# C++ Club UK

Gleb Dolgich

2019-02-28

**Bryce Lelbach**
@blelbach

Following

What proposed C++20 feature do you think is most important?

23% Concepts

40% Modules

16% Ranges

21% Coroutines

1,178 votes • Final results

5:02 PM - 13 Feb 2019

21 Retweets 18 Likes
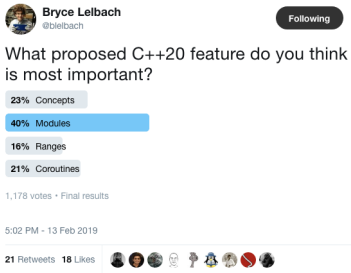
https://www.reddit.com/r/cpp/comments/aq8rxf/bryce_lelbach_on_twitter_what_proposed_c20/

# Twitter



**EWG Chair**
@EWGchair

Nice standard you have there, it would be a real shame if anything happened to it...

| **13** Likes | **1** Retweet |
| --- | --- |
| 18 Feb 2019 at 19:35 | via **Twitter for iPhone** |

# Kona Trip Reports

- Modules! Coroutines! Contracts! Ranges! Constexpr! Spaceships! Calendars! Time zones!
- Bryce Lelbach https://www.reddit.com/r/cpp/comments/au0c4x/201902_kona_iso_c_committee_trip_report_c20/
    - The reports of modules being dead on arrival have been greatly exaggerated. https://www.reddit.com/r/cpp/comments/au0c4x/201902_kona_iso_c_committee_trip_report_c20/eh4stfg/
    - More Reddit https://www.reddit.com/r/programming/comments/au1ie3/201902_kona_iso_c_committee_trip_report_c20/
    - Hacker News https://news.ycombinator.com/item?id=19235702
- Herb Sutter https://herbsutter.com/2019/02/23/trip-report-winter-iso-c-standards-meeting-kona/
    - Reddit https://www.reddit.com/r/cpp/comments/au1skl/herbs_kona_trip_report_winter_iso_c_standards/
      At our next meeting in July, we expect to formally adopt a few additional features that were design-approved at this meeting but didn't complete full wording specification review this week <...> (formatting, flat_map etc.)

Bryce Lelbach:

*TL;DR: C++20 may well be as big a release as C++11.*
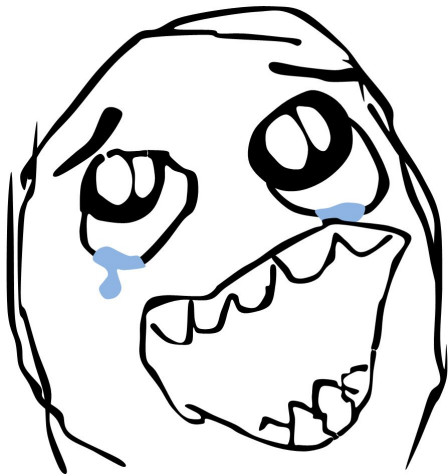
# API design: iterator-based vs. container-based

Pros:

- ► Iterator-based API is more flexible
    - ► works with different containers
    - ► works with types that expose differently-named **begin** and **end** member functions
    - ► works on ranges of data instead of the entire container
- ► Protobuf APIs are iterator-based
- ► Flatbuffers are iterator-based
- ► Standard and Boost algorithms are iterator-based

Cons:

- ► Not as readable as container-based API

# Eigen + MTS + Google Ceres Solver = SEGFAULT

- Eigen http://eigen.tuxfamily.org/index.php?title=Main_Page
- Ceres Solver http://ceres-solver.org/
- Memory managers:
    - MTS http://www.desy.de/user/projects/C++/products/mts.html
        - for every 10000 allocations 13 to 17 were unaligned
    - TCMalloc (Google) http://goog-perftools.sourceforge.net/doc/tcmalloc.html
    - Jemalloc (FreeBSD, Facebook) http://jemalloc.net/

```
1 bool aligned = ((unsigned long)p % 16) == 0;
```

# Cpp On Sea 2019 Trip Report by Arne Mertz

*C++ On Sea is definitely the best conference I have ever been to.*

*The opener was titled "Hello, World", there was a "main()" plenary hall and session rooms titled "const west", "east const", and "unsigned". The latter was the smallest of the session rooms and had an overflow problem a few times, but luckily that did not lead to undefined behavior, because C++ conference attendees seem to be very nice people in general.*

# C++ Binary Compatibility and Pain-Free Upgrades to Visual Studio 2019

https://blogs.msdn.microsoft.com/vcblog/2019/01/31/cpp-binary-compatibility-and-pain-free-upgrades-to-visual-studio-2019/

# Visual Studio 2019 Release Candidate (RC) now available

https://devblogs.microsoft.com/visualstudio/visual-studio-2019-release-candidate-rc-now-available/

# Twitter

**hdevalence @ RustConf**
@hdevalence

↑ 3 Quotes

OH: "Look at these cats. Look how happy they are. They probably don't know what CMake is. We could all learn from their example."

17/08/2018, 05:01 (Friday)
Twitter for iPhone

Retweeted by @volatile_void
19/08/2018, 16:02

129 Likes          41 Retweets          Thread ›

# Twitter

**amye**
@amye

↑ 2 Replies, 91 Quotes

OH: "You can't fix that bug. That's a load-bearing bug."

19/02/2019, 02:36 (Tuesday)
Twitter Web Client

Retweeted by @bilm
20/02/2019, 23:04

| 5228 Likes | 1309 Retweets | Thread ❯ |
|------------|---------------|----------|