Contribution ID: **41** Type: **not specified**

First experiences at the exascale with Parthenon –a performance portable block-structured adaptive mesh refinement framework

Thursday, 26 September 2024 09:30 (20 minutes)

On the path to exascale the landscape of computer device architectures and corresponding programming models has become much more diverse. While various low-level performance portable programming models are available, support at the application level lacks behind. To address this issue, we present the performance portable block-structured adaptive mesh refinement (AMR) framework Parthenon, derived from the well-tested and widely used Athena++ astrophysical magnetohydrodynamics code, but generalized to serve as the foundation for a variety of downstream multi-physics codes. Parthenon adopts the Kokkos programming model, and provides various levels of abstractions from multi-dimensional variables, to packages defining and separating components, to launching of parallel compute kernels. Parthenon allocates all data in device memory to reduce data movement, supports the logical packing of variables and mesh blocks to reduce kernel launch overhead, and employs point-to-point, asynchronous MPI calls to reduce communication overhead in multi-node simulations. At the largest scale, a Parthenon-based hydrodynamics miniapp reaches a total of 17 trillion cell-updates per second on 9,216 nodes (73,728 logical GPUs) on Frontier at ~92% weak scaling parallel efficiency (starting from a single node). In this talk, I will highlight our performance-motivated key design decisions in developing Parthenon. Moreover, I will share our experiences and challenges in scaling up with an emphasis on handling the number of concurrent messages on the interconnect, writing large output files, and post-process them for visualization –which also translate to other applications.

Primary author: GRETE, Philipp (University of Hamburg)

Presenter: GRETE, Philipp (University of Hamburg)

Session Classification: Session 3.2: Exascale-ready codes for astrophysical problems: case studies and implementations