CESM Tutorial

Introduction to CESM2

NCAR Climate and Global Dynamics Laboratory

Alice Bertini

CESM Software Engineering Group

NCAR is sponsored by the National Science Foundation

Outline

- The CESM project webpage
- CESM2 webpage
- CESM2 Quickstart Guide
- Downloading CESM
- CIME and the Case Control System
- Creating & Running a Case
- Getting More Help
- Review of Hands-on Exercises

CESM Web Page

http://www.cesm.ucar.edu

Live Demo…

Take-away points

- CESM project information
- Working Groups information
- Community Project information
  - Large Ensemble
  - Last Millennium Ensemble
- What version of the model should you use?
  - Supported model releases – Symantec versioning
  - Diagnostics plots for supported configurations

CESM2 Web Page

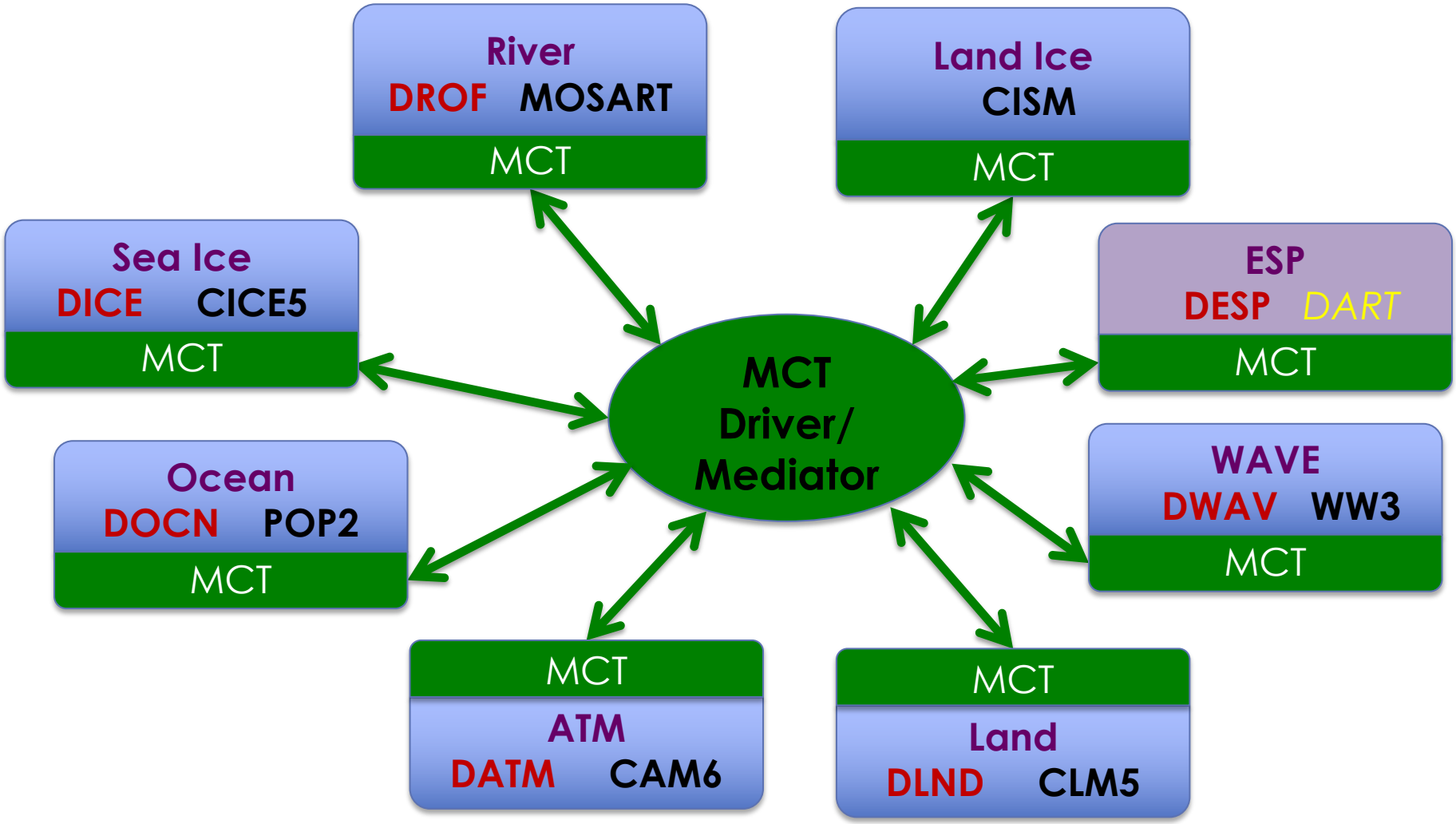
http://www.cesm.ucar.edu/models/cesm2

Live Demo…

Take-away points

- Release notes and supported tags
- Download instructions - no user registration required!
- Scientifically validated configurations
- Port verification using statistical ensemble test
- On-line documentation – Quickstart, CIME
- Prognostic component details and documentation
- User namelist and Case Control System XML configuration settings
- Configuration definitions (compsets, grids and machines)
- Timing information

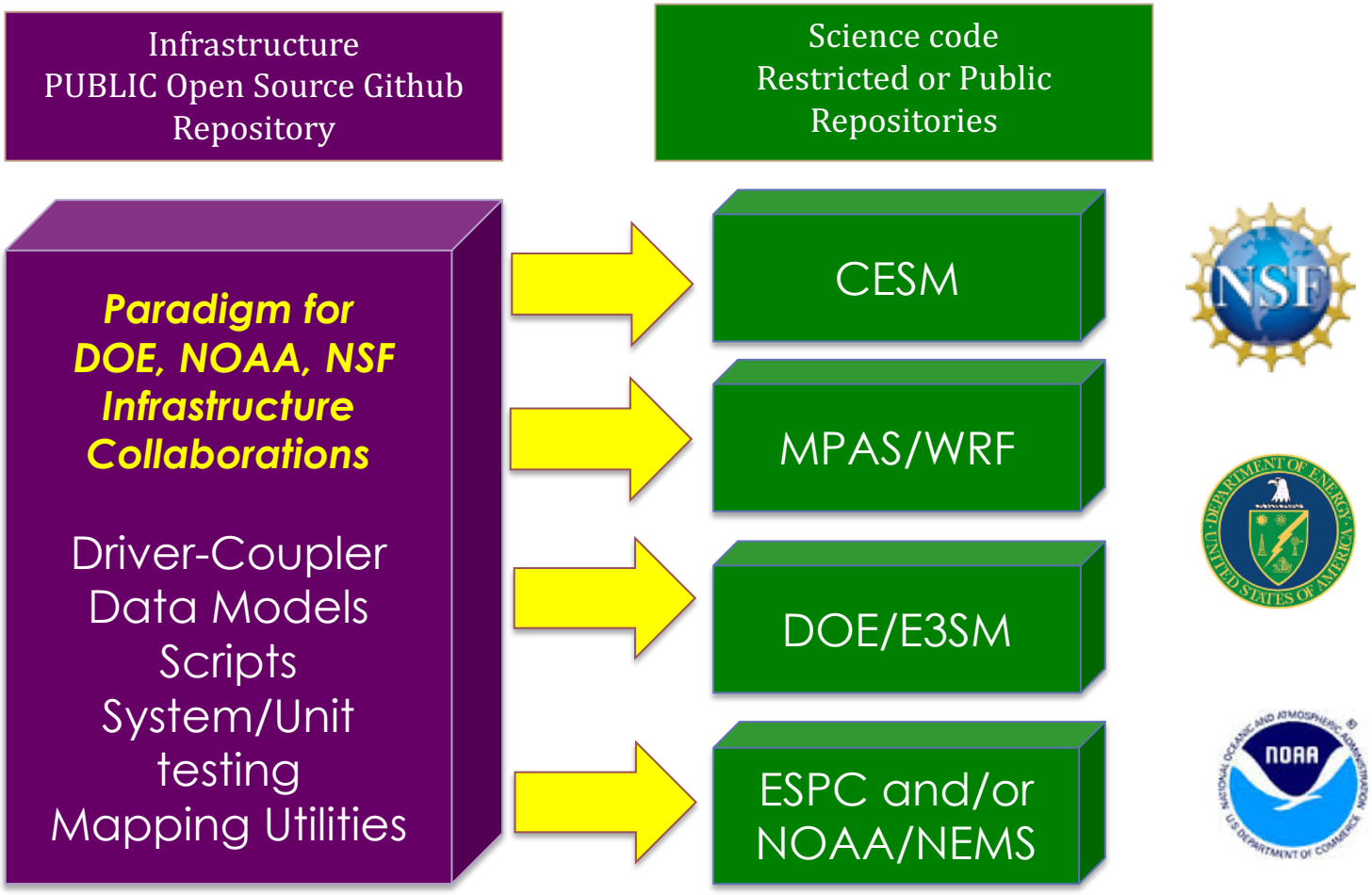
Current CESM2 Coupling – data components permit flexible activation/deactivation of feedbacks



MCT – Model Coupling Toolkit

Coupling Infrastructure for Modeling Earth (CIME)

(new python-based CESM infrastructure)



addresses needs of multiple efforts

CESM2 Quickstart Workflow

https://escomp.github.io/cesm/release-cesm2/

- One-Time Setup Steps
  - Download the CESM code
  - Create an Input Data Root Directory
  - Porting
- Creating & Running a Case
  - Create a New Case
  - Invoke case.setup
  - Build the Executable with case.build
  - Run the Model with case.submit
  - Review Output Data

Download CESM

Note: Try this at home!
The tutorial setup has already done this step for you.

```
[aliceb@cheyenne5:dev]>git clone -b release-cesm2.0.0 https://github.com/ESCOMP/cesm.git cesm2.0.0
Cloning into 'cesm2.0.0'...
remote: Counting objects: 1396, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 1396 (delta 2), reused 2 (delta 0), pack-reused 1390
Receiving objects: 100% (1396/1396), 1.47 MiB | 1.94 MiB/s, done.
Resolving deltas: 100% (670/670), done.
Note: checking out '389886eefc7f28bf86fe71ed383b7351e5837f26'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

[aliceb@cheyenne5:dev]>cd cesm2.0.0
Directory: /glade/work/aliceb/sandboxes/dev/cesm2.0.0
[aliceb@cheyenne5:cesm2.0.0]>ls
ChangeLog  ChangeLog_template  Copyright  Externals.cfg  LICENSE.txt  README.rst  cime_config  doc  manage_externals
[aliceb@cheyenne5:cesm2.0.0]>
```

- Blue items are directories
- Green items are executable files
- cime_config contains CESM specific configuration information for CIME
- manage_externals contains utilities for downloading component models which are defined in the Externals.cfg file

Checkout all the model components

Note: Try this at home!
The tutorial setup has already done this step for you.

```
[aliceb@cheyenne5:cesm2.0.0]>pwd
/gpfs/fs1/work/aliceb/sandboxes/dev/cesm2.0.0
[aliceb@cheyenne5:cesm2.0.0]>./manage_externals/checkout_externals
Processing externals description file : Externals.cfg
Checking status of externals: clm, mosart, ww3, cime, cice, pop, cism, rtm, cam,
Checking out externals: clm, mosart, ww3, cime, cice, pop, cism, rtm, cam,
Processing externals description file : Externals_CLM.cfg
Checking out externals: fates, ptclm,
Processing externals description file : Externals_POP.cfg
Checking out externals: cvmix, marbl,
Processing externals description file : Externals_CISM.cfg
Checking out externals: source_cism,

[aliceb@cheyenne5:cesm2.0.0]>
```

Download listing of CESM

Note: I've switched paths to the pre-downloaded tutorial version of the model

```
[aliceb@cheyenne5:cesm2.0.0_tutorial]>pwd
/gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial
[aliceb@cheyenne5:cesm2.0.0_tutorial]>ls -1
ChangeLog
ChangeLog_template
Copyright
Externals.cfg
LICENSE.txt
README.rst
cime
cime_config
components
doc
manage_externals
[aliceb@cheyenne5:cesm2.0.0_tutorial]>
```

Components listing

```
[aliceb@cheyenne5:cesm2.0.0_tutorial]>cd components/
Directory: /glade/p/cesm/tutorial/cesm2.0.0_tutorial/components
[aliceb@cheyenne5:components]>ls -1
cam
cice
cism
clm
mosart
pop
rtm
ww3
[aliceb@cheyenne5:components]>
```

- cam ← Community Atmosphere Model
- cice ← Community Sea Ice Model
- cism ← Community Ice Sheet Model
- clm ← Community Land Model
- mosart ← Model for Scale Adaptive River Transport
- pop ← Parallel Ocean Program
- rtm ← River Transport Model
- ww3 ← WaveWatch3

```
[aliceb@cheyenne5:components]>cd cam
Directory: /glade/p/cesm/tutorial/cesm2.0.0_tutorial/components/cam
[aliceb@cheyenne5:cam]>ls -1
SVN_EXTERNAL_DIRECTORIES
bld
chem_proc
cime_config
doc
src
test
tools
[aliceb@cheyenne5:cam]>
```

```
[aliceb@cheyenne5:components]>cd clm
Directory: /glade/p/cesm/tutorial/cesm2.0.0_tutorial/components/clm
[aliceb@cheyenne5:clm]>ls -1
CODE_OF_CONDUCT.md
Copyright
Externals.cfg
Externals_CLM.cfg
LICENSE
README
README.rst
README_EXTERNALS.rst
bld
cime_config
doc
manage_externals
parse_cime.cs.status
src
src_clm40
test
tools
[aliceb@cheyenne5:clm]>
```

CIME – Common Infrastructure for Modeling the Earth

https://github.com/ESMCI/cime

Live demo…

Take-away points

- Coupling infrastructure
- Data and stub models for satisfying driver/mediator requirements
- Testing infrastructure
- Python scripts and XML configuration files for the Case Control System

```
[aliceb@cheyenne5:cesm2.0.0_tutorial]>cd cime
Directory: /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime
[aliceb@cheyenne5:cime]>ls -1
CMakeLists.txt
CONTRIBUTING.md
ChangeLog
ChangeLog_template
LICENSE.TXT
README.md
config
doc
index.html
scripts
src
tools
utils
[aliceb@cheyenne5:cime]>
```

XML

eXtensible Markup Language

- XML is used to define documents with a standard format that can be read by any XML-compatible application.
- In CESM, XML is used as a database to store configuration and control settings.

```
<entry id="CIMEROOT" value="/gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime">
  <type>char</type>
  <desc>full pathname of CIME source root directory</desc>
</entry>
```

- CESM Conventions - ***$name*** can be either a shell environment variable or a CESM **XML id** name.
- For example, CIMEROOT refers to the director path location of the cime directory in the CESM checkout.

…. But the shell does not know about $CIMEROOT

For software engineers:

CIME uses XML files as the data store for configuration and variable settings and a set of python modules to parse those XML files and create an experiment case specific environment for setup, build, and batch submission.

CIME Documentation
http://esmci.github.io/cime

```
[aliceb@cheyenne5:cime]>cd scripts
Directory: /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts
[aliceb@cheyenne5:scripts]>ls -1
Tools
create_clone
create_newcase
create_test
data_assimilation
fortran_unit_testing
lib
query_config
query_testlists
tests
[aliceb@cheyenne5:scripts]>
```

Don't be afraid to explore in these directories in the lab session this afternoon!

Work Flow: Super Quick Start

CESM2 can be run with a set of 4 commands

Set of commands to build and run the model on supported machine cheyenne

```
# one time step – create a directory to store your experiment case roots
mkdir ~/cases

# go into scripts subdirectory of cime
cd /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts

# create a new case in the directory "cases" in your home directory
./create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset B1850

# go into the case you just created in the last step
cd ~/cases/b.day1.0

setenv CESM_BLD_TEMPLATE  /glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (tcsh)
export CESM_BLD_TEMPLATE=/glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (bash)

# invoke case.setup
./case.setup

# build the executable (cheyenne specific commands!)
qcmd -- ./case.build

# submit your run to the batch queue
./case.submit
```

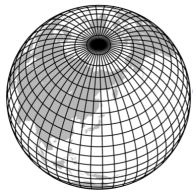
Create a new case experiment

In the cime/scripts directory, create_newcase is the tool that generates a new case.

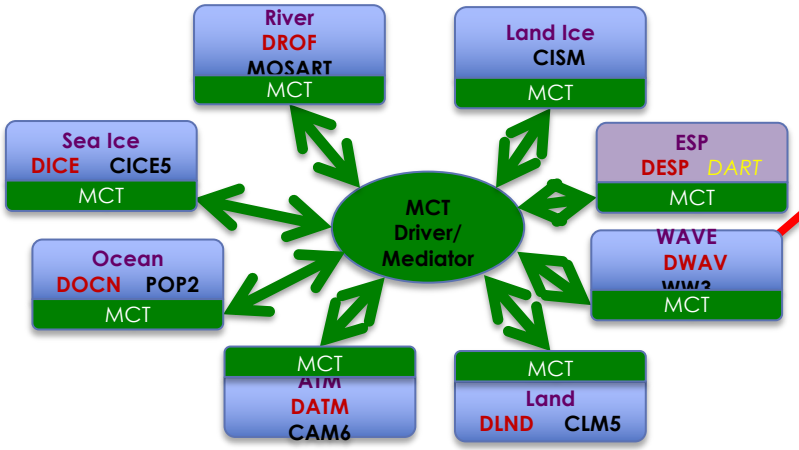
create_newcase requires 3 arguments

What is the casename ?

Which resolution?



Which model configuration ? Which set of components ?



Which machine are you running on?

NOTES:

- for all user scripts, you can run the script name followed by the --h or –-help argument to see help documentation and a list of all command line arguments.
- Double dashes "--" are now required with command line arguments
- --mach is not required on CESM supported machines

create_newcase arguments

create_newcase requires 3 arguments

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

What is the casename ?

case specifies the name and location of the case being created

~/cases/b.day1.0

NOTES:

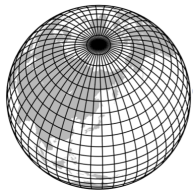
- experiment case naming conventions for CESM are described on the CESM2 webpage at URL:
  http://www.cesm.ucar.edu/models/cesm2/naming_conventions.html
- If a path proceeding the case name is not specified, then the case is created as a subdirectory in the $CIMEROOT/scripts directory.

create_newcase arguments

create_newcase requires 3 arguments

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

Which resolution?



res specifies the model resolution (or grid)

Grid naming convention

Each model resolution can be specified by its alias or long name.

Example of equivalent alias and long name:
- alias: f19_g17 (atm/lnd_ocn/ice)
- long name: a%1.9x2.5_l%1.9x2.5_oi%gx1v7_r%r05_g%gland4_w%ww3a_m%gx1v7

atm — lnd — ocn/ice grid — river — lnd-ice — wave — ocn-ice mask

CESM2 Supported Grid Definitions

http://www.cesm.ucar.edu/models/cesm2/config/grids.html

CIMEROOT/scripts/query_config --grids --long

Live demo…

create_newcase arguments

create_newcase requires 3 arguments

```
create_newcase --case ~/cases/b.day1.0 --res T31_g37 --compset B1850
```

Which component set ?

compset specifies the "component set"

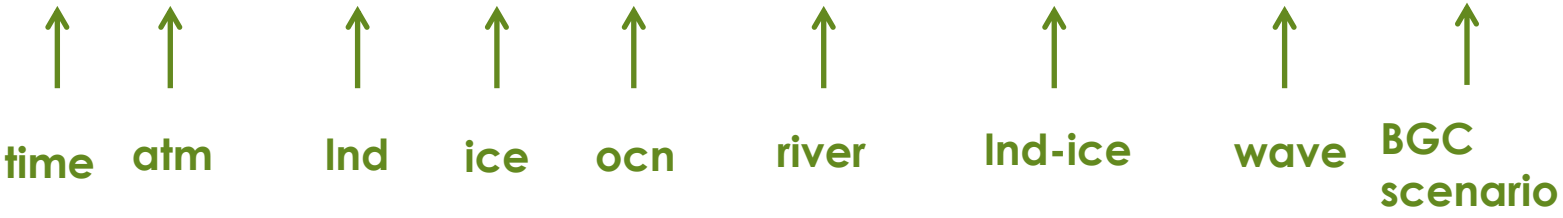
Component set specifies component models, forcing scenarios and physics options for those models

compset naming convention

Each model compset can be specified by its alias or long name.

Example of equivalent alias and long name:

- **alias: B1850**
- **long name = 1850_CAM60_CLM50%BGC_CICE_POP2%ECO_MOSART_CISM2%NOEVOLVE_WW3_BGC%BDRD**



CESM2 Supported compset Definitions

http://www.cesm.ucar.edu/models/cesm2/config/compsets.html

CIMEROOT/scripts/query_config –compsets

Live demo…

Take-away points

- compsets are defined by different model components and cime
- Some compsets are scientifically supported and/or tested while some are only defined
- compsets determine which grid is required

Result of running create_newcase

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

```
testusr1@cheyenne1:/glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts> ./create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset B1850
Compset longname is 1850_CAM60_CLM50%BGC-CROP_CICE_POP2%ECO_MOSART_CISM2%NOEVOLVE_WW3_BGC%BDRD
Compset specification file is /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/../cime_config/config_compsets.xml
Compset forcing is 1850
Com forcing is Biogeochemistry intercomponent  with diagnostic CO2
ATM component is CAM cam6 physics:
LND component is clm5.0:BGC (vert. resol. CN and methane) with prognostic crop:
ICE component is Sea ICE (cice) model version 5
OCN component is POP2 Ecosystem
ROF component is MOSART: MOdel for Scale Adaptive River Transport
GLC component is cism2 (default, higher-order, can run in parallel):cism ice evolution turned off (this is the standard configuration unless you're explici
WAV component is Wave Watch
ESP component is
Pes     specification file is /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/../cime_config/config_pes.xml
Compset specific settings: name is RUN_STARTDATE and value is 0001-01-01
Machine is cheyenne
Pes setting: grid match    is a%1.9x2.5.+l%1.9x2.5.+oi%gx1
Pes setting: machine match is cheyenne
Pes setting: compset_match is CAM.+CLM.+CICE.+POP.+
Pes setting: grid          is a%1.9x2.5_l%1.9x2.5_oi%gx1v7_r%r05_g%gland4_w%ww3a_m%gx1v7
Pes setting: compset       is 1850_CAM60_CLM50%BGC-CROP_CICE_POP2%ECO_MOSART_CISM2%NOEVOLVE_WW3_BGC%BDRD
Pes setting: tasks       is {'NTASKS_ATM': 288, 'NTASKS_ICE': 108, 'NTASKS_CPL': 288, 'NTASKS_LND': 144, 'NTASKS_WAV': 36, 'NTASKS_ROF': 40, 'NTASKS_OCN':
Pes setting: threads     is {'NTHRDS_ICE': 1, 'NTHRDS_ATM': 1, 'NTHRDS_ROF': 1, 'NTHRDS_LND': 1, 'NTHRDS_WAV': 1, 'NTHRDS_OCN': 1, 'NTHRDS_CPL': 1, 'NTHRDS
Pes setting: rootpe      is {'ROOTPE_OCN': 288, 'ROOTPE_LND': 0, 'ROOTPE_ATM': 0, 'ROOTPE_ICE': 144, 'ROOTPE_WAV': 252, 'ROOTPE_CPL': 0, 'ROOTPE_ROF': 0, '
Pes setting: pstrid      is {}
Pes other settings: {}
Pes comments: about 12ypd expected
 Compset is: 1850_CAM60_CLM50%BGC-CROP_CICE_POP2%ECO_MOSART_CISM2%NOEVOLVE_WW3_BGC%BDRD
 Grid is: a%1.9x2.5_l%1.9x2.5_oi%gx1v7_r%r05_g%gland4_w%ww3a_m%gx1v7
 Components in compset are: ['cam', 'clm', 'cice', 'pop', 'mosart', 'cism', 'ww3', 'sesp', 'drv', 'dart']

*****************************************************************************************************************************************************
This compset and grid combination is not scientifically supported, however it is used in 7 tests.
*****************************************************************************************************************************************************

Using project from config_machines.xml: UESM0006
No charge_account info available, using value from PROJECT
Using project from config_machines.xml: UESM0006
cesm model version found: cesm2.0.0
Batch_system_type is pbs
job is case.run USER_REQUESTED_WALLTIME None USER_REQUESTED_QUEUE None
job is case.st_archive USER_REQUESTED_WALLTIME None USER_REQUESTED_QUEUE None
 Creating Case directory /glade/u/home/testusr1/cases/b.day1.0
testusr1@cheyenne1:/glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts>
```

grid info

PE layouts

compset longname

Machine specific info

Success! This is the CASEROOT directory

CASEROOT directory structure after running create_newcase

```
[aliceb@cheyenne2:b.day1.0]>pwd
/gpfs/u/home/testusr1/cases/b.day1.0
[aliceb@cheyenne2:b.day1.0]>ls -1
Buildconf
LockedFiles
README.case
SourceMods
Tools
case.build
case.cmpgen_namelists
case.qstatus
case.setup
case.submit
check_case
check_input_data
env_archive.xml
env_batch.xml
env_build.xml
env_case.xml
env_mach_pes.xml
env_mach_specific.xml
env_run.xml
pelayout
preview_namelists
preview_run
xmlchange
xmlquery
[aliceb@cheyenne2:b.day1.0]>
```

User defined source code modifications (advanced!)

script to check required input data files and download them, if necessary

User Customizable case XML files

script to change XML settings

script to query XML settings

CASEROOT env_*.xml files

env_*.xml contains variables used by scripts -- some can be changed by the user

env_archive.xml	specifies rules for short-term archival script case.st_archive
env_batch.xml	set by create_newcase to define batch specific settings used script case.submit
env_build.xml	specifies build information used by script case.build
env_case.xml	set by create_newcase and cannot be modified
env_mach_pes.xml	specifies PE layout of components used by script case.run
env_mach_specific.xml	specifies machine specific information used by script case.build
env_run.xml	–sets run time information (such as length of run, frequency of restarts, …) User interacts with this file most frequently

- **To query a variable in an xml file use script xmlquery**
- **To modify a variable in an xml file use script xmlchange**

  ./xmlchange STOP_N=20

NOTE: You can edit the XML files manually but it is recommended that you use the xmlchange script to ensure that the XML schema is preserved!

CASEROOT/xmlchange

```
[aliceb@cheyenne2:b.day1.0]>./xmlchange --help
usage: xmlchange [-h] [-d] [-v] [-s] [--caseroot CASEROOT] [--append]
                 [--subgroup SUBGROUP] [--id ID] [--val VAL] [--file FILE]
                 [--delimiter DELIMITER] [--dryrun] [--noecho] [-f]
                 [-loglevel LOGLEVEL]
                 [listofsettings]

Allows changing variables in env_*xml files via a command-line interface.

This provides two main benefits over editing the xml files by hand:
  - Settings are checked immediately for validity
  - Settings are echoed to the CaseStatus file, providing a "paper trail" of
    changes made by the user.

Examples:

   To set a single variable:
      ./xmlchange REST_N=4

   To set multiple variables at once:
      ./xmlchange REST_OPTION=ndays,REST_N=4

   Alternative syntax (no longer recommended, but supported for backwards
   compatibility; only works for a single variable at a time):
      ./xmlchange --id REST_N --val 4
```

etc…..

Note: argument –subgroup applies change to XML variable in XML element named <group>

CESM2 CASEROOT XML settings

https://www.cesm.ucar.edu/models/cesm2/component_settings

Live demo…

Take-away points

- Every component defines its own XML settings in the CASEROOT env_*.xml files

Work Flow: Super Quick Start

```
# one time step – create a directory to store your experiment case roots
mkdir ~/cases

# go into scripts subdirectory of cime
cd /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts

# create a new case in the directory "cases" in your home directory
./create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset B1850

# go into the case you just created in the last step
cd ~/cases/b.day1.0

setenv CESM_BLD_TEMPLATE  /glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (tcsh)
export CESM_BLD_TEMPLATE=/glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (bash)

# invoke case.setup
./case.setup

# build the executable (cheyenne specific commands!)
qcmd -- ./case.build

# submit your run to the batch queue
./case.submit
```

case.setup

Notice the "./" before any command run in the CASEROOT! Run **./case.setup –help** *in the lab session.*

```
testusr1@cheyenne1:~/cases/b.day1.0> ./case.setup
Setting resource.RLIMIT_STACK to -1 from (307200000, -1)
/glade/u/home/testusr1/cases/b.day1.0/env_mach_specific.xml already exists, delete to repla
job is case.run USER_REQUESTED_WALLTIME None USER_REQUESTED_QUEUE None
Creating batch scripts
Writing case.run script from input template /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/ci
Creating file .case.run
Writing case.st_archive script from input template /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tuto
Creating file case.st_archive
Creating user_nl_xxx files for components and cpl
If an old case build already exists, might want to run 'case.build --clean' before building
You can now run './preview_run' to get more info on how your case will be run
testusr1@cheyenne1:~/cases/b.day1.0>
```

case.setup creates:

- RUNDIR and EXEROOT directories
- user_nl_xxx files – user customizable component namelist files
- scripts case.run, case.st_archive, and Macros.make file
- hidden files .case.run and .env_mach_specific.* which can help with debugging
- CaseDocs directory - NOTE: these files should not be edited!

CESM2 Namelist files and settings

https://www.cesm.ucar.edu/models/cesm2/component_settings

Live demo…

Take-away points

- Every component defines its own namelist file in the RUNDIR by combining the default component namelist with the CASEROOT user_nl_[comp] file.
- The CASEROOT preview_namelist script can be used to check user defined namelist settings in the user_nl_[comp] files.

CASEROOT, EXEROOT and RUNDIR
case.setup updates files in the CASEROOT and creates these machine dependent directories

```
testusr1@cheyenne1:~/cases/b.day1.0> ./xmlquery RUNDIR,EXEROOT

Results in group build_def
        EXEROOT: /glade/scratch/testusr1/b.day1.0/bld

Results in group run_desc
        RUNDIR: /glade/scratch/testusr1/b.day1.0/run
testusr1@cheyenne1:~/cases/b.day1.0>
```

Work Flow: Super Quick Start

Set of commands to build and run the model on a supported machine: "cheyenne"

```
# one time step – create a directory to store your experiment case roots
mkdir ~/cases

# go into scripts subdirectory of cime
cd /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts

# create a new case in the directory "cases" in your home directory
./create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset B1850

# go into the case you just created in the last step
cd ~/cases/b.day1.0

setenv CESM_BLD_TEMPLATE  /glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (tcsh)
export CESM_BLD_TEMPLATE=/glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (bash)

# invoke case.setup
./case.setup

# build the executable (cheyenne specific commands!)
qcmd -- ./case.build

# submit your run to the batch queue
./case.submit
```

The "qcmd --" is for Cheyenne only!

The CESM_BLD_TEMPLATE setting speeds up the build time for this tutorial only!

Build the Model

- Modifications before build
  - Change env_build.xml values *before* running case.build
  - Introduce any modified source code in SourceMods/ before building

- To completely rebuild, run case.build --clean-all first

- The case.build script
  - Checks and consolidates the user namelists files into single nl files
  - Builds the individual component libraries and model executable

- If any inputdata is missing,
  - Build aborts, but provides a list of missing files
  - Run ./check_input_data --download to acquire missing data
  - This will use svn or gridftp to put required data in the inputdata directory defined by XML variable DIN_LOC_ROOT
  - Then re-run case.build script

NOTE: On NCAR machine Cheyenne, the case.build script should always be called as follows:

qcmd -- ./case.build

This compiles the model on a compute node reducing the load on the login nodes and prevents a timeout.

Running the case.build Script

```
testusr1@cheyenne1:~/cases/b.day1.0> export CESM_BLD_TEMPLATE=/glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld
testusr1@cheyenne1:~/cases/b.day1.0> qcmd -- ./case.build
Submitting command to PBS using account UESM0006:
    ./case.build

Waiting for job 1486457.chadmin1 to start ...


Building case in directory /gpfs/u/home/testusr1/cases/b.day1.0
sharedlib_only is False
model_only is False
Setting resource.RLIMIT_STACK to -1 from (-1, -1)
Copying bld directory from /glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld
Generating component namelists as part of build
Creating component namelists
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/cam//cime_config/buildnml
     ...calling cam buildcpp to set build time options
CAM namelist copy: file1 /glade/u/home/testusr1/cases/b.day1.0/Buildconf/camconf/atm_in file2 /glade/scratch/testusr1/b.day1.0/run/atm_in
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/clm//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/cice//cime_config/buildnml
     ...calling cice buildcpp to set build time options
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/pop//cime_config/buildnml
     ...calling pop buildcpp to set build time options

   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/mosart//cime_config/buildnml
   Running /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/cism//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/ww3//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/components/stub_comps/sesp/cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/drivers/mct/cime_config/buildnml
   NOTE: ignoring setting of glc2ice_rmapname=idmap in seq_maps.rc
Finished creating component namelists
Building gptl with output to file /glade/scratch/testusr1/b.day1.0/bld/gptl.bldlog.180804-200913
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/build_scripts/buildlib.gptl
Building mct with output to file /glade/scratch/testusr1/b.day1.0/bld/mct.bldlog.180804-200913
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/build_scripts/buildlib.mct
Building pio with output to file /glade/scratch/testusr1/b.day1.0/bld/pio.bldlog.180804-200913
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/build_scripts/buildlib.pio
Building csm_share with output to file /glade/scratch/testusr1/b.day1.0/bld/csm_share.bldlog.180804-200913
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/build_scripts/buildlib.csm_share
        - Building clm4_5/clm5_0 Library
Building lnd with output to /glade/scratch/testusr1/b.day1.0/bld/lnd.bldlog.180804-200913
clm built in 1.202620 seconds
Building atm with output to /glade/scratch/testusr1/b.day1.0/bld/atm.bldlog.180804-200913
Building ice with output to /glade/scratch/testusr1/b.day1.0/bld/ice.bldlog.180804-200913
Building ocn with output to /glade/scratch/testusr1/b.day1.0/bld/ocn.bldlog.180804-200913
Building rof with output to /glade/scratch/testusr1/b.day1.0/bld/rof.bldlog.180804-200913
Building glc with output to /glade/scratch/testusr1/b.day1.0/bld/glc.bldlog.180804-200913
Building wav with output to /glade/scratch/testusr1/b.day1.0/bld/wav.bldlog.180804-200913
Building esp with output to /glade/scratch/testusr1/b.day1.0/bld/esp.bldlog.180804-200913
mosart built in 1.208739 seconds
sesp built in 2.152750 seconds
pop built in 2.684563 seconds
ww built in 3.365579 seconds
Component glc build complete with 3 warnings
cice built in 137.854485 seconds
cam built in 137.854880 seconds
cism built in 137.856576 seconds
Building cesm with output to /glade/scratch/testusr1/b.day1.0/bld/cesm.bldlog.180804-200913
Time spent not building: 44.027308 sec
Time spent building: 169.542823 sec
MODEL BUILD HAS FINISHED SUCCESSFULLY
testusr1@cheyenne1:~/cases/b.day1.0>
```

Namelist creation

Model Build

Success

Work Flow: Super Quick Start

Set of commands to build and run the model on a supported machine: "cheyenne"

```
# one time step – create a directory to store your experiment case roots
mkdir ~/cases

# go into scripts subdirectory of cime
cd /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts

# create a new case in the directory "cases" in your home directory
./create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset B1850

# go into the case you just created in the last step
cd ~/cases/b.day1.0

setenv CESM_BLD_TEMPLATE  /glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (tcsh)
export CESM_BLD_TEMPLATE=/glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld (bash)

# invoke case.setup
./case.setup

# build the executable (cheyenne specific commands!)
qcmd -- ./case.build

# submit your run to the batch queue
./case.submit
```

Dedicated Batch Queues on Cheyenne for tutorial use only!

Day	Time	Queue
Monday, 8/6	2:20 – 5:30 p.m.	R1410465
Tuesday, 8/7	2:20 – 5:30 p.m.	R1410495
Wednesday, 8/8	--	--
Thursday, 8/9	2:20 – 5:30 p.m.	R1410508
Friday, 8/10	1:30 – 3:00 p.m.	R1410520

```
testusr1@cheyenne1:~/cases/b.day1.0> ./xmlquery JOB_QUEUE

Results in group case.run
        JOB_QUEUE: R1410465

Results in group case.st_archive
        JOB_QUEUE: R1410465
testusr1@cheyenne1:~/cases/b.day1.0>
```

Running the Model

```
testusr1@cheyenne1:~/cases/b.day1.0> ./xmlquery DOUT_S
        DOUT_S: TRUE
testusr1@cheyenne1:~/cases/b.day1.0> ./xmlquery STOP_N,STOP_OPTION

Results in group run_begin_stop_restart
        STOP_N: 5
        STOP_OPTION: ndays
testusr1@cheyenne1:~/cases/b.day1.0> ./case.submit
Setting resource.RLIMIT_STACK to -1 from (307200000, -1)
Creating component namelists
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/cam//cime_config/buildnml
CAM namelist copy: file1 /glade/u/home/testusr1/cases/b.day1.0/Buildconf/camconf/atm_in file2 /glade/scratch/testusr1/b.day1.0/run/atm_in
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/clm//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/cice//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/pop//cime_config/buildnml

   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/mosart//cime_config/buildnml
   Running /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/cism//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/components/ww3//cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/components/stub_comps/sesp/cime_config/buildnml
   Calling /gpfs/fs1/p/cesm/tutorial/cesm2.0.0_tutorial/cime/src/drivers/mct/cime_config/buildnml
   NOTE: ignoring setting of glc2ice_rmapname=idmap in seq_maps.rc
Finished creating component namelists
Checking that inputdata is available as part of case submission
Setting resource.RLIMIT_STACK to -1 from (-1, -1)
Loading input file list: 'Buildconf/clm.input_data_list'
Loading input file list: 'Buildconf/cpl.input_data_list'
Loading input file list: 'Buildconf/pop.input_data_list'
Loading input file list: 'Buildconf/ww3.input_data_list'
Loading input file list: 'Buildconf/cice.input_data_list'
Loading input file list: 'Buildconf/cism.input_data_list'
Loading input file list: 'Buildconf/mosart.input_data_list'
Loading input file list: 'Buildconf/cam.input_data_list'
Check case OK
submit_jobs case.run
Submit job case.run
Submitting job script qsub -q R1410465 -l walltime=00:30:00 -A UESM0006 -v ARGS_FOR_SCRIPT='--resubmit' .case.run
Submitted job id is 1487195.chadmin1
Submit job case.st_archive
Submitting job script qsub -q R1410465 -l walltime=0:20:00 -A UESM0006  -W depend=afterok:1487195.chadmin1 -v ARGS_FOR_SCRIPT='--resubmit' case.st_archive
Submitted job id is 1487196.chadmin1
Submitted job case.run with id 1487195.chadmin1
Submitted job case.st_archive with id 1487196.chadmin1
testusr1@cheyenne1:~/cases/b.day1.0> qstat -u testusr1

chadmin1:
                                                            Req'd  Req'd   Elap
Job ID          Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
1487195.chadmin testusr1 R1410465 b.day1.0.r    --   16 576    --  00:30 Q   --
1487196.chadmin testusr1 R1410465 b.day1.0.s    --    1   1    --  00:20 H   --
testusr1@cheyenne1:~/cases/b.day1.0>
```

Check archive and Run options

Check if namelists need to be rebuilt

Check input data

Submit case.run

Submit case.st_archive dependent on the successful completion of case.run

Batch job status qstat –u testusr1

Check the CASEROOT CaseStatus file

```
[aliceb@cheyenne2:b.day1.0]>cat CaseStatus
2018-08-04 18:02:43: case.setup starting
 ---------------------------------------------------
2018-08-04 18:02:45: case.setup success
 ---------------------------------------------------
2018-08-04 18:10:25: case.setup starting
 ---------------------------------------------------
2018-08-04 18:10:25: case.setup success
 ---------------------------------------------------
2018-08-04 20:06:14: case.build starting
 ---------------------------------------------------
2018-08-04 20:06:49: build.clean starting
 ---------------------------------------------------
2018-08-04 20:06:49: build.clean success
 ---------------------------------------------------
2018-08-04 20:09:13: case.build starting
 ---------------------------------------------------
CESM version is cesm2.0.0
Processing externals description file : Externals.cfg
Processing externals description file : Externals_CLM.cfg
Processing externals description file : Externals_POP.cfg
Processing externals description file : Externals_CISM.cfg
Checking status of externals: clm, fates, ptclm, mosart, ww3, cime, cice, pop, cvmix, marbl, cism, source_cism, rtm, cam,
    ./cime
        clean sandbox, on cime_cesm2_0_tutorial
    ./components/cam
        clean sandbox, on cam1/release_tags/cam_cesm2_0_rel_10/components/cam
    ./components/cice
        clean sandbox, on cice5_20180530
    ./components/cism
        clean sandbox, on release-cesm2.0.01
    ./components/cism/source_cism
        clean sandbox, on release-cism2.1.01
    ./components/clm
        clean sandbox, on release-clm5.0.01
    ./components/clm/src/fates
        clean sandbox, on fates_s1.8.1_a3.0.0
    ./components/clm/tools/PTCLM
        clean sandbox, on PTCLM2_180214
    ./components/mosart
        clean sandbox, on release-cesm2.0.00
    ./components/pop
        clean sandbox, on pop2/release_tags/pop2_cesm2_0_rel_n02
    ./components/pop/externals/CVMix
        clean sandbox, on v0.90-beta
    ./components/pop/externals/MARBL
        clean sandbox, on marbl0.29.0
    ./components/rtm
        clean sandbox, on release-cesm2.0.00
    ./components/ww3
        clean sandbox, on ww3/release_tags/ww3_cesm2_0_rel_01
2018-08-04 20:12:47: case.build success
 ---------------------------------------------------
2018-08-04 20:32:47: case.submit starting
 ---------------------------------------------------
2018-08-04 20:32:54: case.submit success case.run:1487195.chadmin1, case.st_archive:1487196.chadmin1
 ---------------------------------------------------
[aliceb@cheyenne2:b.day1.0]>
```

In the Lab:

- Check the files in the RUNDIR as the model is running and once it is finished
- Check the files in the DOUT_S_ROOT directory as the case.st_archive script is running and once it is finished

Success

Preview for postprocessing model output

http://www.cesm.ucar.edu/models/cesm2/scientifically-validated-cesm2.html

Live Demo…

Expert feature: create_clone

- The CIMEROOT/scripts/create_clone tool copies an existing case to make a new copy.
- Things that are copied:
  - Most (not all) env_*.xml settings.
  - user_nl_xxx files
  - Macros
  - SourceMods
  - Batch system files
  - README.case
- Not copied:
  - Logs
  - Timing files

- Invocation (from CIMEROOT/scripts directory):
  - ./create_clone --clone ~/cases/b.day1.0 --case ~/cases/b.day1.2

Best practices for copying cases

- **Using "cp –R" does not work!**
- **When using create_clone, make sure that your changes will be minor:**
  - **Same version of the code!**
  - **Same grid**
  - **Same compset**
  - **Namelist/SourceMods changes not too complex.**

- **Document changes in your case directory so that they are easy to track: README.case is a great place.**

- **If your changes are more complex, if you use multiple code versions, or if you have to create a great many cases at once, consider writing your own script to set up your cases.**

Porting

Porting details will be covered in Wednesday's 1:00 p.m. lecture

CIME Documentation Part 2 – http://esmci.github.io/cime/

- **On supported machines - no porting is necessary**
- **On new machines - porting needs to be done**

From the CESM2 webpage:
http://www.cesm.ucar.edu/models/cesm2/linux_cluster/

NCAR's Experience Porting and Running CESM2 on a Medium-sized Linux Cluster

NCAR typically runs CESM on large super-computers with 4096 cores on yellowstone and 2160 cores on cheyenne. However, we also port, run and regularly tested CESM on a more moderately-sized Linux cluster.

NCAR's Climate and Global Dynamics (CGD) division maintains a medium-size Linux cluster called **hobart** to support research and development.

This page details our experiences on **hobart** that might help other institutions port and run CESM2 on their Linux clusters.

*** NOTE *** This is for information purposes only. Please use the DiscussCESM forums to post your questions regarding porting and running on your particular Linux cluster.

Linux Cluster Hardware Specifications

Single login node with the following specifications:

Hostname :	**hobart**
Operating System :	CentOS Linux release 7.2.1511 (Core) x86_64
Kernal :	3.10.0-327.el7.x86_64
Processor(s) :	16 X Intel(R) Xeon(R) CPU W5580 @ 3.20GHz
CPU MHz :	3192.072
Total Memory :	74.05 GB
Total Swap :	1.04 GB

32 compute nodes with the following specifications for each node:

Operating System :	CentOS Linux release 7.2.1511 (Core) x86_64
Kernal :	3.10.0-327.el7.x86_64
Processor(s) :	48 X Intel(R) Xeon(R) CPU ES-2670 v3 @ 2.30GHz
CPU MHz :	23000.000
Total Memory :	98.59 GB
Total Swap :	1.04 GB

Available shared disk space for run and build directories :
5.0 T

More Information/Getting Help

Model User Guides: http://www.cesm.ucar.edu/models/cesm2.0

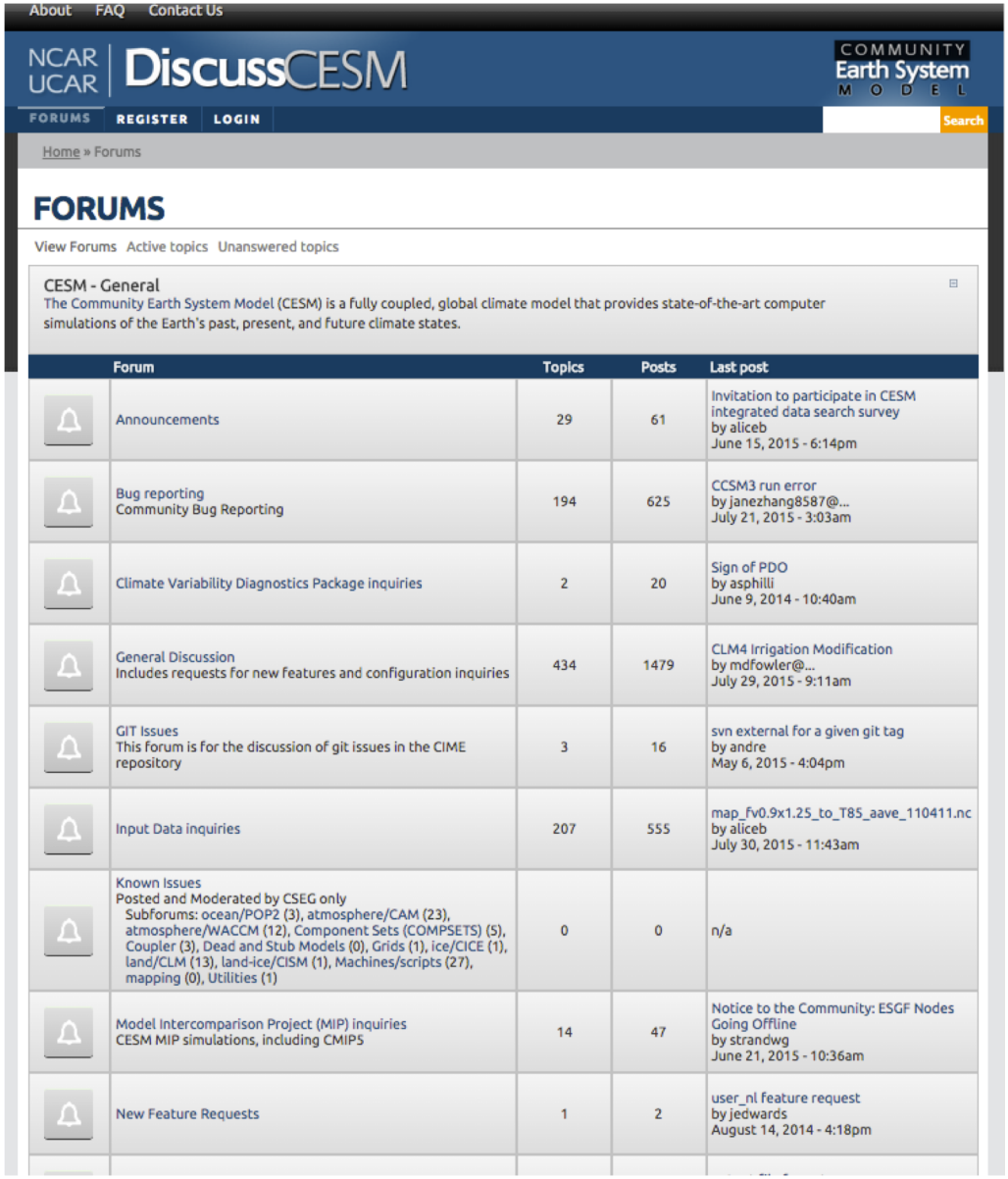
Active or Prognostic Components

Each model component page contains descriptions and documentation for active or prognostic models.

- Atmosphere
- Land
- Land Ice
- Ocean
- Sea Ice
- River Runoff
- Wave

More Information/Getting Help

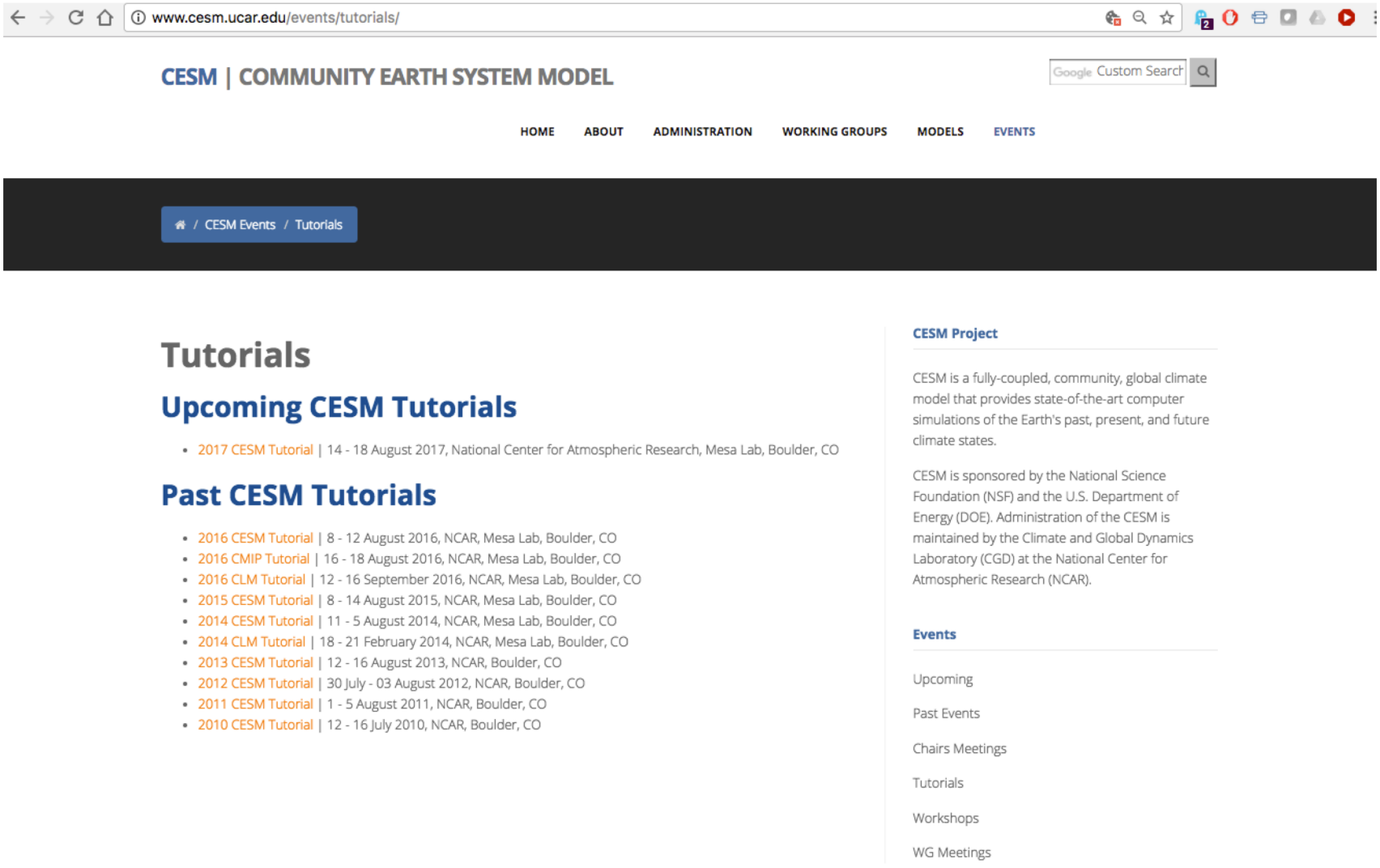
CESM Bulletin Board: http://bb.cgd.ucar.edu/



- **Register** as a forums user by entering your valid information in the registration form
- **Subscribe** to forums of interest - especially the "Announcements" and "Known Problems" – this is one way that we communicate updates to you!
- **Join** the CESM participants email list at: http://mailman.cgd.ucar.edu/mailman/listinfo/ccsm-participants
- **Create** a github account and opt-in to "watch" CESM related repositories

More Information/Getting Help

CESM tutorial: http://www.cesm.ucar.edu/events/tutorials/



Thank You!

The UCAR Mission is:
To advance understanding of weather, climate, atmospheric composition and processes;
To provide facility support to the wider community; and,
To apply the results to benefit society.

NCAR is sponsored by the National Science Foundation

Notes for this tutorial

There are a few things we will do this week that are different from running normally on cheyenne.

- **We will be using code in "/glade/p/cesm/tutorial" this week. Normally, you check out your own version. *The tutorial code refers to a special account key that will not work in the future!***
- **We will be using special queues during the tutorial that will only be available during the times listed on slide 35 of this presentation.**

Some general tips:

- **We will use short case directory names, but in the future you may want to use longer names so that cases are easier to find. Typically, case names should include the compset, grid, and possibly a short name for the experiment.**
- **While CESM is building, you can open a second terminal window and log in to cheyenne again. This allows you to look around or do other things while waiting for a job to complete.**

Day 1 Exercise 0

- This afternoon we will simply be introducing you to the system and running for the first time.
- Log in to cheyenne following the instructions on your compile card and follow these steps.

Step 1: If you are not familiar with the Linux csh environment, then review this cheat sheet with a list of common commands:
http://www.geol.lsu.edu/jlorenzo/ReflectSeismol/labs/unix-cheatsheet.pdf

Step 2: From your tutorial machine window prompt, login to cheyenne:
ssh –Y [username]@cheyenne.ucar.edu

One Time Setup: Check your default login environment settings:

NOTE: all tutorial logins default to tcsh and environment settings are in ~/.tcshrc
If you are using an existing login and bash, then you can copy the commands in /gpfs/fs1/p/cesm/tutorial/bashrc to your .bashrc, logout and login again

env

Day 1 Exercise 1

- This afternoon we will simply be introducing you to the system and running for the first time. After each step, check the files in the CASEROOT, EXEROOT, RUNDIR and DOUT_S_ROOT directories.
- Log in to cheyenne and run the following steps.

```
# One time step
mkdir ~/cases

# go into scripts directory of the tutorial source code download
cd /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts

# (1) create a new case in the directory "cases" in your home directory (don't forget the "./")
./create_newcase --case ~/cases/b.day1.0 --res f19_g17 --compset B1850

# go into the case you just created in the last step
cd ~/cases/b.day1.0
./xmlquery CASEROOT

# (2)  set the CESM_BLD_TEMPLATE environment variable for faster build times on cheyenne
setenv CESM_BLD_TEMPLATE  /glade/p/cesm/tutorial/templates/cesm2.0.0_b1850/bld

# (3) invoke case.setup
./case.setup
./xmlquery EXEROOT
./xmlquery RUNDIR

# (4) check the queue settings for the day to make sure they are correct and change if necessary!
./xmlquery JOB_QUEUE
./xmlchange JOB_QUEUE=[queue-name] --subgroup case.run
./xmlchange JOB_QUEUE=[queue-name] --subgroup case.st_archive

# (5) build the executable on a cheyenne compute node
qcmd -- ./case.build
```

Day 1 Exercise 1 – continued

This afternoon we will simply be introducing you to the system and running for the first time. After each step, check the files in the CASEROOT, EXEROOT, RUNDIR and DOUT_S_ROOT directories.

 # (4) submit your run to the batch queue

NOTE – each day of the tutorial we will be using a different dedicated batch queue for submission.
Prior to each case.submit command, you will want to run the following xmlchange commands:

Monday, 8/6:
./xmlchange –-subgroup case.run JOB_QUEUE=R1410465
./xmlchange –-subgroup case.st_archive JOB_QUEUE=R1410465

Tuesday, 8/7:
./xmlchange –-subgroup case.run JOB_QUEUE=R1410495
./xmlchange –-subgroup case.st_archive JOB_QUEUE=R1410495

Wednesday:
No dedicated queue required.

Thursday, 8/9:
./xmlchange –-subgroup case.run JOB_QUEUE=R1410508
./xmlchange –-subgroup case.st_archive JOB_QUEUE=R1410508

Friday, 8/10:
./xmlchange –-subgroup case.run JOB_QUEUE=R1410520
./xmlchange –-subgroup case.st_archive JOB_QUEUE= R1410520

Now, submit
./case.submit
qstat –u [loginname]
./xmlquery DOUT_S_ROOT

Day 1 Exercises 2-3

```
# Exercise 1: Check on your case and resubmit when it is complete.
qstat -u [loginname]
cat CaseStatus

# Changing options like STOP_N and STOP_OPTION would increase run length.
./xmlchange CONTINUE_RUN=TRUE
qcmd -- ./case.submit

# Note that if you make a mistake, you can kill the job using its ID number displayed when you run qstat
# qdel <job_id>

# Exercise 2: create_clone

# Go back to the CIMEROOT scripts directory
cd /glade/p/cesm/tutorial/cesm2.0.0_tutorial/cime/scripts

# Make a clone of the case
./create_clone --clone ~/cases/b.day1.0 --case ~/cases/b.day1.2

# Take a look in the create_clone directory.

# What is the value of CONTINUE_RUN in the new directory (this is in env_run.xml)?
# What does README.case look like?
# What other files are copied over?
# What would be the next step in building and running the cloned case?
```

Questions to answer on your own:

Yes, these could be on a quiz!

1. What is the value of XML variable CASEROOT ?
2. What do CASEROOT scripts check_case, pelayout, preview_run, and preview_namelists do?
3. When do you need to run ./case.setup --reset ?
4. What files are in the CASEROOT/LockedFiles before case.submit ?
5. Why is there a CASEROOT/LockedFIles directory ?
6. When do you need to run ./case.build --clean ? What about ./case.build --clean-all ?
7. How do you change the JOB_QUEUE XML setting using xmlchange for both the case.run and case.st_archive scripts ?
8. What are some of the XML variables that you need to specify a subgroup argument to ./xmlquery or ./xmlchange ?
9. When can you make XML changes in the workflow ?

Further exercises

Some suggestions if you finish early today:

- **Look through the exercises from Christine Shields to get a preview of this Tuesday's topics.**

- **Look through the CESM2.0 web page and other information online. Try to get a feel for what information you would need to set up your own cases.**

  http://www.cesm.ucar.edu and http://www.cesm.ucar.edu/models/cesm2.0 and https://www2.cisl.ucar.edu/resources/computational-systems/cheyenne

- **Try using the "ncview" command on one of the history files in your run directory. This is a simple but useful tool for taking a quick look at output. First, look at the system modules loaded in your login environment:**

  module list

  If ncview is not listed, then load it into your environment using:

  module load ncview

- **Take a quick look at the NCO utilities for manipulating netCDF files: http://nco.sourceforge.net/nco.html**

- **PLEASE register as a new user on the DiscussCESM Forums website at:**

  http://bb.cgd.ucar.edu

  Include a valid email, name, job title, and organization so I can approve your request and keep the spammers out! At a minimum, subscribe to the "Announcements" and "Known Problems" forums.

- **Sign-up for E-mail Notifications: CCSM Participants Mailman Registration**

Day 1 Auxiliary Exercises

In Wednesday's lab session you will be learning how to run the various diagnostic packages. You will also learn about the types of tools that are commonly used on model output. Here are some exercises that you can do to prepare yourself for Wednesday's lab session.

- Go to the CESM1 Large Ensemble Community Project pagehttp://www.cesm.ucar.edu/projects/community-projects/LENS/ After reading the project overview click on the "Diagnostics" link. Take a look at the available experiments and look at diagnostics output from the atmosphere, sea ice, land, and ocean diagnostics packages. Become familiar with the types of calculations the packages do.

- Go to each of the prognostic model web pages

- See http://www.cesm.ucar.edu/working_groups/CVC/cvdp. The **Climate Variability Diagnostics Package (CVDP)** is different from the other diagnostics packages in that it is usually run over an entire simulation and can be run on numerous simulations (CESM *and non-CESM data*) at once. The CVDP calculates the major modes of variability, trends, and provides a quantifiable metric table. Look at the website example comparisons.

- Go to http://climatedataguide.ucar.edu and explore the site. The **Climate Data Guide** contains information on over 150 different datasets, provides inter-dataset comparisons, and has dataset pros and cons evaluated by expert dataset users.

- The programming language **NCL** is used extensively within the CESM project. You will have the opportunity to run several NCL scripts on Wednesday. Take a look at the NCL Examples page to get an idea of the types of plots NCL can create: http://www.ncl.ucar.edu/Applications/