Adam McCullough

github.com/TheWizardTower
linkedin.com/in/adammccullough

Software Engineer with a systems background, passionate about Functional Programming, correctness, and reliability.

EXPERIENCE

FP Complete, Fully Remote, — *Senior Software Engineer*

July 2022 - October 2023

Worked in a team to triage and cure problems in a legacy codebase. Identified that several different libraries were being used to solve similar problems across the codebase, contributing to team siloing, build times, and build unreliability. Spearheaded an initiative to migrate a hybrid Yesod/Servant setup to use Servant, which involved publishing a library to extend Servant's capabilities (servant-combinators). Improved test coverage, added comments, and refactored many functions to be smaller along the way, without changing functionality. Gave an internal engineering presentation on the ergonomics and design choices of Yesod and Servant.

Daisee, Sydney, Australia — *Lead Systems Engineer*

August 2020 - June 2022

Refactor and simplify a terraform codebase that managed prod infrastructure. Deployed a new production cluster, fixing numerous reliability concerns and several bugs where the codebase would not correctly re-create a full cluster from a clean-slate deployment.
Investigated, developed, demoed, and deployed the infrastructure for V2, which offered numerous improvements, including greater throughput, better introspection and monitoring, easier deployments, greater reliability, and more rapid iteration times.
Reviewed and wrote a significant amount of Haskell, both in business logic and production monitoring. Provided feedback and expert advice on design decisions and architecture choices to the VP of Engineering.

Facebook, Menlo Park, CA — *Production Engineer*

March 2018 - July 2019

Worked as a PE embedded in the Feed And Stories Team. FAST was responsible for maintaining the Facebook News Feed, which was the centerpiece of the Facebook web experience, both in terms of monthly active users, and, consequently, revenue. As such, it was a very large team (1,500 people during intern season), with a comparatively small team of PEs (7). PEs oversaw both operational matters, such as server infra and capacity management, as well as monitoring and deploying new builds of the software that ran News Feed, the C++ binary responsible for ranking the posts in the news feed in particular. We were particularly interested in performance regressions, new crashes arising from new race conditions or configuration mismatches in the Thrift data structure definitions, and impact on advertising revenue.
My main project on this team was to add per-diff address sanitization (ASAN) canary tests, to surface these problems while the developer was working on their code, rather than waiting for the CI process to catch it. This involved working with several teams, including the Sandcastle team (CI), Phabricator team (code review + test maintenance), securing server capacity, and deploying it into production.
I was also on the push oncall rotation for Aggregator, which involved identifying regressions in the categories previously mentioned, and working with SWEs to identify and fix the root cause issue.

SKILLS

Functional Programming

Autodidact

Compulsively curious

Public Speaking

Mentorship

Passion for Programming and Engineering

Accomplished at troubleshooting root causes of customer impact

Linux Administration

Systems Architecture and Design

LANGUAGES

Haskell, C++, Rust, Python, Bash

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

PERSONAL ACHIEVEMENTS

Amateur Extra Radio License, callsign AG7YC

Stackage library, servant-combinators

IMVU, Redwood City, CA — *Software Engineer I*

November 2016 - September 2017

The FIRE team was tasked with identifying and fixing problems in a diverse legacy codebase of PHP, Haskell, CSS, and JS. My projects involved extending and maintaining a customer marketing tool that extracted data about users, such as if they had spent money, when they joined, if they had a "special someone" to offer tailored promotions to them. This necessitated extending the UI the marketing team used to put together these marketing promotions to support the new predicates they requested.
I designed, implemented, and tested a backup-and-restore system for Scylla, a C++ re-implementation of Cassandra, to replace a large vertical Redis shard acting as a data store for our News Feed endpoint. This involved a sizable amount of experimentation and reverse engineering, as the documentation for this was fairly slim.
The FIRE team also worked closely with Ops to help triage and mitigate site outages, as well as participate in post-mortems to identify root causes, and take follow up tasks to implement appropriate fixes.

IMVU, Redwood City, CA — *Systems Engineer II*

June 2013 - November 2016

IMVU's ops team managed the infrastructure side of the outfit, including managing CPU, memory, and storage capacity, ensuring correct,complete, and reliable configuration management, defining and meeting data retention SLAs, and effective stewardship of our monitoring and paging solutions.
While oncall, we were expected to identify and manage mitigation of site issues our monitoring alerted us to, communicate the current situation and remediation path to the company as a whole. We also worked with other teams to identify issues affecting production health and assist them in improving service quality.
Most work was driven by prod cluster interrupts, however my largest project was replacing the caching load balancer used for our server side image rendering from Varnish to Apache Traffic Server. This resolved a memory leak that Varnish exhibited when running on our prod version of Ubuntu. This resulted in both more reliability and a significant reduction in oncall load.
Our prod cluster used Memcache, Redis, MySQL, Apache running PHP, Nginx, HAProxy, as well as Haskell binaries. We used CFAgent to provision and maintain production hardware configuration, and post-mortemed service quality incidents to identify root causes, then identify and implement follow-up fixes.

EDUCATION

Arizona State University, Tempe, AZ — *Computer Systems Engineering*

August 2007 - May 2013

SPEAKING ENGAGEMENTS

Lambda Calculus for the Easily Confused

Speaker - 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.

Monad Transformers for the Easily Confused

Speaker - LambdaConf 2018
Begin with a review of the 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.

Rust's Borrow Checker Proven Correct

Speaker - 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.