

ANDREW BLINN

September 2, 2024

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

VISION

I use programming language theory to explore, explain, and engineer compositional interfaces, trying to make engagement with algebraic abstractions more fluid, tangible, and fun

FOCUSES

Programming Languages + Liveness + Learning · Human-Computer Interaction

PAPERS

[Statically Contextualizing Large Language Models with Typed Holes](#) · OOPSLA · 2024

[Andrew Blinn](#), Kevin Li, June Hyung Kim, Cyrus Omar

Program sketch refinement via collaboration between language models & language servers

[Total Type Error Localization and Recovery with Holes](#) · POPL · 2024

Eric Zhao, Raef Maroof, Anand Dukkupati, [Andrew Blinn](#), Zoe Pan, Cyrus Omar

A formal account of static error localization & recovery as applied to editor semantic services


[Gradual Structure Editing with Obligations](#) · VLHCC · 2023

David Moon, [Andrew Blinn](#), Cyrus Omar

Generalizing program holes to *syntactic obligations* allows more text-like structured editing

[An Integrative Human-Centered Architecture for Interactive Programming Assistants](#) · VLHCC · 2022

[Andrew Blinn](#), David Moon, Eric Griffis, Cyrus Omar

A syncretic account of programming assistants including a formalization of [suggestion](#)  [sensibility](#)

[Filling Typed Holes with Live GUIs](#) · PLDI · 2021

Cyrus Omar, David Moon, [Andrew Blinn](#), Ian Voysey, Nick Collins, Ravi Chugh

Livelits embed persistent user-defined GUIs in code, enabling live inline graphical feedback

WORKSHOP PAPERS

[Toward a Live, Rich, Composable, and Collaborative Planetary Compute Engine](#) · PROPL · 2024

Alexander Bandukwala, [Andrew Blinn](#), Cyrus Omar

A concept sketch for a graphical programming environment for climate science applications

[Tylr - A Tiny Tile-based Structure Editor](#) · TyDe · 2022

David Moon, [Andrew Blinn](#), Cyrus Omar

Tylr combines traditional and structured editing approaches via a novel destructuring mechanism

SCHOOL

[University of Michigan](#) · Ph.D Candidate, Computer Science · Now

Contextualizing coding with types, interfaces, & language models with [Cyrus Omar @ FPLab](#)

[University of Michigan](#) · Master's of Science, Computer Science · 2023

Coursework in PL theory, program synthesis, category theory, HCI, & the psychology of learning

[University of Toronto](#) · H.B.Sc, Mathematics & Computer Science · 2019

Graduate coursework in abstract algebra, compilers, & graphics. Advised by [Gary Baumgartner](#)

INDUSTRY

[TODAQ Toronto](#) · Full-stack development in Clojure · 2019 - 2020

Built novel front-end interfaces to [sharpen the materiality of distributed digital assets](#).

Implemented core back-end features for a decentralized digital asset management protocol

CONFERENCES

Programming Committee Member · *Pasadena*

2024: [Onward!](#) · [HATRA](#) · [LIVE](#)

Invited speaker at RacketCon · 2019 · *Salt Lake City* · [Recorded Talk](#) · [Slides](#)

Introduced [Fructure](#), a prototype structured interaction engine for edit-time term-rewriting

Accepted speaker at Midwest PL Summit · 2023 · *Ann Arbor* · [Slides](#)

Progress report on type-directed prompt construction for LLM-powered code completion

Accepted speaker at VL/HCC · 2022 · *Rome* · [Recorded Talk](#) · [Slides](#)

Presented an integrative human-centered architecture for interactive programming assistants

Guest Lecturer · 2023 & 2022 · *Ann Arbor*

Introduction to metaprogramming featuring Racket for [EECS 490 - Programming Languages](#)

Student Volunteer · *Chicago*

2021: [SPLASH/OOPSLA](#)

Seat Filler · *Rome, Chicago, Salt Lake City, Galiano, Toronto, Eugene, St.Louis, Empire Builder*

2023: [MWPLS](#), [Local First Unconf](#), [Fission TrainJam](#), [Strange Loop](#), [Gradient Retreat](#), [Causal Islands](#)

2020 - 2022: [VL/HCC](#), [Gradient Retreat](#), [SPLASH/OOPSLA](#), [HATRA](#), [LIVE](#)

2018 - 2019: [Racket Summer School](#), [Clojure North](#), [OPLSS](#), [ICFP](#), [Strange Loop](#), [RacketCon](#)

TEACHING

Course Development · 2022 - Now · *University of Michigan*

Wrote assignments and software infrastructure for [EECS 490 - Programming Languages](#)

Implemented Hazel Exercises, an educational editor integration providing progressive live feedback

Course Development · Summer 2018 · *University of Toronto*

Designed course materials for [CSC 324 - Principles of Programming Languages](#) including an

[algebraic stepper](#) illustrating non-determinism, and a [little language](#) demonstrating pattern matching

Teaching Assistantship · 2018 - Now · *Universities of Michigan & Toronto*

2023, 2022, 2021 University of Michigan [EECS 490 - Programming Languages](#)

2019, 2018², 2017 University of Toronto [CSC 324 - Principles of Programming Languages](#)

2018 University of Toronto [CSC 104 - Introduction to Computational Thinking](#)

MENTORSHIP

June (Jacob) Kim: LLM type-directed Hole-filing in TypeScript · 2024 - Now

Extracting semantic information from the TypeScript language server to inform prompt construction

Xiang (Kevin) Li: Type-constrained LLM Code Completion via token masking · 2023 - Now

Researching modifying local language model decoding to ensure semantic as well as syntactic invariants

Zachary Eichenberger & Eric Fan: Deep reinforcement learning for code completion · 2021 - 2023

Applications of typed structured editing for RL-based completion. Co-mentorship with [Ethan Brooks](#)

Yash Gaitonde: Interfaces for live feedback in teaching IDEs · 2021 - 2022

Implementing live test feedback in the Hazel IDE, deployed to a class of 100 undergraduates

PROJECTS

IDE Design, Implementation, Deployment, and Analytics with Cyrus Omar · 2020 - Now

Led a [ground-up rewrite](#) of the [Hazel IDE](#), deployed to 100 undergraduates

Investigations in Dynamic, Interactive Algebraic User Interfaces · 2022 - Now

Exploring tangibility and explorability in expository math and meta-math with [nool](#) and [furl](#)

Variability-aware Data Structures with Marsha Chechik & Ramy Shanin · 2018 - 2019 · [Slides](#)

Research in [variational analysis of SPLs](#) including building [SpyShare](#), a Haskell tool using Graphviz to visualize structure sharing, and designing + formally modelling rewrite-rule based optimizations

Independent Study in Structured Editing in Racket with Gary Baumgartner · Summer 2017

Self-directed studies in languages tooling resulting in [Fructure](#) and [Containment Patterns](#)