

WG14 N1883
INCITS PL22.11/14-001
Date: 2014-10-27
Reply To The Attention Of: Barry Hedquist
PL22.11 Secretary
Email: beh@peren.com

MINUTES
April 7 - 9, 2014
MEETING OF ISO/IEC JTC 1 SC 22/WG 14 AND INCITS PL22.11

Meeting Location

*Parco Area delle Scienze 53/A
I-43124 Parma
Italy*

Meeting Information

[N1788](#)

Local Contact Information

*Roberto Bagnara (bagnara@cs.unipr.it)
Phone: +39 0521 906917
Mobile: +39 339 8593517*

*Department of Mathematics and Computer Science:
Phone: +39 0521 906900
Fax: +39 0521 906950*

Scheduled Meeting Times

07 April 2014 09:00 – 12:00 Lunch 13:30 – 16:30
08 April 2014 09:00 – 12:00 Lunch 13:30 – 16:00
09 April 2014 09:00 – 12:00 Lunch 13:30 – 16:30
10 April 2014 09:00 – 12:00 Lunch 13:30 – 16:30
11 April 2014 09:00 – 12:00

1. Opening Activities

1.1 Opening Comments (Benito, Bagnara)

John Benito and Roberto Bagnara welcomed us to Italy and the University of Parma and described the meeting facilities. The meeting was hosted by the University of Parma.

1.2 Introduction of Participants/Roll Call

<u>Name</u>	<u>Organization</u>	<u>NB</u>	<u>Comments</u>
John Benito	Plum Hall, Inc.	USA	WG14 Convener
Jim Thomas	Tydeman Consulting	USA	
David Keaton	CERT/SEI/CMU	USA	PL22.11 Chair
Daniel Plakosh	CERT/SEI/CMU	USA	
Tana Plauger	Dinkumware, Ltd	USA	
P. J. Plauger	Dinkumware, Ltd	USA	
Blaine Garst	Garst	USA	
Rajan Bhakta	IBM	Canada	HoD - Canada
Clark Nelson	Intel	USA	
John Parks	Intel	USA	Acting Secretary
Clive Pygott	LDRA	USA	
Douglas Walls	Oracle	USA	HoD - USA
Tom Plum	Plum Hall, Inc.	USA	
Martin Sebor	Cisco	USA	
Freek Wiedijk	Radboud Univ. Nijmegen	Netherlands	
Willem Wakker	ACE	Netherlands	HoD - Netherlands
Roberto Bagnara	Univ. of Parma	Italy	HoD - Italy
Larry Jones	Siemens PLM Software		WG14 Project Editor

1.3 Procedures for this Meeting (Benito)

The Meeting Chair and WG14 Convener, John Benito, 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, consider proposals for new work, and work on Defect Reports.

John Parks was the Recording Secretary for the meeting (in the absence of PL22.11 Secretary Barry Hedquist).

1.4 Approval of Previous Minutes (N1764)

Several typos from Hedquist were reported. Benito will send those to JP.

The minutes were approved by unanimous consent with those changes.

Final Minutes from Chicago will be N1819.

Draft Minutes from Parma will be N1820.

1.5 Review of Action Items and Resolutions (Parks)

ACTION: Small editing group: Jim Thomas, Fred Tydeman, David Keaton, John Benito, to review [N1756](#), and modify as needed, for incorporation of agreed to responses to NB comments.

CLOSED

ACTION: Convener to forward [N1756](#) to 30 day review by WG14. Modify the document as needed. Then submit to SC22 for DTS ballot.

CLOSED - document passed DTS ballot

ACTION: Blaine Garst to write up a new Suggested TC for [DR 423](#).

CLOSED

ACTION: Larry Jones to reword Suggested TC footnote for [DR 438](#).

OPEN (CLOSED later in the meeting with [N1823](#))

ACTION: Clark Nelson to write a Suggested TC for [DR 439](#), Item C.

OPEN

ACTION: Blaine Garst to write a Suggested Committee Response for [DR 449](#).

OPEN

ACTION: Blaine Garst to contact the submitter of [N1762/DR 452](#) to prepare to answer his questions

CLOSED – Blaine did contact, still waiting for response

1.6 Approval of Agenda (N1800)

Revisions to Agenda: posted on the Wiki

Added Items: None

Deleted Items: None

Agenda approved by unanimous consent.

1.7 Identify National Body Delegations

US, Canada, Netherlands, Italy

2. Reports on Liaison Activities

2.1 SC 22 (Plum)

Will meet in Madrid in September 2014.

Will meet in Hawaii in September 2015 hosted by Plum Hall.

2.2 PL22.11/WG 14 (Keaton)

This was John Benito's last meeting as WG14 Convener. He will not serve another term. David Keaton has graciously volunteered to replace him.

2.3 PL22.16/WG 21 (Plum)

The ballot is now closed for C++14 and the committee is responding to comments. The next meeting is in Switzerland in a couple months and will conclude the disposition of comments. The C++14 standard is expected to be published in 2014.

WG21 seems to have adopted a “major release followed by minor release” pattern on a 3 year cycle. C++14 is considered a minor release. C++17 will be a major release. The definition of “major/minor” is subjective but people have relatively clear ideas about it. Some features that were considered too “major” were deferred to C++17.

Tana pointed out that WG21 is also working hard on a lot of Technical Specifications. C++17 is expected to include a lot of features that were first defined in this manner.

Tom reported that all of the major C++ compilers have extensions for VLA, and the C++ committee intended to put some support for this concept into C++14. The support was called ARB (array with runtime bound) but it was pulled from C++14 and sidetracked into a Technical Specification. The TS also includes some changes for compatibility with the C++ class library. Unfortunately, there are performance issues since the TS currently requires heap-allocated storage and so it isn't clear yet where the TS is headed.

Concerns were raised about how well we are tracking WG21's progress on atomics. That work has apparently matured somewhat and been moved from a study subgroup up to a major subgroup. Several people noted that we have a number of liaisons who are following that work. People were also encouraged to contact Clark if they have a deeper interest in this. He can point them to the WG21 papers he feels are important.

2.4 PL22 (Plum)

Nothing relevant to WG14 to report.

2.5 WG 23 (Benito)

John Benito had retired as convener and Tom Plum is now acting convener.

WG 23 has completed its work. The US has proposed a letter ballot to disband and SC22 has accepted that proposal. The letter ballot will close July 1st and the expectation is that the group will disband.

2.6 MISRA C (Pygott)

The MISRA C committee is currently doing maintenance work. Issue 3 is C99. There is talk about Issue 4 (C11) but it hasn't yet gained any traction

2.7 Other Liaison Reports

Hans Boehm has sent a report to John Benito on the status of WG21 SG1 (C++ Concurrency Study Group). That is now WG14 document [N1805](#).

Clark is on top of the issues and is making sure CPLEX and SG1 are not working at cross purposes.

3. Reports from Study Groups

3.1 CPLEX Activity Report (Parks)

CPLEX continues to hold well-attended, bi-weekly teleconferences. Clark is serving as both chair and project editor.

The group now has a base document written in LaTeX that includes specifications for counted loops, parallel loops, and parallel loop hint parameters.

Several people praised the group's positive, on-going connections to both the OpenMP and C++ communities.

4. Teleconference Meeting Reports

4.1 Report on Any Teleconference Meetings Held

Rajan reported that the floating-point subgroup is making great progress.

5. Future Meetings

5.1 Future Meeting Schedule

- Fall 2014 – St. Louis, MO, USA, 27–30 Oct 2014, [[N 1811](#)]
- Spring 2015 – Lysaker, Norway, 13–17 April 2015
- Fall 2015 – Kona HI, USA, 26-30 Oct 2015

The St. Louis meeting is being sponsored by Bill Seymour and is planned for 4 days, not the usual 4 ½ days.

The Lysaker meeting is being sponsored by Cisco.

The Kona meeting is being sponsored by Plum Hall and will be the week after WG21 meets. If we decide to shorten the meeting to 4 days it will be Oct 26-29.

If anybody wants to host a future meeting please contact David Keaton. We are looking for a host in Europe for the Spring 2016 meeting.

5.2 Future Mailings

- Post Parma: 12-May-2014
- Pre St. Louis: 29-Sep-2014
- Post St. Louis: 01-Dec-2014
- Pre Lysaker : 16-Mar-2015
- Post Lysaker: 04-May-2015

The preliminary agenda for St. Louis is [N1825](#).

6. Document Review

6.1 Floating-point extensions for C – Part 3: Interchange and extended types [\[N 1796\]](#)

Rajan reported that there has been some confusion about what is actually required for an implementation to conform to Part 3. In fact, Parts 1 and 2 are required, but that does NOT mean that decimal floating-point must be supported. It simply means that the decimal FP conformance macro must be set appropriately.

This was the third meeting the committee reviewed this document. The feeling was that the document is now stable and so the committee agreed to send it immediately to PDTS ballot.

With this schedule, we are hoping to complete all of this work by the end of 2015/beginning of 2016. Note that our work on Part 1 is now complete and it should be published soon. Part 2 is about to go to DTS ballot.

[DECISION: move Part 3 to PDTS ballot]

6.2 Floating-point extensions for C – Part 4: Supplementary functions [\[N 1797\]](#)

Rajan reported that 3 of the trig functions proposed in the TS are not in the IEC 60559 standard (tanpi, asinpi, acospi). These were obvious holes that the IEEE committee didn't feel it was necessary to fill. The WG14 subcommittee, however, wanted to add these for completeness.

Part 4 makes apparent that these functions are not in IEC 60559 by having empty entries in column 1 (IEC 60559 operation) of the table in Section 6.

Some on the committee felt that the Part 4 document should simply be a binding to IEC 60559 and so adding these functions would be inappropriate.

[STRAW POLL: should we keep these new trig functions?]

In favor: 11

Opposed: 4

The subcommittee agreed that this document is now ready for PTDS ballot. There was some discussion about whether to conduct a 90 day review period before advancing to that ballot but people didn't feel that was necessary.

[DECISION: move Part 4 to PDTS ballot]

Rajan reported that the subcommittee has now started drafting Part 5: Supplementary Attributes. Participation is still good. And they are hoping to have a draft ready for review at the WG14 meeting in St. Louis.

6.3 Adding methods() to Structures in C [[N 1816](#)], Max Abramson

Max was not there to present his paper. He has been talking to John Benito about it for about a year. It would have been great to have him there to champion his work. The committee believes it is important to listen to its constituency, they welcome their proposals, and they take them very seriously.

There was concern about the name mangling issues this will introduce which will radically change C compilation.

The paper combines about 5 concepts/proposals that could be separated. Some noted that it would be good to consider these individually.

There was discussion about the motivations offered in the paper: improved compilation time and memory footprint, and increased interoperability with C++. Some were concerned that these changes could have the opposite effect. Another motivation might be that these features could enable other technologies. For example, adding member functions might enable us to add container classes cleanly.

About 20 years ago, there was a serious effort to pull interesting parts of C++ into C. The effort was abandoned after a couple years. Language design is difficult; everything has to hang together. And the world loves C in part because of its simplicity.

There was discussion about finding the committee documents from that effort. Unfortunately, these documents are not available from our [document registry](#) in electronic form. Looking at the document registry, very few of the documents older than N598 (06 Sep 96) are available on-line. And the most interesting documents in this area seems to begin with N298 (Jervis, Classes in C, 16 Nov 93). That was followed by a number of Jervis papers on constructors and destructors, virtual functions, and inheritance over the next couple years.

7. Defect Reports

7.1 Discussion of the Defect Report Process

There was discussion about the minutes from our Defect Report sessions. Some felt it was important that they capture committee sentiment and not fine details of the conversation, lest they discourage people from freely expressing their opinions and changing their minds for fear of being viewed as inconsistent.

John reviewed some aspects of our current process:

- Any time a DR is changed it moves back to Open. During the next meeting, if it isn't changed, it may move from Open to Review. The meeting after that, if it isn't changed, it may move from Review to Closed.
- DRs can be submitted by: national bodies, the Project Editor, or the Convener. John noted, however, that we do not want to stifle input and generally treat any defect as a DR.
- When the committee changes its mind on a Technical Corrigendum, it replaces the existing words "below the line" and does not, in general, save the history. There has been no need for that complexity.
- Proposed TC's that come from the committee are generally presented as separate documents. They are not written directly into the Defect Reports.

For this meeting, Blaine chaired the DR session and Douglas served as recording secretary (in addition to JP who was taking the minutes for the WG14 meeting). Blaine's said his goal was to get all of the DR changes agreed to out in the post-Parma mailing.

7.2 IS 9899:2011

7.2.1 Possible Defect Report: Atomic flag type and operations [\[N 1776\]](#) (Tydeman)

Accepted as a defect. Becomes DR 453.

7.2.2 Possible Defect Report: ATOMIC_VAR_INIT [\[N 1777\]](#) (Tydeman)

The committee handled the 5 points in this paper individually.

1. ATOMIC_VAR_INIT does not set the state of the atomic object; it only sets the value

Committee sentiment was that this is not a defect

2. "initialization-compatible" is not defined

The committee was divided here. Some thought that we don't need to define this term since it is being used in a normal English way. Others thought that it looks like a "defined" term but we don't give it a formal definition. We agreed to ask the author to propose better words.

3. ATOMIC_VAR_INIT is not usable in assignment to an atomic object

This is a question and not a defect. The committee has always accepted both bugs and questions in Defect Reports, however, and so we accepted this as DR 454. See below for more discussion of this issue.

4. What should happen if ATOMIC_VAR_INIT(value) is used in context other than initializing an atomic object of the same type as the value?

This is also a question and not a bug. The answer is almost certainly "it would be (implicitly) undefined behavior". We decided to fold this into DR 454 and deal with it later. See below.

5. Zero initialization of static atomic objects in C requires more than in C++

The committee agreed that this is a good question. It was not discussed in Hans Boehm's liaison report from WG21 SG1. And we very much want to know what WG21 is thinking about it. They have a number of open issues in this area.

We agreed to make this into DR 455. We will also ask Clark to take this to WG21. It is discussed more below.

[ACTION: Clark to take N1777 Part 5 to WG21]

7.2.3 Stability of indeterminate values in C11 [[N 1793](#)] (Wiedijk)

This paper raises issues Freek has encountered with his graduate student Robert Krebbers. It specifically asks whether the value of uninitialized variables can change without the program explicitly changing them. It also asks exactly when operations involving indeterminate values lead to undefined behavior.

[DR 260](#) addresses these questions. At first the committee said that the bit pattern of an indeterminate value could NOT change without direct action of the program. Then in 2004 it changed its mind and said that the bit pattern of an indeterminate value COULD change without direct action of the program. This paper proposes we change our response to DR 260 back to the original statement.

There was lots of discussion about this. Some agreed with Freek that the standard currently doesn't allow for indeterminate values to wobble. Others stated that this would not impact their implementations since the optimizations that can take advantage of this are only enabled in non-standards-conforming modes.

Several committee members were strenuously opposed to requiring that compilers stabilize the values of uninitialized variables. They noted that performance is more

important than theory here, especially since the practical implication of stabilizing uninitialized variables would be to allow users to rely on that in their code, a practice we don't want to support.

It was noted that whatever we decide here is likely to affect C++ too since our memory models in this area are consistent.

We cut the discussion off at this point, agreeing to take it up again when we got the Defect Reports. See DR 451 below.

7.2.4 Possible Defect Report: UINTN_C [[N 1798](#)] (Bhakta)

Committee sentiment: this is a defect. Assigned DR 456.

7.2.5 Correcting the definition of ctime_s [[N 1802](#)] (Keaton)

Committee sentiment: this is a cut-and-paste mistake. Assigned DR 457.

7.2.6 Atomic issues in [DR423](#) and [DR431](#) [[N 1803](#)] (Garst)

Deferred until later in the meeting.

7.2.7 Discussions on [DR440](#), [DR441](#), [DR442](#), [DR444](#), and [DR445](#) [[N 1804](#)] (Garst)

Deferred until later in the meeting.

7.2.8 ATOMIC_XXX_LOCK_FREE macros not constant expressions [[N 1806](#)] (Sebor)

This is discussed in message [13216](#) on the reflector. Committee sentiment: this is a defect. Assigned DR 458.

7.2.9 atomic_load missing const qualifier [[N 1807](#)] (Sebor)

Committee sentiment: this is a defect. Assigned DR 459.

7.2.10 aligned_alloc underspecified [[N 1808](#)] (Sebor)

Committee sentiment: this is a defect. Assigned DR 460.

7.2.11 Problems with references to objects in signal handlers [[N 1812](#)] (Sebor)

Deferred until Martin calls in on Tuesday. Assigned DR 461.

7.2.12 Clarifying objects accessed in signal handlers [[N 1813](#)] (Seacord)

Committee sentiment: this is a defect. Assigned DR 462.

7.2.13 Harmonizing left-shift with C++14 [[N 1817](#)] (Ballman)

The committee expressed concern that C++ is making this change. It has been undefined behavior in C from the start. In any case, we agreed that it is a feature request and not a defect in the standard.

We decided to create Standing Document 3 [[N 1826](#)] to track open feature requests, and then add this to that document. We also assigned this DR 463 and will respond in the DR that we are tracking this as a feature request.

7.3 TS 17961:2013, C Secure Coding Rules (CSCR)

7.3.1 Error in 5.21 example [[N 1801](#)] (Pygott)

The committee decided to accept this as a Defect Report. We did not give it a DR number but will instead refer to it using the N number. We may then republish the TR once we incorporate the change.

We have treated defects against our Embedded TR in similar fashion. For reference, [N 1180](#) is the defect log for that TR.

8. Other Business

8.0 Converting our Base Document to LaTeX

We have quite a few Defect Reports that are now closed and so we would like to think about republishing the standard to incorporate these changes. Recall that we've already published one TC for C11, so this would be our second.

Our base document is written in troff/nroff and extremely difficult to build, so this might be a good time to think about rewriting the standard in LaTeX (like C++ and Fortran) or some other editorial language. There was enthusiastic support for this idea.

We discussed a number of formats. Microsoft Word has been used to write significant standards documents and has the advantage that ITTF can probably publish our documents more quickly. Framemaker has also been used for important documents (including Cobol) but is costly and not as well known. In the end the committee seemed to prefer LaTeX.

There was a long discussion about how we might go about converting the document to LaTeX. Fortran has a fairly large and powerful macro package that might help. And Pete

Becker apparently has quite a bit of experience in this area, so we gave John Benito an action to talk with Pete about the conversion.

There was also a discussion about how we might verify that the conversion is correct. The sense was that everyone on the committee would have to take part and review individual sections. People also believed we should probably convert the base document first, verify that, then incorporate the TCs. That would require backing out the TC's that Larry has already folded in. But we also thought that we shouldn't micro-manage the project. We should find a project owner and let them work out the details.

**[ACTION: Benito to talk to Pete Becker about converting our base document to LaTeX]
CLOSED**

John now has a volunteer from the committee who is now working on this, so the action is no longer necessary.

8.1 Appropriate links for the Web Site

Deferred until later in the meeting.

Discussion of Defect Reports in REVIEW Status

[DR 421 – REVIEW](#)

Committee agreed this is not a defect. Move to **CLOSED**.

[DR 422 – REVIEW](#)

Committee agreed this is not a defect. Move to **CLOSED**.

[DR 428 – REVIEW](#)

This is slightly confusing because the full resolution of all of the problems discussed here requires both this TC and the TC for [DR 433](#). That is stated in the Committee Discussion and is clear enough. Move to **CLOSED**.

[DR 430 – REVIEW](#)

Move to **CLOSED**.

[DR 432 – REVIEW](#)

Move to **CLOSED**.

Discussion of Defect Reports in OPEN Status

[DR 406](#) – **OPEN** Visible sequences of side effects are redundant

This DR proposes a simplification of the memory model to make it clearer. The committee has in the past expressed sympathy with the proposal but has been concerned about remaining consistent with C++.

This proposal became C++ Core Issue 1466 (WG21 document N3914) and there is a proposed solution but we do not know if and when it might be formally adopted and published. We decided to ask Clark to investigate and report back to us. We left this **OPEN**.

[ACTION: Clark to investigate what C++ committee has done about DR 406]

[DR 407](#) – **OPEN** memory_order_seq_cst fence sequencing rules

This issue affects both C and C++ and was tracked as WG21 Library Issue 2130. The DR contains words that have been proposed to the C++ working draft but we do not know if and when they might be formally adopted and published.

We agreed that the words “Proposed Technical Corrigendum” should be removed from the DR since those words were changed in the C++ draft and are no longer appropriate. We also agreed to ask Clark to investigate what WG21 is doing about this and report back to us. We left this **OPEN**.

[ACTION: Clark to investigate what C++ committee has done about DR 407]

[DR 413](#) – **OPEN** Initialization order

This DR sparked a long committee discussion and several proposals over many meetings about initialization order when both explicit and implicit initialization occurs. In the end, we decided to 1) answer the specific question about the result of executing the example code, and 2) add the example to section 6.7.9 of the standard to clarify that implicit initialization does not override explicit initialization. We made that decision at the last meeting. In Parma, we decided to move this to **REVIEW** with no further discussion.

[DR 416](#) – **OPEN** tss_t destruction unspecified

This DR has been discussed over several years and with several papers. Finally, in Chicago, we settled on a TC. In Parma, we decided to move this to **REVIEW**. We also noted that Blaine should editorially change “Proposed Committee Corrigendum” to “Proposed Technical Corrigendum” (our term of art).

DR 423 – OPEN under specification for qualified rvalues

This DR asks whether type generic macros should differentiate between atomic and the corresponding non-atomic types.

Blaine wrote [N1803](#) to discuss this question and related [DR 431](#) which asks for guidance on `atomic_compare_exchange`. In that paper, he discusses the possibility of using padding byte(s) to provide locks. But since he knew of no existing or proposed implementations that did that, he was comfortable recommending a solution that would remove the possibility of such an implementation.

Since submitting the paper, Blaine has learned that there are indeed implementations that have taken that path. And since he didn't want to break them, he withdrew the suggestion he made in N1803 and intends to write another paper with a different proposed TC. Leave **OPEN**.

[ACTION: Blaine to write paper with proposed TC for DR 423]

DR 424 – OPEN under specification of `tss_t`

This is discussed and fully resolved in the TC for [DR 416](#). Move to **REVIEW**.

DR 426 – OPEN G.5.1: `-yv` and `-x/v` are ambiguous

This was discussed at length in Chicago where the committee reluctantly agreed to a modest TC. There was no desire to discuss it further in Parma. Move to **REVIEW**.

DR 427 – OPEN Function Parameter and Return Value Assignments

Several members of the committee were uncomfortable with the proposed wording changes. They were not yet convinced that changing from "assignment" to "initialization" would yield exactly the right set of promotions and conversions. They mentioned clearing up any possible confusion by writing specifically about qualifiers, but they didn't want to draft words to that affect during the meeting. It was decided instead go back to the author for more input.

[ACTION: Blaine to go back to Shao Miller for more input on DR 427]

Blaine will add Committee Discussion capturing these thoughts. Leave **OPEN**.

DR 429 – OPEN Should gets s discard next input line when (s == NULL)?

The committee moved this to **REVIEW** without further discussion.

DR 431 – OPEN atomic compare exchange: what does it mean to say 2 structs compare equal?

Blaine wrote [N1803](#) to discuss this and [DR 423](#) (see above). That paper was predicated on the idea that size and alignment are not interesting but they ARE interesting.

The committee sentiment was that `_Atomic` struct comparison should simply be disallowed (i.e. it is undefined behavior). Blaine offered to write a paper explaining this and addressing deeper issues around `memcmp` and `memcpy`. Leave **OPEN**.

[ACTION: Blaine to write paper on DR 431]

DR 433 – OPEN wide character function arguments involving RSIZE_MAX

The committee was happy with the resolution but offered a couple editorial suggestions. `K.3.9.3.2.2p12` has “nor nor” which we assume Larry will clean up. It was also suggested that “Suggested Technical Corrigendum” with no text following it is somewhat confusing. Perhaps “NA” would help. Move to **REVIEW**.

DR 434 – OPEN Missing constraint w.r.t Atomic

The committee moved this to **REVIEW** without further discussion.

DR 435 – OPEN Missing constraint w.r.t Imaginary

The committee moved this to **REVIEW** without substantive discussion. It was noted editorially that “Suggested Committee Response” should be “Proposed Committee Response.”

DR 436 – OPEN Request for interpretation of C11 6.8.5#6

The committee moved this to **REVIEW** without substantive discussion. It was noted editorially that the term “expression” before “*157:” should be “expression-3”.

DR 437 – OPEN clock overflow problems

The committee felt like it needed input from Nick before proceeding. Leave **OPEN**.

[ACTION: Benito to ask Nick for more input and a new TC for DR 437]

DR 438 – OPEN ungetc/ungetwc file position after discarding push back

At the Chicago meeting, the committee felt that the suggested words were almost correct but requested that the project editor suggest better wording. Larry provided these words in Parma:

Committee Discussion

Larry Jones pointed out that the Standard is correct as written because the intent is that the specified file position indicator is an intermediate state inside the file positioning function after the pushed-back characters are discarded but before the actual seek. That gives you a reliable file position from which to do the seek. It's not intended that the file positioning function doesn't set the file position indicator.

Bill Seymour suggested adding a footnote to avoid confusion.

Proposed Technical Corrigendum

Add a footnote to 7.21.7.10 paragraph 5, second sentence:

Note that a file positioning function may further modify the file position indicator after discarding any pushed-back characters.

Add a footnote to 7.29.3.10 paragraph 5, second sentence:

Note that a file positioning function may further modify the file position indicator after discarding any pushed-back wide characters.

The committee accepted Larry's new words and will add to the DR. Leave **OPEN**.

DR 439 – OPEN Issues with the definition of "full expression"

The committee is waiting for more input from Clark. Leave **OPEN**.

DR 440 – OPEN Floating-point issues in C11 from PDTS 18661-1 UK review, Issue 1

Joseph Myers authored [N1730](#) and [N1731](#), which gave rise to DR 440, DR 441, DR 442, DR 443, DR 444, and DR 445.

Blaine authored [N1804](#) which gives additional information and suggestions for these defect reports.

In Parma, the committee agreed to move the second bullet in the “Suggested Committee Response” section in the DR 440 discussion of N1804 to the “Committee Response” section of the DR.

It was further decided that this is a feature request and not a defect. And since we don’t want to lose track of our feature requests, we agreed to create a new standing document to track these.

[ACTION: Benito to create Standing Document 3 and add DR 440] CLOSED

SD3 is now [N1826](#) and available on the website.

DR 441 – OPEN Floating-point issues in C11 from PPTS 18661-1 UK review, Issue 2

Committee sentiment was that this is not a defect. Editorially, it was noted that the third bullet in the Proposed Committee Response should be F.2 and not F.3. Move to **REVIEW**.

DR 442 – OPEN Floating-point issues in C11 from PPTS 18661-1 UK review, Issue 3

[N1804](#) from Blaine addresses this DR.

Committee sentiment was that this is not a defect and the normative requirements relative to Annex F are clear enough. Blaine offered to write a Proposed Committee Response that says that. Leave **OPEN**.

[ACTION: Blaine to write a Proposed Committee Response to DR 442]

DR 443 – OPEN Floating-point issues in C11 from PPTS 18661-1 UK review, Issue 4

The committee agreed that the FPE (floating-point environment) is not an object but they were uncomfortable with moving footnote 205 into normative text. The sense was that there was no real need to define FPE more formally.

The committee had some sympathy with bullet 3 in the Committee Discussion (the standard does not formally define "system variable") but they had no proposed words to consider. In the end, Blaine offered to write a Proposed Committee Response conveying the sentiment that this is not a defect. Leave **OPEN**.

[ACTION: Blaine to write a Proposed Committee Response to DR 443]

DR 444 – OPEN Issues with alignment in C11, part 1

Joseph Myers provided a suggested TC in [N1804](#) and the committee believes it will work.

If the committee were to adopt those changes there would be no supported way to apply `_Alignas` to non-aggregates. It would become a non-portable extension. Most on the committee believed that was acceptable. Some were skeptical. In the end, the committee decided to simply add a link to [N1804](#) to the Committee Discussion and leave this **OPEN**.

[ACTION: Blaine to add link to N1804 to Committee Discussion in DR 444]

[DR 445 – OPEN Issues with alignment in C11, part 2](#)

Joseph Myers discusses this in [N1804](#) as well. The committee took no action on this. Leave **OPEN**.

[ACTION: Blaine to add link to N1804 to Committee Discussion in DR 445]

[DR 446 – OPEN Use byte instead of character for memcmp, memcpy](#)

The committee moved this to **REVIEW** without substantive discussion.

[DR 447 – OPEN Boolean from complex](#)

The committee moved this to **REVIEW** without substantive discussion.

[DR 448 – OPEN What are the semantics of a # non-directive?](#)

The committee moved this to **REVIEW** without substantive discussion. Editorially, it was recommended that the project editor should add this to Annex J section J.2 (Undefined behavior).

[DR 449 – OPEN value of TSS DTOR ITERATIONS for implementations with no max](#)

The committee agreed that the standard does not define this value intentionally and Douglas agreed to provide words to that affect.

[ACTION: Douglas to provide Proposed Committee Response for DR 449] CLOSED

Douglas provided these additional words to be added to the Proposed Committee Response in a document posted on the wiki (and agreed to by the committee):

Proposed Committee Response

The standard intentionally does not define a value of `TSS_DTOR_ITERATIONS` for implementations with no maximum.

Leave **OPEN**.

DR 450 – OPEN `tmpnam` clears `s[0]` when `maxsize > RSIZE_MAX`

The committee agreed with the sentiment of the DR but wanted the overlong sentence in the Suggested Technical Corrigendum broken into parts to make it more readable.

[ACTION: Martin to provide better words for DR 450]

DR 451 – OPEN Instability of uninitialized automatic variables

This was a continuation of the discussion that began on the first day of the meeting when the committee discussed [N1793](#) and decided to make it DR 451.

The discussion was guided by a slide presentation by Freek Wiedijk which is now document [N1818](#). The presentation sparked a long discussion about wobbly values, compiler optimizations that take advantage of such values, Annex L, and [DR 260](#) (which the authors were proposing to change). In the end, committee sentiment seemed to be that:

1. we never intended the standard to require that implementations fix the values of uninitialized automatic variables (i.e. they may legitimately perform optimizations based on this freedom)
2. the current resolution of [DR 260](#) is correct and should not be changed
3. we may need to do more work in this area to specify this more precisely, possibly in a future revision of the standard
4. when specifying this more precisely, we may want to distinguish padding bytes from other indeterminate values

Blaine offered to write a Proposed Committee Response that says this. Leave **OPEN**.

[ACTION: Blaine to write Proposed Committee Response to DR 451]

There was some discussion later in the meeting about whether or not, if we decide that it is undefined behavior, that fact should be mentioned in Annex L. Committee sentiment was that fetching from uninitialized variables is already in Annex L and so nothing more needs to be done.

There was also more discussion about how we might clarify this in a future revision of the standard. One idea was that we might push “indeterminate value” further from

“unspecified value” and state that reading an indeterminate value results in undefined behavior. This was just an idea and not a serious proposal.

DR 452 – OPEN Effective Type in Loop Invariant

This was discussed earlier in the meeting. Blaine took an action item to go back to the author (Shao Miller) for more information. Leave **OPEN**.

Discussion of NEW Defect Reports

DR 453 [N1776] - Atomic flag type and operations (Tydeman)

There was a long discussion about the word “set”. In the end, the committee agreed that:

1. atomic_flag has 2 states: “set” and “clear”
2. the atomic test-and-set functions return TRUE if the flag was “set” on entry to the function and FALSE if it was “clear”
3. test-and-set sets the value of the flag to “set”

Unfortunately, since “set” is used as both a state and an action, there was confusion and it was difficult to reach agreement on the wording. So Clive took an action to propose words.

[ACTION: Clive to propose new words for DR 453] CLOSED

Clive proposed the following words later in the meeting:

Change - proposed replacement text for 7.17 8.1 #2

7.17.8.1 The atomic_flag_test_and_set functions

#2: Atomically tests the state of the flag pointed to by object and then either sets it if clear else leaves it set

#3: Returns true if the flag was set when tested or false otherwise.

The committee discussed these new words at length but did not accept them. They felt that the phrase “set it if clear else leaves it set” was somewhat confusing. So Rajan agreed to take a shot at it (with help from John Benito).

[ACTION: Rajan to propose new words for DR 453]

The committee also noted that the DR proposes adding words to the Rationale. There is no Rationale for C11, however, so this cannot be accomplished. Leave **OPEN**.

DR 454 [N1777] - ATOMIC VAR INIT (Tydeman)

The committee was concerned about what WG21 is doing about Issue 5 and so asked Clark to investigate.

[ACTION: Clark to investigate what C++ is doing about Issue 5 in N1777/DR 454]

After some discussion about the differences between C and C++, and with agreement that we don't want to change the C standard here, Blaine offered to propose words to clarify this.

[ACTION: Blaine to write Proposed Committee Response for DR 454] CLOSED

Blaine offered this in a paper he put up on the wiki:

Proposed Committee Response

The `ATOMIC_VAR_INIT` macro prepares an atomic value that includes any extra state necessary for a non-lock-free type. Initialization, by definition, ignores all previous state. Assignment must honor the extra state that would indicate another atomic operation in progress; such an assignment takes the non-atomic corresponding value resulting from removing all qualifiers including atomic from the value expression, and will manipulate the extra state held in the object to assure proper atomic assignment semantics. `ATOMIC_VAR_INIT` produces a value appropriate for initialization because it will have any necessary extra state, whereas a value suitable for assignment is the non-qualified version of the assignment expression.

All uses of `ATOMIC_VAR_INIT` other than for initialization result in implicitly undefined behavior.

The committee accepted these words. Leave **OPEN**.

DR 461 [N1812] - problems with references to objects in signal handlers (Sebor)

Martin Sebor called in to the meeting to present this paper.

The committee expressed concern about the first part of the TC - allowing signal handlers to access const objects in this context. It was pointed out that this is not a defect in the standard, it is a request for a new feature and thus inappropriate to add through the DR process. Others pointed out that allowing const objects could be problematic, though perhaps "const and not volatile" might work. In any case, the committee felt that allowing access to const objects should be removed from the DR.

There was also discussion about whether or not the "missing restriction" section should be extended to handle user-allocated storage. There was general agreement that it should.

In the end, Martin agreed to rewrite the paper to 1) remove the const extension, and 2) add another missing restriction on malloc'ed memory.

[ACTION: Sebor to rewrite N1812/DR 461 to remove const extension and add allocated storage (though that may be a separate document)]

DR 456 [N1798] - UINTN C(value) macro (Rajan)

There was quite a bit of discussion about whether or not compiler support is required to implement these macros properly. [DR 209](#) suggests it isn't and Rajan knew of no implementations that currently use such magic, however there was acknowledgement on the committee that it may in fact be required.

Nevertheless, there was no support for removing these macros that people may be using. There was also no support for adding suffixes for char and short, which people felt was too heavyweight a solution.

There was some discussion of a third possibility: relaxing the rules for these macros, but given no written proposal the discussion didn't go far. It was decided to leave this OPEN with the understanding that Rajan may come back with a proposal to relax these rules if he so desires.

Later in the meeting, Rajan brought us a new document (n1798_take_2.txt on the wiki) to provide more background information for this DR. He writes:

7.20.4.1: The macro `UINTN_C(value)` shall expand to an integer constant expression corresponding to the type `uint_leastN_t`.

For $N = 8$ or 16 , on systems with `int` as 32 bits, `char` at 8 bits and `short` at 16 bits, how do you get an expansion that results in a 8 or 16 bit type since there are no literal suffixes for `char` or `short` types?

Since the only way to turn an integer literal into an unsigned char type is through a cast (permitted by 6.6p6), a possible implementation would be:

```
#define UINT8_C(c) (uint_least8_t)c
```

However this does not work since due to 7.20.4p3 this expansion has to work for preprocessing conditional expressions.

Implementations seen:

```
4. #define UINT8_C(c) c ## U
```

Gives {false, false} for the phase {4, 7} comparison due to promotion to {uint_max, unsigned int} (6.3.1.8p1 last point). Note the type is unsigned int, not unsigned char (or unsigned short for UINT16_C).

2) #define UINT8_C(c) c

Gives {true, true} for the phase {4, 7} comparison but does not 'correspond' to an unsigned char type: the type is {int_max, signed int} assuming unsuffixed literals, not signed char (or signed short for UINT16_C).

If you have (1), there are no suffixes for integers with rank less than int. This means there is none for char (8 bits) or short (16 bits) in common implementations. In comparisons between types with the same rank like int and unsigned int, the usual arithmetic conversion rules make both unsigned so in this case unsigned int (6.3.1.8).

If you have (2), the standard says UINT8_C has to have type corresponding to uint_least8_t (7.20.4.1) which is commonly 'unsigned char' (or at least can be). This means the promotion rules should make any argument 'c' an integer literal (or if it is unsigned char type, it would promote into an int) which would make the comparison between int and int so the comparison returns true.

Note: The wording in 7.20.4 does not seem to acknowledge that the integral promotions produce different types depending on whether the evaluation is occurring in phase 7 of translation or in phase 4 as part of the controlling expression of a conditional inclusion though 6.10.1 does seem to try to mention this.

DR 209 seemed to try to address this but the problem seems to remain.

DR 457 [N1802] - asctime s (Keaton)

The committee agreed that this was a cut-and-paste error and agreed with the Suggested TC. Move Suggested TC to Proposed TC and leave **OPEN**.

DR 458 [N1806] - ATOMIC XXX LOCK FREE macros (Sebor)

The committee expressed concern that requiring these macros to be suitable for use in #if expressions may break implementations. Nevertheless, there was agreement that we always intended for that requirement to be there.

Move Suggested TC to Proposed TC and leave **OPEN**.

DR 459 [N1807] - atomic load functions missing const qualifier (Sebor)

The committee agreed that this was a simple oversight.

Move Suggested TC to Proposed TC and leave **OPEN**.

DR 460 [N1808] - *aligned alloc underspecified* (Sebor)

The committee was sympathetic to the first proposal but not the second (which begins with “If the proposal above isn’t acceptable” and adds new function `alignment_is_valid`). There was a bit of hesitation about the first proposal but an understanding that the TC will remain open and the committee will revisit it in 6 months.

Move the first part of the Suggested TC to Proposed TC and leave **OPEN**.

DR 462 [N1813] - *clarifying objects accessed in signal handlers* (Seacord)

The committee agreed with this paper and hoped that Robert would take it to WG21 as well.

Move Suggested TC to Proposed TC and leave **OPEN**.

DR 463 [N1817] - *harmonizing left-shift with C++14* (Ballman)

The committee felt this was a feature request and not a defect and so added it to Standing Document 1 ([N1826](#)). Leave **OPEN**.

[N1801] - *CSCR error in 5.21 example* (Pygott)

As discussed above, the committee decided not to give this a defect number since it is about a Technical Specification and not the C Standard. And the subcommittee that wrote that TS has now disbanded.

It was pointed out that in the Committee Discussion it would be good to change “`malloc(8)`” to “`malloc(2)`”.

Committee sentiment was that we should move Suggested TC to Proposed TC with the “`malloc(8)`” change suggested above. If and when there is a revision of the Technical Specification, the committee can revisit whether or not to change the rule referenced in the paper. Leave **OPEN**.

7. Other Business

8.1 Appropriate links for the Web Site

We have been approached by a fellow with a fairly extensive website that teaches the C language. He would like us to add a link to his website on our official WG14 website.

The committee was mixed on this. We believe it is within our charter to help people use the C language. We want to support our community. Our Defect Report mechanism is a time-consuming way to ask simple questions. And we fully support this fellow's efforts.

On the other hand, the committee felt it that our official WG14 website is an inappropriate place for links like this and so we decided to politely decline this request.

One more suggestion was made. Although our email reflector archive is easy to use -- from the main page URL (<http://www.open-std.org/jtc1/sc22/wg14/>) simply append an email number -- this is not widely known. We agreed it would be good to publicize this better.

8.2 Secretariat

John Benito is retiring as both our Convener and Secretariat September 8, 2014. David Keaton has graciously volunteered to replace him as Convener and we are currently going through the process of nominating and confirming him for that role. We do not have a volunteer to take over John's role as Secretariat, however. That role involves managing our document numbers, bundling our pre and post mailings, and posting things to our website. If anyone would like to take on that role, they should contact David.

9. Resolutions and Decisions Reached

9.1 Review of Decisions Reached (Parks)

1. Move "Floating-point extensions for C – Part 3: Interchange and extended types" to PPTS ballot
2. Move "Floating-point extensions for C – Part 4: Supplementary functions" to PPTS ballot

9.2 Review of Action Items (Parks)

ACTION: Clark to take N1777 Part 5 to WG21

ACTION: Benito talk to Pete Becker about converting standard to LaTeX **CLOSED**

ACTION: Clark to investigate what C++ committee has done about DR 406

ACTION: Clark to investigate what C++ committee has done about DR 407

ACTION: Blaine to write paper with proposed TC for DR 423

ACTION: Blaine to go back to Shao Miller for more input on DR 427

ACTION: Blaine to write paper on DR 431

ACTION: Benito to ask Nick for more input and a new TC for DR 437

ACTION: Benito to create Standing Document 3 and add DR 440 **CLOSED**

ACTION: Blaine to write a Proposed Committee Response to DR 442

ACTION: Blaine to write a Proposed Committee Response to DR 443

ACTION: Blaine to add link to N1804 to Committee Discussion in DR 444

ACTION: Blaine to add link to N1804 to Committee Discussion in DR 445

ACTION: Douglas to provide Proposed Committee Response for DR 449 **CLOSED**

ACTION: Martin to provide better words for DR 450

ACTION: Blaine to write Proposed Committee Response to DR 451

ACTION: Clive to propose new words for DR 453 **CLOSED**

ACTION: Blaine to write Proposed Committee Response for DR 454 **CLOSED**

ACTION: Sebor to rewrite N1812/DR 461 to remove const, add allocated storage

ACTION: Rajan to propose new words for DR 453

ACTION: Clive to propose Secure C rule about arrays

10. Thanks to Host

The Committee expressed its thanks to Roberto Bagnara and the University of Parma for hosting the WG14 meeting in the wonderful city of Parma.

The committee also expressed a sincere thank you to John Benito for leading this committee for the past 17 years.

11. Adjournment

Adjourned at 1400, local time, Wednesday, April 9, 2014

PL22.11 TAG Meeting Minutes (Final-Draft) April 8, 2014 Parma (Italy)

Meeting convened on April 8, 2014, at 16:00 pm by PL22.11 Chair, David Keaton.

Attendees:

<i>Voting Members:</i>		
Name:	Organization: P – Primary, A - Alternate	Comments
John Benito	Plum Hall, Inc. - A	
David Keaton	CERT/SEI/CMU-P	PL22.11 Chair
Daniel Plakosh	CERT/SEI/CMU-A	
P. J. Plauger	Dinkumware, Ltd – P	
Blaine Garst	Garst - P	
Rajan Bhakta	IBM - P	
John Parks	Intel - P	PL22.11 Acting Secretary
Clive Pygott	LDRA - P	
Douglas Walls	Oracle - P	PL22.11 IR
Tom Plum	Plum Hall, Inc. – P	
Jim Thomas	Tydemans Consulting - A	

1. Approval of Agenda

It was suggested that we remove Section 3 (Selection and Review of US Delegation) since we are covered through the end of 2014. See below.

Agenda was approved by unanimous consent. (Walls/Plakosh)

2. Approval of Previous Minutes (PL22.11/13-002)

Minutes were approved by unanimous consent. (Pygott/Benito)

3. Selection and Review of US Delegation.

Not applicable for this meeting. Due to changes in ISO rules regarding delegations, they no longer exist for WG meetings, this item will be likely be removed from future agenda/minutes.

4. INCITS [Antitrust Guidelines and Patent Policy](http://www.incits.org/standards-information/legal-info)

We reviewed the content contained in
<http://www.incits.org/standards-information/legal-info>

5. INCITS official designated member/alternate information.

Be sure to let INCITS know if your designated member or alternate changes, or if their email address changes. Send contact info to Lynn Barra at ITI, lbarra@itic.org.

6. Identification of PL22.11 Voting Members (Parks)

See attendance list above.

9 PL22.11 voting members participated out of 14.

6.1 PL22.11 Members Attaining Voting Rights at this Meeting

None

6.2 Prospective PL22.11 Members Attending Their First Meeting

None

7. Member Status

7.1 Members who have received warnings between meetings.

None

7.2 Members who lost voting rights between meetings.

None

7.3 Members who lost voting rights following this meeting.

Bloomberg - lack of attendance.

7.4 Members who gained voting rights at this meeting.

None

7.5 Members who regained voting rights at this meetings.

None

7.6 Members who will receive a warning following this meeting.

Coverity, Seymour

8. Procedures for Forming a US Position

We were reminded that the best time to get substantial changes into our Technical Specifications is during sub-group work or during full committee meetings, not during the ballot process.

We were reminded that our Floating-point Part 2 TS is now in DTS ballot. Parts 3 and 4 are now in POTS ballot.

9. New Business

It is time for the 5 year review of TR 24747: Mathematical Special Functions. We cannot stabilize it yet but will be able to do that in 5 more years.

Recommendation for ISO/IEC 24747:2009 Information technology -- Programming languages, their environments and system software interfaces -- Extensions to the C Library to support mathematical special functions

United States Response (Suggested Answers):

1. Recommended action

Confirm

2. Has this International Standard been adopted or is it intended to be adopted in the future as a national standard or other publication?

Yes, Comment: INCITS/ISO/IEC 24747:2009[2010]

3. Is the national publication identical to the International Standard or was it modified?

Identical

4. Is this International Standard used in your country without national adoption or are products used in your country based on this standard?

N/A

5. Is this International Standard, or its national adoption, referenced in regulations in your country?

No

Roll Call Vote: Should the US adopt these Suggested Answers?

Yes: 9 (CERT/SEI/CMU, Dinkumware, Garst, IBM, Intel, LDRA, Oracle, Plum Hall, Tydeman Consulting)

No: 0

Abstain: 0

Not Present (but eligible to vote): 4 (Coverity, Microsoft, Perennial, Seymour)

Suggested Answers are adopted: 9/0/0/5/14

10. Next Meeting: St. Louis, MO, USA

Fall 2014: St. Louis, Oct 27-30, Seymour

Spring 2015: Lysaker, Norway April 13-17, Cisco

Fall 2015: Kona, HI, USA, Oct 26-30, Plum Hall

11. Adjournment

The meeting was adjourned at 1630 local, April 8, 2014 by unanimous consent (Tydeman/Garst).