C and C++ Compatibility Study Group Meeting Minutes (Mar 2022)

Reply-to: Aaron Ballman (aaron@aaronballman.com)
Document No: N2944
SG Meeting Date: 2022-03-04

Fri Mar 04, 2022 at 1:05pm EST

Attendees

Aaron Ballman	WG21/WG14	chair
Philipp K. Krause	WG14	
Robert Seacord	WG14	
Hans Boehm	WG21	
JeanHeyd Meneide	WG21/WG14	co-chair
Corentin Jabot	WG21/(14)	
Joshua Cranmer	(21)	
Gaby Dos Reis	WG21	
Jens Mauer	WG21	scribe
Jens Gustedt	WG14	
Hubert Tong	WG21/(14)	
Michael Wong	WG21/(14)	
Erich Keane	WG21	
Martin Uecker	WG14	
Steve Downey	WG21	
Ryan McDougall	WG21	

Code of Conduct: follows ISO, IEC, and WG21 CoCs (no current WG14-specific CoC)

Agenda

Discussing the following papers:

WG14 N2930 (http://www.open-std.org/jtc1/sc22/wg14/www/docs/n2930.pdf) Consider renaming remove_quals
WG21 P2215R1 (https://wg21.link/p2215r1) Undefined behavior and the concurrency memory model

WG14 N2930 Consider renaming remove_quals

Corentin: C23 introduces remove_quals (in addition to typeof), with the same semantics as typeof, except removing qualifiers. If C++ ever wants to adopt this, it will likely want to remove references, but the name is seriously confusing in that case.

The paper proposes to use "unqual_typeof".

JeanHeyd: I fully support this paper. This is fine for me.

Philipp: Yes, they should rename it not to conflict with C++. Prefer typeof_unqual for symmetry.

Martin, JeanHeyd: Agreed with typeof_unqual.

Hubert: What's the proposed semantics for C++ if/when it happens?

Corentin: We want that operator to remove references. C decided not to adopt decltype; the significance of parentheses for decltype might cause issues with macros.

POLL: Does SG22 recommend that WG14 consider changing the name of remove_quals?

Committee	For	Against	Abstain	Notes
WG14	8	0	0	Unanimous consent
WG21	7	0	0	Unanimous consent

Overall: Unanimous consent

WG21 P2215R1 Undefined behavior and the concurrency memory model

Related proposal: Proposal "P1494 Partial program correctness" by Davis Herring, https://wg21.link/P1494R2 Failed to achieve consensus in WG21/EWG.

Hans: Time-travel undefined behavior has bad interactions with concurrency.

This is work in progress.

Martin: C and C++ seem to have a slightly different understanding of undefined behavior. I failed to find actual examples of time-travel undefined behavior.

No polls were taken.

Wrapup

Jens G: Can we pick a new time for meetings that's easier for Europeans?

Aaron: I'll send out a Doodle poll and see when the group wants to meet for summer hours.

End at 1:54pm EST