SCALABLE COMPUTING
SOFTWARE LABORATORY

HDF5 VOL Connector to Apache Arrow

Jie Ye [1] Anthony Kougkas [1] Xian-He Sun [1]

[1]Illinois Institute of Technology

ILLINOIS INSTITUTE
OF TECHNOLOGY

Introduction and Motivation

Recently, Apache Arrow is very popular and widely used in Big Data Analysis and Cloud Computing community because of its standardized in-memory column format [1].It is an open-source, columnar, in-memory data representation that enables analytical systems and data sources to exchange and process data in real-time.It could create an efficient in-memory column store that can be used to manage streamed data.However, most scientific applications store and access data through HDF5 [2], a common used I/O middle-ware on HPC systems. HDF5 is inefficient in accessing column-oriented data streams. Accessing Apache Arrow data through HDF5 calls would allow applications to take advantage of these transient, column-oriented data streams, such as real-time data from high-speed scientific instruments and cameras. Moreover, bridging the gap between science applications and analytic tools that use HDF5 and Apache Arrow data could bring new kinds of data together. Therefore, this work introduces a HDF5 VOL connector which allows applications to access Apache Arrow data through native HDF5 calls.

The object of this work are:

- First, design and implement a HDF5 VOL connector which allows applications to access Apache Arrow data through native HDF5 calls.
- Second, explore its use for analyzing scientific data

Background

HDF5

- A well-established data model, parallel I/O library, and file format for storing and managing data
- A popular and widely used software library among HPC and Big Data Applications

HDF5 Virtual Object Layer

- A storage abstraction layer within the HDF5 library
- Allows applications to connect to different storage mechanisms transparently without significantly code changes

Apache Arrow

- An open-source, standardized columnar, in-memory data representation that enables analytical systems and data sources to exchange and process data in real-time
- Could reduce the overhead of copy and convert when moving data from one system to another
- Support different kinds of file formats, such as parquet file format [3] and arrow feather format [4], for persistent storage
- Support different compression strategies when storing data on disk, such as ZSTD and LZ4
- Arrow Plasma in-memory Object Store [5] is a high performance shared memory object store, which could share data between different applications
- Arrow Flight RPC [6] is an RPC framework for high performance data transportation based on Arrow data, enabling different applications to communicate with each other

Arrow VOL Design and Implementation

Arrow VOL connector is a terminal VOL connector which allows science applications to access Apache Arrow data through native HDF5 calls without significant code modifications. Figure 1 shows the Apache Arrow location within VOL. Figure 2 shows the internal work-flow in Arrow VOL Connector.

Arrow VOL Design and Implementation

Currently, the Arrow VOL connector implementation supports:

- in-memory column data access
- Write/read data through different back-end storage, like Apache Arrow Plasma in-memory object store and parallel file system
- Write/read data through Apache Arrow Flight RPC, enabling different applications to communicate with each other in real-time

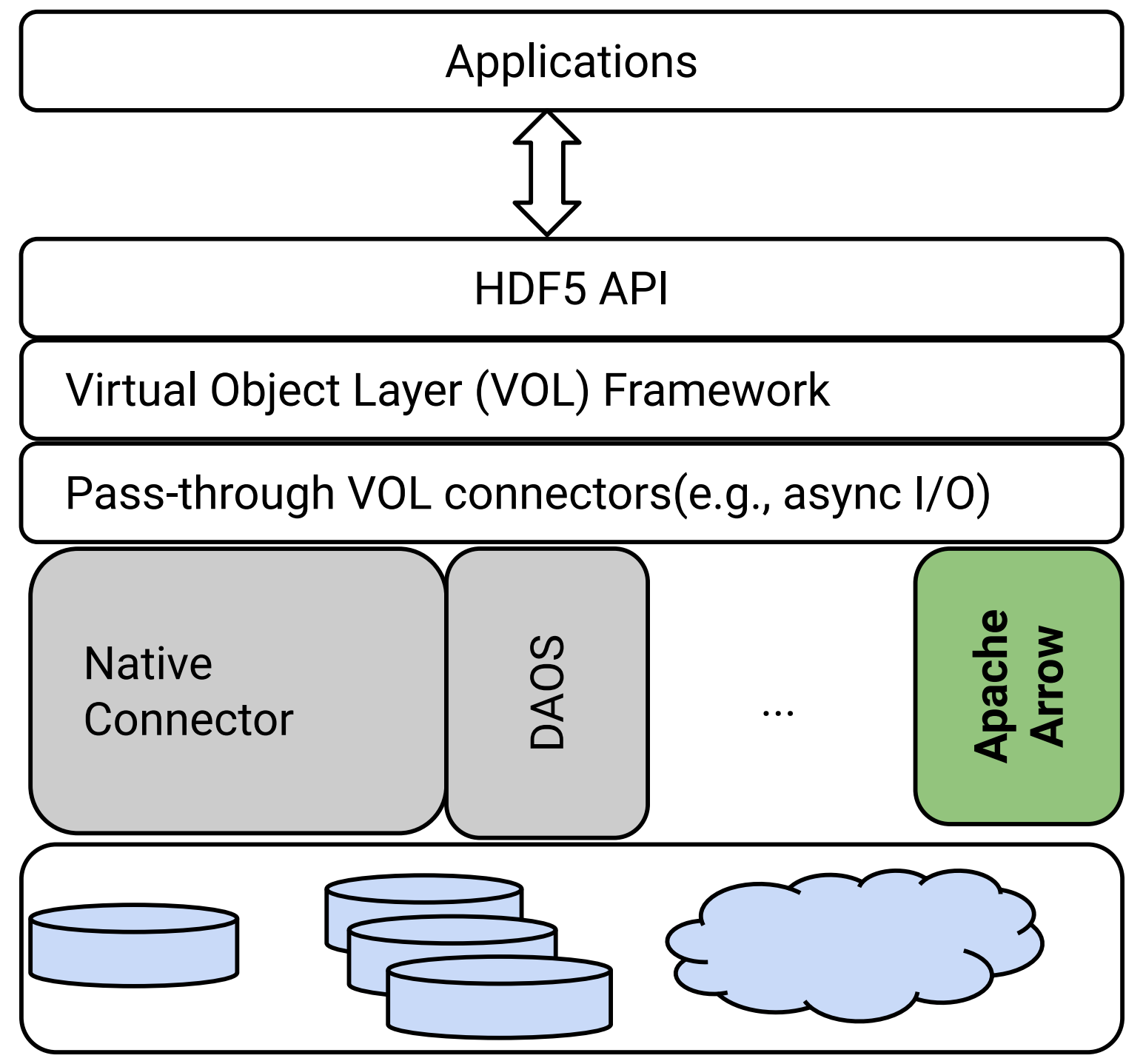
Applications
HDF5 API
Virtual Object Layer (VOL) Framework
Pass-through VOL connectors(e.g., async I/O)
Native Connector
DAOS
...
Apache Arrow

Figure 1: Apache Arrow within VOL

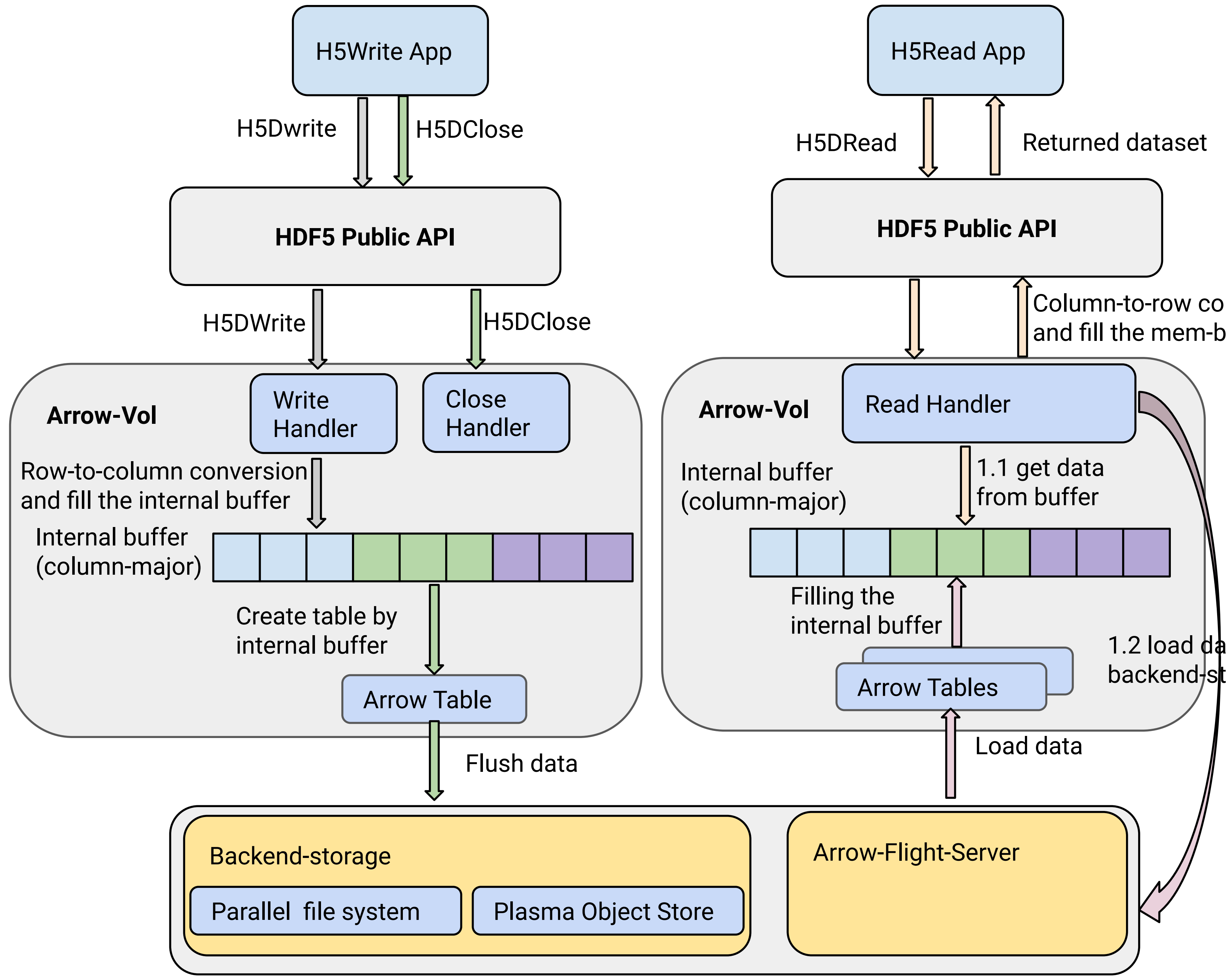

Figure 2: Internal workflow in Arrow VOL Connector

Results

All tests were conducted on the Cori Supercomputer at the National Energy Research Scientific Computing Center (NERSC)

- A Cray XC40 supercomputer with 1630 Intel Xeon Haswell nodes.
- Each node consists of 32 CPU cores and 128GB memory.
- The supporting storage system is Lustre, a widely-used parallel file system in HPC community.

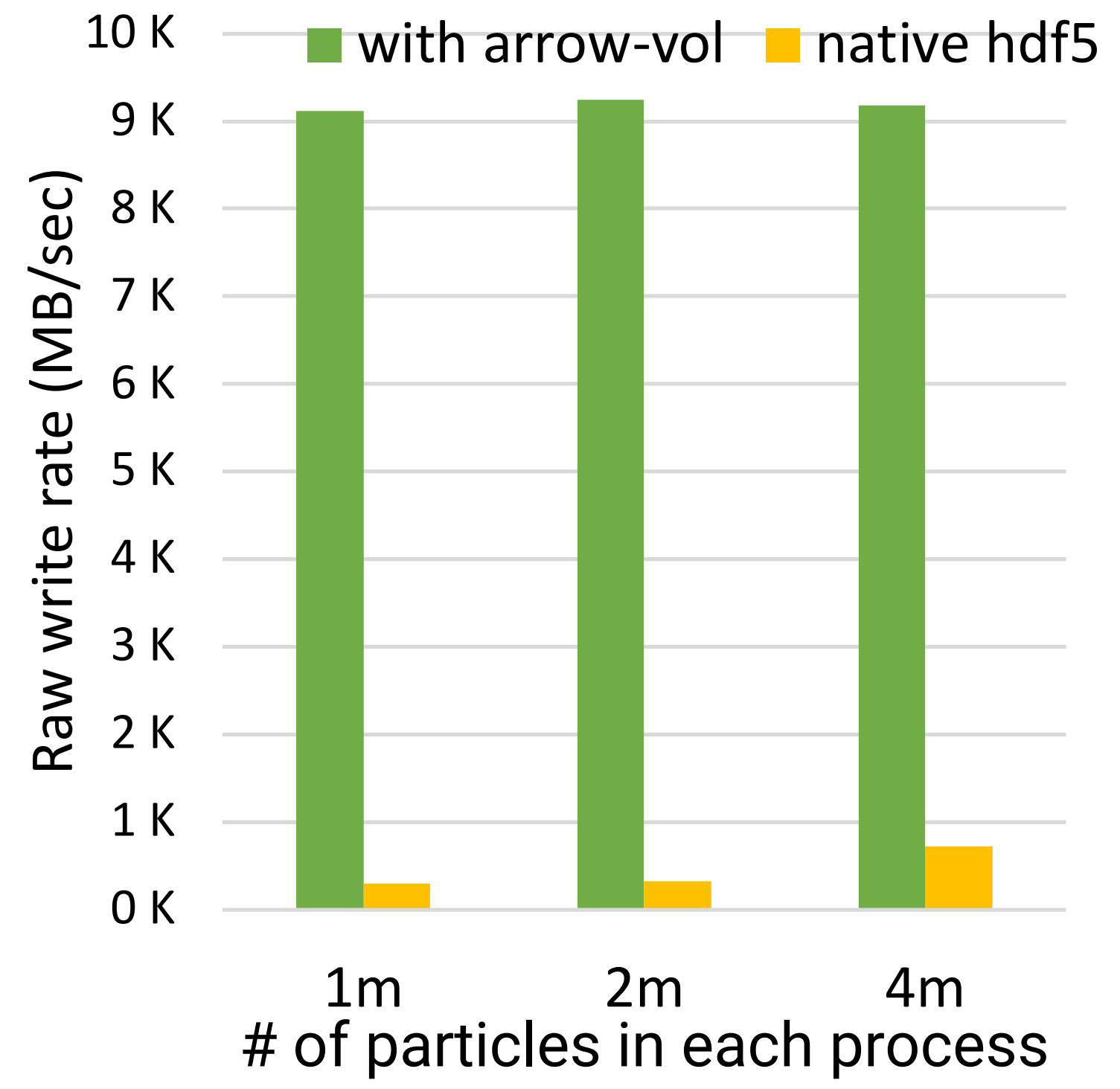

Figure 3: Write Performance

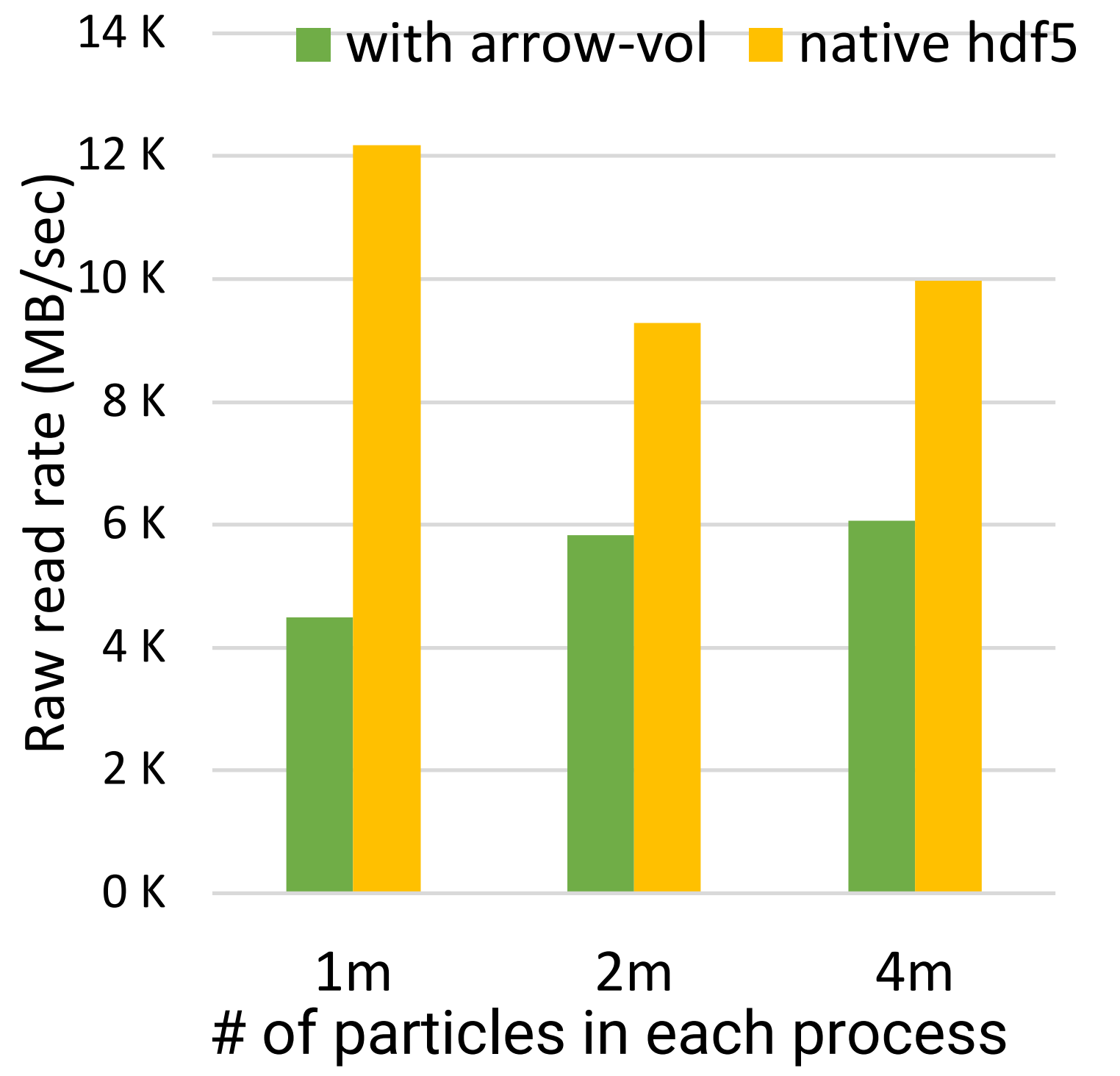

Figure 4: Read Performance

During the experiment, We evaluated the write performance by VPIC-IO, a plasma-physics application's I/O kernel, and the read performance through BD-CAST I/O kernel, which is used for analyzing the data produced by particle simulation. All the experiments are executed on 4 nodes with 128 processes.In VPIC-IO, each MPI process writes a region with different number of particles (such as 1M, 2M and 4M) and each particle has 8 properties. Figure 3 shows the write performance while Figure 4 presents the read performance.

Conclusions

In this work, we designed and implemented a HDF5 VOL connector to Apache Arrow that enables science applications to access Apache Arrow data through native HDF5 calls without changing the original code. We also have seen the initial write/read performance results when using Arrow-VOL Connector and native HDF5. We verified that Apache Arrow can be used into HPC system through this work, which laid the foundation for our future work.

References

[1] "Apache arrow." https://arrow.apache.org/, 2021.

[2] "Hdf5 library." https://portal.hdfgroup.org/display/HDF5/HDF5, 2021.

[3] "Apache parquet." https://parquet.apache.org/, 2021.

[4] "Apache arrow feather file format." https://arrow.apache.org/docs/format/Columnar.html, 2021.

[5] "Apache arrow plasma in-memory object store." https://arrow.apache.org/docs/python/plasma.html, 2021.

[6] "Apache arrow flight rpc framework." https://arrow.apache.org/docs/format/Flight.html, 2021.

jye20@hawk.iit.edu