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

Interactive C++: cling and clang-repl

Vassil Vassilev
No news here: We need to fix 6 tests in Cling
Status. Clang-Repl

- Incremental Input (RFC)
  - D143142 — Enable Lexer to grow its buffer
  - D143144 — Add TryGrowLexerBuffer/SourceFileGrower
  - D143148 — Add basic multiline input support

- Value Handling (RFC)
  - D141215 — Introduce Value and implement pretty printing. Quite far down the review process.
  - D146389 — Initial interactive CUDA support for clang-repl. Almost ready.

The goal is to provide better stability and robustness which can later cling can reuse.
Implemented allocate/deallocate interfaces
Implemented Construct/Destruct interfaces
Enabled smart pointer support
Enabled debugging output (llvm::DebugFlag)
Enabled debugging jitted code (clang-repl mode only)
Added CI builds that enable valgrind checks
Rebased to our cppyy forks.
libInterOp-based cppyy: passes 170/504 tests.
Status. Clad

- Code refactoring of the tests infrastructure
- Added constexpr and consteval test cases
- Improved support of 32bit
- Fixed bugs in clad::array init
Fixed a bug in declaration/statement disambiguation D147319
Upstreaming Patches

- Spreadsheet tracking the progress [here](#).
- Total amount of upstreamed cling patches 26(26+0) out of 52 upstreamable.
Open projects are tracked in our open projects page.
Next Meetings

- Monthly Meeting — 1st June, 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!