IRASA International Scientific Conference
SCIENCE, EDUCATION, TECHNOLOGY AND INNOVATION
SETI VII 2025
Book of Proceedings
https://doi.org/10.62982/seti07.slve.68

ENABLING A SECURE G2G SCENARIO THROUGH THE WRAPPER OF A THREE-DECADES-OLD LEGACY SYSTEM

Slavimir Vesić[310]

Abstract

In the era of digitization, modern government services enable simpler services for its citizens. At the same time, a whole spectrum of digital services is intended for government organizations. The emphasized interoperability of government electronic services enables public and utility companies to improve their operations by making their business processes more efficient. Legacy systems, which are still widely used, can hinder effective business. In this paper, we will analyze different implementation alternatives for solving the mentioned problem in the three-decade-old legacy 4GL system. After that, we will describe the implementation of the wrapper which enables secure data exchange between the 4GL legacy information system of the largest utility company in Belgrade and the Ministry of Public Administration and Local Self-Government of the Republic of Serbia.

Key words: *Security, interoperability, G2G, legacy systems, legacy wrapper.*

Introduction

In the modern world, people rely on numerous digital services to fulfil their needs and perform business tasks. Likewise, companies that utilise digital services can enhance the efficiency of their business processes. Clearly defined, formalised and digitalised business processes enable efficient implementation of business tasks, easier transfer of informations and knowledge within the company, as well as simpler exchange of the same with other companies or government organizations.

In such an environment, where the emphasis is on achieving interoperability between different parties, security is a very important aspect. Cyberspace is an unsafe place in which there are malicious users who, knowing the way networks, network protocols and other IT functions work, constantly find ways to create new security threats [1]. The COVID-19 pandemic has caused people to stay at home, lockdowns, travel disruptions, social distancing and increased Internet use. Many branches of business have switched to work-from-home mode, while some of them, such as education, rely

[310] Slavimir Vesić, PhD, Honorary Academician of IRASA, Professor of vocational studies, Head of software development, PUC „Belgrade Waterworks and Sewerage“, Information Technology School – ITS, Belgrade, Republic of Serbia, vesic.slavimir@gmail.com, ORCID 0000-0003-2020-8418

on their Web platforms. During the pandemic and in the post-COVID period, there was a significant intensification of the activities of cyber attackers and various ATP groups. Individuals, businesses and government organisations are the target of their attacks.

Most government organisations, along with the utility sector, after the military and research institutions, have been pioneers in introducing ICT into their operations. In the past, the new IS supported the operations of the aforementioned organisations for decades. Those systems are legacy systems today. When a system loses its ability to evolve or significantly reduces it, it becomes a legacy system. It is most often made with old technologies, and the process of maintaining it is difficult and at the same time expensive for the company [2]. As it does not simply allow changes, it in some way limits the business, while, on the other hand, it contains valuable and decades-long formalised knowledge, business rules and procedures that the company relies on in its daily operations. At the same time, the legacy systems have high maintenance costs, and practice has also shown that the process of their modernisation is very risky and that there is a significant number of failed modernisation projects, even after significant investment in funds [3]. Legacy systems are directly opposed to modern tendencies to enable the secure exchange of information and knowledge between citizens and government organisations, as well as to intermingle government organisations with each other. In this context, they are a preventing factor for achieving safe interoperability, which is necessary in modern business and society.

In the first part, we will define the concept of legacy systems. Then we will look at a specific business problem from practice with an analysis of possible solutions. After that, we will describe the wrapping of the legacy system, together with the implementation solution on the example of PUC "Belgrade Waterworks and Sewerage".

Legacy systems

Information systems as models of real systems need to evolve. Therefore, software, as its inseparable part, must [4] : adapt to meet the needs of new technology or a new computing environment, upgrade to implement new business requirements, expand to be interoperable with other modern systems or databases, and architecturally reconstruct to maintain itself in an evolving computing environment. When the information system can no longer adapt to changes, it is withdrawn from use. Although it may seem relatively simple, as with the decommissioning of any other product, practice has shown that it is a very complex problem and is embodied in the form of legacy IS (legacy systems for short).

Legacy systems are systems developed in the past, most often with old or outdated technology, designed and implemented in accordance with the then methodological settings and development practices [5]. They are difficult to understand and resist changes, thus limiting business. At the same time, they are expensive to maintain and support key business processes, so the organisation would prefer to replace them, but this is very risky. Bearing this fact in mind, a large number of companies continue to maintain legacy systems even at a high cost. There is almost no area of business where this is not the case. Based on publicly available research by the GAO (Government Accountability Office), the US government planned to spend over $90 billion in 2019 on IT, where about 80% is spent on maintaining legacy systems [6]. The situation is similar in Serbia, where a large number of government agencies, public and utility companies struggle with the challenges that legacy systems pose to their operations. These systems are most often built from proprietary software, which results in a long-term commitment to a specific manufacturer that can dictate business conditions as well as opportunities in terms of expansion and technological training of the system to work with modern IT. Because of this, many organizations are in a very unenviable position when it comes to making long-term IT strategies.

Significant financial resources allocated for the maintenance of legacy systems, a large number of different definitions of legacy systems in theory, large investments in the modernisation of these systems and a large number of failed modernisation projects, despite the significant investment of funds in practice, indicate that [7]:

- The phenomenon of legacy information systems is highly complex
- Legacy systems are insufficiently or isolatedly understood in theory and practice
- That failed modernization projects are often the result of insufficient understanding of these systems and
- That the critical step towards successful modernization is their adequate understanding of the complex relationships between all their elements, as well as the factors of the complex and permanently changing environment

If you look at the scope of different legacy systems that appear in practice, it can be concluded that in addition to numerous problems of a technical nature, there is a whole range of non-technical reasons that have brought these systems to that state. Many of them are related to people who worked on the development of those systems and then left the organisation, as well as to organisational factors where the IT function embodied in the function of the director of the ICT sector is not recognised as very significant. In this paper, we will focus more on the technical reasons. In this context, they can be encountered in different forms, and in general, they can be built from legacy hardware, legacy software, legacy business procedures and rules, where they can have inadequate software architecture and a reduced possibility of integration [8]. They are one of the biggest challenges for IT professionals.

If the organisation decides to implement a certain modernisation procedure, it should bear in mind that it can be very challenging, because not every legacy system is the same, and the modernisation strategy cannot be universal. Badly chosen strategies, in addition to not being able to adequately replace all the functionality provided by the legacy system, can also be counterproductive, and even in some extreme situations, lead to shutdown. In the history of legacy systems, there are a large number of failed modernisation projects, where numerous organisations even after large investment projects, which are measured in billions of dollars, had to give up the same funds [9]. Therefore, it is very important to choose an adequate modernisation strategy.

The problem of the legacy system when implementing a secure G2G scenario

Legacy systems, like systems created in the past, very often have problems when using the latest technologies. This is especially pronounced when the manufacturer of the software in which the legacy system is implemented has not developed a specific technological solution that would make this possible. In addition, a situation may occur where the software manufacturer has provided a certain technology, but only from a subsequent version, and the organisation does not have sufficient capacity to carry out the process of transitioning the legacy system to that new version. A special problem arises if the new version is not backwards compatible with the previous one, so it is very difficult to make a smooth transition to the new version.

In this manuscript, we will look at the case of the PUC "Belgrade Waterworks and Sewerage", in the continuation of the text PUC BWS. It is a public utility company engaged in water production and wastewater disposal services in the territory of the city of Belgrade and is one of the largest utility companies in that city. The business information system PUC BWS is a system over 3 decades old and represents the main information solution of this organisation. It was created in the early 90s, and during that period, the information system was created in the programming language COBOL, and then in the early 2000s, the transition was made to Progress 4GL. In the meantime, the transition to OpenEdge 10.2B from 2009 has been made.

Progress 4GL is a 4th generation programming language and is owned by Progress Software Corporation, hereafter PSC. Progress 4GL belongs to the programming languages of the 4th generation, abbreviated 4GL, which were created in the early 80s with the idea that programming should include as many people as possible, even those who do not have a professional education in the field of ICT or computer science. The main motive is that this new generation of programming languages should respond to the problems of the 3rd generation of programming languages, where the entire development of the industry depends on professional programmers. The disadvantages of 4GL are reflected in [10]: application specificity, ownership, sensationalist thinking and poor design. Progress 4GL also inherits a number of these shortcomings, where ownership stands out. The long-term evolution of an information system depends on how well a system adapts to changes, which can occur on many different levels: business, technological, legal, and which is reflected in terms of positive regulations that regulate business, etc. If the software producer does not innovate sufficiently or does not enable a simple transition to a new version, the organisation may be in serious trouble, because it needs to continue working and its "digital model", i.e. the information system, cannot track it. Therefore, it is a legacy system and directly disrupts business.

In the research problem, a request was presented to the ICT sector of the PUC BWS, in which it is necessary to enable secure data exchange between the PUC BWS and the Central Population Register (CPR), which is operationalised by the Ministry of Public Administration and Local Self-Government in cooperation with the Office for IT and eGovernment. CPR is a unique and centralised electronic database on the population of the Republic of Serbia, which contains citizen data from various records maintained by various institutions [11]. In this interoperable scenario, it was necessary to enable the exchange of data from the legacy system PUC BWS and CPR, so that PUC BWS could implement the business process of updating its customers' data to improve the billing business processes.

CPR, as a platform that provides population data, bases its ICT solution on secure Web services. Web services are implemented as a REST Web API, using public key infrastructure standards. The legacy PUC BWS system implemented in the Progress OpenEdge 10.2B environment does not have the technological capabilities to use REST services. The main problem of the research is to enable secure data exchange of the three-decade-old legacy 4GL system through the selection of an adequate modernisation strategy in a situation where the organisation does not have the capacity to switch to a newer version of the implementation platform. In this context, the legacy system opposes the realisation of a secure G2G scenario of electronic business, and for a specific case, it is necessary to choose one of the alternatives.

Analysis of modernisation strategies and selection of alternatives

In theory and practice related to legacy systems, there are a large number of different ways of modernisation, but it can be said that 3 modernisation strategies stand out: reengineering, wrapping and migration. In the following, we will briefly present each of them.

In the theory of legacy systems, software evolution, and software maintenance, there are many different views on what is meant by software reengineering. The reason for such a thing is that certain terms have changed over time with the progress of ICT, as well as the observations of researchers, and at the same time there is a certain overlap between the related terms of evolution and software maintenance. It can be said that in the process of reengineering, 3 interrelated phases are distinguished: reverse engineering, transformation and forward engineering. Reverse engineering starts with the existing system, intending to understand it, to observe it comprehensively and as it is ("as is") and to gather knowledge about the structure and behaviour of that system based on that. Forward engineering is classic software engineering that aims to create a new system as a product of the user's desire to get the system they want ("to be"). Now that we know the starting point and the destination, the elements of one system are transformed into the elements of another system.

In a situation where we want to bypass an invasive method in solving the problems of legacy systems, such as reengineering a system that is risky to replace or transforming the entire system, we can usewrapping technique. In this way, it is possible for a legacy application or part of a legacy system to function with other applications and execute common business processes [12]. Wrapping enables the gradual replacement of the legacy system or one of its modules, applications, procedures, etc. That way, the legacy application can be used like any other component. According to the authors [13], there are several types of wrappers: database wrappers, system service wrappers, application wrappers, and function wrappers. Wrapping is a non-invasive way of solving the problems of legacy systems with the aim of integrating the legacy systems into a wider whole. It is often suggested

in theory and practice that wrapping be used as an alternative in a situation where it is necessary to fit all or part of the legacy system into a wider service-oriented architecture.
Migration is a term that has been used for a long time in information technology and can mean many things: data migration, database schema migration, software migration, content migration, virtual machine migration, etc. Each of these terms implies the transfer of certain entities into a new form and the continuation of their work. When we talk about legacy systems, we usually mean software migration or data migration. In relation to whether the migration is carried out in its entirety or only a part, we distinguish between complete migration, incremental migration and partial migration [14]. They differ according to the level of activities that need to be carried out, but also the risk that the new system or part of the system will take over the functionality of the old one that is being retired.
Bearing in mind that it is necessary to enable a safe G2G scenario, a comparison of 3 different modernisation strategies of the concrete three-decade-old legacy PUC BWS system was made, according to several criteria: risk, time, required resources and efficiency.

Table 2. Comparison of different modernization strategies

Strategy	Risk	Time	Resources needed	Effectiveness
Reengineering	high	long	great	complete
Wrapping	low	short	small	partial
Migration	medium	long	great	complete

Based on the conducted comparison, the wrapping strategy seems to be the best alternative, but its effectiveness is not complete. On the other hand, reengineering needs to be carried out on the entire system, so that the entire system is transformed, where the effectiveness is complete, but the time is long, the resources are large, and the risk is high. In addition, migration is also effective, but with less risk compared to reengineering.
Given that none of the classic modernisation strategies offers a satisfactory solution, a mixed strategy was chosen that combines wrapping and partial migration and thereby overcomes the shortcomings of both strategies individually. In the continuation of the manuscript, we will describe the solution.

Description of the solution

As previously mentioned, the legacy PUC BWS system does not have the technological capabilities to use the REST Web API to exchange data with the CPR platform and thus realise the G2G scenario of electronic business. It is possible to achieve this with a mixed strategy, combining the wrapping and partial migration strategies. Each of these two strategies allows us to overcome the mentioned problem. The legacy system uses the Progress DB database on OpenEdge 10.2B, which enables the storage of data on various business entities: customers, suppliers, their accounts, payments, debts, etc. When working with applications of the legacy system, the database uses the 4GL connection, which is proprietary and enables communication

exclusively with applications written in the Progress 4GL programming language. In other words, it is a closed system. So, the first step that needs to be done is to enable other programming languages or other environments to access the same database. This is precisely the task of the wrapper, which enables the use of the Progress DB database through an ODBC connection. That wrapper is realised in the form of an adapter.
Since Progress 4GL in version 10.2B does not have the ability to work with REST Web services, it was necessary to enable it in another programming environment. Since the legacy system already has an application for updating business partner data, part of its functionality has been migrated from Progress 4GL to C# so that it continues to live as a new application in .NET. In that application, a new functionality was created that enables data exchange with the CPR platform using the REST Web API. Based on this, it is possible to exchange data on persons who are customers of PUC BWS products and services with the CPR platform. The exchange of data is carried out in a secure manner, regulated by PKCS standards prescribed by the Ministry of Public Administration and Local Self-Government. In this way, secure data exchange is achieved, where the Ministry, together with the Office for IT and eGovernment, regulates which government organisations, administrative bodies, public and utility companies can use CPR. After receiving the data from the CPR, the competent services of the PUC BWS continue their work and can decide how to proceed with the customer's debt. In this way, a safe G2G scenario is enabled, which contributes to the business of PUC BWS.

Conclusion

Legacy systems resist changes due to their reduced evolutionary capacity. In this way, they limit business, because it can be more efficient by applying new business models. The paper describes a mixed approach in the modernization of the legacy system, which allows to overcome technological barriers and realising the G2G scenario of electronic business in a safe way. The use of wrappers and migration enables the interoperability of the legacy, three-decade-old system with the Ministry's platform. The descriptive solution is applicable for a class of legacy systems that are built in 4th-generation programming languages, and there is a need to realise a secure interoperable scenario in a short time with limited resources. It should be emphasised that the modernisation of the described legacy system is partial and that for its complete modernisation, it is necessary to apply a different strategy, where the described mixed strategy can be part of a broader strategy of overall modernisation.

References

[1] Vesić, S. L. (2023a). Impact of COVID-19 on Cybersecurity. In E. Stojić-Karanović & K. Ristić (Eds.), *SIGNIFICANCE and treatment of current world trends* (pp. 115–131). International scientific forum "Danube - river of cooperation."
[2] Vesić, S. L. (2022b). Towards smart city: Modernization of Legacy Systems of the City Waterworks on the Example of PUC Belgrade Waterworks and Sewerage. *IRASA - Fourth*

International Scientific Conference - Science, education, technology and innovation - SETI IV, 439–448.
[3] Vesić, S. L. (2023b). Nasleđeni sistemi, nasleđen softver, modernizacija, razvoj i testiranje postojećih i novih delova nasleđenog softverskog sistema i baza podataka. In V. Kuleto (Ed.), *Informacione tehnologije : odabrana poglavlja / grupa autora* (pp. 15–41). Visoka škola strukovnih studija za informacione tehnologije - ITS.
[4] Pressman, R., & Maxim, B. (2015). *Software Engineering: A Practitioner's Approach* (8th ed.). McGraw-Hill Education.
[5] Vesić, S. L. (2020). Uticaj razvoja IT tehnologija na prevazilaženje problema nasleđenih sisitema. *Kultura Polisa*, *XVII*(42), 667–678.
[6] United States Government Accountability Office. (2019). *GAO-19-471 - INFORMATION TECHNOLOGY - Agencies Need to Develop Modernization Plans for Critical Legacy Systems*. https://www.gao.gov/assets/700/699616.pdf
[7] Vesić, S. L., & Laković, D. (2023). A framework for evaluating legacy systems - a case study. *Culture of Polis*, *20*(1), 32–50. https://doi.org/10.51738/Kpolisa2023.20.1r.32vl
[8] Vesić, S. L. (2022a). Modernizacija nasleđenog sistema u funkciji naplate na primeru portala e.Sandučе. In L. Papić (Ed.), *DQM-POLITEH-2022, 25. međunarodna DQM konferencija upravljanje kvalitetom i pouzdanošću* (pp. 384–388). Istraživački centar DQM.
[9] Ulrich, W. M., & Newcomb, P. H. (2010). Information Systems Transformation: Architecture-Driven Modernization Case Studies. In *Information Systems Transformation*. Morgan Kaufmann OMG Press Titles. https://doi.org/10.1016/C2009-0-19987-7
[10] Grubb, P., & Takang, A. (2003). *Software Maintenance: Concepts and Practice* (2nd ed.). World Scientific Publishing Co. Pte. Ltd.
[11] Ministry of Public Administration and Local Self-Government. (2024). *Central Population Register*. https://mduls.gov.rs/en/about-ministry/sectors/the-sector-for-civil-records-and-registers/central-population-register/
[12] Willem-Jan, van den H. (2009). *Aligning Modern Business Processes and Legacy Systems: A Component-Based Perspective*. The MIT Press. https://doi.org/978-0262513463
[13] Orfali, R., Harkey, D., & Edwards, J. (1995). *The Essential Distributed Objects Survival Guide*. Wiley.
[14] Althani, B., & Khaddaj, S. (2018). Systematic Review of Legacy System Migration. *Proceedings - 2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, DCABES 2017, 2018-Septe*, 154–157. https://doi.org/10.1109/DCABES.2017.41