



IS-ENES2 MILESTONE (M -N°: 10.4) Implementation of the benchmark suite for evaluation of coupling strategies

CERFACS Working Notes, WN/CMGC/16/132

File name: IS-ENES2_M10.4.pdf

Reporting period: 01/10/2014 – 30/03/2016

Author: **Sophie Valcke**

Release date: **30/03/2016**

Abstract

This milestone is linked to the WP10 task 3 « Evaluation of coupling strategies ». The intention of this task is to define and implement a suite of coupled benchmarks based on simplified model components that capture the essence of the coupling challenges in climate models without the complexities of the science. After having defined the main features of the benchmark suite in M10.1, we detail here the test cases that will, in a first step, practically implement the benchmark suite and we report on the part of these test cases that have been already coded by the different partners. In a next step, the test cases chosen will be run on different computing platforms and analysed to compare key characteristics of few coupling technologies.

Project co-funded by the European Commission's Seventh Framework Programme (FP7; 2007-2013) under the grant agreement n°312979

Dissemination Level

PU	Public	X
PP	Restricted to other programme participants including the Commission Services	
RE	Restricted to a group specified by the partners of the IS-ENES2 project	
CO	Confidential, only for partners of the IS-ENES2 project	

Table of contents

1.	Introduction	4
1.1	Objectives	4
1.2	Context	4
1.3	Content of the milestone	5
2.	Stand-alone component specifications	5
3.	Coupling specifications and other benchmark characteristics	6
3.1	Coupling characteristics of the coupled components	6
3.2	Priority combinations of coupling characteristics	8
3.3	Other benchmark characteristics	10
4.	Practical implementation of benchmark test cases	10
4.1	selfgenreglatlon-selfgenreglatlon:	10
4.2	stretchlatlon-selfgenreglatlon:	10
4.3	icosa-icosa :	10
4.4	cubedsphere-selfgenreglatlon :	10
4.5	Stretchlatlon-icosa	11
4.6	Stretchlatlon-gaussred	11
5.	Perspectives	11

Executive Summary

The milestone 10.4 (M10.4) “Implementation of the benchmark suite for evaluation of coupling strategies » is the second milestone of the WP10 task 3 « Evaluation of coupling strategies ». A significant part of the benchmark suite, which was defined in the previous milestone 10.1 (M10.1) “Definition of the benchmark suite for evaluation of coupling strategies » and which was further defined as detailed here, has been implemented and is reported in this document.

1. Introduction

1.1 Objectives

The milestone 10.4 (M10.4) « Implementation of the benchmark suite for evaluation of coupling strategies » is linked to the WP10 task 3 « Evaluation of coupling strategies ». The intention of this task is to define and implement a suite of coupled benchmarks based on simplified model components that capture the essence of the coupling challenges in climate models without the complexities of the science. After having defined the main features of the benchmark suite in M10.1 « Definition of the benchmark suite for evaluation of coupling strategies », we detail here the test cases that will, in a first step, practically implement the benchmark suite and we report on the part of these test cases that have been already coded by the different partners. In a next step, the test cases chosen will be run on different computing platforms and analysed to compare key characteristics of few coupling technologies. The results will be presented in D10.3 “Report on benchmark suite for evaluation of coupling strategies” due month 42.

1.2 Context

In milestone M10.1, we first described the possible functions of coupling technologies and characteristics of the Earth System Models (ESMs) supported by these coupling technologies in a series of mindmaps “CouplingTechnology”, “Components”, “Metadata”, “Composition” and “Deployment”. This community work on the characterization of ESM coupling started during the « 2nd Workshop on Coupling Technologies » held in Boulder in February 2013 and was finalized by IS-ENES2 partners interacting with colleagues of the US project Earth System Bridge (<https://earthsystemcog.org/projects/es-fdl/>).

The next step was to prioritize the coupling characteristics to benchmark; these were identified as:

- the component model grids,
- the number of MPI ranks used to run the component models,
- the number of fields exchanged between the components,
- the frequency of exchange.

It was also mentioned that code intrusion, development time and issues met during development and techniques for overcoming them, which are all aspects of ‘ease of use’, will also be evaluated and reported on, although they are difficult to quantify.

Finally, the main features of a benchmark suite for the evaluation of these coupling characteristics were defined. It was established that the suite will first consist of a number of pre-coded stand-alone components running on different grids; these component will then be assembled thanks to the different coupling technologies in well-defined coupled test-cases; finally the resulting coupled systems will be run on different platforms to measure the performances of the different coupling technologies with respect to the coupling characteristics identified.

1.3 Content of the milestone

In this document, we start by recalling the specifications of the stand-alone components as detailed in M10.1 so to have here a complete specification of the benchmark suite (section 2). Then we describe in detail in section 3, the specifications of the coupled test cases that could be realised with the different coupling technologies based on the stand-alone components to evaluate the priority characteristics identified above. These specifications were started in M10.1 but have been completed since then. Finally, the test cases that will in practice be implemented by the different partners and the status of their implementation is presented in section 4.

2. Stand-alone component specifications

It was defined in M10.1 that the stand-alone components will consist of simple, individual model codes containing no physics or dynamics but representative of real models in term of coupling characteristics. The specifications of these stand-alone components are the following:

- They will be coded in Fortran90 and implemented as a subroutine, or hierarchy of subroutines, wrapped into a driver that will enable their stand-alone execution.
- They will internally be parallel MPI codes.
- They will not refer to any coupling technology.
- They will define a number of 2D Real Fortran arrays, which represent coupling fields that will be received (in) and sent (out) by the model when it is deployed in a coupled context; these coupling fields will appear as:
 - IN and OUT arguments of the subroutine declared in the driver (rga_field1)
 - arrays in shared modules (rma_field1)
 - local data declared at a particular, possibly deep, level in the subroutine call tree (rla_field1 & rla_field2)
- They will run for 100 time steps..
- In the stand-alone mode, the coupling fields will be initialised from a utility library routine or from a NetCDF file.
- They will be parameterised in terms of the number of in/out fields, i.e. number of coupling exchanges.
- Their coupling fields will be defined on specific grids (i.e. on a “central” point defined per cell) used in real climate model components for at least two resolutions, i.e. for a typical “coarse” global grid of ~200 km resolution and a typical “high-resolution” global grid of 20-50 km. These grids, their possible type of parallel distribution (partition type – PT) and the name we will use to refer to, are:
 - Irregular, stretched and rotated latitude-longitude mesh, following the ORCA configuration of the NEMO ocean model (stretchlatlon); the partition type for this grid will be the BOX one¹, i.e. mxn regular rectangle partitions (PTmn)
 - Quasi-uniform icosahedral mesh, following the atmospheric DYNAMICO model (icosa); this grid is naturally partitioned into diamonds and sub-diamonds (PTdiamond)
 - Quasi-uniform cubed sphere mesh (cubedsphere); the partitioning is achieved by splitting the “cube” into its six square panels, then each panel is sub-

¹ see OASIS3_MCT User Guide, p. 13, http://www.cerfacs.fr/oa4web/oasis3-mct_3.0/oasis3mct_UserGuide.pdf

partitioned, using partitions of BOX type, into $n \times n$ rectangular partitions. This results in a total of $6 \times n \times n$ partitions (PT6nn).

- Gaussian Reduced mesh, following the atmospheric ARPEGE model (gaussred)²; the partition considered for this grid is the APPLE one¹ (PTapple)
- Self-generated regular latitude-longitude grid allowing arbitrary resolutions to be used (selfgenreglatlon); this additional flexibility in resolution choices is expected to be useful for very high-resolution tests on the latest largest systems; two partitions of BOX type will be considered for this grid, either the one with $n \times n$ rectangle partitions (PTnn) or the one with $n \times 1$ partitions, i.e. partitioned only in one direction (PTn1).

3. Coupling specifications and other benchmark characteristics

The list of coupling characteristics and their possible combinations in coupled test cases is of course almost infinite. We specify how we consider sampling the priority coupling characteristics in our first coupled benchmark implementations. Then we detail the combination of those characteristics that will be given priority so to evaluate the coupling technology performances. Finally, we provide additional general specifications of the coupled benchmarks.

3.1 Coupling characteristics of the coupled components

The coupling characteristics of the coupled components that will be considered with priority are the following:

3.1.1 Grid type of the coupled components (GT)

We decided to focus on the following combination of grids (see section 2. for detailed definition) because they represent couples of grids in current climate models:

- selfgenreglatlon - selfgenreglatlon
- stretchlatlon - selfgenreglatlon
- icosahedron - icosahedron
- cubedsphere - selfgenreglatlon
- stretchlatlon - gaussred
- stretchlatlon - icosahedron

3.1.2 Grid size of the coupled components (GS)

As written above, each stand-alone components will be coded so to use at least two resolutions, i.e. a typical “coarse” global grid of ~200 km resolution with $O(100 \times 100)$ grid points (small) and a typical “high-resolution” global grid of 20-50 km with $O(1000 \times 1000)$ grid points (large). The self-generated regular latitude-longitude can also be used at “very high resolution” of ~2km with $O(10000 \times 10000)$ (very large). The following combination of grid sizes could be tested:

- GSs: small – small
- GSll: large – large

² In fact, the stand-alone component using the Gaussian Reduced grid will not be coded per se but we include it in the grid type tested as pre-coded coupled toy models based on this grid already exist and will be used to report on coupling technologies performances.

- GSIs: large – small
- GSvv: very large – very large

3.1.3 Number of processes used to run the coupled components (NP)

The components can be parallelised in an arbitrary number of tasks and therefore run on an arbitrary number of cores. We propose to consider cases where:

- both components run on the same number of cores, from O(1) to O(10000) with each one being approximately an order of magnitude larger than the previous one (NP_{xy}, NP_{xyy}, ..., NP_{xyyyyy})
- one component runs on O(10) cores and the other runs on O(10), O(100), O(1000) times more cores (NP_{xy10}, NP_{xy100}, NP_{xy1000})

To have a minimum size of messages, we will constraint these tests so that the minimum of grid points per process is 50. So depending on the grid resolution, some of these numbers of processes might not be tested.

3.1.4 Layout (Lay)

The components can run on the same sets of cores or on disjoint sets of cores:

- LayS: Shared cores
- LayD: Disjoint cores

3.1.5 Number of Coupling fields (NC)

The components can exchange an arbitrary number of distinct coupling fields (i.e. individually received or sent by the code):

- 2, 20, 200, 2000 (NC₂, NC₂₀, NC₂₀₀, NC₂₀₀₀)

3.1.6 Matching of coupling fields between the components

In the components, the receive and send actions will be implemented respectively at the beginning and at the end of each time step. The following matching will be implemented between the different coupling fields of the two components (see section 2 for the definition of the different fields). The subroutine local field rla_field2 will be sent from the first component to the second one that will receive it in the shared module array rma_field1; the second component will then sent a global field rda_field1 (corresponding to rga_field1 in the driver) that will be received in the first component in a local field array rla_field1:

- rla_field2 (component1) -> rma_field1 (component2)
- rla_field1 (component1) <- rda_field1 (component2)

The ordering of the coupling exchanges will be defined by the schedule of the coupled components (see next paragraph).

In the codes, the output coupling fields rla_field2 and rda_field1 will be defined (via rla_field1 and rda_field1 respectively, see below) by an analytical function based on the latitude and the longitude. The coupling algorithm will be such that, in each component, the output coupling field will defined by its input coupling field adding an increment at each time step. Thereby, checking the correctness one coupling field at the end of the run will ensure that all the coupling exchanges were done correctly. To ensure this, the relation between the coupling fields will be as follows:

- rla_field2 = rla_field1 + 1

- $rda_field1 = rma_field1 + 1$

3.1.7 Schedule of coupled Components (SC)

The coupling algorithm can implement concurrent and sequential execution of the coupled components.

- **Concurrent (SCcon)**

In a concurrent schedule, a coupling field delivered by one model at the end of its coupling period is received by the other model at the beginning of the following coupling period and vice-versa. This ensures that both components run their time steps in parallel. The fields required by each component at the beginning of the first coupling period need to be “primed” by some means, e.g. read from a coupling restart file or assigned directly by the component. Concurrent coupling is also known as “asynchronous” coupling in the sense that for each coupling period, a component uses the fields produced by the other component during the previous period.

- **Sequential (SCseq)**

In a sequential schedule, one component uses some priming mechanism to get its input coupling fields at the beginning of the run. This component can then run for the number of time steps corresponding to the coupling period and provide its output coupling fields to a second component. The second component is then able to run. Meanwhile, the first component waits until the second component has run its coupling period. When the second component has completed its coupling period, the second component provides its output coupling fields to the first component, which is then able to run for a second coupling period. The second component then waits, and so on. This schedule results in one component running when the other is waiting and vice-versa. We can note here that if the coupling exchanges occur every time step and if the work of the second component during its time step is null, the sequence of the first component sending a coupling field to the second followed by the second component sending back a coupling field to the first component is simply a “ping-pong” exchange.

3.2 Priority combinations of coupling characteristics

For each couple of components running on grids listed in 3.1.1, the combination of component coupling characteristics having priority so to evaluate the performance of the different aspect of coupling technologies are the following:

3.2.1 Impact of the number of cores per component combined with the impact of the grid sizes

Each coupled model will be run:

- for the type of partition described for each grid in section 2;
- for high-resolution grids on both sides (GSII), for a combination of low and high-resolution grids (GSIs) and also, for the selfgenreglatlon-selfgenreglatlon couple only, for the very high resolution grids (GSvv);

- for a number of cores per component varying from $O(1)$ to $O(10000)$ with each one being approximately an order of magnitude larger than the previous one (NPxxxy, NPxxxyy, ..., NPxxxyyyyy)
- for 2 coupling fields (NC2)
- for the disjoint layout, LayD,
- for the sequential schedule (SCseq)

The total number of runs involved for each couple of grids is therefore 10 for all couple of grids (2 grid sizes x 5 number of cores) except for selfgenreglatlon-selfgenreglatlon where it is 15 (3 grid sizes x 5 number of cores).

3.2.2 Impact of different ratio in the number of processes used for each component

Each coupled model will be run:

- for the type of partition described for each grid in section 2;
- for high-resolution grids on both sides (GSII),
- for a one component running on $O(10)$ cores and the other running on $O(10)$, $O(100)$, $O(1000)$ times more cores (NPxy10, NPxy100, NPxy1000)
- for 2 coupling fields (NC2)
- for the disjoint layout, LayD,
- for the sequential schedule (SCseq)

The total number of runs involved for each couple of grids is therefore 3 (3 ratios of number of cores for the components).

3.2.3 Impact of schedule and layout

Each coupled model will be run:

- for the type of partition described for each grid in section 2;
- for high-resolution grids on both sides (GSII),
- for both components running on 1000 cores (NPxx1000)
- for 2 coupling fields (NC2)
- for the shared and disjoint layouts (LayS, LayD)
- for the sequential and concurrent schedules (SCseq, SCcon)

The total number of runs involved for each couple of grids is therefore 4 (2 layouts x 2 schedules).

3.2.4 Impact of number of coupling fields

Each coupled model will be run:

- for the type of partition described for each grid in section 2;
- for high-resolution grids on both sides (GSII),
- for both components running on 1000 cores (NPxx1000)
- for 2, 20, 200, 2000 coupling fields (NC2, NC20, NC200, NC2000)
- for the disjoint layout (LayD)
- for the sequential schedule (SCseq)

The total number of runs involved for each couple of grids is therefore 4 (4 numbers of coupling fields).

The total number of runs for a couple of grids used to test the impact of all coupling characteristics described above would therefore be 21 (or 26 for selfgenreglatlon-selfgenreglatlon). But one run should be done more then once to check the reproducibility of

the results; if we suppose that each run is performed 3 times, this gives 63 runs per couple of grids (or 78 for selfgenreglatlon-selfgenreglatlon).

3.3 Other benchmark characteristics

3.3.1 Timings

A standard timing library will be developed and timings will be implemented for:

- the full initialisation including the coupling initialisation;
- the first time step;
- the standard deviation, maximum, minimum and average over 98 additional time steps (so to give stable results);
- the last time step

3.3.2 Mechanisms for correctness checking

The benchmark test cases will also propose mechanisms for checking correctness, as well as completion, of the coupled system. For example, coupling fields will be defined at each grid point as an analytical function based on the spatial position of the grid point and the value of the field will be incremented at each timestep; the error, defined as the difference between the value of a coupling fields at the end of the run and the value of the analytical function, incremented by the number of time steps will be calculated.

4. Practical implementation of benchmark test cases

The partners now commit to implement the coupling between components running on the following grids with the following technologies³ (the one currently implemented at the time of writing the milestone are identified):

4.1 selfgenreglatlon-selfgenreglatlon:

- OASIS : CERFACS
- OpenPALM : CERFACS
- ESMF : STFC (implemented)
- MCT : UNIMAN
- YAC: DKRZ (implemented)

4.2 stretchlatlon-selfgenreglatlon:

- OASIS : CERFACS
- OpenPALM : CERFACS
- ESMF : MetO

4.3 icoso-icoso :

- OASIS : CERFACS (implemented)
- OpenPALM : CERFACS (implemented)
- YAC: DKRZ (implemented)

4.4 cubedsphere-selfgenreglatlon :

- OASIS : MetO (implemented)
- ESMF : MetO (implemented)

³ Note that this has evolved slightly since M10.1 as the test cases and related requirements in term of work and computing time are now better defined, as the gaussred grid was not included at the time, and as DKRZ joined the task with the YAC coupler not initially included.

- YAC: DKRZ (implemented)

4.5 Stretchlatlon-icosa

- OASIS : CERFACS
- OpenPALM : CERFACS (implemented)

4.6 Stretchlatlon-gaussred⁴

- OASIS : CERFACS (implemented)
- OpenPALM : CERFACS (implemented)

5. Perspectives

The implementation of the benchmark test case described in section 4 will be completed before the summer. The resulting benchmark suite will be released to the community, including model developers, developers of coupling technologies and developers of computer systems. Anyone one interested in running the test cases described in 3.2 or other test cases will be welcome to do so.

Partners involved in WP10 T3 commit to run the test cases described in 3.2.1. to test the impact of the number of cores per component combined with the impact of the grid sizes on at least 3 different platforms (Bullx Occigen for CERFACS⁵, Cray XC40 for the MetOffice, and a third one with different architecture for STFC and UNIMAN) for the 4.1 selfgenreglatlon-selfgenreglatlon combination with PTnn on both sides for the 5 coupling technologies.

CERFACS also plans to run the same test-cases but for the PTn1 decompositions.

CERFACS also plans to run with OASIS and OpenPALM only the 3.2.1 test cases for other couple of grids (i.e. 4.2 stretchlatlon-selfgenreglatlon, 4.3 icoso-icoso, 4.5 stretchlatlon-icoso and 4.6 stretchlatlon-gaussred combinations), and MetO for the 4.4 cubedsphere-selfgenreglatlon combination. The other test cases 3.2.2, 3.3.3 and 3.2.4 will be run only if time and computing resources permit.

Results will be analysed and presented to the community, for example, at the future IS-ENES coupling workshop planned in March 2017 in Princeton, at the EGU and/or through publication in academic publications.

⁴ As mentioned above, these coupled systems involving the Gaussian Reduced grid are not based on standard stand-alone component but are included here as pre-coded coupled toy models based on these grids already exist and will be used to report on OASIS and OpenPALM performances.

⁵ CERFACS asked for ~500 Kcore-hours on Bullx Occigen thin nodes through GENCI national campaign.