A CNRM-CM6 monitoring using Autosubmit

E. Maisonnave

WN-CMGC-14-39

Abstract

A few members of seasonal forecast experiment using CNRM-CM6 on ECMWF IBM Power 7 has been performed using Autosubmit monitoring. A few day long collaboration at IC3 has been sufficient to adapt the existing CNRM workflow script to Autosubmit non-intrusive requirements. Nevertheless, a more comprehensive work would be necessary to fully exploit Autosubmit capabilities to monitor and control the full workflow (from compiling) on any kind of supercomputer platform.

Table of Contents

1. Context

1.1. IS-ENES2

JRA1/WP9 workpackage of IS-ENES2 FP7 project plans to improve performances of ensemble experiment at high resolution for a subset of Earth System models. Enhancements are focused on 4 topics: coupling, input/output, post-processing and monitoring. This work makes part of the IS-ENES2 activities related to ensemble simulation monitoring.

1.2. Autosubmit

Autosubmit [1] is a tool that helps creating, running and monitoring climate experiments remotely by using computing resources available at supercomputing platforms. It manages the submission of jobs[1] to queue scheduler, until there is no job left to be run. Additionally, it also provides features to suspend, resume, restart and extend similar experiment at later stage.

The IC3's monitoring tool is already used by Ec-Earth community. One goal of this document is to describe modifications needed to manage experiment with a different model (CNRM-CM6).

1.3. CNRM-CM6

As Ec-Earth, the SPECS [2] FP7 project, partly supported by the IS-ENES infrastructure and dedicated to seasonal/decadal forecast improvements, includes experiments using the CNRM-CM6 Meteo-France' s (MF) Earth System Model. This version (v6) is the latest available on HPC machines and is a candidate to initiate the future MF CMIP6 ESM. It includes ARPEGE (atmosphere), NEMO (ocean), SURFEX (land surface), TRIP (river routing) and GELATO (sea ice), coupled with the OASIS library.

At MF, the CNRM-CM6 running environment and monitoring system is local to the supercomputer (BULL B700 DLC). Like Ec-Earth, CNRM-CM6 is widely used by IC3's researchers, that would like to deploy this model on other platforms (like ECMWF machines) and monitor their experiments from IC3 local servers. Autosubmit is the appropriate tool, which proposes a common interface for different models to be submitted on remote platforms.

1.4. Machines

Both Ec-Earth and CNRM-CM models are available on the IBM Power7 ECMWF machine; a multi-model ensemble can then be monitored there, for example from IC3 computers. Such kind of configuration is needed to evaluate possibilities of M4 (Multi-Models Multi-Members) experiment monitoring, according to IS-ENES WP9 agenda.

[1] A climate simulation workflow, due to its length, has to be composed into pieces (or “chunks”). Each chunk can be divided into pre-processing, parallel run and post-processing. All these pieces have to be submitted, following a given order, to a batch scheduler: we call them “jobs”.

2. Implementation

2.1. Initial script replacement

New “template” files are created to add CNRM-CM to the list of possible model handled by Autosubmit.

It consists in 6 scripts (in autosubmit/templates/cnrm6):

- .conf: includes all environment variables necessary to parametrize a simulation
- .ini: executed once, when experiment starts, prepares input files that are common to all the chunks
- .sim : executed at each chunk, is the main script including pre-processing, MPI simulation and sequential post-processing
- .post: executed at each chunk, gathers all post-processing operations (output analysis, format conversion, etc ...) that can be done in parallel to .sim operations
- .clean: executed at each chunk, performs the directory cleaning after each chunk

Three other scripts have also to be filled (.localsetup, .remotesetup for source compiling and .localtransfer for file transfer), but we kept them empty for simplification reasons. Actually, our initial idea was to replace the CNRM existing script and the ARS chaining tool [3] with Autosubmit scripts (written for Ec-Earth). But, as a first step, it seems better to make as few modifications as possible in the CNRM environment to keep some kind of flexibility. Then, our work will present, for future IC3 and CNRM users, an example of what to modify in order to adapt, as simply as possible, an existing CNRM script to Autosubmit requirements.

Accordingly to this method, we found easier to simply include the existing CNRM script in the .sim template file and keep empty the .ini, .post and .clean files. Obviously, in a second step, it will be possible to identify (i) pre-processing actions only needed at experiment start, (ii) post-processing actions that can be done in parallel to the next chunk and (iii) directory cleaning actions that enables a better management of disk space and spread these operations on .ini, .post and .clean template files respectively.

Similarly, the .conf file was modestly enriched with ARPEGE or NEMO parameters. To have a full remote control of CNRM-CM model, this file will have to be completed (but .sim script also modified accordingly).

The existing CNRM script has been entirely copied on .sim template file. Needed modifications mainly consist in ARS chaining tool removal and Autosubmit chunk management implementation.

For the first step, it was necessary:

- to avoid calling scheduler (llsubmit) at script end
- to avoid calling ARS re-launching tool after each chunk

For the second step, we needed to:

- replace ARS defined into-square-brackets variables by Autosubmit defined into-percent-sign variables.

Another set of modifications was necessary to take into account a difference between ARS and Autosubmit in ensemble member handling (directory names have to be changed according to experiment identifier and member rank).

Finally, the CNRM functions enabling date and time management have to be adapted to new Autosubmit format.

In addition to these 6 model-dependant scripts, a machine-dependant script (autosubmit/conf/archdef/ecmwf.conf) has also to be modified, to take into account the 3 scheduler submissions of .sim script in one single operation[2]

After beeing reported on autosubmit repository, a general directory structure, needed for monitoring experiment with CNRM-CM on ECMWF machine, can be created through the "expid.py" command.

Then, a specific experiment must be parametrized on EXPID/conf/expdef_EXPID.conf[3], defining for example the number of ensemble members, the starting date and the chunk number per member. Identically, the monitoring tool itself can be parametrized, modifying for example the frequency of remote control for scheduled jobs on EXPID/conf/autosubmit_EXPID.conf.

Finally, a monitoring structure is created for the EXPID experiment, launching "create_exp.py".

At this stage, Autosubmit is able to export anything necessary to perform our experiment on the target supercomputer. The first two stages of this operation (file transfer and source compiling) has been switched off and previously done by hand[4].

2.2. Run test

Only a few pre-installed tools/library are necessary to use Autosubmit:

- *git*, for repository management
- *ssh/scp/rsync*, for remote control
- *bash*, or any kind of linux shell (on HPC), for script execution
- *SQLite*, for data-base management
- and *python*, for script configuration.

All these elements were already available on IC3 monitoring machine and on ECMWF supercomputer[5]. As Autosubmit itself, they are all free-ware components.

Launching a monitored experiment is done through "autosubmit.py" command.

[2] The original CNRM script starts with a triple scheduler command set. It means that the same script is launched 3 times, with 3 different scheduler options. Following scheduler options, different commands of the script are executed (fork). This specificity of Load Leveler batch scheduler makes possible the scheduling of three different scripts in one single submission. In our case, one of these scripts is parallel, the 2 others are sequential. The second one is queued until the successful completion of the first one. The same for script 3 and 2.

[3] This file and its directory structure is created by the previous command

[4] Assuming that access to ARPEGE source code is restricted, then not compilable. Executables has been provided by MF users and directly copied on an accessible directory of our supercomputer.

[5] Ec-Earth were already monitored by Autosubmit and running and on the ECMWF machine.

The whole directory structure is created on the remote supercomputer and the first elements of the experiment workflow (.ini script of each member) are simultaneously submitted to batch scheduler. Results of these two operations are then regularly scrutinized by the autosubmit.py script and next actions (.sim scripts for the first chunk) submitted. This procedure occurs until the experiment end (or a parametrizable number of failures for all ensemble members).

We show on Figure 1 the result of a checking command, which give a description of the current experiment. On this plot, we can visualize a status of a 2 member ensemble: both members are waiting (pink boxes) an allocation to start processing the second chunk of .sim script ("_2_sim" suffix). All previous operations (including post-processing) have been successfully completed (yellow boxes). Next steps are waiting (grey boxes) the results of .sim operations.

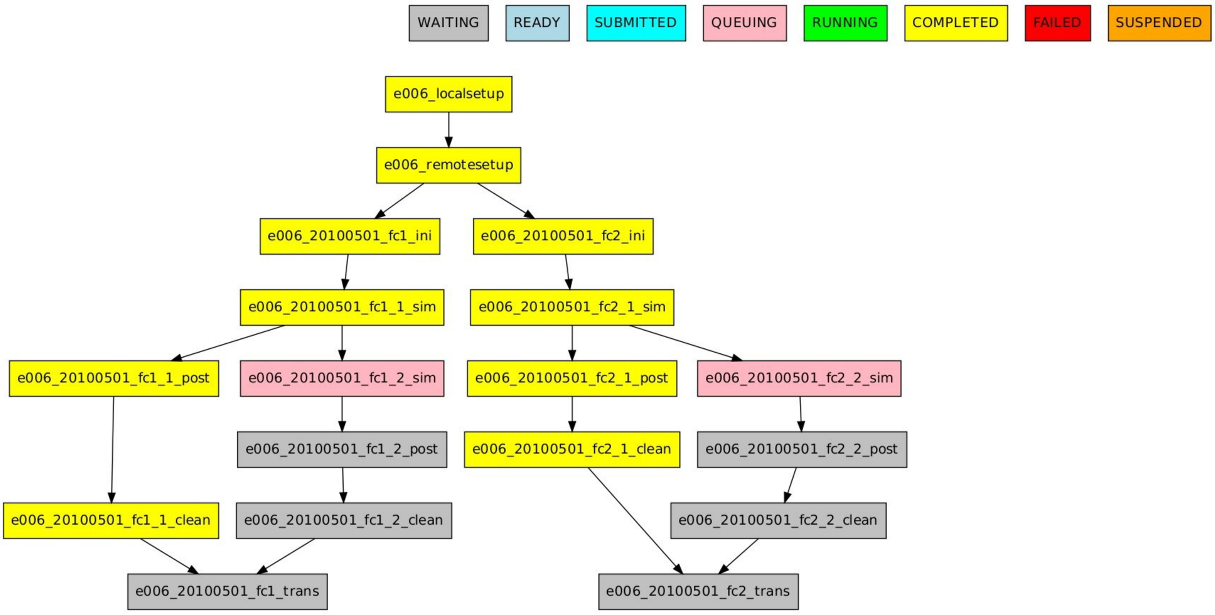

Figure 1: status of a 2 member ensemble of 2 month long CNRM-CM ESM simulations

When a failure occurs, the Autosubmit database is updated and a simulation can stop or be resumed. For some reasons, update is not always performed and database is desynchronized from the actual status of the experiment on the remote supercomputer. In this case, a manual upgrade of database status is still possible, via the recovery.py command.

3. Further improvements

3.1. Parametrization

As previously mentioned, the implemented solution is non intrusive relatively to the original CNRM launching script. It means that a model (and workflow) upgrade in our Autosubmit environment would be relatively straightforward.

Nevertheless, most of the model (and workflow) parameters have to be changed on the supercomputer. Since the Autosubmit tool allows to deploy and use a model remotely, a comprehensive description of our model parameters must be done following Autosubmit standards. It must include compiling options, input file management, namelist parametrization, output file post-treatment, etc.

3.2. Other platforms

Our test experiment were relying on existing IC3 Autosubmit deployment, IC3-ECMWF connection and ECMWF compiling and scheduler management.

For a more comprehensive test, the same experiment could be performed from a different monitoring machine and to a different supercomputer, which would allow to test a different security policy between local and remote machines. We then proposed to MF to extend our test to their “BEAUFIX” BULL supercomputer, where CNRM-CM is already available. CERFACS' machines, although modest, could also contribute to extend our test.

The author strongly acknowledge Domingo Manubens and Paco Doblas-Reyes, who hosted this common work at IC3, Barcelona and kindly reviewed this report. This work has been supported by the IS-ENES2 FP7 infrastructure.

References

[1] D. Manubens: Autosubmit website, http://ic3.cat/wikicfu/index.php/Tools#Autosubmit

[2] SPECS website, http://www.specs-fp7.eu/

[3] A. Braun, 2008: Automatic Relaunch System used for ARPEGE-Climat long integrations, Météo-France/CNRM, CNRS/GAME, (pdf)