

NEON

Engineering Edition

POSTGRES · SERVERLESS · DATABASE BRANCHING

01 INTRODUCTION

What is Neon?

Neon is Postgres **rebuilt for the cloud** — not just "Postgres in the cloud." It decouples compute from storage, **scales to zero** when idle, and lets you **branch a database** the same way you branch code in Git. We use it for both staging and production at Kaiten, where it powers our event-driven architecture with Debezium-based CDC.

02 CORE BENEFITS

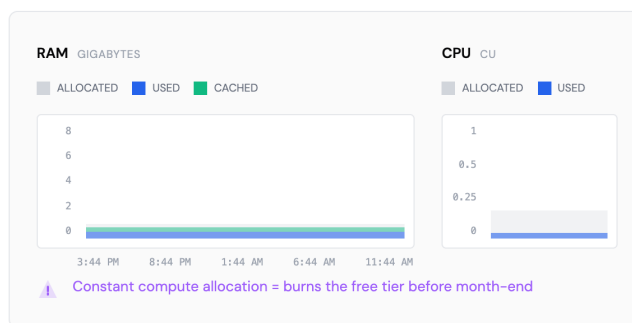
- DECOUPLED ARCHITECTURE** Compute is stateless and ephemeral. Storage is persistent in object storage. Spin up, scale down, replicate cheaply for branching — without copying disks.
- GIT-LIKE BRANCHING** Branch a database in seconds. Preview environments per PR, safe migration testing, instant staging refreshes from production. The killer feature.
- SCALE-TO-ZERO PRICING** No traffic, no compute, no cost. Generous free tier — perfect for early-stage SaaS. (With one important catch — see Section 03.)

03 TECH DEEP DIVE

The Free Tier Trap (For Event-Driven SaaS)

WHY OUR COMPUTE NEVER SCALES TO ZERO

Neon's "scale-to-zero" model assumes idle = no listeners. But event-driven architectures (Debezium, Kafka Connect, NATS bridges) keep an open connection 24/7. The compute stays alive — at 0.5% CPU, but always allocated, always billed.



The lesson: Neon's free tier is brilliant for request-response workloads. For event-driven systems with persistent listeners, expect to pay from day one — and budget accordingly.

04 MARKET LANDSCAPE

Key Competitors

PLATFORM	CHARACTERISTICS
Neon	Serverless, Branching, Open-Source
Supabase	Bundled Backend, Auth-Included, Less Branching
PlanetScale	MySQL-Native, Branching, Wrong Stack for Postgres
AWS RDS / Aurora	Enterprise Standard, No Scale-to-Zero, Expensive