
CppCon 2018
Simon Zeni
OPAL-RT
TECHNOLOGIES

Samedi - Multithreading workshop part 1

By Rainer Grimm - www.modernescpp.com

- Introduction to multithreading
  - Threads & locks
  - Data races & dead locks
  - Condition variables & call once flags
  - Thread local & static variables
  - Futures, async & packaged_task

Dimanche - Multithreading workshop part 2

- Memory model
  - Atomics, atomic_flag & atomic bool
  - User define atomics
  - Spinlocks
  - Memory order (CppMem)
- Parallel STL
  - Execution policy
- C++20
  - Executors
  - Coroutines
  - Latches & barriers

Lundi - part 1

- Concepts: The future of generic programming - Bjarne Stroustrup
  - Little tour of the future of C++, mostly with concepts
- Enough string_view to Hang Ourselves - Victor Ciura
  - The pros and cons of std::string_view
- Trainer Panel I - Michael Caisse • Stephen Dewhurst • Nicolai Josuttis • Scott Meyers
- Modern C++ Design (part 1 of 2) - Titus Winters
  - Designing an API in modern C++

Lundi - part 2

- An Allocator is a Handle to a Heap - Arthur O'Dwyer
  - Introduction to the concept of allocators, and std::pmr
- Surprise in Object Lifetime - Jason Turner
  - Tips and gotchas on when ctors and dtors are called
- Grill the Comitee - Jon Kalb, Marshall Clow , Olivier Giroux, Howard Hinnant, Bjarne Stroustrup, Herb Sutter, Ville Voutilainen
  - Interactive panel with members of the comitee

Mardi - part 1

- Minidumps: gdb-compatible core dumps - Matthew Fleming
  - An explanation of the ore dumps, and how Backtrace made smaller ones
- Compile Time Regular Expression - Hana Dusíková
  - Regex from compile time strings, my favorite talk of the con
- Patterns and Techniques Used in the Houdini 3D Graphics Application - Mark Elendt
  - Transition to C++ applied to a big enterprise
- Make World : The Most Miserable Place in C++ - Jason Turner • Peter Bindels • Robert Maynard • Isabella Muerte • Jussi Pakkanen
  - Build system panel
- The Exciting New Future of Self Reclamation for High Performance - Paul McKenney • Maged Michael • Michael Wong
  - Technical talk about concurrency in C++20

Mardi - part 2

- What Could Possibly Go Wrong? A Tale of Expectations and Exceptions - Simon Brand • Phil Nash
  - Introduction to std::expected and the “herbceptions” (P0709)
- Touring the “C++ Tip of the Week” Series - Jon Cohen • Matt Kulukundis
  - Theatrical reenactment of a few Abseil tips of the week
- Talking to Typelists - Stephen Dewhurst
  - Fun with typelists, for fun
- The Networking TS in Practice: Testable, Composable Asynchronous IO in C++ - Robert Leahey
  - Tour of the C++20 networking features in parallel with Boost.ASIO
- Lightning Talks - Michael Caisse

Mercredi - part 1

- State Machine Battlefield. Native vs STL vs Boost - Kris Jusiak
  - Benchmarking between naive, std::variant, Boost.MSM and Boost.SML
- Simplicity : not just for beginners - Kate Gregory
  - Focus on simple code for better readability
- Lesser known Linux Kernel APIs - Hannes Sowa
  - A quick tour of recent C++ compatible kernel APIs
- Compile-time programming and reflection in C++20 and beyond - Louis Dionne
  - An explanation of the future improvements of constexpr in C++20

Mercredi - part 2

- Refactoring Legacy Codebases with LibTooling - James Bennett
  - Applying all the tools provided by Clang to a real codebase
- Value Semantics: Fast, Safe and Correct by Default - Nicole Mazzuca
  - lvalue, rvalue, and a bit of spaceship operator
- Modern C++ in Embedded Systems - The Saga Continues - Michael Caisse
  - Why and how to use C++ in a C world
- Lightning Talks - Michael Caisse

Jeudi

- OOP is Dead, Long Live Data-oriented Design - Stoyan Nikolov
  - Only watched the last few minutes, sadly
- Thoughts on a More Powerful and Simpler C++ - Herb Sutter
  - Introduction of the famous ‘-Wlifetime’ paper
- Design for Performance: Practical Experience - Fedor Pikus
  - Designing your code with SIMD in mind
- Effective replacement of dynamic polymorphism with std::variant - Mateusz Pusz
  - State machine, again, but more detailed
- Liberating the Debugging Experience with GDB Python API - Jeff Trull
  - More pleasant GDB backtraces from C++
- Compute More in Less Time Using C++ SIMD Wrapper Libraries - Jefferson Amstutz
  - Introduction to SIMD and how to used them easily
- Lightning Talks - Michael Caisse

Vendredi

- Applied Best Practices - Jason Turner
  - Condensed and enhanced C++ Weekly
- The Bits Between the Bits : How We Get to main() - Matt Godbolt
  - How Linux ELF files load and run
- Easy to Use, Hard to Misuse : Declarative Style in C++ - Ben Deane
  - Guidelines for clean and easy to read interfaces
- Spectre : Secrets, Side-Channels, Sandboxes, and Security - Chandler Carruth
  - The Spectre & Meltdown in detail, from the compiler side
- Closing Panel : Spectre - Matt Godbolt • Chandler Carruth • Jon Masters • Matt Miller

Samedi - Accelerated TDD : For More Productive C++

By Phil Nash - https://levelofindirection.com/

- What is TDD
- SOLID concepts
- Good designs for TDD
- Bad designs for TDD
- Practical examples
- “Legacy code is code without tests” - Michael Feathers

Links

- Slides availables on GitHub
  - https://github.com/CppCon/CppCon2018
- Videos of the talks on Youtube
  - https://www.youtube.com/user/CppCon/videos