Michael Ficarra

Email: github.resume@michael.ficarra.me
Address: (REDACTED) · Phone: (REDACTED)

Portfolio

Github `https://github.com/michaelficarra`
Speaker Deck `https://speakerdeck.com/michaelficarra`

Interests

programming language design · compilers · program transformations · protocol analysis
software security · provable security · logic programming · ECMAScript interpreters
ECMAScript static analysis tools · SpiderMonkey AST format

Experience

Application Security Engineer, Groupon *since September 2012*
Augmented various development teams to keep a security eye on their design and implementation. Built a variety of tools and services to help prevent many kinds of fraud and abuse. Redesigned and led implementation of core authentication mechanism as dedicated service.

CoffeeScript Fellow, Groupon *June to September, 2012*
Designed and built an alternative CoffeeScript compiler with an improved compilation strategy, a declarative specification, and standardised interfaces. Contributed to other FOSS projects that were used as dependencies, including adding source map support to the most popular JavaScript code generator.

Google Caja Intern, Google Summer of Code 2010 *May to August, 2010*
Caja is a Google project whose goal is to allow rich web aplications to safely embed untrusted third-party HTML, CSS, and JavaScript. Caja allows communication between modules through exchanges of references to JS objects, relying on an object-capability model for security. Determined why the behaviour of popular JS libraries was not being preserved, and proposed fixes to both the Caja and library developers.

Web Application Developer, Solstice Technologies *January 2008 to August 2010*
Worked on and managed projects that required the use of server-side scripting languages, most commonly PHP, and a relational database such as MySQL. Projects were taken from concept to completion through collaboration and teamwork with fellow employees.

Intern, Skarven Enterprises, Berkeley Heights, NJ *May to August, 2009*
Designed and implemented a custom, domain-specific programming language to be used by people with little prior knowledge of programming languages so they could create integration tests for the company's main product. The compiler and interpreter for the custom language were written in Ruby.

Speaking

SpiderMonkey Parser API: A Standard For Structured JS Representations

An Analysis of the Redesign of the CoffeeScript Compiler

Select Ways to Harness the Power of JavaScript (based largely on Effective JavaScript)

Introducing "CoffeeScript II: The Wrath of Khan"

Achievements

Maintainer of CoffeeScript language and active contributor to both major compilers

Technical editor for the Pragmatic Programmers' CoffeeScript book

Contributor to at least 60 open source projects, mostly JavaScript and CoffeeScript static analysis and transformation tools

Education

Worcester Polytechnic Institute, Worcester, MA

Bachelor of Science in Computer Science, awarded February 2011
Major Qualifying Project: Generating Universal Models for Geometric Theories

Influential Coursework

Software Security Engineering · Protocol Analysis · Database Systems · Algorithms
Techniques of Programming Language Translation · Discrete Maths · Operating Systems