REFERENCES

[1] Accessed: 2024-03-13. 3D XPoint. https://insidehpc.com/2015/07/intel-and-micron-announce-3d-xpoint-non-volatile-memory/

[2] Accessed: 2024-03-13. BFQ Budget Fair Queueing Document. https://www.kernel.org/doc/html/latest/block/bfq-iosched.html

[3] Accessed: 2024-03-13. fio. https://github.com/axboe/fio

[4] Accessed: 2024-03-13. Intel Optane 900P Techinical Specification. https://www.intel.com/content/www/us/en/products/sku/123623/intel-optane-ssd-900p-series-280gb-2-5in-pcie-x4-20nm-3d-xpoint/specifications.html

[5] Accessed: 2024-03-13. Intel® Optane™ SSD DC P5800X Series. https://ark.intel.com/content/www/us/en/ark/products/201859/intel-optane-ssd-dc-p5800x-series-1-6tb-2-5in-pcie-x4-3d-xpoint.html

[6] Accessed: 2024-03-13. Kyber Multiqueue I/O Scheduler. https://lwn.net/Articles/720071/

[7] Accessed: 2024-03-13. Linux I/O Schedulers. https://wiki.ubuntu.com/Kernel/Reference/IOSchedulers

[8] Accessed: 2024-03-13. MQ-Deadline Implementation. https://elixir.bootlin.com/linux/latest/source/block/mq-deadline.c

[9] Accessed: 2024-03-13. The New ext4 Filesystem: Current Status and Future Plans. https://www.kernel.org/doc/ols/2007/ols2007v2-pages-21-34.pdf

[10] Accessed: 2024-03-13. NVM Express. https://nvmexpress.org

[11] Accessed: 2024-03-13. Samsung 980 PRO PCIe 4.0 SSD. https://semiconductor.samsung.com/consumer-storage/internal-ssd/980pro/

[12] Accessed: 2024-03-13. Toshiba Memory Introduces XL-FLASH Storage Class Memory Solution. https://americas.kioxia.com/en-us/business/news/2019/memory-20190805-1.html

[13] Accessed: 2024-03-13. Two New Block I/O Schedulers for 4.12. https://lwn.net/Articles/720675/

[14] Jens Axboe. Accessed: 2024-03-13. Efficient I/O with io_uring. https://kernel.dk/io_uring.pdf

[15] Alan J. Demers, Srinivasan Keshav, and Scott Shenker. 1989. Analysis and Simulation of a Fair Queueing Algorithm. In *Proceedings of the ACM Symposium on Communications Architectures & Protocols, SIGCOMM 1989*. ACM, 1–12.

[16] Diego Didona, Nikolas Ioannou, Radu Stoica, and Kornilios Kourtis. 2020. Toward a Better Understanding and Evaluation of Tree Structures on Flash SSDs. *Proc. VLDB Endow.* 14, 3 (2020), 364–377.

[17] Nima Elyasi, Mohammad Arjomand, Anand Sivasubramaniam, Mahmut T. Kandemir, Chita R. Das, and Myoungsoo Jung. 2017. Exploiting Intra-Request Slack to Improve SSD Performance. In *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS 2017*. ACM, 375–388.

[18] Pawan Goyal, Harrick M. Vin, and Haichen Cheng. 1996. Start-Time Fair Queueing: A Scheduling Algorithm for Integrated Services Packet Switching Networks. In *Proceedings of the ACM SIGCOMM 1996 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, 1996*. ACM, 157–168.

[19] Jiayang Guo, Yimin Hu, and Bo Mao. 2015. Enhancing I/O Scheduler Performance by Exploiting Internal Parallelism of SSDs. In *Algorithms and Architectures for Parallel Processing - 15th International Conference, ICA3PP 2015. Proceedings, Part IV (Lecture Notes in Computer Science, Vol. 9531)*. Springer, 118–130.

[20] Jiayang Guo, Yiming Hu, Bo Mao, and Suzhen Wu. 2017. Parallelism and Garbage Collection Aware I/O Scheduler with Improved SSD Performance. In *2017 IEEE International Parallel and Distributed Processing Symposium, IPDPS 2017*. IEEE Computer Society, 1184–1193.

[21] Mohammad Hedayati, Kai Shen, Michael L. Scott, and Mike Marty. 2019. Multi-Queue Fair Queuing. In *2019 USENIX Annual Technical Conference, USENIX ATC 2019*. USENIX Association, 301–314.

[22] Christoph Hellwig. 2009. XFS: The Big Storage File System for Linux. *login Usenix Mag.* 34, 5 (2009).

[23] Tejun Heo, Dan Schatzberg, Andrew Newell, Song Liu, Saravanan Dhakshinamurthy, Iyswarya Narayanan, Josef Bacik, Chris Mason, Chunqiang Tang, and Dimitrios Skarlatos. 2022. IOCost: Block IO Control for Containers in Datacenters. In *ASPLOS '22: 27th ACM International Conference on Architectural Support for Programming Languages and Operating System 2022*. ACM, 595–608.

[24] Jaehyun Hwang, Midhul Vuppalapati, Simon Peter, and Rachit Agarwal. 2021. Rearchitecting Linux Storage Stack for μs Latency and High Throughput. In *15th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2021*. USENIX Association, 113–128.

[25] Kanchan Joshi, Kaushal Yadav, and Praval Choudhary. 2017. Enabling NVMe WRR Support in Linux Block Layer. In *9th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage 17)*. USENIX Association.

[26] Myoungsoo Jung, Wonil Choi, Shekhar Srikantaiah, Joonhyuk Yoo, and Mahmut T. Kandemir. 2014. HIOS: A Host Interface I/O Scheduler for Solid State Disks. In *ACM/IEEE 41st International Symposium on Computer Architecture, ISCA 2014*. IEEE Computer Society, 289–300.

[27] Jieun Kim, Dohyun Kim, and Youjip Won. 2022. Fair I/O Scheduler for Alleviating Read/Write Interference by Forced Unit Access in Flash Memory. In *HotStorage '22: 14th ACM Workshop on Hot Topics in Storage and File Systems, 2022*. ACM, 86–92.

[28] Changman Lee, Dongho Sim, Joo Young Hwang, and Sangyeun Cho. 2015. F2FS: A New File System for Flash Storage. In *Proceedings of the 13th USENIX Conference on File and Storage Technologies, FAST 2015*. USENIX Association, 273–286.

[29] Shaohong Li, Xi Wang, Xiao Zhang, Vasileios Kontorinis, Sreekumar Kodakara, David Lo, and Parthasarathy Ranganathan. 2020. Thunderbolt: Throughput-Optimized, Quality-of-Service-Aware Power Capping at Scale. In *14th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2020*. USENIX Association, 1241–1255.

[30] Heiner Litz, Javier Gonzalez, Ana Klimovic, and Christos Kozyrakis. 2022. RAIL: Predictable, Low Tail Latency for NVMe Flash. *ACM Trans. Storage* 18, 1 (2022), 5:1–5:21.

[31] Mingzhe Liu, Haikun Liu, Chencheng Ye, Xiaofei Liao, Hai Jin, Yu Zhang, Ran Zheng, and Liting Hu. 2022. Towards Low-Latency I/O Services for Mixed Workloads Using Ultra-Low Latency SSDs. In *ICS '22: 2022 International Conference on Supercomputing, 2022*. ACM, 13:1–13:12.

[32] Hui Lu, Brendan Saltaformaggio, Ramana Rao Kompella, and Dongyan Xu. 2015. vFair: Latency-Aware Fair Storage Scheduling via per-IO Cost-Based Differentiation. In *Proceedings of the Sixth ACM Symposium on Cloud Computing, SoCC 2015*. ACM, 125–138.

[33] Till Miemietz, Hannes Weisbach, Michael Roitzsch, and Hermann Härtig. 2019. K2: Work-Constraining Scheduling of NVMe-Attached Storage. In *IEEE Real-Time Systems Symposium, RTSS 2019*. IEEE, 56–68.

[34] Amy Ousterhout, Joshua Fried, Jonathan Behrens, Adam Belay, and Hari Balakrishnan. 2019. Shenango: Achieving High CPU Efficiency for Latency-Sensitive Datacenter Workloads. In *16th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2019*. USENIX Association, 361–378.

[35] Stan Park and Kai Shen. 2012. FIOS: A Fair, Efficient Flash I/O Scheduler. In *Proceedings of the 10th USENIX conference on File and Storage Technologies, FAST 2012*. USENIX Association, 13.

[36] Bo Peng, Haozhong Zhang, Jianguo Yao, Yaozu Dong, Yu Xu, and Haibing Guan. 2018. MDev-NVMe: A NVMe Storage Virtualization Solution with Mediated Pass-Through. In *2018 USENIX Annual Technical Conference, USENIX ATC 2018*. USENIX Association, 665–676.

[37] Zebin Ren, Krijn Doekemeijer, Nick Tehrany, and Animesh Trivedi. 2024. BFQ, Multiqueue-Deadline, or Kyber? Performance Characterization of Linux Storage Schedulers in the NVMe Era. *To Appear in the Proceedings of the 2024 ACM/SPEC International Conference on Performance Engineering, ICPE 2024*. (2024).

[38] Zebin Ren and Animesh Trivedi. 2023. Performance Characterization of Modern Storage Stacks: POSIX I/O, libaio, SPDK, and io_uring. In *Proceedings of the 3rd Workshop on Challenges and Opportunities of Efficient and Performant Storage Systems, CHEOPS 2023*. ACM, 35–45.

[39] Kai Shen and Stan Park. 2013. FlashFQ: A Fair Queueing I/O Scheduler for Flash-Based SSDs. In *2013 USENIX Annual Technical Conference, 2013*. USENIX Association, 67–78.

[40] Matthew Wachs, Michael Abd-El-Malek, Eno Thereska, and Gregory R. Ganger. 2007. Argon: Performance Insulation for Shared Storage Servers. In *5th USENIX Conference on File and Storage Technologies, FAST 2007*. USENIX, 61–76.

[41] Mingyang Wang and Yiming Hu. 2014. An I/O Scheduler Based on Fine-Grained Access Patterns to Improve SSD Performance and Lifespan. In *Symposium on Applied Computing, SAC 2014*. ACM, 1511–1516.

[42] Caeden Whitaker, Sidharth Sundar, Bryan Harris, and Nihat Altiparmak. 2023. Do We Still Need IO Schedulers for Low-latency Disks?. In *Proceedings of the 15th ACM/USENIX Workshop on Hot Topics in Storage and File Systems, HotStorage 2023*. ACM, 44–50.

[43] Jiwon Woo, Minwoo Ahn, Gyusun Lee, and Jinkyu Jeong. 2021. D2FQ: Device-Direct Fair Queueing for NVMe SSDs. In *19th USENIX Conference on File and Storage Technologies, FAST 2021*. USENIX Association, 403–415.

[44] Kan Wu, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. 2019. Towards an Unwritten Contract of Intel Optane SSD. In *11th USENIX Workshop on Hot Topics in Storage and File Systems, HotStorage 2019*. USENIX Association.

[45] Suzhen Wu, Weiwei Zhang, Bo Mao, and Hong Jiang. 2019. HotR: Alleviating Read/Write Interference with Hot Read Data Replication for Flash Storage. In *Design, Automation & Test in Europe Conference & Exhibition, DATE 2019*. IEEE, 1367–1372.

[46] Qiumin Xu, Huzefa Siyamwala, Mrinmoy Ghosh, Tameesh Suri, Manu Awasthi, Zvika Guz, Anahita Shayesteh, and Vijay Balakrishnan. 2015. Performance Analysis of NVMe SSDs and Their Implication on Real World Databases. In *Proceedings of the 8th ACM International Systems and Storage Conference, SYSTOR 2015*. ACM, 6:1–6:11.

[47] Jinfeng Yang, Bingzhe Li, and David J. Lilja. 2020. Exploring Performance Characteristics of the Optane 3D Xpoint Storage Technology. *ACM Trans. Model. Perform. Evaluation Comput. Syst.* 5, 1 (2020), 4:1–4:28.

[48] Minhoon Yi, Minho Lee, and Young Ik Eom. 2017. CFFQ: I/O Scheduler for Providing Fairness and High Performance in SSD Devices. In *Proceedings of the 11th International Conference on Ubiquitous Information Management and Communication, IMCOM 2017*. ACM, 87.