

Clacc: OpenACC Support for Clang and LLVM

Joel E. Denny, Seyong Lee, Jeffrey S. Vetter Future Technologies Group, ORNL <u>https://ft.ornl.gov/ dennyje@ornl.gov</u> August 31, 2020 OpenACC Summit

ORNL is managed by UT-Battelle, LLC for the US Department of Energy



This research was supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's Office of Science and National Nuclear Security Administration, responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative.



CAK RIDGE



What is Clacc?

- Goal •
 - OpenACC C/C++ support for Clang and LLVM
- Design ۲
 - Translate OpenACC to OpenMP to build on OpenMP support
- Availability
 - Hosted publicly by the LLVM DOE Fork
 - https://github.com/llvm-doe-org/llvm-project/tree/clacc/master _
- Funding •
 - US Exascale Computing Project (ECP)
- Contact •
 - Joel E. Denny (<u>dennyje@ornl.gov</u>)



OpenACC

More Science, Less Programming





Clacc: Two Compilation Modes

- Traditional compilation
 - OpenACC source \rightarrow executable
 - Similar to PGI or GCC
 - OpenMP serves as an internal IR
- Source-to-source
 - OpenACC source \rightarrow OpenMP source
 - Target other OpenMP compilers
 - Reuse other OpenMP tools
 - Port apps or benchmarks
 - Debug representation





Clacc: Does OpenMP have what OpenACC needs?



Source-to-source

• No, such as:

National Laboratory

- present clause \rightarrow TR8 present map type modifier
- no_create clause \rightarrow ?
- Two reference counters \rightarrow ?
- auto clause \rightarrow ?
- kernels construct \rightarrow ?
- Develop OpenMP extensions[†]
 - For correct behavior (e.g., present, no_create)
 - For range of possible behaviors (e.g., kernels, auto)
 - Prototype OpenMP extension before standardizing

Clacc: User Impact of OpenMP Extensions

- Traditional compilation mode
 - OpenMP is just an internal IR: Quietly use OpenMP extensions as needed
- Source-to-source mode
 - Report diagnostic if OpenMP extension is needed
 - Option to choose alternative, good-enough translation to standard OpenMP, such as:
 - present(x) \rightarrow create(x) \rightarrow map(alloc:x)
 - Two ref counters \rightarrow one ref counter
- OpenACC Profiling Interface
 - Built as wrapper around OpenMP 5.0 OMPT interface
 - Events require extensions (some start/end pairs)
 - Profiling data require extensions (e.g., source locations)
 - OMPT implementation must support extensions



Clacc: Status

- Language
 - Basic features are ready (e.g., data, parallel, loop directives and clauses)
 - Some important features missing (e.g., kernels directive, runtime library, C++ support)
- Profiling interface
 - All events ready except wait events
 - Some profiling data missing (e.g., var_name, kernel_name, num_gangs, num_workers, vector_length)
- Ongoing activities
 - CI for x86_64, Power 9, NVIDIA GPUs
 - Merging upstream Clang and LLVM into Clacc
 - Contributing mutually beneficial changes to Clang and LLVM upstream
 - Contributing improvements to the OpenACC spec (eventually OpenMP spec)



Clacc: Preliminary GPU Evaluation SPEC ACCEL (Feb. 2020)



Test Platform

CAK RIDGE

- Ubuntu 18.04.3, 32 GB DRAM
- Intel Core i7-7700HQ 2.8GHz CPU (8 threads)
- NVIDIA GeForce GTX 1050
- PGI Community Edition 19.10
- SPEC ACCEL 1.3, ref workload, non-compliant

314.omriq and 514.pomriq



Benchmarks

- OpenACC = 3xx benchmark
- OpenMP = 5xx benchmark (OpenMP offload version)
- OpenACC* = 3xx benchmark modified to have same levels of parallelism and private clauses as 5xx benchmark

Need descriptive OpenACC interpretation

What is Flacc?

• Goal

- OpenACC Fortran support for LLVM Flang (a.k.a. F18)
- Design
 - Translate OpenACC to MLIR OpenACC dialect
- Availability
 - Upstreamed to LLVM Flang as developed
- Funding
 - US Exascale Computing Project (ECP)
- Contact
 - Valentin Clement (clementv@ornl.gov)



Clacc Takeaways

- Overview
 - Goal: OpenACC C/C++ support for Clang and LLVM
 - Design: Translate OpenACC to OpenMP to build on OpenMP support
 - Availability: <u>https://github.com/llvm-doe-org/llvm-project/tree/clacc/master</u>
 - Contact: Joel E. Denny (dennyje@ornl.gov)
- Join Us
 - Oak Ridge National Laboratory
 - Hiring interns, postdocs, research and technical staff
 - External collaborators welcome

Clacc: Translating OpenACC to OpenMP in Clang, Joel E. Denny, Seyong Lee, and Jeffrey S. Vetter, 2018 IEEE/ACM 5th Workshop on the LLVM Compiler Infrastructure in HPC (LLVM-HPC), Dallas, TX, USA, (2018).

