

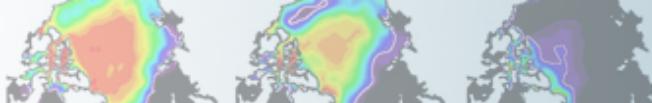
CESM2 Tutorial: Basic Modifications

Christine Shields

August 15, 2017



Office of
Science

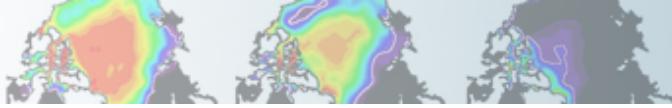


CESM2 Tutorial: Basic Modifications: Review

1. We will use the CESM code located locally on Cheyenne, no need to checkout or download any input data.
2. We will run with resolution f19_g17: (atm/lnd = FV 1.9x2.5 ocn/ice=gx1v7)
3. Default scripts will **automatically** be configured for you using the code/script base prepared uniquely for this tutorial.
4. **For On-site Tutorial ONLY:** Please use compute nodes for compiling and login nodes for all other work, including submission. Please do NOT compile unless you have a compile card. To make the tutorial run smoothly for all, we need to regulate the model compiles. When you run from home, you don't need to compile on the compute nodes.

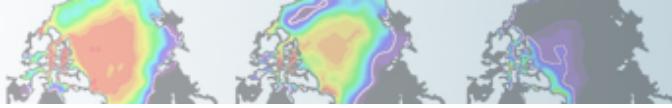
Tutorial Code and script base:

/glade/p/cesm/tutorial/cesm2_0_alpha07c/



CESM2 Tutorial: Basic Modifications: Review

1. Log into Cheyenne
2. Execute `create_newcase`
3. Execute `case.setup`
- 4. Log onto compute node
(`compile_node.csh`)**
- 5. Compile model (`case.build`)**
- 6. Exit compute node (type “exit”)**
7. Run model (`case.submit`)



**This tutorial contains step by step instructions applicable to CESM2
(which has not been officially released yet).**

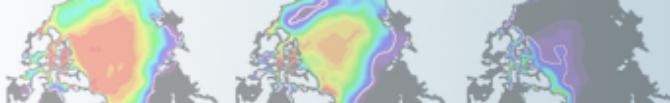
Documentation: Under construction!

<http://www.cesm.ucar.edu/models/cesm2.0/>

Quick Start Guide: Under construction!

<http://cesm-development.github.io/cime/doc/build/html/index.html>

For older releases, please see past tutorials.



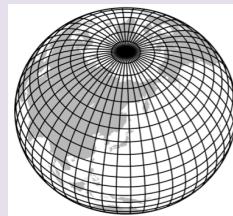
CESM2 Tutorial: Basic Modifications: Review: Creating a new case

What is the casename ?



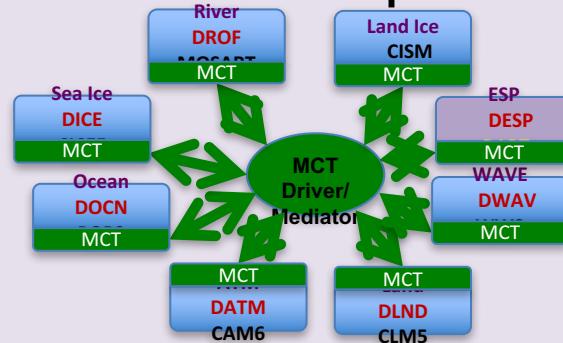
`~/cases/b.day1.0`

Which resolution?

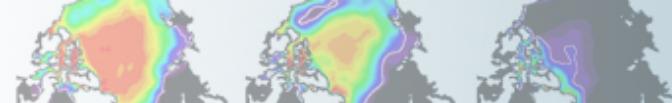


`f19_g17`
(FV 2deg coupled to gx1 ocean)

Which model configuration ?
Which set of components ?



`B1850`



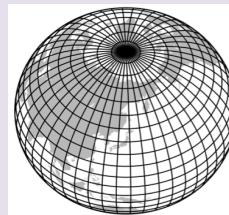
CESM2 Tutorial: Basic Modifications: Review: Creating a new case

What is the casename ?



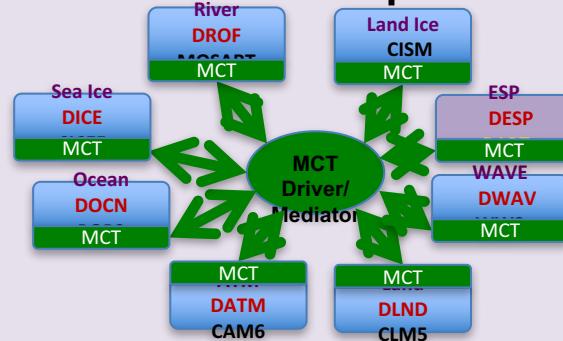
`~/cases/b.day1.0`

Which resolution?



`f19_g17`
(FV 2deg coupled to gx1 ocean)

Which model configuration ?
Which set of components ?



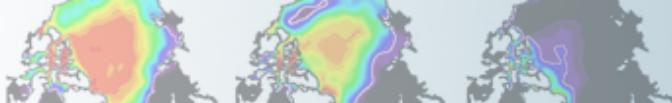
`B1850`

```
create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset  
B1850
```

```
create_newcase --help (full list of arguments)
```

Other arguments you may need when you get home:

--project <> (project number; this is not used by default and job would be charged to your default project number)



CESM2 Tutorial: Basic Modifications: Review: Documentation

Documentation for CESM2 is UNDER CONSTRUCTION, This will CHANGE!

Grid naming convention

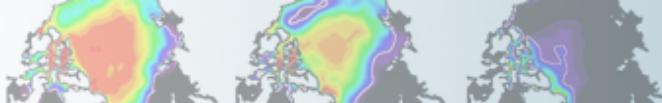
<http://www.cesm.ucar.edu/models/cesm2.0/cesm/grids.html>

Grid Alias: **f19_g17**

```
non-default grids are: atm:1.9x2.5 lnd:1.9x2.5 ocnice:gx1v7  
mask is: gx1v7
```

```
1.9x2.5 is FV 2-deg grid: with domain file(s):  
domain.lnd.fv1.9x2.5_gx1v6.090206.nc (only for mask: gx1v6 grid match: atm|lnd)  
domain.ocn.1.9x2.5_gx1v6_090403.nc (only for mask: gx1v6 grid match: ocnice)  
domain.aqua.fv1.9x2.5.nc (only for mask: null grid match: ocnice)
```

```
gx1v7 is displaced Greenland pole 1-deg grid with Caspian as a land feature: with domain file(s):  
$DIN_LOC_ROOT/share/domains/domain.ocn.gx1v7.151020.nc (only for grid match: atm|lnd)  
$DIN_LOC_ROOT/share/domains/domain.ocn.gx1v7.151020.nc (only for grid match: ocnice)
```



CESM2 Tutorial: Basic Modifications: Review: Documentation

Documentation for CESM2 is UNDER CONSTRUCTION, This will CHANGE!

Compset naming convention:

<http://www.cesm.ucar.edu/models/cesm2.0/cesm/compsets.html>

B1850=

1850_CAM60_CLM50%**BGC**_CICE_POP2%**ECO**_MOSART_CISM2%NOEVOLVE_WW3_BGC%**BDRD**

**Component
Version
Mode/Physics**

ATM = CAM6.0

LND = CLM5.0

ICE = CICE5

OCN = POP2.0

ROF = MOSART

WAV = Wave Watch3

BGC = diagnostic

Component Set Definitions (compsets)

Model Version: CESM2.0
HTML created on: 2017-06-04

This page contains the complete list of component sets aliases and long names.
They are grouped by model components designed to aid browsing.

Clicking on the name of a component will display additional descriptive information.
Click on the "Show Details" button and then ctrl+F key to search for specific strings in this file.

[Show Details](#) [Hide Details](#)

Compset Naming Convention

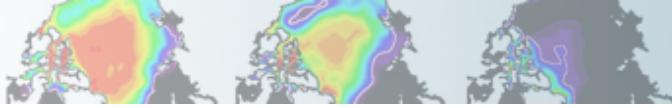
Component: [allactive](#)

Alias	Long Name
B1850	1850_CAM60_CLM50% BGC _CICE_POP2% ECO _MOSART_CISM2%NOEVOLVE_WW3_BGC% BDRD
B1850C4L45BGCPRP	1850_CAM40_CLM45% BGC _CICE_POP2% ECO _MOSART_SGLC_SWAV_BGC% BPRP
B1850C4L45BGCPRP	1850_CAM40_CLM45% BGC _CICE_POP2% ECO _RTM_SGLC_SWAV_BGC% BPRP
B1850C5L45BGC	1850_CAM50_CLM45% BGC _CICE_POP2_MOSART_SGLC_SWAV
B1850C5L45BGCRCR	1850_CAM50_CLM45% BGC _CICE_POP2_RTM_SGLC_SWAV
B1850C5L45BGCPRP	1850_CAM50_CLM45% BGC _CICE_POP2% ECO _RTM_SGLC_SWAV_BGC% BPRP
B1850C5L45BGCRCG	1850_CAM50_CLM45% BGC _CICE_POP2_RTM_CISM2_SWAV
B1850C5L45BGCRCG1	1850_CAM50_CLM45% BGC _CICE_POP2_RTM_CISM1_SWAV
B1850Ws	1850_CAM60_CLM50% BGC _CICE_POP2% ECO _MOSART_CISM2%NOEVOLVE_SWAV_BGC% BDRD
BC5L45BGC	2000_CAM50_CLM45% BGC _CICE_POP2_MOSART_SGLC_SWAV
BC5L45BGCRCR	2000_CAM50_CLM45% BGC _CICE_POP2_RTM_SGLC_SWAV
BC5L45BGCRCG	2000_CAM50_CLM45% BGC _CICE_POP2_RTM_CISM2_SWAV
BHIST	HIST_CAM60_CLM50% BGC _CICE_POP2% ECO _MOSART_CISM2%NOEVOLVE_WW3_BGC% BDRD
BHISTWs	HIST_CAM60_CLM50% BGC _CICE_POP2% ECO _MOSART_CISM2%NOEVOLVE_SWAV_BGC% BDRD
BRCP85C5L45BGC	RPC8_CAM50_CLM45% BGC _CICE_POP2_MOSART_SGLC_SWAV
BRCP85C5L45BGCRCR	RPC8_CAM50_CLM45% BGC _CICE_POP2_RTM_SGLC_SWAV
BW1850	1850_CAM60%WCTS_CLM50% BGC _CICE_POP2% ECO _MOSART_CISM2%NOEVOLVE_WW3

Compset Mode Descriptions

The modes define options for the different component models using a python regular expression syntax for matching when creating a new case.

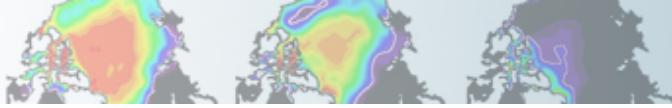
Mode	Description
%ADIAB	CAM adiabatic physics:
%CCTS[%_]	CAM-Chem troposphere / stratosphere chemistry with simplified volatility basis set SOA scheme and modal aerosols :
%CLB[%_]	CAM CLUBB:
%DABIP04	CAM dry adiabatic baroclinic instability (Polvani 2004):
%DCTBM	CAM dynamical core test with baroclinic wave IC and terminator chemistry:
%HS94	CAM Held-Suarez forcing:
%KESSLER	CAM dynamical core test with baroclinic wave IC and Kessler physics:
%PM[%_]	CAM prescribed modal aerosols:
%PORT	CAM Parallel Offline Radiation Tool:
%RCO2[%_]	CAM CO2 ramp:
%SPCAMCLBM	CAM super-parameterized CAM double moment m2005 SAM microphysics using CLUBB
%SPCAMCLBS	CAM super-parameterized CAM one moment SAM microphysics using CLUBB



CESM2 Tutorial: Basic Modifications: Review: Creating a new case

EXERCISE.0: Create and configure an out-of-the-box case (set of scripts) called “b.day2.0” on Cheyenne using FV 2deg atm/lnd coupled to 1deg ocean/ice and compset B1850. Review steps but do not build or run.

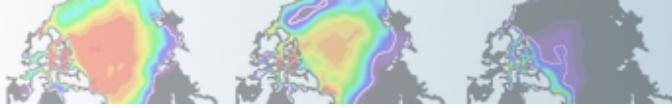
1. Change directories, (“cd”) to tutorial code base scripts directory (on slide 2).
2. View compset and grid choices. View `create_newcase` help page (hint: go to webpages listed in slides 4, 5 and 6).
3. Create initial scripts. (We will use the same “cases” subdirectory as day1).
4. “cd” to your casedir.
5. Setup your case.
6. Explore your directories



CESM2 Tutorial: Basic Modifications: Review: Creating a new case

EXERCISE.0: Create and configure an out-of-the-box case (set of scripts) called “b.day2.0” on cheyenne using f19_g17 and compset B1850. Review steps but do not build or run.

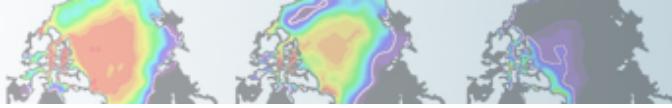
1. `cd /glade/p/cesm/tutorial/cesm2_0_alpha07c/cime/scripts`
2. `./create_newcase --help` (*webpages can be viewed in browser*)
3. `./create_newcase --case ~/cases/b.day2.0 --res f19_g17 --compset B1850`
4. `cd ~/cases/b.day2.0`
5. `./case.setup`
6. *Look over documentation generated by scripts*
7. *What are the next steps if you were to compile and run?*



CESM2 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 cheyenne using FV1.9x2.5_gx1v7 and compset B1850. Review steps but do not build or run.

1. `cd /glade/p/cesm/tutorial/cesm2_0_alpha07c/cime/scripts`
2. `./create_newcase --help` (*webpages can be viewed in browser*)
3. `./create_newcase --case ~/cases/b.day2.0 --res f19_g17 --compset B1850`
4. `cd ~/cases/b.day2.0`
5. `./case.setup`
6. *Look over documentation generated by scripts*
7. `./case.build`
8. `./case.submit`



CESM2 Tutorial: Basic Modifications: Review: Queues and Jobs

Cheyenne

1. Submitting CESM2 jobs:

Type `./case.submit`

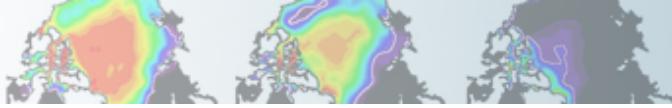
2. Checking jobs:

Type `qstat -u <username>`

3. Killing jobs:

a. Type `qstat | grep <username>` to find your JOBID

b. Type `qdel <JOBID>`, example: `qdel 1547261`

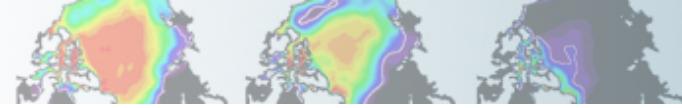


CESM2 Tutorial: Basic Modifications: **Review:** **README**

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

1. **README.case** file: information on your compset, grid, and physics modes
2. **CaseDocs/**: namelist configurations for your components (do not modify)
3. **software_environment.txt**: software information
4. **CaseStatus**: documents your xmlchange commands, builds, submissions, and completions (including errors) with timestamps.

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.



CESM2 Tutorial: Basic Modifications: **create_clone**

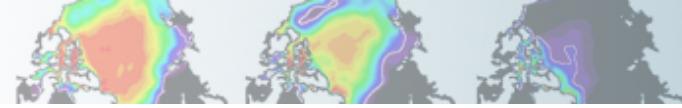
The **create_clone** utility creates an **EXACT** copy of a previously created case.

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. Edit and DOCUMENT changes in README.case



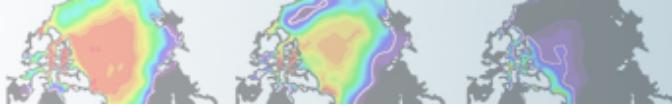
CESM2 Tutorial: Basic Modifications: [create_clone](#)

Edit and DOCUMENT changes in README.case.

Otherwise your README.case file will look exactly like your original case and it will be much harder to backtrack your methods when troubleshooting.

CAVEATS for CREATE_CLONE: you need to use....

- 1) same model tag**
- 2) same machine**
- 3) same compset**
- 4) same resolution**
- 5) same run-type** (slide 25)



CESM2 Tutorial: Basic Modifications: **Model control files**

We control how we compile and run the model with `env_*.xml` files.

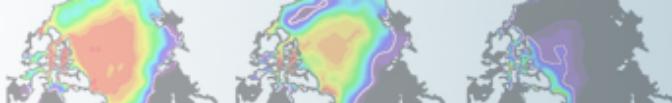
These files are created with `create_newcase`.

We modify `env_run.xml` according to our experimental design. We will practice this in the coming exercises.

We control what we ask of the model components with namelist files, `user_nl_<model>`.

These files are created after `case.setup` is invoked.

We modify the model component namelists according to our experimental design. We will practice some basic examples here, and more complex examples on Thursday.



CESM2 Tutorial: Basic Modifications: **Editing Methods/Tools**

Recommended:

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

However, the user is free to use her/his editor of choice, i.e.

vi

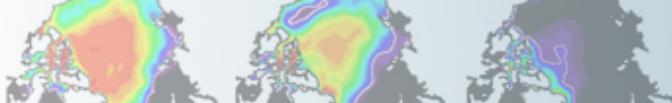
emacs

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
```



CESM2 Tutorial: Basic Modifications: **Editing Methods/Tools**

For help, type *xmlchange --help*

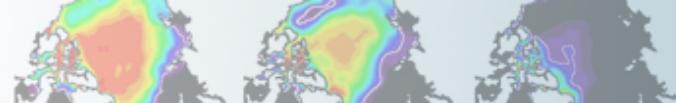
3. Example: Multiple (or list) variable modification method*:*

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,RESUBMIT=2
```

*Note for most component namelist variables, the single modification method is necessary.



CESM2 Tutorial: Basic Modifications: **Editing Methods/Tools**

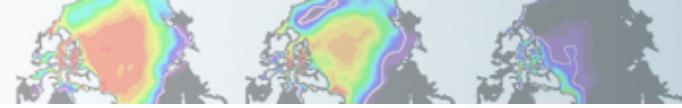
For help, type *xmlchange –help*

4. Example: using Subgroups

For changing variables in env files that have multiple instances, we recommend you use the Subgroup functionality in `xmlchange` to specify which instance you want to change.

To change the default WALLCLOCK time from 12 hours to 1 hour for the short term archiver subgroup, i.e. `<group id="case.st_archive">`, type the following on the command line:

```
./xmlchange --subgroup case.st_archive JOB_WALLCLOCK_TIME=01:00:00
```



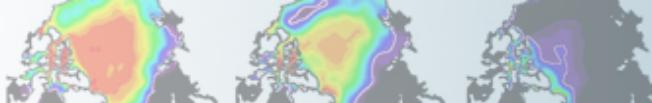
CESM2 Tutorial: Basic Modifications: **Namelist variables**

Namelist variables can be changed using:

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

For a complete list of namelists, please see the on-line documentation for each component model. (More on this later)...

<http://www.cesm.ucar.edu/models/cesm2.0/namelists/index.html>



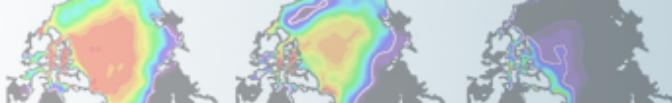
CESM2 Tutorial: Basic Modifications: Runtime variables: **env_run.xml**

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.)



CESM2 Tutorial: Basic Modifications: Runtime variables: **env_run.xml**

1. **RESUBMIT** → sets the number of times to resubmit the run
2. **STOP_OPTION** → nmmonths, ndays, nyears or a specific date
3. **STOP_N** → sets the number of intervals (set by STOP_OPTION) to run

Question:

The tutorial version of FV ~2deg_gx1 CESM on cheyenne simulates ~10 model years per wallclock day.

Maximum wallclock request is 12 hours.

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



CESM2 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?

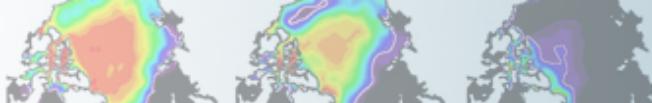
Answer:

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

Model runs 10yrs/day, so $10/2 = 5$ model years per job submission.

STOP_OPTION = nyears, STOP_N = 5 , RESUBMIT = 19

Initial run of 5yrs + (19 resubmits * 5 years per job) = 100 years



CESM2 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 ("driver") history file specification. Note: All other model components specify history file information within the model component namelists!

8. **CCSM_CO2_PPMV** → CO₂ value to be propagated to POP and CLM



CESM2 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 = FALSE; for TRUE, data needs to be pre-staged in executable directory (this is different from CESM1).



CESM2 Tutorial: Basic Modifications: Run-TYPE variables

`env_run.xml`

CESM has four “types” of runs:

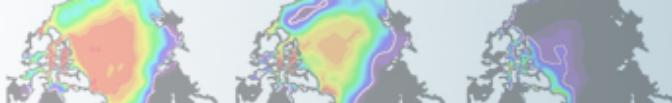
STARTUP: All model components are initialized from basic default initial conditions.

HYBRID:

- The atmosphere is initialized from initial condition files generated by a user-specified CESM simulation.
- The land, ocean and ice are initialized from restart files generated by a user-specified CESM simulation.
- No coupler file is needed
- 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.



CESM2 Tutorial: Basic Modifications: Run-TYPE 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.

Use a hybrid run: for most applications where you do NOT need bit for bit restart. You CAN specify a new start date for your model run.

Use branch run: only for applications which require exact restart. You CANNOT specify a new start date for your model run. It will be assigned by the reference case (RUN_REFDATE). (Example, if you want to change the history output stream mid-run, you will need to branch).



CESM2 Tutorial: Basic Modifications: **env_run.xml**

EXERCISE.1: Create a new fully coupled startup case from 1850 conditions and increase the amount of standard out produced by the model. Change the default location for the log files to a directory on /glade/scratch. Run 1 month.

1. from scripts directory, create your case scripts:

```
./create_newcase --case ~/cases/b.day2.1 --res f19_g17 --compset B1850
```

2. from case directory, change your runtime variables:

```
./xmlchange INFO_DBUG=2,STOP_N=1,STOP_OPTION=nmonths
```

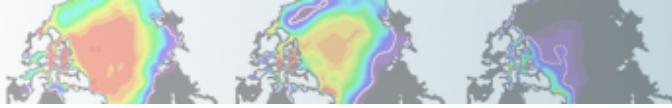
```
./xmlchange --subgroup case.run JOB_QUEUE=R1578615 ← tutorial only
```

```
./xmlchange --subgroup case.st_archive JOB_QUEUE=R1578615
```

```
./xmlchange JOB_WALLCLOCK_TIME=2:00:00
```

3. In env_run file, specify a new log directory on your scratch space (warning: don't copy and paste here, add your username. Also, cut and past characters may not translate).

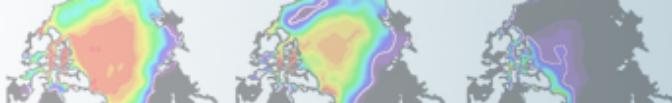
```
./xmlchange --file env_run.xml --id LOGDIR --val "/glade/scratch/$LOGNAME/logs/b.day2.1"
```



CESM2 Tutorial: Basic Modifications: `env_run.xml`

EXERCISE.1 continued:

4. `./case.setup`
5. Manually update your `README.case` file to document your changes (Hint: type “history” on the command line and you will see all command line modifications you have made).
6. `/glade/p/cesm/tutorial/compile_node.csh` (**on-site tutorial only**) to log onto compute node for compilation)
7. `./case.build`
8. Exit compute node.
9. Review `case.run` for informational purposes (normally, you don’t need to touch this).
10. `./case.submit`
Review log files to familiarize yourself with standard out. Start looking at your history files. Check your `CaseDocs/*_in` files. Were your changes applied?



CESM2 Tutorial: Basic Modifications: **HOMEWORK**

Set up now but don't submit until the last 15 minutes of class!

Assuming your b.day2.1 exercise ran successfully in class....

Continue Exercise 1 to produce total of 38 months. You have already run 1 month. This will run overnight in the cheyenne queues.

Your data will be used for the Practical Session tomorrow on *Diagnostics and Output*.

In `env_run.xml`:

1. Set `CONTINUE_RUN` to "TRUE"
2. Keep `STOP_OPTION` set to "nmonths"
3. Set `STOP_N` to "37"
4. Set `INFO_DBUG` to 1
5. Change back to the regular queue (or your job will not run overnight)

`./xmlchange --subgroup case.run JOB_QUEUE=regular`

do NOT cut and paste

`./xmlchange --subgroup case.st_archive JOB_QUEUE=regular`



`./xmlchange JOB_WALLCLOCK_TIME=12:00:00`

6. Submit (`./case.submit`) from your b.day2.1 case directory

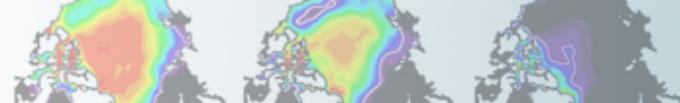


CESM1 Tutorial: Basic Modifications: Ex1: Example README.case

```
2017-08-02 11:55:18: ./create_newcase -case /glade/u/home/testusr1/cases/b.day2.1 -res f19_g17 -compset B1850 --project UESM0005
-----
2017-08-02 11:55:18: Compset longname is 1850_CAM60_CLM50%BGC-CROP_CICE_POP2%ECO_MOSART_CISM2%NOEVOLVE_WW3_BGC%BDRD
-----
2017-08-02 11:55:18: Compset specification file is /glade/p/cesm/tutorial/cesm2_0_alpha07b/cime_config/config_compsets.xml
-----
2017-08-02 11:55:18: Pes      specification file is /glade/p/cesm/tutorial/cesm2_0_alpha07b/cime_config/config_pes.xml
-----
2017-08-02 11:55:18: Component ATM is CAM cam6 physics:
-----
2017-08-02 11:55:18: ATM_GRID is 1.9x2.5
-----
.
.
.
=====
User modifications =====

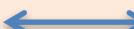
./xmlchange INFO_DBUG=2,STOP_N=2,STOP_OPTION=nmonths
./xmlchange --file env_run.xml --id LOGDIR --val "/glade/scratch/testusr1/logs/b.day2.1"
./xmlchange CCSM_CO2_PPMV=569.4

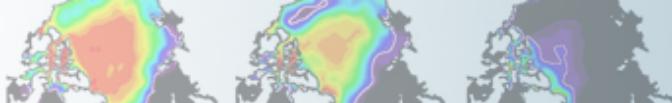
CESM version is cesm2_0_alpha07b
~
```



CESM2 Tutorial: Basic Modifications: Namelist variables: **user_nl_<model>**

- Not all changes can be made in env_run.xml.
- **user_nl_<model>** files appear in the case directory after ./case.setup has been invoked), i.e.

user_nl_cam		atmosphere
user_nl_cice		sea ice
user_nl_cism		land ice
user_nl_clm		land
user_nl_cpl		coupler (driver)
user_nl_mosart		river runoff
user_nl_pop		ocean
user_nl_ww		wave (ocean)



CESM2 Tutorial: Basic Modifications: Namelist tool: **preview_namelists**

- Insert namelist syntax for desired variable change into the appropriate file. To find the proper syntax and see all default namelist values, use **preview_namelists** to create the resolved namelists the model will use at runtime in your run directory (i.e. /glade/scratch/<user>/<case>/run/).
- In your case directory, type **./preview_namelists**
- cd to your run directory and view *_in files:

user_nl_cam	modifies	atm_in
user_nl_cice	modifies	ice_in
user_nl_cism	modifies	cism_in
user_nl_clm	modifies	Ind_in
user_nl_cpl	modifies	drv_in
user_nl_mosart	modifies	mosart_in
user_nl_pop	modifies	pop_in
user_nl_ww	modifies	wav_in



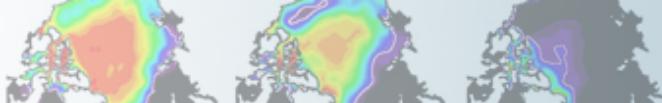
CESM2 Tutorial: Basic Modifications: Namelist tool: **preview_namelists**

Example: Decrease timestep in the ocean model by increasing dt_count from 24 steps per day to 48 steps per day.

1. Edit (vi or emacs) user_nl_pop
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 = 48`
3. Invoke *preview_namelists* again to verify change in your run directory and update the documentation pop_in file in CaseDocs.

Note: POP2 is now coupling every hour, rather than once per day (as in CESM1.2), so choice of dt_count is restricted to multiples of 24. CESM2 POP2 documentation is under construction, for syntax, see the CESM1.2 webpage, however, consult the bulletin board for further details on changing POP timestep.

http://www.cesm.ucar.edu/models/cesm1.2/pop2/doc/faq/#nml_general_change_dt



CESM2 Tutorial: Basic Modifications: **Exercise 2**

EXERCISE.2: BRANCH from the end of Exercise 1 and double CO₂ for atmosphere, land, and ocean. Double methane for the atmosphere. Run 1 month. Restart 1 month. Check your resolved namelists in your run directory.

1. `./create_newcase --case ~/cases/b.day2.2 -res f19_g17 --compset B1850`
2. `cd ~/cases/b.day2.2`
3. `./xmlchange RUN_TYPE=branch,RUN_REFCASE=b.day2.1,RUN_REFDATE=0001-02-01,STOP_OPTION=nmonths,STOP_N=1,RESUBMIT=1,CCSM_CO2_PPMV=569.4`
(Notes: no spaces between arguments; CCSM_CO2_PPMV changes land and ocean only).
`./xmlchange --subgroup case.run JOB_QUEUE=R1578615 ← tutorial only`
`./xmlchange --subgroup case.st_archive JOB_QUEUE=R1578615`
`./xmlchange JOB_WALLCLOCK_TIME=2:00:00`
4. `./case.setup`
5. Place a copy your restart files from your bday2.1 short term archive space to your bday2.2 run directory.

```
cp /glade/scratch/$LOGNAME/archive/b.day2.1/rest/0001-02-01-00000/* (space)
/glade/scratch/$LOGNAME/b.day2.2/run/
```

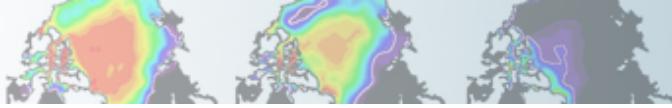


CESM2 Tutorial: Basic Modifications: **Exercise 2**

EXERCISE.2: continued

6. `./preview_namelists`
7. Check CaseDocs/atm_in for co2vmr syntax (and see default values). Cut and paste from atm_in file, (or type) into user_nl_cam:
$$\begin{aligned} \text{co2vmr} &= 569.4e-6 \\ \text{ch4vmr} &= 1583.2e-9 \end{aligned}$$
8. `./preview_namelists` (check atm_in, lnd_in, and pop_in to make sure your changes were implemented)
9. `/glade/p/cesm/tutorial/compile_node.csh`
`./case.build`
`exit`

Continued next page



CESM2 Tutorial: Basic Modifications: **Exercise 2**

EXERCISE.2: continued

10. `./case.submit`
11. Review queues and log files. Where are your logs files (where are they)? How fast does the first month run (timing files are in the case directory and the run directory)? Was the second month resubmitted? What is the value of “CONTINUE_RUN” initially? (Check before the model finishes the first month). What is the value after resubmission? Read the `env_run.xml` documentation for explanation! (Hint: see “RESUBMIT_SETS_CONTINUE_RUN”).
12. After the job completes, go to the short term archive space and explore.



CESM2 Tutorial: Basic Modifications: **env_run.xml vs. user_nl_<model>**

What method is best for changes?

env_run.xml:

- Run_type specification (startup, hybrid, branch, continue)
- Runtime variables (stop_option, resubmits, etc.)
- CO2 changes for land and ocean

user_nl_<model>:

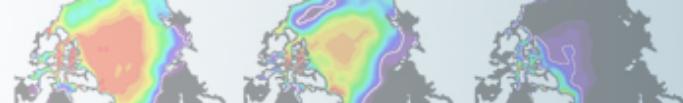
- Swapping out a default inputdata set for a home-grown dataset
- Namelist changes for component models



CESM2 Tutorial: Basic Modifications: `env_run.xml` vs. `user_nl_<model>`

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.



CESM2 Tutorial: Basic Modifications: **Physics Time Step Changes**

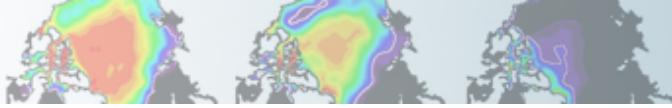
Where and When to Change Time Steps

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.

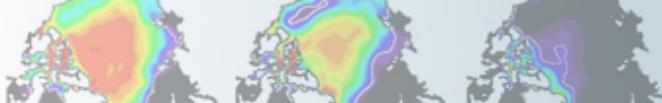
CAM/CLM: ATM_NCPL in env_run.xml.

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



CESM2 Tutorial: Basic Modifications: **Physics Time Step Changes**

1. **CAM6 time step** is set by **ATM_NCPL** in **env_run.xml** and specifies the number of **coupling intervals per day** between the atmosphere/land and the coupled system. Based on ATM_NCPL, the scripts will automatically compute the time step for the atmosphere and land (**DTIME**) and populate the namelist files accordingly.
2. **CLM5 time step** = CAM6 time step; this is automatically set with the CAM time step via ATM_NCPL. You cannot set this separately.
3. **POP2 time step** is changed in the **user_nl_pop** file and is based on **OCN_NCPL** (found in **env_run.xml**), “**dt_count**”, and “**dt_option**”. The default **dt_option** is “**steps_per_day**”.
4. **CICE5 time step** is set by the coupling interval variable **ICE_NCPL** found in **env_run.xml**. Note that **ICE_NCPL** = **ATM_NCL**.



CESM2 Tutorial: Basic Modifications: **More Exercises**

Exercise.3: Hybrid start a fully coupled for Pre-Industrial conditions. Use files from specified directory for your reference case. Change your orbital parameters to use condition from the 1600 AD and change the physics time step in the atmosphere and land to 1200 seconds (default is 1800). (Note: this is an exercise and does not represent any historical period). Run 5 days (default).

1. `./create_newcase --case ~/cases/b.day2.3 --res f19_g17 --compset B1850`

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

3. `./xmlchange RUN_TYPE=hybrid,RUN_REFCASE=b.day2.hybridstart,RUN_REFDATE=0002-01-01,ATM_NCPL=72`

(Why is ATM_NCPL = 72, do the math).

`./xmlchange --subgroup case.run JOB_QUEUE=R1578615` ← tutorial only

`./xmlchange --subgroup case.st_archive JOB_QUEUE=R1578615`

`./xmlchange JOB_WALLCLOCK_TIME=2:00:00`

4. `./case.setup`

5. Place a copy of the specified restart files from your bday2.1 short term archive space to your bday2.2 run directory. (These are files we created for you to practice a hybrid start).

`cp /glade/scratch/shields/archive/b.day2.hybridstart/rest/0002-01-01-00000/* (space)`
`/glade/scratch/$LOGNAME/b.day2.3/run/.`

Continued next page



CESM2 Tutorial: Basic Modifications: More Exercises

Exercise.3: Continued:

6. `./preview_namelists`

(What is the difference between the cam and clm initial files in this Exercise versus the branch files in Exercise 2? Hint: Check *ncdata* and *cam_branch_file* in *atm_in* and *finidat* and *nrevsn* in *Ind_in*).

7. Edit *user_nl_cpl*, after comments, add line: *orb_iyear=1600* (Hint, check *drv_in*).

Update *README.case*

8. Optional: `./preview_namelists`

9. `/glade/p/cesm/tutorial/compile_node.csh`

case.build

exit

10. `./case.submit`

11. Check logs files. The coupler log file should confirm your orbital changes.



CESM2 Tutorial: Basic Modifications: **More Exercises**

Exercise.4: Clone case from Exercise.3. 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 parameterization in CLM. Run 5 days (default).

1. `./create_clone --case ~/cases/b.day2.4 --clone ~/cases/b.day2.3`
2. 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.`
3. `./xmlchange --subgroup case.run JOB_QUEUE=R1578615 ← tutorial only`
`./xmlchange --subgroup case.st_archive JOB_QUEUE=R1578615`
`./xmlchange JOB_WALLCLOCK_TIME=2:00:00`



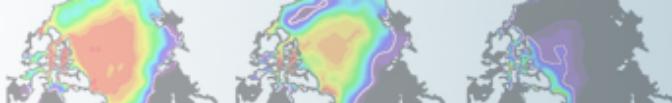
CESM2 Tutorial: Basic Modifications: **More Exercises**

Exercise.4: continued

4. Edit user_nl_pop and add: *sw_absorption_type = 'jerlov'*
5. Edit user_nl_clm and add: *urban_hac = 'OFF'*
6. Update your README.case file to document your changes.
7. *./case.setup*
8. *cp /glade/scratch/shields/archive/b.day2.hybridstart/rest/0002-01-01-00000/* (space)*
/glade/scratch/\$LOGNAME/b.day2.4/run/
9. *./preview_namelists*

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 preview_namelist to update your CaseDocs.

10. Log onto compute node, then, *.case.build*, then exit compute node.
11. *./case.submit*
12. If you want, you can start to look at the history output. Only the ocn will have daily output to view, the default is monthly for most model components. (Use ncview). To use ncview, you will need to type “module load ncview” on your command line. Where is the short term history output located? Go back to earlier exercises to explore monthly history files.
13. Compare b.day2.4 ocn history data to b.day2.3 data. (Use ncdiff). To use ncdiff, you will need to type “module load nco” on your command line.



CESM2 Tutorial: Basic Modifications: **More Exercises**

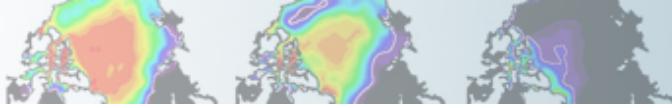
Exercise: On your own...

Continue Exercise.4 (restart) but reduce the snow albedoes in the ice model by half. (This is done in user_nl_cice). You do not need to recompile. Use the restart files that are already in the run directory and run 5 more days.

Know what you are changing. Look up information on namelist variables in the documentation.

https://ncar.github.io/CICE/doc/build/html/users_guide/ice_nml_var.html?highlight=snow%20albedo

Be sure to update your README.case file to keep track of your changes. The model will run regardless of whether or not you remember to include all of your changes. Check your resolved 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, experiment with other namelist variables changes, and compare history files.



CESM2 Tutorial: Basic Modifications: **Bottom Line**

What user-modified files are actually used at runtime?

`./case.setup (or ./preview_namelists)` → \$RUNDIR/atm_in
\$RUNDIR/lnd_in
\$RUNDIR/pop_in
\$RUNDIR/ice_in
\$RUNDIR/drive_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, software_environment.txt

CaseDocs, CaseStatus, LockedFiles, README.case

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



CESM2 Tutorial: Basic Modifications: [env_batch.xml](#)

env_batch.xml is where you can change the CESM2 default values related to job batch submissions.

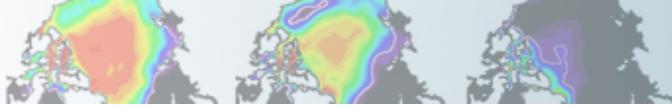
After the tutorial when you are running at your home institution, you may want to change the default queues, wallclock time, or control the project number after you have set up a case.

Wallclock time: <entry id="JOB_WALLCLOCK_TIME" value="12:00:00">
 note: use subgroup <group id="case.run">

Job queue: <entry id="JOB_QUEUE" value="regular">

Project number: <entry id="PROJECT" value="UESM0005">

To find your total wallclock time after running, either check timing file or your standard output file (i.e., b.day2.run.onnnnnnnn) in your case directory.



CESM2 Tutorial: Basic Modifications: **Log Files**

Log Files:

During model execution:

After model completion:

atm.log.yyddmm-nnnnnnn.gz
lnd.log.yyddmm-nnnnnnn.gz
ocn.log.yyddmm-nnnnnnn.gz
ice.log.yyddmm-nnnnnnn.gz
cpl.log.yyddmm-nnnnnnn.gz
rof.log.yyddmm-nnnnnnn.gz
cesm.log.yyddmm-nnnnnnn.gz

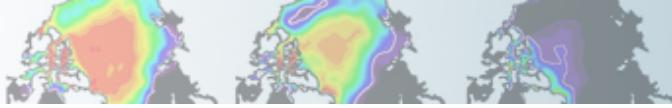
Model runtime standard output

\$RUNDIR/*

\$LOGDIR/*

Files are gzipped after model completion.
Restore by typing *gunzip <logfile>*.

yyddmm = year, month, day
nnnnnn = time id stamp



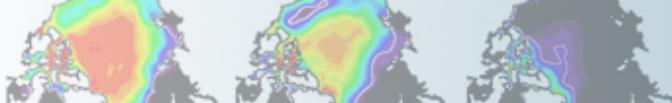
CESM2 Tutorial: Basic Modifications: Other Tips

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 the model is using what you think it is using!

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!



CESM2 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 CESM2 can be found at: <http://www.cesm.ucar.edu/models/cesm2.0/timing/index.html>



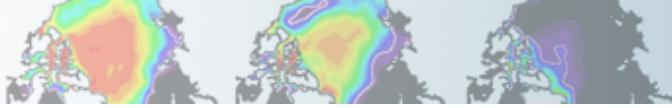
CESM2 Tutorial: Basic Modifications: **HELP!**

Finding Help...

1. Documentation:

<http://www.cesm.ucar.edu/models/cesm2.0/>

2. DiscussCESM: <http://bb.cgd.ucar.edu>



CESM2 Tutorial: Basic Modifications



Have Fun!!!

