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

Interactive C++: cling and clang-repl

Vassil Vassilev
Status. Cling

- Continuing to rebase cling on top of llvm13, fixed the CUDA failures (thanks to Simeon) and resolved the template instantiation assertion.
- Improving the stability of cpt.py used to build and package cling.
Status. Clang-Repl

- Advanced the support for code undo (code unloading).
- Sent an RFC on implementing a laxer-based robust continuation facility in clang.
- More progress on parsing statements on the global scope: D127284
- The jitlink AArch64 backend is ready and clang-repl uses it.
- The shared memory remote execution facility is progressing: D127491 D130392

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

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

- Making progress on the HPCCG benchmarks in the context of error estimation
Documentation

- Working on various documents to improve the documentation of cling and clad
- Restructured the website menus to capture better the project activities
Upstreaming Patches

- Spreadsheet tracking the progress here.
- Total amount of upstreamed cling patches 17 out of 52 upstreamable.
- The template resugaring work now has an godbolt entry.
Open projects are tracked in our open projects page.
Next Meetings

* Monthly Meeting — 1st Sep, 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!