

Curriculum Vitae

Derek Dreyer

Personal Information

Affiliation:	Prof. Dr. Derek Dreyer Tenured faculty, MPI-SWS Honorarprofessor, Saarland University	Citizenship:	U.S.A.
		Birthdate:	May 2, 1980
		E-mail:	dreyer@mpi-sws.org
Address:	MPI-SWS, Campus E1.5 66123 Saarbruecken 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.

- Honorarprofessor, Saarland University. 2017–present
- Tenured faculty. 2013–present
- Tenure-track faculty. 2008–2013

Toyota Technological Institute at Chicago (TTI-C)

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

Awards, Grants, and Fellowships

Max Planck Institute for Software Systems (MPI-SWS)

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 euros) 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
Finalist, Microsoft Research Faculty Fellowship	2011

Professional Activities

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–present.

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 Bob 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:

- 2019 European Conference on Object-Oriented Programming (**ECOOP 2019**).
- 2018 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications (**OOPSLA 2018**).
Recipient of **OOPSLA’18 Distinguished Reviewer Award**.
- 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:

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

Invited speaker:

- 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@POPL 2020, PLMW@SPLASH 2019, PLMW@ICFP 2019, PLMW@POPL 2018, PLMW@ICFP 2017, PLMW@POPL 2017, PLMW@ICFP 2016, PLMW@POPL 2016, PLMW@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:

- **Chung-Kil Hur** (Oct. 2010–Sep. 2012) → Microsoft Research Cambridge → Seoul National University.
- **Jacques-Henri Jourdan** (Apr. 2016–Sep. 2017) → CNRS, LRI, Université Paris Saclay.
- **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** (since Jan. 2019).
- **Pierre-Marie Pédrot** (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–April 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** (since Apr. 2016).
- **Ralf Jung** (since Sep. 2014, PhD expected 2020) → MIT (postdoc).
- **Jan-Oliver Kaiser** (since Sep. 2014).
- **Scott Kilpatrick** (Aug. 2010–Apr. 2016).
 - 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).
- **David Swasey** (since Sep. 2012, PhD expected 2020, co-advised by Deepak Garg).
- **Joshua Yanovski** (since July 2017).
- **Beta Ziliani** (Jan. 2010–Feb. 2015) → Universidad Nacional de Cordoba.
 - PhD thesis (2015): *Interactive Typed Tactic Programming in the Coq Proof Assistant*.

Interns at MPI-SWS:

- **Jeehoon Kang**, Seoul National University (Fall 2015).
- **Lisa Kohl**, Karlsruhe Institute of Technology (Summer 2013).
- **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).
- **Joseph Tassarotti**, Carnegie Mellon University (Summer 2014).
- **Ignacio Tiraboschi**, Universidad Nacional de Córdoba (Fall 2018).
- **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>.

Note: Since August 2017, ACM SIGPLAN has published the proceedings of POPL, ICFP, and OOPSLA as special issues of the newly-formed journal PACMPL (Proceedings of the ACM on Programming Languages). The rationale behind the creation of PACMPL is that conference publications in programming languages are of the same quality—and reviewed with at least as much diligence—as journal publications in other scientific disciplines, and as such should be counted as journal publications. Although I concur with this rationale (and in fact supported the transition to PACMPL when I served on the ACM SIGPLAN Executive Committee), there is nonetheless a qualitative difference between PACMPL publications and traditional long-form journal publications in programming languages: PACMPL publications must obey strict page limits, and the PACMPL reviewing process is tightly constrained by the demands of conference reviewing. For this reason, I list my PACMPL publications below together with my conference publications, even though formally they are journal publications.

Long-Form Journal Publications

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 archival versions of selected papers from ICFP 2016.

This is a significantly revised and expanded synthesis of our ICFP 2016 and ESOP 2017 papers.

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 archival versions of selected papers from ICFP 2013.

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.

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 archival 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.

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 archival versions of selected papers from ICFP 2010.

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 archival versions of selected papers from ICFP 2009.

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 archival versions of selected papers from LICS 2009.

Recursive Type Generativity.

Derek Dreyer.
Journal of Functional Programming (JFP), 17(4&5): 433-471, July & September 2007.
Special issue devoted to archival versions of selected papers from ICFP 2005.

PACMPL, CACM, Conference, and Workshop Publications

Safe Systems Programming in Rust: The Promise and the Challenge.

Ralf Jung, Jacques-Henri Jourdan, Robbert Krebbers, Derek Dreyer.
Invited article, to appear in Communications of the CACM (CACM), 2020.

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*.

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*.

Mtac2: Typed Tactics for Backward Reasoning in Coq.

Jan-Oliver Kaiser, Beta Ziliani, Robbert Krebbers, Yann-Régis Gianas, 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).

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).

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).

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 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).

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).

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).

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).

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 Crary, 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 Crary, Robert Harper, Perry Cheng.

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

Ph.D. Thesis

Understanding and Evolving the ML Module System.

Derek Dreyer.

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

Teaching Experience

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

Co-instructor and course designer

Core graduate/undergraduate course: Semantics.

Course home page: https://courses.ps.uni-saarland.de/sem_ws1718/.

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.

Winter 2014–15

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

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).