Earth System Modeling Framework

Part II - JMC code l First Improvement User's Guide

Chris Hill, Erica Peterson

Contents

1 Introduction

This document provides a brief synopsis of the download, compilation and execution procedure for the the Earth System Modeling Framework (http://www.esmf.ucar.edu) JMC code L First Improvment. The code and its time to solution measurements are part of ESMF joint milestone F. Code L is a coupled $2.8°$ ocean atmosphere configuration of the MIT General Circulaiton Model (MITgcm). This configuration is actively in use for research into air-sea dynamical and biogeochemical couplings, especially on interannual, decadal, centennial and greater timescales. Details of the code configuration can be found under the *Applications* link at http://www.esmf.ucar.edu. Reference material for the MITgcm code can be found at http://mitgcm.org and specific information on the current JMC code L setup and its future evolution over the ESMF project can be found at http://mitgcm.org/projects/ESMF/JMClimprov1.

The milestone F configuration provides an ESMF development cycle reference point for a time to solution metric evaluating ESMF framework overheads. In this test code L has evolved internally and the framework can now replace many parts of the codes' native infrastructure and superstructure. Two forms of code L exist, one outside the framework and one that makes use of the framework. These two forms are used to establish the time to solution, for an identical problem.

The problem used to measure time to solution is a ten-day coupled simulation starting from the saved state of a 50 year coupled spin-up. This milestone will gives the time to solution for both framework and non-framework code. At this stage, part way through framework development, the time to solution is calculated and found to be significantly increased when using the framework. Section 2 describes how to download the code for L first improvement. Sections 4 and 5 describe the steps for compilation, execution and validation of the results on the MITgcm cluster. The exact procedures and pathnames described are tailored to the configuration of the MITgcm cluste system **cg01.lcs.mit.edu**. However, the MITgcm code and the ESMF code can both execute on many other platforms and many of the steps are the same for other platforms and systems.

2 Downloading JMC Code L First Improvement Code and Data

The JMC Code L Improvement 1 code and data can be dowloaded from the download link under http://mitgcm.org/projects/ESMF/J This link will fetch a compresses tar archive called `JMClimprov1.tar.gz`. After downloading this file needs to uncompressed using the command *gzip -d JMClimprov1.tar.gz* and its contents extracted using the command *tar -xvf JMClimprov1.tar* into an appropriate workspace directory (for example a directory under `/s01/USERID` (where USERID refers to a login name) on the **cg01.lcs.mit.edu** system).

3 Directory Structure

The top-level JMC Code L First Code Improvement directory contains `atm`, `ocn` and `esmf_top` subdirectories along with scripts and files required to build execute in batch (`runtest.sh`) and validate (`STDOUT.0000.ref`) the improvement 1 ESMF based code. The file `README.milestonef`, also at the top-level, contains details of the contents of each of these directories, files and scripts as well as pointers to the MITgcm release1 documentation (http://mitgcm.org/sealion). For the `runtest.sh` script to execute correctly the dynamically linked library search paths `/usr/local/pkg/lahey/lf9562/lib` and `/usr/local/pkg/ESMF/ESMF_1_0_4/mpich-1.2.5..10/l` must be used.

4 Build

4.1 System Requirements

A Fortran 90 compiler and system library, the ESMF_1_0_4 toolkit library and a compatible MPI library are required to compile and link this code. All application code routines are in Fortran except for a single timing routine that uses

C. This routine requires a C compiler and a C library with the gettimeofday() function. Appropriate macros for calling a C procedure with with a pointer to a 64-bit float are required and are included.

Note that each MPI combination of compiler and transport layer requires its own complete MPI build. The library paths for appropriate compiler library, include files and compiler executable must be properly set when building MPI and when building ESMF. In addition the MPI libarary paths, the compiler executable path and the ESMF dynamic library and module search paths must be set consistently when compiling or running with either MPI or ESMF and MPI.

4.2 General

The script `runtest.sh` in the JMC Code L First Code Improvement top-level directory creates a single executable from several libraries of code

- `esmf_top/a.out` is the executable for the coupled application. This is the code that is invoked when the ESMF application is executed.

- `ocn/src/tutorial_examples/global_ocean.128x64x15/run/mmout/libmitgcm_org_ocn.a` is the library archive for the ocean component and is built from sources in sub-directories under `ocn/src`. The directory `ocn/src/tutorial_examples/global_ocean.128x64x15/run/mmout` also contains Fortran 90 module files for the ocean component.

- `atm/src/tutorial_examples/aim_5l.LatLon/run/mmout/libmitgcm_org_atm.a` is the library archive for the atmosphere component and is built from sources in sub-directories under `atm/src`. The directory `atm/src/tutorial_examples/aim_5l.LatLon/run/mmout` also contains Fortran 90 module files for the atmosphere component.

The directories `ocn/`, `atm/` are organised using the MITgcm release1 scheme that is described at http://mitgcm.org/sealion.

4.3 Building and executing JMC Code L First Improvement on the MITgcm cluster

The MIT myrinet cluster consists of a gateway node cg01, and sets of cluster nodes, myrinet-3-NN, myrinet-4-NN. Log on to gateway machine cg01.lcs.mit.edu, and unpack the First Code Improvement tar file into a scratch (e.g. /s01/USERID) directory. In the top-level of the unpacked directory execute the command `./runtest.sh`. This will submit a job to compile and run the JMC Code L using ESMF.

5 Verification

After the `runtest.sh` script executes the coupled simulation it carries out a check of output and extracts timing information to determine time to solution. The output check compares the daily global mean atmospheric zonal flow with a previously stored set of values for this configuration. The previoulsy computed set of values are in the file `atm.1/stdout.ref` under the JMC Code L Improvement 1 top-level directory. These stored values are checked against the time series of mean flow written to the file `atm.1/STDOUT.0000` in the script `runtest.pbs`. Time to solution information is also extracted by `runtest.pbs`. The compute time for the 16 processor, 10-day simulation used for the Code L Improvement 1 is converted into a measure of the time required for one year of simulation. This is printed in the summary output file `stdout.pbs`.