of workload under controlled-variable experiments.

2.3 Data Collection

Data	Span	Entry
SMART Logs	2019-11-04~2020-11-14	∼1.8M
Perf. Logs	2020-11-16~2021-03-05	∼84M
Failure Tickets	2019-11-04~2020-11-02	∼20K

Table 1: Data collection period (§2.3).

SMART logs. SMART is a set of attributes widely adopted by vendors and administrators to evaluate the reliability and performance of drives [11]. In our clusters, the reportings of SMART attributes are collected on a daily basis. Readings of the metrics can be either cumulative (e.g., number of media errors) or instantaneous (e.g., temperature). In practice, vendors may not necessarily follow the exact counting or reporting mechanism. Therefore, we standardize the numbers based on the manufacturer manuals.

Performance logs. A major subset of our clusters is equipped with node-level daemons to monitor and record the `iostat`, a Linux kernel performance log. The `iostat` includes vital statistics of storage devices, such as latency, IOPS and throughput. Currently, the daemon runs 3 hours a day (from 9 PM to 12 AM) and only records the average `iostat` values of each monitoring window (15 seconds long). Within the three hours, the traffic is relatively stable (around 70% peak traffic) and dominated mainly by internal workloads and large external clients (i.e., less burst traffic).

Failure tickets. Every node in our clusters has set up a daemon to monitor and report fail-stop failures. Upon reporting, a failure ticket would be generated (and manually checked by engineers), containing basic information of the victim drive (e.g., model and hostname) and the timestamp. Around 35% of our nodes also record an error code, detailing the direct symptom of failure (see Table 3 in §3.1). Upon failures, based on the symptoms, drives would be repaired online (e.g., fsck) or directly put offline for replacement (e.g., drive lost).

2.4 Methodology Correctness

To ensure sound and generalizable conclusions, we adhere to the following principles and measures throughout the study.

First, our study methodology (similar to [19,34,36,41,47]) starts with a general and extensive comparison to identify outstanding dominant factors. If high-level observation is fruitless or suspicious (e.g., spurious correlation led by inter-dependence between different factors as noted in [41]), we would then perform fine-grained controlled variable experiments (e.g., conditioned on workloads, drive models, drive age and the total bytes written) to unravel the underlying root causes and actionable advice for practitioners if any.

Second, we pre-screen the raw datasets to avoid bias (e.g., higher average) led by outliers (e.g., overflowing SMART values, NULL `iostat` recordings). Note that on-site engineers have manually verified all failure tickets before this study. In total, we have dropped around 5.8% and 1.5% untrustworthy records from SMART logs and `iostat`, respectively. Moreover, for generalizability concerns, we also exclude drive models with a smaller population (less than 1K) from the study. Note that different suppliers may register a model by different names, but we treat them as the same one here.

Third, we carefully choose statistical instruments to identify and verify the potential patterns in the NVMe SSD failures. Our rationale is that either such techniques or thresholds have been applied in previous studies, or clear documentation indicates the techniques can be used in the targeted scenarios.

3 Baseline Statistics

3.1 Dataset Overview

SMART logs. We begin by presenting the baseline statistics in Table 2, where the dataset is grouped into three categories: Basic Information, Usage Characteristics, and Health Metrics.

Basic Information					Usage Characteristics			Health Metrics				
Model	**Cap. (GB)**	**NAND**	**Lith./ Layer**	**Total (%)**	**Drive Years**	**OP**	**WAF**	**Crit. Warn.**	**CRC Err.**	**Media Err.**	**P/E Err.**	**ARR (%)**
I-A	800	MLC	15nm	0.1	3.32	28%	1.69	0.0015 / 0	1439.46 / 0	0 / 0	0 / 0	0.34
	2000	MLC	15nm	0.8	3.07	2%	2.05	0.027 / 0	759.73 / 0	3.52 / 0	0 / 0	0.69
	3840	MLC	15nm	0.1	2.87	7%	0.84	0.0025 / 0	3091.59 / 1	0 / 0	0 / 0	0.78
I-B	1600	MLC	15nm	0.7	2.73	28%	1.82	0.011 / 0	0 / 0	0.01 / 0	0 / 0	1.12
	3200	MLC	15nm	0.1	2.99	28%	1.86	0.16 / 0	0 / 0	759.81 / 0	0 / 0	2.34
I-C	4000	3D-TLC	64L	0.1	0.46	2%	1.04	0 / 0	0 / 0	0 / 0	0 / 0	0.66
II-A	1920	MLC	20nm	0.5	3.44	7%	3.68	0.052 / 0	59.46 / 0	0 / 0	1.70 / 0	0.77
II-B	800	MLC	20nm	0.7	3.60	28%	7.82	0 / 0	52.90 / 0	0 / 0	3.10 / 0	0.49
	1600	MLC	20nm	1.3	3.63	28%	7.97	0 / 0	43.52 / 0	2.69 / 0	5.80 / 0	0.63
II-C	960	3D-TLC	32L	3.4	2.55	7%	3.62	0 / 0	1572.77 / 0	0 / 0	0.79 / 0	0.52
	1920	3D-TLC	32L	1.8	2.50	7%	2.88	0.0017 / 0	849.99 / 0	0.49 / 0	1.60 / 0	0.79
	4000	3D-TLC	32L	5.5	2.39	2%	3.36	0.00079 / 0	957.86 / 0	0.34 / 0	3.60 / 1	0.64
II-D	960	3D-TLC	64L	4.9	1.47	7%	2.45	0.00026 / 0	38.66 / 0	1.45 / 0	0.38 / 0	0.26
	1920	3D-TLC	64L	8.4	0.97	7%	2.37	0.00031 / 0	54.56 / 0	0.45 / 0	0.45 / 0	0.56
	3840	3D-TLC	64L	45.3	0.69	7%	1.96	0.000038 / 0	32.72 / 0	5.53 / 0	0.66 / 0	1.12
II-E	370	NEW	20nm	0.5	1.24	0%	-	0 / 0	72.05 / 0	0.71 / 0	0 / 0	1.40
	750	NEW	20nm	0.7	0.18	0%	-	0 / 0	38.92 / 0	16.27 / 0	0 / 0	3.27
III-A	3200	3D-TLC	48L	0.3	2.65	28%	2.59	0 / 0	19.39 / 0	45.28 / 0	0.28 / 0	2.31
III-B	960	3D-TLC	48L	3.4	1.96	7%	3.34	0.0038 / 0	296.41 / 0	2.29 / 0	30.00 / 0	0.60
	1900	3D-TLC	48L	7.4	1.73	7%	2.78	0.0080 / 0	263.04 / 6	0.82 / 0	69.00 / 0	0.69
	3800	3D-TLC	48L	9.9	1.93	7%	1.87	0.010 / 0	469.66 / 6	1.81 / 0	67.00 / 0	1.13
III-C	960	3D-TLC	64L	4.1	0.45	7%	3.96	0.0023 / 0	124.55 / 0	0.02 / 0	5.30 / 0	0.49

Table 2: Baseline statistics of our drives (§3). *The table shows the summarized statistics of NVMe SSD fleet.* ***Cap.****: capacity;* ***NAND****: flash architecture;* ***Lith./Layer****: lithography or numbers of stacking layers;* ***OP****: Over-Provisioning rate;* ***WAF****: Write Amplification Factor;* ***Crit. Warn.****: critical warning;* ***P/E Err.****: program/erase error. In Health Metrics, the two values separated by a slash refer to mean and median values, respectively.*

In Basic Information, we name the drive models as manufacturer-model and use the alphabetic order to refer to the generations of a manufacturer (e.g., I-A stands for the earliest model from manufacturer I). Each model can be further specified by capacities (i.e., Cap. column) and NAND architecture (i.e., NAND column). We mark the planar chips with their lithography (e.g., 15nm for I-A) and the 3D chips with their vertical stacking layers (e.g., 64-layer for I-C). II-E is a unique case as it adopts a novel (neither planar nor 3D stacking) cell, thus named NEW (for anonymity). Finally, we list each model's relative population (i.e., Total%).

The Usage Characteristics describes the high-level administrative information. The first column is the average power-on time in terms of years. The second and third columns respectively present the over-provisioning rate (i.e., OP) and the calculated average Write Amplification Factor (i.e., WAF). WAF is calculated by dividing the number of NAND writes by the number of logical writes. Both numbers are reported by the SSD SMART attributes.

Last, we cover five primary reliability-related metrics.

• *Critical Warning,* introduced by NVM Express [20], indicates that the drive may have serious media errors (i.e., in read-only or degraded mode), possible hardware failures, or exceeding temperature alarm threshold.

• *CRC Error* refers to the number of transmission errors (e.g., the faulty interconnection between the drive and the host).

• *Media Error* refers to the number of data corruption errors (i.e., unable to access stored data in flash media).

• *Program/Erase Error* refers to the number of flash cell programming errors (e.g., unable to program flash cells from a block that is about to be garbage collected during copyback).

• *Annual Replacement Rate (ARR)* is the number of device failures divided by numbers of device years, reflecting the general reliability of drives (a common standard [34, 41]).

Note that readings of the first four health metrics are heavily biased where zeros account for an absolute majority (e.g., 99.97% for critical warning) of valid recordings. Hence, we list both average and median values (i.e., average/median).

Failure tickets. A subset (around 35%) of our drives details the direct cause upon failure reporting. Here, we present the distribution of their symptoms in Table 3. There are a total of five failure symptoms. I/O failure refers to a drive that fails to perform a read/write request. The Link failure indicates either a connection error during the PCI-e transmission or an abnormal bandwidth. The Lost failure refers to a functioning drive to become unfound. The Boot failure describes a drive that fails to initiate (e.g., mounting file system). The Thres. failure refers to one or more SMART attributes to have reached the

	Distribution Statistics				
Type	**Dist.**	**ARR**	**ARR_M**	**ARR_3D**	**ARR_N**
I/O	49.55%	0.40%	0.14%	0.42%	1.07%
Link	11.07%	0.09%	0.01%	0.10%	0.10%
Lost	5.65%	0.05%	0.06%	0.04%	0.01%
Boot	19.59%	0.16%	0.30%	0.14%	0.39%
Thres.	14.15%	0.11%	0.20%	0.10%	0.10%

Table 3: Failure symptom distribution (§3.1). ***Dist.****: distribution;* ***ARR****: overall ARR;* ***ARR_X****: ARR of drives under different flash architecture;* ***I/O****: read/write failure;* ***Link****: connection failure;* ***Lost****: drive unfound;* ***Boot****: booting failure;* ***Thres.****: SMART value over a pre-defined threshold.*

pre-defined threshold(s). For each type of failure, we present its distribution (Dist. column) along with the corresponding ARRs in all NVMe SSDs (ARR), MLC-based (ARR_M), 3D-TLC (ARR_3D), and NEW-NAND ones (ARR_N).

3.2 High Level Observations

Based on Tables 2 and 3, we now associate drive characteristics with health metrics to get a high-level understanding of the NVMe SSD fail-stop failures. Note that, even for the same model, the drive population can be a diverse mix of model and usage characteristics (e.g., NAND type, age, and total bytes written). We have further verified our observations through controlled-variable experiments on such impacting factors.

NVMe SSD vs. SATA/SAS SSD. The ARR of NVMe SSD in our dataset is much higher than that of SATA/SAS SSD from Netapp's enterprise storage systems [34]. We perform a t-test on both sets of ARRs and the corresponding p-value is equal to 3.554e-07. The average and median ARR of our NVMe SSD are 0.98% and 0.69%, which are 2.77× and 2.83× higher than those of SATA/SAS SSD respectively (i.e., 0.26% and 0.18% calculated from Table 1 in [34]). We obtain similar results as we further break down the SSD population by NAND types and lithography. Regarding Alibaba's data centers [19], the trend persists except for two models (i.e., C1 and C2 from Table 1 in [19]).

Moreover, we also compare the failure symptom distribution between NVMe SSD and SATA SSD (see Table 3 in [47]). We observe radical changes as I/O error becomes far more prevalent in NVMe SSD (accounting for 49.55%) while Lost error (i.e., drive unfound) is no longer dominant (i.e., 5.65% in NVMe SSD vs. 53.7% in SATA SSD).

Drive capacity. Within the same drive family, the average P/E errors and ARR are positively correlated with the capacity. For example, in the II-D drive family, as the capacity increases, the average P/E error rises from 0.38 to 0.66 and the ARR surges from 0.26% to 1.12%. This is understandable as drives with larger capacity are more likely to be accessed, thereby increasing the chances of suffering program errors.

NAND type. In our dataset, we find that the ARR of 3D-TLC drives is slightly lower than that of MLC drives, while in SATA SSD [34], the trend is reversed. The ARR of our MLC drives varies from 0.34% to 2.34%, while that of 3D-TLC SSD is from 0.26% to 2.31%. However, we notice that drives with NEW NAND architecture (i.e., II-E family) exhibit around 1.61× and 1.87× higher average ARR than those of MLC and 3D-TLC drives, respectively (the p-values are equal to 2.065e-02 and 4.351e-03). Table 3 further demonstrates an ARR breakdown of failure symptoms among different NAND chips. For NEW-based SSD, the main culprits of its high ARR are the I/O and booting failures.

4 The Fail-stop

Now, we present the three major changes in failure patterns in NVMe SSD. For each aspect, we start with existing patterns in SATA/SAS SSD. We then study the differences in NVMe SSD using a similar setup and further verify our findings under a series of controlled-variable experiments.

4.1 Infant Mortality

Finding 1. *Infant mortality, a notorious failure trend in hardware early deployment period, is not notable in NVMe SSD.*

Existing patterns. The Bathtub Curve [40] is a classic depiction of hardware failure variances through time. Generally, there are three main phases: infant mortality, stable (aka. useful-life) and wear-out period. Previous SATA SSD studies suggest that flash drive also follows this trend [35] (i.e., with an early detection phase followed by the bathtub curve).

Difference in NVMe SSD. For NVMe SSD, we are interested in whether such observation still holds. Here, we adopt the monthly failure conditional probability (FCP) to demonstrate the failure trend (i.e., the same metric as in Section 5.1 of [34]). The FCP is calculated as the number of drives to be replaced that month divided by the number of drives surviving that month. In Figure 1, we present a comparison between the six most popular drive families covering varying NAND architectures (i.e., 15-20nm MLC, 32-64 layers 3D-TLC, and the NEW). Throughout the paper, error bars refer to 95% confidence intervals with bootstrap methods [12] (2,000 iterations). In Figure 1, to calculate the error bar of FCP in month x (with N drives surviving month x-1), we create 2,000 random samples of FCPs; in each sample, the FCP is calculated based on a randomly-chosen set of N drives (i.e., sampling with replacement). Finally, we calculate the 95% confidence interval based on these 2,000 samples of FCPs as the corresponding error bar.

Should the NVMe SSD still follow the bathtub curve, we shall see the FCP, starting from a high value (i.e., the infant mortality), quickly decreases to a stable range (i.e., the useful life) until surging in the wear-out period. However, from visual inspection, we discover that most drive families do not have outstanding infant mortality during early periods. We further calculate the average FCP under various periods (e.g., 1st to 3rd month and the most recent three months).

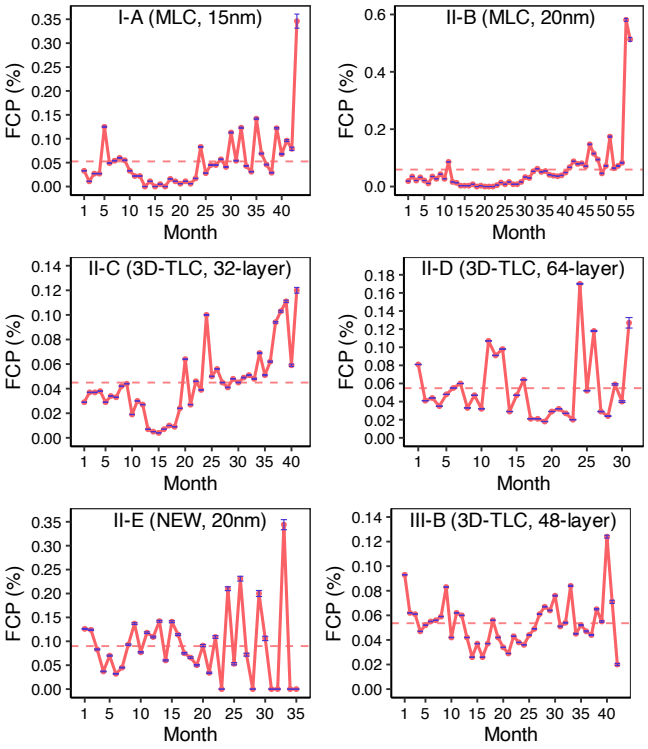

Figure 1: Failure trend (§4.1). *The figures present the Failure Conditional Probability (%) vs. drive's deployed time in months in 6 drive families. The dashed line indicates the average value within each figure. Error bars throughout the paper refer to 95% bootstrap confidence intervals [12] (2,000 iterations).*

As a result, the early period (i.e., the first three months) has equivalent or even lower FCP than the later periods. For example, in I-A, the first three months yield an average FCP of 0.02%, whereas the average FCPs of the next 9 months (months 4-12) and most recent three months are 0.05% and 0.35%, respectively.

Validity analysis. Next, we explore the reason behind this reliability improvement. First, it is unlikely that the stress tests weed out the faulty drives before deployment. Stress tests are usually short (up to one week) and thus not enough for drives suffering infant mortality (the period lasts for 12-15 months long [35]). Second, we exclude the possibility of external impacts (e.g., uneven distribution of workloads and drive age, if any) using two-sided t-tests. Then, we focus on the internal aspect by studying the SMART attributes variances over time. Here, for both II-D and III-B in Figure 2, nearly all health-related metrics experience the infant mortality as they start with a much higher value and then decrease to a stable range over time. Other drive families, in general, also follow this trend. Note that the values in Figure 2 are normalized to the lowest point on each curve and reported in logarithmic reporting. This indicates that NVMe SSD still

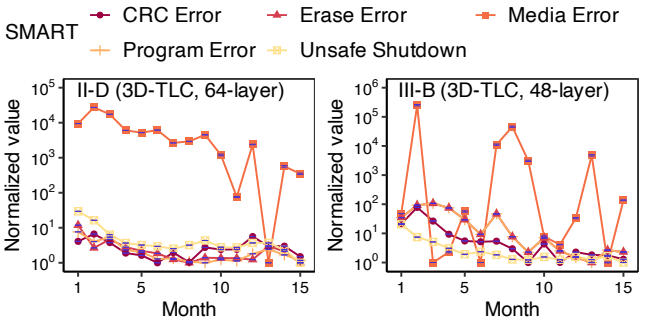

Figure 2: SMART in early age (§4.1). *The figures present the average number of SMART-recorded errors per month during the first 15 months of deployment in two major drive models. Note that the numbers are normalized to the lowest point on each curve.*

accumulates a large number of errors during the early period. Therefore, we assume it is likely that the improvement of FTL error handling makes the NVMe SSD more resilient in the early periods.

Operational advice. We believe the recent advancement in failure handling has set the NVMe SSDs free from suffering infant mortality. This can serve as a relief signal for the supply chain and the on-site administrators as previous practice usually demands that the cloud operators stockpile extra pieces before initial deployment.

4.2 WAF

Finding 2. *NVMe SSD becomes more robust to high write amplification (WAF>2), but extremely low write amplification (WAF≤1) is still rare-but-deadly.*

Existing patterns. Write amplification is a common phenomenon in SSD I/O where the logical writes incur extra data to be written to NAND due to SSD internal operations (e.g., garbage collection and alignment). A higher write amplification factor (i.e., NAND writes size divided by logical writes size) therefore indicates a more random and small-writes-dominant workload. To overcome this disadvantage, manufacturers often use write compression techniques to combine small or buffer repeated writes [9, 46, 49].

Previously, a large-scale SATA SSD failure study by Microsoft pointed out that higher WAF (WAF>2) incurs more SSD failures (i.e., Section 3.5.1 of [36]). Moreover, they suggest that the write compression technique can be damaging where drives with less-than-one WAF have failure rates similar to those with a higher-than-two WAF (see Figure 11 in [36]).

Difference in NVMe SSD. To avoid bias led by model characteristics, we conduct a comparative study within each model family. For each drive family, we first place SSDs into different buckets by WAF with a step of 0.5. Then for each bucket, we calculate its corresponding ARR. Since the 95^{th} percentile of WAF in our entire fleet is around 4, the last bucket includes

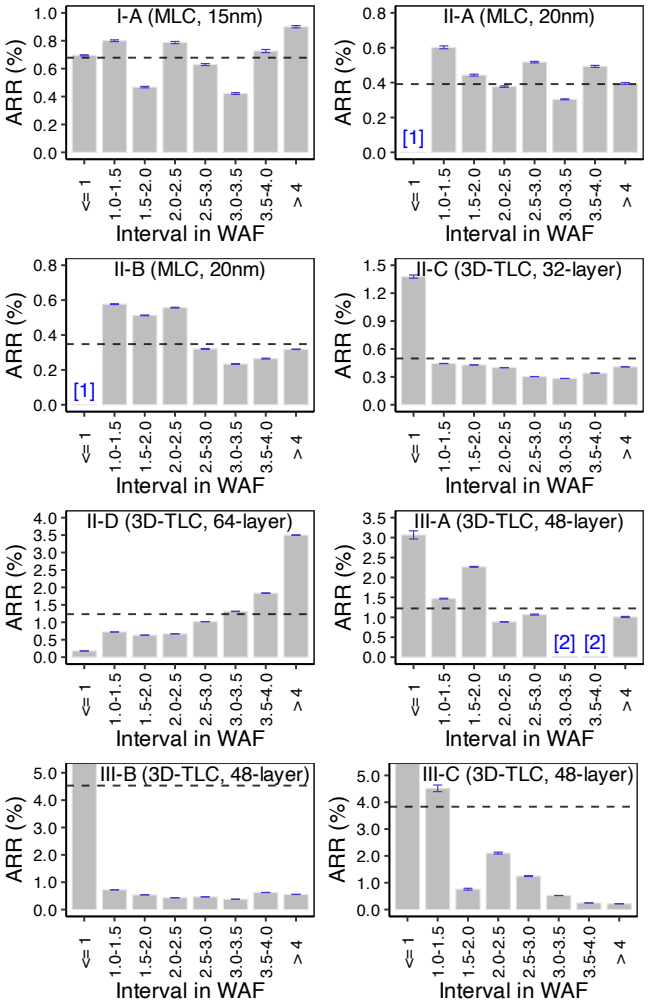

Figure 3: ARR under different WAF levels (§4.2). *The grey dashed line indicates the average value within each figure. "[1]": the bucket includes no drives; "[2]": the calculated ARR is zero (i.e., no failed drives). The first buckets of III-B and III-C are overflowed (>5%).*

drives with WAF above 4. The WAF≤1 bucket contains drives significantly influenced by the write compression technique.

Figure 3 presents the correlation between WAF and failure rates among eight popular drive families, covering different types of NAND and manufacturers. Here, we make two observations. First, for WAF higher than one, we do not observe a strong positive correlation between WAF and ARR in most drive families (the II-D drive family is considered an exception). A set of Spearman's Rank Correlation Coefficient tests [44] further statistically confirm our hypothesis. This indicates that NVMe SSD is less affected by random small writes (a major cause for high write amplification). Second, we observe that, for drives with low WAF (i.e., WAF≤1), their failure rates are still relatively high. On average, these low-WAF drives can have a 2.19× higher ARR rate than average.

Time	Type	SATA SSD	NVMe SSD	Hypo.
Total	node	4.5-73.7%	70.6-96.6%	0.04-9.0%
	rack	28.6-91.4%	79.4-97.6%	27.8-77.4%
(0, 1min]	node	0.8-24.7%	1.1-17.9%	0%
	rack	1.7-27.2%	1.3-17.9%	0%
(1d, 1mon]	node	1.1-39.4%	14.3-57.5%	0.01-1.1%
	rack	6.5-47.9%	15.5-57.2%	2.9-10.0%

Table 4: Intra-node/rack failure distributions across drive types (§4.3). *The table presents the relative percentage of intra-node and intra-rack failures in SATA SSD from a previous study [19], our NVMe SSD fleet, and a hypothetical setting where failures are independent of time and location.*

An extreme example is the III-B drive family (lower left in Figure 3), where low-WAF drives are 6.18× more likely to fail than average. Fortunately, we discover that these low-WAF drives usually occupy only a small proportion (e.g., only 0.09% in III-B). Even for I-A-3840 (i.e., having an average WAF of 0.84), we argue that low-WAF drives can be easily singled out with simple SMART attributes calculation for close monitoring or reallocation.

Validity analysis. Many factors (e.g., workloads, age, wear, and drive model) can influence both the WAF and reliability. To verify our finding, we further conduct a series of experiments by controlling the above variables (not shown due to space limitations). Our evaluation further confirms that none of the factors influences our finding. Therefore, we conclude that, in NVMe SSD, while the low WAF may still be deadly, the high WAF is no longer concerning.

4.3 Intra-node/rack Failures

Finding 3. *Spatially correlated (intra-node/rack) NVMe SSD failures are temporally correlated in the long-term span (i.e., 1 day to 1 month), but no longer prevalent in the short span.*

Existing patterns. Correlated drive failures are notorious for their cascading impact on the reliability of the entire distributed system (e.g., reduced redundancy) [2, 15]. According to a previous study in Alibaba (i.e., Finding 5 of [19]), a non-negligible proportion (i.e., up to 34.3% and 44.2%) of SATA SSD spatial-correlated (intra-node/rack) failures can also be temporal-correlated within a short span (i.e., at most one minute apart), posing a critical challenge to the overall stability.

Difference in NVMe SSD. To study whether such a correlated pattern still plagues the NVMe SSD fleet, we further check the intra-node/rack failure time intervals in our datasets. Here, to be consistent with the previous study [19], we reuse the Relative Percentage of Failures (RPF) to calculate the likelihood of correlated failures. In RPF, the numerator is the number of the sets of failures that occur between a specific period (e.g., 0 to 1 minute). The denominator is the sum of all failures of a particular drive model. Note that, in RPF, the same failure can be counted repeatedly as a member of

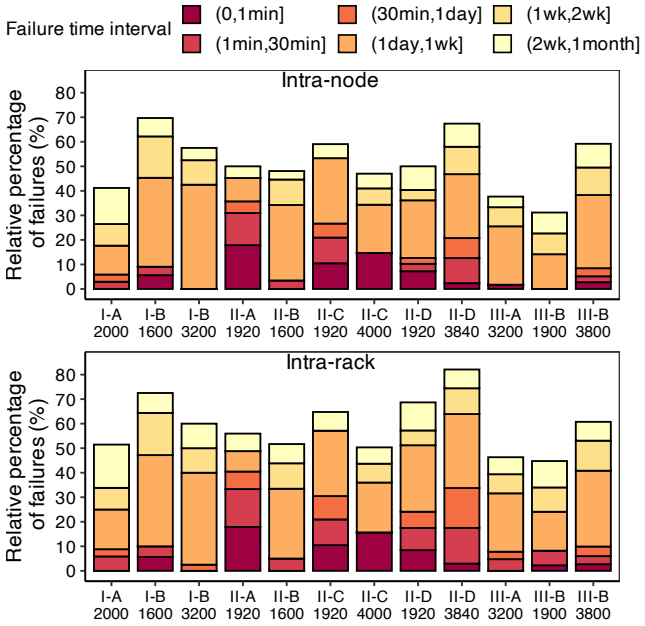

Figure 4: Intra-node/rack failure distributions across drive models (§4.3). *The figures present relative percentage of intra-node and intra-rack failures for each drive model. Note that stacked bars can reach above 100% because failures could be counted multiple times into different buckets.*

different correlated failures sets. For example, consider three failures, say A, B, and C, all occur within one minute inside the same node. Then there are three sets of correlated failures (i.e., [A,B], [B,C], and [A,C]), thus yielding an RPF of 100%.

In Table 4, we list the RPF from the previous study (i.e., SATA SSD column) and ours (i.e., NVMe SSD column). We make the following observations. First, the accumulated RPFs (i.e., Total row) across all NVMe drive models are substantially higher, with an increase up to 14.69× and 1.78× for intra-node and intra-rack scenarios, respectively. The corresponding t-tests return p-values equal to 6.322e-06 and 1.881e-04.

Second, unlike SATA SSD (mostly correlated during short intervals), correlated failures in NVMe SSD are commonly observed only in long intervals (i.e., 1 day to 1 month). Here, we use Figure 4 to demonstrate further the distribution of correlated failure intervals among different models on a weekly breakdown of the long intervals (i.e., 1 day to 1 month). In Figure 4, the upper graph shows the distribution of intra-node intervals, and the lower graph shows the intra-rack ones. We denote each interval with a different color (e.g., darkest for the shortest interval). Figure 4 shows the prevalence of long intervals (i.e., the three lightest/top boxes in each bar) in correlated failures. Conversely, short and medium ones (i.e., the three darkest/bottom boxes) on average occupy less than 17.86% (within 1 minute), 15.48% (1-30 min), and 16.25% (30 min to 1 day) of the total RPF.

Validity analysis. Each rack in our setup hosts hundreds of drives. Under such a considerable number, uniformly distributed failures may also co-occur within a rack. To verify that intra-node/rack failures are indeed a result of non-uniformity, we perform a set of hypothetical experiments where drive failures are redistributed to be uncorrelated (i.e., independent of location and arrival time). First, for each drive model, we sample without replacement to get a new batch of drives and mark them as "failed" drives. Second, we assign a random timestamp from 2019-11-04 to 2020-11-02 to each "failed" drive as its failure time (see Table 1). For a fair analysis, we repeat the above procedures 2,000 times and calculate the average RPFs of intra-node/rack failures for each drive model.

In Table 4, we compare the intra-node/rack failure distributions of hypothetical experiments (i.e., Hypo. column) with those of the original setting (i.e., NVMe SSD column). Our observations are as follows. First, intra-node failures under the hypothetical setting are nearly negligible. For example, the accumulated RPFs (i.e., Total row) of intra-node failures are only 0.04-9.0% under the hypothetical setting, whereas those from the original setting (70.6-96.6%) are much smaller (the p-value is less than 2.2e-16). Second, even though intra-rack failures are non-negligible under the hypothetical setting, their RPFs are consistently smaller than those from the original setting, e.g., with accumulated RPFs of 27.8-77.4% vs. 79.4-97.6% (the p-value is equal to 1.119e-08). Therefore, the non-uniformity in intra-node/rack failures is significant in our dataset in contrast with the hypothetical setting.

Operational advice. While the decline of closely correlated failures implies a lower risk of experiencing system-wise failures, the surging of long-interval correlated failures still poses a pressing threat. An inconvenient fact is that fixing drive failure usually starts with software-based approaches (e.g., data scrubbing and fsck), and such online checking and repairing takes time [16, 33, 47]. In fact, we discover that 43.90%, 14.36%, and 10.90% of the failed drives in our clusters are repaired after one day, one week, and two weeks. Based on our finding, we have refined our operational process by directly putting drives offline upon failures to reduce the chances of suffering long-term correlated failures.

5 The Fail-slow

Apart from the common fail-stop failures, we are also interested in fail-slow failures where drives exhibit performance much less than expected (e.g., considerably high latency under normal traffic). We hypothesize that the ultra-low latency nature of NVMe SSD would make the drive more *susceptible* to fail-slow failures. To verify our assumption, we conducted an extensive study based on the per drive `iostat` traces from more than half a million NVMe SSDs and more than 4 million HDDs during four-month monitoring.

Note that our storage system requires all replicas (three replicas in most cases) ACKed before any write request is

Model	Lith. /Layer /Type	Slow Drive (%)	Event Freq.	Dur. (min)	Event Laten. (us)	Slow-down Ratio	Slow Drive (%)	Event Freq.	Dur. (min)	Event Laten. (us)	Slow-down Ratio
		5min					15min				
I-A-2000	15nm	4.44%	225.06	18.98	195.60	2.39	3.65%	118.01	38.44	200.20	2.39
II-A-1920	20nm	1.25%	24.22	8.60	152.77	1.94	0.57%	8.70	20.36	148.96	1.80
II-C-1920	32L	0.52%	23.50	19.31	263.31	2.19	0.37%	11.84	39.67	256.58	2.19
II-C-4000	32L	0.06%	1.59	8.41	180.67	2.04	0.05%	0.66	21.16	175.62	2.03
II-D-1920	64L	0.17%	4.52	22.00	34.98	2.30	0.12%	2.30	42.59	36.22	2.35
II-D-3840	64L	0.48%	14.08	12.00	152.63	5.87	0.31%	5.99	27.08	122.35	4.48
III-B-1900	48L	3.04%	46.75	9.43	54.67	2.19	1.55%	12.82	24.69	56.68	2.22
III-B-3800	48L	1.31%	44.28	13.51	360.59	6.33	1.05%	20.89	30.39	244.74	4.29
Average	-	1.41%	48.00	14.03	174.40	3.16	0.96%	22.65	30.55	155.17	2.72
H1	CMR	0.32%	4.53	8.56	47370.67	2.18	0.09%	1.15	23.97	55120.74	2.33
H2	CMR	0.24%	2.30	6.92	12355.12	2.04	0.03%	0.32	21.20	14796.03	2.38
H3	CMR	0.04%	0.51	11.72	1962.49	3.01	0.03%	0.36	28.49	2041.27	3.39
Average	-	0.20%	2.45	9.07	20562.76	2.41	0.05%	0.61	24.55	23986.01	2.70
		30min					60min				
I-A-2000	15nm	3.19%	70.46	60.50	207.18	2.38	2.49%	32.95	97.33	203.25	2.24
II-A-1920	20nm	0.45%	3.03	34.53	143.81	1.81	0.11%	0.38	60.25	147.45	1.79
II-C-1920	32L	0.36%	7.63	60.82	263.97	2.18	0.32%	3.61	100.04	272.49	2.14
II-C-4000	32L	0.03%	0.28	39.85	176.39	2.03	0.01%	0.03	97.88	182.67	2.20
II-D-1920	64L	0.08%	1.26	61.72	37.57	2.40	0.04%	0.52	92.95	39.62	2.49
II-D-3840	64L	0.23%	2.60	47.63	132.95	4.31	0.13%	0.85	86.96	128.33	3.62
III-B-1900	48L	0.75%	4.74	48.27	61.94	2.35	0.40%	1.96	86.63	74.55	2.70
III-B-3800	48L	0.91%	11.18	49.53	248.33	4.18	0.63%	4.14	84.23	148.26	2.32
Average	-	0.75%	12.65	50.36	159.02	2.71	0.52%	5.56	88.28	149.58	2.44
H1	CMR	0.04%	0.41	44.79	58673.31	2.43	0.02%	0.11	83.36	62272.65	2.50
H2	CMR	0.01%	0.10	39.86	15496.51	2.55	<0.01%	0.02	98.31	20991.78	3.66
H3	CMR	0.03%	0.21	53.05	2937.19	3.66	0.02%	0.11	96.27	3187.88	5.09
Average	-	0.03%	0.24	45.90	25702.34	2.88	0.01%	0.08	92.65	28817.44	3.75

Table 5: Baseline statistics for fail-slow (§5.1-§5.2). *The table shows the summarized statistics of fail-slow occurrences under the 5-min to 60-min duration requirements.* ***Lith./Layer/Type****: lithography, numbers of stacking layers or HDD type;* ***Event Freq.****: fail-slow event frequency per 1K drives per hour;* ***Dur.****.: average event duration in minutes;* ***Event Laten.****.: average event latency in µs. Note that the SSDs and HDDs under heavy traffic are not included (see §5.1).*

returned, while only one ACKed for each read request. Thus, fail-slow failures are more likely to impact the write performance than the read. Such a phenomenon agrees with most known fail-slow cases in practice. Throughout this section, we focus on the write latency where fail-slow failure can be more destructive.

5.1 Identifying Fail-slow Events and Drives

Currently, a subset of our clusters is equipped with daemons to monitor the `iostat` of the deployed drives. Due to capacity limits and performance concerns, the daemon runs 3 hours (9 P.M. to 12 A.M.) each day. It only records the average statistics of each monitoring window (15 seconds in the current setup), thereby yielding 720 records per drive (3 hours divided by 15 seconds) each day.

Methodology overview. We use the following threshold-based approach to identify fail-slow drives (similar to a previous study on SATA SSD and HDD tail latency [21]). The first step is to select suspicious drives with high latencies. Then, we determine whether the chosen drives are indeed fail-slow by checking the existence of consistent slowdowns.

Identifying suspicious fail-slow drives. In the first step, we observe that the performance (e.g., latency, IOPS, and throughput) records within a cluster generally follow a Positively Skewed Distribution. For example, in one cluster, the median latency is only $49.19\mu s$ while the average latency is $667.85\mu s$. Thus, we can use a latency threshold as $(-\infty, 3rd\_quartile + 2IQR)$ to identify the outliers (i.e., slow drives) [39] where the IQR (interquartile range) is computed by subtracting the first quartile from the third quartile.

If the 3-hour median latency of a drive is beyond the bar, we mark this drive as a suspicious slow drive. To avoid reporting led by heavy traffic (i.e., false-positive), we also rule out high-latency drives under heavy traffic (i.e., IOPS/throughput is also beyond $3rd\_quartile + 2IQR$).

Identifying slowdown events. Note that a suspicious drive may be marked due to transient but time-consuming events (e.g., read retries, unstable connection). Therefore, we further

check whether a suspicious drive has experienced a consistent slowdown event to pinpoint the fail-slow drives. Here, we borrow the idea of measuring slowdown events from a previous SSD performance study [21].

First, we mark the 3-hour `iostat` latency records from the suspicious slow drive and its 11 intra-node peers (12 drives per node) as L_i^k, representing the record i ($i \in \{1,2,...,720\}$) from drive k ($k \in \{1,2,...,12\}$) . Then, we use RL_i^k (Relative Latency) to indicate the slowdown degree of drive k at record i, formally $RL_i^k = \frac{L_i^k}{median(L_i^1, L_i^2, ..., L_i^{12})}$. Then, we formulate an event as $E_{i,j}^k$ by computing the means of RLs of drive k from record i to j, formally $E_{i,j}^k = mean(RL_i^k, RL_{i+1}^k, ..., RL_j^k)$.

For an $E_{i,j}^k$ to be considered a fail-slow event, it must satisfy two requirements. First, $E_{i,j}^k$ needs to be larger than an empirical slowdown degree. We set the slowdown degree as 2 (same in [21]), meaning that, during the event, the victim drive is at least twice slower than its peers. Second, we set four minimum spans as 5, 15, 30 and 60 minutes, meaning the $E_{i,j}^k$ should last longer than 20, 60, 120 or 240 records (a record spans 15 seconds).

To sum up, a drive (i.e., NVMe SSD or HDD) is deemed fail-slow if and only if it has a high median latency (i.e., higher than top 0.04% latency variances in the cluster during 3-hour monitoring) and suffers at least one fail-slow event.

5.2 Dataset and High Level Observations

5.2.1 Dataset Overview

In total, we have identified around 5K and 3K fail-slow NVMe SSDs and HDDs, respectively. Table 5 gives an overview of the fail-slow drives and events among the fleets. We use four quadrants to represent the statistics under different requirements (i.e., 5 to 60 min). The upper half of each quadrant includes 8 major NVMe SSD models (named as brand-model-capacity), while the lower includes the three most popular HDD models (H1, H2, and H3) in our clusters.

For each column, we begin by listing the lithography (for planar NAND), layers (for 3D-NAND), or the type (for HDD, i.e., CMR or SMR). Further, we show the percentage of drives that have been identified as fail-slow ones in that model (Slow Drive%). The Event Freq. describes the numbers of events per 1000 drives per hour, reflecting the fail-slow severity in a mid-sized cluster. The following two columns (Duration and Event Latency) show the average fail-slow event duration and average event latency. The final column (Slowdown Ratio) is the ratio of average event latency to average latency of peer drives (i.e., healthy drives from the same node) during the event. The last row of each sub-quadrant is the average value for each category (i.e., SSD or HDD).

5.2.2 SSD vs. HDD

Finding 4. *Compared to HDD, fail-slow failure in NVMe SSD is much more widespread and frequent, and can degrade the drive to SATA SSD or even HDD level performance.*

We start with the differences between NVMe SSDs and HDDs. First, we observe the disparity between HDD and SSD in slow drive popularity (i.e., Slow Drive %). Comparing the average row in each quadrant reveals that slow drives are 6.05× (i.e., 1.41% to 0.20% in the 5-min quadrant) to 51× (0.52% to 0.01% in the 60-min quadrant) more common in SSDs. Similarly, we also observe that the fail-slow occurrences (i.e., Event Freq.) are much more frequent in SSDs, ranging from 18.59× (5-min) to 68.50× (60-min).

Regarding the event duration, the difference varies. On average, the SSD event lasts up to 55% longer in the first three quadrants, but the trend reverses as the HDD event costs 5% more time in the 60-min quadrant. This indicates that fail-slow events in NVMe SSD are relatively short-termed.

Moreover, while models like II-D-1920 and III-B-1900 still deliver relatively satisfying performance, fail-slow NVMe SSDs usually degrade to SATA-SSD-level latency (i.e., having an average event latency around 160μs, see average rows of event latency from 5-min to 60-min). Even worse, our evaluation shows that the top 1% slowest events in several NVMe SSD models deteriorate to an average latency of around 22ms, an unsatisfying performance even for HDD.

The comparisons of slow drive popularity, event frequency, and performance prove that the NVMe SSD is indeed widely plagued (1.41% affected under 5-min requirement) and severely impacted ($\sim$ 160μs average event latency) by fail-slow failure. Recall that the dataset comes from only three hours of monitoring per day for four months. Plus, all of our models are enterprise-level, and we have already excluded SSDs under heavy traffic. Therefore, we expect the annual fail-slow drive rate to be higher and fail-slow occurrences more frequent in the field.

Operational advice. Experiencing widespread and severe fail-slow faults can be particularly harmful to NVMe SSDs as performance-sensitive jobs are usually placed on them. However, simply putting all fail-slow drives offline can be unacceptably expensive. Recently, we have been experimenting with a "three strikes" approach to tackle the suspiciously slow drives. Specifically, the first time a drive is diagnosed with fail-slow failure, we would clean the drive's data and deploy it again as a new drive. In the second time, we would fully flush the drive with zeroes, reformat and redeploy it. A third timer would be directly put offline for replacement. Unfortunately, we have just deployed this strategy and do not have enough samples for analysis.

Root cause. We have sent 100 slowest SSDs (around the top 2% of the identified slow drives with an average event latency of 4.4 ms) back to vendors for repair. The results show that 33 of them have bad capacitors, causing the malfunctioning buffer and thus the high latency. 46 of them contain bad chips and the root causes of the rest remain unclear.

5.2.3 Differences between SSD models.

Finding 5. *The manufacturer is a dominant factor of fail-slow*

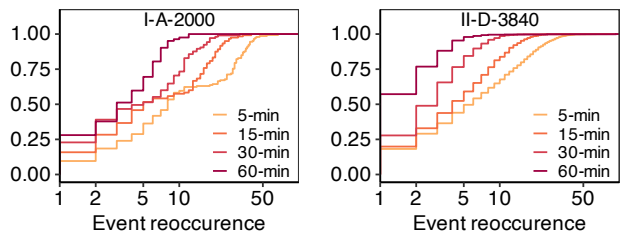

Figure 5: CDF of fail-slow event reoccurrence (§5.2). *The figures present the distribution of event reoccurrence for two models under four duration requirements.*

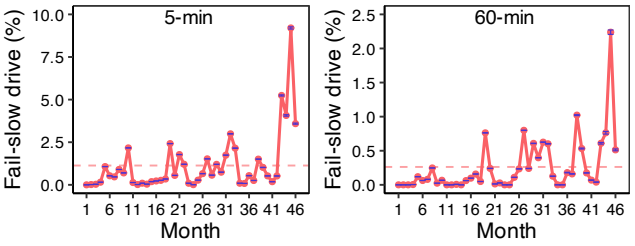

Figure 6: Fail-slow drive percentage across time (§5.3.1). *The figures show the percentage of fail-slow drives per month under the 5-min and 60-min duration requirements. The dashed line indicates the average value within each figure.*

drive population in NVMe SSD.

Now, we dig deeper by focusing on the fail-slow distribution differences between SSD models. First, we look at the influences of the manufacturers. Our dataset includes three manufacturers (i.e., I, II, and III). For slow drive percentage, there is a clear order (i.e., manufacturer I followed by III and II) across the four quadrants. Even the highest value of III (e.g., 3.04% of III-B-1900 in 5-min quadrant) is well behind that of the I's model (i.e., 4.44% of I-A-2000), which also applies to the comparison between III and II. However, we do not observe visible patterns for the event duration, event latency, and slowdown ratio.

Finding 6. *Higher fail-slow drive popularity does not always lead to a higher fail-slow event frequency.*

Moreover, we notice a seemingly counter-intuitive pattern. One may assume a higher fail-slow drive percentage leads to a higher event frequency. While this hypothesis holds in the longest duration requirement (60-min quadrant), we find many counter-examples among shorter ones (e.g., II-A-1920 and II-C-1920 in 5-min quadrant).

A possible explanation is that under a shorter duration requirement, there are more drives with multiple events, resulting in a small slow drive percentage with high event frequency. Here, we further verify this assumption by using Figure 5, a CDF of events per drive under different duration requirements. We can clearly see that drives under shorter durations accumulate more events than those under longer durations.

5.3 Correlating Factors

In this section, we conduct an extensive study on fail-slow failures versus various correlating factors (i.e., drive age, workload and SMART attributes).

5.3.1 Drive Age

Finding 7. *Fail-slow drive population and event frequency are strongly correlated with age, but only for old (power-on time > 41 months) NVMe SSDs.*

Age is widely known for its significant impact on SSD fail-stop failures [34, 35, 41]. We correlate fail-slow metrics with drive power-on time to see the significance of age here.

We first place all fail-slow drives into monthly buckets (e.g., bucket-1 includes fail-slow drives with power-on time between 0 to 1 month). Note that for a drive with multiple event occurrences (e.g., fail-slow events in both 34th and 35th months), we put the drive to the earliest bucket (i.e., 34th-month bucket). Then we calculate the fail-slow drive percentage for each bucket by dividing the numbers of fail-slow drives against the numbers of drives of the same age.

Figure 6 demonstrates the population variances along time under the 5-min (left) and the 60-min (right) requirements where the horizontal dashed line is the average. We can see that the population, in both scenarios, oscillates around the average value at first and then start to surge in the final months. Further, a Spearman's Rank Correlation Coefficient (SRCC) test [44] reveals that the fail-slow population in old drives (>41 months) is highly correlated with age. Specifically, in the 5-min requirement, the SRCC score for old drives is around 0.92 (way beyond the common threshold for positive correlation, i.e., 0.5). In contrast, the scores from the rest (i.e., younger drives) are close to 0, meaning no correlation. Similar trend exists under 15-min, 30-min, and 60-min requirements.

Next, we adopt similar approaches to measure the correlation between age and other metrics, including event frequency, event duration, and slowdown ratio. We find that the event frequency is similar to the fail-slow population where old drives (i.e., $>$ 41 months) are strongly correlated with age while young drives are not. We do not observe a notable correlation for duration and slowdown ratio, indicating that both metrics remain rather stable throughout the lifecycle.

5.3.2 Workload

Finding 8. *The workload can significantly affect various fail-slow characteristics, and heavy traffic workload may have long-lasting impacts on fail-slow occurrences.*

Workload is also a well-known impact factor on the SATA SSD fail-stop failures [1, 24, 36, 41, 47]. The key difference between workloads is the I/O pattern. Therefore, we evaluate the impacts of workloads by studying four representative cloud storage services with drastically different access patterns, namely block storage, buffering, object storage, and query.

G	Age-Wr	Wl.	Slow Drive (%)	Event Freq.	Dur. (min)	Slow-down Ratio
II-D-3840						
1	3rd-2nd	Block	0.02	0.23	9.81	1.99
		Buffer	39.17	1318.51	11.85	2.28
		Query	0.08	2.31	6.83	3.01
2	3rd-3rd	Block	0.01	1.84	19.60	2.15
		Buffer	13.86	466.00	13.38	2.22
III-B-3800						
3	2nd-1st	Block	0.03	0.65	15.29	152.59
		Object	5.86	1187.69	26.67	12.41
4	2nd-2nd	Block	0.01	0.15	7.04	2.06
		Buffer	36.88	1196.75	12.00	2.30
5	3rd-2nd	Block	0.71	12.76	10.09	64.91
		Buffer	10.18	608.78	20.24	2.39

Table 6: Fail-slow statistics for groups of workloads (§5.3.2). *The table presents fail-slow metrics under different workloads with control variables on the drive model, age, and P/E cycle (total bytes written divided by capacity) under 5-min duration requirement.* ***G****: variable-controlling group;* ***Age-Wr****: age-write bucket;* ***Wl.****: workload; "Slow Drive (%)" to "Slow-down Ratio" follow the same metrics in Table 5.*

As factors like age and manufacturers could severely influence the fail-slow failures, we thus conduct this study in a finer granularity by setting multiple variable-controlling groups. In Table 6, based on the fail-slow metrics under 5-min duration requirement, we group the drives (G column) by drive model, age, and P/E cycle (total bytes written divided by capacity). We choose two drive models, II-D-3840 (upper half) and III-B-3800 (lower half), as they are from different manufacturers and both popular among different services. Then, we control other variables as Age-Wr (age and P/E cycle). The age is listed by years and the P/E cycle is broken down into 4 intervals (i.e., $\leq$100, 100$\sim$500, 500$\sim$1K, and 1K$\sim$10K P/E cycles, respectively). For example, the 3rd-2nd from group 1 means the drives from this group have a deployment time between 2 to 3 years and a usage level between 100 to 500 P/E cycles. For each group, we further list the workload (Wl. column) and the corresponding fail-slow metrics ("Slow Drive (%)" to "Slow-down Ratio" column, same as Table 5).

Here, we make the following observations. First, by comparing the metrics within each group, we can see that the workload can significantly affect all four fail-slow metrics. For instance, in groups 1 and 2, the fail-slow population and event frequency of buffering workload can be thousands of times higher than those in block storage (e.g., 39.17% vs. 0.02% in group 1). Similar disparities in event duration and slowdown ratio can be observed between block and object storage in group 3, or between block storage and buffering in group 5.

Second, the patterns can preserve despite model, age, or P/E cycle variances. For example, by comparing groups 2 and 5, while the groups are of different models and usage levels, the buffering workload in both groups has a much higher slow drive percentage and event frequency.

To sum up, the above experiments verify the significant influences of workloads on fail-slow failure. Primarily, we discover that fail-slow failure favors the buffering workload the most. In practice, the drives under the buffering workload usually have constantly heavy traffic (e.g., storing intermediate results of big data workload). Recall that we have already excluded SSDs under heavy traffic from consideration. Therefore, a possible explanation is that the heavy traffic may have a long-lasting effect (e.g., leaving data more scattered), making the drive more susceptible to fail-slow failure.

5.3.3 SMART Attributes

Finding 9. *SMART attributes only exhibit negligible correlation with fail-slow metrics.*

Now, we analyze whether SMART attributes (an essential set of indicators for fail-stop failures) correlate with fail-slow failures. We collect SMART data on the last day of our four-month fail-slow detection period. Further, we divide drives into groups based on drive model, age, P/E cycle, and workload. Within each group, we label drives as either "slow" or "not-slow". Finally, we apply SRCC [44] to examine the correlation between fail-slow failures and SMART attributes.

Under the 5-min duration requirement, we obtain 40 groups of drives. None of them exhibit a clear correlation with any SMART attributes, such as Critical Warning, P/E Error, and CRC Error. The above results preserve under 15-min, 30-min and 60-min duration requirements. Even if we set drives with multiple fail-slow events as "slow", the results remain. Hence, we conclude that the root causes and/or the symptoms of fail-slow failures are not (well) captured by the SMART.

Operational advice. In this case, we decide not to integrate SMART attributes to improve the fail-slow detection. Currently, we have been exploring various approaches. The major hurdle is the lack of verified positive samples (i.e., fail-slow drives) due to the lack of a fail-slow oracle. Therefore, based on the performance, we tried classic statistical methods and discovered that basic linear or polynomial regression is not very practical as they require constant adjustment for the varying traffic (even within the same workload). We leave employing machine learning models as a part of our future work once the "three strikes" yields convincing and abundant cases. Also, we encourage manufacturers to reveal drive characteristics (e.g., flash GC timing) to facilitate fail-slow identification.

5.4 Transition to Failures

Finding 10. *The transition from fail-slow to fail-stop failures is rarely observed, at least not observed within a short time interval (within 5 months).*

Previous case studies indicate that a fail-slow failure may

	Not-replaced	Replaced	Total
Not-slow	98.84% (770965)	0.57% (4429)	99.41% (775394)
Slow	0.59% (4574)	<0.01% (10)	0.59% (4584)
Total	99.43% (775539)	0.57% (4439)	100% (779978)

Table 7: Transition from fail-slow to fail-stop failures (§5.4). *The table presents a contingency table of fail-slow and failed (later in time) drives for NVMe SSDs under the 5-min duration requirement.*

turn into a fail-stop failure [17]. Therefore, we collect up-to-date failure tickets ever since the beginning date of our detection period. The latest failure tickets are about 5 months older than the last recorded fail-slow event.

Table 7 is a sample contingency table recording the frequency counts of drives based on 2 categories: appearing in the failure tickets (Replaced column) or not (Not-replaced column) and having at least one fail-slow event (Slow row) or not (Not-slow row). The result is rather surprising as only 10 drives exhibit fail-slow failures before fail-stop failures, yielding a relatively small population in both slow (around 0.22%) and replaced (around 0.23%) drives. The mean and median transition time are 73 and 67 days, respectively. A possible reason is that fail-slow seldom or may need a long time to transit to a fail-stop failure. Therefore, we conclude the fail-slow failures are unlikely to transit to fail-stop failures, at least not within a few months.

6 Limitation

Environmental differences. The main methodology of this paper is to draw side-by-side comparisons with previous findings. The environmental differences (e.g., workload and hardware setup) may impact the validity of our findings. Therefore, throughout the study, we use controlled-variable experiments to rule out the biases led by such influences and only make findings when statistical confidence is enough. Plus, many previous studies share great similarities with our setups (i.e., cloud storage systems from [19, 35, 36, 47]) and workloads (e.g., object storage in [19, 36, 47] and database storage in [36, 47]). Hence, our findings are a result of the NVMe SSD characteristics instead of the environmental factors.

Comprehensiveness. Previous studies have covered various aspects of SATA SSD failures. Due to the space limit, we are unable to present all of them. In the paper, we do not discuss the topics due to three reasons: missing data sources (e.g., bus power consumption [35]), statistically unconvincing results (e.g., lithography [34]), and unchanged failure patterns.

Fail-slow detection. Unlike fail-stop failures where the oracle is clear (i.e., the five symptoms in Table 3), fail-slow failures are often difficult to pinpoint and thus rely on empirical thresholds. In the study, we place a rather strict threshold and drives under heavy traffic are not considered. The average event latency (close to SATA SSD performance), shown in Table 5, confirms the effectiveness of our detection approach. Even though we may underestimate the impacts of fail-slow occurrences due to the demanding standards, we believe our dataset and findings are sufficient to reveal a rather concerning status quo of fail-slow NVMe SSD in the field.

7 Related Work

SATA/SAS SSD failure study. There are several field studies of SSD failures in large datacenters, including NetApp [34], Google [4, 41], Alibaba [19, 47], Facebook [35], and Microsoft [36]. These studies share important insights regarding the trend, impacting factors, and correlation of the SATA/SAS SSD failures in the field. Our study distinguishes from them in two aspects. First, we focus on NVMe SSD, which can have distinctive failure characteristics due to internal (e.g., RAIN) and external (e.g., NVMe interface) changes. Second, apart from fail-stop failures, we also study the fail-slow failures, a pressing issue especially for the NVMe SSD.

Fail-slow failure study. Fail-slow failure (aka. gray failure) has attracted increasing attention from academia and industry [13, 17, 21, 22, 25, 26, 37]. Specifically, Gunawi et al. [17] collect more than 100 hardware fail-slow cases from various datacenters and perform qualitative analysis to understand the distribution and root causes behind the failures. Moreover, Hao et al. [21] reveal the distribution of tail latency in large-scale SSD/HDD-based RAID systems. Our work is different from the above as we focus on NVMe SSDs inside the general-purpose cloud storage system and perform large-scale quantitative analysis based on the monitoring data.

8 Conclusion

We perform a large-scale failure study of NVMe SSDs in the field. We have identified major changes of NVMe SSD fail-stop failure patterns including failures, robustness under WAF and the temporal correlation. Also, we investigate the fail-slow failures and the impact factors at scale. Altogether, we obtain 10 findings and open-source our dataset.

Acknowledgements

We would like to thank our shepherd and the anonymous reviewers for their insightful comments and suggestions. This research was supported by NSFC (62102424, 61872376, U1736207, 62072306), the Alibaba Innovation Research (AIR) program, National Key R&D Program of China (2018YFB2101102), Program of Hunan Postdoc Innovation (2021RC2069), and Program of Shanghai Academic Research Leader (20XD1402100). The authors thank Ryan Huang, Ennan Zhai, Shiming Wang, and Amber Bi for their feedbacks on early versions of this paper.

References

[1] Nitin Agrawal, Vijayan Prabhakaran, Ted Wobber, John D. Davis, Mark Manasse, and Rina Panigrahy. Design Tradeoffs for SSD Performance. In *Proceedings of the 2008 USENIX Annual Technical Conference (USENIX ATC)*, 2008.

[2] Ramnatthan Alagappan, Aishwarya Ganesan, Yuvraj Patel, Thanumalayan Sankaranarayana Pillai, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. Correlated Crash Vulnerabilities. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016.

[3] Ahmed Alquraan, Hatem Takruri, Mohammed Alfatafta, and Samer Al-Kiswany. An Analysis of Network-Partitioning Failures in Cloud Systems. In *Proceedings of the 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2018.

[4] Jacob Alter, Ji Xue, Alma Dimnaku, and Evgenia Smirni. SSD Failures in the Field: Symptoms, Causes, and Prediction Models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, 2019.

[5] Lakshmi N. Bairavasundaram, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, Garth R. Goodson, and Bianca Schroeder. An Analysis of Data Corruption in the Storage Stack. In *Proceedings of the 6th USENIX Conference on File and Storage Technologies (FAST)*, 2008.

[6] Lakshmi N. Bairavasundaram, Garth R. Goodson, Shankar Pasupathy, and Jiri Schindler. An Analysis of Latent Sector Errors in Disk Drives. In *Proceedings of the 2007 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, 2007.

[7] Lakshmi N. Bairavasundaram, Meenali Rungta, Nitin Agrawa, Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, and Michael M. Swift. Analyzing the Effects of Disk-pointer Corruption. In *Proceedings of the 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2008.

[8] Matias Bjørling, Abutalib Aghayev, Hans Holmberg, Aravind Ramesh, Damien Le Moal, Gregory R. Ganger, and George Amvrosiadis. ZNS: Avoiding the Block Interface Tax for Flash-based SSDs. In *Proceedings of the 2021 USENIX Annual Technical Conference (USENIX ATC)*, 2021.

[9] Feng Chen, Tian Luo, and Xiaodong Zhang. CAFTL: A Content-Aware Flash Translation Layer Enhancing the Lifespan of Flash Memory based Solid State Drives. In *Proceedings of the 9th USENIX Conference on File and Storage Technologies (FAST)*, 2011.

[10] Brian Choi, Randal Burns, and Peng Huang. Understanding and Dealing with Hard Faults in Persistent Memory Systems. In *Proceedings of the 16th European Conference on Computer Systems (EuroSys)*, 2021.

[11] Kingston Technology Corporation. SMART Attribute Details. https://media.kingston.com/support/downloads/MKP_306_SMART_attribute.pdf, 2015.

[12] Thomas DiCiccio and Bradley Efron. Bootstrap confidence intervals. *Statistical science*, 1996.

[13] Thanh Do, Mingzhe Hao, Tanakorn Leesatapornwongsa, Tiratat Patana-anake, and Haryadi S. Gunawi. Limplock: Understanding the Impact of Limpware on Scale-out Cloud Systems. In *Proceedings of the 4th Annual Symposium on Cloud Computing (SoCC)*, 2013.

[14] Daniel Ford, François Labelle, Florentina I. Popovici, Murray Stokely, Van-Anh Truong, Luiz Barroso, Carrie Grimes, and Sean Quinlan. Availability in Globally Distributed Storage Systems. In *Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2010.

[15] Aishwarya Ganesan, Ramnatthan Alagappan, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. Redundancy Does Not Imply Fault Tolerance: Analysis of Distributed Storage Reactions to Single Errors and Corruptions. In *Proceedings of the 15th USENIX Conference on File and Storage Technologies (FAST)*, 2017.

[16] Om Rameshwar Gatla, Muhammad Hameed, Mai Zheng, Viacheslav Dubeyko, Adam Manzanares, Filip Blagojević, Cyril Guyot, and Robert Mateescu. Towards Robust File System Checkers. In *Proceedings of the 16th USENIX Conference on File and Storage Technologies (FAST)*, 2018.

[17] Haryadi S. Gunawi, Riza O. Suminto, Russell Sears, Casey Golliher, Swaminathan Sundararaman, Xing Lin, Tim Emami, Weiguang Sheng, Nematollah Bidokhti, Caitie McCaffrey, Gary Grider, Parks M. Fields, Kevin Harms, Robert B. Ross, Andree Jacobson, Robert Ricci, Kirk Webb, Peter Alvaro, H. Birali Runesha, Mingzhe Hao, and Huaicheng Li. Fail-Slow at Scale: Evidence of Hardware Performance Faults in Large Production Systems. In *Proceedings of the 16th USENIX Conference on File and Storage Technologies (FAST)*, 2018.

[18] Kyuhwa Han, Hyunho Gwak, Dongkun Shin, and Jooyoung Hwang. ZNS+: Advanced Zoned Namespace Interface for Supporting In-Storage Zone Compaction. In *Proceedings of the 15th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2021.

[19] Shujie Han, Patrick P. C. Lee, Fan Xu, Yi Liu, Cheng He, and Jiongzhou Liu. An In-Depth Study of Correlated

Failures in Production SSD-Based Data Centers. In *Proceedings of the 19th USENIX Conference on File and Storage Technologies (FAST)*, 2021.

[20] Jonmichael Hands. How SSDs Fail – NVMe™ SSD Management, Error Reporting, and Logging Capabilities. https://nvmexpress.org/how-ssds-fail-nvme-ssd-management-error-reporting-and-logging-capabilities/, 2020.

[21] Mingzhe Hao, Gokul Soundararajan, Deepak Kenchammana-Hosekote, Andrew A. Chien, and Haryadi S. Gunawi. The Tail at Store: A Revelation from Millions of Hours of Disk and SSD Deployments. In *Proceedings of the 14th USENIX Conference on File and Storage Technologies (FAST)*, 2016.

[22] Mingzhe Hao, Levent Toksoz, Nanqinqin Li, Edward Edberg Halim, Henry Hoffmann, and Haryadi S. Gunawi. LinnOS: Predictability on Unpredictable Flash Storage with a Light Neural Network. In *Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2020.

[23] Bryan Harris and Nihat Altiparmak. Ultra-Low Latency SSDs' Impact on Overall Energy Efficiency. In *Proceedings of the 12th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage)*, 2020.

[24] Jun He, Sudarsun Kannan, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. The Unwritten Contract of Solid State Drives. In *Proceedings of the 12th European Conference on Computer Systems (EuroSys)*, 2017.

[25] Peng Huang, Chuanxiong Guo, Jacob R. Lorch, Lidong Zhou, and Yingnong Dang. Capturing and Enhancing In Situ System Observability for Failure Detection. In *Proceedings of the 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2018.

[26] Peng Huang, Chuanxiong Guo, Lidong Zhou, Jacob R. Lorch, Yingnong Dang, Murali Chintalapati, and Randolph Yao. Gray Failure: The Achilles' Heel of Cloud-Scale Systems. In *Proceedings of the 16th Workshop on Hot Topics in Operating Systems (HotOS)*, 2017.

[27] Sungjoon Koh, Changrim Lee, Miryeong Kwon, and Myoungsoo Jung. Exploring System Challenges of Ultra-Low Latency Solid State Drives. In *Proceedings of the 10th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage)*, 2018.

[28] Gyusun Lee, Seokha Shin, Wonsuk Song, Tae Jun Ham, Jae W. Lee, and Jinkyu Jeong. Asynchronous I/O Stack: A Low-latency Kernel I/O Stack for Ultra-Low Latency SSDs. In *Proceedings of the 2019 USENIX Annual Technical Conference (USENIX ATC)*, 2019.

[29] Xiaojian Liao, Youyou Lu, Erci Xu, and Jiwu Shu. Write Dependency Disentanglement with HORAE. In *Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2020.

[30] Xiaojian Liao, Youyou Lu, Erci Xu, and Jiwu Shu. Max: A Multicore-Accelerated File System for Flash Storage. In *Proceedings of the 2021 USENIX Annual Technical Conference (USENIX ATC)*, 2021.

[31] Xiaojian Liao, Youyou Lu, Zhe Yang, and Jiwu Shu. Crash Consistent Non-Volatile Memory Express. In *Proceedins of the 28th ACM Symposium on Operating Systems Principles (SOSP)*, 2021.

[32] Yixin Luo, Saugata Ghose, Yu Cai, Erich F. Haratsch, and Onur Mutlu. HeatWatch: Improving 3D NAND Flash Memory Device Reliability by Exploiting Self-Recovery and Temperature Awareness. In *Proceedings of the 24th IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2018.

[33] Ao Ma, Chris Dragga, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. ffsck: The Fast File System Checker. In *Proceedings of the 11th USENIX Conference on File and Storage Technologies (FAST)*, 2013.

[34] Stathis Maneas, Kaveh Mahdaviani, Tim Emami, and Bianca Schroeder. A Study of SSD Reliability in Large Scale Enterprise Storage Deployments. In *Proceedings of the 18th USENIX Conference on File and Storage Technologies (FAST)*, 2020.

[35] Justin Meza, Qiang Wu, Sanjev Kumar, and Onur Mutlu. A Large-Scale Study of Flash Memory Failures in the Field. In *Proceedings of the 2015 ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, 2015.

[36] Iyswarya Narayanan, Di Wang, Myeongjae Jeon, Bikash Sharma, Laura Caulfield, Anand Sivasubramaniam, Ben Cutler, Jie Liu, Badriddine Khessib, and Kushagra Vaid. SSD Failures in Datacenters: What? When? And Why? In *Proceedings of the 9th ACM International on Systems and Storage Conference (SYSTOR)*, 2016.

[37] Biswaranjan Panda, Deepthi Srinivasan, Huan Ke, Karan Gupta, Vinayak Khot, and Haryadi S. Gunawi. IASO: A Fail-Slow Detection and Mitigation Framework for Distributed Storage Services. In *Proceedings of the 2019 USENIX Annual Technical Conference (USENIX ATC)*, 2019.

[38] Eduardo Pinheiro, Wolf-Dietrich Weber, and Luiz André Barroso. Failure Trends in a Large Disk Drive Population. In *Proceedings of the 5th USENIX Conference on File and Storage Technologies (FAST)*, 2007.

[39] Peter J. Rousseeuw and Mia Hubert. Robust Statistics for Outlier Detection. *WIREs Data Mining and Knowledge Discovery*, 2011.

[40] Bianca Schroeder and Garth A. 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)*, 2007.

[41] Bianca Schroeder, Raghav Lagisetty, and Arif Merchant. Flash Reliability in Production: The Expected and the Unexpected. In *Proceedings of the 14th USENIX Conference on File and Storage Technologies (FAST)*, 2016.

[42] Bianca Schroeder, Eduardo Pinheiro, and Wolf-Dietrich Weber. DRAM Errors in the Wild: A Large-Scale Field Study. In *Proceedings of the 2009 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, 2009.

[43] Scott Shadle. NAND Flash Media Management Through RAIN. https://www.micron.com/-/media/client/global/documents/products/technical-marketing-brief/brief_ssd_rain.pdf, 2011.

[44] Charles Spearman. The Proof and Measurement of Association Between Two Things. *American Journal of Psychology*, 100(3/4):441–471, 1987.

[45] Amy Tai, Andrew Kryczka, Shobhit O. Kanaujia, Kyle Jamieson, Michael J. Freedman, and Asaf Cidon. Who's Afraid of Uncorrectable Bit Errors? Online Recovery of Flash Errors with Distributed Redundancy. In *Proceedings of the 2019 USENIX Annual Technical Conference (USENIX ATC)*, 2019.

[46] Guanying Wu and Xubin He. Delta-FTL: Improving SSD Lifetime via Exploiting Content Locality. In *Proceedings of the 7th ACM European Conference on Computer Systems (EuroSys)*, 2012.

[47] Erci Xu, Mai Zheng, Feng Qin, Yikang Xu, and Jiesheng Wu. Lessons and Actions: What We Learned from 10K SSD-Related Storage System Failures. In *Proceedings of the 2019 USENIX Annual Technical Conference (USENIX ATC)*, 2019.

[48] Qiao Zhang, Guo Yu, Chuanxiong Guo, Yingnong Dang, Nick Swanson, Xinsheng Yang, Randolph Yao, Murali Chintalapati, Arvind Krishnamurthy, and Thomas Anderson. Deepview: Virtual Disk Failure Diagnosis and Pattern Detection for Azure. In *Proceedings of the 15th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2018.

[49] Xuebin Zhang, Jiangpeng Li, Hao Wang, Kai Zhao, and Tong Zhang. Reducing Solid-State Storage Device Write Stress through Opportunistic In-place Delta Compression. In *Proceedings of the 14th USENIX Conference on File and Storage Technologies (FAST)*, 2016.

[50] Kai Zhao, Wenzhe Zhao, Hongbin Sun, Xiaodong Zhang, Nanning Zheng, and Tong Zhang. LDPC-in-SSD: Making Advanced Error Correction Codes Work Effectively in Solid State Drives. In *Proceedings of the 11th USENIX Conference on File and Storage Technologies (FAST)*, 2013.

A Artifact Appendix

Abstract

The artifact consists of the first large-scale public dataset on real-world operational data of NVMe SSD. With this dataset, we have identified a series of major reliability changes in NVMe SSD. The community could leverage our dataset and findings to understand the major reliability changes in NVMe SSD, and design effective reliability solutions (e.g., detecting and predicting failures) in production environments.

Scope

Most major findings in the main text (i.e., Findings 1-8 and 10) could be validated by exploring the dataset. Moreover, practitioners could make use of this dataset to investigate the *fail-stop* and *fail-slow* failure characteristics of NVMe SSD. For example, the dataset could be used to design fail-slow detection algorithms or to predict fail-stop or fail-slow failure occurrences in large storage systems.

Contents

The dataset primarily covers:

• **SMART logs and failure tickets** of around 700K NVMe SSDs of 11 drive families from three vendors during a one-year span. Practitioners could make use of them to investigate the *fail-stop* failure characteristics of NVMe SSD.

• **Performance logs** (i.e., device-level write latency time series) of around 97K NVMe SSDs and 141K SATA HDDs. Practitioners could make use of them to investigate the *fail-slow* failure characteristics of NVMe SSD, and compare them with those of SATA HDD.

Hosting

The open-source dataset is hosted by Tianchi of Alibaba Cloud at https://tianchi.aliyun.com/dataset/dataDetail?dataId=128972 with detailed instructions. Please refer to the above link for more information. We commit to ensuring the availability of this dataset.