

Aleksandr Gavrilenko

Lead Solution Architect

Spain (CET) · B2B contractor, EU VAT-registered · No sponsorship required
agavrilenko2.0@gmail.com · linkedin.com/in/aleks-gavrilenko · gavr.dev

Summary

- 16 years in enterprise Java, 11 as architect and technical lead, across fintech (banking, credit risk, pricing and billing) and data platforms
- Design high-throughput, mission-critical distributed systems (up to **20k RPS**, **99.99% availability**): greenfield builds, monolith decomposition, a vendor platform being brought in-house
- Own the full engineering lifecycle (architecture, HA/DR, data and migrations, capacity, cost) and build what is missing: CI/CD, testing, release process
- Answer for delivery outcomes: provide technical leadership for teams of up to 25 engineers, oversee vendor delivery and judge decisions by user impact, adoption and cost
- Stay hands-on with PoCs and research, and adopt AI across the SDLC: spec-driven development (SDD) with coding agents and custom LLM agents for code generation

Experience

Lead Solution Architect

GITMAX IT Services, Budapest, Hungary (remote from Spain) | 09/2024 - Present

Client: Top-tier universal bank (retail, SME and corporate)

Project: Bank-wide real-time pricing and billing platform (configurable fees and charges for the bank's products); ~40 Java microservices, in-memory calculations in Rust, 3k RPS synchronous and 15k messages/s asynchronous, 99.99% availability, mission-critical

Role: Architect in the platform team, covering all 3 in-house teams (~30 developers, analysts and QA) and overseeing 2 vendor teams (~40)

Key achievements:

- Designed and drove an 18-month programme to split the platform into retail and corporate subsystems (8 PostgreSQL databases and ~50 Kafka topics migrated in 3 short windows with no DDL freeze) → **no more 1-1.5 week release delays**, defects in one stream no longer block the other
- Designed the resilience architecture now in production (one Kubernetes cluster stretched across 3 DCs replacing an active-passive pair, PostgreSQL on Patroni with a 2-of-3 synchronous quorum) → losing a data centre needs no manual switchover, database **RPO near zero and RTO in seconds**
- Reconstructed the vendor-built platform's architecture documentation (module interactions, synchronous APIs and asynchronous flows, data store and Kafka topic inventories, ADRs) → the basis for bringing development in-house
- Driving the handover of development from the vendor to in-house teams (engineering standards, acceptance of 20+ vendor releases, decisions agreed with both streams, DBA, network and security) → joint delivery today, full insourcing planned
- Sized platform capacity and budget for 2025 and 2026 across all environments (compute, PostgreSQL, in-memory data grid, ELK) → **full budget approved**
- Built the in-house release process for the vendor platform (Jenkins, Ansible, K8s, Helm) with DevSecOps gates (SAST, SCA, Trivy) → release preparation from **~2 days to 1-2 hours**
- Built an LLM-agent pipeline that generates Python and SQL to reconcile data across PostgreSQL, the in-memory data grid and Kafka → onboarding a new data entity from **2-3 weeks to a couple of days**

Stack: Java 21, Rust, PostgreSQL (Patroni), Kafka, in-memory data grid, Kubernetes, Istio, Jenkins

Independent Solution Architect

Self-employed, Montenegro | 02/2024 - 08/2024

Client: Railway operator (software is a support function, not the core business)

Project: Offline-first media delivery platform for passengers (content replicated to onboard train servers during station layovers); tens of TB of content, a sync window of a few hours

Role: The product's first architect in a 12-person team with no tech lead; part-time

Key achievements:

- Designed the migration of content distribution from rsync to object storage, with a pull agent on each train → part-level resumable transfers and manifest-verified content on every train; the team completed the rollout after my engagement
- Designed the split of two monolithic services into independently scalable ones and sequenced it into a roadmap with the team → first stage went into development
- Reconstructed and documented the platform's architecture and integrations (C4 diagrams, OpenAPI contracts between services) and introduced ADRs for key decisions → the team's first architecture documentation and decision log

Stack: Java, Kotlin, Go, PostgreSQL, Docker, object storage, Jenkins, OpenAPI

Principal Solution Architect & Technical Lead

Sberbank, Moscow, Russia | 04/2019 - 01/2024

About: Russia's largest bank (SberData, its big-data unit); a 175+ PB estate across hundreds of Hadoop/Ozone clusters

Project: Federated data-access platform (HDFS data proxy with on-the-fly column-level protection, Hive Metastore federation, access-rights sync); 112 PB/month of traffic, 100 GB/s peaks, 20k RPS, business-critical

Role: End-to-end owner of the platform's architecture and technical lead of the team (14 engineers)

Key achievements:

- Proposed the architecture for proxied access to data across the estate with on-the-fly protection, replacing cluster-to-cluster dataset copies, and took it to production → **storage cost down 3-4x**
- Implemented on-the-fly column-level encryption, tokenisation and masking of Apache Parquet data, per the caller's access rights → no changes to source systems and no separate copy per confidentiality level
- Wrote the MVP of the core service (HDFS Gateway, a fork of Apache HDFS) in 2020 → in production by early 2021, recognised with a commendation from the bank's President
- Scaled the service out to 144 JVMs on several dozen bare-metal servers (heaps up to 300 GB) with a ~600 GB Hazelcast in-memory cache → **proxy overhead held at ~15%** vs direct reads
- Ran the service active-active in 3 data centres with circuit breakers on all external dependencies and zero-downtime upgrades of every stateful node → **99.99% uptime** over two years
- Grew the service into the only way to access HDFS and Ozone data by 2024 → **3,818 MAU** (people and service accounts) and ~52,000 dataset subscriptions
- Hired the team and was its technical lead → a third grew from interns to mid-level and senior, one became a service lead

Stack: Java, Hadoop (HDFS, Hive, Spark), Ozone, Parquet, Hazelcast, Kerberos, Jenkins

Lead Java Developer

Ingram Micro Cloud (now CloudBlue), Moscow, Russia | 04/2018 - 04/2019

About: Cloud arm of a global technology distributor, now CloudBlue (PaaS for cloud marketplaces); 200+ marketplaces in 45+ countries

Project: B2B cloud-commerce platform (catalogue, ordering, provisioning, billing, accounts), a ~15-year-old Java/C++ monolith being decomposed into microservices; 160k customers, ~1,500 cloud service providers

Role: Technical lead of the 7-engineer team extracting user and account management from the monolith

Key achievements:

- Designed a two-phase extraction of the monolith's oldest and most tightly coupled subsystem (new service on the existing schema first, its own storage once proven in production) → every step reversible, with no parallel system or customer migration
- Cut over by routing part of production traffic to the new service → one of the first microservices on the new architecture, live in early 2019 with no disruption to customers
- Saw the two-phase approach adopted by other teams as the default decomposition pattern → a planned rewrite-and-migrate track was dropped, saving **hundreds of person-days** (according to the director of engineering)
- Set up an architecture review board on my own initiative and chaired it (10 tech leads covering every team; a single forum for architecture discussions across the platform)

Stack: Java, Spring, Hibernate, WildFly, PostgreSQL, C++, Kubernetes, Helm, Terraform, Azure, Jenkins

Lead Solution Architect (promoted from Senior Java Developer)

SberTech, Moscow, Russia | 06/2014 - 04/2018

About: IT subsidiary of Russia's largest bank; ~11k staff

Project: Credit risk assessment platform (automated credit decisioning for retail and corporate lending); up to 80k applications a day, business-critical

Role: Lead architect and technical lead of the platform, working through its 4 team leads (25 engineers, 4 cities); three promotions from developer in under three years

Key achievements:

- Became tech lead of a three-engineer credit-history service within months of joining and grew it into the decisioning platform for nearly all of the bank's lending products → still in use today
- Designed the migration of the 24/7 system from Oracle to PostgreSQL and rehearsed it until repeatable (GoldenGate CDC) → the team ran the cutover after I left, with **a pause of minutes**
- Moved raw payloads retained for regulators (~70% of a 20 TB Oracle database) to object storage → **hot data cut to ~6 TB**, unblocking the move off Oracle
- Automated 25 credit workflows on Activiti BPM with inline SAS RTDM scoring, replacing manual underwriting (about a week per case) → minutes for simple retail decisions, minutes to hours for complex corporate ones
- Hired the first ten engineers myself, built onboarding and developed a tech lead in each city to delegate to → the team grew **eightfold** in about a year

- Assessed and architected ~150 change initiatives (10 to 1,000 person-days) for three business units competing for capacity → most went into production

Stack: Java 8, Spring, Activiti BPM, SAS RTDM, Oracle → PostgreSQL, GoldenGate, WildFly, Jenkins

Java Developer

Neoflex, Moscow, Russia | 11/2011 - 05/2014

About: Specialist systems integrator for the financial sector; ~950 people today

Project: Credit origination platforms for banks, built on the Neoflex FrontOffice product (mortgage, car and consumer lending)

Role: Developer in project teams delivering credit origination platforms and integration (4 banks)

- Worked as a developer on NOMOS-BANK's mortgage origination platform (50+ process steps; backend, frontend, IBM BPM workflow, integration) → the platform was delivered in nine months and halved application turnaround time (according to Neoflex)
- Implemented integrations between the platforms and the banks' systems (Oracle SOA Suite, IBM WebSphere ESB) → the platforms were connected to core banking, three credit bureaus, and risk, decisioning and fraud systems

Stack: Java EE, EJB, JSF, AngularJS, Oracle, IBM BPM, Oracle SOA Suite, IBM WebSphere ESB

Junior Java Developer

Akzia.Apps, Moscow, Russia | 09/2010 - 10/2011

About: Mobile development arm of the Akzia publishing house

Project: Four niche cross-platform consumer mobile apps (including an XMPP-based messenger)

Role: Backend developer in the mobile app team (5 engineers)

- Owned the server side of all four apps (Play Framework, Hazelcast, PostgreSQL) → each app ran on a single backend for its iOS, Android and Java ME clients

Stack: Java, Play Framework, Hazelcast, PostgreSQL, XMPP

Unix System Administrator

Media Mir (RBC media group), Moscow, Russia | 03/2008 - 06/2010

About: Entertainment arm of the RBC media group (QIP, Smotri.com, LovePlanet)

Project: Server estate behind the group's consumer internet services; ~400 FreeBSD and Linux servers

Role: System administrator in the operations team (6 engineers on a 24/7 rota)

- Covered monitoring, first-line incident response and escalation on 12-hour shifts → services with millions of users had round-the-clock cover

Stack: FreeBSD, Linux, nginx, Apache

Mentoring & Technical Writing

- 500+ one-to-one mentoring sessions since 2024 with engineers from junior to lead level (system design, Java, DevOps, engineering management, career growth)
- System Design Cheat Sheet: a six-part series on HackerNoon
- In Russian: a column on the Java ecosystem (RB.RU), articles on solution architecture and on moving from monolith to microservices (Skillfactory, itspeaker), a video talk 'Java. Changed the World' (YouTube), and others

Education

Bauman Moscow State Technical University | 2005 - 2011

Specialist degree (Master's level), Computer-Aided Design Systems

Languages

Russian (native) · English (professional working proficiency) · Spanish (elementary)

Skills

Architecture: Solution Architecture, Software Architecture, System Design, Distributed Systems, High Availability, Disaster Recovery, Capacity Planning, Performance Tuning, Load Testing, Architecture Decision Records (ADR), C4 Model

Data: PostgreSQL, Patroni, Logical Replication, Change Data Capture (CDC), Debezium, Zero-Downtime Migration, Oracle Database, Hazelcast, Elasticsearch, Apache Kafka, Apache Hadoop (HDFS, Spark, Parquet, Ozone)

Engineering: Java (Java 8, Java 17, Java 21), Spring Boot, Hibernate, Microservices, Event-Driven Architecture, Domain-Driven Design (DDD), REST APIs, gRPC

Platform: Kubernetes, Istio, Helm, Docker, Jenkins, CI/CD, Ansible, Terraform, HAProxy, Prometheus, Grafana, Linux

AI in engineering: LLM Agents, AI-Assisted Engineering, Spec-Driven Development

Leadership: Technical Leadership, Team Leadership, Vendor Management, Architecture Governance, Stakeholder Management, Mentoring

Domain: FinTech, Banking Systems, Billing Systems