The Next 700 Transaction Processing Engines

Anastasia Ailamaki
EPFL and RAW Labs SA
Lausanne, Switzerland
anastasia.ailamaki@epfl.ch

ABSTRACT

Database systems have always been designed and optimized to maximize a specific target metric. For over four decades, throughput has been the target metric of choice for Online Transaction Processing engines. Traditionally, OLTP engines were deployed on uniprocessors with high-performance disk arrays. Given the guaranteed improvement in single-threaded performance provided by Moore's law, disk-based OLTP engines focused on optimizing for the then-dominating source of overhead – disk I/O. Research into multitasking and buffer caching techniques steadily provided improvements in throughput and research into concurrency control and recovery techniques guaranteed that such improvements could be achieved without compromising data consistency.

Around mid 2000s, Dennard scaling came to a crushing halt and processor vendors were no longer able to raise clock frequencies without substantially increasing power consumption. This led to the birth of multicore processors, which provide explicit thread-level parallelism as an alternative to frequency scaling for increasing throughput. Multicore processors, however, also challenge OLTP engine design, as the transition from multitasking-style parallelism to true concurrency requires both thread synchronization to protect data structure consistency and transaction synchronization to protect database consistency. Thus, OLTP research focused on developing scalable synchronization techniques for exploiting parallelism provided by multicore processors [1,2,3].

Towards the late 2000s, DRAM price free-fall reached a level where it was possible for a single server to have terabytes of memory. With the exception of a few rare cases, it is now possible to fit most operational databases entirely in memory. Research from the database community quickly demonstrated the lack of scalability of traditional disk-based OLTP engines in the new main-memory contexts [4]. Subsequent research efforts on scalable main-memory OLTP engines adopt radically different designs when compared to their disk-based counterparts [5,6,7].

Today, state-of-the-art main-memory OLTP engines can handle millions of transactions per second and provide near-linear scalability under most workloads. However, three recent trends indicate an impending change in OLTP engine design once again: changes in application workloads, shifting hardware landscape, and new target metrics. We describe these trends briefly, and then outline the contents of the talk.

Changes in application workloads. As main-memory OLTP engines are increasingly adopted in new application domains ranging from online gaming to metadata back-ends for high-performance file systems, it has become important to support high-contention workloads with skewed data accesses. Such workloads pose a challenge for even state-of-the-art main-memory OLTP engines, because synchronization inherent to concurrency control protocols emerges as a scalability bottleneck [8,9]. What's more, emerging hybrid transactional and analytical processing (HTAP) workloads demand ACID semantics, high throughput, and performance isolation (for OLTP), as well as interactive response times and data freshness (for OLAP) [16].

Shifting hardware landscape. Today's high-end servers are multi-socket multi-cores, packing hundreds of cores in a single chassis. As the number of cores will soon enter the thousands, there is increasingly high variability in core-to-core communication latencies: cores within a socket communicate faster than cores across sockets. As cores employ increasingly deeper caching hierarchies, communication latencies within a socket are also different, with cores that share a cache being able to communicate faster than cores that do not share a cache. Ignoring variations in communication latency will inevitably result in lost performance optimization opportunities [10], but will also lead to poor scalability under high-contention workloads that already severely stress communication on current multicore processors[11]. At the same time, GPGPUs evolve from memory-limited, niche accelerators into general-purpose processors that support advanced features, which can be used to meet the aforementioned emerging applications' requirements [16].

New target metrics. The past few years have also witnessed a rise in the adoption of cloud-hosted database engines. The migration of OLTP engines to cloud-native setting has resulted in energy efficiency as being recognized as an important metric to be optimized in addition to throughput [12,13]. Improving energy efficiency requires a concerted effort from both hardware and software, as the hardware must be energy proportional and the software must avoid resource underutilization. Unfortunately, recent research has shown that the current state is far from ideal as servers used for deploying OLTP engines are not energy proportional and even state-of-the-art main-memory OLTP engines substantially underutilize hardware [14,15].

In this talk, we discuss the implications of these trends on the design of next-generation transaction processing engines. We revisit old designs, examine current designs, and explore new designs with the twin goal of meeting changing application demands and optimizing for newer metrics by exploiting emerging hardware. We also discuss our ongoing projects to address the issues stemming from the changing hardware and software landscape and adapt engine designs to emerging trends, in order to demonstrate that transaction processing is a dynamic research area with a rich history, a vibrant present, and a revolutionary future.

SIGMOD'17, May 14–19, 2017, Chicago, IL, USA.
ACM 978-1-4503-4197-4/17/05.
http://dx.doi.org/10.1145/3035918.3056095

ACKNOWLEDGEMENTS

This talk includes work by several past and current members of the Data-Intensive Applications and Systems Laboratory at the School of Computer and Communication Sciences at EPFL, as well as our academic and industrial collaborators over the past few years at the University of Toronto, SAP, Oracle, IBM, and Huawei. Special thanks for helping to shape the talk abstract are due to Utku Sirin, and to Dr. Raja Appuswamy. Last but not least, we thank our sponsors who are currently supporting our work on transactional engines: the Swiss National Science Foundation (grant 200021_146407), the European Union (ERC grant 617508), SAP, and Huawei.

REFERENCES

[1] Johnson, R., Pandis, I., Hardavellas, N., Ailamaki, A. and Falsafi, B. **Shore-MT: A Scalable Storage Manager for the Multicore Era**. *Proceedings of the International Conference on Extending Database Technology (EDBT),* 2009.

[2] Pandis, I., Johnson, R., Hardavellas, N., and Ailamaki, A. **Data-oriented transaction execution**. *Proceedings of the VLDB Endowment 3(1),* 2010.

[3] Johnson, R., Pandis, I., Stoica, R., Athanassoulis, M. and Ailamaki, A. **Aether: A scalable approach to logging**. *Proceedings of the VLDB Endowment 3(1),* 2010.

[4] Harizopoulos, S., Abadi, D., Madden, S., and Stonebraker, M. **OLTP through the looking glass, and what we found there**. *Proceedings of the ACM SIGMOD International Conference on Management of Data,* 2008.

[5] Tu, S., Zheng, W., Kohler, E., Liskov, B., and Madden, S. **Speedy transactions in multicore in-memory databases**. *Proceedings of the 24th ACM Symposium on Operating Systems Principles (SOSP),* 2013.

[6] Diaconu, C., Freedman, C., Ismert, E., Larson, P., Mittal, P., Stonecipher, R., Verma, N., and Zwilling, M. **Hekaton: SQL server's memory-optimized OLTP engine.** *Proceedings of the ACM SIGMOD International Conference on Management of Data,* 2013.

[7] Larson, P., and Levandoski, J. **Modern main-memory database systems.** *Proceedings of the VLDB Endowment* 9(13), 2016.

[8] Yu, X., Bezerra, G., Pavlo, A., Devadas, S., and Stonebraker. M. **Staring into the abyss: an evaluation of concurrency control with one thousand cores.** *Proceedings of the VLDB Endowment* 8(3), 2014.

[9] Ren, K., Faleiro, J.M, and Abadi. D.J. **Design Principles for Scaling Multi-core OLTP Under High Contention.** *Proceedings of the ACM SIGMOD International Conference on Management of Data,* 2016.

[10] Porobic, D., Pandis, I., De Oliveira Branco, M.S., Tozun, P. and Ailamaki, A. **Characterization of the Impact of Hardware Islands on OLTP**. *The VLDB Journal, 25(5),* 2016.

[11] Wang, T. and Kimura, H. **Mostly-optimistic concurrency control for highly contended dynamic workloads on a thousand cores.** *Proceedings of the VLDB Endowment 10(2),* 2016.

[12] Barroso, L. A. and Hölzle, U. **The Case for Energy-Proportional Computing.** *IEEE Computer, 40(12),* 2007.

[13] D. Tsirogiannis, S. Harizopoulos, and M. A. Shah. **Analyzing the Energy Efficiency of a Database Server.** *Proc. ACM SIGMOD Intl. Conference on Management of Data,* 2010.

[14] Sirin, U., Appuswamy, R., and Ailamaki, A. **OLTP on a server-grade ARM: power, throughput and latency comparison.** *Proc. Intl. Workshop on Data Management on New Hardware (DaMoN),* 2016.

[15] Sirin, U., Tozun, P., Porobic, D. and Ailamaki, A. (2016) **Micro-architectural Analysis of In-memory OLTP**. *Proc. ACM SIGMOD Intl. Conference on Management of Data, 2016.*

[16] Appuswamy, R., Karpathiotakis, M., Porobic, D., and Ailamaki, A. **The Case for Heterogeneous HTAP.** *Proceedings of the Conference on Innovative Database Systems (CIDR),* 2017.