# Coupling_WRF_CROCO_WW3_with_OASIS

Notes on coupling WRF, CROCO, and WW3 using OASIS for a UBUNTU 18.04 machine for the Benguela case

A mix of "Documentation for coupling with OASIS in CROCO, WRF, WW3" by Swen JULLIEN, Gildas CAMBON (March 7, 2018) and
personal experience. Most ideas were contributed by Swen Jullien, Gildas Cambon, Lionel Renault, and Rachid Benshila.

## Details

Time step in WRF and CROCO has to be a multiple of the coupling frequency (nancouple). For example
180s (WRF), 3600 (CROCO), 3600 (OASIS).

Order of dimensions x - y. (41 42 for Benguela)
Use values that appear in in oce.nc and atm.nc

mpirun -np 2 ./croco : -np 2 ./wrf.exe

CPLMASK to fit ocean model in WRF model.

CROCO domain should be smaller than the WRF domain, to avoit the sponge layer in WRF, by at leas five grid points (LR), and to avoid different SST influence in WRF due to drift in the coupled (vs non-coupled) regions.

WRF should be processed with SST data. Given that sst_update = 1, is in the namelist.input, the non-coupled part of the WRF domain expects SST info in the auxiliar input files.

# .bashrc

```console
export NETCDF_LIBDIR=/home/mosa/netcdf-3.6.3/lib
export NETCDF_INCDIR=/home/mosa/netcdf-3.6.3/include

export DIR=/home/mosa/libraries
export CC=gcc
export CXX=g++
export FC=gfortran
export FCFLAGS=-m64
export F77=gfortran
export FFLAGS=-m64

export PATH=$DIR/netcdf/bin:$PATH
export NETCDF=$DIR/netcdf

export LDFLAGS=-L$DIR/grib2/lib
export CPPFLAGS=-I$DIR/grib2/include

export NETCDF_classic=1

export JASPERLIB=$DIR/grib2/lib
export JASPERINC=$DIR/grib2/include

export WRFIO_NCD_LARGE_FILE_SUPPORT=1
```

# OASIS

Download OASIS-MCT version 3

```console
cd oasis
svn checkout http://oasis3mct.cerfacs.fr/svn/branches/OASIS3-MCT_3.0_branch
cd $HOME/oasis/OASIS3-MCT_3.0_branch/oasis3-mct/util/make_dir
make realclean -f TopMakefileOasis3 > oasis_clean.out
make -f TopMakefileOasis3 > oasis_make.out
```
Check directory

```console
compile_oa3-mct
```
and file oasis_make.out to see if everything went ok

## OASIS - namcouple file

```console
#########################################################################
# This is a typical input file for OASIS3-MCT.
# Keywords used in previous versions of OASIS3
# but now obsolete are marked "Not used"
# Don't hesitate to ask precisions or make suggestions (oasishelp@cerfacs.fr).
#
# Any line beginning with # is ignored. Blank lines are not allowed.
#
#########################################################################
#
# NFIELDS: total number of fields being exchanged
 $NFIELDS
 9
#########################################################################
# NBMODEL: number of models and their names (6 characters)
 $NBMODEL
 2  wrfexe  crocox
###########################################################################
# RUNTIME: total simulated time for the actual run in seconds (<I8)
 $RUNTIME
 86400
###########################################################################
# NLOGPRT: debug and time statistics informations printed in log file
#          First number: 0 one log file for master, and one for other procs
#                        1 one log file for master, and one for other errors
#                        2 one file per proc with normal diagnostics
#                        5 as 2 + initial debug info
#                        10 as 5 + routine calling tree
#                        12 as 10 + some routine calling notes
#                        15 as 12 + even more debug diagnostics
#                        20 as 15 + some extra runtime analysis
#                        30 full debug information
#          Second number: time statistics
#                      0 nothing calculated
#                      1 one file for proc 0 and min/max of other procs
#                      2 as 1 + one file per proc
#                      3 as 2 + proc 0 writes all procs results in its file
```

```
 $NLOGPRT
 1 1
###########################################################################
# Beginning of fields exchange definition
 $STRINGS
#
# For each exchanged field:
#
# line 1: field in sending model, field in target model, unused, coupling
#         period, number of transformation, restart file, field status
# line 2: nb of pts for sending model grid (without halo) first dim, and second dim,
#         for target grid first dim, and second dim, sending model grid name, target
#         model grid name, lag = time step of sending model
# line 3: sending model grid periodical (P) or regional (R), and nb of overlapping
#         points, target model grid periodical (P) or regional (R), and number of
#         overlapping points
# line 4: list of transformations performed
# line 5: parameters for each transformation
#
# See the correspondances between variables in models and in OASIS:
# Note: for CROCO and WRF nesting capability is useable in coupled
#       mode. For CROCO the domain in defined by the last number
#       of coupled field name. For WRF, WRF domain is defined by
#       the number after WRF_d, and the domain of the coupled model
#       (CROCO for example) is given by EXT_d in coupled field name
#
# |--------------------------------------------------------------|
# | Possibly sent fields by CROCO:                 CROCO | OASIS |
# |--------------------------------------------------------------|
# |     t(:,:,N,nnew,itemp)  |    SRMSSTV0                       |
# |                   zeta   |    SRMSSHV0                       |
# |     u v (at rho points)  |    SRMVOCE0 SRMUOCE0              |
# |--------------------------------------------------------------|
# | Possibly received fields by CROCO:            CROCO | OASIS  |
# |--------------------------------------------------------------|
# |                  srflx   |    RRMSRFL0                       |
# |       stflx(:,:,isalt)   |    RRMEVPR0                       |
# |      stflx(,:,:,itemp)   |    RRMSTFL0                       |
# |                  sustr   |    RRMTAUX0                       |
# |                  svstr   |    RRMTAUY0                       |
# |                  smstr   |    RRMTAUM0                       |
# |                  whrm    |    RRM__HS0                       |
# |                  wfrq    |    RRMT0M10                       |
# |                  wdrx    |    RRMCDIR0                       |
# |                  wdre    |    RRMSDIR0                       |
# |--------------------------------------------------------------|
# | Possibly sent fields by WW3:                    WW3 | OASIS  |
# |--------------------------------------------------------------|
# |            not defined   |    WW3_ODRY                       |
# |                   T0M1   |    WW3_T0M1                       |
# |                     HS   |    WW3__OHS                       |
# |                    DIR   |    WW3_CDIR WW3_SDIR              |
# |                    BHD   |    WW3__BHD                       |
# |                    TWO   |    WW3_TWOX WW3_TWOY              |
# |                    UBR   |    WW3__UBR                       |
# |                    FOC   |    WW3__FOC                       |
# |                    TAW   |    WW3_TAWX WW3_TAWY              |
# |                     LM   |    WW3___LM                       |
# |                    CUR   |    WW3_WSSU WW3_WSSV              |
# |                    CHA   |    WW3__CHA                       |
# |                     HS   |    WW3__AHS                       |
# |                     FP   |    WW3___FP                       |
# |--------------------------------------------------------------|
# | Possibly received fields by WW3:                WW3 | OASIS  |
# |--------------------------------------------------------------|
# |            not defined   |    WW3_OWDH WW3_OWDU WW3_OWDV     |
# |                    SSH   |    WW3__SSH                       |
# |                    CUR   |    WW3_OSSU WW3_OSSV              |
# |                    WND   |    WW3__U10 WW3__V10              |
# |--------------------------------------------------------------|
# | Possibly sent fields by WRF:                    WRF | OASIS  |
# |--------------------------------------------------------------|
# |                GSW   |    WRF_d01_EXT_d01_SURF_NET_SOLAR     |
# |        QFX-(RAINCV                                           |
# |       +RAINNCV)/DT   |    WRF_d01_EXT_d01_EVAP-PRECIP        |
# |   GLW-STBOLT*EMISS                                           |
# |     *SST**4-LH-HFX   |    WRF_d01_EXT_d01_SURF_NET_NON-SOLAR |
# | taut * u_uo / wspd   |    WRF_d01_EXT_d01_TAUX               |
# | taut * u_uo / wspd   |    WRF_d01_EXT_d01_TAUY               |
# |               taut   |    WRF_d01_EXT_d01_TAUMOD             |
# |               u_uo   |    WRF_d01_EXT_d01_U_01               |
# |               v_vo   |    WRF_d01_EXT_d01_V_01               |
# |--------------------------------------------------------------|
# | Possibly received fields by WRF:                WRF | OASIS  |
# |--------------------------------------------------------------|
# |                    SST   |    WRF_d01_EXT_d01_SST            |
# |                   UOCE   |    WRF_d01_EXT_d01_UOCE           |
# |                   VOCE   |    WRF_d01_EXT_d01_VOCE           |
# |               CHA_COEF   |    WRF_d01_EXT_d01_CHA_COEF       |
# |--------------------------------------------------------------|
#
#                     ------------------------------------
#                       WRF (wrfexe) ==> CROCO (crocox)
#                     ------------------------------------
#
#~~~~~~~~~~~
# TAUX : zonal stress (N.m-2)
#~~~~~~~~~~~
WRF_d01_EXT_d01_TAUX RRMTAUX0 1 3600 1 atm.nc EXPORTED
100 117 41 42 atmt ocnt  LAG=180
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# TAUY : meridional stress (N.m-2)
#~~~~~~~~~~~
WRF_d01_EXT_d01_TAUY RRMTAUY0 1 3600 1 atm.nc EXPORTED
100 117 41 42 atmt ocnt  LAG=180
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# TAUMOD : stress module (N.m-2)
#~~~~~~~~~~~
WRF_d01_EXT_d01_TAUMOD RRMTAUM0 1 3600 1 atm.nc EXPORTED
100 117 41 42 atmt ocnt  LAG=180
```

```
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# EVAP-PRECIP : E-P flux (kg.m-2.s-1)
#~~~~~~~~~~~
WRF_d01_EXT_d01_EVAP-PRECIP RRMEVPR0 1 3600 1 atm.nc EXPORTED
100 117 41 42 atmt ocnt  LAG=180
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# NET SOLAR FLUX (W.m-2)
#~~~~~~~~~~~
WRF_d01_EXT_d01_SURF_NET_SOLAR RRMSRFL0 1 3600 1 atm.nc EXPORTED
100 117 41 42 atmt ocnt  LAG=180
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# NET NON-SOLAR FLUX (W.m-2)
#~~~~~~~~~~~
WRF_d01_EXT_d01_SURF_NET_NON-SOLAR RRMSTFL0 1 3600 1 atm.nc EXPORTED
100 117 41 42 atmt ocnt  LAG=180
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#                      ------------------------------------
#                        CROCO (crocox) ==> WRF (wrfexe)
#                      ------------------------------------
#
#~~~~~~~~~~~
# SST (K)
#~~~~~~~~~~~
SRMSSTV0 WRF_d01_EXT_d01_SST 1 3600 1 oce.nc EXPORTED
41 42 100 117 ocnt atmt LAG=3600
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# UOCE : zonal current (m.s-1)
#~~~~~~~~~~~
SRMUOCE0 WRF_d01_EXT_d01_UOCE 1 3600 1 oce.nc EXPORTED
41 42 100 117 ocnt atmt LAG=3600
R  0  R  0
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
#~~~~~~~~~~~
# VOCE : meridonal current (m.s-1)
#~~~~~~~~~~~
SRMVOCE0 WRF_d01_EXT_d01_VOCE 1 3600 1 oce.nc EXPORTED
41 42 100 117 ocnt atmt LAG=3600
R  0  R  0
```

```
SCRIPR
BILINEAR LR SCALAR LATLON 1 4
#
############################################################################
$END
```

## OASIS - Create OASIS grid files form WRF

Use: ./create_oasis_grids_for_wrf.sh wrfinput_d01 /home/mosa/COUPLED_BENGUELA/

```console
#!/bin/bash
set -x
## ------------------------------------------------------------------------------ #
## - Create grids.nc, masks.nc, files from WRF for oasis                        - #
## - because call to oasis_grid function not yet implemented in WRF             - #
##                                                                              - #
## Mandatory inputs:                                                            - #
##  - a file from WRF containing lon,lat,mask (with full path)                  - #
##  - the output destination directory                                          - #
##                                                                              - #
## ------------------------------------------------------------------------------ #
#
# Further Information:
# http://www.croco-ocean.org
#
# This file is part of CROCOTOOLS
#
# CROCOTOOLS is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published
# by the Free Software Foundation; either version 2 of the License,
# or (at your option) any later version.
#
# CROCOTOOLS is distributed in the hope that it will be useful, but
# WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place, Suite 330, Boston,
# MA  02111-1307  USA
#
# Copyright (c) 2018 S. Jullien
# swen.jullien@ifremer.fr
## ------------------------------------------------------------------------------ #

# ------------------------------------------------------------------------------- #

gridfile=$1
mydir=$2

# ------------------------------------------------------------------------------- #

echo '********************************************'
echo 'START script create_oasis_grids_for_wrf.sh'
echo '********************************************'
echo ' '
```

```
# First check if inputs are ok
if [[ -z $gridfile ]] || [[ -z $mydir ]] ; then
    echo 'ERROR: inputs are not correctly specified.'
    echo '       this script needs 2 inputs:'
    echo '         - a file from WRF containing the mask, lon and lat (with full
path)'
    echo '       - the output destination directory'
    echo ' Exit...'
    echo ' '
    exit 1
fi
# ------------------------------------------------------------------------------ #

mytmpgrd=$mydir/grd_tmp.nc
grdfile=$mydir/grids.nc
mskfile=$mydir/masks.nc

wrflon=XLONG
wrflat=XLAT
wrfmask=LANDMASK

# Extract lon,lat,mask
echo '---> Extract '${wrflon}', '${wrflat}', and '${wrfmask}' variables...'
ncks -O -v ${wrflon},${wrflat},${wrfmask} -d Time,0 $gridfile ${mytmpgrd}

# Put them on the stag grid
echo '---> Put them on the stag grid'
./to_wrf_stag_grid.sh ${mytmpgrd} ${mytmpgrd}

# remove time dimension
echo '---> Remove time dimension...'
ncwa -O -a Time ${mytmpgrd} ${mytmpgrd}

# compute the last lon and lat
Nlon=`ncdump -h $gridfile | grep "west_east = " | cut -d '=' -f2 | cut -d ';' -f1`
Nlon=${Nlon// /}
Nlat=`ncdump -h $gridfile | grep "south_north = " | cut -d '=' -f2 | cut -d ';' -f1`
Nlat=${Nlat// /}
Nlonstag=$(($Nlon + 1))
Nlatstag=$(($Nlat + 1))
Nlonm1=$(($Nlon - 1))
Nlatm1=$(($Nlat - 1))
echo '---> compute the last lon...'
ncap2 -F -O -s "${wrflon}(:,$Nlonstag)=${wrflon}(:,$Nlon)+(${wrflon}(:,$Nlon)-
${wrflon}(:,$Nlonm1))" ${mytmpgrd} ${mytmpgrd}
echo '---> compute the last lat...'
ncap2 -F -O -s "${wrflat}($Nlatstag,:)=${wrflat}($Nlat,:)+(${wrflat}($Nlat,:)-
${wrflat}($Nlatm1,:))" ${mytmpgrd} ${mytmpgrd}

# change mask from float to integer
echo '---> Change mask from float to integer...'
ncap2 -O -s "${wrfmask}=int(${wrfmask})" ${mytmpgrd} ${mytmpgrd}

# rename dimensions
echo '---> rename dimensions...'
ncrename -d west_east,x_atmt -d south_north,y_atmt ${mytmpgrd}
# rename variables
echo '---> Rename variables...'
ncrename -v ${wrfmask},atmt.msk  -v ${wrflat},atmt.lat ${mytmpgrd}
ncrename -v ${wrflon},atmt.lon ${mytmpgrd}

# create grid file
echo '---> Ceate grid file...'
echo '======================='
ncks -O -v atmt.lon,atmt.lat ${mytmpgrd} ${grdfile}
ncatted -h -O -a ,global,d,, ${grdfile} ${grdfile}
ncatted -h -O -a ,atmt.lon,d,, ${grdfile} ${grdfile}
ncatted -h -O -a ,atmt.lat,d,, ${grdfile} ${grdfile}

# create mask file
echo '---> Create mask file...'
echo '========================='
ncks -O -v atmt.msk ${mytmpgrd} ${mskfile}
ncatted -h -O -a ,global,d,, ${mskfile} ${mskfile}
ncatted -h -O -a ,atmt.msk,d,, ${mskfile} ${mskfile}

rm ${mytmpgrd}

echo 'DONE: grids.wrf.nc and masks.wrf.nc have been created in '$mydir
echo ' '
```

## OASIS - Create restart from calm conditions

Use:

```console
export varlist='WRF_d01_EXT_d01_SURF_NET_SOLAR WRF_d01_EXT_d01_EVAP-PRECIP
WRF_d01_EXT_d01_SURF_NET_NON-SOLAR WRF_d01_EXT_d01_TAUX WRF_d01_EXT_d01_TAUY
WRF_d01_EXT_d01_TAUMOD WRF_d01_EXT_d01_U_01 WRF_d01_EXT_d01_V_01'
   ./create_oasis_restart_from_calm_conditions.sh wrfinput_d01 atm.nc wrf "$varlist"


export varlist='WW3_T0M1 WW3__OHS WW3_CDIR WW3_SDIR WW3__CHA WW3_TAWX WW3_TAWY
WW3_TWOX WW3_TWOY'
   ./create_oasis_restart_from_calm_conditions.sh $ww3file wav.nc ww3 "$varlist"


export varlist='SRMSSTV0 SRMSSHV0 SRMVOCE0 SRMUOCE0'
   ./create_oasis_restart_from_calm_conditions.sh croco_grd.nc oce.nc croco
"$varlist"
```

```console
#!/bin/bash -e

## ---------------------------------------------------------------------------- #
## - Create restart file for oasis                                            - #
## - with all variables set to 0                                              - #
##                                                                            - #
## Mandatory inputs:                                                          - #
##  - a file from this model containing the mask (with full path)             - #
##  - the oasis restart file name (with full path)                            - #
##  - the model: wrf, croco, or ww3 cases are accepted                        - #
##  - the list of variables that have to be generated in this restart file    - #
##                                                                            - #
## ---------------------------------------------------------------------------- #
#
```

```
# Further Information:
# http://www.croco-ocean.org
#
# This file is part of CROCOTOOLS
#
# CROCOTOOLS is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published
# by the Free Software Foundation; either version 2 of the License,
# or (at your option) any later version.
#
# CROCOTOOLS is distributed in the hope that it will be useful, but
# WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place, Suite 330, Boston,
# MA  02111-1307  USA
#
# Copyright (c) 2018 S. Jullien
# swen.jullien@ifremer.fr
## ------------------------------------------------------------------------------ #


# ------------------------------------------------------------------------------- #

filein=$1
fileout=$2
model=$3
varlist=$4

# ------------------------------------------------------------------------------- #
echo '**********************************************************'
echo 'START script create_oasis_restart_from_calm_conditions.sh'
echo '**********************************************************'
echo ' '

# First check if inputs are ok
if [[ -z $filein ]] || [[ -z $fileout ]] || [[ -z $model ]] || [[ -z $varlist ]] ;
then
    echo 'ERROR: inputs are not correctly specified.'
    echo '       this script needs 4 inputs:'
    echo '       - a file from this model containing the mask (with full path)'
    echo '       - the oasis restart file name (with full path)'
    echo '       - the model: wrf croco or ww3 cases are accepted'
    echo '        - the list of variables that have to be generated in this restart
file'
    echo ' Exit...'
    echo ' '
    exit 1
fi
# ------------------------------------------------------------------------------- #

mydir=$(dirname "$fileout")
filetmp=$mydir/rst_tmp.nc

echo 'Initialize restart file '$fileout' to 0 for variables: '$varlist
echo '=========================================================================='
```

```
if [ $model == wrf ] ; then

    # Extract mask
    echo '---> Extract LANDMASK variable...'
    ncks -O -v LANDMASK -d Time,0 $filein ${filetmp}

    # Put it on the stag grid
    echo '---> Put it on the stag grid'
    ./to_wrf_stag_grid.sh ${filetmp} ${filetmp}

    # remove time dimension
    echo '---> Remove time dimension...'
    ncwa -O -a Time ${filetmp} ${filetmp}

    # set the variable to 0 and rename it var0
    echo '---> Set the variable to 0 and rename it var0...'
    ncap2 -O -v -s "var0=LANDMASK*0" ${filetmp} ${filetmp}

elif  [ $model == croco ] ; then

    # Extract dimensions
    xi_rho=`ncdump -h $filein | grep "xi_rho = " | cut -d '=' -f2 | cut -d ';' -f1`
    xi_rho=${xi_rho// /}
    eta_rho=`ncdump -h $filein | grep "eta_rho = " | cut -d '=' -f2 | cut -d ';' -
f1`
    eta_rho=${eta_rho// /}

    # Extract mask
    echo '---> Extract mask_rho variable (only interior grid indices, i.e. in
fortran convention : 2:end-1)...'
    ncks -O -F -d xi_rho,2,$((${xi_rho}-1)) -d eta_rho,2,$((${eta_rho}-1)) -v
mask_rho $filein ${filetmp}

    # set the variable to 0 and rename it var0
    echo '---> Set the variable to 0 and rename it var0...'
    ncap2 -O -v -s "var0=mask_rho*0" ${filetmp} ${filetmp}

elif  [ $model == ww3 ] ; then

    # Extract mask
    echo '---> Extract MAPSTA variable...'
    ncks -O -3 -v MAPSTA $filein ${filetmp}

    # set the variable to 0 and rename it var0
    echo '---> Set the variable to 0 and rename it var0...'
    ncap2 -O -v -s "var0=MAPSTA*0" ${filetmp} ${filetmp}

else

    echo 'ERROR: '$model' case is not implemented yet. Exit...'
    echo ' '
    exit 1

fi # model

# START LOOP on varlist #
#-----------------------#
for var in $varlist ; do
```

```
    echo ' '
    echo '================================='
    echo 'Create variable: '$var'...'
    echo '================================='
    ncks -A -v var0 ${filetmp} $fileout
    ncrename -v var0,$var $fileout

done
rm ${filetmp}

# remove global attributes
ncatted -h -O -a ,global,d,, $fileout $fileout

echo 'DONE for '$varlist' variables => initialized to 0'
echo ' '
```

# WRF

## WRF - Area used (larger than CROCO domain) - Deal overlap with CPLMASK

```console
&geogrid
 parent_id         = 1,
 parent_grid_ratio = 1,
 i_parent_start    = 1,
 j_parent_start    = 1,
 e_we          = 100,
 e_sn          = 117,
 geog_data_res = '10m',
 dx = 30000,
 dy = 30000,
 map_proj =  'lambert',
 ref_lat   = -27.677,
 ref_lon   = 15.892,
 truelat1  = -27.677,
 truelat2  = -27.677,
 stand_lon = 15.892,
 geog_data_path = '/home/mosa/WPS/WPS_GEOG',
 opt_geogrid_tbl_path = '/home/mosa/Domains/Benguela/',
 ref_x = 50.0,
 ref_y = 58.5,
/
```

## WRF - configure.wrf

### Add

```
OA3MCT_ROOT_DIR = /home/mosa/compile_oa3-mct
```

### Change

```
From: CFLAGS          =    $(CFLAGS_LOCAL) -DDM_PARALLEL   \

To:   CFLAGS          =    $(CFLAGS_LOCAL) -DDM_PARALLEL   -DSTUBMPI \
From:  ARCH_LOCAL      =       -DNONSTANDARD_SYSTEM_SUBR  -DWRF_USE_CLM

To:      ARCH_LOCAL        =          -DNONSTANDARD_SYSTEM_FUNC   -DWRF_USE_CLM  -
Dkey_cpp_oasis3

From:                     -I$(WRF_SRC_ROOT_DIR)/chem -I$(WRF_SRC_ROOT_DIR)/inc \
                          -I$(NETCDFPATH)/include \

To:                       -I$(WRF_SRC_ROOT_DIR)/chem -I$(WRF_SRC_ROOT_DIR)/inc \
                         -I$(OA3MCT_ROOT_DIR)/build/lib/psmile.MPI1 \
                         -I$(OA3MCT_ROOT_DIR)/build/lib/mct \
                          -I$(NETCDFPATH)/include \

From:  LIB_EXTERNAL    = \
                          -L$(WRF_SRC_ROOT_DIR)/external/io_netcdf  -lwrfio_nf  -L/home/
mosa/libraries/netcdf/lib -lnetcdff -lnetcdf

To: LIB_EXTERNAL    = \
                          -L$(WRF_SRC_ROOT_DIR)/external/io_netcdf  -lwrfio_nf  -L/home/
mosa/libraries/netcdf/lib -lnetcdff -lnetcdf    \
                         -L$(OA3MCT_ROOT_DIR)/lib -lpsmile.MPI1 -lmct -lmpeu -lscrip
```

~~~
```console
# configure.wrf
#
# Original configure options used:
# ./configure
# Compiler choice: 34
# Nesting option: 1
#
# This file was automatically generated by the configure script in the
# top level directory. You may make changes to the settings in this
# file but be aware they will be overwritten each time you run configure.
# Ordinarily, it is necessary to run configure once, when the code is
# first installed.
#
# To permanently change options, change the settings for your platform
# in the file arch/configure.defaults then rerun configure.
#
SHELL           =       /bin/sh
DEVTOP          =       `pwd`
LIBINCLUDE      =       .
.SUFFIXES: .F .i .o .f90 .c

#### Get core settings from environment (set in compile script)
#### Note to add a core, this has to be added to.

COREDEFS = -DEM_CORE=$(WRF_EM_CORE) \
           -DNMM_CORE=$(WRF_NMM_CORE) -DNMM_MAX_DIM=2600 \
          -DDA_CORE=$(WRF_DA_CORE) \
          -DWRFPLUS=$(WRF_PLUS_CORE)

#### Single location for defining total number of domains.  You need
#### at least 1 + 2*(number of total nests).  For example, 1 coarse
#### grid + three fine grids = 1 + 2(3) = 7, so MAX_DOMAINS=7.

MAX_DOMAINS    =      21
~~~

```
#### DM buffer length for the configuration flags.

CONFIG_BUF_LEN =       65536

#### Size of bitmasks (in 4byte integers) of stream masks for WRF I/O

MAX_HISTORY = 25

IWORDSIZE = 4
DWORDSIZE = 8
LWORDSIZE = 4

OA3MCT_ROOT_DIR = /home/mosa/compile_oa3-mct


##############################################################################
#### The settings in this section are defaults that may be overridden by the
#### architecture-specific settings in the next section.
##############################################################################

##############################################################################
#### NOTE:  Do not modify these default values here.  To override these
####        values, make changes after "Architecture specific settings".
##############################################################################

#### Native size (in bytes) of Fortran REAL data type on this architecture ####
#### Note:  to change real wordsize (for example, to promote REALs from
####        4-byte to 8-byte), modify the definition of RWORDSIZE in the
####        section following "Architecture specific settings".  Do not
####        change NATIVE_RWORDSIZE as is it architecture-specific.
NATIVE_RWORDSIZE = 4

#### Default sed command and script for Fortran source files ####
#SED_FTN = sed -f $(WRF_SRC_ROOT_DIR)/arch/standard.sed
SED_FTN = $(WRF_SRC_ROOT_DIR)/tools/standard.exe

# Hack to work around $(PWD) not changing during OSF1 build.
# $(IO_GRIB_SHARE_DIR) is reset during the OSF1 build only.
IO_GRIB_SHARE_DIR =

#### ESMF switches                 ####
#### These are set up by Config.pl ####
# switch to use separately installed ESMF library for coupling:  1==true
ESMF_COUPLING       = 0
# select dependences on module_utility.o
ESMF_MOD_DEPENDENCE = $(WRF_SRC_ROOT_DIR)/external/esmf_time_f90/module_utility.o
# select -I options for external/io_esmf vs. external/esmf_time_f90
ESMF_IO_INC         = -I$(WRF_SRC_ROOT_DIR)/external/esmf_time_f90
# select -I options for separately installed ESMF library, if present
ESMF_MOD_INC        =  $(ESMF_IO_INC)
# select cpp token for external/io_esmf vs. external/esmf_time_f90
ESMF_IO_DEFS        =
# select build target for external/io_esmf vs. external/esmf_time_f90
ESMF_TARGET         = esmf_time

# ESMFINCLUDEGOESHERE
```

```
#### NETCDF4 pieces

NETCDF4_IO_OPTS = -DUSE_NETCDF4_FEATURES -DWRFIO_NCD_LARGE_FILE_SUPPORT
GPFS            =
CURL            =
HDF5            =
ZLIB            =
DEP_LIB_PATH    =
NETCDF4_DEP_LIB = $(DEP_LIB_PATH) $(HDF5) $(ZLIB) $(GPFS) $(CURL)

# NETCDF4INCLUDEGOESHERE


##############################################################################

LIBWRFLIB = libwrflib.a


#### Architecture specific settings ####

# Settings for    Linux x86_64 ppc64le, gfortran compiler with gcc   (dmpar)
#
DESCRIPTION     =       GNU ($SFC/$SCC)
DMPARALLEL      =        1
OMPCPP          =        -D_OPENMP
OMP             =        -fopenmp
OMPCC           =        -fopenmp
SFC             =       gfortran
SCC             =       gcc
CCOMP           =       gcc
DM_FC           =       mpif90
DM_CC           =       mpicc
FC              =       time $(DM_FC)
CC              =       $(DM_CC) -DFSEEKO64_OK
LD              =       $(FC)
RWORDSIZE       =       $(NATIVE_RWORDSIZE)
PROMOTION       =       #-fdefault-real-8
ARCH_LOCAL      =       -DNONSTANDARD_SYSTEM_FUNC  -DWRF_USE_CLM -Dkey_cpp_oasis3
CFLAGS_LOCAL    =       -w -O3 -c
LDFLAGS_LOCAL   =
CPLUSPLUSLIB    =
ESMF_LDFLAG     =       $(CPLUSPLUSLIB)
FCOPTIM         =       -O2 -ftree-vectorize -funroll-loops
FCREDUCEDOPT   =       $(FCOPTIM)
FCNOOPT        =       -O0
FCDEBUG                   =              #  -g  $(FCNOOPT)  #  -ggdb  -fbacktrace  -fcheck=bounds,do,mem,pointer -ffpe-trap=invalid,zero,overflow
FORMAT_FIXED    =       -ffixed-form
FORMAT_FREE     =       -ffree-form -ffree-line-length-none
FCSUFFIX        =
BYTESWAPIO      =       -fconvert=big-endian -frecord-marker=4
FCBASEOPTS_NO_G =       -w $(FORMAT_FREE) $(BYTESWAPIO)
FCBASEOPTS      =       $(FCBASEOPTS_NO_G) $(FCDEBUG)
MODULE_SRCH_FLAG =
TRADFLAG        =      -traditional-cpp
CPP             =      /lib/cpp -P -nostdinc
AR              =      ar
ARFLAGS         =      ru
M4              =      m4 -G
```

```
RANLIB          =      ranlib
RLFLAGS        =
CC_TOOLS        =      $(SCC)

############################################################
######################
# POSTAMBLE

FGREP = fgrep -iq

ARCHFLAGS       =    $(COREDEFS) -DIWORDSIZE=$(IWORDSIZE) -DDWORDSIZE=$(DWORDSIZE) -
DRWORDSIZE=$(RWORDSIZE) -DLWORDSIZE=$(LWORDSIZE) \
                     $(ARCH_LOCAL) \
                     $(DA_ARCHFLAGS) \
                      -DDM_PARALLEL \
                       \
                      -DNETCDF \
                       \
                       \
                       \
                       \
                       \
                       \
                       \
                       \
                       -DLANDREAD_STUB=1 \
                       \
                       \
                      -DUSE_ALLOCATABLES \
                      -Dwrfmodel \
                      -DGRIB1 \
                      -DINTIO \
                      -DKEEP_INT_AROUND \
                      -DLIMIT_ARGS \
                      -DBUILD_RRTMG_FAST=1 \
                      -DSHOW_ALL_VARS_USED=0 \
                      -DCONFIG_BUF_LEN=$(CONFIG_BUF_LEN) \
                      -DMAX_DOMAINS_F=$(MAX_DOMAINS) \
                      -DMAX_HISTORY=$(MAX_HISTORY) \
                     -DNMM_NEST=$(WRF_NMM_NEST)
CFLAGS          =    $(CFLAGS_LOCAL) -DDM_PARALLEL  \
                      -DLANDREAD_STUB=1 \
                      -DMAX_HISTORY=$(MAX_HISTORY) -DNMM_CORE=$(WRF_NMM_CORE)
FCFLAGS         =    $(FCOPTIM) $(FCBASEOPTS)
ESMF_LIB_FLAGS  =
# ESMF 5 -- these are defined in esmf.mk, included above
 ESMF_IO_LIB     =    -L$(WRF_SRC_ROOT_DIR)/external/esmf_time_f90 -lesmf_time
ESMF_IO_LIB_EXT =    -L$(WRF_SRC_ROOT_DIR)/external/esmf_time_f90 -lesmf_time
INCLUDE_MODULES =    $(MODULE_SRCH_FLAG) \
                     $(ESMF_MOD_INC) $(ESMF_LIB_FLAGS) \
                      -I$(WRF_SRC_ROOT_DIR)/main \
                      -I$(WRF_SRC_ROOT_DIR)/external/io_netcdf \
                      -I$(WRF_SRC_ROOT_DIR)/external/io_int \
                      -I$(WRF_SRC_ROOT_DIR)/frame \
                      -I$(WRF_SRC_ROOT_DIR)/share \
                      -I$(WRF_SRC_ROOT_DIR)/phys \
                      -I$(WRF_SRC_ROOT_DIR)/wrftladj \
                      -I$(WRF_SRC_ROOT_DIR)/chem -I$(WRF_SRC_ROOT_DIR)/inc \
                     -I$(OA3MCT_ROOT_DIR)/build/lib/mct \
                      -I$(OA3MCT_ROOT_DIR)/build/lib/psmile.MPI1 \
                      -I$(NETCDFPATH)/include \

REGISTRY        =    Registry
CC_TOOLS_CFLAGS = -DNMM_CORE=$(WRF_NMM_CORE)

 LIB_BUNDLED     = \
                      $(WRF_SRC_ROOT_DIR)/external/fftpack/fftpack5/libfftpack.a \
                      $(WRF_SRC_ROOT_DIR)/external/io_grib1/libio_grib1.a \
                      $(WRF_SRC_ROOT_DIR)/external/io_grib_share/libio_grib_share.a
\
                      $(WRF_SRC_ROOT_DIR)/external/io_int/libwrfio_int.a \
                      $(ESMF_IO_LIB) \
                      $(WRF_SRC_ROOT_DIR)/external/RSL_LITE/librsl_lite.a \
                      $(WRF_SRC_ROOT_DIR)/frame/module_internal_header_util.o \
                      $(WRF_SRC_ROOT_DIR)/frame/pack_utils.o

 LIB_EXTERNAL    = \
                      -L$(WRF_SRC_ROOT_DIR)/external/io_netcdf -lwrfio_nf \
                      -L$(OA3MCT_ROOT_DIR)/lib -lpsmile.MPI1 -lmct -lmpeu -lscrip \
                      -L/home/mosa/libraries/netcdf/lib -lnetcdff -lnetcdf

LIB             =    $(LIB_BUNDLED) $(LIB_EXTERNAL) $(LIB_LOCAL) $(LIB_WRF_HYDRO)
LDFLAGS         =    $(OMP) $(FCFLAGS) $(LDFLAGS_LOCAL)
ENVCOMPDEFS     =
WRF_CHEM       =      0
CPPFLAGS        =    $(ARCHFLAGS) $(ENVCOMPDEFS) -I$(LIBINCLUDE) $(TRADFLAG)
NETCDFPATH      =    /home/mosa/libraries/netcdf
HDF5PATH        =
WRFPLUSPATH     =
RTTOVPATH       =
PNETCDFPATH     =

bundled:  io_only
external:        io_only         $(WRF_SRC_ROOT_DIR)/external/RSL_LITE/librsl_lite.a
gen_comms_rsllite module_dm_rsllite $(ESMF_TARGET)
io_only:  esmf_time wrfio_nf   \
         wrf_ioapi_includes wrfio_grib_share wrfio_grib1 wrfio_int fftpack


######################
externals: io_only bundled external

gen_comms_serial :
       ( /bin/rm -f $(WRF_SRC_ROOT_DIR)/tools/gen_comms.c )

module_dm_serial :
       ( if [ ! -e module_dm.F ] ; then /bin/cp module_dm_warning module_dm.F ; cat
module_dm_stubs.F >> module_dm.F ; fi )

gen_comms_rsllite :
       ( if [ ! -e $(WRF_SRC_ROOT_DIR)/tools/gen_comms.c ] ; then \
          /bin/cp  $(WRF_SRC_ROOT_DIR)/tools/gen_comms_warning  $(WRF_SRC_ROOT_DIR)/
tools/gen_comms.c ; \
          cat          $(WRF_SRC_ROOT_DIR)/external/RSL_LITE/gen_comms.c          >>
$(WRF_SRC_ROOT_DIR)/tools/gen_comms.c ; fi )

module_dm_rsllite :
       ( if [ ! -e module_dm.F ] ; then /bin/cp module_dm_warning module_dm.F ; \
          cat $(WRF_SRC_ROOT_DIR)/external/RSL_LITE/module_dm.F >> module_dm.F ;
fi )

wrfio_nf :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_netcdf ; \
          make $(J) NETCDFPATH="$(NETCDFPATH)" RANLIB="$(RANLIB)" CPP="$(CPP)" \
          CC="$(SCC)" CFLAGS="$(CFLAGS)" \
          FC="$(SFC) $(PROMOTION) $(OMP) $(FCFLAGS)" TRADFLAG="$(TRADFLAG)"
AR="$(AR)" ARFLAGS="$(ARFLAGS)" )

wrfio_pnf :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_pnetcdf ; \
          make $(J) NETCDFPATH="$(PNETCDFPATH)" RANLIB="$(RANLIB)" CPP="$(CPP)
$(ARCHFLAGS)" \
          FC="$(FC) $(PROMOTION) $(OMP) $(FCFLAGS)" TRADFLAG="$(TRADFLAG)"
AR="$(AR)" ARFLAGS="$(ARFLAGS)" )

wrfio_grib_share :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_grib_share ; \
          make $(J) CC="$(SCC)" CFLAGS="$(CFLAGS)" RM="$(RM)" RANLIB="$(RANLIB)"
CPP="$(CPP)" \
          FC="$(SFC) $(PROMOTION) -I. $(FCDEBUG) $(FCBASEOPTS) $(FCSUFFIX)"
TRADFLAG="$(TRADFLAG)" AR="$(AR)" ARFLAGS="$(ARFLAGS)" archive)

wrfio_grib1 :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_grib1 ; \
          make $(J) CC="$(SCC)" CFLAGS="$(CFLAGS)" RM="$(RM)" RANLIB="$(RANLIB)"
CPP="$(CPP)" \
          FC="$(SFC) $(PROMOTION) -I. $(FCDEBUG) $(FCBASEOPTS) $(FCSUFFIX)"
TRADFLAG="$(TRADFLAG)" AR="$(AR)" ARFLAGS="$(ARFLAGS)" archive)

wrfio_grib2 :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_grib2 ; \
          make $(J) CC="$(SCC)" CFLAGS="$(CFLAGS) " RM="$(RM)" RANLIB="$(RANLIB)" \
          CPP="$(CPP)" \
          FC="$(SFC) $(PROMOTION) -I. $(FCDEBUG) $(FCBASEOPTS) $(FCSUFFIX)"
TRADFLAG="-traditional" AR="$(AR)" ARFLAGS="$(ARFLAGS)" \
          FIXED="$(FORMAT_FIXED)" archive)

wrfio_int :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_int ; \
          make $(J) CC="$(CC)" CFLAGS_LOCAL="$(CFLAGS_LOCAL)" RM="$(RM)"
RANLIB="$(RANLIB)" CPP="$(CPP)" \
          FC="$(FC) $(PROMOTION) $(FCDEBUG) $(FCBASEOPTS) $(OMP)" FGREP="$(FGREP)" \
          TRADFLAG="$(TRADFLAG)" AR="$(AR)" ARFLAGS="$(ARFLAGS)"
ARCHFLAGS="$(ARCHFLAGS)" all )

esmf_time :
       ( cd $(WRF_SRC_ROOT_DIR)/external/esmf_time_f90 ; \
          make $(J) FC="$(SFC) $(PROMOTION) $(FCDEBUG) $(FCBASEOPTS)"
RANLIB="$(RANLIB)" \
          CPP="$(CPP) -I$(WRF_SRC_ROOT_DIR)/inc -I. $(ARCHFLAGS) $(TRADFLAG)"
AR="$(AR)" ARFLAGS="$(ARFLAGS)" )

fftpack :
       ( cd $(WRF_SRC_ROOT_DIR)/external/fftpack/fftpack5 ; \
          make $(J) FC="$(SFC)" FFLAGS="$(PROMOTION) $(FCDEBUG) $(FCBASEOPTS)"
RANLIB="$(RANLIB)" AR="$(AR)" \
          ARFLAGS="$(ARFLAGS)" CPP="$(CPP)" CPPFLAGS="$(CPPFLAGS)" RM="$(RM)" )
```

```
atm_ocn :
       ( cd $(WRF_SRC_ROOT_DIR)/external/atm_ocn ; \
          make $(J) CC="$(SCC)" CFLAGS="$(CFLAGS) " RM="$(RM)" RANLIB="$(RANLIB)" \
          CPP="$(CPP)" CPPFLAGS="$(CPPFLAGS)" \
          FC="$(DM_FC)  $(PROMOTION)  -I.  $(FCDEBUG)  $(FCBASEOPTS)  $(FCSUFFIX)"
TRADFLAG="-traditional" AR="$(AR)" ARFLAGS="$(ARFLAGS)" \
          FIXED="$(FORMAT_FIXED)" )

$(WRF_SRC_ROOT_DIR)/external/RSL_LITE/librsl_lite.a :
       ( cd $(WRF_SRC_ROOT_DIR)/external/RSL_LITE ; make $(J) CC="$(CC) $(CFLAGS)" \
          FC="$(FC) $(FCFLAGS) $(OMP) $(PROMOTION) $(BYTESWAPIO)" \
          CPP="$(CPP)   -I.   $(ARCHFLAGS)   $(OMPCPP)   $(TRADFLAG)"   AR="$(AR)"
ARFLAGS="$(ARFLAGS)" ;\
          $(RANLIB) $(WRF_SRC_ROOT_DIR)/external/RSL_LITE/librsl_lite.a )

######################
#      Macros, these should be generic for all machines

LN      =       ln -sf
MAKE    =       make -i -r
RM      =       rm -f


# These sub-directory builds are identical across all architectures

wrf_ioapi_includes :
       ( cd $(WRF_SRC_ROOT_DIR)/external/ioapi_share ; \
          $(MAKE)   NATIVE_RWORDSIZE="$(NATIVE_RWORDSIZE)"   RWORDSIZE="$(RWORDSIZE)"
AR="$(AR)" ARFLAGS="$(ARFLAGS)" )

wrfio_esmf :
       ( cd $(WRF_SRC_ROOT_DIR)/external/io_esmf ; \
          make FC="$(FC) $(PROMOTION) $(FCDEBUG) $(FCBASEOPTS) $(ESMF_MOD_INC)" \
          RANLIB="$(RANLIB)"     CPP="$(CPP)     $(POUND_DEF)     "     AR="$(AR)"
ARFLAGS="$(ARFLAGS)" )

#      There is probably no reason to modify these rules

.F.i:
       $(RM) $@
       sed -e "s/^\!.*'.*//" -e "s/^ *\!.*'.*//" $*.F > $*.G
       $(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $*.G > $*.i
       mv $*.i $(DEVTOP)/pick/$*.f90
       cp $*.F $(DEVTOP)/pick

.F.o:
       $(RM) $@
       sed -e "s/^\!.*'.*//" -e "s/^ *\!.*'.*//" $*.F > $*.G
       $(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $(OMPCPP) $*.G  > $*.bb
       $(SED_FTN) $*.bb | $(CPP) $(TRADFLAG) > $*.f90
       $(RM) $*.G $*.bb
       @ if echo $(ARCHFLAGS) | $(FGREP) 'DVAR4D'; then \
          echo COMPILING $*.F for 4DVAR ; \
          $(WRF_SRC_ROOT_DIR)/var/build/da_name_space.pl $*.f90 > $*.f90.tmp ; \
          mv $*.f90.tmp $*.f90 ; \
        fi
       $(FC)  -o  $@  -c  $(FCFLAGS)  $(OMP)  $(MODULE_DIRS)  $(PROMOTION)  $(FCSUFFIX)
$*.f90
```

```
.F.f90:
	$(RM) $@
	sed -e "s/^\!.*'.*//" -e "s/^ *\!.*'.*//" $*.F > $*.G
	$(SED_FTN) $*.G > $*.H
	$(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $*.H  > $@
	$(RM) $*.G $*.H

.f90.o:
	$(RM) $@
	$(FC) -o $@ -c $(FCFLAGS) $(PROMOTION) $(FCSUFFIX) $*.f90

setfeenv.o : setfeenv.c
	$(RM) $@
	$(CCOMP) -o $@ -c $(CFLAGS) $(OMPCC) $*.c

.c.o:
	$(RM) $@
	$(CC) -o $@ -c $(CFLAGS) $*.c

# A little more adventurous.  Allow full opt on
# mediation_integrate.o \
# shift_domain_em.o \
# solve_em.o  <-- gets a little kick from SOLVE_EM_SPECIAL too, if defined
# mediation_feedback_domain.o : mediation_feedback_domain.F
# mediation_force_domain.o : mediation_force_domain.F
# mediation_interp_domain.o : mediation_interp_domain.F

# compile these without high optimization to speed compile
track_driver.o : track_driver.F
convert_nmm.o : convert_nmm.F
init_modules_em.o : init_modules_em.F
input_wrf.o : input_wrf.F
module_io.o : module_io.F
module_comm_dm.o : module_comm_dm.F
module_comm_dm_0.o : module_comm_dm_0.F
module_comm_dm_1.o : module_comm_dm_1.F
module_comm_dm_2.o : module_comm_dm_2.F
module_comm_dm_3.o : module_comm_dm_3.F
module_comm_nesting_dm.o : module_comm_nesting_dm.F
module_configure.o : module_configure.F
module_domain.o : module_domain.F
module_domain_type.o : module_domain_type.F
module_alloc_space_0.o : module_alloc_space_0.F
module_alloc_space_1.o : module_alloc_space_1.F
module_alloc_space_2.o : module_alloc_space_2.F
module_alloc_space_3.o : module_alloc_space_3.F
module_alloc_space_4.o : module_alloc_space_4.F
module_alloc_space_5.o : module_alloc_space_5.F
module_alloc_space_6.o : module_alloc_space_6.F
module_alloc_space_7.o : module_alloc_space_7.F
module_alloc_space_8.o : module_alloc_space_8.F
module_alloc_space_9.o : module_alloc_space_9.F
module_tiles.o : module_tiles.F
module_initialize.o : module_initialize.F
module_physics_init.o : module_physics_init.F
module_initialize_squall2d_x.o : module_initialize_squall2d_x.F
module_initialize_squall2d_y.o : module_initialize_squall2d_y.F
```

```
module_initialize_scm_xy.o : module_initialize_scm_xy.F
module_integrate.o : module_integrate.F
module_io_mm5.o : module_io_mm5.F
module_io_wrf.o : module_io_wrf.F
module_si_io.o : module_si_io.F
module_wps_io_arw.o : module_wps_io_arw.F
module_state_description.o : module_state_description.F
output_wrf.o : output_wrf.F
solve_interface.o : solve_interface.F
start_domain.o : start_domain.F
wrf_bdyin.o : wrf_bdyin.F
wrf_bdyout.o : wrf_bdyout.F
wrf_ext_read_field.o : wrf_ext_read_field.F
wrf_ext_write_field.o : wrf_ext_write_field.F
wrf_fddaobs_in.o : wrf_fddaobs_in.F
wrf_histin.o : wrf_histin.F
wrf_histout.o : wrf_histout.F
wrf_inputin.o : wrf_inputin.F
wrf_inputout.o : wrf_inputout.F
wrf_restartin.o : wrf_restartin.F
wrf_restartout.o : wrf_restartout.F
wrf_tsin.o : wrf_tsin.F
nl_get_0_routines.o : nl_get_0_routines.F
nl_get_1_routines.o : nl_get_1_routines.F
nl_set_0_routines.o : nl_set_0_routines.F
nl_set_1_routines.o : nl_set_1_routines.F

track_driver.o \
convert_nmm.o \
init_modules_em.o \
module_initialize.o \
module_initialize_squall2d_x.o \
module_initialize_squall2d_y.o \
module_initialize_scm_xy.o \
module_integrate.o \
module_io_mm5.o \
module_io_wrf.o \
module_si_io.o \
module_wps_io_arw.o \
module_tiles.o \
output_wrf.o \
solve_interface.o \
start_domain.o \
wrf_fddaobs_in.o \
wrf_tsin.o :
        $(RM) $@
        $(SED_FTN) $*.F > $*.b
        $(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $(OMPCPP) $*.b  > $*.f90
        $(RM) $*.b
        @ if echo $(ARCHFLAGS) | $(FGREP) 'DVAR4D'; then \
           echo COMPILING $*.F for 4DVAR ; \
           $(WRF_SRC_ROOT_DIR)/var/build/da_name_space.pl $*.f90 > $*.f90.tmp ; \
           mv $*.f90.tmp $*.f90 ; \
         fi
        if $(FGREP) '!$$OMP' $*.f90 ; then \
           if [ -n "$(OMP)" ] ; then echo COMPILING $*.F WITH OMP ; fi ; \
          $(FC) -c $(PROMOTION) $(FCNOOPT) $(FCBASEOPTS) $(MODULE_DIRS) $(FCSUFFIX)
$(OMP) $*.f90 ; \
         else \
           if [ -n "$(OMP)" ] ; then echo COMPILING $*.F WITHOUT OMP ; fi ; \
          $(FC) -c $(PROMOTION) $(FCNOOPT) $(FCBASEOPTS) $(MODULE_DIRS) $(FCSUFFIX)
 $*.f90 ; \
         fi

 #solve_em.o :
 #      $(RM) $@
 #      $(SED_FTN) $*.F > $*.b
 #      $(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $*.b  > $*.f90
 #      $(RM) $*.b
 #      $(FC)  -o  $@  -c  $(FCFLAGS)  $(MODULE_DIRS)  $(PROMOTION)  $(FCSUFFIX)
 $(SOLVE_EM_SPECIAL) $(OMP) $*.f90

 module_sf_ruclsm.o : module_sf_ruclsm.F

 module_sf_ruclsm.o :
        $(RM) $@
        $(SED_FTN) $*.F > $*.b
        $(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $(OMPCPP) $*.b  > $*.f90
        $(RM) $*.b
        if $(FGREP) '!$$OMP' $*.f90 ; then \
           echo COMPILING $*.F WITH OMP ; \
           if [ -n "$(OMP)" ] ; then echo COMPILING $*.F WITH OMP ; fi ; \
          $(FC)  -c  $(PROMOTION)  $(FCREDUCEDOPT)  $(FCBASEOPTS)  $(MODULE_DIRS)
 $(FCSUFFIX) $(OMP) $*.f90 ; \
         else \
           if [ -n "$(OMP)" ] ; then echo COMPILING $*.F WITHOUT OMP ; fi ; \
          $(FC)  -c  $(PROMOTION)  $(FCREDUCEDOPT)  $(FCBASEOPTS)  $(MODULE_DIRS)
 $(FCSUFFIX) $*.f90 ; \
         fi

 # compile without OMP
 input_wrf.o \
 module_domain.o \
 module_domain_type.o \
 module_physics_init.o \
 module_io.o \
 wrf_bdyin.o \
 wrf_bdyout.o \
 wrf_ext_read_field.o \
 wrf_ext_write_field.o \
 wrf_histin.o \
 wrf_histout.o \
 wrf_inputin.o \
 wrf_inputout.o \
 wrf_restartin.o \
 wrf_restartout.o \
 module_state_description.o \
 module_alloc_space.o \
 module_alloc_space_0.o \
 module_alloc_space_1.o \
 module_alloc_space_2.o \
 module_alloc_space_3.o \
 module_alloc_space_4.o \
 module_alloc_space_5.o \
 module_alloc_space_6.o \
 module_alloc_space_7.o \
 module_alloc_space_8.o \
 module_alloc_space_9.o \
 module_comm_dm.o \
 module_comm_dm_0.o \
 module_comm_dm_1.o \
 module_comm_dm_2.o \
 module_comm_dm_3.o \
 module_comm_nesting_dm.o \
 module_configure.o :
        $(RM) $@
        $(CPP) -I$(WRF_SRC_ROOT_DIR)/inc $(CPPFLAGS) $(OMPCPP) $*.F  > $*.bb
        $(SED_FTN) $*.bb | $(CPP) $(TRADFLAG) > $*.f90
        @ if echo $(ARCHFLAGS) | $(FGREP) 'DVAR4D'; then \
           echo COMPILING $*.F for 4DVAR ; \
           $(WRF_SRC_ROOT_DIR)/var/build/da_name_space.pl $*.f90 > $*.f90.tmp ; \
           mv $*.f90.tmp $*.f90 ; \
         fi
        $(RM) $*.b $*.bb
        $(FC)  -c  $(PROMOTION)  $(FCSUFFIX)  $(FCNOOPT)  $(FCBASEOPTS)  $(MODULE_DIRS)
 $*.f90
```

## WRF - Create CPLMASK

Use something similar to this program named make_CPLMASK.m

```console
nc=netcdf('wrfinput_d01','w');
lm=nc{'LANDMASK'}(:,:,:);
lk=nc{'LAKEMASK'}(:,:,:);
cp=nc{'CPLMASK'}(:,:,:,:);
xlat=nc{'XLAT'}(:,:,:);
xlon=nc{'XLONG'}(:,:,:);
indx0=find(lm==0);
indxl=find(lk==1);
indxN=find(xlat>-25.9869);
indxS=find(xlat<-38.0000);
indxE=find(xlon>22.0000);
indxW=find(xlon<8.0000);
cp(indx0)=1;
cp(indxl)=0;
cp(indxN)=0;
cp(indxS)=0;
cp(indxE)=0;
cp(indxW)=0;
nc{'CPLMASK'}(:,1,:,:)=cp;
close(nc)
```

## WRF - namelist.input

To add variable SST update

in &time_control

```console
io_form_auxinput4         = 2,
auxinput4_interval        = 60,
auxinput4_inname          = "wrflowinp_d<domain>",
```

in &domains

```console
num_ext_model_couple_dom = 1,
```

in &physics

```console
sst_update               = 1,
```

get SST files in grib format using

```console
#!bin/csh
#
# Get SST for a certain date
#
# 28/05/2015 Andres Sepulveda - University of Concepcion (andres@dgeo.udec.cl)
#
set f=20000124
set URL = "ftp://polar.ncep.noaa.gov/pub/history/sst/"
set WGET=/usr/bin/wget

       $WGET -t 0 ${URL}/rtg_sst_grb_0.5.${f}
```

# CROCO

Define MPI and OA_COUPLING in cppdefs.h

## CROCO - cppdefs.h

```console
                      /* Parallelization */
# undef  OPENMP
# define  MPI
                      /* I/O server */
# undef  XIOS
                      /* Non-hydrostatic option */
# undef  NBQ
                      /* Nesting */
# undef  AGRIF
# undef  AGRIF_2WAY
                      /* OA and OW Coupling via OASIS (MPI) */
# define  OA_COUPLING
# undef  OW_COUPLING
```

## CROCO - jobcomp

```console
#NETCDFLIB="-L/home/mosa/libraries/netcdf/lib -lnetcdf"
#NETCDFINC="-I/home/mosa/libraries/netcdf/include"
NETCDFLIB=$(nf-config --flibs)
NETCDFINC=-I$(nf-config --includedir)       #These are the libraries used to compile OASIS
```

```
#
# set MPI directories if needed
#
MPIF90="/usr/bin/mpif90"
MPILIB=""
MPIINC=""

#
# set OASIS-MCT (or OASIS3) directories if needed
#
PRISM_ROOT_DIR=/home/mosa/compile_oa3-mct
```

# WW3

# Errata Compendium

1)
Check for error messages in the following files:
rsl.error.0000 (WRF)
debug.root.01 (OASIS)
nout.000000   (OASIS)

2)

From Rachid BEnshila
"I started back from your Benguela configuration for WRF,
and the standard one for CROCO (not the same grid one).
For what I saw, the pb we had was when oasis was trying
to build its interpolation weights, the code stopped there.
I changed to another method, just to check, and it's running.
I changed in the namcouple for oasis:
DISTWGT LR SCALAR LATLON 1 4
to
BILINEAR LR SCALAR LATLON 1 4
"

3)
ERROR in MPI_Setup: number of MPI-nodes should be   1 instead of 2

Verify the number of CPUs declared for CROCO in param.h
Notice that is possible to declare just one CPU and use MPI paralelization.