Dmitriy Stepanov

Principal Engineer - big data, backend, AI engineering

San Diego, CA | stepanov.dg@gmail.com | linkedin.com/in/dmitriy-stepanov

US permanent resident (green card) - no sponsorship required

Summary

14 years building data platforms - from Hadoop-era product engineering to petabyte-scale cloud migrations. Currently migrating a Fortune-50 ingestion platform to Spark-on-Kubernetes, Flink, and Iceberg, and building agentic AI tooling that automates the engineering workflow around it.

Experience

Principal Data Engineer (since 2025-07; phase 1 EPAM team lead of 4; now IC outstaffed into client ingestion team), EPAM Systems — Jan 2025 - present

Client: Fortune-50 consumer-electronics company (under NDA)

- Led the pipeline-migration workstream of a multi-PB Hadoop/HDFS -> AWS S3 platform migration: dependency-phased migration of ~220 Spark pipelines (ingestion, compaction/latest, hybrids) to Spark-on-Kubernetes in ~6 months - evaluated dependency lineage and migrated leaf-first so each phase gated on the previous one's verified success. AWS-funded program.
- Personally migrated 20-30 of the pipelines, including converting legacy custom Hadoop MapReduce jobs (Wikipedia-corpus data prep for LLM training) to Spark, alongside newer JSON-flattening pipelines.
- Designed the data-parity sign-off process for the migration: HDFS and S3 ran overlapping; both copies landed on a common platform where Spark-notebook comparison jobs proved result equivalence before each dataset was signed off and cut over.
- Dual role - EPAM team lead (4 engineers), co-ran the migration process with the delivery manager while contributing as a hands-on engineer. Framework-conformant jobs migrated fast; final months were the complex custom tail (e.g. Wikipedia LLM-training data-prep jobs).
- Ingestion-team engineer for the client's core pipeline: Cassandra -> Kafka (table archiver + direct-event producers) -> Kafka -> S3 incremental -> per-dataset compaction/latest jobs. Owns schema updates and job changes across the fleet; platform includes an in-house dataset-registry system + schema/config/metrics services.
- Drove a self-service epic for data analysts: a realtime-data enrichment wizard that generates schema-update tickets as structured JSON - machine-readable so schema changes can be AI-automated end-to-end (agent flow -> PR creation). Built the companion AI chat assistant (web) on the Anthropic API with streaming, tool calling, and guardrails.
- Migrating the Kafka-export layer from Spark to Flink: replacing ~1-hour long-running pseudo-streaming Spark jobs (Kafka -> S3 JSON, secondary job JSON -> deduped Parquet, external watchdog scheduler restarts) with Flink-native pipelines. In progress.
- Participated in the table-format/catalog evaluation (AWS Glue vs Nessie vs in-house service); Iceberg-on-S3 selected, migration underway.
- AI-assisted engineering suite around Claude Code, used daily for a near-fully-automated workflow (task gathering -> prioritization -> ticket flow): an IaC tool, then a Terraform provider, managing Claude Code instances from config with profile support (self-adjusting in realtime); an "AI artifactory" pipeline that watches team + OSS AI repos, curates changes through security/poisoning review and performance evals, and installs the vetted results; a router skill choosing best model x effort config per repeatable task; a local RAG system replacing memory files (vector search, no size limits); a ticket workflow with team creation, communication, and retrospectives.
- Stepped in as the single point of contact for the customer during a delivery-manager gap (Aug-Sep), keeping the ingestion workstream aligned and delivering while the team was without a DM; integrated into the client-side team quickly and picked up the ingestion domain.

- Onboarded and mentored engineers joining the ingestion team - the go-to person for ramping new joiners on the domain and codebase.
- As AI champion on the programme, drove and evidenced the account's engineering AI-maturity to Level 2.1 (RAG Green, above the target of 2): authored the L2 evidence artefact and the Level-3 readiness roadmap - an agentic multi-agent Spark-to-AWS migration factory scoped over the ~1,000-job backlog.
- Built the team's AI-usage metrics tooling (per-day spend, DAU/WAU, limit-hit detection) and instrumented the team - daily-active AI usage measured at 87.9%, past the Level-2 >70% bar - and designed its released-only build->review->scan->publish CI/CD pipeline, closing a laptop supply-chain security risk.
- Grew a reusable AI factory - a 108-skill shared library, 39 committed AI-context files, and 4 team plugins published to a released-only internal marketplace - and delivered an AI-SDLC enablement session demonstrating these capabilities to the team.

Team Lead (~10 people) - vAuto Oracle->AWS migration & optimization program, EPAM Systems — Nov 2023 - Sept 2024

Client: COX Automotive

- Architected and delivered an extensible Python framework for Oracle -> S3 (Parquet) migration of a multi-PB estate - client kept expanding scope, so the framework shipped with extension points well beyond the original spec; owned architecture decisions, project structure, DevOps (GitHub Actions, Terraform, CI/CD, linters) and task delegation for the ~10-person team.

Key Engineer / Infrastructure Engineer / Performance Analyst, EPAM Systems — June 2020 - Oct 2023

Client: Verizon (Thingspace IoT)

- Eliminated a chronic distributed-locking failure mode in the IoT device-scheduling system: the Zookeeper-lock-based scheduler hard-locked routinely (lock limits + timing races), failing over-the-air device updates. Redesigned it to an append-only, event-sourced model in the warehouse - locking removed entirely, OTA-update failures from this cause eliminated.
- Overhauled team CI/CD: introduced Artifactory, cut build times, automated security scans into the pipeline; served as the team's DevOps point of contact and release manager (tagging, scans, docs, deploy coordination).
- On-prem -> AWS migration of the Thingspace IoT platform; refactored the legacy self-written gateway to KrakenD (config, K8s components, middleware plugins); added Micrometer/Prometheus metrics across all team services; externalized shared libraries (metrics, logging, security, config).

Skills

Programming Languages: Java, Python, Go, Scala (Spark), SQL

Big Data: Apache Spark (batch + streaming, on K8s/EMR), Apache Kafka (+ Connect, Schema Registry, Debezium CDC), Apache Flink, Apache Iceberg, Hadoop ecosystem (HDFS, MapReduce, Hive, HBase, YARN), Cassandra, Airflow

Cloud & Infrastructure: AWS (S3, EMR, Lambda, ECS, Step Functions, EventBridge, Aurora), Kubernetes (+ Spark-on-K8s, Helm), Terraform/OpenTofu (+ Terragrunt; wrote a custom provider), Azure (ADLS, Data Factory, HDInsight), Docker

AI Engineering: Anthropic API (streaming, tool calling, guardrails), Agentic systems (Claude Code, MCP, skills, multi-agent orchestration), RAG / vector search, Forward-deployed engineering, LLM evaluation harnesses

Observability & DevOps: CI/CD (Jenkins, GitHub Actions, GitLab CI, Artifactory), Prometheus / Grafana / Micrometer, Splunk, Performance engineering (JMeter, profiling)

Spoken Languages: English, Russian

Education

Belarusian State University, Specialist, physicist-engineer in Radiophysics and Computer Technologies (Intelligent Systems dept.)

Certifications

OpenAI: Cyber Deployment Practitioner (2026-08)

Anthropic: Claude Certified Architect - Professional (2026-07-17)

Google Cloud: Certified Partner Specialist Gemini Enterprise Agent Development (2026-07-15)

Google Cloud: Certified Partner Specialist Gemini Enterprise Deployment (2026-07-13)

Anthropic: Claude Certified Architect - Foundations (2026-03-19)

Anthropic: Claude Certified Architect - Foundations Early Adopter (2026-03)