

Doc. no.: N4852
Date: 2020-02-15
Reply to: Titus Winters (titus@google.com)
Audience: WG21

Prague 2020 LEWG Summary

This paper is a summary of the activities of the Library Evolution Working Group during the recent WG21 meeting in Prague.

In brief: we discussed 41 papers during the meeting. About half of our time was spent on papers that are design and policy direction for C++23 design fixes, with the rest of the time spent resolving C++20 issues and discussing Executors. There are approximately 40 papers that are being tracked that have not been discussed by either LEWG or LEWGI.

Acknowledgements

My most profound thanks to everyone that participated in the discussions during the week. I continue to be deeply pleased with the quality and professionalism of our design discussions.

Above and beyond that general gratitude, I must of course also mention the various scribes that helped take notes on these discussions throughout the week. Scribes are critical for this work, and I am very grateful to them for their efforts.

Chair Tenure

Contrary to the date provided in N4845, the upcoming meeting in Varna will be my last meeting as LEWG chair. I have begun the process of trying new potential chairs and anticipate announcements about a replacement soon.

Commitments/Homework

A number of people volunteered to write follow-up papers or otherwise help make progress on some in-flight proposals during the week. The following is a listing of those promises.

- CJ Johnson - Follow up P1655R0 with proposed wording for a new SD describing LEWG design policy.

Policy Discussions

- (Approved) P1999 - Process proposal: double-check evolutionary material via a Tentatively Ready status

- The effect of this is that any non-obvious design work will be parked with a TentativelyReady tag ([matching papers](#)) between meetings, as well as noted in these summaries. These will not advance to the LWG review queue until the beginning of the next meeting. (The LWG chair will be informed of these upcoming papers and may still schedule their review.)
- (Approved) P0592 - To boldly suggest an overall plan for C++23
 - This paper suggests a priority list for features in C++23.
- (Discussed) P2028 - What is ABI and What Should WG21 Do About It? / P1863 - ABI, Now or Never
 - No consensus to coordinate an overall ABI break for C++23. Interest in discussing ABI breaks going forward.
- (Discussed) P2027 - Moved-from objects need not be valid
 - Ville Voutilainen, Jonathan Wakely, Geoffrey Romer, and Andrew Sutton will author a paper proposing a fix to the `move_constructible` concept, which is believed to currently be over-constraining user-defined types.
- (Approved) P1851 - Guidelines For `snake_case` Concept Naming
 - This formalizes the guidelines LEWG has been using for naming concepts since the proposal to move to `snake_case` was approved in Cologne.
- (Approved) P1919 - Expanding the Rights in SD-8
 - This clarifies wording in [SD-8](#) around our additions to the standard library.
- (Weakly endorsed) P2070 - A case for `optional<T&>` and `object_ptr`
 - We had weak consensus to continue work in this space, and encouraged papers on `optional<T&>` and `object_ptr`.
- (Approved) P1656 - Throws: Nothing should be `noexcept`
 - We overturned a long-standing design rule (often referred to as the Lakos Rule). Functions with preconditions that are documented with `Throws: Nothing` are now allowed to be specified with `noexcept`
- (Approved) P1000R4 - C++ IS schedule
 - We approved the meeting schedule and design cut-offs for C++23.
- (Approved) P1732 - Do not promise support for function syntax of operators
 - We approved an update to [SD-8](#) regarding unusual invocations of operators.
- (Approved) P1768 - Contiguous Containers Should Contain `.data()`
 - We approved a general design policy for contiguous containers.
- (Declined) P2065 - naming and aliases
 - We declined a proposal about short namespaces aliases in the standard library.
- P2035 - Value Proposition: Allocator-Aware (AA) Software / P2080 - Polymorphic allocators: There is no such thing as One True Vocabulary Type
 - We reaffirm that we support allocator awareness even for non-allocating types.

Papers Marked Tentatively Ready

These will be forwarded to LWG en masse at the opening of the next meeting.

- P1901 - Enabling the Use of `weak_ptr` as Keys in Unordered Associative Containers ([issue](#))
- P1413 - Deprecate `std::aligned_storage` and `std::aligned_union` ([issue](#))
- P1450 - Enriching type modification traits ([issue](#))

Papers To Be Reviewed for Wording and then Marked Tentatively Ready

Wording experts have been assigned to these. When wording is prepared and reviewed, these will be marked Tentatively Ready (as above).

- P0401 - Providing size feedback in the Allocator interface ([issue](#))
- P1425 - Iterators pair constructors for stack and queue ([issue](#))

Papers Forwarded to LWG

- P2081 - Rebase the Library Fundamentals v3 TS on C++20 ([issue](#))
- P1202 - Asymmetric Fences ([issue](#)) (Aiming for Concurrency TS 2)

Discussed but not Approved nor Forwarded

- P0901 - Size feedback in operator new
- P0443 - A Unified Executors Proposal for C++
 - We approved (Unanimously) the design for Executors. We are awaiting draft wording, and will do a detailed review of the wording-design when that is available (presumably in Varna).
- P1678 - Callbacks and Composition
- P1619 - Functions for Testing Boundary Conditions on Integer Operations
- P1028 - SG14 `status_code` and standard error object for P0709 Zero-overhead deterministic exceptions
- P0211 - Allocator-aware library wrappers for dynamic allocation
- P2037 - String's gratuitous assignment
- P1899 - `stride_view`
- P1843 - Comparison and Hasher Requirements
- P1790 - Networking TS changes to enable better `DynamicBuffer` composition
 - (Design was approved, but needs to be merged after the TS merges with the WD)
- P1843 - Comparison and Hasher Requirements
- P1406 - Add more `std::hash` specializations
- P1927 - Add `std::is_partitioned_until` algorithm
- P1030 - `std::filesystem::path_view`
 - (High-level design was approved, but needs a review of the wording design)