

# Tom Cornebize

Graduate student  
Theoretical computer science

+33(0)6 74 69 59 41  
✉ tom.cornebize@ens-lyon.fr  
🌐 cornebize.net

## Education

<b>École Normale Supérieure de Lyon</b> <i>Master of science in theoretical computer science</i>	<b>Lyon, France</b> <i>current</i>
<b>École Normale Supérieure de Lyon</b> <i>Bachelor of science in theoretical computer science, with great honor</i>	<b>Lyon, France</b> <i>2013–2014</i>
<b>Université Joseph Fourier</b> <i>Undergraduate program in computer science and mathematics</i>	<b>Grenoble, France</b> <i>2011–2013</i>

## Experience

### Research internships

<b>Bull</b> <i>High performance computing : job isolation and distributed optimization of routing.</i> Under the supervision of Matthieu Perotin.	<b>Grenoble, France</b> <i>May 2015–August 2015</i>
<b>Inria, AOSTE team</b> <i>Modelisation and verification of concurrent systems.</i> Under the supervision of Robert de Simone. <ul style="list-style-type: none"><li>- Studied classical models (Büchi automata, Petri nets, temporal logic, synchronous languages).</li></ul>	<b>Sophia-Antipolis, France</b> <i>June 2014–July 2014</i>
<b>Verimag laboratory</b> <i>Monitoring of distributed systems</i> Under the supervision of Yliès Falcone. <ul style="list-style-type: none"><li>- Designed an algorithm for decentralized monitoring of distributed systems.</li><li>- Implemented a benchmark to obtain experimental results.</li><li>- Wrote a report, "Efficient and Generalized Decentralized Monitoring of Regular Languages", published at FORTE 2014.</li></ul>	<b>Grenoble, France</b> <i>June 2013–July 2013</i>
<b>Laboratoire d'informatique de Grenoble (LIG)</b> <i>Monitoring of distributed systems</i> Under the supervision of Yliès Falcone. <ul style="list-style-type: none"><li>- Performed experimentations to assess the efficiency of an algorithm for decentralized monitoring.</li><li>- Proposed several optimizations.</li></ul>	<b>Grenoble, France</b> <i>June 2012</i>

### School projects

<b>Projet Pensées Profondes</b> <ul style="list-style-type: none"><li>- A modular and open source question answering framework. Team of seven students.</li><li>- Developed a question parsing module in Python, with a grammatical approach (Stanford CoreNLP and NLTK libraries).</li></ul>	
<b>SAT solver</b> <ul style="list-style-type: none"><li>- Developed a program implementing the DPLL algorithm to solve the SAT problem, in C++ language.</li><li>- Added <i>watched literals</i> and <i>clause learning</i> heuristics.</li><li>- Added a SMT (satisfiability modulo theories) solver.</li></ul>	
<b>Cellular automata</b> , in C. Used <i>MPI</i> .	
<b>Simulation of distributed search</b> , in Erlang.	
<b>P2P client</b> , in C. Used <i>pthread</i> and <i>socket</i> .	

## Languages

French: Mother tongue	German: Basic
English: Fluent	

## Computer skills

Programming languages: Python, C, C++, OCaml, Erlang, assembly languages (ARM and MIPS)  
Presentation languages:  $\text{\LaTeX}$ , Markdown  
Distributed systems / parallel programming: MPI, pthread, socket  
Miscellaneous: GNU/Linux, Git, unit testing