Engineering Distributed Systems that We Can Trust (and Also Run)

Ilya Sergey
Yale-NUS College and National University of Singapore
Singapore
ilya.sergey@yale-nus.edu.sg

ABSTRACT

The interest in formal methods and verification of correctness-critical distributed systems is on the rise in the past few years. But what are the gains from proving statements about software in full mathematical rigour? Do they justify the high cost of verification? And how far can we extend our trust in formal methods when talking about realistic distributed systems and their client programs?

This talk is in three parts. First, I will provide an overview of the state of the art in machine-assisted reasoning about distributed consensus protocols, their implementations, and applications. Next, I will discuss the trade-offs that have to be made in order to enable mechanised proofs about runnable systems code, as well as implications of the assumptions made to describe the real-world execution environments. Lastly, I will focus on the ongoing work propelled by the programming languages community towards engineering modular proofs about distributed protocols—a way to build correct-by-construction composite systems from verified reusable components.

CCS Concepts/ACM Classifiers

Theory of computation, Logic and verification; Software and its engineering, Distributed programming languages

Author Keywords

Verification; Formal Methods; Distributed Systems; Proofs; Modularity

BIOGRAPHY

Ilya Sergey is an Associate Professor at Yale-NUS College and at the School of Computing of the National University of Singapore. He was previously a faculty at University College London and a postdoctoral researcher at IMDEA Software Institute. He holds a PhD in Computer Science from KU Leuven. Before joining academia he worked as a software engineer at JetBrains. He does research in programming language theory, including, but not limited to types, semantics, software verification, and program synthesis. Lately, he has been mostly focusing on developing sound and scalable methodologies for building provably correct concurrent and distributed systems. Dr. Sergey is a recipient of a 2017 Google Faculty Award and the AITO Dahl-Nygaard Junior Prize 2019 for his contributions to the development and application of programming language techniques to various problems across the programming spectrum, covering object-oriented, functional, distributed, and concurrent programming. He has also designed and co-developed Scilla, a programming language for safe and secure smart contracts, used by Zilliqa, a Singapore blockchain start-up.

PODC '19, July 29–August 2, 2019, Toronto, ON, Canada.

ACM ISBN 978-1-4503-6670-0/19/06.
DOI: https://doi.org/10.1145/3293611.3338839