

Curriculum Vitae

Derek Dreyer

Personal Information

Affiliation:	Prof. Dr. Derek Dreyer Scientific Director, MPI-SWS Honorarprofessor, Saarland University	Citizenship:	USA
		Birthdate:	May 2, 1980
		E-mail:	dreyer@mpi-sws.org
Address:	MPI-SWS Saarland Informatics Campus E1.5 66123 Saarbrücken, Germany	Office Phone:	+49 681 9303 8701
		Mobile Phone:	+49 160 9729 8151
		URL:	www.mpi-sws.org/~dreyer

Academic Background

Carnegie Mellon University

Ph.D. in Computer Science, May 2005. 1997–2005
Advisors: Robert Harper, Karl Crary.

New York University

B.A. in Mathematics and Computer Science, Summa Cum Laude, May 1996. 1993–1996

Employment History

Max Planck Institute for Software Systems (MPI-SWS)

Head of the “Foundations of Programming” Group.

- Scientific director, MPI-SWS. 2022–present
- Honorarprofessor, Saarland University. 2017–present
- Tenured faculty, MPI-SWS. 2013–2022
- Tenure-track faculty, MPI-SWS. 2008–2013

Toyota Technological Institute at Chicago (TTI-C)

Research assistant professor (3-year independent postdoc). 2005–2007

Awards, Grants, and Fellowships

PLDI’22 Distinguished Paper Award	2022
“Busy Beaver Award” for Outstanding Commitment to Teaching (“Semantics”, Winter 2021-22)	2022
POPL’22 Distinguished Paper Award	2022
PLDI’21 Distinguished Paper Award	2021
PLDI’21 Distinguished Artifact Award	2021
Google ASPIRE funding for RefinedC and pKVM Verification (\$470K, joint with Deepak Garg)	2020-2023
Google PhD Fellowship (3 years of funding for Michael Sammler, Ph.D. student)	2020
OOPSLA’18 Distinguished Reviewer Award	2018

ACM SIGPLAN Robin Milner Young Researcher Award	2017
OOPSLA'17 Distinguished Paper Award	2017
ECOOP'17 Distinguished Paper Award	2017
PLDI'17 Distinguished Paper Award	2017
ERC Consolidator Grant (2016–2021, approx. 2 million €) for <i>RustBelt: Logical Foundations for the Future of Safe Systems Programming</i>	2015
Microsoft Research PhD Scholarship (3 years of funding for David Swasey, Ph.D. student)	2013
Google European Doctoral Fellowship (3 years of funding for Georg Neis, Ph.D. student)	2012

Professional Activities

Co-editor-in-chief, Journal of Functional Programming (**JFP**), February 2022–present.

General chair, 2019 ACM SIGPLAN International Conference on Functional Programming (**ICFP 2019**).

Associate editor, ACM Transactions on Programming Languages and Systems (**TOPLAS**), July 2017–present.

Editor, Journal of Functional Programming (**JFP**), July 2019–January 2022.

Member of the editorial board, Journal of Functional Programming (**JFP**), Feb. 2014–June 2019.

Member-at-large (elected) and awards chair, ACM SIGPLAN Executive Committee, July 2012–June 2015.

Member, IFIP Working Group 2.8 – Functional Programming, Aug. 2014–present.

Workshop founder, ACM SIGPLAN Workshop on Higher-Order Programming with Effects (**HOPE**), 2012.

Lead organizer, Dagstuhl Seminar on “Compositional Verification Methods for Next-Generation Concurrency”, May 2015.

Lead organizer, The Cornell, Maryland, Max Planck Pre-doctoral Research School 2018 (**CMMRS 2018**).

Co-editor, Journal of Functional Programming (**JFP**), Special issue devoted to ICFP 2014.

Co-editor, Journal of Functional Programming (**JFP**), Special issue for Robert Harper Festschrift.

Workshop (co-)chair/organizer:

- 2015–16 Programming Languages Mentoring Workshop (**PLMW 2015–16**).
- 2014 Coq Workshop (**Coq 2014**).
- 2012–13 ACM SIGPLAN Workshop on Higher-Order Programming with Effects (**HOPE 2012–13**).
- 2011 ACM SIGPLAN Workshop on Types in Language Design and Implementation (**TLDI 2011**).
- 2007 ACM SIGPLAN Workshop on ML (**ML 2007**).

Program committee (PC) member:

- 2023 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (**OOPSLA 2023**).
- 2022 International Conference on Computer-Aided Verification (**CAV 2022**).
- 2022 European Conference on Object-Oriented Programming (**ECOOP 2022**).
- 2019 European Conference on Object-Oriented Programming (**ECOOP 2019**).
- 2018 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (**OOPSLA 2018**).

- 2017 ACM SIGPLAN Symposium on Principles of Programming Languages (**POPL 2017**).
- 2016 International Conference on Formal Structures for Computation and Deduction (**FSCD 2016**).
- 2015 Conference on the Mathematical Foundations of Programming Semantics (**MFPS 2015**).
- 2015 European Symposium on Programming (**ESOP 2015**).
- 2014 ACM SIGPLAN International Conference on Functional Programming (**ICFP 2014**).
- 2013 ACM SIGPLAN Workshop on ML (**ML 2013**).
- 2013 International Conference on Certified Programs and Proofs (**CPP 2013**).
- 2013 European Symposium on Programming (**ESOP 2013**).
- 2012 International Conference on Compiler Construction (**CC 2012**).
- 2011 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (**POPL 2011**).
- 2010 ACM SIGPLAN International Workshop on Foundations of Object-Oriented Languages (**FOOL 2010**).
- 2009-10 International Workshop on Modules and Libraries for Proof Assistants (**MLPA 2009-10**).
- 2009 Conference on the Mathematical Foundations of Programming Semantics (**MFPS 2009**).
- 2008 ACM SIGPLAN International Conference on Functional Programming (**ICFP 2008**).
- 2007 ACM SIGPLAN Haskell Workshop (**Haskell 2007**).
- 2007 ACM SIGPLAN International Workshop on Foundations and Developments of Object-Oriented Languages (**FOOL/WOOD 2007**).
- 2006 ACM SIGPLAN Workshop on ML (**ML 2006**).

Selection committee member:

- 2019-20 ACM SIGPLAN Robin Milner Young Researcher Award.
- 2017 EAPLS Best PhD Dissertation Award.
- 2017-18 ACM SIGPLAN John C. Reynolds Doctoral Dissertation Award.
- ICFP 2015 Student Research Competition.

PhD thesis committee member or external examiner:

- Milijana Surbatovich, Carnegie Mellon University (advisors: Brandon Lucia, Limin Jia), 2023 (expected).
- Julian Sutherland, Imperial College London (advisor: Philippa Gardner), 2022 (expected).
- Denis Merigoux, INRIA (advisors: Karthikeyan Bhargavan, Jonathan Protzenko), 2021.
- Aymeric Fromherz, Carnegie Mellon University (advisor: Bryan Parno), 2021.
- Jonas Kastberg Hinrichsen, IT University of Copenhagen (advisor: Jesper Bengtson), 2021.
- Dan Frumin, Radboud University Nijmegen (advisors: Herman Geuvers, Robbert Krebbers, Freek Wiedijk), 2021.
- Piotr Polesiuk, University of Wrocław (advisor: Dariusz Biernacki), 2021 (expected).
- Vineet Rajani, MPI-SWS and Saarland University (advisor: Deepak Garg), 2020.

- Steven Schäfer, Saarland University (advisor: Gert Smolka), 2019.
- Christopher Pulte, University of Cambridge (advisor: Peter Sewell), 2019.
- Jeehoon Kang, Seoul National University (advisor: Chung-Kil Hur), 2019.
- Joseph Tassarotti, Carnegie Mellon University (advisor: Robert Harper), 2019.
- Robin Morisset, École Normale Supérieure de Paris (advisor: Francesco Zappa Nardelli), 2017.
- Edward Yang, Stanford University (advisors: David Mazières, John Mitchell), 2017.
- Azalea Raad, Imperial College London (advisor: Philippa Gardner), 2017.

External review committee (ERC) member:

- 2016 ACM SIGPLAN International Conference on Functional Programming (**ICFP 2016**).
- 2016 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (**POPL 2016**).
- 2015 ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI 2015**).
- 2015 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (**POPL 2015**).
- 2013 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (**POPL 2013**).
- 2012 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (**POPL 2012**).
- 2009 ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI 2009**).

Steering committee (SC) member:

- ACM SIGPLAN International Conference on Functional Programming (**ICFP**), 2016–2021.
- ACM SIGPLAN Programming Languages Mentoring Workshop (**PLMW**), 2016–2020.
- ACM SIGPLAN Workshop on Types in Language Design and Implementation (**TLDI**), 2011–2012.
- ACM SIGPLAN Workshop on ML, 2008–2010.

Invited speaker:

- Online Worldwide Seminar on Logic and Semantics (OWLS), December 2020.
- Seminar Series on Mechanized Semantics, Collège de France, February 2020.
- Distinguished Colloquium, University of Maryland, September 2018.
- 2018 European Symposium on Programming (**ESOP 2018**).
- **Milner Award Lecture:**
2018 ACM SIGPLAN Symposium on Principles of Programming Languages (**POPL 2018**).
- Computer Science Colloquium, Cornell University, November 2017.
- 2017 Workshop on Formal Techniques for Java-like Programs (**FTfJP 2017**).
- 2017 Conference on the Mathematical Foundations of Programming Semantics (**MFPS 2017**).
- ACM SIGPLAN Programming Languages Mentoring Workshop (**PLMW**) at ICFP 2021, PLDI 2021, POPL 2020, SPLASH 2019, ICFP 2019, POPL 2018, ICFP 2017, POPL 2017, ICFP 2016, POPL 2016, POPL 2014.
- Distinguished Lecture Series, University of Chicago, May 2016.

- 2016 South of England Regional Programming Languages Seminar (**S-REPLS 3**).
- Workshop on Certification of High-Level and Low-Level Programs, Institut Henri Poincaré thematic trimester on Semantics of Proofs and Certified Mathematics, July 2014.
- 2014 Oregon Programming Languages Summer School (**OPLSS 2014**).
- 2012 Parametricity Workshop, Glasgow, UK.
- 2011 International Workshop on Logical Frameworks and Meta-Languages: Theory and Practice (**LFMTP 2011**).
- 2008 Conference on the Mathematical Foundations of Programming Semantics (**MFPS 2008**).

Workshops co-chair, 2010–11 International Conference on Functional Programming (**ICFP 2010–11**).

Staff representative (Mitarbeitervertreter) of the MPI for Software Systems in the Chemistry, Physics & Technology section of the Max Planck Society, 2010–2016.

Moderator, TYPES and TYPES/announce e-mail forums, April 2009–April 2014.

Senior member, ACM SIGPLAN (Special Interest Group on Programming Languages).

Research Advisees

Postdocs at MPI-SWS:

- **Emanuele D’Osualdo** (since Sep. 2020).
- **Chung-Kil Hur** (Oct. 2010–Sep. 2012) → Microsoft Research Cambridge → Seoul National University.
- **Jacques-Henri Jourdan** (Apr. 2016–Sep. 2017) → Université Paris Saclay, CNRS, ENS Paris-Saclay, LMF.
- **Ralf Jung** (Sep. 2020–Aug. 2021), co-advised by Frans Kaashoek and Nickolai Zeldovich (MIT).
- **Neel Krishnaswami** (Sep. 2011–Sep. 2013) → University of Birmingham → University of Cambridge.
- **Ori Lahav** (Apr. 2016–Sep. 2017, co-advised by Viktor Vafeiadis) → Tel Aviv University.
- **Rodolphe Lepigre** (Jan. 2019–Jan. 2022) → BedRock Systems.
- **Pierre-Marie Pédro** (Oct. 2017–Sep. 2018) → Inria Rennes – Bretagne Atlantique.
- **Azalea Raad** (July 2017–present, co-advised by Viktor Vafeiadis) → Imperial College London.
- **Andreas Rossberg** (Aug. 2007–Jan. 2010) → Google (where he co-developed WebAssembly) → Dfinity.
- **Aaron Turon** (Jan. 2013–Apr. 2014) → Mozilla Research (where he led the Rust team) → Fastly.
Aaron Turon was recipient of the 2014 ACM SIGPLAN John C. Reynolds Doctoral Dissertation Award.
Part II of his dissertation concerns our joint work published in POPL 2013 and ICFP 2013.

Doctoral students at MPI-SWS:

- **Hoang-Hai Dang** (Apr. 2016–Feb. 2022), on leave at BedRock Systems.
- **Lennard Gäher** (since Jan. 2021).
- **Ralf Jung** (Sep. 2014–Aug. 2020) → MIT (postdoc).
 - PhD thesis (2020): *Understanding and Evolving the Rust Programming Language*.
 - Recipient of the **2020 ACM Doctoral Dissertation Award Honorable Mention**.
 - Recipient of the **2021 ACM SIGPLAN John C. Reynolds Doctoral Dissertation Award**.

- Recipient of the **2021 ETAPS Doctoral Dissertation Award**.
- Recipient of the **2021 Otto Hahn Medal**.
- Recipient of the **2021 Dr. Eduard Martin Prize**.
- **Jan-Oliver Kaiser** (Sep. 2014–May 2020), on leave at BedRock Systems.
- **Scott Kilpatrick** (Aug. 2010–Apr. 2016) → Two Sigma.
 - PhD thesis (2019): *Non-Reformist Reform for Haskell Modularity*.
- **Georg Neis** (Nov. 2008–Aug. 2015) → Google Munich.
 - PhD thesis (2018): *Compositional Compiler Correctness via Parametric Simulations*.
- **Michael Sammler** (since May 2019, co-advised by Deepak Garg).
- **Simon Spies** (since Jul. 2020).
- **David Swasey** (Sep. 2012–May 2020, co-advised by Deepak Garg), on leave at BedRock Systems.
- **Joshua Yanovski** (Jul. 2017–Oct. 2020).
- **Fengmin (Paul) Zhu** (since Nov. 2020).
- **Beta Ziliani** (Jan. 2010–Feb. 2015) → Universidad Nacional de Cordoba → Crystal.
 - PhD thesis (2015): *Interactive Typed Tactic Programming in the Coq Proof Assistant*.

Interns at MPI-SWS:

- **Yixuan Chen**, Yale University (Summer 2021).
- **Shabnam Ghasemirad**, ETH Zürich (Summer 2019).
- **Jeehoon Kang**, Seoul National University (Fall 2015).
- **Lisa Kohl**, Karlsruhe Institute of Technology (Summer 2013).
- **Yusuke Matsushita**, University of Tokyo (Fall 2020, Spring 2021).
- **Gaurav Parthasarathy**, ETH Zürich (Summer 2018).
- **George Pîrlea**, University College London (Summer 2019).
- **Piotr Polesiuk**, University of Wrocław (Fall 2014).
- **Marianna Rapoport**, University of Waterloo (Summer 2018).
- **Milijana Surbatovich**, Carnegie Mellon University (Spring 2022).
- **Joseph Tassarotti**, Carnegie Mellon University (Summer 2014).
- **Ignacio Tiraboschi**, Universidad Nacional de Córdoba (Fall 2018).
- **Irene Yoon**, University of Pennsylvania (Summer 2021).
- **Zhen Zhang**, University of Science & Technology of China (Summer 2016).

Publications

All of my papers are accessible from my web site: <http://www.mpi-sws.org/~dreyer/research.html>.

Compass: Strong and Compositional Library Specifications in Relaxed Memory Separation Logic.

Hoang-Hai Dang, Jaehwang Jung, Jaemin Choi, Duc-Thuan Nguyen,
William Mansky, Jeehoon Kang, Derek Dreyer.

In 2022 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2022).

RustHornBelt: A Semantic Foundation for Functional Verification of Rust Programs with Unsafe Code.

Yusuke Matsushita, Xavier Denis, Jacques-Henri Jourdan, Derek Dreyer.

In 2022 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2022).

Recipient of **PLDI 2022 Distinguished Paper Award**.

Islaris: Verification of Machine Code Against Authoritative ISA Semantics.

Michael Sammler, Angus Hammond, Rodolphe Lepigre, Brian Campbell,
Jean Pichon-Pharabod, Derek Dreyer, Deepak Garg, Peter Sewell.

In 2022 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2022).

Finding Real Bugs in Big Programs with Incorrectness Logic.

Quang Loc Le, Azalea Raad, Jules Villard, Josh Berdine, Derek Dreyer, Peter W. O’Hearn.

In 2022 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2022).

Simuliris: A Separation Logic Framework for Verifying Concurrent Program Optimizations.

Lennard Gäher, Michael Sammler, Simon Spies, Ralf Jung,
Hoang-Hai Dang, Robbert Krebbers, Jeehoon Kang, Derek Dreyer.

In 2022 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2022).

Official citation: PACMPL 6, POPL, Article 28, January 2022.

Recipient of **POPL 2022 Distinguished Paper Award**.

VIP: Verifying Real-World C Idioms with Integer-Pointer Casts.

Rodolphe Lepigre, Michael Sammler, Kayvan Memarian, Robbert Krebbers, Derek Dreyer, Peter Sewell.

In 2022 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2022).

Official citation: PACMPL 6, POPL, Article 20, January 2022.

Concurrent Incorrectness Separation Logic.

Azalea Raad, Josh Berdine, Derek Dreyer, Peter W. O’Hearn.

In 2022 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2022).

Official citation: PACMPL 6, POPL, Article 34, January 2022.

GhostCell: Separating Permissions from Data in Rust.

Joshua Yanovski, Hoang-Hai Dang, Ralf Jung, Derek Dreyer.

In 2021 ACM SIGPLAN International Conference on Functional Programming (ICFP 2021).

Official citation: PACMPL 5, ICFP, Article 92, August 2021.

RefinedC: Automating the Foundational Verification of C Code with Refined Ownership Types.

Michael Sammler, Rodolphe Lepigre, Robbert Krebbers, Kayvan Memarian, Derek Dreyer, Deepak Garg.

In 2021 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2021).

Recipient of **PLDI 2021 Distinguished Paper Award** and **PLDI 2021 Distinguished Artifact Award**.

Transfinite Iris: Resolving an Existential Dilemma of Step-Indexed Separation Logic.

Simon Spies, Lennard Gäher, Daniel Gratzer, Joseph Tassarotti, Robbert Krebbers, Derek Dreyer, Lars Birkedal.

In 2021 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2021).

Safe Systems Programming in Rust.

Ralf Jung, Jacques-Henri Jourdan, Robbert Krebbers, Derek Dreyer.
Invited contributed article, Communications of the ACM (CACM), 64(4): 144–152, April 2021.

Transfinite Step-Indexing for Termination.

Simon Spies, Neel Krishnaswami, Derek Dreyer.
In 2021 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2021).
Official citation: *PACMPL 5, POPL, Article 13, January 2021.*

Local Reasoning About the Presence of Bugs: Incorrectness Separation Logic.

Azalea Raad, Josh Berdine, Hoang-Hai Dang, Derek Dreyer, Peter O’Hearn, Jules Villard.
In 2020 International Conference on Computer-Aided Verification (CAV 2020).

RustBelt Meets Relaxed Memory.

Hoang-Hai Dang, Jacques-Henri Jourdan, Jan-Oliver Kaiser, Derek Dreyer.
In 2020 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2020).
Official citation: *PACMPL 4, POPL, Article 34, January 2020.*

Stacked Borrows: An Aliasing Model for Rust.

Ralf Jung, Hoang-Hai Dang, Jeehoon Kang, Derek Dreyer.
In 2020 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2020).
Official citation: *PACMPL 4, POPL, Article 41, January 2020.*

The Future is Ours: Prophecy Variables in Separation Logic.

Ralf Jung, Rodolphe Lepigre, Gaurav Parthasarathy,
Marianna Rapoport, Amin Timany, Derek Dreyer, Bart Jacobs.
In 2020 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2020).
Official citation: *PACMPL 4, POPL, Article 45, January 2020.*

The High-Level Benefits of Low-Level Sandboxing.

Michael Sammler, Deepak Garg, Derek Dreyer, Tadeusz Litak.
In 2020 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2020).
Official citation: *PACMPL 4, POPL, Article 32, January 2020.*

Iris from the Ground Up: A Modular Foundation for Higher-Order Concurrent Separation Logic.

Ralf Jung, Robbert Krebbers, Jacques-Henri Jourdan, Aleš Bizjak, Lars Birkedal, Derek Dreyer.
Journal of Functional Programming (JFP), 28, e20, November 2018.
Special issue devoted to extended versions of selected papers from ICFP 2016.
This is a significantly revised and expanded synthesis of our ICFP 2016 and ESOP 2017 papers.

MoSeL: A General, Extensible Modal Framework for Interactive Proofs in Separation Logic.

Robbert Krebbers, Jacques-Henri Jourdan, Ralf Jung, Joseph Tassarotti,
Jan-Oliver Kaiser, Amin Timany, Arthur Charguéraud, Derek Dreyer.
In 2018 ACM SIGPLAN International Conference on Functional Programming (ICFP 2018).
Official citation: *PACMPL 2, ICFP, Article 77, September 2018.*

Mtacs2: Typed Tactics for Backward Reasoning in Coq.

Jan-Oliver Kaiser, Beta Ziliani, Robbert Krebbers, Yann-Régis Ganas, Derek Dreyer.
In 2018 ACM SIGPLAN International Conference on Functional Programming (ICFP 2018).
Official citation: *PACMPL 2, ICFP, Article 78, September 2018.*

RustBelt: Securing the Foundations of the Rust Programming Language.

Ralf Jung, Jacques-Henri Jourdan, Robbert Krebbers, Derek Dreyer.
In 2018 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2018).
Official citation: *PACMPL 2, POPL, Article 66, January 2018.*

Robust and Compositional Verification of Object Capability Patterns.

David Swasey, Deepak Garg, Derek Dreyer.

In 2017 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2017).

Official citation: *PACMPL 1, OOPSLA, Article 89, October 2017.*

Recipient of **OOPSLA 2017 Distinguished Paper Award.**

Strong Logic for Weak Memory: Reasoning About Release-Acquire Consistency in Iris.

Jan-Oliver Kaiser, Hoang-Hai Dang, Derek Dreyer, Ori Lahav, Viktor Vafeiadis.

In 2017 European Conference on Object-Oriented Programming (ECOOP 2017).

Recipient of **ECOOP 2017 Distinguished Paper Award.**

Repairing Sequential Consistency in C/C++11.

Ori Lahav, Viktor Vafeiadis, Jeehoon Kang, Chung-Kil Hur, Derek Dreyer.

In 2017 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2017).

Recipient of **PLDI 2017 Distinguished Paper Award.**

The Essence of Higher-Order Concurrent Separation Logic.

Robbert Krebbers, Ralf Jung, Aleš Bizjak, Jacques-Henri Jourdan, Derek Dreyer, Lars Birkedal.

In 2017 European Symposium on Programming (ESOP 2017).

A Promising Semantics for Relaxed-Memory Concurrency.

Jeehoon Kang, Chung-Kil Hur, Ori Lahav, Viktor Vafeiadis, Derek Dreyer.

In 2017 ACM SIGPLAN Symposium on Principles of Programming Languages (POPL 2017).

Higher-Order Ghost State.

Ralf Jung, Robbert Krebbers, Lars Birkedal, Derek Dreyer.

In 2016 ACM SIGPLAN International Conference on Functional Programming (ICFP 2016).

Lightweight Verification of Separate Compilation.

Jeehoon Kang, Yoonseung Kim, Chung-Kil Hur, Derek Dreyer, Viktor Vafeiadis.

In 2016 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2016).

Pilsner: A Compositionally Verified Compiler for a Higher-Order Imperative Language.

Georg Neis, Chung-Kil Hur, Jan-Oliver Kaiser, Craig McLaughlin, Derek Dreyer, Viktor Vafeiadis.

In 2015 ACM SIGPLAN International Conference on Functional Programming (ICFP 2015).

Mtac: A Monad for Typed Tactic Programming in Coq.

Beta Ziliani, Derek Dreyer, Neelakantan R. Krishnaswami, Aleksandar Nanevski, Viktor Vafeiadis.

Journal of Functional Programming (JFP), 25, e12, July 2015.

Special issue devoted to extended versions of selected papers from ICFP 2013.

Verifying Read-Copy-Update in a Logic for Weak Memory.

Joseph Tassarotti, Derek Dreyer, Viktor Vafeiadis.

In 2015 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2015).

Iris: Monoids and Invariants as an Orthogonal Basis for Concurrent Reasoning.

Ralf Jung, David Swasey, Filip Sieczkowski, Kasper Svendsen, Aaron Turon, Lars Birkedal, Derek Dreyer.

In 2015 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2015).

GPS: Navigating Weak Memory with Ghosts, Protocols, and Separation.

Aaron Turon, Viktor Vafeiadis, Derek Dreyer.

In 2014 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2014).

F-ing Modules.

Andreas Rossberg, Claudio Russo, Derek Dreyer.
Journal of Functional Programming (JFP), 24(5): 529–607, September 2014.
This is a significantly revised and expanded version of our TLDI 2010 paper.

Backpack: Retrofitting Haskell with Interfaces.

Scott Kilpatrick, Derek Dreyer, Simon Peyton Jones, Simon Marlow.
In 2014 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2014).

Mtac: A Monad for Typed Tactic Programming in Coq.

Beta Ziliani, Derek Dreyer, Neelakantan R. Krishnaswami, Aleksandar Nanevski, Viktor Vafeiadis.
In 2013 ACM SIGPLAN International Conference on Functional Programming (ICFP 2013).

Unifying Refinement and Hoare-Style Reasoning in a Logic for Higher-Order Concurrency.

Aaron Turon, Derek Dreyer, Lars Birkedal.
In 2013 ACM SIGPLAN International Conference on Functional Programming (ICFP 2013).

Internalizing Relational Parametricity in the Extensional Calculus of Constructions.

Neelakantan R. Krishnaswami, Derek Dreyer.
In 2013 EACSL Annual Conference on Computer Science Logic (CSL 2013).

How to Make Ad Hoc Proof Automation Less Ad Hoc.

Georges Gonthier, Beta Ziliani, Aleksandar Nanevski, Derek Dreyer.
Journal of Functional Programming (JFP), 23(4): 357–401, July 2013.
Special issue devoted to extended versions of selected papers from ICFP 2011.

Mixin' Up the ML Module System.

Andreas Rossberg, Derek Dreyer.
ACM Transactions on Programming Languages and Systems (TOPLAS), 35(1), Article 2, April 2013.
This is a significantly revised, corrected, and expanded version of our ICFP 2008 paper.

Logical Relations for Fine-Grained Concurrency.

Aaron Turon, Jacob Thamsborg, Amal Ahmed, Lars Birkedal, Derek Dreyer.
In 2013 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2013).

The Power of Parameterization in Coinductive Proof.

Chung-Kil Hur, Georg Neis, Derek Dreyer, Viktor Vafeiadis.
In 2013 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2013).

Superficially Substructural Types.

Neelakantan R. Krishnaswami, Aaron Turon, Derek Dreyer, Deepak Garg.
In 2012 ACM SIGPLAN International Conference on Functional Programming (ICFP 2012).

The Impact of Higher-Order State and Control Effects on Local Relational Reasoning.

Derek Dreyer, Georg Neis, Lars Birkedal.
Journal of Functional Programming (JFP), 22(4&5): 477–528, September 2012.
Special issue devoted to extended versions of selected papers from ICFP 2010.

The Marriage of Bisimulations and Kripke Logical Relations.

Chung-Kil Hur, Derek Dreyer, Georg Neis, Viktor Vafeiadis.
In 2012 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2012).

How to Make Ad Hoc Proof Automation Less Ad Hoc.

Georges Gonthier, Beta Ziliani, Aleksandar Nanevski, Derek Dreyer.
In 2011 ACM SIGPLAN International Conference on Functional Programming (ICFP 2011).

Non-Parametric Parametricity.

Georg Neis, Derek Dreyer, Andreas Rossberg.
Journal of Functional Programming (JFP), 21(4&5): 497–562, September 2011.
Special issue devoted to extended versions of selected papers from ICFP 2009.

Separation Logic in the Presence of Garbage Collection.

Chung-Kil Hur, Derek Dreyer, Viktor Vafeiadis.
In 2011 IEEE Symposium on Logic in Computer Science (LICS 2011).

Logical Step-Indexed Logical Relations.

Derek Dreyer, Amal Ahmed, Lars Birkedal.
Logical Methods in Computer Science (LMCS), 7(2:16): 1–37, June 2011.
Special issue devoted to extended versions of selected papers from LICS 2009.

A Kripke Logical Relation Between ML and Assembly.

Chung-Kil Hur, Derek Dreyer.
In 2011 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2011).

The Impact of Higher-Order State and Control Effects on Local Relational Reasoning.

Derek Dreyer, Georg Neis, Lars Birkedal.
In 2010 ACM SIGPLAN International Conference on Functional Programming (ICFP 2010).
This paper was nominated by ACM SIGPLAN for a **CACM Research Highlight**.

F-ing Modules.

Andreas Rossberg, Claudio V. Russo, Derek Dreyer.
In 2010 ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI 2010).

A Relational Modal Logic for Higher-Order Stateful ADTs.

Derek Dreyer, Georg Neis, Andreas Rossberg, Lars Birkedal.
In 2010 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2010).

Non-Parametric Parametricity.

Georg Neis, Derek Dreyer, Andreas Rossberg.
In 2009 ACM SIGPLAN International Conference on Functional Programming (ICFP 2009).

Logical Step-Indexed Logical Relations.

Derek Dreyer, Amal Ahmed, Lars Birkedal.
In 2009 IEEE Symposium on Logic in Computer Science (LICS 2009).

State-Dependent Representation Independence.

Amal Ahmed, Derek Dreyer, Andreas Rossberg.
In 2009 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2009).

Mixin' Up the ML Module System.

Derek Dreyer, Andreas Rossberg.
In 2008 ACM SIGPLAN International Conference on Functional Programming (ICFP 2008).

A Type System for Recursive Modules.

Derek Dreyer.
In 2007 ACM SIGPLAN International Conference on Functional Programming (ICFP 2007).

Principal Type Schemes for Modular Programs.

Derek Dreyer, Matthias Blume.
In 2007 European Symposium on Programming (ESOP 2007).

Recursive Type Generativity.

Derek Dreyer.

Journal of Functional Programming (JFP), 17(4&5): 433-471, July & September 2007.

Special issue devoted to extended versions of selected papers from ICFP 2005.

Modular Type Classes.

Derek Dreyer, Robert Harper, Manuel M.T. Chakravarty.

In 2007 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2007).

Recursive Type Generativity.

Derek Dreyer.

In 2005 ACM SIGPLAN International Conference on Functional Programming (ICFP 2005).

Understanding and Evolving the ML Module System.

Derek Dreyer.

Ph.D. Thesis, Carnegie Mellon University Technical Report CMU-CS-05-131, May 2005.

A Type System for Well-Founded Recursion.

Derek Dreyer.

In 2004 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2004).

A Type System for Higher-Order Modules.

Derek Dreyer, Karl Cray, Robert Harper.

In 2003 ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2003).

Typed Compilation of Recursive Datatypes.

Joseph C. Vanderwaart, Derek Dreyer, Leaf Petersen, Karl Cray, Robert Harper, Perry Cheng.

In 2003 ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI 2003).

Teaching Experience

Max Planck Institute for Software Systems (MPI-SWS) / Saarland University

Instructor and course designer

Core graduate/undergraduate course: Semantics.

Recipient of the **“Busy Beaver Award” for Outstanding Commitment to Teaching.**

Course home page: https://cms.sic.saarland/semantics_ws2122/.

Developed and taught a core course on programming language semantics. The first half of the course studies type systems, operational semantics, and semantic models based on logical relations, including how to mechanize these developments in the Coq proof assistant. The second half of the course explores program logics, in particular separation logic, and how to use it to formalize semantic models of type systems in a higher-level way using the Iris framework in Coq.

Winter 2021–22

Co-instructor and course designer

Core graduate/undergraduate course: Semantics.

Course home page: https://courses.ps.uni-saarland.de/sem_ws1920/.

Co-taught a core course on programming language semantics with Prof. Gert Smolka of Saarland University. My half of the course focused on the use of logical-relations models of types to reason about local invariants, semantic safety, and representation independence.

Winter 2019–20

Winter 2017–18

Winter 2015–16

Instructor and course designer

Graduate course: Categorical Logic.

Led an advanced graduate seminar on basic category theory and applications to building models of higher-order separation logic.

Winter 2014–15

Instructor and course designer

Winter 2012–13

Graduate course: Parametricity and Modular Reasoning.

Course home page: <https://wiki.mpi-sws.org/star/paramore>.

Led an advanced graduate seminar on the theory of parametricity, focusing on the use of logical-relations techniques for modular reasoning about a wide variety of semantically complex programming language features.

Co-instructor and course designer

Summer 2011

Graduate course: Concurrent Program Logics.

Course home page: <https://wiki.mpi-sws.org/star/cpl>.

Led an advanced graduate seminar, together with Viktor Vafeiadis, on Hoare-style logics for concurrent shared-memory programs.

Instructor and course designer

Winter 2010–11

Graduate course: Type Systems for Modules.

Course home page: <http://www.mpi-sws.org/~skilpat/modsem/>.

Led an advanced graduate seminar on type systems for modular programming, focusing on the design and evolution of the ML module system.

Instructor and course designer

Winter 2008–09

Graduate course: Typed Operational Reasoning.

Course home page: <http://www.mpi-sws.org/~dreyer/tor/>.

Taught a variant of the *Advanced Type Systems* graduate course that I had previously given in Winter 2006 at the University of Chicago (see below).

University of Chicago

Instructor and course designer

Winter 2006

Graduate course: Advanced Type Systems.

Course home page: <http://tti-c.org/dreyer/course/>.

Designed an original seminar course, in which the students learned how to apply the method of *logical relations* to prove a range of different theorems about program semantics (*e.g.*, strong normalization, decidability of typechecking, parametricity properties, and program equivalence).