JACOB COLVIN

Platform, Site Reliability, and Software Engineer

Cincinnati, Ohio | me@jacobcolvin.com | jacobcolvin.com | colvinjm | MacroPower

ABOUT

My expertise lies in **Site Reliability Engineering**, enriched by a deep focus in **Software** and **Platform Engineering**. My passion lies in building robust **developer platforms**, crafting maintainable **infrastructure as code**, and architecting systems in a way that is simple and straightforward. My experience as a developer, with a focus on **Go** and **Python**, enables my ability to create platforms that are not only reliable and maintainable, but also enjoyable to use.

EXPERIENCE

84.51° / The Kroger Company

Hybrid / Cincinnati, Ohio — May 2016 – Current

Senior Site Reliability Engineer

April 2021 – Current

- Successfully transitioned our **Kubernetes (AKS)** platform from being a set of internal-only clusters, to a production-grade platform hosting multiple critical applications that were collectively responsible for driving over $100MM operating profit; participated in an **on-call rotation** to ensure compliance with SLAs.
- Designed a fully **GitOps** workflow for Kubernetes cluster management, using **FluxCD**, **Terraform**, and **Renovate**, which reduced manual cluster administration work by over 1,000 hours per year.
- Rolled out **Datadog** as a unified observability platform for both **Azure** and on-premises systems, leveraging **Vector**, **Fluentd**, and **OTEL Collectors** for assorted telemetry pipelines; this enabled the deprecation of multiple disparate observability solutions and saved over $500k annually.
- Maintained multiple on-premise **Cloud Foundry** / **Tanzu Application Service** instances; supported cloud migration efforts which moved applications from Cloud Foundry to Kubernetes.
- Developed **Python** application for scaffolding template repositories, as well as centralized **Helm charts** and related **GitHub Actions**, which collectively saved developers more than 3,000 hours during the first year in production.
- Empowered developers to efficiently and independently troubleshoot their applications by acting as lead subject-matter expert for observability platforms; headed support and maintenance of all observability tooling, including **Datadog**, **Grafana**, **Prometheus**, **Thanos**, **Jaeger**, **Dynatrace**, **Fluentd**, **OpenTelemetry Collectors**, and **Vector**.
- Developed **Python** library to centralize instrumentation for **Prometheus** metrics, **OpenTelemetry** tracing, structured logging, and **pprof** profiling, with support for **FastAPI**, **gRPC**, **Databricks** notebooks, and more, which both delivered a positive and consistent experience with our observability platforms, and saved developers over 1,000 hours.
- Led development of multiple **Prometheus exporters** using **Go**, utilizing **Redis** for caching, distributed workers with pub/sub and leader election, **CEL** for custom rule evaluation, **Cue** and custom tooling for validation, and Go text templating.
- Created multiple **Jsonnet** libraries for **Prometheus** and **Alertmanager** configuration, saving over 100 hours per year.
- Contributed fixes to **Grafana**, **Thanos**, **Jaeger**, and other upstream **Git** repositories.

Versatilist Engineer

January 2020 – March 2021

- Created and administered multiple **Azure** environments using **Terraform** in Azure DevOps; said environments frequently included resources such as **Databricks** workspaces, **Datafactory** instances, **Key Vaults**, **Storage Accounts**, and **Postgres** databases.
- Enhanced our **Prometheus** ecosystem by adding high-availability and long-term storage via **Thanos**, thus allowing consumers of Prometheus to track SLOs and KPIs over years instead of days.
- Assisted a mix of seven development and data science teams with **Azure** migration, **Terraform**, **Grafana** dashboards, **Prometheus** alerts, **GitHub Actions**, and **Azure DevOps** pipelines; acted as technical liaison to increase speed and accessibility of support.

ICT Co-op

May 2016 – December 2019

- Designed a **C#** API layer over several legacy systems, and a SPA using **TypeScript** with **React**, to both assist with support, and improve velocity towards deprecation of legacy systems.
- Created a **Prometheus exporter** for **SonarQube** data, along with corresponding dashboards, rules and alerts, to allow security team to gather relevant KPIs.
- Automated **Red Hat Linux** VM deployment through **ServiceNow**, via interactions with **Satellite**, **vCenter**, **SolarWinds** and **Ansible**.
- Designed a custom web framework using **PowerShell** and **Bootstrap** for executing administrative tasks and aggregating events and metrics from many distinct products.
- Automated many miscellaneous tasks using **PowerShell**, **VBA**, **Bash** and **PL/SQL** in Bash; created and documented **Automic** workflows with **Bash** and **SAS ODS**.
- Interfaced with **BOSH** and **Hadoop** to design custom **Pivotal Cloud Foundry** monitoring solution.

EDUCATION

B.S. Information Technology, University of Cincinnati

Aug 2015 – April 2020

- Summa Cum Laude, CeDiD: 20H0-3K9Z-JEND

KEY PROJECTS

OmegaGraf

jacobcolvin.com/OmegaGraf

- An open-source project that seeks to completely automate vCenter monitoring, by orchestrating a small ecosystem of containers, including Telegraf, Prometheus, and Grafana.
- Paper: https://scholar.uc.edu/concern/student_works/jw827c971
- Technologies: Docker, C# / .NET Core, TypeScript, React

Homelab

MacroPower/homelab

- Infrastructure-as-code for my homelab / personal cloud. Defines multiple interconnected Kubernetes clusters, spanning across bare metal (Talos) and multiple Hetzner cloud environments (k3s-on-MicroOS).
- Technologies: Talos, Cilium, ArgoCD, Helm, Jsonnet, Terraform

Prometheus Video Renderer

MacroPower/prometheus_video_renderer

- Just for fun, completely impractical tool that allows you to encode audio and video as Prometheus metrics.
- Featured on the Grafana blog:
  https://grafana.com/blog/2021/07/30/how-to-use-grafana-and-prometheus-to-rickroll-your-friends-or-enemies
- Technologies: Go, Jsonnet

Analytics Panel Plugin

MacroPower/macropower-analytics-panel

- Grafana panel plugin that injects JavaScript into dashboards, which reports user session information to a backend Go server, which in turn exports Prometheus metrics for display in (you guessed it) Grafana.
- Over 1MM downloads, featured by Giant Swarm:
  https://www.giantswarm.io/blog/grafana-ception-or-how-we-do-grafana-analytics-giant-swarm
- Technologies: TypeScript, React, Go

Wakatime Exporter

MacroPower/wakatime_exporter

- Prometheus exporter and Grafana dashboards for Wakatime coding statistics.
- Over 100k downloads
- Technologies: Go