Josh Pollock | CV

2 Earhart St., Unit 703 – Cambridge, MA 02141

(650) 309-3662 • jopo@mit.edu • joshmpollock.com

programming languages + human-centered design = augment thought with computers

Research Statement

Computers can augment human capabilities, but to do so we need to bridge the gap between machine-readable and human-centered interfaces. Drawing on my experience in programming languages, I approach this gulf by designing domain-specific languages (DSLs). I view a DSL as a powerful human-computer interface that provides both expressiveness and simplicity via compositional language primitives.

Areas of interest: Human-computer interaction, programming languages, visualization, compilers.

Education

Massachusetts Institute of Technology — **Cambridge, MA**

Ph.D. Computer Science — *September 2020 - Present*

Advised by Daniel Jackson

Coursework:

Advanced Algorithms, Foundations of Program Analysis

University of Washington — **Seattle, WA**

B.S. Computer Science with College Honors - Magna Cum Laude — *September 2016 - June 2020*

Thesis: "Sidewinder: Designing Correct Program State Visualizations"

Highlighted Coursework:

CS Theory: (Grad) Programming Languages, (Grad) Network Verification and Synthesis, Programming Languages
CS Design: Software Engineering, Human-Computer Interaction, Data Visualization
CS Systems: (Grad) Systems for Machine Learning, Distributed Systems, Operating Systems, Security
Mathematics: Numerical Analysis (2 qtr), Differential Geometry (2 qtr), Topology, Honors Accel. Adv. Calculus (3 qtr)

Publications

Workshop

- Zong, J., Pollock, J., Wootton, D., and Satyanarayan, A. Design spaces of domain-specific languages: Comparing and contrasting approaches in pl and hci, 2021
- Pollock, J., Oh, G., Jun, E., Guo, P., and Tatlock, Z. The essence of program semantics visualizers: A three-axis model, 2020
- Pollock, J., Roesch, J., Woos, D., and Tatlock, Z. Theia: Automatically generating correct program state visualizations. In *Proceedings of the 2019 ACM SIGPLAN Symposium on SPLASH-E* (New York, NY, USA, 2019), SPLASH-E 2019, ACM, pp. 46–56
- Roesch, J., Lyubomirsky, S., Weber, L., Pollock, J., Kirisame, M., Chen, T., and Tatlock, Z. Relay: A new ir for machine learning frameworks. In *Proceedings of the 2nd ACM SIGPLAN International Workshop on Machine Learning and Programming Languages* (New York, NY, USA, 2018), MAPL 2018, ACM, pp. 58–68

ArXiv

- Roesch, J., Lyubomirsky, S., Kirisame, M., Weber, L., Pollock, J., Vega, L., Jiang, Z., Chen, T., Moreau, T., and Tatlock, Z. Relay: A High-Level Compiler for Deep Learning. *arXiv e-prints* (Apr 2019), arXiv:1904.08368

Misc. Writing

- "Writing a Research Paper: A Meta-Guide"
- "E-Graphs Are Minimal Deterministic Finite Tree Automata (DFTAs)" with Altan Haan

- "Fast(ish) Algorithms for Integer Programming: The Lost Lecture of 6.854" with Logan Weber

Research

Vega-Lite Animation — **Massachusetts Institute of Technology**
MIT Research Assistant — *February 2021 - Present*

Can we extend the Grammar of Graphics with data-driven animation?
- Co-first author. Anticipated submission: Q4 2021
- Published "Design Spaces of Domain-Specific Languages" at PLATEAU '21 based on initial work.
- Animating Gapminder, bar chart races, and bird migrations in just ten lines of code.
- Writing a compiler from Vega-Lite language extension into Vega.

Bluefish — **Massachusetts Institute of Technology**
MIT Research Assistant — *September 2020 - Present*

Can we create a grammar of discrete data diagrams?
- First author. Anticipated submission: Q1 2022
- Designing a visualization grammar for hierarchical discrete data.
- Constructing a formal mapping from data relationships to visual relationships based on.
- Implementing an embedded DSL in TypeScript for declaratively specifying diagrams.

Penrose Shape Queries — **Carnegie Mellon University**
CMU Visiting Student Researcher — *June 2021 - August 2021*

Can we identify a set of primitive functions on geometric shapes that support the layout of mathematical diagrams?
- Explored layout optimization techniques including laying out shapes one at a time and multi-resolution layout.
- Pioneered a DSL of shape queries, e.g. Hausdorff distance, to define relationships between objects in math diagrams.

Sidewinder — **University of Washington**
PLSE Research Assistant — *November 2019 - September 2020*

Sidewinder is a tool that facilitates the construction of program semantics visualizers.
- First-authored "The Essence of Program Visualizers: A Three-Axis Model" at PLATEAU '20.
- Identified three key pieces of information for explaining program semantics, transition systems, and state machines.
- Prototyped a visualization and animation DSL for explaining program semantics.
- Sketched a low-level framework for declarative diagram layout.
- Visualized small functional and imperative languages.

Theia — **University of Washington**
PLSE Research Assistant — *January 2019 - October 2019*

Theia is a tool that uses abstract machines to create correct-by-construction visualizations of program execution.
- First-authored "Theia: Automatically Generating Correct Program State Visualizations" at SPLASH-E '19.
- Creating framework for visualizing language semantics to automate existing handcrafted diagrams.
- Developing intermediate representations for abstract machines and their visualizations.
- Rewrote a subset of SML to visualize programs.

TVM and Relay — **University of Washington**
PLSE and SAMPL Researcher Assistant — *January 2018 - June 2019*

TVM is an open-source end-to-end deep learning compiler stack employed by frameworks such as PyTorch and MXNet.
- Co-authored "Relay: A High-Level Compiler for Machine Learning", in submission.
- Co-authored "Relay: A New IR for Machine Learning Frameworks", which appeared in MAPL '18.
- Developed Python frontend compiler and Relay text format to enable better developer experience.

Lean Theorem Prover — **University of Washington**
PLSE Researcher — *June 2017 - August 2017*
- Designed and implemented a transpiler in OCaml/ReasonML to transfer libraries from Coq to Lean.
- Contributed to Lean's open source codebase. Learned graduate-level dependent type theory and cutting-edge software verification tools.

Service

- 2021
  - Graduate Application Coach — MIT Graduate Application Assistance Program (GAAP)
  - Student Volunteer — UIST '21
  - HCI Graduate Student Representative — MIT CSAIL Postdoc and Graduate Student Council
  - Social Organizer and Website Maintainer — HCI Social Planning Committee
- 2019
  - Teaching Assistant — UW Programming Languages
- 2018
  - Teaching Assistant — UW Accelerated Honors Math

Teaching

- **Programming Languages** — **University of Washington**
  Teaching Assistant (10-15 hrs/wk) — *April 2019 - June 2019*

  UW's Programming Languages course covers topics such as type systems, higher-order functions, and double dispatch.
  - Investigated the use of visual explanations in lectures. (Early work on Theia.)
  - Introduced students to UW programming languages research.
  - Developed slides weekly explaining new concepts and deep connections to programming language theory.
  - Held office hours once a week. Taught section once a week. Graded homework weekly.

- **Accelerated Honors Math** — **University of Washington**
  Teaching Assistant (10-15 hrs/wk) — *September 2017 - June 2018*

  Accelerated Honors Math attracts top UW STEM freshmen and provides a rigorous intro to 100- & 300-level math.
  - Facilitated students learning intro calculus, differential equations, and linear algebra from a proof-based perspective.
  - Prepared and presented special topics once a week.
  - Fostered students' interests in deeper math course material.
  - Held office hours twice a week. Taught quiz section once a week. Graded homework weekly.

Work

- **Apple Inc.** — **Cupertino, CA**
  Formal Verification Intern (40 hrs/wk) — *July 2019 - September 2019*

  Apple's Formal Verification group proves properties about Apple's SoCs to ensure they are correct and secure.
  - Wrote >5,000 lines of code in the Isabelle proof assistant.
  - Developed proof-of-concept extensions to existing Isabelle proofs of crucial software running on Apple SoCs.
  - Prototyped extension to VSCode Isabelle plug-in for enhanced proof state inspection.
  - Presented to management on internship work and helped prepare team for presentation to upper-level management.

- **Intel Corporation** — **Hillsboro, OR**
  IPAS Undergraduate Technical Intern (40 hrs/wk) — *June 2018 - September 2018*

  Intel Product Assurance and Security (IPAS) is a research-focused group that seeks to address Intel security threats.
  - Designed and coded proof of concept for concolic execution of BIOS to automatically detect security vulnerabilities.
  - Presented prototype, symbolic and concolic execution, and internship experience to over 60 members of IPAS.
  - Learned about UEFI/BIOS, x86 and x86_64 assembly, Intel Security Development Lifecycle, and concolic execution.

- **Vulcan Robotics** — **San Mateo, CA**
  Cofounder (15 hrs/wk) — *2014-2016*

  Completely student-run high school robotics team. 2015-2016 World Championship 2nd-Place Winning Alliance Captain.
  - Planned team presentations and delegated speaking roles.
  - Evaluated and implemented team organization strategies including long-term planning and competition reflection.
  - Used Tableau visualizations to scale scouting protocols from competitions with 16 to 128 participants.

Presentations

- 2021: MIT HCI Show & Tell - Bluefish
- 2020: Bachelor's Thesis Presentation - Sidewinder
- 2019: SPLASH-E Talk - Theia
- 2019: HACKERS Talks - Machine Learning, Visualization
- 2019: UW CSE Research Poster Session - Relay. *Tied for second place for most impactful research.*
- 2019: Stanford Research Conference Poster Presentation - Relay (∼20% acceptance rate)
- 2018: UW CSE Graduate Research Showcase Poster Session - Relay
- 2017: HACKERS Talk - Formal Methods
- 2015: HACKERS Talk - Math Education
- 2015: HACKERS Lightning Talk - Vulcan Robotics. *Awarded one of the top lightning talks.*