MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Reply to: Project MAC
545 Technology Square
Cambridge, Mass. 02139

Telephone: (617) 864-6900 x6201

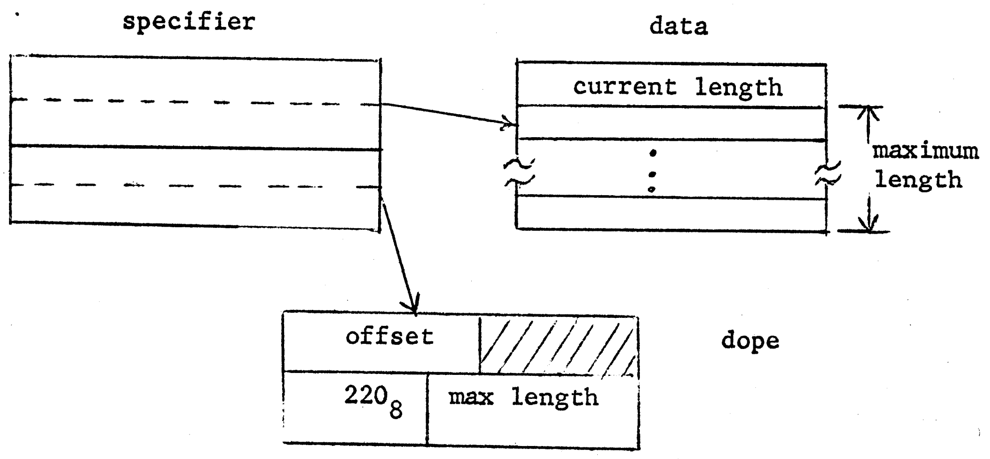
29 December 1966

Mr. M. Douglas McIlroy
Bell Telephone Laboratories
Murray Hill, New Jersey

Dear Doug:

As we gain experience with EPL it is becoming clear that the current implementation of varying strings is rather inefficient. The forthcoming MSPM section on standards for library subroutines, for example, heavily discourages use of varying strings. Allocating and freeing take a fair amount of time, and further the necessity for allocating and freeing means it will never be possible to manipulate varying strings in-line instead of through calls out. Finally, the necessity of initializing and terminating automatic varying strings makes a synthetic epilogue necessary. There has been considerable argument lately about the synthetic epilogue, and the conclusion seems to be that there will be a synthetic epilogue procedure built into Multics but that it is liable to be fairly expensive.

The solution I would like to propose involves forcing the user to use reasonable maxima on his varying strings. A possible implementation is:



A varying string then looks like an aligned non-varying string except that the 020 bit in the identity code is used to indicate that the actual length is contained in the word preceding the data. The error of passing a varying string to a program expecting a non-varying string would then not cause the catastrophic results which I have experienced several times but simply cause the program to see some unnecessary garbage at the end of the string.

Another interesting advantage of this scheme is that it once again becomes possible to give the controlled attribute to varying strings and to aggregates containing them.

Rather than demand that the user specify a reasonable maximum size for a varying string, we could instead have two different representations for the same language data-type: "short varying string" (that proposed here) and "long varying string" (the present implementation). The compiler could then represent varying strings whose maxima are greater than some number N of words by long varying strings, and the rest by short varying strings. N should be very large (say about 1000) and should in general be at the whim of the compiler. One can even imagine an option used with the options attribute such as

varying - string - cutoff (N)

which would let the user specify N. All software which expects varying strings as arguments must then be prepared for either representation, no matter what the declared maximum is.

I would like to see short varying strings implemented in whatever compiler we use for Phase II of Multics. I am personally in favor of eliminating long varying strings altogether but would be happy enough to see the two-sided representation with a user-specified cutoff, as long as the compiler tried very hard to avoid the use of long varying temporaries.

The BP. Sections of MSPM do not contain any information on varying strings at the moment, and it would seem wise to get this matter settled soon so PL/1 documentation can be closed. Therefore I would appreciate an early reply.

Sincerely,

Don Wagner

Donald B. Wagner

cc: F. J. Corbató
E. L. Glaser
R. M. Graham
A. Evans
K. Martin
R. Daley
J. Saltzer✓