C++ Club Meeting Notes

Gleb Dolgich

2018-03-29

Jacksonville C++ Standards Meeting

- Trip reports
  - Vittorio Romeo
  - Guy Davidson
  - using std::cpp
  - CppCast with Patrice Roy
  - Botond Ballo, Reddit thread
- The Plan (according to Herb Sutter – Thanks Bjarne):
  - Executors: TS in the C++20 timeframe, standard in C++23
  - Networking: C++23 (delayed by Executors)
  - Coroutines: C++20
  - Modules: Partially in C++20 with more in C++23 (blame Google)
  - Contracts: C++20
  - Reflection: TS in C++20 timeframe, standard in C++23
  - Ranges: Core in C++20, cool stuff in C++23

- Post
- P0424R0: Reconsidering literal operator templates for strings

```
s = format("{}"_fmt, 42);
```

- P0732R0: Class Types in Non-Type Template Parameters

```
s = format<"{}">(42);
s = format(fmt<"{}">, 42);
s = "{}"_format(42);
```

C++ committee, please look at the big picture

Reddit

> *More disappointment as the C++ standardisation process fails to deliver what programmers need....*
>
> *There have been a lot of improvements but the whole "It must be perfect" ethos rather than 80/20 is really damaging this language and libraries.*
>
> *Maybe the precious backwards compatibility is more damaging to progress?*

Terse Concepts syntax poll

Doodle poll

Which concepts terse syntax looks promising and should be adopted by the committee?

- Adjective syntax – 28 (Peter Sommerlad, Jonathan Müller)
- Original syntax – 18 (Yours truly)
- \{\} syntax – 8
- No terse syntax is needed – 3 (Bryce Lelbach)
- `auto<>` syntax – 2 (Joel Falcou)

28.03.2018, Distributed C++ meet-up 0x02 - Berlin && London && Stockholm

- Blog post
- Reddit announcement
- C++ London
- SwedenCpp
- Berlin C++

The RedMonk Programming Language Rankings: January 2018

Rankings

1. JavaScript
2. Java
3. Python
4. PHP
5. C#
6. C++

GCC8 usability improvements by Red Hat

- Article

Web

StackOverflow

#pragma once has unfixable bugs. It should never be used.

StackOverflow

If your `#include` search path is sufficiently complicated, the compiler may be unable to tell the difference between two headers with the same basename (e.g. `a/foo.h` and `b/foo.h`), so a `#pragma once` in one of them will suppress both. It may also be unable to tell that two different relative includes (e.g. `#include "foo.h"` and `#include "../a/foo.h"`) refer to the same file, so `#pragma once` will fail to suppress a redundant include when it should have.

#pragma once has unfixable bugs. It should never be used (cont.)

The short version of why this is unfixable is that neither the Unix nor the Windows filesystem API offer any mechanism that guarantees to tell you whether two absolute pathnames refer to the same file.

Historical note: The only reason I didn't rip `#pragma once` and `#import` out of GCC when I had the authority to do so, ~12 years ago, was Apple's system headers relying on them. In retrospect, that shouldn't have stopped me.

Reddit Q: Monadic error handling

Reddit

Post

```
1 g++ -O2 -D_GLIBCXX_DEBUG example.cpp
```

Memory Tagging (aka memory coloring, memory tainting, lock and key) and how it improves C/C++ memory safety

PDF

- Every TG (tagging granularity) bytes of memory aligned by TG are associated with a tag of TS (tag size) bits. These TG bytes are called the granule.
- TS bits in the upper part of every pointer contain a tag.
- Memory allocation (e.g. malloc) chooses a tag, associates the memory chunk being allocated with this tag, and sets this tag in the returned pointer.
- Every load and store instruction raises an exception on mismatch between the pointer and memory tags.
- The memory access and tag check do not necessarily occur atomically with respect to each other.

Memory Tagging (cont.)

Implementations:

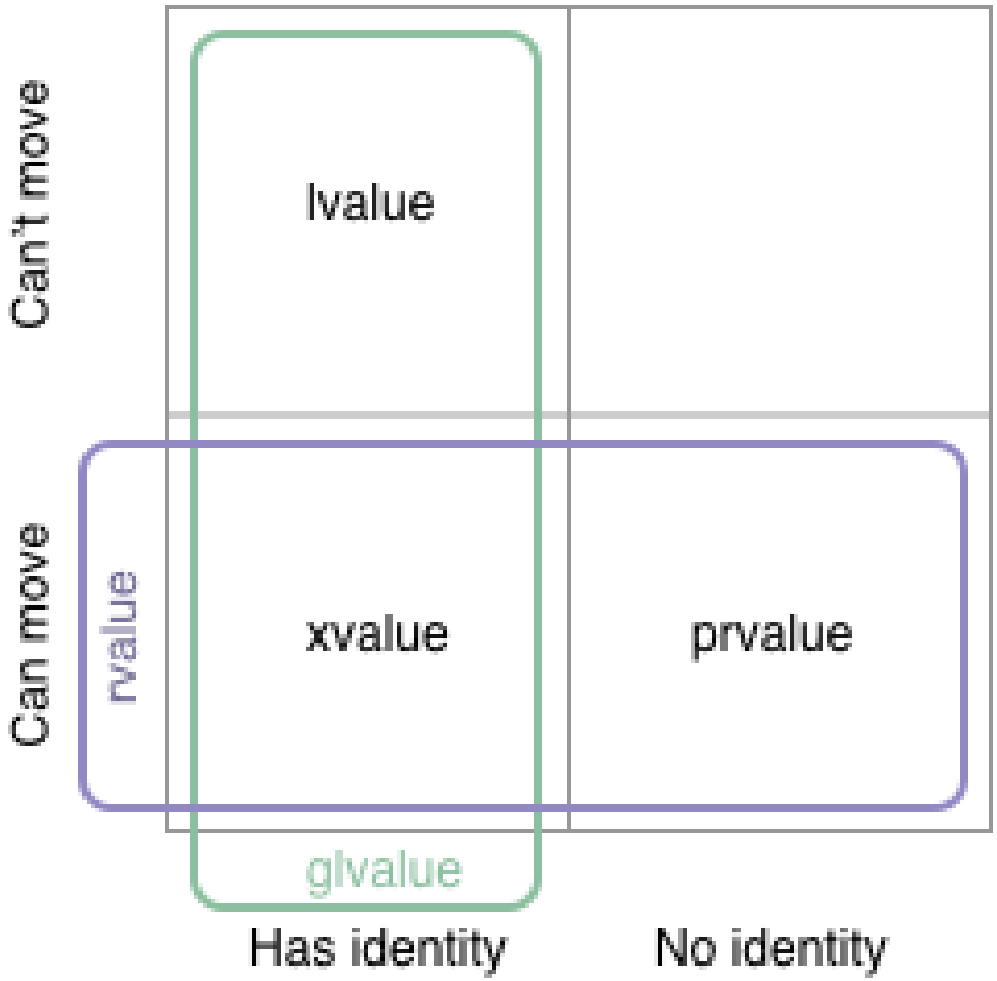
- SPARC ADI: The SPARC ADI hardware extension is supported on SPARC M7/M8 CPUs running Solaris OS. There is also some indication that Linux support is in progress.
- AArch64 HWASAN (hardware-assisted ASAN): an AArch64-specific compiler-based tool.

Post

- Structured bindings
- Fold expressions
- `constexpr if`
- Selection statements with initializer
- Nested namespaces

lvalues, rvalues, glvalues, prvalues, xvalues, help!

Post



Floating point visually explained

- Article

Twitter: C++ 9-year cycle

Andrew Pardoe
@apardoe

@herbsutter has just characterized C++ development in terms of 9-year cycles. C++14/17/20 is such a cycle. It's an interesting and useful observation. I'm looking forward to C++29!

17/03/2018, 15:14 (Today)
Twitter for iPhone

8 Likes | 2 Retweets | Thread >

Twitter: Fast integers – Godbolt, StackOverflow

JF Bastien
@jfbastien

1 Reply

C++ pro-tip on making integers fast:

int_fast32_t operator"" _ᕕᐛᕗ(unsigned long long);
auto 🏎 = 42_ᕕᐛᕗ; // No furious!

Try it! → godbolt.org/g/BXptXc

13/03/2018, 18:29 (Yesterday)
Twitter Web Client

17 Likes | 2 Retweets | Thread >

Quote

Oscar Godson:

> *One of the best programming skills you can have is knowing when to walk away for a while.*