WG14 N1977
INCITS PL22.11-2015-00004
Date: 2015-10-26
Reply To The Attention Of: Barry Hedquist
PL22.11 Secretary
Email: beh@peren.com

MINUTES
13–17 April, 2015
MEETING OF ISO/IEC JTC 1 SC 22/WG 14 AND INCITS PL22.11

Dates and Times

13 April, 2015 09:00 – 12:00 Lunch 13:30 – 16:30
14 April, 2015 09:00 – 12:00 Lunch 13:30 – 16:00
15 April, 2015 09:00 – 12:00 Lunch 13:30 – 16:30
16 April, 2015 09:00 – 12:00 Lunch 13:30 – 16:30
17 April, 2015 09:00 – 12:00

Meeting Location

Cisco Systems
Philip Pedersens vei 1
1366 Lysaker, NORWAY
Phone: +47 67 125 125

Meeting information

N 1893

Local contact information

Olve Maudal (olve.maudal@cisco.com)
Phone: +47 90093309 (mobile)

Teleconference information

To be supplied with the final agenda.

1. Opening Activities

1.1 Opening Comments (Maudal, Keaton)

Olve Maudal welcomed us to Cisco Systems Norway. Lunch will be at 12:00. A table has been set aside for us in the cafeteria. Lunch is provided courtesy of Cisco. There is also a courtesy dinner available at 5:30 Monday and Wednesday.

1.2 Introduction of Participants/Roll Call

Name	***Organization***	***NB***	***Comments***
David Keaton	CERT/SEI/CMU	USA	WG14 Convener
John Parks	Intel	USA	PL22.11 Chair
Daniel Plakosh	CERT/SEI/CMU	USA	
Blaine Garst	Garst	USA	
Rajan Bhakta	IBM	Canada	
Clark Nelson	Intel	USA	
Barry Hedquist	Perennial	USA	Recording Secretary
Clive Pygott	LDRA	USA	
Douglas Walls	Oracle	USA	
Tom Plum	Plum Hall, Inc.	USA	
Fred Tydeman	Tydeman	USA	PL22.11 Vice Chair
Lars Bionnes	Cisco	USA	
Jens Gustedt	Cert	USA	
Olve Maudal	Cisco	USA	
Ismail Pazarbasi	Cisco	USA	

1.3 Procedures for this Meeting (Keaton)

The Meeting Chair and WG14 Convener, David Keaton, announced that procedures would be as per normal. Everyone was encouraged to participate in the discussion and straw polls.

Straw polls are an informal WG14 mechanism used to determine if there is consensus to pursue a particular technical approach or possibly drop a matter for lack of consensus. Straw polls are not formal votes, and do not in any way represent any National Body position. National Body positions are established in accordance with the procedures established by each National Body.

INCITS PL22.11 members reviewed the INCITS Anti-Trust and Patent Policy Guidelines at:

http://www.incits.org/standards-information/legal-info

All 'N' document numbers in these minutes refer to JTC1 SC22/WG14 documents unless otherwise noted.

The primary emphasis of this meeting was to review the progress of our subgroups and work on Defect Reports.

Barry Hedquist is the Recording Secretary for the meeting.

1.4 Approval of Previous Minutes [N1884]

Several typos from were reported by various members and corrected.

The minutes were approved by unanimous consent with those changes. (Garst/Parks)

Final Minutes from St Louis will be N1928.
Draft Minutes from Lysaker will be N1929.

1.5 Review of Action Items and Resolutions

ACTION: Convener to provide meeting info for Lysaker in post meeting mailing.
DONE

ACTION: Blaine to write up an approach for the FP Group that goes a bit beyond the approaches discussed in N1841.
OPEN

ACTION: Convener to communicate our intent include the issues discussed in N1848 into the 'future revisions' SD3 document to David Svoboda.
Done N1899

ACTION: Blaine to update TS 17961 DR Document to reflect the update to N1860.
DONE – N1891

ACTION: Blaine to reword the Proposed Committee Response for DR 441

DONE

ACTION: Convener to add the issues in DR 446 to the future revisions document SD3.
DONE N1917

ACTION: Blaine to write paper with proposed TC for DR 423
DONE – DR Log N1863

ACTION: Blaine to open a DR to clarify the memcmp reference in note 1 in 7.17.7.4; para 3.
DONE N1909

ACTION: Martin to contact Nick for further input for DR 437
DONE N1895

ACTION: Larry to review the words for a Proposed TC, DR 444.
OPEN

ACTION: Martin Sebor to show WG21 DR 445/Proposed TC.
OPEN – Follow up BEH

ACTION: Martin to examine what C++ says about temporary objects (object with a temporary lifetime). RE: DR 452
OPEN

ACTION: Blaine to update DR 452 as needed.
DONE – DR log

ACTION: Douglas to examine the issue and come up with a Proposed TC for DR 431.
DONE N1906

ACTION: Blaine to rework the Proposed TC for DR 453
DONE N1908

ACTION: Blaine to write up a Proposed Committee Response for DR 461, that also explains the intent of ‘refer’.
DONE - DR LOG

ACTION: David to add the suggested changes in DR 461 to SD 3.
DONE N1917

ACTION: Blaine to write a Proposed TC for DR 462 based on N1887, as discussed.
DONE – DR Log

ACTION: David to add suggested changes in DR 463 to SD 3
DONE N1917

ACTION: Blaine to write up a Proposed Committee Response to DR 466.
DONE – DR Log

ACTION: Convener to add the Suggested TC material in DR 466 (N1865) to SD3.
DONE N1917

ACTION: Blaine to add a Proposed Committee Response and Proposed Technical Corrigenda for DR 467.
DONE DR LOG

ACTION: Blaine to adopt Suggested TC as Proposed for DR 468.
DONE

ACTION: Blaine to submit a new paper for DR 469.
DONE N1907

ACTION: Blaine to write a Proposed TC for DR 470.
DONE N1922

ACTION: Blaine to write a Proposed TC for DR 471 based in the Suggested TC.
DONE DR Log

ACTION: Martin will provide words to clarify that a NULL pointer can be a valid argument as a consequence of the Proposed TC for DR 465.
OPEN

RESOLUTIONs:

Send out FP DTS 18661, Parts 3 & 4, for DTS Ballot after editorial review by the Editorial Committee.
DONE – N1896, N1897

1.6 Approval of Agenda

Revisions to the Agenda are posted on the Wiki, and reflected here.
Added Items:

N1922
N1923
N1927

Deleted Items: None

Agenda approved by unanimous consent. (Tydeman/ Bhakta)

1.7 Identify National Bodies Sending Experts

US, Canada

2. Reports on Liaison Activities

2.1 SC 22 (Plum) - No Report

2.2 PL22.11/WG 14 (Parks / Keaton)
PL22.11 – no report
WG14 – SD 1-3 updated N1925 – N1917

2.3 PL22.16/WG 21 (Keaton)
WG21 N4220 – Eliminating Undefined Behavior. Should keep an eye on it.
ACTION – ALL Review & Comment WG21 N4220

Tom Plum: Email 13649: regarding __BOOL_ … subject to more than one interpretation, better if C++ dropped reference to it. WG21 wants input from WG14. We agree that C++ should drop it.

2.4 PL22 (Plum) No Report. Jim Thomas is receiving a Technical Achievement award at the symposium in Las Vegas for his cobtributionb to C Floating Point

2.5 WG 23 (Pygott) WG23 Convener is Stephan Michell.

2.6 MISRA C (Pygott) No Report

2.7 Other Liaison Activities
Austin Group – Jens. POSIX Issue with snprintf. What happens if the size is larger than INT_MAX. POSIX returns an error. Should C do the same? Austin Group could submit a DR to WG14. We cannot make a decision on this without an N paper describing the issues.

3. Reports from Study Groups

3.1 C Floating Point activity report (Rajan)

Part 2 Major editorial issues with ISO. Personnel changeover issues created an incorrect document published in February. WG14 has sent what is to be believed the 'final' version. N1924 (final version) and N1926 (diff marks) represent the final version.

Parts 3 & 4 Recently passed last ballot. Possible editorial items.

Part 5 Working Document is published, working on exception handling part, and is not in the existing working document

3.2 CPLEX activity report (Nelson)

Not a lot of progress since the last WG14 meeting. Do not have a new working paper. Hoping to have a more complete document in October.

4. Teleconference Meeting Reports

4.1 Report on any teleconference meetings held – No new reports

5. Future Meetings

5.1 Future Meeting Schedule

- Fall, 2015 – Kona, HI, USA, 26–29 October, 2015 (following WG21)
- Spring, 2016 – London, UK, 11–15 April, 2016 – BSI
- Fall, 2016 – Pittsburgh, PA, USA (tentative) CERT
- Spring, 2017 – Markham, ON, Canada (tentative) IBM

5.2 Future Mailings

- Post Lysaker – 04-May-2015
- Pre Kona – 28-September-2015
- Post Kona – 30-November-2015
- Pre London – 14-March-2016
- Post London – 09-May-2016

6. Document Review

6.1 N1896, TS 18661-3

This TS is for floating point extensions to C for Interchange and extended types.

TS Ballot passed with no comments. Some editorial changes required by ISO. Need an editorial review committee.

Fred, Clive, Blain, David volunteered for the editorial review committee.

Then, send out for publication

6.2 N1897, TS 18661-4

TS is for floating point extensions to C for supplementary functions.

TS Ballot has passed with no comments. Some editorial changes required by ISO, Need an editorial review committee.

Fred, Clive, Blain, David volunteered for the editorial review committee.

Then, send out for publication

6.3 N1899, Integer Precision Bits Update, Svoboda

This document is a proposed revision to the C Standard to add macros that indicate the number or width bits for the standard unsigned integer types. For unsigned integer types, the number precision bits matches the number of width bits. Precision bits for the signed integer types can be derived by subtracting 1 from the width bits for the corresponding unsigned integer type.

Add this paper to SD3, Proposed Revisions to the Standard.

Discussion:

7.20.2.1 Rajan: Why <=N rather than =N for UINTN_MAX? May also apply to INTN_MAX as well.

Any objection to adding N1899 to SD3 with a note concerning the <= issue of 7.20.2.1. – NO.

ACTION: Convener to add N1899 to SD3 with a note concerning the <= issue of 7.20.2.1

6.4 N1919, First Draft of TS 18661-5 (incomplete but ready to start discussions) (Bhakta)

Part 5 of TS 18661, Floating-point extensions for C, define supplemental attributes recommended by ISO/IEC/IEEE 60559:2011.

Will also include some discussion of N1925, which includes exception handling items.

N1919 is the first draft of Part 5 presented to WG14. Rajan presented as slide presentation on an overview of Part 5. Jens: Any reason some pragmas are not preceded by __STDC_ ? Rajan: Probably an oversight.

Rajan also covered material in N1925, Alternate Exception Handling. The intent is to be able to conform to all the alternate methods presented or the implementation is non-conforming to Part 5. Doing so will likely be quite expensive to implementations. Clark suggested putting the action first in the syntax statement. Rajan agreed.

What happens when an exception action follows a prior action, and one action conflicts with the other – sequencing of actions issue. Spec should leave room for an implementation to define what happens. Nothing is said about what happens with signals.

Schedule: TBD. Expect to have a draft at Kona. David pointed out that the ISO completion date for all of 18661 is Dec 2016. Could be an issue.

Discussion of N1919 lead by Rajan. Fred took notes on this discussion for consideration by the FP Committee for revision to this draft.

Straw Poll: Do volatile and atomics retain their existing sematics with the value changing optimizations?

YES – 12, NO – 0, Abstain – 3

How to handle an IEEE requirement in the IEEE Standard that non-reproducible affects be diagnosed when detecting such as case is not possible.

6.5 N1922, Proposal for DR 470

Moved to DRs

6.6 N1923, Compatibility of Pointers to Arrays with Qualifiers,

Moved from DRs to consideration for SD 3. There does not seem to be a reason to NOT make this change. This would put us in sync with C++. Tom would like to see comparison of pointers included with this item as an add to SD 3. That would require a new paper.

Add to SD 3? YES

ACTION: Convener to add the material in N1923 to the SD3.

6.7 N1927, Possible DR: Misleading Atomic Library References to atomic types (Garst).

Blain sees this as something we should have caught in editing C11, but did not. Is this a DR? Jens thinks it is, because in some place it applies, it some places it does not. This would clear up our 'intent'.

Make this a DR? YES, assigned DR 475.

6.8 N1910, _Alignof Incomplete Arrays

Difference between the C and C++ Standards was created in C++11. C++ considered this item to be 'not the way it should work', and changed it. We did not catch it. Is this a DR or a 'change to our Standard' ? General consensus is it is a change to our Standard. Discussion to add to SD 3.

There does not seem to be a reason why an incomplete type is NOT allowed. C++ allows it, but we are not sure why. An object of an incomplete array type cannot be allocated.

Add to SD 3? Yes – 11, No – 1, Abs – 1

ACTION: Convener to add N1910 to SD 3.

6.9 **__LINE__ and Multiple Lines [N1911]**

It's not clear what value, if any, is added by addressing this as a DR. There is no use for this change.

> Is this a DR. Yes – 4, No 8, Abstain -2
> NOT A DR.
> Put under Document Review.

The potential of a "need" for a change is very unclear. Making the behavior 'unspecified' seems to have some merit.

> Add N1911 to SD 3 as explicit "unspecified behavior" ? y-8, n-5, a-2.

ACTION: Convener to add N1911 as "unspecified" behavior rather than implicit "undefined".

7. Defect Reports

7.1 Discussion on the Defect Report Process

7.2 IS 9899:2011 Defect Reports N1892

DRs in REVIEW

DR 438 – moved to CLOSED

DR 440 – moved to CLOSED

DR 442 – moved to CLOSED

DR 443 – moved to CLOSED

DR 449 – moved to CLOSED

DR 451 – moved to CLOSED

DR 454 – moved to CLOSED

DR 457 – moved to CLOSED

DR 458 – moved to CLOSED

DR 459 – moved to CLOSED

DR 460 – moved to CLOSED

DR 463 – moved to CLOSED

DRs in OPEN Status

DR 406
October 2014 meeting reviewed N1856, and developed a Committee Discussion with proposed words. Are the words sufficient? They represent progress, but possibly not perfection. What are “coherence rules”? (5.1.2.4 p27, last line). DR402 introduces the “read-write coherence” term. Change to “coherence requirements v. coherence rules” ??(yes) .

Item 1. 5.1.2.4;p22 Change ‘value from’ to ‘from which its values are taken’.

Generate a Proposed TC based on above.
Leave OPEN

DR 407
Turn item #2 from Oct 2014 Committee Discussion into a Proposed TC.
Leave OPEN

DR 423
Jens says there are compilers now that interpret generic differently. CLANG and GCC are different. This item is more critical now that previously. We need to clarify our intent. People are starting to use 'generic, and implementations differ. See N1863. Adopt those words as a Proposed TC for DR 423.
Leave OPEN

DR 427
Clark – the Proposed TC is not enough. We must not lose implicit conversion of argument expressions. Clark will look more into this.
Come back Wed.
DONE
Clark is satisfied we've said what we mean. There is no room for confusion.
Everyone agrees.
Add Committee Discussion.
Blaine would rather use the verb 'initialization' rather than 'assignment'.

ACTION: Blain to investigate other concerns for DR 427.

Leave OPEN

DR 431
Subject: atomic_compare_exchange: What does it mean to say two structs compare equal?

atomic_compare_exchange is intended to emulate low level semantics, rather than high level instructions. You get what the machine gives you.

Proposed Technical Corrigendum (N1906)

> 7.17.7.4p2 replace the second sentence with:
>
> Atomically, compares the contents of the memory pointed to by object for equality with that in expected, and if true, replaces the contents of the memory pointed to by object with desired, and if false, updates the value in expected with the value pointed to by object.

We meant to be aligned with C++, and for this, we are not.
Make the above a Proposed TC for this DR? Needs words along the line of… TBD.

Question: Use words along the lines of above to create a Proposed TC: y-11, n-2, abs -2

ACTION: Douglas to write a Proposed TC for DR 431 based on N1906 and editorial add-ons suggested by Fred and Jens. Doug to email the Proposed TC for inclusion in these minutes. DONE – see below:

Words from Douglas: Proposed TC

7.17.7.4p2 replace the second sentence with:

Atomically, compares the contents of the memory pointed to by object for bitwise equality with that in expected, and if true, replaces the contents of the memory pointed to by object with desired, and if false, updates the value in expected with the value pointed to by object.

Leave OPEN

DR 437
Subject: clock overflow problems

ACTION: Rajan to write a Proposed TC.
DONE – See below

Committee discussion direction:
Question 1: No. Yes. Remove the last line of 7.27.2.1p3 (see question 3 for the change).
Question 2: No.
Question 3: No. Say does not result in undefined behaviour and instead is unspecified behaviour.

Proposed TC:
In 7.27.2.1p3 change:
If the processor time used is not available or its value cannot be represented, the function
returns the value (clock_t)(-1). (319)
...
319) In order to measure the time spent in a program, the clock function should be called at the start of
the program and its return value subtracted from the value returned by subsequent calls.

to:

If the processor time used is not available, the function returns the value (clock_t)(-1).

If the value cannot be represented, the function returns an unspecified value. (319)

...

319) This may be due to overflow of the clock_t type.

Some concern about removing the original footnote, but that footnote is the source of this DR.

Q: Do we want to accept Rajan's words above as a Proposed TC y-11, N-0, A-4

ACTION: Blain to add Rajan's words above as a Proposed TC for DR 437
Leave OPEN

DR 439
Subject: Issues with the definition of "full expression"

ACTION: Clark to write a paper on DR 439.

DR 441
Subject: Floating-point issues in C11 from PDTS 18661-1 UK review, Issue 2

NAD, Proposed Committee Response? Fred disagrees.

ACTION: Blain to write a Proposed TC for DR 441 applicable to the suggested change to Annex F. Change "might not" to "need not".

Leave OPEN

DR 444
Subject: Issues with alignment in C11, part 1
No Proposed TC

Clark pointed out that C++ does allow the changes proposed in this DR. Should _Alignas be allowed in a cast? After some discussion, Clark volunteered to write a Proposed TC, without using it in a cast.

ACTION: Clark to write a Proposed TC for DR 444.
DONE – See Below:

6.7.2.1p1: Add a new alternative to specifier-qualifier-list:

specifier-qualifier-list:
type-specifier specifier-qualifier-list$_{opt}$
type-qualifier specifier-qualifier-list$_{opt}$
alignment-specifier specifier-qualifier-list$_{opt}$

specifier-qualifier-list is used in the grammar in only two productions: *struct-declaration* (which relates to the primary purpose of this DR), and *type-name*, which is used only in the definitions of these constructs:

- generic association (generic selection)
- compound literal
- sizeof expression
-_Alignof expression
- cast expression
- atomic type specifier
- alignment specifier

As far as I can tell, the C++ standard does not allow an alignment-specifier in a type-id (which is the C++ name for the construct that corresponds to type-name). However, that prohibition is imposed by a blanket statement about attributes, which can't be used for C.

If we think it doesn't make sense to support an alignment-specifier in any of the above constructs, that could be achieved quite simply by adding a constraint (the first!) to 6.7.7, following paragraph 1 (the syntax): (an optional addition).

> Constraints
>
> The specifier-qualifier list of a type name shall not include an alignment specifier.

Issues noted during drafting (Clark)

In 6.7.3p5, there are two references to specifier-qualifier-list, which should also reference declaration specifiers (and which, for consistency with the rest of the prose, should not use an italicized nonterminal name).

In 6.7.5, paragraphs 2 and 4, there are occurrences of the phrase “alignment attribute” which should instead read “alignment specifier”.

Discussion:

Does the optional constraint serve any real purpose? Fold Clark's input into the Committee Discussion.

Tom's thoughts:
Alignas needs to be applied wherever objects are laid out in memory
struct type will involve the alignment into the type system
other than struct, alignment is not incorporated into the type system

ACTION: Blain will look for a number of constraints that capture Tom's thoughts on DR 444 and incorporate it into the Committee Discussion.

Alignment of arrays, and scalars are two possible instances outside of structs that may be considered.

Leave OPEN

DR 445
Subject: Issues with alignment in C11, part 2

Proposed TC exists. C++ liaison issue?
Leave OPEN – come back later
Dave likes the Proposed TC.
Move to REVIEW

DR 448
Subject: What are the semantics of a # non-directive?
Proposed TC exists.
Moved to REVIEW

DR 450
Proposed TC exists.
Moved to REVIEW

DR 452
Subject: Effective Type in Loop Invariant
How to define the effective type of a temporary object?
Discussion from Oct 2014 has a possible TC. Is it complete? Clark does not think so, and does not like '..behaves as if..' Change to '..behaves as if it were declared with the type of it's value..' or just say we want it to have the effective type.

A temporary object has the effective type of its value. ??

A temporary object has the effective type of …(what). Need to define 'what'.

Rajan will try to define ‘what’ at lunch.

Rajan’s Proposed TC

> In 6.2.4p8 append:
> An object with temporary lifetime behaves as if it were declared with the ~~declared~~ type of its value for the purposes of *effective type* (forward reference to 6.5p6). Such an object is known as a temporary object. A temporary object need not have a unique address. (wordsmithed)
>
> Needs work.
>
> Consider not defining temporary object ?

Douglas rework:

> An object with temporary lifetime behaves as if it were declared with the type of its value for the purposes of effective type (forward reference to 6.5p6). An object with temporary lifetime need not have a unique address.

Blain would like to replace ‘an object with a temporary lifetime’ with ‘such an object.’

ACTION: Blaine Generate a Proposed Technical Corrigenda for DR 452 based on the Committee Discussion of April 2015.

DR 453
Subject: Atomic flag type and operations (7.17.8)
See N1908

Fred: Nothing ties the use of ‘atomically’ in para 2 & 3 to the same atomic operation.

ACTION: Dave to write something up during lunch.
DONE – See Below

The following is proposed as a technical corrigendum for DR 453 on the atomic flag data type.

Proposed Technical Corrigendum

Add to the end of 7.17.8p1:

The value true corresponds to the set state and the value false corresponds to the clear state.
In 7.17.8.1p1, change:
Atomically sets the value pointed to by **`object`** to true.
to:
Atomically places the atomic flag pointed to by **`object`** in the set state and returns the value corresponding to the immediately preceding state.

In 7.17.8.1p2, change:
Atomically, the value of the object immediately before the effects.
to:
The **`atomic_flag_test_and_set`** functions return the value that corresponds to the state of the atomic flag immediately before the effects.

In 7.17.8.2p2, change:
Atomically sets the value pointed to by **`object`** to false.
to:
Atomically places the atomic flag pointed to by **`object`** into the clear state.

Discussion:
Rajan likes this.
Add as a Proposed TC to DR 453 ? YES
Paragraph 1 & 2 should be Paragraph 2 & 3. Some other ed changes as well.

ACTION: DR 453: David to rewrite his proposed TC as needed, forward this to Blaine, to include in the Committee Discussion.

DR 455
Proposed TC exists
Moved to REVIEW

DR 456
Proposed TC Exists
Moved to REVIEW

DR 461
Proposed Committee Response exists
Added to SD 3
Moved to REVIEW

DR 462
Proposed Committee Response exists
David to rewrite for review later today.

The following is proposed as a technical corrigendum for DR 462 on atomic operations in signal handlers. Differences from the previously proposed technical corrigendum are highlighted.

Proposed Technical Corrigendum

Change subclause 7.14.1.1 paragraph 5 from:

> If the signal occurs other than as the result of calling the **abort** or **raise** function, the behavior is undefined if the signal handler refers to any object with static or thread storage duration that is not a lock-free atomic object other than by assigning a value to an object declared as**volatile sig_atomic_t**, or the signal handler calls any function in the standard library other than the **abort** function, the **_Exit** function, the **quick_exit** function, or the **signal** function with the first argument equal to the signal number corresponding to the signal that caused the invocation of the handler. Furthermore, if such a call to the **signal** function results in a **SIG_ERR** return, the value of **errno** is indeterminate.[252)]

to:

> If the signal occurs other than as the result of calling the **abort** or **raise** function, the behavior is undefined if the signal handler refers to any object with static or thread storage duration that is not a lock-free atomic object other than by assigning a value to an object declared as**volatile sig_atomic_t**, or the signal handler calls any function in the standard library other than
>
> - the **abort** function,
> - the **_Exit** function,
> - the **quick_exit** function,
> - the functions in **<stdatomic.h>** (except where explicitly stated otherwise) when the atomic arguments are lock-free,
> - the **atomic_is_lock_free** function with any atomic argument, or
> - the **signal** function with the first argument equal to the signal number corresponding to the signal that caused the invocation of the handler. Furthermore, if such a call to the **signal** function results in a **SIG_ERR** return, the value of **errno** is indeterminate.[252)]

Add a new paragraph after 7.17.2.2 paragraph 3:

> If a signal occurs other than as the result of calling the **abort** or **raise** function, the behavior is undefined if the signal handler calls the **atomic_init** generic function.

In subclause J.2 Undefined behavior, change:

> A signal occurs other than as the result of calling the **abort** or **raise** function, and the signal handler refers to an object with static or thread storage duration that is not a lock-free atomic object other than by assigning a value to an object declared as **volatile sig_atomic_t**, or calls any function in the standard library other than the **abort** function, the **_Exit** function, the **quick_exit** function, or the **signal** function (for the same signal number) (7.14.1.1).

to:

> A signal occurs other than as the result of calling the **abort** or **raise** function, and the signal handler refers to an object with static or thread storage duration that is not a lock-free atomic object other than by assigning a value to an object declared as **volatile sig_atomic_t**, or calls any function in the standard library other than the **abort** function, the **_Exit** function, the **quick_exit** function, the functions in **<stdatomic.h>** (except where explicitly stated otherwise) when the atomic arguments are lock-free, the **atomic_is_lock_free** function with any atomic argument, or the **signal** function (for the same signal number) (7.14.1.1).

In subclause J.2 Undefined behavior, insert in order:

> A signal occurs other than as the result of calling the **abort** or **raise** function, and the signal handler calls the **atomic_init** generic function (7.17.2.2).

Discussion:
Do we want to make the above the new proposed TC for DR 462. YES

ACTION: Blaine to take the new words from David on DR 462 as a Proposed TC.

Leave OPEN

DR 464
Proposed TC and CR exist
Moved to REVIEW

DR 465
Proposed TC exists
Need to modify the second change? No
Blain sees a possible Committee Response with this as well. Fredd suggested an explanatory footnote.

Moved to REVIEW

DR 466
This has been added to SD3 – Not a Defect
Propose Committee Response exists.
Moved to REVIEW

DR 467
Proposed TC exists
Moved to REVIEW
ACTION: David to add macro portion of DR 467 to SD 3

DR 468
Proposed TC exists
Moved to REVIEW

DR 469
Committee Discussion, Oct 2014
See N1907

Issue 1 is moot. Done in DR 414
Issue 2: An unlock succeeds only if all prior locks have been successfully unlocked.
More work needed.
Issue 3: Why say this? We already know it's undefined. The words differ from C++
Q: Specify explicit Undefined Behavior? Y – 8, N – 2, A – 5, - So state.

Leave OPEN

DR 470
Committee Discussion Oct 2014
See N1922

In N1882 it is pointed out C11 does not clearly say that mtx_trylock can spuriously fail, and that although this can be inferred, the wording is far less clear than the corresponding section 30.4.1.2 in the C++11 standard.

Proposed Technical Corrigendum

In 7.26.4.5 The mtx_trylock function replace paragraph 3

The mtx_trylock function returns thrd_success on success, or thrd_busy if the resource requested is already in use, or thrd_error if the request could not be honored.

with

The mtx_trylock function returns thrd_success on success, or thrd_busy if the resource requested is already in use, or thrd_error if the request could not be honored. **In some implementations** mtx_trylock may spuriously fail to lock an unused resource, in which case it shall return thrd_busy.

Adopt the above words as the Proposed TC for DR 470 ? w/o the words “in some implementations”.

Leave OPEN

DR 471
Proposed TC exists from Oct 2014.
Moved to REVIEW

DR 472 (N1902)

Rajan submittal of Suggested TC for DR 472
Suggested TC for N1902:

In 7.3.1#3, change:
Each synopsis specifies a family of functions

to:

Each synopsis other than the CMPLX macros specifies a family of functions

and add a forward reference to 7.3.9.3 after the paragraph.

Adopt the words above as a Proposed TC for DR 472.

DR 473 (N1903)
Q: Accept Fred's proposed TC 8-3-4
Leave OPEN

DR 474 (N1909)
Clarification for atomic_compare_exchange

Suggested TC (Blain N1909)
In 7.17.7.4 The atomic_compare_exchange generic functions paragraph 3 replace:

> NOTE 1 For example, the effect of
> **atomic_compare_exchange_strong** is
>
> with
>
> NOTE 1 For example, the effect of **atomic_compare_exchange_strong** is, for unpadded lock-free integer types, atomically

This proposed note differs from what is contained in C++, which matches what we presently have. Changing our existing note to differ from C++ could cause some level of confusion, even if it is correct. No consensus to make this change.

ACTION: Blain to write a Committee Response for DR 474

Leave OPEN

DR 475 (N1927)

Discussion lead to a number of changes to Blaine's initial paper.

7.2.1 Deferred from Previous Meeting: Proposed Resolution of DR 423 [N1863]

7.2.2 Proposed Technical Corrigendum for DR 448 [N1885]

7.2.3 Possible Defect Report: Complex Math Functions cacosh and ctanh [N1886]

7.2.4 Discussion of DR 452 [N1888]

7.2.5 Updated Technical Corrigendum for DR 437 [N1895]

7.2.6 Possible Defect Report: CMPLX and 7.3.1 Introduction [N1902]
Yes – Assigned DR 472

7.2.7 Possible Defect Report: *Too Large* and expm1, erfc, lgamma [N1903]
Yes – Assigned DR 473

7.2.8 Propose Technical Corrigendum for DR 431 [N1906]

7.2.9 Proposal for DR 469 [N1907]

7.2.10 Proposal for DR 453 [N1908]

7.2.11 Note 1 Clarification for atomic_compare_exchange [N1909]

Yes – Assigned DR 474, related to DR 431
"Proposed TC should be "Suggested TC"

7.2.12 Possible Defect Report: _Alignof Incomplete Arrays [N1910]

Difference between the C and C++ Standards was created in C++11. C++ considered this item to be 'not the way it should work', and changed it. We did not catch it. Is this a DR or a 'change to our Standard' ? General consensus is it is a change to our Standard.

Consider adding this item to SD 3 as a future consideration of a revision. Moved to Document Review.

ACTION: Convener to add N1910 to the SD 3.

7.2.13 Related to DR 464: __LINE__ and Multiple Lines [N1911]

It's not clear what value, if any, is added by addressing this as a DR. There is no use for this change.
Is this a DR. Yes – 4, No 8, Abstain -2
NOT A DR. (NAD)
Put under Document Review.

7.2.14 DR 444

We would like compiler developers to tell us if the Committee Discussion presented in this DR has been implemented. We had asked Larry Jones to review this language to see if it reflected what we want. Who can work on this? Clark? Yes. Rajan – Possible.

Revisit on Wednesday.

7.3 TS 17961:2013 Defect Reports [N1891]
Subject: error in 5.21 example
Proposed TC exists.
Moved to REVIEW.

8. Other Business

8.1 DR 422 – Not a Defect: Fred would like this included in SD 3. David believes this calls for a paper. Jens agreed to write one that will likely include DR 421 as well.

8.2 DR 426 – Committee Discussion is wrong, third bullet. 'neither multiply or divide'.. should be inserted. editorial correction. No effect on Proposed TC.

8.3 Controlling Expression of _Generic primary expression (Gustedt)
Follow-up to DR 423 (CLOSED)
Does the controlling expression of a _Generic primary expression undergo any type of conversion to calculate the type that is used to do the selection? Implementations give different answers to that question. Clark: We explicitly rejected the concept of handling 'const char' as well as 'char'. This paper is presented as a potential DR. If so, we should clarify what our intention was. N1441 was an earlier version. Did we talk about this in Santa Cruz? Several implementations have followed the clang model. Blaine would like to see conflicting cases between clang and gcc. A clearer statement of the problem is needed.

ACTION: Jens to convert his paper on "Controlling Expression of _Generic primary expression" into an 'N' paper for the post-meeting mailing.

9. Resolutions and Decisions Reached

9.1 Review of Decisions Reached

Send DTS 18661 parts 3 & 4 to ISO ITTF for publication.

9.2 Review of Action Items

CARRY-OVER ACTION ITEMS

ACTION: Blaine to write up an approach for the Floating Point Group that goes a bit beyond the approaches discussed in N1841.

ACTION: Larry to review the words for a Proposed TC, DR 444. (Done-OBE)

ACTION: Martin to examine what C++ says about temporary objects (object with a temporary lifetime). RE: DR 452.

ACTION: Martin will provide words to clarify that a NULL pointer can be a valid argument as a consequence of the Proposed TC for DR 465.

NEW ACTION ITEMS

ACTION – Tom Plum to write a paper explaining C++ qualification conversions.

ACTION – Tom Plum to write a paper describing C++ pointer comparison.

ACTION – ALL Review & Comment WG21 N4220, Preprocessor Undefined Behavior paper.

ACTION: Convener to add N1899 to SD 3 with a note concerning the **<=** issue of 7.20.2.1

ACTION: Convener to add N1923 to SD 3

ACTION: Convener to add N1910 to SD 3

ACTION: Convener to add N1911 as “unspecified” behavior rather than implicit “undefined” to SD 3.

ACTION: DR 427: Blaine to investigate other concerns.

ACTION: DR 437: Blaine to add Rajan’s words in Committee Discussion as a Proposed TC.

ACTION: DR 439: Clark to write a paper.

ACTION: DR 444: Blaine will look for a number of constraints that capture Tom’s thoughts and incorporate them into the Committee Discussion.

ACTION: DR 452: Blaine Generate a Proposed Technical Corrigenda based on the Committee Discussion of April 2015.

ACTION: DR 467: Convener to add macro portion to SD 3

ACTION: DR 462: Blaine to take the new words from David as a Proposed TC.

ACTION: DR 453: David to rewrite his proposed TC as needed, forward this to Blaine, to include in the Committee Discussion.

ACTION: Jens to convert his paper on "Controlling Expression of _Generic primary expression" into an 'N' paper for the post-meeting mailing.

10. Thanks to Host

Thanks to Cisco for hosting the meeting.

11. Adjournment

The meeting was adjourned at 1135 hours, April 16, 2015 (Garst/Hedquist)

Minutes for the PL22.11/US TAG Meeting, Tuesday, April 14, 2015 at 16:00 local

Name	***Organization***	***Primary/Alternate***	***Comments***
David Keaton	CERT/SEI/CMU	P	
John Parks	Intel	P	PL22.11 Chair
Daniel Plakosh	CERT/SEI/CMU	P	
Blaine Garst	Garst	P	
Rajan Bhakta	IBM	P	
Clark Nelson	Intel	P	
Barry Hedquist	Perennial	P	PL22.16 Secretary
Clive Pygott	LDRA	P	
Douglas Walls	Oracle	P	
Tom Plum	Plum Hall, Inc.	P	
Fred Tydeman	Tydeman	P	PL22.11 Vice Chair
Lars Bionnes	Cisco	P	
Jens Gustedt	Cert	A	

Olve Maudal	Cisco	A	
Ismail Pazarbasi	Cisco	A	

1. Approval of Agenda

   The agenda was approved by unanimous consent (Keaton/Tydeman).

2. Approval of Previous Minutes

   The prior meeting minutes for St Louis, Oct 28, 2014, were approved by unanimous consent (Garst/Keaton))

3. INCITS Antitrust Guidelines and Patent Policy

4. INCITS official designated member/alternate information

5. Identification of PL22.11 Voting Members
   1. PL22.11 Members Attaining Voting Rights at this Meeting

      none

   2. Prospective PL22.11 Members Attending their First Meeting

      none

6. Members in Jeopardy
   1. Members in jeopardy due to failure to return Letter Ballots

      none

   2. Members in jeopardy due to failure to attend Meetings

      2.1 Members in jeopardy for failure to attend this meeting.

      none

2.2 Members who retained voting rights by attending this meeting

none

2.3 Members who lost voting rights for failure to attend this meeting

none

3. Members who previously lost voting rights who are attending this meeting

   none

7. Procedures for Forming a US Position

   per normal

8. New Business

   1. National Maintenance Review, INCITS/ISO/IEC 24747:2009[2010], Mathematical Special Functions.

   The National Maintenance Review is for the INCITS version of ISO/IEC 24747:2009, Mathematical Special Functions. The ISO/IEC Standard has already REAFFIRMED in an International Ballot on.

   QUESTION: Do you APPROVE of the REAFFIRMATION of INCITS/ISO/IEC 24747:2009[2010] ?

   CERT yes
   Cisco yes
   Plum Hall yes
   Tydeman yes
   IBM abstain
   Perennial yes
   Cisco yes
   Garst yes
   Oracle abstain
   LDRA yes

   APPROVED: 8-0-2

9. Next Meeting

The next meeting of the US TAG for SC22/WG14 will be Tuesday, 27 October, 2015, in Kona, HI.

10. Adjournment

Meeting adjourned by unanimous consent at 16:30 hours, April 14, 2015.

Hedquist/Tydeman