Paul Schoenfelder

Last update on December 6, 2022

paulschoenfelder@fastmail.com • 952-905-4095 • bitwalker.org
Saratoga Springs, New York

About Me

I'm an experienced software engineer with a love for open source, compilers and distributed computing, and a passion for developing high-quality software.

I started out in IT as a network engineer, working with Cisco equipment, but it wasn't long before I had delved in to writing scripts for automation. Once I had a taste for code, I knew it was where I belonged, and I was relentless in pursuing an open developer position at the same company. I was given the opportunity, and I've been a software engineer ever since.

I was thrown in to the world of software engineering, and had to learn fast. I've soaked up books, blogs, talks, whatever I could. I have become proficient in numerous languages, technologies, and practices along the way, and achieved mastery in many of them.

I'm active in open source. I experiment with embedded development. I love functional programming and the challenges of concurrency and distribution. But most of all, I love working on compilers and developer tooling.

Professional Experience

DockYard — Remote

Principal Engineer — Jan '21 -- present

I returned to DockYard to continue work on Firefly, the compiler project I spearheaded during my previous stint at the company. The CEO who originally backed the project came back to the company, and asked if I'd come back and pick up where I left off, with plenty of support - an offer I couldn't refuse! We're currently aiming to ship the first beta release by the end of the year.

Distru — Remote

Senior Software Engineer — Jan '20 -- Jan '21

At Distru, I spearheaded a number of different efforts to improve software reliability and turnaround time for fixes and new features. The most significant of these has been migrating the platform from its Heroku roots to Google Kubernetes Engine, consolidating our infrastructure under GCP, ensuring all of it is managed via Terraform and Helm, setting up a fully automated self-service continuous delivery pipeline, and providing a cutting edge observability stack for engineers to better understand the runtime behavior of the platform and enable them to troubleshoot issues quickly and effectively. By the time I left the company, this had all been accomplished, and I was able to hand off all of what I had built to a pair of more junior engineers who were interested in maintaining it going forward.

DockYard — Remote

Principal Engineer — Oct '17 -- Jan '20

My role at DockYard took two shapes: first as software architect on client projects, primarily focused on deployment and infrastructure; second as a technical lead on various R&D projects. I was originally hired to work on my open source projects, namely Distillery, and get releases integrated into Elixir Core, which happened in Elixir 1.9. The last year of my time at DockYard was spent working on Firefly, an ahead-of-time compiler for Erlang and Elixir that targets WebAssembly. I left because the CEO was stepping down to go work on other things, and the new CEO intended to shelve Firefly.

FireEye — Remote

Senior Staff Software Engineer — Jan '17 -- Oct '17

My role at FireEye was focused on the implementation of a concurrent, distributed, throughput-heavy workflow and data processing engine for security orchestration, built in Elixir.

Exosite — Minneapolis, Minnesota

Senior Web and Infrastructure Engineer — Oct '15 -- Jan '17

Lead developer on some of Exosite's most critical core services, built on Erlang and Elixir, with a few in Go (with parts written in C). Became very proficient with an array of AWS services, as well as Kubernetes in order to implement an OpenShift-based hosting platform for customer and internal applications. The applications I built were distributed, highly concurrent infrastructure services, with all the resultant challenges. One of the more interesting aspects of this role was building a scripting engine on top the Lua virtual machine that was integrated into a Go service invoked whenever commands were issued to/from devices connected to the platform.

My LinkedIn profile has a more complete list of my work experiences/military service, along with recommendations.

Notable Open Source Projects

Firefly

Technical Lead/Maintainer — Jan '19 -- present

Firefly is an ahead-of-time compiler for Erlang and Elixir that can target a variety of architectures, of which WebAssembly is of primary interest. It can be thought of as an alternative to the BEAM virtual machine, except rather than a virtual machine, Firefly compiles to native code. In addition to the compiler, Firefly also includes a runtime that allows it to integrate with browser APIs when targeting WebAssembly. It is built in Rust and C++, and builds on top of MLIR/LLVM.

Distillery

Author/Maintainer — May '16 -- May '20

Distillery is a release-management tool for Elixir applications, effectively a redesign/rewrite of ExRM, the original tool I wrote for the same purpose. It was the primary and officially recommended tool for deploying Elixir-based applications, and formed the basis of the implementation that was introduced into Elixir 1.9. It was a very active OSS project, with many contributors, used by a large proportion of the community.

Timex

Author/Maintainer — Nov '13 -- present

Timex is the richest date/time library for Elixir projects. It provides functionality that even most standard library packages in other languages fail to offer. It provides rich parsing/formatting facilities (including locale-awareness), timezone-aware date/time arithmetic, a modular architecture, operations on intervals, and more.

My GitHub profile contains a list of the many other projects I'm involved in, feel free to take a look!

Conference Talks

2019: CodeMesh LDN, ElixirConf, LoneStar Elixir

2018: The Big Elixir, LoneStar Elixir, CodeBEAM SF

Older: ElixirConf x2, LoneStar Elixir

Podcasts

2021: Rustacean Station

2020: Elixir Wizards

2019: Smart Software with SmartLogic, Elixir Talk, ElixirMix

Skills

Languages: Rust, Erlang, Elixir, C, C++, Go, Javascript, and proficient in many more

Theory: Type theory, compilers, interpreters, data structures and algorithms, distributed consensus, messaging and protocol design

Dev/Ops: Networking, administration of Linux and Windows systems, deployment automation, container orchestration (Kubernetes), continuous integration/delivery with internal and external build systems, GCP/AWS services

Development: I've built applications ranging from simple scripts to desktop applications with graphical interfaces. Single-threaded console utilities, to distributed, heavily-concurrent backend services where performance is critical. I've worked on scripting systems, web applications, some embedded software, and designed and developed remotely upgradeable, hot-pluggable services for gateway devices. If it's something new, I make a point to learn everything I can about the domain, and then dive in.

Interests

Non-exhaustive and in no particular order: programming language design and theory, woodworking, artificial intelligence, game development, music, homebrewing, the outdoors, and more! I love to learn, so there is almost always something new I'm picking up.