Scripps Coupled Ocean-Atmosphere Regional (SCOAR) Model

User Manual

Dian Putrasahan, Art Miller and Hyodae Seo

This manual shows how to install, set it up and run a simple case.

SCOAR has 3 main components: RSM, ROMS and the SST-flux coupler.
For help with RSM, you can refer to http://g-rsm.wikispaces.com/
For help with ROMS, you can refer to https://www.myroms.org/wiki/index.php/Documentation_-Portal

Check that you have C compiler, Fortran90 compiler and adequate space in your system. Other programs and toolbox you may want to have for verification purposes and/or to prepare the files before running the coupled model would include Matlab(R), ROMS toolbox (IRD and Manu's toolbox), GrADS, ncview.

For the purposes of this manual, the colors are coded as follows:
Red is for emphasis
Blue for commands in unix/linux environment
Brown for commands/statements in vi editor
Green for commands in Matlab or GrADS
Black for normal text or comments
Some fortran code in vi mode are in black for easier reading with indentations.

Let's set up and place SCOAR in a directory called Coupled_RsmRoms. There are 5 main folders for SCOAR (Lib/, Model/, Run/, Shell/, log/).

- Lib/ contains any auxiliary files, codes for the coupler, all executable files, and all grid preparation files.
- Model/, as it the name suggests, holds all the source code for ROMS and RSM.
- Shell/ contains all the shell scripts for calling each executable at the appropriate times.
- log/ keeps all the standard outputs that come from running the shell scripts.
- Once SCOAR starts running, everything is placed and carried out in the Run/ folder.

mkdir Couple_RsmRoms
cd Couple_RsmRoms
mkdir Lib/ Model/ Run/ Shell/ log/
cd Lib
mkdir aux-files/ codes/ exec/ grids/
cd exec
mkdir Coupler/ RSM/ ROMS/
cd ../../Model
mkdir ROMS/ RSM/ misc/

1. Download and Installation

Check that netcdf has been installed. If not, go to http://www.unidata.ucar.edu/software/netcdf/ and download netcdf-version.tar file.
tar xvf netcdf-3.6.1.tar
cd netcdf-3.6.1/src
./configure –prefix=/home/username/netcdf/
make check
make install
Set environment for NETCDF

a. RSM

You would need CVS in your system to download RSM. You can check this by typing cvs. If CVS is not installed in your system, go to http://www.nongnu.org/cvs/ and have it installed.

- Set your environment for CVSROOT, example
  setenv CVSROOT :pserver:anoncvs@rokka.ucsd.edu:/rokka1/kana/cvs-server-root/cpscvs
  Note that password for user "anoncvs" is <return>
- Be sure to add current directory "." to your path in .cshrc or equivalent
- cd Couple_RsmRoms/Model/RSM
- cvs co INSTALL
  Creates several different directories (CVS/, def/, expscr/ instscr/ makefiles/) and install executable.
  Interactive installation, proceed as follow.
- ./install
  Install G-RSM library, source and run scripts from cvs repository
  The default version of this run is: trunk
  ./install –help for help. PRESS <ENTER> TO PROCEED.

- <return>

  Model choices

  gsm rsm crsm nhm roms scm gdas cpl ? [return=gsm]==>
- rsm

  Machine architecture:

  Type single/thread/mpi/hybrid [return=single]?==>
- mpi

  Institution

  Enter INSTITUTION, ? for list, [return=sio] ==>? sio, ecpc, esc, navo, ncar, ncep, ncsa, nersc, tacc, sdsc or any name [return=ecpc]==>
- sio

  Machine CPU type

  Type MACHINE, ? for list [return=linux].==>?

  sgi origin ibmsp sun dec nec hp cray t90 t3e es linux mac ==>
- linux

  Machine nickname

  Type the machine nickname [return=atlas] ==> atlas

  Compiler

  Enter COMPILER, ? for list [return=pgi]. ==>?

  pgi or intel ==>
- pgi

  Additional options:

  If you need to change

  mpi-version=1

  compile-in-64-bit=no

  linux-distribution=1

  linux-vsn=0

  netcdf-vsn=3

  debug option=no

  username=m0101

  enter yes, else press enter or no==>
- <return>

  Model resolution and domain specification

  (1) through (43) options

  Type the number associated with your model dimension specification.

  If there is no choice available, type 999 [return=36]==>
- 9

  Number of pes

  rsm: type number of cpus to use [return=4]==>

- 64
  Choice of run script:
  rsm: Which script do you want to run?
  rcases rsim rsm rsmrt test [return=test]==>
- rsm
  Creates libs/, rsm/, rsm_runs/, .hist and tmp.14054
  You have compiled RSM!!!

Node list in your home directory is important.
cd
vi nodelist

Files that need to be added to RSM for coupler to work.
cd ~/Couple_RsmRoms/Model/RSM/expscr/rsm
scp user@home.comp:~/Research/SCOARv2/rsim_scoar.in ./
cd ~/Couple_RsmRoms/Model/RSM/libs/etc
scp user@home.comp:~/Research/SCOARv2/ieee2grb_sst_scoar.in ./
cd ~/Couple_RsmRoms/Model/RSM/libs/etc/utils
scp user@home.comp:~/Research/SCOARv2/ieee2grb_sst_scoar.F ./

Compile added codes:
cd ~/Couple_RsmRoms/Model/RSM/libs/etc/utils
vi Makefile.in
EXECS = date.x ifdef.x ieee2grb_sst.x ieee2grb_sst_scoar.x mpiset.x incdte.x
cd ../..
configure-libs
cd etc/utils
make

Adjustment to allow RSM to use ROMS SST at time t=0:
cd ~/Couple_RsmRoms/Model/RSM/rsm/src/rsml
vi rsmsmf.F
#Line 215: insert:

```
c
c Kei & Dian addition (May 13, 2009)
c Attempting to call sfc.F so as to use the sfc field at beginning of integration
c
#ifdef MP
       if (mype.eq.master) then
#endif
          print *,' **** surface update *** fhour  is ',fhour
```

```
          print *,' **** surface update *** idate  is ',idate
#ifdef MP
        endif
#endif
c       thour = 0.
        call sfc(nrsmo1,idate(4),idate(2),idate(3),idate(1),fhour)
c
c end of addition (May 13, 2009)
c
```

cd ../..
make

b. ROMS

In order to download ROMS, you would need to register with the ROMS community (http://www.myroms.org/). Once you have a username and password, ROMS maybe downloaded using SVN.

- svn checkout –username <username> https://www.myroms.org/svn/src/trunk ~/Couple_-RsmRoms/Model/ROMS/
  Authentication realm: <https://www.myroms.org:443> Subversion Repository
  Password for <username>
- Type in your password and download of ROMS will commence.
  Directories created: Atmosphere/, Compilers/, Data/, Lib/, Master/, ROMS/, .svn/, User/, Waves/, makefile
- Ensure that your netcdf path and mpif90 location is correct.
  cd Compilers
  vi Linux-pgi.mk
  ifdef USE_NETCDF4
  NETCDF_INCDIR ?= /opt/pgisoft/netcdf4/include
  NETCDF_LIBDIR ?= /opt/pgisoft/netcdf4/lib
  HDF5_LIBDIR ?= /opt/pgisoft/hdf5/lib
  else
  NETCDF_INCDIR ?= /share/apps/netcdf/pgi/include
  NETCDF_LIBDIR ?= /share/apps/netcdf/pgi/lib
  endif
  ifdef USE_MPIF90
  FC := /share/apps/mpich1/pgi/bin/mpif90
  LD := $(FC)
  else

LIBS += -Bdynamic -lfmpi-pgi -lmpi-pgi -Bstatic
endif

- Turn on MPI and 64-bit setting
  cd ..
  vi makefile
  USE_MPI ?= on
  USE_MPIF90 ?= on
  USE_LARGE ?= on
  make

c. Coupler

Need to copy over shell scripts, coupler code, and auxiliary files.

- cd ~/Couple_RsmRoms/Shell
  scp user@home.comp:~/Research/SCOARv2/102910_Shell.tar ./
  tar xvf 102910_Shell.tar
- cd ../Lib/aux-files
  scp user@home.comp:~/Research/SCOARv2/090808_aux.tar ./
  tar xvf 090808_aux.tar
- cd ../codes
  scp user@home.comp:~/Research/SCOARv2/102910_code.tar ./
  tar xvf 102910_code.tar
- Make sure path and directories in compilecode.sh is correct, then compile.
  compilecode.sh

2. Grid Set-up

The domain is to be created in RSM first, since there are certain requisites that must be met for RSM to run smoothly with a given grid. In choosing the number of grid points, certain contraints apply.

- igrd (x-direction) should be a product of the integer powers of 2,3 and 5. Also, the integer power of 3 cannot be more than 2.
- jgrd (y-direction) must be an odd number.

To learn more details on this, go to http://g-rsm.wikispaces.com/+Choosing+a+regional+domain

For the purpose of this study, let's take the case for the Humboldt Current System.

a. I) Setting up domain in RSM (LINUX MACHINE)

For the Humboldt Current System (hcs), we want to include 3 upwelling centers (20-22S,32-34S,36-38S). You would want to consider sponge layer when you create your domain. So say you want 5S - 40S, then you'd make your domain from 0S to 45S. We'd like to get at least 600km offshore, that would put it to 90W.

cd ~/Couple_RsmRoms/Model/RSM
cd rsm
vi define.h
#define _igrd_ 128
#define _jgrd_ 237
cd ../rruns
vi rsm
RPROJ=0.
RTRUTH=-20.
RORIENT=-80.
RDELX=20000.
RDELY=20000.
RCENLAT=-20.
RCENLON=-80.
RLFTGRD=63.
RBTMGRD=142.
prmap rsm
ga-> d lat
ga-> quit

Check the domain. Keep changing your number of grid points, resolution, location of cenlat and cenlon, and number of grid points from left and bottom, till you get the domain you desire.

view prmap.ctl
xdef 129 linear 268.037 0.191
ydef 238 levels
-42.842 -42.701 -42.561 .....

b. II) Setup domain in ROMS and RSM (local computer)

Here we create grid.nc files. Note that grdname must NOT contain "-" for the region part. Also note that roms and rsm grid is the same. This is part of the preparation for ROMS spin up run. We will use a mix of ROMS toolboxes (IRD toolbox and Manu's toolbox).

- Set parameters and file names
  cd ~/ROMS/IRD_toolbox/Roms_tools

```
cp -r Preprocessing_tools/ Preprocessing_tools_hcs/
cd !$
vi romstool_param.m
title = 'Humboldt Current System - Ocean Spin Up ';
config = 'hcs';
% values for dl, lonr and latr are taken from prmap.ctl
dl = 0.191;
lonr=[268.037:0.191:268.037+0.191*(129-1)];
latr=[-42.842 -42.701 -42.561 ...
N=30;
theta_s=6.5;
theta_b=0.;
hc=10.;
hmin=75;
vi make_grid.m
grdname='hcs-grid.nc';
vi make_forcing.m
title=['Forcing (COADS)'];
grdname='hcs-grid.nc';
frcname='hcs-forc.nc';
vi make_clim.m
title='Climatology';
grdname='hcs-grid.nc';
frcname='hcs-forc.nc';
clmname='hcs-clim.nc';
ininame='hcs-init.nc';
oaname='hcs-oa.nc';
```

- Preparing the files using matlab.

```
matlab
make_grid
close all
make_forcing
close all
make_clim
close all
exit
```

- Set information of grids in rnt_gridinfo.m . This is to ensure that matlab would search for the right grid files. 3 cases must be provided: gridname = 'hcs', model_resolution_gridname = 'roms20_hcs' and 'rsm20_hcs'.

```
cd ~/ROMS/Manu_toolbox/matlib/rnt/
vi rnt_gridinfo.m
case 'hcs'
gridindo.id = gridid;
gridindo.name = 'Humboldt Current System Ocean Spinup 20km';
gridindo.grdfile = '~/ROMS/IRD_toolbox/Roms_tools/Preprocessing_tools_hcs/hcs-grid.nc';
gridindo.N = 30;
gridindo.thetas = 6.5;
gridindo.thetab = 0.0;
gridindo.tcline = 75;
gridindo.hc = 10;
gridindo.cstfile = '~/ROMS/Manu_toolbox/matlib/rgrd/rgrd_WorldCstLinePacific.mat';
case 'roms20_hcs'
gridindo.id = gridid;
gridindo.name = 'HCS ROMS 20km';
gridindo.grdfile = '~/Research/SCOARv2/Lib/grids/hcs/ROMS/roms20_hcs-grid.nc';
gridindo.N = 30;
gridindo.thetas = 6.5;
gridindo.thetab = 0.0;
gridindo.tcline = 75;
gridindo.hc = 10;
gridindo.cstfile = '~/ROMS/Manu_toolbox/matlib/rgrd/rgrd_WorldCstLinePacific.mat';
case 'rsm20_hcs'
gridindo.id = gridid;
gridindo.name = 'HCS RSM 20km';
gridindo.grdfile = '~/Research/SCOARv2/Lib/grids/hcs/RSM/rsm20_hcs-grid.nc';
gridindo.N = 30;
gridindo.thetas = 6.5;
gridindo.thetab = 0.0;
gridindo.tcline = 75;
gridindo.hc = 10;
gridindo.cstfile = '~/ROMS/Manu_toolbox/matlib/rgrd/rgrd_WorldCstLinePacific.mat';
```

- Create climatology, boundary and initial files for ROMS spin up run.

```
cd ~/ROMS/IRD_toolbox/Roms_tools/Preprocessing_tools_hcs
matlab
grd=rnt_gridload('hcs');
indir='~/ROMS/IRD_toolbox/Roms_tools/Preprocessing_tools_hcs/';
nameof='hcs';
clmfile=[indir,nameof,'-clim.nc'];
```

```
bryfile=[indir,nameof,'-bry.nc'];
initfile=[indir,nameof,'-init.nc'];
forcfile=[indir,nameof,'-forc.nc'];
rnc_CreateBryFile(grd,bryfile);
rnc_SetBryFromClim(grd,clmfile,bryfile);
rnc_CreateIniFile(grd,initfile);
% If you want to set initial month to June,
imon=6;
rnc_SetInitFromClim(grd,clmfile,initfile,imon);
exit
```

3. ROMS spin-up

A spin-up for the ocean model is required in order for the ocean state to be stable. Typically, we would give it a 10 year spin-up run. The atmosphere itself only requires about 2 days for spin up.

- Create folder for ROMS application (LINUX machine)
  ```
  cd ~/Couple_RsmRoms/Model/ROMS/
  mkdir spinup_hcs
  cp -r Compilers/ spinup_hcs/
  cp -r Master/ spinup_hcs/
  cp -r ROMS/ spinup_hcs/
  cp makefile spinup_hcs/
  cp ~/node_list spinup_hcs/
  ```
- Modify makefile provide a name for ROMS application
  ```
  cd spinup_hcs
  vi makefile
  ROMS_APPLICATION ?= SPINUP_HCS
  ```
- Create "apps.h" file to specify all the definitions for the ROMS application. Listing of definitions that can be used are located in ~/Couple_RsmRoms/Model/ROMS/ROMS/Include/cppdefs.h. Many examples of "apps.h" are located in the same folder. Below is a sample for this case study.
  ```
  cd ROMS/Include
  vi spinup_hcs.h
  #define NL_MODEL
  #ifdef NL_MODEL
  #undef ADJOINT
  #undef TANGENT
  #endif
  ```

```
#define UV_ADV
#define UV_COR
#define UV_VIS2
#define UV_LDRAG
#undef MIX_GEO_UV
#define MIX_S_UV
#define TS_U3HADVECTION
#define TS_C4VADVECTION
#define TS_DIF2
#undef TS_DIF4
#undef MIX_GEO_TS
#define MIX_S_TS
#define DJ_GRADPS
#define SALINITY
#define QCORRECTION
#define SCORRECTION
#define NONLIN_EOS
#define CURVGRID
#define MASKING
#define SOLVE3D
#define SPLINES
#define OUT_DOUBLE
#define AVERAGES
#define AVERAGES_FLUXES
#ifdef NL_MODEL
#define LMD_MIXING
#undef MY25_MIXING
# ifdef LMD_MIXING
# undef DIURNAL_SRFLUX
# define SOLAR_SOURCE
# define LMD_RIMIX
# define LMD_CONVEC
# define LMD_NONLOCAL
# define LMD_SKPP
# endif
#else
#undef LMD_MIXING
#endif
#undef CLIM_NUDGING
```

```
#undef CLOSED_OBC
#ifndef CLOSED_OBC
#define EASTERN_WALL
#undef NORTHERN_WALL
#undef SOUTHERN_WALL
#undef WESTERN_WALL
#endif
#undef CLAMPED_BC
#ifndef CLAMPED_BC
# define SPONGE
# define WEST_VOLCONS
# define SOUTH_VOLCONS
# define NORTH_VOLCONS
# define RADIATION_2D
# define WEST_FSGRADIENT
# define WEST_M2RADIATION
# define WEST_M2NUDGING
# define WEST_M3RADIATION
# define WEST_M3NUDGING
# define WEST_TRADIATION
# define WEST_TNUDGING
# define NORTH_FSGRADIENT
# define NORTH_M2RADIATION
# define NORTH_M2NUDGING
# define NORTH_M3RADIATION
# define NORTH_M3NUDGING
# define NORTH_TRADIATION
# define NORTH_TNUDGING
# define SOUTH_FSGRADIENT
# define SOUTH_M2RADIATION
# define SOUTH_M2NUDGING
# define SOUTH_M3RADIATION
# define SOUTH_M3NUDGING
# define SOUTH_TRADIATION
# define SOUTH_TNUDGING
#endif
#define ANA_BSFLUX
#define ANA_BTFLUX
```

- Ensure that sponge layer is turned on and applied to the case.

cd ../Functionals

vi ana_hmixcoef.h

```
#ifdef SPONGE
 !
 !-----------------------------------------------------------------------
 !  Increase horizontal mixing in the sponge areas.
 !-----------------------------------------------------------------------
 !
 !! User modifiable section.  Please specify the appropiate sponge area
 !! by increasing its horizontal mixing coefficients.
 !!
 # if defined ADRIA02
 # elif defined SPINUP_HCS
 !
 !  HUMBOLDT CURRENT SYSTEM , taken from
 !  Southern California Bight sponge areas.
 !  }}
       fac=4.0_r8
  #  if defined UV_VIS2
 !SOUTH
       DO j=JstrR,MIN(6,JendR)
         cff=visc2(ng)+REAL(6-j,r8)*(fac*visc2(ng)-visc2(ng))/6.0_r8
        DO i=IstrR,IendR
           visc2_r(i,j)=cff
           visc2_p(i,j)=cff
         END DO
       END DO
 !NORTH
       DO j=MAX(JstrR,Mm(ng)+1-6),JendR
         cff=fac*visc2(ng)+                                              &
     &       REAL(Mm(ng)+1-j,r8)*(visc2(ng)-fac*visc2(ng))/6.0_r8
         DO i=IstrR,IendR
           visc2_r(i,j)=cff
           visc2_p(i,j)=cff
         END DO
       END DO
 !WEST
       DO i=IstrR,MIN(6,IendR)
         DO j=MAX(JstrR,i),MIN(Mm(ng)+1-i,JendR)
```

```
            cff=visc2(ng)+REAL(6-i,r8)*(fac*visc2(ng)-visc2(ng))/6.0_r8
            visc2_r(i,j)=cff
            visc2_p(i,j)=cff
          END DO
        END DO
#  endif
#  if defined TS_DIF2
        DO j=JstrR,MIN(6,JendR)
          cff1=tnu2(itemp,ng)+                                          &
     &         REAL(6-j,r8)*(fac*tnu2(itemp,ng)-tnu2(itemp,ng))/6.0_r8
          cff2=tnu2(isalt,ng)+                                          &
     &         REAL(6-j,r8)*(fac*tnu2(isalt,ng)-tnu2(isalt,ng))/6.0_r8
          DO i=IstrR,IendR
            diff2(i,j,itemp)=cff1
            diff2(i,j,isalt)=cff2
            DO itrc=3,NT(ng)
              diff2(i,j,itrc)=cff1*4.0_r8
            ENDDO
          END DO
        END DO
        DO j=MAX(JstrR,Mm(ng)+1-6),JendR
          cff1=fac*tnu2(itemp,ng)+                                      &
     &         REAL(Mm(ng)+1-j,r8)*(tnu2(itemp,ng)-                     &
     &                              fac*tnu2(itemp,ng))/6.0_r8
          cff2=fac*tnu2(isalt,ng)+                                      &
     &         REAL(Mm(ng)+1-j,r8)*(tnu2(isalt,ng)-                     &
     &                              fac*tnu2(isalt,ng))/6.0_r8
          DO i=IstrR,IendR
            diff2(i,j,itemp)=cff1
            diff2(i,j,isalt)=cff2
           DO itrc=3,NT(ng)
              diff2(i,j,itrc)=cff1*4.0_r8
            ENDDO
          END DO
        END DO
        DO i=IstrR,MIN(6,IendR)
          DO j=MAX(JstrR,i),MIN(Mm(ng)+1-i,JendR)
            cff1=tnu2(itemp,ng)+                                        &
     &           REAL(6-i,r8)*(fac*tnu2(itemp,ng)-tnu2(itemp,ng))/6.0_r8
```

```
            cff2=tnu2(isalt,ng)+                                          &
     &          REAL(6-i,r8)*(fac*tnu2(isalt,ng)-tnu2(isalt,ng))/6.0_r8
            diff2(i,j,itemp)=cff1
            diff2(i,j,isalt)=cff2
           DO itrc=3,NT(ng)
              diff2(i,j,itrc)=cff1*4.0_r8
            ENDDO
          END DO
        END DO
#  endif
```

- Now to make the ROMS executable file, oceanM will be created in ∼/Couple_RsmRoms/Model/ROMS/spinup_hcs.
  cd ../..
  make
- Create input file (ocean_spinuphcs.in) to ensure that ROMS takes in all the right values, according the the domain, model and cpu specifications. Myriad examples can be found in ∼/Couple_RsmRoms/Model/ROMS/ROMS/External/ . Take one for them and modify accordingly.
  cd ∼/Couple_RsmRoms/Model/ROMS/spinup_hcs
  cp ∼/Couple_RsmRoms/Model/ROMS/ROMS/External/ocean_upwelling.in ./ocean_spinuphcs.in
  vi ocean_spinhcs.in
  TITLE = HUMBOLDT CURRENT SYSTEM
  MyAppCPP = SPINUP_HCS
  VARNAME = ROMS/External/varinfo.dat

  Lm == 127 ! Number of I-direction INTERIOR RHO-points
  Mm == 236 ! Number of J-direction INTERIOR RHO-points
  N == 30 ! Number of vertical levels

  NtileI == 8 ! I-direction partition
  NtileJ == 8 ! J-direction partition

  NTIMES == 518400
  DT == 600.0d0
  NDTFAST == 30

  LDEFOUT == T
  NHIS == 4320
  NDEFHIS == 51840

NTSAVG == 1
NAVG == 4320
NDEFAVG == 51840

TNU2 == 20.0d0 20.0d0 ! m2/s
TNU4 == 0.0d0 0.0d0 ! m4/s
VISC2 == 5.0d0 ! m2/s
VISC4 == 0.0d0 ! m4/s

BLK_ZQ == 2.0d0 ! air humidity
BLK_ZT == 2.0d0 ! air temperature

THETA_S == 6.5d0 ! 0 < THETA_S < 20
THETA_B == 0.0d0 ! 0 < THETA_B < 1
TCLINE == 75.0d0 ! m

DSTART = 15.0d0 ! days
TIDE_START = 0.0d0 ! days
TIME_REF = -1.0d0 ! yyyymmdd.dd

TNUDG == 30.0d0 30.0d0 30.0d0 30.0d0 30.0d0 30.0d0 ! days
ZNUDG == 30.0d0 ! days
M2NUDG == 30.0d0 ! days
M3NUDG == 30.0d0 ! days

OBCFAC == 10.0d0 ! nondimensional
GAMMA2 == -1.0d0

GRDNAME == hcs-grid.nc
ININAME == hcs-init.nc
CLMNAME == hcs-clim.nc
BRYNAME == hcs-bry.nc

NFFILES == 1 ! number of forcing files
FRCNAME == hcs-forc.nc ! forcing file 1, grid 1
RSTNAME == rst.nc
HISNAME == his.nc
AVGNAME == avg.nc

- Transfer of files from home computer to cluster (home computer)

cd ~/ROMS/IRD_toolbox/Roms_tools/Preprocessing_tools_hcs
scp hcs-grid.nc user@comp.cluster:~/Couple_RsmRoms/Model/ROMS/spinup_hcs/.
scp hcs-forc.nc user@comp.cluster:~/Couple_RsmRoms/Model/ROMS/spinup_hcs/.
scp hcs-init.nc user@comp.cluster:~/Couple_RsmRoms/Model/ROMS/spinup_hcs/.
scp hcs-bry.nc user@comp.cluster:~/Couple_RsmRoms/Model/ROMS/spinup_hcs/.

- Initiate spin up run of ROMS (LINUX machine)
  cd ~/Couple_RsmRoms/Model/ROMS/spinup_hcs
  vi launch64
  /share/apps/mpich1/pgi/bin/mpirun -nolocal -np 64 -machinefile node_list oceanM ocean_-spinuphcs.in
  launch64 >& 1spinuphcs.log

4. Executables for RSM and ROMS

With the spin up running, we now prepare for the SCOAR run. Beginning with RSM.

a. RSM

- Create folders for RSM model that is specific to our case study.
  cd ~/Couple_RsmRoms/Model/RSM
  cp -r rruns/ hcs/
  cp -r rsm/ hcs/
  cd hcs
  mv rruns/ runs_64cpu/
  mv rsm/ rsm_64cpu/
  cd rsm_64cpu/def
  cp -r rsm6228r160199/ rsm6228r128237/
- Specifying number of grid points for domain
  cd ~/Couple_RsmRoms/Model/RSM/hcs/rsm_64cpu/def/rsm6228r128237/
  vi define.h
  #define _igrd_ 128
  #define _jgrd_ 237
- Specifying definitions for the physics package.
  cd ~/Couple_RsmRoms/Model/RSM/hcs/rsm_64cpu/def/
  vi physics.h
  #define CLDSLINGO /* Slingo cloud scheme */
  #define INSA_CLD /* Insa's cloudiness adjustment to Slingo scheme */
  #undef CLDADJ /* empirical cloudiness adjustment */
  #define INTERACTIVE_STRATUS /* cloudiness based on cloud water and RH */

/* only applicable for predicted cloud water */
#define ICECLOUD /* cloud water feed into radiation */
#define RAS /* Relaxed Arakawa Schubert scheme */
#define CLD3 /* Song-You Hong prognostic qc/qi, qr/qs */
#define NEW_MPHYS /* additional option for CLD3/CLD5/CLDWSMMPS */
#define VCI /* additional option for CLD3/CLD5/CLDWSMMPS */
#define DIFFQ /* horizontal diffusion of moisture */
#define DIFUC /* strong horizontal diffusion of cloud water variables to stabilize cloud water prediction schemes and allow for larger time step */
#undef RASC2 /* RAS with detarained cloud water like RASV2 */
vi misc.h
#undef STDAMP /* Spectral tendency damping scheme for rsm */
#undef STDAMP3 /* area average log ps correction to be used with stdamp */
#undef NFDR /* national fire danger rating system */
#define SPNG_WID /* wide (original) sponge zone in rsml/rltbini.F */
#undef SPNG_NRW /* narrow sponge zone in rsml/rltbini.F */

- Configure RSM model
  cd ~/Couple_RsmRoms/Model/RSM/hcs/rsm_64cpu
  make clean
  configure-model
- Build file directories in RSM that pertains to application case. While doing so, several things to check for. The grid must match and allow SST from ROMS to be used on the initial step (t=t0) in RSM.
- Edit makefile for source code
  cd src
  vi makefile
  GSM_PROGS = share co2 mtn chgr sfcl fcst pgb sgb sfc0 #p2sig
  RSM_PROGS = share rsml sfcl rgsm rmtn rinpg rfcst rpgb co2 sfc0 #p2sig mtn
  RESCON = co2 #mtn
- Check for matching grid, especially with rtruth ~=0
  cd sfcl
  vi rgetarea.F
  # Originally on line 53-60:

```
              delx=(rlon2-rlon1)/float(kgds(02)-1)
 c            delx=kgds(12)  ! dx (meter) on truth latitude
              dely=kgds(13)  ! dy (meter) on truth latitude
              orient=rlon1
              truth=kgds(09)*1.e-3 ! truth latitude
              cotru=truth          ! co-truth latitude
```

```
              delx=delx*cos(truth*rad)*rerth_*rad
# Modified to include rtruth=-20
 c              delx=(rlon2-rlon1)/float(kgds(02)-1)
              delx=kgds(12)  ! dx (meter) on truth latitude
              dely=kgds(13)  ! dy (meter) on truth latitude
              orient=rlon1
 c             truth=kgds(09)*1.e-3 ! truth latitude
              truth=-20.
              cotru=truth          ! co-truth latitude
 c             delx=delx*cos(truth*rad)*rerth_*rad
```

- Make RSM (set model up)
  cd ~/Couple_RsmRoms/Model/RSM/hcs/rsm_64cpu
  make
- Create RSM executable
  cd ~/Couple_RsmRoms/Model/RSM/hcs/runs_64cpu
  configure-scr rsim_scoar
- Account for change in time step in Southern Hemisphere, because seasons are in reference to Northern Hemisphere. (Only needed when doing SH). Winter requires shorter time steps because of the the storms passing through.
  vi rsim_scoar
  DELTAT_REG_SPRING=90
  DELTAT_REG_SUMMER=60
  DELTAT_REG_FALL=90
  DELTAT_REG_WINTER=100

b. ROMS

This is very similar to the ROMS spin-up set up, except the application name is now HCS instead of SPINUP_HCS. All conducted in the linux machine.

- Create folder for ROMS application
  cd ~/Couple_RsmRoms/Model/ROMS/
  mkdir hcs/
  cd spinup_hcs/
  cp -r Compilers/ ../hcs/
  cp -r Master/ ../hcs/
  cp -r ROMS/ ../hcs/
  cp makefile ../hcs/
  cd ..

cp launch64 hcs/

cp node_list hcs/

- Modify makefile provide a name for ROMS application

  cd hcs

  vi makefile

  ROMS_APPLICATION ?= HCS

- Create "apps.h" file.

  Similar to spinup_hcs.h, but with some adjustments, as noted below for this case study.

  cd ROMS/Include

  mv spinup_hcs.h hcs.h

  vi hcs.h

  #undef QCORRECTION

  #undef SCORRECTION

  #define BULK_FLUXES

  #define COOL_SKIN

  #define LONGWAVE_OUT

  #define EMINUSP

- Ensure that sponge layer is turned on and applied to the case.

  cd ../Functionals

  vi ana_hmixcoef.h

  Substitute SPINUP_HCS with HCS

  #ifdef SPONGE

  # if defined ADRIA02

  # elif defined HCS

- Now to make the ROMS executable file, oceanM will be created in ∼/Couple_RsmRoms/Model/ROMS/hcs

  cd ../..

  make

- Create input file (ocean_hcs.in)

  Also pretty similar to ocean_spinuphcs.in with some adjustments.

  cd ∼/Couple_RsmRoms/Model/ROMS/hcs

  cp ../spinup_hcs/ocean_spinuphcs.in ./ocean_hcs.in

  vi ocean_hcs.in

  TITLE = HUMBOLDT CURRENT SYSTEM

  MyAppCPP = HCS

  VARNAME = varinfo.dat

  Lm == 127 ! Number of I-direction INTERIOR RHO-points

  Mm == 236 ! Number of J-direction INTERIOR RHO-points

  N == 30 ! Number of vertical levels

```
NtileI == 8 ! I-direction partition
NtileJ == 8 ! J-direction partition

NTIMES == 144
DT == 600.0d0
NDTFAST == 30

NRREC == 0
LcycleRST == T
NRST == 145

LDEFOUT == T
NHIS == 145
NDEFHIS == 0
NTSAVG == 1
NAVG == 144
NDEFAVG == 0
NTSDIA == 1
NDIA == 145
NDEFDIA == 0

DSTART = 0.0d0 ! days
TIDE_START = 0.0d0 ! days
TIME_REF = 0.0d0 ! yyyymmdd.dd

GRDNAME == ocean_grd.nc
ININAME == ocean_ini.nc
CLMNAME == ocean_clm.nc
BRYNAME == ocean_bry.nc

NFFILES == 1 ! number of forcing files
FRCNAME == ocean_frc.nc ! forcing file 1, grid 1
RSTNAME == ocean_rst.nc
HISNAME == ocean_his.nc
AVGNAME == ocean_avg.nc

APARNAM = ~/Couple_RsmRoms/Model/ROMS/upwelling/ROMS/External/s4dvar.in
SPOSNAM = ~/Couple_RsmRoms/Model/ROMS/upwelling/ROMS/External/stations.in
```

FPOSNAM = ~/Couple_RsmRoms/Model/ROMS/upwelling/ROMS/External/floats.in
BPARNAM = ~/Couple_RsmRoms/Model/ROMS/upwelling/ROMS/External/bioFasham.in
SPARNAM = ~/Couple_RsmRoms/Model/ROMS/upwelling/ROMS/External/sediment.in
USRNAME = ~/Couple_RsmRoms/Model/ROMS/upwelling/ROMS/External/MyFile.dat

5. Files containing grid information for ROMS, RSM and coupler

- Create grid.nc files for ROMS and RSM (local machine)
  cd ~/Research/SCOARv2/Lib/grids
  mkdir hcs
  cd hcs
  mkdir Coupler/ matlab/ ROMS/ RSM/
  grid.nc files the same as the one created previously (hcs-grid.nc), since grid for ocean and atmosphere is the same. Note that grid.nc files are named as "model-resolution_gridname-grid.nc".
  cp ~/ROMS/IRD_toolbox/Roms_tools/Preprocessing_tools_hcs/hcs-grid.nc ~/Research/SCOARv2/Lib/gri
  hcs-grid.nc cp ~/ROMS/IRD_toolbox/Roms_tools/Preprocessing_tools_hcs/hcs-grid.nc ~/Research/SCOAR
  hcs-grid.nc
- Create associated files (grid info) for ROMS and RSM
  Since we'll use Manu's toolbox for this, check that ~/ROMS/Manu_toolbox/matlib/rnt/rnt_-gridinfo.m contains the correct locations of the grid.nc files
  cd ~/Research/SCOARv2/Lib/grids
  cp gen_grid.m hcs/Coupler/
  cp gridmask.m hcs/matlab/
  cd hcs/Coupler
  vi gen_grid.m
  cd ~/Research/SCOARv2/Lib/grids/hcs/Coupler
  Ensure that all are correctly named (model-resolution_gridname)
  Example in vi editor, you can do substitution of all "word1" into "word2"
  :1,$ s/rsm20_pcr/rsm20_hcs/g
  :1,$ s/roms10_pcr/roms20_hcs/g
  cd ../matlab
  vi gridmask.m
  cd ~/Research/SCOARv2/Lib/grids/hcs/matlab
  rgn1='rsm20_hcs';
  rgn2='roms20_hcs';
  cd ../Coupler
  matlab
  gen_grid

cd ../matlab
gridmask
exit

- Make text file for tiling purposes
  No tiling in this case, so skip this. But below is an example if needed.
  cd ~/Research/SCOARv2/Lib/grids/hcs
  vi hcs_tile.txt
  1
- Make text file (rgrd.dat) for input into ieee2grib_sst_scoar.f
  rgrd.txt contains xlon1a, xlon2a, xlat1a, xlat2a, delx, dely, ortru, idrt
  ortru -> real orientation of regional polar projection or truth for regional mercater projection (needs a '.' at the end)
  idrt -> integer data representation type (0=latlon, 1=mercater, 2=, 3=lambert, 4=gaussian, 5=polar)
  if lon1a=-95.695, then xlon1a=360+(-95.695)=264.305
  delx and dely are horizontal resolution in metres.
  cd ~/Research/SCOARv2/Lib/grids/hcs/RSM
  vi rgrd.dat
  268.037
  292.485
  -42.842
  -2.140
  20000
  20000
  -20.
  1

6. Initial, forcing and boundary files for ROMS and RSM

RSM only requires initial SST forcing from ROMS spin up run. Initial state and boundary conditions for RSM are provided by NCEP Reanalysis II (R2), making RSM a dynamical downscaling of NCEP. ROMS on the other hand can use boundary conditions from a variety of sources (including Levitus, SODA, OFES, etc). Initial state of ROMS can be taken from the spin up run. Forcing for ROMS comes from RSM, but dependent upon which options are chosen (bulk parameterization or not), RSM can provide the necessary variables needed for either computation.

- Create forcing and initial .nc file template for ROMS
  cd ~/Research/SCOARv2/Lib/preparerun/ROMS
  mkdir hcs/

```
cd !$
mkdir general_forcing/ coldstart/ bdry_clim/
cd ../..
cp create_bulk_forc_init.m ROMS/hcs/general_forcing
cp coldstart_init.m ROMS/hcs/coldstart
cp create_clim_bdry.m ROMS/hcs/bdry_clim
cd ROMS/hcs/general_forcing/
vi create_bulk_forc_init.m
cd ~/Research/SCOARv2/Lib/preparerun/ROMS/hcs/general_forcing
nameit='roms20_hcs';
nameit2='hcs';
matlab
create_bulk_forc_init
exit
```

- Create boundary files for ROMS

  Assumes climatology file was already made from SODA products and is named roms20_hcs-clim.nc

```
cd  /Research/SCOARv2/Lib/preparerun/ROMS/hcs/bdry_clim
vi create_clim_bdry.m
nameit='roms20_hcs';
nameit2='hcs';
%dataset=0; % 0=dataset for general clim and bdry that has been made from IRD toolbox
%dataset = 1; % 1=WOA2001 (Uses IRD toolbox)
dataset = 2; % 2=SODA (Uses Manu's toolbox)
%dataset = 3; % 3=Levitus (Uses Manu's toolbox)
BC_dir=['~/Research/SCOARv2/Lib/preparerun/ROMS/hcs/bdry_clim/'];
elseif dataset == 2
nameit='roms20_hcs';
BC_dir=['~/Research/SCOARv2/Lib/preparerun/ROMS/hcs/bdry_clim'];
grd=rnt_gridload(nameit);
clmfile=[nameit,'-clim.nc'];
bryfile=[nameit,'-bry.nc'];
matlab
create_clim_bdry
exit
```

- Create very first initial file for ROMS from spin up run

```
cd  /Research/SCOARv2/Lib/preparerun/ROMS/hcs/coldstart/
scp usr@comp.cluster:~/Couple_RsmRoms/Model/ROMS/spinup_hcs/avg_0010.nc ./
```

  One can use ncview to see which initial file to use, and which initial month to pick.

```
vi coldstart_init.m
cd ~/Research/SCOARv2/Lib/preparerun/ROMS/hcs/coldstart
grd=rnt_gridload('roms20_hcs');
initfile='./roms20_hcs-spinup.nc';
climfile='avg_0010.nc';
rnc_CreateIniFile(grd,initfile);
timeindex=3; %1 to length(t)=12 June-May, 3 is for September
matlab
coldstart_init
exit
```

7. Folders and files in their appropriate places

- Archive folders (local machine)

```
cd ~/storage
mkdir SCOAR_output
cd !$
mkdir hcs/
cd !$
mkdir ROMS/ RSM/
cd RSM
mkdir 2008/
cd ../ROMS
mkdir avg/ dia/ forc/ his/
```

- Create folders to place files needed for the SCOAR application run (LINUX machine)

```
cd ~/Couple_RsmRoms/Model/misc
mkdir hcs
cd ~/Couple_RsmRoms/Lib/grids
mkdir hcs/
cd !$
mkdir ROMS/ RSM/
cd ~/Couple_RsmRoms/Lib/exec
mkdir ROMS/hcs RSM/hcs
```

- Create folder that would contain the scripts for the application run.

```
cd ~/Couple_RsmRoms/Shell
mkdir hcs/
cp couple_Nday.sh hcs
cp Rsm2Roms_bulk.sh hcs
cp prepareROMS.sh hcs
```

cp uauo.sh hcs

cp Roms2Rsm.sh hcs

cp sst_t0.sh hcs

- Edit the driver script according to your application needs.

  cd ~/Couple_RsmRoms/Shell/main_couple

  cp main_couple.sh main_couple_hcs.sh

  vi main_couple_hcs.sh

  YYYYS=2008 #start year

  MMS=09 #start month

  DDS=15 #start day

  YYYYE=2008 #end year

  MME=12 #end month

  DDE=16 #end day

  RESTART=no #restart option

  LastNDay= #Model day for restart

  gridname=hcs #name of application

  rsmNCPU=64 #number of CPUs used for RSM

  romsNCPU=64 #number of CPUs used for ROMS

  Nameit_RSM=rsm20_hcs # model, resolution, grid name

  Nameit_ROMS=roms20_hcs # model, resolution, grid name

  BCFile=SODA #Boundary conditions for ocean model

  archive=yes #archive in local computer, not cluster

  archive_dir_rsm=~/storage/SCOAR_output/hcs/RSM

  archive_dir_roms=~/storage/SCOAR_output/hcs/ROMS

  nd=30 #number of vertical layers in ocean model

  SSTt0=yes #use ROMS SST from spin up run as initial forcing for RSM

  filterSST=no #smoothing of SST fields at each coupling

  needinterp=no #interpolation between ROMS and RSM grids

  tiling=no #tiling between ROMS and RSM grids

- NOTE: At this point, you should check the rest of the shell scripts to make sure you've accounted for any changes.
- Transfer grid information files

  cd ~/Couple_RsmRoms/Lib/grids

  scp -r user@home.comp:~/Research/SCOARv2/Lib/grids/hcs/ROMS ~/Couple_RsmRoms/Lib/grids/hcs

  scp -r user@home.comp:~/Research/SCOARv2/Lib/grids/hcs/RSM ~/Couple_RsmRoms/Lib/grids/hcs

  If tiling was used, then we must also transfer the required file. Example,

  scp user@home.comp:~/Research/SCOARv2/Lib/grids/hcs/hcs-tile.txt ~/Couple_RsmRoms/Lib/grids/hcs
- Transfer ROMS initial and template files

  cd ~/Couple_RsmRoms/Model/misc/hcs

sftp user@home.comp
cd ~/Research/SCOARv2/Lib/preparerun/ROMS/hcs/coldstart
get roms*-spinup.nc
cd ../general_forcing
mget *.nc
exit

- Transfer ROMS files to run ocean model
  cd ~/Couple_RsmRoms/Model/ROMS/hcs
  cp node_list ~/Couple_RsmRoms/Lib/exec/ROMS/hcs/
  vi launch64
  /share/apps/mpich1/pgi/bin/mpirun -nolocal -np 64 -machinefile node_list oceanM ocean.in
  cp launch64 ~/Couple_RsmRoms/Lib/exec/ROMS/hcs/launch64
  cp ocean_hcs.in ~/Couple_RsmRoms/Lib/exec/ROMS/hcs/ocean64_roms30_day_hcs.in
  cp oceanM ~/Couple_RsmRoms/Lib/exec/ROMS/hcs
- Transfer RSM executable file to run atmospheric model
  cd ~/Couple_RsmRoms/Model/RSM/hcs/runs_64cpu
  cp rsim_scoar ~/Couple_RsmRoms/Lib/exec/RSM/hcs/rsm_64cpu_day_hcs
- Create a log dir for keep your log files
  cd ~/Couple_RsmRoms/log
  mkdir hcs_log

8. Running SCOAR

Now at last we can run the model. Typically, you would want to run it on a background.
cd ~/Couple_RsmRoms/Shell/main_couple
main_couple_hcs.sh >& ../../log/hcs_log/log1 &