Вестник МГТУ, том 13, №3, 2010 г.   стр.541-544

УДК 551.51 : 004.9 (045)

Object-oriented hierarchical data structure for Framework Atmospheric Model

O.V. Martynenko, M.M. Gladkikh, I.V. Artamonov
Polytechnic Faculty of MSTU, Physics Chair

Abstract. A universal modeling tool is being developed on the basis of the global Upper Atmosphere Model (UAM) for studying the interrelationship of the broad range of various processes and phenomena in the upper atmosphere. For this purpose the UAM structure was reorganized into the open framework, including the controlling Metamodel Manager and several subordinate models of separate atmospheric regions and processes. They are, for example, an ionosphere model, neutral atmosphere model, electric field model, wind model etc. Each included model is independent from the others and calculates the certain set of physical parameters of the modeling object. These sub-models exchange data through the Manager using the unified interface. This approach allows the integration of a wide range of data sources of different kinds, both experimental and modeled. This work contains a detailed description of model data internal logical organization, which provides the possibility to store and to exchange various kinds of data with a full description in a standardized way.

Аннотация. На основе глобальной численной модели верхней атмосферы UAM разрабатывается универсальный модельный инструмент для исследования взаимосвязей широкого круга процессов и явлений в верхней атмосфере Земли. С этой целью структура UAM реорганизуется в открытую рамочную конструкцию (FrAM – Framework Atmospheric Model), включающую управляющий блок – Менеджер Метамодели – и набор подключаемых к нему моделей различных атмосферных регионов и процессов, таких, как модель ионосферы, модель нейтральной атмосферы, модель электрического поля, модель поля ветров и др. Подключаемые модели независимы друг от друга, и каждая из них самостоятельно рассчитывает определенный набор физических параметров моделируемого объекта. Модели обмениваются данными через посредство Менеджера, используя стандартизованный интерфейс. Такой подход открывает возможности интеграции в единую структуру широкого круга разнообразных источников данных, включая модельные и экспериментальные. Представляемая работа подробно описывает логическую организацию внутренней структуры данных FrAM, позволяющую стандартным способом обмениваться и хранить данные различного вида совместно с их полным описанием.

Ключевые слова: атмосфера, ионосфера, численное моделирование, интеграция моделей
Keywords: atmosphere, ionosphere, numerical modeling, model integration

1. Introduction

The atmosphere is a complex natural system of many interconnecting elements. The amount of computer models of various atmospheric domains increased greatly in recent decades. But these stand-alone models use simplifying assumptions about the interaction of a particular domain with the rest of the system. For the reliable description and prediction of space weather events, however, it is necessary to take into account this interaction including feedbacks.

Hence, it is necessary to use first-principles-based physics models in closed coupling with statistical and/or phenomenological models and satellite and ground-based observations. Because of the complexity of the system it is practically impossible to join all physical domains in a monolithic model code.

That is why the universal framework tool is needed that would provide the simple integration of independently developed models of different processes and phenomena for studying of the coupling and inter-dependencies between them. This tool should control the data flow between the data sources and models; and between different models, as well, without being dependent on the internal structure and data processing methods of the particular model.

Currently there are several frameworks under development in the area of geophysics (*Toth et al.*, 2005; *Hill et al.*, 2004; *Allen et al.*, 2000; *Buis et al.*, 2003). But most of them represent programming kits to build a model system from scratch. Other ones require powerful supercomputers or distributed multiprocessor systems for operation.

For the present moment a simple ready-to-run instrument with moderate hardware requirements is not available for the common researcher.

Our system is intended to fill this gap. Its basic structure includes the first-principles-based physics model UAM (*Namgaladze et al.*, 1988; 1998), which describes the upper atmosphere, ionosphere and plasmasphere of the Earth as a coupled system. The already adjusted system of physical interrelations provides the researcher with a physical modeling environment in which he can place his own model for studying external interdependence of the investigated phenomena. Besides, our framework system can run on the conventional desktop computers, what expands the application's scope of use.

2. Objects hierarchy of framework model

The framework model is being created with the object-oriented approach. The UAM structure was transformed into the system of interacting objects.

An objects hierarchy (Fig. 1) and obligatory functional and interface specifications have been developed for each object. Compliance to these specifications allows the object to be connected to the framework model.

The Metamodel – the frame structure itself – is at the top level of hierarchy. Its functions are to organize the interaction of connected objects and to provide the user with a comprehensive control facility.

The objects of the next hierarchical level are the models (or sub-models). They are connected to the Metamodel. Functionally each Model is a method of obtaining the numerical values of a certain physical parameter set in certain spatial grid nodes (as it is easy to see, the experimental data sources fall into this formal definition as well). Thus, the parameter set and the grid position are obligatory properties of the Model, and they are described by the objects of the next level: the set of the Parameter objects and the Grid object.

The Parameter class describes the type of stored physical parameter, its name, units, the method it was obtained, etc.

Another object class connected to the Metamodel is the Dataset. It contains an array of parameter numerical values in grid nodes, reference to the model which calculated these parameters, and the timestamp. But there is an important feature: any Model can include only one set of Parameter objects and only one Grid object. These properties are the own internal characteristics of the measuring device named "Model": they answer the questions "what" and "where" it can measure-calculate. But the number of the measurement results – Dataset objects, connected to the same Model – is not limited in either way.

One more object connected to the Metamodel is the Data Depository. It "knows how" to save the Datasets to the disk file and to retrieve them from there with complete information about the Model that provided them.

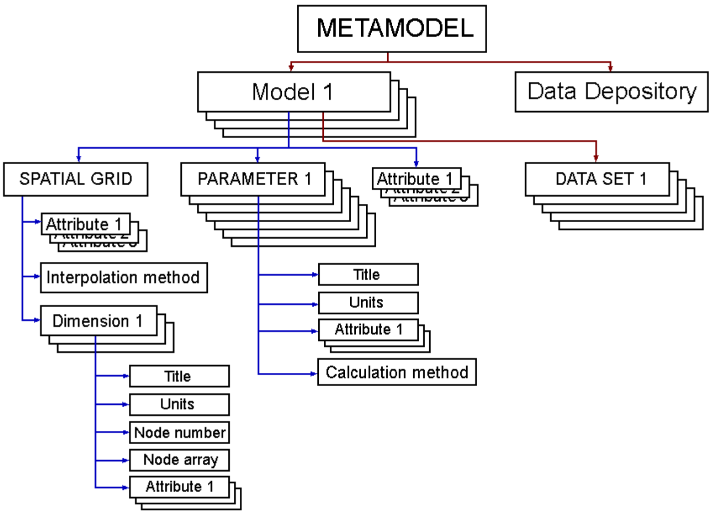

Fig. 1. Object hierarchy of framework model

3. The Grid object

The key element of the framework system is the Grid object. Its purpose is to describe the spatial position of the Model grid nodes – spatial points for which the Model obtains the physical parameter numerical values. The main aim of the developed structure is to provide the means to describe and use a maximally wide range of possible spatial grids.

Generally, the different Models use various methods of choosing the nodes position (from regular equidistant grids to randomly allocated node sets; and from one-dimensional up to three or more dimensions). Accordingly, the specific Grid object implementation can also be different, and the choice of it is the internal business of the Model developers. The mandatory requirements of the framework system to the Model Grid object concern only the most general properties of a Model space area and the external data interfaces.

The spatial characteristics include the Model number of dimensions (one-, two-, or three-dimensional modeling area), the names of coordinates (dimensions), the limits for each of them and units in which they are stored, the quantity of nodes, etc. Knowledge of these characteristics is required for other Models to understand whether they could get their necessary input parameters from this Model output or not.

The mandatory external data interface includes the ability to extract the value of calculated parameters in a point with any specified coordinates within the spatial area limits described by the given Model. On the other hand, the Grid object should "know" how to correctly interpret the input parameters required by the Model that are not provided in the nodes of "native" Model grid (for example, calculated by the other Model). Thus, these interface requirements cover all input/output operations of the Model.

In the framework structure a special data transfer technique has been developed for data exchange between Models. It uses the nodes of an intermediate grid that any connected Models must understand. Such a mandatory grid is described by means of three (in case of three-dimensional space) ordered one-dimensional arrays of node coordinates (variable steps between the nodes are allowed), one for each spatial dimension. As the framework model is oriented for the description of physical processes in the upper atmosphere and ionosphere of the Earth, the following coordinate systems are accepted as known by default: both spherical geographical and geomagnetic, and geomagnetic dipole. The Metamodel "knows" how to make a transfer of data from one of these systems into another. Each of the connected Models should "understand" the input data set at least in one standard grid, and also should know how to return the data in nodes of one of the standard grids (if necessary - to interpolate them from nodes of a "native" grid).

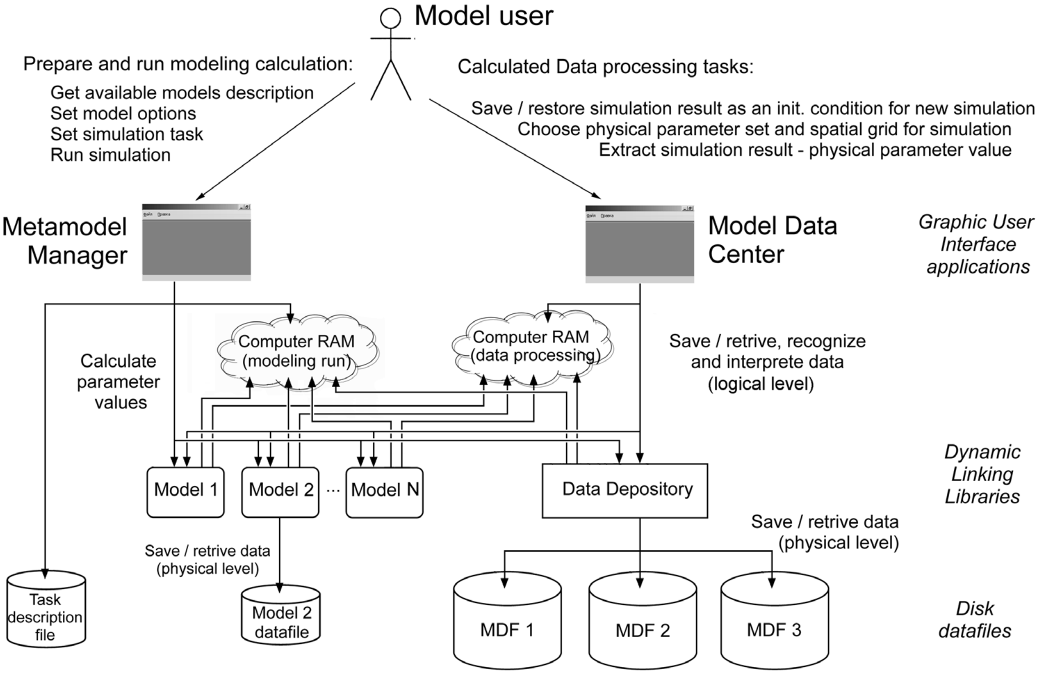

Fig. 2. Framework model objects interaction scheme

4. Framework model elements interaction

The interaction of the above-mentioned objects is carried out in the following way (Fig. 2).

The user can interact either with the Metamodel (through the Metamodel Manager) to control the calculation process, or (through the special application called the Data Control Center) immediately with the Depository for the analysis and processing of earlier received data – that is, the previous modeling calculations results. Also with the help of the Data Control Center the user prepares the initial conditions for the new modeling calculation.

To prepare the new calculation the user creates a new modeling "world" using the Data Control Center: the user defines physical parameters to be modeled and spatial grids where these parameters should be calculated. A special model data file (MDF) is being designed for these purposes. Its structure contains the complete description of data and grid and a full set of data values (it is a projection of the above-described object data model to the internal file structure). After creation of the modeling "world" (new MDF) the user by means of the same Data Control Center fills the "world" (its Datasets) with initial conditions – parameter distributions at the initial time moment from which the modeled environment evolution under specified external forcings will be reproduced during the modeling calculation.

Then the Metamodel Manager can be used to configure the calculation task: to select what Models should be connected to the calculation and specify all necessary control parameters for each connected Model. The Manager gathers and provides the user with the information about all available Models and also checks the completeness and validity of the received task: a calculating Model is selected for each demanded parameter, and each Model has all the necessary information for the operation. The calculation task includes also the list of data which should be saved during the calculation: those calculated parameters, in what areas of space (cross-section, profile, etc.), in what kind and how often they should be saved, etc.

During the model calculation the Metamodel Manager controls the process: it runs the connected Models, and transmits the necessary data to them, including data received from other Models. The Models return the calculation results to the Manager in the form of physical parameter values in the grid nodes. The actual method of obtaining these values is the internal matter of the Model and does not concern the Manager at all. Models cannot cooperate directly, and exchange any information only through the proxy of the Manager. The Manager also provides all interaction with the Depository: it reads the initial conditions and stores the current state of the modeled environment. The real-time saving of the selected task parameters is also carried out by the Metamodel Manager.

The main and mandatory modeling calculation results are the spatial distributions of all physical parameters (their values in grid nodes) at the final moment of time, which are the instant snapshot of the modeled environment. Any intermediate distributions of parameters and timeline of their modification are saved only if the distributions were initially specified in the calculation task.

5. Conclusion

The advantages of the proposed approach include the following: 1) the uniform data exchange technique allows to use the output of one Model as the input for other Models; 2) the mutual interconnection of independent local Models supplies them with the quality of self-consistency of large sophisticated models if they meet the framework requirements; 3) the dynamical environment with feedback allows for the study of interference of various processes and the phenomena; 4) the standardized data description format allows for the easy integration of a wide range of data sources of different kinds, both experimental and modeled; 5) and in addition, such system organization allows for the use of the same software for data access and processing and in particular, to represent data, obtained by various methods, in a uniform way that simplifies their comparison and analysis.

References

Allen G., Benger W., Goodale T., Hege H.-C., Lanfermann G., Radke T., Seidel E., Shalf J. The Cactus code: A problem solving environment for the grid. *In: 9th IEEE International Symposium on High-Performance Distributed Computing, IEEE Computer Society Press, Los Alamitos, CA.*, p.25-32, 2000.

Buis S., Declat D., Gondet E., Massart S., Morel T., Thual O. PALM: A dynamic parallel coupler for data assimilation. *Paper presented at EGS-AGU-EUG Joint Assembly, European Geophysical Society, Nice, France*, 2003.

Hill C., DeLuca C., Balaji V., Suarez M., da Silva A., and the ESMF Joint Specification Team. The architecture of the Earth System Modeling Framework. *Computing in Science and Engineering*, v.6, p.18-28, 2004.

Namgaladze A.A., Korenkov Yu.N., Klimenko V.V., Karpov I.V., Bessarab F.S., Surotkin V.A., Glushchenko T.A., Naumova N.M. Global model of the thermosphere-ionosphere-protonosphere system. *Pure and Applied Geophysics*, v.127, p.219-254, 1988.

Namgaladze A.A., Martynenko O.V., Namgaladze A.N. Global model of the upper atmosphere with variable latitudinal integration step. *International Journal of Geomagnetism and Aeronomy*, v.1 (1), p.53-58, 1998.

Toth G., Sokolov I.V., Gombosi T.I., Chesney D.R., Clauer C.R., De Zeeuw D.L., Hansen K.C., Kane K.J., Manchester W.B., Oehmke R.C., Powell K.G., Ridley A.J., Roussev I.I., Stout Q.F., Volberg O., Wolf R.A., Sazykin S., Chan A., Bin Yu, Kota J. Space Weather Modeling Framework: A new tool for the space science community. *Journal of Geophysical Research*, v.110, A12226, doi:10.1029/2005JA011126, 2005.