Kevin Ferron

Bloomfield Hills, Michigan | kcf@kevinferron.com
linkedin.com/in/kferron | github.com/overfocus | kevinferron.com

SUMMARY

Software engineer who has been building software for more than 30 years, across embedded firmware, backend services, games, and web and mobile products. Most recently wrote charge-controller firmware in C++ and Rust for DC fast chargers deployed worldwide. Former Director of Engineering at Zynga who returned to hands-on work by choice. Strongest where a codebase needs structure: test infrastructure, fault handling, and shared frameworks that other engineers build on.

EXPERIENCE

EcoG
Detroit, Michigan

Senior Developer, R&D
2025 – 2026

EcoG builds the controllers and operating system inside DC fast chargers from more than 40 manufacturers.

- Proposed and built EcoG's integration test framework on my own initiative: a Rust interpreter that runs JSON-defined charging scenarios against hooks in the C++ controller firmware. Scenarios can stand in for the EV and power modules, inject faults and V2G messages, change configuration and manipulate timers. Hooks are compiled out of production builds.
- Framework became a required gate in CI and was adopted by the Systems, Stack, QA and Customer Success teams, growing to about 110 scenarios covering success and fault paths across all supported charging protocols. Replaced manual regression passes that took days to weeks, and made a reproducing scenario the standard proof that a bug was fixed.
- Worked across the DIN 70121, ISO 15118-2 and ISO 15118-20 implementations. Iterated on and improved the DC charge loop; implemented bidirectional power transfer, the scheduled and dynamic control modes of ISO 15118-20, and TLS and Plug & Charge certificate handling. Debugged interoperability against production vehicles throughout, from SLAC link setup to EXI message handling.
- Designed a global fault-handling system for the controller firmware, replacing disorganized error handling that had caused crashes and uncontrolled behavior in the field. Faults are cleared by predicate-based recovery rules, so a charger returns to service without a technician reset.
- Designed a central limits system that resolves the applicable voltage and current limits from configuration, OCPP-driven changes, converter capabilities and the EV, replacing calculations duplicated across the codebase with a single source every subsystem queries.
- Built a framework that maps controller state to MQTT topics to drive a customer's MQTT-controlled power converter, with mappings declared in an external CSV and the same converter logic running over MQTT, Modbus or CAN. Shipped in the customer's charger.
- As sole engineer, assessed feasibility, designed and built a system for running a third party's EV conformance test suite against real EcoG hardware without modifying the tests. Integrated the open-source EVerest stack, running in Docker, with EcoG modules that control the firmware and power electronics over MQTT. Delivered in three months; now used by major car makers.
- Built and maintained the Claude Code skills and agent loop workflow adopted by the team.

Kevin Ferron Tech Consultancy & Digital Agency
Bloomfield Hills, Michigan

Founder and Principal Engineer
2021 – present

- Design, build and operate Next.js sites for client businesses, including a 41-route local-search site with an estimate funnel for a tree-care company, two author sites and a video production company's site.
- Built KruX42, a universal music-link product that opens a track on whichever streaming service the listener uses, with accounts, embeds, playlists and a keyed public API.
- Built smart contracts and mobile apps for several NFT projects, using Solidity on EVM chains, Rust on Solana and React Native.
- Engineering consultant and advisor to agencies and other companies.

Independent Software Consultant
2016 – 2019

- Contract Unity engineering for clients including Imbellus and Age of Learning, and several digital agencies, delivering gameplay and product features.

Zynga San Francisco, California

Director of Engineering 2014 – 2016

- Directed a 12-person engineering organization of individual contributors and managers across Words With Friends (web), Words On Tour, Trivia With Friends and several unannounced titles.
- Led engineering for the web version of the relaunched Words With Friends, a JavaScript client built to feature parity with the mobile game.
- Launched Words On Tour on a custom in-house Unity level editor, which let designers, advertising partners and the product team build content directly.
- Put much of my own effort, and the projects I proposed, into engineering efficiency: tooling, build pipelines and asset management across the titles.
- Stayed hands-on as a director: wrote C++ and Unity code for VR prototypes, including VR poker and a VR game show, on Oculus Rift, Gear VR and HTC Vive.

Booyah San Francisco, California

Studio Technical Director, then CTO 2012 – 2014

- Hired seven engineers and led client engineering across the studio's mobile game portfolio.
- Shipped Pet Town and Early Bird 2 on iOS and Android.
- Extended the Rake build pipeline with dynamic hooks in the core game code, so product managers could schedule merchandising changes themselves.

Electronic Arts Redwood City, California and Montreal, Quebec

Tools Engineer, then Tech Lead 2008 – 2012

- On Littlest Pet Shop Online, an MMO that shipped, integrated EA's backend services for billing, subscriptions, customer support and identity into the game through web services.
- Relocated by EA to Montreal to build out a new mobile studio bringing Playfish's Facebook games to mobile, leading 14 engineers. Shipped Restaurant City Mobile and Pet Society.
- Built a Unity framework that reused Playfish's existing Flash art assets, and backend services on Cassandra, Thrift and ZooKeeper to support the mobile titles.

EARLIER EXPERIENCE

Granicus Software Engineer 2007 – 2008

WiLife (acquired by Logitech) Software Engineer 2006 – 2007

American Greetings Senior Web and Mobile Application Lead 2004 – 2006

Treeflow Technologies System Architect 2001 – 2004

Sterling Computer Consultants Systems Developer 1999 – 2001

SKILLS

Languages: C++, Rust, TypeScript, JavaScript, Python, C#, Swift, Objective-C, Solidity

EV charging: ISO 15118-2 and -20, DIN 70121, OCPP 1.6-J and 2.0.1, CHAdeMO, NACS, MCS, Plug & Charge, V2G, SLAC, EVerest

Embedded and networking: Linux, Linux networking, MQTT, Modbus, CAN, TLS, Docker, IoT and cloud release delivery

Cloud: AWS (EC2, S3, Lambda, DynamoDB, Elastic Beanstalk, IAM), Vercel

Web, mobile and games: React, Next.js, Node.js, PostgreSQL, React Native, Unity

AI-assisted engineering: Claude Code, agent skills, loop engineering