

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 #96

Topic: shell IFS unset **Relevant Sections:** XCU 2.5.3 IFS

The description of the IFS variable states:

(Input Field Separators.) A string treated as a list of characters that is used for field splitting and to split lines into fields with the read command. If IFS is not set, the shell shall behave as if the value of IFS is <space>, <tab>, and <newline>; see Section 2.6.5 (on page 42). Implementations may ignore the value of IFS in the environment at the time the shell is invoked, treating IFS as if it were not set.

There are two problems with this text.

Firstly, the “shall behave as if” part needs to specify which aspects of the shell’s behaviour it applies to. It is loosely implied by the context that it only applies to field splitting and the read command, but the text should make clear that only these aspects are affected, and not such things as the evaluation of `${IFS-unset}` or the value of IFS reported by a plain “set” command.

Secondly, the “treating IFS as if it were not set” part does not match existing practice, which is to set IFS to <space><tab><newline>, not to unset it (nor even to leave it unset if it isn’t set in the environment).

Also, as an editorial comment, I think “<space><tab><newline>” better represents the concatenation of those characters than does “<space>, <tab>, and <newline>”.

Replace the description of IFS with the following:

(Input Field Separators.) A string treated as a list of characters that is used for field splitting and to split lines into fields with the read command. If IFS is not set, field splitting by the shell and line splitting by the read command shall be performed as if the value of IFS is <space><tab><newline>; see Section 2.6.5 (on page 42). Other uses of IFS shall behave as normal for an unset variable, for example “\${IFS-unset}” expands to “unset”.

Implementations may ignore the value of IFS in the environment, or the absence of IFS from the environment, at the time the shell is invoked, in which case the shell shall set IFS to <space><tab><newline> when it is invoked.

Interpretation Response

For the first case 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.

In the second case, the standards states the requirements for IFS, 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.