IEEE Standards Interpretations for IEEE Std 1003.2™-1992 IEEE Standard for Information Technology--Portable Operating System Interfaces (POSIX®)--Part 2: Shell and Utilities

IEEE Standards Department, Copyrights and Permissions, 445 Hoes Lane, Piscataway, New Jersey 08855-1331, USA

Interpretation Request #123

Topic: MAKEFLAGS processing in make **Relevant Clauses:** 6.2.3, 6.2.5.3, 6.2.7.2

Issue 1: From subclause 6.2.5.3, lines 296-305, it says that "The MAKEFLAGS variable shall be accessed from the environment before the makefile is read. At this time, all of the options ... not already included in 'MAKEFLAGS' shall be added to the 'MAKEFLAGS' macro." This assignment of the 'MAKEFLAGS' macro is not referenced in 6.2.7.4, lines 482-490, where it describes macro assignments, so it is unspecified what the relationship is between this macro assignment and assignments from other different sources in terms of if/when this macro can be replaced from another source.

Q1: what is the precedence for this macro assignment with respect to precedence list lines 485-490? [Note:.2b/D10.9 already contains wording changes for lines 482-490 but this case is not handled there either] The 'MAKEFLAGS' macro created in this circumstance, should be between (1) and (2) (between line 485 and 486) and add to beginning of line 308: "The value of the MAKEFLAGS environment variable shall not be used as the 'MAKEFLAGS' macro." [Rationale: we want MAKEFLAGS macro to contain all the options from the command line and from MAKEFLAGS env. variable. and we don't want the env. variable overriding this new value We do want to override this value if 'MAKEFLAGS' macro is defined in the makefile. So, putting it before (2) is necessary to avoid problems with the reversal of (2) and (3) when -e is specified, and ensuring MAKEFLAGS env. variable is not turned into a macro ensures that environment variables in (2) do not override.]

Issue 2: From lines 298-300 "At this time, all of the options (except -f and -p) and command line macros not already included in 'MAKEFLAGS' shall be added to the 'MAKEFLAGS' macro."

Q2: The question is why are command line macros, and macros from the MAKEFLAGS environment variable, being added to the ‘MAKEFLAGS’ macro? Was this non-historical behaviour intentional? a) This seems inconsistent with lines 377-379 which describes the creation of the MAKEFLAGS variable containing all options on the command line except of the -f and -p options. Why are command line macros not included here too? b) This behaviour has never been true in any historical implementation. There seems to be good reason for *not* doing this. For example, given a command line of make CFLAGS=”-s -g” according to line 298-300, there should be a ‘MAKEFLAGS’ macro created which is then put into the environment equivalent to this: MAKEFLAGS=”CFLAGS=-s -g” but now when a 2nd level make gets invoked from a makefile command, it will see inherit this MAKEFLAGS environment and will interpret it as one 1 macro “CFLAGS=- s” and 1 option “-g”. Since “-g” is not a valid option, the 2nd level make will exit with an error message. The rationale did not give any reason for this new behaviour and its not clear what advantage this provides since macros are automatically included in the environment and vice versa which means 2nd level make’s will automatically inherit these values. Remove “and command line macros” from line 298.

Interpretation Response
Issue 1: The standard does not speak to this issue, and as such no conformance distinction can be made between alternative implementations based on this. This is being referred to the sponsor.

Issue 2: The text on page 669, lines 296-305, and page 571, lines 377-380 are in conflict and as such the standard is unclear on this issue, and no conformance distinction can be made between alternative implementations based on this. This is being referred to the sponsor.

Rationale for Interpretation
None.