Methodology for the Development of Industrial Analytical Systems for Data Collection and Processing

A Misnik[1], S Krutalevich[1], S Prakapenka[1], E Lukjanov[2]

[1]Inter-state educational institution of higher education “Belarusian-Russian university”, Mogilev, Belarus
[2]ZAO "GIAPDistCentr", Moscow, Russian Federation

Abstract. This article is devoted to methods of the development of industrial analytical data collection and processing systems. It is shown how to obtain a synergistic effect by combining well-known approaches and ensuring interaction between them using analytical and neural network modules.

1 Introduction

Modern enterprises, as an organizational and technical system, are becoming increasingly complex both in structure and in internal connections between structural components. The prerequisites for these processes are: expansion to foreign markets, high competition, tendency to reduce costs, ensuring the safety of production, the need for a dynamic response to market needs. Also, there are growing demands on product quality, environmental protection, as well as occupational safety and health.

Due to the development of technologies in the field of instrumentation, the amount of data available for analysis on the state of technical devices has grown significantly, besides, it is obvious, that automation of processes reduces the number of errors caused by the human factor. These aspects should be controlled using appropriate analytical data processing information systems, that ensure achievement of goals in conditions of significant uncertainties, especially in the context of long time horizons.

Currently, the growth rate of requirements for information systems and the need for their modification during operation is very high. Very often there is a need to modify the existing data structure, the way of data displaying, new data processing scenarios appear. Involving developers to solve such problems is usually costly both financially and in terms of time.

In our opinion, it is optimal to develop a system tools that allows a specialist in the subject area with the skills of an advanced Excel user (Architect) to create and modify data structure, develop applications for displaying and interacting with data, and implement data processing scenarios.

In: P. Sosnin, V. Maklaev, E. Sosnina (eds.): Proceedings of the IS-2019 Conference, Ulyanovsk, Russia, 24-27 September 2019, published at http://ceur-ws.org

2 Traditional approach to the development of analytical systems for data collection and processing

In the framework of the traditional approach to organizing industrial analytical systems for data collection and processing, each part of such a system is, in fact, a separate software module, with its own static data structure and internal logic that is rigidly defined at the stage of system design and implementation.

This approach has the following disadvantages:

- duplication of data in the system;
- the complexity, and, often, the inability to organize relationships between data in different modules;
- the need to involve developers to create new system applications, make changes to the logic and displaying data in the existing applications.

3 Methodology of the development of analytical systems for data collection and processing

It is proposed to use an object-oriented approach as the basis for organizing data structure - the subsystem “Type Tree”.

Type, in full accordance with the classical approach, is a description of the objects through their common attributes. An attribute of a type is a named property or characteristic of an object of that type. Each attribute of a type is characterized by a name unique within this type and the datatype that this attribute will store. All instances (objects) of the same type have the same set of attributes. Type instances differ in attribute values. Attributes by the values of which each object can be identified from other objects of this type are called key attributes. If there are several key attributes, a composite key is generated. It is proposed to use the following main types of attributes: string, number, counter, date, date and time, text, file. For attributes of the form string, number, date, date and time, the file should be available the ability to set the property "array". Also, an attribute may be a link to another type of system’s data structure [1].

Types should be in a hierarchical relationship and can be arbitrarily connected both with each other and with themselves if required by the data structure or business process. [3]

Such an approach allows, without the involvement of developers and database engineers, to form hierarchical data structures of arbitrary nesting by Architect with the necessary connections between hierarchy levels.

In practice, business processes rarely perfectly reflect the data structure. To create a way of data displaying for the end-user of the system, a subsystem is required that can interconnect data located at different levels of the hierarchy, and in different branches of the "Type Tree". [3]

As such a subsystem, "Applications" is proposed.

Within the framework of this subsystem, the capabilities to configure the displaying of data both in the form of modal forms and in the form of tables for viewing and editing information should be realized. The range of settings should allow the formation of a wide variety of business processes.

User interfaces should have a hierarchical structure that provides consistent output of related data.

This approach allows to abandon the pre-configuration of all possible options for using the business process.

In the industrial analytical systems for data collection and processing it is not enough just to input and output information. A key feature of data processing systems oriented to use by engineering and technical personnel is the need to provide the user with analytical data based on the information available in the system.

To process data and generate analytical information, it is proposed to use the subsystem "Analytics", which should have an intrasystem meta-programming language that allows, first of all, to seamlessly operate data in the system in the best possible user-friendly form, mainly using system dialogs - "masters".

In order, to reduce the user entry threshold for interacting with this subsystem and increase the level of security, it is not recommended to use traditional programming languages, but to develop unique internal meta-programming language. This meta-language should implement the principles of linear programming, including basic algorithmic constructs, have access to the data structure of the subsystem "Type Tree" in a user-friendly form.

Practice shows that the main need of users when working with industrial analytical systems for data collection and processing is to obtain the necessary data, simply process it and record the results. Any algorithmically complex operations for the user can be implemented by programmers within the framework of functions connected to the “Analytics” subsystem. [2]

The application of this approach will allow Architect to form business logic for data processing within the system without involving developers.

The big problem of analytical systems for data collection and processing is the quality of the data, both system inputs and results of analysis. For input data, the problem is exacerbated if they are fully or partially entered by operators. As for the analytical data obtained by calculations, in addition to the risks associated with the source data, the risks of imperfection of the calculation method affect their accuracy.

Existing approaches, such as, for example, double-entry of information or constant monitoring of its entry, are quite expensive both financially and in terms of time. Such solutions are not always reasonable and effective.

One of the possible solutions of the quality problem of source and analytical data in industrial systems is to use the supervisor neural network module, which able in real time to check the data changing in the system. The implementation of such a “Supervisor” in each case should be individual, therefore it is most convenient to implement it in the form of a certain designer tool, within which the Architect can choose the network architecture and topology, as well as input and output parameters. If there is a possibility of incorrect data entry, the supervisor informs the operator about the need to verify the entered data, if the probability of error is recognized in the analytical data, a message is sent to the expert or system Architect. For additional training, the neural network "Supervisor" monitors the response of users to messages sent to them about probable errors. Also, training can take place under the supervision of an Architect. [4]

Using the neural network "Supervisor" allows to improve the quality of the input data and strengthen control over the results of analytical calculations. [5,6]

To summarize, we can present the following simplified diagram of the interaction of subsystems within the industrial analytical systems for data collection and processing (Figure 1).

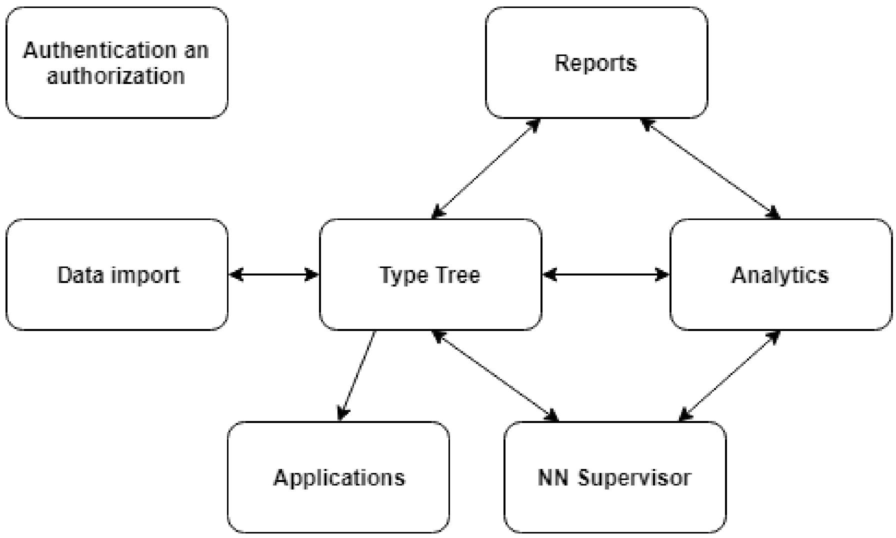

Figure 1. Subsystems interaction

In addition to the described above, the scheme contains the subsystems “Authentication and authorization”, “Data import” and “Reports”, the purpose of which is obvious.

4 Example of methodology implementation

ZAO “GIAP-DISTcenter” was established in 1995, the main activity is to ensure industrial safety of hazardous production facilities.

When carrying out large projects, such as the creation of control systems for technical devices, including as part of unique work to increase the intervals between overhauls of oil and gas and chemical complexes, the forces of relevant leading institutes and expert organizations of the country are consolidated to fulfill the assigned tasks.

To ensure the reliability and safety of industrial plants, it is necessary to take into account the accumulated scientific knowledge and some existing approaches, supported by good engineering practice, which are included in international standards and guidelines.

The developed methodological documents, such as ICTE 1-002-14, ICTE 3-002-14, ICTE 3-003-14, as well as software created by own IT department, allow to optimize the process of analyzing the actual condition of the equipment, timely identify critically dangerous objects and provide industrial safety.

Based on many years of experience and analysis of existing systems, it was decided to develop own industrial analytical systems for data collection and processing, which allows to solve specific problems in the field of ensuring industrial safety of the enterprise, as well as in the field of information support of various business processes in the enterprise. The modularity and flexibility of the system allows to quickly adapt it for implementation at a particular enterprise.

GIAP-DIST CENTER, an industrial information system for data collection and processing, is a client-server application that can be accessed both from stationary computers and mobile devices, both from a local computer network and via the Internet, which significantly increases the flexibility of working with it and also increases the efficiency of access to the necessary data.

This system includes the following main subsystems:

- a subsystem for managing users and user groups, responsible for providing users with access to the system and setting their privileges;
- a subsystem for constructing a type tree responsible for the formation of a data structure;
- a subsystem for constructing a tree of objects that allows managing the data entered into the system;
- a subsystem for configuring and displaying user applications, which is responsible for setting up and ensuring the functioning of the business processes of the system;
- a subsystem of units of measurement, providing the conversion of data from one dimension to another;
- a data conversion subsystem that allows bringing complex-structured data, for example, received from diagnostic devices, into the desired form;
- reporting subsystem responsible for generating documents in the system;
- a subsystem for constructing two-dimensional and three-dimensional schemes, providing visualization of system objects;
- a data import subsystem that allows loading data into the system from various sources;
- subsystem of analytical calculations;
- neural network supervisor.

A general view of the user management subsystem is shown in Figure 2. In addition to standard user operations, this subsystem allows configuring user rights for any system objects, export and import users and their settings, and view user interaction logs with the system. [7]

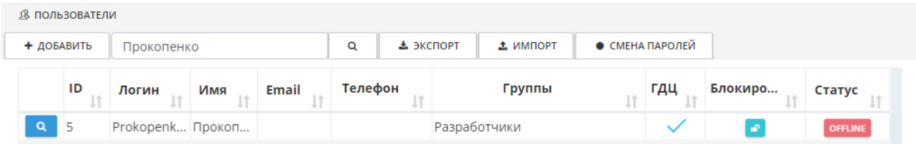

Figure 2. User management subsystem

The general view of the subsystem "Type tree" is shown in Figure 3.

This subsystem allows building a universal data storage system, without the involvement of developers.

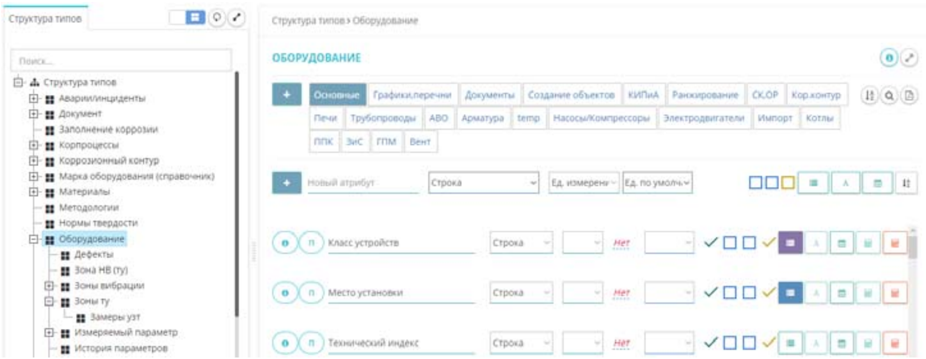

Figure 3. Type Tree subsystem

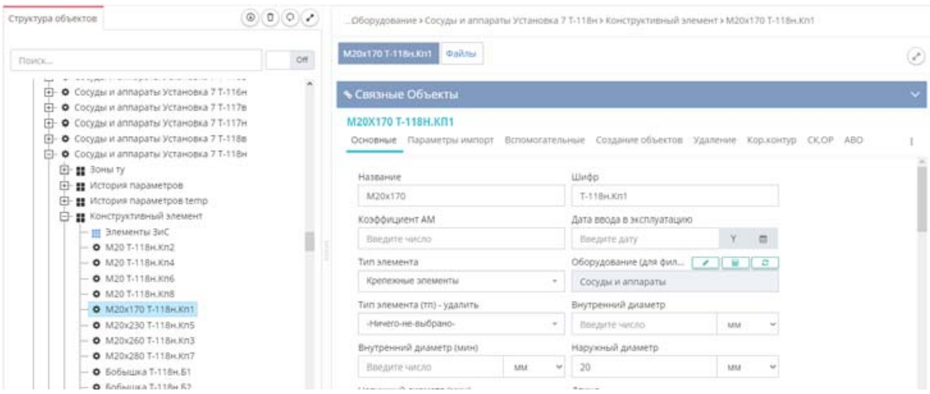

Figure 4. Type Tree subsystem

To view all created objects in the system, the subsystem "Object Tree" has been developed. This subsystem allows to track the status of all objects in the system, view the interconnections of the object, and edit it if necessary.

Objects are presented in a tree structure, which makes it easy to analyze the relationship of the object. A general view of the subsystem "Object tree" is shown in Figure 4.

A general view of the Applications configuration subsystem is shown in Figure 5, examples of applications are shown in Figures 6 and 7.

The tabular display of the application has the functionality of sorting, filtering, coloring data, and also allows for multiple operations with data.

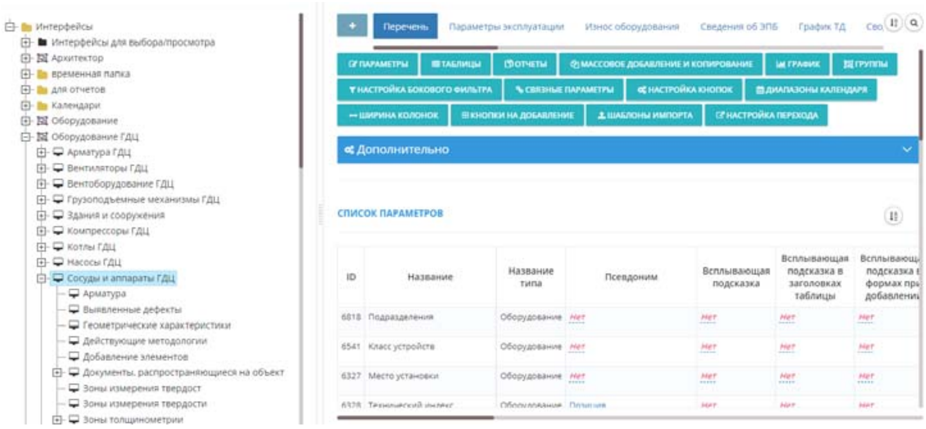

Figure 5. Application subsystem

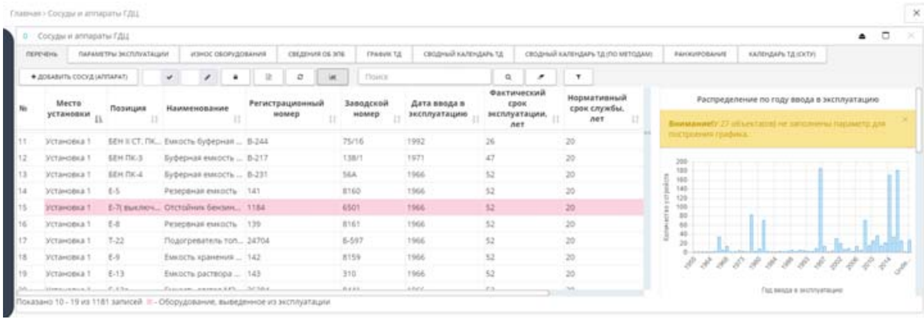

Figure 6. Application table form

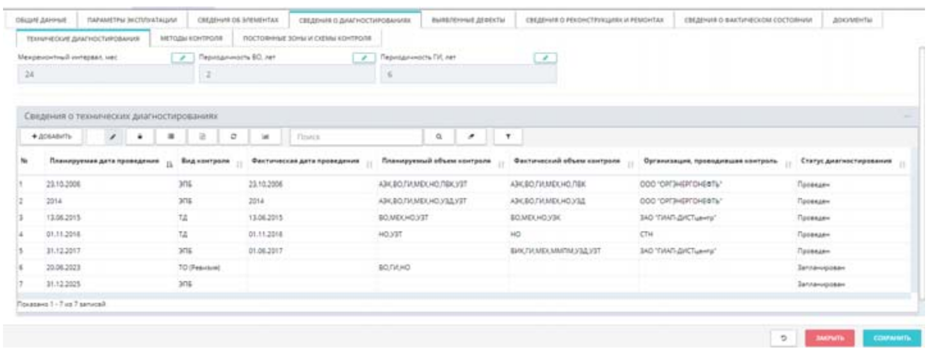

Figure 7. Application combined form

Figure 8 represents a general view of the editor of the subsystem "Analytics". Within the framework of this subsystem, an internal linear programming language is implemented, as well as the following sets of functions:

- mathematical functions;
- logical functions;
- functions of interactions with dates (getting today's date, getting the difference between dates, setting the date, getting the maximum/minimum date);
- statistical functions;
- functions for working with strings;
- conversion functions (converting a string to a number, converting Arabic numbers to Roman, and vice versa)).

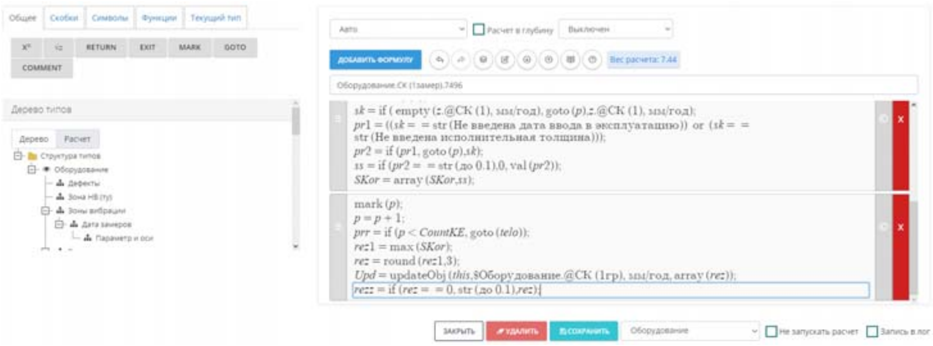

Figure 8. Analytics subsystem

5 Conclusion

In this paper, we describe the issues of developing industrial analytical systems for data collection and processing, the internal structure of such systems. Also, an example of the implementation of the methodology is considered.

References:

1. Bertrand Meyer. Object-oriented software construction, volume 2. Prentice hall New York, 1988.
2. Kim Wald´en and Jean Marc Nerson. Seamless object-oriented software architecture.Prentice-Hall, 1995.
3. Misnik Anton, Krutalevich Siarhei, Intelligent decision support in an industrial data collection and processing system. XVI All-Russian Scientific Conference "Neurocomputers and their application." Abstracts. 2018 p. 204-205
4. Prakapenka Siarhei, Misnik Anton, Decision support system for analysis risks of failure of technical systems. XVI All-Russian Scientific Conference "Neurocomputers and their application." Abstracts. 2018 p. 205-206

5. Gorbunov-Possadov M.M. Extensible programs. Moscow, Poliptih, 1999, 336 p. (in Russian).
6. Andrew S. Tanenbaum, Maarten van Steen. "Distributed Systems. Principles and paradigms". Prentice Hall, Inc., 2002
7. Nancy G Leveson, Mats Per Erik Heimdahl, Holly Hildreth, and Jon Damon Reese. Requirements specification for process-control systems. Software Engineering, IEEE Transactions on, 20(9):684–707, 1994.
8. Gustavo Alonso, Fabio Casati, Harumi Kuno, Vijay Machiraju. Web Services. Concepts,Architectures and Applications. Springer-Verlag, 2004