Design and implementation tests of a general circulation model with flexible and readable code: design of physical process module

Yasuhiro Morikawa[1]; Masaki Ishiwatari[2]; Yoshiyuki O. Takahashi[3]; Masatsugu Odaka[1]; Yoshi-Yuki Hayashi[1]

[1] Department of Cosmosciences, Hokkaido Univ.; [2] Faculty of Environmental Earth Science, Hokkaido University; [3] Dept. of Earth & Planetary Sciences, Kobe Univ.

Aiming for an atmospheric general circulation model (GCM) with flexible and readable source codes, the programs have been designed and their implementation tests have been performed. A flexible GCM whose programs can be easily changed is useful for the considerations on structures of various planetary atmospheres and for the examinations from the viewpoint of comparative planetary science. We have developed a data I/O library (gt4f90io library, Morikawa et al., 2006), a library for numerical fluid dynamics calculations with the spectral method (SPMODEL library, Takehiro et al., 2006) and coding rules including naming rules of variables. Based on these resources, a dynamical core for a GCM with explicit time integration scheme is designed by referring to AGCM5 (SWAMP Project, 1998) written in FORTRAN 77, and by taking advantage of features of Fortran 90/95 (Morikawa et al., EPSU Meeting 2005). In this presentation, we report (1) auto-generation of model document by the use of RDoc Fortran 90/95 parser, (2) improvement of the data I/O library gt4f90io, (3) design of management style of physical process modules.

RDoc is a library for auto-generation of program documents by object-oriented script language Ruby. Using Enhanced version of RDoc Fortran 90/95 parser (Morikawa et al., 2007), documents (reference manuals) of GCM can be auto-generated. By RDoc Fortran 90/95 parser, HTML documents are generated, and we can read model documents via web browsers. Not only documents but also source codes of GCM can be browsed easily and dependencies of modules can be traced by hyperlinks.

Gt4f90io is a data I/O library for Fortran 90/95 programs. The aim of gt4f90io is the clear separation of source codes for information handling from those for time integration of prognostic variables. For this purpose, subroutines for data I/O have been enhanced, and modules for handling command line arguments, date and time, and monitoring of CPU TIME are implemented.

Aiming for flexibility for exchanging the physical processes, management style of physical process modules is examined. In AGCM5, in order to exchange numerical schemes for a certain process, multiple files which contain subroutines with same name but calculate the process with different numerical schemes are prepared. For example, in the case of cumulus convections, two files 'p2cuma.F' and 'p2cumk.F' which contain subroutines with same name are prepared for exchanging of two schemes 'a' and 'k'. By editing Makefile, it is determined which program files should be included in executable files. In management style of AGCM5, however, main program cannot inquire which numerical scheme is used for each physical process. In the model developed here, we try the management style in which module for each numerical scheme is prepared, subroutine in each module has same name, and exchange of numerical scheme is realized by rewriting USE statement in main program. In this style, main program can inquire which numerical scheme is used, and information hiding is possible. As a result, it is expected that management cost will be reduced and that less misleading program structure can be constructed. We will further examine the usefulness of this style with performing implementation tests.

Our model is provided as DCPAM, Dennou-Club Planetary Atmosphere Model, at http://www.gfd-dennou.org/library/dcpam.