

WG21 November 2024 Hybrid meeting

Minutes of Meeting

ISO/IEC JTC1 SC22 WG21 N5000— 2024-12-06

Nina Dinka Ranns, dinka.ranns_at_gmail.com

Chair: John Spicer

18-23 November 2024, Wroclaw, Poland

1. Opening activities

John Spicer opens the meeting at 09:08 AM UTC+1.

The wifi information is on the screen.

1.1 Opening comments, welcome from host

John Spicer welcomes the group.

Welcome from the host.

Nokia representative welcomes the group.

1.2 Meeting guidelines

John Spicer presents.

We have a wiki that contains meeting specific information. Please do not edit the wiki unless you have been explicitly instructed to do so

We also have a github tracker. Please do not update github issues unless you have been explicitly instructed to do so by the relevant chair.

Please speak into the microphone so people participating over Zoom can hear. Please introduce yourself when speaking.

Meetings are not public, we want everyone to be able to speak freely. Please refrain from live tweeting, blogging, taking photos of other people's screens or recording the meetings. You're allowed to take screenshots of presentations for your personal use.

Agenda is on the wiki.

Every participant is responsible for understanding and abiding by the following:

[The ISO Code of Conduct](#)

[The IEC Code of Conduct](#)

[The WG21 Practices and Procedures, and Code of Conduct](#)

Documents are on the wiki. Please get familiar with them. They also include a description of the process we follow.

You are expected to abide by the rules of the code of conduct of your respective NB.

Nina Ranns : Should you at any point find yourselves overwhelmed or needing advice on an uncomfortable situation, do not hesitate to approach a WG21 officer. : Herb, John, or me. If you can't find us, send us an email and we will be in touch. We will hear you out and hopefully find a way together with you to get you back into the rumble.

John Spicer: For plenary polls, you have to be in the ISO global directory to vote. One person, one vote. In working groups and study groups everyone can vote. Please refer to the best practices in the WG21 document - e.g. do not vote unless you are familiar with the issue.

Nevin Liber : Attendance sheet is live. You can find it on wiki, mattermost, and posted to the reflector. If you have registered, it's pre-populated with your information. Please mark your attendance.

John Spicer explains voting procedure for remote and in person attendance.

1.3 Introductions

Introduction of the WG21 officers.
Introduction of admin support roles.
Introduction of the subgroup chairs.
First time attendees introduce themselves.

Herb Sutter welcomes the group.

1.4 Agenda review and approval

John Spicer presents the agenda and timings for the week.

The primary goals of this meeting will be work on C++26 features. For more information on the schedule, please see P1000R6

Motion to approve the meeting agenda.
No objections.
Approved.

1.5 Editor's reports, approval/adoption of working drafts

| Document | Editor's report | Prospective WD |
|----------------------|-----------------------|-----------------------|
| C++ 26 Working Draft | N4994 | N4993 |

Motion to approve the documents above.
No objections.
Approved.

1.6 Approval of the minutes of the previous meetings

| Meeting | Minutes |
|---|-----------------------|
| WG21 St. Louis | N4985 |
| WG21 pre-Wroclaw administrative telecon | N4998 |

Motion to approve the documents above.
No objections.
Approved.

2. Liaison reports, and WG21 study group reports (see pre-meeting WG21 telecon minutes)

No discussion.

3. WG progress reports (Core, Evolution, Library, Library Evolution; see pre-meeting WG21 telecon minutes)

No discussion.

4. New business requiring action by the committee

Herb Sutter : C++23 was just published. Does it mean it is now called C++24 ? No.

5. Organize working groups and study groups, establish working procedures

Jens Maurer presents room assignments.

Room assignments are on the wiki page. Any questions, please let me know. Any issues with remote attending setup, find me.

6. Subgroup sessions

John Spicer presents. The subgroup chairs must arrange for any proposals to be written up in the form of a motion, and made available by 8:00 PM Friday on the straw polls page together with associated papers. Groups are encouraged to make those papers and polls available as soon as possible during the week so people can have time to review them. Core group and library group have a staging area, so keep an eye out on those..

If you have any questions or issues, please bring them up to attention as soon as possible. on reflector, mattermost, or one of the subgroup chairs.

Herb Sutter : if there is a technical objection in the plenary, we will ask the group chair if it were discussed. If it weren't discussed, we will want to know why the objection wasn't raised earlier so we can learn from it.

Nevin Liber : please let the authors know that you will object. Don't surprise them.

7. Review of the meeting

Reminder: Make sure you have marked the attendance sheet, if you have not already done so.

Subgroup status and progress reports. Presentation and discussion of proposals to be considered for consensus adoption by full WG21.

Nina Ranns : Please note that the N number of the admin telco minutes had to change for admin reasons. The agenda has been updated and the minutes will reflect the correct number. There has been no change to the minutes other than the number.

SG1: Concurrency (Giroux/Boehm)

Olivier Giroux reports.

Numbers

- Attendance: averaged ~12 or so, peaked at ~20, about 1/4 online
- Discussions: 20+ discussions, saw some papers twice in the same meeting
- Time used: 3.5 days

Chair's highlights

- Concurrent queues could make '26 if prioritized
 - Incredibly long incubation - as long as I've been coming
 - One of the most important S&R demonstrations, will help teach S&R
 - The wording is a weakness, helpers much appreciated
- Removing `memory_order_consume`
 - 2nd longest incubation?
 - We really tried, time to concede defeat
 - End of an era took 14 minutes

Chair's lowlights

• We have a real problem with a lack of scribes

- Most of the room says they have some kind of a universal trump card:

- "I might be needed *at any time* in the next room". I propose chairs have a rule that rewards scribing with prioritizing review of the scribe's paper if the paper is being discussed in another room while they are scribing.
- "I can follow and rebut -any- argument in English, but I can't take notes"
- "I only have my phone with me"

- If we can't police this, we need to get AI scribes
- I shall not be scribing while chairing again. I propose we have a rule that says if a person is present in the room they should be prepared to take minutes.

Some changes to running the SG1 room

- At this meeting I invited people to audition to be in-room co-chairs / whips
- i.e. be 2nd most aware person, keep it moving, find the consensus
- Ruslan was the first volunteer, he did a great job
- Expect to see Ruslan at the front of SG1 some % of the time going forward
- Congrats Ruslan !

I would like us to have a rule that states that if you are present you should be prepared to take minutes.

Herb Sutter : Thank you for raising the problem of not finding a scribe. Motion to adopt Oliver's rule.

No objections.

SG4: Networking (Snyder/Ažman)

SG4 did not meet during the Wrocław meeting. SG4 are expecting to discuss a proposal for C++ networking based on IETF TAPS, following the direction agreed upon at the Tokyo meeting, at the Hagenberg meeting.

SG6: Numerics (Kretz/Lippincott/McFarlane)

SG6 met on Monday and Tuesday, we talked about floating-point for a total of one complete day. We had few people in the room and a few online for a minimum of our quorum of 6. SG6 additionally met in a joint session with LEWGI (SG18) for another total of a complete day. But I'll let Nevin report on that.

On P3348 "C++26 should refer to C23 not C17" We requested that the $\langle\text{cmath}\rangle$ parts of C23 are processed via their own paper, so that the remaining parts can make quicker progress.

We clarified our expectations of floating-point behavior for the extended floating-point types and determined that we need to implement behavior-changing attributes of the floating-point standard before we can do anything helpful. This became a recurring theme for the remainder of the SG6 sessions.

We discussed:

P3397 Clarify requirements on extended floating point types
P3375 reproducible floating-point results
CWG2752 Excess-precision floating-point literals
P3488 Floating-Point Excess Precision
P3479 Enabling C pragma support in C++

We finally had a very short look at

P3307 "Floating-Point Maximum/Minimum Function Objects"
and encouraged further work. However, if we could pass function templates/function overload sets easily that problem wouldn't even need solving.

Thanks to the minute takers Guy and David!

SG7: Compile-time programming (Dusíková/Vandevoorde)

- We decided on unibrow syntax ^^ for reflection P3381 (forwarded to EWG)
- We decided on private/protected members should be available with reflection (forwarded to EWG)
- On request from EWG we saw P2830, discussed it and gave our recommendation which was overruled by EWG later :)
- We discussed P3385 (attributes on reflection) and encouraged more work.

- We forwarded P3394 (annotations for reflection) to EWG
- We didn't see P0707 (metaclasses functions for generative C++) and we will see it on a telecon which will be announced on EWG and SG7 reflector soon
- We saw P3420 (reflection of templates) and unanimously encouraged more work for Andrei.
- We saw P3435 (reflection and meta-programming) and discussed differences between fragments and token sequences. Didn't get consensus to encourage more work.
- We saw P3334 (cross static variables) and decided not to pursue the problem.

SG9: Ranges (Hollman/Müller)

SG9 met in Wrocław on Monday and Tuesday. We also had a special session with weak quorum on Wednesday morning due to a last-minute room availability opportunity. We forwarded five papers to LEWG—two for parallel range algorithms and three for SIMD. We requested revisions of three other papers, including one on range type-erasure, one on a generalized checked-access (at) mechanism, and one on a range adaptor for lazy scan. We also provide informal feedback to a paper on the ranges-related features of the Unicode in the Standard Library paper targeting C++29.

SG10: Feature test (Revzin/Wakely)

No report. Everyone is doing a good job at keeping up with feature test macros.

SG14: Games & low latency (Wong)

SG14 did not meet, but will continue monthly meetings.

SG15: Tooling (Spencer/Boeckel)

SG15 met for a day and a half this week with between 7 and 10 people. We saw 6 papers and had 1 informal discussion, and forwarded 4.

The primary thing we handled this week is the Ecosystem IS ([P3342r0 - Working Draft, Standard for C++ Ecosystem](#)) which we reviewed and verified that it's ready for EWG to direct the convener to create a new work item.

We also discussed:

Forwarded:

P3081r0 - Core safety Profiles: Specification, adoptability, and impact - We had no concerns with the IS specifying fixits, some of which would be automatically applied.

P2758r3 - Emitting messages at compile time - No concerns with user defined diagnostic tags.

P3335r2 - Structured Core Options - We liked the initial set of portable options and forwarded for the Ecosystem IS.

Not forwarded:

P3470r0 - Interface-Unit-Only Module Library Support - We believe this solves the stated problem, but we don't really want people to do this, so we are not pursuing this unless there is strong user demand. Implementations are of course free to support it as an extension.

Discussions:

P3321r0 - Contracts Interaction With Tooling - Discussed various implementation strategies and agreed on a few things that would be good to cover in the Ecosystem IS.

Compiler Private Directory As posted on the reflector - Discussed how it would be useful for the compiler driver to be given a path to a specific directory for a build to do compiler private things that are scoped to a build.

SG16: Unicode (Honermann/Downey)

SG16 did not meet during the Wrocław meeting and has no further meetings planned for the calendar year. We'll resume meeting in the new year at our regular twice a month cadence. We continue to have plenty to work on.

SG17: EWG Incubator (Keane/Touton)

EWGI discussed 10 papers over the entire Friday of the meeting with the intent of preparing them for presentation in EWG. Of the 10:

4 Were determined to have sufficient motivation and paper readiness to be seen by EWG, and were forwarded. All were given feedback on how to improve presentation for EWG.

4 Were given extensive feedback on how to improve the paper motivation, presentation, and contents, and had sufficient consensus to return to EWG.

Of those, 1 had very weak consensus to return.

2 were discussed extensively, and after discussion, determined to have insufficient motivation to continue or be seen again, but of course are welcome to come back with improved motivation.

- [P3407R0](#) : [Make idiomatic usage of offsetof well-defined](#) : Forwarded to EWG
- [P3423R0](#) : [Extending User-Generated Diagnostic Messages](#) : Forwarded to EWG
- [P2998R0](#) : [Deducing function parameter types using alias template CTAD](#) : Feedback + Return
- [P3469R0](#) : [Virtual deducing this](#) Feedback + Return
- [P3405R0](#) : [Out-of-order designated initializers](#) Feedback + weak Return
- [P3245R2](#) : [Allow \[\[nodiscard\]\] in type alias declarations](#) No Consensus to Continue
- [P3347R0](#) : [Invalid/Prospective Pointer Operations](#) Forwarded to EWG
- [P3398R0](#) : [User specified type decay](#) No Consensus to Continue
- [P3412R0](#) : [String interpolation](#) Feedback + return
- [P3492R0](#) : [Sized deallocation for placement new](#) Forward to EWG

SG18: LEWG Incubator (Baker/Liber)

We met on Tuesday morning and Friday morning. We got through three papers:

[P2879R0](#) proposal of std::dump - we were strongly neutral on it (0-1-8-1-1), and the author has decided not to pursue it.

[P3383R0](#) [mdspan.at](#) - forwarding to LEWG (4-6-0-1-0).

The rest of the time was spent in a joint session with SG6 (Numerics) to continue making progress on [P3045](#) Quantities and units library.

We've maintained our perfect record of never running late.

SG19: Machine Learning (Wong/Reverdy)

SG19 did not meet, but will continue monthly meetings.

SG20: Education (van Winkel/Sattler)

We met Friday Afternoon.

SG20 is focussing on education of C++.

As part of that we are creating C++ teaching guidelines, where we mention the topics that should be taught in different kinds of educational settings. We are making progress in growing our body of materials, but we could use some more help with this. If you are interested in authoring some teaching guidelines for specific topics, please reach out to us.

We would like to point out that we are also there to advise paper authors on the teachability of their new features. So we encourage paper authors and workgroup chairs to send papers to SG20 for teachability feedback.

Finally, we are working on creating new ways of consuming our information, e.g., updating our web page and allowing people to navigate our materials in different ways.

SG21: Contracts (Spicer/Doumler)

SG21 (Contracts) met all day Friday in Wrocław.

First, we adopted fixes for four issues that were flagged during CWG wording review of P2900 ([P3520R0](#)). We also adopted a fix to P2900 that addresses the issue that preconditions in

constructors and postconditions in destructors may use the value of data members that are not yet initialised or already destroyed, respectively ([P3510R1](#)).

Then, we saw [P3100R1](#), which proposes a framework for safety in C++ that unifies the notion of contract violations, erroneous behaviour, and undefined behaviour. We had very strong consensus to pursue this direction for post-C++26, and for the short term, to align the terminology and conforming failure modes of erroneous behaviour with that of contract-violation handling for C++26 to pave the way for that direction.

Finally, we saw two papers that propose to treat other scenarios as contract violations, in particular the situation where an exception hits a noexcept boundary and other exception-handling related failures occur that currently terminate the program ([P3205R0](#)) and a new "throws_nothing" attribute (or context-sensitive keyword) that would allow treating an exception escaping as a contract violation without affecting the noexcept-ness of a function ([P2946R1](#)). Both papers received feedback and will need to come back with a revision.

SG22: C/C++ Liaison (Ranns/Herring,Meneide(for WG14))

SG22 has not met this week.

SG23: Safety/Security (Orr/Craig)

SG23 met for 2 1/2 days in Wrocław - thank you to those who were involved in changing the scheduling to make this possible.

We saw fourteen papers.

We had consensus for other groups to proceed with:

- P2719R1 "Type-aware allocation and deallocation functions" - EWG wanted feedback; we had consensus the paper should move forward
- P3352R0 Taming the Demons (C++ version) - Undefined Behavior and Partial Program Correctness and P3403R0 The Undefined Behavior Question and P1494R4 Partial program correctness
 - SG23 wished to protect volatile observable behavior from "time travel" optimization
 - SG23 asked EWG to consider P3352R0 as an alternative to P1494R4
- P3081R0 Core safety Profiles: Specification, adoptability, and impact Forwarded to EWG for C++26
- P3471R0 Standard library hardening Forwarded to EWG/LEWG
- P3404R0 std::at : Range-checked accesses to arbitrary containers Forwarded to SG9

We had consensus for more work for:

- P3402R1 A Safety Profile Verifying Class Initialization
- P3436R0 Strategy for removing safety-related undefined behavior by default

- P3446R0 Profile invalidation - eliminating dangling pointers
SG23 would like the approach of P3436 "Strategy for removing safety-related undefined behavior by default" to integrate with the Profiles approach in P3446
- P3465R0 Pursue P1179 as a Lifetime TS
SG23 requested the author of P3465R0 "Pursue P1179 as a Lifetime", to produce a working paper for a TS or a white paper.
- P3447R0 Profiles syntax
- P3442R0 `[[invalidate_dereferencing]]` attribute
We want to see a redraft considering interaction with P3465 "Pursue P1179 as a Lifetime Safety TS"

We looked at P3390R0 Safe C++

Unfortunately at present the original author is not in a position to continue with his proposal.

POLL: Which should we prioritize: Profiles or Safe C++ ?

Profiles/Both/Neutral/Safe C++: 19/11/6/9

ABI Group (Vandevoorde)

ABI group did not meet.

Admin (Liber)

We had approximately 149 attendees face-to-face and 73 virtual.

The next mailing will be on December 17th.

The pre-Hagenberg mailing is scheduled for January 13th, which is exactly 4 weeks before the meeting.

Evolution (Bastien/Keane/Dusikova)

This week was dedicated to papers which might plausibly affect C++26.

- Contracts:
 - P2900r11 contracts were forwarded to LEWG and CWG for inclusion in C++26. They are expected to go to plenary in February 2025 in Hagenberg, Austria.
 - Added support for contracts on coroutines as proposed in [P2957R2](#)
 - No added support for contracts on function pointers as explained in [P3327r0](#)
 - No added support for stricter contracts (in addition to the current default relaxed contracts) as suggested in [P3362R0](#) / [P2680](#) / [P3285](#), and keep relaxed the default based on the discussion in [P3499](#)
 - const-ification remains in the contracts proposal, see discussion in [P3261R1](#) / [P3478R0](#)
 - Adopted a few fixes from CWG/SG21, see [P3520r0 Contracts for C++: Wrocław technical fixes](#)

- Adopted [P3510r1 Leftover properties of this in constructor preconditions](#)
- Reflection:
 - SG7 decided to use ^^ (unibrow) syntax for reflection, based on [P3381 — Syntax for Reflection](#)
 - Papers about the accessibility of members through reflection, which we discussed and decided to maintain the P2996 status quo, whereby members which are regularly inaccessible are all accessible through reflection:
 - [P3437 — Proposed default principles: Reflect C++, Generate C++](#)
 - [P3451 — A Suggestion for Reflection Access Control](#)
 - [P3473 — Splicing Should Respect Access Control](#)
 - [P3493 — Syntax for Reflection — Ruminations on reflection and access](#)
 - [P3394 — Annotations for Reflection —](#) forwarded to CWG
 - [P3289 — Consteval blocks —](#) forwarded to CWG
 - [P3273 — Introspection of Closure Types —](#) no consensus
 - [P3293 — Splicing a base class subobject —](#) forwarded to CWG and LEWG
 - [P2830 — Standardized Constexpr Type Ordering —](#) we chose the implementation-defined ordering, and forwarded to LEWG
- Pattern matching:
 - [P2688 — Pattern Matching: `match` Expression —](#) encouraged to come back
 - [P2392 — Pattern matching using is and as —](#) no consensus
 - [P3332 — A simpler notation for PM —](#) no consensus
- Safety related
 - [P3081 — Core safety Profiles: Specification, adoptability, and impact —](#) strong consensus, will be designing profiles over telecons
 - [P3471 — Standard library hardening —](#) unanimously sent to LEWG, will align with contracts / profiles
 - [P2719 — Type-aware allocation and deallocation functions —](#) sent to CWG
- [P1061 — Structured Bindings can introduce a Pack —](#) what a roller coaster! On the straw polls
- [P3176 — The Oxford variadic comma —](#) On the straw polls
- [P3422 — Allow main function in named modules —](#) forwarded, but problems were found, pulled from the straw polls
- [P2786 — Trivial Relocatability For C++26 —](#) Forwarded to LEWG and CWG for inclusion in C++26, but extremely tight consensus
- [P3367 — constexpr coroutines —](#) Forwarded to CWG for inclusion in C++26, but CWG is worried about implementation burden
- [P2866 — Remove Deprecated Volatile Features From C++26 —](#) forwarded to CWG
- [P3074 — trivial unions \(was std::uninitialized\) —](#) forwarded to CWG and LEWG
- [P3477 — There are exactly 8 bits in a byte —](#) forwarded to CWG and LEWG
- [P3421 — Consteval destructors —](#) forwarded to CWG, but lots of neutral, consider life choices
- [P3380 — Extending support for class types as non-type template parameters —](#) sent to LEWG to confirm usage and help the design
- [P3006 — Launder less —](#) forward to CWG
- [P2414 — Pointer lifetime-end zap proposed solutions —](#) combined with angelic non-determinism, use the new feature to make some concurrency easier, but point out to SG1 that they were silly and forgot some things
- [P3439 — Chained comparisons: Safe, correct, efficient —](#) More work encouraged
- [P3324 — Attributes for namespace aliases, template parameters, and lambda captures —](#) More work encouraged

- [CWG2752](#) — [Excess-precision floating-point literals](#), see [P3488 Floating-Point Excess Precision](#) — no consensus based on the SG6 strong recommendations, need to see again
- [P1494](#) — [Partial program correctness](#) — [P3352 \(github\)](#) and [P3403 \(github\)](#) — Davis' original paper remains the status quo, volatile was not added to the list of observable effects
- [P3475](#) — [Defang and deprecate memory_order::consume](#) — defanged, it is gone
- [P0876](#) — [fiber_context - fibers without scheduler](#) — waiting for input on implementability on Windows
- [P3466](#) — [\(Re\)affirm design principles for future C++ evolution](#) — update, and adopt as a standing document

Library Evolution (Levi/Fracassi/Craig)

LEWG has met throughout the week, prioritizing large topics aiming for C++26, and saw ~30 papers.

Due to the acceptance of "[P3399R0: Adjusting Electronic Polls to a Hybrid Workflow](#)" (approved before the meeting), we did not take electronic polls for the papers but forwarded them directly to LWG.

Detailed minutes are on our Wroclaw page on the Wiki (status on GitHub):

<https://wiki.edg.com/bin/view/Wg21wroclaw2024/LibraryEvolutionWorkingGroup>

The four major topics LEWG discussed during the week were:

- Reflection (fixes)
- Sender Receiver (fixes)
- Relocatability
- Contracts

Papers' status

Forwarded:

- P2900R10: Contracts for C++
- P2996R11: Reflection for C++26 (finalized remaining traits)
- P3299R2: Range constructors for std::simd
- P3138R3: views::cache_latest
- P3367R0: constexpr coroutines
- P3449R0: constexpr std::generator
- P3284R1: `finally`, `write_env`, and `unstoppable` Sender Adaptors
- P3449R0: constexpr std::generator
- P3367R0: constexpr coroutines
- P2434R2: Nondeterministic pointer provenance
- P3152R0: Add missing constructors and assignment for indirect and polymorphic
- P2830R4: Standardized Constexpr Type Ordering
- P3074R4: trivial unions (was `std::uninitialized<T>`)
- P2846R4: reserve_hint: Eagerly reserving memory for not-quite-sized lazy ranges

Design Approved:

- P3149R6: async_scope

- P2319R2: Prevent path presentation problems
- P3425R0: Reducing operation-state sizes for subobject child operations
- P3433R0: Allocator Support for Operation States

Feedback Given:

- P3148R0: Formatting of chrono Time Values
- P2945R1: Additional format specifiers for time_point
- P0260R11: C++ Concurrent Queues
- P1030R7: std::filesystem::path_view
- P2645R0: path_view: a design that took a wrong turn
- P3373R0: Of Operation States and Their Lifetimes
- P3388R0: When Do You Know connect Doesn't Throw?
- P3096R3: Function Parameter Reflection in Reflection for C++26
- P2769R0: get_element customization point object
- P3430R0: simd issues: explicit, unsequenced, identity-element position, and members of disabled simd
- P2527R3: std::variant_alternative_index and std::tuple_element_index
- P3094R5: std::basic_fixed_string

Rejected:

- P3415R0: Range interface in std::optional breaks code!
- P3429R0: Reflection header should minimize standard library dependencies (apart from a small bit adopted)

LEWG will continue running Telecons (one Telecon scheduled for Dec 10th, then we will break till Jan 2025) till the Austria meeting, to continue processing C++26 topics and fixes.

Agenda updates here:

- <https://github.com/cplusplus/LEWG/wiki/2024-Telecons>
- <https://github.com/cplusplus/LEWG/wiki/2025-Telecons>

Thank you to all the authors and reviewers for the great work promoting the quality of the standard library,

And especially to our co-chairs and minute takers, helping manage our significant workload: Fabio, Robert, Khalil, Eddie, Steve, Guy.

Looking forward to seeing you in Hagenberg, Austria!

Core (Maurer/Merrill/Caves)

We met every day this week. We spent a total of at least 3 days on Reflection and Contracts. Both are making progress.

There were small adjustments voted in for Contracts in Evolution, which makes reviewing contracts a moving target for us. We shouldn't do so moving forward. There is no problem in reviewing add-on papers, but it makes it difficult when it's added to the standing document revision.

Processing of reflection, contracts, and pattern matching papers will be prioritised for C++26. We believe this is in harmony with P0592R5 and P2000R4. That means other papers forwarded to core may not make it through core in time. If you want to change that, it's the usual matter of prioritisation.

We readied 31 core issues this week.

Concerns were expressed in core about the implementability of co-routines for constant evaluation. People don't think it's impossible to do, but it might be complex. Constexpr co-routines were sent back to Evolution for further consideration.

One of the papers on the straw polls was withdrawn.

Herb Sutter : Does anyone in the room object to removing the poll 9 : Accept as a Defect Report and apply the changes in [P3422R1](#) (Allow main function in named modules) to the C++ Working Paper?

No objections.

I would like to thank Brian and Jason for scribing this week. We are going to have a telecon on Friday 13th of December.

We will have an afternoon session today until 4pm.

CWG polls

1. Accept as Defect Reports and apply the proposed resolutions of all issues in [P3524R0](#) (Core Language Working Group "ready" Issues for the November, 2024 meeting) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

2. Apply the changes in [P3340R0](#) (A Consistent Grammar for Sequences) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

3. Apply the changes in [P2686R5](#) (constexpr structured bindings and references to constexpr variables) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

4. Apply the changes in [P3068R6](#) (Allowing exception throwing in constant-evaluation) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes.

5. Apply the changes in [P3247R2](#) (Deprecate the notion of trivial types) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes.

6. Apply the changes in [P2865R6](#) (Remove Deprecated Array Comparisons from C++26) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes.

7. Apply the changes in [P1061R10](#) (Structured Bindings can introduce a Pack) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes.

8. Apply the changes in [P3176R1](#) (The Oxford variadic comma) to the C++ Working Paper.

No discussion.
Objections in the room.

Herb Sutter clarifies the voting rules.

In favour : 78 (59 in person + 19 online)
Opposed : 1 (1 in person + 0 online)
Abstain : 16 (9 in person + 7 online)

Motion passes.

Library (Wakely/Garland/Kuhl)

Library had a productive week - we reviewed about 20 papers and are moving 19. Four of the moved papers were ready due to telecom before the meeting. It was particularly nice to have all the LWG chair present face-to-face. A first since the pandemic.

Notable papers we're moving include the base SIMD paper (P1928) -- with range construction fixes applied directly to this paper for the ease of the editor. Thanks to Thomasz, Christian, Ruslan and Matthias for major work this week to pull the wording over the finish line. Also like to

call out the LEWG chairs for being flexible and scheduling to address LWG issues -- and timely work on the SIMD interfaces so we could ship a solid base product.

Another paper of note is P3019 which introduces indirect and polymorphic types to facilitate composite class design. This is another paper partially delayed by the pandemic which will now make it into the hands of c++ developers.

Although not 100% complete, we spent significant time on P0447 `std::hive` - a novel type of sequence container. We should complete work on this in telecom before Hagenberg. We also looked at one of the SIMD follow-on papers and expect that to complete quickly in our next meeting.

LWG polls

1. Apply the changes for all Ready and Tentatively Ready issues in [P3504R0](#) (C++ Standard Library Ready Issues to be moved in Wrocław, Nov. 2024) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

2. Apply the changes in [P3136R1](#) (Retiring niebloids) to the C++ working paper.

No discussion.

Objections in the room.

In favour : 68 (52 in person + 16 online)

Opposed : 2 (1 in person + 1 online)

Abstain : 21 (11 in person + 10 online)

Motion passes.

3. Apply the changes in [P3138R5](#) (`views::cache_latest`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

4. Apply the changes in [P3379R0](#) (Constrain `std::expected` equality operators) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

5. Apply the changes in [P0472R2](#) (Put `std::monostate` in `utility`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

6. Apply the changes in P2862R1 (`text_encoding::name()` should never return null values) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

7. Apply the changes in P2897R7 (`aligned_accessor`: An `mdspan` accessor expressing pointer over-alignment) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

8. Apply the changes in P3355R1 (Fix `submdspan` for C++26) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

9. Apply the changes in P3222R0 (Fix C++26 by adding transposed special cases for P2642 layouts) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

10. Apply the changes in P3050R2 (Fix C++26 by optimizing `linalg::conjugated` for noncomplex value types) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

11. Apply the changes in P3396R1 (`std::execution` wording fixes) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

12. Apply the changes in P2835R7 (Expose `std::atomic_ref`'s object address) to the C++ working paper.

No discussion.

Objections in the room.

In favour : 41 (31 in person + 10 online)

Opposed : 6 (4 in person + 2 online)

Abstain : 42 (28 in person + 14 online)

Herb Sutter : The objections are from senior people. Do we have any NB body objections ?

There was a concern about an aspect of the paper within BSI. It is not a national body position.

No NB objections in the room.

Motion passes.

Herb Sutter : Please write papers if you want to remove this feature. If you have an NB body objection, let me know.

13. Apply the changes in P3323R1 (cv-qualified types in atomic and atomic_ref) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

14. Apply the changes in P3508R0 (Wording for "constexpr for specialized memory algorithms") and P3369R0 (constexpr for uninitialized_default_construct) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

15. Apply the changes in P3370R1 (Add new library headers from C23) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

16. Apply the changes in P3309R3 (constexpr atomic and atomic_ref) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

17. Apply the changes in P3019R11 (indirect and polymorphic: Vocabulary Types for Composite Class Design) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

18. Apply the changes in P1928R15 (std::simd — merge data-parallel types from the Parallelism TS 2) to the C++ working paper.

No discussion.
Objections in the room.

In favour : 60 (46 in person + 14 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 27 (17 in person + 10 online)

Motion passes.

19. Apply the changes in [P3325R5](#) (A Utility for Creating Execution Environments) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes.

Direction Group (Hinnant)

In Wroclaw, the Direction group met Wednesday. Usually, we meet once every two weeks.

We consider rapid progress on safety essential given the external deadlines imposed upon our industry as well as long-term goals for C++. By “rapid” we mean “for C++26, deployment, and plans beyond C++26.” Progress is being made on profiles, hardening, erroneous behavior, and contracts. Each makes contributions to improving safety. We encourage people working on these approaches to try to ensure that the result is a coherent whole that can help novices as well as experts.

We are concerned that SGs can get trapped into a too narrow focus. We encourage SGs to consider the impact of their designs on the language as a whole. In particular, to consider use and teachability as part of the design process, rather than an afterthought to language technical concerns. To judge “teachability” try to get input from people with solid teaching experience. Experience with explaining a design to fellow experts can be misleading.

We noted progress on reflection and executors, but didn’t discuss them in detail.

We consider pattern matching important and worry about the lack of usage experience and implementation on the P2688 pattern matching proposal.

The number of papers is very large and the number of reflector messages for some groups are enormous. This unfortunately narrows the number of people who are able to keep up-to-date, participate, and provide informed votes.

The CWG and LWG are overloaded. This is becoming a bottleneck for proposals.

We encourage everyone to maintain a courteous and professional tone of discussions.

8. Closing activities

8.1 Issues delayed until today

Herb Sutter : We are working on moving to having a per person access to the wiki. We also are going to change the wiki password in the next few days. We will send the details to everyone in the ISO global directory. We are also working on a more automated paper system. Monthly bucketing does not align well with meeting deadlines.

8.2 Mailings

Note: These are the closest regular mailings and not special pre/post meeting mailings.

- 2024-12-15: Post-Wroclaw
- 2025-01-15: Pre-Hagenberg

8.3 Plans for the future

No discussion.

8.4 Next and following meetings

- 2025-02-10/15: Hagenberg, Austria ([N4979](#))

Please note the new paper number. The paper has been updated to contain the mention of the Sunday orientation session.

Presentation from the host:

We welcome you to Hagenberg/Austria.

We are pleased that in February 2025, the WG21 committee will visit our small village, Hagenberg, located in a small country (but with a big history), Austria.

In concrete, the plenary will be hosted by the Upper Austria University of Applied Sciences, Campus Hagenberg.

In case there are any problems, e. g. with your room reservations, your travel, confirmation documents for visas, please do not hesitate to contact us:

peter.kulczycki@fh-hagenberg.at

michael.hava@risc-software.at

For all details, please see document N4997.

We welcome you to Hagenberg/Austria

Clarification: No, it will not be possible to ski in Hagenberg.

Justification: Although it is called Hagenberg (i. e., mountain), it is more of a hill, where Hagenberg is located.

- 2025-06-16/21: Sofia, Bulgaria ([N4991](#))
- 2025-11-03/08: Kona, HI, USA;

9. Adjournment

Meeting adjourned at 11:03 AM UTC+1.

10. Attendance

| Attendee | NB |
|-------------------------|------|
| Adams Michael | SCC |
| Adelstein Lelbach Bryce | ANSI |
| Alday Juan | ANSI |
| Alexandrescu Andrei | ANSI |
| Arkipova Olga | ANSI |
| Arutyunyan Ruslan | ANSI |
| Ažman Gašper | BSI |
| Baker Billy | ANSI |
| Baker Lewis | ANSI |
| Balog Pal | ANSI |
| Basith Shanawaz | ANSI |
| Bastien Jean-Francois | SCC |
| Benetkiewicz Paweł | PKN |
| Bentley Matthew | ANSI |
| Berge' Agustin | UNI |
| Bernat Yehezkel | SII |
| Berne Joshua | ANSI |
| Beyls Kristof | BSI |
| Bi Brian | ANSI |
| Bindels P.G.H. | NEN |
| Birbacher Frank | ANSI |
| Blackwell Bianca | SCC |
| Boeckel Ben | ANSI |

| Attendee | NB |
|--------------------------------------|---------|
| Boehm Hans | ANSI |
| Bonaventura Xavier | DIN |
| Brown Bret | ANSI |
| Brown Walter E. | SII |
| Büttner Sebastian | ANSI |
| Cardoso de Souza Rodrigues Guilherme | ASI |
| Cassagnes Aurelien | ANSI |
| Caves Jonathan | ANSI |
| Chen Yuxuan | ANSI |
| Childers Wyatt | ANSI |
| Chochlik Matus | UNMS SR |
| Christensen Alex | ANSI |
| Coe Jonathan Brian | BSI |
| Craig Benjamin | ANSI |
| Craig Philip | BSI |
| Cranmer Joshua | ANSI |
| D'Angelo Giuseppe | ANSI |
| Davidson Guy | BSI |
| de Wever Mark | ANSI |
| Delfino Gianluca | UNI |
| Dionne Louis | SCC |
| Dominiak Michał | PKN |
| Dos Reis Gabriel | AFNOR |
| Douglas Niall | NSAI |
| Doumler Timur | BSI |
| Downey Steve | ANSI |
| Dusikova Hana | UNMZ |
| Dziubinski Matt | ANSI |
| Engert Daniela | DIN |
| Estell Khalil | ANSI |
| Fertig Andreas | DIN |
| Fevold Jake | ANSI |
| Floyd Paul | ANSI |
| Fracassi Fabio | DIN |

| Attendee | NB |
|----------------------------|-------|
| García Sánchez José Daniel | UNE |
| Garland Jeff | ANSI |
| Genovese Walter | ANSI |
| Gill Mungo | NSAI |
| Giroux Olivier | ANSI |
| Goodspeed Nathaniel | ANSI |
| Gordon Fraser | SCC |
| Gozillon Andrew | ANSI |
| Gruber Bernhard | ANSI |
| Gustafsson Bengt | SIS |
| Halpern Pablo | ANSI |
| Hava Michael Florian | ASI |
| Herring Davis | ANSI |
| Hoemmen Mark | ANSI |
| Honermann Tom | ANSI |
| Hughes Lori | ANSI |
| Hunt Oliver | ANSI |
| Izvekov Matheus | ANSI |
| Jabot Corentin | AFNOR |
| Jha Dheeraj | BIS |
| Josuttis Nicolai | DIN |
| KAMINISKI Tomasz | AFNOR |
| Keane Erich | ANSI |
| Khlebnikov Rostislav | ANSI |
| Koeppe Thomas | ANSI |
| Kostur Andre | SCC |
| Kosunen Elias | SFS |
| Kretz Matthias | DIN |
| Krzemienski Andrzej | PKN |
| Kuhl Dietmar | ANSI |
| Kulczycki Peter | ASI |
| Lach Adam | SNV |
| Lakos John | ANSI |
| Lapkowski Christopher | ANSI |
| Lauko Henrich | UNMZ |

| Attendee | NB |
|--------------------------------|------|
| Laverdière-Papineau Marc-André | ANSI |
| Leahy Robert | SCC |
| Lebrun-Grandie Damien | ANSI |
| Levi Inbal | SII |
| Levine Michael | ANSI |
| Li Yihe | ANSI |
| Liber Nevin | ANSI |
| Machutova Jana | UNMZ |
| Maness Wesley | ANSI |
| Marr Greg | ANSI |
| Maurer Jens | ANSI |
| McDougall Ryan | ANSI |
| McKenney Paul | ANSI |
| McMonagle John | BSI |
| Meerwald Christof | ASI |
| Mejstrik Thomas | ASI |
| Meredith Alisdair | ANSI |
| Merrill Jason | ANSI |
| Michael Maged | ANSI |
| Miller Cody | ANSI |
| Morales Nicolas | ANSI |
| Moschovakos Paris | SNV |
| Mueller Gideon | ANSI |
| Müller Jonathan | DIN |
| Nash Phil | BSI |
| Neatu Darius | ANSI |
| Niebler Eric | ANSI |
| Nishanov Gor | ANSI |
| Nolan Edward | ANSI |
| O'Dwyer Arthur | ANSI |
| Olsen David | ANSI |
| Opara Jolanta | PKN |
| Orr Roger | BSI |
| Owen Nathan | ANSI |
| Park Michael | SCC |

| Attendee | NB |
|------------------------------|---------------------|
| Peacock Antony | BSI |
| Persson Jonas | SIS |
| Petersen Ian | ANSI |
| Polukhin Anton | GOST R |
| Preney Paul | SCC |
| Pusz Mateusz | PKN |
| Ranns Nina Dinka | BSI |
| Regev Ran | SII |
| Revzin Barry | ANSI |
| Rigault Jean-Paul | AFNOR |
| Rivera Morell René Ferdinand | ANSI |
| Ronkainen Jari | SFS |
| Rosten Oliver | BSI |
| Roy Patrice | SCC |
| Sandoe Iain | BSI |
| Sankel David | ANSI |
| Satle Ankur | BIS |
| Sattler Florian | ANSI |
| Sharma Saksham | ANSI |
| Snyder Jeff | BSI |
| Song Tim | ANSI |
| Spencer Michael | ANSI |
| Spicer John | ANSI |
| St. Amour Bryan | SCC |
| Stroustrup Bjarne | ANSI |
| Sutter Herb | ISO/IEC JTC 1/SC 22 |
| Szuppe Jakub | ANSI |
| Talbot Alan | ANSI |
| Taylor Matthew | BSI |
| Tenty David | SCC |
| TEODORESCU Lucian Radu | ASRO |
| Teoh Joon Nam | ANSI |
| Tomazos Andrew | ANSI |
| Tong Hubert | SCC |
| Touton James | ANSI |

| Attendee | NB |
|--------------------------|--------|
| Towner Daniel | ANSI |
| Trott Christian | ANSI |
| Tsaousis-Seiras Isidoros | ANSI |
| van Winkel J.C. | NEN |
| Vandevoorde Daveed | ANSI |
| Varlamov Konstantin | ANSI |
| Vasama Lauri | SFS |
| Voicu Alexandru | ANSI |
| Vollmann Detlef | SNV |
| Vormwald Steven | ANSI |
| Voss Michael | ANSI |
| Voutilainen Ville | SFS |
| Wakely Jonathan | ANSI |
| Walker Kelly | ANSI |
| Waterloo Jarrad | ANSI |
| Weis Andreas | DIN |
| Williams Anthony | BSI |
| Williamson Gerald | ANSI |
| Wong Jessica | ANSI |
| Wong Michael | SCC |
| Xie Hui | BSI |
| xu chuanqi | SAC |
| Yaghmour Shafik | ANSI |
| Yuan Zhihao | ANSI |
| Zielonka Bartosz | PKN |
| Zissu Andrei | SII |
| Zverovich Victor | ANSI |
| Anisimov Sergei | GOST R |
| Bagdonas Ignas | BSI |
| Banglawala Neelofer | BSI |
| Brand André | DIN |
| Fisilier Eric | |
| Hollman Daisy | ANSI |
| Jiaming Liang | SAC |
| Kawulak Robert | PKN |

| Attendee | NB |
|--------------------|------|
| Khyzha Artem | BSI |
| Kowalski Maciej | PKN |
| Kutlov Tymofii | |
| Labor Malia | ANSI |
| Larson Brad | |
| Lippincott Lisa | |
| Myers Nathan | |
| Nadolski Maikel | |
| Narolewski Jakub | PKN |
| Nyhammer Henrik | SN |
| Panasiuk Jakub | PKN |
| Petukhou Maksim | PKN |
| Skowron Zbigniew | |
| Starosz Sebastian | PKN |
| Szalkowski Mateusz | PKN |
| Szolnoki Lénárd | BSI |
| Wilmans Jan | |
| Wojakowski Łukasz | PKN |
| Zych Mateusz | DIN |