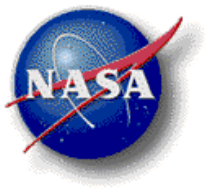

Earth Global Reference Atmospheric Model (GRAM) 2016 User Instructions

Patrick White, Earth-GRAM Lead

NASA/MSFC, Natural Environments, EV44

patrick.w.white@nasa.gov, 256-544-5776

Earth-GRAM 2016 is the new release of Earth-GRAM provided to Earth-GRAM users. Earth-GRAM 2016 is written in a C++, object-oriented programming language as requested by a majority of users. The Earth-GRAM 2016 release includes the source code and a PC executable, as well as instructions for using the model. Running standalone GRAM 2016 with an executable closely resembles the method for running GRAM 2010. Improvements to the user functionality of GRAM standalone should be expected in future releases. Your patience in the improvement of our models is greatly appreciated. A full Earth-GRAM 2016 user's guide will be coming soon. The Earth-GRAM 2010 user's guide will be provided in this release for concurrent methodology.

Earth-GRAM 2016 release package

- The 'Documentation' folder provides the user instructions, previous user's guides, and model history.
- The 'IOfiles' folder provides some of the input data necessary for running the model. There is a sample input and output case provided as well.
- The 'My Test' folder is provided as a test folder to run and execute the model.
- The 'NCEPdata' folder include the NCEP data and includes the ASCII and binary NCEP data. A new period of record (POR) of 1997 to 2015 is used for the next Earth-GRAM release. The 'FixedASCII' folder in the 'NCEPdata' folder includes the ASCII data and the NCEPbinF.f90 program to convert ASCII to binary. The ASCII files need to be converted to binary for users on Linux platforms.
- The 'RRAdata' folder provides the RRA data at sites for 1983, 2006, and 2013.
- The 'Source' folder provides the source code for Earth-GRAM 2016.

New Features for Earth-GRAM 2016

- Code re-written in C++, object oriented programming language.
- New POR for NCEP data of 1997 to 2015. NCEPyr input variable must be set to 9715.
- New input variables included for Earth-GRAM 2016: 'iaux' = auxiliary profile option (0 = no auxiliary profile; > 0 for auxiliary profile), 'mc' = number of Monte Carlo runs.

Instructions for Running Earth-GRAM standalone

- Place the NameRef.txt and GRAM executable in the 'My Test' folder.
- The GRAM2016 NameRef.txt file is different from the GRAM2010 NameRef.txt file so the 2010 file will not work for GRAM2016.
- The directories are not entered in quotes in NameRef.txt.
- Run the executable and enter the home directory (..\EarthGRAM2016\My Test\).
- Enter the namelist file name (NameRef.txt).
- If the directories and inputs are entered correctly the output files (output.txt, special.txt, species.txt and BLTest.txt) will be placed in the My Test folder.
- Trajectory and auxiliary profile files need to be placed in the home directory. Sample trajectory and auxiliary profiles are placed in the 'IOfiles' folder for format specification.

Instructions for Compiling and Running GRAM on Linux or Mac Platform

- Compile procedures for standalone GRAM on Linux: g++ -o GRAM Init.cpp InitP.cpp NCEP.cpp RRA.cpp AuxProf.cpp MET.cpp Pert.cpp Map.cpp MSIS.cpp HWM.cpp JB2008.cpp Atmod.cpp
- Linux executable provided in 'My Test' folder named GRAM.
- A Linux version of the 'NameRef.txt' is located in 'IOfiles' folder.
- Run the model by typing ./GRAM at the command line
- Enter the home directory (/home/GRAM/EarthGRAM2016/MyTest/).
- Enter the namelist file name.
- If the directories and inputs are entered correctly the output files (output.txt, special.txt, species.txt and BLTest.txt) will be placed in the MyTest folder.
- Trajectory and auxiliary profile files need to be placed in the home directory. Sample trajectory and auxiliary profiles are placed in the 'IOfiles' folder for format specification.
- Use NCEPbinF.f90 to convert NCEP ASCII files to binary for Linux platform. Compile using 'gfortran NCEPbinF.f90'.

Instructions for embedding GRAM in a simulation

- Users can embed GRAM in their simulation with the traj member function from the Atm1 class in Atmod1.cpp
- Create an object from Atm1, Atm1 atms1;
- Initialize the data before model runs using initdata member function, atms1.initdata();
- Calculate initial GRAM atmospheric variables using traj and setting initonce = 1, atms1.traj(h, phi, thet, time, iupdate, initonce, …..)
- Calculate next GRAM atmospheric variables using traj and setting initonce = 0, atms1.traj(h, phi, thet, time, iupdate, initonce, …..)
- For Monte Carlo dispersions just reset to start of run and set initonce = 1. The initial random number seed 'nr1' will be iterated when resetting with initonce = 1.

List of source files and classes:

Atmod.cpp – houses the **Atm** class which computes means and standard deviations of pressure, density, temperature and winds for output. As well as perturbed values and atmospheric constituents.

Atmod1.cpp – houses the **Atm1** class which is very similar to the **Atm** class but contains the 'traj' member function. The 'traj' member function allows for calling to Earth-GRAM from a simulation.

AuxProf.cpp – houses the **AuxProf** class which takes an alternate or auxiliary profile to be read in and used in place of Earth-GRAM data.

HWM.cpp – houses the **HWM** class for the Harmonic Wind Model. Used to calculate winds for the JB2008 and MSIS thermosphere models.

Init.cpp – houses the **Init** class for initializing the data for the model prior to runs.

InitP.cpp – houses the **InitPert** class for calculating the initial perturbations.

JB2008.cpp – houses the **JB2008** class for calculating atmospheric variables in the thermosphere.

Map.cpp – houses the **Map** class for calculating atmospheric values in the middle atmosphere region.

Met.cpp – houses the **Met** class for calculating atmospheric variables in the thermosphere.

MSIS.cpp – houses the **MSIS** class for calculating atmospheric variables in the thermosphere.

NCEP.cpp – houses the **NCEPmods** class for reading processing NCEP data for lower atmosphere data.

Pert.cpp – houses the **Pert** class for calculating perturbations.

RRA.cpp – houses the **RRA** class for reading and processing Range Reference Atmosphere (RRA) database site.