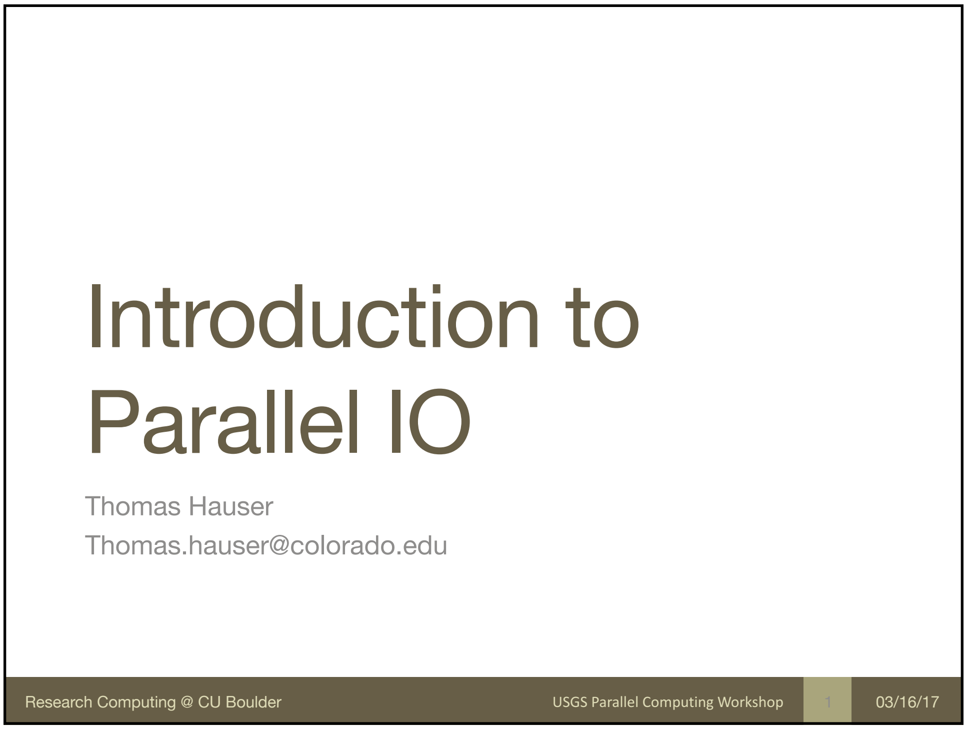
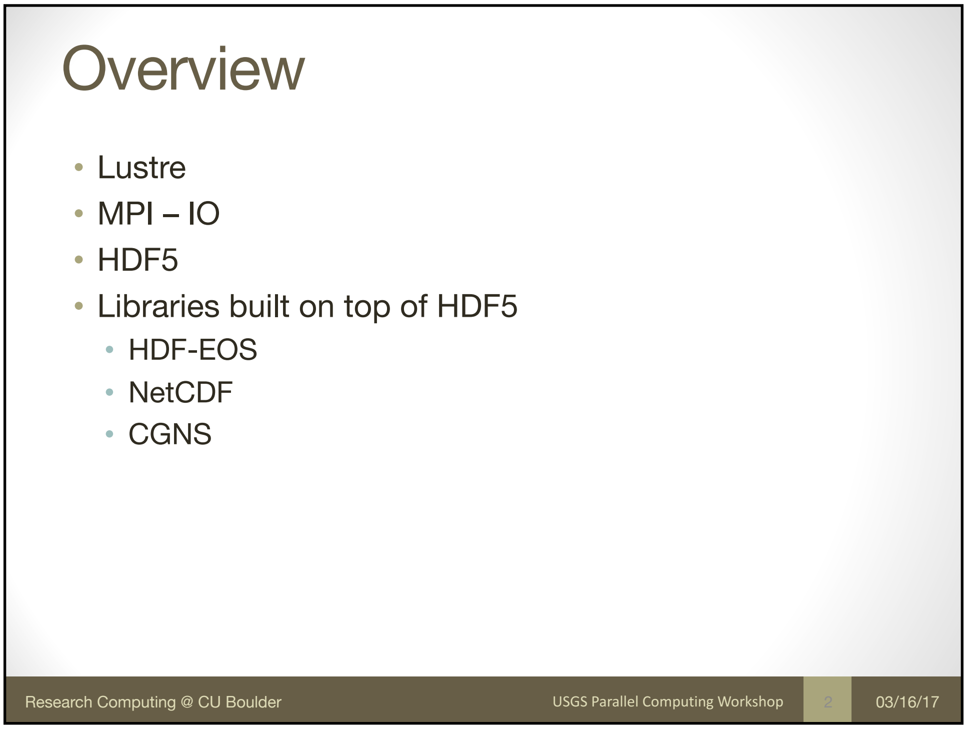


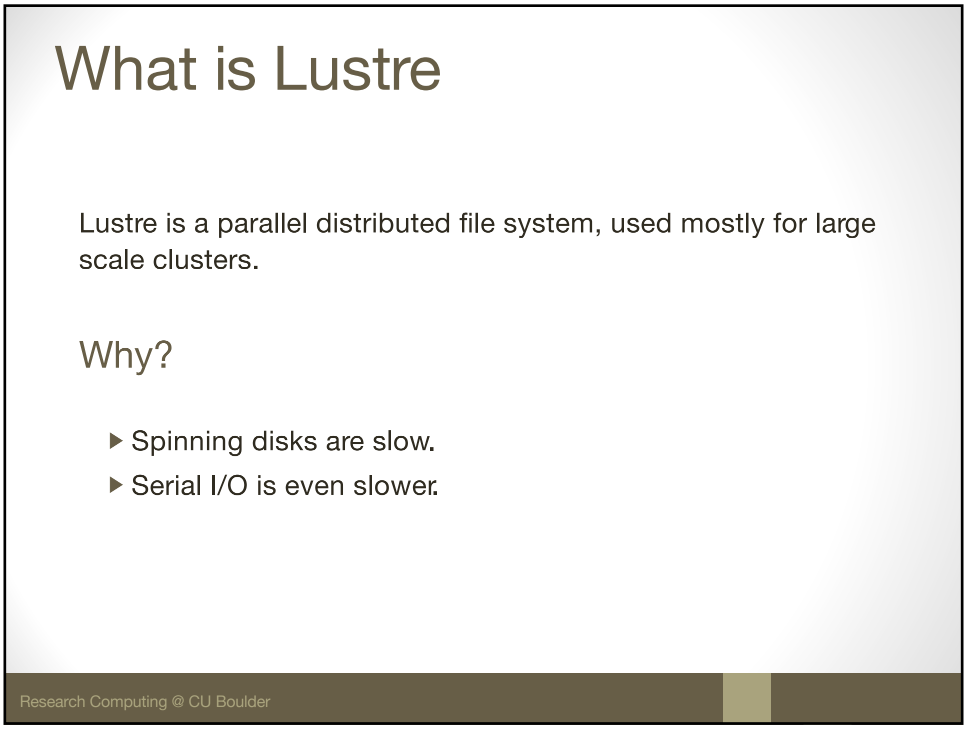
Introduction to Parallel IO

Thomas Hauser
Thomas.hauser@colorado.edu



Overview

- Lustre
- MPI – IO
- HDF5
- Libraries built on top of HDF5
  - HDF-EOS
  - NetCDF
  - CGNS



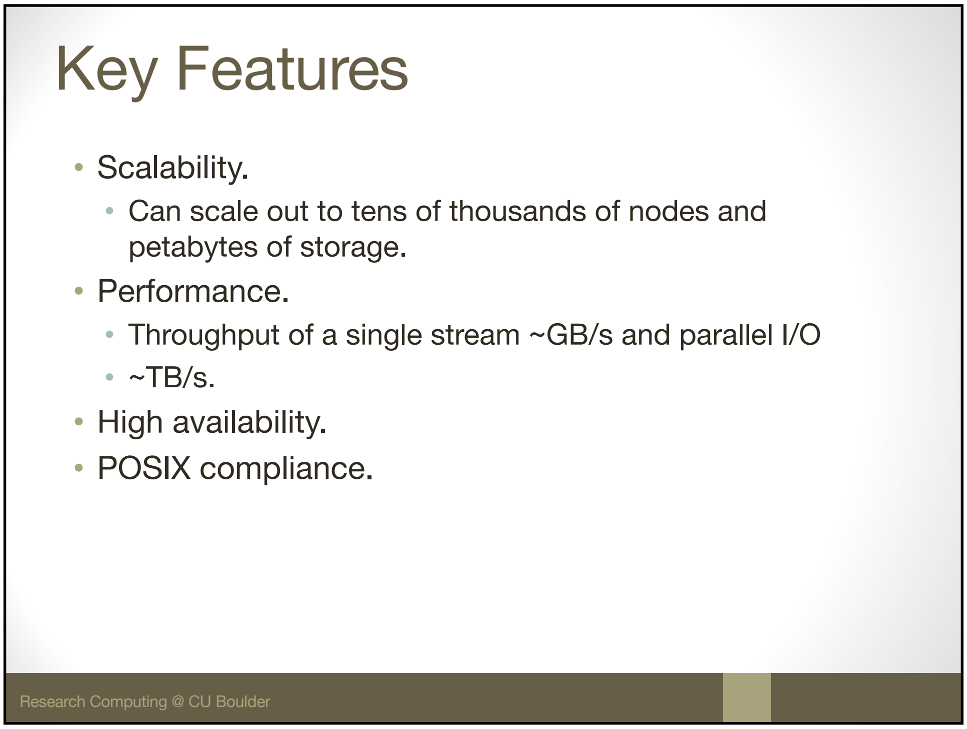
What is Lustre

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

Why?

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

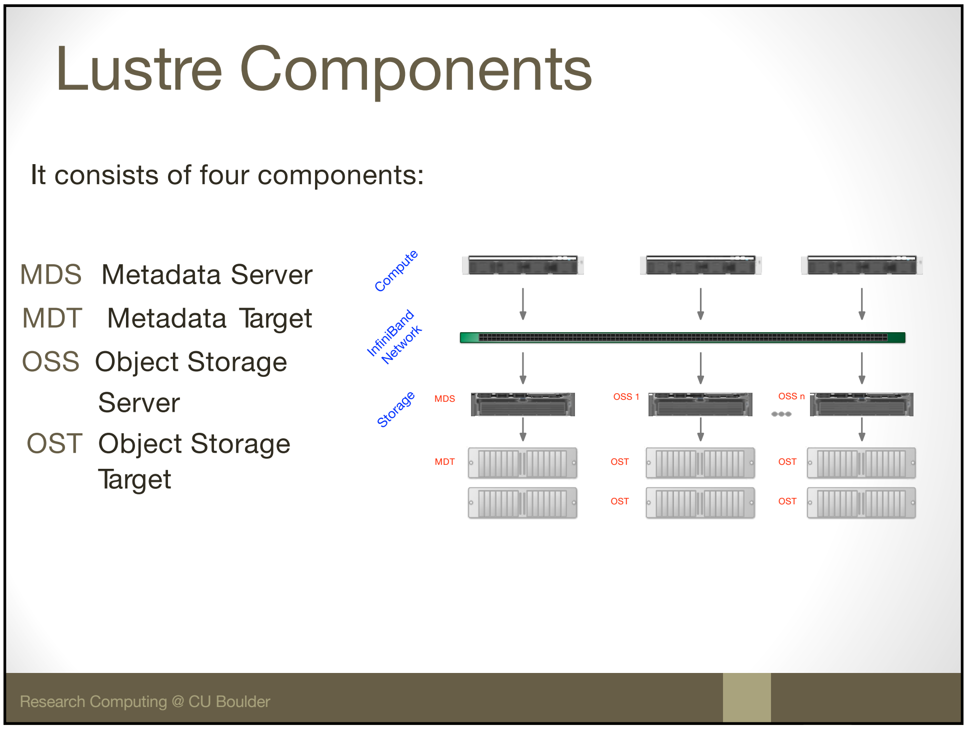
Research Computing @ CU Boulder



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.

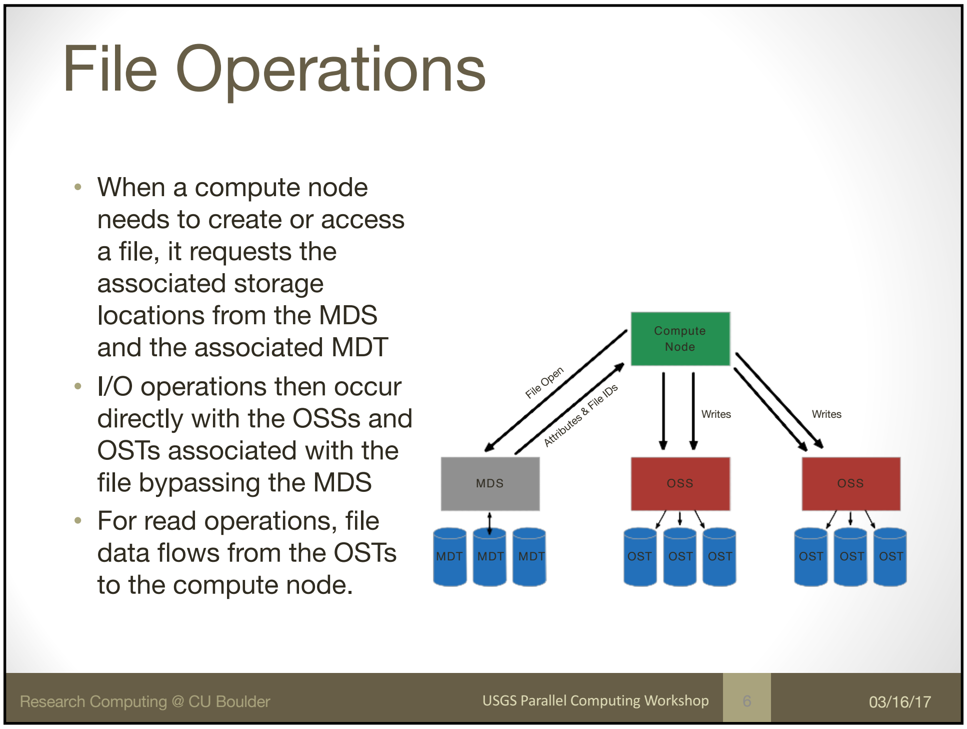
Research Computing @ CU Boulder



Lustre Components

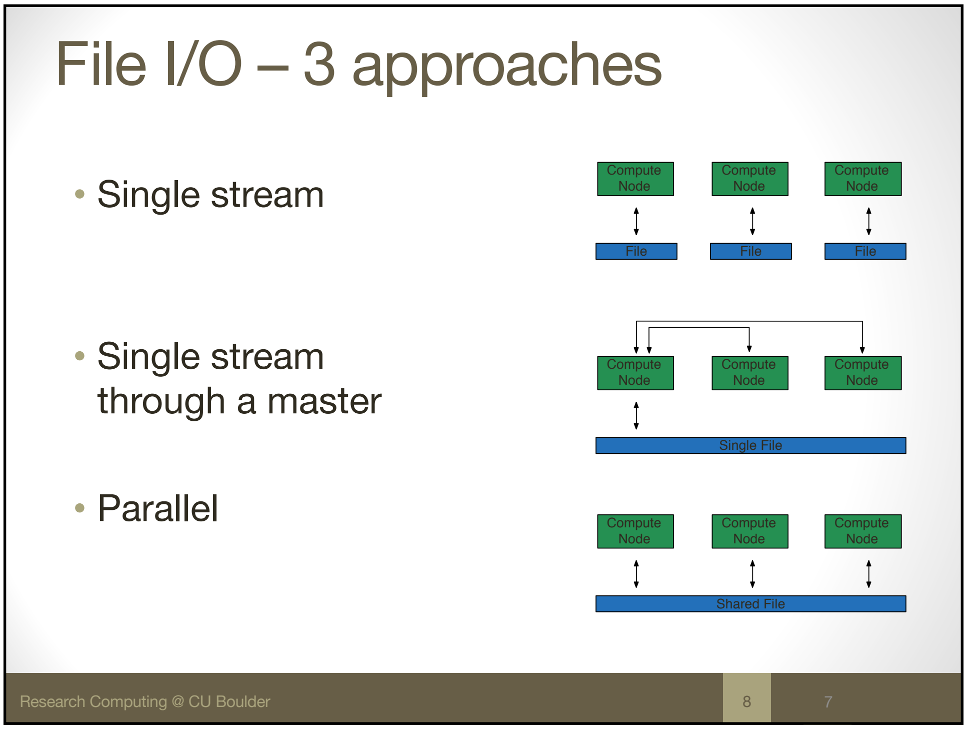
It consists of four components:

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



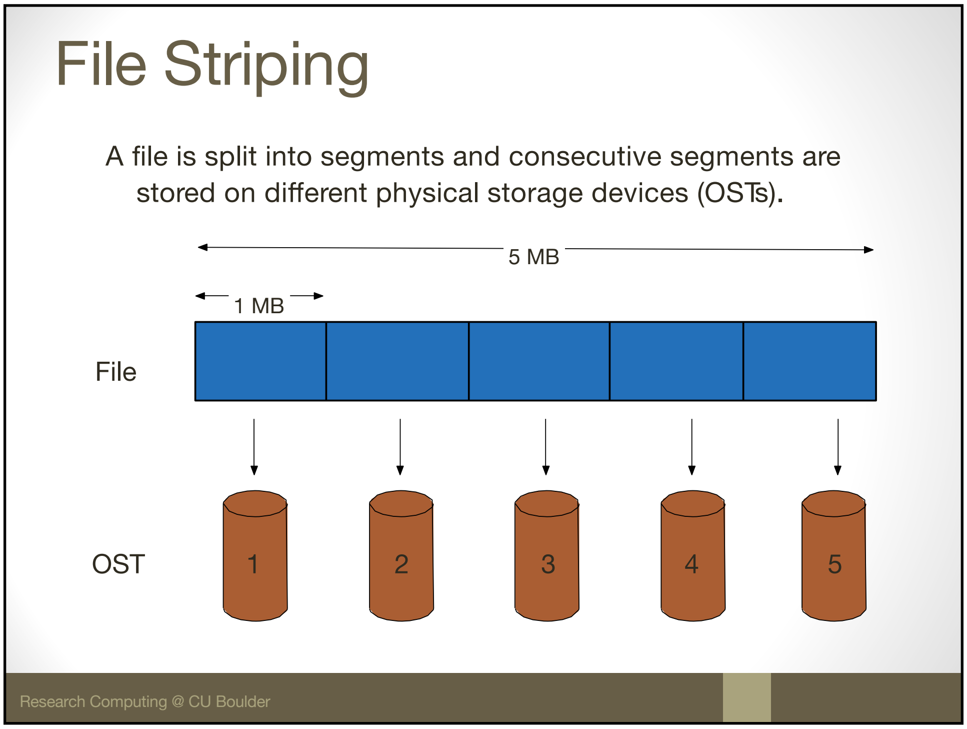
File Operations

- 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.



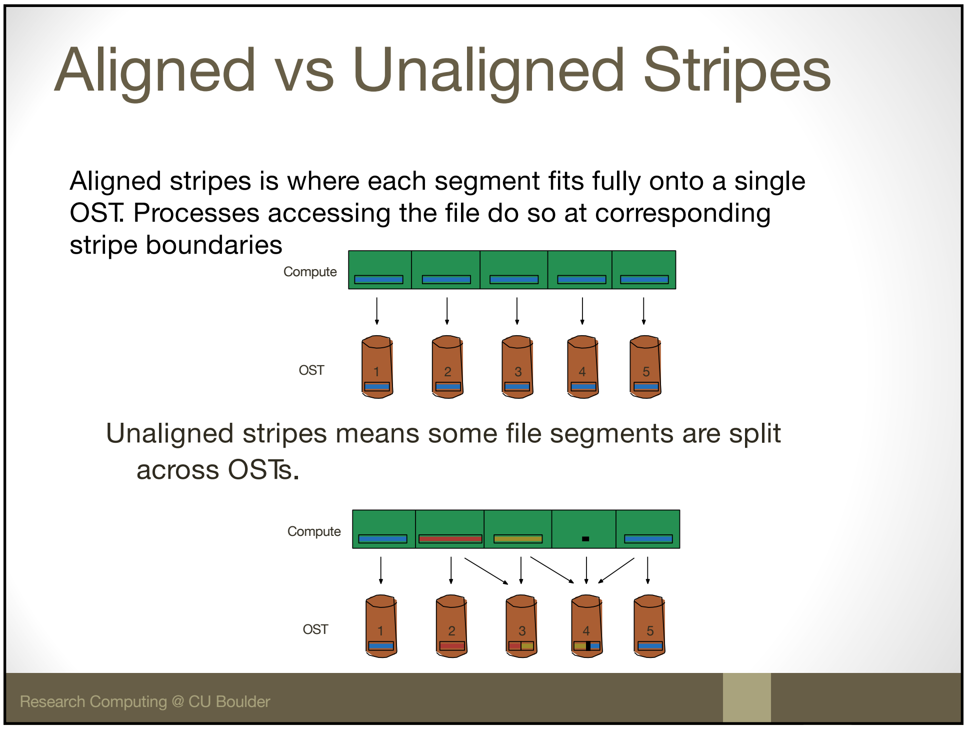
File I/O – 3 approaches

- Single stream
- Single stream through a master
- Parallel



File Striping

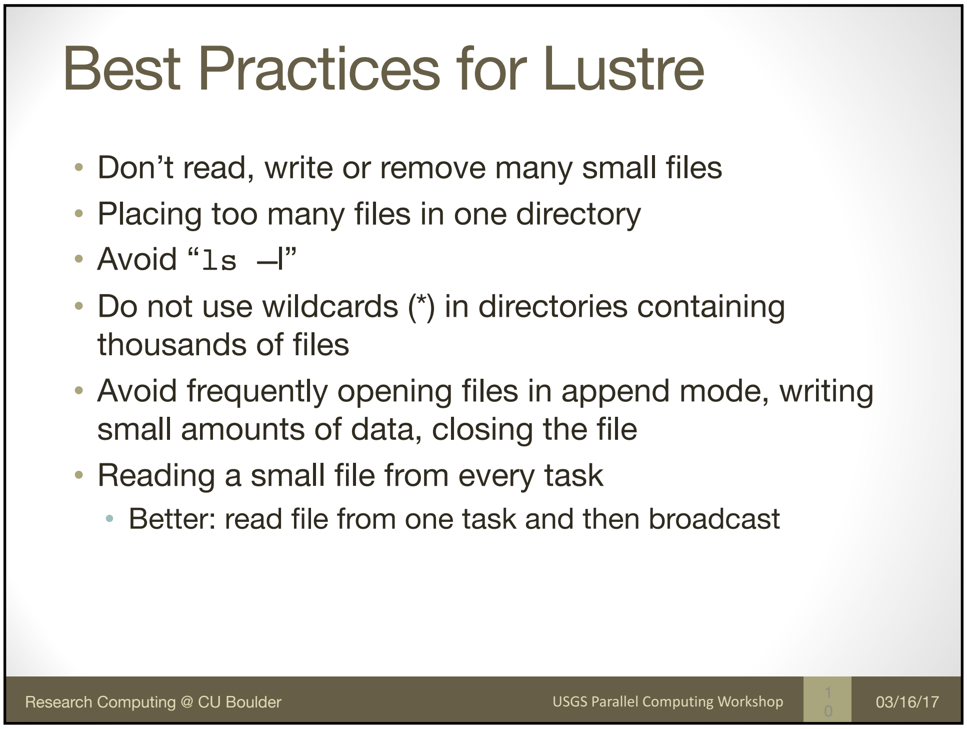
A file is split into segments and consecutive segments are stored on different physical storage devices (OSTs).



Aligned vs Unaligned Stripes

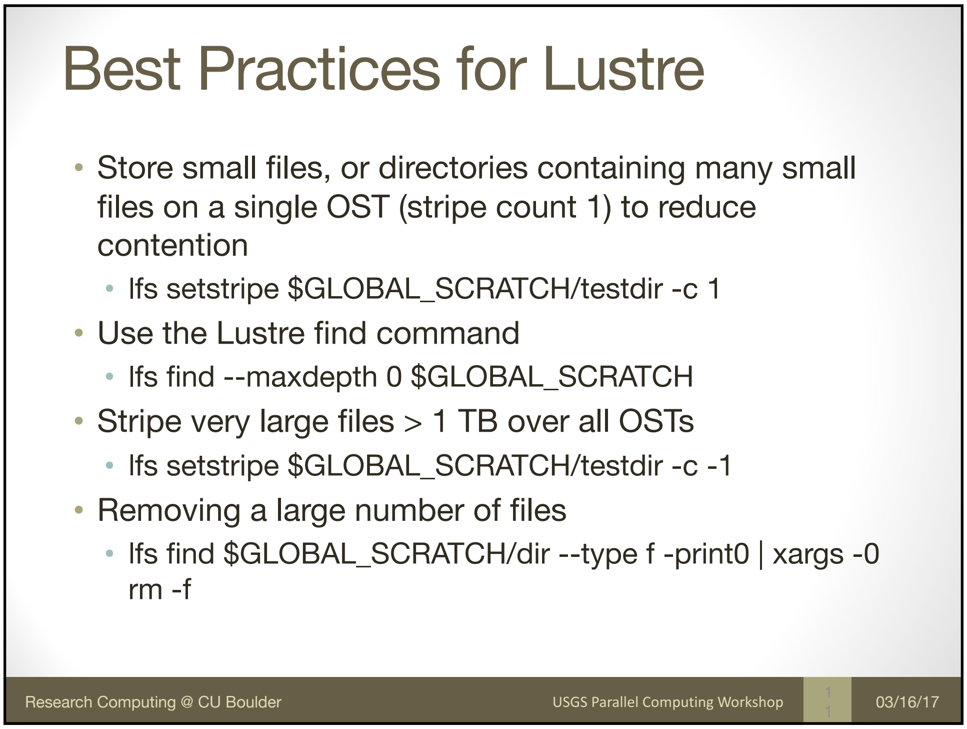
Aligned stripes is where each segment fits fully onto a single OST. Processes accessing the file do so at corresponding stripe boundaries

Unaligned stripes means some file segments are split across OSTs.



Best Practices for Lustre

- Don’t read, write or remove many small files
- Placing too many files in one directory
- Avoid “`ls –l`”
- Do not use wildcards (*) in directories containing thousands of files
- Avoid frequently opening files in append mode, writing small amounts of data, closing the file
- Reading a small file from every task
  - Better: read file from one task and then broadcast

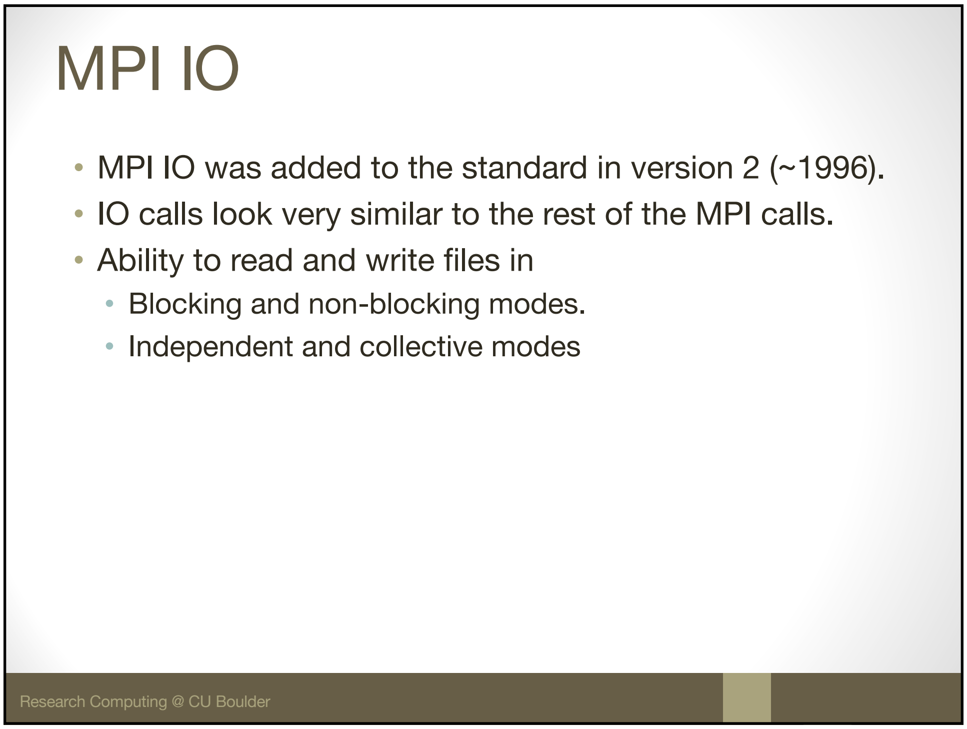


Best Practices for Lustre

- Store small files, or directories containing many small files on a single OST (stripe count 1) to reduce contention
  - lfs setstripe $GLOBAL_SCRATCH/testdir -c 1
- Use the Lustre find command
  - lfs find --maxdepth 0 $GLOBAL_SCRATCH
- Stripe very large files > 1 TB over all OSTs
  - lfs setstripe $GLOBAL_SCRATCH/testdir -c -1
- Removing a large number of files
  - lfs find $GLOBAL_SCRATCH/dir --type f -print0 | xargs -0 rm -f

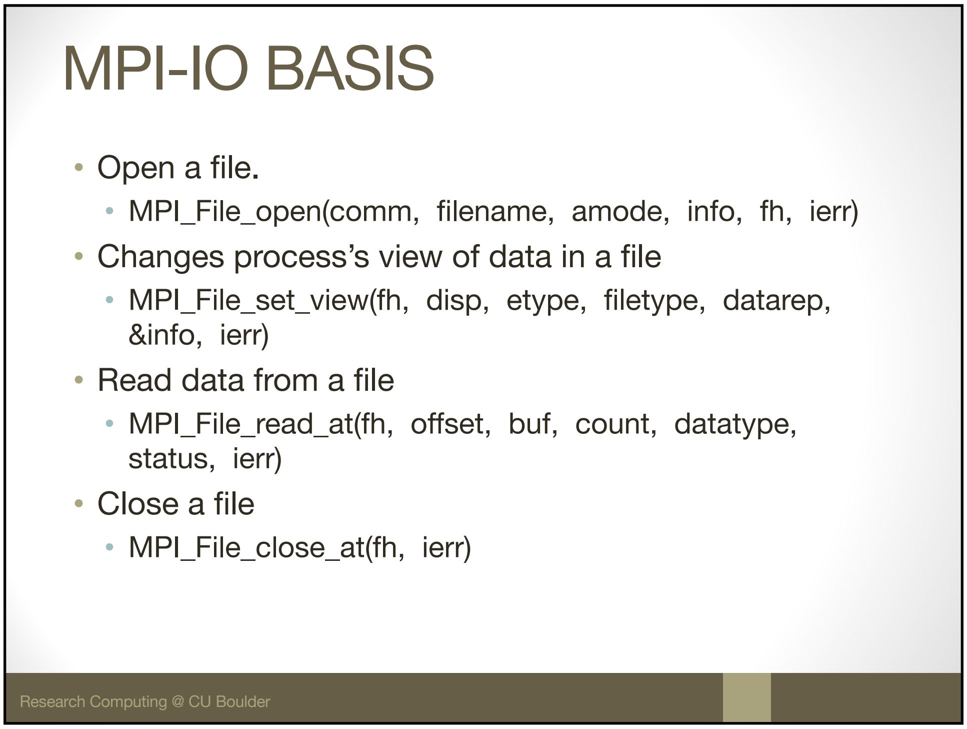
Overview

- Lustre
- MPI – IO
- HDF5



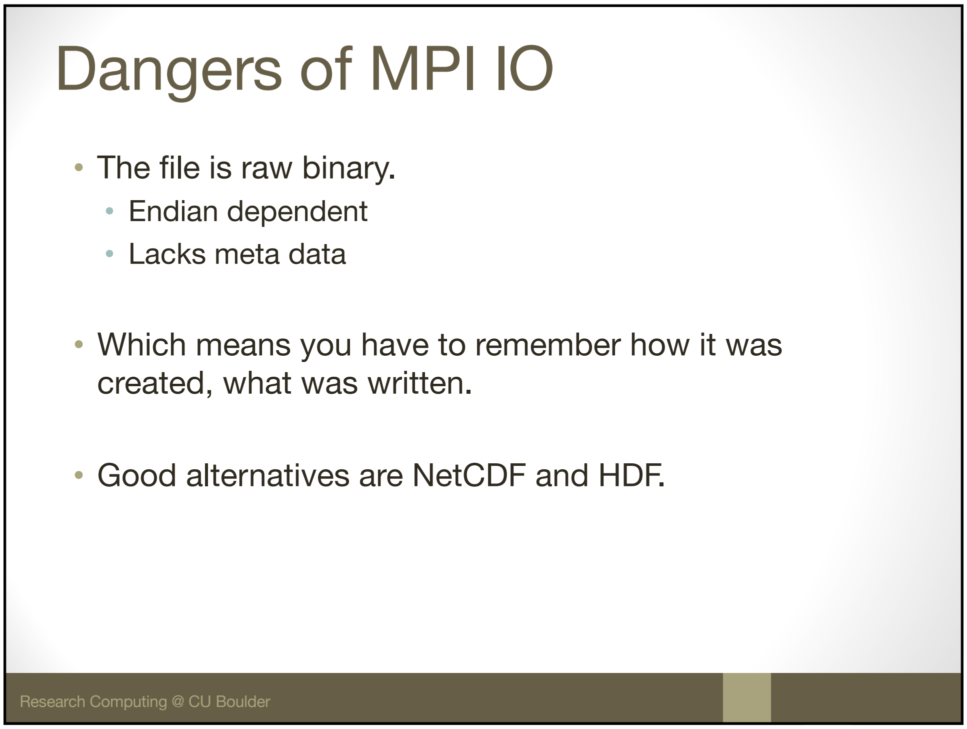
MPI IO

- 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



MPI-IO BASIS

- 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)

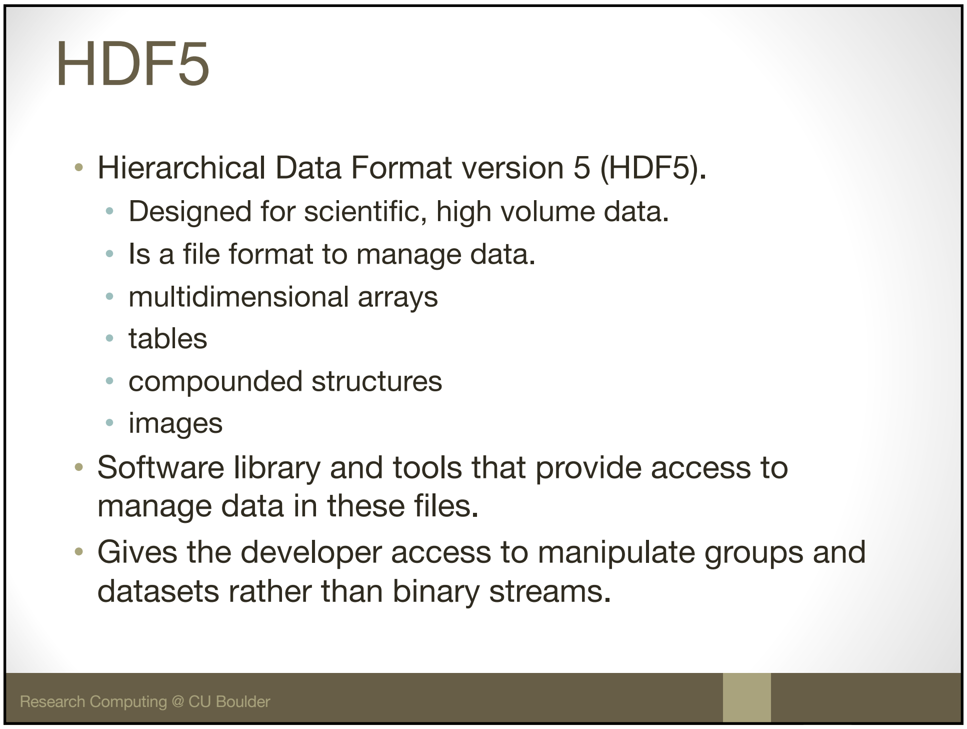


Dangers of MPI IO

- 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.

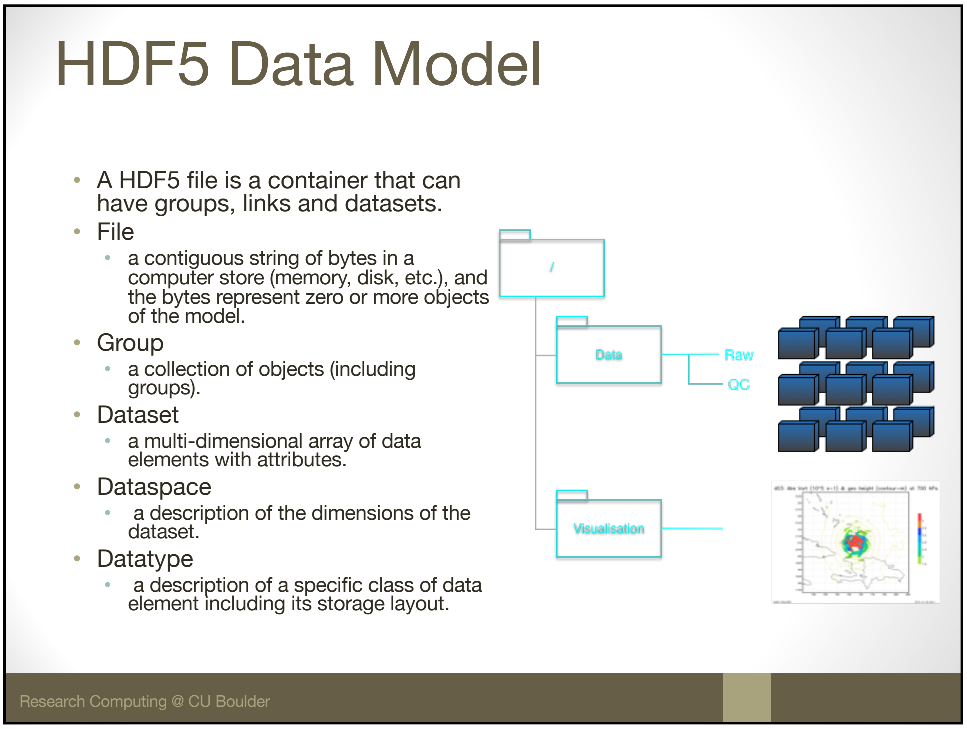
Overview

- Lustre
- MPI – IO
- HDF5



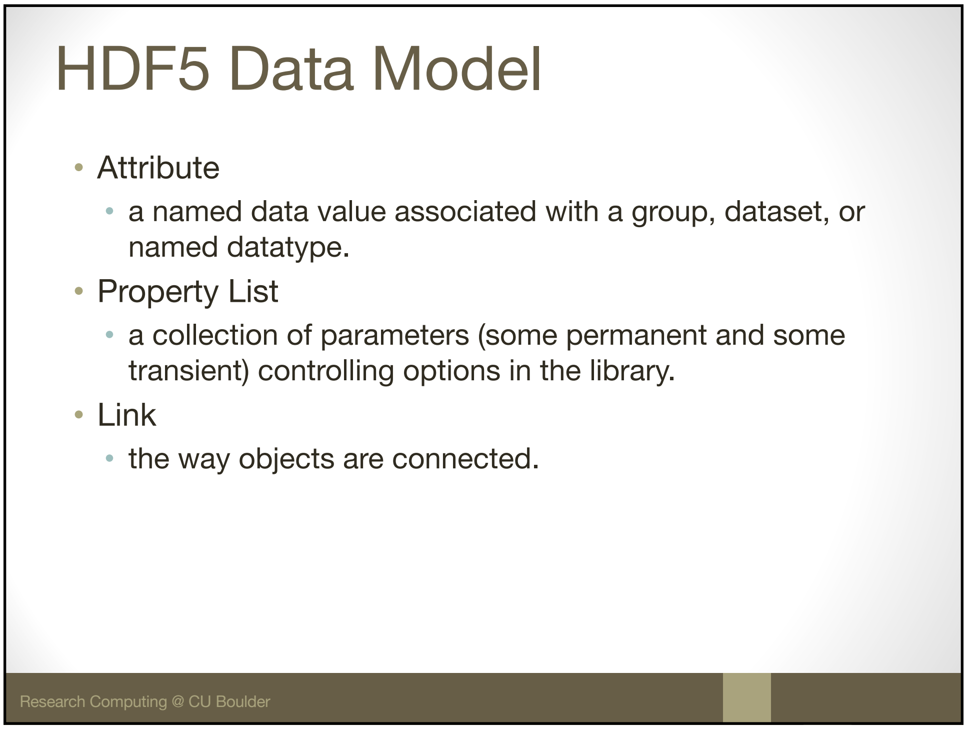
HDF5

- Hierarchical Data Format 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.



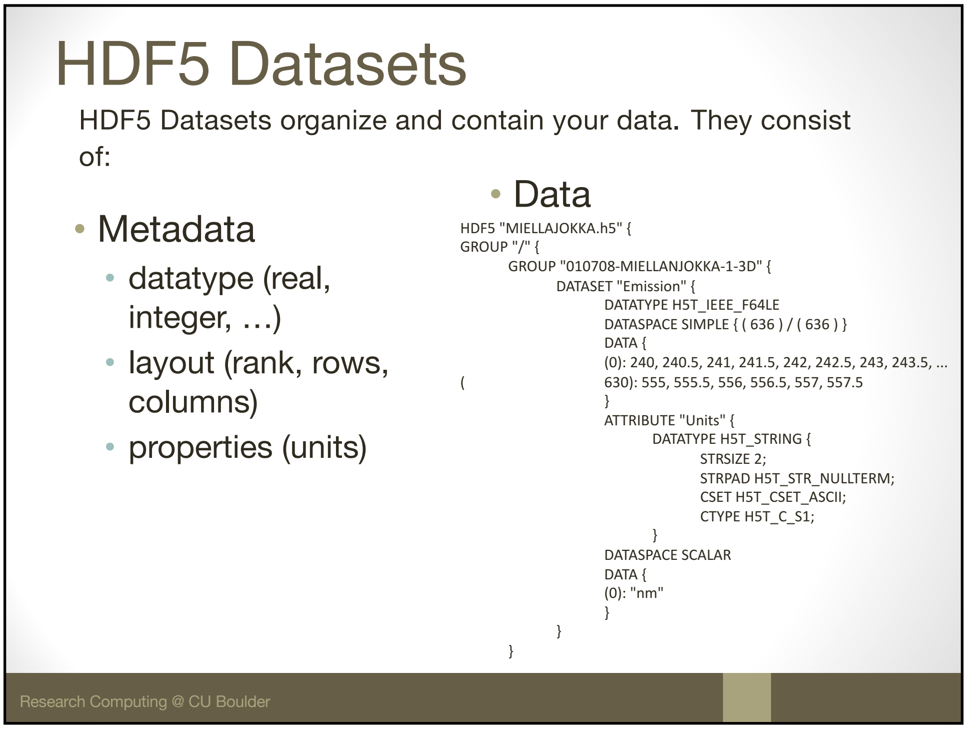
HDF5 Data Model

- 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

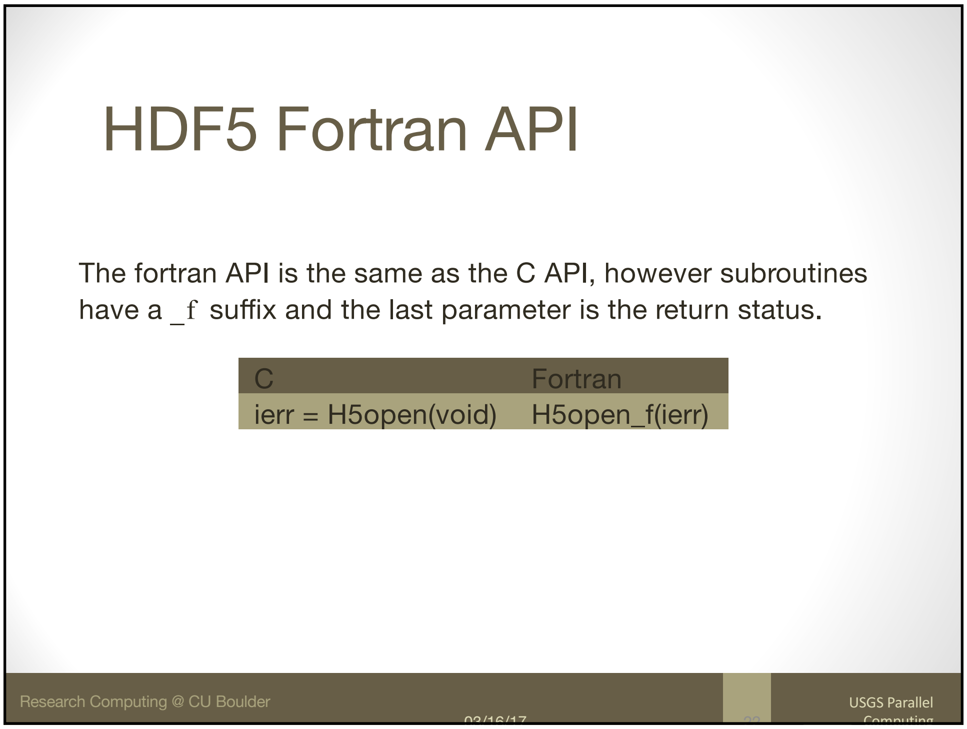
```
HDF5 "MIELLAJOKKA.h5" {
GROUP "/" {
        GROUP "010708-MIELLANJOKKA-1-3D" {
                DATASET "Emission" {
                        DATATYPE H5T_IEEE_F64LE
                        DATASPACE SIMPLE { ( 636 ) / ( 636 ) }
                        DATA {
                        (0): 240, 240.5, 241, 241.5, 242, 242.5, 243, 243.5, ...
(                       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

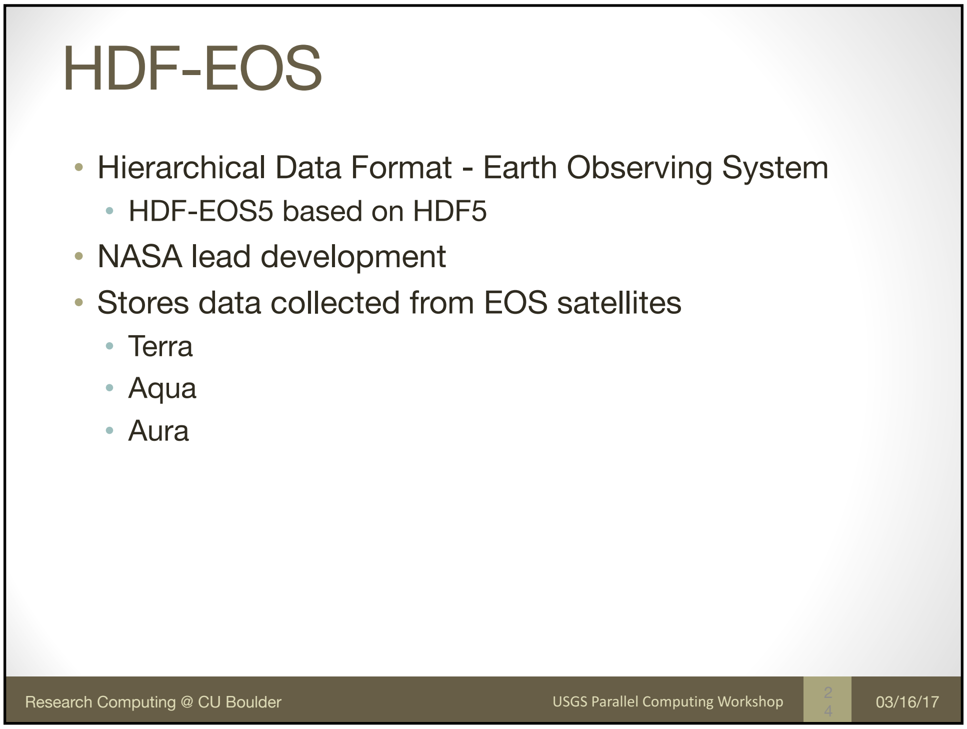
- Very similar to normal IO sequence, only a few additional items need to be specified.
  - open/create a file
  - specify the dataspace
  - create the dataset
  - write the data
  - close the file



HDF5 Fortran API

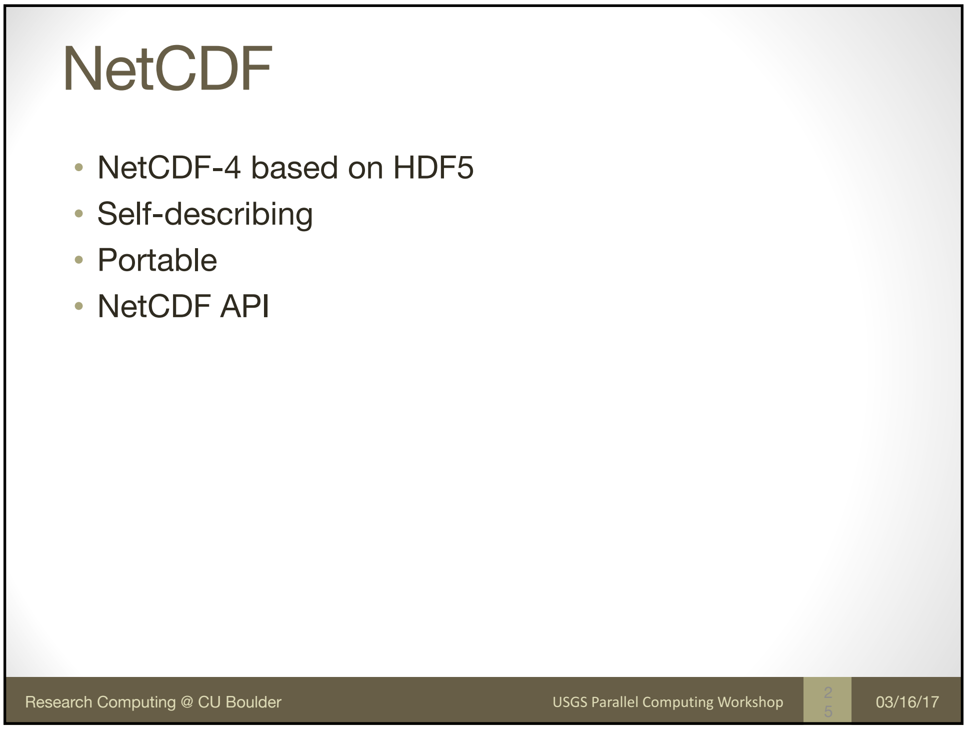
The fortran API is the same as the C API, however subroutines have a _f suffix and the last parameter is the return status.

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



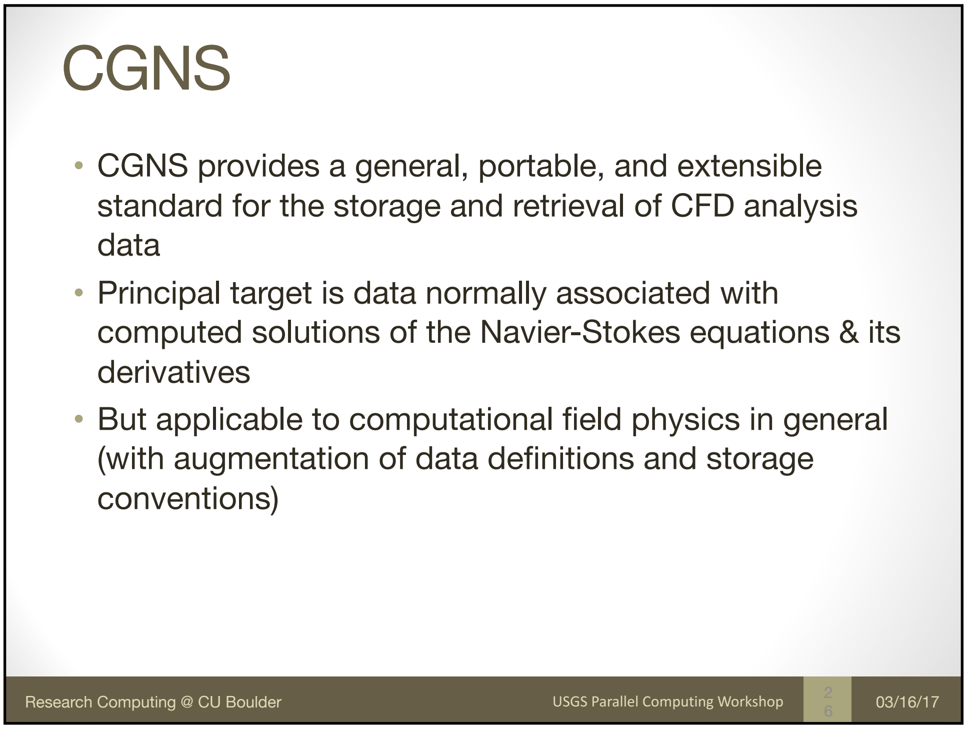
HDF-EOS

- Hierarchical Data Format - Earth Observing System
  - HDF-EOS5 based on HDF5
- NASA lead development
- Stores data collected from EOS satellites
  - Terra
  - Aqua
  - Aura



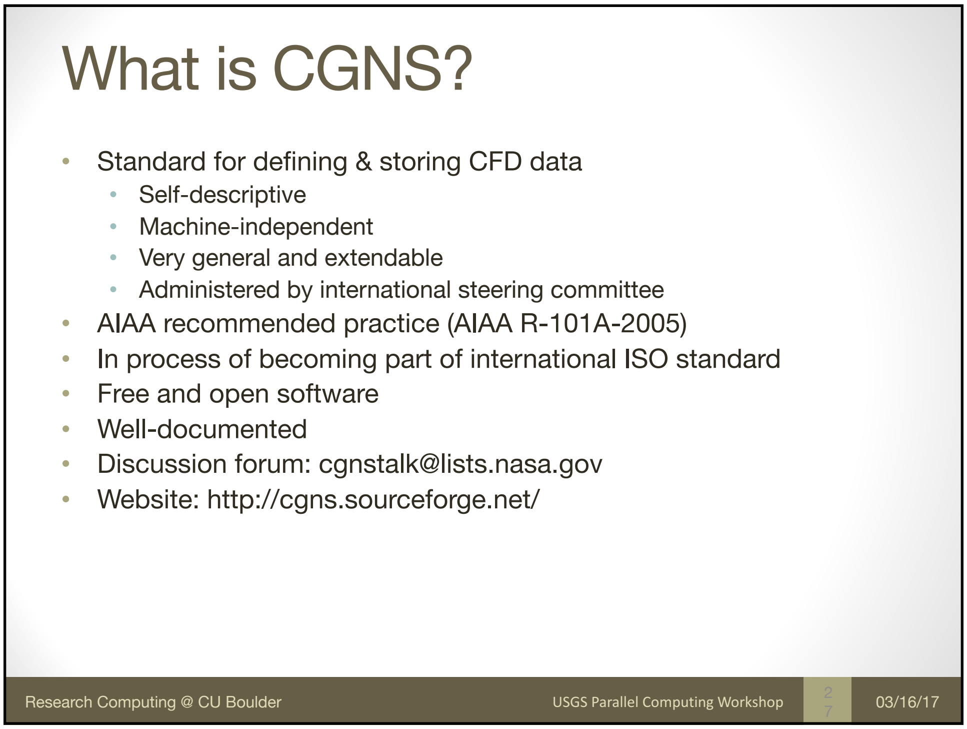
NetCDF

- NetCDF-4 based on HDF5
- Self-describing
- Portable
- NetCDF API



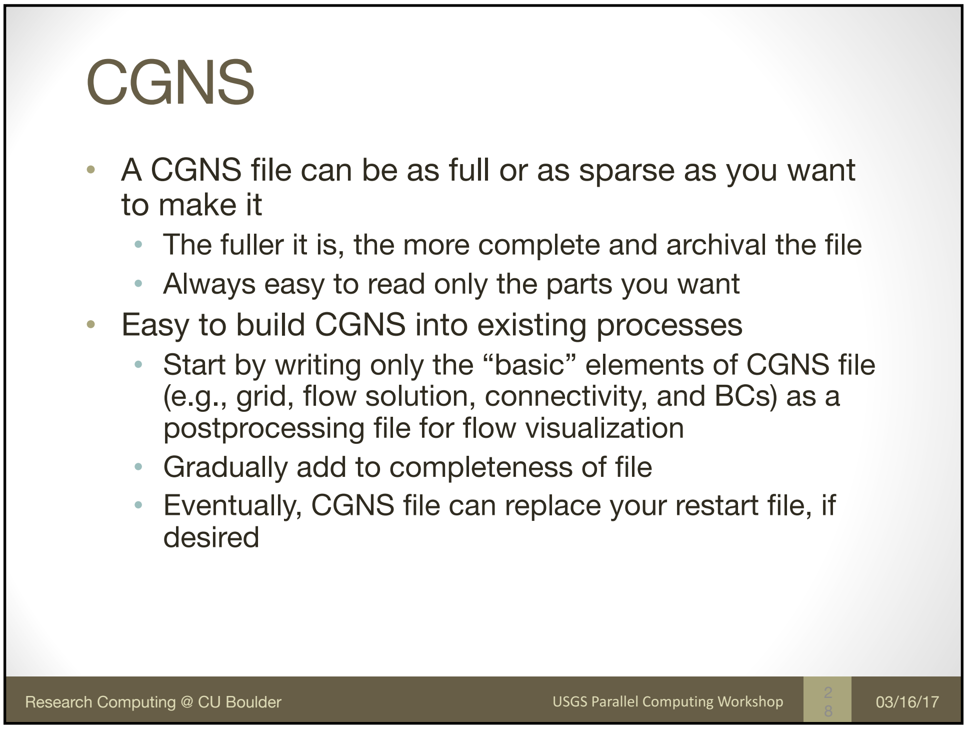
CGNS

- CGNS provides a general, portable, and extensible standard for the storage and retrieval of CFD analysis data
- Principal target is data normally associated with computed solutions of the Navier-Stokes equations & its derivatives
- But applicable to computational field physics in general (with augmentation of data definitions and storage conventions)



What is CGNS?

- Standard for defining & storing CFD data
  - Self-descriptive
  - Machine-independent
  - Very general and extendable
  - Administered by international steering committee
- AIAA recommended practice (AIAA R-101A-2005)
- In process of becoming part of international ISO standard
- Free and open software
- Well-documented
- Discussion forum: cgnstalk@lists.nasa.gov
- Website: http://cgns.sourceforge.net/



CGNS

- A CGNS file can be as full or as sparse as you want to make it
  - The fuller it is, the more complete and archival the file
  - Always easy to read only the parts you want
- Easy to build CGNS into existing processes
  - Start by writing only the "basic" elements of CGNS file (e.g., grid, flow solution, connectivity, and BCs) as a postprocessing file for flow visualization
  - Gradually add to completeness of file
  - Eventually, CGNS file can replace your restart file, if desired