



# Compiler Research

## Status And Plans

Vassil Vassilev

07.03.2024

# Clad — Enabling Differentiable Programming in Science

---

# Source Transformation AD With Clad

---

- ❖ Completed two milestones (v1.3 in Mid Feb and v1.4 end of Feb)
- ❖ Fixed 49 86 197 275 314 429 439 441 465 606 620 650 655 660 664 667 669 672 676 681 687 689 in v1.3
- ❖ Fixed 300 313 636 735 748 753 774 in v1.4
- ❖ Next milestone: v1.5 due end of March

# Source Transformation AD With Clad

---

- ❖ v1.3
  - ❖ Added llvm17 support
  - ❖ Implemented experimental to-be-recorded analysis
  - ❖ Reduce the generated code size
  - ❖ Reduced tape usage
  - ❖ Improved performance
  - ❖ Initial support for pointers
- ❖ v1.4
  - ❖ Bugfixes related to RooFit

# 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

---

❖ No updates

# Status. Clang-Repl

---

- ❖ [clang-repl] Expose RuntimeInterfaceBuilder to allow customization [PR83126](#)
- ❖ [clang-repl] Names declared in if conditions and for-init statements are local to the inner context (C++ 3.3.2p4) [PR84150](#)
- ❖ [clang-repl] Pass triple to IncrementalCompilerBuilder as explicit argument [PR84174](#)
- ❖ [clang-repl] Refactor locking of runtime PTU stack (NFC) [PR84176](#)
- ❖ 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. CppInterOp

---

❖ Released v1.2.0



# Status. Xeus-Cpp

---

- ❖ CppInterOp v1.2.0 enabled working on merging xeus-clang-repl into xeus-cpp PR14

# Status. Xeus-Clang-Repl

---

❖ No updates

# Open Projects

---

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

# Next Meetings

---

- ❖ Monthly Meeting — 4th April, 1700 CET / 0800 PDT
  - ❖ Tentative — BioDynaMo

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.