Poster: A Scalable Two-Phase Parallel I/O Library with Application to a Large Scale Subsurface Simulator

Sarat Sreepathi
North Carolina State University
Raleigh, NC, USA
sarat_s@ncsu.edu

Vamsi Sripathi
North Carolina State University
Raleigh, NC, USA
vksripat@ncsu.edu

Glenn Hammond
Pacific Northwest National Laboratory
Richland, WA, USA
glenn.hammond@pnnl.gov

Richard Mills
Oak Ridge National Laboratory
Oak Ridge, TN, USA
rmills@ornl.gov

G.Kumar Mahinthakumar
North Carolina State University
Raleigh, NC, USA
gmkumar@ncsu.edu

ABSTRACT

This poster describes the development of a highly scalable application layer parallel I/O library (ASCEM-IO) for scientific applications.This library was envisioned to leverage our earlier I/O optimization experience to build a scalable general purpose parallel I/O capability for any application. The parallel I/O library provides a higher level API (Application Programming Interface) to read and write large scientific datasets in parallel at very large processor counts. Specifically, the goal is to take advantage of existing parallel I/O libraries, such as HDF5 which are being widely used by scientific applications and modify these algorithms to better scale on larger number of processors. This is accomplished by dividing the traditional I/O operations (read/write) into two phases, a communication phase and an I/O phase. Results with a real application on the Cray XT/5 indicates significant performance improvement on large processor cores when compared to default HDF collective I/O operations.

Categories and Subject Descriptors

D.0 [**Software**]: General

General Terms

Algorithms, Performance

Keywords

Parallel I/O, High Performance Computing

1. INTRODUCTION

Inefficient parallel I/O is known to be a major bottleneck among scientific applications employed on supercomputers as the number of processor cores grows into the thousands. Our prior experience indicated that parallel I/O libraries such as HDF5 that rely on MPI-IO do not scale well beyond 10K processor cores, especially on parallel file systems (like Lustre) with single point of resource contention. Our previous optimization efforts for a massively parallel multi-phase and multi-component subsurface simulator (PFLOTRAN) led to a two-phase I/O approach at the application level where a set of designated processes participate in the I/O process by splitting the I/O operation into a communication phase and a disk I/O phase. The designated I/O processes are created by splitting the MPI global communicator into multiple sub-communicators. The root process in each sub-communicator is responsible for performing the I/O operations for the entire group and then distributing the data to rest of the group. This approach resulted in over 25X speedup in HDF I/O read performance and 3X speedup in write performance for PFLOTRAN at over 100K processor cores on the ORNL Jaguar supercomputer.

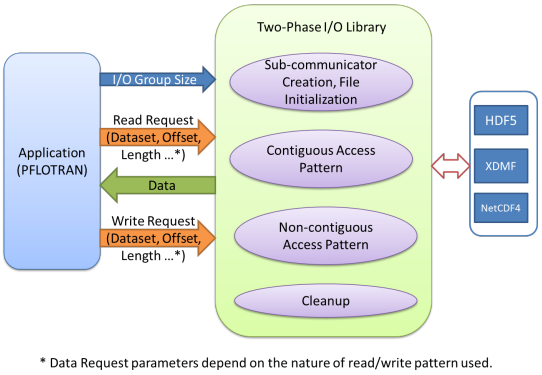

Figure 1: Overall architecture of the ASCEM-IO library

2. ARCHITECTURE

This research describes the design and development of a general purpose parallel I/O library that incorporates our optimized two-phase I/O approach. The basic architecture of the ASCEM-IO library is depicted in Fig. 1.

This library is part of a larger DOE funded effort called Advanced Simulation Capability for Environmental Management (ASCEM) that is developing next generation sim-

Copyright is held by the author/owner(s).
SC'11 Companion, November 12–18, 2011, Seattle, Washington, USA.
ACM 978-1-4503-1030-7/11/11.

ulation tools for environmental management applications. The library provides a simplified higher level abstraction to the user, sitting atop existing parallel I/O libraries (such as HDF5) and implements optimized I/O access patterns that can scale on larger number of processors.The user specifies the I/O group size and basic attributes of the I/O operation. Advanced users can create multiple concurrent I/O groups of different sizes customized for different kinds of access patterns. The library includes support for various types of read and write operations through well-defined Fortran, C and C++ compatible bindings. Performance results with standard benchmark problems and PFLOTRAN indicate that our library is able to maintain the same speedups as before with the added flexibility of being applicable to a wider range of I/O intensive applications.

We integrated this library with a second application - AMANZI, the core ASCEM (Advanced Simulation Capability for Environmental Management) simulation code. We have successfully executed test problems using AMANZI and ASCEM-IO up to 256 processors on Cray XE6 machine at NERSC demonstrating the generic utility of the library.

In order to derive further performance gains while writing multi-dimensional blocks, consolidation strategies are designed and currently in development to reduce the number of independent I/O requests.

2.1 Supported I/O Patterns

The ASCEM-IO library supports various commonly used I/O patterns such as `uniform` and `nonuniform` contiguous access patterns. In the `uniform` scenario, every process accesses the same extent/length of data whereas in the `nonuniform` scenario, each process can access different extents of the dataset. Advanced users have the flexibility to customize their I/O requests and offsets according to their needs using the non-uniform access pattern. In addition to these patterns, a generic block writing pattern is implemented to support writing an n-dimensional data block to a specified offset in the global dataset. The library enhances user productivity by implicitly calculating the requisite offsets and lengths wherever possible.

3. PERFORMANCE

PFLOTRAN is a highly scalable subsurface simulation code that solves multi-phase groundwater flow and multi-component reactive transport in three-dimensional porous media. A micro-benchmark that includes part of PFLOTRAN's I/O behavior was developed to test the efficacy of the ASCEM-IO library. Performance analysis was carried out on the ORNL Jaguar PF (Cray XT5) machine employing up to 144K cores using two real datasets associated with the PFLOTRAN application. The I/O operation consists of reading two one-dimensional datasets from an input file used in the PFLOTRAN application and then writing them to an output file. The two-phase ASCEM-IO was compared against the default HDF collective I/O approach in Fig. 2. Timing is nearly flat for ASCEM-IO for the large file scenario whereas for default HDF I/O it increases nearly exponentially. ASCEM-IO is around 12 times faster than default I/O for 1.7 GB file scenario and more than 4 times faster for the 14 GB file.

We integrated the two-phase I/O library with PFLOTRAN and used it to study performance gains over the default I/O approach for a $960 \times 960 \times 240$ grid test problem (1.7 GB input and 5.8 GB output) as shown in Fig 3. We expect the performance gains to improve further for larger test problems using multiple datasets as PFLOTRAN invokes a file open/read call for every dataset read and file open is an expensive call at scale. Moreover, write performance can be further enhanced by consolidating the writes in the two-phase I/O library.

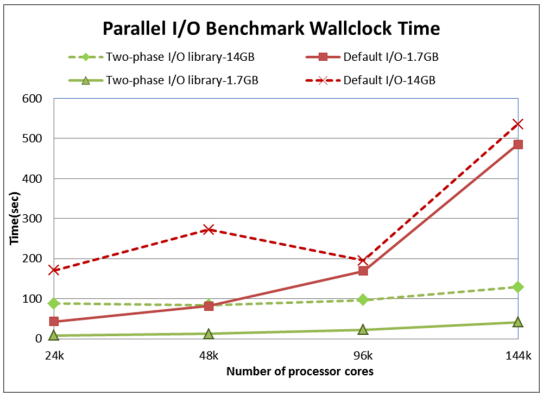

Figure 2: I/O performance of default I/O (HDF collective) with two-phase ASCEM-IO library. At 144k cores the wallclock time for default I/O method is 535 secs compared to 129 secs for two-phase library method for the 14 GB file scenario.

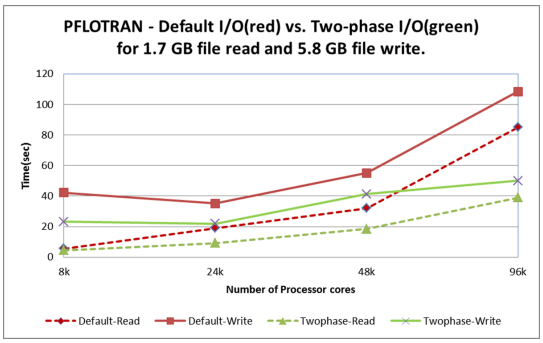

Figure 3: PFLOTRAN performance with default I/O and two-phase I/O library on Jaguar.

4. ACKNOWLEDGMENTS

This research used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the Department of Energy under Contract DE-AC05-00OR22725. The initial two-phase I/O development work specific to the PFLOTRAN application has been supported by the Performance Engineering Research Institute (PERI) and is funded by the U.S. Department of Energy Office of Science, as part of its second Scientific Discovery through Advanced Computing (SciDAC-2) research program. The I/O library development work is funded by the DOE Office of Environmental Management ASCEM project (Advanced Simulation Capability for Environmental Management).