Service-Oriented Architecture Development as an Integrating Platform in the Tourist Area

Dmytro Uhryn[1][0000-0003-4858-4511], Vasyl Andrunyk[2][0000-0003-0697-7384], Lyubomyr Chyrun[3][0000-0002-9448-1751], Nataliya Antonyuk[4][0000-0002-6297-0737], Ivan Dyyak[5][0000-0001-5841-2604], Oleh Naum[6][0000-0001-8700-6998]

[1]Chernivtsi Philosophical and Legal Lyceum, Chernivtsi, Ukraine
[2]Lviv Polytechnic National University, Lviv, Ukraine
[3-5]Ivan Franko National University of Lviv, Lviv, Ukraine
[4]University of Opole, Opole, Poland
[6]Drohobych Ivan Franko State Pedagogical University, Drohobych, Ukraine

ugrund38@gmail.com[1], vasyl.a.andrunyk@lpnu.ua[2], Lyubomyr.Chyrun@lnu.edu.ua[3], nantonyk@yahoo.com[4], ivan.dyyak@lnu.edu.ua[5], oleh.naum@gmail.com[6]

Abstract. The peculiarities of the methodology of service-oriented architecture are determined. The adequacy of the application of the SOA mechanism for information systems of travel agencies and firms is analyzed. Unified process of Rational Unified Process for the tourism sphere and the evolution of the information system based on SOA in the tourism sphere with the inclusion of legacy software is used to build schemes of an automated system creation.

Keywords: Service-Oriented Architecture, Information System, Business Process, Data Integration, Tourism.

1 Introduction

Constant changes in the development of the tourist market and business processes make it difficult for the information services of tourist organizations - by means of automation to minimize the reaction time of the business of tourist organizations to any external or internal changes [1-4]. Reducing the time of development of new tourist services, introduction of new processes of tourist destinations and other innovations, provides tourist organizations, in particular travel agencies, support for competitiveness in the domestic and foreign markets [4-9]. However, the typical computing infrastructure of tourism organizations is characterized by the factual isolation of its application components, which in terms of business leads to delays in the execution of business processes, disruption of interaction between units and partners, impedes management and control, resulting in reduced efficiency of the organization as a whole. In general, the lack of integration calls into question the usefulness of information technology for the tourism.

In most cases, these problems arise from the lack of architectural standards in information technology [10-15]. Moreover, not only standards but also a set of components that are standardized are needed to overcome them. Just as necessary is the business process for execution environment through which the components will linked [16-21].

2 General Problem Formulation

Until the end of the last century, the approach is based on the management and analysis of content flows dominated the construction of information systems. However, over time, a process-oriented approach has emerged as more versatile and informative, in which the choice and modification of the information system structure, the composition of its components, and the main characteristics are made dependent on the process, depending on its construction and end-user characteristics. Such a change in the position of developers has caused a powerful trend in information systems in the theory, methods and technologies of systems development - modeling and management of business processes, based on which all known ERP-systems are created. Nowadays, the analysis application center is moving even more critical - towards manipulating the works, building the tools for generating workflows and managing them. Business process management and workflow management are not the same thing. The electronic business process determines what needs is done, including by establishing the relationship between output data and input data. A business process may include non-automated (or non-automated) components that are manually mastered, and it can use any kind of resources. The workflow describes how a particular result can achieved [22-28]. However, the relationship between the business process and the workflow is quite simple [1, 3, 29-36] and is illustrated by the scheme shown in Fig. 1 [1].

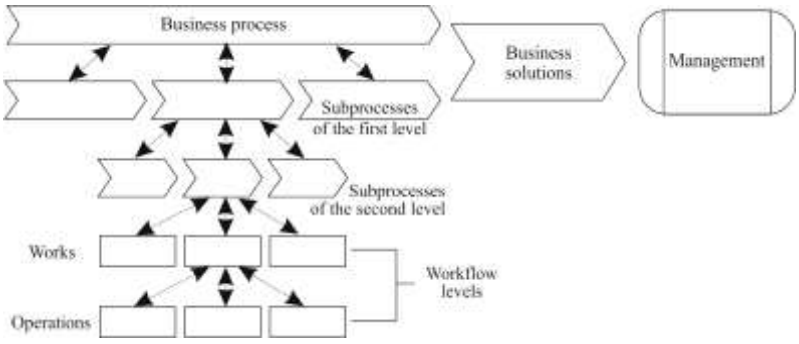

Fig. 1. Relationship between business processes and workflows [1].

As in the previous case, the change in the position of developers is explained by changes in the concept of building information systems. The transition nowadays to service-oriented architecture (SOA) means, first, the need to distinguish functional services – elements of SOA, from which business processes, are synthesized and modernized.

3 Analysis of Recent Research and Publications

OASIS (Organization for the Advancement of Structured Information Standards), which focuses on the standardization of SOA components, defines service-oriented architecture as the paradigm of distributed organizational and utilitarian capabilities that operate under the ownership of domains belonging to different owners. [1, 5, 37-43]. According to IBM definition [1, 6], SOA is an architectural style for creating an enterprise information architecture, based on a service orientation to achieve a closer relationship between business and supporting business systems. SOA introduces service orientation as an approach to business integration based on related services. Coupled with the ability to make rapid changes to processes, the presence of such mechanisms is considered the basis for the effectiveness of information systems based on SOA. Business tools are used to solve the problems, which in the general case will provide [1, 44-52]:

- Building new business functions;
- Establishing links between business functions from existing applications;
- Generating workflows to execute business processes.

In the most general form, SOA involves the presence of three main participants: the service provider - travel agent, the consumer service - tourist, the register of services (Fig. 2) [53-60].

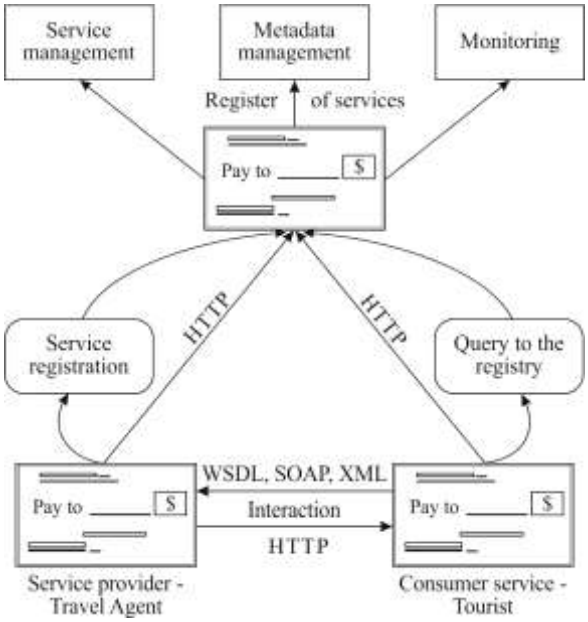

Fig. 2. General scheme of SOA in tourism

Participant interaction looks simple enough: the service provider registers its services in the registry, and the consumer requests the registry. The absence of any of these elements is unacceptable, and the addition of other components is not only possible but also necessary in practice. Such elements may include various middleware, monitoring the order and context of interaction, monitoring and managing services, and managing metadata and other supporting processes. SOA is an application architecture in which components or services, with agreed common interfaces, use a single rule (contract) to determine how to call services and how they will interact with each other. In this system, the concepts of "client" and "server" are situational absolutely. In one case, the application can act as a client and invoke an external service, and sometime after that, it can become a service provider when accessing another application for performing any tasks. If designed properly, SOA is capable of building and supporting customer-to-customer integration. When using SOA, travel organizations can generate flexibly and dynamically new services, combining the logic of existing applications and presenting them through reuse of services.

For information systems of travel agencies and firms, we highlight the goals of SOA:

- Reducing the cost of developing applications by streamlining the process of developing tourism services;
- Extension of code reuse;
- Independence from the platforms, tools, programming languages used;
- Increasing the scalability of the created systems in connection with the development of tourist service;
- Improving the manageability of the created information systems.

SOA principles:

- Architecture, as such, is not tied to any specific technology;
- Independence of organization of the system from the used computing platform;
- Independence of the organization of the system from the used programming languages;
- Use of services, independent of specific applications, with uniform interfaces for access to them.

Service-oriented Architecture is based on the following basic principles that help address the most pressing issues of application integration.

Weak connection. In terms of implementation, services in SOA are independent of each other: they perform certain actions from requests received from other services and return results. All the details of this execution are completely hidden: in the concept of SOA services are "black boxes". Therefore, changes in the implementation of the service will not affect the application component that the service uses, and vice versa.

Weak connectivity provides a simple and quick adaptation of the system to changes in the structure and principles of service delivery.

"Coarse-grained" structure of services. Services in SOA are high-level business logic modules, whereby the interaction between them is reduced to a limited number of messages in the content of business logic instead of many low-level calls that take into

account the details of the implementation of services. This approach reduces network load and contributes to higher system performance. Also important is that in terms of architecture, service, regardless of internal structure and language of implementation, looks like a whole.

Contracts. The concept of contract in SOA is interpreted somewhat more broadly than usual. The contract defines the conditions for the provision and use of services, including functional, technical (not only the protocol, but also, for example, the method of authorization) and other (cost of the permit, guarantees, insurance of the tourist) parameters and forms based on the agreement between the consumer and the service provider at all stages of it the interaction process. One of the common definitions of a provider and a consumer of services is based on who of them gives the contract (travel provider) and who receives (tourist). The contract allows dividing components into two parts - public, more stable, and hidden, one that is constantly changing. Because the way organization of components in SOA corresponds to the way business or computing is organized, the contract outlines not so much the software component as the business or computing functionality, dividing the service into a business (computing) interface and a business (computing) process.

Contracts not only weaken the links between elements of architecture, but, equally important, make them explicit, object-oriented objects of management.

Composition. The main advantage of the composition is the ability to reuse previously implemented functionality. However, the composition also contributes to adaptability, because it allows you to make many changes faster by changing the connections between services. Such changes are much easier than changing the whole service.

Dynamic linking. Relying on the stability of the contract, the consumer can find the necessary copies of services without interruption. This means that the system is resistant to changes not only in addressing available services (moving to another server), but also in their composition, which can change during operation. For example, new partners may emerge - travel agents, or the responsibility for providing the necessary business function may transferred to another provider, including an external one, but this will not affect the work of the consumer. Dynamic linking is supported by a registry that performs SOA accounting functions. Typically, it contains links to contracts, specific instances of services, information about their service providers, and sometimes their customers. All of this, in addition to other equally important functions, is also a configuration of SOA, which determines the parameters of the execution time of the services used for consumers of services (due to the composition of those who are often their providers). SOA is a set of interoperable design principles, and dynamic linking is often referred to as the most specific. Dynamic linking benefits are especially needed when operating large or decentralized information services or when interacting with a number of parties working on an identical or very similar contract. These can be affiliates of a travel company (for example, retail outlets - restaurants, hotels, sanatoriums, and so on), but more often they are partners - travel providers, travel agents, and more.

Scalability of interfaces. It is a good practice to leave the possibility of adding previously unforeseen data elements in the message transmission. Thus, a contract in advance allows for some flexibility if it is possible and necessary in a particular situation.

Reconstruction. Scalability is good but not good enough. The interface should always reflect the functionality as fully as possible, and scalability does not solve major change management issues. Therefore, eventually, the interface has to change, adapt to changing business requirements. What procedures are being launched, each company (in the person of the responsible information manager) determines for its organization, depending on many factors, not the last of which - a balance of logic and intuition in the style of decision making. More stable organizations are characterized by greater regulation of the change process, and less flexible ones.

4 Formation of Goals

The aim of the study is to increase the flexibility of information systems in the tourism industry with data, reduce the cost of application development, increase the speed of response to the constant changes in the requirements of the tourism business, as well as provide the necessary level of integration between information systems with the help of SOA - service-oriented architecture.

5 The System Architecture Development

A number of standards are used to successfully integrate systems based on Web services, the main ones being:

- BPEL is business process implementation language;
- WSDL is service description language;
- SOAP is messaging protocol;
- UDDI is universal directory format for searching and integrating Web services.

The composition of Web services based on elementary works is performed using Business Process Execution Language (BPEL) [1]. It emerged because of the merger of the Web Services Flow Language (WSFL) by IBM and the XLANG language by Microsoft. The language is based on XML notation [1].

The use of this language allows the formation and execution of workflows as a series of logical actions, including:

- Accepting a request to include work in the process;
- Rejection of request;
- Check the description, and if the parameters match, prepare a positive response to the request.

BPEL generates a workflow consisting of sequences of logical actions or active operations, each of which corresponds to its own square in the workflow diagram or function code [1]. There are two possible uses of BPEL:

1. Performed BPEL is a process that is also considered a service and can be an orchestration node [1]. Software products that implement executable BPEL processes are BPEL engines. Here one running process can include another, which gives the effect of including one orchestration (sequence of services) in another, as shown in Fig. 4.

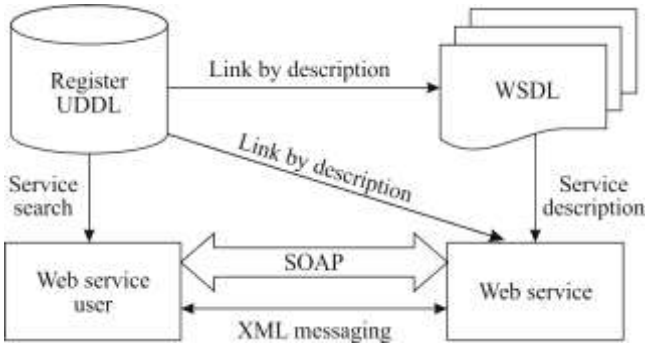

Fig. 3. Standardization of types of services

2. The abstract process is almost identical to the executable process except for the data filling [1]. In this capacity, it represents the logic of the business process and is used for the following purpose:
    - Defines the behavior of the elements of the organizational structure that supports the process;
    - A guide for programmers and developers who automate the process;
    - Considered an entry for commercial software that forms the structure of the system.

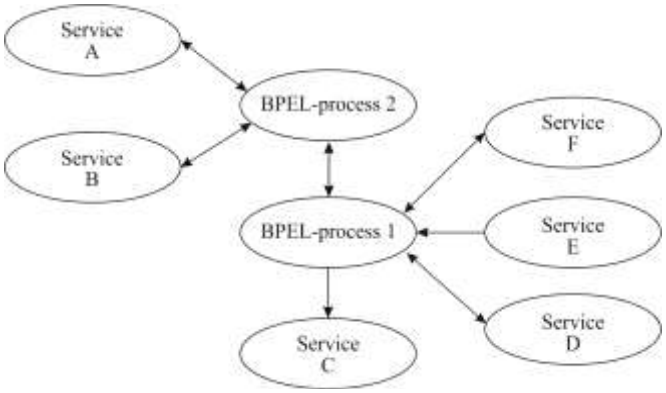

Fig. 4. Scheme of connected orchestration [1].

WSDL (Web Services Description Language) is an XML-based Web server-description language. Provides the right choice of services for transmission from provider to

consumer. WSDL information is used by service developers to integrate services into the system.

SOAP (Simple Object Access Protocol) is also an XML messaging protocol designed to transmit data from Web services. Auto-generated SOAP files include WSDL service description data.

UDDI is a set of rules for registering and retrieving information about available services. When developing programs, programmers can search the UDDI registry for the services they need to include in the programs. This registry may be necessary in the process of executing an application that needs data-providing services, such as the cost of travel services or permits.

Consider at the business processes that IBM has put in place that have introduced a unified process description - Rational Unified Process (RUP). Fig. 5 shown an example of creating an automated system using RUP and IBM software.

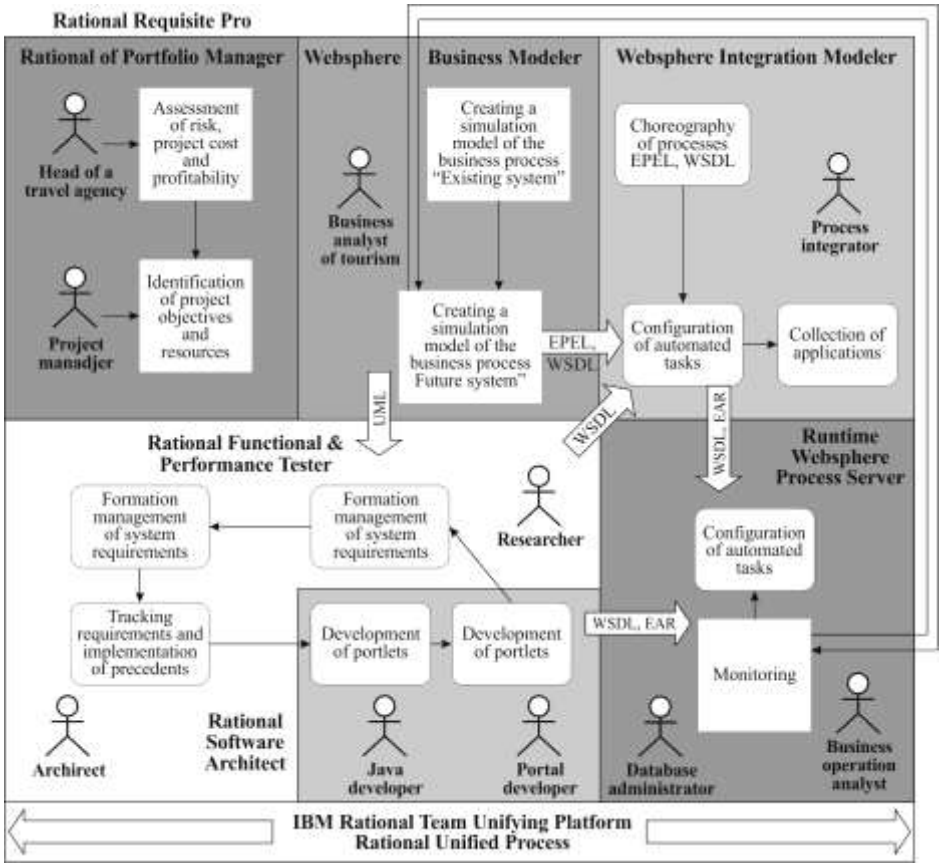

Fig. 5. Creating an automated system using a Rational Unified Process for the tourism industry

Fig.5 demonstrates interaction head of travel agency, project manager, risk assessment, project cost and profitability, creation of an imitation model of business process "existing system", creation of an imitation model of business process "future system", process choreography, configuration of non-automated tasks, collection of applications, forming and managing system requirements, testing, requirements tracking and implementation of precedents, designing, development of portlets, architect, researcher, Java developer, portal developer, monitoring, database administrator, business operations analyst. Below describe is a systematic description of the unified process of creating an information system.

Step 1. The Rational Portfolio Manager application helps you achieve top-level modeling alignment on metrics such as profitability (ROI), cost, and other business conditions.

Step 2. A business analyst develops a business process model using the WebSphere Business Modeler application.

Step 3. Rational Software Architect represents the business process in UML standard notation.

Step 4. Rational Software Architect converts the model in UML notation to Java code.

Step 5. IBM Rational Application Developer automates the process of creating new Web services based on existing resources, such as JavaBeans or EJB components, by automatically generating WSDL files describing Web services, SOAP descriptors, and a client to test Web services. For example, in a portlet form. Portlet is a standard portal component, which is designated for a user interface being build (Wikipedia). The portlet selects the snippets that are embedded in the portal page. You can think of a portal page as a collection of non-overlapping portlet windows, each displaying a portlet. A portlet is as a web server application hosted on a portal. Examples of portlets: e-mail, flight announcements, latest travel news and promotions. With existing portlet standards, developers can create portlets that are built into any standards-compliant portal.

Step 6. WebSphere Integration Developer implements a business process in BPEL format. A developer who integrates web services into a business process can extract a ready-made service from UDDI and incorporate it into the business process, or incorporate functions performed by people. The result is expanded as an application WebSphere Process Server.

When implementing projects that use process management, the project manager and team members use Rational Clearquest to support the development. This product automates task assignment and tracks progress, and is used to test and detect defects in the projected workflow. It is also used when editing, certifying, and verifying services.

IBM Rational Clearcase integrated with Eclipse is used for configuration management tasks.

Often, the argument against the transition to SOA is the claim that it is associated with a complete breakdown of the entire existing system, and therefore requires a much cost. In fact, the transition to SOA does not require a complete change to the existing information system. It uses technologies that ensure compliance with previously introduced standards, as well as heredity with respect to previously installed able-bodied software products and technical equipment. Fig 5 shows the generalized evolutionary

structure of SOA in tourism. Fig. 6 illustrates a typical situation where an SOA-based information system is built on existing software. As an example, such popular applications as SAP, SAS, office packages, as well as client software and applications inherited from previous stages of system development are shown.

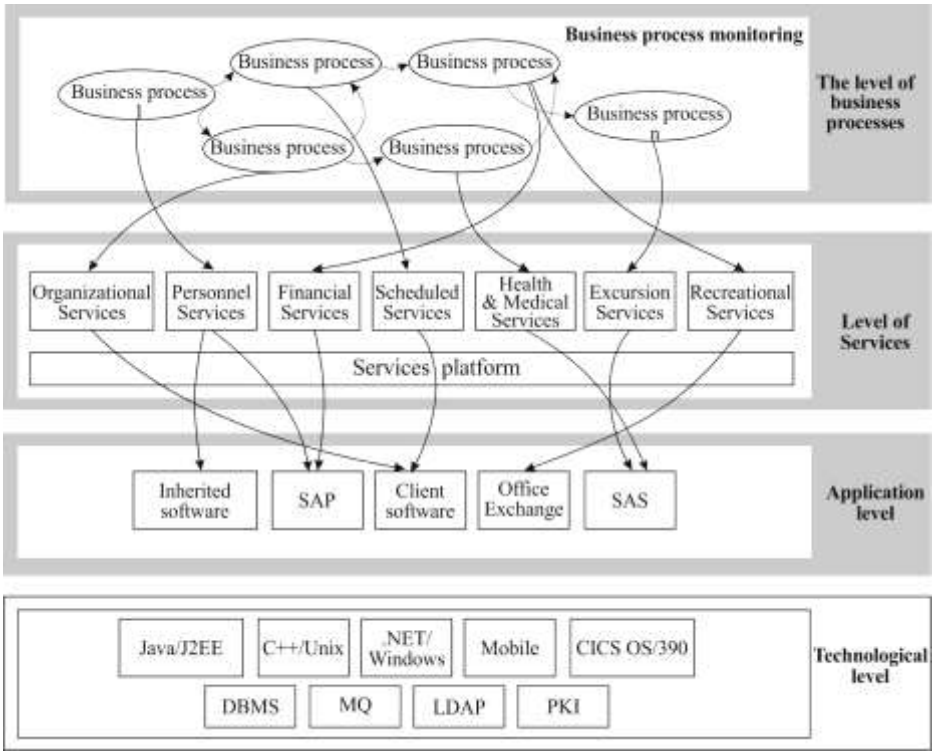

Fig. 6. An example of an evolutionary service-oriented architecture in tourism

Many analysts trace the history of SOA since the end of the last century, referring to the development of a functional approach, and later the concept of COBRA and tools Java. However, it did not become widespread because the architecture used at that time lacked a service level. Plural applications had different GUI, incompatible business process logic, different storage systems of data. All of this made the system difficult. The implementation of business processes is related directly the technological level and depended on the content speed.

Fig 6 demonstrates interaction of processes as monitoring of business processes, business process (x6), level of business processes, organizational services, personnel services, financial services, scheduled services, medical services, excursion services, leisure services, platform of services, level of services, software, SAP, application level, technological level

The transition to a modern SOA and the inclusion of an additional service layer in it forms a line of autonomous services, each of which is tied to the respective business

domain and platform of Web services, which allows carrying out their use largely, regardless of the downstream applications and technological level platform.

The introduction of the service level also creates convenient conditions for the construction of higher-level business processes and their functioning, namely [1]:

- The business services line reproduces a rough set of features that are ready to be incorporated into the business process;;
- The description (contract) of each service in the service line uniquely defines it, forming an interface. The presence of the latter allows you to create business processes independently and without involving knowledge of the technology platform;
- The registry and search tool maintains the dynamism and ability to change business processes by quickly accessing services when needed.;
- The data model implemented at the service level relies on the structure of the business domain and is independent of the data models of individual applications. Because XML is used as a canonical data exchange format, the inclusion of the service in the business process is also carried out regardless of the internal data structure in the applications;
- The service protection model operating within this layer provides universal control of the role distribution and connection of the service, authorizing the use of the service by some process. It also avoids the difficulty of interacting directly with application security tools, with their differences in organization principles, interfaces.

The service level management model generates statistics on the use of services, which is part of the monitoring of the top level business process (business processes).The lack of this level in previous generations of information systems created many problems that contributed to the rapid degradation of business processes, and this factor inhibited the spread of SOA.

Here is an example that recreates the possibility of transition to SOA in the tourism sphere through the evolution and improvement of the system, which does not require one-time high costs and complete destruction of the usual process of the system functioning. In this example, the transition to SOA is carried out in stages (Fig.7).

Fig. 7 shows the developing system, we give an example of two subsystems - personnel, finance, and analyze their databases. In the first stage of evolution, new services are being developed that are included in these subsystems. The following shows services built on inherited applications. These services retain the design features of the old system. To use them in a future system, they are combined into a common shell, which is presented as an interface designed already under the new logic of process implementation.

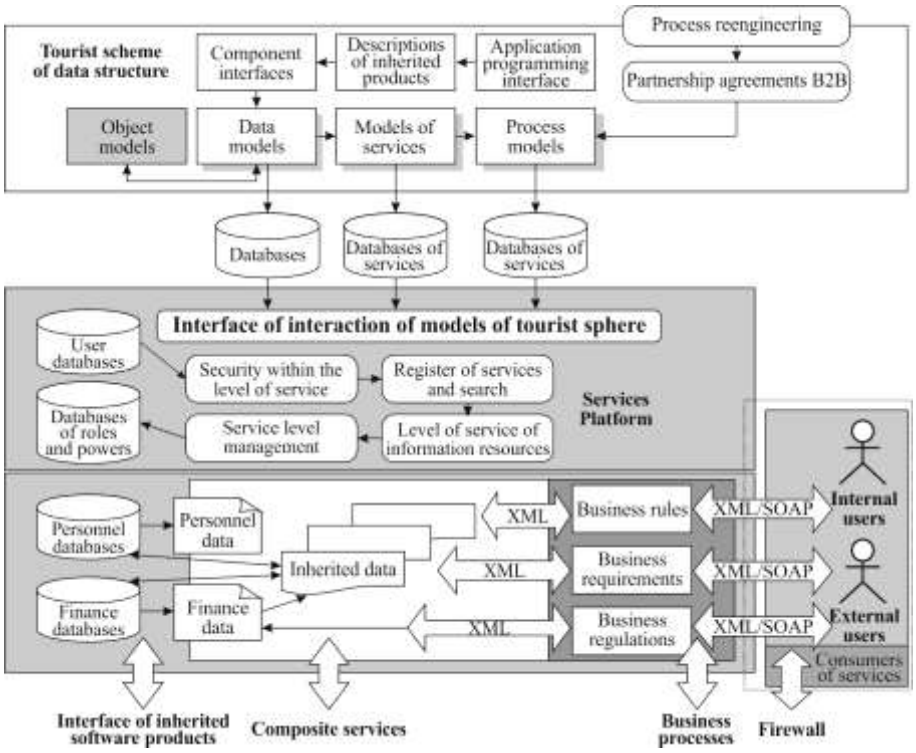

Fig. 7. Evolution of SOA-based information system in tourism with the inclusion of inherited software.

Fig.7 is tourist diagram of data structure, component interfaces, descriptions of inherited products, process reengineering, object models, data models, service models, model processes,

- Databases, services base, process bases, interface of interaction of models of tourist sphere,
- User database, security within the service level, service registry and search, platform of services, level of service of information services, service level management, database of roles and powers, database of frames, data of frames, data of finance, inherited data, business rules, business requirements, business order, the interface of the inherited software products, composite services, business processes, firewall

With the tools included in the service level (registry, service level management services, security, and business process building), a business process is built that integrates both new services and outdated software. The development of such a scheme can serve as the basis for organizing and planning the process of transition to SOA in the tourism sphere through the gradual development of the information system.

6 Conclusions

The article describes the methodology of service-oriented architecture to create information architecture of the tourism industry, which is based on a service orientation to build a closer relationship between business and supporting business systems. The application of SOA service orientation is proposed as an approach to the tourism business integration on the basis of interconnected services through the evolution and improvement of the system, which does not require one-time large expenses and complete destruction of the usual process of system functioning. The application of SOA service orientation is proposed as the tourism business integration on the basis of interconnected services through the evolution and improvement of the system, which does not require one-time large expenses and complete destruction of the usual process of system functioning.

References

1. Shendryk, V., Boiko, A.: Stages of Information System Development in the Process Approach", Procedia Computer Science (2015)
2. Su, J., Sachenko, A., Lytvyn, V., Vysotska, V., Dosyn, D.: Model of Touristic Information Resources Integration According to User Needs. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 113-116. (2018)
3. Mathias Weske. Business Process Management. Springer-Verlag, Berlin-Heidelberg, (2008).
4. Varetskyy, Y., Rusyn, B., Molga, A., Ignatovych, A.: A new method of fingerprint key protection of grid credential. In: Advances in Intelligent and Soft Computing, 84, 99-103. (2010)
5. Bobby Woolf Exploring IBM SOA. Technology&Practice. Maximum press (2009).
6. IBM Redbook. Patterns: Implementing an SOA Using an Enterprise Service Bus, http://www.ibm.com/Index.cfm?ArticleID=22084, last accessed 2020/05/01.
7. Kravets, P.: Game method for coalitions formation in multi-agent systems. In: International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 1-4. (2018)
8. IBM Redbook Patterns: Service-Oriented Architecture and Web Services, http://www.redbooks.ibm.com/abstracts/sg246303.html, last accessed 2020/05/01.
9. Berko, A.Y., Aliekseyeva, K.A.: Quality evaluation of information resources in web-projects. In: Actual Problems of Economics 136(10), 226-234. (2012)
10. SOA Development Using the IBM Rational Software Development Platform, http://www07.ibm.com/sg/soa/downloads/SOA_Development_using_Rational_Software, , last accessed 2020/05/01.
11. Lytvyn, V., Vysotska, V., Burov, Y., Demchuk, A.: Architectural ontology designed for intellectual analysis of e-tourism resources. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 335-338. (2018)
12. Antonyuk, N., Vysotsky, A., Vysotska, V., Lytvyn, V., Burov, Y., Demchuk, A., Lyudkevych, I., Chyrun, L., Chyrun, S., Bobyk, I.: Consolidated Information Web Resource for Online Tourism Based on Data Integration and Geolocation. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 15-20. (2019)

13. Vysotsky, A., Lytvyn, V., Vysotska, V., Dosyn, D., Lyudkevych, I., Antonyuk, N., Naum, O., Vysotskyi, A., Chyrun, L., Slyusarchuk, O.: Online Tourism System for Proposals Formation to User Based on Data Integration from Various Sources. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 92-97. (2019)
14. Shakhovska, N., Shakhovska, K., Fedushko, S.: Some Aspects of the Method for Tourist Route Creation. In: Advances in Artificial Systems for Medicine and Education II, 902, 527-537. (2019)
15. Antonyuk, N., Medykovskyy, M., Chyrun, L., Dverii, M., Oborska, O., Krylyshyn, M., Vysotsky, A., Tsiura, N., Naum, O.: Online Tourism System Development for Searching and Planning Trips with User's Requirements. In: Advances in Intelligent Systems and Computing IV, Springer Nature Switzerland AG 2020, 1080, 831-863. (2020)
16. Lozynska, O., Savchuk, V., Pasichnyk, V.: Individual Sign Translator Component of Tourist Information System. In: Advances in Intelligent Systems and Computing IV, Springer Nature Switzerland AG 2020, Springer, Cham, 1080, 593-601. (2020)
17. Savchuk, V., Lozynska, O., Pasichnyk, V.: Architecture of the Subsystem of the Tourist Profile Formation. In: Advances in Intelligent Systems and Computing, 871, 561-570. (2019)
18. Zhezhnych, P., Markiv, O.: Recognition of tourism documentation fragments from webpage posts. In: 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering, TCSET, 948-951. (2018)
19. Zhezhnych, P., Markiv, O.: Linguistic comparison quality evaluation of web-site content with tourism documentation objects. In: Advances in Intelligent Systems and Computing 689, 656-667. (2018)
20. Zhezhnych, P., Markiv, O.: A linguistic method of web-site content comparison with tourism documentation objects. In: International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 340-343. (2017)
21. Babichev, S.: An Evaluation of the Information Technology of Gene Expression Profiles Processing Stability for Different Levels of Noise Components. In: Data, 3 (4), art. no. 48. (2018)
22. Babichev, S., Durnyak, B., Pikh, I., Senkivskyy, V.: An Evaluation of the Objective Clustering Inductive Technology Effectiveness Implemented Using Density-Based and Agglomerative Hierarchical Clustering Algorithms. In: Advances in Intelligent Systems and Computing, 1020, 532-553. (2020)
23. Lytvyn, V., Vysotska, V.: Designing architecture of electronic content commerce system. In: Computer Science and Information Technologies. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 115-119. (2015)
24. Rusyn, B., Vysotska, V., Pohreliuk, L.: Model and architecture for virtual library information system. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 37-41. (2018)
25. Lytvyn, V., Kuchkovskiy, V., Vysotska, V., Markiv, O., Pabyrivskyy, V.: Architecture of system for content integration and formation based on cryptographic consumer needs. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 391-395. (2018)
26. Lytvyn, V., Vysotska, V., Demchuk, A., Demkiv, I., Ukhanska, O., Hladun, V., Kovalchuk, R., Petruchenko, O., Dzyubyk, L., Sokulska, N.: Design of the architecture of an intelligent system for distributing commercial content in the internet space based on SEO-technologies, neural networks, and Machine Learning. In: Eastern-European Journal of Enterprise Technologies, 2(2-98), 15-34. (2019)

27. Rzheuskyi, A., Kutyuk, O., Vysotska, V., Burov, Y., Lytvyn, V., Chyrun, L.: The Architecture of Distant Competencies Analyzing System for IT Recruitment. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 254-261. (2019)
28. Kazarian, A., Kunanets, N., Pasichnyk, V., Veretennikova, N., Rzheuskyi, A., Leheza, A., Kunanets, O.: Complex Information E-Science System Architecture based on Cloud Computing Model. In: CEUR Workshop Proceedings, Vol-2362, 366-377. (2019)
29. Lytvyn, V., Dosyn, D., Emmerich, M., Yevseyeva, I.: Content formation method in the web systems. In: CEUR Workshop Proceedings, 2136, 42-61. (2018)
30. Lytvyn, V., Rybchak, Z.: Design of airport service automation system. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 195-197. (2015)
31. Lytvyn, V.V.: An approach to intelligent agent construction for determining the group of bank risk basing on ontology. In: Actual Problems of Economics (7), 314-320. (2011)
32. Dosyn, D., Lytvyn, V., Kovalevych, V., Oborska, O., Holoshchuk, R.: Knowledge discovery as planning development in knowledgebase framework. In: Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET, 449-451. (2016)
33. Lypak, O.H., Lytvyn, V., Lozynska, O., (...), Rzheuskyi, A., Dosyn, D.: Formation of Efficient Pipeline Operation Procedures Based on Ontological Approach. In: Advances in Intelligent Systems and Computing, 871, 571-581. (2019)
34. Peleshchak, R., Lytvyn, V., Peleshchak, I., Olyvko, R., Korniak, J.: Decision making model based on neural network with diagonalized synaptic connections. In: Advances in Intelligent Systems and Computing, 853, 321-329. (2019)
35. Pasichnyk, V., Lytvyn, V., Kunanets, N., (...), Bolyubash, Y., Rzheuskyi, A.: Ontological approach in the formation of effective pipeline operation procedures. In: 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 80-83. (2018)
36. Lytvyn, V., Uhryn, D., Fityo, A.: Modeling of territorial community formation as a graph partitioning problem. In: Eastern-European Journal of Enterprise Technologies, 1(4), 47-52. (2016)
37. Lytvyn, V.: The similarity metric of scientific papers summaries on the basis of adaptive ontologies. In: Proceedings of 7th International Conference on Perspective Technologies and Methods in MEMS Design, MEMSTECH, 162. (2011)
38. Lytvyn, V.V., Tsmots, O.I.: The process of managerial decision making support within the early warning system. In: Actual Problems of Economics, 149(11), 222-229. (2013)
39. Kravets, P.: The control agent with fuzzy logic. In: Perspective Technologies and Methods in MEMS Design, MEMSTECH, 40-41. (2010)
40. Kravets, P.: The game method for orthonormal systems construction. In: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 296-298. (2007)
41. Kravets, P., Kyrkalo, R.: Fuzzy logic controller for embedded systems. In: International Conference on Perspective Technologies and Methods in MEMS Design, MEMSTECH, 58-59. (2009)
42. Kravets, P., Prodanyuk, O.: Game task of resource allocation. In: Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 437-438. (2009)
43. Kravets, P.: Game methods of the stochastic boundary problem solution. In: Perspective Technologies and Methods in MEMS Design, MEMSTECH, 71-74. (2007)
44. Kravets, P.: Adaptive method of pursuit game problem solution. In: Modern Problems of Radio Engineering, Telecommunications and Computer Science Proceedings of International Conference, TCSET, 62-65. (2006)

45. Kravets, P.: Game methods of construction of adaptive grid areas. In: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 513-516. (2003)
46. Shakhovska, N., Veres, O., Bolubash, Y., Bychkovska-Lipinska, L.: Data space architecture for Big Data managering. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 184-187. (2015)
47. Vysotska, V., Hasko, R., Kuchkovskiy, V.: Process analysis in electronic content commerce system. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 120-123. (2015)
48. Lytvyn, V., Vysotska, V., Veres, O., Rishnyak, I., Rishnyak, H.: The Risk Management Modelling in Multi Project Environment.. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 32-35. (2017)
49. Kanishcheva, O., Vysotska, V., Chyrun, L., Gozhyj, A.: Method of Integration and Content Management of the Information Resources Network. In: Advances in Intelligent Systems and Computing, 689, Springer, 204-216. (2018)
50. Vysotska, V., Fernandes, V.B., Emmerich, M.: Web content support method in electronic business systems. In: CEUR Workshop Proceedings, Vol-2136, 20-41. (2018)
51. Vysotska, V., Lytvyn, V., Burov, Y., Gozhyj, A., Makara, S.: The consolidated information web-resource about pharmacy networks in city. In: CEUR Workshop Proceedings, 239-255. (2018)
52. Gozhyj, A., Kalinina, I., Vysotska, V., Gozhyj, V.: The method of web-resources management under conditions of uncertainty based on fuzzy logic. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 343-346. (2018)
53. Gozhyj, A., Vysotska, V., Yevseyeva, I., Kalinina, I., Gozhyj, V.: Web Resources Management Method Based on Intelligent Technologies. In: Advances in Intelligent Systems and Computing, 871, 206-221. (2019)
54. Rzheuskyi, A., Kutyuk, O., Voloshyn, O., Kowalska-Styczen, A., Voloshyn, V., Chyrun, L., Chyrun, S., Peleshko, D., Rak, T.: The Intellectual System Development of Distant Competencies Analyzing for IT Recruitment. In: Advances in Intelligent Systems and Computing IV, Springer, Cham, 1080, 696-720. (2020)
55. Rusyn, B., Pohreliuk, L., Rzheuskyi, A., Kubik, R., Ryshkovets Y., Chyrun, L., Chyrun, S., Vysotskyi, A., Fernandes, V. B.: The Mobile Application Development Based on Online Music Library for Socializing in the World of Bard Songs and Scouts' Bonfires. In: Advances in Intelligent Systems and Computing IV, Springer, 1080, 734-756. (2020)
56. Vysotska, V., Rishnyak, I., Chyrun L.: Analysis and evaluation of risks in electronic commerce. In: CAD Systems in Microelectronics, 9th International Conference, 332-333. (2007)
57. Vysotska, V., Chyrun, L.: Analysis features of information resources processing. In: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 124-128. (2015)
58. Andrunyk, V., Chyrun, L., Vysotska, V.: Electronic content commerce system development. In: Proceedings of 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM 2015-February. (2015)
59. Alieksieieva, K., Berko, A., Vysotska, V.: Technology of commercial web-resource processing. In: Proceedings of 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM 2015-February. (2015)
60. Vysotska, V., Chyrun, L.: Methods of information resources processing in electronic content commerce systems. In: Proceedings of 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM 2015-February. (2015)