

## Dmytro Petrashko

École Polytechnique Fédérale de Lausanne  
Lausanne, Switzerland  
+41 787 23 77 55  
dmitry.petrashko@gmail.com

### INTERESTS

Static analysis;  
Compiler construction;  
Developer tools;  
Programming language theory and implementation;  
Parallel algorithms;  
I/O efficient algorithms.

### HIGHLIGHTS

Designed and implemented compiler middle-end and backend for Dotty compiler, the future Scala compiler, together with Martin Odersky. Designed abstractions for the compiler that substantially lowered maintenance cost as well as barriers to entry for new contributors while at the same time reducing compilation time.

Contributed 800+ commits, 160k+ lines of code to Dotty, bootstrapped the compiler and fixed 200+ issues. Introduced an extensive self-verification system into compiler that allows to discover and localize bugs easily.

### EXPERIENCE

**Doctoral Assistant**, *École Polytechnique Fédérale de Lausanne*

September 2013 - Present, Lausanne, Switzerland

Working on evolution of *Scala* (<http://www.scala-lang.org/>), an object-oriented functional programming and scripting statically typed language, designed to concisely express solutions in an elegant, type-safe and lightweight manner.

*Dotty* (<http://dotty.epfl.ch/>) is a compiler for Scala that is being developed by EPFL that is faster, easier to maintain and evolve. On the language level, it simplifies Scala by removing extraneous syntax (e.g. no XML literals), and boiling down Scala's types into a smaller set of more fundamental constructs.

Activities & contributions:

- Co-designed architecture of compiler middle-end and backend;
- Bootstrapped the compiler;
- Tracked down a major performance bottleneck in current Scala compiler that is bad memory locality and long object retention;
- Co-designed and implemented the notion of Mini-Phases, that avoids the bottleneck found in current Scala compiler. Mini-Phases are also a convenient abstraction that allows to express AST transformations in an isolated and maintainable way, while fusing them together in runtime for performance;
- Co-designed and implemented YCheck, extensible self-verification infrastructure of Dotty that is the basis of the continuous integration and testing of the Dotty compiler;
- Co-designed Typed AST(TASTY) – a new interchange format to be used by Scala compilers and tools in Scala ecosystem.
- Implemented many phases of compiler, including: type erasure, recursive call optimization, lazy vals transformation, pattern matching;

*ScalaBlitz* (<https://scala-blitz.github.io/>). A data-parallel programming framework that optimizes collection operations and offers superior performance to that provided by the Scala standard library collections, by reducing abstraction overheads and taking advantage of code-patterns that contemporary Java VMs and CPUs can execute efficiently.

Activities & contributions:

- Co-designed and implemented macro-based parallel collections;
- Performed rigorous benchmarking, including low-level assembly benchmarking;
- Obtained performance comparable to hand-tuned code written in C++ that uses Intel Threading Building Blocks library;
- Developed a method that allows applying optimizations available in *ScalaBlitz* without modifying legacy code.

**Co-founder, technical lead**, *Center of Distance Education*

February 2008 - June 2012, Moscow, Russia

Co-founded a startup together with two professors from Moscow Institute of Physics and Technology. A startup around a distributed system for performing big-scale near-realtime video broadcasting. The intended user-base are students that plan to take high school exit exams and want to get tutoring from best teachers available in university.

Most notorious event had to do with one of our big video broadcasts to around 18000 students (1 Mbit/second per student on average. More at peak times that happen at the same time for all students). Our load triggered connectivity issues in several data-centers that ignored our warnings that were sent weeks upfront. This was a good trial of our fail-over mechanism that worked perfectly, hiding the issue from users.

Responsibilities & activities:

- Designing a high-throughput distributed system from scratch with hard requirements on user-experience and failover times;
- Optimizing the system to reduce operational costs;
- Hiring people to perform various tasks for project, including forming new teams of developers and tracking their progress as well as training them to use novel technology;
- Performing long-term technical planning and participating and evaluating long-term technical opportunities for the business;
- Making sure that system can run under high load safely if I'm on an multi-hour exam and team has knowledge how to react in case of failures in my absence.

**Project Lead**, *Moscow Institute of Open Education*

June 2012 - July 2014, Moscow, Russia

All Russian students take subject exams at the same day after finishing high-school. I was leading governmental project to migrate those exams from paper to an automatic web-based system that would severely reduce operational costs and time needed to check the exams.

The system had to be easy to use both for students as well as people checking the submissions. Semi-automatic graders were provided to ease the work of people evaluating the solutions such as pre-grading and custom techniques used to assign similar solutions to the same graders.

Responsibilities & activities:

- Gathering and analyzing requests from business and governmental customers.
- Taking care of formal standards of private data protection and data retention. Preparing system for governmental certification;
- Developing project architecture and documentation, based on orchestration of multiple cloud systems(Amazon AWS and MS Azure) to support project server architecture during high-load;
- Collaborating with other teams to integrate statistical intrusion detection system and reporting to track causality in the running production system;
- Managing a team of 5 developers.

**Software Developer intern**, *Wikimart.ru*  
February 2012 - July 2012, Moscow, Russia

The Wikimart is a Amazon-like system where users can look for products offered by Wikimart. The most common type of query was a range query, e.g. a query on product price. The underlying system used Cassandra, where those queries are executed very inefficiently, requiring a full scan of stored data. Most known algorithms that improve execution time of such queries require use of locks and hence are inefficient in distributed systems. On the contrary, usage of Fenwick trees does not require forced synchronization and provides eventual consistency guaranties with logarithmic time per operation. I applied these trees as indices.

Responsibilities & activities:

- Analyzing production system to isolate a bottleneck in performance;
- Designed a novel algorithm for range queries, implemented and deployed it. Which led to reduction of the average response time from 300ms to 20ms, while 99-percentile decreased from 1500ms to 300ms.

**Researcher**, *Keldysh Institute of Applied Mathematics*  
June 2011 - July 2013, Moscow, Russia

Responsibilities & activities:

- Modification of Treibers Intelligent Driver Model for multiple number of road lanes, training it on the transport flow of Moscow and then applying it to analyze the behavior on the Moscow Ring Road;
- Development of the practical algorithm that finds the shortest path with specified accuracy in graphs with the known dynamics of edge changes, e.g. the graph obtained from the trained Treibers Intelligent Driver Model. This algorithm is a modification of Dijkstras algorithm in the external memory, with ALT-modification and NaturalCuts heuristics.

## PUBLICATIONS

- Prokopec A., Petrashko D., Odersky M. “Efficient Lock-Free Work-stealing Iterators for Data-Parallel Collections.” *Parallel, Distributed and Network-Based Processing (PDP), 2015 23rd Euromicro International Conference on.* IEEE, 2015;
- Petrashko D. “Investigation on transport flow behavior depending on safe distance” *54th Moscow Institute of Physics and Technology conference: Problems of fundamental, applied and technical sciences in contemporary society*, 2012, Russia, Moscow, P. 99-103;
- Gasnikov A., Dorn Y., Ivkin N., Ishmanov M., Obidina T., Petrashko D, Holodov Y., Hohlov M., Chehovich Y. “Some actual problems of traffic flow mathematical modeling” *Intelligent Information Processing of the 9th International Conference, IIP-2012*, Montenegro, Budva, P. 211-214

- Gasnikov A., Gasnikova E., Petrashko D. “Macro-system approach to web-page ranking models”; *Information Technology and Systems conference*, 2012, Russia, Petrozavodsk.

## SUBMITTED/IN PREPARATION

- Petrashko D., Lhoták O., Ureche V., Odersky M. “Call Graphs for Languages with Parametric Polymorphism”. Accepted to OOPSLA 2016;
- Petrashko D., Lhoták O., Ureche V., Odersky M. “Generic specialization using Call Graphs”.
- Petrashko D., Lhoták O., Renucci A., Odersky M. “Extending CSE to Idempotent Scala expressions”.

## SELECTED CONFERENCE TALKS

- D. Petrashko “How do we make the Dotty compiler fast”, JVM Language summit 2016, Santa Clara, August 1st-4th, 2016;
- D. Petrashko “Dotty Linker: Precise Types Bring Performance”, ScalaDays 2016, New York, May 9th-13th, 2016;
- D. Petrashko “Scala & Dotty current status”, invited keynote, ScalaUA 2016, Kiev, April 8th, 2016;
- D. Petrashko “Making sense of initialization order in Scala”, invited keynote, Scalar 2016, Warsaw, April 16th, 2016;
- D. Petrashko “Whats new in Dotty”, Fby.by: functional conference of Belarus, Minsk, Nov 28, 2015;
- D. Petrashko “Dotty: Exploring the future of Scala”, invited, ScalaWorld Lake District, UK, 2015;
- D. Petrashko “Making your Scala applications smaller and faster with the Dotty linker“, Scaladays, Amsterdam, Jun 8-10, 2015;
- D. Petrashko “Lightning-Fast Standard Collections With ScalaBlitz”, Scala Days, Berlin, Jun 16-18, 2014;
- A. Prokopec, D. Petrashko “Macro-based Scala Parallel Collections”, Scala eX-change, London, Dec 2-3, 2013.

## EDUCATION

**Ph.D.:** *École Polytechnique Fédérale de Lausanne*

September 2013 - Present, Lausanne, Switzerland

Topics: link-time optimizations, static analysis, parallel datastructures, language design.

Supervisor: Martin Odersky.

*Yandex School of Data Analysis*

September 2011 - June 2013, Moscow, Russia

Topics: machine learning, I/O efficient algorithms, natural language processing, discrete analysis and probability theory.

GPA: 4.91/5.

**B.Sc.:** *Moscow Institute of Physics and Technology*

September 2008 - June 2012, Moscow, Russia

Topics: applied mathematics, algorithms and data structures.

Thesis: “Use of persistent Fenwick trees to accelerate range queries in distributed non-relational database Cassandra”.

GPA: 4.88/5 (diploma with honors).