

Dr. Eric Schulte

Pittsburgh, PA

📞 (703) 405 0263 • ✉ schulte.eric@gmail.com
🌐 eschulte.github.io • in eric-schulte-255bb613 • 🌐 eschulte

I automate the daily work of software engineers. I have proposed and led large government funded R&D projects, managing scientists and engineers. I partner with small volunteer groups, open-source communities, top CS academics, and large tech companies.

Professional

Google

Pittsburgh, PA

Tech Lead and Software Engineer in Ads

2022 – present

Develop data analysis platforms and tooling for software development task automation.

- Multiple exploratory agentic systems for quality of life improvements across Ads development and release. Including (1) a customized agentic review agent that is now mandatory after demonstrated improvements in code quality, (2) agentic on-call systems which access build, test, and experiment data to investigate, root-cause, and summarize build/release issues, and (3) a novel 20% project providing “beta-level simulation” of Googlers – essentially AI “innies” – conditioned against specific Googler’s individual working notes, code contributions, communication history, and command-line history to answer questions in their stead.
- Improved a Google Ads Formats team’s feature velocity. Reviewed, analyzed, then planned and managed the optimization of their development pipeline. Improvements included improved testing systems to enable earlier on-device testing reducing downstream experiment time and decoupling slow release process reducing time from code submission to deployment.
- Reduced alert volume for Google Ad’s revenue-based alerting system by ~50% significantly reducing developer toil and increasing alert response rates.
- Main developer for Ads Ground Truth a cross-PA unified monitoring evaluation system.
- Reduced test toil using computer vision and structured differencing to group screenshots.

GrammaTech

Ithaca, NY

Director of Automated Software Engineering

2018 – 2022

Support GrammaTech’s research division building up staff, expertise, frameworks, prototypes, funding, and IT infrastructure. Research and develop tools and techniques advancing the boundaries of automated software development and reverse engineering.

- Propose and manage government funded R&D projects in tens of millions of dollars.
- Manage technical team of 20 including scientists, software engineers, and test engineers.
- Lead the Machine Programming and Binary Rewriting research areas:
 - Machine Programming: automated software analysis, transformation, and synthesis.
 - Binary Rewriting: analysis and transformation of compiled software binaries.
- Leverage techniques and technologies to solve research problems:
 - Machine Learning and Evolutionary Computation for automated software development.
 - Formal methods and Logic Programming for efficient enumeration and program analysis.
- Maintain and grow IT infrastructure, improve engineering efficiency.
- Research and develop tools to automate software (reverse) engineering.
 - GTIRB/DDisasm* : IR supporting a binary analysis and rewriting ecosystem with impact across the binary analysis research community.
 - SEL* : Software Evolution Library for source code analysis and transformation.

Senior Scientist 2014 – 2018

- Contributed to the following research and development efforts:
 - Applied formal methods to build provably-secure N-variant systems from COTS binaries.
 - Applied evolutionary techniques to evolve *exact* C decompilation of COTS binaries.
 - Applied evolutionary techniques to automatically repair flaws in COTS binaries.

University of New Mexico **Albuquerque, NM**
Research Assistant 2009 – 2014

- Research and develop evolutionary techniques for software maintenance and improvement.
- Empirical and theoretical investigation of biological properties of software.

Counsyl **Palo Alto, CA**
Open Source Software Consultant 2010 – 2011

- Used open-source software to automate technical document generation.

The MITRE Corporation **McLean, VA**
Senior Artificial Intelligence Engineer 2004 – 2009

- Lead developer of the Rapid Argus Modeling for Biosurveillance Operations (RAMBO) system. RAMBO provided disease modeling and surveillance supporting ~ 50 biosecurity analysts.
- Prototype the STAT (Statistical Tracking and Analysis of Text) temporal text analysis system.
- Systems administration for production Unix/Linux systems.

Volunteer

Planned Parenthood **Pittsburgh, PA**
Clinic Escort 2019 – present

National Poor People's Campaign
Web Designer 2020 – present

GNU Emacs
Contributor 2009 – 2014

- Developed Emacs Org-mode's facilities to embed executable source code into documents.

Education

University of New Mexico **Albuquerque, NM**
Ph.D., Computer Science 2014

Advisor: Stephanie Forrest

Kenyon College **Gambier, OH**
B.A., Mathematics, Minor Philosophy 2004

Skills

Programming languages: Python, JavaScript, C++, Lisp, OCaml, Haskell, Ruby, Prolog

Technologies: Emacs, (Agent Development Kit) ADK, SMT/SAT, CI, \LaTeX , HTML/CSS

Domains: Software Engineering, Machine Learning, Binary Analysis, Formal Methods

Botanical Art: Graphite, Colored Pencil, Water Color, Mylar

Yoga: Vinyasa Krama

Publications

Doctoral Thesis

Eric Schulte. *Neutral Networks of Real-World Programs and their Application to Automated Software Evolution*. PhD thesis, University of New Mexico, Albuquerque, USA, July 2014. <https://cs.unm.edu/~eschulte/dissertation>.

Patent

Eric Michael Schulte and Antonio Enrique Flores Montoya. Systems and/or methods for generating reassemblable disassemblies of binaries using declarative logic, 2020. US010705814B2.

Refereed Conference Publications

Antonio Flores-Montoya and Eric Schulte. Datalog disassembly. In *29th USENIX Security Symposium (USENIX Security 20)*, 2020. Distinguished Paper.

Vineeth Kashyap, Jason Ruchti, Lucja Kot, Emma Turetsky, Rebecca Swords, David Melski, and Eric Schulte. Automated customized bug-benchmark generation. In *2019 19th International Working Conference on Source Code Analysis and Manipulation (SCAM)*, pages 103–114. IEEE, 2019. Distinguished Paper.

Deborah Katz, Jason Ruchti, and Eric Schulte. Using recurrent neural networks for decompilation. In *Software Analysis, Evolution and Reengineering (SANER)*, 2018. IEEE, 2018.

Eric Schulte, Jonathan Dorn, Stephen Harding, Stephanie Forrest, and Westley Weimer. Post-compiler software optimization for reducing energy. In *Proceedings of the eighteenth international conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '14. ACM, 2014, *Acceptance Rate: 22.6%*.

Eric Schulte, Jonathan DiLorenzo, Westley Weimer, and Stephanie Forrest. Automated repair of binary and assembly programs for cooperating embedded devices. In *Proceedings of the eighteenth international conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '13. ACM, 2013, *Acceptance Rate: 22.8%*.

Eric Schulte, Stephanie Forrest, and Westley Weimer. Automated program repair through the evolution of assembly code. In *Proceedings of the IEEE/ACM international conference on Automated software engineering*, ASE '10, pages 313–316, New York, NY, USA, 2010. ACM, *Acceptance Rate: 17.8%*.

Refereed Journal Articles

Eric Schulte, Zachary Fry, Ethan Fast, Westley Weimer, and Stephanie Forrest. Software mutational robustness. *Genetic Programming and Evolvable Machines*, pages 1–32, 2013, *Impact Factor: 1.333*.

Eric Schulte, Dan Davison, Thomas Dye, and Carsten Dominik. A multi-language computing environment for literate programming and reproducible research. *Journal of Statistical Software*, 46(3):1–24, 1 2012, *Impact Factor: 4.910*.

Paul Lehner, Charles Worrell, Chrissy Vu, Janet Mittel, Stephen Snyder, Eric Schulte, and Warren Greiff. An application of document filtering in an operational system. *Information Processing & Management*, 46(5):611–627, 2010.

Magazine Articles

Eric Schulte and Dan Davison. Active document with org-mode. *Computing in Science & Engineering*, 13(3):66–73, May/June 2011, *Impact Factor: 1.72*.

Workshop Papers

Eric Schulte, Suan Yong, and David Melski. Inuring: Live attacker-guided repair. In *Proceedings of the 3rd ACM Workshop on Forming an Ecosystem Around Software Transformation*, pages 39–45, 2019.

Benoit Baudry, Nicolas Harrant, Eric Schulte, Chris Timperley, Shin Hwei Tan, Marija Selakovic, and Emamurho Ugherughe. A spoonful of devops helps the gi go down. 2018.

Eric Schulte, Jason Ruchti, Matt Noonan, David Ciarletta, and Alexey Loginov. Evolving exact decompilation. In *Binary Analysis Research (BAR)*, 2018, 2018.

Vineeth Kashyap, Rebecca Swords, Eric Schulte, and David Melski. Musynth: Program synthesis via code reuse and code manipulation. In *International Symposium on Search Based Software Engineering*, pages 117–123, 2017.

Eric Schulte, Westley Weimer, and Stephanie Forrest. Repairing COTS router firmware without access to source code or test suites: A case study in evolutionary software repair. In *Genetic Improvement 2015 Workshop*, pages 847–854, Madrid, 11-15 July 2015. ACM. Best Paper.