CHRIS BAILEY

cv@cbailey.co.uk

PREFACE

I am currently working as a developer and consultant for *Erlang Solutions*, having experience primarily around helping startups write MVPs or scale to different markets. I've also done PoC work in fintech.

My core experience and skills revolve around developing concurrent and distributed systems atop the *BEAM Virtual Machine* primarily using **Erlang** or **Elixir**. I also have experience working in a flexible way across the entire tech stack and I am comfortable working with *JavaScript*, *Java*, *PHP*, *Lua* as well as having some *DevOps* experience with *Kubernetes*, *Docker* and *AWS*.

I keep an informal (albeit technical) blog at `https://cbailey.co.uk` and you can find me on GitHub as `@vereis` or `@cbaileyesl`.

PROFESSIONAL EXPERIENCE

Erlang Solutions—Erlang, Elixir Engineer & Consultant (August 2018–Present)

Consulting and software engineering on several different projects for a range of clients. Notable projects listed below:

Developing Fleet/Lease management systems for a multinational car manufacturer

- ⋆ Delivered an MVP lease management portal for the Polish market (and groundwork for pan-EU expansion) working primarily with Phoenix, implementing a set of RESTful APIs for a web portal, set of mobile apps, and integration points for third party hooks.
- ⋆ Fullstack development contributing to feature development and implementing tests for the Ember.js SPA which consumed said lease management APIs.
- ⋆ Developed PoC "*key as a service*" app using Web Bluetooth APIs and Svelte for client hackathon.
- ⋆ Built an extensible pub-sub system hooking into Ecto and allowing for resilient plug-and-play integrations with third party APIs.
- ⋆ Helped integrate tooling to move from line-based logging to structured logging for increased visibility.
- ⋆ Spiked an MVP "*AirBnB for cars*" in server-side rendered Phoenix as a pitch to expand existing product offerings.
- ⋆ Assisted in refactoring and implementing a more idiomatic *"let it crash"* error handling mechanism for several Phoenix APIs as a series of middlewares.

B2B2C carsharing platform market for a pan-EU mobility startup

- ⋆ Delivered a pair of GraphQL APIs exposed via Phoenix using Absinthe.
- ⋆ Implemented a typed GraphQL error response library as an Absinthe middleware due to the lack of general consensus on how best to do error handling in the GraphQL community at the time.
- ⋆ Developed an ABAC microservice and graph database abstraction atop Postgres for flexible entity management which was extensible via a custom Lisp-like DSL.
- ⋆ Improved the project by increasing test coverage from 0% to near–100% over the course of a staggered re-write of the product to better align with idiomatic Elixir standards.
- ⋆ Began migrating set of tightly coupled distributed monoliths with a lot of duplicated code for common functionality to a microservice architecture for better code-reuse and maintainability.
- ⋆ Contributed and helped maintain microservices running on Kubernetes, including groundwork for implementing payment gateway & email microservices.
- ⋆ Developed PoC wire protocol replacement, moving away from the existing HTTP implemention and towards standard Erlang distribution using `libcluster`, effectively removing longterm intermittent failures in service–service communication and improving throughput by 20%.

PoC project increasing transaction throughput for a UK bank

- ⋆ Implemented PoC based on a research paper on the "Demarcation Protocol" to improve banking transaction throughput by two orders of magnitude (on commodity hardware); implementation in Erlang utilising standard distribution with autoscaling scripts built atop Docker Swarm.
- ⋆ Real time graphing and monitoring of transactions, failures, visibility of node crashes and dynamic scaling to replace dead nodes with Wombat.

ionCube—Intern Fullstack Engineer (July 2016–July 2017)

I was responsible for working on a greenfield MVP SPA at ionCube for my industrial placement during my university degree. I did both frontend and backend engineering in *AngularJS* and *Phalcon PHP* respectively. Other technologies utilised include *LXC*, *C*, *C++*.

- ⋆ Implement major features for the customer onboarding flow including pricing model implementation, third party sign on and authentication flows on both frontend and backend.
- ⋆ Feature development work including building a distributed *C++* application on the backend to monitor our customer's infrastructure via DNS/HTTP/SMTP/FTP which was exposed on the frontend via *D3.js* and canvas.

EDUCATION & CERTIFICATIONS

- **Erlang Solutions** – OTP Certification *(April 2020)*
- **Erlang Solutions** – Erlang Certification *(July 2018)*
- **University of Kent, Canterbury** – BSc Computer Science, *1st Class Honours*, *(Sept 2014–July 2018)*