Naveen Iyer Senior Performance Engineer

Sunnyvale, CA · naveen.iyer@gmail.com · +1 408-555-0162 · linkedin.com/in/naveeniyer

PROFILE SUMMARY

- **Senior Performance Engineer** with **8 years** tuning **feed, messaging and search services** at **consumer internet scale**, specializing in **JVM profiling, workload modeling and latency SLOs**.
- Hands-on with load testing (**k6**, **Gatling**), profiling (**async-profiler**, Java Flight Recorder), tracing (**OpenTelemetry**, Jaeger), data (**MySQL**, Couchbase, Apache Kafka) and **Kubernetes**.
- Deep expertise in **GC tuning**, **tail latency analysis**, **capacity forecasting** and **performance gates in CI**, using the **USE method** to find the limit before traffic does.
- Works with **feed, SRE and infrastructure teams** on launch reviews, postmortems and quarterly capacity planning.
- Mentors **3 engineers** on profiling, runs the **weekly latency review** for the feed org, and wrote **5 internal performance playbooks**.

TECHNICAL SKILLS

Languages & Scripting:	Java, Python, Go, Scala, SQL, Bash
Load Testing Tools:	k6, Gatling, JMeter, Locust
Profilers & Runtime Tools:	async-profiler, Java Flight Recorder, pprof, Linux perf, G1 and ZGC tuning
APM & Observability:	OpenTelemetry, Jaeger, Grafana, Prometheus, InGraphs
Databases & Caching:	MySQL, Couchbase, Redis, Apache Kafka, EXPLAIN
Cloud, Containers & CI/CD:	Kubernetes, Docker, Azure, Jenkins, Terraform
Front-End Performance Tools:	Lighthouse, Chrome DevTools, Core Web Vitals, real-user monitoring

EDUCATION

UC San Diego — **B.S. in Computer Science** — La Jolla, CA · Sep 2014 to Jun 2018

WORK EXPERIENCE

LinkedIn — **Senior Software Engineer, Performance** — Sunnyvale, CA · Aug 2022 to Present

- Profiled the feed ranking service with **async-profiler** and Java Flight Recorder, cut allocation in the scoring loop and moved it to **ZGC**, bringing p99 latency from 380 ms to **120 ms**.
- Built per-member workload models from 90 days of production traffic and replayed them in **Gatling**, finding a cache stampede at **4.1x** peak before a major feed launch.
- Set the Kubernetes CPU requests and autoscaling targets for 30 feed services from load test results, raising average utilization from 32% to **58%**.
- Added **k6** benchmarks with latency thresholds to the **Jenkins** pipelines of 6 service teams, catching about **three quarters** of latency regressions before production.
- Agreed p99 budgets for the feed API with product and SRE, with burn-rate alerts and a release review, and kept monthly SLO attainment above **99.5%** for 2 years.

Yahoo — **Performance Engineer** — Sunnyvale, CA · Aug 2018 to Jul 2022

- Instrumented the mail delivery back end with **OpenTelemetry** traces across 40 endpoints, cutting time to find the slow hop in an incident from 3 hours to **25 minutes**.
- Tuned **MySQL** read replicas and Redis caching for the search front end, cutting p95 on the hottest 28 queries from 1.4 s to **310 ms**.
- Cut LCP on the Mail web inbox from 3.1 s to **1.8 s** by splitting the JavaScript bundle and preloading the first message list, measured in **Lighthouse** and real-user data.