



Tuesday afternoon

1:46 ~

# Coupled model applications Tutorial



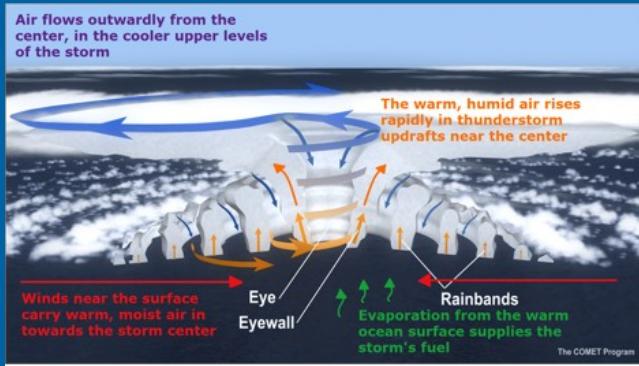
## Projects/Sandy- Hurricane Sandy example

- How does the coupled modeling system work (MCT)
- SCIP interpolation
- Setting up a coupled application

# Why do we couple models?

Usually treated  
independently but  
actually occur together

Atmosphere



Wave

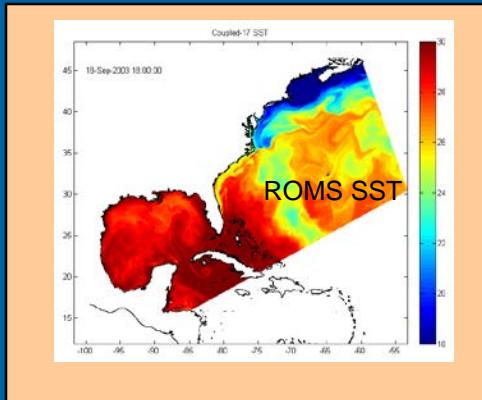
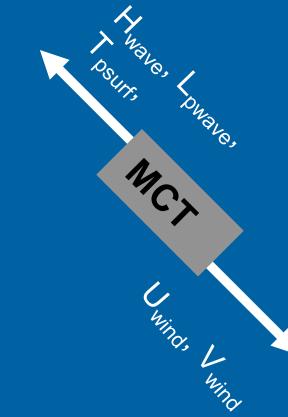
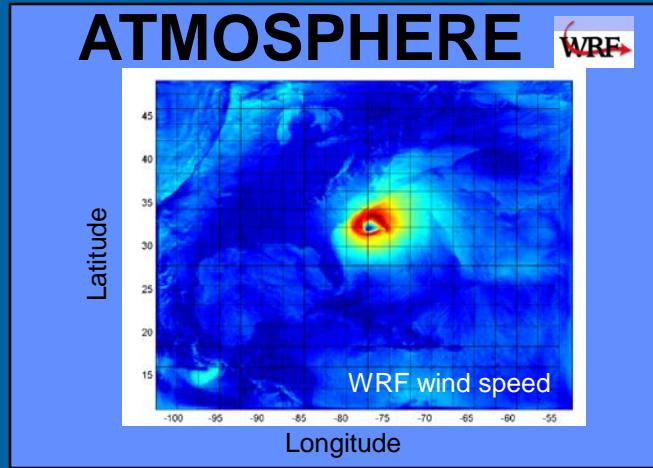


Ocean



Coastal  
geomorphology

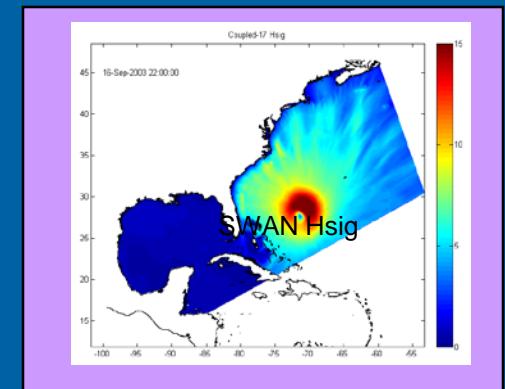
# Model setup



OCEAN



H<sub>wave</sub>, L<sub>mwave</sub>, L<sub>pwave</sub>, D<sub>wave</sub>,  
D<sub>wavep</sub>, T<sub>psurf</sub>, T<sub>mbott</sub>, Q<sub>b</sub>,  
Diss<sub>bot</sub>, Diss<sub>surf</sub>, Diss<sub>wcap</sub>,  
U<sub>bot</sub>



WAVE



SEDIMENT



Ice  
Biology  
Vegetation

# OCN interactions

Use consistent stress  
roms+wrf  
#define  
**ATM2OCN\_FLUXES**



Ustress, Vstress,  
Swrad, Lwrad  
LH, HFX

Use wrf vars in  
COARE  
#define  
**BULK\_FLUXES**



Uwind-ucur,  
Vwind-vcur,  
Swrad, Lwrad,  
RH, Tair,  
cloud



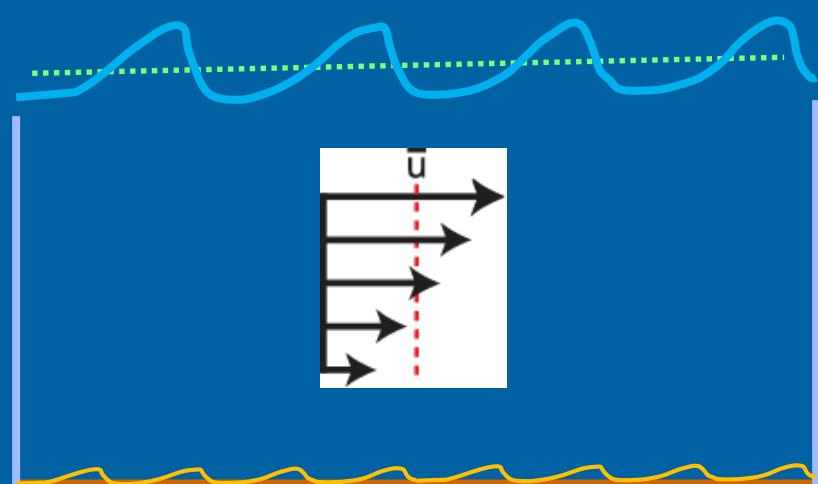
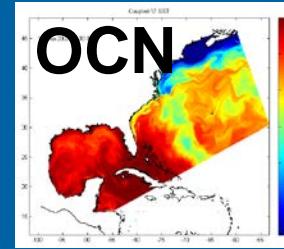
Atm press



**ATM**  
Uwind, Vwind, Patm, RH, Tair,  
cloud, rain, evap, SWrad, Lwrad  
LH, HFX, Ustress, Vstress

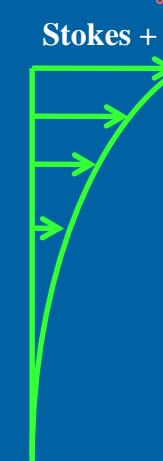
H<sub>wave</sub>, L<sub>mwave</sub>, L<sub>pwave</sub>, D<sub>wave</sub>,  
T<sub>psurf</sub>, T<sub>mbott</sub>, Q<sub>b</sub>,  
Diss<sub>bot</sub>, Diss<sub>surf</sub>, Diss<sub>wcap</sub>,  
U<sub>bot</sub>

**WAVE**



Water column

*vortex force*  
Stokes + VF

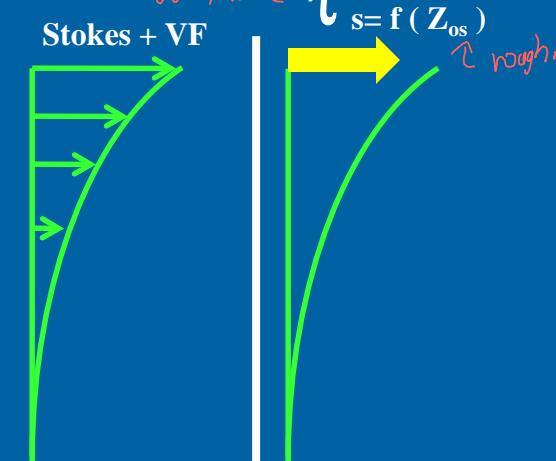


H<sub>wave</sub>, L<sub>mwave</sub>, D<sub>wave</sub>,  
T<sub>psurf</sub>, Q<sub>b</sub>,  
Diss<sub>bot</sub>, Diss<sub>surf</sub>,  
Diss<sub>wcap</sub>

Surface

$$\tau_s = f(Z_{os})$$

*↑ roughness*



Bottom

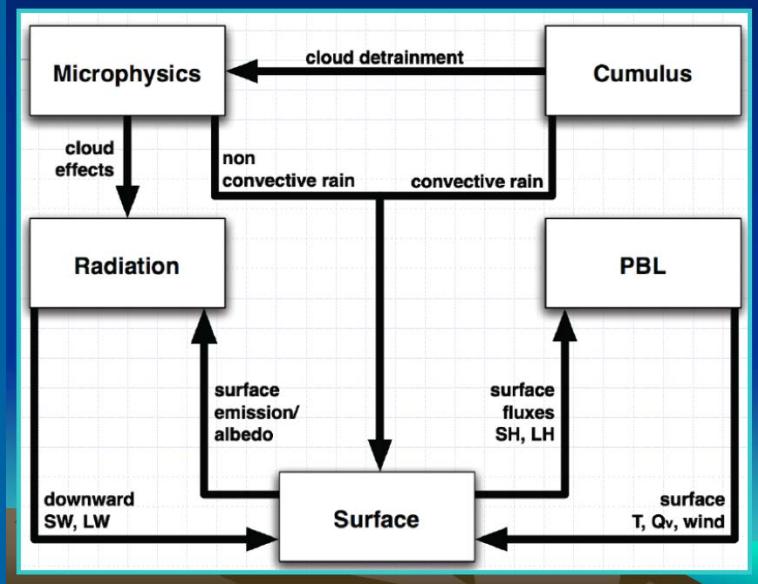
Zoa

$$\tau_b = f(Z_{ob})$$

H<sub>wave</sub>, L<sub>mwave</sub>,  
D<sub>wave</sub>,  
T<sub>mbott</sub>, U<sub>bot</sub>



# ATM interactions



Surface fluxes  
Momentum  
Heat  
Moisture

$$\begin{aligned} F_m &= C_m |\vec{V}_{SL}|^2, \\ F_h &= \rho_1 c_p C_{hq} (\theta_{sk} - \theta_1), \\ F_q &= \rho_1 L C_{hq} M (q_{vsk} - q_{v1}). \end{aligned}$$

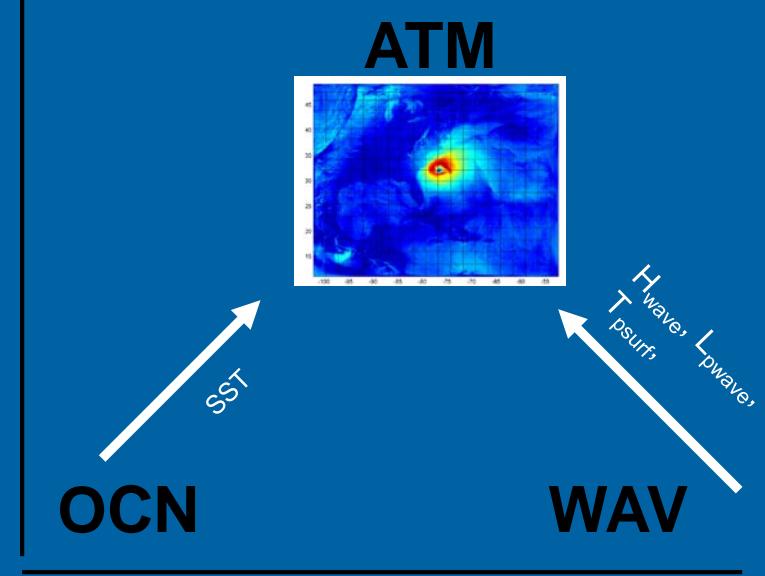
$$|\vec{V}_{SL}|^2 = u^2 + v^2$$

$C_m$  is the exchange coefficient for momentum and is expressed as

$$C_m = \frac{u_*^2}{|\vec{V}_{SL}|^2}.$$

$C_{hq}$  is the exchange coefficient valid for both heat and water vapor as

$$C_{hq} = u_* \kappa \left[ \psi_h \left( \frac{z}{L_{MO}} \right) - \psi_h \left( \frac{z_{0T}}{L_{MO}} \right) + \ln \left( \frac{z}{z_{0T}} \right) \right]^{-1},$$



SST  
OCN

$u_*$  is the friction velocity and is expressed as

$$u_* = \kappa |\vec{V}_{SL}| \left[ \psi_m \left( \frac{z}{L_{MO}} \right) - \psi_m \left( \frac{z_{0m}}{L_{MO}} \right) + \ln \left( \frac{z}{z_{0m}} \right) \right]^{-1}$$

$$z_{0m} = f(H_{wave}, L_{pwave}, T_{psurf})$$

WAV

## OCEAN SURFACE ROUGHNESS CLOSURE MODELS

笔记

CHARNOCK 1955

$$z_{0m} = \frac{0.011(u_*)^2}{g}$$

TAYLOR & YELLAND 2001: TY2001

$$\frac{z_{0m}}{H_s} = 1200 \left( H_s / L_p \right)^{4.5}$$

DRENNAN 2003: DGQH

$$\frac{z_{0m}}{H_s} = 3.35 \left( u_* / C_p \right)^{3.4}$$

OOST 2002: OOST

$$\frac{z_{0m}}{L_p} = \frac{25.0}{\pi} \left( u_* / C_p \right)^{4.5}$$

$H_s$  = significant wave height

$z_0$  = ocean surface roughness

$u_*$  = wind friction velocity

$C_p$  = peak wave celerity

$L_p$  = peak wave length

$\frac{u_*}{C_p}$  = wave age

- Wave steepness based parameterization.
- Based on three datasets representing sea-state conditions ranging from strongly forced to shoaling.

- Wave age based formula to characterize the ocean roughness.

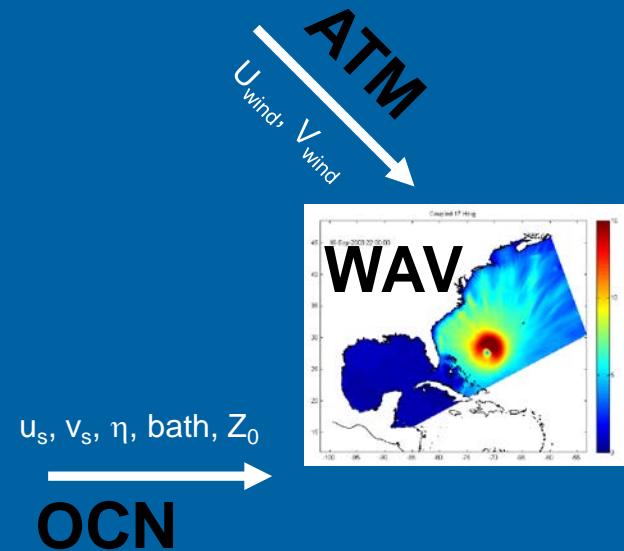
- They combined data from many field experiments representing a variety of condition and grouped the data as a function of the wind friction velocity.

- Wave age dependent formula but it also considers the effect of the wave steepness.

# WAV interactions

$$\frac{\partial N}{\partial t} + \frac{\partial c_x N}{\partial x} + \frac{\partial c_y N}{\partial y} + \frac{\partial c_\sigma N}{\partial \sigma} + \frac{\partial c_\theta N}{\partial \theta} = \frac{S_w}{\sigma}$$

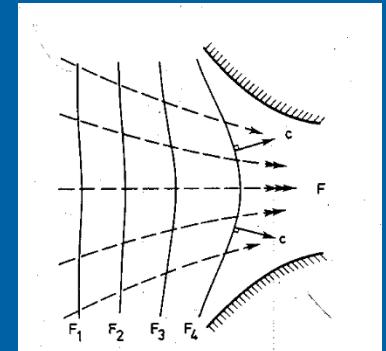
- 1) **Generation** – wind speed forcing is modified by ocean currents:  
 $S(w) = f(U_{wind} - u_s ; V_{wind} - v_s)$



## 2) Propagation

- wave celerity in geographic space is modified by ocean currents

$$c_x = c_{gx} + u_s ; c_y = c_{gy} + v_s$$



- change of wave direction (refraction) due to  $\eta$ , bathy, and currents:

$$C_{g,\theta} = \frac{\sigma}{\sinh(2kh)} \left( \frac{\partial h}{\partial x} \sin \theta - \frac{\partial h}{\partial y} \cos \theta \right) + \cos \theta \left( \frac{\partial U}{\partial x} \sin \theta - \frac{\partial U}{\partial y} \cos \theta \right) + \sin \theta \left( \frac{\partial V}{\partial x} \sin \theta - \frac{\partial V}{\partial y} \cos \theta \right)$$

# Coupled Modeling System

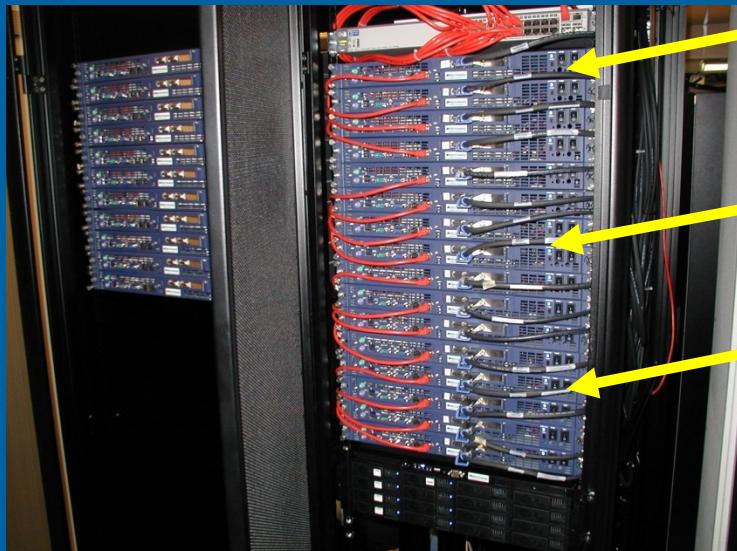
## Model Coupling Toolkit

Mathematics and Computer Science Division Argonne National Laboratory  
<http://www-unix.mcs.anl.gov/mct/>

MPI が必須  
WRF ビルドの前に MCT のインストールは  
完了しているはず

MCT is an open-source package that provides MPI based communications  
between all nodes of a distributed memory modeling component system.

Download and compile as libraries that are linked to.



Model A running on M nodes.

Model B running on N nodes.

Model C .....

.....



MCT provides  
communications  
between all models.

(it also works here)



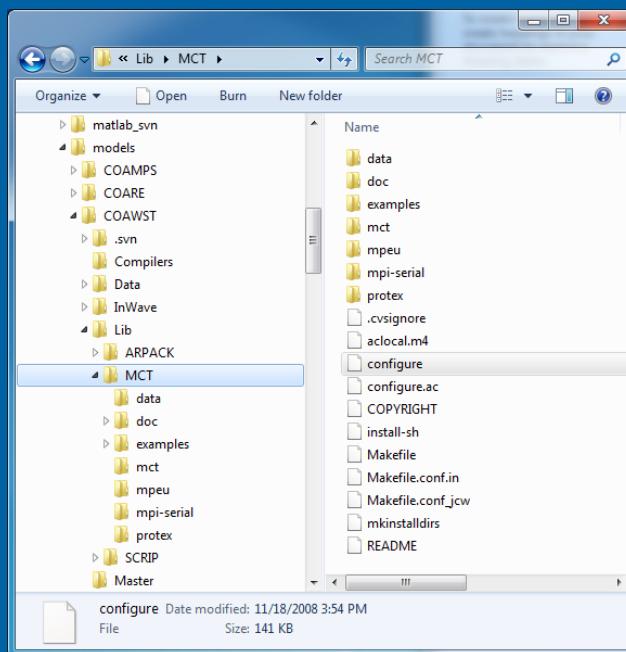
Warner, J.C., Perlin, N., and Skillingstad, E. (2008). Using the Model Coupling Toolkit to couple earth system models. Environmental Modeling and Software



# Libraries

/Lib/MCT/

## ■ MCT - v2.60 or higher (distributed)



1) cd to the MCT dir

2) ./configure

This makes Makefile.conf. you can edit this file.

3) make

4) make install

5) set environment vars

setenv MCT\_INCDIR COAWST/Lib/MCT/include

setenv MCT\_LIBDIR COAWST/Lib/MCT/lib

(or where ever you installed them, see last slide)

# Compilers dir (side note)

```
emacs: Linux-pgi.mk
File Edit View Cmds Tools Options Buffers Makefile Help
Open Dired Save Print Cut Copy Paste Undo Spell Replace Info Compile Debug News
Linux-pgi.mk |
#
# Library locations, can be overridden by environment variables.
#
ifdef USE_NETCDF4
    NETCDF_INCDIR ?= /opt/pgisoft/netcdf4/include
    NETCDF_LIBDIR ?= /opt/pgisoft/netcdf4/lib
        HDF5_LIBDIR ?= /opt/pgisoft/hdf5/lib
else
    NETCDF_INCDIR ?= /opt/pgisoft/netcdf/include
    NETCDF_LIBDIR ?= /opt/pgisoft/netcdf/lib
endif
    LIBS := -L$(NETCDF_LIBDIR) -lnetcdf -lnetcdff -L/opt/mx/lib64 -lcurl -lgssapi_krb5
    LIBS += $(shell nc-config --libs)
ifdef USE_NETCDF4
    LIBS += -L$(HDF5_LIBDIR) -lhdf5_hl -lhdf5 -lz
endif
ifdef USE_MCT
    MCT_INCDIR ?= /opt/pgisoft/mct/include
    MCT_LIBDIR ?= /opt/pgisoft/mct/lib
        FFLAGS += -I$(MCT_INCDIR)
        LIBS += -L$(MCT_LIBDIR) -lmct -lmpeu
endif
ifdef USE_ARPACK
ifdef USE_MPI
    PARPACK_LIBDIR ?= /opt/pgisoft/PARPACK
        LIBS += -L$(PARPACK_LIBDIR) -lparpack
endif
    ARPACK_LIBDIR ?= /opt/pgisoft/PARPACK
        LIBS += -L$(ARPACK_LIBDIR) -larpack
endif
ifdef USE_MPI
    CPPFLAGS += -DMPI
ifdef USE_MPIF90
    FC := mpif90
else
    LIBS += -Bdynamic -fmpi-pgi -lmpi-pgi -Bstatic
endif
endif
ISO8 - ** - XEmacs: Linux-pgi.mk      (Makefile) ---- L74 -- C0 -- 27% --
```

?= means that it  
checks your  
environment first

# Model organization

mpi\_init

init\_file (# procs/model)

SWAN

{  
  init  
  run  
  finalize}

ROMS

{  
  init  
  run  
  finalize}

File Edit View Options Insert Insert... Open Dired Save Print Cut Copy Paste Undo Spell Replace Mail Info Compile Debug News

waves\_coupler.F master.F

```
coawst/MASTER/master.F には include " .../MPI-parallel.h" (*.h)
CALL mpi_init (MyError) この部分は coawst/MASTER/mct-driver.h に存在
Get rank of the local process in the group associated with the する。
CALL mpi_comm_size (MPI_COMM_WORLD, nprocs, MyError) が...の組合せで毎回 include される。
CALL mpi_comm_rank (MPI_COMM_WORLD, MyRank, MyError) プログラム本体は MPI-Parallel
Read in waves-ocean coupling parameters from standard input. に含まれている。
Set temporarily the ocean communicator to current handle before
splitting so the input coupling script name can be broadcasted to
all the nodes.

OCN_COMM_WORLD=MPI_COMM_WORLD
CALL read_CouplePar (iNLM)

Split the communicator into SWAN and ROMS subgroups based on color
and key.

MyKey=0
IF (MyRank.le.peWAV_last) THEN
  MyColor=1
  MyString="COMPONENT_ID=1,COMPONENT_NAME=swan"
! 1=SWAN
ELSE
  MyColor=2
  MyString="COMPONENT_ID=2,COMPONENT_NAME=roms"
! 2=ROMS
ENDIF
CALL mpi_comm_split (MPI_COMM_WORLD, MyColor, MyKey, MyCOMM,
& MyError)

! Run either SWAN or ROMS according to the processor rank.

IF (MyRank.le.peWAV_last) THEN
  CALL SWINITMPI (MyCOMM)
  CALL SWMAIN (nWAV_steps)
  CALL SWEXITMPI
ELSE
  first=.TRUE.
  Nrun=1
  IF (exit_flag.eq.NoError) THEN
    CALL roms_init (first, MyCOMM)
  END IF
  IF (exit_flag.eq.NoError) THEN
    CALL roms_run
  END IF
  CALL roms_finalize
END IF

! Terminates all the MPI processing.

CALL mpi_finalize (MyError)

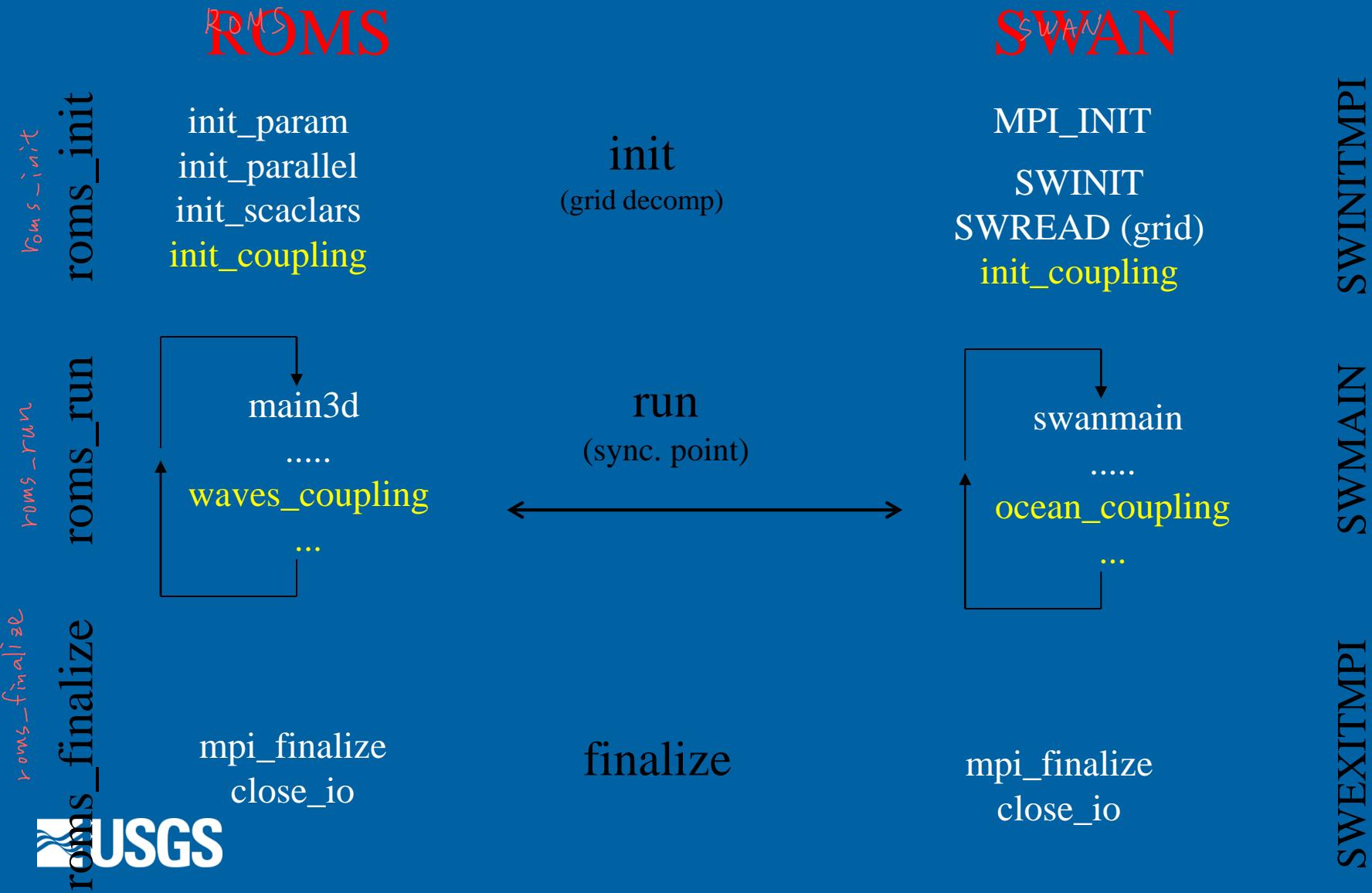
END PROGRAM waves_ocean
```

Raw-----XEmacs: master.F (Fortran Font) ----L84--C0--Bottom-----

Font Courier New:Regular:10::Western

coawst/MASTER/master.F には include " .../MPI-parallel.h" (\*.h)  
この部分は coawst/MASTER/mct-driver.h に存在  
する。  
が...の組合せで毎回 include される。  
PROGRAM本体は MPI-Parallel  
に含まれている。  
MPIがメイン  
(Serialでは動かない)

# init, run, and finalize



# Grid decomposition (during initialization)

ROMS

2次元タイル

19	20	21	22	23	24
13	14	15	16	17	18
7	8	9	10	11	12
1	2	3	4	5	6

ROMS : example grid tiling

→ NTILEI = 1, NTILEJ = 2 ( $1 \times 2 = 2$  tile)

tile0: start(1) = 1, length(1) = 6

start(2) = 7, length(2) = 6

tile1: start(1) = 13, length(1) = 6

start(2) = 19 length(2) = 6

MPI の領域分割の方法

ROMS ではユーザが水平と垂直の分割数を指定する

SWAN では分割数の時指定し、お任せ。

SWAN

1次元タイル (水平)

19	20	21	22	23	24
13	14	15	16	17	18
7	8	9	10	11	12
1	2	3	4	5	6

SWAN : example grid tiling

→ NTILE = 3 長さ2

tile0: start(1) = 1, length(1) = 2

start(2) = 7, length(2) = 2

start(3) = 13, length(3) = 2

start(4) = 19, length(4) = 2

tile1: start(1) = 3, length(1) = 2

start(2) = 9, length(2) = 2

start(3) = 15, length(3) = 2

start(4) = 21, length(4) = 2

tile0: start(1) = 5, length(1) = 2

start(2) = 11, length(2) = 2

start(3) = 17, length(3) = 2

start(4) = 23, length(4) = 2

-Each tile is on a separate processor.

-Each tile registers with MCT.



Figure 1. Example of how SWAN and ROMS decompose the same grid. SWAN decomposition is determined by grid orientation whereas ROMS decomposition is user specified.

# Determine what information to exchange

## ROMS- init\_coupling

processed by each ROMS tile

```
waves_coupler.F - XEmacs
File Edit View Cmds Tools Options Buffers
File Open Dired Save Print Cut Copy Paste Undo Spell Replace Info Compile Debug News
waves_coupler.F master.F

! Determine start and lengths for domain decomposition.
!
Jsize=JendR-JstrR+1
allocate(start(Jsize))
allocate(length(Jsize))
indx=0
DO j=JstrR,JendR
  indx=indx+1
  length(indx)=(IendR-IstrR+1)
  start(indx)=j*(Lm(ng)+2)+IstrR+1
END DO
!
1 CALL GlobalSegMap_init(GSMMapROMS,start,length,0,OCN_COMM_WORLD, &
  & OcnId)

! Initialize Attribute vectors.
!
! size is the number of grid point on this processor
gsmsize = MCT_GSMMap_lsize(GSMMapROMS,OCN_COMM_WORLD)

2 CALL MCT_AtrVt_init(FromWavesAV,rList= &
  &"DISSIP:FORCEX:FORCEY:HSIGN:RTP:SETUP:TMBOT:UBOT:DIR:WLEN:TM01:XP" &
  &"&:YP:TM01",lsize=gsmsize)

! initialize oceanAv with one real attribute for now.
call MCT_AtrVt_init(ToWavesAV,rList="DEPTH:WLEV:VELX:VELY", &
  & lsize=gsmsize)

! initialize a router to the Waves component.
3 call MCT_Router_init(WavId,GSMMapROMS,OCN_COMM_WORLD,RoutROMS)

Raw-----XEmacs: waves_coupler.F      (Fortran Font) ----L165--C0--21%
Font Courier New:Regular:16::Western
```

## SWAN- init\_coupling

processed by each SWAN tile

```
swancplr.fnt - XEmacs
File Edit View Cmds Tools Options Buffers
File Open Dired Save Print Cut Copy Paste Undo Spell ABCs Replace Mail Info Compile Debug News
swancplr.fnt

!
DO j = 1,Jsize
  length(j) = ISIZE
  IF (MXCGL.gt.MYCGL) THEN
    IF (MyRank.eq.0) THEN
      start(j) = MXF + (j-1)*MXCGL
    ELSE
      start(j) = MXF + (j-1)*MXCGL + IHALOX
    ENDIF
  ELSE
    IF (MyRank.eq.0) THEN
      start(j) = MYF + (j-1)*MXCGL
    ELSE
      start(j) = (MYF + IHALOY - 1)*MXCGL + 1 + (j-1)*MXCGL
    ENDIF
  END DO

gsmsize = Isize*Jsize
1 CALL GlobalSegMap_init(GSMMapSWAN,start,length,0, &
  & WAV_COMM_WORLD,WavId)

! Initialize Attribute Vector ToOceanAv to hold the data sent to ROMS.
!
CALL AttrVect_init(ToOceanAV,rlist= &
  &"DISSIP:FORCEX:FORCEY:HSIGN:RTP:SETUP:TMBOT:UBOT:DIR:WLEN:TM01:XP" &
  &"&:YP:TM01",lsize=gsmsize)

! Initialize Attribute Vector FromOceanAV that will have ROMS data in it.
!
CALL AttrVect_init(FromOceanAV,rList="DEPTH:WLEV:VELX:VELY", &
  & lsize=gsmsize)

! Initialize a router to the Waves component.
3 call Router_init(OcnId,GSMMapSWAN,WAV_COMM_WORLD,RoutSWAN)

Raw-----XEmacs: swancplr.fnt      (Fundamental) ----L141--C0--21%
Wrote C:\data\presentations\2006_04_24_WISE\swancplr.fnt
```

# Synchronization (run phase)

<sup>ROMS</sup>  
ROMS- ocean\_coupling

```
X waves_coupler.F - XEmacs
File Edit View Cmds Tools Options Buffers Help
waves_coupler.F master.F

!
! Receive data from SWAN.
!

CALL MCT_Recv(FromWavesAV, RoutROMS, MyError)

!
! Fill local variables with received data.

! Fill attribute vector.

! Send the data
CALL MCT_SEND(ToWavesAV, RoutROMS, MyError)
write(*,*) '++ ROMS sent ocean fields and Myerror??= ', MyError

Raw-----XEmacs: waves_coupler.F      (Fortran Font)----L725--C0--Bot-----
```

受信

送信

MCT

MCT

<sup>SWAN</sup>  
SWAN- waves\_coupling

```
X swancplr.ftn - XEmacs
File Edit View Cmds Tools Options Buffers Help
swancplr.ftn

!
! Fill attribute vectors.

!
! Get output variable id.
IVTYPE = IVTYP(1)
!ask for points in this tile
CALL GlobalSegMap_OrdPnts(GSMAPSWAN, MyRank, points)
DO IP=1,gsmsize
    avdata(IP) = VOQ(points(IP), VOQR(IVTYPE))
END DO
IF (IVTYPE.eq.4) THEN
    CALL AttrVect_importRArr(ToOceanAV, "DEPTH", avdata)
END IF
IF (IVTYPE.eq.7) THEN
    CALL AttrVect_importRArr(ToOceanAV, "DISSIP", avdata)
END IF
.....(+ other vars)

!
! Send wave parameters to ROMS.
CALL MCT_SEND(ToOceanAV, RoutSWAN, MyError)
送信

!
! Receive from ROMS: Water Level, Depth, VELX, and VELY.
CALL MCT_Recv(FromOceanAV, RoutSWAN, MyError)
受信

IP=0
DO IY = Jstr, Jend
    DO IX = Istr, Iend
        IP=IP+1
        INDX = KGRPNT(IX, IY)
        IF (INDX.GT.1) THEN
            ! insert water levels from ROMS into SWAN wlev array.
            COMPDA(INDX, JWLV2)=FromOceanAV%rAttr(2, IP)
            ! insert velx and vely from ROMS into SWAN velocity arrays.
            COMPDA(INDX, JVX2)=FromOceanAV%rAttr(3, IP)
            COMPDA(INDX, JVY2)=FromOceanAV%rAttr(4, IP)
        END IF
    END DO
END DO
Raw-----XEmacs: swancplr.ftn      (Fundamental)----L560--C23--92%-----
```

processed by each SWAN tile

processed by each ROMS tile

# End phase

TextPad - C:\work\models\COAWST\Master\mct\_coupler.h \*

File Edit Search View Tools Macros Configure Window Help

mct\_coupler.h \* x

```
!#if defined SWAN_COUPLING
    IF (MyColor.eq.WAVcolor) THEN
        CALL SWAN_driver_init (MyCOMM)
        CALL SWAN_driver_run
        CALL SWAN_driver_finalize
    END IF
#endif
#ifndef WRF_COUPLING
    IF (MyColor.eq.ATMcolor) THEN
        CALL wrf_init (MyCOMM)
        CALL wrf_run
        CALL wrf_finalize(.TRUE.)
    END IF
#endif
#ifndef ROMS_COUPLING
    IF (MyColor.eq.OCNcolor) THEN
        first=.TRUE.
        Nrun=1
        IF (exit_flag.eq.NoError) THEN
            CALL ROMS_initialize (first, MyCOMM)
        END IF
        IF (exit_flag.eq.NoError) THEN
            run_time=0.0_m8
            DO ng=1,Ngrids
                run_time=MAX(run_time, dt(ng)*ntimes(ng))
            END DO
            CALL ROMS_run (run_time)
        END IF
        CALL ROMS_finalize
    END IF
#endif
!-----
! Terminates all the mpi-processing and coupling.
!-----
    CALL mpi_barrier (MPI_COMM_WORLD, MyError)
    CALL MCTWorld_clean ()
    CALL mpi_finalize (MyError)

    STOP
END PROGRAM mct_coupler
```

coawst /Master/mct\_driver.h

“master.F”

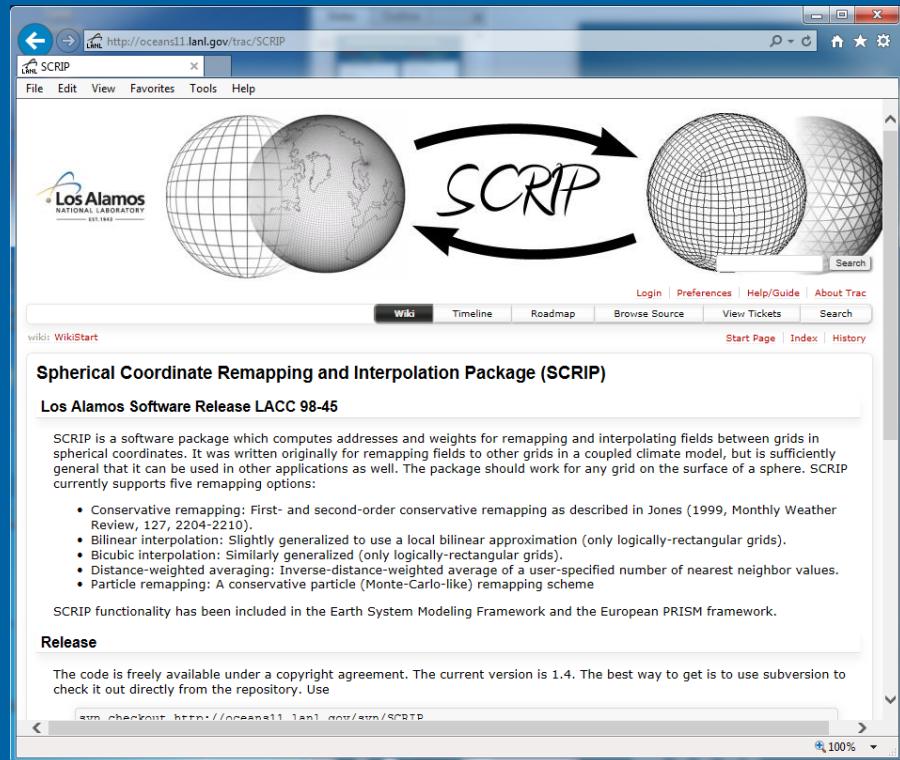
MPI barrier prevents program complete exit until all models arrive at this point.

Sometimes one model may die and other model is waiting.

Tool Output

Desktop > 3:03 PM 8/8/2016

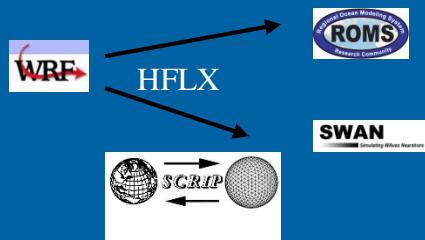
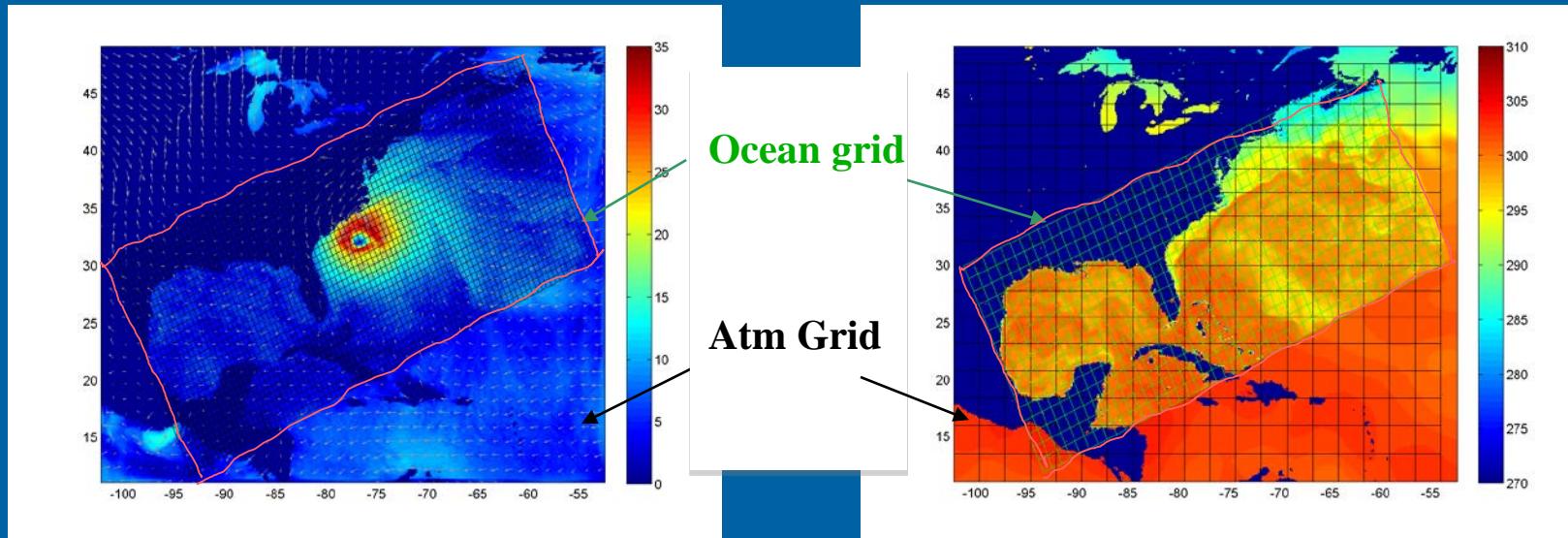
# SCRIP grid interpolation



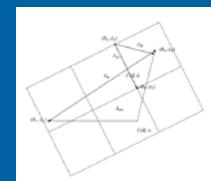
<http://oceans11.lanl.gov/trac/SCRIP>

- Needed when you run more than 1 model and the models are on different grids.
- We started with SCRIP from LANL.
- We adapted the code to read model grids, and write out one NetCDF file with all the interpolation weights.
- We use the ‘group’ capability in NetCDF4.

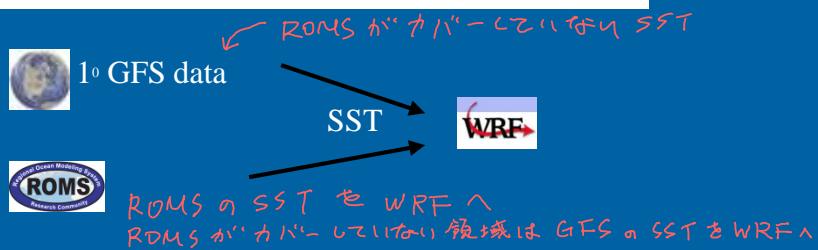
# 3) SCRIP - grid interpolation



WRF の HFLX & ROMS & SWAN の  
Atmosphere model provides heat flux to  
cover entire ocean grid. SCRIP  
interpolations weights needed to remap  
data fields.



Flux conservative  
remapping scheme



Ocean model provides higher resolution and  
coupled response of SST to atmosphere. But the  
ocean grid(s) are limited in spatial coverage so  
atmosphere model must combine data from  
different sources.

内持

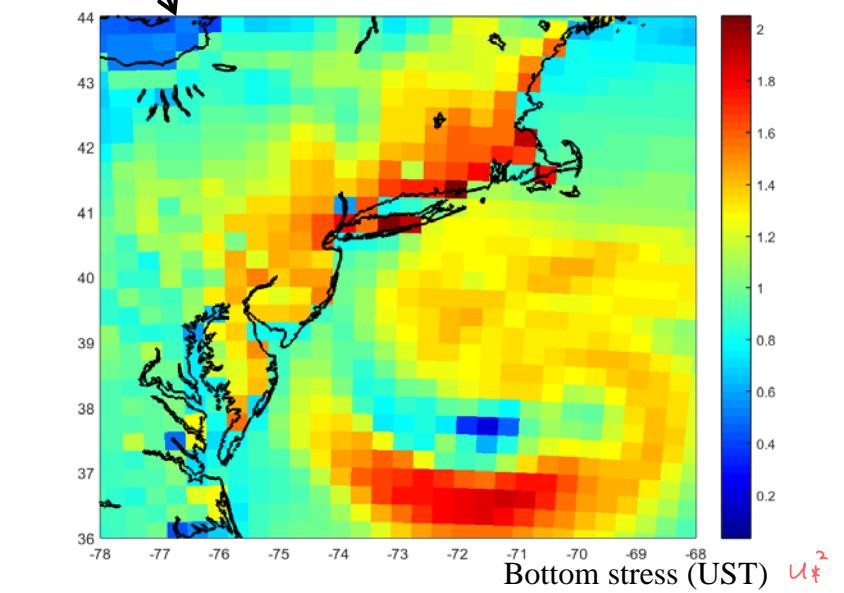
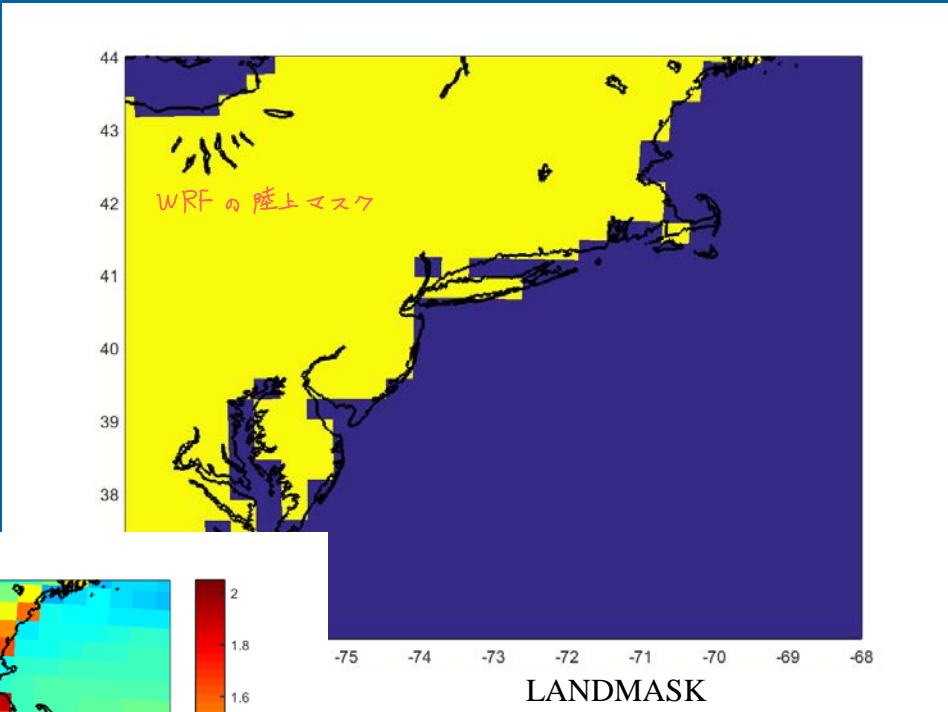
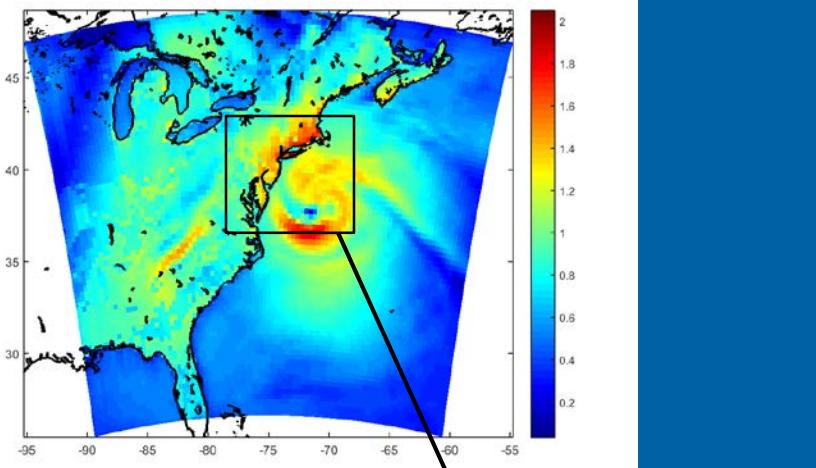
モデルによる SST の入力となるデータソースからの  
データの統合 (ROMS SST と GFS SST) を  
任すのが SCRIP



Two main issues:

- 1) Interpolate data between different grids.
- 2) Combine data from different sources.

# Land masking effects – example of grid interpolation

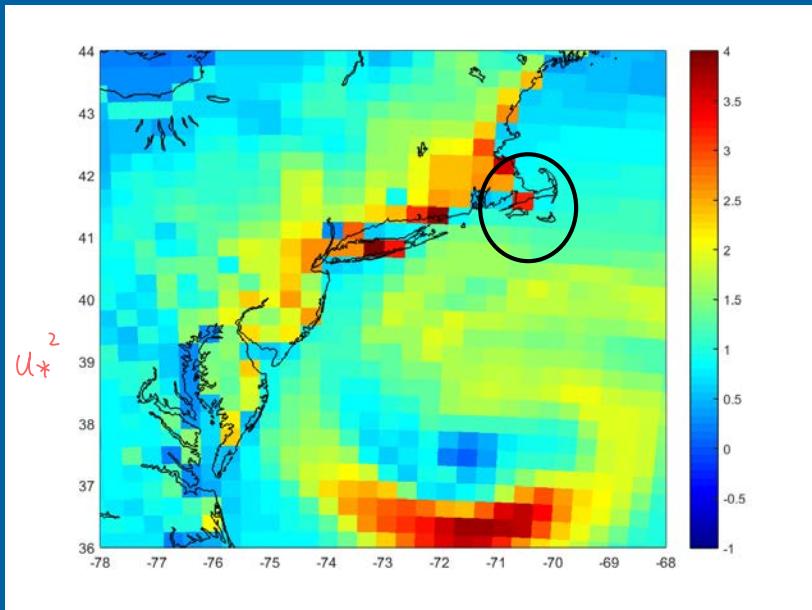


This is a coarse example to demonstrate the effects of land/sea masking on atmospheric processes.

Large bottom stress over land will cause problems on an ocean grid with different masking.

# Grid interpolation

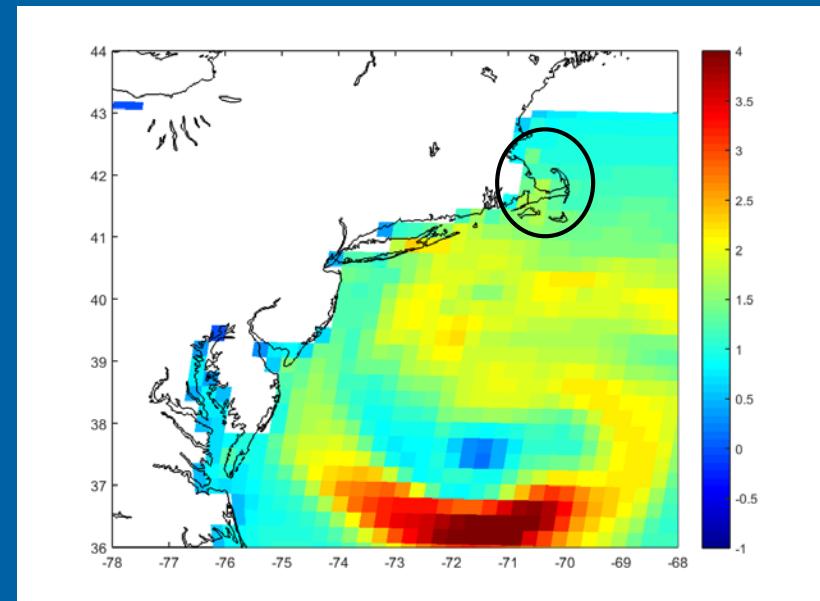
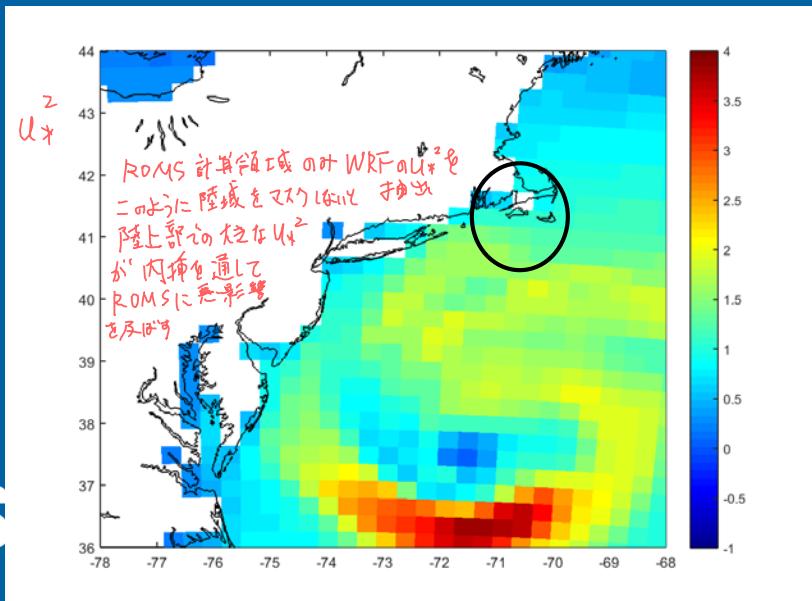
WRF  
UST<sup>2</sup> over  
land + ocean



ATM land mask areas do not  
effect ocean.

ROMS  
su/vstr over  
ocean only

WRF  
UST<sup>2</sup> over  
ocean only

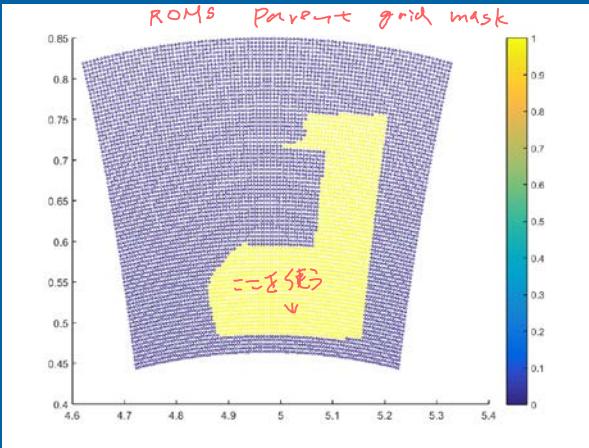
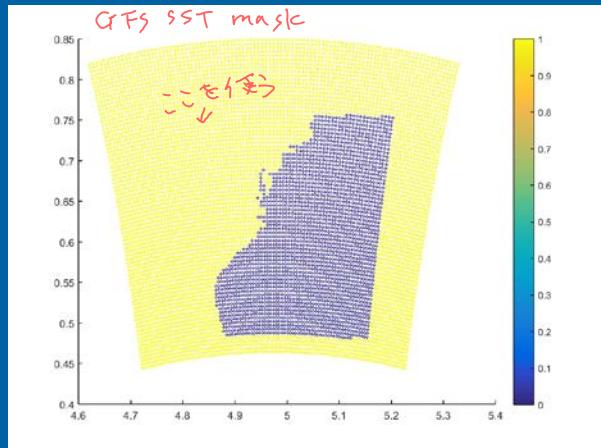


vRF を使う background SST は real.exe が生成する wrflowinp\_d01,-d02 に含まれる。

# Combining multiple sources

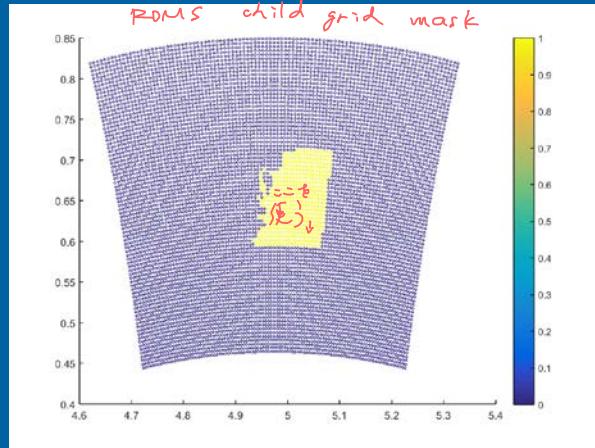
ROMS の SST (1 WRF - ROMS カッコリング) 計算を生成されるもの (あらかじめ存在するものはない)。ここで内挿方法を決めるだけ。  
カッコリング 計算中に内挿するだよ。  
SCRIP で内挿方法を指定する netcdf を作る。

Sum of all 3 masks = 1

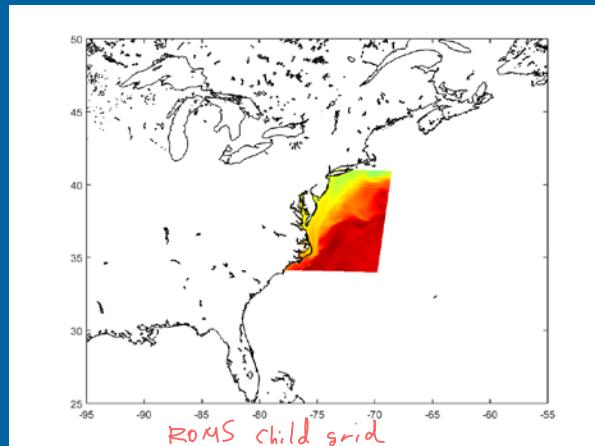
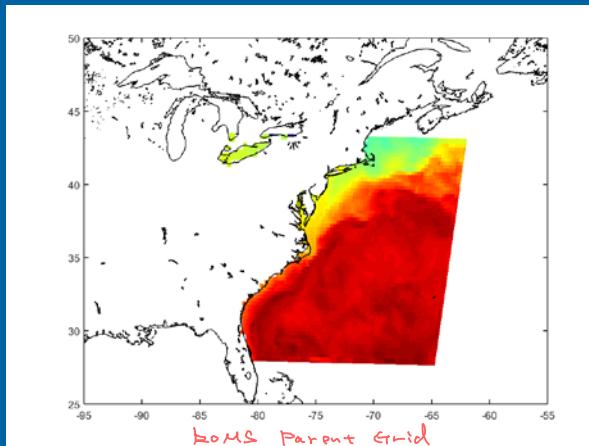
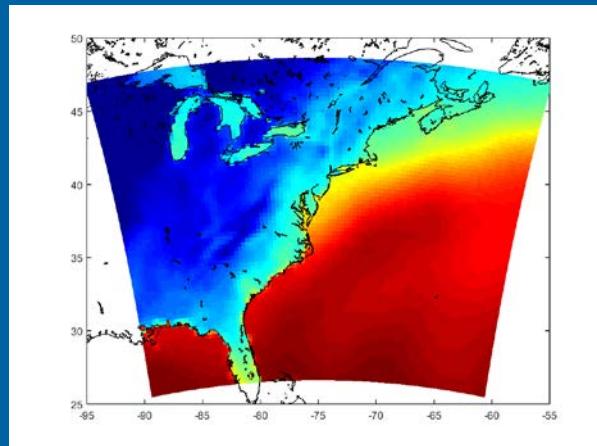


Background GFS SST mask

ocn 1 to atm 1 mask



ocn 2 to atm 1 mask



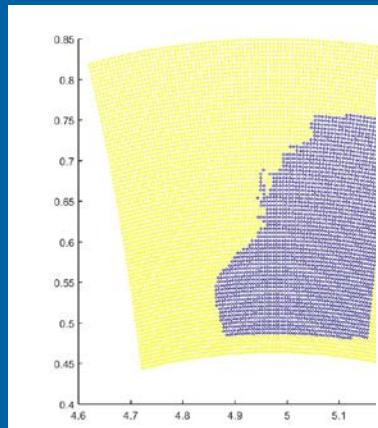
Background GFS SST

ocn 1 SST

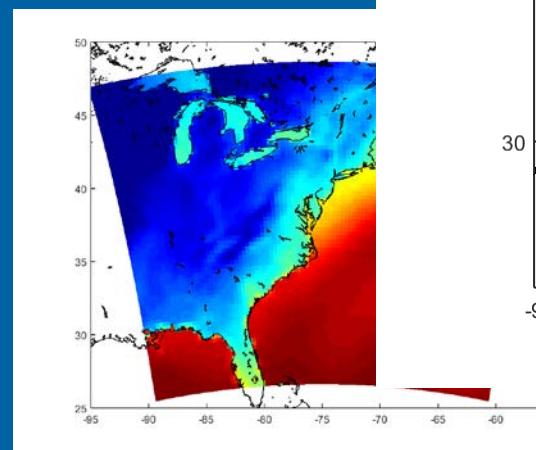
ocn 2 SST

# Combining multiple sources

Sum of all 3 masks = 1



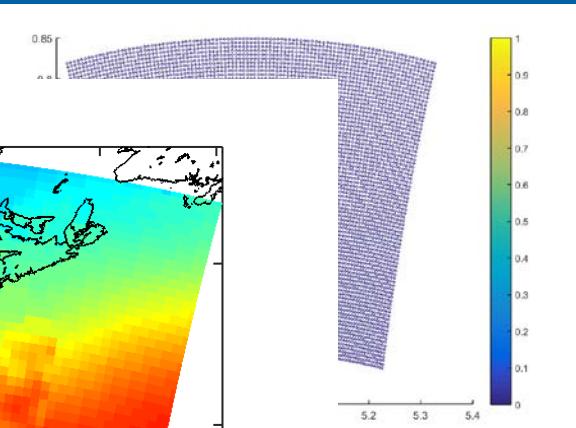
Background GFS



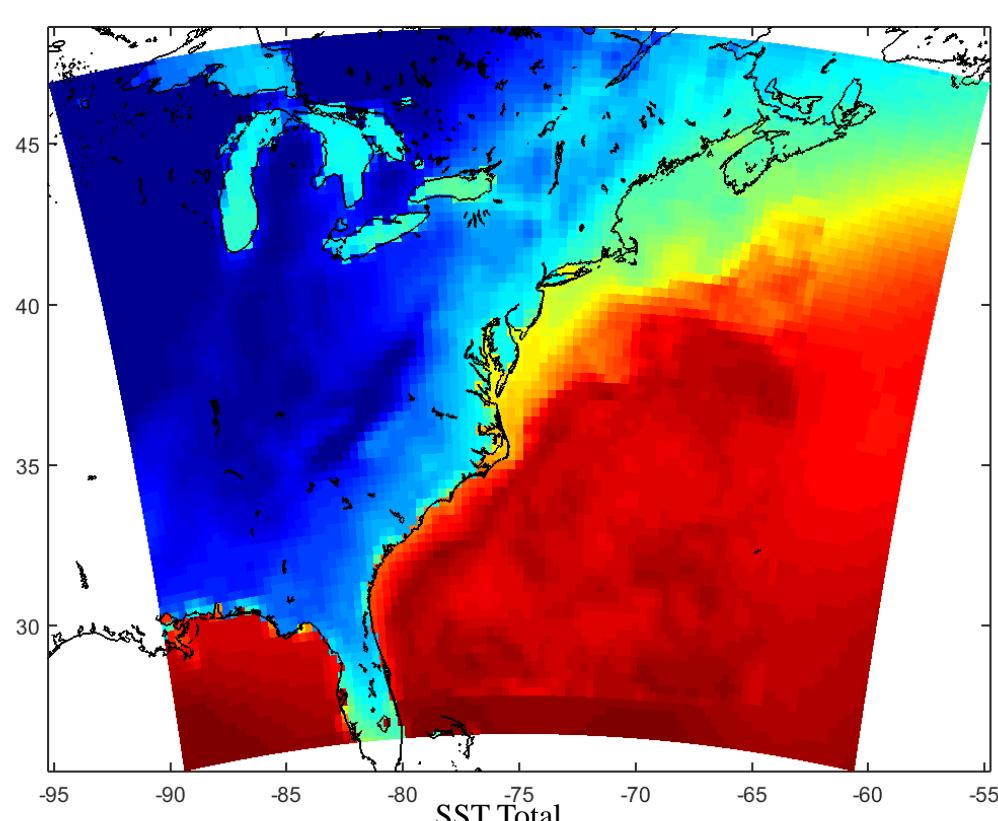
Background GFS SST



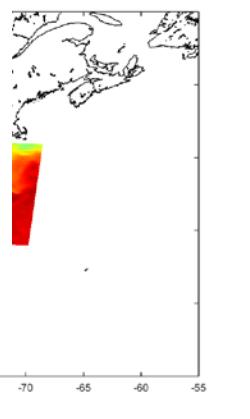
Sum of all 3 masks = 1



1 mask



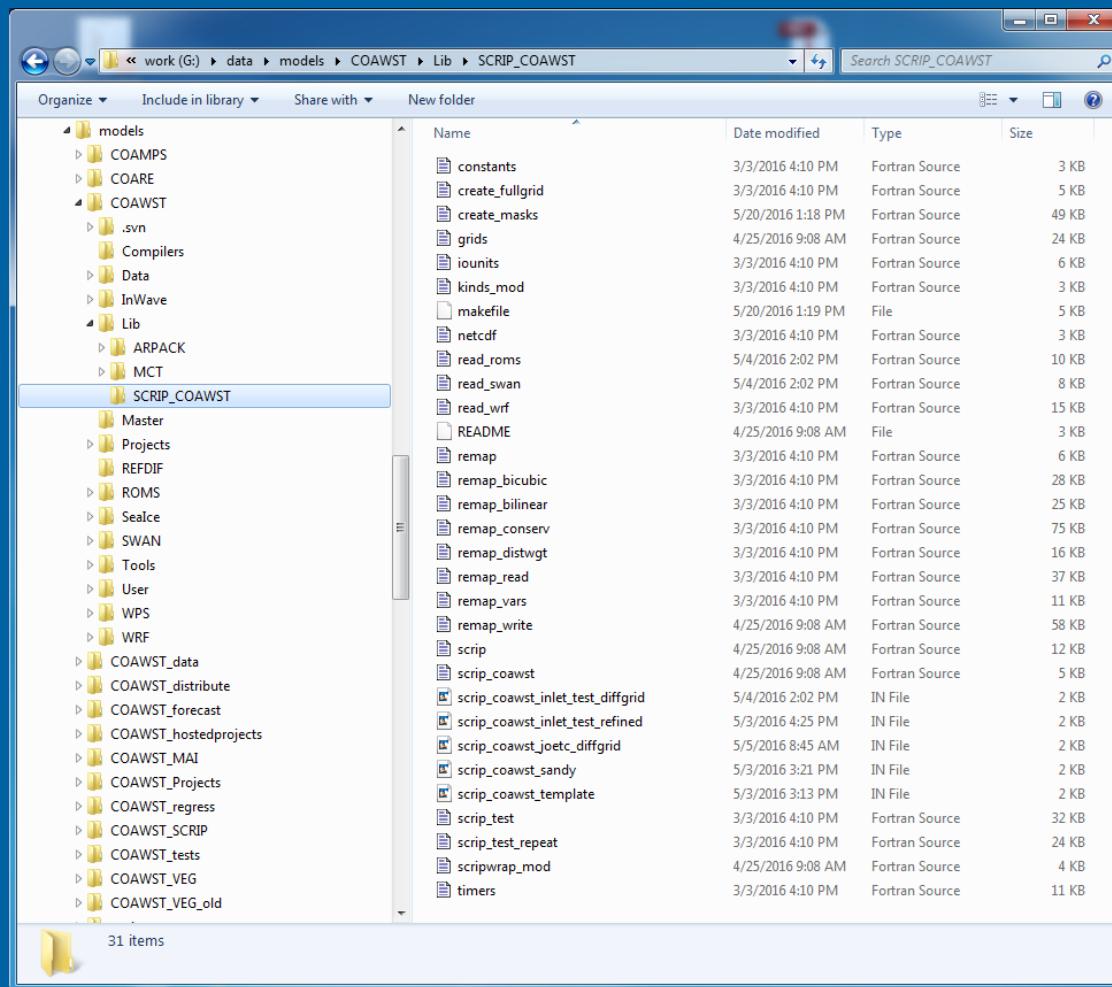
ocn 1 SST



ocn 2 SST

# SCRIP\_COAWST

- based on SCRIP v 1.6 and modified to read  
scrip\_coawst.in



# SCRIP\_COAWST

```
#!/bin/csh
#
# Makefile for scrip coawst
# You need to set: FORT
#
#FORT = ifort
#FORT = gfortran
FORT = pgi ← Set your compiler here

#
# YOU SHOULD NOT NEED TO CHANGE ANYTHING BELOW HERE

COMPILEERS = ../../Compilers ← I=1 P13 Flag & linux-ifort.mk +j

# This determine the operating system.
OS := $(shell uname -s | sed 's/[ \t]/-/g')
OS := $(patsubst CYGWIN%,CYGWIN,$(OS))
OS := $(patsubst MINGW%,MINGW,$(OS))
OS := $(patsubst sn%,UNICOS-sn,$(OS))

include $(COMPILEERS)/$(OS)-$(strip $(FORT)).mk

SRCDIR =
EXEDIR =
NETCDF_INCFILE := netcdf.inc
# NETCDF_MODFILE := netcdf.mod
# TYPESIZES_MODFILE := typesizes.mod

INCLUDE =

OBJSET = \
    kinds_mod.o \
    constants.o \
    iounits.o \
    netcdf.o \
    scripwrap_mod.o \

```

Tool Output

For Help, press F1

9 7 Read Ovr Block Sync Rec Caps

- 1) cd to COAWST/Lib/SCRIP\_COAWST
- 2) edit makefile
- 3) make

This will build the program  
“scrip\_coawst”

# SCRIP\_COAWST – input file

```
script_coawst_template.in ×
```

```
$INPUTS
!
! Input file for script_coawst.
! The $INPUTS line is required at the top of this file.
! Edit this file to enter the correct information below.
! Then run this program as "script_coawst script_coawst.in"
!
! 1) Enter name of output netcdf4 file
!
OUTPUT_NCFILE='script_file.nc' ← 出力ファイル名
!
! 2) Enter total number of ROMS, SWAN, and WRF (max_dom) grids:
!
NGRIDSS_ROMS=2,      2はネスティングの Parent v child grid
NGRIDSS_SWAN=2,
NGRIDSS_WRF=2,
!
! 3) Enter name of the ROMS grid file(s):
!
ROMS_GRIDS(1)='roms_grid1.nc', ROMSの Grid
ROMS_GRIDS(2)='roms_grid2.nc', ←
!
! 4) Enter SWAN information: SWANの Grid
! -the name(s) of the SWAN grid file(s) for coords and bathy.
! -the size of the SWAN grids, and
! -if the swan grids are Spherical(set cartesian=0) or
!   Cartesian(set cartesian=1).
!
SWAN_COORD(1)='swan_coord1.grd',
SWAN_COORD(2)='swan_coord2.grd',
SWAN_BATH(1)='swan_bathy1.bot',
SWAN_BATH(2)='swan_bathy2.bot',
SWAN_NUMX(1)=84,
SWAN_NUMX(2)=116,| SWAN_NUMY(1)=64,
SWAN_NUMY(2)=86,
CARTESIAN(1)=0,
CARTESIAN(2)=0,
```

Output file name

Number of grids  
for each model

ROMS grid names

SWAN grid names

# SCRIP\_COAWST – input file

```
! 4) Enter SWAN information:  
!   -the name(s) of the SWAN grid file(s) for coords and bathy.  
!   -the size of the SWAN grids, and  
!   -if the swan grids are Spherical(set cartesian=0) or  
!     Cartesian(set cartesian=1).  
!
```

```
SWAN_COORD(1)='swan_coord1.grd',  
SWAN_COORD(2)='swan_coord2.grd',  
SWAN_BATH(1)='swan_bathy1.bot',  
SWAN_BATH(2)='swan_bathy2.bot',  
SWAN_NUMX(1)=84,  
SWAN_NUMX(2)=116,  
SWAN_NUMY(1)=64,  
SWAN_NUMY(2)=86,  
CARTESIAN(1)=0,  
CARTESIAN(2)=0,
```

```
! 5) Enter the name of the WRF input grid(s). If the grid is a  
!   moving child nest then enter that grid name as 'moving'.  
!   Also provide the grid ratio, this is used for a moving nest.  
!
```

```
WRF_GRIDS(1)='wrfinput_d01', WRF Grid  
WRF_GRIDS(2)='moving', moving nest の場合  
PARENT_GRID_RATIO(1)=1,  
PARENT_GRID_RATIO(2)=3,  
PARENT_ID(1)=0  
PARENT_ID(2)=1
```

```
!  
!   The $END statement below is required  
!  
$END
```

SWAN grid names

WRF grid names

Enter ‘moving’ for a moving nest.  
Only the last WRF grid can be moving.

SST バックグラウンドはどうすれば  
指定するの？

# SCRIP file(s)

Run the scrip coawst program using

./scrip\_coawst scrip\_coawst.in      実行方法

We use NetCDF 4 functionality to create only 1 netcdf file using the ‘group’ functionality.

To read the data in matlab you can use (for example)

```
dst_grid2=ncread('scrip_sandy_static.nc','/ocn2_to_atm1_weights.nc/dst_grid_imask');
```

---

The old way: created a weight file between each grid.

So if you have 1 WRF grid, and 2 ROMS grids, it will produce

atm1\_to\_ocn1\_weights.nc

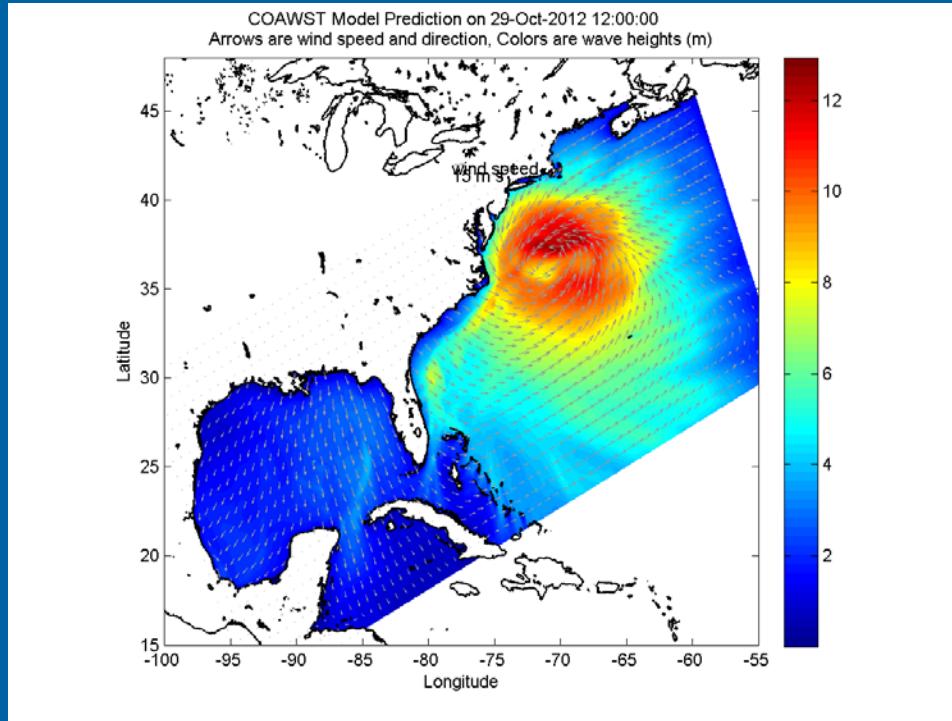
atm1\_to\_ocn2\_weights.nc

ocn1\_to\_atm1\_weights.nc

ocn2\_to\_atm1\_weights.nc



# Setting up a coupled application: Projects/Sandy



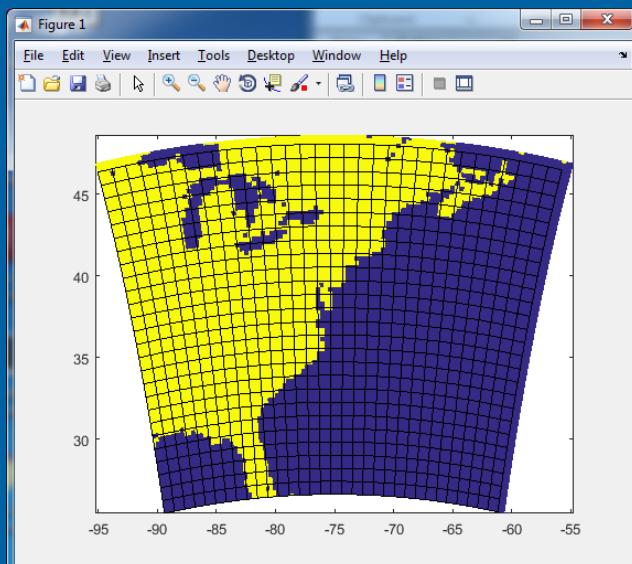
# How to create coupled application

- 1) Create all input, BC, init, forcing, etc files for each model as if running separately. I recommend that you run each model separately first.
- 2) SCRIP (if different grids or grid refinement)  
*/Lib/SCRIP\_COAWST/scrip\_coawst\_sandy.in*
- 3) modify cppdefs in your header file.
- 4) ocean.in, namelist.in, and coupling.in  
*← ROMS*      *← WRF*
- 5) coawst.bash  
*← 説明なし*
- 6) run it as coawstM

# 1) Use each model separately

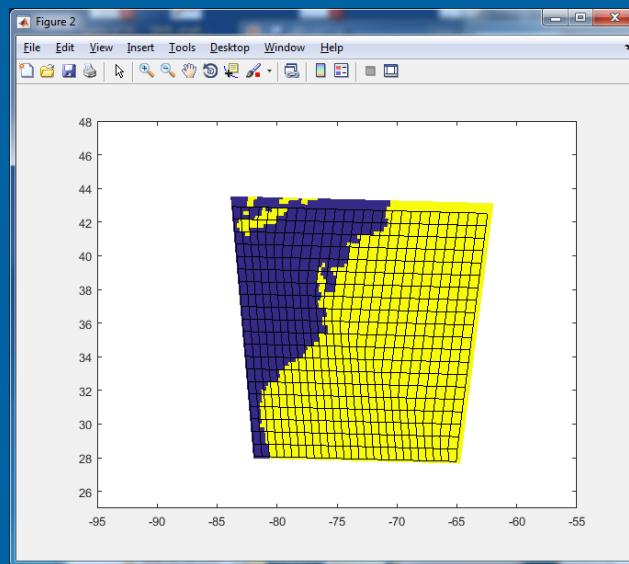
## WRF

- 27 vertical levels
- dt 36 s
- Physics
  - Lin microphysics
  - RRTM longwave, Dudhia shortwave
  - Mellor-Yamada-Janjic (MYJ) PBL
  - .....



## ROMS

- 16 vertical levels
- dt 240, 48
- Physics
  - GLS turbulence closure
  - COARE bulk fluxes
  - BC's from HYCOM
  - .....



# 2) SCRIP

Edit the SCRIP input file:  
COAWST/Lib/SCRIP\_COAWST/scrip\_coawst\_sandy.in

TextPad - F:\data\models\COAWST\Lib\SCRIP\_COAWST\scrip\_coawst\_sandy.in \*

File Edit Search View Tools Macros Configure Window Help

Find incrementally

scrip\_coawst\_sandy.in \*

```
$INPUTS
! Input file for scrip_coawst.
! The $INPUTS line is required at the top of this file.
! Edit this file to enter the correct information below.
! Then run this program as "scrip_coawst scrip_coawst_sandy.in"

! 1) Enter name of output netcdf4 file
!OUTPUT_NCFILE='scrip_sandy_moving.nc'
!OUTPUT_NCFILE='scrip_sandy_static.nc'
!OUTPUT_NCFILE='scrip_sandy_nowavenest.nc'

! 2) Enter total number of ROMS, SWAN, WW3, and WRF grids:
!NGRID_ROMS=2, ← } ROMS と WRF の総格子数を指定
NGRID_SWAN=0,
NGRID_WW3=0,
NGRID_WRF=2, ← }

! 3) Enter name of the ROMS grid file(s):
!ROMS_GRIDS(1)='../Projects/Sandy/Sandy_roms_grid.nc',
!ROMS_GRIDS(2)='../Projects/Sandy/Sandy_roms_grid_ref3.nc', } ROMS の格子を指定

! 4) Enter SWAN information:
! -the name(s) of the SWAN grid file(s) for coords and bathy.
! -the size of the SWAN grids (full number of center points), and
! -if the swan grids are Spherical(set cartesian=0) or
!   Cartesian(set cartesian=1).

!SWAN_COORD(1)='../Projects/Sandy/Sandy_swan_coord.grd',
!SWAN_COORD(2)='../Projects/Sandy/Sandy_swan_coord_ref3.grd',
!SWAN_BATH(1)='../Projects/Sandy/Sandy_swan_bathy.bot',
!SWAN_BATH(2)='../Projects/Sandy/Sandy_swan_bathy_ref3.bot',
!SWAN_NUMX(1)=87,
!SWAN_NUMX(2)=116,
!SWAN_NUMY(1)=65,
!SWAN_NUMY(2)=86,
!CARTESIAN(1)=0,
!CARTESIAN(2)=0,
```

/Lib/SCRIP\_COAWST/scrip\_coawst\_sandy.in

Sandy WRF-ROMS  
カーティジンガの名前

ROMS の格子を指定

ROMS の格子を指定

18 12 Read Ovr Block Sync Rec Caps

## 2) SCRIP

Edit the SCRIP input file:  
COAWST/Lib/SCRIP\_COAWST/scrif\_coawst\_sandy.in

TextPad - F:\data\models\COAWST\Lib\SCRIP\_COAWST\scrif\_coawst\_sandy.in \*

File Edit Search View Tools Macros Configure Window Help

Find incrementally

scrif\_coawst\_sandy.in \*

```
! 5) Enter WW3 information
!   -the name(s) of the WW3 grid file(s) for x- y- coords and bathy.
!   -the size of the WW3 grids (full number of grid center points).
!
WW3_XCOORD(1)='...../Projects/Sandy/ww3_sandy_xcoord.dat',
WW3_YCOORD(1)='...../Projects/Sandy/ww3_sandy_ycoord.dat',
WW3_BATH(1)='...../Projects/Sandy/ww3_sandy_bathy.bot',
WW3_NUMX(1)=84,
WW3_NUMY(1)=64.

! 6) Enter the name of the WRF input grid(s). If the grid is a
! moving child nest then enter that grid name as 'moving'.
!   Also provide the grid ratio, this is used for a moving nest.
!
WRF_GRIDS(1)='...../Projects/Sandy/wrfinput_d01',
WRF_GRIDS(2)='...../Projects/Sandy/wrfinput_d02',
!WRF_GRIDS(2)='moving',
PARENT_GRID_RATIO(1)=1,
PARENT_GRID_RATIO(2)=3,
PARENT_ID(1)=0
PARENT_ID(2)=1

!
! The $END statement below is required
$END
```

WRF の初期条件ファイル (格子情報)

36 | 17 | Read | Ovr | Block | Sync | Rec | Caps

## 2) SCRIP

Lib/SCRIP\_COAWST/ 作成

Run this in COAWST/Lib/SCRIP\_COAWST  
./scrip\_coawst scrip\_coawst\_sandy.in

This will create one file called  
OUTPUT\_NCFILE='scrip\_sandy\_static.nc'

WSL-2 Ubuntu ではこれで動いた  
ITO-A ではエラーとなり、  
mpicxx.hydra を用いパッチ処理して  
動いた。  
しかし両者どちらが netcdf のバージョン  
が異なる。 ncview で比較したが、  
よく分からなかった。  
scrip を生成し、既往を見てか、差は存在しない。  
どうもまずうまく行なうと判断  
生成された scrip\_sandy\_static.nc は  
Projects / Sandy / にコピーする  
(元々存在する時に注意)

### 3) sandy.h



TextPad - F:\data\models\COAWST\Projects\Sandy\sandy.h \*

File Edit Search View Tools Macros Configure Window Help

sandy.h \* scrip\_coawst\_sandy.in \*

```
/*
** svn $Id: sandy.h 25 2007-04-09 23:43:58Z jcwarner $
***** Copyright (c) 2002-2007 The ROMS/TOMS Group ****
** Licensed under a MIT/X style license
** See License_ROMS.txt
*****
** Options for Sandy Test.
**
** Application flag: SANDY
*/
#define ROMS_MODEL }
#define NESTING }
#define WRF_MODEL
#undef SWAN_MODEL
#undef WW3_MODEL
#define MCT_LIB }
#define MCT_INTERP_OC2AT }
#undef MCT_INTERP_WV2AT
#undef MCT_INTERP_OC2WV

#if defined WRF_MODEL && (defined SWAN_MODEL || defined WW3_MODEL)
#define DRAGLIM_DAVIS
#define COARE_TAYLOR_YELLAND
#endif

#ifndef ROMS_MODEL
/* Physics + numerics */
# if defined WW3_MODEL || defined SWAN_MODEL
# define WEC_VF
# define WDISS_WAVEMOD
# define UV_KIRBY
#endif
# define UV_ADV
# define UV_COR
# define UV_VIS2
# define MIX_S_UV
# undef TS_FIXED
# define TS_U3HADVECTION
# define TS_C4VADVECTION
# undef TS_MPDATA

```

For Help, press F1

22 | 8 | Read | Ovr | Block | Sync | Rec | Caps

ROMS + WRF  
application

# 4) coupling.in (this is a ROMS+WRF app)

```

FILE NAVIGATE EDIT BREAKPOINTS
+2 roms_master_climatology_sandy.m nam_narr_2roms.m nam_narr_2swan.m coupling_sandy.in +
37 ! Number of parallel nodes assigned to each model in the coupled system.
38 ! Their sum must be equal to the total number of processors. /Projects/Sandy/coupling_sandy.in
39
40 NnodesATM = 1 ! atmospheric model
41 NnodesWAV = 1 ! wave model
42 NnodesOCN = 1 ! ocean model
43
44 ! Time interval (seconds) between coupling of models.
45
46 TI_ATM2WAV = 1800.0d0 ! atmosphere to wave coupling interval
47 TI_ATM2OCN = 1800.0d0 ! atmosphere to ocean coupling interval
48 TI_WAV2ATM = 1800.0d0 ! wave to atmosphere coupling interval
49 TI_WAV2OCN = 1800.0d0 ! wave to ocean coupling interval
50 TI_OCN2WAV = 1800.0d0 ! ocean to wave coupling interval
51 TI_OCN2ATM = 1800.0d0 ! ocean to atmosphere coupling interval
52
53 ! Enter names of Atm, Wav, and Ocn input files.
54 ! The Wav program needs multiple input files, one for each grid.
55
56 ATM_name = namelist.input ← (namelist.input) ! atmospheric model
57 WAV_name = Projects/Sandy/swan_sandy.in \
      Projects/Sandy/swan_sandy_ref3.in ! wave model
58 OCN_name = Projects/Sandy/ocean_sandy.in ! ocean model
59
60
61 ! Sparse matrix interpolation weights files. You have 2 options:
62 ! Enter "1" for option 1, or "2" for option 2, and then list the
63 ! weight file(s) for that option.
64
65 SCRIP_WEIGHT_OPTION = 1 ← (SCRIP_WEIGHT_OPTION)
66 !
67 ! Option 1: IF you set "SCRIP_WEIGHT_OPTION = 1", then enter name
68 ! of the single netcdf file containing all the exchange

```

← 次ページ参照

plain text file Ln 17 Col 52

set # procs for each model

set coupling interval.  
Can be different for each direction.

input file names. only 1 for WRF, 1 for ROMS, multiple for SWAN

# 4) coupling.in (this is a ROMS+WRF app)

```
Editor - G:\data\models\COAWST\Projects\Sandy\coupling_sandy.in
FILE EDIT BREAKPOINTS
+ roms_master_climatology_sandy.m × nam_narr_2roms.m × nam_narr_2swan.m × coupling_sandy.in × + 
61 ! Sparse matrix interpolation weights files. You have 2 options:
62 ! Enter "1" for option 1, or "2" for option 2, and then list the
63 ! weight file(s) for that option.
64 !
65 SCRIP_WEIGHT_OPTION = 1  |が標準。1→のNetcdf形式
66 !
67 ! Option 1: IF you set "SCRIP_WEIGHT_OPTION = 1", then enter name
68 ! of the single netcdf file containing all the exchange
69 ! weights. This file is created using the code in
70 ! Lib/SCRIP_COAWST/script_coawst[.exe]
71 !
72 ! SCRIP_COAWST_NAME = Projects/Sandy/script_sandy_moving.nc
73 SCRIP_COAWST_NAME = Projects/Sandy/script_sandy_static.nc ← 1→の
74 ! Option 2: THIS OPTION WILL BE REMOVED IN FUTURE VERSIONS.
75 ! If you set "SCRIP_WEIGHT_OPTION = 2", then enter
76 ! the names of the separate files. The file names
77 ! must be provided in a specific order. For example:
78 !
79 W2ONAME == wav1_to_ocn1
80 ! wav1 to ocn2
81 ! wav1 to ocn3 ....for all the ocean models.
82 !
83 ! wav2 to ocn1
84 ! wav2 to ocn2
85 ! wav2 to ocn3 ....for all the ocean models.
86 !
87 W2ONAME == Projects/Sandy/wav1_to_ocn1_weights.nc \
88 Projects/Sandy/wav1_to_ocn2_weights.nc \
89 Projects/Sandy/wav2_to_ocn1_weights.nc \
90 Projects/Sandy/wav2_to_ocn2_weights.nc \
91 W2ANAME == Projects/Sandy/wav1_to_atm1_weights.nc \
92 Projects/Sandy/wav1_to_atm2_weights.nc \
93 Projects/Sandy/wav2_to_atm1_weights.nc \
94 A2ONAME == Projects/Sandy/atm1_to_ocn1_weights.nc \
95 Projects/Sandy/atm1_to_ocn2_weights.nc \
96 Projects/Sandy/atm2_to_ocn1_weights.nc \
97 Projects/Sandy/atm2_to_ocn2_weights.nc \
98 A2WNAME == Projects/Sandy/atm1_to_wav1_weights.nc \
99 Projects/Sandy/atm1_to_wav2_weights.nc \
100 Projects/Sandy/atm2_to_wav1_weights.nc \
101 Projects/Sandy/atm2_to_wav2_weights.nc \
102 O2ONAME == Projects/Sandy/ocn1_to_atm1_weights.nc \
103 Projects/Sandy/ocn1_to_atm2_weights.nc \
104 Projects/Sandy/ocn2_to_atm1_weights.nc \
105 Projects/Sandy/ocn2_to_atm2_weights.nc \
106 O2WNAME == Projects/Sandy/ocn1_to_wav1_weights.nc \
107 Projects/Sandy/ocn1_to_wav2_weights.nc \
108 Projects/Sandy/ocn2_to_wav1_weights.nc \
109 Projects/Sandy/ocn2_to_wav2_weights.nc
110 !
```

SCRIP weights OPTION 1 or 2

SCRIP weights files name  
listed here for Option 1  
(single file)

NetCDF file uses “groups” feature so  
we need netcdf 4.

SCRIP weights files names  
are listed here for Option 2  
(multiple files, this option to  
be phased out.)

# 4) ocean.in

```
Lm == 82 114      ! Number of I-direction INTERIOR RHO-points
Mm == 62 84      ! Number of J-direction INTERIOR RHO-points
N == 16 16        ! Number of vertical levels
ND == 0           ! Number of wave directional bins
```

Nbed = 0 *スティング* ! Number of sediment bed layers

```
NAT = 2          ! Number of active tracers (usually, 2)
NPT = 0          ! Number of inactive passive tracers
NCS = 0          ! Number of cohesive (mud) sediment tracers
NNS = 0          ! Number of non-cohesive (sand) sediment tracers
```

*ROMSの設定*

```
! Domain decomposition parameters for serial, distributed-memory or
! shared-memory configurations used to determine tile horizontal range
! indices (Istr,Iend) and (Jstr,Jend), [1:Ngrids].
```

```
NtileI == 1      ! I-direction partition
NtileJ == 1      ! J-direction partition
```

*プロセス数*

*I-direction partition*

*J-direction partition*

*1×1=1プロセス*

set # procs NtileI NtileJ

```
! Set lateral boundary conditions keyword. Notice that a value is expected
! for each boundary segment per nested grid for each state variable.
```

```
! Each tracer variable requires [1:4,1:NAT+NPT,Ngrids] values. Otherwise,
! [1:4,1:Ngrids] values are expected for other variables. The boundary
! order is: 1=west, 2=south, 3=east, and 4=north. That is, anticlockwise
! starting at the western boundary.
```

```
! The keyword is case insensitive and usually has three characters. However,
! it is possible to have compound keywords, if applicable. For example, the
! keyword "RadNud" implies radiation boundary condition with nudging. This
! combination is usually used in active/passive radiation conditions.
```

*Time-Stepping parameters.*

*DTは*

```
NTIMES == 5760    11520
DT == 30.0d0     15.0d0
NDTFAST == 28     28
```

*Coupling interval の分割*

*である必要がある*

*Model iteration loops parameters.*

*スティング*

```
ERstr = 1
ERend = 1
Nouter = 1
Ninner = 1
Nintervals = 1
```

```
! Number of eigenvalues (NEV) and eigenvectors (NCV) to compute for the
! Lanczos/Arnoldi problem in the Generalized Stability Theory (GST)
! analysis. NCV must be greater than NEV (see documentation below).
```

```
NEV = 2           ! Number of eigenvalues
NCV = 10          ! Number of eigenvectors
```

*Input/Output parameters.*

NRREC == 0 0

dt needs to divide evenly into the  
coupling interval.

# 4/5) namelist.input

TextPad - F:\data\models\COAWST\Projects\Sandy\namelist.input

File Edit Search View Tools Macros Configure Window Help

namelist.input coupling\_sandy.in sandy.h\* scrip\_coawst\_sandy.in\*

```
start_year = 2012, 2012, 2012,
start_month = 10, 10, 10,
start_day = 28, 28, 28,
start_hour = 12, 12, 12,
start_minute = 00, 00, 00,
start_second = 00, 00, 00,
end_year = 2012, 2012, 2012,
end_month = 10, 10, 10,
end_day = 30, 30, 30,
end_hour = 12, 12, 12,
end_minute = 00, 00, 00,
end_second = 00, 00, 00,
interval_seconds = 21600
input_from_file = .true., .true., .false.,
history_interval = 30, 30, 60,
frames_per_outfile = 1000, 1000, 1000,
restart = .false.,
restart_interval = 5000,
io_form_history = 2
io_form_restart = 2
io_form_input = 2
io_form_boundary = 2
debug_level = 0
auxinput4_inname = "wrfflowinp_d<domain>"
auxinput4_interval = 360, 360, 360,
io_form_auxinput4 = 2

&domains
time_step = 180, ←
time_step_fract_num = 0,
time_step_fract_den = 1,
max_dom = 2,
e_we = 85, 100, 94,
e_sn = 82, 100, 91,
e_vert = 48, 48, 48,
p_top_requested = 5000,
num_metgrid_levels = 40,
num_metgrid_soil_levels = 4,
dx = 30000, 10000, 3333.33,
dy = 30000, 10000, 3333.33,
grid_id =
parent_id =
i_parent_start = 0, 1, 2, 3,
j_parent_start = 1, 33, 30, 30,
parent_grid_ratio = 1, 8, 3, 3,
parent_time_step_ratio = 1, 3, 3,
feedback =
smooth_option =
corral_dist =
nproc_x = 2, 2,
nproc_y = 1

} プロセス数 |x| = 192 ←
```

WRF の設定  
3列あればネスティング用  
coawst / にコピー  
すれど必要有

SST と seaIce の時系列  
を利用してオプション

WRFUsersGuide  
P.130

need dt of 180 to divide  
evenly into coupling interval  
of 1800 sec.

set # procs for atm model

# 6) run it as coawstM

5)

coawstM のビルドの前に /WRF/configure.wrf を作成しておく。WRF 単独用ではなく、ニーズのカットリ=7" 設定値は /WRF/ において configure を実行し configure.wrf を作成する必要がある。  
その後 coawstM をビルドする。ITO-A 用のビルドスクリプトをバッチ処理で実行する。

- use total number of procs from coupling.in
- only 1 executable

```
#!/bin/bash
### Job name
#PBS -N cwstv3
### Number of nodes
#PBS -l nodes=1:ppn=8,walltime=120:00:00
### Mail to user
#PBS -m ae
#PBS -M jcwarner@usgs.gov
### Out files
###PBS -e isabel_105.err
###PBS -o isabel_105.out
### PBS queue
###PBS -q standard

umask 0002

echo "this job is running on:"
cat $PBS_NODEFILE

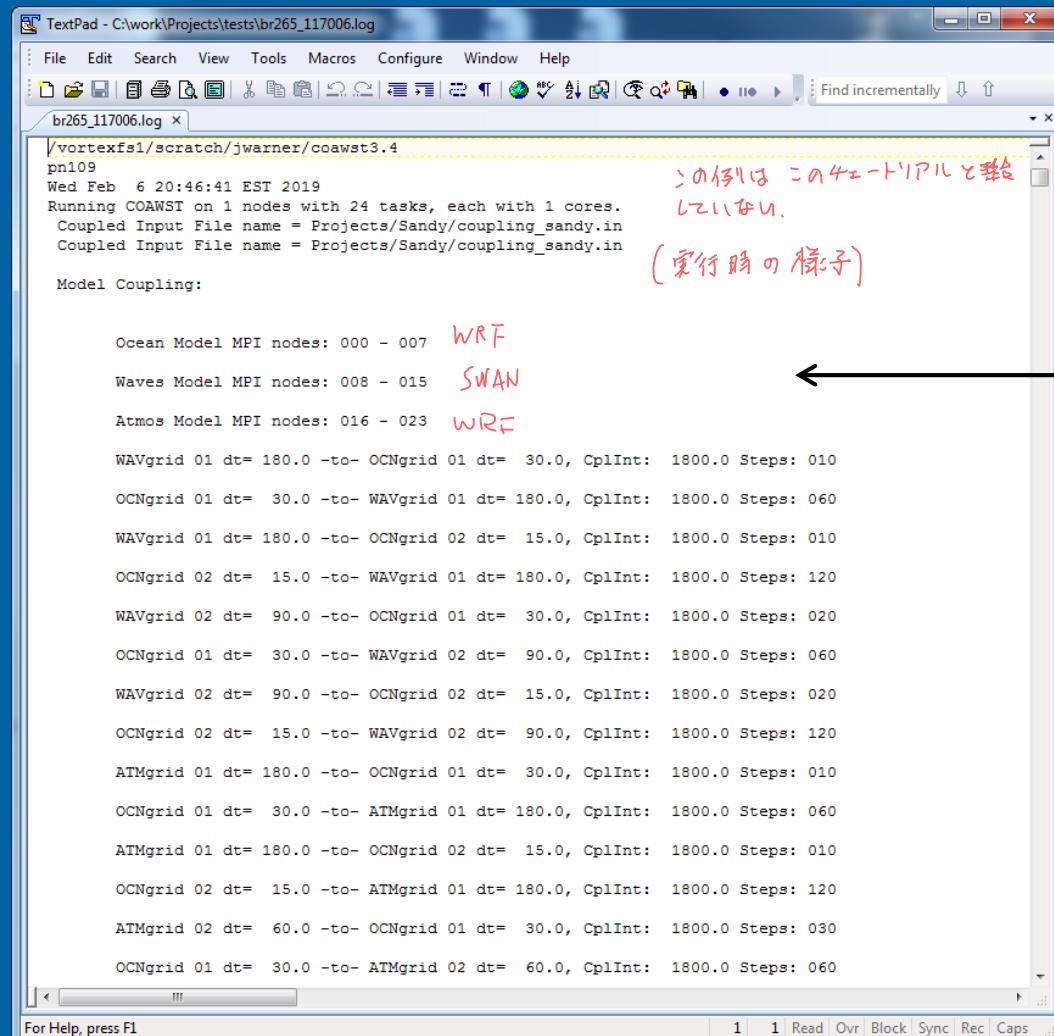
NPROCS=`wc -l < $PBS_NODEFILE`

cd /peach/data0/jcwarner/Models/COAWST_regress

mpirun -np 3 -machinefile $PBS_NODEFILE ./coawstM Projects/Sandy/coupling_sandy.in > cwstv3.out
```

実行用のスクリプト  
1台4処理スクリプトを用意する  
WRFのファイル: wrfinput\_d01, wrfinput\_d02  
wrfbdy\_d01, wrflowinp\_d01, wrflowinp\_d02  
はすべて coawst/ (=コアースト) に必要有  
ROMS のファイルはすべて coawst/Projects/Sandy/ にコピー  
↑ 2台目 WRFが1プロセス、ROMSが1プロセス両者を定めて2プロセス  
(この例ではWRFもROMSも1プロセスではある)。3台目は SWANを1プロセスで実行する。

# Processor allocation



TextPad - C:\work\Projects\tests\br265\_117006.log

File Edit Search View Tools Macros Configure Window Help

br265\_117006.log x

\vortexfs1/scratch/jwarner/coawst3.4  
pn109  
Wed Feb 6 20:46:41 EST 2019  
Running COAWST on 1 nodes with 24 tasks, each with 1 cores.  
Coupled Input File name = Projects/Sandy/coupling\_sandy.in  
Coupled Input File name = Projects/Sandy/coupling\_sandy.in

Model Coupling:

Ocean Model MPI nodes: 000 - 007      WRF  
Waves Model MPI nodes: 008 - 015      SWAN  
Atmos Model MPI nodes: 016 - 023      WRF

WAVgrid 01 dt= 180.0 -to- OCNgrid 01 dt= 30.0, CplInt: 1800.0 Steps: 010  
OCNgrid 01 dt= 30.0 -to- WAVgrid 01 dt= 180.0, CplInt: 1800.0 Steps: 060  
WAVgrid 01 dt= 180.0 -to- OCNgrid 02 dt= 15.0, CplInt: 1800.0 Steps: 010  
OCNgrid 02 dt= 15.0 -to- WAVgrid 01 dt= 180.0, CplInt: 1800.0 Steps: 120  
WAVgrid 02 dt= 90.0 -to- OCNgrid 01 dt= 30.0, CplInt: 1800.0 Steps: 020  
OCNgrid 01 dt= 30.0 -to- WAVgrid 02 dt= 90.0, CplInt: 1800.0 Steps: 060  
WAVgrid 02 dt= 90.0 -to- OCNgrid 02 dt= 15.0, CplInt: 1800.0 Steps: 020  
OCNgrid 02 dt= 15.0 -to- WAVgrid 02 dt= 90.0, CplInt: 1800.0 Steps: 120  
ATMgrid 01 dt= 180.0 -to- OCNgrid 01 dt= 30.0, CplInt: 1800.0 Steps: 010  
OCNgrid 01 dt= 30.0 -to- ATMgrid 01 dt= 180.0, CplInt: 1800.0 Steps: 060  
ATMgrid 01 dt= 180.0 -to- OCNgrid 02 dt= 15.0, CplInt: 1800.0 Steps: 010  
OCNgrid 02 dt= 15.0 -to- ATMgrid 01 dt= 180.0, CplInt: 1800.0 Steps: 120  
ATMgrid 02 dt= 60.0 -to- OCNgrid 01 dt= 30.0, CplInt: 1800.0 Steps: 030  
OCNgrid 01 dt= 30.0 -to- ATMgrid 02 dt= 60.0, CplInt: 1800.0 Steps: 060

For Help, press F1

1 | 1 Read Ovr Block Sync Rec Caps

この例は この4エントリアルと類似している。  
(実行時の様子)

←

stdout reports processor allocation  
and

Number of steps between synchronization.

This looks like from a different run,  
but you get the idea

# Processor allocation

```
emacs: sc81.out
File Edit View Cmds Tools Options Buffers Help
Open Direct Save Print Cut Copy Paste Undo Spell Replace Info Compile Debug News
sc81.out [sc90.out]
Timing for main: time 2003-12-01_02:36:30 on domain 1: 4.33416 elapsed seconds.
 1 176 52974 02:56:00 7.570245E-03 2.292071E+04 2.292072E+04 2.090614E+16
Timing for main: time 2003-12-01_02:37:00 on domain 1: 3.19787 elapsed seconds.
 1 177 52974 02:57:00 7.570758E-03 2.292071E+04 2.292071E+04 2.090614E+16
Timing for main: time 2003-12-01_02:37:30 on domain 1: 3.15102 elapsed seconds.
 1 178 52974 02:58:00 7.571265E-03 2.292070E+04 2.292071E+04 2.090613E+16
Timing for main: time 2003-12-01_02:38:00 on domain 1: 3.17619 elapsed seconds.
 1 179 52974 02:59:00 7.571764E-03 2.292070E+04 2.292071E+04 2.090613E+16
Timing for main: time 2003-12-01_02:38:30 on domain 1: 3.20623 elapsed seconds.
Timing for main: time 2003-12-01_02:39:00 on domain 1: 3.14843 elapsed seconds.
Timing for main: time 2003-12-01_02:39:30 on domain 1: 3.13314 elapsed seconds.
Timing for main: time 2003-12-01_02:40:00 on domain 1: 3.15249 elapsed seconds.
Timing for main: time 2003-12-01_02:40:30 on domain 1: 3.13559 elapsed seconds.
Timing for main: time 2003-12-01_02:41:00 on domain 1: 3.13626 elapsed seconds.
Timing for main: time 2003-12-01_02:41:30 on domain 1: 3.14025 elapsed seconds.
Timing for main: time 2003-12-01_02:42:00 on domain 1: 3.32610 elapsed seconds.

[ i removed some lines in here ..... ]]

Timing for main: time 2003-12-01_02:55:00 on domain 1: 3.13848 elapsed seconds.
Timing for main: time 2003-12-01_02:55:30 on domain 1: 3.13297 elapsed seconds.
Timing for main: time 2003-12-01_02:56:00 on domain 1: 3.14489 elapsed seconds.
Timing for main: time 2003-12-01_02:56:30 on domain 1: 3.13981 elapsed seconds.
Timing for main: time 2003-12-01_02:57:00 on domain 1: 3.13350 elapsed seconds.
Timing for main: time 2003-12-01_02:57:30 on domain 1: 3.15537 elapsed seconds.
Timing for main: time 2003-12-01_02:58:00 on domain 1: 3.14160 elapsed seconds.
Timing for main: time 2003-12-01_02:58:30 on domain 1: 3.14269 elapsed seconds.
Timing for main: time 2003-12-01_02:59:00 on domain 1: 3.14412 elapsed seconds.
Timing for main: time 2003-12-01_02:59:30 on domain 1: 3.22432 elapsed seconds.
== WRF sent atm fields to ROMS
ROMS recv Atm fields
ROMS sent data to WRF
== WRF recvd ocean fields
Timing for main: time 2003-12-01_03:00:00 on domain 1: 3.37826 elapsed seconds.
 1 180 52974 03:00:00 7.572272E-03 2.292070E+04 2.292070E+04 2.090612E+16
Timing for main: time 2003-12-01_03:00:30 on domain 1: 4.32462 elapsed seconds.
Timing for main: time 2003-12-01_03:01:00 on domain 1: 3.13535 elapsed seconds.
Timing for main: time 2003-12-01_03:01:30 on domain 1: 3.13518 elapsed seconds.
Timing for main: time 2003-12-01_03:02:00 on domain 1: 3.13532 elapsed seconds.
Timing for main: time 2003-12-01_03:02:30 on domain 1: 3.14092 elapsed seconds.
Timing for main: time 2003-12-01_03:03:00 on domain 1: 3.13915 elapsed seconds.
  WRT_HIS - wrote history fields (Index=1,1) into time record = 0000004
Timing for main: time 2003-12-01_03:03:30 on domain 1: 3.14202 elapsed seconds.
 1 181 52974 03:01:00 7.572767E-03 2.292069E+04 2.292070E+04 2.090612E+16
Timing for main: time 2003-12-01_03:04:00 on domain 1: 3.14100 elapsed seconds.
 1 182 52974 03:02:00 7.573251E-03 2.292069E+04 2.292070E+04 2.090611E+16
Timing for main: time 2003-12-01_03:04:30 on domain 1: 3.15045 elapsed seconds.
 1 183 52974 03:03:00 7.573725E-03 2.292069E+04 2.292069E+04 2.090611E+16
Timing for main: time 2003-12-01_03:05:00 on domain 1: 3.15188 elapsed seconds.
 1 184 52974 03:04:00 7.574191E-03 2.292068E+04 2.292069E+04 2.090610E+16
Timing for main: time 2003-12-01_03:05:30 on domain 1: 3.14818 elapsed seconds.

ISO8---*** XEmacs: sc81.out (Fundamental) ---L1618--C43--1%
```

"Timing for ...." = WRF

"1 179 52974 02:59:00 " = ROMS

Here is where the model coupling synchronization occurs.  
so probably could re-allocate more nodes to WRF

# Data field exchange min/max values

```
jwarner@poseidon-l1:/vortexfs1/scratch/jwarner/coawst3.4_test10
Timing for main: time 2012-10-28_13:30:00 on domain 2: 0.19166 elapsed seconds
## ROMS grid 1 recv data from WRF grid 1
WRFtoROMS Min/Max GSW (Wm-2): 0.000000E+00 4.476423E+02
WRFtoROMS Min/Max GLW (Wm-2): 0.000000E+00 4.406325E+02
WRFtoROMS Min/Max LH (Wm-2): 0.000000E+00 8.354071E+02
WRFtoROMS Min/Max HFX (Wm-2): -1.985328E+01 2.440330E+02
WRFtoROMS Min/Max USTRESS (Nm-2): -9.503103E-01 1.176217E+00
WRFtoROMS Min/Max VSTRESS (Nm-2): -8.051951E-01 7.339598E-01
WRFtoROMS Min/Max MSLP (mb): 0.000000E+00 1.020593E+03
WRFtoROMS Min/Max RELH (-): 0.000000E+00 9.506385E-01
WRFtoROMS Min/Max T2 (C): 0.000000E+00 2.687080E+01
WRFtoROMS Min/Max RAIN (kgm-2s-1): 0.000000E+00 1.318768E-03
WRFtoROMS Min/Max EVAP (kgm-2s-1): 0.000000E+00 3.425169E-04
## WRF grid 1 sent data to ROMS grid 1
## WRF grid 2 sent data to ROMS grid 1
## ROMS grid 1 recv data from WRF grid 2
WRFtoROMS Min/Max GSW (Wm-2): 0.000000E+00 4.179347E+02
WRFtoROMS Min/Max GLW (Wm-2): 0.000000E+00 4.444828E+02
WRFtoROMS Min/Max LH (Wm-2): 0.000000E+00 8.468719E+02
WRFtoROMS Min/Max HFX (Wm-2): -1.604046E+01 2.516284E+02
WRFtoROMS Min/Max USTRESS (Nm-2): -1.651402E+00 1.887024E+00
WRFtoROMS Min/Max VSTRESS (Nm-2): -1.810099E+00 1.265032E+00
WRFtoROMS Min/Max MSLP (mb): 0.000000E+00 1.003535E+03
WRFtoROMS Min/Max RELH (-): 0.000000E+00 9.839446E-01
WRFtoROMS Min/Max T2 (C): 0.000000E+00 2.655765E+01
WRFtoROMS Min/Max RAIN (kgm-2s-1): 0.000000E+00 1.059673E-02
WRFtoROMS Min/Max EVAP (kgm-2s-1): 0.000000E+00 3.472175E-04
## ROMS grid 2 recv data from WRF grid 1
WRFtoROMS Min/Max GSW (Wm-2): 0.000000E+00 1.476153E+02
WRFtoROMS Min/Max GLW (Wm-2): 0.000000E+00 4.187577E+02
WRFtoROMS Min/Max LH (Wm-2): 0.000000E+00 4.920473E+02
WRFtoROMS Min/Max HFX (Wm-2): -2.440168E+01 9.009011E+01
WRFtoROMS Min/Max USTRESS (Nm-2): -9.670363E-01 0.000000E+00
WRFtoROMS Min/Max VSTRESS (Nm-2): -6.882157E-01 0.000000E+00
WRFtoROMS Min/Max MSLP (mb): 0.000000E+00 1.010489E+03
WRFtoROMS Min/Max RELH (-): 0.000000E+00 9.404117E-01
WRFtoROMS Min/Max T2 (C): 0.000000E+00 2.336388E+01
WRFtoROMS Min/Max RAIN (kgm-2s-1): 0.000000E+00 1.126228E-03
WRFtoROMS Min/Max EVAP (kgm-2s-1): 0.000000E+00 2.017394E-04
## ROMS grid 2 recv data from WRF grid 2
WRFtoROMS Min/Max GSW (Wm-2): 0.000000E+00 3.141597E+02
WRFtoROMS Min/Max GLW (Wm-2): 0.000000E+00 4.400405E+02
WRFtoROMS Min/Max LH (Wm-2): 0.000000E+00 7.637908E+02
WRFtoROMS Min/Max HFX (Wm-2): -2.377396E+01 2.101876E+02
WRFtoROMS Min/Max USTRESS (Nm-2): -1.668581E+00 2.084062E-01
WRFtoROMS Min/Max VSTRESS (Nm-2): -1.696622E+00 2.368454E-01
WRFtoROMS Min/Max MSLP (mb): 0.000000E+00 1.003199E+03
WRFtoROMS Min/Max RELH (-): 0.000000E+00 9.717784E-01
WRFtoROMS Min/Max T2 (C): 0.000000E+00 2.558522E+01
WRFtoROMS Min/Max RAIN (kgm-2s-1): 0.000000E+00 5.698533E-03
WRFtoROMS Min/Max EVAP (kgm-2s-1): 0.000000E+00 3.131542E-04
## ROMS grid 1 sent data to WRF grid 1
## ROMS grid 1 sent data to WRF grid 2
## ROMS grid 2 sent data to WRF grid 1
## ROMS grid 2 sent data to WRF grid 2
## WRF grid 1 sent data to ROMS grid 2
```

WRF 1 to ROMS 1

WRF 2 to ROMS 1

WRF 1 to ROMS 2

WRF 2 to ROMS 2