データベースシステム 123－5
（2001．1．22）

オブジェクト指向による人工衛星・地上観測データ解析参照システムの構築とネットワークデータベースの導入

村田健史、矢原大幹、豊田浩平
〒790-8577 松山市文京町3 愛媛大学工学部
TEL 089-927-9970 FAX 089-927-9852
E-mail murata@cs.ehime-u.ac.jp

Software design via object-oriented methodology and network database for solar-terrestrial observation data

Takeshi MURATA, Hiromasa Yahara, and Kohei Toyota
Department of Computer Science, Ehime University
3 Bunkyo-cho Matsuyama 790-8577 Ehime
TEL 089-927-9970 FAX 089-927-9852
E-mail murata@cs.ehime-u.ac.jp

In collaboration with the NASA, the ESA and the ISAS, an international project has started, which is called "the ISTP project". In the project, many satellites are launched and observe the geo-space and the interplanetary region. The more scientific satellites are launched, with higher quality and quantity the data provided by the satellites are. The programs to analyze and plot the data become complex as more data are included into the programs. A new breakthrough software system is required that refers to and analyzes such a variety type of data. In the present paper, we propose a system that is named "Solar-Terrestrial Analysis and Reference System (STARS)". It works with a variety type of spacecraft data and ground-based observation data. In construction of the STARS, we apply an object-oriented software methodology that is called "Object Modeling Technique (OMT)". How and for what purpose we apply the OMT to the system is discussed.

Key words: satellite data, OMT, object-oriented methodology, network database

1. INTRODUCTION

To obtain global understanding of the interactions and energy transfer between the sun, solar wind, and the magnetosphere of the Earth, multi-point observations are crucial. A lot of scientific satellites have been launched for the purpose of observations in the geo-space and interplanetary regions. Through the analysis of the simultaneously observed data, we investigate physical processes in the regions.

This type of "big project" is hard to be conducted by only one or few countries: it requires collaborations between many countries for the financial, technological, and physical reasons. Each institute launches its own satellites, then observed data are gathered, assembled, and compared. The ISTP (International Solar Terrestrial Physics) project started for the purpose of such multi-point observations. The NASA, the ESA, and the ISAS collaborate in a set of solar-terrestrial missions to be carried out during the 1990s and into the next century [1].

The satellites of the ISTP project bring us a large amount of significant data. It would progress our understandings of space and terrestrial phenomena. However, this large amount

has, on the other hand, become a barrier in the researches. Researchers tend to take long time to construct their programs to analyze the data and make plots. This inconvenience is not negligible now. We need to achieve a new methodology that can provide us with easier programmings.

The present paper proposes a software system for multi-point observation data. The system is named "Solar-Terrestrial data Analysis and Reference System (STARS)". It works with data provided by the ISTP satellites; e.g., GEOTAIL [2], WIND and POLAR. It also works with ground-based observation data; e.g. magnetograms, aurora images, and geomagnetic indices. It is crucial to combine these different type of data in a single software for integrated studies. In the construction of the STARS, we apply an object-oriented methodology called Objected Modeling Technique (OMT) proposed by James Raumbaugh and his co-workers.

2. DATASET AVAILABLE IN THE STARS

The STARS is a system that refers to and analyzes a variety of data. In this section, before description of the STARS, we start with an introduction of the data used in the STARS. In the STARS, various data are included. From the viewpoint of dataplot, most of the data are categorized into the following types: color dot plots, line plots, in numbers, and graphics data.

Color dot plots are used for data observed by multi-channel detectors. The intensity is represented by a color dot at each time and at each channel. For example, GEOTAIL PWI/SFA and GEOTAIL PWI/MCA are in this format. Both data provide us with a dynamic spectrum of plasma waves.

Line plots are used for the data that has one value at each time. In the STARS, many data are shown in this format: GEOTAIL/MGF, GEOTAIL Orbit, AE, Dst and Ground-Based observation data. GEOTAIL PWI/WFC is an instrument that observes a raw plasma wave forms. GEOTAIL MGF provides three components of magnetic fields. GEOTAIL Orbit is shown in the three-dimensional orthogonal coordinates. AE is an index of magnetic activity in the auroral region. Dst is an index to represent an energy of the ring current around the Earth. Ground-Based observation data correspounds to magnetograms which are the geomagnetic data observed at each ground-based station. Some indices are given in form of number. The Kp index gives 8 hour value of the global geomagnetic activities of the Earth.

Graphics data are the data provided in form of graphic files. In the STARS, POLAR/PWI/SFR, POLAR/VIS and WIND/RAD1 are given in graphic file format, not in raw data format.

3. OBJECT MODELING TECHNIQUE

In the construction of the STARS, we adopt a software development methodology that is called Object Modeling Technique (OMT) [3]. The OMT has advantages to the other software methodologies (e.g., [4]) due to its simplicity of maintenance and improvement, reuse, and long time development [5]. The OMT makes developers and researchers easy to communicate each other. This development method consists of the following four steps: (1) Analysis, (2) Object Design, (3) System Design, and (4) Implementation. Before implementation of a program, we start with analysis and design of the system. In this chapter, we describe the process of the STARS construction along with each step shown above.

3.1 Analysis

3.1.1 Request

First of all, developer must obtain users' requests. Herein, the "users" corresponds to "researchers". The researchers make their own request sheets so that the developers can understand what to construct. Next, the developers construct a real-world model from the request sheets. This model is significant to create a system since the real-world model shows what the system is. In the STARS, we make a request sheets as shown in Fig3. Here, we have described, for example, how to get data, plot data, and use various devices.

3.1.2 Object model form request sheet

In the OMT, the developers make three models, which are called an object model, a dynamic model, and a functional model. The object model shows what the system is. The dynamic model shows when the system works. The functional model shows how the system works. We describe these models in the STARS in the following sections respectively.

3.1.2.1 Three models

The object model is the most important model in the three models. It is because this model shows what the system is. The object model shows a static structure of the system. One object model is composed of several classes. Each class is consists of a class name, attributes and methods. An object figure is usually used to show relationships between classes in the object model.

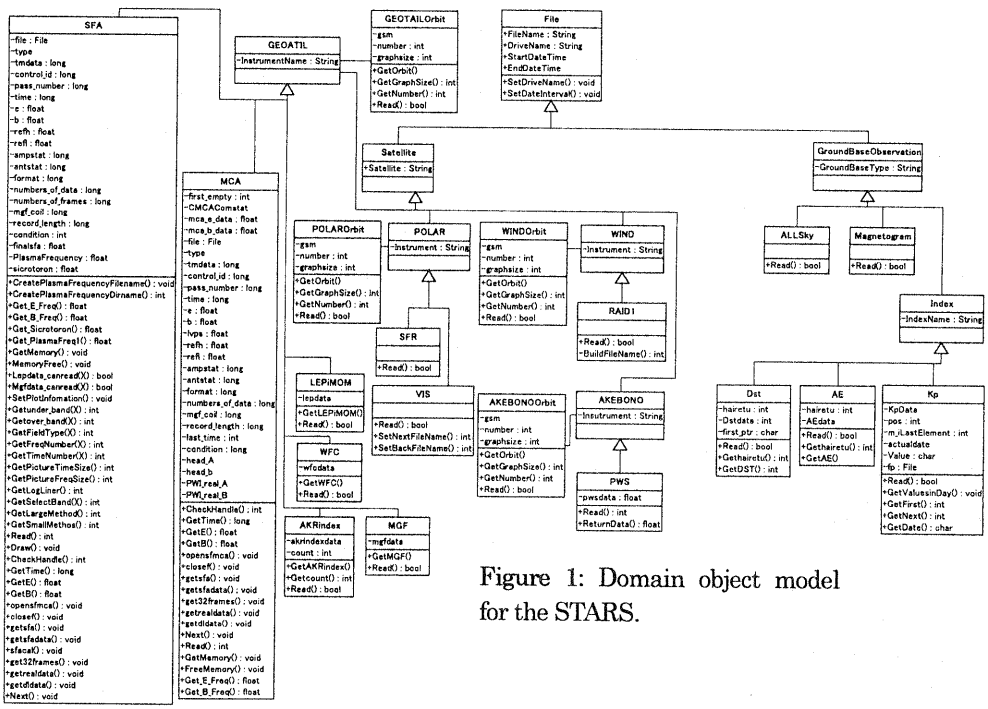

Figure 1: Domain object model for the STARS.

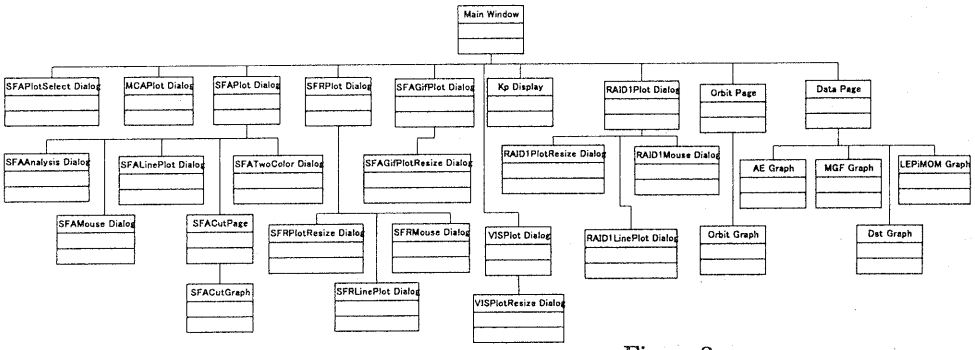

Figure 2
Application object model for the STARS.

An object model usually consists of multiple models depending on the roles of the object. In the STARS, we create three types of object models; a domain object model (Fig.1), an application object model (Fig.2) and utility object model (not shown herein). The domain object model is the most essential model among three models since it shows a static real-world. The application object model is a software model. The utility object model is a model that contains utilities available over the system such as an FFT transform. The three object models are not depicted in the form of UML [6] since they are rather simple in their structures.

3.1.2.2 Dynamic Model

In the object model, the developers have shown static relationships in the system. They next make a dynamic model which shows dynamic relationships in the system. The dynamic model is described by 'events' and 'states.' In the dynamic model, two models are used; a state figure and an event trace figure. The state figure represents state transitions. The event trace figure is a table which has time axis. Each event between different objects is put in time order.

3.1.2.3 Functional Model

Finally the developers make a functional model. The functional model shows an internal activity in the system. This model describes how to analyze input data without time chart. In the implementation of system, all function are not to be modeled as functional models. Since the function model is close to the program itself, we don't show our function models in the present paper.

3.2 System Design, Object Design, and Implementation

The developers perform high-level determination of the system on the stage of system design. They divide a system into subsystems. Here, they fix resources to be used in the system; hardwares and softwares. Fig.3 is the software resource in the STARS. In the STARS, we adopted a graphic library named Quinn-Curtis [7] because of its flexibility.

On the stage of the object design the developers need to complete more detailed determination about the analysis models discussed above. They must decide data structure and algorithm in the system. This design strongly depends on the developers themselves. In the present paper, therefore, the object design in the STARS is not to be shown.

After all of the modelings and designs, the developers start implementing the software. In the

OMT, the implementation is considered to be simply a translation of a model into a real program. Since the program is automatically done along with the designs discussed above, new developers can easily understand the program codes.

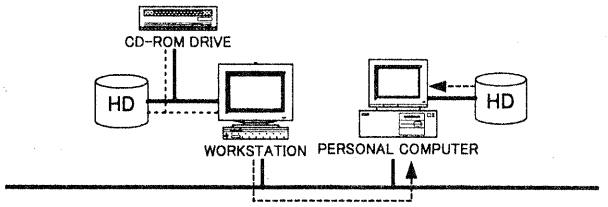

Figure 3: STARS hardware resources. The STARS software works on the PC, which is connected with networks.

4. DISCUSSIONS

In the present study, we design a system that is named "STARS (Solar-Terrestrial data Analysis and Reference System)". The STARS provides a research environment that researchers can refer to and analyze a variety of data with single software. This software design is based on the OMT (Object Modeling Technique). The OMT is one of the object-oriented techniques, which has an advantage in maintenance improvement, reuse and long time development of a system.

In our laboratory (Faculty of Engineering, Ehime University), after our designing of the STARS, we have already started implementing the STARS. A demonstration version (version 1.0) is released in 2000. Any user can download the system from our WWW site (http:// sp.cs.ehime-u.ac.jp).

The present paper is mainly devoted to the design of a data analysis software system. Through our designing, we paid attention so that the design is flexible and applicable when other developers design a software for the similar purpose. If our model is so particular only for our own purpose, it would be useless for other developers.

Through our design of the domain object model, we carefully removed the parts which depend on the system resources, e.g. hardware and software. We put the dependent parts into the application object model. In the present design, therefore, the domain object model and the utility object model are independent of computer resource. This helps anther developer to construct their own system based the present design. They simply modify their own application object models according to their system resource. This division of the design between dependent and independent part into three object models is one of the advantages in the OMT.

If the design of software is completely done along with the OMT, implementation is rather simple and automatic: developers simply map their designs on our programs. If one creates "another STARS" with other programming language such as Java, the programmer simply follows the present system as long as the language is object-oriented language.

Researchers would want to add their data into the STARS. In this case, they simply add their own data class in the domain object model. It is because any satellite data has properties such as time or date, which are inherited from the upper class (see File class in Fig.1). In this way, their effort is less than in other old methodologies.

In the OMT, description format of the system is rather strictly standardized. When new developers take part in STARS project, they have only to understand each model to obtain the overview of the STARS. Then they follow this designs and documents to implement the system.

The OMT makes a new comer easy to join into the project already running.

References:

(1) http://www-istp.gsfc.nasa.gov/
(2) http://www.isas.ac.jp/j/enterp/missions/geotail/index.html
(3) J.Rambaugh, M.Blaha, W.Premerlani, F.Eddy, and W.Lorensen, "Object-Oriented Modeling and Design", Prentice Hall, 1991.
(4) E.Yourdon, "Object-Oriented Systems Design An Integrated Approach", A Simon & Schuster Company, 1994.
(5) J.Rambaugh, "OMT Insights", Sigs Refernce Library, 1996.
(6) G.Booch, J.Rumbaugh, and I.Jacobson, "The Unified Modeling Language User Guide", Addison Wesley, 1999.
(7) http://www.quinn-curtis.co