

## IEEE Standards Interpretation for IEEE Std 1003.1™-2001 IEEE Standard Standard for Information Technology -- Portable Operating System Interface (POSIX®)

Copyright © 2006 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 #98

**Topic:** non-whitespace IFS **Relevant Sections:** XCU 2.6.5 Field Splitting

Discussions on the Austin Group mailing list of how non-whitespace IFS characters are used as delimiters have identified the need for some improvements to the description of IFS and field splitting that would help prevent misinterpretation, plus a correction to the description of the read utility where it refers to IFS characters as separators.

The new descriptions clarify the intended behaviour whereby the IFS characters are used to terminate fields, as confirmed in POSIX.2 interpretation pasc-1003.2-98.

On page 35 line 1427 section 2.5.3 Shell Variables:

Delete "(Input Field Separators.) " from the beginning of the IFS description. (Note that this change should be done after the change in Austin Interpretation AI 096.)

On page 42 line 1729 section 2.6.5 Field Splitting, paragraph 2:

Change "The shell shall treat each character of the IFS as a delimiter and use the delimiters to split the results of parameter expansion and command substitution into fields."

to "The shell shall treat each character of the IFS as a delimiter and use the delimiters as field terminators to split the results of parameter expansion and command substitution into fields."

On page 817 line 31778-31780 section read:

Change “If there are fewer var operands specified than there are fields, the leftover fields and their intervening separators shall be assigned to the last var. If there are fewer fields than vars, the remaining vars shall be set to empty strings.”

to “If there are fewer fields than there are var operands, the remaining vars shall be set to empty strings. If there are fewer vars than fields, the last var shall be set to a value comprising the following elements:

\* the field that corresponds to the last var in the normal assignment sequence described above, \* the delimiter(s) that follow the field corresponding to the last var, and \* the remaining fields and their delimiters, with trailing IFS white space ignored.”

On page 853 line 33116-33120 section sh under ENVIRONMENT VARIABLES:

Replace the description of IFS with a copy of the one from section 2.5.3 after the change specified above has been applied to it. (This will do more than just delete “(Input Field Separators.)”, since Austin Interpretation AI 096 will also have been previously applied to 2.5.3.)

On page 865 line 33622 section sh under RATIONALE:

Insert the following before “The KornShell ignores the contents of IFS” (as a new first sentence of that paragraph):

“The name IFS was originally an abbreviation of Input Field Separators, however this name is misleading as the IFS characters are actually used as field terminators.”

### **Interpretation Response**

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.