WG21 June 2025 Sofia Hybrid meeting
Minutes of Meeting

ISO/IEC JTC1 SC22 WG21 N5016— 2024-07-07
Nina Dinka Ranns, dinka.ranns_at_gmail.com

Chair: John Spicer

16-21 June 2025, Sofia, Bulgaria

1. Opening activities

John Spicer opens the meeting at 09:02 AM UTC+2.

The wifi information is on the screen.

1.1 Opening comments, welcome from host

John Spicer welcomes the group.

Welcome from the host.
Thank you to the host.
Thank you to the sponsors.

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. If you are attending for the first time, please see the page for new committee members.

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. If you have any issues, let me know. Next mailing will be July the 15th.

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

Additional, lower-priority goals include:
Subgroup work that may target versions after C++26

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	N5009	N5008

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

1.6 Approval of the minutes of the previous meetings

Meeting	Minutes
WG21 Hagenberg	N5007
WG21 pre-Sofia administrative telecon	N5012

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

No new business.

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

Jens Maurer presents room assignments.

Jens Maurer presents local amenities.

Room assignments are on the wiki page. Any questions or issues with remote attending setup, please find Jens Maurer.

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. If the poll refers to the new paper that hasn't been in the mailing, it should be attached to the straw polls page. 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.

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.

SG1: Concurrency (Giroux/Boehm/Arutyunyan)

SG1 is still low on work, we had a short meeting. Please send us work.
Highlight : student paper P3703 - motivates us to revisit the ergonomics of types with locks.
Sometimes there are papers that bypass us. Resist (and help others resist) the urge to send papers straight to LEWG.

C++26 is massive for us, but still left a lot of work to do.

Let’s prime this pipeline in Kona!

We would like to see :
* Ergonomics improvement
* Algorithms, algorithms, algorithms
* Shared/unshared/remote memory
* Memory model interoperability
* Execution agents clarification
* Further cooperative progress

SG4: Networking (Snyder/Ažman)

SG4 did not meet this week.

SG6: Numerics (Kretz/Lippincott/McFarlane)

SG6 met on Thursday morning to discuss a trait on value-preserving conversions and integer-related papers. On Friday afternoon SG6 considered all floating-point papers.
We forwarded

* P3605R0 "isqrt: A function to calculate integer square root of the nonnegative integer"

* P3642R0 "Carry-less product: std::clmul"

* P3639R0 "The _BitInt Debate"

* P3565R1 "Virtual floating-point values"

SG6 continued to review P3045R5 "Quantities and units library" together with SG18 (LEWGI), which happened almost exclusively on SG18 time.

SG7: Compile-time programming (Dusíková/Snyder)

We met on Monday after plenary. We saw two papers P3678R0 "arbitrary attribute in define_aggregate" and P3603R0 "consteval-only values and consteval variables". We forwarded both to EWG where neither of them got consensus to being forwarded to C++29.

SG9: Ranges (Müller/Laine)

SG9 met in Sofia on Monday afternoon and Wednesday morning. We have seen 7 papers; we did not have time to see one issue and one late paper.

We forwarded 4 papers to LEWG:

* P3709R1 — Reconsider parallel ranges::rotate_copy and ranges::reverse_copy, a C++26 fix for the return type of the upcoming parallel version of `ranges::rotate_copy` and `ranges::reverse_copy`.

* P3544R0 — ranges::to View, a C++23 defect report for the constraints of `ranges::to`.

* P3725R0 — Filter View Extensions for Input Ranges, a new simpler alternative to `views::filter` that removes pitfalls for beginners. We expect NB comments requesting it as a late addition to C++26. Thank you to Nico for not giving up.

* P3705R1 — A Sentinel for Null-terminated Strings, adding a facility to easily create a range out of a null-terminated C string.

We gave feedback on a new revision for P3411R3 — any_view. We have consensus on the final design and expect to forward it next meeting. We also saw the first revision of P3220R0 - views::delimit. We renamed it to `views::take_before` and will see it again.

We have rejected P3431R0 — Deprecate const-qualifier on begin/end of views. We will thus keep the conditional const qualifiers on views and continue adding it in the future.

SG10: Feature test (Revzin/Wakely)

SG10 did not meet.

SG14: Games & low latency (Wong)

We continue to meet every month online. We did not meet this week. We have new co-chairs : John McFarlane, Bryan St. Amour, Guy Davidson, and Patrice Roy

We continue to look at papers on low latency constrained resource access in games. We also have a paper on exceptions.

SG15: Tooling (Spencer/Boeckel)

The Tooling Study Group met for one session to discuss a paper (P3696R0) on a portable way to declare header units based on experience from Clang modules. This direction was approved and we expect to see an updated paper next meeting

SG16: Unicode (Honermann/Downey)

SG16 did not meet in Sofia. We continue to hold telecons approximately twice monthly and have plenty of interesting work ahead of us.

SG17: EWG Incubator (Keane/Touton)

Despite being scheduled, SG17 EWG-Incubator did not meet this week. Due to EWG's progress, chair availability, and paper availability, the Incubator opted to have members instead stay in EWG. No papers were discussed, however Evolution saw every paper that SG17 had scheduled this week.

SG18: LEWG Incubator (Baker/Liber)

We met Wednesday and Thursday afternoons. We should finish P3045 Quantities and units during the next meeting. 4 other papers were discussed. Three (P3086 R4 Proxy: A Pointer-Semantics-Based Polymorphism Library, P3086 P3161 Unified integer overflow arithmetic, and P3689 Convenience functions for Random number generation) would need revisions if they are to continue. P3546 Explicit return type deduction for std::numeric_limits and numbers was not viewed as a library problem.

Additionally, authors provided feedback on papers in the SG18 backlog.

No longer being pursued.

P1433 Compile Time Regular Expressions
P3178 R1 Retrieval of Exception Information
P3288 R3 std::elide
P2639 Static Allocations

Needs revisions.

P2906 R0 Structured bindings for std::extents
P3018 R0 Low-Level Integer Arithmetic
P3357 R0 NRVO with factory and after_factory
P3095 R0 ABI comparison with reflection

SG19: Machine Learning (Wong/Ratzloff)

We're looking at statistics to advance C++ for machine learning. We've been looking at graph algorithms. This enables vector databases. We continue to look at adding automatic differentiation. With the help of reflection, that will make it more capable.
We have a new chair - Phil Ratzloff.

SG20: Education (van Winkel/Sattler)

SG20 Education met on Thursday morning.

We discussed a timeline for a minimum viable product. We are aiming to have a first fully complete learning path published by the end of 2026. That is, a learning path where all topics are worked out.

We also discussed participation in our group. We need more people helping with the preparation of topic subjects. That is,writing content for new topics and also reviewing existing topics.
We would also like to follow progress in C++ more by having C++ paper editors assist us with the training side of it. These subject matter experts know the subject best and may have advice for us on what should be trained at several experience levels for their accepted paper. It would be even better if papers introducing (larger) features in the language would have some attention for how to train the feature in their paper. SG20 is willing to look at papers as they are in flight, please feel free to send them our way.

Finally, we discussed how to have people "unlearn" patterns from C and old C++. This starts with not teaching the undesigned feature, but also actively using the new way. Though we see some patterns are hard to avoid, e.g., when interfacing with Posix libraries.

SG21: Contracts (Spicer/Doumler)

SG21 has not met this week.

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

SG22 has not met this week.

SG23: Safety/Security (Orr/[open])

SG23 "Safety and Security" met on Tuesday and Friday.

We had presentations, and some discussion, on four papers:
P3498R0 "Stop the Bleeding but, First, Do No Harm"
P3649R0 "A principled approach to safety profiles"
P3704R0 "What are profiles?"
D3751R0 "A gentle introduction to pointer authentication"

We had presentations and polls on seven papers:

P3100R2 "Implicit contract assertions"
Consensus to spending more committee time on the paper

P3402R3 "A Safety Profile Verifying Initialization"
We had consensus to forward this proposal to EWG, after consideration of EWG discussion on Friday

P3640R0 "Make contracts reliable by default"
We had consensus against further work in this direction

P3655R2 "std::zstring_view"
Three direction polls were taken to help direct the paper author

P3700R0 "Making Safe C++ happen"
No polls were taken, the author was encouraged to do more work.
The author's intent is to eventually produce either a guidance paper or a standing document.

P3711R0 "Safer StringViewLike Functions for Replacing char* strings"
Consensus to spending more committee time on the paper
Some direction polls given to help direct the paper author

P3716R0 "Subsetting"
We would like to see P3716R0 worked out in detail

Thank you to all who attended, and to all those who broke the awkward silences by agreeing to be scribes.

ABI Group (Vandevoorde/Merril)

ABI group did not meet this week.

Admin (Liber)

There were 195 attendees representing 25 national bodies, and 129 of those were in person.

The post-Sofia mailing is scheduled for July 15th, and we will resume monthly mailings until the pre-Kona mailing, which is scheduled for October 6th (4 weeks before Kona).

Is there a timescale for moving to per user logon on wiki ?
There have been no updates since the last email. Most of it is done, but we're blocked on figuring out the attachment issue. We may have it in Kona.

We had 24 national bodies at the meeting.

Is the pre Kona mailing before or after NB submission ?
That depends on the NB. After we issue the CD ballot document, it will take a couple of weeks to do the edits and have the review committee do their thing. SC22 will issue the CD ballot after that. CD ballot takes 8 weeks. Your NB may have a different deadline. If you have any comments, please don't wait to submit NB comments. Let us know as soon as possible.

Evolution (Bastien/Dusíková/Keane)

Evolution met for all 5 days this week, seeing a total of 54 papers or Core issues. We discussed 13 papers for C++26 (including clarifications sent to us by CWG), and 37 various papers scheduled for C++29 and beyond.

Additionally EWG discussed 4 papers on the topic of Safety and Security. We decided to join P3100r2 into the UB whitepaper as proposed by 3656R1), which we also confirmed in a previous telecon poll that it is the correct direction.

EWG encouraged more work towards P3598R2's Profile Framework, as well as p3700R0, which is expected to result in a reference document for adding safety/security rules.

Library Evolution (Levi/Fracassi/Weis)

LEWG have met during the full week, and reviewed X papers. We've been working mostly on C++29 features, with few papers for improvements and fixes targeting C++26.

Main Topics Discussed

(already forwarded from LEWG, we discussed improvements or fixes for them)

- Reflection
- Relocatability
- SIMD

- Sender Receivers
- Concurrent Queues
- Ranges & Views

Fix papers forwarded (or sent back) to LWG/EWG for C++26

Reflections

- P3491R2: define_static_{string,object,array}
- P3096R11: Function Parameter Reflection in Reflection for C++26
- P3637R0: Inherit std::meta::exception from std::exception (applied on P3560)

Senders Receivers

- P3682R0: Remove std::execution::split
- P3481R3: std::execution::bulk() issues
- P3570R1: optional variants in sender/receiver
- P2079R8: Parallel Scheduler

SIMD

- P3691R1: Reconsider naming of the namespace for std::simd

Safety

- P3697R0: Minor additions to C++26 standard library hardening

Other Features & Fixes

- P3641R0: Rename std::observable to std::observable_checkpoint, and add a feature-test macro
- P3709R0: Reconsider parallel ranges::rotate_copy and ranges::reverse_copy
- P3223R1: Making std::istream::ignore less surprising
- P3748R0: Inspecting exception_ptr should be constexpr
- P1317R1: Remove return type deduction in std::apply

Papers Approved

The following papers were supported as fixes for C++26, but didn't make it to the draft during this meeting, therefore not included in the draft.

- P3690R0: Consistency fix: Make simd reductions SIMD-generic
- P1789R1: Library Support for Expansion Statements

Papers forwarded (or sent back) to LWG/EWG for C++29

Ranges & Views

- P3725R0: Filter View Extensions for Input Ranges, Rev 1
- P3705R0: A Sentinel for Null-Terminated Strings

Safety

- P0870R5: A proposal for a type trait to detect narrowing conversions

Other Features & Fixes

- P3091R3: Better lookups for `map` and `unordered_map`
- P3104R3: Bit permutations
- P3699R0: Rename `conqueue_errc`
- P3248R3: Require [u]intptr_t

Papers that got feedback and will be seen again by LEWG

- P3718R0: Fixing Customization of continues_on and schedule_from
- P3740R0: Last chance to fix std::nontype (potentially a fix for C++26)
- P3255R1: Expose whether atomic notifying operations are lock-free
- P3440R0: Add n_elements named constructor to std::simd
- P2929R0: simd_invoke
- P3445R0: Add utilities for easier type/bit casting in std::simd
- P2414R7: Pointer lifetime-end zap proposed solutions
- P3655R1: zstring_view (merged with P3710R0)
- P3669R1: Non-Blocking Support for `std::execution`
- P3526R0: Container truncation

Papers that did not get consensus or no actionable at this time

- P1144R13: std::is_trivially_relocatable (consensus against)
- P3631R0: Cleaning up the trivial relocation APIs in C++26 (consensus against, renaming for APIs encouraged)
- P3749R0: Concerns regarding std::zstring_view (feedback for P3655R1)
- P3566R1: You shall not pass `char*` - Safety concerns working with unbounded null-terminated strings (targeting profiles)
- P2902R2: constexpr 'Parallel' Algorithms (no consensus)
- P1028R6: SG14 status_code and standard error object (strong consensus - looking for champion!)
- P3711R1: Safer StringViewLike Functions for Replacing char* strings (no consensus)

Policies discussion

We did not discuss any policies during this meeting. We will resume discussion on policies in Kona, with the topics:

- Explicit Constructors
- (tentative) Unicode support
- (tentative) Error reporting, Preconditions and Hardened Preconditions support

Thank you to all our paper authors, attendees and reviewers, and looking forward to seeing you in Kona!

Core (Maurer/Merrill/Caves)

Thank you to Brian, Jan, and Joshua for taking the minutes.
We spent 2.5 days finishing up the main reflection paper. We succeeded. We also managed to get all the other reflection papers that were on our plate.
We did not get through all the other papers that were on our agenda. Apologies to those authors.
There are two things to point out :

- P2996 had a bug fix applied to it after the paper went up on the staging page, but before it was moved to the main straw poll page. If you have questions, please ask.
- There was a bug fix to the motions after the straw polls went up. We changed the category of CWG 1313 - it is not a DR because it applies to a feature that did not exist in C++23.

Any concerns about the change to the polls ?
None in the room.

CWG polls

1. Accept as Defect Reports and apply the proposed resolutions of all issues except issues 3013, 3014 and 3020 in P3752R0 (Core Language Working Group "ready" Issues for the June, 2025 meeting) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes

2. Apply the proposed resolutions of issues 3013, 3014 and 3020 in P3752R0 (Core Language Working Group "ready" Issues for the June, 2025 meeting) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes

3. Accept as a Defect Report and apply the changes in P3618R0 (Allow attaching main to the global module) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes

4. Apply the changes in P2996R13 (Reflection for C++26) to the C++ Working Paper and accept as Defect Reports core issues 2701 and 3026 resolved thereby.

No discussion.
Objection to unanimous consent.

In favour : 112 (78 in person + 34 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 10 (4 in person + 6 online)

Motion passes

Applause in the room.
Thank you to Dan Katz and everyone who was involved in reflection.

The first constexpr paper was a game changer. We will be looking back at this moment as a pivotal change in C++.

5. Apply the changes in P3394R4 (Annotations for Reflection) to the C++ Working Paper.

No discussion.
Objection to unanimous consent.

In favour : 118 (87 in person + 31 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 13 (5 in person + 8 online)

Motion passes

6. Apply the changes in P3293R3 (Splicing a base class subobject) to the C++ Working Paper.

No discussion.
Objection to unanimous consent.

In favour : 106 (77 in person + 29 online)

Opposed : 0 (0 in person + 0 online)
Abstain : 11 (4 in person + 7 online)

Motion passes

7. Apply the changes in P3491R3 (define_static_{string,object,array}) to the C++ Working Paper.

No discussion.
Objection to unanimous consent.

In favour : 106 (77 in person + 29 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 17 (8 in person + 9 online)

Motion passes

8. Apply the changes in P1306R5 (Expansion Statements) to the C++ Working Paper.

No discussion.
Objection to unanimous consent.

In favour : 112 (84 in person + 28 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 10 (0 in person + 10 online)

Motion passes

9. Apply the changes in P3096R12 (Function Parameter Reflection in Reflection for C++26) to the C++ Working Paper.

No discussion.
Objection to unanimous consent.

In favour : 108 (76 in person + 32 online)
Opposed : 3 (3 in person + 0 online)
Abstain : 14 (6 in person + 8 online)

Motion passes

10. Apply the changes in P3533R2 (constexpr virtual inheritance) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes

11. Apply the changes in P2843R3 (Preprocessing is never undefined) to the C++ Working Paper.

No discussion.
No objection to unanimous consent.
Motion passes

Library (Wakely/Garland/Kuhl)

LWG met all week in Sofia, spending our time trying to review a large number of proposals in our queue. As you can see from the number of polls, we were very busy and very productive. 35 library straw polls is not quite as many as the virtual plenary in 2022, but might be a record for a face-to-face meeting. Half a dozen of our polls are papers that had already been reviewed before this meeting, but the rest were reviewed here in Sofia.

We had several large proposals related to senders and simd, as well as numerous fixes and extensions to the standard library. Library were also involved in reviewing most of the reflection papers being moved by Core.

We only looked at one Library issue this meeting, which was just something we could fit into a few minutes before a break. We will be spending telecons looking at our issues list, and at the NB comments that will be made on the CD.

LWG polls

1. Apply the changes in P3742R0 (C++ Standard Library Issues to be moved in Sofia, Jun. 2025) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

2. Apply the changes in P2988R12 (std::optional<T&>) to the C++ working paper.

No discussion.
Objection to unanimous consent.

In favour : 94 (71 in person + 23 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 19 (10 in person + 9 online)

Motion passes

3. Apply the changes in P3348R4 (C++26 should refer to C23 not C17) to the C++ working paper.

No discussion.
No objection to unanimous consent.

Motion passes

4. Apply the changes in P3037R6 (constexpr std::shared_ptr and friends) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

5. Apply the changes in P3284R4 (write_env and unstoppable Sender Adaptors) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

6. Apply the changes in P3179R9 (Parallel Range Algorithms) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

7. Apply the changes in P3709R2 (Reconsider parallel ranges::rotate_copy and ranges::reverse_copy) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

8. Apply the changes in P3641R0 (Rename std::observable to std::observable_checkpoint, and add a feature-test macro) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

9. Apply the changes in P3044R2 (sub string_view from string) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

10. Apply the changes in P2876R3 (Proposal to extend std::simd with more constructors and accessors) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

11. Apply the changes in P3480R6 (std::simd is a range) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

12. Apply the changes in P2664R11 (Extend std::simd with permutation API) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

13. Apply the changes in P3691R1 (Reconsider naming of the namespace for std::simd) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

14. Apply the changes in P3383R3 (mdspan.at()) to the C++ working paper.

No discussion.
Objection to unanimous consent.

In favour : 86 (66 in person + 20 online)
Opposed : 1 (1 in person + 0 online)
Abstain : 24 (14 in person + 10 online)

Motion passes

15. Apply the changes in P2927R3 (Inspecting exception_ptr) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

16. Apply the changes in P3748R0 (Inspecting exception_ptr should be constexpr) to the C++ working paper.

Should this be reviewed by CWG ?
It was not seen in CWG.
This is adding constexpr to one function which was missing constexpr in the previous paper.
The implementation is in the paper.

No objection to unanimous consent.
Motion passes

17. Apply the changes in P2830R10 (Standardized Constexpr Type Ordering) to the C++ working paper.

No discussion.
No objection to unanimous consent.

Motion passes

18. Apply the changes in P3570R2 (optional variants in sender/receiver) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

19. Apply the changes in P3481R5 (std::execution::bulk() issues) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

20. Apply the changes in P3433R1 (Allocator Support for Operation States) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

21. Apply the changes in P3149R11 (async_scope - Creating scopes for non-sequential concurrency) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

22. Apply the changes in P3682R0 (Remove std::execution::split) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

23. Apply the changes in P2079R10 (Parallel scheduler) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

24. Apply the changes in P3557R3 (High-Quality Sender Diagnostics with Constexpr Exceptions) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

25. Apply the changes in P3560R2 (Error Handling in Reflection) to the C++ working paper.

Did CWG see this paper ?
CWG did not.
Do CWG think it needs to see this paper ?
CWG does not know what's in this paper.
This paper needs parts in the compiler in order for it to work.
There are no changes to the core wording clauses.
When we require compiler support, we have CWG review the change.
Are there any concerns from the compiler implementers ?
No concerns were raised in LEWG, it may make sense for CWG to see it.
We can not add features through NB comments, but we can remove them.

Objection to unanimous consent.

In favour : 85 (60 in person + 25 online)
Opposed : 8 (5 in person + 3 online)
Abstain : 24 (18 in person + 6 online)

Motion passes.

Please look at this as soon as possible.
We don't add features through NB comments. If you know of issues, file them as soon as possible, don't wait for NB issues.
We can remove things through NB comments.
We only do things that increase consensus.
Renaming is an issue level thing.

26. Apply the changes in P3503R3 (Make type-erased allocator use in promise and packaged_task consistent) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

27. Apply the changes in P3008R6 (Atomic floating-point min/max) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

28. Apply the changes in P3111R8 (Atomic Reduction Operations) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

29. Apply the changes in P3060R3 (Add std::views::indices(n)) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

30. Apply the changes in P2319R5 (Prevent path presentation problems) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

31. Apply the changes in P3223R2 (Making std::istream::ignore less surprising) to the C++ working paper.

No discussion.
Objection to unanimous consent.

In favour : 85 (65 in person + 20 online)
Opposed : 3 (2 in person + 1 online)
Abstain : 24 (14 in person + 10 online)

Motion passes.

32. Apply the changes in P2781R9 (std::constant_wrapper) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

33. Apply the changes in P3697R1 (Minor additions to C++26 standard library hardening) to the C++ working paper.

No discussion.
No objection to unanimous consent.
Motion passes

34. Apply the changes in P3552R3 (Add a Coroutine Task Type) to the C++ working paper.

This is the paper that was late.

Statement in the room : std:execution_task is the type that will be used the most in the senders receivers users. We need to get it right. I noticed some potential issues but assumed I can come up with improvements for C++29. I did not realise it was going to come into C++26 and have not been able to see it. I do not know if the issues are solvable. I'm worried about any unknown issues. This was extremely late in the cycle. The risk of this one is high.

Statement in the room : I'm not aware of design issues. I think the issues it has are solvable.
What is the consequence of this not going in at this meeting ?
It can't be in C++26.

Can we still make design changes to the paper ?
Any design changes should be fixes.

Do we have a process for taking features late ?
On p1000 there is a process to schedule exceptions. The votes that are mentioned have not been taken. We have taken plenary votes to say this is a feature we want to prioritise in the previous plenary. Direction group has also mentioned this as their priority

First poll was not taken, the discussion of taking a late paper wasn't had. There was a forwarding poll that was taken and it had weak consensus. It did not meet the bar for forwarding late paper. If noone objects, that's fine. However, we do have objections.

Without this, in C++26 we will have senders receivers, but no non-expert way of consuming them.

P1000 does not bind plenary sessions. We can still vote against this poll.
What is the best outcome for C++ at this point in time? Procedures exist to have a structured way of achieving that, but the outcome is what is best for c++

We spent a lot of time and energy to get where we are. If there are concerns, we can still address them.
We discussed a list of issues.

Does SG1 have any concerns about this paper ?
None in the room.

Do we have enough time and can we do appropriate fixes ? This is a pure library paper, but perhaps we need some core changes. Is this possible before C++26 ?
We don't add new features, we don't overhaul features. If it's a fix, that's fine.

We need a type for non experts, but a design may not be perfect. Can we put it in the experimental namespace ?
No, that's a design change.

Objection to unanimous consent.

In favour : 77 (58 in person + 19 online)
Opposed : 11 (6 in person + 5 online)
Abstain : 29 (19 in person + 10 online)

Motion passes.

35. Apply the changes in P1317R2 (Remove return type deduction in std::apply) to the C++ working paper.

No discussion.
Objection to unanimous consent.

In favour : 76 (64 in person + 12 online)
Opposed : 0 (0 in person + 0 online)

Abstain : 20 (9 in person + 11 online)

Motion passes.

WG21 Poll

1. Appoint an editing committee composed of Jonathan Caves, Daniel Kruegler, Nina Ranns, and Tim Song to approve the correctness of the C++ working paper as modified by the motions approved at this meeting, and to direct the Convener to transmit the approved updated working paper for CD ballot.

No discussion.
Objection to unanimous consent.

In favour : 103 (78 in person + 25 online)
Opposed : 0 (0 in person + 0 online)
Abstain : 5 (2 in person + 3 online)

Motion passes.

Direction Group (Wong)

Good morning, everyone. My name is Michael Wong. As the DG Chair, and on behalf of the Direction Group, I want to start by extending a heartfelt thank you to all of you. The sheer volume and quality of the work being done as we finalize C++26 is truly remarkable. We see the long hours being put in across the subgroups, and the progress you are making on major features is a testament to the dedication of this entire committee.

Welcoming New Voices

In Hagenburg, I announced the retirement of long-time contributor Howard Hinnant from C++. This week, it is with great pleasure that I announce the addition of two distinguished new members to the Direction Group. I'd like to ask them to stand up as I announce them.

- **Paul McKenney's** understanding of concurrency and the Linux kernel is without equal in WG21, and has been very much appreciated by SG1 since its earliest days. DG is grateful for him filling our blind spots in this area.
- **Jeff Garland** brings a wealth of library expertise not only because of his help running LWG, but also with his experience of running the Boost Foundation and the Beman Project. More generally, he has been a level-headed and sensible voice in the C++ community for a long time. Howard left some large shoes to fill, but it looks like they'll fit Jeff very well.

We also have our continuing members who I'd like to call on to stand up (if they are here):

- Roger Orr,

- Daveed Vandevoorde, and
- Bjarne Stroustrup.

They are standing up so you can see whom you should interact with if you have specific technical issues or even just to chat about anything. You can also contact us here or write papers targeted at us: direction@lists.isocpp.org.

The Road to C++26

C++26 is shaping up to be an exciting revision of the standard. However, this progress is not without intense debate and concerns about the growing complexity of the language. We need to do better to maintain a clear and coherent direction. We must address the risk of "diverse bloat and expert-only complexity".

All along, we, the DG, have been discussing technical outcomes of our ongoing processes, and we continue to explore ways to improve these. We'll be publishing a revision of P2000, our general direction paper, that includes our latest suggestions.

We noticed that for at least a handful of proposals, reaching consensus and maturing the proposal was either slow, or failed altogether. Usually, these proposals were ambitious, controversial, or both. Our goal is to encourage authors to build consensus by collaborating with individuals from different backgrounds and with different perspectives, rather than simply gathering allies who already agree. This is about strengthening proposals by incorporating diverse viewpoints early in the process.

Looking Forward: C++29 and Process Evolution

The DG is looking ahead to C++29. We intend to be more proactive in helping to guide competing proposals toward a unified solution. We recognize the immense effort that goes into every paper, and when strong, competing ideas emerge, we want to help the parties find common ground. As the group tasked with providing technical direction, we will facilitate collaboration to help the committee move forward with the best possible feature for C++.

Thank you again for your incredible work. Let's continue this great momentum and deliver an outstanding C++29.

8. Closing activities

8.1 Issues delayed until today

No discussion.

8.2 Mailings

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

2025-07-15: Post-Sofia
2025-10-06: Pre-Kona

LEWG will be taking a month of holiday

8.3 Plans for the future

No discussion.

8.4 Next and following meetings

Thank you to the sponsors of this meeting and the people who made it run smoothly.

- 2025-11-03/08: Kona, HI, USA (N4977)
- 2026-03-23 to 28: London, UK; Phil Nash
- 2026-06-22 to 27: Brno, Czechia; Mendel University in Brno
- 2026-11 southern Europe, possibly Mediterranean
- 2027 Bulgaria

9. Adjournment

Meeting adjourned at 11:20 AM UTC+2.

10. Attendance

Attendee	National Body
Adams, Michael	SCC
Adelstein Lelbach, Bryce	ANSI
Alday, Juan	ANSI
Arutyunyan, Ruslan	ANSI
Ažman, Gašper	BSI

Bagdonas, Ignas	BSI
Baker, Billy	ANSI
Baker, Lewis	ANSI
Balog, Pal	ANSI
Bauman, Jon	ANSI
Benetkiewicz, Paweł	PKN
Berge', Agustin	UNI
Berne, Joshua	ANSI
Beyls, Kristof	BSI
Bi, Brian	ANSI
Bindels, P.G.H.	NEN
Birbacher, Frank	ANSI
Boeckel, Ben	ANSI
Bonaventura, Xavier	DIN
Bott, Harold	ANSI
Brand, André	DIN
Brown, Walter E.	SII
Butler, Matthew	ANSI
Büttner, Sebastian	ANSI
ÇAĞRI, Murat Can	TSE
Cardoso de Souza Rodrigues, Guilherme	ASI
Cassagnes, Aurelien	ANSI
Caves, Jonathan	ANSI
Chen, Jolly	SNV

Childers, Wyatt	ANSI
Corden, Richard	ANSI
Craig, Philip	BSI
Cranmer, Joshua	ANSI
D'Angelo, Giuseppe	ANSI
Davidson, Guy	BSI
de Wever, Mark	ANSI
Delfino, Gianluca	UNI
Dimov, Petar	BDS
Dos Reis, Gabriel	AFNOR
Douglas, Niall	NSAI
Doumler, Timur	ANSI
Downey, Steve	ANSI
Dusikova, Hana	UNMZ
Engert, Daniela	DIN
Estell, Khalil	ANSI
Fertig, Andreas	DIN
Floyd, Paul	ANSI
Foco, Marco	UNI
Fracassi, Fabio	DIN
Ganetsky, Braden	SCC
García Sánchez, José Daniel	UNE
Garland, Jeff	ANSI
Genovese, Walter	ANSI

Gill, Mungo	NSAI
Giroux, Olivier	ANSI
Goodspeed, Nathaniel	ANSI
Gruber, Bernhard	ANSI
Gustafsson, Bengt	SIS
Hagins, Jody	ANSI
Halpern, Pablo	ANSI
Hauswedell, Hannes	IST
Hava, Michael Florian	ASI
Herring, Davis	ANSI
Hoemmen, Mark	ANSI
Honermann, Tom	ANSI
Hunt, Oliver	ANSI
HUSSONG, Charles	JISC
Jabot, Corentin	AFNOR
Josuttis, Nicolai	DIN
Kamiński, Tomasz	PKN
Karampour, Maryam	SCC
Katz, Dan	ANSI
Kawulak, Robert	PKN
Keane, Erich	ANSI
Keir, Paul	BSI
Keremidchiev, Vassil	BDS
Khyzha, Artem	BSI

Koeppe, Thomas	ANSI
Komitov, Ivan	BDS
Kostur, Andre	SCC
Kosunen, Elias	SFS
Kozicki, Bronek	ANSI
Kozicki, Bronek	BSI
Kretz, Matthias	DIN
Krzemienski, Andrzej	PKN
Kuhl, Dietmar	ANSI
Kulczycki, Peter	ASI
Lach, Adam	SNV
Lakos, John	ANSI
Larson, Brad	ANSI
Lauko, Henrich	UNMZ
Laverdière-Papineau, Marc-André	ANSI
Lebrun-Grandie, Damien	ANSI
Levi, Inbal	ANSI
Levi, Inbal	SII
Levine, Michael	ANSI
Li, Yihe	ANSI
Liao, Tian	SAC
Liber, Nevin	ANSI
Lippincott, Lisa	ANSI
López Gómez, Javier	UNE

Machutova, Jana	UNMZ
Majumder, Abhilash	ANSI
Maness, Wesley	ANSI
Maurer, Jens	ANSI
McKenney, Paul	ANSI
McMonagle, John	BSI
Meerwald, Christof	ASI
Mejstrik, Thomas	ASI
Meredith, Alisdair	ANSI
Michael, Maged	ANSI
Miller, Cody	ANSI
Morales, Nicolas	ANSI
Müller, Jonathan	DIN
Myers, Nathan	ANSI
Naumann, Axel	SNV
Neațu, Darius	ASRO
Niebler, Eric	ANSI
Nishanov, Gor	ANSI
Nishanova, Anzhela	ANSI
Nolan, Edward	ANSI
Olsen, David	ANSI
Orr, Roger	BSI
Owen, Nathan	ANSI
Pashov, Vasil	BDS

Peacock, Antony	BSI
Pelliccioni, Fernando	ANSI
Persson, Jonas	SIS
Petersen, Ian	ANSI
Polukhin, Anton	GOST R
Preney, Paul	SCC
Pusz, Mateusz	ANSI
Pusz, Mateusz	PKN
Ranns, Nina Dinka	BSI
Revzin, Barry	ANSI
Rigault, Jean-Paul	AFNOR
Rivera Morell, René Ferdinand	ANSI
Ronkainen, Jari	SFS
Rosten, Oliver	BSI
Roy, Patrice	SCC
Ryan, Christopher	ANSI
Sandoe, Iain	BSI
Sankel, David	ANSI
Satle, Ankur	BIS
Sattler, Florian	ANSI
Schultke, Jan	ANSI
Scogland, Thomas	ANSI
Serebrennikov, Vladislav	ANSI
Snyder, Jeff	BSI

Song, Tim	ANSI
Spencer, Michael	ANSI
Spicer, John	ANSI
St. Amour, Bryan	SCC
Starosz, Sebastian	PKN
Stroustrup, Bjarne	ANSI
Sutter, Herb	ISO/IEC JTC 1/SC 22
Taylor, Matthew	BSI
Tenty, David	ANSI
Tenty, David	SCC
TEODORESCU, Lucian Radu	ASRO
Teoh, Joon Nam	ANSI
Terashima, Yui	JISC
Tong, Hubert	SCC
Towner, Daniel	ANSI
Trott, Christian	ANSI
Tsaousis-Seiras, Isidoros	ANSI
van Winkel, J.C.	NEN
Vandevoorde, Daveed	ANSI
Varlamov, Konstantin	ANSI
Vasama, Lauri	SFS
Vasilev, Vasil	BDS
Vollmann, Detlef	SNV
Voutilainen, Ville	SFS

Wakely, Jonathan	ANSI
Walker, Kelly	ANSI
Waterloo, Jarrad	ANSI
Weis, Andreas	ANSI
Williams, Anthony	BSI
Williamson, Gerald	ANSI
Wong, Jessica	ANSI
Wong, Michael	SCC
Xie, Hui	BSI
Yuan, Zhihao	ANSI
Zimmermann, Philipp	ANSI
Zissu, Andrei	ANSI
Birlica, Iulian	ASRO
Ceyhun Erturk	TSE
Felix Hellmann	DIN
Gray, Alan	
Harry Ng	ITCHKSAR
Ionuț Nicula	ASRO
Ivanov, Ivaylo	NEN
Kale, Vivek	ANSI
Matthias Wippich	DIN
Mohamed Ayoub AKKAOUI	UNMZ
Panagiotis Syskakis	ANSI
Peter Hrenka	DIN

Pulkkinen, Esa	
Vlad Negoita	ASRO