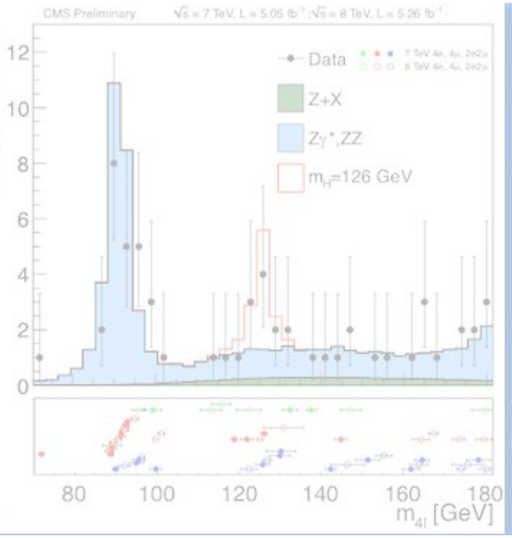


Compiler Research

Status And Plans

Vassil Vassilev

06.06.2024

People

Pavlo Svirin

GSoC24, Kyiv University, UA

ROOT superbuilds.

Info

Isaac M. Santana

GSoC24, University of Granada, ES

Improving performance of BioDynaMo using ROOT C++ Modules.

Info

Chaitanya Shahare

GSoC24 National Institute of Technology Srinagar, India

LLVM.org Website Redesign

Info

Riya Bisht

GSoC24, Graphic Era University, India

Enable CUDA compilation on Cppyy-Numba generated IR.

Info

People

Atell Yehor Krasnopolski

GSoC24, University of Wuerzburg, DE

Implement Differentiating of the Kokkos Framework in Clad

Info

Khushiyant

GSoC24, G.G.S.I.P.U, India

STL/Eigen - Automatic conversion and plugins for Python based ML-backends.

Info

Tharun Anandh

GSoC24, National Institute of Technology, Tiruchirapalli, India

Integrate a Large Language Model with the xeus-cpp Jupyter kernel

Info

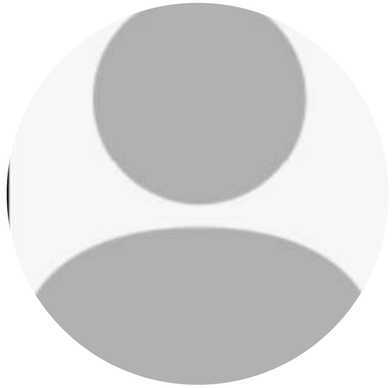
Mihail Mihov

GSoC24, Stara Zagora Math High School, BG

Add support for consteval and constexpr functions in clad

Info

People



Thomas Fransham

GSoC24, UK

Support clang plugins on Windows

Info

Matthew Barton

Open Source Contributor

Continuous Integration, CppInterOp, Xeus-Cpp.

Info

Sahil Patidar

GSoC24, Vindhya Institute of Technology, India

Out-Of-Process execution for Clang-Repl

Info

Clad — Enabling Differentiable Programming in Science

Source Transformation AD With Clad

- Released Clad v1.5
  - Added support for new, delete, alloc, free
  - Reduced size of the generated gradient code
  - Simplified adjoints used for fp error estimation
  - Delayed differentiation until the end of translation unit
  - Removed array_ref dependency in the generated code
- Scientific use-cases
  - RooFit's Clad-based ATLAS Higgs combination benchmark works and scales well. Looking into the CMS combine Higgs analysis with CMS open data. To be presented at ICHEP.
  - Progress on supporting simulation code with HepEmShow.
- Next milestone v1.6 is planned in the end of the month

C++ as a service — rapid software development and dynamic interoperability with Python and beyond

Hands on details can be seen in our showcase presentation.

Status. Cling

- Being upgraded to llvm18.

Status. Clang-Repl

- 7 merged contributions last two months: link
- 2 contributions updated last two months: link
- Making slow progress on:
  - PR84769 — [clang-repl] Implement Value pretty printing for containers. Value Handling (RFC)
  - PR86402 — [clang-repl] Support wasm execution

The goal is to provide better stability and robustness which can later cling can reuse.

Status. CppInterOp

- Released v1.3
  - Added code completion support
  - Better packaging
  - Better CI integration
  - Initial Wasm Support
- CppInterOp.jl Julia package

Status. Xeus-Cpp

- CppInterOp v1.3.0 was integrated in xeus-cpp
- Working on merging more infrastructure xeus-clang-repl into xeus-cpp
- Released v0.5.0
- Releasing a major release to deprecate xeus-cling requires 1 feature to be implemented wrt automatically loading of symbols.

Status. Xeus-Clang-Repl

- No updates

Open Projects

- Open projects are tracked in our open projects page.

Next Meetings

- Monthly Meeting — 11th July, 1700 CET/0800 PDT

If you want to share your knowledge/experience with interactive C++ we can include presentations at an upcoming next meeting

Thank you!

Lingo

- **CppInterOp** is a product of OAC-1931408 and exposes API from Clang and LLVM in a mostly backward compatibe way. The API support downstream tools that utilize interactive C++ by using the compiler as a service. That is, embed Clang and LLVM as a libraries in their codebases. The API are designed to be minimalistic and aid non-trivial tasks such as language interoperability on the fly. In such scenarios CppInterOp can be used to provide the necessary introspection information to the other side helping the language cross talk. The package makes it easy to deploy as it ships Clang as a service without any dependencies.
- **Xeus-Clang-Repl** is a product of OAC-1931408 that is a Jupyter plugin supporting C++ development based on ClangRepl.
- **Xeus-Cpp** is a product of OAC-1931408 in collaboration with the QuantStack company. It is a Jupyter kernel for C++ based on the native implementation of the Jupyter protocol xeus. It is supports the Wasm version of Jupyter – JupyterLite. Generalization of Xeus-Clang-Repl.

Lingo

- **Cling** The first C++11-compliant interpreter used in the field of High-Energy Physics for data analysis and interoperability.
- **ClangRepl** is a generalization of Cling in LLVM/Clang upstream and is a product of OAC- 1931408. It be more reliable, easier to deploy. It follows the best practices adopted by the LLVM developers community. It supports CUDA, OpenMP and Wasm.
- **Cppyy** is an undervalued, cutting-edge Python/C++ language interoperability tool originated by Wim Lavrijsen, LBL. It is the de-facto standard for efficient Python/C++ interoperability in the field of particle physics. As part of OAC-1931408 our group collaborated with LBL improve packaging and reduce the dependencies allowing cppyy to move closer to LLVM orbit.