**Running a test case on Gaea**

The model simulations indicated in this manuscript were run on NOAA’s Research and Development High Performance Computing System (RDHPCS), Gaea platform. This document has information on running a test case on NOAA’s RDHPCS Gaea platform and examining the results. The details on this document are extracted from [here](https://github.com/NOAA-GFDL/MOM6-examples/wiki/Getting-started#running-on-gaea) (last accessed on January 15, 2026).

On Gaea you will be logged into a front-end node on which you compiled. The model executable can only execute in parallel on the compute nodes. The executable is loaded onto the compute nodes with the srun command which is invoked from a batch node. Job scripts (batch scripts) run on the batch nodes and are normally submitted from the front-end nodes. In brief:

* Front-end nodes - can see /home and /gpfs/f5. Use for editing, compiling and submitting jobs.
* Batch nodes - can see /home and /gpfs/f5. Use for launching the executable with srun.
* Compute nodes - can only see /gpfs/f5.
* Use srun instead of mpirun.

1. **Interactively**

To run the model interactively you can obtain an interactive session on a batch node with (for example):

salloc -n 60 -t 125 -p batch --cluster c4

This allocates 60 cores for 125 minutes on cluster c4 in queue=batch. You can also ask for N nodes (one node=36 cores):

salloc -N 10 -t 60 -p batch --cluster c4

to get a session with 360 (10x36) processors for 1 hour. If you want to use the debugger, you can use:

salloc --x11=first -q interactive --cluster c4 -N 2

1. **Simple batch job**

The simplest batch script (say mom.sub) to run the model would look like:

#!/bin/bash

#SBATCH -n 60

## if you prefer to specify nodes, use instead:

##SBATCH -N 10

#SBATCH --time=0:60:00

#SBATCH --job-name="MOM6"

#SBATCH --output=slurm.out

#SBATCH --error=slurm.err

#SBATCH --qos=normal

#SBATCH --partition=batch

#SBATCH --clusters=c4

## obviously use your group account:

#SBATCH --account=gfdl\_o

srun -n 60 ./MOM6

then submit to the scheduler with:

sbatch mom.sub

**Checking and examining your results**

Once you have run the test case you will get some output (and/or a netcdf HDF5 error that will indicate that you did not create the RESTART/ folder above) in the directory from which you ran srun or mpirun (i.e. in MOM6-examples/ocean\_only/double\_gyre/ for the above example).

The numerical results will depend on the specifics of your platform (chip, operating system, compiler vendor, versions, etc.) and so "bit for bit" reproducing the results obtained on GFDL platforms is not guaranteed. Some basics are expected for the checked-in experiments:

* The generated output documenting parameters and configuration (MOM\_parameter\_doc.\*) should be indentical (except for choices of processor layout).
* The absence of any U\_velocity\_truncations or V\_velocity\_truncations files, which if present indicate numerical instabilities.

If you want an example of regression files used by GFDL for testing, you can look in a demonstration regression repository. If you clone this regressions repository, or better yet, maintain one of your own for you platform, then you could check your results by comparing the newly generated ocean.stats file to the committed one:

diff ocean.stats ocean.stats.intel

If you are running on gaea and the ocean.stats files differ, then try removing the Depth\_list.nc file and ensure that you ran the initial test with 8 processors (i.e -np 8). If you are not running on gaea then it's expected that there will be small differences in the ocean.stats files. This is because no two compilers or computers can be relied upon to do all floating point calculations in a consistent way.