Data staging technique for improving post-processing performance in large-scale CFD analysis

Report Number: R19EACA42
Subject Category: JSS2 Inter-University Research
URL: https://www.jss.jaxa.jp/en/ar/e2019/11556/

Responsible Representative

Keichi Takahashi, Assistant Professor, Nara Institute of Science and Technology

Contact Information

Keichi Takahashi(keichi@is.naist.jp)

Members

Keichi Takahashi

Abstract

Conventional post-processing of CFD simulations was achieved by saving the entire simulation output on a parallel file system and then processing the output. However, this approach is becoming increasingly challenging due to the limitations in storage size and IO bandwidth. Therefore, data staging, where the simulator transfers its output to a post-processing application during runtime, is attracting attention. In this research, we evaluate the feasibility of leveraging data staging technologies on HPC environments exemplified JSS2 and analyze the requirements for data staging middleware and HPC environment.

Reasons and benefits of using JAXA Supercomputer System

We used JSS2 because it comprises two subsystems, which are the main compute system (SORA-MA) and the pre/post-processing system (SORA-PP), and it allows communication between the two subsystems.

Achievements of the Year

In this fiscal year, we improved the practicality of staging between SORA-MA and SORA-PP. In the last fiscal year, we ported ADIOS2, which is a staging middleware developed at Oak Ridge National Laboratory, and successfully achieved staging communication between MA and PP. We ran ADIOS2's adios-reorganize utility on the IO nodes in MA to relay the communication between MA and PP.

However, the following two issues remain when applying staging to a large-scale CFD analysis spanning over thousands of processes: (1) low throughput: the achieved throughput is less than 10% of the bandwidth of the underlying interconnect on both subsystems. (2) large memory footprint of the communication bridge: the communication bridge consumes up to 2x - 3x of the data size. We tackled these two issues since they are critical obstacles when applying staging to large-scale CFD analysis.

Regarding issue (1), we believe that this is caused by the fact that ADIOS2's staging engine SST uses TCP/IP

instead of RDMA. In fact, RDMA achieved much higher throughput than TCP/IP on the two systems (measured using network benchmarks). Regarding issue (2), we analyzed the memory allocations and deallocations in adios-reorganize using a heap profiler (Valgrind Massif). The profiling results revealed that the SST engine allocates multiple redundant buffers for communication. Furthermore, dynamically growing the buffers is causing reallocations and making the memory footprint even larger.

We believe that using SSC, which is a newly developed staging engine in ADIOS2, can solve these two issues. Since SSC's backend is MPI, RDMA is used instead of TCP/IP. Also, SSC maintains fewer buffers than SST. Based on these analysis and discussion, we ported the SSC engine to MA and PP and confirmed that functional tests are passing with SSC. In the next year, we conduct a detailed performance evaluation using the SSC engine and verify if issues (1) and (2) are successfully solved.

Publications

- Non peer-reviewed papers

Seiji Tsutsumi, Naoyuki Fujita, Hiroyuki Ito, Daichi Obinata, Keisuke Inoue, Yosuke Matsumura, Keichi Takahashi, Greg Eisenhauer, Norbert Podhorszki, Scott Klasky, "In Situ/In Transit Approaches for Post-Processing in Large-Scale Numerical Simulation", 33rd CFD Symposium.

- Oral Presentations

Seiji Tsutsumi, Naoyuki Fujita, Hiroyuki Ito, Daichi Obinata, Keisuke Inoue, Yosuke Matsumura, Keichi Takahashi, Greg Eisenhauer, Norbert Podhorszki, Scott Klasky, "In Situ and In Transit Visualization for Numerical Simulations in HPC", In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization (ISAV 2019).

Usage of JSS2

Computational Information

Process Parallelization Methods	MPI
Thread Parallelization Methods	N/A
Number of Processes	1 - 128
Elapsed Time per Case	5 Minute(s)

- **Resources Used**

Fraction of Usage in Total Resources[*1](%): 0.00

Details

Computational Resources		
System Name	Amount of Core Time (core x hours)	Fraction of Usage[*2](%)
SORA-MA	11,189.95	0.00
SORA-PP	275.58	0.00
SORA-LM	0.00	0.00
SORA-TPP	0.00	0.00

File System Resources		
File System Name	Storage Assigned (GiB)	Fraction of Usage*2(%)
/home	9.54	0.01
/data	95.37	0.00
/ltmp	1,953.13	0.17

Archiver Resources		
Archiver Name	Storage Used (TiB)	Fraction of Usage*2(%)
J-SPACE	0.00	0.00

[*1]: Fraction of Usage in Total Resources: Weighted average of three resource types (Computing, File System, and Archiver).

[*2]: Fraction of Usage：Percentage of usage relative to each resource used in one year.