

David Barreiro Salgueiro

Senior Software Engineer | Distributed Systems · Data Platforms · Streaming

Summary

Senior Software Engineer with hands-on experience designing and building scalable platforms, distributed systems and real-time data solutions. My background combines backend engineering, data infrastructure and cloud-native platforms. I have worked with Java, Go, Kafka, Spark(Python), ClickHouse, Kubernetes, Docker and modern CI/CD environments, delivering reliable systems for demanding workloads. I enjoy simplifying complex problems, improving performance and building practical solutions that scale. Core areas include backend engineering with Go and Java, distributed systems, Kafka-based streaming architectures, cloud-native platforms, APIs, microservices and DevOps automation.

Experience

- Senior Software Engineer 04/2022 - Present
- Designed and developed scalable real-time data platforms using Kafka, Spark and containerized environments.
- Creation and maintenance of development images with Docker.
 - Message pipeline with Spark for data processing using Scala and Python.
 - Development of low-code solution for data exploration with KSQL.
 - Built backend services and internal tooling in Go and Java to improve operational efficiency.
 - Implementation of Kafka Connect, Debezium, DBT & Airflow for data processing.
 - Enhanced observability and reliability through Grafana dashboards, monitoring and automated deployments.
 - Contributed to architecture decisions, scalability and maintainability of critical systems.
- Software Engineer 03/2019 - 04/2022
- Delivered backend and data engineering solutions for enterprise analytics environments.
- Developed microservices and APIs with Java / Spring Boot.
 - Implemented near real-time pipelines with Kafka Streams.
 - Built monitoring and observability systems with Prometheus.
 - Automated CI/CD and deployment workflows.
 - Improved reliability, scalability and operational efficiency.
- Software Engineer 06/2018 - 03/2019
- Built and operated a real-time data processing platform using Kafka, Spark and Elastic Stack.
- Created and maintained Docker images for reproducible deployments.
 - Developed Java and Scala services for stream processing.
 - Implemented ingestion pipelines from multiple external sources.
 - Built Kibana dashboards for monitoring and analytics.
 - Automated deployment and operational processes.
- Junior Software Engineer 12/2014 - 06/2018
- Managed and developed distributed database platforms for a multinational retail environment
- Administered medium-sized clusters (100-200 servers) supporting large-scale data processing workloads.
 - Built internal tools for orchestration and platform operations.
 - Implemented ingestion processes from DB2, AS/400 and SQL Server systems.
 - Optimized data processing workflows based on MapReduce paradigms.
 - Improved platform stability, performance and operational efficiency.
 - Collaboration in the management of a SLURM cluster (SimpleLinuxUtilityforResource Management)

About

- 📍 A Coruña ES
- ✉ david.barreiro@outlook.es
- ☎ +34 670 308 697
- 🌐 David Barreiro
- 🔗 Personal projects

Programming Languages

Advanced

Go Java Bash Python
Scala Spring Boot

Databases

Advanced

PostgreSQL ClickHouse
MariaDb Redis
MongoDB Snowflake

Data Processing

Advanced

Kafka Flink Databricks
Spark RabbitMQ DBT
Airflow

AI-Assisted Engineering

Intermediate

Agentic Systems
Context Engineering
LLM Integration
Human-in-the-Loop
Development

Infra & DevOps

Advanced

CI/CD (GitHub, Github
Actions, Jenkins, GitLab,
Bitbucket)
Docker Kubernetes
Terraform

Open Source Projects

go-auth-kit

Reusable Go authentication module designed for production-ready backends. It supports configurable authentication strategies (cookie-based sessions or JWT access + refresh tokens), role and permission checks (RBAC), OAuth login providers (GitHub / Google), password reset flows, rate limiting at auth endpoints, and security-oriented audit logs. The objective is to provide a drop-in foundation for secure auth without reimplementing the same patterns in every service.

- Go
- JWT + Cookies
- OAuth
- RBAC
- Audit Logs

Concurrent Worker Runtime

Concurrent processing runtime for asynchronous jobs in Go. It includes dynamic worker pools, exponential retries with jitter, backpressure management, dead-letter queue (DLQ), graceful shutdown, and context cancellation propagation. The focus is reliability under load and predictable behavior during failure scenarios, with clear observability hooks for monitoring throughput, retries, and queue health.

- Concurrency
- Retries
- Backpressure
- DLQ
- Graceful Shutdown

Go Microservice Suite (gRPC + REST)

Reference microservice system composed of User Service, Notification Service, and a mock Billing Service. Internal communication uses gRPC contracts with protobuf, while an external REST gateway exposes public endpoints. It includes timeout and retry policies, transport-level error handling, and service boundaries oriented to real-world distributed systems. Built to demonstrate practical service decomposition and robust inter-service communication.

- gRPC
- REST Gateway
- Protobuf

Event Analytics Pipeline

Event-driven analytics architecture covering event ingestion, data modeling, ETL transformations, metric generation, and analytical serving layers. Implemented with Go and analytical/storage combinations such as DuckDB, ClickHouse, and PostgreSQL. The goal is to provide a pragmatic blueprint for near real-time analytics with clear separation between ingestion, processing, and consumption concerns.

- Event-Driven
- ETL
- Go
- ClickHouse
- PostgreSQL

observekit

Observability demo stack for modern backend systems. It integrates OpenTelemetry instrumentation, distributed tracing, Prometheus metrics, Grafana dashboards, centralized logs, SLO-oriented views, and Superset analytics boards for exploratory analysis. The project is focused on making system behavior observable end-to-end: from request-level traces to service KPIs, alerting, and product-level analytical insights.

- OpenTelemetry
- Tracing
- Prometheus
- Grafana
- Superset
- OpenSpec

CLI Suite

Professional Go CLI toolkit designed as a multi-command platform for operational and engineering workflows. Planned commands include Data Quality CLI, API Audit CLI, Benchmark CLI, Migration CLI, and Kubernetes helper utilities. The suite emphasizes

Cloud

Intermediate

[AWS](#) [Azure](#)

[Google Cloud](#)

Observability & Analytics

Advanced

[Grafana](#) [OpenTelemetry](#)

[Prometheus](#) [Superset](#)

[Elastic \(ELK\)](#) [Power BI](#)

Languages

[Spanish](#) (*Native speaker*)

[Galician](#) (*Native speaker*)

[English](#) (*Intermediate speaker*)

robust UX, strong error handling, testability, and repeatable automation for day-to-day engineering tasks.

- [Go](#)
- [Automation](#)
- [Benchmarking](#)
- [OpenSpec](#)
- [Testing](#)

CLI Games Collection

Progressive terminal game collection used as a technical playground for architecture and algorithmic practice. The roadmap advances from grid-based games (Snake, Sokoban) to physics and action loops (Flappy, Breakout, Space Invaders), and then to more complex systems (Pac-Man style AI, Asteroids movement, platformers, Tetris). It is intentionally structured to improve game loops, input handling, collision systems, and reusable engine patterns.

- [Game Loops](#)
- [Collision Systems](#)
- [Terminal UI](#)
- [Algorithms](#)
- [Architecture](#)
- [OpenSpec](#)

Education

Santiago de Compostela University (USC)

Degree Computer Engineering

01/2008 - 01/2014

International University of La Rioja (UNIR)

Master Big Data and Visual Analytics

01/2016 - 01/2018