Reynold's Creek Experimental Watershed: Single Point Simulations

Katie Murenbeeld

07/31/2020

Contents

1. Introduction . . . 1
2. File Structure and Cloning CTSM . . . 1
3. Set Up for Point Simulations (PTCLM) . . . 2
4. Accelerated Spin Up . . . 3
6. Setting Up and Running a Case . . . 5
7. Model Outputs . . . 5
8. Useful References . . . 5

1. Introduction

Reynold's Creek Experimental Watershed (RCEW)… (a little about the location and CZO).

About the Community Earth Systems Model (CESM), Community Terrestrial Systems Model (CTSM) and Community Land Model (CLM) explain how different although terms are used almost interchangably. Model simulations. Include a link to the escomp CLM5.0 and the CLM4.5 User's Guides.

The purpose of this technical note is to familiarize a LEAF lab user with setting up and running a single point simulation at RCEW with meterological data from one of the AmeriFlux towers on the site. In this document, I am assuming that you have a small or large allocation on the National Center for Atmospheric Research's (NCAR) supercomputer Cheyenne. At this time CLM is not ported on the high performance computer (HPC) R2 at Boise State, but this is a goal to complete soon.

2. File Structure and Cloning CTSM

2.1 File Structure

If you don't have a git working directory set up already on Cheyenne, do so now. It is a best practice to keep projects in their own work directory within this git working directory.

```
cd /glade/work/$USER/ # go to your work directory on Glade
mkdir git             # create a new folder that will be the parent for future projects using CLM/CTSM
cd git                # go to the newly created git folder
```

2.2 Cloning CTSM

From your newly created git folder, clone the CLM/CTSM code. The following code will point to the latest CTSM release:

```
# Check out the latest release branch and create a new branch for your project

git clone -b release-clm5.0 https://github.com/ESCOMP/CTSM.git ctsm_rcew
cd ctsm_rcew   # A new directory was created for your project
git branch   # Check to see the branches you have now created. There should be a master (or main) as we
```

```
# If no new branch, create your own.
git checkout -b ctsm_rcew
ls   # Explore the contents of the new directory
```

Notice that there is a folder within your project direcotry named manage_externals. This contains a very important script, checkout_externals, which downloads all of the components required for building and running CTSM. These include: CLM (land model component), CISM (sea-ice component), RTM (river routing component), MOSART (river routing component), CIME (contains scripts and tools for running CESM), CMEPS (Earth prediction option of CIME), FATES (dynamic vegetation option of CLM), and PTCLM (point simulation option of CLM). More information about checkout_externals can be found in the README_EXTERNALS.rst file in your project folder.

To checkout the externals:

```
./manage_externals/checkout_externals
```

It is also a good idea to create separate folders for different projects in your git directory. You can either re-clone ctsm for each proect or do a recursive copy of another project.

For example, I have several projects that live in my git folder.

```
cd /glade/work/katiem/git
ls
> ctsm  ctsm_py  ctsm_sits ctsm_rcew
```

3. Set Up for Point Simulations (PTCLM)

The **P**oin**T** **CLM** runs through the same steps for creating single point domain and surface data files as in section 2. PTCLM is a simpler way to create these files specifically for tower sites. The file PTCLMmkdata runs the tools to get datasets set up, and copies them to a location you can use, including the changes needed for a case to use the dataset with namelist and XML changes. There are a few steps you will need to take before you can run the PTCLMmkdata script.

First, you will need to add tower site specific data to your location of interest. …

Once again load the appropriate modules in the correct order (same as below). **NOTE** Depending on if you use miniconda you may or may mot need to load all of the modules if they are already in the miniconda environment. For example, here I do not *module load python/2* because I will activate a python 2 miniconda environment later.

```
module load ncarenv/1.3
module load intel/17.0.1
module load ncarcompilers/0.5.0
module load nco/4.7.9
module load netcdf/4.7.3
module load ncl/6.6.2 (H)
module load mpt/2.22
module load esmf_libs/7.1.0r (H)
module load ncview/2.1.7
```

Change directory to the PTCLM folder. From there execute the buildtools script. Buildtools will…

```
CTSMROOT=/glade/work/katiem/git/ctsm_rcew
cd $CTSMROOT/tools/PTCLM
./buildtools
```

There is a known issue within the PTCLMmkdata code which will need to be addressed before you can continue on with the process. Use the text editor of your choice to open PTCLMmkdata. In line 552 you will need to remove the "export" command.

```
- cmd = "export REGRID_PROC=1; "+mkmapdat_dir+"/mkmapdata.sh --gridfile "+scripgridfile+" --res "+clmres
+ cmd = mkmapdat_dir+"/mkmapdata.sh --gridfile "+scripgridfile+" --res "+clmres+" --gridtype regional -
```

Once all of the modules are loaded, the tools are built, and the PTCLMmkdata script is updated, open up a qinteractive session in order to complete the map generation and file creation process. The process can take several hours, so I suggest setting the walltime to 6 hours.

```
qinteractive -X -l walltime=06:00:00
```

Load a python 2 compatible miniconda environment (optional).

```
conda activate python2
```

After entering the qinteractive environment, you will need to reset environmental variables.

```
CSMDATA=/glade/p/cesm/cseg/inputdata
CTSMROOT=/glade/work/katiem/git/ctsm_rcew
SITE=US-Rwe
```

Next, run the PTCLMsublist, which will…

```
./PTCLMsublist -l $SITE -d $CSMDATA -o --verbose --account=UBOI0003 --mach=cheyenne
```

PTCLMsublist will output a command to copy and paste into the terminal. It will look something like this…

```
./PTCLMmkdata -s US-Rwe -d /glade/p/cesm/cseg/inputdata
```

What is this command doing? …

Again, this process can take a long time. You may need to run this command multiple times. However, once the mapping files are created they will be skipped over if you have to rerun the PTCLMmkdata command.

NOTE Make sure to add in where to find the newly created files.

4. Accelerated Spin Up

Something something, working with soils, getting to a steady state. Need to do a spin up.

Within this section, also some of the (not so basic) basics for creating and submitting a CLM "case", or simulation.

Make sure you are in your CTSMROOT (the base directory where "cime" and "components" live). Then change directories to cime/scripts. Within the cime/scripts directory is the create_newcase script.

```
cd cime/scripts/

./create_newcase --compset 2000_DATM%GSWP3v1_CLM50%BGC_SICE_SOCN_SROF_SGLC_SWAV --res f09_g17 --case /gl
```

In order to create a new case there are 3 required arguments: compset, res, and case. The compset… The res… The case… The other arguments used here are run-unsupported and project. The run-unsupported … The project… If you are interested in the arguments please see the help documentation by running

```
./create_newcase --help
```

Once the new case is created, move into the newly created case directory.

```
cd /glade/work/katiem/SITS_cases/Rwe1_CLM50spinup_001
```

From here you will update some enviroenmental variables for the case using the command xmlchange. These variables are specific to the machine (computer). As a note. ./xmlquery… For a list and more information on the variables you can search for the variables on this webpage http://www.cesm.ucar.edu/models/cesm2/settings/2.1.0/drv_input.html. (I suggest bookmarking this page!)

```
./xmlchange MPILIB=mpi-serial
./xmlchange --file env_mach_pes.xml --id COST_PES --val 36
./xmlchange --file env_mach_pes.xml --id TOTALPES --val 1
./xmlchange --file env_mach_pes.xml --id NTASKS --val 1
./xmlchange --file env_mach_pes.xml --id NTASKS_PER_INST --val 1
./xmlchange --file env_mach_pes.xml --id ROOTPE --val 0
```

After you've updated the above variables, you are now ready to setup the case. When you setup the case. . . .

```
# Remember you are still in the case directory

./case.setup
```

Once your case is setup you will update many more variables using the ./xmlchange command.

```
./xmlchange --file env_run.xml --id CLM_FORCE_COLDSTART --val on
./xmlchange --file env_run.xml --id CLM_NML_USE_CASE --val 1850_control
./xmlchange --file env_run.xml --id DATM_CLMNCEP_YR_START --val 1901
./xmlchange --file env_run.xml --id DATM_CLMNCEP_YR_END --val 1920
./xmlchange --file env_run.xml --id DATM_PRESAERO --val clim_1850
./xmlchange --file env_run.xml --id CCSM_CO2_PPMV --val 284.7
./xmlchange --file env_run.xml --id STOP_OPTION --val nyears
./xmlchange --file env_run.xml --id RUN_REFDATE --val 0001-01-01
./xmlchange --file env_run.xml --id RUN_STARTDATE --val 0001-01-01

# Turn on the accelerated spinup
./xmlchange --file env_run.xml --id CLM_ACCELERATED_SPINUP --val on

# Set the total number of years to run.
./xmlchange --file env_run.xml --id STOP_N --val 500
# Set the interval (in years) where a restart file will be created.
./xmlchange --file env_run.xml --id REST_N --val 100
```

Next, you will point to the domain files you created in section 3.

```
./xmlchange --file env_run.xml --id ATM_DOMAIN_FILE --val domain.lnd.1x1pt_US-Rwe_navy.200721.nc
./xmlchange --file env_run.xml --id ATM_DOMAIN_PATH --val /glade/work/katiem/SITS_data
./xmlchange --file env_run.xml --id LND_DOMAIN_FILE --val domain.lnd.1x1pt_US-Rwe_navy.200721.nc
./xmlchange --file env_run.xml --id LND_DOMAIN_PATH --val /glade/work/katiem/SITS_data
```

Next, you will update the user_nl_clm (the CLM user namelist) file. In this file. . . You will set the surface data file to the surfdata file you created in section 3.

```
echo "fsurdat = '/glade/work/katiem/SITS_data/surfdata_1x1pt_US-Rwe_hist_16pfts_Irrig_CMIP6_simyr2000_c
echo "hist_mfilt = 20" >> user_nl_clm
echo "hist_nhtfrq = -8760" >> user_nl_clm

echo "hist_empty_htapes = .true." >> user_nl_clm
echo "hist_fincl1 = 'TOTECOSYSC', 'TOTECOSYSN', 'TOTSOMC', 'TOTSOMN', 'TOTVEGC', 'TOTVEGN', 'TLAI', 'GP
'H2OSNO'" >> user_nl_clm

echo "mapalgo = 'nn','nn','nn','nn','nn'" >> user_nl_datm
```

If you have your own climate or atmospheric forcing data to use (for example you want to use the atmospheric data from the flux tower at your site) you can now copy over the user_datm.streams files you created (see section X).

```
cp ../Rwe1_CLM50spinup_001/user_datm.streams.txt.* .
```

Double check the your namelist file (user_nl_clm) using your text editor of choice or the command.

```
vim user_nl_clm

## or...
cat user_nl_clm
```

Now, build and submit the spinup case!

```
qcmd --./case.build

## This may take a few minutes. When complete you should see:
## MODEL BUILD HAS FINISHED SUCCESSFULLY

./case.submit
```

To check on the progress you can see the jobs you've submitting using the qstat command.

```
qstat -u katiem  # the -u argument is for user, so really it is qstat -u $USER
```

You can also check your scratch or scratch archive to check on the progress of the spinup case.

```
cd /glade/scratch/$USER/archive/$CASE/
ls

# or...
cd /glade/scratch/$USER/$CASE/run
ls
```

6. Setting Up and Running a Case

7. Model Outputs

8. Useful References