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

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

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

得益于 NVMe SSD 的高吞吐量和极低的延迟，它成为了现代数据中心的主要部分，但 NVMe SSD 在大规模部署下的可靠性依然未知。这篇文章收集了超过一百万数量的被部署的 NVMe SSD 的日志并进行了大范围数据分析，并依此确定了 NVMe SSD 的数个主要可靠性变化。好的方面是，NVMe SSD 面对早期的失效以及多样的访问模式时更具有灵活性。坏的方面是，NVMe SSD 面对复杂且交错的失效时更加脆弱，更关键的是，文章中发现极低的延迟特性使得 NVMe SSD 更可能被 fail-slow failure 所影响。

介绍

NVMe SSD 的带宽可以到达 6GB/s，并且做到微秒级的延迟。相比于其他 SATA-based 的存储设备，这提供了显著的性能提升。

但在硬件设备被大范围使用的情况下，除去性能以外，可靠性也是关键的一环。尽管关于 SATA SSD 的失效特性的研究众多，但并不能直接把它们套用到 NVMe SSD 上。

首先，NVMe SSD 由于其低延迟的特性，更容易受到 fail-slow failure 的影响，表现出异常的性能下降，反观 SATA SSD，它们的 fail-slow failure 所造成的影响很可能被其相对较高的延迟所掩盖。

除此之外，NVMe SSD 也不仅仅是 SATA SSD 基础上的接口升级，它的内在结构也有了相当的变化，这些变化致力于提高其性能，然而由于缺少关于大规模的 NVMe SSD 的 fail-stop 的研究，这些结构升级带来的具体提升依然未知。

在这篇文章中，作者研究了 NVMe SSD 的 fail-stop 和 fail-slow failures。他们绘制并且分析了 NVMe SSD 的基本特性，然后梳理了不同影响因子带来的数据集差异。最后，他们针对 fail-slow 严格地确定了其成因。最后，他们得到了了 10 个发现并列举了数个关键发现：

1、NVMe SSD 的早期错误发生率并不突出，前三个月的失效率几乎等同于后期，甚至更低。

2、高写放大因素不再和失效紧密相关，低写放大的 NVMe SSD 的 ARR(年替换率)比高写放大的 NVMe SSD 大 2.19 倍。

3、使用协同定址的 NVMe SSD 的失效更具有时间相关性。

4、fail-slow 对 NVMe SSD 来说是普遍且严重的问题，并且这一失效会让其表现下降到 SATA SSD 甚至 HDD 级别。

5、fail-slow 失效与其 SMART 属性并无联系，并且很少转换为 fail-stop 失效。

背景

文章中使用到的 NVMe SSD 来自全球的多个网络数据中心，候选的 SSD 都是企业级的，驱动模型最早在 2015 年 5 月左右，最新的在 2019 年 7 月。尽管不同的 SSD 负责的任务可能不同，对于工作负担的影响的研究是在控制变量实验中进行的。

实验中收集到的数据的时期如图 1

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

图 1:数据收集时期

SMART Logs 是一组被开放商和管理人员们认同的用于评估驱动的可靠性和性能的属性。实验中该数据每日收集一次。

Perfromance Logs 由一部分配备了节点级守护进程的 SSD 所提供，它们记录了 Linux 内核性能日志 iosata。这一守护进程每日运行三小时，每 15 秒记录一次平均数据。

Failure Tickets 是由每个节点所配备的守护进程所提供的，它们记录了 fail-stop 失效。在报告时，生成一个 failure ticket 后，这一 ticket 会被工程师人工检查。

为了确保实验方法的正确性，在整个研究中必须遵守三条原则：

1、研究起始于总的大范围的比较，并由此确定突出明显的因素。如果某一高层的观察没有成效或令人感到可疑，那么就要进行细致的变量控制实验来发掘潜在的根本原因并给研究人员提供可行的建议。

2、对原始数据进行预筛选来避免极端数据导致的偏差。要求工程师对 failure ticket 进行人工检查就是为此。同时也排除了部分使用较少的驱动模型，并将不同供应商起了不同名字的相同模型视为一个模型来对待。

3、对于统计工具的选择标准，要么先前的研究中使用过这种方法，要么有明确的文件表明这一方法可以被用于指定的情景下。

基础数据

数据总览

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

图 2:驱动模型的基础数据

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

图 3:失效情况分布

高层次的比较

根据图 2 和图 3，我们可以得到一系列区别：

1、NVMe SSD 相比较于 SATA/SAS SSD 有更高的 ARR，且 NVMe SSD 的 I/O 错误占据了首要位置，而在 SATA/SAS SSD 中占主要位置的 Lost 错误在 NVMe SSD 中并不显著。

2、P/E error 的数量和 ARR 与驱动容量成正相关，因为大容量的驱动更可能被访问，因此出现错误的可能性更高。

3、NAND type 是 3D-TLC 的 NVMe SSD 的 ARR 比使用 MLC 的 NVMe SSD 的 ARR 稍微低一些，但在 SATA SSD 中，这正好相反。

之后，基于在 SATA/SAS SSD 中已存在的模板，作者研究了 NVMe SSD 的不同之处

1、NVMe SSD 的早期错误发生率（infant mortality）并不突出，前三个月的失效率几乎等同于后期，甚至更低，如图 4。而 SATA SSD 是遵从 bathtub curve 的。

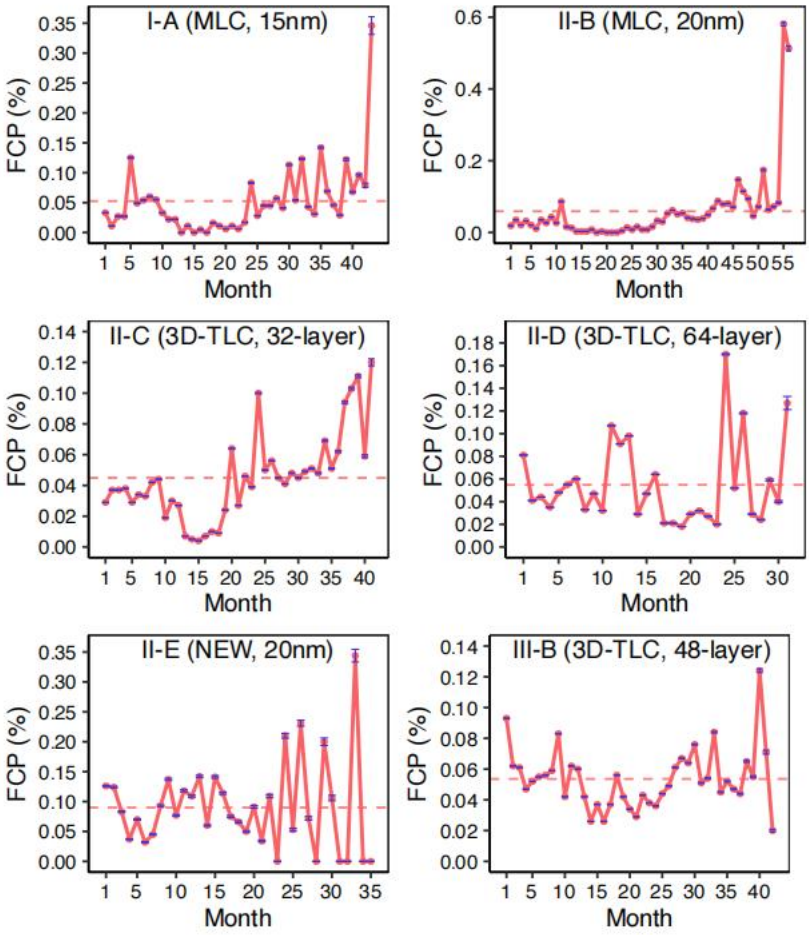

图 4:失效走向

进一步对 NVMe SSD 内部 SMART 属性的研究表明这些属性实际上还是会受到 infant mortality 的影响，如图 5。作者推测是由于错误处理部件的改进使得 NVMe SSD 在早期阶段更有弹性。

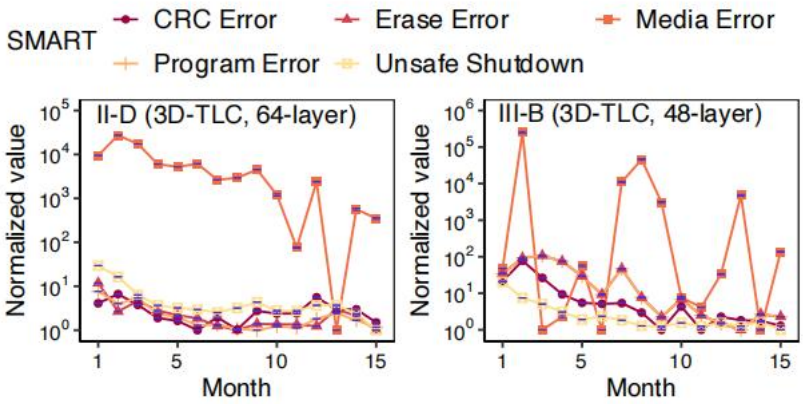

图 5:早期阶段的 SMART

2、NVMe SSD 对于高写放大更加稳健，但极度的低写放大依然是罕见但致命的，如图 6。

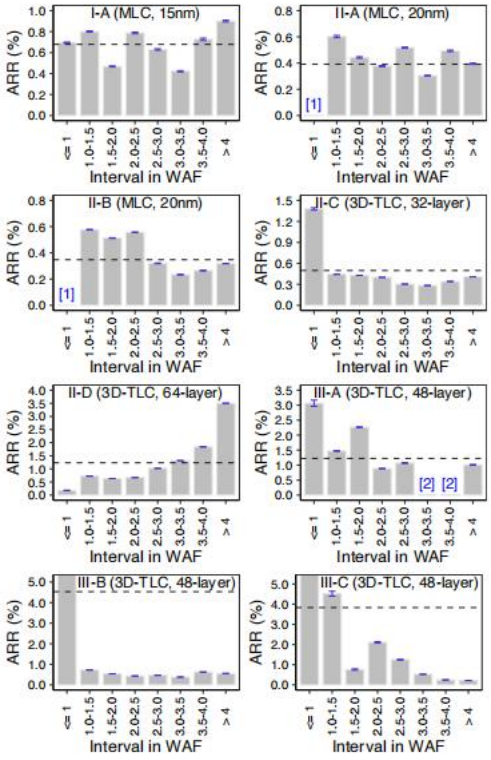

图 6:不同 WAF 级别下的 ARR

3、空间相关的 NVMe SSD(intra-node/rack)的故障在长期(1 天到 1 个月)内更具时间相关性，但短期内并不普遍，如图 7，在。

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

图 7:不同驱动器类型的 intra-node/rack 故障分布

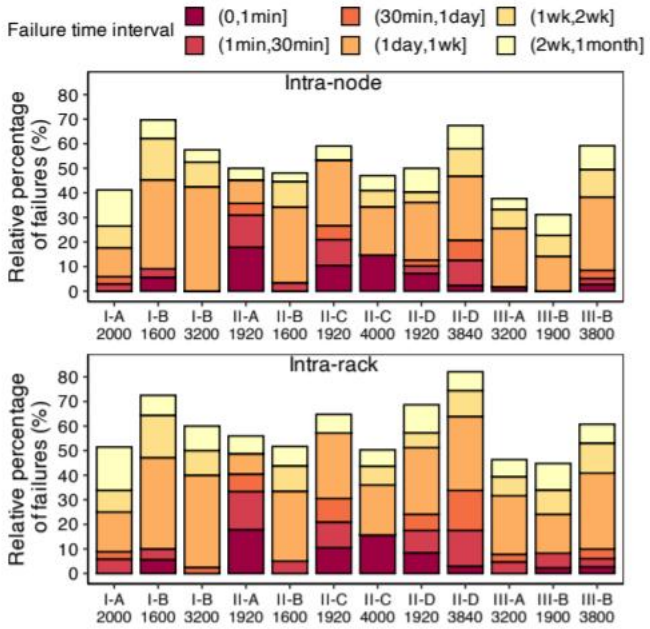

图 8:不同驱动器模型的 intra-node/rack 故障分布

4、相比较于 HDD，NVMe SSD 的 fail-slow 故障更加普遍和频繁，并且会把 NVMe SSD 降级到 SATA 甚至 HDD 级别。

5、不同开放商对于 NVMe SSD 的 fail-slow 失效率影响巨大。

6、发生 fail-slow 的驱动较多时，并不总是导致更高的事件 fail-slow 频率。原因可能是因为在短期情况下，有更多的驱动同时执行多个事件，导致即使驱动较慢，但事件重发率更高。

7、驱动和时间频率的 fail-slow 数据与驱动使用年龄有紧密联系，但只对于使用时间长的 NVMe SSD(使用超过 41 个月)来说是如此。

8、工作量会严重影响多种 fail-slow 特性，大交通量可能会对 fail-slow 的发生率产生长期影响，如图 9，可见 fail-slow 更容易在 buffering 工作中出现，而在实践中，buffer 工作通常带来不断的大交通量，因此推断可能交通量过大会产生长期影响。

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

图 9:不同工作负担分组情况下的 fail-slow 数据

9、SMART 属性与 fail-slow 的联系可以忽略不计。

10、fail-slow 在短期内不大可能转化为 fail-stop 失效，如图 10，在 fail-stop(Replaced)出现前曾出现过 fail-slow(Slow)的数据只有 10 个，少于 0.01%，推测可能是因为 fail-slow 很少转化成 fail-stop 或者需要相当长一段时间来完成转化。

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

图 10:由 fail-slow 向 fail-stop 失效的转化

总结

这篇文章对于 NVMe SSD 的失效进行了大规模的研究，文章中确定了在时间相关和 WAF 情况下的 fail-stop 的包括故障情况、稳健性等在内的故障模式的几个主要差别。文章中也按比例对 fail-slow 故障和影响因素进行了研究。最后，文章中总结了十条发现。