



Compiler Research

Status And Plans

Vassil Vassilev

11.01.2024

Clad — Enabling Differentiable Programming in Science

Source Transformation AD With Clad

- ❖ Establishing collaborations with stakeholders
- ❖ Discussing milestones and deliverables

Source Transformation AD With Clad

- ❖ Implemented initial version of To-Be-Recorded (TBR) analysis
- ❖ Refactored improved the generated derivative code (on average 1 / 3 shorter code for gradients)
- ❖ Upgraded the embedded Enzyme version from 0.0.36 to 0.0.95
- ❖ Improved CMake options spelling
- ❖ Added support for clang17
- ❖ Added basic support for pointers
- ❖ Added unit tests support allowing to write gtests with small programs
- ❖ Improved the CI stability

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

- ❖ Released Cling v1.0 based on llvm13
- ❖ Upgraded to llvm16

Status. Clang-Repl

- ❖ Implemented [ClangRepl] Reland Semantic Code Completion (#75556)
- ❖ Implemented D154382 [ClangRepl] support code completion at a REPL
- ❖ Value Handling (RFC)
 - ❖ D146809 — [clang-repl] Implement Value pretty printing for containers

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

Status. Xeus-Clang-Repl

- ❖ Implemented OSX Support
- ❖ Released v0.2.0

Status. *Xeus-Cpp*

- ❖ Implemented Windows, WASM Support
- ❖ Released v0.3.0

Status. CppInterOp

- ❖ Released v1.0 and v1.1
- ❖ Accepted as a package in conda forge
- ❖ Enabled OSX builds

Open Projects

- ❖ Open projects are tracked in our [open projects page](#).

Next Meetings

- ❖ Monthly Meeting — 1st Feb, 1700 CET / 0800 PDT
 - ❖ TBD

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 compatible 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 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.
- ❖ **Cppy** 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 cppy to move closer to LLVM orbit.