

Martin Saul

Seattle, WA · +1-206-672-2212 · martin@saul.io

linkedin.com/in/martinsaul · martinsaul.com

Backend engineer specializing in control planes for fleets: orchestration, scheduling, and credential infrastructure. 15 years across Google, Amazon, and consulting, building distributed systems in Kotlin, Java, and Python. Architects and operates a self-hosted multi-agent AI platform that ships production systems under human direction.



Full detail & projects →
martinsaul.com

>EXPERIENCE

Software Engineer, Google Cloud, Engineering Productivity (Automation & Bots)

Kirkland, WA (Hybrid) · Jul 2024 - Present

- Co-designed elastic worker orchestration for a dev-infra automation fleet (7,500 hot instances across 15 production cells) on Borg, and owned it solo from POC through production. Replaced hardcoded single-template provisioning with purpose-built ephemeral workers that carry per-task binary versions, right-sized resource profiles, and completion-driven teardown.
- Tiered scheduling reserves the static warm pool for priority traffic while standard traffic shifts to dynamic workers; the rollout is projected to reclaim about half of production fleet capacity and drop the non-prod static footprint entirely.
- Redesigned fleet authentication so the orchestrator holds credential custody. Replaced per-worker refresh-token self-management with short-lived access-token vending, removed hardcoded secrets from source and credential logging, and migrated secrets to managed storage.
- Mentors junior engineers and hosts a SWE intern through Sep 2026.

Software Engineer, Google Cloud, Supply Chain

Seattle, WA · Apr 2021 - Jul 2024

- Sole designer of the real-time sourcing engine behind Google's internal hardware-logistics platform (Java/Kotlin): built the core engine and directed two engineers on client-application integration. It continuously re-plans up to ~10K concurrent orders through hourly reconciliation against inventory shifts and change requests, optimizing across tax codes, freight cost, and regulatory load constraints with packing-aware load estimation. Concept to production in six months.
- Built automated failure-triage for material-transfer errors. It classifies root cause (approved-source stockouts vs. third-party manufacturing delays) and routes tickets to the owning component and assignee by project context, material, location, and urgency, which cut reaction time on material-blocked builds from weeks to days and virtually eliminated orders lost to unowned failures.

Software Engineer, Google, Ads Integrity

Vancouver, BC · Jun 2020 - Apr 2021

- Built dynamic rule distribution for ads-abuse enforcement (Java): sensitive-data and abuse-signal matchers propagate to serving systems in seconds, without restarts. Replacing a manual commit-and-deploy cycle cut time-to-enforcement from hours to seconds and permanently removed ~14 engineer-hours/week of release toil.

SDE II, Amazon, S3 / Retail

Vancouver, BC · May 2019 - Jun 2020

- Planned and executed S3's production DNS migration from a legacy stack to native Route 53 across every zone. Any failed cutover would have meant zone-wide S3 unavailability, taking the broad set of AWS services that depend on S3 down with it. Completed with zero customer-facing incidents.
- Designed an extensible tax-calculation system (Java), initially scoped for Mexico and since adopted worldwide.

Earlier Experience

- **Connected**, Senior Software Engineer, Toronto · Aug 2018 - Mar 2019. IoT content-delivery microservices sustaining 300+ devices/sec (Java, Spring Boot, Kubernetes); ran weekly TDD training for 16+ engineers.
- **TWG**, Senior Software Engineer, Toronto · Sep 2017 - Aug 2018. High-throughput AWS messaging backend (Java, Guice, SQS, RDS); C# Active Directory sync for 15,000+ users and groups.

- **Stefanini** (Dell, AGCO), Senior then Lead Java Developer, Brazil · Nov 2014 - Jul 2017. Led a 4-engineer team; global schematics-sync pipeline (200MB+ XML; Spring Boot, Camel); credit-bureau rules-engine integrations.
- **Dell Brazil; ISul Tecnologia**, Software Developer, Brazil · 2011 - 2014.

>SELECTED PROJECTS

Claudio, self-hosted multi-agent orchestration platform (*Kotlin/Spring Boot, Go, Android; agent-assisted; private*)

- Architected and operate a platform that spawns and supervises fleets of concurrent Claude Code agents (scaled to available compute, no fixed ceiling): a REST + WebSocket control plane, full lifecycle (spawn/interrupt/pause/shelve/revive), inter-agent messaging with event-driven state subscriptions persisted across restarts, and git-worktree isolation for parallel agent work on shared repos.
- Designed and shipped the platform's MCP server (41 tools) plus a production transport migration from server-held SSE to a per-process stdio shim, which eliminated three restart/wedge failure-mode classes with zero client changes.
- Resilience: agents survive orchestrator restarts (tmux detach/reattach plus session resume); transient-failure retry uses jittered exponential backoff; OAuth credentials self-heal automatically.
- Directed the implementation of supporting Go microservices, all running on self-hosted GitLab CI, Traefik, and Docker: a service registry with TTL liveness, JWT token-vending backed by an AES-256-GCM credential vault, a bug tracker, and a knowledge-base API.

Terminus, distributed WebSocket terminal (*Rust/Tokio; agent-implemented under my design*). Master/relay shell access to NAT'd machines; Ed25519 device auth with PIN pairing; static musl builds shipped via a self-hosted signed APT repo.

[TeslaHitch](#) + [teslahacs](#), Tesla home-automation integration (*Kotlin, Python; public*). Self-hosted Tesla Fleet API proxy (Kotlin) paired with a Home Assistant custom integration (Python) for controlling vehicles from home automation.

>EDUCATION

B.S. Computer Science, PUCRS, Brazil · 2014 (completed while working full-time)

Spoken languages: English, French, Portuguese, Japanese