Engineer ソフトウェア

PostgreSQL関数中心のデータ層を設計

34スキーマにまたがる1,275個のストアド関数を持つ、PostgreSQL関数中心のデータ層を設計。すべての読み書きをテーブル直接操作ではなく、データベースで実行権限を付与できる関数経由にした。

2025

1,275

すべての読み書きを守るストアド関数

状況. 業務ロジックは漏れ出しやすい。APIの一部や、ハンドラー内で実行するSQLに分散し、同じルールが少しずつ異なる2通り、3通りで書かれ、データをどう扱うシステムかを指し示せる場所がなくなる。そこからバグやセキュリティの穴が生じる。

課題. ロジックの置き場を1つにし、すべての読み書きをデータベース経由にして、APIを業務ルールを知らない薄いアダプターに保ち、全体を厳密に制限することを目指した。

行動. 関数中心のデータ層にした。スキーマはidentity、organization、review、message、notificationと他29領域、計34に分け、アプリの全操作を1,275個のPostgreSQL関数経由にし、Goからテーブルへ直接アクセスさせない。データベースでこの境界を強制した。APIのログインロールmarinerにはアプリ関数のEXECUTEとスキーマのUSAGEだけを与え、SELECTもINSERTも直接操作も許さない。一括付与でなく約4,000件の明示的な権限を用いた。関数は、ログインできない別ロールfunction_ownerの所有で、固定search_pathを持つSECURITY DEFINERとして実行する。スーパーユーザーはマイグレーションとcron専用にし、稼働アプリから遠ざけた。

結果. 監査できる1か所にロジックがまとまり、APIは良い意味で薄く単調に保たれた。境界は全員の記憶でなくPostgres自体が守る。APIが侵害されても関数の許さない操作はできない。1,275関数では規律に実際の費用があり、新項目はクエリの1行でなく、マイグレーションと関数変更を要する。この摩擦が境界を保つ代償だった。

この経験をお客様のプロジェクトに

「PostgreSQL関数中心のデータ層を設計」を、お客様の組織の似た課題を考える出発点にしてください。この実績を支えたサービスを、お客様のプロジェクトにも提供できます。 まず全体像を設計し、各部分のつながりと費用を、コードを書き始める前に明らかにします。

サービスを見る お問い合わせ →

結果の背景を確かめる

この事例の状況、判断、結果をご覧ください。どの部分がお客様の制約の下でも生かせるか、どこに別の方法が必要かをご相談いただけます。 PostgreSQL関数中心のデータ層を設計 — 1,275 すべての読み書きを守るストアド関数

サービスを見る お問い合わせ →

業務ロジックを監査できる1か所にまとめたいですか？

納品を担当するエンジニアに、プロジェクトについてご相談ください。着手前に、業務範囲、価格の算定方法、引き継ぎ内容を書面で合意します。

担当した役割：データベースエンジニア。 関連する分野： プラットフォーム設計, バックエンド開発, データベース, PostgreSQL, SQL, データガバナンス および セキュリティ. こちらのサービスも得意としています プラットフォーム・ソリューション設計, バックエンド・API開発, データベース設計・モデリング および データガバナンス・品質管理.

お問い合わせ サービスを見る

より多くの実績の **一つです** — そのすべてをこのサイトで公開しています。

どの項目も、状況、課題、実施内容、変化という同じ構成で記載しています。

すべての実績を見る

必要な改善を定める

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

業務ロジックを監査できる 1か所にまとめたいですか？

「PostgreSQL関数中心のデータ層を設計」の判断の背景についてお尋ねください。仕事内容を説明し、お客様のプロジェクトを検討できる担当者が対応します。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://software.engineer.company/ja/portfolio/designed-a-function-first-postgresql-data-layer/