Transactions in the Era of Non Volatile Memory and Heterogeneous Memory Architectures

Paolo Romano
romano@inesc-id.pt
Instituto Superior Técnico, Universidade de Lisboa & INESC-ID, Portugal

ABSTRACT

Transactions are a simple, yet powerful, abstraction that aims at masking programmers from the complexity of having to ensure correct and efficient synchronization of concurrent code.

Originally introduced in the domain of database systems, transactions have recently garnered significant interest in the broader domain of concurrent programming, via the Transactional Memory (TM) paradigm [6, 7, 9]. Nowadays, hardware supports for TM are provided in commodity CPUs (e.g., by Intel and IBM) and, at the software level, TM has been integrated in mainstream programming languages, such as C/C++ and Java.

In this talk I will present the novel challenges and research opportunities that arise in the area of TM due to the emergence of two recent hardware trends, namely Non-Volatile Memory (NVM) and heterogeneous computing architectures.

On the front of NVM, I will focus on the problem of how to allow the execution of transactions over NVM using unmodified commodity hardware TM (HTM) implementations. However, the reliance of commodity HTM implementations on CPU caches raises a crucial problem when applications access data stored in NVM from within a HTM transaction. Since CPU caches are volatile in today's systems, HTM implementations do not guarantee that the effects of a hardware transaction are atomically transposed to PM when the transaction commits — although such effects are immediately visible to subsequent transactions.

In this talk, I will overview somoe recent approaches to tackle this problem [2, 4, 5, 8] and present experimental results highlighting the existence of several bottlenecks that hinder the scalability of existing solutions. Next, I will show how these limitations can be addressed by presenting SPHT [1]. SPHT introduces a novel commit logic that considerably mitigates the scalability bottlenecks of previous alternatives, providing up to 2.6×/2.2× speedups at 64 threads in, resp., STAMP/TPC-C. Moreover, SPHT introduces a novel approach to log replay that employs cross-transaction log linking and a NUMA-aware parallel background replayer. In large persistent heaps, the proposed approach achieves gains of 2.8×.

On the front of heterogeneous computing, I will present the abstraction of Heterogeneous Transactional Memory (HeTM) [3]. HeTM provides programmers with the illusion of a single memory region, shared among the CPUs and the (discrete) GPU(s) of a heterogeneous system, with support for atomic transactions.

Besides introducing the abstract semantics and programming model of HeTM, I will present the design and evaluation of a concrete implementation of the proposed abstraction, which we named Speculative HeTM (SHeTM). SHeTM makes use of a novel design that leverages speculative techniques that aim at hiding the large communication latency between CPUs and discrete GPUs and at minimizing inter-device synchronization overhead.

CCS CONCEPTS

• **Hardware → Memory and dense storage**; • **Software and its engineering**;

KEYWORDS

Concurrent Programming, Transactional Memory, Non-Volatile Memory, Heterogeneous Architectures

ACM Reference Format:
Paolo Romano. 2021. Transactions in the Era of Non Volatile Memory and Heterogeneous Memory Architectures. In *Companion of the 2021 ACM/SPEC International Conference on Performance Engineering (ICPE '21 Companion), April 19–23, 2021, Virtual Event, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3447545.3451904

SHORT BIOGRAPHY

Paolo Romano received his PhD from Rome University "Sapienza" (2007) and his Master degree "summa cum laude from Rome University "Tor Vergata" (2002). He is an Associate Professor at Técnico (U. Lisboa) and a Researcher at INESC-ID. His research interests include parallel and distributed computing, dependability, autonomic systems, performability modelling and evaluation, data management in large scale systems, cloud and high performance computing. In these areas, he published more than 150 papers, receiving 3 best awards, and has coordinated several national and European projects, including a COST Action bringing together researchers from 60 institutions and 17 countries. Paolo serves regularly as Program Committee member and reviewer for renowned international conferences and journals, including EuroSys, DSN, ICDCS, SIGMETRICS, IEEE TKDE, IEEE TPDS, ACM TOPLAS

ACKNOWLEDGMENTS

This work was partially supported by FCT, via projects UIDB /50021/2020 and PTDC/CCI-COM/31901/2017.

REFERENCES

[1] Daniel Castro, Alexandro Baldassin, João Barreto, and Paolo Romano. 2021. Scalable Persistent Hardware Transactions. In *19th USENIX Conference on File and Storage Technologies, FAST*. USENIX Association.

ICPE '21 Companion, April 19–23, 2021, Virtual Event, France

ACM ISBN 978-1-4503-8331-8/21/04.
https://doi.org/10.1145/3447545.3451904

[2] Daniel Castro, Paolo Romano, and João Barreto. 2019. Hardware Transactional Memory meets memory persistency. *J. Parallel and Distrib. Comput.* 130 (2019), 63–79.

[3] Daniel Castro, Paolo Romano, Aleksandar Ilic, and Amin M. Khan. 2019. HeTM: Transactional Memory for Heterogeneous Systems. In *28th International Conference on Parallel Architectures and Compilation Techniques, PACT 2019, Seattle, WA, USA, September 23-26, 2019*. IEEE, 232–244. https://doi.org/10.1109/PACT.2019.00026

[4] Kaan Genç, Michael D. Bond, and Guoqing Harry Xu. 2020. Crafty: Efficient, HTM-Compatible Persistent Transactions. In *41st ACM SIGPLAN Conference on Programming Language Design and Implementation - PLDI 2020*. ACM, London, UK, 16.

[5] Ellis Giles, Kshitij Doshi, and Peter Varman. 2018. Hardware Transactional Persistent Memory. In *Proceedings of the International Symposium on Memory Systems (MEMSYS'18)*. ACM, Alexandria, Virginia, USA, 190–205. https://doi.org/10.1145/3240302.3240305

[6] Rachid Guerraoui and Paolo Romano (Eds.). 2015. *Transactional Memory. Foundations, Algorithms, Tools, and Applications - COST Action Euro-TM IC1001*. Lecture Notes in Computer Science, Vol. 8913. Springer. https://doi.org/10.1007/978-3-319-14720-8

[7] Maurice Herlihy and J. Eliot B. Moss. 1993. Transactional Memory: Architectural Support for Lock-Free Data Structures. *SIGARCH Comput. Archit. News* 21, 2 (May 1993), 289–300. https://doi.org/10.1145/173682.165164

[8] Mengxing Liu, Mingxing Zhang, Kang Chen, and Xuehai Qian. 2017. DUDETM: Building Durable Transactions with Decoupling for Persistent Memory. In *Proceedings of the 22nd International Conference on Architectural Support for Programming Languages and Operating Systems - ASPLOS'17*. ACM Press, Xi'an, China, 329–343.

[9] Nir Shavit and Dan Touitou. 1995. Software Transactional Memory. In *Proceedings of the Fourteenth Annual ACM Symposium on Principles of Distributed Computing (PODC '95)*. ACM, New York, NY, USA, 204–213. https://doi.org/10.1145/224964.224987