

---

# Porting CESM to Shaheen

*Release 0.0.1*

**Ben Johnson**

**Dec 27, 2022**



<b>1</b>	<b>Environment</b>	<b>3</b>
1.1	User Guide . . . . .	3
1.2	Training Materials . . . . .	3
1.3	Programming Environments . . . . .	3
<b>2</b>	<b>SLURM</b>	<b>5</b>
2.1	Aliases . . . . .	5
2.2	Job Queue . . . . .	6
2.3	Resource Binding . . . . .	6
2.4	Node Configuration . . . . .	6
2.5	Default Settings . . . . .	7
2.6	Analog for Cheyenne . . . . .	7
<b>3</b>	<b>Filesystem</b>	<b>9</b>
3.1	A helpful reminder . . . . .	9
<b>4</b>	<b>Recompiling CESM</b>	<b>11</b>
4.1	First failed attempt: recursively copy cesm2_1_3 to scratch . . . . .	11
4.2	Downloading a clean copy of CESM . . . . .	11
<b>5</b>	<b>PIO</b>	<b>15</b>
<b>6</b>	<b>Compiling</b>	<b>17</b>
6.1	Programming Environment . . . . .	17
<b>7</b>	<b>Configuration Notes</b>	<b>19</b>
7.1	Maximum Number of Members? . . . . .	19
<b>8</b>	<b>Generating the Initial Ensemble</b>	<b>21</b>
8.1	Perturb Single Instance Program . . . . .	21
8.2	Files Needed . . . . .	21
8.3	Steps to Generate the Necessary Files . . . . .	21
8.4	Running perturb_single_instance . . . . .	23
<b>9</b>	<b>Sampling Error Table</b>	<b>25</b>
9.1	gen_sampling_err_table . . . . .	25
9.2	DART_config.template . . . . .	25
9.3	References . . . . .	26
<b>10</b>	<b>Recovering from a Failed Assimilation</b>	<b>27</b>

<b>11</b>	<b>NCO</b>	<b>29</b>
<b>12</b>	<b>Experimental configuration</b>	<b>31</b>
12.1	Filters . . . . .	31
12.2	Inflation . . . . .	31
12.3	Saved output . . . . .	31
12.4	References . . . . .	32
<b>13</b>	<b>Steps to begin an experiment</b>	<b>33</b>
<b>14</b>	<b>Testing get_close caching fix</b>	<b>35</b>
14.1	mkmf.template . . . . .	35
14.2	Possible configurations . . . . .	35
14.3	Differences . . . . .	36
<b>15</b>	<b>obs_diag</b>	<b>63</b>
15.1	Running obs_diag on CAM output . . . . .	63
15.2	Using the MATLAB Diagnostic Scripts . . . . .	64
<b>16</b>	<b>Grid and compset</b>	<b>65</b>
16.1	Scientifically supported configuration . . . . .	65
16.2	Custom Grid . . . . .	66
16.3	Compset . . . . .	67
<b>17</b>	<b>Libraries</b>	<b>69</b>
17.1	Parallel NetCDF String . . . . .	69
17.2	Instructions Regarding Macros.make . . . . .	69
17.3	Attempting to Build a Case Using Macros.make . . . . .	70
<b>18</b>	<b>List of Attempts</b>	<b>73</b>
18.1	Trial 102 . . . . .	73
18.2	Trial 101 . . . . .	73
18.3	Trial 100 . . . . .	74
18.4	Trial 099 . . . . .	75
18.5	Trial 098 . . . . .	75
18.6	Trial 095 . . . . .	75
18.7	Trial 094 . . . . .	76
18.8	Trial 093 . . . . .	76
18.9	Trial 092 . . . . .	76
18.10	Trial 091 . . . . .	76
18.11	Trial 090 . . . . .	77
18.12	Trial 089 . . . . .	77
18.13	Trial 088 . . . . .	78
18.14	Trial 087 . . . . .	78
18.15	Trial 086 . . . . .	78
18.16	Trial 085 . . . . .	79
18.17	Trial 084 . . . . .	79
18.18	Trial 083 . . . . .	79
18.19	Trial 082 . . . . .	79
18.20	Trial 081 . . . . .	80
18.21	Trial 079 . . . . .	80
18.22	Trial 078 . . . . .	81
18.23	Trial 077 . . . . .	81
18.24	Trial 076 . . . . .	82
18.25	Trial 075 . . . . .	82

18.26 Trial 074 . . . . .	83
18.27 Trial 073 . . . . .	83
18.28 Trial 072 . . . . .	83
18.29 Trial 071 . . . . .	83
18.30 Trial 070 . . . . .	84
18.31 Trial 069 . . . . .	84
18.32 Trial 068 . . . . .	84
18.33 Trial 067 . . . . .	85
18.34 Trial 066 . . . . .	85
18.35 Trial 065 . . . . .	86
18.36 Trial 064 . . . . .	86
18.37 Trial 063 . . . . .	86
18.38 Trial 062 . . . . .	87
18.39 Trial 061 . . . . .	87
18.40 Trial 060 . . . . .	88
18.41 Trial 059 . . . . .	89
18.42 Trial 058 . . . . .	89
18.43 Trial 057 . . . . .	89
18.44 Trial 056 . . . . .	89
18.45 Trial 055 . . . . .	90
18.46 Trial 054 . . . . .	90
18.47 Trial 049 . . . . .	90
18.48 Trial 048 . . . . .	91
18.49 Trial 047 . . . . .	91
18.50 Trial 046 . . . . .	94
18.51 Trial 045 . . . . .	94
18.52 Trial 044 . . . . .	94
18.53 Trial 043 . . . . .	94
18.54 Trial 042 . . . . .	95
18.55 Trial 041 . . . . .	95
18.56 Trial 040 . . . . .	95
18.57 Trial 039 . . . . .	96
18.58 Trial 038 . . . . .	96
18.59 Trial 037 . . . . .	96
18.60 Trial 036 . . . . .	97
18.61 Trial 035 . . . . .	98
18.62 Trial 034 . . . . .	99
18.63 Trial 033 . . . . .	99
18.64 Trial 032 . . . . .	100
18.65 Trial 031 . . . . .	100
18.66 Trial 030 . . . . .	101
18.67 Trial 029 . . . . .	101
 <b>19 cime_comp_mod.F90</b>	 <b>103</b>
 <b>20 config_machines.xml</b>	 <b>105</b>
20.1 Tasks per node . . . . .	105
20.2 One question . . . . .	105
20.3 mpirun Settings . . . . .	106
 <b>21 cpus-per-task</b>	 <b>107</b>
21.1 Overview . . . . .	107
21.2 Parallel netCDF library . . . . .	107
21.3 Attempt to spoof a Cori Haswell Run . . . . .	108

21.4	Trial 1: srun isn't given the <code>-cpus-per-task 2</code> option . . . . .	109
21.5	Trial 2: srun is given the <code>-cpus-per-task 2</code> option . . . . .	109
<b>22</b>	<b>Scientifically supported configuration</b>	<b>111</b>
22.1	Overview . . . . .	111
22.2	Scientifically supported grid . . . . .	111
22.3	Custom Grid . . . . .	113
22.4	Compset . . . . .	113
<b>23</b>	<b>Run type</b>	<b>115</b>
23.1	Initial files . . . . .	115
23.2	Using <code>perturb_single_instance</code> . . . . .	115
<b>24</b>	<b>Troubleshooting CIME</b>	<b>117</b>
24.1	<code>input_data.checksum.dat</code> . . . . .	117
<b>25</b>	<b>Missing XML Library</b>	<b>119</b>
25.1	Installing perl locally . . . . .	120
<b>26</b>	<b>Load Balancing Tool</b>	<b>121</b>
26.1	Overview . . . . .	121
26.2	Instructions . . . . .	121
<b>27</b>	<b>Case Timing Data</b>	<b>123</b>
27.1	Overview . . . . .	123
27.2	Major Issue: Init Time . . . . .	123
27.3	Previously Encountered Problem . . . . .	124
<b>28</b>	<b>File Striping</b>	<b>125</b>
28.1	Optimizing I/O Performance . . . . .	125
28.2	Advice from KAUST Supercomputing Lab . . . . .	125
28.3	Best LFS Stripe Count for Different Node Counts . . . . .	126
28.4	Striping Commands . . . . .	126
<b>29</b>	<b>Darshan</b>	<b>129</b>
29.1	Installation . . . . .	129
29.2	Finding Job IDs . . . . .	129
29.3	Run Darshan for a Specific Job . . . . .	130
<b>30</b>	<b>Intel VTune Profiler</b>	<b>131</b>
<b>31</b>	<b>Arm Forge</b>	<b>133</b>
31.1	Overview . . . . .	133
31.2	Recompiling . . . . .	133
<b>32</b>	<b>Ensemble Consistency Tests</b>	<b>135</b>
32.1	POP Ensemble Consistency Test . . . . .	135
32.2	CAM Ensemble Consistency Test . . . . .	135
32.3	UF-CAM Ensemble Consistency Test . . . . .	136
<b>33</b>	<b>Regression Tests</b>	<b>137</b>
33.1	Building <code>cprnc</code> . . . . .	137
33.2	Edit <code>config_machines.xml</code> . . . . .	137
33.3	Run the tests . . . . .	137

<b>34</b>	<b>Cori</b>	<b>139</b>
34.1	Login instructions . . . . .	139
34.2	File system . . . . .	139
34.3	CESM . . . . .	140
<b>35</b>	<b>Getting files from the Research Data Archive</b>	<b>143</b>
35.1	Location of files on the GLADE filesystem . . . . .	143
35.2	Untarring all of the instances . . . . .	143
<b>36</b>	<b>Sphinx as a Documentation Tool</b>	<b>145</b>
36.1	Installing Conda . . . . .	145
36.2	Downloading Sphinx . . . . .	145
36.3	Clone the Repository . . . . .	146
36.4	Edit Files and Remake the Documentation . . . . .	146





This documents our attempt to port the [Community Earth System Model \(CESM\)](#) to [Shaheen II](#) which is a Cray XC40 at [King Abdullah University of Science and Technology](#).

The purpose of porting CESM is to test data assimilation algorithms using 1000-member ensembles of the [Community Atmosphere Model \(CAM\)](#).



## ENVIRONMENT

### 1.1 User Guide

Shaheen II User Guide

### 1.2 Training Materials

KAUST provides instructional materials about the programming environments on Shaheen.

- [Programming/Runtime Environment](#)
- [List of User Training Materials](#)

### 1.3 Programming Environments

Shaheen provides three different programming environments

There are three different compiler environments, each with their own compilers:

- PrgEnv-cray (crayftn, craycc, crayCC)
- PrgEnv-intel (ifort, icc, icpc)
- PrgEnv-gnu (gfortran, gcc, g++)

According to the [documentation](#):

*All applications that will run in parallel on the Cray XC should be compiled with standard language wrappers: cc for C, CC for C++, and ftn for Fortran.*

---

**Note:** Using the ftn wrapper means that the appropriate Fortran compiler will be used, regardless of the programming environment.

---

No additional MPI flags are needed as they are included by wrappers.

Again, according to the [documentation](#):

*There is no universally fastest compiler – but performance depends on the application, even input. However Cray is trying to excel with the Cray Compiler Environment. Compiler flags do matter – be ready to spend some effort finding the best ones for your application.*



## SLURM

Shaheen uses the Simple Linux Utility for Resource Management (SLURM) system for job scheduling.

### 2.1 Aliases

Since the status and kill commands for SLURM differ from PBS, it is useful to set up common aliases, for example in `.bashrc`:

```
# Terminate a job when given its job id
alias jkill='scancel'
# Print user's jobs and status
alias jstat='squeue -u $USER'
# Start an interactive session on a compute node
alias jinter='srun -u --pty bash -i'
# Print out jobs that have run since a start date
alias jhist='sacct -u $USER --format="JobID%20,JobName%30,Partition,Account,AllocCPUS,
↳State,ExitCode" -S'
```

The last alias, `jhist`, prints out jobs that have been completed since a given start date:

```
$ jhist 2020-11-13
      JobID              JobName  Partition   Account  AllocCPUS  ↳
↳ State ExitCode
-----
↳ -----
      16435549 run.FHIST_BGC.f09_d025.084.e03      workq      k1421      768  ↳
↳ COMPLETED      0:0
      16435549.batch              batch              k1421      64  ↳
↳ COMPLETED      0:0
      16435549.0              cesm.exe              k1421      384  ↳
↳ COMPLETED      0:0
```

## 2.2 Job Queue

The standard job queue on Shaheen is workq.

## 2.3 Resource Binding

SLURM is [highly configurable](#) with respect to its ability to bind tasks to various resources such as “threads, cores, sockets, NUMA or boards.”

## 2.4 Node Configuration

To see the configuration of Shaheen’s Cray XC40 nodes, invoking `scontrol show nodes` will print out the node configuration:

```
$ scontrol show nodes
NodeName=nid07679 Arch=x86_64 CoresPerSocket=16
CPUAlloc=64 CPUTot=64 CPULoad=64.00
AvailableFeatures=(null)
ActiveFeatures=(null)
Gres=craynetwork:4
NodeAddr=nid07679 NodeHostName=nid07679 Version=20.02.6
OS=Linux 4.12.14-150.17_5.0.91-cray_ari_c #1 SMP Wed May 27 02:24:01 UTC 2020 (6b16d42)
RealMemory=128803 AllocMem=128448 FreeMem=123036 Sockets=2 Boards=1
State=ALLOCATED ThreadsPerCore=2 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
Partitions=workq,72hours
BootTime=2020-12-03T10:05:38 SlurmdStartTime=2020-12-03T10:22:30
CfgTRES=cpu=64,mem=128803M,billing=64
AllocTRES=cpu=64,mem=128448M
CapWatts=n/a
CurrentWatts=47 AveWatts=0
ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
```

When counting CPUs per node, there are:

- Sockets=2
- CoresPerSocket=16
- ThreadsPerCore=2

Thus `cpu=64`. Should we set: `--ntasks-per-node=32` or `--ntasks-per-node=64`?

## 2.5 Default Settings

To see the default settings for SLURM, invoking the `scontrol show config` command will print all of the settings:

```
$ scontrol show config
...
DefMemPerCPU          = 2007
...
MaxTasksPerNode      = 512
...
```

We see, for example, that `DefMemPerCPU = 2007` (which is reported in megabytes, M) and there are 64 cpu/node, thus the memory per node is  $2007M * 64 = 128448M$  which is consistent with the printout of `scontrol show nodes` above.

## 2.6 Analogs for Cheyenne

While Cheyenne uses PBS rather than SLURM it's useful to have their settings as well. The analogous command for `scontrol show nodes` is `pbsnodes -a`.

```
$ pbsnodes -a
r8i5n1
Mom = r8i5n1.ib0.cheyenne.ucar.edu
ntype = PBS
state = free
pcpus = 72
resources_available.arch = linux
resources_available.host = r8i5n1
resources_available.iru = r8i5
resources_available.iru2 = r8i4i5
resources_available.mem = 131567260kb
resources_available.ncpus = 72
resources_available.nodetype = largemem
resources_available.Qlist = system,special,ampsrt,capability,premium,regular,standby,
↪economy,small
resources_available.rack = r8
resources_available.rack16 = r1r2r3r4r5r6r7r8r9r10r11r12r13r14r15r16
resources_available.rack2 = r15r16
resources_available.rack4 = r13r14r15r16
resources_available.rack8 = r9r10r11r12r13r14r15r16
resources_available.switch = r8i5a0s0
resources_available.switchblade = r8i5s0
resources_available.vnode = r8i5n1
resources_assigned.accelerator_memory = 0kb
resources_assigned.hbmem = 0kb
resources_assigned.mem = 0kb
resources_assigned.naccelerators = 0
resources_assigned.ncpus = 0
resources_assigned.vmem = 0kb
comment =
resv_enable = True
sharing = default_shared
```

(continues on next page)

(continued from previous page)

```
license = 1
last_state_change_time = Mon Dec 21 17:14:09 2020
last_used_time = Mon Dec 21 17:14:09 2020
```

We haven't been able to find the analagous command for `scontrol show config`.



## FILESYSTEM

### 3.1 A helpful reminder

---

**Important:** Shaheen, as of 9 January 2022, no longer allows scripts to write to project space.

---

This is the notification from Maciej Olchowik:

Following successful acceptance of the new project filesystem last month, we are now in a position to deploy more storage in production.

In the next two weeks we will transition most of the projects to this new filesystem.

The only impact on the users is that the new `/project` filesystem is now mounted read-only on the compute nodes. This means that any jobs that are configured to write to `/project` directly will fail.

We have always encouraged all our users to run their jobs from the `/scratch` filesystem, this will now be necessary with the above change. Please see this link for the differences between project and scratch:  
<https://www.hpc.kaust.edu.sa/tips/scratch-vs-project>

Thus to compile DART executables it is necessary to move the DART directory to scratch, run the quickbuild scripts and then move the directory back to project space.



## RECOMPILING CESM

---

**Important:** The 25 January 2022 maintenance session on Shaheen caused significant problems for creating new CESM cases. This page documents the steps needed to get a working version of CESM that includes the porting modifications for Shaheen.

---

### 4.1 First failed attempt: recursively copy cesm2\_1\_3 to scratch

```
cd /lustre/project/k1421
cp -R cesm2_1_3 /lustre/scratch/x_johnsobk
cd /lustre/scratch/x_johnsobk/cesm2_1_3/cime/scripts
./create_newcase --case /lustre/scratch/x_johnsobk/cases/FHIST.cesm2_1_3.f09_f09_mg17.
↪e001.n0003 --compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
cd /lustre/scratch/x_johnsobk/cases/FHIST.cesm2_1_3.f09_f09_mg17.e001.n0003
./case.setup
./case.build
```

This attempt fails, with issues similar to what is discussed in this [ROMS thread](#).

### 4.2 Downloading a clean copy of CESM

```
cd /lustre/scratch/x_johnsobk
git clone https://github.com/ESCOMP/CESM.git CESM
cd /lustre/scratch/x_johnsobk/CESM
git checkout release-cesm2.1.3
/lustre/scratch/x_johnsobk/CESM/cime/config/cesm/machines

mv config_batch.xml config_batch_old.xml
mv config_compilers.xml config_compilers_old.xml
mv config_machines.xml config_machines_old.xml

cp /lustre/scratch/x_johnsobk/cesm2_1_3/cime/config/cesm/machines/config_machines.xml ./
cp /lustre/scratch/x_johnsobk/cesm2_1_3/cime/config/cesm/machines/config_compilers.xml ./
cp /lustre/scratch/x_johnsobk/cesm2_1_3/cime/config/cesm/machines/config_batch.xml ./
```

### 4.2.1 First attempt

```
cd /lustre/scratch/x_johnsobk/CESM/cime/scripts
./create_newcase --case /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e001.n0003 --compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --ninst 3 --multi-driver --walltime 2:00:00

cd /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e001.n0003
./case.setup
```

```
Error: /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e001.n0003
./case.setup ERROR: Command: '/usr/bin/xmllint -noout -schema /lustre/scratch/x_johnsobk/CESM/cime/config/xml_schemas/config_compilers_v2.xsd /lustre/scratch/x_johnsobk/CESM/cime/config/cesm/machines/config_compilers.xml' failed with error '/lustre/scratch/x_johnsobk/CESM/cime/config/cesm/machines/config_compilers.xml:78: element INCLDIR: Schemas validity error : Element 'INCLDIR': This element is not expected.'
```

### 4.2.2 Second attempt

Change the config\_compilers xml schema with an older copy.

```
cd /lustre/scratch/x_johnsobk/CESM/cime/config/xml_schemas/
mv config_compilers_v2.xsd config_compilers_v2_old.xsd
cp /lustre/project/k1421/cesm2_1_3/cime/config/xml_schemas/config_compilers_v2.xsd ./
```

Rebuild the case.

```
cd /lustre/scratch/x_johnsobk/CESM/cime/scripts
./create_newcase --case /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e001.n0003 --compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --ninst 3 --multi-driver --walltime 2:00:00

cd /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e001.n0003
./case.setup
./case.build
```

```
Error: cat /lustre/scratch/x_johnsobk/FHIST.CESM.f09_f09_mg17.e001.n0003/bld/cesm.bldlog.220204-072500
```

### 4.2.3 Third attempt

Trying to build with pio2.

```
cd /lustre/scratch/x_johnsobk/CESM/cime/scripts
./create_newcase --case /lustre/scratch/x_johnsobk/cases/FHIST.CESM.PIO2.f09_f09_mg17.e002.n0003 --compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --ninst 3 --multi-driver --walltime 2:00:00
./case.setup
```

(continues on next page)

(continued from previous page)

```
./xmlchange PIO_VERSION=2
./case.build
/usr/bin/ld: /lustre/scratch/x_johnsobk/CESM/cime/src/externals/pio2/src/clib/pio_darray_
↪int.c:1212: undefined reference to `nc_get_vara_float'
```

**Error:** ERROR: BUILD FAIL: buildexe failed, cat /lustre/scratch/x\_johnsobk/FHIST.CESM.f09\_f09\_mg17.e002.n0003/bld/ces041905

#### 4.2.4 Fourth attempt

Trying to change the parallel netcdf library that CESM compiles with, from parallel\_netcdf to netcdf-hdf5parallel.

Editing config\_machines.xml:

```
<!-- <command name="load">cray-parallel-netcdf/1.12.1.4</command> -->
<!-- BKJ 2022-02-07 PIO issues attempt -->
<command name="load">cray-netcdf-hdf5parallel/4.7.4.4</command>
[...]
<!-- BKJ 2022-02-07 PIO issues attempt -->
<!-- <env name="PNETCDF_PATH">/opt/cray/pe/parallel-netcdf/1.12.1.4/INTEL/19.1</env> -->
<env name="PNETCDF_PATH">/opt/cray/pe/netcdf-hdf5parallel/4.7.4.4/INTEL/19.1</env>
```

Attempting to build with these libraries:

```
cd /lustre/scratch/x_johnsobk/CESM/cime/config/cesm/machines
xmlint --noout --schema /lustre/scratch/x_johnsobk/CESM/cime/config/xml_schemas/config_
↪machines.xsd ./config_machines.xml
./config_machines.xml validates
source activate py27
cd /lustre/scratch/x_johnsobk/CESM/cime/scripts
./create_newcase --case /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e002.
↪n0003 --compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
cd /lustre/scratch/x_johnsobk/cases/FHIST.CESM.f09_f09_mg17.e002.n0003
./case.setup
./case.build
```

Many different attempts all ended with variations of this error:

**Error:** CMake Error at /sw/x40cle7up03/cmake/3.22.1/sles15gcc7.5.0/share/cmake-3.22/Modules/FindPackageHandleStandardArgs.cmake:230 (message): Could NOT find PnetCDF\_Fortran (missing: PnetCDF\_Fortran\_LIBRARY PnetCDF\_Fortran\_INCLUDE\_DIR)

## 4.2.5 Eight attempt

Attempt to return to the original parallel-netcdf library.

Editing config\_machines.xml:

```
<command name="load">cray-parallel-netcdf/1.12.1.4</command>
[... ]
<env name="PNETCDF_PATH">/opt/cray/pe/parallel-netcdf/1.12.1.4/INTEL/19.1</env>
```

Edit the Makefile to add another linker flag.

```
vim /lustre/scratch/x_johnsobk/CESM/cime/scripts/Tools/Makefile
```

```
SLIBS += -L$(LIB_PNETCDF) -lpnetcdf_intel -lpnetcdf
```

```
Error: ERROR: BUILD FAIL: buildexe failed, cat /lustre/scratch/x_johnsobk/FHIST.CESM.f09_f09_mg17.e006.n0003/bld/cesm.bld
080053
```

---

CHAPTER  
**FIVE**

---

**PIO**





## COMPILING

## 6.1 Programming Environment

Of the environments available on Shaheen, we are using the Intel programming environment, `PrgEnv-intel`, because the new DART algorithms use `mkl`. As an alternative, the Cray programming environment, `PrgEnv-cray`, contains the `cray-libsci` which provides the same functionality as `mkl`. Brian Dobbins and his former coworkers in ASAP have working software on NERSC's Cori system (which is a Cray XC40) using `PrgEnv-cray` and `cray-libsci`. However, they found that CESM ran ~10-15% faster there using the Intel compilers.

This DART build, `/lustre/project/k1421/DART_beta/`, does not have any `mkl` dependencies so we can configure the build templates.

Two of these templates to actually work:

```
/lustre/project/k1421/DART_beta/mkmf.template.PrgEnv-intel.WORKING
/lustre/project/k1421/DART_beta/mkmf.template.PrgEnv-cray.WORKING
```

They are both built using `mkmf.template.pgi.cray` as a starting point, since it contains the Cray compiler wrappers that we should be using on Shaheen, but they have different compiler flags and paths to `netCDF`.

### 6.1.1 Cray Programming Environment

To build with the Cray compilers using `PrgEnv-cray`:

```
$ module load cray-netcdf
$ nc-config --all
...
--includedir /opt/cray/pe/netcdf/4.6.3.2/include
--libdir      /opt/cray/pe/netcdf/4.6.3.2/CRAYCLANG/9.0/lib
```

Thus, in `mkmf.template` this configuration works:

```
INCS = -I/opt/cray/pe/netcdf/4.6.3.2/include
LIBS = -L/opt/cray/pe/netcdf/4.6.3.2/CRAYCLANG/9.0/lib -lnetcdff -lnetcdf
FFLAGS = -O $(INCS)
LDFLAGS = $(INCS) $(LIBS)
```

The compiler does print out a lot of disconcerting warnings.

## 6.1.2 Intel Programming Environment

To build with Intel compilers using `PrgEnv-intel` and `mkl`:

```
$ module swap PrgEnv-cray PrgEnv-intel
$ module load cray-netcdf
$ module unload cray-libsci
$ nc-config --all
...
--includedir /opt/cray/pe/netcdf/4.6.3.2/include
--libdir      /opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0/lib
...
```

### Best Working Configuration

The following is the best configuration we found for `mkmf.template`. We had an earlier working version, described be low in [Original Working Configuration](#) that had a different `AVX` flag that worked but the compiler threw a strange warning.

In `mkmf.template` this configuration works without throwing errors:

```
MPIFC = ftn
MPILD = ftn
FC = ftn
LD = ftn
INCS = -I/opt/cray/pe/netcdf/4.6.3.2/include
LIBS = -L/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib -lnetcdff -lnetcdf
FFLAGS = -O2 -xCORE-AVX2 -assume buffered_io -mkl $(INCS)
LDFLAGS = $(FFLAGS) $(LIBS)
```

### Original Working Configuration

In `mkmf.template` this configuration also works:

```
MPIFC = ftn
MPILD = ftn
FC = ftn
LD = ftn
INCS = -I/opt/cray/pe/netcdf/4.6.3.2/include
LIBS = -L/opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0/lib -lnetcdff -lnetcdf
FFLAGS = -O -assume buffered_io -xAVX -mkl $(INCS)
LDFLAGS = $(FFLAGS) $(LIBS)
```

But when running `quickbuild.csh` for `lorenz_63` and `lorenz_96`, the compiler prints out a perplexing warning:

:: code-block:

```
ifort: command line warning #10121: overriding '-xCORE-AVX2' with '-xAVX'
```

This doesn't make sense because we're not using the `-xCORE-AVX2` flag in the first place. "AVX" stands for Advanced Vector Extension, which is an extension to the x86 instruction set architecture that makes use of the additional registers in modern CPUs to handle calculations more efficiently.

## CONFIGURATION NOTES

### 7.1 Maximum Number of Members?

Tim Hoar suggested that the maximum number of ensemble members might be limited by a character string of length three, thus limiting the ensemble to 999.

Jeff Anderson said the quickest way to test this is to run Lorenz '63 with 1000 members to see if anything crashes. The likeliest place to look for this would be in the ensemble manager and it might be a two line modification to test the string output.

I was able to run Lorenz '63 with 1001 members, so this seems to be a non-issue.



## GENERATING THE INITIAL ENSEMBLE

In order to generate the initial 1000-member ensemble we will make infinitesimal perturbations to a single initial file from Kevin's CAM reanalysis and then run the integration with filter and adaptive inflation to spread out the ensemble. Only the CAM initial files will be perturbed – the hope is that filter and adaptive inflation will adjust the remainder of the components.

### 8.1 Perturb Single Instance Program

The `perturb_single_instance` program in `/lustre/project/k1421/DART/models/cam-fv/work` will take a single initial file and then will perturb a field within it to generate an initial ensemble.

### 8.2 Files Needed

The program needs the following files in order to run:

1. `input.nml` - This must contain the `&perturb_single_instance_nml` namelist with proper settings for the program to read.
2. `campinput.nc` - A CAM **initial** file – not a restart file – since initial files contain the lev and T fields needed for perturbation.
3. `cam_phis.nc` - A CAM file containing the PHIS, or surface geopotential field ( $\text{m}^2/\text{s}^2$ ), since surface geopotential is needed to compute potential temperature.
4. (*Optional*) `output_file_list.txt` - A text file containing the names of the perturbed files to be created. These names can be provided in the `output_files` field of `input.nml`. However, since we want 1000 perturbed files it is easier to specify them in a separate text file and setting the `output_file_list` field of `input.nml`.

### 8.3 Steps to Generate the Necessary Files

We create an empty directory in which to build our ensemble.

```
$ cd /lustre/scratch/x_johnsobk/archive/f.e21.FHIST_BGC.f09_025.CAM6assim.011/rest
$ mv 2019-08-05-000000 2019-08-05-000000-original
$ mkdir 2019-08-05-000000
$ cd 2019-08-05-000000
```

### 8.3.1 perturb\_single\_instance

Then we copy `perturb_single_instance` from the model's work directory.

```
$ cp /lustre/project/k1421/DART/models/cam-fv/work/perturb_single_instance ./
```

### 8.3.2 input.nml

Then we start by configuring `input.nml`, which must contain the `&perturb_single_instance_nml` namelist in order for the program to work.

```
$ cp /lustre/project/k1421/DART/models/cam-fv/work/input.nml.original ./input.nml
$ vim input.nml
```

We configure the namelist so that the program builds a 1000-member ensemble, looks for an initial file known as `caminput.nc` and a text file containing the names of the desired output files known as `output_file_list.txt`.

```
&perturb_single_instance_nml
  ens_size           = 1000
  input_files        = 'caminput.nc'
  output_files       = ''
  output_file_list   = 'output_file_list.txt'
  perturbation_amplitude = 0.2
/
```

An additional file is also specified in the `cam_phis_filename` field of the `&model_nml` namelist.

```
&model_nml
  cam_template_filename = 'caminput.nc'
  cam_phis_filename     = 'cam_phis.nc'
  ...
/
```

### 8.3.3 caminput.nc

We copy one of the initial files from Kevin's CAM reanalysis.

```
$ cp ../2019-08-05-000000-original/f.e21.FHIST_BGC.f09_025.CAM6assim.011.cam_0001.i.2019-
↪08-05-000000.nc ./caminput.nc
```

### 8.3.4 cam\_phis.nc

We need to transfer `cam_phis.nc` from Cheyenne.

```
$ sftp <user>@data-access.ucar.edu
$ get /glade/scratch/raeder/f.e21.FHIST_BGC.f09_025.CAM6assim.011/run/cam_phis.nc
```

### 8.3.5 output\_file\_list.txt

Then we write a short python script to generate a list of 1000 filenames to insert into output\_file\_list.txt.

```
$ vim make_list.py
```

The make\_list.py script is very simple.

```
#!/usr/bin/env python

prefix = 'f.e21.FHIST_BGC.f09_025.CAM6assim.011.cam_'
suffix = '.i.2019-08-05-000000.nc'

f = open('output_file_list.txt', 'w')

for iensemble in range(1, 1001):
    f.write(prefix+str(iensemble).zfill(4)+suffix+'\n')

f.close()
```

When we run it, it generates output\_file\_list.txt.

```
$ python make_list.py
$ ls -l output_file_list.txt
-rw-r--r-- 1 x_johnsobk g-x_johnsobk 69000 Nov 24 20:18 output_file_list.txt
```

## 8.4 Running perturb\_single\_instance

Finally, with all of those files in place, we can run perturb\_single\_instance by first starting an interactive job on one of Shaheen's compute nodes and then running the program.

**Warning:** Running perturb\_single\_instance on a login node on Shaheen will crash with an error similar to:

```
Fatal error in MPI_Init: Other MPI error, error stack:
MPIR_Init_thread(537):
MPID_Init(246).....: channel initialization failed
MPID_Init(647).....: PMI2 init failed: 1 libhugetlbfs
```

We use the jinter alias that we described in the Aliases section of the [SLURM](#) page to start an interactive job before running the program.

```
$ jinter
srun: job 16821392 queued and waiting for resources
srun: job 16821392 has been allocated resources
$ ./perturb_single_instance
...
-----
Starting ... at YYYY MM DD HH MM SS =
          2020 11 24 21 54 27
Program perturb_single_instance
```

(continues on next page)

(continued from previous page)

```
-----  
...  
-----  
Finished ... at YYYY MM DD HH MM SS =  
          2020 11 24 22  7 10  
Program perturb_single_instance  
-----
```



## SAMPLING ERROR TABLE

### 9.1 gen\_sampling\_err\_table

In order to apply Sampling Error Correction, we use `gen_sampling_err_table` to compute the statistics described in Anderson (2012).<sup>1</sup>

The table with the error correction values is stored in the netCDF file `sampling_error_correction_table.nc` in:

```
${DARTROOT}/assimilation_code/programs/gen_sampling_err_table/work/
```

However, the table typically only contains values for ensemble sizes up to 200 ensemble members. Thus we edit `input.nml` in the above directory thus we edit the following namelist to add ensemble sizes of 250, 500 and 1000:

```
&gen_sampling_error_table_nml  
  ens_sizes = 250, 500, 1000  
  debug = .false.  
/
```

We then run `quickbuild.csh` and `submit.csh` the latter of which simply submits a batch job that runs `gen_sampling_err_table` on a compute node.

The program required 01:53:14 on 8 CPUs to complete and add the correction values for the additional ensemble sizes.

### 9.2 DART\_config.template

Finally, we need to edit Line 325 of the `DART_config.template` script to reflect the fact that we are now able to run ensemble sizes of larger than 200. The template is here:

```
$ vim ${DARTROOT}/models/cam-fv/shell_scripts/cesm2_1/DART_config.template
```

And we change 200 to 1000 here:

```
if ( $num_instances < 3 || $num_instances > 1000 ) then
```

---

<sup>1</sup> Anderson, Jeffrey L. (2012) "Localization and Sampling Error Correction in Ensemble Kalman Filter Data Assimilation." *Monthly Weather Review* **140**: 2359-2371, doi: 10.1175/MWR-D-11-00013.1.

## 9.3 References

## RECOVERING FROM A FAILED ASSIMILATION

Sometimes the `assimilate.csh` script will error out and must be rerun.

For example, when running the Kilo-CAM ensemble for the first time, the integration took much, much longer than it should've and it hit the 12:00:00 job wallclock limit. I resubmitted the job, but there were extraneous CESM logs present due to the failed integration, thus when `assimilate.csh` was run it exited with the following message:

```
ERROR: Too many cesm.log files (3) for the 1 restart sets.
Clean out the cesm.log files from failed cycles.
```

So let's do just that:

```
$ cd /lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.090.e1000/run
$ ls cesm.log*
cesm.log.17201900.201203-174150  cesm.log.17246904.201206-104359.gz  cesm.log.17325936.
  201207-142703
$ mv cesm.log.17201900.201203-174150 ~/
```

The script comments say that:

```
The (resulting) assimilate.csh script is called by CESM with two arguments:
1) the CASEROOT, and
2) the assimilation cycle number in this CESM job
```

To continue past a trivial exit such as this one, the task is to rerun `assimilate.csh`. This script must be run from a batch job in order for the environmental variables to be set properly.

```
$ sbatch run_assimilate.csh
```

All `run_assimilate.csh` does is submit a batch job to SLURM in which `assimilate.csh` is run and passed two variables, `CASEROOT` and `DATA_ASSIMILATION_CYCLES`.

```
#!/bin/csh

#SBATCH --job-name=run_assimilate
#SBATCH --ntasks=320
#SBATCH --ntasks-per-node=32
#SBATCH --time=04:00:00
#SBATCH -A k1421
#SBATCH -p workq
#SBATCH -e run_assimilate.%j.err
#SBATCH -o run_assimilate.%j.out
```

(continues on next page)

(continued from previous page)

```
./assimilate.csh /lustre/project/k1421/cases/FHIST_BGC.f09_d025.090.e1000 1  
exit 0
```

## NCO

One of the CAM assimilation scripts used in the experiment uses nco:

```
/lustre/project/k1421/DART/models/cam-fv/shell_scripts/cesm2_1/assimilate.csh.template
```

It uses `ncgen` (which is already available on Shaheen II) and `ncks`:

```
[...]
994  ncgen -k netCDF-4 -o ${CASE}.dart.r.${scomp}.${ATM_DATE_EXT}.nc inf_restart_list.
→cdl
[...]
1000 module load nco
[...]
1019 ncks -O -v $vars[$c] $f ../run_shadow/$f
```

After the 25 January 2022 software update, the `nco` installation no longer worked, with the error message:

```
Error: nco: This library has not been built for this programming environment... please switch to GNU or Intel
programming environment first with the command (PrgEnv-intel).
```

This error message gets printed even if `PrgEnv-intel` is loaded.

It is possible to use Conda to install `nco`:

```
source activate py27
conda install -c conda-forge nco
```



## EXPERIMENTAL CONFIGURATION

### 12.1 Filters

- The Ensemble Kalman Filter (Evensen, 2003<sup>1</sup> )
- The Ensemble Adjustment Kalman Filter (Anderson, 2003<sup>2</sup> )
- The Ensemble Kalman Filter with Exact Second Order Perturbation Sampling (Hoteit et al. 2015<sup>3</sup> )

---

**Note:** `sort_obs_inc` must be set to `.true.` for EnKF-esops and EnKF.

---

---

**Important:** For Moha's diagnostics: `compute_posterior = .true.`

---

### 12.2 Inflation

Inverse gamma adaptive (Gharamti, 2018<sup>4</sup> ) prior inflation will be used for each of the three experiments.

### 12.3 Saved output

The following files will be saved from the integration:

- All `obs_seq.final` files
- All prior inflation mean and standard deviation files
- The preassim and output stages will be saved.

---

<sup>1</sup> Evensen, G., 2003: The Ensemble Kalman Filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53, 343–367, <https://doi.org/10.1007/s10236-003-0036-9>.

<sup>2</sup> Anderson, J. L., 2003: A Local Least Squares Framework for Ensemble Filtering. *Monthly Weather Review*, **131**, 634–642.

<sup>3</sup> Hoteit, I., D.-T. Pham, M. E. Gharamti, and X. Luo, 2015: Mitigating Observation Perturbation Sampling Errors in the Stochastic EnKF. *Monthly Weather Review*, **143**, 2918–2936.

<sup>4</sup> Gharamti M., 2018: Enhanced Adaptive Inflation Algorithm for Ensemble Filters. *Monthly Weather Review*, **2**, 623-640, doi:10.1175/MWR-D-17-0187.1

## 12.4 References



## **STEPS TO BEGIN AN EXPERIMENT**



## TESTING GET\_CLOSE CACHING FIX

This page describes a collection of ten three-member CAM-FV runs intended to test:

1. The effect of the `get_close_caching` fix in `filter_assim` in [DART pull request #364](#), and
2. The effect of the exact second order perturbation sampling ENKF filter (ESKF) in `assimilation_code/modules/assimilation/assim_tools_mod.f90`.

### 14.1 mkmf.template

The `mkmf.template` is set for the:

1. Cray wrappers for Intel fortran in both serial and parallel using `ftn`
2. Serial `netcdf`
3. No optimization
4. Intel oneAPI Math Kernel Library (oneMKL) using `qmk1`

```
MPIFC = ftn
MPILD = ftn
FC = ftn
LD = ftn

INCS = -I/opt/cray/pe/netcdf/4.7.4.4/INTEL/19.1/include
LIBS = -L/opt/cray/pe/netcdf/4.7.4.4/INTEL/19.1/lib -lnetcdff -lnetcdf

FFLAGS = -O0 -qmk1 $(INCS)
LDFLAGS = $(FFLAGS) $(LIBS)
```

### 14.2 Possible configurations

There are two DART versions:

1. the 2020 version, DART 9.X.X
2. the 2022 version, DART 10.5.3

and two statuses of each version:

1. the original status in which `assim_tools_mod.f90` is unchanged, and
2. the modified status in which `assim_tools_mod.f90` has been modified to include the ESKF filter.

Thus there are 10 possible configurations of these two statuses, two versions and three filters:

	DART	DART-esops	DART_10.5.3	DART_10.5.3-esops
EAKF	2020_original_EAKF	2020_modified_EAKF	2022_original_EAKF	2022_modified_EAKF
ENKF	2020_original_ENKF	2020_modified_ENKF	2022_original_ENKF	2022_original_ENKF
ESKF		2020_modified_ESKF		2022_modified_ESKF

It is only possible to build the ESKF filter with the two modified repositories.

### 14.2.1 input.nml settings

Filter	filter_kind	sort_obs_inc
EAKF	1	.false.
ENKF	2	.true.
ESKF	9	.true.

## 14.3 Differences

There are 13 sensible differences to check between the various repositories:

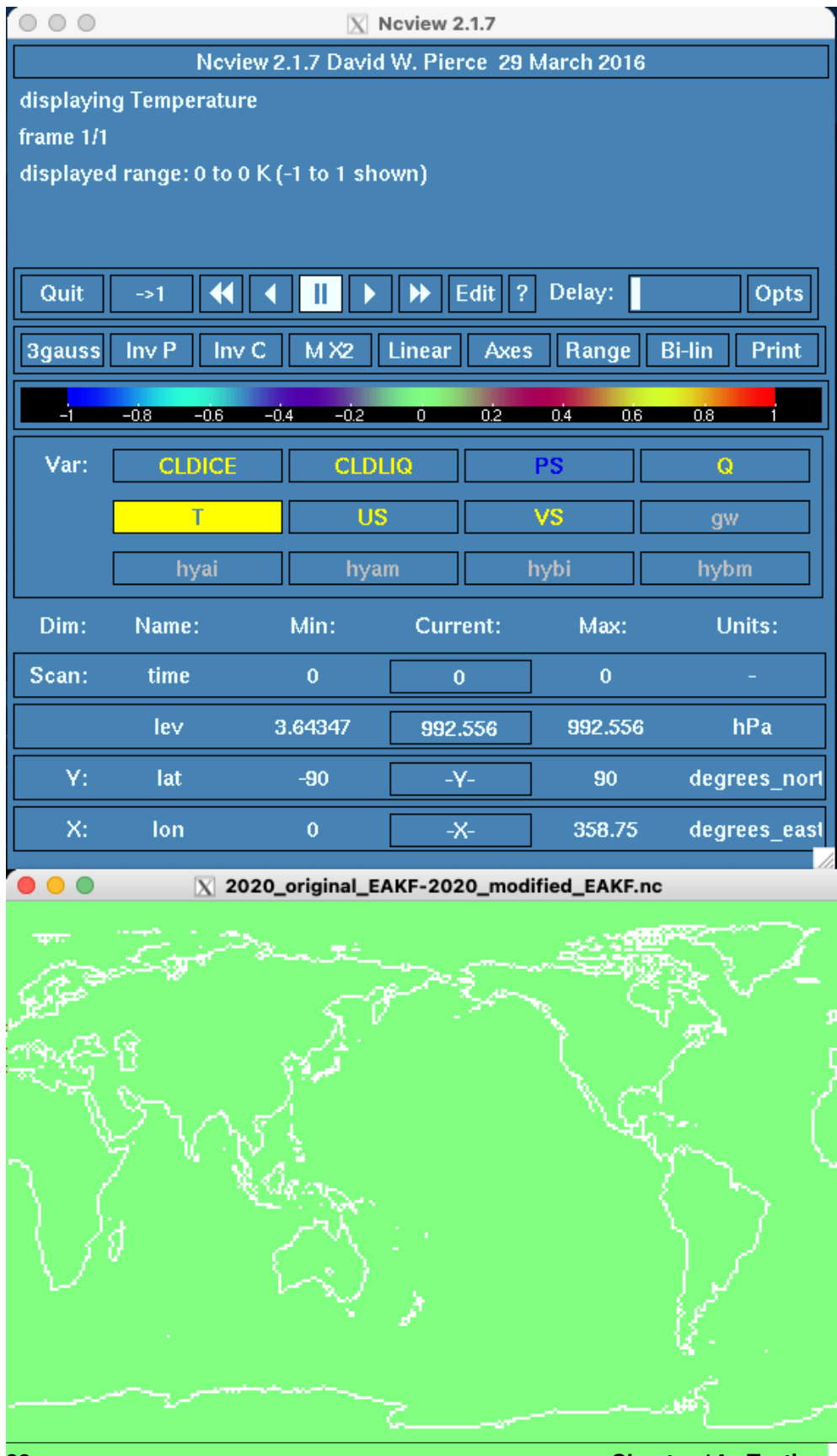
- Six possible differences between the EAKF experiments
- Six possible differences between the ENKF experiments
- 1 possible difference between the ESKF experiments

### 14.3.1 EAKF experiments

All of the EAKF experiments reproduce.

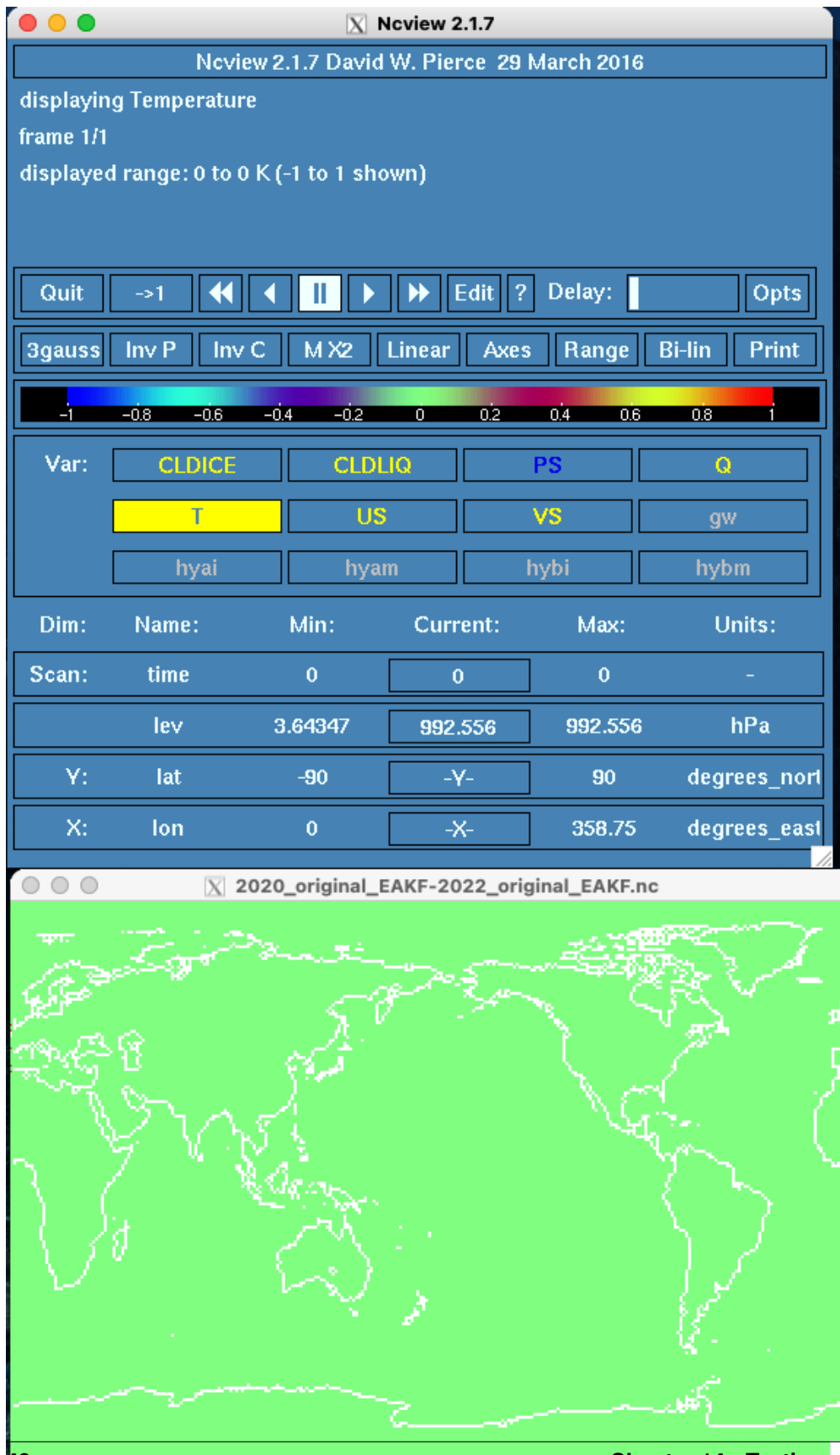


2020\_original\_EAKF-2020\_modified\_EAKF





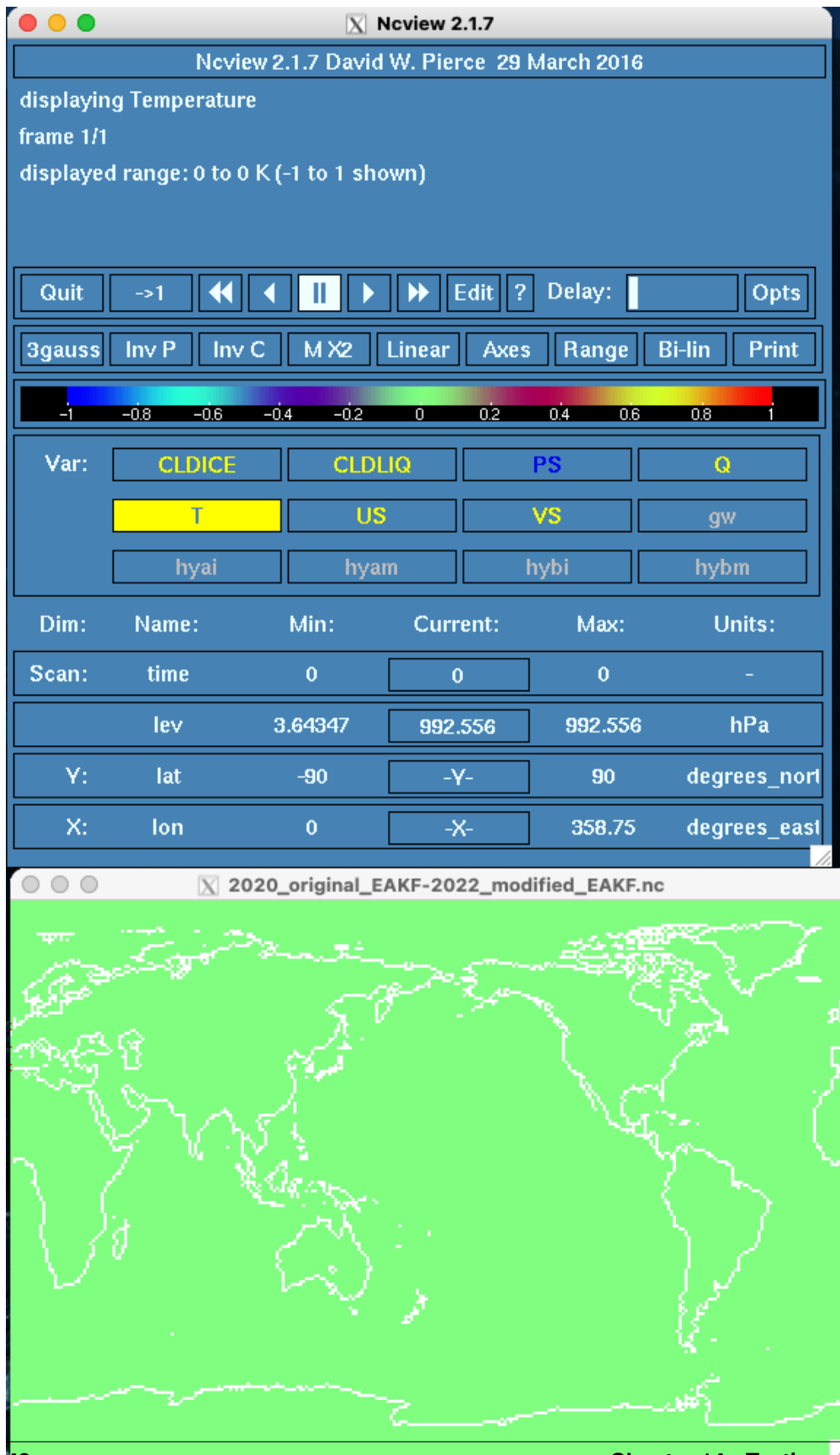
2020\_original\_EAKF-2022\_original\_EAKF





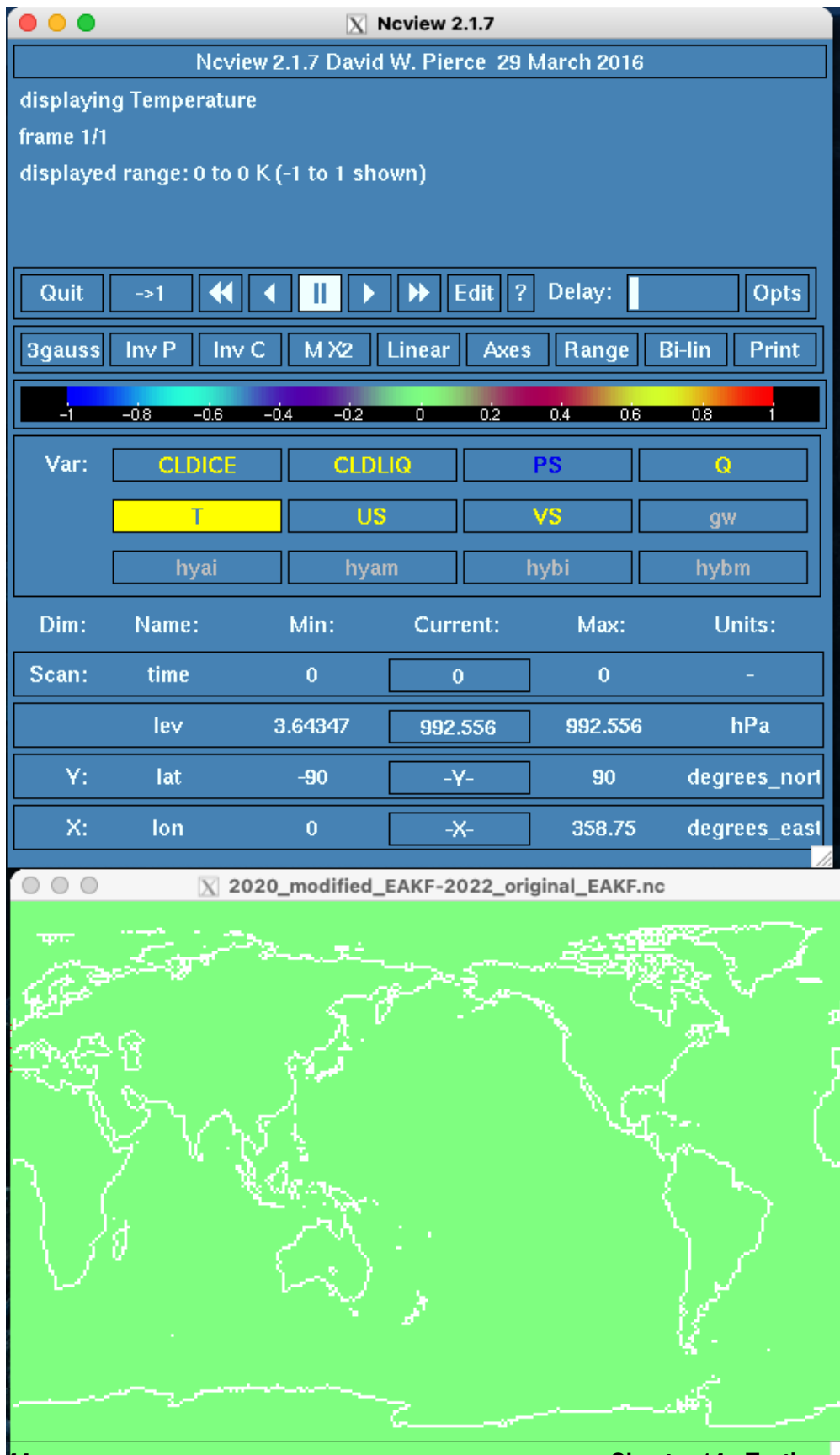


2020\_original\_EAKF-2022\_modified\_EAKF





## 2020\_modified\_EAKF-2022\_original\_EAKF





2020\_modified\_EAKF-2022\_modified\_EAKF





2022\_original\_EAKF-2022\_modified\_EAKF



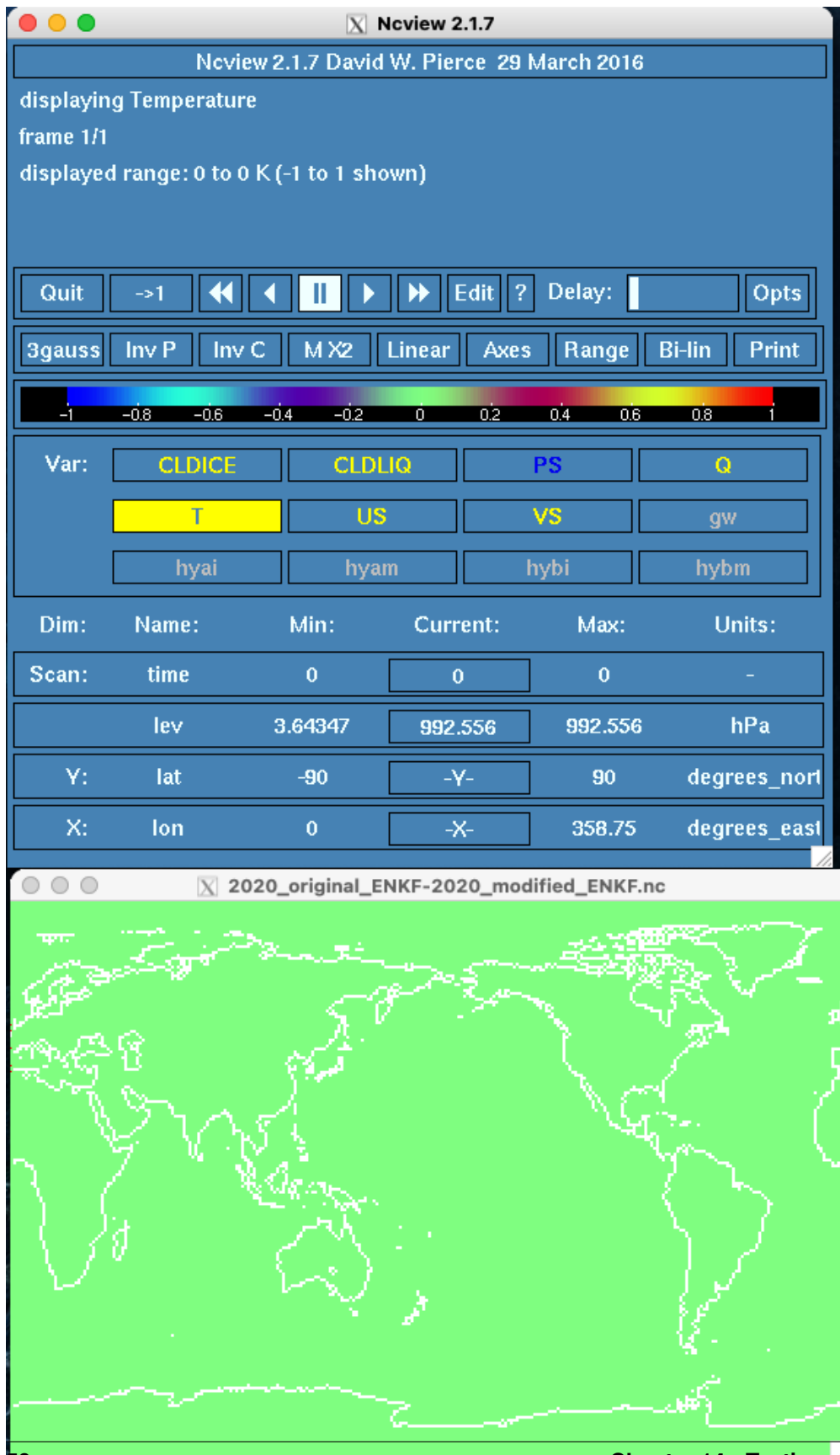


### 14.3.2 ENKF experiments

Only two of the ENKF comparisons reproduce:

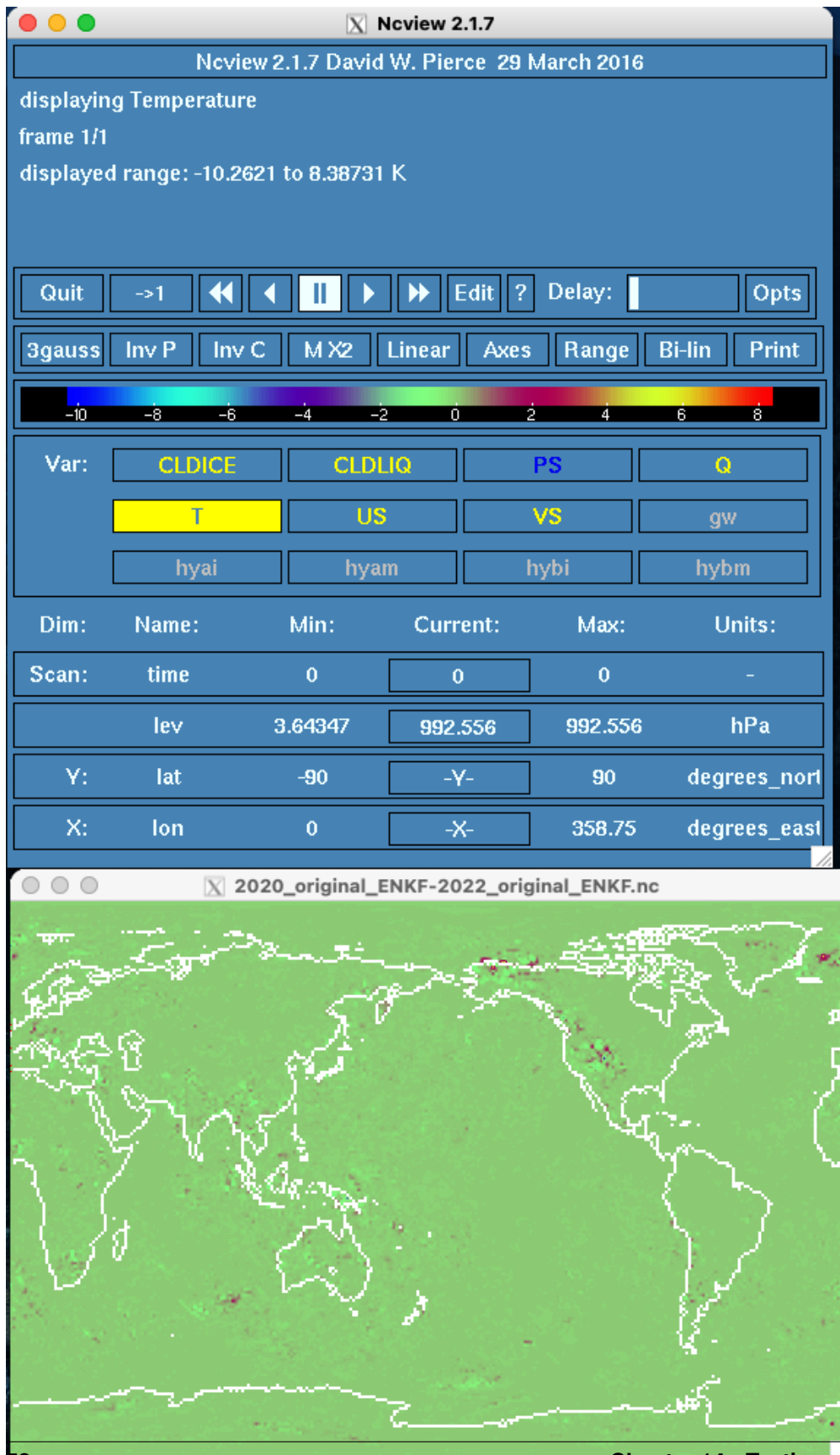
- the 2020 filters, both original and modified
- the 2022 filters, both original and modified

## 2020\_original\_ENKF-2020\_modified\_ENKF



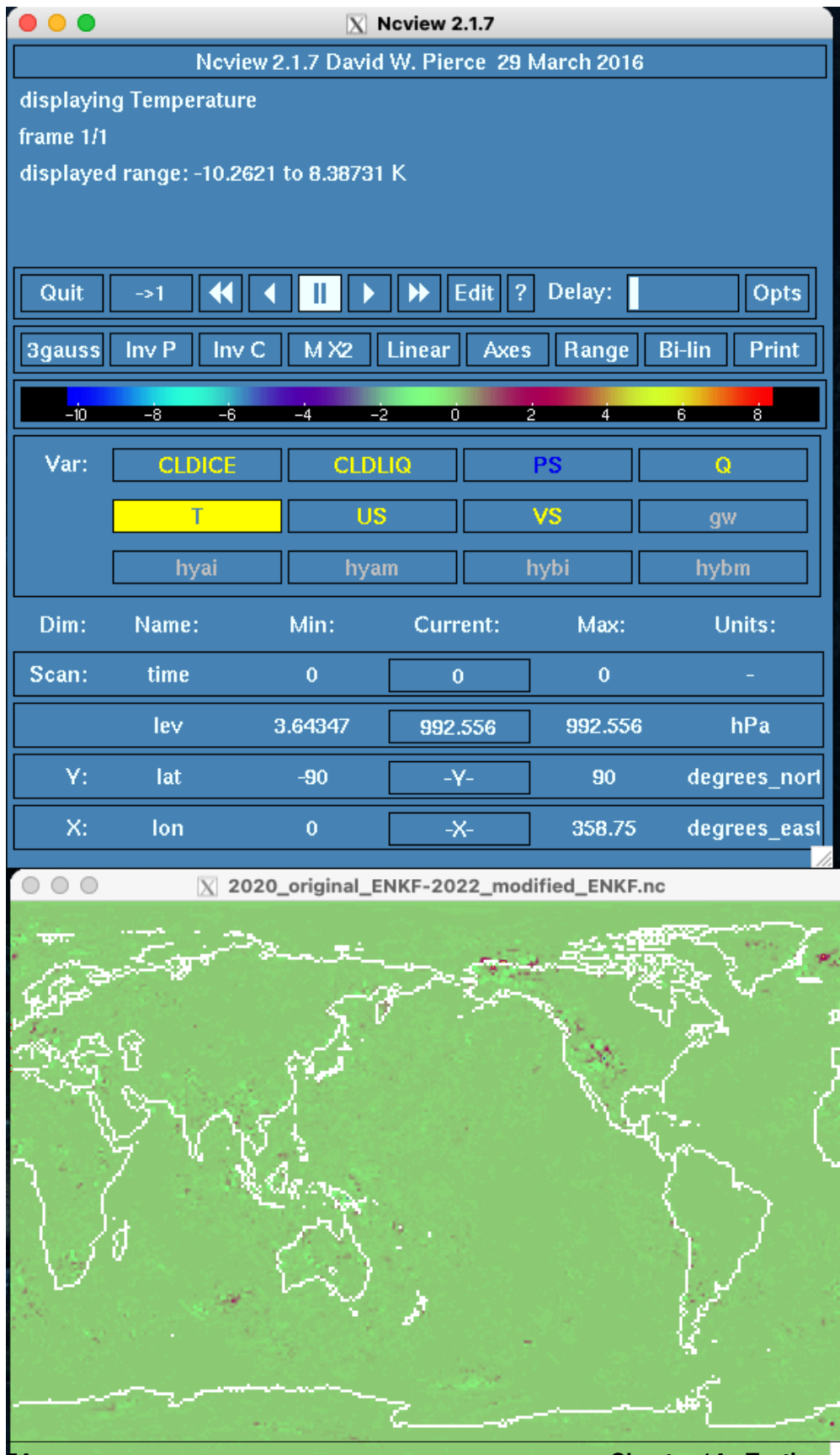


2020\_original\_ENKF-2022\_original\_ENKF





## 2020\_original\_ENKF-2022\_modified\_ENKF





## 2020\_modified\_ENKF-2022\_original\_ENKF





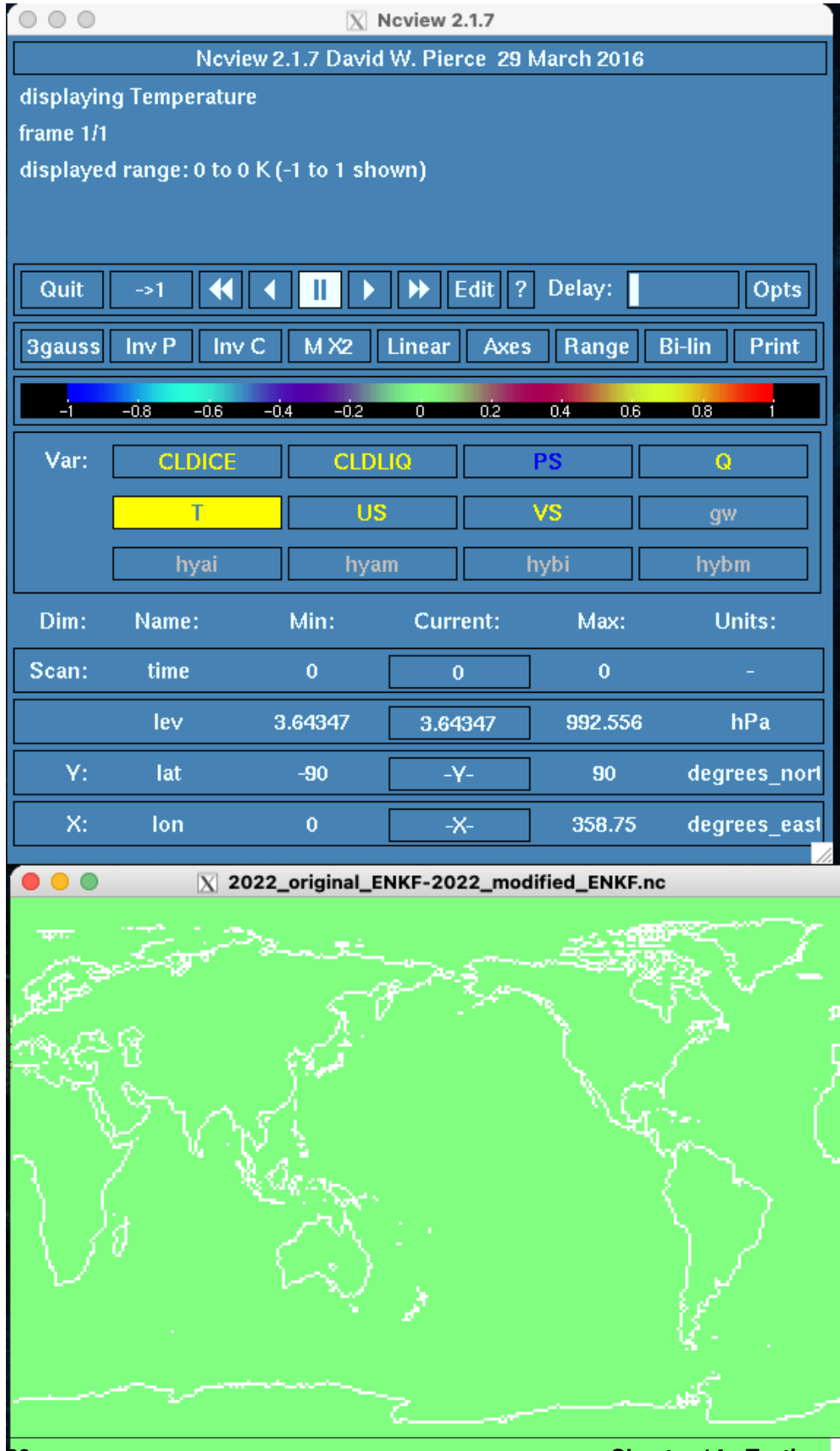


2020\_modified\_ENKF-2022\_modified\_ENKF





## 2022\_original\_ENKF-2022\_modified\_ENKF



### 14.3.3 ESKF experiments

The ESKF experiments do not reproduce.

2020\_modified\_ESKF-2022\_modified\_ESKF



## 15.1 Running obs\_diag on CAM output

Change directory to the case's run directory:

```
$ cd /lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.104.e0250/run
```

Echo a list of all of the obs\_seq.final files into a text file:

```
$ ls -d -1 "$PWD/"\*final\*00000\* > /lustre/project/k1421/DART/models/cam-fv/work/obs_
↪seq_final_list.txt
```

Change directory to DART's CAM-FV work directory:

```
$ cd /lustre/project/k1421/DART/models/cam-fv/work/
```

Edit the &obs\_diag\_nml namelist in input.nml to reflect our experimental configuration:

```
&obs_diag_nml
  obs_sequence_name = ''
  obs_sequence_list = 'obs_seq_final_list.txt'
  first_bin_center = 2019, 8, 6, 12, 0, 0
  last_bin_center  = 2019, 8, 18, 12, 0, 0
  bin_separation   = 0, 0, 1, 0, 0, 0
  bin_width        = 0, 0, 1, 0, 0, 0
  time_to_skip     = 0, 0, 0, 0, 0, 0
  max_num_bins     = 1000
  trusted_obs      = 'null'
  plevel_edges = 1036.5, 962.5, 887.5, 775, 600, 450, 350, 275, 225, 175, 125, 75,
↪ 35, 15, 2
  hlevel_edges = 200, 630, 930, 1880, 3670, 5680, 7440, 9130, 10530, 12290, 14650, 18220,
↪ 23560, 29490, 43000
  Nregions = 3
  lonlim1 = 0.0, 0.0, 0.0
  lonlim2 = 360.0, 360.0, 360.0
  latlim1 = 20.0, -20.0, -90.0
  latlim2 = 90.0, 20.0, -20.0
  reg_names = 'Northern Hemisphere', 'Tropics', 'Southern Hemisphere'
  print_mismatched_locs = .false.
  create_rank_histogram = .true.
  outliers_in_histogram = .true.
```

(continues on next page)

(continued from previous page)

```
use_zero_error_obs = .false.  
verbose           = .false.  
/  

```

Since Shaheen doesn't seem to be able to run the Fortran executables from the login nodes, we need to make a batch script to submit that merely just runs `obs_diag`.

```
$ vim run_obs_diag.csh
```

```
#!/bin/csh  
  
#SBATCH --job-name=run_obs_diag  
#SBATCH --account=k1421  
#SBATCH --ntasks=1  
#SBATCH --ntasks-per-node=1  
#SBATCH --time=00:59:00  
#SBATCH --partition=workq  
#SBATCH --output=run_obs_diag.out.%j  
  
./obs_diag  
  
exit 0
```

Submit the script we just created and wait for it to complete. It will create a file named `obs_diag_output.nc`.

```
$ sbatch run_obs_diag.csh  
[...]  
$ ls -lart  
obs_diag_output.nc
```

## 15.2 Using the MATLAB Diagnostic Scripts

Now that we have `obs_diag_output.nc`, we can use the MATLAB diagnostic scripts to see statistics about the ensemble. Navigate to the DART sub-directory containing the diagnostic scripts and start MATLAB.

```
$ mv obs_diag_output.nc $DARTROOT/diagnostics/matlab  
$ cd $DARTROOT/diagnostics/matlab  
$ matlab
```

Once MATLAB is open, run `plot_evolution.m` using:

```
>> fname = 'obs_diag_output.nc';  
>> copy  = 'bias';  
>> plotdat = plot_evolution(fname, copy);
```

This will output a slew of diagnostic plots.



## GRID AND COMPSET

### 16.1 Scientifically supported configuration

The scientifically supported configuration for CAM6 is the FHIST compset is the f09\_f09\_mg17 grid. For more information, see the [CAM documentation](#). Efficiency testing should be done with this grid before moving on to the CAM6 Reanalysis grid.

#### 16.1.1 Cori

---

**Note:** On Cori the default version of python is 2.7.18. Thus the multi-instance cesm2.1.3 build script doesn't crash because of attempted concatenation of a string and an int. There is no need to create a python 2.7 environment.

---

```
$ cd /project/projectdirs/m3857/cesm2_1_3/cime/scripts
$ ./create_newcase --case /project/projectdirs/m3857/cases/FHIST.f09_f09_mg17.e001.n0003
↪--compset FHIST --res f09_f09_mg17 --machine cori-haswell --project m3857 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
[ ... ]
Creating Case directory /project/projectdirs/m3857/cases/FHIST.f09_f09_mg17.e001.n0003
$ cd /project/projectdirs/m3857/cases/FHIST.f09_f09_mg17.e001.n0003
$ ./case.setup
$ ./case.build
[ ... ]
MODEL BUILD HAS FINISHED SUCCESSFULLY
$ ./case.submit -M begin,end
```

#### 16.1.2 Shaheen

**Caution:** On Shaheen the default version of python is 3.7.7. Thus the multi-instance cesm2.1.3 build script crashes because of attempted concatenation of a string and an int (which isn't allowed in python 3). It was necessary to create a python 2.7 environment using conda. It is activated using the `source activate py27` line.

Additionally, the LibXML2 library is missing from most of the login nodes. Thus it is necessary to build on cdl5.

```
$ ssh cd15
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts
$ source activate py27
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e001.n0003 --
→compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --
→ninst 3 --multi-driver --walltime 2:00:00
[ ... ]
Creating Case directory /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e001.n0003
$ cd /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e001.n0003
$ ./case.setup
$ ./case.build
[ ... ]
MODEL BUILD HAS FINISHED SUCCESSFULLY
$ ./case.submit -M begin,end
```

### 16.1.3 Cheyenne

**Note:** On Cheyenne the default version of python is 2.7.16. Thus the multi-instance cesm2.1.3 build script doesn't crash because of attempted concatenation of a string and an int. There is no need to create a python 2.7 environment.

**Warning:** Running a build of CESM on a Cheyenne login node will likely result in one of the daemons killing the build script due to the build “using an excessive amount of the total available CPU resources.” Thus use the qcmd build wrapper and options for the ./case.build script.

```
$ cd /glade/work/johnsonb/cesm2_1_3/cime/scripts
$ ./create_newcase --case /glade/work/johnsonb/cases/FHIST.f09_f09_mg17.e001.n0003 --
→compset FHIST --res f09_f09_mg17 --machine cheyenne --project P86850054 --run-
→unsupported --ninst 3 --multi-driver --walltime 2:00:00
[ ... ]
Creating Case directory /glade/work/johnsonb/cases/FHIST.f09_f09_mg17.e001.n0003
$ cd /glade/work/johnsonb/cases/FHIST.f09_f09_mg17.e001.n0003
$ ./case.setup
$ qcmd -q share -l select=1 -A P86850054 -- ./case.build
[ ... ]
MODEL BUILD HAS FINISHED SUCCESSFULLY
$ ./case.submit -M begin,end
```

## 16.2 Custom Grid

The port on Shaheen II uses the same grid as the [DART CAM6 Reanalysis](#), which is a custom grid created by Kevin Raeder, called f09\_d025.

The grid specification is available on GLADE here:

```
/glade/work/raeder/Models/CAM_init/SST/config_grids+fv1+2deg_oi0.25_gland20.xml
```

The grid has quarter-degree horizontal resolution, so that CAM can be forced with the [AVHRR quarter-degree SST product](#).

## 16.3 Compset

The integration uses an F with biogeochemistry. Kevin's CASEROOT on GLADE and his README.case are:

```
/glade/work/raeder/Exp/f.e21.FHIST_BGC.f09_025.CAM6assim.011/README.case
```

The create\_newcase command invoked is as follows:

```
/glade/work/raeder/Models/cesm2_1_relsd_m5.6/cime/scripts/create_newcase --case /glade/
↪work/raeder/Exp/f.e21.FHIST_BGC.f09_025.CAM6assim.011 --machine cheyenne --res f09_
↪d025 --project NCIS0006 --queue premium --walltime 1:00 --pecount 36x1 --ninst 80 --
↪compset HIST_CAM60_CLM50%BGC-CROP_CICE%PRES_DOCN%DOM_MOSART_SGLC_SWAV --run-
↪unsupported --multi-driver --gridfile /glade/work/raeder/Models/CAM_init/SST/config_
↪grids+fv1+2deg_oi0.25_gland20.xml
```



## LIBRARIES

### 17.1 Parallel NetCDF String

The default makefile in `${CIMERROOT}/scripts/Tools/Makefile` has a hard-wired compiler flag to link the parallel NetCDF library in it, of the form `-lnetcdf`. This works for the `parallel_netcdf` library since the static library is:

```
``/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib/libpnetcdf.a``
```

However, the `netcdf-hdf5parallel` library has a compiler flag of the form `-lnetcdff_parallel` since the static library is:

```
``/opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0/lib/libnetcdff_parallel.a``
```

I attempted to rewrite the compiler link flag in `${CIMERROOT}/scripts/Tools/Makefile` but perhaps did not rewrite the flag correctly.

### 17.2 Instructions Regarding Macros.make

Brian Dobbins suggests:

You should just be able to modify (as a test) the `Macros.make` file in a case directory and add something like:

```
SLIBS += -lnetcdf_parallel
```

That should let it find that library (though it might complain if `SLIBS` already has `-lnetcdf` in it and it can't find that one, in which case you can do:

```
SLIBS := -lnetcdf_parallel
```

You might need to add other things, like the math libraries or NetCDF libraries, too. For example, on the cloud config, I use:

```
SLIBS := -lalapack -lblas -lnetcdf -lnetcdf
```

## 17.3 Attempting to Build a Case Using Macros.make

We'll try to build a case using Brian's instructions. The first step is editing `config_machines.xml`.

### 17.3.1 config\_machines.xml

```
$ vim /lustre/project/k1421/cesm2_1_3/cime/config/cesm/machines/config_machines.xml
```

Edit the module loaded when the mpi library is not mpi-serial.

```
<modules mpilib="!mpi-serial">
  <command name="load">cray-netcdf-hdf5parallel/4.6.3.2</command>
</modules>
[...]
<environment_variables>
  <env name="PNETCDF_PATH">/opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0</env>
</environment_variables>
```

Again, if we examine the parallel netCDF libraries:

```
$ ls /opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0/lib
[...]
libnetcdf_parallel.a
```

We see that the linker flag should be `-lnetcdff_parallel`.

### 17.3.2 Setup Case

We're trying to follow Brian Dobbins' instructions about using `Macros.make` to change the compiler flags in order to use a different parallel netCDF library.

```
$ source activate py27
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts/
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.f09_f09_mg17.001 --compset_
↪ FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --ninst 3_
↪ --multi-driver --walltime 2:00:00
[...]
Creating Case directory /lustre/project/k1421/cases/FHIST.f09_f09_mg17.001
$ cd /lustre/project/k1421/cases/FHIST.f09_f09_mg17.001
$ ./case.setup
```

### 17.3.3 Edit Macros.make

After `case.setup` is run the `Macros.make` file is available in the `$CASEROOT` directory. According to the instructions from Brian reposted above, we should edit the `Macros.make` to add the `-lnetcdff_parallel` flag:

```
ifeq ($(MPILIB),mvapich2)
  SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif
ifeq ($(MPILIB),mpich2)
```

(continues on next page)

(continued from previous page)

```

    SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif
ifeq ($(MPILIB),mpt)
    SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif
ifeq ($(MPILIB),openmpi)
    SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif
ifeq ($(MPILIB),mpich)
    SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif
ifeq ($(MPILIB),mvapich)
    SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif
ifeq ($(MPILIB),impi)
    SLIBS := $(SLIBS) -lnetcdff_parallel -mkl=cluster
endif

```

After editing Macros.make we attempt to build the case:

```

$ case.build
[...]
ERROR: /lustre/project/k1421/cesm2_1_3/cime/src/build_scripts/buildlib.pio
FAILED, cat /lustre/scratch/x_johnsobk/FHIST.f09_f09_mg17.001/bld/pio.bldlog.210106-
↳234616
cat /lustre/scratch/x_johnsobk/FHIST.f09_f09_mg17.001/bld/pio.bldlog.210106-234616

```

**Error:** In spite of following the instructions, this procedure still results in the same error that we arrived at while attempting to edit the Makefile directly. Here we reprint the relevant section of /lustre/scratch/x\_johnsobk/FHIST.f09\_f09\_mg17.001/bld/pio.bldlog.210106-234616:

```

Configuring incomplete, errors occurred!
See also "/lustre/scratch/x_johnsobk/FHIST.f09_f09_mg17.001/bld/intel/mpt/nodebug/
↳nothreads/pio/pio1/CMakeFiles/CMakeOutput.log".
gmake: Leaving directory '/lustre/scratch/x_johnsobk/FHIST.f09_f09_mg17.001/bld/intel/
↳mpt/nodebug/nothreads/pio/pio1'
cat: Filepath: No such file or directory
cat: Srcfiles: No such file or directory
Building PIO with netcdf support
CMake Error at /sw/xc40cle7/cmake/3.13.4/sles15_gcc7.4.1/install/share/cmake-3.13/
↳Modules/FindPackageHandleStandardArgs.cmake:137 (message):
Could NOT find PnetCDF_Fortran (missing: PnetCDF_Fortran_LIBRARY
PnetCDF_Fortran_INCLUDE_DIR)

```





## LIST OF ATTEMPTS

It took very, very many attempts to get a CAM ensemble performing well on Shaheen.

### 18.1 Trial 102

While we're waiting, why don't we try to build with just the serial NetCDF library in `cray-netcdf-hdf5parallel` to see if that's faster.

Removed this line from `config_machines.xml`:

```
<env name="PNETCDF_PATH">/opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0</env>
```

### 18.2 Trial 101

Tried various configurations of the linker flag in the Makefile:

```
312 SLIBS += -L$(LIB_PNETCDF) -lnetcdff_parallel
```

```
312 SLIBS += -L$(LIB_PNETCDF) -lnetcdff_parallel -lnetcdf_parallel
```

```
312 SLIBS += -L$(LIB_PNETCDF) -lpnetcdf
```

These all error out.

**Error:**

```
-- Configuring incomplete, errors occurred!
See also "/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.101.e0003/bld/intel/mpt/
↳nodebug/nothreads/pio/pio1/CMakeFiles/CMakeOutput.log".
gmake: Leaving directory '/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.101.e0003/bld/
↳intel/mpt/nodebug/nothreads/pio/pio1'
cat: Filepath: No such file or directory
cat: Srcfiles: No such file or directory
Building PIO with netcdf support
CMake Error at /sw/xc40cle7/cmake/3.13.4/sles15_gcc7.4.1/install/share/cmake-3.13/
↳Modules/FindPackageHandleStandardArgs.cmake:137 (message):
Could NOT find PnetCDF_Fortran (missing: PnetCDF_Fortran_LIBRARY
PnetCDF_Fortran_INCLUDE_DIR)
```

## 18.3 Trial 100

Trial 099 showed us that we can't simply remove NetCDF from the build. Let's try to get the build to compile with netcdf-hdf5parallel. We tried this earlier and couldn't get the linking flag to work. Brian Dobbins suggested editing the makefile.

This attempt requires us to make several changes to config\_machines.xml:

```
<modules mpilib="!mpi-serial">
  <command name="load">cray-netcdf-hdf5parallel/4.6.3.2</command>
</modules>

<environment_variables>
  <env name="PNETCDF_PATH">/opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0</env>
</environment_variables>
```

Without changing the makefile, we get the following error.

### Error:

```
- Configuring incomplete, errors occurred! See also "/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.100.e0003/bld/intel/mpt/nodebug/not
gmake: Leaving directory '/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.100.e0003/bld/intel/mpt/nodebug/notreads/pio/pio1'
```

Thus we attempt to edit /lustre/project/k1421/cesm2\_1\_3/cime/scripts/Tools/Makefile.

```
312 SLIBS += -L$(LIB_PNETCDF) -lpnetcdf
```

to reflect the name of the static library:

```
312 SLIBS += -L$(LIB_PNETCDF) -lnetcdff_parallel
```

Even attempted to make a spoof directory:

```
<env name="PNETCDF_PATH">"/lustre/project/k1421/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0</env>
```

with symbolic links:

```
$ ln -s /opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0/lib/libnetcdff_parallel.a
↪ lpnetcdff.a
$ ln -s /opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0/lib/libnetcdf_parallel.a
↪ lpnetcdf.a
```

None of this works.

## 18.4 Trial 099

Commenting out NETCDF\_PATH from config\_machines.xml to see if that affects the initialization time. We are trying to test the idea that only allowing the build to use parallel netCDF might work better.

Also **removing** these lines from config\_compilers.xml:

```
<SLIBS>
  <append> -L$(NETCDF_PATH) -lnetcdff -Wl,--as-needed,-L$(NETCDF_PATH)/lib -lnetcdff -
  ↪lnetcdf </append>
</SLIBS>
<NETCDF_PATH>${ENV{NETCDF_PATH}}</NETCDF_PATH>
```

This errors out.

**Error:** NETCDF not found: Define NETCDF\_PATH or NETCDF\_C\_PATH and NETCDF\_FORTRAN\_PATH

## 18.5 Trial 098

Changing MAX\_TASKS\_PER\_NODE back to 32 since we haven't figured out how it's used in setup\_advanced\_Rean and why it's doubling the requested nodes.

Building a 3 member ensemble for determining a baseline for Init Time.

An example timing file from ensemble member 0003 is here:

```
$ vim /lustre/project/k1421/cases/FHIST_BGC.f09_d025.098.e0003/timing/cesm_timing_0003.
↪FHIST_BGC.f09_d025.098.e0003.17867122.201223-212516
Init Time   :    1375.139 seconds
Run Time    :    266.362 seconds    1065.447 seconds/day
```

The Init Time of 1375.139 seconds is our benchmark for improvement.

## 18.6 Trial 095

Building with 500 ensemble members since we will not be able to get through the experiment, even with 4 million core hours given the current level of performance efficiency.

```
$ cd /lustre/project/k1421/cases/FHIST_BGC.f09_d025.093.e500
```

Also made sure that DEBUG=FALSE in setup\_advanced\_Rean:

```
# DEBUG = TRUE implies turning on run and compile time debugging.
# INFO_DEBUG level of debug output, 0=minimum, 1=normal, 2=more, 3=too much.
./xmlchange DEBUG=FALSE
./xmlchange INFO_DEBUG=0
```

## 18.7 Trial 094

Rebuilding with 1000 members in case the efforts to fix the broken Trial 090 take much longer than expected.

```
./xmlchange DOUT_S=TRUE
```

## 18.8 Trial 093

```
$ cd /lustre/project/k1421/cases/FHIST_BGC.f09_d025.093.e500
$ ./xmlquery JOB_WALLCLOCK_TIME
Results in group case.run
JOB_WALLCLOCK_TIME: 12:00:00
Results in group case.st_archive
JOB_WALLCLOCK_TIME: 1:00
$ ./xmlchange --subgroup case.run JOB_WALLCLOCK_TIME=12:00:00
```

## 18.9 Trial 092

To avoid the error in Trial 91, we changed:

```
&filter_nml ... single_file_in = .false. perturb_from_single_instance = .false. ... / making another attempt.
```

## 18.10 Trial 091

Since 090 takes so long to build, this case runs with 100 members to see if the copied restart files for ensemble members 81-100 are able to be used without issue before we run with 1000 members.

### 18.10.1 Result

The case runs to completion, however, the assimilation fails perhaps because we changed `single_file_in = .true.` to get the perturbation working.

From the log file `/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.091.e100/run/da.log.17183899.201202-061559`:

```
Error:
ERROR FROM:
PE 512: direct_netcdf_mod:
routine: direct_netcdf_mod:
message: If using single_file_in/single_file_out = .true.
message: ... you must have a member dimension in your input/output file.
Rank 512 [Wed Dec 2 08:08:45 2020] [c6-1c2s11n0] application called MPI_
↪ Abort(comm=0x840000004, 99) - process 512
```

## 18.11 Trial 090

Since we don't know how long it'll take to get through the run we change the wallclock time to 12:00:00.

```
$ cd /lustre/project/k1421/cases/FHIST_BGC.f09_d025.090.e1000
$ ./xmlquery JOB_WALLCLOCK_TIME
Results in group case.run
JOB_WALLCLOCK_TIME: 12:00:00
Results in group case.st_archive
JOB_WALLCLOCK_TIME: 1:00
$ ./xmlchange --subgroup case.run JOB_WALLCLOCK_TIME=12:00:00
```

After the error in Trial 091, we change the filter namelist since aren't using perturb\_from\_single\_instance:

```
&filter_nml
...
single_file_in           = .false.
perturb_from_single_instance = .false.
...
/
```

## 18.12 Trial 089

Testing how to use .i. files in a hybrid run.

I'm not sure how to use .i. files in a hybrid run rather than .r. files.

Here is the [relevant page in the CIME documentation](#).

To test whether we can do this, change rpointer atm contents to .i. rather than .r. and see if it works in line 1257 of setup\_advanced\_Rean.

```
$ ncdump -h cam_initial_0001.nc
```

shows that the initial file is an 'i' file rather than an 'r' file.

This, and the change from 'r' to 'i', also seems to suggest that it's an initial file: /lustre/scratch/x\_johnsobk/FHIST\_BGC.f09\_d025.088.e03/run/cam\_initial\_0001.nc.

In setup\_advanced\_Rean Line 1283, we're linking the 'i' files here. Be able to explain what the purpose is of the slwe of rpointer files:

```
@ inst=1
while (\$inst <= $num_instances)
  set inst_string = \`printf _%04d \$inst\`
  ${LINK} -f ${case}.cam\${inst_string}.i.\${restart_time}.nc cam_initial\${inst_
↪string}.nc
  @ inst ++
end
```

## 18.13 Trial 088

Edit DART\_config.template to set:

```
./xmlchange DATA_ASSIMILATION_ATM=TRUE
...
if ($?CIMEROOT) ./xmlchange DATA_ASSIMILATION_SCRIPT=${CASEROOT}/assimilate.csh
```

Do adaptive inflation and run an assimilation cycle.

```
inf_flavor = 5, 0
```

Is it worth considering cray-mpich versus mpt?

In config\_machines.xml try adding:

```
NCAR_LIBS_PNETCDF=-Wl,-Bstatic -lpnetcdf -Wl,-Bdynamic
```

Add -Wl,-Bstatic -lpnetcdf -Wl,-Bdynamic.

## 18.14 Trial 087

Change to 1 node per instance.

Compare this with Trial 086. It looks *nearly* the same for total cost.

```
setenv timewall 2:30
total pes active           : 96
mpi tasks per node         : 32
pe count for cost estimate : 96
Overall Metrics:
  Model Cost:              51504.09  pe-hrs/simulated_year
  Model Throughput:         0.04    simulated_years/day
  Init Time   :    1837.831 seconds
  Run Time    :    1322.879 seconds    5291.516 seconds/day
  Final Time  :         0.081 seconds
```

## 18.15 Trial 086

Changing ESP to 32.

```
total pes active           : 384
mpi tasks per node         : 32
pe count for cost estimate : 384
Overall Metrics:
  Model Cost:              50881.78  pe-hrs/simulated_year
  Model Throughput:         0.18    simulated_years/day
  Init Time   :    1364.705 seconds
  Run Time    :     326.724 seconds    1306.895 seconds/day
  Final Time  :         0.017 seconds
  Actual Ocn Init Wait Time :         0.000 seconds
```

(continues on next page)

(continued from previous page)

```
Estimated Ocn Init Run Time   :      0.006 seconds
Estimated Run Time Correction :      0.006 seconds
(This correction has been applied to the ocean and total run times)
```

## 18.16 Trial 085

Same as Trial 084, except changing:

```
<arg name="binding" > --cpu_bind=cores</arg>
```

## 18.17 Trial 084

Same configuration as in Trial 083 except we restripe the forcing files in accordance with Bilel Hadri's recommendations.

## 18.18 Trial 083

Same configuration as Trial 082 except all of the restart files are no longer striped.

The first submission takes 00:32:41.

The second submission with `./case.submit --skip-preview-namelist -M begin,end` takes 00:32:04.

## 18.19 Trial 082

Setting `config_machines.xml`:

```
<modules mpilib="mpi-serial">
  <command name="load">cray-hdf5/1.10.5.2</command>
  <command name="load">cray-netcdf/4.6.3.2</command>
</modules>
<modules mpilib="!mpi-serial">
  <command name="load">cray-hdf5-parallel/1.10.5.2</command>
  <command name="load">cray-parallel-netcdf/1.11.1.1</command>
</modules>
<env name="NETCDF_PATH">/opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0</env>
<env name="PNETCDF_PATH">/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0</env>
```

Setting `config_compilers.xml`:

```
<SLIBS>
  <append> -L$(NETCDF_PATH) -lnetcdff -Wl,--as-needed,-L$(NETCDF_PATH)/lib -lnetcdff -
  ↪lnetcdf </append>
</SLIBS>
<MPICC> cc </MPICC>
<MPICXX> CC </MPICXX>
```

(continues on next page)

(continued from previous page)

```
<MPIFC> ftn </MPIFC>
<PIO_FILESYSTEM_HINTS>lustre</PIO_FILESYSTEM_HINTS>
<NETCDF_PATH>${ENV{NETCDF_PATH}}</NETCDF_PATH>
```

## 18.20 Trial 081

Setting config\_machines.xml:

```
<command name="load">cray-netcdf/4.6.3.2</command>
<command name="load">cray-netcdf-hdf5parallel/4.6.3.2</command>
```

Setting config\_compilers.xml:

```
<SLIBS>
  <append> -L$(NETCDF_PATH) -lnetcdff -Wl,--as-needed,-L$(NETCDF_PATH)/lib -lnetcdff -
  ↪lnetcdf </append>
</SLIBS>
<MPICC> cc </MPICC>
<MPICXX> CC </MPICXX>
<MPIFC> ftn </MPIFC>
<PIO_FILESYSTEM_HINTS>lustre</PIO_FILESYSTEM_HINTS>
<NETCDF_PATH>${ENV{NETCDF_PATH}}</NETCDF_PATH>
```

## 18.21 Trial 079

Let's try use the slibs tag instead of the parallel netcdf tag.

Setting config\_compilers.xml:

```
<SLIBS>
  <append>-L$(PNETCDF_PATH_KAUST)/lib -lnetcdff_parallel</append>
</SLIBS>
<MPICC> cc </MPICC>
<MPICXX> CC </MPICXX>
<MPIFC> ftn </MPIFC>
<PIO_FILESYSTEM_HINTS>lustre</PIO_FILESYSTEM_HINTS>
<NETCDF_PATH>${ENV{NETCDF_PATH_KAUST}}</NETCDF_PATH>
```

This errors out:

**Error:** 65: PNETCDF not enabled in the build

```
--prefix      -> /opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0
--includedir  -> /opt/cray/pe/netcdf/4.6.3.2/include
--libdir      -> /opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0/lib
```



**Note:** Typos in the CIME documentation: [https://esmci.github.io/cime/versions/master/html/users\\_guide/porting-cime.html](https://esmci.github.io/cime/versions/master/html/users_guide/porting-cime.html) Should be `cime/tools/load_balancing_tool` – no “s”.

## 18.22 Trial 078

Made these changes to `DART_config`:

```
# ./xmlchange DATA_ASSIMILATION_ATM=TRUE
# if ($?CIMEROOT) ./xmlchange DATA_ASSIMILATION_SCRIPT=${CASEROOT}/assimilate.csh
```

When we want to run `assimilate`, we need to undo this change.

Changed `PNETCDF_PATH_KAUST` within `config_machines.xml` to:

```
<env name="PNETCDF_PATH_KAUST">/opt/cray/pe/netcdf-hdf5parallel/4.6.3.2/INTEL/19.0</env>
```

Looking at the `pio.bldlogs`.

### 18.22.1 Working version

`/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.075.e03/bld/pio.bldlog.201027-211925.gz`

In this working build, these lines are printed:

```
-- Found NetCDF_Fortran: /opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0/lib/libnetcdf.a
-- Found PnetCDF_Fortran: /opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib/
↳ libpnetcdf.a
```

### 18.22.2 Non-working version

`/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.078.e03/bld/pio.bldlog.201104-002417`

In this non-working build, this is the final line before a build error:

```
Found NetCDF_Fortran: /opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0/lib/libnetcdf.a
-- Configuring incomplete, errors occurred!
```

Could the issue be that the library is not called `libpnetcdf` and instead is called `libnetcdf_parallel` but `-lpnetcdf` is hard-coded into the `makefile`?

## 18.23 Trial 077

This is the same as Trial 075, we're just rebuilding to sidestep this [NetCDF Issue](#).

```
$ cd /lustre/project/k1421/cesm_store/inputdata/atm/cam/tracer_cnst/
$ mv tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216.nc old_tracer_cnst_halons_
↳ WACCM6_3Dmonthly_L70_1975-2014_c180216.nc
$ nccopy -k cdf5 old_tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216.nc tracer_
↳ cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216.nc
```

## 18.24 Trial 076

Setting config\_compilers.xml:

```
<SLIBS>
<append>-L$(NETCDF_PATH_KAUST)/lib -lnetcdff</append>
</SLIBS>
<!--<append>-L$(NETCDF_PATH_KAUST) -lnetcdff, -L$(PNETCDF_PATH_KAUST) -lpnetcdf</append>--
↪->
```

## 18.25 Trial 075

Comment out the append line in config\_compilers.xml:

```
<!--<append>-L$(NETCDF_PATH_KAUST) -lnetcdff, -L$(PNETCDF_PATH_KAUST) -lpnetcdf</append>--
↪->
```

```
223: MOSART decomp info proc =          95 begr =    192376 endr =    194400 numr =    ↪
↪2025
srun: Job step aborted: Waiting up to 302 seconds for job step to finish.
0: slurmstepd: error: *** STEP 16076077.0 ON nid01376 CANCELLED AT 2020-10-28T19:25:55↪
↪DUE TO TIME LIMIT ***
srun: got SIGCONT
```

The second run completed the atmospheric portion.

This is the last printed statment in all three rof files is:

```
001
hist_htapes_build Successfully initialized MOSART history files
-----
(Rtmini) done
Snow capping will flow out in frozen river runoff
002
hist_htapes_build Successfully initialized MOSART history files
-----
(Rtmini) done
Snow capping will flow out in frozen river runoff
003
hist_htapes_build Successfully initialized MOSART history files
-----
(Rtmini) done
Snow capping will flow out in frozen river runoff
```

Rerunning Trial 075 to see if it hangs at the same spot. Started at 12:42.

## 18.26 Trial 074

Setting config\_compilers.xml:

```
<append>-L$(NETCDF_PATH_KAUST) -lnetcdff, -L$(PNETCDF_PATH_KAUST) -lpnetcdf</append>
```

## 18.27 Trial 073

Setting config\_compilers.xml:

```
-L$(NETCDF_DIR) -lnetcdff -Wl,--as-needed,-L$(NETCDF_DIR)/lib -lnetcdff -lnetcdf
<append>-L$(NETCDF_PATH_KAUST) -lnetcdff -l, -L$(PNETCDF_PATH_KAUST) -lpnetcdf -l</
↪append>
```

**Error:** /usr/bin/ld: cannot find -l,

## 18.28 Trial 072

Setting config\_compilers.xml:

```
<append>-L$(NETCDF_PATH_KAUST)/lib -lnetcdff, -L$(PNETCDF_PATH_KAUST)/lib -lpnetcdf</
↪append>
```

**Error:** /usr/bin/ld: cannot find -lnetcdff

## 18.29 Trial 071

Setting config\_compilers.xml:

```
<append>-L$(NETCDF_PATH_KAUST)/lib -lnetcdff -lnetcdf, -L$(PNETCDF_PATH_KAUST)/lib -
↪lpnetcdf</append>
```

..error:

/usr/bin/ld: cannot find -lnetcdf,

## 18.30 Trial 070

Setting config\_compilers.xml:

```
<append>-L$(NETCDF_PATH_KAUST) -lnetcdff -Wl, -L$(PNETCDF_PATH_KAUST) -lpnetcdf -Wl</append>
```

This builds, but we still get an error.

**Error:** 1: NetCDF: Attempt to use feature that was not turned on when netCDF was built.

Checking the cesm buildlog.

```
-lpnetcdf -Wl -mkl=cluster -L/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib -
lpnetcdf -mkl
ifort: command line warning #10157: ignoring option '-W'; argument is of wrong type
ifort: command line warning #10121: overriding '-mkl=cluster' with '-mkl'
```

## 18.31 Trial 069

Since the compiler so the flag has to match the name of the shared object file `-lpnetcdf` should work but not `-lpnetcdff`.

```
<SLIBS>
  <append> -I$(NETCDF_PATH_KAUST)/include, -I$(PNETCDF_PATH_KAUST)/include, -L$(NETCDF_
PATH_KAUST)/lib -lnetcdff -lnetcdf, -L$(PNETCDF_PATH_KAUST)/lib -lpnetcdf</append>
</SLIBS>
```

## 18.32 Trial 068

Attempting the same configuration as in Trial 067, except comment out `INC_PNETCDF` and `LIB_PNETCDF`. The cesm buildlog reads:

```
ifort: command line warning #10121: overriding '-mkl=cluster' with '-mkl'
/usr/bin/ld: cannot find -lpnetcdff
/usr/bin/ld: cannot find -lnetcdf,
/usr/bin/ld: cannot find -lpnetcdff
/usr/bin/shalsum: /lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.068.e03/bld/cesm.exe: No such file or directory
```

What are the names of the shared files?

```
/opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0/lib ls *so
libbz2_intel.so libmisc_intel.so libnetcdf_c++4_intel.so libnetcdff_intel.so
libnetcdf_intel.so
libbz2.so libmisc.so libnetcdf_c++4.so libnetcdff.so
libnetcdf.so
```

```
/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib ls
libpnetcdf.a      libpnetcdf_intel.so    libpnetcdf_intel.so.3.0.1  libpnetcdf.so.3
↳ pkgconfig
libpnetcdf_intel.a  libpnetcdf_intel.so.3  libpnetcdf.so              libpnetcdf.so.3.0.1
```

## 18.33 Trial 067

Setting config\_machines.xml:

```
<env name="PNETCDF_PATH_KAUST">/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0</env>
<env name="INC_PNETCDF_KAUST">/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/include</
↳ env>
<env name="LIB_PNETCDF_KAUST">/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib</env>
<SLIBS>
  <append> -L$(NETCDF_PATH_KAUST) -lnetcdff -Wl, -L$(PNETCDF_PATH_KAUST) -lpnetcdff -Wl,
↳ --as-needed, -L$(NETCDF_PATH_KAUST)/lib -lnetcdff -lnetcdf, -L
↳ $(PNETCDF_PATH_KAUST)/lib -lpnetcdff -lpnetcdf</append>
</SLIBS>
<MPICC> cc </MPICC>
<MPICXX> CC </MPICXX>
<MPIFC> ftn </MPIFC>
<PIO_FILESYSTEM_HINTS>lustre</PIO_FILESYSTEM_HINTS>
<NETCDF_PATH>${ENV{NETCDF_PATH_KAUST}}</NETCDF_PATH>
<PNETCDF_PATH>${ENV{PNETCDF_PATH_KAUST}}</PNETCDF_PATH>
<INC_PNETCDF>${ENV{INC_PNETCDF_KAUST}}</INC_PNETCDF>
<LIB_PNETCDF>${ENV{LIB_PNETCDF_KAUST}}</LIB_PNETCDF>
```

The PIO\_CONFIG\_ARGS sets the PNETCDF\_PATH argument:

```
<append> -L$(NETCDF_PATH_KAUST) -lnetcdff -Wl, -L$(PNETCDF_PATH_KAUST) -lpnetcdff -Wl, -L
↳ $(NETCDF_PATH_KAUST)/lib -lnetcdff -lnetcdf, -L$(PNETCDF_PATH_KAUST)/lib -lpnetcdff -
↳ lpnetcdf</append>
```

This errors out when running create\_newcase...

Schemas validity error : Element 'INC\_PNETCDF': This element **is not** expected.

## 18.34 Trial 066

This fails, too. What should we attempt next do?

```
<append> -L$(NETCDF_DIR) -lnetcdff -Wl, -L$(PNETCDF_DIR)/lib -lpnetcdff -lpnetcdf, --as-
↳ needed, -L$(NETCDF_DIR)/lib -lnetcdff -lnetcdf</append>
```

I guess the next thing to try is to toggle through different compiler flags. It might be faster to iterate by building DART rather than building CESM.

What does the the --as-needed flag accomplish?

## 18.35 Trial 065

Adding back in the link to `-lpnetcdf` in `config_compilers.xml`:

```
<SLIBS>
  <append> -L$(NETCDF_PATH) -lnetcdff -Wl, --as-needed, -L$(PNETCDF_PATH)/lib -lpnetcdff
</append>
</SLIBS>
```

This fails to build. Remember:

- The `NETCDF_PATH` has a library: `/opt/cray/pe/netcdf/4.6.3.2/INTEL/19.0/lib`
- The `PNETCDF_PATH` also has a library: `/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0/lib`

What to do next? I think the issue is it's linking `netcdf` rather than `pnetcdf`.

### 18.35.1 Spitballing ideas

The directory we're linking to is wrong.

- `-L` goes to the lib directory
- `-I` goes to the include directory

## 18.36 Trial 064

Removed this from `config_compilers.xml`:

```
<SLIBS>
<append> -L$(NETCDF_PATH) -lnetcdff -Wl, --as-needed, -L$(NETCDF_PATH)/lib -lnetcdff, -L
$(PNETCDF_PATH)/lib -lnetcdff</append>
</SLIBS>
```

#### Error:

```
1: pio_support::pio_die:: myrank=          -1 : ERROR: ionf_mod.F90:          235 :
1: NetCDF: Attempt to use feature that was not turned on when netCDF was built.
```

## 18.37 Trial 063

How do we tell CESM to trigger which MPI library to use? MPI Libs [http://www.cesm.ucar.edu/models/cesm1.2/cesm/doc\\_cesm1\\_2\\_1/modelnl/machines.html](http://www.cesm.ucar.edu/models/cesm1.2/cesm/doc_cesm1_2_1/modelnl/machines.html) `!mpi-serial /lustre/project/k1421/cases/FHIST_BGC.f09_d025.062.e03 .xmlquery MPILIB MPILIB: mpt https://bb.cgd.ucar.edu/cesm/threads/viability-of-running-cesm-on-40-cores.4997/page-2#post-37159` You should build and install `netcdf` and `pnetcdf` separately and link both. Tried to building with this one: `<append> -L$(NETCDF_PATH) -lnetcdff -Wl, --as-needed, -L$(NETCDF_PATH)/lib -lnetcdff, -L$(PNETCDF_PATH)/lib -lnetcdff</append>`

## 18.38 Trial 062

If that doesn't work, try linking to PNETCDF path in this line of `config_compilers.xml`:

```
<append> -L$(NETCDF_PATH) -lnetcdff -Wl, --as-needed, -L$(NETCDF_PATH)/lib -lnetcdff</append>
```

Okay that fixes the PNETCDF not enabled in the build issue.

Although now we have a different error.

### Error:

```
1: pio_support::pio_die:: myrank=          -1 : ERROR: ionf_mod.F90:          235 :
1:   NetCDF: Attempt to use feature that was not turned on when netCDF was built.
```

```
$ nc-config --all
...
--has-pnetcdf    -> no
...
```

You have to define an environmental variable that contains the path to the PNETCDF library in `config_machines.xml`.

And then reference that environmental variable when assinging a value to the PNETCDF\_PATH key in `config_compilers.xml` but I actually don't understand how the linker is made aware of that value, because the linker is only given the serial netcdf path.

## 18.39 Trial 061

I'm not sure why we're getting this error.

When trying to set:

```
$ ./xmlchange PIO_TYPENAME=pnetcdf
Did not find pnetcdf in valid values for PIO_TYPENAME: ['netcdf']
```

Examining `config_pio.xml` but I'm not sure how to interpret the results.

```
$ vim /lustre/project/k1421/cesm2_1_3/cime/config/cesm/machines/config_pio.xml
```

In `config_compilers.xml`:

```
<PNETCDF_PATH>${ENV{PARALLEL_NETCDF_DIR}}</PNETCDF_PATH>
```

The error is from Line 238 of `${CIMEROOT}/scripts/lib/CIME/XML/entry_id.py` in the function:

```
get_valid_value_string in "Did not find {} in valid values for {}: {}"
```

What does the entry for CNL look like?

```
<compiler OS="CNL">
  <CMAKE_OPTS>
```

(continues on next page)

(continued from previous page)

```

    <base> -DCMAKE_SYSTEM_NAME=Catamount</base>
  </CMAKE_OPTS>
  <CPPDEFS>
    <append> -DLINUX </append>
    <append MODEL="gpt1"> -DHAVE_NANOTIME -DBIT64 -DHAVE_VPRINTF -DHAVE_BACKTRACE -
    ↪ DHAVE_SLASHPROC -DHAVE_COMM_F2C -    DHAVE_TIMES -DHAVE_GETTIMEOFDAY </append>
  </CPPDEFS>
  <MPICC> cc </MPICC>
  <MPICXX> CC </MPICXX>
  <MPIFC> ftn </MPIFC>
  <NETCDF_PATH>$ENV{NETCDF_DIR}</NETCDF_PATH>
  <PIO_FILESYSTEM_HINTS>lustre</PIO_FILESYSTEM_HINTS>
  <PNETCDF_PATH>$ENV{PARALLEL_NETCDF_DIR}</PNETCDF_PATH>
  <SCC> cc </SCC>
  <SCXX> CC </SCXX>
  <SFC> ftn </SFC>
</compiler>

```

Edited the shaheen entry in config\_compilers.xml:

```

<PIO_FILESYSTEM_HINTS>lustre</PIO_FILESYSTEM_HINTS>
<PNETCDF_PATH>$ENV{PNETCDF_PATH}</PNETCDF_PATH>
<SCC> cc </SCC>

```

And added this to config\_machines.xml:

```

<env name="PNETCDF_PATH">/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0</env>

```

## 18.40 Trial 060

Using /Users/johnsonb/scratch/cheyenne/buildmml which is the same as the reanalysis, except it has had these lines inserted from cesm2\_1\_3:

```

docn_mode = case.get_value("DOCN_MODE")
if docn_mode and 'aqua' in docn_mode:
    config['aqua_planet_sst_type'] = docn_mode
else:
    config['aqua_planet_sst_type'] = 'none'

```

This crashes, with a warning.

**Warning:** PNETCDF not enabled in the build.

Is this warning present in Trial 059 as well?

```

$ vim /lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.059.e80/run/cesm.log.15958125.201019-
↪ 232044.gz
Pattern not found: PNETCDF not enabled in the build

```



According to this *DiscussCESM* post <<https://bb.cgd.ucar.edu/cesm/threads/using-pnetcdf-for-cesm.3084/>>, the message indicates that PNETCDF was not linked with the application.

## 18.41 Trial 059

Keeping OMP\_STACKSIZE and using the old buildnml script.

..code-block:

```
<env name="OMP_STACKSIZE">256M</env>
```

This runs properly.

## 18.42 Trial 058

Changing OMP\_STACKSIZE.

..code-block:

```
<env name="OMP_STACKSIZE">256M</env>
```

Also using Kevin Raeder's buildnml script modification.

## 18.43 Trial 057

Changing OMP\_STACKSIZE.

..code-block:

```
<env name="OMP_STACKSIZE">1024M</env>
```

Doesn't seem to affect performance.

## 18.44 Trial 056

Changing OMP\_STACKSIZE.

..code-block:

```
<env name="OMP_STACKSIZE">128M</env>
```

Doesn't seem to affect performance.

## 18.45 Trial 055

This run timed out with a super user message:

```
Error: run.FHIST_BGC.f09_d025.055.e80 Ended, Run time 01:00:26
```

It also sat in the queue for a really long time so maybe there is traffic on the interconnects.

## 18.46 Trial 054

Toggling settings in config-machines.xml:

```
<env name="MPI_COMM_MAX">16383</env>
<env name="MPI_GROUP_MAX">1024</env>
```

## 18.47 Trial 049

Added echo statements to assimilate.csh.

To see what a bash script to submit a SLURM job looks like:

```
vim /lustre/project/k1421/scripts_logs/run_check_input_data_SMS_Lm13.f10_f10_musgs.
↪ I1850CIm50SpG.shaheen_intel.20200612_215255_kuwhyp
```

```
#!/bin/bash
#
#SBATCH --job-name=run_check_input_data_SMS_Lm13.f10_f10_musgs.I1850CIm50SpG.shaheen_
↪ intel.20200612_215255_kuwhyp
#SBATCH --output=run_check_input_data_SMS_Lm13.f10_f10_musgs.I1850CIm50SpG.shaheen_intel.
↪ 20200612_215255_kuwhyp.txt
#SBATCH --partition=workq
#SBATCH --ntasks=1
#SBATCH --time=23:59:00
#SBATCH --mem-per-cpu=100
python run_check_input_data.py SMS_Lm13.f10_f10_musgs.I1850CIm50SpG.shaheen_intel.
↪ 20200612_215255_kuwhyp
```

Examining assimilate.csh, it's crashing when trying to execute this line:

```
setenv NODENAMES $SLURM_NODELIST
nid00[136-147]: No match.
```

Cross-referencing this with the da.log and the assimilate script:

```
vim /lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.049.e3/run/da.log.15905662.201015-
↪ 053628
vim /lustre/project/k1421/DART/models/cam-fv/shell_scripts/cesm2_1/assimilate.csh.
↪ template
```

```

inf_flavor(1) = 2, using namelist values.
[ Thu Oct 15 18:38:59 2020] [c0-0c0s9n0] Fatal error in MPI_Init: Other MPI error,
↳error stack:
MPIR_Init_thread(537):
MPID_Init(246).....: channel initialization failed
cray-netcdf/4.6.3.2(9):ERROR:150: Module 'cray-netcdf/4.6.3.2' conflicts with the
↳currently loaded module(s) 'cray-netcdf-hdf5parallel/4.6.3.2'
ncks: Command not found
Can't load parallel netcdf and nco at the same time.
MPI_COMM_MAX=16383

```

## 18.48 Trial 048

After differencing the SourceMod of `dyn_comp.F90` we're using and the one in `cesm2_1_3`, it might be worth trying to see if swapping out the SourceMod gets us to a different point in the compiling before crashing.

Removing `lustre/project/k1421/SourceMods/cesm2_1_3/SourceMods/src.cam/dyn_comp.F90`.

This works! The run doesn't complete, since `assimilate.csh` crashes, but this is clear progress. Now what the error log is printing:

## 18.49 Trial 047

**Note:** This is an important debugging trial because we fixed the PE issue in Trial 46 and can move onto determining why the build mysteriously hangs without producing any meaningful error messages.

We focus on determining which task happens right after `GSMAP indices not increasing`. Change the debug settings, with `./xmlchange DEBUG=TRUE` and `./xmlchange INFO_DEBUG=1`.

### 18.49.1 Last working startup case

The last working startup – not hybrid case is Trial 018:

`/lustre/project/k1421/cases/FHIST_BGC.f09_d025.018.e3`

Its buildlog continues past `GSMAP indices not increasing`:

```

MCT::m_Router::initp_: GSMAP indices not increasing...Will correct
0: calsize j,iq,jac, lsfrm,lstoo 1 1 1 26 21
Opened file FHIST_BGC.f09_d025.018.e3.cam_0001.r.1979-01-01-21600.nc to write

```

## 18.49.2 Attempting to clone Kevin’s CAM Reanalysis on Cheyenne

**Error:** CAM Could not be built cat /glade/scratch/johnsonb/FHIST\_BGC.f09\_d025.001.e3/bld/atm.bldlog.201012-214748

## 18.49.3 Attempting to clone Kevin’s CAM Reanalysis on Shaheen

Working on: /glade/work/johnsonb/DART\_Shaheen/models/cam-fv/shell\_scripts/cesm2\_1/setup\_advanced\_Rean Stage directory: /glade/scratch/johnsonb/archive/f.e21.FHIST\_BGC.f09\_025.CAM6assim.011/rest/2019-08-05-000000

gunzip f.e21.FHIST\_BGC.f09\_025.CAM6assim.011.cam\_00[45678]?r.2019-08-05-000000.nc.gz

### Rebuilding with cesm2.1.resl5.6 instead of cesm2\_1\_3

Two candidate source files:

1. ./components/cam/src/chemistry/modal\_aero/modal\_aero\_gasaerexch.F90
2. ./components/cam/src/chemistry/utils/modal\_aero\_calcsize.F90

This is the code that prints the relevant statement in the buildlog:

```
1016      write( 6, '(a,3i3,2i4)' ) 'calcsize j,iq,jac, lsfrm,lstoo', &
1017      j,iq,jac, lsfrm,lstoo
```

This file never gets called: ./components/cam/src/chemistry/utils/modal\_aero\_calcsize.F90

Thus it’s actually hanging in: ./components/cam/src/dynamics/fv/cd\_core.F90

We can attempt a case with a different resolution: FHIST\_BGC.f19\_f19\_mg17.001.e3.

However, using the build scripts don’t work because there isn’t a way to get the SST flux properly into the coupler.

```
source activate py27
cd /lustre/project/k1421/cesm2_1_3/cime/scripts
./create_newcase --case /lustre/project/k1421/cases/FHIST_BGC.f19_f19_mg17.002.e3 --
→machine shaheen --res f19_f19_mg17 --project k1421 --queue workq --walltime 1:00:00 --
→pecount 32x1 --ninst 3 --compset HIST_CAM60_CLM50%BGC-CROP_CICE%PRES_DOCN%DOM_MOSART_
→SGLC_SWAV --multi-driver --run-unsupported
cd /lustre/project/k1421/cases/FHIST_BGC.f19_f19_mg17.002.e3
./case.setup
./case.build
cd /lustre/project/k1421/cases/FHIST_BGC.f19_f19_mg17.002.e3
./case.submit -M begin,end
```

This gives us a “working” start up run with a f19\_f19\_mg17 grid. It’s useful because it provides two clues: where the cesm.log.\* fails and where the atm\_00??.log.\* fails.

## Examining the CESM Log

In a job submission that runs to completion, the CESM log continues past the GMap indices not increasing...Will correct line:

```
/lustre/scratch/x_johnsobk/archive/FHIST_BGC.f19_f19_mg17.002.e3/logs/cesm.log.15883098.
→201014-065036.gz
transitions from the MCT::m_Router to the calcsiz printouts immediately after it.
3226 0: MCT::m_Router::initp_ GMap indices not increasing...Will correct
3227 0: MCT::m_Router::initp_ RGSMap indices not increasing...Will correct
3228 0: MCT::m_Router::initp_ RGSMap indices not increasing...Will correct
3229 0: MCT::m_Router::initp_ GMap indices not increasing...Will correct
3230 64: MCT::m_Router::initp_ GMap indices not increasing...Will correct
3231 64: MCT::m_Router::initp_ RGSMap indices not increasing...Will correct
3232 64: MCT::m_Router::initp_ RGSMap indices not increasing...Will correct
3233 64: MCT::m_Router::initp_ GMap indices not increasing...Will correct
3234 32: MCT::m_Router::initp_ GMap indices not increasing...Will correct
3235 32: MCT::m_Router::initp_ RGSMap indices not increasing...Will correct
3236 32: MCT::m_Router::initp_ RGSMap indices not increasing...Will correct
3237 32: MCT::m_Router::initp_ GMap indices not increasing...Will correct
3238 0: calcsiz j,iq,jac, lsfrm,lstoo 1 1 1 26 21
3239 0: calcsiz j,iq,jac, lsfrm,lstoo 1 1 2 26 21
3240 0: calcsiz j,iq,jac, lsfrm,lstoo 1 2 1 22 15
3241 0: calcsiz j,iq,jac, lsfrm,lstoo 1 2 2 22 15
3242 0: calcsiz j,iq,jac, lsfrm,lstoo 1 3 1 24 17
```

The MCT::m\_router lines are printed from the subroutine initp\_\_(inGMap,inRGSMap,mycomm,Rout,name ) in m\_Router.F90:

```
336 if(myPid == 0) call warn(myname_, 'GMap indices not increasing...Will correct')
337 call GlobalSegMap_OPoints(inGMap,myPid,gpoints)
```

Note well, in mct\_mod.F90, m\_router undergoes association renaming:

```
use m_Router, only: mct_router => Router
```

Additionally, the calcsiz lines are from modal\_aero\_calcsiz.F90:

```
1016 write( 6, '(a,3i3,2i4)' ) 'calcsiz j,iq,jac, lsfrm,lstoo', &
1017 j,iq,jac, lsfrm,lstoo
```

Thus the log from a non-working run, /lustre/scratch/x\_johnsobk/FHIST\_BGC.f09\_d025.047.e3/run/atm\_0001.log.15874648.201013-170022, ends here:

```
4619 FV subcycling - nv, n2, nsplit, dt = 2 1 4
4620 225.00000000000000
```

Line 4620 is printed from ./components/cam/src/dynamics/fv/dyn\_comp.F90:

```
1443 write(iulog,*) 'FV subcycling - nv, n2, nsplit, dt = ', nv, n2, nsplit, dt
```

The working log, /lustre/scratch/x\_johnsobk/archive/FHIST\_BGC.f19\_f19\_mg17.002.e3/logs/atm\_0001.log.15883098.201014-065036.gz, continues:

```
4993 FV subcycling - nv, n2, nsplit, dt =          1          1          4
4994    450.00000000000000
4995 Divergence damping: use 4th order damping
```

Line 4995 is printed from ./components/cam/src/dynamics/fv/cd\_core.F90:

```
545 if (masterproc) write(iulog,*) 'Divergence damping: use 4th order damping'
```

So the key is to determine what happens in between Line 1443 of dyn\_comp.F90 and the invocation of cd\_core which is called only once on line 1862:

```
1862          call cd_core(grid, nx, u, v, pt,
```

## 18.50 Trial 046

Edited config\_machines.xml to:

```
<!-- <MAX_TASKS_PER_NODE>64</MAX_TASKS_PER_NODE> -->
<MAX_TASKS_PER_NODE>32</MAX_TASKS_PER_NODE>
```

---

**Important:** This fixes the PE crash and we now hang at a different point.

---

## 18.51 Trial 045

Forgot to build Trial 044 with OMP\_STACKSIZE=256MB.

Edited config\_machines.xml to set OMP\_STACKSIZE=256MB. Rebuilt the case.

## 18.52 Trial 044

```
./xmlchange ROOTPE_ESP=0,NTHRDS_ESP=$nthreads,NTASKS_ESP=1
```

## 18.53 Trial 043

```
./xmlchange NTASKS_PER_INST_ESP=1
```

For some reason that still results in:

```
NTASKS_PER_INST: ['ATM:128', 'LND:128', 'ICE:128', 'OCN:128', 'ROF:128', 'GLC:128',
'WAV:128', 'ESP:32']
```

## 18.54 Trial 042

```
./xmlchange OMP_STACKSIZE=128MB
```

## 18.55 Trial 041

Attempting to change the stack limit to see if it noticeably affects performance.

```
./xmlchange OMP_STACKSIZE=1024MB
```

## 18.56 Trial 040

```
128: Reading zbgc_nml
0: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
0: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
0: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
0: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
128: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
256: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
128: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
256: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
256: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
128: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
256: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
128: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
361: forrtl: severe (174): SIGSEGV, segmentation fault occurred
128: MCT::m_Router::initp_: RGSMMap indices not increasing...Will correct
256: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
9583 srun: error: nid04049: tasks 37-38,40-42,44,48,50-51,54-55,57,61-63: Exited with
↳ exit code 174
9584 srun: Terminating job step 15810768.0
9585 0: slurmstepd: error: *** STEP 15810768.0 ON nid04048 CANCELLED AT 2020-10-
↳ 08T21:09:34 ***
```

## 18.57 Trial 039

Seems like we should just try this attempt again in case the error was caused by running 3 the script times simultaneously.

## 18.58 Trial 038

Changing `set do_clm_interp = "true"`.

This works! However we should go through the CESM logs to see if it's hanging anywhere.

## 18.59 Trial 037

Omitted `modules.csh`.

Questions:

1. `PIO_TYPENAME = 'pnetcdf'`; Do we need to change it to `netcdf`?
2. `set do_clm_interp = "false"`; Do we need to change it to `true`?

Another issue:

```
/usr/bin/cp: option '--v' is ambiguous; possibilities: '--verbose' '--version'
Try '/usr/bin/cp --help' for more information.
```

Edited `DART_config` to omit `update_dart_namelist`s.

Also copied a file from GLADE:

```
Fetching /glade/p/cesmdata/cseg/inputdata/atm/cam/tracer_cnst/tracer_cnst_halons_WACCM6_
↪ 3Dmonthly_L70_1975-2014_c180216.nc to tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-
↪ 2014_c180216.nc
lustre/project/k1421/cesm_store/inputdata/atm/cam/tracer_cnst
```

This gives us an error.

### Error:

```
Did you mean to set use_init_interp = .true. in user_nl_clm?
94: (Setting use_init_interp = .true. is needed when doing a
94: transient run using an initial conditions file from a non-transient run,
94: or a non-transient run using an initial conditions file from a transient run,
94: or when running a resolution or configuration that differs from the initial_
↪ conditions.
```



## 18.60 Trial 036

Changed NTASKS for ESP=1 and set PIO\_TYPENAME=netcdf

But we still have the same SIGTERM failure. GRRR.

At least it's good to report that the Invalid PIO rearranger issue even occurs in a working run:

```
$ vim /lustre/scratch/x_johnsobk/archive/FHIST_BGC.f09_g17.002.e3/logs/cesm.log.15594937.
↪200910-181624.gz
0: Invalid PIO rearranger comm max pend req (comp2io),          0
0: Resetting PIO rearranger comm max pend req (comp2io) to      64
0: PIO rearranger options:
0:   comm type =
```

Working on contrasting these two runs, since one works and the other doesn't

```
$ cd /lustre/project/k1421/cases/FHIST_BGC.f09_g17.002.e3
$ ./xmlquery --partial PE
RUN_TYPE: startup
$ cd /lustre/project/k1421/cases/FHIST_BGC.f09_d025.036.e3
$ ./xmlquery --partial PE
RUN_TYPE: hybrid
```

The output of these are identical except for RUN\_TYPE.

There are two plausible paths:

1. Go through Kevin's script here: /lustre/project/k1421/DART\_CASES/setup\_advanced\_Rean.original and see if we're missing anything significant. It seems like the stagedir is just the full path where the restart files are. I think that might be it.

In line 1295 of setup\_advanced\_Rean.original

```
1295 ${LINK} -f ${stagedir}/${refcase}.clm2\${inst_string}.r.${init_time}.nc .
./xmlchange RUN_REFDIR=${stagedir}
```

2. Or get Kevin's setup script working, which might actually only entail changing the user\_nl text. This might actually be pretty fast.

```
360 set user_grid = "${user_grid} --gridfile /glade/work/raeder/Models/CAM_init/SST"
361 set user_grid = "${user_grid}/config_grids+fv1+2deg_oi0.25_gland20.xml"
362 setenv sst_dataset \
363   "/glade/work/raeder/Models/CAM_init/SST/avhrr-only-v2.20110101_cat_20111231_
↪gregorian_c190703.nc"
```

We should only need to change these, right?

```
340 set use_tasks_per_node = 36
...
959 set cesm_data_dir = "/glade/p/cesmdata/cseg/inputdata/atm"
960 set cesm_chem_dir = "/gpfs/fs1/p/acom/acom-climate/cmip6inputs/emissions_ssp119
↪"
961 set chem_root      = "${cesm_chem_dir}/emissions-cmip6-ScenarioMIP_IAMC-IMAGE-
↪ssp119-1-1"
962 set chem_dates     = "175001-210012_0.9x1.25_c20181024"
```

## 18.61 Trial 035

Set `tasks_per_node=16` but we still get the segmentation fault. So this might be a PE layout issue. We should really be trying to track down what the correct layout was for the last working ensemble.

### Warning:

```
Warning: missing non-idmap ROF2OCN_LIQ_RMAPNAME for ocn_grid, d.25x.25 and rof_grid_
↪r05
Warning: missing non-idmap ROF2OCN_ICE_RMAPNAME for ocn_grid, d.25x.25 and rof_grid_
↪r05
```

Looking back to `/lustre/project/k1421/cases/FHIST_BGC.f09_g17.002.e3` we see:

```
NTASKS_PER_INST: ['ATM:128', 'LND:128', 'ICE:128', 'OCN:128', 'ROF:128', 'GLC:128',
↪ 'WAV:128', 'ESP:1']
ROOTPE: ['CPL:0', 'ATM:0', 'LND:0', 'ICE:0', 'OCN:0', 'ROF:0', 'GLC:0', 'WAV:0', 'ESP:0']

Results in group mach_pes_last
  COSTPES_PER_NODE: 32
  COST_PES: 384
  MAX_MPITASKS_PER_NODE: 32
  MAX_TASKS_PER_NODE: 64
  TOTALPES: 384
```

I checked to see `run_domain` is the same in the working case and the non-working case.

```
Results in group run_domain
  ATM2LND_FMAPTYPE: X
  ATM2LND_SMAPTYPE: X
  ATM2OCN_FMAPTYPE: X
  ATM2OCN_SMAPTYPE: X
  ATM2OCN_VMAPTYPE: X
  ATM2WAV_SMAPTYPE: Y
  GLC2ICE_RMAPTYPE: Y
  GLC2LND_FMAPTYPE: Y
  GLC2LND_SMAPTYPE: Y
  GLC2OCN_ICE_RMAPTYPE: Y
  GLC2OCN_LIQ_RMAPTYPE: Y
  ICE2WAV_SMAPTYPE: Y
  LND2ATM_FMAPTYPE: Y
  LND2ATM_SMAPTYPE: Y
  LND2GLC_FMAPTYPE: X
  LND2GLC_SMAPTYPE: X
  LND2ROF_FMAPTYPE: X
  OCN2ATM_FMAPTYPE: Y
  OCN2ATM_SMAPTYPE: Y
  OCN2WAV_SMAPTYPE: Y
  ROF2LND_FMAPTYPE: Y
  ROF2OCN_FMAPTYPE: Y
  ROF2OCN_ICE_RMAPTYPE: Y
  ROF2OCN_LIQ_RMAPTYPE: Y
```

(continues on next page)

(continued from previous page)

```

WAV2OCN_SMAPTYPE: X

Results in group mach_pes
  NTASKS_PER_INST: ['ATM:128', 'LND:128', 'ICE:128', 'OCN:128', 'ROF:128', 'GLC:128'
↳ ', 'WAV:128', 'ESP:1']
  ROOTPE: ['CPL:0', 'ATM:0', 'LND:0', 'ICE:0', 'OCN:0', 'ROF:0', 'GLC:0', 'WAV:0',
↳ 'ESP:0']

Results in group mach_pes_last
  COSTPES_PER_NODE: 32
  COST_PES: 384
  MAX_MPITASKS_PER_NODE: 32
  MAX_TASKS_PER_NODE: 64
  TOTALPES: 384

PIO_NETCDF_FORMAT: ['CPL:64bit_offset', 'ATM:64bit_offset', 'LND:64bit_offset',
↳ 'ICE:64bit_offset', 'OCN:64bit_offset', 'ROF:64bit_offset', 'GLC:64bit_offset',
↳ 'WAV:64bit_offset', 'ESP:64bit_offset']

```

## 18.62 Trial 034

In Trial 034 we don't get the problem encountered in Trial 031 after changing PIO\_TYPENAME to netcdf. However, we still get an error.

### Error:

```

96: MCT::m_Router::initp_: RGSM indices not increasing...Will correct
96: MCT::m_Router::initp_: GSMap indices not increasing...Will correct
52: forrtl: severe (174): SIGSEGV, segmentation fault occurred

```

## 18.63 Trial 033

We should build Trial 033 with as a hybrid run and comment out PIO\_TYPENAME.

```

$ cd /lustre/project/k1421/cases/FHIST_BGC.f09_d025.033.e3/
$ cp /lustre/project/k1421/DART/models/cam-fv/shell_scripts/cesm2_1/assimilate.csh.
↳ template assimilate.csh
$ ./xmlchange DATA_ASSIMILATION_SCRIPT=./assimilate.csh
$ ./case.submit -M begin,end

```

The two candidate settings for toggling are RUN\_TYPE AND PIO\_TYPENAME.

Check the PIO buildlog for Trials 33 and 32.

```

Building pio with output to file /lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.033.e3/
↳ bld/pio.bldlog.201007-210614

```

Here's what the config\_machines.xml modules look like:

```
<modules mpilib="mpi-serial">
  <command name="load">cray-hdf5/1.10.5.2</command>
  <command name="load">cray-netcdf/4.6.3.2</command>
</modules>
<modules mpilib="!mpi-serial">
  <command name="load">cray-netcdf-hdf5parallel/4.6.3.2</command>
  <command name="load">cray-hdf5-parallel/1.10.5.2</command>
  <command name="load">cray-parallel-netcdf/1.11.1.1</command>
</modules>
```

Maybe we shouldn't use the restart files for ice\_ic?

## 18.64 Trial 032

Attempting to use a continue run instead of a hybrid run.

### Error:

```
Did not find continue in valid values for RUN_TYPE: ['startup', 'hybrid', 'branch']
RUN_TYPE = 'continue'
```

Aha, continue was permitted in CESM1 but isn't permitted anymore, so this doesn't work.

We can try a startup run next. Alternatively it could be a PIO\_TYPENAME issue?

## 18.65 Trial 031

Attempted to only change NTASKS\_PER\_INST\_ATM.

Okay this is progress as it results in an error.

### Error:

```
130: aborting in ice-pio_ropen with invalid file
130: ERROR: aborting in ice-pio_ropen with invalid file
```

Is the abort trap signal related to [this post](#)?

In user\_nl\_cice, and the variable is "ice\_ic". In Kevin's directory, /glade/work/raeder/Exp/f.e21.FHIST\_BGC.f09\_025.CAM6assim.011, it is set as ice\_ic = 'Rean\_spinup\_2010.cice\_0001.r.2011-01-01-00000.nc'

Hmm...are these files different resolution? Do they need to be both on the atmospheric grid, and not the oceanic grid?

This could be a number of different things:

1. The initial condition specified incorrectly? Or is the grid specified incorrectly?
2. What if we just omit ice\_ic and see what happens.

This wasn't a problem before because we were doing a startup run instead of a hybrid run.

## 18.66 Trial 030

Setting: `use_tasks_per_node = 16`

Which results in this result for our case on Shaheen.

```
$ cd ${CASEROOT}
$ ./xmlquery NTASKS
NTASKS: ['CPL:108', 'ATM:108', 'LND:108', 'ICE:108', 'OCN:108', 'ROF:108', 'GLC:108',
↪ 'WAV:108', 'ESP:36']
```

Comparing it to Kevin's configuration for the reanalysis on Cheyenne.

```
$ cd /glade/work/raeder/Exp/f.e21.FHIST_BGC.f09_025.CAM6assim.011
$ ./xmlquery NTASKS_PER_INST
NTASKS_PER_INST: ['ATM:108', 'LND:108', 'ICE:108', 'OCN:108', 'ROF:108', 'GLC:108',
↪ 'WAV:108', 'ESP:36']
```

So this seems to be the key thing we're missing.

## 18.67 Trial 029

Trying different nodes per instance: `tasks_per_node = 32` and `nodes_per_instance = 4`

```
cd /lustre/project/k1421/cases/FHIST_BGC.f09_d025.031.e3/
cp /lustre/project/k1421/DART/models/cam-fv/shell_scripts/cesm2_1/assimilate.csh.
↪template assimilate.csh
./xmlchange DATA_ASSIMILATION_SCRIPT=/lustre/project/k1421/cases/FHIST_BGC.f09_d025.031.
↪e3/assimilate.csh
./case.submit -M begin,end
```



## **CIME\_COMP\_MOD.F90**

In the process of trying to get more fine-grained timing information from the `cpl_*.log*`





## CONFIG\_MACHINES.XML

### 20.1 Tasks per node

The Cray XC40 is comprised of dual-socket nodes with Haswell processors that have 16 cores per socket (thus 32 cores per node).

There seems to be a nuance here, in that the Haswell CPUs each have two ThreadsPerCore so the MAX\_TASKS\_PER\_NODE should be twice the MAX\_MPITASKS\_PER\_NODE.

```
<MAX_TASKS_PER_NODE>64</MAX_TASKS_PER_NODE>
<MAX_MPITASKS_PER_NODE>32</MAX_MPITASKS_PER_NODE>
```

This is different from the configuration stated in `config_machines.xml` for Cheyenne, which also has dual-socket nodes but with Broadwell processors that have 18 cores per socket (thus 36 cores per node). However, for Cheyenne MAX\_TASKS\_PER\_NODE is set to the same value as MAX\_MPITASKS\_PER\_NODE:

```
<MAX_TASKS_PER_NODE>36</MAX_TASKS_PER_NODE>
<MAX_MPITASKS_PER_NODE>36</MAX_MPITASKS_PER_NODE>
```

### 20.2 One question

The `cpl` logs comparing an 80 member ensemble run on Cheyenne versus an 80 member run on Shaheen show that the Cheyenne ensemble is using much more memory (an order of magnitude greater at peak). Why?

We ran tests trying to see if it's because the `srun` script mistakenly doesn't request all the memory on a node. That doesn't seem to be the case. These timing files are from two jobs, the first with the default configuration and the second with the `#SBATCH --mem=0` directive present in the job script:

```
/lustre/project/k1421/cases/FHIST_BGC.f09_d025.097.e80/timing/cesm_timing_0080.FHIST_BGC.
↪f09_d025.097.e80.17866538.201223-080254
/lustre/project/k1421/cases/FHIST_BGC.f09_d025.097.e80/timing/cesm_timing_0080.FHIST_BGC.
↪f09_d025.097.e80.17866547.201223-085717
```

The Init Time is pretty comparable and the second job actually uses less peak memory.

Should we try to fiddle with the `netCDF` libraries again? Brian Dobbins said that makes the biggest difference with respect to performance.

## 20.3 mpirun Settings

After consulting with Bilel Hadri, we came upon this setting within `config_machines.xml`:

```
<mpirun mpilib="default">
  <executable>srun</executable>
  <arguments>
    <arg name="label"> --label</arg>
    <arg name="num_tasks" > --hint=nomultithread -n {{ total_tasks }}</arg>
    <arg name="binding" > --cpu_bind=cores</arg>
  </arguments>
</mpirun>
```

The settings copied from the `cori-haswell` entry are:

```
<mpirun mpilib="default">
  <executable>srun</executable>
  <arguments>
    <arg name="label"> --label</arg>
    <arg name="num_tasks" > -n {{ total_tasks }}</arg>
    <arg name="binding"> -c {{ srun_binding }}</arg>
  </arguments>
</mpirun>
```

## CPUS-PER-TASK

### 21.1 Overview

One of the ongoing issues is trying to understand how to handle the ambiguity with respect to how a CPU is defined. The Crays have 2 sockets and each processor occupying the socket has 16 cores, but invoking `scontrol show nodes` will print out the node configuration of Shaheen's Cray XC40 nodes:

```
$ scontrol show nodes
NodeName=nid07679 Arch=x86_64 CoresPerSocket=16
CPUAlloc=64 CPUTot=64 CPULoad=64.00
[...]
```

The configuration considers each core to have 2 threads. Thus each node has 32 CPUs per socket and 64 CPUs per node.

In the Shaheen entry of the `config_machines.xml` file, we copy values defined in the Cori Haswell entry and set `MAX_TASKS_PER_NODE` and `MAX_MPI_TASKS_PER_NODE` as follows:

```
<MAX_TASKS_PER_NODE>32</MAX_TASKS_PER_NODE>
<MAX_MPITASKS_PER_NODE>32</MAX_MPITASKS_PER_NODE>
```

From this [DiscussCESM post](#), we know that on Cori Haswell the `srun` command is invoked with the `--cpus-per-task 2` option, which, according to Brian Dobbins instructs the node not to use hyperthreading.

This documents our attempt to build two cases, the first in which `srun` is given the `--cpus-per-task 2` option and the second in which `srun` isn't given the option.

### 21.2 Parallel netCDF library

In this attempt we use the `cray-parallel-netcdf` library.

```
<modules mpilib="!mpi-serial">
  <command name="load">cray-parallel-netcdf/1.11.1.1</command>
</modules>
<environment_variables>
  <env name="PNETCDF_PATH">/opt/cray/pe/parallel-netcdf/1.11.1.1/INTEL/19.0</env>
</environment_variables>
```

This library is the one used in `cime/scripts/Tools/Makefile`:

```
ifdef LIB_PNETCDF
  CPPDEFS += -D_PNETCDF
  SLIBS += -L$(LIB_PNETCDF) -lpnetcdf
endif
```

## 21.3 Attempt to spoof a Cori Haswell Run

```
$ source activate py27
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts/
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.cori-haswell.f09_f09_mg17.
↪001 --compset FHIST --res f09_f09_mg17 --machine cori-haswell --project k1421 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
[...]
$ cd /lustre/project/k1421/cases/FHIST.cori-haswell.f09_f09_mg17.001
$ ./case.setup
ERROR: inputdata root is not a directory or is not readable: /project/projectdirs/ccsm1/
↪inputdata
```

We need to change the paths in `config_machines.xml`

```
<machine MACH="cori-haswell"> <!-- 2021-01-14 BKJ: Changing the Directories for Cori to see if I can execute
./preview_run on Shaheen--> <!--<DIN_LOC_ROOT>/project/projectdirs/ccsm1/inputdata</DIN_LOC_ROOT>
<DIN_LOC_ROOT_CLMFORC>/project/projectdirs/ccsm1/inputdata/atm/dtm7</DIN_LOC_ROOT_CLMFORC>
<DOUT_S_ROOT>${CIME_OUTPUT_ROOT}/archive/${CASE}</DOUT_S_ROOT> <BASE-
LINE_ROOT>/project/projectdirs/ccsm1/ccsm_baselines</BASELINE_ROOT> <CCSM_CPRNC>/project/projectdirs/ccsm1/tools/cprnc/
<DIN_LOC_ROOT>/lustre/project/k1421/cesm_store/inputdata</DIN_LOC_ROOT>
<DIN_LOC_ROOT_CLMFORC>/lustre/project/k1421/cesm_store/inputdata/atm/dtm7</DIN_LOC_ROOT_CLMFORC>
<DOUT_S_ROOT>${CIME_OUTPUT_ROOT}/archive/${CASE}</DOUT_S_ROOT> <BASE-
LINE_ROOT>/lustre/project/k1421/cesm_store/baselines</BASELINE_ROOT> <CCSM_CPRNC>/lustre/project/k1421/CESM/cime/
</machine>
```

```
$ source activate py27
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts/
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.cori-haswell.f09_f09_mg17.
↪002 --compset FHIST --res f09_f09_mg17 --machine cori-haswell --project k1421 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
[...]
$ cd /lustre/project/k1421/cases/FHIST.cori-haswell.f09_f09_mg17.002
$ ./case.setup
ERROR: inputdata root is not a directory or is not readable: /project/projectdirs/ccsm1/
↪inputdata
```

## 21.4 Trial 1: srun isn't given the `--cpus-per-task 2` option

\$ source activate py27

```
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts/ $ ./create_newcase --case
/lustre/project/k1421/cases/FHIST.cpus-per-task.f09_f09_mg17.001 --compset FHIST --res f09_f09_mg17
--machine shaheen --project k1421 --run-unsupported --ninst 3 --multi-driver --walltime 2:00:00 [...]
Creating Case directory /lustre/project/k1421/cases/FHIST.cpus-per-task.f09_f09_mg17.001 $ cd
/lustre/project/k1421/cases/FHIST.cpus-per-task.f09_f09_mg17.001 $ ./case.setup $ ./preview_run [...] srun
--label -n 384 --cpus-per-task 2 /lustre/scratch/x_johnsobk/FHIST.cpus-per-task.f09_f09_mg17.002/bld/cesm.exe
>> cesm.log.$LID 2>&1 $ ./case.build [...]
```

## 21.5 Trial 2: srun is given the `--cpus-per-task 2` option

We edit `/lustre/project/k1421/cesm2_1_3/cime/config/cesm/machines/config_machines.xml` to use `--cpus-per-task 2`.

```
<mpirun mpilib="default">
  <executable>srun</executable>
  <arguments>
    <!-- Default arguments from Cori-Haswell-->
    <arg name="label"> --label</arg>
    <arg name="num_tasks" > -n {{ total_tasks }}</arg>
    <!--<arg name="binding"> -c {{ srun_binding }}</arg-->
    <arg name="binding"> --cpus-per-task 2</arg>
  </arguments>
</mpirun>
```

We then build the case.

```
$ source activate py27
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts/
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.cpus-per-task.f09_f09_mg17.
↪002 --compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
[...]
Creating Case directory /lustre/project/k1421/cases/FHIST.cpus-per-task.f09_f09_mg17.002
$ cd /lustre/project/k1421/cases/FHIST.cpus-per-task.f09_f09_mg17.002
$ ./case.setup
$ ./preview_run
[...]
srun --label -n 384 --cpus-per-task 2 /lustre/scratch/x_johnsobk/FHIST.cpus-per-task.
↪f09_f09_mg17.002/bld/cesm.exe >> cesm.log.$LID 2>&1
$ ./case.build
[...]
Time spent not building: 14.555864 sec
Time spent building: 606.567624 sec
MODEL BUILD HAS FINISHED SUCCESSFULLY
```

**Note:** Can we just build a case and tell `.create_newcase` that we

Third case:

`<arg name="binding"> -cpus-per-task 2</arg> AND`

## SCIENTIFICALLY SUPPORTED CONFIGURATION

### 22.1 Overview

As discussed in the [Grid and compset](#) page, the scientifically supported configuration used in the experiment is the FHIST compset and the f09\_f09\_mg17 grid.

### 22.2 Scientifically supported grid

The scientifically supported grid for the FHIST compset is the f09\_f09\_mg17 grid. For more information, see the [CAM documentation](#). Efficiency testing should be done with this grid before moving on to the CAM6 Reanalysis grid.

#### 22.2.1 Cori

---

**Note:** On Cori the default version of python is 2.7.18. Thus the multi-instance cesm2.1.3 build script doesn't crash because of attempted concatenation of a string and an int. There is no need to create a python 2.7 environment.

---

```
$ cd /project/projectdirs/m3857/cesm2_1_3/cime/scripts
$ ./create_newcase --case /project/projectdirs/m3857/cases/FHIST.f09_f09_mg17.e001.n0003
↪--compset FHIST --res f09_f09_mg17 --machine cori-haswell --project m3857 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
[ ... ]
Creating Case directory /project/projectdirs/m3857/cases/FHIST.f09_f09_mg17.e001.n0003
$ cd /project/projectdirs/m3857/cases/FHIST.f09_f09_mg17.e001.n0003
$ ./case.setup
$ ./case.build
[ ... ]
MODEL BUILD HAS FINISHED SUCCESSFULLY
$ ./case.submit -M begin,end
```

## 22.2.2 Shaheen

**Caution:** On Shaheen the default version of python is 3.7.7. Thus the multi-instance cesm2.1.3 build script crashes because of attempted concatenation of a string and an int (which isn't allowed in python 3). It was necessary to create a python 2.7 environment using conda. It is activated using the `source activate py27` line.

Additionally, the LibXML2 library is missing from most of the login nodes. Thus it is necessary to build on cdl5.

```
$ ssh cdl5
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts
$ source activate py27
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e001.n0003 --
↪compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --
↪ninst 3 --multi-driver --walltime 2:00:00
[ ... ]
Creating Case directory /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e001.n0003
$ cd /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e001.n0003
$ ./case.setup
$ ./case.build
[ ... ]
MODEL BUILD HAS FINISHED SUCCESSFULLY
$ ./case.submit -M begin,end
```

## 22.2.3 Cheyenne

**Note:** On Cheyenne the default version of python is 2.7.16. Thus the multi-instance cesm2.1.3 build script doesn't crash because of attempted concatenation of a string and an int. There is no need to create a python 2.7 environment.

**Warning:** Running a build of CESM on a Cheyenne login node will likely result in one of the daemons killing the build script due to the build "using an excessive amount of the total available CPU resources." Thus use the `qcmd` build wrapper and options for the `./case.build` script.

```
$ cd /glade/work/johnsonb/cesm2_1_3/cime/scripts
$ ./create_newcase --case /glade/work/johnsonb/cases/FHIST.f09_f09_mg17.e001.n0003 --
↪compset FHIST --res f09_f09_mg17 --machine cheyenne --project P86850054 --run-
↪unsupported --ninst 3 --multi-driver --walltime 2:00:00
[ ... ]
Creating Case directory /glade/work/johnsonb/cases/FHIST.f09_f09_mg17.e001.n0003
$ cd /glade/work/johnsonb/cases/FHIST.f09_f09_mg17.e001.n0003
$ ./case.setup
$ qcmd -q share -l select=1 -A P86850054 -- ./case.build
[ ... ]
MODEL BUILD HAS FINISHED SUCCESSFULLY
$ ./case.submit -M begin,end
```



## 22.3 Custom Grid

The port on Shaheen II uses the same grid as the [DART CAM6 Reanalysis](#), which is a custom grid created by Kevin Raeder, called f09\_d025.

The grid specification is available on GLADE here:

```
/glade/work/raeder/Models/CAM_init/SST/config_grids+fv1+2deg_oi0.25_gland20.xml
```

The grid has quarter-degree horizontal resolution, so that CAM can be forced with the [AVHRR quarter-degree SST product](#).

## 22.4 Compset

The integration uses an F with biogeochemistry. Kevin's CASEROOT on GLADE and his README.case are:

```
/glade/work/raeder/Exp/f.e21.FHIST_BGC.f09_025.CAM6assim.011/README.case
```

The create\_newcase command invoked is as follows:

```
/glade/work/raeder/Models/cesm2_1_relsd_m5.6/cime/scripts/create_newcase --case /glade/
↪work/raeder/Exp/f.e21.FHIST_BGC.f09_025.CAM6assim.011 --machine cheyenne --res f09_
↪d025 --project NCIS0006 --queue premium --walltime 1:00 --pecount 36x1 --ninst 80 --
↪compset HIST_CAM60_CLM50%BGC-CROP_CICE%PRES_DOCN%DOM_MOSART_SGLC_SWAV --run-
↪unsupported --multi-driver --gridfile /glade/work/raeder/Models/CAM_init/SST/config_
↪grids+fv1+2deg_oi0.25_gland20.xml
```



## RUN TYPE

I can't find the run type documentation for CESM2, but the analagous documentation from CESM1 can be found on this [Running CESM](#) page.

---

**Note:** There was some confusion on my part about whether the KiloCAM experiment should be a branch or a hybrid run. To extend the Reanalysis from 2019 to 2020, Kevin set up a branch run because it allowed him to change the SST forcing for the reanalysis while, at the same time, using restart files. For the KiloCAM experiment, we require using `perturb_single_instance` to generate the thousand members, which only works on initial files rather than restart files, so it should be set up as a hybrid run.

---

### 23.1 Initial files

The initial files from the CAM6 reanalysis are here:

```
/lustre/project/k1421/Reanalyses/f.e21.FHIST_BGC.f09_025.CAM6assim.011/rest/2018-01-01-000000
```

The individual filenames are of the form:

```
f.e21.FHIST_BGC.f09_025.CAM6assim.011.cam_00??i.2018-01-01-00000.nc
```

### 23.2 Using `perturb_single_instance`

The `&perturb_single_instance_nml` namelist in `input.nml` has this structure:

```
&perturb_single_instance_nml
  ens_size           = 3
  input_files        = 'caminput.nc'
  output_files       = 'cam_pert1.nc', 'cam_pert2.nc', 'cam_pert3.nc'
  output_file_list   = ''
  perturbation_amplitude = 0.2
/
```

The simplest way to generate the initial ensemble is to use a symbolic link for `caminput.nc` and write the names of the desired perturbed initial files into the file declared in `output_file_list`.



## TROUBLESHOOTING CIME

The testing suite for the Common Infrastructure for Modeling the Earth (CIME) doesn't seem to comprehensively cover the use cases we've been conducting. Thus there are many instances in which builds and case submissions fail for reasons that are seemingly perplexing. This document explains a subset of these failures.

### 24.1 input\_data.checksum.dat

**Error:**

```
Trying to download file: '../inputdata_checksum.dat' to path [...]run/inputdata_
↪checksum.dat.raw'
```

This error seems to be caused by the configuration of the wget address path (which gets called since the globus-url-copy tool isn't installed on either Cori or Shaheen II) in \$CIMEROOT/config/cesm/config\_inputdata.xml:

```
<server>
  <protocol>wget</protocol>
  <address>ftp://ftp.cgd.ucar.edu/cesm/inputdata</address>
  <user>anonymous</user>
  <password>user@example.edu</password>
  <checksum>../inputdata_checksum.dat</checksum>
</server>
```

The address path should be appended with a trailing slash `ftp://ftp.cgd.ucar.edu/cesm/inputdata/` since the checksum path starts with `../inputdata_checksum.dat`.



## MISSING XML LIBRARY

When Shaheen II underwent maintenance on two separate occasions – the 22 June 2021 and the 25 January 2022 upgrades – the LibXML library went missing.

This library is used by a pair of perl scripts in CIME:

```
/lustre/project/k1421/cesm2_1_3/cime/utils/perl5lib/compilers_translation_tool.pl  
/lustre/project/k1421/cesm2_1_3/cime/utils/perl5lib/Config/SetupTools.pm
```

After the 22 June 2021 upgrade, Bilel Hadri provided these instructions:

Here is a work around. On the cdl5, connect via ssh cdl5 once you are on shaheen and load the module perl/5.34.0. It should work.

The problem seems to occur when the `XML::LibXML` module isn't installed.

This `xml_test.pl` script triggers the error if the module is not installed:

```
#!/usr/bin/env perl  
  
use strict;  
use warnings;  
use XML::LibXML;  
  
print "Hello world!\n";  
  
my $new_node = XML::LibXML::Element->new("compiler");
```

Run the script as follows:

```
perl xml_test.pl
```

It will merely print “Hello World!” if the module is installed. If it isn't, an error message gets printed:

```
Error:      Can't locate XML/LibXML.pm in @INC (you may need to install the XML::LibXML  
module)      (@INC      contains:      /opt/slurm/default/lib/perl5/site_perl/5.26.1/x86_64-linux-thread-  
multi        /usr/lib/perl5/site_perl/5.26.1/x86_64-linux-thread-multi        /usr/lib/perl5/site_perl/5.26.1  
/usr/lib/perl5/vendor_perl/5.26.1/x86_64-linux-thread-multi        /usr/lib/perl5/vendor_perl/5.26.1  
/usr/lib/perl5/5.26.1/x86_64-linux-thread-multi /usr/lib/perl5/5.26.1 /usr/lib/perl5/site_perl) at xml_test.pl  
line 5. BEGIN failed--compilation aborted at xml_test.pl line 5.
```

## 25.1 Installing perl locally

Following the instructions in this [tutorial](#) and noting that batch jobs on Shaheen cannot write to project or home directories anymore, these steps install perl on scratch. It can then be copied to project space.

```
cd /lustre/scratch/x_johnsobk
wget https://www.cpan.org/src/5.0/perl-5.26.1.tar.gz
tar xzf perl-5.26.1.tar.gz
cd perl-5.26.1/
vim runme.sh
```

The `runme.sh` shell script contains:

```
#!/bin/bash

#SBATCH --job-name=build_perl
#SBATCH --account=k1421
#SBATCH --ntasks=32
#SBATCH --ntasks-per-node=32
#SBATCH --time=11:59:00
#SBATCH --partition=workq
#SBATCH --output=build_perl.out.%j

./Configure -des -Dprefix=/lustre/scratch/x_johnsobk/opt/perl
make
make test
make install

exit 0
```

Submit the shell script:

```
sbatch runme.sh
Submitted batch job 23628102
```

The perl installation path can be prepended to the `$PATH` environmental variable:

```
vim ~/.bashrc
[...]
export PATH=/lustre/scratch/x_johnsobk/opt/perl/bin:$PATH
[...]
source ~/.bashrc
```



## LOAD BALANCING TOOL

### 26.1 Overview

The load balancing tool attempts to find a reasonable processor element (PE) layout for a given compset and resolution.

### 26.2 Instructions

The instructions for the tool are available in the [CIME documentation](#). In order for the load balancing tool to run correctly, the CIME scripts and load balancing tool directories should be added to the PYTHONPATH environmental variable.

```
CIME_DIR=/lustre/project/k1421/cesm2_1_3/cime  
PYTHONPATH=$CIME_DIR/scripts:$CIME_DIR/tools/load_balancing_tool:$PYTHONPATH
```

Then run the submit script after specifying a given resolution, compset and PEs file.

```
$ cd /lustre/project/k1421/cesm2_1_3/cime/tools/load_balancing_tool  
$ ./load_balancing_submit.py -res <RESOLUTION> -compset <COMPSET> -pesfile <PESFILE>
```

After the jobs finish, run the solver:

```
$ ./load_balancing_solve.py -total-tasks <N> -blocksize 8
```



## CASE TIMING DATA

### 27.1 Overview

CIME uses the [General Purpose Timing Library \(GPTL\)](#) to provide detailed timing information.

Timing files are output to `$CASEROOT/timing/$model_timing.$CASE.$datestamp`.

The two areas we plan to focus on to increase the model performance is to understand the bottlenecks occurring during the `Init Time` phase of each job.

#### 27.1.1 Does init time take longer for first job in hybrid run?

No. I suspected that the init time might take longer for the first submission of a given integration since there may be initialization tasks involved for a hybrid run in which `continue_run = FALSE` but this seems to be fals.

For the first submission of `FHIST_BGC.f09_d025.052.e3` `Init Time` : 539.420 seconds while in the second submission `Init Time`: 562.777 seconds.

### 27.2 Major Issue: Init Time

It's unclear how to disassemble the `Init Time` statistic to determine what is causing the initialization to be so slow.

```
$ cd /lustre/project/k1421/cesm2_1_3/cime
$ grep -Rl "Init Time" ./
./doc/source/users_guide/timers.rst
./tools/load_balancing_tool/tests/timing/timing_2
./tools/load_balancing_tool/tests/timing/timing_3
./tools/load_balancing_tool/tests/timing/timing_1
./scripts/lib/CIME/get_timing.py
```

It looks like `get_timing.py` is the best candidate to look through for the relevant code:

```
nmax = self.gettime(' CPL:INIT ')[1]
...
self.write("    Init Time    : {:10.3f} seconds \n".format(nmax))
```

## 27.3 Previously Encountered Problem

In `_create_component_modelio_namelist`s

---

**Note:** This is the file to look in to see what is taking so long. `/lustre/scratch/x_johnsobk/FHIST_BGC.f09_d025.095.e500/run/cpl_0229.l`  
`165938.gz`

---

## FILE STRIPING

### 28.1 Optimizing I/O Performance

Shaheen uses the lustre filesystem, which provides the ability for [file striping](#).

Some general tips for optimizing I/O performance [include](#):

- Increasing the stripe count when multiple processes write to a single shared file, such as when parallel NetCDF is used
- Avoiding striping small files

Striping files and directories must be done with care. A note from Shaheen's documentation:

---

**Note:** *Once a file has been written to Lustre with a particular stripe configuration, you cannot simply use `setstripe` to change it. The file must be re-written with a new configuration. Generally, if you need to change the striping of a file, you can do one of two things:*

- *using `setstripe`, create a new, empty file with the desired stripe settings and then copy the old file to the new file, or*
  - *setup a directory with the desired configuration and `cp` (not `mv`) the file into the directory*
- 

Some other general considerations, per the documentation:

*Large files benefit from higher stripe counts. By striping a large file over many OSTs, you increase bandwidth for accessing the file and can benefit from having many processes operating on a single file concurrently. Conversely, a very large file that is only striped across one or two OSTs can degrade the performance of the entire Lustre system by filling up OSTs unnecessarily. A good practice is to have dedicated directories with high stripe counts for writing very large files into. Another scenario to avoid is having small files with large stripe counts.*

### 28.2 Advice from KAUST Supercomputing Lab

We corresponded with [Dr. Bilel Hadri](#), a computational scientist at the KAUST Supercomputing Lab, for advice on how to configure the file striping for the forcing and restart files. His general suggestions are as follows:

*For the I/O striping, I would suggest to go around 24/48 maximum probably only for the large files around 20-34GB, while keeping the default one at 1. Make sure to check the striping per file to validate them, since the striping is not inherited by the directory only at its creation of the file or copied, while when moved the file is keeping its original striping.*

## 28.3 Best LFS Stripe Count for Different Node Counts

This [presentation](#) contains a benchmarking graph on slide 13 that shows the optimal stripe count is around 100 for processes using more than 500 nodes.

## 28.4 Striping Commands

Below are sets of commands to change the striping of files and directories. Again, once a file has been written with a particular configuration, invoking *setstripe* will not change its file striping. A new file must be created with the desired striping and the old data must be copied into it.

### 28.4.1 Changing the striping of a file

If a file has been previously striped, we must create a new file, assign it the desired stripe configuration, then copy the contents of the old file into the new one.

```
$ cd /lustre/project/k1421/cesm_store/inputdata/atm/cam/tracer_cnst/
$ mv tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216.nc tracer_cnst_halons_
↪WACCM6_3Dmonthly_L70_1975-2014_c180216_old.nc
$ lfs setstripe -c 48 tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216.nc
$ cp tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216_old.nc tracer_cnst_halons_
↪WACCM6_3Dmonthly_L70_1975-2014_c180216.nc
```

Once this procedure has completed, we see that the file possesses the new file striping.

```
$ lfs getstripe tracer_cnst_halons_WACCM6_3Dmonthly_L70_1975-2014_c180216.nc
lmm_stripe_count: 48
lmm_stripe_size: 1048576
lmm_pattern: raid0
lmm_layout_gen: 0
lmm_stripe_offset: 84
```

### 28.4.2 Setting the striping of a directory

If a directory has not been previously striped, it is straightforward to stripe all of the files contained within it:

```
$ cd /lustre/scratch/x_johnsobk/archive/f.e21.FHIST_BGC.f09_025.CAM6assim.011/rest/
$ lfs setstripe --stripe-count 128 2019-08-05-00000
$ lfs getstripe -d 2019-08-05-00000
stripe_count: 128 stripe_size: 1048576 pattern: raid0 stripe_offset: -1
```

### 28.4.3 Changing the striping of a directory

If we want to change the striping of a directory that already has a certain filestriping configuration:

```
$ cd /lustre/scratch/x_johnsobk/archive/f.e21.FHIST_BGC.f09_025.CAM6assim.011/rest/  
$ lfs getstripe -d 2019-08-05-00000  
stripe_count: 128 stripe_size: 1048576 pattern: raid0 stripe_offset: -1
```

We need to move the directory, make a new one:

```
$ mv 2019-08-05-00000 2019-08-05-00000-old  
$ mkdir 2019-08-05-00000
```

Then copy the files from the old directory to the new one (which, of course, will take a while since there are 80 ensemble members):

```
$ cp 2019-08-05-00000-old/* 2019-08-05-00000  
$ rm -rf 2019-08-05-00000-old
```

And we'll see the new striping is present:

```
$ lfs getstripe -d 2019-08-05-00000  
stripe_count: 1 stripe_size: 1048576 pattern: 0 stripe_offset: -1
```





## DARSHAN

Darshan is a tool that produces a graphical summary of performance data for a particular job or a comparison of performance for two jobs.

George Markomanolis presented a [summary of the tool](#).

### 29.1 Installation

```
$ cd /lustre/project/k1421
$ mkdir -p logs/darshan
$ git clone https://github.com/KAUST-KSL/HarshaD.git darshan
$ cd darshan
$ vim open_darshan.sh
```

Set export darshan\_path="/lustre/project/k1421/logs/darshan" in open\_darshan.sh.

### 29.2 Finding Job IDs

The jhist alias, as described on the [SLURM](#) page prints out information for jobs run by a user since a specified date in YYYY-MM-DD format:

```
$ jhist 2020-11-13
      JobID      JobName  Partition  Account  AllocCPUS
↪ State ExitCode
-----
↪ -----
      16435549 run.FHIST_BGC.f09_d025.084.e03  workq    k1421    768
↪ COMPLETED 0:0
      16435549.batch      batch      k1421    64
↪ COMPLETED 0:0
      16435549.0      cesm.exe    k1421    384
↪ COMPLETED 0:0
```

## 29.3 Run Darshan for a Specific Job

Then pass desired the JobID to the Darshan scripts.

```
$ ./open_darshan.sh 16435549
```

## INTEL VTUNE PROFILER

In his S3 2020 Multicore 10 talk, Supreeth Suresh mentions his use of the [Intel VTune Profiler](#).  
He's using it on MOM6 and it seems like it might provide better profiling data than Darshan.



## ARM FORGE

### 31.1 Overview

*Arm Forge* is a suite of HPC development tools. It includes *Arm DDT*, a graphical debugger that can be run from one to thousands of threads.

### 31.2 Recompiling

In order for DDT to work optimally, CESM should be recompiled using two different compiler flags, `-g` which produces debugging information in the operating system's native object format and `-O0` which compiles without optimizations.

#### 31.2.1 Recompiling on Shaheen

Recompiling with the `-g` and `-O0` flags requires editing `/lustre/project/k1421/cesm2_1_3/cime/config/cesm/machines/config_compilers.xml`.

Editing:

```
<FFLAGS>
  <append> -xCORE-AVX2 </append>
  <append DEBUG="FALSE"> -O2 </append>
</FFLAGS>
```

to append this pair of flags when `DEBUG="TRUE"`:

```
<FFLAGS>
  <append> -xCORE-AVX2 </append>
  <append DEBUG="FALSE"> -O2 </append>
  <append DEBUG="TRUE"> -O0 -g </append>
</FFLAGS>
```

Recompiling requires logging onto `cdl5` and turning on debugging before building:

```
$ ssh cdl5
$ cd /lustre/project/k1421/cesm2_1_3/cime/scripts
$ source activate py27
$ ./create_newcase --case /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e002.n0003 --
↪compset FHIST --res f09_f09_mg17 --machine shaheen --project k1421 --run-unsupported --
↪ninst 3 --multi-driver --walltime 2:00:00
```

(continues on next page)

(continued from previous page)

```
[ ... ]
Creating Case directory /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e002.n0003
$ cd /lustre/project/k1421/cases/FHIST.f09_f09_mg17.e002.n0003
$ ./xmlchange DEBUG=TRUE
$ ./xmlquery DEBUG
DEBUG: TRUE
$ ./case.setup
$ ./case.build
```

## ENSEMBLE CONSISTENCY TESTS

The Ensemble Consistency Tests (ECT) are intended to determine whether output from a ported setup is statistically indistinguishable from an ensemble run on a trusted machine. Once the ensembles complete, history files are uploaded to the verification tests page of the [CESM2 python tools website](#).

---

**Note:** When conducting these tests initially on Shaheen, the scripts weren't able to run to completion because they weren't compatible with python3. I rewrote the relevant parts of the scripts and made a [pull request to Allison Baker's PyCECT repository](#). Thus, when running these tests on Shaheen, it was necessary to use python 2.7.

---

There are three sets of tests: a CAM and ultrafast CAM (UF-CAM) test and a POP test.

### 32.1 POP Ensemble Consistency Test

The POP test requires only running a single instance of POP for comparison to the ensemble run on Cheyenne.

```
$ cd /lustre/project/k1421/CESM/cime/tools/statistical_ensemble_test
$ python ensemble.py --case /lustre/scratch/x_johnsobk/cesm2.1.3/G.T62_g17.cesm2.1.3.000_
↪--ect pop --mach shaheen --project k1421 --compset G --res T62_g17 --ensemble 1
```

This passes the verification tests.

### 32.2 CAM Ensemble Consistency Test

The CAM test requires running three instances of CAM for comparison to the ensemble run on Cheyenne.

**Warning:** CGD does not run CAM ensembles for each of the minor releases of CESM. As of the 20th of September, 2020 there was no ensemble available for CESM2.1.3 and (it seems) no plans to create one. Thus, this ensemble will not pass the verification tests.

```
$ cd /lustre/project/k1421/CESM/cime/tools/statistical_ensemble_test
$ python ensemble.py --case /lustre/scratch/x_johnsobk/cesm2.1.3/F2000climo.f19_f19_mg17.
↪cesm2.1.3.000 --ect cam --mach shaheen --project k1421 --compset F2000climo --res f19_
↪f19_mg17 --ensemble 3
```

This fails the verification tests.

## 32.3 UF-CAM Ensemble Consistency Test

The UF-CAM test requires running three instances of CAM for a short 6-hour integration for comparison to the ensemble run on Cheyenne.

```
$ cd /lustre/project/k1421/CESM/cime/tools/statistical_ensemble_test
$ python ensemble.py --case /lustre/scratch/x_johnsobk/cesm2.1.3/uf.F2000climo.f19_f19_
↪mg17.cesm2.1.3.000 --ect cam --uf --mach shaheen --project k1421 --compset F2000climo -
↪-res f19_f19_mg17 --ensemble 3
```

This passes the verification tests.



## REGRESSION TESTS

### 33.1 Building cprnc

In order for the regression tests to work, the `cprnc` tool (which is a truncation of the phrase “compare NetCDF”) must be built.

```
$ cd /lustre/project/k1421/cesm2_1_relsd_m5.6/cime/tools/cprnc
$ CIMEROOT=../../.. ../configure --macros-format=Makefile --mpilib=mpi-serial
$ CIMEROOT=../../.. source ../env_mach_specific.sh && make
```

### 33.2 Edit config\_machines.xml

Now we set the path to `cprnc` within `config_machines.xml`:

```
$ cd /lustre/project/k1421/CESM/cime/config/cesm/machines
$ vim config_machines.xml
<CCSM_CPRNC>/lustre/project/k1421/CESM/cime/tools/cprnc</CCSM_CPRNC>
```

### 33.3 Run the tests

```
$ cd /lustre/project/k1421/CESM/cime/scripts/tests
$ ./scripts_regression_tests.py
```

It is unclear what constitutes a good result in a regression test. In this post by [Zoe Gillett on the DiscussCESM bulletin boards](#) a lot of the regression tests fail and Jim Edwards responds that, “If you are satisfied with the integrity of the experiments you want to run I think you can call this a good result.”



## CORI

Cori is a supercomputer supported by the US Department of Energy (DOE). Like Shaheen, Cori is also a Cray XC40. Unlike Shaheen, since NCAR and DOE partner for Earth System Model Computational Infrastructure, CESM is well-tested and supported on Cori.

Thus DAREs staff use Cori as a “trusted machine” to provide configuration and benchmarking hints for the CESM port on Shaheen.

### 34.1 Login instructions

#### 34.1.1 Multi-factor authentication

Cori uses multi-factor authentication (MFA) which can be configured by logging onto [Iris](#) and clicking on the MFA tab in the navigation menu.

#### 34.1.2 SSH

Cori is accessed via SSH:

```
ssh <username>@cori.nersc.gov
```

Ben’s username is `johnsonb`.

### 34.2 File system

The [NERSC file system](#) has:

- **home:** a global storage system that is mounted everywhere.

```
echo $HOME  
/global/homes/j/johnsonb
```

- **scratch:** a lustre file system for temporary, high-performance storage.

```
echo $scratch  
/global/cscratch1/sd/johnsonb
```

- **common:** a high performance platform for installing software and compiling code. On compute nodes, common is read-only.

- **community:** Large, permanent, medium-performance file system for sharing project files. For the DAREs project, this directory is `project/projectdirs/m3857`.
- **archive (HPSS):** A high capacity tape archive intended for long term storage of inactive and important data

### 34.2.1 Project

The DAREs project on Cori is m3857. The project usage can be found by typing `cmnquota <proj_name>`.

```
cmnquota m3857
----- Space (GB) ----- Inode -----
Project Usage Quota Percent Usage Quota Percent
-----
m3857      0    10      0      1 10000000      0
```

## 34.3 CESM

The CESM build requires an XML library to be installed.

#### Error:

```
ERROR: Command /global/project/projectdirs/m3857/cesm2_1_3/components/clm/bld/build-
↳namelist failed rc=2
out=
err=Can't locate XML/LibXML.pm in @INC (you may need to install the XML::LibXML
↳module)
```

You need to install this library via cpan:

```
cpan XML::LibXML
Result: PASS
SHLOMIF/XML-LibXML-2.0207.tar.gz
Tests succeeded but one dependency not OK (Alien::Libxml2)
SHLOMIF/XML-LibXML-2.0207.tar.gz
[dependencies] -- NA
```

### 34.3.1 Edit config\_machines.xml

Certain libraries have changed and must be updated in order to get CESM to run.

```
vim /project/projectdirs/m3857/cesm2_1_3/cime/config/cesm/machines/config_machines.xml
```

Update the following libraries:

Old	New
cray-mpich/7.7.8	cray-mpich/7.7.10
cray-netcdf-hdf5parallel/4.6.3.0	cray-netcdf-hdf5parallel/4.6.3.2
cray-hdf5-parallel/1.10.5.0	cray-hdf5-parallel/1.10.5.2
cray-parallel-netcdf/1.11.1.0	cray-parallel-netcdf/1.11.1.1

### 34.3.2 Installation

CESM is installed on Cori in the project's community directory.

```
cd /project/projectdirs/m3857
git clone https://github.com/escomp/cesm.git cesm2_1_3
cd cesm2_1_3
git checkout cesm2.1.3
./manageExternals/checkoutExternals
cd cime/scripts
./createNewcase --case /project/projectdirs/m3857/cases/FHIST_BGC.f19_f19_mg17.001.e3 --
↳ machine cori-haswell --res f19_f19_mg17 --project m3857 --walltime 1:00:00 --ninst 3 --
↳ compset HIST_CAM60_CLM50%BGC-CROP_CICE%PRES_DOCN%DOM_MOSART_SGLC_SWAV --multi-driver --
↳ run-unsupported
cd /project/projectdirs/m3857/cases/FHIST_BGC.f19_f19_mg17.001.e3
./case.setup
./case.build
```

Error with LibXml.pm

#### Error:

```
ERROR: Command /global/project/projectdirs/m3857/cesm2_2_0/components/clm/bld/build-
↳ namelist failed rc=2
out=
err=Can't locate XML/LibXML.pm in @INC (you may need to install the XML::LibXML
↳ module) (@INC contains: /global/project/projectdirs/m3857/cesm2_2_0/components/clm/
↳ bld /global/project/projectdirs/m3857/cesm2_2_0/components/clm/bld /global/project/
↳ projectdirs/m3857/cesm2_2_0/cime/scripts/Tools/../../utils/perl5lib /global/project/
↳ projectdirs/m3857/cesm2_2_0/components/clm/bld /usr/common/software/darshan/perl-
↳ packages/perl-Pod-LaTeX/lib/ /usr/lib/perl5/site_perl/5.26.1/x86_64-linux-thread-
↳ multi /usr/lib/perl5/site_perl/5.26.1 /usr/lib/perl5/vendor_perl/5.26.1/x86_64-
↳ linux-thread-multi /usr/lib/perl5/vendor_perl/5.26.1 /usr/lib/perl5/5.26.1/x86_64-
↳ linux-thread-multi /usr/lib/perl5/5.26.1 /usr/lib/perl5/site_perl) at /global/
↳ project/projectdirs/m3857/cesm2_2_0/cime/scripts/Tools/../../utils/perl5lib/Config/
↳ SetupTools.pm line 5.
BEGIN failed--compilation aborted at /global/project/projectdirs/m3857/cesm2_2_0/cime/
↳ scripts/Tools/../../utils/perl5lib/Config/SetupTools.pm line 5.
Compilation failed in require at /global/project/projectdirs/m3857/cesm2_2_0/
↳ components/clm/bld/CLMBuildNamelist.pm line 410
```



## GETTING FILES FROM THE RESEARCH DATA ARCHIVE

The output from the [CAM6+DART Reanalysis](#) is stored in NCAR's Research Data Archive.

### 35.1 Location of files on the GLADE filesystem

The files are available for Cheyenne users on the GLADE filesystem here:

```
cd /gpfs/fs1/collections/rda/data/ds345.0/rest/2018-01/  
cp *2018-01-01* /glade/scratch/johnsonb/archive/f.e21.FHIST_BGC.f09_025.CAM6assim.011/  
↪rest
```

### 35.2 Untarring all of the instances

In order to untar all of the instances, run `untar.py`:

```
#!/usr/bin/env python  
  
from __future__ import print_function  
  
import os  
  
for instance in range(1, 81):  
  
    instance_string = str(instance).zfill(4)  
    print(instance_string)  
  
    arch_string = 'f.e21.FHIST_BGC.f09_025.CAM6assim.011.'+instance_string+'.alltypes.'  
    ↪2018-01-01-00000.tar'  
  
    command = 'tar -xvf '+arch_string  
  
    print(command)  
    os.system(command)
```

Then simply unzip the compressed files:

```
cd /glade/scratch/johnsonb/archive/f.e21.FHIST_BGC.f09_025.CAM6assim.011/rest/2018-01-01-  
↔000000  
gunzip -v *.gz
```



## SPHINX AS A DOCUMENTATION TOOL

This is created using a documentation generation tool written in python known as [Sphinx](#).

If you are a regular python user, and would like to use Sphinx, install it using the package management procedure with which you are most comfortable.

### 36.1 Installing Conda

If you are novice python user, here are instructions for installing Sphinx using the package manager known as [Conda](#) (short for Anaconda) on Mac OSX.

Open a terminal window and execute these three commands:

1. Download the installation script:

```
curl -O https://repo.anaconda.com/miniconda/Miniconda3-latest-MacOSX-x86_64.sh
```

2. Modify the execution permissions of the recently downloaded script:

```
chmod 755 ./Miniconda3-latest-MacOSX-x86_64.sh
```

3. Run the installation script and follow its instructions:

```
./Miniconda3-latest-MacOSX-x86_64.sh
```

### 36.2 Downloading Sphinx

Now that Conda is installed, use it to download Sphinx. Close and reopen the terminal window to ensure conda is activated.

1. Invoke Conda and instruct it to install two packages, Sphinx and the “Read the Docs” CSS theme for Sphinx:

```
conda install sphinx sphinx-rtd-theme
```

## 36.3 Clone the Repository

You're ready to clone the repository:

```
git clone https://github.com/johnsonb-ucar/porting-cesm-to-shaheen.git
cd porting-cesm-to-shaheen
```

## 36.4 Edit Files and Remake the Documentation

The files that comprise the documentation are written in `reStructuredText` which offers a reasonably simple syntax for writing documentation while still accommodating the elements needed for technical writing such as equations, references, code snippets, tables, etc.

Find the `reStructuredText` documents contained in this repository:

```
find . -name "*.rst"
```

Edit them using your favorite text editor and then remake the documentation:

```
make clean
make html
```

View the remade documentation in your favorite web browser by opening the `./_build/html/index.html` using your graphical user interface or via the command line:

```
open ./_build/html/index.html
```