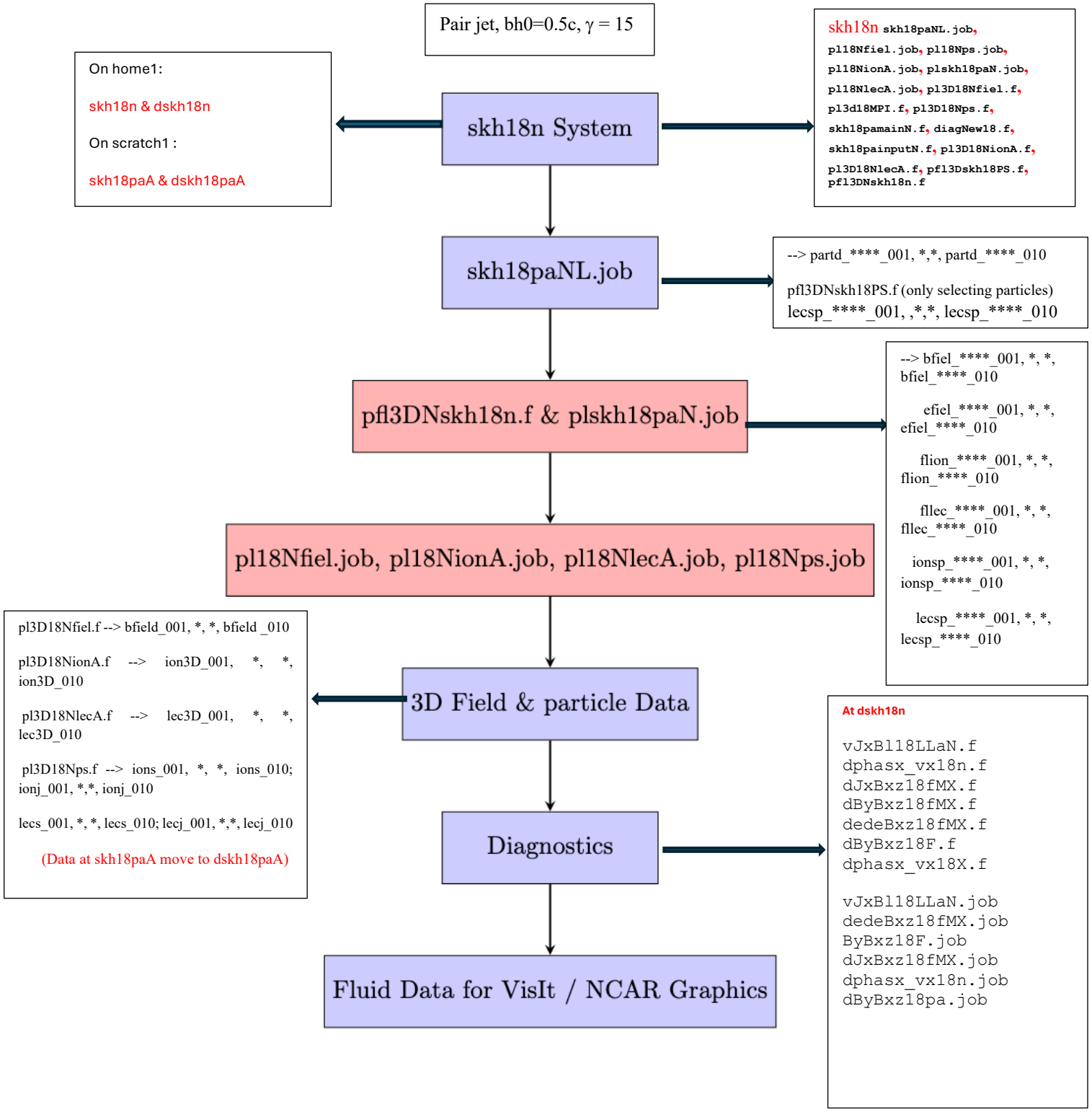
Note 2: Workflow for the new skh18 system (skh18n)

Workflow chart and Frontera manual for new skh18n system



Frontera manual for new skh18n system

1.At first, the directory named skh18n and dskh18n is to be created on Frontera home1

On Frontera

mkdir skh18n

To copy file from local computer to Frontera

scp skh18New.tar ramchandra58088@frontera.tacc.utexas.edu:/home1/09804/ramchandra58088/skh18n

mkdir dskh18

scp dskh18.tar ramchandra58088@frontera.tacc.utexas.edu:/home1/09804/ramchandra58088/dkh18n

2. Copy the provided .tar folder skh18New.tar & skh18njob.tar to skh18n and **dskh18.tar & dskh18job.tar** to **dskh18**

3. Extract the files from sk18New.tar

cd skh18n

tar -xvf skh18New.tar

This command extracts the program files from .tar as they have no unnecessary files. It is different from the sk01.zip file contents. So, we can skip the process of removing unnecessary files and dummy sk01 as in sk01 system.

tar -xvf skh18njob.tar # This command extract program files

lt # list all the extracted files

Here, the code used to extract files on skh18n and dskh18n directories can be done easily for ***.tar files. However, if there are unnecessary files on extracting from .zip or .tar files we can proceed same manual written for sk01 system.

4.Create two temporary directories named skh18paA and dskh18paA on scratch1 for output data storing purpose. These are the temporary directories.

tmp

mkdir skh18paA

mkdir dskh18paA

Note: Learner should be familiar with basic vi command for opening, listing, and editing the program files. For beginner the list of vi command is to be provided.

At this point it is ready to compile main program files and submit the job with necessary modification on job ID's, and you make sure that there is temporary directory skh18paA before job submission for storing output data.

To compile program:

vi comp

Copy and paste the corresponding complier which you want to compile.

For example:

To compile `skh18pamainN.f`

```
vi comp
```

```
mpif90 -r8 -O2 -o xskh18pamainN skh18pamainN.f
```

Copy above line and get-out from it and paste on Frontera, now it gets compile, and it appears as xskh18pamainN i.e. executive files and ready for job run `skh18paNL.job`

How to modify job scripts for job run

e.g.

```
#SBATCH -e skh18paNL%j       # Name of stderr error file
#SBATCH -p normal            # Queue (partition) name
#*SBATCH -N 196              # Total # of nodes
#*SBATCH --tasks-per-node 16 # Total # of mpi tasks
#SBATCH -N 392              # Total # of nodes
#SBATCH --tasks-per-node 8 # Total # of mpi tasks
#SBATCH -n 3136           # Total # of mpi tasks
##SBATCH --tasks-per-node 12  # Total # of mpi tasks
#SBATCH -t 12:30:00         # Run time (hh:mm:ss)
#SBATCH --mail-type=all    # Send email at begin and end of job
#SBATCH -A AST23035      # Project/Allocation name (req'd if you have more than 1)
#SBATCH --mail-user= ramchandra58088@tacc.utexas.edu

# Any other commands must follow all #SBATCH directives...
module list
pwd
date

# cds

# cd skh09ra

cd /scratch1/09804/ramchandra58088/skh18paA


\cp /home1/09804/ramchandra58088/skh18n/xskh18pamainN .

# Launch MPI code...
ibrun ./xskh18pamainN > dumpout   # Use ibrun instead of mpirun
```

Instruction: The user id should be change with own user id name by using cw command at red words and allocate your working directory by checking with pwd and then edit with vi command. We need to do so for both cd and cp very carefully.

After modifying job ID's, it is ready to submit job with command: sb **`skh18paNL.job`**

Check the job status with command: lq

After submission of the job, you will get the output dataset files on skh18paA. To check this

`tmp`

`cd skh18paA`

`lt` --> partd_****_001, *,*, partd_****_010

pfl3DNskh18PS.f (only selecting particles) lecsp_****_001, ,*,*, lecsp_****_010

To extract fluid quantities, first compile pfl3DNskh18n.f and run the job plskh18paN.job

Here complier `ifort -O3 -o xpfl3Dskh18n -shared-intel -mcmodel=large pfl3Dskh18n.f` is used

After complying, It's need to modify job script and submit the job it gives fluid quantities at skh18paA

This generates fluid data in sequential order as it is based on parallel version

--> bfiel_****_001,*,*, bfiel_****_010

efiel_****_001,*,*, efiel_****_010

flion_****_001,*,*, flion_****_010

fllec_****_001,*,*, fllec_****_010

The list of job files to combine the fluid data into 3D fields and particles data are

pl18Nfiel.job, pl18NionA.job, pl18NlecA.job, pl18Nps.job

`pl3d18MPI.f` – This is on progress.

After modifying JOB ID's and lines cd and cp with pwd then running job step by step, the data obtained are

pl3D18Nfiel.f --> bfield_001,*,*, bfield _010

pl3D18NionA.f --> ion3D_001,*,*, ion3D_010

pl3D18NlecA.f --> lec3D_001,*,*, lec3D_010

pl3D18Nps.f --> ions_001,*,*, ions_010; ionj_001,*,*, ionj_010

lecs_001, *,*, lecs_010; lecj_001,*,*, lecj_010

These 3D dataset at skh18paA are move to dskh18paA for further diagnostics

mv bfield* le* ion3D* lec3D* io* lecs* lecj* ../dskh18paA

Diagnostics process start at dskh18n

cd

cd dskh18n

lt

```
-- > vJxBl18LLaN.f
 dphasx_vx18n.f
dJxBxz18fMX.f
dByBxz18fMX.f
dedeBxz18fMX.f
dByBxz18F.f
dphasx_vx18X.f

vJxBl18LLaN.job
dedeBxz18fMX.job
ByBxz18F.job
dJxBxz18fMX.job
dphasx_vx18n.job
dByBxz18pa.job
```

How to get the visitdata

First compile vJxBl18LLaN.f and then modifying the vJxBl18LLaN.job scripts with your own job ID's and path for directories

Modify the job file with the correct directory and ID's, then save and exit with command wq

Path for working directory should be checked with pwd for both cd and cp code lines.

For e.g.

```
cd /scratch1/09804/ramchandra58088/dskh18paA

\cp /home1/09804/ramchandra58088/dskh18n/xvJxBl18LLaN .

./xvJxBl18LLaN > plotout
```

After Submitting the prepared job, you will get the data on dskh18paA by following step

tmp

cd dskh18paA

lt. .....visitdata

Let's change the '**visitdata**' to `vJxBl18LLaN**` **.vtk** files using the following commands:

mv visitdata vJxBl18LLaN**.vtk

Now, the vJxBl18LLaN**.vtk file, move to local computer then run on VisIt in a local computer

Diagnostics with NCAR Graphics to plot plasma fluid quantities for e.g. x-v_x, Jx, By, n, Ex Plot

For this, at dskh18n first, we need to compile the listed program files

cd

cd dskh18n

lt

The list of programs is

```
dphasx_vx18n.f
dJxBxz18fMX.f.
dByBxz18fMX.f.
dedeBxz18fMX.f
dByBxz18F.f
dphasx_vx18X.f


 dedeBxz18fMX.job
 ByBxz18F.job
dJxBxz18fMX.job
dphasx_vx18n.job
dByBxz18pa.job
```

Modify its job script with own user's ID and checking cd and cp then the submit job

Then after submitting job (with necessary id's modification and allocation of correct directory, we get gmeta.

tmp

cd dskh18paA

lt ……. gives gmeta data

The gmeta file needed to be converted into ****.ps file using creams to ****.ps

which is added in alias .bashrc as follows

alias gpg=/usr/local/bin/gpg

alias vib='vi .bashrc'

alias gmeta2ps='creams'

alias ctg='ctrans -d ps.color gmeta' !gmeta to ps this is added alias

alias sob='source .bashrc'

ctg > ****.ps

Move the created *****.ps file to local computer.

To get NCARGraphics plot installation of Okular in Linux can be done with command

sudo apt install Okular

Okular ******.ps # I am using Okular in Linux

The differences between plot3Dsk01Rs.f and pf13DNskh18n.f

If we compare program code **plot3Dsk01Rs.f** with larger serial version simulation code **pf13DNskh18n.f** , It is found that
plot3Dsk01Rs.f -Generates 25 sets of fields and particles data.
In **pf13DNskh18n.f**, the opening of the do loop it runs only once with do in = 10, 10 which means that serial processing of single nst. It generates the fluid quantities in sequential order with a single processor at a time.

```
      open(3,file="topology")
      do i = 0,Nproc-1
       read(3,*) ii,coords(i,1),coords(i,2),coords(i,3)
      end do
      close(3)
c specify "nst" - which data need to be processed
c      nst = 7
c      open(unit=2, file='nstn', status='old')
c      read(2,*) nst
 c = 1.0
c      do in=11,2,-1
c      do in=1,10
      do in=10,10
```

But in case of smaller size MPI based simulation do in = 1, 25 this loop iterates from n=1 to in = 25 giving multiple field and particles datasets at a time. In new simulation the fluid data are generated in sequential order processing one dataset at a time. After generating, those data set are

manually combined into 3D particles and fields data. So, in case of larger new simulation, we need to generate of fluid quantities and combining it into 3D data set manually.

Why MPI Failed to new simulation system?

Previously, we attempted to use MPI to compute fluid quantities but failed as the combining partd_****_010 files across all processors requires **more than 2TB**.