

# Kenny Foner

Email: **kwf@very.science** Website: **very.science**  
Phone: **(508) 318-8593** GitHub: **github.com/kwf**  
Location: **Philadelphia, PA** Pronouns: **they/them/their**

## ★ WHO AM I?

I'm a **software engineer** with deep background in the research and development of **programming languages, formal verification, and applied cryptography**. I love creating type systems, compilers, and tools for automated testing and theorem proving. I've used these techniques in practice to build systems for cryptographic specification, privacy-preserving computation, and more—and I'm just as excited to learn new domains and tackle new challenges!

I take pride in collaboratively creating software that is intuitive, well-documented, and actively helpful, delighting developers and users with polished, thoughtful engineering work. I seek to apply meaningful theory towards real problems, using my craft as a computer scientist to make a positive difference for programmers and for the world.

## ★ EDUCATION

**University of Pennsylvania** (Philadelphia, PA) May 2018  
MASTER OF SCIENCE IN ENGINEERING in Computer and Information Science  
Advised by Dr. Stephanie C. Weirich  
Teaching: *Advanced Programming (Haskell)*, *Software Foundations (Coq)*

**Brandeis University** (Waltham, MA) May 2015  
BACHELOR OF SCIENCE in Computer Science *summa cum laude* with highest departmental honors (inducted ΦBK)  
Advised by Dr. Harry G. Mairson  
Thesis: *Getting a Quick Fix on Comonads* (later published in *Haskell Symposium '15*)  
Teaching: *Functional Programming (Haskell)*, *Structure & Interpretation of Computer Programs (Scheme)*

## ★ WORK EXPERIENCE

**Galois** (Remote) 2018 – present  
SOFTWARE ENGINEER/RESEARCHER  
I've worked on the zero-knowledge voter record verifier for the end-to-end encrypted ElectionGuard voting system, a re-designed language server for the SAW/Cryptol suite of open-source program analysis and cryptographic specification tools, a novel interface to the Crux verification tool for LLVM program analysis, and several other projects. Along the way I've authored and contributed to a number of open source projects (listed in a separate section).

**Microsoft Research** (Redmond, WA) Summer 2016  
RESEARCH INTERN  
I formalized the metatheory of several small programming languages as a stress test for the experimental Dafny language/proof assistant, and designed an intermediate language to verify a multi-part compilation pipeline. I contributed to Dafny's development, implementing bug fixes and feature improvements.

**Galois** (Portland, OR) Summer 2015  
RESEARCH INTERN  
I implemented a user-friendly interactive code generation utility for the open-source SAW suite of program analysis tools, designed to help cryptography domain experts get started more easily with automated formal verification. In a separate project, I worked on the design of a prototype graph query language.

**Galois** (Portland, OR) Summer 2014  
RESEARCH INTERN  
I created an embedded domain-specific language for secure distributed computations, implemented using multi-party secret-sharing. I developed an optimizing compiler, and an efficient bytecode interpreter which was several times faster than the previous best results on a series of established benchmarks.

**MIT Lincoln Laboratory** (Lexington, MA) Summer 2013  
RESEARCH INTERN  
I prototyped applications and protocols to evaluate experimental frameworks for dynamic information flow control (IFC). Within one of these, I implemented a secure distributed multi-player game of *Battleship*, and contributed to a comparative analysis of IFC frameworks published in PLAS '14.

## ★ PROGRAMMING BACKGROUND

**Expert Knowledge: Haskell** – I’ve been programming in Haskell for more than seven years, and currently use it in my day-to-day work. I’ve authored libraries using advanced type-system extensions, implemented compiler plugins, and published novel research at top-tier academic conferences about high-performance immutable data structures, strictness analysis, and generic programming.

**Strong Professional Background: Rust** – I’ve been programming in Rust for several years, and currently use it in my day-to-day work. I’ve contributed several open source packages to the ecosystem, including the Myxine server for rapid language-agnostic GUI prototyping and the Hopscotch data structure for fast tagged skip-queues. I’m familiar with the async ecosystem and HTTP stack, having built multiple applications atop them.

**Assorted Professional Experience: Python, JavaScript** – I’ve used Python to implement a number of projects: notably, bindings to the Myxine server and an idiomatic front-end to the SAW verification tool. I’m also conversant in JavaScript, though not yet expert: Myxine’s front-end is written in vanilla JavaScript, making use of a number of newer web technologies.

**Academic Experience: OCaml, Coq, Scheme** – I’ve used these languages in my academic career to collaborate on research, create course materials, and teach university classes.

**Other Experience: Clojure, C, Java** – I’ve used these languages in less-recent work, in university courses, or in hobby projects. I can read them, and I’m comfortable working in them with the aid of reference materials.

## ★ OPEN SOURCE CONTRIBUTIONS

I’m the primary author and maintainer of the **Myxine** server for rapid language-agnostic GUI prototyping, the **StrictCheck** library in Haskell for randomized dynamic demand analysis, the **Hopscotch** data structure in Rust for efficient tagged skip-queues. I’m also a co-author and co-maintainer of the **Urn** data structure in Haskell for updateable discrete probabilistic sampling. I’m a current contributor to the **Cryptol** specification language and the **Software Analysis Workbench**. I’ve also contributed to the **Glasgow Haskell Compiler** and the **Dafny** language.

## ★ PUBLICATIONS & TALKS

**PL Mentoring Workshop at SPLASH ’19, POPL ’20, ICFP ’20 (upcoming):** “How Can I Academia When My Brain Can’t Even? Mental Health in Grad School and Beyond” (non-technical talk)

**Compose ’19:** “Functors of the World, Unite!” (talk)

**ICFP ’18:** KEEP YOUR LAZINESS IN CHECK. [K. Foner](#), H. Zhang, and L. Lampropoulos. In *Proceedings of the 2018 ACM SIGPLAN International Conference on Functional Programming*.

**ICFP ’18:** “Keep Your Laziness in Check” (talk)

**ICFP ’18:** WHAT’S THE DIFFERENCE? A FUNCTIONAL PEARL ON SUBTRACTING BIJECTIONS. B. Yorgey and [K. Foner](#). In *Proceedings of the 2018 ACM SIGPLAN International Conference on Functional Programming*.

**Haskell ’17:** ODE ON A RANDOM URN (FUNCTIONAL PEARL). L. Lampropoulos, A. Spector-Zabusky, and [K. Foner](#). In *Proceedings of the 2017 ACM SIGPLAN Symposium on Haskell*.

**Compose ’17:** “Choose Your Own Derivative” (talk)

**TyDe ’16:** CHOOSE YOUR OWN DERIVATIVE (EXTENDED ABSTRACT). J. Paykin, A. Spector-Zabusky, and [K. Foner](#). In *Proceedings of the 2016 ACM SIGPLAN Workshop on Type-Driven Development*.

**Compose ’16:** “There and Back Again and What Happened After” (talk)

**Haskell ’15:** FUNCTIONAL PEARL: GETTING A QUICK FIX ON COMONADS. [K. Foner](#). In *Proceedings of the 2015 ACM SIGPLAN Symposium on Haskell*.

**Haskell ’15:** “Functional Pearl: Getting a Quick Fix on Comonads” (talk)

**PLAS ’14:** YOU SANK MY BATTLESHIP!: A CASE STUDY IN SECURE PROGRAMMING. A. Stoughton, A. Johnson, S. Beller, K. Chadha, D. Chen, [K. Foner](#), and M. Zhivich. In *Proceedings of the 2014 ACM Workshop on Programming Languages and Analysis for Security*.

## ★ SERVICE

**#ShutdownPL ’20:** I co-chaired this virtual day of action against racism in the programming languages research community (a part of the larger #ShutDownSTEM day of action). Organized in less than 48 hours, the event drew 165 faculty, industry researchers, and students, and consisted of 6 parallel 5-hour discussion tracks focusing on building lasting commitments to substantive change. Subjects discussed ranged from building actively anti-racist mentoring structures to founding a group to increase the transparency of police and military spending on PL research. I am presently involved in efforts to replicate and grow this anti-racist work as an official component of academic conferences on programming languages.

**Mental health advocacy:** I’ve given recurring invited talks about mental illness within academia and industry. I am outspoken in public about my own experience of bipolar disorder and ADHD and regularly speak with people across my field who experience mental illness and need someone to talk to. I am determined to use my own privileges to work against the stigma attached to these and all other mental illnesses and neurodivergent identities.