Valery V. Vorotyntsev

Experienced software engineer, pragmatic problem solver. I can grow a happy and productive team. People like working with me.

Email: valery.vv@gmail.com
GitHub: vvv
LinkedIn: https://www.linkedin.com/in/vorotylo/

Work

October 2020–now: Rust programmer at Elastio

Elastio is a startup building a **cloud-native** backup and recovery solution for public clouds.

What I did there:

- Programmed gRPC API services in Rust (users, email delivery, AWS Lambda invocation, asset filters, *&c.*)
- Improved maintainability of the CI (applied `github-actions-dhall`)
- Implemented a config crate (à la `aws config`)
- Enhanced the ergonomics of `elastio` CLI tool
- Wrote ransomware detection tools — from early experiments and prototypes to the alpha version

Technologies:

- Rust: `nom`, `sqlx`, `tokio`, `tonic`
- AWS: EBS, EC2, ECR, Lambda, S3
- Protobuf
- Docker
- Bash, `jq`, Python

2011–2020: Distributed object store

Company: Xyratex → Seagate

The project started as an exascale **object storage system** for HPC. Then it pivoted into an archival solution. Then a hybrid cloud... The code had been open-sourced in autumn 2020.

July 2019–August 2020: "Hare" project 🐰

I lead a team of 5 engineers. We had successfully replaced legacy HA system with a simpler solution based on **Consul**.

My main contributions:

- PC3 collaboration model — for 5 months the team was a happy oasis 🌴 set amid enterprise desolation
- Introducing the practice of **RFCs**

- Tests automation, CI, merge bot
- Configuration module

Technologies: Python, **Dhall** 🔪 , Bash; Consul; GitLab CI, Jenkins; GFM-formatted English

May 2017–June 2019: Haskell programmer, team lead

- Inherited a **High Availability** (HA) solution — 50K lines of Haskell code — from Tweag.io developers. Coped with it. 😅
- **Mentored** 5 colleagues, who never programmed Haskell before. We had become a team of Haskell developers.
- Maintenance. New features development.

Technologies: **Cloud Haskell**, `Control.Monad.Operational`

June 2011–May 2017: C programmer

- Configuration caching subsystem (DAG of conf objects, client/server, graph traversal APIs, data format converter, visualization) — design and implementation
- Modular initialization/finalization mechanism — implementation
- Memory-efficient representation of device pools — design and implementation
- Wrote helper scripts for fellow developers

Technologies: C, Python, Bash, a sprinkle of **Expect** and JavaScript

January 2011–May 2011: Embedded developer at Cogent Plus

Integrated third-party TR-069 client with OpenRG middleware (Linux-based). The software ran on ITS Telecom mobile broadband router.

2006–2010: Telecom data processing

Company: UMC → MTS → Vodafone

Maintained and developed CDR processing software in C++ and Python.

1999–2006: Nuclear power plant simulators

Company: ИТЦ ПК, subcontracted by GSE Systems

- Ported the "Plant Process Computer" dashboard system (watch it in action :desktop_computer:) from IRIX to Linux, customized, and integrated
- Designed and implemented a client-server GUI application (C++)
- Wrote a CGI server for generating reports (Python)

Technologies:

- SysV IPC (shmem, sockets), X11, dbm
- C, C++, Bash, awk, Python
- gtkmm, ACE framework, Trac (issues & wiki), DataViews

🎓 Education

- 1993: Kyiv Natural Science Lyceum No. 145

  Award of recognition for outstanding grades in mathematics
- 1999: National Technical University of Ukraine "Kyiv Polytechnic Institute"

  Avionics engineer (diploma with honours)

Community

- LtU-Kyiv hackathon — organizer
- IT volunteering — lead a team of Android developers in 2015
- Haskell study group
- Elm study group
- Rust Hack & Learn, Kyiv — organizer