

OASIS Dedicated Support
6th annual summary
E. Maisonnave
TR/CMGC/22/139

Abstract

The 2022 off-peak pandemics period was targeted for resuming the OASIS dedicated support program. Three new supports were organised, one of them remotely (DWD), while the two others were given on site (Meteo-France, SMHI). Both technological and methodological problems were addressed and solved. The porting of the ICON-NEMO coupled model on a vector machine (NEC SX-Aurora TSUBASA) emphasises the continuous potential of our library to deal with hardware specialisation and heterogeneity. The capability of the new OASIS Python API (pyOASIS) was established during the parallelisation task of a new Python based toy model with aim to efficiently produce SCRIP interpolation weights and addresses files on a pre-processing phase. Easily configurable, this tool has the potential to serve a broader community. From a methodological point of view, interpolation parameters were optimised (coast-line effect on both ICON-NEMO and EC-Earth4 coupled system) and a new interface were design in the wave model MF-WAM

Table of Contents

Introduction.....	5
Remote mission #20 : German Meteorological Service, DWD, Offenbach (Germany).....	6
1. Code porting.....	8
1.1 OASIS.....	8
1.2 XIOS.....	8
1.3 ICON.....	8
2. Coupling enhancements.....	9
2.1 ICON interface.....	9
2.2 Interpolations.....	12
3. Performance.....	15
3.1 NEMO vectorisation.....	15
3.2 Coupling cost.....	16
Mission #21 : Météo-France, PREVIMER, Toulouse (France).....	19
Mission #23 : SMHI, Norrköping (Sweden).....	22
1. Interpolation weights production with Python.....	23
2. EC-Earth coupling improvements.....	27
2.1. Land sea-mask mismatch.....	27
2.2. Ice temperature interpolation.....	32
2.3. Runoff interpolation.....	32
2.4 Load balancing.....	33
3. OASIS repository size reduction.....	34
4. Conclusion.....	35
Community impact.....	36
Bibliography.....	37
Appendix.....	39
Costs/Sustainability.....	39

Introduction

The 2022 off-peak pandemics period was targeted for a resuming of the OASIS dedicated support program. The selection criteria were adjusted to make possible and relevant a remote collaboration with groups. In that purpose, simple implementations, preferentially requiring modifications of the OASIS library itself, were favoured. Three new dedicated supports (out of the 5 candidates and excluding the support provided within the ESIWACE project) were organised. One of them was achieved remotely (DWD), while the two others can be done on site (Meteo-France, SMHI). The task details can be found in this document, following the chronological order.

Remote mission #20 : German Meteorological Service, DWD, Offenbach (Germany)

March/April, 2022

Main Goal: Set up of an ICON-NEMO regional model on a vector supercomputer

Summary

The NEMO-ICON coupled model, including ICON internal and XIOS external I/O servers, is ported on the new DWD vector machine (NEC SX-Aurora TSUBASA). The ICON I/O processes and the NEMO IO servers are pinned on the scalar hosts of the machine, while the computing processes are running on the vector cores. Interpolations are parametrised to better conserve fluxes and minimise errors, including along the coast line, for which the definition is modified in the ICON interface. A first performance evaluation, performed with the OASIS3-MCT v5 library, confirms the modest extra-cost of our coupled interface

ICON-NWP limited area mode	NEMO Baltic
ICON 2.6.4, derived for Baltic sea coupling from ICON_OMCI_v1.0.0 ¹	NEMO 4.0, see [2] for reference
12 km, 231,660 triangular cells, 60 vertical levels	3.7 km, 902x777, 56 vertical levels
For reference description of the coupled version see [1]	For reference description of the NEMO grid used in this configuration (extended to Southern Iceland) see [3]

Supercomputer:

“rcl”, Routine Cluster, TWL Kom, Ludwigshafen, Germany

27,136 cores - Vector Engine Type 10AE 8C SX-Aurora TSUBASA - Memory per VE: 48 Gb, 8 VE per node, 4 cores per VE + 1 host 24-core AMD Rome, 2.8 GHz

<https://www.top500.org/system/179928/>

When this support started, a first version of the new ICON atmosphere mode in climate

1 https://zenodo.org/record/5833118#.Yd_f-d8o8aE

limited-area mode, originally developed for operational weather forecasts, was available at the Deutsches Klimarechenzentrum (DKRZ) scalar supercomputer. This version, made available by Helmholtz-Zentrum HEREON, includes an OASIS interface, which enables the coupling in a regional version of the NEMO ocean model. The first goal of this support was to make the NEMO-ICON coupled model running on the new DWD vector machine (NEC SX-Aurora TSUBASA). Several problems were expected, particularly the compatibility of each of the coupled system components with the vector machine. In particular, the ICON and NEMO I/O servers, ideally running on the scalar hosts of the machine, had to be tested and consistently parametrised.

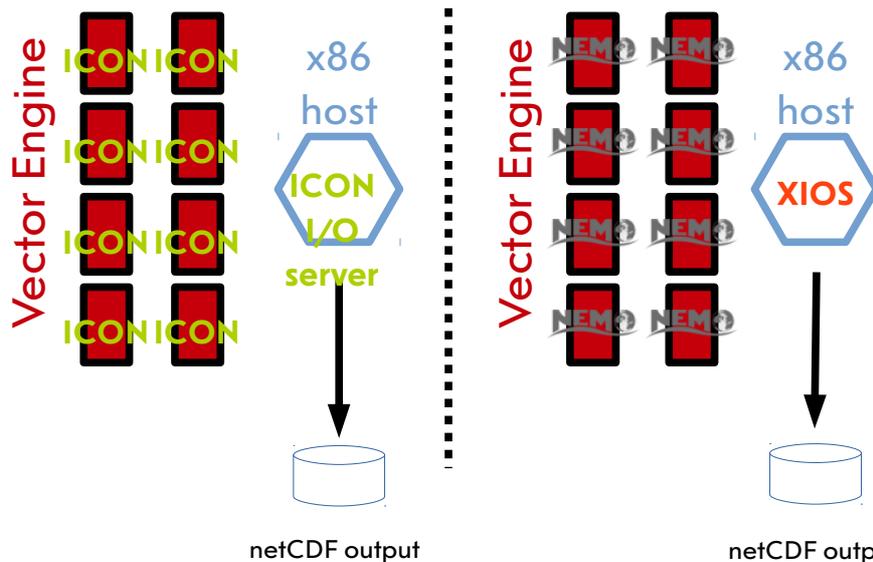


Figure 1: Schematic mapping of the coupled system processes (ICON+ICON IO server, right & NEMO and XIOS server, left) onto the computing elements (Vector engines, red boxes & Scalar host, blue hexagons) of a node of the NEC SX Aurora TSUBASA DWD supercomputer

Actually, the NEC SX-Aurora TSUBASA computing nodes are hybrid: At DWD, each node has 8 vector engines (VE), of 8 vector cores each, and 4 AMD EPYC scalar hosts (VH). The coupled system complexity is partly due to the fact that, for computing performance reasons, the I/O server processes should be pinned onto the scalar hosts, while ICON and NEMO computing processes must run on the vector engines. This makes necessary an appropriate handling by the OASIS coupler, which is supposed to provide the MPI communicators needed between I/O clients and servers.

In a second step, the current interface has to be enhanced to take into account rising coupling issues such as flux conservation or land sea mask mismatch. In a final step, the system computing performance has to be checked.

1. Code porting

Since the NEMO 4.0 code, including the OASIS interface, was already available at DWD, the last three components and libraries to be ported were ICON, OASIS and XIOS.

1.1 OASIS

Our particular hardware structure requires that OASIS is compiled with both vector (NEC) and scalar (GNU) compilers. This exercise was primarily done, e.g. at CERFACS [4], and does not require any code modification. The two libraries are ready to be used in both ICON & NEMO computing processes (vector) and ICON & XIOS I/O server processes (scalar). Since the two libraries rely on the same MPI version, vector communications or hybrid scalar-vector communications will be performed by the same set of routines, while effective exchanges will occur on different physical networks.

1.2 XIOS

Compiling XIOS on these new NEC machines requires a special care, since a pre-installed external library ("boost") was required. In contrast to our previous attempt [5], the library was this time made available by the NEC hotline and machine administrators.

As for OASIS, the XIOS compiling requires two compilers: the vector one to produce the client library to be linked with the NEMO executable, and the scalar one to produce the XIOS server. Notice that this is the scalar OASIS library that must be used to produce the XIOS server executable, while the OASIS vector library will be used to provide the FORTRAN modules necessary for XIOS vector library compiling (client part).

The XIOS library version often varies in conjunction with its NEMO client version. The ocean model in use at DWD requires an XIOS version from branch 2.5, with a later revision than 1587. As described in [6], a new parameter² avoids a deadlock between OASIS and XIOS definition phases. But, despite this modification, the library suddenly stops in a badly defined barrier in both client and server code. A work-around was implemented (the comment of the two inadequate MPI collective operations) and the problem reported to the XIOS developers.

For computing performance described in Section 3, notice that the "prod" compiling option was set and the second level of server enabled, which leads to the online production of the single-file outputs, avoiding the costly horizontal subdomain gathering in a post-processing phase.

1.3 ICON

Similar operations (dual compiling with the appropriate OASIS library) were necessary to use the ICON computing processes on the vector part of the machine and the ICON I/O processes on the scalar part. Unlike NEMO, ICON comes with its own I/O client/server library. Basically, some ICON processes are internally dedicated to the I/O. On NEC, the

² `<variable id="call_oasis_enddef" type="bool">>false</variable>` in `iodef.xml` file

sequence of MPI processes for computing or I/O, internally defined, has to be taken into account in the `mpirun` launching command, since the I/O server processes must be launched on the scalar hosts. In addition, for this vector machine only, the MPI processes sequence is modified ("`shift`" option) to let the first process run on host.

This last particularity was not taken into account in our current ICON interface version (designed for a scalar machine) and the extra scalar process was blocking our coupled simulation. We modified the OASIS interface³ to be able to :

- exclude this first process of the coupling, by using a different MPI communicator of ICON⁴ and
- let the first VH process call `oasis_set_coupl_comm` with `MPI_COMM_NULL` (as the I/O processes do). To do so, a remnant option⁵ was re-organised to prevent this first "`shift`" process to call the OASIS partition and variable declarations (and any other `oasis_get/put` operation).

2. Coupling enhancements

2.1 ICON interface

2.1.1 Coupling sequence

The existing ICON interface was derived from the former COSMO interface, previously built to include the regional atmosphere into a comprehensive Regional Earth System model (RESM). Its Unified Interface [7] is still relevant but an updated checking of the current implementation in its full RESM configuration is missing. In order to simplify the interface, we decided to remove every code portion unrelated to the actual regional ocean coupling, the only one that can be currently tested on this machine and with this version of ICON.

Following the same philosophy, we excluded from the code the various instructions supposed to ensure the coupling of ICON nests. This operation hugely simplified the timing handling of the interface. The OASIS receiving and sending subroutines require a time stamp argument, to enable the coupling exchange at the correct moment, but also a possible time average at every coupling time step. In our simplified version, the model time step is sent as an argument of the OASIS exchange subroutines, but it is reduced by one model time step, to allow for the concurrent use of both ocean and atmosphere models or OASIS restart files, respectively.

The calling sequence of the `oasis_put` and `oasis_get` subroutines is also modified. Originally, both routines were called together, at a location from within the time loop. In most of the community OASIS interface implementations, the received coupling fields are made available at the beginning of the time loop (usually in replacement of the boundary condition

³ See on `rcl` supercomputer :
`~extemais/toymodel/icon*/src/atm_phy_nwp/cpl_oas_interface.f90`

⁴ `p_comm_work_only` instead of `p_comm_work`

⁵ Originally, in the legacy COSMO interface, the `lpe_cpl` option was set to exclude from the coupling every process exclusively composed by masked grid points

reading). Symmetrically, the outgoing fields are sent as soon as they are available or, for computing performance reasons, once in a row at the end of the time loop. This is the stencil we propose in our implementation, although we prefer to keep the receiving operation at its original location. To prevent the use of non-updated arrays at time step zero (and during initialisation phase), the first call of `oasis_get` is placed to a location shortly after the beginning of the initialisation phase. Symmetrically, its first call in the time loop (time step 1) is disabled. This strategy ensures that boundary conditions (coming from the previous simulation chunk and saved in the OASIS restart file) are available from the beginning. It also prevents any extra call or missing call at the last time step.

2.1.2 Land/sea mask

The OASIS interface is supposed to update boundary conditions at the correct time step but it also has to provide this information at the correct geographical position. To do so, it is mandatory to communicate to the coupling library the geographical position of each grid point (latitude/longitude) but also whether these grid points have to be included or not in the interpolation (mask). It is extremely important that OASIS receives the exact land/sea mask as it is used in the model.

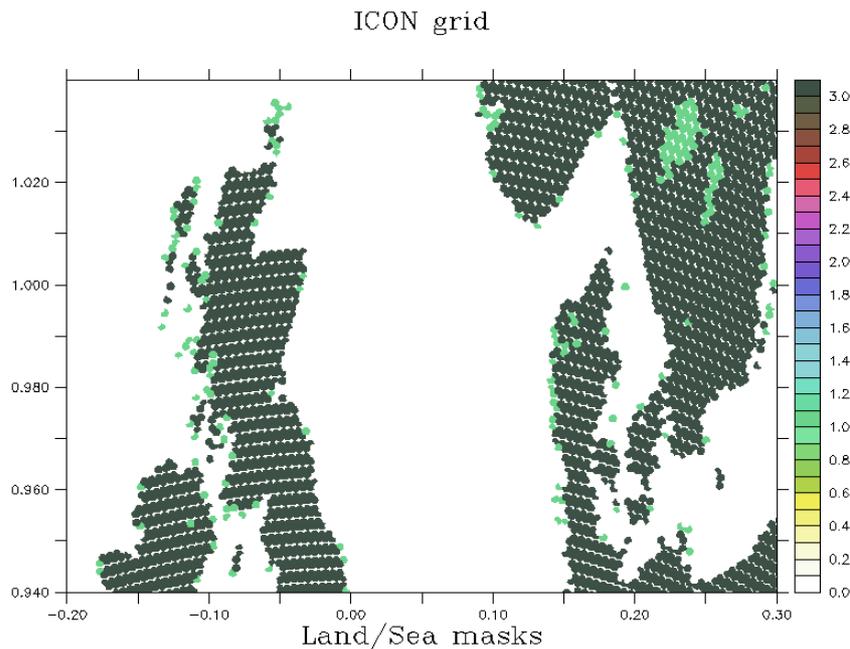


Figure 2: Addition of internal ICON and OASIS land/sea masks. The values are 0: sea for both, 1: sea for internal ICON only, 2: land for OASIS only, 3: land for both internal ICON & OASIS

In the original setup, the OASIS auxiliary file which contains the ICON land/sea mask information⁶ is slightly different from the inner variable that the interface seems to consider as

⁶ `nico.msk` variable from `masks.nc`

the true model land/sea mask. Its value is read again in a separate file⁷ and mixed with the true model variable⁸. This mask is used to selectively update the ocean grid points of the model variable with coupled values. This multiplication of variables complicates the identification of the actual grid points which receive coupled values. In addition, the usual OASIS diagnostics about interpolation or transformation quality (e.g. netCDF visualisation of the coupled fields) are less relevant, knowing that additional and hidden modification will be performed before receiving the actual coupled field in the model.

To clarify this procedure of coupled array updating, we propose first to identify all the ICON ocean model grid points (whole grid) that could be involved in the coupling, then to reduce these points to the interpolated NEMO domain. This information is saved in one single file (OASIS auxiliary file `masks.nc`) and used both by ICON and OASIS.

Figure 2 shows the land/sea mask of two variables discretised on the ICON grids. In black, the masked (land) grid points as they are defined in the model are given. Only unmasked grid points as defined by this variable will be updated by the values coming from OASIS. In green, the masked grid points that are added to the black mask grid points in the original OASIS auxiliary file are given. This mask (black+green) is used by OASIS to perform interpolations from/to the ICON grid to/from the NEMO grid. We deduce from this graphic that:

- it is not the same threshold that discriminates land from ocean grid points in OASIS auxiliary file and ICON
- a set of ICON grid points (in green) does not receive information via the coupler. We guess that ocean flux computations are computed there with values coming from a climatology. This is exactly what is expected on lake grid points, but what must strictly be avoided on coastal grid points⁹.

We propose to rely on the model variables¹⁰ to build a coherent new OASIS mask that excludes lakes from the coupling but keeps as much coastal grid points as possible. Of course, a threshold must be taken (0.5 value) to avoid defining too much ocean grid points and create an artificial coast line mismatch between NEMO and ICON grids. This necessarily implies that grid points including less than 0.5 ocean ratio in the tile definition will calculate ocean fluxes with climatological values. But these fluxes will not be used in the ICON to NEMO coupling and we hope that their effects on the surface physics and dynamics will be small. A more stringent solution would be to modify the atmosphere model tile definition in accordance to the OASIS mask (as implemented in [8]). But the proper evaluation of their side effects is too time consuming to be tackled during this support.

7 `masks_nico.msk` file fills the `mask_cp1` variable

8 `ext_data%atm%llsm_atm_c`. If this variable is equal to 1, `mask_cp1` is forced to 1 (land value). This internal variable comes from the binarisation at 0.5 threshold of the fraction variable `FR_LAND` of the input file `extpar_iconR13B5_DOM01.nc`. The mask covers the whole ICON domain, and not only the domain interpolated from the NEMO domain.

9 Notice that the ICON grid points located outside the NEMO domain boundary should also be excluded from the interpolation. This can be done in a first step, by interpolating the NEMO mask to the ICON grid, using the same interpolation than the one used on-line

10 `FR_LAND` and `FR_LAKE` in `external_parameter_icon_europe011_DOM01_tiles.nc`

2.2 Interpolations

The interpolation usually has a marginal impact to the solution, but the horizontal resolution ratio between source and target grids and coast line mismatch could have both global and local effects. In addition, the question of interpolation conservativeness is still relevant for long-term climate simulations, for global as for regional configurations.

	Avg resolution (Km)	Variation (%)	Resolution ratio
<i>ICON</i>	12.2	6	2.7 to 4.4
<i>NEMO</i>	3.7	30	

Table 1: Quantities related to horizontal resolution of ICON and NEMO models over non masked grid point of the considered area (Baltic + North Atlantic)

We suppose that the original ICON-Atmosphere/ICON-Ocean interpolation (made by YAC [9]) conserves, at least at first order, all quantities between grids. In our case, we have to deal with non-coincident meshes, which has a particularly bad side effect along the coast line. Even though a locally conservative interpolation is available in OASIS (SCRIP 1st order conservative), its use in this particular case would lead to :

- irregular values due to land/sea mask mismatch close to the coastline
- patchy fields in the Northern part of the domain, due to the large difference of horizontal resolution between grids (see Table 1)

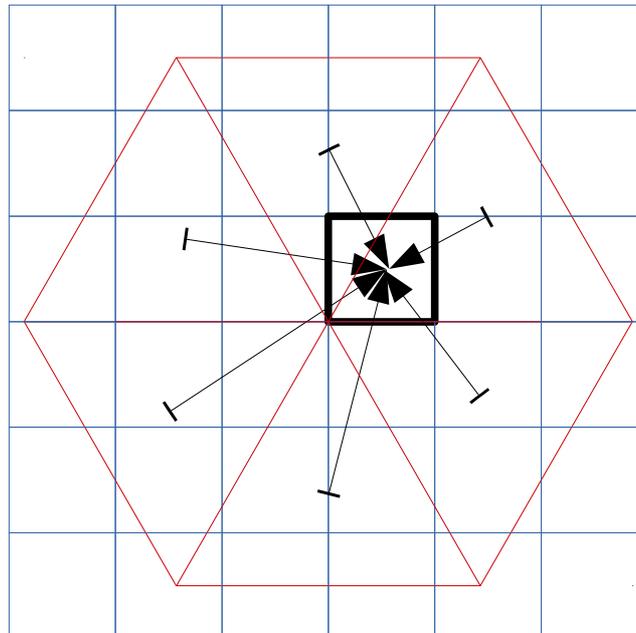


Figure 3: ICON (red) to NEMO (blue) nearest neighbours interpolation (nn=6)

A simple solution, already set up in regional configurations (e.g. in [7]), consists of associating a locally non-conservative interpolation (e.g. a nearest neighbour (NN), weighted by inverse

distance to the source grid point) to a global conservation. The capacity to provide smoothed values along the shore is the main advantage of this solution. The total conservation is ensured with a normalisation, taking into account the total areas of the non-masked grid points of the two grids (which have a difference of areas equal to 0.35%).

2.1.1 ICON to NEMO interpolation

The resolution ratio guides the choice of the interpolation number of neighbours. Too many neighbours lead to overly smoothed fields, but a too small number creates artificial gradients by sub-sampling the source grid information. As in [10], we compare the grid geometries to deduce a first guess (see Fig 3.), and use a small error estimator by interpolating an analytical field to the NEMO meshes.

#neighbours	4	6	7	9
Avg error (%)	0.77	0.62	0.40	0.40

Table 2: SCRIP interpolation error (by comparison of an analytical field on the target grid) for a varying number of neighbours, from NEMO to ICON grids

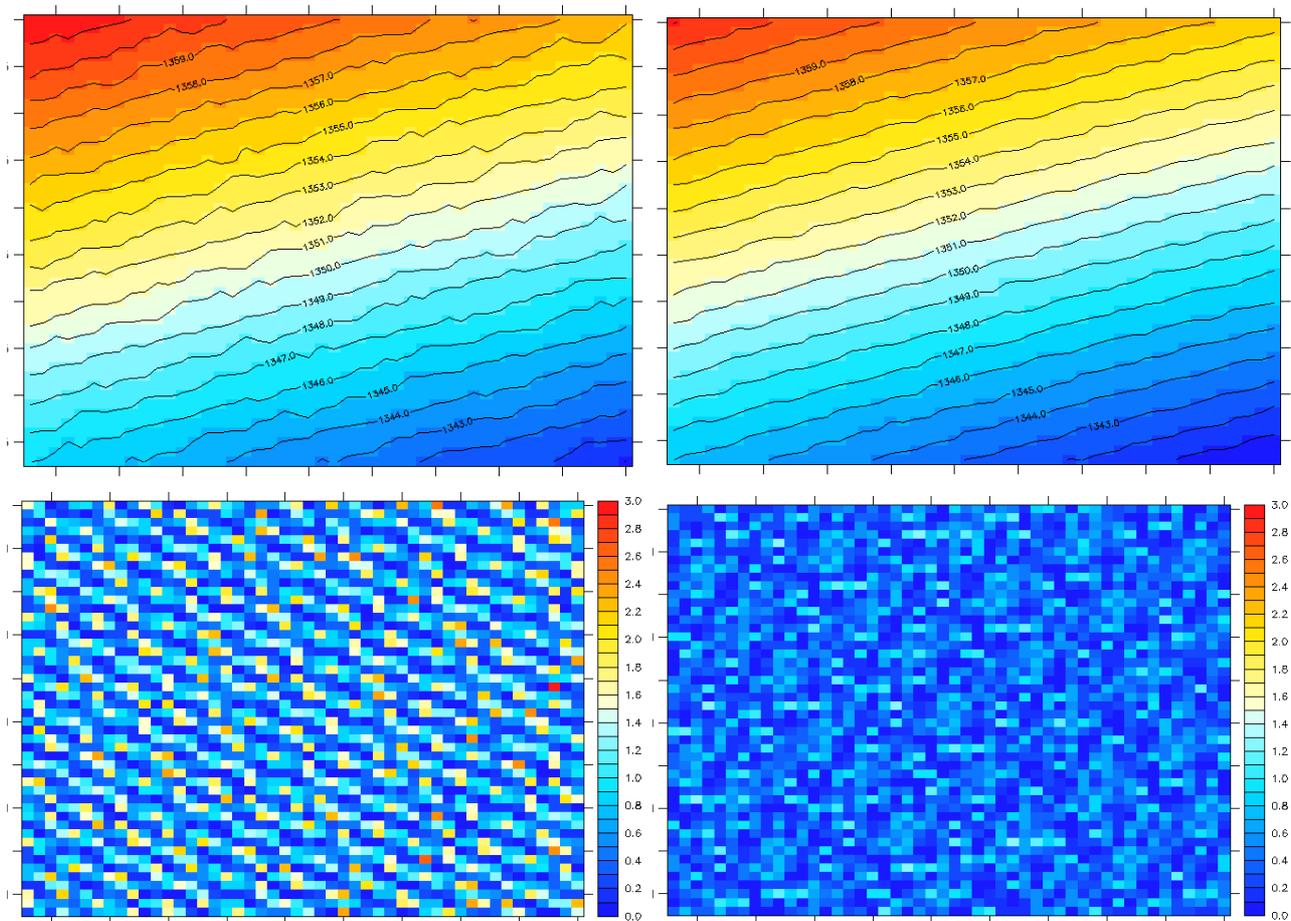


Figure 4: Analytical field interpolated from ICON to NEMO grid with 4 (left) or 7 (right) nearest neighbours. Raw fields are above, differences (%) with the analytical field calculated on the target grid below. Zoom over an Atlantic ocean area without land grid points

While NEMO uses rectangular meshes, triangles are the chosen geometry for the ICON icosahedral model. An optimal number of 6 neighbours is chosen to start testing the interpolation accuracy with our interpolator tool. After testing (see Table 2 for average values and Fig. 4 for a visual evaluation), the value 7 is selected, as it gives one of the smallest errors and optimises the interpolation performance.

A further look to the actual coupled fields transferred to the ICON model such as the ones related to ice (see below) convinces us that there is probably no need for further smoothing of the interpolated fields with an additional Gaussian function.

2.2.2 ICON to NEMO global conservation

The OASIS global conservation requires that mesh areas are provided to the coupling library. A set of NCO [11] commands is defined to deduce them from the `meshmask` (NEMO) and the initial domain definition¹¹ (ICON). At the same time, the coordinates and mask information are also deduced from these files. In a second step, interpolation weights & addresses can be produced on-line by the OASIS library during initialisation phase.

The OASIS global conservation can be performed after the interpolation. Global sums of the initial and interpolated fields are compared and a correction added to the interpolated field in order to conserve the global sum. The same conservation can be added to every non masked interpolated values. Optionally, the interpolated field can be multiplied in such a way that its global sum reaches the source sum. It is this last option that is used in our case, to avoid creating artificial non-zero values where they could lead to ICON numerical instabilities (negative precipitations, heat fluxes over ice in regions not covered by ice, etc). The global conservation is not applied to the surface pressure coupling fields.

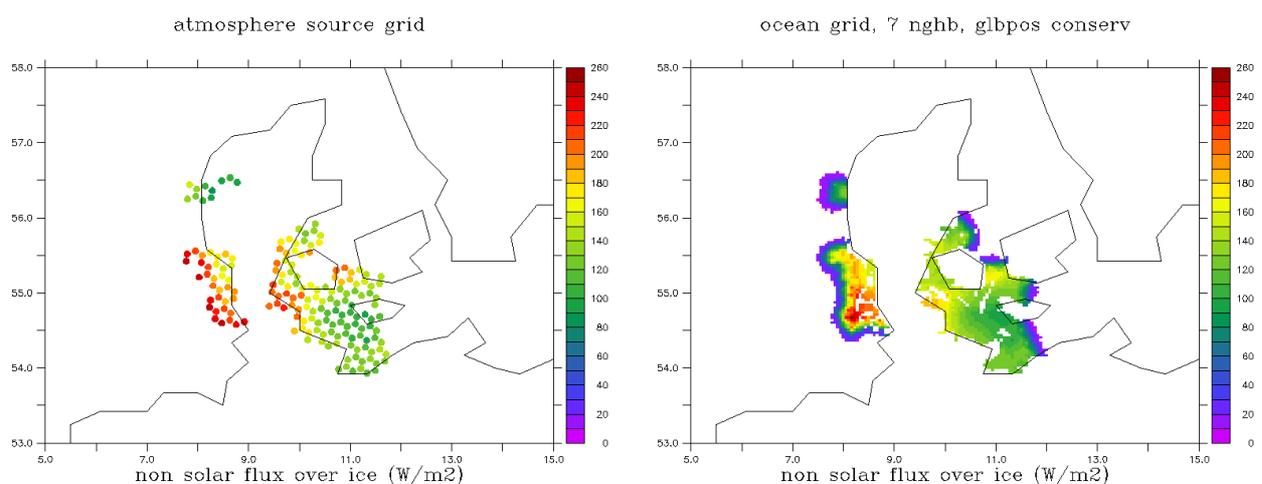


Figure 5: Non-solar heat flux over ice (one coupling time step) before (left) and after interpolation + global conservation, zoom above the Denmark region, arbitrary coupling time step

¹¹ `iconR13B5_DOM01.nc` file

The global correction for every coupling, at every time step, is checked in OASIS log files. Less than 1% correction is observed for all the ocean-related fields. For non-solar heat flux and evaporation over ice, larger corrections can be seen (as shown in Fig. 5).

The relatively high difference can be explained by the spread of non-zero values induced by the ICON to NEMO 7-neighbour interpolation. In addition, it seems that the ICON higher number of ocean grid points close to the coast line eliminates strong values close to the shore, that are redistributed off-shore with the global conservation (limited to grid points with non-zero values).

Before trying to address this specific interpolation issue, we suggest to better understand the physics of the coupling between NEMO-SI³ and ICON. We can assume that interpolations of ice fluxes are often error-prone in Marginal Ice Zones (see the full study in link with this issue and the possible solution provided by an exchange grid in [12]).

To complete the overview of the conservation issue in this ocean-atmosphere coupling, we note that no second order conservation of the wind stresses is performed, since the SCRIP 2nd order conservative interpolation would require additional quantities (spatial derivatives) from ICON.

2.1.3 NEMO to ICON interpolation

Conservation issues are less important when performing the ocean (or sea ice) to atmosphere interpolation. For the same reasons than on the other way round, a simple bilinear interpolation is tested first. But this interpolation only requires 4 neighbours, which sub-samples the source grid point information. According to Table 1, the ratio between the number of non-masked grid points of ICON and NEMO is close to 3, but a larger number must be used to better take into account all intersected meshes.

2.1.4 Runoff interpolation

At the dedicated support time, the hydrology model that should finally be included in the coupled system was not available. An additional work can be planned to better understand how runoff values could be calculated and provided to the ocean model instead of the NEMO climatological values. For information, we mention the availability of a dedicated interpolation in the 5th version of our OASIS3-MCT coupling library, as defined in [13], implemented in [14] and tested in [15].

3. Performance

3.1 NEMO vectorisation

The NEMO ocean model is not part of the ICON NWP suite and for that reason, its efficiency on the new DWD vector supercomputer is not certified. However, preliminary work from

DWD and partner agencies has ensured that the present NEMO-SI3 version could be used. In addition, performance tests [4,5] were performed with the NEMO BENCH configuration on similar platforms.

Our first attempts to get realistic results on the NEC machine show that the legacy `key_vectopt_loop` pre-processing key, although adding some marginal performance gain, was generating a segmentation fault at runtime. Obviously, this pre-processing key, originally developed to perform computations on halos, thus increasing the vector length, has not been maintained for years and leads, in several places, to array overflow.

The array dimensions and the land/sea mask position in the North-Atlantic/Baltic domain allow to increase (until 2) the decomposition in the X direction, a limitation that was the main performance bottleneck in our previous study. This allows to keep a good average vector length for computations (AVL=200), but also to eliminate 14 out of the 56 subdomains (land only) in our first guess decomposition. Further tests could be lead to try to improve the model speed (sea ice model calling frequency, ocean-atmosphere coupling frequency or even ocean model time step).

To prepare future configurations, the new NEMO 4.2 version had also be compiled on the NEC platform. The porting was as easy as it was for 4.0. Similarly, the vectorisation has to be revised, but the out-to-the-box performances look rather comparable to the 4.0.

3.2 Coupling cost

The usual issue of adding a module to an existing configuration (here NEMO added to ICON) comes with a significant slow down of the simulation. As already emphasised [16], this does not mean that coupler interpolations and communications are the culprits. Usually, most of the system slow down comes from the load imbalance between the coupled system components.

With the current version used in this system (OASIS3-MCT v4), a post-processing tool can be applied to traces that the coupling library can produce on-line. This diagnostic is performed, but the best load balancing found still shows a significant cost in ICON-based interpolations (20%). A simple parameter¹² is changed to let OASIS perform all the interpolation on the NEMO partitions, mapped on more computing nodes (5) than ICON (3).

A residual load imbalance on both sides was detected by our load balancing diagnostic. To better understand its origin, the original 4.0 OASIS library version was substituted by the later version 5.0 (in all our components, including I/O servers). During this switch, a new bug in the mapping library MCT was identified¹³. Another bug was found in load balancing computations¹⁴. After correction, it was possible to produce a new load balancing diagnostic,

12 With the `$MAPLOC` option of the `MAPPING` operation set to `dst`, interpolation is performed by the destination model

13 An MPI communication is required by MCT even if the pool of involved process as defined in a previously defined `mpi_group_translate_ranks` command is empty. Ticket #2817

14 The internal ICON I/O server, not involved in the coupling, thus not calling the load balancing computation routine, is wrongly involved in an `MPI_Barrier` common to all coupled processes. Ticket #2816

which simplifies the identification of the load balancing origin.

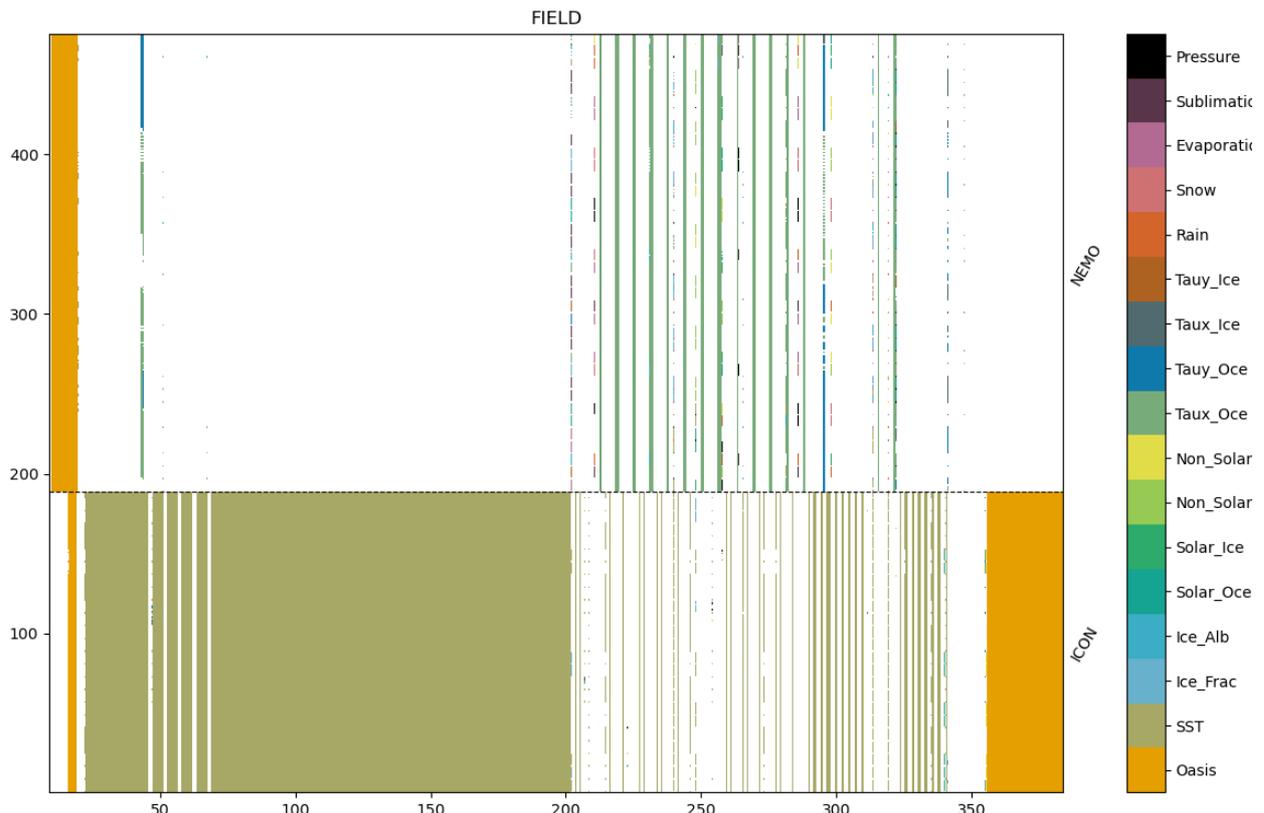


Figure 6: OASIS-related events timeline, for an ICON/NEMO simulation on NEC SX-Aurora TSUBASA. Only processes of computing executables (ICON below, NEMO above) are displayed. White areas represent model computations. The color legend shows the name of the coupling fields corresponding to the periods when they are interpolated/exchanged. OASIS stands for coupling initialisation/termination phases

This new diagnostic consists in the visualisation [17] of the full timeline [18] of all coupling related events during the simulation, in all processes involved in the coupling (excluding XIOS and ICON I/O servers). This timeline is provided for a short lasting test in Fig. 6. A quick look to the beginning of the simulation shows large boxes associated to period were ICON is waiting for the SST. This corresponds to the moment when NEMO is waiting for the completion of the initialisation phase of a third component, the XIOS I/O server. A zoom to the relevant part of the timeline (non-initialisation phase) is shown on Fig. 7.

The zoom reveals the irregular duration of the ICON model time step, which necessarily leads to extra waiting periods (blue) on both ICON and NEMO models. Incidentally, this extra waiting time can be identified as a time spent for interpolation. A more precise zoom (not shown) clarifies the origin of this confusion: this is the model jitter which simply delays the starting of the first interpolation and wrongly identifies new waiting time (due to model imbalance) to an OASIS interpolation operation.

In order to reduce coupling time to its minimum, this kind of analysis would need to be reproduced when the final configuration of the regional ICON-NEMO coupled system will be ready to produce tuned simulations.

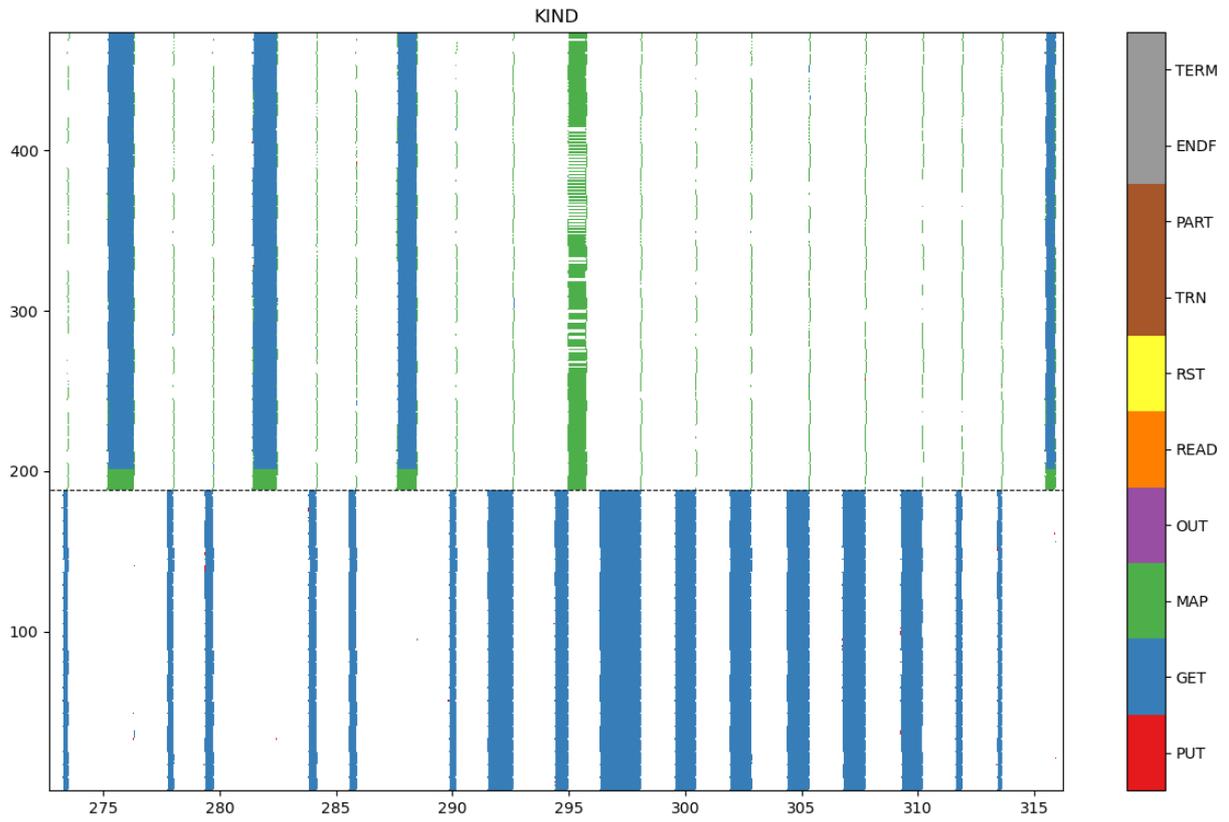


Figure 7: same timeline than Fig. 6, but zoomed during the 274s-316s time period. Color legend shows the kind of coupling operation performed

Mission #21 : Météo-France, PREVIMER, Toulouse (France)

March/April, 2022

Main Goal: Prepare the OASIS interfaces of a future atmosphere-ocean-wave (AROME-NEMO-MFWAM) coupled model

Summary

A support is given to the PREVIMER R&D team to ensure the efficient exchange of OASIS coupling fields between the NEMO ocean and MFWAM wave models. After a training period, the user was able to develop its own toy models, which were gradually replaced by the final models. Small standard adaptations (MPI internal communicator) were necessary to deliver a fully functional NEMO-MFWAM system

NEMO global ocean	MF-WAM
NEMO 4.2, see [2] for reference	Meteo France version of the WAM model
ORCA2, 31 level, reference configuration	
This version is used for technical setup but higher resolution (ORCA025) will be used for production	50 km, 116498 cells (Gaussian grid)

Supercomputer:

“belenos”, Météo-France, Toulouse, France

294,912 cores - AMD EPYC 7742 64C, 2.25GHz, 128 cores and 256Gb per node

<https://www.top500.org/system/179853/>

The WAM wave model has to be included in the Météo-France operational model, jointly with the NEMO ocean model. Our support focuses on the ocean-wave modelling, in collaboration with the ocean prevision team of the french national weather forecast service. On a slightly different organisation than in our previous OASIS dedicated supports, the main part of the interface implementation was set up by the host team (Alice Delphiné), following a period of OASIS training. The main achievements described below are the results of Alice's developments, lead with the help of the OASIS support team.

Starting from the existing uncoupled wave model developed by the WAM consortium and adapted for the use of ECMWF and Météo-France forecasts, a new interface is built to ensure the surface variable exchange with the NEMO ocean model. In that purpose, the existing interface implementation, built at HEREON Rostock on the WAM Cycle 4.5.4, is carefully checked, mainly to try to understand how to ideally deal with the horizontal decomposition between subdomains (mapped on MPI processes). A 1D "apple" decomposition, that best fits with the WAM Gaussian grid requirements, is chosen.

A first validation of the WAM-OASIS interface implementation is performed adding an ocean 'toy' model to the coupled system. The two way exchange of a single constant array (not used by the models) makes possible the validation of the exchange (subdomain decomposition, interpolation, timing).

This testing configuration allows to identify and fix several issues. In particular, the `MPI_COMM_WORLD` communicator for internal parallelisation of MFWAM has to be replaced by a local communicator, provided at the simulation start by the `oasis_get_localcomm` routine. In particular, a useless function¹⁵ has to be commented out, to be able to launch the WAM executable at any position in the `mpirun MPMD` command.

The next step consists in finding the right position of the send/receive functions of OASIS in the WAM code. Usually, the boundary conditions are requested at the beginning of the time loop and the variables that will be used by the other coupled components are available after the diagnostic computations. A `LAG`¹⁶ is defined to be able to read (in an OASIS restart file) the boundary conditions calculated by NEMO at the previous chunk of the simulation.

A WAM/toy model simulation validates the model partitioning and interpolations, by checking the interpolated field received by the toy model. A FERRET script was set up to be able to plot a 2 dimensional grid originally described in a 1D array (WAM Gaussian grid).

The recent 4.2 version of the NEMO (+SI3 sea ice) model is compiled and launched on the targeted supercomputer. The ORCA2 reference grid is chosen to facilitate the coupling interface implementation phase. The wave coupling is activated by `namelist`¹⁷ but also by enabling the OASIS call in a pre-processing phase¹⁸. Basic bilinear and NN interpolation are chosen. The result on the ORCA2 grid is displayed on Fig 8 for Charnock and drag coefficients.

15 `MPL_IARGC` called in `mpl_init_mod.F90`

16 Equal to one WAM model time step duration

17 `ln_wave=.true.`

18 `key_oasis3` in `cpp_NEMO_configuration.fcm`

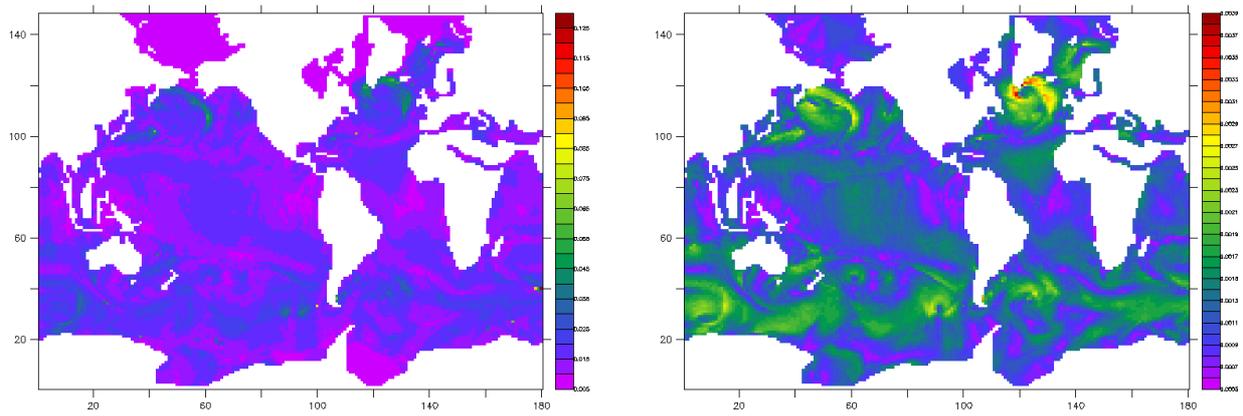


Figure 8: Charnock and drag coefficient example of interpolation on ORCA2 grid

In a further step, it will be necessary to precisely define the physical processes that need to be represented by the coupling, by choosing which quantities have to be exchanged with MFWAM. In that perspective, the choice is facilitated by the NEMO developers, who have previously introduced the `namsbc_wave` namelist to do so.

In the perspective to add the AROME atmosphere model to the ocean-wave coupled model, the existing toy model could be adapted to mimic the coupling of additional quantities with the wave model (such as 10m wind). Technically speaking, everything is now available to make possible the assembling of the full ocean-atmosphere-wave coupled model.

Mission #23 : SMHI, Norrköping (Sweden)

September, 2022

Main Goal: To faster EC-Earth4 interpolation weights and addresses calculations and enhance their accuracy

Summary

Several enhancements have been proposed for both SCRIP interpolation file generation in a pre-processing phase and interpolation parametrisation. A parallel version of the `rdy2cpl` tool is set up and its scaling checked, while better conservative parameters are proposed in the future version of the EC-Earth4 coupling system (ice temperatures of the sea-ice edge). The `rdy2cpl` tool modifications were the occasion to test the new pyOASIS interface and establish the efficacy of Python based coupled toys. Easily configurable, this tool has the potential to serve a broader community

OpenIFS atmosphere model	NEMO global ocean
as part of EC-Earth4, see [19] for reference	NEMO 4.2, see [2] for reference
cycle 43r3v1, tco95, x levels	ORCA1, 31 levels, reference configuration

Supercomputer:

“bifrost”, NSC, Lindköping, Sweden

10,256 cores - Intel Haswell E5-2640v3, 2.6GHz, 16 cores and 256Gb per node

<https://www.top500.org/system/178550/>

The Rossby center of the Swedish Meteorological and Hydrological Institute (SMHI) is one of the main contributor to the EC-Earth model set-up, including the scripting tools that facilitates its use on a set of supercomputers of the community. EC-Earth is a tunable GCM, mainly composed from OpenIFS and NEMO, but which can also be enhance towards Earth System modelling thanks to other components such as the TM5 atmospheric chemistry, the LPJGuess dynamical vegetation, etc. The EC-Earth coupling strategy relies on OASIS3-MCT, and more recently on its version 5. The following support aims at enhancing the computing performance of the new scripting tool developed for the EC-Earth4 version, but also takes benefit of the

collaboration to revisit the interpolation strategy of the coupled system.

1. Interpolation weights production with Python

A comprehensive workflow engine, RunTimeEnvironment (RTE), written with ScriptEngine, is proposed to the EC-Earth. It facilitates the setting of model executable, of experimental conditions and the running on a set of selected EC-Earth community machines. During the pre-processing phase of each experiment, all coupling input files are produced: OASIS auxiliary, interpolation weight and address (W&A) and the `namcouple` parameter files. The model input files are modified beforehand (and accordingly) by the workflow. The coupling setting is managed thanks to a central `yaml` parameter file.

In a first step, a special `namcouple` is produced by translation of the `yaml` file, and provided as an input of a special Python tool : `rdy2cpl`. This tool is a reduced OASIS coupled system, which aims at producing the other OASIS input files required by the coupled simulation. In a second step, another `namcouple` will be produced from the same `yaml` file, this time to feed the model coupled system.

Built within a `mamba` built environment, the `rdy2cpl` tool consists in a small set of Python scripts, from which two main files launched in parallel via the `mpirun` command, and instrumented with the recently developed `pyOASIS` library [21]. The first script (`leader.py`) is the single coupled component that is supposed¹⁹ to receive all coupling fields. It declares all the target variables of the `namcouple`. The second script (`worker.py`) is launched as many times as W&As declared in the `namcouple`²⁰. Each of them produces and writes in the OASIS auxiliary files, the source and target grid variables corresponding to one single coupling field of the `namcouple`. Appropriate Python scripts are available in the `rdy2cpl` library, each of them corresponding to one of the grid in use in the EC-Earth community²¹. In addition to these scripts, if grid related information cannot be easily calculated or store in ASCII format, a set of netCDF input files must also be provided by the workflow²². The worker scripts also declare the corresponding source variables to OASIS and launches the W&A computing²³.

The interpolation W&A computations are performed in OASIS by the SCRIP library. An hybrid MPI-OpenMP parallel version of these computations were implemented in the version 4 of OASIS3-MCT [22]. W&A computations are automatically launched when simulation starts, if not already computed in a previous simulation. In this case, the files where they are stored must be previously copied in the executables directory. W&A are computed by the source

19 This operation is not explicitly performed by the script, which ends with the `pyoasis.Component.enddef` function

20 In order to avoid doubletons, the `rdy2cpl namcouple` must be a reduced version of the simulation `namcouple`

21 For NEMO (ORCA grids), OpenIFS and the EC-Earth runoff mapper model

22 e.g. `DOMAINcfg.nc` for ORCA grids

23 Via this `pyoasis.Component.enddef` function

models. They involve one MPI process per node²⁴ on which the source model is mapped, and as much OpenMP threads as set in the `OMP_NUM_THREADS`²⁵ variable. Each source model is computing its W&A one by one. In addition, a source model cannot start its W&A computations if they are not completed in the previous source model of the coupled system, in such a way that all W&A are performed one by one.

A first modification of the python tool is proposed, in order to reduce the total number of resources required, thus to reduce the `rdy2cpl` computing cost. The role played by the leader script is rather small and can easily be distributed over the workers. Relying on a functionality offered since the first OASIS3-MCT version, both source and target variables can be handled by the same coupled component (for a further MPI exchange within the same single executable) . In our case, worker scripts are modified in such a way that we declare here both source and target variables, and the leader script can be removed.

A first level of parallelism (at source model level) can be easily achieved by commenting out the MPI barrier explicitly coded in the source model loop²⁶ which prevents the concurrent W&A computations on all source models . This barrier aims at reducing the risk of concurrent W&A file writing but also strongly reduce computing performance. Assuming that a control is done beforehand by the function that build the `namcouple` from the `yaml` configuration file, the barrier can be removed and the W&A computations performed by all the source models at the same time.

However, in an attempt to use the officially distributed library without modification, a second solution was investigated. It relies on a new functionality offered by OASIS since the version 4, that allows to set the OASIS global MPI communicator to another value than `MPI_COMM_WORLD`. Practically speaking, this means that MPI processes not managed by the OASIS coupled system can coexist with it. In our case, it would be interesting to make the Python scripts as independent as possible, since no communication is required between them. Ideally, we would like to define one OASIS coupled system per Python script, in such a way that their W&A computations could run independently and concurrently, by avoiding the MPI barrier. A simple modification is brought to our Python script, by defining and sending as a parameter of the OASIS initialisation call a local communicator, defined on one process. Unfortunately, this is not the single modification needed to make possible the concurrent running of several OASIS coupled system :

- i. individual `namcouple` and debug files must be defined for each independent coupled system, thus new indexes defined following the rank in `MPI_COMM_WORLD` communicator
- ii. for that, a new communicator has to be set for `MPI_COMM_WORLD` in addition to the individual global communicator
- iii. the MPI barrier which prevent a concurrent writing of auxiliary variable must be set on

24 As returned by the `MPI_Get_processor_name` function

25 For the best performance, this variable must be set to the number of cores per node. If it is already defined to another value, usually because one of the coupled model is hybrid parallel and its performance requires another value of `OMP_NUM_THREADS`, the W&A computations can be processed on a specific number of threads by defining the `OASIS_OMP_NUM_THREADS` environment variable

26 Barrier on `mpi_comm_global` communicator, with tag `'cpl_setup_n4_global'`, in `oasis_mode_coupler.F90`, aka "Tony's barrier"

this `MPI_COMM_WORLD` communicator instead of the now individual local communicator

It was then not possible to avoid the MPI barrier that makes the W&A computations sequential by simply defining an individual and single process communicator for each `rdy2cpl` instance. We prefer to simply comment out the problematic barrier as described previously. However, our test helps us to better estimate the amount of work necessary to run several OASIS based coupled systems together, e.g. in ensemble runs. Even though our solution was tested on one configuration only and, for simplification reasons, relies on several strong assumptions from which the single number of allocated MPI process for each coupled system, we think that we identified most of the modifications needed to lead a successful simulation of that kind.

As easily as for the first level, a second level of parallelism is set up by increasing the number of OpenMP threads. This requires:

- the compiling of the FORTRAN OASIS API with the appropriate OpenMP flag,
- the set up of the `OMP_NUM_THREADS` environment variable with the appropriate number and
- a special mapping of each MPI process associated to one single worker Python script.

A special care of each particular machine configuration must be given when defining the mapping. On `bifrost`, it did not seem possible to map only one single MPI process per node, but a minimum number of two was required (one process per socket). In this case, the maximum number of eight OpenMP threads can be given (sixteen for hyper-threading).

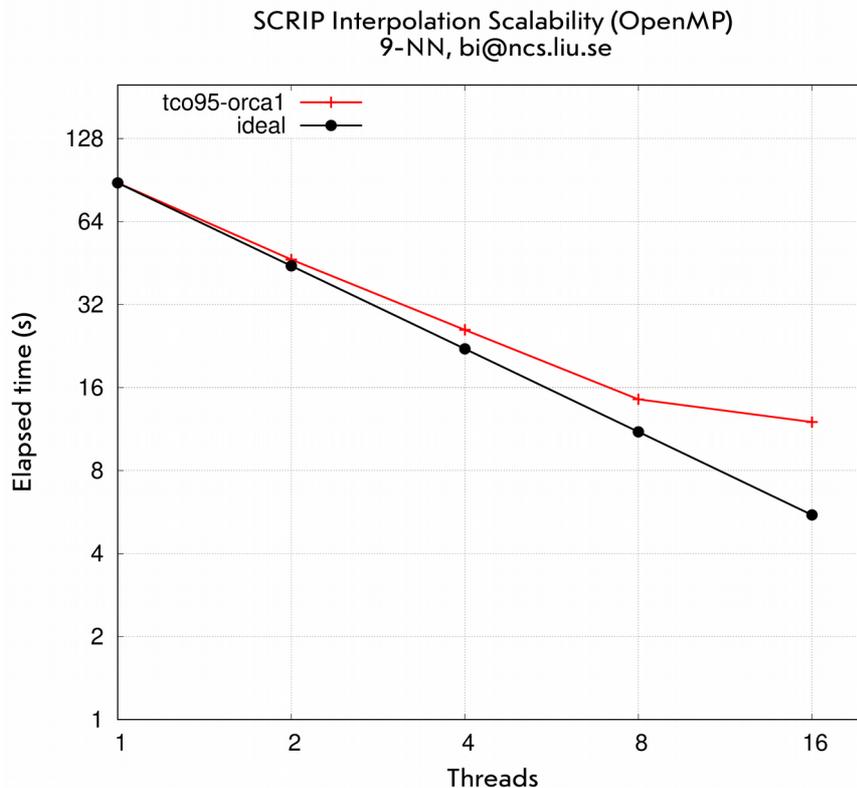


Figure 9: OpenMP scalability of the remapping procedure during a nearest neighbour interpolation

A basic test involving only one interpolation on one socket is performed several times with a varying number of threads. The interpolation is the SCRIP NN (GAUSWGT), requiring 9 neighbours of the tco95 OpenIFS source grid for each ORCA1 target grid points. The scalability is shown in Figure 9. Due to the low resolution of the grids, a saturation can already be noticed on 8 threads, but the overall performance remains good. In addition, one can notice that the hyperthreading option is enhancing the performance, probably because of the low level of the computations memory print. In total, the OpenMP parallel version of rdy2cpl allows a seven fold speedup in that case, which basically validates our approach. Our full parallelisation process can be summarised on Figure 10: every OpenMP parallel rdy2cpl process can now process one W&A concurrently with the others.

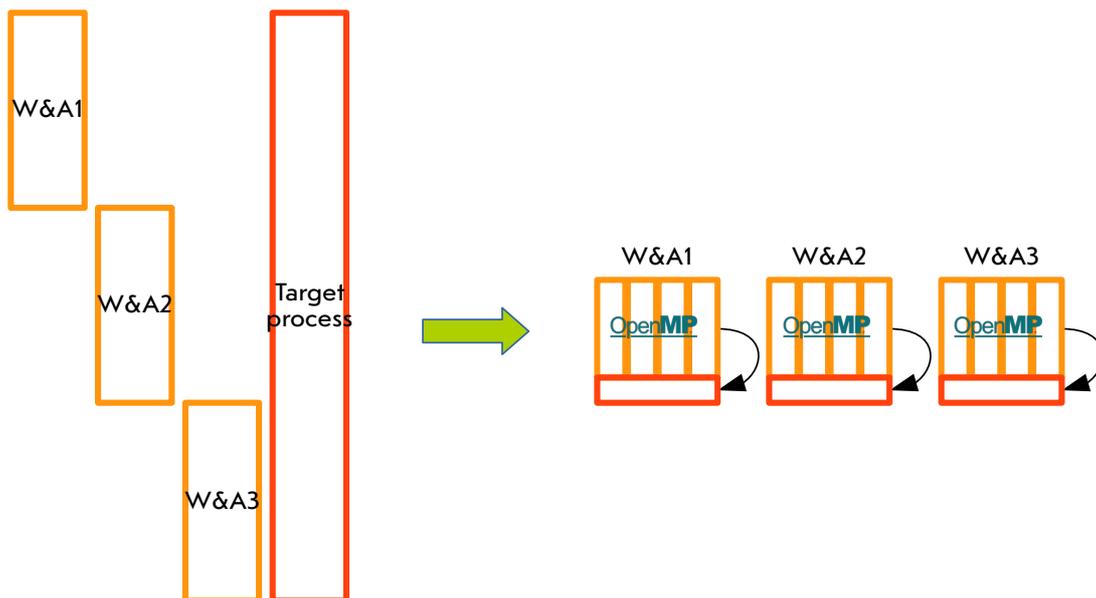


Figure 10: Summary of the tasks aiming at improving the rdy2cpl tool performance: removal of the per interpolation sequentiality, saving of the target process resource, OpenMP parallelism

A fully realistic testing is then done, involving the six interpolations:

- four from ocean to atmosphere back and forth, from which three different ocean grids in the atmosphere to ocean way,
- one from atmosphere to runoff mapper and
- one from runoff mapper to ocean.

The six interpolations are performed by six workers scripts, mapped on six sockets (three nodes). Two sets of test are launched in order to evaluate the performance with two atmosphere grid resolutions. Grid sizes are given in Table 3. All six interpolations use a nine NN algorithm, with Gaussian weights.

Name	IFS tco95	IFS tco319	ORCA1	Runoff mapper
Acronym	IOCL	IOCH	NOTM	RNFA
#points	40,320	421,120	119,160 (360x331)	64,800 (360x180)

Table 3: Model grid sizes

In Figure 11, we compare the total time needed to produce the six W&A files. The reference time is represented on the first line by colour boxes from left to right, proportional to the duration of the three atmosphere to ocean W&A computations (dark blue, yellow, orange), to the atmosphere to runoff mapper one (light blue), to the runoff mapper to ocean one (green) and the ocean to atmosphere one (purple), in that order. Combining the OASIS barrier removal and the use of `rdy2cpl` on six workers shows a 4 (resp. 3) fold speed increase in low (resp. high) resolution set up. These numbers grow to 30 (resp. 23) if computations are spread to 16 OpenMP threads (hyper-threading). With `rdy2cpl`, the total time (red boxes) is the time needed to perform the most expensive interpolation (ORCA1 to IFS t_{co}). We of course checked that W&A variables produced are the same in any cases.

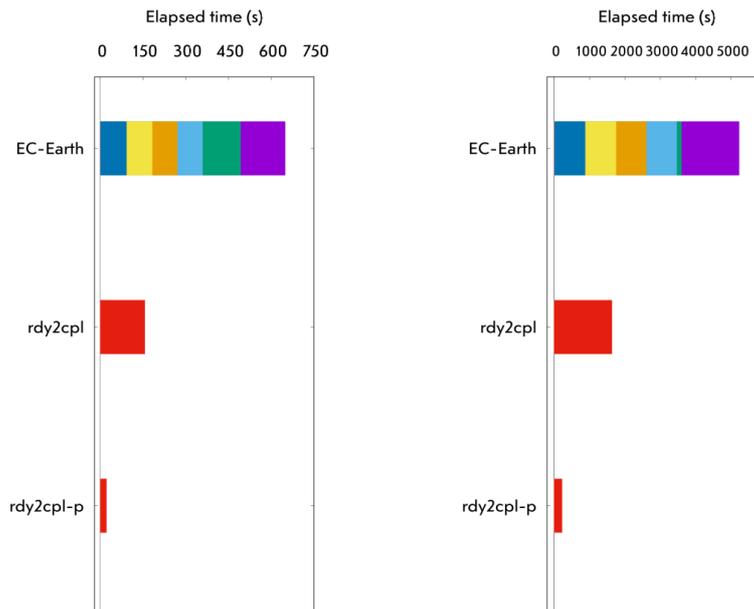


Figure 11: Duration of the six EC-Earth interpolation W&A production phases, with low t_{co95} (left) and high t_{co319} (right) resolution atmosphere grids, using the whole model without parallelism (upper line), using the `rdy2cpl` tool with (lower line) or without (middle line) OpenMP parallelism

2. EC-Earth coupling improvements

On the path to set up the new EC-Earth version 4, a revisiting of the coupling strategies has started in the community. During the limited duration of this ODUS, we focused on conservation issues at the ocean-atmosphere interface, for all the fluxes, including runoff, but also on the problem of surface field interpolation accuracy, particularly at ocean/sea-ice transition zone.

2.1. Land sea-mask mismatch

A simple Gaussian weighted NN interpolation (GAUSWGT) is currently applied to the total heat and water IFS to NEMO fluxes, followed by a global conservation. Despite the robustness of this solution, a small local error is made, particularly along the coast line, and re-distributed,

causing a second error, all over the grid. Its replacement by a more exact locally conservative interpolation (CONSERV) is the object of the next task of our support.

As already stressed during the previous DWD support, the use of the SCRIP CONSERV interpolation cannot be considered without ensuring the best possible matching of ocean and atmosphere model land-sea masks, like e.g. in [8]. Obviously, a land-sea mask mismatch would lead to several side problems, from which:

- the value of a flux distributed to a total ocean grid point area that would be bigger or smaller than the atmosphere grid point area would be distorted, sometimes hugely, which would create numerical anomalies, local biases and sometimes, numerical instabilities in ocean or ice models. Flux is conserved but values can become unrealistic. This class of error is called *interpolation error*.
- source ocean grid points uncovered by any unmasked atmosphere grid points would receive unrealistic values, filled thanks to an additional NN interpolation, which breaks the total conservation again. This class of error is called *conservation error*²⁷.
- on the other direction, independently of the chosen interpolation, the atmosphere unmasked grid points uncovered by any source ocean grid points would be filled with remote areas values of surface variables. The corresponding ocean fluxes would not be transmitted to the ocean and would be lost. This class of error is called *restriction error*.

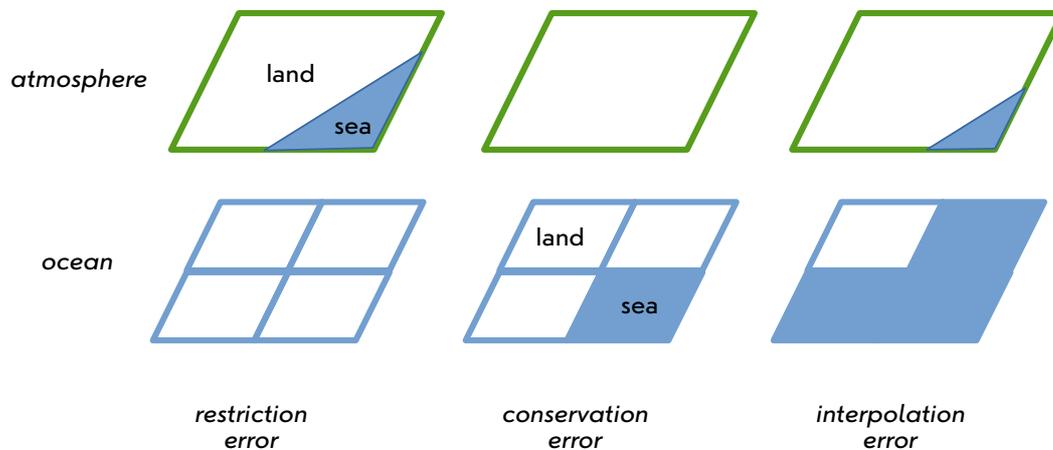


Figure 12: The three possible error classes when interpolating fluxes with locally CONSERVative interpolation without adapting the atmosphere land-sea mask to the interpolated ocean land-sea mask

The perfect matching of ocean and atmosphere land sea masks is achieved by interpolating the ocean mask to the atmosphere grid, using the same interpolation than the one used at runtime. The result is a percentage of land that must be used as a fixed surface variable by the atmosphere model. Unfortunately, many other surface quantities must be modified accordingly, which would require a substantial amount of man power. For that reason, it is important to first evaluate how much fluxes would be modified by using a SCRIP conservative interpolation without fitting the atmosphere to the ocean land sea mask, thus how significant

²⁷ Where we learn that the CONSERV interpolation could lead to conservation errors ...

would be the correction brought by the interpolation change.²⁸

In that purpose, we interpolate the NEMO binary mask²⁹ to the OpenIFS tco95 unmasked grid. For every atmosphere grid point, we get a ratio of NEMO *land* grid point, hereafter named OMASK. We name AMASK the amount of land also, but this time defined in the *atmosphere* model³⁰. Depending on the relative values of these two quantities, every atmosphere grid point can see one and only one of the three different errors previously defined:

- Interpolation error: $OMASK \neq 1$ and $AMASK \neq 1$, but $OMASK \neq AMASK$
- Conservation error: $OMASK \neq 1$ and $AMASK = 1$
- Restriction error: $OMASK = 1$ and $AMASK \neq 1$

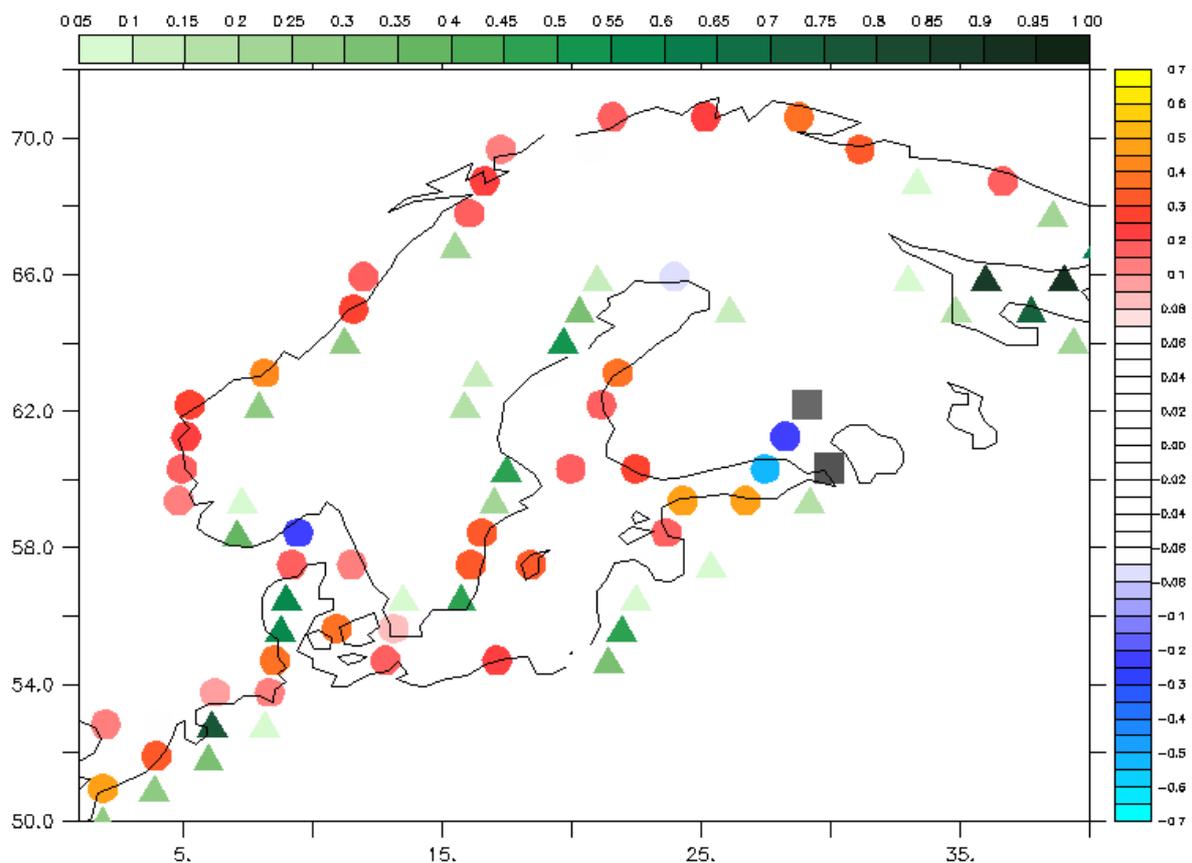


Figure 13: Classification and values of the three kind of interpolation errors due to land-sea mask mismatch between ocean and atmosphere grids: interpolation error (circles, blue if interpolated ocean land-sea mask have more land than atmosphere land-sea mask, red otherwise), conservation error (green triangles), restriction of oceanic flux in atmosphere (gray squares, value~30%). Zoom on Baltic sea

In any cases, the error severity can be evaluated by the difference between the two fractions. A closer look to the position of these grid points can be visualised on Fig. 13, with a focus on the Baltic sea region. As expected, the shore concentrates most of the error points. Many of

²⁸ However, it should be possible to tune another quantity of the OpenIFS surface module: the lake fraction. This workaround started to be evaluated at the end of the support period

²⁹ As seen by OASIS in the `masks.nc` auxiliary file

³⁰ The figures below can be changed if OpenIFS grid points are considered as fully covered by land or ocean, depending on a 50% threshold. This does not modify the main conclusions of this paragraph

them are OpenIFS grid points without any sea fraction but which area is intersecting NEMO sea grid points. This is the case for example above White Sea, which is considered as lake, thus land, by the atmosphere model. Only two atmosphere grid points show restriction error, since the NEMO coast line goes usually less on inland regions than the OpenIFS coastline (as defined as the limit where grid points are totally covered by land or lakes). Most of the error grid points are affected by a land/sea mask fraction difference between the value defined in OpenIFS and the interpolated value from the ocean grid (interpolation error).

A more quantitative evaluation on the whole grid of the error is proposed in Table 4. A relatively high ratio of grid points is erroneous (about 10%). The biggest part deals with interpolation errors, but the level of error is quite low (around 9%) which should not lead to substantial field distortion and numeric instabilities. A smaller part of errors are conservation errors, which can be compensated, as we usually do with non conservative interpolation, by a global average conservation. The last kind of error (restriction), which leads again to a conservation error, affects a negligible amount of grid points and can be compensated by the same mean than the previous error.

	Interpolation error	Conservation error	Restriction error	Total error
# grid points (%)	2776 (6.9 %)	1558 (3.9%)	39 (0.1 %)	4373 (10.8%)
Average error	9 %	44%	19 %	24%

Table 4: Ratio of grid points showing one of the three kind of errors on a tco95 grid (40320 grid points) for a CONSERV interpolation of an ORCA1 grid mask (first line) and average mismatch between atmosphere and ocean interpolated land-sea masks (second line)

To summarise, it seems possible to use the CONSERV interpolation for a tco95 to ORCA1 transformation and without changing the OpenIFS land-sea mask to fit the NEMO one, but it would also lead to local conservation error near the coast line. It is hard to say if this error is larger or smaller than the error resulting from the original non conservative NN one, mainly because there is no "true" value of an interpolated field, thus no exact error value that could be compared. Compared to GAUSWGT, CONSERV is certainly able to reduce the average quantity to be redistributed in the global conservation operation³¹, but the value depends on the variable, and time step, with possible compensation of positive and negative values. It is also difficult to know whether the conservation quality around the coastline would be better or worse, but in addition to the conservation problem, the CONSERV interpolation adds an interpolation error that could lead to numerical instability. Last, but not least, depending on the resolution ratio between grids, the quality of the CONSERV interpolation offshore could also be problematic: when the atmosphere grid is coarser, the local CONSERV could lead to artificial gradients in the ocean interpolated fluxes ("patchy" fields). The best solution, in this case, is to upgrade the conservative interpolation quality by using the second order one, but this requires additional adjustments, such as the computation of derivative, themselves error prone where close to the shore.

Do we have any idea of the error we make when using GAUSWGT instead of CONSERV ? One can compare the number of source grid point used by both method. In the studied

31 With CONSERV, only NEMO coastal grid points are affected by the conservation error, while interpolations for all coastal and offshore grid points are non conservative in the GAUSWGT case

configuration, from both atmosphere to ocean and ocean to atmosphere ways, the GAUSWGT neighbours number is equal 9, while this number varies in CONSERV following the relative resolution of tco95 and ORCA1. This number can be simply calculated by interpolating a one constant value with a previously computed CONSERV W&A files, which weights have been changed and all set to one. The interpolated field on each target grid point will sum up the one value defined on each of their associated source grid points. More resolution is usually observed in the ocean model, which means that the results will be different in the two directions.

From ORCA1 ocean to tco95 atmosphere grid (surface field interpolation), since the tco95 grid is built to keep the grid points areas as constant as possible, the resolution ratio is mainly driven by the ORCA1 area. We split the result into two variables displayed in Fig. 14: target grid points with more (left) or less (right) than 9 sources. Unsurprisingly, the high latitude grid points are requiring more than 9 sources, which would lead to larger missing information in the GAUSWGT case, while the source number overestimation becomes important on Southern hemisphere and very high latitudes only. On areas where source grid point numbers is below 9, target with less than 4 sources are mostly located close to the shore.

A mitigation solution would consist in defining a new surface field interpolation, gathering the number of sources calculated by the CONSERV SCRIP algorithm, but using an inverse distance method to determine the weights. This solution would have the advantage to take into account the right number of neighbours, by using all the neighbours intersected by each atmosphere grid point and only them. Both oversampling and smoothing should be avoided but patchy field could appear and create artificial gradients.

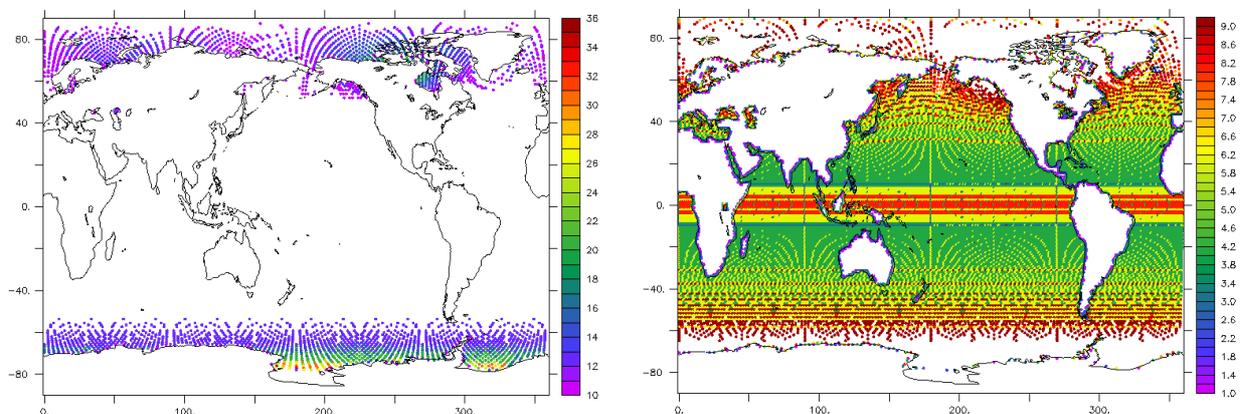


Figure 14: Number of unmasked ORCA1 source grid points for each tco95 target grid point, using a CONSERV interpolation, when number above (left) or below (right) the number of neighbours used for GAUSWGT interpolation (9)

From tco95 atmosphere to ORCA1 ocean grid (fluxes interpolation), the maximum number of area intersections found is 5 (see Fig. 15), which matches with the grid resolution ratio (coarser atmosphere grid). In this case, to implement a variable number of neighbours in the existing GAUSWGT method would not lead to large changes. One would better recommend to reduce the neighbour numbers, even though such reduction would again lead to less smoothing and more artificial gradients.

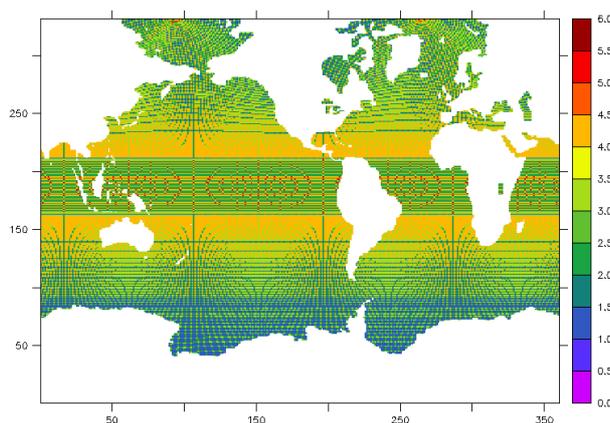


Figure 15: Number of unmasked tco95 source grid points for each ORCA1 target grid point, using a CONSERV interpolation

In short, the use of CONSERV interpolation without atmosphere land-sea mask adjustment seems to bring too many issues compared to its supposed benefit. The initial purpose (to reduce the amount of globally redistributed fluxes) does not lead, as far as we know, to significant local biases, except for runoff, which problematic is summarised below. For any other fields, the robust GAUSWGT with 9 NN, followed by a global conservation, seems to stay the best solution we can propose, if the development time for a land-sea mask adjustment in the atmosphere model (and its associated surface quantities like albedo), to fit the interpolated ocean land-sea mask, cannot be found.

2.2. Ice temperature interpolation

Our ocean to atmosphere coupled variables can either come from the ocean or SI3 ice models. This is the case for surface temperature. Ice temperatures are only available on iced grid points which means that, since there is no possibility of dynamical masking in our coupler, ice temperatures must be completed on non iced grid points by water temperatures. The consequence is that interpolated values close to the sea-ice edge are an average between ice and water temperatures, thus are overestimated (too warm).

As already implemented in several laboratories, we propose to multiply the ice temperatures (& albedos) with their ice fraction before interpolation, and re-divide by the same quantity (but interpolated) after. Of course, it is mandatory to provide zero values where ice temperatures are not available, as described for example in [12]. Long term simulations would be necessary to confirm or not that the effect goes in the expected direction (more ice production during fall).

2.3. Runoff interpolation

As already implemented with similar models (see mission #19 in [15]), we propose to introduce the newly developed locally conservative interpolation in EC-Earth. Similarly to what FOCL community does, a simple runoff mapper (RM) redirects a water content that is

spread on basins onto mouths grid points. In a second step, OASIS interpolates the runoff flux on the ORCA grid with the same Gaussian nearest neighbour interpolation that used for the other fluxes. Without the help of the new OASIS interpolation, the river mouth grid point have to be define on ocean grid point location, to allow GAUSWGT to find enough RM neighbours above the ocean grid points, and avoid loosing too much flux in the non conservative interpolation.

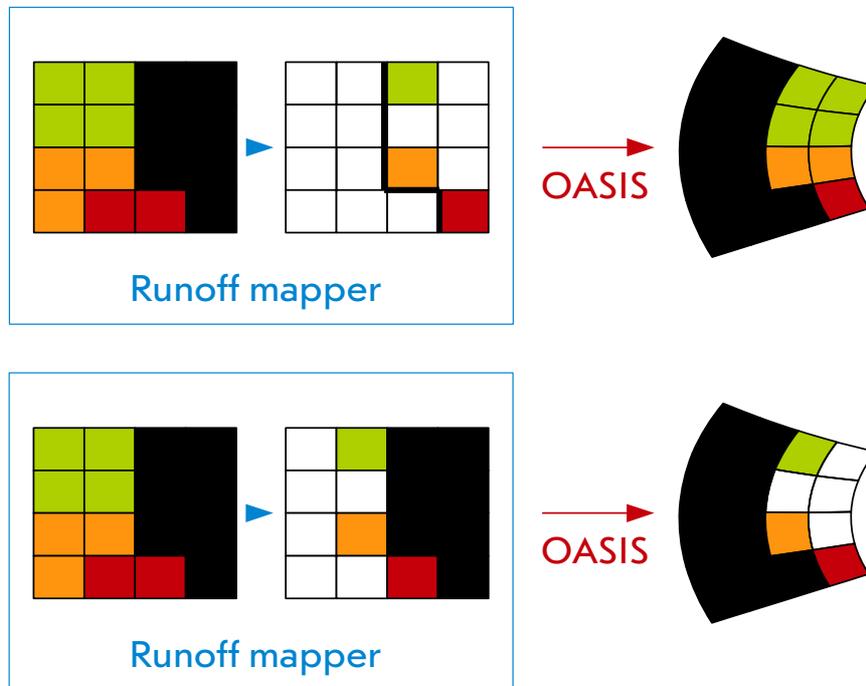


Figure 16: Reference (above) and proposed strategy (below) to interpolate basin water content from the Runoff Mapper grid to the ORCA grid

At the opposite, the new OASIS interpolation is made to be able to redirect and conserve the whole flux of an RM grid point to ORCA grid points, even though these latter are far from the RM grid point center. For that purpose, we propose to slightly change the definition of the river mouths in the RM grid, from land points to ocean points (see Fig. 16). A lack of time prevents us to implement and test this strategy. However, we are convinced that this solution allows (i) to fully conserve the total flux and avoid a global post-processing conservation, (ii) to better control how the outflow is spread over the ORCA grid and (iii) an automatic method to be easily implemented to select coast line grid points of the basin (see [23] for a description of a GIMP based tool and how it could extract contours). This should be convincing enough for a slight change in the river mouth definition (e.g. at higher resolution).

2.4 Load balancing

We take benefit of the `rdy2cp1` Python development to extend slightly its algorithm and add to the initial W&A production a dozen of simple exchanges of one coupling field. The load balancing analysis [18] is enabled, and trace files produced, which validates this function with Python tools. Python is also the language we use to produce graphical output [17] of the OASIS events timeline, and finally evaluate the coupled model load balancing.

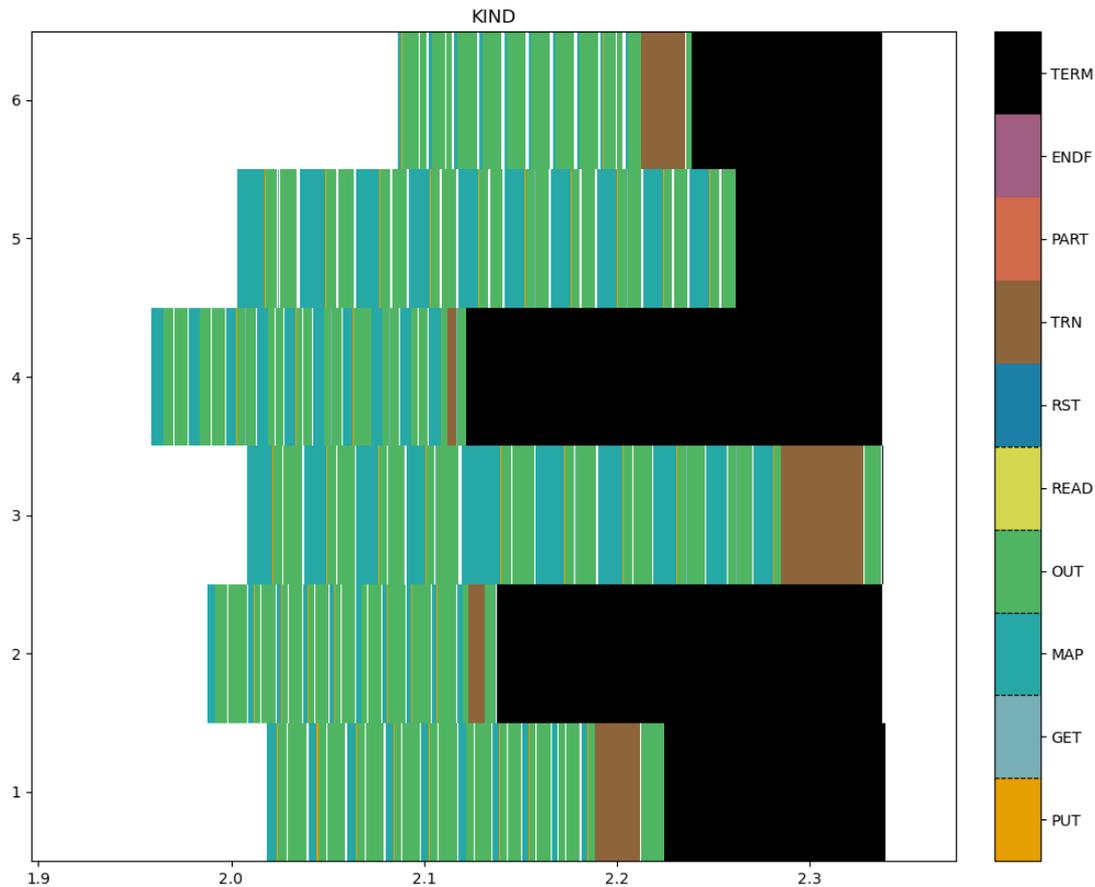


Figure 17: OASIS timeline events for 6 rdy2cpl processes running in parallel

Matplotlib is the only Python module to be added to the rdy2cpl environment, since the other one are already necessary to the coupled simulation (this is exactly the kind of simplification that was targeted when the load balancing function was re-written). The pyLucia.py tool was successfully launched on the bifrost machine, after a small correction in the color bar definition, required by a recent version of Matplotlib (3.6). This will help us to update the script in the OASIS repository.

One possible development of this work would be to make available to the whole community a small Python script to include in any existing system, that could automatically provide the timeline at the end of the simulation, avoiding the post-processing phase.

3. OASIS repository size reduction

OASIS coupler sources are distributed within the main EC-Earth git repository by linking with the official OASIS gitlab server³². Practically speaking, OASIS is declared as a submodule in the EC-Earth repository. Each time that an EC-Earth directory is cloned, one of the official OASIS releases (depending on the EC-Earth version) is also downloaded.

Since version 5.0, the OASIS gitlab release comes with a huge history files³³, due to an old inclusion plus deletion of one of the auxiliary file provided with toy model examples³⁴. This

³² <https://gitlab.com/cerfacs/oasis3-mct>

³³ Included in the .git directory, and visible when its size is displayed

³⁴ Exactly examples/regrid_environment/OASIS/grids.nc

ghost file represents more than 50% of the total 2.2 Gb directory that has to be downloaded each time than OASIS, thus EC-Earth, is cloned.

Thanks to Klaus Zimmermann, we learn the existence of tools³⁵ that could measure and reduce the size of the history trace, but we also learn that this has an effect on the `git` nomenclature and could change the `git` tree and the handling of the local copies at users level. For these reasons, a quick corrective action would be preferable, to avoid affecting more OASIS3-MCT v5 users with an unavoidable corrective action. In addition, large active data files are still to be downloaded by any users, while most of the time, only sources are required. That's why we think that one perennial solution would consist in removing `examples/` and `doc/` directory from the main OASIS `gitlab` repository and propose its distribution in a separate repository.

4. Conclusion

Several enhancements have been proposed for (i) W&A files generation in a pre-processing phase, (ii) EC-Earth interpolation parametrisation and (iii) OASIS `gitlab` providing. A parallel version of the `rdy2cpl` tool is set up and its scaling checked, while better conservative parameters are proposed in the future version of the EC-Earth4 coupling system (ice temperatures of the sea-ice edge). The `rdy2cpl` tool modifications were the occasion to test the new `pyOASIS` interface and establish the efficacy of Python based coupled toys. Easily configurable, this tool has the potential to serve a broader community. Concerning the interpolation choice in EC-Earth, the several options discussed still need to be further investigated before being selected or not. The conservation issue rises many more questions, as always, considering that conservation is a cross cutting issue in many parts of the climate model, and not only a specific problem of the interface.



Figure 18: on the path to the last dedicated support

35 https://docs.gitlab.com/ee/user/project/repository/reducing_the_repo_size_using_git.html

Community impact

During the selection procedure, the panel emphasised the importance of not restraining the support to a one to one collaboration but rather preferring actions that could have a broader impact on communities. We tried to quantify this community impact, in a table that summarises (i) the oral communications organised and the origin of the participant/audience, (ii) code updates in official centralised repositories, from which OASIS `gitlab` and (iii) written communications (emails) to laboratories making part of the hosting laboratory working network. This counting necessarily neglects any action in link with our work, organised by the hosting laboratory, that could take place after the dedicated support period.

	DWD	Météo-France	SMHI
Talks/meetings	* Preliminary meeting (2 people, October 2021)	* Preliminary meeting (~30 people, Météo-France, April 2022) * Presentation meeting (~30 people, July 6 th 2022)	* Results presentation to the team (10 people, October 3 rd , 2022) * Half-day meeting with a Lund University LPJGuess user (October 5th)
Repository updates	XIOS and OASIS bug reports ICON/NEMO repository update: https://gitlab.dkrz.de/icon-nemo/roam	Not yet	rdy2cpl repository: https://github.com/uwefladrich/rdy2cpl.git
Networks	NEC support (several emails). Community contacted later due to remote support: HEREON Geesthacht and Rostock	Community contacted: <ul style="list-style-type: none"> • HEREON Rostock • MF "Building" project • Mercator Ocean 	EC-Earth community

Table 2: Quantification of community level communications during support

The community impact cannot be fully evaluated on such short period and using such dangerously formal criteria, but it gives an idea on how practicality results of the support can be used outside the hosting laboratory. In particular, this post pandemics exercise shows a partial recovery of the networking activities, because of the scientists presence (when on-site service) but a persistent lack of larger interactions (participation to larger conference or community workshops).

The author wishes to acknowledge the use of the Ferret program for analysis and graphics in this report (Ferret is a product of NOAA's Pacific Marine Environmental Laboratory), in addition to graphics from Matplotlib, a Sponsored Project of NumFOCUS, a 501(c)(3) non profit charity in the United State. He also wants to acknowledge for their help Jennifer Brauch, Barbara Frueh, Lofti Aouf, Stéphane Law Chune, Michael Sahlin & Klaus Zimmermann. A particular acknowledgement to Vera Maurer (also for her precise review of our collaborative work), Alice Dalphinnet, Uwe Fladrich & Klaus Wyser, who were the careful hosts of these supports. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824084.

Bibliography

- [1] Ho-Hagemann, Ha Thi Minh, 2022: The OASIS3-MCT Coupling Interface for ICON-CLM (1.0.0). Zenodo. <https://doi.org/10.5281/zenodo.5833118>
- [2] Madec, G. & NEMO System Team, 2019: "NEMO ocean engine", Scientific Notes of Climate Modelling Center (27) – ISSN 1288-1619, Institut Pierre-Simon Laplace (IPSL)
- [3] Wang, S., Dieterich, C., Döscher, R., Höglund, A., Hordoir, R., Meier, H.E.M., Samuelsson, P., & Schimanke, S., 2015: Development and evaluation of a new regional coupled atmosphere-ocean model in the North Sea and Baltic Sea. *Tellus A*, **67**, 24284
- [4] Maisonnave, E., 2021: NEMO performance optimisation on NEC SX-Aurora TSUBASA , Working Note, **WN/CMGC/21/37**, CECI, UMR CERFACS/CNRS No5318, France
- [5] Maisonnave, E., 2022: NEMO performance on pre-Exascale processors : NEC SX-Aurora TSUBASA and Fujitsu PRIMEHPC FX700 , Working Note, **WN/CMGC/22/22**, CECI, UMR CERFACS/CNRS No5318, France
- [6] Caubel, A., 2022: Joint usage of OASIS3-MCT and XIOS in climate models, LSCE/IPSL, https://oasis.cerfacs.fr/wp-content/uploads/sites/114/2022/02/Joint_usage_OASIS3-MCT_XIOS_2022.pdf
- [7] Will, A., Akhtar, N., Brauch, J., Breil, M., Davin, E., Ho-Hagemann, H. T. M., Maisonnave, E., Thürkow, M., & Weiher, S., 2017: [The COSMO-CLM 4.8 regional climate model coupled to regional ocean, land surface and global earth system models using OASIS3-MCT: description and performance](#), *Geosci. Model Dev.*, doi:10.5194/gmd-10-1549-2017
- [8] Voldoire, A., et al., 2017: SURFEX v8.0 interface with OASIS3-MCT to couple atmosphere with hydrology, ocean, waves and sea-ice models, from coastal to global scales, *Geosci. Model Dev.*, **10**, 4207–4227, <https://doi.org/10.5194/gmd-10-4207-2017>
- [9] Hanke M. and Redler, R., 2019: New features with YAC 1.5.0. Reports on ICON, No 3. doi: 10.5676/DWD_pub/nwv/icon_003.
- [10] Maisonnave, E. 2021: How to interpolate atmosphere fluxes to an AGRIF zoom ? The ECHAM & OpenIFS test case (in French) , Technical Report, **TR/CMGC/21/200**, CECI, UMR CERFACS/CNRS No5318, France
- [11] <http://nco.sourceforge.net>
- [12] Maisonnave, E., & Voldoire, A., 2018: Exchange grid: implementation of a test configuration to evaluate fluxes modification , Working Note, **WN/CMGC/18/47**, CECI, UMR CERFACS/CNRS No5318, France
- [13] Voldoire, A., 2020: River to ocean models interpolation. [Research Report] CNRM, Université de Toulouse, Météo-France, CNRS (<https://hal-meteofrance.archives-ouvertes.fr/meteo-02986574>)
- [14] Maisonnave, E., 2020: Locally conservative OASIS interpolation using target grid nearest neighbours , Technical Report, **TR/CMGC/20/166**, CECI, UMR CERFACS/CNRS No5318, France
- [15] Maisonnave, E. & Kjellsson, J., 2021: [OASIS Dedicated Support, 5th annual summary](#), Technical Report, **TR/CMGC/21/150**, CECI, UMR CERFACS/CNRS No5318, France
- [16] Balaji, V., Maisonnave, E., Zadeh, N., Lawrence, B. N., Biercamp, J., Fladrich, U., Aloisio, G., Benson, R., Caubel, A., Durachta, J., Foujols, M.-A., Lister, G., Mocavero, S., Underwood, S., & Wright, G., 2017: CPMIP: Measurements of Real Computational Performance of Earth System Models in CMIP6, *Geosci. Model Dev.*, **46**, 19-34, doi:10.5194/gmd-10-19-2017
- [17] Piacentini, A., & Maisonnave, E., 2020: [Interactive visualisation of OASIS coupled models load imbalance](#), Technical Report, **TR/CMGC/20/177**, CECI, UMR CERFACS/CNRS No5318, France
- [18] Maisonnave, E., Coquart, L., & Piacentini, A., 2020: [A better diagnostic of the load imbalance in OASIS based coupled systems](#), Technical Report, **TR/CMGC/20/176**, CECI, UMR CERFACS/CNRS No5318, France
- [19] Döscher, R., Acosta, M., Alessandri, A., Anthoni, P., Arsouze, T., Bergman, T., Bernardello, R.,

- Boussetta, S., Caron, L.-P., Carver, G., Castrillo, M., Catalano, F., Cvijanovic, I., Davini, P., Dekker, E., Doblas-Reyes, F. J., Docquier, D., Echevarria, P., Fladrich, U., Fuentes-Franco, R., Gröger, M., v. Hardenberg, J., Hieronymus, J., Karami, M. P., Keskinen, J.-P., Koenigk, T., Makkonen, R., Massonnet, F., Ménégos, M., Miller, P. A., Moreno-Chamarro, E., Nieradzik, L., van Noije, T., Nolan, P., O'Donnell, D., Ollinaho, P., van den Oord, G., Ortega, P., Prims, O. T., Ramos, A., Reerink, T., Rousset, C., Ruprich-Robert, Y., Le Sager, P., Schmith, T., Schrödner, R., Serva, F., Sicardi, V., Sloth Madsen, M., Smith, B., Tian, T., Tourigny, E., Uotila, P., Vancoppenolle, M., Wang, S., Wårlind, D., Willén, U., Wyser, K., Yang, S., Yepes-Arbós, X., and Zhang, Q., 2022: The EC-Earth3 Earth system model for the Coupled Model Intercomparison Project 6, *Geosci. Model Dev.*, **15**, pp. 2973–3020, <https://doi.org/10.5194/gmd-15-2973-2022>
- [20] Voldoire, A., Decharme, B., Pianezze, J., Lebeaupin Brossier, C., Sevault, F., Seyfried, L., Garnier, V., Bielli, S., Valcke, S., Alias, A., Accensi, M., Arduin, F., Bouin, M. N., Ducrocq, V., Faroux, S., Giordani, H., Léger, F., Marsaleix, P., Rainaud, R., Redelsperger, J. -L., Richard, E. and Riette, S., 2017: SURFEX v8.0 interface with OASIS3-MCT to couple atmosphere with hydrology, ocean, waves and sea-ice models, from coastal to global scales, *Geosci. Model Dev.*, **10** (11), pp. 4207-4227, doi:10.5194/gmd-10-4207-2017
- [21] Gambron, P., Ford, R., Piacentini, A. and Valcke, S., 2021: pyOASIS - a python and C interface for OASIS3-MCT, CECI, Université de Toulouse, CNRS, CERFACS, Toulouse, France, **TR-CMGC-21-56**, Technical report
- [22] Piacentini, A., Maisonnave, E., Jonville, G., Coquart, L., & Valcke, S., 2018: A parallel SCRIP interpolation library for OASIS , Working Note, **WN/CMGC/18/34**, CECI, UMR CERFACS/CNRS No5318, France
- [23] Maisonnave, E., 2019: persona version 2.1, how to graphically patch a mask variable , Working note, **WN/CMGC/19/107**, CECI, UMR CERFACS/CNRS No5318, France

Appendix

Costs/Sustainability

A partial recovery of quality transports across the continent and a still partial presence of the community scientists on site has finally prevented to provide the ODUS service in this full extension. Only one visit at site has been organised, which necessarily reduced both cost and carbon footprint of the exercise, without presuming of the service quality. In that context, the footprints are mainly caused by our single long distance travelling, because of the low resolution of the model used, and thanks also to the low level of electricity needed by our nomad workstation (Arm Cortex A53).

These pandemic consequences prevent us to take further conclusions about the overall 2019-2022 costs and footprints. The most we can do is to hope that a return to the normal rules will make possible a complementary analysis, and maybe take conclusions about the relevance of terrestrial journey in comparison to the overall service footprint. But one may assume that in a full sustainability conversion of our activities, both limitations in computing power consumption and travelling would be required.

	Cost	Travel	Computing			Total Carbon footprint	
	(€)	(Km)	(KgCO ₂ e) ^{36,37}	(Core.h)	(kWh)	(KgCO ₂ e) ³⁸	(KgCO ₂ e)
<i>DWD</i>	0	0 ³⁹	0	1	7	0	0.00
<i>SMHI</i>	4,000	5,800	650	50	1	0	650
<i>Météo-France</i>	0	0	0	400	2	0.2	0.20
Total	4,000	5,800	650	451	10	0.2	650

36 SNCF carbon intensity high speed train : 2.4 gCO₂equ/Km, intercity : 8.1 gCO₂equ/Km, from <https://www.oui.sncf/aide/calcul-des-emissions-de-co2-sur-votre-trajet-en-train> and <https://ressources.data.sncf.com/explore/dataset/emission-co2-tgv/>

37 For ferry travel, <https://klimatsmartsemester.se>

38 Carbon intensity of High voltage in Sweden (45 g/kWh), France (100) and Germany (599), according to Moro A., Lonza L., 2018: [Electricity carbon intensity in European Member States: Impacts on GHG emissions of electric vehicles](#), Transportation Research Part D: Transport and Environment, 64 , pp. 5-14.

39 A preliminary meeting took place during a short stop in Frankfurt, taking benefit of the previous Toulouse-Kiel railway travel