

# Paul Leimer

pbleimer@gmail.com | Contact for info | Austin, Tx | linkedin.com/in/paulleimer

## Summary

I am an experienced full stack software engineer in scalable web app systems. I have done 3D web development as well, but I am open to all opportunities related to application and platform development in the Austin area. Let's talk!

## Skills

- **Languages:** Golang, Typescript / Javascript, SQL, Bash, Python, Java, C/C++
- **Backend:** REST, gRPC, Protobufs, Temporal, json-schema
- **Frontend:** ReactJS, ThreeJS (webGL), HTML, CSS
- **Databases:** Postgres, MySQL, MongoDB
- **Platform:** Kafka, Kubernetes, Splunk, Prometheus, Grafana, Docker, Jenkins, Github Actions

## Work Experience

### Tesla

Senior Software Engineer

Mar 2023 — Jun 2024

Led the full stack development of a general purpose 3D digital-twin visualization platform. Managed the full project lifecycle, from architecture design to feature prioritization, resulting in a configurable, site agnostic platform that enabled customers to build 3D virtual representations of production areas and bind live-streamed production data to virtual assets

- Led the planning and development for both frontend and backend services, utilizing Postgres (SQL) for persistence and streaming (Listen/Notify + Websocket), Golang for backend services, middleware and Websocket + REST APIs, and ReactJS + Typescript paired with ThreeJS for primary UI functions
- Collaborated with a team to run extensive architecture and UI design sessions, create development plans and project roadmaps. Mentored engineers and interns to onboard and contribute quickly.
- Planned and implemented an in app editor, allowing users to build visualizations in the web browser. Implemented extensive feature sets in Golang, ReactJS, Typescript and ThreeJS involving complex state management in the frontend and backend, 3D algorithms and GPU optimizations
- Implemented extensive design and optimizations for 3D environments, including an algorithm for  $O(\log(n))$  broadphase collision detection (BVH + SAH rotation), increasing max mesh count by 20,000% compared to other open source solutions
- Architected a general component relational data model for tracking components in a spacial environment that utilized json-schema for parameter validation
- Built middleware for OAuth (OpenID) flows in the frontend and backend, utilizing Azure IAM for authentication and a proprietary OPAL solution for RBAC authorization
- Setup project backend, frontend project scaffolding and building blocks such as dependancy injection patterns, config patterns, streaming, state management, 3D primitives, etc to ease developer onboarding and project maintainence
- Integrated data streams from various external systems including Airflow, OPCUA (PLCs), application reporting DBs and Kafka topics
- Managed all aspects of the project development, including automation tooling (codegen, db migrations, testing), monorepo services structure, integration test suites (testify, playwright), quality assurance, and deployment with Docker, Kubernetes and Jenkins
- Spearheaded adoption of the platform throughout the company by running tech talks, customer and director demos and working directly with customers in multiple regions to evaluate features and designs

Software Engineer

Jan 2022 — Feb 2023

Oversaw the provisioning of an event sourced CQRS manufacturing execution system in Ausin for cell producton, spearheaded various features and integration efforts

- Led several integration efforts between multiple teams to synchronize produced component state in an event sourced CQRS MES systems and other downstream and upstream transactional systems.
- Developed gRPC APIs and Kafka event APIs using Golang and Temporal to asynchronously update eventually consistent state of produced components and developed Golang services to synchronize the state across multiple systems
- Coordinated cross team efforts to solve infrastructure issues for our apps, including identifying a MTU size misconfiguration in a Kubernetes cluster load balancer that resulted in periodically dropped packets
- Participated in on-call rotations for the CQRS system. Performed swift diagnosis and remediation in production environments, hot fixes and rectified data discrepancies to insure uninterrupted service delivery
- Instrumented services with metrics and logs, built Grafana and Splunk dashboards for increased app observability

### Red Hat

Software Engineer

May 2019 — Dec 2021

Worked with a team to implement and document the unified Service Telemetry Framework, the official mulit-regional OpenStack + OpenShift (Kubernetes) cloud health monitoring framework

- Led architectural overhaul of core gateway golang service, implementing a micro-kernel architecture that drastically improved development cycle time by minimizing technical debt and increased message throughput performance by more than 3x
- Implemented various services and APIs in Golang, Python, NodeJS and C that improved OpenStack and OpenShift cloud health monitoring and automated cloud provisioning processes
- Standardized monitoring dashboards with Grafana, Prometheus and Elasticsearch
- Advanced RedHat initiatives in open source projects by working with upstream communities (CentOS, RDO and Fedora) to package and release services and features as RPMs and Docker images

### Hughes Network Systems

Software Engineering Intern

May 2018 — Aug 2018

- Developed a web apps using a MEAN + python stack to enable the configuration, automation and execution of tests that hardware engineers must run frequently, reducing test time from hours to minutes
- Built dashboards to configure and run tests that executed profiling tools between geographically dispersed satellite gateways, collected results via REST queries, persisted in MongoDB and displayed results in Angular JS dashboards
- Implemented system for persisting test configurations and scheduling test execution cron jobs

### ATI Industrial Automation

Software Engineering Intern

Aug 2017 — Dec 2017

- Designed, built and documented firmware and hardware design for an ISP firmware flasher using Atmel MCUs and C++, replacing a legacy system and decreasing flash times by over 300%
- Characterized noise of integrated circuit operation in tool-changer sensors for accurate estimates on product documentation ( $< .1\%$ ) using Python and statistical models

## Education

### North Carolina State University

Bachelor of Science

Aug 2014 — May 2019

- Computer Engineering
- Electrical Engineering

## Awards & Recognitions

- **2nd Place Engineering Design Fair:** Won 2nd place in the educational category by building a mobile game in Flutter that made power grid optimization fun and educational for students
- **Eagle Scout**