

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 #74

Topic: `makecontext()` `uc_link` **Relevant Sections:** XSH `makecontext()` Page: 742 Line: 24473

The `makecontext()` description provides the only real detail about the `uc_link` element of `ucontext_t`. In the `getcontext()` man page it is only said that if the pointer is NULL this is the main thread etc.

Now, the complete text of the `uc_link` description is this: The `uc_link` member is used to determine the context that shall be resumed when the context being modified by `makecontext()` returns. The application shall ensure that the `uc_link` member is initialized prior to the call to `makecontext()`.

There is no word on how the `ucontext_t` structure pointed to by `uc_link` has to be initialized. Some people try to interpret this as if it allows any kind of initialization. From the text in `getcontext()` it is clear, though, that this is not intended: The `ucp` argument should be created either by a prior call to `getcontext()` or `makecontext()`, or by being passed as an argument to a signal handler

In the case of `uc_link` the signal handler part goes away but it is possible to use `swapcontext()`. One problem is that the `ucontext_t` structure only needs to be fully initialized at the time the link is used. I suggest adding this requirement to the `uc_link` description.

Add at the end of line 24475: If `uc_link` is not null the structure pointed to by it at the time the pointer is used must be initialized by a prior call to `getcontext()`, `makecontext()`, or `swapcontext()`.

Interpretation Response #74

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.

Rationale for Interpretation

None.