Roberto Segura Sala

DATA PLATFORM ENGINEER · REAL-TIME DATA · APPLIED ML

From Cuenca, Spain · Engineering since 2011
LinkedIn: robersegsal | GitHub: rsegura | rsegura.github.io/CV

I'm on the data platform team at Audiense. I look after the architecture for accessing our data, the real-time pipelines that keep it current, and the ML processes we use to enrich it. Curious by nature: most of what I know I picked up because I wanted to understand how something worked, and I still learn best by building it myself.

Professional experience

Audiense | Backend, Big Data & Real Time Engineer

Jan 2021 - present

Responsible for the data platform end to end: its architecture, a good part of its implementation, and helping product teams use it.

- Introduced Trino as the lake's query engine and Apache Iceberg as its table format; maintain the 15 TB+ lake behind audience enrichment and the 20+ streaming consumers on it. Finished the migration of enrichment from batch to real time.
- Cut Trino report queries by 28-51% at the median and p90 by 42-60%; internal network per query down 93%, CPU per query halved; failure rate from 2.2% in the worst week to under 0.2% sustained.
- Designed VEGA, the asynchronous query API product teams read the lake through, replacing per-team RDS/Mongo copies and the ETL and Spark jobs that fed them.
- Built the interest-classification pipeline (444 categories): LoRA fine-tuning, cross-encoder reranking and a model-agreement rule, served with vLLM on Kubernetes. Human-judged strict precision 36% to 52% on 300 entities / 1,150 judgements.
- Mentor the team on using AI in day-to-day engineering work.

TMC | Data Platform Engineer

Sep 2020 - Jan 2021

Future Space S.A. | Data Platform Engineer

Sep 2019 - Sep 2020

Designed and built a data platform for fraud analysis in motor insurance claims and took it to production. Scala, Kafka, Akka, Cassandra.

everis | Big Data Engineer

Feb 2018 - Sep 2019

Batch and streaming systems for clients; ML model deployment pipelines on Kafka Streams; monitoring on the ELK stack.

Earlier

2011 - 2017

Finect | Senior Developer | 2017 - Node.js/Express APIs, OAuth2, data-analysis stack planning.
playthe.net | Big Data Engineer | 2014 - 2016 - Real-time footfall counting from WiFi detection with Spark Streaming and Cassandra; IoT monitoring.
Primum Health IT | Liferay Engineer | 2013 - 2014 - Medical research data platform with biometric devices.
amaranto Consultores | Senior Developer | 2011 - 2013 · **UCLM** | Junior Developer | 2011

Selected work

THE PROBLEM, WHAT I DID, WHAT CAME OUT OF IT

Interest classification: trusting agreement instead of a score

Assign a few interests per profile from a 444-category taxonomy. The embedding-plus-threshold baseline was confidently wrong on "attractor" categories. LoRA fine-tuning with hard negatives took F1 from 0.255 to 0.390 on a human-labelled test set but did not remove the failure mode. Adding a cross-encoder reranker helped; a threshold on its score did not hold at scale (0.95 outside the bi-encoder top-5 was right 38% of the time; 0.05 inside it, 68%). What shipped: a category survives only if both models rank it top-5. Strict precision 36% to 52%, generous 58% to 71%, at the cost of fewer categories per profile and more profiles left unclassified.

Cutting Trino query time, network and failures

Profiled report queries and S3 access, rewrote the worst steps, added Iceberg table statistics for the cost-based optimizer, a worker-allocation rule for a starved step, and a retry layer. Same query mix, ~3,500 queries before and after: median time -28% to -51%, internal network -93% (five steps from 15-117 GB per query to under 0.05 GB), CPU per query halved, failures under 0.2%.

VEGA: one way in to the data instead of a copy per team

Asynchronous query platform over the lake (FastAPI, queues, traceability, retries, Redis rate limiting, data catalog and query builder). Product teams dropped their local RDS and Mongo copies and the ETL and Spark jobs that fed them; product verticals previously blocked by data volume now run on it.

Also: TikTok mention resolution to Wikidata entities with versioned Iceberg datasets; multilingual bio phrase extraction with spaCy across nine languages; interest-based influencer recommendations where the social graph is sparse.

Personal projects

- **Hermes Expense Tracker** - shared household expenses over Telegram; FastMCP server owning rules and SQLite persistence, one Hermes Agent profile per person. github.com/rsegura/hermes-expense-tracker
- **Voice agent prototype** - Deepgram STT, LLM via OpenRouter, ElevenLabs TTS over LiveKit; explicit conversation state machine, SQLite WAL for crash recovery, tested. Private.
- **Math tutor** - Spanish-speaking voice tutor where a harness validates every model proposal before deterministic code changes learning state. github.com/rsegura/math_tutor
- **AI research wiki** - Obsidian vault maintained with Hermes Agent on agent harnesses, tool use, RAG, observability and turn-taking.

Stack

Data: Apache Iceberg, Trino, Kafka, Redpanda, Kafka Streams, Spark, Athena, Parquet.
ML: Sentence Transformers, LoRA/PEFT, cross-encoders, vLLM, Qwen, Hugging Face, spaCy, LiveKit, Deepgram, ElevenLabs.
Backend & infra: Python, FastAPI, SQL, Scala, Akka HTTP, Node.js, Redis, PostgreSQL, Cassandra, MongoDB, Elasticsearch, SQLite, AWS, Kubernetes, Docker, Terraform.

Education & certifications

Master in Big Data Development | Universidad de Castilla-La Mancha | 2015 - 2016
Bachelor's Degree in Telecommunications | Universidad de Castilla-La Mancha

Generative AI and LLMs: Architecture and Data Preparation - IBM, 2026 · Data Streaming Nanodegree - Udacity, 2021 · Confluent Certified Developer for Apache Kafka - 2018 (expired 2022) · 21 further courses listed at rsegura.github.io/CV/learning.html