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

Interactive C++: cling and clang-repl

Vassil Vassilev

13.01.2022

Status. Cling

- Continuing to rebase cling on top of llvm13

Status. InterOp

- The document is up online here.

Status. Clad

- Developed benchmark capability
  - Ran ADBench and found several bottlenecks
- Developed some design proposals under the GitHub discussions page
  - Changing the custom derivatives design
  - Support for aggregate types

People

Purva Chaudhari
clang-repl
(Feb 2022-May 2022)

Plans

- Working on a detailed work plan for 2022
- 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)
  - Allow interfaces to operate on in-memory buffers with no source location info [Pratyush Das]
- Open projects are tracked in our open projects page.

Next Meetings

- Monthly Meeting — 3rd Feb, 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!