



Vladimir Ivanov

Solving complex engineering problems and delivering projects that matter to the business

Over 10 years developing complex products, including 4 years leading a team. My background spans the 2GIS cross-platform application core, backend components and low-latency C++.

I design architecture, work with concurrent code, profile and debug complex systems, and optimize performance and memory usage.

I take projects from initially unclear requirements to release. I investigate constraints, validate ideas with prototypes and measurements, find simpler solutions and bring people together to implement them. I connect engineering decisions to business priorities and find a practical path to delivery.

Looking for a team building a great product, with ambitious goals and technical challenges.

Experience

Top 9 Placement, Low-latency Data Transfer Challenge

August 2026 • Batumi, Georgia

Placed **9th out of 100+ final submissions**. Built a low-latency market-data fan-out system in a few days. Completed the project end to end: architecture, AI-assisted implementation, infrastructure and performance validation in a distributed environment.

- Built a **C++23** pipeline using UDP unicast and **DPDK kernel bypass**.
- Optimized a lock-free shared-memory SPSC ring buffer. Cut handoff latency by **hundreds of nanoseconds** through cache-line isolation and zero-copy.
- Designed a compact, self-contained, lossless wire format optimized for ENA LLQ/Wide LLQ constraints. Added MTU-aware adaptive batching to balance latency and throughput.
- Tuned Linux for low-jitter execution: CPU pinning and isolation, IRQ affinity, tickless cores, RCU callback offloading and hugepages.
- Built a reproducible AWS EC2 benchmark harness with Terraform, Docker-based builds, automated deployment and orchestration through SSM.
- Investigated hot paths using perf, PMU counters and disassembly. Optimized **p50-p99.99 latency** across load levels and receiver counts.

C++23 • HFT • Low Latency • DPDK • UDP • Lock-Free • Shared Memory • Linux • perf • ThreadSanitizer • AWS • Terraform • Docker

Backend Engineer, =nil; Foundation

May 2024 – July 2025 • Batumi, Georgia

Developed backend components of a distributed system. Took work from architecture and prototypes to integration, deployment and validation on a live cluster. Investigated critical failures and coordinated fixes, helping the team move from a single-machine setup to a distributed deployment.

- **Extracted a standalone RPC node from the monolithic cluster deployment.** Introduced binary Protobuf transport over libp2p and a shared interface for local and remote execution. Separated transport code from application logic while preserving both distributed deployment and simple local startup. Implemented RPC-node deployment and load balancing through Ansible.
- **Reduced archive-node setup to a single bootstrap URL.** Implemented initial configuration retrieval through JSON-RPC and updated node configuration, network addressing and deployment infrastructure.
- **Organized stabilization of the distributed deployment** during the move from a single-machine setup to a cluster. Agreed readiness criteria, mapped issues and dependencies, and coordinated fixes. Personally resolved routing errors, node discovery after restarts and premature RPC cancellation.

Backend • Distributed Systems • RPC • Protobuf • libp2p • GitHub Actions • AWS • Terraform • Ansible

Contact

📍 Batumi, Georgia

✉ un1or@ya.ru

☎ +995 591 02 33 43

🔗 [Un1or](#)

Education

MS, Applied Mathematics and Computer Science

Novosibirsk State Technical University

Class of 2013

Novosibirsk, Russia

BS, Applied Mathematics and Computer Science

Novosibirsk State Technical University

Class of 2011

Novosibirsk, Russia

Skills

Professional

Communication

Teamwork

Planning

Task estimation

Decomposition

Architecture development

Cross-platform development

Debugging

Profiling

Optimization

Problem solving

Code review

Hiring

Feedback

C++ Developer, 2GIS

January 2013 – May 2024 • Novosibirsk, Russia → Batumi, Georgia

Developed the cross-platform core of 2GIS, a complex product with a multi-million audience. Worked from business logic down to low-level data access and 3D-engine integration. Delivered major architectural changes under strict backward-compatibility requirements and a regular release schedule.

Took on problems whose solutions were not yet clear. Built prototypes, tested architectural hypotheses on real data, coordinated with other teams and brought integrations to release. Initiated improvements beyond my assigned tasks: proposed a new advertising-data architecture, drove C++20 adoption and found and validated optimizations in other teams' components.

- **Designed a new advertising-data storage and delivery system:** on-device storage became **3 times smaller**, and queries ran **more than twice as fast**. Validated key architectural hypotheses through inexpensive experiments on existing data and helped the team bring the solution to release. The new system eliminated a large persistent memory footprint and update-time memory spikes.
- **Accelerated map-subsystem initialization by up to 2 times**, making styles available **200–300 ms earlier**. Analyzed the startup dependency graph and removed waits for the general configuration service and file I/O from the critical path.
- Identified repeated expensive computations in the 3D engine and validated caching. The prototype cut **time to first map display by 2.5 seconds (about 20%)** on a slow device. Handed the map team working code, measurements and an explanation of the cause. They used this foundation for the production implementation.
- **Led the mobile application's migration to the new Zenith 3D engine**, from the first prototype to release on Android and iOS. Designed transparent support for both engines with runtime switching to preserve old data-format compatibility and keep development on the main branch. Decomposed the project, maintained the roadmap, ran cross-team syncs and personally implemented complex parts. Built a C++-backed Qt Quick/QML map component with OpenGL rendering.
- **Replaced direct 3D-engine access with SDK integration.** Removed a parallel data-management path, reduced technical debt and enabled the application to reuse capabilities developed in the SDK.
- **Resolved virtual address-space exhaustion** during map-data updates. Built a gperftools-backed memory manager for Google Benchmark to measure both allocations and file mappings. Based on the measurements, replaced whole-ZIP mmap with on-demand mapping of the required ranges.
- **Drove C++20 adoption across all C++ development at the company.** Worked through Clang, GCC, MSVC and dependency constraints. After enabling the standard, actively championed its practical use through my own code and reviews, including concepts, the spaceship operator, consteval, template lambdas and more.
- Implemented a library of **non-null identifiers and optional wrappers over them without memory overhead**. Expressed domain constraints in types, making APIs stricter and identifier handling safer.
- Implemented complex user-bookmark synchronization logic **using TDD**. The tests later enabled two implementation rewrites without significant issues.
- Built packaging for the new **Ubuntu and macOS** desktop application from scratch: deb packages, DMG images, application signing and automated Jenkins builds. Resolved library layout and RPATH issues and established delivery through a deb repository.

C++03/11/14/17/20 • STL • Architecture • Concurrency • Data Storage • Qt Quick • QML • Android • iOS • Windows • Linux • macOS • Google Benchmark • gperftools • Tracy • heaptrack • TDD • CMake

Technical

C++20
Git
CMake
CLion
Tracy
Python
SQL
Linux

Personal qualities

Proactive
Creative
Passionate
Responsible
Attentive
Honest
Friendly

Languages

Russian: Native

English: Working proficiency

Hobbies

Road trips
Visiting new places
Hiking
Cycling
Fitness training

Team Lead of the Mobile Application Core Development Team, 2GIS

August 2018 – September 2022 • Novosibirsk, Russia

Led delivery of two of the company's most technically demanding features: the seamless world map and online mode. Responsible for architecture, planning and team delivery. Wrote code, took part in technical discussions and helped engineers work through difficult parts of the implementation.

Planned quarters and sprints, estimated department-level initiatives, and hired and developed engineers. Built collaboration with other teams and agreed priorities and resources for shared projects.

- **Delivered the two-year transition to a seamless world map.** Replaced separate regions with a unified map. Broke down the large-scale migration, maintained the roadmap, coordinated cross-team dependencies, advised engineers and personally implemented parts of the work.
- **Delivered online mode against a fixed advertising-campaign deadline.** When the project fell behind, revised the plan and secured help from engineers in other teams. Led pre-release stabilization. The feature reduced **application churn from 22.3% to 7.3%.**
- **Designed the protocol for communication with the online business-directory backend.** Validated the idea with a Python test backend I wrote. The team used it for client development before the main server was ready, and the protocol shipped to production.
- **Helped build the mobile SDK team from scratch.** Reviewed 500+ CVs, wrote **hundreds of candidate messages**, conducted **100+ interviews** and helped hire **10+ strong C++ engineers** within a few months. Later transferred interview materials and practices to colleagues so hiring could continue without my involvement at every stage.
- Developed team autonomy by delegating mentoring, interviews and ownership of individual areas to experienced engineers. Conducted performance reviews, provided feedback and helped people broaden their technical expertise. Colleagues took over planning and project coordination in my absence.

C++ • Technical Leadership • Architecture • People Management • Planning • Cross-team Collaboration • Hiring • Mentoring

CI Engineer, 2GIS

March 2015 – November 2016 • Novosibirsk, Russia

Built and developed CI for a large cross-platform C++ project. Reduced the complete build-and-test cycle **from approximately one hour to 10–15 minutes**, improved stability and made infrastructure reproducible.

Set the direction, hired engineers and gradually handed over CI development to them.

- Introduced compiler caching and optimized Git operations, replacing expensive repository cloning with GitLab and Bitbucket Server API calls. The time for one of the slowest Windows builds fell **from 35 to 7 minutes** with a warm clcache cache.
- Traced intermittent **MSBuild** failures to worker-process reuse and eliminated them. Builds stopped randomly failing on valid code.
- Migrated manually configured Jenkins infrastructure from Hyper-V to **OpenStack** and defined it in **Heat Templates and Ansible**. Automated environment and compiler provisioning and set up daily hot backups using burp and LVM snapshots.
- Evolved CI from post-merge master checks to mandatory PR runs and **automated merging of coordinated changes across multiple repositories**. Set up test-coverage reports. Cleaned up Clang warnings and enabled warnings-as-errors.

Jenkins • Python • Bash • Git • ccache • clcache • Ansible • OpenStack • Heat • LCOV • GCOV • Linux • Windows • CI/CD

Personal Projects

wishmap.me

October 2025 – February 2026 • Batumi, Georgia

Created a vision-board service from an idea into a live product with real users and payments.

Did the work end to end: product decisions, architecture, implementation, testing and operations in **Google Cloud**.

- Organized AI-assisted development around **SDD**: task definition, detailed plan review, implementation, code review and test-driven revisions. **Approximately 95% of the code was AI-generated**. TDD and **Playwright/pytest** E2E tests provided fast feedback.
- Built an offline development and testing environment with local storage implementations and a simulator for successful and failed payments. The application and E2E tests run without internet access, including the complete purchase flow.

Python • JavaScript • Google Cloud • SDD • Cursor • Codex • Antigravity • Playwright • pytest

Talks and Publications

Technical interview, architecture section [ru; YouTube automatic subtitles]

June 2022

C++ Russia 2022

Talk about C++ in mobile development [ru; YouTube automatic subtitles]

June 2022

C++ Russia 2022

Come on without b* (bases) [ru; YouTube automatic subtitles]

May 2022

Dump 2022

Dealing with mysterious MSBuild crashes on XamlTaskFactory [ru]

August 2016

Habr