

Replaces

ISO/IEC JTC 1/SC 22
Programming Languages

Document Type: Summary of Voting/Table of Replies

Document Title: Summary of Voting for SC 22 N 4217, FCD 9945, Information technology – Portable Operating System Interface (POSIX)

Document Source: SC 22 Secretariat

Project Number:

Document Status: This document is circulated to SC 22 members for information. This document has passed FCD ballot, and the PAG is instructed to review comments received, prepare a disposition of comments to them, prepare the text for publication. Once these documents have been prepared, the PAG is instructed to send them to the SC 22 Secretariat for submission to ITTF for FDIS balloting.

Action ID: ACT

Due Date:

No. of Pages: 134

Committee: ISO/IEC JTC 1/SC 22, Programming languages

Ballot Number: N4217

Ballot Title: FCD 9945, Information technology – Portable Operating System Interface (POSIX)

Source: JTC 1/SC 22 Secretary

National Body	Approval of the draft	Approval of the draft with comments	Disapproval with comments	Abstain	Comments
Austria					
Canada		x			See below
China	x				
Czech Republic	x				
Denmark					
France					
Germany	x				
Italy	x				
Japan	x				
Republic of Korea	x				
Netherlands				x	
Romania	x				
Russian Federation					
Spain				x	
Switzerland				x	
Ukraine					
United Kingdom		x			See below
United States		x			See below

National Body Comments

Canada

Canada notes that there are a number of significant technical comments to the parallel ballot that was held by IEEE and that these comments are being included with the US NB submission. Canada supports the approval of the FCD with the inclusion of those comments.

United Kingdom

We recommend the changes identified by the US national body be made to the draft, together with all approved interpretations/defect reports against the current standards (SIO/IEC 9945: 2003 Parts 1-4, with TC1).

LX/07-0008

October 18, 2007

RECOMMENDED U.S. COMMENTS ON FCD 9945 POSIX

INCITS Study Group on Linux

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 45 Line: 1424 Section: 3.84		te	<p>Some of the functions added in the revision need to be included in the list of byte input/output functions.</p> <p>Also, perror() seems to have found its way into the list somehow, but it should not be there.</p> <p>Since psiginfo() and psignal() are counterparts to perror(), presumably they should not be added to the list, but should have the same requirement as perror() not to change the stream orientation.</p>	<p>Add getdelim() and getline() to the list.</p> <p>Remove perror() from the list.</p> <p>Cross-volume change to XSH: page 1491 line 48541 section psiginfo, add a new paragraph:</p> <p style="padding-left: 40px;">"The psiginfo() and psignal() functions shall not change the orientation of the standard error stream."</p>	
US	Page: 64 Line: 1964 Section: 3.225		ed	<p>XBD ERN 74 has not been correctly applied. The new text was supposed to use the defined term "real time", not the word "realtime".</p> <p>The aardvark also asked for a change from "realtime" to "real time" in two other places to be considered. The AAM comment says "take the changes as proposed below", which I believe was supposed to include those two changes.</p>	<p>Change from</p> <p>A realtime clock whose value cannot be set via clock_settime() and which cannot have negative clock jumps.</p> <p>to</p> <p>A clock measuring real time whose value cannot be set via clock_settime() and which cannot have negative clock jumps.</p> <p>CROSS VOL XSH On clock_getres() in DESCRIPTION para 4</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

				<p>change from</p> <p>This clock represents the realtime clock for the system.</p> <p>to</p> <p>This clock represents the clock measuring real time for the system.</p>		
US	<p>Page: 67</p> <p>Line: 2039</p> <p>Section: 3.243</p>		TE	<p>The definition of "null pointer" is in conflict with the C Standard.</p> <p>Also there are various uses of "NULL pointer" which should be the defined term "null pointer".</p> <p>The first change is in scope because it is a conflict with the C Standard. The second change should be considered an editorial matter, needed for consistency.</p> <p>There are various other uses of "NULL" where the intended meaning is "a null pointer", such as "if the xxx argument is NULL" or "if the xxx argument is not NULL". However, I don't think it is worth trying to fix these. (It would not be safe to try and do global changes; they would have to be examined individually.) It is reasonably clear that the intended meaning is "if xxx compares (un)equal to NULL", rather than the literal "if xxx is (not) the NULL macro".</p>	<p>Change</p> <p>"The value that is obtained by converting the number 0 into a pointer; for example, (void *) 0. The C language guarantees that this value does not match that of any legitimate pointer, so it is used by many functions that return pointers to indicate an error."</p> <p>to</p> <p>"A pointer obtained by converting an integer constant expression with the value 0, or such an expression cast to type void *, to a pointer type; for example, (char *)0. The C language guarantees that a null pointer compares unequal to a pointer to any object or function, so it is used by many functions that return pointers to indicate an error."</p> <p>Globally (in all volumes) change "NULL pointer" to "null pointer".</p> <p>Also add the following note</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					Note: An issue has been raised with the ISO C committee regarding whether an integer constant expression 0 should be considered a null pointer if it is not cast to a pointer type.	
US	Page: 206 Line: 7089 Section: aio.h		GE	<p>The <aio.h> header uses several types without any requirement that they be declared. Likewise for <mqueue.h>.</p> <p>The draft 3 edits in <signal.h> for struct sigevent introduced a typo. Meanwhile, the use of pthread_attr_t needs a declaration, although it can be an incomplete type.</p> <p><time.h> also uses struct sigevent, but since timer_create explicitly requires the inclusion of <signal.h> (line 65309; unlike lio_listio, line 29611, or mq_notify, line 42861), the inclusion of just <time.h> should be able to leave out union sigval and pthread_attr_t by leaving struct sigevent incomplete.</p>	<p>Add prior to line 7099:</p> <p>The <aio.h> header shall define the following types: off_t, pthread_attr_t, size_t, and ssize_t As described in <sys/types.h>. struct timespec As described in <time.h>.</p> <p>The tag sigevent shall be declared as naming an incomplete structure type, the contents of which are described in the header <signal.h>.</p> <p>Add prior to line 9692:</p> <p>The <mqueue.h> header shall define the following types: pthread_attr_t, size_t, and ssize_t As described in <sys/types.h>. struct timespec As described in <time.h>.</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

The tag `sigevent` shall be declared as naming an incomplete structure type, the contents of which are described in the header `<signal.h>`.

Replace the start of line 9692:

The `<mqueue.h>` header shall define the `sigevent` structure (as described in `<signal.h>`) and the `mq_attr` structure,...

with:

The `<mqueue.h>` header shall define the `mq_attr` structure,...

In `signal.h` Page 312 before I 10806
Add this line with CX shading:

The `<signal.h>` header shall define the `pthread_attr_t` type, as described in `<sys/types.h>`.

Replace on line 10811:

(union `signal`)

with:

(union `sigval`)

In `time.h` p 404 I 13913

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					add this line with CX shading: The tag sigevent shall be declared as naming an incomplete structure type, the contents of which are described in the header <signal.h>.	
US	Page: 224 Line: 7704 Section: fcntl.h		ed	The fcntl.h page says SEEK_CUR etc. are defined "as described in <unistd.h>", but the unistd.h page no longer describes them - it says that <unistd.h> defines them "as described in <stdio.h>".	Change "unistd.h" to "stdio.h".	
US	Page: 224 Line: 7712 Section: <fcntl.h>		ed	fcntl.h() should be <fcntl.h>.	Change fcntl.h() to <fcntl.h> (and italics to bold).	
US	Page: 247 Line: 8410 Section: inttypes.h		ed	EXAMPLES sections are informative, not normative.	Add non-normative shade side-bars to these EXAMPLES sections: <inttypes.h> lines 8410-8419 <stdarg.h> lines 11223-11245	
US	Page: 249 Line: 8485 Section:		te	The addition of nl_langinfo_l to langinfo.h, strfmon_l to monetary.h, strcasecmp_l to strings.h, and strftime_l to time.h, each requires the definition of	Add at line 8485: The locale_t type shall be defined as	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

langinfo.h

locale_t.

Additionally, pid_t is not defined in time.h, required for the CPT shaded clock_getcpuclockid.

described in <locale.h>.

Add after line 9660:

locale_t As described in <locale.h>.

Add at 11970:

The locale_t type shall be defined as described in <locale.h>.

Because of shading issues, rewrite the paragraph at line 13913:

The clock_t, size_t, time_t, <CX>clockid_t, and timer_t</CX> types shall be defined as described in <sys/types.h>.

as the following list:

The <time.h> header shall define the following types:

clock_t As described in <sys/types.h>.

<CX>clockid_t As described in <sys/types.h>.</CX>

<CX>locale_t As described in <locale.h>.</CX>

<CPT>pid_t As described in <sys/types.h>.</CPT>

size_t As described in <sys/types.h>.

time_t As described in <sys/types.h>.

<CX>timer_t As described in <sys/types.h>.</CX>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 267 Line: 9247 Section: locale.h		ge	<p>Most of the headers specified by C, but with CX/XSI shaded extensions required by POSIX 200x, warn the reader that a feature test macro must be used to ensure the POSIX definitions are visible. However, this was not done for locale.h.</p> <p>As this text was not edited in draft 3, it should be considered in light of making the standard internally consistent.</p>	<p>At line 9247, replace:</p> <p><CX>Some of the functionality described on this reference page extends the ISOC standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1-200x defers to the ISO C standard.</CX></p> <p>with:</p> <p><CX>Some of the functionality described on this reference page extends the ISO C standard. Applications shall define the appropriate feature test macro (see XSH Section 2.2, on page 448) to enable the visibility of these symbols in this header.</CX></p>	
US	Page: 312 Line: 10801 Section: signal.h		te	<p><signal.h> refers to the following types which need declaration:</p> <p>uid_t (CX-shaded struct siginfo_t) pthread_t (CX-shaded pthread_kill) struct timespec (CX-shaded sigtimedwait)</p>	<p>In line 10801 change from</p> <p>The <signal.h> header shall define the following data types through typedef:</p> <p>to</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

				<p>Note that the only use of struct timespec in <signal.h> is via a pointer, and that while implementations are allowed to bring in symbols from <time.h>, they are not required to do so.</p> <p>Additionally, the mention of size_t on 10926, resulting from SD5-XBD-ERN-56, is out of place compared to the mention of other typedefs copied from other locations, not to mention that it is XSI-shaded although stack_t is CX-shaded.</p>	<p>The <signal.h> header shall define the following data types:</p> <p>At line 10801, add the following lines (optionally, consider sorting the existing lines):</p> <pre><CX>pthread_t As described in <sys/types.h>.</CX> <CX>size_t As described in <sys/types.h>.</CX> <CX>struct timespec As described in <time.h>.</CX> <CX>uid_t As described in <sys/types.h>.</CX></pre> <p>Delete line 10926:</p> <pre><XSI>The size_t type shall be defined as described in <sys/types.h>.</XSI></pre>
US	Page: 314 Line: 10900-10906 Section: signal.h		ed	Missing CX shading.	<pre>Add CX shading to SA_RESETHAND, ..., SA_NODEFER.</pre>
US	Page: 320 Line: 11103 Section: spawn.h		te	The <spawn.h> header uses several types without declarations. In the case of PS shading, this includes sched_param; which in turn points out issues in <sched.h>. Note that in <sched.h>, a complete struct timespec is required by SS and TSP shading, while an incomplete type will satisfy PS and TPS shading; the proposal merely goes	<p>sched.h p305</p> <p>At line 10596, add:</p> <p>The <sched.h> header shall define the following types:</p> <pre><PS>pid_t As described in</pre>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				with the simpler approach of always requiring struct timespec, while only requiring time_t where a complete timespec is needed.	<p><sys/types.h>.</PS> <SS TSP>time_t As described in <sys/types.h>.</SS TSP> struct timespec As described in <time.h>.</p> <p>spawn.h At line 11103, add:</p> <p>The <spawn.h> header shall define the following types: mode_t As described in <sys/types.h>. pid_t As described in <sys/types.h>. sigset_t As described in <signal.h>.</p> <p>The tag sched_param shall be declared as naming an incomplete structure type, the contents of which are described in the header <sched.h>.</p> <p>Also ensure that <sys/types.h> does not occur in a may make visible phrase in the headers as globally permitted already.</p>	
US	Page: 335 Line: 11629		te	The description of stdio.h was edited in draft 3 to mention that, as a C extension, the type ssize_t is	At line 11628, add, with CX shading:	

¹ **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

² **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

	Section: stdio.h			available (required by the declaration of CX getline). However, off_t was overlooked (required by the declaration of CX fseeko).	off_t As described in <sys/types.h>. Optionally, consider sorting the list, moving va_list after ssize_t.	
US	Page: 342 Line: 11893 Section: string.h		te	The addition of strcoll_l and other functions to string.h requires the definition of locale_t with CX shading. Likewise for wchar.h and wctype.h, along with a typo correction.	After line 11893, add, with CX shading: The <string.h> header shall define the locale_t type as described in <locale.h>. After wchar.h p435 line 15200, add, with CX shading: locale_t As described in <locale.h>. At wctype.h line 15383, replace: The <ctype.h> header... with: The <wctype.h> header... Sort lines 15189 – 15200.	
US	Page: 368 Line: 12695 Section: sys/socket.h		ed	The <sys/socket.h> header requires ssize_t, but not size_t.	At line 12695, replace: The ssize_t type shall be defined... with: The size_t and ssize_t types shall be defined...	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 385 Line: 13302 Section: sys/un.h		ge	The APPLICATION USAGE section of the sys/un.h page needs to be updated for consistency with the change of _POSIX_PATH_MAX from 255 to 256. It should also say "bytes" not "characters".	Change "_{_POSIX_PATH_MAX} characters (255)" to "_{_POSIX_PATH_MAX} bytes (256)"	
US	Page: 397 Line: 13666 Section: termios.h		te	The <termios.h> header uses pid_t without declaration.	Prior to line 13666, add the line: The <termios.h> header shall define the pid_t type, as described in <sys/types.h>.	
US	Page: 404 Line: 13915 Section: getdate_err		te	There are no requirements on getdate_err, other than the implied requirements in ERRORS of getdate() that it can have 8 distinct values. The description should be made consistent with the edits made in draft 3 to errno (line 7557), another example of a global variable/macro that can hold error status.	Fix errno.h Change on I7557 The <errno.h> header shall provide a declaration for errno. to The <errno.h> header shall provide a declaration or definition for errno. Replace line 13915: The <time.h> header shall provide a declaration for getdate_err.	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

with the following, XSI shaded:

The `<time.h>` header shall provide a declaration or definition for `getdate_err`. The symbol `getdate_err` shall expand to an expression of type `int`. It is unspecified whether `getdate_err` is a macro or an identifier

declared with external linkage, and whether or not it is a modifiable lvalue. If a macro definition is suppressed in order to access an actual object, or a program defines an identifier with the name `getdate_err`, the behavior is undefined.

Replace line 32873:

The external variable or macro `getdate_err` is used by `getdate()` to return error values.

with:

The external variable or macro `getdate_err`, which has type `int`, is used by `getdate()` to return error values. It is unspecified whether `getdate_err` is a macro or an identifier declared with external linkage, and whether or not it is a modifiable lvalue. If a macro definition is suppressed in order to access an actual object, or a program defines an identifier with the name `getdate_err`, the behavior is undefined.

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 438 Line: 15339 Section: wchar.h		ed	Cross-volume links for open_wmemstream are missing.	XBD wchar.h: In the list at line 15339, add a link to open_wmemstream(). XSH open_wmemstream: In the list at line 44839, add a link to <wchar.h>. [[NB open_wmemstream is a pointer page so link should go to the real page]]	
US	Page: 0 Line: 0 Section: restrict		te	Some of the functions added in API Set 1 are missing restrict modifiers. Since not all of these functions were modified in draft 3, this should be considered in light of consistency with the rest of the standard. Note also that fmemopen, also added in Set 1, does use restrict). The proposal does not modify alphasort (and is careful in the modification of scandir), on the grounds that comparing a struct dirent with itself should be allowed (and returns 0). The fact that glibc declares alphasort as taking const void* instead of const struct dirent** is unrelated to this aardvark, but is worth considering as well, as is the	At line 11005 (XBD signal.h), change: void psiginfo(siginfo_t *, const char *); to: void psiginfo(const siginfo_t *, const char *); At line 11634 (XBD stdio.h), change: int dprintf(int, const char *, ...) to:	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

question as whether alphasort can/should set errno to EILSEQ in some locales.

The proposal does not modify fexecve, on the grounds that none of the other exec functions use restrict.

The proposal does not modify linkat, on the grounds that link also does not use restrict.

The proposal does not modify renameat, on the grounds that rename also does not use restrict.

The proposal does not modify symlinkat, on the grounds that symlink also does not use restrict.

int dprintf(int, const char *restrict, ...)

[Note - if vdprintf is added, as is the subject aardvark XSHd3 #4 (ERN 32), it would also need restrict.]

At line 11666-11667 (XBD stdio.h), change:

```
ssize_t getdelim(char **, size_t *, int, FILE *);  
ssize_t getline(char **, size_t *, FILE *);
```

to:

```
ssize_t getdelim(char **restrict, size_t  
*restrict, int, FILE *restrict);  
ssize_t getline(char **restrict, size_t *restrict,  
FILE *restrict);
```

At lines 15287-15288 (XBD wchar.h), change:

```
size_t wcsnrtombs(char *, const wchar_t **,  
size_t, size_t, mbstate_t *);
```

to:

```
size_t wcsnrtombs(char *restrict, const  
wchar_t **restrict, size_t, size_t, mbstate_t  
*restrict);
```

At lines 24357 (XSH dprintf) and 29233 (XSH fprintf), change:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>int dprintf(int filedes, const char *format, ...);</p> <p>to:</p> <p>int dprintf(int filedes, const char *restrict format, ...);</p> <p>[Note - if vdprintf is added, as is the subject aardvark XSHd3 #4 (ERN 32), it would also need restrict.]</p> <p>At lines 33041-33043 (XSH getdelim) and 33734 (XSH getline), change:</p> <p>ssize_t getdelim(char **lineptr, size_t *n, int delimiter, FILE *stream); ssize_t getline(char **lineptr, size_t *n, FILE *stream);</p> <p>to:</p> <p>ssize_t getdelim(char **restrict lineptr, size_t *restrict n, int delimiter, FILE *restrict stream); ssize_t getline(char **restrict lineptr, size_t</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>*restrict n, FILE *restrict stream);</p> <p>At line 48532 (XSH psiginfo), change:</p> <pre>void psiginfo(siginfo_t *pinfo, const char *message);</pre> <p>to:</p> <pre>void psiginfo(const siginfo_t *, const char *message);</pre> <p>At lines 68259-68260 (XSH wcsnrtombs) and 68329-68330 (XSH wcsrtombs), change:</p> <pre>size_t wcsnrtombs(char *dst, const wchar_t **src, size_t nwc, size_t len, mbstate_t *ps);</pre> <p>to:</p> <pre>size_t wcsnrtombs(char *restrict dst, const wchar_t **restrict src, size_t nwc, size_t len, mbstate_t *restrict ps);</pre>	
US	Page: 369 Line: 12784 Section: sys/socket.h		ed	Because MSG_NOSIGNAL does not accept arguments, it should not be followed by "(" or be in italics (typographical conventions, page xxxv line 1468). It should match the corresponding text on pages 1803, 1806, and 1809.	Change to XBD Change "The MSG_NOSIGNAL() macro" to "The MSG_NOSIGNAL symbolic constant" and remove italics.	
US	Page: 450 Line: 15684 Section:		te	The namespace tables in section 2.2.2 have some inconsistencies regarding header nesting. Firstly some prefixes have moved from one table to the	In the first table (pages 450 and 451): Delete the "in_" prefix from the	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
	2.2.2			other for their "primary" header, but still appear in the old place in the entry for another header that is allowed to make symbols visible from the primary header. Secondly, only a few headers have entries corresponding to the prefixes (or suffixes or complete names) for other headers from which they can make symbols visible. In most cases there is no explicit entry in the table - matching symbols are allowed by virtue of the statement on the header's main page in XBD that it may make symbols from the other header visible. It would be best to remove the few such prefixes etc. that do appear explicitly.	<p><arpa/inet.h> entry.</p> <p>Delete the <inttypes.h> entry.</p> <p>Delete the "si_" prefix from the <sys/wait.h> entry.</p> <p>In the second table (page 452):</p> <p>Delete the "S_" and "SEEK_" prefixes from the <fcntl.h> entry.</p> <p>Delete the <sys/wait.h> entry.</p>	
US	Page: 450 Line: 15703 Section: 2.2.2		te	There are some shading problems in the table on pages 450 and 451. The first is editorial; the second is for consistency with interfaces that have moved from XSI to Base.	Unshade the <netinet/tcp.h> and <nl_types.h> entries. Unshade the <poll.h>, <sched.h> and <sys/wait.h> entries.	
US	Page: 461 Line: 16136 Section: 2.3		te	Two additions are needed to section 2.3 Error Numbers for consistency with the addition of ENOTRECOVERABLE and EOWNERDEAD to errno.h and the pthread_mutex_lock() page.	After line 16136 add: "[ENOTRECOVERABLE] State not recoverable. The state protected by a robust	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>mutex is not recoverable."</p> <p>After line 16156 add:</p> <p>"[EOWNERDEAD] Previous owner died. The owner of a robust mutex terminated while holding the mutex lock."</p>
US	Page: 467 Line: 16388 Section: 2.4.3		te	<p>Section 2.4.3 Signal Actions needs to be updated for consistency with the changes made to the <signal.h> and sigaction pages to move some members of siginfo_t, and the meaning of si_code <= 0, from XSI to Base.</p> <p>While merging in the new description of si_code, I noticed a problem in that SI_USER and SI_QUEUE are cases where the signal is sent by a process. This overlaps with the (formerly XSI) requirement that si_code is <= 0 when the signal is from a process. Current XSI systems handle this by defining SI_USER and SI_QUEUE with values <= 0. If we extend this requirement to non-XSI systems it will break the ABI of any implementations where SI_USER and/or SI_QUEUE are positive. However, if we allow SI_USER and SI_QUEUE to be positive it will break any XSI applications that just check whether si_code is <= 0. The only solution I can see is to have an explicit XSI requirement that SI_USER and SI_QUEUE are <= 0. Non-XSI POSIX applications would check whether the signal is from a process by seeing if si_code is either SI_USER or SI_QUEUE or a value</p>	<p>Change the table from:</p> <pre> int si_signo Signal number. int si_code Cause of the signal. union sigval si_value Signal value. </pre> <p>to:</p> <pre> int si_signo Signal number. int si_code Cause of the signal. pid_t si_pid Sending process ID. uid_t si_uid Real user ID of sending process. void * si_addr Address of faulting instruction. int si_status Exit value or signal. union sigval si_value Signal value. </pre> <p>At line 16393 change:</p> <p>"The following values are defined for si_code"</p> <p>to</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<= 0.	<p>"The following non-signal-specific values are defined for si_code"</p> <p>At line 16402 change:</p> <p>"If the signal was not generated by one of the functions or events listed above, the si_code shall be set to an implementation-defined value that is not equal to any of the values defined above."</p> <p>to:</p> <p>"Signal-specific values for si_code are also defined, as described in [xref to <signal.h> page].</p> <p>If the signal was not generated by one of the functions or events listed above, si_code shall be set either to one of the signal-specific values described in [xref to <signal.h> page] or to an implementation-defined value that is</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

not equal to any of the values defined above.

If `si_code` is `SI_USER` or `SI_QUEUE`, [XSI] or any value less than or equal to 0 [XSI], then the signal was generated by a process and `si_pid` and `si_uid` shall be set to the process ID and the real user ID of the sender, respectively.

In addition, `si_addr`, `si_pid`, `si_status` and `si_uid` shall be set for certain signal-specific values of `si_code`, as described in [xref to <signal.h> page]."

On page 1871 line 59779 section sigaction, change:

"The `si_code` member contains a code identifying the cause of the signal.

If the value of `si_code` is less than or equal to 0, then the signal was generated by a process and `si_pid` and `si_uid`, respectively, indicate the process ID and the real user ID of the sender. The <signal.h> header description contains information about the signal-specific contents of the

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>elements of the siginfo_t type."</p> <p>to:</p> <p>"The si_code member contains a code identifying the cause of the signal, as described in [xref to section 2.4.3]."</p> <p>Cross-volume changes to XBD:</p> <p>page 318 line 11036 section signal.h, change APPLICATION USAGE from:</p> <p>"None."</p> <p>to:</p> <p>"On systems not supporting the XSI option, the si_pid and si_uid members of siginfo_t are only required to be valid when si_code is SI_USER or SI_QUEUE. On XSI-conforming systems, they are also valid for all si_code values less than or equal to 0; however, it is unspecified whether SI_USER and</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>SI_QUEUE have values less than or equal to zero and therefore XSI applications should check whether si_code has the value SI_USER or SI_QUEUE or is less than or equal to 0 to tell whether si_pid and si_uid are valid."</p>
US	<p>Page: 498 Line: 17677 Section: 2.10.7</p>		te	<p>Section 2.10.7 Socket I/O Mode needs to be updated for consistency with the new asynchronous feature of bind().</p>	<p>Before "The connect() function...", add new text as below as a new para. "The bind() function initiates an address assignment and shall return without blocking when O_NONBLOCK is set; if the socket address cannot be assigned immediately, bind() shall return the error [EINPROGRESS] to indicate that the assignment was initiated successfully, but that it has not yet completed."</p>
US	<p>Page: 499 Line: 17744 Section: 2.10.14</p>		te	<p>Various places in the text require the SIGPIPE signal to be generated without mentioning any exception for MSG_NOSIGNAL. Because the new MSG_NOSIGNAL flag "requests" not to send the signal (page 1802 line 57751) but the signal still "shall be sent" (page 499 line 17744 and others), a literal interpretation requires that the signal be sent regardless of this flag. Text that requires SIGPIPE generation should not require it in situations where the MSG_NOSIGNAL flag was used.</p> <p>SIGPIPE generation is mentioned in a number of places, including in the description of the SO_KEEPALIVE option on both the getsockopt() and setsockopt() pages. 2004 ERN 195 moves this</p>	<p>page 497 line 17640 section 2.10.6 Change from:</p> <p>A SIGPIPE signal is raised if a thread sends on a broken stream (one that is no longer connected).</p> <p>To:</p> <p>A SIGPIPE signal is raised if a thread attempts to send data on a broken stream (one that is no longer connected), except that the signal is suppressed if the MSG_NOSIGNAL flag is used in calls to send(), sendto(), and sendmsg().</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>text to section 2.10.16. The proposed action below removes the mention of SIGPIPE within the description of SO_KEEPALIVE since SIGPIPE is not triggered by SO_KEEPALIVE, but rather by a separate explicit send operation, if the connection is broken. However other sections already require this behavior for such send operations, independent of the SO_KEEPALIVE option. Besides being redundant and now incomplete, it is misleading to mention SIGPIPE here, since an application that performs regular sends has no need for SO_KEEPALIVE. SO_KEEPALIVE is used to prevent the connection from staying open indefinitely if the remote host crashes, and the local application would not normally send data except in response to data received. In this case enabling SO_KEEPALIVE allows the remote crash to eventually be detected by a return from read/recv/poll/select.</p>	<p>page 499 line 17744 section 2.10.14 Change from:</p> <p>The SIGPIPE signal shall be sent to a thread that attempts to send data on a socket that is no longer able to send. In addition, the send operation fails with the error [EPIPE].</p> <p>To:</p> <p>The SIGPIPE signal shall be sent to a thread that attempts to send data on a socket that is no longer able to send (one that is no longer connected), except that the signal is suppressed if the MSG_NOSIGNAL flag is used in calls to send(), sendto(), and sendmsg().</p> <p>Regardless of whether the generation of the signal is suppressed, the send operation shall fail with the error [EPIPE].</p> <p>After applying XSHbug2.txt ERN 195 (against 2004 edition) page 502 line 17846 section 2.10.16/SO_KEEPALIVE</p> <p>Delete:</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

and threads writing to that socket shall be/are notified with a SIGPIPE signal

page 1803 line 57786 section send
page 1805 line 57881 section sendmsg
page 1808 line 57999 section sendto
Change from:

and if the socket is of type SOCK_STREAM,

To:

if the socket is of type SOCK_STREAM or SOCK_SEQPACKET and the MSG_NOSIGNAL flag is not set,

Cross-volume change to XRAT:
page 3405 line 115930 section B.2.3
Change from:

This condition normally generates the signal SIGPIPE; the error is returned if the signal does not terminate the process.

To:

This condition normally generates the signal SIGPIPE; the error is returned if the generation of the signal is suppressed or the signal does not terminate the process.

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 524 Line: 18656 Section: _Exit		TE	<p>The _Exit() page has a list headed "Consequences of Process Termination" which lists rules that apply not only to termination by calling _Exit() and _exit() (the functions described on that page) but to all forms of process termination. Currently this only becomes apparent by noticing that there are references to the list from other parts of the standard that talk about process termination, or "as if" statements referring to _exit() or exit(). We made improvements in this area in the last draft (clarifying which part of the _Exit() page contained the "consequences of termination" referred to elsewhere). However, the _Exit() page itself still does not make clear that the list applies to all forms of termination.</p> <p>In POSIX.1-1990 the list of consequences was in the description of _exit() in section 3.2.2.2 and had a statement after it saying:</p> <p>"These consequences shall occur on process termination for any reason."</p> <p>This statement should be reinstated to make it clear that the list does not only apply to _Exit() and</p>	<p>Change last sentence of 3rd para DESC from:</p> <p>Finally, the calling process is terminated with the consequences described below.</p> <p>To:</p> <p>Finally, the calling process shall be terminated with the consequences described below.</p> <p>Change from</p> <p>These functions shall terminate the calling process with the following consequences:</p> <p>line 18589 [CX] Process termination caused by any reason shall have the following consequences.</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

US	Page: 565 Line: 19891 Section: alphasort		te	<p><code>_exit()</code>.</p> <p>The <code>strcoll()</code> function is allowed to fail, in which case the return value is indeterminate. But <code>qsort()</code> requires a total ordering from its comparison function, which would be violated by indeterminism. Therefore, <code>alphasort</code> needs to be specified how to behave in the face of file names that are invalid encodings in the current locale.</p> <p>As <code>alphasort</code> was not modified in draft 3, this aardvark should be considered in light of making the standard internally consistent.</p> <p>This aardvark tries to be as permissive as possible, introducing only the requirement that encoding errors are handled in a total ordering, without stating what order encoding errors are sorted into.</p> <p>Although this aardvark does not do so, it would also be possible to add a constraint on <code>qsort</code>, that <code>errno</code> is left unchanged unless the comparison function sets it; on <code>scandir</code>, that on success <code>errno</code> is left unchanged unless the comparison or selection function sets it; and on <code>alphasort</code>, that <code>errno</code> is left unchanged unless <code>strcoll</code> detects a failure. That way, an application could detect an encoding error that was handled by <code>alphasort</code> by setting <code>errno</code> to 0, then checking if <code>errno</code> is still 0 after a successful <code>scandir</code>.</p>	<p>Change from</p> <p>The <code>alphasort()</code> function can be used as the comparison function for the <code>scandir()</code> function to sort the directory entries into alphabetical order. Sorting happens as if by calling the <code>strcoll()</code> function on the <code>d_name</code> element of the <code>dirent</code> structures passed as the two parameters. Its parameters are the two directory entries, <code>d1</code> and <code>d2</code>, to compare.</p> <p>to:</p> <p>The <code>alphasort()</code> function can be used as the comparison function for the <code>scandir()</code> function to sort the directory entries, <code>d1</code> and <code>d2</code>, into alphabetical order. Sorting happens as if by calling the <code>strcoll()</code> function on the <code>d_name</code> element of the <code>dirent</code> structures passed as the two parameters. If the <code>strcoll()</code> function fails the return value of <code>alphasort()</code> is unspecified.</p> <p>The <code>alphasort()</code> function shall not change the setting of <code>errno</code> if successful. Since no return value is reserved to indicate an error, an application wishing to check for error situations should set <code>errno</code> to 0, then call <code>alphasort()</code>, then check <code>errno</code>.</p> <p>Change para ~ 19898 from</p> <p>The <code>scandir()</code> function shall scan the directory <code>dir</code>, calling the function referenced by <code>sel</code> on each directory entry. Entries for which the function referenced by <code>sel</code> returns</p>	
----	---	--	----	---	--	--

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>non-zero shall be stored in strings allocated as if by a call to malloc(), and sorted using qsort() with the comparison function compar, and collected in array namelist which shall be allocated as if by a call to malloc(). If sel is a null pointer, all entries shall be selected.</p> <p>to</p> <p>The scandir() function shall scan the directory dir, calling the function referenced by sel on each directory entry. Entries for which the function referenced by sel returns non-zero shall be stored in strings allocated as if by a call to malloc(), and sorted as if by a call to qsort() with the comparison function compar, except that compar need not provide total ordering. The strings are collected in array namelist which shall be allocated as if by a call to malloc(). If sel is a null pointer, all entries shall be selected. If the comparison function /compar/ does not provide total ordering the order in which the directory entries are stored is unspecified.</p> <p>Add to APPLICATION USAGE</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>If <code>dir</code> contains filenames that contain characters outside the domain of the collating sequence of the current locale, <code>alphasort()</code> need not provide a total ordering.</p>
US	<p>Page: 566 Line: 19949 Section: <code>scandir</code></p>		te	<p>The standard already has a nice application warning that the use of abrupt exits inside a callback of <code>ftw</code> can result in a memory leak. <code>scandir</code> is subject to the same problem, but bears no such warning (likewise for <code>nftw</code>). Since <code>scandir</code> was not edited for draft 3, this change should be considered in light of making new material added by API Set 1 consistent with the rest of the specification.</p>	<p>At line 19949, replace the Application Usage of "None." with:</p> <p>The <code>scandir()</code> function may allocate dynamic storage during its operation. If <code>scandir()</code> is forcibly terminated, such as by <code>longjmp()</code> or <code>siglongjmp()</code> being executed by the function pointed to by <code>sel</code> or <code>compar</code>, or by an interrupt routine, <code>scandir()</code> does not have a chance to free that storage, so it remains permanently allocated. A safe way to handle interrupts is to store the fact that an interrupt has occurred, then wait until <code>scandir()</code> returns to act on the interrupt.</p> <p><code>nftw</code> p1336</p> <p>At line 44264, replace the Application Usage of "None." with:</p> <p>The <code>nftw()</code> function may allocate dynamic storage during its operation. If <code>nftw()</code> is forcibly terminated, such as by <code>longjmp()</code> or <code>siglongjmp()</code> being executed by the function pointed to by <code>fn</code> or an interrupt routine, <code>nftw()</code> does not have a chance to free that storage, so it remains permanently allocated. A safe way to handle interrupts is to store the fact that an interrupt has occurred, and arrange to have the function pointed to by <code>fn</code> return a non-zero value at its next invocation.</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 593 Line: 20737 Section: bind		ed	The new EALREADY and EINPROGRESS errors have been added under "may fail". They were supposed to be "shall fail" (as in the description).	Move the EALREADY and EINPROGRESS errors from the "may fail" group (line 20734) to the "shall fail" group (line 20705).	
US	Page: 593 Line: 20749 Section: bind		ge	<p>This is an example program for this interface.</p> <p>As discussed with Andrew off list, the example programs I'm submitting come from already published versions in the Linux man pages, and each was written solely by me. I assign a non-exclusive copyright to The Open Group.</p> <p>This is an example program, intended to clarify the use of this interface for the reader of the standard. Page and line numbers refer to draft 3 of the revision.</p>	<p>substitute line 20749 ("None") with the following text:</p> <p>The following code segment shows how to create a socket and bind it to a name in the AF_UNIX domain.</p> <pre>#define MY_SOCKET_PATH "/somepath" int sfd; struct sockaddr_un my_addr; sfd = socket(AF_UNIX, SOCK_STREAM, 0); if (sfd == -1) /* Handle error */; memset(&my_addr, '\0', sizeof(struct sockaddr_un)); /* Clear structure */ my_addr.sun_family = AF_UNIX; strncpy(my_addr.sun_path,</pre>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<pre> MY_SOCKET_PATH, sizeof(my_addr.sun_path) - 1); if (bind(sfd, (struct sockaddr *) &my_addr, sizeof(struct sockaddr_un)) == -1) /* Handle error */; </pre>
US	Page: 596 Line: 20823 Section: bsearch		te	The bsearch() example has been changed to use the scanf() memory allocation modifier for %s, but it uses the old proposed 'M' instead of the 'm' that was actually added for scanf().	Change "%Ms" to "%ms".
US	Page: 697 Line: 23915 Section: dirfd		te	<p>The call sequences</p> <pre> fd = open(...) d = fdopendir(fd); </pre> <p>and</p> <pre> d = opendir(...) fd = dirfd(d); </pre> <p>effectively set up the same situation, where fd is a file descriptor associated with an open directory stream. However, the descriptions of dirfd() and fdopendir() are inconsistent as regards what can (or can't) be done with this file descriptor.</p> <p>dirfd() says:</p> <p>"The behavior of future calls to readdir() and readdir_r() is undefined if the application attempts to alter the file position indicator using the returned file</p>	<p>Change</p> <p>"The behavior of future calls to readdir() and readdir_r() is undefined if the application attempts to alter the file position indicator using the returned file descriptor. The behavior of future calls to closedir(), readdir(), and readdir_r() is undefined if the application attempts to close the file descriptor."</p> <p>to</p> <p>"If any attempt is made to close the file descriptor, or to modify the state of the associated description, other than by means of closedir(), readdir(), readdir_r(), or rewinddir(), the behavior is undefined."</p> <p>On page 795 line 27015-27016 section fdopendir, insert a comma before "other than by", and change "implementation-defined" to "undefined".</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>descriptor. The behavior of future calls to <code>closedir()</code>, <code>readdir()</code>, and <code>readdir_r()</code> is undefined if the application attempts to close the file descriptor."</p> <p><code>fdopendir()</code> says:</p> <p>"if any attempt is made to close the file descriptor, or to modify the state of the associated description other than by means of <code>closedir()</code>, <code>readdir()</code>, <code>readdir_r()</code>, or <code>rewinddir()</code>, the behavior is implementation-defined."</p> <p>The statement on the <code>fdopendir()</code> page seems the more correct to me. It allows for the possibility that changes to the state of the open file description other than just to its file position indicator could affect the behaviour of the directory stream.</p> <p>I also don't see why the <code>dirfd()</code> statement needs to say the behaviour is undefined if the fd is closed. If directory streams are implemented using a file descriptor then <code>readdir()</code> and <code>closedir()</code> should return an EBADF error. If directory streams are not implemented using a file descriptor then the fd is independently open for the same directory, and</p>		

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

			<p>closing it should not affect readdir() (but closedir() could presumably either succeed or give an EBADF error).</p> <p>Therefore I suggest adopting the fdopendir() statement on both pages, but with two minor changes:</p> <ol style="list-style-type: none"> 1. It seems an unnecessary burden on implementors to make them document all the ways in which a state change affects the behaviour of directory streams, so I think it should say "unspecified" rather than "implementation-defined". 2. An extra comma is needed, since calling closedir() is allowed, but closedir() counts as an "attempt to close the file descriptor". 	
US	<p>Page: 796 Line: 27067 Section: fdopendir</p>		<p>ge</p> <p>The second fdopendir() example has a couple of minor problems.</p> <ol style="list-style-type: none"> 1. The introductory comment doesn't quite match the code. 2. It uses %d to print an off_t value. 	<p>Change</p> <p>"The following program fragment searches through a given directory looking for files larger than 1 MB."</p> <p>to</p> <p>"The following program searches through a given directory looking for files whose name does not begin with a dot and whose size is larger than 1 MiB."</p> <p>Insert before line 27073:</p> <pre>#include <stdint.h></pre> <p>On line 27097 change:</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<pre>printf("%s: %dk\n", dp->d_name, statbuf.st_size / 1024); to: printf("%s: %jdK\n", dp->d_name, (intmax_t)(statbuf.st_size / 1024));</pre>	
US	Page: 797 Line: 27082 Section: fdopendir		te	Line 27013 of fdopendir forbids certain use of a file descriptor after handing it to fdopendir. The example starting at line 27069 currently violates this by using dfd in openat and close after the fact.	remove line 27102 close(dfd); Add a comment to 27101 closedir(d); // note this implicitly closes dfd	
US	Page: 826 Line: 27888 Section: fgetwc		te	<p>The fgetwc() page says:</p> <p>"The last data access timestamp shall be marked for update by the first successful execution of fgetc(), fgets(), fgetwc(), fgetws(), fread(), fscanf(), getc(), getchar(), gets(), or scanf() using stream that returns data not supplied by a prior call to ungetc() or ungetwc()."</p> <p>This list of functions contains byte input functions as well as wide-character input functions. However,</p>	<p>Change</p> <p>"fgetc(), fgets(), fgetwc(), fgetws(), fread(), fscanf(), getc(), getchar(), gets(), or scanf()"</p> <p>to</p> <p>"fgetwc(), fgetws(), fwscanf(), getwc(), getwchar(), vwscanf(), vwscanf(), or wscanf()"</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

				<p>section 2.5.2 states "Byte input/output functions cannot be applied to a wide-oriented stream, and wide-character input/output functions cannot be applied to a byte-oriented stream." The byte input functions should be removed from the list. Likewise ungetc() cannot be used on a wide-oriented stream and should not be mentioned.</p> <p>The descriptions of the other wide-character input functions have the same problem.</p> <p>The fread() page has the opposite problem: it is a byte input function, but the function list includes some wide-character input functions and it mentions ungetwc().</p> <p>The affected paragraphs all have change bars, but the change was in text that comes before the function lists, so just in case this does not meet the narrowing-down rules I should point out that the implication that byte and wide-character input functions can be mixed is in conflict with the C Standard and the change should therefore be in scope anyway.</p>	<p>On line 27889 delete "ungetc() or".</p> <p>Repeat the same two changes for:</p> <p>page 828 line 27965 section fgetws page 953 line 32527 section fwscanf</p> <p>(There may be slight variation in the current function lists, but they are not relevant: in each case the listed functions should be replaced with the same one specified above for the fgetwc() page.)</p> <p>On page 885 line 30035 section fread, delete "fgetwc(), fgetws(),"</p> <p>On page 885 line 30036 section fread, delete "or ungetwc()"</p>	
US	Page: 838 Line: 28309 Section: fmemopen		ed	<p>In two places the current text talks about "advancing the current buffer position behind the current buffer size" (lines 28309 and 28317).</p> <p>I do not understand this terminology, and suspect that this is a simple typo where "behind" should be "beyond".</p>	<p>Replace the word "behind" on lines 28309 and 28317 with the word "beyond".</p>	
US	Page: 839 Line: 28338		ed	<p>The example for fmemopen uses strlen without a prior declaration.</p>	<p>At line 28338, add a line:</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
	Section: fmemopen			As this section was not edited for draft 3, this change should be considered under the rule of making the draft internally consistent (code examples should use valid code).	#include <string.h>	
US	Page: 869 Line: 29497 Section: fprintf		te	The fprintf page is missing a statement about timestamp update by dprintf().	Change "The last data modification and last file status change timestamps of the file shall be marked for update between the call to a successful execution of fprintf() or printf() and the next successful completion of a call to fflush() or fclose() on the same stream or a call to exit() or abort()." to "The last data modification and last file status change timestamps of the file shall be marked for update: a. between the call to a successful execution of fprintf() or printf() and the next successful completion of a call to fflush() or fclose() on the same stream or a call to exit()	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>or abort();</p> <p>b. upon successful completion of a call to dprintf()."</p>
US	<p>Page: 887 Line: 30090 Section: free</p>		te	<p>The list of functions that generate valid pointers for free() is incomplete, in light of the API set additions. Since free was not changed in draft 3, this change should be considered in light of consistency with other additions.</p> <p>Furthermore, strdup() was moved from XSI to Base, but does not exist in the C standard, so now that it is no longer XSI shaded, it should be CX shaded.</p> <p>This aardvark does not address the asymmetry in the fact that there is now strndup but no wcsndup, but if that is addressed independently, free() would also need to list wcsndup.</p>	<p>On page 566 line 19949 (alphasort/scandir), change "None" to</p> <p>For functions that allocate memory as if by malloc(), the application should release such memory when it is no longer required by a call to free(). For scandir(), this is /namelist/ (including all of the individual strings in /namelist/).</p> <p>Page 566 line 19955, remove "malloc()"</p> <p>Page 887 line 30089-30091, change:</p> <p>Otherwise, if the argument does not match a pointer earlier returned by the calloc(), malloc(), <ADV>posix_memalign()</ADV>, realloc(), or strdup() function, or if the space has been deallocated by a call to free() or realloc(), the behavior is undefined.</p> <p>to</p> <p>Otherwise, if the argument does not match a pointer earlier returned by a function in POSIX.1-200x that allocates memory as if by malloc(), or if the space has been deallocated</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>by a call to free() or realloc(), the behavior is undefined.</p> <p>Page 887 line 30107 (not sure if we did this already)</p> <p>Change entire line to: calloc(), malloc(), <ADV>posix_memalign</ADV>, realloc()</p> <p>Page 905, line 30764 (fscanf): add new paragraph</p> <p>For functions that allocate memory as if by malloc(), the application should release such memory when it is no longer required by a call to free(). For fscanf(), this is memory allocated via use of the 'm' assignment-allocation character.</p> <p>Page 934 line 31728 (ftw) Remove malloc() from the list.</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

Page 954 line 32560 (fwscanf): add new paragraph

For functions that allocate memory as if by malloc(), the application should release such memory when it is no longer required by a call

to free(). For fwscanf(), this is memory allocated via use of the 'm' assignment-allocation character.

Page 964 line 32833, delete the sentence "Also, malloc() and free() are used nowhere else in this volume of POSIX.1-200x."

Page 1950 line 62039 (strdup/strndup)

Replace "None" with:

For functions that allocate memory as if by malloc(), the application should release such memory when it is no longer required by a call

to free(). For strdup() and strndup(), this is the return value.

Page 1950 line 62047 remove malloc().

Page 2155 line 68032 (wcsdup)

Replace "None" with:

For functions that allocate memory as if by malloc(), the application should release such memory when it is no longer required by a call

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>to free(). For wcsdup() this is the return value.</p> <p>line 68020 (wcsdup) change from The wcsdup() function shall return a pointer to a new wide-character string, which is the duplicate of the wide-character string string.</p> <p>to: The wcsdup() function shall return a pointer to a new wide-character string, allocated as if by a call to malloc(), which is the duplicate of the wide-character string string.</p> <p>-- realpath p 1715 I 54996 change from If resolved_name is a null pointer, the generated pathname shall be stored as a null-terminated string in a buffer allocated by realpath().</p> <p>to</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>If resolved_name is a null pointer, the generated pathname shall be stored as a null-terminated string in a buffer allocated as if by a call to malloc().</p> <p>Add to APP USAGE For functions that allocate memory as if by malloc(), the application should release such memory when it is no longer required by a call to free(). For realpath() this is the return value.</p>
US	Page 963 line 32823 (getcwd)		ge	<p>Several examples have program fragments that malloc() but then end "..." I believe in general it is good advice to remind readers that the application has a responsibility to free such memory,</p>	<p>Page 963 line 32823 (getcwd)</p> <p>Add new lines after ellipsis: free (buf);</p> <p>Page 980 line 33361 (getgrgid)</p> <p>Add new line after ellipsis free (buffer);</p> <p>Page 983 line 33488 (getgrnam)</p> <p>Add new line after ellipsis free (buffer);</p> <p>Page 1022 line 34653 (getpwnam)</p> <p>Add new line after ellipsis free (buffer);</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>Page 1026 line 34781 (getpwuid)</p> <p>Add new line after ellipsis free (buffer);</p> <p>Page 1022 line 34653 (getpwnam)</p> <p>Add new line after ellipsis free (buffer);</p>	
US	Page: 891 Line: 30262 Section: getaddrinfo		ge	<p>This is an example program for this interface.</p> <p>As discussed with Andrew off list, the example programs I'm submitting come from already published versions in the Linux man pages, and each was written solely by me. I assign a non-exclusive copyright to The Open Group.</p> <p>This is an example program, intended to clarify the use of this interface for the reader of the standard. Page and line numbers refer to draft 3 of the revision.</p>	<pre>#include <stdio.h> #include <stdlib.h> #include <unistd.h> #include <string.h> #include <sys/socket.h> #include <netdb.h> int main(int argc, char *argv[]) { struct addrinfo *result, *rp; int sfd, s;</pre>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

```
if (argc != 2) {
    fprintf(stderr, "Usage: %s port\n",
argv[0]);
    exit(EXIT_FAILURE);
}

struct addrinfo hints = {};
hints.ai_family = AF_UNSPEC;
hints.ai_socktype = SOCK_DGRAM;
hints.ai_flags = AI_PASSIVE;
hints.ai_protocol = 0;

s = getaddrinfo(NULL, argv[1], &hints,
&result);
if (s != 0) {
    fprintf(stderr, "getaddrinfo: %s\n",
gai_strerror(s));
    exit(EXIT_FAILURE);
}

/* getaddrinfo() returns a list of address
structures.
Try each address until a successful
bind().
If socket(2) (or bind(2)) fails, close the
socket
and try the next address. */

for (rp = result; rp != NULL; rp = rp-
>ai_next) {
    sfd = socket(rp->ai_family, rp-
>ai_socktype,
rp->ai_protocol);
    if (sfd == -1)
```

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<pre> continue; if (bind(sfd, rp->ai_addr, rp->ai_addrlen) == 0) break; /* Success */ close(sfd); } if (rp == NULL) { /* No address succeeded */ fprintf(stderr, "Could not bind\n"); exit(EXIT_FAILURE); } freeaddrinfo(result); /* No longer needed */ /* ... use socket bound to sfd ... */ } </pre>	
US	Page: 901 Line: 30572 Section: fscanf		te	The statement "The %s, %S and %[conversion specifiers shall accept an optional assignment-allocation character	Change "%s, %S and %[" to "%c, %s and %[". On page 950 line 32375 section fwscanf,	

¹ **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

² **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

			<p>'m'</p> <p>omits the %c and %C conversion specifiers, which accept 'm'.</p> <p>Also, the %S (and %C if added) should really be XSI shaded, but that would be difficult since the whole para is CX. A solution would be just to list %c, %s and %[here, since %C and %S are stated to be equivalent to %lc and %ls.</p> <p>The fwscanf page has the same problem, but also has a problem in the %c description. Part of it has not been updated in line with the other 'm' character updates.</p>	<p>change "%s, %S and %[" to "%c, %s and %[".</p> <p>On page 953 line 32483 section fwscanf, change</p> <p>"The corresponding argument shall be a pointer to the initial element of a character array large enough to accept the sequence. No null character is added."</p> <p>to</p> <p>"No null character is added. If the 'm' assignment-allocation character is not specified, the application shall ensure that the corresponding argument is a pointer to the initial element of a character array large enough to accept the sequence.</p> <p>[CX]Otherwise, the application shall ensure that the corresponding argument is a pointer to a pointer to a char.[/CX]"</p>	
US	Page: 901 Line: 30578 Section: fscanf()		<p>te</p> <p>The specification currently demands that all memory allocated in scanf due to 'm' modifiers is freed in case the call failed due to an internal ENOMEM error.</p> <p>This is not sufficient. If scanf returns EOF for any other reason the caller has no idea which if the pointers passed in for 'm' parameters points to valid memory and which not. One possibility would be to require that all pointers are set by the implementation to NULL before returning but this seems to be less optimal than the alternative: require all memory to be freed before returning</p>	<p>Change in lines 30578ff:</p> <p>In that case, any memory successfully allocated for other parameters using assignment-allocation character 'm' by this call shall be freed.</p> <p>to</p> <p>If the function returns EOF, any memory successfully allocated for parameters using assignment-allocation character 'm' by this call shall be freed before the function returns.</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				EOF.	<p>In RETURN VALUE section change from If a read CX error occurs, the error indicator for the stream is set, EOF shall bereturned, and errno shall be set to indicate the error.</p> <p>To</p> <p>If any error occurs, EOF shall be returned, [CX]and errno shall be set to indicate the error. [/CX]If a read error occurs the error indicator for the stream shall be set.</p> <p>Change in lines 32382 fwscanf p 950:</p> <p style="padding-left: 40px;">In that case, any memory successfully allocated for other parameters using assignment-allocation character 'm' by this call shall be freed.</p> <p>to</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>If the function returns EOF, any memory successfully allocated for parameters using assignment-allocation character 'm' by this call shall be freed before the function returns.</p> <p>p954 l32533 fwscanf In RETURN VALUE section change from If a read [CX] error occurs, the error indicator for the stream is set, EOF shall be returned, and errno shall be set to indicate the error.</p> <p>To</p> <p>If any error occurs, EOF shall be returned, [CX]and errno shall be set to indicate the error. [/CX]If a read error occurs the error indicator for the stream shall be set.</p>	
US	Page: 915 Line: 31096 Section: fstatat		te	<p>The non-symlink-related changes from XSH ERN 155 are included here, so that this aardvark can supersede XSH ERN 155.</p> <p>This aardvark also supersedes XCU ERN 147 for ls -t.</p> <p>Mandating inodes for symlinks resolves many issues with the current standard, but one issue remains regarding symlink permissions.</p> <p>Currently readlink() is allowed to fail with EACCES if the symlink has no read permission, but there is no allowance for possible failure to read the</p>	<p>On Page: 915 Line: 31096 Section: fstatat Change</p> <p>"For symbolic links, the st_mode member shall contain meaningful information when used with the file type macros, and the st_size member shall contain the length of the pathname contained in the symbolic link. File mode bits and the contents of the remaining members of the stat structure are unspecified. The value returned in the st_size member is the length of the contents of the symbolic link, and does not count any trailing null."</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>contents of a symbolic link during pathname resolution, and there is no requirement on how symlink() sets the permissions on implementations where readlink() honours them.</p> <p>A solution to this is included below: remove the optional readlink() EACCES error and mandate that all interfaces ignore the permissions. The alternative would be to add optional EACCES errors to all functions that take a pathname, along the lines of "Read permission was denied for a symbolic link which needed to be followed to resolve the path argument"; add requirements for symlink() on how it sets the permissions on implementations where readlink() honours them; require that st_mode contains the actual permissions on implementations where readlink() honours them; add a way for applications to query whether readlink() honours symlink permissions; and (probably) add lchmod() and chmod -h.</p>	<p>to</p> <p>"For symbolic links, the st_mode member shall contain meaningful information when used with the file type macros. The file mode bits in st_mode are unspecified. The structure members st_ino, st_dev, st_uid, st_gid, st_atim, st_ctim, and st_mtim shall have meaningful values and the value of the st_nlink member shall be set to the number of (hard) links to the symbolic link. The value of the st_size member shall be set to the length of the pathname contained in the symbolic link not including any terminating null byte."</p> <p>On page 2460 line 79290 section chown, replace the -h description with</p> <p>"For each file operand that names a file of type symbolic link, chown shall attempt to set the user ID of the symbolic link. If a group ID was specified, for each file operand that</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

names a file of type symbolic link, chown shall attempt to set the group ID of the symbolic link."

On page 2460 line 79308 section chown, delete

"if the system supports this operation"

On page 1170 line 39190 section lchown, delete

"[EOPNOTSUPP] The path argument names a symbolic link and the implementation does not support setting the owner or group of a symbolic link."

On page 1180 line 39471 section link, delete

"unless the implementation does not support making hard links to symbolic links in which case the linkat() call shall fail"

On page 1181 line 39505 section link, delete

"[EOPNOTSUPP] The AT_SYMLINK_FOLLOW flag is clear in the flag argument, the path1 argument names a symbolic link, and the implementation does not support making hard links to symbolic links."

On page 1704 line 54608 section readdir,

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>append</p> <p>"The value of the structure's d_ino member shall be set to the file serial number of the file named by the d_name member. If the d_name member names a symbolic link, the value of the d_ino member shall be set to the file serial number of the symbolic link itself."</p> <p>On page 1704 line 54623 section readdir, delete</p> <p>"If the entry names a symbolic link, the value of the d_ino member is unspecified."</p> <p>On page 1706 line 54708 section readdir, insert a new paragraph:</p> <p>"When returning a directory entry for the root of a mounted file system, some historical implementations of readdir() returned the file serial number of the underlying mount point, rather than of the root of the mounted file system. This behavior is considered to be</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

a bug, since the underlying file serial number has no significance to applications."

On page 1709 line 54808 section readlink, delete

"[EACCES] Read permission is denied for the symbolic link referred to by path."

On page 2005 line 63682 section symlink, insert three new paragraphs:

"The symbolic link's user ID shall be set to the process' effective user ID. The symbolic link's group ID shall be set to the group ID of the parent directory or to the effective group ID of the process. Implementations shall provide a way to initialize the symbolic link's group ID to the group ID of the parent directory. Implementations may, but need not, provide an implementation-defined way to initialize the symbolic link's group ID to the effective group ID of the calling process.

The values of the file mode bits for the created symbolic link are unspecified. All interfaces specified by POSIX.1-200x shall behave as if the contents of symbolic links can always be read, except that the value of the file mode bits returned in the `st_mode` field of the `stat` structure is unspecified.

Upon successful completion, `symlink()` shall mark for update the last data access,

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>last data modification, and last file status change timestamps of the symbolic link. Also, the last data modification and last file status change timestamps of the directory that contains the new entry shall be marked for update."</p> <p>Cross-volume changes to XCU:</p> <p>On page 2789 line 92056 section ls, append</p> <p>For a symbolic link, the time used as the sort key is that of the symbolic link itself, unless 'ls' is evaluating its file information to be that of the file referenced by the link (see the -H and -L options).</p> <p>On page 2926 line 97400 section pax, delete</p> <p>"except that if the files to be linked are symbolic links and the system is not capable of making hard links to symbolic links, then separate copies of the symbolic link shall be created instead".</p>	
US	Page: 937		ed	Two finegrain aardvarks (XSHfg ERN 61 and 69)	Change	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

	Line: 31807 Section: futimens			both modified the same text, and part of one (61) got lost in the process.	"The times argument is a null pointer and" to "The times argument is a null pointer, or both tv_nsec values are UTIME_NOW, and"	
US	Page: 938 Line: 31880 Section: futimens		ed	The rename of futimesat still has some editorial fixups needed.	In XSH, at line 31880, change " <i>futimesat</i> ()" to " <i>futimesat</i> ()". In XRAT, at line 124279, under POSIX_FILE_SYSTEM_FD, replace futimesat with utimensat and re-sort the list.	
US	Page: 970 Line: 33066 Section: getdelim		te	The getdelim() page lists the following error related to the stream argument: "[EINVAL] stream is not a valid file descriptor" Apart from the fact that the stream argument is not a "file descriptor", specifying an EINVAL error for a (FILE *) stream argument is inconsistent with the errors listed for other stdio input functions related to the stream argument. The best way to fix the problem would be to use the same approach as the fscanf() page, by adding a reference to the fgetc() page to cover the errors related to the stream, and just listing the additional errors specific to getdelim() and getline() on the getdelim() page.	At line 33062 change: "These functions shall fail if:" to: "For the conditions under which the getdelim() and getline() functions shall fail and may fail, refer to [xref to fgetc()]." In addition, these functions shall fail if:" At line 33063 change: "When lineptr or n are a null pointer." to: "lineptr or n is a null pointer."	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				There is also an editorial correction needed to the first EINVAL error.	Delete line 33066 (the second EINVAL error).	
US	Page: 1178 Line: 39378 Section: lgamma		ge	The description of lgamma() says: "The sign of Gamma(x) is returned in the external integer signgam." But the sign of Gamma(x) is not defined when x is NaN, -Inf or a negative integer. And POSIX doesn't say whether the value is unmodified, it is set to some unspecified sign (-1, 0 or 1) or it is undefined.	After "The sign of Gamma(x) is returned in the external integer signgam." add "If x is NaN, -Inf or a negative integer, the value of signgam is unspecified."	
US	Page: 1184 Line: 39606 Section: linkat		ed	The cross reference from linkat to link takes you to XCU link, not to XSH link.	Change the xref to refer to the link() function on XSH page 1180	
US	Page: 1231 Line: 41075 Section: malloc		ed	The "See Also" section for malloc includes "alphasort()" (which doesn't use malloc, though scandir on the same page does). It also calls out fmemopen, but not open_memstream or open_wmemstream, nor scanf, all of which have an interaction with malloc.	Replace the See Also line with: calloc(), free(), posix_memalign(), realloc(), setrlimit() XBD <stdlib.h>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

				<p>Also, <code>setrlimit()</code> can be used to limit the amount of memory available for allocation, and <code>mmap()</code> can also reduce the amount of memory available; both of these deserve mention in the see also list (both have details about this interaction on the relevant pages)</p>	
US	<p>Page: 1290 Line: 42890 Section: mq_notify</p>		ge	<p>This is an example program for this interface. As discussed with Andrew off list, the example programs I'm submitting come from already published versions in the Linux man pages, and each was written solely by me. I assign a non-exclusive copyright to The Open Group.</p> <p>This is an example program, intended to clarify the use of this interface for the reader of the standard. Page and line numbers refer to draft 3 of the revision.</p>	<p>(Suggest also to replace line 42838 (XSH <code>mq_getattr EXAMPLES</code>) with: "See <code>mq_notify()</code>.)</p> <p>substitute line 42890 ("None") with the following text:</p> <p>The following program registers a notification request for the message queue named in its command-line argument. Notification is performed by creating a thread. The thread executes a function which reads one message from the queue and then terminates the process.</p> <pre>#include <pthread.h> #include <mqueue.h> #include <assert.h> #include <stdio.h> #include <stdlib.h> #include <unistd.h> static void /* Thread start function */ tfunc(union signal sv) {</pre>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<pre> struct mq_attr attr; ssize_t nr; void *buf; mqd_t mqdes = *((mqd_t *) sv.sival_ptr); /* Determine maximum msg size; allocate buffer to receive msg */ if (mq_getattr(mqdes, &attr) == -1) { perror("mq_getattr"); exit(EXIT_FAILURE); } buf = malloc(attr.mq_msgsize); if (buf == NULL) { perror("malloc"); exit(EXIT_FAILURE); } nr = mq_receive(mqdes, buf, attr.mq_msgsize, NULL); if (nr == -1) { perror("mq_receive"); exit(EXIT_FAILURE); } </pre>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

```

    }

    printf("Read %ld bytes from message
queue\n", (long) nr);
    free(buf);
    exit(EXIT_SUCCESS);    /* Terminate
the process */
}

int
main(int argc, char *argv[])
{
    mqd_t mqdes;
    struct sigevent not;

    assert(argc == 2);

    mqdes = mq_open(argv[1], O_RDONLY);
    if (mqdes == (mqd_t) -1) {
        perror("mq_open");
        exit(EXIT_FAILURE);
    }

    not.sigev_notify = SIGEV_THREAD;
    not.sigev_notify_function = tfunc;
    not.sigev_notify_attributes = NULL;
    not.sigev_value.sival_ptr = &mqdes;    /*
Arg. to thread func. */
    if (mq_notify(mqdes, &not) == -1) {
        perror("mq_notify");
        exit(EXIT_FAILURE);
    }

    pause();    /* Process will be terminated by

```

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					thread function */ }	
Us	Page: 1336 Line: 44247-44262 Section: nftw		ge	<p>This is an example program for this interface. As discussed with Andrew off list, the example programs</p> <p>I'm submitting come from already published versions in the Linux man pages, and each was written solely by me. I assign a non-exclusive copyright to The Open Group.</p> <p>This is an example program, intended to clarify the use of this interface for the reader of the standard.</p> <p>Page and line numbers refer to draft 3 of the revision. This example is intended as an improvement of the existing example for this interface.</p>	<pre>#include <ftw.h> #include <stdio.h> #include <stdlib.h> #include <string.h> #include <stdint.h> static int display_info(const char *fpath, const struct stat *sb, int tflag, struct FTW *ftwbuf) { printf("%-3s %2d %7jd %-40s %d %s\n", (tflag == FTW_D) ? "d" : (tflag == FTW_DNR) ? "dnr" : (tflag == FTW_DP) ? "dp" : (tflag == FTW_F) ? "f" : (tflag == FTW_NS) ? "ns" : (tflag == FTW_SL) ? "sl" : (tflag == FTW_SLN) ? "sln" : "???", ftwbuf->level, (intmax_t) sb->st_size, fpath, ftwbuf->base, fpath + ftwbuf- >base);</pre>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<pre> return 0; /* To tell nftw() to continue */ } int main(int argc, char *argv[]) { int flags = 0; if (argc > 2 && strchr(argv[2], 'd') != NULL) flags = FTW_DEPTH; if (argc > 2 && strchr(argv[2], 'p') != NULL) flags = FTW_PHYS; if (nftw((argc < 2) ? "." : argv[1], display_info, 20, flags) == -1) { perror("nftw"); exit(EXIT_FAILURE); } exit(EXIT_SUCCESS); } </pre>	
US	Page: 1345 Line: 44479 Section: open		te	<p>While this is primarily an editorial change, and out of scope for this ballot because the line has not changed since D2R, I believe this problem was missed at the last ballot and is of sufficient magnitude to require immediate remediation rather than waiting for a TC after publication. I am therefore filing this as an objection.</p> <p>The text for O_NONBLOCK has become elided with the text for O_NOFOLLOW. These are two distinct and separate cases. In TC2, the text said:</p>	<p>Break the O_NOFOLLOW line after the first sentence. Add a new list header "O_NONBLOCK".</p> <p>Resulting text should read:</p> <p>O_NOCTTY</p> <p>If set and path identifies a terminal device, open() shall not cause the terminal device to become the controlling terminal for the process.</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>O_NOCTTY If set and path identifies a terminal device, open() shall not cause the terminal device to become the controlling terminal for the process.</p> <p>O_NONBLOCK When opening a FIFO with O_RDONLY or O_WRONLY set: ...</p> <p>The new text says:</p> <p>O_NOCTTY If set and path identifies a terminal device, open() shall not cause the terminal device to become the controlling terminal for the process.</p> <p>O_NOFOLLOW If path names a symbolic link, fail and set errno to [ELOOP]. When opening a FIFO with O_RDONLY or O_WRONLY set: ...</p>	<p>O_NOFOLLOW If path names a symbolic link, fail and set errno to [ELOOP].</p> <p>O_NONBLOCK When opening a FIFO with O_RDONLY or O_WRONLY set: ...</p>	
US	Page: 1350 Line: 44698-44703 Section: open		ge	The paragraph beginning at line 44698 ("For example ...") is misplaced because it is an example of the issue discussed 3 paragraphs back. To avoid confusion, this example should immediately follow the paragraph describing the corresponding issue, as it did in the 2004 edition.	Move lines 44698-44703 immediately after line 44689.	
US	Page: 1352		te	In spite of the changes made to open_memstream	change from	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

Line: 44785
Section:
open_memstream

as part of draft 3, the spec is internally inconsistent. open_memstream requires that the opened stream be seekable (line 44773), fseeko requires that a seekable stream allow the current position to be set beyond the end of a file but that the intermediate data is unspecified unless a write takes place (line 30825), fflush only changes the current position if there is buffered unwritten data, and open_memstream requires that on fflush the variable pointed to by sizep is set to the bytes between the beginning of the buffer and the current position (line 44785).

According to the above rules,

```
char *buf;
```

```
size_t size;
```

```
FILE *f = open_memstream (&buf, &size);
```

```
int err = fseeko (f, 2, SEEK_SET);
```

should leave the stream at position 2, even though the end of the file is at position 0. Following this with

```
fflush (f);
```

would imply that buf now points to at least three bytes, size is 2, and buf[2] is 0, but does not state what buf[0] and buf[1] are set to. If it is truly desirable to always set size according to the current file position, then we should specify what the data

After a successful fflush() or fclose(), the pointer referenced by bufp shall contain the address of the buffer, and the variable pointed to by sizep shall contain the number of bytes for open_memstream() or the number of wide characters for open_wmemstream(), between the beginning of the buffer and the current file position indicator.

to

After a successful fflush() or fclose(), the pointer referenced by bufp shall contain the address of the buffer, and the variable pointed to by sizep shall contain the smaller of the current buffer length and the number of bytes for open_memstream(), or the number of wide characters for open_wmemstream(), between the beginning of the buffer and the current file position indicator.

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>between EOF and the size is set to. But more likely, this should behave like regular files, where seeking beyond the end of a file then flushing does not change the current position of the stream, but also does not change the size of the file visible to stat - it takes a further write at the new position to make the visible size change.</p> <p>For comparison, the existing glibc implementation sets err to -1, but fails to set errno; a followup ftello shows that glibc left the current position at 0, basically making it impossible to seek beyond the end of written data. This behavior cannot be standardized (if a function fails, it must set errno), but the question remains as to whether the glibc bug is in failing to set errno, or in failing to seek.</p> <p>The proposed wording assumes that it is desirable to allow seeks beyond the end of a memstream, at which point subsequent writes fill the gap with NUL bytes, by changing the fflush and fclose rules to state that psize is set to the current end of file (ie. buffer length) if the current position exceeds the current end of file. An alternative fix might be rewording fseek/fseeko to forbid seeking beyond</p>		

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

				the end of a memstream, but that would make memstreams needlessly different from regular file streams.	
US	Page: 1491 Line: 48541 Section: psiginfo		te	<p>The perror function makes it clear that the orientation of stderr is unchanged. A similar specification is lacking for the new psiginfo/psignal interfaces [see the defect raised on XBDd3 3.84]. Further, psiginfo is missing information on how timestamps are affected. Since psiginfo was not edited for draft 3, this change should be considered in light of consistency with other changes, and so that psiginfo does not conflict with the C standard.</p> <p>However, the wording in perror is a bit confusing - if no prior output has been made to the standard error stream, then does perror result in byte or wide character output? And how does it restore stderr back to unspecified orientation (fwrite cannot remove orientation, and freopen loses the position)? The proposed wording takes the approach that if stderr is not yet oriented, that the functions in question result in stderr having byte orientation.</p>	<p>At line 48541, insert into the psiginfo DESCRIPTION:</p> <p>The psiginfo() and psignal() functions shall mark for update the last data modification and last file status change timestamps of the file associated with the standard error stream at some time between their successful completion and exit(), abort(), or the completion of fflush() or fclose() on stderr.</p> <p>Fix similar text in perror p1363 45023</p> <p>The perror() function shall mark for update the last data modification and last file status change timestamps of the file associated with the standard error stream at some time between its successful completion and exit(), abort(), or the completion of fflush() or fclose() on stderr.</p>
US	Page: 1495 Line: 48676 Section: pthread_attr _destroy		te	<p>XSH ERN 215 identified a problem with pthread ESRCH errors that also affects several other pthread "may fail" errors. For consistency the same solution should be applied in these other cases.</p> <p>To summarise the problem: the intention of the POSIX.1c developers was that when a "may fail" condition for a pthreads function is caused by "application error", if the implementation chooses not to detect the optional error then the behaviour would be undefined, but in many cases they neglected to state that the behaviour is undefined in</p>	<p>Delete the following "may fail" EINVAL errors:</p> <p>P1495 L48676 pthread_attr_destroy P1498 L48800 pthread_attr_getdetachstate or pthread_attr_setdetachstate P1500 L48883 pthread_attr_getguardsize or pthread_attr_setguardsize P1503 L48975 pthread_attr_getinheritsched P1503 L48979 (2nd EINVAL for) pthread_attr_setinheritsched P1505 L49043 pthread_attr_getschedparam</p>

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>normative text. (An explicit statement to that effect was needed because POSIX.1-1990 - the base standard which POSIX.1c amended - stated in section 2.4 "Each implementation shall document, in the conformance document, situations in which each of the optional conditions are detected. If no error condition is detected, the action requested shall be successful." This statement also appeared in POSIX.1-1996, the merged standard which incorporated POSIX.1c.)</p> <p>Note that the changes below do not include changes for the pthread_mutex_getprioceiling, pthread_mutex_lock and pthread_mutex_timedlock pages, which are the subject of a separate aardvark.</p> <p>Also, I have not proposed changes related to the pthread_rwlock_*() EDEADLK errors as the standard already has a clear requirement for two optional behaviours: either deadlock or return an EDEADLK error.</p> <p>The changes are in groups of related errors. E.g. the first group is for thread attribute object EINVAL</p>	<p>P1505 L49046 pthread_attr_setschedparam (just "or the value ... object" part) P1507 L49098 pthread_attr_getschedpolicy P1507 L49101 pthread_attr_setschedpolicy (just "or the value ... object" part) P1509 L49159 pthread_attr_getscope P1509 L49162 pthread_attr_setscope (just "or the value ... object" part) P1511 L49228 pthread_attr_getstack P1511 L49232 pthread_attr_setstack (just "or the value ... object" part) P1513 L49301 pthread_attr_getstacksize or pthread_attr_setstacksize P1563 L50573 pthread_create</p> <p>and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name(s) shown above for pthread_attr_destroy):</p> <p>"The behavior is undefined if the value specified by the attr argument to pthread_attr_destroy() does not refer to an initialized thread attributes object."</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

errors. Where an addition to the end of the DESCRIPTION or RATIONALE sections is specified it may be preferable, for some functions within a group, to add the new text at an earlier location. The editors should feel free to use their discretion here.

and add the following statement to the end of the subsequent RATIONALE (again substituting the function names):

"If an implementation detects that the value specified by the attr argument to pthread_attr_destroy() does not refer to an initialized thread attributes object, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EBUSY error:

P1495 L48679 pthread_attr_init

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the attr argument to pthread_attr_init() refers to an already initialized thread attributes object, it is recommended that the function should fail and report an EBUSY error."

Delete the following "may fail" EBUSY errors:

P1524 L49433 pthread_barrier_destroy

P1525 L49442 pthread_barrier_init

and add the following statement to the end of the subsequent RATIONALE:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>"If an implementation detects that the value specified by the barrier argument to pthread_barrier_destroy() or pthread_barrier_init() refers to a barrier that is in use (for example, in a pthread_barrier_wait() call) by another thread, or detects that the value specified by the barrier argument to pthread_barrier_init() refers to an already initialized barrier object, it is recommended that the function should fail and report an EBUSY error."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1524 L49436 pthread_barrier_destroy P1526 L49496 pthread_barrier_wait</p> <p>and in each case add the following statements to the end of the subsequent RATIONALE (substituting the function name shown above for pthread_barrier_destroy):</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

"If an implementation detects that the value specified by the barrier argument to pthread_barrier_destroy() does not refer to an initialized barrier object, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EINVAL error:

P1525 L49445 pthread_barrier_init

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the attr argument to pthread_barrier_init() does not refer to an initialized barrier attributes object, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EINVAL errors:

P1528 L49539 pthread_barrierattr_destroy
P1530 L49591
pthread_barrierattr_getpshared or
pthread_barrierattr_setpshared

and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name(s) shown above for pthread_barrierattr_destroy):

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>"The behavior is undefined if the value specified by the attr argument to pthread_barrierattr_destroy() does not refer to an initialized barrier attributes object."</p> <p>and add the following statement to the end of the subsequent RATIONALE (again substituting the function names):</p> <p>"If an implementation detects that the value specified by the attr argument to pthread_barrierattr_destroy() does not refer to an initialized barrier attributes object, it is recommended that the function should fail and report an EINVAL error."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1541 L49919 pthread_cond_broadcast or pthread_cond_signal P1544 L50036 pthread_cond_destroy</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name(s) shown above for pthread_cond_destroy):

"The behavior is undefined if the value specified by the cond argument to pthread_cond_destroy() does not refer to an initialized condition variable."

and add the following statement to the end of the subsequent RATIONALE (again substituting the function names):

"If an implementation detects that the value specified by the cond argument to pthread_cond_destroy() does not refer to an initialized condition variable, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EBUSY errors:

P1544 L50033 pthread_cond_destroy

P1545 L50042 pthread_cond_init

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the cond argument to

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>pthread_cond_destroy() or pthread_cond_init() refers to a condition variable that is in use (for example, in a pthread_cond_wait() call) by another thread, or detects that the value specified by the cond argument to pthread_cond_init() refers to an already initialized condition variable, it is recommended that the function should fail and report an EBUSY error."</p> <p>Delete the following "may fail" EINVAL error:</p> <p>P1545 L50045 pthread_cond_init</p> <p>and add the following statement to the end of the preceding DESCRIPTION:</p> <p>"The behavior is undefined if the value specified by the attr argument to pthread_cond_init() does not refer to an initialized condition variable attributes object."</p> <p>and add the following statement to the end of the subsequent RATIONALE:</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

"If an implementation detects that the value specified by the attr argument to pthread_cond_init() does not refer to an initialized condition variable attributes object, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EINVAL error:

P1549 L50195 pthread_cond_timedwait or pthread_cond_wait

and add the following statement to the end of the preceding DESCRIPTION:

"The behavior is undefined if the value specified by the cond or mutex argument to these functions does not refer to an initialized condition variable or an initialized mutex object, respectively."

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the cond argument to pthread_cond_timedwait() or pthread_cond_wait() does not refer to an initialized condition variable, or detects that the value specified by the mutex argument to pthread_cond_timedwait() or pthread_cond_wait() does not refer to an initialized mutex object, it is recommended that the function should fail and report an

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>EINVAL error."</p> <p>Delete the following "may fail" EPERM error:</p> <p>P1549 L50200 pthread_cond_timedwait or pthread_cond_wait</p> <p>and add the following after line 50193:</p> <p style="padding-left: 40px;">"[EPERM] The mutex type is PTHREAD_MUTEX_ERRORCHECK or the mutex is a robust mutex, and the current thread does not own the mutex."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1554 L50386 pthread_condattr_destroy P1556 L50436 pthread_condattr_getclock or pthread_condattr_setclock P1558 L50487 pthread_condattr_getpshared or pthread_condattr_setpshared</p> <p>and in each case add the following statement to the end of the preceding</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

DESCRIPTION (substituting the function name(s) shown above for pthread_condattr_destroy):

"The behavior is undefined if the value specified by the attr argument to pthread_condattr_destroy() does not refer to an initialized condition variable attributes object."

and add the following statement to the end of the subsequent RATIONALE (again substituting the function names):

"If an implementation detects that the value specified by the attr argument to pthread_condattr_destroy() does not refer to an initialized condition variable attributes object, it is recommended that the function should fail and report an EINVAL error."

At page 1556 line 50662 section pthread_detach, delete

"The effect of multiple pthread_detach() calls on the same target thread is unspecified."

Delete the following "may fail" EINVAL errors:

P1566 L50669 pthread_detach
P1579 L51066 pthread_join

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name shown above for pthread_detach):</p> <p>"The behavior is undefined if the value specified by the thread argument to pthread_detach() does not refer to a joinable thread."</p> <p>and add the following statement to the end of the subsequent RATIONALE (again substituting the function name):</p> <p>"If an implementation detects that the value specified by the thread argument to pthread_detach() does not refer to a joinable thread, it is recommended that the function should fail and report an EINVAL error."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1577 L51013 pthread_setspecific</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

P1585 L51286 pthread_key_delete

and in each case add the following statement to the end of the subsequent RATIONALE (substituting the function name shown above for pthread_setspecific):

"If an implementation detects that the value specified by the key argument to pthread_setspecific() does not refer to a key value obtained from pthread_key_create() or refers to a key that has been deleted with pthread_key_delete(), it is recommended that the function should fail and report an EINVAL error."

At page 1579 line 51065 section pthread_join, delete:

"or the value of thread specifies the calling thread"

from the description of the EDEADLK error and add the following statement to the end of the preceding DESCRIPTION:

"The behavior is undefined if the value specified by the thread argument to pthread_join() refers to the calling thread."

and add the following statement to the end of the subsequent RATIONALE:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>"If an implementation detects that the value specified by the thread argument to pthread_join() refers to the calling thread, it is recommended that the function should fail and report an EDEADLK error."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1589 L51393 pthread_mutex_consistent P1591 L51454 pthread_mutex_destroy</p> <p>and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name shown above for pthread_mutex_destroy):</p> <p>"The behavior is undefined if the value specified by the mutex argument to pthread_mutex_destroy() does not refer to an initialized mutex."</p> <p>and add the following statement to the end of the subsequent RATIONALE (again</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

substituting the function names):

"If an implementation detects that the value specified by the mutex argument to `pthread_mutex_destroy()` does not refer to an initialized mutex, it is recommended that the function should fail and report an EINVAL error."

At page 1591 line 51429 section `pthread_mutex_destroy`, change:

"Attempting to destroy a locked mutex results in undefined behavior."

to:

"Attempting to destroy a locked mutex or a mutex that is referenced (for example, while being used in a `pthread_cond_timedwait()` or `pthread_cond_wait()`) by another thread results in undefined behavior."

Delete the following "may fail" EBUSY errors:

P1591 L51451 `pthread_mutex_destroy`

P1592 L51461 `pthread_mutex_init`

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the mutex argument to

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>pthread_mutex_destroy() or pthread_mutex_init() refers to a locked mutex or a mutex that is referenced (for example, while being used in a pthread_cond_timedwait() or pthread_cond_wait()) by another thread, or detects that the value specified by the mutex argument to pthread_mutex_init() refers to an already initialized mutex, it is recommended that the function should fail and report an EBUSY error."</p> <p>At page 1592 line 51487 section pthread_mutex_destroy, change:</p> <p>"Many of the error checks were made optional in order to let implementations trade off performance versus degree of error checking according to the needs of their specific applications and execution environment. As a general rule, errors or conditions caused by the system (such as insufficient memory) always need to be reported, but errors due to an erroneously coded application (such as failing to provide</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

adequate synchronization to prevent a mutex from being deleted while in use) are made optional."

to:

"Many error conditions that can occur are not required to be detected by the implementation in order to let implementations trade off performance versus degree of error checking according to the needs of their specific applications and execution environment. As a general rule, conditions caused by the system (such as insufficient memory) are required to be detected, but conditions caused by an erroneously coded application (such as failing to provide adequate synchronization to prevent a mutex from being deleted while in use) are specified to result in undefined behavior."

and append to line 51504:

"When the behavior is undefined, no error number is specified to be returned on implementations that do detect the condition. This is because undefined behavior means *<i>anything</i>* can happen, which includes returning with any value (which might happen to be a valid, but different, error number). However, since the error number might be useful to application writers when diagnosing problems during application development, a recommendation is made in rationale that

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>implementors should return a particular error number if their implementation does detect the condition."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1591 L51463 pthread_mutex_init P1608 L51984 pthread_mutexattr_destroy P1613 L52205 pthread_mutexattr_getprioceiling or pthread_mutexattr_setprioceiling (just the "attr or" part) P1616 L52299 pthread_mutexattr_getprotocol or pthread_mutexattr_setprotocol (just the "attr or" part) P1618 L52361 pthread_mutexattr_getpshared or pthread_mutexattr_setpshared P1620 L52427 pthread_mutexattr_getrobust or pthread_mutexattr_setrobust P1623 L52500 pthread_mutexattr_gettype or pthread_mutexattr_settype</p>	

¹ **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

² **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name(s) shown above for pthread_mutexattr_destroy):

"The behavior is undefined if the value specified by the attr argument to pthread_mutexattr_destroy() does not refer to an initialized mutex attributes object."

and add the following statement to the end of the subsequent RATIONALE (again substituting the function names):

"If an implementation detects that the value specified by the attr argument to pthread_mutexattr_destroy() does not refer to an initialized mutex attributes object, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EINVAL error:

P1630 L52598 pthread_once

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the once_control argument to pthread_once() does not refer to a pthread_once_t object initialized by

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB 1	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>PTHREAD_ONCE_INIT, it is recommended that the function should fail and report an EINVAL error."</p> <p>Delete the following "may fail" EBUSY errors:</p> <p>P1632 L52691 pthread_rwlock_destroy P1633 L52700 pthread_rwlock_init</p> <p>and add the following statement to the end of the subsequent RATIONALE:</p> <p>"If an implementation detects that the value specified by the rwlock argument to pthread_rwlock_destroy() or pthread_rwlock_init() refers to a locked read-write lock object, or detects that the value specified by the rwlock argument to pthread_rwlock_init() refers to an already initialized read-write lock object, it is recommended that the function should fail and report an EBUSY error."</p> <p>Delete the following "may fail" EINVAL</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

errors:

P1632 L52693 pthread_rwlock_destroy

P1636 L52791 pthread_rwlock_rdlock or
pthread_rwlock_tryrdlock

P1638 L52868 pthread_rwlock_timedrdlock
(just "The value specified ... object, or" part)

P1640 L52926 pthread_rwlock_timedwrlock
(just "The value specified ... object, or" part)

P1643 L52987 pthread_rwlock_trywrlock or
pthread_rwlock_wrlock

P1645 L53048 pthread_rwlock_unlock

and in each case add the following
statement to the end of the subsequent
RATIONALE (substituting the function name
shown above for pthread_rwlock_destroy):

"If an implementation detects that the
value specified by the rwlock argument to
pthread_rwlock_destroy() does not refer to an
initialized read-write lock object, it is
recommended that the function should fail
and report an EINVAL error."

Delete the following "may fail" EPERM error:

P1645 L53050 pthread_rwlock_unlock

and add the following statement to the end of
the subsequent RATIONALE:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>"If an implementation detects that the value specified by the rwlock argument to pthread_rwlock_unlock() refers to a read-write lock object for which the current thread does not hold a lock, it is recommended that the function should fail and report an EPERM error."</p> <p>Delete the following "may fail" EINVAL errors:</p> <p>P1633 L52703 pthread_rwlock_init P1648 L53109 pthread_rwlockattr_destroy P1650 L53166 pthread_rwlockattr_getpshared or pthread_rwlockattr_setpshared</p> <p>and in each case add the following statement to the end of the preceding DESCRIPTION (substituting the function name(s) shown above for pthread_rwlockattr_destroy):</p> <p>"The behavior is undefined if the value specified by the attr argument to</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

pthread_rwlockattr_destroy() does not refer to an initialized read-write lock attributes object."

and add the following statement to the end of the subsequent RATIONALE (again substituting the function names):

"If an implementation detects that the value specified by the attr argument to pthread_rwlockattr_destroy() does not refer to an initialized read-write lock attributes object, it is recommended that the function should fail and report an EINVAL error."

Delete the following "may fail" EBUSY error:

P1666 L53556 pthread_spin_destroy or pthread_spin_init

and add the following statement to the end of the subsequent RATIONALE:

"If an implementation detects that the value specified by the lock argument to pthread_spin_destroy() or pthread_spin_init() refers to a locked spin lock object, or detects that the value specified by the lock argument to pthread_spin_init() refers to an already initialized spin lock object, it is recommended that the function

should fail and report an EBUSY error."

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB 1	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>Delete the following "may fail" EINVAL errors:</p> <p>P1666 L53559 pthread_spin_destroy P1668 L53600 pthread_spin_lock or pthread_spin_trylock P1670 L53644 pthread_spin_unlock</p> <p>and in each case add the following statement to the end of the subsequent RATIONALE (substituting the function name shown above for pthread_spin_destroy):</p> <p>"If an implementation detects that the value specified by the lock argument to pthread_spin_destroy() does not refer to an initialized spin lock object, it is recommended that the function should fail and report an EINVAL error."</p> <p>At page 1668 line 53602 section pthread_spin_lock, delete:</p> <p>"or the calling thread already holds the</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>lock."</p> <p>from the description of the EDEADLK error and add the following statement to the end of the subsequent RATIONALE:</p> <p>"If an implementation detects that the value specified by the lock argument to pthread_spin_lock() refers to a spin lock object for which the calling thread already holds the lock, it is recommended that the function should fail and report an EDEADLK error."</p> <p>Delete the following "may fail" EPERM error:</p> <p>P1670 L53645 pthread_spin_unlock</p> <p>and add the following statement to the end of the subsequent RATIONALE:</p> <p>"If an implementation detects that the value specified by the lock argument to pthread_spin_unlock() refers to a spin lock object for which the current thread does not hold the lock, it is recommended that the function should fail and report an EPERM error."</p>	
US	<p>Page: 1596 Line: 51648 Section: pthread_mutex_getprioce</p>		te	<p>Since pthread_mutex_setprioceiling() locks the mutex, changes are needed to its description for consistency with additions to pthread_mutex_lock() such as the behaviour for robust mutexes when the owner has terminated.</p>	<p>Change</p> <p>"The pthread_mutex_setprioceiling() function shall either lock the mutex if it is unlocked, or block until it can successfully lock the mutex, then it shall change the</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
	iling			<p>Also the current description does not properly account for all of the various mutex types. For example, if the mutex type is PTHREAD_MUTEX_ERRORCHECK, detecting EDEADLK should be mandated.</p> <p>Rather than repeat all the details about robust mutexes and mutex types from the pthread_mutex_lock() page, it would be simpler to use "as if" in the description. However, the ERRORS section needs to remain separate as there are subtle differences.</p> <p>In addition to needing ENOTRECOVERABLE and EOWNERDEAD added for robust mutexes, there are some other inconsistencies in the ERRORS section on this page:</p> <ol style="list-style-type: none"> 1. Some of the errors listed under "These functions ..." only apply to pthread_mutex_setprioceiling(). 2. The EAGAIN error that can occur when locking recursive mutexes is missing. 3. The description of 	<p>mutex's priority ceiling and release the mutex. When the change is successful, the previous value of the priority ceiling shall be returned in old_ceiling.</p> <p>The process of locking the mutex need not adhere to the priority protect protocol.</p> <p>If pthread_mutex_setprioceiling() is called while holding the mutex, the result is undefined unless the mutex is of type PTHREAD_MUTEX_RECURSIVE."</p> <p>to</p> <p>"The pthread_mutex_setprioceiling() function shall attempt to lock the mutex as if by a call to pthread_mutex_lock(), except that the process of locking the mutex need not adhere to the priority protect protocol. On acquiring the mutex it shall change the mutex's priority ceiling and then release the mutex as if by a call to pthread_mutex_unlock(). When the change is successful, the previous value of the priority ceiling shall be returned in</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

pthread_mutex_setprioceiling() states that when it locks the mutex, it "need not adhere to the priority protect

protocol". For implementations where it does adhere to the priority protect protocol, the associated EINVAL error should be specified, as per the pthread_mutex_lock() page,

4. The EDEADLK error does not include the condition "A deadlock condition was detected".

While copying the text for the missing errors from the pthread_mutex_lock() page I also noticed some problems on that page:

1. The EAGAIN error is listed as applying to pthread_mutex_unlock().

2. EAGAIN should be a "shall fail" error, not "may fail".

3. The EOWNERDEAD conditions are phrased in terms of the state after the call, not the state on entry to the call. This also applies to other functions that return EOWNERDEAD.

4. The ENOTRECOVERABLE error should not say "The mutex is not locked". This is normal on failure; if we say it for one error we would have to say it for all of them (except EOWNERDEAD). This also applies to other functions that return ENOTRECOVERABLE.

old_ceiling."

Replace the ERRORS section (lines 51661-51668) with:

"These functions shall fail if:

[EINVAL] The protocol attribute of mutex is PTHREAD_PRIO_NONE.

[EPERM] The implementation requires appropriate privileges to perform the operation and the caller does not have appropriate privileges.

The pthread_mutex_setprioceiling() function shall fail if:

[EAGAIN] The mutex could not be acquired because the maximum number of recursive locks for mutex has been exceeded.

[EDEADLK] The mutex type is PTHREAD_MUTEX_ERRORCHECK and the current thread already owns the mutex.

[EINVAL] The mutex was created with the protocol attribute having the value PTHREAD_PRIO_PROTECT and the calling thread's priority is higher than the mutex's current priority ceiling, and the implementation adheres to the priority protect protocol in the process of locking the mutex.

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>Numbers 2-4 also apply to the pthread_mutex_timedlock() page.</p> <p>Finally, some of the "may fail" errors (on all 3 pages) have the same issue as the optional pthread*() ESRCH errors that are being removed by XSH ERN 215. However, there is one difference, which is that some of the errors are mandated for mutexes of type PTHREAD_MUTEX_ERRORCHECK. Therefore instead of removing these errors as was done for ESRCH, they should be turned into "shall fail" errors specifically for PTHREAD_MUTEX_ERRORCHECK mutexes.</p>	<p>[ENOTRECOVERABLE] The mutex is a robust mutex and the state protected by the mutex is not recoverable.</p> <p>[EOWNERDEAD] The mutex is a robust mutex and the process containing the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the pthread_mutex_setprioceiling() call and it is up to the new owner to make the state consistent (see [xref to pthread_mutex_lock() page]).</p> <p>The pthread_mutex_setprioceiling() function may fail if:</p> <p>[EDEADLK] A deadlock condition was detected.</p> <p>[EINVAL] The priority requested by prioceiling is out of range.</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

[EOWNERDEAD]

The mutex is a robust mutex and the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent (see [xref to pthread_mutex_lock() page]).

These functions shall not return an error code of [EINTR]."

On page 1600-1601 line 51763-51822 section pthread_mutex_lock:

After line 51763 add a new paragraph:

"If mutex does not refer to an initialized mutex object, the behavior of pthread_mutex_lock(), pthread_mutex_trylock(), and pthread_mutex_unlock() is undefined."

On line 51775 delete "The mutex is not locked."

Replace the EOWNERDEAD description at line 51777 with:

"The mutex is a robust mutex and the process containing the previous owning thread terminated while holding the mutex

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent."</p> <p>At line 51783 change:</p> <p style="padding-left: 40px;">"The current thread does not own the mutex and the mutex is a robust mutex."</p> <p>to:</p> <p style="padding-left: 40px;">"The mutex type is PTHREAD_MUTEX_ERRORCHECK or the mutex is a robust mutex, and the current thread does not own the mutex."</p> <p>Delete lines 51784-51786:</p> <p style="padding-left: 40px;">"The pthread_mutex_lock(), pthread_mutex_trylock(), and pthread_mutex_unlock() functions may fail if:</p> <p style="padding-left: 80px;">[EINVAL] The value specified by mutex does not refer to an initialized mutex object."</p>	

¹ **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

² **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

Move the EAGAIN error from line 51787 to the first "shall fail" block (after line 51770).

Replace the EOWNERDEAD description at line 51791 with:

"The mutex is a robust mutex and the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent."

At line 51794 change:

"The pthread_mutex_lock() function may fail if:

[EDEADLK] A deadlock condition was detected or the current thread already owns the mutex."

to:

"The pthread_mutex_lock() function shall fail if:

[EDEADLK] The mutex type is PTHREAD_MUTEX_ERRORCHECK and the current thread already owns the mutex.

The pthread_mutex_lock() function may fail if:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>[EDEADLK] A deadlock condition was detected."</p> <p>Delete lines 51797-51798:</p> <p>"The pthread_mutex_unlock() function may fail if:</p> <p>[EPERM] The current thread does not own the mutex."</p> <p>At line 51822 append a new paragraph to RATIONALE:</p> <p>"If an implementation detects that the value specified by the mutex argument does not refer to an initialized mutex object, it is recommended that the function should fail and report an EINVAL error."</p> <p>On page 1604-1605 line 51895-51935 section pthread_mutex_timedlock:</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

After line 51895 add a new paragraph:

"If mutex does not refer to an initialized mutex object, the behavior is undefined."

After line 51900 add:

"[EDEADLK] The mutex type is PTHREAD_MUTEX_ERRORCHECK and the current thread already owns the mutex."

On line 51908 delete "The mutex is not locked."

Replace the EOWNERDEAD description at line 51910 with:

"The mutex is a robust mutex and the process containing the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent."

Delete line 51915:

"[EINVAL] The value specified by mutex does not refer to an initialized mutex object."

Move the EAGAIN error from line 51916 to the "shall fail" block (after line 51900).

At line 51918 change:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>"[EDEADLK] A deadlock condition was detected or the current thread already owns the mutex."</p> <p>to:</p> <p>"[EDEADLK] A deadlock condition was detected."</p> <p>Replace the EOWNERDEAD description at line 51921 with:</p> <p>"The mutex is a robust mutex and the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent."</p> <p>At line 51935 change the RATIONALE from:</p> <p>"None"</p> <p>to</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

"Refer to [xref to pthread_mutex_lock()]."

On page 1549 line 50188 section pthread_cond_timedwait, change:

"[ENOTRECOVERABLE]

The state protected by the mutex is not recoverable. The mutex is not locked.

[EOWNERDEAD]

The mutex is a robust mutex and the process containing the previous owner thread terminated while holding the mutex lock. The mutex lock has been acquired and it is up to the new owner to make the state consistent."

to:

"These functions shall fail if:

[ENOTRECOVERABLE]

The state protected by the mutex is not recoverable.

[EOWNERDEAD]

The mutex is a robust mutex and the process containing the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent."

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>On page 1549 line 50197 section pthread_cond_timedwait, change:</p> <p>"The mutex is a robust mutex and the previous owning thread terminated while holding the mutex lock. The mutex lock has been acquired and it is up to the new owner to make the state consistent."</p> <p>to:</p> <p>"The mutex is a robust mutex and the previous owning thread terminated while holding the mutex lock. The mutex lock shall be acquired by the calling thread and it is up to the new owner to make the state consistent."</p>	
US	Page: 1601 Line: 51821 Section: pthread_mutex_lock		ed	<p>The Rationale section for pthread_mutex_lock() says "For further rationale on the extended mutex types, see the Rationale (Informative) volume of POSIX.1-200x."</p> <p>A cross reference to the relevant place would make it easier to locate this rationale.</p>	<p>Change</p> <p>"For further rationale on the extended mutex types, see the Rationale (Informative) volume of POSIX.1-200x."</p> <p>to</p> <p>"For further rationale on the extended mutex types, see <xref>Threads Extensions</xref</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					(on page 3472) in the Rationale (Informative) volume of POSIX.1-200x."
US	Page: 1708 Line: 54777 Section: readlink		te	With the planned change to mandate file attributes (including timestamps) for symbolic links, the description of readlink() should be updated to state that it marks for update the last data access timestamp of the symbolic link.	After line 54777 add a new paragraph: "On successful completion, readlink() shall mark for update the last data access timestamp of the symbolic link."
US	Page: 1715 Line: 55001 Section: realpath		ed	<p>XSHd2 ERN 51 has not been fully applied. The text:</p> <p>"If resolved_name is a null pointer, the behavior of realpath() is implementation-defined."</p> <p>was supposed to become:</p> <p>"If resolved_name is not a null pointer and {PATH_MAX} is not defined as a constant in the <limits.h> header, the behavior is undefined."</p> <p>The first part of the change has been made, but the change from "implementation-defined" to "undefined" has not.</p> <p>(The term "implementation-defined" is not appropriate here, as XBDd3 section 1.5 says that when a value or behaviour is implementation-defined, "the implementor shall document such a value or behavior so that it can be used correctly by an application", but passing a non-null resolved_name to realpath() when PATH_MAX is not defined as a constant in <limits.h> is not a correct use of realpath() no matter what the implementor might put in the documentation.)</p>	Change "implementation-defined" to "undefined".

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 1785 Line: 57162 Section: sig_timedwait		ge	<p>This is an example program for this interface. As discussed with Andrew off list, the example programs</p> <p>I'm submitting come from already published versions in the Linux man pages, and each was written solely by me. I assign a non-exclusive copyright to The Open Group.</p> <p>This is an example program, intended to clarify the use of this interface for the reader of the standard. Page and line numbers refer to draft 3 of the revision.</p>	<p>(Suggested: under sem_post(), page 1782, line 57099, substitute "None" by "See sem_timedwait().").</p> <p>substitute line 57162 ("None") with the following text:</p> <p>The program shown below operates on an unnamed semaphore. The program expects two command-line arguments. The first argument specifies a seconds value that is used to set an alarm timer to generate a SIGALRM signal. This handler performs a sem_post(3) to increment the semaphore that is being waited on in main() using sem_timedwait(). The second command-line argument specifies the length of the timeout, in seconds, for sem_timedwait().</p> <pre>#include <unistd.h> #include <stdio.h> #include <stdlib.h> #include <semaphore.h> #include <time.h></pre>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

```
#include <assert.h>
#include <errno.h>
#include <signal.h>

sem_t sem;

static void
handler(int sig)
{
    write(STDOUT_FILENO, "sem_post()
from handler\n", 24);
    if (sem_post(&sem) == -1) {
        write(STDERR_FILENO, "sem_post()
failed\n", 18);
        _exit(EXIT_FAILURE);
    }
}

int
main(int argc, char *argv[])
{
    struct sigaction sa;
    struct timespec ts;
    int s;

    if (argc != 3) {
        fprintf(stderr, "Usage: %s <alarm-secs>
<wait-secs>\n",
            argv[0]);
        exit(EXIT_FAILURE);
    }
}
```


Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<pre> if (sem_init(&sem, 0, 0) == -1) { perror("sem_init"); exit(EXIT_FAILURE); } /* Establish SIGALRM handler; set alarm timer using argv[1] */ sa.sa_handler = handler; sigemptyset(&sa.sa_mask); sa.sa_flags = 0; if (sigaction(SIGALRM, &sa, NULL) == -1) { perror("sigaction"); exit(EXIT_FAILURE); } alarm(atoi(argv[1])); /* Calculate relative interval as current time plus number of seconds given argv[2] */ if (clock_gettime(CLOCK_REALTIME, &ts) == -1) { </pre>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<pre> perror("clock_gettime"); exit(EXIT_FAILURE); } ts.tv_sec += atoi(argv[2]); printf("main() about to call sem_timedwait()\n"); while ((s = sem_timedwait(&sem, &ts)) == -1 && errno == EINTR) continue; /* Restart if interrupted by handler */ /* Check what happened */ if (s == -1) { if (errno == ETIMEDOUT) printf("sem_timedwait() timed out\n"); else perror("sem_timedwait"); } else printf("sem_timedwait() succeeded\n"); exit((s == 0) ? EXIT_SUCCESS : EXIT_FAILURE); } </pre>	
US	Page: 1894 Line: 60560 Section: signgam		ed	XSH should include a placeholder page for signgam, since it does not sort anywhere near its description in lgamma.	Between signbit and the sigpause placeholder, at line 60560, add a placeholder page for signgam that redirects to lgamma.	
US	Page: 2069 Line: 65714 Section: tmpfile		ed	The new EMFILE error needs CX shading.	Add CX shading.	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 2118 Line: 66978 Section: vfprintf		te	dprintf was added as part of Extended API Set Part 1, but it appears to be oversight that a corresponding vdprintf was not added. For symmetry with all other standardized *printf functions, the va_list variant vdprintf should be required with CX shading as the counterpart to dprintf.	<p>In XBD page 336 line 11693, insert:</p> <pre><CX> int vdprintf(int, const char *, va_list);</CX></pre> <p>In XBD page 337 line 11734, add dprintf to the list of functions added.</p> <p>In XSH page 2118 line 66975, rewrite the line as:</p> <pre>vdprintf, vfprintf, vprintf, vsnprintf, vsprintf</pre> <p>format output of a stdarg argument list</p> <p>In XSH page 2118 line 66979, insert:</p> <pre><CX> int vdprintf(int filedes, const char *format, va_list ap);</CX></pre> <p>In XSH page 2118 line 66988, rewrite the</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

paragraph as:

The `<CX>vdprintf()`, `</CX> vprintf()`, `vfprintf()`, `vsnprintf()`, and `vsprintf()` functions shall be equivalent to `<CX>dprintf()`, `</CX> printf()`, `fprintf()`, `snprintf()`, and `sprintf()` respectively, except that instead of being called with a variable number of arguments, they are called with an argument list as defined by `<stdarg.h>`.

In XSH page 2118 line 67014, add:

Issue 7

The `vdprintf()` function is added to complement the `dprintf()` function from The Open Group Technical Standard, 2006, Extended API Set Part 1.

Add to XSH page 494 section 2.9.x May occur cancellation list
`vdprintf`

Add `vdprintf` to XRAT page 3394 new functions list

Add `vdprintf` to XRAT page 3608 appendix E line 124250
`POSIX_DEVICE_IO_EXT`

Add a new ptr page

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 2135 Line: 67439 Section: wait		ed	Wrong indentation.	Reduce the indentation for lines 67439-67442 so that it is the same as the lines before and after them.	
US	Page: 2140 Line: 67663 Section: waitid		ed	Inaccurate change history.	Change "recommending that the waitid() function not be used" to "recommending that the waitid() function not be used with idtype equal to P_ALL"	
US	Page: 2214 Line: 69732-69743 Section: write		ed	The OB XSR shading ends prematurely in the middle of the write() STREAMS discussion.	Extend OB XSR shading through line 69743.	
US	Page: 2215 Line: 69801,69802 Section: write		ed	Because pwrite() no longer depends on XSI, these lines should not be XSI shaded.	Remove XSI shading.	
US	#page 970 line 33049 section		te	The proposed specification for getline/getdelim is not consistent with existing practice on trailing NULs, or when *lineptr is NULL but *n is	At line 33050, replace: If *n is non-zero, the application shall ensure	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

getdelim

uninitialized. Additionally, the documentation is not clear about the behavior when end-of-file is reached. Finally, the impact on atime should be explicit.

that *lineptr points to an object of size at least *n bytes.

with:

If *n is non-zero, the application shall ensure that *lineptr either points to an object of size at least *n bytes, or is a null pointer.

At line 33052, replace:

The size of the object pointed to by *lineptr shall be increased to fit the incoming line, if it isn't already large enough. The characters read shall be stored in the string pointed to by the lineptr argument.

with:

The size of the object pointed to by *lineptr shall be increased to fit the incoming line, if it isn't already large enough, including room for the delimiter and a terminating NUL. The characters read, including any delimiter, shall be stored in the string pointed to by the lineptr argument, and a terminating NUL added when the delimiter or end of file is encountered.

At line 33056, add a new paragraph:

The getdelim() and getline() functions may mark the last data access timestamp of the

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>file associated with stream for update. The last data access timestamp shall be marked for update by the first successful execution of <code>fgetc()</code>, <code>fgets()</code>, <code>fread()</code>, <code>fscanf()</code>, <code>getc()</code>, <code>getchar()</code>, <code>getdelim()</code>, <code>getline()</code>, <code>gets()</code>, or <code>scanf()</code> using stream that returns data not supplied by a prior call to <code>ungetc()</code>.</p> <p>At line 33058, replace:</p> <p>Upon successful completion, the <code>getdelim()</code> function shall return the number of characters written into the buffer, including the delimiter character if one was encountered before EOF. Otherwise, it shall return -1 and set <code>errno</code> to indicate the error.</p> <p>with:</p> <p>Upon successful completion, the <code>getline()</code> and <code>getdelim()</code> functions shall return the number of characters written into the buffer, including the delimiter character if one was encountered before EOF, but excluding the terminating</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

NUL character. If no characters were read, and the end-of-file indicator for the stream is set, or if the stream is at end-of-file, the end-of-file indicator for the stream shall be set and the function shall return -1. If an error occurs, the error indicator for the stream shall be set, and the function shall return \hat{a}^1 and set `errno` to indicate the error.

Insert prior to line 33085:

```
if (ferror(fp)) {  
    /* handle error */  
}
```

At line 33091, add two new paragraphs:

The `ferror()` or `feof()` functions should be used to distinguish between an error condition and an end-of-file condition.

Although a NUL terminator is always supplied after the line, note that `strlen(*lineptr)` will be smaller than the return value if the line contains embedded NUL characters.

For each of the following functions, add `getdelim()` and `getline()` to the list of functions that may modify the last data access timestamp:

`fgetc`: line 27711 (3rd line)

`fgets`: line 27832 (3rd line)

`fread`: line 30033 (3rd line)

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					fscanf: line 30725 (3rdline) gets: line 35032 (3rdline)	
US	Page: 2286 Line: 72498 Section: continue		ed	Typo in draft 3's attempt to convert echo to printf usage.	Replace line 72498: printf "\$s" is not a directory.\n "\$i" with: printf "%s" is not a directory.\n "\$i"	
US	Page: 2427 Line: 77976 Section: c99		te	In the c99 synopsis the "pathname" within the repeated group needs to be optional, otherwise a simple "c99 file.c -l m" doesn't fit the pattern. There are also some editorial problems with the synopsis (-D should be -l, spacing, and a missing right bracket).	Change [pathname[-D directory [-L directory] [-l library]]... to [[pathname] [-l directory] [-L directory] [-l library]]...	
US	Page: 2431 Line: 78152 Section: c99			There is a minor editorial mistake in the application of XCUd2 ERN 68. There is also a typo on P2434 L78317.	Change "as the last" to "after the last". On P2434 L78317 change "operaneds" to "operands".	
US	Page: 2483 Line: 80185 Section: cp		ed	cp -r was not completely removed. There are now only three synopses instead of four for cp; and step 4 no longer needs an extra sublevel, since it can now only be reached via the use of -R.	At line 80185, replace the phrase: The third and fourth synopsis forms are denoted...	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					<p>with:</p> <p>The third synopsis form is denoted...</p> <p>At lines 80270, replace:</p> <p>4. Otherwise, the following steps shall be taken:</p> <p>a. If the -R option was specified, the following steps shall be taken:</p> <p>with:</p> <p>4. Otherwise, the -R option was specified, and the following steps shall be taken:</p> <p>At lines 80272-80284, renumber i., ii., and iii. to a., b., and c., and reduce the indentation accordingly.</p>	
US	Page: 2531 Line: 81991 Section: df		ed	The "1024-blocks" columns header specified to be written by df has been changed to "1 024-blocks". Since this text is part of an output format specification, it cannot be subject to the usual editorial convention for other text; it must show precisely the text to be output.	Change "1 024" to "1024".	
US	Page: 2676 Line: 87767 Section: find		ed	Incomplete change history entry for find.	Change "The description of the -name primary is revised and a new example added." to "The description of the -name primary is	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					revised and the -path primary is added (with a new example)."	
US	Page: 2765 Line: 91114 Section: ln		te	<p>A change has been made to the link() function to make it implementation-defined whether or not link() follows symlinks. For consistency the equivalent change should be made for the ln utility, which is currently still required to follow symbolic links.</p> <p>In C programs application writers can choose which behaviour they want by using the linkat() function instead of link() The same choice should be made available in shell scripts by adding -L and -P options to ln.</p>	<p>In both lines of the SYNOPSIS change:</p> <p>"ln [-fs]"</p> <p>to</p> <p>"ln [-fs] [-L -P]"</p> <p>At line 91143 replace item 3 with:</p> <p>"3. If source_file is a symbolic link:</p> <p style="padding-left: 40px;">a. If the -P option is in effect, actions shall be performed equivalent to the linkat() function with source_file as the path1 argument, the destination path as the path2 argument, AT_FDCWD as the fd1 and fd2 arguments, and zero as the flag argument.</p> <p style="padding-left: 40px;">b. If the -L option is in effect, actions shall be performed equivalent to the linkat() function with source_file as the path1 argument, the destination path as the path2</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

argument, AT_FDCWD as the fd1 and fd2 arguments, and AT_SYMLINK_FOLLOW as the flag argument.

The ln utility shall do nothing more with source_file and shall go on to any remaining files."

On line 91152 change "option" to "options".

After line 91153 add:

"-L For each source_file operand that names a file of type symbolic link, create a (hard) link to the file referenced by the symbolic link.

-P For each source_file operand that names a file of type symbolic link, create a (hard) link to the symbolic link itself."

At line 91154 append to the -s description:

"If the -s option is specified, the -L and -P options shall be silently ignored."

Add two new paragraphs after line 91154:

"Specifying more than one of the mutually-exclusive options -L and -P shall not be considered an error. The last option specified shall determine the behavior of the utility

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>(unless the -s option causes it to be ignored).</p> <p>If the -s option is not specified and neither a -L nor a -P option is specified, it is implementation-defined which of the -L and -P options will be used as the default.</p> <p>Add to the end of RATIONALE (line 91229):</p> <p>"The -L and -P options allow for implementing both common behaviors of the In utility. Earlier versions of this standard did not specify these options and required the behavior now described for the -L option. Many systems by default or as an alternative provided a non-conforming In utility with the behavior now described for the -P option. Since applications could not rely on In following links in practice, the -L and -P options were added to specify the desired behavior for the application.</p> <p>The -L and -P options are ignored when -s is specified in order to allow an alias to be created to alter the default behavior when</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					creating hard links (for example: alias ln='ln -L'). They serve no purpose when -s is specified, since source_file is then just a string to be used as the contents of the created symbolic link and need not exist as a file."
US	Page: 2788 Line: 91977 Section: ls		ed	<p>The ls synopsis is not up to date with respect to some changes made to the ls options. Specifically, -f and -s are no longer XSI, and -k has been added.</p> <p>The -x option is also missing from the synopsis.</p> <p>And there is a typo on P2792 L92186.</p>	<p>Move 'f' and 's' from the XSI shaded block of option letters to the initial (unshaded) block of option letters. Add 'k' and 'x' to the latter.</p> <p>This should produce:</p> <p>ls [-ACFRSacdifiklmnpqrstux1] [-H -L] [-go] [file...]</p> <p>where [-go] is XSI shaded.</p> <p>On P2792 L92186 change "of the -k option" to "if the -k option".</p>
US	Page: 2800 Line: 92498 Section: m4		te	<p>Several problems exist in how m4 edits were made for draft 3.</p> <p>The edits of ERN-130 were lost on 'eval' by the edits of ERN-137.</p> <p>The addition of 'mkstemp' by interp 117 also falls under the intent of ERN-130.</p> <p>There is a typo in the Issue 7 History, listing ERN-118 twice.</p>	<p>At line 92498, add one more sentence:</p> <p>The behavior is unspecified if eval is not immediately followed by a left parenthesis.</p> <p>At line 92546, add one more sentence:</p> <p>The behavior is unspecified if mkstemp is not immediately followed by a left parenthesis.</p> <p>At line 92708, replace "SD5-XCU-ERN-118" with "SD5-XCU-ERN-119".</p>
US	Page: 2983		te	Some changes are needed to the pwd utility for	In the description of the -L option (line 99856)

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB 1	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
	Line: 99849 Section: pwd			<p>consistency with the changes being made to <code>getcwd()</code> by XSH ERN 213.</p> <p>A complication for <code>pwd</code> is the <code>PWD</code> variable. The steps performed by <code>cd</code> can produce <code>PWD</code> values longer than <code>PATH_MAX</code>. (This is clear for <code>cd -L</code>, but for <code>cd -P</code> step 10 says that <code>PWD</code> is set to a "pathname" - a fix for this is also included below.) Since <code>PWD</code> can be too long to be a "pathname", the descriptions of <code>PWD</code> on the <code>pwd</code> page and elsewhere should not state that it is a pathname.</p> <p>The description of the <code>pwd -L</code> option contains a condition on whether <code>PWD</code> is "an absolute pathname of the current directory that does not contain the filenames <code>dot</code> or <code>dot-dot</code>". If <code>PWD</code> is longer than <code>PATH_MAX</code> then it is not a "pathname" and the condition is automatically false. If we want the value of <code>PWD</code> to be written when it is longer than <code>PATH_MAX</code> but otherwise satisfies the condition, then we need to change the wording of the condition. However, we need to consider the implications of doing that. Assume for a moment</p>	<p>insert:</p> <p>Otherwise, if the <code>PWD</code> environment variable contains a pathname of the current directory that is longer than <code>PATH_MAX</code> bytes including the terminating null, and the pathname does not contain any components that are <code>dot</code> or <code>dot-dot</code>, it is unspecified whether <code>pwd</code> writes this pathname to standard output or behaves as if the <code>-P</code> option had been specified.</p> <p>before:</p> <p>Otherwise, the <code>-L</code> option shall behave ...</p> <p>Replace the description of the <code>-P</code> option (line 99858) with:</p> <p>The pathname written to standard output shall not contain any components that refer to files of type symbolic link. If there are multiple pathnames that the <code>pwd</code> utility could write to standard output, one beginning with a single slash and one or more beginning with two</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

that we make this change. What would the implementation of pwd need to do to test the condition? For the shell built-in version of pwd there is actually no need to test the condition directly, since the value of PWD is under the control of the shell (and the standard says that if an application sets or unsets the value of PWD, the behavior of pwd is unspecified). However, the exec-able version of pwd does need to test the condition (because the condition could be made false through the use of chdir()), and if required to handle PWD values longer than PATH_MAX then it would not be able to do a simple stat() of the PWD value. I doubt if any implementations do more than the simple stat(), so by requiring pwd to handle longer PWD values we would be forcing all implementors to modify pwd. The changes below allow either behaviour.

A related issue is the handling of the PWD value from the environment by the shell when it is executed. As with the exec-able version of pwd, it needs to test whether PWD refers to the current directory in order to decide whether to use the inherited value or reinitialise it. The changes proposed below handle this the same way as for pwd -L. Note that these changes are also intended to supersede XCU ERN 88 (whose status is unclear - it is shown as "Accept as marked" but the accompanying comments to not specify any change to the text). Another addition to the affected text is to copy some text from cd step 10 about permission problems when setting PWD.

slashes, then it shall write the pathname beginning with a single slash. The pathname shall not contain any unnecessary slashes after the leading one or two slashes.

Replace the APPLICATION USAGE section (line 99900) with:

If the pathname obtained from pwd is longer than {PATH_MAX} bytes, it could produce an error if passed to cd. Therefore in order to return to that directory it may be necessary to break the pathname into sections shorter than PATH_MAX and call cd on each section in turn (the first section being an absolute pathname and subsequent sections being relative pathnames).

At page 2443 line 78636 section cd, change:

the PWD environment variable shall be set to an absolute pathname for the current working directory and shall not contain filename components that, in the context of pathname resolution, refer to a file of type symbolic link.

to:

the PWD environment variable shall be set to the string that would be output by pwd -P.

Replace the PWD description at page 3077 line 103479 section sh with:

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>This variable shall represent an absolute pathname of the current working directory. Assignments to this variable may be ignored.</p> <p>Replace the PWD description at page 2252 line 71151 section 2.5.3 with:</p> <p>Set by the shell and by the cd utility. In the shell the value shall be initialized from the environment as follows. If a value for PWD is passed to the shell in the environment when it is executed, the value is an absolute pathname of the current working directory that is no longer than PATH_MAX bytes including the terminating null byte, and the value does not contain any components that are dot or dot-dot, then the shell shall set PWD to the value from the environment. Otherwise, if a value for PWD is passed to the shell in the environment when it is executed, the value is an absolute pathname of the current working directory, and the value does not contain any components that are dot or dot-dot, then it is</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

unspecified whether the shell sets PWD to the value from the environment or sets PWD to the pathname that would be output by pwd -P. Otherwise, the sh utility sets PWD to the pathname that would be output by pwd -P. In cases where PWD is set to the value from the environment, the value can contain components that refer to files of type symbolic link. In cases where PWD is set to the pathname that would be output by pwd -P, if there is insufficient permission on the current working directory, or on any parent of that directory, to determine what that pathname would be, the value of PWD is unspecified. Assignments to this variable may be ignored. If an application sets or unsets the value of PWD, the behaviors of the cd and pwd utilities are unspecified.

Cross-volume change to XBD
page 164 line 5602 section 8.3, change:

This variable shall represent an absolute pathname of the current working directory. It shall not contain any filename components of dot or dot-dot. The value is set by the cd utility.

to:

This variable shall represent an absolute pathname of the current working directory. It shall not contain any components that are dot or dot-dot. The value is set by the cd

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
US	Page: 3040 Line: 102064 Section: read		te	The description of the read utility describes prompting behaviour for here documents. Since here documents are not more intimately related to read than to any other utility, their prompting behaviour should be described at the place where they are introduced.	<p>utility, and by the sh utility during initialization.</p> <p>Remove line 102064</p> <p>Change lines 102060 to 102063 to become just one sentence:</p> <p>If standard input is a terminal device and the invoking shell is interactive, read shall prompt for a continuation line when it reads an input line ending with a backslash <newline>, unless the -r option is specified.</p> <p>On line 102112, delete from ", or" to the full stop .</p> <p>On page 2261, section 2.7.4 (Here-Document), after line 71499, insert a new paragraph:</p> <p>When a here document is read from a terminal device and the shell is interactive, it shall write the contents of the variable PS2, processed as described in section 2.5.3 (Shell Variables) to standard error before reading</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

					each line of input until the delimiter has been recognized.	
US	Page: 3045 Line: 102283 Section: renice		ed	Typo	Change "forma" to "forms".	
US	Page: 3120 Line: 105213 Section: tail		ge	<p>Interpretation AI-100 added the following to the description of the tail -c option:</p> <p>"The application shall ensure that if the sign of the number option-argument is '+', the number option-argument is a positive decimal integer."</p> <p>When this text is read as part of AI-100 it is clear from the context that its purpose is to forbid the use of "-c +0". However, when it is read just as part of the -c description in the standard, it is at first confusing; it takes time to realise that the only aspect of "positive" that matters is that positive numbers are not zero.</p> <p>Since this change was made before draft 3, amending it is out of scope according to the narrowing-down rules. Hence this aardvark is a comment, not an objection. I believe the group can, at their discretion, accept such comments and I hope that a minor clarification to this text will be seen as worthwhile and will be made.</p>	<p>On lines 105214 and 105332 change "positive" to "non-zero".</p> <p>On line 105207 change</p> <p>"decimal integer whose sign affects"</p> <p>to</p> <p>"decimal integer, optionally including a sign. The sign shall affect"</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of comment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
				<p>There is also another change to the -c description that could be made at the same time. It says:</p> <p>"The application shall ensure that the number option-argument is a decimal integer whose sign affects the location in the file, measured in bytes, to begin the copying:"</p> <p>The sentence needs to be rearranged so that it doesn't appear to place a requirement on the application to ensure that the sign affects the location in the file.</p>		
US	Page: 3313 Line: 112244 Section: A.1.1		ed	<p>The change in the name the standard calls itself has resulted in the following statement in the XRAT introduction:</p> <p>"POSIX.1-200x is known as POSIX.1"</p>	<p>Change</p> <p>"The family of standards extends to many topics; POSIX.1-200x is known as POSIX.1 and consists of both operating system interfaces and shell and utilities."</p> <p>to</p> <p>"The family of standards extends to many topics; POSIX.1 consists of both operating system interfaces and shell and utilities."</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

US	Page: 3318 Line: 112465 Section: Portability		ed	<p>The example given on lines 112465-112467 is obsolete, in that the THR option no longer exists:</p> <p>For example, if functionality is marked with THR in the margin, it will be available on all systems supporting the Threads option, but may not be available on some others.</p>	<p>Change the example to use one of the margin codes that is still in effect. I have chosen RPP here, but any valid margin code could be used.</p> <p>Change lines 112465-112467 to:</p> <p>For example, if functionality is marked with RPP in the margin, it will be available on all systems supporting the Robust Mutex Priority Protection option, but may not be available on some others.</p>	
US	Page: 3337 Line: 113258 Section: A.3		ge	<p>The rationale associated with the "symbolic link" definition needs to be updated to account for the planned change to mandate inodes for symbolic links, and with other changes related to symbolic links that have already been made since the 2004 edition of the standard.</p> <p>There is also a problem with the statement in the third paragraph:</p> <p>"a trailing slash is considered to be the final component of a pathname"</p> <p>The term "pathname component" is defined to be a synonym for "filename", and a filename cannot contain a slash, therefore a trailing slash cannot be a "component of a pathname" (i.e. pathname component).</p>	<p>Replace the first five paragraphs under "Symbolic Link" (lines 113258-113292) with the following:</p> <p>"Earlier versions of this standard did not require symbolic links to have attributes such as ownership and a file serial number. This was because the 4.4 BSD implementation did not have them, and it was expected that other implementations may wish to do the same. However, experience with 4.4 BSD has shown that symbolic links implemented in this way cause problems for users and application writers, and later BSD systems have reverted to using inodes to implement symbolic links. Allowing no-inode symbolic links also caused problems in the standard. For example, leaving the st_ino value for symbolic links unspecified meant that the common technique of comparing the st_dev and st_ino values for two pathnames to see if they refer to the same file could only be used</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB 1	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>with stat() in conforming applications and not with lstat(). The standard now requires symbolic links to have meaningful values for the same struct stat fields as regular files, except for the file mode bits in st_mode. Historically the file mode bits were unused (the contents of a symbolic link could always be read), but implementations differed as to whether the file mode bits (as returned in st_mode or reported by ls -l) were set according to the umask or just to a fixed value such as 0777. Accordingly the standard requires the file mode bits to be ignored by readlink() and when a symbolic link is followed during pathname resolution, but leaves the corresponding part of the value returned in st_mode unspecified.</p> <p>Historical implementations were followed when determining which interfaces should apply to symbolic links. Interfaces that historically followed symbolic links include chmod(), stat() and utime(). Interfaces that historically did not follow symbolic links include lstat(), rename(), remove(), rmdir(),</p>	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

and unlink(). For chown() and link() historical implementations differed. POSIX.1-200x inherited the lchown() function from the Single UNIX Specification version 2, and therefore requires chown() to follow symbolic links. Earlier versions of this standard required link() to follow symbolic links, but with the addition of the linkat() function (which has a flag to indicate whether to follow symbolic links) both behaviors are now allowed for link().

When the final component of a pathname is a symbolic link, the standard requires that a trailing slash causes the link to be followed. This is the behavior of historical implementations. For example, for /a/b and /a/b/, if /a/b is a symbolic link to a directory, then /a/b refers to the symbolic link, and /a/b/ refers to the directory to which the symbolic link points."

Replace the second paragraph under "Third Domain" (lines 113344-113351) with:

"The intention of the Shell and Utilities volume of POSIX.1-200x is that the operation that the utility is performing is applied to the symbolic link itself, if that operation is applicable to symbolic links. If the operation is not applicable to symbolic links, the symbolic link should be ignored. Specifically, by default, no change should be made to the file referenced by the symbolic link."

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>At line 113389 delete:</p> <p>"The only standard utilities that require the -P option are cd and pwd; see the note below."</p> <p>At line 113413 delete:</p> <p>"Earlier versions of this standard did not require the pwd utility to be a built-in utility. Now that pwd is required to set an environment variable in the current shell execution environment, it must be a built-in utility."</p> <p>At line 113419 change:</p> <p>"Symbolic links in 4.4 BSD do not have"</p> <p>to:</p> <p>"Symbolic links in 4.4 BSD did not have"</p>	
US	Page: 3352 Line: 113842 Section:		ge	The statement made on line 113842-113844 has always been wrong.	change from As of September 2000, 24 leap seconds had	

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

Seconds

<quote>

As of September 2000, 24 leap seconds had been added to UTC since the Epoch, 1 January, 1970. Historically, one leap second is added every 15 months on average, so this offset can be expected to grow steadily with time. </quote>

According to NIST, at this point (i.e. today), 23 leap seconds have been added. In September 2000, only 22 had been added.

From NIST:

The first leap second was inserted into the UTC time scale on June 30, 1972. Leap seconds are used to keep the difference between UT1 and UTC to within 0.9 s. The table below lists all leap seconds that have already occurred, or are scheduled to occur.

All leap seconds listed in the table are positive leap seconds, which means an extra second is inserted into the UTC time scale. The sequence of events is:

23h 59m 59s - 23h 59m 60s - 00h 00m 00s

NOTE: No leap second will be added at the end of December 2007

been added to UTC since the Epoch, 1 January, 1970. Historically, one leap second is added every 15 months on average, so this offset can be expected to grow steadily with time.

to

As of December 2007, 23 leap seconds had been added to UTC since the Epoch, 1 January, 1970. Historically, one leap second is added every 15 months on average, so this offset can be expected to grow with time.

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)																																
MB ¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment ²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition																																
				<p>Leap Seconds Inserted into the UTC Time Scale</p> <table> <thead> <tr> <th>Date</th> <th>MJD</th> </tr> </thead> <tbody> <tr><td>2005-12-31</td><td>53735</td></tr> <tr><td>1998-12-31</td><td>51178</td></tr> <tr><td>1997-06-30</td><td>50629</td></tr> <tr><td>1995-12-31</td><td>50082</td></tr> <tr><td>1994-06-30</td><td>49533</td></tr> <tr><td>1993-06-30</td><td>49168</td></tr> <tr><td>1992-06-30</td><td>48803</td></tr> <tr><td>1990-12-31</td><td>48256</td></tr> <tr><td>1989-12-31</td><td>47891</td></tr> <tr><td>1987-12-31</td><td>47160</td></tr> <tr><td>1985-06-30</td><td>46246</td></tr> <tr><td>1983-06-30</td><td>45515</td></tr> <tr><td>1982-06-30</td><td>45150</td></tr> <tr><td>1981-06-30</td><td>44785</td></tr> <tr><td>1979-12-31</td><td>44238</td></tr> </tbody> </table>	Date	MJD	2005-12-31	53735	1998-12-31	51178	1997-06-30	50629	1995-12-31	50082	1994-06-30	49533	1993-06-30	49168	1992-06-30	48803	1990-12-31	48256	1989-12-31	47891	1987-12-31	47160	1985-06-30	46246	1983-06-30	45515	1982-06-30	45150	1981-06-30	44785	1979-12-31	44238		
Date	MJD																																					
2005-12-31	53735																																					
1998-12-31	51178																																					
1997-06-30	50629																																					
1995-12-31	50082																																					
1994-06-30	49533																																					
1993-06-30	49168																																					
1992-06-30	48803																																					
1990-12-31	48256																																					
1989-12-31	47891																																					
1987-12-31	47160																																					
1985-06-30	46246																																					
1983-06-30	45515																																					
1982-06-30	45150																																					
1981-06-30	44785																																					
1979-12-31	44238																																					

1 **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

2 **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

				<p>1978-12-31 43873</p> <p>1977-12-31 43508</p> <p>1976-12-31 43143</p> <p>1975-12-31 42777</p> <p>1974-12-31 42412</p> <p>1973-12-31 42047</p> <p>1972-12-31 41682</p> <p>1972-06-30 41498</p> <p>(source http://tf.nist.gov/pubs/bulletin/leapsecond.htm)</p>		
US	Page: 3394 Line: 115461 Section: B.1.1		ed	The list of new functions in Issue 7 omits open_wmemstream.	In sorted order in the table at line 115441, add open_wmemstream().	
US	Page: 3396 Line: 115543 Section: B.1.1		ed	Since utime() is a base function, not XSI, it should be in the list of obsolescent base functions.	Move utime() from the 2nd table on the page to the 1st table.	
US	Page: 3520 Line: 0 Section: B.3.1		ge	<p>Replying to your resolution detail "would welcome detailed wording instructions" from my comment (comment #10 in previous ballot):</p> <p>I have no access to all P1003.1 versions, so I cannot provide such wording. But I propose to simply include B.3.1 from all previous version of P1003.1 standard.</p>	<p>Add a new section B.3.2 System Interfaces removed from the Previous Revision</p> <p>The following system interfaces, headers, and external variables were removed in the previous revision of this standard:</p> <p>advance(), brk(), chroot(), compile(), cuserid(), gamma(), getdtablesize(), getpagesize(), getpass(), getw(), putw(),</p>	

Template for National Body comments and Proposed Editors Disposition

Date:	Document: ISO/IEC JTC 1 N
-------	----------------------------------

1	2	(3)	4	5	(6)	(7)
MB¹	Clause No./ Subclause No./ Annex (e.g. 3.1)	Paragraph/ Figure/Table/ Note (e.g. Table 1)	Type of com- ment²	Comment (justification for change) by the MB	Proposed change by the MB	Proposed Editors Disposition
					<p>re_comp(), re_exec(), regcmp(), regex(), sbrk(), sigstack(), step(), ttyslot(), valloc(), wait3(), <re_comp.h>, <regexp.h>, <varargs.h>, loc1, __loc1, loc2, locs 1</p> <p>Make Exclusion of Utilities in C.4 to become C.4.3</p> <p>Add a new section C.4.1 Utilities removed from this Revision</p> <p>None.</p> <p>Add a new section C.4.2 Utilities removed from the Previous Revision</p> <p>The following utilities were removed in the previous revision of this standard:</p> <p>calendar, cancel, cc, col, cpio, cu, dircmp, dis, egrep, fgrep, line, lint, lpstat, mail, pack, pcat, pg, spell, sum, tar, unpack, uulog, uuname, uupick, uuto</p>	
US	Page: 3609 Line: 124279		ed	Line 124279 mentions the function futimesat(), but this function has been renamed utimesnsat() (see	At page 3609 line 124279: Change: "futimesat()"	

¹ **MB** = Member body (enter the ISO 3166 two-letter country code, e.g. CN for China; comments from the ISO/CS editing unit are identified by **)

² **Type of comment:** **ge** = general **te** = technical **ed** = editorial – For technical comments, please indicate whether your comment is a MAJOR or MINOR technical comment.

NOTE Columns 1, 2, 4, 5 and 6 are compulsory.

	Section: E.1			XSH line 31880).	to "utimesnsat()"	
					All line numbers refer to draft 3 of the specification.	
US	ALL		ge	There are several approved interpretations and defect reports. that have been approved against IS 9945:2003 (including TC1).	Include all approved interpretations and defect reports against the old version of the standard. See www.opengroup.org/austin/interps and www.opengroup.org/austin/aardvark .	