Object-Oriented Programming When Developing Software in Geology and Geophysics

R K Ahmadulin and L N Bakanovskaya

Industrial University of Tyumen, 38 Volodarskogo st., Tyumen, 625000, Russia

E-mail: ark@tsogu.ru

Abstract. The paper reviews the role of object-oriented programming when developing software in geology and geophysics. Main stages have been identified at which it is worthwhile to apply principles of object-oriented programming when developing software in geology and geophysics. The research was based on a number of problems solved in Geology and Petroleum Production Institute. Distinctive features of these problems are given and areas of application of the object-oriented approach are identified. Developing applications in the sphere of geology and geophysics has shown that the process of creating such products is simplified due to the use of object-oriented programming, firstly when designing structures for data storage and graphical user interfaces.

1. Introduction
Modern information technologies have become part of all spheres of human activity, from organizing the educational process to optimizing the duration of production cycles[1, 2]. Nowadays, it is hard to imagine the area of activity in which a person would not have come to the aid of modern information technology in general and software in particular. One such area is geology and geophysics.

Design considerations of geophysical research, processing of the results, geological modeling, forming geological engineering reports, etc., often require significant amounts of data processing. The development itself can be performed both completely independently[3, 4], and on the basis of the existing earlier developments realized using the methods of structural and modular programming [5, 6]. However, in order to effectively use the earlier results (models, algorithms and software) it is advisable to use a more modern object-oriented programming paradigm (OOP).

The given paper based on [7] reviews the role of the principles of object-oriented programming when developing software in geology and geophysics and stages, at which it is worthwhile to apply the stated principles in the first place.

2. Research
In the period from 2002 to 2015 the authors took part in a number of projects based on the OOP technology. In the course of these projects the following was developed:

- information system of storage, processing and analysis of geotechnical data[3];
- input module for the geotechnical reports from computers that are not connected to the corporate network [4], with the possibility of uploading them to the database;
software package designed for use in the design, quality assessment and optimization of field geophysical studies for oil and gas[5];

program for solving the inverse problem of the near-field transient electromagnetic sounding method, based on the existing functional [6], etc.

Common properties possessed by the abovementioned developments are the following.

1. In most cases, as a basis has been accepted an existing mathematical apparatus implemented in an earlier programming language not compatible with the OOP paradigm (particularly - FORTRAN and C, for example [5, 6]).

2. Processed data generally have a hierarchical structure (for example [3, 4, 6]).

3. It is required to represent data in different forms, depending on the user's needs at a particular time(FOREXAMPLE[4, 6]).

4. There are cases where, depending on the specific situation, the data set should be significantly or slightly different(FOREXAMPLE[3, 4, 6]).

5. For the user the most convenient is the graphical user interface including the “tree” that displays a set of hierarchical data(FOREXAMPLE[4, 6], figure 1).

![Figure 1](image.png)

Figure 1. An example of the program interface including the “tree”.

Based on the work [7], we will consider the experience of using object-oriented programming principles to:

a) process data;

b) store and transmit data;

c) display data on the screen.

To process data in geological and geophysical problems experts frequently use complex algorithms often developed on the basis of the author's techniques and already implemented in the form of programs. For example, the result of [8] became programs implemented in FORTRAN using the structural and modular approaches. As noted in [5, 6], a minus of such developments is the lack of a
user-friendly graphical interface, which urges their re-implementation (development) with the use of modern tools.

At the same time, the algorithms themselves are usually complex and require considerable time for their transcoding to modern programming languages, and further debugging. In the case of the problem in question were limited to time resources, so it was decided to leave their implementation “as is” by connecting them to developing products in the form of a library. This approach was tested, for example, in [5, 6].

Recoding the existing algorithms in modern programming languages is expedient in cases where the need to accelerate the work of programs, for example, by the methods of parallel programming, is more important than the time spent on development.

An important component of the geological and geophysical program is data storage and transmission. As noted above, such data may have a complex hierarchical structure (Figure 2). An example of such a structure is presented in [7].

![Diagram of class hierarchy](image)

**Figure 2.** An example of a class hierarchy implemented for data storage and transmission.

It is for the implementation of the hierarchical structures that the OOP paradigm was applied in the first place. Within the framework of these structures the works have used such principles as:
- inheritance, to create structures obtained on the basis of others;
- encapsulation, to combine geological and geophysical data with the methods necessary for their support and ensuring their correctness, as well as to conceal the internal service fields and methods;
- polymorphism, due to which data “variability” is provided depending on a particular task.

The program's flexibility when working with heterogeneous data was also guaranteed through the use of virtual and abstract methods.

Also, the use of classes for storage allowed a convenient data transmission to different modules for display in different formats, which was also important in the case of geological and geophysical programs.

We also used the methods of object-oriented programming to develop graphical interfaces required to display data on the screen.
It should be noted that the use of OOP to create a graphical interface is widely used in various domains, such as [7, 9]. The considered problems have some distinctive features. Geological and geophysical applications typically have a variety of graphical interfaces presenting data in different formats, often slightly differing. Another situation is the need to present different data in the same format (Figures 3, 4).

**Figure 3.** An example of a calculation program interface for solving the inverse problem of the near-field transient electromagnetic sounding method.
OOP capabilities are very useful in the development of a set of graphical interfaces, namely:
- inheritance allows you to create a hierarchy of graphical interfaces with similar features in a set of controls and / or software implementation;
- polymorphism and abstract methods provide the ability to use a single graphical interface to represent different data, different graphical interfaces to represent the same data and also to create a hierarchical tree, each node of which allows you to open the data using a particular graphical user interface;
- virtual methods allow you to specify the desired functionality for both the methods defined in parent classes, and for the standard forms methods. And the ability to override them provides the developer with a powerful tool for rapid creation of interfaces on the basis of ready-made components, but with their own functions.

Developments carried out on the basis of the OOP principles and addressed in this paper help solve a variety of problems in geology and geophysics.

The information system of storage, processing and analysis of geotechnical data [3, 4] enabled automatic provision of the urban development activity with geotechnical data. A unified electronic bank of geotechnical data, software and methodological support was created for the evaluation, monitoring and forecast of geotechnical conditions. As a result, access to the right information on the desired object can be accessed from any computer that has access to the database and installed developed software. The system has practical application.

The software package of design and quality evaluation and optimization of field geophysical studies for oil and gas [5] allows solving a wide range of problems related to calculation and analysis of the interference systems and observation networks during seismic survey. The package was used to solve many practical problems. These include construction of a model (structural map) of a reflecting

**Figure 4.** An example of a calculation program interface for solving the inverse problem of the near-field transient electromagnetic sounding method.
horizon, study of the constructed model stability and the observation network redundancy, construction of the generalized indicator of the quality of object selection, etc. Another field of software system application is its use to train geophysics students.

The program for solving the inverse problem of the near-field transient electromagnetic sounding method [6] was also used to solve industrial problems and demonstrated high efficiency on a large geological material. The algorithm was applied to interpret field data of the near-field transient electromagnetic sounding method, in the search for electrically polarized objects, etc.

3. Conclusion
Thus, the experience gained in the development of applications in the field of geology and geophysics, showed that object-oriented programming simplifies the process of creating similar programs. The main stages at which it is advisable to use these methods in this domain in the first place is the development of structures for data storage and graphical user interfaces.

Using the experience described in the work can be extended to develop programs to address environmental challenges.

References
[1] Mokeeva N S and Bakanovskaya L N 2009 Creation automatic worker of the place of the technologist for designing the technological process Programmye produkty i sistemy 4 52
[2] Bakanovskaya L N 2015 On the question of the development of neural network expert systems Materials of the XI International scientific and practical conference “Conduct of modern science” Economic science (Sheffield; Science and education LTD) 3 11-4
[3] Turenko S K, Aksarin M Yu, Morozov M V and Parhimovich R V 2008 Information system of storage, processing and analysis of geotechnical data Proc. of Institutions of Higher Education. Oil and Gas 5 12-20
[4] Ahmadulin R K, Turenko S K and Morozov M V 2010 The program for the local data input for AS INZHGI Crystals of creativity: Proceedings of the Student Academy of Sciences (Tyumen; TyumGNGU) 300-3
[5] Turenko S K and Ahmadulin R K 2015 Development of software to optimize the field geophysical studies for oil and gas Geology, geophysics and development of oil and gas fields 8 30-3
[6] Ahmadulin R K and Dmitriev A N 2015 Implementation of the program to solve the inverse problem by the ZSB method with a GUI based on the existing functionality Oil and gas of West Siberia: Materials of the International scientific and technical conference dedicated to the 90th anniversary of Kosukhin Anatoly Nikolaevich (Tyumen; TyumGNGU) 1 73-6
[7] Ahmadulin R K 2016 Experience in applying the principles of object-oriented programming in the development of applications in geology and geophysics Automation, telemechanics and connection in the oil industry 1 38-42
[8] Turenko S K 1995 Computer technology for optimization of field geophysical studies for oil and gas (on the example of seismic survey) Author’s abstract of Cand. Sc. Eng. Disseration (Moscow) 36
[9] Mokeeva N S and Bakanovskaya L N 2009 Development of an automated system software module for design process Molodoy uchenyi 3 38-41