Adam McCullough

Experienced Backend/Infra engineer, passionate about agility and reliability through types, tests, and observability.

✉ TheWizardTower@protonmail.com
 github.com/TheWizardTower
in linkedin.com/in/adammccullough
 thewizardtower.github.io

WORK EXPERIENCE

JULY 2022 - OCTOBER 2023 (FT)

FP Complete
Senior Software Engineer

Role Served as an expert in both robust infra and maintainable, scalable code, available to help customers that had secured a contract.

Server Overhaul Spearheaded a core server rewrite for a customer that suffered from long build times, brittle code, and flaky tests.

- Consolidated codebase from leveraging two separate web frameworks to a single framework to ease maintenance.
- Reduced build time and improved test coverage
- Simplified, commented, and typed extensive parts of the code base
- Published servant-combinators to extend Servant's capabilities
- Gave an internal presentation to engineering team about the ergonomics and trade-offs of the two libraries

Rust Conversion Wrote a utility to convert a Haskell server binary to a collection of flat files, enabling us to host the same content on Cloudflare, rather than on an AWS instance.

AUGUST 2020 - JUNE 2022 (FT)

Daisee
Lead Systems Engineer

Role Served as Head of Operations for a SaaS company that was just graduating from start-up to scale-up

Infra Overhaul Spent significant time debugging and overhauling the existing legacy terraform code

App Redesign Lead the infra-side of an app re-design

- Migrated to an on-demand instance approach managed by ECS, which significantly reduced costs without affecting throughput
- Consolidated to a multi-tenant approach, so adding new customers did not require an infra change

Haskell Work Migrated prod code to a new program monad in core-program

- Included many optimizations to run-time configuration and developer conveniences for things like logging or command line argument parsing

Observability Added observability to Haskell code via Honeycomb and a purpose-built Haskell OpenTelemetry library.

- This helped enormously in both finding fruitful places for optimization work, as well as quickly understanding production outages

MARCH 2018 - JULY 2019 (FT)

Facebook
Production Engineer

Role Worked as a Production Engineer (PE) on the Feed and Stories Team (FAST)

- FAST was responsible for News Feed, the centerpiece of the Facebook web experience, both w/r/t MAU and ad revenue.

TOOLS

Haskell, Rust, Python, Bash, Terraform, Postgres, Redis, Kubernetes

SPEAKING ENGAGEMENTS

Rust's Borrow Checker Proven Correct
LambdaConf 2019

Gave an overview of bugs in various compilers, and how difficult they are to identify and debug. Describe how the Rust devs used formal verification to prove that the semantics of the borrow-checker would produce programs without race conditions. Conclude with some motivating examples to demonstrate why various designs fail the typechecker, and what race conditions they prevent.

Monad Transformers for the Easily Confused
LambdaConf 2018

Begin with a review of Monoid, Functor, Applicative, and Monad typeclasses. Demonstrate that Monads do not compose, then demonstrate how Monad Transformers solve this problem, first with the IdentityT monad, then the MaybeT monad.

Lambda Calculus for the Easily Confused
LambdaConf 2017

Walkthrough of the mechanics of Lambda Calculus, with a particular emphasis on how these mechanics can give an intuition for things in Haskell, like partial function application or higher-kinded types. Concludes with a demonstration of the Y combinator.

VOLUNTEER WORK

Board Member, IT Director, People's Pantry of Ferry County

Organized and ran a Haskell class at IMVU, 2016

President, ASULUG, August 2010-August 2012

- FAST was a large team (1.5k at intern season), while the embedded PE team was much smaller (7)
- We oversaw server infra, compute capacity planning, building and monitoring new releases of the software that ran news feed, aggregator

Oncall Member of the push oncall rotation for Aggregator

- identify regressions in engagement, ad revenue, or new crashes arising from race conditions or Thrift configuration mismatches

Project Added per-diff Address Sanitization (ASAN) canary tests for aggregator

- Required working cross-functionally with Sandcastle (CI), Phabricator (code review + test stewardship), as well as securing server capacity, and deploying into production use

Stewardship Also spent time refactoring legacy C++ projects to use safer, more modern idioms, which lead to updating and improving code for the iterators of some data structures in the Folly library.

NOVEMBER 2016 - SEPTEMBER 2017 (FT)

IMVU
Software Engineer I

Role Worked as a developer on the FIRE team

- FIRE was responsible for finding and fixing problems in a diverse legacy codebase, including PHP, Haskell, JS, and CSS.

Project Maintained and extended a customer marketing tool

- Tool extracted data about customers, such as if they'd spent money, when they joined, if they had a "special someone", etc.
- Then used this information to tailor promotions for customers.

Project Designed, implemented, and deployed a backup-and-restore system for Scylla

- Scylla is a C++ re-implementation of Cassandra
- This required extensive reverse-engineering, as documentation for the backup procedure was extremely thin at the time.
- This also managed and pruned backup files according to a retention policy, to help with data storage volume constraints

JUNE 2013 - NOVEMBER 2016 (FT)

IMVU
Systems Engineer II

Role
- Maintain, triage, and resolve issues impacting the production cluster or service quality, 24/7
- Root-cause incidents, identify steps to prevent it repeating in the future, then implement those steps.

Oncall Carry a pager for issues that affected service quality, 24/7

- Responsibilities include responding to pages within 15 minutes while oncall, and communicating to the company: observed impact, path to remediation, and time to resolution.

Project Replaced Varnish caches with Apache Traffic Server

- Varnish had a significant memory leak on the kernel version deployed to prod. Switching to ATS significantly reduced oncall load and improved service quality.