Multi-model multi-member integrated simulation:
Computing performances of a demonstrator
and other single high-resolution climate models

E. Maisonnave[#], D. Manubens*, U. Fladrich°, Ø. Seland[~]
[#] CERFACS, Toulouse, France
* BSC, Barcelona, Spain
° SMHI, Norrköping, Sweden
[~] Met.no, Oslo, Norway

Table of Contents

Abstract

A set of scripts and configuration files necessary to launch an ARPEGE-NEMO/EC-Earth multi-model multi-member high-resolution (M4-HR) experiment are developed and the existing single model workflow adapted. The Autosubmit tool ensures a remote management of simulations performed on the BSC MareNostrum III supercomputer. The pre-existing single model launching scripts are easily parametrised to let Autosubmit cope with difficulties related to remote connection and multi-member simulation handling. The new developments include a final analysis that delivers computational metrics that helps to evaluate models and whole M4-HR experiment performances.

Executive Summary

Objective

The goal of IS-ENES work package 9/JRA1 is testing and evaluating performances of an Multi-Model Multi-Member High-Resolution (M4-HR) Earth System Model on a single Tier-0 HPC system. The large amount of resources needed to deploy the M4-HR configuration qualify this climate community tool for the PRACE tier-0 facilities. At the same time, this community platform is the suitable place to test and share all developments that facilitate the handling of HR models: the improvements realised for the coupler, I/O subsystem and job submission tools are evaluated on this M4-HR platform.

Results

Two high resolution models (ARPEGE-NEMO and EC-Earth) were parametrised to perform an ensemble of high resolution coupled simulations (*demonstrator*). Up-to-date versions of auxiliary libraries and tools (OASIS coupler, XIOS I/O subsystem, CDO post processing tool and Autosubmit job controller) were included in this demonstrator. Its performances show that their integration in the M4-HR context, were effectively completed at the end of the project. The metrics we used to evaluate computing performances are based on the emerging standard for computational performance of climate models, while focusing on those used to evaluate the performances of the initial configurations at the beginning of the project. Complementary to the demonstrator, other laboratories involved in the work package completed the developments of their HR models and provide the corresponding computational performance. The results of this analysis prove their technical capability to perform the CMIP6 (HiResMIP) exercise.

Perspectives

Regarding coupling and I/O management, the authors strongly encourage the community to converge to standards that could contribute to federate all EU laboratory efforts. Another crucial disposal that must be enhanced is the availability of shared Tier-0 supercomputers at EU level. The important effort of standardisation for computational performance metrics, proposed by GFDL (Princeton, USA) and intensively tested and enhanced by our work, needs to be extended and consolidated. It is now necessary to promote measurements and reports of model performances at larger scale and propose an on-line platform to share the results.

Introduction

Two of the five high-resolution climate models (ARPEGE-NEMO and EC-Earth) participating in this IS-ENES2 work package [1] have been ported to the BSC MareNostrum III (MN3) supercomputer[1]. Their existing workflows[2] are adapted to fit the machine and its environment requirements (system architecture, file system, directory structure, scheduler, etc.). The EC-Earth model version 3.2 includes all technical and physical improvements implemented since the beginning of the project, which is not the case for ARPEGE-NEMO (v5)/CNRM-CM6, due to the lack of synchronization between the CMIP6 HiResMIP [2] project and IS-ENES2. Ideally, all five participating modelling groups should have been involved in the multi-model multi-member integrated experiment. However, the parallel development of the CMIP6 related HiResMIP project, involving the same models but with a different time schedule, does not allow to set up, in time, the required and scientifically up-to-date high-resolution (HR) models. Consequently, we choose to develop a demonstrator, including less models to facilitate its set up, but that gathers some improvements which matter from the perspective of computational performances. To give a broader view of HR model developments in the ENES community, enhancements of CMIP6-HiResMIP models not included in our demonstrator are evaluated in a separate paragraph.

The two demonstrator models take benefit of WP9 task 2 efforts on coupling performances [3],[4]. Both models include also functionalities offered by the I/O servers developed in task 3 [5],[6]. The major contribution of WP9 activity emphasised by the demonstrator deals with task 5: For a remote control of several simulations made with different models, an efficient job control is required [7,8].

It has to be mentioned that the set up of another demonstrator was proposed a few months ago. This demonstrator had a broader scientific purpose (cyclone tracking), and was including a larger set of WP9 models (CESM-NEMO, ARPEGE-NEMO, EC-Earth, and NorESM). The lack of computing resources, due to temporary restrictions on PRACE allocations, made preferable the solution described below.

Objectives

Two high-resolution models are parametrised to perform an ensemble of high-resolution coupled simulations. The large amount of resources needed to deploy the M4-HR configuration qualify this climate community tool for the PRACE tier-0 facilities. At the same time, this community platform is the suitable place to test and share all developments that facilitate the handling of HR models. The purpose of this demonstrator is technical: it reproduces the realistic behaviour of a workflow producing the data of two high-resolution models. The output data is converted to the standard format required for analysis (NETCDF), but the scientific results, even though meaningful, are not analysed. This framework is adequate to estimate how easily an M4-HR can be leaded on our platform; the lack of real geophysical studies allows to reduce the simulation length (2 months) and the number of members of the ensemble (2) to a strict minimum, compatible with our reduced computing resources. Nevertheless, the demonstrator is designed to show the same computing performances as realistic experiment configurations

---

1 https://www.bsc.es/marenostrum-support-services/mn3

2   We call "workflow" the various scripts (including compilation) necessary to launch the FORTRAN executables included in a climate model

and to evaluate the improvements made during the project. The metrics we used are based on the emerging standard for computational performances of climate models [9], while focusing on those used to evaluate the performances of our initial configurations [1]:

simulation speed. The HR climate models includes two main components (ocean and atmosphere). NEMO (ocean model, ORCA025-L75 grid) is used by both models. ARPEGE (atmosphere) performs its calculations on a T359-L31 grid (50km), IFS[3] on T511-L91. The models are thus used at similar resolutions as required for the HiResMIP exercise. Their speed, measured in simulated years per day (SYPD) is realistic. This gives a good idea of the quality of algorithmic improvements, particularly those related to parallelism. The **memory bloat** is another quantity that helps to evaluate the model capacity to fit the machine requirements and to enhance its performances.

coupling cost: both models are using the last version of the OASIS coupler (OASIS3-MCT, V3), which includes performance analysis functions. Consequently, the coupler performances and the efficiency of model load balancing, as described in [3], are quantities that can be evaluated with our demonstrator. Coupling costs are provided in a dedicated deliverable (D9.2)..

data output cost and data intensity: the XIOS I/O server is a common component of the two coupled models. With an appropriate choice of output fields, a measurement of these two quantities in our demonstration run gives a good idea of the performances, in a HR context, of the I/O library developed in task 3. Performances of the I/O servers XIOS and CDI-pio are fully described in deliverable D9.4.

actual simulation speed: this metric is aggregating (i) the performances of the various operations included in the workflow, (ii) the performances of the job manager that organises this workflow for several chunks[4] of several ensemble members of simulations made with two models and (iii) the supercomputer behaviour during the simulation (machine load, bandwidth, scheduler policies, etc.). This last contribution makes the metric more random and its normalised evaluation would require a broader experiment. However, this metric gives a first guess about how fast a realistic M4-HR can be lead on the platform. It also gives a lower limit[5] of the performances of Autosubmit management during the embarrassingly parallel M4-HR experiment. The quality of the job control tool, developed in task 5, could then be evaluated.

We took benefit of these metrics to measure the evolution of our model performances and evaluate how efficiently an M4-HR experiment can be carried out. The results will be provided in the final section of this document. One metric is missing in this standard list: the one that could measure how efficiently such demonstrator can be set up on a Tier-0 system. This also will be discussed in the final section.

In parallel to the demonstrator development, the other laboratories involved in the work-package keep developing their HR models, following the CMIP6 (HiResMIP) requirements. In the next section, the performances of their last model version are compared to the measurements done at the beginning of the project.

---

3 EC-Earth components: IFS (atmosphere) and NEMO (ocean)

4 Every job or chunk, is described in a shell script, submitted to the scheduler and parametrised to perform one month of climate simulation

5   Although Autosubmit better organizes scheduling of large set of jobs included in a multi-model ensemble experiment, its performances are limited by the machine load and the scheduler policy.

Results

Development of the M4-HR demonstrator

The demonstrator development started from two independent sets of scripts that usually drive ARPEGE-NEMO and EC-Earth simulations. A preliminary work, based on previous porting of these models on the targeted supercomputer (MN3, BSC), consisted in testing these scripts, together with the model executables and initial data. Nowadays, the availability of the FORTRAN Intel compilers on almost every supercomputers facilitates the porting phase. The standard environment installed on MN3 (working directory and scheduler) also contributed to speed up this preliminary work. The low amount of produced data avoided to organize the transfer of huge quantities across the network.

An important characteristic of our demonstrator is that the simulations are driven from a workstation located outside the supercomputing centre (CERFACS and SMHI workstations). For this, Autosubmit must be installed locally and parametrised considering (i) the kind of ensemble experiment we want to process, (ii) the various elements included in the workflows (pre/post processing phases) and (iii) the characteristics of the local-workstation-to-supercomputer communication link. Moreover, the two initial sets of scripts are modified to give to Autosubmit the capacity to drive together the two ensemble members (see appendix "Autosubmit experiment configuration").

There have been many improvements in Autosubmit since the IS-ENES2 milestone MS92 was achieved ("Further developments of Autosubmit") as described in [7]. The major 3.0 release includes many improvement. The most relevant for this work are :

- Workflow flexibility enhancements (possible to remove dependencies, add new jobs, etc)
- New feature to download projects from Git, SVN and local path
- New command line interface
- Better error management, proper logs and clear console output
- New communication method based in a single SSH connection
- New platforms available to run (possible to run in fat-nodes using the Slurm scheduler, CCA using ecaccess client, etc)
- New naming convention for experiments
- New options to create dummy and test experiments
- Allow jobs to execute at user defined frequency and many others implemented from 3.0.0 to 3.7.6 (latest release)[6]

Furthermore, an assessment has been carried out and documented in deliverable D9.3 ("Assessment report on Autosubmit, Cylc and ecFlow"), in which the capability of Autosubmit to run multi-model multi-member experiments has been demonstrated.

In the appendix, we describe how to configure Autosubmit to be able to launch our M4-HR experiment on the MN3 supercomputer. From the BSC FTP site, two tar files can be download to modify local Autosubmit parameter files and provide models/corresponding workflow

---

6   https://earth.bsc.es/gitlab/es/autosubmit/blob/master/CHANGELOG

scripts to be installed on MN3.

Demonstrator performance results

The implementation of the demonstrator on MN3, lead by two scientific developers from modelling groups and one from the Autosubmit team, was relatively easy, taking about one week for porting, plus a couple of days to install and set up Autosubmit locally, and adapt the workflow to Autosubmit requirements. During this work, we strongly appreciated the possibility to access supercomputing resources and to use at BSC, without restriction, two models from two distinct communities. Regular working sessions and discussions between the developers also strongly speeded up the implementation phase. This results from the tight collaboration between modelling groups and the computing centre, that the two IS-ENES(1 and 2) European projects contributed to strengthen during the last eight years. We also could measure how important can be the handling of common tools, (OASIS, XIOS, Autosubmit, NEMO, etc.) developed during the project: a pre-existing knowledge related to these tools was crucial to identify the origin of issues during the implementation of this complex system, and get results rapidly.

Two month long simulations were launched on MN3, with two member ensembles of two models. Table 1 summarises the performance results for the demonstrator, and lists previous results for comparison. Measurement details are described in Balaji et al. 2017.

Overall, the table shows consistent performances between the current (columns 1 and 2) and earlier experiments (columns 3 and 4). Nevertheless, both experiments differ from:
- the hardware (supercomputer) used: CURIE and MN3 include rather similar processors, but EKMAN provides previous generation CPU.
- the software: the EC-Earth model was upgraded since 2014 (from 3.1 to 3.2 version). Component/libraries included in both models/workflows were enhanced by IS-ENES2 WP9.
- the production mode: 2014 experiments were lead at optimum scalability, which is not the case with our demonstrator.
- the scheduling type: Autosubmit allows the concurrent processing of both ARPEGE-NEMO and EC-Earth climate models.

The comparison of the two sets of experiment necessarily integrates the effects of the four differences. Some are supposed to have a beneficial effect on overall performances (hardware upgrade, sophisticated job controller), some others could downgrade them (production mode). Changes on software could diversely affect the performances. However, the demonstrator allows to concurrently run the two models at performances comparable with the one initially measured in 2014. Most of the performance metrics of the standalone models are not affected by the demonstrator setup, nevertheless, some of the results need further explanation.

Model name	Demonstrator		2014 version	
	ARPEGE-NEMO	EC-Earth	ARPEGE-NEMO	EC-Earth
Model version	5	3.2	5	3.1
Speed (SYPD)	3.5	1.2	5.0	2.6
Actual speed (ASYPD)	0.11		1.60	n/a
Computing cost (CHPSY)	2783	14521	5190	10353
Energy cost (JPSY)	50.8	294.5	n/a	n/a
Memory bloat	12.8	19.0	n/a	n/a
Data Output cost (%)	1.50	1.12	1.40	n/a
Data intensity (GB/CH)	0.0053	0.0045	0.3	n/a
Platform name	MARE NOSTRUM III	MARE NOSTRUM III	CURIE	EKMAN
Total core number	48,896	48,896	80,640	9,600
Platform processor	Sandy Bridge	Sandy Bridge	Sandy Bridge	Opteron
Clock cycle concurrency	8	8	8	4

Table 1: Computing performance of ARPEGE/NEMO and EC-Earth, measured with the demonstrator (column 1 and 2) and collected during 2014 previous exercise (column 3 and 4)

The simulation speed (SYPD) for both models is lower in the demonstrator experiment than in their standalone runs, as measured in 2014. The most important reason for this is the resources that were available for the M4-HR experiment: In order to fit the multi-model experiment on to the MN3 machine in normal operation (i.e. not dedicated), only moderate parallelism was used for both models. This reflects the fact that the models were configured for good throughput, not optimum speed. This is also seen from the computational costs (CHPSY) for ARPEGE-NEMO, which is considerably lower for the M4-HR run than previously. The increased computational costs of EC-Earth can be attributed to the newer version of the model, which is computationally more complex. A substantial part of the differences in EC-Earth computational cost and speed is also explained by the different platforms that were used for the current and earlier experiments.

A major implication of the M4-HR set-up is the low actual simulation speed (ASYPD). It is measured once for the complete multi-model experiment, i.e. the timing is started when the first simulation chunk of the first model is submitted, and stopped once the performance has been computed after the post-processing of the last chunk of the last model has completed. The main difference between simulation speed and actual simulation speed is due to queue waiting time, and not from differences in other workflow component performances like data movement. It turns out that the batch system has basically sequentialised the work flow of the M4-HR experiment (EC-Earth and ARPEGE-NEMO computations can not actually be processed simultaneously). One explanation could be that the fair-share policy on the system will not favour multiple jobs of any one user if the system is busy. For the rest of the

performance metrics, it is worth to notice that the results presented in Table 1 are consistent with the data published about the two models in [9].

Up-to-date versions of auxiliary libraries or tools (OASIS coupler, XIOS I/O client/server, CDO post processing tool and Autosubmit job controller) are included in our demonstrator. A detailed analysis of their performances is given in IS-ENES2 deliverables D9.2 (HR ESM performance resulting from OASIS updates), D9.3 (Assessment report on Autosubmit and Cylc), D9.4 (Benchmark of the two I/O servers) and D9.5 (CDO performance analysis). The demonstrator performances show that their integration on HR models, in a M4 context, is efficiently completed at the end of the project.

Performance analysis of other HR models

Modelling groups involved in this work-package contributed to the metrics definition and participated to the performance measurements at the beginning and the end of the project. CMCC (CESM-NEMO) and MetOffice/NCAS (HadGem-GC) described their HR configuration. Met.no proposed its ESM at lower resolution, which exhibits a higher "complexity" than the first two.

Model name	CESM-NEMO	HadGEM-GC	NorESM
Model version	1.2.2	3	2
Parallel mode	Throughput	Throughput	Throughput
Speed (SYPD)	0.86 (*0.31*)	1.0 (*0.6*)	7 (*15.4*)
Actual speed (ASYPD)	0.04 (*0.06*)	0.56 (*0.6*)	5
Computing cost (CHPSY)	59,100 (*163,000*)	6,504 (*64,000*)	4,225 (*1,100*)
Energy cost (MJPSY)	7.87	0.33	n/a
Memory bloat	9.1	n/a	37
Data Output cost (%)	1 % (*25 %*)	n/a	6 %
Data intensity (GB/CH)	0.0014	n/a	0.006
Resolution (MGP)	120 (*115*)	180 (*178*)	12 (*50*)
Complexity	103 (*176*)	66	236 (*135*)
Platform name	**ATHENA**	**ARCHER**	**VILJE**
Total core number	7,712	118,080	22,464
Platform processor	Sandy Bridge	Ivy Bridge	Sandy Bridge
Clock cycle concurrency	8	8	8

Table 2: Computing performance of WP9 models, measured at the end of the project and during the previous exercise (2014, quoted *italic*).

The performances shown in table 2 still reveals the difficulty to perform simulations at a speed compatible with long experiments (hundreds of years). A reasonable limit for such experiment can be assessed at 3 SYPD (about 100 years per month) in throughput (production) mode. Only a single (non HR) configuration rises above this limit. Another clear concern about speed

is disclosed by comparing speed and actual speed: except for "ARCHER", machines poorly succeed in performing calculations at maximum speed. The quality of the other metrics is difficult to estimate, mainly because most of them are measured for the first time by the modelling groups. This is one of the most valuable result of the project: a finer measurement of model performances becomes more and more widespread in our laboratories.

Perspectives

The demonstrator emphasised the possibility for two different groups to operate together their respective high-resolution climate models in a unified environment, after a relatively light implementation effort. It also reveals the limits of the exercise. The lack of interest in common auxiliary components (e.g. OASIS and XIOS) could explain the limited number of models included in the demonstrator. The existence of two different job controllers in the community limits the interest for other groups that are more familiar with the other software (Cylc). But we assumed that a demonstrator using the two job controllers together would have been useless or even less efficient. A release of Autosubmit, through GNU license will be made available to the ENES community in MS99. The software is already available from PyPi (https://pypi.python.org/pypi/autosubmit) and can be easily installed[7] on a GNU/Linux platform. The constraint of high resolution is another explanation (most of the laboratory only maintain low resolution configurations) to the lack of contributors to the demonstrator.

Regarding coupling and I/O management, the authors strongly encourage the community to converge to standards that could contribute to federate all EU laboratory efforts. The case of code coupling is particularly worrisome: after decades of standardisation and interoperability, the community seems to change its good practice and follow a different path with different tools (XIOS-coupler, YAC, ESMF, etc.). Innovation must probably be paid for with the coexistence of redundant tools, but we hope that this phase will rapidly lead to an emerging new standard, which is an essential element to maintain a large exchange in a free movement of models within the EU.

Another important disposal that allows to quickly set up this demonstrator is the availability of a shared supercomputer resources at EU level. The authors wish that PRACE2 will rapidly provide the necessary resources, considering that the Tier-0 machines are the only one able to host M4-HR simulations and preliminary tests such as the present one.

The important effort of standardisation for computational performance metrics, proposed by GFDL (Princeton, USA) and intensively tested and enhanced by our work needs to be extended and consolidated. The coming CMIP6 exercise must be the right occasion to extend measurements to low resolution and ESM configurations in the laboratories of our IS-ENES community (Met.no, CMCC, MetOffice/NCAS, DKRZ/MPI-met, IPSL, Météo-France/CERFACS, EC-Earth) and to involve more laboratories around the world (Australian and Japanese laboratories have already been contacted). The IS-ENES activity related to CMIP6 meta data definition (ES-DOC) already includes computing performance metrics in their standards and procedures to declare CMIP6 model performances were developed. It is now necessary to promote measurements and reports of model performances at larger scale and propose an on-line platform to share the results.

---

7 http://autosubmit.readthedocs.io/en/latest/installation.html

Acknowledgements

References

[1] Fladrich, U., Maisonnave, E., 2014: A new set of metrics for the computational performance of IS-ENES Earth System Models ,Technical Report, **TR/CMGC/14/73**, SUC au CERFACS, URA CERFACS/CNRS No1875, France

[2] Haarsma, R. J., Roberts, M. J., Vidale, P. L., Senior, C. A., Bellucci, A., Bao, Q., Chang, P., Corti, S., Fučkar, N. S., Guemas, V., von Hardenberg, J., Hazeleger, W., Kodama, C., Koenigk, T., Leung, L. R., Lu, J., Luo, J.-J., Mao, J., Mizielinski, M. S., Mizuta, R., Nobre, P., Satoh, M., Scoccimarro, E., Semmler, T., Small, J., and von Storch, J.-S.: High Resolution Model Intercomparison Project (HighResMIP v1.0) for CMIP6, Geosci. Model Dev., 9, 4185-4208, doi:10.5194/gmd-9-4185-2016, 2016.

[3] Maisonnave, E., Caubel, A., 2014: LUCIA, load balancing tool for OASIS coupled systems ,Technical Report, **TR/CMGC/14/63**, SUC au CERFACS, URA CERFACS/CNRS No1875, France

[4] Maisonnave, E., Coquart, L., Fladrich, U., Fogli, P.-G., Hill, R., Seland, O., Voldoire, A., 2016: IS-ENES model interface updates for compliance with new OASIS coupler Technical Report, **TR/CMGC/16/29506**, CECI, UMR CERFACS/CNRS No5318, France

[5] Meurdesoif, Y., 2016: I/O server release, IS-ENES2 Milestone MS9.8

[6] Maisonnave, E., 2016: Realistic Output without real model: an XIOS client toy Working Note, **WN/CMGC/16/214**, CECI, UMR CERFACS/CNRS No5318, France

[7] Maisonnave, E., 2014: A CNRM-CM6 monitoring using Autosubmit ,Working Note, **WN/CMGC/14/39**, SUC au CERFACS, URA CERFACS/CNRS No1875, France

[8] Manubens-Gil, D., Vegas-Regidor, J., Prodhomme, C., Mula-Valls O. and Doblas-Reyes, F. J., 2016: Seamless management of ensemble climate prediction experiments on HPC platforms, International Conference on High Performance Computing & Simulation (HPCS), Innsbruck, 2016, pp. 895-900. doi: 10.1109/HPCSim.2016.7568429

[9] 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., and 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

Appendix

Autosubmit experiment configuration

While the whole simulation (from pre- to post-processing) takes place on the supercomputer, the job control is performed locally. This implies that some elements of the workflow must be duplicated (locally and on the supercomputer, see next section) or that some locally defined parameters must be exported, by Autosubmit, to the remote machine.
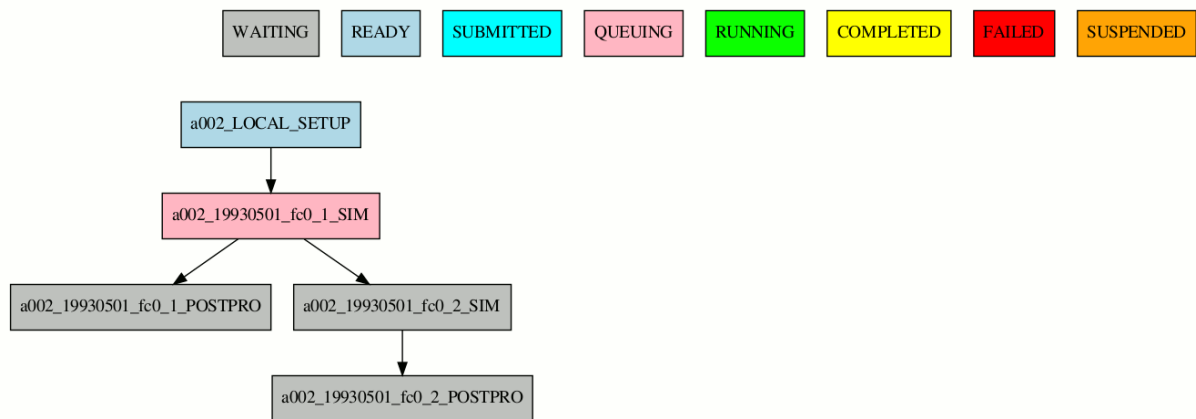

Figure 1: Sequence of workflow elements of an ARPEGE-NEMO single member simulation, as driven by Autosubmit. Colour definition for each possible status of the elements.

Autosubmit experiments are configured by parameters in three main configuration files:

expdef.conf
In this file, the usual parameters that characterise ensemble runs can be set: the list of members, the number and length of chunks, start date, etc. Path to the original shell scripts that leads the two ARPEGE-NEMO and EC-Earth simulations is also declared here.

jobs.conf
Here, we define the sequence, dependencies and characteristics of each element included in our workflow, like:
- the initialisation phase: a copy of the original shell scripts is done from the local repository to MN. This operation is done once per model. It is suffixed "_SETUP" in Figure 1.
- the simulation (including the launching of the main MPI parallel program, suffixed "_SIM") and the post-pro phases (suffixed "_POSTPRO"). These two operations are performed sequentially as described in Figure 1. Members are executed in parallel, and chunks are executed sequentially.
- A final operation is performed at the very end of the experiment (suffixed "_PERF_METRICS" in Figure 2). This module calculates and provides information about computing performances.

platforms.conf
This file contains all parameters for connection with the supercomputer (host name, scheduler, remote path, etc.)
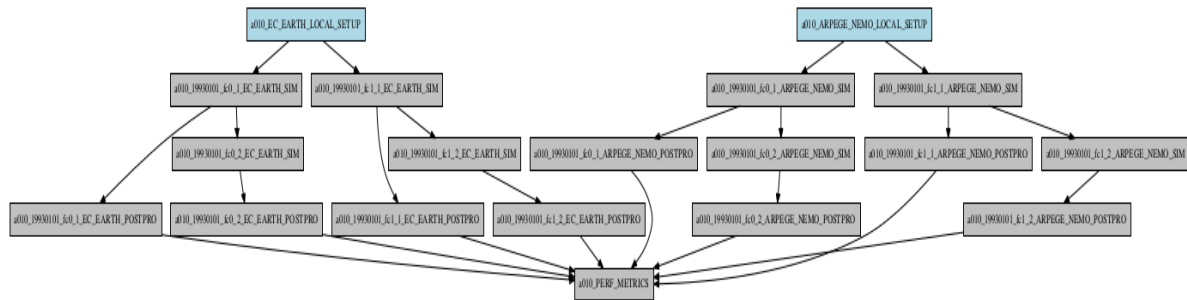

Figure 2: Sequence of workflow elements of an ARPEGE-NEMO/EC-Earth multi-member simulation, as driven by Autosubmit.

A fourth configuration file (`autosubmit.cfg`) is created, defining variables (resources per component) that must be parameters of the original workflows and need to be changed locally.

These user defined variables, and all Autosubmit parameter described in the previous section, must be transferred into the original workflow scripts. To do so, the developer just needs to replace the corresponding variables of the original scripts with variables pre-processed by Autosubmit[8].

The original scripts, if written for a single member usage, must be changed to take into account the rank in ensemble for input file selection, running/output directory naming, output file naming, etc.)

Finally, to be able to provide a standard and unified set of computational performances associated to our experiment, we wrote a new shell script (`perf.sh`) that takes information from the Autosubmit "log" files [`%HPCROOTDIR%/LOG_%EXPID%`, e.g.]. Metrics described in [9] are provided, e.g. simulation speed, actual simulation speed, energy cost, computing cost, memory bloat, data output cost and data intensity. Coupling cost (using the LUCIA tool [3]) could be included in a future version.

M4-HR demonstrator user guide

The M4-HR demonstrator consists of

- the ARPEGE-NEMO model executables,
- the EC-Earth model executables,
- the set of EC-Earth initial data files,
- the M4-HR runtime environment files, and
- the Autosubmit experiment configuration files.

8 All variables locally defined in .cfg files can be recovered remotely if inserted with a special syntax (prefixed and suffixed with the "%" sign)

All of these file sets have been either packaged as tar files or provided at the file system of the MN3 supercomputer. To run the demonstrator, the M4-HR runtime environment and the Autosubmit experiment configuration files have to be downloaded to the local computer that is used to control the M4-HR experiment. The main steps to prepare, run, and assess an M4-HR experiment using the demonstrator are:

1. Install Autosubmit on the local computer

2. Initialise a new Autosubmit experiment

3. Overwrite the Autosubmit experiment configuration with the provided files

4. Adapt the Autosubmit experiment configuration files for the actual user environment

5. Create the Autosubmit M4-HR experiment

6. Launch the M4-HR experiment

7. Assess the performance results

The individual steps are explained in more detail below, as far as they are special for the M4-HR demonstrator. For more general information on how to use Autosubmit, references are given.

Autosubmit installation

Detailed installation instructions can be found at the Autosubmit web site (`http://autosubmit.readthedocs.io`). Autosubmit can be installed inside a virtual machine. An important requirement for using Autosubmit is passwordless access, using SSH keys, from the local computer to MN3. This implies an account on the supercomputer and permissions to transfer files between local and remote file systems.

The M4-HR experiment does not require a special version of Autosubmit. Both 3.6.0 and 3.7.5 version were tested at CERFACS and SMHI. For a complete description of Python libraries required (dependencies) and Autosubmit package installation on your local computer, users can refer to the Autosubmit user guide[9].

Initialise, configure, and create the Autosubmit M4-HR experiment

As a first step, Autosubmit is asked to provide an experiment name and the corresponding set of directory and database space with the following command:

```
autosubmit expid –H marenostrum3 -d "your_description"
```

A new directory ("aNNN", named $EXPID, with integer NNN increasing with each new experiment) is created in the "autosubmit" directory. Under the "autosubmit/ $EXPID/conf", directory, the newly created Autosubmit configuration files have to be replaced by the package "`ftp://bscesftp.bsc.es/M4HR/conf_a011.tar.gz`" and modified as follows:

```
cd autosubmit/conf; tar zxvf conf_a011.tar.gz
```

All configuration file names have to be changed according to the $EXPID. In addition,

9 http://autosubmit.readthedocs.io/en/latest/installation.html

"autosubmit_aNNN.conf" and "expdef_aNNN.conf" file content also changes with your $EXPID. In "platforms_$EXPID.conf", change USER and PROJECT variable with your settings

Then, the workflow script set must be copied into the local directory from the BSC FTP site (ftp://bscesftp.bsc.es/M4HR/M4HR.tar.gz)

```
cd path_of_local_directory; tar zxvf M4HR.tar.gz
```

The local file are now ready to be declared to Autosubmit:

```
autosubmit create $EXPID
```

In the autosubmit/plot directory, you can graphically visualise the initial status of your experiment. You can now start your simulation.

Launch an M4-HR experiment

The M4-HR experiment is launched from the local computer with this command:

```
autosubmit run $EXPID
```

A remote check of the experiment status is possible through the following instruction. A pdf file is created in your autosubmit/plot subdirectory (different other graphical format are available)

```
autosubmit monitor $EXPID
```

If something has to be corrected in the local workflow scripts, a re-submission can be operated like follows:

```
autosubmit refresh $EXPID
autosubmit setstatus $EXPID -t WAITING -fs Any -s
autosubmit run $EXPID
```

Assess the performance results

A successful simulation produces a report on computing performances, which can be assessed locally in:

```
autosubmit/$EXPID/tmp/LOG_[$EXPID]/[$EXPID]_PERF_METRICS_[job_ID].out
```

In addition, the standard statistics produced by Autosubmit (related to time spent in queue or in workflow operations, number of fauilures, etc) are produced with the command:

```
autosubmit stats $EXPID
```
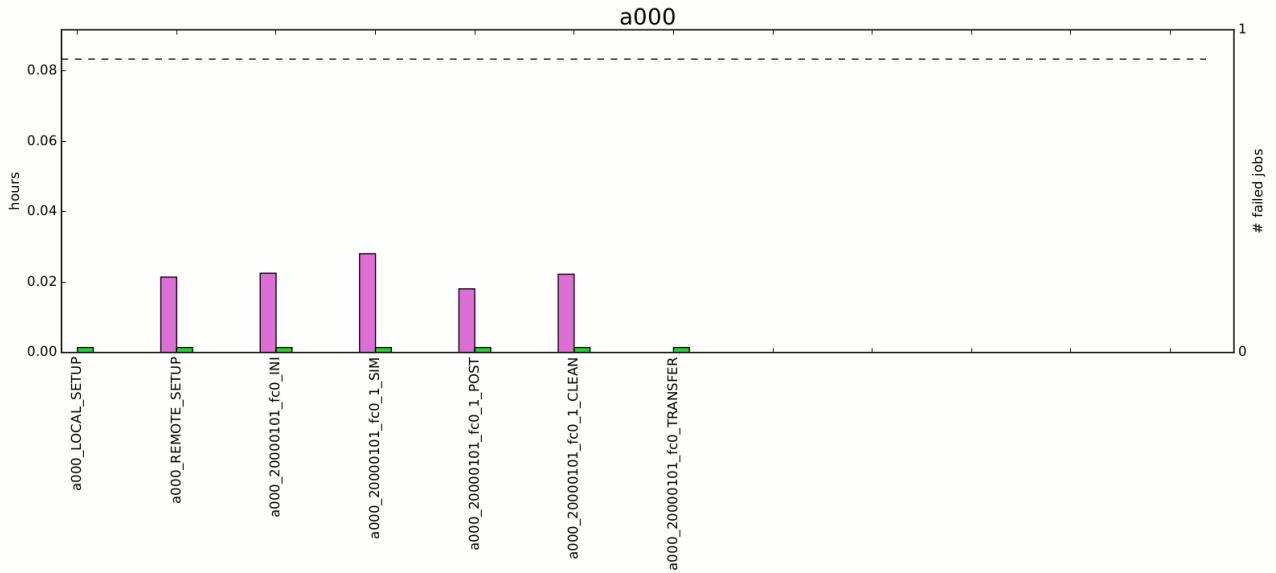
STATS - a000

Figure 3: Example of statistics related to M4-HR test experiments,
as provided by Autosubmit.