Document Number: P3151R0
Date: 2024-2-15
Authors: Michael Wong
Project: Programming Language C++, SG19 Machine Learning
Reply to: Michael Wong <michael@codeplay.com>

SG19: Machine Learning virtual Meeting Minutes to 2023/12/14-2024/02/8

Contents

Minutes for 2023/12/14 SG19 Conference Call

On Wed, Dec 13, 2023 at 1:30 AM Michael Wong <fraggamuffin_at_[hidden]> wrote:
> Hi all, this is the SG19 Machine Learning meeting will focus on graphs.
> Phil has attached the following paper:
> D1709R4.pdf <https://lists.isocpp.org/sg19/att-0312/D1709R4.pdf>
> We can also have an update on stats.
> Are there any other suggested topics?
> Thank you.
>
>
> Michael Wong is inviting
> you to a scheduled Zoom meeting.
>
> Topic: SG19 monthly
> Time: 02:00 PM Eastern Time (US and Canada)
> Every month on the Second Thu,
>
>
> Join from PC, Mac, Linux, iOS or Android:
>
> https://iso.zoom.us/j/93084591725?pwd=K3QxZjJlcnljaE13ZWU5cTlLNkx0Zz09
> Password: 035530

>
> Or iPhone one-tap :
> US: +13017158592,,93084591725# or +13126266799,,93084591725#
> Or Telephone:
> Dial(for higher quality, dial a number based on your current location):
> US: +1 301 715 8592 or +1 312 626 6799 or +1 346 248 7799 or +1
> 408 638 0968 or +1 646 876 9923 or +1 669 900 6833 or +1 253 215 8782
> or 877 853 5247 (Toll Free)
> Meeting ID: 930 8459 1725
> Password: 035530
> International numbers available: https://iso.zoom.us/u/agewu4X97
>
> Or Skype for Business (Lync):
> https://iso.zoom.us/skype/93084591725
>
> Agenda:
>
> 1. Opening and introductions
>
> The ISO Code of conduct:
> https://www.iso.org/files/live/sites/isoorg/files/store/en/PUB100397.pdf
>
> IEC Code of Conduct:
>
> https://www.iec.ch/basecamp/iec-code-conduct-technical-work
>
> ISO patent policy.
>
>
>
https://isotc.iso.org/livelink/livelink/fetch/2000/2122/3770791/Common_Policy.htm?no
deid=6344764&vernum=-2
>
> The WG21 Practices and Procedures and Code of Conduct:
>
> https://isocpp.org/std/standing-documents/sd-4-wg21-practices-and-procedures
>
> 1.1 Roll call of participants
>
Phil Ratzloff, Andrew Lumsdaine , Richard Dosselmann, Michael Wong, Scott
McMilan

>
>
> 1.2 Adopt agenda
>
> 1.3 Approve minutes from previous meeting, and approve publishing
> previously approved minutes to ISOCPP.org
>
> 1.4 Action items from previous meetings
>
> 2. Main issues (125 min)
>
> 2.1 General logistics
>
> Meeting plan, focus on one paper per meeting but does not preclude other
> paper updates:
>
> Vice chair Nomination
>

Phil Ratzloff
Andrew Lumsdaine

>
> C++23 and C++26 status
>

C++26:
Hazarda Ptrs
RCU
Parallel TS 2 SIMD
Tokyo meeting:
https://isocpp.org/files/papers/N4961.pdf
https://www.google.com/maps?cid=5380832602234125438
Face-to-Face location
Belle Salle Tokyo Nihonbashi
Japan
〒103-6005 Tokyo
Chuo City
Nihonbashi
2-chōme−7−1 東京日本橋タワー B2・4F・5F
<https://maps.google.com/?cid=5380832602234125438>

<https://maps.google.com/?cid=5380832602234125438>Paper publication dates before TOkyo <https://maps.google.com/?cid=5380832602234125438> <https://maps.google.com/?cid=5380832602234125438>Aim to publish next 3 dates, but vote out in February to go to SG6 and LEWG concurrently. <https://maps.google.com/?cid=5380832602234125438>Dec 18 <https://maps.google.com/?cid=5380832602234125438>Jan 15 <https://maps.google.com/?cid=5380832602234125438>Feb 15 <https://maps.google.com/?cid=5380832602234125438>

,

>
>
>
> Apr 13: Graph
> May 11: Cancelled
> June 15: Varna F2F (cancelled)
> July 13: Stats
> Aug 10: Cancelled
> Sep 14: Graph
> Oct 12: Cancelled
> Nov 9: Kona F2F Cancelled
> Dec 14: Graph
>
> Future meetings
an 11 , Feb 15

>
> ISO meeting status
>
> future C++ Std meetings
>
> 2.2 Paper reviews
>
> 2.2.1: ML topics
>
> 2.2.1.1 Graph Proposal Phil Ratsloff et al
>
> Latest paper:
>
> Here's a link to the paper (different than the previous paper reviewed).

> There are some additional updates I'm planning on making before the
> meeting.
>
>
>
https://docs.google.com/document/d/1OpH-xxRri7tJTtJJIZTYmSHkkrZJkdBwm9zJ7L
qolfQ/edit?usp=sharing
>
>
>
> review of P1709:
Going back to bellman ford: probably need to add a remark to bellman ford
about issues with negative weight cycles (can be detected within N steps I
think but algo will not converge).
Breaking up paper like freestanding
https://docs.google.com/spreadsheets/d/1JnUJBO72QVURttkKr7gn0_WjP--P0vAne8
JBfzbRiy0/edit#gid=0

>
> P1709R3:
>
>
https://docs.google.com/document/d/1kLHhbSTX7j0tPeTYECQFSNx3R35Mu3xO5_d
yYdRy4dM/edit?usp=sharing
>
>
>
https://docs.google.com/document/d/1QkfDzGyfNQKs86y053M0YHOLP6frzhTJqzg1
Ug_vkkE/edit?usp=sharing
>
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html>
>
> <
>
>
https://docs.google.com/document/d/175wIm8o4BNGti0WLq8U6uZORegKVjmnpfc-_
E8PoGS0/edit?ts=5fff27cd#heading=h.9ogkehmdmtel
> *>*
>
> Array copy semantics:
> array copy-semantics paper P1997 "Relaxing Restrictions on Arrays",

> https://wg21.link/p1997
>
> Stats feedback:
>
> P2376R0
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2021/p2376r0.pdf>
> Comments
> on Simple Statistical Functions (p1708r4): Contracts, Exceptions and
> Special cases Johan Lundberg
>
> 2.2.1.2 Reinforcement Learning Larry Lewis Jorge Silva
>
> Reinforcement Learning proposal:
>
> 2.2.1.3 Differential Calculus:
>
>
>
https://docs.google.com/document/d/175wIm8o4BNGti0WLq8U6uZORegKVjmnpfc-_E8PoGS0/edit?ts=5fff27cd#heading=h.9ogkehmdmtel
>
> 2.2.1.4: Stats paper
>
> P2681R0
> <https://www.open-std.org/jtc1/sc22/wg21/docs/papers/2022/p2681r0.pdf>
> More
> Stats Functions Richard Dosselmann, Michael Wong
> Current github
>
> https://github.com/cplusplus/papers/issues/475
>
> https://github.com/cplusplus/papers/issues/979
>
> Stats review Richard Dosselman et al
>
> http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2021/p1708r4.pdf
>
> Feedback from Johan Lundberg and Oleksandr Korval
>
> https://isocpp.org/files/papers/D2376R0.pdf
>

> P1708R3: Math proposal for Machine Learning: 3rd review
>
> PXXXX: combinatorics: 1st Review
>
> *> std.org/jtc1/sc22/wg21/docs/papers/2020/p1708r2
> <http://std.org/jtc1/sc22/wg21/docs/papers/2020/p1708r2>*
> *> above is the stats paper that was reviewed in Prague*
> *> http://wiki.edg.com/bin/view/Wg21prague/P1708R2SG19
> <http://wiki.edg.com/bin/view/Wg21prague/P1708R2SG19>*
> *>*
> *> Review Jolanta Polish feedback.*
> *> http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html>*
>
>
> 2.2.1.4: Matrix paper
>
> 2.2.3 any other proposal for reviews?
>
> 2.3 Other Papers and proposals
>
> P1416R1: SG19 - Linear Algebra for Data Science and Machine Learning
>
>
https://docs.google.com/document/d/1IKUNiUhBgRURW-UkspK7fAAyIhfXuMxjk7xKikK4Yp8/edit#heading=h.tj9hitg7dbtr
>
> P1415: Machine Learning Layered list
>
>
https://docs.google.com/document/d/1elNFdIXWoetbxjO1OKol_Wj8fyi4Z4hogfj5tLVSj64/edit#heading=h.tj9hitg7dbtr
>
> 2.2.2 SG14 Linear Algebra progress:
> Different layers of proposal
>
>
https://docs.google.com/document/d/1poXfr7mUPovJC9ZQ5SDVM_1Nb6oYAXlK_d0ljdUAtSQ/edit
>
> 2.5 Future F2F meetings:

>
> 2.6 future C++ Standard meetings:
> https://isocpp.org/std/meetings-and-participation/upcoming-meetings
>
> None
>
> 3. Any other business
>
> New reflector
>
> http://lists.isocpp.org/mailman/listinfo.cgi/sg19
>
> Old Reflector
> https://groups.google.com/a/isocpp.org/forum/#!newtopic/sg19
> <https://groups.google.com/a/isocpp.org/forum/?fromgroups=#!forum/sg14>
>
> Code and proposal Staging area
>
> 4. Review
>
> 4.1 Review and approve resolutions and issues [e.g., changes to SG's
> working draft]
>
> 4.2 Review action items (5 min)
>
> 5. Closing process
>
> 5.1 Establish next agenda
>
>
> 5.2 Future meeting
> Apr 13: Graph
> May 11: Cancelled
> June 15: Varna F2F (cancelled)
> July 13: Stats
> Aug 10: Cancelled
> Sep 14: Graph
> Oct 12: Cancelled
> Nov 9: Kona F2F Cancelled
> Dec 14: Graph

Minutes for 2024/1/11 SG19 Conference Call

 Minutes from the last meeting.
On Tue, Jan 9, 2024 at 1:57 PM Michael Wong <fraggamuffin_at_[hidden]> wrote:

> Hi all, this is the SG19 Machine Learning meeting that will focus on
> graphs. Phil has attached the following paper:
> D1709R4.pdf <https://lists.isocpp.org/sg19/att-0312/D1709R4.pdf>
> We can also have an update on stats.
> Are there any other suggested topics?
>
> We will finalize sg19 vice chairs Phil Ratzloff and Andrew Lumsdaine.
>
> We will have an SG14 meeting the day before on Wednesday to review Graph:
> Topic: SG14 monthly
> Time: 2nd Wednesdays 02:00 PM Eastern Time (US and Canada)
> Every month on the Second Wed,
>
> Join from PC, Mac, Linux, iOS or Android:
> https://iso.zoom.us/j/93151864365?pwd=aDhOcDNWd2NWdTJuT1IoeXpKbTcydz09
> Password: 789626
> Thank you.
>
>
> Michael Wong is inviting
> you to a scheduled Zoom meeting.
>
> Topic: SG19 monthly
> Time: 02:00 PM Eastern Time (US and Canada)
> Every month on the Second Thu,
>
>
> Join from PC, Mac, Linux, iOS or Android:
> https://iso.zoom.us/j/93084591725?pwd=K3QxZjJIcnljaE13ZWU5cTlLNkx0Zz09
>

> https://iso.zoom.us/j/93084591725?pwd=K3QxZjJIcnljaE13ZWU5cTlLNkx0Zz09
> Password: 035530
>
> Or iPhone one-tap :
> US: +13017158592,,93084591725# or +13126266799,,93084591725#
> Or Telephone:
> Dial(for higher quality, dial a number based on your current location):
> US: +1 301 715 8592 or +1 312 626 6799 or +1 346 248 7799 or +1
> 408 638 0968 or +1 646 876 9923 or +1 669 900 6833 or +1 253 215 8782
> or 877 853 5247 (Toll Free)
> Meeting ID: 930 8459 1725

> Password: 035530
> International numbers available: https://iso.zoom.us/u/agewu4X97
>
> Or Skype for Business (Lync):
> https://iso.zoom.us/skype/93084591725
>
> Agenda:
>
> 1. Opening and introductions
>
> The ISO Code of conduct:
> https://www.iso.org/files/live/sites/isoorg/files/store/en/PUB100397.pdf
>
> IEC Code of Conduct:
>
> https://www.iec.ch/basecamp/iec-code-conduct-technical-work
>
> ISO patent policy.
>
>
>
https://isotc.iso.org/livelink/livelink/fetch/2000/2122/3770791/Common_Policy.htm?nodeid=6344764&vernum=-2
>
> The WG21 Practices and Procedures and Code of Conduct:
>
> https://isocpp.org/std/standing-documents/sd-4-wg21-practices-and-procedures
>
> 1.1 Roll call of participants
>
Richard Dosselman, Boguslaw Cyganek, Michael Wong Guy Davidson, Phil
Ratzloff, Andrew Lumsdaine, Brett, Muhammad Osama, Scott McMillan

>
>
> 1.2 Adopt agenda
>
> 1.3 Approve minutes from previous meeting, and approve publishing
> previously approved minutes to ISOCPP.org
>
> 1.4 Action items from previous meetings
>
> 2. Main issues (125 min)
>
> 2.1 General logistics
>
> Meeting plan, focus on one paper per meeting but does not preclude other paper
> updates.
>
> 2024 planning
> C++23 and C++26 status

\>
\>
\> * Jan 11, 2024 02:00 PM ET: Graph
\> * Feb 8, 2024 02:00 PM ET: Graph
\> * Mar 14, 2024 02:00 PM ET: Cancelled due to Tokyo 3-18-23
\> * Apr 11, 2024 02:00 PM ET: Stats
\> * May 9, 2024 02:00 PM ET: Graph
\> * June 13, 2024 02:00 PM ET: Embedded; St.louis 6-24-29
\> * July 11, 2024 02:00 PM ET: Stats
\> * Aug 15, 2024 02:00 PM ET: Graph
\> * Sep 12, 2024 02:00 PM ET: CPPCON Sept 15-20 so canceled
\> * Oct 10, 2024 02:00 PM ET: Stats
\> * Nov 14, 2024 02:00 PM ET: Cancelled Wroclaw F2F
\> * Dec 12, 2024 02:00 PM ET: Graph
\>
\>
\> ISO meeting status
\>
\> future C++ Std meetings
\>
\> 2.2 Paper reviews
\> Review BSI Graph feedback:
\> As Oliver (Rosten) said "The basic premise is important, and it would be
\> fantastic to have support for graphs in the standard."
\>
\> The main items identified were:
\> Oliver:
\> - This paper is long and incomplete, it has lots of details which I think
\> to be irrelevant, however things that are definitely relevant are missing
\> from the paper - for example definition of graph - since people have
\> different ideas. We need to add a mathematical perspective to the paper.
\>
\> - The structure of the paper completely changed in the new revision, so
\> now it's hard to understand what and why they have done
\>
\> - Another missing part is discussion of graph invariants
\>
\> Tom (Deakin): There's a big missing part in "Prior art" part, GraphBLAS (
\> https://graphblas.org) eminently.
\>
\> Some other things to add:
\>
\> 1. The electrical circuit example needs more explanation, and I think this
\> will highlight some deep issues around representing things which are
\> seemingly trivially graphs, as graphs in practice. In what sense is a
\> bog-standard resistor directed? I assume the reason that the graph is
\> directed is because current has a sign and in an undirected graph it
\> becomes ambiguous which way the current is flowing (also you may want
\> components like diodes). But the directed representation also has issues:
\> "can current flow from 'Vdd' to 'n0'?" should be immediately answerable
\> from the properties of Vdd and its edges. There are other ways to represent

> an electrical circuit. One is as a directed graph but with incident edges
> recorded - but iiuc, this is excluded from the latest version of the paper.
> Alternatively, one could have a mathematical object, the name of which I
> actually don't know: it looks like an undirected graph, but where each
> partial edge has additional, unique, end-point data, as well as the common
> weight. Things like this are the reason why I think we need a broader group
> to look at this proposal (i.e. beyond SG19) and if we possibly can we
> should involve someone from the mathematics community. Otherwise there's a
> real danger we end up missing important insights.
>
> 2. My comment about the structure of the paper changing was a reference to
> previous comparisons with boost::graph. I'm sure these were in an earlier
> version, or am I misremembering? Either way, it would be very helpful to
> have a proper discussion of e.g. the move away from visitors.
>
> 3. Re. the definition of a graph, there needs to be a proper discussion
> about whether the paper's definition of graph is what some authors call a
> multigraph and whether it does/does not include loops. These things are
> mentioned, in passing, when introducing algorithms, but terminology needs
> to be properly established.
>
> 4. I think we're trying to do too much in one go in this paper. I think a
> great first step would be to build on mdspan and try to standardize (or at
> least understand) what might reasonably be called an unstructured span.
> This could be represented as a vector of vectors or as a vector with some
> auxiliary storage indicating where the partitions fall. The point is that
> an unstructured span, with the right invariants, is an adjacency list. If
> we can understand unstructured span and its desirable api, I think this
> will be incredibly valuable guidance for what a standardized graph
> container might look like.
>
> 5. IIUC, this paper excludes pure connectivity graphs. These are
> incredibly helpful and, if I've understood correctly that they are not
> supported, would be a major omission. Another good reason, imo, to start
> with unstructured span!
>
> 6. I'm not convinced by the load api. We don't have a load api for vector
> etc. Moreover, would it not be preferable to have appropriate constructors?
>
>
> 2.2.1: ML topics
>
> 2.2.1.1 Graph Proposal Phil Ratsloff et al
>
> Latest paper:
>
> Here's a link to the paper (different than the previous paper reviewed).
> There are some additional updates I'm planning on making before the
> meeting.
>
>

>
https://docs.google.com/document/d/1OpH-xxRri7tJTtJJIZTYmSHkkrZJkdBwm9zJ7LqolfQ/edit?usp=shar
ing
>
>
>
> P1709R3:
>
>
https://docs.google.com/document/d/1kLHhbSTX7j0tPeTYECQFSNx3R35Mu3xO5_dyYdRy4dM/edit?us
p=sharing
>
>
https://docs.google.com/document/d/1QkfDzGyfNQKs86y053M0YHOLP6frzhTJqzg1Ug_vkkE/edit?usp=
sharing
>
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html>
>
> <
>
>
https://docs.google.com/document/d/175wIm8o4BNGti0WLq8U6uZORegKVjmnpfc-_E8PoGS0/edit?ts=5
fff27cd#heading=h.9ogkehmdmtel
> *>*
>
> Array copy semantics:
> array copy-semantics paper P1997 "Relaxing Restrictions on Arrays",
> https://wg21.link/p1997
>
> Stats feedback:
>
> P2376R0
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2021/p2376r0.pdf>
> Comments
> on Simple Statistical Functions (p1708r4): Contracts, Exceptions and
> Special cases Johan Lundberg
>
> 2.2.1.2 Reinforcement Learning Larry Lewis Jorge Silva
>
> Reinforcement Learning proposal:
>
> 2.2.1.3 Differential Calculus:
>
>
>
https://docs.google.com/document/d/175wIm8o4BNGti0WLq8U6uZORegKVjmnpfc-_E8PoGS0/edit?ts=5
fff27cd#heading=h.9ogkehmdmtel
>
> 2.2.1.4: Stats paper

>
> P2681R0
> <https://www.open-std.org/jtc1/sc22/wg21/docs/papers/2022/p2681r0.pdf>
> More
> Stats Functions Richard Dosselmann, Michael Wong
> Current github
>
> https://github.com/cplusplus/papers/issues/475
>
> https://github.com/cplusplus/papers/issues/979
>
> Stats review Richard Dosselman et al
>
> http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2021/p1708r4.pdf
>
> Feedback from Johan Lundberg and Oleksandr Korval
>
> https://isocpp.org/files/papers/D2376R0.pdf
>
> P1708R3: Math proposal for Machine Learning: 3rd review
>
> PXXXX: combinatorics: 1st Review
>
> *> std.org/jtc1/sc22/wg21/docs/papers/2020/p1708r2
> <http://std.org/jtc1/sc22/wg21/docs/papers/2020/p1708r2>*
> *> above is the stats paper that was reviewed in Prague*
> *> http://wiki.edg.com/bin/view/Wg21prague/P1708R2SG19
> <http://wiki.edg.com/bin/view/Wg21prague/P1708R2SG19>*
> *>*
> *> Review Jolanta Polish feedback.*
> *> http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html>*
>
>
> 2.2.1.4: Matrix paper
>
> 2.2.3 any other proposal for reviews?
>
> 2.3 Other Papers and proposals
>
> P1416R1: SG19 - Linear Algebra for Data Science and Machine Learning
>
>
https://docs.google.com/document/d/1IKUNiUhBgRURW-UkspK7fAAyIhfXuMxjk7xKikK4Yp8/edit#heading=h.tj9hitg7dbtr
>
> P1415: Machine Learning Layered list
>
>
https://docs.google.com/document/d/1elNFdIXWoetbxjO1OKol_Wj8fyi4Z4hogfj5tLVSj64/edit#heading=h.tj9hitg7dbtr

>
> 2.2.2 SG14 Linear Algebra progress:
> Different layers of proposal
>
> https://docs.google.com/document/d/1poXfr7mUPovJC9ZQ5SDVM_1Nb6oYAXIK_d0IjdUAtSQ/edit
>
> 2.5 Future F2F meetings:
>
> 2.6 future C++ Standard meetings:
> https://isocpp.org/std/meetings-and-participation/upcoming-meetings
>
> None
>
> 3. Any other business
>
> New reflector
>
> http://lists.isocpp.org/mailman/listinfo.cgi/sg19
>
> Old Reflector
> https://groups.google.com/a/isocpp.org/forum/#!newtopic/sg19
> <https://groups.google.com/a/isocpp.org/forum/?fromgroups=#!forum/sg14>
>
> Code and proposal Staging area
>
> 4. Review
>
> 4.1 Review and approve resolutions and issues [e.g., changes to SG's
> working draft]
>
> 4.2 Review action items (5 min)
>
> 5. Closing process
>
> 5.1 Establish next agenda
>
>
> 5.2 Future meeting
> * Jan 11, 2024 02:00 PM ET: Graph
> * Feb 8, 2024 02:00 PM ET: Graph
> * Mar 14, 2024 02:00 PM ET: Cancelled due to Tokyo 3-18-23
> * Apr 11, 2024 02:00 PM ET: Stats
> * May 9, 2024 02:00 PM ET: Graph
> * June 13, 2024 02:00 PM ET: Embedded; St.louis 6-24-29
> * July 11, 2024 02:00 PM ET: Stats
> * Aug 15, 2024 02:00 PM ET: Graph
> * Sep 12, 2024 02:00 PM ET: CPPCON Sept 15-20 so cancelled
> * Oct 10, 2024 02:00 PM ET: Stats
> * Nov 14, 2024 02:00 PM ET: Cancelled Wroclaw F2F
> * Dec 12, 2024 02:00 PM ET: Graph

Minutes for 2024/2/8 SG19 Conference Call

On Wed, Feb 7, 2024 at 11:15 AM Michael Wong <fraggamuffin_at_[hidden]>
wrote:
> Hi all, this is the SG19 Machine Learning meeting that will focus on
> graphs. Please see Phil's email update from Feb 5th on the split papers and
> the attachment. Thanks Phil.
>
> *Paper*
>
> *Status*
>
> *Pages*
>
> D9901 Overview & Introduction
>
> Active
>
> 7
>
> D9902 Algorithms
>
> Active
>
> 17
>
> D9903 Operators
>
> Active
>
> 6
>
> D9904 Views
>
> Active
>
> 11
>
> D9905 Graph Container Interface
>

> Active
>
> 10
>
> D9906 Graph Containers
>
> Active
>
> 5
>
> D9907 Adaptors
>
> Future
>
>
>
> D9908 Background and Terminology
>
> Future
>
>
> We can also have an update on stats.
> Are there any other suggested topics?
>
> We have finalized sg19 vice chairs Phil Ratzloff and Andrew Lumsdaine.
>
> We will have an SG14 meeting the day before on Wednesday to review Graph:
> Topic: SG14 monthly
> Time: 2nd Wednesdays 02:00 PM Eastern Time (US and Canada)
> Every month on the Second Wed,
>
> Join from PC, Mac, Linux, iOS or Android:
>
https://iso.zoom.us/j/93151864365?pwd=aDhOcDNWd2NWdTJuT1loeXpKbTcydz
09
> Password: 789626
> Thank you.
>
>
> Michael Wong is inviting

> you to a scheduled Zoom meeting.
>
> Topic: SG19 monthly
> Time: 02:00 PM Eastern Time (US and Canada)
> Every month on the Second Thu,
>
>
> Join from PC, Mac, Linux, iOS or Android:
>
> https://iso.zoom.us/j/93084591725?pwd=K3QxZjJIcnljaE13ZWU5cTlLNkx0Zz09
> Password: 035530
>
> Or iPhone one-tap :
> US: +13017158592,,93084591725# or +13126266799,,93084591725#
> Or Telephone:
> Dial(for higher quality, dial a number based on your current location):
> US: +1 301 715 8592 or +1 312 626 6799 or +1 346 248 7799 or +1
> 408 638 0968 or +1 646 876 9923 or +1 669 900 6833 or +1 253 215 8782
> or 877 853 5247 (Toll Free)
> Meeting ID: 930 8459 1725
> Password: 035530
> International numbers available: https://iso.zoom.us/u/agewu4X97
>
> Or Skype for Business (Lync):
> https://iso.zoom.us/skype/93084591725
>
> Agenda:
>
> 1. Opening and introductions
>
> The ISO Code of conduct:
> https://www.iso.org/files/live/sites/isoorg/files/store/en/PUB100397.pdf
>
> IEC Code of Conduct:
>
> https://www.iec.ch/basecamp/iec-code-conduct-technical-work
>
> ISO patent policy.
>
>

>
https://isotc.iso.org/livelink/livelink/fetch/2000/2122/3770791/Common_Policy.htm?nodeid=6344764&vernum=-2
>
> The WG21 Practices and Procedures and Code of Conduct:
>
> https://isocpp.org/std/standing-documents/sd-4-wg21-practices-and-procedures
>
> 1.1 Roll call of participants
>
> Michael, Phil, Andrew, Brett, Oliver, Richard, Nathan

> 1.2 Adopt agenda
>
> 1.3 Approve minutes from previous meeting, and approve publishing
> previously approved minutes to ISOCPP.org
>
> 1.4 Action items from previous meetings
>
> 2. Main issues (125 min)
>
> 2.1 General logistics
>
> Meeting plan, focus on one paper per meeting but does not preclude other paper
> updates.
>
> 2024 planning
> C++23 and C++26 status
>
>
> * Jan 11, 2024 02:00 PM ET: Graph
> * Feb 8, 2024 02:00 PM ET: Graph
> * Mar 14, 2024 02:00 PM ET: Cancelled due to Tokyo 3-18-23
> * Apr 11, 2024 02:00 PM ET: Stats
> * May 9, 2024 02:00 PM ET: Graph
> * June 13, 2024 02:00 PM ET: Embedded; St.louis 6-24-29
> * July 11, 2024 02:00 PM ET: Stats
> * Aug 15, 2024 02:00 PM ET: Graph
> * Sep 12, 2024 02:00 PM ET: CPPCON Sept 15-20 so canceled
> * Oct 10, 2024 02:00 PM ET: Stats

> * Nov 14, 2024 02:00 PM ET: Cancelled Wroclaw F2F
> * Dec 12, 2024 02:00 PM ET: Graph
>
>
> ISO meeting status
>
> future C++ Std meetings
>
> 2.2 Paper reviews
> Review BSI Graph feedback:
> As Oliver (Rosten) said "The basic premise is important, and it would be
> fantastic to have support for graphs in the standard."
>
>
The main items identified were:
> Oliver:
> - This paper is long and incomplete, it has lots of details which I think
> to be irrelevant, however things that are definitely relevant are missing
> from the paper - for example definition of graph - since people have
> different ideas. We need to add a mathematical perspective to the paper.
>
> - The structure of the paper completely changed in the new revision, so
> now it's hard to understand what and why they have done
>
> - Another missing part is discussion of graph invariants
>
> Tom (Deakin): There's a big missing part in "Prior art" part, GraphBLAS (
> https://graphblas.org) eminently.
>
> Some other things to add:
>
> 1. The electrical circuit example needs more explanation, and I think this
> will highlight some deep issues around representing things which are
> seemingly trivially graphs, as graphs in practice. In what sense is a
> bog-standard resistor directed? I assume the reason that the graph is
> directed is because current has a sign and in an undirected graph it
> becomes ambiguous which way the current is flowing (also you may want
> components like diodes). But the directed representation also has issues:
> "can current flow from 'Vdd' to 'n0'?" should be immediately answerable
> from the properties of Vdd and its edges. There are other ways to represent

> an electrical circuit. One is as a directed graph but with incident edges
> recorded - but iiuc, this is excluded from the latest version of the paper.
> Alternatively, one could have a mathematical object, the name of which I
> actually don't know: it looks like an undirected graph, but where each
> partial edge has additional, unique, end-point data, as well as the common
> weight. Things like this are the reason why I think we need a broader group
> to look at this proposal (i.e. beyond SG19) and if we possibly can we
> should involve someone from the mathematics community. Otherwise there's a
> real danger we end up missing important insights.
>
> 2. My comment about the structure of the paper changing was a reference to
> previous comparisons with boost::graph. I'm sure these were in an earlier
> version, or am I misremembering? Either way, it would be very helpful to
> have a proper discussion of e.g. the move away from visitors.
>
> 3. Re. the definition of a graph, there needs to be a proper discussion
> about whether the paper's definition of graph is what some authors call a
> multigraph and whether it does/does not include loops. These things are
> mentioned, in passing, when introducing algorithms, but terminology needs
> to be properly established.
>
> 4. I think we're trying to do too much in one go in this paper. I think a
> great first step would be to build on mdspan and try to standardize (or at
> least understand) what might reasonably be called an unstructured span.
> This could be represented as a vector of vectors or as a vector with some
> auxiliary storage indicating where the partitions fall. The point is that
> an unstructured span, with the right invariants, is an adjacency list. If
> we can understand unstructured span and its desirable api, I think this
> will be incredibly valuable guidance for what a standardized graph
> container might look like.
>
> 5. IIUC, this paper excludes pure connectivity graphs. These are
> incredibly helpful and, if I've understood correctly that they are not
> supported, would be a major omission. Another good reason, imo, to start
> with unstructured span!
>
> 6. I'm not convinced by the load api. We don't have a load api for vector
> etc. Moreover, would it not be preferable to have appropriate constructors?
>
>

P1709 original unified paper now split
D9901r0
split 8 papers
Oliver: subtleties need discussion?
have a readers guide and maybe order it

Algorithms (9902):
allocator polled as Andrew is uncomfortable with that
 because many algo dont need additional allocators
removed exception throwing, just use preconditions
Self loop? OK to add
bFS seems only for shortest path and not general: based onnetwork BFS Boots
graph has parameterized, and every line can be parameter but we dont want
that, trying to avoid the complications from BGL by using adaptors
BFS for shortest path is an implementation when a more flexible version
under the views, and I would argue this does not have the flexibility of
BGL, can you first time reach the node and can access the late function
object? No right now just the ealy in the node, but can get both; Ok but
what about the edge? No. Ok that could be a gap.

pre and post processing can be added after initial BFS?

comparison on with BGL with this

Operators (9903):
create one graph from another
complexity needs to filled in
pre/post condition: std vector of the right things , this should be able to
operate on all these kinds, should these be captured in operators and
maintain graph invariants? some are captured in concepts, RT vs compile
time requirements
range of range or span of span and say this is the adjacency list for my
graph and it might point beyond the end, can that be captured by a RT
property like concept, so may be better as a precondition? u r right, some

ops mutate rather then create new graphs, its something we have not discussed a lot. Yes this part id the least mature of everything in how we can reflect that to these operators so why we might not release this paper. yes we might want a const ref graph

Views (9004)
not changed in anyway

Containers (9905)
overload the target ID fn the CPO and to accomplish that they would overlid in STD namespace
becaause it has to be in the namesapce as the data structure is not going to fly
5.3 adjacency list concept dealing with vertex IDs and not vertex references, this is what the views are going to need
Oliver: make smallest leap from STL, MD span is a graph, compose STL
This comes from graphs embedded in products and be able to use and adapt and different types of graphs, like sparse ID storing in a map, concept is general enough to allow the algo to operatie on as many graphs as possible that are found in the wild
dynamic graph is an important

graph containers (9906)
has compressed graph identify characteristics: can it store wrights in edges and nodes? yes. Like CSR . Need an allocator if its holding edge weights? edge value type is the weights
, the type is irrelevant as we have to use it for everything. Can you have separate allocators for edges and nodes. Its just rebinded for edges and nodes
Editorial note: on diffrerenign pt of view of what cgoes in a graph - CSR, restores vertex depends on if VV is void; want to build compound data structure; people might not want to compose it themselves
what about limitation of uint_32, right need to be size_t
missing an explicit
what is the rest of the interface for compressed graph? it uses the rest of the graph container interface, it is part of the api

terminology (9908)
STL style, but some datastructure that also dont fit into that like mdspan,

regularity in the inner dimension, and this is what graphs look like
presents many different design space for alternatives design
Terminology
fuzzy naming and sloppy definitions
this is precise unambiguous language on graph
abstract notion, vs representation of a graph

Oliver: multiple edges, self loops, dynamic undirected graph wanats to
delete edges and maintain invariants, these come with a cost
directedness is a runtime property, should there be a type so its compile
type? What is the use? BFS goes to its neighbor, does not matter if its
edge was directed or undirected. If I want a DAg acyclic graph, a type that
is directed and invariant. vertex i and j, if directed they are adjacent to
each other, then its a vector of vector for entry i and j, cant enforce
that at compile time. Can each edge be pair?

electrical circuit example need to be replaced? highlights some of the
really deep subtlety of graph, merit a discussion, yes true

Range paper proposal
sparse representation of adjacency matrix in both coordinate and compressed
form
a vector of list should be a graph, can do BFS
library is algo centrioc, so concept is only used to bound polymorphism of
an algo
this makes algo operate on representation,
algo use concepts to contrain the algo,
can we use the range syntax

Main poll to exit graph from SG19 to LEWG
1/2/0/0/2 Not passed
irregular spans, containers
important needs to be addressed

>
> Here's a link to the paper (different than the previous paper reviewed).
> There are some additional updates I'm planning on making before the
> meeting.
>
>
>
https://docs.google.com/document/d/1OpH-xxRri7tJTtJJIZTYmSHkkrZJkdBwm9zJ7LqolfQ/edit?usp=sharing
>
>
>
>
> P1709R3:
>
>
https://docs.google.com/document/d/1kLHhbSTX7j0tPeTYECQFSNx3R35Mu3xO5_dyYdRy4dM/edit?usp=sharing
>
>
>
https://docs.google.com/document/d/1QkfDzGyfNQKs86y053M0YHOLP6frzhTJqzg1Ug_vkkE/edit?usp=sharing
>
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html>
>
> <
>
>
https://docs.google.com/document/d/175wIm8o4BNGti0WLq8U6uZORegKVjmnpfc-_E8PoGS0/edit?ts=5fff27cd#heading=h.9ogkehmdmtel
> *>*
>
> Array copy semantics:
> array copy-semantics paper P1997 "Relaxing Restrictions on Arrays",
> https://wg21.link/p1997
>
> Stats feedback:
>
> P2376R0

> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2021/p2376r0.pdf>
> Comments
> on Simple Statistical Functions (p1708r4): Contracts, Exceptions and
> Special cases Johan Lundberg
>
> 2.2.1.2 Reinforcement Learning Larry Lewis Jorge Silva
>
> Reinforcement Learning proposal:
>
> 2.2.1.3 Differential Calculus:
>
>
>
https://docs.google.com/document/d/175wIm8o4BNGti0WLq8U6uZORegKVjmnpf
c-_E8PoGS0/edit?ts=5fff27cd#heading=h.9ogkehmdmtel
>
> 2.2.1.4: Stats paper
>
> P2681R0
> <https://www.open-std.org/jtc1/sc22/wg21/docs/papers/2022/p2681r0.pdf>
> More
> Stats Functions Richard Dosselmann, Michael Wong
> Current github
>
> https://github.com/cplusplus/papers/issues/475
>
> https://github.com/cplusplus/papers/issues/979
>
> Stats review Richard Dosselman et al
>
> http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2021/p1708r4.pdf
>
> Feedback from Johan Lundberg and Oleksandr Korval
>
> https://isocpp.org/files/papers/D2376R0.pdf
>
> P1708R3: Math proposal for Machine Learning: 3rd review
>
> PXXXX: combinatorics: 1st Review
>

> *> std.org/jtc1/sc22/wg21/docs/papers/2020/p1708r2
> <http://std.org/jtc1/sc22/wg21/docs/papers/2020/p1708r2>*
> *> above is the stats paper that was reviewed in Prague*
> *> http://wiki.edg.com/bin/view/Wg21prague/P1708R2SG19
> <http://wiki.edg.com/bin/view/Wg21prague/P1708R2SG19>*
> *>*
> *> Review Jolanta Polish feedback.*
> *> http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html
> <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2020/p2119r0.html>*
>
>
> 2.2.1.4: Matrix paper
>
> 2.2.3 any other proposal for reviews?
>
> 2.3 Other Papers and proposals
>
> P1416R1: SG19 - Linear Algebra for Data Science and Machine Learning
>
>
https://docs.google.com/document/d/1IKUNiUhBgRURW-UkspK7fAAyIhfXuMxjk7xKikK4Yp8/edit#heading=h.tj9hitg7dbtr
>
> P1415: Machine Learning Layered list
>
>
https://docs.google.com/document/d/1eINFdIXWoetbxjO1OKol_Wj8fyi4Z4hogfj5tLVSj64/edit#heading=h.tj9hitg7dbtr
>
> 2.2.2 SG14 Linear Algebra progress:
> Different layers of proposal
>
>
https://docs.google.com/document/d/1poXfr7mUPovJC9ZQ5SDVM_1Nb6oYAXlK_d0IjdUAtSQ/edit
>
> 2.5 Future F2F meetings:
>
> 2.6 future C++ Standard meetings:
> https://isocpp.org/std/meetings-and-participation/upcoming-meetings

>
> None
>
> 3. Any other business
>
> New reflector
>
> http://lists.isocpp.org/mailman/listinfo.cgi/sg19
>
> Old Reflector
> https://groups.google.com/a/isocpp.org/forum/#!newtopic/sg19
> <https://groups.google.com/a/isocpp.org/forum/?fromgroups=#!forum/sg14>
>
> Code and proposal Staging area
>
> 4. Review
>
> 4.1 Review and approve resolutions and issues [e.g., changes to SG's
> working draft]
>
> 4.2 Review action items (5 min)
>
> 5. Closing process
>
> 5.1 Establish next agenda
>
>
> 5.2 Future meeting
> * Jan 11, 2024 02:00 PM ET: Graph DONE
> * Feb 8, 2024 02:00 PM ET: Graph
> * Mar 14, 2024 02:00 PM ET: Cancelled due to Tokyo 3-18-23
> * Apr 11, 2024 02:00 PM ET: Stats
> * May 9, 2024 02:00 PM ET: Graph
> * June 13, 2024 02:00 PM ET: Embedded; St.louis 6-24-29
> * July 11, 2024 02:00 PM ET: Stats
> * Aug 15, 2024 02:00 PM ET: Graph
> * Sep 12, 2024 02:00 PM ET: CPPCON Sept 15-20 so cancelled
> * Oct 10, 2024 02:00 PM ET: Stats
> * Nov 14, 2024 02:00 PM ET: Cancelled Wroclaw F2F
> * Dec 12, 2024 02:00 PM ET: Graph