

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

Copyright © 1996 by the Institute of Electrical and Electronics Engineers, Inc. 3 Park Avenue New York, New York 10016-5997 USA All Rights Reserved.

Interpretations are issued to explain and clarify the intent of a standard and **do not** constitute an alteration to the original standard. In addition, interpretations are not intended to supply consulting information. Permission is hereby granted to download and print one copy of this document. Individuals seeking permission to reproduce and/or distribute this document in its entirety or portions of this document must contact the IEEE Standards Department for the appropriate license. Use of the information contained in this document is at your own risk.

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

Interpretation Request #80

Topic: awk **Relevant Clauses:** 4.1

Question 1: Awk POSIX.2, p.167, 4.1.7.1 last paragraph In IEEE 1003.2-1992 Subclause 4.1.7.1, the execution of an awk program is described: "(...) Then each file operand (...) will be processed in turn (...) until a record separator is seen (...), splitting the current record into fields using the current value of FS according to the rules in (...), evaluating each pattern in the program in the order of occurrence, and executing the action associated with each pattern that matches the current record. (...)". Existing practice has always been to implement splitting into fields in a lazy fashion. If only \$0 was needed, no splitting occurs. If awk must be changed, the obvious modification will cause a noticeable slow-down since all lines will split regardless of whether they will be processed. Was the change from existing practice intentional?

Question 2: Awk POSIX.2, p.164, 4.1.4 POSIX.2 Subclause 4.1.4 states: "Thus, an assignment before the first file argument shall be executed after the BEGIN actions (if any),..." Existing practice has been to make the command line variable assignments all at once before the BEGIN. Was this change intentional?

Interpretation Response

Question 1: The standard clearly states the effects expected from awk and a conforming implementation must just support the effects expected from awk. It does not define the implementation and an implementation could still continue to split fields in a lazy fashion.

Question 2: The standard states the behavior for the assignment argument for awk, and conforming implementations must conform to this. However, concerns have been raised

about this which are being referred to the sponsor.

Rationale for Interpretation

None.