Reservation-based I/O Performance Guarantee for MPI-IO Applications using Shared Storage Systems

Yusuke Tanimura1), Rosa Filgueira2), Isao Kojima1) and Malcolm Atkinson2)
1) National Institute of Advanced Industrial Science and Technology (AIST), Japan, 2) The University of Edinburgh, UK

Background
There is a big problem in concurrent use of the parallel storage systems on HPC Clusters. This might spoil the performance improvement that might otherwise be obtained by optimizations of MPI-IO, such as data sieving, two-phase collective I/O and etc.

- MPI-IO is important for large data analysis.
- Initial input, final output, and checkpointing
- Total throughput degradation and instability by various, internal and external interferences occur in HPC clusters.
- For example, access contention may happen, a) when more than two parallel programs call MPI-IO, b) when external access (for pre/post processing) is executed.

Advance Reservation Approach
Our approach to achieve performance guarantees is to allow users / applications to explicitly reserve I/O throughput of the storage systems on HPC Clusters, with start and end time of the access.

- Neither overprovisioning nor reactive QoS mechanisms
- SLO (Service Level Objectives) : Read or write throughput (e.g., MB/sec) in a single access (open ~ close).
- Striped access is automatically enabled if necessary.
- Measurement granularities are user-defined.

Integration with the batch scheduler
Users tell necessary I/O throughput (and job execution or I/O time) to a batch scheduler (BS) when they submit a job. Then BS reserves I/O throughput, a) during entire execution of the job, b) during specific I/O time.

We expect that iterated jobs and system-level checkpointing may have steady execution which allows users to estimate execution time of the job and its I/O time.

Design and Implementation
We have been developing a performance guarantee software called Papio. In order to examine an effect of the performance guarantee of MPI-IO, we developed the ADIO layer of Papio for Dynamic-CoMPI.

- Papio [Y. Tanimura et al. in Grid 2010]: Parallel I/O storage architecture with performance guarantee functionality based on advance reservation (Note that the reservation is mandatory for now.)
  - Assign available resources to the reserved access: fully occupied or shared
  - If shared, control I/O throughput of the storage system and I/O scheduling of disks
- Dynamic-CoMPI [R. Filgueira et al. in J. Supercomputing 2010]: Implement advance features based on MPICH2
  - Locality aware strategy for Two-Phase I/O: Optimized data aggregation into contiguous buffers, and sequential transfers into the file/storage system
  - Adaptive-CoMPI: Run-time adaptive message compression
- ADIO layer of Papio (ad_papio) in ROMIO
  - Support collective calls: MPI_File_write_all(), MPI_File_read_all()