ML/I User's Manual — Appendix N

Implementation on the Olivetti L1

R.D. Eager

September 1985

This implementation is based on version CJB of ML/I.

Copyright © 1985 R.D. Eager

N.1 Restrictions and Additions

This implementation of ML/I contains all the features described in the *ML/I User's Manual*, 4th Edition, August 1970, plus New Features 1 to 4 as described in the supplements to that manual (startlines, stop markers, controlled line numbers and optional warning markers). In addition it contains a facility for suppressing the context print-out following a message produced by a call of the operation macro `MCNOTE`.

N.2 Operating instructions and I/O

The following options are accepted by ML/I. Upper and lower case option letters are accepted, although only lower case ones are shown below.

-v   Print the version number of this implementation of ML/I.

-w *n*   Set the amount of workspace available to ML/I to *n* words (the default is 5000 words).

-l *file*   Nominate *file* as the listing file. The default is that no listing is produced. The name – is taken to mean the standard output.

-d *file*   Nominate *file* as the debugging file. By default, this is the standard error stream (usually the user's terminal). The name – is taken to mean the standard output.

-o *file*   Nominate *file* as an output file. Up to two output files may be specified; each must be preceded by the -o flag. If no output files are specified, the standard output is used. An output file named – is also taken to be the standard output.

All other arguments to ML/I are taken to be the names of input files; there may be no more than five of these. If no input file is specified, the standard input is used. The name – is also taken to mean the standard input.

N.2.1 Control of input

Input may be read from any one of the input streams; the value of S10 controls the selection. The possible values are:

S10 = 1   Input is taken from the first input file given as an argument. If there are no input files specified, input is taken from the standard input.

S10 = 2   Input is taken from the second input file given as an argument. If this argument is omitted, a fatal error will result when S10 is set to 2.

S10 = 3   Input is taken from the third input file given as argument. If this argument is omitted, a fatal error will result when S10 is set to 3.

S10 = 4   Input is taken from the fourth input file given as argument. If this argument is omitted, a fatal error will result when S10 is set to 4.

S10 = 5   Input is taken from the fifth input file given as argument. If this argument is omitted, a fatal error will result when S10 is set to 5.

If `S10` is set to zero, ML/I treats this as "end of file" and ceases processing. If `S10` is set to any illegal value (negative, greater than five, or a value between one and five associated with an input stream which has not been specified) then the process is aborted.

If a change of input stream is made, the original stream is not "forgotten". Any attempt to read from this stream again will cause ML/I to carry on where it left off. When the end of an input stream is reached, ML/I checks to see if it is the "revert stream". If it is, the process is terminated; otherwise input is switched to the revert stream, and processing continues. The revert stream is initially 1; its value is held in `S23` and may be altered by the user if required.

There is no restriction on the length of an input line other than that imposed by the operating system.

N.2.1.1 Input translation facility

It is possible to designate that one character be translated to another on input. This makes it possible to input a character that a device does not support. However, only one character code can be translated in this way.

If it is desired to perform a translation, `S16` should be set to the ASCII code of the character to be translated, and `S17` to the ASCII code of the character that is to replace it. For example, if `%` (ASCII 37) was to represent a tab (ASCII 9), `S16` and `S17` should be set in the following way:

```
MCSET S16 = 37
MCSET S17 = 9
```

Initially, `S16` has the value -1, which since it does not correspond to a valid internal code, will not cause any translations to be made.

N.2.1.2 Ordering of input operations

The ordering of input operations is as follows:

a. Checking for `S10` equal to zero.

b. Checking for invalid values of `S10`.

c. Check for end of file (if the revert stream is selected as a consequence of this check, return to b)).

d. Translation using `S16` and `S17`.

e. Checking for illegal characters.

N.2.2 Control of output

Output may be directed to any combination of the nominated output files, including none at all. The values of `S21` and `S22` control the selection; `S21` controls output to the first file, and `S22` controls output to the second file. If `S21` is zero, output to the first file is suppressed; if `S21` is one, output to the first file is generated. If `S22` is zero, output to the second file is suppressed; if `S22` is one, output to the second file is generated.

Any attempt to send output to an output file not explicitly or implicitly specified on the command line is silently ignored.

There is no restriction on the length of an output line, save that imposed by the operating system.

A listing of the output from ML/I may be directed to the listing file specified in the call of ML/I. Listing is controlled by the value of S20. If S20 is zero, no listing is produced at all. If S20 is one, a listing without line numbers is generated; if S20 is two, line numbers are included in the listing. S20 has an initial value of zero.

N.2.3 Workspace

ML/I uses a workspace area which is allocated at the start of the run. Its size is controlled by the -w flag. The default is 5000 words.

N.3 Character set

The character set used by ML/I is 8-bit ASCII (codes from 0 to 255 decimal). Since all possible codes are used, the error character is never used.

N.4 Error handling

Error messages are output to the debugging file specified in the call of ML/I; this defaults to standard error (usually the user's terminal). With reference to Chapter 6 of the *ML/I User's Manual*, the number 2N (the maximum number of characters inserted into an error message without truncation) is 64.

A count of processing errors (i.e. occurrences of the word Error(s) on the debugging file) is maintained in S5. At the end of a process, ML/I checks this value; if it is nonzero, ML/I sets the shell "exit status" to 254, otherwise it is set to zero. This allows shell files to detect the success (or otherwise) of an ML/I process. Incidentally, an exit status of 255 is given if a fatal error caused ML/I to terminate the process prematurely.

An output lines limit is imposed on the debugging file, to curb excessive output from a process that has gone badly wrong. The limit is implemented by holding a quota of "lines left" in S12; if S12 ever goes negative, the process is aborted. S12 is initially 500, but may be changed by the user.

At the end of a process, a message of the form

    At end of process: *N* lines, *M* calls

is output to the debugging file. If S18 is nonzero, this message is preceded by a list of the currently defined constructions. S18 is initially set to zero.

All files are opened as soon as ML/I is entered. Failure to open any file causes an appropriate message to be output, and ML/I immediately exits.

N.4.1 Error messages

The following run-time messages are peculiar to this implementation. They may be followed by other, advisory, messages which are self-explanatory.

N.4.1.1 Too many lines to the debugging file

Message

    Debugging file lines quota exhausted

Description

The value of S12 (the quota of remaining lines allowed to the debugging file) has become negative.

System Action

The current process is aborted.

N.4.1.2 Illegal input stream

Message

    S10 has illegal value, viz n

Description

S10 has been set to the value n, which is either outside the range 0–5, or is associated with an input stream that was not specified in the call of ML/I. Note that this error may be caused by S23 (the revert stream) being set to an illegal value, and end of file then being reached on another input stream.

System Action

The current process is aborted.

N.4.1.3 Error on writing to output file

Message

    Error while writing to name file

Description

An error has occurred while writing to the file indicated by *name*.

System Action

The current process is aborted.

N.5 Integer calculations

The initial environment contains ten permanent variables, all set to zero. All integers in, or derived from, macro expressions should be less than 32767 in magnitude. Overflow is not detected, except in the case of division by zero, and its effect is undefined.

N.6 Layout keywords

The following are the layout keywords for this implementation:

`SPACE`	meaning a space.
`NL`	meaning a newline.
`TAB`	meaning a tab.
`SL`	meaning the imaginary startline character.
`SPACES`	meaning a sequence of one or more spaces.

N.7 S-variables

There are 23 system variables. `S1` to `S9` are independent of the implementation, and are used to control and monitor ML/I itself. `S10` to `S23` are implementation dependent, and are used to control input/output, etc. If an S-variable is set to any value other than those given below, the effect is undefined (except for invalid values of `S10`, which always cause the process to be terminated).

N.7.1 Use of S1–S9

`S1`   If `S1` is one, the imaginary startline character is inserted on input. If `S1` is zero, no startlines are inserted; this is the initial setting.

`S2`   The current source text line number is held in `S2`; it may be changed at any time.

`S3`   If `S3` is one, the error message normally generated if a warning marker is not followed by a macro name is suppressed. If `S3` is zero (the initial value), the message is produced.

`S4`   If `S4` is one, the context print-out normally given after a call of `MCNOTE` is suppressed. If `S4` is zero, the context print-out is given; this is the initial setting.

`S5`   Count of processing errors.

`S6`   Not currently used.

`S7`   Not currently used.

`S8`   Not currently used.

`S9`   Not currently used.

N.7.2 Use of S10–S26

`S10`   Controls input selection; a value of zero forces end of all input. Values between 1 and 5 select the appropriate input stream.

Values of `S10` other than those given above (apart from zero) will cause an error. The initial value of `S10` is 1.

S11 Not currently used.

S12 `S12` contains the quota of lines on the debugging file. It is initially 500, and every time ML/I outputs a line to the debugging file (whether via an error message or a `MCNOTE`) it decreases `S12` by one. If `S12` ever becomes negative, the process is aborted. The user is at liberty to adjust the value of `S12` at any time.

S13 Not currently used.

S14 Not currently used.

S15 Not currently used.

S16 Used to control character code translation. Characters with the code given by `S16` are translated to characters with the code given by `S17`, on input. Initially `S16` is −1, so no translations are performed.

S17 See `S16` above.

S18 If `S18` is nonzero at the end of a process, a list is given of all currently defined constructions. The list isoutput to the debugging file (and is not subject to the quota of lines imposed by `S12`).

S19 The current line number of the output text is held in `S19`. It may be changed if desired.

S20 The value of `S20` controls output to the listing file. See Section N.2.2 for details.

S21 The value of `S21` controls output to the first output file; a value of zero suppresses output, and a value of one allows it. Its initial value is one.

S22 The value of `S21` controls output to the second output file; a value of zero suppresses output, and a value of one allows it. Its initial value is zero.

S23 `S23` contains the current revert stream. See Section N.2.1 for details.