

WG21 2022-07 Virtual Meeting

Record of Discussion

ISO/IEC JTC1 SC22 WG21 P2627— 2022-08-09

Nina Dinka Ranns, dinka.ranns_at_gmail.com

Chair: John Spicer

Monday 2022-07-25

08:00 N.Am. Pacific Time

1. Opening activities

John Spicer opens the meeting at 08:03 AM N.Am. Pacific Time

1.1 Opening comments (PL22.16)

John Spicer presents.

1.2 Meeting guidelines

John Spicer presents.

Meetings are not public, we want everyone to be able to speak freely. Please refrain from live tweeting, blogging, taking photos or videos.

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 INCITS Code of Conduct \(PL22.16\)](#)

[The IEC Code of Conduct](#)

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

John Spicer presents the slides. They are also linked in the agenda.

If you have any questions or concerns about CoC issues, please approach a committee officer or a NB representative and bring it to their attention.

1.3 The ISO Code of Conduct

John Spicer presents CoC slides.

The primary purpose of CoC is to ensure we can get our work done in a productive way, and make sure that disrespectful and abusive behavior is avoided.

1.4 Membership, voting rights, and procedures for the meeting (PL22.16)

John Spicer presents voting rights.

If you are a non US member registered in the global directory, you get one vote.

If you are a US member, each company gets one vote.

If you are representing an organization that is considering formally joining PL22.16, or your organization is already a member and you wish to change your voting status, please inform an officer - John Spicer, Hal Finkel, and Barry Hedquist.

Hal Finkel presents. We don't use the attendance sheet for virtual meetings, please make sure your details are correct in the zoom client. If there are any issues with the recorded attendance, please let me know. We have a paper handling system for creating document numbers. If you need any assistance please send me an e-mail.

John Spicer presents how to vote using the telecon client.

1.5 Introductions

New members introduce themselves.

John Spicer welcomes new members.

1.6 Agenda review and approval (PL22.16 motion, WG21 poll)

John Spicer presents the agenda.

The meeting goals described above are derived from the schedule adopted in 2020 and described in: [P1000R4](#).

PL22/16 motion to approve the meeting agenda

Bryce Adelstein Lelbach moves

Casey Carter seconds.

The motion is unanimously approved by PL22/16.

WG21 motion to approve the meeting agenda.

The motion is unanimously approved by WG21.

1.7 Editor's reports, approval of working drafts

Document	Editor's report	Prospective WD
C++23 Standard	N4911	N4910
Library Fundamentals Version 3	N4909	N4908

WG21 motion to approve the working drafts.
The motion is unanimously approved by WG21.

1.8 Approval of the minutes of the previous meetings (PL22.16 motion, WG21 poll)

Meeting	Minutes
WG21 February Virtual	N4907
PL22.16 February Virtual	pl22.16-2022-00014
WG21 pre-July Virtual administrative telecon	N4914

PL22/16 motion to approve the minutes.
Bryce Adelstein Lelbach moves.
Casey Carter seconds.
The motion is unanimously approved by PL22/16.

Hubert Tong : The minutes refer to the Concurrency TS2 Editor's Report, but there is no number for the document. We can give it a number or minute the content of the Editor's report.

WG21 motion to approve the minutes with amendment to include the information from the Concurrency TS2 Editor's Report :

N4895 is the proposed working draft of Concurrency TS Version 2. It contains changes to the Concurrency TS as directed by the committee at the June 2021 virtual plenary meeting, and editorial changes.

N4895 contains P1121R3 and P1122R4 from the June 2021 virtual plenary.

Thank you to the Editing team of Michael Wong, Paul McKenney, Maged Michael, and Jens Maurer.

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 (Core, Evolution, Library, Library Evolution; see pre-meeting WG21 telecon minutes)

No discussion.

4. New business requiring action by the committee

Herb Sutter presents.

We now need to follow ISO rules strictly for observers and guests. Those rules are that the convenor can invite a guest for one meeting (such as Kona) and I must notify the guest's national body. If that national body objects, the guest is not allowed to participate. After one meeting, the guest must join some national body to keep attending meetings. I know that various national bodies are looking into their national procedures to see how they can be of help, but note these procedures are different for every national body so questions about a specific one should be directed to that national body's chair.

One option is to join a US national body, even if you are not in the US. You will not be able to participate in the US position if your company is not US domiciled. There are fees to pay to join, however.

5. Discussion and Straw Polls

WG and SG status and progress reports should be mailed to meeting@ no later than 7 days before the start of the meeting.

Plenary straw polls must be posted on the meeting wiki Straw Polls no later than 7 days before the start of the meeting.

Presentation and discussion of proposals to be considered for consensus adoption by full WG21.

5.1 CWG Polls

Jens Maurer presents. The difference between the first two polls is that the second poll has issues that do not refer to a released standard.

1. Accept as Defect Reports all issues except 2507 and 2586 in [P2622R0](#) (Core Language Working Group "ready" Issues for the July, 2022 meeting) and apply their proposed resolutions to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

2. Apply the proposed resolution of issues 2507 and 2586 in [P2622R0](#) (Core Language Working Group "ready" Issues for the July, 2022 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 [P2468R2](#)  (The Equality Operator You Are Looking For) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

4. Accept as a Defect Report and apply the changes in [P2327R1](#) (De-deprecating volatile compound operations) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

5. Apply the changes in [P2437R1](#) (Support for #warning) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

6. Apply the changes in [P2362R3](#) (Remove non-encodable wide character literals and multicharacter wide character literals) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

7. Apply the changes in [P2324R2](#) (Labels at the end of compound statements (C compatibility)) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

8. Apply the changes in [P2290R3](#) (Delimited escape sequences) to the C++ Working Paper.

No discussion.

Objection to unanimous consent.

Herb Sutter reminds of the voting rules.

In favor : 38
Opposed : 0
Abstain : 15

Motion passes.

9. Apply the changes in [P2448R2](#) (Relaxing some constexpr restrictions) to the C++ Working Paper.

No discussion.

Objection to unanimous consent.

Herb Sutter reminds of the voting rules.

In favor : 46
Opposed : 1
Abstain : 12

Motion passes

10. Apply the changes in [P2266R3](#) (Simpler implicit move) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

11. Apply the changes in [P2071R2](#) (Named universal character escapes) to the C++ Working Paper.

Mike Miller : We plan to object because we think that it has a noticeable impact on the storage footprint of the compiler and that it is a relatively obscure feature for which there exist workarounds.

Herb Sutter : Was this discussed in EWG?

Erich Keane : Yes, it was. The room was not concerned about the overhead after discussion.

Tom Honermann : the paper does address this concern. Corentin implemented this in clang recently and may have information about the overhead incurred.

Nicolai Josuttis : I would like to understand what overhead we're talking about.

Erich Keane : overhead in disk size taken by the database of mapping between named character escapes and the unicode code points.

Nicolai Josuttis : For each program that is run ?

Erich Keane : For the compiler.

Objection to unanimous consent.

In favor : 37

Opposed : 4

Abstain : 21

Motion passes

12. Apply the changes in [P1169R4](#) (static operator()) to the C++ Working Paper.

Mike Miller : With this change the impact of taking an address of the call operator depends on what kind of class it is in. It may negatively impact genericity in template instances.

Nico Josuttis : Was this discussed in EWG ?

Erich Keane : I don't think so.

Antony Polukhin : Can you describe the problem ?

Mike Miller : Taking the address of the call operator would always be a pointer to a member function. Now you might get a regular function pointer or a pointer to member function. This might have an effect on the generic code.

Michal Dominiak : Would taking the address of a call operator that was defined using deduced this have the same problem ?

Gašper Ažman : Yes, this is already true of deducing this.

Objection to unanimous consent.

In favor : 34

Opposed : 4

Abstain : 23

Motion passes

13. Accept as a Defect Report and apply the changes in [P2280R4](#) (Using unknown pointers and references in constant expressions) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

14. Apply the changes in [P1467R9](#) (Extended floating-point types and standard names) to the C++ Working Paper.

Erich Keane : There was concern in the room regarding the lack of implementation experience with the overload resolution changes this might cause.

David Olsen : There was experience after the EWG poll was taken. It is described in the paper.

Hubert Tong : Is that implementation or deployment experience?

David Olsen : Implementation experience.

Herb Sutter : Erich, are you saying this was discussed in EWG ?

Erich Keane : Correct. It was mentioned in the comments of electronic polling.

David Olsen : We did the work to get the implementation experience after those comments.

Objection to unanimous consent.

In favor : 35

Opposed : 3

Abstain : 20

Motion passes.

15. Accept as a Defect Report [P2493R0](#) (Missing feature test macros for C++20 core papers).

(This paper was adopted at the February, 2022 meeting.)

Bryce Adelstein Lelbach : Can we move slower so companies with multiple representatives can share views regarding concerns that are raised in this meeting ?

John Spicer : We advise people to raise concerns about polls before the meeting. Please let us know if you need more time to discuss among yourself before running the poll.

Jens Maurer : This paper does not change the working draft. It modifies the status of a change already made to a DR.

No objection to unanimous consent.

Motion passes.

16. Apply the changes in [P2582R1](#) (Wording for class template argument deduction from inherited constructors) to the C++ Working Paper.

Mike Miller : Is there implementation experience with this change ?

Erich Keane: This paper was not discussed in EWG during my time.

Timur Doumler : I don't think so.

John Spicer : Did this paper go through EWG?

Hubert Tong : This paper went through EWG in the previous cycle, but the wording was delayed.

Timur Doumler/Jens Maurer : This design went through EWG and was approved for C++20. The wording didn't get through CWG until now.

Objection to unanimous consent.

In favor : 29

Opposed : 3

Abstain : 25

Motion passes.

17. Apply the changes in [P1774R8](#) (Portable assumptions) to the C++ Working Paper.

Gabriel Dos Reis : Has this been implemented or deployed?

Jonathan Caves : We implemented it, but we removed it because it led to some crashes. We tried it and our experience was poor.

Timur Doumler : In gcc and clang we had assumptions for a decade.

Gabriel Dos Reis : I am asking about this current specification. I understand the current specification leads to ABI breakage

Timur Doumler : The syntax is different, but the functionality is the same.

Gabriel Dos Reis : Are you aware of ABI implications of the current specification ?

Timur Doumler : Concerns were raised, the paper went back to EWG and we discussed it quite extensively.

Erich Keane : We had extensive discussions regarding ABI implications of ODR use in template instantiations of expressions in the assumption and what level of ignorance the compiler is allowed to have for unsupported attributes.

Gabriel Dos Reis : What was the sentiment for implementers ?

Chandler Caruth : The paper says that the ABI implications are the same as what is currently implemented.

Gabriel Dos Reis : I am told clang does not implement this feature as specified.

Timur Doumler : The difference is in the syntax.

Bjarne Stroustrup : This change will make it more popular to make assumptions. As a more popular feature I fear it to be a source of bad crashes.

John Spicer : Can chairs comment on whether these issues have been considered ?

Erich Keane : There is no new information in this discussion.

Jens Maurer : I have nothing to add.

John Lakos : This would give us experience as we proceed with contracts.

Objection to unanimous consent.

In favor : 28
Opposed : 10
Abstain : 22

Herb Sutter polls national bodies.

In favor : 7
Opposed : 1
Abstain : 8

Motion passes.

18. Apply the changes in [P2295R6](#) (Support for UTF-8 as a portable source file encoding) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

19. Accept as a Defect Report and apply the changes in [P2513R3](#) (char8_t Compatibility and Portability Fix) to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

20. Accept as a Defect Report and apply the changes in [P2460R2](#) (Relax requirements on wchar_t to match existing practices) to the C++ Working Paper.

No discussion.

Objection to unanimous consent.

In favor : 31
Opposed : 0
Abstain : 24

Motion passes

21. Accept as a Defect Report and apply the changes in [P2579R0](#) (Mitigation strategies for P2036 "Changing scope for lambda trailing-return-type") to the C++ Working Paper.

No discussion.

No objection to unanimous consent.

Motion passes.

22. Apply the changes in [P2617R0](#) (Responses to NB comments on DTS 12907 "Extensions to C++ for Transactional Memory Version 2") to the Working Paper for Extensions to C++ for Transactional Memory Version 2.

No discussion.

No objection to unanimous consent.

Motion passes.

5.2 LWG Polls

1. Apply the changes for all Ready issues in [P2618R0](#) (C++ Standard Library Issues to be moved in Virtual Plenary, Jul. 2022) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

2. Apply the changes in [P0009R18](#) (MDSPAN) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

3. Apply the changes in [P2599R2](#) (`index_type` & `size_type` in `mdspan`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

4. Apply the changes in [P2604R0](#) (`mdspan`: rename `pointer` and `contiguous`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

5. Apply the changes in [P2613R1](#) (Add the missing `empty` to `mdspan`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

6. Apply the changes in [P0429R9](#) (A Standard `flat_map`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

7. Apply the changes in [P1222R4](#) (A standard `flat_set`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

8. Apply the changes in [P1223R5](#) (`find_last`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

9. Apply the changes in [P1642R11](#) (Freestanding Library: Easy [utilities], [ranges], and [iterators]) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

10. Apply the changes in [P1899R3](#) (`stride_view`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

11. Apply the changes in [P2093R14](#) (Formatted output) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

12. Apply the changes in [P2165R4](#) (Compatibility between `tuple` and tuple-like objects) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

13. Apply the changes in [P2278R4](#) (`cbegin` should always return a constant iterator) to the C++ working paper.

Nico Josuttis : I am surprised this is not a defect against C++20. What is the reason for that ?

Jonathan Wakely : This is a very large new feature which is addressing a problem that was raised against C++20, but it was an intentional design in C++20. If you want it as a DR, we can discuss it once it's in the draft.

Nico Josuttis : Can we have a vote to have it as a DR ?

Jonathan Wakely : Not now.

Timur Doumler : You can raise an NB comment to make it a DR.

Herb Sutter : NB comment is only for C++23. We can do DRs at any time if someone wants it.

Mike Miller : Can we amend polls ?

Jonathan Wakely : We don't have guidance from the design group to do so.

Bryce Adelstein Lelbach : We haven't done official DR's in LEWG in a while. This issue was done late in the cycle so we didn't do a specific vote.

John Spicer : We have just moved a paper to a DR in core polls. We have a precedent for making something a DR at a later point in time.

Bryce Adelstein Lelbach : We did discuss the implications this would have in C++20. It would not be unreasonable to vote this as a DR. We can do so in the next plenary.

No objection to unanimous consent.

Motion passes.

14. Apply the changes in [P2286R8](#) (Formatting Ranges) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

15. Apply the changes in [P2291R3](#) (Add `constexpr` Modifiers to Functions `to_chars` and `from_chars` for Integral Types in `<charconv>` Header) to the C++ working paper.

Jonathan Wakely : The full name of the paper is not visible on the poll page.

Nina Ranns : The minutes will reflect the full name.

No objection to unanimous consent.

Motion passes.

16. Apply the changes in [P2302R4](#) (`std::ranges::contains`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

17. Apply the changes in [P2322R6](#) (`ranges::fold`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

18. Apply the changes in [P2374R4](#) (`views::cartesian_product`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

19. Apply the changes in [P2540R1](#) (Empty Product for certain Views) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

20. Apply the changes in [P2404R3](#) (Move-only types for `equality_comparable_with`, `totally_ordered_with`, and `three_way_comparable_with`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

21. Apply the changes in [P2408R5](#) (Ranges iterators as inputs to non-Ranges algorithms) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

22. Apply the changes in [P2417R2](#) (A more constexpr bitset) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

23. Apply the changes in [P2419R2](#) (Clarify handling of encodings in localized formatting of chrono types) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

24. Apply the changes in [P2438R2](#) (`std::string::substr() &&`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

25. Apply the changes in [P2446R2](#) (`views::as_rvalue`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

26. Apply the changes in [P2465R3](#) (Standard Library Modules `std` and `std.compat`) to the C++ working paper.

No discussion.

Objection to unanimous consent.

In favor : 47

Opposed : 1

Abstain : 12

Motion passes.

27. Apply the changes in [P2445R1](#) (`std::forward_like`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

28. Apply the changes in [P2467R1](#) (Support exclusive mode for `fstreams`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

29. Apply the changes in [P2474R2](#) (`views::repeat`) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

30. Apply the changes in [P2494R2](#) (Relaxing range adaptors to allow for move only types) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

31. Apply the changes in [P2499R0](#) (string_view range constructor should be explicit) to the C++ working paper.

No discussion.

Objection to unanimous consent.

In favor : 34
Opposed : 0
Abstain : 23

Motion passes.

32. Apply the changes in [P2502R2](#) (std::generator: Synchronous Coroutine Generator for Ranges) to the C++ working paper.

No discussion.

Objection to unanimous consent.

In favor : 36
Opposed : 1
Abstain : 21

Motion passes.

33. Apply the changes in [P2508R1](#) (Exposing std::basic-format-string<charT, Args...>) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

34. Apply the changes in [P2517R1](#) (Add a conditional noexcept specification to std::apply) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

35. Apply the changes in [P2520R0](#) (`move_iterator<T*>` should be a random access iterator) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

36. Apply the changes in [P2549R1](#) (`std::unexpected` should have `error()` as member accessor) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

37. Apply the changes in [P2585R1](#) (Improving default container formatting) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

38. Apply the changes in [P2590R2](#) (Explicit lifetime management) to the C++ working paper.

No discussion.

No objection to unanimous consent.

Motion passes.

5.3 WG21 Polls

Poll 1

Appoint a review committee composed of Hans Boehm, Steve Downey, and Jens Maurer to approve the correctness of the Working Paper for Extensions to C++ for Transactional Memory Version 2 as modified by the polls approved at this meeting, and direct the Convener to transmit the approved updated Working Paper for publication.

No discussion.

No objection to unanimous consent.

Motion passes.

Poll 2

Move to appoint an editing committee composed of Daniel Krügler, Davis Herring, Nina Ranns, and Erich Keane to approve the correctness of the C++ Working Paper as modified by the polls approved at this meeting, and to direct the Convener to transmit the approved updated working paper for CD ballot.

No discussion.

No objection to unanimous consent.

Motion passes.

John Spicer : thank you to everyone for working under challenging circumstances during the pandemic.

6. Closing activities

6.1 Other business

No discussion

6.2 PL2.16 motions, if any

No discussion.

7 Plans for the future (PL22.16)

7.1 Next and following meetings

Herb Sutter presents. Next meeting is in November, Kona. There will be a zoom component to that meeting. We're still figuring out the details, but we are gated on the number of people willing to do zoom hosting support.

We may try a dry run with an SG14 meeting in September, if they have one.

We are planning three meetings in 2023. First one has a tentative host in the North East US. We are still looking for funding for the first 2023 meeting. If you are able to help, please send me and John Spicer an email.

In the summer we have a tentative meeting in Europe. The November 2023 meeting will be in Kona again.

Before the pandemic, we had a long pipeline of meeting hosts. This has changed because of the uncertainty caused by the pandemic. There is now an opportunity for new meeting hosts. Please let me know if you are interested in hosting a meeting.

- 2022-11-07 to 12 Kona, HI, USA
- 2023-0?-?? -- Winter TBD

7.2 Mailings

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

- 2022-08-15: Post-July
- 2022-10-15: Pre-Kona

Paul Preney: for the Kona meeting, is there an idea how zoom balancing will work out ? What can be said about that ?

Herb Sutter : we are gated on time of the people willing to do zoom hosting duty. Chairs wanted EWG and LEWG to have first priority, then the wording and SG subgroups. We are aiming to cover as many sessions as possible, but if we can't, we will prefer EWG and LEWG.

8. Adjournment (PL22.16 motion)

Bryce Adelstein Lelbach moves.

Casey Carter seconds.

John Spicer adjourns the meeting at 10:40 AM N.Am. Pacific Time

9. Attendance

Name	Country	Organization
Aaron Ballman	US	Intel Corporation

Alisdair Meredith	US	Bloomberg
Andreas Fertig	Germany	
Andreas Weis	Germany	
Antony Polukhin	Russia	
Barry Revzin	US	Jump Trading LLC
Ben Craig	US	National Instruments
Bernhard Manfred Gruber	Switzerland	
Bill Ash	US	INCITS Secretariat
Billy Baker	US	FlightSafety International
Bjarne Stroustrup	US	Morgan Stanley
Björn Andersson	Sweden	
Bronek Kozicki	United Kingdom	
Bryan St. Amour	Canada	
Casey Carter	US	Microsoft Corporation
Christof Meerwald	Austria	
Chuanqi Xu	China	
Corentin Jabot	France	
Daniela Engert		
David Olsen	US	NVIDIA
David Sankel	US	Adobe Systems Inc
David Tenty	Canada	
Davis Herring	US	Los Alamos National Laboratory
Detlef Vollmann	Switzerland	
Dietmar Kuhl	US	Bloomberg
Erich Keane	US	Intel Corporation
Fabio Fracassi	Germany	

Federico Kircheis	Italy	
Frank Birbacher	US	Bloomberg
Gabriel Dos Reis	US	Microsoft
Gašper Ažman	United Kingdom	
Gonzalo Brito	Germany	NVIDIA
Hana Dusíková	Czech Republic	
Herb Sutter	US	Microsoft Corporation
Howard Hinnant	US	Ripple Labs
Hubert Tong	Canada	IBM Corporation
Hui Xie	United Kingdom	
Inbal Levi	Israel	
JC van Winkel	Netherlands	
Jean-Paul Rigault	France	
Jeff Snyder	United Kingdom	
Jens Maurer	US	Edison Design Group
John Spicer	US	
Jonas Persson	Sweden	
Jonathan Caves	US	Microsoft Corporation
Jonathan Wakely	United Kingdom	IBM Corporation
Jose Alcorta		
Jose Daniel Garcia Sanchez	Spain	
Joshua Berne	US	Bloomberg
Juan Alday	US	GreenWireSoft
Kelly Walker	US	Stellar Science
Krzysztof Wiśniewski	Poland	
Loïc Joly	France	

Louis Dionne	Canada	Apple
Mark Hoemmen	US	NVIDIA
Mateusz Pusz	Poland	EPAM Systems Inc
Matthew Butler		
Matthias Kretz	Germany	
Michael Adams	Canada	
Michael Garland	US	NVIDIA
Michael Hava	Austria	
Michael Wong	Canada	
William M. Miller	US	Edison Design Group
Mingxin Wang	China	
Mungo Gill	Ireland	
Nemanja Boric	US	Amazon Corporate LLC
Nicolai Josuttis	Germany	
Nina Ranns	United Kingdom	Edison Design Group
Olivier Giroux	US	Apple
Pablo Halpern	US	Halpern-Wight Inc
Patrice Roy	Canada	
Paul Preney	Canada	
Peter Brett	United Kingdom	
Peter Kulczycki	Austria	
Phil Ratzloff	US	SAS Institute Inc
René Ferdinand Rivera Morell	US	The C Plus Plus Alliance Inc
Richard Corden	US	Programming Research Ltd
Robert Douglas	US	Aquatic Group LLC
Roger Orr	United Kingdom	

Stephen S. Schurr	US	Ripple Labs
Sebastian Büttner		
Steve Downey	US	Bloomberg
Timur Doumler	United Kingdom	
Tom Honermann	US	Intel Corporation
Victor Zverovich		
Walter E Brown	US	(Emeritus)