, , 1 ()

Continuation-Based Multiprocessing Revisited

MITCHELL WAND * wand@ccs.neu.edu
College of Computer Science, Northeastern University, Boston MA 02115

In the spring of 1979, I had seen a number of papers on implementing Scheme directly on VLSI chips. This led me to ask what were the minimal extensions to Scheme necessary to build an operating system for such a chip. It appeared that three facilities were necessary: data protection, process saving, and elementary exclusion. Although exclusion clearly required support from the underlying machine, data protection could be performed easily by hiding data in closures. The main contribution of the paper was to point out that process-saving, like data protection, required no new extensions; it could be accomplished using the technology of continuations.

The model I adopted would now be called *multithreading*: the management of multiple threads of execution in a shared address space. This model seems to have been adopted in many Scheme implementations; it is somewhat harder to determine whether the underlying implementations are based on `call/cc` but it appears that at least some are.

The programs were written in SCHEME 3.1, a Scheme implementation I had been working on at the time. Unlike Sussman and Steele's original implementation, this version translated the input Scheme code into a byte-code-like abstract machine code, thus forcing a clear separation between Scheme code and Lisp code.

Except for reindentation, I have kept the original code intact. `(catch` x e`)` would now be written `(call/cc (lambda (`x`)` e`))`; `block` is now called `begin`, `asetq` is now called `set!`, and `labels` is now called `letrec`. In addition, I used a syntax for `define` whose meaning should be self-evident; `de` is a syntax for definition in the underlying Lisp.

Likewise, except for fixing a few typographical errors, the text has been left unchanged, including a number of parenthetical remarks that I now find annoying and a number of occurrences of "which" that I would now change to "that."

A few clarifications and afterwords have been added as endnotes.