

Performance portability of random number generation in SYCL

The increasing number of high-performance computing centers around the globe is providing physicists and other researchers access to heterogeneous systems – comprising multiple central processing units and graphics processing units per node – with various platforms. However, it is more often than not the case that domain scientists have limited resources such that writing multiple implementations of their codes to target the different platforms is unfeasible. To help address this, a number of portability layers are being developed that aim to allow programmers to achieve performant, portable codes; for example, Intel Data Parallel C++ (DPC++), which is based on the SYCL programming model. Nevertheless, portable application programming interfaces often lack some features and tools that are manifest in a platform-specific API. Physicists in particular rely heavily on large sets of random numbers in nearly their entire workflow, from event generation to analysis. In this talk, we detail the implementation of AMD and NVIDIA GPU support into oneMKL, permitting random number generation within SYCL-based applications using optimized libraries. By utilizing existing optimizations and SYCL interoperability, we demonstrate the ability to achieve nearly native performance in cross-platform applications.

Are you are a member of the APS Division of Particles and Fields?

Yes

Primary author: PASCUZZI, Vincent (Lawrence Berkeley National Lab. (US))

Presenter: PASCUZZI, Vincent (Lawrence Berkeley National Lab. (US))

Session Classification: Computation, Machine Learning, and AI

Track Classification: Computation, Machine Learning, and AI