RUN-TIME FOLD EXPRESSIONS

Just an idea...

Core C++ 2023

06th June 2023

Vittorio Romeo

mail@vittorioromeo.com

@supahvee1234

Fold expressions

- Fold a parameter pack into a single final result
  - Using a given binary operator
- Compile-time operation
- Declarative, concise, efficient
- No mutable state

```
std::string concat(const auto&... xs)
{
    std::ostringstream oss;
    (oss << ... << xs);
    return oss.str();
}
```

Run-time...?

- Fold expressions can not be used on run-time containers
  - What is the next best thing?

```
std::string concat(auto iterator b, auto iterator e)
{
    std::ostringstream oss;

    for (; b != e; ++b)
    {
        oss << *b;
    }

    return oss.str();
}
```

- Not concise or declarative anymore

Run-time...?

- Fold expressions can not be used on run-time containers
  - What is the next best thing?

```
std::string concat(auto range r)
{
    std::ostringstream oss;
    std::ranges::for_each(r, [&](const auto& x){ oss << x; });
    return oss.str();
}
```

- More declarative, but...
  - Still not concise
  - Function call indirection
  - Worse build time and debug performance
  - More complex

The idea

- Let’s allow fold expressions at run-time
  - Syntactic sugar for loops

```
std::string concat(auto range r)
{
    std::ostringstream oss;
    (oss << ... << r[:]);
    return oss.str();
}
```

```
std::string concat(auto range r)
{
    std::ostringstream oss;
    (oss << ... << for r);  // alternative syntax
    return oss.str();
}
```

```
std::string concat(auto range r)
{
    std::ostringstream oss;
    (oss << ... << r[:]);
    return oss.str();
}
```

```
std::string concat(auto range r)
{
    std::ostringstream oss;

    {
        auto it = begin(r);
        decltype(auto) result = oss << *it;
        for (; it != end(r); ++it) result = result << *it;
    }

    return oss.str();
}
```

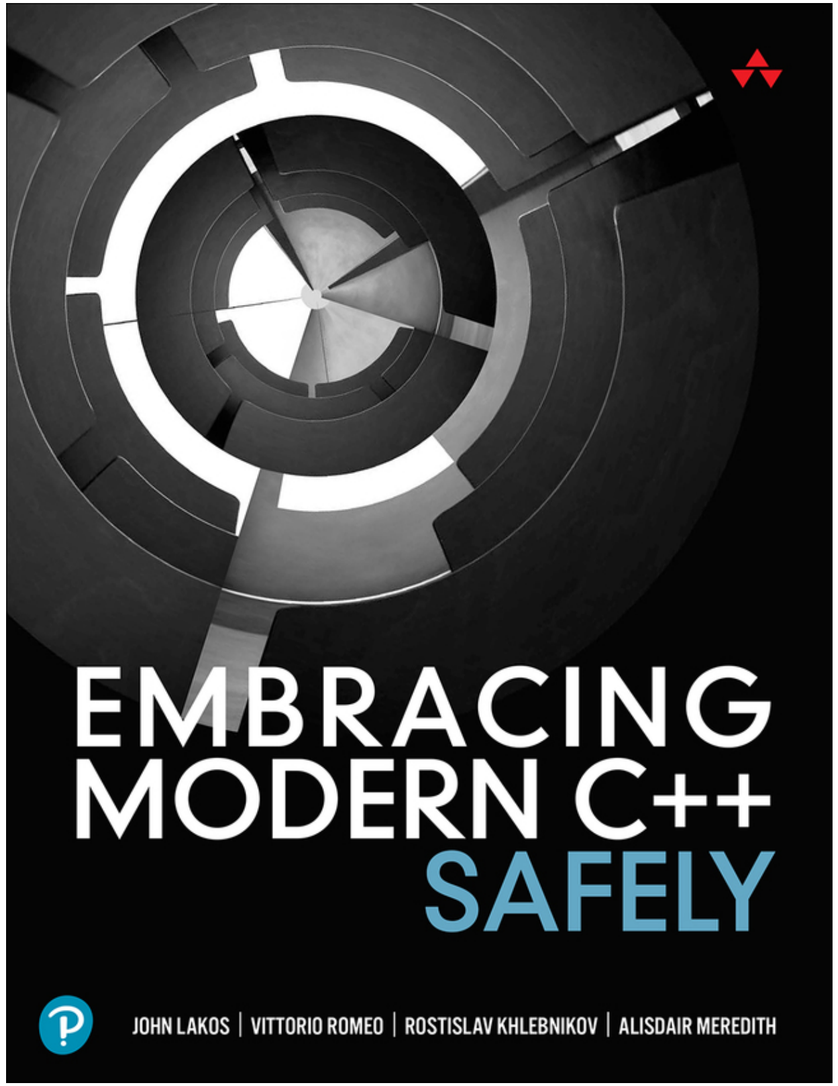
The idea

- Tweet #0
  - https://pbs.twimg.com/media/Fqa0nWvXsAA3zSv?format=jpg&name=large
- Tweet #1
  - https://pbs.twimg.com/media/Fqa1vgOXsAY0vtl?format=jpg&name=large
- Interactive demo
  - https://vittorioromeo.info/Misc/Linked/desugar_runtime_fold.html

- Interested in writing a paper? Let me know!
  - mail@vittorioromeo.com
  - @supahvee1234

Thank you!

- Interested in compilation times?
  - **“Improving Compilation Times: Tools & Techniques”**
  - Hall 3 – 12:00-13:00
- Detailed analysis of Modern C++ features: **EMC++S**!
  - 📕 https://emcpps.com
  - No opinions: just *facts*, *use cases*, *pitfalls*, and *annoyances*
- SFML 3.x – *“Simple and Fast Multimedia Library”*
  - https://sfml-dev.org/
- Open Hexagon – open-source arcade game made with SFML
  - https://openhexagon.org/
- Let’s keep in touch!
  - mail@vittorioromeo.com



- Thanks!
  - 🙋 Questions? 💁 Comments? 🙅 Criticism? 🙇 Stories?
  - https://vittorioromeo.com | @supahvee1234 | https://github.com/vittorioromeo | mail@vittorioromeo.com