SSD Failures in Datacenters: What? When? and Why?

Iyswarya Narayanan*, Di Wang†, Myeongjae Jeon†, Bikash Sharma†, Laura Caulfield†,
Anand Sivasubramaniam*, Ben Cutler†, Jie Liu†, Badriddine Khessib†, Kushagra Vaid†

*The Pennsylvania State University, †Microsoft Corporation

*{iun106,anand}@cse.psu.edu,
†{wangdi,myeoje,bsharma,laura.caulfield,bcutler,jie.liu,bkhessib,kushagra.vaid}@microsoft.com

Abstract

Despite the growing popularity of Solid State Disks (SSDs) in the datacenter, little is known about their reliability characteristics in the field. The little knowledge is mainly vendor supplied, and such information cannot really help understand how SSD failures can manifest and impact the operation of production systems, in order to take appropriate remedial measures. Besides actual failure data and the symptoms exhibited by SSDs before failing, a detailed characterization effort requires wide set of data about factors influencing SSD failures, right from provisioning factors to the operational ones. This paper presents an extensive SSD failure characterization by analyzing a wide spectrum of data from over half a million SSDs that span multiple generations spread across several datacenters which host a wide spectrum of workloads over nearly 3 years. By studying the diverse set of design, provisioning and operational factors on failures, and their symptoms, our work provides the first comprehensive analysis of the what, when and why characteristics of SSD failures in production datacenters.

Categories and Subject Descriptors B.8.1 [*Hardware*]: Reliability, Testing, and Fault-Tolerance

Keywords solid state drives; reliability; characterization;

1. Introduction

Storage system reliability is of paramount importance because storage component failures can lead to data corruption, or even permanent data loss. Consequently, beyond multi-level redundancies, timely replacement of storage devices is common in production datacenters. A direct consequence is the hardware replacement costs. An indirect consequence is the associated downtime to fix the problem and/or replace the device. It can even take several days to repair/replace a storage component after its failure, with associated server being unusable during this period. To account for this downtime, datacenters resort to over-provisioning (which can add significant cost) in order to meet the desired application availability Service Level Agreements (SLAs).

In the storage stack, SSDs are obviously at an advantage compared to HDDs in terms of failure rates. However, (i) SSDs are between 4X-40X costlier per GB than HDDs, depending on their grade (neutralizing, and in fact out-weighing the lower failure rate advantage); and (ii) an SSD-related failure ticket in our dataset results in a replacement 79% of the time compared to 11% for HDD-related tickets (i.e. SSD related failure tickets are more critical in the datacenter). These factors, together with rapid SSDs adoption[3, 13], motivate us to understand SSD reliability.

The current knowledge on SSD failure rate is primarily vendor supplied, based on accelerated lab testing under controlled conditions. In addition to the parameters they are tested for, numerous other factors in a production environment (e.g., diverse sets of workloads, environment, management policies, etc.) may not have been considered. Also, simply understanding vendor specified failure rates may not suffice, even if they hold in practice. A datacenter operator may need to understand the what, when and why characteristics for appropriate provisioning and operational decisions, that are not easily captured by a single (failure rate) metric.

Understanding SSD failure characteristics can be valuable in several ways. Though not comprehensive, some use-cases include: (i) pick vendors and models with more rigor by evaluating the performability-capacity-cost trade-offs; (ii) evaluate the consequences of SSD failures not just on its TCO, but also on the datacenter as a whole based on associated server downtimes; (iii) provision (hot or cold) spares accordingly; (iv) deployment for the right workloads based on their read-write characteristics; and (v) anticipating failures and taking appropriate actions including pro-active service, re-purposing for different workloads, etc.

However, with SSDs being relatively in their infancy compared to their HDD counterparts (past 5-8 years vs. sev-

SYSTOR '16, June 6–8, 2016, Haifa, Israel.

ACM 978-1-4503-4381-7/16/06…$15.00.
http://dx.doi.org/10.1145/2928275.2928278

eral decades), little is understood about their real-word failure properties, in order to be useful for the above purposes. There are a few studies (e.g. [10, 15, 33]) that examine specific errors and their effects in a laboratory setting. To our knowledge, prior large-scale field studies on SSD failures are from Facebook [24] and Google [31]. However, these studies (i) examine a single kind of SSD hardware failure (bit error rates) and not all possible SSD failures that could take the server down; and (ii) analyze correlating factors independently, which makes it difficult to understand the what, when and why answers that could depend on several workload and datacenter spatio-temporal factors.

We present an extensive characterization of failure data from over half a million SSDs that span five very large and several edge datacenters, over a span of nearly 3 years. These SSDs serve a wide spectrum of workloads including Big Data Analytics, Content Distribution Caches and Web Search Engines. that exhibit diverse characteristics. Beyond failure data, several other influential factors such as design, provisioning, and workload evolution data (read/write volumes, write amplification, etc.) have also been collected at fine spatial (datacenter , rack and server levels) and temporal resolution. In addition, SSD failure symptoms provided by the SMART (Self-Monitoring, Analysis and Reporting Technology) [2] attributes have also been captured. Using a comprehensive multi-factor analysis of this large dataset, this paper makes the following important contributions:

- The observed Annualized Failure Rate (AFR) in these production datacenters for some models is significantly higher (as much as 70%) than that quoted in SSD specifications, reiterating the need for this kind of field study.
- Four symptoms - Data Errors (Uncorrectable and CRC), Sector Reallocations, Program/Erase Failures and SATA Downshift - experienced by SSDs at the lower levels are the most important (in that order) of those captured by the SMART attributes.
- Even though Uncorrectable Bit Errors in our environment are not as high as in a prior study [24], it is still at least an order of magnitude higher than the target rates [26].
- There is a higher likelihood of the symptoms (captured by SMART) preceding SSD failures, with an intense manifestation preventing their survivability beyond a few months. However, our analysis shows that these symptoms are not a sufficient indicator for diagnosing failures.
- Other provisioning (what model? where deployed? etc.) and operational parameters (write rates, write amplification, etc.) all show some correlation with SSD failures. This motivates the need for not just a relative ordering of their influence (to be useful to a datacenter operator), but also a systematic multi-factor analysis of them all to better answer the what, when and why of SSD failures.
- We use machine learning models and graphical causal models to jointly evaluate the impact of all relevant factors on failures. We show that (i) Failed devices can be differentiated from healthy ones with high precision (87%) and recall (71%) using failure signatures from tens of important factors and their threshold values; (ii) Top factors used in accurate identification of failed devices include: Failure symptoms of data errors and reallocated sectors, device and server level workload factors such as total NAND writes, total reads and writes, memory utilization, etc.; (iii) Devices are more likely to fail in less than a month after their symptoms match failure signatures, but, they tend to survive longer if the failure signature is entirely based on workload factors; (iv) Causal analysis suggests that symptoms and the device model have direct impact on failures, while workload factors tend to impact failures via media wear-out.

2. Data collection

Datacenter hierarchy: Our study covers five large and several small edge (closer to users) datacenter facilities with diverse properties packaging, cooling and availability design. The datacenter facility spans numerous racks. A *rack* hosts multiple servers and network equipment of a specific configuration, that is referred to as *Rack SKU*. In these datacenters, a rack is the smallest unit of deployment, and all the servers of a rack are assigned the same workload type. Our dataset consists of six rack SKUs from two different vendors. In our study, each *server* has 2 sockets, 6-8 cores per socket, 32-128GB memory, 0-2 SSDs, and 4 HDDs based on the SKU configuration.

Runtime collects data for monitoring system health and performance issues through various sources in the datacenter. For instance, SMART monitoring system is employed in HDDs and SSDs to detect and report various failure indicators in addition to normal usage. Performance counters (e.g., cpu, memory, storage utilization, etc.) are constantly used to track the performance and well-being of operating systems and applications. Table 1 presents the data collected and used for this study.

SSD Population: We examine over half a million SSDs in these datacenters. Table 2 summarizes their salient characteristics by categorizing them into five groups based on their vendors and models. Each group has hundreds of thousands of devices. Majority of them come from a single vendor (named as 1) spanning multiple generations. The older generation SSDs (1-A, 1-B and 1-C) have a capacity of 160GB and have been operational for over 2.5 years. The newer generations in 1-D and 2-A have a capacity of 480GB each, with mean age slightly below 2 years. They all use MLC based flash medium. Models from vendor 1 are consumer class whereas the model 2-A SSDs are enterprise class.

Workloads: The SSDs under study are used by different classes of cloud applications and we identify the following four major categories: (i) W1 - platform for big data analysis, (ii) W2 - content caching in edge nodes close to customers, (iii) W3 - datacenter management software, (iv) W4

Design/Provisioning Features : Categorical		
Attributes		**Type**
Facility, Rack SKU, SSD Vendor, SSD Model		static
Server level workload : Numeric		
Attributes		**Type**
Utilization of CPU, Memory, Network		daily average
Space utilization of SSDs, HDDs		daily average
Disk queue length		daily average
Device level workload : Numeric		
Attribute	**SMART value**	**Type**
Reads	raw	cumulative, daily average
Host writes	raw	cumulative, daily average
NAND writes	raw	cumulative, daily average
Reads+Writes	raw	cumulative, daily average
WAF	derived	cumulative
Read/Write ratio	derived	cumulative
Media wear-out	val	normalized
Device level symptoms : Numeric		
Attribute	**SMART value**	**Type**
Reallocated sectors	raw	cumulative event
Program fail	raw	cumulative event
Erase fail	raw	cumulative event
Reserve space	val	normalized
Uncorrectable errors	raw	cumulative event
CRC errors	raw	cumulative event
SATA downshift	raw	cumulative event

Table 1: List of features/factors/attributes. Type represents the information type captured. An instance of these features represents a device's signature.

Model	Size	μ_{age}	μ_{reads}	μ_{writes}	Lith.
1-A	160GB	3.17 yrs.	NA	42.8 TB	34nm
1-B	160GB	3.31 yrs.	138.7 TB	25.1 TB	25nm
1-C	160GB	2.69 yrs.	99.9 TB	11.7 TB	25nm
1-D	480GB	1.92 yrs.	145.2 TB	40.3 TB	25nm
2-A	480GB	1.8 yrs.	NA	NA	20nm

Table 2: SSDs under study. μ_{age} - average age, μ_{reads} and μ_{writes} - the average amount of data read and written, respectively, per disk since its deployment, Lith. - lithograph, NA - unavailable data.

- web search which includes indexing, multimedia, object store, advertisement, and others. These workloads have very different access patterns and read/write intensities, and Figure 1 captures their difference in daily average of read/write characteristics. Note, except for workload W1 (big data analysis) all the other workloads have higher reads than writes. This is distinct from the previous study by Facebook where the difference between reads and writes is not that significant as shown in Figure 1.

3. SSD reliability

3.1 Flash Reliability Basics

NAND flash cells can sustain only a certain number of program/erase cycles, as specified by the *endurance rating*, before they permanently wear out. SSDs use *wear-leveling* to distribute wear evenly across cells. Despite this, wear out can lead to capacity fade over time. They are also prone to some reversible failure phenomenon resulting in data errors. First, they are susceptible to *retention errors* caused due to leakage current, which worsens with time when not acted upon. Second, they also suffer from phenomenon such as *read disturb* and *program disturb* errors, where read or program of a row or block of cells affects the threshold voltage of untouched cells in its vicinity. Flash controllers have proactive and reactive mechanisms in place, to prevent the flash error propagation to higher levels in the system stack. Consequently, not all of the above-mentioned failures propagate to upper layers. But, ones that do propagate can result in fail-stop failures. Infrastructure management layer at the datacenter [18] captures such failures and creates a failure ticket for further inspection. In this work, we analyze and characterize these fail-stop failures arising from SSD failures[1] that propagated up to the software in a production datacenter environment.

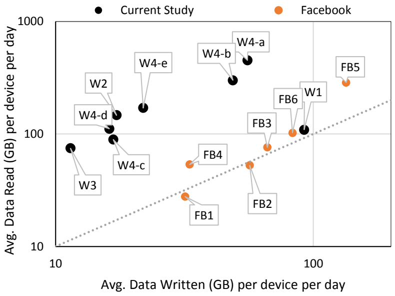


Figure 1: Workloads and SSD usage characteristics. Dotted line represents equal amount of daily data reads and writes. Read (write) dominant workloads are above (below) the dotted line.

3.2 SSD Failures and Their Implications

DEFINITION 1. *Failed Device: We identify an SSD to fail-stop, if the result of some underlying SSD events/failures propagates to the corresponding server, causing it to be shutdown for external (sometimes physical) intervention or investigation. The device will be replaced or repaired subsequently*[2]*. We refer to a device that fail-stops anytime during our observation window, as a "Failed" device.*

DEFINITION 2. *Healthy Device: Any SSD that does not fail-stop during observation window is a "Healthy" device.*

Henceforth, a failure refers to a fail-stop failure in this paper. Any other SSD error/failure that does not take down the server is referred to as a symptom. Even if a device does not fail-stop within our observation time window, it could fail-stop immediately after. In order to prevent mis-classification, we use a larger observation window (of 34 months) to classify devices, and conduct analysis for the data in the first 30 months, leaving 4 months for only classification.

Why focus on fail-stop failures? Fail-stop failures are significant in our data set, and the consequent downtime can

[1] We investigated failure tickets and identified SSDs as the failure source.

[2] Yet, we conservatively term it as failed in this study, since the downtime is itself a significant concern/cost. Nearly 80% of the fail-stopped SSDs were replaced in our observed datacenters during the observation window.

lead to substantial Quality-of-Service (QoS), cost and availability concerns. To quantify these failures, we use Annualized Failure Rate (AFR) as the metric, which is commonly used to report hardware failures [28, 29]:

$$AFR = \frac{\text{Total devices with failures}}{\text{Total device years}} \text{in } \%$$

Figure 2 shows the AFRs for various SSD models under study. Model 2-A (an enterprise class SSD) has much lower AFR compared to the other consumer class models. For the latter, SSD specifications report AFRs between 0.61% - 0.73%. However, except model 1-A which is in the specified range, both model 1-B and model 1-C exceed the published AFR, while model 1-D is well below. *This indicates that a production datacenter environment running real workloads can affect a SSD's failure characteristics very differently from the operating conditions imposed by their corresponding manufacturer's test environments.*

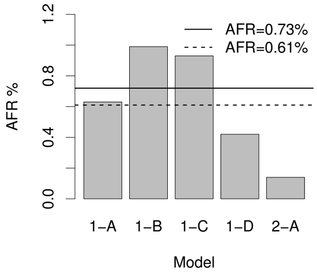


Figure 2: AFR vs. SSD models. Horizontal lines - Vendors reported AFR.

As mentioned earlier, the consequences of fail-stop failures are multi-fold. This includes: (i) lower Quality-of-Service and availability to hosted applications (and associated loss in revenue), and/or (ii) higher capital and operating expenses in provisioning extra capacities to guarantee a promised availability to these applications.

As SSD footprint continues to rapidly increase in datacenters [3, 13], SSDs are replacing HDDs in Edge datacenters, CDNs [1] and public clouds [25], beyond its widespread role as data caches. Therefore, it becomes extremely crucial to understand the what, when and why of SSD failures from the field. Towards this goal, we begin by examining symptoms experienced by SSDs to understand whether they can help identify devices that fail. After showing that they do not suffice, we conduct a more in-depth analysis across a wide range of parameters to study the correlations/causalities.

3.3 SSD Failure Symptoms

Even before the onset of a fail-stop failure, SSDs can exhibit the following symptoms from underlying problems, captured using SMART attributes – Reallocated Sector Count, Program/Erase Fail Count, CRC and Uncorrectable Error Count, and SATA Downshift Count. Some of these symptoms can by themselves be debilitating enough to be viewed as a "failure" in certain situations [24, 31], even if they do not immediately result in a fail-stop. However, in this paper, we still treat these as symptoms, and consider only the fail-stop events, as described previously, to be failures. We measure the extent of symptom occurrence in the datacenter as the percentage of devices where the corresponding SMART attribute value is non-zero in the total population.

Reallocated (Realloc) Sector Count: It captures the number of times a sector was re-allocated elsewhere on the SSD. Reallocations result in reduction of reserve space, which plays an important role in reducing write amplification and wear-leveling [17]. While this reduction may not immediately result in fail-stop failures, it could have a long-term impact on the lifetime of NAND cells. The oldest model 1-A shows the highest extent of reallocations with more than 80% of the devices exhibiting at least one reallocation. Interestingly, a younger model 1-D also shows a significant reallocations, affecting more than 7% of the devices. As shown later, the reallocation count is an important (though not sufficient) symptom leading up to fail-stop failures.

Program/Erase (P/E) fail count: It captures failures in program/erase operations that indicate problems in the underlying flash medium. Nearly, 0.5%-3.23% of the devices show this symptom. Such program/erase failures can also cascade into sector reallocations.

CRC and Uncorrectable errors: These events stem from data errors in media or errors in the communication link. Even if such symptoms may not necessarily propagate to higher levels because of ECC (Error Correction Code), they could have performance consequences. These errors affect about 0.47% to 2.8% of devices. Bit Error Rate (BER) (as in [15, 24]) is another standard metric used to capture the rate at which such errors occur, relative to the total data read by the device. Uncorrectable Bit Error Rate (UBER) is similarly computed for SSD uncorrectable errors (but can be host-correctable). Prior works such as [15, 24], provide extensive characterization of data errors using BER metric. Table 3 presents the UBER for models that expose both uncorrectable error count and total data usage. *Even though uncorrectable errors are not as common in our datacenters as those seen in the Facebook study [24], the UBER observed in our dataset (10^{-11} to 10^{-14}) is at least an order of magnitude higher than the target rate (10^{-15}) [26].*

Model	1-B	1-C	1-D
UBER/device	6.15 x 10^{-14}	1.27 x 10^{-11}	1.24 x 10^{-14}

Table 3: Uncorrectable Bit Error Rate (UBER).

SATA downshift count: The SATA interface downgrades to a lower signaling rate when it encounters more errors. This low signaling rate (and the underlying cause) can potentially result in performance degradation. The reason for downshift could be either a temporary disturbance or a permanent problem in the storage/communication medium. Selecting a lower signaling rate is not uncommon in SSDs, with more than 5% of devices in model 1-C going down to a lower signalling speed during their lifetime.

Failure symptoms correlate with AFR of SSDs. Figure 3 shows AFR for two groups of devices, those exhibiting symptoms and those not, for the four symptom categories. The presence of any of these symptoms increases AFR consistently by as much as 3X to 20X. In particular, data errors have a significant impact on AFR of SSDs with as much as 20X difference for the two groups. And, the presence of Reallocations and SATA downshift increases the AFR by 4X. Program and Erase fail events show a less pronounced difference of 2.75X.

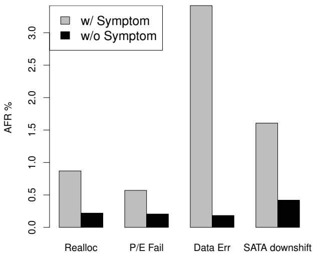
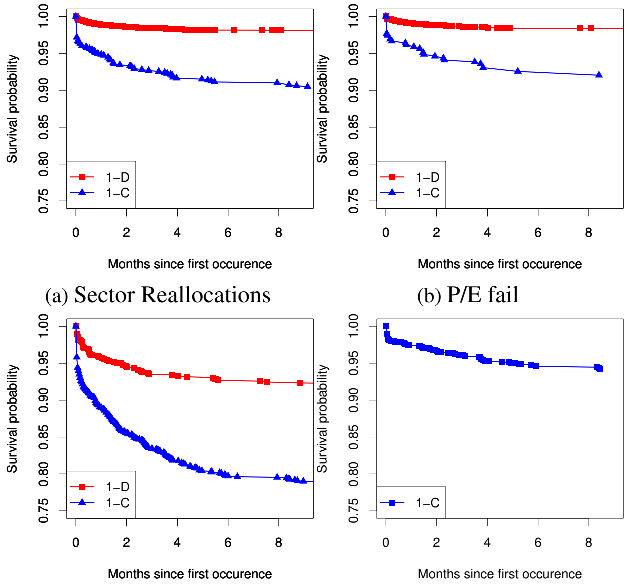


Figure 3: AFR in the presence and absence of symptoms.

3.4 Failures vs. SMART Symptoms

Given the preliminary indications that the failed SSDs have higher likelihood of exhibiting symptoms, we investigate whether they are a sufficient indicator. We use survival probability to study how long a device survives once it starts exhibiting symptoms. It is computed using Kaplan-Meier estimator [22] as follows: $S(t) = \prod_{t_i \le t} \frac{n_i - f_i}{n_i}$

where n_i is the number of survivors, and f_i is the number of failed devices at time t_i. n_0 is the population in the beginning at risk, with non-zero SMART counter values.

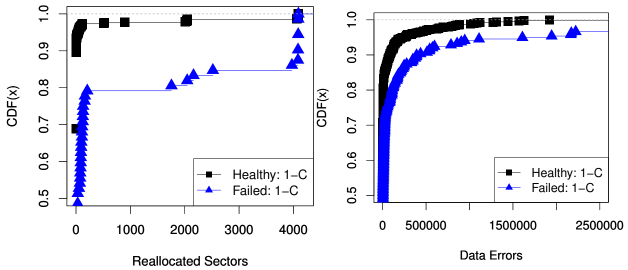


(a) Sector Reallocations (b) P/E fail

(c) Data Errors (d) SATA downshift

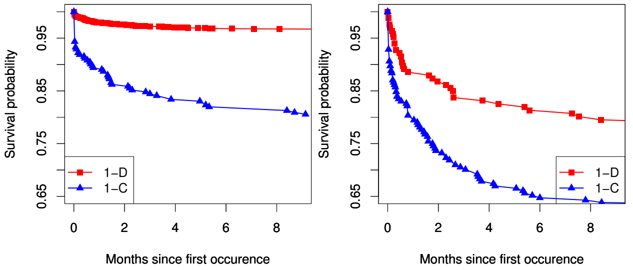
Figure 4: Survivability in the presence of symptoms.

Impact of Symptom Occurrence: Figure 4 presents the survival probability for devices with any of the four symptoms for up to 9 months after its first occurrence (note that the lines flatten out beyond this point). In the interest of space, we only show results for Model 1-C and Model 1-D to represent the older and younger model of devices, respectively. As shown in Figure 4a, the survival probability of 1-D is little affected by the presence of reallocation events, whereas for 1-C, 10% of devices fail within a few months after their sectors are reallocated. Symptoms P/E fail (Figure 4b) and SATA downshift (Figure 4d) also show a similar effect, though less pronounced. Data errors (Figure 4c), on the other hand, have the most significant influence on survival probability for both the models. Nearly 22% and 8% of devices from models 1-C and 1-D respectively fail within a window of 9 months after this symptom was exhibited.



(a) Sector Reallocations (b) Data Errors

Figure 5: CDFs of symp. intensity in failed and healthy devices.



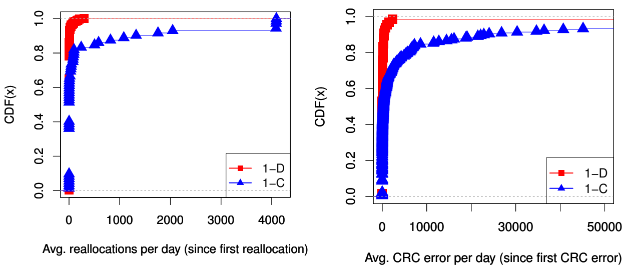
(a) Sector Reallocations (b) Data Errors

Figure 6: Survivability at high risk. High risk group has symptom intensity greater than 80^{th} percentile value.

Impact of Symptom Intensity: Just one occurrence of a symptom, may be too ephemeral towards classifying whether a device would fail or not. For instance, Figures 5a and 5b show the Cumulative Distribution Functions (CDFs) of reallocated sectors and data errors respectively, for failed and healthy devices of model 1-C. They show that symptoms are more prevalent in failed devices. In particular, a large fraction of healthy devices have no symptoms, and even if the symptoms occur, they are very mild in intensity, e.g., 80% of healthy devices have fewer than 2 reallocated sectors. In contrast, failed devices have much higher symptom occurrences, e.g., 20% of failed devices have over 1000 sectors reallocated. The other symptoms also show similar behavior and data is not explicitly presented here.

We next analyze the survival probability of SSDs that are at higher risk. Specifically, we use the 80^{th} percentile of each symptom's CDF for all devices (both healthy and failed) to represent high risk. Figure 6 shows that devices of both models fail significantly faster when the risk is high. This effect is more prominent in model 1-C, where nearly 20% and 36% of high risk devices fail within 9 months after first occurrence of reallocation and data errors, respectively.

Symptom's Progression Rate: A device which has started exhibiting symptoms can deteriorate progressively until it fails. To capture the progression of the symptoms (and hence the possible progression of the underlying cause), we calcu-



(a) Sector Reallocations

(b) Data Errors

Figure 7: Progression of the symptoms. Similar effects for P/E fail and SATA downshift.

late the symptom's incremental rate once a device starts exhibiting it in Figure 7. We can see that more than 20% of the devices show rapid progressive acceleration of symptoms in reallocation events (over 94 per day) in model 1-C, compared to less than 3% in model 1-D. Data errors also show a significant difference with 40% of the devices having a higher incremental rate (over 790 events per day) in model 1-C compared to less than 10% of the devices in model 1-D. As can be seen from model 1-C, faster progression of symptoms is associated with lower survival rates once the symptoms start to manifest, as shown in Figure 4 and Figure 7 . *There is a higher likelihood of the symptoms preceding SSD failures, with an intense manifestation and rapid progression preventing their survivability beyond a few months.*

Symptoms and Prognosis: Much of the data discussed in this subsection, seems to indicate that devices are likely to exhibit symptoms before (6-9 months) they fail. But, these symptoms alone are not a *sufficient* indicator of failure. Figure 8 compares the percentage of "Failed" and "Healthy" SSD population that exhibit these symptoms. From this data, we can observe the following:

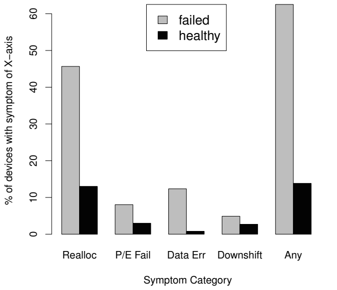
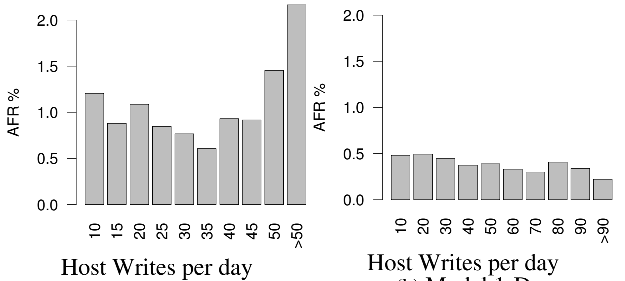


Figure 8: % of failed/healthy devices exhibit symptoms.

- Around 62% of "failed" devices displayed at least one of the above 4 symptoms (refer bar 'Any'), suggesting that a failed device has higher chance of having experienced one of these symptoms.
- However, 38% of the failed devices did not experience any such symptom. This suggests that a pure "symptom" based diagnosis of failures, may not be very accurate.
- Of the symptoms, while sector re-allocation dominates over the others, it is also prevalent among Healthy devices (and not just the Failed ones), suggesting it may not be a sufficient indicator of failure. The data errors, on the other hand, are rarely (only 1%) experienced by Healthy devices even if approximately 12% of failed devices have experienced it, i.e. a SSD experiencing a data error has a high likelihood of failure, even if not experiencing this symptom cannot be used for a "Healthy" prognosis.



(a) Model 1-C

(b) Model 1-D

Figure 9: AFR vs. Avg. Host writes per day (GB).

These suggest that, even though tracking symptoms is important, prognosis of whether a SSD will fail(-stop) or not, cannot be made entirely based on the symptoms. This motivates us to study other factors, beyond SMART symptoms, to better understand the characteristics of failed devices.

3.5 Failures vs. Correlating Factors

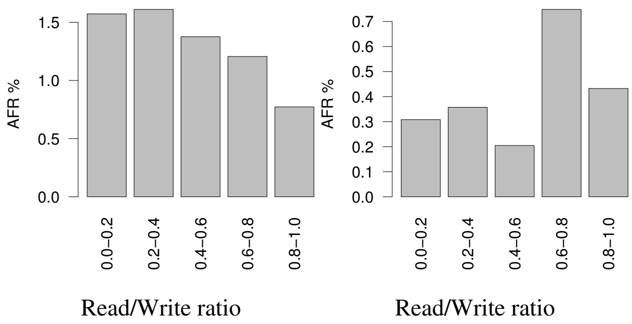
Various operational, design and provisioning decisions in the datacenter can affect SSD reliability. In this section, we study the relationships of such factors on SSD failures.

3.5.1 Device Level Factors

SSD usage (reads & writes) and its idiosyncrasies (e.g. write amplification) can impact its failure characteristics.

Host Writes: As explained earlier, flash cells can undergo a limited number of Program/Erase (P/E) cycles referred to as the endurance rating. The wear-out due to P/E cycles degrades SSD capacity over time and may eventually lead to its failure. The P/E operations are directly affected by the volume of writes to the device. Since we lack direct measurement of P/E cycles, we use host directed writes as its proxy. Figure 9 shows the relationship between mean host writes per device per day (in x-axis) and SSD AFR (in y-axis). Writes are histogrammed into buckets of equal value, and any bucket with less than 1% of the total devices are assigned to its nearest bin due to its small population size. In general, the impact of writes on SSD reliability varies across the SSD models. For instance, we observe a 2X to 4X increase in AFR as the average writes per day increases for the older models of 1-A, 1-B and 1-C (only 1-C is shown here). However, the AFR of the relatively younger model, 1-D, does not reveal any obvious correlation with the write usage – this model has higher capacity and is still far from its quoted endurance rating.

Reads: Understanding the impact of reads on SSD failure is essential for many systems that already employ SSDs as read caches. We studied the relationship between average reads per day and the SSD AFR. The data did not show much correlation between SSD AFR and the data read rate from these devices. This again re-iterates the point of previous



(a) Model 1-C

(b) Model 1-D

Figure 10: AFR vs. Read/Write Ratio. Model 1-B shows similar trends as model 1-C. Data unavailable for models 1-A and 2-A.

studies [7, 24, 26] that read disturbance, a failure mode seen at the flash chip level, is not predominant to manifest as a significant fail-stop failure in production datacenters.

Read/Write Ratio: In addition to the absolute read and write rates, we also study their combined impact using the read/write ratio: $\frac{\text{Reads (TB)}}{\text{Reads (TB)+ Writes (TB)}}$. It captures the dominance of reads vs. writes, and Figure 10 plots the AFR of devices for different bins of Read/Write Ratios. The plot shows that for model 1-C, write dominant devices have higher failures, which corresponds to the observation made in Figure 9(a) that higher write rate increases AFR for model 1-C. Model 1-B shows similar behavior (not shown here). On the other hand, model 1-D shows that AFR is skewed towards the read dominant region (x-axis > 0.6). Note that for model 1-D, using just the absolute rates of reads and host writes do not show any significant correlations.

Write Amplification (WAF): WAF is the ratio of data written to the flash memory to the host directed writes. It depends on factors such as workload characteristics (sequential/random access), background activities like garbage collection, etc. These additional writes may not only affect performance of a write request, but also consume additional P/E cycles to reduce lifetime. Figure 11 shows the correlation between WAF and SSD AFR. It shows that devices with either very high or very low write amplification have higher failure rates. While the latter category may appear counter intuitive, WAF can fall below 1 when the device performs compression. Despite such optimizations, they have higher failure rate as this category of devices also exhibited higher wearout. Also, note that SSD failures for model 1-D did not indicate any obvious correlation to *Host Writes*, but shows correlation with higher WAF. These justify the need for multi-factor analysis to understand the interactions among various factors.

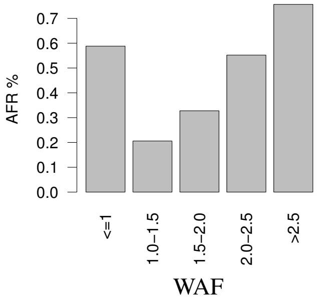
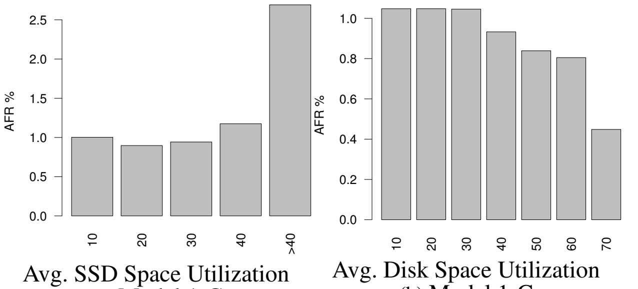


Figure 11: SSD AFR vs. WAF for model 1-D.

3.5.2 Server Level Factors

At each server, the usage of sub-components and their interactions with SSDs, can also affect SSD reliability.

Storage Space Utilization: We first look at the space utilization of the storage sub-components: HDDs and SSDs. Figure 12 shows the AFRs as a function of the space that is utilized (written) on the SSD and HDD on a daily basis. In general, higher SSD space correlates with an increase in SSD AFR. This is expected, as higher space utilization typically indicates more valid data written to the SSDs. Apart from P/E wear issues due to high write traffic, higher utilization translates to lower free space, leading to higher induced writes, garbage collection, wear leveling and their corresponding P/E wear. In contrast, we find an inverse relationship between SSD AFR and HDD space utilization on the corresponding servers, where higher HDD space utilization correlates with reduced SSD AFRs. This pattern is consistent across all the device models we observe. This may be due to the fact that most of the servers in our study use SSDs as buffers/caches for HDDs. A web search workload (write once, read many times) may cause high HDD utilization, but only reads on SSDs which do not generally reduce SSD lifetime. In contrast, a big data analysis workload (write many times) may not necessarily impose high HDD utilization, but may impose heavy writes on SSDs for better performance, which can significantly reduce SSD lifetime.



(a) Model 1-C

(b) Model 1-C

Figure 12: AFR vs. Space Utilization.

Memory and CPU Utilization: We have also studied the impact of memory (space) and CPU utilization of the servers on the failure characteristics of their respective SSDs. A workload (and corresponding server) that is memory intensive, is also likely to be storage intensive, since memory is typically used as a staging area for the storage. Consequently, higher memory utilization does tend to correlate with SSD failures for most models. The data does not show much statistical significance to draw conclusions to correlate SSD failures with processor utilization.

3.5.3 Datacenter Level Factors

At the datacenter level, design decisions such as configurations, packaging and cooling technologies etc. can impact the failure rates of different components [19, 30]. In the interest of space, we only show the correlation with Rack SKU.

Rack SKU: A Rack SKU represents the vendor, model, capacity, and configuration of the compute/storage devices (see Table 4). Figure 13 shows their SSD AFRs for different models. The results indicate: (i) For the same SSD model, there is a large difference in AFRs between the SKUs of different vendors. e.g. AFR of model 1-A in S1-1 is 14X as high as that in SKU S2-1. This indicates that factors external to SSDs also play a role in determining the failure rates; (ii) HDD deployed in SKU makes difference in AFR. Specifically, SKU S1-3a and SKU S1-3b are similar in all aspects but their HDD capacity, with SKU S1-3a having higher HDD capacity than SKU S1-3b. Both SSD models in SKU S1-3b show 2X difference in AFRs compared to the same models in SKU S1-3a.

SKU	Configuration
S1-1	1x 160GB SSD
S1-2	2x 160GB SSDs
S1-3a	2x 480GB SSDs
S1-3b	2x 480GB SSDs
S2-1	1x 160GB SSD
S2-2	2x 160GB SSDs

Table 4: Rack SKUs from 2 vendors (S1, S2). Suffix represent generation. S1-3a and S1-3b only differ in their HDD capacity.

Apart from other differences between the SKUs (such as where they are placed in the datacenter and associated environmental parameters), one important consequence of the SKU is the class of applications that they host, and as was observed earlier, the workloads do impact the failure characteristics of the SSDs.

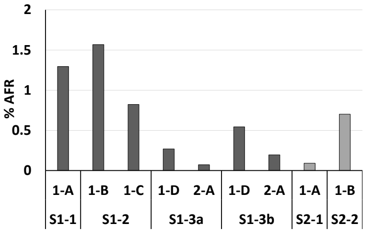


Figure 13: AFR for various SKUs, sorted by vendor and generation.

4. Characterization of Failed Devices

So far, we have identified several factors that correlate with SSD failures. The next step is to identify the important factors, their order of importance, and understand their temporal and causal relationships to better characterize the failed devices. In this section, we explore the *what*, *when*, and *why* of SSD failures by jointly considering them all. In particular, this is useful for a datacenter operator to take appropriate operational and provisioning decisions, which are not easily achieved by simply analyzing individual factors.

4.1 Understanding the *What?*

We use machine learning classification models to decide whether the signature of a SSD corresponds to failed or healthy class. The *signature* of a device represents the values of its correlating factors. Labeled failed and healthy devices, dataset uses the data from entire 2.5 years period. We use features including symptoms, device/server level factors, and design/provisioning factors as shown in Table 1. Among all the classification models (boosted decision tree, SVM, logistic regression, etc.) we have evaluated, we choose random forest model [5] (based on an ensemble of decision trees) due to its better performance and robustness in the presence of noisy inputs. It takes a device signature as input and classifies it to that of a failed or a healthy device. We also apply SMOTE [12] for over-sampling during the training phase to mitigate imbalance in the dataset as the number of failed devices is much smaller than the number of healthy ones. We perform 5-fold cross validation to avoid biased results – the dataset is divided into 5 folds, where each round of cross validation uses 4 folds for training and one for testing. The results present the average performance for all 5 rounds. To evaluate the classifier's performance (i.e., the ability to differentiate the failed from the healthy), we use standard classification metrics of precision and recall: precision $= \frac{|A \cap P|}{|P|}$, recall $= \frac{|A \cap P|}{|A|}$ where A is the true set of failed devices, and P is a set of failed devices identified by the model. High precision and recall is desired.

Category	Feature	Importance
Symptom	DataErrors	1
Symptom	ReallocSectors	0.943
Device workload	TotalNANDWrites	0.526
Device workload	HostWrites	0.517
Device workload	TotalReads+Writes	0.516
Server workload	AvgMemory	0.504
Server workload	AvgSSDSpace	0.493
Device workload	UsagePerDay	0.491
Device workload	TotalReads	0.475
Device workload	ReadsPerDay	0.469

Table 5: Top-10 features that affect accurate identification of failed devices (using permutation feature ranking).

Can we differentiate the failed and the healthy devices using their signature? We answer this affirmatively. Our classification model has a recall of 0.71 and precision of 0.87. i.e., it is able to identify 71% of all the true failed devices, and of all those classified as failed by the model, 87% of them are truly failed (the other 13% are false positives). Compared to the previous sections, this multi-factor classification model has much higher precision and coverage in identifying the failed devices than just using individual factors or symptoms.

What are the important factors for identifying failed devices accurately?

We use permutation feature ranking (PFR) [5] to answer this question. PFR metric reports the importance of each feature based on how much the model accuracy changes as the value of the feature is permuted.

Table 5 shows the top-10 most important fac-

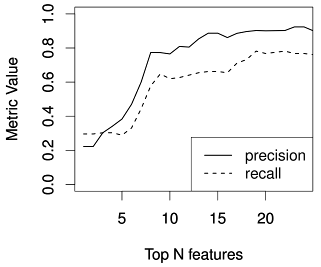


Figure 14: Precision and recall with factors added in their order of importance.

tors, with importance value normalized to that of the highest one. It shows that symptoms, which represent some underlying issues, rank the highest. Total NAND writes, which directly consumes P/E cycles, also ranks as a significant factor. The next few important factors are all based on device or server-level workloads such as total reads and writes, average memory and SSD space utilization, etc. Factors that rank lower may either correlate with higher ranked ones or suggest that they are bad indicators. So, we further analyze whether or not we need all the features.

We study the change in overall classification accuracy when only a subset of features is used to build the model. This is identified by the order of importance obtained using PFR. Figure 14 shows that there is a significant improvement in precision and recall as we add the top-8 most important features in Table 5. The accuracy even approaches close to the case of using all features. This indicates that in addition to top ranking symptoms, other highly ranked factors also play an important role in accurate identification of failed devices. In contrast, adding features that are ranked below the top-8 incurs only a modest increase in precision/recall.

What constitute the signature of failed devices? The signature of a failed device includes both the factors and their corresponding values. The random forest presents a list of rules to classify the devices. A rule is a set of conditions that evaluates the values of multiple features. Understanding the rules is an important but a challenging task, as even a model in modest size (like ours) contains several thousands of rules (> 3500). So, we investigate the properties of most frequent *patterns*, which are a part of the original rules and contain a small number of conditions (e.g., < 3). In general, patterns are easier to interpret, and frequent ones impact decisions more. Also, note that patterns with more conditions could deliver better performance at the cost of less interpretability. Using `inTrees` framework [14], we present patterns that are critical for identification of the failed device's signature.

Pattern	Condition	Class
P1	DataErrors<=1 & ReallocSectors<=5	H
P2	DataErrors<=1& WAF<=1	H
P3	MediaWearout=100 & WAF<=1	H
P4	AvgSSDspace >=10	F

Table 6: Most frequent patterns seen in the classification rules. MediaWearout is normalized to 0-100 range. 100 represents no wear, and 0 represents high wear. "H" - Healthy and "F" - Failed.

Table 6 presents four most frequent patterns with lowest error rate in their classification accuracy. They compare the thresholds for Data Errors, Reallocated Sectors, Media wearout, WAF and average SSD space usage. Patterns P1 and P2 identify the thresholds for individual symptoms in the healthy devices while accounting for other symptoms/factors. This is in consensus with what we observed in Section 3.4 that the symptom's intensity is important for it to manifest as failure. These patterns represent that a data error of 1 and a reallocation count of at most 5 is acceptable in certain scenarios. Pattern P2 and P3 identifies the signature of healthy devices in the low WAF group as the ones that do not exhibit data errors nor suffer from media wearout. This also agrees with our observation in section 3.5 that low WAF devices had higher AFR due to media wearout. The last pattern represents that increase in SSD space usage corresponds to failure when other factors are accounted for.

Implications: While being used to characterize the failed and healthy devices in this paper, the machine learning model that identifies tens of important features, their thresholds and their combinations can be leveraged to predict devices that are going to fail and take appropriate measures.

4.2 Understanding the *When?*

To understand when a device fails, we leverage the rules (identified in Section 4.1) with high frequency and lower error rate. We represent the time at which a device's signature matches the classification rules as t_m and the actual time of failure as t_f, $t_f >= t_m$. We present the CDF of the difference between these two timestamps (t_f-t_m) as the time to fail in Figure 15. It shows that the CDF is shifted towards the left with respect to an uniform distribution, indicating the devices are likely to fail sooner once their signature matches. We also observe devices with high intensity symptom and high rate of progression tend to fail in less than a month from t_m. In contrast, the devices that survive for a longer period tend to match rules based on device/server level workloads rather than symptoms. Some of these devices did not exhibit any symptoms at time t_m, but they presented with symptoms just before the time t_f. This indicates that, unlike symptoms, workload related factors have a long term accumulative effect in leading to SSD failures.

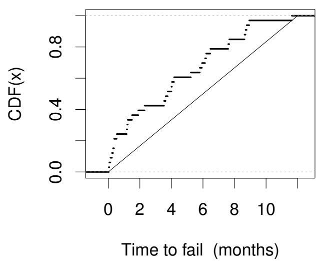


Figure 15: CDF of Time to fail ($t_f - t_m$). Mean = 4 months

Implications: The results suggest that: i) There exists a sufficient time window of opportunity to identify devices that are likely to fail, and take proactive actions to prevent unintended consequences of SSD failures. ii) The scope is especially large for signatures that are entirely based on the workload characteristics as they tend to survive longer. This indicates an opportunity to intervene and extend device lifetime such as by tailoring the workload assignment, device repurposing for an entirely different workload, etc. We leave such an in-depth study as a part of our future work.

4.3 Understanding the *Why?*

To understand "why", we study the causal relationship between factors, symptoms and failures. Typically, randomized controlled experiments are used to find such relationships.

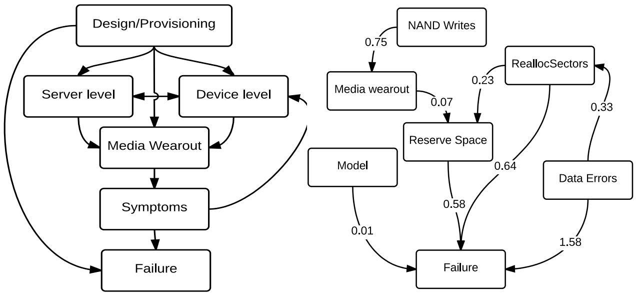
However, they are often impractical or even infeasible due to their time and cost constraints. Instead, we intend to find the causal relationships from the observational data. We use R package `pcalg` [21] that leverages causal graphical models [23] to learn causal structure and Pearl's do-calculus [27] to estimate causal effects, from our dataset. Here, graph models encode conditional independencies to capture the causal structure, where nodes represent features and edges represent the direction of causation. Though the true causal structure is very difficult to obtain, the framework can estimate the Markov equivalence class of the true causal graph. In addition, Pearl's do-calculus operation (mapping interventional and observational distributions in the causal graph) is applied to obtain possible causal measure. For instance, to measure the causal effect of node V_x on node V_y, the change of V_y in mean is used, i.e., $\frac{\partial}{\partial x}E[V_y|do(V_x = x)]$, where $P[V_y|do(V_x = x)]$ represents the resulting distribution of V_y after manipulating V_x.

Figure 16a shows a high level view of the causal structure,and Figure 16b focuses a part of it with effect measure shown on the edges (complete graph not shown due to space limitation). We observe the following: (i) symptoms (i.e. data errors and reallocated) have direct effect on failures. (ii) in addition, the observational data suggests that design/provisioning factors such as device model also have a direct impact on failures; (iii) other design/provisioning/ operational factors at server and device levels impact failures through Media Wearout. For instance, NAND writes indirectly affect Failure via Media wearout which reduces Reserve Space; (iv) the factors with stronger causal impact (e.g., Data Errors, ReallocSectors, NAND Writes in Figure 16b) match well with the top ranked ones in Table 5. Although the results assume no hidden variables (possible confounding factors which are not observed/collected in our study) are present, interestingly, we also see similar causal structure when hidden variables are considered.

Implications: Although this is only an estimate of possible causal relationships, it still provides useful insights and can serve as effective guidance for designing and prioritizing experiments (which are expensive in production datacenters) to identify and measure causality. For instance, the model identifies important provisioning knobs (SSD model) and control knobs (workload factors that affect Media Wearout) that directly or indirectly influence SSD failures in datacenters.

5. Related Work

Several works have examined failure trends of raw flash chips [4, 6, 8, 10, 15, 16, 26, 32]. Prior work has analyzed various modes of flash failures such as data retention [7, 10], program disturb [8], read disturb [9], endurance [4, 11], and power faults [32, 33]. The performance and robustness of SSDs under various read/write characteristics and power faults have also been studied [20, 33]. While these studies provide insights into SSD failure mechanisms, they are highly limited by their scale (observations over tens of devices), controlled testing environment, and the use of synthetic workloads. Their impact on production systems is unclear. The closest related work to ours, by Meza et. al. [24], examined the failure trends of flash based SSD in Facebook production environments using uncorrectable errors as the failure metric. This work mostly focuses on a single parameter analysis of SSD failures for a fleet comprising mainly of devices with equivalent read/write rates. To our knowledge, no prior work has considered SSD failures at scale in a production environment, by investigating a wide set of failures that really impact SSD/server downtimes, and studied the impact of a diverse set of design, provisioning and operational parameters on the failures (and their symptoms).



(a) High level interaction. (b) A partial view.

Figure 16: Causal structure and effect for factors on failure

6. Concluding Remarks

This paper presents an extensive characterization of SSD failures using field data.We first show that SSD failure rates in the field can be very different from what vendors specify. Next, we identify and quantify four types of SMART failure symptoms exhibited by the SSDs, and provide characteristics of symptom occurrence, intensity and progression rate. We show that despite their presence, the symptoms alone cannot be a sufficient indicator of failures. We have also studied the impact of multiple provisioning and operational factors across different layers of the datacenter hierarchy on SSD reliability. Many of these factors are individually influencing, and can also interact with each other in complicated ways, to impact SSD failures. We have used machine learning and graphical model based approaches to systematically consider the impact of multiple influential factors towards answering the *what*, *when* and *why* of SSD failures. We believe the insights gained from this paper can greatly influence the design, provisioning and operational decisions for SSDs in datacenters.

Acknowledgments

This work was supported, in part, by the National Science Foundation under grants 1302225 and 1302557.

References

[1] Enhanced Content Distribution Network with Intel Solid-State Drives. http://www.intel.fr/content/dam/www/public/us/en/documents/case-studies/cloud-computing-ssd-beijing-fastwebcase-study.pdf.

[2] American National Standards Institute. AT attachment 8 - ATA/ATAPI command set (ATA8-ACS), 2008. URL http://www.t13.org/documents/uploadeddocuments/docs2008/d1699r6a-ata8-acs.pdf.

[3] D. G. Andersen and S. Swanson. Rethinking Flash in the Data Center. *IEEE Micro*, 2010.

[4] S. Boboila and P. Desnoyers. Write Endurance in Flash Drives: Measurements and Analysis. In *USENIX FAST*, 2010.

[5] L. Breiman. Random Forests. *Machine learning*, 2001.

[6] Y. Cai, E. Haratsch, O. Mutlu, and K. Mai. Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis. In *DATE*, 2012.

[7] Y. Cai, G. Yalcin, O. Mutlu, E. F. Haratsch, A. Cristal, O. S. Unsal, and K. Mai. Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime. In *ICCD*, 2012.

[8] Y. Cai, O. Mutlu, E. F. Haratsch, and K. Mai. Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation. In *ICCD*, 2013.

[9] Y. Cai, Y. Luo, S. Ghose, E. F. Haratsch, K. Mai, and O. Mutlu. Read Disturb Errors in MLC NAND Flash Memory: Characterization, Mitigation, and Recovery. In *IEEE/IFIP DSN*, 2015.

[10] Y. Cai, Y. Luo, E. F. Haratsch, K. Mai, and O. Mutlu. Data Retention in MLC NAND Flash Memory: Characterization, Optimization, and Recovery. In *HPCA*, 2015.

[11] P. Cappelletti, R. Bez, D. Cantarelli, and L. Fratin. Failure Mechanisms of Flash Cell in Program/Erase Cycling. In *IEDM Tech. Dig.*, 1994.

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Int. Res.*, 2002.

[13] B. Debnath, S. Sengupta, and J. Li. FlashStore: High Throughput Persistent Key-value Store. *Proc. VLDB Endow.*, 2010.

[14] H. Deng. Interpreting Tree Ensembles with inTrees. *arXiv preprint arXiv:1408.5456*, 2014.

[15] L. M. Grupp, A. M. Caulfield, J. Coburn, S. Swanson, E. Yaakobi, P. H. Siegel, and J. K. Wolf. Characterizing Flash Memory: Anomalies, Observations, and Applications. In *MICRO*, 2009.

[16] L. M. Grupp, J. D. Davis, and S. Swanson. The Bleak Future of NAND Flash Memory. In *USENIX FAST*, 2012.

[17] X.-Y. Hu, E. Eleftheriou, R. Haas, I. Iliadis, and R. Pletka. Write amplification analysis in flash-based solid state drives. In *ACM SYSTOR*, 2009.

[18] M. Isard. Autopilot: Automatic Data Center Management. *SIGOPS Oper. Syst. Rev.*, 2007.

[19] W. Jiang, C. Hu, Y. Zhou, and A. Kanevsky. Are Disks the Dominant Contributor for Storage Failures?: A Comprehensive Study of Storage Subsystem Failure Characteristics. *Trans. Storage*, 2008.

[20] M. Jung and M. Kandemir. Revisiting Widely Held SSD Expectations and Rethinking System-level Implications. In *ACM SIGMETRICS*, 2013.

[21] M. Kalisch. Package pcalg. 2015.

[22] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 1958.

[23] S. L. Lauritzen. *Graphical models*. 1996.

[24] J. Meza, Q. Wu, S. Kumar, and O. Mutlu. A Large-Scale Study of Flash Memory Failures in the Field. ACM SIGMETRICS, 2015.

[25] Microsoft Azure Premium Storage. Microsoft Azure Premium Storage, 2015. https://azure.microsoft.com/en-us/blog/azure-premium-storage-now-generally-available-2/.

[26] N. Mielke, T. Marquart, N. Wu, J. Kessenich, H. Belgal, E. Schares, F. Trivedi, E. Goodness, and L. Nevill. Bit error rate in nand flash memories. In *IRPS 2008.*, 2008.

[27] J. Pearl. *Causality: Models, Reasoning, and Inference*. 2000.

[28] E. Pinheiro, W.-D. Weber, and L. A. Barroso. Failure Trends in a Large Disk Drive Population. In *USENIX FAST*, 2007.

[29] B. Schroeder and G. A. Gibson. Disk Failures in the Real World: What Does an MTTF of 1,000,000 Hours Mean to You? In *USENIX FAST*, 2007.

[30] B. Schroeder, E. Pinheiro, and W.-D. Weber. DRAM Errors in the Wild: A Large-scale Field Study. In *ACM SIGMETRICS*, 2009.

[31] B. Schroeder, R. Lagisetty, and A. Merchant. Flash reliability in production: The expected and the unexpected. In *FAST*, 2016.

[32] H.-W. Tseng, L. Grupp, and S. Swanson. Understanding the Impact of Power Loss on Flash Memory. In *DAC*, 2011.

[33] M. Zheng, J. Tucek, F. Qin, and M. Lillibridge. Understanding the Robustness of SSDs Under Power Fault. In *USENIX FAST*, 2013.