WG14
N173

(Action items are indicated by +++, resolutions by ***.)

1. Opening Activities

The meeting was convened by Plauger at 9:30 at the ITSCJ offices, Kikai-Shinko building in Tokyo, Japan.

Ms. Mariko Takeuchi from ITSCJ welcomed WG14 members to Japan.

1.1 Goals and purposes

Plauger indicated that the purpose of this meeting was to continue work on the various parts of the normative addendum.

1.2 Selection of meeting chair

Plum agreed to serve as meeting chair. Jaeschke agreed to serve as meeting secretary.

1.3 Approval of previous minutes (N159)

Jaeschke indicated the following changes/corrections:

Point 1.5: Steve Carter is NOT the chair of X3J16. Strike this phrase.

Point 2.1 The meeting of X3J11 was 6-8 March, NOT May.

Point 2.2 Strike the sentence ``They do not have a standard yet." since NCEG is NOT working on a standard.

Point 2.4.2 Change ``X3J11" to ``X3J16."

Simonsen indicated the following changes/corrections:

Point 2.4.1 ??? (details missing)

Point ??? (details missing)

With these changes the minutes were approved.

1.4 Review of action items

#1. Simonsen Done. See N152 and N153.

#2. Plauger. Action pending.

#3. Simonsen will rewrite N117 ... Not yet done.

#4. Jaeschke. SC22 advised that WG14 CANNOT simply create a working group as RNE since RNE is much more than a short-term ad-hoc group. Jaeschke recommended that WG14 continue to monitor NCEG informally and NOT create a formal working group.

#5. Plauger. Requested in N162. Result pending.

#6. Japanese delegation. Reformat of N111 done as N145.

#7. Japanese delegation is working with X/OPEN and UNIX International on an ongoing basis.

#8. Simonsen. E-mail distributed but no formal response done yet.

#9. Simonsen. N143 revision done.

#10. Chacon. Not done.

#11. Martin. Done.

#12. Keizer. Done.

#13. Chacon. Unknown.

It was reported that van Brumen forwarded a letter containing other members' `signatures' without their authorization. This caused concern from those members.

+++ ACTION: Jones will check with van Brumen to see what, if anything, was done re the validation letter to SC22. If necessary, Jones will request procedure information from SC22 re validation.

#14. Plum. Done.

#15. Plauger. See N162 request. Reply outstanding.

#16. Simonsen. Status unknown.

#17. Simonsen. Not done.

1.5 Approval of agenda (N161)

The agenda was revised as follows:

2.5 is to include liaison reports for X3H5 and SC2.

Since 4.1 was used for two consecutive items, this section should be renumbered as 4.1-4.4.

The WG14 representation at the SC22 meeting in Vienna was added to 6.

1.6 Introduction of participants

Makoto Noda (Japan) *
Norihiro Kumagai (Japan)
Hiroshi Fukutomi (Japan)
Yoichi Suehiro (Japan)
Derek Jones (UK) *

Rex Jaeschke (USA) *
Tom Plum (USA)
John Benito (USA)
Frank Farance (USA)
P.J. Plauger (USA)
Terry Colligan (USA)

* indicates the heads of delegation.

1.7 Procedures for this meeting

Continue informal debate on proposals re the normative addendum.

1.8 Distribution of new documents

N164	Jaeschke - US position on digraphs
N165	Jones - Style of UK addendum
N166	Jones - UK addendum
N167	Inose - Request fro review
N168	Kumagai and Fukutomi - Rationale
N169	ITSCJ - MSE revision
N170	Noda - Collection of WG14 E-mail
N171	Plum - X3J11 letter ballot results
N172	Simonsen - E-mail to Plauger

1.9 Roll call

Delegates from three nations were identified: Japan, UK, and USA.

2. Reports in liaison activities

2.1 X3J11

Plum reported that Brodie (X3J11 chair) recently attended an ANSI SPARC meeting where he requested that X3J11 be able to publish a document containing responses to all Requests for Interpretation (RFI) to date. This was approved and Gwyn will edit the document.

+++ ACTION: Jaeschke will monitor production of this document and will keep WG14 informed of the expected due date and how it will be distributed.

*** Resolution #1: WG14 requests that when X3J11 produces the RFI document they include ISO C Standard section references as well as those to the ANSI Standard.

Plum also reported that X3J11 has just completed a letter ballot to replace X3.159-1989 with ISO/IEC:9899-1989(E) as the ANSI C Standard. The final vote was: 23 Yes, 3 No. X3J11 is required to respond to all No votes according to ANSI rules. This vote paves the way for X3J11 to create an `I' project and to track the WG14 work more closely.

Jaeschke reported the following:

The US had formally considered the latest Danish proposal for Digraphs. See N164.

The December '91 and May '92 meetings would be joint with WG14. The December '92 meeting would be 9-11 in Washington DC.

The US was supportive of the MSE proposal from Japan.

Jaeschke reported on his attendance at the US TAG meeting for WG20, a new ISO committee for internationalization.

2.2 NCEG

Jaeschke reported on the March '91 meeting. NCEG is now officially X3J11.1. That is, a working group within X3J11. The main areas of activity are:

1. complex arithmetic
2. restricted pointers
3. variable-dimensioned arrays
4. IEEE/floating-point support issues
5. array syntax
6. extended integer precision
7. initialization issues

NCEG expects to have drafts for parts 1, 2, 4, and 7 available about January of '92 for public review.

Jaeschke will give a presentation of NCEG to Japanese members Thursday May 16th 13:00.

The ANSI committee X3H5 has drafted a parallel processing standard containing a model and Fortran and C language bindings. The NCEG parallel processing group was been reactivated to monitor this activity. An E-mail distribution list has been established to address this. It is called: nceg-par@convex.com. To get your address added to this list contact Austin Curley at curley@convex.com.

NCEG is now supplying meeting minutes to WG14.

There is now a general NCEG E-mail distribution list. It is called nceg@cray.com. To get on this list contact Tom MacDonald at tam@cray.com.

2.3 X3J16

Plum reported on the C Compatibility subgroup. Notable progress has been made within X3J16 for compatibility with the ISO C library. But there are some problem areas:

* There is serious concern as to whether setjmp/longjmp will be supported at all and if it is, whether it will always work reliably (as in the case of destructors being called).

* There is also concern about whether the traditional C signal handling will be supported.

* X3J16 is interested in how WG14 will handle the MSE from Japan. X3J16 intends to have string classes/libraries for char strings and widechar strings as well as corresponding support in the streamio package.

+++ ACTION: Plum will make sure that the relevant X3J16 library information re strings be distributed to the MSE proposers.

2.4 WG15

No report since Martin, Chacon, Simonsen were not present.

2.5 Other liaison activities

SC2 - Plauger reinforced that the C Standard did NOT specify character set mapping. Source and execution sets could be different. There was a lengthy discussion on '\0' and '\n'.

The discussion then moved to the requirement that L'a' == 'a'. It was agreed that L'a' must be a constant expression. And if we intended this equality to hold true across all locales used in one program, then a 10646-based locale could not change this equality.

Plauger described at length why he though this equality was a good idea. He used writing scanf as an example where the format was a multibyte character string. The main problem was how to recognize white space in the format and how to test for it since isspace is not currently defined for a widechar. (This is a problem since the multibyte characters in the string must first be converted to widechars before further processing.)

1) '\0' has long been used for end-of-string in string.h (not containing multibyte characters) and this will NOT change.

2) '\n' has also long been used to indicate end-of-line by stdio functions.

3) It is convenient to require that '0' + 1 == '1'. This is true for all known character sets.

4) L'\0' is used to indicate the end-of-widechar-string in stdlib.h and for MSE proposal. But L'\0' need not equal any external character named "NUL".

5) L'a' == 'a' is required for the basic C execution code set only.

+++ ACTION: Plum will write a defense of requiring L'a' == 'a' and after circulating to WG14 by E-mail, will forward it to SC2.

X3H5 - Parallel C binding

*** Resolution #2: WG14 requests X3H5 to follow the namespace requirements in their C binding. For example, new keywords and external identifiers begin with either two leading underscores or an underscore and a capital letter.

+++ ACTION: Jaeschke will formally notify X3H5 of this resolution.

3. Current status of WG14

The ISO C Standard is final. It is called ISO/IEC:9899-1990(E).

X3J11 is attempting to adopt this as the ANSI C Standard, replacing X3.159-1989.

4. Normative addendum

4.1 UK clarifications

Jones presented N165.

Plauger spoke at length on the merits of using examples in the clarifications.

*** Resolution #3: WG14 requests the UK clarification part of the normative addendum be organized as examples with explanatory text with an introductory note regarding the intent of the clarifications is to NOT change the meaning of the standard.

There was significant discussion of whether or not the RFI report being produced by X3J11 can and should be somehow coupled to this clarification effort. It was finally agreed they were two separate efforts and that if something could be shared, fine; otherwise not.

4.2 Danish alternative to trigraphs

The `paper' is a recent E-mail from Simonsen.

Jones briefly presented the UK's latest position based on the recent C Panel meeting. Essentially, the UK showed no real support for the proposal.

Plauger gave a detailed history of the whole issue. He argued strongly in favor of WG14 something something along the lines of what was being proposed since it does adhere to the guidelines WG14 established in Copenhagen.

Jaeschke saw no problem in selling the latest proposal to X3J11.

Jones suggested that the name <iso646.h> can't be used since it contains digits. It was pointed out that while an implementation must be able to recognize and handle a standard header name, this need not be the actual name of any resulting text file by the same name. A file system that did not support digits in file names may have to translate it to something it can support just like they may have to do with a period now.

Jaeschke suggested that xor be named bitxor to be consistent with bitor and bitand. Also, why is != a macro? It is not needed and if it is to be proposed why isn't == there too? He proposed removing != and maybe ! as well. He also advised that :> was an extended operator in Microsoft C V6.0 for based pointer operations.

Plum described X3J16's position re the previous version of this proposal and how it conflicted with that of X3J11.

Plum/Plauger discussed at length the merits of having digraphs dealt with in phase 1 versus phase 5. Plum argued the merits of having it in phase 1. This allowed a separate tool to do the conversion and the compiler lexer would need no change. However, it was shown that this would disallow the use of < ## : from producing the token <:.

Plauger/Colligan argued loudly that we should `do it right.' If we are going to add these as operators they must behave like all the other operators. This means dealing with them in phase 5. This also avoids breaking existing code that uses the stringize operator #. For example:

```
#define STR(x) #x
```

```
STR(<%)
```

results in a string "<%". However, in phase 5,

```
void f() <% %>
```

results in the preprocessing token pairs < and %, and % and > to be converted to language tokens <% and %>, respectively.

A discussion then followed regarding benign redefinition. For example, are:

```
#define begin <%
#define begin {
```

the same thing? Plauger thought definitely not. The two sets contain different preprocessing tokens.

Sekiguchi (Japan's liaison to X/OPEN) arrived and commented that maybe some of these changes would affect embedded preprocessors such as SQL. There was some discussion of this but no real issue was found.

Kumagai spoke strongly in favor of preserving the strict definition of a pure extension. That is, not breaking existing strictly conforming programs.

*** Resolution #4: WG14 endorses the digraph proposal from Denmark (E-mail WG14.109 page 3) that involves the addition of five alternate spellings <%, %>, <:, :>, and %% for {, }, [,], and #, respectively, and the new header <iso646.h>. [Implicit in this was that these new spellings be recognized in translation phase 5 NOT phase 1.]

4.3 Japanese Multibyte Support Extension (MSE)

N168, N169

Jones had questions/comments on the rationale N168.

Kumagai presented a short history/introduction of N169.

There was a discussion to identify the specific topics we should focus on.

Why do we need to add %C and %S to the printf/scanf family if this can be handled by the new functions?

Sekiguchi responded by saying that X/OPEN's next draft specified them this way since the proposed new widechar versions were not yet part of Standard C. Also, if you wanted to mix a small amount of multibyte I/O with single byte I/O you could do it easier using these enhanced versions. Besides, we already allow multibyte characters in the format for these functions so it's not much different to allow them in other arguments too.

Plauger indicated that if these changes to printf/scanf were adopted the wording would have to be cleaned-up to handle shift state issues properly.

The issues of namespace pollution of the proposed header wchar.h were next.

Plauger noted that C continues to evolve and that do we want the MSE proposal to be a first-class addition to standard C or not.

The following four ways of handling the new functions were identified:

1) Have a new header wchar.h but its contents would be pure extensions. That is, all names with external linkage must be in implementor namespace. For example, __fwprintf. The straw vote on strongly support/strongly opposed was 1/5.

2) Have wchar.h but all names in it with external language are in the user namespace. For example, fwprintf. Straw vote 7/0.

3) Have no wchar.h and instead split the new functions up according to functionality and place each subgroup in their corresponding existing standard header using names in the user namespace. For example, fwprintf goes in stdio.h. Straw vote 2/3.

4) Hide those new names that are not covered in future directions reservation for library in wchar.h and put others in existing headers. For example, is* go in ctype.h and __fwprintf goes in wchar.h. Straw vote 1/4.

The clear preference was for 2) with none opposed and the majority supporting.

*** Resolution #5: WG14 supports the creation of a new header wchar.h whose names having external linkage are in the user namespace and spelled as in N169.

There followed a discussion of just what names should be in wchar.h. Specifically, the 9 duplicate names should be removed from Annex A since they were already declared in stdio.h. (These were the 3 printf, 3 scanf, and 3 vprintf.)

wchar_t, FILE, and struct tm should be defined in this header. va_list should not and must be implemented in a hidden manner just as is currently required in stdio.h.

When asked by Plum, Fukutomi reported that perhaps half of the proposed MSE functions had been implemented by various vendors. Plauger felt he and others will have even more implementation experience with these before the Milan meeting.

4.4 Additional proposals

4.4.1 Extended identifiers

There was a discussion on compile-time locales and their implications. Even though we don't currently talk about it as such, we do sort of require a compile-time locale now. For example, the encoding of L'a' and L'abc' are `locale-dependent' at compile-time.

Plum presented Simonsen's original proposal with changes to accommodate multibyte characters. He proposed a new predicate iswident(wchar_t) to be sued to define which characters could be used in an identifier. The first character must not be iswdigit, however.

Plum then presented his own proposal where any character not specifically designated for other use could be in an identifier. Denmark has apparent indicated they support Plum's proposal over their own.

(The attendees agreed to extend the meeting time beyond 12:00 on Wednesday as necessary but no later than 17:00.)

Discussion of the actual requirements imposed on us from SC22 re this issue.

Plum also raised the point discussed by X3J11 re the following:

```
#define a$b 1

#if a
  int main() { return 0; }
#endif
```

This was recently determined to be a valid program and thus, seemed to get in the way of allowing extending identifiers.

There was considerable debate about the implications of accepting any not-otherwise-used characters as being allowed in identifiers. Plauger and Colligan spoke against allowing this.

Plauger suggested the simplest solution was simply to allow identifiers to be extended in an implementation-defined manner and thus avoid the whole compile-time locale issue completely. While the locale solution may well be the best solution long-term it would take considerable effort to phrase it in standards' language such that it was both correct and clear.

Straw vote: Should WG14 support allowing extended identifiers via some method? 11 Yes, 0 No.

A straw vote was conducted on the 3 possible approaches.

1) Permissive approach - allow `anything'. 0 really like, 9 really dislike.

2) Minimal approach - implementation-defined extension. 6 really like, 1 really dislikes.

3) Translation-time locale approach. 8 really like, 0 really dislike.

4.4.2 Locales

What, if anything, should WG14 do re collecting/specifying locales? Since there were no prepared papers there was no further discussion.

4.4.3 General characters

N155 Simonsen. Since he was not present and there was no other input there was no further discussion.

4.4.4 Other

Plum introduced the idea of adding a file copy function such as:

wfcopy(FILE *, FILE *)

Colligan and Farance thought this was an operating system function and belonged to POSIX, etc. There would be problems with describing the opening mode. Would need to use filenames rather than FILE pointers.

There was no further interest.

Jaeschke announced he would be seeking public/vendor input on defining __STDC__ for the new ISO C Standard. Since X3J11 did not reserve values for future use nor give direction in this area some vendors already define __STDC__ to be 2 or 3, for example, to indicate a superset. Jaeschke suggested at this stage he favored using something like a value of 1993 (or whatever year the revised standard takes affect) to avoid conflicting with such implementations.

5. Interpretations

N162 Plauger - reply from SC22 outstanding re how we must proceed.

Plauger requested that SC22 Vienna attendees try to find out the process so we can discuss it in Milan with X3J11.

+++ ACTION: Jaeschke will ask Brodie to put aside X3J11 agenda time for Milan meeting to discuss long-term interpretations.

*** Resolution #6: WG14 requests X3J11 to process as a Request for Interpretation the question posed in E-mail WG14.108.

+++ ACTION: Jaeschke will register this RFI with X3.

6. Other business

+++ ACTION: Plum will represent the WG14 convenor at the SC22 meeting in Vienna. Other members may also attend and are encouraged to do so.

Attendees thanked ITSCJ and members of the Japanese delegation for hosting the meeting and providing other valuable assistance.

Kumagai asked when we thought we might be done with the normative addendum. After considerable discussion about future meeting schedule and synchronization with X3J11, etc., it was agreed we should aim to issue a Committee Draft (CD, which is the equivalent of the old DIS) soon after the Salt Lake City meeting in May '92. The final standard could then reasonably be expected by Spring '93.

Plauger needs copies of the revised normative addendum documents in his hands by September 15 latest so he can include them in the mailing.

+++ ACTION: Plauger will discuss with Brodie the best way to integrate the agendas for the Milan meeting since we have only 3 days for both meetings.

+++ ACTION: Plum was instructed to get a clear statement from the SC22 Vienna meeting re our ability to split the addendum into separate addenda.

7. Future meetings

After the SLC meeting in May '92 it was suggested we either meet jointly once again with X3J11 (December 9-11 '92 tentatively in Washington DC) or sometime early in '93. No need was seen for us to meet earlier than that. As such, it was agreed that we will decide on this date of that meeting, in Milan.

+++ ACTION: Plum will make sure meeting information about Milan is available for the next mailing.

*** Resolution #7: Given the principal business of the Milan meeting will be to make a detailed pass of the normative addendum and to take full advantage of the joint meeting with X3J11, WG14 strongly urges the delegations from Denmark, Japan, and UK to have as much representation at that meeting as possible.

6. Other business (continued)

Resolutions 1-3 and 5-7 were adopted 3/0. Resolution #4 (digraph proposal) was opposed by the UK.

The action items resulting from this meeting were reviewed.

4.1 UK clarifications (revisited)

Jones had concerns how he would organize some of the issues as examples. He cited N166 cai003 and cai010 as examples. It was generally agreed that all his issues could benefit from examples.

There was a discussion re synchronizing the interpretations requests that appeared in the UK section of the addenda with those to appear in the interpretations bulletin to be published by X3J11. Perhaps they should contain the exact same set.

+++ ACTION: Plum will work with Gwyn to see if/how this can be made to happen by the Milan meeting.

4.3 MSE (revisited)

There were a number of concerns about when a stream becomes a `wide' stream and the mixing of char and widechar I/O on a given stream.

There was support for fixing the initial mode at the first I/O action rather than as a mode flag at open-time. Then make it undefined if mixed afterwards so implementations can give semantics if they can handle mixed modes.

Would it be useful to have a widechar equivalent of the ferror machinery? Probably not.

How can ftell be made to store shift state information for subsequent restoration with fseek? Plauger/Jaeschke argued that that's a Quality of Implementation issue and that ftell could simply fail. fgetpos and fsetpos clearly were the preferred functions here since they had to work across shift state changes. If ftell does succeed in such cases, however, then fseek must also restore the shift state as well as the file position.

What if you seek to eof? What is the shift state there? Currently the MSE says it's undefined although this has flip-flopped in recent versions of the proposal. Possibly, we could either not allow it, require the file to begin and end in the initial shift state, or require the implementation to write a (possibly redundant) start shift sequence there. Perhaps a member could be added to FILE to indicate the shift state to be used when reaching eof via a seek.

It was generally agreed that a seek to eof and subsequent append should work properly in that the appended data's beginning shift state be predictable.

8. Adjournment

The meeting was adjourned at 15:55.

Rex

Rex Jaeschke | Journal of C Language Translation | C Users Journal
(703) 860-0091 | 2051 Swans Neck Way | DEC PROFESSIONAL
rex@aussie.COM | Reston, Virginia 22091, USA | Programmers Journal

---

Convener of the Numerical C Extensions Group (NCEG)
X3J11 member and US International Representative to ISO C (WG14)
