

# The Approach to Solving the Problem of Legacy Systems on the Examples of INTEC and OPTCON Software

Alexander Gornov

*Matrosov Institute for System  
 Dynamics and Control Theory  
 of Siberian Branch of the  
 Russian Academy of Sciences  
 Irkutsk, Russia  
 gornov@icc.ru*

Liudmila Massel

*Melentiev Energy Systems  
 Institute of Siberian Branch of  
 the Russian Academy of  
 Sciences  
 Irkutsk, Russia  
 massel@isem.irk.ru*

Pavel Sorokovikov

*Matrosov Institute for System  
 Dynamics and Control Theory  
 of Siberian Branch of the  
 Russian Academy of Sciences  
 Irkutsk, Russia*

Anton Anikin

*Matrosov Institute for System  
 Dynamics and Control Theory  
 of Siberian Branch of the  
 Russian Academy of Sciences  
 Irkutsk, Russia*

Tatiana Zarodnyuk

*Matrosov Institute for System  
 Dynamics and Control Theory  
 of Siberian Branch of the  
 Russian Academy of Sciences  
 Irkutsk, Russia*

**Abstract**—The paper proposes a methodological approach to the construction of modern software systems, as well as the creation of computing resources available via the Internet. The presented approach has been tested in practice. In the article it is considered on the example of the implementation of the integrated software INTEC for the study of the problem of energy security. Proven methods and technologies are also used to solve the problem of legacy software OPTCON for optimal control problems, which was also used to solve applied problems from various scientific and technical fields.

**Keywords**—*legacy software, client-server architecture, optimal control problem, energy security.*

## I. INTRODUCTION

In recent years, the development of information technology is a clear example of the fact that the development lines of application software systems and the Internet, or rather, Web technologies, tend to combine into a single open system that integrates both components [1, 2]. The need to use open system capabilities is due, in particular, to the increasing use of a variety of hardware platforms running applications designed to solve an ever wider range of problems. The introduction of heterogeneous systems and the desire to share data between such systems have led to the need for their work together. In addition, compatible binary computers promised software portability from platform to platform.

The impulse to use open system capabilities comes not only from users. It comes from both developers concerned about the need to reduce costs

and time to migrate their applications to various platforms, as well as from manufacturers trying to develop new platforms on which a wide range of applications can run.

The term legacy software is used in cases when it comes to software that, due to the development of information technologies, has ceased to meet the requirements of modern basic hardware and/or software or due to some factors is morally obsolete. Transferring programs from single-user and single-task operating system DOS, within which a large number of programs have been implemented, to the operational space of the Internet is one of the most popular tasks. This is primarily due to the transition from individual work with software to projects distributed in time and space.

In fact, the entire software potential accumulated over the last 15–20 years turned out to be unclaimed, firstly, due to the lack of support for compatibility of modern operating systems with DOS, and secondly, due to the failure of users from outdated interfaces. The accumulation of a mass of legacy software became especially noticeable in scientific institutes in connection with the departure of most of the developers of software packages.

## II. THE PROBLEM OF LEGACY SOFTWARE

A natural way to solve the problem of legacy software is to rewrite programs using debugged algorithms for new hardware and software architectures, which is not possible, at least in the near future, due to a large number of legacy software and the lack of qualified personnel. The situation is

aggravated by the fact that during the crisis of domestic science, the source codes of many programs were lost.

Thus, it can be stated that:

- 1) there are a large number of legacy software,
- 2) there is a need to use it,
- 3) it is impossible to rewrite most of this software.

Therefore, it is required to set the task of finding a universal way to use legacy software.

Generally speaking, the problem of legacy software arises in the world of information technology every time a revolutionary change in technology.

One of the most popular solutions to the problem of legacy software is the use of middleware. The main idea of the approach is the creation of some intermediate layer between the legacy software and the end-user or some other software. In other words, such a layer will provide the required user or program interface and hide all details of "communication" with legacy software.

When creating software systems INTEC and OPTCON (which will be discussed below) the authors also had to deal with the legacy software problem. In the case of INTEC, system such a legacy component was a program which solves the mathematical programming problems. In the course of this work, the basic principles of creating programs based on legacy software were formulated [3, 4].

Let's start with the analysis of the structure of any program. All programs can be logically represented as three interrelated parts:

1. Functional part – contains the logic of the program and implementations of all internal algorithms.
2. Program interface (API) – contains a set of «public» functions with defined parameters and return values.
3. User interface (UI) – contains a set of any external manifestations of the program execution. Naturally, we will not consider extreme cases when, for example, the program manifests itself only by the sound of the computer speaker or by flashing of the keyboard LEDs. Space of the program user interface includes any information transmitted to the user or received from the user.

Based on this, we can reformulate the problem of legacy software as follows: access to the functional part of the program by using other software or user interface.

### III. THE TECHNOLOGIES FOR SOLVING THE LEGACY SOFTWARE PROBLEM

Historically, one of the first industrial technologies aimed at solving the considered problem was the CORBA technology (Common Object Request Broker Architecture) proposed by the OMG consortium (Object Management Group). Then came the "proprietary" technologies RMI (Remote Method Invocation) from Sun Microsystems company and COM/DCOM (Component Object Model/ Distributed COM) from Microsoft Corporation.

The basis of all technologies is the same mechanism for the remote calling of object procedures (Object Remote Procedure Call – ORPC), the scheme of which is presented in Fig. 1.

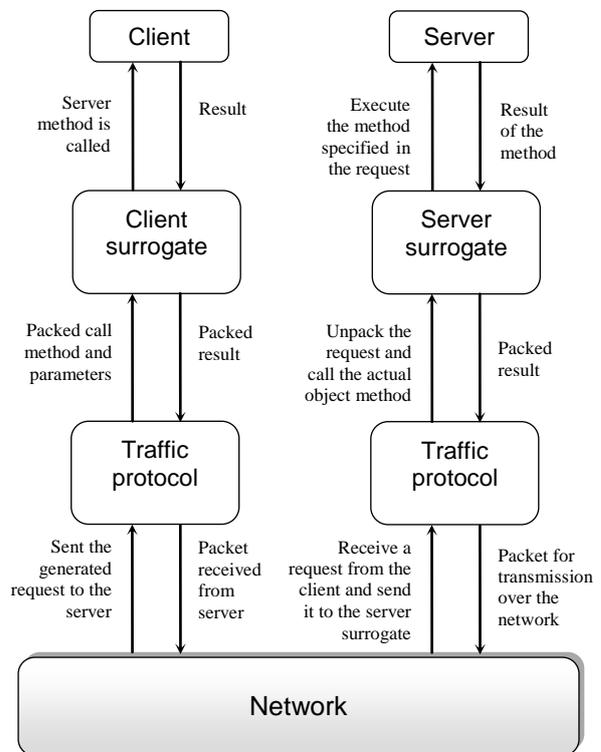


Fig. 1. The scheme of the ORPC mechanism.

The application of the above technologies to solve the problem of legacy software is very similar and consists of several consecutive steps:

1. Determining the composition of the server interface and describing it in the interface specification language of the selected technology (CORBA, RMI, COM/DCOM, or others).
2. Creating a wrapper application server in a language that supports the selected technology (CORBA, RMI, COM/DCOM, or others). The client's applications will interact with the server application (in accordance with Fig. 1).
3. Generating the client surrogate and the server surrogate for the server application.

4. Writing the actual client applications using the implemented server.

A simplified architecture of such the system is shown in Fig. 2. The client sends a request to the server application, which converts it to a form that the legacy program understands. After the query is executed, the legacy program returns the result to the server application, which it converts into a form that the Client understands.

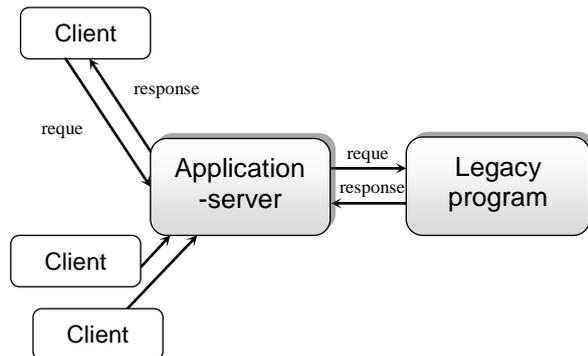


Fig. 2. Software architecture providing access to legacy software

The main problem when working with legacy software implemented in the scientific environment is its solidity (that is, in fact, mixing the user interface and the computational core), which significantly complicates its integration into modern software.

The OPTCON software package (Section 3.1) is a console-based dialogue application running in the "question-answer" mode. Accordingly, the simplest way, in this case, is to create middleware, which, on the one hand, would provide the RMI interface, and on the other hand, could access OPTCON using software channels that are standard in Unix and Windows OS to call a mid-level program led to the sending of an appropriate message to OPTCON.

Using the terms of the client-server architecture technology, in the proposed scheme for solving the problem of legacy software, the following three levels can be distinguished:

- the data level: the inherited software;
- the level of subject logic: the middleware;
- the presentation level: software that implements the user interface.

The correspondence between the levels in the client-server architecture and the possible architecture of the software that uses legacy software as the main computing core is shown in Fig. 2. Generally speaking, in such a scheme there can be not three, but more levels. The division into levels may imply the degree of the software distribution.

#### A. OPTCON Software

OPTCON, focused on the study of optimization problems of controlled dynamic systems, was used as legacy software. The first version of OPTCON [5] was implemented in C language for the MS-DOS operating system and was handed over to the customer in 1990. This version was used to test the legacy software methodology developed by the authors.

The name of the long-term OPTCON project on the numerical solution of a wide class of optimization problems originates from the abbreviation of the terms "OPTimal CONTROL" since the first of the implemented software systems was focused on solving optimal control problems. The project is created at Matrosov Institute for System Dynamics and Control Theory of SB RAS (formerly the Irkutsk Computing Center of the East-Siberian Branch of the USSR Academy of Sciences) since 1988. This organization also implements MAPR projects ("Mathematical programming in multidimensional problems" for BESM-6, the head of the project was Prof. A.I. Tyatyushkin [6]) and KONUS ("Integrated optimization of nonlinear controlled systems" for ES EVM, the project leader was Ph.D. A.I. Zholudev) [7, 8]. The OPTCON project is the development of these projects implemented for IBM-compatible computers [9].

Within the OPTCON project, the following software systems have been developed: OPTCON-I for nonlinear optimal control problems with terminal and phase constraints [5]; Computing server OPTCON-II for solving optimal control problems (OCPs) in remote access mode via the Internet [10]; OPTCON-MD for problems of phase estimation and approximation of the reachable set [11]; OPTCON-III for non-linear non-convex OCPs [12]; OPTCON-F for the study of functional differential equations of pointwise type [13] and others. The functional content of the OPTCON series software is composed of multi-method algorithms based on both the theory of finite-dimensional optimization and the theory of optimal control.

Using the software OPTCON, a lot of applied problems were solved from various fields, in particular, the optimization problem of the silicon-germanium quantum structure (the Keating potential minimization) with more than 20 million static variables [14] was solved. The nonconvex optimization problem of the molecular structure for Morse potential with a record dimension of 240 atoms (720 variables) [15], the problem of optimization quantum dots in microwave field [16] was investigated by considerable software.

#### B. INTEC Software

INTEC PC [17] is used to study the development directions of the fuel and energy complex at the regional and federal levels.

INTEC Software was designed and implemented in a three-tier client-server architecture:

- The first level is the user interface level, it is responsible for presenting information to the user and maps actions into function calls.
- The second level is the level of domain logic that implements the main functionality of the application.
- The third level is the level of resources. Usually, it has a SQL server, legacy systems, and other necessary applications.

The interaction between the first and second levels is organized using RMI technology, between the second and third levels – using the SQL language. The main advantages of this architecture are scalability and a high degree of reuse [18], as well as flexibility, portability, and customizability [19].

In INTEC, the user interface is a program implemented using the Swing classes of the Java language, which implies the use of Sun's java-plugin on client computers. In the implementation of the OPTCON PC, it was decided to apply the user interface in the context of an Internet browser, which allows for greater mobility in the operation of a PC.

The application of the component approach involves the design and implementation of software-based PCs so that later you can use the components of this PC to implement other PCs. The meaning of the component approach is to create programs from small components within a cohesive environment and then use these components for other projects.

As a classic example of a component approach to programming, Sun's Enterprise Java Beans (EJB) technology can be cited. EJB components operate in a so-called EJB-container and can send messages to each other by calling methods or have shared.

#### IV. CONCLUSIONS

The rapid development of information technologies has led to the fact that in recent years the situation in the field of developing software and information support has changed significantly. The spread of the Internet has served as an additional incentive to spread the object approach and the concept of open systems. The problems of cross-platform portability have become actual, the component approach to the design and implementation of software has become common.

The paper proposes a methodological approach to the construction of modern software systems, as well as the creation of computing resources available via the Internet. The proposed approach has been tested in practice. In the article, it is considered on the example of the implementation of the integrated software INTEC for the study of the problem of energy security. Proven methods and technologies are also used to solve the problem of legacy software OPTCON for optimal control problems, which was also used to solve applied problems from various scientific and technical fields.

#### ACKNOWLEDGEMENTS

The work is partially supported by the Russian Foundation of Basic Research, grant No 18-07-00587.

#### REFERENCES

- [1] N. Prokofyeva and V. Boltunova, Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems, *Procedia Computer Science*. 104(2017) 51–56.
- [2] M. Tekdal, S. Sayginer and F. Baz, Developments of Web technologies and their reflections to education: a comparative study, *J. Educational and Instructional Studies in the World*. 8(1)(2018) 17–27.
- [3] L.V. Massel and V.A. Boldirev, Simulation and development of modern software for energy research, *J. Comput. Tech.* 4(4)(2002) 59–70. [Modelirovanie i razrabotka sovremennyh programmyh kompleksov dlya issledovaniy energetiki, *Vychislitel'nye tekhnologii*, 4(4), 2002, 59-70] (In Russian)
- [4] L.V. Massel, A.Yu. Gornov and D.V. Podkamenniy, Creating computational resources on the Internet based on legacy software, Joint issue of *J. Comput. Tech.* 7(2002) and *Vestnik KazNU*. 4(32) based on Proc. Int. Conf. "Computational Technologies and Mathematical Modeling in Science, Engineering, Education" (Novosibirsk-Almaty, 2002), pp. 247–252 [Sozдание vychislitel'nyh resursov v Internet na osnove unaslodovannogo programmno obespecheniya, *Vychislitel'nye tekhnologii*, 7, 2002, Vestnik KazNU. 4(32), Novosibirsk-Almaty, 2002, Pp. 247–252](in Russian).
- [5] A. Yu. Gornov and A. O. Divakov, OPTCON software for solving optimal control problems: user guide (Irkutsk, 1993).
- [6] R. Gabasov, A. I. Tyatyushkin, A.I. Zholudev, et al., Application software package "Mathematical programming of multidimensional problems", *Algorithms and programs: inform. bullet.* 2(71)(1986) 33. [Paket prikladnyh programm "Matematicheskoe programmirovaniye mnogomernykh zadach", *Algoritmy i programmy*, 2(71), 1986, 33] (In Russian)
- [7] A. I. Zholudev, A. I. Tyatyushkin, N. M. Erinchek, Numerical methods for optimization of controlled systems, *Izv. Academy of Sciences of the USSR. Tech. cybernetics*. 4(1989) 14–31. [CHislennyye metody optimizatsii upravlyaemykh sistem, *Izvestiya AN SSSR. Tekhnicheskaya kibernetika*, 4, 1989, 14-31] (In Russian)
- [8] A. I. Tyatyushkin, SPP KONUS for optimization of continuous controlled systems. Application packages: experience of use (Science, Moscow, 1989). [PPP KONUS dlya optimizatsii nepreryvnykh upravlyaemykh sistem, *Pakety prikladnyh programm: Opyt ispol'zovaniya*, Nauka, Moskva, 1989] (In Russian)
- [9] A. Yu. Gornov, Technology for solving optimization problems of continuous dynamic systems based on the OPTCON software package, in *Simulation of non-equilibrium systems: mes. reports V Vseros. a seminar.* (Krasnoyarsk, October 18–20, 2002), pp. 50–51. [Tekhnologiya resheniya zadach optimizatsii nepreryvnykh dinamicheskikh sistem, osnovannaya na komplekse programm OPTCON, Modelirovanie neravnovesnykh sistem: Tezisy dokladov V Vserossiyskogo seminapa, Krasnoyarsk, 50-51] (In Russian)
- [10] L. Massel, A. Gornov, T. Zarodnyuk, Internet-technology for remote user support OPTCON, in Vth International workshop "Critical infrastructures: Contingency management, Intelligent, Agent-based, Cloud computing and Cyber security" (IWCI 2018), Ser. *Advances in Intelligent Systems Research*, 158 (2018) 124–128.
- [11] A. Yu. Gornov, T. S. Zarodnyuk, E. A. Finkelshtein and A. S. Anikin, The Method of Uniform Monotonous

- Approximation of the Reachable Set Border for a Controllable System, *J Glob Optim*, 66(1) (2016) 53–64.
- [12] T. S. Zarodnyuk, A. Yu. Gornov, The basic components of the OPTCON-III software for solving nonconvex optimal control problems, in Proc. XVI Baikal All-Russian Conf. "Information and Mathematical Technologies in Science and Controls". Part III (2010), pp. 202–208. [Bazovye komponenty programmnoogo kompleksa OPTCON-III dlya resheniya nevyuklykh zadach optimal'nogo upravleniya, Informacionnye i matematicheskie tekhnologii v nauke i upravlenii, Chast' III, 202-208, 2010] (In Russian)
- [13] T. S. Zarodnyuk, A. S. Anikin, E. A. Finkelstein, A. L. Beklaryan and F. A. Belousov, Technology for solving boundary value problems for nonlinear systems of functional-differential equations of pointwise type, *Modern technologies. System analysis. Simulating*, 1 (49)(2016) 19–26. [Tekhnologiya resheniya kraevykh zadach dlya nelinejnykh sistem funkcional'no-differencial'nykh uravnenij tochechnogo tipa, Sovremennye tekhnologii. Sistemnyy analiz. Modelirovanie, 1 (49), 2016, 19-26] (In Russian)
- [14] A. S. Anikin, A. Yu. Gornov, Computational technologies for solving the problem of optimizing the Keating potential of dimension  $10^7$  variables, "Lyapunov readings" (2012) 4. [Vychislitel'naya tekhnologiya resheniya zadachi optimizacii potenciala Kitinga razmernosti  $10^7$  peremennykh, Lyapunovskie chteniya, 4, 2012] (In Russian)
- [15] A. S. Anikin, A. Yu. Gornov, A. N. Andrianov, Architecture and Functional Content of the OPTCON-M Software for Optimization Potentials of Atomic-Molecular Clusters, in Proc. XVIIIth Baikal Conf. "Information and mathematical technologies in science and management." Part III (2013), pp. 213–219. [Arhitektura i funkcional'noe napolnenie programmnoogo kompleksa OPTCON-M dlya zadach optimizacii potencialov atomno-molekulyarnykh klasterov, Informacionnye i matematicheskie tekhnologii v nauke i upravlenii, Chast' III, 213-219, 2013]
- [16] A. F. Zinovieva, A. V. Nenashev, A. A. Koshkarev, T. S. Zarodnyuk, A. Yu. Gornov and A. V. Dvurechenskii. Quantum Gates with Spin States in Continuous Microwave Field, *Russian Microelectronics*, 47(4) (2018) 268–278. [Kvantovye logicheskie operacii na spinovykh sostoyaniyah v nepreryvnom SVCH-pole, Mikroelektronika, 47(4), 268-278, 2018] (In Russian)
- [17] L. V. Massel, N. N. Makagonova, V. V. Triputina, et al., Decision Support System for Energy Security, *Izv. RAN "Energy"*, 6(2000) 40–48. [Sistema podderzhki prinyatiya reshenij po obespecheniyu energeticheskoy bezopasnosti, Izvestiya RAN "Energetika", 6, 40-48, 2000] (In Russian)
- [18] S. Wilson, B. Maples, T. Landgrave, The principle of designing and developing software. Training course MCSD: translation from English (Publishing house "Russian edition", Moscow, 2000).
- [19] T. Lewis, The big software chill, *IEEE Computer*, 3( 1996) 12–14.