Create new surface dataset

Steps to create a complete set of files for input to CLM

1. (if NOT already done) Create SCRIP grid datasets: use **mkmapgrids or mkscripgrid.ncl or mknoocnmap.pl**
2. (if NOT already done) Create domain dataset: use **gen_domain**
3. (if NOT already done) Create mapping files for mksurfdata_map: use **mkmapdata.sh**
4. Create surface datasets: use **mksurfdata_map**

(There may be some bugs in the above tools, happy debugging!)

5. Enter the new datasets into the build-namelist XML database
6. Create some sort of initial condition dataset, e.g. a) Use spinup-procedures to create initial condition datasets; b) Use interpinic to interpolate existing initial condition datasets
7. Create site level meteorological forcing data. Use R scripts to extract data to a specific site from NCEP or ERA global dataset

For steps 1 to 4, you can use the tools under the following directory:
`/cluster/projects/nn2345k/yfan/ctsm-dev/tools/`
where the shell scripts and namelist files are already modified and the built tools are ready to use.

PTCLM is adapted and ready to use f**or point sites** on fram. See `/cluster/projects/nn2345k/yfan/ctsm-dev/tools/PTCLM/README-YF`

PTCLM:
To solve the out of memory issue with UGRID_1km-merge-10min_HYDRO1K-merge-nomask_c130402.nc, set in PTCLM/batchque_bigmem.py: 'fram' :"--nodes=4 --ntasks-per-node=1 --cpus-per-task=1 --partition=bigmem --mem-per-cpu=128G " dedicated for this map.

#first submit PTCLMsublist with normal queue to create most mapping files except 1km-merge-10min_HYDRO1K

```
./PTCLMsublist -l Chersky,Twitchell -d $CSMDATA --account=nn2345k --mach=fram --
cesm_root ../../
./PTCLMsublist -l FrForTr -d $CSMDATA --account=nn2345k --mach=fram --
cesm_root ../../ -o "--crop --map_gdate 190718 --sdate 190718"
#-o is for extra options --crop --map_gdate passed to PTCLMmkdata

#or directly run PTCLMmkdata with command line options:

sbatch --nodes=1 --ntasks-per-node=1 --cpus-per-task=1 --qos=preproc --
account=nn2345k --time 00:30:00  ./PTCLMmkdata --cesm_root ../../ -s FrForTr -d
/cluster/shared/noresm/site_inputdata --crop --map_gdate 190718
```

#then submit PTCLMsublist_bigmem to produce mapping for 1km-merge-10min_HYDRO1K

```
./PTCLMsublist_bigmem -l Chersky -d $CSMDATA --account=nn2345k --mach=fram --
cesm_root ../../
```

#for the last step mksurfdata_map in PTCLM, make sure all raw data files exist in the CSMDATA directory and all mapping files have the date corresponding to -usr_gdate 190717

```
#rename all mapping files by:
rename '190719' '190718' map_*190719.nc
rename '190717' '190718' map_*190717.nc

#Last step run mksurfdata.pl on the command line or use PTCLMsublist
```

```
cd $project/ctsm-dev/tools/mksurfdata_map/
mksurfdata.pl -res usrspec -usr_gname 1x1pt_Twitchell -usr_gdate 190717 -
usr_mapdir $project/ctsm-dev/tools/PTCLM/mydatafiles/1x1pt_Twitchell -dinlc
/cluster/shared/noresm/site_inputdata -y 2000 -soil_cly 20.0 -soil_snd 40.0 -
pft_frc "[100.0]" -pft_idx "[13]" -no-crop

cd $project/ctsm-dev/tools/PTCLM
./PTCLMsublist -l FrForTr,CRForTr,VaCropTr,GeCropTr -d $shared/site_inputdata --
account=nn2345k --mach=fram --cesm_root ../../ -o "--crop --map_gdate 190718 --
sdate 190718"
```

#To create flanduse.timeseries file

```
#first create PTCLM_sitedata/*LuTr_dynpftdata.txt
#and then submit PTCLMmkdata with option --clmnmlusecase 20thC_transient

#rename all mapping files of VaCropTr,GeCropTr,FrForTr,CRForTr to
VaLuTr,GeLuTr,FrLuTr,CRLuTr to avoid recreating
rename 'VaCropTr' 'VaLuTr' mydatafiles/1x1pt_VaCropTr/map_*.nc
mv mydatafiles/1x1pt_VaCropTr/ mydatafiles/1x1pt_VaLuTr/

#run PTCLMmkdata on the command line
./PTCLMmkdata --cesm_root ../../ -s VaLuTr -d
/cluster/shared/noresm/site_inputdata --crop --map_gdate 190718 --clmnmlusecase
20thC_transient

#or submit to batch
./PTCLMsublist -l VaLuTr,GeLuTr,FrLuTr,CRLuTr -d $CSMDATA --account=nn2345k --
mach=fram --cesm_root ../../ -o "--crop --map_gdate 190718 --clmnmlusecase
20thC_transient"
```

Steps for regional dataset

See **models/lnd/clm/tools/README** for details

Step 1: Creating a regional SCRIP grid file at a resolution to run the model on

```
        a.) For standard resolutions these files will already be created (done).

        b.) To create regular lat-lon regional/single-point grids run
mknoocnmap.pl

         This will create both SCRIP grid files and a mapping file that will be
valid if the region includes NO ocean whatsoever (so you can skip step 2).  You
can also use this script to create SCRIP grid files for a region (or even a
global grid) that DOES include ocean if you use step 2 to create mapping files
for it (simply discard the non-ocean map created by this script).

          Example, for single-point over Boulder Colorado.
             cd mkmapdata
             ./mknoocnmap.pl -p 40,255 -n 1x1_boulderCO

        c.) To convert from CLM or CAM grid files

         To convert CLM grid files or CAM grid files to SCRIP grid files use the
deprecated mkgriddata program if you have to. This is also a required step if
you want to run fine-mesh as it's the only way to get CLM topo files.

        d.) General case (done here, see step 1 option d)

         You'll need to convert or create SCRIP grid files on your own (using
scripts or other tools) for the general case where you have an unstructured
grid, or a grid that is not regular in latitude and longitude.
```

#Option 1.b.

To create a regular latitude/longitude regional grid, the first step is to use the tool **mknoocnmap.pl** in `~/NorESM/components/lnd/clm/tools/shared/mkmapdata` to create both the SCRIP grid file that can then be used as input to **mkmapdata.sh** AND an identity mapping file (mapping from ocn to atm, that is then input to **gen_domain**) assuming there is NO ocean in your grid domain. These files are used in the next step.

#First set the environmental variable ESMFVIN_PATH and other library paths
For hexagon:

```
export ESMFBIN_PATH=/work/apps/esmf/6.3.0rp1-dso-
gnu/bin/binO/Unicos.gfortran.64.mpi.default/

export NCARG_ROOT=/work/shared/noresm/ncl_ncarg-6.1.2/
export PATH=$NCARG_ROOT/bin:$PATH
export PATH=/work/apps/nco/4.5.0-ncap2-gnu/bin:$PATH
module swap PrgEnv-cray PrgEnv-intel
module load intel/16.0.3
#module load cray-netcdf/4.3.3.1
#module load cray-mpich/7.5.3
```

For GWDG:

```
export ESMFBIN_PATH=/usr/users/yfan1/local/ncl_ncarg-6.1.2/bin
export NCARG_ROOT=/usr/users/yfan1/local/ncl_ncarg-6.1.2
export PATH=$NCARG_ROOT/bin:$PATH
export PATH=/usr/users/yfan1/local/bin:$PATH
module load intel/compiler/64
```

#Run the mknoocnmap.pl tool

```
cd ~/NorESM/components/clm/tools/shared/mkmapdata/
     mknoocnmap.pl
 -centerpoint [or -p] <lat,lon> Center latitude,longitude of the grid to create.
 -name [-or -n] <name>     Name to use to describe point
 -dx <number> Size of total grid in degrees in longitude direction (default 0.1)
 -dy <number> Size of total grid in degrees in latitude direction (default 0.1)
 -nx <number>                  Number of longitudes (default is 1)
 -ny <number>                  Number of latitudes  (default is 1)
```

For Sumatra regional grid (get the grid center lon/lat from mksrf_Sumatra_LC2010_3x3min_Landsat_c170921.nc so that the domain lon/lat of input raw data and the SCRIP gridmap lon/lat could match)

```
./mknoocnmap.pl -centerpoint -0.025,100.575 -name 3x3min_Sumatra -dx
11.25 -dy 12.25 -nx 225 -ny 245
```

(3 minute = 3*1.85 km, dx and dy are the total size of the whole gridmap =nx / ny * resolution 0.05)

Output files:
SCRIPgrid_1x1min_Sumatra_noocean_c170617.nc
SCRIPgrid_1x1min_Sumatra_nomask_c170617.nc
map_1x1min_Sumatra_noocean_to_1x1min_Sumatra_nomask_aave_da_170617.nc

#Align the regional grid to a regular global grid with the same resolution (starting from -90 lat and -180 lon), setting center point to the nearest grid center of the intended global grid

```
./mknoocnmap.pl -centerpoint -0.025,107 -name 3x3min_Indonesia -dx 24 -dy
12.25 -nx 480 -ny 245
```

(resolution: 24/480, 12.25/245 = 0.05 degree =3min, dx and dy are the total size of grid)

Output files:
SCRIPgrid_3x3min_Indonesia_nomask_c170626.nc
SCRIPgrid_3x3min_Indonesia_noocean_c170626.nc
map_3x3min_Indonesia_noocean_to_3x3min_Indonesia_nomask_aave_da_170626.nc

The SCRIPgrid nomask file is the grid file at a resolution to run the model on and it is used as input to mkmapdata.sh. SCRIP grid is a descriptor file for your grid, which includes the locations of cell centers and cell corners. There is also a "mask" field, but in this case the mask is set to 1 everywhere (i.e. all of the masks for the output model grid are "nomask")

The third map file is an identity mapping file (from the ocn to atm). It assumes there is NO ocean in your grid domain. The mapping file is then used as input to gen_domain to make atmosphere and ocean domain files. Do not use this mapping file for mksrf_fvegtyp in mksurfdata_map if the region DOES include ocean!

Try Use mkunitymap-yfan.ncl to create a new identify mapping file with two same grids “SCRIPgrid_1x1min_Sumatra_nomask_c170617.nc”. However, the output map file

"map_1x1min_nomask_to_1x1min_Sumatra_nomask_aave_da_c170621.nc" still cannot be used for mksrf_fvegtyp because it has nomask whereas the raw land cover file 1x1min_Sumatra has a specific land mask.

```
export NCARG_ROOT=/work/shared/noresm/ncl_ncarg-6.1.2/
export PATH=$NCARG_ROOT/bin:$PATH
```
ncl mkunitymap-yfan.ncl

Add a new resolution and the above SCRIP regional file in namelist_defaults_clm4_5_tools.xml

#Option 1.c/d.

Use mkmapgrids to make SCRIPgrid directly from "mksrf_Sumatra_LC2010_1x1min_c170615.nc"

```
Otherwise there will be domain_checksame ERROR at step 6: input domain mask and
gridmap mask are not the same

##Compile the tool mkmapgrids
      cd ~/post4.5Crop_yfan/models/lnd/clm/tools/shared/mkmapgrids
      cd src
      export LIB_NETCDF=/usr/product/applsw/netcdf-4.3.0/lib
      export INC_NETCDF=/usr/product/applsw/netcdf-4.3.0/include
      export USER_FC=mpiifort
      export USER_CC=mpiicc
      module load intel/compiler/64
      module load intel/mpi/64/2017/2.174
      gmake
```

#make the SCRIPgrid from land cover data mksrf_Sumatra_LC2010_3x3min_Landsat_c170921.nc

```
      ./mkmapgrids < namelist-Sumatra

      In the namelist file:

      &mkmapgrids_in

        fname_in  =
      '/usr/users/yfan1/cesm_input/lnd/clm2/rawdata/pftlandusedyn.1x1min.Indones
      ia.c170615/mksrf_Sumatra_LC2010_3x3min_Landsat_c170921.nc'

        fname_out = 'SCRIPgrid_3x3min_Sumatra_Landsat_c170921.nc'
      /

      Output file SCRIPgrid_3x3min_Sumatra_Landsat_c170921.nc, will be used by
      mkmapdata.sh (mkmapgrids tool by default assumes a global grid and
      computes cell corner locations with min/max lat of -90 to 90!!!
      Make sure to add cell corner variables LONW/LONE/LATS/LATN to the mksrf
      file)
```

Step 2: Create ocean to atmosphere mapping file (if needed)

```
        a.) Standard resolutions (done)

        If this is a standard resolution with a standard ocean resolution --
this step is already done, the files already exist.

        b.) Region without Ocean (done in step 1.b)
```

```
        IF YOU RAN mknoocnmap.pl FOR A REGION WITHOUT OCEAN THIS STEP IS ALREADY
DONE.

        c.) New atmosphere or ocean resolution

        If the region DOES include ocean, use gen_domain/gen_maps.sh to create a
mapping file for it.

    Example:

    cd gen_domain
    ./gen_maps.sh -focn <ocngrid> -fatm <atmgrid> -nocn <ocnname> -natm
<atmname>
    This scrip will use another script gen_ESMF_mapping_file/create_ESMF_map.sh
```

Option 2.c.

Sumatra or Indonesia region does have ocean, need to create an ocean to atmosphere mapping file that cannot be provided by step 1 option b.

```
For GWDG

export ESMFBIN_PATH=/usr/users/yfan1/local/ncl_ncarg-6.1.2/bin

export NCARG_ROOT=/usr/users/yfan1/local/ncl_ncarg-6.1.2
export PATH=$NCARG_ROOT/bin:$PATH
export PATH=/usr/users/yfan1/local/bin:$PATH
export PATH=/usr/users/yfan1/local/nco4.4.2/bin:$PATH

cd ~/post4.5Crop_yfan/tools/mapping/gen_mapping_files
see README

./gen_cesm_maps.sh -focn
~/post4.5Crop_yfan/models/lnd/clm/tools/shared/mkmapgrids/SCRIPgrid_3x3min
_Landsat_Indonesia_c170628.nc -fatm
~/post4.5Crop_yfan/models/lnd/clm/tools/shared/mkmapgrids/SCRIPgrid_3x3min
_Indonesia_nomask_c170628.nc -nocn 3x3min_Landsat_navy -natm
3x3min_Indonesia_nomask --typeocn regional --typeatm regional

Output: Successfully created mapping file
map_3x3min_Landsat_navy_TO_3x3min_Indonesia_nomask_aave.170927.nc


Then create domain using the above map at step 5
./gen_domain -m
~/post4.5Crop_yfan/tools/mapping/gen_mapping_files/map_3x3min_Landsat_navy
_TO_3x3min_Indonesia_nomask_aave.170927.nc -l 3x3min_Indonesia -o
Landsat_navy
```

OR use ncl script to create ocngrid (navy) and atmgrid (nomask) for the region that does have ocean. The script will use a baseline land/ocean mask of 1x1 degree resolution. See details in the script.

```
cd ~/post4.5Crop_yfan/models/lnd/clm/tools/region
> ncl script_regionSumatra-navy.ncl

Three outputs ocngrid, atmgrid, ocn-atm mapping file:
srcGridFile=navy/SCRIP_Sumatra_3x3min_navy.nc,
dstGridFile=nomask/SCRIP_Sumatra_3x3min_nomask.nc,
mapfile/map_Sumatra_3x3min_navy_to_Sumatra_3x3min_nomask_aave_da.nc
```

Use the above map file to make domains with gen_domain at step 5

Step 3 EDIT NAMELIST FILES

Edit models/lnd/clm/bld/namelist_files/namelist_defaults_clm4_5_tools.xml

```
<!-- <lmask type="veg"              >MODIS</lmask> -->
<lmask type="veg"              >Landsat</lmask>
<!-- <hgrid type="veg"              >0.5x0.5</hgrid> -->
<hgrid type="veg"              >1x1min</hgrid>
<scripgriddata hgrid="1x1min"  lmask="Landsat"
   >lnd/clm2/mappingdata/grids/SCRIPgrid_1x1min_Sumatra_Landsat_c170622.nc
</scripgriddata>
```

Add namelist items for high resolution PFT data:

```
<!--modify the mask of global 3x3min grid with the landmask of Landsat for
the region of Indonesia, Y.Fan -->
<scripgriddata hgrid="3x3min"    lmask="MODIS"
   >lnd/clm2/mappingdata/grids/SCRIPgrid_3x3min_MODIS_Global-
Indonesia_c170628.nc</scripgriddata>
<!-- other 3x3min data still use the original MODIS mask-->
```

Add namelist items for high resolution LAI:

```
<!-- add new grid mask from Landsat data, Y.Fan -->
<lmask type="veg"              >Landsat</lmask>

<!-- add higher resolution grid for LAI and PFT, Y.Fan-->
<hgrid type="lai"              >0.25x0.25</hgrid>
<hgrid type="veg"              >1x1min</hgrid>
<hgrid type="veg"              >3x3min</hgrid>

<!--add new high resolution global LAI dataset including 75pfts, Y.Fan
07.2017 -->
<mksrf_flai hgrid="0.25x0.25" lmask="MODIS" crop="on"
>lnd/clm2/rawdata/pftcftlandusedynharv.0.25x0.25.MODIS.simyr1850-
2015.c170412/mksrf_lai_78pfts_simyr2005.c170413.nc
</mksrf_flai>

<scripgriddata hgrid="0.25x0.25"
lmask="MODIS"   >lnd/clm2/mappingdata/grids/SCRIPgrid_0.25x0.25_MODIS_c170
412.nc</scripgriddata>
```

Edit namelist_definition_clm4_5.xml to include the new resolution and new lmask "Landsat" for the Landsat derived data of Indonesia:

```
<entry id="lmask" type="char*10" category="mksurfdata"
       group="default_settings"
       valid_values="nomask,navy,AVHRR,MODIS,USGS,IGBPmergeICESatGIS,IGBP-
GSDP,ISRIC-WISE,LandScan2004,GLOBE-Gardner,GLOBE-Gardner-
mergeGIS,GRDC,HYDRO1K-merge-nomask,Landsat">
Land mask description for mksurfdata input files
</entry>
```

```
<entry id="hgrid" type="char*10" category="mksurfdata"
       group="default_settings"
valid_values="0.1x0.1,0.5x0.5,10x10min,5x5min,360x720cru,0.9x1.25,19basin,
1km,1km-merge-10min,3x3min,1x1min,0.25x0.25">
Horizontal grid resolutions for mksurfdata input files
</entry>

<entry id="res" type="char*30" category="default_settings"
       group="default_settings"
       valid_values=
"512x1024,360x720cru,128x256,64x128,48x96,32x64,8x16,94x192,0.23x0.31,0.9x
1.25,1.9x2.5,2.5x3.33,4x5,10x15,5x5_amazon,1x1pt_Bariri,1x1pt_Jambi,1x1_tr
opicAtl,1x1_camdenNJ,1x1_vancouverCAN,1x1_mexicocityMEX,1x1_asphaltjungleN
J,1x1_brazil,1x1_urbanc_alpha,1x1_numaIA,1x1_smallvilleIA,0.1x0.1,0.5x0.5,
3x3min,5x5min,10x10min,0.33x0.33,ne4np4,ne16np4,ne30np4,ne60np4,ne120np4,n
e240np4,wus12,us20,1km,1km-merge-10min,1x1min,1x1min_Indonesia,0.25x0.25">
Horizontal resolutions
Note: 0.1x0.1, 0.5x0.5, 5x5min, 10x10min, 3x3min and 0.33x0.33 are only
used for CLM tools
</entry>
```

Step 4: Run mkmapdata.sh to create mapping file for mksurfdata_map tool

#mkmapdata is used to create SCRIP mapping data file from SCRIP grid files. Each raw dataset that has a different grid, or land-mask needs a different mapping file for it, but many different raw datasets share the same grid/land-mask as other files. Hence, there doesn't need to be a different mapping file for EACH raw dataset -- just for each DIFFERENT raw dataset

Edit mkmapdata.sh to include the new resolutions and mask for clm4_5

```
[ "$phys" = "clm4_5" ]; then
    grids=(                     \
           "0.5x0.5_AVHRR"      \
           "0.5x0.5_MODIS"      \
           "3x3min_LandScan2004" \
           "3x3min_MODIS"       \
           "3x3min_USGS"        \
           "5x5min_nomask"      \
           "5x5min_IGBP-GSDP"   \
           "5x5min_ISRIC-WISE"  \
           "10x10min_nomask"    \
           "10x10min_IGBPmergeICESatGIS" \
#           "3x3min_GLOBE-Gardner" \
#           "3x3min_GLOBE-Gardner-mergeGIS" \
           "0.9x1.25_GRDC" \
           "360x720cru_cruncep" \
           "0.5x0.5_MODIS" \
           "0.25x0.25_MODIS" \
           "3x3min_Landsat" \
#           "1km_Landsat" \
           "1km_nomask" )
#           "1km-merge-10min_HYDRO1K-merge-nomask")
```

```
Output mapping file will be used for raw data
mksrf_Indonesia_LC2010_1km_Landsat_c170625.nc

Creating mapping file:
map_1km_Landsat_to_1x1min_Sumatra_nomask_aave_da_c170625.nc

From input grid:
/usr/users/yfan1/cesm_input/lnd/clm2/mappingdata/grids/SCRIPgrid_1km_Lands
at_Indonesia_c170625.nc

For output grid:
/usr/users/yfan1/post4.5Crop_yfan/models/lnd/clm/tools/shared/mkmapgrids/S
CRIPgrid_1x1min_Sumatra_nomask_c170622.nc
```

Before running the script, setup $CSMDATA and $ESMFBIN_PATH

For Fram: `see /cluster/projects/nn2345k/yfan/ctsm-dev/tools/mkmapdata/mpijob.mkmap_ORNL`

```
#!/bin/bash

##SBATCH --account=nn2345k --partition=bigmem
#SBATCH --account=nn2345k
#SBATCH --job-name=mkmap.ORNL
```

```
#SBATCH --time=00:30:00
#SBATCH --output=mkmap.out
##SBATCH --mail-type=END
##SBATCH --mail-user=yuanchao.fan@uni.no
## Number of nodes:
#SBATCH --nodes=1
#SBATCH --qos=preproc
## Number of tasks to start on each node:
#SBATCH --ntasks-per-node=1
## Set OMP_NUM_THREADS
#SBATCH --cpus-per-task=1
## To request big memory:
##e.g. below each task can use 512GB memory total
##SBATCH --nodes=4 --ntasks-per-node=1 --cpus-per-task=1
##SBATCH --mem-per-cpu=512G

cd /cluster/shared/noresm/site_inputdata/tools/mkmapdata
export ESMFBIN_PATH=/cluster/software/ESMF/6.3.0rp1-intel-2017a-HDF5-
1.8.18/bin
export CSMDATA=/cluster/shared/noresm/site_inputdata
module load NCO/4.6.6-intel-2017a

## Run the application
#srun ./mkmapdata.sh --
gridfile ../mkmapgrids/SCRIPgrid_1x1pt_ORNL_nomask_c180806.nc --res
1x1pt_ORNL --gridtype regional
#using srun will cause the script to stop after making each map, directly
run the script
./mkmapdata.sh --
gridfile ../mkmapgrids/SCRIPgrid_1x1pt_ORNL_nomask_c180806.nc --res
1x1pt_ORNL --gridtype regional
```

Submit the job: sbatch mpijob.mkmap_ORNL

For hexagon:

```
export CSMDATA=/work/shared/noresm/point_inputdata

export ESMFBIN_PATH=/work/apps/esmf/6.3.0rp1-dso-
gnu/bin/binO/Unicos.gfortran.64.mpi.default/

export NCARG_ROOT=/work/shared/noresm/ncl_ncarg-6.1.2/
export PATH=$NCARG_ROOT/bin:$PATH
export PATH=/work/apps/nco/4.5.0-ncap2-gnu/bin:$PATH

module swap PrgEnv-cray PrgEnv-intel
module load intel/16.0.3
```

For GWDG:

#First set the environmental ESMFVIN_PATH to ESMF_RegridWeightGen and NCL, NCO.

```
export ESMFBIN_PATH=/usr/users/yfan1/local/ncl_ncarg-6.1.2/bin

#OR use a newer version of ESMF from hexagon to avoid errors in mapping

#add ESMFBIN_PATH and also add the shared library object libesmf.so to
LD_LIBRARY_PATH (still missing netcdf c++ linker)



export NCARG_ROOT=/usr/users/yfan1/local/ncl_ncarg-6.1.2
export PATH=$NCARG_ROOT/bin:$PATH
```

```
      export PATH=/usr/users/yfan1/local/bin:$PATH
      export PATH=/usr/users/yfan1/local/nco4.4.2/bin:$PATH
      ##if libnco.4.4.2 cannot be found, then has to export
LD_LIBRARY_PATH=/usr/users/yfan1/local/nco4.4.2/lib:$LD_LIBRARY_PATH
      module load intel/compiler/64
      module load intel/mpi/64/2017/2.174
      ###do not load netcdf (otherwise may conflict with nco library)

      export CSMDATA=/usr/users/yfan1/cesm_input
```

On GWDG

```
>bsub -ISs -q fat -n 64 -R 'span[hosts=1]' -R np64 -W 00:10
/bin/bash ./mkmapdata-Indonesia.sh --gridfile
~/post4.5Crop_yfan/models/lnd/clm/tools/shared/mkmapgrids/SCRIPgrid_3x3mi
n_Indonesia_nomask_c170628.nc --res 3x3min_Indonesia --gridtype regional
```

```
./mkmapdata-yfan.sh --gridfile
~/cesm_input/lnd/clm2/mappingdata/grids/SCRIPgrid_1x1min_Sumatra_nomask_c
170622.nc --res 1x1min_Sumatra --gridtype regional
```

#above did not work, there are frac_dst > 1 at some latitudes in the output mapping file, not accepted by mksurfdata_map (see solution below)

#for the specific regional grid, set additional arguments --src_regional --dst_regional (according to Robert Oehmke robert.oehmke@noaa.gov, Y.Fan 2017)

```
      cmd="$mpirun $MY_ESMF_REGRID --ignore_unmapped -s ${INGRID[nfile]} --
src_regional --dst_regional "

      #cmd="$mpirun $MY_ESMF_REGRID --ignore_unmapped -s ${INGRID[nfile]} "
```

./mkmapdata-regional.sh --gridfile ../mkmapgrids/SCRIPgrid_3x3min_Indonesia_nomask_c170628.nc --res 3x3min_Indonesia --gridtype regional

#Run the modified mkmapdata-regional.sh tool **only for** regional HYDRO-1km grid **SCRIPgrid_1km_nomask_AU.c130227.nc** and the Indonesia land cover type grid **SCRIPgrid_3x3min_Landsat_Indonesia_c170628.nc**. The ESMF_REGRID tool will then be able to identify that they are regional grids as below in red. For other global source grids, use mkmapdata-Indonesia.sh to make maps.

```
      Starting weight generation with these inputs:
         Source File:
/usr/users/yfan1/cesm_input/lnd/clm2/mappingdata/grids/SCRIPgrid_3x3min_Landsat_Indonesia
_c170628.nc
         Destination File: ../mkmapgrids/SCRIPgrid_3x3min_Indonesia_nomask_c170628.nc
         Weight File: map_3x3min_Landsat_to_3x3min_Indonesia_nomask_aave_da_c170714.nc
         Source File is in SCRIP format
         Source Grid is a regional grid
         Source Grid is a logically rectangular grid
         Destination File is in SCRIP format
         Destination Grid is a regional grid
         Destination Grid is a logically rectangular grid
         Regrid Method: conserve
         Pole option: NONE
         Ignore unmapped destination points
         Output weight file in 64bit offset NetCDF file format
```

```
Completed weight generation successfully.
```

For some large datasets, e.g. SCRIPgrid_3minx3min_LandScan2004, SCRIPgrid_3minx3min_GLOBE-Gardner-mergeGIS, if core memory is not sufficient, you can run the EMSF tool directly, example：

```
mpirun -np 24 /home-
gk/users/nscc1726_SXY/local/esmf/bin/binO/Linux.intel.64.mpich2.default/ESMF_RegridWeight
Gen --ignore_unmapped -s /home-
gk/users/nscc1726_SXY/linshan/cesm1_2_2/inputdata/lnd/clm2/mappingdata/grids/SCRIPgrid_3m
inx3min_LandScan2004_c120517.nc -d /home-
gk/users/nscc1726_SXY/linshan/cesm1_2_2/inputdata/lnd/region/nomask/SCRIP_180_201_nomask.
nc -m conserve -w /home-
gk/users/nscc1726_SXY/linshan/cesm1_2_2/models/lnd/clm/tools/shared/mkmapdata/map_3x3min_
LandScan2004_to_180_201_nomask_aave_da_c161019.nc --dst_regional --src_type SCRIP
regional --dst_type SCRIP regional --64bit_offset >& linshan3.log
```

On hexagon

Use qsub to request resources (nodes, walltime etc) and then submit the job with aprun. The compute nodes on Hexagon have in total 32GB RAM per a whole node with 32 CPU cores (1GB per CPU core, so mppmem <=1Gb)

#Option 1: interactive session, first request job resource:

```
qsub -l mppwidth=128,walltime=0:59:00 -A nn2345k -I
```

Then submit the job with aprun (run on 4 nodes, 2 cores per node, total 8 processes, each 16Gb memory) :

```
aprun -n 8 -N 2 ./mkmapdata.sh --gridfile
$CSMDATA/lnd/clm2/mappingdata/grids/SCRIPgrid_1x1pt_Endalen_nomask_c161119.nc --
res 1x1pt_Endalen --gridtype regional
```

#Or one node with 16Gb memory is sufficient (run on 1 node (-n=1), 1 cores per node (-N=1), total 1 process, with 16Gb memory):

```
qsub -l mppwidth=32,walltime=0:59:00 -A nn2345k -I

cd
/work/shared/noresm/point_inputdata/NorESM/components/clm/tools/shared/mkmapdata
_yfan

export CSMDATA=/work/shared/noresm/point_inputdata

export ESMFBIN_PATH=/work/apps/esmf/6.3.0rp1-dso-
gnu/bin/binO/Unicos.gfortran.64.mpi.default/

aprun -n 1 -N 1 ./mkmapdata.sh --gridfile
/work/shared/noresm/point_inputdata/lnd/clm2/mappingdata/grids/SCRIPgrid_3x3min_
Indonesia_nomask_c170628.nc --res 3x3min_Indonesia --gridtype regional
```

#Option 2: silent mode with job script: qsub mpijob.regional

```
cat << EOF > mpijob.regional
#!/bin/bash

#PBS -N "mpijob.regional"
#PBS -l walltime=02:29:00
#PBS -l mppwidth=128
```

```
#PBS -A nn2345k
##PBS -l mppnppn=8
#PBS -o mpi.out.region
#PBS -e mpi.err.region
#PBS -m abe
#PBS -M yuanchao.fan@uni.no

# use /work directory
cd /work/shared/noresm/point_inputdata/NorESM/components/clm/tools/shared/mkmapdata_yfan
#cd /home/uni/yfa008/NorESM/components/clm/tools/shared/mkmapdata
#cannot run the tool in /home directory, otherwise ESMF_LogOpen error

export CSMDATA=/work/shared/noresm/point_inputdata
export ESMFBIN_PATH=/work/apps/esmf/6.3.0rp1-dso-
gnu/bin/binO/Unicos.gfortran.64.mpi.default/

aprun -n 8 -N 2 ./mkmapdata.sh --gridfile
/work/shared/noresm/point_inputdata/lnd/clm2/mappingdata/grids/SCRIPgrid_3x3min_Indonesia
_nomask_c170628.nc --res 3x3min_Indonesia --gridtype regional

EOF
```

#Job suspend after making one map. This is because the tool rmdups.ncl (called after RegridWeightGen) cannot return to main script mkmapdata.sh when it is submitted via aprun

Add command "quit" at the end of rmdups.ncl (called by mkmapdata.sh) still can not solve the problem!

#Copy the output map file to standard location
`/work/shared/noresm/point_inputdata/lnd/clm2/mappingdata/maps/3x3min_Indonesia`

The mapping files tell mksurfdata_map how to map between the output grid and the raw datasets that it uses as input. The output of mksurfdata_map is a surface dataset that you then use for running the model.

Step 5: Convert map of ocean to atm for use by DATM and CLM with gen_domain

```
       (See mapping/README for more help on doing this)

       - gen_domain uses the map from step 2 (or previously created CESM maps)

       Example:

        cd ../../../../mapping/src
        gmake
        cd ..
        setenv CDATE       090206
        setenv OCNGRIDNAME gx1v6
        setenv ATMGRIDNAME fv1.9x2.5
        setenv MAPFILE
$CSMDATA/cpl/cpl6/map_${OCNGRIDNAME}_to_${ATMGRIDNAME}_aave_da_${CDATE}.nc
        ./gen_domain -m $MAPFILE -o $OCNGRIDNAME -l $ATMGRIDNAME
NOTE: $ATMGRID is the same as LNDGRID except that ATMGRID is unmasked

        Normally for I compsets running CLM only you will discard the ocean
domain file, and only use the atmosphere domain file for datm and as the
```

```
fatmlndfrc file for CLM. Output domain files will be named according to the
input OCN/LND gridnames.

README: Computation of land mask and cell fraction

    This code adds "cell fraction" data to the output domain files.
    The "cell fraction" indicates how much of each grid cell is active.
    Typically ocean models do not have fractional cells (their fraction
    is either 0 or 1), where as land models do have fractional cells.
    This code generates domain files where gridocn has fractions of either
    0 or 1 (for grid cells that are masked or unmasked, respectively) and
    gridlnd has fractions that represent the complement of gridocn fraction
    data (lnd is totally opposite to ocn), as computed by the input mapping
data.  Thus gridocn is intended
    to be an ocean domain file and gridlnd is intended to be the complementary
    land domain file.  Related, the input mapping data, filemap, should be
    a conservative mapping: ocean -> atmosphere.

    Computed land fractions will be truncated into the range [0,1]
    after the min/max land fraction values have been documented.
    Computed land fractions that are less than fminval will be truncated to 0
    to avoid active land cells with tiny land fractions.

    The input atmosphere grid is assumed to be unmasked (global) and the land
    and atmosphere grids are assumed to be identical, except for cell fractions
    and masks.  Land cells whose fraction is zero will have land mask = 0.
```

For GWDG:

```
module load intel/compiler/64
module load intel/mpi/64/2017/2.174

cd ~/post4.5Crop_yfan/tools/mapping/gen_domain_files/
#for regional grid containing ocean
./gen_domain -m
~/post4.5Crop_yfan/models/lnd/clm/tools/region/mapfile/map_Sumatra_3x3min_
navy_to_Sumatra_3x3min_nomask_aave_da.nc -o navy -l 3x3min_Sumatra
```

*****NOTE: the gen_domain tool makes two opposite domains for lnd and ocn (wrong!)** the output domain.lnd.3x3min_Sumatra_navy.170928.nc should be exchanged with domain.ocn.3x3min_Sumatra_navy.170928.nc, where domain.ocn.3x3min_Sumatra_navy.170928.nc has actually the correct mask for lnd (=1) and ocean(=0) as compared with global lnd.domain files! Ocean cells should have a mask=0, lnd cells have a mask=1. Ocn and lnd frac always sum to 1 (complementary) at any location. Check the mask and frac with ncview.

The output of gen_domain is an atmosphere domain file "domain.lnd (actually use domain.ocn which has the correct mask for lnd)" used by both CLM and DATM for I compsets and an ocean domain file that is ignored.

The noocean_to_nomask map only works for point runs which assumes no ocean. When this map is used to create domain, the output domain does not have land/sea mask. All mask==1, which will include both land and ocean grid cells in the run.

Also possible to use the following map (based on Landsat converted grid made with R) to make the domain file:

```
./gen_domain -m
~/cesm_input/lnd/clm2/mappingdata/maps/3x3min_Indonesia/map_3x3min_Landsat
_to_3x3min_Indonesia_nomask_aave_da_c170714.nc -l 3x3min_Indonesia -o navy
```

#Put the lnd domain files at correct locations

```
cp domain.lnd.3x3min_Indonesia_nomask.170714.nc
~/cesm_input/atm/datm7/domain.clm/domain.lnd.3x3min_Indonesia_nomask.nc
cp domain.lnd.3x3min_Indonesia_navy.170714.nc
~/cesm_input/atm/datm7/domain.clm/domain.lnd.3x3min_Indonesia_navy.nc

cp domain.lnd.3x3min_Indonesia_nomask.170714.nc
~/cesm_input/share/domains/domain.clm/domain.lnd.3x3min_Indonesia_nomask.nc
cp domain.lnd.3x3min_Indonesia_navy.170714.nc
~/cesm_input/share/domains/domain.clm/domain.lnd.3x3min_Indonesia_navy.nc
```

On Hexagon: qsub gendomain.pbs

```
cat << EOF > gendomain.pbs
#!/bin/bash
#PBS -N "job.gendomain"
#PBS -l walltime=00:10:00
#PBS -l mppwidth=1
#PBS -A nn2345k
#PBS -m abe
#PBS -M yuanchao.fan@uni.no

cd
/work/shared/noresm/point_inputdata/NorESM/cime/tools/mapping/gen_domain_files

aprun -n 1 -N 1 ./gen_domain -m
/work/shared/noresm/point_inputdata/NorESM/components/clm/tools/shared/mkm
apgrids/map_3x3min_Indonesia_noocean_to_3x3min_Indonesia_nomask_aave_da_17
0922.nc -l 3x3min_Indonesia -o navy

aprun -n 1 -N 1 ./gen_domain -m
/work/shared/noresm/point_inputdata/NorESM/components/clm/tools/shared/mkm
apgrids/map_3x3min_Indonesia_noocean_to_3x3min_Indonesia_nomask_aave_da_17
0922.nc -l 3x3min_Indonesia -o nomask

EOF
```

"/work/shared/noresm/point_inputdata/share/domains/domain.clm/" and
"/work/shared/noresm/point_inputdata/atm/datm7/domain.clm"

Step 6: Use mksurfdata_map to create surface datasets.

Use the new regional grid, mapping files and new raw data file to create fsurfdata

On Fram:

#re-compile the mksurfdata_map tool

```
cd /cluster/shared/noresm/site_inputdata/tools/mksurfdata_map/
cd src
export LIB_NETCDF=/cluster/software/netCDF/4.4.1.1-intel-2017a-HDF5-1.8.18/lib64
export INC_NETCDF=/cluster/software/netCDF/4.4.1.1-intel-2017a-HDF5-1.8.18/include
export USER_FC=mpiifort
export USER_CC=mpiicc
module load intel/2017a

rm mksurfdata.o
```

make

cd /cluster/shared/noresm/site_inputdata/tools/mksurfdata_map/
cd src
export LIB_NETCDF=/cluster/software/netCDF-Fortran/4.4.3-intel-2016a/lib
export INC_NETCDF=/cluster/software/netCDF-Fortran/4.4.3-intel-2016a/include
export USER_FC=mpiifort
export USER_CC=mpiicc
module load intel/2017a
#must load 2017a version of intel compiler

rm mksurfdata.o
make

On GWDG
#copy a latest CLM5 version of mksurfdata_map from noresm-dev

```
cd ~/post4.5Crop_yfan/models/lnd/clm/tools/clm4_5/mksurfdata_map_clm5
```

#first make the tool

```
cd src
export LIB_NETCDF=/usr/product/applsw/netcdf-4.3.0/lib
export INC_NETCDF=/usr/product/applsw/netcdf-4.3.0/include
export USER_FC=mpiifort
export USER_CC=mpiicc
module load intel/compiler/64
module load intel/mpi/64
#(cluster updated 2015.10, don't specify a version)
>make or gmake
```

#edit namelist and create surface map

```
module load intel/compiler/64
module load intel/mpi/64
cd ~/post4.5Crop_yfan/models/lnd/clm/tools/clm4_5/mksurfdata_map_noresm
  ./mksurfdata_map < namelist-3x3min-Indonesia
```

#NOTE: CLM5 requires new variables GLACIER_REGION and zbedrock (soil depth), and the namelist file needs to add new inputs: fglacierregion and fsoildepth and remove flndtopo (map and mksrf). Otherwise, there will be Error: namelist input resulted in error code
##The latest CLM5 version works on Fram: /cluster/projects/nn2345k/yfan/ctsm-dev/tools_clm5/mksurfdata_map
#The NorESM-work version (2016.07) also works
rsync -arv ~/NorESM-work/components/clm/tools/clm4_5/mksurfdata_map/ yfan1@transfer.gwdg.de:/usr/users/yfan1/post4.5Crop_yfan/models/lnd/clm/tools/clm4_5/mksurfdata_map_noresm
Make a copy to GWDG and use directly. No need to rebuild!

#if any raw mksrf file is > 8GB, use job submission script:

```
cat << EOF > job.mksurfdata.sh
#!/bin/bash

module load intel/compiler/64
module load intel/mpi/64
```

```
      cd ~/post4.5Crop_yfan/models/lnd/clm/tools/clm4_5/mksurfdata_map_noresm
             ./mksurfdata_map < namelist-3x3min-Indonesia-lai0.25
      EOF
#interactive job submission (use sh script, cannot run executable directly)
      >bsub -ISs -q fat -n 1 -W 00:30 ./job.mksurfdata.sh
      >bsub -ISs -q fat -n 64 -R 'span[hosts=1]' -R np64 -W
      00:30 ./job.mksurfdata.sh
```

Notable changes in namelist-1x1min-Sumatra

```
      mksrf_fgrid      =
      '/usr/users/yfan1/cesm_input/lnd/clm2/mappingdata/maps/1x1min_Sumatra/map_
      1x1min_Landsat_to_1x1min_Sumatra_nomask_aave_da_c170622.nc'
       map_fpft         =
      '/usr/users/yfan1/cesm_input/lnd/clm2/mappingdata/maps/1x1min_Sumatra/map_
      1x1min_Landsat_to_1x1min_Sumatra_nomask_aave_da_c170622.nc'

      mksrf_fvegtyp  =
      '/usr/users/yfan1/cesm_input/lnd/clm2/rawdata/pftlandusedyn.1x1min.Indones
      ia.c170615/mksrf_Sumatra_LC2010_1x1min_c170615.nc'
```

Developer's guide on adding mapping files

On 12/07/2017
Hi Yuanchao,

Thank you for the very complete description of your problem. From the description it sounds like both of your grids are regional. Do you have the --src_regional and --dst_regional flags set when you run the regrid application? If not, give that a try and see if it fixes the problem.

If that doesn't help, would you please send me the full command (including all arguments) that you're using to run ESMF_RegridWeightGen as well as the output.

Thanks,

- Bob

Thu 29/06/2017
In theory, frac_dst should never be greater than 1. But if I remember correctly, mapping tools sometimes result in frac_dst > 1 to correct for small errors that can arise as part of mapping. But I don't think I've ever seen errors this big.

A few ideas:

(1) If you have NCL installed, it comes with a tool called scrip_check_input (which should be installed in the bin directory of ncl). Try running your scrip files through that tool:
scrip_check_input filename.nc

(2) Did you check the mask on the 1x1km grid to make sure it is always 0 or 1?

(3) I assume you're making the mapping files with ESMF. What version of ESMF are you using for this? If an old one, you could try redoing this with a newer version.

(4) If none of this shines light on the problem, you could contact Bob Oehmke (robert.oehmke@noaa.gov), who is the main developer of the ESMF remapping tool.

(5) A final solution would be to remove the error checks in mksurfdata_map so that it can proceed with these large frac_dst values – but in that case you should check the resulting output maps to make sure they look reasonable, because I'm not sure what the implications are of having these large frac_dst values.

Sorry I can't be of more immediate help.

Bill Sacks

```
II. Adding mapping files for a raw data file with a new grid / landmask
=======================================================================
```

If your raw data file is on a new grid, or just has a new landmask on an existing grid, you will need to perform a number of additional steps, as laid out here.

- First, move your data file to the inputdata directory and give it its final name. (This will ensure that the appropriate metadata is put in the SCRIP grid file.)

- Make a scrip grid file from your data file using mkmapgrids, and move it to the inputdata directory

- Add a scripgriddata entry for the new scrip grid file in bld/namelist_files/namelist_defaults_clm4_5_tools.xml

- If necessary, add other entries in namelist_defaults_clm4_5_tools.xml giving information about your scrip grid file:
    - If this is a high resolution grid (e.g., 3min or higher), add a scripgriddata_lrgfile_needed entry, saying we need 64bit_offset (or netcdf4) support for mapping files made with this scrip grid file
    - If the grid file is in UGRID format rather than SCRIP grid format, add scripgriddata_type and scripgriddata_meshname entries. If you don't know what I'm talking about, then your grid file is in SCRIP format and you can ignore this.

- If necessary, add new grid and/or landmask to lists of valid values for hgrid, res and lmask in bld/namelist_files/namelist_definition_clm4_5.xml
    - Note that a new resolution currently needs to be added to both the hgrid and res lists of valid values, although in the future this should probably be changed so that these raw data grids just appear in hgrid

- Add the new grid-landmask combo to the 'mapgrids' list in bld/namelist_files/checkmapfiles.ncl

- Add the new grid-landmask combo to the 'grids' list in tools/shared/mkmapdata/mkmapdata.sh (in the clm4_5 branch of the conditional)

- Make mapping files, from tools/shared/mkmapdata
    - Modify mkmapdata.sh:
        - edit the grids list so it only contains your new grid
    - Modify regridbatch.sh as desired, e.g.:
        - project number
        - number of processors (BSUB -n line, span, and the regrid_num_proc setting)
        - wall-clock limit
        - if ESMFBIN_PATH is in your environment, you may want to unset it; this can be important to allow mkmapdata.sh choose a different executable for mpi vs serial runs
        - if you renamed the mkmapdata.sh script, be sure to call the renamed script at the bottom of regridbatch.sh
    - Submit regridbatch.sh

- When mapping files have all been created, run createXMLEntries.pl from tools/shared/mkmapdata (usage: just run the script with no arguments)

- Cut and paste the xml entries from mapping_entries.txt (created by createXMLEntries.pl) into bld/namelist_files/namelist_defaults_clm4_5.xml, in the correct locations

- Move mapping files to correct location, either using mv_cmds.sh created by createXMLEntries.pl, or using tools/shared/mkmapdata/mvNimport.sh.

Prepare other regional dataset from global dataset if necessary

```
See models/lnd/clm/tools/README                                    Oct/13/2012

This README provide detailed explanation on the steps of using CLM tools for
analysis of CLM history files -- or for creation or modification of CLM input
files.
```

After creation of regional grid, domain and mapping files for the land, it also needs to create regional grid for the atmosphere component to use the forcing data, or use global atm grid with fine mesh to couple the finer resolution lnd gird to courser resolution atm grid.

```
V.  Notes on which input datasets are needed for CLM

      Fine mesh is when you are running CLM at a finer resolution than you are
      coupling it to the atmosphere. So the course resolution is the "atm" grid
      and the fine mesh is the "lnd" grid.

      global or regional/single-point grids (NO fine mesh)
        - need fsurdata and fatmlndfrc
      global grids (with fine mesh)
        - assume lat/lon grids only
          (can't use unstructured grids for either atm or lnd grid)
        - need fsurdata, fatmgrid, fatmlndfrc and fatmtopo
          (use deprecated mkgriddata to get fatmgrid, fatmlndfrc and fatmtopo)

    fsurdata ---- from mksurfdata_map in step (III.7)
    fatmlndfrc -- use the domain.lnd file from gen_domain in step (III.6)
```

Create missing input dataset for Point/Regional mode using getregional_datasets.pl

- Step 1: download pre-compiled NCL tool and setenv

```
tar -xvzf ncl_ncarg-6.1.2.Linux_RHEL5.6_x86_64_nodap_gcc412.tar.gz
#and copy the extracted bin/include/lib folders into local/ncl_ncarg-6.1.2
setenv NCARG_ROOT /usr/users/yfan1/local/ncl_ncarg-6.1.2
setenv PATH $NCARG_ROOT/bin:$PATH
```

- Step 2: install NCO

```
cd /usr/users/yfan1/downloads/nco-4.3.0
setenv NETCDF4_ROOT /usr/users/yfan1/local/netcdf-gwdg
./configure --prefix=/usr/users/yfan1/local --enable-netcdf4
make
make check
make install
```

- Step 3: Extract out regional datasets from global datasets and add to the standard locations for the specified area. By default, it extracts data for SIM_YEAR=2000, SIM_YEAR_RANGE=constant

```
#Establish a "user dataset identifier name" and own dataset location
setenv MYDATAID 1x1pt_Bariri_global
```

```
setenv MYCSMDATA /usr/users/yfan1/cesm_input
cd ~/cesm1_0/models/lnd/clm/tools/ncl_scripts

# (-1 to -3), (120 to 121) is about a 1-degree pixel from the f19_g16 grid surrounding the Bariri site

#NOTE: have to make a new folder "/domain.clm" under "cesm_input/atm/datm7", otherwise get error messages running the following code

./getregional_datasets.pl -ne -1,121 -sw -3,120 \
-id $MYDATAID -mycsmdata $MYCSMDATA
```

Create LUC dataset flanduse_timeseries/fdynpft for 20C transient simulations

#Use mksurfdata_map tool to create the file flanduse_timeseries, which contains LUC from C3 grass/C4 crop to forest

Method 1 (using pft_frc/pft_idx override values from single_point_dynpft_files):

/cluster/shared/noresm/site_inputdata/ctsm_clm5.0/tools/mksurfdata_map_clm5/fdynpft_1x1pt_DUKE_16pfts_CMIP6_c180823.namelist

NOTE: Set pft_frc = 100 /n pft_idx = 13 in the namelist, so that pft fraction override is on, then mksurfdata_map will interpret string as PFT and harvesting override values from below .txt file

And LUC is specified in file: `mksrf_fdynuse =single_point_dynpft_files/landuse_timeseries_1x1pt_DUKE_simyr1850-2000.txt`

```
NOTE: Don't change the format and space of this txt file, the mksurfdat.F90 module reads in each line of the file to get pft_f, pft_i, harv and graz values to override the pft fraction and harvesting for each year (year "1850" first number starts from column or line position 197/198), by calling functions mkpft_parse_oride and mkharvest_parse_oride
```

Run: `sbatch mpijob.mksurfdata.DUKE.fdynpft`

error: year for harvest not equal to year for PFT files (reason: pft_frc is not set on the namelist and mksurfdat.F90 interpret string as a filename, which is the case when the txt list are paths to netcdf files)

error: end-of-file during read, unit -5, file Internal List-Directed Read (after setting pft_frc overriding mode, always have this error even with the example list of smallvilleIA. The resulting fdyndat "landuse_timeseries_1x1pt_DUKE_simyr1850-2000_c180823.nc" has no values for PCT_NAT_CFT / PCT_CFT. Not solved!!)

One possible solution: use NCL to fill site history values to the above empty file "landuse_timeseries_1x1pt_DUKE_simyr1850-2000_c180823.nc"

Method 2: first use a list of global landuse netcdf files for mksrf_fdynuse to generate a fdyndat file. Then modify the PCT_NAT_CFT / PCT_CFT values in the output file fdyndat with NCL or R (see previous example file for Jambi: surfdata.pftdyn_1x1pt_Jambi_hist_simyr2000-2002_c160121.nc, which is post-processed with "modify pftdyn data.R")

cd /cluster/shared/noresm/site_inputdata/ctsm_clm5.0/tools/mksurfdata_map_clm5

Use namelist: landuse_timeseries_1x1pt_DUKE_simyr1800-2015.namelist which use netcdf file list: mksrf_fdynuse = 'landuse_timeseries_hist_LUH2_simyr1800-2015.txt' (note: must remove two lines in the namelist: pft_frc = 100, pft_idx = 13, otherwise the tool will follow the pft_frc overriding routine in Method 1)

NOTE: the text file "landuse_timeseries_hist_ LUH2_simyr1800-2015.txt" contains a list of paths to global land use/pft data files:

/cluster/shared/noresm/inputdata/lnd/clm2/rawdata/pftcftlandusedynharv.0.25x0.25.MODIS.simyr1850-2015.c170412/mksrf_landuse_histclm50_LUH2_*

Two duplicated lines for each year, because one path is used by mkpftMod.F90 to make pft_frc and another used by mkharvestMod.F90 to make timber harvest and grazing allocations.

After replacing the default glade path, the spaces before year 1*** must be adjusted so that, `the first digit of year "1850" starts from column or line position 197/198`.

Otherwise, the mksurfdata_map tool will not be able to read the year correctly!

Finally run the tool:

./mksurfdata_map < landuse_timeseries_1x1pt_DUKE_simyr1850-2015.namelist

Successfully created surface dataset:

landuse_timeseries_1x1pt_DUKE_simyr1850-2015_c180823.nc

Use R or NCL or NCO to modify variables PCT_CROP, PCT_NAT_PFT, PCT_CFT in the above file "landuse_timeseries_1x1pt_DUKE_simyr1850-2015_c180823.nc" (from method 2), according to DUKE and ORNL site history.

Set HARVEST_VH1 (harvest from primary forest), HARVEST_VH2 (harvest from primary non forest), HARVEST_SH1 (harvest from secondary mature-forest), HARVEST_SH2 (harvest from secondary young-forest), HARVEST_SH3 (harvest from secondary non-forest) and GRAZING (grazing of herbacous pfts) all to 0

See NCL scripts ~/scripts_yfan/modf_dynpft_DUKE1800-2015.ncl

USE finally product: flanduse_timeseries = landuse_timeseries_1x1pt_DUKE_simyr1800-2015_c181005ed.nc / landuse_timeseries_1x1pt_ORNL_simyr1750-2015_c181005ed.nc

Follow the land use history from 1800 to 1982 (DUKE) or 1750 to 1987 (ORNL) described in Walker_et_al-2014-JGR-Biogeoscience for 20Ctransient simulation.

The model will automatically change PFTs and let the new land cover type develop. But it might need to take additional years for the BDT or NET forests to grow sufficiently large before the FACE experimental period starts. Check the aboveground vegetation carbon (totvegc-leafc-rootc) compared to aboveground woody biomass values at the start of experiment.

Create CMIP6 SSP-RCP future landuse.timeseries data

mv /cluster/shared/noresm/site_inputdata/ctsm_clm5.0/tools /cluster/projects/nn2345k/yfan/ctsm-dev/tools_clm5
NOTE: the clm tools need to be placed under the ctsm-dev directory so that the tools can have access to the ctsm-dev/bld scripts.

cd /cluster/projects/nn2345k/yfan/ctsm-dev/tools_clm5/mksurfdata_map

First create "namelist-1.9x2.5_SSP2-4.5_78pfts_CMIP6" according to global information of "landuse.timeseries_1.9x2.5_SSP5-8.5_78pfts_CMIP6_simyr1850-2100_c181204.nc" which tells the raw data files and mapping files used for each variable.

Then create a txt file "landuse_timeseries_SSP2-4.5_78pfts_simyr1850-2100.txt" that contains two duplicated lines for each year of raw data from 1850-2100, because one path is used by mkpftMod.F90 to make pft_frc and another used by mkharvestMod.F90 to make timber harvest and grazing allocations.

After replacing the default glade path to fram path, the spaces before years 18** or 20** must be adjusted so that the first digit of year "1850" starts from column or line position 197/198.

Then create the landuse.timeseries file using
./mksurfdata_map < namelist-1.9x2.5_SSP2-4.5_78pfts_CMIP6
Or submit via script: `sbatch mpijob.mkmap_SSP2-4.5`

```
cat << EOF > mpijob.mkmap_SSP2-4.5
      #!/bin/bash

      #SBATCH  --job-name=job.mksurf
      #SBATCH  --output=job.out
      #SBATCH  --exclusive
      #SBATCH  --export=ALL
      #SBATCH  --qos=preproc
      #SBATCH  --nodes=1
      ##SBATCH  --ntasks=10
      #SBATCH  --ntasks=1
      #SBATCH  --cpus-per-task=1
      #SBATCH  --account=nn2345k
      ##SBATCH  --mail-user=yfa008@uib.no
      #SBATCH  --mail-type=END
      #SBATCH  --time 00:30:00


      module load intel/2017a
      #module load netCDF-Fortran/4.4.3-intel-2016a
      #module load iimpi/2017a

      cd /cluster/projects/nn2345k/yfan/ctsm-dev/tools_clm5/mksurfdata_map

      ## Run the application
      #mpirun ./mksurfdata_map < namelist-1.9x2.5_SSP2-4.5_78pfts_CMIP6
      srun ./mksurfdata_map < namelist-1.9x2.5_SSP2-4.5_78pfts_CMIP6

EOF

ERROR:
netcdf error from mkharvest_init rcode =          -36  error =
 NetCDF: Invalid argument
```

```
Reason: the CLM5 mksurfdata_map tool requires a new input file for harvest:
mksrf_fhrvtyp  =
'/cluster/shared/noresm/inputdata/lnd/clm2/rawdata/pftcftdynharv.0.25x0.25.LUH2.
histsimyr1850-2015.c170629/mksrf_landuse_histclm50_LUH2_2000.c170629.nc'
Which is the same as "mksrf_fvegtyp"
```

After solving this, successfully made the output file for RCP4.5 simulation:

```
landuse.timeseries_1.9x2.5_SSP2-4.5_78pfts_CMIP6_simyr1850-2100_c190116.nc
```