**Supplementary Information**

**Table SI1. Profile setting (~/.kshrc) used for compiling UKESM-TOPAZ version 1.0.**

|  |
| --- |
| module load python/2.7.3 subversion/1.8.11  #  ###### UKESM u-bf935 #####  module load rose/2018.02.0  module load cylc/7.8.0  module load fcm/2017.10.0  module load grib\_api/1.13.0  ###### for fcm\_make #####  module load glib  module load cray-netcdf-hdf5parallel/4.3.2  module load jasper/1.900.1  module load libpng/1.2.51  export FCM\_HOME=/opt/kma/fcm/2017.10.0  export PATH=$FCM\_HOME/bin:$PATH  export TMPDIR=~/tmpdir/  mkdir -p $TMPDIR  export UMDIR\_PY=~/UKESM1/ANCL/u-bq616\_ukesm-topaz/mule.vn1.4  export PATH=$PATH:$UMDIR\_PY/binexport PYTHONPATH=$PYTHONPATH:$UMDIR\_PY/lib:~/UKESM1/ANCL/u-bq616\_ukesm-topaz/pythonlib |

**Table SI2. Module list used in compiling the UKESM-TOPAZ.**

|  |
| --- |
| 1) modules/3.2.10.4  2) switch/1.0-1.0502.60522.1.61.ari  3) craype-network-aries  4) cce/8.3.7  5) craype/2.2.1  6) totalview-support/1.2.0.3  7) totalview/8.14.1  8) cray-libsci/13.0.1  9) udreg/2.3.2-1.0502.10518.2.17.ari  10) ugni/6.0-1.0502.10863.8.29.ari  11) pmi/5.0.6-1.0000.10439.140.2.ari  12) dmapp/7.0.1-1.0502.11080.8.76.ari  13) gni-headers/4.0-1.0502.10859.7.8.ari  14) xpmem/0.1-2.0502.64982.5.3.ari  15) dvs/2.5\_0.9.0-1.0502.2188.1.116.ari  16) rca/1.0.0-2.0502.60530.1.62.ari  17) atp/1.7.5  18) cray-mpich/7.1.1  19) cray-snplauncher/7.1.1  20) PrgEnv-cray/5.2.82  21) pbs/18.2.3.20181206140456  22) craype-haswell  23) bison/2.4.2.0  24) gd/2.0.35.0  25) graphviz/2.28.0  26) python/2.7.3  27) subversion/1.8.11  28) rose/2018.02.0  29) cylc/7.8.0  30) fcm/2017.10.0  31) grib\_api/1.13.0  32) libffi/3.0.11  33) glib/2.34.3  34) cray-netcdf-hdf5parallel/4.3.2  35) jasper/1.900.1  36) libpng/1.2.51 |

**Table SI3. fc\_nemo for TOPAZ compilation.**

|  |
| --- |
| FC = ftn  CC = cc  NF\_INC = -I/opt/cray/netcdf-hdf5parallel/4.3.2/CRAY/83/include  NF\_LIB = -L/opt/cray/netcdf-hdf5parallel/4.3.2/CRAY/83/lib -lnetcdff -lnetcdf |

**Table SI4. mkmf.template for TOPAZ compilation.**

|  |
| --- |
| CPPDEFS = -DUSE\_OCEAN\_BGC -DENABLE\_ODA -DSPMD -DLAND\_BND\_TRACERS -Duse\_netCDF -DINTERNAL\_FILE\_NML  FC\_TOPAZ = $(FC)  CC\_TOPAZ = $(CC)  TOPAZ\_CPPFLAGS = -I./include $(NF\_INC)  LDFLAGS = $(NF\_LIB)  TOPAZ\_FFLAGS = -em -s integer32 -s real64 -O0 -e0 -eZ |

**Table SI5. Representation of the modified contents of OPA\_SRC/oce.F90 among NEMO source codes of UKESM1 u-bf935 suite for UKESM-TOPAZ. The Line number is expressed by adding L, and the actual modified content is displayed**

|  |  |
| --- | --- |
| Line | Modification |
| L90 | #if defined key\_topaz  LOGICAL, PUBLIC, PARAMETER :: ln\_topaz=.TRUE. ! Medusa switched on or off.  #else  LOGICAL, PUBLIC, PARAMETER :: ln\_topaz=.FALSE. ! Medusa switched on or off.  #endif |
| L147 | IF (ln\_topaz) THEN  ! We only actually need these arrays to be allocated if coupling and TOPAZ  ! are enabled  ALLOCATE( CO2Flux\_out\_cpl(jpi,jpj), DMS\_out\_cpl(jpi,jpj), &  chloro\_out\_cpl(jpi,jpj), &  PCO2a\_in\_cpl(jpi,jpj), Dust\_in\_cpl(jpi,jpj), STAT=ierr(5) )  ENDIF  #endif |

**Table SI6. As Table SI5 but for OPA\_SBC/SBC/sbccpl.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L38 | ln\_medusa, ln\_topaz |
| L153 | INTEGER, PARAMETER :: jpsnd = 33 |
| L571 | IF (TRIM(sn\_rcv\_atm\_pco2%cldes) == 'topaz') THEN  srcv(jpr\_atm\_pco2)%laction = .TRUE.  END IF  srcv(jpr\_atm\_dust)%clname = 'OATMDUST'  IF (TRIM(sn\_rcv\_atm\_dust%cldes) == 'topaz') THEN  srcv(jpr\_atm\_dust)%laction = .TRUE.  END IF |
| L848 | IF (ln\_topaz) THEN  IF( srcv(jpr\_atm\_pco2)%laction) PCO2a\_in\_cpl(:,:) = frcv(jpr\_atm\_pco2)%z3(:,:,1)  IF( srcv(jpr\_atm\_dust)%laction) Dust\_in\_cpl(:,:) = frcv(jpr\_atm\_dust)%z3(:,:,1)  ENDIF |
| L1279 | ! ! ------------------------- !  ! ! MEDUSA output fields !  ! ! ------------------------- !  ! Surface dimethyl sulphide from Medusa  ssnd(jps\_bio\_dms)%clname = 'OBioDMS'  IF( TRIM(sn\_snd\_bio\_dms%cldes) == 'medusa' ) ssnd(jps\_bio\_dms )%laction = .TRUE.  ! Surface dimethyl sulphide from Topaz  IF( TRIM(sn\_snd\_bio\_dms%cldes) == 'topaz' ) ssnd(jps\_bio\_dms )%laction = .FALSE. !.TRUE.  ! Surface CO2 flux from Medusa  ssnd(jps\_bio\_co2)%clname = 'OBioCO2'  IF( TRIM(sn\_snd\_bio\_co2%cldes) == 'medusa' ) ssnd(jps\_bio\_co2 )%laction = .TRUE.  ! Surface CO2 flux from Topaz  IF( TRIM(sn\_snd\_bio\_co2%cldes) == 'topaz' ) ssnd(jps\_bio\_co2 )%laction = .FALSE. !.TRUE.  ! Surface chlorophyll from Medusa  ssnd(jps\_bio\_chloro)%clname = 'OBioChlo'  IF( TRIM(sn\_snd\_bio\_chloro%cldes) == 'medusa' ) ssnd(jps\_bio\_chloro )%laction = .TRUE.  ! Surface chlorophyll from Topaz  IF( TRIM(sn\_snd\_bio\_chloro%cldes) == 'topaz' ) ssnd(jps\_bio\_chloro )%laction = .FALSE. !.TRUE. |
| L2473 | IF (ln\_topaz) THEN  ! ! ! ---------------------------------------------- !  ! ! ! CO2 flux, DMS and chlorophyll from MEDUSA !  ! ! ! ---------------------------------------------- !  IF ( ssnd(jps\_bio\_co2)%laction ) THEN  CALL cpl\_snd( jps\_bio\_co2, isec, RESHAPE( CO2Flux\_out\_cpl, (/jpi,jpj,1/) ), info )  ENDIF  IF ( ssnd(jps\_bio\_dms)%laction ) THEN  CALL cpl\_snd( jps\_bio\_dms, isec, RESHAPE( DMS\_out\_cpl, (/jpi,jpj,1/) ), info )  ENDIF  IF ( ssnd(jps\_bio\_chloro)%laction ) THEN  CALL cpl\_snd( jps\_bio\_chloro, isec, RESHAPE( chloro\_out\_cpl, (/jpi,jpj,1/) ), info )  ENDIF  ENDIF |

**Table SI7. As Table SI5 but for OPA\_SBC/ZDF/zdfkpp.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L1275 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L1282 | INTEGER :: temp\_jptra |
| L1283 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L1292 | DO jn = 1, temp\_jptra !! jptra 🡪 temp\_jptra |

**Table SI8. As Table SI5 but for TOP\_SRC/trcini.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L41 | #if defined key\_topaz  USE generic\_TOPAZ\_model, ONLY: generic\_TOPAZ\_init  USE ocean\_var\_4topaz, ONLY: tp\_mbathy, tp\_tmask, tp\_e1t, tp\_e2t  USE iom ! I/O manager  USE fldread ! time interpolation  #endif |
| L80 | #if defined key\_topaz  INTEGER :: ios, numnam\_cfg, numnam\_ref, ierr  INTEGER :: numpar, numapr  INTEGER :: stf\_cnt  ! REAL(wp), DIMENSION(365) :: zsteps !: maximum number of times record in a file  ! REAL(wp), DIMENSION(12) :: zsteps\_mon  CHARACTER(LEN=100) :: cn\_dir, cn\_dir2, cn\_dir3  TYPE(FLD\_N) :: sn\_par  TYPE(FLD\_N), DIMENSION(13) :: sn\_stf  TYPE(FLD\_N) :: sn\_apr  ! TYPE(FLD), ALLOCATABLE, DIMENSION(:) :: sf\_par  ! LOGICAL :: ln\_varpar  ! REAL(wp) :: parlux  NAMELIST/namopt/cn\_dir, sn\_par, ln\_varpar, parlux  NAMELIST/namstf/cn\_dir2, sn\_stf, sn\_cnt  NAMELIST/namapr/cn\_dir3, sn\_apr, ln\_presatm, aprcnst  #endif |
| L125 | #if defined key\_topaz  IF (.NOT. allocated(tp\_mbathy)) allocate(tp\_mbathy(jpi,jpj)); tp\_mbathy=0  IF (.NOT. allocated(tp\_tmask)) allocate(tp\_tmask(jpi,jpj,jpk)); tp\_tmask=0.0  IF (.NOT. allocated(tp\_e1t)) allocate(tp\_e1t(jpi,jpj)); tp\_e1t=0.0  IF (.NOT. allocated(tp\_e2t)) allocate(tp\_e2t(jpi,jpj)); tp\_e2t=0.0  DO ji = 1,jpi; DO jj = 1,jpj; DO jk = 1,jpk  tp\_mbathy(ji,jj)=mbathy(ji,jj)  tp\_tmask(ji,jj,jk)=tmask(ji,jj,jk)  tp\_e1t(ji,jj)=e1t(ji,jj)  tp\_e2t(ji,jj)=e2t(ji,jj)  END DO; END DO; END DO  CALL ctl\_opn( numnam\_cfg, 'namelist\_topaz\_cfg', 'OLD', 'FORMATTED', 'SEQUENTIAL', -1, numout, .FALSE.)  CALL ctl\_opn( numnam\_ref, 'namelist\_topaz\_ref', 'OLD', 'FORMATTED', 'SEQUENTIAL', -1, numout, .FALSE.)  !!SWFRAC  REWIND(numnam\_ref)  READ(numnam\_ref,namopt, IOSTAT=ios)  IF (ios /= 0 ) THEN  IF(lwp) WRITE(numout,\*) 'topaz namelist-namopt error'  END IF  IF ( ln\_varpar ) THEN  ALLOCATE( sf\_par(1), STAT=ierr )  CALL fld\_fill( sf\_par, (/ sn\_par /), cn\_dir, 'topaz\_init', 'Variable PAR fraction', 'namopt' )  ALLOCATE( sf\_par(1)%fnow(jpi,jpj,1) )  IF( sn\_par%ln\_tint ) ALLOCATE( sf\_par(1)%fdta(jpi,jpj,1,2) )  ! CALL iom\_open ( TRIM( sn\_par%clname ) , numpar )  ! CALL iom\_gettime( numpar, zsteps, kntime=ntimes\_par) ! get number of record in file  END IF  !!Tracer Surface Flux  REWIND(numnam\_ref)  READ(numnam\_ref, namstf, IOSTAT=ios)  IF (ios /= 0 ) THEN  IF(lwp) WRITE(numout,\*) 'topaz namelist-namstf error'  END IF  ALLOCATE( sf\_stf(sn\_cnt), STAT=ierr )  DO stf\_cnt = 1, sn\_cnt  ALLOCATE( sf\_stf(stf\_cnt)%fnow(jpi,jpj,1) )  IF( sn\_stf(stf\_cnt)%ln\_tint ) ALLOCATE( sf\_stf(stf\_cnt)%fdta(jpi,jpj,1,2) )  END DO  CALL fld\_fill ( sf\_stf, (/ sn\_stf /), cn\_dir2, 'topaz\_init', 'Tracer surface flux' , 'namstf')  !!Atmospheric pressure for air-sea o2/co2 flux  REWIND(numnam\_ref)  READ(numnam\_ref, namapr, IOSTAT=ios)  IF (ios /= 0 ) THEN  IF(lwp) WRITE(numout,\*) 'topaz namelist-namapr error'  END IF  IF ( ln\_presatm ) THEN  ALLOCATE( sf\_apr(1), STAT=ierr )  CALL fld\_fill( sf\_apr, (/ sn\_apr /), cn\_dir3, 'topaz\_init', 'Atmospheric pressure', 'namapr' )  ALLOCATE( sf\_apr(1)%fnow(jpi,jpj,1) )  IF ( sn\_apr%ln\_tint ) ALLOCATE( sf\_apr(1)%fdta(jpi,jpj,1,2))  ! CALL iom\_open ( TRIM( sn\_apr%clname ) , numapr )  ! CALL iom\_gettime( numapr, zsteps, kntime=ntimes\_apr) ! get number of record in file  END IF |

call generic\_TOPAZ\_init(jpi, jpj, jpk, rdttrc(1)) !TOPAZ bio-model init

|  |
| --- |
| #endif |

**Table SI9. As Table A5 but for TOP\_SRC/trcsms.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L25 | USE par\_oce , ONLY : jpi, jpj, jpk  #if defined key\_topaz  USE generic\_TOPAZ\_model, ONLY: generic\_TOPAZ\_column\_physics, topaz\_optic  USE ocean\_var\_4topaz  USE fldread  #endif |
| L30 | # include "top\_substitute.h90" |
| L49 | CHARACTER (len=15) :: ktt  REAL :: dt\_topaz  INTEGER :: i, j, k, jn, stf\_cnt |
| L60 | #if defined key\_topaz  do i = 1, jpi  do j = 1, jpj  do k = 1, jpk  ! tp\_temp(i,j,k)=tsn(i,j,k,jp\_tem)  ! tp\_salt(i,j,k)=tsn(i,j,k,jp\_sal)  tp\_fse3w(i,j,k)=fse3w(i,j,k)  tp\_fsdept(i,j,k)=fsdept(i,j,k)  tp\_fsdepw(i,j,k)=fsdepw(i,j,k)  tp\_fse3t(i,j,k)=fse3t(i,j,k)  tp\_rhd(i,j,k)=rhd(i,j,k)  tp\_avt(i,j,k)=avt(i,j,k)  if ( kt .eq. 1 ) then  tp\_temp(i,j,k)=tsn(i,j,k,jp\_tem) \* tmask(i,j,k)  tp\_salt(i,j,k)=tsn(i,j,k,jp\_sal) \* tmask(i,j,k)  do jn = 1, jptra  tp\_trc(i,j,k,jn)=trn(i,j,k,jn) \* tmask(i,j,k)  end do  dt\_topaz=rdttrc(1)  else if ( kt .ne. 1 .and. .not. ln\_top\_euler ) then  tp\_temp(i,j,k)=tsb(i,j,k,jp\_tem) \* tmask(i,j,k)  tp\_salt(i,j,k)=tsb(i,j,k,jp\_sal) \* tmask(i,j,k)  do jn = 1, jptra  tp\_trc(i,j,k,jn)=trb(i,j,k,jn) \* tmask(i,j,k)  ! tp\_trc(i,j,k,jn)=trn(i,j,k,jn) !only trp -> sms  end do  dt\_topaz=rdttrc(1)\*2.  else if ( kt .ne. 1 .and. ln\_top\_euler ) then  tp\_temp(i,j,k)=tsn(i,j,k,jp\_tem) \* tmask(i,j,k)  tp\_salt(i,j,k)=tsn(i,j,k,jp\_sal) \* tmask(i,j,k)  do jn = 1, jptra  tp\_trc(i,j,k,jn)=trn(i,j,k,jn) \* tmask(i,j,k)  end do  dt\_topaz=rdttrc(1)  end if  end do !do k = 1, jpk  tp\_hmlpt(i,j)=hmlpt(i,j)  tp\_qsr(i,j)=qsr(i,j)  tp\_wndm(i,j)=wndm(i,j)  tp\_fr\_i(i,j)=fr\_i(i,j)  end do !do j = 1, jpj  end do !do i = 1, jpi  !!APR  IF ( ln\_presatm ) THEN  ! IF ( kt == nit000 .OR. ( kt /= nit000 .AND. ntimes\_apr > 1 ) ) THEN  CALL fld\_read(kt, 1, sf\_apr)  tp\_apr(:,:) = sf\_apr(1)%fnow(:,:,1)  ! END IF  ELSE  tp\_apr(:,:) = aprcnst  ENDIF  !!SWFRAC  IF ( ln\_varpar ) THEN  ! IF( kt == nit000 .OR. ( kt /= nit000 .AND. ntimes\_par > 1 ) ) THEN  CALL fld\_read(kt, 1, sf\_par)  tp\_swfrac(:,:) = sf\_par(1)%fnow(:,:,1)  ! ENDIF  ELSE  tp\_swfrac(:,:)=parlux  ENDIF  call topaz\_optic  IF ( ln\_qsr\_bio ) THEN  do i = 1, jpi; do j = 1, jpj; do k = 1, jpk  etot3(i,j,k)=tp\_etot3(i,j,k)\*tmask(i,j,k)  end do; end do; end do  END IF  CALL fld\_read(kt, 1, sf\_stf)  do i = 1, jpi  do j = 1, jpj  tp\_flux\_no3\_wet(i,j)=sf\_stf(1)%fnow(i,j,1)  tp\_flux\_no3\_dry(i,j)=sf\_stf(2)%fnow(i,j,1)  tp\_flux\_nh4\_wet(i,j)=sf\_stf(3)%fnow(i,j,1)  tp\_flux\_nh4\_dry(i,j)=sf\_stf(4)%fnow(i,j,1)  tp\_flux\_lith(i,j)=sf\_stf(5)%fnow(i,j,1)  tp\_flux\_fed(i,j)=sf\_stf(6)%fnow(i,j,1)  IF ( kt == nit000 ) THEN  tp\_o2\_flux\_alpha(i,j)=sf\_stf(7)%fnow(i,j,1)  tp\_o2\_flux\_sc\_no(i,j)=sf\_stf(8)%fnow(i,j,1)  tp\_o2\_flux\_csurf(i,j)=sf\_stf(9)%fnow(i,j,1)  tp\_co2\_flux\_alpha(i,j)=sf\_stf(10)%fnow(i,j,1)  tp\_co2\_flux\_sc\_no(i,j)=sf\_stf(11)%fnow(i,j,1)  tp\_co2\_flux\_csurf(i,j)=sf\_stf(12)%fnow(i,j,1)  ENDIF  end do  end do  call generic\_TOPAZ\_column\_physics(kt, dt\_topaz)  do i = 1, jpi  do j = 1, jpj  do k = 1, jpk  do jn = 1, jptra  if ( ln\_top\_euler ) then  trn(i,j,k,jn)=tp\_trc(i,j,k,jn)\*tmask(i,j,k)  trb(i,j,k,jn)=tp\_trc(i,j,k,jn)\*tmask(i,j,k)  else if ( .NOT. ln\_top\_euler ) then  trb(i,j,k,jn)=tp\_trc(i,j,k,jn)\*tmask(i,j,k) !only sms -> trp  ! trn(i,j,k,jn)=tp\_trc(i,j,k,jn)!\*tmask(i,j,k) !only trp -> sms  end if  end do  end do  end do  end do  #endif |

**Table SI10. As Table SI5 but for TOP\_SRC/par\_trc.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L21 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_ntrc, topaz\_diag\_ntrc, topaz\_prog\_ntrc  USE fldread, ONLY: FLD  #endif |
| L30 | #if defined key\_topaz  INTEGER, PUBLIC, PARAMETER :: jptra = jp\_pisces + jp\_cfc + jp\_c14b + jp\_my\_trc + topaz\_ntrc  #else  INTEGER, PUBLIC, PARAMETER :: jptra = jp\_pisces + jp\_cfc + jp\_c14b + jp\_age + jp\_my\_trc + jp\_idtra + jp\_medusa  #endif |
| L45 | #ifdef key\_topaz  TYPE(FLD), ALLOCATABLE, DIMENSION(:) :: sf\_stf ! structure of input stf  TYPE(FLD), ALLOCATABLE, DIMENSION(:) :: sf\_par ! structure of input par  TYPE(FLD), ALLOCATABLE, DIMENSION(:) :: sf\_apr ! structure of input apr  ! INTEGER :: ntimes\_par, ntimes\_apr ! number of time steps in a file  INTEGER :: sn\_cnt  LOGICAL :: ln\_varpar, ln\_presatm  REAL(wp) :: parlux, aprcnst  #endif |

**Table SI11. As Table SI5 but for TOP\_SRC/trcwri.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L27 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc, topaz\_diag\_ntrc  #endif |
| L60-69 | #ifdef key\_topaz  DO jn = 1, topaz\_prog\_ntrc + topaz\_diag\_ntrc  cltra = TRIM( ctrcnm(jn) ) ! short title for tracer  CALL iom\_put( cltra, trn(:,:,:,jn) )  END DO  #else  ! write the tracer concentrations in the file  ! ---------------------------------------  IF( lk\_pisces ) CALL trc\_wri\_pisces ! PISCES  IF( lk\_cfc ) CALL trc\_wri\_cfc ! surface fluxes of CFC  IF( lk\_c14b ) CALL trc\_wri\_c14b ! surface fluxes of C14  IF( lk\_age ) CALL trc\_wri\_age ! AGE tracer  IF( lk\_my\_trc ) CALL trc\_wri\_my\_trc ! MY\_TRC tracers  IF( lk\_idtra ) CALL trc\_wri\_idtra ! Idealize tracers  IF( lk\_medusa ) CALL trc\_wri\_medusa ! MESDUSA  #endif |

**Table SI12. As Table SI5 but for TOP\_SRC/trcnam.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L203 | #if defined key\_topaz  IF(lwp) WRITE(numout,\*) ' TOPAZ used'  #else  IF(lwp) WRITE(numout,\*) ' TOPAZ not used'  #endif |

**Table SI13. As Table SI5 but for TOP\_SRC/TRP/trdmxl\_trc\_rst.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L15 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L45,L139 | INTEGER :: temp\_jptra |
| L46,L140 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L71, L83,L161,L172 | DO jn = 1, temp\_jptra ! jptra🡪temp\_jptra |

**Table SI14. As Table SI5 but for TOP\_SRC/TRP/trcsbc.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L24 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L140, L203 | #ifdef key\_topaz  DO jn = 1, topaz\_prog\_ntrc  #else  DO jn = 1, jptra  #endif |

**Table SI15. As Table SI5 but for TOP\_SRC/TRP/trcadv.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L31 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L79 | INTEGER :: temp\_jptra |
| L80 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L116-L157 | !! Add conditional statements about “key\_topaz” and move the sentence of “allocate” and “deallocate” into conditional statement.  #ifdef key\_topaz  IF( l\_trdtrc ) THEN  CALL wrk\_alloc( jpi, jpj, jpk, temp\_jptra, ztrtrd )  ztrtrd(:,:,:,:) = tra(:,:,:,:)  ENDIF  !  SELECT CASE ( nadv ) !== compute advection trend and add it to general trend ==!  CASE ( 1 ) ; CALL tra\_adv\_cen2 ( kt, nittrc000, 'TRC', zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc ) ! 2nd order centered  CASE ( 2 ) ; CALL tra\_adv\_tvd ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc ) ! TVD  CASE ( 3 ) ; CALL tra\_adv\_muscl ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, tra, topaz\_prog\_ntrc, ln\_trcadv\_msc\_ups ) ! MUSCL  CASE ( 4 ) ; CALL tra\_adv\_muscl2( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc ) ! MUSCL2  CASE ( 5 ) ; CALL tra\_adv\_ubs ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc ) ! UBS  CASE ( 6 ) ; CALL tra\_adv\_qck ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc ) ! QUICKEST  !  CASE (-1 ) !== esopa: test all possibility with control print ==!  CALL tra\_adv\_cen2 ( kt, nittrc000, 'TRC', zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc )  WRITE(charout, FMT="('adv1')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_tvd ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc )  WRITE(charout, FMT="('adv2')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_muscl ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, tra, topaz\_prog\_ntrc, ln\_trcadv\_msc\_ups )  WRITE(charout, FMT="('adv3')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_muscl2( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc )  WRITE(charout, FMT="('adv4')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_ubs ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc )  WRITE(charout, FMT="('adv5')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_qck ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, topaz\_prog\_ntrc )  WRITE(charout, FMT="('adv6')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  !  END SELECT  IF( l\_trdtrc ) THEN ! save the advective trends for further diagnostics  DO jn = 1, temp\_jptra  ztrtrd(:,:,:,jn) = tra(:,:,:,jn) - ztrtrd(:,:,:,jn)  CALL trd\_tra( kt, 'TRC', jn, jptra\_totad, ztrtrd(:,:,:,jn) )  END DO  CALL wrk\_dealloc( jpi, jpj, jpk, temp\_jptra, ztrtrd )  ENDIF  #else  IF( l\_trdtrc ) THEN  CALL wrk\_alloc( jpi, jpj, jpk, jptra, ztrtrd )  ztrtrd(:,:,:,:) = tra(:,:,:,:)  ENDIF  SELECT CASE ( nadv ) !== compute advection trend and add it to general trend ==!  CASE ( 1 ) ; CALL tra\_adv\_cen2 ( kt, nittrc000, 'TRC', zun, zvn, zwn, trb, trn, tra, jptra ) ! 2nd order centered  CASE ( 2 ) ; CALL tra\_adv\_tvd ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra ) ! TVD  CASE ( 3 ) ; CALL tra\_adv\_muscl ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, tra, jptra, ln\_trcadv\_msc\_ups ) ! MUSCL  CASE ( 4 ) ; CALL tra\_adv\_muscl2( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra ) ! MUSCL2  CASE ( 5 ) ; CALL tra\_adv\_ubs ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra ) ! UBS  CASE ( 6 ) ; CALL tra\_adv\_qck ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra ) ! QUICKEST  !  CASE (-1 ) !== esopa: test all possibility with control print ==!  CALL tra\_adv\_cen2 ( kt, nittrc000, 'TRC', zun, zvn, zwn, trb, trn, tra, jptra )  WRITE(charout, FMT="('adv1')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_tvd ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra )  WRITE(charout, FMT="('adv2')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_muscl ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, tra, jptra, ln\_trcadv\_msc\_ups )  WRITE(charout, FMT="('adv3')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_muscl2( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra )  WRITE(charout, FMT="('adv4')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_ubs ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra )  WRITE(charout, FMT="('adv5')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  CALL tra\_adv\_qck ( kt, nittrc000, 'TRC', r2dt, zun, zvn, zwn, trb, trn, tra, jptra )  WRITE(charout, FMT="('adv6')") ; CALL prt\_ctl\_trc\_info(charout)  CALL prt\_ctl\_trc(tab4d=tra, mask=tmask, clinfo=ctrcnm,clinfo2='trd')  !  END SELECT  IF( l\_trdtrc ) THEN ! save the advective trends for further diagnostics  DO jn = 1, jptra  ztrtrd(:,:,:,jn) = tra(:,:,:,jn) - ztrtrd(:,:,:,jn)  CALL trd\_tra( kt, 'TRC', jn, jptra\_totad, ztrtrd(:,:,:,jn) )  END DO  CALL wrk\_dealloc( jpi, jpj, jpk, jptra, ztrtrd )  ENDIF  #endif |

**Table SI16. As Table SI5 but for TOP\_SRC/TRP/trcbbl.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L29 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L58 | INTEGER :: temp\_jptra |
| L61 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L68 | ALLOCATE(ztrtrd( 1:jpi, 1:jpj, 1:jpk, 1:temp\_jptra )) |
| L75 | CALL tra\_bbl\_dif( trb, tra, temp\_jptra ) |
| L86 | CALL tra\_bbl\_adv( trb, tra, temp\_jptra ) |
| L95 | DO jn = 1, temp\_jptra |

**Table SI17. As Table SI5 but for TOP\_SRC/TRP/trcdmp.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L27 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L88, L189 | INTEGER :: temp\_jptra |
| L89, L191 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L102,L302 | DO jn = 1, temp\_jptra |

**Table SI18. As Table SI5 but for TOP\_SRC/TRP/trcldf.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L30 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L62 | INTEGER :: temp\_jptra |
| L67 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L83 | ALLOCATE( ztrtrd ( 1:jpi, 1:jpj, 1:jpk, 1:temp\_jptra) ) |
| L88 | CASE ( 0 ) ; CALL tra\_ldf\_lap ( kt, nittrc000, 'TRC', gtru, gtrv, gtrui, gtrvi, trb, tra, temp\_jptra ) |
| L91 | CALL tra\_ldf\_iso\_grif( kt, nittrc000, 'TRC', gtru, gtrv, trb, tra, temp\_jptra, rn\_ahtb\_0 ) |
| L93 | CALL tra\_ldf\_iso ( kt, nittrc000, 'TRC', gtru, gtrv, gtrui, gtrvi, trb, tra, temp\_jptra, rn\_ahtb\_0 ) |
| L95 | CASE ( 2 ) ; CALL tra\_ldf\_bilap ( kt, nittrc000, 'TRC', gtru, gtrv, gtrui, gtrvi, trb, tra, temp\_jptra ) |
| L96 | CASE ( 3 ) ; CALL tra\_ldf\_bilapg( kt, nittrc000, 'TRC', trb, tra, temp\_jptra ) |
| L99 | CALL tra\_ldf\_lap ( kt, nittrc000, 'TRC', gtru, gtrv, gtrui, gtrvi, trb, tra, temp\_jptra ) |
| L103 | CALL tra\_ldf\_iso\_grif( kt, nittrc000, 'TRC', gtru, gtrv, trb, tra, temp\_jptra, rn\_ahtb\_0 ) |
| L105 | CALL tra\_ldf\_iso ( kt, nittrc000, 'TRC', gtru, gtrv, gtrui, gtrvi, trb, tra, temp\_jptra, rn\_ahtb\_0 ) |
| L109 | CALL tra\_ldf\_bilap ( kt, nittrc000, 'TRC', gtru, gtrv, gtrui, gtrvi, trb, tra, temp\_jptra ) |
| L112 | CALL tra\_ldf\_bilapg( kt, nittrc000, 'TRC', trb, tra, temp\_jptra ) |

**Table SI19. As Table SI5 but for TOP\_SRC/TRP/trcnxt.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L40 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc, topaz\_ntrc  #endif |
| L99 | INTEGER :: temp\_jptra |
| L107 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra #endif |
| L112,L132,L140,  L148,L155,L167,  L178,L194 | DO jn = 1, temp\_jptra |
| L129 | CALL wrk\_alloc( jpi, jpj, jpk, temp\_jptra, ztrdt ) |
| L187 | & sbc\_trc, sbc\_trc\_b, temp\_jptra ) |
| L188 | ELSE ; CALL tra\_nxt\_fix( kt, nittrc000, 'TRC', trb, trn, tra, temp\_jptra ) |
| L203 | IF( l\_trdtrc) CALL wrk\_dealloc( jpi, jpj, jpk, temp\_jptra, ztrdt ) |

**Table SI20. As Table SI5 but for TOP\_SRC/TRP/trcrad.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L20 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L63-L66 | #ifdef key\_topaz  CALL trc\_rad\_sms( kt, trb, trn, 1, topaz\_prog\_ntrc, cpreserv='Y' ) !TOPAZ model  #else  IF( lk\_cfc ) CALL trc\_rad\_sms( kt, trb, trn, jp\_cfc0 , jp\_cfc1 ) ! CFC model  IF( lk\_c14b ) CALL trc\_rad\_sms( kt, trb, trn, jp\_c14b0, jp\_c14b1 ) ! bomb C14  IF( lk\_pisces ) CALL trc\_rad\_sms( kt, trb, trn, jp\_pcs0 , jp\_pcs1, cpreserv='Y' ) ! PISCES model  IF( lk\_my\_trc ) CALL trc\_rad\_sms( kt, trb, trn, jp\_myt0 , jp\_myt1 ) ! MY\_TRC model  #endif |
| L101-L102 | #ifdef key\_topaz  REAL(wp), DIMENSION (jpi,jpj,jpk,jp\_sms1), INTENT( inout ) :: &  ptrb, ptrn !: before and now traceur concentration  #else  REAL(wp), DIMENSION (jpi,jpj,jpk,jptra), INTENT( inout ) :: &  ptrb, ptrn !: before and now traceur concentration  #endif |

**Table SI21. As Table SI5 but for TOP\_SRC/TRP/trczdf.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L25 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc, topaz\_ntrc  #endif |
| L70 | INTEGER :: ji, jj  INTEGER :: temp\_jptra  REAL(wp) :: r\_vvl |
| L71 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L83 | CALL wrk\_alloc( jpi, jpj, jpk, temp\_jptra, ztrtrd ) |
| L89 | CALL tra\_zdf\_exp( kt, nittrc000, 'TRC', r2dt, nn\_trczdf\_exp, trb, tra, temp\_jptra ) |
| L92 | CALL tra\_zdf\_imp( kt, nittrc000, 'TRC', r2dt, trb, tra, temp\_jptra ) |
| L95 | CASE ( 0 ) ; CALL tra\_zdf\_exp( kt, nittrc000, 'TRC', r2dt, nn\_trczdf\_exp, trb, tra, temp\_jptra ) |
| L96 | CASE ( 1 ) ; CALL tra\_zdf\_imp( kt, nittrc000, 'TRC', r2dt, trb, tra, temp\_jptra ) |
| L99 | #ifdef key\_topaz  DO jn = temp\_jptra+1, topaz\_ntrc !update diagnostics tra<-trb  DO jk = 1, jpk  DO jj = 1, jpj  DO ji = 1, jpi  tra(ji,jj,jk,jn) = trb(ji,jj,jk,jn)  ENDDO  ENDDO  ENDDO  ENDDO  #endif |
| L103,L111,L117 | DO jn = 1, temp\_jptra ! |
| L120 | CALL wrk\_dealloc( jpi, jpj, jpk, jptra, ztrtrd ) |

**Table SI22. As Table SI5 but for TOP\_SRC/TRP/trdmxl\_trc.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L38 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L52 | #ifdef key\_topaz  INTEGER, DIMENSION(topaz\_prog\_ntrc) :: nidtrd, nh\_t  #else  INTEGER, DIMENSION(jptra) :: nidtrd, nh\_t  #endif |
| L82 | #ifdef key\_topaz  ALLOCATE( ztmltrd2(jpi,jpj,jpltrd\_trc,topaz\_prog\_ntrc) , &  #else  ALLOCATE( ztmltrd2(jpi,jpj,jpltrd\_trc,jptra) , &  #endif |
| L397,L1159 | INTEGER :: temp\_jptra |
| L398,L1161 | #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |
| L400, L401, L418, L427, L239, L458, L471, L481, L510, L518, L557, L594, L602, L645, L743, L799, L824, L831, L857, L888, L889, L1325, L1361, L1394 | jptra🡪temp\_jptra |
| L400-L401 | CALL wrk\_alloc( jpi, jpj, temp\_jptra, ztmltot , ztmlres , ztmlatf , ztmlrad )  CALL wrk\_alloc( jpi, jpj, temp\_jptra, ztmltot2, ztmlres2, ztmlatf2, ztmlrad2, ztmltrdm2 ) |
| L418,L427,L439,L458,L471,  L481,L510,L518,L557,L594,  L602,L645,L743,L799,L824,  L831,L857,L1325,L1361,L1394 | DO jn = 1, temp\_jptra |
| L888-L889 | CALL wrk\_dealloc( jpi, jpj, temp\_jptra, ztmltot , ztmlres , ztmlatf , ztmlrad )  CALL wrk\_dealloc( jpi, jpj, temp\_jptra, ztmltot2, ztmlres2, ztmlatf2, ztmlrad2, ztmltrdm2 ) |

**Table SI23. As Table SI5 but for TOP\_SRC/TRP/trdtrc\_oce.F90**

|  |  |
| --- | --- |
| Line | Modfication |
| L12 | #if defined key\_topaz  USE ocean\_var\_4topaz, ONLY: topaz\_prog\_ntrc  #endif |
| L24 | #ifdef key\_topaz  LOGICAL, DIMENSION(topaz\_prog\_ntrc) :: ln\_trdtrc !: large trends diagnostic to write or not (namelist)  #else  LOGICAL, DIMENSION(jptra) :: ln\_trdtrc !: large trends diagnostic to write or not (namelist)  #endif |
| L131 | INTEGER :: temp\_jptra  #ifdef key\_topaz  temp\_jptra=topaz\_prog\_ntrc  #else  temp\_jptra=jptra  #endif |

**Table SI24. namelist\_topaz\_ref**

|  |
| --- |
| !'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''  &namopt ! parameters for optics  !,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,  ! ! file name ! frequency (hours) ! variable ! time interp. ! clim ! 'yearly'/ ! weights ! rotation ! land/sea mask !  ! ! ! (if <0 months) ! name ! (logical) ! (T/F) ! 'monthly' ! filename ! pairing ! filename !  sn\_par = 'par.orca1' , 24 , 'fr\_par' , .true. , .true. , 'yearly' , '' , '' , ''  cn\_dir = './' ! root directory for the location of the dynamical files  ln\_varpar = .true. ! constant swfrac (F) or from a file (T)  parlux = 0.43 ! Fraction of shortwave as PAR  /  !'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''  &namstf ! parameters for surface tracer flux  !,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,  ! ! file name ! frequency (hours) ! variable ! time interp. ! clim ! 'yearly'/ ! weights ! rotation ! land/sea mask !  ! ! ! (if <0 months) ! name ! (logical) ! (T/F) ! 'monthly' ! filename ! pairing ! filename !  sn\_stf(1) = 'depflux\_total.mean.1860\_orca1\_month', -1 , 'NO3\_WET\_DEP' , .true. , .true. , 'yearly' , '' , '' , ''  sn\_stf(2) = 'depflux\_total.mean.1860\_orca1\_month', -1 , 'NO3\_DRY\_DEP' , .true. , .true. , 'yearly' , '' , '' , ''  sn\_stf(3) = 'depflux\_total.mean.1860\_orca1\_month', -1 , 'NH4\_WET\_DEP' , .true. , .true. , 'yearly' , '' , '' , ''  sn\_stf(4) = 'depflux\_total.mean.1860\_orca1\_month', -1 , 'NH4\_DRY\_DEP' , .true. , .true. , 'yearly' , '' , '' , ''  sn\_stf(5) = 'Mineral\_Fe\_Flux\_PI\_orca1\_month' , -1 , 'FLUX' , .true. , .true. , 'yearly' , '' , '' , ''  sn\_stf(6) = 'Soluble\_Fe\_Flux\_PI\_orca1\_month' , -1 , 'FLUX' , .true. , .true. , 'yearly' , '' , '' , ''  sn\_stf(7) = 'ocean\_topaz\_airsea\_flux\_orca1.res' , -12 , 'o2\_flux\_alpha\_ocn' , .false. , .true. , 'yearly' , '' , '' , ''  sn\_stf(8) = 'ocean\_topaz\_airsea\_flux\_orca1.res' , -12 , 'o2\_flux\_sc\_no\_ocn' , .false. , .true. , 'yearly' , '' , '' , ''  sn\_stf(9) = 'ocean\_topaz\_airsea\_flux\_orca1.res' , -12 , 'o2\_flux\_csurf\_ocn' , .false. , .true. , 'yearly' , '' , '' , ''  sn\_stf(10) = 'ocean\_topaz\_airsea\_flux\_orca1.res' , -12 , 'co2\_flux\_alpha\_ocn', .false. , .true. , 'yearly' , '' , '' , ''  sn\_stf(11) = 'ocean\_topaz\_airsea\_flux\_orca1.res' , -12 , 'co2\_flux\_sc\_no\_ocn', .false. , .true. , 'yearly' , '' , '' , ''  sn\_stf(12) = 'ocean\_topaz\_airsea\_flux\_orca1.res' , -12 , 'co2\_flux\_csurf\_ocn', .false. , .true. , 'yearly' , '' , '' , ''  cn\_dir2 = './' !INPUT/flux\_clim/'  sn\_cnt = 12  /  !'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''  &namapr ! Atmospheric prrssure  !,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,  ! ! file name ! frequency (hours) ! variable ! time interp. ! clim ! 'yearly'/ ! weights ! rotation ! land/sea mask !  ! ! ! (if <0 months) ! name ! (logical) ! (T/F) ! 'monthly' ! filename ! pairing ! filename !  ! sn\_apr = 'presatm' , -1 , 'patm' , .true. , .true. , 'yearly' , '' , '' , ''  ! sn\_apr = 'slp.1960.23OCT2012', 6 , 'SLP' , .true. , .true. , 'yearly' , 'weights\_core\_orca2\_bilinear\_noc.nc' , '' , ''  ! sn\_apr = 'slp.1948-2009.23OCT2012\_Fill\_clm', 6 , 'SLP' , .true. , .true. , 'yearly' , 'weights\_rmp\_atm3\_to\_tor1\_nomask\_BILINEA.nc' , '', ''  sn\_apr = 'slp.1948-2009.23OCT2012\_Fill\_clm.ORCA1', 6 , 'SLP' , .true. , .true. , 'yearly' , '' , '', ''  cn\_dir3 = './' ! root directory for the location of the dynamical files  ! ln\_presatm = .false. ! #.true. ! constant atmopsheric pressure (F) or from a file (T)  ln\_presatm = .true. ! constant atmopsheric pressure (F) or from a file (T)  aprcnst = 100000 ! 101000. ! [Pa] |