Steffen Reindl

Email: steffen@malignat.us || Portfolio: github.com/MordecaiMalignatus

Experience

Stripe, Senior Software Engineer, (Berlin, Germany) **February 2024 – August 2024**

Maintained core CI infrastructure under tight constraints and SLOs, built on top of AWS.

Improved CI UX for Stripe's developers, reducing feedback cycle times and improving velocity.

Migrated Docker across several major versions, updating a core component of the infrastructure with no downtimes and no notice to users.

Interfaced with several teams to hunt down complex performance issues inside a distributed system.

Terramate, Senior Software Engineer, (Berlin, Germany) **June 2023 – November 2023**

Developing and maintaining the Terramate CLI, importing ideas from build tools like Bazel and Nix to add features and improve performance.

Creating and architecting the newly-created Terramate Cloud product backend, together with significant input on business and product development.

Started and popularized incident reviews/retrospectives to learn from things having gone wrong.

Wayfair, Senior Platform Engineer, (Berlin, Germany) **July 2021 – April 2023**

Maintaining and developing tools for Vault, Consul and Puppet at Wayfair as part of the Configuration Management Team.

Developed a unified distribution platform on top of Vault, allowing other developers to not worry about secret distribution. For this we also built multiple interfaces for automated access and distribution.

Planned, communicated and executed a several-major-versions upgrade of Vault with no downtime.

Created and maintained developer-facing operational tools around Vault, incorporating principles of Cognitive Systems Engineering.

Held multiple internal talks about the space of Resilience Engineering, Human Factors & Ergonomics.

PriceHubble, Site Reliability Engineer, (Berlin, Germany) **November 2020 – June 2021**

Maintained and developed tools for a large-scale Kubernetes cluster. This served as the basis of infrastructure to product teams.

Started a cultural shift towards developer empowerment and autonomy, away from a classic Ops/Dev split. The incentives and final structure were modelled after the classic SRE literature.

Transitioned company to new BI infrastructure based on GCP's BigQuery.

Ryte, Backend Software Engineer, (Munich, Germany) **May 2017 – Sep 2019**

Maintained a Scala web crawler central to the company, distributed over YARN and Flink, analyzed by a Scala service written in Spark. This system is running on AWS EMR, and was subsequently improved to support Javascript rendering and crawling single-page applications by controlling Chromium.

Education

Technical University of Munich, Incomplete B.Sc. Computer Science, (2016 – 2018)

Technologies

Programming Languages	Confident engineering skills in Go, Ruby, Rust, Python, and Elixir.
Infrastructure as Code	As much as I can I rely on Infrastructure as Code, through whatever tool is most appropriate. These include: Terraform, Helm, CloudFormation, Puppet, Ansible, Nix.
Kubernetes	I have maintained and developed for several clusters, mostly using GKE.Istio and Helm were also used.
Platforms	I have used AWS and GCP to power large-scale CI systems, web crawlers, analysis pipelines and ML training clusters, website back-ends and have developed additional features for them based on the need of other team mates.
Databases	Primarily worked with PostgreSQL and variations of SQL databases. In addition I also made use of BigQuery, DynamoDB, MongoDB, Kafka and various caches.
CI/CD	Experience in setting up the software, process and tooling needed to continuously test and deploy services and products. Ability to troubleshoot flakey tests, reliability issues and aggressively taking aim at problems that make engineers less confident in CI.

Interests & Focuses

Observability	Modern, complex systems are hard to grasp and harder to change successfully, and so having insight into what is happening in production is invaluable. I build systems as transparent and observable as I can.
Cognitive Systems Engineering	I focus on the sociotechnical aspects of a system, rather than on either, human or computer, allowing me to optimize for the overall functionality. This means, for example, designing to avoid alert blindness, and designing to assist debugging in case of failure.
Systems Design & Resilience	I design systems for failure, which results in simple, but reliable architectures, easy to maintain and expand.
Developer Tooling & UX	I am very fond of having good tools to work with, and will create and maintain tools that enable and simplify future changes.
Learning From Incidents	As frustrating as they are, production outages and incidents provide valuable opportunity for learning and improving, and as such, should not be wasted. I tend to set up incident reviews and analyses, and teach people how to do them.