

Iaroslav Postovalov

Hannover, Germany | postovalovya@gmail.com | [linkedin.com/in/iaroslav-postovalov](https://www.linkedin.com/in/iaroslav-postovalov) | github.com/commandertvis | commandertvis.github.io

Compiler and JVM developer, 21 years old. Sole maintainer of a production blockchain language (~197K LOC Kotlin). On the Kotlin/JVM compiler team at JetBrains from age 17, part-time through university: IR backend, bytecode optimizations, inline class interop. Published research on JIT compilation targeting JVM, JavaScript, and WebAssembly.

Languages: native Russian, fluent English | **Legal status:** Russian citizenship, EU Blue Card (Germany)

EXPERIENCE

Software Engineer — ChromaWay

Apr 2025 – Present

Remote • Full-time (concurrent with online education)

- Sole maintainer (since Feb 2026) of Rell (github.com/ChromiaProject/rell), the primary language for the Chromia relational blockchain. Tree-walk interpreter with SQL generation, ~197K LOC Kotlin.
- New compiler IR: an immutable resolved tree decoupling parsing and typechecking from execution; replaces a runtime-coupled AST and opens a path to alternative backends (LLVM, GraalVM Truffle). FlatBuffers wire format; fuzz testing and round-trip checks on every build.
- Codebase modularization into separate compiler frontend, IR, and runtime components. Runtime snapshots for blockchain state (co-author). PostgreSQL backend: query cancellation propagation, per-process schema isolation for parallel tests, unclosed-connection tracking.
- New language features: `struct.copy()` for immutable updates with selective field overrides; `try_call_catch / try_call_result<T>` for structured error handling with automatic database rollback; `@disabled` test annotation; `@mount` for renaming entity fields without DB schema changes.
- Build and developer tooling: Maven-to-Gradle migration (incremental builds, config cache); JUnit 4-to-5 migration with parallel execution. Maintainer of the LSP server, client stub generator, and stdlib doc plugin (all merged into the Rell monorepo). IntelliJ plugin: Chromia CLI integration with auto-detection and Docker-run fallback.
- Codebase health: dependency reductions (Guava, Apache Commons, HttpClient → Kotlin/Java stdlib; Picocli → Clikt); locale-determinism fixes with a dedicated testing tool; reflection-based grammar hacks removed; Python integration tests rewritten in Kotlin. GitLab-to-GitHub auto-mirroring for 19 Chromia repositories. Parser benchmarks, end-to-end profiling tool, contributor and release-process documentation.

Software Developer — JetBrains (Kotlin Team)

Sep 2022 – Dec 2024

Yerevan, then Bremen (Germany) • Part-time (concurrent with full-time education)

- Kotlin/JVM compiler team during the K2 frontend transition. IR backend, performance work, language features.
- Memory allocation reductions in compiler frontend and IR lowering passes. Bytecode-level optimizations targeting boxing conversions in generated code.
- `@JvmExposeBoxed` annotation: seamless integration of inline/value classes with Java callers.
- Android testing infrastructure migration to the official Android Gradle Plugin framework. Compiler test renovation for the new K2 frontend, including test runs on Android.
- Bug fixes in IR serialization and lazy declaration deserialization causing production compiler crashes.

Software Developer (Intern) — MiLaboratories

Jul – Aug 2022

Remote

- Kotlin DSL with statistical and serialization features for an internal plotting library based on `lets-plot-kotlin`.

Researcher — JetBrains Research, Nuclear Physics Methods Lab

Sep 2020 – Apr 2022

Dolgoprudny (Russia), then Remote

- Group's research scope: non-accelerator particle physics, numerical simulations, and software development for experimental physics.
- Core developer of KMath (github.com/SciProgCentre/kmath, 700+ stars), a cross-platform mathematical library for Kotlin. Core design contributions to the abstract algebra API, separating operations from data structures. N-dimensional array support, automatic differentiation, integrations with math libraries (EJML, ND4J, Commons Math).
- Cross-platform expression compiler targeting JVM bytecode (ObjectWeb ASM), JavaScript, and WebAssembly from a single AST. Boxing/unboxing optimization pass with bytecode type state tracking. Paper: "Compilation of mathematical expressions in Kotlin" (arXiv:2102.07924).
- `kmath-gsl`: Kotlin/Java bindings to the GNU Scientific Library via C interop, implementing matrices and vectors. Native build/toolchain setup, CI, documentation, performance benchmarking.
- KMath-to-LaTeX renderer for computable expressions in Jupyter notebooks.

- Interactive system for studying probability distributions used in Monte Carlo methods (gitlab.com/CMDR_Tvis/monte-carlo-script). Custom DSL for defining formulas, with C++ tree interpreter and bytecode interpreter for performance research.
- Kotlin/Ktor REST API for plot generation and function evaluation. Cross-platform bindings: Python via `ctypes`, Java/Kotlin via JNI. Diploma at the 58th International Scientific Student Conference (MNSK-2020).

TECHNICAL SKILLS

Languages	Kotlin (expert), Java, C, Python, C#, SQL
Compiler engineering	Language design; lexer/parser implementation (ANTLR); AST and IR design; lowering passes; type inference and checking; semantic analysis; code generation; optimization passes (constant folding, inlining, boxing/unboxing elimination); tree-walk interpreters; multi-target backends (JVM bytecode, JavaScript, WebAssembly, SQL); end-to-end compiler testing
JVM internals	Bytecode generation (ObjectWeb ASM); invokedynamic; inline/value classes; allocation profiling; class loading; JNI/FFI; GraalVM
Performance	Microbenchmarking (JMH), profiling, performance regression analysis, test automation
Tooling & infra	Git, Gradle, GitLab CI, PostgreSQL, Docker, JUnit 5, Linux, Kotlin Multiplatform, open source maintenance
Domain	Blockchain smart contract languages, SQL codegen, scientific computing, mathematical library design, automatic differentiation, AI-assisted development

EDUCATION

BSc Applied Computer Science — *Constructor University Bremen* Sep 2023 – Aug 2027

Applied Mathematics & CS — *Lomonosov Moscow State University, Yerevan Branch* Sep 2022 – Jun 2023

PUBLICATIONS & HONORS

- **Preprint:** *Compilation of mathematical expressions in Kotlin* — arXiv, Feb 2021 ([arXiv:2102.07924](https://arxiv.org/abs/2102.07924)).
- **Paper:** *NMPUD: a computer system for sampling and examining probability distributions* — ITMM-2020 proceedings, Tomsk, 2021. Co-authored with Prof. A. Voytishchik.
- **Talk:** *Dynamic compilation of mathematical expressions with Kotlin* — SnowOne 2022 (JUGNsk), Novosibirsk.
- **Winner, National Technology Olympiad** — “Automation of Business Processes” track (1C partner case). HSE University, Moscow • Mar 2022.
- **Volunteering:** Translator at Translators without Borders (Apr 2022 – Feb 2023, 11 months).