Leveraging a user-land page table to efficiently use modern memory

— Advisor : Jana Toljaga and Gaël Thomas
— Location : Benagil team, Telecom SudParis/Inria, Palaiseau building

1 Context

The upcoming launch of CXL [26] will profoundly impact resource management in a data center. CXL defines a cache-coherent protocol at the scale of a whole machine. The cache-coherency domain includes the PCIe devices, the system memory and the CPUs. CXL opens the way to a fully disaggregated data center because we can connect the PCIe buses of a cluster of machines to a CXL fabric [11, 14]. In this setting, the loads and stores emitted by a processor are routed to the CXL fabric through the PCIe bus of the emitter, and then to the PCIe bus and the memory of the receiver. At the software level, accessing a local or a *far* memory located in another machine becomes thus totally transparent since an expression as simple as `a = 42` can be routed transparently to any memory of any machine connected to the CXL fabric.

Additionally to transparency, CXL also brings efficiency. First experiments report a far memory access latency that ranges from 300ns to 600ns [14, 26] : roughly 10 times faster than sending a page via RDMA [12, 23], and only 1.5 to 3 times slower than an access to a memory located in another domain on a Non-Uniform Memory Access (NUMA) architecture.

In this context, the old n-tier design that has dominated for thirty years become inadequate. The n-tier design allows the computation and the storage to scale independently, but induces a high cost because of data exchange between the tiers. This data exchange requires data transformations, which annihilates the advantage of an optimized cache-coherency protocol at the scale of a cluster of machines.

2 Subject

In the DiVa project of the PEPR Cloud, we have the goal of rethinking the architecture of the cloud applications at the era of CXL. For that, we propose to decouple the memory from the process. In detail, as in a multi-threaded setting, we propose to consider a global memory pool, and a set of processes that act on the memory pool. However, contrarily to a classical multi-threaded setting, the memory exists regardless of the processes : the memory acts as a permanent storage for ephemeral processes, which are launched on demand to serve clients or to perform large data analytics. Because any process can directly access the global memory, the architecture avoids the high cost of transforming data when it is exchanged between the processes.

Central to this architecture is a naming service, which, at a high-level, comes in replacement of a classical file system. This naming service has to allow a process to retrieve an object produced by another process from an identifier. The research project has the goal of studying how we can design such a naming service (a map) in shared memory able to scale to thousands of processes. For that, we will study how we can reuse a memory management unit to accelerate the access to the naming service.

3 Advisors expertise

Gaël Thomas (Benagil team) is an expert in systems. He has been working on NUMA architectures [4, 9, 10, 28], privacy [25, 29], performance analysis [5, 17], persistent memory [6,13], concurrent programming [15,16,21,22], bug analysis [19,20,24], and language runtime designs [1, 7, 8, 27]. Jana Toljaga (Banegil team) is a PhD candidate expert in virtualization and operating systems.

4 Expected skills

The candidate must have a good background in system programming, concurrent programming, distributed systems and C/C++.

Références

[1] Koutheir Attouchi, Gaël Thomas, Gilles Muller, Julia Lawall, and André Bottaro. Incinerator - eliminating stale references in dynamic OSGi applications.

In *Proceedings of the international conference on Dependable Systems and Networks, DSN'15*, page 11, Rio de Janeiro, Brazil, 2015. IEEE Computer Society.

[2] Mathieu Bacou, Grégoire Todeschi, Alain Tchana, and Daniel Hagimont. Nested virtualization without the nest. In *Proceedings of the International Conference on Parallel Processing, ICPP'19*, pages 1–10. ACM, 2019.

[3] Mathieu Bacou, Grégoire Todeschi, Alain Tchana, Daniel Hagimont, Baptiste Lepers, and Willy Zwaenepoel. Drowsy-dc : Data center power management system. In *Proceedings of the International Parallel and Distributed Processing Symposium, IPDPS'19*, pages 825–834. IEEE Computer Society, 2019.

[4] Bao Bui, Djob Mvondo, Boris Teabe, Kevin Jiokeng, Lavoisier Wapet, Alain Tchana, Gaël Thomas, Daniel Hagimont, Gilles Muller, and Noel De Palma. When eXtended para-virtualization (XPV) meets NUMA. In *Proceedings of the EuroSys European Conference on Computer Systems, EuroSys'19*, page 15, Dresden, Germany, 2019. ACM.

[5] Florian David, Gaël Thomas, Julia Lawall, and Gilles Muller. Continuously measuring critical section pressure with the Free-Lunch profiler. In *Proceedings of the conference on Object Oriented Programming Systems Languages and Applications, OOPSLA'14*, page 14, Portland, Oregon, US, 2014. ACM.

[6] Rémi Dulong, Rafael Pires, Andreia Correia, Valerio Schiavoni, Pedro Ramalhete, Pascal Felber, and Gaël Thomas. NVCache : A plug-and-play NVMM-based I/O booster for legacy systems. In *Proceedings of the international conference on Dependable Systems and Networks, DSN'21*, page 13, Taipei, Taiwan, 2021. IEEE Computer Society.

[7] Nicolas Geoffray, Gaël Thomas, Julia Lawall, Gilles Muller, and Bertil Folliot. VMKit : a substrate for managed runtime environments. In *Proceedings of the international conference on Virtual Execution Environments, VEE'10*, pages 51–62, Pittsburgh, PA, USA, 2010. ACM.

[8] Nicolas Geoffray, Gaël Thomas, Gilles Muller, Pierre Parrend, Stéphane Frénot, and Bertil Folliot. I-JVM : a java virtual machine for component isolation in OSGi. In *Proceedings of the international conference on Dependable Systems and Networks, DSN'09*, pages 544–553, Estoril, Portugal, 2009. IEEE Computer Society.

[9] Lokesh Gidra, Gaël Thomas, Julien Sopena, and Marc Shapiro. A study of the scalability of stop-the-world garbage collectors on multicores. In *Proceedings of the conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS'13*, pages 229–240, Houston, Texas, USA, 2013. ACM.

[10] Lokesh Gidra, Gaël Thomas, Julien Sopena, Marc Shapiro, and Nhan Nguyen. NumaGiC : a garbage collector for big data on big NUMA machines. In *Proceedings of the conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS'15*, page 14, Istanbul, Turkey, 2015. ACM.

[11] Donghyun Gouk, Sangwon Lee, Miryeong Kwon, and Myoungsoo Jung. Direct access, high-performance memory disaggregation with DirectCXL. In *Proceedings of the Usenix Annual Technical Conference, USENIX ATC'22*, pages 287–294, 2022.

[12] Juncheng Gu, Youngmoon Lee, Yiwen Zhang, Mosharaf Chowdhury, and Kang G. Shin. Efficient memory disaggregation with infiniswap. In *Proceedings of the conference on Networked Systems Design and Implementation, NSDI'17*. USENIX Association, 2017.

[13] Anatole Lefort, Yohan Pipereau, Kwabena Amponsem, Pierre Sutra, and Gaël Thomas. J-NVM : Off-heap persistent objects in java. In *Proceedings of the Symposium on Operating Systems Principles, SOSP'21*, page 16, online, 2021. ACM.

[14] Huaicheng Li, Daniel S. Berger, Lisa Hsu, Daniel Ernst, Pantea Zardoshti, Stanko Novakovic, Monish Shah, Samir Rajadnya, Scott Lee, Ishwar Agarwal, Mark D. Hill, Marcus Fontoura, and Ricardo Bianchini. Pond : CXL-based memory pooling systems for cloud platforms. In *Proceedings of the conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS'23*, page 574–587, 2023.

[15] Jean-Pierre Lozi, Florian David, Gaël Thomas, Julia Lawall, and Gilles Muller. Remote core locking : migrating critical-section execution to improve the performance of multithreaded applications. In *Proceedings of the Usenix Annual Technical Conference, USENIX ATC'12*, pages 65–76, Boston, MA, USA, 2012. USENIX Association.

[16] Jean-Pierre Lozi, Florian David, Gaël Thomas, Julia Lawall, and Gilles Muller. Fast and portable locking for multicore architectures. *ACM Transactions on Computer Systems (TOCS)*, 33(4) :13 :1–13 :62, January 2016.

[17] Mohamed Said Mosli, François Trahay, Alexis Lescouet, Gauthier Voron, Rémi Dulong, Amina Guermouche, Élisabeth Brunet, and Gaël Thomas. Using differential execution analysis to identify thread interference. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 30(12) :13, 2019.

[18] Djob Mvondo, Mathieu Bacou, Kevin Nguetchouang, Lucien Ngale, Stéphane Pouget, Josiane Kouam, Renaud Lachaize, Jinho Hwang, Tim Wood, Daniel

Hagimont, Noël De Palma, Bernabé Batchakui, and Alain Tchana. Ofc : An opportunistic caching system for faas platforms. In *Proceedings of the EuroSys European Conference on Computer Systems, EuroSys'21*, page 228–244. ACM, 2021.

[19] Nicolas Palix, Gaël Thomas, Suman Saha, Christophe Calvès, Julia Lawall, and Gilles Muller. Faults in linux : ten years later. In *Proceedings of the conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS'11*, pages 305–318, Newport Beach, CA, USA, 2011. ACM.

[20] Nicolas Palix, Gaël Thomas, Suman Saha, Christophe Calvès, Gilles Muller, and Julia Lawall. Faults in linux 2.6. *ACM Transactions on Computer Systems (TOCS)*, 32(2) :4 :1–4 :40, 2014.

[21] Thomas Preud'Homme, Julien Sopena, Gaël Thomas, and Bertil Folliot. BatchQueue : fast and memory-thrifty core to core communication. In *Proceedings of the international Symposium on Computer Architecture and High Performance Computing, SBAC-PAD'10*, pages 215–222, Petrópolis, Brazil, 2010. IEEE Computer Society.

[22] Thomas Preud'homme, Julien Sopena, Gaël Thomas, and Bertil Folliot. An improvement of OpenMP pipeline parallelism with the BatchQueue algorithm. In *Proceedings of the International Conference on Parallel and Distributed Systems, ICPADS'12*, page 8, Singapore, 2012. IEEE Computer Society.

[23] Zhenyuan Ruan, Malte Schwarzkopf, Marcos K. Aguilera, and Adam Belay. AIFM : High-Performance, Application-Integrated far memory. In *Proceedings of the conference on Operating Systems Design and Implementation, OSDI'20*. USENIX Association, 2020.

[24] Suman Saha, Jean-Pierre Lozi, Gaël Thomas, Julia Lawall, and Gilles Muller. Hector : Detecting resource-release omission faults in error-handling code for systems software. In *Proceedings of the international conference on Dependable Systems and Networks, DSN'13*, page 12, Budapest, Hungary, 2013. IEEE Computer Society.

[25] Vasily A. Sartakov, Stefan Brenner, Sonia Ben Mokhtar, Sara Bouchenak, Gaël Thomas, and Rüdiger Kapitza. Eactors : Fast and flexible trusted computing using sgx. In *Proceedings of the International Conference on Middleware, Middleware'18*, page 12, Rennes, France, 2018. ACM.

[26] Debendra Das Sharma. Compute Express Link (CXL) : Enabling heterogeneous data-centric computing with heterogeneous memory hierarchy. *IEEE Micro*, 2022.

[27] Gaël Thomas, Nicolas Geoffray, Charles Clément, and Bertil Folliot. Designing highly flexible virtual machines : the JnJVM experience. *Software - Practice & Experience (SP&E)*, 38(15) :1643–1675, 2008.

[28] Gauthier Voron, Gaël Thomas, Vivien Quéma, and Pierre Sens. An interface to implement NUMA policies in the xen hypervisor. In *Proceedings of the EuroSys European Conference on Computer Systems, EuroSys'17*, page 14, Belgrade, Serbia, 2017. ACM.

[29] Peterson Yuhala, Jämes Ménétrey, Pascal Felber, Valerio Schiavoni, Alain Tchana, Gaël Thomas, Hugo Guiroux, and Jean-Pierre Lozi. Montsalvat : Intel SGX shielding for GraalVM native images. In *Proceedings of the International Conference on Middleware, Middleware'21*, page 13, Québec, Canada, 2021. ACM.