MATTER AND TECHNOLOGIES

Contribution ID: **86** Type: **Poster**

Heterogeneous Programming APIs for Parallel Applications

Our Abstraction Library for Parallel Kernel Acceleration (Alpaka, [1]) grows in popularity and other heterogeneous programming C++ APIs are about to emerge. The ultimate goal is to be able to write easy-to-read and performance-portable single-source codes which run efficiently on multi-core CPUs, NVIDIA or AMD GPUs, or even FPGAs. We evaluate new coding standards and their implementations to not only increase platform coverage but also to simplify our programming interfaces. Alpaka already has a convenient CUPLA [2] layer to run CUDA codes on non-NVIDIA platforms. But how about Khronos' SYCL standard [3] or AMD's HC & HIP languages[4]? Are they conforming to the C++ standard or are they just a set of cumbersome language extensions?

This research is crucial for building a future-proof, yet convenient and expressive parallel programming interfaces for our application developers. We want to create lightweight and high-performant C++ modules which allow fast prototyping and integration of different algorithms, e.g. for image reconstruction. This poster will show the current work in progress that includes comparisons of the latest SYCL, HC, HIP and CUDA releases on different platforms.

[1] https://github.com/ComputationalRadiationPhysics/alpaka
[2] https://github.com/ComputationalRadiationPhysics/cupla
[3] https://www.khronos.org/sycl/
[4] https://rocm-documentation.readthedocs.io/en/latest/Programming_Guides/Programming-Guides.html

Primary authors: Mr STEPHAN, Jan (Helmholtz-Zentrum Dresden-Rossendorf); Mr WERNER, Matthias (Helmholtz-Zentrum Dresden-Rossendorf)

Co-author: Dr BUSSMANN, Michael (Helmholtz-Zentrum Dresden - Rossendorf)

Presenter: Mr STEPHAN, Jan (Helmholtz-Zentrum Dresden-Rossendorf)

Track Classification: DMA