ISO/IEC JTC1/SC22
Languages
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC22
N 1253

SEPTEMBER 1992

TITLE: Minutes of meeting of WG14 - C held in Salt Lake City USA, 1992-05-13/15

SOURCE: Secretariat ISO/IEC JTC1/SC22

WORK ITEM: JTC1.22.20.02

STATUS: New

CROSS REFERENCE: N/A

DOCUMENT TYPE: Minutes of WG meeting

ACTION: For information to SC22 Member Bodies.

Address reply to: ISO/IEC JTC1/SC22 Secretariat
J.L. Côté
Treasury Board Secretariat
140 O'Connor St., 10th Floor, Ottawa, Ontario, Canada, K1A 0R5
Tel.: (613)957-2496 Telex: 053-3336 Fax: (613)996-2690

MINUTES OF
13-15 MAY 1992
MEETING OF SC22/WG14
PROGRAMMING LANGUAGE C

X3J11/92-025
WG14/N224

15 May 92
Project 381-D

MINUTES OF 13-15 MAY 92 MEETING OF ISO/JTC1/SC22/WG14 AND ANSI X3J11

13 May 92 09:00-12:00 13:30-17:00
14 May 92 08:30-12:00 13:30-17:30
15 May 92 08:00-12:00 13:30-15:30

Radisson Hotel
Salt Lake City Airport
2177 West North Temple
Salt Lake City, UT 84116
(801) 364-5800

1. Opening Activities

1.1 (X3J11) Opening Comments

The 13-15 May 92 meeting of WG14/X3J11 was held in Salt Lake City UT. Host for the meeting was DECUS. Jim Brodie convened the X3J11 meeting at 09:00 and served as chair. P.J. Plauger and Linda Stanberry served as secretaries.

Brodie observed that the Milan meeting of X3J11 adjourned for want of a quorum. He reported that the interpretations made to date have been approved for publication as a Technical Information Bulletin (Project 878-D) by a vote of 35/0/1/1. He also reported that ISO 9899 has been approved for forwarding to ANSI BSR as a replacement C standard. The letter ballot approving an I Project passed 26/2 – final approval is still in the works.

Brodie has sent two annual reports to Sparc since the last meeting. A major open issue is how to handle interpretations once the I Project is in place. We can:

- pass them to WG14 without comment
- pass them with a suggested interpretation
- supply the official US interpretation

There was some discussion of the alternatives. It was observed that SC22 has formed an ad hoc group on interpretations, headed by John Hill.

Brodie decreed that missing the Milan meeting will not affect voting eligibility. He reported that new SD2 procedures are now in place. They evidently require that interpretations be henceforth approved by letter ballot.

Brodie stated that the goals of the joint meeting were to review technical issues related to the drafts of the three components of the Normative Addendum mandated by SC22 and to review interpretations forwarded from CBEMA.

1.2 (X3J11) Host for the Meeting

Brodie announced that the host for this meeting was DECUS. All requests for copying should be directed to Terrazas.

1.3 (X3J11) Approval of Previous Minutes

Plauger submitted 92-004, minutes of the abbreviated Milan meeting. There were no corrections.

Brodie submitted 91-032, minutes of the Norwood MA meeting. Gwyn observed that his name is not Gwynn. Prosser requested that his paper 91-041 be accepted as corrections to these minutes. The Norwood minutes were accepted as amended.

1.1 (WG14) Opening Comments

Plauger convened the WG14 meeting at 09:30. He suggested that the two committes meet as committee of the whole with Brodie serving as chair. A separate WG14 meeting would be held later. Agreed.

1.2 (WG14) Selection of Meeting Chair

Plauger suggested that the WG14 chair selection be deferred until the separate meeting. Agreed.

1.3 (WG14) Approval of Previous Minutes

Plauger submitted N201, minutes of the Milan meeting. Myers observed that Prosser made a misstatement about standard-conforming programs on p. 5. (Both agreed that the minutes were correct, however.) Simonsen observed that on p. 8 the minutes stated that Denmark agreed to drop the NOT macros from their proposal. He emphasized that he was not speaking for his delegation, merely expressing his willingness to try to convince them. With these clarifications, the minutes were accepted.

1.4 Review of Action Items and Resolutions

Simonsen will serve as liaison to WG20. Done.

Plum will respond to FIMS C binding. Done.

» Plum and Plauger will respond to SC2/WG2 (ISO 10646). Not yet.

Plum will describe issues with extra characters in identifiers. Done.

Molenda and Simonsen will specify ways to permit extra characters in identifiers. Done.

Simonsen will provide an amended Danish proposal. Done.

Molenda will propose ways to specify shift states to multibyte functions. Done.

Prosser will propose a specification for a restartable mbstowcs. Done.

Plum will respond to AFNOR on their request for adding extern "C" to C. Done.

Keizer will send ISO/IEC directives to all members. Done.

Plauger will provide mailing labels to Keizer. Done.

Jaeschke will supply information on this meeting. Done.

Plauger will coordinate agendas for this meeting with Brodie. Done.

D. Jones (hereinafter Jones the Lesser) will arrange a tentative Aug 92 London meeting. Done.

1.5 Approval of Agenda (WG14/N202)

Brodie submitted N202 as a proposed agenda for the meeting. With several changes and amplifications, the agenda was approved. The revised agenda, N202R, is attached to these minutes.

1.6 Introduction of Participants

All attendees introduced themselves. Plum reviewed voting status. An attendance sheet is attached to these minutes, showing who has voting rights.

1.7 Information on Next Meeting

Jaeschke will host the next meeting in Herndon VA, near Dulles Airport. It will be a combined X3J11.1/WG14/X3J11 meeting over the week of 7-11 Dec 92.

1.8 Procedures for This Meeting

Brodie suggested that the two committees operate in parallel as much as possible, except when formal votes must be taken.

1.9 Distribution of New Documents

Several new documents were introduced. All accompany these minutes in the mailing.

There was some discussion of future distribution of committee documents via e-mail. Plum offered to provide distribution either via e-mail or fax to all members.

» Simonsen and Plum will check availability of WG14 reflector for X3J11 correspondence.

Gwyn agreed to get new e-mail forms for the X3J11 reflector. (x3j11@nisc.sri.com is the actual group, x3j11-request@nisc.sri.com to add or delete.)

A belated review of action items for X3J11 showed two outstanding items.

» Still need a liaison for X3H5, X3T2.

» Still need support for handling requests for extensions.

1.9 Roll Call

For X3J11, Plum determined that 18 members of a possible 31 were unequivocably eligible to vote, with two more subject to committee approval. This constituted a quorum.

Plum/Terrazas

"Move we restore voting status for Amdahl . Intel."

Motion carried 18/0/0/12/30.

[Key to the new scriptures: That's 18 for, 0 opposed, 0 explicitly abstaining, 12 not present, of a total 30 eligible to vote.]

There were thus 20 present who could vote, out of a total eligible of 32. An attendance sheet is attached showing voting status.

For WG14, four nations were represented: UK (Jones the Lesser), Japan (Noda), USA (Jaeschke), and Denmark (Simonsen).

2. Reports on Liaison Activities

2.1 X3J11 Report

Jaeschke reported that no X3J11 meetings have occurred since the last WG14 meeting in Milan.

2.2 X3J11.1 (NCEG)

Jaeschke reported that the Numerical C Extensions Group, a.k.a. X3J11.1, met for four days in Jan and the immediately preceding two days in May. He noted that three documents are being distributed for preliminary review — on compound literals (N225), restricted pointers (N226), and floating-point (N227). Comments are solicited from WG14 and X3J11. The revised completion date for the X3J11.1 Technical Report is Dec 94.

2.3 X3J16/WG21

Simonsen reported that the latest working draft of the C++ standard includes the digraphs requested by Denmark, but not the operator <> which is subject to some contention with X3J11.1.

Plum reported that X3 has decided that X3J11 is *not* a coordinating liaison to X3J16. Anyone who wants to have an influence on the C++ standard should

join that committee. Plum also reported that the X3J16 I project has been approved. All meetings are held jointly with WG21 and all documents are distributed via e-mail.

There was some discussion of the digraph proposal adopted by X3J16. Jones felt that C++ now has digraphs because that committee thinks they're going into C. Jaeschke observed that macro names in the C proposal are now keywords in C++. Plum emphasized that the C++ committee strongly supports digraphs.

Simonsen observed that the C++ committee is discussing ways to register libraries, to control the external name space better. He felt that the C community should join this effort. Plauger suggested that such an activity is still premature.

2.4 WG20

Simonsen reported that WG20 (internationalization) met in April in Paris (sigh). They have proposed an NP for a locale registry similar to that for POSIX. Completion is expected by mid '94. Plum noted that his proposal (WG14/N207) was a potential candidate.

WG20 also has an NP on ISO/TR10176, "Guidelineson the Design of Programming Languages," which includes a multibyte form for ISO10646. Yet another NP is "Requirements and Reference Model on Internationalization." All have been submitted to the Aug '92 plenary of SC22.

Simonsen drew attention to two papers on extended identifiers, N203 and N214, a topic of continuing interest in internationalization.

2.5 WG15

Simonsen reported that the POSIX committee meet the previous week in New Zealand. Ballots have been approved on "Shell and Utilities," 1003.2 and 1003.2a. Both are now forwarded for DIS balloting, with IS balloting expected by early '93. That group discussed the use of symbolic character names in locales. They are also considering using the alpha class to identify characters valid for identifiers.

2.6 Other Liaison Activities

Plauger summarized issues with the draft standard 10646 (SC2/WG2). He observed that various groups object to parts of the standard, but that it does meet the constraints imposed in the C Standard. There was some discussion. Brodie observed that the US TAG must reply to JTC1 by 19 May.

Brodie reported several issues from SPARC. Should we have a liaison for X3H5, developing a parallel-processing model for HLLs? Jaeschke observed that Curley at Convex was the informal liaison with X3J11.1. The latter group is trying to avoid overlap of effort.

"Move we form a liaison with X3H5."

Motion carried 7/5/8/12/32

» Farance will serve as liaison to X3H5.

Brodie noted several other requests for liaison with IEEE POSIX and directory services applications interface. No support.

Jaeschke reported that liaison with FIPS is now active, but there are no current issues.

L. Jones (hereinafter Jones the Greater) reported that there are several graphics bindings out for review.

3. Report on Current Status

Given.

4. Normative Addendum

4.1 UK Clarification

Jones the Lesser presented N208, the revised UK addendum, and described it briefly.

Plum expressed the concern that interpretations *can* affect what goes in test suites. Hence, any differences between the UK addendum and the X3J11 TIB could lead to different validation requirements for ANSI and ISO C. The two documents are hard to reconcile exactly, however, because of their different formats. The UK addendum is a series of examples with explanations, while the TIB is questions and answers.

Gwyn felt that four open issues remain with the UK addendum:

1) The UK draft says that <...> always parses as a header name, which hardly has universal support.

2) The type of a function call expression is not specified in the C Standard and still needs an X3J11 interpretation.

3) The nature of a function returning const void is still unclear.

4) The floating-point representation given doesn't take the form of an example.

After some discussion, Gwyn agreed that #3 was resolved. Jones the Lesser, Gwyn, and MacDonald agreed to propose amendments to the UK addendum later in the meeting to resolve these open issues.

Gwyn presented 91-055 as the draft TIB with 92-011 edits. The committee agreed to accept all edits except #18 (RFI 17 #37).

Jones the Lesser asked whether the TIB would be converted from ANSI to ISO section numbering. Gwyn showed the mapping table included in the TIB, but said he lacked the time to do more.

Meyers/Weidenhofer

"Move we adopt 91-055 as the base document for the Technical Information Bulletin."

Motion carried 17/0/0/15/32.

Prosser requested offline time with Gwyn to review edits to the TIB. Granted.

4.2 Danish Alternative to Trigraphs

Plauger observed that the amended Danish proposal was not available in written form. He objected to any further discussion until such a document was produced. Simonsen agreed to produce it.

4.3 Japanese Multibyte Support Extension

Fukutomi presented N210, the redactor's report on the Japanese MSE (N205). He explained why iswpunct should not be as comprehensive as ispunct (isprint minus isalnum and isspace). No objection.

There was some discussion of "C" locale restrictions on iswspace, and what supersetting was permitted. Prosser proposed removing most "C" locale restrictions from the isw* functions. Plauger suggested making the isw* functions pure supersets of their is* buddies. Simonsen cautioned that N209 p. 5 lists POSIX constraints that we should be careful not to conflict with.

Fukutomi and Kumagai reviewed other open issues outlined in N217. Weil pointed out additional issues in N220 - the treatment of %C and %S in the w*printf functions and the naming of wcswcs (wcsstr preferred).

4.2 Danish Alternative to Trigraphs (Revised)

Simonsen submitted N218 as the latest Danish proposal. There was considerable discussion.

Prosser objected to the presence of NOT and NE macros as superfluous. He observed that BITAND is a poor name for & used as the address-of operator. And he noted that %% conflicts with prior art (the language C*).

Jaeschke observed that the <> digraph conflicts with an X3J11.1 proposal, as has been pointed out in the past. He felt that the macros AND, BITAND, and AND_EQ were unnecessary.

MacDonald asked why there was no NEQ macro.

Plum noted that X3J16/WG21 failed to include %%%% (probably an oversight) and-<> (more deliberately). Also NE is spelled NOT_EQ.

Frankel saw other uses for %%. He also observed that both NE and <> simply remap !=.

Gwyn objected (yet again) that the entire digraph approach was an attempt to solve a vanishing problem.

Plauger noted that no really good technical solution is likely to surface after so many years of study. The issue is now much more politics than technology – are we going to help Denmark find a palatable solution and will Denmark accept one without still more changes? If not, then WG14 must give up on consensus in this area. Otherwise, the entire normative addendum will be at risk.

Simonsen defended the alternatives to the ! operator as convenient for EBCDIC machines.

Jarvis noted that the current proposal now differs from what is in the C++ draft.

Farance felt that %% and <> cause problems for other extensions.

The discussion continued in this vein for some time. Eventually, Plum suggested that we either drop the Danish proposal entirely or revert to the version already adopted by X3J16. Frankel suggested a variation on this theme.

Simonsen agreed to draft yet another proposal for consideration later in the meeting.

4.3 Japanese MSE (Revisited)

Kumagai raised several issues regarding conversion errors in wide character streams (N217 #1):

Should we describe conversion errors for each function that can cause them, or just once? Prosser suggested that all wide character I/O be described "as if" they call fgetwc and fputwc. Then we can describe conversion error for just those two functions and clarify elsewhere as needed.

Should fscanf report a conversion error on a bad multibyte string? Prosser felt that we need a new term less easily confused with older concepts. Plauger observed that the issue is not confined simply to fscanf.

What should we do with such conversion errors? Prosser noted that prior art exists in this area. We should set errno to EILSEQ on an error converting either way between wide characters and multibyte. Agreed.

» Prosser will suggest wording changes for the MSE.

Kumagai asked (N217 #2) whether we needed to distinguish between a wide character and type wchar_t. It was understood that not all representable values of wchar_t make valid wide characters. What should we say about wide character encodings? As little as possible. Should we clarify that a multibyte

shift sequence does not constitute a wide character? Yes. Should we change the description of MB_CUR_MAX to make it clearer? No, it's good enough.

Kumagain asked (N217 #3) whether we should continue to leave out print-position counting in the *printf functions. Many people in China, Japan, and Korea want it. Still no support.

Kumagai asked (N217 #4) whether we should add functions that expose control of shift states when parsing/converting multibyte strings. Molenda has proposed adding a type state_t and new functions for this purpose. Deferred until later in the meeting.

Weil proposed (N220) that we change the usage of the %C and %S format specifiers. He favored defining these in the w*printf functions to print char data. He also suggested that we use %hc and %hs for char in all functions, %lc and %ls for wchar in all functions. There was some discussion of the need for this capability.

Straw vote:
 3 add char printing to the w*printf functions
 9 no

Weil asked (N220) that we change the name of wcswcs to wcsstr. There was a brief discussion.

Straw vote:
15 change wcswcs to wcsstr
 4 no

Weil asked (N220) whether it is permissible to declare the MSE names in existing headers. Prosser noted that many are covered under "future directions" but not all.

Straw vote:
 1 reserve all MSE names in relevant older headers
17 no

Prosser requested (91-072, p. 6) that we remove the "C" locale restrictions for iswalpha, iswspace, iswlower, iswupper, and iswpunct. Instead, he proposed a supersetting constraint for each of these, of the form:

```
isalpha(c) != 0  ==> iswalpha(c) != 0
```

Straw vote:
20 replace "C" locale restriction with supersetting rules
 2 no

This change subsumes the Japanese request to relax the constraint on iswpunct.

Prosser asked (91-072, p. 5) what we should do for printf("w.pS", wcs) when the precision p is too small to put out all of wcs. The agreed behavior is to put out as many complete characters as possible as a valid multibyte

string (including any trailing shift states) such that no more than p characters are produced.

Also, what constitutes a terminating space for fscanf("%S", &wcs)? Since fscanf is intended to remain multibyte ignorant, this should be any (1-byte) char for which isspace(c) !=

Straw vote:
15 isspace(c) stops %S in fscanf
3 no

» Plauger and Prosser will assist in writing the rationale for fscanf("%S", ...).

4.1 TIB (Revisited)

Gwyn asked (92-024 re p. 26 of 91-055) whether it is permissible to declare

extern const void end;

so that one can write &end and avoid an explicit constraint violation. Yes, thanks to a loophole. He also proposed a wording change to Q32, p. 55, interpretation of str(n)cmp. Accepted.

Jones the Lesser proposed changing "programmer" to "translationunit" in TIB #58, but Plum objected. To be revisited later in the meeting.

» Plum and Jones the Lesser will proof read edits made by Gwyn to the TIB.

5. Interpretations

The committee broke into subgroups to preprocess interpretations.

4.3 Japanese MSE (Revisited)

Prosser suggested we add an explicit state memory argument to wcstok (of type wchar_t **), as does POSIX, to eliminate need for static memory. Agreed.

Plauger summarized arguments for and against dropping the %[...] format specifier from the w*scanf functions.

Straw vote:
13 drop %[...] from w*scanf
7 no

Plauger presented arguments favoring binary wide character streams, which are currently undefined.

Straw vote
12 define binary wide character streams
2 leave undefined

Note that seeks on such streams still have the same constraints as for text and multibyte streams.

Plauger presented a request that we drop the statement "precision is ignored" for %C in f*printf, as an overspecification. Agreed.

Prosser requested that we change the environmental limit on the w*printf functions from 509 wide characters to 509 bytes. Agreed.

Prosser requested that we say iswctype is undefined, not implementation defined, if the wc_prop argument is befuddled. Agreed.

Plauger presented arguments for adding multibyte conversion functions with explicit state arguments. (This permits a program to parse multiple strings at the same time, keeping state memory for each.) The specific proposal is from Molenda: Add the type state_t and the macro MBSS_I (for the initial shift state). Then define alternate versions of mblen, mbstowcs, wcstombs, mbtowc, and wctomb. (The first three functions are restartable.)

Straw vote:
11 add conversion functions with controllable shift states
4 no

Still other extensions were not presented. These include exposing the shift state of a stream so that it can be controlled and allowing multiple multibyte encodings for different streams,

4.2 Danish Alternative to Trigraphs (Revisited)

Simonsen presented N222, a modified proposal. It adds the macros BEGIN and END for braces and XOR for ^=. The tokens are:

```
<:   :>   <%   <%<% <>
[    ]    #    ##       !=
```

Several people immediately objected to the continued inclusion of <>. Weil observed that :> collides with a popular extension to Microsoft C.

After some discussion, Simonsen offered to resubmit N218 (the "Tokyo" solution) with two changes – dropping <> and replacing NOT_EQ with NE.

Straw vote:
2 accept N218 with the two changes
15 no

Straw vote:
8 accept N222 with <> replaced by NE
9 no

Straw vote:
7 try to solve digraph problem some way
12 no

5. Interpretations

Jaeschke presented #17, 92-014, p. 41, on our failure to specify the type of a function call expression. Some confusion arose over the distinction between the representation of the return value and its type. It was understood that widespread practice permits the representation to be more generous than required by the type. With that understanding, there was no disagreement that the type of a call to function returning X is X.

At issue, then, is whether such a clarification requires an amendment to the C Standard or whether it can be clarified by an interpretation. There was lots of discussion.

Straw vote:
20 fix the type of function calls as an interpretation
0 make an amendment

Meyers agreed to supply section number references for the interpretation.

Jaeschke presented a request for interpretation from Meyers. Is puts("ñ") strictly conforming? Some felt that it is not. Others felt that its behavior is locale specific, hence strictly conforming. There was lots of discussion.

Straw vote:
13 printing funny characters is not strictly conforming
6 yes it is

Straw vote (X3J11 only):
15 printing funny characters is not strictly conforming
5 yes it is

>>>>>> U.S. TAG Meeting <<<<<<

Jaeschke convened the US TAG meeting, after asking everyone not representing US-based enterprises to leave the room. He requested guidance on the position the US should take on the three parts of the normative addendum.

Jaeschke/Terrazas

"Move we endorse the UK portion of the normative addendum including previously approved interpretations."

Motion carried 20/0/0/12/32.

Plauger/Jaeschke

"Move we direct the US representative to support the MSE, given its current description, and empower the US delegation to deal with open issues."

Motion carried 18/0/2/12/32.

Jaeschke/Terrazas

"Move we direct the US representative to oppose adoption of any alternate spelling of tokens as part of the normative addendum."

Motion carried 16/1/3/12/32.

Prosser/Weil

"Movewe direct the US representative to oppose any additional contributions to the nomrative addendum."

Motion carried 18/0/2/12/32.

Jaeschke asked that we clarify that such opposition is in the interest of getting the normative addendum out. Many topics may be of interest in the future. Frankel asked that the C++ committee be informed of the rationale behind these votes.

Plauger asked for any further input on ISO 10646. There was none.

Plum observed that we needed to approve a US delegation to the WG14 meeting.

Plauger/Farance

"Movewe approve as US delegation to the 15 May WG14 meeting Plum, Plauger, Molenda, Sivakumar, Prosser, Farance, Jaeschke, and Weil."

Motion carried 20/0/0/12/32.

Jaeschke/Farance

"Move we approve as US delegation to subsequent WG14 meetings Plum, Plauger, Molenda, Bonito, Sivakumar, Prosser, Farance, Jaeschke, and Keaton."

Motion carried 20/0/0/12/32.

The US TAG meeting adjourned.

>>>>>>WG14 Meeting<<<<<<

Plauger convened the WG14-specific meeting at 8:30 am on Wed 15 May '92, while X3J11 continued with interpretations. (Stanberry minutes follow these.) Jones the Lesser (hereinafter Jones the Only for these particular minutes) agreed to serve as chair, by popular demand.

4.1 UK Proposal (WG14)

Jones the Only reported that the UK draft is ready, except for one or two holes. He stated that the function call type issue is the only outstanding issue of major importance. Given an X3J11 interpretation, he expects the final draft to be ready for balloting by Dec '92. Jaeschke expressed US support for all interpretations to date, but couldn't guarantee support if additional interpretations are added.

4.2 Danish Proposal (WG14)

Simonsen reported that the Danish C Panel insisted on reinstating <> after the Milan meeting. He asked where the other members now stood on the Danish proposal.

Jones the Only said that the UK now favors macros, but no digraphs because they might change existing code.

Jaeschke reported that the US now opposes any digraph solution, because the proposals keep changing.

Noda said that Japan wants to support the Danish proposal from an international standpoint, but doesn't want to delay the normative addendum. He asked whether the proposal can be made optional instead of part of the C Standard. Simonsen replied that Denmark wants digraphs to be mandatory.

Plauger suggested that Denmark make one more try to draft a proposal that can achieve consensus. If they fail to do so by the Dec '92 meeting, however, they must face the serious possibility that the normative addendum will be voted out for balloting without a digraph proposal.

4.3 Japanese MSE Proposal (WG14)

Noda reported that the MSE will be ready for final review by Dec '92. Prosser noted that he provided specifications for restartable mb(s)towc(s), as requested. He expressed concern, however, that so much new machinery added to the MSE might delay stability.

In light of these comments, Plauger suggested that the tentative Aug '92 meeting in London not be held. Instead, he encouraged all three redactors to prepare final drafts in plenty of time for mail distribution and review *before* the Dec '92 meeting in Washington DC. **The cutoff date for the last mailing is 7 Nov '92.** All submissions must be received by Plauger on or before that date.

» Molenda will have IBM duplicate and mail WG14 proceedings (subject to approval).

4.4 Additional Proposals (WG14)

Jaeschke requested that the additional proposals not be considered at this time, in order to expedite completing the normative addendum. Simonsen disagreed, saying they were important. He wanted to make some of them part of the normative addendum. Jaeschke reported tha the US would vote against including anything else. Jones the Only and Kumagai agreed that new additions should not be considered at this time. Simonsen felt that this position was unfair.

Plauger asked for guidance on ISO 10646. Denmark and Japan felt that the UTF (proposed multibyte encoding described in an appendix) should be normative. Several people objected to floating diacriticals.

Simonsen asked for time to discuss extended identifiers in C (N214, N219) and was granted limited agenda time. Plum noted that his N203 summarized some of the problems. One approach is to define a "source locale" and add an isident class. It was noted that work on this topic is also under way in WG15, WG20, and WG21. Molenda favored isalpha as the class for identifier characters, so long as the "C" locale is restricted to the usual 26*2 letters for portability.

Plum observed that linking with object libraries is also an issue, since these provide external names. Molenda noted that the whole compile process must be defined in terms of some codeset. There was further discussion of the inclusion of large character sets in identifiers.

Plauger suggested that WG14 needs a new work item to proceed. He agreed to raise the issue before SC22.

5. Interpretations (WG14)

Plauger expressed the opinion that interpretations are very much up in the air within SC22. Plum noted that Keizer and Simonsen both have supported the existing "defectreporting" machinery for processing interpretations, even if the leadership doesn't yet agree. The consensus was that we must muddle on as before until we get clearer guidance from SC22.

» Plum agreed to convert the ANSI TIB to ISO section numbers.

6++. Administrivia (WG14)

There were no formal votes to summarize and no other business.

The next meeting of WG14 will be held 8-10 Dec '92 in Washington DC. This partially overlaps both the X3J11.1 (7-8) and the X3J11 meetings (9-11), making everyone equally rushed and uncomfortable. Plauger said he would organize the agenda as much as possible so that pressed for time could leave early. There were several grumblings from people who wanted to attend the other meetings as well.

Future meetings are tentatively planned for:

Jul '93	London UK	BSI
Dec '93	Kona coast HI	Plum Hall

Jones the Only thanked our host, DECUS, for providing excellent support. Accepted by acclamation. The meeting adjourned to rejoin X3J11 at 11:15 am on Fri 15 May '92.

6. Interpretations (revisited)

Note: all interpretation responses were accepted unanimously by the committee, except where noted otherwise.

Stanberry presented RFI #32 (91-036). The question is whether or not a conforming implementation may allow a comma expression in a constant expres-

sion. Section 3.4, page 56, line 10 expressly disallows the comma operator (except within the operand of a sizeof operator), but page 57, line 1 says "An implementation may accept other forms of constant expressions." The consensus of the committee was that a comma operator in a constant expression must be diagnosed as a constraint violation by a conforming implementation. Further discussion clarified the intent of allowing "other forms" was to accept, for example, the expansion of the offsetof macro, which might otherwise yield forms of expressions that would not be semantically acceptable.

Stanberry presented RFI #36 (91-040). This was a formal request to confirm the informal decision by the committee previously that the standard allows a floating-point constant to be represented with more precision than implied by its type (see 91-032, item 9, page 13). The committee agreed with the arguments presented by Tydeman.

Keaton presented RFI #37 (91-043). Is it true that both 10646 and UNICODE fail the rules in section 2.2.1.2, page 12, lines 22 and 31-32, and therefore cannot be used as C multibyte character sets? The committee cited section 1.6, page 3, lines 1-7 and 23-25 (the definitions of byte, character, and multibyte character), and concluded that if byte size < UNICODE character size, then can't be used; otherwise can be used. UNICODE does violate these rules since it uses 8-bit bytes, and 16 or 32-bit character codes. It was also noted that this request was with respect to an old UNICODE, but it applies to the new UNICODE as well. (10646 is obsolete and was therefore not addressed.)

Keaton presented RFI #38 (91-046). Are macros in function argument expressions completely expanded, or only until a preprocessing token is formed? According to section 2.1.1.2, page 6, lines 3-5, they are completely expanded.

Jervis presented RFI #40 (91-062). There were 9 items addressed:

[1] Composite type. For:

```
void f(const int);
void f(int a)
{
a = 4;
}
```

What is the composite type of f? The type is "void f(int a)", i.e., the composite type of the parameter is the type without qualifiers. Is the assignment legal? Yes; the type of a is int. In the discussion of determining composite type of functions and their parameters, it was reaffirmed that the composite type is determined as described in previous interpretation #13.1.

[2] Is an implementation that fails to meet Environmental Limits in sections 4.9.2, 4.9.3, 4.9.4.4, 4.9.5.5, and 4.10.2.1 non-conforming? Yes. An implementation must conform to all environmental limits expressed as shall's.

[3] Must an implementation diagnose violations of Environmental Constraints? No, section 2.1.1.3 applies only to constraints in section 3 (the language section). See the definition of constraint violation in section 1.6.

[4] This is the same as RFI #17.39, namely, on how

```
if (a<b||c>d) ;
```

should be lexed. The committee reaffirmed the answer given to the previous RFI, as amended by 92-024, that <...> tokens are lexed specially only in the context of #include's. Brodie asked for confirmation of the amendement to TIB first; although this particular amendment was accepted, the status of the TIB was still open (pending resolution of RFI #17.37).

[5] In section 3.5.2, line 3: what is meant by "set"? Is

```
int int i;
```

a constraint violation that must be diagnosed? Yes. It is allowed to rearrange the order of specifiers within a set, but not to duplicate them.

[6] Questions about the offsetof macro.

(a) Is it legal to use with an incomplete struct type?

```
struct t1 { char c;
  short s;
  int i[offsetof(struct t1, s)];
}
```

Not legal; cannot refer to an incomplete struct in an offsetof macro, and struct t1 is incomplete at time of reference (section 3.7.2).

(b)-(d) Also not legal because they refer to struct members that are not yet in scope.

[7] Questions about sizeof.

(a) The argument of sizeof must be an expression. In:

```
void f(int c, char a[sizeof(c)]);
```

is "c" an expression? Yes, refers to a function parameter (section 3.5.4.3).

(b) In:

```
int i;
struct { int i;
    char a[sizeof(i)]; };
```

is "i" an expression? Yes, i refers to the extenal declaration since member i is in a differ name space (section 3.1.2.3).

[8] Questions about implementation-defined behavior of assignments to char objects (section 3.1.2.5). Consider:

```
char c = 7;
c = 'a';
c++;
c = '1';
c++;
```

Discussion of 3.1.2.5, especially with English reading of ":" in cited statements followed. Committee agreed that "reasonable reading" of this section (and use of : as a connective) implies that none of these assignments are implementation-defined. Also considered:

```
char d = '\07';
d = '$';
```

The former is not implementation-defined (section 3.1.3.4, page 30, lines 33-35), but the latter is locale-specific and therefore implementation-defined.

[9] Request to reconsider our interpretation for RFI #17.27. Deferred to later (and reaffirmed then).

Plum suggested at this point that at least two letter ballots be sent on the interpretations, one for the unanimously approved ones, and a separate one for the controversial ones. Meyers asked what happens if we don't get response to an interpretation within 2 years. X3 rules need to be studied for answer.

Gwyn presented RFI #43 (92-009).

Q1: ... Standard C appears to permit ...

#define NULL 4-4

in the standard headers. Is this correct? If not, what are the requirements regarding NULL?

A1: NULL is an object like macro, and the restrictions that are given for function-like macros in section 4.1.6 (e.g., that they be fully protected by parentheses where necessary) are reasonable to extend to object-like macros.

Q2: Can an identifier that starts with underscore be defined as a macro in a [strictly conforming] source file that includes at least one standard header?

A2: No, shall not define a macro whose name begins with underscore followed by an uppercase letter or another underscore since these names are reserved (section 4.1.2.1). Footnote 91 also expressly disallows macro names matching any reserved names.

Gwyn presented RFI #44 (92-010).

Q1: How is section 4.1.5, page 99, lines 24-30, regarding the offsetof macro to be interpreted? Is the expansion of offsetof (a) an expression which can be evaluated during translation, the value of which is in the range representable by a size_t type; or (b) an expression as (a) above, but further constrained to be an "integral constant expression" as defined in section 3.4, page 54, lines 17-21?

A1: Exact reading of cited section requires (b) as the correct interpretation.

Q2: Must a conforming implementation provide strictly confoming expansions of macros defined by standard headers, such that any use of the resulting preprocessing token sequence, and ultimately the token sequence, beyond phase 4 does not alter the behaviour of an otherwise strictly conforming program?

A2: Suggested response is that a conforming implementation need not provide strictly conforming expansion of macros defined by the standard headers. Discussion noted that there is no explicit section to cite to justify this response. It was suggested that this was deliberately left unspecified in the standard.

Q3-Q5: not addressed because they are conditional on A1 being "a" instead of "b".

Q6: What mechanism exists to determine whether token sequences originated from the program source? A6: The standard does not require a mechanism to determine where token sequences originated.

Q7: How to interpret "strictly conforming program" in section 1.7, given definition of translation unit in section 2.1.1.1? A7: A program is not rendered not strictly conforming by any characteristic of the implementation. What to cite to support this?

The responses to this RFI were not agreed upon unanimously, and further discussion was postponed until after all other RFI's had been presented.

Frankel presented RFI #39 (91-061).

Q1: Is value of MB_CUR_MAX 1 in the C-locale?

A1: The "facts" cited by the requestor do not logically lead to the conclusion, and fact #3 is not correct as stated (assuming the intent of the statement was to assert that character constants are of type char, this is an error because character constants have type int). The facts also deal with single byte chars, and not the extended character set. A counterexample was presented: a minimal environment may still require more than one byte for multibyte characters, so it is false to conclude that MB_CUR_MAX is 1 in the C-locale (sections 4.10 and 4.4.1.1).

Q2: Request for a change to the standard. Response could be annotated with example on how to determine if this is the C-locale. Also there was discussion on 91-075. Was this ever recorded as official RFI with CBEMA?

» Plum will check status of 91-075 as an official request for interpretation.

Frankel presented RFI #41 (91-076).

Q1: Are the characters in 2.2.1 always classified as implied by 2.2.1 regardless of the locale specified?

A1: Yes, as is indicated in the cited section, and in sections 4.3.1.6 (islower()) and 4.3.1.10 (isupper()), which require that upper and lowercase letters be present in basic source and execution character sets. Further, section 2.2.1.2 indicates that these single-byte characters must also be present in multibyte character sets. Hence,

Q2: Are uppercase and lowercase letters in every locale?

A2: Yes.

Frankel presented RFI #42 (92-001). Background citations were given for object: section 1.6, page 3, lines 26-30, and section 4.11.1, page 163, lines 5-7; and for overlapping object: section 4.11.2.1, page 163, line 16. It was agreed that the "objects" referred to by memcpy() are exactly the regions of data storage pointed to by the pointers and dynamically determined to be N bytes in length.

Q1(a): Are the objects in the description of memcpy the largest objects into which the arguments can be construed as pointing?

A1(a): No.

Q1(b): Is the behavior of the call of memcpy in:

```
void f1(void) {
extern char a[2][N];
```

```
memcpy(a[1], a[0], N);
}
```

defined, because the arguments point into the disjoint array objects a[1] and a[0]?

A1(b): Yes.

Q1(c): Or is the behavior undefined because the arguments both point into the same array object a?

A1(c): The behavior is defined because the pointers point into different non-overlapping objects.

It was further agreed that objects are regions of data storage, unrelated to declarations or types.

Q2(a): For the purposes of the description of memcpy, can a contiguous sequence of elements within an array be regarded as an object in its own right?

A2(a): Yes, for memcpy.

Q2(b): If so, are the objects in the description of memcpy the smallest contiguous sequences of bytes that can be construed as the objects into which the arguments point?

A2(b): No, not the smallest either.

Q2(c): In the following:

```
void f2(void) {
extern char b[2*N];
memcpy(b+N, b, N);
}
```

can each of the first and last half of array b be regarded as an object in its own right, so that the behavior of the call of memcpy is defined? A2(c): Yes, the non-overlapping halves of the array can be regarded as objects in their own right.

Q2(d): Or is the behavior undefined, since both arguments point into the same array object b?

A2(d): No, the behavior is defined.

Q2(e): In the following:

```
void f3(void) {
void *p = malloc(2*N); /* Allocates an object. */
{    char (*q)[N] = p; /* The object pointed to be p may
 be interpreted as having type
 (char [2][N]) when referenced
 through q. */
/* ... */
memcpy(q[1], q[0], N);
/* ... */
}
{    char *r = p; /* The object pointed to by p may
 be interpreted as having type
 (char [2*N]) when referenced
 through r. */
```

```
/* ... */
memcpy(r+N, r, N);
/* ... */
}
}
```

is the behavior of each call of memcpy defined?

A2(e): Yes, each call is well-defined. The definition of object is independent of the method of storage allocation. The array length is determined by "various methods."

Q2(f): Which of these calls of memcpy give defined behavior?

A2(f): All give defined behavior.

With respect to the char array objects operated on by the string handling functions, it was agreed that the lengths of these arrays are determined dynamically, by "various methods" as described in section 4.11.1. For strings in which all bytes are accessed, length is inferred by null byte termination. For mbstowcs, wcstombs, strftime, vsprintf, sscanf, sprintf, and all similar functions, it was the intent of the standard that the rules in 4.11.1 be applicable by extension (i.e., the objects and lengths are similarly dynamically determined).

Q3(a): Consider a library function for which the number of bytes accessed or modified is affected by the values of the bytes. Is the object associated with each of its pointer arguments the smallest contiguous sequence of bytes actually accessed or modified through the pointer?

A3(a): Yes, the object is the contiguous byte sequence actually accessed or modified through a pointer.

Q3(b): In the following:

```
void f4(void) {
extern char b[2*N];
strcpy(b+N, b);
}
```

is the behavior defined if N > strlen(b)?

A3(b): Yes, the behavior is defined if N > strlen(b).

Q3(c): In the following:

```
void f5(void) {
extern char c[N+M];
strcat(c+N, c);
}
```

is the behavior defined if both N > strlen(c) and M > strlen(c) + strlen(c+N)?

A3(c): Yes.

Meyers presenetd RFI #27 (91-008).

Q: May a standard conforming implementation make characters in its character set that are not in the required source character set identifier characters?

A: References cited are section 1.7, lines 17-21, and 91-007, RFI #26, and its response from this committee. Since this is not strictly conforming, free to extend identifiers. Thus, Appendix F5.2 describes an extension change to syntax in 3.1.2, and a comforming implementation must issue at least one diagnostic for any program using this extension.

Concern was expressed that the committee *not* endorse allowing, e.g., $ in standard conforming character set. There was much discussion, and no clear consensus. It was suggested that this interpretation be deferred.

Straw Vote: Should the committee accept the suggested interpretation response?
Yes 11, No 4, Abstain 5

Brodie asked that we determine what criteria to use to forward an interpretation for letter ballot. Should we require 2/3 or majority vote of the committee to forward an interpretation/response?

Straw Vote (Preference vote)
In favor of majority vote required to forward? 5
In favor of 2/3 vote required to forward? 13

After this vote, repeated vote for Meyers' proposed interpretation response.

Straw Vote: Should the committee accept the suggested interpretation response?
Yes 11, No 4, Abstain 5

Therefore, response for RFI #27 will not be included in the letter ballot.

The question was raised whether the ISO members had any input. No additional discussion on this issue was requested.

Brodie noted that RFI #44 was also deferred until our next meeting.

Jaeschke reported the results of the WG14 independent meeting that took place concurrently with our interpretations sessions. No ISO meeting will be held in August, but they will meet jointly with X3J11 again in December as planned. They will not be submitting the Normative Addenda now, but plan to vote it out at the December meeting.

Meyers presented further arguments on RFI #17.37 (90-056). Suggested rewording of our response on the type of a function call expression. Citations: section 3.3.2.2, line 35; section 3.6.6.4, lines 29-32; and section 3.5.4.3. Suggests that these clearly specifies type of function call expression, therefore sizeof a function call has the same value as sizeof the return type of the function designator. We should insert the wording "Thus the type of the result of the function call expression is the return type of the function designator." The committee agreed to add the insertion to its response.

Jervis presented RFI #40, 91-062 #8, p. 5. Is "char c = 7" implementation defined? No. Related examples:

```
c = 'a', c += 1;        is implementation defined
c = '0', c += 1;        is not implementation defined
c = '\07';              is defined
c = '$';                is locale specific
```

6. Other Business

Brodie revisited the discussion of how X3J11 can handle interpretations: forward without remark to WG14, forward with US recommendations, or simply develop ANSI interpretations. He said that SPARC favors the second alternative. ISO is still deciding how best to handle ISO interpretations. Plauger agreed to ask for guidance at the SC22 plenary in Aug '92.

Brodie asked whether we should forward 91-055, as edited, as the TIB. Gwyn agreed not to include #37 which is still to be approved. No objection.

Plum/Jones the Greater

"Move we empower the chair to take whatever steps are necessary to submit the TIB to X3."

Motion carried 17/0/0/15/32.

» Brodie will shepherd the TIB to X3.

Meyers/Jaeschke

"Move we send out letter ballots on the interpretations, one covering those with unanimous support, the rest in separate ballots."

Motion carried 17/0/0/15/32.

7. Future Meetings

7.1 Future Meeting Schedule

The Washington DC meeting will actually be held in Herndon VA, near Dulles Airport. There was continued grumblings about the overlap of meetings, but no clear solution.

Future meetings currently planned are:

Herndon VA	09-11 Dec '92	Dec Professional
New York NY	19-21 May '93	Farance Inc.
Kona Coast HI	06-10 Dec '93	Plum Hall

7.2 Future Agenda Items

Plauger and Brodie agreed to coordinate interpretations and final review of the normative addendum as much as possible. Jaeschke asked for time at the appropriate point to hold another US TAG meeting.

8. Resolutions

8.1 Review of Decisions Reached

Plauger reviewed each of the resolutions passed at this meeting. The committee also voted thanks to Terrazas and DECUS for their excellent support.

8.2 Formal Vote on Resolutions

The votes are distributed throughout the minutes.

8.3 Review of Action Items

Action items are marked throughout the minutes with a chevron in the left margin. A few last minute commitments are:

» Jaeschke will get information on the Dec '92 meeting.

» Brodie will shepherd the I Project through SPARC.

» Gwyn will edit the TIB to completion.

» Plum will check the status of 91-075.

» Plauger and Brodie will merge agendas for the Dec '92 meeting.

9. Adjournment

The meeting adjourned at 3:30 pm on Friday 15 May '92.

Attrachments:
- Revised agenda
- Attendance