

# Fabien Gaud

Multicore programming expert

Vancouver, BC, Canada

📞 778-985-2709

✉ me@fabien.gaud.net

🌐 www.fabiengaud.net

👤 github.com/fgaud

📄 fabien.gaud

I am a software engineer interested in improving the performance of applications, runtimes and operating systems. During my research days, I published several significant papers related to task scheduling and memory access optimizations. At Coho Data, I had the opportunity to explore some of the challenges of a scale-out storage architecture. I like environments offering both technical challenges and an opportunity to tackle new problems.

## Employments

- Since July 2016 **Senior Staff Engineer, Coho Data, Vancouver, Canada.**  
*Working on supporting data deduplication at the filesystem level*
- Designed cluster membership mechanism and distributed locking
  - Implemented a large portion of the storage support for data deduplication
  - Redesigned the snapshot mechanism to make it scalable and more flexible
- May 2014 **Senior Software Engineer, Coho Data, Vancouver, Canada.**  
July 2016 *Worked on performance and fault tolerance of a distributed NFS server*
- Worked on improving the scalability of the filesystem with millions of small files by using *containers* which drastically reduced metadata operations latency by 50% and rebalancing duration from hours to minutes
  - Worked on 2-ack support to improve consistency after disaster recovery
  - Wrote several tools to identify performance bottlenecks within the stack
- October 2011 **Post-doc, Simon Fraser University, Vancouver, Canada.**  
May 2014 *Worked on efficiently exploiting large pages for NUMA multicore architectures*
- Implemented a solution to transparently use large pages when they are beneficial while avoiding costly NUMA effects
  - Increased the performance of various applications (scientific and MapReduce applications, data servers) by up to 100% compared to Linux with small pages and up to 81% compared to Linux with large pages
- Worked on traffic management for NUMA multicore architectures*
- Implemented a solution to replicate memory pages in the Linux kernel
  - Redesigned thread and memory placement algorithms to consider traffic congestion on memory controllers and interconnects
  - Increased the performance of scientific and MapReduce applications by up to 150% compared to Linux
  - Code is available at <https://github.com/Carrefour>
- January 2011 **Research engineer, INRIA, Grenoble, France.**  
September 2011 **Worked on improving the performance of data servers on NUMA multicore architectures**
- Designed a runtime able to leverage the asymmetric performance of recent NUMA multicore processors
  - Evaluated the performance impact of task management strategies for multi-tier servers on NUMA multicore machine
- September 2007 **Ph.D candidate, Grenoble University, Grenoble, France.**  
December 2010 *Worked on improving the performance of data servers on multicore architectures*
- Studied the Apache Web server on multicore NUMA machines. Improved performance by up to 30% on the SPECWeb benchmark
  - Designed a multicore event-driven runtime up to 70% more efficient than the state-of-the-art solution on data servers.
  - Code is available at <https://github.com/fgaud/Mely>
- September 2007 **Teaching assistant, Université Joseph Fourier, Grenoble, France.**  
June 2010 *Instructor and teaching assistant*
- Participated in various courses around middleware and distributed systems

- March 2007 **MSc candidate**, INRIA, Grenoble, France.  
August 2007 *Worked on autonomic management of event-based execution flows*
  - Designed a new scheduling mechanism capable of switching from a thread-based execution to an event-based execution for the staged event-driven architecture (SEDA)
- June 2006 **Internship**, INRIA, Grenoble, France.  
September 2006 *Created tools to automatically deploy distributed file systems on GRID5000, a grid of computers*

---

## Education

- 2010 **Ph.D. in Computer Science**, Grenoble University, Grenoble, France.  
*Title* : Improving the performance of data servers on multicore architectures
- 2007 **Master of Science in Computer Science**, Université Joseph Fourier, Grenoble, France.  
*Title* : Autonomic management of event-based execution flows in component-based systems
- 2007 **Engineering degree in Computer Science**, Polytech'Grenoble, Grenoble, France.

---

## Computing skills

- Languages **C, Python**, Bash, C++, Perl, JAVA, Latex  
Systems Linux kernel and multicore programming, Scheduling, Memory management, Distributed systems, File systems  
Software **GNU tools, Git, Vim, Gdb, Valgrind**, Buildbot, Jira, Crucible  
OS **Linux**, Windows  
Methods Agile, Unit tests, Continuous integration

---

## Spoken languages

- French Native  
English Fluent

---

## Selected Publications

### International conferences

- J.P. Lozi, B. Lepers, J. Funston, F. Gaud, A. Fedorova and V. Quéma.  
***The Linux Scheduler: A Decade of Wasted Cores.***  
European Conference on Computer Systems (EuroSys), 2016.
- F. Gaud, B. Lepers, J. Funston, J. Decouchant, J. Funston, A. Fedorova and V. Quéma.  
***Large Pages May be Harmful on NUMA Systems.***  
USENIX Annual Technical Conference (USENIX ATC), 2014.
- M. Dashti, A. Fedorova, J. Funston, F. Gaud, R. Lachaize, B. Lepers, V. Quéma, and M. Roth.  
***Traffic Management: A Holistic Approach to Memory Placement on NUMA Systems.***  
International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2013.
- T. Dwyer, A. Fedorova, S. Blagodurov, M. Roth, F. Gaud and J. Pei.  
***A Practical Method for Estimating Performance Degradation on Multicore Processors and its Application to HPC Workloads.***  
Supercomputing Conference (SC), 2012.
- F. Gaud, S. Genevès, R. Lachaize, B. Lepers, F. Mottet, G. Muller, and V. Quéma.  
***Efficient Workstealing for Multicore Event-Driven Systems.***  
International Conference on Distributed Computing Systems (ICDCS), 2010.

### Journals

- F. Gaud, B. Lepers, J. Funston, M. Dashti, A. Fedorova, V. Quéma, R. Lachaize, and M. Roth  
***Challenges of memory management on modern NUMA systems.***  
Communication of the ACM 58, 12, pp. 59-66, 2015.