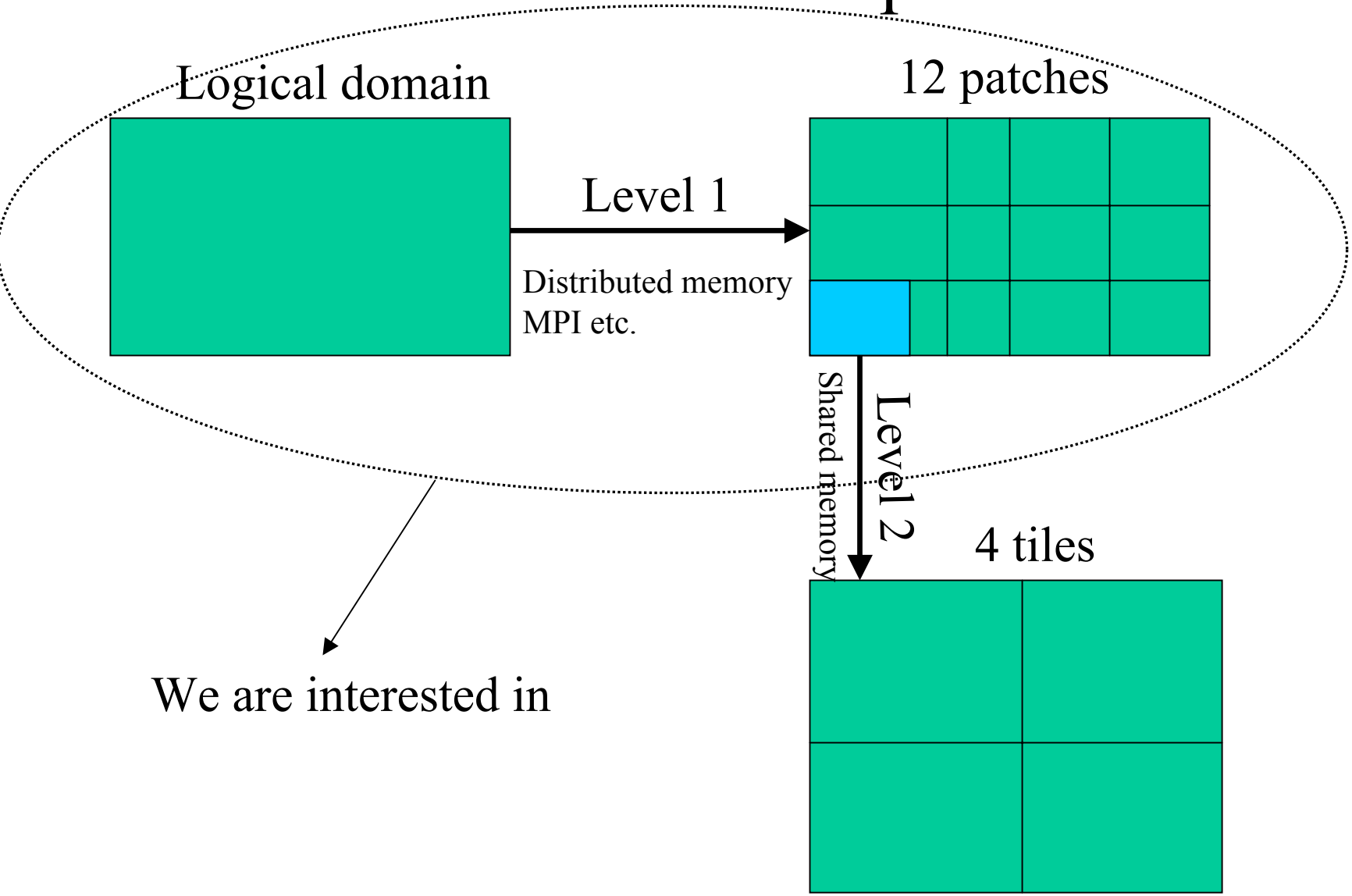
HDF5 WRF I/O module

http://www.ncsa.uiuc.edu/apps/WRF-ROMS

WRF(Weather Research Forecasting Model)

- A limited-area weather model for research and prediction
- Combine NCAR/PSU MM5(research model, dynamics) with NCEP ETA(operational model, physics)
- Have the tendency to become the most popular limited-area weather model in this country and in the world
- Many Potential users
- Most codes in Fortran 77 and Fortran 90

Two level decomposition



WRF data characteristics

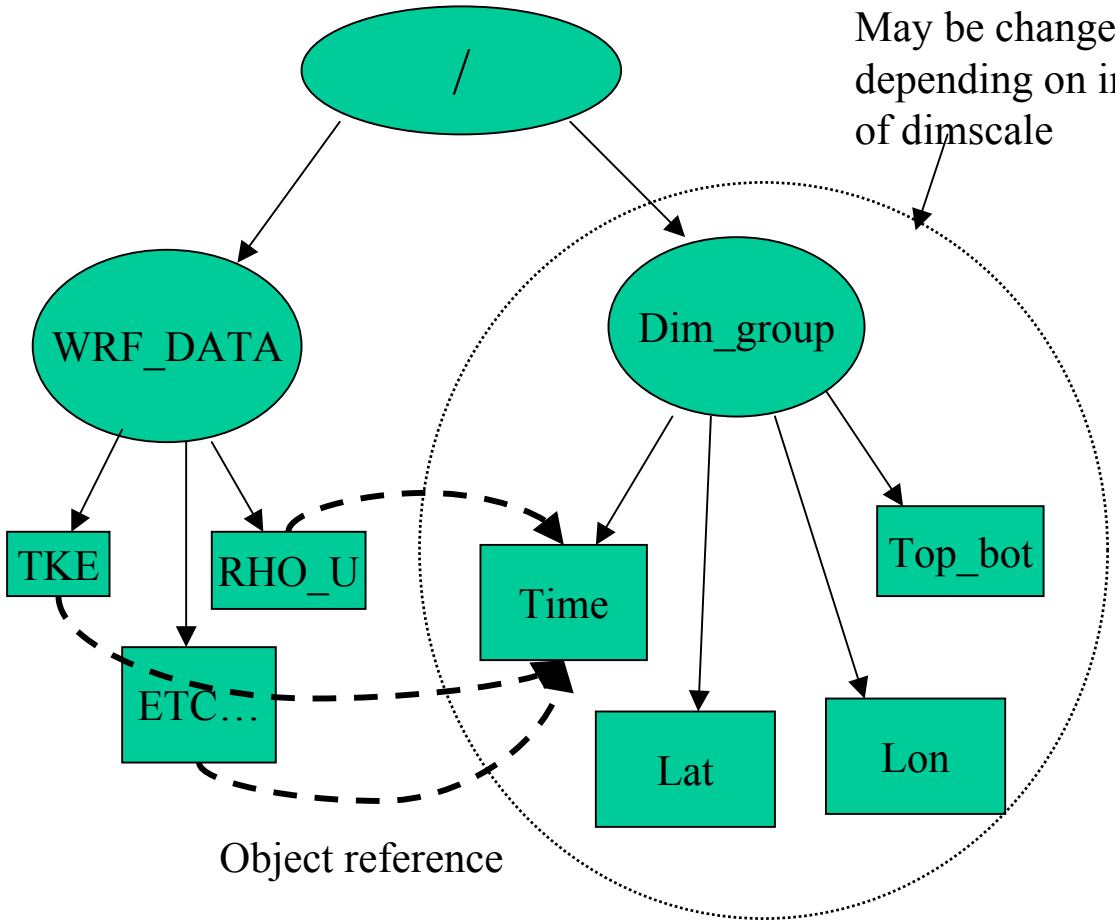
- Many HDF5-equivalent datasets
  - Currently output >100 datasets, each one < 1MB
  - Some application can have close to 100 MB per dataset
  - Some dataset only includes 1 element
  - typical dataset: 4-D real array with the dimension of time extensible
- Weakly dimensional scale requirement
  - no real dimensional scale data
- May need datasets to be extensible
- WRF has history, restart, initial and boundary dataset type. Each WRF dataset is equivalent to one HDF5 group.

An incomplete WRF-HDF5 writer (in Jan. 2003)

- Can write raw data in HDF5 format
- Doesn't implement attributes yet
- Doesn't implement dimensional scale yet

  - wait for implementation of HDF5 dimensional scale
- Has been tested on O2K, PC linux and NCAR IBM SP

First design of Schematic HDF5 File Structure of WRF Output (in Jan. 2003)



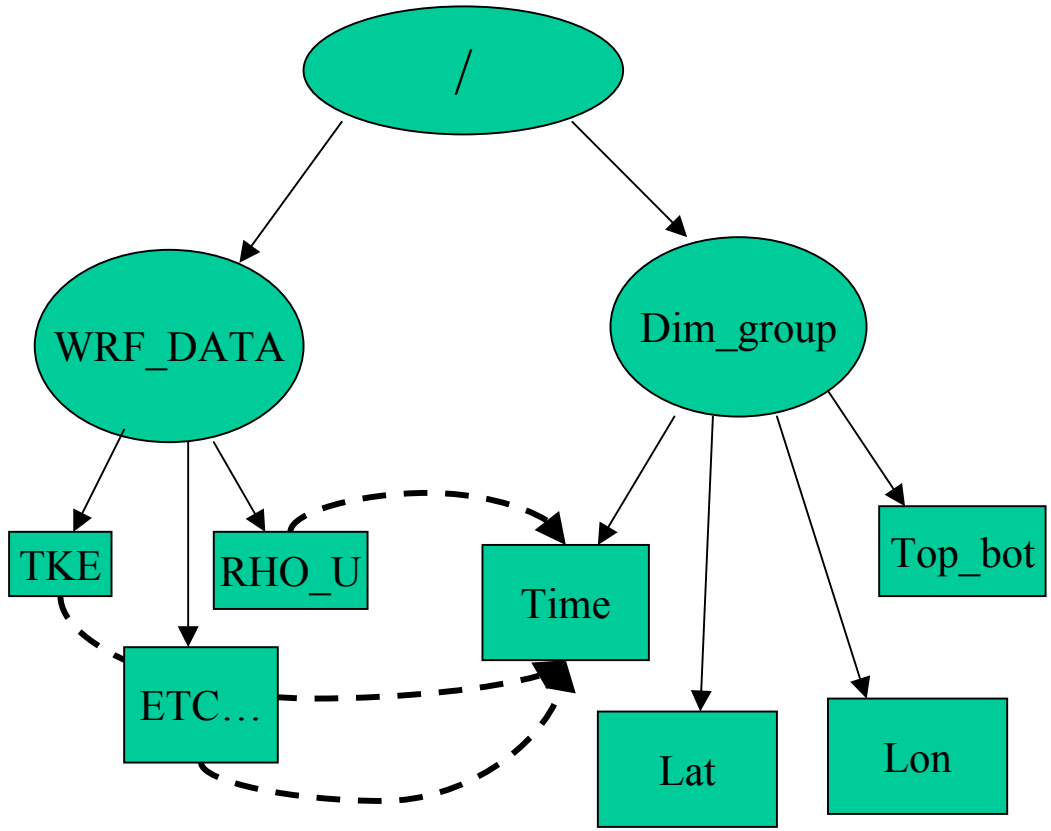
Solid line: HDF5 datasets or sub-groups (the arrow points to) that are members of the HDF5 parent group.
Dash line: the association of one HDF5 object to another HDF5 object;in terms of HDF5, object reference.

HDF5 dataset(WRF field)

HDF5 group(WRF dataset)

Schematic HDF5 File Structure of WRF Output

WRF has history, restart, initial and boundary dataset type. Each WRF dataset is equivalent to one HDF5 group. Since WRF developers request to store different types of WRF datasets in different files, we store at most two groups inside an HDF5 file: a group containing WRF data, and a group that contains dimensions. TK and U are names of WRF data fields. WRF may not need to use the dimensions group in real applications.



Solid line: HDF5 datasets or sub-groups (the arrow points to) that are members of the HDF5 parent group.
Dash line: the association of one HDF5 object to another HDF5 object;in terms of HDF5, object reference.

1. Different WRF datasets will output to different HDF5 files.

One WRF dataset represents one domain.

2. Different dataset type(history,restart,initial,boundary) will also output to different HDF5 files.

Solid line represents HDF5 datasets or sub-groups (the arrow points to) that belong to the HDF5 parent group.
Dash line represents the association of one HDF5 object to another HDF5 object;in terms of HDF5, it is called object reference.

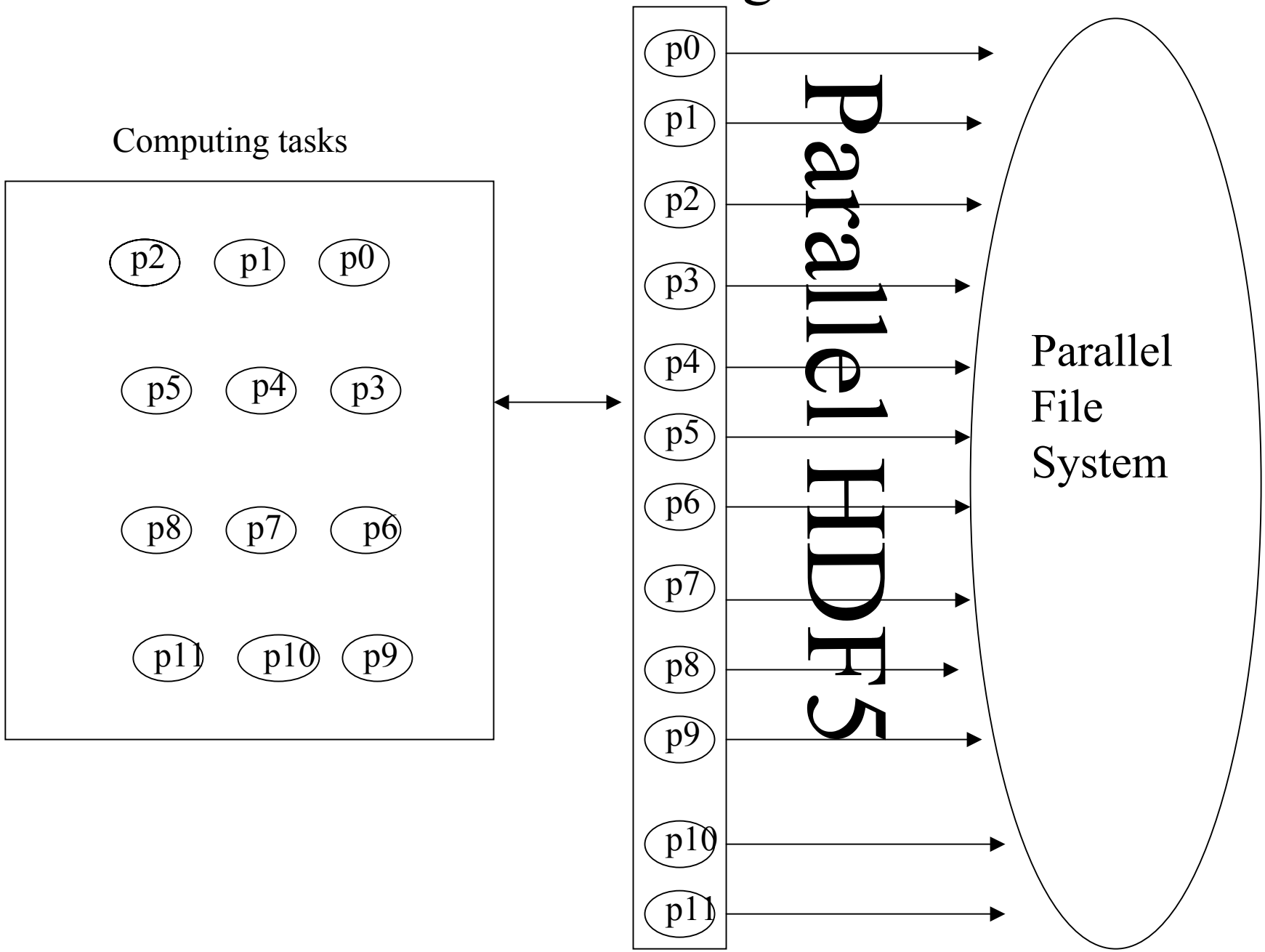
A complete WRF-HDF5 sequential module

- Can generate HDF5 attributes
- Dimensional scale has been implemented with a table
- Have both reader and writer
- Has been tested on NCSA Linux cluster, PC linux and NCAR IBM SP

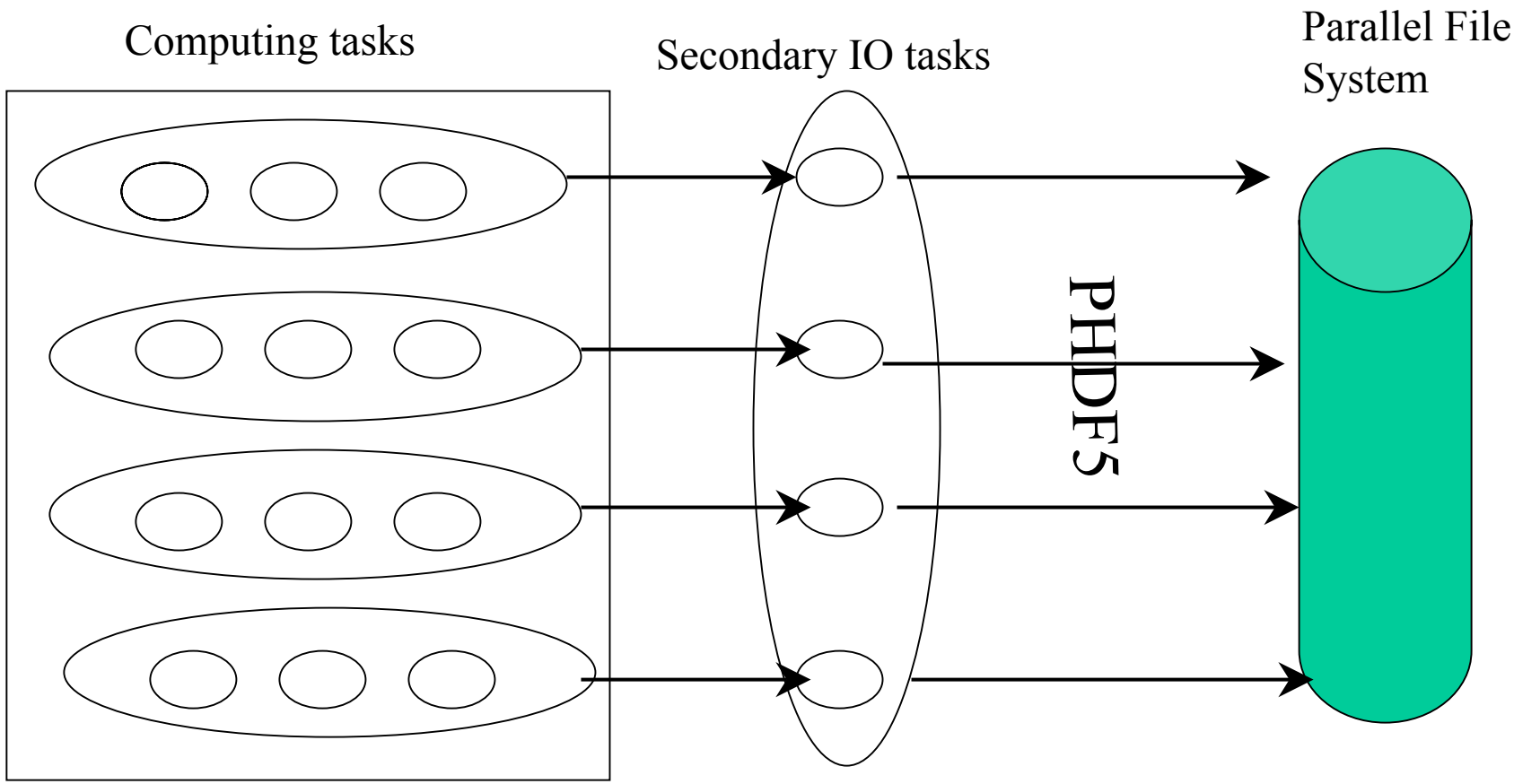
Some facts related to WRF-HDF5 module

			WRF-HDF5 module
Sequential Computing		Sequential IO	Sequential HDF5
Parallel Computing		Sequential IO	Sequential HDF5
Parallel Computing		Parallel IO	Parallel HDF5

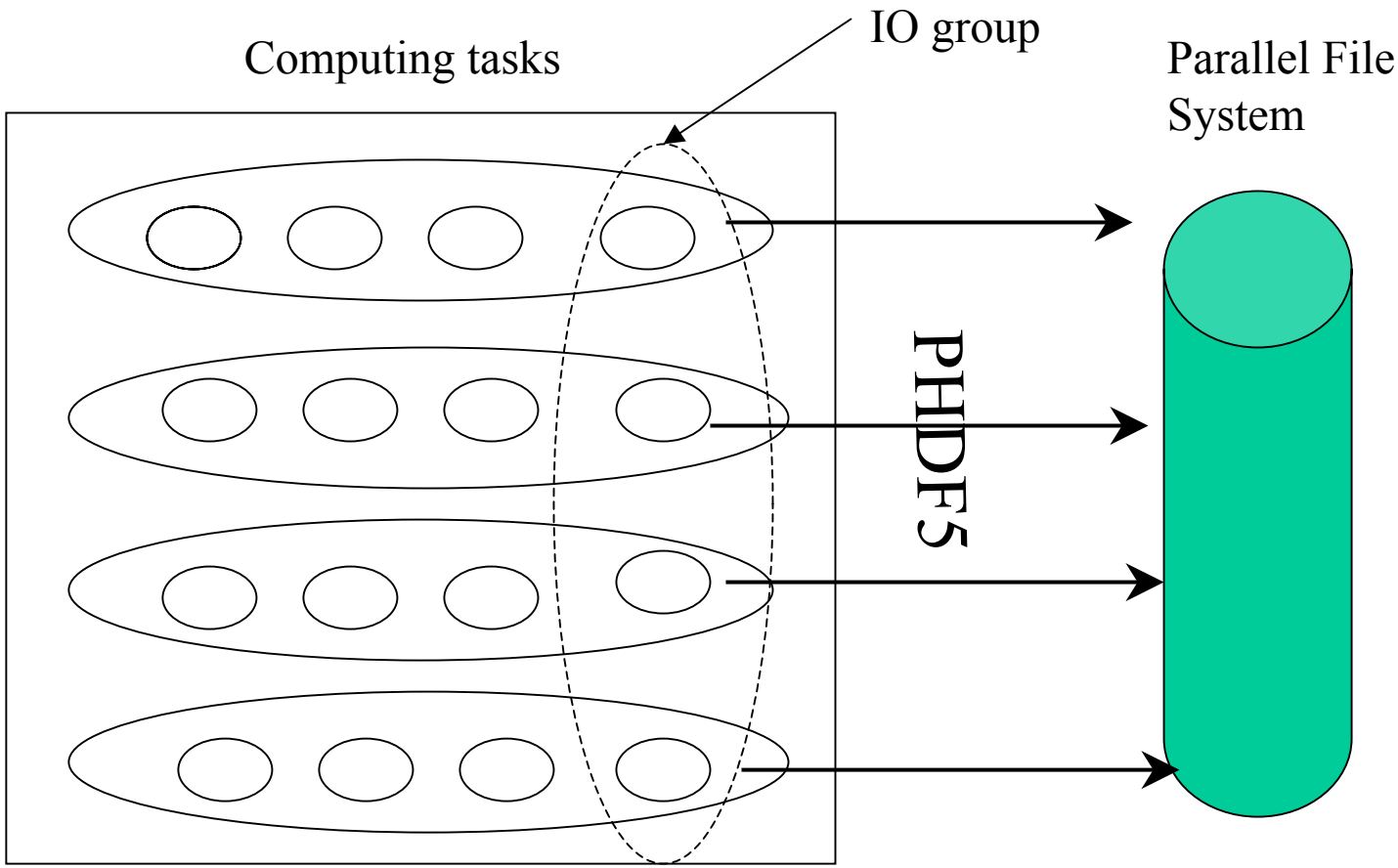
Parallel WRF-HDF5 design - Solution 1



Parallel WRF-HDF5 design-solution 2



Parallel WRF-HDF5 design-solution 3



The current approach

Since HDF5 will implement two-phase IO in its parallel module in the future and the two-phase IO idea is very similar to that in approach 2 and approach 3 and also because there is a short-time need for parallel HDF5 for WRF, we decided to implement approach 1.

3. Analysis procedures:

1) Use stable parallel file system; currently GPFS on IBM SP
2) Check whether the parallel output is the same as sequential output
3) Compare the wall clock running time with sequential run
4) Find the reason if the performance is not good
5) Look for alternative method to improve the performance

The performance analysis

When setting the chunking size the same as the whole domain size, The wall-clock time of parallel HDF5 module is much slower than that of the sequential HDF5 module.

Reason:
The fastest changing dimension of the hyperslab has to be multiple of the fastest changing dimension of the chunking .

When setting the chunking size the same as the whole domain size; Since there is no cache mechanism in the current implementation of parallel HDF5, it reads one row each time to access disk so it slows down the performance, this has been verified by a sample program.

Solution:?

Set chunk size to be equal to the hyperslab size in each processor!

Phenomena: the program is hung.

Reason:
Because the hyperslab size for different process is different; so the chunking size for each process is different and this causes the program hung.

Why the hyperslab size for different process is different? Can we make them the same?

Model input grid information

Some definitions of the user input parameters

s_we (default value of 1)	This is the start index in x (west-east) direction
e_we (default value is 32)	This is the end index in x (west-east) direction
s_sn (default value of 1)	This is the start index in y (south-north) direction
e_sn (default value is 32)	This is the end index in y (south-north) direction
s_vert (default value of 1)	This is the start index in z (vertical) direction
e_vert (default value is 31)	This is the end index in z (vertical) direction - number of full zeta levels

Ds: domain starting point

De: domain ending point

Ps: patch starting point

Pe: patch ending point

Now an example to show the sub domain division of the model run

1) We only use two processes, we set the grid as follows:

s_we: 1, s_sn:1 s_vert: 1
E_we: 16, e_sn:16 e_vert: 15

2) After the model finishes computing, it passes the following info. as well as data to IO module:

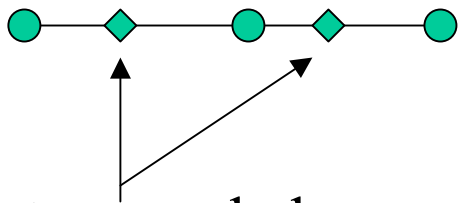
For process 1:
U: XZY
ds 1 1 1 de 16 14 15
ps 1 1 9 pe 16 14 15

V:
ds 1 1 1 de 15 14 16
ps 1 1 9 pe 15 14 16

WW:
ds 1 1 1 de 15 15 15
ps 1 1 9 pe 15 15 15

Why the patch size is not equal divided?

Because many WRF datasets are staggered in different direction!



U- staggered along west-east direction

V in south-nirth direction
W in vertical
So the patch size is different

Final discussion:

We cannot see the changing of HDF5 library in the Very near future. We decide to change our WRF Output layout to use contigouous storage in parallel HDF5 module.