

## Andre Kuhlenschmidt

---

CONTACT INFORMATION	1212 NE 198th St Shoreline, WA, 98155	<i>E-mail:</i> andre.kuhlenschmidt@gmail.com <i>Phone:</i> (812) 325-7906 <i>Github:</i> github.com/akuhlens <i>WWW:</i> akuhlens.github.io
RESEARCH INTERESTS	I am interested in designing, building, and maintaining compilers, runtime systems, type systems, and other tools that make it easier to produce quality software. My most recent work focuses on implementing ahead of time compilers targeting LLVM.	
EDUCATION	<b>Indiana University</b> , Bloomington, Indiana USA <i>Ph.D. Candidate ABD</i> , Computer Science, May 2021 Advisor: Jeremy Siek <i>M.S.</i> , Computer Science, May 2016 <i>B.S.</i> , Business, Major: Entrepreneurship, May 2010	
EXPERIENCE	<b>Meta</b> , Redmond, Washington USA <i>Software Engineer</i> December 2020 – April 2023 <ul style="list-style-type: none"><li>• Rated as Exceeding Expectations for Individual Contributor Level 4 in 2022.</li><li>• Collaborated on the design of a programming language to make developing performant distributed applications easier.</li><li>• Started and led a research paper reading group to regularly review and discuss papers relevant to our compiler's design and implementation.</li><li>• Implemented algebraic datatypes, closures, mutable arrays, local mutable variables, and a compiler backend targeting LLVM.</li></ul> <b>Indiana University</b> , Bloomington, Indiana USA <i>Graduate Research Assistant</i> January 2014 – December 2020 <ul style="list-style-type: none"><li>• Evaluated implementation techniques for sound gradually typed programming languages.</li><li>• Developed an ahead-of-time compiler called Grift.</li><li>• Measured performance improvements in compiler via benchmark experiments.</li><li>• Designed semantics that facilitate safety and efficiency.</li></ul> <b>Meta</b> , Seattle, Washington USA <i>Software Engineering Ph.D. Intern</i> May 2019 – August 2019 <ul style="list-style-type: none"><li>• Extended Flow type checker to interactively edit code based on type inference.</li><li>• Extended Flow to automatically fix a class of errors that are caused by omitting type annotations.</li></ul> <b>Indiana University</b> , Bloomington, Indiana USA <i>Assistant Instructor</i> January 2016 – May 2018 August 2012 – May 2014 <ul style="list-style-type: none"><li>• Planned and constructed course materials and software for courses in programming language semantics and implementation, operating systems, and embedded systems.</li><li>• Instructed lab sessions of 5-30 students.</li><li>• Recognized as Assistant Instructor of the Year in 2018.</li></ul>	
CONFERENCE PAPERS	<b>Toward Efficient Gradual Typing for Structural Types via Coercions</b> Andre Kuhlenschmidt, Deyaaeldeen Almahallawi, and Jeremy G. Siek. In Programming Language Design and Implementation 2019.	
SKILLS	<b>Languages:</b> Racket, Scala, C, Haskell, Java, Python, Coq, C++ <b>Tools and Platforms:</b> Bash, Hg, Git, Make, Linux, Mac OS	

REFEREED  
ARTICLES

**An Efficient Compiler for the Gradually Typed Lambda Calculus** Andre Kuhlenschmidt, Deyaaeldeen Almahallawi, Jeremy G. Siek. In Scheme and Functional Programming Workshop, 2018.

**A Systematic Performance Evaluation of Gradually Typed Functions and References.** Andre Kuhlenschmidt, Deyaaeldeen Almahallawi, Jeremy G. Siek. In Scripts to Programs Workshop, STOP, 2016.

**Towards Absolutely Efficient Gradual Typing** Andre Kuhlenschmidt, Deyaaeldeen Almahallawi, and Jeremy G. Siek. In Scripts to Programs Workshop, STOP, 2015.

OPEN SOURCE  
PROJECTS

**Grift**

**May 2014 – December 2021**

- Collaborator on an optimizing compiler for a gradually typed Lisp to native code.
- Utilizes *space-efficient coercions* to enforce soundness of the static type system, which results in a performance increase that is multiple orders of magnitude.

PROFESSIONAL  
ACTIVITIES

IU Luddy Graduate Education Committee, Student Representative	January 2018 - August 2020
OOPSLA Artifact Evaluation Committee	August 2018
SPLASH Student Research Competition Program Committee	August 2018