

Michael Dresser

michaeldresser.io | michaelmdresser@gmail.com

STACKWATCH (KUBECOST) SENIOR SOFTWARE ENGINEER

Feb 2022 - Present | Remote

- Served as tech lead and de facto product lead for Kubernetes (K8s) cost optimization. Guided engineers, interviewed stakeholders, and created and reviewed design proposals.
- Designed/implemented fast quantile-based K8s container right-sizing recommendations using log-linear histograms which scaled to 250,000+ container environments. (Go)
- Designed a filtering language for core APIs. Built a lexer, parser, and interpreter for the language, enabling previously-impossible queries. (Go)
- Extended a K8s controller, providing simple APIs for cost optimization. (Go)
- Enabled users to upgrade to new versions by leading design and implementation of schema backwards-compatibility. (Go)
- Interviewed, onboarded, mentored, and supported engineers.
- Demoed in K8s community meetups and podcasts.

STACKWATCH (KUBECOST) FULL STACK ENGINEER

Jan 2021 - Feb 2022 | Remote

- Built/marketed an open-source CLI. 400+ GitHub stars. (Go)
- Built a feature to determine if K8s workloads can be rescheduled onto spot nodes, leading to >50% savings. (Go, TypeScript, React)
- Reduced benchmark feedback loop from 15+ to 3 minutes. (Go)
- Built a nightly CI+release process, enabling users to test bleeding-edge features. (GitHub Actions)

GOOGLE SOFTWARE ENGINEERING INTERN

May 2020 - Aug 2020 | Remote

- Contributed to K8s test infra, simplifying E2E test workflow in an environment running >90,000 tests/month. (Go)
- Contributed patches to repos in the K8s project. (Go)

TWILIO SOFTWARE ENGINEERING INTERN

May 2019 - Aug 2019 | Denver, CO

- Designed+built a microservice to enable testing. Moved service into Kubernetes. Drove adoption in 3 teams. (Go)
- Created service to recover failed operations by recirculating Kafka messages. (Go)

LOGRHYTHM SOFTWARE ENGINEERING INTERN

Jun 2018 - Aug 2018 | Boulder, CO

- Developed microservice to write logs to GCP Pub/Sub. (Go)
- Built microservice for preventing PII leaks. (Go, Elasticsearch)

EDUCATION

UNIVERSITY OF COLORADO BOULDER

B.S. AND M.S. IN COMPUTER SCIENCE

GPA: 4.0 / 4.0

SKILLS

LANGUAGES

Go • Python • TypeScript • Javascript • Clojure • Bash • Rust • Lisp • SQL • Scala • OCaml • Nim • Java

TOOLS

Linux • Kubernetes (K8s) • Docker • Git • SQLite • Prometheus • GCP • AWS • GitHub Actions • Jekyll • Spark

PROJECTS

BOOK CLUB TOOLS

Python, Rust

Tools to support a weekly book club. Epub processor to convert books into weekly reading sections with estimated reading times. Discord bot for voting on books using quadratic voting.

TEEMO TOOL

Javascript, Mithril.js, Twitch Website that gathers and displays (fake money) live bet data from Twitch stream chat. Provides information advantage to bettors.

THESIS: ARUGULA

Scala

Arugula is a probabilistic programming language that I designed and implemented. It has special semantics that support delayed sampling of distributions combined with symbolic inference, which improves the accuracy of estimators.