Andrei Kashin

Resume

Info

Email `career@akashin.me`

Webpage `https://akashin.me`

Experience

Google DeepMind

2017-Now **Senior Software Engineer**, London, United Kingdom

Working on a platform for fast and reproducible research experiments.

- Implemented core components of XManager platform to provide fair distribution and high utilization of compute resources for DeepMind and Alphabet researchers.

Yandex

2014-2017 **Senior Software Engineer**, Moscow, Russia

Worked on distributed highly available storage and computational platform (YT).

- Scaled MapReduce scheduler component to support 100k of computing cores and 1000s of users by parallelizing resource allocation algorithm using persistent data structures.
- Enabled highly configurable resource allocation component (supports preemption, guaranteed resources, hierarchical weights) by designing and implementing novel resource fairness algorithm.

Facebook

2014 **Software Engineer Intern**, *3 months internship*, Menlo Park, California

Worked on Scuba, distributed in-memory database and real-time processing engine for interactive data analysis.

- Increased stored data retention time 10 times by implementing caching layer for SSD storage.
- Enabled user defined arithmetic expressions in query language by building expression evaluation engine.
- Supported multidimensional histograms and aggregates computation.

Education

2015-2017 **Master's Degree in Computer Science**, *NRU Higher School of Economics*, Russia

2012-2015 **Yandex School of Data Analysis, Computer Science Specialization**, *Yandex*, Russia

2011-2015 **Bachelor's Degree in Computer Science**, *Moscow Institute of Physics and Technology*, Russia

Online courses

2013-2015 **Mining Massive Datasets, Machine Learning, Discrete Optimization, Cryptography I**, *Coursera*

Competitions

2016 **5th place at Challenge24, Top10 at Deadline24, Top100 at VKCup**

2015 **Top10 at Challenge24, Top200 at Facebook Hacker Cup**

2011-2016 **Top500 at Google Code Jam, Top500 at Distributed Code Jam 2016**

Projects

FAML Implementation of machine learning algorithms in C++. Uses linear algrebra library Eigen and C++11.
Project site: `https://github.com/akashin/faml`

Oberon0 Interpreter Implementation of Oberon0 interpreter. Built using C++, Flex and Bison.
Project site: `https://github.com/akashin/Oberon0-Interpreter`