

Reigniting Fuse, an Online Partial Evaluator for Scheme

PAULETTE KORONKEVICH, University of British Columbia

Partial evaluation is the process of running programs with only the information available at compile-time. This technique is particularly useful as an optimization technique for compilers, as it creates smaller, more concise programs *before* compilation. Online partial evaluators can be thought of as program interpreters, but instead of returning a program value, an online partial evaluator returns a new program. This new program contains values that could be computed without the knowledge of run-time, or *dynamic*, values. These dynamic values often cause termination issues for online partial evaluators. Evaluating a recursive function over dynamic values will not terminate as the dynamic value forces both the base and recursive cases of a function to be evaluated. Fuse [1], an automatic online partial evaluator for Scheme, employs a novel use of *online generalization* which allows terminating recursive calls to be fully evaluated and enables termination on dynamic values. I employ Fuse's technique in my own online partial evaluator for Racket's linklets — primitive elements of compilation, bytecode marshaling, and evaluation. My lightning talk will both explain and demonstrate Fuse's termination technique for online partial evaluation of Racket programs.

REFERENCES

- [1] Daniel Weise, Roland Conybeare, Erik Ruf, and Scott Seligman. 1991. Automatic online partial evaluation. In *Functional Programming Languages and Computer Architecture*, John Hughes (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg.