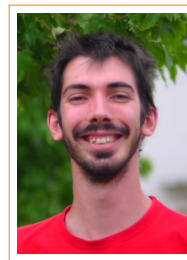


Gabriel Scherer

PRL, Northeastern University, Boston
✉ gabriel.scherer@gmail.com
🏠 www.ccs.neu.edu/home/gasche/



I am currently a post-doc working with [Amal Ahmed](#) in the [PRL group](#) at Northeastern University, Boston.

I was previously a PhD student under the supervision of [Didier Rémy](#) in the [Gallium project-team](#) at INRIA Paris-Rocquencourt.

Publications

- 2017 Gabriel Scherer. “[Search for Program Structure](#)”. *SNAPL*.
Gabriel Scherer. “[Deciding equivalence with sums and the empty type](#)”. *POPL*.
- 2015 Gabriel Scherer and Didier Rémy. “[Which simple types have a unique inhabitant?](#)” *ICFP*.
Guillaume Munch-Maccagnoni and Gabriel Scherer. “[Polarised Intermediate Representation of Lambda Calculus with Sums](#)”. *LICS*.
Gabriel Scherer. “[Multi-focusing on extensional rewriting with sums](#)”. *TLCA*.
Gabriel Scherer and Didier Rémy. “[Full reduction in the face of absurdity](#)”. *ESOP*.
Pierre-Évariste Dagand and Gabriel Scherer. “[Normalization by realizability also evaluates](#)”. *JFLA*.
- 2013 Gabriel Scherer and Jan Hoffmann. “[Tracking Data-Flow with Open Closure Types](#)”. *LPAR*.
Gabriel Scherer and Didier Rémy. “[GADTs meet subtyping](#)”. *ESOP*.
- 2012 Andreas Abel and Gabriel Scherer. “[On Irrelevance and Algorithmic Equality in Predicative Type Theory](#)”. *Logical Methods in Computer Science* (2012).
- 2010 Gabriel Scherer and Jérôme Vouillon. “[Macaque: Interrogation sûre et flexible de bases de données depuis OCaml](#)”. *JFLA*.

Teaching

- 2014–2015 Teaching assistant: Java (bachelor), C (bachelor)
- 2013–2014 Teaching assistant: Java (bachelor), Advanced Functional Programming (master)

2008–2010, Teaching assistant: Caml-Light (bachelor)
2012

2006–now Helping beginners learn programming in all kinds of online forums

Thesis

Which types have a unique inhabitant?
Focusing on pure program equivalence

Supervised by [Didier Rémy](#).

Defended on March 30th, 2016.

Abstract:

Some programming language features (coercions, type-classes, implicits) rely on inferring a part of the code that is determined by its usage context. In order to better understand the theoretical underpinnings of this mechanism, we ask: when is it the case that there is a unique program that could have been guessed, or in other words that all possible guesses result in equivalent program fragments? Which types have a unique inhabitant?

To approach the question of unicity, we build on work in proof theory on more canonical representation of proofs. Using the proofs-as-programs correspondence, we can adapt the logical technique of focusing to obtain more canonical program representations.

In the setting of simply-typed λ -calculus with sums and the empty type, equipped with the strong $\beta\eta$ -equivalence, we show that uniqueness is decidable. We present a saturating focused logic that introduces irreducible cuts on positive types “as soon as possible”. Goal-directed proof search in this logic gives an effective algorithm that returns either zero, one or two distinct inhabitants for any given type.

Presentations in peer-reviewed workshops

2016 Gabriel Scherer, Luc Maranget, and Thomas Réfis. “[Ambiguous pattern variables](#)”. *ML workshop*.

2014 Thomas Braibant (presenter), Jonathan Protzenko, and Gabriel Scherer. “[Well-typed generic smart-fuzzing for APIs](#)”. *ML Workshop*.

Gabriel Scherer. “[Deciding unique inhabitants with sums](#)”. *TYPES*.

Gabriel Scherer. “[Github Pull Requests for OCaml development: a field report](#)”. *OCaml Workshop*.

2013 Gabriel Scherer. “[Mining opportunities for unique inhabitants in dependent programs](#)”. *Dependently Typed Programming (DTP)*.

2012 Gabriel Scherer and Didier Rémy. “[GADT meet subtyping](#)”. *ML Workshop*.

Research internships

2012 Visit to the FLINT team in Yale, lead by Zhong Shao

- 2011 “Universe subtyping in Martin-Löf Type Theory”, master thesis supervised by Andreas Abel (TCS, LMU, Munich)
- 2010 “ ML_{ω}^F : extending ML^F to higher-kinded types”, supervised by Didier Rémy (Gallium, INRIA, Rocquencourt)
- 2009 “Safe and flexible database queries from OCaml”, supervised by Jérôme Vouillon (PPS, CNRS, Paris)
- 2009 Bachelor thesis, “Goodstein Sequences and Incompleteness of Peano Arithmetic”, with Silvain Rideau, supervised by François Loeser (ENS)
- 2008 Bachelor thesis (TIPE) : “Monads, Category Theory and Functional Programming”

Reviews

- 2017 ICFP’17 (program committee member, 18 reviews)
- 2016 LICS’16 (three reviews), OOPSLA (three reviews), MSCS (two journal reviews)
- 2015 TYPES’14 (one review), ICFP’15 (two reviews), ML’14 post-proceedings (one review)
- 2014 POPL’15 (two reviews), JFLA’15 (one review)
- 2013 POPL’14 (one review), ESOP’14 (two reviews)
- 2012 ESOP’12 (one review)

Software skills

- Languages OCaml, Coq, Haskell, C, Oz, Racket, Prolog, Python, MIPS asm
- Tools GNU/Linux, Git, Darcs, Emacs, \LaTeX

Education

- 2016–now **Post-Doc**, working with [Amal Ahmed](#), Northeastern University, Boston.
- 2011–2016 **Doctoral programme**, PhD student in project-team Gallium, INRIA, France.
- 2009–2011 **Master**, Master Parisien de Recherche en Informatique.
- 2008–2009 **Licence**, ENS : math major, minor in computer science.
- 2008–2012 student in “École Normale Supérieure (Paris)” (ENS)
- 2006–2008 **Classe préparatoire**, MPSI and MP* in lycée Louis-le-Grand, Paris.