Andrew Blinn </>

October 15, 2022

andrewblinn.com me@andrewblinn.com github.com/disconcision twitter.com/disconcision

Research Interests

Programming Languages • Human-Computer Interaction • Computing Education

Publications

An Integrative Human-Centered Architecture for Interactive Programming Assistants • **VLHCC** • *2022*
Andrew Blinn, David Moon, Eric Griffis, Cyrus Omar
A conceptual architecture for programming assistants addressing integrative design challenges, instantiated in OCaml-based prototype UIs for type & example-based interactive program synthesis

Tylr: A Tiny Tile-based Structure Editor • **TyDe** • *2022*
David Moon, Andrew Blinn, Cyrus Omar
A new kind of structure editing, maintaining term structure while retaining linear editing affordances

Filling typed holes with live GUIs • **PLDI** • *2021*
Cyrus Omar, David Moon, Andrew Blinn, Ian Voysey, Nick Collins, Ravi Chugh
Livelits allow users to fill program holes by directly manipulating user-defined GUIs embedded persistently into code, providing continuous graphical feedback.

Education

University of Michigan • **Ph.D Student, Computer Science** • *September 2020 - Current*
Researching user interfaces for/as programming languages at Cyrus Omar's FP Lab.
Coursework in program synthesis, category theory, HCI, programming languages

University of Toronto • **H.B.Sc, Math & Computer Science** • *May 2019*
Built a Racket-based x86/C compiler for a λ-calculus-based language with macro system.
Graduate coursework in abstract algebra, compilers, & graphics.

Industry Experience

TODAQ Toronto • **Software Development in Clojure** • *May 2019 - August 2020*
Around front: Building novel interfaces to sharpen the materiality of distributed digital assets.
Out back: Implementing features and services supporting a new protocol for decentralized digital asset management based on a Merkel-trie-derived distributed data structure.

Conferences

Invited speaker at RacketCon • *2019* • *Salt Lake City*
Spoke about Fructure, a prototype structured editor focused on edit-time term-rewriting
Recorded Talk • Fructure Slides

Presenter at VL/HCC • *2022* • *Rome*
Presented an Integrative Human-Centered Architecture for Interactive Programming Assistants
Recorded Talk • VLHCC Slides

Seat Filler • *Chicago, Salt Lake City, Toronto, Eugene, St.Louis*
2022, 2021: VL/HCC, SPLASH/OOPSLA
2019: Racket's How to Design Languages Summer School, Clojure North.
2018: Oregon Programming Languages Summer School, ICFP, Strange Loop, RacketCon

Research Projects

Hazel: Experimental IDE Design, Implementation, and At-Scale Deployment • *2020 - Current*
Led a ground-up rewrite of the Hazel IDE, extending David Moon's tylr restructuring engine into a full-fledged editor, language server, and educational tool which was deployed to 100 undergraduates.
Designed & implemented novel bidirectional type system features. More about the Hazel project

Techniques in Variability-aware Data Structures with Marsha Chechik • *2018 - 2019*
Built & profiled Haskell data structures supporting variational analysis of software product lines.
Designed & built SpyShare, a Graphviz-based tool to visually inspect data sharing.
Created and modelled a system of GHC rewrite-rules using PLT Redex.
Project Report • Presentation Slides

Independent Study in Structured Editing in Racket with Gary Baumgartner • *Summer 2017*
Self-initiated study of existing refactoring, live programming & direct manipulation tooling.
Began work on Fructure, a Racket-based polyglot structure editor, and Containment Patterns, which extend pattern matching to capture contexts as composable continuations.

Teaching

Course Development • *Summer 2018* • *University of Toronto*
Designed assignments and course materials for CSC324 - Principles of Programming Languages.
Specified and built Ductile, a toy language demonstrating exhaustive pattern matching on ADTs.
Implemented an algebraic stepper to illustrate continuations and non-determinism in Scheme.

Teaching Assistance • *University of Michigan*

2022	EECS490 Programming Languages
2021	EECS490 Programming Languages

Teaching Assistance • *University of Toronto*

2019, 2018	CSC324 Principles of Programming Languages
2018	CSC104 Introduction to Computational Thinking
2018, 2017	CSC324 Principles of Programming Languages

Mentorship

PL x Deep Reinforcement Learning for Code Completion • *2021 Fall - Current*
ML/PL collaboration: Mentoring two undergraduates, providing design and architecture guidance for a specialized version of the Hazel editor used for reinforcement-learning-based code completion research.

Interfaces for Live testing in online IDEs • *2021 - 2022*
Mentored an undergraduate who helped implement live testing features of the Hazel language and IDE, successfully deployed to a class of 100 undergraduate students for a course in Fall 2022.