# John Ravi, et al. GPU Direct I/O with HDF5

## 1前言

GPUIDirect Storage (GDS)避免CPU内存作为中间缓冲，使用GPU内存与存储器之间直连路径。

本文将GDS引入HDF5，可以在HDF5中利用GDS。为此，开发了一个原型的HDF5 Virtual File Driver (VFD)，应用程序可使用HDF5利用GDS，无需使用GDS API。

## 2背景

2.1 GDS

2.2 HDF5: Virtual File Driver

如图2，HDF5提供丰富的API来描述各种数据模型以及组织数据对象并将其表述为元数据。HDF5包含各种组件来管理内存、转换数据类型、将数据存储为chunk，I/O filter（如压缩与解压）等。HDF5的VFD层在HDF5地址空间与存储之间实施映射()。应用程序可指定使用何种VFD，可使用HDF5 API或通过设置环境变量HDF5\_DRIVER。

默认使用SEC2 VFD提供POSIX文件系统函数调用，如read and write，实施I/O到单个文件。HPC应用常用的VFD是MPI和MPI-IO。还有DIRECT VFD，通过使用O\_DIRECT标识，强制数据直接写到文件系统，无需拷贝到系统核缓存区。

本文利用GDS，创建HDF5应用的映射，通过POSIX-like系统调用提供直接GPU I/O，读写到一个文件。



## 3 设计与实施

3.1 与SEC2 VFD的区别

3.2 GDS VFD效率调优参数

## 4 试验评估与结果

比较使用默认的HDF5 SEC2 VFD, DIRECT VFD, GDS VFD。默认的SEC2 VFD选项，数据I/O包含CPU与GPU之间显式的数据转移（调用cudaMemcpy）。

NVIDIA DGX-2，安装有NVMe-based当地存储和Lustre文件系统。当地存储配置RAID 0，带有2个NVMe驱动，理论最大串行写带宽1.8GB/s。系统上配置Lustre使用progressive文件布局。

GDS处于beta测试阶段，后面的效率会有不同。

4.1当地文件系统上的表现

4.2使用多线程的Lustre上的表现

使用多CPU I/O线程，将I/O请求分割为更小的块。使用GDS VFD调优这些参数。

使用单个GPU，写到Lustre文件系统。

4.3使用多个GPUs的表现

利用MPI，每个MPI rank写到一个单独的文件。

试验：各MPI rank操作2 GB数据和4 MB blocksize。



## 5 下一步工作

1. Per System Tuning；
2. 异步I/O：利用CUDA Stream，重叠GDS与计算。扩展ASYNC VOL支持GDS异步操作。
3. 多线程POSIX I/O
4. 并行I/O：实施直接存储GPU数据的并行I/O。包括：使用背景线程转移CPU与GPU之间的数据，使用启用GDS的MPI-IO，异步执行I/O。