

BIG DATA COMMUNITY TECH TALK · 40 MIN

Plug and Play Data Analytics, *the SQL-to-insight pipeline.*

How to answer stakeholders' ad-hoc data questions with LangChain and PostgreSQL, without letting an LLM anywhere near your production database.

THE PROBLEM

It's 4 pm and a stakeholder needs a "quick data pull". On the surface it's one SQL query. In practice it's a context switch: connect, check the joins, clean the CSV.

Dashboards don't fix it. They're static, so the first new filter sends the request straight back to your inbox. Data engineers end up as human SQL translators instead of building pipelines.

FOR

Data engineers and analytics professionals

STACK

LangChain

PostgreSQL

OpenAI LLMs

Docker

WHAT THE TALK COVERS

01 The risk

Why pointing an LLM at a production PostgreSQL instance is an operational hazard: unindexed joins on large tables, and hallucinated joins on a schema it doesn't understand.

02 OLTP vs OLAP

Transactional databases are built for writes, not questions. Analytical questions belong on a copy, an analytical reflection, not on the row-store production replica.

03 Schema as context

Giving LangChain the schema (tables, columns, types from information_schema) so the SQL it writes is grounded in your real model.

04 Medallion for AI

Bronze, silver and gold layers, so the model reads clean, named, documented tables instead of guessing at raw ones.

05 Two live demos

The Northwind e-commerce database, and a copy of a real web analytics (Umami) database answering questions like "What is the top visited page?".

KEY TAKEAWAYS

- Context is king: the answer is only as good as the schema and data the model is given.
- Never let an LLM query production OLTP. Query a copy built for analytics.
- RAG isn't an AI feature; it's a retrieval optimisation problem, which is a data engineering problem.
- With the plumbing right, the work moves up a level: from writing queries to understanding the business logic.

Watch the recording

www.youtube.com/watch?v=XS0vLK9x9nc

Read the write-up

jalcocert.github.io/JAlcocerT/plug-and-play-data-analytics

PostgreSQL in Docker

github.com/JAlcocerT/Home-Lab/tree/main/postgresql

Want this in *your stack*?

A 30-minute decision call turns this pattern into a plan for your team.

consulting.jalcocertech.com