

Work Experience

Port Zero

BERLIN, GERMANY

CTO

July 2018 – present

All of the work that I did as a senior software engineer, plus the pleasure and privilege of working with a diverse team of skilled engineers, taking responsibility for all of our work.

- Wrote a portfolio management system in Pharo Smalltalk for a Global 500 company that is able to visualize thousands of projects and tens of thousands of people working on them. We were able to help guide the budgeting through making KPIs and other facts about the company visible.
- Helps in hiring and onboarding fantastic engineers. We're a small company of mostly senior developers with a passion for what we do, and we always search for like-minded people.
- Takes responsibility for the work that we do, and helps guide the decision process for our tooling and infrastructure. We're a truly democratic company and no single person can claim to know everything, so I ask a lot of questions!
- Talks to clients on a regular basis about expectations, performance, and a mixture of nitty-gritty technical details and high-level budgeting topics.

Port Zero

BERLIN, GERMANY

Senior Software Engineer

June 2016 – June 2018

Built software for a diverse set of clients, including Internet Service Providers, startups, and Global 500 companies.

- Wrote an API for BIRD, an internet routing daemon, in Go, with microsecond latency and high configurability. It is currently being used by a variety of Internet Service Providers.
- Audited software systems written in Java and Python for faults and helped find security holes. This included communicating these issues to the teams in a blameless and constructive way.
- Built internal and Open Source tooling using Python and Django; C; and JavaScript for the company to help track clients and projects. This freed up time otherwise spent on synchronization between departments.
- Built Customer and Operations portals and help desks using Javascript and React; Python and Django; and Go that helped speed customer onboarding up from days to hours.
- Worked with Python and NumPy to speed core sections of a client's code base used for calculating resource availabilities up by a factor of 5-7.

Bright

BERLIN, GERMANY

Lead Software Developer

July 2015 – June 2016

Developed the backend of Bright's products, written in Python using Flask through several iterations. Brought it from a Minimum Viable Product to a set of maintainable services that helped artists monetize their art. Took care of code reviews across projects on Github and kept track of the code climate through Continuous Integration and Deployment. Was responsible for deployment on Heroku and AWS Lambda.

Freelancer

BERLIN, GERMANY

Software Development & Consultancy

February 2015 – present

Collaborated with a wide variety of clients in the Berlin area, mostly on web-based systems of various kinds. Developed, audited, and migrated systems while learning to remain calm under pressure.

- Maintained and extended customer-facing websites written in JavaScript and Angular. Split up applications into modules, drastically reducing onboarding time of other maintainers.
- Helped mitigate a DoS attack on a client's infrastructure while it happened. Moved integral and parts of the system into the AWS cloud for better load balancing and uptime while the system was running.

HTW INKA

BERLIN, GERMANY

Research Assistant (SIGNAL & FKI)

February 2014 – July 2015

Education

Recurse Center

NEW YORK CITY, NY

Recurser (Sabbatical)

May 2017 – August 2017

Researched and explored personal projects during a self-directed 3-month programming retreat. Work included reinventing language features using Lisp macros; building compilers and virtual machines in Rust and OCaml; and producing digital generative art using shaders and P5.js.

Non-Work Projects

e

e is a lightweight, minimal editor. It supports a wide variety of Operating Systems and is customizable through Lua. The executable is smaller than that of any other editor of comparable feature size.

zepto

zepto is a programming language with a focus on naïveté, malleability, and friendliness. It is both a testbench for new ideas in PLT and a pragmatic, usable implementation of a Scheme with a "batteries included" mindset.

Carp

Carp is a compiled Lisp with type inference and without garbage collection. It uses new ideas to get rid of the mundane parts of programming. I am one of the core contributors and maintain the standard library.