Harnessing the full power of modern hardware accelerators using idomatic C++

STEFAN BUTZ, University of Hagen, Germany

1 EXPOSE

As part of the seminar system architecture, I would like to present a novel approach to harness the full power of modern hardware accelerators in pure C++.

C++17 allows users to request parallel execution of algorithms provided by the standard library.

The SYCL programming model is a roalty-free, high-level programming model developed by the Khronous Group. It features a special compiler allowing users to write single-source C++ code that can be executed on different hardware accelerators. In case of adaptive cpp, it can support different backends, such as AMD ROCm, OpenCL, and CUDA. Empowering users to make easily use of wide range of modern hardware accelerators.

Aksel Alpay and Vincent Heuveline of Heidelberg University have shown in their paper *AdaptiveCpp Stdpar: C++ Standard Parallelism Integrated Into a SYCL Compiler* how the SYCL programming model can be used as a backend for C++ standard parallelism. Their work allows users to seamlessly integrate parallel programming into their C++ code, while using idomatic C++ constructs.

My paper will introduce the reader to C++ standard parallelism interface and the SYCL programming model. On the basis of the work of Aksel Alpay and Vincent Heuveline, I will show how the two interfaces can be unified. I will identify the challenges that arise when unifying the two interfaces and show how the authors have solved these problems. I will compare the performance of their solution to basic SYCL programming and summarize the advantages and disadvantages that can guide users. Finally, I will give an outlook on the features of C++26 and how these features can help to integrate parallel programming into C++ even more seamlessly.

Author's address: Stefan Butz, University of Hagen, Hagen, Germany, stefan.butz@studium.feruni-hagen.de.