

Oleksii Sholik

Lucca, Italy · [linkedin.com/in/oleksii-sholik](https://www.linkedin.com/in/oleksii-sholik) · github.com/alco · oleksii@sholik.dev

Senior software engineer with 15+ years across distributed data systems, Postgres internals, and BEAM/OTP performance engineering. Founding engineer at **Electric**; early **Elixir** core contributor.

SKILLS

Domain depth: BEAM runtime (process scheduling, memory management, OTP architecture) · Postgres internals (replication, MVCC, connection mgmt, storage) · backpressure & load shedding · OpenTelemetry & observability

Languages: Elixir, Go, TypeScript, Python, C/C++

Infrastructure: AWS (ECS Fargate/EC2, ALB, VPC, EBS), GCP, Cloudflare (Workers, DO), Terraform, Docker, Kubernetes

Systems & Linux: containers, cgroups, process & signal semantics, memory accounting, shell scripting

WORK EXPERIENCE

Founding Engineer @ Electric

Apr 2023 – present

- Core contributor to [electric-sql/electric](#), an open-source Postgres sync engine also offered as a managed, multi-region cloud service: owned the replication path (Postgres logical replication, connection management, snapshot consistency, data type correctness) and supported the service in production through observability, reliability engineering, incident response, and infrastructure work (AWS ECS, Cloudflare edge).
- Identified and mitigated the thundering-herd pattern behind production reconnect storms (50K+ errors, 15x retry amplification, a 2M-message process mailbox): implemented low-cost admission control and backpressure on overloaded internal processes, and mapped out the remaining mitigations (load-adaptive Ret ry -After, staged post-restart ramp-up, a stale-caller rendezvous protocol).
- Designed correctness safeguards between Postgres and the engine's durable logs: enumerated every state-divergence vector with its detection mechanism (point-in-time-recovery detection, LSN/xid divergence checks, idempotent writes for at-least-once redelivery); shipped restart continuity, letting clients resume across server restarts with no missed updates and no local-state loss.
- Drove observability-led memory optimization of the sync engine: used production BEAM telemetry and per-process heap attribution to locate the dominant memory costs; shipped fixes via structural sharing, process hibernation, explicit GC, and direct writes of fixed-size transaction chunks to storage, bounding memory and making data durable sooner.
- Built the OpenTelemetry tracing layer: span parent-child propagation across process boundaries, client-side trace-context propagation, export to Honeycomb, integration-test coverage.
- Shipped BEAM observability into production: `observer_cli`, `etop`, rich BEAM metrics including anomaly detection and correlation with ECS CPU/memory metrics.
- Authored the team's foundational design documents: Client authentication architecture; Definitive PG/SQLite/TS data type mapping; Reliability roadmap targeting 1M concurrent connections; Horizontal-scaling RFC (solving slot-naming conflicts, leader election via a heartbeat table, connection-pool budgeting, and a client-affinity design); Electric's minimum-privilege role model, mapping each PG interaction to its exact required privilege.
- Owned release and CI engineering for the sync engine: multi-platform Docker image publishing, per-commit canary and prerelease images built once and reused across workflows, and a CI overhaul (workflow cleanup, caching, migration to faster runners) that cut cycle times and cost.

Technologies: Elixir/OTP, PostgreSQL, SQLite, AWS, Cloudflare, TypeScript, OpenTelemetry, Honeycomb, Docker.

Tech Lead @ Packlane

Jun 2021 – Oct 2022

- Audited the company's infrastructure on my own initiative and found production services spread across more than six platforms. Contracted two ops specialists and directed the consolidation.
- Cut the monthly AWS bill by ~\$1,000 by culling unused services, right-sizing EC2, and a planned migration of the primary database to a smaller preallocated disk.
- Designed and ran the technical interview loop for Mid/Senior Backend Engineer and Product Manager roles; interviewed 20+ candidates.
- Drove the team toward a maintainable, well-tested codebase through code review, and made the company's undocumented business logic legible to other engineers.

Technologies: Elixir, Phoenix, PostgreSQL, TypeScript, React, Magento, AWS, GCP, Heroku, Datadog.

Senior Software Engineer @ Packlane

Jul 2020 – May 2021

- Designed and built a REST API integrating Magento with our backend.
- Built and maintained internal tooling for other departments, working directly with the teams that used it.
- Reverse-engineered the company's undocumented business rules by working through the code with stakeholders. Corrected mismatches between the rules as stated and as implemented, eliminated large volumes of duplicate logic, and put previously vague rules under test.
- Refactored the test suite, parallelizing hundreds of unit tests and significantly speeding up test runs.

Technologies: Elixir, Phoenix, PostgreSQL, TypeScript, React, Magento, AWS, GCP, Heroku, Datadog.

Senior Backend Engineer @ Hammer

2018 – Jul 2020

- Designed and built the core Elixir service at the center of the company's platform rewrite, handling import, processing, search, and export of dealership vehicle inventories.
- Wrote a custom indexing CSV parser to fix ingestion performance on large inventory files, bounding memory use while making subsequent per-dealer imports substantially faster.
- Built a parallelized image pipeline that fetched vehicle photos from dealer sites, stored them in GCS, and exported them to third-party marketplaces including Facebook Marketplace.
- Designed the integration layer between the new service and the legacy Ruby system, enabling incremental migration and gradually increasing reliance on the new service for all inventory-related needs.

Technologies: Elixir, Phoenix, PostgreSQL, GKE, RabbitMQ, Facebook Marketplace API.

Senior Backend Engineer @ Contractbook

2017 – 2018

- Designed and implemented JWT-based cryptographic signing via Criipto, letting users sign contracts with their national electronic ID (BankID in Sweden, NemID in Denmark).
- Specified and built the role-based access control system, writing the requirements in a form that brought stakeholders and engineers to agreement.
- Found that incorrect Service Worker caching left users running stale builds and convinced leadership to disable it.

Technologies: Elixir, Phoenix, PostgreSQL, RabbitMQ, React/Redux, JWT, Heroku, GKE.

Backend Engineer @ PSPDFKit

2014 – 2017

- Co-built Instant: multiplayer real-time syncing of PDF annotations across devices, with offline editing support.
- Built a stateful property-based test suite that surfaced concurrency bugs conventional testing had missed.
- Wrote [Testing HTTP APIs in Elixir](#), picked up by Elixir Radar and Elixir Weekly, the top Elixir newsletters.

Technologies: Elixir, PostgreSQL, QuviQ, Docker.

iOS Developer

2008 – 2014

- Built games and their supporting developer tools from scratch at a small studio.
- Delivered features and bug fixes across client apps and games as a freelancer.

Technologies: Objective-C, C, C++, Python, OpenGL ES, Cocos2D, Qt.

EDUCATION

Bachelor in Applied Mathematics, National Technical University of Ukraine "Kyiv Polytechnic Institute" · 2006 – 2010

PROJECTS & OPEN SOURCE

- **Porcelain** - Elixir library for starting and communicating with external OS processes; 2.3M+ downloads.
- **Hashids** - Elixir library for obfuscating numerical identifiers as short strings; 2.8M+ downloads.
- **Benchfella** - the first micro-benchmarking tool for Elixir; 1.5M+ downloads.
- Contributed to the **Elixir programming language** (2012–2016); credited on elixir-lang.org:
 - Developed UX improvements for IEx (colors, history, `.iex.exs`) and built the original test suite for IEx helpers.
 - Implemented core functions in the Enum, List, OptionParser, String, and System modules.
 - Built out a major part of the `mix archive.*` and `mix escript.*` tasks.
- Technical reviewer for books on Elixir: **Programming Elixir** (*Pragmatic Bookshelf*), **Introducing Elixir** (*O'Reilly*), **Elixir in Action** (*Manning*), **Elixir Cookbook** (*Packt*).