

Aditya Sule

Building scalable systems and driving impact through technology and execution

Senior Software Engineer with a strong background in distributed systems, large scale infrastructure and high performance data processing. Proven track record of designing and delivering scalable solutions within global organisations such as Uber, Plaid and Sumo Logic. Combines deep technical expertise in backend development including Go, Java, Scala and Python with strong organisational skills, enabling complex migrations and cross team initiatives at scale.

Experienced in delivering end to end solutions from system design and implementation to rollout and adoption, while aligning multiple stakeholders in high pressure environments. Recently focused on applying GenAI to enhance engineering workflows, improve automation and accelerate decision making. Known for a pragmatic and results driven approach with a strong focus on performance, reliability and business impact.



Contact

06-829944077
me@adityasule.com
Amsterdam, Netherlands
www.linkedin.com/in/aditya-sule

Education

Bachelor of Science in Computer Engineering
University of Illinois at Urbana Champaign
2017

Skills

Programming Languages

C/C++
Python
Scala
Java
Typescript
Go

Technologies

Cassandra
Kafka
JVM
Docker
AWS
ElasticSearch
MySQL
Gradle

Languages

English
Native

Work Experience

Senior Software Engineer - Uber, Amsterdam, Netherlands | 2023 – Present

- Lead the identification and migration of clients from TChannel across large-scale monorepos (Go and Java) by implementing a system based on static analysis and heuristic methods.
- Automate issue creation and tracking, improving visibility into migration progress and remaining work across teams.
- Enable company-wide migration at scale, allowing contractors to ramp up quickly and successfully migrate 80% of previously unowned clients.
- Drive the evaluation and selection of an edge protection vendor during large-scale DDoS incidents, collaborating with internal teams and security experts.
- Develop a custom load testing tool, reducing dependency on third-party providers and significantly lowering costs.
- Facilitate collaboration between distributed teams under high-pressure conditions, resolving conflicts and improving decision-making efficiency.
- Accelerate load testing cycles, reducing turnaround time and saving substantial engineering effort.

Software Engineer – Plaid | 2021 – 2023

- Contribute as a core member of the international organisation at Plaid, driving the development of new products from POC to market for the European business.
- Develop a payments product that enables customers to accept instant payments via the Plaid API using Go.
- Integrate multiple Open Banking APIs across European regions, countries, and banks, and automate registration processes.
- Implement OAuth authentication and consent management, enabling end users to securely grant access to account data.
- Automate certificate management processes for Plaid developers, enabling secure authentication with banks using Python and TypeScript.
- Enhance internal configuration capabilities for managing customer and end-user data using Python and TypeScript.
- Achieve over 80% coverage of financial institutions across Germany, the Netherlands, Poland, France, Latvia, Lithuania, and Estonia

Distributed Systems Engineer - Sumologic, San Francisco, USA | 2017 – 2020

- Develop ingest infrastructure capable of processing 1M datapoints per second across hundreds of nodes using Kafka, Cassandra, and Elasticsearch.
- Develop and maintain a scalable distributed reactive stream processing framework used by microservices within Sumo Logic to consume data from Kafka across hundreds of nodes with exactly-once delivery guarantees.
- Design infrastructure solutions that enable hot swapping of active Kafka clusters handling production traffic, allowing maintenance without downtime.
- Optimize over-the-wire serialization, reducing CPU usage by 50%.
- Design and implement a resource-aware partitioning scheme to shard ingest streams across hundreds of nodes.
- Optimize microservices performance by benchmarking, profiling, and tuning Scala code and JVM parameters.