

From System F to Typed Assembly Language

GREG MORRISETT and DAVID WALKER

Cornell University

KARL CRARY

Carnegie Mellon University

and

NEAL GLEW

Cornell University

We motivate the design of a *typed assembly language* (TAL) and present a type-preserving translation from System F to TAL. The typed assembly language we present is based on a conventional RISC assembly language, but its static type system provides support for enforcing high-level language abstractions, such as closures, tuples, and user-defined abstract data types. The type system ensures that well-typed programs cannot violate these abstractions. In addition, the typing constructs admit many low-level compiler optimizations. Our translation to TAL is specified as a **sequence of type-preserving transformations, including CPS** and closure conversion phases;