Stephan Brandauer

Computer Science and Software Development.

Address	Dag Hammarkjölds v. 29A, 752 37 Uppsala, Sweden	Phone	+46 700 599 236
		E-Mail	stephan.brandauer@gmail.com
		Web	http://stbr.me
Nationality	Austrian	Other	sbrandauer, kaeluka
Languages	German (*native*), English (*fluent*) Swedish (*basic working proficiency*)		stephan-brandauer

Personal Profile

I have recently defended (opponent: Prof. Doug Lea, SUNY) my PhD in computer science at Uppsala University, Sweden. My research has focused on aliasing (several variables holding references to the same datum) in imperative programming languages. Aliasing makes both writing, understanding, and optimising code hard. Aliasing has been the common theme that has been tying together my research spanning from **type system/language design** to **dynamic analysis of program corpora** to a domain-specific language for **data structure-design and -optimisation**.
I deeply care about program performance, but I also think that most programmers shouldn't need to. To this end, I've been trying to find ways to constrain high level languages in just the right way: constraints that don't hurt writing code in practise, yet give enough information to compiler, runtime, or framework to do optimisations that can provide great program performance.
Although my research work's focus has been quite narrow, at this point I'm mainly interested in getting experience in software engineering: I want to learn all about how your team works together, the business domains you're solving, how you're communicating with your customers. Basically: all the good things I have missed out on during my PhD.

Education

2013-2018 PhD in Computer Science (defended in Jan, 2019) - Uppsala University

Research on programming language design, analysis, and implementation.

- I designed *Disjointness Domains*, a type system to express fine grained alias invariants in data structures.
- I designed and implemented *Spencer*, a dynamic analysis tool that runs real world Java software, collects extensive program traces and lets users analyse these traces using a specifically designed *domain-specific language* executed by a *web service*.
- I designed and implemented *C♭*, a domain-specific language and compiler using Java's annotation framework that let users implement data structures that are simple, have good performance, and can adapt their performance to match a wide range of use cases.
- I worked, with others, on the compiler of the *Encore* research language, an object oriented programming language with concurrently executing actors as objects.

2011-2013 Master of Science, Computer Science. Uppsala University.

Degree project: design, implement, and benchmark a mailbox data structure for an actor-based language (called "Joelle") that permits *parallel processing of messages within an actor.*

2007-2011 Bachelor of Science, Cognitive Informatics. Bielefeld University.

Degree project: Build a 2D rigid body and particle physics engine with an interactive UI.

Selected Publications

Onward!'18 *C♭: A New Approach to Efficient and Tunable Collections*

St. Brandauer, E. Castegren, T. Wrigstad

Onward!: Symposium on New Ideas in Programming and Reflections on Software 2018. Boston, USA.

A domain specific language to develop data structures that are simple, have good performance, and can adapt to many different use cases.

`http://stbr.me/cflat`

MSR'17 Spencer: Interactive Heap Analysis for the Masses

St. Brandauer and T. Wrigstad

Int'l Conf. on Mining Software Repositories (MSR) 2017. Buenos Aires, AR.

The paper that introduces the Spencer project.

`http://stbr.me/spencer`

QAPL'17 Mining for Safety using Interactive Trace Analysis

St. Brandauer and T. Wrigstad

Workshop on Quantitative Aspects of Programming Languages and Systems (QAPL) 2017. Uppsala, SE.

An application of Spencer to a corpus of programs, looking for safety properties of objects (like uniqueness, immutability, etc).

`http://stbr.me/spencer`

SFM'15 Parallel Objects for Multicores: A Glimpse at the Parallel Language Encore

St. Brandauer, E. Castegren, D. Clarke, F. Fernández, E. Broch Johnsen, Ka I Pun, S. Lizeth Tapia Tarifa, T. Wrigstad, and A. Yang

15th Int'l School on Formal Methods f. Design of Comp., Comm. and Software Systems (SFM) 2015. Bertinoro, IT.

An overview of the Encore language.

`http://stbr.me/Encore-Glimpse`

OOPSLA'15 Disjointness Domains for Fine-Grained Aliasing

St. Brandauer, D. Clarke, and T. Wrigstad

Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) 2015. Pittsburgh, PA, USA.

A novel type system for alias control.

`http://stbr.me/Disjointness-Domains-for-Fine-Grained-Aliasing`

Employment History

2013 - 2018 PhD Student in programming language design, implementation, analysis.

Uppsala University

See in Education section above.

Feb 2009 - Jun 2010 Research Assistant

Bielefeld University, AI Group

Work on, and maintain, virtual reality applications for cognitive science studies. Teaching assistant (run lab sessions, grade home work assignments).

Sep 2007 Mar 2008 Freelance Programmer, C#

Comet Consulting, Salzburg (now part of Rhomberg)

Develop 3D image recognition algorithms and software in **C#** for 3D LIDAR scanners to monitor safety procedures at railway tunnel construction sites. Much of the job was on-site, but overlap with the social work job was during nights and evenings.

Sep 2006 – Mar 2008 **Social Work**
Laube Sozialpsychiatrische Aktivitäten GmbH

Austrian civil service, as an alternative to being drafted for the military. Work with chronically mentally ill people. Learned lots.

Software Engineering Skills

- ## Programming Languages

  (roughly in order of familiarity)

  Java - My go-to language.

  Scala - have used it on several occasions, for example the *Spencer* and *C♭* DSL-compilers.

  Haskell - used (and loved) it lots for the first 3 years of my PhD studies, working on the *Encore* compiler.

  Rust - Have implemented a throw-away prototype version of *C♭*.

  C - have been teaching basic C to university students every year of my PhD.

  C++ - know the core principles that distinguish it from C (smart pointers, RAII, objects, zero-cost abstractions, templates, . . .), but would like to know more.

  SQL (Postgres-SQL) - have used **Postgres-SQL** to implement complex graph queries in *Spencer*.

- ## Miscellaneous

  Data Analysis: I have used **python/pandas**, **Apache Spark** (single node; also **GraphX**), and **Postgres**.

  Optimisation of JVM code: optimisation is a big part of my research work on *C♭*. I have used the **JMH** framework, **VisualVM**, and **JITWatch**.

  Version control: mostly using **git**. Also, although rusty, **Mercurial** and **SVN**.

  Compilers: I have worked on several **compilers**: one for a general purpose language (*Encore*) and have developed two compilers for domain-specific languages (for *Spencer*, compiling to SQL; and for *C♭*, compiling to Java bytecode).

  Java Bytecode: I have used both the **ASM framework** and **bytebuddy** to a) **programmatically modify Java programs as they are running**, and b) **generate Java code** from high level specifications.

Teaching Assistant

Fall '17	■ Imperative and Object Oriented Programming Methodology ■ Secure Computer Systems ■ Introduction to Parallel Programming
Fall '16	■ Imperative and Object Oriented Programming Methodology ■ Algorithms and Data Structures 2
Fall '15	■ Imperative and Object Oriented Programming Methodology ■ Project CS (group project using Scrum and Erlang)
Fall '14	■ Imperative and Object Oriented Programming Methodology ■ Project CS (group project using Scrum and Erlang)
Fall '13	■ Imperative and Object Oriented Programming Methodology

Community Contributions

2017—present	Co-founder/organiser of "Papers&Pizza", a semi-regular series of presentations by/for local PhD students and friends from different groups at the IT Department at Uppsala University.
ECOOP'17	Member of Artifact Evaluation Committee

OOPSLA'16	Member of Artifact Evaluation Committee
Annual Scala Workshop'14	Local arrangements
ECOOP'14	Web master
ECOOP'14	Student Volunteer Programme: local organiser
various	Student volunteer

References

Tobias Wrigstad PhD advisor Associate Professor at Uppsala University
`tobias.wrigstad@it.uu.se`
He has supervised both my master's and my PhD thesis work.

Sophia Drossopoulou Research collaborator Professor at Imperial College London
`s.drossopoulou@imperial.ac.uk`
We were both members of the Encore team that involved groups from several universities.

Hobbies

Food (producing and consuming), and photography (`https://www.instagram.com/kaelukaphotos/`).