WG21 2018-11 San Diego
Record of Discussion

ISO/IEC JTC1 SC22 WG21 P1338R1 — 2018-11-26

Nina Dinka Ranns, nina@edg.com

November 5 - November 10, 2018 - San Diego, CA, USA

Chair John Spicer

Changes from previous revision:

- Fixing various typos
- Amending time zone details

1. Opening activities

John Spicer opened the meeting at 9:05 PST.

1.1 Opening comments, welcome from host

Marshall Clow welcomes the group and presents the information about the location of the meeting. Lunch is provided at the hotel by Qualcomm
Thank you to Qualcomm.

John Spicer presents the schedule for the meeting. Working groups will meet until Friday. Plenary is scheduled for Saturday at 8:30 am PST. There are several evening sessions scheduled. Some subgroups may have a shorter lunch period, please check with your group chair.

C++ foundation meeting will happen this evening in the plenary room. The attendance is optional.

If you have any questions, the schedule can be found on the wiki

Marshall: If you are new, please ask the person next to you for information on how to access the meeting wiki.

1.2 Meeting Guidelines

John Spicer presents.
Every participant is responsible for understanding and abiding by the following:

- The INCITS Antitrust Guidelines (PL22.16)
- The INCITS Patent Policy (PL22.16)
- The ISO Code of Conduct
- The WG21 Practices and Procedures, and Code of Conduct

Please review the guidelines. WG21 procedures and Code of Conduct documents are new and very important. We want to make sure that everybody interacts in a professional and respectful way.

If you have any questions talk to one of the officials.

1.3 Membership, voting rights, and procedures for the meeting

John Spicer presents. Meetings are not public, but we allow visitors. Speak freely, don't do live tweeting, don't quote people without their permissions. Photography is fine, please post after the meeting.

For the US organization 1 vote per company, for other countries, one vote per national expert.
In the WG and SG everyone can take part in straw polls
For more information about membership talk to Hal Finkel.

Barry Hedquist presents. PL22.16 has the ballot process on Saturday. The Saturday vote is approval of Systematic Review for ISO/IEC TS 19217:2015 -- C++ Extensions for Concepts. If you are not here on Saturday, and you are a voting member, please send an email to John, Hal or Barry.

1.4 Introductions

Officers and WG chairs introduce themselves
First time attendees introduce themselves.
John Spicer welcomes first time attendees.

1.5 Agenda review and approval

John Spicer presents the agenda for the meeting. The meeting will finish no later than 2pm on Saturday, but WG may continue working.

Barry Hedquist moves. Adam Martin seconds.
The motion is unanimously approved by PL22/16.
The motion is unanimously approved by WG21.

1.6 Editor's reports, approval of working drafts

Document	Editor's report	Prospective WD
C++20 Standard	N4779	N4778
Coroutines TS	N4776	N4775
Networking TS	N4772	N4771
Parallelism v2 TS	DONE	DONE
Reflection TS	N4767	N4766
Library Fundamentals TS	N4759	N4758

Motion to approve editor's reports and working drafts.
The motion is unanimously approved by WG21.

1.7 Approval of the minutes of the previous meetings

Meeting	Minutes
WG21 Rapperswil	N4781
PL22.16 Rapperswil	pl22.16-2018-00010
WG21 pre-San Diego administrative telecon	N4784

Motion to approve the minutes of the previous meeting.

The motion is unanimously approved by WG21.

PL22/16 motion to approve the minutes of the previous meeting.
Marshall Clow moves, Barry Hedquist seconds.
The motion is unanimously approved by PL22/16.
The motion is unanimously approved by WG21.

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

No discussion.

3. WG progress reports and work plans for the week (Core, Evolution, Library, Library Evolution; see pre-meeting WG21 telecon minutes)

No discussion.

4. New business requiring action by the committee

Herb Sutter presents : There are 181 people at this meeting. We have grown a lot and the number of papers has grown a lot too. We used to have less papers than people, but that is no longer the case. We still want to look at all of the papers
We have 12 NB, next year we may have 14 NB
To be able to see all the papers, we have scaled the groups and added incubation groups at the meeting. We are still bottlenecked at LWG and CWG which need to do the final review of the papers.
We have a lot of customers and they have been struggling to adopt everything we have added to the language. We want to make sure that everything we add counts.
We should try to avoid single purpose solutions
When we look at the new proposal, and especially at the language side, they fall into 4 categories.
New features where we asked for papers. These are likely to succeed.
Improvements to existing features which help to generalise a feature. These are likely to succeed
Narrow scope features. We would like to ask incubators to take a look at those to see if there are any features that relate to each other.
Solo features which may be a good idea. They are unlikely to succeed unless they can be generalised.

Herb presents the WG21 groups.
Herb presents what days which group meets.
Herb presents the meeting schedule.
P1000 is the paper that will contain the progress of the standardisation.
This is the last meeting for new C++20 features, and features that need LWG changes
Anything in the incubators is not C++20 intended.

Mike Miller : I see Prague on the schedule of future meetings.
Herb : yes, the first meeting 2020 will be in Prague, we don't have any details yet. When there are details, they will be posted on the isocpp page.

Herb continues.
You don't have to vote in the subgroups. If you haven't kept up with the discussion, it isn't helpful

Titus Winters : what about voting neutral ?
Herb : if you vote neutral , it means you know what is going on, but you have no opinion.
In subgroup anyone can vote, but please only vote if you have an informed opinion.
We will try to explicitly include the shipping vehicle in the subgroup polls.
Tom Honermann: how do we interpret consensus in polling ?
Herb : look at SD4 on standardisation page. Chair can decide what is consensus within their subgroup.
John : the link to SD4 is in the agenda. See “The WG21 Practices and Procedures, and Code of Conduct”.

Herb continues : we will try to improve the queue of comments, to avoid out of order insertion and tangents.
Stuart Davis Herring : scribes, please write down to who something was a re-sponse.
Bryce Adelstein Lelbach : it is important to capture the name . If you are taking the minutes, please ask for the name.
Herb : we always take notes. The scribe always gets to interrupt and make sure the name is correct.
Nathan Meyers : in WG sometimes people ask a question. Can they still ask them ?
Herb : what I said is a general rule, but chairs should take each case separately.

5. Plans for the future (special Monday session to discuss Cologne)

Plans for the future (PL22.16)

1. Next and following meetings
   - 2019-02-18/23: Kona, HI, US (N4765)
   - 2019-07-15/20: Cologne, Germany (N4780)

- 209-11-4/09 Belfast , Northern Ireland

How many people plan not to attend Kona : 15
How many people plan not to attend Cologne 17
How many people plan not to attend Belfast : 15

Nico Josuttis presents : Cologne meeting is not sponsored by a single company. Michael Spertus contributed too. If you plan to do the same, please contact me. There is an issue with the invitation, there will be an updated paper in the post San Diego mailing with a working link and other fixes. Please register before 10th of May. Any questions, please send me an email.

Jens Maurer presents : 14th of December is the registration cut off for Kona. Please book your hotel soon.

6. WG and SG sessions

John Spicer presents : during the week, CWG and LWG will be putting out proposal that will be voted on Saturday. Those need to be finalised and made available by 8pm PST Friday.
Only new papers go on the straw poll page, the older papers can be found in the mailing. If you have concerns, please make yourself known to chairs or on the mailings.

Jens Maurer presents meeting room schedule. The schedule can be found on the wiki.
Show of hands for each group to see if we need to amend the assigned rooms.
Jens gives directions for each room.

SG5 will meet after the plenary meeting.
Lawrence Crowl : we will do issue triage today.

SG10 will meet during the week. See John Spicer.

Jens presents evening sessions.

Monday: Direction of metaprogramming(Ville Voutilainen).
Ville : This is an information-gathering session, for the chairs of EWG and LEWG to find out what the concerns with SG7's directional guidance are. It is not a debate. It is not a session that will take a poll. EWG and LEWG regulars are invited, as is Direction; the rest of the committee is uninvited. If you have concerns you wish to air, try to find an EWG or LEWG regular to air those concerns.
The duration of this session will be 60 minutes, after which further concerns will need to be conveyed offline.

Monday: Editorial session (Richard Smith) in room "Brittany". Schedule includes [structure.specifications] guidance.

Tuesday: Signedness and span (Bryce Adelstein Lelbach)
Bryce : this is only an informative session, we will not be making any decisions.
Titus Winters: LEWG will have to schedule the actual vote for span's signedness, I haven't done that yet. I also encourage people to attend the evening session because the arguments have evolved a bit.

Wednesday: Education of C++ (Jan Christiaan van Winkel)
Thursday: SG15 Tooling (Titus Winters)
Thursday: LWG issues processing in room "Fairfield" (Marshall Clow)

Jens presents meeting times and the break times.
Breakfast should be available earlier than the meeting starts.

7. Review of the meeting (Saturday 8:30 AM)

John Spicer opened the meeting at 8:34am PST.

Herb Sutter presents.
Herb Sutter goes over the voting procedure.
Thank you to Michael Wong for chairing SG5. Michael will be stepping down and Hans Boehm has volunteered to take over as chair of SG5.

There are two new study groups :
SG19: Machine Learning (Michael Wong)
SG20: Education (JC van Winkel).

WG and SG status and progress reports.

- **SG5: Transactional memory (Wong)**

Michael Wong presents. There is a continuous discussion on reduced transactional memory in conjunction with the industry . Thank you to Hans for picking this up

- **SG6: Numerics (Crowl)**

Herb presents. SG6 met twice this week and proposed several papers.

- **SG7: Compile-time programming (Carruth)**

Chandler Carruth presents. Sg7 met for an afternoon and talked about 3 things.

Formal guidelines for what kinds of papers need review by sg7 before going to other groups, preview of what reflection facilities might look like if we build them on top of the new constexpr facilities, and new perspective on reflection on top of constexpr facilities, but from a different group. We will continue with both approaches so we can make a better decision on the direction we should take.
There was also an informational session in the evening to help spread information on compile time programming facilities added to constexpr.

- ## SG10: Feature test (Nelson)

John Spicer presents. SG10 is encouraging library and core group to check if a proposal should have a feature test macro. SG10 also went through some papers to see if they need FT macros. All decisions should be made through CWG an EWG as a part of the normal proposal handling.

- ## SG12: Undefined and unspecified behavior (Dos Reis)

Herb reads out report by Gabriel Dos Reis. SG12 had a joint session with EWG on Wednesday morning regarding UB and contracts. A partial resolution was sent to Core.
Thursday and Friday was a joint session with WG23. The group primarily recommended mitigation measures for vulnerabilities related to control structures. It is remarkable that those measures were already covered by the C++ Core Guidelines. Stephen Michell will make the new document available on WG23 website.

- ## SG13: HMI & I/O (Human/Machine Interface) (Roger Orr)

Roger Orr presents. The group is re-activating because of the P0267, the 2D graphics paper. SG13 voted to encourage the paper author to work with the contra papers (P1225R0 and P1062R0). The group also looked at web view paper (P1108R1) and encouraged further work. SG13 would like input from graphics experts but there is a problem of arranging a time to meet them.

- ## SG14: Games & low latency (Wong)

Michael Wong presents. SG14 had an impromptu meeting on linear algebra. There was a high level discussion of scope. SG15 also discussed frequency of meetings. Please contact Michael to ask about the meetings. Michael will assign chairs for each meeting.
SG14 plans to start having more face to face meetings. Guy Davidson, Ben Craig, and Ben Saks might take the lead on that.
SG14 will continue with two telecoms every month and triaging regular SG14 matters. Linear Algebra will be breaking up as a sub group/

Walter Brown : is numerics SG involved in linear algebra work ?
Micheal Wong : yes, and a few other SG have interest in linear algebra.

- ## SG15: Tooling (Winters)

Bryce Adelstein Lelbach presents. SG15 had an evening session to discuss seven papers. There was a lot of good discussion. The direction of the proposals is not clear yet.

- ## SG16: Unicode (Honermann)

Tom Honermann presents. SG16 had first face to face meeting, it was well attended. SG16 talked about the direction and strategy, and identified a couple of priorities that the group will focus on. SG16 wants to improve support for transcoding SG16 looked at P1275 and worked on advancing on other papers, including some unicode features, mandating the encoding that's used for 16 and 32 integers, and normalisation.
SG16 looked at papers on efficient string building. The group also provided feedback to other SG on questions they had.

- ## SG1: Concurrency (Giroux)

Olivier Giroux presents. SG1 processed 34 paper, and talked about dismantling volatile.

Giroux presents SG1 delivery pipeline to C++20

jthread + interrupt_token that supports callbacks, moved SG1 → LEWG.
Forward executors with one-way/one-way-bulk only, moved SG1 → LEWG.
The “C++20 Synchronization Library” omnibus paper moved LEWG → LWG.
Memory model repairs/improvements moving CWG → IS20.

No new/novel c++20 papers from now on. There is room for integration with existing features and fixes, any new work should be aimed at the next shipping vehicle.

Please don’t rename concurrency term without consultation with SG1.

Co-routines: no SG1 issues; no library additions. Forward defects 31 & 32 to CWG.
Additional library support for “awaitable” types needs more bake time in SG1.

Freestanding C++ direction needs clarifying. The interest among SG1 members is pretty substantial. It would be good if we could avoid disjointed polls.

Evening session summary in post mailing.

- ## SG17: EWG Incubator (JF Bastien)

Erich Keane presents
EWGI met all-day Monday / Tuesday / Wednesday.
These are the papers that EWG likely won't get to see in San Diego. EWGI will see the ones it can. We've prioritized papers that could plausibly make it to C++20 (either because they target a C++20 feature, or they're small). EWGI goal was first to disambiguate the shipping vehicle, and either forward to EWG for consideration this week (unlikely), or provide paper feedback.
EWGI saw 30 papers:

- 5 papers sent as-is to EWG
- 2 papers sent to EWG asking for feedback (will be seen again by EWGI)
- 6 papers had no consensus to move forward (we recorded objections, authors know what would get people to change their mind)
- Other papers got feedback, some need input from other groups, they'll be seen again by EWGI
- EWGI have a backlog of 27 papers
- EWGI spent most of Wednesday afternoon looking at the merged pattern matching proposal, and gave plenty of feedback to the authors

Papers we sent to EWG:

- P1009R0 Array size deduction in new-expressions
- P1041R1 Make char16_t/char32_t string literals be UTF-16/32
- P1097R1 Named character escapes
- P1099R2 Using Enum
- P1301R0 nodiscard should have a reason: request that LEWG see this as well: maybe they want to add reasons to current nodiscard in the STL. Maybe EWG should see this after LEWG decides whether / when they want to use it.
- P1203R0 Modular main(). Want EWG feedback, choose:
    1. Take the suggested direction
    2. Take the suggested direction and modify the signature of main (return type, return omission, argc/argv or other parameters, Unicode, environment, etc)
    3. Don't do modular main, instead add an entrypoint attribute
- P1040R2 std::embed is forwarded to EWG not because it's ready, but because EWGI felt that EWG feedback was required on the approach:
    1. As proposed, allow a constexpr std::embed which takes a constexpr string (which can be created at compile-time!). There's worry that doing this breaks module's dependency scanning because we need to do full constexpr evaluation before discovering that a module depends on a file being embedded.
    2. Somehow use module syntax to import the file to embed, creating a constexpr datastructure. There's worry that this forbids generating the file name at compile-time.

Daveed Vandevoorde : you mentioned p1009r0 is seen by core ?
Timur Doumler : no, it hasn't been seen yet.

- ## Evolution (Voutilainen)

Ville Voutilainen presents :
Concepts:
We reviewed P1141R1, Yet another approach for constrained declarations, and approved it. Core moved it in San Diego. In addition, we approved P0848R0, Conditionally Trivial Special Member Functions.
We did not look at P0782R2, Constraining Concepts Overload Sets, despite it being on the agenda. This proposal changes dependent calls in constrained templates to consider only the overloads that were used to satisfy a constraint as viable candidates. We will possibly revisit the proposal in Kona, as we need to decide the fate of this change before C++20 ships.

Modules:
The merged Modules proposal, P1103R1 was approved, with a bunch of fixes, including

- Making the module and import keywords context-sensitive
- Support for inline module partitions
- Change to when a module preamble is considered to end
- Fixes to redefinitions in legacy imports

The global module fragment was not removed, and the paper on retiring pernicious language constructs in module contexts was rejected.

The plan has been for quite some time to wait until Kona before moving Modules, so as to give implementers more time to gain implementation experience.

Coroutines:
We discussed P1063R1, Core Coroutines. Evolution, again, still, has consensus (but far from unanimity) to move the merge of the Coroutines TS forward.

Aggregate initialization:
The proposal to allow paren-initializing aggregates, P0960, was approved.

Contracts:
The change to allow preconditions and postconditions to use private mambers, P1289R0, was approved.
We discussed UB and optimizations at quite some length; the work on that problem is ongoing, and we'll see more in Kona.
Some constexpr extensions

- P1327, allowing dynamic_cast in constant expression was approved. It's also possible to use typeid and its comparison in constant expressions.
- We approved being able to change the active member of a union in constant expressions.
- We approved P1073, "consteval".

- We discussed "for..." and "for constexpr". Further work is expected with encouragement that we want facilities like that.

Template argument deduction:
We approved fixes that allow CTAD on aggregates, alias templates, and using inherited constructors.

All Your Spaceships Are Belong to Us:
The papers P1185 and P1186 were approved. We expect further discussion on spaceship in Kona.
There was no consensus on making comparison operators chain, nor was there encouragement for further work in that area.

Structured bindings:
We discussed removing the need for tuple_element and allowing explicit typing of bindings. There was no consensus for either; Pattern Matching can solve the latter problem.

Defaulted special member functions and noexcept:
We approved P1286, which changes noexcept on defaulted special member functions to no longer require the noexcept-spec to match what would've been implicitly generated.

Guaranteed constant initialization for non-constexpr variables:
We discussed P1143, Adding the [[constinit]] attribute, and we expect to see a revision in Kona. The current guidance is for a keyword instead of an attribute.

Deprecations:
Deprecating volatile was discussed; a revised proposal is expected for Kona.
P1161, Deprecate uses of the comma operator in subscripting expressions, was approved. This paves the way to multi-argument subscripting. We didn't approve making that possible yet, but the deprecation is approved.

◦ SG18: LEWG Incubator (Bryce Adelstein Lelbach)

Bryce Adelstein Lelbach presents: LEWGI met for 3.5 days, but we had a room for an extra morning so we processed some additional work. Our approach was :
direction : do we want to solve this problem
design : how should we solve this problem
describe: how do we specify the solution

The process was :
- overview of the proposal,
- directional discussion : do we want to solve this problem ?
- take a poll : do we want to spend more time on this proposal ?
If yes, then do some design discussion and design improvement until it is ready to go to the library group.

No consensus to spend more time: 10
Sent to LEWG for C++20: 3
Sent to LEWG for input: 2
Sent to SG1 for Conc v2 TS: 1
Ready for LEWG(I) design review for C++23: 10
Further refinement in LEWGI needed: 6
Total LEWGI papers seen: 32

C++20 papers (sent to LEWG)
P1207R0 Moveability of Single-pass Iterator
P1310R0 Unifying the Many Ways to Compare
P0288R2 std::unique_function

Concurrency v2 TS papers
P0876R3 std::fiber_context
P0260R2 Concurrent Queues

LEWG Input Needed
P1072R1 Optimized Initialization for std::string
P1010R1 Container Support for Implicit Lifetime Types

Refine in LEWGI
P0959R1 std::uuid
P1275R0 std::environment
P0447R4 std::colony
P1276R0 Type Manipulation Facilities
P1069R0 Refining Library CTAD Support
P0638R0 Crochemore-Perrin std::search

C++23 (design review in LEWG(I))
C++23 Smart Pointers:
P0468R1 Intrusive Smart Pointers
P1132R1 std::out_ptr
P0211R0 std::allocate_unique
P1066R1 std::exception_ptr inspection
P0993R0
P1066R1 std::find_backwards
P1317R0 std::apply enhancements
P1318R0
P0205R0 Seeding with std::random_device

No Consensus to Spend More Time
P0275R4 Dynamic Loading Library
P1159R0 std::any_iterator
P1201R0 std::variant Direct Comparisons
P1196R0 Value-based std::error_category Comparison ABI break
P1197R0 Non-allocating std::error_category::message ABI break
P1198R0 std::error_category::failed ABI break

P0343R0 High-order Metaprogramming Functions
P1081R0 Empty Struct Types
P0319R0 Emplace for std::promise/std::future

Ville : did you discus ranges based solution for find_backwards algorithm ?
Bryce : no. We originally did, but the feeling in the room was that it wasn't important enough to prioritise for c++20

◦ Library Evolution (Winters)

Titus Winters presents. 64 papers reviewed (44 last time), 14 papers left unreviewed (~28 last time) of which ~5 that could potentially target C++20

Pablo Halpern : can you attribute the increase in productivity to anything ?
Titus: it is helpful to have the incubator group, we didn't look at things that weren't plausibly baked. It's also where we are in the cycle, a lot of things are just about done. The fact we panicked a little bit and may have pushed the schedule and discussion accordingly

Titus continues. Policy Changes discussed :
● Do we provide extensions by inspecting user-provided types?
● When do we mark constructors explicit?

Forwarded to LWG for IS
● P1085 Should Span Be Regular
● P1024 Usability Enhancements for std::span
● P0784 Standard containers and constexpr
● P0533 constexpr for <cmath> and <cstdlib>
● P1251 A more constexpr bitset
● P0595 std::is_constant_evaluated()
● P0340 Making std::underlying_type SFINAE-friendly
● P0920 Precalculated hash values in lookup
● P0627 Function to mark unreachable code
● P1165 Fixing allocator usage for operator+(basic_string)
● P1210 Completing the Rebase of Library Fundamentals, Version 3, Working Draft
● P0325 to_array from LFTS with updates
● P1083 Move resource_adaptor from Library TS to the C++ WP
● P1209 Adopt Consistent Container Erasure from Library Fundamentals 2 for C++20
● P1187 A type trait for std::compare_3way()'s type
● P1191 Adding operator<=> to types that are not currently comparable
● P1248 Fixing Relations
● P1295 Spaceship library update
● P1252 Ranges Design Cleanup
● P1243 Rangify New Algorithms

- P0645 Text Formatting
- P0881 A Proposal to add stack trace library
- P0883 Fixing Atomic Initialization
- P1233 Shift-by-negative in shift_left and shift_right
- P0549 Adjuncts to std::hash
- P1272 Byteswapping for fun&&nuf
- P0631 Math Constants
- P1280 Integer Width Literals
- P1186 When do you actually use <=>?
- P1135 The C++20 Synchronization Library
- P0201 polymorphic_value
- P1227 Signed size() functions
  - span::size() is unsigned, adds std::ssize()
  - Compatible with P1089

Discussed, Will Prioritize
- P1208 Adopt source_location from Library Fundamentals V3 for C++20
- P1293 ostream_joiner
- P1035 Input range adaptors
- P1206 Range constructors for standard containers and views
- P0798 Monadic operations for std::optional
- P0443 Unified Executors
- P0437 Numeric Traits for the Next Standard Library
- P0556 (revisited) Change ceil2 input requirements
- P0586 Safe integral comparisons
- P1222 A Standard flat_set
- P0660 A Cooperatively Interruptible Joining Thread
- P1072 Optimized Initialization for basic_string and vector

Discussed but not forwarded
- P1259 Merge most of Networking TS into C++ Working Draft
- P0863 Fixing the partial_order comparison algorithm
- P1312 Comparison Concepts
- P1291 std::ranges::less<> Should Be More!
- P1255 A view of 0 or 1 elements: view::maybe
- P0943 Support C atomics in C++
- P1175 a simple and practical optional reference for C++

Discussed and rejected
- P0657 - Deprecate certain declarations in the global namespace
  - We are not aiming for user code to reclaim the global namespace
- P1182 - New names for the power-of-2 templates
- P0310 - Splitting node and array allocation in allocators
- P1163 Explicitly Implicifying explicit Constructors
- P0262 A Class for Status and Optional Value
- P0891 Let strong_order Truly Be a Customization Point!
  - LEWG would rather make types hashable than comparable

- ## Core (Mike Miller)

Mike Miller presents. CWG is preparing for c++20 and has looked at various papers that were forwarded from EWG. Total of 26 papers, 16 will be moved today including co routines. CWG looked at, but did not move today, modules paper. CWG need to refine that and do a review for Kona.
There are 4 issue resolutions that will be moved today. CWG reviewed additional number of issue resolutions at this meeting and 4 editorial issues that needed cwg input. Of the issues that were not approved, some were forwarded to EWG for input.
There will be a session after this meeting to process some remaining issues resolutions and prioritisation, and 3 telephone conferences which will take place first monday every month before the Kona meeting. CWG will be reviewing modules or doing issues processing at these telephone conferences.
There is a large backlog of issues resolutions where CWG determined a direction, but the drafting has not yet materialised. We would like to clear up that backlog.

CWG Motions

Motion 1
Move to accept as Defect Reports all issues in P1350R0 (Core Language Working Group "tentatively ready" Issues for the November, 2018 (San Diego) meeting) and apply the proposed resolutions to the C++ working paper.

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

Motion 2
Move to apply the changes in the "Consolidated wording" section of P0668R5 (Revising the C++ memory model) to the C++ working paper.

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

Motion 3
Move to apply the changes in P0982R1 (Weaken Release Sequences) to the C++ working paper.

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

Motion 4
Move to apply the changes in P1084R2 (Today's *return-type-requirement*s Are Insufficient) to the C++ working paper.

Mike Miller : there was an email just this morning to CWG that there is a conflict with another paper later on. We are aware of the conflict and the authors are in contact, so this shouldn't be a problem.

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

Motion 5
Move to apply the changes in P1131R2 (Core Issue 2292: *simple-template-id* is ambiguous between *class-name* and *type-name*) to the C++ working paper.

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

Motion 6
Move to apply the changes in P1289R1 (Access control in contract conditions) to the C++ working paper.

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

Motion 7
Move to apply the changes in N4775 (Working Draft, C++ Extensions for Coroutines) as modified by the changes in P1356R0 (Coroutine TS ready issues (25 and 27)) to the C++ working paper.

No discussion.
There are objections in the room.

Herb Sutter: we had this vote in Rapperswill. This was proposed for c++17, then we decided to do a TS for alternative proposal to be brought forward. There weren't any then, but there are now. Here we get to say yes or no, but we don't decide on the design.
We get new Coroutines papers every meeting, so that's why they are put to vote again.
If this fails, we will probably see this again in Kona.
This is not just a repeat of rapperswill, there are also features from core coroutines proposal added to the current coroutines proposal.

Chandler : in Rapperswill there was an early presentation of core coroutines. It failed plenary. We decided to have it come back and get a better understanding. We had that. There was a much better presentation at this meeting. I don't think

there will be substantial increase of information before Kona. The big chunk of work is not one meeting away. I don't think there will be big change in current TS between now and Kona. We need to decide if we want co routines in C++20.

Fabio Fracassi : Is the merge supposed to be finished by c++20 ?
Chandler : the merge has not been discussed in Evolution.
Herb : we are talking two merges. The issues were merged, but the whole proposals were not merged.

Eric Niebler : noone is suggesting that a merge of the two proposals will be available in Kona.
Roger Orr : Bulgarian paper was a good response to current TS, and they have some serious concerns. We would like to look at those concerns and address them
Bjarne : I think we waited long enough, If we don't go the current way, we are looking at C++23 or even C++26. We have been waiting far too long

Herb : NBs are saying they want co routines in C++20. They also want to explore the other directions. We can't have both.

Herb polls the room
In favour : 34
Opposed : 17
Abstain : 10

Herb polls national bodies by roll call:

Bulgaria Opposed
Canada Opposed
Finland In favour
France In favour
Germany Opposed
Netherlands In favour
Poland Opposed
Russia In favour
Spain Abstain
Switzerland Opposed
UK Opposed
US In favour

Motion fails

Herb : NB approval has decreased compared to the similar straw poll during the previous meeting.

Motion 8
Move to apply the changes in P1002R1 (Try-catch blocks in constexpr functions) to the C++ working paper.

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

Motion 9
Move to apply the changes in P1327R1 (Allowing `dynamic_cast`, polymorphic `typeid` in Constant Expressions) to the C++ working paper.

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

Motion 10
Move to apply the changes in P1236R1 (Alternative Wording for P0907R4 Signed Integers are Two's Complement) to the C++ working paper.

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

Motion 11
Move to apply the changes in P0482R6 (char8_t: A type for UTF-8 characters and strings (Revision 6)) to the C++ working paper.

No discussion.
There are objections in the room
In favour 37
Opposed : 1
Abstained : 17

Motion passes.

Motion 12
Move to apply the changes in P1353R0 (Missing Feature Test Macros) to the C++ working paper.

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

Motion 13
Move to apply the changes in P1073R3 (Immediate functions) to the C++ working paper.

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

Motion 14

Move to apply the changes in P0595R2 (`std::is_constant_evaluated()`) to the C++ working paper.

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

Motion 15
Move to apply the changes in P1141R2 (Yet another approach for constrained declarations) to the C++ working paper.

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

Motion 16
Move to apply the changes in P1094R2 (Nested Inline Namespaces) to the C++ working paper.

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

Motion 17
Move to apply the changes in P1330R0 (Changing the active member of a union inside constexpr) to the C++ working paper.

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

Mike Miller : What do we do with the co routines issues that were ready ?
Herb : I'd rather not add more polls. Can we poll them in Kona ?
Mike : Yes

- ## LWG (Clow)

Marshal Clow presents.

- LWG saw about 45 papers
- Moving 25 - one quite large
- Small groups worked quite well for initial review
- Lots of work will be done on the reflector between now and Kona
    - Issue processing
    - Initial paper review
- LWG is planning on having two issue resolution telecoms between now and Kona (probably one in December, one in January)

• LWG still has have a ton of work to do for C++20.
• A copy of these bits will be in an update to P0985 (LWG Chair post-meeting report) in the post-meeting mailing.

LWG Motions

Concurrency TS
Motion 1
Move to apply to the Concurrency TS working paper the proposed resolution of issue 2697 in P1224R0 (C++ Standard Library Issues to be moved in San Diego).

Botond Ballo : is this for Concurrency TS v2 ?
John : yes
No objection to unanimous consent.
Motion passes.

Library Fundamentals TS
Motion 2
Move to apply to the Library Fundamentals TS working paper the proposed resolution of issues 2960 ~~and 3134~~ in P1224R0 (C++ Standard Library Issues to be moved in San Diego).

Marshall: there were two issues in the pre meeting mailing, but we decided not to move 3134 here because the next motions covers that issue.

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

Motion 3
Move to apply to the Library Fundamentals TS working paper the proposed wording in P1210R0 (Completing the Rebase of Library Fundamentals, Version 3, Working Draft). This also resolves LWG issue 3134.

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

Issues
Motion 4
Move to apply to the C++ working paper the proposed resolutions of all of the issues except 2697, 2960 and 3134 in P1224R0 (C++ Standard Library Issues to be moved in San Diego).

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

Draft Standard
Motion 5
Move to apply to the C++ working paper the proposed wording in P1123R0 (Editorial Guidance for merging P0019r8 and P0528r3).

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

Motion 6
Move to apply to the C++ working paper the proposed wording in P0487R1(Fixing operator>>(basic_istream&, CharT*) (LWG 2499)).

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

Motion 7
Move to apply to the C++ working paper the proposed wording in P0602R4(variant and optional should propagate copy/move triviality).

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

Motion 8
Move to apply to the C++ working paper the proposed wording in P0655R1 (visit<R>: Explicit Return Type for visit).

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

Motion 9
Move to apply to the C++ working paper the proposed wording in P0972R0 (<chrono> zero(), min(), and max() should be noexcept).

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

Motion 10
Move to apply to the C++ working paper the proposed wording in P1006R1 (Constexpr in std::pointer_traits).

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

Motion 11
Move to apply to the C++ working paper the proposed wording in P1032R1 (Misc constexpr bits).

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

Motion 12
Move to apply to the C++ working paper the proposed wording in P1148R0 (Cleaning up Clause 20).

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

Marshall : LWG will meet after the plenary. We will do paper review or issues review, depending on the number of people. Poll for the number of people. Enough people to split into little groups.

Titus: LEWG will meet if we have enough people. Poll for people. There is a quorum.

Ville : EWG if we have enough people. Poll for people. There is a quorum.

Motion 13
Move to apply to the C++ working paper the proposed wording in P0318R1 (unwrap_ref_decay and unwrap_reference)

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

Motion 14
Move to apply to the C++ working paper the proposed wording in P0357R3 (reference_wrapper for incomplete types)

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

Motion 15
Move to apply to the C++ working paper the proposed wording in P0608R3 (A sane variant converting constructor)

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

Motion 16
Move to apply to the C++ working paper the proposed wording in P0771R1
(std::function move constructor should be noexcept)

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

Motion 17
Move to apply to the C++ working paper the proposed wording in P1007R3
(`std::assume_aligned`)

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

Motion 18
Move to apply to the C++ working paper the proposed wording in P1020R1 (Smart pointer creation with default initialization)

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

Motion 19
Move to apply to the C++ working paper the proposed wording in P1285R0 (Improving Completeness Requirements for Type Traits)

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

Motion 20
Move to apply to the C++ working paper the proposed wording in P1248R1 (Remove CommonReference requirement from StrictWeakOrdering (a.k.a Fixing Relations))

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

Motion 21

Move to apply to the C++ working paper the proposed wording in P0591R4 (Utility functions to implement uses-allocator construction)

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

Motion 22
Move to apply to the C++ working paper the proposed wording in P0899R1 (LWG 3016 is Not a Defect)

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

Motion 23
Move to apply to the C++ working paper the proposed wording in P1085R2 (Should Span be Regular?)

No discussion.
There are objections in the room.
In favour : 30
Opposed : 1
Abstain : 25

Motion passes.

Motion 24
Move to apply to the C++ working paper the proposed wording in P1165R1 (Make stateful allocator propagation more consistent for `operator+(basic_string)`)

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

Motion 25
Move to apply to the C++ working paper the proposed wording in P0896R4 (The One Ranges Proposal)

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

Motion 26
Move to apply to the C++ working paper the proposed wording in P0356R5 (Simplified partial function application).

No discussion.

No objection to unanimous consent.
Motion passes.

Motion 27
Move to apply to the C++ working paper the proposed wording in P0919R3 (Heterogeneous lookup for unordered containers).

No discussion.
There are objections in the room.
In favour : 26
Opposed: 2
Abstain: 25
Motion passes

Motion 28
Move to apply to the C++ working paper the proposed wording in P1209R0
(Adopt Consistent Container Erasure from Library Fundamentals 2 for C++20).

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

◦ Direction Group

Bjarne Stroustrup presents. For the new people, we have been around for about a year. The idea is to look at issues that go across the individual technical issues. We take input, but we can't do everything. If you have things that concern the language direction, contact us.

People ask for more detail of long-term direction. The original paper we wrote P0939Rx is focused on what would be a good direction on C++20, but gets more vague as we get into the future.
Now we are going to work on C++23. We never have clear straight forward answers. We say it would be a good idea to go in a certain direction.
People asked feedback on specific topics which indicates these are hot topics:
- signed/unsigned comparisons (and span)
- define what/who is the target of freestanding/embedded
- static reflection
- pattern matching
- ABI (what is breakage)
- is <=> really an improvement ?
- networking - is it right to delay past c++20 ? Why do we need more design review, when we had a design review a couple of years ago ?

Marshall: the review in Cologne wasn't really a design review, it was a discussion on how it would fit in the current library. It was a minor design review, but mostly a wording review.

Bjarne continues. If you have opinions about how something fits into the big picture, e-mail us, talk to use, we are not meant to be working in isolation. Thank you.

Herb : during the break there were more questions about co routines. We are likely to have another vote in Kona. There are NB's that said may change their vote to yes if they have more time. There will be papers for the next meeting, and they will go to an EWG schedule, and EWG can put the the poll forward again.

7. Closing activities

7.1 Issues delayed until today

No discussion.

7.2. PL22.16 motions, if any

John Spicer presents voting rules.

1) **Approval of Systematic Review for ISO/IEC TS 19217:2015 -- C++ Extensions for Concepts**

Barry Hedquist moves.
Marshall Clow seconds.
In favour : 26 in the room + 3 by e-mail
Opposed : 0
Abstain : 0

Motion passes

Walter Brown presents.
Thank you the host and the sponsors.
Thank you to everyone that participated and those who helped us participate.

7.3 Mailings

- 2018-11-26: Post-San Diego
- 2019-01-21: Pre-Kona

Hal Finkel : web form to get paper number is going to stay up. Please use it over the next couple of weeks
Lawrence Crawl : where is the form ?
Hal : you will find it on the wiki page. You can also email me

8. Adjournment

Marshall Clow moves. Barry Hedquist seconds.
Approved by unanimous consent.
John Spicer adjourns the meeting at 11:01 am PST.

9. Attendance

Name	Representing	NB
Aaron Ballman	GrammaTech Inc	
ADAM David Alan Martin	MongoDB Inc	
Agustin Berge	Louisiana State University	
Alexandru Voicu	AMD	
Alisdair Meredith	Bloomberg	
Andrew Sutton	University of Akron	
Andrew Turner	Qualcomm Inc	
Anna Gringauze	Microsoft Corporation	
Anton Polukhin	Yandex.Taxi	RU
Ashley Hedberg	Google	

Name	Representing	NB
Attila Feher	Bloomberg	
Barry Hedquist	Perennial	
Barry Revzin	Jump Trading	
Ben Craig	National Instruments	
Benjamin Saks	Saks & Associates	
Billy Baker	FlightSafety International	
Billy R. O'Neal III	Microsoft Corporation	
Bjarne Stroustrup	Morgan Stanley	
Blaise Watson		
Bob Steagall	KEWB Computing	
Botond Ballo	Mozilla	CA
Brian Van Straalen	Lawrence Berkeley National Laboratory	
Bruno Lopes	Apple	
Bryce Adelstein Lelbach	NVidia Corporation	
Carter Edwards	NVidia Corporation	
Casey Carter	Microsoft Corporation	
Chandler Carruth	Google	
Chanyoung Park		CA
Chris Kennelly	Google	
Chris Mysen	Google	
Christof Meerwald	Programming Research Ltd	
Christopher Earl	Lawrence Livermore National Laboratory	
Corentin Jabot		FR
Damien Lebrun-Grandie	Oak Ridge National Laboratory	

Name	Representing	NB
Dan Kalowsky		
Daniel Sunderland	Sandia National Laboratories	
Daveed Vandevoorde	Edison Design Group	
David Goldblatt	Facebook	
David Hollman	Sandia National Laboratories	
David Olsen	NVidia Corporation	
David Sankel	Bloomberg	
David Stone	Google	
Davis Herring	Los Alamos National Laboratory	
Detlef Vollmann	Vollmann Engineering	CH
Dietmar Kühl	Bloomberg	
Dmitrii Kozhevnikov	Jetbrains	
Eric Fiselier	Google	
Eric Niebler	Facebook	
Erich Keane	Intel Corporation	
Fabio Fracassi		DE
Faisal Vali		
Frank Birbacher	Bloomberg	
Gabriel Dos Reis	Microsoft Corporation	
Gary Powell	Sony Interactive Entertainment	
Geoffrey Romer	Google	
Gevorg Voskanyan		
Glen Fernandes	C Plus Plus Alliance Inc	
Gor Nishanov	Microsoft Corporation	
Graham Lopez	Oak Ridge National Laboratory	

Name	Representing	NB
Hal Finkel	Argonne National Laboratory	
Hans Boehm	Google	
Herb Sutter	Microsoft Corporation	
Howard Hinnant	Ripple Labs	
Hubert Tong	IBM Corporation	CA
Hyrum Wright	Google	
Iain Sandoe		
J. Daniel García	University Carlos III of Madrid	ES
J.C. van Winkel		NL
James Bennett		
James Dennett	Google	
James Touton	Blizzard	
Jared Hoberock	NVidia Corporation	
Jason Carey	MongoDB Inc	
Jason Merrill	Red Hat Inc	
Jason Rice		
Jean-Francois Bastien	Apple	CA
Jean-Paul Rigault	Université de Nice S.A.	FR
JeanHeyd Meneide		
Jeff Chapman		
Jeff Snyder	PDT Partners	GB
Jeffrey Mendelson	Bloomberg	
Jennifer Yao	Microsoft Corporation	
Jens Maurer	Edison Design Group	
Jerry Coffin		

Name	Representing	NB
John Lakos	Bloomberg	
John Spicer	Edison Design Group	
Jon Kalb	C Plus Plus Alliance Inc	
Jonathan Brian Coe		GB
Jonathan Caves	Microsoft Corporation	
Jonathan Wakely	Red Hat Inc	GB
Jorg Brown		
Juan Alday	GreenWireSoft	
Kelly Walker	Stellar Science	
Kirk Shoop	Facebook	
Lars Gullik Bjønnes	Cisco Systems Inc	
Lawrence Crowl	Perennial	
Lee Howes	Facebook	
Lewis Baker	Facebook	
Li-Ta Lo	Los Alamos National Laboratory	
Lisa Lippincott	Tanium	
Louis Dionne	Apple	CA
Maged Michael	Facebook	
Mark Hoemmen	Sandia National Laboratories	
Mark Zeren	VMware Inc	
Marshall Clow	C Plus Plus Alliance Inc	
Mateusz Pusz	EPAM Systems Inc	PL
Mathias Stearn	MongoDB Inc	
Matt Calabrese	Google	
Matthew Woehlke	Kitware, Inc	

Name	Representing	NB
Matthias Kretz	GSI	DE
Mauro Bianco	CSCS	CH
Michael Caisse		
Michael Garland	NVidia Corporation	
Michael McLaughlin	Bob Taco Industries	
Michael Park		
Michael Spencer	Sony Interactive Entertainment	
Michael Spertus	Symantec	
Michael Wong	Codeplay	CA
Michael Young		
Michał Dominiak	NVidia Corporation	PL
Mihail Mihailov		
Minas Charalambides	UIUC	
Muerte Isabella	Target	
Nathan Burgers		
Nathan Myers	Maystreet	
Nathan Sidwell	Facebook	
Nathaniel Goodspeed	Linden Research, Inc	
Neal Meyer	Amazon Corporate LLC	
Nevin Liber	Ocient	
Nicolai Josuttis		DE
Nicolas Lesser		
Nina Dinka Ranns	Edison Design Group	
Oleg Zhylin		
Olivier Giroux	NVidia Corporation	

Name	Representing	NB
P.J. Plauger	Dinkumware Ltd	
Pablo Halpern	Intel Corporation	
Paul McKenney	IBM Corporation	
Phil Nash	Jetbrains	
Rene Rivera	C Plus Plus Alliance Inc	
Richard Smith	Google	
Robert Douglas	RKR Capital	
Robert Romey		
Robert Simpson	Qualcomm Inc	
Roger Orr		GB
Ronan Keryell	Xilinx	
Ryan McDougall	Zoox Inc	CA
S.W. Holeman II		
Sebastian Büttner		
Sergei Murzin	Bloomberg	
Shuo Feng Liu	IBM Corporation	CA
Sifang Li	Hummingbird Code	
Siu Chi Chan	AMD	
Stephan Lavavej	Microsoft Corporation	
Stéphanie Even	SILKAN	
Stephen Schurr	Ripple Labs	
Tana Plauger	Dinkumware Ltd	
Thomas Köppe	Google	
Thomas Rodgers	Red Hat Inc	
Thomas Scogland	Lawrence Livermore National Laboratory	

Name	Representing	NB
Tim Northover	Apple	
Tim Shen	Google	
Timur Doumler	Jetbrains	GB
Titus Winters	Google	
Tom Honermann	Synopsys Inc	
Tomasz Kamiński	Sabre	PL
Tony Van Eerd	Christie Digital	CA
Tyler Sutton	Lock3 Software	
Victor Zverovich	Facebook	
Ville Voutilainen	Plum Hall Inc	FI
Vincent Falco	C Plus Plus Alliance Inc	
Vincent Reverdy	Paris Observatory	
Walter Brown	Brown	
Wesley Maness	Schonfeld Tools LLC	
Will Deacon	ARM Ltd	GB
William Miller	Edison Design Group	
William Seymour	Seymour	
Wyatt Childers		
Zach Laine	Cadence	
Zhihao Yuan	DePaul University	