Shu-yu Guo

461 2nd St. C203
San Francisco, CA 94107

Phone: (302) 252-1530
Email: shu@rfrn.org

Summary

I am a systems programmer who specializes in virtual machines, compilers, and developer tooling. I have 5+ years of experience writing virtual machines at web scale. I am also a formally trained programming language semanticist and enjoy standards work.

Key Accomplishments

Core contributor to Firefox. Specifically SpiderMonkey, the JavaScript virtual machine. Contributes to the frontend, runtime, JIT compilers, and developer tools, along all three axes of performance, features, and correctness.

Architected SpiderMonkey's parser and bytecode compiler. Most recently, led the modernization of binding name analysis.

Developed platform support for the JavaScript debugger to seamlessly debug optimized JIT code. Also led development of JIT-aware sampling profiler.

Published several academic papers, both theoretical and practical, contributing to the compiler correctness and type inference fields.

Is a member of the ECMAScript standardization committee. Wrote SharedArrayBuffer's memory model.

Technical Qualifications

C++

JavaScript

x86 and some ARM assembly

Virtual machine design and implementation (frontend, runtime, GC, JITs)

Compiler implementation

Developer tooling

Formal semantics writing

Technical specification writing and programming language lawyering

Education

Ph.D. Computer Science, **UCLA**, (dropout)
M.S. Computer Science, **UCLA**, 2011
B.S. Computer Science, **University of Chicago**, 2008
B.A. Linguistics, **University of Chicago**, 2008

Industry

Senior Software Engineer 2017–
Bloomberg LP
Engineering of core UI technologies of the Bloomberg terminal with a focus on JavaScript VMs; delegate at ECMAScript standardization committee.

Staff Platform Engineer 2016–2017
Mozilla
Tech lead and manager of JavaScript language team and ECMAScript compliance; delegate at ECMAScript standardization committee. The team is responsible for the JavaScript parser, bytecode compiler, runtime, and the implementation of new JavaScript features.

Staff Research Engineer 2012–2016
Mozilla
ECMAScript compliance; JIT optimizations; JIT-aware debugger and optimization-aware sampling profiler; extending JavaScript with parallelism; Flash VM in JavaScript.

Research Intern 2010, 2011
Mozilla
Designed and implemented a one-pass parse-time SSA construction algorithm for JavaScript. Worked on the tracing JIT compiler and hybrid type inference for JavaScript.

Academia

Ph.D. Student 2008–2011
Jens Palsberg **UCLA**
Programming language semantics. Investigating proof techniques for correctness of dynamic compilation techniques, specifically, proving correctness for trace compilation.

Research Assistant 2006–2008
Robby Findler **University of Chicago**
Modeling of software contracts on immutable data structures. Analysis of complexity and optimization of software contract implementation in PLT Scheme.

Publications

EMME: a formal tool for ECMAScript Memory Model Evaluation.
Cristian Mattarei, Clark Barrett, Shu-yu Guo, Bradley Nelson, Ben Smith, JF Bastien. TACAS 2018.

Optimization Coaching for JavaScript.
Vincent St-Amour and Shu-yu Guo. ECOOP 2015.

Fast and Precise Hybrid Type Inference for JavaScript.
Brian Hackett and Shu-yu Guo. PLDI 2012.

The Essence of Compiling with Traces.
Shu-yu Guo and Jens Palsberg. POPL 2011.

Lazy Contract Checking for Immutable Data Structures.
Robby Findler, Shu-yu Guo, and Anne Rogers. IFL 2007.

Blog Posts

Parsing Binding Names for Efficient Representation. 2016.

Debugging in the Time of JITs. 2014.

Two Reasons Functional Style Is Slow in SpiderMonkey. 2013.

Committees

Member of TC39, ECMAScript standardization committee.