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

Interactive C++: cling and clang-repl

Ioana Ifrim & Vassil Vassilev
Continuing to rebase cling on top of llvm13, need to support unloading for CallFunc; working on a prototype for the JIT deadlock issue

Improving the stability of cpt.py used to build and package cling
Status. Clang-Repl

- Error recovery for template instantiations completed
- Added support for weak symbols
- Fixed PPC exception issues for compiled and interpreted code
- Taught clang to parse statements on the global scope: D127284

The goal is to provide a more stable error recovery approach than the currently available one in cling
Within clang-repl, a python interactive session can now be invoked; here, python is aware of the variables declared in C++
Clang-Repl and Xeus

- We have Clang-Repl and Xeus integration for the C++ kernel
Clang-Repl and Xeus

- Working on bringing python features working in clang-repl to Jupyter
Status. InterOp

- Working on a full surgery of cppy where we split it into libInterOp
- Working on simplifying CallFunc and moving it in libInterOp: PR10850
Status. Clad

- Added basic infrastructure for integration with Enzyme
- Ongoing integration of Clad in RooFit, now the hf001 example works.
Upstreaming Patches

- Spreadsheet tracking the progress here.
- Reduced 8 more patches.
CaaS Open Projects

- Open projects are tracked in our open projects page.
Next Meetings

- Monthly Meeting — 4th Aug, 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!