Reference Business Processes-Based Method for Multi-Tenant SaaS Architecture Deployment and Adaptation

Artem Levchenko
Voronezh State University
Voronezh, Russian Federation
artem.levchenko@sap.com

Victor Taratukhin
Higher School of Economics, ERCIS
Moscow, Russian Federation
vtaratoukhine@hse.ru

***Abstract*—The implementation of multi-tenant SaaS architecture is still new for the market and causes Business and IT misalignment and, as a result, lower business value realization and project complexity. Reference business processes' role significantly changes in SaaS architecture because of its lesser system functionality flexibility. It has become essential not only for fit-gap workshops during deployment but also for constant SaaS architecture adaptation. The functional requirements realization models and methods that were designed for on-premise cannot be applied in the case of SaaS.**

This article introduces the functional requirements realization model for implementation and adaptation of multi-tenant SaaS architecture based on reference business process models. System analysis and general control theory techniques have been applied to formalize the problem. The decision support method, based on reference business processes, has been developed to maximize the business value of SaaS implementation. Value management has been applied to justify the economic feasibility of the applied method. The suggested method has been validated based on SAP platform-based SaaS solution implementation projects for procurement business process management in the Russian Federation.

I. Introduction

Value creation is a fundamental element of each business model [1]. Software as a Service (SaaS) is one of the three main business models of cloud computing, providing mature and complete software components and functionalities to companies [2]. In the SaaS delivery model, business applications are offered as Internet services by a SaaS provider and remotely consumed by many different customer organizations (called tenants). Unlike on-premise solutions, enterprises can start using SaaS solutions with minimal implementation time since they merely require a subscription [3]. The economic viability and cost-effectiveness of a SaaS offering, from the SaaS provider's perspective, depends strongly on two principles: maximal automation of its operation, and self-service (allowing tenant organizations themselves to customize and configure different aspects of the service to their specific needs) [4]. This software delivery model has various benefits for companies, including improved cost, ease of use, and maintenance capabilities, and reduced timescales. The most significant advantage of SaaS is that it creates value through financial savings for enterprises by offering affordable subscription services [5]. Whereas its benefits have been delineated frequently in the literature, less than ten years ago, no model had yet been attempted to show how such applications could leverage business value in practice [6].

However, many Russian companies are still waiting to see how attitudes to SaaS implementation develop, despite their now clearly demonstrated value [7]. The implementation of multi-tenant SaaS architecture is still new for the market and causes Business and IT misalignment and, as a result, lower business value realization and project complexity. Russian companies lack managerial experience and competency in using ubiquitous technologies and SaaS applications to create business value [8]. Because of this, adoption of SaaS often fails to deliver promised additional business value. One of the most important reasons is that after its adoption, IT departments may fail to reengineer business processes for the updated realities created by SaaS implementation [9]. Meanwhile, customization requirements can range from adding new functions on top of the native functions to creating a considerably modified version of the existing native functions of the same software shared by other clients [10].

Enormous challenges are presented to companies when sudden economic crises arise, and fundamental business models have to be changed radically. For example, the economic crisis in 2020 has led to the need to apply remote work methods for employees. Top management changed the requirements for enterprise systems to enable rapid business process adaptation via simplified system configurations. Business owners began optimizing business maintenance costs. Because of the rapid and straightforward installation and quick and cost-effective upgrades characterizing SaaS [8], its popularity is now growing.

The core of any enterprise system's quick deployment is an accurate description of the standard embedded business model. Accelerators like References Business Processes (RBP) play a crucial role in speeding up on-premise solution deployment (RBP may also be known as Best Practices or Typical Business Processes models). RBPs are used during the conceptual design phase of the implementation to describe the target business processes more quickly. However, RBP's role significantly changes in SaaS architecture. In the case of on-premise solutions, the design of business processes is required first, followed by an IT system that allows you to automate the designed business process [11]. But SaaS is changing this paradigm: enterprise systems begin with embedded functionality to which companies' business processes may then have to be aligned during and following SaaS implementation. For these reasons, the Functional Requirements (FR) realization models and methods designed for on-premise cannot be applied in SaaS. SaaS has less system functionality flexibility and thus demands a different deployment methodology.

RBP has become essential for fit-gap workshops during deployment, and for ongoing SaaS architecture adaptation following routine updates. When performing upgrades in a long-established system at run time, mechanisms are required to work on a per-tenant basis [12]. Adaption of the SaaS model requires significant effort and expertise from software companies to establish or transform their business modeling and re-engineer their business processes [13]. Business process improvement is necessary for organizations to gain a competitive advantage and survive [14].

The current investigation area relates to business value maximization by improving alignment between Business and IT departments. This scientific research aims to develop a decision support method for FR realizations during the deployment and adaptation of multi-tenant SaaS architecture. The design should aim at business value maximization and be based on RBP. The following research steps, according to system analysis theory, were used:

1) Increase the line spacing definition of system elements for the FR realization model during the implementation and adaptation of multi-tenant SaaS architecture,

2) description of system elements' relationship in typical on-premise and SaaS cases,

3) development of decision support method based on RBP for business value maximization,

4) method validation.

The remainder of this paper is structured as follows: Section 2 presents the relevant scientific articles in the areas of FR realization, estimation of business value of cloud computing, multi-tenant SaaS architecture-specific highlighting, RBP design, business and IT alignment. The science knowledge gap is highlighted. Then, Section 3 presents and discusses the definition of system elements for the FR realization model during implementation and adaptation of multi-tenant SaaS architecture, extended with a description of system elements' relationship in typical on-premise and SaaS cases in Section 4. Section 5 presents the decision support method, based on RBP for business value maximization. Section 6 discusses method validation and efficiency results. Finally, Section 7 concludes the paper and defines the future work direction.

II. Related works

The related works were analyzed in the following areas: the business value of the cloud computing estimation methods, business and IT alignment models, multi-tenant SaaS architecture deployment and adaptation methods, reference business process-based methods, decision support method for FR realizations in SaaS. The analysis below follows a similar sequence of topics.

The challenges of alignment between business and IT strategy in enterprises to create business value using cloud SaaS applications are described in [8]. The positive and negative accounts of aligning IT strategy (application of SaaS software tools) with business strategy (meeting service demands, accessibility needs, storage, and resource needs, application customizability), and cost-effectiveness are listed. The research was done by a holistic case study of two enterprises using SaaS, provides good examples of SaaS experience, but does not consist of recommendations. The [9] paper examined the factors that influence the alignment of cloud computing and enterprise. Based on the dynamic capability theory, the authors proposed a research model that studies how dynamic enterprise capability, organization flexibility, and IT flexibility affect cloud computing's alignment within enterprises in an ever-changing environment. Service innovation in the cloud, described in [15], supports findings from the service science literature that service transformation is a process that changes how and where value is created. The analytical background was considered, but no practical models were provided to use in current research. The [16] article resolved quantification of financial value from cloud computing investments by the analytical framework also based on scientific articles. The outputs were taken to confirm the findings of the current research. Besides the analytical searches, the [17] article describes the empirical research of cloud computing business value. The conceptual basis is relevant for the current research.

Business and IT alignment aspects and conceptual model of business and IT alignment for SaaS solutions was introduced to connect business needs, IT needs, and value creation [8]. It was a base for this work with some changes, such as "needs" specified as "functional requirements." The existing model is scaled up and extended in customizability and rapid elasticity areas. Unlike [8], we consider value generated by usage or RBP and concentrate more on FR, less on subscription cost, security, and other topics. It also highlights that mapping between business processes and software processes for SaaS [18] was introduced by using the Business Process Execution Language. The SaaS application layer's customization model is suggested, but it does not solve the decision issue about realizing the business requirements based on RBP. Furthermore, it is interesting that the procurement processes have the most risks in applying cloud computing technology for logistics activities [24]. In our article, we also decided to analyze the procurement area.

The authors of the [13] article could not find any review that systematically discusses the adoption of the SaaS model by software companies. That shows the knowledge gap in this area. Two important adaptation factors were discussed: facilitating upgrading, modification, and customization processes and obtaining agility and scalability of the development and deployment. RBP is relevant for both these adaptation factors. The SaaS adaptation process covers support needs for the continuous evolution of multi-tenant SaaS applications. In the [4] article, the authors introduced self-adaptive middleware for dynamic upgrade enactment with support for tenant mediation.

These basic concepts were used for our research. The [10] paper has outlined a generic process model for the customization of SaaS software. The basis is that the client defines its customization requirements in terms of software functionality. This is crucial for SaaS deployment and adaptation. The process from this article was actively used and enhanced to meet our research goals. Similar tasks were resolved in the [19] paper with a unified business-driven cloud management framework, enabling optimization of business metrics without limiting business to a specific cloud provider. It describes the approach towards reducing complexity in IT system management, where IT systems self-manage using policies. A detailed and useful taxonomy was provided in this article. One more useful taxonomy, the data meta-model, was suggested in [2]. The authors made a selection of the most suitable business processes for outsourcing; the most appropriate SaaS composition to adopt requires matching the functionalities of both of them. This article is about preparation for the project, not about implementation and adaptation phases.

The [20] article presents an integrated framework for continuously evaluating, managing, and evolving dynamics between heterogeneous SaaS applications and business processes. The primary focus is on the functionalities of heterogeneous SaaS applications and their integrity with business process automation. The authors identified an integrated framework to correlate SaaS functions and business operations. The article solves quite a similar task – the differences is that it does not provide a decision support method and does not consider RBP. The authors of [14] described the process model repository impact for requirement realizations and business value calculation. Many organizations develop process model repositories, which extensively capture organizational knowledge. Our research aims at relatively similar goals but with a different approach, targeting more decision-making at the deployment and adaptation phases. This research is one of the most relevant for us. The [22] article discusses implementing DevOps practices in highly regulated environments. This article also highlighted the lack of a centralized document repository as a critical challenge for cloud deployment. RBP aims to partially solve this issue. A similar problem of business process documentation was highlighted for the case of business processes outsourcing [23].

In [21], the authors proposed a scalable system architecture conceptual design for deploying a business process management system in a multi-tenanted cloud environment. It also highlighted the challenges of transitioning RBP from on-premise solutions to SaaS. However, it considers Business Process Management Systems, not the core business processes management, such as procurement, so the architecture is not designed to work with FR as an input. The highlighted multi-tenancy specifics were considered for our research. The [6] article focuses on allowing rapid deployment at low IT cost and highlighting the value creation classification. According to this classification, our article will focus on increased efficiency due to better information symmetries regarding this classification.

As a result of the analysis of related work, we would like to point out the research gap in the area of RBP-based decision-making for FR realization during the deployment and adaptation of multi-tenant SaaS architecture. No articles exist about how RBP usage influences the maximization of business value. Some mentioned models and methods from related areas were used as a basis for this research. The current article's science value lies in its new model, based on system analysis and control theory, and in the method, it establishes for decision-making during SaaS deployment and implementation.

IV. FR REALIZATION MODEL DEFINITION

The classical understanding of an enterprise systems implementation project is to ensure that the customer's FR are realized in the system and that the project result is a program that realizes those requirements. The requirements implementation is necessary to get economic benefits and profit, called business value. The business value from the enterprise system implementation and adaptation can be presented as the following function:

$$V = f_V(r),$$

where r is a set of FR, V is a business value generated by FR realization. The system deployment is linked with the implementation and maintenance costs. The more FR, the more complex the project and the higher costs. The cost of implementation of the enterprise system can be described with the following function:

$$C = f_C(r),$$

where C is a cost for FR realization, including FR formalization, realization, maintenance cost, and associated others. The economic viability of enterprise system deployment is achieved if the economic benefits from the implementation of the FR exceed the cost of the implementation and maintenance, that is, the sum of business value $f_V(r)$ is greater than sum of associated costs $f_C(r)$ of FR for all r in project scope.

In practice the calculation has various challenges, for example due to qualitative value existence, such as NPS increase as a result of UI improvement. In the model we tried to quantify the qualitative benefits where possible. For example, UI improvements can increase operational excellence and reduce transaction execution time. The second challenge is time-dependence for the business values and costs – calculation should include one-time costs, for example, a non-standard RF programming, as well regular costs of development maintenance. To consider all types of costs associated with FR realization the model considers a 3-year time period, based on an average SaaS contract duration. Quite important to mention is that the model does not include SaaS subscription or on-premise software cost, because the aim of the research is to consider decision-making for FR realization, not to make a decision about enterprise system implementation. This can be a topic for further research.

In case of a new enterprise system being developed from scratch, function $f_C(r)$ will represent the expenses for a project team, external developers, partly involved business users, support users and other project team members. Minimization of the $f_C(r)$ function can be achieved by purchasing a developed enterprise system instead of developing one. The use of widely used enterprise systems reduces the cost of designing the system functionality. Compared with systems developed during the project, the software products are set up based on FR that have been included in the system design (that is, which are economically efficient) – usually, additional program code can be created for requirements that were not originally included in the original system design. The use of developed software products in case of a huge number of FR reduces the project's timeline and cost. However, the use of out-of-the-box function realization can impose constraints on the implementation of all FR, if, for example, they are contrary to the core system logic. Thus, the benefit and cost functions can be broken down into cost for the enterprise system standard functionality implementation, and the cost function of implementing the out-of-the-box functionality, such as, the development of new system functions. In case of non-standard functionality deployment this causes additional costs $f_e(r)$ for development maintenance. The e costs also influence the further FR realization and increase the realization complexity and influence on $f_C(r)$. Put simply,

$$f_C(r+1) = f_e(r) + f_C(r),$$

where $f_C(r+1)$ is a costs of FR realization after the non-standard function's realization. The final goal of enterprise system implementation and adaptation is to maximize the sum of $f_V(r)$ and minimize the sum of $f_C(r)$. The contradiction is $f_e(r)$ has nonlinear influence on both functions $f_V(r)$ and $f_C(r)$. The task is related to decision-making over $f_e(r)$ deployment for each FR iteration.

IV. The model of on-premise and SaaS deployment and adaptation

The conversion from FR gathering to business value from FR realization can be modeled using the control theory approach. The classical on-premise control system model for the system deployment is represented in Fig. 1.

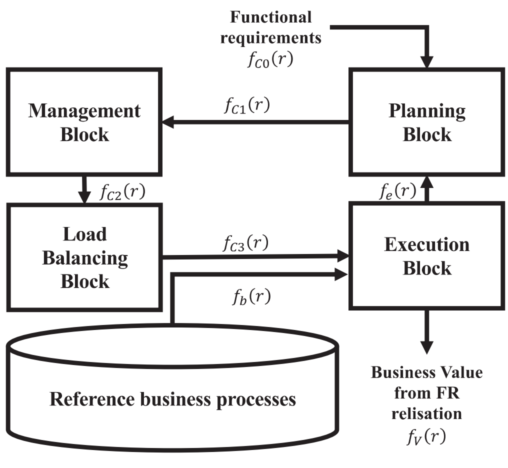

Fig. 1 Control system model for on-premise system deployment

The control system model includes four blocks:

Planning Block. The purpose of this block is to formalize the FR for the enterprise system considering external environment regulations, such as the requirements of related departments, company shareholder's requirements, market requirements, legal requirements. Typically, the implementation projects in this block are represented by the methodology team and the functional department's managers. The added block's cost for the FR procession is calculated as subtraction $f_{C1}(r)$ from $f_{C0}(r)$.

Management Block. This block's responsibility is to align and approve the FR based on the strategy and project charter. In terms of control theory, this block represents a decision-maker. The added block's cost for the FR procession is calculated as subtraction $f_{C2}(r)$ from $f_{C1}(r)$.

Load Balancing Block. The purpose of this block is to allocate load on the Execution Block based on the priority of FR, their dependence on each other, and the current load of the Execution Block. On the project, this block is represented by the project management team. The added block's cost for the FR procession is calculated as subtraction $f_{C3}(r)$ from $f_{C2}(r)$.

Execution Block. The purpose of this block is to build the programmatic implementation of the FR. The added block's cost for the FR procession is calculated as subtraction ($f_{C3}(r)$+ $f_b(r)$) from $f_V(r)$. The $f_b(r)$ represents improving the realization of the requirements by RBP applying. The FR covered by the standard system functionality can be realized using RBP, which includes process descriptions, enterprise system configuration descriptions, roles descriptions, test case descriptions, and other documents. One more purpose of this block is to classify the FR into standard and non-standard requirements based on the system in-box capabilities, creating a feedback loop to the Planning Block. Non-standard requirements cause developments; developments cause system complexity to increase. The $f_{C2}(r)$ function represents the influence of systems enchantments to the next iteration from the Planning Block run.

Implemented enterprise system requires support and upgrades due to regularly changing requirements from the external environment, such as legal changes. As a result, the control system model looks differently compare with the implementation stage (Fig. 2).

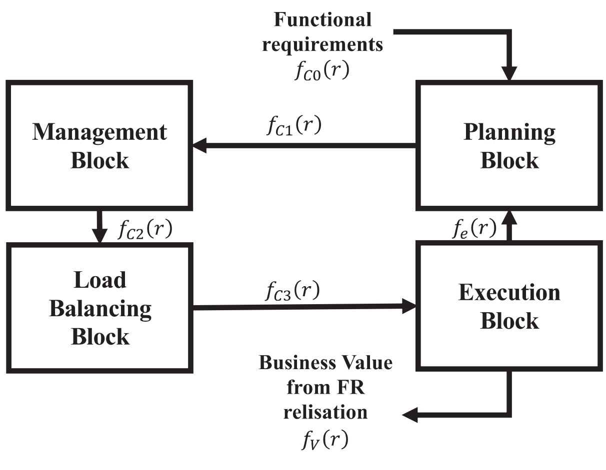

Fig. 2 Control system model for on-premise system adaptation

In case of support of the Enterprise System, Load Balancing Block and Execution Blocks are often represented by the in-house IT department. RBP models are missing because enterprise system support is usually performed without having access to the RBP base. Almost all enterprise systems support methodologies do not use RBP as a source for system enhancements.

The evolution of enterprise systems allowed applying the SaaS model, implementation and support approaches were radically changed. The relevant control system model is presented in Fig. 3.

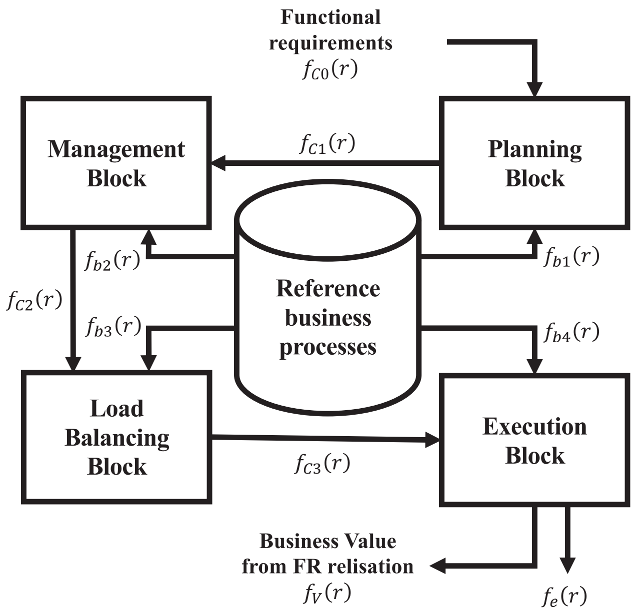

Fig. 3 Control system model for SaaS deployment and adaptation

The use of the micro-service architecture has allowed dividing the support for standard and non-standard FR. The usage of bimodal architecture allows isolating the influence of $f_e(r)$ function to the Planning Block. It means that adding the new development to the system will not make the SaaS functionality more complicated. The use of cloud technology has enabled the system's support by using the RBP, when the external environment changes can be embedded to the models by vendor centrally, rather than by in-house IT department. The lack of feedback between the Execution Bloch and the Planning Block will reduce the number of communication's iterations and reduce the $f_{C1}(r)$ function's input.

The data flow across the model's blocks within the implementation project is provided by the core system's elements – elementary work with defined duration and thus cost. In this way, and $f_{Cx}(r)$ and $f_{bx}(r)$ functions can be calculated. For the calculation, the process model's stratum, which has the most detail, was used. It consists of process steps executed by each business user. Formalization of the FR was carried out using fuzzy logic for the classical control model case and standard logic for of SaaS. SaaS enables formalizing standard FR, the implement is as separate system functions and combining them during the project. Blueprint phase has been replaced by a fit-gap analysis phase for SaaS during implementation, which significantly reduces the $f_{Cx}(r)$ function costs.

The control system model research for the on-premise enterprise system and control system model for the SaaS has resulted in the following conclusions. Both systems belong to a class of open systems, purposeful systems, large and well-organized. A significant difference is that the on-premise control system model does not have the properties of their organization, that is, the ability to grow by itself, adapting to change. The complexity of the on-premise control system model is increasing over time, while the complexity of the SaaS model's evolution remains constant. This feature may be due to the "needed diversity" law that the diversity of the management system should be greater than or equal to the managed object's diversity. Many additional developments always have less diversity than in multiple business processes at an enterprise, which in turn less than the diversity of RBP models designed to match the diversity of businesses for various industries and various business models.

V. Decision support method for FR realization in SaaS

The decision between custom development realization and following the RBP in case of FR is not standard has a significant influence on the SaaS value realization. Fig. 4 presents the decision support method for FR realization.

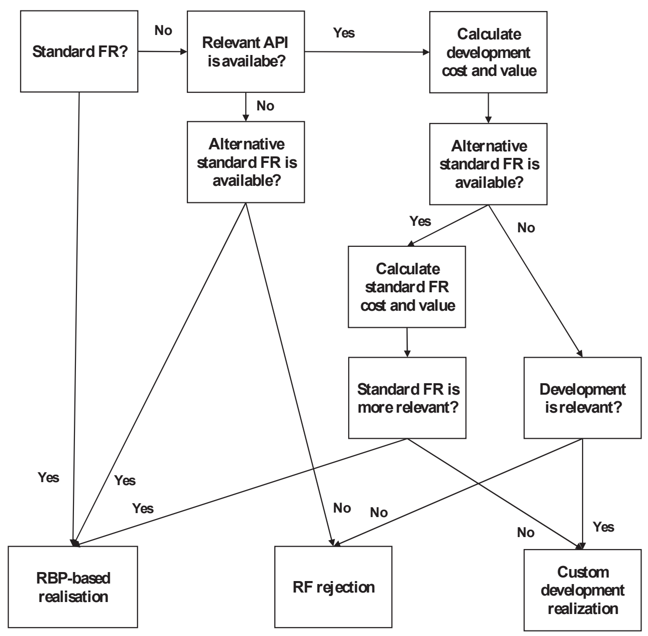

Fig. 4 Decision support method for FR realization in SaaS

If the standard system functionality cannot cover FR and relevant Application Programming Interface (API) is not available for the system enchantment, then alternative standard FR can be considered. The decision is depending on the volume of change management that is required to adopt the FR.

If relevant API is available, then potentially, the custom development can be realized to cover the FR. The formula

$$CDV = f_V(r) - f_{C0}(r) - f_{C1}(r) - f_{C2}(r) - f_{C3}(r) - f_e(r)$$

is used to calculate the custom development value for the organization. The $f_e(r)$ includes maintenance cost for the development. The next step represents analyzing the alternative FR that can be covered by standard system functionality. If that exists, then the standard alternative FR cost and value are calculated by the following formula:

$$FRV = f_V(r) + f_{b1}(r) + f_{b2}(r) + f_{b3}(r) + f_{b4}(r) - f_{C0}(r) - f_{C1}(r) - f_{C2}(r) - f_{C3}(r)$$

The next stage is a comparison of *CDV* and *FRV* values. In case if *FRV* is greater or equal to *CDV*, then development realization is declined, and RBP-based alternative FR realization is used.

If alternative RF does not exist and custom development can be realized, then *CDV* should be evaluated to decide the development relevancy.

IV. Results, Limitations and Discussion

The suggested method's validation was done based on implementation projects of SAP platform-based SaaS solution for procurement business processes management in the Russian Federation. To calculate $f_V(r)$, we used value management (not activity-based costing) and estimated the business value by process steps. Subject experts give the estimation for the variables in the model for the eighteen non-standard FR. It is worth noting that the received results were used to prove the final decisions during the SaaS system deployment project. Table I presents the values of variables for selected five non-standard FR and the decision-making method results. The currency is excluded from the research. The five non-standard FR are listed below:

- Purchase requisition budget checking during its creation
- Supplier data verification in the external system during qualification procedure
- External reporting tools used for the purchase requisition budget collection
- Purchasing contract expert to the external system for achieving
- Purchase requisition approval via an external system

TABLE I. The values of variables for selected FR in decision making method

FR #	**1**	**2**	**3**	**4**	**5**
C0(r)	3	4	1	1	3
C1(r)	2	3	2	3	3
C2(r)	3	1	1	2	3
C3(r)	2	3	1	2	5
b1(r)	1	3	1	1	2
b2(r)	1	1	1	1	1
b3(r)	2	1	1	1	2
b4(r)	1	1	1	1	1
V(r)	30	21	69	17	22
C0'(r)	3	4	1	1	3
C1'(r)	2	3	2	3	3
C2'(r)	4	1	1	4	3
C3'(r)	23	10	3	9	5
e(r)	9	1	10	9	7
V'(r)	89	30	64	22	73
Development realization	yes	no	no	no	yes
Alternative FR realization	no	yes	yes	yes	no

The research limitations are the following: the research has been done with the usage of the procurement business processes, the study based on the results of completed implementation projects performed by SAP-platform based SaaS. We assume that similar results should be achieved for the other dataset. In this study, all functions were estimated by the subject experts. As a next step, we are considering developing the methodology for the estimation of the automatic values based on previous project experience. The assumption was made for the change management for the alternative FR realization is minimal. Current research does not cover the estimation of the solution error and assessment of its effectiveness. It is considering as a next step. The reelevates of the research results have been confirmed by practice. Despite the mentioned limitations, the study's aim was achieved, and the next steps are defined to fulfill the limitation-based gaps.

IV. Conclusion and Future Work

Business value realization from multi-tenant SaaS architecture deployment and adaptation is today a high demand topic. It is necessary for profitable business and IT alignment. The role of RBP is significantly changing towards cloud services specifics. It has become essential for fit-gap workshops and the realization phase during deployment, as well as constant SaaS architecture adaptation. The RBP-based models and methods that have been designed for on-premise use cannot be used for SaaS.

This article describes the FR realization model during the implementation and adaptation of multi-tenant SaaS architecture based on reference business processes models. System analysis and general control theory techniques have been applied to formalize RBP usage for SaaS and its influence on the business value creation chain. The decision support method is based on RBP, making a quantitative analysis for the case when SaaS standard functionality cannot cover FR. The following research steps, according to system analysis theory, were used: definition of system elements, description of system elements' relationship, creation of the decision support method, method validation. Value management was applied to justify the feasibility of the applied method. The suggested method has been validated based on SAP platform-based SaaS solution implementation projects for procurement business process management in the Russian Federation.

In the control system models for SaaS implementation and adoption, four blocks were defined and described: the Planning Block, the Management Block, the Load Balancing Block, and the Execution Block. The block connections were described to calculate the transfer function during FR realization. It allows us to define the relationship between business value realization and realization and maintenance costs of the FR. Use of system analysis methods has enabled the creation of a useful control system model to address similar challenges in future projects with SaaS technology.

This article highlights that the complexity of on-premise control systems increases over time, while the complexity of the SaaS control systems remains constant. This is one of the factors that causes lower-than-expected business value realization and IT-business misalignment. RBP optimization has significant potential not only for SaaS deployment but also for adaptation. This will be the focus of the next stage of our research.

REFERENCES

[1] M. N. Nxumalo, I. N. Mba, and M. O. Adigun, “The value perspective to deploying a cloudlet business model for SMEs,” 2018 Int. Conf. Intell. Innov. Comput. Appl. ICONIC 2018, pp. 1–6, 2019.

[2] N. Fattouch, M. Rekik, A. A. Wakrime, and K. Boukadi, “A model-driven engineering approach for business process based saas services composition,” Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA, vol. 2019-Novem, 2019.

[3] W.-T. Tsai, P. Zhong, and Y. Chen, “Tenant-centric Sub-Tenancy Architecture in Software-as-a-Service,” CAAI Trans. Intell. Technol., vol. 1, no. 2, pp. 150–161, 2016.

[4] F. Gey, D. Van Landuyt, and W. Joosen, “Evolving multi-tenant SaaS applications through self-adaptive upgrade enactment and tenant mediation,” Proc. - 11th Int. Symp. Softw. Eng. Adapt. Self-Managing Syst. SEAMS 2016, pp. 151–157, 2016.

[5] E. Loukis, M. Janssen, and I. Mintchev, “Determinants of software-as-a-service benefits and impact on firm performance,” Decis. Support Syst., vol. 117, no. December 2018, pp. 38–47, 2019.

[6] A. Mladenow, E. Fuchs, P. Dohmen, and C. Strauss, “Value creation using clouds: Analysis of value drivers for start-ups and small and medium sized enterprises in the textile industry,” Proc. – 26th IEEE Int. Conf. Adv. Inf. Netw. Appl. Work. WAINA 2012, pp. 1215–1220, 2012.

[7] A. Bataev, “Performance Evaluation of Cloud Services for Russian Companies,” Conf. Open Innov. Assoc. Fruct, vol. 2020-April, pp. 46–51, 2020.

[8] M. B. Ali, T. Wood-harper, and R. Ramlogan, “Challenges of value creation through cloud SaaS: Business / IT alignment in service-oriented industries,” Proc. – Conference: 25th UK Academy for Information Systems International, 2020.

[9] M. SHAO and X. LI, “An Empirical Study of Impact Factors on the Alignment of Cloud Computing and Enterprise,” pp. 70–74, 2020.

[10] K. M. Khan, A. Nhlabatsi, and N. Khan, “A Process Model for Customisation of Software in Multi-tenant SaaS Model,” Proc. - 2015 IEEE/ACM 8th Int. Conf. Util. Cloud Comput. UCC 2015, pp. 418–419, 2015.

[11] K. Abu and U. Raissa, “Designing business processes based on an integrated approach: Process, model and architectural,” ACM Int. Conf. Proceeding Ser., 2020.

[12] F. Gey, D. Van Landuyt, and W. Joosen, “Continuous Evolution of Multi-tenant SaaS Applications: A Customizable Dynamic Adaptation Approach,” no. May, 2015.

[13] A. Saltan and A. Seffah, “Engineering and business aspects of SaaS model adoption: Insights from a mapping study,” CEUR Workshop Proc., vol. 2305, pp. 115–127, 2018.

[14] D. Iren and H. A. Reijers, “Leveraging business process improvement with natural language processing and organizational semantic knowledge,” ACM Int. Conf. Proceeding Ser., vol. Part F1287, pp. 100–108, 2017.

[15] R. R. Harmon and E. G. Castro-Leon, “Service innovation in the cloud: Implications for strategy development,” PICMET 2018 - Portl. Int. Conf. Manag. Eng. Technol. Manag. Technol. Entrep. Engine Econ. Growth, Proc., 2018.

[16] P. Rosati, G. Fox, D. Kenny, and T. Lynn, “Quantifying the Financial Value of Cloud Investments: A Systematic Literature Review,” Proc. Int. Conf. Cloud Comput. Technol. Sci. CloudCom, vol. 2017-Decem, pp. 194–201, 2017.

[17] N. Kyriakou and E. Loukis, “Cloud Computing Business Value and Human Determinants,” Proc. 21st Pan-Hellenic Conf. Informatics - PCI 2017, pp. 1–6, 2017.

[18] K. F. Li, H. Y. Wen, and Z. J. Yang, “Business logic structure of saas-based E-commerce system,” Proc. - 2010 IEEE 17th Int. Conf. Ind. Eng. Eng. Manag. IE EM2010, no. 72104994, pp. 316–319, 2010.

[19] P. P. Indonesia, S. Milwandhari, and P. P. Indonesia, “Risk Analysis of Cloud Computing in the Logistics Process,” pp. 8–12, 2020.

[20] Q. Lu et al., “A Unified Business-Driven Cloud Management Framework,” IEEE Trans. Serv. Comput., vol. 9, no. 6, pp. 872–882, 2016.

[21] V. S. Shah, “Evolving dynamics of heterogeneous saas applications and business process automation in real-time enterprises,” Towar. Digit. World Ind. X.0 - Proc. 29th Int. Conf. Int. Assoc. Manag. Technol. IAMOT 2020, no. September, pp. 47–61, 2020.

[22] J. A. Morales, H. Yasar, and A. Volkman, “Implementing DevOps practices in highly regulated environments,” ACM Int. Conf. Proceeding Ser., vol. Part F1477, 2018.

[23] S. Saito, “Understanding Key Business Processes for Business Process Outsourcing Transition,” Proc. - 2019 ACM/IEEE 14th Int. Conf. Glob. Softw. Eng. ICGSE 2019, pp. 35–39, 2019.

[24] H. M. Arthur, C. Ouyang, M. Adams, A. H. M. Hofstede, and Y. Yu, “Towards the design of a scalable business process management system architecture in the cloud,” 2018.