

Andi Bejleri

CONTACT INFORMATION

Lausteschläger strasse 36
64289 Darmstadt
Germany

E-mail: bejleri@cs.tu-darmstadt.de
Mobile: +49 1525 289-7049

AREAS OF SPECIALISATION

Theory of computation: Distributed, Concurrency, Computability, Logic, Semantics and reasoning;
Information systems: MapReduce-based systems, Database transaction processing;
Software Architecture: Object oriented, Cooperating communicating processes, Publish-subscribe/
event-based, Tightly coupled;
Software and its engineering: Correctness, Software verification and validation, Formal methods,
Formal language definitions;
General programming languages: Java, Scala, C, C++, Smalltalk, Ruby, Python, Javascript;
System description languages: WS-CDL, BPEL;

EDUCATION

Imperial College, London, UK

Ph.D. in Computing, 2011
M.S. in Advanced Computing, 2007 (Distinction on master project)

University of Pisa, Pisa, Italy

B.S. in Computer Science, 2005 (Graduated with Summa Cum Laude)

Carnegie Mellon University, Pittsburgh, USA

Senior Thesis, Computer Science, 2004

PROFESSIONAL EXPERIENCE

Technische Universität, Darmstadt, Germany

Postdoctoral Researcher

Software Technology Group

October, 2013 - present

Supervised work-packages in the *PACE* project: design, implement, and validate a programming model to radically improve the development and quality of applications in cloud environment.

1. Defined correctness properties and proofs for a scheduler based on the multi-version control transaction algorithm.
2. Designed a model of *cooperative decoupled processes* to support reasoning of control and shared events without sacrificing decoupling for event-driven programming.
3. Designed a type system, and defined properties along with proofs for a co-contextual, bottom-up type checker for a formal model of Java that allows incremental type-checking.
4. Authored and co-authored 5 papers

Distributed Systems Group

January, 2015 - September, 2016

Supervised work-packages in the *LiveSoft* project: static techniques to verify failure-prone aspects of distributed software.

1. Designed a model and type theory for structured interactions in message-passing distributed systems.
2. Designed and implemented (a) a spec language to model structured interactions and fine grained failure handling; (b) a Java library to implement fault tolerant distributed systems; (c) a static analysis to check conformance of corresponding software to their specification.
3. Authored and co-authored 2 papers.

Council of Ministers, Tirana, Albania

Advisor to the Prime Minister

September, 2012 - August, 2013

1. Developed a *national strategy for Cloud computing* to decrease IT spendings.
2. Designed and implemented a project of an *electronic register* in pre-university education to strengthen ties between teachers, students and parents.
3. Developed an electronic platform of petitions for communication with the citizens to increase open-governance.
4. Contributed in establishing a *centre for talented youth* for pre-university students to produce new elites.

HP Labs, Bristol, UK

Researcher

March, 2010 - February, 2011

Oversaw the works in the *SmartFrog* project: a framework for configuring and automatically activating distributed applications.

1. Designed an agent-based model to *specify software management systems* that (a) coordinates and controls components of a distributed computation.
2. Implemented a *Test-as-a-Service* (TaaS) for HP's G-Cloud demo.
3. Authored 1 paper.

Imperial College, London, UK

Research Assistant

November, 2007 - January, 2010

Supervised work-packages in the *Type-based security analysis for safe mobile computing* project: high-level structure architectural description of a distributed system in terms of its types.

1. Designed a type theory for *synchronous structured interactions* that guarantees communication safety for distributed systems.
2. Performed an empirical study of the performance of a *safe Java IO* library; the study showed that the library performs competitively against *Java IO* and *RMI*.
3. Implemented a library for parallel algorithms in a *safe Java IO* library and *MPI*.
4. Designed a type theory for parallel programming to guarantee communication safety.
5. Authored and co-authored 6 papers.

LANGUAGES

English and **Italian**: Fluent; **German**: B2 (in progress); **French**: A2; **Albanian**: Native

TEACHING

Imperial College: Object Oriented Programming, Mathematical Methods, Discrete Mathematics

TU Darmstadt: Advanced Type Systems for Distributed Programming, Design and Implementation of Event-driven and Reactive Programming, Formal Models for Distributed Programs

New York University of Tirana: Advanced Topics in Software Engineering, Data Structures, Programming Languages, Analysis and Development of Software