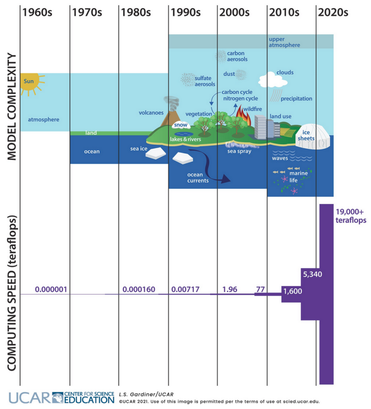
Enabling Reliable Workflow Development with an Advanced Testing Suite

Alejandro García, Leo Arriola, Gilbert Montané, Miguel Castrillo,

Bruno de Paula Kinoshita, Eric Ferrer Escuin and Aina Gaya

Earth Sciences Department, Barcelona Supercomputing Center, Spain (eric.ferrer@bsc.es)

Earth simulation complexity



Simulating climate, weather, or air quality is not a simple task. These systems are complex and involve interactions between physical, chemical, and dynamical processes that take place on a wide range of spatial and temporal scales.

The models used are highly detailed and computationally demanding. Although they are designed to be as accurate and efficient as possible, even small changes, such as changing a parameterization or fixing a bug, can end up causing undesired effects, like noticeable differences in the outputs.

On top of that, the workflows that support these simulations add their own set of challenges. Aspects like compiler options, hardware differences can affect how the model runs, sometimes in unexpected ways. This makes developing new features or improving existing ones especially tricky, not just in the model itself, but also in the workflow that runs it. Making sure that changes do not break the model operation or introduce inconsistencies requires careful and solid testing and good traceability.

The Autosubmit workflow manager

The Autosubmit workflow manager was developed as a unified framework to support complex tasks, such as running weather and climate experiments, and to carry out Earth science-related simulations, mainly in HPC environments.

Autosubmit is a Python-based tool to create, manage and monitor many types of experiment workflows remotely without manual intervention. It can orchestrate all tasks in a workflow by managing dependencies, communicating with the involved platforms, and handling any potential errors.

AUTOSUBMIT

Exhaustive testing challenge

Climate simulations require highly complex workflows that often require extensive testing to ensure accuracy and reliability. However, exhaustive testing is time-consuming and resource-intensive. There is a pressing need to streamline this process while maintaining the integrity of the simulations and ensuring that they produce reliable results.

In response to these challenges, the Testing Suite, a Python package designed for batch operations with the Autosubmit workflow manager, enables developers to create, configure, run, and monitor experiment suites in parallel. It performs integration testing to detect regressions in workflow execution, simulation output, or computational performance, reducing compatibility issues across configurations and helping ensure stability before merging changes.

The testing suite includes built-in checks and reporting tools, and can be integrated with pipeline automation tools such as Jenkins, GitLab CI/CD, or GitHub Actions. This enables automated, regular tests and streamlines permission handling while keeping logs organized and accessible for debugging and traceability.

Jenkins

Testing suite capability

The Testing Suite has been successfully implemented in real-world climate and air quality simulation projects. It is also used to test the climate DT workflow in the Destination Earth project, proving its effectiveness in streamlining the testing process. By automating integration tests and scientific result verification, it ensures that the simulations run efficiently and produce consistent, reliable outputs, significantly improving the development cycle. The following table shows the list of experiments to be executed at the same time.

test id	details	hpc
t0n2	weekly test case: ORCA1L75 + nudging + surface restoring	marenostrum5
t0n3	weekly test case: T255L91 (IFS only)	marenostrum5
t0n4	weekly test case: T255L91-ORCA1L75 (t0nf ICs). ScenarioMIP/ssp245 outclass	marenostrum5
t0n5	weekly test case: T255L91-TM5 full chem	marenostrum5
t0n6	weekly test case: T255L91-ORCA1L75-TM5 full chem	marenostrum5
t0nc	weekly test case: T255L91-ORCA1L75 - CLEAN direct transfer	ecmwf-hpc2020
t0nd	weekly test case: ORCA1L75	nord3v2

The Testing Suite allows workflow test experiments to be executed in batches while bypassing responses in a structured way. The testing suite simplifies the interaction between users and their test cases by allowing them to operate on multiple instances at the same time, ensuring that each component and configuration are thoroughly tested. The most widely used functionalities are:

- Easily copying existing test cases.
- Performing autosubmit operations on tests, such as creation, refreshing of their code repository, cleaning up their outputs and run the experiments.
- Displaying and comparing experiment run parameters and outputs between each run and their predecessor.
- Creating reports and useful outputs.

Performance metrics

The Testing Suite can provide CPMIP (Computational Performance Model Intercomparison Project) metrics, providing insights into the efficiency of the workflows. The following table shows the metrics reported by the Testing suite:

test id	status	sim run time avg	sim runs	#PROC	SYPD	ASYPD	details	hpc
t0n7	Succsessful	1:20:33	3	1008	17.87	15.5	weekly test case: T255L91-ORCA1L75-LPJG-PISCES	marenostrum5
t0n8	Successful	2:09:46	3	560	9.75	8.78	weekly test case: T255L91-ORCA1L75-LPJG-PISCES-TM5 co2,co2fb	marenostrum5
t0nb	Successful	0:02:31	2	784	47.68	19.15	weekly test case: ORCA1L75 CMORIZATION FALSE	marenostrum5
t0nc	Successful	0:09:21	4	896	12.82	11.57	weekly test case: T255L91-ORCA1L75 - CLEAN direct transfer	ecmwf-hpc2020

SYPD is the number of simulated years per day for the model. **ASYPD** is the actual SYPD, taking into account system interruptions, queue waiting time, or model workflow issues. This number may be lower than SYPD. The **sim run time avg** value refers to the average time it takes to run the simulation taking into account the number of **sim runs**. **#PROC** is the number of processors used for the simulation.

It is also possible to compare these metrics with previous executions of the same experiments.

Output checker

By performing bit-by-bit comparisons of the output data, the Output Checker ensures that the results are consistent and free from any unintended regressions. This tool is especially valuable for detecting discrepancies in large datasets, which can be challenging to identify manually. It significantly improves the reliability of simulations by providing an additional layer of validation for the results.

This comparison is done not only to verify that results are consistent across various configurations but also to identify any subtle changes that may indicate issues with the workflow. The following table shows an example of the experiments checked.

test id	benchmark	missing files check	output checker	failures	description	hpc
t0fw	t0fq	Successful	FAILED	`ADRYDEP AERO_ACPREC AERO_CUPREC AERO_DEPDRY pm2p5 t2 WETDEP`	GLOBAL_AEROSOLS cold start with SPINUP	marenostrum5
t0fu	t0fo	Successful	Successful		REGIONAL_CHEM CAMS2_40 fc 0.2 restart	marenostrum5
t0i8	t0i7	Successful	FAILED	`ACPREC ADRYDEP AERO_ACPREC AERO_CUPREC AERO_DEPDRY aerosol_optical_depth CUPREC pm10 RLWTOA sconco3 SMC t2 T2 WETDEP`	CHEM global	marenostrum5

Conclusions

- The suite reduces the time and resources required for comprehensive testing while ensuring workflows are reliable and reproducible, providing confidence in the stability of the code before release.
- CI testing streamlines both major and minor releases, contributing to a smoother development process.
- Automatic data comparison swiftly detects regressions during test execution.

References

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

– Montané Pinto, G., Ferrer, E., Olid, M., Garcia, A., Bonet, G., and Mozaffari, A.: Collaboratively developing workflows at the BSC-ES, EGU General Assembly 2024, Vienna, Austria, 14–19 Apr 2024, EGU24-2152, https://doi.org/10.5194/egusphere-egu24-2152, 2024.

– 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.: CPMIP: measurements of real computational performance of Earth system models in CMIP6, Geosci. Model Dev., 10, 19–34, https://doi.org/10.5194/gmd-10-19-2017, 2017.

Acknowledgements

- The research leading to these results has received funding from the EU HE Framework Programme under grant agreement n° GA 101081460
- We acknowledge the RESPIRE project, which is part of the Recovery, Transformation and Resilience Plan (Plan de Recuperación, Transformación y Resiliencia, PRTyR) funded by theEuropean Union – NextGenerationEU

Funded by the European Union NextGenerationEU