Trevor Elliott

Experience

2019–Present **Infrastructure Engineer**, *Stripe*, Portland, OR.

Working on the sorbet typechecker and compiler for ruby.

- Key contributor on the sorbet compiler, implementing optimizations and improving coverage for the ruby language. Generated native code with llvm, targeting the ruby vm's c api.
- Implemented type-system features and improved type-checker runtime performance.
- Mentored an intern on the sorbet compiler project, and organized/ran regular meetings with external contributors from other companies.

2017–2019 **Compiler Engineer**, *Groq, Inc.*, Portland, OR.

Worked on a compiler for tensorflow models, targeting a custom ASIC that accelerates inference. Focusing on compiler performance and optimization of results.

2007–2017 **Engineer and Researcher**, *Galois, Inc.*, Portland, OR.

Contributed to a broad range of projects, notably:

- Developed domain specific languages in Haskell, including the msf-haskell metasploit DSL and the Ivory DSL for memory-safe embedded programming
- Developed and contributed to multiple full language implementations, including Cryptol and the Salty GR(1) reactive synthesis DSL
- Implemented the HaNS network stack
- Aided ASIC design and verification for lightweight cryptographic primitives from Cryptol specifications

2004–2007 **Developer**, *CollegeNET*, Portland, OR.

Helped to transition a large desktop application to a suite of web services, and developed tools based off of those web services. Also assisted with some front-end web development, and implemented an LDAP-based authentication system.

2002–2004 **Technical Support**, *CollegeNET*, Portland, OR.

Provided technical support to a large client base.

Education

2008 **B.S. Computer Science**, *Portland State University*, Portland, OR.

Open Source Projects

sorbet	The sorbet typechecker and compiler for ruby
cereal	Haskell library for fast binary format parsing/rendering
ivory	Haskell EDSL for memory-safe embedded programming
salty	Reactive synthesis DSL for GR(1) specifications
HaNS	A TCP/IP network stack implemented in haskell
Cryptol	DSL for implementation and verification of cryptographic algorithms
HaLVM	Port of the GHC runtime to the Xen hypervisor
llvm-pretty	An EDSL for generating textual llvm bitcode
huff	A fast-forward based classical planning EDSL

Skills

Languages	haskell, c/c++, rust, java, javascript, fsharp, ocaml, python, ruby, lua, assembly (x86, arm), coq, isabelle/hol, shell scripting
Tools	z3, smt-lib, lldb, gdb, bazel, gnu make, vim/neovim

(503) 936-6575 • *awesomelyawesome@gmail.com* • *elliottt*

Publications

Trevor Elliott, Mohammed Alshiekh, Laura R. Humphrey, Lee Pike, and Ufuk Topcu. Salty-a domain specific language for gr(1) specifications and designs. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4545–4551, 2019.

Trevor Elliott, Lee Pike, Simon Winwood, Pat Hickey, James Bielman, Jamey Sharp, Eric Seidel, and John Launchbury. Guilt free ivory. In *ACM SIGPLAN Notices*, volume 50, pages 189–200. ACM, 2015.

Patrick C Hickey, Lee Pike, Trevor Elliott, James Bielman, and John Launchbury. Building embedded systems with embedded dsls. In *ACM SIGPLAN Notices*, volume 49, pages 3–9. ACM, 2014.

John Launchbury and Trevor Elliott. Concurrent orchestration in haskell. In *ACM Sigplan Notices*, volume 45, pages 79–90. ACM, 2010.

Dylan McNamee and Trevor Elliott. Secure historian access in scada systems. *Galios, White Paper*, 2011.

Lee Pike, Pat Hickey, Trevor Elliott, Eric Mertens, and Aaron Tomb. Trackos: A security-aware real-time operating system. In *International Conference on Runtime Verification*, pages 302–317. Springer, Cham, 2016.

Lee Pike, Patrick Hickey, James Bielman, Trevor Elliott, Thomas DuBuisson, and John Launchbury. Programming languages for high-assurance autonomous vehicles. In *Proceedings of the ACM SIGPLAN 2014 Workshop on Programming Languages meets Program Verification*, pages 1–2. ACM, 2014.

Tristan Ravitch, E Rogan Creswick, Aaron Tomb, Adam Foltzer, Trevor Elliott, and Ledah Casburn. Multi-app security analysis with fuse: Statically detecting android app collusion. In *Proceedings of the 4th Program Protection and Reverse Engineering Workshop*, page 4. ACM, 2014.