on Accelerating Data Management Systems Using Modern Processor and Storage Architectures - (ADMS).

[25] Feng Chen, Binbing Hou, and Rubao Lee. 2016. Internal Parallelism of Flash Memory-Based Solid-State Drives. *Transactions on Storage (TOS)* 12, 3, Article 13 (May 2016), 39 pages. https://doi.org/10.1145/2818376

[26] Shimin Chen, Phillip B. Gibbons, Todd C. Mowry, and Gary Valentin. 2002. Fractal Prefetching B$^+$-Trees: Optimizing Both Cache and Disk Performance. In *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*. 157–168.

[27] Douglas Comer. 1979. The Ubiquitous B-Tree. 11, 2 (June 1979), 121–137.

[28] Alexander Conway, Ainesh Bakshi, Yizheng Jiao, William Jannen, Yang Zhan, Jun Yuan, Michael A. Bender, Rob Johnson, Bradley C. Kuszmaul, Donald E. Porter, and Martin Farach-Colton. 2017. File Systems Fated for Senescence? Nonsense, Says Science!. In *15th USENIX Conference on File and Storage Technologies (FAST)*. 45–58.

[29] Alex Conway, Ainesh Bakshi, Yizheng Jiao, Yang Zhan, Michael A. Bender, William Jannen, Rob Johnson, Bradley C. Kuszmaul, Donald E. Porter, Jun Yuan, and Martin Farach-Colton. 2017. How to Fragment Your File System. *;login:* 42, 2 (2017). https://www.usenix.org/publications/login/summer2017/conway

[30] Alexander Conway, Martin Farach-Colton, and Philip Shilane. 2018. Optimal Hashing in External Memory. In *Proceedings of the 45th International Colloquium on Automata, Languages and Programming (ICALP)*. 39:1–39:14. https://doi.org/10.4230/LIPIcs.ICALP.2018.39

[31] Alex Conway, Eric Knorr, Yizheng Jiao, Michael A. Bender, William Jannen, Rob Johnson, Donald E. Porter, and Martin Farach-Colton. 2019. Filesystem Aging: ItâĂŹs more Usage than Fullness. In *11th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage)*.

[32] Peter Desnoyers. 2013. What Systems Researchers Need to Know about NAND Flash. In *Proceedings of the 5th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage)*.

[33] John Esmet, Michael A. Bender, Martin Farach-Colton, and Bradley C. Kuszmaul. 2012. The TokuFS Streaming File System. In *Proceedings of the 4th USENIX Conference on Hot Topics in Storage and File Systems (HotStorage)*. 14.

[34] Matteo Frigo, Charles E. Leiserson, Harald Prokop, and Sridhar Ramachandran. 2012. Cache-Oblivious Algorithms. *ACM Transactions on Algorithms (TALG)* 8, 1 (2012), 4.

[35] Pedram Ghodsnia, Ivan T. Bowman, and Anisoara Nica. 2014. Parallel I/O Aware Query Optimization. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*. 349–360. https://doi.org/10.1145/2588555.2595635

[36] Google, Inc. [n. d.]. LevelDB: A fast and lightweight key/value database library by Google. https://github.com/google/leveldb, Last Accessed Sep. 26, 2018.

[37] Jun He, Sudarsun Kannan, Andrea C. Arpaci-Dusseau, and Remzi H. Arpaci-Dusseau. 2017. The Unwritten Contract of Solid State Drives. In *Proceedings of the Twelfth European Conference on Computer Systems (EuroSys)*. 127–144.

[38] IBM. 2017. Buffered inserts in partitioned database environments. https://www.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.apdv.embed.doc/doc/c0061906.html.

[39] IBM Informix. [n. d.]. Understanding SQL insert cursors. https://www.ibm.com/support/knowledgecenter/en/SSBJG3_2.5.0/com.ibm.gen_busug.doc/c_fgl_InsertCursors_002.htm

[40] Riko Jacob and Nodari Sitchinava. 2017. Lower Bounds in the Asymmetric External Memory Model. In *Proceedings of the 29th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*. 247–254. https://doi.org/10.1145/3087556.3087583

[41] William Jannen, Jun Yuan, Yang Zhan, Amogh Akshintala, John Esmet, Yizheng Jiao, Ankur Mittal, Prashant Pandey, Phaneendra Reddy, Leif Walsh, Michael A. Bender, Martin Farach-Colton, Rob Johnson, Bradley C. Kuszmaul, and Donald E. Porter. 2015. BetrFS: A Right-Optimized Write-Optimized File System. In *Proceedings of the 13th USENIX Conference on File and Storage Technologies (FAST)*. 301–315.

[42] Chris Jermaine, Anindya Datta, and Edward Omiecinski. 1999. A Novel Index Supporting High Volume Data Warehouse Insertion. In *Proceedings of 25th International Conference on Very Large Data Bases (VLDB)*. 235–246. http://www.vldb.org/conf/1999/P23.pdf

[43] Bradley C. Kuszmaul. 2009. How Fractal Trees Work. In *OpenSQL Camp*. Portland, OR, USA. An expanded version was presented at the MySQL User Conference, Santa Clara, CA, USA April 2010.

[44] Amanda McPherson. [n. d.]. A Conversation with Chris Mason on Btrfs: the next generation file system for Linux. https://www.linuxfoundation.org/blog/2009/06/a-conversation-with-chris-mason-on-btrfs/, Last Accessed Sep. 26, 2018.

[45] MySQL 5.7 Reference Manual. [n. d.]. Chapter 15 The InnoDB Storage Engine. http://dev.mysql.com/doc/refman/5.7/en/innodb-storage-engine.html.

[46] NuDB. 2016. NuDB: A fast key/value insert-only database for SSD drives in C++11. https://github.com/vinniefalco/NuDB.

[47] Patrick O'Neil, Edward Cheng, Dieter Gawlic, and Elizabeth O'Neil. 1996. The Log-Structured Merge-Tree (LSM-tree). *Acta Informatica* 33, 4 (1996), 351–385. https://doi.org/10.1007/s002360050048

[48] Oracle. 2017. Tuning the Database Buffer Cache. https://docs.oracle.com/database/121/TGDBA/tune_buffer_cache.htm.

[49] Oracle Corporation. [n. d.]. MySQL 5.5 Reference Manual. https://dev.mysql.com/doc/refman/5.5/en/innodb-file-space.html, Last Accessed Sep. 26, 2018.

[50] Oracle Corporation. 2015. Oracle BerkeleyDB Reference Guide. http://sepp.oetiker.ch/subversion-1.5.4-rp/ref/am_conf/pagesize.html, Last Accessed August 12, 2015.

[51] Oracle Corporation. 2016. Setting Up Your Data Warehouse System. https://docs.oracle.com/cd/B28359_01/server.111/b28314/tdpdw_system.htm.

[52] Anastasios Papagiannis, Giorgos Saloustros, Pilar González-Férez, and Angelos Bilas. 2016. Tucana: Design and Implementation of a Fast and Efficient Scale-up Key-value Store. In *Proceedings of the USENIX 2016 Annual Technical Conference (USENIX ATC)*. 537–550.

[53] John Paul. [n. d.]. Teradata Thoughts. http://teradata-thoughts.blogspot.com/2013/10/teradata-13-vs-teradata-14_20.html, Last Accessed Sep. 26, 2018.

[54] Harald Prokop. 1999. *Cache-Oblivious Algorithms*. Master's thesis. Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.

[55] Pandian Raju, Rohan Kadekodi, Vijay Chidambaram, and Ittai Abraham. 2017. PebblesDB: Building Key-Value Stores using Fragmented Log-Structured Merge Trees. In *Proceedings of the 26th Symposium on Operating Systems Principles (SOSP)*. 497–514. https://doi.org/10.1145/3132747.3132765

[56] Mendel Rosenblum and John K. Ousterhout. 1992. The Design and Implementation of a Log-structured File System. *ACM Trans. Comput. Syst.* 10, 1 (Feb. 1992), 26–52. https://doi.org/10.1145/146941.146943

[57] C Ruemmler and J. Wilkes. 1994. An introduction to disk drive modeling. *IEEE Computer* 27, 3 (1994), 17–29.

[58] SAP. 2017. RLV Data Store for Write-Optimized Storage. http://help-legacy.sap.com/saphelp_iq1611_iqnfs/helpdata/en/a3/13783784f21015bf03c9b06ad16fc0/content.htm.

[59] Keith A. Smith and Margo I. Seltzer. 1997. File System Aging — Increasing the Relevance of File System Benchmarks. In *Measurement and Modeling of Computer Systems*. 203–213.

[60] TokuDB. [n. d.]. https://github.com/percona/PerconaFT, Last Accessed Sep. 24 2018..

[61] Tokutek, Inc. [n. d.]. TokuMX—MongoDB Performance Engine. https://www.percona.com/software/mongo-database/percona-tokumx, Last Accessed Sep. 26, 2018.

[62] Tokutek, Inc. 2013. TokuDB: MySQL Performance, MariaDB Performance. http://www.tokutek.com/products/tokudb-for-mysql/.

[63] Vertica. 2017. WOS (Write Optimized Store). https://my.vertica.com/docs/7.1.x/HTML/Content/Authoring/Glossary/WOSWriteOptimizedStore.htm.

[64] Jeffrey Scott Vitter. 2001. External memory algorithms and data structures: Dealing with massive data. *ACM Computing surveys (CsUR)* 33, 2 (2001), 209–271.

[65] James Christopher Wyllie. 1979. *The Complexity of Parallel Computations*. Ph.D. Dissertation. Ithaca, NY, USA. AAI8004008.

[66] Jimmy Xiang. 2012. Apache HBase Write Path. http://blog.cloudera.com/blog/2012/06/hbase-write-path/.

[67] Jun Yuan, Yang Zhan, William Jannen, Prashant Pandey, Amogh Akshintala, Kanchan Chandnani, Pooja Deo, Zardosht Kasheff, Leif Walsh, Michael A. Bender, Martin Farach-Colton, Rob Johnson, Bradley C. Kuszmaul, and Donald E. Porter. 2016. Optimizing Every Operation in a Write-optimized File System. In *Proceedings of the 14th USENIX Conference on File and Storage Technologies (FAST)*. 1–14.

[68] Jun Yuan, Yang Zhan, William Jannen, Prashant Pandey, Amogh Akshintala, Kanchan Chandnani, Pooja Deo, Zardosht Kasheff, Leif Walsh, Michael A. Bender, Martin Farach-Colton, Rob Johnson, Bradley C. Kuszmaul, and Donald E. Porter. 2017. Writes Wrought Right, and Other Adventures in File System Optimization. *TOS* 13, 1 (2017), 3:1–3:26.

[69] Yang Zhan, Alexander Conway, Yizheng Jiao, Eric Knorr, Michael A. Bender, Martin Farach-Colton, William Jannen, Rob Johnson, Donald E. Porter, and Jun Yuan. 2018. The Full Path to Full-Path Indexing. In *Proceedings of the 16th USENIX Conference on File and Storage Technologies (FAST)*. 123–138.