Available online at www.prace-ri.eu

Partnership for Advanced Computing in Europe

Parallel I/O Performance Benchmarking and Investigation on Multiple HPC Architectures

B. Lawrence[a], C. Maynard[b], A. Turner[c], X. Guo[c], D. Sloan-Murphy[c]

[a]*University of Reading, United Kingdom*
[b]*Met Office, United Kingdom*
[c]*Edinburgh Parallel Computing Centre, University of Edinburgh, United Kingdom*

Summary

Solving the bottleneck of I/O is key in the move towards exascale computing. Centres of Excellence and research communities must be informed of the I/O performance of current petascale resources in order to make informed plans for future procurements and software development projects. This paper therefore presents benchmarks for the write capabilities of the following HPC systems:

- ARCHER: the UK national supercomputing service, with a Cray Sonexion Lustre file system.
- COSMA: one of the DiRAC UK HPC resources, using a DDN implementation of the IBM GPFS file system.
- UK-RDF DAC: the Data Analytic Cluster attached to the UK Research Data Facility, also using DDN GPFS.
- JASMIN: a data analysis cluster delivered by the STFC, using the Panasas parallel file system.

We run *benchio*, a parallel benchmarking application which writes a three-dimensional distributed dataset to a single shared file. On all systems, we measure MPI-IO performance and, in select cases, compare this with HDF5 and NetCDF equivalent implementations.

We find a reasonable expectation is for approximately 50% of the theoretical system maximum bandwidth to be attainable in practice. Contention is shown to have a dramatic effect on performance. MPI-IO, HDF5 and NetCDF are found to scale similarly but the high-level libraries introduce a small amount of performance overhead.

For the Lustre file system, on a single shared file, maximum performance is found by maximising the stripe count and matching the individual stripe size to the magnitude of I/O operation performed. HDF5 is discovered to scale poorly on Lustre due to an unfavourable interaction with the H5Fclose() routine.