An auto climate model continuous integration strategy

Pablo Echevarria[1] and Miguel Castrillo[1]

[1]Computational Earth Sciences at the Earth Sciences department

Barcelona Supercomputing Center

In the Earth Sciences department from the Barcelona Supercomputing Center (BSC), the Computational Earth Sciences group developed a robust and efficient strategy to synchronize the latest release of the EC-Earth model, used operationally in the department. This strategy provides the outcome of always being able to run the model version, an achievement for both the computational and the scientific point of view.

INTRODUCTION

Very costly simulations are needed to predict climate change, or to produce a seasonal or decadal weather forecast. On the other hand, there is a wide spectrum of different products which cover the variety of experiments performed in the department, using different resolutions, enabling different components of the Eath System model, or adding processes like in data assimilation, etc. EC-Earth is a numerical weather prediction (NWP) coupled model made up from different components as IFS (atmosphere) + NEMO (ocean) + LIM (ice), etc see fig: 1.

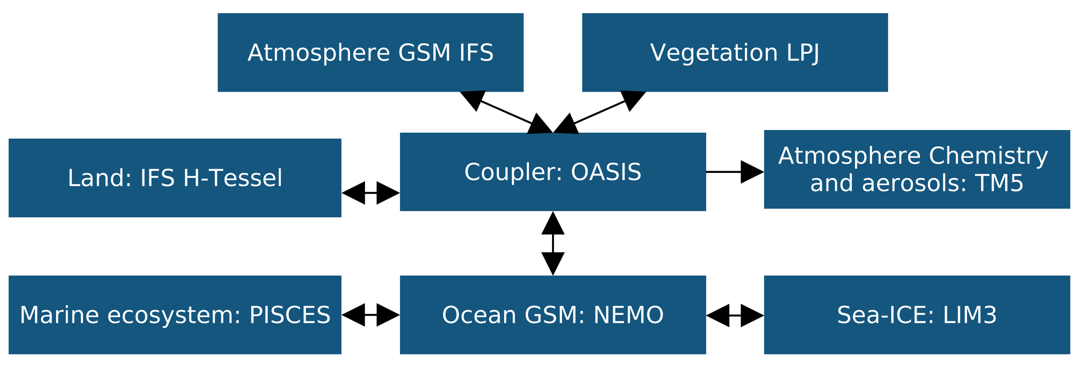

Fig. 1: Ec-Earth components

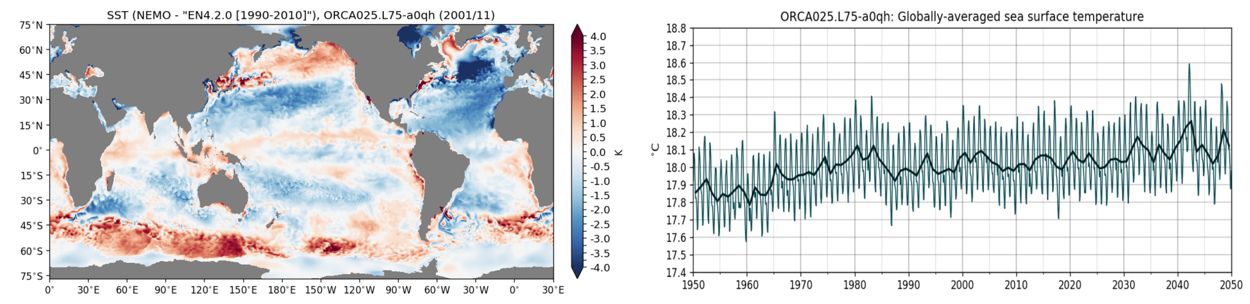

Fig. 2: The model output could be the sea surface temperature average or a map for the same variable

The main High Performance Computer (HPC) in the BSC is MareNostrum4 (13.7 PFlops, 3500 nodes 2 x Intel Xeon Platinum 8160 CPU with 24 cores each 2.10GHz). Simulations can take up to a 120 hs of intensive computing using more than 1500 cores, see Table 1. This fact may entail a specific layer of software to allow divide the task in different chunks, since the maximum time in a regular queue is 24 hs. Other taks are to compute the pre-processing, deploy, compile, post-processing and visualization.

	High resolution	Low resolution
Cores	1500	1250
4 months (seasonal)	1 h	20 min
1 year	12 hs	4 hs
10 years	120 hs	40 hs

Table 1: Computing time in a couple run simulations

With the aim of helping in this task we use a workflow manager which is Autosubmit. The package uniting EC-Earth + Autosubmit is called Auto-EC+Earth.

AUTO-ECEARTH

Auto-EC-Earth is an Auto-model, and provides the flexibility of the modularity and at the same time it helps to monitor the complete simulation workflow, keeping track of eventual distortions of the single processes within the whole simulation, i.e pre-processing, filtering, post-processing and visualization see fig 3. All these tasks require a new layer of software [see https://goo.gl/VgQpD2].

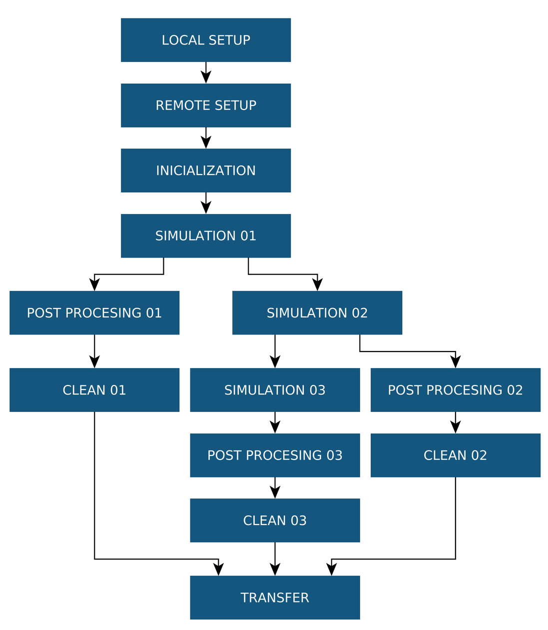

Fig. 3: Auto-ecearth example of the workflow

THE PROBLEM

Developing new features or only keeping the interface updated in this scenario could be challenging. Furthermore, a new complexity added here is that Ec-Earth is developed by a consortium of several institutions and more than 400 developers. There must be compromises between software developers and scientists in order to gather all necessities.

THE CONTINUOUS INTEGRATION STRATEGY

Auto-EC-Earth should be updated with the latest development version of the model (svn trunk). However, sometimes we need to add some extra features and to include them into the trunk takes some time. For this reason we have two extra branches, bsc_trunk and bsc_nemo see fig 1. As a procedure, every week these branches are updated.

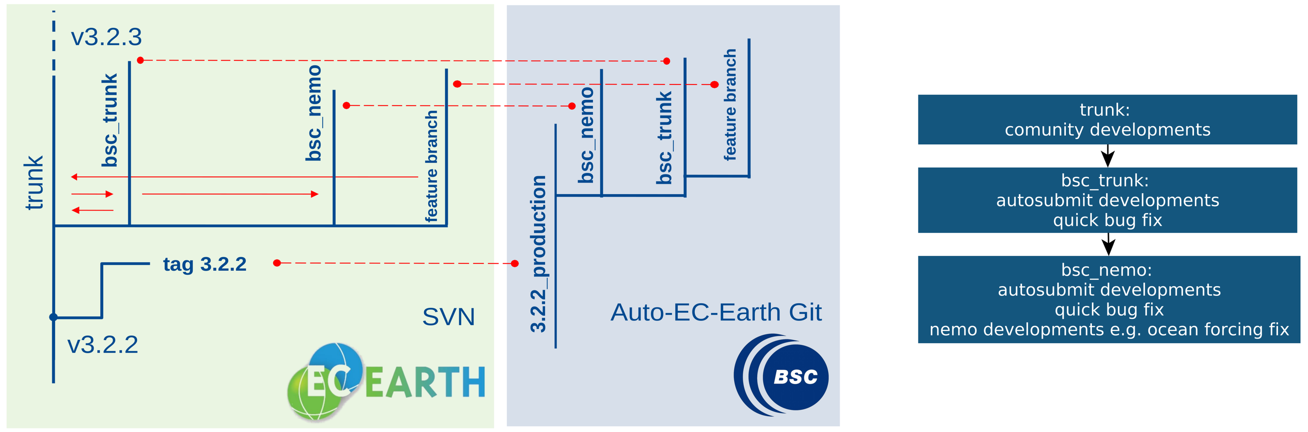

Figure 1: Branches Auto-ecearth - Ecearth

After rebase has been done, a weekly test is performed. By this approach a minimum set of cases are covered but this is all we can afford since running tests takes a long computing time. After each release we run a big set of tests which cover: 3 different clusters (nord3, ECMWF-CCA, and Marenostrum4), Low and High resolution and couple and ocean and atmosphere components.

RESULTS/PROBLEMS AVOID

The most important achievement is getting the last changes on a weekly basis from the community development repository. These changes trigger modifications in the Autosubmit layer. This methodology helps to discover problems before running long simulations. In our working team the following problems were found and fixed: The ocean model does not store in the restarts files the budget of evaporated water so the simulation was losing water mass. This means that after 10 years the ocean level was 5 meters below the real level. The ocean forcing was not working, it was being read a climatology but not the real yearly forcing files.

LESSONS LEARNED

Provide a small set of the most common cases fully running A tag is created when a weekly version works The most important thing to keep in mind is "be flexible and adapt the methodology to the final user necessities". Submodules can be a source of problems, so use them only if you really need them. It will prevent from mistakes creating one branch with the same name for all submodules that you use.

FUTURE WORK

Don't waste time in repetitive tasks: use Jenkins to automate and run the tests. Modularization the similar functionality, bottom up strategy, add module / functions tests. Add a reproducibility methodology [https://goo.gl/Z3akxN] to made sure that the differences are insignificant

Acknowledgement: Domingo Manubens | Mario Acosta | Alicia Sanchez Lorente Kim Serradell | Francisco Doblas Reyes

pablo.echevarria@bsc.es | earth.bsc.es

LaTeX TikZposter