
MOM4 Quickstart Guide

Niki Zadeh <Niki.Zadeh@noaa.gov>

Brief instructions for running MOM4 experiment

Table of Contents

1. Where to start?	1
2. How to get the source code and scripts	1
3. How to compile and run the MOM4 tests	1
4. Notes	2
5. How to prepare input data	2
6. Examine the output	2

1. Where to start?

A good place to start is the "doc" directory, where you will find "user" guides for the MOM.

2. How to get the source code and scripts

You can get the tar ball for source code by logging into your GFDL GForge account and navigate to the Files section of MOM4p1 project area. Then you need to untar the tar bundle and cd to the resulting directory. For convenience, this directory will be referred to as the ROOT directory. A README file in the ROOT directory will tell you the contents of each subdirectory under ROOT.

In the sections below, `test_case` is a generic name referring to the name of a specific experiment you are working on. Thus, `test_case` in the path `ROOT/exp/test_case` should be replaced by a concrete name (for example `mom4p1_solo`, `mom4p1_cp1d`, etc.)

3. How to compile and run the MOM4 tests

MOM4 requires that NetCDF and MPI libraries be installed on users' platform.

There are 5 types of test models provided for this release in the `exp` directory all using the GFDL shared infrastructure (FMS). Each type has a compile and a run script.

1. Solo : stand alone MOM4p1 Ocean model.
2. Coupled : MOM4p1 coupled with GFDL ice model (besides null versions of atmosphere and land models).
3. EBM : MOM4p1 coupled with GFDL ice and atmosphere energy balanced model
4. ICCM : MOM4p1 coupled with GFDL ice, land and atmosphere in low resolution setup.
5. CM2.1p1 : GFDL CM2.1 model coupled with MOM4p1

To run Solo models:

1. Ensure that you have the right environment variable for your platform in the file `bin/environs.YOUR_PLATFORM_ID`. `YOUR_PLATFORM_ID` could be any string that identifies your platform. The file `bin/environs.YOUR_PLATFORM_ID` gets sourced at the beginning of all compile and run scripts and is there to make sure all compile and run time library paths are found.
2. `cd` to `exp` and run `mom4p1_solo_compile.csh` first.
3. Modify the 'name' variable in the script `mom4p1_solo_run.csh` to be the name of the test you want to run. A list of available tests is included in the script. Also make sure you have a large enough working directory and made a symbolic link to it called "work" in your top directory. I.e., `> cd mom4p1_pubrel_dec2009 > ln -s YOUR_LARGE_WORK_DIR work`
4. Get the required input data for the test from GFDL ftp site. You can get the info by running the script `mom4p1_solo_run.csh` and following the instructions.
5. Run `mom4p1_solo_run.csh`
6. The results go into "work"

To run all other models foo: Do the same steps above to `foo_compile.csh` and `foo_run.csh`

4. Notes

The scripts have been tested fully only with Intel Fortran compiler on ia64 platform and partly with pgi and path-scale compilers on x86_64 and also gfortran4.3 compiler on Core2Duo processor.

Some of these tests require a large disk space to save the input data. Choose a partition with enough space (1-2 G) to untar the code and data bundles.

If you want to use the same root directory for all three experiments we suggest to compile in the order of increasing complexity: solo -> coupled -> ebm . You may have to remove some of the existing .o and .mod files if you want to compile ebm experiment after the coupled experiment. E.g., you probably have to: `cd ../exec; rm mpp*.o mpp*.mod fms_io*.o fms_io*.mod`

IBM platform users might want to add the following line to the top of the run scripts `setenv LDR_CNTRL MAXDATA=0xD0000000@DSA`

The compile scripts provide the basic capability to use static memory allocation which might be faster on some platforms. In that case you need to adjust the values of domain bounds properly according to the number of processors and layout.

The compile scripts use netcdf3 by default. If you want to use netcdf4 libraries instead you can do so by deleting the "-Duse_netCDF3" from the CPPs in compile scripts and then recompile. There are 5 types of MOM4 tests provided in the `exp` directory, both using the GFDL shared infrastructure (FMS) :

5. How to prepare input data

The input data needed to run the selected experiments (tests) that are included in this release are available via [anonymous ftp](ftp://ftp.gfdl.noaa.gov/perm/MOM4/mom4p1_pubrel_dec2009/exp) [ftp://ftp.gfdl.noaa.gov/perm/MOM4/mom4p1_pubrel_dec2009/exp]. Note that data in `ASCII` , `HISTORY` , `RESTART` directories are *NOT* needed for running experiments. They are the outputs of the experiments and are provided for the purpose of comparing your results with results produced at GFDL.

Tools are provided so that users can create data from scratch for their own experiments. For more details refer to `ROOT/src/preprocessing`.

6. Examine the output

To keep the runscript simple all output files of a model run will be in the work directory. There are three types of output files:

1. `ascii` file with `.fms.out` extension: the description of the setup of the run and verbose comments printed out during the run.
2. restart files in `RESTART` directory: the model fields necessary to initialize future runs of the model.
3. history files with `.nc.tar` extension: output of the model, both averaged over specified time intervals and snapshots.

The `ascii` file contains everything written to the screen during model execution. The total time for model execution as well as the times of separate modules are reported here. All `.tar` files should be decompressed for viewing. The decompress command is:

```
tar -xvf filename.tar
```

Users will see result files in NetCDF format. Postprocessing tools such as Ferret, ncview, grads or matlab can be used to view data in these files.

The outputs of the selected experiments are provided on [GFDL ftp site](ftp://ftp.gfdl.noaa.gov/perm/MOM4/mom4p1_pubrel_dec2009/exp) [ftp://ftp.gfdl.noaa.gov/perm/MOM4/mom4p1_pubrel_dec2009/exp] for the purpose of comparing your results with results produced at GFDL.