EAMxx 3.25 km Western Pacific RRM Technical Note

Jishi Zhang (zhang73@llnl.gov)[1]
[1]Lawrence Livermore National Laboratory

1 Introduction

The Western Pacific Convection-Permitting (CP) Regionally Refined Model (RRM) used for 2023 Jingjinji Flood event is developed based on the Simple Cloud-Resolving E3SM Atmosphere Model (SCREAM) version 0 (Fortran code) under the United States (U.S) Department of Energy (DOE) Energy Exascale Earth System Model (E3SM) project (Caldwell et al., 2021) and the regionally refined model (RRM) configuration (Tang et al., 2019, 2023; Zhang et al., 2024; Bogenschutz et al., 2024).

SCREAM WPRRM related code changes are located in https://github.com/jsbamboo/E3SM/tree/jzhang/WPRRMxx.

2 Resources

Table 1. Main resources for references.

Category	Link
New grid homepage	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/872579110/Running+E3SM+on+New+Grids
RRM grid Library	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/3690397775/Library+of+Regionally-Refined+Model+%28R
Topography	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/2720202817/Topography+Generation
atm initial condition	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/1002373272/Generate+atm+initial+condition+from+analysis
streamfile	https://esmci.github.io/cime/versions/ufs_release_v1.1/html/data_models/data-ocean.html
Nudging	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/20153276/How+to+perform+nudging+simulations+with+the
SE grid visualization	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/1210023949/Plotting+data+on+SE+native+grid
TempestRemap algorithm	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/178848194/Recommended+Mapping+Procedures+for+E3SM
ncremap	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/754286611/Regridding+E3SM+Data+with+ncremap
lnd initial condition	https://github.com/zarzycki/betacast/tree/master/land-spinup
CARRM v0 Technical Note	https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/3804299340/SCREAM+California+RRM+v0+Technical+No

Contents

3 Prepare E3SM codebase for standalone tools

```
git clone git@github.com:E3SM-Project/E3SM.git E3SM_tool_250318
git checkout -b jzhang/tools/revert_toprad_250318
git revert afb3c3221f6c439cfbbe7c4e9725fa014fa44867 567
    fa1d665a2909266a3ee774e725e97b30a7f05
vi components/elm/bld/namelist_files/namelist_defaults_tools.xml
git add components/elm/bld/namelist_files/namelist_defaults_tools.xml
git rm components/elm/tools/mksurfdata_map/src/mktopradMod.F90
git revert --continue
vi components/elm/bld/namelist_files/namelist_defaults.xml
vi components/elm/bld/namelist_files/namelist_defaults_tools.xml
git add components/elm/bld/namelist_files/namelist_defaults.xml components/elm/bld/
    namelist_files/namelist_defaults_tools.xml
git rm components/elm/tools/mksurfdata_map/src/mktopradMod.F90
git revert --continue
git submodule update --init --recursive; git submodule sync --recursive; git submodule
    update --recursive
vi components/elm/bld/namelist_files/namelist_defaults_tools.xml # forget something ..
git add components/elm/bld/namelist_files/namelist_defaults_tools.xml
git commit --amend
```

4 RRM grid design

This section is largely inherited (and simplified) from the SCREAM CARRM v0 Technical Note: https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/3804299340/SCREAM+California+RRM+v0+Technical+Note.

The main reference page (https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/872579110/Running+E3SM+on+New+Grids) is a veritable repository of the entire process of generating RRM grids and the associated files needed to run RRM, down to the specific commands for each step and and installation commands of tools. Due to the rapid development of grid tools, many steps have multiple choices (based on different tools or different packages). For example, for mapping function, TempestRemap, mbtempset and ESMF_Regridweightgen are well encapsulated by ncremap; users can choose which tool commands to use by their preference.

4.1 RRM exodus mesh

There are multiple WPRRM meshes. Two versions used in the summary paper are documented here:

- WP10ne32x32v1 (3km 10deg x 10deg)

- WP20ne32x32v1 (3km 20deg x 20deg)

1. Use SQuadGen to get the RRM grid and tempestremap to get the pg2 grid and SCRIP files:

```
git clone git@github.com:ClimateGlobalChange/squadgen.git squadgen-v1.2.2; cd squadgen-
    v1.2.2
vi /p/lustre2/zhang73/GitTmp/E3SM_tool_250318/cime_config/machines/config_machines.xml
     # find NETCDF_FORTRAN_PATH for specific mach in E3SM
vi src/Makefile # copy NETCDF_FORTRAN_PATH
make all
./SQuadGen --output WP10ne32x32v1.g --lat_ref 7.5 --lon_ref 165 --refine_rect "
    160,2.5,170,12.5,5" --refine_level 5 --resolution 32 --smooth_type SPRING
./SQuadGen --output WP20ne32x32v1.g --lat_ref 7.5 --lon_ref 165 --refine_rect "
    155,-2.5,175,17.5,5" --refine_level 5 --resolution 32 --smooth_type SPRING

conda create -n tempest-remap-2.2.0 -c conda-forge tempest-remap; conda activate
    tempest-remap-2.2.0
GenerateVolumetricMesh --in WP10ne32x32v1.g --out WP10ne32x32v1pg2.g --np 2 --uniform
ConvertMeshToSCRIP --in WP10ne32x32v1.g --out WP10ne32x32v1_scrip.nc
ConvertMeshToSCRIP --in WP10ne32x32v1pg2.g --out WP10ne32x32v1pg2_scrip.nc
GenerateVolumetricMesh --in WP20ne32x32v1.g --out WP20ne32x32v1pg2.g --np 2 --uniform
ConvertMeshToSCRIP --in WP20ne32x32v1.g --out WP20ne32x32v1_scrip.nc
ConvertMeshToSCRIP --in WP20ne32x32v1pg2.g --out WP20ne32x32v1pg2_scrip.nc

rsync -av WP10ne32x32v1.g WP20ne32x32v1.g $DIN_LOC_ROOT/atm/cam/inic/homme/
```

The mesh of Western Pacific RRMs are shown in Fig. 1, plotted using NCL:
`WPRRM_Doc_scripts/grid_WL.01.SQuadGen_03.draw_SCRIP_demo.ncl`

4.2 mapping file

WPRRM model grid for atm, land, ocnice are all on WP10ne32x32v1/WP20ne32x32v1, while the river transport model is on a regular latitude-longitude grid with spacing of 0.125°. The mapping files are used for OCN2ATM ATM2ROF (flux, state/vector) and LND2ROF/ROF2LND (flux). All use FV→ FV aave (conservative, monotone, 1st order) algorithm.

Also need to create the mapping file for the Simple Prescribed Aerosol (SPA) scheme used in EAMxx. This is always from ne30np4 to the new grid pg2.

We highly suggest that readers refer to this web page for the knowledge of mapping procedures (notation, validation, issues, etc.) especially if readers plan to use bi-grid or tri-grid for RRM: https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/178848194/Recommended+Mapping+Procedures+for+E3SM+Atmosphere+Grids#RecommendedMappingProceduresforE3SMAtmosph

- OCN2ATM:

```
ncremap -5 -a fv2fv_mono -s $DIN_LOC_ROOT/ocn/mpas-o/oRRS18to6v3/ocean.oRRS18to6v3.scrip
    .181106.nc -g ${mapping_root}/${atm_grid_name}.g -m ${mapping_root}//map_${
    ocn_grid_name}_to_${atm_grid_name}.TRaave.${date_tag}.nc
```

- LND2ROF:

```
ncremap -5 -a fv2fv_mono -s WP10ne32x32v1pg2.g -g $DIN_LOC_ROOT/share/meshes/rof/
    MOSART_global_8th.scrip.20180211c.nc -m map_WP10ne32x32v1pg2_to_r0125.TRaave
    .20250318.nc
```

- ROF2LND:

```
ncremap -5 -a fv2fv_mono -s $DIN_LOC_ROOT/share/meshes/rof/MOSART_global_8th.scrip
    .20180211c.nc -g WP10ne32x32v1pg2.g -m map_r0125_to_WP10ne32x32v1pg2.TRaave
    .20250318.nc
```

- mapping file for SPA (intbilin, SE -> FV):

```
ncremap -5 -a intbilin_se2fv -s $DIN_LOC_ROOT/atm/cam/inic/homme/ne30.g -g
    WP10ne32x32v1pg2.g -m map_ne30np4_to_WP10ne32x32v1pg2.intbilin.20250318.nc
```

4.3 domain file

Here the domain files are generated for "oRRS18to6v3" data-ocean (streamfile) and "WP10ne32x32v1/WP20ne32x32v1" atm & land.

WPRRM_Doc_scripts/grid_WL.02.gen_domain.sh

```
#!/bin/bash

RRMgrid=WP10ne32x32v1
e3sm_root=/p/lustre2/zhang73/GitTmp/E3SM_tool_250318/
gen_domain=${e3sm_root}/cime/tools/mapping/gen_domain_files/gen_domain
mapping_root=/p/lustre2/zhang73/grids2/${RRMgrid}/
ocn_grid_name=oRRS18to6v3
atm_grid_name=${RRMgrid}pg2
date_tag=$(date +"%Y%m%d")
lnd_grid_name=${atm_grid_name}

do_step="build"
do_step="remap"
do_step="gen_domain"
```

```
# ------------------------------------------------------------------------
if [ "${do_step}" == "build" ];then
cd `dirname ${gen_domain}`/src
eval $(${e3sm_root}/cime/CIME/Tools/get_case_env)
${e3sm_root}/cime/CIME/scripts/configure --macros-format Makefile --mpilib mpi-serial # =>
     get .env_mach_specific.sh
#--- addtional changes needed for dane:
#       [Makefile] LDFLAGS += -L$(NETCDF_C_PATH)/lib -lnetcdf -L$(NETCDF_FORTRAN_PATH)/lib
     -lnetcdff
# .or.  [Makefile] LDFLAGS += $(shell nc-config --libs) $(shell nf-config --flibs)
# .and. [Makefile] LDFLAGS += -Wl,-rpath,$(NETCDF_C_PATH)/lib -Wl,-rpath,$(
    NETCDF_FORTRAN_PATH)/lib
# .or.  [.env_mach_specific.sh] export LD_LIBRARY_PATH=$HDF5_ROOT/lib:$NETCDF_C_PATH/lib:
    $NETCDF_FORTRAN_PATH/lib:$LD_LIBRARY_PATH
source .env_mach_specific.sh
gmake
exit 1
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "remap" ];then
source /usr/workspace/e3sm/apps/e3sm-unified/load_latest_e3sm_unified_dane.sh

ncremap -a fv2fv_mono -s $DIN_LOC_ROOT/ocn/mpas-o/oRRS18to6v3/ocean.oRRS18to6v3.scrip
    .181106.nc -g ${mapping_root}/${atm_grid_name}.g -m ${mapping_root}//map_${
    ocn_grid_name}_to_${atm_grid_name}.TRaave.${date_tag}.nc
ncremap -a fv2fv_mono -s ${mapping_root}/${atm_grid_name}.g -g $DIN_LOC_ROOT/share/meshes/
    rof/MOSART_global_8th.scrip.20180211c.nc -m ${mapping_root}/map_${atm_grid_name}
    _to_r0125.TRaave.${date_tag}.nc
ncremap -a fv2fv_mono -s $DIN_LOC_ROOT/share/meshes/rof/MOSART_global_8th.scrip.20180211c.
    nc -g ${mapping_root}/${atm_grid_name}.g -m ${mapping_root}/map_r0125_to_${
    atm_grid_name}.TRaave.${date_tag}.nc
ncremap -a intbilin_se2fv -s $DIN_LOC_ROOT/atm/cam/inic/homme/ne30.g -g ${mapping_root}/${
    atm_grid_name}.g -m ${mapping_root}/map_ne30np4_to_${atm_grid_name}.intbilin.${
    date_tag}.nc
```

```
rsync -av ${mapping_root}/map*r0125*TRaave.${date_tag}.nc  $DIN_LOC_ROOT/cpl/gridmaps/${
    atm_grid_name}/
rsync -av ${mapping_root}/map_ne30np4_to_${atm_grid_name}.intbilin.${date_tag}.nc
    $DIN_LOC_ROOT/atm/scream/maps/
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "gen_domain" ];then
# source `dirname ${gen_domain}`/src/.env_mach_specific.sh

domain_root=${mapping_root}
cd ${domain_root}
for target_grid_name in ${lnd_grid_name} ${atm_grid_name}; do
    map_ocn_to_target=${mapping_root}/map_${ocn_grid_name}_to_${target_grid_name}.TRaave.$
    {date_tag}.nc
    ${gen_domain} -m ${map_ocn_to_target} -o ${ocn_grid_name} -l ${target_grid_name}
done
fi

rsync -av domain.ocn.${atm_grid_name}_${ocn_grid_name}.${date_tag}.nc $DIN_LOC_ROOT/share/
    domains/
```

4.4 topography

The topography file was generated using the NCAR topography toolchain (Lauritzen et al., 2015), with tensor hyperviscosity enabled for the RRM grid. V3 topography tool chain was used in WPRRM codebase (https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/2712338924/V3+Topography+GLL+PG2+grids).

WPRRM_Doc_scripts/grid_WL.03.topo_v3_demo_WP10ne32x32v1.sh

```
#!/bin/bash

RRMgrid=WP10ne32x32v1
e3sm_root=/p/lustre2/zhang73/GitTmp/E3SM_tool_250318/
e3sm_root_env=/p/lustre2/zhang73/GitTmp/SCREAM_tool/
machine=dane-intel
homme_tool_root=${e3sm_root}/components/homme/test/tool
grids2=/p/lustre2/zhang73/grids2/
```

```
do_step="step1.1_homme_tool_np4_build"
do_step="step1.2_homme_tool_np4_run"
do_step="step1.3_homme_tool_np4_ncl"
do_step="step2.1_cube_to_target_run1_build"
do_step="step2.2_cube_to_target_run1_run"
# do_step="step3_homme_tool_smoothing"
# do_step="step4_cube_to_target_run2"
# do_step="step5_ncks_smoothedtopo"

echo ${do_step} '...'
# ------------------------------------------------------------------------------

#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
#--- Step 1: Create GLL and pg2 grid template files for !!!
#       the "USGS-topo-cube3000" high res data and the  !!!
#       target EAM grid.                                !!!
#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
#--- Generate GLL SCRIP file for target grid: for RRM grids, this SCRIP files are good
    enough
#--- for topo downsampling, but not conservative enough for use in the coupled model:
# ------------------------------------------------------------------------------
if [ "${do_step}" == "step1.1_homme_tool_np4_build" ];then
#--- 1.1 build homme_tool ---
# eval $(${e3sm_root}/cime/CIME/Tools/get_case_env)
rm -rf ${e3sm_root}/cmake_homme && mkdir ${e3sm_root}/cmake_homme && cd ${e3sm_root}/
    cmake_homme
source ${e3sm_root_env}/components/eam/tools/topo_tool/bin_to_cube/.env_mach_specific.sh
cmake \
    -C ${e3sm_root}/components/homme/cmake/machineFiles/dane-intel.cmake \
    -DBUILD_HOMME_WITHOUT_PIOLIBRARY=OFF \
    -DPREQX_PLEV=26 ${e3sm_root}/components/homme/
make -j4 homme_tool
exit 1
fi

# ------------------------------------------------------------------------------
if [ "${do_step}" == "step1.2_homme_tool_np4_run" ];then
#--- 1.2 run homme_tool ---
```

```
cd /p/lustre2/zhang73/grids2/${RRMgrid}/

rm -f input.nl
cat > input.nl <<EOF
&ctl_nl
ne = 0
mesh_file = "/p/lustre2/zhang73/grids2/${RRMgrid}/${RRMgrid}.g"
/
&vert_nl
/

&analysis_nl
tool = 'grid_template_tool'
output_dir = "./"
output_timeunits=1
output_frequency=1
output_varnames1='area','corners','cv_lat','cv_lon'
!output_type='netcdf'
output_type='netcdf4p'  ! needed for ne1024
io_stride = 16
/
EOF

rm -f homme_tool_inputnl.sh
cat > homme_tool_inputnl.sh <<EOF
#!/bin/bash
#
#SBATCH --account=focus
#SBATCH --job-name=topo_gene
#SBATCH --nodes=1
##SBATCH -C cpu
#SBATCH --time=00:05:00
#SBATCH -p pdebug

source /p/lustre2/zhang73/GitTmp/SCREAM_tool/components/eam/tools/topo_tool/bin_to_cube/.
    env_mach_specific.sh
srun  -K -c 1 -N 1 /p/lustre2/zhang73/GitTmp/E3SM_tool_250318/cmake_homme/src/tool/
    homme_tool < input.nl
```

```
EOF
sbatch --exclusive homme_tool_inputnl.sh
exit 1
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "step1.3_homme_tool_np4_ncl" ];then
#--- 1.3 NCL ---
# build NCL: conda install -c conda-forge ncl
source ~/.bashrc_all_stable
# ---make the 'scrip' file for target GLL grid
ncks -O -v lat,lon,area,cv_lat,cv_lon ne0np4_tmp1.nc ${RRMgrid}np4_tmp.nc
ncl ${e3sm_root}/components/homme/test/tool/ncl/HOMME2SCRIP.ncl  name=\"${RRMgrid}np4\"
    ne=0  np=4
exit 1
fi


#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
#--- Step 2: cube_to_target, run 1: Compute phi_s on the np4 grid.   !!!
#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
# ------------------------------------------------------------------------------
if [ "${do_step}" == "step2.1_cube_to_target_run1_build" ];then
#--- build cube_to_target ---
export OS=Linux
cd ${e3sm_root}/components/eam/tools/topo_tool/cube_to_target
# eval $(${e3sm_root}/cime/CIME/Tools/get_case_env)
${e3sm_root}/cime/CIME/scripts/configure
source .env_mach_specific.sh
#--- changes added to ${e3sm_root}/components/eam/tools/topo_tool/cube_to_target/Makefile:
#       LDFLAGS += $(USER_LDFLAGS)
# .and. LDFLAGS += -Wl,-rpath,$(NETCDF_C_PATH)/lib -Wl,-rpath,$(NETCDF_FORTRAN_PATH)/lib
    INC_NETCDF="`nf-config --includedir`" \
        LIB_NETCDF="`nc-config --libdir`" USER_FC="`nc-config --fc`" \
        USER_LDFLAGS="`nc-config --libs` `nf-config --flibs`" make
exit 1
fi
```

```
# ------------------------------------------------------------------------------
if [ "${do_step}" == "step2.2_cube_to_target_run1_run" ];then
${e3sm_root}/components/eam/tools/topo_tool/cube_to_target/cube_to_target \
--target-grid ${grids2}/${RRMgrid}/${RRMgrid}np4_scrip.nc \
--input-topography ${grids2}/USGS-topo-cube3000.nc \
--output-topography ${grids2}/${RRMgrid}/${RRMgrid}np4_gtopo30.nc
exit 1
fi

# ------------------------------------------------------------------------------
if [ "${do_step}" == "step3_homme_tool_smoothing" ];then
#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
#--- Step 3: homme_tool:                                                  !!!
#     Starting with the unsmoothed topo data on the GLL grid,     !!!
#     apply dycore specific smoothing. This uses the standard     !!!
#     tensor laplace smoothing algorithm with 6 iterations.       !!!
#     We use the naming convention "xNt", where N is the value    !!!
#     used for smooth_phis_numcycle and "t" denotes the use of    !!!
#     the tensor laplace.                                         !!!
#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
cd /p/lustre2/zhang73/grids2/${RRMgrid}/

rm -f input2.nl
cat > input2.nl <<EOF
&ctl_nl
ne = 0
mesh_file = '/p/lustre2/zhang73/grids2/${RRMgrid}/${RRMgrid}.g'
smooth_phis_p2filt = 0
smooth_phis_numcycle = 6       ! increase for more smoothing
!smooth_phis_numcycle = 12              !for 16xdel2
!hypervis_order = 2                     !for 16xdel2
smooth_phis_nudt = 4e-16
hypervis_scaling = 2
se_ftype = 2 ! actually output NPHYS; overloaded use of ftype
/
&vert_nl
/
&analysis_nl
```

```
tool = 'topo_pgn_to_smoothed'
infilenames = './${RRMgrid}np4_gtopo30.nc', './${RRMgrid}np4_smoothed_phis_x6t'
! output_type = 'netcdf'
io_stride = 16
/
EOF

rm -f homme_tool_inputnl2.sh
cat > homme_tool_inputnl2.sh <<EOF
#!/bin/bash
#
#SBATCH --account=focus
#SBATCH --job-name=topo_gene
#SBATCH --nodes=1
##SBATCH -C cpu
#SBATCH --time=00:05:00
#SBATCH -p pdebug

source /p/lustre2/zhang73/GitTmp/SCREAM_tool/components/eam/tools/topo_tool/bin_to_cube/.
    env_mach_specific.sh
srun  -K -c 1 -N 1 /p/lustre2/zhang73/GitTmp/E3SM_tool_250318/cmake_homme/src/tool/
    homme_tool < input2.nl
EOF
sbatch --exclusive homme_tool_inputnl2.sh
exit 1
fi

# ------------------------------------------------------------------------------
if [ "${do_step}" == "step4_cube_to_target_run2" ];then
# #!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
# #--- Step 4: cube_to_target, run 2: Compute SGH, SGH30, LANDFRAC,       !!!
# #       and LANDM_COSLAT on the pg2 grid, using the pg2 phi_s data.     !!!
# #!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

${e3sm_root}/components/eam/tools/topo_tool/cube_to_target/cube_to_target \
--target-grid ${grids2}/${RRMgrid}/${RRMgrid}pg2_scrip.nc \
--input-topography ${grids2}/USGS-topo-cube3000.nc \
--smoothed-topography ${grids2}/${RRMgrid}/${RRMgrid}np4_smoothed_phis_x6t1.nc \
```

```
  --output-topography ${grids2}/${RRMgrid}/GTOPO30_${RRMgrid}np4pg2_x6t.nc
  exit 1
  fi

  # ------------------------------------------------------------------------
  if [ "${do_step}" == "step5_ncks_smoothedtopo" ];then
  # !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  # --- Step 5: ncks: Append the GLL phi_s data to the output of step 4.   !!!
  # !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  cd /p/lustre2/zhang73/grids2/${RRMgrid}/
  source /usr/workspace/e3sm/apps/e3sm-unified/load_latest_e3sm_unified_dane.sh

  ncks -A ${RRMgrid}np4_smoothed_phis_x6t1.nc GTOPO30_${RRMgrid}np4pg2_x6t.nc

  rsync -av GTOPO30_${RRMgrid}np4pg2_x6t.nc $DIN_LOC_ROOT/atm/cam/topo/
  exit 1
  fi
```

4.5 land surface data

Two steps are needed to generate land surface data (*fsurdat*):

1. preparation for the land surface data (`grid_WL.fsurdat_0102.mkmapdata_mksurfdata_pl_map_demo_WP10ne32x32`
   1.1. generate mapping files for each land surface input data file to input grid files (mkmapdata.sh)
   1.2. build "mksurfdata_map" tool
   1.3. generate namelist using mksurfdata.pl, and modify it (if needed)
2. create the land surface data by "mksurfdata_map" (`grid_WL.fsurdat_02.mksurfdata_map_namelist.bash`)

Beside the land surface data, the land use file (*flanduse_timeseries*) is also needed for a grid-specified compset in multi-year simulations. The additional steps are:

1. create a LUT (Land Use Translator) file list
2. assign the *mksrf_fdynuse* to the LUT list in ksurfdata_map's namelist
3. assign the *fdyndat* to the name of the land use file you want to create

`WPRRM_Doc_scripts/grid_WL.04.fsurdat_0102.mkmapdata_mksurfdata_pl_map_demo_WP10ne32x32v1.s`

```
  #!/bin/sh

  RRMgrid=WP10ne32x32v1
```

```
#---use the up-to-date elm tool, and revert the toprad commits to avoid `the map_0.01x0.01
    does not found error'
e3sm_root=/p/lustre2/zhang73/GitTmp/E3SM_tool_250318
grids2=/p/lustre2/zhang73/grids2/
GRIDFILE=${grids2}/${RRMgrid}/${RRMgrid}pg2_scrip.nc
INPUTDATA_ROOT=$DIN_LOC_ROOT
date_tag=$(date +"%Y%m%d")
year_fsurdat=2015
# rcp_tag=""
rcp_level=8.5
rcp_tag="-rcp ${rcp_level}"

do_step="step1_mkmapdata"
do_step="step3_build_mksurfdata_map"
do_step="step3_gen_mksurfdata_pl"
do_step="step4_run_mksurfdata_map"

use_multiN=false #only needed for map_1km-merge-10min_HYDRO1K-merge-nomask

# ------------------------------------------------------------------------
if [ "${do_step}" == "step1_mkmapdata" ];then
cd ${e3sm_root}/components/elm/tools/mkmapdata

echo '--- ESMFBIN_PATH start mkmapdata_sbatch.sh ---'
#--- use esmf-mpi from unified env ---
export ESMFBIN_PATH=/g/g92/zhang73/miniconda3/envs/esmf/bin
export ESMFBIN_PATH=/usr/WS1/e3sm/apps/e3sm-unified/base/envs/e3sm_unified_1.10.0_login/
    bin
echo $ESMFBIN_PATH
#--- env with ncl and nco
source ~/.bashrc_all_stable

#--- Speed: WP10ne32x32v1 (WP20ne32x32v1) took 7.5min (<29min) for 1km-merge-10min_HYDRO1K
    -merge-nomask
export mpiexec="srun --account=focus --time=00:29:00 -p pbatch -N 20"

#--- dry-run to check the mapping file list
```

```
# ./mkmapdata.sh --gridfile ${GRIDFILE} --inputdata-path ${INPUTDATA_ROOT} --res ${RRMgrid
    }pg2 --gridtype global --esmf-path ${ESMFBIN_PATH} --output-filetype 64bit_offset --
    debug -v --list
# exit 1

#--- regridding
if $use_multiN; then
./mkmapdata.sh --mpiexec "${mpiexec}" --gridfile ${GRIDFILE} --inputdata-path ${
    INPUTDATA_ROOT} --res ${RRMgrid}pg2 --gridtype global --esmf-path ${ESMFBIN_PATH}  --
    output-filetype 64bit_offset -v --batch
else
./mkmapdata.sh --gridfile ${GRIDFILE} --inputdata-path ${INPUTDATA_ROOT} --res ${RRMgrid}
    pg2 --gridtype global --esmf-path ${ESMFBIN_PATH} --output-filetype 64bit_offset -v
fi
echo " ./mkmapdata.sh --gridfile ${GRIDFILE} --inputdata-path ${INPUTDATA_ROOT} --res ${
    RRMgrid}pg2 --gridtype global --output-filetype 64bit_offset -v"
exit 1
fi

# ------------------------------------------------------------------------
if  [ "${do_step}" == "step3_gen_mksurfdata_pl" ];then
cd ${e3sm_root}/components/elm/tools/mksurfdata_map

./mksurfdata.pl -res usrspec -usr_gname ${RRMgrid}pg2 -usr_gdate 250320 -y ${year_fsurdat}
     ${rcp_tag} -d -dinlc ${INPUTDATA_ROOT} -usr_mapdir ${e3sm_root}/components/elm/tools/
    mkmapdata

#---modify it manually if needed, e.g., add the landuse.timeseries <mksrf_fdynuse> <
    fdyndat>
cp namelist namelist_${RRMgrid}pg2_rcp${rcp_level}-${year_fsurdat}
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "step2_build_mksurfdata_map" ];then
cd ${e3sm_root}/components/elm/tools/mksurfdata_map/src

# eval $(${e3sm_root}/cime/CIME/Tools/get_case_env)
${e3sm_root}/cime/CIME/scripts/configure --macros-format Makefile --mpilib mpi-serial
```

```
source ${e3sm_root}/components/elm/tools/mksurfdata_map/src/.env_mach_specific.sh

INC_NETCDF="`nf-config --includedir`" \
    LIB_NETCDF="`nc-config --libdir`" USER_FC="`nc-config --fc`" \
    USER_LDFLAGS="`nc-config --libs` `nf-config --flibs` -Wl,-rpath,${NETCDF_C_PATH}/lib -
    Wl,-rpath,${NETCDF_FORTRAN_PATH}/lib" make
exit 1
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "step4_run_mksurfdata_map" ];then
cd ${e3sm_root}/components/elm/tools/mksurfdata_map

source ${e3sm_root}/components/elm/tools/mksurfdata_map/src/.env_mach_specific.sh
./mksurfdata_map < namelist_${RRMgrid}pg2_rcp${rcp_level}-${year_fsurdat}

mv surfdata_${RRMgrid}pg2_rcp8.5_simyr2015_${date_tag}.nc  landuse.timeseries_${RRMgrid}
    pg2_rcp8.5_simyr2015-2100_${date_tag}.nc  ${grids2}/${RRMgrid}/
rsync -av ${grids2}/${RRMgrid}/surfdata_${RRMgrid}pg2_rcp8.5_simyr2015_${date_tag}.nc
    landuse.timeseries_${RRMgrid}pg2_rcp8.5_simyr2015-2100_${date_tag}.nc  $DIN_LOC_ROOT/
    lnd/clm2/surfdata_map/
fi
```

4.6 dry deposition

WPRRM_Doc_scripts/grid_WL.05.mkatmsrffile_demo_WP10ne32x32v1.sh

```
#!/bin/bash

RRMgrid=WP10ne32x32v1
e3sm_root=/p/lustre2/zhang73/GitTmp/E3SM_tool_250318
mkatmsrffile=${e3sm_root}/components/eam/tools/mkatmsrffile/old_mkatmsrffile
grids2=/p/lustre2/zhang73/grids2/
INPUTDATA_ROOT=$DIN_LOC_ROOT
date_tag=$(date +"%Y%m%d")

do_step="step1_mkatmsrffile_build"
do_step="step2_gen_map"
do_step="step3_mkatmsrffile_run"
```

```
# ------------------------------------------------------------------------
if [ "${do_step}" == "step1_mkatmsrffile_build" ];then
cd ${mkatmsrffile} && echo ${mkatmsrffile}
# eval $(${e3sm_root}/cime/CIME/Tools/get_case_env)
${e3sm_root}/cime/CIME/scripts/configure --macros-format=Makefile
source .env_mach_specific.sh

#---Modify Makefile:
#       INC = -I$(shell nf-config --includedir)
#       LIB = -L$(shell nc-config --libdir) -lnetcdf -lnetcdff
#       LIB += $(USER_LDFLAGS)
 FC="`nc-config --fc`" \
    USER_LDFLAGS="`nc-config --libs` `nf-config --flibs` -Wl,-rpath,${NETCDF_C_PATH}/lib -
    Wl,-rpath,${NETCDF_FORTRAN_PATH}/lib" make
exit 1
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "step2_gen_map" ];then
source /usr/workspace/e3sm/apps/e3sm-unified/load_latest_e3sm_unified_dane.sh

ncremap -5 -a fv2fv_mono -s ${grids2}/1x1d.nc -g ${grids2}/${RRMgrid}/${RRMgrid}pg2.g -m $
    {grids2}/${RRMgrid}/map_1x1_to_${RRMgrid}pg2.TRaave.${date_tag}.nc
exit 1
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "step3_mkatmsrffile_run" ];then
cat > ${mkatmsrffile}/nml_atmsrf <<EOF
&input
srfFileName = '${grids2}/1x1d.nc'
landFileName = '${INPUTDATA_ROOT}/atm/cam/chem/trop_mozart/dvel/regrid_vegetation.nc'
soilwFileName = '${INPUTDATA_ROOT}/atm/cam/chem/trop_mozart/dvel/clim_soilw.nc'
atmFileName = '${grids2}/${RRMgrid}/${RRMgrid}pg2_scrip.nc'
srf2atmFmapname = '${grids2}/${RRMgrid}/map_1x1_to_${RRMgrid}pg2.TRaave.${date_tag}.nc'
outputFileName = '${grids2}/${RRMgrid}/atmsrf_${RRMgrid}pg2_${date_tag}.nc'
/
```

```
EOF

cd ${mkatmsrffile}
./mkatmsrffile

mv nml_atmsrf nml_atmsrf_${RRMgrid}pg2

rsync -av ${grids2}/${RRMgrid}/atmsrf_${RRMgrid}pg2_${date_tag}.nc  $DIN_LOC_ROOT/atm/cam/
    chem/trop_mam/
fi
```

5 Initial conditions

5.1 atmosphere IC

The atmosphere IC was generated with the HICCUP package (https://github.com/E3SM-Project/HICCUP), which has a built-in download of ERA5 pressure level data, a call to NCO's vertical interpolation algorithm (https://nco.sourceforge.net/nco.html), a call to TempestRemap horizontal interpolation algorithm (https://github.com/ClimateGlobalChange/tempestremap), and a procedure for adjusting surface temperature and pressure with topography following the ECMWF practice (Trenberth et al., 1993).

A good step-by-step tutorial can be found here: https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/1002373272/Generate+atm+initial+condition+from+analysis+data. The readers are suggested to learn the HICCUP tool (https://github.com/E3SM-Project/HICCUP) and the relevant references for a better sense.

Optionally, one can manually generate the mapping files for np4 (IC) and pg2 (nudging) -> ERA5 0.25 deg and pass them to the HICCP run scripts (*hiccup_data.map_file*):

```
ncremap -a fv2se_stt --grd_src=/pscratch/sd/z/zhang73/DATA/data_hiccup/ERA5_721x1440_scrip
    .20220907.nc --grd_dst=/global/cfs/cdirs/e3sm/zhang73/grids2/WP10ne32x32v1/
    WP10ne32x32v1.g --map=/global/cfs/cdirs/e3sm/zhang73/grids2/WP10ne32x32v1/
    map_ERA5_721x1440_to_WP10ne32x32v1np4.TRhighorder.20250322.nc
ncremap -a fv2fv_mono --grd_src=/pscratch/sd/z/zhang73/DATA/data_hiccup/
    ERA5_721x1440_scrip.20220907.nc --grd_dst=/global/cfs/cdirs/e3sm/zhang73/grids2/
    WP10ne32x32v1/WP10ne32x32v1pg2_scrip.nc --map=/global/cfs/cdirs/e3sm/zhang73/grids2/
    WP10ne32x32v1/map_ERA5_721x1440_to_WP10ne32x32v1pg2.TRaave.20250322.nc
```

We did not need to spin up the atmosphere and to adjust the hyperviscosity additionally. The hyperviscosity timestep for dynamics were set to the default value inferred by the scaling factor from the coarse resolution to RRM.

HICCUP scripts (IC, nudging):

```
zhang73@perlmutter:login39:/global/cfs/cdirs/e3sm/zhang73/GitTmp/SourceCode/HICCUP/
    template_hiccup_scripts>
zhang73@perlmutter:login01:/pscratch/sd/z/zhang73/GitTmp/SourceCode/HICCUP/
    template_hiccup_scripts>
-rw-rw---- 1 zhang73 zhang73  12K Sep 29  2024 process_nudging_data_EAMxx.2010-01
    _L128_CAne32x32v1pg2.py
-rw-rw---- 1 zhang73 zhang73 8.5K Mar 22 15:12 create_EAMxx_IC_from_ERA5.2014-10-01
    _L128_WP10ne32x32v1.py
-rw-rw---- 1 zhang73 zhang73 8.5K Mar 22 15:18 create_EAMxx_IC_from_ERA5.2014-10-01
    _L128_WP20ne32x32v1.py
-rw-rw---- 1 zhang73 zhang73  12K Mar 25 17:42 process_nudging_data_EAMxx.2014-10
    _L128_WP20ne32x32v1pg2.py
-rw-rw---- 1 zhang73 zhang73  12K Mar 25 17:42 process_nudging_data_EAMxx.2014-10
    _L128_WP10ne32x32v1pg2.py
```

5.2 land IC

Since the WPRRMs include little land areas, we dont need to use a spunup land for shot-time hindcasts. For the climate-length simulations, we can directly use a land IC (e.g., v1 DECK) interpolated from a well-spunup run.

We only need:

1. *source_inic_file*: a land restart file from a balanced run to get the balanced land condition
2. *target_inic_file*: a land restart file from the new grid run to get the grid info

`WPRRM_Doc_scripts/grid_WL.06.finidat.IELM_interpinic.demo_E3SMv1toRRM_WP10ne32x32v1.sh`

```
#!/bin/bash

RRMgrid=WP20ne32x32v1
e3sm_root=/p/lustre2/zhang73/GitTmp/E3SM_tool_250318 #---this netcdf version is not high
    enough to activate NF_FORMAT_64BIT_OFFSET

interpinic=${e3sm_root}/components/elm/tools/interpinic/
output_root=/p/lustre2/zhang73/grids2/finidat_interpinic
lnd_grid_name=${RRMgrid}pg2

do_step="step1_interpinic_build"
```

```
do_step="step2_interpinic_run"

# ------------------------------------------------------------------------
if [ "${do_step}" == "step1_interpinic_build" ];then
cd ${interpinic}/src
#---this netcdf version is not high enough to activate NF_FORMAT_64BIT_OFFSET
# eval $(${e3sm_root}/cime/CIME/Tools/get_case_env)
# ${e3sm_root}/cime/CIME/scripts/configure --macros-format Makefile --mpilib mpi-serial
# source .env_mach_specific.sh
source /p/lustre2/zhang73/GitTmp/SCREAM_tool/components/eam/tools/topo_tool/bin_to_cube/.
    env_mach_specific.sh

INC_NETCDF="`nf-config --includedir`" \
    LIB_NETCDF="`nc-config --libdir`" USER_FC="`nc-config --fc`" \
    USER_LDFLAGS="`nc-config --libs` `nf-config --flibs` -Wl,-rpath,${NETCDF_C_PATH}/lib -
    Wl,-rpath,${NETCDF_FORTRAN_PATH}/lib" make
exit 1
fi

# ------------------------------------------------------------------------
if [ "${do_step}" == "step2_interpinic_run" ];then
  source_inic_file=/p/lustre2/zhang73/HPSS/20180215.DECKv1b_H1.ne30_oEC.edison/20180215.
    DECKv1b_H1.ne30_oEC.edison.clm2.r.2015-01-01-00000.nc
  output_inic_file=${output_root}/${lnd_grid_name}.elm.r.2015-01-01.nc
  if [ "${RRMgrid}" == "WP10ne32x32v1" ];then
  target_inic_file=/p/lustre1/zhang73/E3SM_simulations/wprrmxx_p3/WPRRMxx_p3.
    WP10ne32x32v1pg2_WP10ne32x32v1pg2.F2010-SCREAMv1.dane/tests/1120x1_nhoursx1_UVTQ3h-
    s20141001-finicold-O3p3/run/WPRRMxx_p3.WP10ne32x32v1pg2_WP10ne32x32v1pg2.F2010-
    SCREAMv1.dane.elm.r.2014-10-01-03600.nc
  fi
  if [ "${RRMgrid}" == "WP20ne32x32v1" ];then
  target_inic_file=/p/lustre1/zhang73/E3SM_simulations/wprrmxx_p3/WPRRMxx.
    WP20ne32x32v1pg2_WP20ne32x32v1pg2.F2010-SCREAMv1.dane/tests/2240x1_nhoursx1_UV3h-
    s20141001-finicold-O1/run/WPRRMxx.WP20ne32x32v1pg2_WP20ne32x32v1pg2.F2010-SCREAMv1.
    dane.elm.r.2014-10-01-03600.nc
  fi

  if ! test -f ${output_inic_file}; then cp ${target_inic_file} ${output_inic_file}; fi
```

```
    cd ${e3sm_root}/components/elm/tools/interpinic
    source /p/lustre2/zhang73/GitTmp/SCREAM_tool/components/eam/tools/topo_tool/bin_to_cube
      /.env_mach_specific.sh
    ./interpinic -i ${source_inic_file} -o ${output_inic_file}

    rsync -av ${output_inic_file} ${output_root}/../${RRMgrid}/
  fi
```

6 Model configurations (CIME xml)

A series of CIME xml files need to be modified/created to support the new RRM grid and the specific compset required for the simulation. The first part is related to the *grid*. The second part is mainly related to the *compset* (partially subject to common requirements of grid and compset). The RRM related XML list can be found here: https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/872579110/Running+E3SM+on+New+Grids.

The CIME xml files related to WPRRM are listed as follows:

- `grid`
  - `cime_config/config_grids.xml`
  - `components/eam/bld/config_files/horiz_grid.xml`
  - `driver-mct/cime_config/config_component_e3sm.xml`
  - `components/eamxx/cime_config/namelist_defaults_eamxx.xml`
  - `components/eam/bld/namelist_files/namelist_defaults_eam.xml`
  - `components/elm/bld/namelist_files/namelist_definition.xml`
- `grid & compset`
  - `components/elm/bld/namelist_files/namelist_defaults.xml`

Note that for SCREAMv1: 1) dont need to add ncdata in `components/eam/bld/namelist_files/namelist_defaults_e` 2) dont need to add landuse as the the `2010\_defscream\_control` compset set the *sim_year_range* to be constant.

We directly use the *FSSP585-SCREAM-HR* compset developed in CARRM v0 Tech Note: https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/3804299340/SCREAM+California+RRM+v0+Technical+Note. So, only the grid-related XML changes are documented here.

6.1 Prepare E3SM codebase for simulations

```
git clone git@github.com:E3SM-Project/E3SM.git WPRRMxx
git remote add fork git@github.com:jsbamboo/E3SM.git
git checkout -b jzhang/WPRRMxx
git submodule update --init --recursive; git submodule sync --recursive; git submodule
    update --recursive
# After modify all xml
git add .
git commit -m "set WP10ne32x32v1 and WP20ne32x32v1 grids for eamxx"
git checkout -b jzhang/WPRRMxx_p3-extra-diags
git fetch origin
# Merge p3-extra-diags branch
git merge origin/aarondonahue/eamxx/mahf708/p3-extra-diags
```

6.2 grid

1. Specify the grid-related files in eamxx's namelist:

- *Filename* (full pathname of initial atmospheric state dataset in NetCDF format)
- *spa_remap_file* (mapping file for SPA scheme)
- *topography_filename* (full pathname of time-invariant boundary dataset for topography fields)
- *mesh_file* (exodus format grid file used when *se_ne*=0, i.e., RRM grids)

2. Specify the grid-related files in eam's namelist:
   - *drydep_srf_file* (dry deposition surface values interpolated to model grid, required for unstructured atmospheric grids with modal chemistry)
   - *bnd_topo*, same as *topography_filename* in eamxx nl
   - *mesh_file*, same as in eamxx nl
3. Set the parameters related to the stability of the model in eamxx's namelist:
   - *rad_frequency* (rad timestep, specified as number of atm steps)
   - *number_of_subcycles* (how many times to subcycle this atm process)
4. Set the parameters related to the stability of the model in both eamxx's and eam's namelists:
   - *nu_top* (second-order viscosity applied only near the model top [m2/s]), set to follow the the finest resolution grid
   - *se_tstep* (dynamics timesteps)
   - *dt_tracer_factor* (the tracer advection timestep is *dt_tracer_factor*se_tstep*, where *se_tstep* is the dynamics timesteps)
   - *hypervis_subcycle_q* (number of hyperviscosity subcycles done in tracer advection code)
   - *hv_ref_profiles* (Modifications to hyperviscosity to minimize dissipation of background reference states)
   - *pgrad_correction* (Turn on use of balanced geopotential state to reduce pressure gradient discretization error)

`cime_config/config_grids.xml`

```
      <model_grid alias="WP10ne32x32v1pg2_WP10ne32x32v1pg2">
        <grid name="atm">ne0np4_WP10ne32x32v1.pg2</grid>
        <grid name="lnd">ne0np4_WP10ne32x32v1.pg2</grid>
        <grid name="ocnice">ne0np4_WP10ne32x32v1.pg2</grid>
        <grid name="rof">r0125</grid>
        <grid name="glc">null</grid>
        <grid name="wav">null</grid>
        <mask>oRRS18to6v3</mask>
      </model_grid>

      <model_grid alias="WP20ne32x32v1pg2_WP20ne32x32v1pg2">
        <grid name="atm">ne0np4_WP20ne32x32v1.pg2</grid>
        <grid name="lnd">ne0np4_WP20ne32x32v1.pg2</grid>
        <grid name="ocnice">ne0np4_WP20ne32x32v1.pg2</grid>
        <grid name="rof">r0125</grid>
```

```
    <grid name="glc">null</grid>
    <grid name="wav">null</grid>
    <mask>oRRS18to6v3</mask>
  </model_grid>

  <domain name="ne0np4_WP10ne32x32v1.pg2">
    <nx>96240</nx>
    <ny>1</ny>
    <file grid="atm|lnd" mask="oRRS18to6v3">$DIN_LOC_ROOT/share/domains/domain.lnd.
  WP10ne32x32v1pg2_oRRS18to6v3.250318.nc</file>
    <file grid="ice|ocn" mask="oRRS18to6v3">$DIN_LOC_ROOT/share/domains/domain.ocn.
  WP10ne32x32v1pg2_oRRS18to6v3.250318.nc</file>
    <desc>1-deg with 10 deg x 10 deg 3 km over Western Pacific version 1 pg2:</desc>
  </domain>

  <domain name="ne0np4_WP20ne32x32v1.pg2">
    <nx>267792</nx>
    <ny>1</ny>
    <file grid="atm|lnd" mask="oRRS18to6v3">$DIN_LOC_ROOT/share/domains/domain.lnd.
  WP20ne32x32v1pg2_oRRS18to6v3.250318.nc</file>
    <file grid="ice|ocn" mask="oRRS18to6v3">$DIN_LOC_ROOT/share/domains/domain.ocn.
  WP20ne32x32v1pg2_oRRS18to6v3.250318.nc</file>
    <desc>1-deg with 20 deg x 20 deg 3 km over Western Pacific version 1 pg2:</desc>
  </domain>

  <gridmap atm_grid="ne0np4_WP10ne32x32v1.pg2" rof_grid="r0125">
   <map name="ATM2ROF_FMAPNAME">cpl/gridmaps/WP10ne32x32v1pg2/
  map_WP10ne32x32v1pg2_to_r0125.TRaave.20250318.nc</map>
   <map name="ATM2ROF_SMAPNAME">cpl/gridmaps/WP10ne32x32v1pg2/
  map_WP10ne32x32v1pg2_to_r0125.TRaave.20250318.nc</map>
   <map name="LND2ROF_FMAPNAME">cpl/gridmaps/WP10ne32x32v1pg2/
  map_WP10ne32x32v1pg2_to_r0125.TRaave.20250318.nc</map>
   <map name="ROF2LND_FMAPNAME">cpl/gridmaps/WP10ne32x32v1pg2/
  map_r0125_to_WP10ne32x32v1pg2.TRaave.20250318.nc</map>
  </gridmap>

  <gridmap atm_grid="ne0np4_WP20ne32x32v1.pg2" rof_grid="r0125">
```

```
    <map name="ATM2ROF_FMAPNAME">cpl/gridmaps/WP20ne32x32v1pg2/
    map_WP20ne32x32v1pg2_to_r0125.TRaave.20250318.nc</map>
    <map name="ATM2ROF_SMAPNAME">cpl/gridmaps/WP20ne32x32v1pg2/
    map_WP20ne32x32v1pg2_to_r0125.TRaave.20250318.nc</map>
    <map name="LND2ROF_FMAPNAME">cpl/gridmaps/WP20ne32x32v1pg2/
    map_WP20ne32x32v1pg2_to_r0125.TRaave.20250318.nc</map>
    <map name="ROF2LND_FMAPNAME">cpl/gridmaps/WP20ne32x32v1pg2/
    map_r0125_to_WP20ne32x32v1pg2.TRaave.20250318.nc</map>
    </gridmap>
```

components/eam/bld/config_files/horiz_grid.xml

```
<horiz_grid dyn="se" hgrid="ne0np4_WP10ne32x32v1"          ncol="216542" csne="0" csnp=
    "4" npg="0" />
<horiz_grid dyn="se" hgrid="ne0np4_WP10ne32x32v1.pg2"      ncol="96240"  csne="0" csnp=
    "4" npg="2" />
<horiz_grid dyn="se" hgrid="ne0np4_WP20ne32x32v1"          ncol="602534" csne="0" csnp=
    "4" npg="0" />
<horiz_grid dyn="se" hgrid="ne0np4_WP20ne32x32v1.pg2"      ncol="267792" csne="0" csnp=
    "4" npg="2" />
```

components/eam/bld/namelist_files/namelist_defaults_eam.xml

```
<bnd_topo hgrid="ne0np4_WP10ne32x32v1"  >atm/cam/topo/GTOPO30_WP10ne32x32v1np4pg2_x6t.nc</
    bnd_topo>
<bnd_topo hgrid="ne0np4_WP20ne32x32v1"  >atm/cam/topo/GTOPO30_WP20ne32x32v1np4pg2_x6t.nc</
    bnd_topo>
<drydep_srf_file hgrid="ne0np4_WP10ne32x32v1">atm/cam/chem/trop_mam/
    atmsrf_WP10ne32x32v1pg2_20250321.nc</drydep_srf_file>
<drydep_srf_file hgrid="ne0np4_WP20ne32x32v1">atm/cam/chem/trop_mam/
    atmsrf_WP20ne32x32v1pg2_20250321.nc</drydep_srf_file>
<se_ne hgrid="ne0np4_WP10ne32x32v1"> 0 </se_ne>
<se_ne hgrid="ne0np4_WP20ne32x32v1"> 0 </se_ne>
<mesh_file hgrid="ne0np4_WP10ne32x32v1"  >atm/cam/inic/homme/WP10ne32x32v1.g</mesh_file>
<mesh_file hgrid="ne0np4_WP20ne32x32v1"  >atm/cam/inic/homme/WP20ne32x32v1.g</mesh_file>
<nu_top dyn_target="theta-l" hgrid="ne0np4_WP10ne32x32v1"> 1e4 </nu_top>
<nu_top dyn_target="theta-l" hgrid="ne0np4_WP20ne32x32v1"> 1e4 </nu_top>
<se_tstep            dyn_target="theta-l" hgrid="ne0np4_WP10ne32x32v1"> 8.333333333333333
    d0 </se_tstep>
```

```
  <dt_tracer_factor    dyn_target="theta-l" hgrid="ne0np4_WP10ne32x32v1"> 6 </
      dt_tracer_factor>
  <hypervis_subcycle_q dyn_target="theta-l" hgrid="ne0np4_WP10ne32x32v1"> 6 </
      hypervis_subcycle_q>
  <se_tstep            dyn_target="theta-l" hgrid="ne0np4_WP20ne32x32v1"> 8.333333333333333
      d0 </se_tstep>
  <dt_tracer_factor    dyn_target="theta-l" hgrid="ne0np4_WP20ne32x32v1"> 6 </
      dt_tracer_factor>
  <hypervis_subcycle_q dyn_target="theta-l" hgrid="ne0np4_WP20ne32x32v1"> 6 </
      hypervis_subcycle_q>
```

components/eamxx/cime_config/namelist_defaults_eamxx.xml

```
        <spa_remap_file hgrid="ne0np4_WP10ne32x32v1">${DIN_LOC_ROOT}/atm/scream/maps/
      map_ne30np4_to_WP10ne32x32v1pg2.intbilin.20250318.nc</spa_remap_file>
        <spa_remap_file hgrid="ne0np4_WP20ne32x32v1">${DIN_LOC_ROOT}/atm/scream/maps/
      map_ne30np4_to_WP20ne32x32v1pg2.intbilin.20250318.nc</spa_remap_file>
        <rad_frequency hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">3</rad_frequency>
        <number_of_subcycles hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">1</
      number_of_subcycles>
      <Filename hgrid="ne0np4_WP10ne32x32v1" nlev="128">${DIN_LOC_ROOT}/atm/scream/init/
      HICCUP.atm_era5.2014-10-01.highorder_WP10ne32x32v1.L128.nc</Filename>
      <Filename hgrid="ne0np4_WP20ne32x32v1" nlev="128">${DIN_LOC_ROOT}/atm/scream/init/
      HICCUP.atm_era5.2014-10-01.highorder_WP20ne32x32v1.L128.nc</Filename>
      <topography_filename hgrid="ne0np4_WP10ne32x32v1">${DIN_LOC_ROOT}/atm/cam/topo/
      GTOPO30_WP10ne32x32v1np4pg2_x6t.nc</topography_filename>
      <topography_filename hgrid="ne0np4_WP20ne32x32v1">${DIN_LOC_ROOT}/atm/cam/topo/
      GTOPO30_WP20ne32x32v1np4pg2_x6t.nc</topography_filename>
      <nc hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">0.0</nc>
      <ni hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">0.0</ni>
      <hv_ref_profiles hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">0</hv_ref_profiles>
      <nu_top hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">1.0e4</nu_top>
      <pgrad_correction hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">0</
      pgrad_correction>
      <se_ne hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1">0</se_ne>
      <se_tstep hgrid="ne0np4_WP10ne32x32v1|ne0np4_WP20ne32x32v1" constraints="gt 0">
      8.3333333333333</se_tstep>
      <mesh_file hgrid="ne0np4_WP10ne32x32v1">${DIN_LOC_ROOT}/atm/cam/inic/homme/
      WP10ne32x32v1.g</mesh_file>
```

```
    <mesh_file hgrid="ne0np4_WP20ne32x32v1">${DIN_LOC_ROOT}/atm/cam/inic/homme/
    WP20ne32x32v1.g</mesh_file>
```

components/elm/bld/namelist_files/namelist_definition.xml

```
 <entry id="res" type="char*30" category="default_settings"
        group="default_settings"
        valid_values=
 "...,ne0np4_WP10ne32x32v1.pg2,ne0np4_WP20ne32x32v1.pg2,...">
 Horizontal resolutions
```

components/elm/bld/namelist_files/namelist_defaults.xml

```
 <fsurdat hgrid="ne0np4_WP10ne32x32v1.pg2"   sim_year="2010" use_crop=".false." >
 lnd/clm2/surfdata_map/surfdata_WP10ne32x32v1pg2_rcp8.5_simyr2015_c250320.nc</fsurdat>
 <fsurdat hgrid="ne0np4_WP20ne32x32v1.pg2"   sim_year="2010" use_crop=".false." >
 lnd/clm2/surfdata_map/surfdata_WP20ne32x32v1pg2_rcp8.5_simyr2015_c250320.nc</fsurdat>
```

driver-mct/cime_config/config_component_e3sm.xml

```
   <entry id="ATM_NCPL">
     <type>integer</type>
     <default_value>48</default_value>
     <values match="last">
       ...
       <value compset=".+" grid="a%ne0np4_WP10ne32x32v1">864</value>
       <value compset=".+" grid="a%ne0np4_WP20ne32x32v1">864</value>
```

6.3 compset

Simply use the “F2010-SCREAMv1” compset.

7 Boundary conditions

This section is quite different from CARRM v0 Tech Note. E3SMv1 1° outputs were used to generate the nudging inputs as well as SST & ice cover, while ERA5 and OISST were used for Beijing Flood hindcasts.

7.1 create lower BL (SST, ice cover)

SST & ice cover were obtained from NOAA OISST as lower boundary conditions to drive Data Ocean (*PRES_DOCN*) and Prescribed CICE (*SPBC_CICE*) as a streamfile:

1. use HICCUP to download NOAA SST and sea ice
2. use “poisson_grid_fill” function in NCL to fill missing values over land
3. replace missing values of ice cover by 0, and add date & datesec variables in streamfile

7.2 create lateral BL (nudging files)

To constrain the lateral boundary conditions, the nudging capability has been plugged into the E3SM RRM framework. A global wind nudging is used with the nudging coefficient set by 1 over the whole globe and a consistent nudging strength in the vertical direction. A good step-by-step tutorial to use nudging for RRM can be found here: https://acme-climate.atlassian.net/wiki/spaces/DOC/pages/20153276/How+to+perform+nudging+simulations+with+the+regional+refined+model+RRM.

The hourly wind profiles were interpolated vertically by nco and horizontally by the TempestRemp monotune algorithm (“fv2fv_flx” in nco) to the SCREAM Beijing RRM grid as nudging inputs. Either HICCUP or a Bash script can be used to regrid the nudging files while we use HICCUP to generate all nudging data in eamxx format in this work. The surface adjustment is turned off for nudging. Hourly nudging inputs were linearly interpolated (online) onto the model physical time step with a relaxation timescale of 3 hour.

To enable the window (regional) nudging in SCREAMv1, we need to generate a nudging weights file offline and prescrib its full path in the runscript.

How to generate the nudging weights file:

```
git clone git@github.com:E3SM-Project/eamxx-scripts.git
cd /p/lustre2/zhang73/GitTmp/SourceCode/eamxx-scripts/run_scripts/RRM_example_scripts
cp SCREAMv1_create_nudging_weights.py SCREAMv1_create_nudging_weights_WP10ne32x32v1pg2.py
vi SCREAMv1_create_nudging_weights_WP10ne32x32v1pg2.py # Modify ``USER DEFINED SETTINGS''
python3  SCREAMv1_create_nudging_weights_WP10ne32x32v1pg2.py  -datafile /p/lustre2/zhang73
    /grids2/WP10ne32x32v1/GTOPO30_WP10ne32x32v1np4pg2_x6t.nc -nlev 128 -lat lat -lon lon -
    weightsfile /p/lustre2/zhang73/grids2/WP10ne32x32v1/WP10ne32x32v1pg2_weighting_file.nc
```

7.3 output yaml

```
ncremap -a fv2fv_flx -s /p/lustre2/zhang73/grids2/WP10ne32x32v1/WP10ne32x32v1pg2_scrip.nc
    -g /p/lustre2/zhang73/grids2/ne30/ne30_pg2_scrip.nc -m /p/lustre2/zhang73/grids2/
    WP10ne32x32v1/map_WP10ne32x32v1pg2_to_ne30pg2.TRaave.20250430.nc
ncremap -a fv2fv_flx -s /p/lustre2/zhang73/grids2/WP20ne32x32v1/WP20ne32x32v1pg2_scrip.nc
    -g /p/lustre2/zhang73/grids2/ne30/ne30_pg2_scrip.nc -m /p/lustre2/zhang73/grids2/
    WP10ne32x32v1/map_WP20ne32x32v1pg2_to_ne30pg2.TRaave.20250430.nc
```

7.4 user namelists

The user namelists need to be modified to enable the lower/lateral BLs generated in the previous step. All the modifications are put in the `runscript.sh`.

- add prescribed SST and ice cover settings
  - set streams for DOCN in `user_nl_docn`
  - set SSTICE_DATA_FILENAME, SSTICE_YEAR_ALIGN, SSTICE_YEAR_START, SSTICE_YEAR_END by xmlchange
  - set stream_fldfilename for SPBC_CICE in `user_nl_cice`
- add GHG forcing by atmchange
- add nudging settings by atmchange
- set yaml outputs by atmchange
- set RUN_STARTDATE, STOP_OPTION, RUN_TYPE, RESUBMIT etc. by xmlchange

`runscript.sh`

```
...
#-----------------------------------------------------
case_setup() {
...
    ./xmlchange EPS_AGRID=1e-9

    ./xmlchange PIO_NETCDF_FORMAT="64bit_data"
...
}

...
user_nl() {

# let's put all user namelist setup here

cat << EOF >> user_nl_cpl
 ocn_surface_flux_scheme = 2
EOF

# cice && docn nl is needed if you want to set the realistic SST and ice_cov forcing in
    hindcasts
cat > user_nl_cice << 'eof'
```

```
stream_fldfilename = '/p/lustre2/zhang73/DATA/data_hiccup/sst_ice.daymean.2014.fillmsg.fmt
    -c250322.nc'
stream_domfilename = '/usr/workspace/e3sm/ccsm3data/inputdata/ocn/docn7/domain.ocn.0.25x0
    .25.c20190221.nc'
 model_year_align               = 2014
 stream_fldvarname              = 'ice_cov'
 stream_year_first              = 2014
 stream_year_last               = 2014
eof

cat > user_nl_docn << 'eof'
 streams = 'docn.streams.txt.prescribed 2014 2014 2014'
eof

}

...

runtime_options() {

    echo $'\n----- Starting runtime_options -----\n'
    pushd ${CASE_SCRIPTS_DIR}

    # Set simulation start date
    if [ ! -z "${START_DATE}" ]; then
        ./xmlchange RUN_STARTDATE=${START_DATE}
    fi
    # Set temperature cut off in dycore threshold to 180K
    ./atmchange vtheta_thresh=180

    # Set nudging
    ./case.setup
    ./atmchange mac_aero_mic::atm_procs_list=tms,shoc,cldFraction,spa,p3,nudging
    ./case.setup
    # make sure that ``time'' is set to unlimited o/w we'll receive SIGSEGV: "invalid
    memory reference without other clues"
    ./atmchange physics::mac_aero_mic::nudging::nudging_filenames_patterns=${NUDGING_ROOT
    }/HICCUP.atm_era5.201410??_??.mono_WP20ne32x32v1pg2.L128.nc
```

```
./atmchange physics::mac_aero_mic::nudging::nudging_fields=U,V
./atmchange mac_aero_mic::nudging::source_pressure_type="TIME_DEPENDENT_3D_PROFILE"
# we do can activate online horiz_remap + weighted nudging at the same time
#   << if you want that, comment the EKAT MSG with ``coarse'' and ``weighted'' in
eamxx_nudging_process_interface.cpp in the source code
./atmchange mac_aero_mic::nudging::nudging_refine_remap_mapfile="no-file-given"
./atmchange physics::mac_aero_mic::nudging::skip_vert_interpolation=true
./atmchange physics::mac_aero_mic::nudging::nudging_timescale=10800
# need to generate a netcdf file of nudging_weights.  Please see the script
#  SCREAMv1_create_nudging_weights.py to do this.
./atmchange physics::mac_aero_mic::nudging::use_nudging_weights=true
./atmchange physics::mac_aero_mic::nudging::nudging_weights_file=/p/lustre2/zhang73/
grids2/WP20ne32x32v1/WP20ne32x32v1pg2_weighting_file.nc
# dont know why now we cannot ask for compute_tendencies for nudging. error: "The key
'nudging_T_mid_tend' is not associated to any registered product"
#./atmchange physics::mac_aero_mic::nudging::compute_tendencies=T_mid,qv

# Set atmos IC file
# Allow for tendency outputs
./atmchange physics::mac_aero_mic::shoc::compute_tendencies=T_mid,qv
./atmchange physics::mac_aero_mic::p3::compute_tendencies=T_mid,qv
./atmchange physics::rrtmgp::compute_tendencies=T_mid
./atmchange homme::compute_tendencies=T_mid,qv

# use GHG levels more appropriate for 2019
./atmchange co2vmr=410.5e-6
./atmchange ch4vmr=1877.0e-9
./atmchange n2ovmr=332.0e-9
./atmchange orbital_year=2019
# use CO2 the same in land model

./xmlchange CCSM_CO2_PPMV=410.5

#user_nl #if you set user_docn.streams instead, must activate it here
./xmlchange SSTICE_GRID_FILENAME="/usr/workspace/e3sm/ccsm3data/inputdata/ocn/docn7/
domain.ocn.0.25x0.25.c20190221.nc"
./xmlchange SSTICE_DATA_FILENAME="/p/lustre2/zhang73/DATA/data_hiccup/sst_ice.daymean
.2014.fillmsg.fmt-c250322.nc"
```

```
./xmlchange SSTICE_YEAR_ALIGN="2014"
./xmlchange SSTICE_YEAR_START="2014"
./xmlchange SSTICE_YEAR_END="2014"

# Segment length
./xmlchange STOP_OPTION=${STOP_OPTION,,},STOP_N=${STOP_N}

# Restart frequency
./xmlchange REST_OPTION=${REST_OPTION,,},REST_N=${REST_N}

# Coupler history
./xmlchange HIST_OPTION=${HIST_OPTION,,},HIST_N=${HIST_N}

# Coupler budgets (always on)
./xmlchange BUDGETS=TRUE

# Set resubmissions
if (( RESUBMIT > 0 )); then
    ./xmlchange RESUBMIT=${RESUBMIT}
fi

# Run type
# Start from default of user-specified initial conditions
if [ "${MODEL_START_TYPE,,}" == "initial" ]; then
    ./xmlchange RUN_TYPE="startup"
    ./xmlchange CONTINUE_RUN="FALSE"

# Continue existing run
elif [ "${MODEL_START_TYPE,,}" == "continue" ]; then
    ./xmlchange CONTINUE_RUN="TRUE"

elif [ "${MODEL_START_TYPE,,}" == "branch" ] || [ "${MODEL_START_TYPE,,}" == "hybrid"
]; then
    ./xmlchange RUN_TYPE=${MODEL_START_TYPE,,}
    ./xmlchange GET_REFCASE=${GET_REFCASE}
    ./xmlchange RUN_REFDIR=${RUN_REFDIR}
    ./xmlchange RUN_REFCASE=${RUN_REFCASE}
    ./xmlchange RUN_REFDATE=${RUN_REFDATE}
```

```
        echo 'Warning: $MODEL_START_TYPE = '${MODEL_START_TYPE}
        echo '$RUN_REFDIR = '${RUN_REFDIR}
        echo '$RUN_REFCASE = '${RUN_REFCASE}
        echo '$RUN_REFDATE = '${START_DATE}

    else
        echo 'ERROR: $MODEL_START_TYPE = '${MODEL_START_TYPE}' is unrecognized. Exiting.'
        exit 380
    fi

    cp ${YAML_ROOT}"/scream_1hI_pg2.yaml" .
    ./atmchange output_yaml_files="./scream_1hI_pg2.yaml"
    cp ${YAML_ROOT}"/scream_1dA_pg2.yaml" .
    ./atmchange output_yaml_files+="./scream_1dA_pg2.yaml"

    popd
}
...
```

Acknowledgements. This documentation is supported by LLNL LDRD project "IMPACT OF GLOBAL CLI". Work at LLNL was performed under the auspices of the U.S. DOE by the Lawrence Livermore National Laboratory under contract (grant no. DE-AC52-07NA27344; IM release: LLNL-TR-2012974).

References

Bogenschutz, P., Zhang, J., Tang, Q., and Cameron-Smith, P.: Atmospheric River Induced Precipitation in California as Simulated by the Regionally Refined Simplified Convective Resolving E3SM Atmosphere Model, EGUsphere [preprint], https://doi.org/https://doi.org/10.5194/egusphere-2024-839, 2024.

Caldwell, P. M., Terai, C. R., Hillman, B., Keen, N. D., Bogenschutz, P., Lin, W., Beydoun, H., Taylor, M., Bertagna, L., Bradley, A. M., Clevenger, T. C., Donahue, A. S., Eldred, C., Foucar, J., Golaz, J. C., Guba, O., Jacob, R., Johnson, J., Krishna, J., Liu, W., Pressel, K., Salinger, A. G., Singh, B., Steyer, A., Ullrich, P., Wu, D., Yuan, X., Shpund, J., Ma, H. Y., and Zender, C. S.: Convection-Permitting Simulations With the E3SM Global Atmosphere Model, Journal of Advances in Modeling Earth Systems, 13, https://doi.org/10.1029/2021ms002544, 2021.

Lauritzen, P. H., Bacmeister, J. T., Callaghan, P. F., and Taylor, M. A.: NCAR_Topo (v1.0): NCAR global model topography generation software for unstructured grids, Geoscientific Model Development, 8, 3975–3986, https://doi.org/10.5194/gmd-8-3975-2015, cz6xg Times Cited:22 Cited References Count:49, 2015.

Tang, Q., Klein, S. A., Xie, S. C., Lin, W. Y., Golaz, J. C., Roesler, E. L., Taylor, M. A., Rasch, P. J., Bader, D. C., Berg, L. K., Caldwell, P., Giangrande, S. E., Neale, R. B., Qian, Y., Riihimaki, L. D., Zender, C. S., Zhang, Y. Y., and Zheng, X.: Regionally refined test bed in E3SM atmosphere model version 1 (EAMv1) and applications for high-resolution modeling, Geoscientific Model Development, 12, 2679–2706, https://doi.org/10.5194/gmd-12-2679-2019, ih4ju Times Cited:31 Cited References Count:92, 2019.

Tang, Q., Golaz, J.-C., Van Roekel, L. P., Taylor, M. A., Lin, W., Hillman, B. R., Ullrich, P. A., Bradley, A. M., Guba, O., Wolfe, J. D., Zhou, T., Zhang, K., Zheng, X., Zhang, Y., Zhang, M., Wu, M., Wang, H., Tao, C., Singh, B., Rhoades, A. M., Qin, Y., Li, H.-Y., Feng, Y., Zhang, Y., Zhang, C., Zender, C. S., Xie, S., Roesler, E. L., Roberts, A. F., Mametjanov, A., Maltrud, M. E., Keen, N. D., Jacob, R. L., Jablonowski, C., Hughes, O. K., Forsyth, R. M., Di Vittorio, A. V., Caldwell, P. M., Bisht, G., McCoy, R. B., Leung, L. R., and Bader, D. C.: The fully coupled regionally refined model of E3SM version 2: overview of the atmosphere, land, and river results, Geoscientific Model Development, 16, 3953–3995, https://doi.org/10.5194/gmd-16-3953-2023, 2023.

Trenberth, K. E., Berry, J. C., and Buja, L. E.: Vertical interpolation and truncation of model-coordinate data, National Center for Atmospheric Research, Climate and Global Dynamics Division, 1993.

Zhang, J., Bogenschutz, P., Tang, Q., Cameron-smith, P., and Zhang, C.: Leveraging regional mesh refinement to simulate future climate projections for California using the Simplified Convection-Permitting E3SM Atmosphere Model Version 0, Geoscientific Model Development, 17, 3687–3731, https://doi.org/10.5194/gmd-17-3687-2024, 2024.