SETL Newsletter Number 15

February 9, 1971
J. T. Schwartz

A proposed SETL implementation plan through the end of the bootstrap phase

Phase 1. Define and write a set of BALM macros to create analogs of some of the principal SETL features, especially $\forall x$ iterations, $\exists[x]$, functional application, set-former, the data structures to be used here being basically simple BALM lists.

This will create a very preliminary working tool, designated BALMSETL.

Phase 2. In BALMSETL, and working from the SETL algorithms now available, write the following programs:

a. Lexical scan setup routine

b. Lexical scanner

c. Preparser

d. Postparse setup routine

e. Postparser.

This will create a general language-defining front end, whose output will be BALM lists of a standard form to be specified.

Phase 3. Write SETL-defining syntax information for this set of programs. Run the setup routines against this information, to create BALM programs embodying an approximate SETL front end. Compile this.

This will give a front end producing standard form BALM lists representing digested codes in what is approximately SETL.

The language acceptable to this parsing structure will be designated BOOTSETL.

Phase 4. Design BALM data structures for the SETL objects which are more efficient than naive lists, and write the BALM procedures to apply all the basic SETL operations to these data structures. This will create the BOOTSETL back end; hook it to the BOOTSETL front end, thereby getting a running BOOTSETL - unoptimized.

Phase 5. Design a first set of optimization procedures to work on the code-lists produced by the BOOTSETL front end, and to transform them into improved code-lists of the same form. Insert this middle section between the front and back ends, to obtain a running BOOTSETL - optimized.

After this, a period of experimentation with BOOTSETL - optimized would seem to be in order.