Carbon Footprint of Storage in Data Centers: The Impact of using SSDs for Key-Value Stores

Jakob Nibler and Thomas Ropars
Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG
Grenoble, France
uga@jnibler.de, thomas.ropars@univ-grenoble-alpes.fr

***Abstract*—The greenhouse gas emissions of data centers and the question of how to reduce them are a broad problem that has come into focus with the ongoing climate crisis. Yet, the emissions of storage infrastructures specifically are not well understood. Recent work shows that the manufacturing emissions of SSDs are significantly higher than those of HDDs. Also, the energy consumption of high-end SSDs is in general higher than the one of HDDs. This raises the question of whether the performance improvements provided by SSDs over HDDs are enough to also provide an advantage in terms of carbon footprint. In this paper, we analyze the lifecycle carbon footprint of SSDs and HDDs when used as storage devices for Key-Value stores. Considering state-of-the-art Key-Value stores specifically designed to make best use of HDDs or SSDs, we use specialized hardware to measure the power consumption of NVMe SSDs, as well as of the processor and memory to analyze the impact of different Key-Value stores and workloads on power. We conduct an analysis to determine if SSD-based systems outperform HDD-based systems with respect to their carbon footprint. Our results show that in most cases, the high operational energy efficiency of SSDs allow systems based on SSDs to have lower carbon footprint. However, HDD-based solutions should be considered in cases where the full potential of SSDs cannot be exploited and when the carbon intensity of the electricity powering the data center is low.**

I. Introduction

With the ongoing climate crisis, lowering the carbon emissions of data centers becomes a major concern. As of today, it is estimated that information communication technology (ICT) accounts for 1.8% to 2.8% of the Greenhouse Gas (GHG) emissions worldwide [1], with data centers being responsible for a significant fraction of it. The carbon emissions of data centers include embodied emissions and operational emissions. Efforts have already been made to improve the operational emissions [2], [3]. Different solutions, such as optimizing the Power Usage Efficiency (PUE) [2] and relying on renewable energy, have lead to a situation where operational emissions can represent less than 50% of the carbon emissions of data centers [4], and even much less in some cases [5]. Hence, it is of major importance to study solutions to reduce the embodied carbon footprint of data centers [6].

Analyses of the embodied carbon emissions of data centers identify storage devices as one major contributor [6], [7]. This is especially true when considering servers equipped with Solid-State Drives (SSDs): In Azure data centers, SSDs account for 30% of the embodied emissions in *compute* racks and 80% in *storage* racks [7], making up more than 60% of the embodied emissions of the data center.

Comparisons between the embodied emissions of SSDs and Hard-Disk Drives (HDDs) show that SDDs are much more carbon intensive to manufacture [7], [8]. Therefore, one could wonder if a straightforward solution to reduce the embodied carbon footprint of data centers could be to use more HDDs and fewer SSDs. Yet, SSDs offer orders of magnitude better performance than HDDs [9]. It is then unclear whether using HDDs actually reduces the global carbon emissions or if the gains with respect to the embodied emissions would be cancelled out by the increase in operational emissions.

The goal of this work is to assess whether using SSDs for storage in data centers is always the best solution to reduce the carbon emissions, or if there are cases where using HDDs can be more beneficial. Answering this question in the general case is difficult as many kinds of storage abstractions and databases are used in data centers [10], generating different access patterns to the underlying storage devices. Hence, for this study, we focus on a specific kind of databases commonly used in many cloud applications [11], [12], that is, Key-Value stores (KV stores). KV stores optimized to run on HDDs exist since several years [11], [13], [14]. KV stores designed to make the best use of the capabilities of modern NVMe SSDs have recently been proposed [15], [16], allowing us to run a fair comparison between the two types of systems.

Our analysis requires data about the embodied and the operational carbon footprint of KV stores running on the two types of devices. To estimate the embodied carbon footprint of storage devices, we rely on existing publications [8]. On the other hand, we measure the power consumption to deduce the operational carbon footprint. More specifically, we run tests with different types of workloads [17] with two state-of-the-art KV stores [11], [15], using specialized wattmeters to precisely measure the energy consumption of storage devices. We also use Intel RAPL counters [18] to capture the energy consumption of the processors and the memory in the storage server running the KV stores.

In addition to the lower embodied carbon footprint of HDDs and the higher performance of SSDs, multiple factors can impact the relative benefits of using SSDs for KV stores in data centers with respect to carbon emissions. The most obvious one is the carbon intensity of the electricity used to power the servers, which determines the operational emissions [4]. Another factor is the power consumption of SSDs. Contrary to HDDs that have nearly constant power consumption, the power consumption of SSDs varies significantly depending on the access patterns and can be much higher than the one of HDDs, as was observed in previous studies [19]. Our study includes precise measurements of the energy consumption of modern NVMe SSDs to document this point. A last important factor is the algorithms that are used to implement KV stores depending on the storage device. Whereas KV stores running on top of HDDs need to rely on *complex* algorithms such as LSM-trees [14], [20] to achieve high performance, KV stores targeting SSDs use much simpler algorithms. This implies different loads on the CPU and the memory. The CPU being the most power-consuming component in a server, this has a significant impact on the operational emissions. At the end, our problem boils down to the following question: taking all these factors into account, are the operational carbon emissions of servers based on SSDs low enough to compensate for their higher embodied emissions in a time that is lower than the lifetime of the devices.

We present an extensive study based on the data we collected through experiments. Our study includes two different NVMe SSD models to avoid drawing conclusions based on a single device that might not be representative. Our results show that in most cases, with KV stores, the high operational energy efficiency of SSDs leads to systems based on SSDs having a lower carbon footprint despite their higher embodied carbon emissions. The performance difference between NVMe SSDs and HDDs is so high that, in many cases, multiple HDDs would have to be used to reach the same performance as a single SSD device can provide, leading to a higher embodied footprint for the HDD-based system. Still, using HDDs can be beneficial when SSDs are not used at their full potential. Also, in data centers powered with less carbon-intensive electricity, the advantages of SSDs become less salient. Hence, the choice between SSDs and HDDs as storage device should still be made with care.

To summarize, the main contributions of this paper are:

- An extensive campaign of experiments to measure the energy consumption of SSD-based and HDD-based KV stores, and a break-down of this energy consumption per hardware component.
- A thorough analysis to determine the conditions under which SSD-based solutions allow reducing the carbon footprint of storage in data centers compared to HDD-based solutions, considering the case of KV stores.

The rest of the paper is organized as follows. Section II presents the background on the carbon footprint on storage devices and introduces KV stores. It also presents an evaluation of the energy efficiency of NVMe SSDs. Section III shows the result of our experiments measuring the energy efficiency of KV stores running on HDDs and SSDs. Finally, we analyze the lifecycle carbon footprint of HDD and SSD-based KV stores in Section IV.

II. Background

Several studies [1], [21] have documented the ICT contribution to global warming through GHG emissions, as well as its continuous growth. Datacenters play a significant role in these trends. In this paper, we use the term *carbon emission* (or *carbon footprint*) to refer to GHG emissions, as the global warming potential of activities is commonly translated into CO_2-equivalent (CO_2e) emissions [22].

Following the GHG protocol [23], the carbon emissions of a datacenter can be divided into 3 scopes [5]: the direct emissions (scope 1), the indirect emissions from purchased energy (scope 2, *i.e.*, the operational carbon footprint), the indirect emissions from manufacturing and transporting products (scope 3, *i.e.*, the embodied carbon footprint). In the case of datacenters, the focus is on operational and embodied emissions, as scope 1 emissions are negligible [5], [7].

In the past years, significant work has been put by cloud providers into reducing their operational emissions. It includes optimizing the PUE of datacenters [2], [24] as well as the energy usage of computing resources [25], [26], and taking advantage of renewable energy [5], [27], [28]. These improvements lead to a situation where embodied emissions are as important, or even more important, than operational emissions in hyperscale datacenters [4], [5]. Analyses of this embodied carbon footprint show that storage devices are a significant contributor, especially SSDs [7].

SSDs are a popular solution because of their much better performance than HDDs. However, SSDs have a higher embodied footprint per byte and consume more power per byte [7]. Hence, we can wonder which type of devices should be used to minimize the carbon footprint of datacenters.

In this section, we start by presenting data about the carbon footprint of storage devices. For our study, we need a good understanding of the energy consumption of SSDs. Hence, we present fine-grained measurements we collected on two different NVMe SSDs. Finally, since our study focuses on KV stores as an example of storage system used in datacenters, we finish by providing the necessary background about this kind of database.

A. The carbon footprint of storage devices

The motivation for this work is the observation that the embodied carbon footprint of SSDs is much higher per byte than the one of HDDs [8], [29]. When studying the embodied carbon emissions per byte (named Storage Embodied Factor – SEF), it can be observed that SSDs have a 2x to 10x higher SEF than HDDs. The most extreme numbers presented in [8] even show a 68x difference between the best HDD and the worst SSD. These numbers are, of course, to be taken with care as they mostly come from industry reports [29] and Life Cycle Analysis reports from computer vendors that are not focused only on storage devices [8].

Regarding the operational footprint of storage devices, the work by Harris and Altiparmak [19] that compares the energy efficiency of servers equipped with different kinds of storage devices (an HDD, an NVMe SSD, and an Intel Optane device), provides us with some useful information. First, the power consumption of HDDs is more or less constant no matter the workload. Second, the energy consumption of modern

SSDs varies significantly depending on the workload, and can be much higher than the one of HDDs. However, during their experiments the authors were only able to collect power consumption measurements at the level of the power supply of the server. Other studies also analyze power consumption of SSDs [30], [31]. However, these publications are rather old and are not representative of current SSDs. In Section II-B, we present a detailed evaluation to characterize the power consumption of NVMe SSDs according to the workload.

B. Energy consumption of SSDs

To precisely measure the power consumption of SSDs, we use a Quarch Power Analysis Module (PAM)[1]. The device is plugged on the same PCIe port as the SSD and measures the current flowing through the connection. It samples at a very high frequency of 250KHz, which we aggregate to values every 1ms using a function provided by the API of the device.

For our measurements, we consider two NVMe devices: a 1.6 TB SSD Dell Express Flash NVMe PM1725 AIC (called `NVMe1` hereafter) and a 1.6 TB SSD NVME Dell Samsung PM1735 (`NVMe2`). The complete description of the experimental testbed is presented is Section III-A. Note that our study considers only NVMe SSDs as this interface is able to take full advantage of the performance of modern SSDs, contrary to other interfaces like SATA [9], [15].

Our goal is to precisely measure the power consumption of the SSD devices and their energy efficiency (measured in MiB per joule). We want to observe the impact of the workload as well as the size of the written blocks on these metrics. To this end, we use `Fio`, the Flexible I/O Tester[2]. For all tests, we use `libaio` as I/O engine, which is the Linux native asynchronous engine. We set I/O to be non-buffered (`O_DIRECT`) to bypass the OS page cache, and we configure `Fio` to refill the I/O buffers on every I/O submission to prevent the NVMe from deduplicating data. Through empirical testing, we set the number of querying threads to 4 with a queue depth of 64, as these were the lowest numbers where we started saturating the drives.

We test 6 I/O patterns: sequential reads (`read`), random reads (`randread`), sequential writes (`write`), random write (`randwrite`), and two mixes of random reads and writes (`mix5050`, a 50/50 mix, and `mix8020`, a mix with 80% of reads and 20% of writes). We execute these workloads for blocksizes from 1kiB to 256kiB. Experiments run for 120 seconds. Each workload is executed three times and results are averaged. Negligible variations were observed between the runs for one workload.

Results show significant differences in power consumption between the two devices: `NVMe1` reaches a peak power consumption of around 18 W, *i.e.* roughly three times its base consumption (6.18 W), while `NVMe2` peaks at about 14 W, about two times its base consumption (7.17 W). It should be noted that these values are much higher than those of typical HDDs (The HDD we used consumes 6.02 W). The block size and the I/O pattern impact the power consumption of the 2 SSDs similarly. For both devices, the power consumption for smaller block sizes is lower, correlating with lower I/O rates. Also, read workloads, which have a higher IO rate, consume more power than writes for high block sizes (>16 KiB).

To better compare the two devices, we compute their energy efficiency (in MiB/J) for each workload and each block size by computing the ratio between I/O rate and power. Results are presented in Figure 1a and Figure 1b for `NVMe1` and `NVMe2` respectively. These results show that `NVMe1` is more energy efficient for small block sizes while `NVMe2` becomes more efficient for larger block sizes. For large block sizes, `NVMe2` is up to 40% more efficient for read-only workloads. Conversely, the two devices also have some similarities: Both are equally efficient on sequential and random workloads; Both are ∼65% more energy efficient on read workloads than on write workloads; Their energy efficiency for mixed workloads cannot be directly inferred from their energy efficiency on read-only and write-only workloads. To illustrate this last point, if we consider a block size of 256 kB and `NVMe1`, the energy efficiency for the 50/50 workload is 27% lower than the weighted average of the efficiencies from random reads and randoms writes whereas it is only 19% lower for the 80/20 ratio (and the results are significantly different for `NVMe2`).

These results show that the power consumption of SSDs can be high and that their energy efficiency greatly depends on the I/O pattern and the size of the written blocks. Furthermore, the energy efficiency of mixed I/O patterns is difficult to deduce from the energy efficiency of read-only and write-only workloads. Thus, we think that assessing the operational carbon footprint of SSDs in datacenters requires collecting data on realistic workloads. In this study, we consider the case of a type of storage engine commonly used in cloud environments, persistent KV-stores.

C. Key-value stores

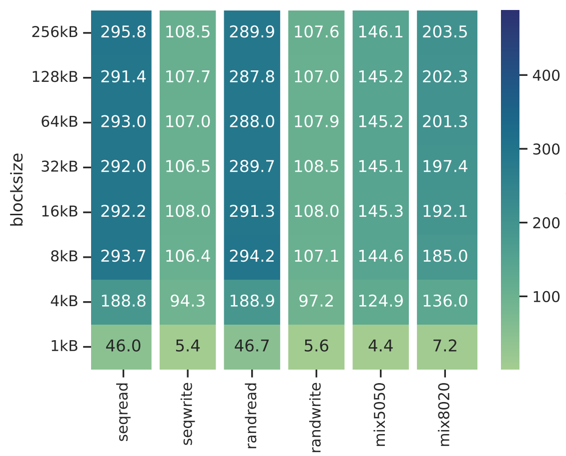
KV-stores are commonly used as storage engines in Cloud applications [32] and serve a large diversity of workloads [11], [33]. The most common data structure for KV-stores on HDDs is the Log-Structured Merge (LSM) Tree [11], [14], [20], which ensures most disk accesses are sequential. In this work, we use RocksDB[3] as an example of LSM-Tree-based persistent KV-store. RocksDB is an open-source KV-Store heavily used internally at Facebook [11].

Studies have shown that LSM-Tree-based solutions are not suited for NVMe SSDs [15], [16]. The random-access performance of such devices is so high that the complexity of LSM-Tree algorithms is counter-productive: The CPU becomes the performance bottleneck and prevents fully using the I/O bandwidth. Hence, simpler designs optimized for NVMe SSDs have been proposed [15], [16]. For our study, we consider KVell [15]. KVell does not sort data before writing them to

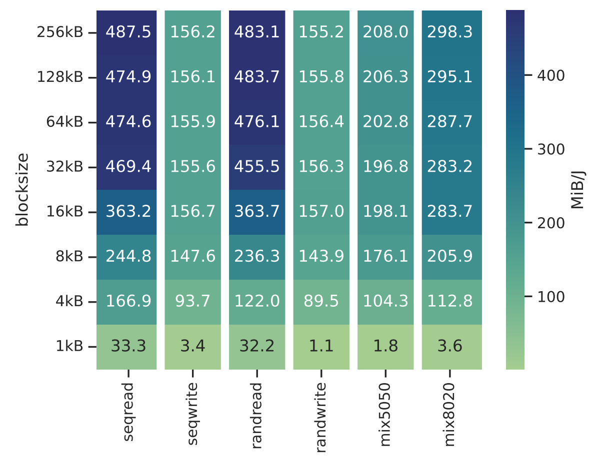
[1]https://quarch.com/products/power-analysis-module/

[2]https://fio.readthedocs.io

[3]https://rocksdb.org/



(a) `NVMe1` efficiency



(b) `NVMe2` efficiency

Fig. 1: Energy efficiency of NVMe SSDs in MiB/J for different workloads and block size granularities

disk and uses a combination of multiple techniques (batching, page cache, etc.) to reduce CPU overhead.

The solutions used to implement KV-stores on HDDs and SSDs are rather different. Hence, when studying the impact of the storage devices on the operational carbon footprint of KV-stores, one should not only consider the energy consumed by the storage devices, but also the impact on the other components of the servers. We can foresee that the CPU energy consumption per request will be lower with KVell compared to RocksDB, thanks to its simpler design. In the next section, we evaluate the energy efficiency of these two systems under different workloads, measuring power consumption not only at the level of the storage devices, but also taking into account the power consumption of the CPU and the memory of the storage servers.

III. Energy consumption of KV stores

To analyze the impact of SSDs and HDDs on the carbon footprint of datacenters, we need to evaluate the operational footprint of storage engines with both types of devices. This section evaluates the energy consumption of two KV stores, RocksDB and KVell, using an HDD and an NVMe SSD as storage device respectively, taking into account not only the energy consumption of the storage devices but also the energy consumption of the CPU and the memory of our storage server. We start this section be describing our hardware setup and methodology. Then, we present the results we obtained.

A. Methodology

1) Hardware setup: To run a fair comparison between HDD-based and SSD-based KV stores, we run all experiments with the same server. It is a Dell PowerEdge R940 server equipped with 4 16-cores Intel Xeon Gold 6130 CPUs and 768 GiB (24 x 32 GiB Hynix DDR4) of memory. A second server is used for load injection. The two servers communicate through a 100 Gbps Omni-Path network, hence the network is never a performance bottleneck in our experiments.

In addition to the two NVMe SSD devices described in Section II-B, our storage server features a 2.0 TB SAS Seagate ST2000NX0463 HDD. These three storage devices are used to store the KV store data depending on the experiments. The storage server includes one more SATA SSD (480 GB Intel SSDSC2KG480G7R) that hosts the operating systems and is used to store all collected measurements.

2) Measuring power consumption: To get a good understanding of the power consumption of KV stores, we measure power consumption at different levels. As described in Section II-B, we precisely measure the power consumption of the NVMe SSDs using Quarch PAM modules. On the other hand we do not have any special equipment to measure the power consumption of our HDD. However, previous studies have shown that the power consumption of HDDs is more or less constant when it is active [19], as the main contributor to power consumption is the rotation mechanic. Therefore, since the HDD is always active during experiments where it is used, we assume it to consume a constant power of 6.02W, the value provided by the data sheet of the constructor.

To measure the power consumption of the processor and the memory, we use Mojitos[4] to read the content of the Running Average Power Registers (RAPL) of Intel processors. These registers log the power consumption of the processor and the memory, with separate values for each package and each memory NUMA node. Note that the storage server we use to run the KV stores is way too powerful in terms of CPU and memory compared to the actual needs of RocksDB and KVell. Hence, to avoid over-estimating the CPU and memory power consumption associated with running the KV stores, we pin the KV-store process on a single NUMA node, to ensure that it uses the cores of a single processor and the corresponding memory. This way, the CPU/memory power consumption of the database is given by the RAPL values for the corresponding node. Moreover, we pin the monitoring

[4]https://gitlab.irit.fr/sepia-pub/mojitos

software (Mojitos and Quarch) on a different NUMA node to ensure that the corresponding power consumption is not included in the power consumption attributed to the database.

3) Software setup: All experiments are run with Debian 11 and Linux kernel version 5.10. The tested version of RocksDB is 9.7.4 and the version of KVell is commit af10b7a.

To stress the KV stores with different workloads, we use the YCSB benchmark [17]. However, to ensure that our power measurements are not impacted by the CPU-intensive activity of data generation in YCSB, we created our own client-server version of YCSB in Rust, where the generation of the benchmark data happens on a separate machine. This client sends requests to the KV store executed on a different machine where the power consumption is measured. Note that data generation is CPU-intensive because we need to generate random keys for each request, but also random values for write requests, to prevent unrealistic compression by the KV stores.

We run tests with workloads A (50% read, 50% update), B (95% read, 5% update), C (100% read) and E (95% scan with max length 100, 5% insert) of YCSB. We execute the workloads on databases with 100M entries and use 8-byte keys and 1kB values. We choose keys for requests either according to a uniform distribution or a zipfian distribution with exponent 0.99. The zipfian distribution allows simulating locality in the data accesses, as it has been observed in several real workloads [11], [33]. We choose to not evaluate benchmarks D and F. Workload D is a read-latest workload, but we already have a scenario with non-uniform data access through the zipfian distribution. Workload F reads and then modifies records, but KVell does not support these queries.

Each workload is executed three times and results are averaged. We issue 100M requests for workloads A, B and C and 20M requests for workload E when testing the SSD systems. For the HDD database, we issue 10M request for workloads A,B and C as well as 2M requests for workload E. This is to keep run times in check given the comparatively low throughput of HDDs.

We test RocksDB on the HDD (called `RocksHDD` hereafter), but also on `NVMe1` (called `RocksNVMe`) to see how a more traditional database design behaves in terms of power consumption when running on NVMe SSDs. Additionally we test KVell on the two NVMe devices introduced in Section II-B. Each database is allocated 30% of the dataset size as memory (30GB). RocksDB is configured to use a 10GB LRU cache, a bloom filter and a 64KB block size on HDD versus a 16KB block size on SSD. Also, the compaction readahead size for RocksHDD is set to 8MB. KVell is configured to use a page cache of 25GB.

B. Results

This section discusses the results obtained through the previous set of experiments. Table I presents a performance summary of all systems across all workloads. As expected, RocksHDD performs the worst, between 14 times (YCSB E, KVell) and 1233 times (YCSB C, KVell) slower than the other databases depending on the workload. In the rest of this section, we discuss the average power consumption of each system before factoring in the performance of the databases by computing the energy needed to serve a request.

Dist.	Database	YCSB A	YCSB B	YCSB C	YCSB E
Uniform	RocksHDD	1.8k	403	377	172
	RocksNVMe	63k	221k	243k	66k
	KVell/NVMe1	149k	357k	461k	11.5k
	KVell/NVMe2	260k	409k	465k	11.8k
Zipfian	RocksHDD	2.2k	1.6k	1.6k	274
	RocksNVMe	228k	750k	977k	155k
	KVell/NVMe1	256k	750k	1083k	22.2k
	KVell/NVMe2	466k	952k	1114k	23.0k

TABLE I: Throughput per second of each database for both uniform and zipfian request patterns for all workloads

1) Power consumption: To begin the analysis, we compare the average power consumption of the CPU, memory and storage device for each system under each workload. Results are presented in Figure 2.

The first major observation is that RocksHDD consumes between 30W and 95W less power than the other systems across all workloads. This is expected as the CPU is spending most of its time waiting for disk I/O, but it raises the question of how the power consumption evolves when all systems are queried with a throughput that RocksHDD can handle. To check this, we ran experiments on all systems with a constant number of requests per second low enough that RocksHDD could handle them (e.g. less than 377 on YCSB C). Results from these additional experiments again show that RocksHDD consumes less power than the SSD-based systems. Therefore, to optimize the lifecycle carbon footprint, if an HDD-based system can handle the load it should be preferred over an SSD-based alternative, assuming a lower SEF for the HDD.

Knowing that KVell was developed to avoid the CPU overhead that limits LSM-Tree-based KV stores such as RocksDB on NVMe SSDs, it is interesting to see that KVell consumes less power than RocksNVMe across all workloads and both with respect to the CPU (up to 35%), the memory (up to 25%) and the storage (up to 33%). This shows that employing a database adapted for modern NVMe SSDs is not only beneficial in terms of performance, but also in terms of power.

2) Energy per request: As we have seen in the previous section, because RocksHDD is I/O bottlenecked, its CPU and memory power consumption are much lower than that of RocksNVMe and KVell. Of course, its throughput is also orders of magnitude lower. To represent this, we calculate the energy required to serve a request by summing the total energy consumption over the duration of the workload and divide it by the throughput handled by each system. We present these results as energy per request in mJ in Table II.

Unsurprisingly, the NVMe-based systems consume orders of magnitude less energy per request than RocksHDD. For example, for uniform requests, RocksHDD consumes up to 88 times more energy per request for YCSB A and 743 times more for YCSB C. Indeed, workload C is the worst case for an HDD-based system, as it consists of only random reads.

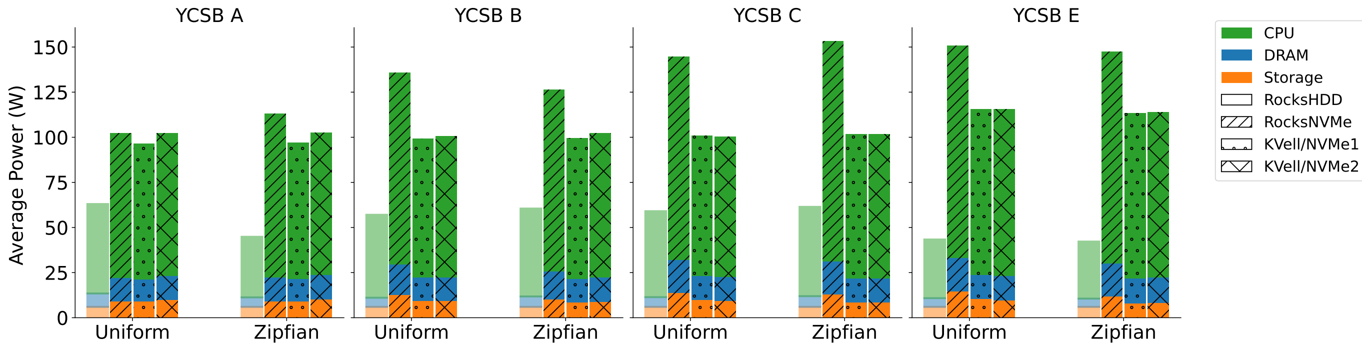


Fig. 2: Average power consumption of CPU, memory and storage for RocksHDD, RocksNVMe, KVell/NVMe1 and KVell/NVMe2 across workloads A, B, C and E for uniform and zipfian request distributions

	Workload (Uniform)				Workload (Zipfian)			
Database	A	B	C	E	A	B	C	E
RocksHDD	34.3	141	156	251	20.7	38.7	39.5	154
RocksNVMe	1.62	0.61	0.60	2.29	0.49	0.17	0.16	1.62
KVell/NVMe1	0.65	0.28	0.21	9.96	0.38	0.13	0.09	5.04
KVell/NVMe2	0.39	0.24	0.21	9.75	0.22	0.11	0.09	4.92

TABLE II: Energy per request (as sum of package, memory and storage) in mJ for each workload and database

Note that because of the mostly sequential accesses (and the relatively poor performance of KVell), the difference in energy is less pronounced for YCSB E.

For the zipfian access pattern, we observe the same trends, although the gap between RocksHDD and the SSD databases is a bit smaller. Indeed, because of the high access latency, RocksHDD benefits more from most accesses being in memory (around 90% under the tested configuration) than the SSDs.

IV. Comparing the carbon footprint of SSDs and HDDs

To compare the lifecycle carbon footprint of SSDs to that of HDDs, we need to link the embedded and operational emissions of both setups. When operating at high throughput, SSDs are more energy efficient. However, due to their higher embodied emissions, they need to produce lifetime operational emissions low enough to ultimately have lower lifecycle emissions than HDD-based systems. This introduces three additional factors: the lifetime of the devices, the work handled during that time, and the carbon intensity of the electricity powering the devices.

A. Lifecycle carbon emissions of storage devices

In a first step, we compare the lifecycle carbon emissions of KV stores running on the different storage devices, considering solely the emissions and power related to the storage devices themselves.

Equation 1 defines the condition for a KV store running on NVMe SSDs to have a lower carbon footprint than a KV store running on HDDs. In this equation, r is the number of handled requests (our variable), $E(DB_{NVMe,HDD})$ is the energy per request for NVMe/HDD based databases respectively, C is the carbon intensity of the electricity source, S is the size of the storage device and $SEF_{NVMe,HDD}$ are the storage embodied factors for NVMe and HDD. Here $E(DB_{NVMe,HDD})$ only includes the energy consumed by the storage device. It is deduced from the results we presented in Section III-B.

$$r * E(DB_{NVMe}) * C + S * SEF_{NVMe} < \\ r * E(DB_{HDD}) * C + S * SEF_{HDD} \quad (1)$$

The solution to this inequation indicates the minimum number of requests an SSD-based system needs to serve during its lifetime for its lifecycle carbon footprint to be lower than that of an HDD-based system. Taking into account the throughput of the database running on a NVMe SSD, we get the time the SSD needs to operate to outperform the HDD in terms of carbon footprint.

Figure 3 presents the results in number of days as a function of the carbon intensity of the electricity powering the datacenter. It shows results for RocksNVMe as well as KVell on `NVMe1` and `NVMe2`. As discussed in Section II-A the embodied carbon cost of hardware is difficult to determine and results can differ significantly depending on device model or data source [8]. Different SEF values have a strong impact on our analysis: the higher the embodied footprint of SSDs in relation to that of HDDs, the more the SSD will have to compensate during operation. Thus, we also evaluate each system with three different assumptions on the Storage Embodied Factor (SEF), using the numbers collected by Tannu et al. [8]: an average case (`ac`) where we take the average SEF for both SSD (0.16kG/GB CO2e) and HDD (0.02kG/GB CO2e), a best case for HDDs (`bc`) with the highest value for SSDs (0.34kG/GB CO2e) and the lowest for HDDs (0.005kG/GB CO2e) as well as a worst case (`wc`) with the lowest value for SSDs (0.033kG/GB CO2e) and the highest for HDDs (0.06kG/GB CO2e).

Figure 3 shows results for a uniform distribution of the requests for YCSB A, which is the workload where the performance gap (in terms of energy per request) between RocksHDD and the SSD-based databases is the smallest. To sim-

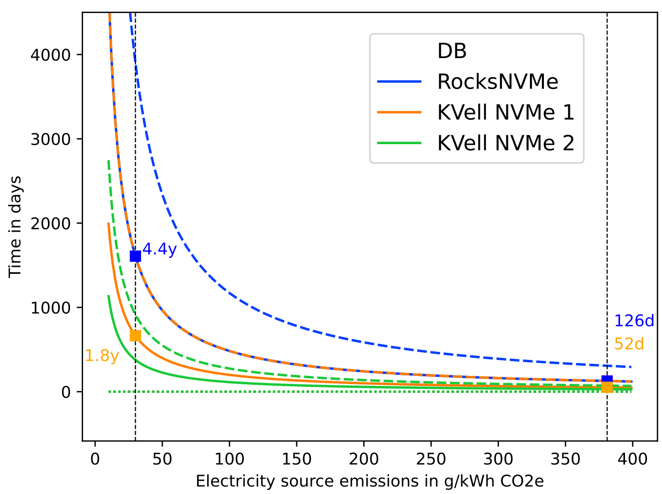


Fig. 3: Break-even point after which the lifecycle emissions of storage of RocksHDD surpass those of RocksNVMe and KVell as a function of the electricity carbon intensity, workload A, uniform requests, maximum throughput. The full, dashed and dotted lines represent the average, best and worst case assumptions about the SEF of storage respectively. Below the curve, the lifecycle emissions of RocksHDD are lower, above the emissions of the SSD database are lower

plify the reading, we make the results for two carbon-intensity mixes appear explicitly: the one of Germany (381g/kWh CO2e), a country with an ongoing transition to renewable energies, and the one of Norway (30g/kWh CO2e), one of the least carbon-intensive electricities worldwide. We also choose to focus on the uniform distribution because it stresses the storage device more than the zipfian distribution, where most queries will be served from memory.

A number of observations can be made on this plot. First, the carbon intensity of the datacenter electricity has a strong influence on the comparison between HDD and SSD. Indeed, the dirtier the energy source (ex. Germany), the quicker SSDs will pay off as the higher energy consumption per request of HDDs has a bigger impact when the electricity has high emissions. This is why for the example of Germany, amortization times for RocksNVMe and KVell are short. On the contrary, with a cleaner power source (ex. Norway), the operational energy consumption contributes less to the lifecycle carbon footprint. Therefore, SSD-based systems need to run for longer before they can compensate their embodied footprint and outperform RocksHDD in terms of lifecycle footprint. As can be seen, with Norwegian electricity KVell on `NVMe1` would have to run for 1.8 years before its lifecycle footprint becomes lower than that of RocksHDDs. Moreover, RocksNVMe would need to run for 4.4 years which introduces the question about device lifetimes: if an SSD-based system would need more time than what can be expected as lifetime to compensate its embodied footprint, it would not be beneficial to employ SSDs. We investigate this question in Section IV-B.

The general trend of the curves shows that the less carbon-intensive the electricity powering the datacenter is, the more time it takes for SSD-based systems to compensate their higher embodied footprint with a better energy efficiency during operation. Considering the increasing efforts to use cleaner energy to reduce the carbon footprint of datacenter operation [5], [27], [28], it implies that the question of the best storage device type will remain valid.

We can also observe on the figure that the different assumptions for the SEF of the devices can strongly impact the conclusions. In the considered best-case scenario (`bc` – dashed lines), that is, when the SEF of HDDs is lowest and that of SSDs is highest, the time needed for the SSD to become advantageous increases significantly compared to the average case. On the other hand, in the worst-case scenario (`wc` – dotted lines) SSDs would be unconditionally superior as their embodied footprint would be better than the one of HDDs. In the rest of the analysis, we will only consider the average SEF values for clarity, but we need to keep in mind that different SEF estimates could lead to different conclusions. Perhaps surprising is that the operational differences between two SSD models barely influence the results. NVMe 2 delivers on average 20-30% higher throughput than NVMe 1 with a similar power consumption, yet their amortization times are in the same order of magnitude. This hints that when reasoning about lifecycle carbon footprints of SSDs, the performance is less important than the embodied footprint.

We omit results for workloads B, C and E because the plots follow the same trends as workload A. Workloads B and C are worst-case scenarios for RocksHDD as they consist of almost exclusively random reads. The SSD-based systems are superior here as they are not negatively impacted by such a workload, and on the contrary perform better on read queries. For YCSB E, plotting the results would not give any additional insights compared to YCSB A. Yet, interestingly, despite RocksHDD performing relatively better on this workload compared to the other databases, the energy per request is higher for all systems. Therefore, the proportion of the operational emissions increases, which leads to lower amortization times for SSDs than what we observed for YCSB A.

The plots for the zipfian request distribution again follow the same pattern as Figure 3, with overall slightly higher amortization times because of the reduced performance gap between HDD and SSD compared to the uniform accesses. Still, it is difficult to interpret these results as in this configuration, most queries will be handled from memory. Thus, in the following we will continue to focus on the uniform distribution.

While useful to get an initial comparison between SSD and HDD, the approach presented in this section has some limitations. First, it does not take into account the full power consumption of the system (CPU, memory) and only considers storage. More importantly, it ignores the fact that to accomplish the same amount of work as an SSD running at maximum throughput over its lifetime, an HDD would either have to run for much longer, well beyond its expected lifetime, or we would have to add more HDDs to run in parallel. On the other hand, given that we used the numbers of Section III as base here, the SSD-based systems were assumed to always run at full capacity, which is probably almost always unrealistic. Therefore, in the next section, we follow a different approach to consider more realistic scenarios.

B. Lifecycle carbon emissions of storage systems

In this section, we run an analysis taking into account the power consumption of not only the storage devices but also of the CPU and memory resources used by the KV store. Furthermore, we want to find *realistic* configurations in which HDD-based systems could have a lower lifecycle carbon footprint than SSD-based ones. When SSDs run at maximum throughput, HDD-based systems cannot compete in terms of performance. However, when considering cases where the bandwidth of SSDs is not fully used, we can envision scenarios where multiple HDDs, typically in a RAID-0 configuration [34], would be used instead of a single SSD. We now explore such configurations.

Equation 1 needs to be updated to take into account our new assumptions. Namely, we introduce Equation 2, where H represents the number of HDDs. We obtain H with Equation 3 where T is the target throughput, $T(DB_{HDD})$ is the throughput of RocksHDD and sc is the scaling factor to estimate the performance of a multi-drive database. We introduce sc to take into account that it is difficult to anticipate the exact scaling of a KV store when running on top of multiple disks, as it depends very much on the access patterns. We run our analysis with different values for sc.

Note also that in Equation 2, $E(DB_{NVMe,HDD})$ has been replaced by $E_{full}(DB_{NVMe,HDD})$ to represent the power consumption of the CPU and the memory of our server, in addition to the power consumption of the storage device.

$$r * E_{full}(DB_{NVMe}) * C + S * SEF_{NVMe} < r * E_{full}(DB_{HDD}) * C + S * SEF_{HDD} * H \quad (2)$$

$$H = \lceil \frac{T}{T(DB_{HDD}) * sc} \rceil \quad (3)$$

1) Instantiating the model: To run an analysis under our new assumptions, we need to collect data for the case where an NVMe-based KV store runs below its max performance. The results presented in Section IV-A show that results with RocksNVMe and the two instances of KVell are qualitatively the same. Hence, to simplify the analysis, we only consider KVell running on `NVMe1` in the following. Furthermore, we focus on workloads A and E as RocksHDD performs best on these two workloads, and our goal is to identify whether there are cases where a system based on HDDs would perform better with respect to the carbon footprint than a system based on SSDs. Because estimating the performance of a RAID system under a zipfian access pattern is difficult and we did not have the hardware to perform the necessary benchmarks, we will only discuss a uniform request distribution.

Here is the configuration that we consider. First, since we are not running at maximum performance, we assume a configuration where the KV store uses only 10% of the database size as memory, instead of the 30% used in Section III. Second, we assume that KVell processes on average 4000 requests per second for YCSB A and 400 requests per second for YCSB E. This corresponds to 7.5% and 20% of KVell's maximum throughput with 10% memory. These numbers are rather low. Hence, if we cannot find a configuration where an HDD-based solution has a better lifecycle carbon footprint than an SSD-based solution, it means that SSDs should always be used to minimize the carbon footprint of KV stores.

We should mention that based on our measurements, using a single HDD and 10% of memory, RocksHDD achieves a throughput of 722 requests per second for YCSB A and 112 requests per second for YCSB E. Hence, assuming a perfect scaling, 6 HDDs are needed for RocksHDD to handle the 4000 requests per second for YCSB A, and 4 HDDs are needed to handle 400 requests per second for YCSB E.

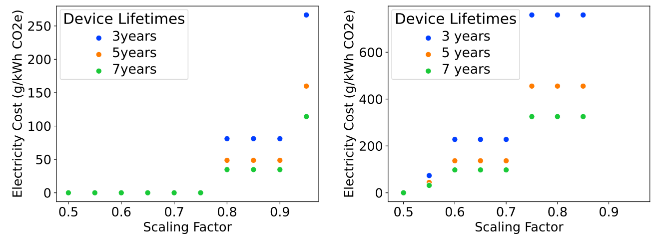
Finally, regarding the embodied carbon cost, in the following we consider the average case discussed in Section IV-A, where the SEF of an SSD is 8 times higher than that of an HDD.

2) Results with uniform workloads: Results for the solutions of Equation 2 are presented in Figure 4. Considering different lifetimes for the devices (coded by color), the figure presents the maximum carbon intensity of the electricity powering the databases for which RocksHDD would have a lower lifecycle carbon footprint than KVell according to the scalability factor of RocksHDD when running with multiple HDDs. Figures 4a and 4b show the results for YCSB A and YCSB E respectively.

To illustrate the results, we consider the case of YCSB A assuming a device lifetime of 5 years and a scaling factor of 0.8. Figure 4a show that in this case, if the database is powered with an electricity emitting 48 g/kWh CO2e or less, the lifecycle carbon footprint of RocksHDD would be lower than that of KVell. If the electricity was more carbon intensive, KVell's lifecycle carbon emissions would be lower.

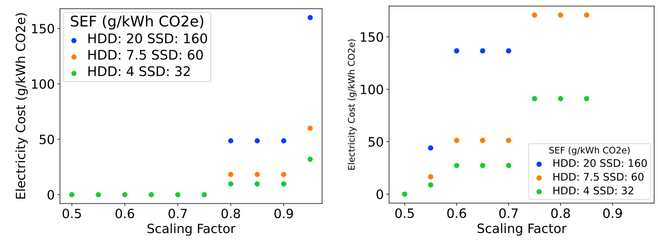
Overall, both workloads exhibit a similar pattern. Whenever a lower scaling factor implies that a new HDD needs to be added to reach the target performance (e.g. from sc 0.95 to 0.9 on YCSB A), the embodied carbon of RocksHDD increases and as a consequence, the electricity needs to be less carbon intensive for RocksHDD to keep a lower carbon footprint than KVell. As soon as 8 or more HDDs are needed ($sc <$ 0.8 for YCSB A), the embodied cost of RocksHDD is the same (or higher) as that of KVell running on a single NVMe, and KVell will have a lower lifecycle footprint, no matter the carbon intensity of the electricity. Taking again the example of Norway's electricity carbon intensity (30 g/kWh CO2e), and assuming a scaling factor of 0.8, RocksHDD would be the better choice for YCSB A even with an expected storage device lifetime of 7 years. This suggests that when low-carbon electricity is available, there are indeed scenarios where employing more HDDs instead of upgrading to an SSD would be better in terms of lifecycle carbon footprint.

Results for YCSB E need to be taken with care, mostly because the performance scaling is harder to predict for scan workloads and it is expected to be lower than for point lookup workloads. The 400 requests per second mean that we need 5 or fewer drives for a scaling factor of 0.75 or higher. This results in KVell relying on very carbon intensive energy to



(a) YCSB A, 4000 requests per second (b) YCSB E, 400 requests per second

Fig. 4: Carbon intensity of electricity thresholds below which RocksHDD's lifecycle carbon footprint is lower than KVell's as a function of the scaling factor applied for the multiple HDDs of RocksHDD, comparing different device lifetime assumptions



(a) YCSB A, 4000 requests per second (b) YCSB E, 400 requests per second

Fig. 5: Carbon intensity of electricity thresholds below which RocksHDD's lifecycle carbon footprint is lower than KVell's as a function of the scaling factor applied for the multiple HDDs of RocksHDD, comparing different SEF assumptions

compensate its embodied emissions or not being able to do so at all in case of a scaling of 0.9 or higher. Yet, it is not clear if such a scaling is realistic. Moreover, while 400 requests correspond to 20% of KVell's maximum throughput with 10% memory, Section III showed that RocksNVMe delivers much better performance on YCSB E and is thus better suited for this type of workload.

The key takeaways of this analysis are that first, a more carbon intensive electricity favors SSDs, in line with Section IV-A. Second, long device lifetimes ($>$ 5 years) also favor SSDs, keeping in mind that storage device lifetimes in datacenters are not well studied [7]. Ultimately, unless certain conditions are fulfilled (low carbon electricity, short device lifetimes), the results suggest that SSD-based systems will have a lower lifecycle carbon footprint.

3) The Impact of SEF Values: Section IV-A showed that the SEF values impact the comparison of the systems. In that section, we considered the case of different ratios between the SEF of SSDs and HDDs. Yet, another phenomenon could impact the SEF of storage devices. As per the results presented in [29], the SEF of SSDs and HDDs tends to decrease as manufacturers improve their production process.

To study the impact of this point, Figure 5 presents the same results as Figure 4, assuming a device lifetime of 5 years and still assuming that SSDs have an 8 times higher SEF than HDDs, but assuming different values of SEF for the devices.

As can be seen, lower SEF values favor the SSD-based systems. This is because a lower SEF implies that the proportion of the embodied footprint in the lifecycle footprint will be lower. Therefore, since the SSD-based systems have a lower operational footprint, they become better overall.

C. Discussion

The results presented in this section tend to show that, despite their higher embodied carbon emissions, SSDs should be preferred over HDDs to reduce the overall carbon footprint of datacenters. Only in cases where SSDs would be far from being used at their full potential, and where the carbon intensity of the electricity powering the datacenters would be very low, could a solution based on HDDs be beneficial. However, our analysis shows that it is a complex problem where a large set of factors need to be taken into account to draw accurate conclusions.

Of course, several questions remain open at the end of this study. We list a few of them in the following. First, we only considered the case of KV stores. It would be important to understand how different the results would be for other types of databases. Second, the results we obtained heavily depend on the SEF considered for the different devices. Existing studies show that the values vary dramatically depending on the source (*e.g.*, storage device manufacturers vs hardware vendors) providing the information [8], [29]. More research is required in this domain to obtain reliable numbers.

Our study assumes that the lifetime of SSDs and HDDs is the same. This is probably not true. However, we miss data regarding this matter since the common approach in datacenters is to replace storage devices before they get too old [7]. Since extending the lifetime of devices is a simple solution to reduce the embodied footprint of datacenters, we should see more research in this direction in the coming years.

Finally, it should be noted that the systems considered in this study have been designed with the goal of optimizing performance but not energy consumption. Alternative solutions, maybe still to be designed, might be able to achieve better energy efficiency both for HDD and SSD-based systems. Such solutions might involve designing new algorithms but also adapting the hardware by using less power-hungry processors for storage servers or running processors at lower frequency.

V. Conclusion

This paper studies the carbon footprint of storage in datacenters, comparing the benefits of using SSDs and HDDs. Considering the case of KV stores as storage engine, we present an extensive set of experiments with state-of-the-art solutions to evaluate the power consumption and energy efficiency of systems based on the two types of devices. Using specialized wattmeters and considering different representative workloads, we show that the energy consumption of SSDs is complex and can be much higher than that of HDDs. However, because of their better performance, using NVMe SSDs leads to better energy efficiency, especially if we consider the total power consumed by storage servers, including the power consumed by the CPU and the memory.

Using the obtained data, we ran analyses about the overall carbon footprint of such storage systems, taking into account both their embodied and operational carbon footprint. Our results show that SSDs allow achieving a lower carbon footprint than HDDs in most cases. The much higher operational efficiency of NVMe SSDs allows them to compensate for their higher embodied emissions. Only in cases where the electricity powering the datacenter would have a very low carbon intensity and where the full throughput potential of SSDs would not be used, could HDDs provide some advantages.

As our analysis shows, the question tackled is the paper is complex. It was impossible for us to take all the parameters that might influence the results into account. More research on this topic should be conducted, for instance, to understand if the presented results are specific to KV stores, or if they remain valid no matter the storage engine.

Acknowledgments

We would like to thank Pierre Neyron for his invaluable support in the setup of the experiments. We also thank Quarch Technology for lending us their Power Analysis Module. Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see https://www.grid5000.fr).

References

[1] C. Freitag, M. Berners-Lee, K. Widdicks, B. Knowles, G. S. Blair, and A. Friday, "The real climate and transformative impact of ict: A critique of estimates, trends, and regulations," *Patterns*, vol. 2, no. 9, 2021.

[2] L. A. Barroso, U. Hölzle, and P. Ranganathan, *The datacenter as a computer: Designing warehouse-scale machines*. Springer Nature, 2019.

[3] E. Masanet, A. Shehabi, N. Lei, S. Smith, and J. Koomey, "Recalibrating global data center energy-use estimates," *Science*, vol. 367, no. 6481, pp. 984–986, 2020.

[4] J. Lyu, J. Wang, K. Frost, C. Zhang, C. Irvene, E. Choukse, R. Fonseca, R. Bianchini, F. Kazhamiaka, and D. S. Berger, "Myths and misconceptions around reducing carbon embedded in cloud platforms," in *2nd Workshop on Sustainable Computer Systems*, 2023, pp. 1–7.

[5] U. Gupta, Y. G. Kim, S. Lee, J. Tse, H.-H. S. Lee, G.-Y. Wei, D. Brooks, and C.-J. Wu, "Chasing carbon: The elusive environmental footprint of computing," in *IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 2021, pp. 854–867.

[6] J. Wang, D. S. Berger, F. Kazhamiaka, C. Irvene, C. Zhang, E. Choukse, K. Frost, R. Fonseca, B. Warrier, C. Bansal *et al.*, "Designing cloud servers for lower carbon," in *ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA)*, 2024, pp. 452–470.

[7] S. McAllister, F. Kazhamiaka, D. S. Berger, R. Fonseca, K. Frost, A. Ogus, M. Sah, R. Bianchini, G. Amvrosiadis, N. Beckmann *et al.*, "A call for research on storage emissions," in *Proceedings of the 3rd Workshop on Sustainable Computer Systems (HotCarbon)*, 2024.

[8] S. Tannu and P. J. Nair, "The dirty secret of ssds: Embodied carbon," *ACM SIGENERGY Energy Informatics Review*, vol. 3, no. 3, 2023.

[9] Q. Xu, H. Siyamwala, M. Ghosh, T. Suri, M. Awasthi, Z. Guz, A. Shayesteh, and V. Balakrishnan, "Performance analysis of nvme ssds and their implication on real world databases," in *Proceedings of the 8th ACM International Systems and Storage Conference*, 2015, pp. 1–11.

[10] M. Kleppmann, *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems.* " O'Reilly Media, Inc.", 2017.

[11] Z. Cao, S. Dong, S. Vemuri, and D. H. Du, "Characterizing, modeling, and benchmarking RocksDB Key-Value workloads at facebook," in *18th USENIX Conference on File and Storage Technologies*, 2020.

[12] J. Yang, Y. Yue, and K. Rashmi, "A large-scale analysis of hundreds of in-memory key-value cache clusters at twitter," *ACM Transactions on Storage (TOS)*, vol. 17, no. 3, pp. 1–35, 2021.

[13] C. Lai, S. Jiang, L. Yang, S. Lin, G. Sun, Z. Hou, C. Cui, and J. Cong, "Atlas: Baidu's key-value storage system for cloud data," in *31st Symposium on Mass Storage Systems and Technologies*, 2015.

[14] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, "Bigtable: A distributed storage system for structured data," *ACM Transactions on Computer Systems (TOCS)*, vol. 26, no. 2, pp. 1–26, 2008.

[15] B. Lepers, O. Balmau, K. Gupta, and W. Zwaenepoel, "Kvell: the design and implementation of a fast persistent key-value store," in *27th ACM Symposium on Operating Systems Principles*, 2019, pp. 447–461.

[16] A. Conway, A. Gupta, V. Chidambaram, M. Farach-Colton, R. Spillane, A. Tai, and R. Johnson, "SplinterDB: closing the bandwidth gap for NVMe Key-Value stores," in *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, 2020, pp. 49–63.

[17] B. F. Cooper, A. Silberstein, E. Tam, R. Ramakrishnan, and R. Sears, "Benchmarking cloud serving systems with ycsb," in *Proceedings of the 1st ACM symposium on Cloud computing*, 2010, pp. 143–154.

[18] K. N. Khan, M. Hirki, T. Niemi, J. K. Nurminen, and Z. Ou, "Rapl in action: Experiences in using rapl for power measurements," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems (TOMPECS)*, vol. 3, no. 2, pp. 1–26, 2018.

[19] B. Harris and N. Altiparmak, "Ultra-low latency SSDs'impact on overall energy efficiency," in *12th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage 20)*, 2020.

[20] J. Ousterhout and F. Douglis, "Beating the i/o bottleneck: A case for log-structured file systems," *ACM SIGOPS Operating Systems Review*, vol. 23, no. 1, pp. 11–28, 1989.

[21] B. Knowles, "Acm techbrief: Computing and climate change," 2021.

[22] D. Pandey, M. Agrawal, and J. S. Pandey, "Carbon footprint: current methods of estimation," *Environmental monitoring and assessment*, vol. 178, pp. 135–160, 2011.

[23] J. F. Green, "Private standards in the climate regime: the greenhouse gas protocol," *Business and Politics*, vol. 12, no. 3, pp. 1–37, 2010.

[24] C. Zhang, A. G. Kumbhare, I. Manousakis, D. Zhang, P. A. Misra, R. Assis, K. Woolcock, N. Mahalingam, B. Warrier, D. Gauthier *et al.*, "Flex: High-availability datacenters with zero reserved power," in *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*. IEEE, 2021, pp. 319–332.

[25] A. G. Kumbhare, R. Azimi, I. Manousakis, A. Bonde, F. Frujeri, N. Mahalingam, P. A. Misra, S. A. Javadi, B. Schroeder, M. Fontoura *et al.*, "Prediction-Based power oversubscription in cloud platforms," in *2021 USENIX Annual Technical Conference*, pp. 473–487.

[26] Q. Wu, Q. Deng, L. Ganesh, C.-H. Hsu, Y. Jin, S. Kumar, B. Li, J. Meza, and Y. J. Song, "Dynamo: Facebook's data center-wide power management system," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 3, pp. 469–480, 2016.

[27] B. Acun, B. Lee, F. Kazhamiaka, K. Maeng, U. Gupta, M. Chakkaravarthy, D. Brooks, and C.-J. Wu, "Carbon explorer: A holistic framework for designing carbon aware datacenters," in *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, 2023, pp. 118–132.

[28] A. A. Chien, C. Zhang, and L. Lin, "Beyond pue: Flexible datacenters empowering the cloud to decarbonize," *USENIX Hot Carbon*, 2022.

[29] U. Gupta, M. Elgamal, G. Hills, G.-Y. Wei, H.-H. S. Lee, D. Brooks, and C.-J. Wu, "Act: Designing sustainable computer systems with an architectural carbon modeling tool," in *Proceedings of the 49th Annual International Symposium on Computer Architecture*, 2022, pp. 784–799.

[30] S. Park, Y. Kim, B. Urgaonkar, J. Lee, and E. Seo, "A comprehensive study of energy efficiency and performance of flash-based ssd," *Journal of Systems Architecture*, vol. 57, no. 4, pp. 354–365, 2011.

[31] S. Shin and D. Shin, "Power consumption characterization of flash memory ssd," in *ICEIC: International Conference on Electronics, Informations and Communications*, 2010, pp. 14–18.

[32] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, "Dynamo: Amazon's highly available key-value store," *ACM SIGOPS operating systems review*, vol. 41, no. 6, pp. 205–220, 2007.

[33] B. Atikoglu, Y. Xu, E. Frachtenberg, S. Jiang, and M. Paleczny, "Workload analysis of a large-scale key-value store," in *12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems*, 2012, pp. 53–64.

[34] D. A. Patterson, G. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks (raid)," in *Proceedings of the 1988 ACM SIGMOD international conference on Management of data*, 1988, pp. 109–116.