Interactive C++: cling and clang-repl

Vassil Vassilev

02.09.2021

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



Status. Clang-Repl

Clang-repl supersedes examples / clang-interpreter • ACAT.

Submit abstracts on automatic differentiation and interactive C++ for



Status. Cling

Continuing to rebase cling on top of llvm13 — we need to split the monorepo commits in our forks llvm.git and clang.git *

Worked towards xeus-clad combining clad and xeus-cling conda packages





Status. InterOp

The ctype-based template instantiation is complete. More details in my talk next month.





Status. Clad

- Added tutorials about clad under <u>compiler-research.org/tutorials/</u> • benchmarks!
- improvements are left.
- mode.

[Baidyanath] Hessian support is done. Integrated in ROOT. Working on

Garima] The error estimation framework is done. Tutorial is underway. The numerical differentiation fallback is complete. A few post-merge

[Parth] Work on automatic cross verification between forward and reverse



Plans

- Prepare a paper about the work we've completed.
- Enable error recovery for advanced C++ code (eg template instantiation) Accelerate upstreaming clang patches
- •
- Automatically differentiate the CUDA kernels (including computation scheduler)



CaaS Open Projects

Patches against clang.git

- Implement FileManager uncaching
- * Adapt the user of invalidateCache to its new signature
- Mark the file entry invalid, until reread
- Propagate cache flags from LookupFile() to FileManager::getFile()
- Pass the OpenFile flag also to DirectoryLookup
- * Do not load the source file just to get an irrelevant SourceLoc (ROOT-7111)

Open projects are tracked in our <u>open projects page</u>.

* Allow interfaces to operate on in-memory buffers with no source location info [Pratyush Das]



Next Meetings

- Monthly Meeting 6th October, 1700 CET/0800 PDT • Tentative talk schedule:
 - Language Interop Progress, Vassil, Princeton, Oct
 - LLDB, Raphael Isemann, Apple, Nov

If you want to share your knowledge/experience with interactive C++ we can include presentations at an upcoming next meeting



Thank you!