# Marco Perronet

Software Engineer


perronet.github.io

github.com/perronet

linkedin.com/in/marco-perronet

perronet.marco@gmail.com


## About me

I am a recent graduate student with experience in software engineering and research. I spent the past years exploring different fields of computer science while tinkering on personal software projects in my free time.

I moved to Germany after my Bachelor's to pursue a *Master+PhD* program at Max Planck Institute: in this program, students study for courses while working on research projects.

After obtaining my Master's degree and a scientific publication, I decided to move from academia to industry to pursue a career as a software developer.

## Languages

- **Italian** *(mother tongue)*
- **English** *(fluent)*
- **French** *(intermediate)*
- **German** *(intermediate)*

## Skills

I have programming experience in **Java**, **Python**, **Rust**, and **OCaml**. Being interested in operating systems and a long-time **Linux** user, I have extensive experience with **Bash** scripting and **C** programming.

During my three years of **research** experience I learned how to tackle complex problems. I can design, document, present my solution, then implement and evaluate it. This experience taught me how to deal with open-ended projects: from defining goals to measuring progress and pivoting when necessary.

## Education

2019-2022 **Master's degree in CS** Technische Universität Kaiserslautern, Germany

2016-2019 **Bachelor's degree in CS** Università degli studi di Torino, Italy

## Experience

2022 Jun-Aug **Software engineer intern** Meta, London

Infer is an open-source static analysis tool developed at Meta. During my internship, I extended Infer to support an analysis based on declarative logic programming with *Datalog*. I had to study the papers that describe the approach, learn *OCaml*, and collaborate with the team through meetings and code reviews. [Website] [Code]

2019-2022 **PhD student** Max Planck Institute For Software Systems, Germany

I worked in the field of real-time operating systems, and my project focused on trace-based response-time analysis on Linux. During this time, I designed and developed **DMXtrace**: a tool written in *Rust* that traces the processes running on the system, extracts a formal model, and uses it to analyze the timing correctness of the system, which is a crucial aspect of real-time systems. [Paper] [Code]

## Projects

2022 **Treecodes**

As an aesthetically pleasing alternative to QR codes, Treecodes encode information inside the topology of a tree, rather than a grid of bits. Treecodes are more flexible than QR codes when it comes to the drawing itself: whether the tree is a piece of art or is drawn like a stick figure, it will contain the same information. This enables anyone to choose their own style for their Treecode. The project is written in *Javascript*. [Demo on the website] [Code]

2019 **Linux kernel exploration**

I have been fiddling with the kernel codebase both for my research work and personal interest. I made a small contribution by showing the existence of a minor bug in the real-time scheduler, and collaborating with a kernel developer to create a patch to fix it. [Patch]

2018 **NP to SAT transform**

An implementation in *C* of the Cook-Levin theorem from complexity theory. The tool turns a (Turing machine, input problem) tuple into a formula for SAT solvers. [Code]

2016 **Interactive 3D map**

I 3D mapped the building of my high school and implemented it as a playable map in a videogame. In this project, I learned about level design and 3D modeling with *Blender*.