HACC I/O Benchmark Summary

Summary Version
1.0

Purpose of Benchmark

The purpose of this benchmark is to evaluate the performance of the I/O system for the Hardware Accelerated Cosmology Code (HACC) simulation. The HACC framework uses N-body techniques to simulate the formation of structure in collisionless fluids under the influence of gravity in an expanding universe.

Characteristics of Benchmark

The HACC I/O benchmark capture the I/O patterns of the HACC simulation code. This includes the checkpoint and restarts as well as the analysis outputs produced by the simulation. It also captures the various I/O interfaces used in HACC, namely, POSIX I/O, MPI Collective I/O and MPI Independent I/O. Additionally, the benchmark can either write out a single shared file, file per process, and a file per group of processes (partition).

Mechanics of Building Benchmark

The HACC I/O code is an MPI-only code. There is a makefile to build the benchmark. It relies on POSIX and MPI-IO.

Offeror may modify the RestartIO_GLEAN.cxx file to customize the __initalizePartitionInfo function for their I/O platform. A generic partitioning mechanism is included.

Mechanics of Running Benchmark

The benchmark takes in a number of command line parameters. These include the number of particles per rank, the type of I/O mechanism to be used (POSIX, MPI-IO – both collective and independent I/O), and filename.

Example runs:

Small problem: Representative of the analysis outputs
Particle data will need ~2% of the entire system memory

Medium problem:
Particle data will need ~10% of the entire system memory

Large problem:
Particle data will need ~60% of the entire system memory

CORAL class problem:
Particle data will need ~60% of the entire system memory

1

Verification of Results

The benchmark has capabilities to validate the I/O. A known data pattern is written out as a checkpoint. This is read back in and verified with the known pattern.