CESM1.2 Tutorial: Basic Modifications
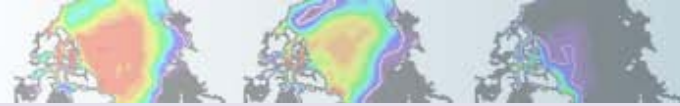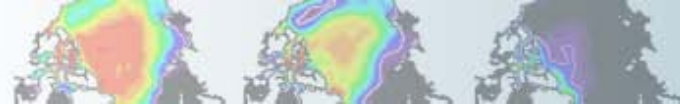
Christine A. Shields

August 13, 2012

CESM1 Tutorial: Basic Modifications: Review

1. We will use the CESM code located locally on yellowstone, no need to checkout or download any input data.

2. We will run with resolution T31_gx3v7.

3. Default scripts will **automatically** be configured for you using the code/script base prepared uniquely for this tutorial.

4. Please log into your assigned yellowstone (or caldera) node and do NOT compile unless you have a compile card. To make the tutorial run smoothly for all, we need to control and distribute all work evenly across yellowstone.

Tutorial Code and script base:

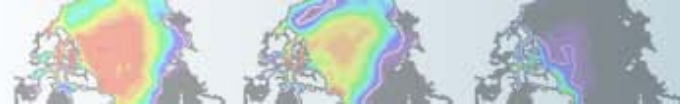# */glade/p/cesm/tutorial/cesm1_2_0.tutorial*

This tutorial contains step by step instructions applicable to the current release (cesm1.2) only.

For older releases, please see past tutorials.

New features will be labelled with

NEW

CESM1 Tutorial: Basic Modifications: Review: Creating a new case

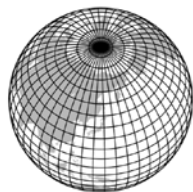**create_newcase** requires **4 arguments**

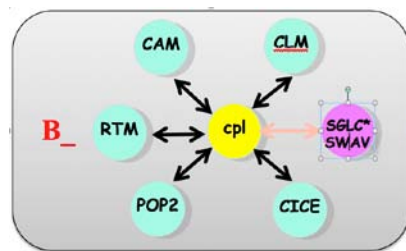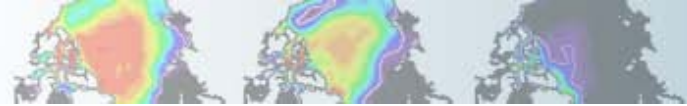| **What is the casename ?**	**Which resolution?**	**Which model configuration ? Which set of components ?**	**Which machine are you running on?**

case01	**T31_gx3v7**	**B_1850_CN**	**yellowstone**

create_newcase –case case01 –res T31_g37 -compset B_1850_CN -mach yellowstone

courtesy: c.hannay

New grid naming convention

Each model resolution can be specified by its alias, short name and long name.

Example of equivalent alias, short name and long name:

- **alias: T31_g37 (atm/lnd_ocn/ice)**
- **short name: T31_gx3v7**
- **long name = a%T31_l%T31_oi%gx3v7_r%r05_m%gx3v7_g%null_w%null**

NEW

atm **lnd** **ocn/ice** **river** **lnd mask** **lnd-ice** **wave**
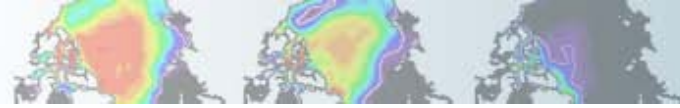
New compset naming convention

Each model compset can be specified by its alias, short name and long name.

Example of equivalent alias, short name and long name:

- **alias: B1850CN**
- **short name: B_1850_CN**
- **long name =** 1850_CAM4_CLM40%CN_CICE_POP2_RTM_SGLC_SWAV

NEW

Time Period **Model** **Physics submode** **"S"=stub/inactive**

CESM1 Tutorial: Basic Modifications: Review: Creating a new case

EXERCISE.0: Create and configure an out-of-the-box set of scripts called "b.day2.0" on yellowstone using T31_gx3v7 and compset B_1850_CN . Build and run the model.

Explanation of steps

1. Change directories, ("cd") to tutorial code base scripts directory.
2. View compset and grid choices.
3. Create initial scripts. (We will use the same "cases" subdirectory as day1).
4. "cd" to your casedir.
5. Invoke cesm_setup.
6. Build the model.
7. Submit a 5-day test.
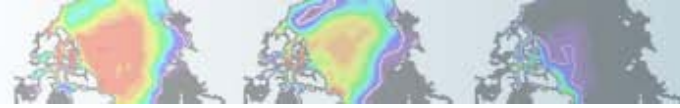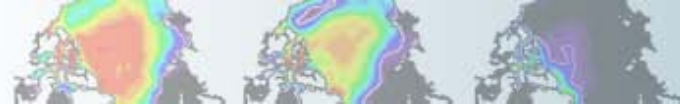8. "cd" to your runtime directory and explore

CESM1 Tutorial: Basic Modifications: Review: Creating a new case

EXERCISE.0: Create and configure an out-of-the-box set of scripts called "b.day2.0" on yellowstone using T31_gx3v7 and compset B_1850_CN. Build the model.

1. *cd   /glade/p/cesm/tutorial/cesm1_2_0.tutorial/scripts*
2. *./create_newcase –list compsets   (and –list grids)*
⭐3. *./create_newcase -case   ~/cases/b.day2.0   -res T31_gx3v7   -mach yellowstone   -compset B_1850_CN*
4. *cd ~/cases/b.day2.0*
⭐5. *./cesm_setup*
⭐6. *./b.day2.0.build*
⭐7. *./b.day2.0.submit*
8. *cd /glade/scratch/$LOGNAME/b.day2.0*
9. After the job completes, go back to your case directory and explore the "logs" directory and the "timing" directory. How long did you job take to complete? What was the "throughput", i.e. estimated model years per actual day?
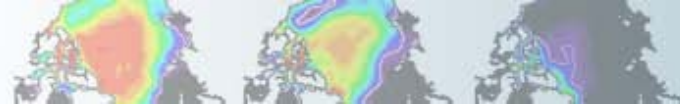
CESM1 Tutorial: Basic Modifications: Review: Queues and Jobs

Yellowstone

1. Submitting jobs:

    Type $CASE.submit i.e. *b.day2.0.submit*

2. Checking jobs:

    a. Type *bjobs* or

    b. Type *bjobs -u all* to see everyone's jobs,   or

3. Killing jobs:

    a. Find your JOBID after typing *bjobs*

    b. Type *bkill <JOBID>*

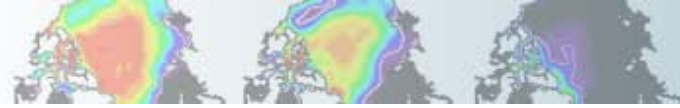## CESM1 Tutorial: Basic Modifications: Review: README

In your case directory, in addition to your scripts, you will find automatically generated documentation files.

1. **README.science_support**: refers you to the on-line documentation.

2. **README.case file**: detailed information on your compset and grid, including whether your configuration has science support.
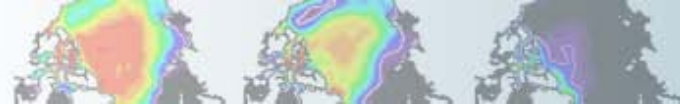
    SCIENCE_SUPPORT: NO

README.case, **we highly recommend YOU document any changes you make to the default scripts. It is YOUR paper trail and opportunity to list modifications.**

CESM1 Tutorial: Basic Modifications: create_clone

1. The **create_clone** utility creates an exact copy of a previously created case.

2. The create_clone utility is very handy when the user wishes to run a slightly modified version of a previous experiment.

    a. Invoke create_clone to create an exact copy of an old case by typing the following on the command line:

    create_clone -case <new case> -clone <case to clone>

    b. Implement desired modifications before building and running . (We will learn numerous way to modify the scripts during this presentation).

    c. DOCUMENT changes in README.case.

CESM1 Tutorial: Basic Modifications: create_clone

EXERCISE.1: Clone b.day2.0 from Exercise.0. Make no changes yet, we are simply practicing cloning a case. We will not build/run.

1. "cd" to tutorial code base scripts directory
2. *./create_clone -case ~/cases/b.day2.1 -clone ~/cases/b.day2.0*
3. *cd ~/cases/b.day2.1*
4. Review files that exist in directory before invoking cesm_setup
5. *./cesm_setup*
6. What files have been added after cesm_setup?

CESM1 Tutorial: Basic Modifications: Pre "cesm-setup" files

```
-rwxr-xr-x  1 shields ncar  15789 Jun  7 17:33 archive_metadata.sh
drwxr-xr-x  2 shields ncar  16384 Jun 25 16:11 Buildconf      NEW
-rw-r--r--  1 shields ncar    162 Jun 25 16:11 CaseStatus
-rwxr-xr-x  1 shields ncar  14495 Jun  7 17:33 cesm_setup
-rwxr-xr-x  1 shields ncar    837 Jun  7 17:33 check_case
-rwxr-xr-x  1 shields ncar  10126 Jun  7 17:33 check_input_data
-rwxr-xr-x  1 shields ncar   3672 Jun  7 17:33 create_production_test
-rw-r--r--  1 shields ncar    301 Jun 25 16:11 Depends.intel
-rw-r--r--  1 shields ncar  10841 Jun 25 16:11 env_build.xml
-rw-r--r--  1 shields ncar   4441 Jun 25 16:11 env_case.xml
-rw-r--r--  1 shields ncar   6992 Jun 25 16:11 env_mach_pes.xml
-rwxr-xr-x  1 shields ncar   2678 Jun 25 16:11 env_mach_specific
-rw-r--r--  1 shields ncar  23069 Jun 25 16:11 env_run.xml
drwxr-xr-x  2 shields ncar  16384 Jun 25 16:11 LockedFiles
-rwxr-xr-x  1 shields ncar   2127 Jun 25 16:11 preview_namelists
-rw-r--r--  1 shields ncar   1630 Jun 25 16:11 README.case
-rw-r--r--  1 shields ncar    380 Jun 25 16:11 README.science_support
drwxr-xr-x 11 shields ncar  16384 Jun 25 16:11 SourceMods
-rwxr-xr-x  1 shields ncar  13233 Jun 25 16:11 b.day2.1.build
-rwxr-xr-x  1 shields ncar   1048 Jun 25 16:11 b.day2.1.clean_build
-rwxrwxr-x  1 shields ncar    947 Jun 25 16:11 b.day2.1.l_archive
-rwxr-xr-x  1 shields ncar    608 Jun 25 16:11 b.day2.1.submit
drwxr-xr-x  3 shields ncar  16384 Jun 25 16:11 Tools
-rwxr-xr-x  1 shields ncar  12569 Jun  7 17:33 xmlchange
-rwxr-xr-x  1 shields ncar  10503 Jun  7 17:33 xmlquery
```
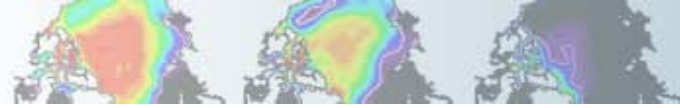
ENV Files

Tools

Case Scripts

Documentation

Buildconf scripts (do not modify) NEW

SourceMods (day 4 of tutorial)

CESM1 Tutorial: Basic Modifications: Post "cesm-setup" files

```
drwxr-xr-x  2 shields ncar  16384 Jun 25 16:18 CaseDocs
-rw-r--r--  1 shields ncar      9 Jun 25 16:18 hostfile
-rw-r--r--  1 shields ncar    949 Jun 25 16:18 Macros
-rwxr-xr-x  1 shields ncar  10287 Jun 25 16:18 b.day0.1.run
-rw-r--r--  1 shields ncar    115 Jun 25 16:18 user_nl_cam
-rw-r--r--  1 shields ncar    367 Jun 25 16:18 user_nl_cice
-rw-r--r--  1 shields ncar   1257 Jun 25 16:18 user_nl_clm
-rw-r--r--  1 shields ncar   2284 Jun 25 16:18 user_nl_cpl
-rw-r--r--  1 shields ncar   2949 Jun 25 16:18 user_nl_pop2
-rw-r--r--  1 shields ncar    573 Jun 25 16:18 user_nl_rtm
```

ENV Files

Tools

Case Scripts

Documentation

Buildconf scripts (do not modify) NEW

SourceMods (day 4 of tutorial)

Namelist modifications NEW

All case modifications are applied in **env_run.xml** or **user_nl_**<model> NEW

CESM1 Tutorial: Basic Modifications: Editing Methods/Tools

When modifying files, the user is free to use her/his editor of choice, i.e.
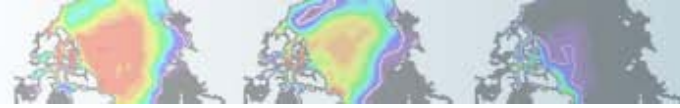
vi

emacs

When modifying "xml" files, the user may also use the tool, **xmlchange**.

1. For help, type *xmlchange -help*

2. **Example: Single variable modification, original (long) method:**

    You want to manually resubmitting an initial case that previously had a RESUBMIT value of 0, (i.e. you did not initially resubmit the run).

     Edit env_run.xml via the xmlchange tool, type
        ./xmlchange -file env_run.xml -id CONTINUE_RUN -val TRUE

CESM1 Tutorial: Basic Modifications: Editing Methods/Tools

For help, type *xmlchange -help*

3.   **Example: Multiple (or list) variable modification method:*:**

You may want to change more than one item at a time, or to use this abbreviated "list" method for a single modification. To manually resubmit the model, plus tell the scripts to automatically do this twice more…

a. Edit env_run.xml via the xmlchange tool, type

./xmlchange CONTINUE_RUN=TRUE,RESBMIT=2

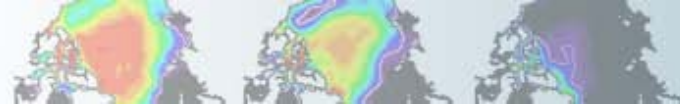*Note for most component namelist variables, the single modification method is necessary. (more on this later).

CESM1 Tutorial: Basic Modifications: Runtime variables:
env_run.xml

Runtime variables can be changed in env_run.xml *at any point* during the run and control the mechanics of the run, i.e length, resbumits, and archiving.

Common variables to change include

   1. **RESUBMIT**   → sets the number of times to resubmit the run

   2. **STOP_OPTION** → sets the run length time interval type, i.e. nmonths, ndays, nyears or a specific date

   3**. STOP_N**   → sets the number of intervals (set by STOP_OPTION) to run the model during the specified wallclock time. Wallclock time is set in your *.run file and is a measure of the actual time.

STOP_OPTION and STOP_N control the length of the run per computer job submission. A typical simulation is comprised of many job submissions. (You can only stay in the computer queue for a specified time. This queue time limit is often shorter than the desired simulation length.

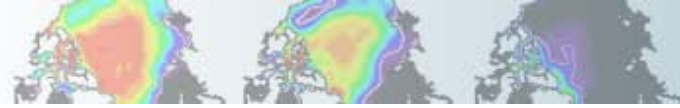## CESM1 Tutorial: Basic Modifications: Runtime variables: env_run.xml

1. **RESUMBIT** → sets the number of times to resubmit the run

2. **STOP_OPTION** → nmonths, ndays, nyears or a specific date

3. **STOP_N** → sets the number of intervals (set by STOP_OPTION) to run

Question:

The special tutorial version of T31_gx3 CESM on yellowstone (runs on multiple nodes) simulates ~44 model years per wallclock day.

Maximum wallclock requests is 12 hours.

If you want to run 100 years, what values should be set for STOP_OPTION, STOP_N, and RESUBMIT?

CESM1 Tutorial: Basic Modifications: Runtime variables
env_run.xml

Question:

If you want to run 100 years, what values should be set for STOP_OPTION, STOP_N, and RESUBMIT?
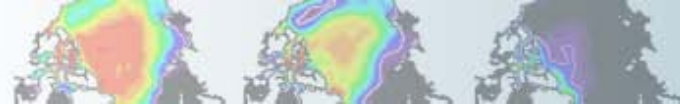
Answer:

Assume 2 jobs submissions per day, (2 12-hr jobs).

Model runs 44 yrs/day, so 44/2 = 22 model years per job submission.

Round down to 20 model years/submission to be safe (machines hang).

STOP_OPTION = nyears, STOP_N = 20 , RESUBMIT = 4

Initial run of 20yrs + (4 resubmits * 20 years per job) = 100 years

CESM1 Tutorial: Basic Modifications: Runtime variables
env_run.xml

env_run.xml continued… more common runtime variables to change include

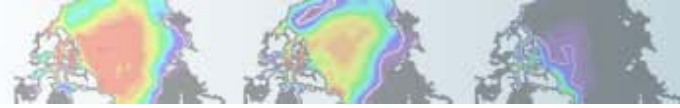4. **CONTINUE_RUN** → if TRUE, implies a CONTINUE run.

Note: if RESUBMIT is > 0 and it is an initial run (i.e. CONTINUE_RUN=FALSE), CONTINUE_RUN will automatically update to TRUE upon completion of initial run.

5. **INFO_DBUG**   → sets level of stdout (standard out) print statements. If debugging, a higher value may be set.

6. **DOUT_S**   → turns on short-term archiving. DOUT_S is TRUE by default.

7. **HIST_OPTION** → coupler history file specification. Note: All other model components specify history file information within the model component namelists! (More on namelists in a few slides).
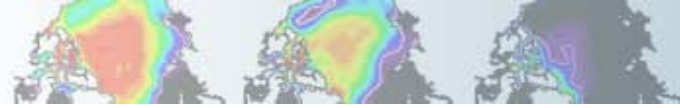
Take some time to review all other env_run.xml settings….

CESM1 Tutorial: Basic Modifications: Run-type variables: env_run.xml

Run-type variables define type of run (startup, hybrid) and physical controls (namelist parameters). Sample variables specified in this file include:

1. **RUN_TYPE**　　→ startup, hybrid, branch
2. **RUN_REFCASE** → if branch/hybrid, case name you are starting from
3. **RUN_REFDATE** → if " " , date stamp of reference case you are starting from
4. **GET_REFCASE** → default = TRUE; automatically retrieves refcase restart data;
assumes refcase is on HPSS

5. **CCSM_CO2_PPM** → CO_2 value to be propagated to CAM and CLM

6. **CAM_NAMELIST_OPTS**→ **CAM** namelist options that differ from default values
7. **CLM_NAMELIST_OPTS**→ **CLM** namelist options that differ from default values
8. **CICE_NAMELIST_OPTS**→ **CICE** namelist options for that differ from default values
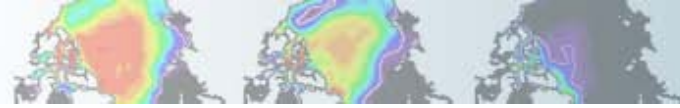
CESM1 Tutorial: Basic Modifications: Runtype variables
env_run.xml

CESM has four "types" of runs:

STARTUP: All model components are initialized from basic default initial conditions.

HYBRID: - The atmosphere and land are initialized from initial condition files generated by a user-specified CESM simulation.

- The ocean and ice are initialized from restart files generated by a user-specified CESM simulation.

- Initial conditions and restart files use the same reference case and reference date.

BRANCH: All model components are initialized from restart files generated by a user-specified CESM simulation.

CONTINUE: Continuation runs for all run types.

Default T31_gx3 cases using compset B_1850_CN (INCLUDING OUR TUTORIAL) are HYBRID runs that are initialized from the CESM1 T31x3 1850 Control.
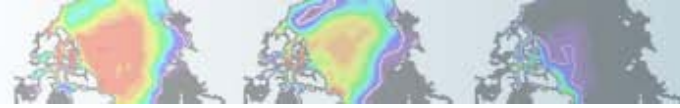
CESM1 Tutorial: Basic Modifications: Runtype variables:
Branch vs Hybrid

Branch and hybrid runs are useful if you have an experiment which only slightly differs from your control, but you want to make a slight modification, add history output, or start your simulation from a CESM spun-up initial state.

Your default tutorial exercises start as a hybrid run from the CCSM4 T31_gx3v7 control 1850 simulation.

Use a hybrid run: for most applications where you do NOT need bit for bit restart.

Use branch run: only for applications which require exact restart.

CESM1 Tutorial: Basic Modifications: Namelist variables

Namelist variables can be changed using:

env_run.xml   or

user_nl_<model> (i.e. user_nl_cam, user_nl_pop2, etc)

CAM, CLM, CICE, RTM namelist variables can be changed using both methods.

POP, CPL can be only changed using the user_nl_<model> method.

For a complete list of namelist, please see the on-line documentation for each
    component model.

CAM4: http://www.cesm.ucar.edu/cgi-bin/eaton/namelist/nldef2html-pub

CAM5: http://www.cesm.ucar.edu/cgi-bin/eaton/namelist/nldef2html-cam5_2

EXERCISE.2: Create a new fully coupled (with CN) hybrid case from 1850 conditions, but double the CO_2 values and turn on the ice runoff in the land model. Increase the amount of standard out produced by the model and change the default location for the log files to a directory on /glade/scratch. Run 2 months.

1. from scripts directory, create your case scripts:
   ./create_newcase -case   ~/cases/b.day2.2   -res T31_gx3v7 - mach yellowstone -compset B_1850_CN

2. from case directory, change your runtime variables:
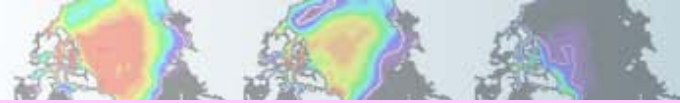   *./xmlchange INFO_DBUG =2,STOP_N=2,STOP_OPTION=nmonths*

3. *vi (or emacs) env_run.xml to change LOGDIR to /glade/scratch/$CCSMUSER/logs/$CASE , **or***
   ./xmlchange -file env_run.xml -id LOGDIR -val '/glade/scratch/$CCSMUSER/logs/$CASE'
   (note that single quotes prevent evaluation of environment variables)

CESM1 Tutorial: Basic Modifications: env_run.xml

EXERCISE.2 continued:

4.   Make your physics namelist changes:

   a. Use the list method: *./xmlchange CCSM_CO2_PPMV=569.4*

   b. Use the original method:

      ./xmlchange -file env_run.xml -id RTM_NAMELIST_OPTS -val ice_runoff=.true.

5.   *./cesm_setup*

6.   Manually update your README.case file to reflect changes
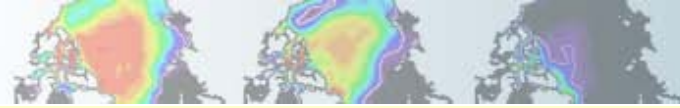
7.   *./b.day2.2.build*

8.   Review *b.day2.2.run*:

   a) *#BSUB -P UESM0001* ⟵ your project number
   b) *#BSUB –q small* ⟵ requested queue
   Note this is for tutorial purposes ONLY.

9.   *./b.day2.2.submit*

   Review log files to familiarize yourself with standard out. Start looking at your history files. Check your CaseDocs/*_in files. Were your namelist changes applied?

CESM1 Tutorial: Basic Modifications: HOMEWORK

Before you leave for the day:

Resubmit and continue to run Exercise 2 for 24 more months. This will run overnight in the yellowstone queues.
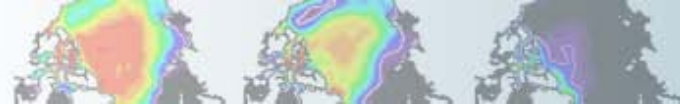
Your data will be used for the Practical Session tomorrow on *Diagnostics and Output*. Assuming your b.day2.2 exercise ran successfully in class….

In env_run.xml:

1. Set *CONTINUE_RUN* to *"TRUE"*

2.   Keep *STOP_OPTION set* to *"nmonths"*

3. Set *STOP _N* to *"24"*

4. Set *INFO_DBUG* to *1*

5. Remove *"-U CESM_WS"* from BATCHSUBMIT variable in env_run.xml
    should look like   *<entry id="BATCHSUBMIT"   value="bsub <" />*

Try to get as many exercises in this presentation completed in class *BEFORE* you go back and set up this continue run.

Note: The reserved nodes are for day-time use only. Overnight jobs can either be run in the workshop queue or regular queue.

CESM1 Tutorial: Basic Modifications: Namelist variables/tool: user_nl_<model>

• Not all changes can be made in env_run.xml.

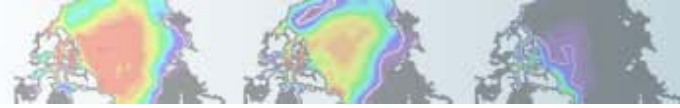• **user_nl_<model>** files appear in the case directory after ./cesm_setup has been invoked), i.e.

user_nl_cam	atmosphere
user_nl_clm	land
user_nl_cice	ice
user_nl_pop2	ocean
user_nl_cpl	coupler (driver)
user_nl_rtm	river transport

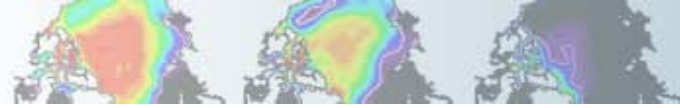## CESM1 Tutorial: Basic Modifications: Namelist tool:
preview_namelist

•Insert namelist syntax for desired variable change into the appropriate file. To find the proper syntax and see all default namelist values, use **preview_namelist** to create the resolved namelists the model will use at runtime in your run directory (i.e. /glade/scratch/<user>/<case>/run/). Note that cesm_setup automatically runs preview_namelist so if you have already run *./cesm_setup*, you don't need to do this again.

•In your case directory, type   **./preview_namelists**

•cd to your run directory and view *_in files:

user_nl_cam modifies	⟶	atm_in
user_nl_clm modifies	⟶	lnd_in
user_nl_cice modifies	⟶	ice_in
user_nl_pop2 modifies	⟶	pop2_in
user_nl_cpl modifies	⟶	drv_in
user_nl_rtm modifies	⟶	rof_in

•Default namelist can also be reviewed in your CaseDocs subdirectory (found in your case directory after cesm_setup has been invoked).
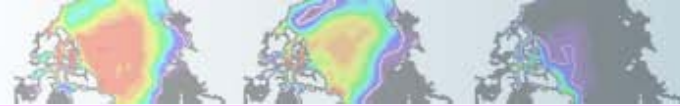
CESM1 Tutorial: Basic Modifications: Namelist tool: preview_namelist

Example: Increase timestep in the ocean model from 12 steps per day to 23 steps per day.

1. Edit (vi or emacs) user_nl_pop2

2. Insert correct syntax as a new line at the end of the comment section in the form of: *namelist_var = new_namelist_value*

    i.e.,

             dt_count = 23

3.   Invoke *preview_namelists* again to verify change in you run directory and update the documentation pop2_in file in CaseDocs.

CESM1 Tutorial: Basic Modifications: user_nl_<model>

EXERCISE.3: BRANCH from the end (of the second month) in Exercise.2 (not the end of the homework). Turn off ice runoff in the RTM (runoff) model and quadruple CO_2 and double methane. Run 1 month. Restart for 1 month. Use the user_nl_<model> method. Check your resolved namelists in your run directory.

1.   *./create_newcase -case ~/cases/b.day2.3 -res T31_gx3v7 -mach yellowstone -compset B_1850_CN*

2.   *cd   ~/cases/b.day2.3*

3. *./xmlchange RUN_TYPE=branch,RUN_REFCASE=b.day2.2,RUN_REFDATE=0001-03-01, GET_REFCASE=FALSE,STOP_OPTION=nmonths,STOP_N=1,RESUBMIT=1,CCSM_CO2_PPMV =1138.8*   (no spaces between arguments).

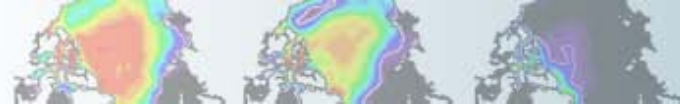**Why do we need to set GET_REFCASE=FALSE?**

CESM1 Tutorial: Basic Modifications: user_nl_<model>

EXERCISE.3: continued

4. *./xmlchange -file env_run.xml -id CAM_NAMELIST_OPTS -val ch4vmr=1583.2e-9*

5. *./xmlchange -file env_run.xml -id RTM_NAMELIST_OPTS -val ice_runoff=.false.*

Why do we use the single ./xmlchange method here?

6. *./cesm_setup*

7. Copy your restart files from your bday2.2 short term archive space to your bday2.3 $RUNDIR:

    cp /glade/scratch/<user>/archive/b.day2.2/rest/0001-03-01-00000/ /glade/scratch/<user>/b.day2.3/run/.*

8. *./b.day2.3.build*

9. *./b.day2.3.submit*

10. Review queues and log files. How fast does the first month run? Was the second month resubmitted?

CESM1 Tutorial: Basic Modifications:
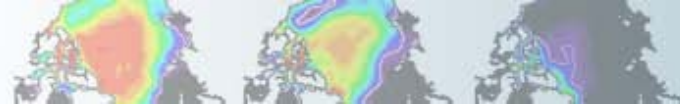## env_run.xml   vs.   user_nl_<model>

What method is best for namelist changes?

env_run.xml:

- Run_type specification (startup, hybrid, branch, continue)

- Runtime variables (stop_option, resubmits, etc.)

- Namelist changes for CAM,CLM,CICE,RTM before cesm_setup is invoked
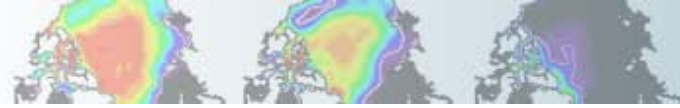
user_nl_<model>:

- Swapping out a default inputdata set for a home-grown dataset

- Namelist changes for POP or CPL

- Namelist changes that you forgot to include in env_run.xml before invoking cesm_setup.

- Namelist changes from scripts created when cloning another case.

CESM1 Tutorial: Basic Modifications:
env_run.xml   vs.   user_nl_<model>

CAUTION!! Namelist changes to env_run.xml that are made *after* cesm_setup has been invoked will not make it into the resolved namelists until cesm_setup or preview_namelist is re-run. At runtime, the scripts will automatically re-populate your resolved namelists based on env_run and the user_nl_<model> files, however, it is always good to document and check your changes BEFORE runtime.

Always check your resolved *_in files (run directory, i.e. /glade/scratch) to make sure your changes have been applied.

CESM1 Tutorial: Basic Modifications: Time Step Changes
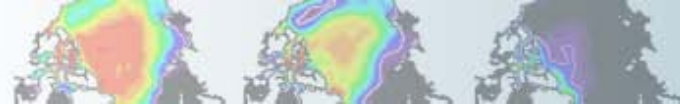
Where and When to Change Time Steps

When the model crashes due to large, temporary instabilities, one method to overcome the problem is to change the time step.

This is typically done in either the atmosphere or ocean components.

CAM/CLM: ATM_NCPL in env_run.xml.

POP: dt_count in POP namelist.   Edit and change the user_nl_pop file.
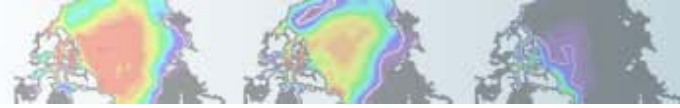
GOTCHAS?

CESM1 Tutorial: Basic Modifications: Time Step Changes

NEW

1. **ATM_NCPL** in env_run.xml specifies the number of **coupling intervals per day** between the atmosphere/land and the coupled system. Based on ATM_CPL, the scripts will automatically compute the timestep for the atmosphere and land (DTIME) and populate the namelist files accordingly.

2. **CLM time step** = CAM time step; this is automatically set with the CAM timestep via ATM_NCPL. You cannot set this separately.

NEW

3. **RTM time step** , now its own model component, sets its timestep explicitly and is no longer relative to the CLM timestep. (Default RTM timestep is 3 hours).
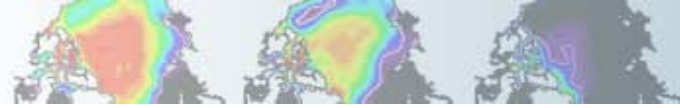
CESM1 Tutorial: Basic Modifications: More Exercises

Exercise.4: Create new case, hybrid start, fully coupled for Pre-Industrial conditions (with CN). Quadruple the specified methane trace gas value in the atmosphere, but decrease the solar constant by 0.1%, and set the orbital parameters to 1600. (Note: this is an exercise and does not represent any historical period). Run 5 days (default). Resubmit for another 5 days. What happens to *CONTINUE_RUN* in env_run.xml <u>after</u> completion of the initial 5day run (assuming you set RESUBMIT = 1) ?

1. ./create_newcase -case ~/cases/b.day2.4 -res T31_gx3v7 -mach yellowstone -compset B_1850_CN

2. *cd   ~/cases/b.day2.4*

3. *./xmlchange RESUBMIT =1*

4. *./xmlchange -file env_run.xml -id CAM_NAMELIST_OPTS -val solar_const=1359.53,ch4vmr=3166.40e-9*

5. *./cesm_setup*

CESM1 Tutorial: Basic Modifications: More Exercises

Exercise.4: Continued:

6.   Edit user_nl_cpl, after comments, add line: *orb_iyear=1600*

7.   Update README.case

8.   Optional: *./preview_namelists*

9.   *./b.day2.4.build*
       Check your $RUNDIR and review your namelist changes in atm_in and drv_in
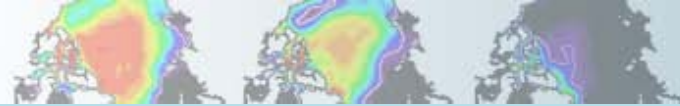
10. Review *b.day2.4.run*:

   Note queue (small) and project number (UESM0001) are for tutorial purposes ONLY.

   Normally, you will run with your own project number in the regular queue.

11. *./b.day2.4.submit*

After the job completes, go to the short term archive space and explore.

CESM1 Tutorial: Basic Modifications: More Exercises

Exercise.5: Clone case from Exercise.4. Include all modifications from Exercise.4 except instead of specifying orbital year, assign individual parameters (eccentricity=0, obliquity=23., and precession=0.) Include new modification to use a different short wave absorption parameterization in POP called "jerlov". (The default sw absorption parameterization is geography-specific and called "chlorophyll". "Jerlov" is typically used for paleoclimate simulations where the geography is different from present day). Turn off the Urban paramterization in CLM. Run 5 days (default).

1. *./create_clone -case ~/cases/b.day2.5 -clone ~/cases/b.day2.4*
2 . Clean up stderr and stdout that don't belong to your new case.
3.   Edit user_nl_cpl and change the following:
   a. Remove *orb_iyear*
   b. Add *orb_mode = 'fixed_parameters'*
   c. Add *orb_eccen = 0.*
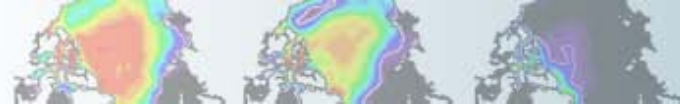   d. Add *orb_mvelp = 0.*
   e. Add *orb_obliq = 23.*

CESM1 Tutorial: Basic Modifications: More Exercises

Exercise.5:   continued

4. Edit user_nl_pop2 and add: *sw_absorption_type = 'jerlov'*

5. Edit user_nl_clm and add: urban_hac = 'OFF'

6. Note that in env_run.xml, create_clone has automatically set *CONTINUE_RUN* back to *FALSE*, this is an initial run.

7. Update your README.case file. Document your changes

8. *./cesm_setup*

   Because you cloned this case you already had your user_nl_<model> file in your case case directory, but you have changed them and therefore need to invoke cesm_setup or preview_namelist to update your CaseDocs, create the resolved namelists in your run directory, and create your runscript.

CESM1 Tutorial: Basic Modifications: More Exercises

Exercise.5:   continued

9. *./b.day.2.5.build*

10. *./b.day2.5.submit*

11. Go over all log files and timing files. If you want, you can start to look at the history output. Only the atm/ocn will have daily output to view. (ncview). Where is the short term history output located?

12. Compare b.day2.5 history data to b.day2.4 history data. (ncdiff).

14. Optional: Resubmit Exercise.4 and Exercise.5 for 1 month to obtain monthly history files for all model components. What will you need to change?
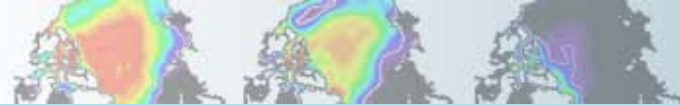
CESM1 Tutorial: Basic Modifications: More Exercises
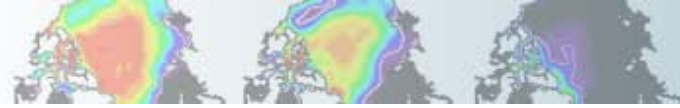
Exercise.6: On your own…

Do a "startup" run. Change the atm/lnd timestep to 900 seconds. (Default value is 1800 seconds with ATM_NCPL=48). Run 1 month. Be sure all time steps are compatible. Be sure to update your README.case file to keep track of your changes. Review logs to verify time step changes.

Exercise.7: On your own…

Branch start using Exercise.6 restart files. Include all Exercise 6 modifications. Reduce the snow albedoes over sea ice in CICE by half. Run 1 month.

Know what you are changing. Look up information on namelist variables in the documentation. (http://www.cesm.ucar.edu/models/cesm1.2/cice/doc/index.html).

Be sure to update your README.case file to keep track of your changes. The model will run regardless of whether or not your remember to include all of your changes. Check your resolved/processed namelist files ($RUNDIR/<model>_in files) to make sure all changes are included. If you like, resubmit and continue the run for 1 more month.

CESM1 Tutorial: Basic Modifications: Bottom Line

What user-modified files are actually used at runtime?

./cesm_setup (or *./preview_namelists*) ⟶ $RUNDIR/atm_in

$RUNDIR/lnd_in

$RUNDIR/pop2_in
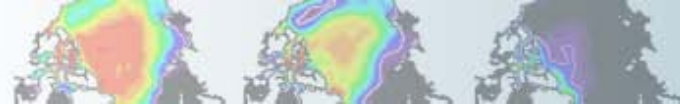
$RUNDIR/ice_in

$RUNDIR/drv_in

Bottom Line: User modifications should be implemented in the env_run.xml or the user_nl_<model> files.

What files are for documentation purposes?

Buildconf/*.input_data_list

CaseDocs, CaseStatus, LockedFiles, README.*

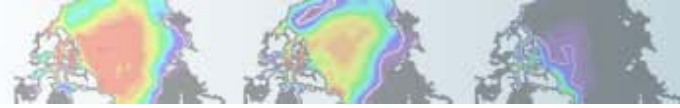Note: Buildconf/*conf directories are created after cesm_setup. The user does NOT need to touch these files.

CESM1 Tutorial: Basic Modifications: Run Scripts

Review run script: *mycase.run*

Common BSUB command to change:

#BSUB -q **small**   queue type*

#BSUB -o cesm.stdout.%J   machine standard out

#BSUB -e cesm.stderr.%J   machine standard error

#BSUB -J b.day2.t   job name

#BSUB -W 2:00   wallclock time requested

#BSUB -P **UESM0001**   **project number**

Note: Maximum allowable wall clock time per job submission varies from machine to machine and queue to queue. Yellowstone (NCAR's IBM) regular queue is 12 hours . Submissions requesting under an hour typically have shorter wait times in the queue.

CESM1 Tutorial: Basic Modifications: Log Files

Log Files:

Model runtime standard output

During model execution: $RUNDIR/*
After model completion: $LOGDIR/*

atm.log.yyddmm-nnnnnn.gz   Files are gzipped after model completion.
 lnd.log.yyddmm-nnnnnn.gz   Restore by typing *gunzip <logfile>.*
ocn.log.yyddmm-nnnnnn.gz
ice.log.yyddmm-nnnnnn.gz   yyddmm = year, month, day
cpl.log.yyddmm-nnnnnn.gz   nnnnnn = time id stamp
rof.log.yyddmm-nnnnnn.gz
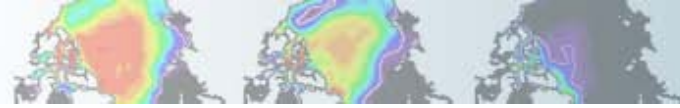cesm.log.yyddmm-nnnnnn.gz

CESM1 Tutorial: Basic Modifications: Other Tips

CHECK your resolved namelists!

- Before you submit your job, it is always good to double check your $RUNDIR/<model>_in namelist files. These are the files the model will actually use at runtime and are based on your env_run.xml and user_nl_<model> files.

- Verify that <u>the model is using what you think it is using</u>!

DOCUMENT everything you do!

- A paper trail of your procedures and thoughts is good scientific practice. The README.case file is the perfect place to write notes. You will thank yourself months (years) later, when you are trying to figure out what you did oh-so-long ago!

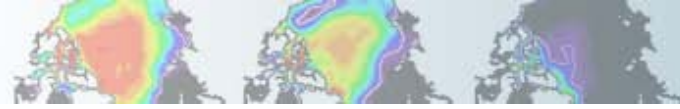# CESM1 Tutorial: Basic Modifications: Post Run Tips

Check logs

- Check your log files to make sure there are no hidden problems and to verify the model is running smoothly and as you expect. The log files may also help you verify your modifications were included in your run.

Check output

- Check your history files. It is a good idea to run a small test sample of your experiment before launching your full production run. For example, if you want to run a 500 year control with various modifications, first run 10 years. Check the history output files and verify the model is running as you designed before continuing with the full 500 years. It is always best to find errors early, rather than later, in the run.

Check timings

- Check your timings. After model completion, a timing subdirectory will be placed in your scripts directory. Check the timings after several job completions to verify that the model is running efficiently and as expected. Double check your timings with the CESM default timings for your specific model resolution and machine. Default timings for CESM1.1.1 can be found at: http://www.cesm.ucar.edu/models/cesm1.1/timing/

CESM1 Tutorial: Basic Modifications

Have Fun!!!