GC5 coupling

March 24, 2021

Contents

1 Introduction

This document describes all the coupling variables sent between the atmosphere (including land surface), ocean (including sea ice), waves and chemistry (UKESM) components of the Met Office's global coupled models. It is intended as a reference tool so developers can see what variables are sent between the components, what STASH codes they originate from (for UM diagnostics) and any pre or post processing applied to the data.

The current coupling setup documented here is the GC5.0 coupling setup (including UKESM and wave components) but it is intended that this document will evolve with time as models develop.

Sections 2 to 7 list stash codes and variable names of all the coupling variables in the UM. Section 8 shows how data is averaged in the UM and send to NEMO. Sections 9 and onwards show tables of all pre and post processing done on each coupling variable. They are written in such a way that they contain snippets of code so the developer should be able to easily find the source code.

2 Atmospheric variables that receive data from the ocean

Field to couple	**STASH code**	**variable name**
Sea surface temperature	507	tstar_sea
Sea ice layer temperature on categories	415	ti_cat
Sea ice fraction on categories	413	ice_fract_cat
Sea ice thickness on categories	414	ice_thick_cat
Snow thickness on categories	416	snodep_sea_cat
Ocean zonal velocity	28	u_sea
Ocean meridional velocity	29	v_sea
Melt pond fraction on categories	428	pond_frac_cat
Melt pond depth on categories	429	pond_depth_cat
Sea ice conductivity on categories	440	ice_k_cat

3 Atmospheric variables that send data to the ocean

Field to couple	**STASH code**	**variable name**
Zonal wind stress	3392	taux_ssi in diagnostics_bl.F90
Meridional wind stress	3394	tauy_ssi in diagnostics_bl.F90
Surface net solar radiation	1203	netsea in riadnostics_rad.F90 or swsea in rad_ctl.F90
Sensible heat flux	3228	h_sea in diagnstics_bl.F90
Evaporation	3232	e_sea in diagnstics_bl.F90
Surface net longwave radiation	2203	netsea in diagnostics_rad.F90 or lwsea in rad_ctl.F90
Large scale rain	4203	ls_rain in diagnostics_lsrain.F90
Convective rain	5205	conv_rain in diagnostics_conf.F90
Large scale snow	4204	ls_snow in diagnostics_lsrain.F90
Convective rain	5206	conv_snow in diagnostics_conf.F90
Evaporation	3232	e_sea in diagnstics_bl.F90
Sublimation from sea ice on categories	3509	ei_sice in diagnstics_bl.F90
10m wind speed	3230	c_w10 in d1 array (prognostic)
Top melt from sea ice on categories	3257	sf_diag%sice_mlt_htf in diagnstics_bl.F90
Surface skin temperature of sea ice on categories	441	tstar_sice_cat in d1 array (prognostic) or tstar_sice_sicat in surf_couple_implicit_mod.F90
Surface conductive heat flux of sea ice on categories	3510	surf_ht_flux_sice in diagnstics_bl.F90

4 UKESM specific atmospheric variables that receive data from the ocean

Field to couple	**STASH code**	**variable name**
DMS concentration in seawater	132	dms_conc
CO2 ocean flux	250	CO2flux
Ocean near surface chlorophyll	96	chloro_sea

5 UKESM specific atmospheric variables that send data to the ocean

Field to couple	**STASH code**	**variable name**
Total Greenland ice mass	240	snodep_tile in ice_sheet_mass.F90
Total Antarctic ice mass	240	snodep_tile in ice_sheet_mass.F90
Partial CO2 pressure in surface layer	0,252 or fixed value	co2 in tracer_restore.F90 or puta2o.F90
Total dust flux	3,440	c_dust_dep in oasis3_puta2o.F90

6 Atmospheric variables that receive data from waves

Field to couple	**STASH code**	**variable name**
Charnoc's coefficient	517	charnock_w

7 Atmospheric variables that send data to waves

Field to couple	**STASH code**	**variable name**
Zonal wind stress (WW3 vn7)*	3392	taux_ssi in diagnostics_bl.F90
Meridional wind stress (WW3 vn7)*	3394	tauy_ssi in diagnostics_bl.F90
Surface air density (WW3 vn7)*	3562	rho1 in diagnostics_bl.F90
Zonal neutral 10m wind speed (WW3 vn7)*	3368	c_u10m_w in d1 array
Meridional neutral 10m wind speed (WW3 vn7)*	3369	c_v10m_w in d1 array
Zonal 10m wind speed (WW3 vn4.18)	3209	u10m in diagnostics_bl.F90
Meridional 10m wind speed (WW3 vn4.18)	3210	v10m in diagnostics_bl.F90

* Functionality not present in the UM trunk yet. See branch
fcm:um.xm_br/dev/johnmedwards/vn11.1_wave_coupling_stress@82620
for an implementation in UM vn11.1. Notice that this branch will modify the UM variables used for 10m wind coupling with waves.

Current (ocean/wave coupled) operational configurations are based on WAVEWATCHIII vn4.18, and coupling research is taking place using WAVEWATCHIII vn7. It is the intention of having wave coupling capabilities in GC5 using WAVEWATCHIIII vn7. as an optional configuration.

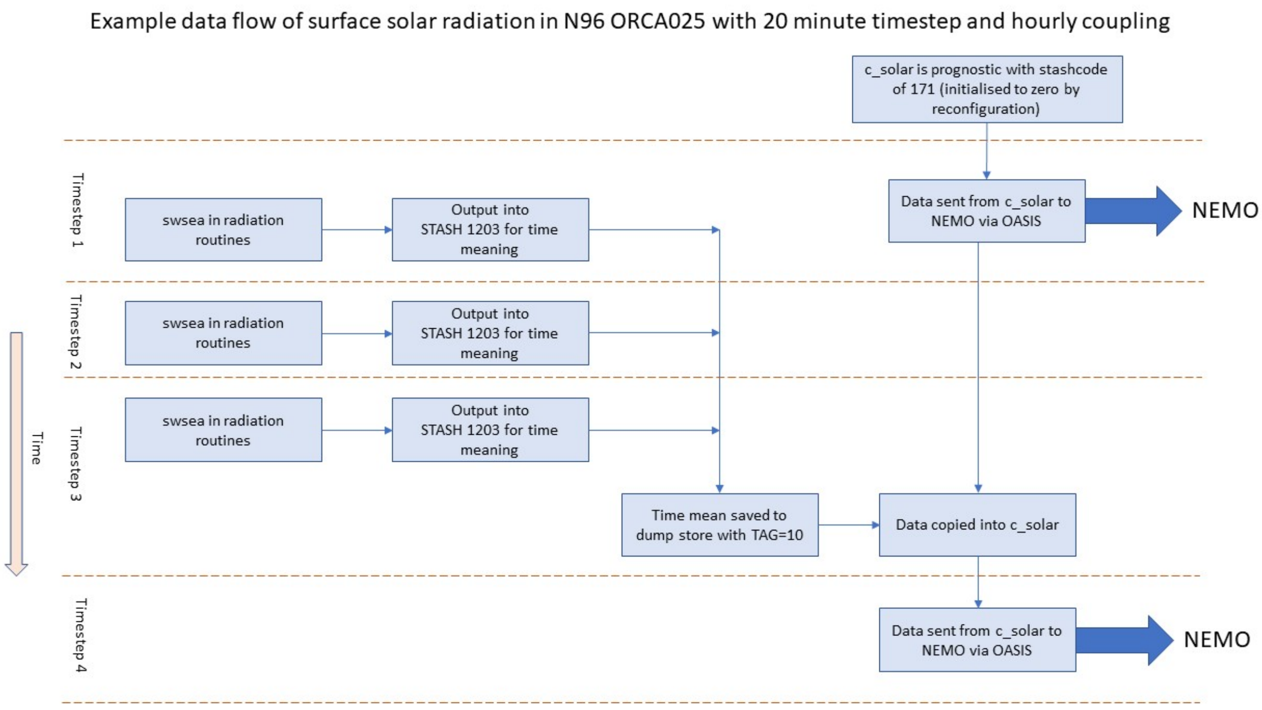

Figure 1: Example of data flow

8 Example of data flow from Atmosphere to Ocean

Figure 1 shows how surface solar radiation (swsea) gets from the radiation routines where it is calculated into NEMO. At each timestep it is output via stash but instead of going to disk it goes to a data store (aka dump store). In the STASH usage panel this data must be listed as going to "Dump store with user specified TAG" and TAG=10. It must also be time meaned with the same meaning period as the coupling frequency (1 hour). This is then copied into a prognostic variable within the oasis_updatecpl.F90 routine. At the start of the next timestep it is sent to NEMO via OASIS, within the oasis3_puta2o.F90 routine.

9 Ocean to atmosphere exchange

9.1 Sea surface temperatures (SSTs)

Field to couple	Sea surface temperatures (SST)
From prognostic to temporary variable in sbccpl.F90	ztmp1(:,:) = tsn(:,:,1,jp_tem)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	ztmp1(:,:) = ztmp1(:,:) + rt0 (converts from Celcius to Kelvin)
NEMO index to ssnd array	jps_toce=2
NEMO namelist controlling string	sn_snd_temp%cldes='oce only'
NEMO namcouple variable name	'O_SSTSST'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'ocn_sst'
UM TRANSDEF index and vind variable	vind_ocn_sst = 25
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_sst(:,:,:)

Operations applied to recieved data	IF (ocn_sst < 1.0) THEN ocn_sst=ocn_sst_orig
	(If SSTs coming through coupler are invalid then keep original SSTs)
	IF (ocn_freeze > 0.0) ocn_sst=tfs
	(If there is sea ice set the SST to the freezing point of sea water - not salinity dependent)

Assign portion of D1 array by stash code	jtstar_sea = si(507,Sect_No,im_index)
Point to D1 array	tstar_sea => d1(jtstar_sea)
Copy to D1 array	tstar_sea=ocn_sst

9.2 Sea ice layer temperature

Field to couple	Sea ice layer temperature
From prognostic to temporary variable in sbccpl.F90	ztmp3(:,:,1:jpl) = t1_ice(:,:,1:jpl) * a_i(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	Multipled by ice fraction before coupling (see line above)
NEMO index to ssnd array	jps_ttilyr = 38
NEMO namelist controlling factor	sn_snd_ttilyr='weighted ice','yes'
NEMO namcouple variable name	'O_TtiLyr_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'oicetXX'
UM TRANSDEF index and vind variable	vind_ocn_icetn = 41,42,43,44,45
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_icetn(:,:,n:n)

Operations applied to recieved data	IF (ocn_freezen < aicenmin) THEN ocn_icetn(i,j,k)=0.0
	(If ice area coming through coupler is less than a minimum area then set temperature to zero)
	IF (ocn_freezen(i,j,k) > 0.0) THEN ocn_icetn(i,j,k)=ocn_icetn(i,j,k)/ocn_freezen(i,j,k)
	(Divide by ice fraction)
	IF (ocn_icetn(i,j,k) > TI_max) THEN ocn_icetn(i,j,k) = TI_max
	(Apply a maximum temperature)
	IF ((ocn_icetn(i,j,k) < TI_min) .AND. (ocn_icetn(i,j,k) /= 0.0)) THEN ocn_icetn(i,j,k) = TI_min
	(Apply a minimum temperature - unless it has already been reset to zero)
	ocn_icet_gb(i,j) = ocn_icet_gb(i,j) + (ocn_freezen(i,j,k)/ocn_freeze(i,j,1)) * ocn_icetn(i,j,k)
	(Generate a weighted mean of ice layer temperatures, over all cats, for output diagnostics)

Assign portion of D1 array by stash code	jti_cat = si(415,Sect_No,im_index)
Point to D1 array	ti_cat => d1(jti_cat)
Copy to D1 array	ti_cat(i,j,k)=ocn_icetn(i,j,k)

9.3 Sea ice fraction

Field to couple	Sea ice fraction
From prognostic to temporary variable in sbccpl.F90	ztmp3(:,:,1:jpl) = a_i(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	None
NEMO index to ssnd array	jps_fice=1
NEMO namelist controlling factor	k_ice (aka nn_ice) / = 0
NEMO namcouple variable name	'OIceFrc_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'ofrznXX'
UM TRANSDEF index and vind variable	vind_ocn_freezen = 26,27,28,29,30
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_freezen(:,:,:)

Operations applied to recieved data	IF (ocn_hicen/ocn_freezen < hi_min) THEN ocn_freezen = 0.0
	(If ice thickness coming through coupler is less than a minimum thickness then remove all ice)
	IF (ocn_freezen < aicenmin) THEN ocn_freezen=0.0
	(If ice area coming through coupler is less than a minimum area then remove all ice)
	IF (ocn_sst < 1.0) THEN ocn_freezen = ice_fract_cat
	(If invalid SSTs then retain the sea ice fraction from before the coupling)
	IF (ocn_freeze(i,j,1) > 1.0) THEN ocn_freezen(i,j,k)=ocn_freezen(i,j,k)/ocn_freeze(i,j,1)
	(If total ice fraction is greater than 1 then scale all category ice fractions down so that the total ice fraction is 1)

Assign portion of D1 array by stash code	jice_fract_cat = si(413,Sect_No,im_index)
Point to D1 array	ice_fract_cat => d1(jice_fract_cat)
Copy to D1 array	ice_fract_cat = ocn_freezen

9.4 Sea ice thickness

Field to couple	Sea ice thickness
From prognostic to temporary variable in sbccpl.F90	ztmp3(:,:,1:jpl) = h_i(:,:,1:jpl) * a_i(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	(Ice thickness on each category is multiplied by ice area on each category. See line above)
NEMO index to ssnd array	jps_hice = 7
NEMO namelist controlling factor	sn_snd_thick='weighted ice and snow','yes'
NEMO namcouple variable name	'OIceTck_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'ohicnXX'
UM TRANSDEF index and vind variable	vind_ocn_hicen = 36,37,38,39,40
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_hicen(:,:,n:n)

Operations applied to recieved data	IF (ocn_freezen < aicenmin) THEN ocn_hicen=0.0
	(If ice area coming through coupler is less than a minimum area then remove all ice)
	ocn_hicen=ocn_hicen+ocn_snowthickn*(kappai/kappai_snow)
	(Increase ice thickness by the snow thickness, taking into account relative densities)
	IF (ocn_freezen > 0.0) THEN ocn_hicen=ocn_hicen/ocn_freezen
	(Ice thickness on each category is divided by ice area on each category, undoing the scaling that was applied before coupling.)

Assign portion of D1 array by stash code	jice_thick_cat = si(414,Sect_No,im_index)
Point to D1 array	ice_thick_cat => d1(jice_thick_cat)
Copy to D1 array	ice_thick_cat=ocn_hicen

9.5 Snow thickness (snow on sea ice)

Field to couple	Snow thickness
From prognostic to temporary variable in sbccpl.F90	ztmp4(:,:,1:jpl) = h_s(:,:,1:jpl) * a_i(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	(Snow thickness on each category is multiplied by ice area on each category. See line above)
NEMO index to ssnd array	jps_hsnw = 8
NEMO namelist controlling factor	sn_snd_thick='weighted ice and snow','yes'
NEMO namcouple variable name	'OSnwTck_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'osnwtnXX'
UM TRANSDEF index and vind variable	vind_ocn_snowthickn = 31,32,33,34,35
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_snowthickn(:,:,n:n)

Operations applied to recieved data	IF (ocn_freezen < aicenmin) THEN ocn_snowthickn=0.0
	(If ice area coming through coupler is less than a minimum area then remove all ice)
	IF (ocn_freezen > 0.0) THEN ocn_snowthickn=ocn_snowthickn*rhosnow/ocn_freezen
	(Ice thickness on each category is divided by ice area on each category, undoing the scaling that was applied before coupling. Also convert to kg/m2.)

Assign portion of D1 array by stash code	jsnodep_sea_cat= si(416,Sect_No,im_index)
Point to D1 array	snodep_sea_cat => d1(jjsnodep_sea_cat)
Copy to D1 array	snodep_sea_cat = ocn_snowthickn

9.6 Ocean zonal velocity

Field to couple	Ocean zonal velocity
From prognostic to temporary variable in sbccpl.F90	zotx1 is calculated by averaging between the ocean U velocities (un) in the corners to give a U velocity along the side of the grid box. This is then combined with sea ice U velocities (u_ice, also converted to sides of grid box) by using leads and sea ice fractions to form a weighted average.
Operations applied to temporary variable in sbccpl.F90 prior to coupling	zotx1 is then averaged over surrounding 4 points to move from the T grid to the U,V grid. It is then rotated using repcmo
NEMO index to ssnd array	jps_ocx1 = 9
NEMO namelist controlling factor	sn_snd_crt='mixed oce-ice','no','spherical','eastward-northward','U,V'
NEMO namcouple variable name	'O_OCurx1'
OASIS Remapping method	Bilinear
UM namcouple variable name	'sunocean'
UM TRANSDEF index and vind variable	vind_ocn_u = 51
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_u(:,:,n:n)
Operations applied to recieved data	Polar row corrected using Correct_Polar_UV for ENDGame. We will not need to do this for the LFRic cubesphere grid.
Assign portion of D1 array by stash code	ju_sea = si(28,Sect_No,im_index)
Point to D1 array	u_sea => d1(ju_sea)
Copy to D1 array	u_sea(udims%i_start:udims%i_end,udims%j_start:udims%j_end) = ocn_u(1:oasis_imt,1:oasis_jmt_u) Indexing corrected in the line above.

9.7 Ocean meridional velocity

Field to couple	Ocean meridional velocity
From prognostic to temporary variable in sbccpl.F90	zoty1 is calculated by averaging between the ocean V velocities (vn) in the corners to give a V velocity along the side of the grid box. This is then combined with sea ice V velocities (v_ice, also converted to sides of grid box) by using leads and sea ice fractions to form a weighted average.
Operations applied to temporary variable in sbccpl.F90 prior to coupling	zoty1 is then averaged over surrounding 4 points to move from the T grid to the U,V grid. It is then rotated using repcmo
NEMO index to ssnd array	jps_ocy1 = 10
NEMO namelist controlling factor	sn_snd_crt='mixed oce-ice','no','spherical','eastward-northward','U,V'
NEMO namcouple variable name	'O_OCury1'
OASIS Remapping method	Bilinear
UM namcouple variable name	'svnocean'
UM TRANSDEF index and vind variable	vind_ocn_v = 52
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_v(:,:,n:n)
Operations applied to recieved data	Polar row corrected using Correct_Polar_UV for ENDGame. We will not need to do this for the LFRic cubesphere grid.
Assign portion of D1 array by stash code	jv_sea = si(29,Sect_No,im_index)
Point to D1 array	v_sea => d1(jv_sea)
Copy to D1 array	v_sea(vdims%i_start:vdims%i_end,vdims%j_start:vdims%j_end) = ocn_v(1:oasis_imt,1:oasis_jmt_v) Indexing corrected in the line above.

9.8 Melt pond fraction

Field to couple	Melt pond fraction
From prognostic to temporary variable in sbccpl.F90	ztmp3(:,:,1:jpl) = a_ip_eff(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	None
NEMO index to ssnd array	jps_a_p = 34
NEMO namelist controlling factor	sn_snd_mpnd='ice only','yes'
NEMO namcouple variable name	'OPndFrc_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'pndfrcnX'
UM TRANSDEF index and vind variable	vind_pond_frac_n = 56,57,58,59,60
UM pointer used to point to recieved data	transient_o2a(tc)%field => pond_frac_n(:,:,n:n)

Operations applied to recieved data	IF (ocn_freezen < aicenmin) THEN pond_frac_n(i,j,k)=0.0
	(If ice area coming through coupler is less than a minimum area then remove all melt ponds)
	IF (pond_frac_n(i,j,k) < 0.0) THEN pond_frac_n(i,j,k) = 0.0
	(Do not allow negative pond fractions)
	IF (pond_frac_n(i,j,k) > 1.0) THEN pond_frac_n(i,j,k) = 1.0
	(Do not allow pond fractions greater than 1)

Assign portion of D1 array by stash code	jpond_frac_cat = si(428,Sect_No,im_index)
Point to D1 array	pond_frac_cat => d1(jpond_frac_cat)
Copy to D1 array	pond_frac_cat(i,j,k)=pond_frac_n(i,j,k)

9.9 Melt pond depth

Field to couple	Melt pond depth
From prognostic to temporary variable in sbccpl.F90	ztmp4(:,:,1:jpl) = h_ip(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	None
NEMO index to ssnd array	jps_ht_p = 35
NEMO namelist controlling factor	sn_snd_mpnd='ice only','yes'
NEMO namcouple variable name	'OPndTck_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'pndtcknX'
UM TRANSDEF index and vind variable	vind_pond_depth_n = 61,62,63,64,65
UM pointer used to point to recieved data	transient_o2a(tc)%field => pond_depth_n(:,:,n:n)

Operations applied to recieved data	IF (ocn_freezen < aicenmin) THEN pond_depth_n(i,j,k)=0.0
	(If ice area coming through coupler is less than a minimum area then remove all melt ponds)
	IF (pond_frac_n(i,j,k) < 0.0) THEN pond_depth_n(i,j,k) = 0.0
	(When resetting negative pond fractions to zero do the same for pond depths)
	IF (pond_depth_n(i,j,k) < 0.0) THEN pond_depth_n(i,j,k) = 0.0
	(Do not allow negative pond depths)

Assign portion of D1 array by stash code	jpond_depth_cat = si(429,Sect_No,im_index)
Point to D1 array	pond_depth_cat => d1(jpond_depth_cat)
Copy to D1 array	pond_depth_cat(i,j,k)=pond_depth_n(i,j,k)

9.10 Sea ice conductivity

Field to couple	Sea ice conductivity
From prognostic to temporary variable in sbccpl.F90	ztmp3(:,:,1:jpl) = cnd_ice(:,:,1:jpl) * a_i(:,:,1:jpl)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	Multipled by ice fraction to get grid box mean (see line above)
NEMO index to ssnd array	jps_kice = 36
NEMO namelist controlling factor	sn_snd_cond='weighted ice','yes'
NEMO namcouple variable name	'OIceKn_catXX'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'oicekXX'
UM TRANSDEF index and vind variable	vind_ocn_icekn = 46,47,48,49,50
UM pointer used to point to recieved data	transient_o2a(tc)%field => ocn_icekn(:,:,n:n)

Operations applied to recieved data	IF (ocn_freezen < aicenmin) THEN ocn_icekn(i,j,k)=0.0
	(If ice area coming through coupler is less than a minimum area then set copnductivity to zero)
	IF (ocn_freezen(i,j,k) > 0.0) THEN ocn_icekn(i,j,k)=ocn_icekn(i,j,k)/ocn_freezen(i,j,k)
	(Divide by ice fraction)

Assign portion of D1 array by stash code	jice_k_cat = si(440,Sect_No,im_index)
Point to D1 array	ice_k_cat => d1(jice_k_cat)
Copy to D1 array	ice_k_cat(i,j,k) = ocn_icekn(i,j,k)

10 Atmosphere to ocean exchange

10.1 Zonal wind stresses

Field to couple	Zonal wind stresses
Original diagnostic variable and routine it is saved to STASH in	taux_ssi in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3392 hourly means
Prognostic to carry the data into the next timestep	STASH 176 using jc_taux pointer and c_taux variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_taux(i,j) = d1(ja_taux+i-1+((j-1)*oasis_imt)) ja_taux is the index of the d1 array where the previous hourly mean is stored c_taux is the prognostic which will carry wind stress into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => taux(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	None for UM. For LFRic we should make sure that this wind stress is in the zonal (East-West) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_taux = 23
UM namcouple variable name	'taux'
OASIS Remapping method	Bilinear
NEMO namcouple variable name	'O_OTaux1'
NEMO index to frcv array	jpr_otx1 = 1
NEMO namelist controlling factor	sn_rcv_tau='oce only','no','spherical','eastward-northward','U,V,F' srcv(jpr_otx1)%clgrid = 'U'
Operations applied in sbccpl.F90 after coupling	CALL rot_rep(frcv(jpr_otx1)%z3(:,:,1), frcv(jpr_oty1)%z3(:,:,1), srcv(jpr_otx1)%clgrid, 'en− >i', ztx) (Zonal and meridional wind stresses are rotated so that that are aligned with the NEMO grid)
Final variable name in NEMO-SI3	utau(:,:) = frcv(jpr_otx1)%z3(:,:,1)

10.2 Meridional wind stresses

Field to couple	Meridional wind stresses
Original diagnostic variable and routine it is saved to STASH in	tauy_ssi in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3394 hourly means
Prognostic to carry the data into the next timestep	STASH 177 using jc_tauy pointer and c_tauy variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_tauy(i,j) = d1(ja_tauy+i-1+((j-1)*oasis_imt)) ja_tauy is the index of the d1 array where the previous hourly mean is stored c_tauy is the prognostic which will carry wind stress into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => tauy(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	None for UM. For LFRic we should make sure that this wind stress is in the meridional (North-South) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_tauy = 24
UM namcouple variable name	'tauy'
OASIS Remapping method	Bilinear
NEMO namcouple variable name	'O_OTauy1'
NEMO index to frcv array	jpr_oty1 = 2
NEMO namelist controlling factor	sn_rcv_tau='oce only','no','spherical','eastward-northward','U,V,F' srcv(jpr_oty1)%clgrid = 'U'
Operations applied in sbccpl.F90 after coupling	CALL rot_rep(frcv(jpr_otx1)%z3(:,:,1), frcv(jpr_oty1)%z3(:,:,1), srcv(jpr_otx1)%clgrid, 'en− >j', zty) (Zonal and meridional wind stresses are rotated so that that are aligned with the NEMO grid)
Final variable name in NEMO-SI3	vtau(:,:) = frcv(jpr_oty1)%z3(:,:,1)

10.3 Solar radiation

Field to couple	Solar radiation
Original diagnostic variable and routine it is saved to STASH in	netsea in radiation_control/diagnostics_rad.F90 or swsea in radiation_control/rad_ctl.F90 = Solar radiation in the sea (leads) part of the grid box
STASH code and time meaning period	STASH 1203 hourly means
Prognostic to carry the data into the next timestep	STASH 171 using jc_solar pointer and c_solar variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_solar(i,j) = d1(ja_solar+i-1+((j-1)*oasis_imt)) ja_solar is the index of the d1 array where the previous hourly mean is stored c_solar is the prognostic which will carry solar radiation into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => solar2d(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	None.
UM TRANSDEF index and vind variable	vind_solar = 54
UM namcouple variable name	'solar'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'O_QsrOce'
NEMO index to frcv array	jpr_qsroce = 13
NEMO namelist controlling factor	sn_rcv_qsr='oce only'
Temporary variable in sbccpl.F90 to manipulate data	zqsr(:,:) = frcv(jpr_qsroce)%z3(:,:,1)
Operations applied in sbccpl.F90 after coupling	None
Final variable name in NEMO-SI3	qsr(:,:) = zqsr(:,:)

10.4 Surface heat flux

Field to couple	Surface heat flux (combination of sensible, latent and longwave heat fluxes)
Original diagnostic variable and routine it is saved to STASH in	h_sea in boundary_layer/diagnostics_bl.F90 = Sensible heat flux in the sea (leads) part of the grid box e_sea in boundary_layer/diagnostics_bl.F90 = Evaporation in the sea (leads) part of the grid box netsea in radiation_control/diagnostics_rad.F90 or lwsea in radiation_control/rad_ctl.F90 = Longwave in the sea (leads) part of the grid box
STASH code and time meaning period	Sensible heat flux = STASH 3228 hourly means Evaporation = STASH 3232 hourly means Surface longwave = STASH 2203 hourly means
Prognostic to carry the data into the next timestep	STASH 179 using jc_sensible pointer and c_sensible variable name. STASH 181 using jc_evap pointer and c_evap variable name. STASH 174 using jc_longwave pointer and c_longwave variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_sensible(i,j) = d1(ja_sensible+i-1+((j-1)*oasis_imt)) ja_sensible is the index of the d1 array where the hourly mean of sesible heat flux is stored c_sensible is the prognostic which will carry sensible heat flux into the next timestep c_evap(i,j) = d1(ja_evap+i-1+((j-1)*oasis_imt)) ja_evap is the index of the d1 array where the hourly mean of evaporation is stored c_evap is the prognostic which will carry evaporation into the next timestep c_longwave(i,j) = d1(ja_longwave+i-1+((j-1)*oasis_imt)) ja_longwave is the index of the d1 array where the hourly mean of surface longwave is stored c_longwave is the prognostic which will carry surface longwave into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => heatflux(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	If land fraction less than 1 then heatflux(i,j,1)=longwave2d(i,j) - (sensible2d(i,j)+latentHeatOfCond*evap2d(i,j,1)) Merges together longwave, sensible and latent heat fluxes into a total heat flux.
UM TRANSDEF index and vind variable	vind_heatflux = 1
UM namcouple variable name	'heatflux'
OASIS Remapping method	Second order conservative destination area
NEMO namcouple variable name	'O_QnsOce'
NEMO index to frcv array	jpr_qnsoce = 16
NEMO namelist controlling factor	sn_rcv_qns='oce only'
Temporary variable in sbccpl.F90 to manipulate data	zqns(:,:) = frcv(jpr_qnsoce)%z3(:,:,1)
Operations applied in sbccpl.F90 after coupling	zqns(:,:) = zqns(:,:) - zemp(:,:) * sst_m(:,:) * rcp (Increases in MASS require an increase in the heat content as the model is formulated using heat content, not temperature) zqns(:,:) = zqns(:,:) - frcv(jpr_snow)%z3(:,:,1) * rLfus (As snow melts it cools the sea surface)
Final variable name in NEMO-SI3	qns(:,:) = zqns(:,:)

10.5 Total rain

Field to couple	Total rain
Original diagnostic variable and routine it is saved to STASH in	ls_rain in large_scale_precipitation/diagnostics_lsrain.F90 = large scale rain conv_rain in convection/diagnostics_conv.F90 = convective rain
STASH code and time meaning period	Large scale rain = STASH 4203 hourly means Convective rain = STASH 5205 hourly means
Prognostic to carry the data into the next timestep	STASH 186 using jc_lsrain pointer and c_lsrain variable name. STASH 188 using jc_cvrain pointer and c_cvrain variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_lsrain(i,j) = d1(ja_lsrain+i-1+((j-1)*oasis_imt)) ja_lsrain is the index of the d1 array where the hourly mean of large scale rain is stored c_lsrain is the prognostic which will carry large scale rain into the next timestep c_cvrain(i,j) = d1(ja_cvrain+i-1+((j-1)*oasis_imt)) ja_cvrain is the index of the d1 array where the hourly mean of convective rain is stored c_cvrain is the prognostic which will carry convective rain into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => totalrain(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	IF (l_param_conv) THEN rainconv(i,j) = c_cvrain(i,j) ELSE rainconv(i,j) = 0.0 (Set convective rain to zero if we don't use a convection scheme) totalrain(i,j,1)=c_lsrain(i,j)+rainconv(i,j) (Total rain is large scale rain and convective rain combined)
UM TRANSDEF index and vind variable	vind_train = 5
UM namcouple variable name	'train'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OTotRain'
NEMO index to frcv array	jpr_rain = 19
NEMO namelist controlling factor	sn_rcv_emp='conservative','yes'
Temporary variable in sbccpl.F90 to manipulate data	zemp(:,:) = frcv(jpr_tevp)%z3(:,:,1) - (frcv(jpr_rain)%z3(:,:,1) + frcv(jpr_snow)%z3(:,:,1)) (zemp is evaporation minus procipitation (combined rain and snow)) ztprecip(:,:) = frcv(jpr_rain)%z3(:,:,1) + frcv(jpr_snow)%z3(:,:,1) (total precipitation is rain and snow)
Operations applied in sbccpl.F90 after coupling	(Combined with evaporation and snow (see lines above)) zemp_tot(:,:) = frcv(jpr_tevp)%z3(:,:,1) - ztprecip(:,:) (Evaporation minus precipitation) Also snow blowing off of ice into the sea alters zemp_ice and zemp_oce
Final variable name in NEMO-SI3	emp(:,:) = zemp(:,:) tprecip (:,:) = ztprecip (:,:) emp_tot (:,:) = zemp_tot (:,:)

10.6 Total snow

Field to couple	Total snow
Original diagnostic variable and routine it is saved to STASH in	ls_snow in large_scale_precipitation/diagnostics_lsrain.F90 = large scale snow conv_snow in convection/diagnostics_conv.F90 = convective snow
STASH code and time meaning period	Large scale snow = STASH 4204 hourly means Convective snow = STASH 5206 hourly means
Prognostic to carry the data into the next timestep	STASH 187 using jc_lssnow pointer and c_lssnow variable name. STASH 189 using jc_cvsnow pointer and c_cvsnow variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_lssnow(i,j) = d1(ja_lssnow+i-1+((j-1)*oasis_imt)) ja_lssnow is the index of the d1 array where the hourly mean of large scale snow is stored c_lssnow is the prognostic which will carry large scale snow into the next timestep c_cvsnow(i,j) = d1(ja_cvsnow+i-1+((j-1)*oasis_imt)) ja_cvsnow is the index of the d1 array where the hourly mean of convective snow is stored c_cvsnow is the prognostic which will carry convective snow into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => totalsnow(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	IF (l_param_conv) THEN snowconv(i,j) = c_cvsnow(i,j) ELSE snowconv(i,j) = 0.0 (Set convective snow to zero if we don't use a convection scheme) totalsnow(i,j,1)=c_lssnow(i,j)+snowconv(i,j) (Total snow is large scale snow and convective snow combined)
UM TRANSDEF index and vind variable	vind_tsnow = 6
UM namcouple variable name	'tsnow'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OTotSnow'
NEMO index to frcv array	jpr_snow = 20
NEMO namelist controlling factor	sn_rcv_emp='conservative','yes'
Temporary variable in sbccpl.F90 to manipulate data	zemp(:,:) = frcv(jpr_tevp)%z3(:,:,1) - (frcv(jpr_rain)%z3(:,:,1) + frcv(jpr_snow)%z3(:,:,1)) (zemp is evaporation minus procipitation (combined rain and snow)) zsprecip(:,:) = frcv(jpr_snow)%z3(:,:,1) (total snow variable) ztprecip(:,:) = frcv(jpr_rain)%z3(:,:,1) + frcv(jpr_snow)%z3(:,:,1) (total precipitation is rain and snow)
Operations applied in sbccpl.F90 after coupling	(Combined with evaporation and rain (see lines above)) zemp_tot(:,:) = frcv(jpr_tevp)%z3(:,:,1) - ztprecip(:,:) (Evaporation minus precipitation) zemp_ice is sublimation (zevap_ice_total) minus snow Also snow blowing off of ice into the sea alters zemp_ice and zemp_oce
Final variable name in NEMO-SI3	emp(:,:) = zemp(:,:) sprecip (:,:) = zsprecip (:,:) tprecip (:,:) = ztprecip (:,:) emp_tot (:,:) = zemp_tot (:,:)

10.7 Evaporation

Field to couple	Evaporation
Original diagnostic variable and routine it is saved to STASH in	e_sea in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3232 hourly means = evaporation in the sea (leads) part of the grid box
Prognostic to carry the data into the next timestep	STASH 181 using jc_evap pointer and c_evap variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_evap(i,j) = d1(ja_evap+i-1+((j-1)*oasis_imt)) ja_evap is the index of the d1 array where the hourly mean of evaporation is stored c_evap is the prognostic which will carry evaporation into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => evap2d(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	None although it is used to modify the surface heat flux (see section above)
UM TRANSDEF index and vind variable	vind_evap2d = 7
UM namcouple variable name	'evap2d'
OASIS Remapping method	Second order conservative destination area
NEMO namcouple variable name	'OTotEvap'
NEMO index to frcv array	jpr_tevp = 21
NEMO namelist controlling factor	sn_rcv_emp='conservative','yes'
Temporary variable in sbccpl.F90 to manipulate data	zemp(:,:) = frcv(jpr_tevp)%z3(:,:,1) - (frcv(jpr_rain)%z3(:,:,1) + frcv(jpr_snow)%z3(:,:,1)) (zemp is evaporation minus procipitation (combined rain and snow)) zevap_oce(:,:) = frcv(jpr_tevp)%z3(:,:,1) - zevap_ice_total(:,:) * picefr(:,:) (evaporation over the ocean is total evaporation minus sea ice evaporation)
Operations applied in sbccpl.F90 after coupling	(Combined with rain and snow (see lines above)) zemp_tot(:,:) = frcv(jpr_tevp)%z3(:,:,1) - ztprecip(:,:) (Evaporation minus precipitation) Also snow blowing off of ice into the sea alters zemp_ice and zemp_oce
Final variable name in NEMO-SI3	emp(:,:) = zemp(:,:) emp_tot (:,:) = zemp_tot (:,:)

10.8 Sublimation from sea ice

Field to couple	Sublimation from sea ice (aka ice evaporation)
Original diagnostic variable and routine it is saved to STASH in	ei_sice in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3509 hourly means = sublimation from sea ice (on categories)
Prognostic to carry the data into the next timestep	STASH 182 using jc_sublim pointer and c_sublim variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_sublim(i,j,k)=d1(ja_sublim) ja_sublim is the index of the d1 array where the hourly mean of sublimation is stored c_sublim is the prognostic which will carry sublimation into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => sublim(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	sublim(i,j,k) = sublim(i,j,k) / ice_fract_cat_future(i,j,k) (sublimation is converted from a grid box mean to a sea ice mean using ice fractions that have just been passed from NEMO)
UM TRANSDEF index and vind variable	vind_sublim = 18,19,20,21,22
UM namcouple variable name	'sublimXX'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OIceEvap_catXX'
NEMO index to frcv array	jpr_ievp = 22
NEMO namelist controlling factor	sn_rcv_emp='conservative','yes'
Temporary variable in sbccpl.F90 to manipulate data	zevap_ice(:,:,:) = frcv(jpr_ievp)%z3(:,:,:) * a_i_last_couple(:,:,:) / a_i(:,:,:) (zevap_ice is sea ice mean sublimation but converted up to grid box mean and back down to sea ice mean again using ice areas from different timesteps)
Operations applied in sbccpl.F90 after coupling	(Uses the conversion above using ice areas from different timesteps) zevap_ice_total is the sea ice mean sublimation averaged over all categories (weighted averaging)
Final variable name in NEMO-SI3	emp(:,:) = zemp(:,:) evap_ice(:,:,:) = zevap_ice(:,:,:)

10.9 10m wind speed

Field to couple	10m wind speed
Original diagnostic variable and routine it is saved to STASH in	ws10m_p in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3230 hourly means = 10m wind speeds
Prognostic to carry the data into the next timestep	STASH 191 using jc_w10 pointer and c_w10 variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_w10(i,j) = d1(ja_w10+i-1+((j-1)*oasis_imt)) ja_w10 is the index of the d1 array where the hourly mean of 10m wind speed is stored c_w10 is the prognostic which will carry 10m wind speed into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => w10(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	None
UM TRANSDEF index and vind variable	vind_w10 = 4
UM namcouple variable name	'w10'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'O_Wind10'
NEMO index to frcv array	jpr_w10m = 26
NEMO namelist controlling factor	sn_rcv_w10m='coupled'
Temporary variable in sbccpl.F90 to manipulate data	None
Operations applied in sbccpl.F90 after coupling	None
Final variable name in NEMO-SI3	wndm(:,:) = frcv(jpr_w10m)%z3(:,:,1)

10.10 Top melt from sea ice

Field to couple	Top melt from sea ice
Original diagnostic variable and routine it is saved to STASH in	sf_diag%sice_mlt_htf in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3257 hourly means = top melt (on categories)
Prognostic to carry the data into the next timestep	STASH 185 using jc_topmeltn pointer and c_topmeltn variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_topmeltn(i,j,k)=d1(ja_topmeltn) ja_topmeltn is the index of the d1 array where the hourly mean of top melt is stored c_topmeltn is the prognostic which will carry top melt into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => topmeltn(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	topmeltn(i,j,k) = topmeltn(i,j,k) / ice_fract_cat_future(i,j,k) (top melt is converted from a grid box mean to a sea ice mean using ice fractions that have just been passed from NEMO)
UM TRANSDEF index and vind variable	vind_topmeltn = 8, 9,10,11,12
UM namcouple variable name	'tmltXX'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OTopMlt_catXX'
NEMO index to frcv array	jpr_topm = 32
NEMO namelist controlling factor	sn_rcv_iceflx='coupled','yes'
Temporary variable in sbccpl.F90 to manipulate data	qml_ice(:,:,:) = frcv(jpr_topm)%z3(:,:,:) * a_i_last_couple(:,:,:) / a_i(:,:,:) (qml_ice is sea ice mean top melt but converted up to grid box mean and back down to sea ice mean again using ice areas from different timesteps)
Operations applied in sbccpl.F90 after coupling	(Uses the conversion above using ice areas from different timesteps)
Final variable name in NEMO-SI3	qml_ice(:,:,:)

10.11 Surface skin temperature of sea ice

Field to couple	Surface skin temperature of sea ice
Original diagnostic variable and routine it is saved to STASH in	tstar_sice_cat is a prognostic under STASH 441 which is modified by JULES (it appears as tstar_sice_sicat in surf_couple_implicit_mod.F90)
STASH code and time meaning period	STASH 441 hourly means = Surface skin temperature of sea ice (on categories)
Prognostic to carry the data into the next timestep	STASH 195 using jc_tstar_sicen pointer and c_tstar_sicen variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_tstar_sicen(i,j,k)=d1(ja_tstar_sicen) ja_tstar_sicen is the index of the d1 array where the hourly mean of skin temperature is stored c_tstar_sicen is the prognostic which will carry skin temperature into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => tstar_sicen(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	tstar_sicen(i,j,k) = c_tstar_sicen(i,j,k) - zerodegc (skin temperature is converted from Kelvin to Celcius)
UM TRANSDEF index and vind variable	vind_tstar_sicen = 66,67,68,69,70
UM namcouple variable name	'tsfiXX'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OTsfIce_catXX'
NEMO index to frcv array	jpr_ts_ice = 57
NEMO namelist controlling factor	sn_rcv_ts_ice='ice','yes'
Temporary variable in sbccpl.F90 to manipulate data	ztsu(:,:,:) = frcv(jpr_ts_ice)%z3(:,:,:) + rt0
Operations applied in sbccpl.F90 after coupling	If temperature warmer than 0.0 oC then set to 0.0oC If temperature colder than -60.0 oC then set to -60oC Converted from Celcius to Kelvin (see row above)
Final variable name in NEMO-SI3	pist(:,:,:) = ztsu(:,:,:)

10.12 Surface conductive heat flux from sea ice

Field to couple	Surface conductive heat flux from sea ice
Original diagnostic variable and routine it is saved to STASH in	surf_ht_flux_sice in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3510 hourly means = heat flux from sea ice (on categories)
Prognostic to carry the data into the next timestep	STASH 184 using jc_fcondtopn pointer and c_fcondtopn variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_fcondtopn(i,j,k)=d1(ja_fcondtopn) ja_fcondtopn is the index of the d1 array where the hourly mean of heat flux is stored c_fcondtopn is the prognostic which will carry heat flux into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => fcondtopn(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	fcondtopn(i,j,k) = fcondtopn(i,j,k) / ice_fract_cat_future(i,j,k) (heat flux is converted from a grid box mean to a sea ice mean using ice fractions that have just been passed from NEMO)
UM TRANSDEF index and vind variable	vind_fcondtopn = 13,14,15,16,17
UM namcouple variable name	'fconXX'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OBotMlt_catXX'
NEMO index to frcv array	jpr_botm = 33
NEMO namelist controlling factor	sn_rcv_iceflx='coupled','yes'
Temporary variable in sbccpl.F90 to manipulate data	qcn_ice(:,:,:) = frcv(jpr_botm)%z3(:,:,:) * a_i_last_couple(:,:,:) / a_i(:,:,:) (qcn_ice is sea ice mean heat flux but converted up to grid box mean and back down to sea ice mean again using ice areas from different timesteps)
Operations applied in sbccpl.F90 after coupling	(Uses the conversion above using ice areas from different timesteps)
Final variable name in NEMO-SI3	qcn_ice(:,:,:)

10.13 River outflow

Here for information although the first NGMS coupled model prototype can have river routing turned off. Eventually this data will be sent straight from the river routing model (TRIP) to NEMO using the coupling method described below.

Field to couple	River outflow
Original diagnostic variable and routine it is saved to STASH in	riverout_rgrid in JULES control/um/diagnostics_riv.F90
STASH code and time meaning period	STASH 26005 instantanious when river routing code run with the same timestep as the coupling frequency = River outflow on TRIP grid (kg/s)
Prognostic to carry the data into the next timestep	STASH 198 using jc_fcondtopn pointer and c_fcondtopn variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_riverout_trip=d1(ja_riverout_trip) ja_riverout_trip is the index of the d1 array where the river outflow is stored c_riverout_trip is the prognostic which will carry river outflow into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => runoff_total(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	runoff_points(point_number(river_number),river_number) = river_outflow (Populate an array where each row is for a different river number and each column is each grid point feeding into that river) CALL gcg_rvecsumrf(..., runoff_points, runoff_total(:,1,1)) (Perform a global sum, adding up all the river outflows that contribute to each river)
UM TRANSDEF index and vind variable	vind_runoff_1d = 74
UM namcouple variable name	'runoffa'
OASIS Remapping method	BLASOLD: multiply by 1 and add 0
NEMO namcouple variable name	'ORunff1D'
NEMO index to frcv array	jpr_rnf_1d = 60
NEMO namelist controlling factor	sn_rcv_rnf='coupled1d','no'
Temporary variable in sbccpl.F90 to manipulate data	CALL cpl_rnf_1d_to_2d(frcv(jpr_rnf_1d)%z3(:,:,:)) (Data sent to cpl_rnf_1d_to_2d routine where the 1D river outflow data is redistrinuted over runoff points, making use of the river_number ancillary file)
Operations applied in sbccpl.F90 after coupling	cpl_rnf_1d_to_2d divides the river outflow (in kg s-1) by the river outflow area to get the units into kg m-2 s-1
Final variable name in NEMO-SI3	rnf(:,:)

10.14 Penetrating solar radiation into sea ice

Here for information although the first NGMS prototype will have penetrating solar turned off. This will need to be added for GC6.0 though.

Field to couple	Penetrating solar radiation into sea ice
Original diagnostic variable and routine it is saved to STASH in	penabs_rad in in radiation_control/diagnostics_rad.F90
STASH code and time meaning period	STASH 1570 hourly means = penetrating solar into sea ice (on categories)
Prognostic to carry the data into the next timestep	STASH 200 using jc_penabs_radn pointer and c_penabs_radn variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_penabs_radn=d1(ja_penabs_radn) ja_penabs_radn is the index of the d1 array where the hourly mean of penetrating solar is stored c_penabs_radn is the prognostic which will carry heat flux into the next timestep
UM pointer used to point to data to send	transient_a2o(tc)%field => penabs_radn(:,:,n:n)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	penabs_radn(i,j,k) = penabs_radn(i,j,k) / ice_fract_cat_future(i,j,k) (penetrating solar is converted from a grid box mean to a sea ice mean using ice fractions that have just been passed from NEMO)
UM TRANSDEF index and vind variable	vind_penabs_radn = 131,132,133,134,135
UM namcouple variable name	'qtrXX'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OQtr_catXX'
NEMO index to frcv array	jpr_qtr = 61
NEMO namelist controlling factor	sn_rcv_qtr='coupled','yes'
Temporary variable in sbccpl.F90 to manipulate data	zqtr_ice_top(:,:,:) = frcv(jpr_qtr)%z3(:,:,:) * a_i_last_couple(:,:,:) / a_i(:,:,:) (zqtr_ice_top is sea ice mean solar penetrating radiation but converted up to grid box mean and back down to sea ice mean again using ice areas from different timesteps)
Operations applied in sbccpl.F90 after coupling	(Uses the conversion above using ice areas from different timesteps)
Final variable name in NEMO-SI3	qtr_ice_top(:,:,:)

11 Waves to atmosphere exchange

11.1 Charnock's coefficient

Field to couple	Charnock's coefficient
From prognostic to temporary variable in w3updtmd.ftn (vn4.18) and w3agcmmd.ftn (vn7)	CALL MPI_GATHER(CHARN(IAPROC), 1, WW3_COUP_VEC, couplework, 1, WW3_COUP_VEC, OAPROC, MPI_COMM_WAVE, IERR) (vn4.18) WHERE(CHARN(1:NSEAL) /= UNDEF) TMP(1:NSEAL)=CHARN(1:NSEAL) (vn7)
Operations applied to temporary variable in w3updtmd.ftn (vn4.18) and w3agcmmd.ftn (vn7) prior to coupling	WHERE(couplework > 0.32) couplework = 0.32 (Cap value to 0.32 in vn4.18) TMP(1:NSEAL) = 0.0 (Set to zero where not defined in vn7)
WWIII index to SND_FLD array	myvar_index=1 (vn4.18) Dynamically set depending on fields actually coupled (vn7)
WWIII namelist controlling string	T in fixed-format input file ww3_shel.inp.template (vn4.18) TYPE%COUPLING%SENT = 'CHA' (vn7)
WWIII namcouple variable name	'ZCHARN' (vn4.18) 'WW3_ACHA' (vn7)
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'wave_cha'
UM TRANSDEF index and vind variable	vind_wave_charnock = 120
UM pointer used to point to recieved data	transient_w2a(tc)%field => wave_charnock(:,:)
Operations applied to recieved data	None
Assign portion of D1 array by stash code	jcharnock = si(517,Sect_No,im_index)
Point to D1 array	charnock_w => d1(jcharnock)
Copy to D1 array	charnock_w=wave_charnock

12 Atmosphere to waves exchange

12.1 Zonal wind stresses

Field to couple	Zonal wind stresses
Original diagnostic variable and routine it is saved to STASH in	taux_ssi in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3392 hourly means
Prognostic to carry the data into the next timestep	STASH 176 using jc_taux pointer and c_taux variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_taux(i,j) = d1(ja_taux+i-1+((j-1)*oasis_imt)) ja_taux is the index of the d1 array where the previous hourly mean is stored c_taux is the prognostic which will carry wind stress into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => taux_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the zonal (East-West) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_taux = 23*
UM namcouple variable name	'taux_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'WW3_UTAU'
WWIII index to RCV_FLD array	Dynamically set depending on fields actually coupled (vn7)
WWIII namelist controlling factor	TYPE%COUPLING%RECEIVED = 'TAU'
Operations applied in w3updtmd.ftn after coupling	Conversion of stress components to stress module and direction
Final variable name in WWIII	TAUA (stress module) and TAUADIR (stress direction)

* A new vind variable is probably needed for this coupling exchange, in order to properly set the coupling order when coupling the atmosphere to ocean and waves simultaneously

12.2 Meridional wind stresses

Field to couple	Meridional wind stresses
Original diagnostic variable and routine it is saved to STASH in	tauy_ssi in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3394 hourly means
Prognostic to carry the data into the next timestep	STASH 177 using jc_tauy pointer and c_tauy variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_tauy(i,j) = d1(ja_tauy+i-1+((j-1)*oasis_imt)) ja_tauy is the index of the d1 array where the previous hourly mean is stored c_tauy is the prognostic which will carry wind stress into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => tauy_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the meridional (North-South) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_tauy = 24*
UM namcouple variable name	'tauy_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'WW3_VTAU'
WWIII index to RCV_FLD array	Dynamically set depending on fields actually coupled (vn7)
WWIII namelist controlling factor	TYPE%COUPLING%RECEIVED = 'TAU'
Operations applied in w3updtmd.ftn after coupling	Conversion of stress components to stress module and direction
Final variable name in WWIII	TAUA (stress module) and TAUADIR (stress direction)

* A new vind variable is probably needed for this coupling exchange, in order to properly set the coupling order when coupling the atmosphere to ocean and waves simultaneously

12.3 Surface air density

Field to couple	Surface air density
Original diagnostic variable and routine it is saved to STASH in	rho1 in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3562 hourly means
Prognostic to carry the data into the next timestep	STASH 518 using jc_rhoa pointer and c_rho_air variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_rho_air(i,j) = d1(ja_rhoa+i-1+((j-1)*oasis_imt)) ja_rhoa is the index of the d1 array where the previous hourly mean is stored c_rho_air is the prognostic which will carry air density into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => rho_air_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the meridional (North-South) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_rhoa_w = 103
UM namcouple variable name	'rhoa_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'WW3_RHOA'
WWIII index to RCV_FLD array	Dynamically set depending on fields actually coupled (vn7)
WWIII namelist controlling factor	TYPE%COUPLING%RECEIVED = 'RHO'
Operations applied in w3updtmd.ftn after coupling	None
Final variable name in WWIII	RHOAIR

12.4 Zonal neutral 10m wind speed

Field to couple	Zonal neutral 10m wind speed
Original diagnostic variable and routine it is saved to STASH in	u10m_nb in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3365 hourly means
Prognostic to carry the data into the next timestep	STASH 515 using jc_u10 pointer and c_u10m_w variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_u10m_w(i,j) = d1(ja_u10+i-1+((j-1)*oasis_imt)) ja_u10 is the index of the d1 array where the previous hourly mean is stored c_u10m_w is the prognostic which will carry wind into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => u10m_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the zonal (East-West) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_u10m_w = 101
UM namcouple variable name	'u10m_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'WW3__U10'
WWIII index to RCV_FLD array	Dynamically set depending on fields actually coupled (vn7)
WWIII namelist controlling factor	TYPE%COUPLING%RECEIVED = 'WND'
Operations applied in w3updtmd.ftn after coupling	Conversion of wind components to wind velocity and direction
Final variable name in WWIII	UA (wind speed) and UD (wind direction)

12.5 Meridional neutral 10m wind speed

Field to couple	Meridional neutral 10m wind speed
Original diagnostic variable and routine it is saved to STASH in	v10m_nb in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3366 hourly means
Prognostic to carry the data into the next timestep	STASH 516 using jc_v10 pointer and c_v10m_w variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_v10m_w(i,j) = d1(ja_v10+i-1+((j-1)*oasis_imt)) ja_v10 is the index of the d1 array where the previous hourly mean is stored c_v10m_w is the prognostic which will carry wind into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => v10m_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the zonal (East-West) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_v10m_w = 102
UM namcouple variable name	'v10m_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'WW3__V10'
WWIII index to RCV_FLD array	Dynamically set depending on fields actually coupled (vn7)
WWIII namelist controlling factor	TYPE%COUPLING%RECEIVED = 'WND'
Operations applied in w3updtmd.ftn after coupling	Conversion of wind components to wind velocity and direction
Final variable name in WWIII	UA (wind speed) and UD (wind direction)

12.6 Zonal 10m wind speed

Field to couple	Zonal 10 wind speed
Original diagnostic variable and routine it is saved to STASH in	u10m in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3209 hourly means
Prognostic to carry the data into the next timestep	STASH 515 using jc_u10 pointer and c_u10m_w variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_u10m_w(i,j) = d1(ja_u10+i-1+((j-1)*oasis_imt)) ja_u10 is the index of the d1 array where the previous hourly mean is stored c_u10m_w is the prognostic which will carry wind into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => u10m_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the zonal (East-West) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_u10m_w = 101
UM namcouple variable name	'u10m_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'u10mwnd'
WWIII index to RCV_FLD array	myvar_index=2 (vn4.18)
WWIII namelist controlling factor	'T F T' in fixed-format input file ww3_shel.inp.template
Operations applied in w3updtmd.ftn after coupling	Conversion of wind components to wind velocity and direction
Final variable name in WWIII	UA (wind speed) and UD (wind direction)

12.7 Meridional 10m wind speed

Field to couple	Meridional 10m wind speed
Original diagnostic variable and routine it is saved to STASH in	v10m in boundary_layer/diagnostics_bl.F90
STASH code and time meaning period	STASH 3210 hourly means
Prognostic to carry the data into the next timestep	STASH 516 using jc_v10 pointer and c_v10m_w variable name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_v10m_w(i,j) = d1(ja_v10+i-1+((j-1)*oasis_imt)) ja_v10 is the index of the d1 array where the previous hourly mean is stored c_v10m_w is the prognostic which will carry wind into the next timestep
UM pointer used to point to data to send	transient_a2w(tc)%field => v10m_w(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2w.F90)	None for UM. For LFRic we should make sure that this wind stress is in the zonal (East-West) direction and not lined up with the LFRic grid.
UM TRANSDEF index and vind variable	vind_v10m_w = 102
UM namcouple variable name	'v10m_w'
OASIS Remapping method	Bilinear
WWIII namcouple variable name	'v10mwnd'
WWIII index to RCV_FLD array	myvar_index=2 (vn4.18)
WWIII namelist controlling factor	'T F T' in fixed-format input file ww3_shel.inp.template
Operations applied in w3updtmd.ftn after coupling	Conversion of wind components to wind velocity and direction
Final variable name in WWIII	UA (wind speed) and UD (wind direction)

13 UKESM specific ocean to atmosphere exchange

13.1 DMS concentration in seawater

Field to couple	DMS concentration
From prognostic to temporary variable in trcrst.F90, trcini_medusa.F90 and bio_medusa_diag_slice.F90	DMS_out_cpl(:,:) = zn_dms_srf(:,:)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	Nil
NEMO index to ssnd array	jps_bio_dms=35
NEMO namelist controlling string	sn_snd_temp%cldes='medusa'
NEMO namcouple variable name	'OBioDMS'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'dms_conc'
UM TRANSDEF index and vind variable	vind_dms_conc = 90
UM pointer used to point to recieved data	transient_o2a(tc)%field => dms_conc_in(:,:,:)
Operations applied to recieved data	Nil
Assign portion of D1 array by stash code	jdms_conc = si(132,Sect_No,im_index)
Point to D1 array	dms_conc => d1(jdms_conc)
Copy to D1 array	dms_conc=dms_conc_in

13.2 CO2 Ocean flux

Field to couple	CO2 ocean flux
From prognostic to temporary variable in trcrst.F90, trcini_medusa.F90 and bio_medusa_diag_slice.F90	CO2Flux_out_cpl(:,:) = zn_co2_flx(:,:)
Operations applied to temporary variable in sbccpl.F90 prior to coupling	Nil
NEMO index to ssnd array	jps_bio_dms=34
NEMO namelist controlling string	sn_snd_temp%cldes='medusa'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'oCO2flux'
UM TRANSDEF index and vind variable	vind_CO2flux = 91
UM pointer used to point to recieved data	transient_o2a(tc)%field => CO2flux_in(:,:,:)
Operations applied to recieved data	Nil
Assign portion of D1 array by stash code	j_co2flux = si(250,Sect_No,im_index)
Point to D1 array	co2flux => d1(j_co2flux)
Copy to D1 array	dms_conc=dms_conc_in

13.3 Ocean near surface chlorophyll

Field to couple	Ocean near surface chlorophyll
From prognostic to temporary variable in trcrst.F90, trcini_medusa.F90 and bio_medusa_fin.F90, multiplying by scaling constant (typically 0.5)	chloro_out_cpl(:,:) = zn_chl_srf(:,:) * scl_chl
Operations applied to temporary variable in sbccpl.F90 prior to coupling	Nil
NEMO index to ssnd array	jps_bio_chloro=36
NEMO namelist controlling string	sn_snd_temp%cldes='medusa'
NEMO namcouple variable name	'OBioChlo'
OASIS Remapping method	First order conservative fractional area
UM namcouple variable name	'obiochlo'
UM TRANSDEF index and vind variable	vind_chloro_conc = 94
UM pointer used to point to recieved data	transient_o2a(tc)%field => chloro_conc_in(:,:,:)
Operations applied to recieved data	Nil
Assign portion of D1 array by stash code	jchloro_sea = si(96,Sect_No,im_index)
Point to D1 array	chloro_sea => d1(jchloro_sea)
Copy to D1 array	chloro_sea=chloro_conc_in

14 UKESM specific atmosphere to ocean exchange

14.1 Total Greenland and Antarctic ice masses

Field to couple	Total Greenland ice mass and total Antarctic ice mass
Original diagnostic variable and routine it is saved to STASH in	icenorth and icesouth from snodep_tile in ice_sheet_mass.F90
STASH code and time meaning period	240, no time meaning, instaneous value.
Prognostic to carry the data into the next timestep	STASH 240 prognostic using jsnodep_tile pointer and snodep_tile name.
In oasis_updatecpl.F90 copy the data into the prognostic	No copy necessary, standard UM prognostic.
UM pointer used to point to data to send	transient_a2o(tc)%field => icenorth(:,:,:), transient_a2o(tc)%field => icesouth(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	None for UM.
UM TRANSDEF index and vind variable	vind_icesheet_n = 72, vind_icesheet_s = 73
UM namcouple variable name	'icenorth', 'icesouth'
OASIS Remapping method	Bilinear
NEMO namcouple variable name	'OGrnmass', 'OAntmass'
NEMO index to frcv array	jpr_grnm = 44, jpr_antm = 45
NEMO namelist controlling factor	sn_rcv_grnm='coupled','no',”,”,”' and sn_rcv_antm='coupled','no',”,”,”'
Operations applied in sbccpl.F90 after coupling	If 2d field employed: Global sum and average over ocean points to obtain single value. If 0d field employed: Nil.
Final variable name in NEMO-SI3	greenland_icesheet_mass, antarctica_icesheet_mass

14.2 Partial CO2 pressure

Field to couple	Partial CO2 pressure
Original diagnostic variable and routine it is saved to STASH in	co2 or co2_3D. Prognostic D1 variable, conditionally updated in numerous locations.
STASH code and time meaning period	Interactive CO2: 252, Time mean calculated in accordance w/ coupling frequency. Static CO2: No STASH code or time meaning.
Prognostic to carry the data into the next timestep	Interactive CO2: STASH 196 using jc_surf_co2 pointer and c_surf_CO2 name.
In oasis_updatecpl.F90 copy the data into the prognostic	Interactive CO2: c_surf_CO2(i,j) indexed by ja_CO2 from d1. Static CO2: Nil.
UM pointer used to point to data to send	transient_a2o(tc)%field => pCO2_out(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	Nil.
UM TRANSDEF index and vind variable	vind_pCO2 = 92
UM namcouple variable name	'atmpco2'
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	'OATMPCO2'
NEMO index to frcv array	jpr_atm_pco2 = 46
NEMO namelist controlling factor	sn_rcv_atm_pco2 = 'medusa','no','','',''
Operations applied in sbccpl.F90 after coupling	Nil. Moved to 'f_xco2a' in trcbio_medusa.F90
Final variable name in NEMO-SI3	PCO2a_in_cpl(:,:) = frcv(jpr_atm_pco2)%z3(:,:,1)

14.3 Dust deposition flux

Field to couple	Dust deposition flux
Original diagnostic variable and routine it is saved to STASH in	tot_dust_dep_flux in diagnostics_bl.F90
STASH code and time meaning period	3440 , Time mean calculated in accordance w/ coupling frequency
Prognostic to carry the data into the next timestep	STASH 197 using jc_dust_dep pointer and c_dust_dep name.
In oasis_updatecpl.F90 copy the data into the prognostic	c_dust_dep(i,j)=d1(ja_dust_dep+i-1+((j-1)*oasis_imt))
UM pointer used to point to data to send	transient_a2o(tc)%field => dust_dep_out(:,:,:)
Operations applied to prognostic prior to coupling (in oasis3_puta2o.F90)	Nil.
UM TRANSDEF index and vind variable	vind_dust_dep = 93
UM namcouple variable name	’atmdust’
OASIS Remapping method	First order conservative destination area
NEMO namcouple variable name	’OATMDUST’
NEMO index to frcv array	jpr_atm_dust = 47
NEMO namelist controlling factor	sn_rcv_atm_dust=’medusa’,’no’,”,”,”’
Operations applied in sbccpl.F90 after coupling	Nil. Moved to ’dust’ in trcsed_medusa.F90
Final variable name in NEMO-SI3	Dust_in_cpl(:,:) = frcv(jpr_atm_dust)%z3(:,:,1)