# Suren Byna, et al. Tuning HDF5 subfiling performance on parallel file systems.

## 1 前言

Subfiling是用在并行文件系统（见图1）上的一项降低当多个计算节点与相同的I/O节点或存储目标交互时，发生的Locking与Contention问题的技术。

Subfiling在单个共享文件方法（探讨PFS上的Lock contention问题）和一个进程一个文件（one file per process，导致产生大量、无法管理数目的文件）之间做出折中(compromise)。

Subfiling，收集在拓扑上相互紧邻的计算rank到一个group，每个group访问一个单独的文件，这将增加存储到文件系统的带宽，因为可避免大多数的资源竞争问题。

并行文件格式库，如PnetCDF和ADIOS，都实施了subfiling技术。



图1 典型的HPC存储层级：短期的节点存储和数据的快速存储，以及全局的用于长期存储的并行文件系统(PFS) (Zheng et al., 2022)

本文对HDF5实施subfiling，并测试在NERSC的超算系统Cori和Edsion上的I/O存储效率。

Edison是基于disk的Lustre文件系统；Cori是包含一个基于SSD的burst buffer存储系统（Cray DataWarp管理）和一个基于disk的Lustre文件系统。

表明：相比写出到一个单独的共享HDF5文件，subfiling写出有1.2~6.5x的加速。

推荐了选择subfiles数和在存储节点上如何布局数据。

本文内容：

（1）介绍HDF5中的subfiling的实施，以及HDF5的Virtual Datasets (VDS)的特点。

（2）在PFS上，HDF5的subfiling效率评估。Plasma physics场景。

（3）调试参数，如：subfiles数、存储目标数，获得使用SSD-based burst buffer PFS以及disk-based Lustre PFS上的最佳效率。

## 2 背景知识(subfiling)

HDF5使用POSIX I/O或MPI I/O读写存储在HDF5 Dataset中的数据，这允许多个进程访问一个HDF5文件中相同的HDF5 Dataset，或相同文件中多个数据集。应用程序也可写出分割的HDF5文件，然后依赖其他软件整合这些文件，或使用HDF5的外部Link或文件挂载整合数据，用户后处理（如可视化）。

每个进程写出一个文件，或n个进程写出m个文件，这对HPC应用很有吸引力，特别是需要checkpointing data dump，且每个时间步管理大量文件不是问题的场景，如PnetCDF和ADIOS实施了写出分离的subfile的功能。

HDF5 1.10.0引入Virtual Dataset (VDS)的概念，允许在分割的HDF5文件（称为source files）和数据集（称为source datasets）存储分片的数据集，但从master文件的角度视为一个单个HDF5数据集，如图2。

目前，只能串行访问VDS数据，VDS可以使用h5repack命令工具再封装到一个HDF5数据集（传统的连续或chunked存储）。



## 3 实施Subfiling

通过分组并行任务，产生比file-per-process更少数目的文件，改善I/O效率，且可维护管理系统上的文件数。

图3显示了进程分组使用subfiling写出到一个单个共享的数据集。



3.1 HDF5 Subfiling编程模型

1）使用开启subfiling功能创建一个HDF5文件

2）使用开启subfiling功能创建一个HDF5数据集

3）Writing

4）Reading

## 4 试验设置

## 5 结果

5.2 调试subfiling

1）Number of subfiles

2）存储布局(Storage layout)

optimal number of Lustre stripe configuration for subfiles，测试从4K进程写出32个subfiles

## 6结论

6.1建议

6.2局限

## 参考文献

Zheng Huihuo et al. HDF5 Cache VOL: Efficient and Scalable Parallel I/O through Caching Data on Node-local Storage.