

Catherine Alvarado

Senior Software Engineer · AI Infrastructure

New York, NY

Backend and infrastructure engineer who turns ambiguous, fast-shifting product areas into production-grade platforms other teams build on. In seven years at Slack I shipped the company's first production LLM response-streaming, built the multi-turn agent infrastructure behind its production AI assistant, and re-architected the quality and evaluation data platform behind every Slack AI feature.

TECHNICAL SKILLS

AI / ML infrastructure: LLM response streaming (token/SSE), agentic multi-turn systems, retrieval-augmented generation (RAG) applications, AWS Sagemaker / Bedrock, Anthropic model clients, experimentation, A/B testing & staged rollout, evaluation and quality-metric frameworks

Languages & data systems: Hack, PHP, Python, Go, SQL; Vitess/MySQL sharding, Elasticsearch, Memcache, Cassandra, Spark, S3

Reliability & observability: Grafana, Prometheus, Logstash; capacity planning and monitoring

Frontend & APIs: React, JavaScript, GraphQL, RESTful API design, real-time event systems (RTM)

EXPERIENCE

Slack · Senior Software Engineer 2019 – Present *New York, NY / San Francisco, CA*

AI / Applied ML Infrastructure, Search & AI 2024 – Present

- **Unblocked and shipped Slack's first production LLM response-streaming** — a stalled, previously rolled-back effort I took ownership of as DRI and drove to 100% of paid workspaces, managing the full staged release. The token-streaming pattern I established became Slack's company-wide standard for AI features, **now adopted by 20+ implementations spanning 10+ product surfaces** (Recap, Catch-up, Slackbot AI, Canvas AI, and more).
- Designed and built the **multi-turn agent infrastructure that became the reusable foundation for Slack's production AI assistant** — a shared task base providing tool execution, live tool-status, agent-level response streaming, and an offline LLM evaluation harness that new agents inherited by default. Owned the evaluation pipeline end-to-end and secured the first legal approval to evaluate it against real conversation data.
- Re-architected the **AI feedback and tracing platform behind every Slack AI feature (the quality and evaluation data backbone for Slack AI)**, with Slackbot as its highest-volume surface): **eliminated recurring daily production out-of-memory failures across all 16 database shards and cut average feedback payload 77% (1.57 MB → 370 KB)** via cursor pagination, query optimization, and a storage redesign. Consolidated all writes through a single feedback store, defined the tracing-blob schema, and moved payloads and traces to S3-backed storage with a zero-downtime dual-write migration.
- Set the **product vision and information architecture for the internal feedback and trace-review tool**, designing how teams inspect each AI turn (sentiment, scope, model performance, per-tool latency and token usage, citations, and remediation). The design turned raw feedback into actionable data and made the internal tool compelling enough that product managers and engineers outside the team began contributing to it themselves: **unique contributors grew from 11 to 26 and 28 engineers across 6+ teams shipped 55 platform improvements in 10 weeks**.
- Shipped **3 commercial AI Search releases to GA** for Slack's **RAG answer product** that generates answers grounded in retrieved content spanning internal Slack objects and external data: authored each spec and drove staged rollout; the flagship keyword-routing release measurably increased engagement with generated answers (27% of treated queries expanded an answer) with no quality regression. Introduced new experiment metrics and resolved cross-team confusion by cataloging all existing metrics.
- Improved LLM-serving efficiency for **message and huddle summarization** (among Slack's first production AI features, **~230K jobs/week**): eliminated wasteful polling on the shared high-priority AI job queue and built the load-test harness for the **shared AI concurrency-control cache that now gates LLM traffic across every Slack AI feature**. Shipped to 100% with production monitoring.

Messaging 2019 – 2024

- As **sole backend owner**, led design and delivery of **restricting broadcast mentions (@here/@channel)** — the #1 requested product gap, spanning 37 enterprise customers — coordinating product, design, QA, and 5 engineers; **enabled across 25K workspaces and 73K channels**.
- Designed and evangelized a **generics-based data-backfill framework** optimized for database sharding (sole author and owner): cut per-backfill code duplication 80% and time-to-implement 75%. **Adopted across 200+ backfills by 100+ engineers spanning 65+ product areas**.
- One of a **small set of engineers authorized to review and approve all production table schema changes** on Vitess (**15+ keyspaces, 1 PB+**).
- Re-architected Slack's **mentions-processing pipeline** (the code that runs on every posted message to persist mentions and fan out notifications, **450M+ messages/day**), consolidating scattered procedural functions into typed, reusable, Vitess-sharded classes. Built the side-effect-free inspection engine behind an internal on-call debugging tool, letting engineers see exactly which mentions, notifications, and jobs a message would generate to triage notification regressions.
- Led the **backend migration of Slack's Android push notifications** from the legacy FCM API to HTTP v1, converting all ~20 push types and rolling out to **100% of Android traffic across all customer tiers** ahead of Google's hard removal of the legacy API — the only path to preserving push notifications for Slack's entire Android user base. Previously reduced iOS push volume 10% through push-token hygiene.

Qventus · Backend Software Engineer 2017 – 2019 *Mountain View, CA*

- Built real-time and predictive-analytics dashboards for hospital operations with a cross-functional team.
- Built a self-serve production-data query tool that removed engineering as a dependency, and a hospital-data anonymization + time-shifting system for QA, sales demos, and engineering tests; added query-timeout protection for database resources.

Insight Data Science · Data Engineering Fellow 2017 *Palo Alto, CA*

- Built a reliable distributed ETL pipeline on AWS (S3 → Spark → Cassandra → backend API) over global-events (GDELT) data.

PATENTS & EDUCATION

Patents (granted, U.S.): *Mobile-generated desktop reminders*, cross-device reminder delivery (US11599235B1, lead inventor); *Mentions processor*, @mention parsing and notification routing (US11916862B1)

Education: Wesleyan University, BA, Computer Science, 2012 – 2016