Curtis McEnroe

cmcenroe.me curtis@causal.agency github.com/programble

I'm interested in learning the intricate details of systems and building well-crafted tools. I want to make software that works.

Here are some of the technologies I've enjoyed working with:

- C, make, mdoc, zlib, curses
- Rust, Protobuf, American Fuzzy Lop
- Ruby, TypeScript/JavaScript, PostgreSQL, SQLite
- FreeBSD, NetBSD, macOS, Arch Linux

I write code because I enjoy solving problems and learning how to do things myself. I've worked on web API clients, stylesheets, development tools, data structures, hobby operating systems and creative projects. For some time I helped out in Rust IRC channels, which was helpful for my own learning. I also try to write about new things I've learned on my technical blog.

Funemployed at home, August 2017 –

Away from work, I tried to focus on creativity and communication. I created a public collaborative ASCII art project as an SSH service using memory-mapped files, local sockets and curses. I started a terminal IRC client, also available over SSH, where I focused on improved UX. I wrote a PNG encoder and optimizer, and used bugs in my code to create visually interesting "glitch" images. I designed my own terminal colour scheme and programmed a bitmap font editor.

Software developer at Hopper, May 2017 – August 2017

I learned a little bit of Scala, HBase and Elasticsearch. I didn't stay long due to lasting negative mental health effects of my previous job.

Software developer at AdGear, October 2015 – April 2017

I designed and implemented a replacement for the primary ETL providing timesensitive state to a global real-time bidding system. The design solved performance, correctness, and concurrency issues in the legacy system, and served as a strong example of Rust in the company. I vectorized a hash table implementation in a specialized key-value database service to take advantage of SIMD instruction sets and measured the performance increase with microbenchmarks. I set up fuzzing jobs for many Erlang NIFs written in C to try to find edge case crashes, which can bring down the BEAM.

Software developer at Busbud, December 2013 – April 2015

I migrated geographic data from CouchDB to PostgreSQL, and replaced Elasticsearch with Postgres materialized views and trigram indexes. This centralized data and improved performance. I introduced promises and functional programming to the Node.js server, which facilitated writing organized, asynchronous code. I helped redesign the interfaces and data models for integrating partner APIs into Busbud’s, which were previously too restricted and lossy.