

From: Douglas Walls
Document Number: WG14/N433 X3J11/95-034
Date: 1995-06-06
Subject: (SC22WG14.1364) Review of N412: C9X Draft 2

Overall I think Frank did a great job on N412, C9X Draft 2. Upon reviewing it, I believe the following corrections from TC1 have not been applied correctly. I don't know if any of these were amongst the "typos" from draft 1, having neither a list of such typos nor a copy of draft 1.

Page 20

Add to subclause 6.1.2, (Semantics):

When preprocessing tokens are converted to tokens during translation phase 7, if a preprocessing token could be converted to either a keyword or an identifier, it is converted to a keyword.

This TC was applied to the end of subclause 6.1.2, (Implementation limits). It should have been applied to the end of subclause 6.1.2, (Semantics), a few paragraphs above "Implementation limits".

Page 32

Add to subclause 6.1.7, (Semantics):

A header name preprocessing token is recognized only within a #include preprocessing directive.

This TC was applied as follows:

A header preprocessing token is recognized only within a #include preprocessing directive.

The third word of the sentence to be added was dropped, specifically the word "name".

Page 38

In subclause 6.3, lines 18-21, change:

An object shall have its stored value accessed only by an lvalue expression that has one of the following types: 36
-- the declared type of the object,
-- a qualified version of the declared type of the object,

to:

An object shall have its stored value accessed only by an lvalue expression that has one of the following types: 36
-- a type compatible with the declared type of the object,
-- a qualified version of a type compatible with the declared type of the object,

This TC was applied as follows:

An object shall have its stored value accessed only by an

- lvalue that has one of the following types: 36
- a type compatible with the declared type of the object,
- a qualified version of a type compatible with the declared type of the object,

Note that the 13th word of the lead-in was dropped, specifically the word "expression".

Page 54

Add to subclause 6.3.16.1, another Example:

In the fragment:

```
char c;  
int i;  
long l;
```

```
l = ( c = i );
```

...rest of TC not included here...

This TC was applied as follows:

2.

In the fragment:

```
char c;  
int i;  
long l;
```

```
l = ( c = i );
```

...rest of TC not included here...

The example number "2." is on a line by itself. For example 1 on the same page the example number is on the same line as the lead-in:

1. In the program fragment

This TC has been applied correctly, but would look a little better if the example number and lead-in were on the same line.

Page 63

Add to subclause 6.5.2.3, another Example:

An enumeration type is compatible with some integral type. An implementation may delay the choice of which integral type until all enumeration constants have been seen. Thus in:

```
enum f { c = sizeof(enum f) };
```

This TC was applied as follows:

Page 63

Add to subclause 6.5.2.3, another Example:

An enumeration type is compatible with some integral type. An implementation may delay the choice of which integral type until all enumeration constants have been seen. Thus in:


```
enum f { c = sizeof(enum f); };
```

^
^ This semicolon should not be here.

I also believe the following changes from Amendment 1 have not been applied correctly:

Minor typo while applying Page 10 of amendment 1, --Extensions to 7.9.6.1--, "Adjust the description of the qualifiers h, l, and L ...", was applied to page 136 of C9X draft 2 in part as follows on the 4th to last line of the description of the qualifiers h, l, and L:

following s conversion specifier applies to applies to a

Should be:

following s conversion specifier applies to a

Page 11 of Amendment 1, the example does not appear to have been applied correctly. On Page 139 of the C9X draft 2 in the example the second parameter of each of the fprintf calls is an incomplete string ending with:

|0

these should be complete strings ending with:

|\n"

Page 15 of Amendment 1, there is a discrepancy between what was applied for the e,E specifier and what page 194 of the C9X draft 2 shows for the e,E specifier. Specifically, the C9X draft 2 page 194, for the e,E specifier reads in part:

specifier[+]d.dde+dd

where page 15 of Amendment 1, for the e,E specifier reads in part:

specifier[-]d.dde+dd

I believe Amendment 1 is in error.

Minor typo while applying page 15 of Amendment 1 the specifier g, G, the C9X draft 2, page 195, for the specifier g,G reads in part "less than 1964 or greater than" this should read "less than -4 or greater than".

Page 195 of the C9X draft2, specifier s, second paragraph first line needs aligning with the rest of the paragraph.

Page 196 of the C9X draft 2, first sentence of the example, the symbol for pi has been changed to a p.

Page 196 of the C9X draft 2, the example, each of the fprintf calls (two occurrences) contain L" strings which end with a 0. They should end with \n"

Page 213 of the C9X draft 2, the example appears to have lost 3: \t characters, page 31 of Amendment 1 for this example has \t characters in part as follows:

```
static wchar_t str2[] = L"\t \t";  
  
t = wcstok(str2, L" \t", &ptr2); /* t is a null pointer */
```

Page 240 of C9X draft 2, Annex D, titles D.14 as:

Alternative <iso646.h>

page 39 of Amendment 1 Annex A, titles the equivalent, A.2 as:

Alternative spellings <iso646.h>

Page 266 of C9X draft2, Annex H, H.9.3. The diagram is missing.

Douglas Walls
SunSoft (A Sun Microsystems Business)
MS UMTV12-33
2550 Garcia Avenue
Mountain View, CA 94043-1100

E-mail: Douglas.Walls@Eng.Sun.COM
Phone: (415) 336-3454
Fax: (415) 964-0946

Member X3J11, ANSI C Committee

Member of the US delegation to ISO C committee SC22/WG14
