

Marco Maida

Tech Lead, Inference Performance

🌐 maida.me ✉ mmmaidacs@gmail.com

I'm a software engineer with over **ten years of experience** building systems that address complex, real-world problems. I worked in the fields of **industrial software**, **videogames**, **real-time systems**, **formal verification**, and **autonomous vehicles**. I'm currently based in **London**, where I lead the inference performance team at **Wayve**. We focus on improving the latency and reliability of Wayve's self-driving stack. I work mainly in **C++**, **Rust**, and **Python**, and I'm particularly interested in problems where **performance**, **correctness**, and **practical constraints** come together.

Professional Experience

- Since 2022 **Software Engineer – Autonomous Vehicles** Wayve Ltd
I designed, implemented, and maintained several key systems within Wayve's on-board software stack for autonomous vehicles. This includes sensor data collection, alignment, packaging, data upload, and real-time neural network inference. Since December 2024, I have been serving as the **Tech Lead** of the inference performance team, responsible for the core inference component. (*ROS2, Linux Kernel, C++, Rust, Python*).
- 2016-2019 **Software Engineer – Videogames** 34BigThings
I contributed to five major game titles, focusing on game infrastructure, AI, gameplay, and development tools. I worked on both single-player and online multiplayer games shipped on Steam, PS4, XboxOne, Switch, and mobile platforms. (*Unity3D, C#, Unreal Engine, C++*).
- 2015-2016 **Software Engineer – Videogames, Simulation** Maserati, Choralia
I led two freelance B2B projects: developing an educational game for mobile and browsers, and collaborating to create a 3D visualization tool for product presentation. (*Unity3D, C#, JavaScript*).
- 2013-2016 **Software Engineer – Industrial Software** R.O. srl
I contributed to a suite of software solutions for glass processing factories, focusing on order tracking and optimizing machine work, product waste, and logistics. During my third year, I managed a team of 4 junior engineers. (*C, C++, C#, SQL*).

Education and Research

- 2022 **Research Internship** Bloomberg LP
I worked on accelerating SAT solving using GPUs. (*C++, CUDA*).
- 2019-2022 **PhD Student** Max Planck Institute
My research focused on formal verification and real-time systems, specifically verifying the timeliness of software systems. I published three papers and mentored three interns. (*COQ, C, Rust*).
- 2019-2022 **Master's in Computer Science** Technische Universität Kaiserslautern
I specialized in real-time systems as part of a joint master's and PhD program.
- 2016-2019 **Bachelor's in Computer Science** Università degli Studi di Torino
I specialized in computability and formal methods.

Selected Publications

- 2025 **Claycode: Stylable and Deformable 2D Scannable Codes** SIGGRAPH 2025
★ *Journal publication in Transactions of Graphics. Selected for CAF trailer.*
We introduced Claycodes, a new type of visual code that breaks free from traditional rigid grids like QR codes. Claycodes encode data as a tree structure and allow full customizability and deformation, with real-time scanning capabilities. (<https://arxiv.org/abs/2505.08666>).
- 2022 **From Intuition to Coq: A Case Study in Verified Response-Time Analysis, FIFO Scheduling** RTSS 2022
We developed a formally verified response-time analysis for FIFO schedulers, challenging traditional pen-and-paper methods. (<https://people.mpi-sws.org/~kbedarka/rtss22.pdf>).
- 2021 **Foundational Response-Time Analysis as Explainable Evidence of Timeliness** ECRTS 2022
★ *Outstanding Paper Award.*
I developed POET, a tool for formally verified worst-case scenario timing analysis. (<https://drops.dagstuhl.de/opus/volltexte/2022/16336/pdf/LIPIcs-ECRTS-2022-19.pdf>).

Selected Open-Source Projects

2025	Claycode Inventor and lead contributor. Claycode includes a generator website, an Android scanner app, and a large amount of research code.	https://claycode.io
2020-2022	PROSA Main contributor to PROSA, one of the leading formally-verified frameworks in the real-time systems community.	https://gitlab.mpi-sws.org/RT-PROOFS/rt-proofs
2022	Treecode The precursor to Claycode, a novel 2D scannable code that encodes messages as unique trees.	www.maida.me/treecode
2021	POET First-ever implementation of a foundational response-time analysis tool, created as part of my first academic publication.	https://gitlab.mpi-sws.org/RT-PROOFS/POET
2018	Fast Mobile Cycle (FMC) Framework and Toolkit Developed a Unity3D framework that accelerates production-ready casual game creation, paired with a Python toolkit for bulk operations.	www.github.com/340openThings