Contribution ID: **111** Type: **Poster**

Adoption of the Alpaka Performance Portability Library in the CMS Software

Since the beginning of Run 3 of LHC the CMS experiment, to cope with the higher luminosity and a larger number of simultaneous proton-proton collisions (pile-up), is offloading some of the most computational intensive tasks of the online (HLT) reconstruction to NVIDIA GPUs, while the support for AMD and Intel GPUs is under development.

Offloading the pixel tracks and calorimeter reconstruction has allowed to increase the rate of event processed while being cheaper in terms of power consumption and hardware costs and also improving the physics performance of CMS reconstruction. The success of this experience has propelled a series of efforts within the collaboration to allow more and more algorithm to be executed on heterogeneous architectures.

To avoid the need to write, validate and maintain a separate implementation of the reconstruction algorithms for each back-end, CMS decided to adopt a performance portability framework. After evaluating different alternative, as the solution for Run-3, it was decided to adopt Alpaka, a header-only C++ library that provides performance portability across different back-ends, abstracting the underlying levels of parallelism.

This contribution will show how Alpaka is used inside CMSSW to write a single code base; to use different toolchains to build the code for each supported back-end, and link them into a single application; and to select the best back-end at runtime.

Primary author: DI FLORIO, Adriano (Istituto Nazionale di Fisica Nucleare)

Session Classification: Parallel Demo + Poster Session

Track Classification: Esperimenti e Calcolo Teorico