

Unified Approach to Software Development of Automated Control Systems for Oil Equipment

Aleksey N. Iliukhin^{1*}, Shafik Sh. Khuzyatov², Rustam A. Valiev³

^{1, 2, 3} Kazan Federal University, Naberezhnye Chelny Institute,

^{*}*Email: ANIliukhin@kpfu.ru; 89274544457*

Received: 21st October 2017 Accepted: 16th November 2017, Published: 31st December 2017

Abstract

We proposed a decision to reduce the volume of works performed when developing the automated process control systems for oil production, the complexity of which is due to a large number of parameters and data, as well as their diverse interrelationship. The implemented pattern-software design is based on the uniformity of algorithms for processing sensor data and generating control signals for the actuators at the oil production facilities. In the developed template projects, we predetermined the structure and functionality of the software of both the lower and the upper levels of the automated control systems. As the elements of a low hierarchical level, we adopted the same type of project objects such as analog signals, latches, pumps, liquid and gas meters. For them, we identified the data types, ways of their storage and organization of the inter-level exchange, controls functionality in the template projects of the lower and upper levels. The predefined architecture greatly simplifies the development of software for the specific automated control systems, shortens the design and implementation time. Template projects for the oil production facilities are developed on the basis of hardware and software automation tools of the Company Siemens AG. The universal nature of the approaches of the proposed methodology makes it possible to use the template design of the automated control systems in other industries as well.

Keywords: Automated Control System, Software, Template Design, Dispatching Control, Oil Equipment

Introduction

It is used a large number of sensors and actuators in the automation of oil production facilities, the data from which are used in the software of both lower and upper levels of the automated process control systems (APCS). Due to the intensive information flow from the process equipment, as well as their complex interrelationship, the creation of the APCS software becomes a complex task that requires high skill of programmers and a large amount of time.

In many oil production facilities, such as a preliminary water discharge unit, booster pump station, gas turbine unit for associated petroleum gas utilization, the algorithms for processing sensor data and generating control signals for the actuators are the same. From the point of view of automation, the difference from each other in the number of actuators and sensors is characteristic for these objects. This makes it expedient

to use template projects in which the software structure and functionality of the lower and upper levels are predetermined [1 - 3].

The use of modern software development tools for the controllers and SCADA-systems greatly simplifies the process of creating a control system [4, 5]. But with a large number of sensors and actuators, the software development for the APCS requires a lot of time. At the same time, it should be noted that the information processing and data exchange via the network between the user program of the controller and the human-machine interface software takes place in a real time mode [6].

Methods

The development of template projects is based on a pattern technique that allows ranking and analyzing objects in such a way to imagine a complex interconnection of the system elements [7-9].

According to this technique, the same type of project objects, such as analog signals, latches, pumps, liquid and gas meters, which have the same algorithms for data processing and generation of control signals [10, 11], are adopted as the elements of a low hierarchical level. For them, we identified the data types, ways of their storage and organization of the inter-level exchange, controls functionality in the template projects of the lower and upper levels [12, 13].

Template projects for the oil production facilities are developed on the basis of hardware and software automation tools of the Company Siemens. The software for data collection on the process parameters, automatic control and regulation (lower level) is implemented in Step 7 system, and for monitoring and process dispatching control (upper level) - in WinCC system in SCADA [4]. At the same time, the exchange organization between the lower and the upper level of the APCS occurs through the use of built-in drivers, which significantly increases the control system reliability.

Results and Discussion

We created the structured data types for each type of sensor or actuator. In addition to input and output data, the fields of these structural types include some parameters that are used to implement algorithms for both generating control signals and data processing.

The template projects executed in Step 7 and WinCC systems, these structural data have the same number of fields, the types of which coincide in terms of the data storage method. This is implemented on the basis of

choosing the data types - BOOL, INT, REAL - in Step 7 system and the corresponding data types - BIT, WORD, FLOAT - in WinCC system [14, 15].

Data storage of the same type of actuators and sensors in the lower level template project is implemented in the data block. It is an array of objects of a structural type. Due to the data block application, the organization of exchange between the software of the upper and lower levels is greatly simplified. This enables the exchange of data blocks between WinCC and Step 7 systems.

The structural tags of a corresponding type are grouped together to store the data of each type of actuator and sensor in an upper level template project. The structural tag address is the offset in the data block. It is equal to the offset for the array element from the lower level software. Due to such data organization, the array elements correspond to the structural tags at the exchange between the upper and lower level software.

The functionality of single-type objects of template projects includes automatic control of technological equipment, alarm when technological indicators deviate for certain boundaries, monitoring and dispatching equipment control.

In the lower-level template project, the generation of control signals, as well as the data processing of the same type of actuators and sensors, is performed using the corresponding arrays created in the project. The

data processing algorithm for a group of similar objects is implemented as a function block or function that is called in the organizational block of cyclic interrupts. In this case, the calling frequency of this block should meet the requirements of response time for the occurring changes in the oil production facilities, which is established when adjusting the controller parameters. The functions or function blocks are implemented in the SCL language, which allows processing the array elements cyclically unlike other languages (FBD, LAD, STL and SFC) supported by Step 7 system.

In the upper level template project, we implemented the functionality for monitoring and dispatching process control, as well as for setting up the equipment parameters during commissioning, tag archiving, alarm messages, and report generation. The upper level software is a hierarchical system consisting of the main process screen (object screen) and the subordinate process screens (status screens).

The objects screen contains graphic objects of typical sensors and actuators, for example, analog signals, latches, pumps (Figure 1). Current state of these objects is displayed in the form of various color effects, which is achieved by dynamizing the object properties using the Dynamic Dialog window or by cyclic execution of VB- or C-scripts.

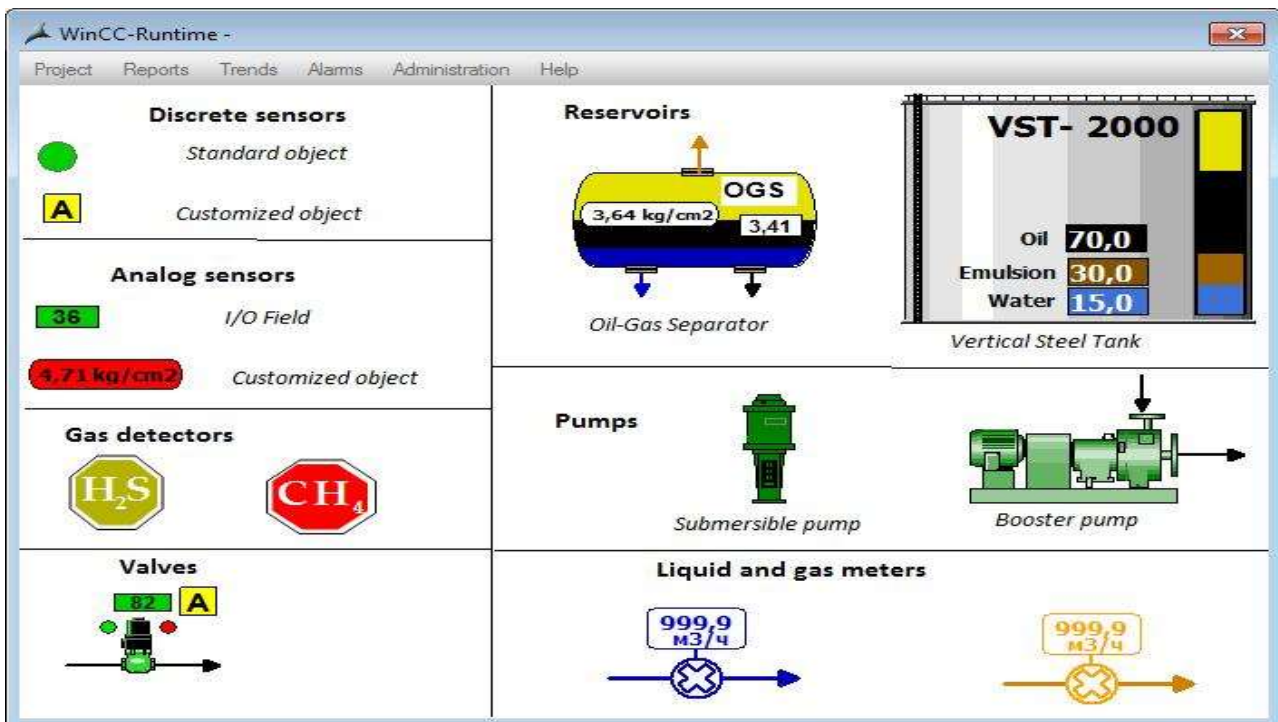


Fig. 1 An Example of Objects Screen of the Template Project

For each type of sensor or actuator of the object screen, we created the process screens - the status screens, which are subordinate to them, that display the values of the input and output signals of the object in real time, signal change trends and alarms. The status screen also allows configuring the parameters of the selected object in the process of commissioning,

process diagnostics and dispatch control. The analog signal status screen on the example of the oil and gas separator pressure sensor is shown in Figure 2. For the information perception convenience, the values of all analog signals are displayed in percent relative to the maximum value. This transformation is carried out using the C-script of value linearization.

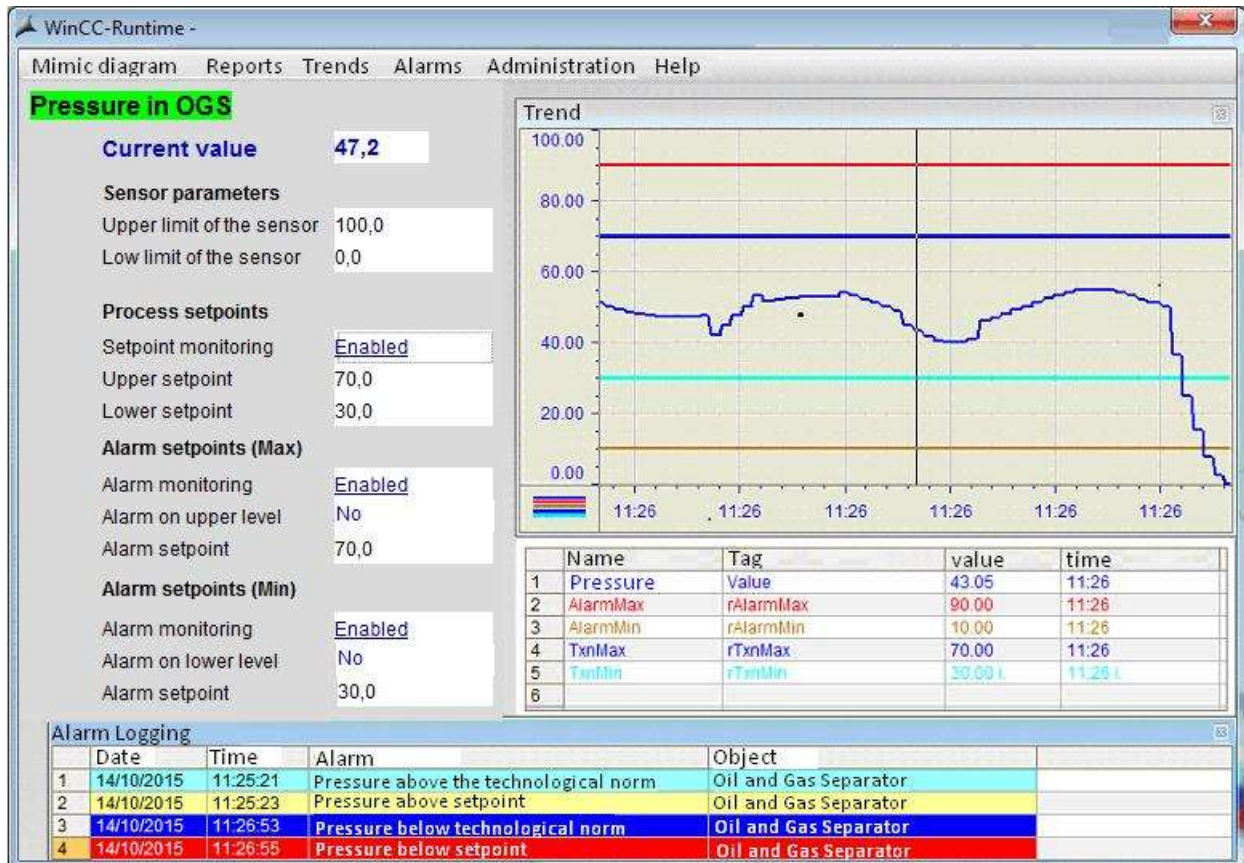


Fig. 2. An example of status screen of analog signal

The display of the parameters and data of the sensor or the actuator in the input/output fields is implemented in the VB script, which is called as the event handler for the graphic object. The process tag name is formed in the code for this script by the name of the graphic object, and this name is assigned to an indirect tag that is attached to the input/output field of the status screen. A filter for the alarm display window is similarly displayed.

The development of upper level software for a particular oil production facility begins with creating a copy of the template project using the Project Duplicator utility. At the same time, the process screens and scripts of the template project are transferred to the upper level project being developed.

The main process screen for a particular oil production facility is developed in the form of a mnemonic process diagram by copying graphic objects of typical sensors and actuators from the object screen, which occurs with the preservation of the properties of graphic objects and related scripts. If necessary, the status screen of a particular sensor or actuator is called by the event handler of the graphic object on the mnemonic diagram.

The mnemonic diagram of the process displays all objects of the control system and their basic characteristics. Figure 3 shows, as an example, a simplified mnemonic diagram of a preliminary water discharge unit, developed on the basis of a template project.

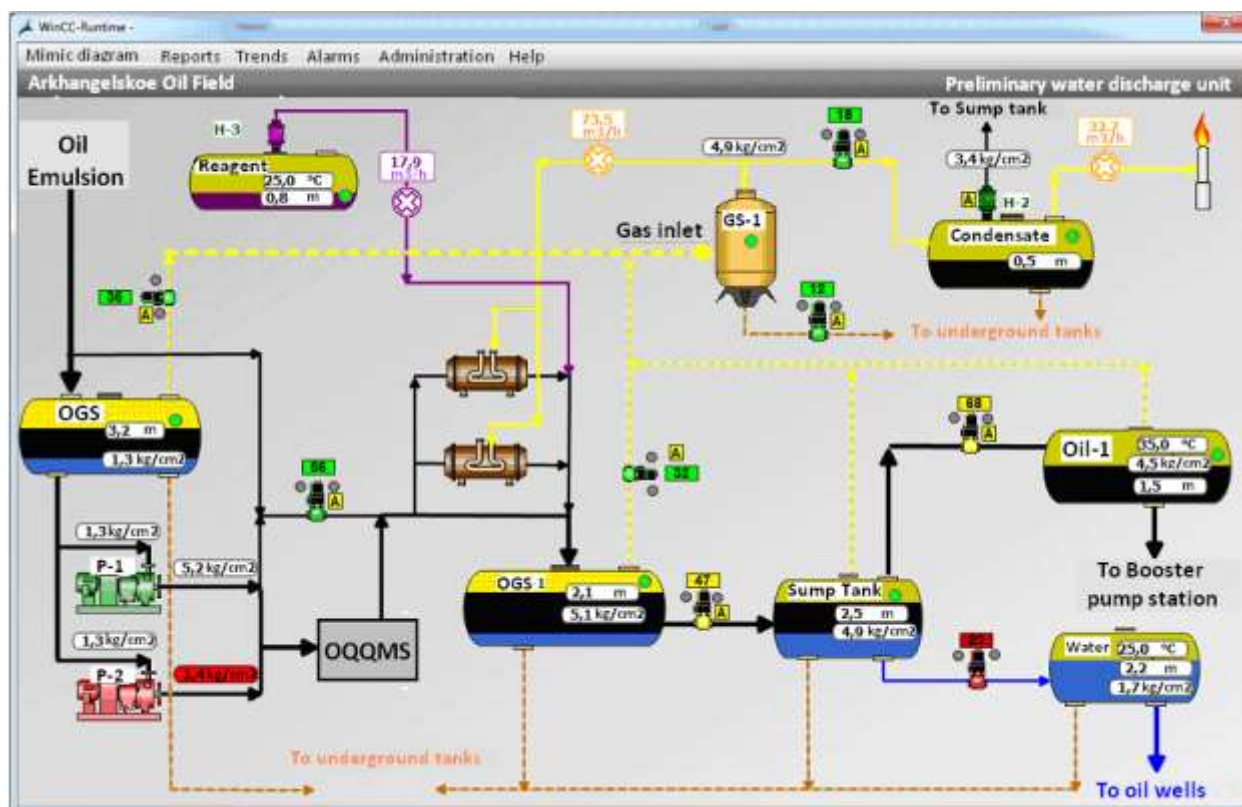


Fig. 3. Simplified Mimic Diagram of the Preliminary Water Discharge Unit

Commands of the custom menu of the template project allow displaying the trends of the main technological parameters, generating reports in the form of Excel documents and diagnosing the state of equipment. To solve new tasks of monitoring and analyzing the state of technological process, it is possible to expand and improve the functionality of the upper level template project by creating the new user menu commands.

Conclusions

The developed technique greatly simplifies the process of designing software for the automated process control systems with a large number of the same type of automation elements.

The structure and functionality of the software of the automated process control system are predetermined in the template projects created on the basis of this methodology. Predetermination of the functionality of controls, the ways of data storage, as well as the organization of exchange between the lower and upper level software, greatly facilitates the adaptation of these projects in accordance with the requirements of a particular control system.

The lower level software is created on the basis of a template project, while it is necessary to determine the number of elements of a corresponding array for each type of actuator and sensor. When creating the upper level software using a special utility, a template project is first copied, and then the required number of tags is created and a process mnemonic diagram is developed using the graphical objects of typical sensors and actuators.

Summary

Template design has shown its effectiveness in the development of APCS software for various oil production facilities. At the same time, the project cost is significantly reduced by reducing the development and implementation time, as well as using less skilled software developers. The universal nature of approaches makes this technique universally valid for the automation of various industrial facilities.

Acknowledgements

The work is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

References

1. Ibragimov N.G., Zabbarov R.G., Idiatova V.R. Operational supervision and oil production control system based on monitoring of telemetry-controlled well performance in ARMITS corporate information system // Oil Industry, 2014. – Iss. 4. – pp. 106-109.
2. Firoozshahi A. Intelligent and innovative monitoring of LAR GE tank farm based on DCS // Source of the Document Proceedings of the IASTED International Conference on Automation, Control, and Information Technology - Control, Diagnostics, and Automation, ACIT-CDA, 2010. – pp. 203-210.
3. Butta R. An overview of oil drilling and production monitoring system using SCADA automation in Oiland Natural Gas Corporation

- Ltd. // International Conference on Electrical, Electronics, Signals, Communication and Optimization, EESCO 2015, September 2015. – A.n. 7253920.
4. Reeser J., Jankowski T., Kemper G.M. Maintaining HMI and SCADA systems through computer virtualization // IEEE Transactions on Industry Applications, 2015. – Vol. 51. – Iss. 3. – pp. 2558-2564.
5. Wang Q.-C., Hu L.-K. Research on WinCC-based SCADA software for acrylic fibres filature // Huagong Zidonghua Ji Yibiao, Control and Instruments in Chemical Industry, 2006. – Vol. 33(1). – pp. 35-38.
6. Khuzyatov Sh.Sh., Valiev R.A. Network exchange control between SIEMENS controller and microprocessor devices based on the Modbus protocol. // Modern problems of science and education. – 2015. – No. 1. <http://www.science-education.ru/121-19411>
7. Khalil M. Pattern-based methods for model-based safety-critical software architecture design // Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI), 2013. – pp. 493-499.
8. Valiev R.A., Khairullin A.Kh., Shibakov V.G. Automated Design Systems for Manufacturing Processes // Russian Engineering Research. – 2015. – Vol. 35(9). – pp. 662-665.
9. Valiyev P.R.A., Galiullin L.A., Iliukhin A.N. Approaches to Organization of the Software Development // International Journal of Soft Computing. – 2015. – Vol. 10(5). – pp. 336-339.
10. Khuzyatov Sh.Sh., Valiev R.A., Sitdikov I.M. Development of the APCS software system for oil production on the basis of template projects // Achievements, problems and prospects for the development of oil and gas industry. Materials of the International Scientific and Practical Conference (Almetyevsk, October 28-29, 2016). In 2 Volumes. V. 2. - Almetyevsk: AGNI Publishing House, 2016. – P. 275 – 278.
11. Sitdikov I.M., Farkhetdinov R.S., Jakupov F.A. Building automation systems using the principle of partitioning functions // Modern Automation Technologies, 2011. – Iss. 1. – pp. 54-59.
12. Khuzyatov Sh.Sh., Valiev R.A. Design of automated control systems based on the pattern technique // Scientific and Technical Bulletin of the Volga Region. – 2015. – No. 2. – P. 215–218.
13. Valiev R.A., Khuzyatov Sh.Sh. Pattern-design software of automated control systems. 2nd International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2016 – Proceedings. URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7910942&isnumber=7910885>
14. Berger H. Automating with STEP 7 in STL and SCL: SIMATIC S7-300/400 Programmable Controllers, 6th Edition. 2012. – P. 553.
15. SIMATIC HMI. WinCC V7.2: Working with WinCC. System manual. Siemens AG, Industry Sector, Nurnberg, Germany. 2013. – P. 2216. <https://cache.industry.siemens.com/files/WinCC.pdf>