Parallel IO

https://github.com/ResearchComputing/USGS_2016_02_09-10/

February 10, 2016

Timothy Brown

Research Computing

UNIVERSITY OF COLORADO **BOULDER**

Overview

Lustre

MPI IO

HDF5

Example

Overview

Lustre

MPI IO

HDF5

Example

What is Lustre

Lustre is a parallel distributed file system, used mostly for large scale clusters.

Why?

- Spinning disks are slow.
- Serial I/O is even slower.

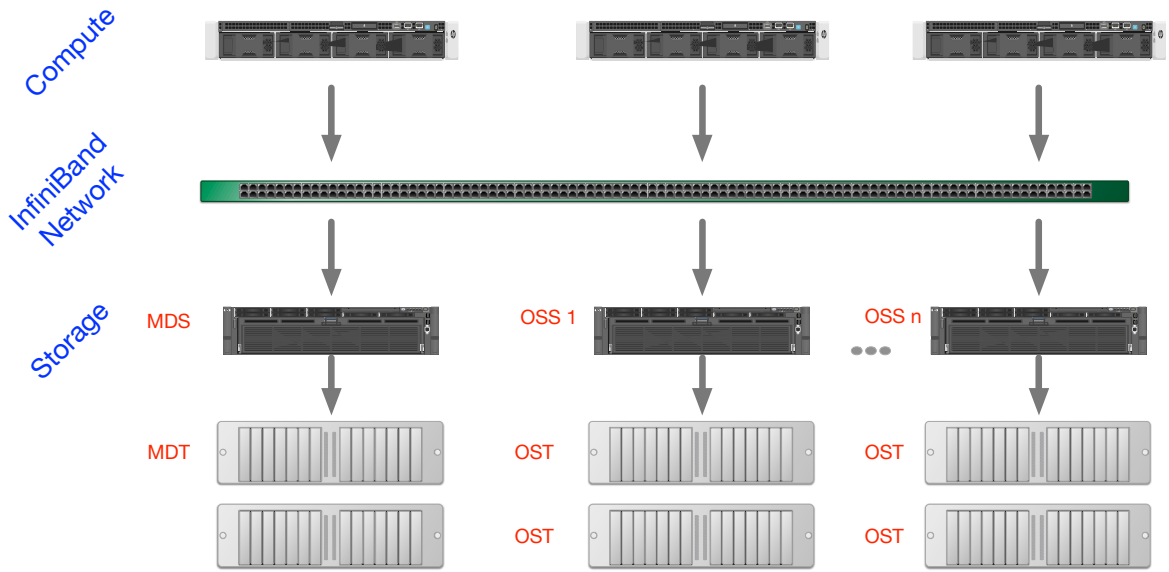
Key Features

- Scalability.
  Can scale out to tens of thousands of nodes and petabytes of storage.
- Performance.
  Throughput of a single stream ~GB/s and parallel I/O ~TB/s.
- High availability.
- POSIX compliance.

Lustre Components

It consists of four components:

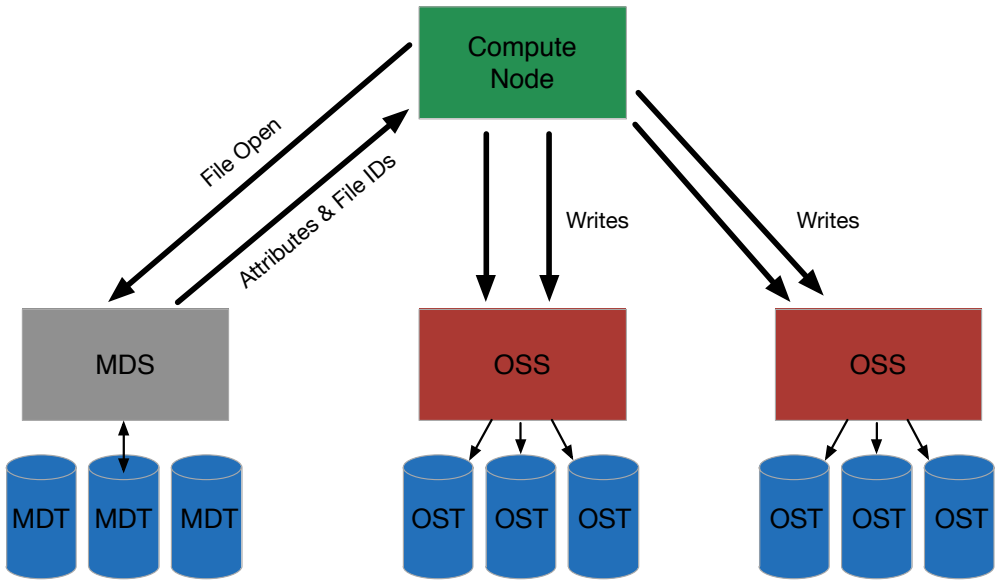
- MDS Metadata Server
- MDT Metadata Target
- OSS Object Storage Server
- OST Object Storage Target



File Operations

- When a compute node needs to create or access a file, it requests the associated storage locations from the MDS and the associated MDT.

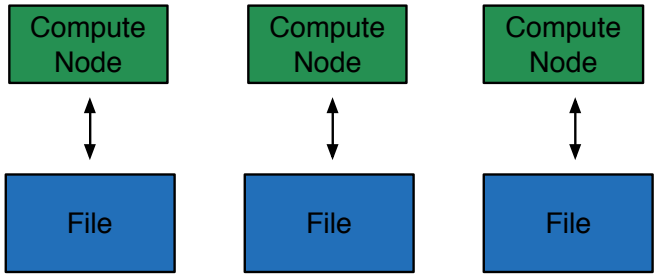
- I/O operations then occur directly with the OSSs and OSTs associated with the file bypassing the MDS.
- For read operations, file data flows from the OSTs to the compute node.



File I/O

Three cases of file I/O:

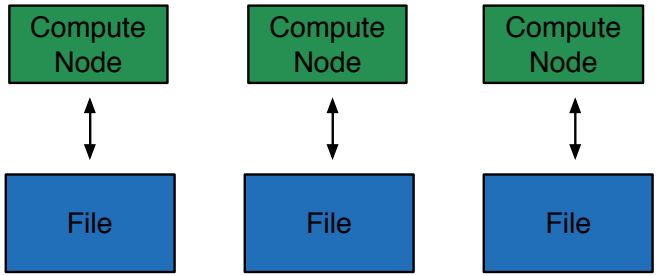
- Single stream.



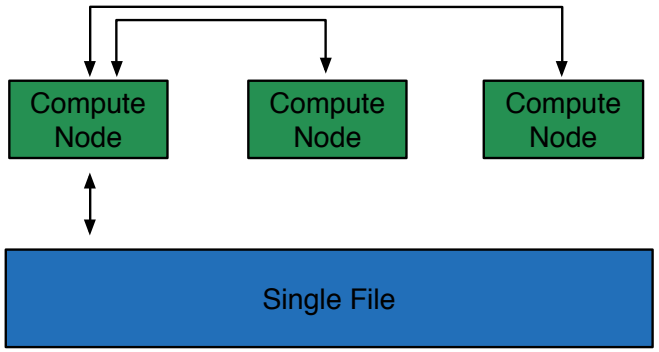
File I/O

Three cases of file I/O:

- Single stream.



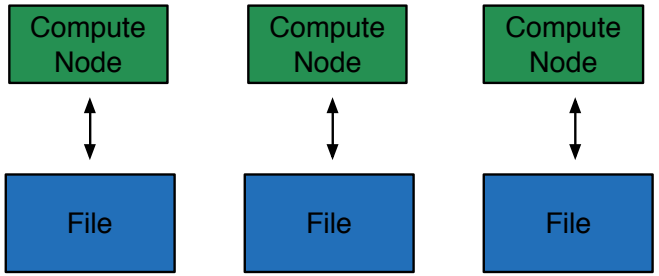
- Single stream through a master.



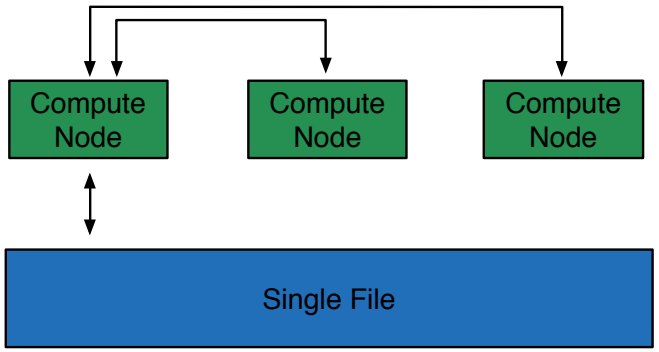
File I/O

Three cases of file I/O:

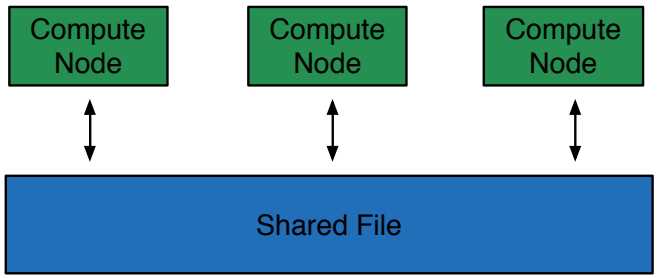
- Single stream.



- Single stream through a master.

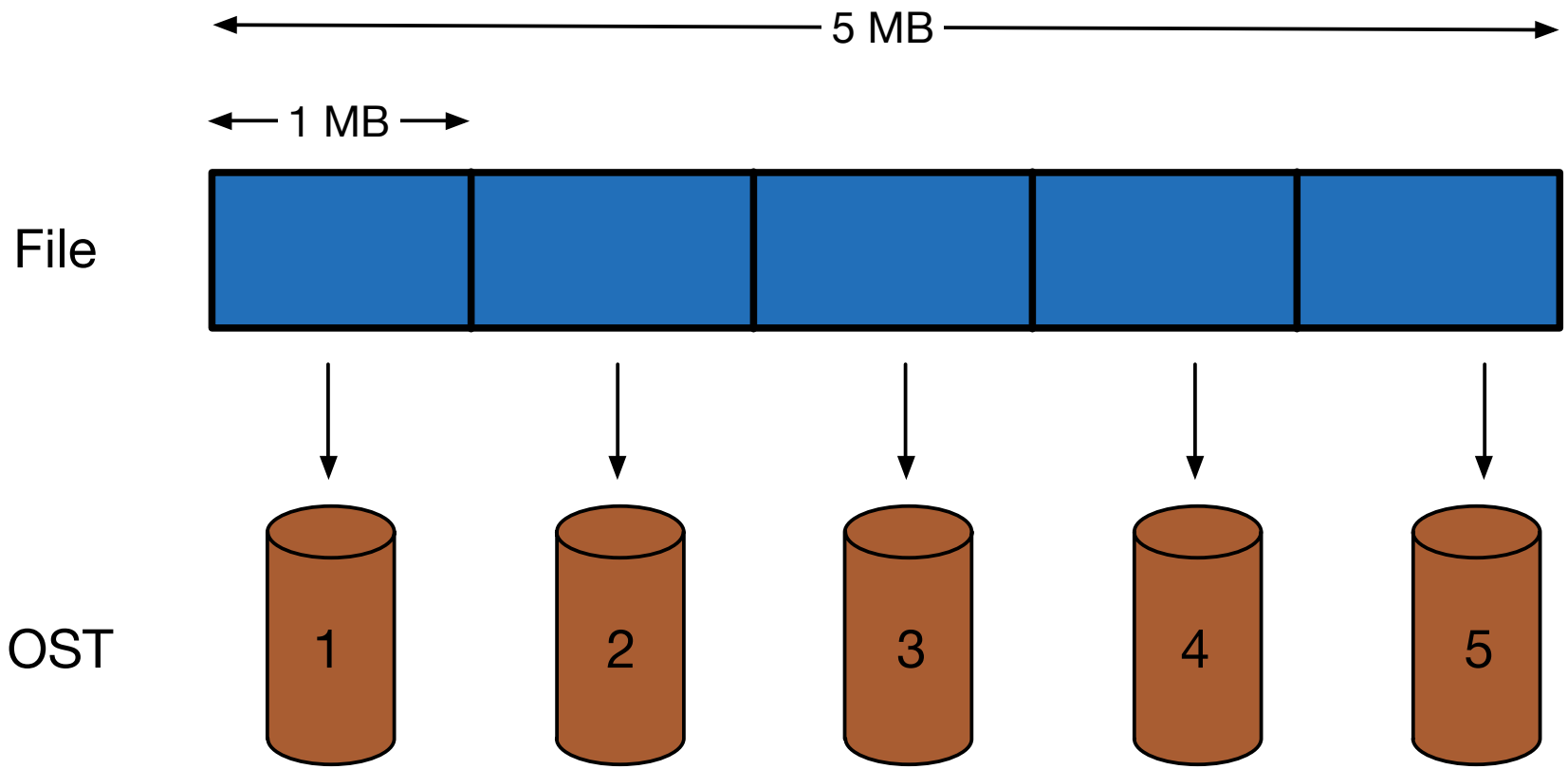


- Parallel.



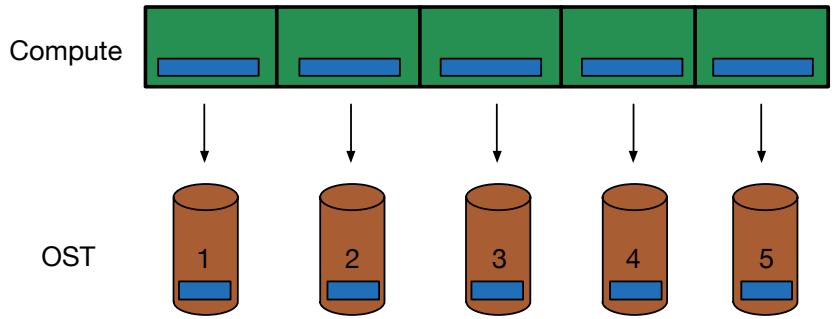
File Striping

- A file is split into segments and consecutive segments are stored on different physical storage devices (OSTs).

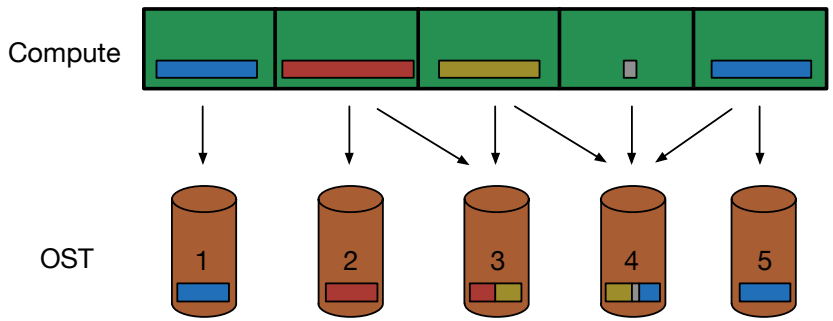


Aligned vs Unaligned Stripes

- Aligned stripes is where each segment fits fully onto a single OST. Processes accessing the file do so at corresponding stripe boundaries.



- Unaligned stripes means some file segments are split across OSTs.



Overview

Lustre

MPI IO

HDF5

Example

MPI IO

- MPI IO was added to the standard in version 2 (~1996).
- IO calls look very similar to the rest of the MPI calls.
- Ability to read and write files in
  - Blocking and non-blocking modes.
  - Independent and collective modes.

- Open a file.

```
MPI_File_open(comm, filename, amode, info, fh, ierr)
```

- Changes process's view of data in a file

```
MPI_File_set_view(fh, disp, etype, filetype, datarep, &
                  info, ierr)
```

- Read data from a file

```
MPI_File_read_at(fh, offset, buf, count, datatype, &
                 status, ierr)
```

- Close a file

```
MPI_File_close_at(fh, ierr)
```

Dangers of MPI IO

The file is raw binary.

- Endian dependent
- Lacks meta data

Which means you have to remember how it was created, what was written.

Good alternatives are NetCDF and HDF.

Overview

Lustre

MPI IO

HDF5

Example

HDF5

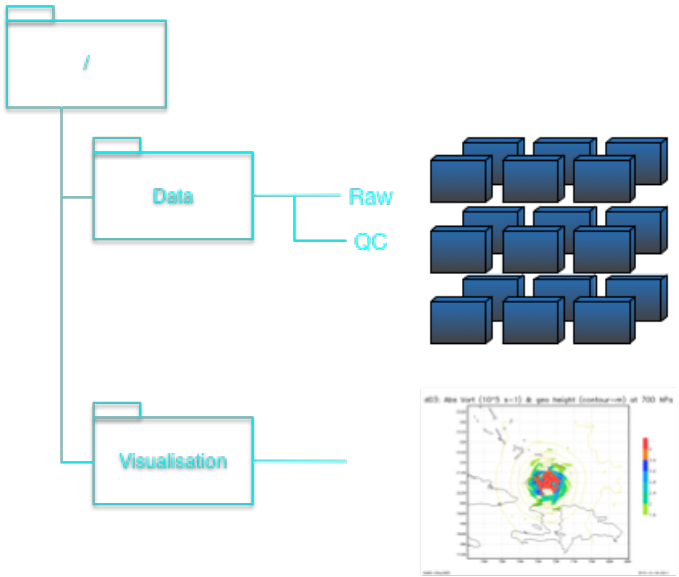
Hierarchical **D**ata **F**ormat version 5 (HDF5).

- Designed for scientific, high volume data.
- Is a file format to manage data.
  - multidimensional arrays
  - tables
  - compounded structures
  - images
- Software library and tools that provide access to manage data in these files.
- Gives the developer access to manipulate groups and datasets rather than binary streams.

HDF5 Data Model

A HDF5 file is a container that can have groups, links and datasets.

- File - a contiguous string of bytes in a computer store (memory, disk, etc.), and the bytes represent zero or more objects of the model.
- Group - a collection of objects (including groups).
- Dataset - a multi-dimensional array of data elements with attributes.
- Dataspace - a description of the dimensions of the dataset.
- Datatype - a description of a specific class of data element including its storage layout.



HDF5 Data Model

- Attribute - a named data value associated with a group, dataset, or named datatype.
- Property List - a collection of parameters (some permanent and some transient) controlling options in the library.
- Link - the way objects are connected.

HDF5 Datasets

HDF5 Datasets organize and contain your data. They consist of:

- Metadata
  - datatype (real, integer, …)
  - layout (rank, rows, columns)
  - properties (units)
- Data

```
HDF5 "MIELLAJOKKA.h5" {
GROUP "/" {
   GROUP "010708-MIELLANJOKKA-1-3D" {
      DATASET "Emission" {
         DATATYPE  H5T_IEEE_F64LE
         DATASPACE  SIMPLE { ( 636 ) / ( 636 ) }
         DATA {
         (0): 240, 240.5, 241, 241.5, 242, 242.5, 243, 2
         ...
         (630): 555, 555.5, 556, 556.5, 557, 557.5
         }
         ATTRIBUTE "Units" {
            DATATYPE  H5T_STRING {
               STRSIZE 2;
               STRPAD H5T_STR_NULLTERM;
               CSET H5T_CSET_ASCII;
               CTYPE H5T_C_S1;
            }
            DATASPACE  SCALAR
            DATA {
            (0): "nm"
            }
         }
      }
```

Virtual File Layers

HDF5 provides a virtual file layer which you can extend.

- POSIX
- STDIO
- MPI-IO

You do not need to be an MPI expert to use the parallel IO layer in HDF5.

HDF5 IO Sequence

Very similar to normal IO sequence, only a few additional items need to be specified.

- open/create a file
- specify the dataspace
- create the dataset
- write the data
- close the file

HDF5 Fortran API

The fortran API is the same as the C API, however subroutines have a `_f` suffix and the last parameter is the return status.

C	Fortran
ierr = H5open(void)	H5open_f(ierr)

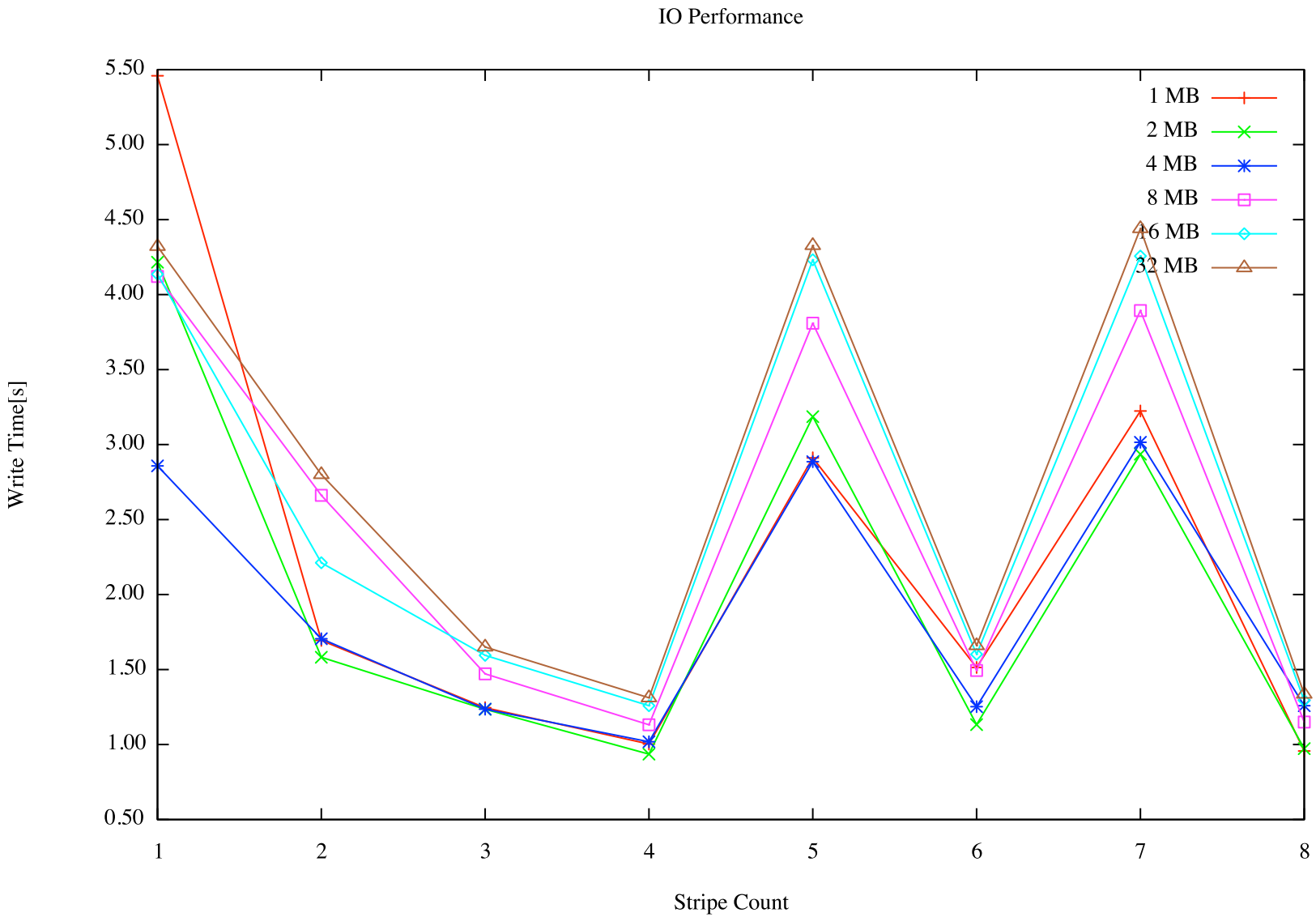
MPI IO Hints

You can set the Lustre stripe count and size using `MPI_Info`.

```
integer :: info
integer(kind=hid_t) :: p_id, f_id
character(len=32) :: lcount, lsize

write(lcount, '(I4)') 4
write(lsize,  '(I8)') 4 * 1024 *1024
call mpi_info_create(info, ierr)
call mpi_info_set(info, "striping_factor", lcount, ierr)
call mpi_info_set(info, "striping_unit", lsize, ierr)

call h5pcreate_f(H5P_FILE_ACCESS_F, p_id, ierr);
call h5pset_fapl_mpio_f(p_id, MPI_COMM_WORLD, info, ierr)
call h5fcreate_f(filename, H5F_ACC_TRUNC_F, f_id, ierr, &
                 access_prp = p_id)
```


IO Performance
Write Time[s]
Stripe Count
5.50
5.00
4.50
4.00
3.50
3.00
2.50
2.00
1.50
1.00
0.50
1
2
3
4
5
6
7
8
1 MB
2 MB
4 MB
8 MB
16 MB
32 MB

Overview

Lustre

MPI IO

HDF5

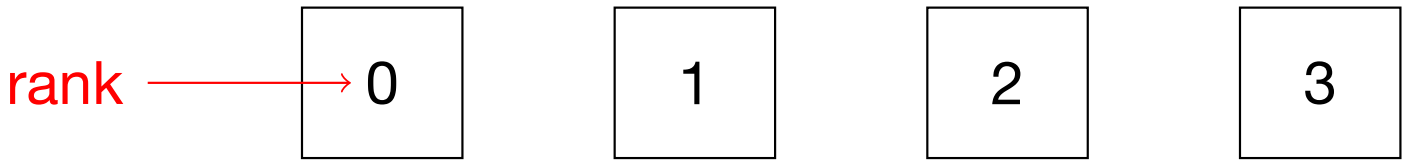
Example

Processor Domain

A 4 processor MPI job with a 2D gridded domain.

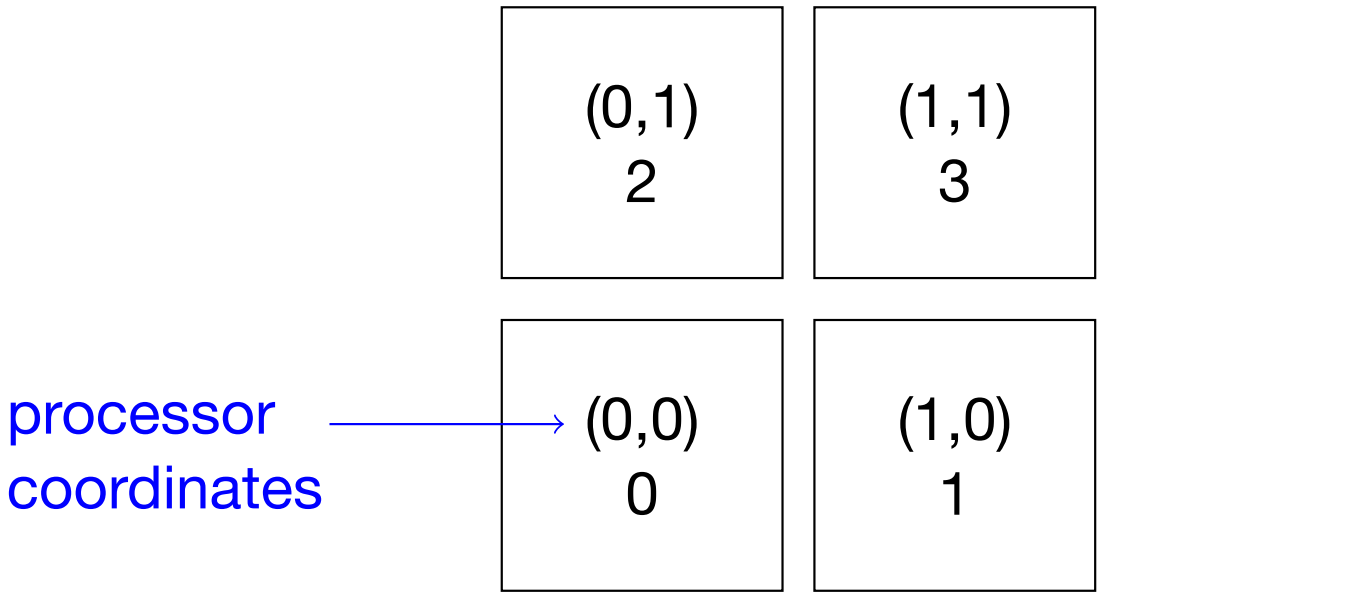
```
mpiexec -np 4 ./hdf_pwrite
```

```
call mpi_comm_size(MPI_COMM_WORLD, nprocs, ierr)
call mpi_comm_rank(MPI_COMM_WORLD, rank, ierr)
```

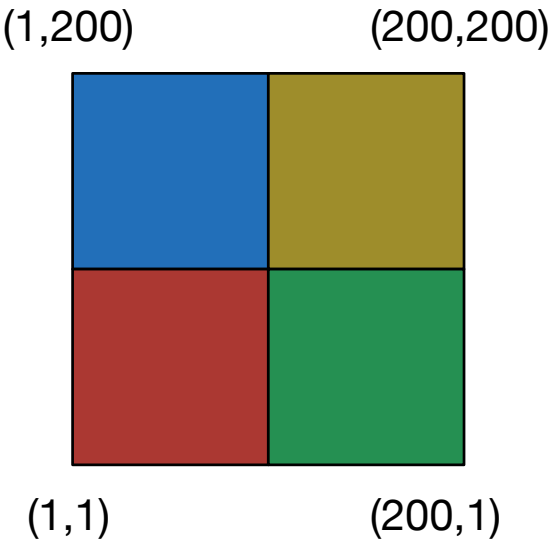


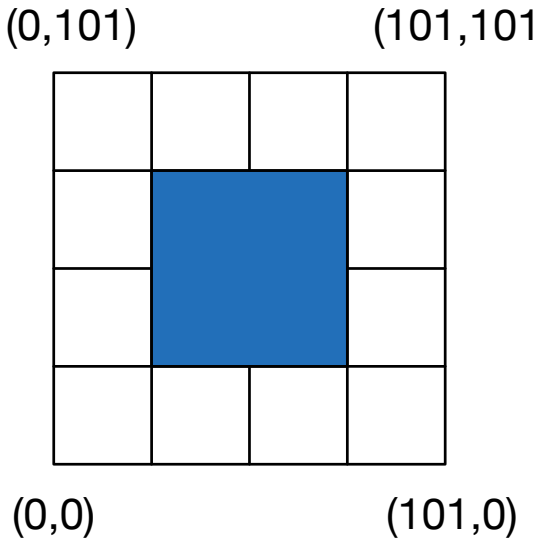
- Create a 2D domain.

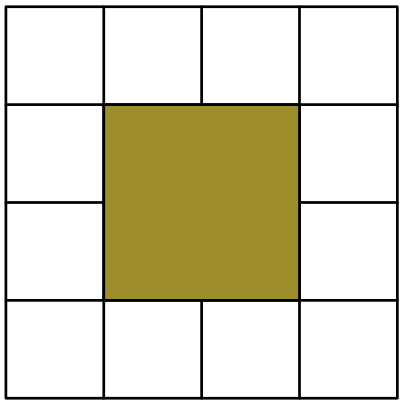
```
call mpi_dims_create(nprocs, 2, pdims, ierr)
call mpi_cart_create(MPI_COMM_WORLD, 2, pdims,      &
                     perodic, reorder, MPI_COMM_2D, &
                     ierr)
call mpi_cart_coords(MPI_COMM_2D, rank, 2, pcoords, ierr)
```

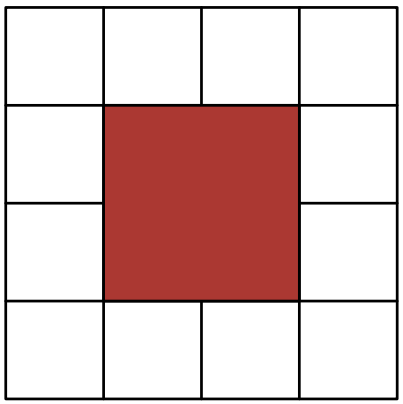


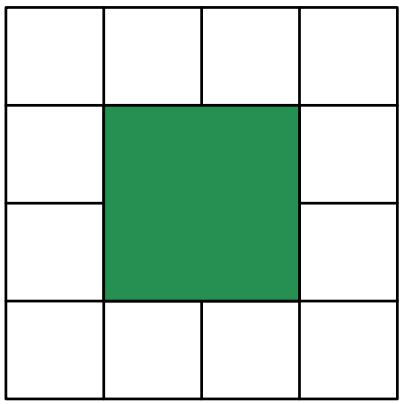
Local Grids



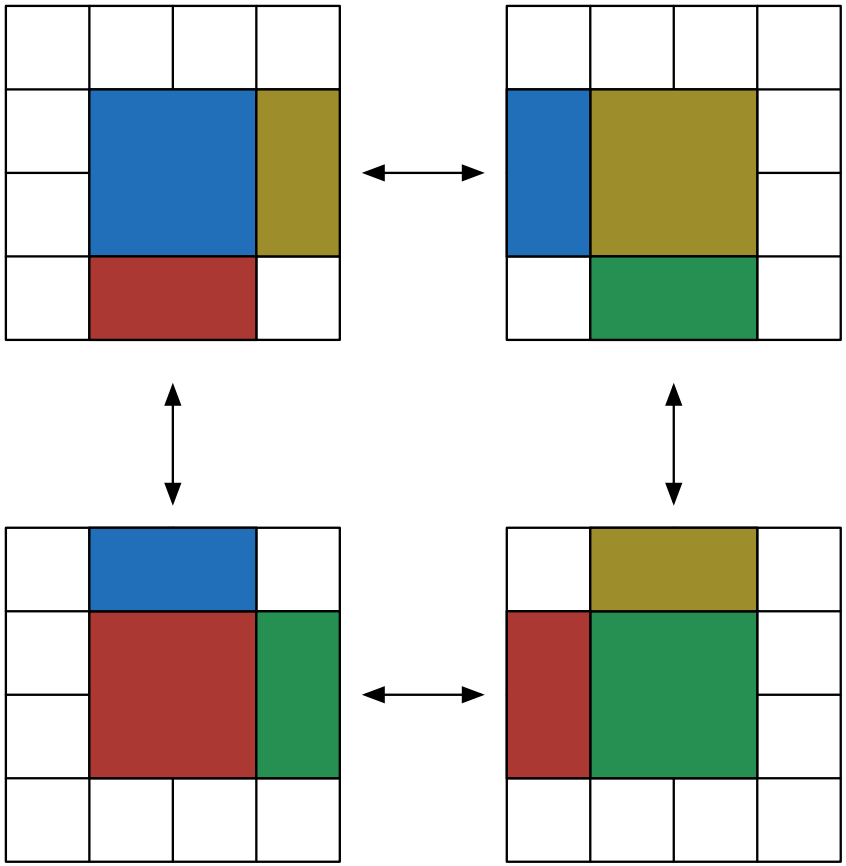








Halo Exchange



MPI 3 has neighbourhood collectives.

```
send(1:D,1) = grid(1,  1:D)
send(1:D,2) = grid(D,  1:D)
send(1:D,3) = grid(1:D,1)
send(1:D,4) = grid(1:D,D)

call MPI_Neighbor_alltoall(send, L, MPI_INTEGER, &
                           recv, L, MPI_INTEGER, &
                           MPI_COMM_2D, ierr)

grid(0,  1:D) = recv(1:D,1)
grid(D+1,1:D) = recv(1:D,2)
grid(1:D,0)   = recv(1:D,3)
grid(1:D,D+1) = recv(1:D,4)
```

Parallel IO with HDF5

There are a fair few steps involved.

- Create a hyperslab to represent the local grid in memory, without the halo elements.
- Create a hyperslab for the global grid on disk.
- Assign `H5FD_MPIO_COLLECTIVE` property to the dataset.

`USGS_2016_02_09-10/Parallel_IO/example/`

Questions?

Online Survey

<Timothy.Brown-1@colorado.edu>

License