

ADAM CHLIPALA

Cambridge, MA

USA

adam@chlipala.net

<http://adam.chlipala.net/>

Research interests

- Engineering issues in interactive computer theorem proving, with a special focus on compiler verification and the metatheory of programming languages in general
- Dependent type systems and other type systems with versatile facilities for type-level computation
- Design and implementation of functional programming languages, especially domain-specific languages
- Statically-typed metaprogramming

Education

- **University of California, Berkeley**
Electrical Engineering and Computer Science Department
Computer Science Division
Doctor of Philosophy (PhD) in Computer Science
8/2003 – 9/2007
Advisor: George Necula
Cumulative GPA: 4.0 out of 4.0
Thesis: Implementing Certified Programming Language Tools in Dependent Type Theory
- **University of California, Berkeley**
Electrical Engineering and Computer Science Department
Computer Science Division
Master of Science (MS) in Computer Science
12/2004
Advisor: George Necula
Thesis: An Untrusted Verifier for Typed Assembly Language
- **Carnegie Mellon University, Pittsburgh, PA**
Bachelor of Science (BS) in Computer Science with a minor in Mathematical Sciences and University Honors
8/2000 – 5/2003
Cumulative GPA: 4.0 out of 4.0
- **Emmaus High School, Emmaus, PA**
High school diploma
9/1996 – 6/2000

Refereed conference papers

- Adam Chlipala. **Ur: Statically-Typed Metaprogramming with Type-Level Record Computation**. Proceedings of the ACM SIGPLAN 2010 Conference on Programming Language Design and Implementation (PLDI'10). June 2010.

- Adam Chlipala. **A Verified Compiler for an Impure Functional Language**. Proceedings of the 37th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'10). January 2010.
- Adam Chlipala, Gregory Malecha, Greg Morrisett, Avraham Shinnar, Ryan Wisnesky. **Effective Interactive Proofs for Higher-Order Imperative Programs**. Proceedings of the 14th ACM SIGPLAN International Conference on Functional Programming (ICFP'09). August 2009.
- Adam Chlipala. **Parametric Higher-Order Abstract Syntax for Mechanized Semantics**. Proceedings of the 13th ACM SIGPLAN International Conference on Functional Programming (ICFP'08). September 2008.
- Adam Chlipala. **A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language**. Proceedings of the ACM SIGPLAN 2007 Conference on Programming Language Design and Implementation (PLDI'07). June 2007.
- Adam Chlipala. **Modular Development of Certified Program Verifiers with a Proof Assistant**. Proceedings of the 11th ACM SIGPLAN International Conference on Functional Programming (ICFP'06). September 2006.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula. **A Framework for Certified Program Analysis and Its Applications to Mobile-Code Safety**. Proceedings of the 7th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI'06). January 2006.
- Dirk Beyer, Adam Chlipala, Thomas Henzinger, Ranjit Jhala, Rupak Majumdar. **Generating Tests from Counterexamples**. Proceedings of the 26th International Conference on Software Engineering (ICSE'04), IEEE Computer Society Press. May 2004.

Refereed journal articles

- Adam Chlipala. **Modular Development of Certified Program Verifiers with a Proof Assistant**. Journal of Functional Programming (JFP). 18(5/6). 599-647, 2008. Cambridge University Press.

Refereed workshop papers

- Adam Chlipala. **Position Paper: Thoughts on Programming with Proof Assistants**. Proceedings of the Programming Languages meets Program Verification Workshop (PLPV'06). August 2006.
- Adam Chlipala, George C. Necula. **Cooperative Integration of an Interactive Proof Assistant and an Automated Prover**. Proceedings of the 6th International Workshop on Strategies in Automated Deduction (STRATEGIES'06). August 2006.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula, Robert R. Schneck. **The Open Verifier Framework for Foundational Verifiers**. Proceedings of the 2nd ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'05). January 2005.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula, Robert R. Schneck. **Type-Based Verification of Assembly Language for Compiler Debugging**. Proceedings of the 2nd ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'05). January 2005.

- Adam Chlipala, Leaf Petersen, Robert Harper. **Strict Bidirectional Type Checking**. Proceedings of the 2nd ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'05). January 2005.

Refereed poster sessions

- Adam Chlipala. **Developing Certified Program Verifiers with a Proof Assistant**. Proceedings of the International Workshop on Proof-Carrying Code (PCC'06). August 2006.

Invited conference papers

- Dirk Beyer, Adam Chlipala, Thomas Henzinger, Ranjit Jhala, Rupak Majumdar. **The Blast Query Language for Software Verification**. Proceedings of the 11th Static Analysis Symposium (SAS'04), Lecture Notes in Computer Science 3148, Springer-Verlag. August 2004.

Technical reports

- Adam Chlipala. **Generic Programming and Proving for Programming Language Metatheory**. Technical Report UCB/EECS-2007-147. 2007.
- Adam Chlipala. **Implementing Certified Programming Language Tools in Dependent Type Theory**. Technical Report UCB/EECS-2007-113. 2007.
- Adam Chlipala. **Scrap Your Web Application Boilerplate, or Metaprogramming with Row Types**. Technical Report UCB/EECS-2006-120. 2006.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula. **A Framework for Certified Program Analysis and Its Applications to Mobile-Code Safety**. Technical Report UCB/ERL M05/32. UC Berkeley EECS Department. 2005.
- Adam Chlipala. **An Untrusted Verifier for Typed Assembly Language**. MS Project Report. Technical Report UCB/ERL M04/41. UC Berkeley EECS Department. 2004.

Talks

- "A Sane Approach to Modern Web Application Development". Boston Lisp. February 2010.
- "A Verified Compiler for an Impure Functional Language". POPL'10. January 2010.
- "Towards the Ultimate Web Application Framework, via Fancy Types". New England F# User Group. November 2009.
- "Syntactic Proofs of Compositional Compiler Correctness". NJPLS. October 2009.
- "Metaprogramming AJAX Apps with Static Types". DEFUN'09. September 2009.
- "Engineering a Verified Functional Language Compiler" (invited talk). WMM'09. September 2009.
- "Effective Interactive Proofs for Higher-Order Imperative Programs". ICFP'09. August 2009.
- "Metaprogramming AJAX Apps with Static Types". Microsoft Research Redmond. July 2009.

- “Liberating Semi-Automated PL Proofs from Binder Bookkeeping”. Northeastern University Programming Languages Seminar. February 2009.
- “Liberating Semi-Automated PL Proofs from Binder Bookkeeping”. Boston University Programming Languages Reading Group. February 2009.
- “Statically-Checked Metaprogramming for Web Applications”. NEPLS 21. November 2008.
- “Parametric Higher-Order Abstract Syntax for Mechanized Semantics”. ICFP’08. September 2008.
- “Generic Programming and Proving for Programming Language Metatheory”. WMM’07. October 2007.
- “A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language”. PLDI’07. June 2007.
- “A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language”. Open Source Quality Project Retreat. May 2007.
- “A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language”. ProjeT Gallium seminar. January 2007.
- “Modular Development of Certified Program Verifiers with a Proof Assistant”. ICFP’06. September 2006.
- “Position Paper: Thoughts on Programming with Proof Assistants”. PLPV’06. August 2006.
- “Cooperative Integration of an Interactive Proof Assistant and an Automated Prover”. STRATEGIES’06. August 2006.
- “Developing Sound Program Analysis Tools by Programming with Proofs”. Open Source Quality Project Retreat. May 2006.
- “A Framework for Certified Program Analysis and Its Applications to Mobile-Code Safety”. VMCAI’06. January 2006.
- “Proof-Carrying Verifiers”. Open Source Quality Project Retreat. May 2005.
- “The Open Verifier Framework for Foundational Verifiers”. TLDI’05. January 2005.

Professional service

- 5th International Workshop on Systems Software Verification (SSV’10), program committee
- Mathematically Structured Functional Programming 2010 (MSFP’10), program committee
- The Coq Workshop 2010 (Coq’10), program committee
- Programming Languages meets Program Verification Workshop (PLPV’10), program committee
- 4th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice (LFMTP’09), program committee
- 3rd Informal ACM SIGPLAN Workshop on Mechanizing Metatheory (WMM’08), program committee
- External reviewer for: ICFP’04, LPAR’05, LICS’06, APLAS’06, TLDI’07, RTA’07, POPL’08, VMCAI’08, PLDI’08, ICFP’08, POPL’09, TLDI’09, ESOP’09, TYPES’08, PLDI’09, ICFP’09, POPL’10, FoSSaCS’10, TACAS’10, MFPS’10

- Referee for: ESL, IPL, JAR, TOPLAS

Employment

- **Postdoctoral Fellow**
School of Engineering and Applied Sciences
Harvard University, Cambridge, MA
6/2008 – ??
Advisor: Greg Morrisett
- **Instructor**
COMPSCI 252: Certified Programming with Dependent Types
School of Engineering and Applied Sciences
Harvard University, Cambridge, MA
9/2008 – 1/2009
- **OCaml Hacker**
Jane Street Capital
9/2007 – 4/2008
- **Graduate Student Researcher**
The Open Verifier project
Computer Science Division
University of California, Berkeley
9/2003 – 8/2007
PI: George Necula
- **Instructor**
CS294-9: Interactive Computer Theorem Proving
Computer Science Division
University of California, Berkeley
8/2006 – 12/2006
- **Research Intern**
The Singularity project
Software Productivity Tools group, Redmond, WA
Microsoft Research
6/2005 – 8/2005
Mentor: Manuel Fahndrich
- **Graduate Student Instructor**
CS172: Computability and Complexity
Computer Science Division
University of California, Berkeley
1/2005 – 5/2005
Instructor: Brian Lucena
- **Graduate Student Researcher**
The BLAST project
Computer Science Division
University of California, Berkeley
6/2003 – 8/2003
PI: Thomas Henzinger

- **Research Assistant**
The TILT type-directed Standard ML compiler project
Computer Science Department
Carnegie Mellon University, Pittsburgh, PA
6/2002 – 5/2003
PIs: Robert Harper, Karl Crary
- **Teaching Assistant**
15-212: Principles of Programming (introduction to formal reasoning about programs and functional programming with Standard ML)
Computer Science Department
Carnegie Mellon University, Pittsburgh, PA
1/2002 – 5/2002
Instructors: Michael Erdmann, Jeannette Wing
- **Intern/Software Developer**
Avaya Communication, Holmdel, NJ
6/2001 – 8/2001
- **Software Developer**
Trifecta Technologies, Allentown, PA
Summers, 1998 - 2000

Academic honors

- **National Defense Science and Engineering Graduate Fellowship** winner, 2004
- **National Science Foundation Graduate Research Fellowship** winner, 2004
- **California Microelectronics Fellowship** winner, UC Berkeley EECS Department, 8/2003 – 5/2004
- Inducted into **Phi Kappa Phi**
- Inducted into **Phi Beta Kappa**
- Honorable Mention, **National Science Foundation Graduate Research Fellowship** competition, 2003
- **Andrew Carnegie Scholarship** winner, Carnegie Mellon University, Pittsburgh, PA, 8/2000 – 5/2003

Software technologies

- **I have expert-level experience with and have hacked on implementations of:** ML, Coq, C
- **I've written significant amounts of code in:** F#, Java, SQL, x86 and Z80 assembly languages
- **I'm conversant in:** Haskell, C++, XSLT
- **I have some familiarity with:** Twelf, Scheme, Common Lisp, Prolog, C#, Visual Basic, UNIX shell scripting, Perl
- **I've done system administration of these daemons on UNIX systems with at least 100 users:** Apache, djbdns, Courier IMAP, Exim, Mailman, SpamAssassin

Citizenship

- American citizen

Summer schools

- **Summer School on Software Security: Theory to Practice**, University of Oregon, 6/2004

Software

- **Ur/Web** (<http://www.impredicative.com/ur/>), a prototype domain-specific programming language design and implementation supporting metaprogramming of web applications with strong static guarantees
- **Cooperative Internet hosting tools** (<http://hcoop.sourceforge.net/>), including **DomTool** (<http://wiki.hcoop.net/DomTool>), a domain-specific language in support of shared UNIX system configuration by mutually-untrusting users
- **Dynamic web site tools for Standard ML** (<http://smlweb.sourceforge.net/>), including separately usable libraries for accessing SQL databases

Other activities

- Founder, president, and chief software developer of **HCoop, Inc.** (<http://hcoop.net/>), a democratically-run Internet hosting cooperative
- Black belt in Karate (no longer training)