Tom Cornebize

PhD student in computer science

Contact

tom.cornebize@
gmail.com
+33 6 74 69 59 41

Web

cornebize.net
github.com/Ezibenroc

Skills

Python	★★★★★
LaTeX	★★★★☆
GNU/Linux	★★★★☆
C	★★★★☆
C++	★★★★☆
Java	★★★★☆
MPI	★★★☆☆
R	★★★☆☆
SQL	★★☆☆☆

Languages

French	★★★★★
English	★★★★☆
German	★☆☆☆☆

Education

2017 - 2020, Grenoble (FR)
PhD in Computer Science — Grenoble Alps University
Under the supervision of Arnaud Legrand.
Topics of interest: high performance computing, distributed systems, performance evaluation.

2015 - 2017, Grenoble (FR)
Master's & Engineering Degrees in Computer Science — Ensimag
Graduate specialization in parallel and distributed systems.
Obtained a Master of Science, with highest honor.

2013 - 2015, Lyon (FR)
Bachelor's Degree in Theoretical Computer Science — ENS Lyon
Undergraduate and postgraduate intensive program in theoretical computer science.
Obtained a Bachelor of Science, with great honor.

2011 - 2013, Grenoble (FR)
Undergraduate program — Joseph Fourier University
Undergraduate program in computer science and mathematics.

Internships

Oct/17 - Dec/17, Chicago (US)
Performance variability in supercomputers — Argonne Laboratory
Under the supervision of Swann Perarnau.

Feb/17 - Jul/17, Grenoble (FR)
Efficient simulation of large scale MPI applications — Inria
Under the supervision of Arnaud Legrand.

- Profiled and generated traces of the simulator's execution.
- Modeled the expensive functions to inject their expected duration in the simulation.
- Replaced large allocations by fake allocations.
- Used huge pages to decrease the page table size.
- Outcome: simulate executions several orders of magnitude larger while keeping a small error.
- Obtained results for the large scale experiment of [1].
- Master's thesis: `https://hal.inria.fr/hal-01544827v1`

May/16 - Aug/16, Walldorf (DE)
Multicast communication in SAP HANA — SAP
Under the supervision of Norman May.

- Analyzed several multicast algorithms.
- Implemented these algorithms in C++, using HANA codebase.
- Implemented functionnal and performance tests in Python.

May/15 - Aug/15, Grenoble (FR)
Job isolation in fat tree topologies — Bull
Under the supervision of Matthieu Perotin.

- Designed several algorithms to prevent the leak of sensitive information in a cluster.
- Implemented a proof of concept, in Python, and worked on its integration in Bull's software stack.
- Results published in [2].

Jun/14 - Jul/14, Sophia-Antipolis (FR)
Modelisation and verification of concurrent systems — Inria
Under the supervision of Robert de Simone.

Jun/13 - Jul/13, Grenoble (FR)
Monitoring of distributed systems — Joseph Fourier University
Under the supervision of Yliès Falcone.

- Designed an algorithm for decentralized monitoring of distributed systems. Implementation in OCaml.
- Results published in [3].

Jun/12, Grenoble (FR)
Monitoring of distributed systems — Joseph Fourier University
Under the supervision of Yliès Falcone.

Software projects

May/16 - now **Contribution to Roaring bitmap** roaringbitmap.org

Fast and lightweight set for unsigned 32 bits integers.

- Implemented several functionnalities of the C library.
  - Implemented range constructor.
  - Implemented select query.
  - Implemented subset test.
  - Fixed several bugs.
  - Repository: github.com/roaringBitmap/CRoaring
- Developed a Python wrapper for the C library.
  - Functionalities of the C library directly usable in Python.
  - Implementation made using Cython.
  - Several order of magnitude faster than the builtin set.
  - Extensive tests caught several bugs of the C library.
  - Repository: github.com/Ezibenroc/PyRoaringBitMap
- Analyzed the performance of Roaring bitmap union.
  - Condudcted a full factorial experiment for the C library.
  - Modeled and analyzed the duration of the operation as a function of the size and densities of the sets, for both the Python and the C libraries.
  - Repository: github.com/Ezibenroc/roaring_analysis

Sep/14 - Dec/14 **Platypus** askplatyp.us

Modular and open source question answering framework.

- Developed a question parsing module in Python, with a grammatical approach (Stanford CoreNLP and NLTK libraries).
- Framework currently used and valorized by Lexistems SAS.

Jan/14 - May/14 **SAT Solver** github.com/Ezibenroc/satsolver

C++ program to solve the SAT problem.

- Implemented the DPLL algorithm, Watched literals and clause learning heuristics.

Publications

[1] Predicting the Energy Consumption of MPI Applications at Scale Using a Single Node
Heinrich, F. C.; Cornebize, T.; Degomme, A.; Legrand, A.; Carpen-Amarie, A.; Hunold, S.; Orgerie, A.-C., and Quinson, M.
URL: `https://hal.inria.fr/hal-01523608`
Cluster, 2017

[2] Isolating Jobs for Security on High-Performance Fabrics
Perotin, M. and Cornebize, T.
2017 IEEE 3rd International Workshop on High-Performance Interconnection Networks in the Exascale and Big-Data Era (HiPINEB), 2017

[3] Efficient and Generalized Decentralized Monitoring of Regular Languages
Falcone, Y.; Cornebize, T., and Fernandez, J.-C.
URL: `https://hal.archives-ouvertes.fr/hal-00972559`
34th Formal Techniques for Networked and Distributed Systems (FORTE), 2014