

Manoj Babu Katragadda

PRINCIPAL ENGINEER

Remote, Anywhere

☎ (+91) 9160808538 | ✉ manojbabukatragadda@gmail.com | 🗣️ tardunge | 🌐 manojbabuiit

Education

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Bachelors of Technology in Mining Engineering

IIT KHARAGPUR, INDIA

Jul. 2009 - May. 2014

Skills

GCP AWS DATA LAKES DATA MESH AIRFLOW PREFECT LAKEFS CUBEJS
TALARIA APACHE ICEBERG POSTGRES BIGQUERY SPARK TRINO REDSHIFT
SOLID GOLANG PYTHON JAVA RUST NODEJS LUA
DATA DEVOPS INFRA BACKEND EXPERIMENTATION OPENSOURCE
KUBERNETES DOCKER TERRAFORM *NIX ZSH TMUX NEOVIM

Contributions

- Delivered a talk at [PyCon Malaysia 2023](#), sharing insights and learning experiences on implementing Prefect-Spark-on-K8s-Operator.
- Authored and open-sourced an [Apache Spark Connector](#) for reading data from the Talaria database.
- Authored google cloud pubsub, bigquery connectors for Talaria database.
- Contributor and maintainer of the open-source timeseries database [Talaria](#).
- Authored an open-source [Prefect integration](#) to run scheduled Apache Spark jobs on Kubernetes using the Spark-on-K8s-Operator.
- Authored an open-source [Prefect integration](#) to run scheduled LakeFS data lake housekeeping jobs using the LakeFS open API.
- Authored a [Tmux status line plugin](#) to display relevant data while working with Kubernetes clusters.
- Active community member of several open-source organizations, including Apache Iceberg, Trino, Cube.js, LakeFS, and Prefect.

Experience

ZEROTOONE AI (CONSULTANT)

REMOTE, BALI

PRINCIPAL PLATFORM ENGINEER

Apr. 2025 - Apr. 2026

- Reduced spark job runtime from **60 MINS TO 4 MINS**, using storage partition join and shuffle hash hints on a 2 billion row iceberg table. [Read More for Detailed Writeup](#).
- Deployed Spark-on-Kubernetes operator with **S3 SHUFFLE PLUGIN**, enabling Spark ETL workloads to run entirely on AWS spot instances — reducing infrastructure costs by 90% across production workloads processing 10TB+ datasets. [Read More for Detailed Writeup](#).
- Built Zest, a full-stack **MLOPS PLATFORM** (FastAPI + React) that provides a declarative Python SDK like terraform for managing the entire ML lifecycle — 17 resource types from data segmentation to model deployment, with an **AST-BASED COMPILER** that generates type-safe Spark jobs, topological dependency resolution, artifact versioning, and multi-tenant workspace isolation across test/stg/prd. [Read More for Detailed Writeup](#).
- Architected **COMPOSABLE PYSPARK ETL FRAMEWORK** (lakehouse-spark-core) with hook system, plugin-based extractors/compactors, and WAP data quality gates — reduced per-pipeline code by 75% and became the execution backbone for the ML operations layer, powering feature generation, label pipelines, and model training data across all environments. [Read More for Detailed Writeup](#).
- Developed custom **AIRFLOW PROVIDERS AND OPERATORS** — KubeRay operator for Ray job orchestration with fault-tolerant reattachment, Redis provider with async triggers, ANSI-to-HTML log rendering, and hot-reload DAG bundle system from Kubernetes ConfigMaps. [Read More for Detailed Writeup](#).

- Engineered a Taxi Queue Management Microservice, leveraging **ONION ARCHITECTURE** principles and NestJS framework, to enhance taxi allocation efficiency in high-demand areas.
- Implemented this framework as a foundational model, facilitating the migration of legacy Node.js microservices to NestJS.
- Implemented a database abstraction layer isolating business entities from database models, enabling smooth migration between MongoDB and Firestore databases.
- The framework's scalability is supported by distributed Redis lock management for caching and its fully event-driven architecture, implemented through a pattern-based pubsub message controller.
- Pioneered **DATA MESH** adoption within the organization and executed its implementation, employing Apache Airflow for scheduling, Apache Spark for processing, Apache Iceberg for datalakes, Trino for querying, Cube.js for self-serve analytics, and Spotify's Backstage portal for defining our governance measures.
- **MENTORED** a staff data engineer in developing a Python-based ETL framework to streamline data modeling of raw events, inspired by Snowplow's principles and leveraging the **MEDALLION ARCHITECTURE**.
- **MENTORED** a senior software engineer in developing a backstage portal for defining various entities, including access control policies, JSON-schema definitions for raw events, A/B experiment definition and management, and JWT token issuance.
- Engaged in active collaboration and debugging with peers across software and data teams, effectively resolving their day-to-day technical blockers.

LEAD DATA ENGINEER - EXPERIMENTATION, DATA COE

Jan. 2021 - Dec. 2021

- Created a data streaming framework to collect relevant data for experiment evaluation, feeding it into our stats engine for real time analysis.
- Collaborated with product managers from various lines of business across AirAsia, who are integral stakeholders for the experimentation platform. Advocated for creating, defining experiments, and establishing evaluation criteria to interpret results effectively.
- Authored and open-sourced a Spark connector written in Java to read data from our open-source time series database, Talaria, facilitating processing with Apache Spark and writing to our **ICEBERG DATALAKE**.
- Refactored the **TALARIA** database's ingestion and flush engines to adopt consistent encoding and decoding methods, known internally as blocks and rows. By delegating data format conversions to the connector implementation, the database system achieves format agnosticism.

SENIOR DATA ENGINEER - EXPERIMENTATION, DATA COE

Sep. 2019 - Dec. 2020

- Developed a **CQRS** (Command Query Responsibility Segregation) framework using Golang to manage A/B experiment lifecycles, from defining and compiling experiment definitions into a runtime Lua script to scheduling and executing them. This work was inspired by principles from the book "Trustworthy Experiments."
- Designed and implemented an event sourcing framework using Golang that served as the foundation for our **CEP** (Complex Event Processing platform).
- Created a Pub/Sub connector for the Talaria database to stream raw events in real time, ensuring exactly-once delivery guarantee.
- Developed connectors for the Talaria database to efficiently load data into BigQuery and Google Cloud Storage.
- Developed a boilerplate code to deploy **AIRFLOW** on Kubernetes using Terraform. Over 10 airflow systems have been deployed using this module each running over 200 jobs per day. Earlier, there used to be a single node airflow - one for each team.
- Authored Terraform modules and Kubernetes manifests alongside GitLab CI templates to deploy essential infrastructure and services on Google Kubernetes Engine (GKE).

HONESTBEE

SINGAPORE/BANGALORE, INDIA

DATA ENGINEER II

Jun. 2018 - Mar. 2019

- Designed and implemented numerous ETL processes using Airflow, Spark, AWS Glue and **REDSHIFT**.
- Collaborated closely with data scientists to schedule model runs using Airflow and deploy model server APIs on Kubernetes.
- Authored an Airflow operator to run AWS Glue jobs.

ZIVAME

BANGALORE, INDIA

SENIOR SOFTWARE ENGINEER

Apr. 2016 - May. 2018

- Designed and implemented a recommendation system using Apache Spark to suggest similar products based on user preferences and behavior.
- Enhanced website performance by implementing an **NGINX** caching layer.
- Led the migration of our search functionality from Solr to Elasticsearch, ensuring high availability and scalability.

SOFTWARE ENGINEER - FULLSTACK

Aug. 2015 - Mar. 2016

- Designed and implemented the product listing page for our website with filtering, sorting, searching, and auto-completion, and developed Java APIs to power these UI features.
- Implemented centralized log collection and analytics using the **ELK** stack across all services.

KORE AI

HYDERABAD, INDIA

SOFTWARE ENGINEER

Sep. 2014 - July. 2015

- Developed APIs for the Node.js backend.