

ISO/TC97/SC22

W.614/N034

X3J11/87-249

11 Dec 87

Project 381-D

X3J11 Meeting No. 19

07-11 Dec 87

The Driskill Hotel
604 Brazos
Austin TX 78701
(800) 252-9367

1. Opening Activities (Brodie)

1.1 Opening Comments

The meeting convened at 9:00 a.m. on Monday, 07 Dec 87, by Chairman Jim Brodie, who welcomed participants new and old to the nineteenth meeting of X3J11, Programming Language C (X3 Project 381D). Host for the meeting was Tymlabs (Khushf) with an assist from Motorola (Paton). Tom Plum served as Vice Chairman. P.J. Plauger served as Secretary.

Brodie emphasized that the goals for this meeting is to be prepared to vote on sending out the draft for second public review. He observed that there is still much work to do, given the number of papers left to review. He thanked Prosser, Plum, and Gwyn for preparing the response document between meetings, and Hudson and Prosser for editing the Rationale and Draft.

1.2 Housekeeping

Brodie informed the committee that Tymlabs was serving as host for this meeting, so any requests for copies of documents should be funneled through Khushf. Motorola was also providing assistance.

1.3 Approval of Previous Minutes

Plauger submitted the minutes of the previous meeting (87-196) for amendment or correction.

Meloy observed that the reference to Paris on p. 8 should be to Boulder.

Meloy corrected her comment on p. 20 to say she required changes in setlocale, not just clarification.

With these changes, the minutes were accepted.

1.4 Approval of Agenda

Brodie submitted a Preliminary Agenda (87-214) for approval. Several small changes were accepted.

Plum/Gwyn

"Move we schedule sessions for Tuesday and Wednesday evenings, from 7:30 to 9:30."

Motion carried, lots/2.

With these changes, the Agenda was approved. The revised Agenda (87-214R) is Attachment II to these minutes.

1.5 Introduction of New Participants

All attendees introduced themselves to the Committee.

1.6 Next Meeting

The next meeting will be held in Nashua NH, at the Sheraton Tara Hotel, 18-22 Apr 88, hosted by Honeywell. Osten reported that a car was desirable, but not absolutely necessary. Limousine service is available from Logan Airport in Boston.

1.7 Procedures for this Meeting

Brodie described the schedule as uncomfortably tight, but necessary if we are to cover everything. He warned that he would enforce starting times rigorously, and would encourage less discussion of issues than in the past.

1.8 Distribution of New Documents

A number of documents were distributed.

1.9 Review of Action Items

Brodie scanned the previous minutes (87-196) for action items noted.

1.10 X3 Notices (Brodie)

Brodie passed on various bits of news from X3.

1.11 Roll Call

An attendance sheet was circulated and all attendees were asked to verify or write down their names and addresses. Those in attendance are listed in Attachment I. Forty-two attendees were deemed eligible to vote, having attended at least one of the preceding two meetings and having expressed the intention of becoming voting members. These were:

Graham Andrews, Edinburgh Portable Compilers
Jim Baumbach, Advanced Computer Techniques
Don Bixler, Unisys
Oliver Bradley, SAS Institute, Inc.

Jim Brodie, Jim Brodie & Associates
Steve Davies, Concurrent Computer Corporation
Shawn Elliott, IBM
Frank Farance, Farance Inc.
Don Fosbury, Control Data
Douglas Gwyn, U.S. Army Ballistic Research Lab.
Randy Hudson, Intermetrics Inc.
Paul Jackson, HCR Corp.
Rex Jaeschke, DEC Professional
Patricia Jenkins, Masscomp
Bob Jervis, Borland International
Gary Jeter, Harris Computer Systems Div.
Andrew Johnson, Prime Computer, Inc.
Larry Jones, SDRC
Monika Khushf, Tymlabs
Jacklin Kotikian, Masscomp
Tom MacDonald, Cray Research, Inc.
Michael Meissner, Data General
Sue Meloy, Hewlett-Packard
Randy Meyers, Digital Equipment Corp.
Daniel Mickey, Chemical Abstracts
Thomas Osten, Honeywell Bull
Michael Paton, Motorola, Inc.
John Peyton, Apollo Computer Inc.
P.J. Plauser, Whitesmiths, Ltd.
Tom Plum, Plum Hall
Courtney Prodehl, Mark Williams Co.
David Prosser, AT&T
Chuck Rasbold, Lawrence Livermore National Laboratory
Henry Richardson, Tandem Computers
Larry Rosenthal, Sierra Systems
Fred Rozakis, Wang
Honey Schrecker, International Computers Ltd.
Rick Schubert, NCR Corp.
Carl Sutton, Tektronix
Neil Weidenhofer, Amdahl Corp.
Dave Weil, Microsoft Corp.
Jim Williams, Naval Research Lab

2. Liaison Activities

2.1 IEEE 1003

Gwyn reported that IEEE 1003 is currently voting to accept Draft 12 as a full standard. The letter ballot closes 15 Dec 87. He identified a few remaining areas of conflict with the X3J11 Draft. There are still many more definitions in POSIX <limits.h> than in ours, and there is a semantic difference in how signal handlers are entered.

Prosser reported a difference in PATH_MAX as well.

Plauser reported that the National Bureau of Standards is moving ahead with a FIPS standard for POSIX, even before that standard is frozen. He attended a Nov 87 meeting at Gaithersburg MD, where NBS sought advice on how to pin down

some of the implementation defined features of POSIX. A verification suite will soon be out for beta testing, and a trial use FIPS standard is scheduled for early 88.

2.2 ISO (Plaugar)

Plaugar reported that ISO WG14 met 16-17 Nov 87 in Amsterdam NL, to draft recommendations to X3J11 on how to change the Draft to meet the needs of the ISO community. He submitted the minutes of the meeting (87-247) and the WG14 recommendations to X3J11 (87-248).

Plaugar reminded the committee that AFNOR has stated that it will vote against an ISO draft that fails to honor grouping of parentheses or that fails to complete support for multi-byte characters. These and other points important to WG14 are summarized in 87-248.

2.3 SQL (Brodie)

Brodie reported that he took the liberty of approving the SQL C binding, on behalf of X3J11. The formal vote to do so was overlooked at the Framingham meeting, and SQL was prepared to drop the C binding in the absence of approval from X3J11. SQL accepted Brodie's representation. There was no objection from the committee to this action.

2.4 X/OPEN

Schrecker reported that X/OPEN is mostly preoccupied with POSIX issues at the moment. They did, however, express a dislike for the monetary capabilities recently added, and have suggested an alternative (87-229).

2.5 Other Groups (Brodie)

Plaugar submitted recent correspondence from the Japanese on multi-byte character support (87-246).

3. Reports of the Redactors

Hudson presented an updated draft of the Rationale (87-219). He reported that there are a few open issues for which he still needs wording.

Prosser presented an updated draft (87-221) and his report (87-222).

4. Approval of the Current Draft

Prosser/Jaeschke

"Move we accept 87-221 as the current draft C standard, and 87-219 as the Rationale."

Motion carried, lots/0/0.

5. Multi-byte Support

Plauser presented a proposal (87-199) that we introduce new literals to support wide characters: a character literal of type `wchar_t` and a string literal of type array of `wchar_t`. Several notations were debated for these literals.

Straw vote:

```
1 prefer W"..." (3 would veto)
10 prefer \"..." (0)
5 prefer "\\w..." (4)
9 prefer "...\"L (4)
```

There was further discussion. Weil suggested some form of macro, such as `wc(...)`, to hide implementation details.

Straw vote:

```
3 prefer L"..." or W"..." (0 would veto)
12 prefer \"..." (4)
6 prefer "\\w..." (1)
5 prefer "...\"L (5)
3 prefer wc(...) (1)
```

Straw vote:

```
21 use L"..." or W"..."
7 no
```

Straw vote:

```
12 use L"..."
11 use W"..."
```

Plauser/Elliott

"Move we add L'...' `wchar` literal and L"..." `wchar` string literal."

Motion carried, 25/0/10.

Johnston, abstain -- L clashes with `wchar_t`.

Weil, abstain -- has no better solution, but doesn't like it.

Plauser asked for agreement on the rules for concatenating string literals. All agreed that "..." "..." should be a `char` string, as before, and that L"..." L"..." should be a `wchar` string. He suggested that at least one of the mixed forms be permitted so that `wchar` strings could be constructed using the stringizing capability of macros.

Elliott observed that token pasting could be used to prepend an L to a stringized macro argument, so there was no need for this license.

Straw vote:

```
lots disallow L"..." "..."
```


8 make it L"..."

Straw vote:

lots disallow "... " L"..."

1 make it L"..."

0 make it "..."

Straw vote:

11 need some way to stringize to L"..."

7 not needed

Plauger/Johnson

"Move we say that concatenating strings of mixed type is undefined."

Motion carried, lots/1/5.

Plauger suggested (87-199) that we require comments, string literals, character literals, and format strings to end in an initial shift state. He also proposed changing the specification of wctomb to ensure that converting a zero code left the multi-byte string nul terminated and in an initial shift state. Both of these changes were accepted with no objection.

Plauger proposed altering the names of the two macros giving the worst case and current length of the longest multi-byte sequence.

Straw vote:

9 use MB_LEN_MAX and MB_CUR_MAX

3 use MB_LEN_MAX and mb_max

The former set was accepted with no objection.

Plauger proposed two additional functions (87-199) for converting between multi-byte and wchar strings, mbstowc and wcstomb. He agreed that both should return type size_t, and that the error return value should be given as (size_t)(-1).

Plauger/Gwyn

"Move we adopt mbstowc and wcstomb functions, as described in 87-199 with amendments."

Motion carried, lots/0/4.

Plauger/Elliott

"Move we permit an array of wchar_t to be initialized by a wchar string literal."

Motion carried, lots/0/1.

Straw vote:

2 permit "... " '...' to concatenate to '...'
lots no

Goldberg submitted specifications (87-234) for a set of wchar string functions analogous to the char string functions.

Plauser acknowledged that the specification appeared well thought out, but reiterated his reluctance to require so many additional functions of all implementations when they have not yet been proven by widespread practice.

Straw vote:

2 add some wcs* functions
lots no

Elliott asked that the minutes show he supported the addition of these functions.

Elliott requested that we leave the use of the 'l' (lower case L) qualifier undefined, in printf and scanf, rather than requiring it to be ignored when no semantics are given. Accepted with no objection.

Prosser/MacDonald

"Move we reserve the names wcs[a-z]* for functions in <string.h>, under Future Directions."

Motion carried, 19/5/7.

Elliott presented a request (87-235) that we add the function ismbc.

Prosser observed that the functionality was contained in wctomb.

Straw vote:

2 add ismbc
lots no

Elliott requested that the minutes show he favored adding ismbc.

7. Aliasing

Plauser presented a proposal (87-198) to alter the meaning of pointer to const and pointer to volatile, so that optimizers could be more aggressive in introducing parallelism within C programs. He presented twelve "commandments," ten of which were intended to state access and aliasing rules that the Committee had already agreed to. The eleventh commandment stated that accesses via a pointer to const could be safely moved within a function activation, since no other agency was allowed to modify the object within the same function. The twelfth commandment assured that a pointer to nonvolatile could not be used to access a volatile object, and hence

could be safely optimized more aggressively.

The Committee also considered a proposal (87-243) by Darnell, that we disallow overlap of any function parameters declared as arrays, rather than pointers. There was some concern that this might break considerable working code, but considerable enthusiasm for some variation of the proposal as a way to make parameter accesses more optimizable.

Osten also proposed (87-201) a more simply stated version of Plauger's proposal.

There was considerable discussion.

Prosser objected to a change in the semantics of pointers to const, since this would be a deviation from C++, whose semantics have been exercised for several years.

Straw vote:

- 26 accept Plauger's aliasing rules (2 would veto)
- 16 accept Darnell's parameter rules (3)
- 0 no change (4)

There was more discussion.

Straw vote:

- 28 accept Plauger's change to pointer to const
- 3 no

Prosser reiterated his intention to vote against the draft if this change were adopted.

MacDonald agreed to provide wording for three motions, one to accept commandments 1-10, one for commandment 11, and one for commandment 12.

8. Grouping

Plauger presented a proposal (87-197) that grouping be honored in expressions at all times. Rearrangement of trees with adjacent associative operators would be permitted only by the "as if" rules.

Bradley presented an argument (87-233) against such a change. He expressed concern that in an environment where signed integer arithmetic overflow is trapped, significant optimizations could not be performed under the "as if" rule.

MacDonald offered a compromise (87-211) that required regrouping to preserve precision, but not necessarily avoid introducing overflows. With this compromise, he also suggested we endorse the common practice of rewriting $A+B$ as $A+(-B)$ to increase the opportunity for regrouping.

There was considerable discussion.

Straw vote:

26 honor grouping (2 would veto)
6 no

There was additional discussion.

Straw vote:

28 honor grouping (1 would veto)
5 no (2)

Plauser/MacDonald

"Move we honor grouping in expressions."

Motion carried, 29/5/1.

Bradley, no -- it would force signed arithmetic to suppress overflow or be suboptimal.

Weidenhofer, no -- it overloads parentheses something fierce.

Elliott/Khushf

"Move we eliminate the special grouping properties of unary +."

Motion carried, lots/0/0.

Prosser suggested eliminating unary + altogether.

Plauser defended its presence, as a way of writing data initializers that emphasize which numbers are positive and which are negative.

Straw vote:

5 eliminate unary +
lots no

9. Miscellaneous

Prosser requested (87-143) that we also reserve the macro names SIG_*, as well as SIG[A-Z]*. Accepted with no objection.

He also proposed a clarification of the safe contexts for a call to setjmp. Accepted with no objection.

Meyers presented a request (87-162) that strxfrm have a failure return, in case it cannot transform a string from one locale to another in a form suitable for ordering.

Prosser stated that such an eventuality was discussed at the time strxfrm was designed, but that he still saw no need for a failure code.

Williams requested (87-171) additional translation limits,

giving the minimum/maximum number of struct, union, or enum members, and the minimum/maximum number of structure nesting levels.

Straw vote:

lots require 127 members, 15 levels
4 no

Accepted with no objection.

10. Classification of Macros

Plauser presented a list (87-182, 87-203) of library definitions and the headers that reference them. He asked for guidance in determining which, if any, definitions should occur in multiple headers.

Straw vote:

0 each in one place only
12 NULL in <stddef.h> and <stdio.h>
19 NULL and size_t as needed
3 everything referenced everywhere

Plauser/Prosser

"Move we define NULL and size_t in each header that references them, as in 87-203."

Motion carried, lots/0/2.

Plauser asked for clarification of the nature of several macros. The Committee agreed that CLK_TCK should be an arithmetic rvalue (not necessarily known at compile time).

Plauser suggested that several macros, including assert(), va_start(), and va_end(), be classified as "statement macros," so that they are safely usable only in a context where an expression is permitted.

Prosser showed how assert() can be written as a void expression, and argued that we should require assert() and va_start() to be usable in any context that accepts a void expression. Accepted with no objection.

Plauser/Gwyn

"Move we say that va_end is a void expression macro, not a function."

Motion carried, lots/1/3.

Prosser, no -- don't know why anybody needs macro version only.

11.

Unsigned memcmp

Plum presented a request (87-177) that we add a memucmp function, guaranteeing to perform its comparison on unsigned chars.

Straw vote:

0 add memcmp

lots no

Plum interjected a request for a new liaison to X3H2 (SQL) to replace Bradbury.

** Crockett will serve as X3H2 liaison.

12. Setlocale and is* Functions

Meloy requested (87-188, 87-226) that extra space characters be recognized by isspace in alternate locales.

Prosser objected that this would permit only 1 byte alternate space characters, since isspace has a restricted domain.

There was some discussion.

Straw vote:

21 make isspace locale dependent

3 no

Meloy/Jervis

"Move we make isspace locale dependent, as in 87-226, p. 3."

Motion carried, lots/1/8.

Meloy also requested that isdigit and isxdigit also be locale dependent. Various people objected, since the value of each digit must also be known in many cases.

Straw vote:

4 make isdigit and isxdigit locale dependent

14 no

Meloy asked that the minutes show she voted yes.

Meloy then requested that the strt* functions be implemented for unexpected input, so that they can be extended to accept alternate digits. After some discussion, she accepted the limitation that extra characters can be recognized only outside the C locale.

Straw vote:

lots allow extra strt* characters outside C locale

1 no

Accepted with no objection.

13. <stdarg.h> Semantics

Several issues were raised (87-181) about the semantics of the `va_*` macros. Can you use multiple `va_start/va_end` pairs in a single function. The consensus was that you could, provided that each `va_start` on a `va_list` be balanced by a `va_end` call. Can you nest calls (using different `va_list`'s). Yes. Can the macros occur at different block levels? Yes, because they are void function macros, not statement macros.

Gwyn agreed to provide Rationale words to clarify these semantics.

14. Extended Discussion Time

Prosser suggested that we require `'0'` through `'9'` to be contiguous values, in both source and execution character sets. It was pointed out that other languages impose similar limitations, and all known character sets honor this restriction.

Straw vote:

lots require `'0'` through `'9'` be contiguous
1 no

Gwyn/Prosser

"Move we require `'0'` through `'9'` be contiguous values in source and execution character sets."

Motion carried, lots/1/1.

Peyton, no -- concerned that we're restricting representations of character sets.

Brodie asked for a review of outstanding no votes on the draft:

Elliott -- international leftovers, `const` aliasing
Prosser -- `const` aliasing
Meloy -- `localeconv`
Jackson -- "same type" terminology
Bradley -- grouping
Gwyn -- POSIX conflicts

15. Ambiguous Initializers

Prosser reported (87-222 #4) an ambiguity discovered by Brader in the interpretation of braces in data initializers. He cited the example

```
int a[2][2] = {{1}}, {2}};
```


Is {1} an initializer for a[0][0], with redundant braces around a scalar, or is it an initializer for a[0], with a[0][1] unspecified? He proposed that we disallow redundant braces in scalars within a complex initializer.

Plauser objected to disallowing a widespread practice. He also repeated his plea (87-200 #4) that we make anything except full bracketing or no bracketing undefined behavior.

Khushf observed that that such a response would not eliminate the ambiguity.

Straw vote:

- 21 status quo
- 2 make embedded redundant {} undefined
- 10 make partial bracketing undefined

It was agreed that the ambiguity should always be resolved by assuming the braces are not redundant. Hence, {1} in the example above initializes a[0].

** Gwyn agreed to discuss the {} initializer ambiguity with Brader.

16. Whitespace in *scanf

Jervis presented a request (87-175) that we strike the requirement that the scanf whitespace directive fails if there are no whitespace characters in the input, since this is not current practice. Accepted with no objection.

17. *printf("%-03f", s)

MacDonald (87-154 #2) and Prosser (87-222 #2.12) pointed out two anomalous properties of the current definition of zero padding with *printf. Right padding can cause 5 to be printed as 500, and left padding can cause -5 to be printed as 0-5. It was agreed that the zero fill should not occur in these cases. Accepted with no objection.

18. Pointer Normalization

Prosser observed (87-222 #2.2) that we require equal pointers of compatible type to point to the same thing, but do not say that two pointers to the same thing must compare equal.

Jervis asked also for a footnote emphasizing that this requirement might not be met after undefined operations have been performed on either of the pointers.

Both changes accepted with no objection.

Meloy (87-229) reported that X/OPEN objects to localeconv, because it has a style inconsistent with strftime, returns a pointer to a struct which is hard to extend, and is affected by two different locale categories. She requested that

localeconv either be replaced with a proposed function strfmon, or be dropped.

Straw vote:

0 add strfmon

lots no

Straw vote:

11 drop localeconv (1 will veto)

6 no (0)

Meloy/Prosser

"Move we drop localeconv."

Motion defeated, 11/10/15.

Elliott (87-236) requested that we add more information on how to format thousands separators. He demonstrated the usual mind boggling diversity of notation across different cultures.

Plauser suggested that he reverse the role of the codes 0 and MAX_CHAR, so that the strings would not contain embedded nulls.

The modified proposal was accepted with no objection.

Elliott further suggested that there be a positive_sign string, much like the negative_sign. For the C locale, this would be "".

Straw vote:

14 add positive_sign

2 no

Jackson entered a plea that the term "same type" be changed to something less misleading. Several terms were discussed, the least objectionable being "compatible type."

Straw vote:

lots change "same type"

0 no

The change to "compatible type" was accepted with no objection.

Gwyn (87-210) requested that the semantics of signal handlers be altered to embrace newer systems that have closed the window of vulnerability inherent in traditional UNIX signals. Accepted with no objection.

19. ParmN of Default Type

Prosser (87-222 #2.6) requested that we require the last parameter before the variable argument list to have a type

that does not widen. Accepted with no objection.

20. stderr not Buffered

Prosser (87-222 #2.10, 87-203 #10) requested that we say that only stdin and stdout are buffered, not stderr. Accepted with no objection.

21. Extended Discussion Time

Prosser (87-222 #2.1) reported a difference of opinion on how to handle unbalanced ' and ' as preprocessor tokens. Status quo is to treat them as one character tokens, but some people have objected to having to backtrack from the end of a line when an unbalanced quote is detected.

Straw vote:
17 make unbalanced quotes undefined (Ø will veto)

8 leave as one character tokens (Ø)

Prosser/Gwyn

"Move we say that unbalanced single or double quotes are undefined behavior."

Motion carried, lots/5/2.

22. Resolution of Deferred Items

There were none at this time.

23. struct and Type Equivalence

The request (87-139) was handled earlier.

24. Initialization of a Formal

The request (87-150) was handled earlier.

25. Plauser Edits

Prosser reported that the edits Plauser had requested (87-153 #1) had either been accepted or resolved to Plauser's satisfaction. Plauser agreed.

26. X/OPEN Proposals

No issues from X/OPEN (87-164) stimulated debate.

27. printf Cleanups

Jervis (87-172) asked that the %% printf specifier not be obliged to handle zero fill, etc., as in %05%. He requested that any flags, width, or precision have undefined behavior.

Straw vote:

23 say %5%, etc. is undefined
6 no

Accepted with no objection.

Prosser presented a request by Colligan (87-183) that we clarify the effect of %.0g. Prosser reported that UNIX treats this the same as %.lg.

Straw vote:

12 make %.0g behave like %.lg
4 no

Prosser/Weil

"Move we treat %.0g the same as %.lg in *printf."

Motion carried, 16/2/11.

Plauser (87-191, 87-200 #10) requested that scanf define %x input in terms of strtoul, and a store in an unsigned int *, so that it can accept all values put out by printf with %x.

Prosser observed that strtoul doesn't accept an optional sign.

Straw vote:

22 allow optional sign in strtoul input
4 no

Accepted with no objection.

Straw vote:

20 change fscanf %o and %x to store unsigned int
1 no

Accepted with no objection.

28. \ Newline Limitations

Jervis (87-172 #3) requested that we say line continuation (via a backslash at end of line) need not be supported outside character and string literals.

Straw vote:

4 make \ continuation undefined outside quotes
22 no

29. Interactive Device

Jervis (87-174) suggested wording to clarify that what we mean by "interactive device" is implementation defined. Accepted with no objection.

30. scanf/ungetc Rationale

Jervis (87-176) suggested Rationale wording to clarify that ungetc must support one character of pushback, even in the presence of scanf calls. Accepted with no objection.

31. Limit on Conditional Nesting

The Committee considered a request by Bradbury (87-178) to change the limit on #if conditional nesting to 8 from 6. Accepted with no objection.

32. Boundary Case for memcpy

The Committee considered a request by Bradbury (87-179) that we explicitly disallow calling memcpy with null pointers, even if the move count is zero.

Gwyn saw no reason to single out this particular function. Prosser asserted that 4.1.6 contains a blanket prohibition against null pointer arguments to library functions, unless specific semantics is assigned to the null value. Meissner reported that his implementation checks that some pointers are valid even before they are dereferenced, so this prohibition is important.

Straw vote:

.1 permit null pointer to 0-sized object
lots no

33. size_t Rationale

The Committee accepted Rationale wording (87-180) from Bradbury on size_t, with no objection.

34. Why is errno Volatile?

Meissner asked (87-184) why we require that errno be volatile, since that clashes with old style declarations of the form "extern int errno;".

Prosser explained that it was made volatile because signal handlers may set it as a side effect of calling signal (with bad arguments).

There was some discussion, yet again, of what was wrong with errno and what could be done to fix it. There was no resolution, yet again.

Straw vote:

21 need to fix errno
0 no

35. Initialization Clarifications

Osten (87-190) requested that the appendix on unspecified

behavior also include the alignment of bitfields. Accepted with no objection.

36. Size of Character Escapes

Plauser (87-200 #1, 87-153) requested that we clarify that the value of a character escape sequence must fit in an unsigned char, to bless the common practice of writing \377 for -1. Accepted with no objection.

37. Plauser (87-200 #2) requested that we add a MAX_BOUND macro, to ease writing portable malloc and calloc. After some discussion, he withdrew his request. Plum requested a straw vote anyway.

Straw vote:

10 add MAX_BOUND
22 no

38. Plauser (87-200 #3) withdrew his request that useless expressions (such as x;) be branded as undefined behavior.

39. Plauser (87-200 #5) requested that redeclaration of entities with no linkage be a constraint violation at file level as well as at block level. Prosser reported that he had already made the change editorially. Accepted with no objection.

40. Plauser (87-200 #6) requested that we state that void * have the same representation as char *. Accepted with no objection.

41. Plauser (87-200 #7) requested that we clarify that behavior explicitly labelled as undefined have no different status than violations of a "shall" or "shall not," or behavior that is not spelled out. Accepted with no objection.

42. Plauser (87-200 #8) requested that we specify for bsearch which of the arguments to the user supplied function points at the key and which at the table entry. Gwyn agreed to determine which is current practice in UNIX. Accepted with no objection.

43. Plauser (87-200 #9) requested that we change the specification of memmove to say the source argument is not a pointer to const.

Prosser objected, on the same grounds that he opposed the change in the meaning of const *.

Straw vote:

21 remove const from 2nd argument to memmove
4 no

The Committee agreed to bundle this issue with the discussion of explicit motions on the change to const *.

44. Enumeration Issue

Osten (87-209) requested that we change the grammatical definition of "enumerator" from "enumerator-constant" to "identifier."

Straw vote:

4 change grammatical definition of enumerator
lots no

Osten also pointed out lapses in the application of the constraint that certain values (enumeration constant values, array sizes, bitfield sizes, and case label values) be "integer constant expressions." His wording changes were accepted with no objection.

Osten proposed that the offsetof macro not be required to be computable at compile time.

Straw vote:

1 offsetof need not be integer constant expression
lots status quo

45. File Rewind Issues

MacDonald requested (87-213) that we relax the requirement that a file be rewound on open. He suggested the FORTRAN rule, that the file position be implementation defined on open.

Straw vote:

1 say file position is implementation defined on open
lots no

46. Implied int in Types

The Committee considered a request by Satchell (87-215) that we make obsolescent the omission of all type keywords in a declaration. There was some sympathy, but little support for the change.

Straw vote:

2 make implied int obsolescent
lots no

47. Requested Clarification

The Committee considered a request by Pugh (87-216) that we write const type qualifiers to the right of other type keywords in all examples in the draft.

Straw vote:

6 write const to the right in examples
8 no

Pugh also requested that we specify a limit on the required

parentheses nesting in declarations, as well as expressions.

Straw vote:

16 limit parens nesting on declarations
3 no

Straw vote:

lots set declaration parens nesting limit to 31
3 no

MacDonald/Jaeschke

"Move we add a translation limit for the maximum parentheses nesting in declarations, and set it to 31."

Motion carried, lots/3/10.

Bixler, no -- we're overgeneralizing.

48. Difference of Pointers

The Committee considered a request by Brader (87-217) that we define the difference between two pointers into the same object even after the object has been deallocated.

Bradley showed how this can fail on an implementation that uses a descriptor portion of a pointer to determine (sub)object size information. Once an object has been deallocated, such pointers can be used for very few operations.

Straw vote:

lots can't assign, compare, or add to dead pointers
2 no

The consensus was that the draft should forbid the use of an object pointer value in an expression once the object has been deallocated. Accepted with no objection.

49. \$ in Identifiers

Peyton (87-220) requested that we permit \$ in identifiers, and make all such identifiers reserved to the implementor.

Straw vote:

5 permit \$ in reserved identifiers
lots no

50. Deferred Issues

Prosser (87-222 #3) requested that we move most restrictions on constant expressions from constraint violations to undefined behavior. He would leave as constraint errors the use of assigning operators and the comma operator, and computing a value too big for the target. Accepted with no objection.

He also suggested eliminating an oversight that currently permits objects (but not functions) to be retyped when redeclared. Accepted with no objection.

Prosser also suggested clarifying the nature of return values for setvbuf. Accepted with no objection.

Crockett (87-227) requested that the file position indicator be left undefined by fflush on a stream in input mode. This prompted some discussion of the use of fflush and fseek to switch modes, and their interaction with ungetc. The mention of ungetc stimulated the appearance of Farance (like the ghost of Christmas Past), who smiled benignly over the ensuing debate.

Straw vote:

```
11 fflush may discard input
9 no
```

Prosser then requested, as a fallback, that it be undefined whether fflush/fseek, etc. restore the characters masked by ungetc, at least for successful calls to those functions. Accepted with no objection.

That prompted further discussion of the use of fflush on input streams (streams opened for input or opened for update and currently in input mode).

Straw vote:

```
21 fflush on input mode stream is undefined
2 no
```

Accepted with no objection.

Plauger presented a request by Tøndering (87-223, 87-224, 87-248) that we add digraphs for braces and brackets, and other readability aids to avoid most uses for trigraphs.

Straw vote:

```
4 need to solve trigraph readability problem
lots no
```

Plauger presented a request from WG14 (87-248) that we change "Greenwich Mean Time (GMT)" to "Universal Time Coordinated (UTC)." Accepted with no objection.

Plauger presented a request from WG14 (87-248) that we clarify where preprocessor tokens become C tokens. Already done.

Plauger presented a request from WG14 (87-248) that we clarify the effect of

```
#define a defined
#define b c
```


#if a b

It was agreed that this is undefined behavior.

Plauser presented a request from WG14 (87-248) that we clarify whether multiple token pasting, as in a ## b ## c, must result in valid preprocessing tokens regardless of the choice of intermediate states. It was agreed that all possible intermediate states must be valid.

Plauser presented a request from WG14 (87-248) that we define a text file such that a line can end (in the file) with a form feed, carriage return, or vertical tab, as well as a new line. Some concern was expressed that this might be too rash.

Straw vote:

16 don't require FF and VT be transparent
7 no

Plauser/Bixler

"Move we relax the requirement on text files that form feed and vertical tab be transparently output and input."

Motion carried, 19/7/4.

Plauser presented a request from WG14 (87-248) that we make it implementation defined whether a text file is truncated beyond the end of its last write. Accepted with no objection.

Plauser presented a request from WG14 (87-248) that we refine the type matching requirements for the relational and equality operators. Already done.

Plauser presented a request from AFNOR (87-225 #3) that we permit p.x, where p->x is valid.

Straw vote:

5 permit p.x as well as p->x
30 no

The Committee considered a request from Pugh (87-231) that we not say old style function definitions are obsolescent. There was no support for the change.

The Committee reviewed a submission from Bowman (87-237). Rozakis reported that he found no issues in the submission that demanded Committee time.

MacDonald (87-241) requested that we reserve strt* names to implementors in <stdlib.h>. Prosser observed that we should also explicitly reserve names in the external name space when we reserve function names. Better wording to be provided later in the meeting.

51. Darnell Proposal Revisited

The Committee resumed discussion of Darnell's proposal (87-243) that we give special meaning to function parameters of array type. A declaration of the form

```
int vadd(int va[], int vb[], int vc[]);
```

promises that va, vb, and vc do not overlap and hence may be vectorized.

There was considerable discussion of alternate notations, since some members expressed concern that existing programs would break if all array type parameters had special meaning. One proposal was to restrict the special meaning to array types declared within prototypes, since this is newer notation.

Straw vote:

- 17 restrict va[] meaning to prototypes
- 18 use new syntax
- lots have some facility a la Darnell's proposal

There was more discussion.

Straw vote:

- 2 no new syntax for Darnell's proposal
- lots some new syntax

A final resolution was deferred until later in the meeting.

52. Subgroups

The Committee broke into subgroups to preprocess the remaining technical contributions.

53. Subgroup Presentations

Johnson presented several editorial changes by Korpela (87-246). He suggested removing the definition of function, adding rationale on unspecified vs. undefined, and clarifying the meaning of a "stable" volatile object. Accepted with no objection.

Weil presented an assortment of edits from Olson (87-232) which were accepted with no objection.

MacDonald (87-242) requested that we introduce a type definition "extended" and use it in place of "double" for <math.h> arguments and function return types. Lewis, from Apple, was present to second these changes, as a way of improving support for IEEE floating point.

There was much discussion.

Straw vote:

5 use extended instead of double in <math.h>
19 no

MacDonald requested that we define the return value of atan over the closed interval $[-\pi/2, +\pi/2]$ instead of the open interval. Accepted with no objection.

MacDonald requested that we define the return value of atan2 over the closed interval $[-\pi, +\pi]$ instead of the half open interval. After some discussion, accepted with no objection.

MacDonald requested that we say atan2 report a domain error only if $\text{atan2}(0, 0)$ is not representable, that log and log10 report a range error only if $\log(0)$ is not representable, and that pow report a domain error only if $\text{pow}(0, \leq 0)$ is not representable.

There was much discussion of whether these changes would make it impossible to test for library errors in a portable fashion.

Straw vote:

15 okay if errno not always set by <math.h>
6 no

Final resolution was deferred until later in the meeting.

54. Aliasing

Jervis reported on a compromise position regarding the aliasing of const pointers. The compromise was subscribed to by Prosser and Elliott, who were otherwise prepared to vote against the draft with the changes agreed to earlier, and by others with a stake in enhancing the prospects of better aliasing analysis of C programs.

Jervis proposed adding a third type qualifier, a la const and volatile, that would guarantee that an lvalue would have no other aliases in scope. The qualifiers const and volatile would retain their old meaning, in the absence of this new qualifier, and would take on the new meanings in its presence.

Jervis presented a table of all combinations of type qualifiers:

---	ordinary (optimizable) lvalue
C--	no writes permitted via this lvalue
-V-	other agencies can alter this object
--N	only this lvalue reads or writes the object
C-N	only this lvalue reads object (no writes)
CV-	others can alter, but this lvalue only reads
-VN	others can alter, but not other lvalues
CVN	others can alter, but only this lvalue reads

Plauger's commandment #11 applies only to C-N and CVN.
 Commandment #12 applies only to -VN and CVN.

There was further explanation, and considerable discussion, of this proposal. Several members expressed nervousness at making such a change, including adding a keyword, at this late date. There was strong sentiment, however, that the semantics had been well explored.

Weidenhofer asked that the two week rule be invoked to end discussion on this topic. Brodie, after some thought, ruled that there was ample justification for the discussion, given the papers prepared on the subject.

Straw vote:
 36 add noalias type qualifier
 0 no

The actual spelling of the new keyword was deferred until later in the meeting.

55. Monday Papers

Jervis presented a request from Clunie (87-230) that we list repeating a macro parameter name, as in

```
#define x(a, a) 2 * a
```

as a constraint error. Accepted with no objection.

Jones presented a request from Korpela (87-246) that we clarify the reservation of library external names. Accepted with no objection.

Gwyn presented a request from Brader (87-238) that we not require that an array printed with %.5s be null terminated (beyond the last character printed). Accepted with no objection.

Other clarifications also accepted were to say that all null pointers compare equal, to fix the description of fopen append mode, and to add a footnote emphasizing that multiple calls to the same function may occur at exit.

Gwyn presented several areas of conflict between POSIX and the draft. OPEN_MAX and PATH_MAX have been changed recently by IEEE 1003 to have different meaning than in the draft. There are also a number of additional macros defined by POSIX in <limits.h> that are not permitted by the draft.

There was considerable discussion of how to resolve the differences.

Straw vote:
 10 get POSIX to change
 10 change the draft to match POSIX

Plauger/Farance

"Move we tell IEEE 1003 that their Draft 12 is unacceptable because of conflicts in <limits.h>."

Motion carried, 20/1/8.

Meissner, abstain -- a shouting match will only do damage.

** Gwyn will draft a letter to POSIX stating objections.

Gwyn/Plum

"Move we rename <limits.h> as <stdlim.h>, and redefine OPEN_MAX (min/max 8), and PATH_MAX (min/max 255) to match POSIX standard, if IEEE 1003 does not conform by 31 Dec 87.

Motion carried, 16/3/5. Gwyn asked if we need to define the evaluation order for integer constant expressions, to clarify the effect of dividing by zero, for instance. In particular, he asked if (1 || 2/0) were short circuit evaluated at compile time. The consensus was that it is. Hence the expression is well formed.

Osten (87-239, 87-240) proposed a new grammar for declarations to minimize the number of reduce/reduce conflicts in the current grammatical description.

Straw vote:

4 adopt new grammar for declarations
17 no

Bradley (87-248) asked that we make it implementation defined which binary files truncate on write and which do. The consensus was that, so long as an implementation provides at least one form of binary file that does not truncate on writes, it is conforming. All other forms of file I/O may be viewed as useful, if nonconforming, extensions. Bradley accepted this interpretation.

Plum reported an observation by Tolland (87-251) that our current rule for when g format switches from fixed to exponential format is off by one from current practice. A change to match current practice was accepted with no objection.

MacDonald requested on behalf of Bray (87-244) that we add the keyword unshared, as a type qualifier, to the list of common extensions. After some discussion, the request was withdrawn.

MacDonald revisited <math.h> errors. He proposed that we permit a function to return an encoding for plus or minus infinity in lieu of setting errno, with the understanding that HUGE_VAL would then be plus infinity. Otherwise, if a

function returns a NAN (not a number), then errno is assuredly set to EDQM.

With this understanding, his requested changes to atan2, log, log10, and pow were accepted with no objection.

MacDonald requested that we drop the requirements that +E be the same as 0+E, and -E be the same as 0-E, since they are not in IEEE floating point [sigh]. Accepted with no objection.

MacDonald presented a request from Hough (87-250) that we clarify that ceil produces a correct result whether or not its value is representable as an integer type. Plauger objected, saying the description of <math.h> already makes this clear. No change was accepted.

Prosser presented a request (87-250 #12) that we require translators not to fold (1.0/3.0) at translation time, if the rounding mode can change at run time. Plauger insisted that this is already covered by the "as if" rule.

Prosser presented a request (87-250 #9) that we add a code to FLT_ROUNDS for rounding the halfway case away from zero. Not accepted.

Prosser presented a request (87-250 #17) that we say fmod(x, 0) either return 0 or report a domain error. Accepted with no objection.

56. Wording Changes

Brodie asked for volunteers for a draft review committee, to meet at AT&T Summit NJ on 7 Jan 88. Prosser will serve as master of ceremonies.

** Hudson, Johnson, Crockett, MacDonald, Weidenhofer, Gwyn, Elliott, and Plauger will participate in the draft review.

A number of proposed wording changes were distributed for review by the Committee.

Several changes to translation limits were accepted with no objection.

Wording changes describing file truncation were accepted with no objection.

Reservation of str[a-z]* names in <stdlib.h> for implementors caused some discussion. Deferred until later in the meeting.

Gwyn proposed that we honor an IEEE 1003 request to permit argc to be zero on calls to main. Accepted with no objection.

Changes to ungetc, fopen, fseek, and fflush, agreed to

earlier, were accepted with no objection.

Changes to clarify how the key argument is used by bsearch were accepted with no objection.

Changes in the description of constant expressions were accepted with no objection.

Additions to localeconv were accepted with no objection.

Other wording changes agreed to earlier in the meeting were reviewed and accepted with no objection.

Prosser outlined the changes needed to accompany the new type qualifier.

He defined a "handle" as that part of an lvalue that designates the address of an object.

An implementation is allowed to behave as if there is a separate copy of each object associated with each handle of an lvalue with noalias-qualified type.

Calling a function that has a parameter of type pointer to noalias-qualified type effectively passes along rights to that handle's copy.

Returning from such a function effectively copies back the object pointed at by the pointer parameter.

There was more elucidation and discussion.

Straw vote:
lots adopt noalias type qualifier
0 no

Various candidate spellings were proposed for the new keyword.

Straw vote:
16 noalias
10 unaliased
3 safe
1 unique
2 unshared

Straw vote:
23 noalias
14 unaliased

Jervis/Meissner

"Move we adopt this stuff."

Motion carried, lots/0/1.

Weidenhofer, abstain -- makes me nervous.

Jackson, yes but -- if we muddy it up, will vote no on draft.

Elliott (87-119) proposed that we permit freopen with a null filename argument to reopen the specified stream with a change of mode. There was some discussion as to whether this was always possible.

Straw vote:

14 permit mode change on freopen(NULL, ...)
9 no

No motion made.

57. Summary of Decisions Reached

Brodie despaired of summarizing all of the decisions agreed to during the week, so he did not ask Plauger to give his usual recital.

58. Vote on Submission for 2nd Public Review

Prosser/Elliott

"Move we send out the amended draft for second public review."

A roll call vote was requested:

Prosser, yes
Baumbach, yes
Weidenhofer, no -- no longer C with grouping,
noalias makes me nervous

Peyton, yes
Jervis, yes
Jaeschke, yes
Mickey, yes
Davies, yes
Fosbury, yes
MacDonald, yes
Meyers, yes
Meissner, yes
Andrews, yes
Farance, yes
Jackson, yes
Jeter, yes
Meloy, yes
Osten, yes
Elliott, yes
Hudson, yes
Schrecker, yes
Brodie, yes
Kotikian, yes
Rasbold, yes
Prodehl, yes

Jenkins, yes
 Weil, yes
 Paton, yes
 Schubert, yes
 Williams, yes
 Plum, yes
 Johnson, yes
 Bradley, no -- unhappy with grouping of signed arithmetic
 Jones, yes
 Rosenthal, yes
 Richardson, yes
 Sutton, yes
 Khushf, yes
 Bixler, yes
 Gwyn, yes
 Rozakis, yes
 Plaugar, yes

Motion carried, 39/2/1.

Brodie asked Weidenhofer and Bradley if they felt that they had sufficient discussion time on the issues to which they had objected. Both conceded that no additional discussion time would have helped.

Prosser asked that the review committee be given reasonable latitude in responding to a partial response from POSIX. There was no objection.

59. Action Items

Plaugar reviewed the action items agreed to during the week. These are marked in these minutes with a ** in the left margin.

60. Future X3J11 Meeting Schedule

The schedule for future meetings is now:

18-22 Apr 88	Nashua NH	Honeywell
15-19 Aug 88	Cupertino CA	Borland/Tandem/H-P
12-16 Dec 88	Seattle WA	Microsoft
10-11 Apr 89	Phoenix AZ	Brodie

61. Other Business

** Weidenhofer will do the mailing for the next meeting.

Plum set 25 Mar 88 as the cutoff date for the last mailing. Material must reach him by that date to be included.

Prosser agreed to produce diff marks vs. the current draft, for the next draft.

Brodie agreed to review X3 acceptance and maintenance procedures and report on them at the next meeting.

The Committee thanked Tymlabs for hosting the meeting.

62. Adjournment

Hudson/

"Move we adjourn."

Motion carried, lots/0.

The meeting adjourned at 11:55 a.m. on Friday, 11 Dec 87.

Attachments:

Attendance Sheets
Revised Agenda

[Not circulated to IST/5/14]
[simply to save paper.]