

# Fabian Ritter

Campus E1 3, Rm. 403, 66123 Saarbrücken, Germany  
[fabian.ritter@cs.uni-saarland.de](mailto:fabian.ritter@cs.uni-saarland.de)

## EDUCATION

---

- SINCE 2017 | **PhD student** in the INTERNATIONAL MAX PLANCK RESEARCH SCHOOL for Computer Science, at the COMPILER DESIGN LAB at SAARLAND UNIVERSITY
- 2016 – 2017 | **Doctoral Preparatory Phase** at SAARBRÜCKEN GRADUATE SCHOOL OF COMPUTER SCIENCE, SAARLAND UNIVERSITY
- 2012 – 2015 | **Bachelor of Science** in Computer Science with minor in Mathematics, SAARLAND UNIVERSITY
- 2004 – 2012 | **Abitur** at LEIBNIZ GYMNASIUM, St. Ingbert

## SCHOLARSHIPS AND CERTIFICATES

---

- SINCE 2017 | Fellow of the INTERNATIONAL MAX PLANCK RESEARCH SCHOOL for Computer Science
- 2016 – 2017 | Scholarship holder of SAARBRÜCKEN GRADUATE SCHOOL OF COMPUTER SCIENCE, SAARLAND UNIVERSITY
- 2013 – 2015 | Member of BACHELOR FÖRDERPROGRAMM, SAARLAND UNIVERSITY

## ACADEMIC ACTIVITIES

---

- 2014 – 2017 | Tutor/Teaching assistant at SAARLAND UNIVERSITY for the lectures:
- ▷ Compiler Construction
  - ▷ Nebenläufige Programmierung (concurrent programming)
  - ▷ Grundzüge der Theoretischen Informatik (theoretical computer science)
  - ▷ Systemarchitektur (system architecture)

## SKILLS

---

- LANGUAGE | ▷ German: native  
▷ English: fluent  
▷ French: basic knowledge
- TECHNICAL | ▷ C/C++: experienced, used in research projects, course work, private projects, especially using the LLVM compiler framework  
▷ SCALA, PYTHON: used in research projects and private projects  
▷ JAVA, C#, VERILOG: used in course work  
▷ L<sup>A</sup>T<sub>E</sub>X: used for writing academic documents, designing slides and posters

## RESEARCH PROJECTS

---

- 2017 | **Memory Safety in C** (current research)
- ▷ Understanding memory-safety-induced vulnerabilities as a problem of the programming language definition (rather than an eternal sequence of attacks and counter-measures)
  - ▷ Exploring ways of making C a “safe” programming language
- 2016 | **Supporting Transcendental Functions in Daisy, a Sound Verification Tool for the Precision of Floating-Point Computations**  
(Research Immersion Lab, AUTOMATED VERIFICATION AND APPROXIMATION group, MAX PLANCK INSTITUTE FOR SOFTWARE SYSTEMS)
- ▷ Extended Daisy for soundly estimating round-off errors caused by using floating-point operations for trigonometric and exponential functions
  - ▷ Developed algorithms for obtaining sound rational bounds for real-valued results of transcendental functions
- 2016 | **Memory Safety Analysis in Sprattus**  
(Research Immersion Lab, REAL-TIME AND EMBEDDED SYSTEMS LAB, SAARLAND UNIVERSITY)
- ▷ Implemented analysis domains for obtaining information about accessed memory ranges and allocated memory regions in our symbolic abstraction framework
  - ▷ Designed a structured memory model for LLVM bitcode for use in symbolic abstraction
  - ▷ Evaluated on benchmarks from the Software Verification Competition 2016 with promising results
- 2015 | **Compiler Optimizations using Symbolic Abstraction**  
(Bachelor’s Thesis, COMPILER DESIGN LAB, SAARLAND UNIVERSITY)
- ▷ Extended a framework for static analysis of LLVM bitcode by symbolic abstraction
  - ▷ Implemented classical compiler transformations in the clang compiler based on the found analysis results
  - ▷ Investigated how combining these analyses influences transformation quality