Engineer

Modeled 348 tables across 34 Postgres schemas

Modeled the maritime domain into 348 normalized tables across 34 PostgreSQL schemas — professionals, companies, ships, jobs, reviews and the rest — with SMALLINT lookups and UUID v7 keys.

2025

348

tables modelling the maritime domain

Situation. The heart of the platform is a dense maritime domain — professionals, companies, ships, jobs, reviews — and these things reference each other constantly. A professional sails on ships, works for companies, leaves reviews; a company owns ships and posts jobs. Nearly every feature is a query across that web, so how well the data is modelled decides how well most of the app performs and how sane it is to extend.

Task. That domain had to be modelled so it stayed fast and kept its integrity, and so that adding the next entity type didn't mean fighting the schema.

Action. It's laid out as normalized PostgreSQL schemas organised by domain — 348 tables across 34 of them by now. The many small, stable enumerations — statuses, types, categories — became SMALLINT lookup tables, which keeps the rows compact and the joins cheap instead of storing text codes everywhere. Entities get UUID v7 primary keys, so they're globally unique but still time-ordered in the index. One naming convention runs throughout — plural table names inside singular-named schemas — applied without exception, which as a bonus sidesteps a lot of reserved-word collisions. And the relationships are held up by real foreign keys and constraints, so integrity is the database's job, not something the application has to remember to do.

Result. What came out is a data model that's consistent and quick, and predictable to work in because the same rules hold everywhere — there aren't special cases to memorise. Adding a feature usually means extending the schema along the existing grain rather than working against it, which is the only reason 34 schemas is a structure rather than a sprawl. It's the floor the rest of the platform stands on.

Bring this experience to your project

Use "Modeled 348 tables across 34 Postgres schemas" as a starting point for a similar challenge in your organisation. The services behind this work are available for your project. The shape of a system worked out on paper first, so the parts fit and the cost is known before anyone writes code.

Explore our services Get in touch →

Look behind the result

Read the situation, decisions and result in this example. We can discuss which parts transfer to your constraints and which need another approach. Modeled 348 tables across 34 Postgres schemas — 348 tables modelling the maritime domain

Explore our services Get in touch →

Need a solid data model from day one?

Talk through your project with the engineer responsible for delivery. We agree the scope, price basis and handover in writing before work begins.

Role context: Data Engineer. It draws on Platform Architecture, Databases, PostgreSQL, SQL, Data Engineering, Data Governance and Performance Tuning. And while you are here, we are rather good at Platform & Solution Architecture, Database Design & Modeling, Data Governance & Quality and AI Data Readiness.

Get in touch Explore our services

One piece of **a longer record** — and it is all on this site.

Every entry is written the same way: the situation, the task, what we did and what changed.

See the whole record

Define the improvement you need

Describe the workflow, the existing interfaces and the users who need the change. We can define a testable delivery and a handover that lets your team maintain it.

See what it costs Get in touch →

Need a solid data model from day one?

Ask about the decisions behind “Modeled 348 tables across 34 Postgres schemas”. You will speak with the people who can explain the work and assess your own project.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://engineer.company/portfolio/modeled-348-tables-across-34-postgres-schemas/