

ANDREW BECKWITH

Senior Staff Engineer | Technical Lead | Application & Platform Engineering, Distributed Systems & AI
Denver, CO | andy@elevationcraft.ai | (612) 849-7369 | linkedin.com/in/andy-beckwith-10a5052 | elevationcraft.ai

SUMMARY

Senior Staff backend engineer, 24+ years, hands-on at every level and leading from the code. Ten years on the email channel behind Zendesk Support (90,000+ customers, 60M+ emails a day): built the Email Storage Service in Go and the email-parsing library, led Email Status from RFC to shipped product, and wrote most of the agentic workflows and guardrails the email team runs its SDLC on. Earlier: Best Buy's developer API platform; nearly seven years at Wells Fargo. Volunteer structural and wildland firefighter and EMT in Colorado.

CORE SKILLS

Backend & Distributed Systems: Distributed Systems • Backend Development • Event-Driven Architecture • Kafka • Sidekiq • Resque • REST APIs • GraphQL • SQL (MySQL, PostgreSQL) • Data Modeling • Scalability

Reliability & Delivery: SLOs & SLIs • Circuit Breakers & Retries • Canary Deploys & Gradual Rollouts • Feature Flags & A/B Testing • On-Call & Incident Response • Runbooks & Documentation • Observability • APM • Datadog • PagerDuty • Sentry • Kibana • Automated Testing • CI/CD • Kubernetes • AWS

Languages & Frameworks: Go • Ruby • Rails • Java • Spring MVC • Python • TypeScript • JavaScript • Node.js • React • Next.js

AI & Agents: Agentic Systems & LLM Orchestration • MCP • Guardrails • Coding Agents (Claude Code, Codex, GitHub Copilot, Cursor) • AWS Bedrock • Tool Calling

Security & Compliance: OAuth 2.0 / OIDC / SAML • SOC 2, GDPR, HIPAA, PCI, PII

Leadership: Hands-on Technical Lead • RFCs & ADRs • Design Reviews • Code and PR Review • Cross-team Influence & Delivery • Mentorship

EXPERIENCE

Elevation Craft Technologies (independent product work) | Denver, Colorado

Founder & Principal Engineer

Apr 2026 - Present

Stack: TypeScript, React, Next.js, Vercel, Neon Postgres, Anthropic, Zod, Resend, Google Workspace, Donorbox

- Built localsocialpost.co, a multi-agent social publishing product in limited production use: specialist agents weigh local factors, weather, season, and brand voice, with cheaper, faster models where they suffice. A content orchestrator routes through six specialists with Zod-validated I/O contracts and recommends skipping the post when timing is wrong. Next.js on Vercel and Neon Postgres, Anthropic models for the agents, magic-link plus OTP auth.
- Built ecfesa.org pro bono for the fire department's 501(c)(3) association: calendar sync with Google Workspace, a minimal CMS, mailing-list signups, and Donorbox donations, for \$9 a year on free tiers. Accessibility menu with light, dark, and system themes and a red-green colorblind-safe palette; WCAG 2.2 AA on every component; the anonymous-write path hardened behind four gates.
- Trialing the Vercel AI SDK and the Anthropic Agents SDK.

Zendesk | Denver Metropolitan Area

Senior Staff Engineer, Technical Lead

Mar 2023 - Mar 2026

Stack: Ruby, Rails, TypeScript, React, Datadog, PagerDuty, MCP, LLM orchestration, AWS Bedrock, SageMaker, LibreChat, Claude Code, Codex, GitHub Copilot, Cursor, Go, gRPC, HashiCorp Vault, Kafka, GraphQL, StatsD, Vector, Grafana Loki, Java, Spring MVC, Kubernetes, Spinnaker, AWS, APM, Redis, Memcached

- Technical lead for the email channel behind Zendesk Support, 90,000+ customers: ingestion, classification, routing, and ticket creation for 15M inbound and 45-55M outbound emails a day.
- Designed the Email Status data model that surfaces Kafka deliverability events in the Support agent UI and worked in Zendesk's federated GraphQL layer, a Java Spring MVC service, to deliver it; led the project from the RFC and ADR through cross-team coordination to the shipped product (2024-2025).
- Built the Email Storage Service alone in Go, the email team's first Go service: it stores every inbound and outbound email, keyed by account ID and ULID for idempotent writes, behind circuit breakers with retries,

with its own SLOs and SLIs. One of the first services on our new Kubernetes and Spinnaker deploy patterns, which helped shape what became CICD-toolkit.

- Contributed major improvements to the shared encryption libraries behind Zendesk's bring-your-own-key encryption, where enterprise customers hold the key, and to its gRPC encryption service. Also worked on gRPC event pipelines.
- Helped migrate team secrets from HashiCorp Vault to a newer secret service, wired into Kubernetes through a secret sidecar, with rotation and short-lived secrets.
- Built sender authentication and authorization into the email pipeline with the team: SPF and DKIM to confirm who sent a message, then whether the sender is an end user in the conversation or an agent allowed to update the ticket, and which ticket fields they may change. Also led inbound-email-to-identity resolution and forwarding authentication and verification.
- Kept customer data apart at the application level: queries scoped by account and user, with cross-account checks throughout the email pipeline, on Zendesk's pod-based multi-tenant platform, placed for load and data locality, with trial accounts in their own pod.
- Built and maintained the email channel's REST APIs, including View Original email, Email Status, and new-user verification by email link, with pagination and caching.
- Applied service rate limits and per-user, per-sender rate limiting (Memcached backed by Redis), timeouts, retries with backoff, idempotent handlers, and fallbacks across the inbound and outbound pipelines.
- Made Zendesk replies fold like native email threads in Gmail and other clients by quoting the stored original instead of resending the whole thread each time. Conceived and wrote the PCI fallback: emails that may carry card data are never stored, and replies quote the last three comments with card numbers scrubbed.
- Implemented many of the email pipeline's SLOs and SLIs and set the first throughput, latency, and error-rate targets with team leadership; changes shipped through staging, pre-prod, and canary, then pod by pod, lowest-risk pods first and the largest customers last; contributed to the org-wide SLO standards.
- Helped lead the team's observability work, SLOs and SLIs, APM tracing, API circuit breakers, and flag-gated monitored rollouts: self-inflicted incidents grew rarer, smaller, and faster to recover.
- Contributed to logging improvements across the pipeline: cut log cost while keeping logs useful, added PII scrubbing, and helped bring in in-house Grafana Loki for our largest, most expensive indices. Day to day: Datadog logs, StatsD metrics, APM tracing, and Vector for aggregation.
- Performance work on processing time and outliers, and right-sizing the resources behind each service.
- Contributed across the full stack: backend services, APIs, and frontend integration points.
- Delivered Email Data Masking, the email channel's part of a top-40 Support-wide project letting enterprise customers hide personal identifiers from restricted and custom agent roles: full original scope in 10 business days against a 4-month estimate, two paired engineers, zero bugs in testing, zero post-launch incidents.
- Cut rollout issue detection from hours to minutes with a cross-tool agentic delivery system (implementation, testing, flag-gated rollout monitoring) on Datadog and PagerDuty; contributed its deployment and feature-flag rollout skills to the support org's shared skills marketplace.
- Built agentic JIRA story-implementation workflows with custom team-persona guardrails, adopted by four engineering teams; cut story-review time 60-70%, measured from live JIRA sprint data. Chose a custom JIRA CLI over MCP: faster, able to do what MCP could not yet, and lighter on tokens.
- Partnered with product in customer discovery calls, including the out-of-office auto-reply decision: ingesting automated replies would have added 30% inbound volume, a capacity and customer-trust tradeoff.
- Led architectural direction, delivery strategy, and reliability posture for customer-interaction infrastructure; led design reviews, execution planning, and incident response across multiple teams, including upstream and downstream incidents that landed on us.
- On-call every 3-4 weeks plus backup for teammates; runbooks for every service and incident shape; a postmortem, detailed RCA, and incident report on every incident; weekly and project retros.
- Mentored senior engineers and emerging tech leads on decomposition, tradeoff analysis, and execution discipline.
- Early adopter, with a partner team, of gen-AI models and coding agents; wrote most of the workflows, shared skills, and guardrails the email team runs its SDLC on.
- Helped evaluate new frontier models on AWS Bedrock.
- Pulled into 4-5+ large customer escalations a year as the go-to technical lead: designed and shipped solutions on tight timelines, communicating directly with all levels of leadership and legal.

Staff Engineer, Technical Lead

Jul 2018 - Mar 2023

Stack: Ruby, Rails, React, Redux, Kubernetes, Spinnaker, Jsonnet, vendir, Kafka, Resque, Sidekiq, MySQL, zendesk/adrian, Arturo, OAuth 2.0, Gmail API, S3, Camo, Datadog, Sentry, Redis, Memcached

- Designed an email-parsing library that removed robotic quote delimiters for 90%+ of users across 125,000+ connected mailboxes, validated by A/B testing in production behind zendesk/arturo (open-source) feature flags with stats collection: default-on for new customers, opt-in for existing, making ticket emails read far more human.
- Took over as tech lead of the move from provisioned, non-scaling infrastructure to Kubernetes, and wrote the team's Kubernetes manifest generation and Spinnaker configs, then YAML and Jsonnet (libsonnet) templating as config as code: patterns that later fed into CICD-toolkit, the self-service deploy standard a foundation team built.
- Used vendir to share org-wide config (Sentry, Datadog, pod-level settings) across services and to seed new services from boilerplate, part of CICD-toolkit's self-service path.
- Made Zendesk's OAuth 2.0 Gmail connector a reliable part of the email pipeline with the team, working through its authorization flows, token refresh, and mailbox scope handling against Google's APIs.
- Led the cross-team effort that retired a bearer-style reply token with an unbounded lifetime from outbound ticket emails: removed body token lookup behind an announced deprecation and staged rollout, then put the encoding secret on a current, upcoming, deprecated key rotation, sequenced so separately deployed minting and consuming services never broke a legitimate customer reply.
- Led backend architecture for ticket ingestion and notification: queues and workers across zendesk/adrian (open-source; inbound mail), Kafka (deliverability events), Resque (durable outbound, file cleanup), and Sidekiq (Gmail sync), idempotent so retries never produced duplicate tickets, comments, or notifications.
- Ran inbound and outbound mail processing under heavy concurrency, with timeouts, retries with backoff, and API circuit breakers at every stage; ETag caching for Camo-proxied images and S3-served attachments and raw emails.
- Coordinated multi-team migrations and performance work across distributed pipelines.
- Partnered with product and frontend teams on new customer-interaction features.
- Carried out large MySQL migrations and database optimization at email scale with the team: query and index analysis, index design, and data correctness under retried, idempotent writes.
- Worked to SLAs and service levels with the team, alongside our SLOs.
- Shaped engineering standards and architectural strategy beyond my own team.

Senior Software Engineer

Oct 2015 - Jul 2018

Stack: Ruby, Rails, Ember, React, Redux, Artifactory, LocalStack, Redis, Memcached

- One of the first four engineers dedicated to email, Zendesk's first channel and by then its roughest: modernized the pipeline with the team; a leader in the HTML email work, especially agent-forwarded emails.
- Senior contributor on the inbound and outbound email infrastructure, including webhook-delivered email triggers, supported through all three Zendesk roles.
- Drove team onboarding automation (laptop setup, dependencies, tooling), default developer tooling, runbooks, and documentation: a working environment in about half a day. Contributor to ZDI, Zendesk's Docker development environment: images, Artifactory, and LocalStack (S3).
- Built full-stack across Rails services, Ember, and React/Redux.
- Solved parsing problems in deeply nested HTML and inconsistent client email formats.

Origin Meals | Minneapolis-St. Paul, Minnesota

Product Designer & Engineer (part-time, concurrent)

Jun 2015 - Jun 2018

Stack: Ruby, Meteor, Firebase, Shopify, Liquid, Stripe, JavaScript

- Brought in by the owner to wrangle the order-management and recurring-subscription deficiencies of the original Shopify implementation and its subscription app, which had no way to express pickup location, delivery day, weekly order deadline, or recurrence.
- Planned and ran a live subscriber-base migration across recurring-billing platforms: wrote the migration and reconciliation scripts and sequenced the cutover around the weekly billing run.
- Redesigned and rebuilt the storefront UI so customers set pickup location, delivery day, weekly deadline, and recurrence in the cart before checkout.
- Designed and built the back-of-house ERP with the kitchen staff: Ruby services behind a Meteor and Firebase front end covering order ingest, an exception log for malformed rows, cook lists split across prep

and ship days, production reporting, and pack-line labels. Handed it to the incoming firm on relocating to Colorado, which continued using it; Origin Meals was later acquired by another meal-delivery provider.

Best Buy (contract) | Richfield, Minnesota

Web Application Developer, Core UI and Checkout

Feb 2015 - Oct 2015

Stack: Node.js, JavaScript

- Built Node.js applications for server-side rendering and client experiences in high-traffic ecommerce checkout flows handling customer PII under PCI-bound constraints.
- On the team that built PLSDK, a presentation-layer SDK for testing UI changes on a local machine against staging services with per-developer isolation.

Architect, Developer APIs and Platform

Feb 2014 - Feb 2015

Stack: Java, Ruby

- Architected and built the external Developer API ecosystem and portal (products, stores, recommendations APIs): API design, partner service patterns, onboarding, and standards, with access issued by API key and rate-limited with a leaky bucket.

Ruby Developer, Integration Engineering

Oct 2012 - Feb 2014

Stack: Ruby, Rails, Postgres

- Built a Rails/Postgres marketplace automation platform with vendor data pipelines integrating Best Buy product APIs with eBay listing systems.
- Automated open-box and clearance listing, pricing, and inventory sync from Best Buy stores to eBay, turning a manual side channel into a continuous one for inventory that could not sell at full price in-store.
- Implemented batch and event-driven pipelines for marketplace operations.
- Helped launch eBay's in-store pickup option for large retail sellers.

Wells Fargo | Minneapolis, Minnesota

Senior Web Developer

Jul 2009 - Oct 2012

Stack: Java, Spring MVC, Backbone.js, Maven, Gradle, Hudson

- Named a Wells Fargo Technology Operations Group Top Performer, top 1% of the technology organization, with five-of-five ratings and multiple merit advances.
- Led the move to Gradle builds, cutting build times from 30 minutes to 10, and from CruiseControl to Hudson.
- On the team that moved legacy SOAP/JPF services to RESTful Spring MVC in regulated private banking.
- Grew from junior contributor to trusted technical mentor over five years, coaching engineers on architecture, testing, and delivery.

Web Application Developer

Jul 2007 - Jun 2009

Stack: Java, Spring MVC, Backbone.js, SOAP, REST

- Helped design and build the internal Wealth Management CRM for Private Banking, covering sales, leads, opportunities, and referrals across commercial real estate loans, private aircraft financing, and trust instruments.
- Built the Banker Loan Information System as an integrated companion product.
- Built full-stack features across Java and Spring MVC services and Backbone.js frontends consuming REST and SOAP APIs in a regulated banking environment.

Modis International | Minneapolis, Minnesota

Software Consultant, contracted to Wells Fargo

Feb 2006 - Jun 2007

Stack: Java

- Consulting engineer on Wells Fargo web applications ahead of converting to a full-time role in 2007.

TCF Bank | Minneapolis, Minnesota

Intern, Web Development

Sep 2004 - Sep 2006

Stack: ColdFusion MX

- Built an intranet CMS on ColdFusion MX.

i2m (Idea to Market) | Madison, Wisconsin

Web Designer & Developer

Jun 2002 - May 2003

Stack: ColdFusion

- Designed and built the company website for founder Bruce Winkler in ColdFusion over the summer of 2002, then kept it maintained and updated through the following school year.

University of Wisconsin, Madison | Madison, Wisconsin

Undergraduate Research Assistant

Sep 2001 - May 2003

Stack: C, Fortran

- Converted research code from Fortran to C for Bormin Huang's lossless zerotree-wavelet compression testing on three-dimensional hyperspectral sounder satellite data, work later carried into a VLSI hardware implementation; two school years, 2001-2002 and 2002-2003.

VOLUNTEER SERVICE

Elk Creek Fire Protection District | Conifer, Colorado

Lieutenant | Structural Firefighter | Wildland Firefighter | EMT

2021 - Present

Certifications: CO FF1 (Structural), FFT2 (Wildland), Haz Mat Ops, EMR (2021-2023), EMT, IV-certified (2023-present).

- Volunteer service alongside varied crews and agencies: structure fires, wildland fires, and medical calls - interior attack, defensive operations, progressive hose lays and fireline construction on initial attack, and emergency medical care across many calls. A volunteer lieutenant leads when ranking on scene and follows when scope or experience ranks first; a paramedic runs an ALS medical.

EDUCATION

St. Cloud State University | Computer Science coursework. 2004 - 2005

Transferred back to Minnesota for the TCF Bank internship; Wells Fargo contract work followed, and the full-time Wells Fargo offer came before a degree did.

University of Wisconsin, Madison | Computer Science coursework, AI concentration. 2001 - 2004

Machine learning, backpropagation neural networks, fuzzy logic, image recognition, and heuristics.

Bethel College (now Bethel University) | Computer Science coursework. 1999 - 2001

Where the coursework started, before transferring to a bigger program.