Jon Sterling

Email: jon@jonmsterling.com
Homepage: jonmsterling.com
GitHub: github.com/jonsterling

Background

Reed College (Young Scholars), 2009–2010

I studied Latin at Reed College during my last year of High School.

UC Berkeley, 2010–2013.

BA in Linguistics, with interest in historical linguistics, compositional syntax & semantics, and philology (Ancient Greek, Sumerian, Akkadian, Hittite, Latin, Old English); I am particularly interested in the syntax of hyperbaton in ancient Indo-European languages, and how it corresponds with the associativity of function composition in type-theoretic grammars. During my time at Berkeley, I held a research apprenticeship cataloguing and translating Sumerian administrative records from the Ur III period (2100-1900 BCE) at Umma.

Skills & Interests.

I am proficient in Haskell, Agda, Idris, Coq, Objective-C, JavaScript, TypeScript; I am also interested in OCaml, SML and Twelf. I study type theory, proof theory, (multi-modal) combinatory categorial grammar, linguistics, and digital humanities. In my free time, I implement variants of Martin-Löf Type Theory, including ETT and OTT.

Recent Experience

Yardsale/FOBO, 2012–2014

Reponsible for architecture and implementation of the Yardsale and FOBO iOS client applications, along with one junior engineer; I also maintained the server code for FOBO. (Objective-C, TypeScript, Haskell)

Open Source Development

Vinyl is a Haskell package which provides type-safe extensible records with subtyping and lenses; Vinyl is *morally* a dependently typed notion, implemented using the usual GHC suspects (type families, data kinds, and singletons).

TT-Reflection is a type checker for an experimental type theory which introduces strict extensional equality in manageable doses, unlike traditional Extensional Type Theory. This is my most important work!

ReactiveFormlets is an Objective-C library for form abstraction, backed by **ReactiveCocoa**. My work has so far involved creating a very large number of client-side data entry forms with complex validation requirements; ReactiveFormlets made it possible to achieve this rapidly and without incident.

Other Experience

Small Society, Summers 2010 & 2011 (iOS Engineer)